

new/usr/src/cmd/Adm/sun/Makefile

1

```
*****
2156 Tue Nov 25 12:56:23 2014
new/usr/src/cmd/Adm/sun/Makefile
5344 etc/ftusers breaks some incrementals
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #
21 #
22 # Copyright (c) 1990, 2010, Oracle and/or its affiliates. All rights reserved.
23 # Copyright 2010 Nexenta Systems, Inc. All rights reserved.
24 #

26 ETCFILES= iocctl.syscon passwd shadow motd
27 FTPDFILES= ftpusers
28 KVMFILES= README
29 SMBFILES= smbpasswd

31 include ../../Makefile.cmd

33 ROOTETCFILES= $(ETCFILES:%=$(ROOTETC)/%)
34 ROOTETCFTPDFILES= $(FTPDFILES:%=$(ROOTETCFTPD)/%)
35 ROOTUSRKVMFILES= $(KVMFILES:%=$(ROOTUSRKVM)/%)
36 ROOTVARSMBFILES= $(SMBFILES:%=$(ROOTVARSMB)/%)

38 FILEMODE= 0644

40 ROOTETCFTPUSERSLINK= $(ROOTETC)/ftusers

42 $(ROOTETC)/shadow := FILEMODE = 400
43 $(ROOTVARSMB)/smbpasswd := FILEMODE = 0400

46 .KEEP_STATE:

48 $(ROOTETCFTPUSERSLINK): $(ROOTETCFTPDFILES)
49 $(RM) $@; $(SYMLINK) ftpd/ftusers $@
49 $(SYMLINK) ftpd/ftusers $@

51 all: $(ETCFILES) $(KVMFILES) $(SMBFILES) $(FTPDFILES)

53 install: all $(ROOTETCFILES) $(ROOTETCFTPDFILES) $(ROOTUSRKVMFILES) $(ROOTVARSMB)

55 clean:
56 $(RM) $(ROOTETCFTPUSERSLINK)
```

new/usr/src/cmd/Adm/sun/Makefile

2

```
58 lint:

60 clobber:

62 motd: FRC
63 @-$(ECHO) "rebuilding motd"
64 @$ (RELEASE_BUILD)-$(ECHO) "The Illumos Project\tSunOS $(RELEASE)\t$(VERS
65 @$ (NOT_RELEASE_BUILD)-$(ECHO) "The Illumos Project\tSunOS $(RELEASE)\t$(
66 @$ (NOT_RELEASE_BUILD)-$(ECHO) $(DEV_CM) | sed -e "s/(@#)//" >> motd
67 @-$(CAT) release_info >> motd

69 smbpasswd:
70 $(TOUCH) smbpasswd

72 clean:

74 lint:

76 clobber:
77 $(RM) motd smbpasswd

79 FRC:
```

19601 Tue Nov 25 12:56:24 2014

new/usr/src/lib/libm/Makefile.com

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```

1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
14 #
15 #
16 LIBRARY      = libm.a
17 VERS         = .2
18 #
19 LIBMDIR      = $(SRC)/lib/libm
20 #
21 m9xsseOBJS_i386 = \
22     __fex_hdlr.o \
23     __fex_i386.o \
24     __fex_sse.o \
25     __fex_sym.o \
26     fex_log.o
27 #
28 m9xsseOBJS   = $(m9xsseOBJS_$(TARGET_ARCH))
29 #
30 m9xOBJS_amd64 = \
31     __fex_sse.o \
32     feprec.o
33 #
34 m9xOBJS_sparc = \
35     lrint.o \
36     lrintf.o \
37     lrintl.o \
38     lround.o \
39     lroundf.o \
40     lroundl.o
41 #
42 m9xOBJS_i386 = \
43     __fex_sse.o \
44     feprec.o \
45     lrint.o \
46     lrintf.o \
47     lrintl.o \
48     lround.o \
49     lroundf.o \
50     lroundl.o
51 #
52 #
53 # lrint.o, lrintf.o, lrintl.o, lround.o, lroundf.o & lroundl.o are 32-bit only
54 #
55 m9xOBJS      = \
56     $(m9xOBJS_$(TARGET_ARCH)) \

```

```

57     __fex_$(MACH).o \
58     __fex_hdlr.o \
59     __fex_sym.o \
60     fdim.o \
61     fdimf.o \
62     fdiml.o \
63     feexcept.o \
64     fenv.o \
65     feround.o \
66     fex_handler.o \
67     fex_log.o \
68     fma.o \
69     fmaf.o \
70     fmal.o \
71     fmax.o \
72     fmaxf.o \
73     fmaxl.o \
74     fmin.o \
75     fminf.o \
76     fminl.o \
77     frexp.o \
78     frexpf.o \
79     frexpl.o \
80     ldexp.o \
81     ldexpf.o \
82     ldexpl.o \
83     llrint.o \
84     llrintf.o \
85     llrintl.o \
86     llround.o \
87     llroundf.o \
88     llroundl.o \
89     modf.o \
90     modff.o \
91     modfl.o \
92     nan.o \
93     nanf.o \
94     nanl.o \
95     nearbyint.o \
96     nearbyintf.o \
97     nearbyintl.o \
98     nexttoward.o \
99     nexttowardf.o \
100    nexttowardl.o \
101    remquo.o \
102    remquof.o \
103    remquol.o \
104    round.o \
105    roundf.o \
106    roundl.o \
107    scalbln.o \
108    scalblnf.o \
109    scalblnl.o \
110    tgamma.o \
111    tgammaf.o \
112    tgammal.o \
113    trunc.o \
114    truncf.o \
115    trunclo.o
116 #
117 OBJM9XSSE   = $(m9xsseOBJS:%=pics/%)
118 #
119 COBJS_i386  = \
120     __libx_errno.o
121 #
122 COBJS_sparc = \

```

```

123      $(COBJS_i386) \
124      _TBL_atan.o \
125      _TBL_exp2.o \
126      _TBL_log.o \
127      _TBL_log2.o \
128      _TBL_tan.o \
129      _tan.o \
130      __tanf.o

132 #
133 # atan2pi.o and sincospi.o is for internal use only
134 #

136 COBJS_amd64 = \
137     _TBL_atan.o \
138     _TBL_exp2.o \
139     _TBL_log.o \
140     _TBL_log2.o \
141     __tan.o \
142     __tanf.o \
143     _TBL_tan.o \
144     copysign.o \
145     exp.o \
146     fabs.o \
147     fmod.o \
148     ilogb.o \
149     isnan.o \
150     nextafter.o \
151     remainder.o \
152     rint.o \
153     scalbn.o

155 COBJS_sparcv9 = $(COBJS_amd64)

157 COBJS      = \
158     $(COBJS_$(TARGET_ARCH)) \
159     _cos.o \
160     _lgamma.o \
161     _rem_pio2.o \
162     _rem_pio2m.o \
163     _sin.o \
164     _sincos.o \
165     _xpg6.o \
166     _lib_version.o \
167     _SVID_error.o \
168     _TBL_ipio2.o \
169     _TBL_sin.o \
170     acos.o \
171     acosh.o \
172     asin.o \
173     asinh.o \
174     atan.o \
175     atan2.o \
176     atan2pi.o \
177     atanh.o \
178     cbrt.o \
179     ceil.o \
180     cos.o \
181     cosh.o \
182     erf.o \
183     exp10.o \
184     exp2.o \
185     expm1.o \
186     floor.o \
187     gamma.o \
188     gamma_r.o \

```

```

189     hypot.o \
190     j0.o \
191     j1.o \
192     jn.o \
193     lgamma.o \
194     lgamma_r.o \
195     log.o \
196     log10.o \
197     logp.o \
198     log2.o \
199     logb.o \
200     matherr.o \
201     pow.o \
202     scalb.o \
203     siggam.o \
204     significand.o \
205     sin.o \
206     sincos.o \
207     sincospi.o \
208     sinh.o \
209     sqrt.o \
210     tan.o \
211     tanh.o

213 #
214 # LSARC/2003/658 adds isnanl
215 #
216 QOBJS_sparc      = \
217     _TBL_atanl.o \
218     _TBL_expl.o \
219     _TBL_expm1l.o \
220     _TBL_logl.o \
221     finitel.o \
222     isnanl.o

224 QOBJS_sparcv9   = $(QOBJS_sparc)

226 QOBJS_amd64     = \
227     finitel.o \
228     isnanl.o

230 #
231 # atan2pil.o, ieee_funcl.o, rndintl.o, sinpil.o, sincospil.o
232 # are for internal use only
233 #
234 # LSARC/2003/279 adds the following:
235 #      gammal.o      1
236 #      gammal_r.o   1
237 #      j0l.o        2
238 #      j1l.o        2
239 #      jnl.o        2
240 #      lgammal_r.o  1
241 #      scalbl.o     1
242 #      significandl.o 1
243 #
244 QOBJS      = \
245     $(QOBJS_$(TARGET_ARCH)) \
246     _cosl.o \
247     _lgammal.o \
248     _poly_libmq.o \
249     _rem_pio2l.o \
250     _sincosl.o \
251     _sinl.o \
252     _tanl.o \
253     _TBL_cosl.o \
254     _TBL_ipio2l.o \

```

```

255     _TBL_sinl.o \
256     _TBL_tanl.o \
257     acoshl.o \
258     acosl.o \
259     asinhl.o \
260     asinl.o \
261     atan2l.o \
262     atan2pil.o \
263     atanh1.o \
264     atanl.o \
265     cbrtl.o \
266     copysignl.o \
267     coshl.o \
268     cosl.o \
269     erfl.o \
270     expl0l.o \
271     exp2l.o \
272     expl.o \
273     expmil.o \
274     fabs1.o \
275     floorl.o \
276     fmodl.o \
277     gammal.o \
278     gammal_r.o \
279     hypotl.o \
280     ieee_funcl.o \
281     ilogbl.o \
282     j0l.o \
283     j1l.o \
284     jnl.o \
285     lgammal.o \
286     lgammal_r.o \
287     logl0l.o \
288     loglpl.o \
289     log2l.o \
290     logbl.o \
291     logl.o \
292     nextafterl.o \
293     powl.o \
294     remainderl.o \
295     rintl.o \
296     rndintl.o \
297     scalbl.o \
298     scalbnl.o \
299     signgaml.o \
300     significandl.o \
301     sincosl.o \
302     sincospil.o \
303     sinhl.o \
304     sinl.o \
305     sinpil.o \
306     sqrtl.o \
307     tanhl.o \
308     tanl.o

310 #
311 # LSARC/2003/658 adds isnanf
312 #
313 ROBJs_sparc = \
314     __cosf.o \
315     __sincosf.o \
316     __sinf.o \
317     isnanf.o

319 ROBJs_sparcv9 = $(ROBJs_sparc)

```

```

321 ROBJs_amd64 = \
322     isnanf.o \
323     __cosf.o \
324     __sincosf.o \
325     __sinf.o

327 #
328 # atan2pif.o, sincosf.o, sincospif.o are for internal use only
329 #
330 # LSARC/2003/279 adds the following:
331 #     besself.o      6
332 #     scalbf.o       1
333 #     gammaf.o       1
334 #     gammaf_r.o     1
335 #     lgammaf_r.o    1
336 #     significandf.o 1
337 #
338 ROBJs = \
339     $(ROBJs_$(TARGET_ARCH)) \
340     _TBL_r_atan_.o \
341     acosf.o \
342     acoshf.o \
343     asinf.o \
344     asinhf.o \
345     atan2f.o \
346     atan2pif.o \
347     atanf.o \
348     atanhf.o \
349     besself.o \
350     cbrtf.o \
351     copysignf.o \
352     cosf.o \
353     coshf.o \
354     erff.o \
355     expl0f.o \
356     exp2f.o \
357     expf.o \
358     expmlf.o \
359     fabsf.o \
360     floorf.o \
361     fmodf.o \
362     gammaf.o \
363     gammaf_r.o \
364     hypotf.o \
365     ilogbf.o \
366     lgammaf.o \
367     lgammaf_r.o \
368     log10f.o \
369     log1pf.o \
370     log2f.o \
371     logbf.o \
372     logf.o \
373     nextafterf.o \
374     powf.o \
375     remainderf.o \
376     rintf.o \
377     scalbf.o \
378     scalbnf.o \
379     signgamf.o \
380     significandf.o \
381     sinf.o \
382     sinh.o \
383     sincosf.o \
384     sincospif.o \
385     sqrtf.o \
386     tanf.o \

```

```

387          tanhf.o

389 #
390 # LSARC/2003/658 adds isnanf/isnanl
391 #

393 SOBJS_sparc    = \
394     copysign.o \
395     exp.o \
396     fabs.o \
397     fmod.o \
398     ilogb.o \
399     isnan.o \
400     nextafter.o \
401     remainder.o \
402     rint.o \
403     scalbn.o

405 SOBJS_i386     = \
406     __reduction.o \
407     finitef.o \
408     finitel.o \
409     isnanf.o \
410     isnanl.o \
411     $(SOBJS_sparc)

413 SOBJS_amd64    = \
414     __swapFLAGS.o
415 #
416 #     _xtoll.o \
417 #     _xtoull.o \

419 SOBJS          = \
420     $(SOBJS_$(TARGET_ARCH))

422 complexOBJS    = \
423     cabs.o \
424     cabsf.o \
425     cabsl.o \
426     cacos.o \
427     cacosf.o \
428     cacosh.o \
429     cacoshf.o \
430     cacoshl.o \
431     cacosl.o \
432     carg.o \
433     cargf.o \
434     cargl.o \
435     casin.o \
436     casinf.o \
437     casin.h.o \
438     casinhf.o \
439     casinhl.o \
440     casinl.o \
441     catan.o \
442     catanf.o \
443     catanh.o \
444     catanhf.o \
445     catanhl.o \
446     catanl.o \
447     ccos.o \
448     ccosf.o \
449     ccosh.o \
450     ccoshf.o \
451     ccoshl.o \
452     ccosl.o \

```

```

453     cexp.o \
454     cexpf.o \
455     cexpl.o \
456     cimag.o \
457     cimagf.o \
458     cimagl.o \
459     clog.o \
460     clogf.o \
461     clogl.o \
462     conj.o \
463     conjf.o \
464     conjl.o \
465     cpow.o \
466     cpowf.o \
467     cpowl.o \
468     cproj.o \
469     cprojf.o \
470     cprojl.o \
471     creal.o \
472     crealf.o \
473     creall.o \
474     csin.o \
475     csinf.o \
476     csinh.o \
477     csinhf.o \
478     csinhl.o \
479     csinl.o \
480     csqrt.o \
481     csqrtf.o \
482     csqrtl.o \
483     ctan.o \
484     ctanf.o \
485     ctanh.o \
486     ctanhf.o \
487     ctanhl.o \
488     ctanl.o \
489     k_atan2.o \
490     k_atan2l.o \
491     k_cexp.o \
492     k_cexpl.o \
493     k_clog_r.o \
494     k_clog_rl.o

496 OBJECTS        = $(COBJS) $(ROBJS) $(QOBS) $(SOBJS) $(m9xOBJS) $(complexOBJS)

498 include        $(SRC)/lib/Makefile.lib
499 include        $(LIBMDIR)/Makefile.libm.com
500 include        $(SRC)/lib/Makefile.rootfs

502 SRCDIR         = ../common/
503 LIBS           = $(DYNLIB) $(LINTLIB)

505 LINTERROFF     = -erroff=E_FUNC_SET_NOT_USED
506 LINTERROFF     += -erroff=E_FUNC_RET_ALWAYS_IGNORE2
507 LINTERROFF     += -erroff=E_FUNC_RET_MAYBE_IGNORED2
508 LINTERROFF     += -erroff=E_IMPL_CONV_RETURN
509 LINTERROFF     += -erroff=E_NAME_MULTIPLY_DEF2
510 LINTFLAGS      += $(LINTERROFF)
511 LINTFLAGS64    += $(LINTERROFF)
512 LINTFLAGS64    += -errchk=longptr64

514 CPPFLAGS       += -DLIBM_BUILD

514 CFLAGS         += $(C_BIGPICFLAGS)
515 CFLAGS64       += $(C_BIGPICFLAGS)

```

```

517 m9x_IL = $(LIBMDIR)/common/m9x/___fenv_$(TARGET_ARCH).il
519 SRCS_LD_i386_amd64 = \
520 ../common/LD/finitel.c \
521 ../common/LD/isnanl.c \
522 ../common/LD/nextafterl.c
524 SRCS_LD = \
525 $(SRCS_LD_i386_$(TARGET_ARCH)) \
526 ../common/LD/___cosl.c \
527 ../common/LD/___lgammal.c \
528 ../common/LD/___poly_libmq.c \
529 ../common/LD/___rem_pio2l.c \
530 ../common/LD/___sincosl.c \
531 ../common/LD/___sinl.c \
532 ../common/LD/___tanl.c \
533 ../common/LD/_TBL_cosl.c \
534 ../common/LD/_TBL_ipio2l.c \
535 ../common/LD/_TBL_sinl.c \
536 ../common/LD/_TBL_tanl.c \
537 ../common/LD/acoshl.c \
538 ../common/LD/asinhl.c \
539 ../common/LD/atan2pil.c \
540 ../common/LD/atanhl.c \
541 ../common/LD/cbrtl.c \
542 ../common/LD/coshl.c \
543 ../common/LD/cosl.c \
544 ../common/LD/erfl.c \
545 ../common/LD/gammal.c \
546 ../common/LD/gammal_r.c \
547 ../common/LD/hypotl.c \
548 ../common/LD/j0l.c \
549 ../common/LD/j1l.c \
550 ../common/LD/jnl.c \
551 ../common/LD/lgammal.c \
552 ../common/LD/lgammal_r.c \
553 ../common/LD/log1pl.c \
554 ../common/LD/logbl.c \
555 ../common/LD/scalbl.c \
556 ../common/LD/signgaml.c \
557 ../common/LD/significandl.c \
558 ../common/LD/sincosl.c \
559 ../common/LD/sincospil.c \
560 ../common/LD/sinh1.c \
561 ../common/LD/sinl.c \
562 ../common/LD/sinpil.c \
563 ../common/LD/tanh1.c \
564 ../common/LD/tanl.c
566 SRCS_LD_i386 = \
567 $(SRCS_LD)
569 SRCS_R_amd64 = \
570 ../common/R/___tanf.c \
571 ../common/R/isnanf.c \
572 ../common/R/___cosf.c \
573 ../common/R/___sincosf.c \
574 ../common/R/___sinf.c \
575 ../common/R/acosf.c \
576 ../common/R/asinf.c \
577 ../common/R/atan2f.c \
578 ../common/R/copysignf.c \
579 ../common/R/exp10f.c \
580 ../common/R/exp2f.c \
581 ../common/R/expmlf.c \
582 ../common/R/fabsf.c \

```

```

583 ../common/R/hypotf.c \
584 ../common/R/ilogbf.c \
585 ../common/R/log10f.c \
586 ../common/R/log2f.c \
587 ../common/R/nextafterf.c \
588 ../common/R/powf.c \
589 ../common/R/rintf.c \
590 ../common/R/scalbnf.c
592 # sparc + sparcv9
593 SRCS_R_sparc = \
594 ../common/R/___tanf.c \
595 ../common/R/___cosf.c \
596 ../common/R/___sincosf.c \
597 ../common/R/___sinf.c \
598 ../common/R/isnanf.c \
599 ../common/R/acosf.c \
600 ../common/R/asinf.c \
601 ../common/R/atan2f.c \
602 ../common/R/copysignf.c \
603 ../common/R/exp10f.c \
604 ../common/R/exp2f.c \
605 ../common/R/expmlf.c \
606 ../common/R/fabsf.c \
607 ../common/R/fmodf.c \
608 ../common/R/hypotf.c \
609 ../common/R/ilogbf.c \
610 ../common/R/log10f.c \
611 ../common/R/log2f.c \
612 ../common/R/nextafterf.c \
613 ../common/R/powf.c \
614 ../common/R/remainderf.c \
615 ../common/R/rintf.c \
616 ../common/R/scalbnf.c
618 SRCS_R = \
619 $(SRCS_R_$(MACH)) \
620 $(SRCS_R_$(TARGET_ARCH)) \
621 ../common/R/_TBL_r_atan_.c \
622 ../common/R/acoshf.c \
623 ../common/R/asinhf.c \
624 ../common/R/atan2pif.c \
625 ../common/R/atanf.c \
626 ../common/R/atanhf.c \
627 ../common/R/besself.c \
628 ../common/R/cbrtf.c \
629 ../common/R/cosf.c \
630 ../common/R/coshf.c \
631 ../common/R/erff.c \
632 ../common/R/expf.c \
633 ../common/R/floorf.c \
634 ../common/R/gammaf.c \
635 ../common/R/gammaf_r.c \
636 ../common/R/lgammaf.c \
637 ../common/R/lgammaf_r.c \
638 ../common/R/log1pf.c \
639 ../common/R/logbf.c \
640 ../common/R/logf.c \
641 ../common/R/scalbf.c \
642 ../common/R/signgamf.c \
643 ../common/R/significandf.c \
644 ../common/R/sinf.c \
645 ../common/R/sinhf.c \
646 ../common/R/sincosf.c \
647 ../common/R/sincospif.c \
648 ../common/R/sqrtf.c \

```

```

649 ../common/R/tanf.c \
650 ../common/R/tanhf.c \

652 SRCS_Q = \
653 ../common/Q/_TBL_atanl.c \
654 ../common/Q/_TBL_expl.c \
655 ../common/Q/_TBL_expm1l.c \
656 ../common/Q/_TBL_logl.c \
657 ../common/Q/_finitel.c \
658 ../common/Q/_isnarl.c \
659 ../common/Q/_cosl.c \
660 ../common/Q/_lgammal.c \
661 ../common/Q/_poly_libmq.c \
662 ../common/Q/_rem_pio2l.c \
663 ../common/Q/_sincosl.c \
664 ../common/Q/_sinl.c \
665 ../common/Q/_tanl.c \
666 ../common/Q/_TBL_cosl.c \
667 ../common/Q/_TBL_ipio2l.c \
668 ../common/Q/_TBL_sinl.c \
669 ../common/Q/_TBL_tanl.c \
670 ../common/Q/_acoshl.c \
671 ../common/Q/_acosl.c \
672 ../common/Q/_asinh1.c \
673 ../common/Q/_asinl.c \
674 ../common/Q/_atan2l.c \
675 ../common/Q/_atan2pil.c \
676 ../common/Q/_atanhl.c \
677 ../common/Q/_atanl.c \
678 ../common/Q/_cbrtl.c \
679 ../common/Q/_copysignl.c \
680 ../common/Q/_coshl.c \
681 ../common/Q/_cosl.c \
682 ../common/Q/_erfl.c \
683 ../common/Q/_expl0l.c \
684 ../common/Q/_exp2l.c \
685 ../common/Q/_expl.c \
686 ../common/Q/_expm1l.c \
687 ../common/Q/_fabs1.c \
688 ../common/Q/_floorl.c \
689 ../common/Q/_fmodl.c \
690 ../common/Q/_gammal.c \
691 ../common/Q/_gammal_r.c \
692 ../common/Q/_hypotl.c \
693 ../common/Q/_ieee_funcl.c \
694 ../common/Q/_ilogbl.c \
695 ../common/Q/_j0l.c \
696 ../common/Q/_j1l.c \
697 ../common/Q/_jn1.c \
698 ../common/Q/_lgammal.c \
699 ../common/Q/_lgammal_r.c \
700 ../common/Q/_log10l.c \
701 ../common/Q/_log1pl.c \
702 ../common/Q/_log2l.c \
703 ../common/Q/_logbl.c \
704 ../common/Q/_logl.c \
705 ../common/Q/_nextafterl.c \
706 ../common/Q/_powl.c \
707 ../common/Q/_remainderl.c \
708 ../common/Q/_rintl.c \
709 ../common/Q/_rndintl.c \
710 ../common/Q/_scalbl.c \
711 ../common/Q/_scalbnl.c \
712 ../common/Q/_signgaml.c \
713 ../common/Q/_significandl.c \
714 ../common/Q/_sincosl.c \

```

```

715 ../common/Q/_sincospil.c \
716 ../common/Q/_sinhl.c \
717 ../common/Q/_sinl.c \
718 ../common/Q/_sinpil.c \
719 ../common/Q/_sqrtl.c \
720 ../common/Q/_tanhl.c \
721 ../common/Q/_tanl.c \

723 SRCS_Q_sparc = \
724   $(SRCS_Q)

726 SRCS_complex = \
727   ../common/complex/cabs.c \
728   ../common/complex/cabsf.c \
729   ../common/complex/cabs1.c \
730   ../common/complex/cacos.c \
731   ../common/complex/cacosf.c \
732   ../common/complex/cacosh.c \
733   ../common/complex/cacoshf.c \
734   ../common/complex/cacosh1.c \
735   ../common/complex/cacos1.c \
736   ../common/complex/carg.c \
737   ../common/complex/cargf.c \
738   ../common/complex/carg1.c \
739   ../common/complex/casin.c \
740   ../common/complex/casinf.c \
741   ../common/complex/casinh.c \
742   ../common/complex/casinhf.c \
743   ../common/complex/casinh1.c \
744   ../common/complex/casin1.c \
745   ../common/complex/catan.c \
746   ../common/complex/catanf.c \
747   ../common/complex/catanh.c \
748   ../common/complex/catanhf.c \
749   ../common/complex/catanhl.c \
750   ../common/complex/catan1.c \
751   ../common/complex/ccos.c \
752   ../common/complex/ccosf.c \
753   ../common/complex/ccosh.c \
754   ../common/complex/ccoshf.c \
755   ../common/complex/ccosh1.c \
756   ../common/complex/ccosl.c \
757   ../common/complex/cexp.c \
758   ../common/complex/cexpf.c \
759   ../common/complex/cexpl.c \
760   ../common/complex/cimag.c \
761   ../common/complex/cimagf.c \
762   ../common/complex/cimag1.c \
763   ../common/complex/clog.c \
764   ../common/complex/clogf.c \
765   ../common/complex/clog1.c \
766   ../common/complex/conj.c \
767   ../common/complex/conjf.c \
768   ../common/complex/conj1.c \
769   ../common/complex/cpow.c \
770   ../common/complex/cpowf.c \
771   ../common/complex/cpowl.c \
772   ../common/complex/cproj.c \
773   ../common/complex/cprojf.c \
774   ../common/complex/cproj1.c \
775   ../common/complex/creal.c \
776   ../common/complex/crealf.c \
777   ../common/complex/creall.c \
778   ../common/complex/csin.c \
779   ../common/complex/csinf.c \
780   ../common/complex/csinh.c \

```

```

781 ../common/complex/csinhf.c \
782 ../common/complex/csinhl.c \
783 ../common/complex/csinl.c \
784 ../common/complex/csqrt.c \
785 ../common/complex/csqrtrf.c \
786 ../common/complex/csqrtrl.c \
787 ../common/complex/ctan.c \
788 ../common/complex/ctanf.c \
789 ../common/complex/ctanh.c \
790 ../common/complex/ctanhf.c \
791 ../common/complex/ctanhl.c \
792 ../common/complex/ctanl.c \
793 ../common/complex/k_atan2.c \
794 ../common/complex/k_atan2l.c \
795 ../common/complex/k_cexp.c \
796 ../common/complex/k_cexpl.c \
797 ../common/complex/k_clog_r.c \
798 ../common/complex/k_clog_rl.c

800 SRCS_m9x_i386 = \
801 ../common/m9x/__fex_sse.c \
802 ../common/m9x/feprec.c \
803 ../common/m9x/__fex_i386.c

805 SRCS_m9x_i386_i386 = \
806 ../common/m9x/lroundf.c

808 SRCS_m9x_i386_amd64 = \
809 ../common/m9x/llrint.c \
810 ../common/m9x/llrintf.c \
811 ../common/m9x/llrintl.c \
812 ../common/m9x/nexttowardl.c \
813 ../common/m9x/remquo.c \
814 ../common/m9x/remquof.c \
815 ../common/m9x/round.c \
816 ../common/m9x/roundl.c \
817 ../common/m9x/scalbln.c \
818 ../common/m9x/scalblnf.c \
819 ../common/m9x/scalblnl.c \
820 ../common/m9x/trunc.c \
821 ../common/m9x/truncl.c

823 # sparc
824 SRCS_m9x_sparc_sparc = \
825 ../common/m9x/lrint.c \
826 ../common/m9x/lrintf.c \
827 ../common/m9x/lrintl.c \
828 ../common/m9x/lround.c \
829 ../common/m9x/lroundf.c \
830 ../common/m9x/lroundl.c

832 SRCS_m9x_sparc = \
833 ../common/m9x/__fex_sparc.c \
834 ../common/m9x/llrint.c \
835 ../common/m9x/llrintf.c \
836 ../common/m9x/llrintl.c \
837 ../common/m9x/nexttowardl.c \
838 ../common/m9x/remquo.c \
839 ../common/m9x/remquof.c \
840 ../common/m9x/remquol.c \
841 ../common/m9x/round.c \
842 ../common/m9x/roundl.c \
843 ../common/m9x/scalbln.c \
844 ../common/m9x/scalblnf.c \
845 ../common/m9x/scalblnl.c \
846 ../common/m9x/trunc.c \

```

```

847 ../common/m9x/truncl.c

849 SRCS_m9x = \
850 $(SRCS_m9x_$(MACH)) \
851 $(SRCS_m9x_sparc_$(TARGET_ARCH)) \
852 $(SRCS_m9x_i386_$(TARGET_ARCH)) \
853 ../common/m9x/__fex_hdr.c \
854 ../common/m9x/__fex_sym.c \
855 ../common/m9x/fdim.c \
856 ../common/m9x/fdimf.c \
857 ../common/m9x/fdiml.c \
858 ../common/m9x/feexcept.c \
859 ../common/m9x/fenv.c \
860 ../common/m9x/feround.c \
861 ../common/m9x/fex_handler.c \
862 ../common/m9x/fex_log.c \
863 ../common/m9x/fma.c \
864 ../common/m9x/fmaf.c \
865 ../common/m9x/fmal.c \
866 ../common/m9x/fmax.c \
867 ../common/m9x/fmaxf.c \
868 ../common/m9x/fmaxl.c \
869 ../common/m9x/fmin.c \
870 ../common/m9x/fminf.c \
871 ../common/m9x/fminl.c \
872 ../common/m9x/frexp.c \
873 ../common/m9x/frexp.c \
874 ../common/m9x/frexp.c \
875 ../common/m9x/ldexp.c \
876 ../common/m9x/ldexp.c \
877 ../common/m9x/ldexpl.c \
878 ../common/m9x/llround.c \
879 ../common/m9x/llroundf.c \
880 ../common/m9x/llroundl.c \
881 ../common/m9x/modf.c \
882 ../common/m9x/modff.c \
883 ../common/m9x/modfl.c \
884 ../common/m9x/nan.c \
885 ../common/m9x/nanf.c \
886 ../common/m9x/nanl.c \
887 ../common/m9x/nearbyint.c \
888 ../common/m9x/nearbyintf.c \
889 ../common/m9x/nearbyintl.c \
890 ../common/m9x/nexttoward.c \
891 ../common/m9x/nexttowardf.c \
892 ../common/m9x/roundf.c \
893 ../common/m9x/tgamma.c \
894 ../common/m9x/tgammaf.c \
895 ../common/m9x/tgamma.c \
896 ../common/m9x/truncf.c

898 SRCS_C_sparc = \
899 ../common/C/__tan.c \
900 ../common/C/_TBL_atan.c \
901 ../common/C/_TBL_exp2.c \
902 ../common/C/_TBL_log.c \
903 ../common/C/_TBL_log2.c \
904 ../common/C/_TBL_tan.c \
905 ../common/C/acos.c \
906 ../common/C/asinh.c \
907 ../common/C/atan.c \
908 ../common/C/atan2.c \
909 ../common/C/ceil.c \
910 ../common/C/cos.c \
911 ../common/C/exp.c \
912 ../common/C/exp10.c \

```



```

913 ../common/C/exp2.c \
914 ../common/C/expm1.c \
915 ../common/C/floor.c \
916 ../common/C/fmod.c \
917 ../common/C/hypot.c \
918 ../common/C/ilogb.c \
919 ../common/C/isnan.c \
920 ../common/C/log.c \
921 ../common/C/log10.c \
922 ../common/C/log2.c \
923 ../common/C/pow.c \
924 ../common/C/remainder.c \
925 ../common/C/rint.c \
926 ../common/C/scalbn.c \
927 ../common/C/sin.c \
928 ../common/C/sincos.c \
929 ../common/C/tan.c

931 SRCS_i386_i386 = \
932 ../common/C/___libx_errno.c

934 SRCS_sparc_sparc = \
935 $(SRCS_i386_i386)

937 SRCS_sparc_sparcv9 = \
938 ../common/C/copysign.c \
939 ../common/C/fabs.c \
940 ../common/C/nextafter.c

942 SRCS_i386_amd64 = \
943 ../common/C/___TBL_atan.c \
944 ../common/C/___TBL_exp2.c \
945 ../common/C/___TBL_log.c \
946 ../common/C/___TBL_log2.c \
947 ../common/C/___tan.c \
948 ../common/C/___TBL_tan.c \
949 ../common/C/copysign.c \
950 ../common/C/exp.c \
951 ../common/C/fabs.c \
952 ../common/C/ilogb.c \
953 ../common/C/isnan.c \
954 ../common/C/nextafter.c \
955 ../common/C/rint.c \
956 ../common/C/scalbn.c \
957 ../common/C/acos.c \
958 ../common/C/asin.c \
959 ../common/C/atan.c \
960 ../common/C/atan2.c \
961 ../common/C/ceil.c \
962 ../common/C/cos.c \
963 ../common/C/exp10.c \
964 ../common/C/exp2.c \
965 ../common/C/expm1.c \
966 ../common/C/floor.c \
967 ../common/C/hypot.c \
968 ../common/C/log.c \
969 ../common/C/log10.c \
970 ../common/C/log2.c \
971 ../common/C/pow.c \
972 ../common/C/sin.c \
973 ../common/C/sincos.c \
974 ../common/C/tan.c

976 SRCS_C = \
977 $(SRCS_C_$(MACH)) \
978 $(SRCS_C_i386_$(TARGET_ARCH)) \

```

```

979 ../common/C/___cos.c \
980 ../common/C/___lgamma.c \
981 ../common/C/___rem_pio2.c \
982 ../common/C/___rem_pio2m.c \
983 ../common/C/___sin.c \
984 ../common/C/___sincos.c \
985 ../common/C/___xpg6.c \
986 ../common/C/___lib_version.c \
987 ../common/C/___SVID_error.c \
988 ../common/C/___TBL_ipio2.c \
989 ../common/C/___TBL_sin.c \
990 ../common/C/acosh.c \
991 ../common/C/asinh.c \
992 ../common/C/atan2pi.c \
993 ../common/C/atanh.c \
994 ../common/C/cbrt.c \
995 ../common/C/cosh.c \
996 ../common/C/erf.c \
997 ../common/C/gamma.c \
998 ../common/C/gamma_r.c \
999 ../common/C/j0.c \
1000 ../common/C/j1.c \
1001 ../common/C/jn.c \
1002 ../common/C/lgamma.c \
1003 ../common/C/lgamma_r.c \
1004 ../common/C/loglp.c \
1005 ../common/C/logb.c \
1006 ../common/C/matherr.c \
1007 ../common/C/scalb.c \
1008 ../common/C/signgam.c \
1009 ../common/C/significand.c \
1010 ../common/C/sincospi.c \
1011 ../common/C/sinh.c \
1012 ../common/C/sqrt.c \
1013 ../common/C/tanh.c

1015 SRCS = \
1016 $(SRCS_Q_$(MACH)) \
1017 $(SRCS_LD_$(MACH)) \
1018 $(SRCS_R) \
1019 $(SRCS_complex) \
1020 $(SRCS_C)

```

```
1022 .KEEP_STATE:
```

```
1024 all: $(LIBS)
```

```
1026 lint: lintcheck
```

new/usr/src/lib/libm/Makefile.libm.com

1

2668 Tue Nov 25 12:56:25 2014
new/usr/src/lib/libm/Makefile.libm.com
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
14 #
15 #
16 LIBMDIR = $(SRC)/lib/libm
17 #
18 LIBMSRC = $(LIBMDIR)/common
19 #
20 CPP_CMD = $(CC) -E -Xs
21 #
22 ASSUFFIX_sparc = S
23 ASSUFFIX_i386 = s
24 ASSUFFIX = $(ASSUFFIX_$(MACH))
25 #
26 # C99MODE of neither enabled nor disabled is "no_lib", whereby we expect
27 # C99-the-language, but don't modify the behaviour of library routines. This
28 # is VERY IMPORTANT, as -xc99=%all, for instance, would link us with
29 # values-xpg6, which would introduce an __xpg6 to our object with the C99
30 # flags set, causing us to default C99 libm behaviour on, breaking
31 # compatibility.
32 C99MODE =
33 #
34 M4FLAGS = -D_STDC -DPIC
35 M4FLAGS = -D_STDC -DELFOBJ -DPIC
36 #
37 LDBLDIR_sparc = Q
38 LDBLDIR_i386 = LD
39 LDBLDIR = $(LDBLDIR_$(MACH))
40 #
41 LM_IL = $(LIBMDIR)/$(TARGET_ARCH)/src/locallibm.il
42 #
43 CFLAGS += $(C_PICFLAGS) $(XSTRCONST) $(LM_IL)
44 CFLAGS64 += $(C_PICFLAGS) $(XSTRCONST) $(LM_IL)
45 CFLAGS += $(C_PICFLAGS) -D_INLINE $(XSTRCONST) $(LM_IL)
46 CFLAGS64 += $(C_PICFLAGS) -D_INLINE $(XSTRCONST) $(LM_IL)
47 sparc_CFLAGS += -Wa,-xarch=v8plus
48 #
49 CPPFLAGS += -I$(LIBMSRC)/C \
50 CPPFLAGS += -DELFOBJ \
51 -DLIBM_MT_FEX_SYNC \
52 -I$(LIBMSRC)/C \
53 -I$(LIBMSRC)/$(LDBLDIR) -I$(LIBMDIR)/$(TARGET_ARCH)/src
54 #
55 # GCC needs __C99FEATURES__ such that the implementations of isunordered,
56 # isgreaterequal, islessequal, etc, exist. This is basically equivalent to
```

new/usr/src/lib/libm/Makefile.libm.com

2

```
51 # providing no -xc99 to Studio, in that it gets us the C99 language features,
52 # but not values-xpg6, the reason for which is outline with C99MODE.
53 CFLAGS += -_gcc=-D__C99FEATURES__
54 CFLAGS64 += -_gcc=-D__C99FEATURES__
55 #
56 # libm depends on integer overflow characteristics
57 CFLAGS += -_gcc=-fno-strict-overflow
58 CFLAGS64 += -_gcc=-fno-strict-overflow
59 #
60 $(DYNLIB) := LDLIBS += -lc
61 #
62 $(LINTLIB) := SRCS = $(LIBMSRC)/$(LINTSRC)
63 #
64 CLEANFILES += pics/*.s pics/*.S
65 #
66 FPDEF_amd64 = -DARCH_amd64
67 FPDEF_sparc = -DCG89 -DARCH_v8plus -DFPADD_TRAPS_INCOMPLETE_ON_NAN
68 FPDEF_sparcv9 = -DARCH_v9 -DFPADD_TRAPS_INCOMPLETE_ON_NAN
69 FPDEF = $(FPDEF_$(TARGET_ARCH))
70 #
71 ASFLAGS = -P -D_ASM $(FPDEF)
72 #
73 XARCH_sparc = v8plus
74 XARCH_sparcv9 = v9
75 XARCH_i386 = f80387
76 XARCH_amd64 = amd64
77 XARCH = $(XARCH_$(TARGET_ARCH))
78 #
79 ASOPT_sparc = -xarch=$(XARCH) $(AS_PICFLAGS)
80 ASOPT_sparcv9 = -xarch=$(XARCH) $(AS_PICFLAGS)
81 ASOPT_i386 =
82 ASOPT_amd64 = -xarch=$(XARCH) $(AS_PICFLAGS)
83 ASOPT = $(ASOPT_$(TARGET_ARCH))
84 #
85 ASFLAGS += $(ASOPT)
86 #
87 CPPFLAGS_sparc = -DFPADD_TRAPS_INCOMPLETE_ON_NAN \
88 -DFDTOS_TRAPS_INCOMPLETE_IN_FNS_MODE
89 #
90 CPPFLAGS += $(CPPFLAGS_$(MACH))
91 ASFLAGS += $(CPPFLAGS)
```

new/usr/src/lib/libm/amd64/src/__swapFLAGS.s

1

```
*****
3690 Tue Nov 25 12:56:25 2014
new/usr/src/lib/libm/amd64/src/__swapFLAGS.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "__swapFLAGS.s"

31 #include "libm.h"
32 #include "libm_synonyms.h"

33 /*
34  * swap exception masks
35  *
36  * Put the complement of bits 5-0 of the argument into FPCW bits 5-0
37  * and MXCSR bits 12-7, return the complement of the previous FPCW
38  * bits 5-0.
39  */
40     ENTRY(__swapTE)
41     subq   $8,%rsp
42     fstcw  (%rsp)
43     movq   (%rsp),%rcx
44     movw   %cx,%ax
45     orw   $0x3f,%cx
46     andw  $0x3f,%di
47     xorw  %di,%cx
48     movw  %cx,(%rsp)
49     fldcw (%rsp)
50     stmxcsr (%rsp)
51     movq   (%rsp),%rcx
52     orw   $0x1f80,%cx
53     shlw  $7,%di
54     xorw  %di,%cx
55     movq  %rcx,(%rsp)
56     ldmxcsr (%rsp)
57     andq  $0x3f,%rax
```

new/usr/src/lib/libm/amd64/src/__swapFLAGS.s

2

```
58     xorq   $0x3f,%rax
59     addq   $8,%rsp
60     ret
61     .align 16
62     SET_SIZE(__swapTE)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/acosl.s

1

1679 Tue Nov 25 12:56:26 2014

new/usr/src/lib/libm/amd64/src/acosl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

29 .file "acosl.s"

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(acosl,function)
33 #include "libm_synonyms.h"
```

35 #undef fabs

```
34 ENTRY(acosl)
35 fldt 8(%rsp) / push x
36 fldl / push 1
37 fld %st(1) / x, 1, x
38 fabs / |x|, 1, x
39 fucomip %st(1),%st
40 ja 9f
41 fadd %st(1),%st / 1+x,x
42 fldz
43 fucomip %st(1),%st
44 jp .L1
45 jne .L1
46 / x is -1
47 fstp %st(0) / -1
48 fstp %st(0) / empty NPX stack
49 fldpi
50 ret
51 .L1:
52 fxch %st(1) / x,1+x
53 fldl / 1,x,1+x
54 fsubp %st,%st(1) / 1-x,1+x
55 fdivp %st,%st(1) / (1-x)/(1+x)
```

new/usr/src/lib/libm/amd64/src/acosl.s

2

```
56 fsqrt
57 fldl / 1,sqrt((1-x)/(1+x))
58 fpatan
59 fadd %st(0),%st
60 ret
61 9:
62 / |x| > 1
63 fstp %st(0) / x
64 fsub %st,%st(0) / +/-0 or NaN+invalid
65 fdiv %st,%st(0) / NaN+invalid or NaN
66 ret
67 .align 16
68 SET_SIZE(acosl)
unchanged_portion_omitted
```

new/usr/src/lib/libm/amd64/src/asinl.s

1

1535 Tue Nov 25 12:56:26 2014

new/usr/src/lib/libm/amd64/src/asinl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "asinl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(asinl,function)
```

```
33 #include "libm_synonyms.h"
```

```
35 #undef fabs
```

```
34 ENTRY(asinl)
35 fldt 8(%rsp) / push x
36 fldl / push 1
37 fld %st(1) / x, 1, x
38 fabs / |x|, 1, x
39 fucomip %st(1),%st
40 ja 9f
41 fadd %st(1),%st / 1+x,x
42 fldl / 1,1+x,x
43 fsub %st(2),%st / 1-x,1+x,x
44 fmulp %st,%st(1) / (1-x)*(1+x),x
45 fsqrt / sqrt((1-x)*(1+x)),x
46 fpatan / atan(x/sqrt((1-x)*(1+x)))
47 ret
48 9:
49 / |x| > 1
50 fstp %st(0) / x
51 fsub %st,%st(0) / +/-0 or NaN+invalid
52 fdiv %st,%st(0) / NaN+invalid or NaN
53 ret
54 .align 16
55 SET_SIZE(asinl)
```

new/usr/src/lib/libm/amd64/src/atan21.s

1

1203 Tue Nov 25 12:56:27 2014

new/usr/src/lib/libm/amd64/src/atan21.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file    "atan21.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(atan21,function)
33 #include "libm_synonyms.h"

34     ENTRY(atan21)
35     fldt    8(%rsp)          / push y
36     fldt    24(%rsp)         / push x
37     fpatan                          / return atan2(y,x)
38     ret
39     .align  16
40     SET_SIZE(atan21)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/atanl.s

1

1192 Tue Nov 25 12:56:27 2014

new/usr/src/lib/libm/amd64/src/atanl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "atanl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(atanl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(atanl)
```

```
35     fldt    8(%rsp)           / push arg
```

```
36     fldl                    / push 1.0
```

```
37     fpatan                    / atan(arg/1.0)
```

```
38     ret
```

```
39     .align 16
```

```
40     SET_SIZE(atanl)
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/copysignl.s

1

1261 Tue Nov 25 12:56:28 2014

new/usr/src/lib/libm/amd64/src/copysignl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "copysignl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(copysignl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(copysignl)
35     movl    16(%rsp),%eax
36     movl    32(%rsp),%ecx
37     andl    $0x7fff,%eax
38     andl    $0x8000,%ecx
39     orl     %ecx,%eax
40     movl    %eax,16(%rsp)
41     fldt   8(%rsp)
42     ret
43     .align 16
44     SET_SIZE(copysignl)
```

unchanged_portion_omitted


```

*****
3519 Tue Nov 25 12:56:28 2014
new/usr/src/lib/libm/amd64/src/exp101.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "exp101.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp101,function)
33 #include "libm_synonyms.h"

33     .data
34     .align 16
35 lt2_hi: .4byte 0xfbd00000, 0x9a209a84, 0x3ffd, 0x0
36 lt2_lo: .4byte 0x653f4837, 0x8677076a, 0xbfc9, 0x0

38     ENTRY(exp101)
39     movl 16(%rsp),%ecx      / cx <-- sign&bexp(x)
40     andl $0x7fff,%ecx     / ecx <-- zero_xtnd(bexp(x))
41     cmpl $0x3ffd,%ecx     / Is |x| < log10(2)?
42     jb   .shortcut       / If so, take a shortcut.
43     je   .check_tail     / maybe |x| only slightly < log10(2)
44 .general_case:
45     cmpl $0x7fff,%ecx     / Here, |x| > log10(2) or x is NaN
46     je   .not_finite     / bexp(|x|) = bexp(INF)?
47     cmpl $0x400e,%ecx     / if so, x is not finite
48     jb   .finite_non_special / |x| < 32768 = 2^15?
49     fldt 8(%rsp)         / if so, proceed with argument reduction
50     fldl %st(0)          / x
51     jmp 1f              / 1, x
52 .finite_non_special:
53     fldt 8(%rsp)         / Here, log10(2) < |x| < 2^15
54     fld  %st(0)          / x
55     fldl2t %st(0)        / x, x
56     fmulp                / log2(10), x, x
                          / z := x*log2(10), x

```

```

57     frndint              / [z], x
58     fst  %st(2)          / [z], x, [z]
59     PIC_SETUP(1)
60     fldt PIC_L(lt2_hi)   / lt2_hi, [z], x, [z]
61     fmulp                / [z]*lt2_hi, x, [z]
62     fsubrp %st,%st(1)    / x-[z]*lt2_hi, [z]
63     fldt PIC_L(lt2_lo)   / lt2_lo, x-[z]*lt2_hi, [z]
64     PIC_WRAPUP
65     fmul  %st(2),%st     / [z]*lt2_lo, x-[z]*lt2_hi, [z]
66     fsubrp %st,%st(1)    / r := x-[z]*log10(2), [z]
67     fldl2t                / log2(10), r, [z]
68     fmulp                / f := r*log2(10), [z]
69     f2xml                / 2^f-1, [z]
70     fldl                / 1, 2^f-1, [z]
71     faddp  %st,%st(1)    / 2^f, [z]
72 1:
73     fscale                / 10^x, [z]
74     fstp  %st(1)
75     ret

77 .check_tail:
78     movl 12(%rsp),%ecx   / ecx <-- hi_32(sgnfcnd(x))
79     cmpl $0x9a209a84,%ecx / Is |x| < log10(2)?
80     ja   .finite_non_special
81     jb   .shortcut
82     movl 8(%rsp),%edx    / edx <-- lo_32(sgnfcnd(x))
83     cmpl $0xfbcff798,%edx / Is |x| slightly > log10(2)?
84     ja   .finite_non_special / branch if |x| slightly > log10(2)
85 .shortcut:
86     / Here, |x| < log10(2), so |z| = |x/log10(2)| < 1
87     / whence z is in f2xml's domain.
88     fldt 8(%rsp)        / x
89     fldl2t                / log2(10), x
90     fmulp                / z := x*log2(10)
91     f2xml                / 2^z-1
92     fldl                / 1, 2^z-1
93     faddp  %st,%st(1)    / 10^x
94     ret

96 .not_finite:
97     movl 12(%rsp),%ecx   / ecx <-- hi_32(sgnfcnd(x))
98     cmpl $0x80000000,%ecx / hi_32(sgnfcnd(x)) = hi_32(sgnfcnd(INF))
99     jne .NaN_or_pinf     / if not, x is NaN or unsp.
100    movl 8(%rsp),%edx     / edx <-- lo_32(sgnfcnd(x))
101    cmpl $0,%edx          / lo_32(sgnfcnd(x)) = 0?
102    jne .NaN_or_pinf     / if not, x is NaN
103    movl 16(%rsp),%eax    / ax <-- sign&bexp(x)
104    andl $0x8000,%eax     / here, x is infinite, but +/-?
105    jz   .NaN_or_pinf     / branch if x = +INF
106    fldz                / Here, x = -inf, so return 0
107    ret

109 .NaN_or_pinf:
110    / Here, x = NaN or +inf, so load x and return immediately.
111    fldt 8(%rsp)
112    ret
113    .align 16
114    SET_SIZE(exp101)
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/amd64/src/exp21.s

1

2950 Tue Nov 25 12:56:29 2014

new/usr/src/lib/libm/amd64/src/exp21.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

29 .file "exp21.s"

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp21,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(exp21)
35 movl 16(%rsp),%ecx / cx <-- sign&bexp(x)
36 andl $0x7fff,%ecx / ecx <-- zero_xtnd(bexp(x))
37 cmpl $0x3fff,%ecx / Is |x| <= 1?
38 jb .shortcut / If so, take a shortcut.
39 je .check_tail / |x| may be slightly > 1
40 .general_case:
41 cmpl $0x7fff,%ecx / Here, |x| > 1 or x is NaN
42 je .not_finite / bexp(|x|) = bexp(INF)?
43 .finite_non_special:
44 fldt 8(%rsp) / if so, x is not finite
45 fld %st(0) / Here, 1 < |x| < INF
46 frndint / push arg
47 fucomi %st(1),%st / duplicate stack top
48 je .x_integral / [x],x
49 fxch / x integral?
50 fsub %st(1),%st / x-[x], [x]
51 f2xml / 2**(x-[x])-1, [x]
52 fldl / 1,2**(x-[x])-1, [x]
53 faddp %st,%st(1) / 2**(x-[x]), [x]
54 fscale / 2**x = 2**(arg), [x]
55 fstp %st(1)
56 ret
```

new/usr/src/lib/libm/amd64/src/exp21.s

2

```
58 .x_integral:
59 fstp %st(0) / ,x
60 fldl / l = 2**0, x
61 fscale / 2**(0 + x) = 2**x, x
62 fstp %st(1) / 2**x
63 ret

65 .check_tail:
66 movl 12(%rsp),%ecx / ecx <-- hi_32(sgnfcnd(x))
67 cmpl $0x80000000,%ecx / Is |x| <= 1?
68 ja .finite_non_special
69 movl 8(%rsp),%edx / edx <-- lo_32(sgnfcnd(x))
70 cmpl $0x0,%edx / Is |x| slightly > 1?
71 ja .finite_non_special / branch if |x| slightly > 1
72 .shortcut:
73 / Here, |x| < 1,
74 / whence x is in f2xml's domain.
75 fldt 8(%rsp) / push x
76 f2xml / 2**x - 1
77 fldl / 1,2**x - 1
78 faddp %st,%st(1) / 2**x
79 ret

81 .not_finite:
82 movl 12(%rsp),%ecx / ecx <-- hi_32(sgnfcnd(x))
83 cmpl $0x80000000,%ecx / hi_32(|x|) = hi_32(INF)?
84 jne .NaN_or_pinf / if not, x is NaN
85 movl 8(%rsp),%edx / edx <-- lo_32(x)
86 cmpl $0,%edx / lo_32(x) = 0?
87 jne .NaN_or_pinf / if not, x is NaN
88 movl 16(%rsp),%eax / ax <-- sign&bexp(x)
89 andl $0x8000,%eax / here, x is infinite, but +/-?
90 jz .NaN_or_pinf / branch if x = +INF
91 fldz / Here, x = -inf, so return 0
92 ret

94 .NaN_or_pinf:
95 / Here, x = NaN or +inf, so load x and return immediately.
96 fldt 8(%rsp)
97 ret
98 .align 16
99 SET_SIZE(exp21)
```

unchanged portion omitted

new/usr/src/lib/libm/amd64/src/expl.s

1

```
*****
3708 Tue Nov 25 12:56:29 2014
new/usr/src/lib/libm/amd64/src/expl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "expl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expl,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 16
36 ln2_hi: .4byte 0xd1d00000, 0xb17217f7, 0x3ffe, 0x0
37 ln2_lo: .4byte 0x4c67fc0d, 0x8654361c, 0xbfce, 0x0

39     ENTRY(expl)
40     movl 16(%rsp),%ecx      / cx <-- sign&bexp(x)
41     andl $0x7fff,%ecx      / ecx <-- zero_xtnd(bexp(x))
42     cmpl $0x3ffe,%ecx      / Is |x| < 0.5?
43     jb   .check_tail       / If so, see which shortcut to take
44     je   .general_case     / More checking if 0.5 <= |x| < 1
45 .general_case:
46     cmpl $0x7fff,%ecx      / Here, |x| >= 1 or x is NaN
47     je   .not_finite       / bexp(|x|) = bexp(INF)?
48     cmpl $0x400e,%ecx      / if so, x is not finite
49     jb   .finite_non_special / |x| < 32768 = 2^15?
50     fldt 8(%rsp)           / if so, proceed with argument reduction
51     fldl %st(0)            / x
52     jmp  1f                / 1, x
53 .finite_non_special:
54     fldt 8(%rsp)           / Here, ln(2) < |x| < 2^15
55     fld  %st(0)            / x
56     fldl2e %st(0)         / x, x
57     fmulp                / z := x*log2(e), x
```

new/usr/src/lib/libm/amd64/src/expl.s

2

```
58     frndint                / [z], x
59     fst  %st(2)            / [z], x, [z]
60     PIC_SETUP(1)
61     fldt PIC_L(ln2_hi)     / ln2_hi, [z], x, [z]
62     fmulp                / [z]*ln2_hi, x, [z]
63     fsubrp %st,%st(1)      / x-[z]*ln2_hi, [z]
64     fldt PIC_L(ln2_lo)     / ln2_lo, x-[z]*ln2_hi, [z]
65     PIC_WRAPUP
66     fmul %st(2),%st        / [z]*ln2_lo, x-[z]*ln2_hi, [z]
67     fsubrp %st,%st(1)      / r := x-[z]*ln(2), [z]
68     fldl2e                / log2(e), r, [z]
69     fmulp                / f := r*log2(e), [z]
70     f2xml                / 2^f-1, [z]
71     fldl                / 1, 2^f-1, [z]
72     faddp %st,%st(1)      / 2^f, [z]
73 1:
74     fscale                / e^x, [z]
75     fstp %st(1)
76     ret

78 2:
79     cmpl $0x3fbe,%ecx      / Here, |x| < 0.5
80     jae .shortcut         / Is |x| >= 2^-65?
81     fldt 8(%rsp)          / If so, take a shortcut
82     fldl                / x
83     faddp %st,%st(1)      / 1, x
84     ret                    / 1+x (for inexact & directed rounding)

86 .check_tail:
87     movl 12(%rsp),%ecx      / ecx <-- hi_32(sgnfncnd(x))
88     cmpl $0xb17217f7,%ecx  / Is |x| < ln(2)?
89     ja   .finite_non_special
90     jb   .shortcut
91     movl 8(%rsp),%edx       / edx <-- lo_32(x)
92     cmpl $0xd1cf79ab,%edx  / Is |x| slightly < ln(2)?
93     ja   .finite_non_special / branch if |x| slightly > ln(2)
94 .shortcut:
95     / Here, |x| < ln(2), so |z| = |x/ln(2)| < 1,
96     / whence z is in f2xml's domain.
97     fldt 8(%rsp)          / x
98     fldl2e                / log2(e), x
99     fmulp                / x*log2(e)
100    f2xml                / 2^(x*log2(e))-1 = e^x-1
101    fldl                / 1, e^x-1
102    faddp %st,%st(1)      / e^x
103    ret

105 .not_finite:
106    movl 12(%rsp),%ecx      / ecx <-- hi_32(sgnfncnd(x))
107    cmpl $0x80000000,%ecx   / hi_32(|x|) = hi_32(INF)?
108    jne  .NaN_or_pinf      / if not, x is NaN
109    movl 8(%rsp),%edx       / ecx <-- lo_32(x)
110    cmpl $0,%edx           / lo_32(x) = 0?
111    jne  .NaN_or_pinf      / if not, x is NaN
112    movl 16(%rsp),%eax      / ax <-- sign&bexp((x))
113    andl $0x8000,%eax      / here, x is infinite, but +/-?
114    jz   .NaN_or_pinf      / branch if x = +INF
115    fldz                / Here, x = -inf, so return 0
116    ret

118 .NaN_or_pinf:
119    / Here, x = NaN or +inf, so load x and return immediately.
120    fldt 8(%rsp)
121    fadd %st(0),%st        / quiet SNaN
122    ret
123    .align 16
```

new/usr/src/lib/libm/amd64/src/expl.s

3

124 SET_SIZE(expl)
 unchanged_portion_omitted

```

*****
3719 Tue Nov 25 12:56:30 2014
new/usr/src/lib/libm/amd64/src/expm11.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "expm11.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expm11,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 16
36 ln2_hi: .4byte 0xd1d00000, 0xb17217f7, 0x3ffe, 0x0
37 ln2_lo: .4byte 0x4c67fc0d, 0x8654361c, 0xbfce, 0x0

39     ENTRY(expm11)
40     movl 16(%rsp),%ecx      / cx <-- sign&bexp(x)
41     movl %ecx,%eax        / ax <-- sign&bexp(x)
42     andl $0x7fff,%ecx     / ecx <-- zero_xtnd(bexp(x))
43     cmpl $0x3ffe,%ecx     / Is |x| < ln(2)?
44     jb   .shortcut       / If so, take a shortcut.
45     je   .check_tail     / |x| may be only slightly < ln(2)
46 .general_case:
47     cmpl $0x7fff,%ecx     / Here, |x| > ln(2) or x is NaN
48     je   .not_finite     / bexp(|x|) = bexp(INF)?
49     andl $0xffff,%eax     / if so, x is not finite
50     cmpl $0xc006,%eax     / x <= -128?
51     jae 1f               / if so, simply return -1
52     cmpl $0x400d,%ecx     / |x| < 16384 = 2^14?
53     jb   .finite_non_special / if so, proceed with argument reduction
54     fldt 8(%rsp)         / x >= 16384; x
55     fldl 1, x
56     fscale                / +Inf, x
57     fstp %st(1)         / +Inf

```

```

58     ret

60 .finite_non_special:
61     fldt 8(%rsp)         / -128 < x < -ln(2) || ln(2) < x < 2^14
62     fld  %st(0)         / x
63     fldl2e                / log2(e), x, x
64     fmulp                / z := x*log2(e), x
65     frndint              / [z], x
66     fst  %st(2)         / [z], x, [z]
67     PIC_SETUP(1)
68     fldt PIC_L(ln2_hi)   / ln2_hi, [z], x, [z]
69     fmulp                / [z]*ln2_hi, x, [z]
70     fsubrp %st,%st(1)    / x-[z]*ln2_hi, [z]
71     fldt PIC_L(ln2_lo)   / ln2_lo, x-[z]*ln2_hi, [z]
72     PIC_WRAPUP
73     fmul %st(2),%st      / [z]*ln2_lo, x-[z]*ln2_hi, [z]
74     fsubrp %st,%st(1)    / r := x-[z]*ln(2), [z]
75     fldl2e                / log2(e), r, [z]
76     fmulp                / f := r*log2(e), [z]
77     f2xml                / 2^f-1, [z]
78     fldl 1, 2^f-1, [z]
79     faddp %st,%st(1)     / 2^f, [z]
80     fscale                / e^x, [z]
81     fstp %st(1)         / e^x
82     fldl 1, e^x
83     fsubrp %st,%st(1)    / e^x-1
84     ret

86 .check_tail:
87     movl 12(%rsp),%ecx   / ecx <-- hi_32(sgnfcnd(x))
88     cmpl $0xb17217f7,%ecx / Is |x| < ln(2)?
89     ja   .finite_non_special
90     jb   .shortcut
91     movl 8(%rsp),%edx    / edx <-- lo_32(x)
92     cmpl $0xd1cf79ab,%edx / Is |x| slightly < ln(2)?
93     ja   .finite_non_special / branch if |x| slightly > ln(2)
94 .shortcut:
95     / Here, |x| < ln(2), so |z| = |x/ln(2)| < 1,
96     / whence z is in f2xml's domain.
97     fldt 8(%rsp)         / x
98     fldl2e                / log2(e), x
99     fmulp                / z := x*log2(e)
100    f2xml                / 2^(x*log2(e))-1 = e^x-1
101    ret

103 .not_finite:
104    movl 12(%rsp),%ecx   / ecx <-- hi_32(sgnfcnd(x))
105    cmpl $0x80000000,%ecx / hi_32(|x|) = hi_32(INF)?
106    jne .NaN_or_pinf     / if not, x is NaN
107    movl 8(%rsp),%edx    / edx <-- lo_32(x)
108    cmpl $0,%edx        / lo_32(x) = 0?
109    jne .NaN_or_pinf     / if not, x is NaN
110    movl 16(%rsp),%eax   / ax <-- sign&bexp(x)
111    andl $0x8000,%eax   / here, x is infinite, but +/-?
112    jz   .NaN_or_pinf     / branch if x = +INF
113 1:
114    fldl                / Here, x = -inf, so return -1
115    fchs
116    ret

118 .NaN_or_pinf:
119    / Here, x = NaN or +inf, so load x and return immediately.
120    fldt 8(%rsp)
121    ret
122    .align 16
123    SET_SIZE(expm11)

```

new/usr/src/lib/libm/amd64/src/fabsl.s

1

1141 Tue Nov 25 12:56:30 2014

new/usr/src/lib/libm/amd64/src/fabsl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file      "fabsl.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fabsl,function)
33 #include "libm_synonyms.h"

34     ENTRY(fabsl)
35     fldt      8(%rsp)
36 #undef fabs
37     fabs
38     ret
39     .align 16
40     SET_SIZE(fabsl)
41
42     _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/floor1.s

1

1869 Tue Nov 25 12:56:30 2014

new/usr/src/lib/libm/amd64/src/floor1.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "floor1.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ceil, function)
33 LIBM_ANSI_PRAGMA_WEAK(floor1, function)
34 #include "libm_synonyms.h"
```

```
35     ENTRY(ceil)
36     subq   $16, %rsp
37     fstcw  (%rsp)
38     fldt   24(%rsp)
39     movw   (%rsp), %cx
40     orw   $0x0c00, %cx
41     xorw   $0x0400, %cx
42     movw   %cx, 4(%rsp)
43     fldcw  4(%rsp)           / set RD = up
44     frndint
45     fstcw  4(%rsp)           / restore RD
46     movw   4(%rsp), %dx
47     andw   $0xf3ff, %dx
48     movw   (%rsp), %cx
49     andw   $0x0c00, %cx
50     orw   %dx, %cx
51     movw   %cx, (%rsp)
52     fldcw  (%rsp)           / restore RD
53     addq   $16, %rsp
54     ret
55     .align 16
56     SET_SIZE(ceil)
```

unchanged portion omitted

new/usr/src/lib/libm/amd64/src/fmod.s

1

```
*****
1754 Tue Nov 25 12:56:31 2014
new/usr/src/lib/libm/amd64/src/fmod.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmod.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmod,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

35     ENTRY(fmod)
36     push    %rbp
37     movq   %rsp,%rbp
38     subq   $16,%rsp
39     movlpd %xmm1,-16(%rbp)
40     movlpd %xmm0,-8(%rbp)

42     movl   -12(%rbp),%eax    / eax <-- hi_32(y)
43     andl   $0x7fffffff,%eax / eax <-- hi_32(|y|)
44     orl   -16(%rbp),%eax    / eax <-- lo_32(y)|hi_32(|y|)
45     je     .yzero

47     fldl   -16(%rbp)        / y
48     fldl   -8(%rbp)         / x
49 .loop:
50     fprem                                / partial remainder
51     fstsw  %ax               / store status word
52     andw  $0x400,%ax         / check for incomplete reduction
53     jne   .loop              / loop while reduction incomplete
54     fstpl -8(%rbp)
55     movsd -8(%rbp),%xmm0
56     fstp  %st(0)
57     leave
```

new/usr/src/lib/libm/amd64/src/fmod.s

2

```
58     ret

60 .yzero:
61     PIC_SETUP(1)
62     movl   $27,%edi
63     movl   $2,%eax
64     call  PIC_F(_SVID_libm_err)
65     PIC_WRAPUP
66     leave
67     ret
68     .align 4
69     SET_SIZE(fmod)
unchanged_portion_omitted
```


new/usr/src/lib/libm/amd64/src/fmodf.s

1

1496 Tue Nov 25 12:56:31 2014

new/usr/src/lib/libm/amd64/src/fmodf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmodf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmodf,function)
33 #include "libm_synonyms.h"

34     ENTRY(fmodf)
35     push    %rbp
36     movq   %rsp,%rbp
37     subq   $16,%rsp
38     movss %xmm1,-8(%rbp)
39     movss %xmm0,-4(%rbp)
40     flds  -8(%rbp)           / load arg y
41     flds  -4(%rbp)           / load arg x
42 .loop:
43     fprem                               / partial remainder
44     fstsw  %ax                 / store status word
45     andw  $0x400,%ax           / check whether reduction complete
46     jne   .loop                / loop while reduction incomplete
47     fstps -4(%rbp)
48     movss -4(%rbp),%xmm0
49     fstp  %st(0)
50     leave
51     ret
52     .align 4
53     SET_SIZE(fmodf)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/fmodl.s

1

1367 Tue Nov 25 12:56:32 2014

new/usr/src/lib/libm/amd64/src/fmodl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmodl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmodl,function)
33 #include "libm_synonyms.h"

34     ENTRY(fmodl)
35     fldt    24(%rsp)        / load arg y
36     fldt    8(%rsp)        / load arg x
37 .mod_loop:
38     fprem                    / partial fmod
39     fstsw   %ax              / store status word
40     andw   $0x400,%ax       / check for incomplete reduction
41     jne    .mod_loop        / while incomplete, do fprem again
42     fstp   %st(1)
43     ret
44     .align 16
45     SET_SIZE(fmodl)

unchanged portion omitted
```

new/usr/src/lib/libm/amd64/src/ieee_funcl.s

1

3181 Tue Nov 25 12:56:32 2014

new/usr/src/lib/libm/amd64/src/ieee_funcl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "ieee_funcl.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(isinfl,function)
33 LIBM_ANSI_PRAGMA_WEAK(isnormall,function)
34 LIBM_ANSI_PRAGMA_WEAK(issubnormall,function)
35 LIBM_ANSI_PRAGMA_WEAK(iszerol,function)
36 LIBM_ANSI_PRAGMA_WEAK(signbitl,function)
37 #include "libm_synonyms.h"
```

```
33     ENTRY(isinfl)
34     movl    16(%rsp),%eax        / ax <-- sign and bexp of x
35     notl   %eax
36     andq   $0x7fff,%rax
37     jz     .L6
38     movq   $0,%rax
39 .not_inf:
40     ret

42 .L6:
43     movl   12(%rsp),%ecx        / here, (eax) = 0.0
44     xorl   $0x80000000,%ecx    / handle unsupported implicitly
45     orl   8(%rsp), %ecx
46     jnz   .not_inf
47     movq  $1,%rax
48     ret
49     .align 16
50     SET_SIZE(isinfl)
```

unchanged portion omitted

new/usr/src/lib/libm/amd64/src/ilogbl.s

1

2378 Tue Nov 25 12:56:33 2014

new/usr/src/lib/libm/amd64/src/ilogbl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "ilogbl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ilogbl,function)
33 #include "libm_synonyms.h"
34 #include "xpg6.h"

35     .data
36     .align 16
37 two63: .4byte 0x0,0x43d00000    / 2**63

39     ENTRY(ilogbl)
40     movq 16(%rsp),%rax    / eax <-- sign and bexp of x
41     andq $0x7fff,%rax    / eax <-- bexp(x)
42     jz   .bexp_0         / jump iff x is 0 or subnormal
43     / here, biased exponent is non-zero
44     testl $0x80000000,12(%rsp) / test msb of hi_32(sgnfcnd(x))
45     jz   .ilogbl_not_finite / jump if unsupported format
46     cmpq $0x7fff,%rax
47     je   .ilogbl_not_finite
48     subq $16383,%rax    / unbias exponent by 16383 = 0x3fff
49     ret

51 .ilogbl_not_finite:
52     movq $0x7fffffff,%rax / x is NaN/inf/unsup
53     jmp 0f

55 .bexp_0:
56     movq 8(%rsp),%rax    / rax <-- sgnfcnd(x)
57     orq  %rax,%rax
```

new/usr/src/lib/libm/amd64/src/ilogbl.s

2

```
58     jnz   .ilogbl_subnorm    / jump iff x is subnormal
59     movq  $-2147483647,%rax  / x is +/-0, so return 1-2^31

60 0:
61     PIC_SETUP(0)
62     PIC_G_LOAD(movzwq, __xpg6,rcx)
63     PIC_WRAPUP
64     andl  $_C99SUSv3_ilogb_0InfNaN_raises_invalid,%ecx
65     cmpl  $0,%ecx
66     je   1f
67     fldz
68     fdivp %st,%st(0)        / raise invalid as per SUSv3
69 1:
70     ret

73 .ilogbl_subnorm:
74     fldt  8(%rsp)           / push x, setting D-flag
75     PIC_SETUP(1)
76     fnull PIC_L(two63)     / x*2**63
77     PIC_WRAPUP
78     subq  $16,%rsp
79     fstpt (%rsp)
80     movq  $0x7fff,%rax
81     andq  8(%rsp),%rax     / eax <-- sign and bexp of x*2**63
82     subq  $16445,%rax     / unbias it by (16,383 + 63)
83     addq  $16,%rsp
84     ret
85     .align 16
86     SET_SIZE(ilogbl)

    unchanged_portion_omitted
```

new/usr/src/lib/libm/amd64/src/log101.s

1

1225 Tue Nov 25 12:56:33 2014

new/usr/src/lib/libm/amd64/src/log101.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "log101.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log101,function)
33 #include "libm_synonyms.h"

34     ENTRY(log101)
35     fldlg2
36     fldt     8(%rsp)          / st = arg, st(1) = log10(2)
37     fyl2x    / st = log10(arg) = log10(2)*log2(arg)
38     ret
39     .align 16
40     SET_SIZE(log101)
unchanged_portion_omitted
```

new/usr/src/lib/libm/amd64/src/log2l.s

1

1191 Tue Nov 25 12:56:34 2014

new/usr/src/lib/libm/amd64/src/log2l.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "log2l.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(log2l,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(log2l)
```

```
35     fldl                / push 1.0
```

```
36     fldt    8(%rsp)      / push x
```

```
37     fyl2x               / st = 1.0*log2(arg)
```

```
38     ret
```

```
39     .align 16
```

```
40     SET_SIZE(log2l)
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/logl.s

1

1212 Tue Nov 25 12:56:34 2014

new/usr/src/lib/libm/amd64/src/logl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "logl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(logl,function)
33 #include "libm_synonyms.h"

34     ENTRY(logl)
35     fldln2
36     fldt     8(%rsp)          / st = arg, st(1) = loge(2)
37     fyl2x          / st = ln(arg) = loge(2)*log2(arg)
38     ret
39     .align 16
40     SET_SIZE(logl)

unchanged_portion_omitted
```

```

*****
9493 Tue Nov 25 12:56:34 2014
new/usr/src/lib/libm/amd64/src/powl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "powl.s"

31 / Special cases:
32 /
33 / x ** 0 is 1
34 / 1 ** y is 1                               (C99)
35 / x ** NaN is NaN
36 / NaN ** y (except 0) is NaN
37 / x ** 1 is x
38 / +-(|x| > 1) ** +inf is +inf
39 / +-(|x| > 1) ** -inf is +0
40 / +-(|x| < 1) ** +inf is +0
41 / +-(|x| < 1) ** -inf is +inf
42 / (-1) ** +-inf is +1                       (C99)
43 / +0 ** +y (except 0, NaN)                  is +0
44 / -0 ** +y (except 0, NaN, odd int)         is +0
45 / +0 ** -y (except 0, NaN)                  is +inf (z flag)
46 / -0 ** -y (except 0, NaN, odd int)         is +inf (z flag)
47 / -0 ** y (odd int)                         is - (+0 ** x)
48 / +inf ** +y (except 0, NaN)                is +inf
49 / +inf ** -y (except 0, NaN)                is +0
50 / -inf ** +y (except 0, NaN)                is -0 ** -+y (NO z flag)
51 / x ** -1 is 1/x
52 / x ** 2 is x*x
53 / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
54 / x ** y (x negative & y not integer) is NaN (i flag)

56 #include "libm.h"
57 LIBM_ANSI_PRAGMA_WEAK(powl,function)
58 #include "libm_synonyms.h"

```

```

58 #include "xpg6.h"
59
61 #undef fabs
62
63     .data
64     .align 16
65     negzero:
66     .float -0.0
67     half:
68     .float 0.5
69     one:
70     .float 1.0
71     negone:
72     .float -1.0
73     two:
74     .float 2.0
75     Snan:
76     .4byte 0x7f800001
77     pinfinity:
78     .4byte 0x7f800000
79     ninfinity:
80     .4byte 0xff800000
81
82     ENTRY(powl)
83     pushq %rbp
84     movq %rsp,%rbp
85     PIC_SETUP(1)
86
87     fldt 16(%rbp)                / x
88     fxam                          / determine class of x
89     fnstsw %ax                   / store status in %ax
90     movb %ah,%dh                 / %dh <- condition code of x
91
92     fldt 32(%rbp)                / y , x
93     fxam                          / determine class of y
94     fnstsw %ax                   / store status in %ax
95     movb %ah,%dl                 / %dl <- condition code of y
96
97     call .pow_main                /// LOCAL
98     PIC_WRAPUP
99     leave
100    ret
101
102 .pow_main:
103 / x ** 0 is 1
104 movb %dl,%cl
105 andb $0x45,%cl
106 cmpb $0x40,%cl                / C3=1 C2=0 C1=? C0=0 when +-0
107 jne 1f
108 fstp %st(0)                    / x
109 fstp %st(0)                    / stack empty
110 fldl 1f
111
112 / y is not zero
113 PIC_G_LOAD(movzwbq, __xpg6, rax)
114 andl $_C99SUSv3_pow_treats_Inf_as_an_even_int,%eax
115 cmpl $0,%eax
116 je 1f
117
118 / C99: 1 ** anything is 1
119 fldl 1f                          / 1, y, x
120 fucomip %st(2),%st              / y, x
121 jp 1f                            / so that pow(NaN1,NaN2) returns NaN2
122 jne 1f

```



```

122     fstp    %st(0)           / x
123     ret

125 1:
126     / x ** NaN is NaN
127     movb   %dl,%cl
128     andb   $0x45,%cl
129     cmpb   $0x01,%cl       / C3=0 C2=0 C1=? C0=1 when +-NaN
130     jne    lf
131     fstp   %st(1)         / y
132     ret

134 1:
135     / NaN ** y (except 0) is NaN
136     movb   %dh,%cl
137     andb   $0x45,%cl
138     cmpb   $0x01,%cl       / C3=0 C2=0 C1=? C0=1 when +-NaN
139     jne    lf
140     fstp   %st(0)         / x
141     ret

143 1:
144     / x is not NaN
145     / x ** 1 is x
146     fldl   %st(1),%st     / 1, y, x
147     fcomip %st(1),%st    / y, x
148     jne    lf
149     fstp   %st(0)         / x
150     ret

151 1:
152     / y is not 1
153     / ++-(x > 1) ** +inf is +inf
154     / ++-(x > 1) ** -inf is +0
155     / ++-(x < 1) ** +inf is +0
156     / ++-(x < 1) ** -inf is +inf
157     / ++-(x = 1) ** +-inf is NaN
158     movb   %dl,%cl
159     andb   $0x47,%cl
160     cmpb   $0x05,%cl       / C3=0 C2=1 C1=0 C0=1 when +inf
161     je     .yispinf
162     cmpb   $0x07,%cl       / C3=0 C2=1 C1=1 C0=1 when -inf
163     je     .yisninf

164     / +0 ** +y (except 0, NaN)      is +0
165     / -0 ** +y (except 0, NaN, odd int) is +0
166     / +0 ** -y (except 0, NaN)      is +inf (z flag)
167     / -0 ** -y (except 0, NaN, odd int) is +inf (z flag)
168     / -0 ** y (odd int)             is - (+0 ** x)
169     movb   %dh,%cl
170     andb   $0x47,%cl
171     cmpb   $0x40,%cl       / C3=1 C2=0 C1=0 C0=0 when +0
172     je     .xispzero
173     cmpb   $0x42,%cl       / C3=1 C2=0 C1=1 C0=0 when -0
174     je     .xisnzero

176     / +inf ** +y (except 0, NaN)    is +inf
177     / +inf ** -y (except 0, NaN)    is +0
178     / -inf ** +-y (except 0, NaN)   is -0 ** +-y (NO z flag)
179     movb   %dh,%cl
180     andb   $0x47,%cl
181     cmpb   $0x05,%cl       / C3=0 C2=1 C1=0 C0=1 when +inf
182     je     .xispinf
183     cmpb   $0x07,%cl       / C3=0 C2=1 C1=1 C0=1 when -inf
184     je     .xisninf

186     / x ** -1 is 1/x
187     flds   PIC_L(negone)      / -1, y, x

```

```

188     fcomip %st(1),%st     / y, x
189     jne    lf
190     fld    %st(1)         / x, y, x
191     fdivrs PIC_L(one)     / 1/x, y, x
192     jmp    .signok        / check for over/underflow

194 1:
195     / y is not -1
196     / x ** 2 is x*x
197     flds   PIC_L(two)     / 2, y, x
198     fcomip %st(1),%st     / y, x
199     jne    lf
200     fld    %st(1)         / x, y, x
201     fld    %st(0)         / x, x, y, x
202     fmulp   %st(0),%st    / x^2, y, x
203     jmp    .signok        / check for over/underflow

204 1:
205     / y is not 2
206     / x ** 1/2 is sqrt(x)
207     flds   PIC_L(half)    / 1/2, y, x
208     fcomip %st(1),%st     / y, x
209     jne    lf
210     fld    %st(1)         / x, y, x
211     fsqrt   %st(0),%st    / sqrt(x), y, x
212     jmp    .signok        / check for over/underflow

213 1:
214     / y is not 1/2
215     / make copies of x & y
216     fld    %st(1)         / x, y, x
217     fld    %st(1)         / y, x, y, x

218     / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
219     / x ** y (x negative & y not integer) is NaN
220     movl   $0,%ecx        / track whether to flip sign of result
221     fldz   %st(0),%st     / 0, y, x, y, x
222     fcomip %st(2),%st     / compare 0 with %st(2)
223     jnb   .merge         / 0 < x
224     / x < 0
225     call   .y_is_int
226     cmpl   $0,%ecx
227     jne    lf
228     / x < 0 & y != int so x**y = NaN (i flag)
229     fstp   %st(0)         / x, y, x
230     fstp   %st(0)         / y, x
231     fstp   %st(0)         / x
232     fstp   %st(0)         / stack empty
233     fldz   %st(0)
234     fdiv   %st,%st(0)     / 0/0
235     ret

237 1:
238     / x < 0 & y = int
239     fxch   %st(0),%st(1)  / x, y, y, x
240     fchs   %st(0),%st(1)  / px = -x, y, y, x
241     fxch   %st(0),%st(1)  / y, px, y, x
242     .merge:
243     / px > 0
244     fxch   %st(0),%st(1)  / px, y, y, x

245     / x**y = exp(y*ln(x))
246     fyl2x %st(0)         / t=y*log2(px), y, x
247     fld    %st(0)         / t, t, y, x
248     frndint %st(0)       / [t], t, y, x
249     fxch   %st(0),%st(1)  / t, [t], y, x
250     fucomi %st(1),%st     / t is integral
251     je     lf
252     fsub   %st(1),%st     / t-[t], [t], y, x
253     f2xml %st(1),%st     / 2**(-[t])-1, [t], y, x

```

```

254      fadds  PIC_L(one)          / 2**(t-[t]), [t], y, x
255      fscale          / 2**t = px**y, [t], y, x
256      jmp      2f
257 1:
258      fstp   %st(0)          / t=[t], y, x
259      fldl   $1, %ecx        / 1, t, y, x
260      fscale          / 1*2**t = x**y, t, y, x
261 2:
262      fstp   %st(1)          / x**y, y, x
263      cmpl  $1, %ecx
264      jne   .signok
265      fchs
266 .signok:
267      fstp   %st(2)          / y, x**y
268      fstp   %st(0)          / x**y
269      ret

271 / -----

273 .xispinf:
274      fldz
275      fcomip %st(1), %st      / compare 0 with %st(1)
276      jb     .retpinf        / 0 < y
277      jmp    .retpzero       / y < 0

279 .xisninf:
280      / -inf ** +-y is -0 ** -+y
281      fchs
282      flds  PIC_L(negzero)    / -0, -y, x
283      fstp  %st(2)
284      jmp   .xisnzero

286 .yispinf:
287      fld   %st(1)           / x, y, x
288      fabs          / |x|, y, x
289      flds  PIC_L(one)       / 1, |x|, y, x
290      fcomip %st(1), %st     / |x|, y, x
291      fstp  %st(0)           / y, x
292      je    .retponeorinvalid / x == -1      C99
293      jb    .retpinf         / 1 < |x|
294      jmp   .retpzero        / |x| < 1

296 .yisninf:
297      fld   %st(1)           / x, y, x
298      fabs          / |x|, y, x
299      flds  PIC_L(one)       / 1, |x|, y, x
300      fcomip %st(1), %st     / |x|, y, x
301      fstp  %st(0)           / y, x
302      je    .retponeorinvalid / x == -1      C99
303      jb    .retpzero        / 1 < |x|
304      jmp   .retpinf         / |x| < 1

306 .xispzero:
307      / y cannot be 0 or NaN ; stack has y, x
308      fldz
309      fcomip %st(1), %st     / compare 0 with %st(1)
310      jb     .retpzero       / 0 < y
311      / x = +0 & y < 0 so x**y = +inf
312      jmp    .retpinfzflag   / ret +inf & z flag

314 .xisnzero:
315      / y cannot be 0 or NaN ; stack has y, x
316      call  .y_is_int
317      cmpl  $1, %ecx
318      jne   lf
319      / y is not an odd integer

```

```

320      fldz
321      fcomip %st(1), %st     / compare 0 with %st(1)
322      jb     .retnzero       / 0 < y
323      / x = -0 & y < 0 (odd int)
324      / x = -inf & y != 0 or NaN
325      movb  %dh, %cl
326      andb  $0x45, %cl
327      cmpb  $0x05, %cl      / C3=0 C2=1 C1=? C0=1 when +-inf
328      je    2f
329      fdiv  %st, %st(1)     / y / x, x (raise z flag)
330 2:
331      fstp  %st(0)          / x
332      fstp  %st(0)          / stack empty
333      flds  PIC_L(ninfinite) / -inf
334      ret

336 1:
337      fldz
338      fcomip %st(1), %st     / compare 0 with %st(1)
339      jb     .retpzero       / 0 < y
340      / x = -0 & y < 0 (not odd int)
341      / x = -inf & y not 0 or NaN
342      movb  %dh, %cl
343      andb  $0x45, %cl
344      cmpb  $0x05, %cl      / C3=0 C2=1 C1=? C0=1 when +-inf
345      jne   .retpinfzflag   / ret +inf & divide-by-0 flag
346      jmp   .retpinf        / return +inf (NO z flag)

348 .retpzero:
349      fstp  %st(0)          / x
350      fstp  %st(0)          / stack empty
351      fldz
352      ret

354 .retnzero:
355      fstp  %st(0)          / x
356      fstp  %st(0)          / stack empty
357      flds  PIC_L(negzero)  / -0
358      ret

360 .retponeorinvalid:
361      PIC_G_LOAD(movzwdq, __xpg6, rax)
362      andl  $_C99SUSv3_pow_treats_Inf_as_an_even_int, %eax
363      cmpl  $0, %eax
364      je    lf
365      fstp  %st(0)          / x
366      fstp  %st(0)          / stack empty
367      fldl  $1
368      ret

370 1:
371      fstp  %st(0)          / x
372      fstp  %st(0)          / stack empty
373      flds  PIC_L(Snan)     / Q NaN (i flag)
374      fwait
375      ret

377 .retpinf:
378      fstp  %st(0)          / x
379      fstp  %st(0)          / stack empty
380      flds  PIC_L(pinfinit) / +inf
381      ret

383 .retpinfzflag:
384      fstp  %st(0)          / x
385      fstp  %st(0)          / stack empty

```

```
386      fldz
387      fdivrs PIC_L(one)      / 1/0
388      ret

390 / Set %ecx to 2 if y is an even integer, 1 if y is an odd integer,
391 / 0 otherwise. Assume y is not zero. Do not raise inexact or modify
392 / %edx.
393 .y_is_int:
394      movl    40(%rbp),%eax
395      andl    $0x7fff,%eax      / exponent of y
396      cmpl    $0x403f,%eax
397      jae     1f                / |y| >= 2^64, an even int
398      cmpl    $0x3fff,%eax
399      jb      2f                / |y| < 1, can't be an int
400      movl    %eax,%ecx
401      subl    $0x403e,%ecx
402      negl    %ecx                / 63 - unbiased exponent of y
403      movq    32(%rbp),%rax
404      bsfq    %rax,%rax          / index of least sig. 1 bit
405      cmpl    %ecx,%eax
406      jb      2f
407      ja      1f
408      movl    $1,%ecx
409      ret
410 1:
411      movl    $2,%ecx
412      ret
413 2:
414      xorl    %ecx,%ecx
415      ret
416      .align 16
417      SET_SIZE(powl)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/remainder.s

1

```
*****
2016 Tue Nov 25 12:56:35 2014
new/usr/src/lib/libm/amd64/src/remainder.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainder.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainder,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

35     ENTRY(remainder)
36     push    %rbp
37     movq   %rsp,%rbp
38     subq   $16,%rsp
39     movlpd %xmm1,-16(%rbp)
40     movlpd %xmm0,-8(%rbp)

42     ucomisd %xmm0,%xmm1      / if x or y is NaN, use fprem1
43     jp     lf

45     movl   -12(%rbp),%eax    / eax <-- hi_32(y)
46     andl   $0x7fffffff,%eax / eax <-- hi_32(|y|)
47     orl   -16(%rbp),%eax    / eax <-- lo_32(y)|hi_32(|y|)
48     je     .yzero_or_xinf

50     movl   -4(%rbp),%eax    / eax <-- hi_32(x)
51     andl   $0x7fffffff,%eax / eax <-- hi_32(|x|)
52     cmpl   $0x7ff00000,%eax
53     jne   lf
54     cmpl   $0,-8(%rbp)
55     je     .yzero_or_xinf
56 1:
57     fldl   -16(%rbp)      / y
```

new/usr/src/lib/libm/amd64/src/remainder.s

2

```
58     fldl   -8(%rbp)      / x
59 .rem_loop:
60     fprem1                                / partial remainder
61     fstsw  %ax                    / store status word
62     andw  $0x400,%ax              / check for incomplete reduction
63     jne   .rem_loop              / while incomplete, do fprem1 again
64     fstpl -8(%rbp)
65     movsd -8(%rbp),%xmm0
66     fstp  %st(0)
67     leave
68     ret

70 .yzero_or_xinf:
71     PIC_SETUP(1)
72     movl  $28,%edi
73     movl  $2,%eax
74     call PIC_F(_SVID_libm_err)
75     PIC_WRAPUP
76     leave
77     ret
78     .align 4
79     SET_SIZE(remainder)
_____ unchanged portion omitted_____
```

new/usr/src/lib/libm/amd64/src/remainderf.s

1

1530 Tue Nov 25 12:56:35 2014

new/usr/src/lib/libm/amd64/src/remainderf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainderf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainderf,function)
33 #include "libm_synonyms.h"

34     ENTRY(remainderf)
35     push    %rbp
36     movq   %rsp,%rbp
37     subq   $16,%rsp
38     movss %xmm1,-8(%rbp)
39     movss %xmm0,-4(%rbp)
40     flds  -8(%rbp)           / load arg y
41     flds  -4(%rbp)           / load arg x
42 .rem_loop:
43     fprem1                    / partial remainder
44     fstsw  %ax                / store status word
45     andw  $0x400,%ax          / check whether reduction complete
46     jne   .rem_loop          / while reduction incomplete, do fprem1
47     fstps -4(%rbp)
48     movss -4(%rbp),%xmm0
49     fstp  %st(0)
50     leave
51     ret
52     .align 4
53     SET_SIZE(remainderf)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/remainderl.s

1

1400 Tue Nov 25 12:56:36 2014

new/usr/src/lib/libm/amd64/src/remainderl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainderl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainderl,function)
33 #include "libm_synonyms.h"

34     ENTRY(remainderl)
35     fldt    24(%rsp)          / load arg y
36     fldt    8(%rsp)          / load arg x
37 .rem_loop:
38     fprem1                          / partial remainder
39     fstsw   %ax                / store status word
40     andw   $0x400,%ax         / check whether reduction complete
41     jne    .rem_loop         / while reduction incomplete, do fprem1
42     fstp   %st(1)
43     ret
44     .align 16
45     SET_SIZE(remainderl)

unchanged portion omitted
```

new/usr/src/lib/libm/amd64/src/remquol.s

1

1879 Tue Nov 25 12:56:37 2014

new/usr/src/lib/libm/amd64/src/remquol.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "remquol.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remquol,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"
35     ENTRY(remquol)
36     fldt     24(%rsp)          / load arg y
37     fldt     8(%rsp)          / load arg x
38 .Lreml_loop:
39     fpreml
40     fstsw   %ax              / store status word
41     andw   $0x400,%ax       / check whether reduction complete
42     jne    .Lreml_loop      / while reduction incomplete, do fpreml
43     fstsw   %ax
44     fwait
45     fstp   %st(1)
46     movw   %ax,%dx
47     andw   $0x4000,%dx      / get C3
48     sarw   $13,%dx
49     movw   %ax,%cx
50     andw   $0x100,%cx       / get C0
51     sarw   $6,%cx
52     addw   %cx,%dx
53     andw   $0x200,%ax       / get C1
54     sarw   $9,%ax
55     addw   %dx,%ax
56     cwtl
57     movl   16(%rsp),%edx     / sign and bexp of x
58     movl   32(%rsp),%ecx     / sign and bexp of y
```

new/usr/src/lib/libm/amd64/src/remquol.s

2

```
58     andl   $0x8000,%edx     / edx <- sign(x)
59     andl   $0x8000,%ecx     / ecx <- sign(y)
60     cmpl   %edx,%ecx
61     je     lf
62     negl   %eax              / negative n
63 1:
64     movl   %eax,(%rdi)      / last 3 significant bits of quotient
65     ret
66     .align 16
67     SET_SIZE(remquol)
unchanged_portion_omitted
```

new/usr/src/lib/libm/amd64/src/rintl.s

1

1187 Tue Nov 25 12:56:37 2014

new/usr/src/lib/libm/amd64/src/rintl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "rintl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(rintl,function)
33 #include "libm_synonyms.h"

34     ENTRY(rintl)
35     fldt     8(%rsp)           / load x
36     frndint           / [x], per rounding mode
37     fwait
38     ret
39     .align 16
40     SET_SIZE(rintl)

unchanged_portion_omitted
```


new/usr/src/lib/libm/amd64/src/rndintl.s

1

2915 Tue Nov 25 12:56:38 2014

new/usr/src/lib/libm/amd64/src/rndintl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "rndintl.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(aintl,function)
33 LIBM_ANSI_PRAGMA_WEAK(irintl,function)
34 LIBM_ANSI_PRAGMA_WEAK(anintl,function)
35 LIBM_ANSI_PRAGMA_WEAK(nintl,function)
36 #include "libm_synonyms.h"
```

```
38 #undef fabs
```

```
33     ENTRY(aintl)
34     movq    %rsp,%rax
35     subq    $16,%rsp
36     fstcw  -8(%rax)
37     fldt   8(%rax)
38     movw   -8(%rax),%cx
39     orw   $0x0c00,%cx
40     movw   %cx,-4(%rax)
41     fldcw -4(%rax)           / set RD = to_zero
42     frndint
43     fstcw  -4(%rax)
44     movw   -4(%rax),%dx
45     andw   $0xf3ff,%dx
46     movw   -8(%rax),%cx
47     andw   $0x0c00,%cx
48     orw   %dx,%cx
49     movw   %cx,-8(%rax)
50     fldcw -8(%rax)           / restore RD
51     addq   $16,%rsp
```

new/usr/src/lib/libm/amd64/src/rndintl.s

2

```
52     ret
53     .align 16
54     SET_SIZE(aintl)
_____ unchanged_portion_omitted_____
```

new/usr/src/lib/libm/amd64/src/scalbnl.s

1

1224 Tue Nov 25 12:56:38 2014

new/usr/src/lib/libm/amd64/src/scalbnl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalbnl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalbnl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalbnl)
35     subq   $16,%rsp
36     movl   %edi,(%rsp)
37     fildl  (%rsp)
38     fldt  24(%rsp)
39     addq   $16,%rsp
40     fscale
41     fstp  %st(1)
42     ret
43     .align 16
44     SET_SIZE(scalbnl)
```

unchanged_portion_omitted

new/usr/src/lib/libm/amd64/src/sqrtl.s

1

1139 Tue Nov 25 12:56:39 2014

new/usr/src/lib/libm/amd64/src/sqrtl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file     "sqrtl.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(sqrtl,function)
33 #include "libm_synonyms.h"

34     ENTRY(sqrtl)
35     fldt     8(%rsp)
36     fsqrt
37     ret
38     .align  16
39     SET_SIZE(sqrtl)
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/acos.c

1

```
*****
4619 Tue Nov 25 12:56:39 2014
new/usr/src/lib/libm/common/C/acos.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __acos = acos
31 #pragma weak acos = __acos
32
33 /* INDENT OFF */
34 /*
35  * acos(x)
36  * Method :
37  *   acos(x) = pi/2 - asin(x)
38  *   acos(-x) = pi/2 + asin(x)
39  * For |x|<=0.5
40  *   acos(x) = pi/2 - (x + x*x^2*R(x^2))      (see asin.c)
41  * For x>0.5
42  *   acos(x) = pi/2 - (pi/2 - 2asin(sqrt((1-x)/2)))
43  *             = 2asin(sqrt((1-x)/2))
44  *             = 2s + 2s*z*R(z)      ...z=(1-x)/2, s=sqrt(z)
45  *             = 2f + (2c + 2s*z*R(z))
46  *   where f=hi part of s, and c = (z-f*f)/(s+f) is the correction term
47  *   for f so that f+c ~ sqrt(z).
48  * For x<-0.5
49  *   acos(x) = pi - 2asin(sqrt((1-|x|)/2))
50  *             = pi - 0.5*(s+s*z*R(z)), where z=(1-|x|)/2,s=sqrt(z)
51  *
52  * Special cases:
53  *   if x is NaN, return x itself;
54  *   if |x|>1, return NaN with invalid signal.
55  *
56  * Function needed: sqrt
57  */
58 /* INDENT ON */
```

new/usr/src/lib/libm/common/C/acos.c

2

```
59 #include "libm_synonyms.h" /* __acos, __sqrt, __isnan */
59 #include "libm_protos.h" /* _SVID_libm_error */
60 #include "libm_macros.h"
61 #include <math.h>
62
63 /* INDENT OFF */
64 static const double xxx[] = {
65  /* one */ 1.000000000000000000000000e+00, /* 3FF00000, 00000000 */
66  /* pi */ 3.14159265358979311600e+00, /* 400921FB, 54442D18 */
67  /* pio2_hi */ 1.57079632679489655800e+00, /* 3FF921FB, 54442D18 */
68  /* pio2_lo */ 6.12323399573676603587e-17, /* 3C91A626, 33145C07 */
69  /* pS0 */ 1.66666666666666666657415e-01, /* 3FC55555, 55555555 */
70  /* pS1 */ -3.25565818622400915405e-01, /* BFD4D612, 03EB6F7D */
71  /* pS2 */ 2.01212532134862925881e-01, /* 3FC9C155, 0E884455 */
72  /* pS3 */ -4.00555345006794114027e-02, /* BFA48228, B5688F3B */
73  /* pS4 */ 7.91534994289814532176e-04, /* 3F49EFE0, 7501B288 */
74  /* pS5 */ 3.47933107596021167570e-05, /* 3F023DE1, 0DFDF709 */
75  /* qS1 */ -2.40339491173441421878e+00, /* C0033A27, 1C8A2D4B */
76  /* qS2 */ 2.02094576023350569471e+00, /* 40002AE5, 9C598AC8 */
77  /* qS3 */ -6.88283971605453293030e-01, /* BFE6066C, 1B8D0159 */
78  /* qS4 */ 7.70381505559019352791e-02 /* 3FB3B8C5, B12E9282 */
79 };
80
81 unchanged portion omitted
```

```

*****
2562 Tue Nov 25 12:56:40 2014
new/usr/src/lib/libm/common/C/acosh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __acosh = acosh
30 #pragma weak acosh = __acosh

32 /* INDENT OFF */
33 /*
34  * acosh(x)
35  * Method :
36  *   Based on
37  *   acosh(x) = log [ x + sqrt(x*x-1) ]
38  *   we have
39  *   acosh(x) := log(x)+ln2, if x is large; else
40  *   acosh(x) := log(2x-1/(sqrt(x*x-1)+x)) if x > 2; else
41  *   acosh(x) := loglp(t+sqrt(2.0*t*t)); where t = x-1.
42  *
43  * Special cases:
44  *   acosh(x) is NaN with signal if x < 1.
45  *   acosh(NaN) is NaN without signal.
46 */
47 /* INDENT ON */

49 #include "libm_synonyms.h" /* __acosh, __log, __loglp */
49 #include "libm_protos.h" /* _SVID_libm_error */
50 #include "libm_macros.h"
51 #include <math.h>

53 static const double
54     one = 1.0,
55     ln2 = 6.93147180559945286227e-01; /* 3FE62E42, FEFA39EF */

```

```

57 double
58 acosh(double x) {
59     double t;
60     int hx;

62     hx = ((int *) &x)[HIWORD];
63     if (hx < 0x3ff00000) { /* x < 1 */
64         if (isnan(x))
65             #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
66                 return (hx >= 0xffff80000 ? x : (x - x) / (x - x));
67             /* assumes sparc-like QNaN */
68         #else
69                 return (x - x) / (x - x);
70         #endif
71         else
72             return (_SVID_libm_err(x, x, 29));
73     } else if (hx >= 0x41b00000) {
74         /* x > 2**28 */
75         if (hx >= 0x7ff00000) { /* x is inf of NaN */
76             #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
77                 return (hx >= 0x7ff80000 ? x : x + x);
78             /* assumes sparc-like QNaN */
79         #else
80                 return (x + x);
81             #endif
82         } else /* acosh(huge)=log(2x) */
83             return (log(x) + ln2);
84     } else if (((hx - 0x3ff00000) | ((int *) &x)[LOWORD]) == 0) {
85         return (0.0); /* acosh(1) = 0 */
86     } else if (hx > 0x40000000) {
87         /* 2**28 > x > 2 */
88         t = x * x;
89         return (log(2.0 * x - one / (x + sqrt(t - one))));
90     } else {
91         /* 1 < x < 2 */
92         t = x - one;
93         return (loglp(t + sqrt(2.0 * t + t * t)));
94     }
95 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/C/asin.c

1

```
*****
4821 Tue Nov 25 12:56:40 2014
new/usr/src/lib/libm/common/C/asin.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __asin = asin
30 #pragma weak asin = __asin
32 /* INDENT OFF */
33 /*
34  * asin(x)
35  * Method :
36  * Since asin(x) = x + x^3/6 + x^5*3/40 + x^7*15/336 + ...
37  * we approximate asin(x) on [0,0.5] by
38  * asin(x) = x + x*x^2*R(x^2)
39  *
40  * where
41  * R(x^2) is a rational approximation of (asin(x)-x)/x^3
42  * and its remez error is bounded by
43  * |(asin(x)-x)/x^3 - R(x^2)| < 2^(-58.75)
44  *
45  * For x in [0.5,1]
46  * asin(x) = pi/2 - 2*asin(sqrt((1-x)/2))
47  * Let y = (1-x), z = y/2, s := sqrt(z), and pio2_hi+pio2_lo=pi/2;
48  * then for x>0.98
49  * asin(x) = pi/2 - 2*(s+s*z*R(z))
50  * = pio2_hi - (2*(s+s*z*R(z)) - pio2_lo)
51  * For x<=0.98, let pio4_hi = pio2_hi/2, then
52  * f = hi part of s;
53  * c = sqrt(z) - f = (z-f*f)/(s+f) ...f+c=sqrt(z)
54  *
55  * and
56  * asin(x) = pi/2 - 2*(s+s*z*R(z))
57  * = pio4_hi+(pio4-2s)-(2s*z*R(z)-pio2_lo)
58  * = pio4_hi+(pio4-2f)-(2s*z*R(z)-(pio2_lo+2c))

```

new/usr/src/lib/libm/common/C/asin.c

2

```
58 * Special cases:
59 * if x is NaN, return x itself;
60 * if |x|>1, return NaN with invalid signal.
61 *
62 */
63 /* INDENT ON */
65 #include "libm_synonyms.h" /* __asin, __sqrt, __isnan */
65 #include "libm_protos.h" /* _SVID_libm_error */
66 #include "libm_macros.h"
67 #include <math.h>
69 /* INDENT OFF */
70 static const double xxx[] = {
71 /* one */ 1.0000000000000000000000e+00, /* 3FF00000, 00000000 */
72 /* huge */ 1.000e+300,
73 /* pio2_hi */ 1.57079632679489655800e+00, /* 3FF921FB, 54442D18 */
74 /* pio2_lo */ 6.12323399573676603587e-17, /* 3C91A626, 33145C07 */
75 /* pio4_hi */ 7.85398163397448278999e-01, /* 3FE921FB, 54442D18 */
76 /* coefficient for R(x^2) */
77 /* pS0 */ 1.66666666666666657415e-01, /* 3FC55555, 55555555 */
78 /* pS1 */ -3.25565818622400915405e-01, /* BFD4D612, 03EB6F7D */
79 /* pS2 */ 2.01212532134862925881e-01, /* 3FC9C155, 0E884455 */
80 /* pS3 */ -4.00555345006794114027e-02, /* BFA48228, B5688F3B */
81 /* pS4 */ 7.91534994289814532176e-04, /* 3F49EFE0, 7501B288 */
82 /* pS5 */ 3.47933107596021167570e-05, /* 3F023DE1, 0DFDF709 */
83 /* qS1 */ -2.40339491173441421878e+00, /* C0033A27, 1C8A2D4B */
84 /* qS2 */ 2.02094576023350569471e+00, /* 40002AE5, 9C598AC8 */
85 /* qS3 */ -6.88283971605453293030e-01, /* BFE6066C, 1B8D0159 */
86 /* qS4 */ 7.70381505559019352791e-02 /* 3FB3B8C5, B12E9282 */
87 };
unchanged_portion_omitted

```

new/usr/src/lib/libm/common/C/asinh.c

1

```
*****
2393 Tue Nov 25 12:56:41 2014
new/usr/src/lib/libm/common/C/asinh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __asinh = asinh
30 #pragma weak asinh = __asinh

32 /* INDENT OFF */
33 /*
34  * asinh(x)
35  * Method :
36  *   Based on
37  *       asinh(x) = sign(x) * log [ |x| + sqrt(x*x+1) ]
38  *   we have
39  *       asinh(x) := x if 1+x*x == 1,
40  *                := sign(x)*(log(x)+ln2)) for large |x|, else
41  *                := sign(x)*log(2|x|+1/(|x|+sqrt(x*x+1))) if |x| > 2, else
42  *                := sign(x)*loglp(|x|+x^2/(1+sqrt(1+x^2)))
43  */
44 /* INDENT ON */

46 #include "libm_synonyms.h" /* __asinh */
46 #include "libm_macros.h"
47 #include <math.h>

49 static const double xxx[] = {
50 /* one */      1.00000000000000000000e+00, /* 3FF00000, 00000000 */
51 /* ln2 */     6.93147180559945286227e-01, /* 3FE62E42, FEFA39EF */
52 /* huge */    1.00000000000000000000e+300
53 };
    unchanged_portion_omitted
```



```

*****
18614 Tue Nov 25 12:56:41 2014
new/usr/src/lib/libm/common/C/atan2.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __atan2 = atan2
29 #pragma weak atan2 = __atan2

31 #include "libm.h"

33 /*
34 * Let t(0) = 1 and for i = 1, ..., 160, let t(i) be the slope of
35 * the line bisecting the conical hull of the set of points (x,y)
36 * where x and y are positive normal floating point numbers and
37 * the high order words hx and hy of their binary representations
38 * satisfy |hx - hy - i * 0x8000| <= 0x4000. Then:
39 *
40 * TBL[4*i+2] is t(i) rounded to 21 significant bits (i.e., the
41 * low order word is zero), and
42 *
43 * TBL[4*i] + TBL[4*i+1] form a doubled-double approximation to
44 * atan(TBL[4*i+2]).
45 *
46 * Finally, TBL[4*161] = TBL[4*161+1] = TBL[4*161+2] = 0.
47 *
48 * Now for any (x,y) with 0 < y <= x and any 0 < t <= 1, we have
49 * atan(y/x) = atan(t) + atan((y-t*x)/(x+t*y)). By choosing t =
50 * TBL[4*i+2], where i is the multiple of 0x8000 nearest hx - hy,
51 * if this multiple is less than 161, and i = 161 otherwise, we
52 * find that |(y-t*x)/(x+t*y)| <~ 2^-5.
53 */
54 static const double TBL[] = {
55     7.8539816339744827900e-01, +3.0616169978683830179e-17,
56     1.000000000000000000e+00, +0,
57     7.7198905126506112140e-01, +2.6989956960083153652e-16,

```

```

58     9.7353506088256835938e-01, +0,
59     7.6068143954461309164e-01, -3.5178810518941914972e-16,
60     9.5174932479858398438e-01, +0,
61     7.4953661876353638860e-01, -3.2548100004524337476e-16,
62     9.3073129653930664062e-01, +0,
63     7.3854614984728339522e-01, -2.0775571023910406668e-16,
64     9.1042709350585937500e-01, +0,
65     7.2770146962041337702e-01, +3.8883249403168348802e-16,
66     8.9078664779663085938e-01, +0,
67     7.1699492488093774512e-01, -4.0468841511547224071e-16,
68     8.7176513671875000000e-01, +0,
69     7.0641813488653149022e-01, +5.6902424353981484031e-17,
70     8.5331964492797851562e-01, +0,
71     6.9596351101035658360e-01, +2.8245513321075021303e-16,
72     8.3541154861450195312e-01, +0,
73     6.8562363680534943455e-01, -4.2316970721658854064e-16,
74     8.1800508499145507812e-01, +0,
75     6.7539055666438230219e-01, +4.3535917281300047233e-16,
76     8.0106592178344726562e-01, +0,
77     6.6525763346931832132e-01, +1.1830431602404727977e-17,
78     7.8456401824951171875e-01, +0,
79     6.5521767574310185722e-01, -1.7435923100651044208e-16,
80     7.6847028732299804688e-01, +0,
81     6.4526390999481897381e-01, -1.4741927403093983947e-16,
82     7.5275802612304687500e-01, +0,
83     6.3538979894204850041e-01, +1.5734535069995660853e-16,
84     7.3740243911743164062e-01, +0,
85     6.2558914346942717799e-01, -2.8175588856316910960e-16,
86     7.2238063812255859375e-01, +0,
87     6.1585586476157949676e-01, -4.3056167357725226449e-16,
88     7.0767116546630859375e-01, +0,
89     6.0618408027576098362e-01, +1.5018013918429320289e-16,
90     6.9325399398803710938e-01, +0,
91     5.9656817827486730010e-01, +5.5271942033557644157e-17,
92     6.7911052703857421875e-01, +0,
93     5.8700289083426504533e-01, -8.2411369282676383293e-17,
94     6.6522359848022460938e-01, +0,
95     5.7748303053627658699e-01, +4.9400383775709159558e-17,
96     6.5157699584960937500e-01, +0,
97     5.6800353968303252117e-01, +2.9924431103311109543e-16,
98     6.3815546035766601562e-01, +0,
99     5.5855953863493823519e-01, -2.0306003403868777403e-16,
100    6.2494468688964843750e-01, +0,
101    5.4914706708329674711e-01, +2.8255378613779667461e-17,
102    6.1193227767944335938e-01, +0,
103    5.3976176660618069292e-01, +1.6370248781078747995e-16,
104    5.9910583496093750000e-01, +0,
105    5.3039888601412332747e-01, -7.6196097360093680134e-17,
106    5.8645296096801757812e-01, +0,
107    5.2105543924318808990e-01, -2.2400815668154739561e-16,
108    5.7396411895751953125e-01, +0,
109    5.117278873967050828e-01, -3.6888136019899681185e-16,
110    5.6162929534912109375e-01, +0,
111    5.0241199666452196482e-01, -2.5412891474397011281e-16,
112    5.4943847656250000000e-01, +0,
113    4.9310493954293743712e-01, +4.4132186128251152229e-16,
114    5.3738307952880859375e-01, +0,
115    4.8380436844750995817e-01, -2.784438790776656488e-16,
116    5.2545595169067382812e-01, +0,
117    4.7450670361463753721e-01, -2.0494355197368286028e-16,
118    5.1364850997924804688e-01, +0,
119    4.6367660027976320691e-01, +3.1709878607954760668e-16,
120    5.0003623962402343750e-01, +0,
121    4.5304753104003925301e-01, +3.3593436122420574865e-16,
122    4.8681926727294921875e-01, +0,
123    4.4423658037407065535e-01, +2.1987183192008082015e-17,

```

```

124 4.7596645355224609375e-01, +0,
125 4.3567016972500294258e-01, +3.0118422805369552650e-16,
126 4.6550178527832031250e-01, +0,
127 4.2733152672544871820e-01, -3.2667693224866479909e-16,
128 4.5539522171020507812e-01, +0,
129 4.1920540176693954493e-01, -2.2454273841113897647e-16,
130 4.4561982154846191406e-01, +0,
131 4.1127722812701872357e-01, -3.1620568973494653391e-16,
132 4.3615055084228515625e-01, +0,
133 4.0353384063084263289e-01, -3.5932009901481421723e-16,
134 4.2696499824523925781e-01, +0,
135 3.9596319345246833166e-01, -4.0281533417458698585e-16,
136 4.1804289817810058594e-01, +0,
137 3.88554052220339722661e-01, +1.6132231486045176674e-16,
138 4.0936565399169921875e-01, +0,
139 3.8129566313738116889e-01, +1.7684657060650804570e-16,
140 4.0091586112976074219e-01, +0,
141 3.7417884791401867517e-01, +2.6897604227426977619e-16,
142 3.9267849922180175781e-01, +0,
143 3.6719421967585041955e-01, -4.5886151448673745001e-17,
144 3.8463878631591796875e-01, +0,
145 3.6033388248727771241e-01, +1.5804115573136074946e-16,
146 3.7678408622741699219e-01, +0,
147 3.5358982224579182940e-01, +1.2624619863035782939e-16,
148 3.6910200119018554688e-01, +0,
149 3.4695498404186952968e-01, +9.3221684607372865177e-17,
150 3.6158156394958496094e-01, +0,
151 3.4042268308109679964e-01, +2.7697913559445449137e-16,
152 3.5421252250671386719e-01, +0,
153 3.3398684598563566084e-01, +3.6085337449716011085e-16,
154 3.4698557853698730469e-01, +0,
155 3.2764182824591436827e-01, +2.0581506352606456186e-16,
156 3.3989214897155761719e-01, +0,
157 3.2138200938788497041e-01, -1.9015787485430693661e-16,
158 3.3292388916015625000e-01, +0,
159 3.1520245348069497737e-01, +2.6961839659264087022e-16,
160 3.2607340812683105469e-01, +0,
161 3.0909871873117023000e-01, -1.5641891686756272625e-16,
162 3.1933403015136718750e-01, +0,
163 3.0306644308947827682e-01, +2.8801634211591956223e-16,
164 3.1269931793212890625e-01, +0,
165 2.9710135482774191473e-01, -4.3148994478973365819e-16,
166 3.0616307258605957031e-01, +0,
167 2.9120015759141004708e-01, -6.8539854790808585159e-17,
168 2.9972028732299804688e-01, +0,
169 2.8535879880370362827e-01, -1.2231638445300492682e-16,
170 2.9336524009704589844e-01, +0,
171 2.7957422506893880865e-01, -4.6707752931043135528e-17,
172 2.8709340095520019531e-01, +0,
173 2.7384352102802367313e-01, -4.1215636366229625876e-16,
174 2.8090047836303710938e-01, +0,
175 2.6816369484161040049e-01, -2.3700583122400495333e-16,
176 2.7478218078613281250e-01, +0,
177 2.6253212627627764419e-01, +2.3123213692190889610e-16,
178 2.6873469352722167969e-01, +0,
179 2.5694635355759309903e-01, -4.0638513814701264145e-16,
180 2.6275444030761718750e-01, +0,
181 2.5140385572454615470e-01, -3.4795333793554943723e-16,
182 2.5683784484863281250e-01, +0,
183 2.4500357070096612233e-01, +6.6542334848010259289e-17,
184 2.5002646446228027344e-01, +0,
185 2.3877766609573036760e-01, -2.7756633678549343650e-16,
186 2.4342155456542968750e-01, +0,
187 2.3365669377188336142e-01, +3.2700803838522067998e-16,
188 2.3800384998321533203e-01, +0,
189 2.2870810463931334766e-01, -4.4279127662219799521e-16,

```

```

190 2.3278105258941650391e-01, +0,
191 2.2391820542294382790e-01, +3.7558889374284208052e-16,
192 2.2773718833923339844e-01, +0,
193 2.1927501815429550902e-01, -1.4829838176513811186e-16,
194 2.2285830974578857422e-01, +0,
195 2.1476740847367459253e-01, -2.0535381496063397578e-17,
196 2.1813154220581054688e-01, +0,
197 2.1038568111737454558e-01, -4.28267677738736168650e-16,
198 2.1354568004608154297e-01, +0,
199 2.0612057974373865221e-01, +4.2108051749502232359e-16,
200 2.0909011363983154297e-01, +0,
201 2.0196410359405447821e-01, +3.5157118083511092869e-16,
202 2.0475566387176513672e-01, +0,
203 1.9790861144712756925e-01, +3.7894950972257700994e-16,
204 2.0053362846374511719e-01, +0,
205 1.9394752160084305359e-01, +2.8270367403478935534e-16,
206 1.9641649723052978516e-01, +0,
207 1.9007440763641536563e-01, -2.0842758095683676397e-16,
208 1.9239699840545654297e-01, +0,
209 1.8628369629742813629e-01, +3.4710917040399448932e-16,
210 1.8846881389617919922e-01, +0,
211 1.8256998712939509488e-01, +1.1053834120570125251e-16,
212 1.8462586402893066406e-01, +0,
213 1.7892875067284830237e-01, +3.0486232913366680305e-16,
214 1.8086302280426025391e-01, +0,
215 1.7535529778449010507e-01, -2.3810135019970148624e-16,
216 1.7717504501342773438e-01, +0,
217 1.7184559192514736736e-01, +5.1432582846210893916e-17,
218 1.7355740070343017578e-01, +0,
219 1.6839590847744290159e-01, +3.1605623296041433586e-18,
220 1.7000591754913330078e-01, +0,
221 1.6500283902547518977e-01, +1.5405422268770998251e-16,
222 1.6651678085327148438e-01, +0,
223 1.6166306303174859949e-01, +4.0042241517254928672e-16,
224 1.6380829512786865234e-01, +0,
225 1.5837358268281231943e-01, -2.2786616251622967291e-16,
226 1.5971112251281738281e-01, +0,
227 1.5513160990288810126e-01, -3.7547723514797166336e-16,
228 1.5638816356658935547e-01, +0,
229 1.5193468535499299321e-01, +4.3497510505554267446e-16,
230 1.5311467647552490234e-01, +0,
231 1.4878033155427861089e-01, -2.3102860235324261895e-16,
232 1.4988791942596435547e-01, +0,
233 1.4566628729590647140e-01, +9.9227592950040279415e-17,
234 1.4670538902282714844e-01, +0,
235 1.4259050967286590605e-01, -3.3869909683813096906e-18,
236 1.4356482028961181641e-01, +0,
237 1.3955105903633846509e-01, +1.5500435650773331566e-17,
238 1.4046406745910644531e-01, +0,
239 1.3654610022831903393e-01, +3.3965918616682805753e-16,
240 1.3740110397338867188e-01, +0,
241 1.3357402082462854764e-01, +2.7572431581527535421e-16,
242 1.3437414169311523438e-01, +0,
243 1.3063319828908959153e-01, -3.4667213797076707331e-16,
244 1.3138139247894287109e-01, +0,
245 1.2772200049776749609e-01, +3.1089261947725651968e-16,
246 1.2842106819152832031e-01, +0,
247 1.2436931430778752627e-01, -4.0654251891464630059e-16,
248 1.250145435332519531e-01, +0,
249 1.2111683701666819957e-01, -3.9381654342464836012e-16,
250 1.2171256542205810547e-01, +0,
251 1.1844801833536511282e-01, -3.6673155595150283444e-16,
252 1.1900508403778076172e-01, +0,
253 1.1587365536613614125e-01, -1.5026628801318421951e-16,
254 1.1639505624771118164e-01, +0,
255 1.1338607085741525538e-01, +1.2886806274050538880e-16,

```

256 1.1387449502944946289e-01, +0,
257 1.10978444020819369604e-01, +2.3848343623577768044e-16,
258 1.1143630743026733398e-01, +0,
259 1.0864456107308662069e-01, +4.2065430313285469408e-16,
260 1.0907405614852905273e-01, +0,
261 1.0637891628473727934e-01, -4.6883543790348472687e-18,
262 1.0678201913833618164e-01, +0,
263 1.0417650062205296990e-01, +1.4774925414624453292e-16,
264 1.0455501079559326172e-01, +0,
265 1.0203276464730581807e-01, -1.5677032794816452332e-16,
266 1.0238832235336303711e-01, +0,
267 9.9943617083734892503e-02, +3.4511310907979792828e-16,
268 1.0027772188186645508e-01, +0,
269 9.7905249824711049200e-02, +3.4489485563461708496e-16,
270 9.8219275474548339844e-02, +0,
271 9.5914316649349906641e-02, -1.3214510886789011569e-17,
272 9.6209526062011718750e-02, +0,
273 9.3967698614664918466e-02, +1.1048427091217964090e-16,
274 9.4245254993438720703e-02, +0,
275 9.2062564267554769515e-02, -3.7297463814697759309e-16,
276 9.2323541641235351562e-02, +0,
277 9.0196252506350660383e-02, -3.5280143043576718079e-16,
278 9.0441644191741943359e-02, +0,
279 8.8366391663268650802e-02, -6.1140673227541621183e-17,
280 8.8597118854522705078e-02, +0,
281 8.6570782100201526532e-02, -2.0998844594957629702e-16,
282 8.6787700653076171875e-02, +0,
283 8.4807337678923566671e-02, +3.9530981588194673068e-16,
284 8.5011243820190429688e-02, +0,
285 8.3074323040850828193e-02, -4.3022503210464894539e-17,
286 8.3265960216522216797e-02, +0,
287 8.1369880712663267275e-02, -6.3063867569127169744e-18,
288 8.1549942493438720703e-02, +0,
289 7.9692445771216036121e-02, -5.0787623072962671502e-17,
290 7.9861581325531005859e-02, +0,
291 7.8040568735575632786e-02, -3.8810063021216721741e-16,
292 7.8199386596679687500e-02, +0,
293 7.6412797391314235540e-02, +4.1246529500495762995e-16,
294 7.6561868190765380859e-02, +0,
295 7.4807854772808823896e-02, -3.7025599052186724156e-16,
296 7.4947714805603027344e-02, +0,
297 7.3224639528778112663e-02, +4.2209138483206712401e-17,
298 7.3355793952941894531e-02, +0,
299 7.1661929761571485642e-02, -3.2074473649855177622e-16,
300 7.1784853935241699219e-02, +0,
301 7.0118738881148168218e-02, -2.5371257235753296804e-16,
302 7.0233881473541259766e-02, +0,
303 6.8594137996416115755e-02, +3.3796987842548399135e-16,
304 6.8701922893524169922e-02, +0,
305 6.7087137393172291411e-02, +5.50614922696328852397e-17,
306 6.7187964916229248047e-02, +0,
307 6.5596983299946565182e-02, -2.1580863111502565280e-16,
308 6.5691232681274414062e-02, +0,
309 6.4122802037412718335e-02, -3.1315661827469233434e-16,
310 6.4210832118988037109e-02, +0,
311 6.2426231582525915087e-02, -2.5758980071296622188e-16,
312 6.2507450580596923828e-02, +0,
313 6.0781559928021700046e-02, +1.3736899336217710591e-16,
314 6.0856521129608154297e-02, +0,
315 5.9432882624005145544e-02, +2.2246097394328856474e-16,
316 5.9502959251403808594e-02, +0,
317 5.8132551274581167888e-02, -6.2525053236379489390e-18,
318 5.8198124170303344727e-02, +0,
319 5.6876611930681164608e-02, -2.6589930995607417149e-16,
320 5.6938022375106811523e-02, +0,
321 5.5661522654748551986e-02, -4.2736362859832186197e-16,

322 5.5719077587127685547e-02, +0,
323 5.4484124463757943602e-02, -1.6708067365310384253e-16,
324 5.4538100957870483398e-02, +0,
325 5.3341582449436764080e-02, +3.3271673004611311850e-17,
326 5.3392231464385986328e-02, +0,
327 5.2231267345892007370e-02, -3.5593396674200571616e-16,
328 5.2278816699981689453e-02, +0,
329 5.1150874758829623090e-02, +1.4432815841187114832e-16,
330 5.1195532083511352539e-02, +0,
331 5.0098306612679444072e-02, +9.4680943793589404083e-17,
332 5.0140261650085449219e-02, +0,
333 4.9071641675614507960e-02, +2.1131168520301896817e-16,
334 4.9111068248748779297e-02, +0,
335 4.8069135772851545596e-02, +1.6035336741307516296e-16,
336 4.8106193542480468750e-02, +0,
337 4.7089192241088539959e-02, -2.2491738698796901479e-16,
338 4.7124028205871582031e-02, +0,
339 4.6130362086062248750e-02, -1.5111423469578965206e-16,
340 4.6163111925125122070e-02, +0,
341 4.5191314382707403752e-02, +4.1989325207399786612e-16,
342 4.5222103595733642578e-02, +0,
343 4.4270836390474244126e-02, -4.1432635292331004454e-16,
344 4.4299781322479248047e-02, +0,
345 4.3367774164955186222e-02, -3.0615383054587355892e-16,
346 4.3394982814788818359e-02, +0,
347 4.2481121875321825598e-02, -3.6730166956273555173e-16,
348 4.2506694793701171875e-02, +0,
349 4.1609902899457651415e-02, -4.4226425958068821782e-16,
350 4.1633933782577514648e-02, +0,
351 4.0753259129372665370e-02, +1.9801161516527046872e-16,
352 4.0775835514068603516e-02, +0,
353 3.9910361780060910064e-02, +8.2560620036613164573e-18,
354 3.9931565523147583008e-02, +0,
355 3.9080441183869218946e-02, +3.9908991939242971628e-17,
356 3.9100348949432373047e-02, +0,
357 3.8262816593271686827e-02, +9.5182237812195590276e-17,
358 3.8281500339508056641e-02, +0,
359 3.7456806948784837630e-02, +1.5213508760679563439e-16,
360 3.7474334239959716797e-02, +0,
361 3.6661849947035918262e-02, +7.3335516005184616486e-17,
362 3.6678284406661987305e-02, +0,
363 3.5877353272533163420e-02, -1.3007348019891714540e-16,
364 3.5892754793167114258e-02, +0,
365 3.5102754135096780885e-02, -2.9903662298950558656e-16,
366 3.5117179155349731445e-02, +0,
367 3.4337638360670830195e-02, +2.9656295131966114331e-16,
368 3.4351140260696411133e-02, +0,
369 3.3581472523789734907e-02, +3.4810947205572817820e-16,
370 3.3594101667404174805e-02, +0,
371 3.2833871859357266487e-02, -3.8885440174405159838e-16,
372 3.2845675945281982422e-02, +0,
373 3.2094421679560447558e-02, +5.8805134853032009978e-17,
374 3.2105445861816406250e-02, +0,
375 3.1243584858944295490e-02, +2.8737383773884313066e-17,
376 3.1253755092620849609e-02, +0,
377 0, 0, 0, 0
378 };
unchanged portion omitted

new/usr/src/lib/libm/common/C/atan2pi.c

1

```
*****
1345 Tue Nov 25 12:56:42 2014
new/usr/src/lib/libm/common/C/atan2pi.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak atan2pi = __atan2pi
```

```
29 /*
30 * atan2pi(x) = atan2(x)/pi
31 */

33 #include "libm.h"

35 static const double invpi = 0.3183098861837906715377675;

37 double
38 atan2pi(double y, double x) {
39     int    ix, iy;

41     if (x == 0.0 && y == 0.0) {
42         ix = ((int *)&x)[HIWORD];
43         iy = ((int *)&y)[HIWORD];
44         if (ix >= 0)
45             return (y);
46         return ((iy >= 0)? 1.0 : -1.0);
47     }
48     return (atan2(y, x) * invpi);
49 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/atanh.c

1

```
*****
2061 Tue Nov 25 12:56:42 2014
new/usr/src/lib/libm/common/C/atanh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __atanh = atanh
30 #pragma weak atanh = __atanh

32 /* INDENT OFF */
33 /*
34  * atanh(x)
35  * Code originated from 4.3bsd.
36  * Modified by K.C. Ng for SUN 4.0 libm.
37  * Method :
38  *
39  *      atanh(x) = --- * log(1 + -----) = 0.5 * loglp(2 * -----)
40  *                  2          1 - x          1 - x
41  * Note: to guarantee atanh(-x) = -atanh(x), we use
42  *      sign(x)      |x|
43  *      atanh(x) = ----- * loglp(2*-----).
44  *                  2          1 - |x|
45  *
46  * Special cases:
47  *      atanh(x) is NaN if |x| > 1 with signal;
48  *      atanh(NaN) is that NaN with no signal;
49  *      atanh(+/-1) is +/-INF with signal.
50 */
51 /* INDENT ON */

53 #include "libm.h"
54 #include "libm_synonyms.h"
54 #include "libm_protos.h"
55 #include <math.h>
```

new/usr/src/lib/libm/common/C/atanh.c

2

```
57 double
58 atanh(double x) {
59     double t;

61     if (isnan(x))
62         return (x * x);          /* switched from x + x for Cheetah */
63     t = fabs(x);
64     if (t > 1.0)
65         return (_SVID_libm_err(x, x, 30));    /* sNaN */
66     if (t == 1.0)
67         return (_SVID_libm_err(x, x, 31));    /* x/0; */
68     t = t / (1.0 - t);
69     return (copysign(0.5, x) * loglp(t + t));
70 }
_____unchanged_portion_omitted_
```

new/usr/src/lib/libm/common/C/ceil.c

1

```
*****
1725 Tue Nov 25 12:56:43 2014
new/usr/src/lib/libm/common/C/ceil.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __ceil = ceil
29 #pragma weak ceil = __ceil
30
31 /*
32 * ceil(x) returns the least integral value bigger than or equal to x.
33 * NOTE: ceil(x) returns result with the same sign as x's, including 0.
34 *
35 * Modified 8/4/04 for performance.
36 */
37
38 #include "libm.h"
39
40 static const double
41     zero = 0.0,
42     one = 1.0,
43     two52 = 4503599627370496.0;
44
45 double
46 ceil(double x) {
47     double t, w;
48     int hx, lx, ix;
49
50     hx = ((int *)&x)[HIWORD];
51     lx = ((int *)&x)[LOWORD];
52     ix = hx & ~0x80000000;
53     if (ix >= 0x43300000) /* return x if |x| >= 2^52, or x is NaN */
54         return (x * one);
55     t = (hx >= 0)? two52 : -two52;
56     w = x + t;
57     t = w - t;
```

new/usr/src/lib/libm/common/C/ceil.c

2

```
58     if (ix < 0x3ff00000) {
59         if ((ix | lx) == 0)
60             return (x);
61         else
62             return ((hx < 0)? -zero : one);
63     }
64     return ((t >= x)? t : t + one);
65 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/copysign.c

1

1200 Tue Nov 25 12:56:43 2014

new/usr/src/lib/libm/common/C/copysign.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __copysign = copysign
30 #if defined(ELFOSBJ)
31 #pragma weak copysign = __copysign
32 #endif
```

```
32 #include "libm.h"
```

```
34 double
35 copysign(double x, double y) {
36     int hx, hy;
37
38     hx = ((int *) &x)[HIWORD];
39     hy = ((int *) &y)[HIWORD];
40     return (hx ^ hy) >= 0 ? (x) : (-x);
41 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/cos.c

1

```
*****
6131 Tue Nov 25 12:56:44 2014
new/usr/src/lib/libm/common/C/cos.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28 #pragma weak __cos = cos
29 #pragma weak cos = __cos
30
31 /* INDENT OFF */
32 /*
33  * cos(x)
34  * Accurate Table look-up algorithm by K.C. Ng, May, 1995.
35  *
36  * Algorithm: see sincos.c
37  */
38
39 #include "libm.h"
40
41 static const double sc[] = {
42 /* ONE = */ 1.0,
43 /* NONE = */ -1.0,
44 /*
45  * |sin(x) - (x+pp1*x^3+pp2*x^5)| <= 2^-58.79 for |x| < 0.008
46  */
47 /* PP1 = */ -0.1666666666666316558867252052378889521480627858683055567,
48 /* PP2 = */ .0083333315652997472323564894248466758248475374977974017927,
49 /*
50  * |(sin(x) - (x+p1*x^3+...+p4*x^9)|
51  * ----- | <= 2^-57.63 for |x| < 0.1953125
52  * x
53  */
54 /* P1 = */ -1.666666666666629669805215138920301589656e-0001,
55 /* P2 = */ 8.333333332390951295683993455280336376663e-0003,
56 /* P3 = */ -1.984126237997976692791551778230098403960e-0004,
57 /* P4 = */ 2.753403624854277237649987622848330351110e-0006,
```

new/usr/src/lib/libm/common/C/cos.c

2

```
58 /*
59  * |cos(x) - (1+qq1*x^2+qq2*x^4)| <= 2^-55.99 for |x| <= 0.008 (0x3f80624d)
60 */
61 /* QQ1 = */ -0.4999999999975492381842911981948418542742729,
62 /* QQ2 = */ 0.041666542904352059294545209158357640398771740,
63 /* Q1 = */ -0.5,
64 /* Q2 = */ 4.166666666500350703680945520860748617445e-0002,
65 /* Q3 = */ -1.388888596436972210694266290577848696006e-0003,
66 /* Q4 = */ 2.478563078858589473679519517892953492192e-0005,
67 /* PI02_H = */ 1.570796326794896557999,
68 /* PI02_L = */ 6.123233995736765886130e-17,
69 /* PI02_L0 = */ 6.123233995727922165564e-17,
70 /* PI02_L1 = */ 8.843720566135701120255e-29,
71 /* PI302_H = */ 4.712388980384689673997,
72 /* PI302_L = */ 1.836970198721029765839e-16,
73 /* PI302_L0 = */ 1.836970198720396133587e-16,
74 /* PI302_L1 = */ 6.336322524749201142226e-29,
75 /* PI502_H = */ 7.853981633974482789995,
76 /* PI502_L = */ 3.061616997868382943065e-16,
77 /* PI502_L0 = */ 3.061616997861941598865e-16,
78 /* PI502_L1 = */ 6.441344200433640781982e-28,
79 */;
-----
unchanged portion omitted
```


new/usr/src/lib/libm/common/C/cosh.c

1

```
*****
2452 Tue Nov 25 12:56:44 2014
new/usr/src/lib/libm/common/C/cosh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __cosh = cosh
29 #pragma weak cosh = __cosh

31 /* INDENT OFF */
32 /*
33  * cosh(x)
34  * Code originated from 4.3bsd.
35  * Modified by K.C. Ng for SUN 4.0 libm.
36  * Method :
37  *     1. Replace x by |x| (cosh(x) = cosh(-x)).
38  *     2.
39  *
40  * 0      <= x <= 0.3465 : cosh(x) := 1 + -----
41  *                                     [ exp(x) - 1 ]^2
42  *
43  *                                     exp(x) + 1/exp(x)
44  * 0.3465 <= x <= 22    : cosh(x) := -----
45  *
46  * 22      <= x <= lnovft : cosh(x) := exp(x)/2
47  * lnovft <= x < INF    : cosh(x) := scalbn(exp(x-1024*ln2),1023)
48  *
49  * Note: .3465 is a number near one half of ln2.
50  *
51  * Special cases:
52  *     cosh(x) is |x| if x is +INF, -INF, or NaN.
53  *     only cosh(0)=1 is exact for finite x.
54  */
55 /* INDENT ON */

57 #include "libm.h"
```

new/usr/src/lib/libm/common/C/cosh.c

2

```
59 static const double
60     ln2 = 6.93147180559945286227e-01,
61     ln2hi = 6.93147180369123816490e-01,
62     ln2lo = 1.90821492927058770002e-10,
63     lnovft = 7.09782712893383973096e+02;

65 double
66 cosh(double x) {
67     double t, w;

69     w = fabs(x);
70     if (!finite(w))
71         return (w * w);
72     if (w < 0.3465) {
73         t = expml(w);
74         w = 1.0 + t;
75         if (w != 1.0)
76             w = 1.0 + (t * t) / (w + w);
77         return (w);
78     } else if (w < 22.0) {
79         t = exp(w);
80         return (0.5 * (t + 1.0 / t));
81     } else if (w <= lnovft) {
82         return (0.5 * exp(w));
83     } else {
84         w = (w - 1024 * ln2hi) - 1024 * ln2lo;
85         if (w >= ln2)
86             return (_SVID_libm_err(x, x, 5));
87         else
88             return (scalbn(exp(w), 1023));
89     }
90 }

_____unchanged_portion_omitted_____
```

```

*****
13821 Tue Nov 25 12:56:44 2014
new/usr/src/lib/libm/common/C/erf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __erf = erf
31 #pragma weak __erfc = erfc
30 #pragma weak erf = __erf
31 #pragma weak erfc = __erfc

33 /* INDENT OFF */
34 /*
35  * double erf(double x)
36  * double erfc(double x)
37  *
38  *          x
39  * erf(x) = ----- \int_0^x exp(-t*t)dt
40  *          sqrt(pi)
41  *
42  *
43  * erfc(x) = 1-erf(x)
44  * Note that
45  *          erf(-x) = -erf(x)
46  *          erfc(-x) = 2 - erfc(x)
47  *
48  * Method:
49  * 1. For |x| in [0, 0.84375]
50  *   erf(x) = x + x*R(x^2)
51  *   erfc(x) = 1 - erf(x)          if x in [-.84375,0.25]
52  *   = 0.5 + ((0.5-x)-x*R)       if x in [0.25,0.84375]
53  *   where R = P/Q where P is an odd poly of degree 8 and
54  *   Q is an odd poly of degree 10.
55  *
56  *          -57.90
57  *          | R - (erf(x)-x)/x | <= 2

```

```

57 *
58 *
59 * Remark. The formula is derived by noting
60 *   erf(x) = (2/sqrt(pi))*(x - x^3/3 + x^5/10 - x^7/42 + ...)
61 * and that
62 *   2/sqrt(pi) = 1.128379167095512573896158903121545171688
63 * is close to one. The interval is chosen because the fix
64 * point of erf(x) is near 0.6174 (i.e., erf(x)=x when x is
65 * near 0.6174), and by some experiment, 0.84375 is chosen to
66 * guarantee the error is less than one ulp for erf.
67 *
68 *
69 * 2. For |x| in [0.84375,1.25], let s = |x| - 1, and
70 *   c = 0.84506291151 rounded to single (24 bits)
71 *   erf(x) = sign(x) * (c + P1(s)/Q1(s))
72 *   erfc(x) = (1-c) - P1(s)/Q1(s) if x > 0
73 *             1+(c+P1(s)/Q1(s)) if x < 0
74 *   |P1/Q1 - (erf(|x|)-c)| <= 2**-59.06
75 * Remark: here we use the taylor series expansion at x=1.
76 *   erf(1+s) = erf(1) + s*Poly(s)
77 *             = 0.845.. + P1(s)/Q1(s)
78 * That is, we use rational approximation to approximate
79 *   erf(1+s) - (c + (single)0.84506291151)
80 * Note that |P1/Q1| < 0.078 for x in [0.84375,1.25]
81 * where
82 *   P1(s) = degree 6 poly in s
83 *   Q1(s) = degree 6 poly in s
84 *
85 * 3. For x in [1.25,1/0.35(-2.857143)],
86 *   erfc(x) = (1/x)*exp(-x*x-0.5625+R1/S1)
87 *   erf(x) = 1 - erfc(x)
88 * where
89 *   R1(z) = degree 7 poly in z, (z=1/x^2)
90 *   S1(z) = degree 8 poly in z
91 *
92 * 4. For x in [1/0.35,28]
93 *   erfc(x) = (1/x)*exp(-x*x-0.5625+R2/S2) if x > 0
94 *           = 2.0 - (1/x)*exp(-x*x-0.5625+R2/S2) if -6<x<0
95 *           = 2.0 - tiny (if x <= -6)
96 *   erf(x) = sign(x)*(1.0 - erfc(x)) if x < 6, else
97 *   erf(x) = sign(x)*(1.0 - tiny)
98 * where
99 *   R2(z) = degree 6 poly in z, (z=1/x^2)
100 *   S2(z) = degree 7 poly in z
101 *
102 * Notel:
103 * To compute exp(-x*x-0.5625+R/S), let s be a single
104 * precision number and s := x; then
105 *   -x*x = -s*s + (s-x)*(s+x)
106 *   exp(-x*x-0.5625+R/S) =
107 *       exp(-s*s-0.5625)*exp((s-x)*(s+x)+R/S);
108 *
109 * Note2:
110 * Here 4 and 5 make use of the asymptotic series
111 *   exp(-x*x)
112 *   erfc(x) ~ ----- * ( 1 + Poly(1/x^2) )
113 *             x*sqrt(pi)
114 * We use rational approximation to approximate
115 *   g(s)=f(1/x^2) = log(erfc(x)*x) - x*x + 0.5625
116 * Here is the error bound for R1/S1 and R2/S2
117 *   |R1/S1 - f(x)| < 2**(-62.57)
118 *   |R2/S2 - f(x)| < 2**(-61.52)
119 *
120 * 5. For inf > x >= 28
121 *   erf(x) = sign(x) *(1 - tiny) (raise inexact)
122 *   erfc(x) = tiny*tiny (raise underflow) if x > 0
123 *           = 2 - tiny if x<0

```

```

123 *      7. Special case:
124 *          erf(0) = 0, erf(inf) = 1, erf(-inf) = -1,
125 *          erfc(0) = 1, erfc(inf) = 0, erfc(-inf) = 2,
126 *          erfc/erf(NaN) is NaN
127 */
128 /* INDENT ON */

130 #include "libm_synonyms.h" /* __erf, __erfc, __exp */
130 #include "libm_macros.h"
131 #include <math.h>

133 static const double xxx[] = {
134 /* tiny */          1e-300,
135 /* half */         5.0000000000000000000000e-01, /* 3FE00000, 00000000 */
136 /* one */          1.0000000000000000000000e+00, /* 3FF00000, 00000000 */
137 /* two */          2.0000000000000000000000e+00, /* 40000000, 00000000 */
138 /* erx */          8.45062911510467529297e-01, /* 3FE0AC1, 60000000 */
139 */
140 * Coefficients for approximation to erf on [0,0.84375]
141 */
142 /* efx */          1.28379167095512586316e-01, /* 3FC06EBA, 8214DB69 */
143 /* efx8 */         1.02703333676410069053e+00, /* 3FF06EBA, 8214DB69 */
144 /* pp0 */          1.28379167095512558561e-01, /* 3FC06EBA, 8214DB68 */
145 /* pp1 */          -3.25042107247001499370e-01, /* BFD4CD7D, 691CB913 */
146 /* pp2 */          -2.84817495755985104766e-02, /* BF9D2A51, DBD7194F */
147 /* pp3 */          -5.77027029648944159157e-03, /* BF77A291, 236668E4 */
148 /* pp4 */          -2.37630166566501626084e-05, /* BEF8EAD6, 120016AC */
149 /* qq1 */          3.97917223959155352819e-01, /* 3FD97779, CDDADC09 */
150 /* qq2 */          6.50222499887672944485e-02, /* 3FB0A54C, 5536CEBA */
151 /* qq3 */          5.08130628187576562776e-03, /* 3F74D022, C4D36B0F */
152 /* qq4 */          1.32494738004321644526e-04, /* 3F215DC9, 221C1A10 */
153 /* qq5 */          -3.96022827877536812320e-06, /* BED09C43, 42A26120 */
154 */
155 * Coefficients for approximation to erf in [0.84375,1.25]
156 */
157 /* pa0 */          -2.36211856075265944077e-03, /* BF6359B8, BEF77538 */
158 /* pa1 */          4.14856118683748331666e-01, /* 3FDA8D00, AD92B34D */
159 /* pa2 */          -3.72207876035701323847e-01, /* BFD7D240, FBB8C3F1 */
160 /* pa3 */          3.18346619901161753674e-01, /* 3FD45FCA, 805120E4 */
161 /* pa4 */          -1.10894694282396677476e-01, /* BFBC6398, 3D3E28EC */
162 /* pa5 */          3.54783043256182359371e-02, /* 3FA22A36, 599795EB */
163 /* pa6 */          -2.16637559486879084300e-03, /* BF61BF38, 0A96073F */
164 /* qa1 */          1.06420880400844228286e-01, /* 3FBB3E66, 18EEE323 */
165 /* qa2 */          5.40397917702171048937e-01, /* 3FE14AF0, 92EB6F33 */
166 /* qa3 */          7.18286544141962662868e-02, /* 3FB2635C, D99FE9A7 */
167 /* qa4 */          1.261712198087616422112e-01, /* 3FC02660, E763351F */
168 /* qa5 */          1.36370839120290507362e-02, /* 3F8BEDC2, 6B51DD1C */
169 /* qa6 */          1.19844998467991074170e-02, /* 3F888B54, 5735151D */
170 */
171 * Coefficients for approximation to erfc in [1.25,1/0.35]
172 */
173 /* ra0 */          -9.86494403484714822705e-03, /* BF843412, 600D6435 */
174 /* ra1 */          -6.93858572707181764372e-01, /* BFE63416, E4BA7360 */
175 /* ra2 */          -1.05586262253232909814e+01, /* C0251E04, 41B0E726 */
176 /* ra3 */          -6.23753324503260060396e+01, /* C04F300A, E4CBA38D */
177 /* ra4 */          -1.62396669462573470355e+02, /* C0644CB1, 84282266 */
178 /* ra5 */          -1.84605092906711035994e+02, /* C067135C, EBCCABB2 */
179 /* ra6 */          -8.12874355063065934246e+01, /* C0545265, 57E4D2F2 */
180 /* ra7 */          -9.81432934416914548592e+00, /* C023A0EF, C69AC25C */
181 /* sa1 */          1.96512716674392571292e+01, /* 4033A6B9, BD707687 */
182 /* sa2 */          1.37657754143519042600e+02, /* 4061350C, 526AE721 */
183 /* sa3 */          4.34565877475229228821e+02, /* 407B290D, D58A1A71 */
184 /* sa4 */          6.45387271733267880336e+02, /* 40842B19, 21EC2868 */
185 /* sa5 */          4.29008140027567833386e+02, /* 407AD021, 57700314 */
186 /* sa6 */          1.08635005541779435134e+02, /* 405B28A3, EE48AE2C */
187 /* sa7 */          6.57024977031928170135e+00, /* 401A47EF, 8E484A93 */

```

```

188 /* sa8 */          -6.04244152148580987438e-02, /* BFAEEFF2, EE749A62 */
189 */
190 * Coefficients for approximation to erfc in [1/.35,28]
191 */
192 /* rb0 */          -9.86494292470009928597e-03, /* BF843412, 39E86F4A */
193 /* rb1 */          -7.99283237680523006574e-01, /* BFE993BA, 70C285DE */
194 /* rb2 */          -1.77579549177547519889e+01, /* C031C209, 555F995A */
195 /* rb3 */          -1.60636384855821916062e+02, /* C064145D, 43C5ED98 */
196 /* rb4 */          -6.37566443368389627722e+02, /* C083EC88, 1375F228 */
197 /* rb5 */          -1.02509513161107724954e+03, /* C0900461, 6A2E5992 */
198 /* rb6 */          -4.83519191608651397019e+02, /* C07E384E, 9BDC383F */
199 /* sb1 */          3.03380607434824582924e+01, /* 403E568B, 261D5190 */
200 /* sb2 */          3.25792512996573918826e+02, /* 40745CAE, 221B9F0A */
201 /* sb3 */          1.53672958608443695994e+03, /* 409802EB, 189D5118 */
202 /* sb4 */          3.19985821950859553908e+03, /* 40A8FFB7, 688C246A */
203 /* sb5 */          2.55305040643316442583e+03, /* 40A3F219, CEDF3BE6 */
204 /* sb6 */          4.74528541206955367215e+02, /* 407DA874, E79FE763 */
205 /* sb7 */          -2.24409524465858183362e+01, /* C03670E2, 42712D62 */
206 */

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/exp.c

1

```
*****
14865 Tue Nov 25 12:56:45 2014
new/usr/src/lib/libm/common/C/exp.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */
28 #pragma weak __exp = exp
29 #pragma weak exp = __exp
30
31 /*
32  * exp(x)
33  * Hybrid algorithm of Peter Tang's Table driven method (for large
34  * arguments) and an accurate table (for small arguments).
35  * Written by K.C. Ng, November 1988.
36  * Method (large arguments):
37  * 1. Argument Reduction: given the input x, find r and integer k
38  *    and j such that
39  *        x = (k+j/32)*(ln2) + r, |r| <= (1/64)*ln2
40  *
41  * 2. exp(x) = 2^k * (2^(j/32) + 2^(j/32)*expml(r))
42  *    a. expml(r) is approximated by a polynomial:
43  *        expml(r) ~ r + t1*r^2 + t2*r^3 + ... + t5*r^6
44  *        Here t1 = 1/2 exactly.
45  *    b. 2^(j/32) is represented to twice double precision
46  *        as TBL[2j]+TBL[2j+1].
47  *
48  * Note: If divide were fast enough, we could use another approximation
49  *    in 2.a:
50  *        expml(r) ~ (2r)/(2-R), R = r - r^2*(t1 + t2*r^2)
51  *        (for the same t1 and t2 as above)
52  *
53  * Special cases:
54  *    exp(INF) is INF, exp(NaN) is NaN;
55  *    exp(-INF) = 0;
56  *    for finite argument, only exp(0)=1 is exact.
57  */
```

new/usr/src/lib/libm/common/C/exp.c

2

```
58 * Accuracy:
59 * According to an error analysis, the error is always less than
60 * an ulp (unit in the last place). The largest errors observed
61 * are less than 0.55 ulp for normal results and less than 0.75 ulp
62 * for subnormal results.
63 *
64 * Misc. info.
65 * For IEEE double
66 * if x > 7.09782712893383973096e+02 then exp(x) overflow
67 * if x < -7.45133219101941108420e+02 then exp(x) underflow
68 */
70 #include "libm.h"
71
72 static const double TBL[] = {
73 1.000000000000000000000000e+00, 0.0000000000000000000000e+00,
74 1.02189714865411662714e+00, 5.10922502897344389359e-17,
75 1.04427378242741375480e+00, 8.55188970553796365958e-17,
76 1.06714040067682369717e+00, -7.89985396684158212226e-17,
77 1.09050773266525768967e+00, -3.04678207981247114697e-17,
78 1.11438674259589243221e+00, 1.04102784568455709549e-16,
79 1.13878863475669156458e+00, 8.91281267602540777782e-17,
80 1.1637248587757747552e+00, 3.82920483692409349872e-17,
81 1.18920711500272102690e+00, 3.98201523146564611098e-17,
82 1.21524735998046895524e+00, -7.71263069268148813091e-17,
83 1.24185781207348400201e+00, 4.65802759183693679123e-17,
84 1.26905095719173321989e+00, 2.66793213134218609523e-18,
85 1.29683955465100964055e+00, 2.53825027948883149593e-17,
86 1.32523664315974132322e+00, -2.85873121003886075697e-17,
87 1.35425554693689265129e+00, 7.70094837980298946162e-17,
88 1.38390988196383202258e+00, -6.77051165879478628716e-17,
89 1.41421356237309514547e+00, -9.66729331345291345105e-17,
90 1.44518080697704665027e+00, -3.02375813499398731940e-17,
91 1.47682614593949934623e+00, -3.48399455689279579579e-17,
92 1.50916442759342284141e+00, -1.01645532775429503911e-16,
93 1.54221082540794074411e+00, 7.94983480969762085616e-17,
94 1.57598084510788649659e+00, -1.01369164712783039808e-17,
95 1.61049033194925428347e+00, 2.47071925697978878522e-17,
96 1.64575547815396494578e+00, -1.01256799136747726038e-16,
97 1.68179283050742900407e+00, 8.19901002058149652013e-17,
98 1.71861929812247793414e+00, -1.85138041826311098821e-17,
99 1.75625216037329945351e+00, 2.96014069544887330703e-17,
100 1.79470907500310716820e+00, 1.82274584279120867698e-17,
101 1.83400808640934243066e+00, 3.28310722424562658722e-17,
102 1.87416763411029996256e+00, -6.12276341300414256164e-17,
103 1.91520656139714740007e+00, -1.06199460561959626376e-16,
104 1.95714412417540017941e+00, 8.96076779103666776760e-17,
105 };
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/exp10.c

1

2581 Tue Nov 25 12:56:45 2014

new/usr/src/lib/libm/common/C/exp10.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak exp10 = __exp10
```

```
29 /* INDENT OFF */
30 /*
31 * exp10(x)
32 * Code by K.C. Ng for SUN 4.0 libm.
33 * Method :
34 *     n = nint(x*(log10/log2));
35 *     exp10(x) = 10**x = exp(x*ln(10)) = exp(n*ln2+(x*ln10-n*ln2))
36 *             = 2**n*exp(ln10*(x-n*log2/log10))
37 *     If x is an integer < 23 then use repeat multiplication. For
38 *     10**22 is the largest representable integer.
39 */
40 /* INDENT ON */
```

```
42 #include "libm.h"
```

```
44 static const double C[] = {
45     3.3219280948736234787, /* log(10)/log(2) */
46     2.3025850929940456840, /* log(10) */
47     3.0102999565860955045E-1, /* log(2)/log(10) high */
48     5.3716447674669983622E-12, /* log(2)/log(10) low */
49     0.0,
50     0.5,
51     1.0,
52     10.0,
53     1.0e300,
54     1.0e-300,
55 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/exp2.c

1

2067 Tue Nov 25 12:56:46 2014

new/usr/src/lib/libm/common/C/exp2.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __exp2 = exp2
```

```
29 #pragma weak exp2 = __exp2
```

```
31 /* INDENT OFF */
32 /*
33  * exp2(x)
34  * Code by K.C. Ng for SUN 4.0 libm.
35  * Method :
36  *     exp2(x) = 2**x = 2**((x-anint(x))+anint(x))
37  *             = 2**anint(x)*2**(x-anint(x))
38  *             = 2**anint(x)*exp((x-anint(x))*ln2)
39  */
40 /* INDENT ON */
```

```
42 #include "libm.h"
```

```
44 static const double C[] = {
45     0.0,
46     1.0,
47     0.5,
48     6.93147180559945286227e-01,
49     1.0e300,
50     1.0e-300,
51 };
```

unchanged_portion_omitted


```
124 * constants. The decimal values may be used, provided that the
125 * compiler will convert from decimal to binary accurately enough
126 * to produce the hexadecimal values shown.
127 */
128 /* INDENT ON */

130 #include "libm_synonyms.h" /* __expm1 */
130 #include "libm_macros.h"
131 #include <math.h>

133 static const double xxx[] = {
134 /* one */ 1.0,
135 /* huge */ 1.0e+300,
136 /* tiny */ 1.0e-300,
137 /* o_threshold */ 7.09782712893383973096e+02, /* 40862E42 FEFA39EF */
138 /* ln2_hi */ 6.93147180369123816490e-01, /* 3FE62E42 FEE00000 */
139 /* ln2_lo */ 1.90821492927058770002e-10, /* 3DEA39EF 35793C76 */
140 /* invln2 */ 1.44269504088896338700e+00, /* 3FF71547 652B82FE */
141 /* scaled coefficients related to expm1 */
142 /* Q1 */ -3.333333333333331316428e-02, /* BFA11111 111110F4 */
143 /* Q2 */ 1.58730158725481460165e-03, /* 3F5A01A0 19FE5585 */
144 /* Q3 */ -7.93650757867487942473e-05, /* BF14CE19 9EAADB7 */
145 /* Q4 */ 4.00821782732936239552e-06, /* 3ED0CFCA 86E65239 */
146 /* Q5 */ -2.01099218183624371326e-07 /* BE8AFDB7 6E09C32D */
147 };
unchanged_portion_omitted
```


new/usr/src/lib/libm/common/C/fabs.c

1

```
*****
1178 Tue Nov 25 12:56:46 2014
new/usr/src/lib/libm/common/C/fabs.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __fabs = fabs
30 #pragma weak fabs = __fabs
```

```
32 #include "libm.h"
33 #include "libm_synonyms.h"
33 #include "libm_macros.h"
34 #include <math.h>
```

```
36 double
37 fabs(double x) {
38     int *px = (int *) &x;

40     px[HIWORD] &= ~0x80000000;
41     return (x);
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/floor.c

1

```
*****
1730 Tue Nov 25 12:56:47 2014
new/usr/src/lib/libm/common/C/floor.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __floor = floor
30 #pragma weak floor = __floor
31
32 /*
33 * floor(x) returns the biggest integral value less than or equal to x.
34 * NOTE: floor(x) returns result with the same sign as x's, including 0.
35 * Modified 8/4/04 for performance.
36 */
37
38 #include "libm.h"
39
40 static const double
41     zero = 0.0,
42     one = 1.0,
43     two52 = 4503599627370496.0;
44
45 double
46 floor(double x) {
47     double t, w;
48     int hx, lx, ix;
49
50     hx = ((int *)&x)[HIWORD];
51     lx = ((int *)&x)[LOWORD];
52     ix = hx & ~0x80000000;
53     if (ix >= 0x43300000) /* return x if |x| >= 2^52, or x is NaN */
54         return (x * one);
55     t = (hx >= 0)? two52 : -two52;
56     w = x + t;
57     t = w - t;
```

new/usr/src/lib/libm/common/C/floor.c

2

```
58     if (ix < 0x3ff00000) {
59         if ((ix | lx) == 0)
60             return (x);
61         else
62             return ((hx < 0)? -one : zero);
63     }
64     return ((t <= x)? t : t - one);
65 }
_____unchanged_portion_omitted_____
```

```

*****
3207 Tue Nov 25 12:56:47 2014
new/usr/src/lib/libm/common/C/fmod.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __fmod = fmod
29 #pragma weak fmod = __fmod

31 #include "libm.h"

33 static const double zero = 0.0;

35 /*
36 * The following implementation assumes fast 64-bit integer arith-
37 * metic. This is fine for sparc because we build libm in v8plus
38 * mode. It's also fine for sparcv9 and amd64, although we have
39 * assembly code on amd64. For x86, it would be better to use
40 * 32-bit code, but we have assembly for x86, too.
41 */
42 double
43 fmod(double x, double y) {
44     double      w;
45     long long   hx, ix, iy, iz;
46     int         nd, k, ny;

48     hx = *(long long *)&x;
49     ix = hx & ~0x8000000000000000ull;
50     iy = *(long long *)&y & ~0x8000000000000000ull;

52     /* handle special cases */
53     if (iy == 0ll)
54         return (_SVID_libm_err(x, y, 27));

56     if (ix >= 0x7ff0000000000000ll || iy > 0x7ff0000000000000ll)
57         return ((x * y) * zero);

```

```

59     if (ix <= iy)
60         return ((ix < iy)? x : x * zero);

62     /*
63     * Set:
64     *   ny = true exponent of y
65     *   nd = true exponent of x minus true exponent of y
66     *   ix = normalized significand of x
67     *   iy = normalized significand of y
68     */
69     ny = iy >> 52;
70     k = ix >> 52;
71     if (ny == 0) {
72         /* y is subnormal, x could be normal or subnormal */
73         ny = 1;
74         while (iy < 0x0010000000000000ll) {
75             ny -= 1;
76             iy += iy;
77         }
78         nd = k - ny;
79         if (k == 0) {
80             nd += 1;
81             while (ix < 0x0010000000000000ll) {
82                 nd -= 1;
83                 ix += ix;
84             }
85         } else {
86             ix = 0x0010000000000000ll | (ix & 0x000ffffffffffffll);
87         }
88     } else {
89         /* both x and y are normal */
90         nd = k - ny;
91         ix = 0x0010000000000000ll | (ix & 0x000ffffffffffffll);
92         iy = 0x0010000000000000ll | (iy & 0x000ffffffffffffll);
93     }

95     /* perform fixed point mod */
96     while (nd-- > 0) {
97         iz = ix - iy;
98         if (iz >= 0)
99             ix = iz;
100         ix += ix;
101     }
102     iz = ix - iy;
103     if (iz >= 0)
104         ix = iz;

106     /* convert back to floating point and restore the sign */
107     if (ix == 0ll)
108         return (x * zero);
109     while (ix < 0x0010000000000000ll) {
110         ix += ix;
111         ny -= 1;
112     }
113     while (ix > 0x0020000000000000ll) { /* XXX can this ever happen? */
114         ny += 1;
115         ix >>= 1;
116     }
117     if (ny <= 0) {
118         /* result is subnormal */
119         k = -ny + 1;
120         ix >>= k;
121         *(long long *)&w = (hx & 0x8000000000000000ull) | ix;
122         return (w);
123     }

```

new/usr/src/lib/libm/common/C/fmod.c

3

```
124     *(long long *)&w = (hx & 0x8000000000000000ull) |  
125     ((long long)ny << 52) | (ix & 0x000ffffffffffff1l);  
126     return (w);  
127 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/C/gamma.c

1

```
*****
1333 Tue Nov 25 12:56:48 2014
new/usr/src/lib/libm/common/C/gamma.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __gamma = gamma
29 #pragma weak gamma = __gamma
```

```
31 #include "libm.h"
```

```
33 extern int signgam;
```

```
35 double
```

```
36 gamma(double x) {
37     double g;
```

```
39     if (!finite(x))
40         return (x * x);
```

```
42     g = rint(x);
43     if (x == g && x <= 0.0) {
44         signgam = 1;
45         return (_SVID_libm_err(x, x, 41));
46     }
```

```
48     g = __k_lgamma(x, &signgam);
49     if (!finite(g))
50         g = _SVID_libm_err(x, x, 40);
51     return (g);
52 }
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/gamma_r.c

1

1126 Tue Nov 25 12:56:48 2014

new/usr/src/lib/libm/common/C/gamma_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __gamma_r = gamma_r
29 #pragma weak gamma_r = __gamma_r
```

```
31 #include "libm.h"
```

```
33 double
34 gamma_r(double x, int *signgamp) {
35     return (lgamma_r(x, signgamp));
36 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/C/hypot.c

1

```
*****
5483 Tue Nov 25 12:56:48 2014
new/usr/src/lib/libm/common/C/hypot.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __hypot = hypot
30 #if defined(ELFOBJ)
31 #pragma weak hypot = __hypot
32 #endif
33
34 /* INDENT OFF */
35 /*
36  * Hypot(x, y)
37  * by K.C. Ng for SUN 4.0 libm, updated 3/11/2003.
38  * Method :
39  * A. When rounding is rounded-to-nearest:
40  *   If z = x * x + y * y has error less than sqrt(2) / 2 ulp than
41  *   sqrt(z) has error less than 1 ulp.
42  *   So, compute sqrt(x*x+y*y) with some care as follows:
43  *   Assume x > y > 0;
44  *   1. Check whether save and set rounding to round-to-nearest
45  *   2. if x > 2y use
46  *       xh*xh+(y*y+((x-xh)*(x+xh))) for x*x+y*y
47  *   where xh = x with lower 32 bits cleared; else
48  *   3. if x <= 2y use
49  *       x2h*yh+((x-y)*(x-y)+(x2h*(y-yh)+(x2-x2h)*y))
50  *   where x2 = 2*x, x2h = 2x with lower 32 bits cleared, yh = y with
51  *   lower 32 bits chopped.
```

new/usr/src/lib/libm/common/C/hypot.c

2

```
50 *
51 * B. When rounding is not rounded-to-nearest:
52 *   The following (magic) formula will yield an error less than 1 ulp.
53 *   z = sqrt(x * x + y * y)
54 *   hypot(x, y) = x + (y / ((x + z) / y))
55 *
56 * NOTE: DO NOT remove parenthesis!
57 *
58 * Special cases:
59 *   hypot(x, y) is INF if x or y is +INF or -INF; else
60 *   hypot(x, y) is NAN if x or y is NAN.
61 *
62 * Accuracy:
63 *   hypot(x, y) returns sqrt(x^2+y^2) with error less than 1 ulps
64 *   (units in the last place)
65 */
66
67 #include "libm.h"
68
69 static const double
70 zero = 0.0,
71 onep1u = 1.00000000000000022204e+00, /* 0x3ff00000 1 = 1+2**-52 */
72 twom53 = 1.11022302462515654042e-16, /* 0x3ca00000 0 = 2**-53 */
73 twom768 = 6.441148769597133308e-232, /* 2^-768 */
74 two768 = 1.552518092300708935e+231; /* 2^768 */
75
76 /* INDENT ON */
77
78 double
79 hypot(double x, double y) {
80     double xh, yh, w, ax, ay;
81     int i, j, nx, ny, ix, iy, iscale = 0;
82     unsigned lx, ly;
83
84     ix = ((int *) &x)[HIWORD] & ~0x80000000;
85     lx = ((int *) &x)[LOWORD];
86     iy = ((int *) &y)[HIWORD] & ~0x80000000;
87     ly = ((int *) &y)[LOWORD];
88 /*
89  * Force ax = |x| ~>~ ay = |y|
90  */
91     if (iy > ix) {
92         ax = fabs(y);
93         ay = fabs(x);
94         i = ix;
95         ix = iy;
96         iy = i;
97         l = lx;
98         lx = ly;
99         ly = i;
100    } else {
101        ax = fabs(x);
102        ay = fabs(y);
103    }
104    nx = ix >> 20;
105    ny = iy >> 20;
106    j = nx - ny;
107 /*
108  * x >= 2^500 (x*x or y*y may overflow)
109  */
110    if (nx >= 0x5f3) {
111        if (nx == 0x7ff) { /* inf or NaN, signal of sNaN */
112            if (((ix - 0x7ff00000) | lx) == 0)
113                return (ax == ay ? ay : ax);
114            else if (((iy - 0x7ff00000) | ly) == 0)
115                return (ay == ax ? ax : ay);
```

```

116         else
117             return (ax * ay);          /* + -> * for Cheetah */
118     } else if (j > 32) {                /* x >> y */
119         if (j <= 53)
120             ay *= twom53;
121         ax += ay;
122         if (((int *) &ax)[HIWORD] == 0x7ff00000)
123             ax = _SVID_libm_err(x, y, 4);
124         return (ax);
125     }
126     ax *= twom768;
127     ay *= twom768;
128     iscale = 2;
129     ix -= 768 << 20;
130     iy -= 768 << 20;
131 }
132 /*
133 * y < 2^-450 (x*x or y*y may underflow)
134 */
135     else if (ny < 0x23d) {
136         if ((ix | lx) == 0)
137             return (ay);
138         if ((iy | ly) == 0)
139             return (ax);
140         if (j > 53)                /* x >> y */
141             return (ax + ay);
142         iscale = 1;
143         ax *= two768;
144         ay *= two768;
145         if (nx == 0) {
146             if (ax == zero) /* guard subnormal flush to zero */
147                 return (ax);
148             ix = ((int *) &ax)[HIWORD];
149         } else
150             ix += 768 << 20;
151         if (ny == 0) {
152             if (ay == zero) /* guard subnormal flush to zero */
153                 return (ax * twom768);
154             iy = ((int *) &ay)[HIWORD];
155         } else
156             iy += 768 << 20;
157         j = (ix >> 20) - (iy >> 20);
158         if (j > 32) {                /* x >> y */
159             if (j <= 53)
160                 ay *= twom53;
161             return ((ax + ay) * twom768);
162         }
163     } else if (j > 32) {                /* x >> y */
164         if (j <= 53)
165             ay *= twom53;
166         return (ax + ay);
167     }
168 /*
169 * Medium range ax and ay with max{|ax/ay|, |ay/ax|} bounded by 2^32
170 * First check rounding mode by comparing oneplu*oneplu with oneplu+twom53.
171 * Make sure the computation is done at run-time.
172 */
173     if (((lx | ly) << 5) == 0) {
174         ay = ay * ay;
175         ax += ay / (ax + sqrt(ax * ax + ay));
176     } else
177     if (oneplu * oneplu != oneplu + twom53) {
178         /* round-to-zero, positive, negative mode */
179         /* magic formula with less than an ulp error */
180         w = sqrt(ax * ax + ay * ay);
181         ax += ay / ((ax + w) / ay);

```

```

182     } else {
183         /* round-to-nearest mode */
184         w = ax - ay;
185         if (w > ay) {
186             ((int *) &xh)[HIWORD] = ix;
187             ((int *) &xh)[LOWORD] = 0;
188             ay = ay * ay + (ax - xh) * (ax + xh);
189             ax = sqrt(xh * xh + ay);
190         } else {
191             ax = ax + ax;
192             ((int *) &xh)[HIWORD] = ix + 0x00100000;
193             ((int *) &xh)[LOWORD] = 0;
194             ((int *) &yh)[HIWORD] = iy;
195             ((int *) &yh)[LOWORD] = 0;
196             ay = w * w + ((ax - xh) * yh + (ay - yh) * ax);
197             ax = sqrt(xh * yh + ay);
198         }
199     }
200     if (iscale > 0) {
201         if (iscale == 1)
202             ax *= twom768;
203         else {
204             ax *= two768;          /* must generate side effect here */
205             if (((int *) &ax)[HIWORD] == 0x7ff00000)
206                 ax = _SVID_libm_err(x, y, 4);
207         }
208     }
209     return (ax);
210 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/ilogb.c

1

```
*****
2226 Tue Nov 25 12:56:49 2014
new/usr/src/lib/libm/common/C/ilogb.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __ilogb = ilogb
30 #if defined(ELFOBJ)
31 #pragma weak ilogb = __ilogb
32 #endif

32 #include "libm.h"
33 #include "xpg6.h"

35 #if defined(__x86)
37 #if defined(USE_FPSCALE) || defined(__x86)
36 static const double two52 = 4503599627370496.0;
37 #else
38 /*
39  * v: high part of a non-zero subnormal |x|; w: low part of |x|
40  */
41 static int
42 ilogb_subnormal(unsigned v, unsigned w) {
43     int r = -1022 - 52;

45     if (v)
46         r += 32;
47     else
48         v = w;
```

new/usr/src/lib/libm/common/C/ilogb.c

2

```
49     if (v & 0xffff0000)
50         r += 16, v >>= 16;
51     if (v & 0xff00)
52         r += 8, v >>= 8;
53     if (v & 0xf0)
54         r += 4, v >>= 4;
55     v <<= 1;
56     return (r + ((0xffffaa50 >> v) & 0x3));
57 }
58 #endif /* defined(__x86) */
60 #endif /* defined(USE_FPSCALE) */

60 static int
61 raise_invalid(int v) { /* SUSv3 requires ilogb(0,+/-Inf,NaN) raise invalid */
62 #ifndef lint
63     if ((__xpg6 & _C99SUSv3_ilogb_0InfNaN_raises_invalid) != 0) {
64         static const double zero = 0.0;
65         volatile double dummy;

67         dummy = zero / zero;
68     }
69 #endif
70     return (v);
71 }

73 int
74 ilogb(double x) {
75     int *px = (int *) &x, k = px[HIWORD] & ~0x80000000;

77     if (k < 0x00100000) {
78         if ((px[LOWORD] | k) == 0)
79             return (raise_invalid(0x80000001));
80     } else {
81 #if defined(__x86)
82         x *= two52;
83         return (((px[HIWORD] & 0x7ff00000) >> 20) - 1075);
84 #else
85         return (ilogb_subnormal(k, px[LOWORD]));
86 #endif
87     } else if (k < 0x7ff00000)
88         return ((k >> 20) - 1023);
89     else
90         return (raise_invalid(0x7fffffff));
91 }
92 }

    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/j0.c

1

```
*****
8797 Tue Nov 25 12:56:49 2014
new/usr/src/lib/libm/common/C/j0.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * Floating point Bessel's function of the first and second kinds
32  * of order zero: j0(x),y0(x);
33  *
34  * Special cases:
35  *   y0(0)=y1(0)=yn(n,0) = -inf with division by zero signal;
36  *   y0(-ve)=y1(-ve)=yn(n,-ve) are NaN with invalid signal.
37 */
38
39 #pragma weak __j0 = j0
40 #pragma weak __y0 = y0
39 #pragma weak j0 = __j0
40 #pragma weak y0 = __y0
41
42 #include "libm.h"
43 #include "libm_synonyms.h"
43 #include "libm_protos.h"
44 #include <math.h>
45 #include <values.h>
46
47 #define GENERIC double
48 static const GENERIC
49 zero = 0.0,
50 small = 1.0e-5,
51 tiny = 1.0e-18,
52 one = 1.0,
53 eight = 8.0,
54 invsqrtpi = 5.641895835477562869480794515607725858441e-0001,
55 tpi = 0.636619772367581343075535053490057448;
```

new/usr/src/lib/libm/common/C/j0.c

2

```
57 static GENERIC pzero(GENERIC), qzero(GENERIC);
58 static const GENERIC r0[4] = { /* [1.e-5, 1.28] */
59   -2.500000000000003622131880894830476755537e-0001,
60   1.095597547334830263234433855932375353303e-0002,
61   -1.819734750463320921799187258987098087697e-0004,
62   9.977001946806131657544212501069893930846e-0007,
63 };
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/j1.c

1

9167 Tue Nov 25 12:56:50 2014

new/usr/src/lib/libm/common/C/j1.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 /*
31  * floating point Bessel's function of the first and second kinds
32  * of order zero: j1(x),y1(x);
33  *
34  * Special cases:
35  *   y0(0)=y1(0)=yn(n,0) = -inf with division by zero signal;
36  *   y0(-ve)=y1(-ve)=yn(n,-ve) are NaN with invalid signal.
37 */
```

```
39 #pragma weak __j1 = j1
40 #pragma weak __y1 = y1
39 #pragma weak j1 = __j1
40 #pragma weak y1 = __y1
```

```
42 #include "libm.h"
43 #include "libm_synonyms.h"
43 #include "libm_protos.h"
44 #include <math.h>
45 #include <values.h>
```

```
47 #define GENERIC double
48 static const GENERIC
49 zero = 0.0,
50 small = 1.0e-5,
51 tiny = 1.0e-20,
52 one = 1.0,
53 invsqrtpi = 5.641895835477562869480794515607725858441e-0001,
54 tpi = 0.636619772367581343075535053490057448;
```

new/usr/src/lib/libm/common/C/j1.c

2

```
56 static GENERIC pone(GENERIC), qone(GENERIC);
57 static const GENERIC r0[4] = {
58     -6.2500000000000002203053200981413218949548e-0002,
59     1.600998455640072901321605101981501263762e-0003,
60     -1.963888815948313758552511884390162864930e-0005,
61     8.263917341093549759781339713418201620998e-0008,
62 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/jn.c

1

```
*****
7423 Tue Nov 25 12:56:50 2014
new/usr/src/lib/libm/common/C/jn.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __jn = jn
31 #pragma weak __yn = yn
32 #pragma weak jn = __jn
33 #pragma weak yn = __yn
34
35 /*
36  * floating point Bessel's function of the 1st and 2nd kind
37  * of order n: jn(n,x), yn(n,x);
38  *
39  * Special cases:
40  * y0(0)=y1(0)=yn(n,0) = -inf with division by zero signal;
41  * y0(-ve)=y1(-ve)=yn(n,-ve) are NaN with invalid signal.
42  * Note 2. About jn(n,x), yn(n,x)
43  * For n=0, j0(x) is called,
44  * for n=1, j1(x) is called,
45  * for n<x, forward recursion is used starting
46  * from values of j0(x) and j1(x).
47  * for n>x, a continued fraction approximation to
48  * j(n,x)/j(n-1,x) is evaluated and then backward
49  * recursion is used starting from a supposed value
50  * for j(n,x). The resulting value of j(0,x) is
51  * compared with the actual value to correct the
52  * supposed value of j(n,x).
53  *
54  * yn(n,x) is similar in all respects, except
55  * that forward recursion is used for all
56  * values of n>1.
57 */
```

new/usr/src/lib/libm/common/C/jn.c

2

```
58 #include "libm.h"
59 #include <float.h> /* DBL_MIN */
60 #include <values.h> /* X_TLOSS */
61 #include "xpg6.h" /* __xpg6 */
62
63 #define GENERIC double
64
65 static const GENERIC
66 invsqrtpi = 5.641895835477562869480794515607725858441e-0001,
67 two = 2.0,
68 zero = 0.0,
69 one = 1.0;
70
71 GENERIC
72 jn(int n, GENERIC x) {
73     int i, sgn;
74     GENERIC a, b, temp = 0;
75     GENERIC z, w, ox, on;
76
77     /*
78      * J(-n,x) = (-1)^n * J(n, x), J(n, -x) = (-1)^n * J(n, x)
79      * Thus, J(-n,x) = J(n,-x)
80      */
81     ox = x; on = (GENERIC)n;
82     if (n < 0) {
83         n = -n;
84         x = -x;
85     }
86     if (isnan(x))
87         return (x*x); /* + -> * for Cheetah */
88     if (!(((int) _lib_version == libm_ieee ||
89         (__xpg6 & _C99SUSv3_math_errexcept) != 0)) {
90         if (fabs(x) > X_TLOSS)
91             return (_SVID_libm_err(on, ox, 38));
92     }
93     if (n == 0)
94         return (j0(x));
95     if (n == 1)
96         return (j1(x));
97     if ((n&1) == 0)
98         sgn = 0; /* even n */
99     else
100         sgn = signbit(x); /* odd n */
101     x = fabs(x);
102     if (x == zero || !finite(x)) b = zero;
103     else if ((GENERIC)n <= x) {
104         /*
105          * Safe to use
106          * J(n+1,x)=2n/x *J(n,x)-J(n-1,x)
107          */
108         if (x > 1.0e91) {
109             /*
110              * x >> n**2
111              * Jn(x) = cos(x-(2n+1)*pi/4)*sqrt(2/x*pi)
112              * Yn(x) = sin(x-(2n+1)*pi/4)*sqrt(2/x*pi)
113              * Let s=sin(x), c=cos(x),
114              * xn=x-(2n+1)*pi/4, sqrt2 = sqrt(2), then
115              *
116              *          n      sin(xn)*sqrt2      cos(xn)*sqrt2
117              *          -----
118              *          0      s-c                  c+s
119              *          1      -s-c                 -c+s
120              *          2      -s+c                 -c-s
121              *          3      s+c                  c-s
122              */
123         }
124     }
125 }
```

```

123     switch (n&3) {
124         case 0: temp = cos(x)+sin(x); break;
125         case 1: temp = -cos(x)+sin(x); break;
126         case 2: temp = -cos(x)-sin(x); break;
127         case 3: temp = cos(x)-sin(x); break;
128     }
129     b = invsqrtpi*temp/sqrt(x);
130 } else {
131     a = j0(x);
132     b = j1(x);
133     for (i = 1; i < n; i++) {
134         temp = b;
135         b = b*((GENERIC)(i+i)/x) - a; /* avoid underflow */
136         a = temp;
137     }
138 }
139 } else {
140     if (x < 1e-9) { /* use J(n,x) = 1/n!*(x/2)^n */
141         b = pow(0.5*x, (GENERIC) n);
142         if (b != zero) {
143             for (a = one, i = 1; i <= n; i++) a *= (GENERIC)i;
144             b = b/a;
145         }
146     } else {
147         /*
148         * use backward recurrence
149         *
150         *  $J(n,x)/J(n-1,x) = \frac{x}{2n} - \frac{x^2}{2(n+1)} - \frac{x^2}{2(n+2)} \dots$ 
151         *
152         *
153         * (for large x) =  $\frac{1}{2n} - \frac{1}{2(n+1)} - \frac{1}{2(n+2)} \dots$ 
154         *
155         *
156         *
157         *
158         * Let w = 2n/x and h = 2/x, then the above quotient
159         * is equal to the continued fraction:
160         *
161         * 
$$= \frac{1}{w - \frac{1}{w+h - \frac{1}{w+2h - \dots}}}$$

162         *
163         *
164         *
165         *
166         *
167         *
168         * To determine how many terms needed, let
169         * Q(0) = w, Q(1) = w(w+h) - 1,
170         * Q(k) = (w+k*h)*Q(k-1) - Q(k-2),
171         * When Q(k) > 1e4 good for single
172         * When Q(k) > 1e9 good for double
173         * When Q(k) > 1e17 good for quaduple
174         */
175     }
176     /* determin k */
177     GENERIC t, v;
178     double q0, q1, h, tmp; int k, m;
179     w = (n+n)/(double)x; h = 2.0/(double)x;
180     q0 = w; z = w + h; q1 = w*z - 1.0; k = 1;
181     while (q1 < 1.0e9) {
182         k += 1; z += h;
183         tmp = z*q1 - q0;
184         q0 = q1;
185         q1 = tmp;
186     }
187     m = n+n;
188     for (t = zero, i = 2*(n+k); i >= m; i -= 2) t = one/(i/x-t);

```

```

189     a = t;
190     b = one;
191     /*
192     * estimate log((2/x)^n*n!) = n*log(2/x)+n*ln(n)
193     * hence, if n*(log(2n/x)) > ...
194     * single 8.8722839355e+01
195     * double 7.09782712893383973096e+02
196     * long double 1.1356523406294143949491931077970765006170e+04
197     * then recurrent value may overflow and the result is
198     * likely underflow to zero
199     */
200     tmp = n;
201     v = two/x;
202     tmp = tmp*log(fabs(v*tmp));
203     if (tmp < 7.09782712893383973096e+02) {
204         for (i = n-1; i > 0; i--) {
205             temp = b;
206             b = ((i+i)/x)*b - a;
207             a = temp;
208         }
209     } else {
210         for (i = n-1; i > 0; i--) {
211             temp = b;
212             b = ((i+i)/x)*b - a;
213             a = temp;
214             if (b > 1e100) {
215                 a /= b;
216                 t /= b;
217                 b = 1.0;
218             }
219         }
220     }
221     b = (t*j0(x)/b);
222 }
223 }
224 if (sgn == 1)
225     return (-b);
226 else
227     return (b);
228 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/lgamma.c

1

1336 Tue Nov 25 12:56:51 2014

new/usr/src/lib/libm/common/C/lgamma.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __lgamma = lgamma
```

```
29 #pragma weak lgamma = __lgamma
```

```
31 #include "libm.h"
```

```
33 extern int signgam;
```

```
35 double
```

```
36 lgamma(double x) {
37     double g;
```

```
39     if (!finite(x))
40         return (x * x);
```

```
42     g = rint(x);
43     if (x == g && x <= 0.0) {
44         signgam = 1;
45         return (_SVID_libm_err(x, x, 15));
46     }
```

```
48     g = __k_lgamma(x, &signgam);
49     if (!finite(g))
50         g = _SVID_libm_err(x, x, 14);
51     return (g);
52 }
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/lgamma_r.c

1

1336 Tue Nov 25 12:56:51 2014

new/usr/src/lib/libm/common/C/lgamma_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __lgamma_r = lgamma_r
```

```
29 #pragma weak lgamma_r = __lgamma_r
```

```
31 #include "libm.h"
```

```
33 double
34 lgamma_r(double x, int *signgamp) {
35     double g;
36
37     if (isnan(x))
38         return (x * x);
39
40     g = rint(x);
41     if (x == g && x <= 0.0) {
42         *signgamp = 1;
43         return (_SVID_libm_err(x, x, 15));
44     }
45
46     g = __k_lgamma(x, signgamp);
47     if (!finite(g))
48         g = _SVID_libm_err(x, x, 14);
49     return (g);
50 }
```

unchanged_portion_omitted

```

*****
5100 Tue Nov 25 12:56:51 2014
new/usr/src/lib/libm/common/C/libm.h
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #ifndef _LIBM_H
30 #define _LIBM_H

32 #include <sys/isa_defs.h>

34 #ifdef _ASM
35 /* BEGIN CSTYLED */

37 /*
38 * Disable amd64 assembly code profiling for now.
39 */
40 #if defined(__amd64)
41 #undef PROF
42 #endif

44 #include <sys/asm_linkage.h>

46 #define NAME(x) x
47 #define TEXT .section ".text"
48 #define DATA .section ".data"
49 #define RO_DATA .section ".rodata"
50 #define IDENT(x) .ident x

52 #if defined(__sparc)

54 #define LIBM_ANSI_PRAGMA_WEAK(sym,stype) \
55     .weak __/**/sym; \
56     .type __/**/sym,#stype; \
57     __/**/sym = sym
55     .weak sym; \

```

```

56     .type sym,#stype; \
57     = __/**/sym

59 #ifndef SET_FILE
60 #define SET_FILE(x) \
61     .file x
62 #endif /* !defined(SET_FILE) */

64 #ifdef PIC
65 /*
66  * One should *never* pass o7 to PIC_SETUP.
67 */
68 #define PIC_SETUP(via) \
69 9: call 8f; \
70     sethi %hi(NAME(_GLOBAL_OFFSET_TABLE_)-(9b-.)),%via; \
71 8: or %via,%lo(NAME(_GLOBAL_OFFSET_TABLE_)-(9b-.)),%via; \
72     add %via,%o7,%via
73 /*
74  * Must save/restore %o7 in leaf routines; may *not* use jmp!
75 */
76 #define PIC_LEAF_SETUP(via) \
77 or %g0,%o7,%gl; \
78 9: call 8f; \
79     sethi %hi(NAME(_GLOBAL_OFFSET_TABLE_)-(9b-.)),%via; \
80 8: or %via,%lo(NAME(_GLOBAL_OFFSET_TABLE_)-(9b-.)),%via; \
81     add %via,%o7,%via; \
82     or %g0,%gl,%o7
83 #ifdef __sparcv9
84 #define PIC_SET(via,sym,dst) ldx [%via+sym],%dst
85 #else /* defined(__sparcv9) */
86 #define PIC_SET(via,sym,dst) ld [%via+sym],%dst
87 #endif /* defined(__sparcv9) */
88 #else /* defined(PIC) */
89 #define PIC_SETUP(via)
90 #define PIC_LEAF_SETUP(via)
91 #ifdef __sparcv9
92 /*
93  * %gl is used as scratch register in V9 mode
94 */
95 #define PIC_SET(via,sym,dst) setx sym,%gl,%dst
96 #else /* defined(__sparcv9) */
97 #define PIC_SET(via,sym,dst) set sym,%dst
98 #endif /* defined(__sparcv9) */
99 #endif /* defined(PIC) */

101 /*
102  * Workaround for 4337025: MCOUNT in asm_linkage.h does not support __sparcv9
103 */
104 #if defined(PROF) && defined(__sparcv9)

106 #undef MCOUNT_SIZE
107 #undef MCOUNT

109 #if !defined(PIC)
110 #define MCOUNT_SIZE (9*4) /* 9 instructions */
111 #define MCOUNT(x) \
112     save %sp, -SA(MINFRAME), %sp; \
113     sethi %hh(.L_**/x/**/1), %o0; \
114     sethi %lm(.L_**/x/**/1), %o1; \
115     or %o0, %hm(.L_**/x/**/1), %o0; \
116     or %o1, %lo(.L_**/x/**/1), %o1; \
117     sllx %o0, 32, %o0; \
118     call _mcount; \
119     or %o0, %o1, %o0; \
120     restore; \
121     .common .L_**/x/**/1, 8, 8

```



```

122 #elif defined(PIC32)
123 #define MCOUNT_SIZE      (10*4) /* 10 instructions */
124 #define MCOUNT(x) \
125     save    %sp,-SA(MINFRAME),%sp; \
126 1:        call    .+8; \
127     sethi   %hi(_GLOBAL_OFFSET_TABLE_-(1b-.)),%o0; \
128     sethi   %hi(.L/**/x/**/1),%o1; \
129     add     %o0,%lo(_GLOBAL_OFFSET_TABLE_-(1b-.)),%o0; \
130     add     %o1,%lo(.L/**/x/**/1),%o1; \
131     add     %o0,%o7,%o0; \
132     call    _mcount; \
133     ldx     [%o0+%o1],%o0; \
134     restore; \
135     .common .L/**/x/**/1,8,8
136 #else /* PIC13 */
137 #define MCOUNT_SIZE      (8*4) /* 8 instructions */
138 #define MCOUNT(x) \
139     save    %sp,-SA(MINFRAME),%sp; \
140 1:        call    .+8; \
141     sethi   %hi(_GLOBAL_OFFSET_TABLE_-(1b-.)),%o0; \
142     add     %o0,%lo(_GLOBAL_OFFSET_TABLE_-(1b-.)),%o0; \
143     add     %o0,%o7,%o0; \
144     call    _mcount; \
145     ldx     [%o0+%lo(.L/**/x/**/1)],%o0; \
146     restore; \
147     .common .L/**/x/**/1,8,8
148 #endif /* !defined(PIC) */
149 #endif /* defined(PROF) && defined(__sparcv9) */

151 #elif defined(__x86)

153 #define LIBM_ANSI_PRAGMA_WEAK(sym,stype) \
154     .weak  __/**/sym; \
155     .type  __/**/sym,@stype; \
156     __/**/sym = sym
154     .weak sym; \
155     .type sym,@stype; \
156     sym = __/**/sym

158 #ifndef PIC
159 #if defined(__amd64)
160 #define PIC_SETUP(x)
161 #define PIC_WRAPUP
162 #define PIC_F(x)          x@PLT
163 #define PIC_G(x)          x@GOTPCREL(%rip)
164 #define PIC_L(x)          x(%rip)
165 #define PIC_G_LOAD(insn,sym,dst) \
166     movq   PIC_G(sym),%dst; \
167     insn   (%dst),%dst
168 #else
169 #define PIC_SETUP(label) \
170     pushl  %ebx; \
171     call   .label; \
172     .label: popl  %ebx; \
173     addl   $_GLOBAL_OFFSET_TABLE_+[.-.label],%ebx
174 #define PIC_WRAPUP        popl  %ebx
175 #define PIC_F(x)          x@PLT
176 #define PIC_G(x)          x@GOT(%ebx)
177 #define PIC_L(x)          x@GOTOFF(%ebx)
178 #define PIC_G_LOAD(insn,sym,dst) \
179     mov    PIC_G(sym),%dst; \
180     insn   (%dst),%dst
181 #endif
182 #else /* defined(PIC) */
183 #define PIC_SETUP(x)
184 #define PIC_WRAPUP

```

```

185 #define PIC_F(x)          x
186 #define PIC_G(x)          x
187 #define PIC_L(x)          x
188 #define PIC_G_LOAD(insn,sym,dst)      insn   sym,%dst
189 #endif /* defined(PIC) */

191 #else
192 #error Unknown architecture
193 #endif

195 /* END CSTYLED */
196 #else /* defined(_ASM) */

198 #include "libm_macros.h"
199 #include "libm_synonyms.h"
199 #include "libm_protos.h"
200 #include "libm_inlines.h"
201 #include <math.h>
202 #if defined(__SUNPRO_C)
203 #include <sunmath.h>
204 #endif

206 #endif /* defined(_ASM) */

208 #endif /* _LIBM_H */

```

new/usr/src/lib/libm/common/C/libm_protos.h

1

```
*****
5457 Tue Nov 25 12:56:52 2014
new/usr/src/lib/libm/common/C/libm_protos.h
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29 #ifndef _C_LIBM_PROTOS_H
30 #define _C_LIBM_PROTOS_H

32 /*
33  * Many symbols used to be namespaced with __libm to prevent collisions. All
34  * but these two were otherwise scoped local and directly bound, so that
35  * collision could not occur.
36  *
37  * For reasons unknown, these two are global (but private).
38  */
39 #ifndef LIBMOPT_BUILD
40 #define __TBL_cos __libmopt_TBL_cos
41 #define __TBL_exp2_512 __libmopt_TBL_exp2_512
42 #define __TBL_ipio2_inf __libmopt_TBL_ipio2_inf
43 #define __TBL_jlog_n1 __libmopt_TBL_jlog_n1
44 #define __TBL_jlog_n2 __libmopt_TBL_jlog_n2
45 #define __TBL_jlog_p1 __libmopt_TBL_jlog_p1
46 #define __TBL_jlog_p2 __libmopt_TBL_jlog_p2
47 #define __TBL_log10 __libmopt_TBL_log10
48 #define __TBL_log2_14 __libmopt_TBL_log2_14
49 #define __TBL_log2_9 __libmopt_TBL_log2_9
50 #define __TBL_sin __libmopt_TBL_sin
51 #define __TBL_sincos __libmopt_TBL_sincos
52 #define __TBL_xexp __libmopt_TBL_xexp
53 #define __TBL_xlog __libmopt_TBL_xlog
54 #define __k_cos __libmopt_k_cos
55 #define __k_sin __libmopt_k_sin
56 #define __k_sincos __libmopt_k_sincos
57 #endif
58 #endif
```

new/usr/src/lib/libm/common/C/libm_protos.h

2

```
50 #define __reduction __libmopt_reduction
51 #define __rem_pio2 __libmopt_rem_pio2
52 #define __rem_pio2m __libmopt_rem_pio2m
53 #else /* defined(LIBMOPT_BUILD) */
54 #ifdef LIBM_BUILD
55 #define __SVID_libm_err __libm_SVID_libm_err /* not used by -lsunmath */
56 #define __TBL_atan __libm_TBL_atan
57 #define __TBL_atan1 __libm_TBL_atan1
58 #define __TBL_atan_hi __libm_TBL_atan_hi /* not used by -lsunmath */
59 #define __TBL_atan_lo __libm_TBL_atan_lo /* not used by -lsunmath */
60 #define __TBL_exp2_hi __libm_TBL_exp2_hi /* not used by -lsunmath */
61 #define __TBL_exp2_lo __libm_TBL_exp2_lo /* not used by -lsunmath */
62 #define __TBL_ipio2_inf __libm_TBL_ipio2_inf
63 #define __TBL_log __libm_TBL_log
64 #define __TBL_log2_hi __libm_TBL_log2_hi /* not used by -lsunmath */
65 #define __TBL_log2_lo __libm_TBL_log2_lo /* not used by -lsunmath */
66 #define __TBL_log_hi __libm_TBL_log_hi /* not used by -lsunmath */
67 #define __TBL_log_lo __libm_TBL_log_lo /* not used by -lsunmath */
68 #define __TBL_sincos __libm_TBL_sincos
69 #define __TBL_sincosx __libm_TBL_sincosx
70 #define __TBL_tan_hi __libm_TBL_tan_hi /* not used by -lsunmath */
71 #define __TBL_tan_lo __libm_TBL_tan_lo /* not used by -lsunmath */
72 #define __k_cexp __libm_k_cexp /* C99 libm */
73 #define __k_cexpl __libm_k_cexpl /* C99 libm */
74 #define __k_clog_r __libm_k_clog_r /* C99 libm */
75 #define __k_clog_rl __libm_k_clog_rl /* C99 libm */
76 #define __k_atan2 __libm_k_atan2 /* C99 libm */
77 #define __k_atan2l __libm_k_atan2l /* C99 libm */
78 #define __k_cos __libm_k_cos
79 #define __k_lgamma __libm_k_lgamma
80 #define __k_sin __libm_k_sin
81 #define __k_sincos __libm_k_sincos
82 #define __k_tan __libm_k_tan
83 #define __reduction __libm_reduction /* i386 only */
84 #define __rem_pio2 __libm_rem_pio2
85 #define __rem_pio2m __libm_rem_pio2m
86 #define __k_cosf __libm_k_cosf /* C99 libm */
87 #define __k_cosl __libm_k_cosl /* C99 libm */
88 #define __k_lgammal __libm_k_lgammal /* C99 libm */
89 #define __k_sincosf __libm_k_sincosf /* C99 libm */
90 #define __k_sincosl __libm_k_sincosl /* C99 libm */
91 #define __k_sinf __libm_k_sinf /* C99 libm */
92 #define __k_sinl __libm_k_sinl /* C99 libm */
93 #define __k_tanf __libm_k_tanf /* C99 libm */
94 #define __k_tanl __libm_k_tanl /* C99 libm */
95 #define __poly_libmq __libm_poly_libmq /* C99 libm */
96 #define __rem_pio2l __libm_rem_pio2l /* C99 libm */
97 #define __TBL_atanl_hi __libm_TBL_atanl_hi /* C99 libm */
98 #define __TBL_atanl_lo __libm_TBL_atanl_lo /* C99 libm */
99 #define __TBL_cosl_hi __libm_TBL_cosl_hi /* C99 libm */
100 #define __TBL_cosl_lo __libm_TBL_cosl_lo /* C99 libm */
101 #define __TBL_expl_hi __libm_TBL_expl_hi /* C99 libm */
102 #define __TBL_expl_lo __libm_TBL_expl_lo /* C99 libm */
103 #define __TBL_expm1l __libm_TBL_expm1l /* C99 libm */
104 #define __TBL_expm1lx __libm_TBL_expm1lx /* C99 libm */
105 #define __TBL_ipio2l_inf __libm_TBL_ipio2l_inf /* C99 libm */
106 #define __TBL_logl_hi __libm_TBL_logl_hi /* C99 libm */
107 #define __TBL_logl_lo __libm_TBL_logl_lo /* C99 libm */
108 #define __TBL_r_atan_hi __libm_TBL_r_atan_hi /* C99 libm */
109 #define __TBL_r_atan_lo __libm_TBL_r_atan_lo /* C99 libm */
110 #define __TBL_sinl_hi __libm_TBL_sinl_hi /* C99 libm */
111 #define __TBL_sinl_lo __libm_TBL_sinl_lo /* C99 libm */
112 #define __TBL_tanl_hi __libm_TBL_tanl_hi /* C99 libm */
113 #define __TBL_tanl_lo __libm_TBL_tanl_lo /* C99 libm */
114 #endif /* defined(LIBM_BUILD) */
115 #endif /* defined(LIBMOPT_BUILD) */
```

```

42 #ifndef _ASM
43 #ifdef __STDC__
44 #define __P(p) p
45 #else
46 #define __P(p) ()
47 #endif

49 #include <sys/ieeefp.h>

51 extern double _SVID_libm_err __P((double, double, int));
52 extern double __k_cos __P((double, double));
53 extern double __k_cos_ __P((double *));
54 extern double __k_lgamma __P((double, int *));
55 extern double __k_sin __P((double, double));
56 extern double __k_sin_ __P((double *));
57 extern double __k_sincos __P((double, double, double *));
58 extern double __k_sincos_ __P((double *, double *));
59 extern double __k_tan __P((double, double, int));
60 extern double __k_cexp __P((double, int *));
61 extern long double __k_cexpl __P((long double, int *));
62 extern double __k_clog_r __P((double, double, double *));
63 extern long double __k_clog_rl __P((long double, long double, long double *));
64 extern double __k_atan2 __P((double, double, double *));
65 extern long double __k_atan2l __P((long double, long double, long double *));
66 extern int __rem_pio2 __P((double, double *));
67 extern int __rem_pio2m __P((double *, double *, int, int, int, const int *));

69 /*
70 * entry points that are in-lined
71 */
72 extern double copysign __P((double, double));
73 extern int finite __P((double));
74 extern enum fp_class_type fp_class __P((double));
75 extern double infinity __P((void));
76 extern int isinf __P((double));
77 extern int signbit __P((double));

79 /*
80 * new C99 entry points
81 */
82 extern double fdim __P((double, double));
83 extern double fma __P((double, double, double));
84 extern double fmax __P((double, double));
85 extern double fmin __P((double, double));
86 extern double frexp __P((double, int *));
87 extern double ldexp __P((double, int));
88 extern double modf __P((double, double *));
89 extern double nan __P((const char *));
90 extern double nearbyint __P((double));
91 extern double nexttoward __P((double, long double));
92 extern double remquo __P((double, double, int *));
93 extern double round __P((double));
94 extern double scalbln __P((double, long int));
95 extern double tgamma __P((double));
96 extern double trunc __P((double));
97 extern float fdimf __P((float, float));
98 extern float fmaf __P((float, float, float));
99 extern float fmaxf __P((float, float));
100 extern float fminf __P((float, float));
101 extern float frexpf __P((float, int *));
102 extern float ldexpf __P((float, int));
103 extern float modff __P((float, float *));
104 extern float nanf __P((const char *));
105 extern float nearbyintf __P((float));
106 extern float nextafterf __P((float, float));

```

```

107 extern float nexttowardf __P((float, long double));
108 extern float remquoof __P((float, float, int *));
109 extern float roundf __P((float));
110 extern float scalblnf __P((float, long int));
111 extern float tgammaf __P((float));
112 extern float truncf __P((float));
113 extern long double frexpl __P((long double, int *));
114 extern long double fdiml __P((long double, long double));
115 extern long double fmal __P((long double, long double, long double));
116 extern long double fmaxl __P((long double, long double));
117 extern long double fminl __P((long double, long double));
118 extern long double ldexpl __P((long double, int));
119 extern long double modfl __P((long double, long double *));
120 extern long double nanl __P((const char *));
121 extern long double nearbyintl __P((long double));
122 extern long double nextafterl __P((long double, long double));
123 extern long double nexttowardl __P((long double, long double));
124 extern long double remquol __P((long double, long double, int *));
125 extern long double roundl __P((long double));
126 extern long double scalblnl __P((long double, long int));
127 extern long double tgamma __P((long double));
128 extern long double trunc __P((long double));
129 extern long int lrint __P((double));
130 extern long int lrintf __P((float));
131 extern long int lrintl __P((long double));
132 extern long int lround __P((double));
133 extern long int lroundf __P((float));
134 extern long int lroundl __P((long double));
135 extern long long int llrint __P((double));
136 extern long long int llrintf __P((float));
137 extern long long int llrintl __P((long double));
138 extern long long int llround __P((double));
139 extern long long int llroundf __P((float));
140 extern long long int llroundl __P((long double));
141 #endif /* _ASM */

143 #endif /* _C_LIBM_PROTOS_H */

```

```

*****
7989 Tue Nov 25 12:56:53 2014
new/usr/src/lib/libm/common/C/log.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2 * CDDL HEADER START
3 *
4 * The contents of this file are subject to the terms of the
5 * Common Development and Distribution License (the "License").
6 * You may not use this file except in compliance with the License.
7 *
8 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

```

```

29 #pragma weak __log = log
29 #pragma weak log = __log

```

```

31 /* INDENT OFF */
32 /*
33 * log(x)
34 * Table look-up algorithm with product polynomial approximation.
35 * By K.C. Ng, Oct 23, 2004. Updated Oct 18, 2005.
36 *
37 * (a). For x in [1-0.125, 1+0.1328125], using a special approximation:
38 * Let f = x - 1 and z = f*f.
39 * return f + ((a1*z) *
40 * ((a2 + (a3*f)*(a4+f)) + (f*z)*(a5+f))) *
41 * (((a6 + f*(a7+f)) + (f*z)*(a8+f)) *
42 * ((a9 + (a10*f)*(a11+f)) + (f*z)*(a12+f)))
43 * a1 -6.88821452420390473170286327331268694251775741577e-0002,
44 * a2 1.97493380704769294631262255279580131173133850098e+0000,
45 * a3 2.24963218866067560242072431719861924648284912109e+0000,
46 * a4 -9.02975906958474405783476868236903101205825805664e-0001,
47 * a5 -1.47391630715542865104339398385491222143173217773e+0000,
48 * a6 1.86846544648220058704168877738993614912033081055e+0000,
49 * a7 1.82277370459347465292410106485476717352867126465e+0000,
50 * a8 1.25295479915214102994980294170090928673744201660e+0000,
51 * a9 1.96709676945198275177517643896862864494323730469e+0000,
52 * a10 -4.00127989749189894030934055990655906498432159424e-0001,
53 * a11 3.0167552858798333733648178167641162872314453125e+0000,
54 * a12 -9.52325445049240770778453679668018594384193420410e-0001,
55 *
56 * with remez error |(log(1+f) - P(f))/f| <= 2**-56.81 and
57 *

```

```

58 * (b). For 0.09375 <= x < 24
59 * Use an 8-bit table look-up (3-bit for exponent and 5 bit for
60 * significand):
61 * Let ix stands for the high part of x in IEEE double format.
62 * Since 0.09375 <= x < 24, we have
63 * 0x3fb80000 <= ix < 0x40380000.
64 * Let j = (ix - 0x3fb80000) >> 15. Then 0 <= j < 256. Choose
65 * a Y[j] such that HIWORD(Y[j]) ~ 0x3fb8400 + (j<<15) (the middle
66 * number between 0x3fb80000 + (j<<15) and 3fb80000 + ((j+1)<<15)),
67 * and at the same time 1/Y[j] as well as log(Y[j]) are very close
68 * to 53-bits floating point numbers.
69 * A table of Y[j], 1/Y[j], and log(Y[j]) are pre-computed and thus
70 * log(x) = log(Y[j]) + log(1 + (x-Y[j])*(1/Y[j]))
71 * = log(Y[j]) + log(1 + s)
72 * where
73 * s = (x-Y[j])*(1/Y[j])
74 * We compute max (x-Y[j])*(1/Y[j]) for the chosen Y[j] and obtain
75 * |s| < 0.0154. By applying remez algorithm with Product Polynomial
76 * Approximation, we find the following approximated of log(1+s)
77 * (b1*s)*(b2+s*(b3+s))*((b4+s*b5)+(s*s)*(b6+s))*(b7+s*(b8+s))
78 * with remez error |log(1+s) - P(s)| <= 2**-63.5
79 *
80 * (c). Otherwise, get "n", the exponent of x, and then normalize x to
81 * z in [1,2). Then similar to (b) find a Y[i] that matches z to 5.5
82 * significant bits. Then
83 * log(x) = n*ln2 + log(Y[i]) + log(z/Y[i]).
84 *
85 * Special cases:
86 * log(x) is NaN with signal if x < 0 (including -INF) ;
87 * log(+INF) is +INF; log(0) is -INF with signal;
88 * log(NaN) is that NaN with no signal.
89 *
90 * Maximum error observed: less than 0.90 ulp
91 *
92 * Constants:
93 * The hexadecimal values are the intended ones for the following constants.
94 * The decimal values may be used, provided that the compiler will convert
95 * from decimal to binary accurately enough to produce the hexadecimal values
96 * shown.
97 *
98 /* INDENT ON */

```

```
100 #include "libm.h"
```

```
102 extern const double _TBL_log[];
```

```

104 static const double P[] = {
105 /* ONE */ 1.0,
106 /* TWO52 */ 4503599627370496.0,
107 /* LN2HI */ 6.93147180369123816490e-01, /* 3fe62e42, fee00000 */
108 /* LN2LO */ 1.90821492927058770002e-10, /* 3dea39ef, 35793c76 */
109 /* A1 */ -6.88821452420390473170286327331268694251775741577e-0002,
110 /* A2 */ 1.97493380704769294631262255279580131173133850098e+0000,
111 /* A3 */ 2.24963218866067560242072431719861924648284912109e+0000,
112 /* A4 */ -9.02975906958474405783476868236903101205825805664e-0001,
113 /* A5 */ -1.47391630715542865104339398385491222143173217773e+0000,
114 /* A6 */ 1.86846544648220058704168877738993614912033081055e+0000,
115 /* A7 */ 1.82277370459347465292410106485476717352867126465e+0000,
116 /* A8 */ 1.25295479915214102994980294170090928673744201660e+0000,
117 /* A9 */ 1.96709676945198275177517643896862864494323730469e+0000,
118 /* A10 */ -4.00127989749189894030934055990655906498432159424e-0001,
119 /* A11 */ 3.0167552858798333733648178167641162872314453125e+0000,
120 /* A12 */ -9.52325445049240770778453679668018594384193420410e-0001,
121 /* B1 */ -1.25041641589283658575482149899471551179885864258e-0001,
122 /* B2 */ 1.87161713283355151891381127914642725337613123482e+0000,
123 /* B3 */ -1.89082956295731507978530316904652863740921020508e+0000,

```

new/usr/src/lib/libm/common/C/log.c

3

```
124 /* B4      */ -2.50562891673640253387134180229622870683670043945e+0000,  
125 /* B5      */  1.64822828085258366037635369139024987816810607910e+0000,  
126 /* B6      */ -1.24409107065868340669112512841820716857910156250e+0000,  
127 /* B7      */  1.70534231658220414296067701798165217041969299316e+0000,  
128 /* B8      */  1.99196833784655646937267192697618156671524047852e+0000,  
129 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/log10.c

1

```
*****
7235 Tue Nov 25 12:56:53 2014
new/usr/src/lib/libm/common/C/log10.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28 #pragma weak __log10 = log10
29 #pragma weak log10 = __log10
30
31 /* INDENT OFF */
32 /*
33  * log10(x) = log(x)/log10
34  *
35  * Base on Table look-up algorithm with product polynomial
36  * approximation for log(x).
37  *
38  * By K.C. Ng, Nov 29, 2004
39  *
40  * (a). For x in [1-0.125, 1+0.125], from log.c we have
41  *   log(x) = f + ((a1*f^2) *
42  *   ((a2 + (a3*f)*(a4+f)) + (f^3)*(a5+f))) *
43  *   (((a6 + f*(a7+f)) + (f^3)*(a8+f)) *
44  *   ((a9 + (a10*f)*(a11+f)) + (f^3)*(a12+f)))
45  *   where f = x - 1.
46  *   (i) modify a1 <- a1 / log10
47  *   (ii) 1/log10 = 0.4342944819...
48  *   = 0.4375 - 0.003205518... (7 bit shift)
49  *   Let lgv = 0.4375 - 1/log10, then
50  *   lgv = 0.003205518096748172348871081083395...
51  *   (iii) f*0.4375 is exact because f has 3 trailing zero.
52  *   (iv) Thus, log10(x) = f*0.4375 - (lgv*f - Ppoly)
53  *
54  * (b). For 0.09375 <= x < 24
55  *   Let j = (ix - 0x3fb80000) >> 15. Look up Y[j], 1/Y[j], and log(Y[j])
56  *   from _TBL_log.c. Then
57  *   log10(x) = log10(Y[j]) + log10(1 + (x-Y[j])*(1/Y[j]))
```

new/usr/src/lib/libm/common/C/log10.c

2

```
58 *   = log(Y[j])(1/log10) + log10(1 + s)
59 *   where
60 *   s = (x-Y[j])*(1/Y[j])
61 *   From log.c, we have log(1+s) =
62 *   2
63 *   (b s) (b + b s + s ) [b + b s + s (b + s)] (b + b s + s )
64 *   1 2 3 4 5 6 7 8
65 *
66 *   By setting b1 <- b1/log10, we have
67 *   log10(x) = 0.4375 * T - (lgv * T - POLY(s))
68 *
69 * (c). Otherwise, get "n", the exponent of x, and then normalize x to
70 * z in [1,2). Then similar to (b) find a Y[i] that matches z to 5.5
71 * significant bits. Then
72 *   log(x) = n*ln2 + log(Y[i]) + log(z/Y[i]).
73 *   log10(x) = n*(ln2/ln10) + log10(z).
74 *
75 * Special cases:
76 *   log10(x) is NaN with signal if x < 0 (including -INF) ;
77 *   log10(+INF) is +INF; log10(0) is -INF with signal;
78 *   log10(NaN) is that NaN with no signal.
79 *
80 * Maximum error observed: less than 0.89 ulp
81 *
82 * Constants:
83 * The hexadecimal values are the intended ones for the following constants.
84 * The decimal values may be used, provided that the compiler will convert
85 * from decimal to binary accurately enough to produce the hexadecimal values
86 * shown.
87 */
88 /* INDENT ON */
89
90 #include "libm.h"
91
92 extern const double _TBL_log[];
93
94 static const double P[] = {
95 /* ONE */ 1.0,
96 /* TWO52 */ 4503599627370496.0,
97 /* LNAHI */ 3.01029995607677847147e-01, /* 3FD34413 50900000 */
98 /* LNALO */ 5.63033480667509769841e-11, /* 3DCEF3FD E623E256 */
99 /* A1 */ -2.9142521960136582507385480707044582802184e-02,
100 /* A2 */ 1.99628461483039965074226529395673424005508422852e+0000,
101 /* A3 */ 2.26812367662950720159642514772713184356689453125e+0000,
102 /* A4 */ -9.05030639084976384900471657601883634924888610840e-0001,
103 /* A5 */ -1.48275767132434044270894446526654064655303955078e+0000,
104 /* A6 */ 1.88158320939722756293122074566781520843505859375e+0000,
105 /* A7 */ 1.83309386046986411145098827546462416648864746094e+0000,
106 /* A8 */ 1.24847063988317086291601754055591300129890441895e+0000,
107 /* A9 */ 1.98372421445537705508854742220137268304824829102e+0000,
108 /* A10 */ -3.94711735767898475035764249696512706577777862549e-0001,
109 /* A11 */ 3.07890395362954372160402272129431366920471191406e+0000,
110 /* A12 */ -9.60099585275022149311041630426188930869102478027e-0001,
111 /* B1 */ -5.4304894950350052960838096752491540286689e-02,
112 /* B2 */ 1.87161713283355151891381127914642725337613123482e+0000,
113 /* B3 */ -1.89082956295731507978530316904652863740921020508e+0000,
114 /* B4 */ -2.50562891673640253387134180229622870683670043945e+0000,
115 /* B5 */ 1.64822828085258366037635369139024987816810607910e+0000,
116 /* B6 */ -1.24409107065868340669112512841820716857910156250e+0000,
117 /* B7 */ 1.70534231658220414296067701798165217041969299316e+0000,
118 /* B8 */ 1.99196833784655646937267192697618156671524047852e+0000,
119 /* LGH */ 0.4375,
120 /* LGL */ 0.003205518096748172348871081083395,
121 /* LG10V */ 0.43429448190325182765112891891660509576226,
122 };
123
124 #ifndef unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/loglp.c

1

```

*****
6364 Tue Nov 25 12:56:53 2014
new/usr/src/lib/libm/common/C/loglp.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __loglp = loglp
29 #pragma weak loglp = __loglp

31 /* INDEXT OFF */
32 /*
33  * Method :
34  * 1. Argument Reduction: find k and f such that
35  *      1+x = 2^k * (1+f),
36  *      where sqrt(2)/2 < 1+f < sqrt(2) .
37  *
38  * Note. If k=0, then f=x is exact. However, if k != 0, then f
39  * may not be representable exactly. In that case, a correction
40  * term is need. Let u=1+x rounded. Let c = (1+x)-u, then
41  * log(1+x) - log(u) ~ c/u. Thus, we proceed to compute log(u),
42  * and add back the correction term c/u.
43  * (Note: when x > 2**53, one can simply return log(x))
44  *
45  * 2. Approximation of loglp(f).
46  * Let s = f/(2+f) ; based on log(1+f) = log(1+s) - log(1-s)
47  *      = 2s + 2/3 s**3 + 2/5 s**5 + .....
48  *      = 2s + s*R
49  * We use a special Reme algorithm on [0,0.1716] to generate
50  * a polynomial of degree 14 to approximate R The maximum error
51  * of this polynomial approximation is bounded by 2**-58.45. In
52  * other words,
53  *
54  *      2      4      6      8      10      12      14
55  *      R(z) ~ Lp1*s +Lp2*s +Lp3*s +Lp4*s +Lp5*s +Lp6*s +Lp7*s
56  * (the values of Lp1 to Lp7 are listed in the program)
57  * and
58  *      |          2          14          |          -58.45

```

new/usr/src/lib/libm/common/C/loglp.c

2

```

58 *      | Lp1*s +...+Lp7*s - R(z) | <= 2
59 *
60 * Note that 2s = f - s*f = f - hfsq + s*hfsq, where hfsq = f*f/2.
61 * In order to guarantee error in log below lulp, we compute log
62 * by
63 *      loglp(f) = f - (hfsq - s*(hfsq+R)).
64 *
65 * 3. Finally, loglp(x) = k*ln2 + loglp(f).
66 *      = k*ln2_hi+(f-(hfsq-(s*(hfsq+R)+k*ln2_lo)))
67 * Here ln2 is splitted into two floating point number:
68 *      ln2_hi + ln2_lo,
69 * where n*ln2_hi is always exact for |n| < 2000.
70 *
71 * Special cases:
72 * loglp(x) is NaN with signal if x < -1 (including -INF) ;
73 * loglp(+INF) is +INF; loglp(-1) is -INF with signal;
74 * loglp(NaN) is that NaN with no signal.
75 *
76 * Accuracy:
77 * according to an error analysis, the error is always less than
78 * 1 ulp (unit in the last place).
79 *
80 * Constants:
81 * The hexadecimal values are the intended ones for the following
82 * constants. The decimal values may be used, provided that the
83 * compiler will convert from decimal to binary accurately enough
84 * to produce the hexadecimal values shown.
85 *
86 * Note: Assuming log() return accurate answer, the following
87 * algorithm can be used to compute loglp(x) to within a few ULP:
88 *
89 *      u = 1+x;
90 *      if (u == 1.0) return x ; else
91 *          return log(u)*(x/(u-1.0));
92 *
93 * See HP-15C Advanced Functions Handbook, p.193.
94 */
95 /* INDEXT ON */

97 #include "libm.h"

99 static const double xxx[] = {
100 /* ln2_hi */ 6.93147180369123816490e-01, /* 3fe62e42 fee00000 */
101 /* ln2_lo */ 1.90821492927058770002e-10, /* 3dea39ef 35793c76 */
102 /* two54 */ 1.80143985094819840000e+16, /* 43500000 00000000 */
103 /* Lp1 */ 6.66666666666666735130e-01, /* 3FE55555 55555593 */
104 /* Lp2 */ 3.999999999940941908e-01, /* 3FD99999 9997FA04 */
105 /* Lp3 */ 2.857142874366239149e-01, /* 3FD24924 94229359 */
106 /* Lp4 */ 2.222219843214978396e-01, /* 3FCC71C5 1D8E78AF */
107 /* Lp5 */ 1.818357216161805012e-01, /* 3FC74664 96CB03DE */
108 /* Lp6 */ 1.531383769920937332e-01, /* 3FC39A09 D078C69F */
109 /* Lp7 */ 1.479819860511658591e-01, /* 3FC2F112 DF3E5244 */
110 /* zero */ 0.0
111 };
    unchanged_portion_omitted

```

```

*****
7366 Tue Nov 25 12:56:54 2014
new/usr/src/lib/libm/common/C/log2.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __log2 = log2
31 #pragma weak log2 = __log2
32
33 /* INDENT OFF */
34 * log2(x) = log(x)/log2
35 *
36 * Base on Table look-up algorithm with product polynomial
37 * approximation for log(x).
38 *
39 * By K.C. Ng, Nov 29, 2004
40 *
41 * (a). For x in [1-0.125, 1+0.125], from log.c we have
42 *   log(x) = f + ((a1*f^2) *
43 *     ((a2 + (a3*f)*(a4+f)) + (f^3)*(a5+f))) *
44 *     ((a6 + f*(a7+f)) + (f^3)*(a8+f)) *
45 *     ((a9 + (a10*f)*(a11+f)) + (f^3)*(a12+f)))
46 *
47 *   where f = x - 1.
48 *   (i) modify a1 <- a1 / log2
49 *   (ii) 1/log2 = 1.4426950408889634...
50 *         = 1.5 - 0.057304959... (4 bit shift)
51 *
52 *   Let lv = 1.5 - 1/log2, then
53 *   lv = 0.057304959111036592640075318998107956665325,
54 *   (iii) f*1.5 is exact because f has 3 trailing zero.
55 *   (iv) Thus, log2(x) = f*1.5 - (lv*f - PPoly)
56 *
57 * (b). For 0.09375 <= x < 24
58 *   Let j = (ix - 0x3fb80000) >> 15. Look up Y[j], 1/Y[j], and log(Y[j])
59 *   from _TBL_log.c. Then

```

```

58 *   log2(x) = log2(Y[j]) + log2(1 + (x-Y[j])*(1/Y[j]))
59 *           = log(Y[j])(1/log2) + log2(1 + s)
60 *
61 *   where s = (x-Y[j])*(1/Y[j])
62 *   From log.c, we have log(1+s) =
63 *
64 *           2           2
65 *       (b s) (b + b s + s) [b + b s + s (b + s)] (b + b s + s )
66 *         1     2     3         4     5         6         7     8
67 *
68 *   By setting bl <- bl/log2, we have
69 *   log2(x) = 1.5 * T - (lv * T - POLY(s))
70 *
71 * (c). Otherwise, get "n", the exponent of x, and then normalize x to
72 * z in [1,2). Then similar to (b) find a Y[i] that matches z to 5.5
73 * significant bits. Then
74 *   log2(x) = n + log2(z).
75 *
76 * Special cases:
77 *   log2(x) is NaN with signal if x < 0 (including -INF) ;
78 *   log2(+INF) is +INF; log2(0) is -INF with signal;
79 *   log2(NaN) is that NaN with no signal.
80 *
81 * Maximum error observed: less than 0.84 ulp
82 *
83 * Constants:
84 * The hexadecimal values are the intended ones for the following constants.
85 * The decimal values may be used, provided that the compiler will convert
86 * from decimal to binary accurately enough to produce the hexadecimal values
87 * shown.
88 /* INDENT ON */
89
90 #include "libm.h"
91 #include "libm_synonyms.h"
92 #include "libm_protos.h"
93
94 extern const double _TBL_log[];
95
96 static const double P[] = {
97 /* ONE */ 1.0,
98 /* TWO52 */ 4503599627370496.0,
99 /* LN10V */ 1.4426950408889634073599246810018920433347, /* 1/log10 */
100 /* ZERO */ 0.0,
101 /* A1 */ -9.6809362455249638217841932228967194640116e-02,
102 /* A2 */ 1.99628461483039965074226529395673424005508422852e+0000,
103 /* A3 */ 2.26812367662950720159642514772713184356689453125e+0000,
104 /* A4 */ -9.05030639084976384900471657601883634924888610840e-0001,
105 /* A5 */ -1.48275767132434044270894446526654064655303955078e+0000,
106 /* A6 */ 1.88158320939722756293122074566781520843505859375e+0000,
107 /* A7 */ 1.83309386046986411145098827546462416648864746094e+0000,
108 /* A8 */ 1.24847063988317086291601754055591300129890441895e+0000,
109 /* A9 */ 1.98372421445537705508854742220137268304824829102e+0000,
110 /* A10 */ -3.9471173576789847503576424969651270657777862549e-0001,
111 /* A11 */ 3.07890395362954372160402272129431366920471191406e+0000,
112 /* A12 */ -9.60099585275022149311041630426188930869102478027e-0001,
113 /* B1 */ -1.8039695622547469514898963204616532885451e-01,
114 /* B2 */ 1.87161713283355151891381127914642725337613123482e+0000,
115 /* B3 */ -1.89082956295731507978530316904652863740921020508e+0000,
116 /* B4 */ -2.50562891673640253387134180229622870683670043945e+0000,
117 /* B5 */ 1.64822828085258366037635369139024987816810607910e+0000,
118 /* B6 */ -1.24409107065868340669112512841820716857910156250e+0000,
119 /* B7 */ 1.70534231658220414296067701798165217041969299316e+0000,
120 /* B8 */ 1.99196833784655646937267192697618156671524047852e+0000,
121 /* LGH */ 1.5,
122 /* LGL */ 0.057304959111036592640075318998107956665325,
123 };

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/C/logb.c

1

```
*****
2116 Tue Nov 25 12:56:54 2014
new/usr/src/lib/libm/common/C/logb.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __logb = logb
31 #pragma weak _logb = logb
32 #if defined(ELFOSBJ)
31 #pragma weak logb = __logb
32 #pragma weak _logb = __logb
33 #endif
34
35 #include "libm.h"
36 #include "xpg6.h" /* __xpg6 */
37 #define _C99SUSv3_logb _C99SUSv3_logb_subnormal_is_like_ilogb
38
39 #if defined(__x86)
40 #if defined(USE_FPSCALE) || defined(__x86)
41 static const double two52 = 4503599627370496.0;
42 #else
43 /*
44  * v: high part of a non-zero subnormal |x|; w: low part of |x|
45  */
46 static int
47 ilogb_subnormal(unsigned v, unsigned w) {
48     int r = -1022 - 52;
49
50     if (v)
```

new/usr/src/lib/libm/common/C/logb.c

2

```
48         r += 32;
49     else
50         v = w;
51     if (v & 0xffff0000)
52         r += 16, v >>= 16;
53     if (v & 0xff00)
54         r += 8, v >>= 8;
55     if (v & 0xf0)
56         r += 4, v >>= 4;
57     v <<= 1;
58     return (r + ((0xffffaa50 >> v) & 0x3));
59 }
60 #endif /* defined(__x86) */
61 #endif /* defined(USE_FPSCALE) */
62
63 double
64 logb(double x) {
65     int *px = (int *) &x, k = px[HIWORD] & ~0x80000000;
66
67     if (k < 0x00100000) {
68         if ((px[LOWORD] | k) == 0)
69             return (_SVID_libm_err(x, x, 45));
70         else if ((__xpg6 & _C99SUSv3_logb) != 0) {
71             #if defined(__x86)
72             #if defined(USE_FPSCALE) || defined(__x86)
73                 x *= two52;
74                 return ((double) ((px[HIWORD] & 0x7ff00000) >> 20)
75                     - 1075));
76             #else
77                 return ((double) ilogb_subnormal(k, px[LOWORD]));
78             #endif
79         } else if (k < 0x7ff00000)
80             return ((double) ((k >> 20) - 1023));
81     else
82         return (x * x);
83 }
84
85 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/nextafter.c

1

```
*****
2272 Tue Nov 25 12:56:55 2014
new/usr/src/lib/libm/common/C/nextafter.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
29 #pragma weak __nextafter = nextafter
30 #pragma weak _nextafter = nextafter
31 #pragma weak nextafter = __nextafter
32 #pragma weak _nextafter = __nextafter
32 #include "libm.h"
33 #include <float.h>          /* DBL_MIN */
35 double
36 nextafter(double x, double y) {
37     int      hx, hy, k;
38     double   ans;
39     unsigned  lx;
40     volatile double dummy;
42     hx = ((int *)&x)[HIWORD];
43     lx = ((int *)&x)[LOWORD];
44     hy = ((int *)&y)[HIWORD];
45     k = (hx & ~0x80000000) | lx;
47     if (x == y)
48         return (y);          /* C99 requirement */
49     if (x != x || y != y)
50         return (x * y);
51     if (k == 0) {           /* x = 0 */
52         k = hy & 0x80000000;
53         ((int *)&ans)[HIWORD] = k;
54         ((int *)&ans)[LOWORD] = 1;
55     } else if (hx >= 0) {
56         if (x > y) {
```

new/usr/src/lib/libm/common/C/nextafter.c

2

```
57         ((int *)&ans)[LOWORD] = lx - 1;
58         k = (lx == 0)? hx - 1 : hx;
59         ((int *)&ans)[HIWORD] = k;
60     } else {
61         ((int *)&ans)[LOWORD] = lx + 1;
62         k = (lx == 0xfffffff)? hx + 1 : hx;
63         ((int *)&ans)[HIWORD] = k;
64     }
65 } else {
66     if (x < y) {
67         ((int *)&ans)[LOWORD] = lx - 1;
68         k = (lx == 0)? hx - 1 : hx;
69         ((int *)&ans)[HIWORD] = k;
70     } else {
71         ((int *)&ans)[LOWORD] = lx + 1;
72         k = (lx == 0xfffffff)? hx + 1 : hx;
73         ((int *)&ans)[HIWORD] = k;
74     }
75 }
76 k = (k >> 20) & 0x7fff;
77 if (k == 0x7fff) {
78     /* overflow */
79     return (_SVID_libm_err(x, y, 46));
80 #if !defined(__lint)
81     } else if (k == 0) {
82         /* underflow */
83         dummy = DBL_MIN * copysign(DBL_MIN, x);
84 #endif
85     }
86     return (ans);
87 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/pow.c

1

```
*****
10175 Tue Nov 25 12:56:55 2014
new/usr/src/lib/libm/common/C/pow.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __pow = pow
30 #if defined(ELFOBJ)
31 #pragma weak pow = __pow
32 #endif

32 /*
33  * pow(x,y) return x**y
34  *      n
35  * Method: Let x = 2 ** (1+f)
36  *      1. Compute and return log2(x) in two pieces:
37  *          log2(x) = w1 + w2,
38  *          where w1 has 24 bits trailing zero.
39  *      2. Perform y*log2(x) by simulating muti-precision arithmetic
40  *      3. Return x**y = exp2(y*log(x))
41  *
42  * Special cases:
43  *      1. (anything) ** +-0 is 1
44  *      1'. 1 ** (anything) is 1 (C99; 1 ** +-INF/NAN used to be NAN)
45  *      2. (anything) ** 1 is itself
46  *      3. (anything except 1) ** NAN is NAN ("except 1" is C99)
47  *      4. NAN ** (anything except 0) is NAN
48  *      5. +-(|x| > 1) ** +INF is +INF
49  *      6. +-(|x| > 1) ** -INF is +0
```

new/usr/src/lib/libm/common/C/pow.c

2

```
50 *      7. +-(|x| < 1) ** +INF is +0
51 *      8. +-(|x| < 1) ** -INF is +INF
52 *      9. -1 ** +-INF is 1 (C99; -1 ** +-INF used to be NAN)
53 *      10. +0 ** (+anything except 0, NAN) is +0
54 *      11. -0 ** (+anything except 0, NAN, odd integer) is +0
55 *      12. +0 ** (-anything except 0, NAN) is +INF
56 *      13. -0 ** (-anything except 0, NAN, odd integer) is +INF
57 *      14. -0 ** (odd integer) = -(+0 ** (odd integer))
58 *      15. +INF ** (+anything except 0, NAN) is +INF
59 *      16. +INF ** (-anything except 0, NAN) is +0
60 *      17. -INF ** (anything) = -0 ** (-anything)
61 *      18. (-anything) ** (integer) is (-1)**(integer)*(+anything**integer)
62 *      19. (-anything except 0 and inf) ** (non-integer) is NAN
63 *
64 * Accuracy:
65 *      pow(x,y) returns x**y nearly rounded. In particular
66 *      pow(integer, integer)
67 *      always returns the correct integer provided it is representable.
68 */

70 #include "libm.h"
71 #include "xpg6.h" /* __xpg6 */
72 #define _C99SUSv3_pow _C99SUSv3_pow_treats_Inf_as_an_even_int

74 static const double zero = 0.0, one = 1.0, two = 2.0;

76 extern const double _TBL_log2_hi[], _TBL_log2_lo[];
77 static const double
78 two53 = 9007199254740992.0,
79 A1_hi = 2.8853900432586669921875,
80 A1_lo = 3.8519259825035041963606002e-8,
81 A2 = 2.885390081777926817222541963606002026086e+0000,
82 A2 = 9.617966939207270828380543979852286255862e-0001,
83 A3 = 5.770807680887875964868853124873696201995e-0001,
84 B0_hi = 2.8853900432586669921875,
85 B0_lo = 3.8519259822532793056374320585e-8,
86 B0 = 2.885390081777926814720293056374320585689e+0000,
87 B1 = 9.61796693925975138949202350396200257632e-0001,
88 B2 = 5.770780163585687000782112776448797953382e-0001,
89 B3 = 4.121985488948771523290174512461778354953e-0001,
90 B4 = 3.207590534812432970433641789022666850193e-0001;

92 static double
93 log2_x(double x, double *w) {
94     double f, s, z, qn, h, t;
95     int *px = (int *) &x;
96     int *pz = (int *) &z;
97     int i, j, ix, ni;

99     n = 0;
100    ix = px[HIWORD];
101    if (ix >= 0x3fef03f1 && ix < 0x3ff08208) { /* 65/63 > x > 63/65 */
102        double f1, v;
103        f = x - one;
104        if (((ix - 0x3ff00000) | px[LOWORD]) == 0) {
105            *w = zero;
106            return (zero); /* log2(1) = +0 */
107        }
108        qn = one / (two + f);
109        s = f * qn; /* |s| < 2** -6 */
110        v = s * s;
111        h = (double) ((float) s);
112        f1 = (double) ((float) f);
113        t = qn * (((f - two * h) - h * f1) - h * (f - f1));
114        /* s = h+t */
115        f1 = h * B0_lo + s * (v * (B1 + v * (B2 + v * (B3 + v * B4))));
```

```
116         t = f1 + t * B0;
117         h *= B0_hi;
118         s = (double) ((float) (h + t));
119         *w = t - (s - h);
120         return (s);
121     }
122     if (ix < 0x00100000) { /* subnormal x */
123         x *= two53;
124         n = -53;
125         ix = px[HIWORD];
126     }
127     /* LARGE N */
128     n += ((ix + 0x1000) >> 20) - 0x3ff;
129     ix = (ix & 0x000fffff) | 0x3ff00000; /* scale x to [1,2] */
130     px[HIWORD] = ix;
131     i = ix + 0x1000;
132     pz[HIWORD] = i & 0xffffe000;
133     pz[LOWORD] = 0;
134     qn = one / (x + z);
135     f = x - z;
136     s = f * qn;
137     h = (double) ((float) s);
138     t = qn * ((f - (h + h) * z) - h * f);
139     j = (i >> 13) & 0x7f;
140     f = s * s;
141     t = t * A1 + h * A1_lo;
142     t += (s * f) * (A2 + f * A3);
143     qn = h * A1_hi;
144     s = n + _TBL_log2_hi[j];
145     h = qn + s;
146     t += _TBL_log2_lo[j] - ((h - s) - qn);
147     f = (double) ((float) (h + t));
148     *w = t - (f - h);
149     return (f);
150 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/C/remainder.c

1

```
*****
2344 Tue Nov 25 12:56:55 2014
new/usr/src/lib/libm/common/C/remainder.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
29 #pragma weak __remainder = remainder
29 #pragma weak remainder = __remainder
31 /*
32  * remainder(x,p)
33  * Code originated from 4.3bsd.
34  * Modified by K.C. Ng for SUN 4.0 libm.
35  * Return :
36  *     returns x REM p = x - [x/p]*p as if in infinite precise arithmetic,
37  *     where [x/p] is the (infinite bit) integer nearest x/p (in half way
38  *     case choose the even one).
39  * Method :
40  *     Based on fmod() return x-[x/p]chopped*p exactly.
41  */
43 #include "libm.h"
45 static const double zero = 0.0, half = 0.5;
47 double
48 remainder(double x, double p) {
49     double halfp;
50     int ix, hx, hp;
52     ix = ((int *)&x)[HIWORD];
53     hx = ix & ~0x80000000;
54     hp = ((int *)&p)[HIWORD] & ~0x80000000;
56     if (hp > 0x7ff00000 || (hp == 0x7ff00000 && ((int *)&p)[LOWORD] != 0))
57         return (x * p);
```

new/usr/src/lib/libm/common/C/remainder.c

2

```
58     if (hx > 0x7ff00000 || (hx == 0x7ff00000 && ((int *)&x)[LOWORD] != 0))
59         return (x * p);
61     if ((hp | ((int *)&p)[LOWORD]) == 0 || hx == 0x7ff00000)
62         return (_SVID_libm_err(x, p, 28));
64     p = fabs(p);
65     if (hp < 0x7fe00000)
66         x = fmod(x, p + p);
67     x = fabs(x);
68     if (hp < 0x00200000) {
69         if (x + x > p) {
70             if (x == p) /* avoid x-x=-0 in RM mode */
71                 return ((ix < 0)? -zero : zero);
72             x -= p;
73             if (x + x >= p)
74                 x -= p;
75         }
76     } else {
77         halfp = half * p;
78         if (x > halfp) {
79             if (x == p) /* avoid x-x=-0 in RM mode */
80                 return ((ix < 0)? -zero : zero);
81             x -= p;
82             if (x >= halfp)
83                 x -= p;
84         }
85     }
86     return ((ix < 0)? -x : x);
87 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/rint.c

1

```
*****
2042 Tue Nov 25 12:56:56 2014
new/usr/src/lib/libm/common/C/rint.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __rint = rint
30 #pragma weak rint = __rint
31
32 /*
33 * rint(x) return x rounded to integral according to the rounding direction
34 * rint(x) returns result with the same sign as x's, including 0.0.
35 */
36
37 #include "libm.h"
38
39 #if defined(__i386) && !defined(__amd64) && (!defined(__FLT_EVAL_METHOD__) || \
40  __FLT_EVAL_METHOD__ != 0)
41 extern enum fp_precision_type __swapRP(enum fp_precision_type);
42 #define DECLRP(x)          enum fp_precision_type x;
43 #define SWAPRP(new, x)    x = __swapRP(new);
44 #define RESTRP(x)        (void) __swapRP(x);
45 #else
46 #define DECLRP(x)
47 #define SWAPRP(new, x)
48 #define RESTRP(x)
49 #endif
50
51 static const double
52 two52 = 4503599627370496.0,
53 zero = 0.0,
54 one = 1.0;
55
56 double
57 rint(double x) {
58     DECLRP(rp)
```

new/usr/src/lib/libm/common/C/rint.c

2

```
58     double t, w;
59     int ix, hx;
60
61     ix = ((int *)&x)[HIWORD];
62     hx = ix & ~0x80000000;
63
64     if (hx >= 0x43300000)
65         return (x * one);
66     t = (ix < 0)? -two52 : two52;
67     SWAPRP(fp_double, rp) /* set precision mode to double */
68     w = x + t; /* x+sign(x)*2**52 rounded */
69     RESTRP(rp) /* restore precision mode */
70     if (w == t)
71         return ((ix < 0)? -zero : zero);
72     return (w - t);
73 }
74
75 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/scalb.c

1

```
*****
2082 Tue Nov 25 12:56:56 2014
new/usr/src/lib/libm/common/C/scalb.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __scalb = scalb
30 #pragma weak _scalb = scalb
31 #pragma weak scalb = __scalb
32 #pragma weak _scalb = __scalb

32 #include "libm.h"

34 double
35 scalb(double x, double fn) {
36     int    hn, in, n;
37     double z;

39     if (isnan(x) || isnan(fn))
40         return (x * fn);

42     in = ((int *)&fn)[HIWORD];
43     hn = in & ~0x80000000;
44     if (hn == 0x7ff00000) /* fn is inf */
45         return (_SVID_libm_err(x, fn, 47));

47     /* see if fn is an integer without raising inexact */
48     if (hn >= 0x43300000) {
49         /* |fn| >= 2^52, so it must be an integer */
50         n = (in < 0)? -65000 : 65000;
51     } else if (hn < 0x3ff00000) {
52         /* |fn| < 1, so it must be zero or non-integer */
53         return ((fn == 0.0)? x : (x - x) / (x - x));
54     } else if (hn < 0x41400000) {
55         /* |fn| < 2^21 */
56         if ((hn & ((1 << (0x413 - (hn >> 20))) - 1))
```

new/usr/src/lib/libm/common/C/scalb.c

2

```
57         | ((int *)&fn)[LOWORD])
58             return ((x - x) / (x - x));
59     n = (int)fn;
60 } else {
61     if (((int *)&fn)[LOWORD] & ((1 << (0x433 - (hn >> 20))) - 1))
62         return ((x - x) / (x - x));
63     n = (in < 0)? -65000 : 65000;
64 }
65 z = scalbn(x, n);
66 if (z != x) {
67     if (z == 0.0)
68         return (_SVID_libm_err(x, fn, 33));
69     if (!finite(z))
70         return (_SVID_libm_err(x, fn, 32));
71 }
72 return (z);
73 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/C/scalbn.c

1

```
*****
2763 Tue Nov 25 12:56:57 2014
new/usr/src/lib/libm/common/C/scalbn.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */
29 #pragma weak __scalbn = scalbn
29 #pragma weak scalbn = __scalbn
31 #include "libm.h"
33 static const double
34     one = 1.0,
35     huge = 1.0e300,
36     tiny = 1.0e-300,
37     twom54 = 5.5511151231257827021181583404541015625e-17;
39 #if defined(__x86)
39 #if defined(USE_FPSCALE) || defined(__x86)
40 static const double two52 = 4503599627370496.0;
41 #else
42 /*
43  * Normalize non-zero subnormal x and return biased exponent of x in [-51,0]
44  */
45 static int
46 ilogb_biased(unsigned *px) {
47     int s = 52;
48     unsigned v = px[HIWORD] & ~0x80000000, w = px[LOWORD], t = v;
50     if (t)
```

new/usr/src/lib/libm/common/C/scalbn.c

2

```
51     s -= 32;
52     else
53         t = w;
54     if (t & 0xffff0000)
55         s -= 16, t >>= 16;
56     if (t & 0xff00)
57         s -= 8, t >>= 8;
58     if (t & 0xf0)
59         s -= 4, t >>= 4;
60     t <<= 1;
61     s -= (0xfffffaa50 >> t) & 0x3;
62     if (s < 32) {
63         v = (v << s) | w >> (32 - s);
64         w <<= s;
65     } else {
66         v = w << (s - 32);
67         w = 0;
68     }
69     px[HIWORD] = (px[HIWORD] & 0x80000000) | v;
70     px[LOWORD] = w;
71     return (1 - s);
72 }
73 #endif /* defined(__x86) */
73 #endif /* defined(USE_FPSCALE) */
75 double
76 scalbn(double x, int n) {
77     int *px, ix, hx, k;
79     px = (int *)&x;
80     ix = px[HIWORD];
81     hx = ix & ~0x80000000;
82     k = hx >> 20;
84     if (k == 0x7ff) /* x is inf or NaN */
85         return (x * one);
87     if (k == 0) {
88         if ((hx | px[LOWORD]) == 0 || n == 0)
89             return (x);
90 #if defined(__x86)
90 #if defined(USE_FPSCALE) || defined(__x86)
91         x *= two52;
92         ix = px[HIWORD];
93         k = ((ix & ~0x80000000) >> 20) - 52;
94 #else
95         k = ilogb_biased((unsigned *)px);
96         ix = px[HIWORD];
97 #endif
98         /* now k is in the range -51..0 */
99         k += n;
100        if (k > n) /* integer overflow occurred */
101            k = -100;
102    } else {
103        /* k is in the range 1..1023 */
104        k += n;
105        if (k < n) /* integer overflow occurred */
106            k = 0x7ff;
107    }
109    if (k > 0x7fe)
110        return (huge * ((ix < 0)? -huge : huge));
111    if (k < 1) {
112        if (k <= -54)
113            return (tiny * ((ix < 0)? -tiny : tiny));
114        k += 54;
```


new/usr/src/lib/libm/common/C/scalbn.c

3

```
115         px[HIWORD] = (ix & ~0x7ff00000) | (k << 20);
116         return (x * twom54);
117     }
118     px[HIWORD] = (ix & ~0x7ff00000) | (k << 20);
119     return (x);
120 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/signgam.c

1

1067 Tue Nov 25 12:56:57 2014

new/usr/src/lib/libm/common/C/signgam.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __signgam = signgam
30 #pragma weak signgam = __signgam
```

```
32 #include "libm_synonyms.h"
32 #include <math.h>
```

```
34 int signgam = 0;
```

new/usr/src/lib/libm/common/C/significand.c

1

1455 Tue Nov 25 12:56:57 2014

new/usr/src/lib/libm/common/C/significand.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __significand = significand
31 #if defined(ELFOSBJ)
32 #pragma weak significand = __significand
33 #endif
34
35 #include "libm.h"
36
37 double
38 significand(double x) {
39     int ix = ((int *) &x)[HIWORD] & ~0x80000000;
40
41     /* weed out 0/+-Inf/NaN because C99 ilogb raises invalid on them */
42     if ((ix | ((int *) &x)[LOWORD]) == 0 || ix >= 0x7ff00000)
43         return ((ix & 0x800000) != 0 ? x : x + x);
44     /* assumes sparc-like QNaN */
45     return (x + x);
46 }
47
48 else
49     return (scalbn(x, -ilogb(x)));
50 }
```

unchanged_portion_omitted

```

*****
5187 Tue Nov 25 12:56:58 2014
new/usr/src/lib/libm/common/C/sin.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

28 #pragma weak __sin = sin
29 #pragma weak sin = __sin

31 /* INDEXT OFF */
32 /*
33  * sin(x)
34  * Accurate Table look-up algorithm by K.C. Ng, May, 1995.
35  *
36  * Algorithm: see sincos.c
37 */

39 #include "libm.h"

41 static const double sc[] = {
42 /* ONE = */ 1.0,
43 /* NONE = */ -1.0,
44 /*
45  * |sin(x) - (x+pp1*x^3+pp2*x^5)| <= 2^-58.79 for |x| < 0.008
46  */
47 /* PP1 = */ -0.16666666666666666316558867252052378889521480627858683055567,
48 /* PP2 = */ .0083333315652997472323564894248466758248475374977974017927,
49 /*
50  * |(sin(x) - (x+p1*x^3+...+p4*x^9)|
51  * |-----| <= 2^-57.63 for |x| < 0.1953125
52  *      x
53  */
54 /* P1 = */ -1.66666666666666629669805215138920301589656e-0001,
55 /* P2 = */ 8.333333332390951295683993455280336376663e-0003,
56 /* P3 = */ -1.984126237997976692791551778230098403960e-0004,
57 /* P4 = */ 2.753403624854277237649987622848330351110e-0006,

```

```

58 /*
59  * |cos(x) - (1+qq1*x^2+qq2*x^4)| <= 2^-55.99 for |x| <= 0.008 (0x3f80624d)
60 */
61 /* QQ1 = */ -0.4999999999975492381842911981948418542742729,
62 /* QQ2 = */ 0.041666542904352059294545209158357640398771740,
63 /* PI_H = */ 3.1415926535897931159979634685,
64 /* PI_L = */ 1.22464679914735317722606593227425e-16,
65 /* PI_L0 = */ 1.22464679914558443311283879205095e-16,
66 /* PI_L1 = */ 1.768744113227140223300005233735517376e-28,
67 /* PI2_H = */ 6.2831853071795862319959269370,
68 /* PI2_L = */ 2.44929359829470635445213186454850e-16,
69 /* PI2_L0 = */ 2.44929359829116886622567758410190e-16,
70 /* PI2_L1 = */ 3.53748822645428044660010467471034752e-28,
71 };
_____unchanged_portion_omitted_____

```

```

*****
10669 Tue Nov 25 12:56:58 2014
new/usr/src/lib/libm/common/C/sincos.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29 #pragma weak __sincos = sincos
29 #pragma weak sincos = __sincos

31 /* INDENT OFF */
32 /*
33  * sincos(x,s,c)
34  * Accurate Table look-up algorithm by K.C. Ng, 2000.
35  *
36  * 1. Reduce x to x>0 by cos(-x)=cos(x), sin(-x)=-sin(x).
37  * 2. For 0<= x < 8, let i = (64*x chopped)-10. Let d = x - a[i], where
38  * a[i] is a double that is close to (i+10.5)/64 (and hence |d|< 10.5/64)
39  * and such that sin(a[i]) and cos(a[i]) is close to a double (with error
40  * less than 2** -8 ulp). Then
41  *
42  * cos(x) = cos(a[i]+d) = cos(a[i])cos(d) - sin(a[i])*sin(d)
43  *         = TBL_cos_a[i]*(1+QQ1*d^2+QQ2*d^4) -
44  *           TBL_sin_a[i]*(d+PP1*d^3+PP2*d^5)
45  *         = TBL_cos_a[i] + (TBL_cos_a[i]*d^2*(QQ1+QQ2*d^2) -
46  *           TBL_sin_a[i]*(d+PP1*d^3+PP2*d^5))
47  *
48  * sin(x) = sin(a[i]+d) = sin(a[i])cos(d) + cos(a[i])*sin(d)
49  *         = TBL_sin_a[i]*(1+QQ1*d^2+QQ2*d^4) +
50  *           TBL_cos_a[i]*(d+PP1*d^3+PP2*d^5)
51  *         = TBL_sin_a[i] + (TBL_sin_a[i]*d^2*(QQ1+QQ2*d^2) +
52  *           TBL_cos_a[i]*(d+PP1*d^3+PP2*d^5))
53  *
54  * Note: for x close to n*pi/2, special treatment is need for either
55  * sin or cos:
56  * i in [81, 100] ( pi/2 +-10.5/64 => tiny cos(x) = sin(pi/2-x)
57  * i in [181,200] ( pi +-10.5/64 => tiny sin(x) = sin(pi-x)

```

```

58 * i in [282,301] ( 3pi/2+-10.5/64 => tiny cos(x) = sin(x-3pi/2)
59 * i in [382,401] ( 2pi +-10.5/64 => tiny sin(x) = sin(x-2pi)
60 * i in [483,502] ( 5pi/2+-10.5/64 => tiny cos(x) = sin(5pi/2-x)
61 *
62 * 3. For x >= 8.0, use kernel function __rem_pio2 to perform argument
63 * reduction and call __k_sincos_ to compute sin and cos.
64 *
65 * kernel function:
66 * __rem_pio2 ... argument reduction routine
67 * __k_sincos_ ... sine and cosine function on [-pi/4,pi/4]
68 *
69 * Method.
70 * Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
71 * 1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
72 * [-pi/2, +pi/2], and let n = k mod 4.
73 * 2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
74 *
75 * n sin(x) cos(x) tan(x)
76 * -----
77 * 0 S C S/C
78 * 1 C -S -C/S
79 * 2 -S -C S/C
80 * 3 -C S -C/S
81 * -----
82 *
83 * Special cases:
84 * Let trig be any of sin, cos, or tan.
85 * trig(+/-INF) is NaN, with signals;
86 * trig(NaN) is that NaN;
87 *
88 * Accuracy:
89 * TRIG(x) returns trig(x) nearly rounded (less than 1 ulp)
90 */

92 #include "libm.h"

94 static const double sc[] = {
95 /* ONE = */ 1.0,
96 /* NONE = */ -1.0,
97 /*
98  * |sin(x) - (x+pp1*x^3+pp2*x^5)| <= 2^-58.79 for |x| < 0.008
99  */
100 /* PP1 = */ -0.1666666666666316558867252052378889521480627858683055567,
101 /* PP2 = */ .0083333315652997472323564894248466758248475374977974017927,
102 /*
103  * |(sin(x) - (x+p1*x^3+...+p4*x^9)|
104  * |-----| <= 2^-57.63 for |x| < 0.1953125
105  * x
106  */
107 /* P1 = */ -1.666666666666629669805215138920301589656e-0001,
108 /* P2 = */ 8.333333332390951295683993455280336376663e-0003,
109 /* P3 = */ -1.984126237997976692791551778230098403960e-0004,
110 /* P4 = */ 2.753403624854277237649987622848330351110e-0006,
111 /*
112  * |cos(x) - (1+qq1*x^2+qq2*x^4)| <= 2^-55.99 for |x| <= 0.008 (0x3f80624d)
113  */
114 /* QQ1 = */ -0.4999999999975492381842911981948418542742729,
115 /* QQ2 = */ 0.041666542904352059294545209158357640398771740,
116 /* Q1 = */ -0.5,
117 /* Q2 = */ 4.16666666500350703680945520860748617445e-0002,
118 /* Q3 = */ -1.388888596436972210694266290577848696006e-0003,
119 /* Q4 = */ 2.478563078858589473679519517892953492192e-0005,
120 /* PIO2_H = */ 1.570796326794896557999,
121 /* PIO2_L = */ 6.123233995736765886130e-17,
122 /* PIO2_L0 = */ 6.123233995727922165564e-17,
123 /* PIO2_L1 = */ 8.843720566135701120255e-29,

```

```
124 /* PI_H      = */ 3.1415926535897931159979634685,  
125 /* PI_L      = */ 1.22464679914735317722606593227425e-16,  
126 /* PI_L0     = */ 1.22464679914558443311283879205095e-16,  
127 /* PI_L1     = */ 1.768744113227140223300005233735517376e-28,  
128 /* PI302_H   = */ 4.712388980384689673997,  
129 /* PI302_L   = */ 1.836970198721029765839e-16,  
130 /* PI302_L0  = */ 1.836970198720396133587e-16,  
131 /* PI302_L1  = */ 6.336322524749201142226e-29,  
132 /* PI2_H     = */ 6.2831853071795862319959269370,  
133 /* PI2_L     = */ 2.44929359829470635445213186454850e-16,  
134 /* PI2_L0    = */ 2.44929359829116886622567758410190e-16,  
135 /* PI2_L1    = */ 3.537488226454280446600010467471034752e-28,  
136 /* PI502_H   = */ 7.853981633974482789995,  
137 /* PI502_L   = */ 3.061616997868382943065e-16,  
138 /* PI502_L0  = */ 3.061616997861941598865e-16,  
139 /* PI502_L1  = */ 6.441344200433640781982e-28,  
140 };
```

unchanged portion omitted

```

*****
5486 Tue Nov 25 12:56:59 2014
new/usr/src/lib/libm/common/C/sincospi.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak sincospi = __sincospi

30 /* INDENT OFF */
31 /*
32  * void sincospi(double x, double *s, double *c)
33  * *s = sin(pi*x); *c = cos(pi*x);
34  *
35  * Algorithm, 10/17/2002, K.C. Ng
36  * -----
37  * Let y = |4x|, z = floor(y), and n = (int)(z mod 8.0) (displayed in binary).
38  * 1. If y == z, then x is a multiple of pi/4. Return the following values:
39  *
40  *      n x mod 2   sin(x*pi)   cos(x*pi)   tan(x*pi)
41  * -----
42  *      000 0.00     +0 _____ +1 _____ +0
43  *      001 0.25     +\0.5         +\0.5         +1
44  *      010 0.50     +1 _____ +0 _____ +inf
45  *      011 0.75     +\0.5         -\0.5         -1
46  *      100 1.00     -0 _____ -1 _____ +0
47  *      101 1.25     -\0.5         -\0.5         +1
48  *      110 1.50     -1 _____ -0 _____ +inf
49  *      111 1.75     -\0.5         +\0.5         -1
50  * -----
51  * 2. Otherwise,
52  * -----
53  *      n      t      sin(x*pi)   cos(x*pi)   tan(x*pi)
54  * -----
55  *      000 (y-z)/4   sinpi(t)    cospi(t)    tanpi(t)
56  *      001 (z+1-y)/4 cospi(t)    sinpi(t)    1/tanpi(t)

```

```

57 *      010 (y-z)/4   cospi(t)    -sinpi(t)   -1/tanpi(t)
58 *      011 (z+1-y)/4 sinpi(t)    -cospi(t)   -tanpi(t)
59 *      100 (y-z)/4   -sinpi(t)   -cospi(t)   tanpi(t)
60 *      101 (z+1-y)/4 -cospi(t)   -sinpi(t)   1/tanpi(t)
61 *      110 (y-z)/4   -cospi(t)   sinpi(t)    -1/tanpi(t)
62 *      111 (z+1-y)/4 -sinpi(t)   cospi(t)    -tanpi(t)
63 *
64 * -----
65 * NOTE. This program compute sinpi/cospi(t<0.25) by __k_sin/cos(pi*t, 0.0).
66 * This will return a result with error slightly more than one ulp (but less
67 * than 2 ulp). If one wants accurate result, one may break up pi*t in
68 * high (tpi_h) and low (tpi_l) parts and call __k_sin/cos(tpi_h, tpi_lo)
69 * instead.
70 */

72 #include "libm.h"
75 #include "libm_synonyms.h"
73 #include "libm_protos.h"
74 #include "libm_macros.h"
75 #include <math.h>
76 #if defined(__SUNPRO_C)
77 #include <sunmath.h>
78 #endif

80 static const double
81     pi = 3.14159265358979323846, /* 400921FB,54442D18 */
82     sqsrth_h = 0.70710678118654757273731092936941422522068023681640625,
83     sqsrth_l = -4.8336466567264565185935844299127932213411660131004e-17;
84 /* INDENT ON */

86 void
87 sincospi(double x, double *s, double *c) {
88     double y, z, t;
89     int n, ix, k;
90     int hx = ((int *) &x)[HIWORD];
91     unsigned h, lx = ((unsigned *) &x)[LOWORD];

93     ix = hx & ~0x80000000;
94     n = (ix >> 20) - 0x3ff;
95     if (n >= 51) { /* |x| >= 2**51 */
96         if (n >= 1024)
97             #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
98                 *s = *c = ix >= 0x7ff80000 ? x : x - x;
99                 /* assumes sparc-like QNaN */
100 #else
101                 *s = *c = x - x;
102 #endif
103     } else {
104         if (n >= 53) {
105             *s = 0.0;
106             *c = 1.0;
107         }
108         else if (n == 52) {
109             if ((lx & 1) == 0) {
110                 *s = 0.0;
111                 *c = 1.0;
112             }
113             else {
114                 *s = -0.0;
115                 *c = -1.0;
116             }
117         }
118         else { /* n == 51 */
119             if ((lx & 1) == 0) {
120                 *s = 0.0;
121                 *c = 1.0;

```

```

122     }
123     else {
124         *s = 1.0;
125         *c = 0.0;
126     }
127     if ((lx & 2) != 0) {
128         *s = -*s;
129         *c = -*c;
130     }
131 }
132 }
133 }
134 else if (n < -2) /* |x| < 0.25 */
135 *s = __k_sincos(pi * fabs(x), 0.0, c);
136 else {
137     /* y = |4x|, z = floor(y), and n = (int)(z mod 8.0) */
138     if (ix < 0x41C00000) { /* |x| < 2**29 */
139         y = 4.0 * fabs(x);
140         n = (int) y; /* exact */
141         z = (double) n;
142         k = z == y;
143         t = (y - z) * 0.25;
144     }
145     else { /* 2**29 <= |x| < 2**51 */
146         y = fabs(x);
147         k = 50 - n;
148         n = lx >> k;
149         h = n << k;
150         ((unsigned *) &z)[LOWORD] = h;
151         ((int *) &z)[HIWORD] = ix;
152         k = h == lx;
153         t = y - z;
154     }
155     if (k) { /* x = N/4 */
156         if ((n & 1) != 0)
157             *s = *c = sqrth_h + sqrth_l;
158         else
159             if ((n & 2) == 0) {
160                 *s = 0.0;
161                 *c = 1.0;
162             }
163             else {
164                 *s = 1.0;
165                 *c = 0.0;
166             }
167         y = (n & 2) == 0 ? 0.0 : 1.0;
168         if ((n & 4) != 0)
169             *s = -*s;
170         if (((n + 1) & 4) != 0)
171             *c = -*c;
172     }
173     else {
174         if ((n & 1) != 0)
175             t = 0.25 - t;
176         if (((n + (n & 1)) & 2) == 0)
177             *s = __k_sincos(pi * t, 0.0, c);
178         else
179             *c = __k_sincos(pi * t, 0.0, s);
180         if ((n & 4) != 0)
181             *s = -*s;
182         if (((n + 2) & 4) != 0)
183             *c = -*c;
184     }
185 }
186 if (hx < 0)
187     *s = -*s;

```

```

188 }
    unchanged_portion_omitted

```


new/usr/src/lib/libm/common/C/sinh.c

1

```
*****
2079 Tue Nov 25 12:56:59 2014
new/usr/src/lib/libm/common/C/sinh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
29 #pragma weak __sinh = sinh
29 #pragma weak sinh = __sinh
31 /* INDENT OFF */
32 /*
33  * sinh(x)
34  * Code originated from 4.3bsd.
35  * Modified by K.C. Ng for SUN 4.0 libm.
36  * Method :
37  *     1. reduce x to non-negative by sinh(-x) = - sinh(x).
38  *     2.
39  *
40  *           expml(x) + expml(x)/(expml(x)+1)
41  * 0 <= x <= lnovft : sinh(x) := -----
42  *
43  * lnovft <= x < INF : sinh(x) := exp(x-1024*ln2)*2**1023
44  *
45  *
46  * Special cases:
47  *     sinh(x) is x if x is +INF, -INF, or NaN.
48  *     only sinh(0)=0 is exact for finite argument.
49  *
50  */
51 /* INDENT ON */
53 #include "libm.h"
55 static const double
56     ln2hi = 6.93147180369123816490e-01,
57     ln2lo = 1.90821492927058770002e-10,
```

new/usr/src/lib/libm/common/C/sinh.c

2

```
58     lnovft = 7.09782712893383973096e+02;
60 double
61 sinh(double x) {
62     double ox, r, t;
64     ox = x;
65     r = fabs(x);
66     if (!finite(x))
67         return (x * r);
68     if (r < lnovft) {
69         t = expml(r);
70         r = copysign((t + t / (1.0 + t)) * 0.5, x);
71     } else {
72         if (r < 1000.0)
73             x = copysign(exp((r - 1024 * ln2hi) - 1024 * ln2lo), x);
74         r = scalbn(x, 1023);
75     }
76     if (!finite(r))
77         r = _SVID_libm_err(ox, ox, 25);
78     return (r);
79 }
-----
unchanged portion omitted
```

new/usr/src/lib/libm/common/C/sqrt.c

1

```
*****
1220 Tue Nov 25 12:57:00 2014
new/usr/src/lib/libm/common/C/sqrt.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */
29 #pragma weak __sqrt = sqrt
29 #pragma weak sqrt = __sqrt
31 #include "libm.h"
33 #ifdef __INLINE
34 extern double __inline_sqrt(double);
36 double
37 sqrt(double x) {
38     double z = __inline_sqrt(x);
40     if (isnan(x))
41         return (z);
42     return ((x < 0.0)? _SVID_libm_err(x, x, 26) : z);
43 }
46 #else /* defined(__INLINE) */
48 /*
49  * Warning: This correctly rounded sqrt is extremely slow because it computes
50  * the sqrt bit by bit using integer arithmetic.
51  */
```

new/usr/src/lib/libm/common/C/sqrt.c

2

```
53 static const double big = 1.0e30, small = 1.0e-30;
55 double
56 sqrt(double x)
57 {
58     double z;
59     unsigned r, t1, s1, ix1, q1;
60     int ix0, s0, j, q, m, n, t;
61     int *px = (int *)&x, *pz = (int *)&z;
63     ix0 = px[HIWORD];
64     ix1 = px[LOWORD];
65     if ((ix0 & 0x7ff00000) == 0x7ff00000) { /* x is inf or NaN */
66         if (ix0 == 0xffff0000 && ix1 == 0)
67             return (_SVID_libm_err(x, x, 26));
68         return (x + x);
69     }
70     if (((ix0 & 0x7fffffff) | ix1) == 0) /* x is zero */
71         return (x);
73     /* extract exponent and significand */
74     m = ilogb(x);
75     z = scalbn(x, -m);
76     ix0 = (pz[HIWORD] & 0x000ffff) | 0x00100000;
77     ix1 = pz[LOWORD];
78     n = m >> 1;
79     if (n + n != m) {
80         ix0 = (ix0 << 1) | (ix1 >> 31);
81         ix1 <<= 1;
82         m -= 1;
83     }
85     /* generate sqrt(x) bit by bit */
86     ix0 = (ix0 << 1) | (ix1 >> 31);
87     ix1 <<= 1;
88     q = q1 = s0 = s1 = 0;
89     r = 0x00200000;
91     for (j = 1; j <= 22; j++) {
92         t = s0 + r;
93         if (t <= ix0) {
94             s0 = t + r;
95             ix0 -= t;
96             q += r;
97         }
98         ix0 = (ix0 << 1) | (ix1 >> 31);
99         ix1 <<= 1;
100        r >>= 1;
101    }
103    r = 0x80000000;
104    for (j = 1; j <= 32; j++) {
105        t1 = s1 + r;
106        t = s0;
107        if (t < ix0 || (t == ix0 && t1 <= ix1)) {
108            s1 = t1 + r;
109            if ((t1 & 0x80000000) == 0x80000000 &&
110                (s1 & 0x80000000) == 0)
111                s0 += 1;
112            ix0 -= t;
113            if (ix1 < t1)
114                ix0 -= 1;
115            ix1 -= t1;
116            q1 += r;
117        }
118    }
```

```
118         ix0 = (ix0 << 1) | (ix1 >> 31);
119         ix1 <<= 1;
120         r >>= 1;
121     }
122
123     /* round */
124     if ((ix0 | ix1) == 0)
125         goto done;
126     z = big - small;      /* trigger inexact flag */
127     if (z < big)
128         goto done;
129     if (q1 == 0xffffffff) {
130         q1 = 0;
131         q += 1;
132         goto done;
133     }
134     z = big + small;
135     if (z > big) {
136         if (q1 == 0xffffffe)
137             q += 1;
138         q1 += 2;
139         goto done;
140     }
141     q1 += (q1 & 1);
142 done:
143     pz[HIWORD] = (q >> 1) + 0x3fe00000;
144     pz[LOWORD] = q1 >> 1;
145     if ((q & 1) == 1)
146         pz[LOWORD] |= 0x80000000;
147     return (scalbn(z, n));
148 }
149
150 #endif /* defined(__INLINE) */
```

new/usr/src/lib/libm/common/C/tan.c

1

```
*****
1852 Tue Nov 25 12:57:00 2014
new/usr/src/lib/libm/common/C/tan.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __tan = tan
31 #pragma weak tan = __tan
32
33 /* INDEXT OFF */
34 /*
35  * tan(x)
36  * Table look-up algorithm by K.C. Ng, November, 1989.
37  *
38  * kernel function:
39  *   __k_tan      ... tangent function on [-pi/4,pi/4]
40  *   __rem_pio2  ... argument reduction routine
41  */
42 /* INDEXT ON */
43
44 #include "libm.h"
45 #include "libm_synonyms.h"
46 #include "libm_protos.h"
47 #include <math.h>
48
49 double
50 tan(double x) {
51     double y[2], z = 0.0;
52     int n, ix;
53
54     /* high word of x */
55     ix = ((int *) &x)[HIWORD];
56
57     /* |x| ~< pi/4 */
58     ix &= 0x7fffffff;
```

new/usr/src/lib/libm/common/C/tan.c

2

```
57     if (ix <= 0x3fe921fb)
58         return (__k_tan(x, z, 0));
59
60     /* tan(Inf or NaN) is NaN */
61     else if (ix >= 0x7ff00000) {
62 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
63         return (ix >= 0x7ff80000 ? x : x - x); /* NaN */
64         /* assumes sparc-like QNaN */
65 #else
66         return (x - x); /* NaN */
67 #endif
68     }
69
70     /* argument reduction needed */
71     else {
72         n = __rem_pio2(x, y);
73         return (__k_tan(y[0], y[1], n & 1));
74     }
75 }
76
77 _____
78 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/C/tanh.c

1

```
*****
2494 Tue Nov 25 12:57:00 2014
new/usr/src/lib/libm/common/C/tanh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __tanh = tanh
30 #pragma weak tanh = __tanh
31
32 /* INDEXT OFF */
33 /*
34  * TANH(X)
35  * RETURN THE HYPERBOLIC TANGENT OF X
36  * code based on 4.3bsd
37  * Modified by K.C. Ng for sun 4.0, Jan 31, 1987
38  *
39  * Method :
40  * 1. reduce x to non-negative by tanh(-x) = - tanh(x).
41  * 2.
42  * 0 < x <= 1.e-10 : tanh(x) := x
43  * -expml(-2x)
44  * 1.e-10 < x <= 1 : tanh(x) := -----
45  * expml(-2x) + 2
46  * 2
47  * 1 <= x <= 22.0 : tanh(x) := 1 - -----
48  * expml(2x) + 2
49  * 22.0 < x <= INF : tanh(x) := 1.
50  *
51  * Note: 22 was chosen so that fl(1.0+2/(expml(2*22)+2)) == 1.
52  *
53  * Special cases:
54  * tanh(NaN) is NaN;
55  * only tanh(0)=0 is exact for finite argument.
56 */
```

new/usr/src/lib/libm/common/C/tanh.c

2

```
58 #include "libm.h"
59 #include "libm_synonyms.h"
59 #include "libm_protos.h"
60 #include <math.h>
61
62 static const double
63     one = 1.0,
64     two = 2.0,
65     small = 1.0e-10,
66     big = 1.0e10;
67 /* INDEXT ON */
68
69 double
70 tanh(double x) {
71     double t, y, z;
72     int signx;
73     volatile double dummy;
74
75     if (isnan(x))
76         return (x * x); /* + -> * for Cheetah */
77     signx = signbit(x);
78     t = fabs(x);
79     z = one;
80     if (t <= 22.0) {
81         if (t > one)
82             z = one - two / (expml(t + t) + two);
83         else if (t > small) {
84             y = expml(-t - t);
85             z = -y / (y + two);
86         } else {
87             /* raise the INEXACT flag for non-zero t */
88             dummy = t + big;
89 #ifdef lint
90                 dummy = dummy;
91 #endif
92             return (x);
93         }
94     } else if (!finite(t))
95         return (copysign(1.0, x));
96     else
97         return (signx == 1 ? -z + small * small : z - small * small);
98
99     return (signx == 1 ? -z : z);
100 }
_____unchanged_portion_omitted_____
```

```

*****
14533 Tue Nov 25 12:57:01 2014
new/usr/src/lib/libm/common/LD/__lgamma.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 /*
31  * long double __k_lgamma(long double x, int *signgamp);
32  * K.C. Ng, August, 1989.
33  *
34  * We choose [1.5,2.5] to be the primary interval. Our algorithms
35  * are mainly derived from
36  *
37  *
38  *
39  *  $\lgamma(2+s) = s*(1-euler) + \frac{\zeta(2)-1}{2} * s^2 - \frac{\zeta(3)-1}{3} * s^3 + \dots$ 
40  *
41  *
42  *
43  * Note 1. Since  $\gamma(1+s)=s*\gamma(s)$ , hence
44  *  $\lgamma(1+s) = \log(s) + \lgamma(s)$ , or
45  *  $\lgamma(s) = \lgamma(1+s) - \log(s)$ .
46  * When s is really tiny (like roundoff),  $\lgamma(1+s) \sim s(1-enler)$ 
47  * Hence  $\lgamma(s) \sim -\log(s)$  for tiny s
48  *
49 */

51 #include "libm.h"
52 #include "libm_synonyms.h"
52 #include "longdouble.h"

54 static long double neg(long double, int *);
55 static long double poly(long double, const long double *, int);
56 static long double polytail(long double);
57 static long double primary(long double);

```

```

59 static const long double
60 c0 = 0.0L,
61 ch = 0.5L,
62 c1 = 1.0L,
63 c2 = 2.0L,
64 c3 = 3.0L,
65 c4 = 4.0L,
66 c5 = 5.0L,
67 c6 = 6.0L,
68 pi = 3.1415926535897932384626433832795028841971L,
69 tiny = 1.0e-40L;

71 long double
72 __k_lgamma(long double x, int *signgamp) {
73     long double t, y;
74     int i;

76     /* purge off +-inf, NaN and negative arguments */
77     if (!finitel(x))
78         return (x*x);
79     *signgamp = 1;
80     if (signbitl(x))
81         return (neg(x, signgamp));

83     /* for x < 8.0 */
84     if (x < 8.0L) {
85         y = anintl(x);
86         i = (int) y;
87         switch (i) {
88             case 0:
89                 if (x < 1.0e-40L)
90                     return (-logl(x));
91                 else
92                     return (primary(x)-loglpl(x))-logl(x);
93             case 1:
94                 return (primary(x-y)-logl(x));
95             case 2:
96                 return (primary(x-y));
97             case 3:
98                 return (primary(x-y)+logl(x-c1));
99             case 4:
100                return (primary(x-y)+logl((x-c1)*(x-c2)));
101             case 5:
102                return (primary(x-y)+logl((x-c1)*(x-c2)*(x-c3)));
103             case 6:
104                return (primary(x-y)+logl((x-c1)*(x-c2)*(x-c3)*(x-c4)));
105             case 7:
106                return (primary(x-y)+logl((x-c1)*(x-c2)*(x-c3)*(x-c4)*(x-c5)));
107             case 8:
108                return primary(x-y)+
109                    logl((x-c1)*(x-c2)*(x-c3)*(x-c4)*(x-c5)*(x-c6));
110         }
111     }

113     /* 8.0 <= x < 1.0e40 */
114     if (x < 1.0e40L) {
115         t = logl(x);
116         return (x*(t-c1)-(ch*t-polytail(c1/x)));
117     }

119     /* 1.0e40 <= x <= inf */
120     return (x*(logl(x)-c1));
121 }

```

unchanged portion omitted

new/usr/src/lib/libm/common/LD/__poly_libmq.c

1

1162 Tue Nov 25 12:57:01 2014

new/usr/src/lib/libm/common/LD/__poly_libmq.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #include "libm.h"
31 #include "libm_synonyms.h"
```

```
32 long double __poly_libmq(x,n,p)
33 long double x,p[];
34 int n;
35 {
36     long double t; int i;
37     t = p[n-1];
38     for(i=n-2;i>=0;i--) t = p[i] + x*t;
39     return t;
40 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/LD/acoshl.c

1

```
*****
1471 Tue Nov 25 12:57:02 2014
new/usr/src/lib/libm/common/LD/acoshl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __acoshl = acoshl
30 #if defined(ELFOSBJ)
31 #pragma weak acoshl = __acoshl
32 #endif
33
34 #include "libm.h"
35
36 static const long double
37 zero = 0.0L,
38 ln2 = 6.931471805599453094172321214581765680755e-0001L,
39 one = 1.0L,
40 big = 1.e+20L;
41
42 long double
43 acoshl(long double x) {
44     long double t;
45
46     if (isnanl(x))
47         return (x + x);
48     else if (x > big)
49         return (logl(x) + ln2);
50     else if (x > one) {
51         t = sqrtl(x - one);
```

new/usr/src/lib/libm/common/LD/acoshl.c

2

```
50         return (loglpl(t * (t + sqrtl(x + one))));
51     } else if (x == one)
52         return (zero);
53     else
54         return ((x - x) / (x - x));
55 }
unchanged_portion_omitted
```


new/usr/src/lib/libm/common/LD/asinhl.c

1

1610 Tue Nov 25 12:57:02 2014

new/usr/src/lib/libm/common/LD/asinhl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __asinhl = asinhl
30 #if defined(ELFOBJ)
31 #pragma weak asinhl = __asinhl
32 #endif
```

```
32 #include "libm.h"
```

```
34 static const long double
35     ln2      = 6.931471805599453094172321214581765680755e-0001L,
36     one      = 1.0L,
37     big      = 1.0e+20L,
38     tiny     = 1.0e-20L;
```

```
40 long double
41 asinhl(long double x) {
42     long double t, w;
43     #ifndef lint
44     volatile long double dummy;
45     #endif
```

```
47     w = fabsl(x);
48     if (isnanl(x))
49         return (x + x); /* x is NaN */
```

new/usr/src/lib/libm/common/LD/asinhl.c

2

```
50     if (w < tiny) {
51     #ifndef lint
52         dummy = x + big; /* inexact if x != 0 */
53     #endif
54         return (x); /* tiny x */
55     } else if (w < big) {
56         t = one / w;
57         return (copysignl(loglpl(w + w / (t + sqrtl(one + t * t))), x));
58     } else
59         return (copysignl(logl(w) + ln2, x));
60 }
    _____unchanged_portion_omitted
```

new/usr/src/lib/libm/common/LD/atan2pil.c

1

1321 Tue Nov 25 12:57:02 2014

new/usr/src/lib/libm/common/LD/atan2pil.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak atan2pil = __atan2pil
```

```
30 #include "libm.h"
33 #include "libm_synonyms.h"
```

```
32 #define GENERIC long double
33 #define ATAN2PI atan2pil
34 #define ATAN2 atan2l
```

```
36 /* ATAN2PI(y,x)
37  *
38  * ATAN2PI(y,x) = ATAN2(y,x)/pi
39 */
```

```
41 extern GENERIC ATAN2();
```

```
43 static GENERIC
44 invpi = (GENERIC) 3.183098861837906715377675267450287240689e-0001L;
```

```
46 GENERIC ATAN2PI(y,x)
47 GENERIC y,x;
48 {
49     return ATAN2(y,x)*invpi;
50 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/LD/atanhl.c

1

```
*****
1979 Tue Nov 25 12:57:03 2014
new/usr/src/lib/libm/common/LD/atanhl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __atanhl = atanhl
30 #pragma weak atanhl = __atanhl

32 #include "libm.h"
33 #include "libm_synonyms.h"

34 #define GENERIC long double
35 #define ATANH atanhl

37 /* ATANH(x)
38  *
39  * 
$$\text{ATANH}(x) = \frac{1}{2} * \text{LOG}\left(1 + \frac{2x}{1-x}\right) = 0.5 * \text{LOG1P}\left(2 * \frac{x}{1-x}\right)$$

40  *
41  * Note: to guarantee ATANH(-x) = -ATANH(x), we use
42  * 
$$\text{ATANH}(x) = \frac{\text{sign}(x)}{2} * \text{LOG1P}\left(2 * \frac{|x|}{1-|x|}\right).$$

43  *
44  *
45  *
46  * Special cases:
47  * ATANH(x) is NaN if |x| > 1 with signal;
48  * ATANH(NaN) is that NaN with no signal;
49  * ATANH(+1) is +-INF with signal.
50  *
51  */

53 #define FABS fabsl
54 #define LOG1P loglpl
55 #define COPYSIGN copysignl
```

new/usr/src/lib/libm/common/LD/atanhl.c

2

```
58 extern GENERIC FABS(),LOG1P(),COPYSIGN();

60 static GENERIC
61 zero = (GENERIC) 0.0,
62 half = (GENERIC) 0.5,
63 one = (GENERIC) 1.0;

65 GENERIC ATANH(x)
66 GENERIC x;
67 {
68     GENERIC t;
69     t = FABS(x);
70     if (t == one) return x/zero;
71     t = t/(one-t);
72     return COPYSIGN(half,x)*LOG1P(t+t);
73 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/LD/cbrtl.c

1

1752 Tue Nov 25 12:57:03 2014

new/usr/src/lib/libm/common/LD/cbrtl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cbrtl = cbrtl
30 #if defined(ELFOSBJ)
31 #pragma weak cbrtl = __cbrtl
32 #endif
```

```
32 #include "libm.h"
33 #include "longdouble.h"
```

```
35 static const double d_one = 1.0;
```

```
37 long double
38 cbrtl(long double x) {
39     long double s, t, r, w, y;
40     double dx, dy;
41     int *py = (int *) &dy;
42     int n, m, m3, n0, sx;
43
44     if (!finitel(x))
45         return (x + x);
46     if (iszerol(x))
47         return (x);
48     n0 = 0;
49     if (*(int *) &d_one == 0)
```

new/usr/src/lib/libm/common/LD/cbrtl.c

2

```
50         n0 = 1;
51         sx = signbitl(x);
52         x = fabsl(x);
53         n = ilogbl(x);
54         m = n / 3;
55         m3 = m + m + m;
56         y = scalbnl(x, -m3);
57         dx = (double) y;
58         dy = cbrt(dx);
59         py[1 - n0] += 2;
60         if (py[1 - n0] == 0)
61             py[n0] += 1;
62
63         /* one step newton iteration to 113 bits with error < 0.667ulps */
64         t = (long double) dy;
65         t = scalbnl(t, m);
66         s = t * t;
67         r = x / s;
68         w = t + t;
69         r = (r - t) / (w + r);
70         t += t * r;
71
72         return (sx == 0 ? t : -t);
73 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/coshl.c

1

```
*****
2788 Tue Nov 25 12:57:04 2014
new/usr/src/lib/libm/common/LD/coshl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __coshl = coshl
30 #if defined(ELFOBJ)
31 #pragma weak coshl = __coshl
32 #endif

32 #include "libm.h"
33 #include "longdouble.h"

35 /*
36  * COSH(X)
37  * RETURN THE HYPERBOLIC COSINE OF X
38  *
39  * Method :
40  * 1. Replace x by |x| (COSH(x) = COSH(-x)).
41  * 2.
42  *          [ EXP(x) - 1 ]^2
43  *          0      <= x <= 0.3465 : COSH(x) := 1 + -----
44  *                                     2*EXP(x)
45  *
46  *          EXP(x) + 1/EXP(x)
47  *          0.3465 <= x <= thresh : COSH(x) := -----
48  *                                     2
49  *          thresh <= x <= lnovft : COSH(x) := EXP(x)/2
```

new/usr/src/lib/libm/common/LD/coshl.c

2

```
50 *          lnovft <= x < INF      : COSH(x) := SCALBN(EXP(x-MEP1*ln2),ME)
51 *
52 *
53 * here
54 * 0.3465          a number that is near one half of ln2.
55 * thresh          a number such that
56 *                  EXP(thresh)+EXP(-thresh)=EXP(thresh)
57 * lnovft          logarithm of the overflow threshold
58 *                = MEP1*ln2 chopped to machine precision.
59 * ME              maximum exponent
60 * MEP1            maximum exponent plus 1
61 *
62 * Special cases:
63 * COSH(x) is |x| if x is +INF, -INF, or NaN.
64 * only COSH(0)=1 is exact for finite x.
65 */

67 static const long double C[] = {
68     0.5L,
69     1.0L,
70     0.3465L,
71     45.0L,
72     1.135652340629414394879149e+04L,
73     7.004447686242549087858985e-16L,
74     2.710505431213761085018632e-20L,
75 };
_____unchanged_portion_omitted_
```

```

*****
2789 Tue Nov 25 12:57:04 2014
new/usr/src/lib/libm/common/LD/cosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cosl = cosl
30 #pragma weak cosl = __cosl

32 /* INDENT OFF */
33 /* cosl(x)
34  * Table look-up algorithm by K.C. Ng, November, 1989.
35  *
36  * kernel function:
37  *   __k_sinl      ... sin function on [-pi/4,pi/4]
38  *   __k_cosl     ... cos function on [-pi/4,pi/4]
39  *   __rem_pio2l  ... argument reduction routine
40  *
41  * Method.
42  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
43  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
44  *   [-pi/2 , +pi/2], and let n = k mod 4.
45  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
46  *
47  *           n          sin(x)      cos(x)      tan(x)
48  *   -----
49  *           0           S           C           S/C
50  *           1           C          -S          -C/S
51  *           2          -S          -C           S/C
52  *           3          -C           S          -C/S
53  *   -----
54  *
55  * Special cases:
56  *   Let trig be any of sin, cos, or tan.
57  *   trig(+/-INF) is NaN, with signals;

```

```

58  *   trig(NaN) is that NaN;
59  *
60  * Accuracy:
61  *   computer TRIG(x) returns trig(x) nearly rounded.
62  */
63 /* INDENT ON */

65 #include "libm.h"
66 #include "libm_synonyms.h"
66 #include "longdouble.h"

68 #include <sys/isa_defs.h>

70 long double
71 cosl(long double x) {
72     long double y[2], z = 0.0L;
73     int n, ix;
74     int *px = (int *) &x;

76     /* trig(Inf or NaN) is NaN */
77     if (!finitel(x))
78         return x - x;

80     /* High word of x. */
81     #if defined(__i386) || defined(__amd64)
82         XTOI(px, ix);
83     #else
84         ix = px[0];
85     #endif

87     /* |x| ~< pi/4 */
88     ix &= 0x7fffffff;
89     if (ix <= 0x3ffe9220)
90         return __k_cosl(x, z);

92     /* argument reduction needed */
93     else {
94         n = __rem_pio2l(x, y);
95         switch (n & 3) {
96             case 0:
97                 return __k_cosl(y[0], y[1]);
98             case 1:
99                 return -__k_sinl(y[0], y[1]);
100             case 2:
101                 return -__k_cosl(y[0], y[1]);
102             case 3:
103                 return __k_sinl(y[0], y[1]);
104             /* NOTREACHED */
105             }
106         }
107     return 0.0L;
108 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/erfl.c

1

```
*****
12826 Tue Nov 25 12:57:04 2014
new/usr/src/lib/libm/common/LD/erfl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
30 /* long double function erf,erfc (long double x)
31  * K.C. Ng, September, 1989.
32  *
33  *
34  *      erf(x) = ----- \int_0^x exp(-t*t)dt
35  *                sqrt(pi)
36  *
37  *
38  *      erfc(x) = 1-erf(x)
39  *
40  * method:
41  * Since erf(-x) = -erf(x), we assume x>=0.
42  * For x near 0, we have the expansion
43  *
44  *      erf(x) = (2/sqrt(pi))*(x - x^3/3 + x^5/10 - x^7/42 + ...).
45  *
46  * Since 2/sqrt(pi) = 1.128379167095512573896158903121545171688,
47  * we use x + x*P(x^2) to approximate erf(x). This formula will
48  * guarantee the error less than one ulp where x is not too far
49  * away from 0. We note that erf(x)=x at x = 0.6174..... After
50  * some experiment, we choose the following approximation on
51  * interval [0,0.84375].
52  *
53  * For x in [0,0.84375]
54  *
55  *      P = P(x) = (p0 + p1 * x + p2 * x^2 + ... + p20 * x^40)
56  *
57  *      erf(x) = x + x*P
58  *      erfc(x) = 1 - erf(x)          if x<=0.25
```

new/usr/src/lib/libm/common/LD/erfl.c

2

```
59 *      = 0.5 + ((0.5-x)-x*P) if x in [0.25,0.84375]
60 *      precision: |P(x^2)-(erf(x)-x)/x| <= 2**(-122.50)
61 *
62 * For x in [0.84375,1.25], let s = x - 1, and
63 * c = 0.84506291151 rounded to single (24 bits)
64 *      erf(x) = c + P1(s)/Q1(s)
65 *      erfc(x) = (1-c) - P1(s)/Q1(s)
66 *      precision: |P1/Q1 - (erf(x)-c)| <= 2**(-118.41)
67 *
68 *
69 * For x in [1.25,1.75], let s = x - 1.5, and
70 * c = 0.95478588343 rounded to single (24 bits)
71 *      erf(x) = c + P2(s)/Q2(s)
72 *      erfc(x) = (1-c) - P2(s)/Q2(s)
73 *      precision: |P1/Q1 - (erf(x)-c)| <= 2**(-123.83)
74 *
75 *
76 * For x in [1.75,16/3]
77 *      erfc(x) = exp(-x*x)*(1/x)*R1(1/x)/S1(1/x)
78 *      erf(x) = 1 - erfc(x)
79 *      precision: absolute error of R1/S1 is bounded by 2**(-124.03)
80 *
81 * For x in [16/3,107]
82 *      erfc(x) = exp(-x*x)*(1/x)*R2(1/x)/S2(1/x)
83 *      erf(x) = 1 - erfc(x) (if x>=9 simple return erf(x)=1 with inexact)
84 *      precision: absolute error of R2/S2 is bounded by 2**(-120.07)
85 *
86 * Else if inf > x >= 107
87 *      erf(x) = 1 with inexact
88 *      erfc(x) = 0 with underflow
89 *
90 * Special case:
91 *      erf(inf) = 1
92 *      erfc(inf) = 0
93 */
95 #pragma weak __erfl = erfl
96 #pragma weak __erfc1 = erfc1
95 #pragma weak erfl = __erfl
96 #pragma weak erfc1 = __erfc1
98 #include "libm.h"
99 #include "longdouble.h"
101 static long double
102 tiny = 1e-40L,
103 nearunfl = 1e-4000L,
104 half = 0.5L,
105 one = 1.0L,
106 onehalf = 1.5L,
107 L16_3 = 16.0L/3.0L;
108 /*
109  * Coefficients for even polynomial P for erf(x)=x+x*P(x^2) on [0,0.84375]
110  */
111 static long double P[] = { /* 21 coeffs */
112  1.283791670955125738961589031215451715556e-0001L,
113  -3.761263890318375246320529677071815594603e-0001L,
114  1.128379167095512573896158903121205899135e-0001L,
115  -2.686617064513125175943235483344625046092e-0002L,
116  5.223977625442187842111846652980454568389e-0003L,
117  -8.548327023450852832546626271083862724358e-0004L,
118  1.205533298178966425102164715902231976672e-0004L,
119  -1.492565035840625097674944905027897838996e-0005L,
120  1.646211436588924733604648849172936692024e-0006L,
121  -1.636584469123491976815834704799733514987e-0007L,
122  1.480719281587897445302529007144770739305e-0008L,
```

new/usr/src/lib/libm/common/LD/erfl.c

3

```
123 -1.229055530170782843046467986464722047175e-0009L,  
124  9.422759064320307357553954945760654341633e-0011L,  
125 -6.711366846653439036162105104991433380926e-0012L,  
126  4.463224090341893165100275380693843116240e-0013L,  
127 -2.783513452582658245422635662559779162312e-0014L,  
128  1.634227412586960195251346878863754661546e-0015L,  
129 -9.060782672889577722765711455623117802795e-0017L,  
130  4.741341801266246873412159213893613602354e-0018L,  
131 -2.272417596497826188374846636534317381203e-0019L,  
132  8.069088733716068462496835658928566920933e-0021L,  
133  };  
    unchanged_portion_omitted
```


new/usr/src/lib/libm/common/LD/finitel.c

1

1412 Tue Nov 25 12:57:05 2014

new/usr/src/lib/libm/common/LD/finitel.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
31 #pragma weak finitel = __finitel
32 #endif
```

```
30 #include "libm.h"
```

```
32 #if defined(__sparc)
33 int
34 finitel(long double x) {
35     int *px = (int *) &x;
36     return ((px[0] & ~0x80000000) < 0x7fff0000);
37 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/LD/gammal.c

1

1319 Tue Nov 25 12:57:05 2014

new/usr/src/lib/libm/common/LD/gammal.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __gammal = gammal
30 #pragma weak gammal = __gammal

32 /*
33  * long double gammal(long double x);
34 */

36 #include "libm.h"
37 #include "libm_synonyms.h"
37 #include "longdouble.h"

39 extern int signgam;
40 extern int signgaml;

42 long double
43 gammal(long double x) {
44     long double y = __k_lgammal(x, &signgaml);

46     signgam = signgaml;    /* SUSv3 requires the setting of signgam */
47     return y;
48 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/LD/gammal_r.c

1

1231 Tue Nov 25 12:57:06 2014

new/usr/src/lib/libm/common/LD/gammal_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * long double gammal_r(long double x, int *signgamp);
32 */
33
34 #pragma weak __gammal_r = gammal_r
34 #pragma weak gammal_r = __gammal_r
35
36 #include "libm.h"
37 #include "longdouble.h"
38
39 long double
40 gammal_r(long double x, int *signgamp) {
41     return __k_lgammal(x, signgamp);
42 }
43
44 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/LD/hypot1.c

1

```
*****
3742 Tue Nov 25 12:57:06 2014
new/usr/src/lib/libm/common/LD/hypot1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __hypot1 = hypot1
30 #if defined(ELFOBJ)
31 #pragma weak hypot1 = __hypot1
32 #endif

32 /*
33  * hypot1(x,y)
34  * Method :
35  *   If z=x*x+y*y has error less than sqrt(2)/2 ulp than sqrt(z) has
36  *   error less than 1 ulp.
37  *   So, compute sqrt(x*x+y*y) with some care as follows:
38  *   Assume x>y>0;
39  *   1. save and set rounding to round-to-nearest
40  *   2. if x > 2y use
41  *       x1*x1+(y*y+(x2*(x+x2))) for x*x+y*y
42  *   where x1 = x with lower 32 bits cleared, x2 = x-x1; else
43  *   3. if x <= 2y use
44  *       t1*y1+((x-y)*(x-y)+(t1*y2+t2*y))
45  *   where t1 = 2x with lower 64 bits cleared, t2 = 2x-t1, y1 = y with
46  *   lower 32 bits cleared, y2 = y-y1.
47  *
48  * NOTE: DO NOT remove parenthesis!
49  */
```

new/usr/src/lib/libm/common/LD/hypot1.c

2

```
50  * Special cases:
51  *   hypot(x,y) is INF if x or y is +INF or -INF; else
52  *   hypot(x,y) is NAN if x or y is NAN.
53  *
54  * Accuracy:
55  *   hypot(x,y) returns sqrt(x^2+y^2) with error less than 1 ulps (units
56  *   in the last place)
57  */

59 #include "libm.h"

61 #if defined(__x86)
62 extern enum fp_direction_type __swap87RD(enum fp_direction_type);

64 #define k      0x7fff

66 long double
67 hypot1(long double x, long double y) {
68     long double t1, t2, y1, y2, w;
69     int *px = (int *) &x, *py = (int *) &y;
70     int *pt1 = (int *) &t1, *py1 = (int *) &y1;
71     enum fp_direction_type rd;
72     int j, nx, ny, nz;

74     px[2] &= 0x7fff;          /* clear sign bit and padding bits of x and y */
75     py[2] &= 0x7fff;
76     nx = px[2];              /* biased exponent of x and y */
77     ny = py[2];
78     if (ny > nx) {
79         w = x;
80         x = y;
81         y = w;
82         nz = ny;
83         ny = nx;
84         nx = nz;
85     }                          /* force nx >= ny */
86     if (nx - ny >= 66)
87         return (x + y); /* x / y >= 2**65 */
88     if (nx < 0x5ff3 && ny > 0x205b) { /* medium x,y */
89         /* save and set RD to Rounding to nearest */
90         rd = __swap87RD(fp_nearest);
91         w = x - y;
92         if (w > y) {
93             pt1[2] = px[2];
94             pt1[1] = px[1];
95             pt1[0] = 0;
96             t2 = x - t1;
97             x = sqrtl(t1 * t1 - (y * (-y) - t2 * (x + t1)));
98         } else {
99             x += x;
100            py1[2] = py[2];
101            py1[1] = py[1];
102            py1[0] = 0;
103            y2 = y - y1;
104            pt1[2] = px[2];
105            pt1[1] = px[1];
106            pt1[0] = 0;
107            t2 = x - t1;
108            x = sqrtl(t1 * y1 - (w * (-w) - (t2 * y1 + y2 * x)));
109        }
110        if (rd != fp_nearest)
111            __swap87RD(rd); /* restore rounding mode */
112        return (x);
113    } else {
114        if (nx == k || ny == k) { /* x or y is INF or NaN */
115            /* since nx >= ny; nx is always k within this block */
```

```
116         if (px[1] == 0x80000000 && px[0] == 0)
117             return (x);
118         else if (ny == k && py[1] == 0x80000000 && py[0] == 0)
119             return (y);
120         else
121             return (x + y);
122     }
123     if (ny == 0) {
124         if (y == 0.L || x == 0.L)
125             return (x + y);
126         ptl[2] = 0x3fff + 16381;
127         ptl[1] = 0x80000000;
128         ptl[0] = 0;
129         pyl[2] = 0x3fff - 16381;
130         pyl[1] = 0x80000000;
131         pyl[0] = 0;
132         x *= t1;
133         y *= t1;
134         return (y1 * hypot1(x, y));
135     }
136     j = nx - 0x3fff;
137     px[2] -= j;
138     py[2] -= j;
139     ptl[2] = nx;
140     ptl[1] = 0x80000000;
141     ptl[0] = 0;
142     return (t1 * hypot1(x, y));
143 }
144 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/LD/isnanl.c

1

1585 Tue Nov 25 12:57:07 2014

new/usr/src/lib/libm/common/LD/isnanl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __isnanl = isnanl
30 #if defined(ELFOSBJ)
31 #pragma weak isnanl = __isnanl
32 #endif
```

```
32 #include "libm.h"
```

```
34 #if defined(__sparc)
35 int
36 isnanl(long double x) {
37     int *px = (int *) &x;
38     return ((px[0] & ~0x80000000) >= 0x7fff0000 &&
39             ((px[0] & ~0xffff0000) | px[1] | px[2] | px[3]) != 0);
40 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/jnl.c

1

```
*****
6885 Tue Nov 25 12:57:08 2014
new/usr/src/lib/libm/common/LD/jnl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __jnl = jnl
31 #pragma weak __ynl = ynl
32 #if defined(ELFOBJ)
33 #pragma weak jnl = __jnl
34 #pragma weak ynl = __ynl
35 #endif
36
37 /*
38  * floating point Bessel's function of the 1st and 2nd kind
39  * of order n: jn(n,x), yn(n,x);
40  *
41  * Special cases:
42  * y0(0)=y1(0)=yn(n,0) = -inf with division by zero signal;
43  * y0(-ve)=y1(-ve)=yn(n,-ve) are NaN with invalid signal.
44  * Note 2. About jn(n,x), yn(n,x)
45  * For n=0, j0(x) is called,
46  * for n=1, j1(x) is called,
47  * for n>x, forward recursion is used starting
48  * from values of j0(x) and j1(x).
49  * for n>x, a continued fraction approximation to
50  * j(n,x)/j(n-1,x) is evaluated and then backward
51  * recursion is used starting from a supposed value
52  * for j(n,x). The resulting value of j(0,x) is
```

new/usr/src/lib/libm/common/LD/jnl.c

2

```
49 * compared with the actual value to correct the
50 * supposed value of j(n,x).
51 *
52 * yn(n,x) is similar in all respects, except
53 * that forward recursion is used for all
54 * values of n>1.
55 *
56 */
57
58 #include "libm.h"
59 #include "longdouble.h"
60 #include <float.h> /* LDBL_MAX */
61
62 #define GENERIC long double
63
64 static const GENERIC
65 invsqrtpi = 5.641895835477562869480794515607725858441e-0001L,
66 two = 2.0L,
67 zero = 0.0L,
68 one = 1.0L;
69
70 GENERIC
71 jnl(n, x) int n; GENERIC x; {
72     int i, sgn;
73     GENERIC a, b, temp = 0, z, w;
74
75     /*
76      * J(-n,x) = (-1)^n * J(n, x), J(n, -x) = (-1)^n * J(n, x)
77      * Thus, J(-n,x) = J(n,-x)
78      */
79     if (n < 0) {
80         n = -n;
81         x = -x;
82     }
83     if (n == 0) return (j0l(x));
84     if (n == 1) return (j1l(x));
85     if (x != x) return x+x;
86     if ((n&1) == 0)
87         sgn = 0; /* even n */
88     else
89         sgn = signbitl(x); /* odd n */
90     x = fabsl(x);
91     if (x == zero || !finitel(x)) b = zero;
92     else if ((GENERIC)n <= x) {
93         /*
94          * Safe to use
95          * J(n+1,x)=2n/x *J(n,x)-J(n-1,x)
96          */
97         if (x > 1.0e91L) {
98             /*
99              * x >> n**2
100             * Jn(x) = cos(x-(2n+1)*pi/4)*sqrt(2/x*pi)
101             * Yn(x) = sin(x-(2n+1)*pi/4)*sqrt(2/x*pi)
102             * Let s=sin(x), c=cos(x),
103             * xn=x-(2n+1)*pi/4, sqt2 = sqrt(2), then
104             *
105             *          n   sin(xn)*sqt2   cos(xn)*sqt2
106             * -----
107             *          0   s-c           c+s
108             *          1  -s-c           -c+s
109             *          2  -s+c           -c-s
110             *          3   s+c           c-s
111             */
112             switch (n&3) {
113                 case 0: temp = cosl(x)+sinl(x); break;
114                 case 1: temp = -cosl(x)+sinl(x); break;
```

```

115         case 2: temp = -cosl(x)-sinl(x); break;
116         case 3: temp =  cosl(x)-sinl(x); break;
117     }
118     b = invsqrtpi*temp/sqrtl(x);
119 } else {
120     a = j0l(x);
121     b = j1l(x);
122     for (i = 1; i < n; i++) {
123         temp = b;
124         b = b*((GENERIC)(i+i)/x) - a; /* avoid underflow */
125         a = temp;
126     }
127 }
128 } else {
129     if (x < 1e-17L) { /* use J(n,x) = 1/n!*(x/2)^n */
130         b = powl(0.5L*x, (GENERIC) n);
131         if (b != zero) {
132             for (a = one, i = 1; i <= n; i++) a *= (GENERIC)i;
133             b = b/a;
134         }
135     } else {
136         /*
137         * use backward recurrence
138         *
139         * 
$$J(n,x)/J(n-1,x) = \frac{x}{2n} - \frac{x^2}{2(n+1) - 2(n+2)} - \dots$$

140         *
141         *
142         * (for large x) 
$$= \frac{1}{2n} - \frac{1}{2(n+1)} - \frac{1}{2(n+2)} - \dots$$

143         *
144         *
145         * 
$$\frac{1}{x} - \frac{1}{x} - \frac{1}{x} - \dots$$

146         *
147         *
148         * Let w = 2n/x and h=2/x, then the above quotient
149         * is equal to the continued fraction:
150         *
151         * 
$$= \frac{1}{w - \frac{1}{w+h - \frac{1}{w+2h - \dots}}}$$

152         *
153         *
154         *
155         *
156         *
157         * To determine how many terms needed, let
158         * Q(0) = w, Q(1) = w(w+h) - 1,
159         * Q(k) = (w+k*h)*Q(k-1) - Q(k-2),
160         * When Q(k) > 1e4    good for single
161         * When Q(k) > 1e9    good for double
162         * When Q(k) > 1e17   good for quaduple
163         */
164     }
165     /* determin k */
166     GENERIC t, v;
167     double q0, q1, h, tmp; int k, m;
168     w = (n+n)/(double)x; h = 2.0/(double)x;
169     q0 = w; z = w+h; q1 = w*z - 1.0; k = 1;
170     while (q1 < 1.0e17) {
171         k += 1; z += h;
172         tmp = z*q1 - q0;
173         q0 = q1;
174         q1 = tmp;
175     }
176     m = n+n;
177     for (t = zero, i = 2*(n+k); i >= m; i -= 2) t = one/(i/x-t);
178     a = t;
179     b = one;
180     /*

```

```

181     * Estimate log((2/x)^n*n!) = n*log(2/x)+n*ln(n)
182     * hence, if n*(log(2n/x)) > ...
183     * single 8.872283935e+01
184     * double 7.09782712893383973096e+02
185     * long double 1.135652340629414394949193107797076500617
186     * then recurrent value may overflow and the result is
187     * likely underflow to zero.
188     */
189     tmp = n;
190     v = two/x;
191     tmp = tmp*logl(fabsl(v*tmp));
192     if (tmp < 1.1356523406294143949491931077970765e+04L) {
193         for (i = n-1; i > 0; i--) {
194             temp = b;
195             b = ((i+i)/x)*b - a;
196             a = temp;
197         }
198     } else {
199         for (i = n-1; i > 0; i--) {
200             temp = b;
201             b = ((i+i)/x)*b - a;
202             a = temp;
203             if (b > 1e1000L) {
204                 a /= b;
205                 t /= b;
206                 b = 1.0;
207             }
208         }
209     }
210     b = (t*j0l(x)/b);
211 }
212 }
213 if (sgn == 1)
214     return -b;
215 else
216     return b;
217 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/lgammal.c

1

1323 Tue Nov 25 12:57:08 2014

new/usr/src/lib/libm/common/LD/lgammal.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __lgammal = lgammal
30 #pragma weak lgammal = __lgammal
31
32 /*
33  * long double lgammal(long double x);
34 */
35
36 #include "libm.h"
37 #include "libm_synonyms.h"
37 #include "longdouble.h"
38
39 extern int signgam;
40 extern int signgaml;
41
42 long double
43 lgammal(long double x) {
44     long double y = __k_lgammal(x, &signgaml);
45
46     signgam = signgaml;    /* SUSv3 requires the setting of signgam */
47     return y;
48 }
49
50 _____
51 unchanged_portion_omitted
52 _____
```

new/usr/src/lib/libm/common/LD/lgamma_r.c

1

1235 Tue Nov 25 12:57:09 2014

new/usr/src/lib/libm/common/LD/lgamma_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 /*
31 * long double lgamma_r(long double x, int *signgamp);
32 */
```

```
34 #pragma weak __lgamma_r = lgamma_r
34 #pragma weak lgamma_r = __lgamma_r
```

```
36 #include "libm.h"
37 #include "longdouble.h"
```

```
39 long double
40 lgamma_r(long double x, int *signgamp) {
41     return __k_lgamma(x, signgamp);
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/log1pl.c

1

1592 Tue Nov 25 12:57:09 2014

new/usr/src/lib/libm/common/LD/log1pl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __log1pl = log1pl
30 #if defined(ELFOSBJ)
31 #pragma weak log1pl = __log1pl
32 #endif
33
34 /*
35  * log1pl(x)
36  * Kahan's trick based on log(1+x)/x being a slow varying function.
37 */
38
39 #include "libm.h"
40
41 #if defined(__x86)
42 #define __swapRD __swap87RD
43 #endif
44 extern enum fp_direction_type __swapRD(enum fp_direction_type);
45
46 long double
47 log1pl(long double x) {
48     long double y;
49     enum fp_direction_type rd;
50
51     if (x != x)
```

new/usr/src/lib/libm/common/LD/log1pl.c

2

```
50         return (x + x);
51     if (x < -1.L)
52         return (logl(x));
53     rd = __swapRD(fp_nearest);
54     y = 1.L + x;
55     if (y != 1.L) {
56         if (y == x)
57             x = logl(x);
58         else
59             x *= logl(y) / (y - 1.L);
60     }
61     if (rd != fp_nearest)
62         (void) __swapRD(rd);
63     return (x);
64 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/LD/logbl.c

1

```
*****
2468 Tue Nov 25 12:57:10 2014
new/usr/src/lib/libm/common/LD/logbl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __logbl = logbl
30 #if defined(ELFOBJ)
31 #pragma weak logbl = __logbl
32 #endif

32 #include "libm.h"
33 #include "xpg6.h" /* __xpg6 */
34 #define _C99SUSv3_logb _C99SUSv3_logb_subnormal_is_like_ilogb

36 #if defined(__sparc)
37 #define ISNORMALL(k, x) (k != 0x7fff) /* assuming k != 0 */
38 #define X86PDNRM(k, x)
39 #define XSCALE_OFFSET 0x406f /* 0x3fff + 112 */
40 static const long double xscale = 5192296858534827628530496329220096.0L;
41 /* 2^112 */
42 #elif defined(__x86)
43 /*
44  * if pseudo-denormal, replace by the equivalent normal
45  */
46 #define X86PDNRM(k, x) if (k == 0 && (((int *) &x)[1] & 0x80000000) != 0) \
47 ((int *) &x)[2] |= k = 1
48 #if defined(HANDLE_UNSUPPORTED)
49 #define ISNORMALL(k, x) (k != 0x7fff && (((int *) &x)[1] & 0x80000000) != 0)

```

new/usr/src/lib/libm/common/LD/logbl.c

2

```
50 #else
51 #define ISNORMALL(k, x) (k != 0x7fff)
52 #endif
53 #define XSCALE_OFFSET 0x403e /* 0x3fff + 63 */
54 static const long double xscale = 9223372036854775808.0L; /* 2^63 */
55 #endif

57 static long double
58 raise_division(long double v) {
59 #pragma STDC FENV_ACCESS ON
60 static const long double zero = 0.0L;
61 return (v / zero);
62 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/LD/nextafterl.c

1

```
*****
2732 Tue Nov 25 12:57:10 2014
new/usr/src/lib/libm/common/LD/nextafterl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __nextafterl = nextafterl
30 #if defined(ELFOBJ)
31 #pragma weak nextafterl = __nextafterl
32 #endif
33
34 #include "libm.h"
35 #include <float.h>
36
37 #if defined(__sparc)
38 #define n0 0
39 #define n1 1
40 #define n2 2
41 #define n3 3
42 #define X86PDNRML(x)
43 #define INC(px) { \
44     if (++px[n3] == 0) \
45         if (++px[n2] == 0) \
46             if (++px[n1] == 0) \
47                 ++px[n0]; \
48 }
49 #define DEC(px) { \
50     if (--px[n3] == 0xfffffff) \
51         if (--px[n2] == 0xfffffff) \
```

new/usr/src/lib/libm/common/LD/nextafterl.c

2

```
52     if (--px[n1] == 0xffffffff) \
53         --px[n0]; \
54 }
55 #elif defined(__x86)
56 #define n0 2
57 #define n1 1
58 #define n2 0
59 #define n3 0
60 /*
61  * if pseudo-denormal, replace by the equivalent normal
62  */
63 #define X86PDNRML(x) if (XBIASED_EXP(x) == 0 && (((int *) &x)[1] & \
64     0x80000000) != 0) \
65     ((int *) &x)[2] |= 1
66 #define INC(px) { \
67     if (++px[n2] == 0) \
68         if ((++px[n1] & ~0x80000000) == 0) \
69             px[n1] = 0x80000000, ++px[n0]; \
70 }
71 #define DEC(px) { \
72     if (--px[n2] == 0xffffffff) \
73         if (--px[n1] == 0x7fffff) \
74             if ((--px[n0] & 0x7fff) != 0) \
75                 px[n1] |= 0x80000000; \
76 }
77 #endif
78
79 long double
80 nextafterl(long double x, long double y) {
81     int *px = (int *) &x;
82     int *py = (int *) &y;
83
84     if (x == y)
85         return (y);
86     /* C99 requirement */
87     if (x != x || y != y)
88         return (x * y);
89
90     if (ISZEROL(x)) {
91         /* x == 0.0 */
92         px[n0] = py[n0] & XSGNMSK;
93         px[n1] = px[n2] = 0;
94         px[n3] = 1;
95     } else {
96         X86PDNRML(x);
97         if ((px[n0] & XSGNMSK) == 0) {
98             /* x > 0.0 */
99             if (x > y)
100                 /* x > y */
101                 DEC(px);
102             else
103                 INC(px);
104         } else {
105             if (x < y)
106                 /* x < y */
107                 DEC(px);
108             else
109                 INC(px);
110         }
111     }
112 #ifndef lint
113     {
114         volatile long double dummy;
115         int k = XBIASED_EXP(x);
116
117         if (k == 0)
118             dummy = LDBL_MIN * copysign(LDBL_MIN, x);
119         else if (k == 0x7fff)
120             dummy = LDBL_MAX * copysign(LDBL_MAX, x);
121     }
122 #endif
123 }
124 #endif
```

new/usr/src/lib/libm/common/LD/nextafter1.c

3

```
116     return (x);  
117 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/LD/scalbl.c

1

```
*****
1768 Tue Nov 25 12:57:10 2014
new/usr/src/lib/libm/common/LD/scalbl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __scalbl = scalbl
30 #pragma weak scalbl = __scalbl

32 /*
33  * scalbl(x,n): return x * 2**n by manipulating exponent.
34 */

36 #include "libm.h"
37 #include "longdouble.h"

39 #include <sys/isa_defs.h>

41 long double
42 scalbl(long double x, long double fn) {
43     int *py = (int *) &fn, n;
44     long double z;

46     if (isnanl(x) || isnanl(fn))
47         return x * fn;

49     /* fn is +/-Inf */
50 #if defined(_BIG_ENDIAN)
51     if ((py[0] & 0x7fff0000) == 0x7fff0000) {
52         if ((py[0] & 0x80000000) != 0)
53 #else
54     if ((py[2] & 0x7fff) == 0x7fff) {
55         if ((py[2] & 0x8000) != 0)
56 #endif
57         return x / (-fn);
```

new/usr/src/lib/libm/common/LD/scalbl.c

2

```
58         else
59             return x * fn;
60     }
61     if (rintl(fn) != fn)
62         return (fn - fn) / (fn - fn);
63     if (fn > 65000.0L)
64         z = scalbnl(x, 65000);
65     else if (-fn > 65000.0L)
66         z = scalbnl(x, -65000);
67     else {
68         n = (int) fn;
69         z = scalbnl(x, n);
70     }
71     return z;
72 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/LD/signgaml.c

1

1094 Tue Nov 25 12:57:11 2014

new/usr/src/lib/libm/common/LD/signgaml.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __signgaml = signgaml
30 #pragma weak signgaml = __signgaml
```

```
32 #include "libm.h"
33 #include "libm_synonyms.h"
33 #include "longdouble.h"
```

```
35 int signgaml = 0;
```

new/usr/src/lib/libm/common/LD/significandl.c

1

1226 Tue Nov 25 12:57:11 2014

new/usr/src/lib/libm/common/LD/significandl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __significandl = significandl
30 #if defined(ELFOSBJ)
31 #pragma weak significandl = __significandl
32 #endif
```

```
32 #include "libm.h"
```

```
34 long double
35 significandl(long double x) {
36     if (ISZEROL(x) || XBIASED_EXP(x) == 0x7fff) /* 0/+Inf/NaN */
37         return (x + x);
38     else
39         return (scalbnl(x, -ilogbl(x)));
40 }
```

unchanged_portion_omitted_

```

*****
2900 Tue Nov 25 12:57:12 2014
new/usr/src/lib/libm/common/LD/sincosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __sincosl = sincosl
31 #pragma weak sincosl = __sincosl
32
33 /* INDENT OFF */
34 /* cosl(x)
35  * Table look-up algorithm by K.C. Ng, November, 1989.
36  *
37  * kernel function:
38  *   __k_sincosl    ... sin and cos function on [-pi/4,pi/4]
39  *   __rem_pio2l   ... argument reduction routine
40  *
41  * Method.
42  * Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
43  * 1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
44  *    [-pi/2, +pi/2], and let n = k mod 4.
45  * 2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
46  *
47  *      n      sin(x)      cos(x)      tan(x)
48  * -----
49  *      0      S          C          S/C
50  *      1      C          -S         -C/S
51  *      2      -S         -C          S/C
52  *      3      -C          S         -C/S
53  * -----
54  *
55  * Special cases:
56  * Let trig be any of sin, cos, or tan.
57  * trig(+/-INF) is NaN, with signals;
58  * trig(NaN) is that NaN;

```

```

58 *
59 * Accuracy:
60 * computer TRIG(x) returns trig(x) nearly rounded.
61 */
62 /* INDENT ON */
63
64 #include "libm.h"
65 #include "libm_synonyms.h"
66 #include "longdouble.h"
67
68 #include <sys/isa_defs.h>
69
70 void
71 sincosl(long double x, long double *s, long double *c) {
72     long double y[2], z = 0.0L;
73     int n, ix;
74     #if defined(__i386) || defined(__amd64)
75     int *px = (int *) &x;
76     #endif
77
78     /* trig(Inf or NaN) is NaN */
79     if (!finitel(x)) {
80         *s = *c = x - x;
81         return;
82     }
83
84     /* High word of x. */
85     #if defined(__i386) || defined(__amd64)
86     XTOI(px, ix);
87     #else
88     ix = *(int *) &x;
89     #endif
90
91     /* |x| ~< pi/4 */
92     ix &= 0x7fffffff;
93     if (ix <= 0x3ffe9220)
94         *s = __k_sincosl(x, z, c);
95
96     /* argument reduction needed */
97     else {
98         n = __rem_pio2l(x, y);
99         switch (n & 3) {
100            case 0:
101                *s = __k_sincosl(y[0], y[1], c);
102                break;
103            case 1:
104                *c = -__k_sincosl(y[0], y[1], s);
105                break;
106            case 2:
107                *s = -__k_sincosl(y[0], y[1], c);
108                *c = -*c;
109                break;
110            case 3:
111                *c = __k_sincosl(y[0], y[1], s);
112                *s = -*s;
113            }
114 }
115
116 _____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/LD/sincospil.c

1

```
*****
5985 Tue Nov 25 12:57:12 2014
new/usr/src/lib/libm/common/LD/sincospil.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak sincospil = __sincospil
31
32 /*
33  * void sincospil(long double x, long double *s, long double *c)
34  * *s = sinl(pi*x); *c = cosl(pi*x);
35  *
36  * Algorithm, 10/17/2002, K.C. Ng
37  * -----
38  * Let y = |4x|, z = floor(y), and n = (int)(z mod 8.0) (displayed in binary).
39  * 1. If y == z, then x is a multiple of pi/4. Return the following values:
40  *
41  *      n  x mod 2   sin(x*pi)   cos(x*pi)   tan(x*pi)
42  * -----
43  *      000  0.00    +0          +1          +0
44  *      001  0.25    +\0.5      +\0.5      +1
45  *      010  0.50    +1          +0          +inf
46  *      011  0.75    +\0.5      -\0.5      -1
47  *      100  1.00    -0          -1          +0
48  *      101  1.25    -\0.5      -\0.5      +1
49  *      110  1.50    -1          -0          +inf
50  *      111  1.75    -\0.5      +\0.5      -1
51  * -----
52  * 2. Otherwise,
53  *
54  *      n      t      sin(x*pi)   cos(x*pi)   tan(x*pi)
55  * -----
56  *      000  (y-z)/4   sinpi(t)   cospi(t)   tanpi(t)
57  *      001  (z+1-y)/4 cospi(t)   sinpi(t)   1/tanpi(t)
58  *      010  (y-z)/4   cospi(t)  -sinpi(t)  -1/tanpi(t)
```

new/usr/src/lib/libm/common/LD/sincospil.c

2

```
57 *      011  (z+1-y)/4   sinpi(t)  -cospi(t)  -tanpi(t)
58 *      100  (y-z)/4    -sinpi(t) -cospi(t)  tanpi(t)
59 *      101  (z+1-y)/4  -cospi(t) -sinpi(t)  1/tanpi(t)
60 *      110  (y-z)/4    -cospi(t) sinpi(t)   -1/tanpi(t)
61 *      111  (z+1-y)/4  -sinpi(t) cospi(t)   -tanpi(t)
62 * -----
63 *
64 * NOTE. This program compute sinpi/cospi(t<0.25) by __k_sin/cos(pi*t, 0.0).
65 * This will return a result with error slightly more than one ulp (but less
66 * than 2 ulp). If one wants accurate result, one may break up pi*t in
67 * high (tpi_h) and low (tpi_l) parts and call __k_sin/cos(tpi_h, tpi_lo)
68 * instead.
69 */
70
71 #include "libm.h"
72 #include "libm_synonyms.h"
73 #include "longdouble.h"
74
75 #include <sys/isa_defs.h>
76
77 #define I(q, m) ((int *) &(q))[m]
78 #define U(q, m) ((unsigned *) &(q))[m]
79 #if defined(__i386) || defined(__amd64)
80 #define LDBL_MOST_SIGNIF_I(ld) ((I(ld, 2) << 16) | (0xffff & (I(ld, 1) >> 15)))
81 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, 0)
82 #define PREC 64
83 #define PRECM1 63
84 #define PRECM2 62
85 static const long double twoPRECM2 = 9.2233720368547758080000000000000000e+18L;
86 #else
87 #define LDBL_MOST_SIGNIF_I(ld) I(ld, 0)
88 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, sizeof(long double) / sizeof(int) - 1)
89 #define PREC 113
90 #define PRECM1 112
91 #define PRECM2 111
92 static const long double twoPRECM2 = 5.192296858534827628530496329220096e+33L;
93 #endif
94
95 static const long double
96 zero = 0.0L,
97 quater = 0.25L,
98 one = 1.0L,
99 pi = 3.141592653589793238462643383279502884197e+0000L,
100 sqrtth = 0.70710678118654752440084436210484903284835937688474,
101 tiny = 1.0e-100;
102
103 void
104 sincospil(long double x, long double *s, long double *c) {
105     long double y, z, t;
106     int hx, n, k;
107     unsigned lx;
108
109     hx = LDBL_MOST_SIGNIF_I(x);
110     lx = LDBL_LEAST_SIGNIF_U(x);
111     k = ((hx & 0x7fff0000) >> 16) - 0x3fff;
112     if (k >= PRECM2) { /* |x| >= 2**(Prec-2) */
113         if (k >= 16384) {
114             *s = *c = x - x;
115         }
116     } else {
117         if (k >= PREC) {
118             *s = zero;
119             *c = one;
120         }
121     } else if (k == PRECM1) {
122         if ((lx & 1) == 0) {
```

```

122         *s = zero;
123         *c = one;
124     }
125     else {
126         *s = -zero;
127         *c = -one;
128     }
129 }
130 else { /* k = Prec - 2 */
131     if ((lx & 1) == 0) {
132         *s = zero;
133         *c = one;
134     }
135     else {
136         *s = one;
137         *c = zero;
138     }
139     if ((lx & 2) != 0) {
140         *s = -*s;
141         *c = -*c;
142     }
143 }
144 }
145 }
146 else if (k < -2) /* |x| < 0.25 */
147     *s = __k_sincosl(pi * fabs1(x), zero, c);
148 else {
149     /* y = |4x|, z = floor(y), and n = (int)(z mod 8.0) */
150     y = 4.0L * fabs1(x);
151     if (k < PRECM2) {
152         z = y + twoPRECM2;
153         n = LDBL_LEAST_SIGNIF_U(z) & 7; /* 3 LSB of z */
154         t = z - twoPRECM2;
155         k = 0;
156         if (t == y)
157             k = 1;
158         else if (t > y) {
159             n -= 1;
160             t = quater + (y - t) * quater;
161         }
162         else
163             t = (y - t) * quater;
164     }
165     else { /* k = Prec-3 */
166         n = LDBL_LEAST_SIGNIF_U(y) & 7; /* 3 LSB of z */
167         k = 1;
168     }
169     if (k) { /* x = N/4 */
170         if ((n & 1) != 0)
171             *s = *c = sqrth + tiny;
172         else
173             if ((n & 2) == 0) {
174                 *s = zero;
175                 *c = one;
176             }
177             else {
178                 *s = one;
179                 *c = zero;
180             }
181         if ((n & 4) != 0)
182             *s = -*s;
183         if (((n + 1) & 4) != 0)
184             *c = -*c;
185     }
186     else {
187         if ((n & 1) != 0)

```

```

188         t = quater - t;
189         if (((n + (n & 1)) & 2) == 0)
190             *s = __k_sincosl(pi * t, zero, c);
191         else
192             *c = __k_sincosl(pi * t, zero, s);
193         if ((n & 4) != 0)
194             *s = -*s;
195         if (((n + 2) & 4) != 0)
196             *c = -*c;
197     }
198 }
199 if (hx < 0)
200     *s = -*s;
201 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/LD/sinh1.c

1

```
*****
2237 Tue Nov 25 12:57:12 2014
new/usr/src/lib/libm/common/LD/sinh1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __sinh1 = sinh1
30 #pragma weak sinh1 = __sinh1
31
32 #include "libm.h"
33 #include "longdouble.h"
34
35 /* SINH(X)
36  * RETURN THE HYPERBOLIC SINE OF X
37  *
38  * Method :
39  * 1. reduce x to non-negative by SINH(-x) = - SINH(x).
40  * 2.
41  *
42  *          EXPM1(x) + EXPM1(x)/(EXPM1(x)+1)
43  * 0 <= x <= lnovft      : SINH(x) := -----
44  *                                     2
45  *
46  * lnovft <= x < INF      : SINH(x) := EXP(x-MEP1*ln2)*2**ME
47  *
48  * here
49  * lnovft      logarithm of the overflow threshold
50  *              = MEP1*ln2 chopped to machine precision.
51  * ME          maximum exponent
52  * MEP1       maximum exponent plus 1
53  *
54  * Special cases:
55  * SINH(x) is x if x is +INF, -INF, or NaN.
56  * only SINH(0)=0 is exact for finite argument.
57  */
```

new/usr/src/lib/libm/common/LD/sinh1.c

2

```
58 */
59
60 static const long double C[] = {
61     0.5L,
62     1.0L,
63     1.135652340629414394879149e+04L,
64     7.004447686242549087858985e-16L
65 };
_____unchanged_portion_omitted_
```

```

*****
2839 Tue Nov 25 12:57:13 2014
new/usr/src/lib/libm/common/LD/sinl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __sinl = sinl
30 #pragma weak sinl = __sinl

32 /* INDENT OFF */
33 /* sinl(x)
34  * Table look-up algorithm by K.C. Ng, November, 1989.
35  *
36  * kernel function:
37  *   __k_sinl      ... sin function on [-pi/4,pi/4]
38  *   __k_cosl     ... cos function on [-pi/4,pi/4]
39  *   __rem_pio2l  ... argument reduction routine
40  *
41  * Method.
42  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
43  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
44  *   [-pi/2 , +pi/2], and let n = k mod 4.
45  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
46  *
47  *   n      sin(x)      cos(x)      tan(x)
48  *   -----
49  *   0      S          C          S/C
50  *   1      C          -S         -C/S
51  *   2      -S         -C         S/C
52  *   3      -C          S         -C/S
53  *   -----
54  *
55  * Special cases:
56  *   Let trig be any of sin, cos, or tan.
57  *   trig(+/-INF) is NaN, with signals;

```

```

58 *   trig(NaN) is that NaN;
59 *
60 * Accuracy:
61 *   computer TRIG(x) returns trig(x) nearly rounded.
62 */
63 /* INDENT ON */

65 #include "libm.h"
66 #include "libm_synonyms.h"
66 #include "longdouble.h"

68 #include <sys/isa_defs.h>

70 long double
71 sinl(long double x) {
72     long double y[2], z = 0.0L;
73     int n, ix;
74     #if defined(__i386) || defined(__amd64)
75     int *px = (int *) &x;
76     #endif

78     /* sin(Inf or NaN) is NaN */
79     if (!finitel(x))
80         return x - x;

82     /* High word of x. */
83     #if defined(__i386) || defined(__amd64)
84     XTOI(px, ix);
85     #else
86     ix = *(int *) &x;
87     #endif
88     /* |x| ~< pi/4 */
89     ix &= 0x7fffffff;
90     if (ix <= 0x3ffe9220)
91         return __k_sinl(x, z);

93     /* argument reduction needed */
94     else {
95         n = __rem_pio2l(x, y);
96         switch (n & 3) {
97             case 0:
98                 return __k_sinl(y[0], y[1]);
99             case 1:
100                return __k_cosl(y[0], y[1]);
101             case 2:
102                return -__k_sinl(y[0], y[1]);
103             case 3:
104                return -__k_cosl(y[0], y[1]);
105                /* NOTREACHED */
106            }
107         }
108     return 0.0L;
109 }

unchanged_portion_omitted

```


new/usr/src/lib/libm/common/LD/sinpil.c

1

```
*****
5538 Tue Nov 25 12:57:13 2014
new/usr/src/lib/libm/common/LD/sinpil.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak sinpil = __sinpil
31
32 /* long double sinpil(long double x),
33  * return long double precision sinl(pi*x).
34  *
35  * Algorithm, 10/17/2002, K.C. Ng
36  * -----
37  * Let y = |4x|, z = floor(y), and n = (int)(z mod 8.0) (displayed in binary).
38  * 1. If y == z, then x is a multiple of pi/4. Return the following values:
39  * -----
40  *
41  *      n  x mod 2  sin(x*pi)  cos(x*pi)  tan(x*pi)
42  * -----
43  *      000  0.00      +0 -----      +1 -----      +0
44  *      001  0.25      +\sqrt{0.5}      +\sqrt{0.5}      +1
45  *      010  0.50      +1 -----      +0 -----      +inf
46  *      011  0.75      +\sqrt{0.5}      -\sqrt{0.5}      -1
47  *      100  1.00      -0 -----      -1 -----      +0
48  *      101  1.25      -\sqrt{0.5}      -\sqrt{0.5}      +1
49  *      110  1.50      -1 -----      -0 -----      +inf
50  *      111  1.75      -\sqrt{0.5}      +\sqrt{0.5}      -1
51  * -----
52  * 2. Otherwise,
53  * -----
54  *
55  *      n      t      sin(x*pi)      cos(x*pi)      tan(x*pi)
56  * -----
57  *      000  (y-z)/4      sinpi(t)      cospi(t)      tanpi(t)
58  *      001  (z+1-y)/4      cospi(t)      sinpi(t)      1/tanpi(t)
59  *      010  (y-z)/4      cospi(t)      -sinpi(t)      -1/tanpi(t)
60  *      011  (z+1-y)/4      sinpi(t)      -cospi(t)      -tanpi(t)
```

new/usr/src/lib/libm/common/LD/sinpil.c

2

```
57 *      100  (y-z)/4      -sinpi(t)      -cospi(t)      tanpi(t)
58 *      101  (z+1-y)/4      -cospi(t)      -sinpi(t)      1/tanpi(t)
59 *      110  (y-z)/4      -cospi(t)      sinpi(t)      -1/tanpi(t)
60 *      111  (z+1-y)/4      -sinpi(t)      cospi(t)      -tanpi(t)
61 * -----
62 *
63 * NOTE. This program compute sinpi/cospi(t<0.25) by __k_sin/cos(pi*t, 0.0).
64 * This will return a result with error slightly more than one ulp (but less
65 * than 2 ulp). If one wants accurate result, one may break up pi*t in
66 * high (tpi_h) and low (tpi_l) parts and call __k_sin/cos(tip_h, tip_lo)
67 * instead.
68 */
69
70 #include "libm.h"
71 #include "libm_synonyms.h"
72 #include "longdouble.h"
73
74 #include <sys/isa_defs.h>
75
76 #define I(q, m) ((int *) &(q))[m]
77 #define U(q, m) ((unsigned *) &(q))[m]
78 #if defined(__i386) || defined(__amd64)
79 #define LDBL_MOST_SIGNIF_I(ld) ((I(ld, 2) << 16) | (0xffff & (I(ld, 1) >> 15)))
80 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, 0)
81 #define PREC 64
82 #define PRECM1 63
83 #define PRECM2 62
84 static const long double twoPRECM2 = 9.223372036854775808000000000000000000e+18L;
85 #else
86 #define LDBL_MOST_SIGNIF_I(ld) I(ld, 0)
87 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, sizeof(long double) / sizeof(int) - 1)
88 #define PREC 113
89 #define PRECM1 112
90 #define PRECM2 111
91 static const long double twoPRECM2 = 5.192296858534827628530496329220096e+33L;
92 #endif
93
94 static const long double
95 zero = 0.0L,
96 quater = 0.25L,
97 one = 1.0L,
98 pi = 3.141592653589793238462643383279502884197e+0000L,
99 sqrt2 = 0.707106781186547524400844362104849039284835937688474,
100 tiny = 1.0e-100;
101
102 long double
103 sinpil(long double x) {
104     long double y, z, t;
105     int hx, n, k;
106     unsigned lx;
107
108     hx = LDBL_MOST_SIGNIF_I(x);
109     lx = LDBL_LEAST_SIGNIF_U(x);
110     k = ((hx & 0x7fff0000) >> 16) - 0x3fff;
111     if (k >= PRECM2) { /* |x| >= 2** (Prec-2) */
112         if (k >= 16384)
113             y = x - x;
114         else {
115             if (k >= PREC)
116                 y = zero;
117             else if (k == PRECM1)
118                 y = (lx & 1) == 0 ? zero : -zero;
119             else { /* k = Prec - 2 */
120                 y = (lx & 1) == 0 ? zero : one;
121                 if ((lx & 2) != 0)
122                     y = -y;
123             }
124         }
125     }
126 }
```

```

122     }
123 }
124 }
125 else if (k < -2) /* |x| < 0.25 */
126 y = __k_sinl(pi * fabsl(x), zero);
127 else {
128 /* y = |4x|, z = floor(y), and n = (int)(z mod 8.0) */
129 y = 4.0L * fabsl(x);
130 if (k < PRECM2) {
131 z = y + twoPRECM2;
132 n = LDBL_LEAST_SIGNIF_U(z) & 7; /* 3 LSB of z */
133 t = z - twoPRECM2;
134 k = 0;
135 if (t == y)
136 k = 1;
137 else if (t > y) {
138 n -= 1;
139 t = quater + (y - t) * quater;
140 }
141 else
142 t = (y - t) * quater;
143 }
144 else { /* k = Prec-3 */
145 n = LDBL_LEAST_SIGNIF_U(y) & 7; /* 3 LSB of z */
146 k = 1;
147 }
148 if (k) { /* x = N/4 */
149 if ((n & 1) != 0)
150 y = sqrth + tiny;
151 else
152 y = (n & 2) == 0 ? zero : one;
153 if ((n & 4) != 0)
154 y = -y;
155 }
156 else {
157 if ((n & 1) != 0)
158 t = quater - t;
159 if (((n + (n & 1)) & 2) == 0)
160 y = __k_sinl(pi * t, zero);
161 else
162 y = __k_cosl(pi * t, zero);
163 if ((n & 4) != 0)
164 y = -y;
165 }
166 }
167 return hx >= 0 ? y : -y;
168 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/LD/tanh1.c

1

```
*****
2601 Tue Nov 25 12:57:14 2014
new/usr/src/lib/libm/common/LD/tanh1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __tanh1 = tanh1
30 #if defined(ELFOBJ)
31 #pragma weak tanh1 = __tanh1
32 #endif

32 /*
33  * tanh1(x) returns the hyperbolic tangent of x
34  *
35  * Method :
36  * 1. reduce x to non-negative: tanh1(-x) = - tanh1(x).
37  * 2.
38  * 0 < x <= small : tanh1(x) := x
39  * -expm1(-2x)
40  * small < x <= 1 : tanh1(x) := -----
41  * expm1(-2x) + 2
42  * 2
43  * 1 <= x <= threshold : tanh1(x) := 1 - -----
44  * expm1(2x) + 2
45  * threshold < x <= INF : tanh1(x) := 1.
46  *
47  * where
48  * single : small = 1.e-5 threshold = 11.0
49  * double : small = 1.e-10 threshold = 22.0
```

new/usr/src/lib/libm/common/LD/tanh1.c

2

```
50 * quad : small = 1.e-20 threshold = 45.0
51 *
52 * Note: threshold was chosen so that
53 * fl(1.0+2/(expm1(2*threshold)+2)) == 1.
54 *
55 * Special cases:
56 * tanh1(NaN) is NaN;
57 * only tanh1(0.0)=0.0 is exact for finite argument.
58 */

60 #include "libm.h"
61 #include "longdouble.h"

63 static const long double small = 1.0e-20L, one = 1.0, two = 2.0,
64 #ifndef lint
65 big = 1.0e+20L,
66 #endif
67 threshold = 45.0L;

69 long double
70 tanh1(long double x) {
71     long double t, y, z;
72     int signx;
73 #ifndef lint
74     volatile long double dummy;
75 #endif

77     if (isnanl(x))
78         return (x + x); /* x is NaN */
79     signx = signbitl(x);
80     t = fabsl(x);
81     z = one;
82     if (t <= threshold) {
83         if (t > one)
84             z = one - two / (expm1l(t + t) + two);
85         else if (t > small) {
86             y = expm1l(-t - t);
87             z = -y / (y + two);
88         } else {
89 #ifndef lint
90             dummy = t + big;
91             /* inexact if t != 0 */
92 #endif
93             return (x);
94         }
95     } else if (!finitel(t))
96         return (copysignl(one, x));
97     else
98         return (signx ? -z + small * small : z - small * small);
99     return (signx ? -z : z);
100 }

_____ unchanged_portion_omitted
```

```

*****
2613 Tue Nov 25 12:57:15 2014
new/usr/src/lib/libm/common/LD/tanl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __tanl = tanl
30 #pragma weak tanl = __tanl

32 /* INDENT OFF */
33 /* cosl(x)
34  * Table look-up algorithm by K.C. Ng, November, 1989.
35  *
36  * kernel function:
37  *   __k_tanl      ... tangent function on [-pi/4,pi/4]
38  *   __rem_pio2l  ... argument reduction routine
39  *
40  * Method.
41  * Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
42  * 1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
43  *    [-pi/2, +pi/2], and let n = k mod 4.
44  * 2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
45  *
46  *      n      sin(x)      cos(x)      tan(x)
47  * -----
48  *      0      S          C          S/C
49  *      1      C          -S         -C/S
50  *      2      -S         -C         S/C
51  *      3      -C          S         -C/S
52  * -----
53  *
54  * Special cases:
55  * Let trig be any of sin, cos, or tan.
56  * trig(+/-INF) is NaN, with signals;
57  * trig(NaN) is that NaN;

```

```

58 *
59 * Accuracy:
60 * computer TRIG(x) returns trig(x) nearly rounded.
61 */
62 /* INDENT ON */

64 #include "libm.h"
65 #include "libm_synonyms.h"
65 #include "longdouble.h"

67 #include <sys/isa_defs.h>

69 long double
70 tanl(long double x) {
71     long double y[2], z = 0.0L;
72     int n, ix;
73 #if defined(__i386) || defined(__amd64)
74     int *px = (int *) &x;
75 #endif

77     /* trig(Inf or NaN) is NaN */
78     if (!finitel(x))
79         return x - x;

81     /* High word of x. */
82 #if defined(__i386) || defined(__amd64)
83     XTOI(px, ix);
84 #else
85     ix = *(int *) &x;
86 #endif

88     /* |x| ~< pi/4 */
89     ix &= 0x7fffffff;
90     if (ix <= 0x3ffe9220)
91         return __k_tanl(x, z, 0);

93     /* argument reduction needed */
94     else {
95         n = __rem_pio2l(x, y);
96         return __k_tanl(y[0], y[1], n & 1);
97     }
98 }

unchanged_portion_omitted

```

new/usr/src/lib/libm/common/Q/acoshl.c

1

1471 Tue Nov 25 12:57:15 2014

new/usr/src/lib/libm/common/Q/acoshl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __acoshl = acoshl
30 #if defined(ELFOSBJ)
31 #pragma weak acoshl = __acoshl
32 #endif
```

```
32 #include "libm.h"
```

```
34 static const long double
35     zero      = 0.0L,
36     ln2       = 6.931471805599453094172321214581765680755e-0001L,
37     one       = 1.0L,
38     big       = 1.e+20L;
```

```
40 long double
41 acoshl(long double x) {
42     long double t;
43
44     if (isnanl(x))
45         return (x + x);
46     else if (x > big)
47         return (logl(x) + ln2);
48     else if (x > one) {
49         t = sqrtl(x - one);
```

new/usr/src/lib/libm/common/Q/acoshl.c

2

```
50         return (loglpl(t * (t + sqrtl(x + one))));
51     } else if (x == one)
52         return (zero);
53     else
54         return ((x - x) / (x - x));
55 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/acosl.c

1

```
*****
1830 Tue Nov 25 12:57:16 2014
new/usr/src/lib/libm/common/Q/acosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * arccosin function
32  *
33  *
34  * 
$$\text{acos}(x) = 2 \cdot \text{atan2}\left(\frac{\sqrt{1-x}}{\sqrt{1+x}}, 1\right)$$

35  *
36  *
37  *
38  *
39  * 
$$= 2 \cdot \text{atan}\left(\frac{\sqrt{1-x}}{\sqrt{1+x}}\right)$$
 for non-exceptional x.
40  *
41  *
42  * Special cases:
43  *   if x is NaN, return x itself;
44  *   if |x|>1, return NaN with invalid signal.
45  */
46
47 #pragma weak __acosl = acosl
48 #pragma weak acosl = __acosl
49
50 #include "libm.h"
51
52 static const long double zero = 0.0L, one = 1.0L;
53
54 long double
55 acosl(long double x) {
56     if (isnanl(x))
57         return (x + x);
58     else if (fabsl(x) < one)
```

new/usr/src/lib/libm/common/Q/acosl.c

2

```
58         x = atanl(sqrtl((one - x) / (one + x)));
59     else if (x == -one)
60         x = atan2l(one, zero); /* x <- PI */
61     else if (x == one)
62         x = zero;
63     else { /* |x| > 1 create invalid signal */
64         return (zero / zero);
65     }
66     return (x + x);
67 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/asinhl.c

1

1590 Tue Nov 25 12:57:16 2014

new/usr/src/lib/libm/common/Q/asinhl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __asinhl = asinhl
30 #if defined(ELFOSBJ)
31 #pragma weak asinhl = __asinhl
32 #endif
```

```
32 #include "libm.h"
```

```
34 static const long double
35     ln2      = 6.931471805599453094172321214581765680755e-0001L,
36     one      = 1.0L,
37     big      = 1.0e+20L,
38     tiny     = 1.0e-20L;
```

```
40 long double
41 asinhl(long double x) {
42     long double t, w;
43     volatile long double dummy;
```

```
45     w = fabsl(x);
46     if (isnanl(x))
47         return (x + x); /* x is NaN */
48     if (w < tiny) {
49 #ifndef lint
```

new/usr/src/lib/libm/common/Q/asinhl.c

2

```
50         dummy = x + big;          /* inexact if x != 0 */
51 #endif
52         return (x);                /* tiny x */
53     } else if (w < big) {
54         t = one / w;
55         return (copysignl(loglpl(w + w / (t + sqrtl(one + t * t))), x));
56     } else
57         return (copysignl(logl(w) + ln2, x));
58 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/Q/asinl.c

1

```
*****
2010 Tue Nov 25 12:57:16 2014
new/usr/src/lib/libm/common/Q/asinl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __asinl = asinl
30 #if defined(ELFOBJ)
31 #pragma weak asinl = __asinl
32 #endif

32 /*
33  *      asinl(x) = atan2l(x,sqrt(1-x*x));
34  *
35  * For better accuracy, 1-x*x is computed as follows
36  *      1-x*x      if x < 0.5,
37  *      2*(1-|x|)-(1-|x|)*(1-|x|) if x >= 0.5.
38  *
39  * Special cases:
40  *      if x is NaN, return x itself;
41  *      if |x|>1, return NaN with invalid signal.
42  */

44 #include "libm.h"

46 static const long double zero = 0.0L, small = 1.0e-20L, half = 0.5L, one = 1.0L;
47 #ifndef lint
48 static const long double big = 1.0e+20L;
49 #endif
```

new/usr/src/lib/libm/common/Q/asinl.c

2

```
51 long double
52 asinl(long double x) {
53     long double t, w;
54     volatile long double dummy;

56     w = fabsl(x);
57     if (isnanl(x))
58         return (x + x);
59     else if (w <= half) {
60         if (w < small) {
61 #ifndef lint
62             dummy = w + big;
63             /* inexact if w != 0 */
64 #endif
65         } else
66             return (x);
67     } else if (w < one) {
68         return (atanl(x / sqrtl(one - x * x)));
69     } else if (w <= one) {
70         t = one - w;
71         w = t + t;
72         return (atanl(x / sqrtl(w - t * t)));
73     } else if (w == one)
74         return (atan2l(x, zero)); /* asin(+1) = +- PI/2 */
75     else
76         return (zero / zero); /* |x| > 1: invalid */
76 }
_____unchanged_portion_omitted_____
```



```
124         case 2:
125             return (PI + tiny);    /* atan(+...,-INF) */
126         case 3:
127             return (-PI - tiny);   /* atan(-...,-INF) */
128     }
129 }
130
131 /* when y is INF */
132 if (!finitel(y))
133     return (signy == 1 ? -PIo2 - tiny : PIo2 + tiny);
134
135 /* compute y/x */
136 x = fabs1(x);
137 y = fabs1(y);
138 t = PI_lo;
139 k = (ilogbl(y) - ilogbl(x));
140
141 if (k > 120)
142     z = PIo2 + half * t;
143 else if (m > 1 && k < -120)
144     z = zero;
145 else
146     z = atan1(y / x);
147
148 switch (m) {
149 case 0:
150     return (z);    /* atan(+,+) */
151 case 1:
152     return (-z);   /* atan(-,+) */
153 case 2:
154     return (PI - (z - t)); /* atan(+,-) */
155 case 3:
156     return ((z - t) - PI); /* atan(-,-) */
157 }
158 /* NOTREACHED */
159 return 0.0L;
160 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/atan2pil.c

1

1229 Tue Nov 25 12:57:17 2014

new/usr/src/lib/libm/common/Q/atan2pil.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak atan2pil = __atan2pil

30 #include "libm.h"

32 /*
33  * atan2pil(y,x) = atan2l(y, x) / pi
34 */

36 static const long double invpi = 3.183098861837906715377675267450287240689e-1L;

38 long double
39 atan2pil(long double y, long double x) {
40     return (atan2l(y, x) * invpi);
41 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/atanhl.c

1

```
*****
1833 Tue Nov 25 12:57:18 2014
new/usr/src/lib/libm/common/Q/atanhl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __atanhl = atanhl
30 #pragma weak atanhl = __atanhl

32 #include "libm.h"

34 /*
35  *
36  * 
$$\operatorname{atanhl}(x) = \frac{1}{2} * \operatorname{LOG}\left(1 + \frac{2x}{1-x}\right) = 0.5 * \operatorname{loglpl}\left(2 * \frac{x}{1-x}\right)$$

37  *
38  * Note: to guarantee  $\operatorname{atanhl}(-x) = -\operatorname{atanhl}(x)$ , we use
39  *
40  * 
$$\operatorname{atanhl}(x) = \frac{\operatorname{sign}(x)}{2} * \operatorname{loglpl}\left(2 * \frac{|x|}{1-|x|}\right).$$

41  *
42  *
43  * Special cases:
44  *  $\operatorname{atanhl}(x)$  is NaN if  $|x| > 1$  with signal;
45  *  $\operatorname{atanhl}(\text{NaN})$  is that NaN with no signal;
46  *  $\operatorname{atanhl}(\pm 1)$  is  $\pm \text{INF}$  with signal.
47  *
48  */

50 static const long double zero = 0.0L, half = 0.5L, one = 1.0L;

52 long double
53 atanhl(long double x) {
54     long double t;

56     t = fabsl(x);
57     if (t == one)
```

new/usr/src/lib/libm/common/Q/atanhl.c

2

```
58         return (x / zero);
59         t = t / (one - t);
60         return (copysign(half, x) * loglpl(t + t));
61 }
_____unchanged_portion_omitted_____
```



```

124         if (ix < 0x3fc50000) { /* if |x| < 2**(-prec/2-2) */
125             s = one;
126             *(3 - i0 + (int *) &s) = -1; /* s = 1-ulp */
127             *(1 + (int *) &s) = -1;
128             *(2 + (int *) &s) = -1;
129             *(i0 + (int *) &s) -= 1;
130             if ((int) (s * x) < 1)
131                 return (x); /* raise inexact */
132         }
133         z = x * x;
134         if (ix < 0x3fe20000) { /* if |x| < 2**(-prec/4-1) */
135             return (x + (x * z) * p1);
136         } else { /* if |x| < 2**(-prec/6-2) */
137             return (x + (x * z) * (p1 + z * p2));
138         }
139     }
140     z = x * x;
141     return (x + (x * z) * (p1 + z * (p2 + z * (p3 + z * (p4 +
142         z * (p5 + z * (p6 + z * (p7 + z * (p8 + z * (p9 +
143         z * (p10 + z * (p11 + z * (p12 + z * p13))))))))));
144 }

146 /* for |x| >= 8.0 */
147 if (ix >= 0x40020000) {
148     px[i0] = ix;
149     if (ix < 0x40050400) { /* x < 65 */
150         r = one / x;
151         z = r * r;
152         /*
153          * poly1
154          */
155         y = r * (one + z * (p1 + z * (p2 + z * (p3 +
156             z * (p4 + z * (p5 + z * (p6 + z * (p7 +
157             z * (p8 + z * (p9 + z * (p10 + z * (p11 +
158             z * (p12 + z * p13))))))))));
159     } else if (ix < 0x40260000) { /* x < 2**(prec/3+2) */
160         r = one / x;
161         z = r * r;
162         /*
163          * poly2
164          */
165         y = r * (one + z * (q1 + z * (q2 + z * (q3 + z * (q4 +
166             z * (q5 + z * (q6 + z * q7))))));
167         y -= pio2lo;
168     } else if (ix < 0x40720000) { /* x < 2**(prec+2) */
169         y = one / x - pio2lo;
170     } else if (ix < 0x7fff0000) { /* x < inf */
171         y = -pio2lo;
172     } else { /* x is inf or NaN */
173         if (((ix - 0x7fff0000) | px[1] | px[2] | px[i1]) != 0)
174             return (x - x);
175         y = -pio2lo;
176     }
177 }

179 if (sign == 0)
180     return (pio2hi - y);
181 else
182     return (y - pio2hi);
183 }

185 /* now x is between 1/8 and 8 */
186 px[i0] = ix;
187 iy = (ix + 0x00000800) & 0x7ffff000;
188 py[i0] = iy;
189 py[i1] = py[2] = py[i1] = 0;

```

```

190         j = (iy - 0x3ffc0000) >> 12;
192         if (sign == 0)
193             s = (x - y) / (one + x * y);
194         else
195             s = (y - x) / (one + x * y);
196         z = s * s;
197         if (ix == iy)
198             p = s * (one + z * (q1 + z * (q2 + z * (q3 + z * q4)));
199         else
200             p = s * (one + z * (q1 + z * (q2 + z * (q3 + z * (q4 +
201             z * (q5 + z * (q6 + z * q7))))));
202         if (sign == 0) {
203             r = p + _TBL_atan1_lo[j];
204             return (r + _TBL_atan1_hi[j]);
205         } else {
206             r = p - _TBL_atan1_lo[j];
207             return (r - _TBL_atan1_hi[j]);
208         }
209     }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/cbrtl.c

1

```
*****
1680 Tue Nov 25 12:57:18 2014
new/usr/src/lib/libm/common/Q/cbrtl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __cbrtl = cbrtl
30 #pragma weak cbrtl = __cbrtl
31
32 #include "libm.h"
33 #include "longdouble.h"
34
35 #define n0      0
36
37 long double
38 cbrtl(long double x) {
39     long double s, t, r, w, y;
40     double dx, dy;
41     int *py = (int *) &dy;
42     int n, m, m3, sx;
43
44     if (!finitel(x))
45         return (x + x);
46     if (iszerol(x))
47         return (x);
48     sx = signbitl(x);
49     x = fabsl(x);
50     n = ilogbl(x);
51     m = n / 3;
52     m3 = m + m + m;
53     y = scalbnl(x, -m3);
54     dx = (double) y;
55     dy = cbrt(dx);
56     py[1 - n0] += 2;
57     if (py[1 - n0] == 0)
```

new/usr/src/lib/libm/common/Q/cbrtl.c

2

```
58         py[n0] += 1;
59
60     /* one step newton iteration to 113 bits with error < 0.667ulps */
61     t = (long double) dy;
62     t = scalbnl(t, m);
63     s = t * t;
64     r = x / s;
65     w = t + t;
66     r = (r - t) / (w + r);
67     t += t * r;
68
69     return (sx == 0 ? t : -t);
70 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/copysignl.c

1

1237 Tue Nov 25 12:57:19 2014

new/usr/src/lib/libm/common/Q/copysignl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __copysignl = copysignl
30 #if defined(ELFOSBJ)
31 #pragma weak copysignl = __copysignl
32 #endif
```

```
32 #include "libm.h"
```

```
34 long double
35 copysignl(long double x, long double y) {
36     int *px = (int *) &x;
37     int *py = (int *) &y;
38
39     px[HIXWORD] = (px[HIXWORD] & ~XSGNMSK) | (py[HIXWORD] & XSGNMSK);
40     return (x);
41 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/coshl.c

1

```
*****
2933 Tue Nov 25 12:57:20 2014
new/usr/src/lib/libm/common/Q/coshl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __coshl = coshl
30 #pragma weak coshl = __coshl
31
32 #include "libm.h"
33 #include "longdouble.h"
34
35
36 /*
37  * coshl(X)
38  * RETURN THE HYPERBOLIC COSINE OF X
39  *
40  * Method :
41  * 1. Replace x by |x| (coshl(x) = coshl(-x)).
42  * 2.
43  *
44  *      0      <= x <= 0.3465 : coshl(x) := 1 + -----
45  *                                     [ expl(x) - 1 ]^2
46  *                                     2*expl(x)
47  *
48  *      0.3465 <= x <= thresh : coshl(x) := -----
49  *                                     expl(x) + 1/exp(x)
50  *                                     2
51  *      thresh <= x <= lnovft : coshl(x) := expl(x)/2
52  *      lnovft <= x < INF    : coshl(x) := scalbnl(expl(x-1024*ln2),1023)
53  *
54  * here
55  *   thr1      a number that is near one half of ln2.
56  *   thr2      a number such that
57  *              expl(thresh)+expl(-thresh)=expl(thresh)
58  *   lnovft:   logarithm of the overflow threshold
```

new/usr/src/lib/libm/common/Q/coshl.c

2

```
58 *      = MEP1*ln2 chopped to machine precision.
59 *      ME      maximum exponent
60 *      MEP1    maximum exponent plus 1
61 *
62 * Special cases:
63 *      coshl(x) is |x| if x is +INF, -INF, or NaN.
64 *      only coshl(0)=1 is exact for finite x.
65 */
66
67 #define ME      16383
68 #define MEP1    16384
69 #define LNOVFT  1.135652340629414394949193107797076342845e+4L
70 /* last 32 bits of LN2HI is zero */
71 #define LN2HI    6.931471805599453094172319547495844850203e-0001L
72 #define LN2LO    1.667085920830552208890449330400379754169e-0025L
73 #define THR1    0.3465L
74 #define THR2    45.L
75
76 static const long double
77     half      = 0.5L,
78     tiny1     = 7.5e-37L,
79     one       = 1.0L,
80     ln2hi    = LN2HI,
81     ln2lo    = LN2LO,
82     lnovftL  = LNOVFT,
83     thr1     = THR1,
84     thr2     = THR2;
85
86 long double
87 coshl(long double x) {
88     long double t, w;
89
90     w = fabsl(x);
91     if (!finitel(w))
92         return (w + w);          /* x is INF or NaN */
93     if (w < thr1) {
94         t = w < tiny1 ? w : expm1l(w);
95         w = one + t;
96         if (w != one)
97             w = one + (t * t) / (w + w);
98         return (w);
99     } else if (w < thr2) {
100        t = expl(w);
101        return (half * (t + one / t));
102    } else if (w <= lnovftL)
103        return (half * expl(w));
104    else {
105        return (scalbnl(expl((w - MEP1 * ln2hi) - MEP1 * ln2lo), ME));
106    }
107 }
_____unchanged_portion_omitted_____
```

```

*****
2670 Tue Nov 25 12:57:21 2014
new/usr/src/lib/libm/common/Q/cosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 /*
31  * cosl(x)
32  * Table look-up algorithm by K.C. Ng, November, 1989.
33  *
34  * kernel function:
35  *   __k_sinl      ... sin function on [-pi/4,pi/4]
36  *   __k_cosl     ... cos function on [-pi/4,pi/4]
37  *   __rem_pio2l  ... argument reduction routine
38  *
39  * Method.
40  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
41  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
42  *   [-pi/2 , +pi/2], and let n = k mod 4.
43  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
44  *
45  *           n      sin(x)      cos(x)      tan(x)
46  *   -----
47  *           0          S          C          S/C
48  *           1          C         -S         -C/S
49  *           2         -S         -C          S/C
50  *           3         -C          S         -C/S
51  *   -----
52  *
53  * Special cases:
54  *   Let trig be any of sin, cos, or tan.
55  *   trig(+INF) is NaN, with signals;
56  *   trig(NaN)  is that NaN;
57  *
58  * Accuracy:

```

```

59  *      computer TRIG(x) returns trig(x) nearly rounded.
60  */

62 #pragma weak __cosl = cosl
62 #pragma weak cosl = __cosl

64 #include "libm.h"
65 #include "longdouble.h"

67 long double
68 cosl(long double x) {
69     long double y[2], z = 0.0L;
70     int n, ix;

72     ix = *(int *) &x;                /* High word of x */

74     ix &= 0x7fffffff;
75     if (ix <= 0x3ffe9220)            /* |x| ~< pi/4 */
76         return (__k_cosl(x, z));
77     else if (ix >= 0x7fff0000)       /* trig(Inf or NaN) is NaN */
78         return (x - x);
79     else {                            /* argument reduction needed */
80         n = __rem_pio2l(x, y);
81         switch (n & 3) {
82         case 0:
83             return (__k_cosl(y[0], y[1]));
84         case 1:
85             return (-__k_sinl(y[0], y[1]));
86         case 2:
87             return (-__k_cosl(y[0], y[1]));
88         case 3:
89             return (__k_sinl(y[0], y[1]));
90         }
91     }
92     /* NOTREACHED */
93     return 0.0L;
94 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/Q/erfl.c

1

```
*****
13609 Tue Nov 25 12:57:21 2014
new/usr/src/lib/libm/common/Q/erfl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * long double function erf,erfc (long double x)
32  * K.C. Ng, September, 1989.
33  *
34  *
35  *      2
36  *      erf(x) = ----- \int_0^x exp(-t*t)dt
37  *      sqrt(pi)
38  *
39  *
40  *      erfc(x) = 1-erf(x)
41  *
42  * method:
43  * Since erf(-x) = -erf(x), we assume x>=0.
44  * For x near 0, we have the expansion
45  *
46  *      erf(x) = (2/sqrt(pi))*(x - x^3/3 + x^5/10 - x^7/42 + ....).
47  *
48  * Since 2/sqrt(pi) = 1.128379167095512573896158903121545171688,
49  * we use x + x*P(x^2) to approximate erf(x). This formula will
50  * guarantee the error less than one ulp where x is not too far
51  * away from 0. We note that erf(x)=x at x = 0.6174..... After
52  * some experiment, we choose the following approximation on
53  * interval [0,0.84375].
54  *
55  * For x in [0,0.84375]
56  *
57  *      P = P(x) = (p0 + p1 * x + p2 * x^2 + ... + p20 * x^40)
58  *
59  *      erf(x) = x + x*P
```

new/usr/src/lib/libm/common/Q/erfl.c

2

```
59 *      erfc(x) = 1 - erf(x) if x<=0.25
60 *      = 0.5 + ((0.5-x)-x*P) if x in [0.25,0.84375]
61 * precision: |P(x^2)-(erf(x)-x)/x| <= 2**-122.50
62 *
63 * For x in [0.84375,1.25], let s = x - 1, and
64 * c = 0.84506291151 rounded to single (24 bits)
65 * erf(x) = c + P1(s)/Q1(s)
66 * erfc(x) = (1-c) - P1(s)/Q1(s)
67 * precision: |P1/Q1 - (erf(x)-c)| <= 2**-118.41
68 *
69 *
70 * For x in [1.25,1.75], let s = x - 1.5, and
71 * c = 0.95478588343 rounded to single (24 bits)
72 * erf(x) = c + P2(s)/Q2(s)
73 * erfc(x) = (1-c) - P2(s)/Q2(s)
74 * precision: |P1/Q1 - (erf(x)-c)| <= 2**-123.83
75 *
76 *
77 * For x in [1.75,16/3]
78 * erf(x) = exp(-x*x)*(1/x)*R1(1/x)/S1(1/x)
79 * erf(x) = 1 - erfc(x)
80 * precision: absolute error of R1/S1 is bounded by 2**-124.03
81 *
82 * For x in [16/3,107]
83 * erf(x) = exp(-x*x)*(1/x)*R2(1/x)/S2(1/x)
84 * erf(x) = 1 - erfc(x) (if x>=9 simple return erf(x)=1 with inexact)
85 * precision: absolute error of R2/S2 is bounded by 2**-120.07
86 *
87 * Else if inf > x >= 107
88 * erf(x) = 1 with inexact
89 * erfc(x) = 0 with underflow
90 *
91 * Special case:
92 * erf(inf) = 1
93 * erfc(inf) = 0
94 */
95
96 #pragma weak __erfl = erfl
97 #pragma weak __erfc1 = erfc1
98 #pragma weak erfl = __erfl
99 #pragma weak erfc1 = __erfc1
100
101
102 #include "libm.h"
103 #include "longdouble.h"
104
105 static const long double
106 tiny = 1e-40L,
107 nearunfl = 1e-4000L,
108 half = 0.5L,
109 one = 1.0L,
110 onehalf = 1.5L,
111 Ll6_3 = 16.0L/3.0L;
112
113 /*
114  * Coefficients for even polynomial P for erf(x)=x+x*P(x^2) on [0,0.84375]
115 */
116 static const long double P[] = { /* 21 coeffs */
117 1.283791670955125738961589031215451715556e-0001L,
118 -3.761263890318375246320529677071815594603e-0001L,
119 1.128379167095512573896158903121205899135e-0001L,
120 -2.686617064513125175943235483344625046092e-0002L,
121 5.223977625442187842111846652980454568389e-0003L,
122 -8.548327023450852832546626271083862724358e-0004L,
123 1.205533298178966425102164715902231976672e-0004L,
124 -1.492565035840625097674944905027897838996e-0005L,
125 1.646211436588924733604648849172936692024e-0006L,
126 -1.636584469123491976815834704799733514987e-0007L,
```

new/usr/src/lib/libm/common/Q/erfl.c

3

```
123 1.480719281587897445302529007144770739305e-0008L,  
124 -1.229055530170782843046467986464722047175e-0009L,  
125 9.422759064320307357553954945760654341633e-0011L,  
126 -6.711366846653439036162105104991433380926e-0012L,  
127 4.463224090341893165100275380693843116240e-0013L,  
128 -2.783513452582658245422635662559779162312e-0014L,  
129 1.634227412586960195251346878863754661546e-0015L,  
130 -9.060782672889577722765711455623117802795e-0017L,  
131 4.741341801266246873412159213893613602354e-0018L,  
132 -2.272417596497826188374846636534317381203e-0019L,  
133 8.069088733716068462496835658928566920933e-0021L,  
134 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/exp101.c

1

```
*****
2846 Tue Nov 25 12:57:22 2014
new/usr/src/lib/libm/common/Q/exp101.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak exp101 = __exp101
30 #include "libm.h"
31 #include "longdouble.h"
33 /*
34  * exp101(x)
35  *   n = nint(x*(log10/log2)) ;
36  *   exp10(x) = 10**x = exp(x*ln(10)) = exp(n*ln2+(x*ln10-n*ln2))
37  *             = 2**n*exp(ln10*(x-n*log2/log10))
38  *   If x is an integer <= M then use repeat multiplication. For
39  *   10**M is the largest representable integer, where
40  *       M = 10      single precision (24 bits)
41  *       M = 22      double precision (53 bits)
42  *       M = 48      quadruple precision (113 bits)
43  */
45 #define TINY 1.0e-20L /* single: 1e-5, double: 1e-10, quad: 1e-20 */
46 #define LG10OVT 4933.L /* single: 39, double: 309, quad: 4933 */
47 #define LG10UFT -4966.L /* single: -45, double: -323, quad: -4966 */
48 #define M 48
49 /* logt2hi : last 32 bits is zero for quad prec */
50 #define LOGT2HI 0.30102999566398119521373889472420986034688L
51 #define LOGT2LO 2.831664213089468167896664371953e-31L
53 static const long double
54 zero = 0.0L,
55 tiny = TINY * TINY,
56 one = 1.0L,
```

new/usr/src/lib/libm/common/Q/exp101.c

2

```
57 lg10 = 3.321928094887362347870319429489390175865e+0000L,
58 ln10 = 2.302585092994045684017991454684364207601e+0000L,
59 logt2hi = LOGT2HI,
60 logt2lo = LOGT2LO,
61 lg10ovt = LG10OVT,
62 lg10uft = LG10UFT;
64 long double
65 exp101(long double x) {
66     long double t, temp;
67     int k;
69     if (!finitel(x)) {
70         if (isnanl(x) || x > zero)
71             return (x + x);
72         else
73             return (zero);
74     }
75     if (fabsl(x) < tiny)
76         return (one + x);
77     if (x <= lg10ovt)
78         if (x >= lg10uft) {
79             k = (int) x;
80             temp = 10.0L;
81             /* x is a small +integer */
82             if (0 <= k && k <= M && (long double) k == x) {
83                 t = one;
84                 if (k & 1)
85                     t *= temp;
86                 k >>= 1;
87                 while (k) {
88                     temp *= temp;
89                     if (k & 1)
90                         t *= temp;
91                     k >>= 1;
92                 }
93                 return (t);
94             }
95             t = anintl(x * lg10);
96             return (scalbnl(expl(ln10 * ((x - t * logt2hi) -
97                 t * logt2lo)), (int) t));
98         } else
99             return (scalbnl(one, -50000)); /* underflow */
100     else
101         return (scalbnl(one, 50000)); /* overflow */
102 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/Q/exp21.c

1

```
*****
2095 Tue Nov 25 12:57:22 2014
new/usr/src/lib/libm/common/Q/exp21.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __exp21 = exp21
31 #pragma weak exp21 = __exp21
32
33 #include "libm.h"
34 #include "longdouble.h"
35
36 /*
37  * exp21(x) = 2**x = 2**((x-anint(x))+anint(x))
38  *          = 2**anint(x)*2**(x-anint(x))
39  *          = 2**anint(x)*exp((x-anint(x))*ln2)
40 */
41 #define TINY 1.0e-20L /* single: 1e-5, double: 1e-10, quad: 1e-20 */
42 #define OVFLXP 16400 /* single: 130, double 1030, quad: 16400 */
43 #define UNFLXP -16520 /* single:-155, double -1080, quad:-16520 */
44
45 static const long double
46 zero = 0.0L,
47 tiny = TINY * TINY,
48 half = 0.5L,
49 ln2 = 6.931471805599453094172321214581765680755e-0001L,
50 one = 1.0L;
51
52 static const int
53 ovflexp = OVFLXP,
54 unflexp = UNFLXP;
55
56 long double
57 exp21(long double x) {
```

new/usr/src/lib/libm/common/Q/exp21.c

2

```
58 long double t;
59
60 if (!finitel(x)) {
61     if (isnanl(x) || x > zero)
62         return (x + x);
63     else
64         return (zero);
65 }
66 t = fabsl(x);
67 if (t < half) {
68     if (t < tiny)
69         return (one + x);
70     else
71         return (expl(ln2 * x));
72 }
73 t = anintl(x);
74 if (t < ovflexp) {
75     if (t >= unflexp)
76         return (scalbnl(expl(ln2 * (x - t)), (int) t));
77     else
78         return (scalbnl(one, unflexp)); /* underflow */
79 } else
80     return (scalbnl(one, ovflexp)); /* overflow */
81 }
82
83 unchanged_portion_omitted
```


new/usr/src/lib/libm/common/Q/expl.c

3

```
124         _TBL_expl_lo[j]);  
125     return (scalbnl(x, m));  
126 }  
_____unchanged_portion_omitted_
```


new/usr/src/lib/libm/common/Q/fabsl.c

1

1142 Tue Nov 25 12:57:23 2014

new/usr/src/lib/libm/common/Q/fabsl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __fabsl = fabsl
30 #pragma weak fabsl = __fabsl
```

```
32 #include "libm.h"
```

```
34 long double
35 fabsl(long double x) {
36     int *px = (int *) &x;
37
38     px[0] &= 0x7fffffff;
39     return (x);
40 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/Q/finitel.c

1

1412 Tue Nov 25 12:57:24 2014

new/usr/src/lib/libm/common/Q/finitel.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
31 #pragma weak finitel = __finitel
32 #endif
```

```
30 #include "libm.h"
```

```
32 #if defined(__sparc)
33 int
34 finitel(long double x) {
35     int *px = (int *) &x;
36     return ((px[0] & ~0x80000000) < 0x7fff0000);
37 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/Q/floorl.c

1

1849 Tue Nov 25 12:57:24 2014

new/usr/src/lib/libm/common/Q/floorl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 /*
31  * ceil(x)    return the biggest integral value below x
32  * floor(x)   return the least integral value above x
33  *
34  * NOTE: aintl(x), anintl(x), ceil(x), floor(x), and rintl(x) return result
35  * with the same sign as x's, including 0.0.
36 */

38 #pragma weak __ceil = ceil
39 #pragma weak __floorl = floorl
38 #pragma weak ceil = __ceil
39 #pragma weak floorl = __floorl

41 #include "libm.h"
42 #include "longdouble.h"

44 static const long double gone = 1.0L;

46 long double
47 ceil(long double x) {
48     long double t;

50     if (!finitel(x))
51         return (x + x);
52     t = rintl(x);
53     if (t >= x)                /* already ceil(x) */
54         return (t);
55     else                        /* t < x case: return t+1 */
56         return (copysignl(t + gone, x));
```

new/usr/src/lib/libm/common/Q/floorl.c

2

57 }

unchanged_portion_omitted

```

*****
5771 Tue Nov 25 12:57:25 2014
new/usr/src/lib/libm/common/Q/fmodl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __fmodl = fmodl
30 #pragma weak fmodl = __fmodl

32 #include "libm.h"

34 static const int
35     is = -0x7fffffff - 1,
36     im = 0x0000ffff,
37     iu = 0x00010000;

39 static const long double
40     zero = 0.0L,
41     one = 1.0L;

43 #ifdef __LITTLE_ENDIAN
44 #define __H0(x) *(3 + (int *) &x)
45 #define __H1(x) *(2 + (int *) &x)
46 #define __H2(x) *(1 + (int *) &x)
47 #define __H3(x) *(0 + (int *) &x)
48 #else
49 #define __H0(x) *(0 + (int *) &x)
50 #define __H1(x) *(1 + (int *) &x)
51 #define __H2(x) *(2 + (int *) &x)
52 #define __H3(x) *(3 + (int *) &x)
53 #endif

55 long double
56 fmodl(long double x, long double y) {
57     long double a, b;

```

```

58     int n, ix, iy, k, sx;
59     int hx;
60     int x0, y0, z0, carry;
61     unsigned xl, x2, x3, yl, y2, y3, z1, z2, z3;

63     hx = __H0(x);
64     x1 = __H1(x);
65     x2 = __H2(x);
66     x3 = __H3(x);
67     y0 = __H0(y);
68     y1 = __H1(y);
69     y2 = __H2(y);
70     y3 = __H3(y);

72     sx = hx & 0x80000000;
73     x0 = hx ^ sx;
74     y0 &= 0x7fffffff;

76     /* purge off exception values */
77     if (x0 >= 0x7fff0000 || /* !finitel(x) */
78         (y0 > 0x7fff0000) || (y0 == 0x7fff0000 && ((y1 | y2 | y3) != 0)) ||
79         (y0 | y1 | y2 | y3) == 0) /* isnanl(y) || y = 0 */
80         return ((x * y) / (x * y));
81     a = fabsl(x);
82     b = fabsl(y);
83     if (a <= b) {
84         if (a < b)
85             return (x);
86         else
87             return (zero * x);
88     }
89     /* determine ix = ilogbl(x) */
90     if (x0 < iu) { /* subnormal x */
91         ix = -16382;
92         while (x0 == 0) {
93             ix -= 16;
94             x0 = x1 >> 16;
95             x1 = (x1 << 16) | (x2 >> 16);
96             x2 = (x2 << 16) | (x3 >> 16);
97             x3 = (x3 << 16);
98         }
99         while (x0 < iu) {
100             ix -= 1;
101             x0 = (x0 << 1) | (x1 >> 31);
102             x1 = (x1 << 1) | (x2 >> 31);
103             x2 = (x2 << 1) | (x3 >> 31);
104             x3 <<= 1;
105         }
106     } else {
107         ix = (x0 >> 16) - 16383;
108         x0 = iu | (x0 & im);
109     }

111     /* determine iy = ilogbl(y) */
112     if (y0 < iu) { /* subnormal y */
113         iy = -16382;
114         while (y0 == 0) {
115             iy -= 16;
116             y0 = y1 >> 16;
117             y1 = (y1 << 16) | (y2 >> 16);
118             y2 = (y2 << 16) | (y3 >> 16);
119             y3 = (y3 << 16);
120         }
121         while (y0 < iu) {
122             iy -= 1;
123             y0 = (y0 << 1) | (y1 >> 31);

```

```

124         y1 = (y1 << 1) | (y2 >> 31);
125         y2 = (y2 << 1) | (y3 >> 31);
126         y3 <<= 1;
127     }
128 } else {
129     iy = (y0 >> 16) - 16383;
130     y0 = iu | (y0 & im);
131 }
132
133 /* fix point fmod */
134 n = ix - iy;
135 while (n-- > 0) {
136     while (x0 == 0 && n >= 16) {
137         n -= 16;
138         x0 = x1 >> 16;
139         x1 = (x1 << 16) | (x2 >> 16);
140         x2 = (x2 << 16) | (x3 >> 16);
141         x3 = (x3 << 16);
142     }
143     while (x0 < iu && n >= 1) {
144         n -= 1;
145         x0 = (x0 << 1) | (x1 >> 31);
146         x1 = (x1 << 1) | (x2 >> 31);
147         x2 = (x2 << 1) | (x3 >> 31);
148         x3 = (x3 << 1);
149     }
150     carry = 0;
151     z3 = x3 - y3;
152     carry = (z3 > x3);
153     if (carry == 0) {
154         z2 = x2 - y2;
155         carry = (z2 > x2);
156     } else {
157         z2 = x2 - y2 - 1;
158         carry = (z2 >= x2);
159     }
160     if (carry == 0) {
161         z1 = x1 - y1;
162         carry = (z1 > x1);
163     } else {
164         z1 = x1 - y1 - 1;
165         carry = (z1 >= x1);
166     }
167     z0 = x0 - y0 - carry;
168     if (z0 < 0) { /* double x */
169         x0 = x0 + x0 + ((x1 & is) != 0);
170         x1 = x1 + x1 + ((x2 & is) != 0);
171         x2 = x2 + x2 + ((x3 & is) != 0);
172         x3 = x3 + x3;
173     } else {
174         if (z0 == 0) {
175             if ((z1 | z2 | z3) == 0) { /* 0: done */
176                 __H0(a) = hx & is;
177                 __H1(a) = __H2(a) = __H3(a) = 0;
178                 return (a);
179             }
180             /* x = z << 1 */
181             z0 = z0 + z0 + ((z1 & is) != 0);
182             z1 = z1 + z1 + ((z2 & is) != 0);
183             z2 = z2 + z2 + ((z3 & is) != 0);
184             z3 = z3 + z3;
185             x0 = z0;
186             x1 = z1;
187             x2 = z2;
188             x3 = z3;
189

```

```

190     }
191 }
192
193 carry = 0;
194 z3 = x3 - y3;
195 carry = (z3 > x3);
196 if (carry == 0) {
197     z2 = x2 - y2;
198     carry = (z2 > x2);
199 } else {
200     z2 = x2 - y2 - 1;
201     carry = (z2 >= x2);
202 }
203 if (carry == 0) {
204     z1 = x1 - y1;
205     carry = (z1 > x1);
206 } else {
207     z1 = x1 - y1 - 1;
208     carry = (z1 >= x1);
209 }
210 z0 = x0 - y0 - carry;
211 if (z0 >= 0) {
212     x0 = z0;
213     x1 = z1;
214     x2 = z2;
215     x3 = z3;
216 }
217 /* convert back to floating value and restore the sign */
218 if ((x0 | x1 | x2 | x3) == 0) {
219     __H0(a) = hx & is;
220     __H1(a) = __H2(a) = __H3(a) = 0;
221     return (a);
222 }
223 while (x0 < iu) {
224     if (x0 == 0) {
225         iy -= 16;
226         x0 = x1 >> 16;
227         x1 = (x1 << 16) | (x2 >> 16);
228         x2 = (x2 << 16) | (x3 >> 16);
229         x3 = (x3 << 16);
230     } else {
231         x0 = x0 + x0 + ((x1 & is) != 0);
232         x1 = x1 + x1 + ((x2 & is) != 0);
233         x2 = x2 + x2 + ((x3 & is) != 0);
234         x3 = x3 + x3;
235         iy -= 1;
236     }
237 }
238
239 /* normalize output */
240 if (iy >= -16382) {
241     __H0(a) = sx | (x0 - iu) | ((iy + 16383) << 16);
242     __H1(a) = x1;
243     __H2(a) = x2;
244     __H3(a) = x3;
245 } else { /* subnormal output */
246     n = -16382 - iy;
247     k = n & 31;
248     if (k != 0) {
249         if (k <= 16) {
250             x3 = (x2 << (32 - k)) | (x3 >> k);
251             x2 = (x1 << (32 - k)) | (x2 >> k);
252             x1 = (x0 << (32 - k)) | (x1 >> k);
253             x0 >>= k;
254         } else {
255             x3 = (x2 << (32 - k)) | (x3 >> k);

```

```
256         x2 = (x1 << (32 - k)) | (x2 >> k);
257         x1 = (x0 << (32 - k)) | (x1 >> k);
258         x0 = 0;
259     }
260 }
261 while (n >= 32) {
262     n -= 32;
263     x3 = x2;
264     x2 = x1;
265     x1 = x0;
266     x0 = 0;
267 }
268 __H0(a) = x0 | sx;
269 __H1(a) = x1;
270 __H2(a) = x2;
271 __H3(a) = x3;
272 a *= one;
273 }
274 return (a);
275 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/Q/gammal.c

1

1321 Tue Nov 25 12:57:25 2014

new/usr/src/lib/libm/common/Q/gammal.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __gammal = gammal
30 #pragma weak gammal = __gammal
31
32 /*
33  * long double gammal(long double x);
34 */
35
36 #include "libm.h"
37 #include "longdouble.h"
38
39 extern int signgam;
40 extern int signgaml;
41
42 long double
43 gammal(long double x) {
44     long double y = __k_lgammal(x, &signgaml);
45
46     signgam = signgaml; /* SUSv3 requires the setting of signgam */
47     return (y);
48 }
49
50 unchanged portion omitted
```

new/usr/src/lib/libm/common/Q/gammal_r.c

1

1233 Tue Nov 25 12:57:25 2014

new/usr/src/lib/libm/common/Q/gammal_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 /*
31  * long double gammal_r(long double x, int *signgamlp);
32 */
```

```
34 #pragma weak __gammal_r = gammal_r
34 #pragma weak gammal_r = __gammal_r
```

```
36 #include "libm.h"
37 #include "longdouble.h"
```

```
39 long double
40 gammal_r(long double x, int *signgamlp) {
41     return (__k_lgammal(x, signgamlp));
42 }
```

unchanged_portion_omitted

```

*****
3902 Tue Nov 25 12:57:26 2014
new/usr/src/lib/libm/common/Q/hypot1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __hypot1 = hypot1
30 #pragma weak hypot1 = __hypot1

32 /*
33  * long double hypot1(long double x, long double y);
34  * Method :
35  *   If z=x*x+y*y has error less than sqrt(2)/2 ulp than sqrt(z) has
36  *   error less than 1 ulp.
37  *   So, compute sqrt(x*x+y*y) with some care as follows:
38  *   Assume x>y>0;
39  *   1. save and set rounding to round-to-nearest
40  *   2. if x > 2y use
41  *       x1*x1+(y*y+(x2*(x+x2))) for x*x+y*y
42  *   where x1 = x with lower 64 bits cleared, x2 = x-x1; else
43  *   3. if x <= 2y use
44  *       t1*y1+((x-y)*(x-y)+(t1*y2+t2*y))
45  *   where t1 = 2x with lower 64 bits cleared, t2 = 2x-t1, y1 = y with
46  *   lower 64 bits chopped, y2 = y-y1.
47  *
48  *   NOTE: DO NOT remove parenthesis!
49  *
50  * Special cases:
51  *   hypot(x,y) is INF if x or y is +INF or -INF; else
52  *   hypot(x,y) is NAN if x or y is NAN.
53  *
54  * Accuracy:
55  *   hypot(x,y) returns sqrt(x^2+y^2) with error less than 1 ulps (units
56  *   in the last place)
57 */

```

```

59 #include "libm.h"
60 #include "longdouble.h"

62 extern enum fp_direction_type __swapRD(enum fp_direction_type);

64 static const long double zero = 0.0L, one = 1.0L;

66 long double
67 hypot1(long double x, long double y) {
68     int n0, n1, n2, n3;
69     long double t1, t2, y1, y2, w;
70     int *px = (int *) &x, *py = (int *) &y;
71     int *pt1 = (int *) &t1, *py1 = (int *) &y1;
72     enum fp_direction_type rd;
73     int j, k, nx, ny, nz;

75     if ((* (int *) &one) != 0) { /* determine word ordering */
76         n0 = 0;
77         n1 = 1;
78         n2 = 2;
79         n3 = 3;
80     } else {
81         n0 = 3;
82         n1 = 2;
83         n2 = 1;
84         n3 = 0;
85     }

87     px[n0] &= 0x7fffffff; /* clear sign bit of x and y */
88     py[n0] &= 0x7fffffff;
89     k = 0x7fff0000;
90     nx = px[n0] & k; /* exponent of x and y */
91     ny = py[n0] & k;
92     if (ny > nx) {
93         w = x;
94         x = y;
95         y = w;
96         nz = ny;
97         ny = nx;
98         nx = nz;
99     } /* force x > y */
100     if ((nx - ny) >= 0x00730000)
101         return (x + y); /* x/y >= 2**116 */
102     if (nx < 0x5ff30000 && ny > 0x205b0000) { /* medium x,y */
103         /* save and set RD to Rounding to nearest */
104         rd = __swapRD(fp_nearest);
105         w = x - y;
106         if (w > y) {
107             pt1[n0] = px[n0];
108             pt1[n1] = px[n1];
109             pt1[n2] = pt1[n3] = 0;
110             t2 = x - t1;
111             x = sqrtl(t1 * t1 - (y * (-y) - t2 * (x + t1)));
112         } else {
113             x = x + x;
114             py1[n0] = py[n0];
115             py1[n1] = py[n1];
116             py1[n2] = py1[n3] = 0;
117             y2 = y - y1;
118             pt1[n0] = px[n0];
119             pt1[n1] = px[n1];
120             pt1[n2] = pt1[n3] = 0;
121             t2 = x - t1;
122             x = sqrtl(t1 * y1 - (w * (-w) - (t2 * y1 + y2 * x)));
123         }

```

```
124         if (rd != fp_nearest)
125             (void) __swapRD(rd); /* restore rounding mode */
126         return (x);
127     } else {
128         if (nx == k || ny == k) { /* x or y is INF or NaN */
129             if (isinfl(x))
130                 t2 = x;
131             else if (isinfl(y))
132                 t2 = y;
133             else
134                 t2 = x + y; /* invalid if x or y is sNaN */
135             return (t2);
136         }
137         if (ny == 0) {
138             if (y == zero || x == zero)
139                 return (x + y);
140             t1 = scalbnl(one, 16381);
141             x *= t1;
142             y *= t1;
143             return (scalbnl(one, -16381) * hypot1(x, y));
144         }
145         j = nx - 0x3fff0000;
146         px[n0] -= j;
147         py[n0] -= j;
148         pt1[n0] = nx;
149         pt1[n1] = pt1[n2] = pt1[n3] = 0;
150         return (t1 * hypot1(x, y));
151     }
152 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/Q/ieee_func1.c

1

2552 Tue Nov 25 12:57:26 2014

new/usr/src/lib/libm/common/Q/ieee_func1.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELF_OBJ)
31 #pragma weak isinfl = __isinfl
32 #pragma weak isnormall = __isnormall
33 #pragma weak issubnormall = __issubnormall
34 #pragma weak iszerol = __iszerol
35 #pragma weak signbitl = __signbitl
36 #endif
```

```
30 #include "libm.h"
```

```
32 #if defined(__sparc)
33 int
34 isinfl(long double x) {
35     int *px = (int *) &x;
36     return ((px[0] & ~0x80000000) == 0x7fff0000 && px[1] == 0 &&
37            px[2] == 0 && px[3] == 0);
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/ilogbl.c

1

```
*****
2452 Tue Nov 25 12:57:27 2014
new/usr/src/lib/libm/common/Q/ilogbl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __ilogbl = ilogbl
30 #if defined(ELFOBJ)
31 #pragma weak ilogbl = __ilogbl
32 #endif

32 #include "libm.h"
33 #include "xpg6.h" /* __xpg6 */

35 #if defined(__sparc)
36 #define ISNORMALL(k, x) (k != 0x7fff) /* assuming k != 0 */
37 #define X86PDNRM(k, x)
38 #define XSCALE_OFFSET 0x406f /* 0x3fff + 112 */
39 static const long double xscale = 5192296858534827628530496329220096.0L;
40 /* 2^112 */
41 #elif defined(__x86)
42 /*
43  * if pseudo-denormal, replace by the equivalent normal
44  */
45 #define X86PDNRM(k, x) if (k == 0 && (((int *) &x)[1] & 0x80000000) != 0) \
46 ((int *) &x)[2] |= k = 1
47 #if defined(HANDLE_UNSUPPORTED)
48 #define ISNORMALL(k, x) (k != 0x7fff && (((int *) &x)[1] & 0x80000000) != 0)
49 #else
```

new/usr/src/lib/libm/common/Q/ilogbl.c

2

```
50 #define ISNORMALL(k, x) (k != 0x7fff)
51 #endif
52 #define XSCALE_OFFSET 0x403e /* 0x3fff + 63 */
53 static const long double xscale = 9223372036854775808.0L; /* 2^63 */
54 #endif

56 static int
57 raise_invalid(int v) { /* SUSv3 requires ilogbl(0,+/-Inf,NaN) raise invalid */
58 #ifndef lint
59     if ((__xpg6 & _C99SUSv3_ilogb_0InfNaN_raises_invalid) != 0) {
60         static const double zero = 0.0;
61         volatile double dummy;

63         dummy = zero / zero;
64     }
65 #endif
66     return (v);
67 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/isnanl.c

1

1581 Tue Nov 25 12:57:27 2014

new/usr/src/lib/libm/common/Q/isnanl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __isnanl = isnanl
30 #if defined(ELFOSBJ)
31 #pragma weak isnanl = __isnanl
32 #endif
```

```
32 #include "libm.h"
```

```
34 #if defined(__sparc)
35 int
36 isnanl(long double x) {
37     int *px = (int *) &x;
38     return ((px[0] & ~0x80000000) >= 0x7fff0000 &&
39             ((px[0] & ~0xffff0000) | px[1] | px[2] | px[3]) != 0);
40 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/jnl.c

1

```
*****
7136 Tue Nov 25 12:57:28 2014
new/usr/src/lib/libm/common/Q/jnl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __jnl = jnl
31 #pragma weak __ynl = ynl
32 #if defined(ELFOSBJ)
33 #pragma weak jnl = __jnl
34 #pragma weak ynl = __ynl
35 #endif

33 /*
34  * floating point Bessel's function of the 1st and 2nd kind
35  * of order n: jn(n,x), yn(n,x);
36  *
37  * Special cases:
38  *   y0(0)=y1(0)=yn(n,0) = -inf with division by zero signal;
39  *   y0(-ve)=y1(-ve)=yn(n,-ve) are NaN with invalid signal.
40  * Note 2. About jn(n,x), yn(n,x)
41  *   For n=0, j0(x) is called,
42  *   for n=1, j1(x) is called,
43  *   for n>x, forward recursion us used starting
44  *   from values of j0(x) and j1(x).
45  *   for n>x, a continued fraction approximation to
46  *   j(n,x)/j(n-1,x) is evaluated and then backward
47  *   recursion is used starting from a supposed value
48  *   for j(n,x). The resulting value of j(0,x) is
```

new/usr/src/lib/libm/common/Q/jnl.c

2

```
49 * compared with the actual value to correct the
50 * supposed value of j(n,x).
51 *
52 * yn(n,x) is similar in all respects, except
53 * that forward recursion is used for all
54 * values of n>1.
55 *
56 */

58 #include "libm.h"
59 #include "longdouble.h"
60 #include <float.h> /* LDBL_MAX */

62 #define GENERIC long double

64 static const GENERIC
65 invsqrtpi = 5.641895835477562869480794515607725858441e-0001L,
66 two = 2.0L,
67 zero = 0.0L,
68 one = 1.0L;

70 GENERIC
71 jnl(n, x) int n; GENERIC x; {
72     int i, sgn;
73     GENERIC a, b, temp, z, w;

75     /*
76      * J(-n,x) = (-1)^n * J(n, x), J(n, -x) = (-1)^n * J(n, x)
77      * Thus, J(-n,x) = J(n,-x)
78      */
79     if (n < 0) {
80         n = -n;
81         x = -x;
82     }
83     if (n == 0)
84         return (j0l(x));
85     if (n == 1)
86         return (j1l(x));
87     if (x != x)
88         return (x+x);
89     if ((n&1) == 0)
90         sgn = 0; /* even n */
91     else
92         sgn = signbitl(x); /* odd n */
93     x = fabsl(x);
94     if (x == zero || !finitel(x)) b = zero;
95     else if ((GENERIC)n <= x) {
96         /*
97          * Safe to use
98          * J(n+1,x)=2n/x *J(n,x)-J(n-1,x)
99          */
100         if (x > 1.0e91L) {
101             /*
102              * x >> n**2
103              * Jn(x) = cos(x-(2n+1)*pi/4)*sqrt(2/x*pi)
104              * Yn(x) = sin(x-(2n+1)*pi/4)*sqrt(2/x*pi)
105              * Let s=sin(x), c=cos(x),
106              * xn=x-(2n+1)*pi/4, sqrt2 = sqrt(2), then
107              *
108              *          n      sin(xn)*sqrt2      cos(xn)*sqrt2
109              *          -----
110              *          0      s-c                  c+s
111              *          1      -s-c                 -c+s
112              *          2      -s+c                 -c-s
113              *          3      s+c                  c-s
114              */

```

```

115     switch (n&3) {
116         case 0: temp = cosl(x)+sinl(x); break;
117         case 1: temp = -cosl(x)+sinl(x); break;
118         case 2: temp = -cosl(x)-sinl(x); break;
119         case 3: temp = cosl(x)-sinl(x); break;
120     }
121     b = invsqrtpi*temp/sqrtl(x);
122 } else {
123     a = j0l(x);
124     b = j1l(x);
125     for (i = 1; i < n; i++) {
126         temp = b;
127         b = b*((GENERIC)(i+i)/x) - a; /* avoid underflow */
128         a = temp;
129     }
130 }
131 } else {
132     if (x < 1e-17L) { /* use J(n,x) = 1/n!*(x/2)^n */
133         b = powl(0.5L*x, (GENERIC)n);
134         if (b != zero) {
135             for (a = one, i = 1; i <= n; i++) a *= (GENERIC)i;
136             b = b/a;
137         }
138     } else {
139         /* use backward recurrence */
140         /*
141          *
142          *  $J(n,x)/J(n-1,x) = \frac{x}{2n} - \frac{x^2}{2(n+1)} - \frac{x^2}{2(n+2)} \dots$ 
143          *
144          *
145          * (for large x) =  $\frac{1}{2n} - \frac{1}{2(n+1)} - \frac{1}{2(n+2)} \dots$ 
146          *
147          *
148          *  $\frac{1}{x} - \frac{1}{x} - \frac{1}{x}$ 
149          *
150          *
151          * Let w = 2n/x and h=2/x, then the above quotient
152          * is equal to the continued fraction:
153          *
154          * 
$$= \frac{1}{w - \frac{1}{w+h - \frac{1}{w+2h - \dots}}}$$

155          *
156          *
157          *
158          *
159          *
160          *
161          * To determine how many terms needed, let
162          * Q(0) = w, Q(1) = w(w+h) - 1,
163          * Q(k) = (w+k*h)*Q(k-1) - Q(k-2),
164          * When Q(k) > 1e4 good for single
165          * When Q(k) > 1e9 good for double
166          * When Q(k) > 1e17 good for quaduple
167          */
168     /* determin k */
169     GENERIC t, v;
170     double q0, q1, h, tmp; int k, m;
171     w = (n+n)/(double)x; h = 2.0/(double)x;
172     q0 = w; z = w+h; q1 = w*z - 1.0; k = 1;
173     while (q1 < 1.0e17) {
174         k += 1; z += h;
175         tmp = z*q1 - q0;
176         q0 = q1;
177         q1 = tmp;
178     }
179     m = n+n;
180     for (t = zero, i = 2*(n+k); i >= m; i -= 2) t = one/(i/x-t);

```

```

181     a = t;
182     b = one;
183     /*
184     * estimate log((2/x)^n*n!) = n*log(2/x)+n*ln(n)
185     * hence, if n*(log(2n/x)) > ...
186     * single 8.8722839355e+01
187     * double 7.09782712893383973096e+02
188     * long double 1.1356523406294143949491931077970765006170e+04
189     * then recurrent value may overflow and the result is
190     * likely underflow to zero
191     */
192     tmp = n;
193     v = two/x;
194     tmp = tmp*logl(fabsl(v*tmp));
195     if (tmp < 1.1356523406294143949491931077970765e+04L) {
196         for (i = n-1; i > 0; i--) {
197             temp = b;
198             b = ((i+i)/x)*b - a;
199             a = temp;
200         }
201     } else {
202         for (i = n-1; i > 0; i--) {
203             temp = b;
204             b = ((i+i)/x)*b - a;
205             a = temp;
206             if (b > 1e1000L) {
207                 a /= b;
208                 t /= b;
209                 b = 1.0;
210             }
211         }
212     }
213     b = (t*j0l(x)/b);
214 }
215 }
216 if (sgn == 1)
217     return (-b);
218 else
219     return (b);
220 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/lgamma.c

1

1325 Tue Nov 25 12:57:29 2014

new/usr/src/lib/libm/common/Q/lgamma.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __lgamma = lgamma
30 #pragma weak lgamma = __lgamma

32 /*
33  * long double lgamma(long double x);
34 */

36 #include "libm.h"
37 #include "longdouble.h"

39 extern int signgam;
40 extern int signgaml;

42 long double
43 lgamma(long double x) {
44     long double y = __k_lgamma(x, &signgaml);

46     signgam = signgaml; /* SUSv3 requires the setting of signgam */
47     return (y);
48 }
unchanged portion omitted
```

new/usr/src/lib/libm/common/Q/lgamma_r.c

1

1237 Tue Nov 25 12:57:29 2014

new/usr/src/lib/libm/common/Q/lgamma_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 /*
31  * long double lgamma_r(long double x, int *signgamp);
32 */
```

```
34 #pragma weak __lgamma_r = lgamma_r
34 #pragma weak lgamma_r = __lgamma_r
```

```
36 #include "libm.h"
37 #include "longdouble.h"
```

```
39 long double
40 lgamma_r(long double x, int *signgamp) {
41     return (__k_lgamma(x, signgamp));
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/log101.c

1

```
*****
3044 Tue Nov 25 12:57:30 2014
new/usr/src/lib/libm/common/Q/log101.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __log101 = log101
30 #if defined(ELFOBJ)
31 #pragma weak log101 = __log101
32 #endif

32 /*
33  * log101(X)
34  *
35  * Method :
36  *   Let log10_2hi = leading 98(SPARC)/49(x86) bits of log10(2) and
37  *   log10_2lo = log10(2) - log10_2hi,
38  *   ivln10 = 1/log(10) rounded.
39  *   Then
40  *       n = ilogb(x),
41  *       if (n<0) n = n+1;
42  *       x = scalbn(x,-n);
43  *       LOG10(x) := n*log10_2hi + (n*log10_2lo + ivln10*log(x))
44  *
45  * Note1:
46  *   For fear of destroying log10(10**n)=n, the rounding mode is
47  *   set to Round-to-Nearest.
48  *
49  * Special cases:
```

new/usr/src/lib/libm/common/Q/log101.c

2

```
50 *   log10(x) is NaN with signal if x < 0;
51 *   log10(+INF) is +INF with no signal; log10(0) is -INF with signal;
52 *   log10(NaN) is that NaN with no signal;
53 *   log10(10**N) = N for N=0,1,...,22.
54 *
55 * Constants:
56 * The hexadecimal values are the intended ones for the following constants.
57 * The decimal values may be used, provided that the compiler will convert
58 * from decimal to binary accurately enough to produce the hexadecimal values
59 * shown.
60 */

62 #include "libm.h"
63 #include "longdouble.h"

65 #if defined(__x86)
66 #define __swapRD __swap87RD
67 #endif
68 extern enum fp_direction_type __swapRD(enum fp_direction_type);

70 static const long double
71     zero = 0.0L,
72     ivln10 = 4.342944819032518276511289189166050822944e-0001L,
73     one = 1.0L,
74 #if defined(__x86)
75     log10_2hi = 3.010299956639803653501985536422580480576e-01L,
76     log10_2lo = 8.298635403410822349787106337291183585413e-16L;
77 #elif defined(__sparc)
78     log10_2hi = 3.010299956639811952137388947242098603469e-01L,
79     log10_2lo = 2.831664213089468167896664371953210945664e-31L;
80 #else
81 #error Unknown Architecture!
82 #endif

84 long double
85 log101(long double x) {
86     long double y, z;
87     enum fp_direction_type rd;
88     int n;

90     if (!finitel(x))
91         return (x + fabsl(x)); /* x is +-INF or NaN */
92     else if (x > zero) {
93         n = ilogbl(x);
94         if (n < 0)
95             n += 1;
96         rd = __swapRD(fp_nearest);
97         y = n;
98         x = scalbnl(x, -n);
99         z = y * log10_2lo + ivln10 * logl(x);
100        z += y * log10_2hi;
101        if (rd != fp_nearest)
102            (void) __swapRD(rd);
103        return (z);
104    } else if (x == zero) /* -INF */
105        return (-one / zero);
106    else /* x < 0, return NaN */
107        return (zero / zero);
108 }

unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/loglpl.c

1

```
*****
6978 Tue Nov 25 12:57:30 2014
new/usr/src/lib/libm/common/Q/loglpl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #ifdef __LITTLE_ENDIAN
31 #define H0(x) *(3 + (int *) &x)
32 #define H1(x) *(2 + (int *) &x)
33 #define H2(x) *(1 + (int *) &x)
34 #define H3(x) *(int *) &x
35 #else
36 #define H0(x) *(int *) &x
37 #define H1(x) *(1 + (int *) &x)
38 #define H2(x) *(2 + (int *) &x)
39 #define H3(x) *(3 + (int *) &x)
40 #endif
42 /*
43  * loglpl(x)
44  * Table look-up algorithm by modifying logl.c
45  * By K.C. Ng, July 6, 1995
46  *
47  * (a). For 1+x in [31/33,33/31], using a special approximation:
48  * s = x/(2.0+x); ... here |s| <= 0.03125
49  * z = s*s;
50  * return x-s*(x-z*(B1+z*(B2+z*(B3+z*(B4+...+z*B9)...)));
51  * (i.e., x is in [-2/33,2/31])
52  *
53  * (b). Otherwise, normalize 1+x = 2^n * 1.f.
54  * Here we may need a correction term for 1+x rounded.
55  * Use a 6-bit table look-up: find a 6 bit g that match f to 6.5 bits,
56  * then
57  * log(1+x) = n*ln2 + log(1.g) + log(1.f/1.g).
58  * Here the leading and trailing values of log(1.g) are obtained from
```

new/usr/src/lib/libm/common/Q/loglpl.c

2

```
59 * a size-64 table.
60 * For log(1.f/1.g), let s = (1.f-1.g)/(1.f+1.g). Note that
61 * 1.f = 2^-n(1+x)
62 *
63 * then
64 * log(1.f/1.g) = log((1+s)/(1-s)) = 2s + 2/3 s^3 + 2/5 s^5 +...
65 * Note that |s|<2^-8=0.00390625. We use an odd s-polynomial
66 * approximation to compute log(1.f/1.g):
67 * s*(A1+s^2*(A2+s^2*(A3+s^2*(A4+s^2*(A5+s^2*(A6+s^2*A7))))))
68 * (Precision is 2**(-136.91 bits, absolute error)
69 *
70 * CAUTION:
71 * For x>=1, compute 1+x will lost one bit (OK).
72 * For x in [-0.5,-1), 1+x is exact.
73 * For x in (-0.5,-2/33]U[2/31,1), up to 4 last bits of x will be lost
74 * in 1+x. Therefore, to recover the lost bits, one need to compute
75 * 1.f-1.g accurately.
76 *
77 * Let hx = HI(x), m = (hx>>16)-0x3fff (=ilogbl(x)), note that
78 * -2/33 = -0.0606... = 2^-5 * 1.939...,
79 * 2/31 = 0.09375 = 2^-4 * 1.500...,
80 * so for x in (-0.5,-2/33], -5<=m<=-2, n = -1, 1+f=2*(1+x)
81 * for x in [2/33,1), -4<=m<=-1, n = 0, f=x
82 *
83 * In short:
84 * if x>0, let g: hg = ((hx + (0x200<<(-m)))>>(10-m))<<(10-m)
85 * then 1.f-1.g = x-g
86 * if x<0, let g': hg' = ((ix-(0x200)<<(-m-1))>>(9-m))<<(9-m)
87 * (ix=hx&0x7fffffff)
88 * then 1.f-1.g = 2*(g'+x),
89 *
90 * (c). The final result is computed by
91 * (n*ln2_hi+_TBL_logl_hi[j]) +
92 * ((n*ln2_lo+_TBL_logl_lo[j]) + s*(A1+...))
93 *
94 * Note.
95 * For ln2_hi and _TBL_logl_hi[j], we force their last 32 bit to be zero
96 * so that n*ln2_hi + _TBL_logl_hi[j] is exact. Here
97 * _TBL_logl_hi[j] + _TBL_logl_lo[j] match log(1+j*2**(-6)) to 194 bits
98 *
99 *
100 * Special cases:
101 * log(x) is NaN with signal if x < 0 (including -INF) ;
102 * log(+INF) is +INF; log(0) is -INF with signal;
103 * log(NaN) is that NaN with no signal.
104 *
105 * Constants:
106 * The hexadecimal values are the intended ones for the following constants.
107 * The decimal values may be used, provided that the compiler will convert
108 * from decimal to binary accurately enough to produce the hexadecimal values
109 * shown.
110 */
112 #pragma weak __loglpl = loglpl
112 #pragma weak loglpl = __loglpl
114 #include "libm.h"
116 extern const long double _TBL_logl_hi[], _TBL_logl_lo[];
118 static const long double
119 zero = 0.0L,
120 one = 1.0L,
121 two = 2.0L,
122 ln2hi = 6.931471805599453094172319547495844850203e-0001L,
123 ln2lo = 1.667085920830552208890449330400379754169e-0025L,
```


new/usr/src/lib/libm/common/Q/log2l.c

1

```
*****
1606 Tue Nov 25 12:57:31 2014
new/usr/src/lib/libm/common/Q/log2l.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * log2l(x)
32  * RETURN THE BASE 2 LOGARITHM OF X
33  *
34  * Method:
35  *   purge off 0, INF, and NaN.
36  *   n = ilogb(x)
37  *   if (n<0) n+=1
38  *   z = scalbn(x,-n)
39  *   LOG2(x) = n + (1/ln2)*log(x)
40 */
41
42 #pragma weak __log2l = log2l
42 #pragma weak log2l = __log2l
43
44 #include "libm.h"
45 #include "longdouble.h"
46
47 static const long double
48     zero    = 0.0L,
49     half    = 0.5L,
50     one     = 1.0L,
51     invln2  = 1.442695040888963407359924681001892137427e+0000L;
52
53 long double
54 log2l(long double x) {
55     int n;
56
57     if (x == zero || !finitel(x))
```

new/usr/src/lib/libm/common/Q/log2l.c

2

```
58         return (logl(x));
59     n = ilogbl(x);
60     if (n < 0)
61         n += 1;
62     x = scalbnl(x, -n);
63     if (x == half)
64         return (n - one);
65     return (n + invln2 * logl(x));
66 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/Q/logbl.c

1

2454 Tue Nov 25 12:57:31 2014

new/usr/src/lib/libm/common/Q/logbl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __logbl = logbl
30 #if defined(ELFOSBJ)
31 #pragma weak logbl = __logbl
32 #endif
33
34 #include "libm.h"
35 #include "xpg6.h" /* __xpg6 */
36 #define _C99SUSv3_logb _C99SUSv3_logb_subnormal_is_like_ilogb
37
38 #if defined(__sparc)
39 #define ISNORMALL(k, x) (k != 0x7fff) /* assuming k != 0 */
40 #define X86PDNRM(k, x)
41 #define XSCALE_OFFSET 0x403e /* 0x3fff + 112 */
42 static const long double xscale = 5192296858534827628530496329220096.0L;
43 /* 2^112 */
44 #elif defined(__x86)
45 /*
46  * if pseudo-denormal, replace by the equivalent normal
47  */
48 #define X86PDNRM(k, x) if (k == 0 && (((int *) &x)[1] & 0x80000000) != 0) \
49 ((int *) &x)[2] |= k = 1
50 #if defined(HANDLE_UNSUPPORTED)
51 #define ISNORMALL(k, x) (k != 0x7fff && (((int *) &x)[1] & 0x80000000) != 0)
52 #endif
53 #endif
```

new/usr/src/lib/libm/common/Q/logbl.c

2

```
50 #else
51 #define ISNORMALL(k, x) (k != 0x7fff)
52 #endif
53 #define XSCALE_OFFSET 0x403e /* 0x3fff + 63 */
54 static const long double xscale = 9223372036854775808.0L; /* 2^63 */
55 #endif
56
57 static long double
58 raise_division(long double v) {
59 #pragma STDC FENV_ACCESS ON
60 static const long double zero = 0.0L;
61 return (v / zero);
62 }
63
64 unchanged_portion_omitted
```

```
*****
5512 Tue Nov 25 12:57:32 2014
new/usr/src/lib/libm/common/Q/logl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __logl = logl
30 #pragma weak logl = __logl
32 /*
33  * logl(x)
34  * Table look-up algorithm
35  * By K.C. Ng, March 6, 1989
36  *
37  * (a). For x in [31/33,33/31], using a special approximation:
38  *   f = x - 1;
39  *   s = f/(2.0+f); ... here |s| <= 0.03125
40  *   z = s*s;
41  *   return f-s*(f-z*(B1+z*(B2+z*(B3+z*(B4+...+z*B9)...)));
42  *
43  * (b). Otherwise, normalize x = 2^n * 1.f.
44  *   Use a 6-bit table look-up: find a 6 bit g that match f to 6.5 bits,
45  *   then
46  *   log(x) = n*ln2 + log(1.g) + log(1.f/1.g).
47  *   Here the leading and trailing values of log(1.g) are obtained from
48  *   a size-64 table.
49  *   For log(1.f/1.g), let s = (1.f-1.g)/(1.f+1.g), then
50  *   log(1.f/1.g) = log((1+s)/(1-s)) = 2s + 2/3 s^3 + 2/5 s^5 + ...
51  *   Note that |s|<2**=-8=0.00390625. We use an odd s-polynomial
52  *   approximation to compute log(1.f/1.g):
53  *   s*(A1+s^2*(A2+s^2*(A3+s^2*(A4+s^2*(A5+s^2*(A6+s^2*A7))))))
54  *   (Precision is 2**=-136.91 bits, absolute error)
55  *
56  * (c). The final result is computed by
57  *   (n*ln2_hi+_TBL_logl_hi[j]) +
```

```
58 *   ((n*ln2_lo+_TBL_logl_lo[j]) + s*(A1+...))
59 *
60 * Note.
61 * For ln2_hi and _TBL_logl_hi[j], we force their last 32 bit to be zero
62 * so that n*ln2_hi + _TBL_logl_hi[j] is exact. Here
63 *   _TBL_logl_hi[j] + _TBL_logl_lo[j] match log(1+j*2**=-6) to 194 bits
64 *
65 *
66 * Special cases:
67 *   log(x) is NaN with signal if x < 0 (including -INF) ;
68 *   log(+INF) is +INF; log(0) is -INF with signal;
69 *   log(NaN) is that NaN with no signal.
70 *
71 * Constants:
72 * The hexadecimal values are the intended ones for the following constants.
73 * The decimal values may be used, provided that the compiler will convert
74 * from decimal to binary accurately enough to produce the hexadecimal values
75 * shown.
76 */
78 #include "libm.h"
80 extern const long double _TBL_logl_hi[], _TBL_logl_lo[];
82 static const long double
83 zero = 0.0L,
84 one = 1.0L,
85 two = 2.0L,
86 twoll3 = 10384593717069655257060992658440192.0L,
87 ln2hi = 6.931471805599453094172319547495844850203e-0001L,
88 ln2lo = 1.667085920830552208890449330400379754169e-0025L,
89 A1 = 2.0000000000000000000000000000000000000000000024e+0000L,
90 A2 = 6.6666666666666666666666666666666666091393804e-0001L,
91 A3 = 4.0000000000000000000000000000000000040716707220671e-0001L,
92 A4 = 2.857142857142857142730077490612903681164e-0001L,
93 A5 = 2.2222222222222242577702836920812882605099e-0001L,
94 A6 = 1.818181816435493395985912667105885828356e-0001L,
95 A7 = 1.538537835211839751112067512805496931725e-0001L,
96 B1 = 6.6666666666666666666666666666666666666691498329e-0001L,
97 B2 = 3.99999999999999999999999999999990037655042358e-0001L,
98 B3 = 2.857142857142857142857273426428347457918e-0001L,
99 B4 = 2.2222222222222221353229049747910109566e-0001L,
100 B5 = 1.818181818181821503532559306309070138046e-0001L,
101 B6 = 1.538461538453809210486356084587356788556e-0001L,
102 B7 = 1.33333344463358756121456892645178795480e-0001L,
103 B8 = 1.176460904783899064854645174603360383792e-0001L,
104 B9 = 1.057293869956598995326368602518056990746e-0001L;
106 long double
107 logl(long double x) {
108     long double f, s, z, qn, h, t;
109     int *px = (int *) &x;
110     int *pz = (int *) &z;
111     int i, j, ix, io, il, n;
113     /* get long double precision word ordering */
114     if (*(int *) &one == 0) {
115         io = 3;
116         il = 0;
117     } else {
118         io = 0;
119         il = 3;
120     }
122     n = 0;
123     ix = px[io];
```

```

124     if (ix > 0x3fffe0f8) { /* if x > 31/33 */
125         if (ix < 0x3fff1084) { /* if x < 33/31 */
126             f = x - one;
127             z = f * f;
128             if (((ix - 0x3fff0000) | px[i1] | px[2] | px[1]) == 0) {
129                 return (zero); /* log(1)= +0 */
130             }
131             s = f / (two + f); /* |s|<2**-8 */
132             z = s * s;
133             return (f - s * (f - z * (B1 + z * (B2 + z * (B3 +
134                 z * (B4 + z * (B5 + z * (B6 + z * (B7 +
135                 z * (B8 + z * B9))))))));
136         }
137         if (ix >= 0x7fff0000)
138             return (x + x); /* x is +inf or NaN */
139         goto LARGE_N;
140     }
141     if (ix >= 0x00010000)
142         goto LARGE_N;
143     i = ix & 0x7fffffff;
144     if ((i | px[i1] | px[2] | px[1]) == 0) {
145         px[i0] |= 0x80000000;
146         return (one / x); /* log(0.0) = -inf */
147     }
148     if (ix < 0) {
149         if ((unsigned) ix >= 0xffff0000)
150             return (x - x); /* x is -inf or NaN */
151         return (zero / zero); /* log(x<0) is NaN */
152     }
153     /* subnormal x */
154     x *= twoll3;
155     n = -113;
156     ix = px[i0];
157     LARGE_N:
158     n += ((ix + 0x200) >> 16) - 0x3fff;
159     ix = (ix & 0x0000ffff) | 0x3fff0000; /* scale x to [1,2] */
160     px[i0] = ix;
161     i = ix + 0x200;
162     pz[i0] = i & 0xfffffc00;
163     pz[i1] = pz[1] = pz[2] = 0;
164     s = (x - z) / (x + z);
165     j = (i >> 10) & 0x3f;
166     z = s * s;
167     qn = (long double) n;
168     t = qn * ln2lo + _TBL_logl_lo[j];
169     h = qn * ln2hi + _TBL_logl_hi[j];
170     f = t + s * (A1 + z * (A2 + z * (A3 + z * (A4 + z * (A5 +
171         z * (A6 + z * A7))))));
172     return (h + f);
173 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/nextafter1.c

1

```
*****
2732 Tue Nov 25 12:57:32 2014
new/usr/src/lib/libm/common/Q/nextafter1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __nextafter1 = nextafter1
30 #if defined(ELFOBJ)
31 #pragma weak nextafter1 = __nextafter1
32 #endif
33
34 #include "libm.h"
35 #include <float.h>
36
37 #if defined(__sparc)
38 #define n0 0
39 #define n1 1
40 #define n2 2
41 #define n3 3
42 #define X86PDNRML(x)
43 #define INC(px) { \
44     if (++px[n3] == 0) \
45         if (++px[n2] == 0) \
46             if (++px[n1] == 0) \
47                 ++px[n0]; \
48 }
49 #define DEC(px) { \
50     if (--px[n3] == 0xfffffff) \
51         if (--px[n2] == 0xfffffff) \
```

new/usr/src/lib/libm/common/Q/nextafter1.c

2

```
52     if (--px[n1] == 0xffffffff) \
53         --px[n0]; \
54 }
55 #elif defined(__x86)
56 #define n0 2
57 #define n1 1
58 #define n2 0
59 #define n3 0
60 /*
61  * if pseudo-denormal, replace by the equivalent normal
62  */
63 #define X86PDNRML(x) if (XBIASED_EXP(x) == 0 && (((int *) &x)[1] & \
64     0x80000000) != 0) \
65     ((int *) &x)[2] |= 1
66 #define INC(px) { \
67     if (++px[n2] == 0) \
68         if ((++px[n1] & ~0x80000000) == 0) \
69             px[n1] = 0x80000000, ++px[n0]; \
70 }
71 #define DEC(px) { \
72     if (--px[n2] == 0xffffffff) \
73         if (--px[n1] == 0x7fffff) \
74             if ((--px[n0] & 0x7fff) != 0) \
75                 px[n1] |= 0x80000000; \
76 }
77 #endif
78
79 long double
80 nextafter1(long double x, long double y) {
81     int *px = (int *) &x;
82     int *py = (int *) &y;
83
84     if (x == y)
85         return (y);
86     if (x != x || y != y)
87         return (x * y);
88
89     if (ISZEROL(x)) {
90         /* x == 0.0 */
91         px[n0] = py[n0] & XSGNMSK;
92         px[n1] = px[n2] = 0;
93         px[n3] = 1;
94     } else {
95         X86PDNRML(x);
96         if ((px[n0] & XSGNMSK) == 0) { /* x > 0.0 */
97             if (x > y)
98                 /* x > y */
99                 DEC(px);
100             else
101                 INC(px);
102         } else {
103             if (x < y)
104                 /* x < y */
105                 DEC(px);
106             else
107                 INC(px);
108         }
109     }
110 #ifndef lint
111     {
112         volatile long double dummy;
113         int k = XBIASED_EXP(x);
114
115         if (k == 0)
116             dummy = LDBL_MIN * copysign(LDBL_MIN, x);
117         else if (k == 0x7fff)
118             dummy = LDBL_MAX * copysign(LDBL_MAX, x);
119     }
120 #endif
121 }
122 #endif
```

new/usr/src/lib/libm/common/Q/nextafter1.c

3

```
116     return (x);  
117 }
```

_____unchanged_portion_omitted_____


```
116     t += s * (v * (A2 + v * (A3 + v * (A4 + v * (A5 + v * (A6 +
117               v * A7))))));
118     v = qn * ln2hi + _TBL_logl_hi[j];
119     s = h + v;
120     t += (h - (s - v));
121     z = (long double) ((double) (s + t));
122     *w = t - (z - s);
123     return (z);
124 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/Q/remainder1.c

1

```
*****
2216 Tue Nov 25 12:57:33 2014
new/usr/src/lib/libm/common/Q/remainder1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __remainder1 = remainder1
30 #pragma weak remainder1 = __remainder1

32 #include "libm.h"
33 #include "longdouble.h"

35 /*
36  * remainder1(x,p)
37  * returns x REM p = x - [x/p]*p as if in infinite
38  * precise arithmetic, where [x/p] is the (inifinite bit)
39  * integer nearest x/p (in half way case choose the even one).
40  * Method :
41  * Based on fmodl() return x-[x/p]chopped*p exactly.
42  */

44 #define HFMAX 5.948657476786158825428796633140035080982e+4931L
45 #define DBMIN 6.724206286224187012525355634643505205196e-4932L

47 static const long double
48 zero = 0.0L,
49 half = 0.5L,
50 hfmmax = HFMAX, /* half of the maximum number */
51 dbmin = DBMIN; /* double of the minimum (normal) number */

53 long double
54 remainder1(long double x, long double p) {
55     long double hp;
56     int sx;
```

new/usr/src/lib/libm/common/Q/remainder1.c

2

```
58     if (isnanl(p))
59         return (x + p);
60     if (!finitel(x))
61         return (x - x);
62     p = fabsl(p);
63     if (p <= hfmmax)
64         x = fmodl(x, p + p);
65     sx = signbitl(x);
66     x = fabsl(x);
67     if (p < dbmin) {
68         if (x + x > p) {
69             if (x == p)
70                 x = zero;
71             else
72                 x -= p; /* avoid x-x=-0 in RM mode */
73             if (x + x >= p)
74                 x -= p;
75         }
76     } else {
77         hp = half * p;
78         if (x > hp) {
79             if (x == p)
80                 x = zero;
81             else
82                 x -= p; /* avoid x-x=-0 in RM mode */
83             if (x >= hp)
84                 x -= p;
85         }
86     }
87     return (sx == 0 ? x : -x);
88 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/rintl.c

1

```
*****
2131 Tue Nov 25 12:57:33 2014
new/usr/src/lib/libm/common/Q/rintl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __rintl = rintl
30 #pragma weak rintl = __rintl

32 /*
33  * rintl(long double x) return x rounded to integral according to
34  * the prevailing rounding direction
35  *
36  * NOTE: aintl(x), anintl(x), ceill(x), floorl(x), and rintl(x) return result
37  * with the same sign as x's, including 0.0L.
38  */

40 #include "libm.h"
41 #include "longdouble.h"

43 extern enum fp_precision_type __swapRP(enum fp_precision_type);

45 static const double one = 1.0;
46 static const long double qzero = 0.0L;

48 long double
49 rintl(long double x) {
50     enum fp_precision_type rp;
51     long double t, w, twoll12;
52     int *pt = (int *) &twoll12;

54     if (!finitel(x))
55         return (x + x);

57     if (*(int *) &one != 0) { /* set twoll12 = 2^112 */
```

new/usr/src/lib/libm/common/Q/rintl.c

2

```
58         pt[0] = 0x406f0000;
59         pt[1] = pt[2] = pt[3] = 0;
60     } else {
61         pt[3] = 0x406f0000;
62         pt[0] = pt[1] = pt[2] = 0;
63     }

65     if (fabsl(x) >= twoll12)
66         return (x); /* already an integer */
67     t = copysignl(twoll12, x);
68     rp = __swapRP(fp_extended); /* make sure precision is long double */
69     w = x + t; /* x+sign(x)*2^112 rounded to integer */
70     (void) __swapRP(rp); /* restore precision mode */
71     if (w == t)
72         return (copysignl(qzero, x)); /* x rounded to zero */
73     else
74         return (w - t);
75 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/Q/rndintl.c

1

```
*****
2826 Tue Nov 25 12:57:34 2014
new/usr/src/lib/libm/common/Q/rndintl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak aintl = __aintl
30 #pragma weak anintl = __anintl
31 #pragma weak irintl = __irintl
32 #pragma weak nintl = __nintl

29 /*
30  * aintl(x)    return x chopped to integral value
31  * anintl(x)   return sign(x)*(|x|+0.5) chopped to integral value
32  * irintl(x)   return rint(x) in integer format
33  * nintl(x)    return aint(x) in integer format
34  *
35  * NOTE: aintl(x), anintl(x), ceil(x), floor(x), and rint(x) return result
36  * with the same sign as x's, including 0.0.
37  */

39 #include "libm.h"
40 #include "longdouble.h"

42 extern enum fp_direction_type __swapRD(enum fp_direction_type);

44 static const long double qone = 1.0L, qhalf = 0.5L, qmhalf = -0.5L;

46 long double
47 aintl(long double x) {
48     long double t, w;

50     if (!finitel(x))
51         return (x + x);
52     w = fabs(x);
53     t = rint(w);
```

new/usr/src/lib/libm/common/Q/rndintl.c

2

```
54     if (t <= w)
55         return (copysignl(t, x));          /* NaN or already aint(|x|) */
56     else /* |t| > |x| case */
57         return (copysignl(t - qone, x));   /* |t-1|*sign(x) */
58 }
-----unchanged_portion_omitted-----
```

new/usr/src/lib/libm/common/Q/scalbl.c

1

```
*****
1650 Tue Nov 25 12:57:35 2014
new/usr/src/lib/libm/common/Q/scalbl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __scalbl = scalbl
30 #pragma weak scalbl = __scalbl
31
32 /*
33  * scalbl(x,n): return x * 2^n by manipulating exponent.
34 */
35
36 #include "libm.h"
37
38 #define n0      0
39
40 long double
41 scalbl(long double x, long double fn) {
42     int *py = (int *) &fn, n;
43     long double z;
44
45     if (isnanl(x) || isnanl(fn))
46         return (x * fn);
47
48     /*
49      * fn is inf or NaN
50      */
51     if ((py[n0] & 0x7fff0000) == 0x7fff0000) {
52         if ((py[n0] & 0x80000000) != 0)
53             return (x / (-fn));
54         else
55             return (x * fn);
56     }
57     if (rintl(fn) != fn)
```

new/usr/src/lib/libm/common/Q/scalbl.c

2

```
58         return ((fn - fn) / (fn - fn));
59     if (fn > 65000.0L)
60         z = scalbnl(x, 65000);
61     else if (-fn > 65000.0L)
62         z = scalbnl(x, -65000);
63     else {
64         n = (int) fn;
65         z = scalbnl(x, n);
66     }
67     return (z);
68 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/Q/scalbnl.c

1

```
*****
2490 Tue Nov 25 12:57:35 2014
new/usr/src/lib/libm/common/Q/scalbnl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __scalbnl = scalbnl
30 #if defined(ELFOBJ)
31 #pragma weak scalbnl = __scalbnl
32 #endif

32 #include "libm.h"
33 #include <float.h>          /* LDBL_MAX, LDBL_MIN */
34 #include <stdlib.h>        /* abs */

36 #if defined(__sparc)
37 #define XSET_EXP(k, x) (((int *) &x)[0] = (((int *) &x)[0] & ~0x7fff0000) | \
38                      (k << 16))
39 #define ISINFNANL(k, x) (k == 0x7fff)
40 #define XTWOT_OFFSET 113
41 static const long double xtwt = 10384593717069655257060992658440192.0L,
42                      twomt1 = 4.814824860968089632639944856462318296E-35L; /* 2^113 */
43                      /* 2^-114 */
44 #elif defined(__x86)
45 #define XSET_EXP(k, x) (((int *) &x)[2] = (((int *) &x)[2] & ~0x7fff) | k
46 #if defined(HANDLE_UNSUPPORTED)
47 #define ISINFNANL(k, x) (k == 0x7fff || k != 0 && \
48                      (((int *) &x)[1] & 0x80000000) == 0)
49 #else

```

new/usr/src/lib/libm/common/Q/scalbnl.c

2

```
50 #define ISINFNANL(k, x) (k == 0x7fff)
51 #endif
52 #define XTWOT_OFFSET 64
53 static const long double xtwt = 18446744073709551616.0L, /* 2^64 */
54                      twomt1 = 2.7105054312137610850186E-20L; /* 2^-65 */
55 #endif

57 long double
58 scalbnl(long double x, int n) {
59     int k = XBIASED_EXP(x);

61     if (ISINFNANL(k, x))
62         return (x + x);
63     if (ISZEROL(x) || n == 0)
64         return (x);
65     if (k == 0) {
66         x *= xtwt;
67         k = XBIASED_EXP(x) - XTWOT_OFFSET;
68     }
69     if ((unsigned) abs(n) >= 131072) /* cast to unsigned for -2^31 */
70         n >>= 1; /* avoid subsequent integer overflow */
71     k += n;
72     if (k > 0x7ffe)
73         return (LDBL_MAX * copysignl(LDBL_MAX, x));
74     if (k <= -XTWOT_OFFSET - 1)
75         return (LDBL_MIN * copysignl(LDBL_MIN, x));
76     if (k > 0) {
77         XSET_EXP(k, x);
78         return (x);
79     }
80     k += XTWOT_OFFSET + 1;
81     XSET_EXP(k, x);
82     return (x * twomt1);
83 }
_____
unchanged portion omitted_
```

new/usr/src/lib/libm/common/Q/signgaml.c

1

1070 Tue Nov 25 12:57:36 2014

new/usr/src/lib/libm/common/Q/signgaml.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __signgaml = signgaml
30 #pragma weak signgaml = __signgaml
```

```
32 #include "libm.h"
```

```
34 int signgaml = 0;
```

new/usr/src/lib/libm/common/Q/significandl.c

1

1226 Tue Nov 25 12:57:36 2014

new/usr/src/lib/libm/common/Q/significandl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __significandl = significandl
30 #if defined(ELFOBJ)
31 #pragma weak significandl = __significandl
32 #endif
```

```
32 #include "libm.h"
```

```
34 long double
35 significandl(long double x) {
36     if (ISZEROL(x) || XBIASED_EXP(x) == 0x7fff) /* 0/+Inf/NaN */
37         return (x + x);
38     else
39         return (scalbnl(x, -ilogbl(x)));
40 }
```

unchanged_portion_omitted_

```

*****
2709 Tue Nov 25 12:57:37 2014
new/usr/src/lib/libm/common/Q/sincosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 /*
31  * sincosl(x)
32  * Table look-up algorithm by K.C. Ng, November, 1989.
33  *
34  * kernel function:
35  *   __k_sincosl    ... sin and cos function on [-pi/4,pi/4]
36  *   __rem_pio2l   ... argument reduction routine
37  *
38  * Method.
39  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
40  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
41  *   [-pi/2 , +pi/2], and let n = k mod 4.
42  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
43  *
44  *           n      sin(x)      cos(x)      tan(x)
45  *   -----
46  *           0      S           C           S/C
47  *           1      C           -S          -C/S
48  *           2      -S          -C           S/C
49  *           3      -C           S           -C/S
50  *   -----
51  *
52  * Special cases:
53  *   Let trig be any of sin, cos, or tan.
54  *   trig(++INF) is NaN, with signals;
55  *   trig(NaN)  is that NaN;
56  *
57  * Accuracy:
58  *   computer TRIG(x) returns trig(x) nearly rounded.

```

```

59 */
61 #pragma weak __sincosl = sincosl
61 #pragma weak sincosl = __sincosl

63 #include "libm.h"
64 #include "longdouble.h"

66 void
67 sincosl(long double x, long double *s, long double *c) {
68     long double y[2], z = 0.0L;
69     int n, ix;

71     ix = *(int *) &x;      /* High word of x */

73     /* |x| ~< pi/4 */
74     ix &= 0x7fffffff;
75     if (ix <= 0x3ffe9220)
76         *s = __k_sincosl(x, z, c);
77     else if (ix >= 0x7fff0000)
78         *s = *c = x - x;      /* trig(Inf or NaN) is NaN */
79     else {                    /* argument reduction needed */
80         n = __rem_pio2l(x, y);
81         switch (n & 3) {
82             case 0:
83                 *s = __k_sincosl(y[0], y[1], c);
84                 break;
85             case 1:
86                 *c = -__k_sincosl(y[0], y[1], s);
87                 break;
88             case 2:
89                 *s = -__k_sincosl(y[0], y[1], c);
90                 *c = -*c;
91                 break;
92             case 3:
93                 *c = __k_sincosl(y[0], y[1], s);
94                 *s = -*s;
95                 break;
96         }
97     }
98 }

```

unchanged portion omitted


```

*****
5939 Tue Nov 25 12:57:37 2014
new/usr/src/lib/libm/common/Q/sincospil.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak sincospil = __sincospil
31
32 /*
33  * void sincospil(long double x, long double *s, long double *c)
34  * *s = sinl(pi*x); *c = cosl(pi*x);
35  *
36  * Algorithm, 10/17/2002, K.C. Ng
37  * -----
38  * Let y = |4x|, z = floor(y), and n = (int)(z mod 8.0) (displayed in binary).
39  * 1. If y == z, then x is a multiple of pi/4. Return the following values:
40  *
41  *      n  x mod 2   sin(x*pi)   cos(x*pi)   tan(x*pi)
42  * -----
43  *      000  0.00    +0          +1          +0
44  *      001  0.25    +\0.5      +\0.5      +1
45  *      010  0.50    +1          +0          +inf
46  *      011  0.75    +\0.5      -\0.5      -1
47  *      100  1.00    -0          -1          +0
48  *      101  1.25    -\0.5      -\0.5      +1
49  *      110  1.50    -1          -0          +inf
50  *      111  1.75    -\0.5      +\0.5      -1
51  * -----
52  * 2. Otherwise,
53  *
54  *      n      t      sin(x*pi)   cos(x*pi)   tan(x*pi)
55  * -----
56  *      000  (y-z)/4   sinpi(t)   cospi(t)   tanpi(t)
57  *      001  (z+1-y)/4 cospi(t)   sinpi(t)   1/tanpi(t)
58  *      010  (y-z)/4   cospi(t)  -sinpi(t)  -1/tanpi(t)

```

```

57 *      011  (z+1-y)/4   sinpi(t)  -cospi(t)  -tanpi(t)
58 *      100  (y-z)/4    -sinpi(t) -cospi(t)   tanpi(t)
59 *      101  (z+1-y)/4  -cospi(t) -sinpi(t)  1/tanpi(t)
60 *      110  (y-z)/4    -cospi(t) sinpi(t)   -1/tanpi(t)
61 *      111  (z+1-y)/4  -sinpi(t)  cospi(t)  -tanpi(t)
62 * -----
63 *
64 * NOTE. This program compute sinpi/cospi(t<0.25) by __k_sin/cos(pi*t, 0.0).
65 * This will return a result with error slightly more than one ulp (but less
66 * than 2 ulp). If one wants accurate result, one may break up pi*t in
67 * high (tpi_h) and low (tpi_l) parts and call __k_sin/cos(tpi_h, tpi_lo)
68 * instead.
69 */
70
71 #include "libm.h"
72 #include "longdouble.h"
73
74 #define I(q, m) ((int *) &(q))[m]
75 #define U(q, m) ((unsigned *) &(q))[m]
76 #if defined(__LITTLE_ENDIAN) || defined(__x86)
77 #define LDBL_MOST_SIGNIF_I(ld) ((I(ld, 2) << 16) | (0xffff & (I(ld, 1) >> 15)))
78 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, 0)
79 #define PREC 64
80 #define PRECM1 63
81 #define PRECM2 62
82 static const long double twoPRECM2 = 9.2233720368547758080000000000000e+18L;
83 #else
84 #define LDBL_MOST_SIGNIF_I(ld) I(ld, 0)
85 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, sizeof (long double) / sizeof (int) - 1)
86 #define PREC 113
87 #define PRECM1 112
88 #define PRECM2 111
89 static const long double twoPRECM2 = 5.192296858534827628530496329220096e+33L;
90 #endif
91
92 static const long double
93 zero = 0.0L,
94 quater = 0.25L,
95 one = 1.0L,
96 pi = 3.141592653589793238462643383279502884197e+0000L,
97 sqrt2 = 0.707106781186547524400844362104849039284835937688474,
98 tiny = 1.0e-100;
99
100 void
101 sincospil(long double x, long double *s, long double *c) {
102     long double y, z, t;
103     int hx, n, k;
104     unsigned lx;
105
106     hx = LDBL_MOST_SIGNIF_I(x);
107     lx = LDBL_LEAST_SIGNIF_U(x);
108     k = ((hx & 0x7fff0000) >> 16) - 0x3fff;
109     if (k >= PRECM2) { /* |x| >= 2**(Prec-2) */
110         if (k >= 16384) {
111             *s = *c = x - x;
112         } else {
113             if (k >= PREC) {
114                 *s = zero;
115                 *c = one;
116             } else if (k == PRECM1) {
117                 if ((lx & 1) == 0) {
118                     *s = zero;
119                     *c = one;
120                 } else {
121                     *s = -zero;
122                     *c = -one;

```

```

123     } else {
124         /* k = Prec - 2 */
125         if ((lx & 1) == 0) {
126             *s = zero;
127             *c = one;
128         } else {
129             *s = one;
130             *c = zero;
131         }
132         if ((lx & 2) != 0) {
133             *s = -*s;
134             *c = -*c;
135         }
136     }
137 } else if (k < -2) /* |x| < 0.25 */
138     *s = __k_sincosl(pi * fabsl(x), zero, c);
139 else {
140     /* y = |4x|, z = floor(y), and n = (int)(z mod 8.0) */
141     y = 4.0L * fabsl(x);
142     if (k < PRECM2) {
143         z = y + twoPRECM2;
144         n = LDBL_LEAST_SIGNIF_U(z) & 7; /* 3 LSb of z */
145         t = z - twoPRECM2;
146         k = 0;
147         if (t == y)
148             k = 1;
149         else if (t > y) {
150             n -= 1;
151             t = quater + (y - t) * quater;
152         } else
153             t = (y - t) * quater;
154     } else {
155         /* k = Prec-3 */
156         n = LDBL_LEAST_SIGNIF_U(y) & 7; /* 3 LSb of z */
157         k = 1;
158     }
159     if (k) { /* x = N/4 */
160         if ((n & 1) != 0)
161             *s = *c = sqrth + tiny;
162         else
163             if ((n & 2) == 0) {
164                 *s = zero;
165                 *c = one;
166             } else {
167                 *s = one;
168                 *c = zero;
169             }
170         if ((n & 4) != 0)
171             *s = -*s;
172         if (((n + 1) & 4) != 0)
173             *c = -*c;
174     } else {
175         if ((n & 1) != 0)
176             t = quater - t;
177         if (((n + (n & 1)) & 2) == 0)
178             *s = __k_sincosl(pi * t, zero, c);
179         else
180             *c = __k_sincosl(pi * t, zero, s);
181         if ((n & 4) != 0)
182             *s = -*s;
183         if (((n + 2) & 4) != 0)
184             *c = -*c;
185     }
186 }
187 if (hx < 0)
188     *s = -*s;

```

```

189 }
    unchanged_portion_omitted

```

new/usr/src/lib/libm/common/Q/sinhl.c

1

```
*****
2392 Tue Nov 25 12:57:37 2014
new/usr/src/lib/libm/common/Q/sinhl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __sinhl = sinhl
30 #pragma weak sinhl = __sinhl
32 #include "libm.h"
33 #include "longdouble.h"
35 /*
36  * sinhl(X)
37  * RETURN THE HYPERBOLIC SINE OF X
38  *
39  * Method :
40  * 1. reduce x to non-negative by sinhl(-x) = - sinhl(x).
41  * 2.
42  *
43  *
44  * 0 <= x <= lnovft      : sinhl(x) :=  $\frac{\expm1(x) + \expm1(x)/(\expm1(x)+1)}{2}$ 
45  *
46  *
47  * lnovft <= x < INF    : sinhl(x) :=  $\exp(x - \text{MEP1} * \ln 2) * 2 * \text{ME}$ 
48  *
49  * here
50  * lnovft:      logarithm of the overflow threshold
51  *              = MEP1 * ln 2 chopped to machine precision.
52  * ME          maximum exponent
53  * MEP1       maximum exponent plus 1
54  *
55  * Special cases:
56  * sinhl(x) is x if x is +INF, -INF, or NaN.
57  * only sinhl(0)=0 is exact for finite argument.
```

new/usr/src/lib/libm/common/Q/sinhl.c

2

```
58 *
59 */
61 #define ME      16383
62 #define MEP1   16384
63 #define LNOVFT 1.135652340629414394949193107797076342845e+4L
64 /* last 32 bits of LN2HI is zero */
65 #define LN2HI  6.931471805599453094172319547495844850203e-0001L
66 #define LN2LO  1.667085920830552208890449330400379754169e-0025L
68 static const long double
69     half = 0.5L,
70     one  = 1.0L,
71     ln2hi = LN2HI,
72     ln2lo = LN2LO,
73     lnovftL = LNOVFT;
75 long double
76 sinhl(long double x) {
77     long double r, t;
79     if (!finitel(x))
80         return (x + x); /* sinh of NaN or +-INF is itself */
81     r = fabsL(x);
82     if (r < lnovftL) {
83         t = expm1(r);
84         r = copysignl((t + t / (one + t)) * half, x);
85     } else {
86         r = copysignl(expL((r - MEP1 * ln2hi) - MEP1 * ln2lo), x);
87         r = scalbnL(r, ME);
88     }
89     return (r);
90 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/Q/sinl.c

1

```
*****
2674 Tue Nov 25 12:57:38 2014
new/usr/src/lib/libm/common/Q/sinl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * sinl(x)
32  * Table look-up algorithm by K.C. Ng, November, 1989.
33  *
34  * kernel function:
35  *   __k_sinl      ... sin function on [-pi/4,pi/4]
36  *   __k_cosl     ... cos function on [-pi/4,pi/4]
37  *   __rem_pio2l  ... argument reduction routine
38  *
39  * Method.
40  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
41  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
42  *   [-pi/2 , +pi/2], and let n = k mod 4.
43  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
44  *
45  *       n      sin(x)      cos(x)      tan(x)
46  *   -----
47  *       0          S          C          S/C
48  *       1          C         -S         -C/S
49  *       2         -S         -C          S/C
50  *       3         -C          S         -C/S
51  *   -----
52  *
53  * Special cases:
54  *   Let trig be any of sin, cos, or tan.
55  *   trig(+INF) is NaN, with signals;
56  *   trig(NaN)  is that NaN;
57  *
58  * Accuracy:
```

new/usr/src/lib/libm/common/Q/sinl.c

2

```
59  *      computer TRIG(x) returns trig(x) nearly rounded.
60  */
61
62 #pragma weak __sinl = sinl
62 #pragma weak sinl = __sinl
63
64 #include "libm.h"
65 #include "longdouble.h"
66
67 long double
68 sinl(long double x) {
69     long double y[2], z = 0.0L;
70     int n, ix;
71
72     ix = *(int *) &x;          /* High word of x */
73     ix &= 0x7fffffff;
74     if (ix <= 0x3ffe9220)     /* |x| ~< pi/4 */
75         return (__k_sinl(x, z));
76     else if (ix >= 0x7fff0000) /* sin(Inf or NaN) is NaN */
77         return (x - x);
78     else {                    /* argument reduction needed */
79         n = __rem_pio2l(x, y);
80         switch (n & 3) {
81             case 0:
82                 return (__k_sinl(y[0], y[1]));
83             case 1:
84                 return (__k_cosl(y[0], y[1]));
85             case 2:
86                 return (-__k_sinl(y[0], y[1]));
87             case 3:
88                 return (-__k_cosl(y[0], y[1]));
89         }
90     }
91     /* NOTREACHED */
92     return 0.0L;
93 }
```

unchanged_portion_omitted

```

*****
5517 Tue Nov 25 12:57:38 2014
new/usr/src/lib/libm/common/Q/sinpil.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak sinpil = __sinpil
31
32 /*
33  * long double sinpil(long double x),
34  * return long double precision sinl(pi*x).
35  *
36  * Algorithm, 10/17/2002, K.C. Ng
37  * -----
38  * Let  $y = |4x|$ ,  $z = \text{floor}(y)$ , and  $n = (\text{int})(z \bmod 8.0)$  (displayed in binary).
39  * 1. If  $y == z$ , then  $x$  is a multiple of  $\pi/4$ . Return the following values:
40  *
41  *
42  *
43  *
44  *
45  *
46  *
47  *
48  *
49  *
50  * 2. Otherwise,
51  *
52  *
53  *
54  *
55  *
56  *

```

n	x mod 2	sin(x*pi)	cos(x*pi)	tan(x*pi)
000	0.00	+0	+1	+0
001	0.25	$+\sqrt{0.5}$	$+\sqrt{0.5}$	+1
010	0.50	+1	+0	+inf
011	0.75	$+\sqrt{0.5}$	$-\sqrt{0.5}$	-1
100	1.00	-0	-1	+0
101	1.25	$-\sqrt{0.5}$	$-\sqrt{0.5}$	+1
110	1.50	-1	-0	+inf
111	1.75	$-\sqrt{0.5}$	$+\sqrt{0.5}$	-1

```

-----
n      t      sin(x*pi)  cos(x*pi)  tan(x*pi)
-----
000  (y-z)/4  sinpi(t)   cospi(t)   tanpi(t)
001  (z+1-y)/4  cospi(t)  sinpi(t)   1/tanpi(t)
010  (y-z)/4   cospi(t)  -sinpi(t)  -1/tanpi(t)

```

```

57 *      011  (z+1-y)/4  sinpi(t)   -cospi(t)  -tanpi(t)
58 *      100  (y-z)/4   -sinpi(t)  -cospi(t)  tanpi(t)
59 *      101  (z+1-y)/4  -cospi(t)  -sinpi(t)  1/tanpi(t)
60 *      110  (y-z)/4   -cospi(t)  sinpi(t)   -1/tanpi(t)
61 *      111  (z+1-y)/4  -sinpi(t)  cospi(t)   -tanpi(t)
62 *
63 *
64 * NOTE. This program compute sinpi/cospi(t<0.25) by  $\frac{\sinpi(t)}{\cospi(t)}$  by  $\frac{\sinpi(t)}{\cospi(t)}$  (0.0).
65 * This will return a result with error slightly more than one ulp (but less
66 * than 2 ulp). If one wants accurate result, one may break up pi*t in
67 * high (tpi_h) and low (tpi_l) parts and call  $\frac{\sinpi(t)}{\cospi(t)}$  (tip_h, tip_lo)
68 * instead.
69 */
70
71 #include "libm.h"
72 #include "longdouble.h"
73
74 #define I(q, m) ((int *) &(q))[m]
75 #define U(q, m) ((unsigned *) &(q))[m]
76 #if defined(__LITTLE_ENDIAN) || defined(__x86)
77 #define LDBL_MOST_SIGNIF_I(ld) ((I(ld, 2) << 16) | (0xffff & (I(ld, 1) >> 15)))
78 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, 0)
79 #define PREC 64
80 #define PREC1 63
81 #define PREC2 62
82 static const long double twoPREC2 = 9.22337203685477580800000000000000e+18L;
83 #else
84 #define LDBL_MOST_SIGNIF_I(ld) I(ld, 0)
85 #define LDBL_LEAST_SIGNIF_U(ld) U(ld, sizeof(long double) / sizeof(int) - 1)
86 #define PREC 113
87 #define PREC1 112
88 #define PREC2 111
89 static const long double twoPREC2 = 5.192296858534827628530496329220096e+33L;
90 #endif
91
92 static const long double
93 zero = 0.0L,
94 quater = 0.25L,
95 one = 1.0L,
96 pi = 3.141592653589793238462643383279502884197e+0000L,
97 sqrt2 = 0.707106781186547524400844362104849039284835937688474,
98 tiny = 1.0e-100;
99
100 long double
101 sinpil(long double x) {
102     long double y, z, t;
103     int hx, n, k;
104     unsigned lx;
105
106     hx = LDBL_MOST_SIGNIF_I(x);
107     lx = LDBL_LEAST_SIGNIF_U(x);
108     k = ((hx & 0x7fff0000) >> 16) - 0x3fff;
109     if (k >= PREC2) { /* |x| >= 2**(Prec-2) */
110         if (k >= 16384)
111             y = x - x;
112         else {
113             if (k >= PREC)
114                 y = zero;
115             else if (k == PREC1)
116                 y = (lx & 1) == 0 ? zero : -zero;
117             else { /* k = Prec - 2 */
118                 y = (lx & 1) == 0 ? zero : one;
119                 if ((lx & 2) != 0)
120                     y = -y;
121             }
122         }
123     }

```

```

123     } else if (k < -2)      /* |x| < 0.25 */
124         y = __k_sinl(pi * fabsl(x), zero);
125     else {
126         /* y = |4x|, z = floor(y), and n = (int)(z mod 8.0) */
127         y = 4.0L * fabsl(x);
128         if (k < PRECM2) {
129             z = y + twoPRECM2;
130             n = LDBL_LEAST_SIGNIF_U(z) & 7; /* 3 LSB of z */
131             t = z - twoPRECM2;
132             k = 0;
133             if (t == y)
134                 k = 1;
135             else if (t > y) {
136                 n -= 1;
137                 t = quater + (y - t) * quater;
138             } else
139                 t = (y - t) * quater;
140         } else {
141             /* k = Prec-3 */
142             n = LDBL_LEAST_SIGNIF_U(y) & 7; /* 3 LSB of z */
143             k = 1;
144         }
145         if (k) {
146             /* x = N/4 */
147             if ((n & 1) != 0)
148                 y = sqrth + tiny;
149             else
150                 y = (n & 2) == 0 ? zero : one;
151             if ((n & 4) != 0)
152                 y = -y;
153         } else {
154             if ((n & 1) != 0)
155                 t = quater - t;
156             if (((n + (n & 1)) & 2) == 0)
157                 y = __k_sinl(pi * t, zero);
158             else
159                 y = __k_cosl(pi * t, zero);
160             if ((n & 4) != 0)
161                 y = -y;
162         }
163     }
164     return (hx >= 0 ? y : -y);
165 }

```

unchanged portion omitted

new/usr/src/lib/libm/common/Q/sqrt1.c

1

```
*****
9966 Tue Nov 25 12:57:39 2014
new/usr/src/lib/libm/common/Q/sqrt1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __sqrt1 = sqrt1
30 #pragma weak sqrt1 = __sqrt1

32 #include "libm.h"
33 #include "longdouble.h"

35 extern int __swapTE(int);
36 extern int __swapEX(int);
37 extern enum fp_direction_type __swapRD(enum fp_direction_type);

39 /*
40  * in struct longdouble, msw consists of
41  *   unsigned short  sgn:1;
42  *   unsigned short  exp:15;
43  *   unsigned short  frac1:16;
44  */

46 #ifdef __LITTLE_ENDIAN

48 /* array indices used to access words within a double */
49 #define HIWORD  1
50 #define LOWORD  0

52 /* structure used to access words within a quad */
53 union longdouble {
54     struct {
55         unsigned int  frac4;
56         unsigned int  frac3;
57         unsigned int  frac2;
```

new/usr/src/lib/libm/common/Q/sqrt1.c

2

```
58         unsigned int  msw;
59     } l;
60     long double  d;
61 };
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/Q/tanh1.c

1

```
*****
2581 Tue Nov 25 12:57:39 2014
new/usr/src/lib/libm/common/Q/tanh1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __tanh1 = tanh1
30 #if defined(ELFOBJ)
31 #pragma weak tanh1 = __tanh1
32 #endif
33
34 /*
35  * tanh1(x) returns the hyperbolic tangent of x
36  *
37  * Method :
38  * 1. reduce x to non-negative: tanh1(-x) = - tanh1(x).
39  * 2.
40  * 0 < x <= small : tanh1(x) := x
41  *                    -expm1(-2x)
42  *                    -----
43  *                    expm1(-2x) + 2
44  * 1 <= x <= threshold : tanh1(x) := 1 - -----
45  *                    expm1(2x) + 2
46  * threshold < x <= INF : tanh1(x) := 1.
47  *
48  * where
49  * single : small = 1.e-5 threshold = 11.0
50  * double : small = 1.e-10 threshold = 22.0
```

new/usr/src/lib/libm/common/Q/tanh1.c

2

```
50 * quad : small = 1.e-20 threshold = 45.0
51 *
52 * Note: threshold was chosen so that
53 * fl(1.0+2/(expm1(2*threshold)+2)) == 1.
54 *
55 * Special cases:
56 * tanh1(NaN) is NaN;
57 * only tanh1(0.0)=0.0 is exact for finite argument.
58 */
59
60 #include "libm.h"
61 #include "longdouble.h"
62
63 static const long double small = 1.0e-20L, one = 1.0, two = 2.0,
64 #ifndef lint
65 big = 1.0e+20L,
66 #endif
67 threshold = 45.0L;
68
69 long double
70 tanh1(long double x) {
71     long double t, y, z;
72     int signx;
73     volatile long double dummy;
74
75     if (isnanl(x))
76         return (x + x); /* x is NaN */
77     signx = signbitl(x);
78     t = fabsl(x);
79     z = one;
80     if (t <= threshold) {
81         if (t > one)
82             z = one - two / (expm1(t + t) + two);
83         else if (t > small) {
84             y = expm1(-t - t);
85             z = -y / (y + two);
86         } else {
87 #ifndef lint
88             dummy = t + big;
89             /* inexact if t != 0 */
90 #endif
91             return (x);
92         }
93     } else if (!finitel(t))
94         return (copysignl(one, x));
95     else
96         return (signx ? -z + small * small : z - small * small);
97     return (signx ? -z : z);
98 }
99
100 _____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/Q/tanl.c

1

```
*****
2433 Tue Nov 25 12:57:40 2014
new/usr/src/lib/libm/common/Q/tanl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 /*
31  * tanl(x)
32  * Table look-up algorithm by K.C. Ng, November, 1989.
33  *
34  * kernel function:
35  *   __k_tanl    ... tangent function on [-pi/4,pi/4]
36  *   __rem_pio2l ... argument reduction routine
37  *
38  * Method.
39  *   Let S and C denote the sin and cos respectively on [-PI/4, +PI/4].
40  *   1. Assume the argument x is reduced to y1+y2 = x-k*pi/2 in
41  *   [-pi/2, +pi/2], and let n = k mod 4.
42  *   2. Let S=S(y1+y2), C=C(y1+y2). Depending on n, we have
43  *
44  *   n      sin(x)      cos(x)      tan(x)
45  *   -----
46  *   0      S           C           S/C
47  *   1      C           -S          -C/S
48  *   2      -S          -C          S/C
49  *   3      -C           S          -C/S
50  *   -----
51  *
52  * Special cases:
53  *   Let trig be any of sin, cos, or tan.
54  *   trig(++INF) is NaN, with signals;
55  *   trig(NaN) is that NaN;
56  *
57  * Accuracy:
58  *   computer TRIG(x) returns trig(x) nearly rounded.
```

new/usr/src/lib/libm/common/Q/tanl.c

2

```
59 */
60
61 #pragma weak __tanl = tanl
61 #pragma weak tanl = __tanl
62
63 #include "libm.h"
64 #include "longdouble.h"
65
66 long double
67 tanl(long double x) {
68     long double y[2], z = 0.0L;
69     int n, ix;
70
71     ix = *(int *) &x;          /* High word of x */
72     ix &= 0x7fffffff;
73     if (ix <= 0x3ffe9220)     /* |x| ~< pi/4 */
74         return (__k_tanl(x, z, 0));
75     else if (ix >= 0x7fff0000) /* trig(Inf or NaN) is NaN */
76         return (x - x);
77     else {                    /* argument reduction needed */
78         n = __rem_pio2l(x, y);
79         return (__k_tanl(y[0], y[1], (n & 1)));
80     }
81 }
82
83 _____
84 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/acosf.c

1

```
*****
1257 Tue Nov 25 12:57:40 2014
new/usr/src/lib/libm/common/R/acosf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __acosf = acosf
29 #pragma weak acosf = __acosf
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f;
```

```
35 float
```

```
36 acosf(float x) {
37     int ix;
```

```
39     ix = *(int *)&x & ~0x80000000;
```

```
40     if (ix > 0x3f800000) /* |x| > 1 or x is nan */
```

```
41         return ((x * zero) / zero);
```

```
42     return ((float)acos((double)x));
```

```
43 }
```

```
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/acoshf.c

1

1264 Tue Nov 25 12:57:41 2014

new/usr/src/lib/libm/common/R/acoshf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __acoshf = acoshf
```

```
29 #pragma weak acoshf = __acoshf
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f;
```

```
35 float
```

```
36 acoshf(float x) {
37     int    hx;
```

```
39     hx = *(int *)&x;
```

```
40     if (hx < 0x3f800000 || hx > 0x7f800000) /* x < 1 or x is nan */
```

```
41         return ((x * zero) / zero);
```

```
42     return ((float)acosh((double)x));
```

```
43 }
```

```
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/asinf.c

1

```
*****
1257 Tue Nov 25 12:57:41 2014
new/usr/src/lib/libm/common/R/asinf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __asinf = asinf
29 #pragma weak asinf = __asinf
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f;
```

```
35 float
```

```
36 asinf(float x) {
37     int    ix;
```

```
39     ix = *(int *)&x & ~0x80000000;
```

```
40     if (ix > 0x3f800000) /* |x| > 1 or x is nan */
```

```
41         return ((x * zero) / zero);
```

```
42     return ((float)asin((double)x));
```

```
43 }
```

```
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/asinhf.c

1

1187 Tue Nov 25 12:57:41 2014

new/usr/src/lib/libm/common/R/asinhf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __asinhf = asinhf
30 #pragma weak asinhf = __asinhf
```

```
32 #include "libm.h"
```

```
34 float
35 asinhf(float x) {
36     if (isnanf(x)) {
37         return (x * x);          /* + -> * for Cheetah */
38     } else {
39         return ((float) asinh((double) x));
40     }
41 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/R/atan2f.c

1

8324 Tue Nov 25 12:57:42 2014

new/usr/src/lib/libm/common/R/atan2f.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __atan2f = atan2f
```

```
29 #pragma weak atan2f = __atan2f
```

```
31 #include "libm.h"
```

```
33 #if defined(__i386) && !defined(__amd64)
```

```
34 extern int __swapRP(int);
```

```
35 #endif
```

```
37 /*
```

```
38 * For i = 0, ..., 192, let x[i] be the double precision number whose
```

```
39 * high order 32 bits are 0x3f900000 + (i << 16) and whose low order
```

```
40 * 32 bits are zero. Then TBL[i] := atan(x[i]) to double precision.
```

```
41 */
```

```
43 static const double TBL[] = {
```

```
44 1.56237286204768313e-02,
```

```
45 1.66000375562312640e-02,
```

```
46 1.75763148444955872e-02,
```

```
47 1.85525586258889763e-02,
```

```
48 1.95287670414137082e-02,
```

```
49 2.05049382324763683e-02,
```

```
50 2.14810703409090559e-02,
```

```
51 2.24571615089905717e-02,
```

```
52 2.34332098794675855e-02,
```

```
53 2.44092135955758099e-02,
```

```
54 2.53851708010611396e-02,
```

```
55 2.63610796402007873e-02,
```

```
56 2.73369382578244127e-02,
```

```
57 2.83127447993351995e-02,
```

new/usr/src/lib/libm/common/R/atan2f.c

2

```
58 2.92884974107309737e-02,
59 3.02641942386252458e-02,
60 3.12398334302682774e-02,
61 3.31909314971115949e-02,
62 3.51417768027967800e-02,
63 3.70923545503918164e-02,
64 3.90426499551669928e-02,
65 4.09926482452637811e-02,
66 4.29423346623621707e-02,
67 4.48916944623464972e-02,
68 4.68407129159696539e-02,
69 4.87893753095156174e-02,
70 5.07376669454602178e-02,
71 5.26855731431300420e-02,
72 5.46330792393594777e-02,
73 5.65801705891457105e-02,
74 5.85268325663017702e-02,
75 6.04730505641073168e-02,
76 6.24188099959573500e-02,
77 6.43088949198234884e-02,
78 6.63088949198234884e-02,
79 6.83088949198234884e-02,
80 7.03088949198234884e-02,
81 7.23088949198234884e-02,
82 7.43088949198234884e-02,
83 7.63088949198234884e-02,
84 7.83088949198234884e-02,
85 8.03088949198234884e-02,
86 8.23088949198234884e-02,
87 8.43088949198234884e-02,
88 8.63088949198234884e-02,
89 8.83088949198234884e-02,
90 9.03088949198234884e-02,
91 9.23088949198234884e-02,
92 9.43088949198234884e-02,
93 9.63088949198234884e-02,
94 9.83088949198234884e-02,
95 1.003088949198234884e-01,
96 1.023088949198234884e-01,
97 1.043088949198234884e-01,
98 1.063088949198234884e-01,
99 1.083088949198234884e-01,
100 1.103088949198234884e-01,
101 1.123088949198234884e-01,
102 1.143088949198234884e-01,
103 1.163088949198234884e-01,
104 1.183088949198234884e-01,
105 1.203088949198234884e-01,
106 1.223088949198234884e-01,
107 1.243088949198234884e-01,
108 1.263088949198234884e-01,
109 1.283088949198234884e-01,
110 1.303088949198234884e-01,
111 1.323088949198234884e-01,
112 1.343088949198234884e-01,
113 1.363088949198234884e-01,
114 1.383088949198234884e-01,
115 1.403088949198234884e-01,
116 1.423088949198234884e-01,
117 1.443088949198234884e-01,
118 1.463088949198234884e-01,
119 1.483088949198234884e-01,
120 1.503088949198234884e-01,
121 1.523088949198234884e-01,
122 1.543088949198234884e-01,
123 1.563088949198234884e-01,
```

```

124 4.63647609000806094e-01,
125 4.88333951056405535e-01,
126 5.12389460310737732e-01,
127 5.35811237960463704e-01,
128 5.58599315343562441e-01,
129 5.80756353567670414e-01,
130 6.02287346134964152e-01,
131 6.23199329934065904e-01,
132 6.43501108793284371e-01,
133 6.63202992706093286e-01,
134 6.82316554874748071e-01,
135 7.00854407884450192e-01,
136 7.18829999621624527e-01,
137 7.36257428981428097e-01,
138 7.53151280962194414e-01,
139 7.69526480405658297e-01,
140 7.85398163397448279e-01,
141 8.15691923316223422e-01,
142 8.44153986113171051e-01,
143 8.70903457075652976e-01,
144 8.96055384571343927e-01,
145 9.19719605350416858e-01,
146 9.42000040379463610e-01,
147 9.62994330680936206e-01,
148 9.82793723247329054e-01,
149 1.00148313569423464e+00,
150 1.01914134426634972e+00,
151 1.03584125300880014e+00,
152 1.05165021254837376e+00,
153 1.06663036531574362e+00,
154 1.08083900054116833e+00,
155 1.09432890732118993e+00,
156 1.10714871779409041e+00,
157 1.13095374397916038e+00,
158 1.15257199721566761e+00,
159 1.17227388112847630e+00,
160 1.19028994968253166e+00,
161 1.20681737028525249e+00,
162 1.22202532321098967e+00,
163 1.23605948947808186e+00,
164 1.24904577239825443e+00,
165 1.26109338225244039e+00,
166 1.27229739520871732e+00,
167 1.28274087974427076e+00,
168 1.29249666778978534e+00,
169 1.30162883400919616e+00,
170 1.31019393504755555e+00,
171 1.31824205101683711e+00,
172 1.32581766366803255e+00,
173 1.33970565959899957e+00,
174 1.35212738092095464e+00,
175 1.36330010035969384e+00,
176 1.37340076694501589e+00,
177 1.38257482149012589e+00,
178 1.39094282700241845e+00,
179 1.39860551227195762e+00,
180 1.40564764938026987e+00,
181 1.41214106460849531e+00,
182 1.41814699839963154e+00,
183 1.42371797140649403e+00,
184 1.42889927219073276e+00,
185 1.43373015248470903e+00,
186 1.43824479449822262e+00,
187 1.44247309910910193e+00,
188 1.44644133224813509e+00,
189 1.45368758222803240e+00,

```

```

190 1.46013910562100091e+00,
191 1.46591938806466282e+00,
192 1.47112767430373470e+00,
193 1.47584462045214027e+00,
194 1.48013643959415142e+00,
195 1.48405798811891154e+00,
196 1.48765509490645531e+00,
197 1.49096634108265924e+00,
198 1.49402443552511865e+00,
199 1.49685728913695626e+00,
200 1.49948886200960629e+00,
201 1.50193983749385196e+00,
202 1.50422816301907281e+00,
203 1.50636948736934317e+00,
204 1.50837751679893928e+00,
205 1.51204050407917401e+00,
206 1.51529782154917969e+00,
207 1.51821326518395483e+00,
208 1.52083793107295384e+00,
209 1.52321322351791322e+00,
210 1.52537304737331958e+00,
211 1.52734543140336587e+00,
212 1.52915374769630819e+00,
213 1.53081763967160667e+00,
214 1.53235373677370856e+00,
215 1.53377621092096650e+00,
216 1.53509721411557254e+00,
217 1.53632722579538861e+00,
218 1.53747533091664934e+00,
219 1.53854944435964280e+00,
220 1.53955649336462841e+00,
221 1.54139303859089161e+00,
222 1.54302569020147562e+00,
223 1.54448660954197448e+00,
224 1.54580153317597646e+00,
225 1.54699130060982659e+00,
226 1.54807296595325550e+00,
227 1.54906061995310385e+00,
228 1.54996600675867957e+00,
229 1.55079899282174605e+00,
230 1.55156792769518947e+00,
231 1.55227992472688747e+00,
232 1.55294108165534417e+00,
233 1.55355665560036682e+00,
234 1.55413120308095598e+00,
235 1.55466869295126031e+00,
236 1.55517259817441977e+00,
237 };

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/atan2pif.c

1

```
*****
1473 Tue Nov 25 12:57:42 2014
new/usr/src/lib/libm/common/R/atan2pif.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak atan2pif = __atan2pif
```

```
29 #include "libm.h"
```

```
31 static const double invpi = 0.3183098861837906715377675;
```

```
33 float
34 atan2pif(float y, float x) {
35     int    ix, iy, hx, hy;
36
37     ix = *(int *)&x;
38     iy = *(int *)&y;
39     hx = ix & ~0x80000000;
40     hy = iy & ~0x80000000;
41     if (hx > 0x7f800000 || hy > 0x7f800000) /* x or y is nan */
42         return (x * y);
43     if ((hx | hy) == 0) {
44         /* x and y are both zero */
45         if (ix == 0)
46             return (y);
47         return ((iy == 0)? 1.0f : -1.0f);
48     }
49     return ((float)(invpi * atan2((double)y, (double)x)));
50 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/atanf.c

1

```
*****
5444 Tue Nov 25 12:57:43 2014
new/usr/src/lib/libm/common/R/atanf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
29
30 #pragma weak __atanf = atanf
30 #pragma weak atanf = __atanf
31
32 /* INDENT OFF */
33 /*
34  * float atanf(float x);
35  * Table look-up algorithm
36  * By K.C. Ng, March 9, 1989
37  *
38  * Algorithm.
39  *
40  * The algorithm is based on atan(x)=atan(y)+atan((x-y)/(1+x*y)).
41  * We use poly1(x) to approximate atan(x) for x in [0,1/8] with
42  * error (relative)
43  * |atan(x)-poly1(x)|/x <= 2^-115.94 long double
44  * |atan(x)-poly1(x)|/x <= 2^-58.85 double
45  * |atan(x)-poly1(x)|/x <= 2^-25.53 float
46  * and use poly2(x) to approximate atan(x) for x in [0,1/65] with
47  * error (absolute)
48  * |atan(x)-poly2(x)| <= 2^-122.15 long double
49  * |atan(x)-poly2(x)| <= 2^-64.79 double
50  * |atan(x)-poly2(x)| <= 2^-35.36 float
51  * and use poly3(x) to approximate atan(x) for x in [1/8,7/16] with
52  * error (relative, on for single precision)
53  * |(atan(x)-poly1(x))/x| <= 2^-25.53 float
54  *
55  * Here poly1-3 are odd polynomial with the following form:
56  * x + x^3*(a1+x^2*(a2+...))
57  */
```

new/usr/src/lib/libm/common/R/atanf.c

2

```
58 * (0). Purge off Inf and NaN and 0
59 * (1). Reduce x to positive by atan(x) = -atan(-x).
60 * (2). For x <= 1/8, use
61 * (2.1) if x < 2^(-prec/2-2), atan(x) = x with inexact
62 * (2.2) Otherwise
63 * atan(x) = poly1(x)
64 * (3). For x >= 8 then
65 * (3.1) if x >= 2^(prec+2), atan(x) = atan(1/x) - pio2lo
66 * (3.2) if x >= 2^(prec/3+2), atan(x) = atan(1/x) - 1/x
67 * (3.3) if x > 65, atan(x) = atan(1/x) - poly2(1/x)
68 * (3.4) Otherwise, atan(x) = atan(1/x) - poly1(1/x)
69 *
70 * (4). Now x is in (0.125, 8)
71 * Find y that match x to 4.5 bit after binary (easy).
72 * If iy is the high word of y, then
73 * single : j = (iy - 0x3e000000) >> 19
74 * (single is modified to (iy-0x3f000000)>>19)
75 * double : j = (iy - 0x3fc00000) >> 16
76 * quad : j = (iy - 0x3ffc0000) >> 12
77 *
78 * Let s = (x-y)/(1+x*y). Then
79 * atan(x) = atan(y) + poly1(s)
80 * = _TBL_r_atan_hi[j] + (_TBL_r_atan_lo[j] + poly2(s))
81 *
82 * Note. |s| <= 1.5384615385e-02 = 1/65. Maxium occurs at x = 1.03125
83 *
84 */
85
86 #include "libm.h"
87
88 extern const float _TBL_r_atan_hi[], _TBL_r_atan_lo[];
89 static const float
90 big = 1.0e37F,
91 one = 1.0F,
92 p1 = -3.333185951111688247225368498733544672172e-0001F,
93 p2 = 1.969352894213455405211341983203180636021e-0001F,
94 q1 = -3.332921964095646819563419704110132937456e-0001F,
95 a1 = -3.333323465223893614063523351509338934592e-0001F,
96 a2 = 1.999425625935277805494082274808174062403e-0001F,
97 a3 = -1.417547090509737780085769846290301788559e-0001F,
98 a4 = 1.016250813871991983097273733227432685084e-0001F,
99 a5 = -5.137023693688358515753093811791755221805e-0002F,
100 pio2hi = 1.570796371e+0000F,
101 pio2lo = -4.371139000e-0008F;
102 /* INDENT ON */
103
104 float
105 atanf(float xx) {
106 float x, y, z, r, p, s;
107 volatile double dummy;
108 int ix, iy, sign, j;
109
110 x = xx;
111 ix = *(int *) &x;
112 sign = ix & 0x80000000;
113 ix ^= sign;
114
115 /* for |x| < 1/8 */
116 if (ix < 0x3e000000) {
117 if (ix < 0x38800000) { /* if |x| < 2**(-prec/2-2) */
118 dummy = big + x; /* get inexact flag if x != 0 */
119 #ifdef lint
120 dummy = dummy;
121 #endif
122 return (x);
123 }
124 }
```

```

124     z = x * x;
125     if (ix < 0x3c000000) { /* if |x| < 2**(-prec/4-1) */
126         x = x + (x * z) * p1;
127         return (x);
128     } else {
129         x = x + (x * z) * (p1 + z * p2);
130         return (x);
131     }
132 }
133
134 /* for |x| >= 8.0 */
135 if (ix >= 0x41000000) {
136     *(int *) &x = ix;
137     if (ix < 0x42820000) { /* x < 65 */
138         r = one / x;
139         z = r * r;
140         y = r * (one + z * (p1 + z * p2)); /* poly1 */
141         y -= pio2lo;
142     } else if (ix < 0x44800000) { /* x < 2**(prec/3+2) */
143         r = one / x;
144         z = r * r;
145         y = r * (one + z * q1); /* poly2 */
146         y -= pio2lo;
147     } else if (ix < 0x4c800000) { /* x < 2**(prec+2) */
148         y = one / x - pio2lo;
149     } else if (ix < 0x7f800000) { /* x < inf */
150         y = -pio2lo;
151     } else { /* x is inf or NaN */
152         if (ix > 0x7f800000) {
153             return (x * x); /* - -> * for Cheetah */
154         }
155         y = -pio2lo;
156     }
157
158     if (sign == 0)
159         x = pio2hi - y;
160     else
161         x = y - pio2hi;
162     return (x);
163 }
164
165
166 /* now x is between 1/8 and 8 */
167 if (ix < 0x3f000000) { /* between 1/8 and 1/2 */
168     z = x * x;
169     x = x + (x * z) * (a1 + z * (a2 + z * (a3 + z * (a4 +
170         z * a5))));
171     return (x);
172 }
173 *(int *) &x = ix;
174 iy = (ix + 0x00040000) & 0x7ff80000;
175 *(int *) &y = iy;
176 j = (iy - 0x3f000000) >> 19;
177
178 if (ix == iy)
179     p = x - y; /* p=0.0 */
180 else {
181     if (sign == 0)
182         s = (x - y) / (one + x * y);
183     else
184         s = (y - x) / (one + x * y);
185     z = s * s;
186     p = s * (one + z * q1);
187 }
188 if (sign == 0) {
189     r = p + _TBL_r_atan_lo[j];

```

```

190         x = r + _TBL_r_atan_hi[j];
191     } else {
192         r = p - _TBL_r_atan_lo[j];
193         x = r - _TBL_r_atan_hi[j];
194     }
195     return (x);
196 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/atanhf.c

1

1322 Tue Nov 25 12:57:43 2014

new/usr/src/lib/libm/common/R/atanhf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __atanhf = atanhf
```

```
29 #pragma weak atanhf = __atanhf
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f;
```

```
35 float
```

```
36 atanhf(float x) {
37     int ix;
```

```
39     ix = *((int *)&x) & ~0x80000000;
40     if (ix > 0x3f800000) /* |x| > 1 or x is nan */
41         return ((x * zero) / zero);
42     if (ix == 0x3f800000) /* |x| == 1 */
43         return (x / zero);
44     return ((float)atanh((double)x));
45 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/besself.c

1

```
*****
20641 Tue Nov 25 12:57:43 2014
new/usr/src/lib/libm/common/R/besself.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __j0f = j0f
30 #pragma weak __j1f = j1f
31 #pragma weak __jnf = jnf
32 #pragma weak __y0f = y0f
33 #pragma weak __y1f = y1f
34 #pragma weak __ynf = ynf
29 #pragma weak j0f = __j0f
30 #pragma weak j1f = __j1f
31 #pragma weak jnf = __jnf
32 #pragma weak y0f = __y0f
33 #pragma weak y1f = __y1f
34 #pragma weak ynf = __ynf
```

```
36 #include "libm.h"
37 #include <float.h>

39 #if defined(__i386) && !defined(__amd64)
40 extern int __swapRP(int);
41 #endif

43 static const float
44     zerof = 0.0f,
45     onef = 1.0f;

47 static const double C[] = {
48     0.0,
49     -0.125,
50     0.25,
51     0.375,
52     0.5,
```

new/usr/src/lib/libm/common/R/besself.c

2

```
53     1.0,
54     2.0,
55     8.0,
56     0.5641895835477562869480794515607725858441, /* 1/sqrt(pi) */
57     0.636619772367581343075535053490057448, /* 2/pi */
58     1.0e9,
59 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/cbrtf.c

1

```
*****
1197 Tue Nov 25 12:57:45 2014
new/usr/src/lib/libm/common/R/cbrtf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cbrtf = cbrtf
30 #pragma weak cbrtf = __cbrtf
```

```
32 #include "libm.h"
```

```
34 float
35 cbrtf(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) cbrt((double) x));
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/copysignf.c

1

1189 Tue Nov 25 12:57:45 2014

new/usr/src/lib/libm/common/R/copysignf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __copysignf = copysignf
30 #if defined(ELFOSBJ)
31 #pragma weak copysignf = __copysignf
32 #endif
33
34 #include "libm.h"
35
36 float
37 copysignf(float x, float y) {
38     float w;
39
40     *(int *) &w = (*(int *) &x & ~0x80000000) | (*(int *) &y & 0x80000000);
41     return (w);
42 }
43
44 _____
45 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/cosf.c

1

```
*****
3872 Tue Nov 25 12:57:45 2014
new/usr/src/lib/libm/common/R/cosf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __cosf = cosf
29 #pragma weak cosf = __cosf
```

```
31 /*
32 * See sincosf.c
33 */
```

```
35 #include "libm.h"
```

```
37 extern const int _TBL_ipio2_inf[];
38 extern int __rem_pio2m(double *, double *, int, int, int, const int *);
39 #if defined(__i386) && !defined(__amd64)
40 extern int __swapRP(int);
41 #endif
```

```
43 static const double C[] = {
44     1.85735322054308378716204874632872525989806770558e-0003,
45     -1.95035094218403635082921458859320791358115801259e-0004,
46     5.38400550766074785970952495168558701485841707252e+0002,
47     -3.31975110777873728964197739157371509422022905947e+0001,
48     1.09349482127188401868272000389539985058873853699e-0003,
49     -5.0332428598964979398034700054920226866107675091e-0004,
50     2.43792880266971107750418061559602239831538067410e-0005,
51     9.14499072605666582228127405245558035523741471271e+0002,
52     -3.63151270591815439197122504991683846785293207730e+0001,
53     0.636619772367581343075535, /* 2^-1 * 1.45F306DC9C883 */
54     0.5,
55     1.570796326734125614166, /* 2^0 * 1.921FB54400000 */
56     6.077100506506192601475e-11, /* 2^-34 * 1.0B4611A626331 */
57 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/coshf.c

1

```
*****
1354 Tue Nov 25 12:57:46 2014
new/usr/src/lib/libm/common/R/coshf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __coshf = coshf
29 #pragma weak coshf = __coshf
```

```
31 #include "libm.h"
```

```
33 float
34 coshf(float x) {
35     double c;
36     float w;
37     int ix;

39     ix = *(int *)&x & ~0x80000000;
40     if (ix >= 0x7f800000) {
41         /* coshf(x) is |x| if x is +-Inf or NaN */
42         return (x * x);
43     }
44     if (ix >= 0x43000000) /* coshf(x) trivially overflows */
45         c = 1.0e100;
46     else
47         c = cosh((double)x);
48     w = (float)c;
49     return (w);
50 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/erff.c

1

1706 Tue Nov 25 12:57:46 2014

new/usr/src/lib/libm/common/R/erff.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __erff = erff
30 #pragma weak __erfcf = erfcf
29 #pragma weak erff = __erff
30 #pragma weak erfcf = __erfcf
```

```
32 #include "libm.h"
```

```
34 #if defined(__i386) && !defined(__amd64)
35 extern int __swapRP(int);
36 #endif
```

```
38 float
39 erff(float x) {
40     int ix;

42     ix = *(int *)&x & ~0x80000000;
43     if (ix > 0x7f800000) /* x is NaN */
44         return (x * x);
45     return ((float)erf((double)x));
46 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/R/exp10f.c

1

1199 Tue Nov 25 12:57:47 2014

new/usr/src/lib/libm/common/R/exp10f.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak exp10f = __exp10f
```

```
30 #include "libm.h"
```

```
32 extern double exp10(double);
```

```
34 float
35 exp10f(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) exp10((double) x));
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/exp2f.c

1

```
*****
1197 Tue Nov 25 12:57:47 2014
new/usr/src/lib/libm/common/R/exp2f.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __exp2f = exp2f
30 #pragma weak exp2f = __exp2f
```

```
32 #include "libm.h"
```

```
34 float
35 exp2f(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) exp2((double) x));
42 }
```

unchanged_portion_omitted

```

*****
16892 Tue Nov 25 12:57:48 2014
new/usr/src/lib/libm/common/R/expf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29 #pragma weak __expf = expf
29 #pragma weak expf = __expf

31 /* INDENT OFF */
32 /*
33  * float expf(float x);
34  * Code by K.C. Ng for SUN 5.0 libmopt
35  * 11/5/99
36  * Method :
37  *   1. For |x| >= 2^7, either underflow/overflow.
38  *      More precisely:
39  *         x > 88.722839355...(0x42B17218) => overflow;
40  *         x < -103.97207642...(0xc2CFF1B4) => underflow.
41  *   2. For |x| < 2^-6, use polynomial
42  *         exp(x) = 1 + x + p1*x^2 + p2*x^3
43  *   3. Otherwise, write |x|=(1+r)*2^n, where 0<r<1.
44  *         Let t = 2^n * (1+r) .... x > 0;
45  *             t = 2^n * (1-r) .... x < 0. (x = -2**(n+1)+t)
46  *         Since -6 <= n <= 6, we may break t into
47  *         six 6-bits chunks:
48  *             -5      -11      -17      -23      -29
49  *         t=j *2+j *2 +j *2 +j *2 +j *2 +j *2
50  *             1      2      3      4      5      6
51  *
52  *         where 0 <= j < 64 for i = 1,...,6.
53  *
54  *         Note that since t has only 24 significant bits,
55  *         either j or j must be 0.
56  *             1      6
57  *

```

```

58  *   One may define j by (int) ( t * 2 ) mod 64
59  *
60  *   mathematically. In actual implementation, they can
61  *   be obtained by manipulating the exponent and
62  *   mantissa bits as follow:
63  *   Let ix = (HEX(x)&0x007fffff)|0x00800000.
64  *   If n>=0, let ix=ix<<n, then j =0 and
65  *
66  *           j = ix>>(30-6i) mod 64 ...i=1,...,5
67  *
68  *   Otherwise, let ix=ix<<(j+6), then j = 0 and
69  *
70  *           j = ix>>(36-6i) mod 64 ...i=2,...,6
71  *
72  *
73  *   4. Compute exp(t) by table look-up method.
74  *   Precompute ET[k] = exp(j*2^(7-6i)), k=j+64*(6-i).
75  *   Then
76  *   exp(t) = ET[j +320]*ET[j +256]*ET[j +192]*
77  *           1      2      3
78  *           ET[j +128]*ET[j +64]*ET[j ]
79  *           4      5      6
80  *
81  *
82  *           n+1
83  *   5. If x < 0, return exp(-2 )* exp(t). Note that
84  *   -6 <= n <= 6. Let k = n - 6, then we can
85  *   precompute
86  *           k-5      n+1
87  *   EN[k] = exp(-2 ) = exp(-2 ) for k=0,1,...,12.
88  *
89  *
90  * Special cases:
91  *   exp(INF) is INF, exp(NaN) is NaN;
92  *   exp(-INF) = 0;
93  *   for finite argument, only exp(0) = 1 is exact.
94  *
95  * Accuracy:
96  *   All calculations are done in double precision except for
97  *   the case |x| < 2^-6. When |x| < 2^-6, the error is less
98  *   than 0.55 ulp. When |x| >= 2^-6 and the result is normal,
99  *   the error is less than 0.51 ulp. When FDTOS_TRAPS_... is
100 *   defined and the result is subnormal, the error can be as
101 *   large as 0.75 ulp.
102 */
103 /* INDENT ON */

105 #include "libm.h"

107 /*
108  * ET[k] = exp(j*2^(7-6i)) , where j = k mod 64, i = k/64
109  */
110 static const double ET[] = {
111 1.000000000000000000000000e+00, 1.00000000186264514923e+00,
112 1.00000000372529029846e+00, 1.00000000558793544769e+00,
113 1.00000000745058059692e+00, 1.00000000931322574615e+00,
114 1.00000001117587089539e+00, 1.0000000130385160462e+00,
115 1.00000001490116119385e+00, 1.00000001676380656512e+00,
116 1.00000001862645171435e+00, 1.00000002048909686359e+00,
117 1.00000002235174201282e+00, 1.00000002421438716205e+00,
118 1.00000002607703253332e+00, 1.00000002793967768255e+00,
119 1.00000002980232283178e+00, 1.00000003166496798102e+00,
120 1.00000003352761335229e+00, 1.00000003539025850152e+00,
121 1.00000003725290365075e+00, 1.00000003911554879998e+00,
122 1.00000004097819417126e+00, 1.00000004284083932049e+00,
123 1.00000004470348446972e+00, 1.00000004656612984100e+00,

```

124 1.00000004842877499023e+00, 1.00000005029142036150e+00,
125 1.00000005215406551073e+00, 1.00000005401671088201e+00,
126 1.00000005587935603124e+00, 1.00000005774200140252e+00,
127 1.00000005960464655175e+00, 1.00000006146729192302e+00,
128 1.00000006332993707225e+00, 1.00000006519258244353e+00,
129 1.00000006705522759276e+00, 1.00000006891787296404e+00,
130 1.00000007078051811327e+00, 1.00000007264316348454e+00,
131 1.00000007450580863377e+00, 1.00000007636845400505e+00,
132 1.00000007823109937632e+00, 1.00000008009374452556e+00,
133 1.00000008195638989683e+00, 1.00000008381903526811e+00,
134 1.00000008568168063938e+00, 1.00000008754432578861e+00,
135 1.00000008940697115989e+00, 1.00000009126961653116e+00,
136 1.00000009313226190244e+00, 1.00000009499490705167e+00,
137 1.00000009685755242295e+00, 1.00000009872019779422e+00,
138 1.00000010058284316550e+00, 1.00000010244548853677e+00,
139 1.00000010430813368600e+00, 1.00000010617077905728e+00,
140 1.00000010803342442856e+00, 1.00000010989606979983e+00,
141 1.00000011175871517111e+00, 1.00000011362136054238e+00,
142 1.00000011548400591366e+00, 1.00000011734665128493e+00,
143 1.0000000000000000000e+00, 1.00000011920929665621e+00,
144 1.00000023841860752327e+00, 1.00000035762793260119e+00,
145 1.00000047683727188996e+00, 1.00000059604662538959e+00,
146 1.00000071525599310007e+00, 1.00000083446537502141e+00,
147 1.00000095367477115360e+00, 1.00000107288418149665e+00,
148 1.00000119209360605055e+00, 1.00000131130304481530e+00,
149 1.00000143051249779091e+00, 1.00000154972196497738e+00,
150 1.00000166893144637470e+00, 1.00000178814094198287e+00,
151 1.00000190735045180190e+00, 1.00000202655997583179e+00,
152 1.00000214576951407253e+00, 1.00000226497906652412e+00,
153 1.00000238418863318657e+00, 1.00000250339821405987e+00,
154 1.00000262260780914403e+00, 1.00000274181741843904e+00,
155 1.00000286102704194491e+00, 1.00000298023667966163e+00,
156 1.00000309944633158921e+00, 1.00000321865599772764e+00,
157 1.00000333786567807692e+00, 1.00000345707537263706e+00,
158 1.00000357628508140806e+00, 1.00000369549480438991e+00,
159 1.00000381470454158261e+00, 1.00000393391429298617e+00,
160 1.00000405312405860059e+00, 1.00000417233383842586e+00,
161 1.00000429154363246198e+00, 1.00000441075344070896e+00,
162 1.00000452996326316679e+00, 1.00000464917309983548e+00,
163 1.00000476838295071502e+00, 1.00000488759281580542e+00,
164 1.00000500680269510667e+00, 1.00000512601258861878e+00,
165 1.00000524522249634174e+00, 1.00000536443241827556e+00,
166 1.00000548364235442023e+00, 1.00000560285230477575e+00,
167 1.0000057206226934213e+00, 1.00000584127224811937e+00,
168 1.00000596048224110746e+00, 1.00000607969224830640e+00,
169 1.00000619890226971620e+00, 1.00000631811230533685e+00,
170 1.00000643732235516836e+00, 1.00000655653241921073e+00,
171 1.00000667574249746394e+00, 1.00000679495258992802e+00,
172 1.00000691416269660294e+00, 1.00000703337281748873e+00,
173 1.00000715258295258536e+00, 1.00000727179310189285e+00,
174 1.000007391000326541120e+00, 1.00000751021344314040e+00,
175 1.0000000000000000000e+00, 1.00000762942363508046e+00,
176 1.00001525890547848796e+00, 1.00002288844553022251e+00,
177 1.00003051804379095024e+00, 1.00003814770026133729e+00,
178 1.00004577741494138365e+00, 1.00005340718783175546e+00,
179 1.00006103701893311886e+00, 1.00006866690824547383e+00,
180 1.00007629685576948653e+00, 1.00008392686150582307e+00,
181 1.00009155692545448346e+00, 1.00009918704761613384e+00,
182 1.00010687122799144033e+00, 1.00011444746658040295e+00,
183 1.0001220776338368781e+00, 1.00012970811840196106e+00,
184 1.00013733853163522269e+00, 1.00014496900308413885e+00,
185 1.00015259953274937565e+00, 1.00016023012063093311e+00,
186 1.00016786076672947736e+00, 1.00017549147104567453e+00,
187 1.00018312223357952462e+00, 1.00019075305433191581e+00,
188 1.0001983839330284809e+00, 1.00020601487049298761e+00,
189 1.00021364586590300050e+00, 1.00022127691953288675e+00,

190 1.00022890803138353455e+00, 1.00023653920145494389e+00,
191 1.00024417042974778091e+00, 1.00025180171626271175e+00,
192 1.00025943306099973640e+00, 1.00026706446395974304e+00,
193 1.00027469592514273167e+00, 1.00028232744454959047e+00,
194 1.00028995902218031944e+00, 1.00029759065803558471e+00,
195 1.00030522235211605242e+00, 1.00031285410442172257e+00,
196 1.00032048591495348333e+00, 1.00032811778371155675e+00,
197 1.000335749710699616488e+00, 1.00034338169590819589e+00,
198 1.00035101373934764979e+00, 1.00035864584101541475e+00,
199 1.00036627800091149076e+00, 1.00037391021903676602e+00,
200 1.00038154249539146257e+00, 1.00038917482997580244e+00,
201 1.00039680722279067382e+00, 1.00040443967383629875e+00,
202 1.00041207218311289928e+00, 1.00041970475062136359e+00,
203 1.000427373737636191371e+00, 1.00043497006033499375e+00,
204 1.00044260280254104778e+00, 1.00045023560298029786e+00,
205 1.00045786846165363215e+00, 1.00046550137856127272e+00,
206 1.00047313435370366363e+00, 1.00048076738708124900e+00,
207 1.0000000000000000000e+00, 1.00048840047869447289e+00,
208 1.00097703949241645383e+00, 1.00146591715766675179e+00,
209 1.00195503359100279717e+00, 1.00244438890903908579e+00,
210 1.00293398322844673487e+00, 1.00342381666595459322e+00,
211 1.00391388933834746489e+00, 1.00440420136246855165e+00,
212 1.0048947528521656645e+00, 1.0053855439354861993e+00,
213 1.00587657471447822211e+00, 1.00636784531507639251e+00,
214 1.00685935585247099411e+00, 1.00735110644384739942e+00,
215 1.00784309720644804642e+00, 1.00833532825757243856e+00,
216 1.00882779971457803292e+00, 1.00932051169487890796e+00,
217 1.00981346431594687374e+00, 1.01030665769531102782e+00,
218 1.0108009195055753324e+00, 1.01129376719933050666e+00,
219 1.01178768355933157430e+00, 1.01228184114831898377e+00,
220 1.01277624008410960244e+00, 1.01327088048457714109e+00,
221 1.01376576246765282008e+00, 1.01426088615132625748e+00,
222 1.01475625165364347069e+00, 1.01525185909270931894e+00,
223 1.01574770858668572693e+00, 1.01624380025379235093e+00,
224 1.01674013421230657883e+00, 1.01723671058056375216e+00,
225 1.01773352947695694404e+00, 1.01823059101993673714e+00,
226 1.01872789532801233392e+00, 1.01922544251975000229e+00,
227 1.01972323271377418585e+00, 1.02022126602876750390e+00,
228 1.02071954258347008526e+00, 1.02121806249668067856e+00,
229 1.02171682588725554197e+00, 1.0222184461695410910934e+00,
230 1.02271508357621376817e+00, 1.02321457811260052573e+00,
231 1.02371431660235789884e+00, 1.02421429916463280207e+00,
232 1.02471452591863054771e+00, 1.02521499698361440167e+00,
233 1.02571571247890602763e+00, 1.02621667252388526492e+00,
234 1.02671787723799012859e+00, 1.02721932674071725344e+00,
235 1.02772102115162167202e+00, 1.02822296059031659254e+00,
236 1.02872514517647339893e+00, 1.02922757502982276101e+00,
237 1.02973025027015285815e+00, 1.030233171017310933959e+00,
238 1.03073633739120262831e+00, 1.03123974951179242510e+00,
239 1.0000000000000000000e+00, 1.03174340749910276038e+00,
240 1.06449445891785954288e+00, 1.09828514030782575794e+00,
241 1.13314845306682632220e+00, 1.1691844616950433284e+00,
242 1.20623024942098067136e+00, 1.24452010776609522935e+00,
243 1.28402541668774139438e+00, 1.32478475872886569675e+00,
244 1.36683794117379631139e+00, 1.41022603492571074746e+00,
245 1.45499141461820125087e+00, 1.50117780000012279729e+00,
246 1.54883029863413312910e+00, 1.5979954995063325104e+00,
247 1.64872127070012819416e+00, 1.70105730184840076014e+00,
248 1.75505465696029849809e+00, 1.81076607211938722664e+00,
249 1.8682459574322232613e+00, 1.9275504501675467113e+00,
250 1.98873746958229191684e+00, 2.05186677348797674725e+00,
251 2.11700001661267478426e+00, 2.18420081081561789915e+00,
252 2.25353478712320854561e+00, 2.32506966027712103084e+00,
253 2.39887529396709808793e+00, 2.47502376996302508871e+00,
254 2.55358945806292680913e+00, 2.63464908881563086851e+00,
255 2.71828182845904553488e+00, 2.80456935623722669604e+00,

```

256 2.89359594417176113623e+00, 2.98544853936535581340e+00,
257 3.08021684891803104733e+00, 3.17799342753883840018e+00,
258 3.27887376793867346692e+00, 3.38295639409246895468e+00,
259 3.49034295746184142217e+00, 3.60113833627217561073e+00,
260 3.71545073794110392029e+00, 3.83339180475841034834e+00,
261 3.95507672292057721464e+00, 4.08062433502646015882e+00,
262 4.21015725614395996956e+00, 4.34380199356104235164e+00,
263 4.48168907033806451778e+00, 4.62395315278208052234e+00,
264 4.77073318196760265408e+00, 4.92217250943229078786e+00,
265 5.07841903718008147450e+00, 5.23962536212848917216e+00,
266 5.40594892514116676097e+00, 5.57755216479125959239e+00,
267 5.75460267600573072144e+00, 5.93727337374560715233e+00,
268 6.12574266188198635064e+00, 6.32019460743274397174e+00,
269 6.52081912033011246166e+00, 6.72781213889469142941e+00,
270 6.94137582219703555605e+00, 7.16171874249371143151e+00,
271 1.00000000000000000000e+00, 7.38905609893065040694e+00,
272 5.45981500331442362040e+01, 4.03428793492735110249e+02,
273 2.98095798704172830185e+03, 2.20264657948067178950e+04,
274 1.62754791419003915507e+05, 1.20260428416477679275e+06,
275 8.88611052050787210464e+06, 6.56599691373305097222e+07,
276 4.85165195409790277481e+08, 3.58491284613159179688e+09,
277 2.64891221298434715271e+10, 1.95729609428838775635e+11,
278 1.44625706429147509766e+12, 1.06864745815244628906e+13,
279 7.89629601826806875000e+13, 5.83461742527454875000e+14,
280 4.31123154711519500000e+15, 3.18559317571137560000e+16,
281 2.35385266837020000000e+17, 1.73927494152050099200e+18,
282 1.28516001143593082880e+19, 9.49611942060244828160e+19,
283 7.01673591209763143680e+20, 5.18470552858707204506e+21,
284 3.83100800071657691546e+22, 2.83075330327469394756e+23,
285 2.09165949601299610311e+24, 1.54553893559010391826e+25,
286 1.14200738981568423454e+26, 8.43835666874145383188e+26,
287 6.23514908081161674391e+27, 4.60718663433129178064e+28,
288 3.40427604993174075827e+29, 2.51543867091916687979e+30,
289 1.85867174528412788702e+31, 1.37338297954017610775e+32,
290 1.01480038811388874615e+33, 7.49841699699012090701e+33,
291 5.54062238439350983445e+34, 4.09399696212745451138e+35,
292 3.02507732220114256223e+36, 2.23524660373471497416e+37,
293 1.65163625499400180987e+38, 1.22040329431784083418e+39,
294 9.01762840503429851945e+39, 6.66317621641089618500e+40,
295 4.92345828601205826106e+41, 3.63797094760880474988e+42,
296 2.68811714181613560943e+43, 1.98626483613765434356e+44,
297 1.46766223015544238535e+45, 1.08446385529002313207e+46,
298 8.01316426400059069850e+46, 5.92097202766466993617e+47,
299 4.37503944726134096988e+48, 3.23274119108485947460e+49,
300 2.38869060142499127023e+50, 1.76501688569176554670e+51,
301 1.30418087839363225614e+52, 9.63666567360320166416e+52,
302 7.12058632688933793173e+53, 5.26144118266638596909e+54,
303 };

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/expm1f.c

1

1201 Tue Nov 25 12:57:48 2014

new/usr/src/lib/libm/common/R/expm1f.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __expm1f = expm1f
30 #pragma weak expm1f = __expm1f
```

```
32 #include "libm.h"
```

```
34 float
35 expm1f(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) expm1((double) x));
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/fabsf.c

1

```
*****
1113 Tue Nov 25 12:57:48 2014
new/usr/src/lib/libm/common/R/fabsf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __fabsf = fabsf
30 #pragma weak fabsf = __fabsf

32 #include "libm.h"

34 float
35 fabsf(float x) {
36     *(int *) &x &= ~0x80000000;
37     return (x);
38 }
unchanged_portion_omitted
```


new/usr/src/lib/libm/common/R/floorf.c

1

2643 Tue Nov 25 12:57:49 2014

new/usr/src/lib/libm/common/R/floorf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ceilf = ceilf
31 #pragma weak __floorf = floorf
30 #pragma weak ceilf = __ceilf
31 #pragma weak floorf = __floorf
```

```
33 /* INDENT OFF */
34 /*
35  * ceilf(x)    return the biggest integral value (in float) below x
36  * floorf(x)   return the least integral value (in float) above x
37  *
38  * NOTE: ceilf(x) and floorf(x) return result
39  * with the same sign as x's, including 0.0F.
40 */
```

```
42 #include "libm.h"
```

```
44 static const float xf[] = {
45 /* ZEROF */    0.0f,
46 /* ONEF */    1.0f,
47 /* MONEF */   -1.0f,
48 /* HUGEFP */  1.0e30f,
49 };
```

unchanged_portion_omitted

```

*****
3443 Tue Nov 25 12:57:49 2014
new/usr/src/lib/libm/common/R/fmodf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __fmodf = fmodf
29 #pragma weak fmodf = __fmodf

31 #include "libm.h"

33 /* INDENT OFF */
34 static const int
35     is = (int)0x80000000,
36     im = 0x007fffff,
37     ii = 0x7f800000,
38     iu = 0x00800000;
39 /* INDENT ON */

41 static const float zero = 0.0;

43 float
44 fmodf(float x, float y) {
45     float    w;
46     int      hx, ix, iy, iz, k, ny, nd;

48     hx = *(int *)&x;
49     ix = hx & 0x7fffffff;
50     iy = *(int *)&y & 0x7fffffff;

52     /* purge off exception values */
53     if (ix >= ii || iy > ii || iy == 0) {
54         w = x * y;
55         w = w / w;
56     } else if (ix <= iy) {
57         if (ix < iy)

```

```

58         w = x; /* return x if |x|<|y| */
59     else
60         w = zero * x; /* return sign(x)*0.0 */
61 } else {
62     /* INDENT OFF */
63     /*
64      * scale x,y to "normal" with
65      * ny = exponent of y
66      * nd = exponent of x minus exponent of y
67      */
68     /* INDENT ON */
69     ny = iy >> 23;
70     k = ix >> 23;

72     /* special case for subnormal y or x */
73     if (ny == 0) {
74         ny = 1;
75         while (iy < iu) {
76             ny -= 1;
77             iy += iy;
78         }
79         nd = k - ny;
80         if (k == 0) {
81             nd += 1;
82             while (ix < iu) {
83                 nd -= 1;
84                 ix += ix;
85             }
86         } else {
87             ix = iu | (ix & im);
88         }
89     } else {
90         nd = k - ny;
91         ix = iu | (ix & im);
92         iy = iu | (iy & im);
93     }

95     /* fix point fmod for normalized ix and iy */
96     /* INDENT OFF */
97     /*
98      * while (nd--) {
99      *     iz = ix - iy;
100     *     if (iz < 0)
101     *         ix = ix + ix;
102     *     else if (iz == 0) {
103     *         *(int *) &w = is & hx;
104     *         return w;
105     *     }
106     *     else
107     *         ix = iz + iz;
108     * }
109     */
110     /* INDENT ON */
111     /* unroll the above loop 4 times to gain performance */
112     k = nd >> 2;
113     nd -= k << 2;
114     while (k--) {
115         iz = ix - iy;
116         if (iz >= 0)
117             ix = iz + iz;
118         else
119             ix += ix;
120         iz = ix - iy;
121         if (iz >= 0)
122             ix = iz + iz;
123         else

```

```
124         ix += ix;
125         iz = ix - iy;
126         if (iz >= 0)
127             ix = iz + iz;
128         else
129             ix += ix;
130         iz = ix - iy;
131         if (iz >= 0)
132             ix = iz + iz;
133         else
134             ix += ix;
135         if (iz == 0) {
136             *(int *)&w = is & hx;
137             return (w);
138         }
139     }
140     while (nd--) {
141         iz = ix - iy;
142         if (iz >= 0)
143             ix = iz + iz;
144         else
145             ix += ix;
146     }
147     /* end of unrolling */
148
149     iz = ix - iy;
150     if (iz >= 0)
151         ix = iz;
152
153     /* convert back to floating value and restore the sign */
154     if (ix == 0) {
155         *(int *)&w = is & hx;
156         return (w);
157     }
158     while (ix < iu) {
159         ix += ix;
160         ny -= 1;
161     }
162     while (ix > (iu + iu)) {
163         ny += 1;
164         ix >>= 1;
165     }
166     if (ny > 0) {
167         *(int *)&w = (is & hx) | (ix & im) | (ny << 23);
168     } else {
169         /* subnormal output */
170         k = -ny + 1;
171         ix >>= k;
172         *(int *)&w = (is & hx) | ix;
173     }
174 }
175 return (w);
176 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/gammaf.c

1

1095 Tue Nov 25 12:57:50 2014

new/usr/src/lib/libm/common/R/gammaf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __gammaf = gammaf
```

```
29 #pragma weak gammaf = __gammaf
```

```
31 #include "libm.h"
```

```
33 float
```

```
34 gammaf(float x) {
```

```
35     return (lgammaf(x));
```

```
36 }
```

```
_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/gammaf_r.c

1

1130 Tue Nov 25 12:57:50 2014

new/usr/src/lib/libm/common/R/gammaf_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __gammaf_r = gammaf_r
```

```
29 #pragma weak gammaf_r = __gammaf_r
```

```
31 #include "libm.h"
```

```
33 float
34 gammaf_r(float x, int *signgamfp) {
35     return (lgammaf_r(x, signgamfp));
36 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/R/hypotf.c

1

```
*****
1821 Tue Nov 25 12:57:51 2014
new/usr/src/lib/libm/common/R/hypotf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __hypotf = hypotf
30 #pragma weak hypotf = __hypotf
31
32 #include "libm.h"
33
34 float
35 hypotf(float x, float y) {
36     double dx, dy;
37     float w;
38     int ix, iy;
39
40     ix = (*(int *) &x) & 0x7fffffff;
41     iy = (*(int *) &y) & 0x7fffffff;
42     if (ix >= 0x7f800000) {
43         if (ix == 0x7f800000)
44             *(int *) &w = x == y ? iy : ix; /* w = |x| = inf */
45         else if (iy == 0x7f800000)
46             *(int *) &w = x == y ? ix : iy; /* w = |y| = inf */
47         else
48             w = fabsf(x) * fabsf(y); /* + -> * for Cheetah */
49     } else if (iy >= 0x7f800000) {
50         if (iy == 0x7f800000)
51             *(int *) &w = x == y ? ix : iy; /* w = |y| = inf */
52         else
53             w = fabsf(x) * fabsf(y); /* + -> * for Cheetah */
54     } else if (ix == 0)
55         *(int *) &w = iy; /* w = |y| */
56     else if (iy == 0)
57         *(int *) &w = ix; /* w = |x| */
```

new/usr/src/lib/libm/common/R/hypotf.c

2

```
58     else {
59         dx = (double) x;
60         dy = (double) y;
61         w = (float) sqrt(dx * dx + dy * dy);
62     }
63     return (w);
64 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/ilogbf.c

1

```
*****
2102 Tue Nov 25 12:57:51 2014
new/usr/src/lib/libm/common/R/ilogbf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __ilogbf = ilogbf
30 #if defined(ELFOBJ)
31 #pragma weak ilogbf = __ilogbf
32 #endif
33
34 #include "libm.h"
34 #include "xpg6.h" /* __xpg6 */
35
35 #if defined(__x86)
37 #if defined(USE_FPSCALE) || defined(__x86)
36 static const float two25 = 33554432.0F;
37 #else
38 /*
39  * v: a non-zero subnormal |x|
40  */
41 static int
42 ilogbf_subnormal(unsigned v) {
43     int r = -126 - 23;
44
45     if (v & 0xffff0000)
46         r += 16, v >>= 16;
47     if (v & 0xff00)
48         r += 8, v >>= 8;
```

new/usr/src/lib/libm/common/R/ilogbf.c

2

```
49     if (v & 0xf0)
50         r += 4, v >>= 4;
51     v <<= 1;
52     return (r + ((0xffffaa50 >> v) & 0x3));
53 }
54 #endif /* defined(__x86) */
56 #endif /* defined(USE_FPSCALE) */
57
58 static int
57 raise_invalid(int v) { /* SUSv3 requires ilogbf(0,+/-Inf,NaN) raise invalid */
58 #ifndef lint
59     if ((__xpg6 & _C99SUSv3_ilogb_0InfNaN_raises_invalid) != 0) {
60         static const double zero = 0.0;
61         volatile double dummy;
62
63         dummy = zero / zero;
64     }
65 #endif
66     return (v);
67 }
68
69 int
70 ilogbf(float x) {
71     int k = *((int *) &x) & ~0x80000000;
72
73     if (k < 0x00800000) {
74         if (k == 0)
75             return (raise_invalid(0x80000001));
76         else {
77 #if defined(__x86)
79 #if defined(USE_FPSCALE) || defined(__x86)
78             x *= two25;
79             return (((int *) &x) & 0x7f800000) >> 23) - 152);
80 #else
81             return (ilogbf_subnormal(k));
82 #endif
83         }
84     } else if (k < 0x7f800000)
85         return ((k >> 23) - 127);
86     else
87         return (raise_invalid(0x7fffffff));
88 }
89
90 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/isnanf.c

1

1154 Tue Nov 25 12:57:52 2014

new/usr/src/lib/libm/common/R/isnanf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __isnanf = isnanf
31 #pragma weak _isnanf = isnanf
32 #if defined(ELFOSBJ)
33 #pragma weak isnanf = __isnanf
34 #pragma weak _isnanf = __isnanf
35 #endif
```

```
33 #include "libm.h"
```

```
35 int
36 isnanf(float x) {
37     return ((* (int *) &x & ~0x80000000) > 0x7f800000);
38 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/R/lgammaf.c

1

1268 Tue Nov 25 12:57:52 2014

new/usr/src/lib/libm/common/R/lgammaf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __lgammaf = lgammaf
```

```
29 #pragma weak lgammaf = __lgammaf
```

```
31 #include "libm.h"
```

```
33 extern int signgamf;
```

```
35 float
```

```
36 lgammaf(float x) {
37     float y;
```

```
39     if (isnanf(x))
```

```
40         return (x * x);
```

```
41     y = (float)__k_lgamma((double)x, &signgamf);
```

```
42     signgam = signgamf; /* SUSv3 requires the setting of signgam */
```

```
43     return (y);
```

```
44 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/lgammaf_r.c

1

1183 Tue Nov 25 12:57:52 2014

new/usr/src/lib/libm/common/R/lgammaf_r.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __lgammaf_r = lgammaf_r
```

```
29 #pragma weak lgammaf_r = __lgammaf_r
```

```
31 #include "libm.h"
```

```
33 float
34 lgammaf_r(float x, int *signgamfp) {
35     if (isnanf(x))
36         return (x * x);
37     return ((float)__k_lgamma((double)x, signgamfp));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/log10f.c

1

1409 Tue Nov 25 12:57:53 2014

new/usr/src/lib/libm/common/R/log10f.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __log10f = log10f
```

```
29 #pragma weak log10f = __log10f
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f, mone = -1.0f;
```

```
35 float
36 log10f(float x) {
37     int    hx, ix;
38     float  w;
39
40     hx = *(int *)&x;
41     ix = hx & ~0x80000000;
42     if (ix > 0x7f800000)
43         return (x * x);
44     if (ix == 0x7f800000)
45         return (x + x * x);
46     if (ix == 0) {
47         w = mone;
48         return (w / zero);
49     }
50     if (hx < 0) {
51         w = zero;
52         return (w / zero);
53     }
54     return ((float)log10((double)x));
55 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/loglpf.c

1

```
*****
1416 Tue Nov 25 12:57:53 2014
new/usr/src/lib/libm/common/R/loglpf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __loglpf = loglpf
29 #pragma weak loglpf = __loglpf
```

```
31 #include "libm.h"
```

```
33 static const float zero = 0.0f;
```

```
35 float
36 loglpf(float x) {
37     int    ix;

39     ix = *(int *)&x;
40     if (ix >= 0x7f800000) {
41         /* x is +inf or nan */
42         return (x * x);
43     }
44     if (ix < 0) {
45         ix &= ~0x80000000;
46         if (ix == 0x3f800000) /* x is -1 */
47             return (x / zero);
48         if (ix > 0x3f800000) /* x is < -1 or nan */
49             return ((x * zero) / zero);
50     }
51     return ((float)loglp((double)x));
52 }
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/R/log2f.c

1

1197 Tue Nov 25 12:57:54 2014

new/usr/src/lib/libm/common/R/log2f.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __log2f = log2f
30 #pragma weak log2f = __log2f
```

```
32 #include "libm.h"
```

```
34 float
35 log2f(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) log2((double) x));
42 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/logbf.c

1

```
*****
2080 Tue Nov 25 12:57:54 2014
new/usr/src/lib/libm/common/R/logbf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __logbf = logbf
30 #if defined(ELFOSBJ)
31 #pragma weak logbf = __logbf
32 #endif
33
34 #include "libm.h"
34 #include "xpg6.h" /* __xpg6 */
34 #define _C99SUSv3_logb _C99SUSv3_logb_subnormal_is_like_ilogb
35
36 #if defined(__x86)
36 #if defined(USE_FPSCALE) || defined(__x86)
37 static const float two25 = 33554432.0F;
38 #else
39 /*
40  * v: a non-zero subnormal |x|
41  */
42 static int
43 ilogbf_subnormal(unsigned v) {
44     int r = -126 - 23;
45
46     if (v & 0xffff0000)
47         r += 16, v >>= 16;
48     if (v & 0xff00)
```

new/usr/src/lib/libm/common/R/logbf.c

2

```
49         r += 8, v >>= 8;
50     if (v & 0xf0)
51         r += 4, v >>= 4;
52     v <<= 1;
53     return (r + ((0xffffaa50 >> v) & 0x3));
54 }
55 #endif /* defined(__x86) */
57 #endif /* defined(USE_FPSCALE) */
58
59 static float
60 raise_division(float t) {
61     #pragma STDC FENV_ACCESS ON
62     static const float zero = 0.0F;
63     return (t / zero);
64 }
65
66 float
67 logbf(float x) {
68     int k = *((int *) &x) & ~0x80000000;
69
70     if (k < 0x00800000) {
71         if (k == 0)
72             return (raise_division(-1.0F));
73         else if ((__xpg6 & _C99SUSv3_logb) != 0) {
74             #if defined(__x86)
74             #if defined(USE_FPSCALE) || defined(__x86)
75                 x *= two25;
76                 return ((float) (((*(int *) &x) & 0x7f800000) >> 23) -
77                     152));
78             #else
79                 return ((float) ilogbf_subnormal(k));
80             #endif
81         } else
82             return (-126.F);
83     } else if (k < 0x7f800000)
84         return ((float) ((k >> 23) - 127));
85     else
86         return (x * x);
87 }
88
89 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/R/logf.c

1

```
*****
4679 Tue Nov 25 12:57:55 2014
new/usr/src/lib/libm/common/R/logf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28 #pragma weak __logf = logf
29 #pragma weak logf = __logf
30
31 /*
32  * Algorithm:
33  *
34  * Let y = x rounded to six significant bits. Then for any choice
35  * of e and z such that y = 2^e z, we have
36  *
37  * log(x) = e log(2) + log(z) + log(1+(x-y)/y)
38  *
39  * Note that (x-y)/y = (x'-y')/y' for any scaled x' = sx, y' = sy;
40  * in particular, we can take s to be the power of two that makes
41  * ulp(x') = 1.
42  *
43  * From a table, obtain l = log(z) and r = 1/y'. For |s| <= 2^-6,
44  * approximate log(1+s) by a polynomial p(s) where p(s) := s+s*s*
45  * (K1+s*(K2+s*K3)). Then we compute the expression above as
46  * e*ln2 + l + p(r*(x'-y')) all evaluated in double precision.
47  *
48  * When x is subnormal, we first scale it to the normal range,
49  * adjusting e accordingly.
50  *
51  * Accuracy:
52  *
53  * The largest error is less than 0.6 ulps.
54  */
55 #include "libm.h"
```

new/usr/src/lib/libm/common/R/logf.c

2

```
58 /*
59  * For i = 0, ..., 12,
60  * TBL[2i] = log(1 + i/32) and TBL[2i+1] = 2^-23 / (1 + i/32)
61  *
62  * For i = 13, ..., 32,
63  * TBL[2i] = log(1/2 + i/64) and TBL[2i+1] = 2^-23 / (1 + i/32)
64  */
65 static const double TBL[] = {
66 0.000000000000000000e+00, 1.192092895507812500e-07,
67 3.077165866675368733e-02, 1.155968868371212153e-07,
68 6.062462181643483994e-02, 1.121969784007352926e-07,
69 8.961215868968713805e-02, 1.089913504464285680e-07,
70 1.177830356563834557e-01, 1.059638129340277719e-07,
71 1.451820098444978890e-01, 1.030999260979729787e-07,
72 1.718502569266592284e-01, 1.003867701480263102e-07,
73 1.978257433299198675e-01, 9.781275040064102225e-08,
74 2.231435513142097649e-01, 9.536743164062500529e-08,
75 2.478361639045812692e-01, 9.304139672256097884e-08,
76 2.719337154836417580e-01, 9.082612537202380448e-08,
77 2.954642128938358980e-01, 8.871388989825581272e-08,
78 3.184537311185345887e-01, 8.669766512784091150e-08,
79 -3.522205935893520934e-01, 8.477105034722222546e-08,
80 -3.302416868705768671e-01, 8.292820142663043248e-08,
81 -3.087354816496132859e-01, 8.116377160904255122e-08,
82 -2.876820724517809014e-01, 7.947285970052082892e-08,
83 -2.670627852490452536e-01, 7.785096460459183052e-08,
84 -2.468600779315257843e-01, 7.629394531250000159e-08,
85 -2.270574506353460753e-01, 7.479798560049019504e-08,
86 -2.076393647782444896e-01, 7.335956280048077330e-08,
87 -1.885911698075500298e-01, 7.197542010613207272e-08,
88 -1.698990367953974734e-01, 7.064254195601851460e-08,
89 -1.515498981272009327e-01, 6.935813210227272390e-08,
90 -1.335313926245226268e-01, 6.811959402901785336e-08,
91 -1.15831815251217008e-01, 6.692451343201754014e-08,
92 -9.844007281325252434e-02, 6.577064251077586116e-08,
93 -8.134563945395240081e-02, 6.465588585805084723e-08,
94 -6.453852113757117814e-02, 6.357828776041666578e-08,
95 -4.800921918636060631e-02, 6.253602074795082293e-08,
96 -3.174869831458029812e-02, 6.152737525201612732e-08,
97 -1.574835696813916761e-02, 6.055075024801586965e-08,
98 0.000000000000000000e+00, 5.960464477539062500e-08,
99 };
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/nextafterf.c

1

```
*****
1937 Tue Nov 25 12:57:55 2014
new/usr/src/lib/libm/common/R/nextafterf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __nextafterf = nextafterf
30 #if defined(ELFOBJ)
31 #pragma weak nextafterf = __nextafterf
32 #endif

32 #include "libm.h"

34 float
35 nextafterf(float x, float y) {
36     float w;
37     int *pw = (int *) &w;
38     int *px = (int *) &x;
39     int *py = (int *) &y;
40     int ix, iy, iz;

42     ix = px[0];
43     iy = py[0];
44     if ((ix & ~0x80000000) > 0x7f800000)
45         return (x * y); /* + -> * for Cheetah */
46     if ((iy & ~0x80000000) > 0x7f800000)
47         return (y * x); /* + -> * for Cheetah */
48     if (ix == iy || (ix | iy) == 0x80000000)
49         return (y); /* C99 requirement */
```

new/usr/src/lib/libm/common/R/nextafterf.c

2

```
50     if ((ix & ~0x80000000) == 0)
51         iz = 1 | (iy & 0x80000000);
52     else if (ix > 0) {
53         if (ix > iy)
54             iz = ix - 1;
55         else
56             iz = ix + 1;
57     } else {
58         if (iy < 0 && ix < iy)
59             iz = ix + 1;
60         else
61             iz = ix - 1;
62     }
63     pw[0] = iz;
64     ix = iz & 0x7f800000;
65     if (ix == 0x7f800000) {
66         /* raise overflow */
67         volatile float t;

69         *(int *) &t = 0x7f7fffff;
70         t *= t;
71     } else if (ix == 0) {
72         /* raise underflow */
73         volatile float t;

75         *(int *) &t = 0x00800000;
76         t *= t;
77     }
78     return (w);
79 }
unchanged_portion_omitted
```


new/usr/src/lib/libm/common/R/powf.c

1

```
*****
8104 Tue Nov 25 12:57:55 2014
new/usr/src/lib/libm/common/R/powf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23  */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27  */

29 #pragma weak __powf = powf
29 #pragma weak powf = __powf

31 #include "libm.h"
32 #include "xpg6.h" /* __xpg6 */
33 #define _C99SUSv3_pow _C99SUSv3_pow_treats_Inf_as_an_even_int

35 #if defined(__i386) && !defined(__amd64)
36 extern int __swapRP(int);
37 #endif

39 /* INDENT OFF */
40 static const double
41     ln2 = 6.93147180559945286227e-01, /* 0x3fe62e42, 0xfefa39ef */
42     invln2 = 1.44269504088896338700e+00, /* 0x3ff71547, 0x652b82fe */
43     dtwo = 2.0,
44     done = 1.0,
45     dhalf = 0.5,
46     d32 = 32.0,
47     dl_32 = 0.03125,
48     A0 = 1.9999999999813723303647511146995966439250e+0000,
49     A1 = 6.666910817935858533770138657139665608610e-0001,
50     t0 = 2.000000000004777489262405315073203746943e+0000,
51     t1 = 1.666663408349926379873111932994250726307e-0001;

53 static const double S[] = {
54     1.000000000000000000e+00, /* 3FF0000000000000 */
55     1.02189714865411662714e+00, /* 3FF059B0D3158574 */
56     1.04427378242741375480e+00, /* 3FF0B5586CF9890F */
57     1.06714040067682369717e+00, /* 3FF11301D0125B51 */
```

new/usr/src/lib/libm/common/R/powf.c

2

```
58     1.09050773266525768967e+00, /* 3FF172B83C7D517B */
59     1.11438674259589243221e+00, /* 3FF1D4873168B9AA */
60     1.13878863475669156458e+00, /* 3FF2387A6E756238 */
61     1.1637248587757747552e+00, /* 3FF29E9DF51FDEE1 */
62     1.18920711500272102690e+00, /* 3FF306FE0A31B715 */
63     1.21524735998046895524e+00, /* 3FF371A7373AA9CB */
64     1.24185781207348400201e+00, /* 3FF3DEA64C123422 */
65     1.26905095719173321989e+00, /* 3FF44E086061892D */
66     1.29683955465100964055e+00, /* 3FF4BFDAD5362A27 */
67     1.32523664315974132322e+00, /* 3FF5342B569D4F82 */
68     1.35425554693689265129e+00, /* 3FF5AB07DD485429 */
69     1.38390988196383202258e+00, /* 3FF6247EB03A5585 */
70     1.41421356237309514547e+00, /* 3FF6A09E667F3BCD */
71     1.44518080697704665027e+00, /* 3FF71F75E8EC5F74 */
72     1.47682614593949934623e+00, /* 3FF7A11473EB0187 */
73     1.50916442759342284141e+00, /* 3FF82589994CCE13 */
74     1.54221082540794074411e+00, /* 3FF8ACE5422A0DB */
75     1.57598084510788649659e+00, /* 3FF93737B0CDC5E5 */
76     1.61049033194925428347e+00, /* 3FF9C49182A3F090 */
77     1.64575547815396494578e+00, /* 3FFA5503B23E255D */
78     1.68179283050742900407e+00, /* 3FFAE89F995AD3AD */
79     1.71861929812247793414e+00, /* 3FFB7F76F2FB5E47 */
80     1.75625216037329945351e+00, /* 3FFC199BDD85529C */
81     1.79470907500310716820e+00, /* 3FFCB720DCEF9069 */
82     1.83400808640934243066e+00, /* 3FFD5818DCFBA487 */
83     1.87416763411029996256e+00, /* 3FFDFC97337B9B5F */
84     1.91520656139714740007e+00, /* 3FFEA4AFA2A490DA */
85     1.95714412417540017941e+00, /* 3FFF50765B6E4540 */
86 };
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/remainderf.c

1

1381 Tue Nov 25 12:57:56 2014

new/usr/src/lib/libm/common/R/remainderf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __remainderf = remainderf
31 #if defined(ELFOSBJ)
32 #pragma weak remainderf = __remainderf
33 #endif
34
35 #include "libm.h"
36
37 float
38 remainderf(float x, float y) {
39     if (isnanf(x) || isnanf(y))
40         return (x * y);
41     if (y == 0.0f || (*(int *) &x & ~0x80000000) == 0x7f800000) {
42         /* y is 0 or x is infinite; raise invalid and return NaN */
43         y = 0.0f;
44         *(int *) &x = 0x7f800000;
45         return (x * y);
46     }
47     return ((float) remainder((double) x, (double) y));
48 }
49
50 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/R/rintf.c

1

3660 Tue Nov 25 12:57:56 2014

new/usr/src/lib/libm/common/R/rintf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __rintf = rintf
30 #if defined(ELFOBJ)
31 #pragma weak aintf = __aintf
32 #pragma weak anintf = __anintf
33 #pragma weak irintf = __irintf
34 #pragma weak nintf = __nintf
35 #pragma weak rintf = __rintf
36 #endif
37
38 /* INDEXT OFF */
39 /*
40  * aintf(x)    return x chopped to integral value
41  * anintf(x)   return sign(x)*(|x|+0.5) chopped to integral value
42  * irintf(x)   return rint(x) in integer format
43  * nintf(x)    return anint(x) in integer format
44  * rintf(x)    return x rounded to integral according to the rounding direction
45  *
46  * NOTE: rintf(x), aintf(x) and anintf(x) return results with the same sign as
47  * x's, including 0.0.
48 */
49
50 #include "libm.h"
```

new/usr/src/lib/libm/common/R/rintf.c

2

```
46 static const float xf[] = {
47 /* ZEROF */      0.0f,
48 /* TWO_23F */    8.3886080000e6f,
49 /* MTWO_23F */  -8.3886080000e6f,
50 /* ONEF */       1.0f,
51 /* MONEF */      -1.0f,
52 /* HALFF */      0.5f,
53 /* MHALFF */     -0.5f,
54 /* HUGEFF */    1.0e30f,
55 };
56
57 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/scalbf.c

1

```
*****
1725 Tue Nov 25 12:57:57 2014
new/usr/src/lib/libm/common/R/scalbf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __scalbf = scalbf
29 #pragma weak scalbf = __scalbf
```

```
31 #include "libm.h"
```

```
33 float
34 scalbf(float x, float y) {
35     int    ix, iy, hx, hy, n;

37     ix = *(int *)&x;
38     iy = *(int *)&y;
39     hx = ix & ~0x80000000;
40     hy = iy & ~0x80000000;

42     if (hx > 0x7f800000 || hy >= 0x7f800000) {
43         /* x is nan or y is inf or nan */
44         return ((iy < 0)? x / -y : x * y);
45     }

47     /* see if y is an integer without raising inexact */
48     if (hy >= 0x4b000000) {
49         /* |y| >= 2^23, so it must be an integer */
50         n = (iy < 0)? -65000 : 65000;
51     } else if (hy < 0x3f800000) {
52         /* |y| < 1, so it must be zero or non-integer */
53         return ((hy == 0)? x : (x - x) / (x - x));
54     } else {
55         if (hy & ((1 << (0x96 - (hy >> 23))) - 1))
56             return ((y - y) / (y - y));
57         n = (int)y;
```

new/usr/src/lib/libm/common/R/scalbf.c

2

```
58     }
59     return (scalbnf(x, n));
60 }
_____unchanged_portion_omitted_
```

new/usr/src/lib/libm/common/R/scalbnf.c

1

```
*****
2388 Tue Nov 25 12:57:57 2014
new/usr/src/lib/libm/common/R/scalbnf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __scalbnf = scalbnf
30 #if defined(ELFOBJ)
31 #pragma weak scalbnf = __scalbnf
32 #endif
33
34 #include "libm.h"
34 #include <float.h>          /* FLT_MAX, FLT_MIN */
34 #include <stdlib.h>        /* abs */
35
36 static const float twom25f = 2.98023223876953125e-8F;
37 #if defined(__x86)
38 #if defined(USE_FPSCALE) || defined(__x86)
38 static const float two23f = 8388608.0F;
39 #else
40 /*
41  * v: a non-zero subnormal |x|; returns [-22, 0]
42  */
43 static int
44 ilogbf_biased(unsigned v) {
45     int r = -22;
46
47     if (v & 0xffff0000)
48         r += 16, v >>= 16;
```

new/usr/src/lib/libm/common/R/scalbnf.c

2

```
49     if (v & 0xff00)
50         r += 8, v >>= 8;
51     if (v & 0xf0)
52         r += 4, v >>= 4;
53     v <<= 1;
54     return (r + ((0xffffaa50 >> v) & 0x3));
55 }
56 #endif /* defined(__x86) */
58 #endif /* defined(USE_FPSCALE) */
59
58 float
59 scalbnf(float x, int n) {
60     int *px = (int *) &x, ix, k;
61
62     ix = *px & ~0x80000000;
63     k = ix >> 23;
64     if (k == 0xff)
65 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
66         return (ix > 0x7f800000 ? x * x : x);
67 #else
68         return (x + x);
69 #endif
70     if (ix == 0 || n == 0)
71         return (x);
72     if (k == 0) {
73 #if defined(__x86)
74 #if defined(USE_FPSCALE) || defined(__x86)
75         x *= two23f;
76         k = ((*px & ~0x80000000) >> 23) - 23;
77 #else
78         k = ilogbf_biased(ix);
79         *px = (*px & 0x80000000) | (ix << (-k + 1));
80 #endif
81 #endif
82     }
83     if ((unsigned) abs(n) >= 131072) /* cast to unsigned for -2^31 */
84         n >>= 1; /* avoid subsequent integer overflow */
85     k += n;
86     if (k > 0xfe)
87         return (FLT_MAX * copysignf(FLT_MAX, x));
88     if (k <= -25)
89         return (FLT_MIN * copysignf(FLT_MIN, x));
90     if (k > 0) {
91         *px = (*px & ~0x7f800000) | (k << 23);
92         return (x);
93     }
94     k += 25;
95     *px = (*px & ~0x7f800000) | (k << 23);
96     return (x * twom25f);
97 }
98
99 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/R/signgamf.c

1

1070 Tue Nov 25 12:57:58 2014

new/usr/src/lib/libm/common/R/signgamf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __signgamf = signgamf
30 #pragma weak signgamf = __signgamf
```

```
32 #include "libm.h"
```

```
34 int signgamf = 0;
```

new/usr/src/lib/libm/common/R/significandf.c

1

1337 Tue Nov 25 12:57:58 2014

new/usr/src/lib/libm/common/R/significandf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __significandf = significandf
30 #if defined(ELFOBJ)
31 #pragma weak significandf = __significandf
32 #endif
```

```
32 #include "libm.h"
```

```
34 float
35 significandf(float x) {
36     int ix = *(int *) &x & ~0x80000000;
37
38     if (ix == 0 || ix >= 0x7f800000) /* 0/+Inf/NaN */
39 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
40         return (ix > 0x7f800000 ? x * x : x);
41 #else
42         return (x + x);
43 #endif
44     else
45         return (scalbnf(x, -ilogbf(x)));
46 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/sincosf.c

1

```
*****
5085 Tue Nov 25 12:57:58 2014
new/usr/src/lib/libm/common/R/sincosf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28 #pragma weak __sincosf = sincosf
29 #pragma weak sincosf = __sincosf
30
31 /* INDENT OFF */
32 /*
33 * For  $|x| < \pi/4$ , let  $z = x * x$ , and approximate  $\sin(x)$  by
34 *
35 *  $S(x) = x(S0 + S1*z)(S2 + S3*z + z*z)$ 
36 * where
37 *  $S0 = 1.85735322054308378716204874632872525989806770558e-0003,$ 
38 *  $S1 = -1.95035094218403635082921458859320791358115801259e-0004,$ 
39 *  $S2 = 5.38400550766074785970952495168558701485841707252e+0002,$ 
40 *  $S3 = -3.31975110777873728964197739157371509422022905947e+0001,$ 
41 *
42 * with error bounded by  $|(\sin(x) - S(x))/x| < 2^{*-28.2}$ , and
43 *  $\cos(x)$  by
44 *
45 *  $C(x) = (C0 + C1*z + C2*z*z) * (C3 + C4*z + z*z)$ 
46 * where
47 *  $C0 = 1.09349482127188401868272000389539985058873853699e-0003$ 
48 *  $C1 = -5.03324285989964979398034700054920226866107675091e-0004$ 
49 *  $C2 = 2.43792880266971107750418061559602239831538067410e-0005$ 
50 *  $C3 = 9.14499072605666582228127405245558035523741471271e+0002$ 
51 *  $C4 = -3.63151270591815439197122504991683846785293207730e+0001$ 
52 *
53 * with error bounded by  $|\cos(x) - C(x)| < 2^{*-34.2}$ .
54 */
55 /* INDENT ON */
57 #include "libm.h"
```

new/usr/src/lib/libm/common/R/sincosf.c

2

```
59 extern const int _TBL_ipio2_inf[];
60 extern int __rem_pio2m(double *, double *, int, int, int, const int *);
61 #if defined(__i386) && !defined(__amd64)
62 extern int __swapRP(int);
63 #endif
64
65 static const double C[] = {
66     1.85735322054308378716204874632872525989806770558e-0003,
67     -1.95035094218403635082921458859320791358115801259e-0004,
68     5.38400550766074785970952495168558701485841707252e+0002,
69     -3.31975110777873728964197739157371509422022905947e+0001,
70     1.09349482127188401868272000389539985058873853699e-0003,
71     -5.03324285989964979398034700054920226866107675091e-0004,
72     2.43792880266971107750418061559602239831538067410e-0005,
73     9.14499072605666582228127405245558035523741471271e+0002,
74     -3.63151270591815439197122504991683846785293207730e+0001,
75     0.636619772367581343075535, /* 2^-1 * 1.45F306DC9C883 */
76     0.5,
77     1.570796326734125614166, /* 2^0 * 1.921FB54400000 */
78     6.077100506506192601475e-11, /* 2^-34 * 1.0B4611A626331 */
79 };
_____ unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/R/sincospif.c

1

```
*****
1347 Tue Nov 25 12:57:59 2014
new/usr/src/lib/libm/common/R/sincospif.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak sincospif = __sincospif
```

```
30 #include "libm.h"
```

```
32 extern void sincospi(double, double *, double *);
```

```
34 void
35 sincospif(float x, float *s, float *c) {
36     double ds, dc;

38 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
39     if (isnanf(x))
40         *s = *c = x * x;
41     else {
42 #endif
43         sincospi((double) x, &ds, &dc);
44         *s = (float) ds;
45         *c = (float) dc;
46 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
47     }
48 #endif
49 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/sinf.c

1

```
*****
3911 Tue Nov 25 12:58:00 2014
new/usr/src/lib/libm/common/R/sinf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak __sinf = sinf
29 #pragma weak sinf = __sinf

31 /*
32 * See sincosf.c
33 */

35 #include "libm.h"

37 extern const int _TBL_ipio2_inf[];
38 extern int __rem_pio2m(double *, double *, int, int, int, const int *);
39 #if defined(__i386) && !defined(__amd64)
40 extern int __swapRP(int);
41 #endif

43 static const double C[] = {
44     1.85735322054308378716204874632872525989806770558e-0003,
45     -1.95035094218403635082921458859320791358115801259e-0004,
46     5.38400550766074785970952495168558701485841707252e+0002,
47     -3.31975110777873728964197739157371509422022905947e+0001,
48     1.09349482127188401868272000389539985058873853699e-0003,
49     -5.0332428598964979398034700054920226866107675091e-0004,
50     2.43792880266971107750418061559602239831538067410e-0005,
51     9.14499072605666582228127405245558035523741471271e+0002,
52     -3.63151270591815439197122504991683846785293207730e+0001,
53     0.636619772367581343075535, /* 2^-1 * 1.45F306DC9C883 */
54     0.5,
55     1.570796326734125614166, /* 2^0 * 1.921FB54400000 */
56     6.077100506506192601475e-11, /* 2^-34 * 1.0B4611A626331 */
57 };
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/R/sinhf.c

1

```
*****
1390 Tue Nov 25 12:58:00 2014
new/usr/src/lib/libm/common/R/sinhf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __sinhf = sinhf
29 #pragma weak sinhf = __sinhf
```

```
31 #include "libm.h"
```

```
33 float
34 sinhf(float x) {
35     double s;
36     float w;
37     int hx, ix;

39     hx = *(int *)&x;
40     ix = hx & ~0x80000000;
41     if (ix >= 0x7f800000) {
42         /* sinhf(x) is x if x is +-Inf or NaN */
43         return (x * 1.0f);
44     }
45     if (ix >= 0x43000000) /* sinhf(x) trivially overflows */
46         s = (hx < 0)? -1.0e100 : 1.0e100;
47     else
48         s = sinh((double)x);
49     w = (float)s;
50     return (w);
51 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/R/sqrtf.c

1

```
*****
1138 Tue Nov 25 12:58:01 2014
new/usr/src/lib/libm/common/R/sqrtf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */
29 #pragma weak __sqrtf = sqrtf
29 #pragma weak sqrtf = __sqrtf
31 #include "libm.h"
33 #ifdef __INLINE
34 extern float __inline_sqrtf(float);
36 float
37 sqrtf(float x) {
38     return (__inline_sqrtf(x));
39 }
42 #else /* defined(__INLINE) */
44 static const float huge = 1.0e35F, tiny = 1.0e-35F, zero = 0.0f;
46 float
47 sqrtf(float x) {
48     float    dz, w;
49     int      *pw = (int *)&w;
50     int      ix, j, r, q, m, n, s, t;
```

new/usr/src/lib/libm/common/R/sqrtf.c

2

```
52     w = x;
53     ix = pw[0];
54     if (ix <= 0) {
55         /* x is <= 0 or nan */
56         j = ix & 0x7fffffff;
57         if (j == 0)
58             return (w);
59         return ((w * zero) / zero);
60     }
62     if ((ix & 0x7f800000) == 0x7f800000) {
63         /* x is +inf or nan */
64         return (w * w);
65     }
67     m = ir_ilogb_(&w);
68     n = -m;
69     w = r_scalbn_(&w, (int *)&n);
70     ix = (pw[0] & 0x007fffff) | 0x00800000;
71     n = m / 2;
72     if ((n + n) != m) {
73         ix = ix + ix;
74         m -= 1;
75         n = m / 2;
76     }
78     /* generate sqrt(x) bit by bit */
79     ix <<= 1;
80     q = s = 0;
81     r = 0x01000000;
82     for (j = 1; j <= 25; j++) {
83         t = s + r;
84         if (t <= ix) {
85             s = t + r;
86             ix -= t;
87             q += r;
88         }
89         ix <<= 1;
90         r >>= 1;
91     }
92     if (ix == 0)
93         goto done;
95     /* raise inexact and determine the ambient rounding mode */
96     dz = huge - tiny;
97     if (dz < huge)
98         goto done;
99     dz = huge + tiny;
100    if (dz > huge)
101        q += 1;
102    q += (q & 1);
104 done:
105    pw[0] = (q >> 1) + 0x3f000000;
106    return (r_scalbn_(&w, (int *)&n));
107 }
109 #endif /* defined(__INLINE) */
```

new/usr/src/lib/libm/common/R/tanf.c

1

4309 Tue Nov 25 12:58:01 2014

new/usr/src/lib/libm/common/R/tanf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __tanf = tanf
```

```
29 #pragma weak tanf = __tanf
```

```
31 #include "libm.h"
```

```
33 extern const int _TBL_ipio2_inf[];
34 extern int __rem_pio2m(double *, double *, int, int, int, const int *);
35 #if defined(__i386) && !defined(__amd64)
36 extern int __swapRP(int);
37 #endif
```

```
39 static const double C[] = {
40     1.0,
41     4.46066928428959230679140546271810308098793029785e-0003,
42     4.92165316309189027066395283327437937259674072266e+0000,
43     -7.11410648161473480044492134766187518835067749023e-0001,
44     4.08549808374053391446523164631798863410949707031e+0000,
45     2.50411070398050927821032018982805311679840087891e+0000,
46     1.11492064560251158411574579076841473579406738281e+0001,
47     -1.50565540968422650891511693771462887525558471680e+0000,
48     -1.81484378878349295050043110677506774663925170898e+0000,
49     3.33335997532835641297409611782510896641e-0001,
50     2.999997598248363761541668282006867229939e+00,
51     0.636619772367581343075535, /* 2^-1 * 1.45F306DC9C883 */
52     0.5,
53     1.570796326734125614166, /* 2^0 * 1.921FB54400000 */
54     6.077100506506192601475e-11, /* 2^-34 * 1.0B4611A626331 */
55 };
```

unchanged portion omitted

new/usr/src/lib/libm/common/R/tanhf.c

1

1197 Tue Nov 25 12:58:01 2014

new/usr/src/lib/libm/common/R/tanhf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __tanhf = tanhf
30 #pragma weak tanhf = __tanhf
```

```
32 #include "libm.h"
```

```
34 float
35 tanhf(float x) {
36 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
37     if (isnanf(x))
38         return (x * x);
39     else
40 #endif
41     return ((float) tanh((double) x));
42 }
```

unchanged_portion_omitted

```

*****
4644 Tue Nov 25 12:58:02 2014
new/usr/src/lib/libm/common/complex/cabs.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */
28
29 #pragma weak __cabs = cabs
29 #pragma weak cabs = __cabs
30
31 #include "libm_synonyms.h"
31 #include <math.h>
32 #include "complex_wrapper.h"
33
34 /*
35  * If C were the only standard we cared about, cabs could just call
36  * hypot. Unfortunately, various other standards say that hypot must
37  * call matherr and/or set errno to ERANGE when the result overflows.
38  * Since cabs should do neither of these things, we have to either
39  * make hypot a wrapper on another internal function or duplicate
40  * the hypot implementation here. I've chosen to do the latter.
41  */
42
43 static const double
44 zero = 0.0,
45 oneplu = 1.000000000000000022204e+00, /* 0x3ff00000 1 = 1+2**-52 */
46 twom53 = 1.11022302462515654042e-16, /* 0x3ca00000 0 = 2**-53 */
47 twom768 = 6.441148769597133308e-232, /* 2^-768 */
48 two768 = 1.552518092300708935e+231; /* 2^768 */
49
50 double
51 cabs(dcomplex z)
52 {
53     double x, y, xh, yh, w, ax, ay;
54     int i, j, nx, ny, ix, iy, iscale = 0;
55     unsigned lx, ly;

```

```

57     x = D_RE(z);
58     y = D_IM(z);
59
60     ix = ((int *)&x)[HIWORD] & ~0x80000000;
61     lx = ((int *)&x)[LOWORD];
62     iy = ((int *)&y)[HIWORD] & ~0x80000000;
63     ly = ((int *)&y)[LOWORD];
64
65     /* force ax = |x| ~>~ ay = |y| */
66     if (iy > ix) {
67         ax = fabs(y);
68         ay = fabs(x);
69         i = ix;
70         ix = iy;
71         iy = i;
72         i = lx;
73         lx = ly;
74         ly = i;
75     } else {
76         ax = fabs(x);
77         ay = fabs(y);
78     }
79     nx = ix >> 20;
80     ny = iy >> 20;
81     j = nx - ny;
82
83     if (nx >= 0x5f3) {
84         /* x >= 2^500 (x*x or y*y may overflow) */
85         if (nx == 0x7ff) {
86             /* inf or NaN, signal of sNaN */
87             if (((ix - 0x7ff00000) | lx) == 0)
88                 return ((ax == ay)? ay : ax);
89             else if (((iy - 0x7ff00000) | ly) == 0)
90                 return ((ay == ax)? ax : ay);
91             else
92                 return (ax * ay);
93         } else if (j > 32) {
94             /* x >> y */
95             if (j <= 53)
96                 ay *= twom53;
97             ax += ay;
98             return (ax);
99         }
100         ax *= twom768;
101         ay *= twom768;
102         iscale = 2;
103         ix -= 768 << 20;
104         iy -= 768 << 20;
105     } else if (ny < 0x23d) {
106         /* y < 2^-450 (x*x or y*y may underflow) */
107         if ((ix | lx) == 0)
108             return (ay);
109         if ((iy | ly) == 0)
110             return (ax);
111         if (j > 53) /* x >> y */
112             return (ax + ay);
113         iscale = 1;
114         ax *= two768;
115         ay *= two768;
116         if (nx == 0) {
117             if (ax == zero) /* guard subnormal flush to zero */
118                 return (ax);
119             ix = ((int *)&x)[HIWORD];
120         } else {
121             ix += 768 << 20;
122         }

```

```

123     if (ny == 0) {
124         if (ay == zero) /* guard subnormal flush to zero */
125             return (ax * twom768);
126         iy = ((int *)&ay)[HIWORD];
127     } else {
128         iy += 768 << 20;
129     }
130     j = (ix >> 20) - (iy >> 20);
131     if (j > 32) {
132         /* x >> y */
133         if (j <= 53)
134             ay *= twom53;
135         return ((ax + ay) * twom768);
136     }
137 } else if (j > 32) {
138     /* x >> y */
139     if (j <= 53)
140         ay *= twom53;
141     return (ax + ay);
142 }
143
144 /*
145  * Medium range ax and ay with max{|ax/ay|, |ay/ax|} bounded by 2^32.
146  * First check rounding mode by comparing oneplu*oneplu with oneplu
147  * + twom53. Make sure the computation is done at run-time.
148  */
149 if (((lx | ly) << 5) == 0) {
150     ay = ay * ay;
151     ax += ay / (ax + sqrt(ax * ax + ay));
152 } else if (oneplu * oneplu != oneplu + twom53) {
153     /* round-to-zero, positive, negative mode */
154     /* magic formula with less than an ulp error */
155     w = sqrt(ax * ax + ay * ay);
156     ax += ay / ((ax + w) / ay);
157 } else {
158     /* round-to-nearest mode */
159     w = ax - ay;
160     if (w > ay) {
161         ((int *)&xh)[HIWORD] = ix;
162         ((int *)&xh)[LOWORD] = 0;
163         ay = ay * ay + (ax - xh) * (ax + xh);
164         ax = sqrt(xh * xh + ay);
165     } else {
166         ax = ax + ax;
167         ((int *)&xh)[HIWORD] = ix + 0x00100000;
168         ((int *)&xh)[LOWORD] = 0;
169         ((int *)&yh)[HIWORD] = iy;
170         ((int *)&yh)[LOWORD] = 0;
171         ay = w * w + ((ax - xh) * yh + (ay - yh) * ax);
172         ax = sqrt(xh * yh + ay);
173     }
174 }
175 if (iscale > 0) {
176     if (iscale == 1)
177         ax *= twom768;
178     else
179         ax *= two768; /* must generate side effect here */
180 }
181 return (ax);
182 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cabsf.c

1

1139 Tue Nov 25 12:58:02 2014

new/usr/src/lib/libm/common/complex/cabsf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cabsf = cabsf
30 #pragma weak cabsf = __cabsf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 float
36 cabsf(fcomplex z) {
37     return (hypotf(F_RE(z), F_IM(z)));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cabs1.c

1

1148 Tue Nov 25 12:58:03 2014

new/usr/src/lib/libm/common/complex/cabs1.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cabs1 = cabs1
30 #pragma weak cabs1 = __cabs1
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 long double
36 cabs1(ldcomplex z) {
37     return (hypot1(LD_RE(z), LD_IM(z)));
38 }
```

unchanged_portion_omitted

```

*****
12776 Tue Nov 25 12:58:03 2014
new/usr/src/lib/libm/common/complex/cacos.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cacos = cacos
30 #pragma weak cacos = __cacos

32 /* INDENT OFF */
33 /*
34  * dcomplex cacos(dcomplex z);
35  *
36  * Alogrithm
37  * (based on T.E.Hull, Thomas F. Fairgrieve and Ping Tak Peter Tang's
38  * paper "Implementing the Complex Arcsine and Arccosine Functins Using
39  * Exception Handling", ACM TOMS, Vol 23, pp 299-335)
40  *
41  * The principal value of complex inverse cosine function cacos(z),
42  * where z = x+iy, can be defined by
43  *
44  *      cacos(z) = acos(B) - i sign(y) log (A + sqrt(A*A-1)),
45  *
46  * where the log function is the natural log, and
47  *
48  *      A = --- / (x+1) + y + --- / (x-1) + y
49  *          2 \ /          2 \ /
50  *
51  *
52  *      B = --- / (x+1) + y - --- / (x-1) + y .
53  *          2 \ /          2 \ /
54  *
55  *
56  * The Branch cuts are on the real line from -inf to -1 and from 1 to inf.
57  * The real and imaginary parts are based on Abramowitz and Stegun

```

```

58 * [Handbook of Mathematic Functions, 1972]. The sign of the imaginary
59 * part is chosen to be the generally considered the principal value of
60 * this function.
61 *
62 * Notes:1. A is the average of the distances from z to the points (1,0)
63 *        and (-1,0) in the complex z-plane, and in particular A>=1.
64 *        2. B is in [-1,1], and A*B = x
65 *
66 * Basic relations
67 *      cacos(conj(z)) = conj(cacos(z))
68 *      cacos(-z)      = pi - cacos(z)
69 *      cacos( z )     = pi/2 - casin(z)
70 *
71 * Special cases (conform to ISO/IEC 9899:1999(E)):
72 *      cacos(+0 + i y ) = pi/2 - i y for y is +-0, +-inf, NaN
73 *      cacos( x + i inf) = pi/2 - i inf for all x
74 *      cacos( x + i NaN) = NaN + i NaN with invalid for non-zero finite x
75 *      cacos(-inf + i y ) = pi - i inf for finite +y
76 *      cacos( inf + i y ) = 0 - i inf for finite +y
77 *      cacos(-inf + i inf) = 3pi/4- i inf
78 *      cacos( inf + i inf) = pi/4 - i inf
79 *      cacos(+-inf+ i NaN) = NaN - i inf (sign of imaginary is unspecified)
80 *      cacos(NaN + i y ) = NaN + i NaN with invalid for finite y
81 *      cacos(NaN + i inf) = NaN - i inf
82 *      cacos(NaN + i NaN) = NaN + i NaN
83 *
84 * Special Regions (better formula for accuracy and for avoiding spurious
85 * overflow or underflow) (all x and y are assumed nonnegative):
86 * case 1: y = 0
87 * case 2: tiny y relative to x-1: y <= ulp(0.5)*|x-1|
88 * case 3: tiny y: y < 4 sqrt(u), where u = minimum normal number
89 * case 4: huge y relative to x+1: y >= (1+x)/ulp(0.5)
90 * case 5: huge x and y: x and y >= sqrt(M)/8, where M = maximum normal number
91 * case 6: tiny x: x < 4 sqrt(u)
92 * -----
93 * case 1 & 2. y=0 or y/|x-1| is tiny. We have
94 *
95 *
96 *      / / (x+1) + y = |x+1| / / 1 + (-----)
97 *          2 \ /          2 \ /          |x+1|
98 *
99 *
100 *      ~ |x+1| ( 1 + --- (-----) )
101 *                   2 |x+1|
102 *
103 *
104 *      2
105 *      = |x+1| + -----
106 *                   2|x+1|
107 *
108 * Consequently, it is not difficult to see that
109 *
110 *
111 *      [ 1 + ----- , if x < 1,
112 *        2(1+x)(1-x)
113 *
114 *
115 *      [ x, if x = 1 (y = 0),
116 *
117 *      [
118 *        x * y
119 *      [ x + ----- ~ x, if x > 1
120 *        2(x+1)(x-1)
121 *
122 * and hence
123 *

```

```

124 *
125 *      A + \ / A - 1 ~ 1 + ----- + -----, if x < 1,
126 *                      2
127 *                      sqrt((x+1)(1-x)) 2(x+1)(1-x)
128 *
129 *      ~ x + sqrt((x-1)*(x+1)),          if x >= 1.
130 *
131 *
132 *      2
133 *      [ x(1 - -----) ~ x,   if x < 1,
134 *      [   2(1+x)(1-x)
135 *      B = x/A ~ [
136 *      [ 1,                      if x = 1,
137 *      [
138 *      [   2
139 *      [ 1 - -----,          if x > 1,
140 *      [   2(x+1)(x-1)
141 *
142 *      Thus
143 *      [ acos(x) - i y/sqrt((x-1)*(x+1)),   if x < 1,
144 *      [
145 *      cacos(x+i*y)~ [ 0 - i 0,              if x = 1,
146 *      [
147 *      [ y/sqrt(x*x-1) - i log(x+sqrt(x*x-1)), if x > 1.
148 *
149 *      Note: y/sqrt(x*x-1) ~ y/x when x >= 2**26.
150 *      case 3. y < 4 sqrt(u), where u = minimum normal x.
151 *      After case 1 and 2, this will only occurs when x=1. When x=1, we have
152 *      A = (sqrt(4+y*y)+y)/2 ~ 1 + y/2 + y^2/8 + ...
153 *      and
154 *      B = 1/A = 1 - y/2 + y^2/8 + ...
155 *      Since
156 *      cos(sqrt(y)) ~ 1 - y/2 + ...
157 *      we have, for the real part,
158 *      acos(B) ~ acos(1 - y/2) ~ sqrt(y)
159 *      For the imaginary part,
160 *      log(A+sqrt(A*A-1)) ~ log(1+y/2+sqrt(2*y/2))
161 *      = log(1+y/2+sqrt(y))
162 *      = (y/2+sqrt(y)) - (y/2+sqrt(y))^2/2 + ...
163 *      ~ sqrt(y) - y*(sqrt(y)+y/2)/2
164 *      ~ sqrt(y)
165 *
166 *      case 4. y >= (x+1)/ulp(0.5). In this case, A ~ y and B ~ x/y. Thus
167 *      real part = acos(B) ~ pi/2
168 *      and
169 *      imag part = log(y+sqrt(y*y-one))
170 *
171 *      case 5. Both x and y are large: x and y > sqrt(M)/8, where M = maximum x
172 *      In this case,
173 *      A ~ sqrt(x*x+y*y)
174 *      B ~ x/sqrt(x*x+y*y).
175 *      Thus
176 *      real part = acos(B) = atan(y/x),
177 *      imag part = log(A+sqrt(A*A-1)) ~ log(2A)
178 *      = log(2) + 0.5*log(x*x+y*y)
179 *      = log(2) + log(y) + 0.5*log(1+(x/y)^2)
180 *
181 *      case 6. x < 4 sqrt(u). In this case, we have
182 *      A ~ sqrt(1+y*y), B = x/sqrt(1+y*y).
183 *      Since B is tiny, we have
184 *      real part = acos(B) ~ pi/2
185 *      imag part = log(A+sqrt(A*A-1)) = log(A+sqrt(y*y))
186 *      = log(y+sqrt(1+y*y))
187 *      = 0.5*log(y^2+2y*sqrt(1+y^2)+1+y^2)
188 *      = 0.5*log(1+2y(y+sqrt(1+y^2)));
189 *      = 0.5*loglp(2y(y+A));

```

```

190 *
191 *      cacos(z) = acos(B) - i sign(y) log(A + sqrt(A*A-1)),
192 *
193 /* INDENT ON */
194
195 #include "libm.h"
196 #include "complex_wrapper.h"
197
198 /* INDENT OFF */
199 static const double
200     zero = 0.0,
201     one = 1.0,
202     E = 1.11022302462515654042e-16,          /* 2**(-53) */
203     ln2 = 6.93147180559945286227e-01,
204     pi = 3.1415926535897931159979634685,
205     pi_1 = 1.224646799147353177e-16,
206     pi_2 = 1.570796326794896558e+00,
207     pi_2_1 = 6.123233995736765886e-17,
208     pi_4 = 0.78539816339744827899949,
209     pi_4_1 = 3.061616997868382943e-17,
210     pi3_4 = 2.356194490192344836998,
211     pi3_4_1 = 9.184850993605148829195e-17,
212     Foursqrts = 5.96667258496016539463e-154, /* 2**(-509) */
213     Acrossover = 1.5,
214     Bcrossover = 0.6417,
215     half = 0.5;
216 /* INDENT ON */
217
218 dcomplex
219 cacos(dcomplex z) {
220     double x, y, t, R, S, A, Aml, B, y2, xml, xpl, Apx;
221     int ix, iy, hx, hy;
222     unsigned lx, ly;
223     dcomplex ans;
224
225     x = D_RE(z);
226     y = D_IM(z);
227     hx = HI_WORD(x);
228     lx = LO_WORD(x);
229     hy = HI_WORD(y);
230     ly = LO_WORD(y);
231     ix = hx & 0x7fffffff;
232     iy = hy & 0x7fffffff;
233
234     /* x is 0 */
235     if ((ix | lx) == 0) {
236         if (((iy | ly) == 0) || (iy >= 0x7ff00000)) {
237             D_RE(ans) = pi_2;
238             D_IM(ans) = -y;
239             return (ans);
240         }
241     }
242
243     /* |y| is inf or NaN */
244     if (iy >= 0x7ff00000) {
245         if (ISINF(iy, ly)) { /* cacos(x + i inf) = pi/2 - i inf */
246             D_IM(ans) = -y;
247             if (ix < 0x7ff00000) {
248                 D_RE(ans) = pi_2 + pi_2_1;
249             } else if (ISINF(ix, lx)) {
250                 if (hx >= 0)
251                     D_RE(ans) = pi_4 + pi_4_1;
252                 else
253                     D_RE(ans) = pi3_4 + pi3_4_1;
254             } else {
255                 D_RE(ans) = x;

```

```

256     } else {
257         /* cacos(x + i NaN) = NaN + i NaN */
258         D_RE(ans) = y + x;
259         if (ISINF(ix, lx))
260             D_IM(ans) = -fabs(x);
261         else
262             D_IM(ans) = y;
263     }
264     return (ans);
265 }
267 x = fabs(x);
268 y = fabs(y);
270 /* x is inf or NaN */
271 if (ix >= 0x7ff00000) { /* x is inf or NaN */
272     if (ISINF(ix, lx)) { /* x is INF */
273         D_IM(ans) = -x;
274         if (iy >= 0x7ff00000) {
275             if (ISINF(iy, ly)) {
276                 /* INDENT OFF */
277                 /* cacos(inf + i inf) = pi/4 - i inf */
278                 /* cacos(-inf+ i inf) =3pi/4 - i inf */
279                 /* INDENT ON */
280                 if (hx >= 0)
281                     D_RE(ans) = pi_4 + pi_4_l;
282                 else
283                     D_RE(ans) = pi3_4 + pi3_4_l;
284             } else
285                 /* INDENT OFF */
286                 /* cacos(inf + i NaN) = NaN - i inf */
287                 /* INDENT ON */
288                 D_RE(ans) = y + y;
289         } else
290             /* INDENT OFF */
291             /* cacos(inf + iy) = 0 - i inf */
292             /* cacos(-inf+ iy) = pi - i inf */
293             /* INDENT ON */
294             if (hx >= 0)
295                 D_RE(ans) = zero;
296             else
297                 D_RE(ans) = pi + pi_l;
298     } else { /* x is NaN */
299         /* INDENT OFF */
300         /*
301          * cacos(NaN + i inf) = NaN - i inf
302          * cacos(NaN + i y) = NaN + i NaN
303          * cacos(NaN + i NaN) = NaN + i NaN
304          */
305         /* INDENT ON */
306         D_RE(ans) = x + y;
307         if (iy >= 0x7ff00000) {
308             D_IM(ans) = -y;
309         } else {
310             D_IM(ans) = x;
311         }
312     }
313     if (hy < 0)
314         D_IM(ans) = -D_IM(ans);
315     return (ans);
316 }
318 if ((iy | ly) == 0) { /* region 1: y=0 */
319     if (ix < 0x3ff00000) { /* |x| < 1 */
320         D_RE(ans) = acos(x);
321         D_IM(ans) = zero;

```

```

322     } else {
323         D_RE(ans) = zero;
324         if (ix >= 0x43500000) /* |x| >= 2**54 */
325             D_IM(ans) = ln2 + log(x);
326         else if (ix >= 0x3ff80000) /* x > Acrossover */
327             D_IM(ans) = log(x + sqrt((x - one) * (x +
328                                     one)));
329         else {
330             xml = x - one;
331             D_IM(ans) = loglp(xml + sqrt(xml * (x + one)));
332         }
333     }
334 } else if (y <= E * fabs(x - one)) { /* region 2: y < tiny*|x-1| */
335     if (ix < 0x3ff00000) { /* x < 1 */
336         D_RE(ans) = acos(x);
337         D_IM(ans) = y / sqrt((one + x) * (one - x));
338     } else if (ix >= 0x43500000) { /* |x| >= 2**54 */
339         D_RE(ans) = y / x;
340         D_IM(ans) = ln2 + log(x);
341     } else {
342         t = sqrt((x - one) * (x + one));
343         D_RE(ans) = y / t;
344         if (ix >= 0x3ff80000) /* x > Acrossover */
345             D_IM(ans) = log(x + t);
346         else
347             D_IM(ans) = loglp((x - one) + t);
348     }
349 } else if (y < Foursqrtu) { /* region 3 */
350     t = sqrt(y);
351     D_RE(ans) = t;
352     D_IM(ans) = t;
353 } else if (E * y - one >= x) { /* region 4 */
354     D_RE(ans) = pi_2;
355     D_IM(ans) = ln2 + log(y);
356 } else if (ix >= 0x5fc00000 || iy >= 0x5fc00000) { /* x,y>2**509 */
357     /* region 5: x+1 or y is very large (>= sqrt(max)/8) */
358     t = x / y;
359     D_RE(ans) = atan(y / x);
360     D_IM(ans) = ln2 + log(y) + half * loglp(t * t);
361 } else if (x < Foursqrtu) {
362     /* region 6: x is very small, < 4sqrt(min) */
363     D_RE(ans) = pi_2;
364     A = sqrt(one + y * y);
365     if (iy >= 0x3ff80000) /* if y > Acrossover */
366         D_IM(ans) = log(y + A);
367     else
368         D_IM(ans) = half * loglp((y + y) * (y + A));
369 } else { /* safe region */
370     y2 = y * y;
371     xpl = x + one;
372     xml = x - one;
373     R = sqrt(xpl * xpl + y2);
374     S = sqrt(xml * xml + y2);
375     A = half * (R + S);
376     B = x / A;
377     if (B <= Bcrossover)
378         D_RE(ans) = acos(B);
379     else { /* use atan and an accurate approx to a-x */
380         Apx = A + x;
381         if (x <= one)
382             D_RE(ans) = atan(sqrt(half * Apx * (y2 / (R +
383                                     xpl) + (S - xml))) / x);
384         else
385             D_RE(ans) = atan((y * sqrt(half * (Apx / (R +
386                                     xpl) + Apx / (S + xml)))) / x);
387     }

```

```
388         if (A <= Acrossover) {
389             /* use loglp and an accurate approx to A-1 */
390             if (x < one)
391                 Aml = half * (y2 / (R + xpl) + y2 / (S - xml));
392             else
393                 Aml = half * (y2 / (R + xpl) + (S + xml));
394             D_IM(ans) = loglp(Aml + sqrt(Aml * (A + one)));
395         } else {
396             D_IM(ans) = log(A + sqrt(A * A - one));
397         }
398     }
399     if (hx < 0)
400         D_RE(ans) = pi - D_RE(ans);
401     if (hy >= 0)
402         D_IM(ans) = -D_IM(ans);
403     return (ans);
404 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/complex/cacosf.c

1

```
*****
1313 Tue Nov 25 12:58:04 2014
new/usr/src/lib/libm/common/complex/cacosf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cacosf = cacosf
30 #pragma weak cacosf = __cacosf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 cacosf(fcomplex z) {
37     dcomplex dz, dans;
38     fcomplex ans;

40     D_RE(dz) = (double) (F_RE(z));
41     D_IM(dz) = (double) (F_IM(z));
42     dans = cacos(dz);
43     F_RE(ans) = (float) (D_RE(dans));
44     F_IM(ans) = (float) (D_IM(dans));
45     return (ans);
46 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cacosh.c

1

```
*****
1735 Tue Nov 25 12:58:04 2014
new/usr/src/lib/libm/common/complex/cacosh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __cacosh = cacosh
30 #pragma weak cacosh = __cacosh
31
32 /* INDENT OFF */
33 /*
34  * dcomplex cacosh(dcomplex z);
35  * cacosh z = +-i cacos z .
36  * In order to make conj(cacosh(z))=cacosh(conj(z)),
37  * we define
38  * cacosh z = sign(Im(z))*i cacos z .
39  *
40  */
41 /* INDENT ON */
42
43 #include "libm.h" /* fabs/isnan/isinf/signbit */
44 #include "complex_wrapper.h"
45
46 /* need to work on special cases according to spec */
47
48 dcomplex
49 cacosh(dcomplex z) {
50     dcomplex w, ans;
51     double x, y;
52
53     w = cacos(z);
54     x = D_RE(w);
55     y = D_IM(w);
56     if (isnan(y)) {
57         D_IM(ans) = y + y;
```

new/usr/src/lib/libm/common/complex/cacosh.c

2

```
58         if (isinf(x))
59             D_RE(ans) = fabs(x);
60         else
61             D_RE(ans) = y;
62     } else if (signbit(y) == 0) {
63         D_RE(ans) = -D_IM(w);
64         D_IM(ans) = D_RE(w);
65     } else {
66         D_RE(ans) = D_IM(w);
67         D_IM(ans) = -D_RE(w);
68     }
69     return (ans);
70 }
unchanged_portion_omitted
```


new/usr/src/lib/libm/common/complex/cacoshf.c

1

1372 Tue Nov 25 12:58:05 2014

new/usr/src/lib/libm/common/complex/cacoshf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cacoshf = cacoshf
30 #pragma weak cacoshf = __cacoshf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 /* need to work on special cases according to spec */
```

```
37 fcomplex
38 cacoshf(fcomplex z) {
39     dcomplex dz, dans;
40     fcomplex ans;
41
42     D_RE(dz) = (double) (F_RE(z));
43     D_IM(dz) = (double) (F_IM(z));
44     dans = cacosh(dz);
45     F_RE(ans) = (float) (D_RE(dans));
46     F_IM(ans) = (float) (D_IM(dans));
47     return (ans);
48 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/complex/cacoshl.c

1

```
*****
1740 Tue Nov 25 12:58:05 2014
new/usr/src/lib/libm/common/complex/cacoshl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cacoshl = cacoshl
30 #pragma weak cacoshl = __cacoshl

32 #include "libm.h" /* fabsl/isnanl/isinfl/signbitl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"

36 /* INDENT OFF */
37 /*
38  * ldcomplex cacoshl(ldcomplex z);
39  * cacosh z = +-i cacos z .
40  * In order to make conj(cacosh(z))=cacosh(conj(z)),
41  * we define
42  * cacosh z = sign(Im(z))*i cacos z .
43  *
44  */
45 /* INDENT ON */

47 ldcomplex
48 cacoshl(ldcomplex z) {
49     ldcomplex w, ans;
50     long double x, y;

52     w = cacosl(z);
53     x = LD_RE(z);
54     y = LD_IM(z);
55     if (isnanl(y)) {
56         LD_IM(ans) = y + y;
57         if (isinfl(x))
```

new/usr/src/lib/libm/common/complex/cacoshl.c

2

```
58         LD_RE(ans) = fabsl(x);
59         else
60             LD_RE(ans) = y;
61     } else if (signbitl(y) == 0) {
62         LD_RE(ans) = -LD_IM(w);
63         LD_IM(ans) = LD_RE(w);
64     } else {
65         LD_RE(ans) = LD_IM(w);
66         LD_IM(ans) = -LD_RE(w);
67     }
68     return (ans);
69 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cacosl.c

1

```
*****
7682 Tue Nov 25 12:58:05 2014
new/usr/src/lib/libm/common/complex/cacosl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cacosl = cacosl
30 #pragma weak cacosl = __cacosl

32 #include "libm.h" /* acosl/atanl/fabsl/isinfl/loglpl/logl/sqrtl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"

36 /* INDENT OFF */
37 static const long double
38 zero = 0.0L,
39 one = 1.0L,
40 Acrossover = 1.5L,
41 Bcrossover = 0.6417L,
42 half = 0.5L,
43 ln2 = 6.931471805599453094172321214581765680755e-0001L,
44 Foursqrtu = 7.3344154702193886624856495681939326638255e-2466L, /* 2**-8189 */
45 #if defined(__x86)
46 E = 5.4210108624275221700372640043497085571289e-20L, /* 2**-64 */
47 pi = 3.141592653589793238295968524909085317631252110004425048828125L,
48 pi_1 = 1.666748583704175665659172893706807721468195923078e-19L,
49 pi_2 = 1.5707963267948966191479842624545426588156260L,
50 pi_2_1 = 8.3337429185208783282958644685340386073409796e-20L,
51 pi_4 = 0.78539816339744830957399213122727132940781302750110626220703125L,
52 pi_4_1 = 4.166871459260439164147932234267019303670489807695410e-20L,
53 pi3_4 = 2.35619449019234492872197639368181398822343908250331878662109375L,
54 pi3_4_1 = 1.250061437778131749244379670280105791101146942308e-19L;
55 #else
56 E = 9.6296497219361792652798897129246365926905e-35L, /* 2**-113 */
57 pi = 3.1415926535897932384626433832795027974790680981372955730045043318L,
```

new/usr/src/lib/libm/common/complex/cacosl.c

2

```
58 pi_1 = 8.6718101301237810247970440260433519687623233462565303417759356862e-35L,
59 pi_2 = 1.5707963267948966192313216916397513987395340L,
60 pi_2_1 = 4.3359050650618905123985220130216759843811616e-35L,
61 pi_4 = 0.785398163397448309615660845819875699369767024534323893251126L,
62 pi_4_1 = 2.167952532530945256199261006510837992190580836564132585443e-35L,
63 pi3_4 = 2.35619449019234492884698253745962709810930107360297167975337824L,
64 pi3_4_1 = 6.503857597592835768597783019532513976571742509692397756331e-35L;
65 #endif
66 /* INDENT ON */

68 #if defined(__x86)
69 static const int ip1 = 0x40400000; /* 2**65 */
70 #else
71 static const int ip1 = 0x40710000; /* 2**114 */
72 #endif

74 ldcomplex
75 cacosl(ldcomplex z) {
76     long double x, y, t, R, S, A, Aml, B, y2, xml, xpl, Apx;
77     int ix, iy, hx, hy;
78     ldcomplex ans;

80     x = LD_RE(z);
81     y = LD_IM(z);
82     hx = HI_XWORD(x);
83     hy = HI_XWORD(y);
84     ix = hx & 0x7fffffff;
85     iy = hy & 0x7fffffff;

87     /* x is 0 */
88     if (x == zero) {
89         if (y == zero || (iy >= 0x7fff0000)) {
90             LD_RE(ans) = pi_2 + pi_2_1;
91             LD_IM(ans) = -y;
92             return (ans);
93         }
94     }

96     /* |y| is inf or NaN */
97     if (iy >= 0x7fff0000) {
98         if (isinfl(y)) { /* cacos(x + i inf) = pi/2 - i inf */
99             LD_IM(ans) = -y;
100             if (ix < 0x7fff0000) {
101                 LD_RE(ans) = pi_2 + pi_2_1;
102             } else if (isinfl(x)) {
103                 if (hx >= 0)
104                     LD_RE(ans) = pi_4 + pi_4_1;
105                 else
106                     LD_RE(ans) = pi3_4 + pi3_4_1;
107             } else {
108                 LD_RE(ans) = x;
109             }
110         } else { /* cacos(x + i NaN) = NaN + i NaN */
111             LD_RE(ans) = y + x;
112             if (isinfl(x))
113                 LD_IM(ans) = -fabsl(x);
114             else
115                 LD_IM(ans) = y;
116         }
117         return (ans);
118     }

120     y = fabsl(y);

122     if (ix >= 0x7fff0000) { /* x is inf or NaN */
123         if (isinfl(x)) { /* x is INF */
```

```

124 LD_IM(ans) = -fabsl(x);
125 if (iy >= 0x7fff0000) {
126     if (isinfl(y)) {
127         /* INDEXT OFF */
128         /* cacos(Inf + i Inf) = pi/4 - i Inf */
129         /* cacos(-Inf + i Inf) = 3pi/4 - i Inf */
130         /* INDEXT ON */
131         if (hx >= 0)
132             LD_RE(ans) = pi_4 + pi_4_l;
133         else
134             LD_RE(ans) = pi3_4 + pi3_4_l;
135     } else
136         /* INDEXT OFF */
137         /* cacos(Inf + i NaN) = NaN - i Inf */
138         /* INDEXT ON */
139         LD_RE(ans) = y + y;
140     } else {
141         /* INDEXT OFF */
142         /* cacos(Inf + iy) = 0 - i Inf */
143         /* cacos(-Inf + iy) = pi - i Inf */
144         /* INDEXT ON */
145         if (hx >= 0)
146             LD_RE(ans) = zero;
147         else
148             LD_RE(ans) = pi + pi_l;
149     }
150 } else { /* x is NaN */
151     /* INDEXT OFF */
152     /*
153     * cacos(NaN + i Inf) = NaN - i Inf
154     * cacos(NaN + i Y) = NaN + i NaN
155     * cacos(NaN + i NaN) = NaN + i NaN
156     */
157     /* INDEXT ON */
158     LD_RE(ans) = x + y;
159     if (iy >= 0x7fff0000) {
160         LD_IM(ans) = -y;
161     } else {
162         LD_IM(ans) = x;
163     }
164 }
165 if (hy < 0)
166     LD_IM(ans) = -LD_IM(ans);
167 return (ans);
168 }

170 if (y == zero) { /* region 1: y=0 */
171     if (ix < 0x3fff0000) { /* |x| < 1 */
172         LD_RE(ans) = acosl(x);
173         LD_IM(ans) = zero;
174     } else {
175         LD_RE(ans) = zero;
176         x = fabsl(x);
177         if (ix >= ip1) /* i386 ? 2**65 : 2**114 */
178             LD_IM(ans) = ln2 + logl(x);
179         else if (ix >= 0x3fff8000) /* x > Acrossover */
180             LD_IM(ans) = logl(x + sqrtl((x - one) * (x +
181 one)));
182         else {
183             xml = x - one;
184             LD_IM(ans) = loglpl(xml + sqrtl(xml * (x +
185 one)));
186         }
187     }
188 } else if (y <= E * fabsl(fabsl(x) - one)) {
189     /* region 2: y < tiny*||x|-1| */

```

```

190     if (ix < 0x3fff0000) { /* x < 1 */
191         LD_RE(ans) = acosl(x);
192         x = fabsl(x);
193         LD_IM(ans) = y / sqrtl((one + x) * (one - x));
194     } else if (ix >= ip1) { /* i386 ? 2**65 : 2**114 */
195         if (hx >= 0)
196             LD_RE(ans) = y / x;
197         else {
198             if (ix >= ip1 + 0x00040000)
199                 LD_RE(ans) = pi + pi_l;
200             else {
201                 t = pi_l + y / x;
202                 LD_RE(ans) = pi + t;
203             }
204         }
205     }
206     LD_IM(ans) = ln2 + logl(fabsl(x));
207 } else {
208     x = fabsl(x);
209     t = sqrtl((x - one) * (x + one));
210     LD_RE(ans) = (hx >= 0)? y / t : pi - (y / t - pi_l);
211     if (ix >= 0x3fff8000) /* x > Acrossover */
212         LD_IM(ans) = logl(x + t);
213     else
214         LD_IM(ans) = loglpl(t - (one - x));
215 }
216 } else if (y < Foursqrtu) { /* region 3 */
217     t = sqrtl(y);
218     LD_RE(ans) = (hx >= 0)? t : pi + pi_l;
219     LD_IM(ans) = t;
220 } else if (E * y - one >= fabsl(x)) { /* region 4 */
221     LD_RE(ans) = pi_2 + pi_2_l;
222     LD_IM(ans) = ln2 + logl(y);
223 } else if (ix >= 0x5ffb0000 || iy >= 0x5ffb0000) {
224     /* region 5: x+1 and y are both (>= sqrt(max)/8) i.e. 2**8188 */
225     t = x / y;
226     LD_RE(ans) = atan2l(y, x);
227     LD_IM(ans) = ln2 + logl(y) + half * loglpl(t * t);
228 } else if (fabsl(x) < Foursqrtu) {
229     /* region 6: x is very small, < 4sqrt(min) */
230     LD_RE(ans) = pi_2 + pi_2_l;
231     A = sqrtl(one + y * y);
232     if (iy >= 0x3fff8000) /* if y > Acrossover */
233         LD_IM(ans) = logl(y + A);
234     else
235         LD_IM(ans) = half * loglpl((y + y) * (y + A));
236 } else { /* safe region */
237     t = fabsl(x);
238     y2 = y * y;
239     xpl = t + one;
240     xml = t - one;
241     R = sqrtl(xpl * xpl + y2);
242     S = sqrtl(xml * xml + y2);
243     A = half * (R + S);
244     B = t / A;
245
246     if (B <= Bcrossover)
247         LD_RE(ans) = (hx >= 0)? acosl(B) : acosl(-B);
248     else { /* use atan and an accurate approx to a-x */
249         Apx = A + t;
250         if (t <= one)
251             LD_RE(ans) = atan2l(sqrtl(half * Apx * (y2 /
252 (R + xpl) + (S - xml))), x);
253         else
254             LD_RE(ans) = atan2l((y * sqrtl(half * (Apx /
255 (R + xpl) + Apx / (S + xml))))), x);
256     }

```

```
256         if (A <= Acrossover) {
257             /* use loglp and an accurate approx to A-1 */
258             if (ix < 0x3fff0000)
259                 Aml = half * (y2 / (R + xpl) + y2 / (S - xml));
260             else
261                 Aml = half * (y2 / (R + xpl) + (S + xml));
262             LD_IM(ans) = loglp(Aml + sqrtl(Aml * (A + one)));
263         } else {
264             LD_IM(ans) = logl(A + sqrtl(A * A - one));
265         }
266     }

268     if (hy >= 0)
269         LD_IM(ans) = -LD_IM(ans);

271     return (ans);
272 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/complex/carg.c

1

1547 Tue Nov 25 12:58:06 2014

new/usr/src/lib/libm/common/complex/carg.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 #pragma weak __carg = carg
29 #pragma weak carg = __carg
```

```
31 #include "libm_synonyms.h"
31 #include <math.h> /* atan2 */
32 #include "complex_wrapper.h"
```

```
34 static const double
35     pi = 3.14159265358979311600e+00,
36     pi_lo = 1.22464679914735320717e-16;
```

```
38 double
39 carg(dcomplex z) {
40     int ix, iy;

42     ix = ((int *)&(D_RE(z)))[HIWORD];
43     iy = ((int *)&(D_IM(z)))[HIWORD];
44     if (((ix | iy) & ~0x80000000) | ((int *)&(D_RE(z)))[LOWORD] |
45         ((int *)&(D_IM(z)))[LOWORD]) == 0) {
46         /* x and y are both zero */
47         if (ix == 0)
48             return (D_IM(z));
49         return ((iy == 0)? pi + pi_lo : -pi - pi_lo);
50     }
51     return (atan2(D_IM(z), D_RE(z)));
52 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/complex/cargf.c

1

1154 Tue Nov 25 12:58:06 2014

new/usr/src/lib/libm/common/complex/cargf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cargf = cargf
30 #pragma weak cargf = __cargf
```

```
32 #include "libm.h" /* atan2f */
33 #include "complex_wrapper.h"
```

```
35 float
36 cargf(fcomplex z) {
37     return (atan2f(F_IM(z), F_RE(z)));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cargl.c

1

1148 Tue Nov 25 12:58:07 2014

new/usr/src/lib/libm/common/complex/cargl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cargl = cargl
30 #pragma weak cargl = __cargl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 long double
36 cargl(ldcomplex z) {
37     return (atan2l(LD_IM(z), LD_RE(z)));
38 }
```

unchanged_portion_omitted


```

*****
12214 Tue Nov 25 12:58:07 2014
new/usr/src/lib/libm/common/complex/casin.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __casin = casin
30 #pragma weak casin = __casin

32 /* INDENT OFF */
33 /*
34  * dcomplex casin(dcomplex z);
35  *
36  * Alogrithm
37  * (based on T.E.Hull, Thomas F. Fairgrieve and Ping Tak Peter Tang's
38  * paper "Implementing the Complex Arcsine and Arccosine Functins Using
39  * Exception Handling", ACM TOMS, Vol 23, pp 299-335)
40  *
41  * The principal value of complex inverse sine function casin(z),
42  * where z = x+iy, can be defined by
43  *
44  *      casin(z) = asin(B) + i sign(y) log (A + sqrt(A*A-1)),
45  *
46  * where the log function is the natural log, and
47  *
48  *      A = --- / (x+1) + y + --- / (x-1) + y
49  *          2 \ /                2 \ /
50  *
51  *
52  *      B = --- / (x+1) + y - --- / (x-1) + y .
53  *          2 \ /                2 \ /
54  *
55  *
56  * The Branch cuts are on the real line from -inf to -1 and from 1 to inf.
57  * The real and imaginary parts are based on Abramowitz and Stegun

```

```

58  * [Handbook of Mathematic Functions, 1972]. The sign of the imaginary
59  * part is chosen to be the generally considered the principal value of
60  * this function.
61  *
62  * Notes:1. A is the average of the distances from z to the points (1,0)
63  *        and (-1,0) in the complex z-plane, and in particular A>=1.
64  *        2. B is in [-1,1], and A*B = x.
65  *
66  * Special notes: if casin( x, y) = ( u, v), then
67  *                casin(-x, y) = (-u, v),
68  *                casin( x,-y) = ( u,-v),
69  * in general, we have casin(conj(z)) = conj(casin(z))
70  *                casin(-z) = -casin(z)
71  *                casin(z) = pi/2 - cacos(z)
72  *
73  * EXCEPTION CASES (conform to ISO/IEC 9899:1999(E)):
74  * casin( 0 + i 0 ) = 0 + i 0
75  * casin( 0 + i NaN ) = 0 + i NaN
76  * casin( x + i inf ) = 0 + i inf for finite x
77  * casin( x + i NaN ) = NaN + i NaN with invalid for finite x != 0
78  * casin( inf + iy ) = pi/2 + i inf finite y
79  * casin( inf + i inf) = pi/4 + i inf
80  * casin( inf + i NaN) = NaN + i inf
81  * casin( NaN + i y ) = NaN + i NaN for finite y
82  * casin( NaN + i inf) = NaN + i inf
83  * casin( NaN + i NaN) = NaN + i NaN
84  *
85  * Special Regions (better formula for accuracy and for avoiding spurious
86  * overflow or underflow) (all x and y are assumed nonnegative):
87  * case 1: y = 0
88  * case 2: tiny y relative to x-1: y <= ulp(0.5)*|x-1|
89  * case 3: tiny y: y < 4 sqrt(u), where u = minimum normal number
90  * case 4: huge y relative to x+1: y >= (1+x)/ulp(0.5)
91  * case 5: huge x and y: x and y >= sqrt(M)/8, where M = maximum normal number
92  * case 6: tiny x: x < 4 sqrt(u)
93  * -----
94  * case 1 & 2. y=0 or y/|x-1| is tiny. We have
95  *
96  *      / / (x+1) + y = |x+1| / / 1 + (-----)
97  *      \ \                \ / |x+1|
98  *
99  *
100 *
101 *      ~ |x+1| ( 1 + --- (-----) )
102 *                    2 |x+1|
103 *
104 *
105 *      2
106 *      = |x+1| + -----
107 *                    2|x+1|
108 *
109 * Consequently, it is not difficult to see that
110 *
111 *
112 *      [ 1 + ----- , if x < 1,
113 *        2(1+x)(1-x)
114 *
115 *
116 *      [ x, if x = 1 (y = 0),
117 *
118 *      A ~ = [
119 *              x * y
120 *      [ x + ----- , if x > 1
121 *              2(1+x)(x-1)
122 *
123 * and hence

```

```

124 *
125 *
126 *      A + \sqrt{A^2 - 1} ~ 1 + \frac{y}{\sqrt{(x+1)(1-x)}} + \frac{y^2}{2(x+1)(1-x)}, if x < 1,
127 *
128 *
129 *
130 *      ~ x + \sqrt{(x-1)(x+1)}, if x >= 1.
131 *
132 *
133 *
134 *      B = x/A ~ [ x(1 - \frac{y^2}{2(1+x)(1-x)}), if x < 1,
135 *                [
136 *                  [ 1, if x = 1,
137 *                  [
138 *                    [ 1 - \frac{y^2}{2(1+x)(1-x)}, if x > 1,
139 *                    [
140 *                      [
141 *                        [
142 *                          [
143 *
144 *      Thus
145 *      casin(x+i*y)=[
146 *                    [ pi/2 + i log(x+\sqrt{x*x-1}), if x >= 1
147 *
148 * case 3. y < 4 sqrt(u), where u = minimum normal x.
149 * After case 1 and 2, this will only occurs when x=1. When x=1, we have
150 * A = (\sqrt{4+y*y}+y)/2 ~ 1 + y/2 + y^2/8 + ...
151 * and
152 * B = 1/A = 1 - y/2 + y^2/8 + ...
153 * Since
154 * asin(x) = pi/2-2*asin(\sqrt{(1-x)/2})
155 * asin(x) = x + x^3/6 + x^5*3/40 + x^7*15/336 + ...
156 * we have, for the real part asin(B),
157 * asin(1-y/2) ~ pi/2 - 2 asin(\sqrt{y/4})
158 * ~ pi/2 - \sqrt{y}
159 * For the imaginary part,
160 * log(A+\sqrt{A*A-1}) ~ log(1+y/2+\sqrt{2*y/2})
161 * = log(1+y/2+\sqrt{y})
162 * = (y/2+\sqrt{y}) - (y/2+\sqrt{y})^2/2 + ...
163 * ~ \sqrt{y} - y*(\sqrt{y}+y/2)/2
164 * ~ \sqrt{y}
165 *
166 * case 4. y >= (x+1)ulp(0.5). In this case, A ~ y and B ~ x/y. Thus
167 * real part = asin(B) ~ x/y (be careful, x/y may underflow)
168 * and
169 * imag part = log(y+\sqrt{y*y-one})
170 *
171 *
172 * case 5. Both x and y are large: x and y > sqrt(M)/8, where M = maximum x
173 * In this case,
174 * A ~ \sqrt{x*x+y*y}
175 * B ~ x/\sqrt{x*x+y*y}.
176 * Thus
177 * real part = asin(B) = atan(x/y),
178 * imag part = log(A+\sqrt{A*A-1}) ~ log(2A)
179 * = log(2) + 0.5*log(x*x+y*y)
180 * = log(2) + log(y) + 0.5*log(1+(x/y)^2)
181 *
182 * case 6. x < 4 sqrt(u). In this case, we have
183 * A ~ \sqrt{1+y*y}, B = x/\sqrt{1+y*y}.
184 * Since B is tiny, we have
185 * real part = asin(B) ~ B = x/\sqrt{1+y*y}
186 * imag part = log(A+\sqrt{A*A-1}) = log(A+\sqrt{y*y})
187 * = log(y+\sqrt{1+y*y})
188 * = 0.5*log(y^2+2y\sqrt{1+y^2}+1+y^2)
189 * = 0.5*log(1+2y(y+\sqrt{1+y^2}));

```

```

190 *      = 0.5*loglp(2y(y+A));
191 *
192 *      casin(z) = asin(B) + i sign(y) log(A + \sqrt{A*A-1}),
193 */
194 /* INDENT ON */
195
196 #include "libm.h" /* asin/atan/fabs/log/loglp/sqrt */
197 #include "complex_wrapper.h"
198
199 /* INDENT OFF */
200 static const double
201 zero = 0.0,
202 one = 1.0,
203 E = 1.11022302462515654042e-16, /* 2**-53 */
204 ln2 = 6.93147180559945286227e-01,
205 pi_2 = 1.570796326794896558e+00,
206 pi_2_1 = 6.123233995736765886e-17,
207 pi_4 = 7.85398163397448278999e-01,
208 Foursqrtu = 5.96667258496016539463e-154, /* 2**(-509) */
209 Acrossover = 1.5,
210 Bcrossover = 0.6417,
211 half = 0.5;
212 /* INDENT ON */
213
214 dcomplex
215 casin(dcomplex z) {
216     double x, y, t, R, S, A, Aml, B, y2, xml, xpl, Apx;
217     int ix, iy, hx, hy;
218     unsigned lx, ly;
219     dcomplex ans;
220
221     x = D_RE(z);
222     y = D_IM(z);
223     hx = HI_WORD(x);
224     lx = LO_WORD(x);
225     hy = HI_WORD(y);
226     ly = LO_WORD(y);
227     ix = hx & 0x7fffffff;
228     iy = hy & 0x7fffffff;
229     x = fabs(x);
230     y = fabs(y);
231
232     /* special cases */
233
234     /* x is inf or NaN */
235     if (ix >= 0x7ff00000) { /* x is inf or NaN */
236         if (ISINF(ix, lx)) { /* x is INF */
237             D_IM(ans) = x;
238             if (iy >= 0x7ff00000) {
239                 if (ISINF(iy, ly))
240                     /* casin(inf + i inf) = pi/4 + i inf */
241                     D_RE(ans) = pi_4;
242                 else /* casin(inf + i NaN) = NaN + i inf */
243                     D_RE(ans) = y + y;
244             } else /* casin(inf + iy) = pi/2 + i inf */
245                 D_RE(ans) = pi_2;
246         } else { /* x is NaN */
247             if (iy >= 0x7ff00000) {
248                 /* INDENT OFF */
249                 /*
250                  * casin(NaN + i inf) = NaN + i inf
251                  * casin(NaN + i NaN) = NaN + i NaN
252                  */
253                 /* INDENT ON */
254                 D_IM(ans) = y + y;
255                 D_RE(ans) = x + x;

```

```

256     } else {
257         /* casin(NaN + i y ) = NaN + i NaN */
258         D_IM(ans) = D_RE(ans) = x + y;
259     }
260 }
261 if (hx < 0)
262     D_RE(ans) = -D_RE(ans);
263 if (hy < 0)
264     D_IM(ans) = -D_IM(ans);
265 return (ans);
266 }

268 /* casin(+0 + i 0 ) = 0 + i 0. */
269 if ((ix | lx | iy | ly) == 0)
270     return (z);

272 if (iy >= 0x7ff00000) { /* y is inf or NaN */
273     if (ISINF(iy, ly)) { /* casin(x + i inf) = 0 + i inf */
274         D_IM(ans) = y;
275         D_RE(ans) = zero;
276     } else { /* casin(x + i NaN) = NaN + i NaN */
277         D_IM(ans) = x + y;
278         if ((ix | lx) == 0)
279             D_RE(ans) = x;
280         else
281             D_RE(ans) = y;
282     }
283     if (hx < 0)
284         D_RE(ans) = -D_RE(ans);
285     if (hy < 0)
286         D_IM(ans) = -D_IM(ans);
287     return (ans);
288 }

290 if ((iy | ly) == 0) { /* region 1: y=0 */
291     if (ix < 0x3ff00000) { /* |x| < 1 */
292         D_RE(ans) = asin(x);
293         D_IM(ans) = zero;
294     } else {
295         D_RE(ans) = pi_2;
296         if (ix >= 0x43500000) /* |x| >= 2**54 */
297             D_IM(ans) = ln2 + log(x);
298         else if (ix >= 0x3ff80000) /* x > Acrossover */
299             D_IM(ans) = log(x + sqrt((x - one) * (x +
300 one)));
301         else {
302             xml = x - one;
303             D_IM(ans) = loglp(xml + sqrt(xml * (x + one)));
304         }
305     }
306 } else if (y <= E * fabs(x - one)) { /* region 2: y < tiny*|x-1| */
307     if (ix < 0x3ff00000) { /* x < 1 */
308         D_RE(ans) = asin(x);
309         D_IM(ans) = y / sqrt((one + x) * (one - x));
310     } else {
311         D_RE(ans) = pi_2;
312         if (ix >= 0x43500000) { /* |x| >= 2**54 */
313             D_IM(ans) = ln2 + log(x);
314         } else if (ix >= 0x3ff80000) /* x > Acrossover */
315             D_IM(ans) = log(x + sqrt((x - one) * (x +
316 one)));
317         else
318             D_IM(ans) = loglp((x - one) + sqrt((x - one) *
319 (x + one)));
320     }
321 } else if (y < Foursqrtu) { /* region 3 */

```

```

322     t = sqrt(y);
323     D_RE(ans) = pi_2 - (t - pi_2_1);
324     D_IM(ans) = t;
325 } else if (E * y - one >= x) { /* region 4 */
326     D_RE(ans) = x / y; /* need to fix underflow cases */
327     D_IM(ans) = ln2 + log(y);
328 } else if (ix >= 0x5fc00000 || iy >= 0x5fc00000) { /* x,y>2**509 */
329     /* region 5: x+1 or y is very large (>= sqrt(max)/8) */
330     t = x / y;
331     D_RE(ans) = atan(t);
332     D_IM(ans) = ln2 + log(y) + half * loglp(t * t);
333 } else if (x < Foursqrtu) {
334     /* region 6: x is very small, < 4sqrt(min) */
335     A = sqrt(one + y * y);
336     D_RE(ans) = x / A; /* may underflow */
337     if (iy >= 0x3ff80000) /* if y > Acrossover */
338         D_IM(ans) = log(y + A);
339     else
340         D_IM(ans) = half * loglp((y + y) * (y + A));
341 } else { /* safe region */
342     y2 = y * y;
343     xpl = x + one;
344     xml = x - one;
345     R = sqrt(xpl * xpl + y2);
346     S = sqrt(xml * xml + y2);
347     A = half * (R + S);
348     B = x / A;

350     if (B <= Bcrossover)
351         D_RE(ans) = asin(B);
352     else { /* use atan and an accurate approx to a-x */
353         Apx = A + x;
354         if (x <= one)
355             D_RE(ans) = atan(x / sqrt(half * Apx * (y2 /
356 (R + xpl) + (S - xml))));
357         else
358             D_RE(ans) = atan(x / (y * sqrt(half * (Apx /
359 (R + xpl) + Apx / (S + xml))));
360     }
361     if (A <= Acrossover) {
362         /* use loglp and an accurate approx to A-1 */
363         if (x < one)
364             Aml = half * (y2 / (R + xpl) + y2 / (S - xml));
365         else
366             Aml = half * (y2 / (R + xpl) + (S + xml));
367         D_IM(ans) = loglp(Aml + sqrt(Aml * (A + one)));
368     } else {
369         D_IM(ans) = log(A + sqrt(A * A - one));
370     }
371 }

373 if (hx < 0)
374     D_RE(ans) = -D_RE(ans);
375 if (hy < 0)
376     D_IM(ans) = -D_IM(ans);

378 return (ans);
379 }

```

unchanged portion omitted

new/usr/src/lib/libm/common/complex/casinf.c

1

1313 Tue Nov 25 12:58:07 2014

new/usr/src/lib/libm/common/complex/casinf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __casinf = casinf
30 #pragma weak casinf = __casinf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 casinf(fcomplex z) {
37     dcomplex dz, dans;
38     fcomplex ans;
39
40     D_RE(dz) = (double) (F_RE(z));
41     D_IM(dz) = (double) (F_IM(z));
42     dans = casin(dz);
43     F_RE(ans) = (float) (D_RE(dans));
44     F_IM(ans) = (float) (D_IM(dans));
45     return (ans);
46 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/casinh.c

1

1348 Tue Nov 25 12:58:08 2014

new/usr/src/lib/libm/common/complex/casinh.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __casinh = casinh
30 #pragma weak casinh = __casinh
```

```
32 /* INDENT OFF */
33 /*
34  * dcomplex casinh(dcomplex z);
35  * casinh z = -i casin iz .
36 */
37 /* INDENT ON */
```

```
39 #include "libm.h"
40 #include "complex_wrapper.h"
```

```
42 dcomplex
43 casinh(dcomplex z) {
44     dcomplex w, r, ans;
45
46     D_RE(w) = -D_IM(z);
47     D_IM(w) = D_RE(z);
48     r = casin(w);
49     D_RE(ans) = D_IM(r);
50     D_IM(ans) = -D_RE(r);
51     return (ans);
52 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/complex/casinhf.c

1

1251 Tue Nov 25 12:58:08 2014

new/usr/src/lib/libm/common/complex/casinhf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __casinhf = casinhf
30 #pragma weak casinhf = __casinhf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 casinhf(fcomplex z) {
37     fcomplex w, r, ans;
38
39     F_RE(w) = -F_IM(z);
40     F_IM(w) = F_RE(z);
41     r = casinf(w);
42     F_RE(ans) = F_IM(r);
43     F_IM(ans) = -F_RE(r);
44     return (ans);
45 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/casinh1.c

1

1262 Tue Nov 25 12:58:09 2014

new/usr/src/lib/libm/common/complex/casinh1.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __casinh1 = casinh1
30 #pragma weak casinh1 = __casinh1
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 casinh1(ldcomplex z) {
37     ldcomplex w, r, ans;
38
39     LD_RE(w) = -LD_IM(z);
40     LD_IM(w) = LD_RE(z);
41     r = casinl(w);
42     LD_RE(ans) = LD_IM(r);
43     LD_IM(ans) = -LD_RE(r);
44     return (ans);
45 }
```

unchanged_portion_omitted

```

*****
6438 Tue Nov 25 12:58:10 2014
new/usr/src/lib/libm/common/complex/casinl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __casinl = casinl
30 #pragma weak casinl = __casinl

32 #include "libm.h"          /* asinl/atanl/fabsl/isinfl/loglpl/logl/sqrtl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"

36 /* INDENT OFF */
37 static const long double
38 zero = 0.0L,
39 one = 1.0L,
40 Acrossover = 1.5L,
41 Bcrossover = 0.6417L,
42 half = 0.5L,
43 ln2 = 6.931471805599453094172321214581765680755e-0001L,
44 Foursqrtu = 7.3344154702193886624856495681939326638255e-2466L, /* 2**-8189 */
45 #if defined(__x86)
46 E = 5.4210108624275221700372640043497085571289e-20L, /* 2**-64 */
47 pi_4 = 0.7853981633974483096156608458198756993697670L,
48 pi_4_1 = 4.1668714592604391641479322342670193036704898e-20L,
49 pi_2 = 1.5707963267948966191479842624545426588156260L,
50 pi_2_1 = 8.3337429185208783282958644685340386073409796e-20L;
52 #else
53 E = 9.6296497219361792652798897129246365926905e-35L, /* 2**-113 */
54 pi_4 = 0.7853981633974483096156608458198756993697670L,
55 pi_4_1 = 2.1679525325309452561992610065108379921905808e-35L,
56 pi_2 = 1.5707963267948966192313216916397513987395340L,
57 pi_2_1 = 4.3359050650618905123985220130216759843811616e-35L;

```

```

59 #endif
60 /* INDENT ON */

62 #if defined(__x86)
63 static const int ip1 = 0x40400000; /* 2**65 */
64 #else
65 static const int ip1 = 0x40710000; /* 2**114 */
66 #endif

68 ldcomplex
69 casinl(ldcomplex z) {
70     long double x, y, t, R, S, A, Aml, B, y2, xml, xpl, Apx;
71     int ix, iy, hx, hy;
72     ldcomplex ans;

74     x = LD_RE(z);
75     y = LD_IM(z);
76     hx = HI_XWORD(x);
77     hy = HI_XWORD(y);
78     ix = hx & 0x7fffffff;
79     iy = hy & 0x7fffffff;
80     x = fabsl(x);
81     y = fabsl(y);

83     /* special cases */

85     /* x is inf or NaN */
86     if (ix >= 0x7fff0000) { /* x is inf or NaN */
87         if (isinfl(x)) { /* x is INF */
88             LD_IM(ans) = x;
89             if (iy >= 0x7fff0000) {
90                 if (isinfl(y))
91                     /* casin(inf + i inf) = pi/4 + i inf */
92                     LD_RE(ans) = pi_4 + pi_4_1;
93                 else /* casin(inf + i NaN) = NaN + i inf */
94                     LD_RE(ans) = y + y;
95             } else /* casin(inf + iy) = pi/2 + i inf */
96                 LD_RE(ans) = pi_2 + pi_2_1;
97         } else { /* x is NaN */
98             if (iy >= 0x7fff0000) {
99                 /* INDENT OFF */
100                /*
101                 * casin(NaN + i inf) = NaN + i inf
102                 * casin(NaN + i NaN) = NaN + i NaN
103                 */
104                /* INDENT ON */
105                LD_IM(ans) = y + y;
106                LD_RE(ans) = x + x;
107            } else {
108                /* INDENT OFF */
109                /* casin(NaN + i y) = NaN + i NaN */
110                /* INDENT ON */
111                LD_IM(ans) = LD_RE(ans) = x + y;
112            }
113        }
114        if (hx < 0)
115            LD_RE(ans) = -LD_RE(ans);
116        if (hy < 0)
117            LD_IM(ans) = -LD_IM(ans);
118        return (ans);
119    }

121     /* casin(+0 + i 0) = 0 + i 0. */
122     if (x == zero && y == zero)
123         return (z);

```



```

125     if (iy >= 0x7fff0000) { /* y is inf or NaN */
126         if (isinf(y)) { /* casin(x + i inf) = 0 + i inf */
127             LD_IM(ans) = y;
128             LD_RE(ans) = zero;
129         } else { /* casin(x + i NaN) = NaN + i NaN */
130             LD_IM(ans) = x + y;
131             if (x == zero)
132                 LD_RE(ans) = x;
133             else
134                 LD_RE(ans) = y;
135         }
136         if (hx < 0)
137             LD_RE(ans) = -LD_RE(ans);
138         if (hy < 0)
139             LD_IM(ans) = -LD_IM(ans);
140         return (ans);
141     }

142     if (y == zero) { /* region 1: y=0 */
143         if (ix < 0x3fff0000) { /* |x| < 1 */
144             LD_RE(ans) = asinl(x);
145             LD_IM(ans) = zero;
146         } else {
147             LD_RE(ans) = pi_2 + pi_2_1;
148             if (ix >= ipl) /* i386 ? 2**65 : 2**114 */
149                 LD_IM(ans) = ln2 + logl(x);
150             else if (ix >= 0x3fff8000) /* x > Acrossover */
151                 LD_IM(ans) = logl(x + sqrtl((x - one) * (x +
152                     one)));
153             else {
154                 xml = x - one;
155                 LD_IM(ans) = loglpl(xml + sqrtl(xml * (x +
156                     one)));
157             }
158         }
159     }
160     } else if (y <= E * fabs(x - one)) { /* region 2: y < tiny*|x-1| */
161         if (ix < 0x3fff0000) { /* x < 1 */
162             LD_RE(ans) = asinl(x);
163             LD_IM(ans) = y / sqrtl((one + x) * (one - x));
164         } else {
165             LD_RE(ans) = pi_2 + pi_2_1;
166             if (ix >= ipl) /* i386 ? 2**65 : 2**114 */
167                 LD_IM(ans) = ln2 + logl(x);
168             else if (ix >= 0x3fff8000) /* x > Acrossover */
169                 LD_IM(ans) = logl(x + sqrtl((x - one) * (x +
170                     one)));
171             else
172                 LD_IM(ans) = loglpl((x - one) + sqrtl((x -
173                     one) * (x + one)));
174         }
175     } else if (y < Foursqrtu) { /* region 3 */
176         t = sqrtl(y);
177         LD_RE(ans) = pi_2 - (t - pi_2_1);
178         LD_IM(ans) = t;
179     } else if (E * y - one >= x) { /* region 4 */
180         LD_RE(ans) = x / y; /* need to fix underflow cases */
181         LD_IM(ans) = ln2 + logl(y);
182     } else if (ix >= 0x5ffb0000 || iy >= 0x5ffb0000) {
183         /* region 5: x+1 and y are both (>= sqrt(max)/8) i.e. 2**8188 */
184         t = x / y;
185         LD_RE(ans) = atanl(t);
186         LD_IM(ans) = ln2 + logl(y) + half * loglpl(t * t);
187     } else if (x < Foursqrtu) {
188         /* region 6: x is very small, < 4sqrt(min) */
189         A = sqrtl(one + y * y);

```

```

190         LD_RE(ans) = x / A; /* may underflow */
191         if (iy >= 0x3fff8000) /* if y > Acrossover */
192             LD_IM(ans) = logl(y + A);
193     } else
194         LD_IM(ans) = half * loglpl((y + y) * (y + A));
195     } else { /* safe region */
196         y2 = y * y;
197         xpl = x + one;
198         xml = x - one;
199         R = sqrtl(xpl * xpl + y2);
200         S = sqrtl(xml * xml + y2);
201         A = half * (R + S);
202         B = x / A;
203         if (B <= Bcrossover)
204             LD_RE(ans) = asinl(B);
205     } else { /* use atan and an accurate approx to a-x */
206         Apx = A + x;
207         if (x <= one)
208             LD_RE(ans) = atanl(x / sqrtl(half * Apx * (y2 /
209                 (R + xpl) + (S - xml))));
210     } else
211         LD_RE(ans) = atanl(x / (y * sqrtl(half * (Apx /
212             (R + xpl) + Apx / (S + xml))));
213     }
214     } if (A <= Acrossover) {
215         /* use loglp and an accurate approx to A-1 */
216         if (x < one)
217             Aml = half * (y2 / (R + xpl) + y2 / (S - xml));
218         else
219             Aml = half * (y2 / (R + xpl) + (S + xml));
220         LD_IM(ans) = loglpl(Aml + sqrtl(Aml * (A + one)));
221     } else {
222         LD_IM(ans) = logl(A + sqrtl(A * A - one));
223     }
224     }

225     if (hx < 0)
226         LD_RE(ans) = -LD_RE(ans);
227     if (hy < 0)
228         LD_IM(ans) = -LD_IM(ans);
229     return (ans);
230 }
231 }
232 }

```

unchanged_portion_omitted

```

*****
8750 Tue Nov 25 12:58:10 2014
new/usr/src/lib/libm/common/complex/catan.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __catan = catan
30 #pragma weak catan = __catan

32 /* INDENT OFF */
33 /*
34  * dcomplex catan(dcomplex z);
35  *
36  * If
37  *     z = x + iy,
38  *
39  * then
40  *
41  * 
$$\operatorname{Re} w = \frac{1}{2} \arctan\left(\frac{2x}{1-x^2-y^2}\right) = \frac{1}{2} \operatorname{ATAN2}(2x, 1-x^2-y^2)$$

42  *
43  * 
$$\operatorname{Im} w = \frac{1}{4} \log\left(\frac{(x+(y+1))^2}{(x+(y-1))^2}\right) = \frac{1}{4} \log\left[1 + \frac{4y}{x^2+(y-1)^2}\right]$$

44  *
45  * 
$$= t - 2t^2 + \frac{16}{3}t^3 - \dots, \text{ where } t = \frac{y}{x^2+(y-1)^2}$$

46  *
47  * Note that: if catan(x, y) = (u, v), then
48  *     catan(-x, y) = (-u, v)
49  *     catan(x, -y) = (u, -v)

```

```

58 *
59 * Also, catan(x,y) = -i*catanh(-y,x), or
60 *     catanh(x,y) = i*catan(-y,x)
61 * So, if catanh(y,x) = (v,u), then catan(x,y) = -i*(-v,u) = (u,v), i.e.,
62 *     catan(x,y) = (u,v)
63 *
64 * EXCEPTION CASES (conform to ISO/IEC 9899:1999(E)):
65 *     catan(0, 0) = (0, 0)
66 *     catan(NaN, 0) = (NaN, 0)
67 *     catan(0, 1) = (0, +inf) with divide-by-zero
68 *     catan(inf, y) = (pi/2, 0) for finite y
69 *     catan(NaN, y) = (NaN, NaN) with invalid for finite y != 0
70 *     catan(x, inf) = (pi/2, 0) for finite x
71 *     catan(inf, inf) = (pi/2, 0)
72 *     catan(NaN, inf) = (NaN, 0)
73 *     catan(x, NaN) = (NaN, NaN) with invalid for finite x
74 *     catan(inf, NaN) = (pi/2, +-0)
75 */
76 /* INDENT ON */

78 #include "libm.h" /* atan/atan2/fabs/log/loglp */
79 #include "complex_wrapper.h"

81 /* INDENT OFF */
82 static const double
83     pi_2 = 1.570796326794896558e+00,
84     zero = 0.0,
85     half = 0.5,
86     two = 2.0,
87     ln2 = 6.931471805599453094172321214581765680755e-0001,
88     one = 1.0;
89 /* INDENT ON */

91 dcomplex
92 catan(dcomplex z) {
93     dcomplex ans;
94     double x, y, ax, ay, t;
95     int hx, hy, ix, iy;
96     unsigned lx, ly;

98     x = D_RE(z);
99     y = D_IM(z);
100     ax = fabs(x);
101     ay = fabs(y);
102     hx = HI_WORD(x);
103     lx = LO_WORD(x);
104     hy = HI_WORD(y);
105     ly = LO_WORD(y);
106     ix = hx & 0x7fffffff;
107     iy = hy & 0x7fffffff;

109     /* x is inf or NaN */
110     if (ix >= 0x7ff00000) {
111         if (ISINF(ix, lx)) {
112             D_RE(ans) = pi_2;
113             D_IM(ans) = zero;
114         } else {
115             D_RE(ans) = x + x;
116             if ((iy | ly) == 0 || (ISINF(iy, ly)))
117                 D_IM(ans) = zero;
118             else
119                 D_IM(ans) = (fabs(y) - ay) / (fabs(y) - ay);
120         }
121     } else if (iy >= 0x7ff00000) {
122         /* y is inf or NaN */
123         if (ISINF(iy, ly)) {

```

```

124         D_RE(ans) = pi_2;
125         D_IM(ans) = zero;
126     } else {
127         D_RE(ans) = (fabs(x) - ax) / (fabs(x) - ax);
128         D_IM(ans) = y;
129     }
130 } else if ((ix | lx) == 0) {
131     /* INDEXT OFF */
132     /*
133     * x = 0
134     * A = --- * atan2(2x, 1-x*x-y*y) = --- atan2(0,1-|y|)
135     *      2                        2
136     *
137     * B = 1/4 log [ (y+1)*(y+1) ] = 1/2 log(1+ 2/y-1) or 1/2 log(1+ 2y/1-y)
138     *
139     */
140     /* INDEXT ON */
141     t = one - ay;
142     if (((iy - 0x3ff00000) | ly) == 0) {
143         /* y=1: catan(0,1)=(0,+inf) with 1/0 signal */
144         D_IM(ans) = ay / ax;
145         D_RE(ans) = zero;
146     } else if (iy >= 0x3ff00000) { /* y>1 */
147         D_IM(ans) = half * logp(two / (-t));
148         D_RE(ans) = pi_2;
149     } else { /* y<1 */
150         D_IM(ans) = half * logp((ay + ay) / t);
151         D_RE(ans) = zero;
152     }
153 } else if (iy < 0x3e200000 || ((ix - iy) >> 20) >= 30) {
154     /* INDEXT OFF */
155     /*
156     * Tiny y (relative to 1+|x|)
157     * |y| < E*(1+|x|)
158     * where E=2**(-29, -35, -60 for double, double extended, quad precision)
159     *
160     * A = --- * atan2(2x, 1-x*x-y*y) ~ [ x<=1: atan(x) / 2 ] [ x>=1: -atan2(2,(1-x)*(---)) / x ]
161     *
162     * B ~ t*(1-2t), where t = y/x / (x + (y-1)*(y-1)/x) is tiny
163     */
164     /* INDEXT ON */
165     if (ix < 0x3ff00000)
166         D_RE(ans) = atan(ax);
167     else
168         D_RE(ans) = half * atan2(two, (one - ax) * (one + one / ax));
169     if ((iy | ly) == 0) {
170         D_IM(ans) = ay;
171     } else {
172         if (ix < 0x3e200000)
173             t = ay / ((ay - one) * (ay - one));
174         else if (ix > 0x41c00000)
175             t = (ay / ax) / ax;
176         else
177             t = ay / (ax * ax + (ay - one) * (ay - one));
178         D_IM(ans) = t * (one - (t + t));
179     }
180 } else if (iy >= 0x41c00000 && ((iy - ix) >> 20) >= 30) {
181     /* INDEXT OFF */

```

```

190     /*
191     * Huge y relative to 1+|x|
192     * |y| > Einv*(1+|x|), where Einv=2**(prec/2+3),
193     *
194     * A ~ --- * atan2(2x, -y*y) ~ pi/2
195     *      2
196     *
197     * B ~ t*(1-2t), where t = y / ((y-1)*(y-1)) is tiny
198     */
199     /* INDEXT ON */
200     D_RE(ans) = pi_2;
201     t = (ay / (ay - one)) / (ay - one);
202     D_IM(ans) = t * (one - (t + t));
203 } else if (((iy - 0x3ff00000) | ly) == 0) {
204     /* INDEXT OFF */
205     /*
206     * y = 1
207     * A = --- * atan2(2x, -x*x) = --- atan2(2,-x)
208     *      2                        2
209     *
210     * B = -log [ (x*x + 4) ] = -log(1+ 4/x*x) = [ 0.5(log2-logx) if |x|<E, else 0.25*logp((2/x)*(2/x))
211     *      4 [ x*x ] 4 x*x [ logp((2/x)*(2/x)) ]
212     */
213     /* INDEXT ON */
214     D_RE(ans) = half * atan2(two, -ax);
215     if (ix < 0x3e200000)
216         D_IM(ans) = half * (ln2 - log(ax));
217     else {
218         t = two / ax;
219         D_IM(ans) = 0.25 * logp(t * t);
220     }
221 } else if (ix >= 0x43900000) {
222     /* INDEXT OFF */
223     /*
224     * Huge x:
225     * when |x| > 1/E^2,
226     * A ~ --- * atan2(2x, -x*x-y*y) ~ --- pi
227     *      2                        2
228     *
229     * B ~ t*(1-2t), where t = y / (x*x+(y-1)*(y-1) + ((y-1)/x)^2) = (y/x) / (1+((y-1)/x)^2)
230     */
231     /* INDEXT ON */
232     D_RE(ans) = pi_2;
233     t = ((ay / ax) / (one + ((ay - one) / ax) * ((ay - one) / ax))) / ax;
234     D_IM(ans) = t * (one - (t + t));
235 } else if (ix < 0x38b00000) {
236     /* INDEXT OFF */
237     /*
238     * Tiny x:
239     * when |x| < E^4, (note that y != 1)
240     * A = --- * atan2(2x, 1-x*x-y*y) ~ --- * atan2(2x,(1-y)*(1+y))
241     *      2                        2
242     *
243     * B = 1/4 log [ (y+1)*(y+1) ] = 1/2 log(1+ 2/y-1) or 1/2 log(1+ 2y/1-y)
244     *
245     */
246     /* INDEXT ON */
247     D_RE(ans) = half * atan2(ax + ax, (one - ay) * (one + ay));

```

```

256         if (iy >= 0x3ff00000)
257             D_IM(ans) = half * loglp(two / (ay - one));
258         else
259             D_IM(ans) = half * loglp((ay + ay) / (one - ay));
260     } else {
261         /* INDENT OFF */
262         /*
263          * normal x,y
264          *
265          * A = --- * atan2(2x, 1-x*x-y*y)
266          *      2
267          *
268          * B = - log [x*x+(y+1)*(y+1)] / 4 = - log (1+ 4y / [x*x+(y-1)*(y-1)])
269          *
270          *      4 [x*x+(y-1)*(y-1)] / 4
271          */
272         /* INDENT ON */
273         t = one - ay;
274         if (iy >= 0x3fe00000 && iy < 0x40000000) {
275             /* y close to 1 */
276             D_RE(ans) = half * (atan2((ax + ax), (t * (one + ay) -
277                                     ax * ax)));
278         } else if (ix >= 0x3fe00000 && ix < 0x40000000) {
279             /* x close to 1 */
280             D_RE(ans) = half * atan2((ax + ax), ((one - ax) *
281                                     (one + ax) - ay * ay));
282         } else
283             D_RE(ans) = half * atan2((ax + ax), ((one - ax * ax) -
284                                     ay * ay));
285         D_IM(ans) = 0.25 * loglp((4.0 * ay) / (ax * ax + t * t));
286     }
287     if (hx < 0)
288         D_RE(ans) = -D_RE(ans);
289     if (hy < 0)
290         D_IM(ans) = -D_IM(ans);
291     return (ans);
292 }

```

unchanged_portion_omitted

```

*****
3576 Tue Nov 25 12:58:10 2014
new/usr/src/lib/libm/common/complex/catanf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28 #pragma weak __catanf = catanf
29 #pragma weak catanf = __catanf
30
31 #include "libm.h"
32 #include "complex_wrapper.h"
33
34 #if defined(__i386) && !defined(__amd64)
35 extern int __swapRP(int);
36 #endif
37
38 static const float
39     pi_2 = 1.570796326794896558e+00F,
40     zero = 0.0F,
41     half = 0.5F,
42     two = 2.0F,
43     one = 1.0F;
44
45 fcomplex
46 catanf(fcomplex z) {
47     fcomplex    ans;
48     float       x, y, ax, ay, t;
49     double      dx, dy, dt;
50     int         hx, hy, ix, iy;
51
52     x = F_RE(z);
53     y = F_IM(z);
54     ax = fabsf(x);
55     ay = fabsf(y);
56     hx = THE_WORD(x);
57     hy = THE_WORD(y);

```

```

58     ix = hx & 0x7fffffff;
59     iy = hy & 0x7fffffff;
60
61     if (ix >= 0x7f800000) { /* x is inf or NaN */
62         if (ix == 0x7f800000) {
63             F_RE(ans) = pi_2;
64             F_IM(ans) = zero;
65         } else {
66             F_RE(ans) = x * x;
67             if (iy == 0 || iy == 0x7f800000)
68                 F_IM(ans) = zero;
69             else
70                 F_IM(ans) = (fabsf(y) - ay) / (fabsf(y) - ay);
71         }
72     } else if (iy >= 0x7f800000) { /* y is inf or NaN */
73         if (iy == 0x7f800000) {
74             F_RE(ans) = pi_2;
75             F_IM(ans) = zero;
76         } else {
77             F_RE(ans) = (fabsf(x) - ax) / (fabsf(x) - ax);
78             F_IM(ans) = y * y;
79         }
80     } else if (ix == 0) {
81         /* INDEXT OFF */
82         /*
83          * x = 0
84          *
85          * A = --- * atan2(2x, 1-x*x-y*y) = --- atan2(0,1-|y|)
86          *     2
87          *
88          * B = - log [ (y+1)*(y+1) ] = - log (1+ ---) or - log(1+ ---)
89          *     4 [ (y-1)*(y-1) ] 2          y-1 2          1-y
90          */
91         /* INDEXT ON */
92         t = one - ay;
93         if (iy == 0x3f800000) {
94             /* y=1: catan(0,1)=(0,+inf) with 1/0 signal */
95             F_IM(ans) = ay / ax;
96             F_RE(ans) = zero;
97         } else if (iy > 0x3f800000) { /* y>1 */
98             F_IM(ans) = half * log1pf(two / (-t));
99             F_RE(ans) = pi_2;
100        } else { /* y<1 */
101            F_IM(ans) = half * log1pf((ay + ay) / t);
102            F_RE(ans) = zero;
103        }
104    } else {
105        /* INDEXT OFF */
106        /*
107         * use double precision x,y
108         *
109         * A = --- * atan2(2x, 1-x*x-y*y)
110         *     2
111         *
112         * B = 1 [ x*x+(y+1)*(y+1) ] = 1 [ x*x+(y-1)*(y-1) ]
113         *     4 log [ ----- ] = - log (1+ -----)
114         *     4 [ x*x+(y-1)*(y-1) ] 4          x*x + (y-1)*(y-1)
115         */
116        /* INDEXT ON */
117        #if defined(__i386) && !defined(__amd64)
118            int rp = __swapRP(fp_extended);
119        #endif
120        #endif
121        dx = (double)ax;
122        dy = (double)ay;
123        F_RE(ans) = (float)(0.5 * atan2(dx + dx,

```

```
124         1.0 - dx * dx - dy * dy));
125         dt = dy - 1.0;
126         F_IM(ans) = (float)(0.25 * loglp(4.0 * dy /
127             (dx * dx + dt * dt)));
128 #if defined(__i386) && !defined(__amd64)
129         if (rp != fp_extended)
130             (void) __swapRP(rp);
131 #endif
132     }
133     if (hx < 0)
134         F_RE(ans) = -F_RE(ans);
135     if (hy < 0)
136         F_IM(ans) = -F_IM(ans);
137     return (ans);
138 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/complex/catanh.c

1

1428 Tue Nov 25 12:58:11 2014

new/usr/src/lib/libm/common/complex/catanh.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __catanh = catanh
30 #pragma weak catanh = __catanh
```

```
32 /* INDENT OFF */
33 /*
34  * z := x + iy
35  * catanh(z) = -i catan(iz)
36  *           = -i catan(-y+ix)
37  *           = (Im(catan(-y+ix)), -Re(catan(-y+ix)))
38 */
39 /* INDENT ON */
```

```
41 #include "libm.h"
42 #include "complex_wrapper.h"
```

```
44 dcomplex
45 catanh(dcomplex z) {
46     double x, y;
47     dcomplex ans, ct;
48
49     x = D_RE(z);
50     y = D_IM(z);
51     D_RE(z) = -y;
52     D_IM(z) = x;
53     ct = catan(z);
54     D_RE(ans) = D_IM(ct);
55     D_IM(ans) = -D_RE(ct);
56     return (ans);
57 }
```

new/usr/src/lib/libm/common/complex/catanhf.c

1

1281 Tue Nov 25 12:58:11 2014

new/usr/src/lib/libm/common/complex/catanhf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __catanhf = catanhf
30 #pragma weak catanhf = __catanhf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 catanhf(fcomplex z) {
37     float x, y;
38     fcomplex ans, ct;
39
40     x = F_RE(z);
41     y = F_IM(z);
42     F_RE(z) = -y;
43     F_IM(z) = x;
44     ct = catanf(z);
45     F_RE(ans) = F_IM(ct);
46     F_IM(ans) = -F_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/complex/catanhl.c

1

1298 Tue Nov 25 12:58:12 2014

new/usr/src/lib/libm/common/complex/catanhl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __catanhl = catanhl
30 #pragma weak catanhl = __catanhl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 catanhl(ldcomplex z) {
37     long double x, y;
38     ldcomplex ans, ct;
39
40     x = LD_RE(z);
41     y = LD_IM(z);
42     LD_RE(z) = -y;
43     LD_IM(z) = x;
44     ct = catanl(z);
45     LD_RE(ans) = LD_IM(ct);
46     LD_IM(ans) = -LD_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

```

*****
10206 Tue Nov 25 12:58:12 2014
new/usr/src/lib/libm/common/complex/catanl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __catanl = catanl
30 #pragma weak catanl = __catanl

32 /* INDENT OFF */
33 /*
34  * ldcomplex catanl(ldcomplex z);
35  *
36  * Atan(z) return A + Bi where,
37  *
38  *      1
39  *      A = --- * atan2(2x, 1-x*x-y*y)
40  *      2
41  *
42  *      1          [ x*x + (y+1)*(y+1) ]   1          4y
43  *      B = --- log [ ----- ] = - log (1+ -----)
44  *      4          [ x*x + (y-1)*(y-1) ]   4          x*x + (y-1)*(y-1)
45  *
46  *      = t - 2t + --- t - ..., where t = -----
47  *      3          3          x*x + (y-1)*(y-1)
48  * Proof:
49  * Let w = atan(z=x+yi) = A + B i. Then tan(w) = z.
50  * Since sin(w) = (exp(iw)-exp(-iw))/(2i), cos(w)=(exp(iw)+exp(-iw))/(2),
51  * Let p = exp(iw), then z = tan(w) = ((p-1/p)/(p+1/p))/i, or
52  * iz = (p*p-1)/(p*p+1), or, after simplification,
53  *      p*p = (1+iz)/(1-iz) ... (1)
54  * LHS of (1) = exp(2iw) = exp(2i(A+Bi)) = exp(-2B)*exp(2iA)
55  *              = exp(-2B)*(cos(2A)+i*sin(2A)) ... (2)
56  *              1-y+ix (1-y+ix)*(1+y+ix) 1-x*x-y*y + 2xi
57  * RHS of (1) = ----- = ----- = ----- ... (3)

```

```

58  *      1+y-ix (1+y)**2 + x**2 (1+y)**2 + x**2
59  *
60  * Comparing the real and imaginary parts of (2) and (3), we have:
61  *      cos(2A) = 1-x*x-y*y = sin(2A) : 2x
62  * and hence
63  *      tan(2A) = 2x/(1-x*x-y*y), or
64  *      A = 0.5 * atan2(2x, 1-x*x-y*y) ... (4)
65  *
66  * For the imaginary part B, Note that |p*p| = exp(-2B), and
67  *      |1+iz| |i-z| hypot(x,(y-1))
68  *      |----| = |----| = -----
69  *      |1-iz| |i+z| hypot(x,(y+1))
70  * Thus
71  *      x*x + (y+1)*(y+1)
72  *      exp(4B) = -----, or
73  *      x*x + (y-1)*(y-1)
74  *
75  *      1 [x^2+(y+1)^2] 1 4y
76  *      B = - log [-----] = - log(1+ -----) ... (5)
77  *      4 [x^2+(y-1)^2] 4 x^2+(y-1)^2
78  *
79  * QED.
80  *
81  * Note that: if catan(x, y) = (u, v), then
82  *      catan(-x, y) = (-u, v)
83  *      catan(x, -y) = (u, -v)
84  *
85  * Also, catan(x,y) = -i*catanh(-y,x), or
86  *      catanh(x,y) = i*catan(-y,x)
87  * So, if catanh(y,x) = (v,u), then catan(x,y) = -i*(-v,u) = (u,v), i.e.,
88  *      catan(x,y) = (u,v)
89  *
90  * EXCEPTION CASES (conform to ISO/IEC 9899:1999(E)):
91  *      catan( 0 , 0 ) = ( 0 , 0 )
92  *      catan( NaN, 0 ) = (NaN , 0 )
93  *      catan( 0 , 1 ) = ( 0 , +inf) with divide-by-zero
94  *      catan( inf, y ) = (pi/2 , 0 ) for finite +y
95  *      catan( NaN, y ) = (NaN , NaN) with invalid for finite y != 0
96  *      catan( x , inf ) = (pi/2 , 0 ) for finite +x
97  *      catan( inf, inf ) = (pi/2 , 0 )
98  *      catan( NaN, inf ) = (NaN , 0 )
99  *      catan( x , NaN ) = (NaN , NaN) with invalid for finite x
100 *      catan( inf, NaN ) = (pi/2 , +-0)
101 */
102 /* INDENT ON */

104 #include "libm.h" /* atan2l/atanl/fabsl/isinfl/iszerol/loglpl/logl */
105 #include "complex_wrapper.h"
106 #include "longdouble.h"

108 /* INDENT OFF */
109 static const long double
110 zero = 0.0L,
111 one = 1.0L,
112 two = 2.0L,
113 half = 0.5L,
114 ln2 = 6.931471805599453094172321214581765680755e-0001L,
115 pi_2 = 1.570796326794896619231321691639751442098584699687552910487472L,
116 #if defined(__x86)
117 E = 2.910383045673370361328125000000000000000e-11L, /* 2**-35 */
118 Einv = 3.435973836800000000000000000000000000000e+10L; /* 2**+35 */
119 #else
120 E = 8.673617379884035472059622406959533691406e-19L, /* 2**-60 */
121 Einv = 1.152921504606846976000000000000000000000e18L; /* 2**+60 */
122 #endif
123 /* INDENT ON */

```

```

125 ldcomplex
126 catanl(ldcomplex z) {
127     ldcomplex ans;
128     long double x, y, t1, ax, ay, t;
129     int hx, hy, ix, iy;

131     x = LD_RE(z);
132     y = LD_IM(z);
133     ax = fabsl(x);
134     ay = fabsl(y);
135     hx = HI_XWORD(x);
136     hy = HI_XWORD(y);
137     ix = hx & 0x7fffffff;
138     iy = hy & 0x7fffffff;

140     /* x is inf or NaN */
141     if (ix >= 0x7fff0000) {
142         if (isinfl(x)) {
143             LD_RE(ans) = pi_2;
144             LD_IM(ans) = zero;
145         } else {
146             LD_RE(ans) = x + x;
147             if (iszerol(y) || (isinfl(y)))
148                 LD_IM(ans) = zero;
149             else
150                 LD_IM(ans) = (fabsl(y) - ay) / (fabsl(y) - ay);
151         }
152     } else if (iy >= 0x7fff0000) {
153         /* y is inf or NaN */
154         if (isinfl(y)) {
155             LD_RE(ans) = pi_2;
156             LD_IM(ans) = zero;
157         } else {
158             LD_RE(ans) = (fabsl(x) - ax) / (fabsl(x) - ax);
159             LD_IM(ans) = y;
160         }
161     } else if (iszerol(x)) {
162         /* INDENT OFF */
163         /*
164         * x = 0
165         * A = --- * atan2(2x, 1-x*x-y*y) = --- atan2(0,1-|y|)
166         *     2                       2
167         *
168         * B = - log [ (y+1)*(y+1) ] 1 2 1 2y
169         *     4 [ (y-1)*(y-1) ] 2 2 y-1 2 log(1+ ---) or - log(1+ ---)
170         *
171         */
172         /* INDENT ON */
173         t = one - ay;
174         if (ay == one) {
175             /* y=1: catan(0,1)=(0,+inf) with 1/0 signal */
176             LD_IM(ans) = ay / ax;
177             LD_RE(ans) = zero;
178         } else if (ay > one) { /* y>1 */
179             LD_IM(ans) = half * loglpl(two / (-t));
180             LD_RE(ans) = pi_2;
181         } else { /* y<1 */
182             LD_IM(ans) = half * loglpl((ay + ay) / t);
183             LD_RE(ans) = zero;
184         }
185     } else if (ay < E * (one + ax)) {
186         /* INDENT OFF */
187         /*
188         * Tiny y (relative to 1+|x|)

```

```

190     * |y| < E*(1+|x|)
191     * where E=2**(-29, -35, -60 for double, extended, quad precision)
192     *
193     * 1 [x<=1: atan(x)
194     * A = - * atan2(2x,1-x*x-y*y) ~ [ 1 1+x
195     * 2 [x>=1: - atan2(2,(1-x)*(-----))
196     * 2 x
197     *
198     * y/x
199     * B ~ t*(1-2t), where t = ----- is tiny
200     * x + (y-1)*(y-1)/x
201     *
202     * Y
203     * (when x < 2**(-60, t = -----)
204     * (y-1)*(y-1)
205     */
206     /* INDENT ON */
207     if (ay == zero)
208         LD_IM(ans) = ay;
209     else {
210         t1 = ay - one;
211         if (ix < 0x3fc30000)
212             t = ay / (t1 * t1);
213         else if (ix > 0x403b0000)
214             t = (ay / ax) / ax;
215         else
216             t = ay / (ax * ax + t1 * t1);
217         LD_IM(ans) = t * (one - two * t);
218     }
219     if (ix < 0x3fff0000)
220         LD_RE(ans) = atanl(ax);
221     else
222         LD_RE(ans) = half * atan2l(two, (one - ax) * (one +
223         one / ax));

225     } else if (ay > Einv * (one + ax)) {
226         /* INDENT OFF */
227         /*
228         * Huge y relative to 1+|x|
229         * |y| > Einv*(1+|x|), where Einv=2**(prec/2+3),
230         *
231         * A ~ --- * atan2(2x, -y*y) ~ pi/2
232         * 2
233         *
234         * B ~ t*(1-2t), where t = ----- is tiny
235         * (y-1)*(y-1)
236         */
237         /* INDENT ON */
238         LD_RE(ans) = pi_2;
239         t = (ay / (ay - one)) / (ay - one);
240         LD_IM(ans) = t * (one - (t + t));
241     } else if (ay == one) {
242         /* INDENT OFF */
243         /*
244         * y=1
245         * 1 1
246         * A = - * atan2(2x, -x*x) = --- atan2(2,-x)
247         * 2 2
248         *
249         * 1 [ x*x+4 ] 1 4 [ 0.5(log2-logx) if
250         * B = - log [ ----- ] = - log (1+ ---) = [ |x|<E, else 0.25*
251         * 4 [ x*x ] 4 x*x [ loglp((2/x)*(2/x))
252         *
253         */
254         /* INDENT ON */
255         LD_RE(ans) = half * atan2l(two, -ax);
256         if (ax < E)

```

```

256         LD_IM(ans) = half * (ln2 - logl(ax));
257     else {
258         t = two / ax;
259         LD_IM(ans) = 0.25L * loglpl(t * t);
260     }
261 } else if (ax > Einv * Einv) {
262     /* INDENT OFF */
263     /*
264     * Huge x:
265     * when |x| > 1/E^2,
266     *  $A \sim \frac{1}{2} \operatorname{atan2}(2x, -x^2 - y^2) \sim \frac{\pi}{2}$ 
267     *  $B \sim t(1-2t)$ , where  $t = \frac{y}{x^2x+(y-1)(y-1)} = \frac{y/x}{1+((y-1)/x)^2}$ 
268     */
269     /* INDENT ON */
270     LD_RE(ans) = pi_2;
271     t = ((ay / ax) / (one + ((ay - one) / ax) * ((ay - one) /
272         ax))) / ax;
273     LD_IM(ans) = t * (one - (t + t));
274 } else if (ax < E * E * E * E * E) {
275     /* INDENT OFF */
276     /*
277     * Tiny x:
278     * when |x| < E^4, (note that y != 1)
279     *  $A = \frac{1}{2} \operatorname{atan2}(2x, 1 - x^2 - y^2) \sim \frac{1}{2} \operatorname{atan2}(2x, 1 - y^2)$ 
280     *  $B = -\frac{1}{4} \log \left[ \frac{(y+1)(y+1)}{(y-1)(y-1)} \right] = -\frac{1}{2} \log \left( 1 + \frac{2y}{y-1} \right)$  or  $-\frac{1}{2} \log \left( 1 + \frac{2y}{1-y} \right)$ 
281     */
282     /* INDENT ON */
283     LD_RE(ans) = half * atan2l(ax + ax, (one - ay) * (one + ay));
284     if (ay > one) /* y>1 */
285         LD_IM(ans) = half * loglpl(two / (ay - one));
286     else /* y<1 */
287         LD_IM(ans) = half * loglpl((ay + ay) / (one - ay));
288 } else {
289     /* INDENT OFF */
290     /*
291     * normal x,y
292     *  $A = \frac{1}{2} \operatorname{atan2}(2x, 1 - x^2 - y^2)$ 
293     *  $B = -\frac{1}{4} \log \left[ \frac{x^2x+(y+1)(y+1)}{x^2x+(y-1)(y-1)} \right] = -\frac{1}{4} \log \left( 1 + \frac{4y}{x^2x+(y-1)(y-1)} \right)$ 
294     */
295     /* INDENT ON */
296     t = one - ay;
297     if (iy >= 0x3ffe0000 && iy < 0x40000000) {
298         /* y close to 1 */
299         LD_RE(ans) = half * (atan2l((ax + ax), (t * (one +
300             ay) - ax * ax)));
301     } else if (ix >= 0x3ffe0000 && ix < 0x40000000) {
302         /* x close to 1 */
303         LD_RE(ans) = half * atan2l((ax + ax), ((one - ax) *
304             (one + ax) - ay * ay));
305     } else
306         LD_RE(ans) = half * atan2l((ax + ax), ((one - ax *
307             ax) - ay * ay));
308 }

```

```

322         LD_IM(ans) = 0.25L * loglpl((4.0L * ay) / (ax * ax + t * t));
323     }
324     if (hx < 0)
325         LD_RE(ans) = -LD_RE(ans);
326     if (hy < 0)
327         LD_IM(ans) = -LD_IM(ans);
328     return (ans);
329 }

```

unchanged_portion_omitted_

new/usr/src/lib/libm/common/complex/ccos.c

1

```
*****
1420 Tue Nov 25 12:58:13 2014
new/usr/src/lib/libm/common/complex/ccos.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ccos = ccos
30 #pragma weak ccos = __ccos
```

```
32 /* INDENT OFF */
33 /*
34  * dcomplex ccos(dcomplex z);
35  *
36  * z := x+iy; since ccos(iz) = cosh(z), we have
37  * ccos(z)      = ccos((-1)*(-z)) = ccosh(i*i*(-z))
38  *              = ccosh(i*(-z)) = ccosh(i*(-x-yi))
39  *              = ccosh(y-ix)
40  */
41 /* INDENT ON */
```

```
43 #include "libm.h"
44 #include "complex_wrapper.h"
```

```
46 dcomplex
47 ccos(dcomplex z) {
48     double x, y;
49
50     x = D_RE(z);
51     y = D_IM(z);
52     D_RE(z) = y;
53     D_IM(z) = -x;
54     return (ccosh(z));
55 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/ccosf.c

1

1198 Tue Nov 25 12:58:13 2014

new/usr/src/lib/libm/common/complex/ccosf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ccosf = ccosf
30 #pragma weak ccosf = __ccosf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 ccosf(fcomplex z) {
37     float x, y;
38
39     x = F_RE(z);
40     y = F_IM(z);
41     F_RE(z) = y;
42     F_IM(z) = -x;
43     return (ccoshf(z));
44 }
unchanged_portion_omitted
```

```

*****
3894 Tue Nov 25 12:58:13 2014
new/usr/src/lib/libm/common/complex/ccosh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __ccosh = ccosh
30 #pragma weak ccosh = __ccosh

32 /* INDENT OFF */
33 /*
34  * dcomplex ccosh(dcomplex z);
35  *
36  *          z      -z      x      -x
37  *          e  +  e  +  e  (cos(y)+i*sin(y)) + e  (cos(-y)+i*sin(-y))
38  * cosh z = ----- = -----
39  *          2          2
40  *          x      -x      x      -x
41  *          cos(y) ( e  + e  ) + i*sin(y) ( e  - e  )
42  *          = -----
43  *          2
44  *          = cos(y) cosh(x) + i sin(y) sinh(x)
45  *
46  * Implementation Note
47  * -----
48  *
49  *
50  *          |x|      -|x|      |x|      -2|x|      -2|x|      -P-4
51  * Note that e  + e  = e  ( 1 + e  ). If e  < 2  , where
52  *
53  * P stands for the number of significant bits of the machine precision,
54  * then the result will be rounded to e|x|. Therefore, we have
55  *
56  *
57  *          z

```

```

58  *          e
59  *          cosh z = ----- if |x| >= (P/2 + 2)*ln2
60  *          2
61  *
62  * EXCEPTION (conform to ISO/IEC 9899:1999(E)):
63  *          ccosh(0,0)=(1,0)
64  *          ccosh(0,inf)=(NaN,+0)
65  *          ccosh(0,NaN)=(NaN,+0)
66  *          ccosh(x,inf) = (NaN,NaN) for finite non-zero x
67  *          ccosh(x,NaN) = (NaN,NaN) for finite non-zero x
68  *          ccosh(inf,0) = (inf, 0)
69  *          ccosh(inf,y) = (inf*cos(y),inf*sin(y)) for finite non-zero y
70  *          ccosh(inf,inf) = (+-inf,NaN)
71  *          ccosh(inf,NaN) = (+inf,NaN)
72  *          ccosh(NaN,0) = (NaN,+0)
73  *          ccosh(NaN,y) = (NaN,NaN) for non-zero y
74  *          ccosh(NaN,NaN) = (NaN,NaN)
75  */
76 /* INDENT ON */

78 #include "libm.h"          /* cosh/exp/fabs/scalbn/sinh/sincos/__k_cexp */
79 #include "complex_wrapper.h"

81 dcomplex
82 ccosh(dcomplex z) {
83     double t, x, y, S, C;
84     int hx, ix, lx, hy, iy, ly, n;
85     dcomplex ans;

87     x = D_RE(z);
88     y = D_IM(z);
89     hx = HI_WORD(x);
90     lx = LO_WORD(x);
91     ix = hx & 0x7fffffff;
92     hy = HI_WORD(y);
93     ly = LO_WORD(y);
94     iy = hy & 0x7fffffff;
95     x = fabs(x);
96     y = fabs(y);

98     (void) sincos(y, &S, &C);
99     if (ix >= 0x403c0000) { /* |x| > 28 = prec/2 (14,28,34,60) */
100         if (ix >= 0x40862E42) { /* |x| > 709.78... ~ log(2**1024) */
101             if (ix >= 0x7ff00000) { /* |x| is inf or NaN */
102                 if ((iy | ly) == 0) {
103                     D_RE(ans) = x;
104                     D_IM(ans) = y;
105                 } else if (iy >= 0x7ff00000) {
106                     D_RE(ans) = x;
107                     D_IM(ans) = x - y;
108                 } else {
109                     D_RE(ans) = C * x;
110                     D_IM(ans) = S * x;
111                 }
112             } else {
113                 t = __k_cexp(x, &n);
114                 /* return exp(x)=t*2**n */
115                 D_RE(ans) = scalbn(C * t, n - 1);
116                 D_IM(ans) = scalbn(S * t, n - 1);
117             }
118         } else {
119             t = exp(x) * 0.5;
120             D_RE(ans) = C * t;
121             D_IM(ans) = S * t;
122         }
123     } else {

```

new/usr/src/lib/libm/common/complex/ccosh.c

3

```
124         if ((ix | lx) == 0) { /* x = 0, return (C,0) */
125             D_RE(ans) = C;
126             D_IM(ans) = 0.0;
127         } else {
128             D_RE(ans) = C * cosh(x);
129             D_IM(ans) = S * sinh(x);
130         }
131     }
132     if ((hx ^ hy) < 0)
133         D_IM(ans) = -D_IM(ans);
134     return (ans);
135 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/ccoshf.c

1

```
*****
2523 Tue Nov 25 12:58:15 2014
new/usr/src/lib/libm/common/complex/ccoshf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __ccoshf = ccoshf
29 #pragma weak ccoshf = __ccoshf
30
31 #include "libm.h"
32 #include "complex_wrapper.h"
33
34 #if defined(__i386) && !defined(__amd64)
35 extern int __swapRP(int);
36 #endif
37
38 static const float zero = 0.0F, half = 0.5F;
39
40 fcomplex
41 ccoshf(fcomplex z) {
42     float      t, x, y, S, C;
43     double     w;
44     int        hx, ix, hy, iy, n;
45     fcomplex   ans;
46
47     x = F_RE(z);
48     y = F_IM(z);
49     hx = THE_WORD(x);
50     ix = hx & 0x7fffffff;
51     hy = THE_WORD(y);
52     iy = hy & 0x7fffffff;
53     x = fabsf(x);
54     y = fabsf(y);
55
56     sincosf(y, &S, &C);
57     if (ix >= 0x41600000) { /* |x| > 14 = prec/2 (14,28,34,60) */
```

new/usr/src/lib/libm/common/complex/ccoshf.c

2

```
58     if (ix >= 0x42B171AA) { /* |x| > 88.722... ~ log(2**128) */
59         if (ix >= 0x7f800000) { /* |x| is inf or NaN */
60             if (iy == 0) {
61                 F_RE(ans) = x;
62                 F_IM(ans) = y;
63             } else if (iy >= 0x7f800000) {
64                 F_RE(ans) = x;
65                 F_IM(ans) = x - y;
66             } else {
67                 F_RE(ans) = C * x;
68                 F_IM(ans) = S * x;
69             }
70         } else {
71 #if defined(__i386) && !defined(__amd64)
72             int    rp = __swapRP(fp_extended);
73 #endif
74             /* return (C, S) * exp(x) / 2 */
75             w = __k_cexp((double)x, &n);
76             F_RE(ans) = (float)scalbn(C * w, n - 1);
77             F_IM(ans) = (float)scalbn(S * w, n - 1);
78 #if defined(__i386) && !defined(__amd64)
79             if (rp != fp_extended)
80                 (void) __swapRP(rp);
81 #endif
82         } else {
83             t = expf(x) * half;
84             F_RE(ans) = C * t;
85             F_IM(ans) = S * t;
86         }
87     } else {
88         if (ix == 0) { /* x = 0, return (C,0) */
89             F_RE(ans) = C;
90             F_IM(ans) = zero;
91         } else {
92             F_RE(ans) = C * coshf(x);
93             F_IM(ans) = S * sinh(x);
94         }
95     }
96     if ((hx ^ hy) < 0)
97         F_IM(ans) = -F_IM(ans);
98     return (ans);
99 }
100 }
_____unchanged_portion_omitted_____
```

```

*****
2363 Tue Nov 25 12:58:15 2014
new/usr/src/lib/libm/common/complex/ccoshl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __ccoshl = ccoshl
30 #pragma weak ccoshl = __ccoshl

32 #include "libm.h" /* coshl/expl/fabsl/scalbnl/sincosl/sinh1l/_k_cexpl */
33 #include "complex_wrapper.h"

35 /* INDENT OFF */
36 static const long double zero = 0.0L, half = 0.5L;
37 /* INDENT ON */

39 ldcomplex
40 ccoshl(ldcomplex z) {
41     long double t, x, y, S, C;
42     int hx, ix, hy, iy, n;
43     ldcomplex ans;

44     x = LD_RE(z);
45     y = LD_IM(z);
46     hx = HI_XWORD(x);
47     ix = hx & 0x7fffffff;
48     hy = HI_XWORD(y);
49     iy = hy & 0x7fffffff;
50     x = fabsl(x);
51     y = fabsl(y);

52     (void) sincosl(y, &S, &C);
53     if (ix >= 0x4004e000) { /* |x| > 60 = prec/2 (14,28,34,60) */
54         if (ix >= 0x400C62E4) { /* |x| > 11356.52... ~ log(2**16384) */
55             if (ix >= 0x7fff0000) { /* |x| is inf or NaN */

```

```

58         if (y == zero) {
59             LD_RE(ans) = x;
60             LD_IM(ans) = y;
61         } else if (iy >= 0x7fff0000) {
62             LD_RE(ans) = x;
63             LD_IM(ans) = x - y;
64         } else {
65             LD_RE(ans) = C * x;
66             LD_IM(ans) = S * x;
67         }
68     } else {
69         t = __k_cexpl(x, &n);
70         /* return exp(x)=t*2**n */
71         LD_RE(ans) = scalbnl(C * t, n - 1);
72         LD_IM(ans) = scalbnl(S * t, n - 1);
73     }
74 } else {
75     t = expl(x) * half;
76     LD_RE(ans) = C * t;
77     LD_IM(ans) = S * t;
78 }
79 } else {
80     if (x == zero) { /* x = 0, return (C,0) */
81         LD_RE(ans) = C;
82         LD_IM(ans) = zero;
83     } else {
84         LD_RE(ans) = C * coshl(x);
85         LD_IM(ans) = S * sinh1(x);
86     }
87 }
88 if ((hx ^ hy) < 0)
89     LD_IM(ans) = -LD_IM(ans);
90 return (ans);
91 }
_____ unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/complex/ccosl.c

1

1210 Tue Nov 25 12:58:15 2014

new/usr/src/lib/libm/common/complex/ccosl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ccosl = ccosl
30 #pragma weak ccosl = __ccosl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 ccosl(ldcomplex z) {
37     long double x, y;

39     x = LD_RE(z);
40     y = LD_IM(z);
41     LD_RE(z) = y;
42     LD_IM(z) = -x;
43     return (ccoshl(z));
44 }
```

unchanged_portion_omitted

```

*****
3080 Tue Nov 25 12:58:16 2014
new/usr/src/lib/libm/common/complex/cexp.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cexp = cexp
30 #pragma weak cexp = __cexp

32 /* INDENT OFF */
33 /*
34  * dcomplex cexp(dcomplex z);
35  *
36  * x+iy      x
37  * e      = e (cos(y)+i*sin(y))
38  *
39  * Over/underflow issue
40  * -----
41  * exp(x) may be huge but cos(y) or sin(y) may be tiny. So we use
42  * function __k_cexp(x,&n) to return exp(x) = __k_cexp(x,&n)*2**n.
43  * Thus if exp(x+iy) = A + Bi and t = __k_cexp(x,&n), then
44  *      A = t*cos(y)*2**n,  B = t*sin(y)*2**n
45  *
46  * Purge off all exceptional arguments:
47  *      (x,0) --> (exp(x),0)      for all x, include inf and NaN
48  *      (+inf, y) --> (+inf, NaN) for inf, nan
49  *      (-inf, y) --> (+-0, +-0) for y = inf, nan
50  *      (x,+inf/NaN) --> (NaN,NaN) for finite x
51  * For all other cases, return
52  *      (x,y) --> exp(x)*cos(y)+i*exp(x)*sin(y)
53  *
54  * Algorithm for out of range x and finite y
55  * 1. compute exp(x) in factor form (t=__k_cexp(x,&n))*2**n
56  * 2. compute sincos(y,&s,&c)
57  * 3. compute t*s+i*(t*c), then scale back to 2**n and return.

```

```

58 */
59 /* INDENT ON */

61 #include "libm.h" /* exp/scalbn/sincos/__k_cexp */
62 #include "complex_wrapper.h"

64 static const double zero = 0.0;

66 dcomplex
67 cexp(dcomplex z) {
68     dcomplex ans;
69     double x, y, t, c, s;
70     int n, ix, iy, hx, hy, lx, ly;

72     x = D_RE(z);
73     y = D_IM(z);
74     hx = HI_WORD(x);
75     lx = LO_WORD(x);
76     hy = HI_WORD(y);
77     ly = LO_WORD(y);
78     ix = hx & 0x7fffffff;
79     iy = hy & 0x7fffffff;
80     if ((iy | ly) == 0) { /* y = 0 */
81         D_RE(ans) = exp(x);
82         D_IM(ans) = y;
83     } else if (ISINF(ix, lx)) { /* x is +-inf */
84         if (hx < 0) {
85             if (iy >= 0x7ff00000) {
86                 D_RE(ans) = zero;
87                 D_IM(ans) = zero;
88             } else {
89                 sincos(y, &s, &c);
90                 D_RE(ans) = zero * c;
91                 D_IM(ans) = zero * s;
92             }
93         } else {
94             if (iy >= 0x7ff00000) {
95                 D_RE(ans) = x;
96                 D_IM(ans) = y - y;
97             } else {
98                 (void) sincos(y, &s, &c);
99                 D_RE(ans) = x * c;
100                D_IM(ans) = x * s;
101            }
102        }
103    } else {
104        (void) sincos(y, &s, &c);
105        if (ix >= 0x40862E42) { /* |x| > 709.78... ~ log(2**1024) */
106            t = __k_cexp(x, &n);
107            D_RE(ans) = scalbn(t * c, n);
108            D_IM(ans) = scalbn(t * s, n);
109        } else {
110            t = exp(x);
111            D_RE(ans) = t * c;
112            D_IM(ans) = t * s;
113        }
114    }
115    return (ans);
116 }

```

unchanged_portion_omitted

```

*****
2356 Tue Nov 25 12:58:16 2014
new/usr/src/lib/libm/common/complex/cexpf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __cexpf = cexpf
30 #pragma weak cexpf = __cexpf
31
32 #include "libm.h"
33 #include "complex_wrapper.h"
34
35 #if defined(__i386) && !defined(__amd64)
36 extern int __swapRP(int);
37 #endif
38
39 static const float zero = 0.0F;
40
41 fcomplex
42 cexpf(fcomplex z) {
43     fcomplex      ans;
44     float          x, y, c, s;
45     double         t;
46     int            n, ix, iy, hx, hy;
47
48     x = F_RE(z);
49     y = F_IM(z);
50     hx = THE_WORD(x);
51     hy = THE_WORD(y);
52     ix = hx & 0x7fffffff;
53     iy = hy & 0x7fffffff;
54     if (iy == 0) { /* y = 0 */
55         F_RE(ans) = expf(x);
56         F_IM(ans) = y;
57     } else if (ix == 0x7f800000) { /* x is +-inf */
58         if (hx < 0) {

```

```

58         if (iy >= 0x7f800000) {
59             F_RE(ans) = zero;
60             F_IM(ans) = zero;
61         } else {
62             sincosf(y, &s, &c);
63             F_RE(ans) = zero * c;
64             F_IM(ans) = zero * s;
65         }
66     } else {
67         if (iy >= 0x7f800000) {
68             F_RE(ans) = x;
69             F_IM(ans) = y - y;
70         } else {
71             sincosf(y, &s, &c);
72             F_RE(ans) = x * c;
73             F_IM(ans) = x * s;
74         }
75     }
76 } else {
77     sincosf(y, &s, &c);
78     if (ix >= 0x42b171aa) { /* |x| > 88.722... ~ log(2**128) */
79 #if defined(__i386) && !defined(__amd64)
80         int      rp = __swapRP(fp_extended);
81 #endif
82         t = __k_cexp(x, &n);
83         F_RE(ans) = (float)scalbn(t * (double)c, n);
84         F_IM(ans) = (float)scalbn(t * (double)s, n);
85 #if defined(__i386) && !defined(__amd64)
86         if (rp != fp_extended)
87             (void) __swapRP(rp);
88 #endif
89     } else {
90         t = expf(x);
91         F_RE(ans) = t * c;
92         F_IM(ans) = t * s;
93     }
94 }
95 return (ans);
96 }
_____unchanged_portion_omitted_____

```

```

*****
2248 Tue Nov 25 12:58:17 2014
new/usr/src/lib/libm/common/complex/cexpl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cexpl = cexpl
30 #pragma weak cexpl = __cexpl

32 #include "libm.h"          /* expl/isinfl/iszerol/scalbnl/sincosl */
33 #include "complex_wrapper.h"

35 extern int isinfl(long double);
36 extern int iszerol(long double);

38 /* INDENT OFF */
39 static const long double zero = 0.0L;
40 /* INDENT ON */

42 ldcomplex
43 cexpl(ldcomplex z) {
44     ldcomplex ans;
45     long double x, y, t, c, s;
46     int n, ix, iy, hx, hy;

48     x = LD_RE(z);
49     y = LD_IM(z);
50     hx = HI_XWORD(x);
51     hy = HI_XWORD(y);
52     ix = hx & 0x7fffffff;
53     iy = hy & 0x7fffffff;
54     if (iszerol(y)) { /* y = 0 */
55         LD_RE(ans) = expl(x);
56         LD_IM(ans) = y;
57     } else if (isinfl(x)) { /* x is +-inf */

```

```

58         if (hx < 0) {
59             if (iy >= 0x7fff0000) {
60                 LD_RE(ans) = zero;
61                 LD_IM(ans) = zero;
62             } else {
63                 sincosl(y, &s, &c);
64                 LD_RE(ans) = zero * c;
65                 LD_IM(ans) = zero * s;
66             }
67         } else {
68             if (iy >= 0x7fff0000) {
69                 LD_RE(ans) = x;
70                 LD_IM(ans) = y - y;
71             } else {
72                 (void) sincosl(y, &s, &c);
73                 LD_RE(ans) = x * c;
74                 LD_IM(ans) = x * s;
75             }
76         }
77     } else {
78         (void) sincosl(y, &s, &c);
79         if (ix >= 0x400C62E4) { /* |x| > 11356.52... ~ log(2**16384) */
80             t = __k_cexpl(x, &n);
81             LD_RE(ans) = scalbnl(t * c, n);
82             LD_IM(ans) = scalbnl(t * s, n);
83         } else {
84             t = expl(x);
85             LD_RE(ans) = t * c;
86             LD_IM(ans) = t * s;
87         }
88     }
89     return (ans);
90 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/complex/cimag.c

1

1123 Tue Nov 25 12:58:17 2014

new/usr/src/lib/libm/common/complex/cimag.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cimag = cimag
30 #pragma weak cimag = __cimag
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 double
36 cimag(dcomplex z) {
37     return (D_IM(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cimagf.c

1

1125 Tue Nov 25 12:58:17 2014

new/usr/src/lib/libm/common/complex/cimagf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cimagf = cimagf
30 #pragma weak cimagf = __cimagf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 float
36 cimagf(fcomplex z) {
37     return (F_IM(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cimagl.c

1

1133 Tue Nov 25 12:58:18 2014

new/usr/src/lib/libm/common/complex/cimagl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cimagl = cimagl
30 #pragma weak cimagl = __cimagl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 long double
36 cimagl(ldcomplex z) {
37     return (LD_IM(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/clog.c

1

```
*****
3776 Tue Nov 25 12:58:18 2014
new/usr/src/lib/libm/common/complex/clog.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29 #pragma weak clog = __clog

31 /* INDENT OFF */
32 /*
33  * dcomplex clog(dcomplex z);
34  *
35  *
36  * 
$$\log(x+iy) = \log\left(\sqrt{x^2 + y^2}\right) + i \tan^{-1} \frac{y}{x}$$

37  *
38  *
39  *
40  * 
$$= \frac{1}{2} \log(x^2 + y^2) + i \tan^{-1} \frac{y}{x}$$

41  *
42  *
43  *
44  * Note that the arctangent ranges from -PI to +PI, thus the imaginary
45  * part of clog is atan2(y,x).
46  *
47  * EXCEPTION CASES (conform to ISO/IEC 9899:1999(E)):
48  * clog(-0 + i 0 ) = -inf + i pi
49  * clog( 0 + i 0 ) = -inf + i 0
50  * clog( x + i inf ) = -inf + i pi/2, for finite x
51  * clog( x + i NaN ) = NaN + i NaN with invalid for finite x
52  * clog(-inf + iy ) = +inf + i pi, for finite positive-signed y
53  * clog(+inf + iy ) = +inf + i 0, for finite positive-signed y
54  * clog(-inf + i inf)= inf + i 3pi/4
55  * clog(+inf + i inf)= inf + i pi/4
56  * clog(+/-inf+ i NaN)= inf + i NaN
57  * clog(NaN + i y )= NaN + i NaN for finite y
58  * clog(NaN + i inf)= inf + i NaN
```

new/usr/src/lib/libm/common/complex/clog.c

2

```
59 * clog(NaN + i NaN)= NaN + i NaN
60 */
61 /* INDENT ON */

63 #include "libm_synonyms.h"
63 #include <math.h> /* atan2/fabs/log/loglp */
64 #include "complex_wrapper.h"
65 #include "libm_protos.h" /* __k_clog_r */

68 static const double half = 0.5, one = 1.0;

70 dcomplex
71 __clog(dcomplex z) {
72 clog(dcomplex z) {
73     dcomplex      ans;
74     double        x, y, t, ax, ay, w;
75     int           n, ix, iy, hx, hy;
76     unsigned      lx, ly;

77     x = D_RE(z);
78     y = D_IM(z);
79     hx = HI_WORD(x);
80     lx = LO_WORD(x);
81     hy = HI_WORD(y);
82     ly = LO_WORD(y);
83     ix = hx & 0x7fffffff;
84     iy = hy & 0x7fffffff;
85     ax = fabs(x);
86     ay = fabs(y);
87     D_IM(ans) = carg(z);
88     if (ix < iy || (ix == iy && lx < ly)) {
89         /* swap x and y to force ax >= ay */
90         t = ax;
91         ax = ay;
92         ay = t;
93         n = ix, ix = iy;
94         iy = n;
95         n = lx, lx = ly;
96         ly = n;
97     }
98     n = (ix - iy) >> 20;
99     if (ix >= 0x7ff00000) { /* x or y is Inf or NaN */
100         if (ISINF(ix, lx))
101             D_RE(ans) = ax;
102         else if (ISINF(iy, ly))
103             D_RE(ans) = ay;
104         else
105             D_RE(ans) = ax * ay;
106     } else if ((iy | ly) == 0) {
107         D_RE(ans) = ((ix | lx) == 0)? -one / ax : log(ax);
108     } else if (((0x3fffffff - ix) ^ (ix - 0x3fe00000)) >= 0) {
109         /* 0.5 <= x < 2 */
110         if (ix >= 0x3ff00000) {
111             if (((ix - 0x3ff00000) | lx) == 0)
112                 D_RE(ans) = half * loglp(ay * ay);
113             else if (n >= 60)
114                 D_RE(ans) = log(ax);
115             else
116                 D_RE(ans) = half * (loglp(ay * ay + (ax - one) * (ax + one)));
117         } else if (n >= 60) {
118             D_RE(ans) = log(ax);
119         } else {
120             D_RE(ans) = __k_clog_r(ax, ay, &w);
121         }
122     }
}
```

```
123     } else if (n >= 30) {
124         D_RE(ans) = log(ax);
125     } else if (ix < 0x5f300000 && iy >= 0x20b00000) {
126         /* 2**-500 < y < x < 2**500 */
127         D_RE(ans) = half * log(ax * ax + ay * ay);
128     } else {
129         t = ay / ax;
130         D_RE(ans) = log(ax) + half * loglp(t * t);
131     }
132     return (ans);
133 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/clogf.c

1

```
*****
2129 Tue Nov 25 12:58:19 2014
new/usr/src/lib/libm/common/complex/clogf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __clogf = clogf
30 #pragma weak clogf = __clogf
31
32 #include "libm.h"
33 #include "complex_wrapper.h"
34
35 #if defined(__i386) && !defined(__amd64)
36 extern int __swapRP(int);
37 #endif
38
39 fcomplex
40 clogf(fcomplex z) {
41     fcomplex    ans;
42     float       x, y, ax, ay;
43     double      dx, dy;
44     int         ix, iy, hx, hy;
45
46     x = F_RE(z);
47     y = F_IM(z);
48     hx = THE_WORD(x);
49     hy = THE_WORD(y);
50     ix = hx & 0x7fffffff;
51     iy = hy & 0x7fffffff;
52     ay = fabsf(y);
53     ax = fabsf(x);
54     F_IM(ans) = atan2f(y, x);
55     if (ix >= 0x7f800000 || iy >= 0x7f800000) {
56         /* x or y is Inf or NaN */
57         if (iy == 0x7f800000)
58             F_RE(ans) = ay;
```

new/usr/src/lib/libm/common/complex/clogf.c

2

```
58         else if (ix == 0x7f800000)
59             F_RE(ans) = ax;
60         else
61             F_RE(ans) = ax + ay;
62     } else {
63 #if defined(__i386) && !defined(__amd64)
64         int     rp = __swapRP(fp_extended);
65 #endif
66         dx = (double)ax;
67         dy = (double)ay;
68         if (ix == 0x3f800000)
69             F_RE(ans) = (float)(0.5 * loglp(dy * dy));
70         else if (iy == 0x3f800000)
71             F_RE(ans) = (float)(0.5 * loglp(dx * dx));
72         else if ((ix | iy) == 0)
73             F_RE(ans) = -1.0f / ax;
74         else
75             F_RE(ans) = (float)(0.5 * log(dx * dx + dy * dy));
76 #if defined(__i386) && !defined(__amd64)
77         if (rp != fp_extended)
78             (void) __swapRP(rp);
79 #endif
80     }
81     return (ans);
82 }
_____unchanged_portion_omitted_____
```

```

*****
2709 Tue Nov 25 12:58:19 2014
new/usr/src/lib/libm/common/complex/clogl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __clogl = clogl
30 #pragma weak clogl = __clogl

32 #include "libm.h" /* atan2l/fabsl/isinfl/loglpl/logl/___k_clog_rl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"

36 #if defined(__sparc)
37 #define SIGP7 120
38 #define HSIGP7 60
39 #elif defined(__x86)
40 #define SIGP7 70
41 #define HSIGP7 35
42 #endif

44 /* INDENT OFF */
45 static const long double zero = 0.0L, half = 0.5L, one = 1.0L;
46 /* INDENT ON */

48 ldcomplex
49 clogl(ldcomplex z) {
50     ldcomplex ans;
51     long double x, y, t, ax, ay;
52     int n, ix, iy, hx, hy;

54     x = LD_RE(z);
55     y = LD_IM(z);
56     hx = HI_XWORD(x);
57     hy = HI_XWORD(y);

```

```

58     ix = hx & 0x7fffffff;
59     iy = hy & 0x7fffffff;
60     ay = fabsl(y);
61     ax = fabsl(x);
62     LD_IM(ans) = atan2l(y, x);
63     if (ix < iy || (ix == iy && ix < 0x7fff0000 && ax < ay)) {
64         /* swap x and y to force ax>=ay */
65         t = ax;
66         ax = ay;
67         ay = t;
68         n = ix, ix = iy;
69         iy = n;
70     }
71     n = (ix - iy) >> 16;
72     if (ix >= 0x7fff0000) { /* x or y is Inf or NaN */
73         if (isinfl(ax))
74             LD_RE(ans) = ax;
75         else if (isinfl(ay))
76             LD_RE(ans) = ay;
77         else
78             LD_RE(ans) = ax + ay;
79     } else if (ay == zero)
80         LD_RE(ans) = logl(ax);
81     else if (((0x3fffffff - ix) ^ (ix - 0x3ffe0000)) >= 0) {
82         /* 0.5 <= x < 2 */
83         if (ix >= 0x3fff0000) {
84             if (ax == one)
85                 LD_RE(ans) = half * loglpl(ay * ay);
86             else if (n >= SIGP7)
87                 LD_RE(ans) = logl(ax);
88             else
89                 LD_RE(ans) = half * (loglpl(ay * ay + (ax -
90                                     one) * (ax + one)));
91         } else if (n >= SIGP7)
92             LD_RE(ans) = logl(ax);
93         else
94             LD_RE(ans) = ___k_clog_rl(x, y, &t);
95     } else if (n >= HSIGP7)
96         LD_RE(ans) = logl(ax);
97     else if (ix < 0x5f3f0000 && iy >= 0x20bf0000)
98         /* 2**-8000 < y < x < 2**8000 */
99         LD_RE(ans) = half * logl(ax * ax + ay * ay);
100     else {
101         t = ay / ax;
102         LD_RE(ans) = logl(ax) + half * loglpl(t * t);
103     }
104     return (ans);
105 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/conj.c

1

1137 Tue Nov 25 12:58:20 2014

new/usr/src/lib/libm/common/complex/conj.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __conj = conj
30 #pragma weak conj = __conj
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 dcomplex
36 conj(dcomplex z) {
37     D_IM(z) = -D_IM(z);
38     return (z);
39 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/conjf.c

1

1140 Tue Nov 25 12:58:21 2014

new/usr/src/lib/libm/common/complex/conjf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __conjf = conjf
30 #pragma weak conjf = __conjf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 conjf(fcomplex z) {
37     F_IM(z) = -F_IM(z);
38     return (z);
39 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/conjl.c

1

1144 Tue Nov 25 12:58:21 2014

new/usr/src/lib/libm/common/complex/conjl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __conjl = conjl
30 #pragma weak conjl = __conjl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 conjl(ldcomplex z) {
37     LD_IM(z) = -LD_IM(z);
38     return (z);
39 }
```

unchanged_portion_omitted


```

*****
9501 Tue Nov 25 12:58:22 2014
new/usr/src/lib/libm/common/complex/cpow.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cpow = cpow
30 #pragma weak cpow = __cpow

32 /* INDENT OFF */
33 /*
34  * dcomplex cpow(dcomplex z);
35  *
36  * z**w analytically equivalent to
37  *
38  * cpow(z,w) = cexp(w clog(z))
39  *
40  * Let z = x+iy, w = u+iv.
41  * Since
42  *
43  *
44  * 
$$\log(x+iy) = \log\left(\sqrt{x^2 + y^2}\right) + i \tan^{-1} \frac{y}{x}$$

45  *
46  *
47  * 
$$= \frac{1}{2} \log(x^2 + y^2) + i \tan^{-1} \frac{y}{x}$$

48  *
49  *
50  * 
$$(u+iv) * \log(x+iy) = \frac{u}{2} \log(x^2 + y^2) - v \tan^{-1} \frac{y}{x} + \quad (1)$$

51  *
52  *
53  *
54  * 
$$i * \left[ \frac{v}{2} \log(x^2 + y^2) + u \tan^{-1} \frac{y}{x} \right] \quad (2)$$

55  *
56  *
57  *

```

```

58 *
59 *
60 * Therefore,
61 * 
$$w^{r+iq} = e^{r \log(w) + i q \log(w)}$$

62 * 
$$z = e^{r \log(x+iy) + i q \operatorname{atan2}(y,x)}$$

63 *
64 *
65 * Here e can be expressed as:  $u \sqrt{x^2 + y^2}^{-v \operatorname{atan2}(y,x)}$ 
66 *
67 *
68 * Special cases (in the order of appearance):
69 * 1. (anything) ** 0 is 1
70 * 2. (anything) ** 1 is itself
71 * 3. When v = 0, y = 0:
72 * If x is finite and negative, and u is finite, then
73 *  $x ** u = \exp(u \pi i) * \operatorname{pow}(|x|, u);$ 
74 * otherwise,
75 *  $x ** u = \operatorname{pow}(x, u);$ 
76 * 4. When v = 0, x = 0 or |x| = |y| or x is inf or y is inf:
77 *  $(x + y i) ** u = r * \exp(q i)$ 
78 * where
79 *  $r = \operatorname{hypot}(x,y) ** u$ 
80 *  $q = u * \operatorname{atan2pi}(y, x)$ 
81 *
82 * 5. otherwise, z**w is NAN if any x, y, u, v is a Nan or inf
83 *
84 * Note: many results of special cases are obtained in terms of
85 * polar coordinate. In the conversion from polar to rectangle:
86 *  $r \exp(q i) = r * \cos(q) + r * \sin(q) i,$ 
87 * we regard r * 0 is 0 except when r is a NaN.
88 */
89 /* INDENT ON */

91 #include "libm.h" /* atan2/exp/fabs/hypot/log/pow/scalbn */
92 /* atan2pi/exp2/sincos/sincospi/_k_clog_r/_k_atan2 */
93 #include "complex_wrapper.h"

95 extern void sincospi(double, double *, double *);

97 static const double
98 huge = 1e300,
99 tiny = 1e-300,
100 invln2 = 1.44269504088896338700e+00,
101 ln2hi = 6.93147180369123816490e-01, /* 0x3fe62e42, 0xfee00000 */
102 ln2lo = 1.90821492927058770002e-10, /* 0x3dea39ef, 0x35793c76 */
103 one = 1.0,
104 zero = 0.0;

106 static const int hiinf = 0x7ff00000;
107 extern double atan2pi(double, double);

109 /*
110  * Assuming |t[0]| > |t[1]| and |t[2]| > |t[3]|, sum4fp subroutine
111  * compute t[0] + t[1] + t[2] + t[3] into two double fp numbers.
112  */
113 static double
114 sum4fp(double ta[], double *w) {
115     double t1, t2, t3, t4, w1, w2, t;
116     t1 = ta[0]; t2 = ta[1]; t3 = ta[2]; t4 = ta[3];
117     /*
118      * Rearrange ti so that |t1| >= |t2| >= |t3| >= |t4|
119      */
120     if (fabs(t4) > fabs(t1)) {
121         t = t1; t1 = t3; t3 = t;
122         t = t2; t2 = t4; t4 = t;
123     } else if (fabs(t3) > fabs(t1)) {

```

```
124         t = t1; t1 = t3;
125         if (fabs(t4) > fabs(t2)) {
126             t3 = t4; t4 = t2; t2 = t;
127         } else {
128             t3 = t2; t2 = t;
129         }
130     } else if (fabs(t3) > fabs(t2)) {
131         t = t2; t2 = t3;
132         if (fabs(t4) > fabs(t2)) {
133             t3 = t4; t4 = t;
134         } else
135             t3 = t;
136     }
137     /* summing r = t1 + t2 + t3 + t4 to w1 + w2 */
138     w1 = t3 + t4;
139     w2 = t4 - (w1 - t3);
140     t = t2 + w1;
141     w2 += w1 - (t - t2);
142     w1 = t + w2;
143     w2 += t - w1;
144     t = t1 + w1;
145     w2 += w1 - (t - t1);
146     w1 = t + w2;
147     *w = w2 - (w1 - t);
148     return (w1);
149 }
```

unchanged_portion_omitted

```

*****
4751 Tue Nov 25 12:58:22 2014
new/usr/src/lib/libm/common/complex/cpowf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __cpowf = cpowf
30 #pragma weak cpowf = __cpowf
31
32 #include "libm.h"
33 #include "complex_wrapper.h"
34
35 extern void sincospi(double, double *, double *);
36 extern void sincospif(float, float *, float *);
37 extern double atan2pi(double, double);
38 extern float atan2pif(float, float);
39
40 #if defined(__i386) && !defined(__amd64)
41 extern int __swapRP(int);
42 #endif
43
44 static const double
45 dpi = 3.1415926535897931160E0, /* Hex 2^ 1 * 1.921FB54442D18 */
46 dhalf = 0.5,
47 dsqrt2 = 1.41421356237309514547, /* 3FF6A09E 667F3BCD */
48 dinvpi = 0.3183098861837906715377675;
49
50 static const float one = 1.0F, zero = 0.0F;
51
52 #define hiinf 0x7f800000
53
54 fcomplex
55 cpowf(fcomplex z, fcomplex w) {
56     fcomplex ans;
57     float x, y, u, v, t, c, s;
58     double dx, dy, du, dv, dt, dc, ds, dp, dq, dr;

```

```

58     int ix, iy, hx, hy, hv, hu, iu, iv, j;
59
60     x = F_RE(z);
61     y = F_IM(z);
62     u = F_RE(w);
63     v = F_IM(w);
64     hx = THE_WORD(x);
65     hy = THE_WORD(y);
66     hu = THE_WORD(u);
67     hv = THE_WORD(v);
68     ix = hx & 0x7fffffff;
69     iy = hy & 0x7fffffff;
70     iu = hu & 0x7fffffff;
71     iv = hv & 0x7fffffff;
72
73     j = 0;
74     if (iv == 0) { /* z**(real) */
75         if (hu == 0x3f800000) { /* (anything) ** 1 is itself */
76             F_RE(ans) = x;
77             F_IM(ans) = y;
78         } else if (iu == 0) { /* (anything) ** 0 is 1 */
79             F_RE(ans) = one;
80             F_IM(ans) = zero;
81         } else if (iy == 0) { /* (real)**(real) */
82             F_IM(ans) = zero;
83             if (hx < 0 && ix < hiinf && iu < hiinf) {
84                 /* -x ** u is exp(i*pi*u)*pow(x,u) */
85                 t = powf(-x, u);
86                 sincospif(u, &s, &c);
87                 F_RE(ans) = (c == zero)? c: c * t;
88                 F_IM(ans) = (s == zero)? s: s * t;
89             } else {
90                 F_RE(ans) = powf(x, u);
91             }
92         } else if (ix == 0 || ix >= hiinf || iy >= hiinf) {
93             if (ix > hiinf || iy > hiinf || iu > hiinf) {
94                 F_RE(ans) = F_IM(ans) = x + y + u;
95             } else {
96                 v = fabsf(y);
97                 if (ix != 0)
98                     v += fabsf(x);
99                 t = atan2pif(y, x);
100                sincospif(t * u, &s, &c);
101                F_RE(ans) = (c == zero)? c: c * v;
102                F_IM(ans) = (s == zero)? s: s * v;
103            }
104        } else if (ix == iy) { /* if |x| == |y| */
105            #if defined(__i386) && !defined(__amd64)
106                int rp = __swapRP(fp_extended);
107            #endif
108            dx = (double)x;
109            du = (double)u;
110            dt = (hx >= 0)? 0.25 : 0.75;
111            if (hy < 0)
112                dt = -dt;
113            dr = pow(dsqrt2 * dx, du);
114            sincospi(dt * du, &ds, &dc);
115            F_RE(ans) = (float)(dr * dc);
116            F_IM(ans) = (float)(dr * ds);
117            #if defined(__i386) && !defined(__amd64)
118                if (rp != fp_extended)
119                    (void) __swapRP(rp);
120            #endif
121        } else {
122            j = 1;
123        }

```

```

124         if (j == 0)
125             return (ans);
126     }
127     if (iu >= hiinf || iv >= hiinf || ix >= hiinf || iy >= hiinf) {
128         /*
129          * non-zero imaginary part(s) with inf component(s) yields NaN
130          */
131         t = fabsf(x) + fabsf(y) + fabsf(u) + fabsf(v);
132         F_RE(ans) = F_IM(ans) = t - t;
133     } else {
134 #if defined(__i386) && !defined(__amd64)
135         int    rp = __swapRP(fp_extended);
136 #endif
137         /* INDENT OFF */
138         /*
139          * r = u*log(hypot(x,y))-v*atan2(y,x),
140          * q = u*atan2(y,x)+v*log(hypot(x,y))
141          * or
142          * r = u*log(hypot(x,y))-v*pi*atan2pi(y,x),
143          * q/pi = u*atan2pi(y,x)+v*log(hypot(x,y))/pi
144          * ans = exp(r)*(cospi(q/pi) + i sinpi(q/pi))
145          */
146         /* INDENT ON */
147         dx = (double)x;
148         dy = (double)y;
149         du = (double)u;
150         dv = (double)v;
151         if (ix > 0x3f000000 && ix < 0x40000000) /* .5 < |x| < 2 */
152             dt = dhalf * loglp((dx - 1.0) * (dx + 1.0) + dy * dy);
153         else if (iy > 0x3f000000 && iy < 0x40000000) /* .5 < |y| < 2 */
154             dt = dhalf * loglp((dy - 1.0) * (dy + 1.0) + dx * dx);
155         else
156             dt = dhalf * log(dx * dx + dy * dy);
157         dp = atan2pi(dy, dx);
158         if (iv == 0) { /* dv = 0 */
159             dr = exp(du * dt);
160             dq = du * dp;
161         } else {
162             dr = exp(du * dt - dv * dp * dpi);
163             dq = du * dp + dv * dt * dinvpi;
164         }
165         sincospi(dq, &ds, &dc);
166         F_RE(ans) = (float)(dr * dc);
167         F_IM(ans) = (float)(dr * ds);
168 #if defined(__i386) && !defined(__amd64)
169         if (rp != fp_extended)
170             (void) __swapRP(rp);
171 #endif
172     }
173     return (ans);
174 }

```

unchanged_portion_omitted

```

*****
7657 Tue Nov 25 12:58:23 2014
new/usr/src/lib/libm/common/complex/cpow1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __cpow1 = cpow1
30 #pragma weak cpow1 = __cpow1

32 #include "libm.h" /* __k_clog_rl/__k_atan2l */
33 /* atan2l/atan2pil/exp2l/exp1/fabs1/hypot1/isinfl/log1/pow1/sincosl/sincospil */
34 #include "complex_wrapper.h"
35 #include "longdouble.h"

37 #if defined(__sparc)
38 #define HALF(x) ((int *) &x)[3] = 0; ((int *) &x)[2] &= 0xfe000000
39 #define LAST(x) ((int *) &x)[3]
40 #elif defined(__x86)
41 #define HALF(x) ((int *) &x)[0] = 0
42 #define LAST(x) ((int *) &x)[0]
43 #endif

45 /* INDENT OFF */
46 static const int hiinf = 0x7fff0000;
47 static const long double
48     tiny = 1.0e-4000L,
49     huge = 1.0e4000L,
50 #if defined(__x86)
51     /* 43 significant bits, 21 trailing zeros */
52     ln2hil = 0.693147180559890330187045037746429443359375L,
53     ln2lol = 5.497923018708371174712471612513436025525412068e-14L,
54 #else /* sparc */
55     /* 0x3FF962E4 2FEFA39E F35793C7 00000000 */
56     ln2hil = 0.693147180559945309417231592858066493070671489074L,
57     ln2lol = 5.28600110075004828645286235820646730106802446566153e-25L,

```

```

58 #endif
59     invln2 = 1.442695040888963407359924681001892137427e+0000L,
60     one = 1.0L,
61     zero = 0.0L;
62 /* INDENT ON */

64 /*
65  * Assuming |t[0]| > |t[1]| and |t[2]| > |t[3]|, sum4fp1 subroutine
66  * compute t[0] + t[1] + t[2] + t[3] into two long double fp numbers.
67 */
68 static long double sum4fp1(long double ta[], long double *w)
69 {
70     long double t1, t2, t3, t4, w1, w2, t;
71     t1 = ta[0]; t2 = ta[1]; t3 = ta[2]; t4 = ta[3];
72     /*
73      * Rearrange ti so that |t1| >= |t2| >= |t3| >= |t4|
74      */
75     if (fabsl(t4) > fabsl(t1)) {
76         t = t1; t1 = t3; t3 = t;
77         t = t2; t2 = t4; t4 = t;
78     } else if (fabsl(t3) > fabsl(t1)) {
79         t = t1; t1 = t3;
80         if (fabsl(t4) > fabsl(t2)) {
81             t3 = t4; t4 = t2; t2 = t;
82         } else {
83             t3 = t2; t2 = t;
84         }
85     } else if (fabsl(t3) > fabsl(t2)) {
86         t = t2; t2 = t3;
87         if (fabsl(t4) > fabsl(t2)) {
88             t3 = t4; t4 = t;
89         } else
90             t3 = t;
91     }
92     /* summing r = t1 + t2 + t3 + t4 to w1 + w2 */
93     w1 = t3 + t4;
94     w2 = t4 - (w1 - t3);
95     t = t2 + w1;
96     w2 += w1 - (t - t2);
97     w1 = t + w2;
98     w2 += t - w1;
99     t = t1 + w1;
100    w2 += w1 - (t - t1);
101    w1 = t + w2;
102    *w = w2 - (w1 - t);
103    return (w1);
104 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cproj.c

1

```
*****
1721 Tue Nov 25 12:58:23 2014
new/usr/src/lib/libm/common/complex/cproj.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __cproj = cproj
30 #pragma weak cproj = __cproj
31
32 /* INDENT OFF */
33 /*
34  * dcomplex cproj(dcomplex z);
35  *
36  * If one of the component of z = (x,y) is an inf, then
37  *   cproj(z) = (+inf, copysign(0,y));
38  * otherwise,
39  *   cproj(z) = z
40  */
41 /* INDENT ON */
42
43 #include "libm.h" /* fabs */
44 #include "complex_wrapper.h"
45
46 static const double zero = 0.0;
47
48 dcomplex
49 cproj(dcomplex z) {
50     double x, y;
51     int ix, iy, hx, hy, lx, ly;
52
53     x = D_RE(z);
54     y = D_IM(z);
55     hx = HI_WORD(x);
56     lx = LO_WORD(x);
57     hy = HI_WORD(y);
```

new/usr/src/lib/libm/common/complex/cproj.c

2

```
58     ly = LO_WORD(y);
59     ix = hx & 0x7fffffff;
60     iy = hy & 0x7fffffff;
61     if (ISINF(iy, ly)) {
62         D_RE(z) = fabs(y);
63         D_IM(z) = hy >= 0 ? zero : -zero;
64     } else if (ISINF(ix, lx)) {
65         D_RE(z) = fabs(x);
66         D_IM(z) = hy >= 0 ? zero : -zero;
67     }
68     return (z);
69 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/cprojf.c

1

1511 Tue Nov 25 12:58:24 2014

new/usr/src/lib/libm/common/complex/cprojf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cprojf = cprojf
30 #pragma weak cprojf = __cprojf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 /* INDENT OFF */
36 static const float zero = 0.0F;
37 /* INDENT ON */
```

```
39 fcomplex
40 cprojf(fcomplex z) {
41     float x, y;
42     int ix, iy, hx, hy;
43
44     x = F_RE(z);
45     y = F_IM(z);
46     hx = THE_WORD(x);
47     hy = THE_WORD(y);
48     ix = hx & 0x7fffffff;
49     iy = hy & 0x7fffffff;
50     if (iy == 0x7f800000) {
51         F_RE(z) = fabsf(y);
52         F_IM(z) = hy >= 0 ? zero : -zero;
53     } else if (ix == 0x7f800000) {
54         F_RE(z) = fabsf(x);
55         F_IM(z) = hy >= 0 ? zero : -zero;
56     }
57     return (z);
```

new/usr/src/lib/libm/common/complex/cprojf.c

2

```
58 }
    _____  
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/complex/cproj1.c

1

```
*****
1542 Tue Nov 25 12:58:24 2014
new/usr/src/lib/libm/common/complex/cproj1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __cproj1 = cproj1
30 #pragma weak cproj1 = __cproj1
```

```
32 #include "libm.h" /* fabsl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"
```

```
36 /* INDENT OFF */
37 static const long double zero = 0.0L;
38 /* INDENT ON */
```

```
40 ldcomplex
41 cproj1(ldcomplex z) {
42     long double x, y;
43     int hy;
44
45     x = LD_RE(z);
46     y = LD_IM(z);
47     #if defined(__x86)
48     hy = ((int *) &y)[2] << 16;
49 #else
50     hy = ((int *) &y)[0];
51 #endif
52     if (isinf1(y)) {
53         LD_RE(z) = fabs1(y);
54         LD_IM(z) = hy >= 0 ? zero : -zero;
55     } else if (isinf1(x)) {
56         LD_RE(z) = fabs1(x);
57         LD_IM(z) = hy >= 0 ? zero : -zero;
```

new/usr/src/lib/libm/common/complex/cproj1.c

2

```
58     }
59     return (z);
60 }
_____unchanged_portion_omitted_
```


new/usr/src/lib/libm/common/complex/creal.c

1

1123 Tue Nov 25 12:58:25 2014

new/usr/src/lib/libm/common/complex/creal.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __creal = creal
30 #pragma weak creal = __creal
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 double
36 creal(dcomplex z) {
37     return (D_RE(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/crealf.c

1

1125 Tue Nov 25 12:58:25 2014

new/usr/src/lib/libm/common/complex/crealf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __crealf = crealf
30 #pragma weak crealf = __crealf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 float
36 crealf(fcomplex z) {
37     return (F_RE(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/creall.c

1

1133 Tue Nov 25 12:58:26 2014

new/usr/src/lib/libm/common/complex/creall.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __creall = creall
30 #pragma weak creall = __creall
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 long double
36 creall(ldcomplex z) {
37     return (LD_RE(z));
38 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/csin.c

1

```
*****
1558 Tue Nov 25 12:58:26 2014
new/usr/src/lib/libm/common/complex/csin.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __csin = csin
30 #pragma weak csin = __csin

32 /* INDENT OFF */
33 /*
34  * dcomplex csin(dcomplex z);
35  *
36  * If z = x+iy, then since csin(iz) = i*c sinh(z), we have
37  *
38  * csin(z)      = csin((-1)*(-z)) = csin(i*i*(-z))
39  *              = i*c sinh(i*(-z)) = i*c sinh(i*(-x-yi))
40  *              = i*c sinh(y-ix)
41  *              = -Im(c sinh(y-ix))+i*Re(c sinh(y-ix))
42  */
43 /* INDENT ON */

45 #include "libm.h"
46 #include "complex_wrapper.h"

48 dcomplex
49 csin(dcomplex z) {
50     double x, y;
51     dcomplex ans, ct;

53     x = D_RE(z);
54     y = D_IM(z);
55     D_RE(z) = y;
56     D_IM(z) = -x;
57     ct = csinh(z);
```

new/usr/src/lib/libm/common/complex/csin.c

2

```
58     D_RE(ans) = -D_IM(ct);
59     D_IM(ans) = D_RE(ct);
60     return (ans);
61 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/complex/csinf.c

1

1275 Tue Nov 25 12:58:26 2014

new/usr/src/lib/libm/common/complex/csinf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __csinf = csinf
30 #pragma weak csinf = __csinf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 csinf(fcomplex z) {
37     float x, y;
38     fcomplex ans, ct;
39
40     x = F_RE(z);
41     y = F_IM(z);
42     F_RE(z) = y;
43     F_IM(z) = -x;
44     ct = csinhf(z);
45     F_RE(ans) = -F_IM(ct);
46     F_IM(ans) = F_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

```

*****
3923 Tue Nov 25 12:58:27 2014
new/usr/src/lib/libm/common/complex/csinh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __csinh = csinh
30 #pragma weak csinh = __csinh

32 /* INDENT OFF */
33 /*
34  * dcomplex csinh(dcomplex z);
35  *
36  *          z      -z      x      -x
37  *          e      e      e (cos(y)+i*sin(y)) - e (cos(-y)+i*sin(-y))
38  * sinh z = ----- = -----
39  *          2          2
40  *          x      -x      x      -x
41  *          cos(y) ( e - e ) + i*sin(y) ( e + e )
42  *          = -----
43  *          2
44  *          = cos(y) sinh(x) + i sin(y) cosh(x)
45  *
46  * Implementation Note
47  * -----
48  *
49  *
50  * Note that  $e^{|x|} + e^{-|x|} = e^{|x|} (1 + e^{-2|x|})$ . If  $e^{-2|x|} < 2^{-P-4}$ , where
51  * P stands for the number of significant bits of the machine precision,
52  *
53  * then the result will be rounded to  $e^{|x|}$ . Therefore, we have
54  *
55  *
56  *
57  *

```

```

58 *
59 *      e
60 *      sinh z = ----- if |x| >= (P/2 + 2)*ln2
61 *      2
62 * EXCEPTION (conform to ISO/IEC 9899:1999(E)):
63 *      csinh(0,0)=(0,0)
64 *      csinh(0,inf)=(+0,NaN)
65 *      csinh(0,NaN)=(+0,NaN)
66 *      csinh(x,inf) = (NaN,NaN) for finite positive x
67 *      csinh(x,NaN) = (NaN,NaN) for finite non-zero x
68 *      csinh(inf,0) = (inf, 0)
69 *      csinh(inf,y) = (inf*cos(y),inf*sin(y)) for positive finite y
70 *      csinh(inf,inf) = (+-inf,NaN)
71 *      csinh(inf,NaN) = (+-inf,NaN)
72 *      csinh(NaN,0) = (NaN,0)
73 *      csinh(NaN,y) = (NaN,NaN) for non-zero y
74 *      csinh(NaN,NaN) = (NaN,NaN)
75 */
76 /* INDENT ON */

78 #include "libm.h"          /* cosh/exp/fabs/scalbn/sinh/sincos/__k_cexp */
79 #include "complex_wrapper.h"

81 dcomplex
82 csinh(dcomplex z) {
83     double t, x, y, S, C;
84     int hx, ix, lx, hy, iy, ly, n;
85     dcomplex ans;

87     x = D_RE(z);
88     y = D_IM(z);
89     hx = HI_WORD(x);
90     lx = LO_WORD(x);
91     ix = hx & 0x7fffffff;
92     hy = HI_WORD(y);
93     ly = LO_WORD(y);
94     iy = hy & 0x7fffffff;
95     x = fabs(x);
96     y = fabs(y);

98     (void) sincos(y, &S, &C);
99     if (ix >= 0x403c0000) { /* |x| > 28 = prec/2 (14,28,34,60) */
100         if (ix >= 0x40862E42) { /* |x| > 709.78... ~ log(2**1024) */
101             if (ix >= 0x7ff00000) { /* |x| is inf or NaN */
102                 if ((iy | ly) == 0) {
103                     D_RE(ans) = x;
104                     D_IM(ans) = y;
105                 } else if (iy >= 0x7ff00000) {
106                     D_RE(ans) = x;
107                     D_IM(ans) = x - y;
108                 } else {
109                     D_RE(ans) = C * x;
110                     D_IM(ans) = S * x;
111                 }
112             } else {
113                 /* return exp(x)=t*2**n */
114                 t = __k_cexp(x, &n);
115                 D_RE(ans) = scalbn(C * t, n - 1);
116                 D_IM(ans) = scalbn(S * t, n - 1);
117             }
118         } else {
119             t = exp(x) * 0.5;
120             D_RE(ans) = C * t;
121             D_IM(ans) = S * t;
122         }
123     } else {

```

```
124         if ((ix | lx) == 0) { /* x = 0, return (0,S) */
125             D_RE(ans) = 0.0;
126             D_IM(ans) = S;
127         } else {
128             D_RE(ans) = C * sinh(x);
129             D_IM(ans) = S * cosh(x);
130         }
131     }
132     if (hx < 0)
133         D_RE(ans) = -D_RE(ans);
134     if (hy < 0)
135         D_IM(ans) = -D_IM(ans);
136     return (ans);
137 }
_____unchanged_portion_omitted_____
```

```

*****
2552 Tue Nov 25 12:58:27 2014
new/usr/src/lib/libm/common/complex/csinhf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
28
29 #pragma weak __csinhf = csinhf
30 #pragma weak csinhf = __csinhf
31
32 #include "libm.h"
33 #include "complex_wrapper.h"
34
35 #if defined(__i386) && !defined(__amd64)
36 extern int __swapRP(int);
37 #endif
38
39 static const float zero = 0.0F, half = 0.5F;
40
41 fcomplex
42 csinhf(fcomplex z) {
43     float      x, y, S, C;
44     double     t;
45     int        hx, ix, hy, iy, n;
46     fcomplex   ans;
47
48     x = F_RE(z);
49     y = F_IM(z);
50     hx = THE_WORD(x);
51     ix = hx & 0x7fffffff;
52     hy = THE_WORD(y);
53     iy = hy & 0x7fffffff;
54     x = fabsf(x);
55     y = fabsf(y);
56
57     sincosf(y, &S, &C);
58     if (ix >= 0x41600000) { /* |x| > 14 = prec/2 (14,28,34,60) */

```

```

58         if (ix >= 0x42B171AA) { /* |x| > 88.722... ~ log(2**128) */
59             if (ix >= 0x7f800000) { /* |x| is inf or NaN */
60                 if (iy == 0) {
61                     F_RE(ans) = x;
62                     F_IM(ans) = y;
63                 } else if (iy >= 0x7f800000) {
64                     F_RE(ans) = x;
65                     F_IM(ans) = x - y;
66                 } else {
67                     F_RE(ans) = C * x;
68                     F_IM(ans) = S * x;
69                 }
70             } else {
71 #if defined(__i386) && !defined(__amd64)
72                 int    rp = __swapRP(fp_extended);
73 #endif
74                 /* return (C, S) * exp(x) / 2 */
75                 t = __k_cexp((double)x, &n);
76                 F_RE(ans) = (float)scalbn(C * t, n - 1);
77                 F_IM(ans) = (float)scalbn(S * t, n - 1);
78 #if defined(__i386) && !defined(__amd64)
79                 if (rp != fp_extended)
80                     (void) __swapRP(rp);
81 #endif
82             }
83         } else {
84             t = expf(x) * half;
85             F_RE(ans) = C * t;
86             F_IM(ans) = S * t;
87         }
88     } else {
89         if (ix == 0) { /* x = 0, return (0,S) */
90             F_RE(ans) = zero;
91             F_IM(ans) = S;
92         } else {
93             F_RE(ans) = C * sinh(x);
94             F_IM(ans) = S * cosh(x);
95         }
96     }
97     if (hx < 0)
98         F_RE(ans) = -F_RE(ans);
99     if (hy < 0)
100         F_IM(ans) = -F_IM(ans);
101     return (ans);
102 }

```

unchanged portion omitted


```

*****
2395 Tue Nov 25 12:58:28 2014
new/usr/src/lib/libm/common/complex/csinhl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __csinhl = csinhl
30 #pragma weak csinhl = __csinhl

32 #include "libm.h" /* coshl/expl/fabsl/scalbnl/sincosl/sinhl/__k_cexpl */
33 #include "complex_wrapper.h"

35 /* INDENT OFF */
36 static const long double zero = 0.0L, half = 0.5L;
37 /* INDENT ON */

39 ldcomplex
40 csinhl(ldcomplex z) {
41     long double t, x, y, S, C;
42     int hx, ix, hy, iy, n;
43     ldcomplex ans;

45     x = LD_RE(z);
46     y = LD_IM(z);
47     hx = HI_XWORD(x);
48     ix = hx & 0x7fffffff;
49     hy = HI_XWORD(y);
50     iy = hy & 0x7fffffff;
51     x = fabsl(x);
52     y = fabsl(y);

54     (void) sincosl(y, &S, &C);
55     if (ix >= 0x4004e000) { /* |x| > 60 = prec/2 (14,28,34,60) */
56         if (ix >= 0x400C62E4) { /* |x| > 11356.52... ~ log(2**16384) */
57             if (ix >= 0x7fff0000) { /* |x| is inf or NaN */

```

```

58         if (y == zero) {
59             LD_RE(ans) = x;
60             LD_IM(ans) = y;
61         } else if (iy >= 0x7fff0000) {
62             LD_RE(ans) = x;
63             LD_IM(ans) = x - y;
64         } else {
65             LD_RE(ans) = C * x;
66             LD_IM(ans) = S * x;
67         }
68     } else {
69         /* return exp(x)=t*2**n */
70         t = __k_cexpl(x, &n);
71         LD_RE(ans) = scalbnl(C * t, n - 1);
72         LD_IM(ans) = scalbnl(S * t, n - 1);
73     }
74 } else {
75     t = expl(x) * half;
76     LD_RE(ans) = C * t;
77     LD_IM(ans) = S * t;
78 }
79 } else {
80     if (x == zero) { /* x = 0, return (0,S) */
81         LD_RE(ans) = zero;
82         LD_IM(ans) = S;
83     } else {
84         LD_RE(ans) = C * sinhl(x);
85         LD_IM(ans) = S * coshl(x);
86     }
87 }
88 if (hx < 0)
89     LD_RE(ans) = -LD_RE(ans);
90 if (hy < 0)
91     LD_IM(ans) = -LD_IM(ans);
92 return (ans);
93 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/csinl.c

1

1292 Tue Nov 25 12:58:28 2014

new/usr/src/lib/libm/common/complex/csinl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __csinl = csinl
30 #pragma weak csinl = __csinl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 csinl(ldcomplex z) {
37     long double x, y;
38     ldcomplex ans, ct;

40     x = LD_RE(z);
41     y = LD_IM(z);
42     LD_RE(z) = y;
43     LD_IM(z) = -x;
44     ct = csinhl(z);
45     LD_RE(ans) = -LD_IM(ct);
46     LD_IM(ans) = LD_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/complex/csqrt.c

1

```
*****
5665 Tue Nov 25 12:58:28 2014
new/usr/src/lib/libm/common/complex/csqrt.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __csqrt = csqrt
30 #pragma weak csqrt = __csqrt
31
32 /* INDENT OFF */
33 /*
34  * dcomplex csqrt(dcomplex z);
35  *
36  * Let  $w=r+is = \sqrt{x+iy}$ . Then  $(r + is)^2 = r^2 - s^2 + i 2sr = x + iy$ .
37  * Hence  $x = r^2 - s^2$ ,  $y = 2sr$ .
38  *
39  * Note that  $x^2+y^2 = (s^2+r^2)^2$ . Thus, we have
40  *
41  * (1)  $r^2 + s^2 = \sqrt{\frac{x^2 + y^2}{2}}$ ,
42  *
43  * (2)  $r^2 - s^2 = x$ 
44  *
45  * (3)  $2sr = y$ .
46  *
47  * Perform (1)-(2) and (1)+(2), we obtain
48  *
49  * (4)  $2r = \sqrt{\frac{x^2 + y^2}{2}} + x$ ,
50  *
51  * (5)  $2s = \sqrt{\frac{x^2 + y^2}{2}} - x$ 
52  *
53  *
54  *
55  *
56  *
57  *
58  *
59  *
60  *
61  *
62  *
63  *
64  *
65  *
66  *
67  *
68  *
69  *
70  *
71  *
72  *
73  *
74  *
75  *
76  *
77  *
78  *
79  *
80  *
81  *
82  *
83  *
84  *
85  *
86  *
87  *
88  *
89  *
90  *
91  *
92  *
93  *
94  *
95  *
96  *
97  *
98  *
99  *
100 *
101 *
102 */
103 /* INDENT ON */
104
105 #include "libm.h"
106 #include "complex_wrapper.h"
107
108 /* INDENT OFF */
109 static const double
110 two300 = 2.03703597633448608627e+90,
111 twom300 = 4.90909346529772655310e-91,
112 two599 = 2.07475778444049647926e+180,
113 twom601 = 1.20495993255144205887e-181,
114 two = 2.0,
115 zero = 0.0,
116 half = 0.5;
117 /* INDENT ON */
118
119 dcomplex
120 csqrt(dcomplex z) {
121     dcomplex ans;
122     double x, y, t, ax, ay;
123     int n, ix, iy, hx, hy, lx, ly;
```

new/usr/src/lib/libm/common/complex/csqrt.c

2

```
58 *
59 *
60 * where  $\text{hypot}(x,y) = \sqrt{x^2 + y^2}$ .
61 *
62 * In order to avoid numerical cancellation, we use formula (4) for
63 * positive x, and (5) for negative x. The other component is then
64 * computed by formula (3).
65 *
66 *
67 * ALGORITHM
68 * -----
69 *
70 * (assume x and y are of medium size, i.e., no over/underflow in squaring)
71 *
72 * If  $x \geq 0$  then
73 *
74 *
75 * 
$$r = \frac{\sqrt{\frac{x^2 + y^2}{2}} + x}{2}, \quad s = \frac{y}{2r}; \quad (6)$$

76 *
77 *
78 * (note that we choose  $\text{sign}(s) = \text{sign}(y)$  to force  $r \geq 0$ ).
79 * Otherwise,
80 *
81 *
82 * 
$$s = \frac{\sqrt{\frac{x^2 + y^2}{2}} - x}{2}, \quad r = \frac{y}{2s}; \quad (7)$$

83 *
84 *
85 *
86 *
87 * EXCEPTION:
88 *
89 * One may use the polar coordinate of a complex number to justify the
90 * following exception cases:
91 *
92 * EXCEPTION CASES (conform to ISO/IEC 9899:1999(E)):
93 *  $\text{csqrt}(\pm 0 + i 0) = 0 + i 0$ 
94 *  $\text{csqrt}(x + i \text{inf}) = \text{inf} + i \text{inf}$  for all x (including NaN)
95 *  $\text{csqrt}(x + i \text{NaN}) = \text{NaN} + i \text{NaN}$  with invalid for finite x
96 *  $\text{csqrt}(-\text{inf} + iy) = 0 + i \text{inf}$  for finite positive-signed y
97 *  $\text{csqrt}(\text{inf} + iy) = \text{inf} + i 0$  for finite positive-signed y
98 *  $\text{csqrt}(-\text{inf} + i \text{NaN}) = \text{NaN} + -i \text{inf}$ 
99 *  $\text{csqrt}(\text{inf} + i \text{NaN}) = \text{inf} + i \text{NaN}$ 
100 *  $\text{csqrt}(\text{NaN} + i y) = \text{NaN} + i \text{NaN}$  for finite y
101 *  $\text{csqrt}(\text{NaN} + i \text{NaN}) = \text{NaN} + i \text{NaN}$ 
102 */
103 /* INDENT ON */
104
105 #include "libm.h"
106 #include "complex_wrapper.h"
107
108 /* INDENT OFF */
109 static const double
110 two300 = 2.03703597633448608627e+90,
111 twom300 = 4.90909346529772655310e-91,
112 two599 = 2.07475778444049647926e+180,
113 twom601 = 1.20495993255144205887e-181,
114 two = 2.0,
115 zero = 0.0,
116 half = 0.5;
117 /* INDENT ON */
118
119 dcomplex
120 csqrt(dcomplex z) {
121     dcomplex ans;
122     double x, y, t, ax, ay;
123     int n, ix, iy, hx, hy, lx, ly;
```

```

125     x = D_RE(z);
126     y = D_IM(z);
127     hx = HI_WORD(x);
128     lx = LO_WORD(x);
129     hy = HI_WORD(y);
130     ly = LO_WORD(y);
131     ix = hx & 0x7fffffff;
132     iy = hy & 0x7fffffff;
133     ay = fabs(y);
134     ax = fabs(x);
135     if (ix >= 0x7ff00000 || iy >= 0x7ff00000) {
136         /* x or y is Inf or NaN */
137         if (ISINF(iy, ly))
138             D_IM(ans) = D_RE(ans) = ay;
139         else if (ISINF(ix, lx)) {
140             if (hx > 0) {
141                 D_RE(ans) = ax;
142                 D_IM(ans) = ay * zero;
143             } else {
144                 D_RE(ans) = ay * zero;
145                 D_IM(ans) = ax;
146             }
147         } else
148             D_IM(ans) = D_RE(ans) = ax + ay;
149     } else if ((iy | ly) == 0) { /* y = 0 */
150         if (hx >= 0) {
151             D_RE(ans) = sqrt(ax);
152             D_IM(ans) = zero;
153         } else {
154             D_IM(ans) = sqrt(ax);
155             D_RE(ans) = zero;
156         }
157     } else if (ix >= iy) {
158         n = (ix - iy) >> 20;
159         if (n >= 30) { /* x >> y or y=0 */
160             t = sqrt(ax);
161         } else if (ix >= 0x5f300000) { /* x > 2**500 */
162             ax *= twom601;
163             y *= twom601;
164             t = two300 * sqrt(ax + sqrt(ax * ax + y * y));
165         } else if (iy < 0x20b00000) { /* y < 2**500 */
166             ax *= two599;
167             y *= two599;
168             t = twom300 * sqrt(ax + sqrt(ax * ax + y * y));
169         } else
170             t = sqrt(half * (ax + sqrt(ax * ax + ay * ay)));
171         if (hx >= 0) {
172             D_RE(ans) = t;
173             D_IM(ans) = ay / (t + t);
174         } else {
175             D_IM(ans) = t;
176             D_RE(ans) = ay / (t + t);
177         }
178     } else {
179         n = (iy - ix) >> 20;
180         if (n >= 30) { /* y >> x */
181             if (n >= 60)
182                 t = sqrt(half * ay);
183             else if (iy >= 0x7fe00000)
184                 t = sqrt(half * ay + half * ax);
185             else if (ix <= 0x00100000)
186                 t = half * sqrt(two * (ay + ax));
187             else
188                 t = sqrt(half * (ay + ax));
189         } else if (iy >= 0x5f300000) { /* y > 2**500 */

```

```

190             ax *= twom601;
191             y *= twom601;
192             t = two300 * sqrt(ax + sqrt(ax * ax + y * y));
193         } else if (ix < 0x20b00000) { /* x < 2**500 */
194             ax *= two599;
195             y *= two599;
196             t = twom300 * sqrt(ax + sqrt(ax * ax + y * y));
197         } else
198             t = sqrt(half * (ax + sqrt(ax * ax + ay * ay)));
199         if (hx >= 0) {
200             D_RE(ans) = t;
201             D_IM(ans) = ay / (t + t);
202         } else {
203             D_IM(ans) = t;
204             D_RE(ans) = ay / (t + t);
205         }
206     }
207     if (hy < 0)
208         D_IM(ans) = -D_IM(ans);
209     return (ans);
210 }
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/complex/csqrtf.c

1

```
*****
2221 Tue Nov 25 12:58:29 2014
new/usr/src/lib/libm/common/complex/csqrtf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __csqrtf = csqrtf
30 #pragma weak csqrtf = __csqrtf
32 #include "libm.h"          /* sqrt/fabsf/sqrtf */
33 #include "complex_wrapper.h"
35 /* INDENT OFF */
36 static const float zero = 0.0F;
37 /* INDENT ON */
39 fcomplex
40 csqrtf(fcomplex z) {
41     fcomplex ans;
42     double dt, dx, dy;
43     float x, y, t, ax, ay, w;
44     int ix, iy, hx, hy;
46     x = F_RE(z);
47     y = F_IM(z);
48     hx = THE_WORD(x);
49     hy = THE_WORD(y);
50     ix = hx & 0x7fffffff;
51     iy = hy & 0x7fffffff;
52     ay = fabsf(y);
53     ax = fabsf(x);
54     if (ix >= 0x7f800000 || iy >= 0x7f800000) {
55         /* x or y is Inf or NaN */
56         if (iy == 0x7f800000)
57             F_IM(ans) = F_RE(ans) = ay;
```

new/usr/src/lib/libm/common/complex/csqrtf.c

2

```
58         else if (ix == 0x7f800000) {
59             if (hx > 0) {
60                 F_RE(ans) = ax;
61                 F_IM(ans) = ay * zero;
62             } else {
63                 F_RE(ans) = ay * zero;
64                 F_IM(ans) = ax;
65             }
66         } else
67             F_IM(ans) = F_RE(ans) = ax + ay;
68     } else if (iy == 0) {
69         if (hx >= 0) {
70             F_RE(ans) = sqrtf(ax);
71             F_IM(ans) = zero;
72         } else {
73             F_IM(ans) = sqrtf(ax);
74             F_RE(ans) = zero;
75         }
76     } else {
77         dx = (double) ax;
78         dy = (double) ay;
79         dt = sqrt(0.5 * (sqrt(dx * dx + dy * dy) + dx));
80         t = (float) dt;
81         w = (float) (dy / (dt + dt));
82         if (hx >= 0) {
83             F_RE(ans) = t;
84             F_IM(ans) = w;
85         } else {
86             F_IM(ans) = t;
87             F_RE(ans) = w;
88         }
89     }
90     if (hy < 0)
91         F_IM(ans) = -F_IM(ans);
92     return (ans);
93 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/complex/csqrtl.c

1

```
*****
3769 Tue Nov 25 12:58:30 2014
new/usr/src/lib/libm/common/complex/csqrtl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __csqrtl = csqrtl
30 #pragma weak csqrtl = __csqrtl
31
32 #include "libm.h"          /* fabsl/isinfl/sqrtl */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"
35
36 /* INDENT OFF */
37 static const long double
38     twom9001 = 2.6854002716003034957421765100615693043656e-2710L,
39     twom4500 = 2.3174987687592429423263242862381544149252e-1355L,
40     two8999 = 9.3095991180122343502582347372163290310934e+2708L,
41     two4500 = 4.314996898727097428377803545571722250806e+1354L,
42     zero = 0.0L,
43     half = 0.5L,
44     two = 2.0L;
45 /* INDENT ON */
46
47 ldcomplex
48 csqrtl(ldcomplex z) {
49     ldcomplex ans;
50     long double x, y, t, ax, ay;
51     int n, ix, iy, hx, hy;
52
53     x = LD_RE(z);
54     y = LD_IM(z);
55     hx = HI_XWORD(x);
56     hy = HI_XWORD(y);
57     ix = hx & 0x7fffffff;
```

new/usr/src/lib/libm/common/complex/csqrtl.c

2

```
58     iy = hy & 0x7fffffff;
59     ay = fabsl(y);
60     ax = fabsl(x);
61     if (ix >= 0x7fff0000 || iy >= 0x7fff0000) {
62         /* x or y is Inf or NaN */
63         if (isinfl(y))
64             LD_IM(ans) = LD_RE(ans) = ay;
65         else if (isinfl(x)) {
66             if (hx > 0) {
67                 LD_RE(ans) = ax;
68                 LD_IM(ans) = ay * zero;
69             } else {
70                 LD_RE(ans) = ay * zero;
71                 LD_IM(ans) = ax;
72             }
73         } else
74             LD_IM(ans) = LD_RE(ans) = ax + ay;
75     } else if (y == zero) {
76         if (hx >= 0) {
77             LD_RE(ans) = sqrtl(ax);
78             LD_IM(ans) = zero;
79         } else {
80             LD_IM(ans) = sqrtl(ax);
81             LD_RE(ans) = zero;
82         }
83     } else if (ix >= iy) {
84         n = (ix - iy) >> 16;
85 #if defined(__x86) /* 64 significant bits */
86         if (n >= 35)
87 #else /* 113 significant bits */
88         if (n >= 60)
89 #endif
90             t = sqrtl(ax);
91     } else if (ix >= 0x5f3f0000) { /* x > 2**8000 */
92         ax *= twom9001;
93         y *= twom9001;
94         t = two4500 * sqrtl(ax + sqrtl(ax * ax + y * y));
95     } else if (iy <= 0x20bf0000) { /* y < 2**-8000 */
96         ax *= two8999;
97         y *= two8999;
98         t = twom4500 * sqrtl(ax + sqrtl(ax * ax + y * y));
99     } else
100         t = sqrtl(half * (ax + sqrtl(ax * ax + y * y)));
101
102     if (hx >= 0) {
103         LD_RE(ans) = t;
104         LD_IM(ans) = ay / (t + t);
105     } else {
106         LD_IM(ans) = t;
107         LD_RE(ans) = ay / (t + t);
108     }
109 } else {
110     n = (iy - ix) >> 16;
111 #if defined(__x86) /* 64 significant bits */
112     if (n >= 35) { /* } */
113 #else /* 113 significant bits */
114     if (n >= 60) {
115 #endif
116         if (n >= 120)
117             t = sqrtl(half * ay);
118         else if (iy >= 0x7ffe0000)
119             t = sqrtl(half * ay + half * ax);
120         else if (ix <= 0x00100000)
121             t = half * (sqrtl(two * (ax + ay)));
122         else
123             t = sqrtl(half * (ax + ay));
```

```
124         } else if (iy >= 0x5f3f0000) { /* y > 2**8000 */
125             ax *= twom9001;
126             y  *= twom9001;
127             t = two4500 * sqrtl(ax + sqrtl(ax * ax + y * y));
128         } else if (ix <= 0x20bf0000) {
129             ax *= two8999;
130             y  *= two8999;
131             t = twom4500 * sqrtl(ax + sqrtl(ax * ax + y * y));
132         } else
133             t = sqrtl(half * (ax + sqrtl(ax * ax + y * y)));
134
135         if (hx >= 0) {
136             LD_RE(ans) = t;
137             LD_IM(ans) = ay / (t + t);
138         } else {
139             LD_IM(ans) = t;
140             LD_RE(ans) = ay / (t + t);
141         }
142     }
143     if (hy < 0)
144         LD_IM(ans) = -LD_IM(ans);
145     return (ans);
146 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/complex/ctan.c

1

```
*****
1558 Tue Nov 25 12:58:30 2014
new/usr/src/lib/libm/common/complex/ctan.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __ctan = ctan
30 #pragma weak ctan = __ctan
31
32 /* INDENT OFF */
33 /*
34  * dcomplex ctan(dcomplex z);
35  *
36  * If z = x+iy, then since ctan(iz) = i*ctanh(z), we have
37  *
38  * ctan(z)      = ctan((-1)*(-z)) = ctan(i*i*(-z))
39  *              = i*ctanh(i*(-z)) = i*ctanh(i*(-x-yi))
40  *              = i*ctanh(y-ix)
41  *              = -Im(ctanh(y-ix))+i*Re(ctanh(y-ix))
42  */
43 /* INDENT ON */
44
45 #include "libm.h"
46 #include "complex_wrapper.h"
47
48 dcomplex
49 ctan(dcomplex z) {
50     double x, y;
51     dcomplex ans, ct;
52
53     x = D_RE(z);
54     y = D_IM(z);
55     D_RE(z) = y;
56     D_IM(z) = -x;
57     ct = ctanh(z);
```

new/usr/src/lib/libm/common/complex/ctan.c

2

```
58     D_RE(ans) = -D_IM(ct);
59     D_IM(ans) = D_RE(ct);
60     return (ans);
61 }
_____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/complex/ctanf.c

1

1275 Tue Nov 25 12:58:30 2014

new/usr/src/lib/libm/common/complex/ctanf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ctanf = ctanf
30 #pragma weak ctanf = __ctanf
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 fcomplex
36 ctanf(fcomplex z) {
37     float x, y;
38     fcomplex ans, ct;
39
40     x = F_RE(z);
41     y = F_IM(z);
42     F_RE(z) = y;
43     F_IM(z) = -x;
44     ct = ctanhf(z);
45     F_RE(ans) = -F_IM(ct);
46     F_IM(ans) = F_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

```

*****
5411 Tue Nov 25 12:58:31 2014
new/usr/src/lib/libm/common/complex/ctanh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __ctanh = ctanh
30 #pragma weak ctanh = __ctanh
32 /* INDENT OFF */
33 /*
34  * dcomplex ctanh(dcomplex z);
35  *
36  *          tanh x + i tan y      sinh 2x + i sin 2y
37  * ctanh z = ----- = -----
38  *          1 + i tanh(x)tan(y)      cosh 2x + cos 2y
39  *
40  * For |x| >= prec/2 (14,28,34,60 for single, double, double extended, quad),
41  * we use
42  *
43  *          1      2x      2 sin 2y
44  * cosh 2x = sinh 2x = --- e      and hence ctanh z = 1 + i -----
45  *          2
46  *
47  *
48  * otherwise, to avoid cancellation, for |x| < prec/2,
49  *
50  *          2x      2      2      2
51  *          (e - 1)      2      2
52  * cosh 2x + cos 2y = 1 + ----- + cos y - sin y
53  *          2 e
54  *
55  *          1      2x      2      -2x      2
56  *          = --- (e - 1) e      + 2 cos y
57  *

```

```

58 * and
59 *
60 *          [ 2x      2x ]
61 *          [ e - 1 + ----- ]
62 * sinh 2x = --- [ e - 1 + ----- ]
63 *          2 [ e - 1 + ----- ]
64 *          [ e ]
65 *
66 * Implementation notes: let t = expml(2x) = e2x - 1, then
67 *
68 *          1 [ t*t      2 ]      1 [ t ]
69 * cosh 2x + cos 2y = --- * [ ----- + 4 cos y ]; sinh 2x = --- * [ t + --- ]
70 *          2 [ t+1 ]      2 [ t+1 ]
71 *
72 * Hence,
73 *
74 *
75 *          t*t+2t      [4(t+1)(cos y)]*(sin y)
76 * ctanh z = ----- + i -----
77 *          t*t+[4(t+1)(cos y)](cos y)      t*t+[4(t+1)(cos y)](cos y)
78 *
79 * EXCEPTION (conform to ISO/IEC 9899:1999(E)):
80 * ctanh(0,0)=(0,0)
81 * ctanh(x,inf) = (NaN,NaN) for finite x
82 * ctanh(x,NaN) = (NaN,NaN) for finite x
83 * ctanh(inf,y) = 1+ i*0*sin(2y) for positive-signed finite y
84 * ctanh(inf,inf) = (1, +-0)
85 * ctanh(inf,NaN) = (1, +-0)
86 * ctanh(NaN,0) = (NaN,0)
87 * ctanh(NaN,y) = (NaN,NaN) for non-zero y
88 * ctanh(NaN,NaN) = (NaN,NaN)
89 */
90 /* INDENT ON */
92 #include "libm.h" /* exp/expml/fabs/sin/tanh/sincos */
93 #include "complex_wrapper.h"
95 static const double four = 4.0, two = 2.0, one = 1.0, zero = 0.0;
97 dcomplex
98 ctanh(dcomplex z) {
99     double t, r, v, u, x, y, S, C;
100     int hx, ix, lx, hy, iy, ly;
101     dcomplex ans;
103     x = D_RE(z);
104     y = D_IM(z);
105     hx = HI_WORD(x);
106     lx = LO_WORD(x);
107     ix = hx & 0x7fffffff;
108     hy = HI_WORD(y);
109     ly = LO_WORD(y);
110     iy = hy & 0x7fffffff;
111     x = fabs(x);
112     y = fabs(y);
114     if ((iy | ly) == 0) { /* ctanh(x,0) = (x,0) for x = 0 or NaN */
115         D_RE(ans) = tanh(x);
116         D_IM(ans) = zero;
117     } else if (iy >= 0x7ff00000) { /* y is inf or NaN */
118         if (ix < 0x7ff00000) /* catanh(finite x,inf/nan) is nan */
119             D_RE(ans) = D_IM(ans) = y - y;
120         else if (((ix - 0x7ff00000) | lx) == 0) { /* x is inf */
121             D_RE(ans) = one;
122             D_IM(ans) = zero;
123         } else {

```

```

124         D_RE(ans) = x + y;
125         D_IM(ans) = y - y;
126     }
127 } else if (ix >= 0x403c0000) {
128     /*
129     * |x| > 28 = prec/2 (14,28,34,60)
130     * ctanh z ~ 1 + i (sin2y)/(exp(2x))
131     */
132     D_RE(ans) = one;
133     if (iy < 0x7fe00000) /* t = sin(2y) */
134         S = sin(y + y);
135     else {
136         (void) sincos(y, &S, &C);
137         S = (S + S) * C;
138     }
139     if (ix >= 0x7fe00000) { /* |x| > max/2 */
140         if (ix >= 0x7ff00000) { /* |x| is inf or NaN */
141             if (((ix - 0x7ff00000) | lx) != 0)
142                 D_RE(ans) = D_IM(ans) = x + y;
143                 /* x is NaN */
144             else
145                 D_IM(ans) = zero * S; /* x is inf */
146         } else
147             D_IM(ans) = S * exp(-x); /* underflow */
148     } else
149         D_IM(ans) = (S + S) * exp(-(x + x));
150                                     /* 2 sin 2y / exp(2x) */
151 } else {
152     /* INDENT OFF */
153     /*
154     *
155     *      t*t+2t
156     *      ctanh z = ----- +
157     *                  t*t+[4(t+1)(cos y)](cos y)
158     *
159     *                  [4(t+1)(cos y)]*(sin y)
160     *      i -----
161     *                  t*t+[4(t+1)(cos y)](cos y)
162     */
163     /* INDENT ON */
164     (void) sincos(y, &S, &C);
165     t = expml(x + x);
166     r = (four * C) * (t + one);
167     u = t * t;
168     v = one / (u + r * C);
169     D_RE(ans) = (u + two * t) * v;
170     D_IM(ans) = (r * S) * v;
171 }
172 if (hx < 0)
173     D_RE(ans) = -D_RE(ans);
174 if (hy < 0)
175     D_IM(ans) = -D_IM(ans);
176 return (ans);
177 }

```

unchanged_portion_omitted

```

*****
3091 Tue Nov 25 12:58:31 2014
new/usr/src/lib/libm/common/complex/ctanhf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __ctanhf = ctanhf
30 #pragma weak ctanhf = __ctanhf

32 #include "libm.h"          /* expf/expmlf/fabsf/sincosf/sinf/tanhf */
33 #include "complex_wrapper.h"

35 /* INDENT OFF */
36 static const float four = 4.0F, two = 2.0F, one = 1.0F, zero = 0.0F;
37 /* INDENT ON */

39 fcomplex
40 ctanhf(fcomplex z) {
41     float r, u, v, t, x, y, S, C;
42     int hx, ix, hy, iy;
43     fcomplex ans;

45     x = F_RE(z);
46     y = F_IM(z);
47     hx = THE_WORD(x);
48     ix = hx & 0x7fffffff;
49     hy = THE_WORD(y);
50     iy = hy & 0x7fffffff;
51     x = fabsf(x);
52     y = fabsf(y);

54     if (iy == 0) {          /* ctanh(x,0) = (x,0) for x = 0 or NaN */
55         F_RE(ans) = tanhf(x);
56         F_IM(ans) = zero;
57     } else if (iy >= 0x7f800000) { /* y is inf or NaN */

```

```

58     if (ix < 0x7f800000) /* catanh(finite x,inf/nan) is nan */
59         F_RE(ans) = F_IM(ans) = y - y;
60     else if (ix == 0x7f800000) { /* x is inf */
61         F_RE(ans) = one;
62         F_IM(ans) = zero;
63     } else {
64         F_RE(ans) = x + y;
65         F_IM(ans) = y - y;
66     }
67 } else if (ix >= 0x41600000) {
68     /*
69     * |x| > 14 = prec/2 (14,28,34,60)
70     * ctanh z ~ 1 + i (sin2y)/(exp(2x))
71     */
72     F_RE(ans) = one;
73     if (iy < 0x7f000000) { /* t = sin(2y) */
74         S = sinf(y + y);
75     } else {
76         (void) sincosf(y, &S, &C);
77         S = (S + S) * C;
78     }
79     if (ix >= 0x7f000000) { /* |x| > max/2 */
80         if (ix >= 0x7f800000) { /* |x| is inf or NaN */
81             if (ix > 0x7f800000) /* x is NaN */
82                 F_RE(ans) = F_IM(ans) = x + y;
83             else
84                 F_IM(ans) = zero * S; /* x is inf */
85         } else
86             F_IM(ans) = S * expf(-x); /* underflow */
87     } else
88         F_IM(ans) = (S + S) * expf(-(x + x)); /* 2 sin 2y / exp(2x) */
89     } else {
90         /* INDENT OFF */
91         /*
92         *
93         *          t*t+2t
94         *   ctanh z = -----
95         *          t*t+[4(t+1)(cos y)](cos y)
96         *
97         *          [4(t+1)(cos y)]*(sin y)
98         *   i -----
99         *          t*t+[4(t+1)(cos y)](cos y)
100        */
101        /* INDENT ON */
102        (void) sincosf(y, &S, &C);
103        t = expmlf(x + x);
104        r = (four * C) * (t + one);
105        u = t * t;
106        v = one / (u + r * C);
107        F_RE(ans) = (u + two * t) * v;
108        F_IM(ans) = (r * S) * v;
109    }
110    if (hx < 0)
111        F_RE(ans) = -F_RE(ans);
112    if (hy < 0)
113        F_IM(ans) = -F_IM(ans);
114    return (ans);
115 }

```

_____unchanged_portion_omitted_____

```

*****
3183 Tue Nov 25 12:58:32 2014
new/usr/src/lib/libm/common/complex/ctanh.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __ctanh = ctanh
30 #pragma weak ctanh = __ctanh

32 #include "libm.h" /* expl/expml1/fabs1/isinfl/isnanl/sincosl/sinl/tanh1 */
33 #include "complex_wrapper.h"
34 #include "longdouble.h"

36 /* INDENT OFF */
37 static const long double four = 4.0L, two = 2.0L, one = 1.0L, zero = 0.0L;
38 /* INDENT ON */

40 ldcomplex
41 ctanh(ldcomplex z) {
42     long double r, u, v, t, x, y, S, C;
43     int hx, ix, hy, iy;
44     ldcomplex ans;

46     x = LD_RE(z);
47     y = LD_IM(z);
48     hx = HI_XWORD(x);
49     ix = hx & 0x7fffffff;
50     hy = HI_XWORD(y);
51     iy = hy & 0x7fffffff;
52     x = fabs1(x);
53     y = fabs1(y);

55     if (y == zero) { /* ctanh(x,0) = (x,0) for x = 0 or NaN */
56         LD_RE(ans) = tanh1(x);
57         LD_IM(ans) = zero;

```

```

58     } else if (iy >= 0x7fff0000) { /* y is inf or NaN */
59         if (ix < 0x7fff0000) /* catanh(finite x,inf/nan) is nan */
60             LD_RE(ans) = LD_IM(ans) = y - y;
61         else if (isinfl(x)) { /* x is inf */
62             LD_RE(ans) = one;
63             LD_IM(ans) = zero;
64         } else {
65             LD_RE(ans) = x + y;
66             LD_IM(ans) = y - y;
67         }
68     } else if (ix >= 0x4004e000) {
69         /* INDENT OFF */
70         /*
71          * |x| > 60 = prec/2 (14,28,34,60)
72          * ctanh z ~ 1 + i (sin2y)/(exp(2x))
73          */
74         /* INDENT ON */
75         LD_RE(ans) = one;
76         if (iy < 0x7ffe0000) /* t = sin(2y) */
77             S = sinl(y + y);
78         else {
79             (void) sincosl(y, &S, &C);
80             S = (S + S) * C;
81         }
82         if (ix >= 0x7ffe0000) { /* |x| > max/2 */
83             if (ix >= 0x7fff0000) { /* |x| is inf or NaN */
84                 if (isnanl(x)) /* x is NaN */
85                     LD_RE(ans) = LD_IM(ans) = x + y;
86                 else
87                     LD_IM(ans) = zero * S; /* x is inf */
88             } else
89                 LD_IM(ans) = S * expl(-x); /* underflow */
90         } else
91             LD_IM(ans) = (S + S) * expl(-(x + x)); /* 2 sin 2y / exp(2x) */
92     } else {
93         /* INDENT OFF */
94         /*
95          *
96          *          t*t+2t
97          * ctanh z = -----
98          *          t*t+[4(t+1)(cos y)](cos y)
99          *
100          *          [4(t+1)(cos y)]*(sin y)
101          *          i -----
102          *          t*t+[4(t+1)(cos y)](cos y)
103          */
104         /* INDENT ON */
105         sincosl(y, &S, &C);
106         t = expml1(x + x);
107         r = (four * C) * (t + one);
108         u = t * t;
109         v = one / (u + r * C);
110         LD_RE(ans) = (u + two * t) * v;
111         LD_IM(ans) = (r * S) * v;
112     }
113     if (hx < 0)
114         LD_RE(ans) = -LD_RE(ans);
115     if (hy < 0)
116         LD_IM(ans) = -LD_IM(ans);
117     return (ans);
118 }

```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/complex/ctanl.c

1

1292 Tue Nov 25 12:58:32 2014

new/usr/src/lib/libm/common/complex/ctanl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ctanl = ctanl
30 #pragma weak ctanl = __ctanl
```

```
32 #include "libm.h"
33 #include "complex_wrapper.h"
```

```
35 ldcomplex
36 ctanl(ldcomplex z) {
37     long double x, y;
38     ldcomplex ans, ct;
39
40     x = LD_RE(z);
41     y = LD_IM(z);
42     LD_RE(z) = y;
43     LD_IM(z) = -x;
44     ct = ctanhl(z);
45     LD_RE(ans) = -LD_IM(ct);
46     LD_IM(ans) = LD_RE(ct);
47     return (ans);
48 }
```

unchanged portion omitted

new/usr/src/lib/libm/common/m9x/__fex_hdr.c

1

21395 Tue Nov 25 12:58:33 2014

new/usr/src/lib/libm/common/m9x/__fex_hdr.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

30 #include "fenv_synonyms.h"

30 #undef lint

31 #include <signal.h>

32 #include <siginfo.h>

33 #include <ucontext.h>

34 #include <stdio.h>

35 #include <stdlib.h>

36 #include <unistd.h>

37 #include <thread.h>

38 #include <math.h>

39 #if defined(__SUNPRO_C)

40 #include <sunmath.h>

41 #endif

42 #include <fenv.h>

43 #include "fex_handler.h"

44 #include "fenv_inlines.h"

46 #if defined(__sparc) && !defined(__sparcv9)

47 #include <sys/procfs.h>

48 #endif

50 /* 2.x signal.h doesn't declare sigemptyset or sigismember

51 if they're #defined (see sys/signal.h) */

new/usr/src/lib/libm/common/m9x/__fex_hdr.c

2

```
52 extern int sigemptyset(sigset_t *);
53 extern int sigismember(const sigset_t *, int);
```

55 /* external globals */

56 void (*__mt_fex_sync)() = NULL; /* for synchronization with libmtsk */

57 #pragma weak __mt_fex_sync

60 #ifdef LIBM_MT_FEX_SYNC

59 void (*__libm_mt_fex_sync)() = NULL; /* new, improved version of above */

60 #pragma weak __libm_mt_fex_sync

63 #endif

62 /* private variables */

63 static fex_handler_t main_handlers;

64 static int handlers_initialized = 0;

65 static thread_key_t handlers_key;

66 static mutex_t handlers_key_lock = DEFAULTMUTEX;

68 static struct sigaction oact = { 0, SIG_DFL };

69 static mutex_t hdr_lock = DEFAULTMUTEX;

70 static int hdr_installed = 0;

72 /* private const data */

73 static const int te_bit[FEX_NUM_EXC] = {

74 1 << fp_trap_inexact,

75 1 << fp_trap_division,

76 1 << fp_trap_underflow,

77 1 << fp_trap_overflow,

78 1 << fp_trap_invalid,

79 1 << fp_trap_invalid,

80 1 << fp_trap_invalid,

81 1 << fp_trap_invalid,

82 1 << fp_trap_invalid,

83 1 << fp_trap_invalid,

84 1 << fp_trap_invalid,

85 1 << fp_trap_invalid

86 };

unchanged portion omitted

148 #ifdef LIBM_MT_FEX_SYNC

145 /*

146 * The following function may be used for synchronization with any

147 * internal project that manages multiple threads

148 */

149 enum __libm_mt_fex_sync_actions {

150 __libm_mt_fex_start_master = 0,

151 __libm_mt_fex_start_slave,

152 __libm_mt_fex_finish_master,

153 __libm_mt_fex_finish_slave

154 };

unchanged portion omitted

207 #endif

204 #if defined(__sparc)

206 /*

207 * Code for setting or clearing interval mode on US-III and above.

208 * This is embedded as data so we don't have to mark the library

209 * as a v8plus/v9b object. (I could have just used one entry and

210 * modified the second word to set the bits I want, but that would

211 * have required another mutex.)

212 */

213 static const unsigned int siam[][2] = {

214 { 0x81c3e008, 0x81b01020 }, /* retl, siam 0 */

215 { 0x81c3e008, 0x81b01024 }, /* retl, siam 4 */

216 { 0x81c3e008, 0x81b01025 }, /* retl, siam 5 */

```
217     { 0x81c3e008, 0x81b01026 }, /* retl, siam 6 */
218     { 0x81c3e008, 0x81b01027 } /* retl, siam 7 */
219 };
___unchanged_portion_omitted_

798 /*
799 * Update the trap enable bits according to the selected modes
800 */
801 void
802 ___fex_update_te()
803 {
804     struct fex_handler_data *thr_handlers;
805     struct sigaction        act, tmpact;
806     sigset_t                blocked;
807     unsigned long          fsr;
808     int                     te;

810     /* determine which traps are needed */
811     thr_handlers = ___fex_get_thr_handlers();
812     ___fenv_getfsr(&fsr);
813     te = ___fex_te_needed(thr_handlers, fsr);

815     /* install ___fex_hdr as necessary */
816     if (!hdlr_installed && te) {
817         act.sa_handler = ___fex_hdr;
818         sigemptyset(&act.sa_mask);
819         act.sa_flags = SA_SIGINFO;
820         sigaction(SIGFPE, &act, &tmpact);
821         if (tmpact.sa_handler != ___fex_hdr)
822             {
823                 mutex_lock(&hdlr_lock);
824                 oact = tmpact;
825                 mutex_unlock(&hdlr_lock);
826             }
827         hdlr_installed = 1;
828     }

830     /* set the new trap enable bits (only if SIGFPE is not blocked) */
831     if (sigprocmask(0, NULL, &blocked) == 0 &&
832         !sigismember(&blocked, SIGFPE)) {
833         ___fenv_set_te(fsr, te);
834         ___fenv_setfsr(&fsr);
835     }

837     /* synchronize with libm_tsk */
838     ___mt_fex_sync = ___fex_sync_with_libm_tsk;

845 #ifdef LIBM_MT_FEX_SYNC
840     /* synchronize with other projects */
841     ___libm_mt_fex_sync = ___fex_sync_with_threads;
848 #endif
842 }
___unchanged_portion_omitted_
```


new/usr/src/lib/libm/common/m9x/___fex_i386.c

1

```
*****
36554 Tue Nov 25 12:58:33 2014
new/usr/src/lib/libm/common/m9x/___fex_i386.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #include "fenv_synonyms.h"
30 #include <stdio.h>
31 #include <unistd.h>
32 #include <stdlib.h>
33 #include <string.h>
34 #include <signal.h>
35 #include <siginfo.h>
36 #include <ucontext.h>
37 #include <thread.h>
38 #include <math.h>
39 #if defined(__SUNPRO_C)
40 #include <sunmath.h>
41 #endif
42 #include <fenv.h>
43 #include "fex_handler.h"
44 #include "fenv_inlines.h"
46 #if defined(__amd64)
47 #define test_sse_hw 1
48 #else
49 /*
50  * The following variable lives in libc on Solaris 10, where it
51  * gets set to a nonzero value at startup time on systems with SSE.
52  */
53 int _sse_hw = 0;
54 #pragma weak _sse_hw
55 #define test_sse_hw &_sse_hw && _sse_hw
56 #endif
```

new/usr/src/lib/libm/common/m9x/___fex_i386.c

2

```
58 static int accrued = 0;
59 static thread_key_t accrued_key;
60 static mutex_t accrued_key_lock = DEFAULTMUTEX;
62 int *
63 ___fex_accrued()
64 {
65     int *p;
67     if (thr_main())
68         return &accrued;
69     else {
70         p = NULL;
71         mutex_lock(&accrued_key_lock);
72         if (thr_getspecific(accrued_key, (void **)&p) != 0 &&
73             thr_keycreate(&accrued_key, free) != 0) {
74             mutex_unlock(&accrued_key_lock);
75             return NULL;
76         }
77         mutex_unlock(&accrued_key_lock);
78         if (!p) {
79             if ((p = (int*) malloc(sizeof(int))) == NULL)
80                 return NULL;
81             if (thr_setspecific(accrued_key, (void *)p) != 0) {
82                 (void)free(p);
83                 return NULL;
84             }
85             *p = 0;
86         }
87         return p;
88     }
89 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/__fex_sparc.c

1

```
*****
21295 Tue Nov 25 12:58:34 2014
new/usr/src/lib/libm/common/m9x/__fex_sparc.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #if defined(__sparc)
31 #include "fenv_synonyms.h"
32 #include <stdio.h>
33 #include <unistd.h>
34 #include <string.h>
35 #include <signal.h>
36 #include <siginfo.h>
37 #include <thread.h>
38 #include <ucontext.h>
39 #include <math.h>
40 #if defined(__SUNPRO_C)
41 #include <sunmath.h>
42 #endif
43 #include <fenv.h>
44 #include "fenv_inlines.h"
45 #include "libm_inlines.h"
47 #ifdef __sparcv9
49 #define FPreg(X)      &uap->uc_mcontext.fpregs.fpu_fr.fpu_regs[X]
51 #define FPREG(X)     &uap->uc_mcontext.fpregs.fpu_fr.fpu_dregs[(X>>1) | \
52                      ((X&1)<<4)]
54 #else
56 #include <sys/procfs.h>
```

new/usr/src/lib/libm/common/m9x/__fex_sparc.c

2

```
58 #define FPxreg(X)    &((prxregset_t*)uap->uc_mcontext.xrs.xrs_ptr)->pr_un.pr_
60 #define FPreg(X)     &uap->uc_mcontext.fpregs.fpu_fr.fpu_regs[X]
62 #define FPREG(X)     ((X & 1)? FPxreg(X - 1) : FPreg(X))
64 #endif /* __sparcv9 */
66 #include "fex_handler.h"
68 /* avoid dependence on libsunmath */
69 static enum fp_class_type
70 my_fp_class1(long double *a)
71 {
72     int          msw = *(int*)a & ~0x80000000;
74     if (msw >= 0x7fff0000) {
75         if ((msw & 0xffff) | *(1+(int*)a) | *(2+(int*)a) | *(3+(int*)a)
76             return fp_infinity;
77         else if (msw & 0x8000)
78             return fp_quiet;
79         else
80             return fp_signaling;
81     } else if (msw < 0x10000) {
82         if ((msw | *(1+(int*)a) | *(2+(int*)a) | *(3+(int*)a)) == 0)
83             return fp_zero;
84         else
85             return fp_subnormal;
86     } else
87         return fp_normal;
88 }
_____unchanged_portion_omitted_____
```

```

*****
39067 Tue Nov 25 12:58:34 2014
new/usr/src/lib/libm/common/m9x/___fex_sse.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */

30 #include "fenv_synonyms.h"
30 #include <ucontext.h>
31 #include <fenv.h>
32 #if defined(__SUNPRO_C)
33 #include <sunmath.h>
34 #else
35 #include <sys/ieeefp.h>
36 #endif
37 #include "fex_handler.h"
38 #include "fenv_inlines.h"

40 #if !defined(REG_PC)
41 #define REG_PC EIP
42 #endif

44 #if !defined(REG_PS)
45 #define REG_PS EFL
46 #endif

48 #ifdef __amd64
49 #define regno(X) ((X < 4)? REG_RAX - X : \
50                ((X > 4)? REG_RAX + 1 - X : REG_RSP))
51 #else
52 #define regno(X) (EAX - X)
53 #endif

55 /*
56  * Support for SSE instructions
57 */

```

```

59 /*
60  * Decode an SSE instruction. Fill in *inst and return the length of the
61  * instruction in bytes. Return 0 if the instruction is not recognized.
62 */
63 int
64 ___fex_parse_sse(ucontext_t *uap, sseinst_t *inst)
65 {
66     unsigned char *ip;
67     char *addr;
68     int i, dbl, simd, rex, modrm, sib, r;

70     i = 0;
71     ip = (unsigned char *)uap->uc_mcontext.gregs[REG_PC];

73     /* look for pseudo-prefixes */
74     dbl = 0;
75     simd = SIMD;
76     if (ip[i] == 0xF3) {
77         simd = 0;
78         i++;
79     } else if (ip[i] == 0x66) {
80         dbl = DOUBLE;
81         i++;
82     } else if (ip[i] == 0xF2) {
83         dbl = DOUBLE;
84         simd = 0;
85         i++;
86     }

88     /* look for AMD64 REX prefix */
89     rex = 0;
90     if (ip[i] >= 0x40 && ip[i] <= 0x4F) {
91         rex = ip[i];
92         i++;
93     }

95     /* parse opcode */
96     if (ip[i++] != 0x0F)
97         return 0;
98     switch (ip[i++]) {
99     case 0x2A:
100         inst->op = (int)cvtsi2ss + simd + dbl;
101         if (!simd)
102             inst->op = (int)inst->op + (rex & 8);
103         break;

105     case 0x2C:
106         inst->op = (int)cvttss2si + simd + dbl;
107         if (!simd)
108             inst->op = (int)inst->op + (rex & 8);
109         break;

111     case 0x2D:
112         inst->op = (int)cvtsi2ss + simd + dbl;
113         if (!simd)
114             inst->op = (int)inst->op + (rex & 8);
115         break;

117     case 0x2E:
118         /* oddball: scalar instruction in a SIMD opcode group */
119         if (!simd)
120             return 0;
121         inst->op = (int)ucomiss + dbl;
122         break;

```

```

124     case 0x2F:
125         /* oddball: scalar instruction in a SIMD opcode group */
126         if (!simd)
127             return 0;
128         inst->op = (int)comiss + dbl;
129         break;

131     case 0x51:
132         inst->op = (int)sqrtss + simd + dbl;
133         break;

135     case 0x58:
136         inst->op = (int)addss + simd + dbl;
137         break;

139     case 0x59:
140         inst->op = (int)mulss + simd + dbl;
141         break;

143     case 0x5A:
144         inst->op = (int)cvtss2sd + simd + dbl;
145         break;

147     case 0x5B:
148         if (dbl) {
149             if (simd)
150                 inst->op = cvtpps2dq;
151             else
152                 return 0;
153         } else {
154             inst->op = (simd)? cvtdq2ps : cvtpps2dq;
155         }
156         break;

158     case 0x5C:
159         inst->op = (int)subss + simd + dbl;
160         break;

162     case 0x5D:
163         inst->op = (int)minss + simd + dbl;
164         break;

166     case 0x5E:
167         inst->op = (int)divss + simd + dbl;
168         break;

170     case 0x5F:
171         inst->op = (int)maxss + simd + dbl;
172         break;

174     case 0xC2:
175         inst->op = (int)cmpss + simd + dbl;
176         break;

178     case 0xE6:
179         if (simd) {
180             if (dbl)
181                 inst->op = cvtttpd2dq;
182             else
183                 return 0;
184         } else {
185             inst->op = (dbl)? cvtpd2dq : cvtdq2pd;
186         }
187         break;

189     default:

```

```

190         return 0;
191     }

193     /* locate operands */
194     modrm = ip[i++];

196     if (inst->op == cvtss2si || inst->op == cvtss2si ||
197         inst->op == cvtss2si || inst->op == cvtss2si ||
198         inst->op == cvtss2siq || inst->op == cvtss2siq ||
199         inst->op == cvtss2siq || inst->op == cvtss2siq) {
200         /* op1 is a gp register */
201         r = ((rex & 4) << 1) | ((modrm >> 3) & 7);
202         inst->op1 = (sseoperand_t *)&uap->uc_mcontext.gregs[regno(r)];
203     } else if (inst->op == cvtps2pi || inst->op == cvtpps2pi ||
204         inst->op == cvtpd2pi || inst->op == cvtppd2pi) {
205         /* op1 is a mmx register */
206 #ifdef __amd64
207         inst->op1 = (sseoperand_t *)&uap->uc_mcontext.fpregs.fp_reg_set.
208             fpchip_state.st[(modrm >> 3) & 7];
209 #else
210         inst->op1 = (sseoperand_t *) (10 * ((modrm >> 3) & 7) +
211             (char *)&uap->uc_mcontext.fpregs.fp_reg_set.
212             fpchip_state.state[7]);
213 #endif
214     } else {
215         /* op1 is a xmm register */
216         r = ((rex & 4) << 1) | ((modrm >> 3) & 7);
217         inst->op1 = (sseoperand_t *)&uap->uc_mcontext.fpregs.
218             fp_reg_set.fpchip_state.xmm[r];
219     }

221     if ((modrm >> 6) == 3) {
222         if (inst->op == cvtsi2ss || inst->op == cvtsi2sd ||
223             inst->op == cvtsi2ssq || inst->op == cvtsi2sdq) {
224             /* op2 is a gp register */
225             r = ((rex & 1) << 3) | (modrm & 7);
226             inst->op2 = (sseoperand_t *)&uap->uc_mcontext.
227                 gregs[regno(r)];
228         } else if (inst->op == cvtpi2ps || inst->op == cvtppi2pd) {
229             /* op2 is a mmx register */
230 #ifdef __amd64
231             inst->op2 = (sseoperand_t *)&uap->uc_mcontext.fpregs.
232                 fp_reg_set.fpchip_state.st[modrm & 7];
233 #else
234             inst->op2 = (sseoperand_t *) (10 * (modrm & 7) +
235                 (char *)&uap->uc_mcontext.fpregs.fp_reg_set.
236                 fpchip_state.state[7]);
237 #endif
238         } else {
239             /* op2 is a xmm register */
240             r = ((rex & 1) << 3) | (modrm & 7);
241             inst->op2 = (sseoperand_t *)&uap->uc_mcontext.fpregs.
242                 fp_reg_set.fpchip_state.xmm[r];
243         }
244     } else if ((modrm & 0xc7) == 0x05) {
245 #ifdef __amd64
246         /* address of next instruction + offset */
247         r = i + 4;
248         if (inst->op == cmpss || inst->op == cmpps ||
249             inst->op == cmpsd || inst->op == cmppd)
250             r++;
251         inst->op2 = (sseoperand_t *) (ip + r + *(int *) (ip + i));
252 #else
253         /* absolute address */
254         inst->op2 = (sseoperand_t *) (*(int *) (ip + i));
255 #endif

```

```
256         i += 4;
257     } else {
258         /* complex address */
259         if ((modrm & 7) == 4) {
260             /* parse sib byte */
261             sib = ip[i++];
262             if ((sib & 7) == 5 && (modrm >> 6) == 0) {
263                 /* start with absolute address */
264                 addr = (char *) (uintptr_t) (*(int *) (ip + i));
265                 i += 4;
266             } else {
267                 /* start with base */
268                 r = ((rex & 1) << 3) | (sib & 7);
269                 addr = (char *) uap->uc_mcontext.gregs[regno(r)];
270             }
271             r = ((rex & 2) << 2) | ((sib >> 3) & 7);
272             if (r != 4) {
273                 /* add scaled index */
274                 addr += uap->uc_mcontext.gregs[regno(r)]
275                     << (sib >> 6);
276             }
277         } else {
278             r = ((rex & 1) << 3) | (modrm & 7);
279             addr = (char *) uap->uc_mcontext.gregs[regno(r)];
280         }
281
282         /* add displacement, if any */
283         if ((modrm >> 6) == 1) {
284             addr += (char) ip[i++];
285         } else if ((modrm >> 6) == 2) {
286             addr += *(int *) (ip + i);
287             i += 4;
288         }
289         inst->op2 = (sseoperand_t *) addr;
290     }
291
292     if (inst->op == cmpss || inst->op == cmpps || inst->op == cmppd ||
293         inst->op == cmppd) {
294         /* get the immediate operand */
295         inst->imm = ip[i++];
296     }
297
298     return i;
299 }
```

_____ unchanged_portion_omitted

```

*****
7672 Tue Nov 25 12:58:35 2014
new/usr/src/lib/libm/common/m9x/___fex_sym.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #include "fenv_synonyms.h"
30 #include <elf.h>
31 #include <stdio.h>
32 #include <stdlib.h>
33 #include <unistd.h>
34 #include <fcntl.h>
35 #include <procfs.h>
36 #include <string.h>
37 #include <sys/stat.h>

39 #if defined(__sparcv9) || defined(__amd64)

41 #define Elf_Ehdr      Elf64_Ehdr
42 #define Elf_Phdr      Elf64_Phdr
43 #define Elf_Shdr      Elf64_Shdr
44 #define Elf_Sym        Elf64_Sym
45 #define ELF_ST_BIND    ELF64_ST_BIND
46 #define ELF_ST_TYPE    ELF64_ST_TYPE

48 #else

50 #define Elf_Ehdr      Elf32_Ehdr
51 #define Elf_Phdr      Elf32_Phdr
52 #define Elf_Shdr      Elf32_Shdr
53 #define Elf_Sym        Elf32_Sym
54 #define ELF_ST_BIND    ELF32_ST_BIND
55 #define ELF_ST_TYPE    ELF32_ST_TYPE

57 #endif /* __sparcv9 */

```

```

59 /* semi-permanent data established by ___fex_sym_init */
60 static prmap_t      *pm = NULL;      /* prmap_t array */
61 static int          npm = 0;          /* number of entries in

63 /* transient data modified by ___fex_sym */
64 static prmap_t      *lpm = NULL;     /* prmap_t found in last call */
65 static Elf_Phdr      *ph = NULL;     /* program header array */
66 static int          phsize = 0;      /* size of ph */
67 static int          nph;             /* number of entries in
68 static char          *stbuf = NULL;   /* symbol and string table buffer */
69 static int          stbufsize = 0;   /* size of stbuf */
70 static int          stoffset;        /* offset of string tabl
71 static int          nsyms;           /* number of symbols in

73 /* get a current prmap_t list (must call this before each stack trace) */
74 void
75 ___fex_sym_init()
76 {
77     struct stat      statbuf;
78     long             n;
79     int              i;

81     /* clear out the previous prmap_t list */
82     if (pm != NULL)
83         free(pm);
84     pm = lpm = NULL;
85     npm = 0;

87     /* get the current prmap_t list */
88     if (stat("/proc/self/map", &statbuf) < 0 || statbuf.st_size <= 0 ||
89         (pm = (prmap_t*)malloc(statbuf.st_size)) == NULL)
90         return;
91     if ((i = open("/proc/self/map", O_RDONLY)) < 0)
92     {
93         free(pm);
94         pm = NULL;
95         return;
96     }
97     n = read(i, pm, statbuf.st_size);
98     close(i);
99     if (n != statbuf.st_size)
100     {
101         free(pm);
102         pm = NULL;
103     }
104     else
105         npm = (int) (n / sizeof(prmap_t));
106 }

```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/m9x/fdim.c

1

1460 Tue Nov 25 12:58:35 2014

new/usr/src/lib/libm/common/m9x/fdim.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
30 #pragma weak fdim = __fdim
32 #endif
```

```
32 /*
33  * fdim(x,y) returns x - y if x > y, +0 if x <= y, and NaN if x and
34  * y are unordered.
35  *
36  * fdim(x,y) raises overflow or inexact if x > y and x - y overflows
37  * or is inexact. It raises invalid if either operand is a signaling
38  * NaN. Otherwise, it raises no exceptions.
39 */
```

```
41 #include "libm.h" /* for islessequal macro */
```

```
43 double
44 __fdim(double x, double y) {
45     if (islessequal(x, y)) {
46         x = 0.0;
47         y = -x;
48     }
49     return (x - y);
50 }
```

_____unchanged_portion_omitted_____

new/usr/src/lib/libm/common/m9x/fdimf.c

1

1522 Tue Nov 25 12:58:36 2014

new/usr/src/lib/libm/common/m9x/fdimf.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
30 #pragma weak fdimf = __fdimf
32 #endif
```

```
32 #include "libm.h" /* for islessequal macro */
```

```
34 float
35 __fdimf(float x, float y) {
36     /*
37      * On SPARC v8plus/v9, this could be implemented as follows
38      * (assuming %f0 = x, %f1 = y, return value left in %f0):
39      *
40      * fcmps      %fcc0,%f0,%f1
41      * st         %g0,[scratch] ! use fzero instead of st/ld
42      * ld        [scratch],%f2 ! if VIS is available
43      * fnegs     %f2,%f3
44      * fmovsle   %fcc0,%f2,%f0
45      * fmovsle   %fcc0,%f3,%f1
46      * fsubs     %f0,%f1,%f0
47      */
48     if (islessequal(x, y)) {
49         x = 0.0f;
50         y = -x;
51     }
52     return (x - y);
53 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/fdiml.c

1

```
*****
1196 Tue Nov 25 12:58:37 2014
new/usr/src/lib/libm/common/m9x/fdiml.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
30 #pragma weak fdiml = __fdiml
32 #endif
```

```
32 #include "libm.h" /* for islessequal macro */
```

```
34 long double
35 __fdiml(long double x, long double y) {
36     if (islessequal(x, y)) {
37         x = 0.01;
38         y = -x;
39     }
40     return (x - y);
41 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/feexcept.c

1

```
*****
3206 Tue Nov 25 12:58:37 2014
new/usr/src/lib/libm/common/m9x/feexcept.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __feclearexcept = feclearexcept
31 #pragma weak __feraiseexcept = feraiseexcept
32 #pragma weak __fetestexcept = fetestexcept
33 #pragma weak __fegetexceptflag = fegetexceptflag
34 #pragma weak __fesetexceptflag = fesetexceptflag
35
36 #pragma weak feclearexcept96 = feclearexcept
37 #pragma weak feraiseexcept96 = feraiseexcept
38 #pragma weak fetestexcept96 = fetestexcept
39 #pragma weak fegetexceptflag96 = fegetexceptflag
40 #pragma weak fesetexceptflag96 = fesetexceptflag
41
42 #pragma weak feclearexcept = __feclearexcept
43 #pragma weak feraiseexcept = __feraiseexcept
44 #pragma weak fetestexcept = __fetestexcept
45 #pragma weak fegetexceptflag = __fegetexceptflag
46 #pragma weak fesetexceptflag = __fesetexceptflag
47
48 #include "fenv_synonyms.h"
49 #include <fenv.h>
50 #include <sys/ieee754.h>
51 #include <ucontext.h>
52 #include <thread.h>
53 #include "fex_handler.h"
```

new/usr/src/lib/libm/common/m9x/feexcept.c

2

```
47 #include "fenv_inlines.h"
48
49 int feclearexcept(int e)
50 {
51     unsigned long fsr;
52
53     __fenv_getfsr(&fsr);
54     __fenv_set_ex(fsr, __fenv_get_ex(fsr) & ~e);
55     __fenv_setfsr(&fsr);
56     if (fex_get_log())
57         __fex_update_te();
58     return 0;
59 }
60
61 _____unchanged_portion_omitted_____
```

```

*****
2813 Tue Nov 25 12:58:37 2014
new/usr/src/lib/libm/common/m9x/fenv.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __fex_merge_flags = fex_merge_flags
30 #pragma weak fex_merge_flags = __fex_merge_flags

32 #pragma weak __feholdexcept = feholdexcept
33 #pragma weak __feupdateenv = feupdateenv
34 #pragma weak __fegetenv = fegetenv
35 #pragma weak __fesetenv = fesetenv

37 #pragma weak feupdateenv96 = feupdateenv
38 #pragma weak fegetenv96 = fegetenv
39 #pragma weak fesetenv96 = fesetenv
32 #pragma weak feholdexcept = __feholdexcept
33 #pragma weak feupdateenv = __feupdateenv
34 #pragma weak fegetenv = __fegetenv
35 #pragma weak fesetenv = __fesetenv

37 #pragma weak feholdexcept96 = __feholdexcept96
38 #pragma weak feupdateenv96 = __feupdateenv96
39 #pragma weak fegetenv96 = __fegetenv96
40 #pragma weak fesetenv96 = __fesetenv96

42 #include "fenv_synonyms.h"
41 #include <fenv.h>
42 #include <ucontext.h>
43 #include <thread.h>
44 #include "fex_handler.h"
45 #include "fenv_inlines.h"

47 const fenv_t __fenv_dfl_env = {

```

```

48     {
49         { FEX_NONSTOP, (void*)() } 0 },
50         { FEX_NONSTOP, (void*)() } 0 },
51         { FEX_NONSTOP, (void*)() } 0 },
52         { FEX_NONSTOP, (void*)() } 0 },
53         { FEX_NONSTOP, (void*)() } 0 },
54         { FEX_NONSTOP, (void*)() } 0 },
55         { FEX_NONSTOP, (void*)() } 0 },
56         { FEX_NONSTOP, (void*)() } 0 },
57         { FEX_NONSTOP, (void*)() } 0 },
58         { FEX_NONSTOP, (void*)() } 0 },
59         { FEX_NONSTOP, (void*)() } 0 },
60         { FEX_NONSTOP, (void*)() } 0 },
61     },
62 #ifdef __x86
63     0x13000000
64 #else
65     0
66 #endif
67 };
_____unchanged_portion_omitted_____

```

new/usr/src/lib/libm/common/m9x/feprec.c

1

1452 Tue Nov 25 12:58:38 2014

new/usr/src/lib/libm/common/m9x/feprec.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __fegetprec = fegetprec
31 #pragma weak __fesetprec = fesetprec
30 #pragma weak fegetprec = __fegetprec
31 #pragma weak fesetprec = __fesetprec
```

```
33 #include "fenv_synonyms.h"
33 #include <fenv.h>
34 #include <ucontext.h>
35 #include <thread.h>
36 #include "fex_handler.h"
```

```
38 int fegetprec(void)
39 {
40     unsigned long fsr;

42     __fenv_getfsr(&fsr);
43     return __fenv_get_rp(fsr);
44 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/feround.c

1

1936 Tue Nov 25 12:58:38 2014

new/usr/src/lib/libm/common/m9x/feround.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __fegetround = fegetround
31 #pragma weak __fesetround = fesetround
30 #pragma weak fegetround = __fegetround
31 #pragma weak fesetround = __fesetround
```

```
33 #pragma weak fegetround96 = __fegetround
34 #pragma weak fesetround96 = __fesetround96
```

```
36 #include "fenv_synonyms.h"
35 #include <fenv.h>
36 #include <ucontext.h>
37 #include <thread.h>
38 #include "fex_handler.h"
39 #include "fenv_inlines.h"
```

```
41 #if defined(__i386) && !defined(__amd64)
42 #include <float.h>
43 #endif
```

```
45 int fegetround(void)
46 {
47     unsigned long fsr;
49     __fenv_getfsr(&fsr);
50     return (int)__fenv_get_rd(fsr);
51 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/fex_handler.c

1

2429 Tue Nov 25 12:58:39 2014

new/usr/src/lib/libm/common/m9x/fex_handler.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __fex_get_handling = fex_get_handling
31 #pragma weak __fex_set_handling = fex_set_handling
32 #pragma weak __fex_getexcepthandler = fex_getexcepthandler
33 #pragma weak __fex_setexcepthandler = fex_setexcepthandler
30 #pragma weak fex_get_handling = __fex_get_handling
31 #pragma weak fex_set_handling = __fex_set_handling
32 #pragma weak fex_getexcepthandler = __fex_getexcepthandler
33 #pragma weak fex_setexcepthandler = __fex_setexcepthandler
```

```
35 #include "fenv_synonyms.h"
35 #include <fenv.h>
36 #include <ucontext.h>
37 #include <thread.h>
38 #include "fex_handler.h"
```

```
40 int fex_get_handling(int e)
41 {
42     struct fex_handler_data *thr_handlers;
43     int i;
44
45     thr_handlers = __fex_get_thr_handlers();
46     for (i = 0; i < FEX_NUM_EXC; i++)
47         if (e & (1 << i))
48             return thr_handlers[i].__mode;
49     return FEX_NOHANDLER;
50 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/fex_log.c

1

```
*****
9349 Tue Nov 25 12:58:40 2014
new/usr/src/lib/libm/common/m9x/fex_log.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __fex_get_log = fex_get_log
31 #pragma weak __fex_set_log = fex_set_log
32 #pragma weak __fex_get_log_depth = fex_get_log_depth
33 #pragma weak __fex_set_log_depth = fex_set_log_depth
34 #pragma weak __fex_log_entry = fex_log_entry
35 #pragma weak fex_get_log = __fex_get_log
36 #pragma weak fex_set_log = __fex_set_log
37 #pragma weak fex_get_log_depth = __fex_get_log_depth
38 #pragma weak fex_set_log_depth = __fex_set_log_depth
39 #pragma weak fex_log_entry = __fex_log_entry
40
41 #include "fenv_synonyms.h"
42 #include <stdio.h>
43 #include <stdlib.h>
44 #include <unistd.h>
45 #include <string.h>
46 #include <signal.h>
47 #include <ucontext.h>
48 #include <sys/frame.h>
49 #include <fenv.h>
50 #include <sys/ieeefp.h>
51 #include <thread.h>
52 #include "fex_handler.h"
53
54 #if !defined(PC)
55 #if defined(REG_PC)
56 #define PC REG_PC
57 #else
58 #error Neither PC nor REG_PC is defined!
59 #endif
60 #endif
```

new/usr/src/lib/libm/common/m9x/fex_log.c

2

```
53 #endif
54 #endif
55
56 static FILE *log_fp = NULL;
57 static mutex_t log_lock = DEFAULTMUTEX;
58 static int log_depth = 100;
59
60 FILE *fex_get_log(void)
61 {
62     FILE *fp;
63
64     mutex_lock(&log_lock);
65     fp = log_fp;
66     mutex_unlock(&log_lock);
67     return fp;
68 }
69
70 unchanged_portion_omitted_
```

new/usr/src/lib/libm/common/m9x/fma.c

1

```
*****
11289 Tue Nov 25 12:58:40 2014
new/usr/src/lib/libm/common/m9x/fma.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak fma = __fma
32 #endif

32 #include "libm.h"
33 #include "fma.h"
34 #include "fenv_inlines.h"

36 #if defined(__sparc)

38 static const union {
39     unsigned i[2];
40     double d;
41 } C[] = {
    unchanged_portion_omitted
```



```

*****
5636 Tue Nov 25 12:58:41 2014
new/usr/src/lib/libm/common/m9x/fmaf.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak fmaf = __fmaf
32 #endif
33
34 #include "libm.h"
35 #include "fma.h"
36 #include "fenv_inlines.h"
37
38 #if defined(__sparc)
39 /*
40  * fmaf for SPARC: 32-bit single precision, big-endian
41 */
42 float
43 __fmaf(float x, float y, float z) {
44     union {
45         unsigned i[2];
46         double d;
47     } xy, zz;
48     unsigned u, s;
49     int exy, ez;
50
51     /*
52      * the following operations can only raise the invalid exception,
53      * and then only if either x*y is of the form Inf*0 or one of x,
54      * y, or z is a signaling NaN
55      */

```

```

55     xy.d = (double) x * y;
56     zz.d = (double) z;
57
58     /*
59      * if the sum xy + z will be exact, just compute it and cast the
60      * result to float
61      */
62     exy = (xy.i[0] >> 20) & 0x7fff;
63     ez = (zz.i[0] >> 20) & 0x7fff;
64     if ((ez - exy <= 4 && exy - ez <= 28) || exy == 0x7fff || exy == 0 ||
65         ez == 0x7fff || ez == 0) {
66         return ((float) (xy.d + zz.d));
67     }
68
69     /*
70      * collapse the tail of the smaller summand into a "sticky bit"
71      * so that the sum can be computed without error
72      */
73     if (ez > exy) {
74         if (ez - exy < 31) {
75             u = xy.i[1];
76             s = 2 << (ez - exy);
77             if (u & (s - 1))
78                 u |= s;
79             xy.i[1] = u & ~(s - 1);
80         } else if (ez - exy < 51) {
81             u = xy.i[0];
82             s = 1 << (ez - exy - 31);
83             if ((u & (s - 1)) | xy.i[1])
84                 u |= s;
85             xy.i[0] = u & ~(s - 1);
86             xy.i[1] = 0;
87         } else {
88             /* collapse all of xy into a single bit */
89             xy.i[0] = (xy.i[0] & 0x80000000) | ((ez - 51) << 20);
90             xy.i[1] = 0;
91         }
92     } else {
93         if (exy - ez < 31) {
94             u = zz.i[1];
95             s = 2 << (exy - ez);
96             if (u & (s - 1))
97                 u |= s;
98             zz.i[1] = u & ~(s - 1);
99         } else if (exy - ez < 51) {
100            u = zz.i[0];
101            s = 1 << (exy - ez - 31);
102            if ((u & (s - 1)) | zz.i[1])
103                u |= s;
104            zz.i[0] = u & ~(s - 1);
105            zz.i[1] = 0;
106        } else {
107            /* collapse all of zz into a single bit */
108            zz.i[0] = (zz.i[0] & 0x80000000) | ((exy - 51) << 20);
109            zz.i[1] = 0;
110        }
111    }
112
113     return ((float) (xy.d + zz.d));
114 }
115
116 unchanged_portion_omitted
117 #endif
118
119 #else
120 #error Unknown architecture
121 #endif

```

new/usr/src/lib/libm/common/m9x/fmal.c

1

28108 Tue Nov 25 12:58:41 2014

new/usr/src/lib/libm/common/m9x/fmal.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak fmal = __fmal
32 #endif
33
34 #include "libm.h"
35 #include "fma.h"
36 #include "fenv_inlines.h"
37
38 #if defined(__sparc)
39
40 static const union {
41     unsigned i[2];
42     double d;
43 } C[] = {
44     unchanged_portion_omitted
45 }
```

new/usr/src/lib/libm/common/m9x/fmax.c

1

1982 Tue Nov 25 12:58:41 2014
new/usr/src/lib/libm/common/m9x/fmax.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak fmax = __fmax
32 #endif
33
34 /*
35  * fmax(x,y) returns the larger of x and y. If just one of the
36  * arguments is NaN, fmax returns the other argument. If both
37  * arguments are NaN, fmax returns NaN.
38  * See fmaxf.c for a discussion of implementation trade-offs.
39 */
40
41 #include "libm.h" /* for isgreaterequal macro */
42 #include <fenv.h>
43
44 double
45 __fmax(double x, double y) {
46     union {
47         unsigned i[2];
48         double d;
49     } xx, yy;
50     unsigned s;
51
52     /* if y is nan, replace it by x */
53     if (y != y)
54         y = x;
```

new/usr/src/lib/libm/common/m9x/fmax.c

2

```
55     /* if x is nan, replace it by y */
56     if (x != x)
57         x = y;
58
59     /* At this point, x and y are either both numeric, or both NaN */
60     if (!isnan(x) && !isgreaterequal(x, y))
61         x = y;
62
63     /*
64      * clear the sign of the result if either x or y has its sign clear
65      */
66     xx.d = x;
67     yy.d = y;
68 #if defined(__sparc)
69     s = ~(xx.i[0] & yy.i[0]) & 0x80000000;
70     xx.i[0] &= ~s;
71 #elif defined(__x86)
72     s = ~(xx.i[1] & yy.i[1]) & 0x80000000;
73     xx.i[1] &= ~s;
74 #else
75 #error Unknown architecture
76 #endif
77
78     return (xx.d);
79 }
80
81 unchanged_portion_omitted
```

```

*****
4298 Tue Nov 25 12:58:42 2014
new/usr/src/lib/libm/common/m9x/fmaxf.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak fmaxf = __fmaxf
32 #endif
33
34 /*
35  * fmax(x,y) returns the larger of x and y. If just one of the
36  * arguments is NaN, fmax returns the other argument. If both
37  * arguments are NaN, fmax returns NaN (ideally, one of the
38  * argument NaNs).
39  *
40  * C99 does not require that fmax(-0,+0) = fmax(+0,-0) = +0, but
41  * ideally fmax should satisfy this.
42  *
43  * C99 makes no mention of exceptions for fmax. I suppose ideally
44  * either fmax never raises any exceptions or else it raises the
45  * invalid operation exception if and only if some argument is a
46  * signaling NaN. In the former case, fmax should always return
47  * one of its arguments. In the latter, fmax shouldn't return a
48  * signaling NaN, although when both arguments are signaling NaNs,
49  * this ideal is at odds with the stipulation that fmax should
50  * always return one of its arguments.
51  *
52  * Commutativity of fmax follows from the properties listed above
53  * except when both arguments are NaN. In that case, fmax may be
54  * declared commutative by fiat because there is no portable way
55  * to tell different NaNs apart. Ideally fmax would be truly com-
56  * mutative for all arguments.

```

```

55 *
56 * On SPARC V8, fmax must involve tests and branches. Ideally,
57 * an implementation on SPARC V9 should avoid branching, using
58 * conditional moves instead where necessary, and be as efficient
59 * as possible in its use of other resources.
60 *
61 * It appears to be impossible to attain all of the aforementioned
62 * ideals simultaneously. The implementation below satisfies the
63 * following (on SPARC):
64 *
65 * 1. fmax(x,y) returns the larger of x and y if neither x nor y
66 * is NaN and the non-NaN argument if just one of x or y is NaN.
67 * If both x and y are NaN, fmax(x,y) returns x unchanged.
68 * 2. fmax(-0,+0) = fmax(+0,-0) = +0.
69 * 3. If either argument is a signaling NaN, fmax raises the invalid
70 * operation exception. Otherwise, it raises no exceptions.
71 */
72
73 #include "libm.h" /* for isgreaterequal macro */
74
75 float
76 __fmaxf(float x, float y) {
77     /*
78      * On SPARC v8plus/v9, this could be implemented as follows
79      * (assuming %f0 = x, %f1 = y, return value left in %f0):
80      *
81      * fcmps    %fcc0,%f1,%f1
82      * fmovsu   %fcc0,%f0,%f1
83      * fcmps    %fcc0,%f0,%f1
84      * fmovsul   %fcc0,%f1,%f0
85      * st       %f0,[x]
86      * st       %f1,[y]
87      * ld       [x],%l0
88      * ld       [y],%l1
89      * and      %l0,%l1,%l2
90      * sethi    %hi(0x80000000),%l3
91      * andn    %l3,%l2,%l2
92      * andn    %l0,%l2,%l0
93      * st       %l0,[x]
94      * ld       [x],%f0
95      *
96      * If VIS instructions are available, use this code instead:
97      *
98      * fcmps    %fcc0,%f1,%f1
99      * fmovsu   %fcc0,%f0,%f1
100     * fcmps    %fcc0,%f0,%f1
101     * fmovsul   %fcc0,%f1,%f0
102     * fands    %f0,%f1,%f2
103     * fzeros   %f3
104     * fnegs    %f3,%f3
105     * fandnot2s %f3,%f2,%f2
106     * fandnot2s %f0,%f2,%f0
107     *
108     * If VIS 3.0 instructions are available, use this:
109     *
110     * flcmps   %fcc0,%f0,%f1
111     * fmovslg   %fcc0,%f1,%f0 ! move if %fcc0 is 1 or 2
112     */
113
114     union {
115         unsigned i;
116         float f;
117     } xx, yy;
118     unsigned s;
119
120     /* if y is nan, replace it by x */

```

```
121     if (y != y)
122         y = x;

124     /* if x is nan, replace it by y */
125     if (x != x)
126         x = y;

128     /* At this point, x and y are either both numeric, or both NaN */
129     if (!isnan(x) && !isgreater(x, y))
130         x = y;

132     /*
133      * clear the sign of the result if either x or y has its sign clear
134      */
135     xx.f = x;
136     yy.f = y;
137     s = ~(xx.i & yy.i) & 0x80000000;
138     xx.i ^= s;

140     return (xx.f);
141 }
_____unchanged_portion_omitted_____
```

```

*****
1847 Tue Nov 25 12:58:42 2014
new/usr/src/lib/libm/common/m9x/fmaxl.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak fmaxl = __fmaxl
32 #endif

32 #include "libm.h"      /* for isgreaterEqual macro */

34 long double
35 __fmaxl(long double x, long double y) {
36     union {
37 #if defined(__sparc)
38         unsigned i[4];
39 #elif defined(__x86)
40         unsigned i[3];
41 #else
42 #error Unknown architecture
43 #endif
44         long double ld;
45     } xx, yy;
46     unsigned s;

48     /* if y is nan, replace it by x */
49     if (y != y)
50         y = x;

52     /* if x is nan, replace it by y */
53     if (x != x)
54         x = y;

```

```

56     /* At this point, x and y are either both numeric, or both NaN */
57     if (!isnan(x) && !isgreaterEqual(x, y))
58         x = y;

60     /*
61      * clear the sign of the result if either x or y has its sign clear
62      */
63     xx.ld = x;
64     yy.ld = y;
65 #if defined(__sparc)
66     s = ~(xx.i[0] & yy.i[0]) & 0x80000000;
67     xx.i[0] &= ~s;
68 #elif defined(__x86)
69     s = ~(xx.i[2] & yy.i[2]) & 0x8000;
70     xx.i[2] &= ~s;
71 #else
72 #error Unknown architecture
73 #endif

75     return (xx.ld);
76 }
_____ unchanged portion omitted

```

new/usr/src/lib/libm/common/m9x/fmin.c

1

```
*****
1979 Tue Nov 25 12:58:43 2014
new/usr/src/lib/libm/common/m9x/fmin.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak fmin = __fmin
32 #endif

32 /*
33  * fmin(x,y) returns the smaller of x and y. If just one of the
34  * arguments is NaN, fmin returns the other argument. If both
35  * arguments are NaN, fmin returns NaN.
36  *
37  * See fmaxf.c for a discussion of implementation trade-offs.
38 */

40 #include "libm.h" /* for islessequal macro */

42 #include "fenv_inlines.h"
43 #include <stdio.h>
44 #include <sys/isa_defs.h>

46 double
47 __fmin(double x, double y) {
48     union {
49         unsigned i[2];
50         double d;
51     } xx, yy;
52     unsigned s;
53
54     /* if y is nan, replace it by x */
```

new/usr/src/lib/libm/common/m9x/fmin.c

2

```
55     if (y != y)
56         y = x;

58     /* if x is nan, replace it by y */
59     if (x != x)
60         x = y;

62     /* At this point, x and y are either both numeric, or both NaN */
63     if (!isnan(x) && !islessequal(x, y))
64         x = y;

66     /*
67      * set the sign of the result if either x or y has its sign set
68      */
69     xx.d = x;
70     yy.d = y;
71 #if defined(_BIG_ENDIAN)
72     s = (xx.i[0] | yy.i[0]) & 0x80000000;
73     xx.i[0] |= s;
74 #else
75     s = (xx.i[1] | yy.i[1]) & 0x80000000;
76     xx.i[1] |= s;
77 #endif

79     return (xx.d);
80 }
_____unchanged_portion_omitted_
```

```

*****
2386 Tue Nov 25 12:58:43 2014
new/usr/src/lib/libm/common/m9x/fminf.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak fminf = __fminf
32 #endif
33
34 #include "libm.h" /* for islessequal macro */
35
36 float
37 __fminf(float x, float y) {
38     /*
39      * On SPARC v8plus/v9, this could be implemented as follows
40      * (assuming %f0 = x, %f1 = y, return value left in %f0):
41      *
42      * fcmps    %fcc0,%f1,%f1
43      * fmovsu   %fcc0,%f0,%f1
44      * fcmps    %fcc0,%f0,%f1
45      * fmovsug  %fcc0,%f1,%f0
46      * st       %f0,[x]
47      * st       %f1,[y]
48      * ld       [x],%l0
49      * ld       [y],%l1
50      * or       %l0,%l1,%l2
51      * sethi   %hi(0x80000000),%l3
52      * and     %l3,%l2,%l2
53      * or     %l0,%l2,%l0
54      * st     %l0,[x]
55      * ld     [x],%f0
56     */

```

```

55     * If VIS instructions are available, use this code instead:
56     *
57     * fcmps    %fcc0,%f1,%f1
58     * fmovsu   %fcc0,%f0,%f1
59     * fcmps    %fcc0,%f0,%f1
60     * fmovsug  %fcc0,%f1,%f0
61     * fors     %f0,%f1,%f2
62     * fzeros   %f3
63     * fnegs    %f3,%f3
64     * fands    %f3,%f2,%f2
65     * fors     %f0,%f2,%f0
66     *
67     * If VIS 3.0 instructions are available, use this:
68     *
69     * flcmps    %fcc0,%f0,%f1
70     * fmovsge   %fcc0,%f1,%f0 ! move if %fcc0 is 0 or 2
71     */
72
73     union {
74         unsigned i;
75         float f;
76     } xx, yy;
77     unsigned s;
78
79     /* if y is nan, replace it by x */
80     if (y != y)
81         y = x;
82
83     /* if x is nan, replace it by y */
84     if (x != x)
85         x = y;
86
87     /* At this point, x and y are either both numeric, or both NaN */
88     if (!isnan(x) && !islessequal(x, y))
89         x = y;
90
91     /*
92      * set the sign of the result if either x or y has its sign set
93      */
94     xx.f = x;
95     yy.f = y;
96     s = (xx.i | yy.i) & 0x80000000;
97     xx.i |= s;
98
99     return (xx.f);
100 }

```

unchanged portion omitted


```

*****
1833 Tue Nov 25 12:58:44 2014
new/usr/src/lib/libm/common/m9x/fminl.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak fminl = __fminl
32 #endif

32 #include "libm.h"      /* for islessequal macro */

34 long double
35 __fminl(long double x, long double y) {
36     union {
37 #if defined(__sparc)
38         unsigned i[4];
39 #elif defined(__x86)
40         unsigned i[3];
41 #else
42 #error Unknown architecture
43 #endif
44         long double ld;
45     } xx, yy;
46     unsigned s;

48     /* if y is nan, replace it by x */
49     if (y != y)
50         y = x;

52     /* if x is nan, replace it by y */
53     if (x != x)
54         x = y;

```

```

56     /* At this point, x and y are either both numeric, or both NaN */
57     if (!isnan(x) && !islessequal(x, y))
58         x = y;

60     /*
61      * set the sign of the result if either x or y has its sign set
62      */
63     xx.ld = x;
64     yy.ld = y;
65 #if defined(__sparc)
66     s = (xx.i[0] | yy.i[0]) & 0x80000000;
67     xx.i[0] |= s;
68 #elif defined(__x86)
69     s = (xx.i[2] | yy.i[2]) & 0x8000;
70     xx.i[2] |= s;
71 #else
72 #error Unknown architecture
73 #endif

75     return (xx.ld);
76 }
_____ unchanged portion omitted

```

```

*****
2799 Tue Nov 25 12:58:44 2014
new/usr/src/lib/libm/common/m9x/frexp.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak frexp = __frexp
32 #endif
33
34 /*
35  * frexp(x, exp) returns the normalized significand of x and sets
36  * *exp so that x = r*2^(*exp) where r is the return value. If x
37  * is finite and nonzero, 1/2 <= |r| < 1.
38  *
39  * If x is zero, infinite or NaN, frexp returns x and sets *exp = 0.
40  * (The relevant standards do not specify *exp when x is infinite or
41  * NaN, but this code sets it anyway.)
42  *
43  * If x is a signaling NaN, this code returns x without attempting
44  * to raise the invalid operation exception. If x is subnormal,
45  * this code treats it as nonzero regardless of nonstandard mode.
46  */
47
48 #include "libm.h"
49
50 double
51 __frexp(double x, int *exp) {
52     union {
53         unsigned i[2];
54         double d;
55     } xx, yy;
56     double t;

```

```

55     unsigned hx;
56     int e;
57
58     xx.d = x;
59     hx = xx.i[HIWORD] & ~0x80000000;
60
61     if (hx >= 0x7ff00000) { /* x is infinite or NaN */
62         *exp = 0;
63         return (x);
64     }
65
66     e = 0;
67     if (hx < 0x00100000) { /* x is subnormal or zero */
68         if ((hx | xx.i[LOWORD]) == 0) {
69             *exp = 0;
70             return (x);
71         }
72
73         /*
74          * normalize x by regarding it as an integer
75          *
76          * Here we use 32-bit integer arithmetic to avoid trapping
77          * or emulating 64-bit arithmetic. If 64-bit arithmetic is
78          * available (e.g., in SPARC V9), do this instead:
79          *
80          * long lx = ((long) hx << 32) | xx.i[LOWORD];
81          * xx.d = (xx.i[HIWORD] < 0)? -lx : lx;
82          *
83          * If subnormal arithmetic doesn't trap, just multiply x by
84          * a power of two.
85          */
86         yy.i[HIWORD] = 0x43300000 | hx;
87         yy.i[LOWORD] = xx.i[LOWORD];
88         t = yy.d;
89         yy.i[HIWORD] = 0x43300000;
90         yy.i[LOWORD] = 0;
91         t -= yy.d; /* t = |x| scaled */
92         xx.d = ((int)xx.i[HIWORD] < 0)? -t : t;
93         hx = xx.i[HIWORD] & ~0x80000000;
94         e = -1074;
95     }
96
97     /* now xx.d is normal */
98     xx.i[HIWORD] = (xx.i[HIWORD] & ~0x7ff00000) | 0x3fe00000;
99     *exp = e + (hx >> 20) - 0x3fe;
100     return (xx.d);
101 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/frexp.c

1

```
*****
1638 Tue Nov 25 12:58:45 2014
new/usr/src/lib/libm/common/m9x/frexp.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak frexp = __frexp
32 #endif

32 #include "libm.h"

34 float
35 __frexp(float x, int *exp) {
36     union {
37         unsigned i;
38         float f;
39     } xx;
40     unsigned hx;
41     int e;

43     xx.f = x;
44     hx = xx.i & ~0x80000000;

46     if (hx >= 0x7f800000) { /* x is infinite or NaN */
47         *exp = 0;
48         return (x);
49     }

51     e = 0;
52     if (hx < 0x00800000) { /* x is subnormal or zero */
53         if (hx == 0) {
54             *exp = 0;
```

new/usr/src/lib/libm/common/m9x/frexp.c

2

```
55         return (x);
56     }

58     /* normalize x by regarding it as an integer */
59     xx.f = (int) xx.i < 0 ? -(int) hx : (int) hx;
60     hx = xx.i & ~0x80000000;
61     e = -149;
62 }

64 /* now xx.f is normal */
65 xx.i = (xx.i & ~0x7f800000) | 0x3f000000;
66 *exp = e + (hx >> 23) - 0x7e;
67 return (xx.f);
68 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/frexpl.c

1

```
*****
2623 Tue Nov 25 12:58:45 2014
new/usr/src/lib/libm/common/m9x/frexpl.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak frexpl = __frexpl
32 #endif

32 #include "libm.h"

34 #if defined(__sparc)

36 long double
37 __frexpl(long double x, int *exp) {
38     union {
39         unsigned i[4];
40         long double q;
41     } xx;
42     unsigned hx;
43     int e, s;

45     xx.q = x;
46     hx = xx.i[0] & ~0x80000000;

48     if (hx >= 0x7fff0000) { /* x is infinite or NaN */
49         *exp = 0;
50         return (x);
51     }

53     e = 0;
54     if (hx < 0x00010000) { /* x is subnormal or zero */
```

new/usr/src/lib/libm/common/m9x/frexpl.c

2

```
55     if ((hx | xx.i[1] | xx.i[2] | xx.i[3]) == 0) {
56         *exp = 0;
57         return (x);
58     }

60     /* normalize x */
61     s = xx.i[0] & 0x80000000;
62     while ((hx | (xx.i[1] & 0xffff0000)) == 0) {
63         hx = xx.i[1];
64         xx.i[1] = xx.i[2];
65         xx.i[2] = xx.i[3];
66         xx.i[3] = 0;
67         e -= 32;
68     }
69     while (hx < 0x10000) {
70         hx = (hx << 1) | (xx.i[1] >> 31);
71         xx.i[1] = (xx.i[1] << 1) | (xx.i[2] >> 31);
72         xx.i[2] = (xx.i[2] << 1) | (xx.i[3] >> 31);
73         xx.i[3] <<= 1;
74         e--;
75     }
76     xx.i[0] = s | hx;
77 }

79     /* now xx.q is normal */
80     xx.i[0] = (xx.i[0] & ~0x7fff0000) | 0x3ffe0000;
81     *exp = e + (hx >> 16) - 0x3ffe;
82     return (xx.q);
83 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/ldexp.c

1

1614 Tue Nov 25 12:58:46 2014

new/usr/src/lib/libm/common/m9x/ldexp.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __ldexp = ldexp
30 #if defined(ELFOSBJ)
31 #pragma weak ldexp = __ldexp
32 #endif
33
34 #include "libm.h"
35 #include <errno.h>
36
37 double
38 ldexp(double x, int n) {
39     int *px = (int *) &x, ix = px[HIWORD] & ~0x80000000;
40     if (ix >= 0x7ff00000 || (px[LOWORD] | ix) == 0)
41 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
42         return (ix >= 0x7ff80000 ? x : x + x);
43 #else
44         /* assumes sparc-like QNaN */
45         return (x + x);
46 #endif
47     x = scalbn(x, n);
48     ix = px[HIWORD] & ~0x80000000;
49     /*
50      * SVID3 requires both overflow and underflow cases to set errno
```

new/usr/src/lib/libm/common/m9x/ldexp.c

2

```
50     * XPG3/XPG4/XPG4.2/SUSv2 requires overflow to set errno
51     */
52     if (ix >= 0x7ff00000 || (px[LOWORD] | ix) == 0)
53         errno = ERANGE;
54     return (x);
55 }
56 _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/ldexpf.c

1

1106 Tue Nov 25 12:58:46 2014

new/usr/src/lib/libm/common/m9x/ldexpf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __ldexpf = ldexpf
30 #if defined(ELFOBJ)
31 #pragma weak ldexpf = __ldexpf
32 #endif
```

```
32 #include "libm.h"
```

```
34 float
35 ldexpf(float x, int n) {
36     return (scalbnf(x, n));
37 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/ldexpl.c

1

1118 Tue Nov 25 12:58:46 2014

new/usr/src/lib/libm/common/m9x/ldexpl.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __ldexpl = ldexpl
30 #if defined(ELFOSBJ)
31 #pragma weak ldexpl = __ldexpl
32 #endif
33
34 #include "libm.h"
35
36 long double
37 ldexpl(long double x, int n) {
38     return (scalbnl(x, n));
39 }
40
41 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/llrint.c

1

```
*****
2280 Tue Nov 25 12:58:47 2014
new/usr/src/lib/libm/common/m9x/llrint.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __llrint = llrint
30 #if defined(ELFOBJ)
31 #pragma weak llrint = __llrint
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lrint = llrint
33 #pragma weak __lrint = llrint
33 #pragma weak lrint = __llrint
34 #pragma weak __lrint = __llrint
35 #endif
34 #endif

36 /*
37  * llrint(x) rounds its argument to the nearest integer according
38  * to the current rounding direction and converts the result to a
39  * 64 bit signed integer.
40  *
41  * If x is NaN, infinite, or so large that the nearest integer would
42  * exceed 64 bits, the invalid operation exception is raised. If x
43  * is not an integer, the inexact exception is raised.
44  */

46 #include "libm.h"
```

new/usr/src/lib/libm/common/m9x/llrint.c

2

```
48 long long
49 llrint(double x) {
50     /*
51      * Note: The following code works on x86 (in the default rounding
52      * precision mode), but one should just use the fistpll instruction
53      * instead.
54      */
55     union {
56         unsigned i[2];
57         double d;
58     } xx, yy;
59     unsigned hx;

61     xx.d = x;
62     hx = xx.i[HIWORD] & ~0x80000000;

64     if (hx < 0x43300000) { /* |x| < 2^52 */
65         /* add and subtract a power of two to round x to an integer */
66         #if defined(__sparc) || defined(__amd64)
67             yy.i[HIWORD] = (xx.i[HIWORD] & 0x80000000) | 0x43300000;
68         #elif defined(__i386) /* !defined(__amd64) */
69             yy.i[HIWORD] = (xx.i[HIWORD] & 0x80000000) | 0x43e00000;
70         #else
71         #error Unknown architecture
72         #endif
73         yy.i[LOWORD] = 0;
74         x = (x + yy.d) - yy.d;
75     }

77     /* now x is nan, inf, or integral */
78     return ((long long) x);
79 }
_____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/m9x/llrintf.c

1

```
*****
1996 Tue Nov 25 12:58:47 2014
new/usr/src/lib/libm/common/m9x/llrintf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __llrintf = llrintf
30 #if defined(ELFOBJ)
31 #pragma weak llrintf = __llrintf
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lrintf = llrintf
33 #pragma weak __lrintf = llrintf
33 #pragma weak lrintf = __llrintf
34 #pragma weak __lrintf = __llrintf
35 #endif
34 #endif

36 #include "libm.h"

38 long long
39 llrintf(float x) {
40     /*
41      * Note: The following code works on x86 (in the default rounding
42      * precision mode), but one should just use the fistpll instruction
43      * instead.
44      */
45     union {
46         unsigned i;
47         float f;
```

new/usr/src/lib/libm/common/m9x/llrintf.c

2

```
48     } xx, yy;
49     unsigned hx;

51     xx.f = x;
52     hx = xx.i & ~0x80000000;

54     if (hx < 0x4b000000) { /* |x| < 2^23 */
55         /* add and subtract a power of two to round x to an integer */
56 #if defined(__sparc) || defined(__amd64)
57         yy.i = (xx.i & 0x80000000) | 0x4b000000;
58 #elif defined(__i386)
59         /* assume 64-bit precision */
60         yy.i = (xx.i & 0x80000000) | 0x5f000000;
61 #else
62 #error Unknown architecture
63 #endif
64         x = (x + yy.f) - yy.f;

66         /*
67          * on LP32 architectures, we can just convert x to a 32-bit
68          * integer and sign-extend it
69          */
70         return ((long) x);
71     }

73     /* now x is nan, inf, or integral */
74     return ((long long) x);
75 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/llrint1.c

1

```
*****
4313 Tue Nov 25 12:58:48 2014
new/usr/src/lib/libm/common/m9x/llrint1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __llrintl = llrintl
30 #if defined(ELFOBJ)
31 #pragma weak llrintl = __llrintl
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lrintl = llrintl
33 #pragma weak __lrintl = llrintl
33 #pragma weak lrintl = __llrintl
34 #pragma weak __lrintl = __llrintl
35 #endif
36 #endif
37
38 #include "libm.h"
39
40 #if defined(__sparc)
41 #include "fma.h"
41 #include "fenv_inlines.h"
42
43 long long
44 llrintl(long double x) {
45     union {
46         unsigned i[4];
47         long double q;

```

new/usr/src/lib/libm/common/m9x/llrint1.c

2

```
48     } xx;
49     union {
50         unsigned i[2];
51         long long l;
52     } zz;
53     union {
54         unsigned i;
55         float f;
56     } tt;
57     unsigned int hx, sx, frac, fsr;
58     int rm, j;
59     volatile float dummy;
60
61     xx.q = x;
62     sx = xx.i[0] & 0x80000000;
63     hx = xx.i[0] & ~0x80000000;
64
65     /* handle trivial cases */
66     if (hx > 0x403e0000) { /* |x| > 2^63 + ... or x is nan */
67         /* convert an out-of-range float */
68         tt.i = sx | 0x7f000000;
69         return ((long long) tt.f);
70     } else if ((hx | xx.i[1] | xx.i[2] | xx.i[3]) == 0) /* x is zero */
71         return (0LL);
72
73     /* get the rounding mode */
74     __fenv_getfsr32(&fsr);
75     rm = fsr >> 30;
76
77     /* flip the sense of directed roundings if x is negative */
78     if (sx)
79         rm ^= rm >> 1;
80
81     /* handle |x| < 1 */
82     if (hx < 0x3fff0000) {
83         dummy = 1.0e30f; /* x is nonzero, so raise inexact */
84         dummy += 1.0e-30f;
85         if (rm == FSR_RP || (rm == FSR_RN && (hx >= 0x3ffe0000 &&
86             ((hx & 0xffff) | xx.i[1] | xx.i[2] | xx.i[3])))
87             return (sx ? -1LL : 1LL);
88         return (0LL);
89     }
90
91     /* extract the integer and fractional parts of x */
92     j = 0x406f - (hx >> 16);
93     xx.i[0] = 0x10000 | (xx.i[0] & 0xffff);
94     if (j >= 96) {
95         zz.i[0] = 0;
96         zz.i[1] = xx.i[0] >> (j - 96);
97         frac = ((xx.i[0] << 1) << (127 - j)) | (xx.i[1] >> (j - 96));
98         if (((xx.i[1] << 1) << (127 - j)) | xx.i[2] | xx.i[3])
99             frac |= 1;
100     } else if (j >= 64) {
101         zz.i[0] = xx.i[0] >> (j - 64);
102         zz.i[1] = ((xx.i[0] << 1) << (95 - j)) | (xx.i[1] >> (j - 64));
103         frac = ((xx.i[1] << 1) << (95 - j)) | (xx.i[2] >> (j - 64));
104         if (((xx.i[2] << 1) << (95 - j)) | xx.i[3])
105             frac |= 1;
106     } else {
107         zz.i[0] = ((xx.i[0] << 1) << (63 - j)) | (xx.i[1] >> (j - 32));
108         zz.i[1] = ((xx.i[1] << 1) << (63 - j)) | (xx.i[2] >> (j - 32));
109         frac = ((xx.i[2] << 1) << (63 - j)) | (xx.i[3] >> (j - 32));
110         if ((xx.i[3] << 1) << (63 - j))
111             frac |= 1;
112     }

```

```
114     /* round */
115     if (frac && (rm == FSR_RP || (rm == FSR_RN && (frac > 0x80000000u ||
116         (frac == 0x80000000 && (zz.i[1] & 1)))))) {
117         if (++zz.i[1] == 0)
118             zz.i[0]++;
119     }
120
121     /* check for result out of range (note that z is |x| at this point) */
122     if (zz.i[0] > 0x80000000u || (zz.i[0] == 0x80000000 && (zz.i[1] ||
123         !sx))) {
124         tt.i = sx | 0x7f000000;
125         return ((long long) tt.f);
126     }
127
128     /* raise inexact if need be */
129     if (frac) {
130         dummy = 1.0e30F;
131         dummy += 1.0e-30F;
132     }
133
134     /* negate result if need be */
135     if (sx) {
136         zz.i[0] = ~zz.i[0];
137         zz.i[1] = ~zz.i[1];
138         if (zz.i[1] == 0)
139             zz.i[0]++;
140     }
141     return (zz.l);
142 }
unchanged portion omitted
```

new/usr/src/lib/libm/common/m9x/llround.c

1

```
*****
2185 Tue Nov 25 12:58:48 2014
new/usr/src/lib/libm/common/m9x/llround.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __llround = llround
30 #if defined(ELFOBJ)
31 #pragma weak llround = __llround
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lround = llround
33 #pragma weak __lround = llround
33 #pragma weak lround = __llround
34 #pragma weak __lround = __llround
35 #endif
34 #endif
35
36 /*
37  * llround(x) rounds its argument to the nearest integer, rounding
38  * ties away from zero, and converts the result to a 64 bit signed
39  * integer.
40  *
41  * If x is NaN, infinite, or so large that the nearest integer
42  * would exceed 64 bits, the invalid operation exception is raised.
43  */
44
45 #include "libm.h"
46
47 long long
```

new/usr/src/lib/libm/common/m9x/llround.c

2

```
48 llround(double x) {
49     union {
50         unsigned i[2];
51         double d;
52     } xx;
53     unsigned hx, sx, i;
54
55     xx.d = x;
56     hx = xx.i[HIWORD] & ~0x80000000;
57     sx = xx.i[HIWORD] & 0x80000000;
58
59     if (hx < 0x43300000) { /* |x| < 2^52 */
60         /* handle |x| < 1 */
61         if (hx < 0x3ff00000) {
62             if (hx >= 0x3fe00000)
63                 return (sx ? -1LL : 1LL);
64             return (0LL);
65         }
66
67         /* round x at the integer bit */
68         if (hx < 0x41300000) {
69             i = 1 << (0x412 - (hx >> 20));
70             xx.i[HIWORD] = (xx.i[HIWORD] + i) & ~(i | (i - 1));
71             xx.i[LOWORD] = 0;
72         } else {
73             i = 1 << (0x432 - (hx >> 20));
74             xx.i[LOWORD] += i;
75             if (xx.i[LOWORD] < i)
76                 xx.i[HIWORD]++;
77             xx.i[LOWORD] &= ~(i | (i - 1));
78         }
79     }
80
81     /* now x is nan, inf, or integral */
82     return ((long long) xx.d);
83 }
84
85 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/llroundf.c

1

```
*****
1795 Tue Nov 25 12:58:49 2014
new/usr/src/lib/libm/common/m9x/llroundf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __llroundf = llroundf
30 #if defined(ELFOBJ)
31 #pragma weak llroundf = __llroundf
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lroundf = llroundf
33 #pragma weak __lroundf = llroundf
33 #pragma weak lroundf = __llroundf
34 #pragma weak __lroundf = __llroundf
35 #endif
34 #endif

36 #include "libm.h"

38 long long
39 llroundf(float x) {
40     union {
41         unsigned i;
42         float f;
43     } xx;
44     unsigned hx, sx, i;

46     xx.f = x;
47     hx = xx.i & ~0x80000000;
```

new/usr/src/lib/libm/common/m9x/llroundf.c

2

```
48     sx = xx.i & 0x80000000;

50     if (hx < 0x4b000000) { /* |x| < 2^23 */
51         /* handle |x| < 1 */
52         if (hx < 0x3f800000) {
53             if (hx >= 0x3f000000)
54                 return (sx ? -1LL : 1LL);
55             return (0LL);
56         }

58         /* round x at the integer bit */
59         i = 1 << (0x95 - (hx >> 23));
60         xx.i = (xx.i + i) & ~((i << 1) - 1);

62         /*
63          * on LP32 architectures, we can just convert x to a 32-bit
64          * integer and sign-extend it
65          */
66         return ((long) xx.f);
67     }

69     /* now x is nan, inf, or integral */
70     return ((long long) x);
71 }

unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/llroundl.c

1

```
*****
3819 Tue Nov 25 12:58:49 2014
new/usr/src/lib/libm/common/m9x/llroundl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __llroundl = llroundl
30 #if defined(ELFOBJ)
31 #pragma weak llroundl = __llroundl
31 #if defined(__sparcv9) || defined(__amd64)
32 #pragma weak lroundl = llroundl
33 #pragma weak __lroundl = llroundl
33 #pragma weak lroundl = __llroundl
34 #pragma weak __lroundl = __llroundl
35 #endif
34 #endif
35
36 #include "libm.h"
37
38 #if defined(__sparc)
39 long long
40 llroundl(long double x) {
41     union {
42         unsigned i[4];
43         long double q;
44     } xx;
45     union {
46         unsigned i[2];
47         long long l;
```

new/usr/src/lib/libm/common/m9x/llroundl.c

2

```
48     } zz;
49     union {
50         unsigned i;
51         float f;
52     } tt;
53     unsigned hx, sx, frac;
54     int j;
55
56     xx.q = x;
57     sx = xx.i[0] & 0x80000000;
58     hx = xx.i[0] & ~0x80000000;
59
60     /* handle trivial cases */
61     if (hx > 0x403e0000) { /* |x| > 2^63 + ... or x is nan */
62         /* convert an out-of-range float */
63         tt.i = sx | 0x7f000000;
64         return ((long long) tt.f);
65     }
66
67     /* handle |x| < 1 */
68     if (hx < 0x3fff0000) {
69         if (hx >= 0x3ffe0000)
70             return (sx ? -1LL : 1LL);
71         return (0LL);
72     }
73
74     /* extract the integer and fractional parts of x */
75     j = 0x406f - (hx >> 16);
76     xx.i[0] = 0x10000 | (xx.i[0] & 0xffff);
77     if (j >= 96) {
78         zz.i[0] = 0;
79         zz.i[1] = xx.i[0] >> (j - 96);
80         frac = ((xx.i[0] << 1) << (127 - j)) | (xx.i[1] >> (j - 96));
81         if ((xx.i[1] << 1) << (127 - j)) | xx.i[2] | xx.i[3])
82             frac |= 1;
83     } else if (j >= 64) {
84         zz.i[0] = xx.i[0] >> (j - 64);
85         zz.i[1] = ((xx.i[0] << 1) << (95 - j)) | (xx.i[1] >> (j - 64));
86         frac = ((xx.i[1] << 1) << (95 - j)) | (xx.i[2] >> (j - 64));
87         if ((xx.i[2] << 1) << (95 - j)) | xx.i[3])
88             frac |= 1;
89     } else {
90         zz.i[0] = ((xx.i[0] << 1) << (63 - j)) | (xx.i[1] >> (j - 32));
91         zz.i[1] = ((xx.i[1] << 1) << (63 - j)) | (xx.i[2] >> (j - 32));
92         frac = ((xx.i[2] << 1) << (63 - j)) | (xx.i[3] >> (j - 32));
93         if ((xx.i[3] << 1) << (63 - j))
94             frac |= 1;
95     }
96
97     /* round */
98     if (frac >= 0x80000000u) {
99         if (++zz.i[1] == 0)
100             zz.i[0]++;
101     }
102
103     /* check for result out of range (note that z is |x| at this point) */
104     if (zz.i[0] > 0x80000000u || (zz.i[0] == 0x80000000 && (zz.i[1] ||
105         !sx))) {
106         tt.i = sx | 0x7f000000;
107         return ((long long) tt.f);
108     }
109
110     /* negate result if need be */
111     if (sx) {
112         zz.i[0] = ~zz.i[0];
113         zz.i[1] = -zz.i[1];
```

new/usr/src/lib/libm/common/m9x/llround1.c

3

```
114         if (zz.i[1] == 0)
115             zz.i[0]++;
116     }
117
118     return (zz.l);
119 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/lrint.c

1

```
*****
2236 Tue Nov 25 12:58:50 2014
new/usr/src/lib/libm/common/m9x/lrint.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __lrint = lrint
30 #if defined(ELFOBJ)
31 #pragma weak lrint = __lrint
32 #endif

32 /*
33  * lrint(x) rounds its argument to the nearest integer according
34  * to the current rounding direction and converts the result to
35  * a 32 bit signed integer.
36  *
37  * If x is NaN, infinite, or so large that the nearest integer
38  * would exceed 32 bits, the invalid operation exception is raised.
39  * If x is not an integer, the inexact exception is raised.
40  */

42 #include <sys/isa_defs.h>      /* _ILP32 */
43 #include "libm.h"

45 #if defined(_ILP32)
46 long
47 lrint(double x) {
48     /*
49      * Note: The following code works on x86 (in the default rounding
```

new/usr/src/lib/libm/common/m9x/lrint.c

2

```
50     * precision mode), but one should just use the fistpl instruction
51     * instead.
52     */
53     union {
54         unsigned i[2];
55         double d;
56     } xx, yy;
57     unsigned hx;

59     xx.d = x;
60     hx = xx.i[HIWORD] & ~0x80000000;
61     if (hx < 0x43300000) { /* |x| < 2^52 */
62         /* add and subtract a power of two to round x to an integer */
63 #if defined(__sparc)
64         yy.i[HIWORD] = (xx.i[HIWORD] & 0x80000000) | 0x43300000;
65 #elif defined(__x86)
66         yy.i[HIWORD] = (xx.i[HIWORD] & 0x80000000) | 0x43e00000;
67 #else
68 #error Unknown architecture
69 #endif
70         yy.i[LOWORD] = 0;
71         x = (x + yy.d) - yy.d;
72     }

74     /* now x is nan, inf, or integral */
75     return ((long) x);
76 }
_____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/m9x/lrintf.c

1

```
*****
1868 Tue Nov 25 12:58:50 2014
new/usr/src/lib/libm/common/m9x/lrintf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __lrintf = lrintf
30 #if defined(ELFOBJ)
31 #pragma weak lrintf = __lrintf
32 #endif
33
34 #include <sys/isa_defs.h> /* _ILP32 */
35 #include "libm.h"
36
37 #if defined(_ILP32)
38 long
39 lrintf(float x) {
40     /*
41      * Note: The following code works on x86 (in the default rounding
42      * precision mode), but one should just use the fistpl instruction
43      * instead.
44      */
45     union {
46         unsigned i;
47         float f;
48     } xx, yy;
49     unsigned hx;
50
51     xx.f = x;
```

new/usr/src/lib/libm/common/m9x/lrintf.c

2

```
50     hx = xx.i & ~0x80000000;
51     if (hx < 0x4b000000) { /* |x| < 2^23 */
52         /* add and subtract a power of two to round x to an integer */
53         #if defined(__sparc)
54             yy.i = (xx.i & 0x80000000) | 0x4b000000;
55         #elif defined(__x86)
56             /* assume 64-bit precision */
57             yy.i = (xx.i & 0x80000000) | 0x5f000000;
58         #else
59             #error Unknown architecture
60         #endif
61         x = (x + yy.f) - yy.f;
62         return ((long) x);
63     }
64
65     /* now x is nan, inf, or integral */
66     return ((long) x);
67 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/lrintl.c

1

```
*****
3883 Tue Nov 25 12:58:50 2014
new/usr/src/lib/libm/common/m9x/lrintl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __lrintl = lrintl
30 #if defined(ELFOBJ)
31 #pragma weak lrintl = __lrintl
32 #endif

32 #include <sys/isa_defs.h> /* _ILP32 */
33 #include "libm.h"

35 #if defined(_ILP32)
36 #if defined(__sparc)

38 #include "fma.h"
39 #include "fenv_inlines.h"

41 long
42 lrintl(long double x) {
43     union {
44         unsigned int i[4];
45         long double q;
46     } xx;
47     union {
48         unsigned int i;
49         float f;

```

new/usr/src/lib/libm/common/m9x/lrintl.c

2

```
50     } tt;
51     unsigned int hx, sx, frac, l, fsr;
52     int rm, j;
53     volatile float dummy;

55     xx.q = x;
56     sx = xx.i[0] & 0x80000000;
57     hx = xx.i[0] & ~0x80000000;

59     /* handle trivial cases */
60     if (hx > 0x401e0000) { /* |x| > 2^31 + ... or x is nan */
61         /* convert an out-of-range float */
62         tt.i = sx | 0x7f000000;
63         return ((long) tt.f);
64     } else if ((hx | xx.i[1] | xx.i[2] | xx.i[3]) == 0) /* x is zero */
65         return (0L);

67     /* get the rounding mode */
68     __fenv_getfsr32(&fsr);
69     rm = fsr >> 30;

71     /* flip the sense of directed roundings if x is negative */
72     if (sx)
73         rm ^= rm >> 1;

75     /* handle |x| < 1 */
76     if (hx < 0x3fff0000) {
77         dummy = 1.0e30F; /* x is nonzero, so raise inexact */
78         dummy += 1.0e-30F;
79         if (rm == FSR_RN || (rm == FSR_RN && (hx >= 0x3ffe0000 &&
80             ((hx & 0xffff) | xx.i[1] | xx.i[2] | xx.i[3])))
81             return (sx ? -1L : 1L);
82         return (0L);
83     }

85     /* extract the integer and fractional parts of x */
86     j = 0x406f - (hx >> 16); /* 91 <= j <= 112 */
87     xx.i[0] = 0x10000 | (xx.i[0] & 0xffff);
88     if (j >= 96) { /* 96 <= j <= 112 */
89         l = xx.i[0] >> (j - 96);
90         frac = ((xx.i[0] << 1) << (127 - j)) | (xx.i[1] >> (j - 96));
91         if (((xx.i[1] << 1) << (127 - j)) | xx.i[2] | xx.i[3])
92             frac |= 1;
93     } else { /* 91 <= j <= 95 */
94         l = (xx.i[0] << (96 - j)) | (xx.i[1] >> (j - 64));
95         frac = (xx.i[1] << (96 - j)) | (xx.i[2] >> (j - 64));
96         if ((xx.i[2] << (96 - j)) | xx.i[3])
97             frac |= 1;
98     }

100     /* round */
101     if (frac && (rm == FSR_RP || (rm == FSR_RN && (frac > 0x80000000U ||
102         (frac == 0x80000000 && (1 & 1)))))
103         l++;

105     /* check for result out of range (note that z is |x| at this point) */
106     if (l > 0x80000000U || (l == 0x80000000U && !sx)) {
107         tt.i = sx | 0x7f000000;
108         return ((long) tt.f);
109     }

111     /* raise inexact if need be */
112     if (frac) {
113         dummy = 1.0e30F;
114         dummy += 1.0e-30F;
115     }

```

new/usr/src/lib/libm/common/m9x/lrintl.c

3

```
117     /* negate result if need be */
118     if (sx)
119         l = -l;
120     return ((long) l);
121 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/lround.c

1

```
*****
2169 Tue Nov 25 12:58:51 2014
new/usr/src/lib/libm/common/m9x/lround.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __lround = lround
30 #if defined(ELFOBJ)
31 #pragma weak lround = __lround
32 #endif

32 /*
33  * lround(x) rounds its argument to the nearest integer, rounding ties
34  * away from zero, and converts the result to a 32 bit signed integer.
35  *
36  * If x is NaN, infinite, or so large that the nearest integer
37  * would exceed 32 bits, the invalid operation exception is raised.
38  */

40 #include <sys/isa_defs.h> /* _ILP32 */
41 #include "libm.h"

43 #if defined(_ILP32)
44 long
45 lround(double x) {
46     union {
47         unsigned i[2];
48         double d;
49     } xx;
```

new/usr/src/lib/libm/common/m9x/lround.c

2

```
50     unsigned hx, sx, i;

52     xx.d = x;
53     hx = xx.i[HIWORD] & ~0x80000000;
54     sx = xx.i[HIWORD] & 0x80000000;
55     if (hx < 0x43300000) { /* |x| < 2^52 */
56         if (hx < 0x3ff00000) { /* |x| < 1 */
57             if (hx >= 0x3fe00000)
58                 return (sx ? -1L : 1L);
59             return (0L);
60         }

62         /* round x at the integer bit */
63         if (hx < 0x41300000) {
64             i = 1 << (0x412 - (hx >> 20));
65             xx.i[HIWORD] = (xx.i[HIWORD] + i) & ~(i | (i - 1));
66             xx.i[LOWORD] = 0;
67         } else {
68             i = 1 << (0x432 - (hx >> 20));
69             xx.i[LOWORD] += i;
70             if (xx.i[LOWORD] < i)
71                 xx.i[HIWORD]++;
72             xx.i[LOWORD] &= ~(i | (i - 1));
73         }
74     }

76     /* now x is nan, inf, or integral */
77     return ((long) xx.d);
78 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/lroundf.c

1

1675 Tue Nov 25 12:58:51 2014

new/usr/src/lib/libm/common/m9x/lroundf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __lroundf = lroundf
30 #if defined(ELFOBJ)
31 #pragma weak lroundf = __lroundf
32 #endif
```

```
33 #include <sys/isa_defs.h> /* _ILP32 */
33 #include "libm.h"
```

```
35 #if defined(_ILP32)
36 long
37 lroundf(float x) {
38     union {
39         unsigned i;
40         float f;
41     } xx;
42     unsigned hx, sx, i;
43
44     xx.f = x;
45     hx = xx.i & ~0x80000000;
46     sx = xx.i & 0x80000000;
47     if (hx < 0x4b000000) { /* |x| < 2^23 */
48         if (hx < 0x3f800000) { /* |x| < 1 */
49             if (hx >= 0x3f000000)
```

new/usr/src/lib/libm/common/m9x/lroundf.c

2

```
50         return (sx ? -1L : 1L);
51     return (0L);
52 }
53
54     /* round x at the integer bit */
55     i = 1 << (0x95 - (hx >> 23));
56     xx.i = (xx.i + i) & ~(i << 1) - 1;
57     return ((long) xx.f);
58 }
59
60     /* now x is nan, inf, or integral */
61     return ((long) x);
62 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/lroundl.c

1

```
*****
3372 Tue Nov 25 12:58:52 2014
new/usr/src/lib/libm/common/m9x/lroundl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __lroundl = lroundl
30 #if defined(ELFOBJ)
31 #pragma weak lroundl = __lroundl
32 #endif
33
33 #include <sys/isa_defs.h> /* _ILP32 */
33 #include "libm.h"
34
35 #if defined(_ILP32)
36 #if defined(__sparc)
37 long
38 lroundl(long double x) {
39     union {
40         unsigned i[4];
41         long double q;
42     } xx;
43     union {
44         unsigned i;
45         float f;
46     } tt;
47     unsigned hx, sx, frac, l;
48     int j;
```

new/usr/src/lib/libm/common/m9x/lroundl.c

2

```
50     xx.q = x;
51     sx = xx.i[0] & 0x80000000;
52     hx = xx.i[0] & ~0x80000000;
53
54     /* handle trivial cases */
55     if (hx > 0x401e0000) { /* |x| > 2^31 + ... or x is nan */
56         /* convert an out-of-range float */
57         tt.i = sx | 0x7f000000;
58         return ((long) tt.f);
59     }
60
61     /* handle |x| < 1 */
62     if (hx > 0x3fff0000) {
63         if (hx >= 0x3ffe0000)
64             return (sx ? -1L : 1L);
65         return (0L);
66     }
67
68     /* extract the integer and fractional parts of x */
69     j = 0x406f - (hx >> 16); /* 91 <= j <= 112 */
70     xx.i[0] = 0x10000 | (xx.i[0] & 0xffff);
71     if (j >= 96) { /* 96 <= j <= 112 */
72         l = xx.i[0] >> (j - 96);
73         frac = ((xx.i[0] << 1) << (127 - j)) | (xx.i[1] >> (j - 96));
74         if (((xx.i[1] << 1) << (127 - j)) | xx.i[2] | xx.i[3])
75             frac |= 1;
76     } else { /* 91 <= j <= 95 */
77         l = (xx.i[0] << (96 - j)) | (xx.i[1] >> (j - 64));
78         frac = (xx.i[1] << (96 - j)) | (xx.i[2] >> (j - 64));
79         if ((xx.i[2] << (96 - j)) | xx.i[3])
80             frac |= 1;
81     }
82
83     /* round */
84     if (frac >= 0x80000000U)
85         l++;
86
87     /* check for result out of range (note that z is |x| at this point) */
88     if (l > 0x80000000U || (l == 0x80000000U && !sx)) {
89         tt.i = sx | 0x7f000000;
90         return ((long) tt.f);
91     }
92
93     /* negate result if need be */
94     if (sx)
95         l = -l;
96     return ((long) l);
97 }
unchanged_portion_omitted
```

```

*****
2414 Tue Nov 25 12:58:52 2014
new/usr/src/lib/libm/common/m9x/modf.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak modf = __modf
32 #pragma weak _modf = __modf
33 #endif
34
35 /*
36  * modf(x, iptr) decomposes x into an integral part and a fractional
37  * part both having the same sign as x. It stores the integral part
38  * in *iptr and returns the fractional part.
39  *
40  * If x is infinite, modf sets *iptr to x and returns copysign(0.0,x).
41  * If x is NaN, modf sets *iptr to x and returns x.
42  *
43  * If x is a signaling NaN, this code does not attempt to raise the
44  * invalid operation exception.
45 */
46
47 #include "libm.h"
48
49 double
50 __modf(double x, double *iptr) {
51     union {
52         unsigned i[2];
53         double d;
54     } xx, yy;
55     unsigned hx, s;

```

```

55     xx.d = x;
56     hx = xx.i[HIWORD] & ~0x80000000;
57
58     if (hx >= 0x43300000) { /* x is NaN, infinite, or integral */
59         *iptr = x;
60         if (hx == 0x7ff00000 || (hx == 0x7ff00000 &&
61             xx.i[LOWORD] == 0)) {
62             xx.i[HIWORD] &= 0x80000000;
63             xx.i[LOWORD] = 0;
64         }
65         return (xx.d);
66     }
67
68     if (hx < 0x3ff00000) { /* |x| < 1 */
69         xx.i[HIWORD] &= 0x80000000;
70         xx.i[LOWORD] = 0;
71         *iptr = xx.d;
72         return (x);
73     }
74
75     /* split x at the binary point */
76     s = xx.i[HIWORD] & 0x80000000;
77     if (hx < 0x41400000) {
78         yy.i[HIWORD] = xx.i[HIWORD] & ~((1 << (0x413 - (hx >> 20))) -
79             1);
80         yy.i[LOWORD] = 0;
81     } else {
82         yy.i[HIWORD] = xx.i[HIWORD];
83         yy.i[LOWORD] = xx.i[LOWORD] & ~((1 << (0x433 - (hx >> 20))) -
84             1);
85     }
86     *iptr = yy.d;
87     xx.d -= yy.d;
88     xx.i[HIWORD] = (xx.i[HIWORD] & ~0x80000000) | s;
89     /* keep sign of x */
90     return (xx.d);
91 }

```

_____unchanged_portion_omitted_____

```

*****
1691 Tue Nov 25 12:58:53 2014
new/usr/src/lib/libm/common/m9x/modff.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak modff = __modff
31 #pragma weak _modff = __modff
33 #endif

33 #include "libm.h"

35 float
36 __modff(float x, float *iptr) {
37     union {
38         unsigned i;
39         float f;
40     } xx, yy;
41     unsigned hx, s;

43     xx.f = x;
44     hx = xx.i & ~0x80000000;

46     if (hx >= 0x4b000000) { /* x is NaN, infinite, or integral */
47         *iptr = x;
48         if (hx <= 0x7f800000)
49             xx.i &= 0x80000000;
50         return (xx.f);
51     }

53     if (hx < 0x3f800000) { /* |x| < 1 */
54         xx.i &= 0x80000000;

```

```

55         *iptr = xx.f;
56         return (x);
57     }

59     /* split x at the binary point */
60     s = xx.i & 0x80000000;
61     yy.i = xx.i & ~(1 << (0x96 - (hx >> 23))) - 1);
62     *iptr = yy.f;
63     xx.f -= yy.f;
64     xx.i = (xx.i & ~0x80000000) | s;
65     /* restore sign in case difference is 0 */
66     return (xx.f);
67 }

```

unchanged_portion_omitted


```

*****
3532 Tue Nov 25 12:58:53 2014
new/usr/src/lib/libm/common/m9x/modfl.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak modfl = __modfl
32 #endif
33
34 #include "libm.h"
35
36 #if defined(__sparc)
37
38 long double
39 __modfl(long double x, long double *iptr) {
40     union {
41         unsigned i[4];
42         long double q;
43     } xx, yy;
44     unsigned hx, s;
45
46     xx.q = x;
47     hx = xx.i[0] & ~0x80000000;
48
49     if (hx >= 0x406f0000) { /* x is NaN, infinite, or integral */
50         *iptr = x;
51         if (hx < 0x7fff0000 || (hx == 0x7fff0000 &&
52             (xx.i[1] | xx.i[2] | xx.i[3]) == 0)) {
53             xx.i[0] &= 0x80000000;
54             xx.i[1] = xx.i[2] = xx.i[3] = 0;
55         }
56     }
57     return (xx.q);
58 }
59

```

```

55     }
56
57     if (hx < 0x3fff0000) { /* |x| < 1 */
58         xx.i[0] &= 0x80000000;
59         xx.i[1] = xx.i[2] = xx.i[3] = 0;
60         *iptr = xx.q;
61         return (x);
62     }
63
64     /* split x at the binary point */
65     s = xx.i[0] & 0x80000000;
66     if (hx < 0x40100000) {
67         yy.i[0] = xx.i[0] & ~((1 << (0x400f - (hx >> 16))) - 1);
68         yy.i[1] = yy.i[2] = yy.i[3] = 0;
69     } else if (hx < 0x40300000) {
70         yy.i[0] = xx.i[0];
71         yy.i[1] = xx.i[1] & ~((1 << (0x402f - (hx >> 16))) - 1);
72         yy.i[2] = yy.i[3] = 0;
73     } else if (hx < 0x40500000) {
74         yy.i[0] = xx.i[0];
75         yy.i[1] = xx.i[1];
76         yy.i[2] = xx.i[2] & ~((1 << (0x404f - (hx >> 16))) - 1);
77         yy.i[3] = 0;
78     } else {
79         yy.i[0] = xx.i[0];
80         yy.i[1] = xx.i[1];
81         yy.i[2] = xx.i[2];
82         yy.i[3] = xx.i[3] & ~((1 << (0x406f - (hx >> 16))) - 1);
83     }
84     *iptr = yy.q;
85
86     /*
87      * we could implement the following more efficiently than by using
88      * software emulation of fsubq, but we'll do it this way for now
89      * (and hope hardware support becomes commonplace)
90     */
91     xx.q -= yy.q;
92     xx.i[0] = (xx.i[0] & ~0x80000000) | s; /* keep sign of x */
93     return (xx.q);
94 }

```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/nan.c

1

1500 Tue Nov 25 12:58:53 2014

new/usr/src/lib/libm/common/m9x/nan.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
```

```
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #if defined(ELFOBJ)
30 #pragma weak nan = __nan
32 #endif
```

```
32 /*
33  * nan(c) returns a NaN. This implementation ignores c.
34 */
```

```
36 #include "libm.h"
37 #include <sys/isa_defs.h>
```

```
39 #if defined(__sparc)
```

```
41 static const union {
42     unsigned i[2];
43     double d;
44 } __nan_union = { 0x7fffffff, 0xffffffff };
_____unchanged_portion_omitted_
```

1196 Tue Nov 25 12:58:54 2014

new/usr/src/lib/libm/common/m9x/nanf.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak nanf = __nanf
32 #endif

32 #include "libm.h"

34 static const union {
35     unsigned i;
36     float f;
37 } __nanf_union = { 0x7fffffff };
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/nanl.c

1

1443 Tue Nov 25 12:58:55 2014

new/usr/src/lib/libm/common/m9x/nanl.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak nanl = __nanl
32 #endif

32 #include "libm.h"

34 #if defined(__sparc)

36 static const union {
37     unsigned i[4];
38     long double ld;
39 } __nanl_union = { 0x7fffffff, 0xffffffff, 0xffffffff, 0xffffffff };
    unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/nearbyint.c

1

```
*****
3162 Tue Nov 25 12:58:55 2014
new/usr/src/lib/libm/common/m9x/nearbyint.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #if defined(ELFOBJ)
31 #pragma weak nearbyint = __nearbyint
32 #endif
33
34 /*
35  * nearbyint(x) returns the nearest fp integer to x in the direction
36  * corresponding to the current rounding direction without raising
37  * the inexact exception.
38  *
39  * nearbyint(x) is x unchanged if x is +/-0 or +/-inf. If x is NaN,
40  * nearbyint(x) is also NaN.
41  */
42
43 #include "libm.h"
44 #include "fenv_synonyms.h"
45 #include <fenv.h>
46
47 double
48 __nearbyint(double x) {
49     union {
50         unsigned i[2];
51         double d;
52     } xx;
```

new/usr/src/lib/libm/common/m9x/nearbyint.c

2

```
50     unsigned hx, sx, i, frac;
51     int rm, j;
52
53     xx.d = x;
54     sx = xx.i[HIWORD] & 0x80000000;
55     hx = xx.i[HIWORD] & ~0x80000000;
56
57     /* handle trivial cases */
58     if (hx >= 0x43300000) { /* x is nan, inf, or already integral */
59         if (hx >= 0x7ff00000) /* x is inf or nan */
60 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
61             return (hx >= 0x7ff80000 ? x : x + x);
62             /* assumes sparc-like QNaN */
63 #else
64             return (x + x);
65 #endif
66     } else if ((hx | xx.i[LOWORD]) == 0) /* x is zero */
67         return (x);
68
69     /* get the rounding mode */
70     rm = fegetround();
71
72     /* flip the sense of directed roundings if x is negative */
73     if (sx && (rm == FE_UPWARD || rm == FE_DOWNWARD))
74         rm = (FE_UPWARD + FE_DOWNWARD) - rm;
75
76     /* handle |x| < 1 */
77     if (hx < 0x3ff00000) {
78         if (rm == FE_UPWARD || (rm == FE_TONEAREST &&
79             (hx >= 0x3fe00000 && ((hx & 0xffff) | xx.i[LOWORD])))
80             xx.i[HIWORD] = sx | 0x3ff00000;
81         else
82             xx.i[HIWORD] = sx;
83         xx.i[LOWORD] = 0;
84         return (xx.d);
85     }
86
87     /* round x at the integer bit */
88     j = 0x433 - (hx >> 20);
89     if (j >= 32) {
90         i = 1 << (j - 32);
91         frac = ((xx.i[HIWORD] << 1) << (63 - j)) |
92             (xx.i[LOWORD] >> (j - 32));
93         if (xx.i[LOWORD] & (i - 1))
94             frac |= 1;
95         if (!frac)
96             return (x);
97         xx.i[LOWORD] = 0;
98         xx.i[HIWORD] &= ~(i - 1);
99         if ((rm == FE_UPWARD) || ((rm == FE_TONEAREST) &&
100             ((frac > 0x80000000u) || ((frac == 0x80000000) &&
101                 (xx.i[HIWORD] & i))))))
102             xx.i[HIWORD] += i;
103     } else {
104         i = 1 << j;
105         frac = (xx.i[LOWORD] << 1) << (31 - j);
106         if (!frac)
107             return (x);
108         xx.i[LOWORD] &= ~(i - 1);
109         if ((rm == FE_UPWARD) || ((rm == FE_TONEAREST) &&
110             (frac > 0x80000000u) || ((frac == 0x80000000) &&
111                 (xx.i[LOWORD] & i)))) {
112             xx.i[LOWORD] += i;
113             if (xx.i[LOWORD] == 0)
114                 xx.i[HIWORD]++;
115         }
```

new/usr/src/lib/libm/common/m9x/nearbyint.c

3

```
116         }
117     }
118     return (xx.d);
119 }
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/nearbyintf.c

1

```
*****
3970 Tue Nov 25 12:58:56 2014
new/usr/src/lib/libm/common/m9x/nearbyintf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #if defined(ELFOBJ)
30 #pragma weak nearbyintf = __nearbyintf
32 #endif

32 #include "libm.h"
35 #include "fenv_synonyms.h"
33 #include <fenv.h>

35 float
36 __nearbyintf(float x) {
37     union {
38         unsigned i;
39         float f;
40     } xx;
41     unsigned hx, sx, i, frac;
42     int rm;

44     xx.f = x;
45     sx = xx.i & 0x80000000;
46     hx = xx.i & ~0x80000000;

48     /* handle trivial cases */
49     if (hx >= 0x4b000000) { /* x is nan, inf, or already integral */
```

new/usr/src/lib/libm/common/m9x/nearbyintf.c

2

```
50         if (hx > 0x7f800000) /* x is nan */
51             return (x * x); /* + -> * for Cheetah */
52         return (x);
53     } else if (hx == 0) /* x is zero */
54         return (x);

56     /* get the rounding mode */
57     rm = fegetround();

59     /* flip the sense of directed roundings if x is negative */
60     if (sx && (rm == FE_UPWARD || rm == FE_DOWNWARD))
61         rm = (FE_UPWARD + FE_DOWNWARD) - rm;

63     /* handle |x| < 1 */
64     if (hx < 0x3f800000) {
65         if (rm == FE_UPWARD || (rm == FE_TONEAREST && hx > 0x3f000000))
66             xx.i = sx | 0x3f800000;
67         else
68             xx.i = sx;
69         return (xx.f);
70     }

72     /* round x at the integer bit */
73     i = 1 << (0x96 - (hx >> 23));
74     frac = hx & (i - 1);
75     if (!frac)
76         return (x);

78     hx &= ~(i - 1);
79     if (rm == FE_UPWARD || (rm == FE_TONEAREST && (frac > (i >> 1) ||
80         ((frac == (i >> 1)) && (hx & i)))))
81         xx.i = sx | (hx + i);
82     else
83         xx.i = sx | hx;
84     return (xx.f);
85 }

unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/nearbyintl.c

1

4408 Tue Nov 25 12:58:56 2014

new/usr/src/lib/libm/common/m9x/nearbyintl.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #if defined(ELFOBJ)
31 #pragma weak nearbyintl = __nearbyintl
32 #endif
33
34 #include "libm.h"
35 #include "fma.h"
36 #include "fenv_inlines.h"
37
38 #if defined(__sparc)
39
40 static union {
41     unsigned i;
42     float f;
43 } snan = { 0x7f800001 };
44
45 unchanged_portion_omitted
```


new/usr/src/lib/libm/common/m9x/nexttoward.c

1

4646 Tue Nov 25 12:58:57 2014

new/usr/src/lib/libm/common/m9x/nexttoward.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak nexttoward = __nexttoward
32 #endif

32 /*
33  * nexttoward(x, y) delivers the next representable number after x
34  * in the direction of y. If x and y are both zero, the result is
35  * zero with the same sign as y. If either x or y is NaN, the result
36  * is NaN.
37  *
38  * If x != y and the result is infinite, overflow is raised; if
39  * x != y and the result is subnormal or zero, underflow is raised.
40  * (This is wrong, but it's what C99 apparently wants.)
41 */

43 #include "libm.h"

45 #if defined(__sparc)

47 static union {
48     unsigned i[2];
49     double d;
50 } C[] = {
_____unchanged_portion_omitted_
```

new/usr/src/lib/libm/common/m9x/nexttowardf.c

1

3656 Tue Nov 25 12:58:57 2014

new/usr/src/lib/libm/common/m9x/nexttowardf.c

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #if defined(ELFOBJ)
30 #pragma weak nexttowardf = __nexttowardf
32 #endif

32 #include "libm.h"

34 static union {
35     unsigned i;
36     float f;
37 } C[] = {
    _____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/nexttowardl.c

1

```
*****
2735 Tue Nov 25 12:58:58 2014
new/usr/src/lib/libm/common/m9x/nexttowardl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __nexttowardl = nexttowardl
30 #if defined(ELFOBJ)
31 #pragma weak nexttowardl = __nexttowardl
32 #endif
33
34 #include "libm.h"
35 #include <float.h> /* LDBL_MAX, LDBL_MIN */
36
37 #if defined(__sparc)
38 #define n0 0
39 #define n1 1
40 #define n2 2
41 #define n3 3
42 #define X86PDNRML(x)
43 #define INC(px) { \
44     if (++px[n3] == 0) \
45         if (++px[n2] == 0) \
46             if (++px[n1] == 0) \
47                 ++px[n0]; \
48 }
49 #define DEC(px) { \
50     if (--px[n3] == 0xffffffff) \
51         if (--px[n2] == 0xffffffff) \
```

new/usr/src/lib/libm/common/m9x/nexttowardl.c

2

```
52     if (--px[n1] == 0xffffffff) \
53         --px[n0]; \
54 }
55 #elif defined(__x86)
56 #define n0 2
57 #define n1 1
58 #define n2 0
59 #define n3 0
60 /*
61  * if pseudo-denormal, replace by the equivalent normal
62  */
63 #define X86PDNRML(x) if (XBIASED_EXP(x) == 0 && (((int *) &x)[1] & \
64     0x80000000) != 0) \
65     ((int *) &x)[2] |= 1
66 #define INC(px) { \
67     if (++px[n2] == 0) \
68         if ((++px[n1] & ~0x80000000) == 0) \
69             px[n1] = 0x80000000, ++px[n0]; \
70 }
71 #define DEC(px) { \
72     if (--px[n2] == 0xffffffff) \
73         if (--px[n1] == 0x7fffff) \
74             if ((--px[n0] & 0x7fff) != 0) \
75                 px[n1] |= 0x80000000; \
76 }
77 #endif
78
79 long double
80 nexttowardl(long double x, long double y) {
81     int *px = (int *) &x;
82     int *py = (int *) &y;
83
84     if (x == y)
85         return (y); /* C99 requirement */
86
87     if (x != x || y != y)
88         return (x * y);
89
90     if (ISZEROL(x)) { /* x == 0.0 */
91         px[n0] = py[n0] & XSGNMSK;
92         px[n1] = px[n2] = 0;
93         px[n3] = 1;
94     } else {
95         X86PDNRML(x);
96         if ((px[n0] & XSGNMSK) == 0) { /* x > 0.0 */
97             if (x > y) /* x > y */
98                 DEC(px);
99             else
100                 INC(px);
101         } else {
102             if (x < y) /* x < y */
103                 DEC(px);
104             else
105                 INC(px);
106         }
107     }
108 #ifndef lint
109     {
110         volatile long double dummy;
111         int k = XBIASED_EXP(x);
112
113         if (k == 0)
114             dummy = LDBL_MIN * copysign(LDBL_MIN, x);
115         else if (k == 0x7fff)
116             dummy = LDBL_MAX * copysign(LDBL_MAX, x);
117     }
118 #endif
119 }
```

new/usr/src/lib/libm/common/m9x/nexttowardl.c

3

```
116     return (x);  
117 }
```

_____unchanged_portion_omitted_____

```

*****
5960 Tue Nov 25 12:58:58 2014
new/usr/src/lib/libm/common/m9x/remquo.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __remquo = remquo
30 #pragma weak remquo = __remquo

32 /* INDENT OFF */
33 /*
34  * double remquo(double x, double y, int *quo) return remainder(x,y) and an
35  * integer pointer quo such that *quo = N mod {2**31}, where N is the
36  * exact integral part of x/y rounded to nearest even.
37  *
38  * remquo call internal fmodquo
39  */
40 /* INDENT ON */

42 #include "libm.h"
43 #include "libm_synonyms.h"
43 #include "libm_protos.h"
44 #include <math.h> /* fabs() */
45 #include <sys/isa_defs.h>

47 #if defined(_BIG_ENDIAN)
48 #define HIWORD 0
49 #define LOWORD 1
50 #else
51 #define HIWORD 1
52 #define LOWORD 0
53 #endif
54 #define __HI(x) ((int *) &x)[HIWORD]
55 #define __LO(x) ((int *) &x)[LOWORD]

```

```

57 static const double one = 1.0, Zero[] = {0.0, -0.0};

59 static double
60 fmodquo(double x, double y, int *quo) {
61     int n, hx, hy, hz, ix, iy, sx, sq, i, m;
62     unsigned lx, ly, lz;

64     hx = __HI(x); /* high word of x */
65     lx = __LO(x); /* low word of x */
66     hy = __HI(y); /* high word of y */
67     ly = __LO(y); /* low word of y */
68     sx = hx & 0x80000000; /* sign of x */
69     sq = (hx ^ hy) & 0x80000000; /* sign of x/y */
70     hx ^= sx; /* |x| */
71     hy &= 0x7fffffff; /* |y| */

73     /* purge off exception values */
74     *quo = 0;
75     if ((hy | ly) == 0 || hx >= 0x7ff00000 || /* y=0, or x !finite */
76         ((ly | -ly) >> 31) > 0x7ff00000) /* or y is NaN */
77         return ((x * y) / (x * y));
78     if (hx <= hy) {
79         if (hx < hy || lx < ly)
80             return (x); /* |x|<|y| return x */
81         if (lx == ly) {
82             *quo = 1 + (sq >> 30);
83             /* |x|=|y| return x*0 */
84             return (Zero[(unsigned) sx >> 31]);
85         }
86     }

88     /* determine ix = ilogb(x) */
89     if (hx < 0x00100000) { /* subnormal x */
90         if (hx == 0) {
91             for (ix = -1043, i = lx; i > 0; i <= 1)
92                 ix -= 1;
93         } else {
94             for (ix = -1022, i = (hx << 11); i > 0; i <= 1)
95                 ix -= 1;
96         }
97     } else
98         ix = (hx >> 20) - 1023;

100     /* determine iy = ilogb(y) */
101     if (hy < 0x00100000) { /* subnormal y */
102         if (hy == 0) {
103             for (iy = -1043, i = ly; i > 0; i <= 1)
104                 iy -= 1;
105         } else {
106             for (iy = -1022, i = (hy << 11); i > 0; i <= 1)
107                 iy -= 1;
108         }
109     } else
110         iy = (hy >> 20) - 1023;

112     /* set up {hx,lx}, {hy,ly} and align y to x */
113     if (ix >= -1022)
114         hx = 0x00100000 | (0x000ffff & hx);
115     else { /* subnormal x, shift x to normal */
116         n = -1022 - ix;
117         if (n <= 31) {
118             hx = (hx << n) | (lx >> (32 - n));
119             lx <<= n;
120         } else {
121             hx = lx << (n - 32);
122             lx = 0;

```

```

123     }
124   }
125   if (iy >= -1022)
126     hy = 0x00100000 | (0x000fffff & hy);
127   else { /* subnormal y, shift y to normal */
128     n = -1022 - iy;
129     if (n <= 31) {
130       hy = (hy << n) | (ly >> (32 - n));
131       ly <<= n;
132     } else {
133       hy = ly << (n - 32);
134       ly = 0;
135     }
136   }

138   /* fix point fmod */
139   n = ix - iy;
140   m = 0;
141   while (n-- > 0) {
142     hz = hx - hy;
143     lz = lx - ly;
144     if (lx < ly)
145       hz -= 1;
146     if (hz < 0) {
147       hx = hx + hx + (lx >> 31);
148       lx = lx + lx;
149     } else {
150       m += 1;
151       if ((hz | lz) == 0) { /* return sign(x)*0 */
152         if (n < 31)
153           m <<= 1 + n;
154         else
155           m = 0;
156         m &= 0x7fffffff;
157         *quo = sq >= 0 ? m : -m;
158         return (Zero[(unsigned) sx >> 31]);
159       }
160       hx = hz + hz + (lz >> 31);
161       lx = lz + lz;
162     }
163     m += m;
164   }
165   hz = hx - hy;
166   lz = lx - ly;
167   if (lx < ly)
168     hz -= 1;
169   if (hz >= 0) {
170     hx = hz;
171     lx = lz;
172     m += 1;
173   }
174   m &= 0x7fffffff;
175   *quo = sq >= 0 ? m : -m;

177   /* convert back to floating value and restore the sign */
178   if ((hx | lx) == 0) { /* return sign(x)*0 */
179     return (Zero[(unsigned) sx >> 31]);
180   }
181   while (hx < 0x00100000) { /* normalize x */
182     hx = hx + hx + (lx >> 31);
183     lx = lx + lx;
184     iy -= 1;
185   }
186   if (iy >= -1022) { /* normalize output */
187     hx = (hx - 0x00100000) | ((iy + 1023) << 20);
188     __HI(x) = hx | sx;

```

```

189     __LO(x) = lx;
190   } else { /* subnormal output */
191     n = -1022 - iy;
192     if (n <= 20) {
193       lx = (lx >> n) | ((unsigned) hx << (32 - n));
194       hx >>= n;
195     } else if (n <= 31) {
196       lx = (hx << (32 - n)) | (lx >> n);
197       hx = sx;
198     } else {
199       lx = hx >> (n - 32);
200       hx = sx;
201     }
202     __HI(x) = hx | sx;
203     __LO(x) = lx;
204     x *= one; /* create necessary signal */
205   }
206   return (x); /* exact output */
207 }

```

unchanged_portion_omitted

```

*****
5166 Tue Nov 25 12:58:59 2014
new/usr/src/lib/libm/common/m9x/remquof.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
30 #pragma weak __remquof = remquof
30 #pragma weak remquof = __remquof
32 /* INDENT OFF */
33 /*
34  * float remquof(float x, float y, int *quo) return remainderf(x,y) and an
35  * integer pointer quo such that *quo = N mod (2**31), where N is the
36  * exact integral part of x/y rounded to nearest even.
37  *
38  * remquof call internal fmodquof
39  */
41 #include "libm.h"
42 #include "libm_synonyms.h"
42 #include "libm_protos.h"
43 #include <math.h>
44 extern float fabsf(float);
46 static const int
47     is = (int) 0x80000000,
48     im = 0x007fffff,
49     ii = 0x7f800000,
50     iu = 0x00800000;
52 static const float zero = 0.0F, half = 0.5F;
53 /* INDENT ON */
55 static float
56 fmodquof(float x, float y, int *quo) {

```

```

57     float w;
58     int hx, ix, iy, iz, k, ny, nd, m, sq;
60     hx = *(int *) &x;
61     ix = hx & 0x7fffffff;
62     iy = *(int *) &y;
63     sq = (iy ^ hx) & is; /* sign of x/y */
64     iy &= 0x7fffffff;
66     /* purge off exception values */
67     *quo = 0;
68     if (ix >= ii || iy > ii || iy == 0) {
69         w = x * y;
70         w = w / w;
71     } else if (ix <= iy) {
72         if (ix < iy)
73             w = x; /* return x if |x|<|y| */
74         else {
75             *quo = 1 + (sq >> 30);
76             w = zero * x; /* return sign(x)*0.0 */
77         }
78     } else {
79         /* INDENT OFF */
80         /*
81          * scale x,y to "normal" with
82          * ny = exponent of y
83          * nd = exponent of x minus exponent of y
84          */
85         /* INDENT ON */
86         ny = iy >> 23;
87         k = ix >> 23;
89         /* special case for subnormal y or x */
90         if (ny == 0) {
91             ny = 1;
92             while (iy < iu) {
93                 ny -= 1;
94                 iy += iy;
95             }
96             nd = k - ny;
97             if (k == 0) {
98                 nd += 1;
99                 while (ix < iu) {
100                     nd -= 1;
101                     ix += ix;
102                 }
103             } else
104                 ix = iu | (ix & im);
105         } else {
106             nd = k - ny;
107             ix = iu | (ix & im);
108             iy = iu | (iy & im);
109         }
110         /* INDENT OFF */
111         /* fix point fmod for normalized ix and iy */
112         /*
113          * while (nd--) {
114          *     iz = ix - iy;
115          *     if (iz < 0)
116          *         ix = ix + ix;
117          *     else if (iz == 0) {
118          *         *(int *) &w = is & hx;
119          *         return w;
120          *     } else
121          *         ix = iz + iz;
122          * }

```

```

123     */
124     /* INDENT ON */
125     /* unroll the above loop 4 times to gain performance */
126     m = 0;
127     k = nd >> 2;
128     nd -= (k << 2);
129     while (k--) {
130         iz = ix - iy;
131         if (iz >= 0) {
132             m += 1;
133             ix = iz + iz;
134         } else
135             ix += ix;
136         m += m;
137         iz = ix - iy;
138         if (iz >= 0) {
139             m += 1;
140             ix = iz + iz;
141         } else
142             ix += ix;
143         m += m;
144         iz = ix - iy;
145         if (iz >= 0) {
146             m += 1;
147             ix = iz + iz;
148         } else
149             ix += ix;
150         m += m;
151         iz = ix - iy;
152         if (iz >= 0) {
153             m += 1;
154             ix = iz + iz;
155         } else
156             ix += ix;
157         m += m;
158         if (iz == 0) {
159             iz = (k << 2) + nd;
160             if (iz < 32)
161                 m <<= iz;
162             else
163                 m = 0;
164             m &= 0x7fffffff;
165             *quo = sq >= 0 ? m : -m;
166             *(int *) &w = is & hx;
167             return (w);
168         }
169     }
170     while (nd--) {
171         iz = ix - iy;
172         if (iz >= 0) {
173             m += 1;
174             ix = iz + iz;
175         } else
176             ix += ix;
177         m += m;
178     }
179     /* end of unrolling */

181     iz = ix - iy;
182     if (iz >= 0) {
183         m += 1;
184         ix = iz;
185     }
186     m &= 0x7fffffff;
187     *quo = sq >= 0 ? m : -m;

```

```

189     /* convert back to floating value and restore the sign */
190     if (ix == 0) {
191         *(int *) &w = is & hx;
192         return (w);
193     }
194     while (ix < iu) {
195         ix += ix;
196         ny -= 1;
197     }
198     while (ix > (iu + iu)) {
199         ny += 1;
200         ix >>= 1;
201     }
202     if (ny > 0)
203         *(int *) &w = (is & hx) | (ix & im) | (ny << 23);
204     else { /* subnormal output */
205         k = -ny + 1;
206         ix >>= k;
207         *(int *) &w = (is & hx) | ix;
208     }
209     }
210     return (w);
211 }

```

unchanged portion omitted


```

*****
6845 Tue Nov 25 12:58:59 2014
new/usr/src/lib/libm/common/m9x/remquo1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __remquo1 = remquo1
30 #pragma weak remquo1 = __remquo1
31
32 #include "libm.h"
33 #include "libm_synonyms.h"
34 #if defined(__SUNPRO_C)
35 #include <sunmath.h> /* fabs1 */
36 #endif
37 /* INDENT OFF */
38 static const int
39     is = -0x7fffffff - 1,
40     im = 0x0000ffff,
41     iu = 0x00010000;
42
43 static const long double zero = 0.0L, one = 1.0L;
44 /* INDENT ON */
45
46 #if defined(__sparc)
47 #define __H0(x) ((int *) &x)[0]
48 #define __H1(x) ((int *) &x)[1]
49 #define __H2(x) ((int *) &x)[2]
50 #define __H3(x) ((int *) &x)[3]
51 #else
52 #error Unsupported architecture
53 #endif
54
55 /*
56  * On entrance: *quo is initialized to 0, x finite and y non-zero & ordered

```

```

57 static long double
58 fmodquo1(long double x, long double y, int *quo) {
59     long double a, b;
60     int n, ix, iy, k, sx, sq, m;
61     int hx;
62     int x0, y0, z0, carry;
63     unsigned xl, x2, x3, y1, y2, y3, z1, z2, z3;
64
65     hx = __H0(x);
66     x1 = __H1(x);
67     x2 = __H2(x);
68     x3 = __H3(x);
69     y0 = __H0(y);
70     y1 = __H1(y);
71     y2 = __H2(y);
72     y3 = __H3(y);
73
74     sx = hx & is;
75     sq = (hx ^ y0) & is;
76     x0 = hx ^ sx;
77     y0 &= ~0x80000000;
78
79     a = fabs1(x);
80     b = fabs1(y);
81     if (a <= b) {
82         if (a < b)
83             return (x);
84         else {
85             *quo = 1 + (sq >> 30);
86             return (zero * x);
87         }
88     }
89     /* determine ix = ilogbl(x) */
90     if (x0 < iu) { /* subnormal x */
91         ix = 0;
92         ix = -16382;
93         while (x0 == 0) {
94             ix -= 16;
95             x0 = x1 >> 16;
96             x1 = (x1 << 16) | (x2 >> 16);
97             x2 = (x2 << 16) | (x3 >> 16);
98             x3 = (x3 << 16);
99         }
100        while (x0 < iu) {
101            ix -= 1;
102            x0 = (x0 << 1) | (x1 >> 31);
103            x1 = (x1 << 1) | (x2 >> 31);
104            x2 = (x2 << 1) | (x3 >> 31);
105            x3 <<= 1;
106        }
107    } else {
108        ix = (x0 >> 16) - 16383;
109        x0 = iu | (x0 & im);
110    }
111
112     /* determine iy = ilogbl(y) */
113     if (y0 < iu) { /* subnormal y */
114         iy = -16382;
115         while (y0 == 0) {
116             iy -= 16;
117             y0 = y1 >> 16;
118             y1 = (y1 << 16) | (y2 >> 16);
119             y2 = (y2 << 16) | (y3 >> 16);
120             y3 = (y3 << 16);
121         }
122        while (y0 < iu) {

```

```

123         iy -= 1;
124         y0 = (y0 << 1) | (y1 >> 31);
125         y1 = (y1 << 1) | (y2 >> 31);
126         y2 = (y2 << 1) | (y3 >> 31);
127         y3 <<= 1;
128     }
129 } else {
130     iy = (y0 >> 16) - 16383;
131     y0 = iu | (y0 & im);
132 }

135 /* fix point fmod */
136 n = ix - iy;
137 m = 0;
138 while (n--) {
139     while (x0 == 0 && n >= 16) {
140         m <<= 16;
141         n -= 16;
142         x0 = x1 >> 16;
143         x1 = (x1 << 16) | (x2 >> 16);
144         x2 = (x2 << 16) | (x3 >> 16);
145         x3 = (x3 << 16);
146     }
147     while (x0 < iu && n >= 1) {
148         m += m;
149         n -= 1;
150         x0 = (x0 << 1) | (x1 >> 31);
151         x1 = (x1 << 1) | (x2 >> 31);
152         x2 = (x2 << 1) | (x3 >> 31);
153         x3 = (x3 << 1);
154     }
155     carry = 0;
156     z3 = x3 - y3;
157     carry = z3 > x3;
158     if (carry == 0) {
159         z2 = x2 - y2;
160         carry = z2 > x2;
161     } else {
162         z2 = x2 - y2 - 1;
163         carry = z2 >= x2;
164     }
165     if (carry == 0) {
166         z1 = x1 - y1;
167         carry = z1 > x1;
168     } else {
169         z1 = x1 - y1 - 1;
170         carry = z1 >= x1;
171     }
172     z0 = x0 - y0 - carry;
173     if (z0 < 0) { /* double x */
174         x0 = x0 + x0 + ((x1 & is) != 0);
175         x1 = x1 + x1 + ((x2 & is) != 0);
176         x2 = x2 + x2 + ((x3 & is) != 0);
177         x3 = x3 + x3;
178         m += m;
179     } else {
180         m += 1;
181         if (z0 == 0) {
182             if ((z1 | z2 | z3) == 0) {
183                 /* 0: we are done */
184                 if (n < 31)
185                     m <<= (1 + n);
186                 else
187                     m = 0;
188                 m &= ~0x80000000;

```

```

189         *quo = sq >= 0 ? m : -m;
190         __H0(a) = hx & is;
191         __H1(a) = __H2(a) = __H3(a) = 0;
192         return (a);
193     }
194 }
195 /* x = z << 1 */
196 z0 = z0 + z0 + ((z1 & is) != 0);
197 z1 = z1 + z1 + ((z2 & is) != 0);
198 z2 = z2 + z2 + ((z3 & is) != 0);
199 z3 = z3 + z3;
200 x0 = z0;
201 x1 = z1;
202 x2 = z2;
203 x3 = z3;
204 m += m;
205 }
206 }
207 carry = 0;
208 z3 = x3 - y3;
209 carry = z3 > x3;
210 if (carry == 0) {
211     z2 = x2 - y2;
212     carry = z2 > x2;
213 } else {
214     z2 = x2 - y2 - 1;
215     carry = z2 >= x2;
216 }
217 if (carry == 0) {
218     z1 = x1 - y1;
219     carry = z1 > x1;
220 } else {
221     z1 = x1 - y1 - 1;
222     carry = z1 >= x1;
223 }
224 z0 = x0 - y0 - carry;
225 if (z0 >= 0) {
226     x0 = z0;
227     x1 = z1;
228     x2 = z2;
229     x3 = z3;
230     m += 1;
231 }
232 m &= ~0x80000000;
233 *quo = sq >= 0 ? m : -m;

235 /* convert back to floating value and restore the sign */
236 if ((x0 | x1 | x2 | x3) == 0) {
237     __H0(a) = hx & is;
238     __H1(a) = __H2(a) = __H3(a) = 0;
239     return (a);
240 }
241 while (x0 < iu) {
242     if (x0 == 0) {
243         iy -= 16;
244         x0 = x1 >> 16;
245         x1 = (x1 << 16) | (x2 >> 16);
246         x2 = (x2 << 16) | (x3 >> 16);
247         x3 = (x3 << 16);
248     } else {
249         x0 = x0 + x0 + ((x1 & is) != 0);
250         x1 = x1 + x1 + ((x2 & is) != 0);
251         x2 = x2 + x2 + ((x3 & is) != 0);
252         x3 = x3 + x3;
253         iy -= 1;
254     }

```

```
255     }
256
257     /* normalize output */
258     if (iy >= -16382) {
259         __H0(a) = sx | (x0 - iy) | ((iy + 16383) << 16);
260         __H1(a) = x1;
261         __H2(a) = x2;
262         __H3(a) = x3;
263     } else { /* subnormal output */
264         n = -16382 - iy;
265         k = n & 31;
266         if (k <= 16) {
267             x3 = (x2 << (32 - k)) | (x3 >> k);
268             x2 = (x1 << (32 - k)) | (x2 >> k);
269             x1 = (x0 << (32 - k)) | (x1 >> k);
270             x0 >>= k;
271         } else {
272             x3 = (x2 << (32 - k)) | (x3 >> k);
273             x2 = (x1 << (32 - k)) | (x2 >> k);
274             x1 = (x0 << (32 - k)) | (x1 >> k);
275             x0 = 0;
276         }
277         while (n >= 32) {
278             n -= 32;
279             x3 = x2;
280             x2 = x1;
281             x1 = x0;
282             x0 = 0;
283         }
284         __H0(a) = x0 | sx;
285         __H1(a) = x1;
286         __H2(a) = x2;
287         __H3(a) = x3;
288         a *= one;
289     }
290     return (a);
291 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/round.c

1

```
*****
1921 Tue Nov 25 12:59:00 2014
new/usr/src/lib/libm/common/m9x/round.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __round = round
30 #if defined(ELFOBJ)
31 #pragma weak round = __round
32 #endif

32 #include "libm.h"

34 double
35 round(double x) {
36     union {
37         unsigned i[2];
38         double d;
39     } xx;
40     unsigned hx, sx, i;

42     xx.d = x;
43     hx = xx.i[HIWORD] & ~0x80000000;
44     sx = xx.i[HIWORD] & 0x80000000;
45     if (hx < 0x43300000) { /* |x| < 2^52 */
46         if (hx < 0x3ff00000) { /* |x| < 1 */
47             if (hx >= 0x3fe00000)
48                 return (sx ? -1.0 : 1.0);
49             return (sx ? -0.0 : 0.0);

```

new/usr/src/lib/libm/common/m9x/round.c

2

```
50     }

52     /* round x at the integer bit */
53     if (hx < 0x41300000) {
54         i = 1 << (0x412 - (hx >> 20));
55         xx.i[HIWORD] = (xx.i[HIWORD] + i) & ~(i | (i - 1));
56         xx.i[LOWORD] = 0;
57     } else {
58         i = 1 << (0x432 - (hx >> 20));
59         xx.i[LOWORD] += i;
60         if (xx.i[LOWORD] < i)
61             xx.i[HIWORD]++;
62         xx.i[LOWORD] &= ~(i | (i - 1));
63     }
64     return (xx.d);
65 } else if (hx < 0x7ff00000)
66     return (x);
67 else
68 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
69     return (hx >= 0x7ff80000 ? x : x + x);
70 /* assumes sparc-like QNaN */
71 #else
72     return (x + x);
73 #endif
74 }

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/common/m9x/roundf.c

1

```
*****
1686 Tue Nov 25 12:59:00 2014
new/usr/src/lib/libm/common/m9x/roundf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __roundf = roundf
30 #if defined(ELFOSBJ)
31 #pragma weak roundf = __roundf
32 #endif

32 #include "libm.h"

34 float
35 roundf(float x) {
36     union {
37         unsigned i;
38         float f;
39     } xx;
40     unsigned hx, sx, i;

42     xx.f = x;
43     hx = xx.i & ~0x80000000;
44     sx = xx.i & 0x80000000;
45     if (hx < 0x4b000000) { /* |x| < 2^23 */
46         if (hx < 0x3f800000) { /* |x| < 1 */
47             if (hx >= 0x3f000000)
48                 return (sx ? -1.0F : 1.0F);
49             return (sx ? -0.0F : 0.0F);

```

new/usr/src/lib/libm/common/m9x/roundf.c

2

```
50     }

52     /* round x at the integer bit */
53     i = 1 << (0x95 - (hx >> 23));
54     xx.i = (xx.i + i) & ~((i << 1) - 1);
55     return (xx.f);
56     } else if (hx < 0x7f800000) /* |x| is integral */
57         return (x);
58     else
59 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
60         return (hx > 0x7f800000 ? x * x : x + x);
61 #else
62         return (x + x);
63 #endif
64 }

_____unchanged_portion_omitted_____
```

```

*****
3658 Tue Nov 25 12:59:00 2014
new/usr/src/lib/libm/common/m9x/roundl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */

30 #pragma weak __roundl = roundl
30 #if defined(ELFOBJ)
31 #pragma weak roundl = __roundl
32 #endif

32 #include "libm.h"

34 #if defined(__sparc)
35 long double
36 roundl(long double x) {
37     union {
38         unsigned i[4];
39         long double q;
40     } xx;
41     unsigned hx, sx, v;
42     int j;

44     xx.q = x;
45     sx = xx.i[0] & 0x80000000;
46     hx = xx.i[0] & ~0x80000000;

48     /* handle trivial cases */
49     if (hx >= 0x406f0000) /* |x| >= 2^112 + ... or x is nan */

```

```

50     return (hx >= 0x7fff0000 ? x + x : x);

52     /* handle |x| < 1 */
53     if (hx < 0x3fff0000) {
54         if (hx >= 0x3ffe0000)
55             return (sx ? -1.0L : 1.0L);
56         return (sx ? -0.0L : 0.0L);
57     }

59     xx.i[0] = hx;
60     j = 0x406f - (hx >> 16); /* 1 <= j <= 112 */
61     if (j >= 96) { /* 96 <= j <= 112 */
62         v = (1U << (j - 96)) >> 1;
63         if (v) {
64             if (xx.i[0] & v)
65                 xx.i[0] += v;
66             xx.i[0] &= ~(v - 1);
67         } else if (xx.i[1] & 0x80000000)
68             ++xx.i[0];
69         xx.i[1] = xx.i[2] = xx.i[3] = 0;
70     } else if (j >= 64) { /* 64 <= j <= 95 */
71         v = (1U << (j - 64)) >> 1;
72         if (v) {
73             if (xx.i[1] & v) {
74                 xx.i[1] += v;
75                 if (xx.i[1] < v)
76                     ++xx.i[0];
77             }
78             xx.i[1] &= ~(v - 1);
79         } else if (xx.i[2] & 0x80000000) {
80             if (++xx.i[1] == 0)
81                 ++xx.i[0];
82         }
83         xx.i[2] = xx.i[3] = 0;
84     } else if (j >= 32) { /* 32 <= j <= 63 */
85         v = (1U << (j - 32)) >> 1;
86         if (v) {
87             if (xx.i[2] & v) {
88                 xx.i[2] += v;
89                 if (xx.i[2] < v) {
90                     if (++xx.i[1] == 0)
91                         ++xx.i[0];
92                 }
93             }
94             xx.i[2] &= ~(v - 1);
95         } else if (xx.i[3] & 0x80000000) {
96             if (++xx.i[2] == 0) {
97                 if (++xx.i[1] == 0)
98                     ++xx.i[0];
99             }
100         }
101         xx.i[3] = 0;
102     } else { /* 1 <= j <= 31 */
103         v = 1U << (j - 1);
104         if (xx.i[3] & v) {
105             xx.i[3] += v;
106             if (xx.i[3] < v) {
107                 if (++xx.i[2] == 0) {
108                     if (++xx.i[1] == 0)
109                         ++xx.i[0];
110                 }
111             }
112         }
113         xx.i[3] &= ~(v - 1);
114     }

```

new/usr/src/lib/libm/common/m9x/roundl.c

3

```
116     /* negate result if need be */
117     if (sx)
118         xx.i[0] |= 0x80000000;
119     return (xx.q);
120 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/scalbln.c

1

```
*****
2659 Tue Nov 25 12:59:01 2014
new/usr/src/lib/libm/common/m9x/scalbln.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __scalbln = scalbln
30 #if defined(ELFOBJ)
31 #pragma weak scalbln = __scalbln
32 #endif

32 #include "libm.h"
33 #include <float.h>          /* DBL_MAX, DBL_MIN */

35 static const double twom54 = 5.5511151231257827021181583404541015625e-17;
36 #if defined(__x86)
37 #if defined(USE_FPSCALE) || defined(__x86)
37 static const double two52 = 4503599627370496.0;
38 #else
39 /*
40  * Normalize non-zero subnormal x and return biased exponent of x in [-51,0]
41  */
42 static int
43 ilogb_biased(unsigned *px) {
44     int s = 52;
45     unsigned v = px[HIWORD] & ~0x80000000, w = px[LOWORD], t = v;

47     if (t)
48         s -= 32;
```

new/usr/src/lib/libm/common/m9x/scalbln.c

2

```
49     else
50         t = w;
51     if (t & 0xffff0000)
52         s -= 16, t >>= 16;
53     if (t & 0xff00)
54         s -= 8, t >>= 8;
55     if (t & 0xf0)
56         s -= 4, t >>= 4;
57     t <<= 1;
58     s -= (0xffffaa50 >> t) & 0x3;
59     if (s < 32) {
60         v = (v << s) | w >> (32 - s);
61         w <<= s;
62     } else {
63         v = w << (s - 32);
64         w = 0;
65     }
66     px[HIWORD] = (px[HIWORD] & 0x80000000) | v;
67     px[LOWORD] = w;
68     return (1 - s);
69 }
70 #endif /* defined(__x86) */
72 #endif /* defined(USE_FPSCALE) */

72 double
73 scalbln(double x, long n) {
74     int *px = (int *) &x, ix, k;

76     ix = px[HIWORD] & ~0x80000000;
77     k = ix >> 20;
78     if (k == 0x7ff)
79 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
80         return ((px[HIWORD] & 0x800000) != 0 ? x : x + x);
81         /* assumes sparc-like QNaN */
82 #else
83         return (x + x);
84 #endif
85     if ((px[LOWORD] | ix) == 0 || n == 0)
86         return (x);
87     if (k == 0) {
88 #if defined(__x86)
89 #if defined(USE_FPSCALE) || defined(__x86)
90         x *= two52;
91         k = ((px[HIWORD] & ~0x80000000) >> 20) - 52;
92     }
93 #endif
94     k = ilogb_biased((unsigned *) px);
95     k += (int) n;
96     if (n > 5000 || k > 0x7fe)
97         return (DBL_MAX * copysign(DBL_MAX, x));
98     if (n < -5000 || k <= -54)
99         return (DBL_MIN * copysign(DBL_MIN, x));
100     if (k > 0) {
101         px[HIWORD] = (px[HIWORD] & ~0x7ff00000) | (k << 20);
102         return (x);
103     }
104     k += 54;
105     px[HIWORD] = (px[HIWORD] & ~0x7ff00000) | (k << 20);
106     return (x * twom54);
107 }

_____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/common/m9x/scalblnf.c

1

```
*****
2273 Tue Nov 25 12:59:01 2014
new/usr/src/lib/libm/common/m9x/scalblnf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24  */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28  */
29
30 #pragma weak __scalblnf = scalblnf
30 #if defined(ELFOBJ)
31 #pragma weak scalblnf = __scalblnf
32 #endif
33
34 #include "libm.h"
35 #include <float.h>          /* FLT_MAX, FLT_MIN */
36
37 static const float twom25f = 2.98023223876953125e-8F;
36 #if defined(__x86)
38 #if defined(USE_FPSCALE) || defined(__x86)
37 static const float two23f = 8388608.0F;
38 #else
39 /*
40  * v: a non-zero subnormal |x|; returns [-22, 0]
41  */
42 static int
43 ilogbf_biased(unsigned v) {
44     int r = -22;
45
46     if (v & 0xffff0000)
47         r += 16, v >>= 16;
48     if (v & 0xff00)
```

new/usr/src/lib/libm/common/m9x/scalblnf.c

2

```
49         r += 8, v >>= 8;
50     if (v & 0xf0)
51         r += 4, v >>= 4;
52     v <<= 1;
53     return (r + ((0xffffaa50 >> v) & 0x3));
54 }
55 #endif /* defined(__x86) */
56 #endif /* defined(USE_FPSCALE) */
57
58 float
59 scalblnf(float x, long n) {
60     int *px = (int *) &x, ix, k;
61
62     ix = *px & ~0x80000000;
63     k = ix >> 23;
64     if (k == 0xff)
65         return (ix > 0x7f800000 ? x * x : x);
66 #else
67     return (x + x);
68 #endif
69     if (ix == 0 || n == 0)
70         return (x);
71     if (k == 0) {
72 #if defined(__x86)
74 #if defined(USE_FPSCALE) || defined(__x86)
73         x *= two23f;
74         k = ((*px & ~0x80000000) >> 23) - 23;
75 #else
76         k = ilogbf_biased(ix);
77         *px = (*px & 0x80000000) | (ix << (-k + 1));
78 #endif
79     }
80     k += (int) n;
81     if (n > 5000 || k > 0xfe)
82         return (FLT_MAX * copysignf(FLT_MAX, x));
83     if (n < -5000 || k <= -25)
84         return (FLT_MIN * copysignf(FLT_MIN, x));
85     if (k > 0) {
86         *px = (*px & ~0x7f800000) | (k << 23);
87         return (x);
88     }
89     k += 25;
90     *px = (*px & ~0x7f800000) | (k << 23);
91     return (x * twom25f);
92 }
93
94 _____
95 unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/scalblnl.c

1

```
*****
2404 Tue Nov 25 12:59:02 2014
new/usr/src/lib/libm/common/m9x/scalblnl.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __scalblnl = scalblnl
30 #if defined(ELFOSBJ)
31 #pragma weak scalblnl = __scalblnl
32 #endif

32 #include "libm.h"
33 #include <float.h>          /* LDBL_MAX, LDBL_MIN */

35 #if defined(__sparc)
36 #define XSET_EXP(k, x)  (((int *) &x)[0] = (((int *) &x)[0] & ~0x7fff0000) | \
37                        (k << 16))
38 #define ISINFNANL(k, x) (k == 0x7fff)
39 #define XTWOT_OFFSET    113
40 static const long double xtwtot = 10384593717069655257060992658440192.0L,
41                    twomtml = 4.814824860968089632639944856462318296E-35L; /* 2^113 */
42 #elif defined(__x86)
43 #define XSET_EXP(k, x)  (((int *) &x)[2] = (((int *) &x)[2] & ~0x7fff) | k
44 #if defined(HANDLE_UNSUPPORTED)
45 #define ISINFNANL(k, x) (k == 0x7fff || \
46                        (k != 0 && (((int *) &x)[1] & 0x80000000) == 0))
47 #else
48 #define ISINFNANL(k, x) (k == 0x7fff)
49 #endif

```

new/usr/src/lib/libm/common/m9x/scalblnl.c

2

```
50 #endif
51 #define XTWOT_OFFSET    64
52 static const long double xtwtot = 18446744073709551616.0L, /* 2^64 */
53                    twomtml = 2.7105054312137610850186E-20L; /* 2^-65 */
54 #endif

56 long double
57 scalblnl(long double x, long n) {
58     int k = XBIASED_EXP(x);

60     if (ISINFNANL(k, x))
61         return (x + x);
62     if (ISZEROL(x) || n == 0)
63         return (x);
64     if (k == 0) {
65         x *= xtwtot;
66         k = XBIASED_EXP(x) - XTWOT_OFFSET;
67     }
68     k += (int) n;
69     if (n > 50000 || k > 0x7ffe)
70         return (LDBL_MAX * copysignl(LDBL_MAX, x));
71     if (n < -50000 || k <= -XTWOT_OFFSET - 1)
72         return (LDBL_MIN * copysignl(LDBL_MIN, x));
73     if (k > 0) {
74         XSET_EXP(k, x);
75         return (x);
76     }
77     k += XTWOT_OFFSET + 1;
78     XSET_EXP(k, x);
79     return (x * twomtml);
80 }

unchanged portion omitted

```

```

*****
68964 Tue Nov 25 12:59:02 2014
new/usr/src/lib/libm/common/m9x/tgamma.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __tgamma = tgamma
30 #if defined(ELFOBJ)
31 #pragma weak tgamma = __tgamma
32 #endif

32 /* INDENT OFF */
33 /*
34  * True gamma function
35  * double tgamma(double x)
36  *
37  * Error:
38  * -----
39  * Less than one ulp for both positive and negative arguments.
40  *
41  * Algorithm:
42  * -----
43  * A: For negative argument
44  * (1) gamma(-n or -inf) is NaN
45  * (2) Underflow Threshold
46  * (3) Reduction to gamma(1+x)
47  * B: For x between 1 and 2
48  * C: For x between 0 and 1
49  * D: For x between 2 and 8

```

```

50 * E: Overflow threshold {see over.c}
51 * F: For overflow_threshold >= x >= 8
52 *
53 * Implementation details
54 * -----
55 *
56 * (A) For negative argument, use gamma(-x) = -----
57 * (sin(pi*x)*gamma(1+x))
58 *
59 * (1) gamma(-n or -inf) is NaN with invalid signal by SUSv3 spec.
60 * (Ideally, gamma(-n) = 1/sinpi(n) = (-1)**(n+1) * inf.)
61 *
62 * (2) Underflow Threshold. For each precision, there is a value T
63 * such that when x>T and when x is not an integer, gamma(-x) will
64 * always underflow. A table of the underflow threshold value is given
65 * below. For proof, see file "under.c".
66 *
67 * Precision underflow threshold T =
68 * -----
69 * single 41.000041962 = 41 + 11 ULP
70 * (machine format) 4224000B
71 * double 183.00000000000312639 = 183 + 11 ULP
72 * (machine format) 4066E000 0000000B
73 * quad 1774.000000000000000000000000000017749370 = 1774 + 9 ULP
74 * (machine format) 4009BB800000000000000000000000009
75 * -----
76 *
77 * (3) Reduction to gamma(1+x).
78 * Because of (1) and (2), we need only consider non-integral x
79 * such that 0<x<T. Let k = [x] and z = x-[x]. Define
80 * sin(x*pi) cos(x*pi)
81 * kpsin(x) = ----- and kpcos(x) = ----- . Then
82 * pi pi
83 *
84 * gamma(-x) = -----
85 * -kpsin(x)*gamma(1+x)
86 *
87 * Since x = k+z,
88 * -sin(x*pi) = -sin(k*pi+z*pi) = (-1)^(k+1) * sin(z*pi),
89 * k+1
90 * we have -kpsin(x) = (-1) * kpsin(z). We can further
91 * reduce z to t by
92 * (I) t = z when 0.00000 <= z < 0.31830...
93 * (II) t = 0.5-z when 0.31830... <= z < 0.681690...
94 * (III) t = 1-z when 0.681690... <= z < 1.00000
95 * and correspondingly
96 * (I) kpsin(z) = kpsin(t) ... 0<= z < 0.3184
97 * (II) kpsin(z) = kpcos(t) ... |t| < 0.182
98 * (III) kpsin(z) = kpsin(t) ... 0<= t < 0.3184
99 *
100 * Using a special Remez algorithm, we obtain the following polynomial
101 * approximation for kpsin(t) for 0<=t<0.3184:
102 *
103 * Computation note: in simulating higher precision arithmetic, kcpsin
104 * return head = t and tail = ks[0]*t^3 + (...) to maintain extra bits.
105 *
106 * Quad precision, remez error <= 2**(-129.74)
107 * 3 5 27
108 * kpsin(t) = t + ks[0] * t + ks[1] * t + ... + ks[12] * t
109 *
110 * ks[ 0] = -1.64493406684822643647241516664602518705158902870e+0000
111 * ks[ 1] = 8.11742425283353643637002772405874238094995726160e-0001
112 * ks[ 2] = -1.90751824122084213696472111835337366232282723933e-0001
113 * ks[ 3] = 2.61478478176548005046532613563241288115395517084e-0002
114 * ks[ 4] = -2.34608103545582363750893072647117829448016479971e-0003
115 * ks[ 5] = 1.48428793031071003684606647212534027556262040158e-0004

```

```

116 *   ks[ 6] = -6.97587366165638046518462722252768122615952898698e-0006
117 *   ks[ 7] =  2.53121740413702536928659271747187500934840057929e-0007
118 *   ks[ 8] = -7.30471182221385990397683641695766121301933621956e-0009
119 *   ks[ 9] =  1.71653847451163495739958249695549313987973589884e-0010
120 *   ks[10] = -3.34813314714560776122245796929054813458341420565e-0012
121 *   ks[11] =  5.507249922622033449487808306969135431411753047e-0014
122 *   ks[12] = -7.67678132753577998601234393215802221104236979928e-0016
123 *
124 * Double precision, Remez error <= 2**(-62.9)
125 *           3           5           15
126 *   kpsin(t) = t + ks[0] * t + ks[1] * t + ... + ks[6] * t
127 *
128 *   ks[0] = -1.644934066848226406065691 (0x3ffa51a6 625307d3)
129 *   ks[1] =  8.11742425283341655883668741874008920850698590621e-0001
130 *   ks[2] = -1.90751824120862873825597279118304943994042258291e-0001
131 *   ks[3] =  2.6147847763255427831728962832654539353521911570e-0002
132 *   ks[4] = -2.3460797851020271037761719027873552535437705866e-0003
133 *   ks[5] =  1.48413292290051695897242899977121846763824221705e-0004
134 *   ks[6] = -6.8773076963754348810868872677687262485357072242e-0006
135 *
136 * Single precision, Remez error <= 2**(-34.09)
137 *           3           5           9
138 *   kpsin(t) = t + ks[0] * t + ks[1] * t + ... + ks[3] * t
139 *
140 *   ks[0] = -1.64493404985645811354476665052005342839447790544e+0000
141 *   ks[1] =  8.11740794458351064092797249069438269367389272270e-0001
142 *   ks[2] = -1.90703144603551216933075809162889536878854055202e-0001
143 *   ks[3] =  2.55742333994264563281155312271481108635575331201e-0002
144 *
145 * Computation note: in simulating higher precision arithmetic, kpcsin
146 * return head = t and tail = kc[0]*t^3 + (...) to maintain extra bits
147 * precision.
148 *
149 * And for kpcos(t) for |t| < 0.183:
150 *
151 * Quad precision, remez <= 2**(-122.48)
152 *           2           4           22
153 *   kpcos(t) = 1/pi + pi/2 * t + kc[2] * t + ... + kc[11] * t
154 *
155 *   kc[2] =  1.29192819501249250731151312779548918765320728489e+0000
156 *   kc[3] = -4.25027339979557573976029596929319207009444090366e-0001
157 *   kc[4] =  7.4908066165099096109672954618317623888421628613e-0002
158 *   kc[5] = -8.21458866111282287985539464173976555436050215120e-0003
159 *   kc[6] =  6.14202578809529228503205255165761204750211603402e-0004
160 *   kc[7] = -3.33073432691149607007217330302595267179545908740e-0005
161 *   kc[8] =  1.36970959047832085796809745461530865597993680204e-0006
162 *   kc[9] = -4.41780774262583514450246512727201806217271097336e-0008
163 *   kc[10] =  1.14741409212381858820016567664488123478660705759e-0009
164 *   kc[11] = -2.44261236114707374558437500654381006300502749632e-0011
165 *
166 * Double precision, remez < 2**(-61.91)
167 *           2           4           12
168 *   kpcos(t) = 1/pi + pi/2 * t + kc[2] * t + ... + kc[6] * t
169 *
170 *   kc[2] =  1.29192819501230224953283586722575766189551966008e+0000
171 *   kc[3] = -4.25027339940149518500158850753393173519732149213e-0001
172 *   kc[4] =  7.49080625187015312373925142219429422375556727752e-0002
173 *   kc[5] = -8.21442040906099210866977352284054849051348692715e-0003
174 *   kc[6] =  6.1041135682951541457556656473363253233904115968e-0004
175 *
176 * Single precision, remez < 2**(-30.13)
177 *           2           6
178 *   kpcos(t) = kc[0] + kc[1] * t + ... + kc[3] * t
179 *
180 *   kc[0] =  3.18309886183790671537767526745028724068919291480e-0001
181 *   kc[1] = -1.57079581447762568199467875065854538626594937791e+0000

```

```

182 *   kc[2] =  1.29183528092558692844073004029568674027807393862e+0000
183 *   kc[3] = -4.20232949771307685981015914425195471602739075537e-0001
184 *
185 * Computation note: in simulating higher precision arithmetic, kpcos
186 * return head = 1/pi chopped, and tail = pi/2 * t^2 + (tail part of 1/pi
187 * + ...) to maintain extra bits precision. In particular, pi/2 * t^2
188 * is calculated with great care.
189 *
190 * Thus, the computation of gamma(-x), x>0, is:
191 * Let k = int(x), z = x-k.
192 * For z in (I)
193 *
194 *           k+1
195 *   gamma(-x) = ----- ;
196 *           (-1)
197 *           kpsin(z)*gamma(1+x)
198 *
199 * otherwise, for z in (II),
200 *           k+1
201 *   gamma(-x) = ----- ;
202 *           (-1)
203 *           kpcos(0.5-z)*gamma(1+x)
204 *
205 * otherwise, for z in (III),
206 *           k+1
207 *   gamma(-x) = ----- .
208 *           (-1)
209 *           kpsin(1-z)*gamma(1+x)
210 *
211 * Thus, the computation of gamma(-x) reduced to the computation of
212 * gamma(1+x) and kpsin(), kpcos().
213 *
214 * (B) For x between 1 and 2. We break [1,2] into three parts:
215 *   GT1 = [1.0000, 1.2845]
216 *   GT2 = [1.2844, 1.6374]
217 *   GT3 = [1.6373, 2.0000]
218 *
219 * For x in GTi, i=1,2,3, let
220 *   z1 = 1.134861805732790769689793935774652917006
221 *   gz1 = gamma(z1) = 0.9382046279096824494097535615803269576988
222 *   tz1 = gamma'(z1) = -0.3517214357852935791015625000000000000000
223 *
224 *   z2 = 1.461632144968362341262659542325721328468e+0000
225 *   gz2 = gamma(z2) = 0.8856031944108887002788159005825887332080
226 *   tz2 = gamma'(z2) = 0.00
227 *
228 *   z3 = 1.819773101100500601787868704921606996312e+0000
229 *   gz3 = gamma(z3) = 0.9367814114636523216188468970808378497426
230 *   tz3 = gamma'(z3) = 0.2805306315422058105468750000000000000000
231 *
232 * and
233 *   y = x-zi ... for extra precision, write y = y.h + y.l
234 * Then
235 *   gamma(x) = gzi + tzi*(y.h+y.l) + y*y*Ri(y),
236 *            = gzi.h + (tzi*y.h + ((tzi*y.l+gzi.l) + y*y*Ri(y)))
237 *            = gy.h + gy.l
238 *
239 * where
240 * (I) For double precision
241 *
242 *   Ri(y) = Pi(y)/Qi(y), i=1,2,3;
243 *
244 *   P1(y) = p1[0] + p1[1]*y + ... + p1[4]*y^4
245 *   Q1(y) = q1[0] + q1[1]*y + ... + q1[5]*y^5
246 *
247 *   P2(y) = p2[0] + p2[1]*y + ... + p2[3]*y^3
248 *   Q2(y) = q2[0] + q2[1]*y + ... + q2[6]*y^6

```

```

248 *      P3(y) = p3[0] + p3[1]*y + ... + p3[4]*y^4
249 *      Q3(y) = q3[0] + q3[1]*y + ... + q3[5]*y^5
250 *
251 *      Remez precision of Ri(y):
252 *      |gamma(x)-(gzi+tzi*y) - y*y*Ri(y)|  <= 2**-62.3 ... for i = 1
253 *      <= 2**-59.4 ... for i = 2
254 *      <= 2**-62.1 ... for i = 3
255 *
256 *      (II) For quad precision
257 *
258 *      Ri(y) = Pi(y)/Qi(y), i=1,2,3;
259 *
260 *      P1(y) = p1[0] + p1[1]*y + ... + p1[9]*y^9
261 *      Q1(y) = q1[0] + q1[1]*y + ... + q1[8]*y^8
262 *
263 *      P2(y) = p2[0] + p2[1]*y + ... + p2[9]*y^9
264 *      Q2(y) = q2[0] + q2[1]*y + ... + q2[9]*y^9
265 *
266 *      P3(y) = p3[0] + p3[1]*y + ... + p3[9]*y^9
267 *      Q3(y) = q3[0] + q3[1]*y + ... + q3[9]*y^9
268 *
269 *      Remez precision of Ri(y):
270 *      |gamma(x)-(gzi+tzi*y) - y*y*Ri(y)|  <= 2**-118.2 ... for i = 1
271 *      <= 2**-126.8 ... for i = 2
272 *      <= 2**-119.5 ... for i = 3
273 *
274 *      (III) For single precision
275 *
276 *      Ri(y) = Pi(y), i=1,2,3;
277 *
278 *      P1(y) = p1[0] + p1[1]*y + ... + p1[5]*y^5
279 *
280 *      P2(y) = p2[0] + p2[1]*y + ... + p2[5]*y^5
281 *
282 *      P3(y) = p3[0] + p3[1]*y + ... + p3[4]*y^4
283 *
284 *      Remez precision of Ri(y):
285 *      |gamma(x)-(gzi+tzi*y) - y*y*Ri(y)|  <= 2**-30.8 ... for i = 1
286 *      <= 2**-31.6 ... for i = 2
287 *      <= 2**-29.5 ... for i = 3
288 *
289 *      Notes. (1) GTi and zi are choosen to balance the interval width and
290 *      minimize the distant between gamma(x) and the tangent line at
291 *      zi. In particular, we have
292 *      |gamma(x)-(gzi+tzi*(x-zi))|  <= 0.01436... for x in [1,z2]
293 *      <= 0.01265... for x in [z2,z]
294 *
295 *      (2) zi are slightly adjusted so that tzi=gamma'(zi) is very
296 *      close to a single precision value.
297 *
298 *      Coefficients: Single precision
299 *      i = 1:
300 *      p1[0] = 7.09087253435088360271451613398019280077561279443e-0001
301 *      p1[1] = -5.17229560788652108545141978238701790105241761089e-0001
302 *      p1[2] = 5.23403394528150789405825222323770647162337764327e-0001
303 *      p1[3] = -4.54586308717075010784041566069480411732634814899e-0001
304 *      p1[4] = 4.20596490915239085459964590559256913498190955233e-0001
305 *      p1[5] = -3.57307589712377520978332185838241458642142185789e-0001
306 *
307 *      i = 2:
308 *      p2[0] = 4.28486983980295198166056119223984284432464344578e-0001
309 *      p2[1] = -1.30704539487709138528680121627899735386650103914e-0001
310 *      p2[2] = 1.60856285038051955072861219352655851542955430871e-0001
311 *      p2[3] = -9.22285161346010583774458802067371182158937943507e-0002
312 *      p2[4] = 7.19240511767225260740890292605070595560626179357e-0002
313 *      p2[5] = -4.88158265593355093703112238534484636193260459574e-0002

```

```

314 *
315 *      i = 3
316 *      p3[0] = 3.82409531118807759081121479786092134814808872880e-0001
317 *      p3[1] = 2.65309888180188647956400403013495759365167853426e-0002
318 *      p3[2] = 8.06815109775079171923561169415370309376296739835e-0002
319 *      p3[3] = -1.54821591666137613928840890835174351674007764799e-0002
320 *      p3[4] = 1.76308239242717268530498313416899188157165183405e-0002
321 *
322 *      Coefficients: Double precision
323 *      i = 1:
324 *      p1[0] = 0.70908683619977797008004927192814648151397705078125000
325 *      p1[1] = 1.71987061393048558089579513384356441668351720061e-0001
326 *      p1[2] = -3.19273345791990970293320316122813960527705450671e-0002
327 *      p1[3] = 8.36172645419110036267169600390549973563534476989e-0003
328 *      p1[4] = 1.13745336648572838333152213474277971244629758101e-0003
329 *      q1[0] = 1.0
330 *      q1[1] = 9.71980217826032937526460731778472389791321968082e-0001
331 *      q1[2] = -7.43576743326756176594084137256042653497087666030e-0002
332 *      q1[3] = -1.19345944932265559769719470515102012246995255372e-0001
333 *      q1[4] = 1.59913445751425002620935120470781382215050284762e-0002
334 *      q1[5] = 1.12601136853374984566572691306402321911547550783e-0003
335 *      i = 2:
336 *      p2[0] = 0.42848681585558601181418225678498856723308563232421875
337 *      p2[1] = 6.53596762668970816023718845105667418483122103629e-0002
338 *      p2[2] = -6.97280829631212931321050770925128264272768936731e-0003
339 *      p2[3] = 6.4634235902198171894720860567481326016611663889e-0003
340 *      q2[0] = 1.0
341 *      q2[1] = 4.57572620560506047062553957454062012327519313936e-0001
342 *      q2[2] = -2.52182594886075452859655003407796103083422572036e-0001
343 *      q2[3] = -1.8297094540778594681348166040103197178711552827e-0002
344 *      q2[4] = 2.43574726993169566475227642128830141304953840502e-0002
345 *      q2[5] = -5.20390406466942525358645957564897411258667085501e-0003
346 *      q2[6] = 4.7952025138327983763552431988023256031951133885e-0004
347 *      i = 3:
348 *      p3[0] = 0.382409479734567459008331979930517263710498809814453125
349 *      p3[1] = 1.42876048697668161599069814043449301572928034140e-0001
350 *      p3[2] = 3.42157571052250536817923866013561760785748899071e-0003
351 *      p3[3] = -5.01542621710067521405087887856991700987709272937e-0004
352 *      p3[4] = 8.892858148667409101233834688163838287618332122670e-0004
353 *      q3[0] = 1.0
354 *      q3[1] = 3.04253086629444201002215640948957897906299633168e-0001
355 *      q3[2] = -2.23162407379999477282555672834881213873185520006e-0001
356 *      q3[3] = -1.0506086774195206592180981193367013142755290367e-0002
357 *      q3[4] = 1.70511763916186982473301861980856352005926669320e-0002
358 *      q3[5] = -2.12950201683609187927899416700094630764182477464e-0003
359 *
360 *      Note that all pi0 are exact in double, which is obtained by a
361 *      special Remez Algorithm.
362 *
363 *      Coefficients: Quad precision
364 *      i = 1:
365 *      p1[0] = 0.709086836199777919037185741507610124611513720557
366 *      p1[1] = 4.45754781206489035827915969367354835667391606951e-0001
367 *      p1[2] = 3.21049298735832382311662273882632210062918153852e-0002
368 *      p1[3] = -5.71296796342106617651765245858289197369688864350e-0003
369 *      p1[4] = 6.04666892891998977081619174969855831606965352773e-0003
370 *      p1[5] = 8.99106186996888711939627812174765258822658645168e-0004
371 *      p1[6] = -6.96496846144407741431207008527018441810175568949e-0005
372 *      p1[7] = 1.52597046118984020814225409300131445070213882429e-0005
373 *      p1[8] = 5.68521076168495673844711465407432189190681541547e-0007
374 *      p1[9] = 3.30749673519634895220582062520286565610418952979e-0008
375 *      q1[0] = 1.0+0000
376 *      q1[1] = 1.35806511721671070408570853537257079579490650668e+0000
377 *      q1[2] = 2.975678101534295340530714009603086994072952961e-0001
378 *      q1[3] = -1.52956835982588571502954372821681851681118097870e-0001
379 *      q1[4] = -2.88248519561420109768781615289082053597954521218e-0002

```

```

380 *      q1[5] = 1.03475311719937405219789948456313936302378395955e-0002
381 *      q1[6] = 4.12310203243891222368965360124391297374822742313e-0004
382 *      q1[7] = -3.126537081522290867248931925120380729518332507388e-0004
383 *      q1[8] = 2.36672170850409745237358105667757760527014332458e-0005
384 *
385 *      i = 2:
386 *      p2[0] = 0.428486815855585429730209907810650616737756697477
387 *      p2[1] = 2.63622124067885222919192651151581541943362617352e-0001
388 *      p2[2] = 3.85520683670028865731877276741390421744971446855e-0002
389 *      p2[3] = 3.05065978278128549958897133190295325258023525862e-0003
390 *      p2[4] = 2.48232934951723128892080415054084339152450445081e-0003
391 *      p2[5] = 3.67092777065632360693313762221411547741550105407e-0004
392 *      p2[6] = 3.81228045616085789674530902563145250532194518946e-0006
393 *      p2[7] = 4.61677225867087554059531455133839175822537617677e-0006
394 *      p2[8] = 2.18209052385703200438239200991201916609364872993e-0007
395 *      p2[9] = 1.00490538985245846460006244065624754421022542454e-0008
396 *      q2[0] = 1.0
397 *      q2[1] = 9.20276350207639290567783725273128544224570775056e-0001
398 *      q2[2] = -4.79533683654165107448020515733883781138947771495e-0003
399 *      q2[3] = -1.24538337585899300494444600248687901947684291683e-0001
400 *      q2[4] = 4.49866050763472358547524708431719114204535491412e-0003
401 *      q2[5] = 7.20715455697920560621638325356292640604078591907e-0003
402 *      q2[6] = -8.68513169029126780280798337091982780598228096116e-0004
403 *      q2[7] = -1.25104431629401181525027098222745544809974229874e-0004
404 *      q2[8] = 3.10558344839000038489191304550998047521253437464e-0005
405 *      q2[9] = -1.76829227852852176018537139573609433652506765712e-0006
406 *
407 *      i = 3
408 *      p3[0] = 0.3824094797345675048502747661075355640070439388902
409 *      p3[1] = 3.42198093076618495415854906335908427159833377774e-0001
410 *      p3[2] = 9.63828189500585568303961406863153237440702754858e-0002
411 *      p3[3] = 8.76069421042696384852462044188520252156846768667e-0003
412 *      p3[4] = 1.86477890389161491224872014149309015261897537488e-0003
413 *      p3[5] = 8.16871354540309895879974742853701311541286944191e-0004
414 *      p3[6] = 6.83783483674600322518695090864659381650125625216e-0005
415 *      p3[7] = -1.10168269719261574708565935172719209272190828456e-0006
416 *      p3[8] = 9.66243228508380420159234853278906717065629721016e-0007
417 *      p3[9] = 2.31858885579171250541163820671121664974334728142e-0008
418 *      q3[0] = 1.0
419 *      q3[1] = 8.25479821168813634632437430090376252512793067339e-0001
420 *      q3[2] = -1.62251363073937769739639623669295110346015576320e-0002
421 *      q3[3] = -1.10621286905916732758745130629426559691187579852e-0001
422 *      q3[4] = 3.48309693970985612644446415789230015515365291459e-0003
423 *      q3[5] = 6.73553737487488333032431261131289672347043401328e-0003
424 *      q3[6] = -7.63222008393372630162743587811004613050245128051e-0004
425 *      q3[7] = -1.35792670669190631476784768961953711773073251336e-0004
426 *      q3[8] = 3.19610150954223587006220730065608156460205690618e-0005
427 *      q3[9] = -1.82096553862822346610109522015129585693354348322e-0006
428 *
429 * (C) For x between 0 and 1.
430 * Let P stand for the number of significant bits in the working precision.
431 * 
$$1 - x^P \approx \frac{1}{x} \text{ for } x \leq 2^{-P}$$

432 * (1) For  $0 \leq x \leq 2^{-P}$ ,  $\gamma(x)$  is computed by  $---$  rounded to nearest.
433 *
434 * The error is bound by  $0.739 \text{ ulp}(\gamma(x))$  in IEEE double precision.
435 * Proof.
436 * 
$$\frac{1}{\gamma(x)} \sim x + 0.577... * x^{-1} - \dots$$

437 * Since  $\frac{1}{\gamma(x)} \sim x + 0.577... * x^{-1} - \dots$ , we have, for small  $x$ ,
438 * 
$$\frac{1}{\gamma(x)} < \frac{1}{x} \text{ and } \frac{1}{\gamma(x)} > \frac{1}{x(1+0.578x)}$$

439 *
440 * 
$$0 < \frac{1}{\gamma(x)} - \frac{1}{x} < \frac{1}{x(1+0.578x)}$$

441 *
442 * 
$$0 < \frac{1}{\gamma(x)} - \frac{1}{x} < \frac{1}{x(1+0.578x)}$$

443 *
444 * 
$$\frac{1}{\gamma(x)} - \frac{1}{x} < \frac{1}{x(1+0.578x)}$$

445 *

```

```

446 * The error is thus bounded by  $---$  ulp( $---$ ) + 0.578. Since  $x \leq 2^{-P}$ ,
447 * 
$$\frac{1}{x} \geq 2^P, \text{ ulp}(\frac{1}{x}) \geq \text{ulp}(2^P) \geq 2.$$
 Thus  $0.578 = 0.289 * 2 \leq 0.289 \text{ ulp}(\frac{1}{x})$ 
448 *
449 * 
$$\frac{1}{x} \geq 2^P, \text{ ulp}(\frac{1}{x}) \geq \text{ulp}(2^P) \geq 2.$$
 Thus  $0.578 = 0.289 * 2 \leq 0.289 \text{ ulp}(\frac{1}{x})$ 
450 *
451 * Thus
452 * 
$$|\gamma(x) - \text{rounded}(\gamma(x))| \leq (0.5 + 0.289) * \text{ulp}(\frac{1}{x}).$$

453 *
454 * Note that for  $x \leq 2^{-P}$ , it is easy to see that  $\text{ulp}(\frac{1}{x}) = \text{ulp}(\gamma(x))$ 
455 *
456 * except only when  $x = 2^{-n}$ , ( $n \leq -53$ ). In such cases,  $---$  is exact
457 *
458 * and therefore the error is bounded by
459 * 
$$0.298 * \text{ulp}(\frac{1}{x}) = 0.298 * 2^P * \text{ulp}(\gamma(x)) = 0.578 \text{ ulp}(\gamma(x)).$$

460 *
461 * Thus we conclude that the error in gamma is less than 0.739 ulp.
462 *
463 * (2) Otherwise, for  $x$  in  $GT_{i-1}$  (see B), let  $y = x - (z_i - 1)$ . From (B) we obtain
464 * 
$$\gamma(1+x) = \gamma.y.h + \gamma.y.l$$

465 * then compute  $\gamma(x)$  by  $-----$ .
466 * 
$$\frac{\gamma.y.h}{x}$$

467 *
468 * Implementaion note. Write  $x = x.h + x.l$ , and Let  $th = -----$  chopped to
469 * 
$$\frac{\gamma.y.h}{x}$$

470 *
471 * 20 bits, then
472 * 
$$\gamma(x) = th + \frac{\gamma.y.h + \gamma.y.l}{x} - th$$

473 *
474 * 
$$= th + \frac{1}{x} * (\gamma.y.h - th * x + \gamma.y.l - th * x.l)$$

475 *
476 * (D) For  $x$  between 2 and 8. Let  $n = 1+x$  chopped to an integer. Then
477 * 
$$\gamma(x) = (x-1) * (x-2) * \dots * (x-n) * \gamma(x-n)$$

478 *
479 * Since  $x-n$  is between 1 and 2, we can apply (B) to compute  $\gamma(x)$ .
480 *
481 * Implementation detail. The computation of  $(x-1)(x-2)\dots(x-n)$  in simulated
482 * higher precision arithmetic can be somewhat optimized. For example, in
483 * computing  $(x-1)(x-2)(x-3)(x-4)$ , if we compute  $(x-1)(x-4) = z.h + z.l$ ,
484 * then  $(x-2)(x-3) = z.h + 2z.l$  readily. In below, we list the expression
485 * of the formula to compute  $\gamma(x)$ .
486 *
487 * Assume  $x-n$  is in  $GT_i$  ( $i=1,2$ , or 3, see B for detail). Let  $y = x - n - z_i$ .
488 * By (B) we have  $\gamma(x-n) = \gamma.y.h + \gamma.y.l$ . If  $x = x.h + x.l$ , then we have
489 *  $n=1$  ( $x$  in  $[2,3]$ ):
490 * 
$$\gamma(x) = (x-1) * \gamma(x-1) = (x-1) * (\gamma.y.h + \gamma.y.l)$$

491 * 
$$= [(x.h-1) + x.l] * (\gamma.y.h + \gamma.y.l)$$

492 *
493 *  $n=2$  ( $x$  in  $[3,4]$ ):
494 * 
$$\gamma(x) = (x-1)(x-2) * \gamma(x-2) = (x-1)(x-2) * (\gamma.y.h + \gamma.y.l)$$

495 * 
$$= [(x.h-2) + x.l] * [(x.h-1) + x.l] * (\gamma.y.h + \gamma.y.l)$$

496 * 
$$= [x.h * (x.h-3) + 2x.l * (x + (x.h-3))] * (\gamma.y.h + \gamma.y.l)$$

497 *
498 *  $n=3$  ( $x$  in  $[4,5]$ ):
499 * 
$$\gamma(x) = (x-1)(x-2)(x-3) * \gamma(x-3)$$

500 * 
$$= (x.h * (x.h-3) + 2x.l * (x + (x.h-3))) * [(x.h-3) + x.l] * (\gamma.y.h + \gamma.y.l)$$

501 *
502 *  $n=4$  ( $x$  in  $[5,6]$ ):
503 * 
$$\gamma(x) = [(x-1)(x-4)] * [(x-2)(x-3)] * (\gamma.y.h + \gamma.y.l)$$

504 * 
$$= [(x.h * (x.h-5) + 4x.l * (x + (x.h-5)))] * [(x-2)(x-3)] * (\gamma.y.h + \gamma.y.l)$$

505 * 
$$= (y.h + y.l) * (y.h + 1 + y.l) * (\gamma.y.h + \gamma.y.l)$$

506 *
507 *  $n=5$  ( $x$  in  $[6,7]$ ):
508 * 
$$\gamma(x) = [(x-1)(x-4)] * [(x-2)(x-3)] * [(x-5) * (\gamma.y.h + \gamma.y.l)]$$

509 *

```

```

512 *      n=6 (x in [7,8])
513 *      gamma(x) = [(x-1)(x-6)]*[(x-2)(x-5)]*[(x-3)(x-4)]*(gy.h+gy.l)]
514 *              = [(y.h+y.l)(y.h+4+y.l)][(y.h+6+y.l)(gy.h+gy.l)]
515 *
516 * (E)Overflow Threshold. For x > Overflow threshold of gamma,
517 *      return huge*huge (overflow).
518 *
519 * By checking whether lgamma(x) >= 2**{128,1024,16384}, one can
520 * determine the overflow threshold for x in single, double, and
521 * quad precision. See over.c for details.
522 *
523 * The overflow threshold of gamma(x) are
524 *
525 * single: x = 3.5040096283e+01
526 *          = 0x420C290F (IEEE single)
527 * double: x = 1.71624376956302711505e+02
528 *          = 0x406573FAE561F647 (IEEE double)
529 * quad:   x = 1.7555483429044629170038892160702032034177e+03
530 *          = 0x4009B6E3180CD66A5C4206F128BA77F4 (quad)
531 *
532 * (F)For overflow_threshold >= x >= 8, we use asymptotic approximation.
533 * (1) Stirling's formula
534 *
535 *      log(G(x)) ~ (x-.5)*(log(x)-1) + .5(log(2*pi)-1) + (1/x)*P(1/(x*x))
536 *              = L1 + L2 + L3,
537 * where
538 *      L1(x) = (x-.5)*(log(x)-1),
539 *      L2   = .5(log(2pi)-1) = 0.41893853....,
540 *      L3(x) = (1/x)P(1/(x*x)),
541 *
542 * The range of L1,L2, and L3 are as follows:
543 *
544 * -----
545 * Range(L1) = (single) [8.09...,88.30...] = [2** 3.01...,2** 6.46...]
546 *             (double) [8.09...,709.3...] = [2** 3.01...,2** 9.47...]
547 *             (quad)  [8.09...,11356.10...] = [2** 3.01...,2** 13.47...]
548 * Range(L2) = 0.41893853....
549 * Range(L3) = [0.0104..., 0.00048...] = [2**-6.58...,2**-11.02...]
550 * -----
551 *
552 * Gamma(x) is then computed by exp(L1+L2+L3).
553 *
554 * (2) Error analysis of (F):
555 * -----
556 * The error in Gamma(x) depends on the error inherited in the computation
557 * of L= L1+L2+L3. Let L' be the computed value of L. The absolute error
558 * in L' is t = L-L'. Since exp(L') = exp(L-t) = exp(L)*exp(t) ~
559 * (1+t)*exp(L), the relative error in exp(L') is approximately t.
560 *
561 * To guarantee the relatively accuracy in exp(L'), we would like
562 * |t| < 2**(-P-5) where P denotes for the number of significant bits
563 * of the working precision. Consequently, each of the L1,L2, and L3
564 * must be computed with absolute error bounded by 2**(-P-5) in absolute
565 * value.
566 *
567 * Since L2 is a constant, it can be pre-computed to the desired accuracy.
568 * Also |L3| < 2**(-6); therefore, it suffices to compute L3 with the
569 * working precision. That is,
570 *      L3(x) approximate log(G(x))-(x-.5)(log(x)-1)-.5(log(2pi)-1)
571 * to a precision bounded by 2**(-P-5).
572 *
573 *
574 *
575 *      L1(x):  
$$\frac{2^{*(-6)}}{V}$$

576 *
577 *      L2:    
$$\frac{1}{V}$$


```

```

578 *
579 *      + L3(x):  
$$\frac{1}{V}$$

580 *
581 *      -----
582 *      [leading] + [Trailing]
583 *
584 * For L1(x)=(x-0.5)*(log(x)-1), we need ilogb(L1(x))+5 extra bits for
585 * both multiplicants to guarantee L1(x)'s absolute error is bounded by
586 * 2**(-P-5) in absolute value. Here ilogb(y) is defined to be the unbiased
587 * binary exponent of y in IEEE format. We can get x-0.5 to the desired
588 * accuracy easily. It remains to compute log(x)-1 with ilogb(L1(x))+5
589 * extra bits accuracy. Note that the range of L1 is 88.30..., 709.3..., and
590 * 11356.10... for single, double, and quadruple precision, we have
591 *
592 *
593 *      single      double      quadruple
594 *      -----
595 *      ilogb(L1(x))+5 <= 11          14          18
596 *
597 * (3) Table Driven Method for log(x)-1:
598 * -----
599 * Let x = 2**n * y, where 1 <= y < 2. Let Z={z(i),i=1,...,m}
600 * be a set of predetermined evenly distributed floating point numbers
601 * in [1, 2]. Let z(j) be the closest one to y, then
602 *      log(x)-1 = n*log(2)-1 + log(y)
603 *              = n*log(2)-1 + log(z(j)*y/z(j))
604 *              = n*log(2)-1 + log(z(j)) + log(y/z(j))
605 *              = T1(n) + T2(j) + T3,
606 * where T1(n) = n*log(2)-1 and T2(j) = log(z(j)). Both T1 and T2 can be
607 * pre-calculated and be looked-up in a table. Note that 8 <= x < 1756
608 * implies 3<n<=10 implies 1.079.. < T1(n) < 6.931.
609 *
610 *
611 * For T3, let s =  $\frac{y-z(i)}{y+z(i)}$ ; then  $\frac{y}{z(i)} = \frac{1+s}{1-s}$  and
612 *
613 *      T3 = log( $\frac{1+s}{1-s}$ ) = 2s +  $\frac{2}{3}s^3$  +  $\frac{2}{5}s^5$  + ...
614 *
615 * Suppose the first term 2s is compute in extra precision. The
616 * dominating error in T3 would then be the rounding error of the
617 * second term 2/3*s**3. To force the rounding bounded by
618 * the required accuracy, we have
619 *
620 *      single: |2/3*s**3| < 2**(-11) ==> |s| < 0.09014...
621 *      double: |2/3*s**3| < 2**(-14) ==> |s| < 0.04507...
622 *      quad : |2/3*s**3| < 2**(-18) ==> |s| < 0.01788... = 2**(-5.80..)
623 *
624 * Base on this analysis, we choose Z = {z(i)|z(i)=1+i/64+1/128, 0<=i<=63}.
625 * For any y in [1,2], let j = [64*y] chopped to integer, then z(j) is
626 * the closest to y, and it is not difficult to see that |s| < 2**(-8).
627 * Please note that the polynomial approximation of T3 must be accurate
628 *
629 *      -24-11  -35  -53-14  -67  -113-18  -131
630 * to 2      =2, 2      = 2, and 2      =2
631 * for single, double, and quadruple precision respectively.
632 *
633 * Implementation notes.
634 * (1) Table look-up entries for T1(n) and T2(j), as well as the calculation
635 * of the leading term 2s in T3, are broken up into leading and trailing
636 * part such that (leading part)* 2**24 will always be an integer. That
637 * will guarantee the addition of the leading parts will be exact.
638 *
639 *
640 *
641 *      T1(n):  
$$\frac{2^{*(-24)}}{V}$$

642 *
643 *

```


new/usr/src/lib/libm/common/m9x/tgammaf.c

1

```
*****
15234 Tue Nov 25 12:59:03 2014
new/usr/src/lib/libm/common/m9x/tgammaf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #pragma weak __tgammaf = tgammaf
30 #pragma weak tgammaf = __tgammaf
31
32 /*
33  * True gamma function
34  *
35  * float tgammaf(float x)
36  *
37  * Algorithm: see tgamma.c
38  *
39  * Maximum error observed: 0.87ulp (both positive and negative arguments)
40  */
41
42 #include "libm.h"
43 #include "libm_synonyms.h"
43 #include <math.h>
44 #if defined(__SUNPRO_C)
45 #include <sunmath.h>
46 #endif
47 #include <sys/isa_defs.h>
48
49 #if defined(_BIG_ENDIAN)
50 #define HIWORD 0
51 #define LOWORD 1
52 #else
53 #define HIWORD 1
54 #define LOWORD 0
55 #endif
56 #define __HI(x) ((int *) &x)[HIWORD]
```

new/usr/src/lib/libm/common/m9x/tgammaf.c

2

```
57 #define __LO(x) ((unsigned *) &x)[LOWORD]
58
59 /* Coefficients for primary intervals GTi() */
60 static const double cr[] = {
61     /* p1 */
62     +7.09087253435088360271451613398019280077561279443e-0001,
63     -5.17229560788652108545141978238701790105241761089e-0001,
64     +5.23403394528150789405825222323770647162337764327e-0001,
65     -4.54586308717075010784041566069480411732634814899e-0001,
66     +4.20596490915239085459964590559256913498190955233e-0001,
67     -3.57307589712377520978332185838241458642142185789e-0001,
68
69     /* p2 */
70     +4.28486983980295198166056119223984284434264344578e-0001,
71     -1.30704539487709138528680121627899735386650103914e-0001,
72     +1.60856285038051955072861219352655851542955430871e-0001,
73     -9.22285161346010583774458802067371182158937943507e-0002,
74     +7.19240511767225260740890292605070595560626179357e-0002,
75     -4.88158265593355093703112238534484636193260459574e-0002,
76
77     /* p3 */
78     +3.82409531118807759081121479786092134814808872880e-0001,
79     +2.65309888180188647956400403013495759365167853426e-0002,
80     +8.06815109775079171923561169415370309376296739835e-0002,
81     -1.54821591666137613928840890835174351674007764799e-0002,
82     +1.76308239242717268530498313416899188157165183405e-0002,
83
84     /* GZi and TZi */
85     +0.9382046279096824494097535615803269576988, /* GZ1 */
86     +0.8856031944108887002788159005825887332080, /* GZ2 */
87     +0.9367814114636523216188468970808378497426, /* GZ3 */
88     -0.3517214357852935791015625, /* TZ1 */
89     +0.280530631542205810546875, /* TZ3 */
90 };
91
92 #ifndef unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/tgammal.c

1

40060 Tue Nov 25 12:59:03 2014

new/usr/src/lib/libm/common/m9x/tgammal.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
```

```
30 #pragma weak __tgammal = tgammal
30 #if defined(ELFOSBJ)
31 #pragma weak tgammal = __tgammal
32 #endif
```

```
32 #include "libm.h"
33 #include <sys/isa_defs.h>
```

```
35 #if defined(_BIG_ENDIAN)
36 #define H0_WORD(x)      ((unsigned *) &x)[0]
37 #define H3_WORD(x)      ((unsigned *) &x)[3]
38 #define CHOPPED(x)      (long double) ((double) (x))
39 #else
40 #define H0_WORD(x)      (((int *) &x)[2] << 16) | \
41                          (0x0000ffff & (((unsigned *) &x)[1] >> 15)))
42 #define H3_WORD(x)      ((unsigned *) &x)[0]
43 #define CHOPPED(x)      (long double) ((float) (x))
44 #endif
```

```
46 struct LDouble {
47     long double h, l;
48 };
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/trunc.c

1

```
*****
1772 Tue Nov 25 12:59:04 2014
new/usr/src/lib/libm/common/m9x/trunc.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __trunc = trunc
30 #if defined(ELFOBJ)
31 #pragma weak trunc = __trunc
32 #endif

32 #include "libm.h"

34 double
35 trunc(double x) {
36     union {
37         unsigned i[2];
38         double d;
39     } xx;
40     unsigned hx, sx, i;

42     xx.d = x;
43     hx = xx.i[HIWORD] & ~0x80000000;
44     sx = xx.i[HIWORD] & 0x80000000;
45     if (hx < 0x43300000) { /* |x| < 2^52 */
46         if (hx < 0x3ff00000) /* |x| < 1 */
47             return (sx ? -0.0 : 0.0);

49         /* chop x at the integer bit */
```

new/usr/src/lib/libm/common/m9x/trunc.c

2

```
50     if (hx < 0x41300000) {
51         i = 1 << (0x412 - (hx >> 20));
52         xx.i[HIWORD] &= ~(i | (i - 1));
53         xx.i[LOWORD] = 0;
54     } else {
55         i = 1 << (0x432 - (hx >> 20));
56         xx.i[LOWORD] &= ~(i | (i - 1));
57     }
58     return (xx.d);
59 } else if (hx < 0x7ff00000)
60     return (x);
61     else
62 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
63     return (hx >= 0x7ff80000 ? x : x + x);
64     /* assumes sparc-like QNaN */
65 #else
66     return (x + x);
67 #endif
68 }
unchanged_portion_omitted
```

new/usr/src/lib/libm/common/m9x/truncf.c

1

1610 Tue Nov 25 12:59:05 2014

new/usr/src/lib/libm/common/m9x/truncf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
```

```
30 #pragma weak __truncf = truncf
30 #if defined(ELFOBJ)
31 #pragma weak truncf = __truncf
32 #endif
```

```
32 #include "libm.h"
```

```
34 float
35 truncf(float x) {
36     union {
37         unsigned i;
38         float f;
39     } xx;
40     unsigned hx, sx, i;
41
42     xx.f = x;
43     hx = xx.i & ~0x80000000;
44     sx = xx.i & 0x80000000;
45     if (hx < 0x4b000000) { /* |x| < 2^23 */
46         if (hx < 0x3f800000) /* |x| < 1 */
47             return (sx ? -0.0F : 0.0F);
48
49         /* chop x at the integer bit */
```

new/usr/src/lib/libm/common/m9x/truncf.c

2

```
50         i = 1 << (0x95 - (hx >> 23));
51         xx.i &= ~(i << 1) - 1;
52         return (xx.f);
53     } else if (hx < 0x7f800000) /* |x| is integral */
54         return (x);
55     else
56 #if defined(FPADD_TRAPS_INCOMPLETE_ON_NAN)
57         return (hx > 0x7f800000 ? x * x : x + x);
58 #else
59         return (x + x);
60 #endif
61 }
```

unchanged_portion_omitted

new/usr/src/lib/libm/common/m9x/trunc1.c

1

```
*****
2668 Tue Nov 25 12:59:05 2014
new/usr/src/lib/libm/common/m9x/trunc1.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #pragma weak __trunc1 = trunc1
30 #if defined(ELFOSBJ)
31 #pragma weak trunc1 = __trunc1
32 #endif

32 #include "libm.h"

34 #if defined(__sparc)
35 long double
36 trunc1(long double x) {
37     union {
38         unsigned i[4];
39         long double q;
40     } xx;
41     unsigned hx, sx;
42     int j;

44     xx.q = x;
45     sx = xx.i[0] & 0x80000000;
46     hx = xx.i[0] & ~0x80000000;

48     /* handle trivial cases */
49     if (hx >= 0x406f0000) /* |x| >= 2^112 + ... or x is nan */
```

new/usr/src/lib/libm/common/m9x/trunc1.c

2

```
50         return (hx >= 0x7fff0000 ? x + x : x);

52     /* handle |x| < 1 */
53     if (hx < 0x3fff0000)
54         return (sx ? -0.0L : 0.0L);

56     j = 0x406f - (hx >> 16);          /* 1 <= j <= 112 */
57     xx.i[0] = hx;
58     if (j >= 96) {                    /* 96 <= j <= 112 */
59         xx.i[0] &= ~((1 << (j - 96)) - 1);
60         xx.i[1] = xx.i[2] = xx.i[3] = 0;
61     } else if (j >= 64) {             /* 64 <= j <= 95 */
62         xx.i[1] &= ~((1 << (j - 64)) - 1);
63         xx.i[2] = xx.i[3] = 0;
64     } else if (j >= 32) {            /* 32 <= j <= 63 */
65         xx.i[2] &= ~((1 << (j - 32)) - 1);
66         xx.i[3] = 0;
67     } else                            /* 1 <= j <= 31 */
68         xx.i[3] &= ~((1 << j) - 1);

70     /* negate result if need be */
71     if (sx)
72         xx.i[0] |= 0x80000000;
73     return (xx.q);
74 }
unchanged_portion_omitted
```



```

209     __fmaf;
210     __fmal;
211     __fmax;
212     __fmaxf;
213     __fmaxl;
214     __fmin;
215     __fminf;
216     __fminl;
217     __frexp;
218     __gammaf;           #LSARC/2003/279
219     __gammaf_r;        #LSARC/2003/279
220     __gamma;          #LSARC/2003/279
221     __gammal_r;       #LSARC/2003/279
222     __hypotf;
223     __hypotl;
224     __ilogbf;
225     __ilogbl;
226     __j0f;            #LSARC/2003/279
227     __j0l;            #LSARC/2003/279
228     __j1f;            #LSARC/2003/279
229     __j1l;            #LSARC/2003/279
230     __jn_f;           #LSARC/2003/279
231     __jnl;            #LSARC/2003/279
232     __ldexp;
233     __lgammaf;
234     __lgammaf_r;      #LSARC/2003/279
235     __lgamma;
236     __lgammal_r;      #LSARC/2003/279
237 $if amd64 || sparcv9
238     __llrint           { FLAGS = NODYNSORT };
239     __llrintf          { FLAGS = NODYNSORT };
240     __llrintl          { FLAGS = NODYNSORT };
241     __llround          { FLAGS = NODYNSORT };
242     __llroundf         { FLAGS = NODYNSORT };
243     __llroundl         { FLAGS = NODYNSORT };
244 $else
245 #endif /* ! codereview */
246     __lrint;
247     __lrintf;
248     __lrintl;
249     __lround;
250     __lroundf;
251     __lroundl;
252 $endif
253 #endif /* ! codereview */
254     __log1pf;
255     __log1pl;
256     __log2;
257     __log2f;
258     __log2l;
259     __logbf;
260     __logbl;
261 $if amd64 || sparcv9
262     __lrint           { FLAGS = NODYNSORT };
263     __lrintf          { FLAGS = NODYNSORT };
264     __lrintl          { FLAGS = NODYNSORT };
265     __lround          { FLAGS = NODYNSORT };
266     __lroundf         { FLAGS = NODYNSORT };
267     __lroundl         { FLAGS = NODYNSORT };
268 $else
269     __lrint;
270     __lrintf;
271     __lrintl;
272     __lround;
273     __lroundf;
274     __lroundl;

```

```

275 $endif
276     __modf;
277     __nan;
278     __nanf;
279     __nanl;
280     __nearbyint;
281     __nearbyintf;
282     __nearbyintl;
283     __nextafterf;
284     __nextafterl;
285     __nexttoward;
286     __nexttowardf;
287     __nexttowardl;
288     __remainderf;
289     __remainderl;
290     __remquo;
291     __remquo_f;
292     __remquo_l;
293     __rintf;
294     __rintl;
295     __round;
296     __roundf;
297     __roundl;
298     __scalbf;         #LSARC/2003/279
299     __scalbl;        #LSARC/2003/279
300     __scalbln;
301     __scalblnf;
302     __scalblnl;
303     __scalbnf;
304     __scalbnl;
305     __signgamf;      #LSARC/2003/279
306     __signgaml;      #LSARC/2003/279
307     __significandf;  #LSARC/2003/279
308     __significandl; #LSARC/2003/279
309     __sincos;        #LSARC/2003/279
310     __sincosf;       #LSARC/2003/279
311     __sincosl;       #LSARC/2003/279
312     __tgamma;
313     __tgammaf;
314     __tgammal;
315     __trunc;
316     __truncf;
317     __truncl;
318     __xpg6 { FLAGS = NODIRECT }; #private contract with libc grou
319     __y0f;           #LSARC/2003/279
320     __y0l;           #LSARC/2003/279
321     __y1f;           #LSARC/2003/279
322     __y1l;           #LSARC/2003/279
323     __ynf;           #LSARC/2003/279
324     __ynl;           #LSARC/2003/279
325     __acosf;
326     __acoshf;
327     __acoshl;
328     __acosl;
329     __asinf;
330     __asinhf;
331     __asinhl;
332     __asinl;
333     __atan2f;
334     __atan2l;
335     __atanf;
336     __atanhf;
337     __atanhl;
338     __atanl;
339     __cabs;
340     __cabsf;

```

```

341      cabs1;
342      cacos;
343      cacosf;
344      cacosh;
345      cacoshf;
346      cacoshl;
347      cacosl;
348      carg;
349      cargf;
350      cargl;
351      casin;
352      casinl;
353      casinh;
354      casinhf;
355      casinhl;
356      casinl;
357      catan;
358      catanf;
359      catanh;
360      catanhf;
361      catanhl;
362      catanl;
363      cbrtf;
364      cbrtl;
365      ccos;
366      ccosf;
367      ccosh;
368      ccoshf;
369      ccoshl;
370      ccosl;
371      ceilf;
372      ceil;
373      cexp;
374      cexpf;
375      cexpl;
376      cimag;
377      cimagf;
378      cimagl;
379      clog;
380      clogf;
381      clogl;
382      conj;
383      conjf;
384      conjl;
385      copysignf;
386      copysignl;
387      cosf;
388      coshf;
389      coshl;
390      cosl;
391      cpow;
392      cpowf;
393      cpowl;
394      cproj;
395      cprojf;
396      cprojl;
397      creal;
398      crealf;
399      creall;
400      csin;
401      csinl;
402      csinh;
403      csinhf;
404      csinhl;
405      csinl;
406      csqrt;

```

```

407      csqrtf;
408      csqrtl;
409      ctan;
410      ctanf;
411      ctanh;
412      ctanhf;
413      ctanhl;
414      ctanl;
415      erf;
416      erf;
417      erfl;
418      exp2;
419      exp2f;
420      exp2l;
421      expf;
422      expl;
423      expmlf;
424      expml;
425      fabsf;
426      fabs;
427      fdim;
428      fdimf;
429      fdiml;
430      felearexcept;
431      fegetenv;
432      fegetexceptflag;
433      $if _x86
434      fegetprec;
435      $endif
436      fegetround;
437      feholdexcept;
438      feraiseexcept;
439      fesetenv;
440      fesetexceptflag;
441      $if _x86
442      fesetprec;
443      $endif
444      fesetround;
445      fetestexcept;
446      feupdateenv;
447      fex_get_handling;
448      fex_get_log;
449      fex_get_log_depth;
450      fex_getexcepthandler;
451      fex_log_entry;
452      fex_merge_flags;
453      fex_set_handling;
454      fex_set_log;
455      fex_set_log_depth;
456      fex_setexcepthandler;
457      floorf;
458      floorl;
459      fma;
460      fmaf;
461      fmal;
462      fmax;
463      fmaxf;
464      fmaxl;
465      fmin;
466      fminf;
467      fminl;
468      fmodf;
469      fmodl;
470      frexp;
471      frexpf;
472

```



```

473 frexpl;
474 gammaf; #LSARC/2003/279
475 gammaf_r; #LSARC/2003/279
476 gammal; #LSARC/2003/279
477 gammal_r; #LSARC/2003/279
478 hypotf;
479 hypotl;
480 ilogbf;
481 ilogbl;
482 j0f; #LSARC/2003/279
483 j0l; #LSARC/2003/279
484 j1f; #LSARC/2003/279
485 j1l; #LSARC/2003/279
486 jnf; #LSARC/2003/279
487 jnl; #LSARC/2003/279
488 ldexp;
489 ldexpf;
490 ldexpl;
491 lgammaf;
492 lgammaf_r; #LSARC/2003/279
493 lgammal;
494 lgammal_r; #LSARC/2003/279
495 $if amd64 || sparcv9
496 llrint { FLAGS = NODYNSORT };
497 llrintf { FLAGS = NODYNSORT };
498 llrintl { FLAGS = NODYNSORT };
499 llround { FLAGS = NODYNSORT };
500 llroundf { FLAGS = NODYNSORT };
501 llroundl { FLAGS = NODYNSORT };
502 $else
503 llrint;
504 llrintf;
505 llrintl;
506 llround;
507 llroundf;
508 llroundl;
509 $endif
510 log10f;
511 log10l;
512 log1pf;
513 log1pl;
514 log2;
515 log2f;
516 log2l;
517 logbf;
518 logbl;
519 logf;
520 logl;
521 lrint;
522 lrintf;
523 lrintl;
524 lround;
525 lroundf;
526 lroundl;
527 modf;
528 modff;
529 modfl;
530 nan;
531 nanf;
532 nanl;
533 nearbyint;
534 nearbyintf;
535 nearbyintl;
536 nextafterf;
537 nextafterl;
538 nexttoward;

```

```

539 nexttowardf;
540 nexttowardl;
541 powf;
542 powl;
543 remainderf;
544 remainderl;
545 remquo;
546 remquof;
547 remquol;
548 rintf;
549 rintl;
550 round;
551 roundf;
552 roundl;
553 scalbf; #LSARC/2003/279
554 scalbl; #LSARC/2003/279
555 scalbln;
556 scalblnf;
557 scalblnl;
558 scalbnf;
559 scalbnl;
560 signgamf; #LSARC/2003/279
561 signgaml; #LSARC/2003/279
562 significandf; #LSARC/2003/279
563 significandl; #LSARC/2003/279
564 sincos; #LSARC/2003/279
565 sincosf; #LSARC/2003/279
566 sincosl; #LSARC/2003/279
567 sinf;
568 sinhf;
569 sinhl;
570 sinl;
571 sqrtf;
572 sqrtl;
573 tanf;
574 tanhf;
575 tanhl;
576 tanl;
577 tgamma;
578 tgammaf;
579 tgammal;
580 trunc;
581 truncf;
582 trunc1;
583 y0f; #LSARC/2003/279
584 y0l; #LSARC/2003/279
585 y1f; #LSARC/2003/279
586 y1l; #LSARC/2003/279
587 ynf; #LSARC/2003/279
588 ynl; #LSARC/2003/279
589 } SUNW_1.1.1;

591 SYMBOL_VERSION SUNW_1.1.1 {
592     global:
593         __acosf;
594         __acosl;
595         __asinf;
596         __asinl;
597         __atan2f;
598         __atan2l;
599         __atanf;
600         __atanl;
601         __ceilf;
602         __ceill;
603         __cosf;
604         __coshf;

```

```

605     __coshl;
606     __cosl;
607     __expf;
608     __expl;
609     __fabsf;
610     __fabsl;
611     __floorf;
612     __floorl;
613     __fmodf;
614     __fmodl;
615     __frexpf;
616     __frexpl;
617     __ldexpf;
618     __ldexpl;
619     __log10f;
620     __log10l;
621     __logf;
622     __logl;
623     __modff;
624     __modfl;
625     __powf;
626     __powl;
627     __sinf;
628     __sinhf;
629     __sinhl;
630     __sinl;
631     __sqrtf;
632     __sqrtl;
633     __tanf;
634     __tanhf;
635     __tanhl;
636     __tanl;
637 } SUNW_1.1;

639 SYMBOL_VERSION SUNW_1.1 {
640     global:
641         __acos;
642         __acosh;
643         __asin;
644         __asinh;
645         __atan;
646         __atan2;
647         __atanh;
648         __cbrt;
649         __ceil;
650         __copysign;
651         __cos;
652         __cosh;
653         __erf;
654         __erfc;
655         __exp;
656         __expml;
657         __fabs;
658         __floor;
659         __fmod;
660         __gamma;
661         __gamma_r;
662         __hypot;
663         __ilogb;
664         __isnan;
665         __j0;
666         __j1;
667         __jn;
668         __lgamma;
669         __lgamma_r;
670         __log;

```

```

671         __log10;
672         __loglp;
673         __logb;
674         __nextafter;
675         __pow;
676         __remainder;
677         __rint;
678         __scalb;
679         __scalbn;
680         __signgam;
681         __significand;
682         __sin;
683         __sinh;
684         __sqrt;
685         __tan;
686         __tanh;
687         __y0;
688         __y1;
689         __yn;
690         __acos;
691         __acosh;
692         __asin;
693         __asinh;
694         __atan;
695         __atan2;
696         __atanh;
697         __cbrt;
698         __ceil;
699         __copysign;
700         __cos;
701         __cosh;
702         __erf;
703         __erfc;
704         __exp;
705         __expml;
706         __fabs;
707         __floor;
708         __fmod;
709         __gamma;
710         __gamma_r;
711         __hypot;
712         __ilogb;
713         __isnan;
714         __j0;
715         __j1;
716         __jn;
717         __lgamma;
718         __lgamma_r;
719         __log;
720         __log10;
721         __loglp;
722         __logb;
723         __matherr { FLAGS = NODIRECT };
724         __matherr;
725         __nextafter;
726         __pow;
727         __remainder;
728         __rint;
729         __scalb;
730         __scalbn;
731         __signgam;
732         __significand;
733         __sin;
734         __sinh;
735         __sqrt;
736         __tan;

```

new/usr/src/lib/libm/common/mapfile-vers

11

```
736          tanh;  
737          y0;  
738          y1;  
739          yn;  
740 };
```

unchanged_portion_omitted_

new/usr/src/lib/libm/i386/src/__reduction.s

1

```
*****
2385 Tue Nov 25 12:59:06 2014
new/usr/src/lib/libm/i386/src/__reduction.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "__reduction.s"

31 /
32 /   After argument reduction which returns n:
33 /   n mod 4   sin(x)   cos(x)   tan(x)
34 /   -----
35 /         0         S         C         S/C
36 /         1         C        -S        -C/S
37 /         2        -S        -C         S/C
38 /         3        -C         S        -C/S
39 /   -----

41 #include "libm.h"
42 #include "libm_synonyms.h"
42 #include "libm_protos.h"
44 #undef fabs

44 ENTRY(__reduction)
45 #ifndef PIC
46     movl    12(%esp),%eax        / load the high part of arg
47 #else
48     movl    16(%esp),%eax        / load the high part of arg
49 #endif
50     andl    $0x7fffffff,%eax    / clear sign
51     cmpl   $0x3fe921fb,%eax    / Is |x| < pi/4 (= 0x3fe921fb54...) ?
52     jbe    .L0
53     cmpl   $0x7ff00000,%eax    / Is arg a NaN or an Inf ?
54     jb    .L1
55 .L0:
56 #ifndef PIC
```

new/usr/src/lib/libm/i386/src/__reduction.s

2

```
57     fldl    8(%esp)            / push arg
58 #else
59     fldl    12(%esp)           / push arg
60 #endif
61     fwait
62     movl    $0,%eax            / set n = 0
63     ret
64 .L1:
65     pushl   %ebp
66     movl    %esp,%ebp
67     subl   $16,%esp
68     PIC_SETUP(1)
69     leal   -16(%ebp),%eax      / address of y[0]
70     pushl   %eax
71 #ifndef PIC
72     pushl   16(%ebp)
73     pushl   12(%ebp)
74 #else
75     pushl   20(%ebp)
76     pushl   16(%ebp)
77 #endif
78     call   PIC_F(__rem_pio2)   / call __rem_pio2(x,&y)
79     fldl   -8(%ebp)           / y[1]
80     fldl   -16(%ebp)          / y[0], y[1]
81     faddp  %st,%st(1)         / y[0]+y[1] round-to-extended
82     addl   $28,%esp           / 16+4*3
83     andl   $3,%eax
84     PIC_WRAPUP
85     leave
86     ret
87     .align 4
88     SET_SIZE(__reduction)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/acos.s

1

```
*****
1992 Tue Nov 25 12:59:06 2014
new/usr/src/lib/libm/i386/src/acos.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "acos.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(acos,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

36 #undef fabs

35     ENTRY(acos)
36     fldl    4(%esp)           / push x
37     fldl    / push 1
38     fld     %st(1)           / x , 1 , x
39     fabs    / |x| , 1 , x
40     fucomp
41     ftsw   %ax
42     sahf
43     ja     .ERR
44     fadd   %st(1),%st       / 1+x,x
45     fldz
46     fucomp
47     ftsw   %ax
48     sahf
49     jp     .L1
50     jne   .L1
51     / x is -1
52     fstp  %st(0)           / -1
53     fstp  %st(0)           / empty NPX stack
54     fldpi
55     ret
```

new/usr/src/lib/libm/i386/src/acos.s

2

```
56 .L1:
57     fxch   %st(1)           / x,1+x
58     fldl   / 1,x,1+x
59     fsubp  %st,%st(1)       / 1-x,1+x
60     fdivp  %st,%st(1)       / (1-x)/(1+x)
61     fsqrt
62     fldl   / 1,sqrt((1-x)/(1+x))
63     fpatan
64     fadd   %st(0),%st
65     ret

67 .ERR:
68     / |x| > 1
69     pushl  %ebp
70     movl   %esp,%ebp
71     PIC_SETUP(1)
72     fstp   %st(0)           / x
73     fstp   %st(0)           / empty NPX stack
74     pushl  $1
75     pushl  12(%ebp)         / high x
76     pushl  8(%ebp)          / low x
77     pushl  12(%ebp)         / high x
78     pushl  8(%ebp)          / low x
79     call   PIC_F(_SVID_libm_err) / report SVID result/error
80     addl   $20,%esp
81     PIC_WRAPUP
82     leave
83     ret
84     .align 4
85     SET_SIZE(acos)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/acosf.s

1

```
*****
1705 Tue Nov 25 12:59:07 2014
new/usr/src/lib/libm/i386/src/acosf.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "acosf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(acosf,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

36 #undef fabs

35     ENTRY(acosf)
36     flds    4(%esp)           / push x
37     fldl
38     fld     %st(1)           / x , 1 , x
39     fabs
40     fucomp
41     ftsw   %ax
42     sahf
43     ja     .ERR
44     fadd   %st(1),%st        / 1+x,x
45     fldz
46     fucomp
47     ftsw   %ax
48     sahf
49     jp     .L1
50     jne   .L1
51     / x is -1
52     fstp  %st(0)            / x
53     fstp  %st(0)            / empty NPX stack
54     fldpi
55     ret
```

new/usr/src/lib/libm/i386/src/acosf.s

2

```
56 .L1:
57     fxch   %st(1)           / x,1+x
58     fldl
59     fsubp  %st,%st(1)       / 1,x,1+x
60     fdivp  %st,%st(1)       / 1-x,1+x
61     fsqrt
62     fldl
63     fpatan
64     fadd   %st(0),%st       / 1,sqrt((1-x)/(1+x))
65     ret

67 .ERR:
68     / |x| > 1
69     fstp  %st(0)           / x
70     fstp  %st(0)           / empty NPX stack
71     fldz
72     fdiv  %st(0),%st       / 0/0
73     ret
74     .align 4
75     SET_SIZE(acosf)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/acosl.s

1

1689 Tue Nov 25 12:59:07 2014

new/usr/src/lib/libm/i386/src/acosl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

29 .file "acosl.s"

31 #include "libm.h"

32 LIBM_ANSI_PRAGMA_WEAK(acosl,function)

33 #include "libm_synonyms.h"

35 #undef fabs

```
34 ENTRY(acosl)
35 fldt 4(%esp) / push x
36 fldl / push 1
37 fld %st(1) / x, 1, x
38 fabs / |x|, 1, x
39 fucomp
40 fstsw %ax
41 sahf
42 ja 9f
43 fadd %st(1),%st / 1+x,x
44 fldz
45 fucomp
46 fstsw %ax
47 sahf
48 jp .L1
49 jne .L1
50 / x is -1
51 fstp %st(0) / -1
52 fstp %st(0) / empty NPX stack
53 fldpi
54 ret
55 .L1:
```

new/usr/src/lib/libm/i386/src/acosl.s

2

```
56 fxch %st(1) / x,1+x
57 fldl / 1,x,1+x
58 fsubp %st,%st(1) / 1-x,1+x
59 fdivp %st,%st(1) / (1-x)/(1+x)
60 fsqrt
61 fldl / 1,sqrt((1-x)/(1+x))
62 fpatan
63 fadd %st(0),%st
64 ret
65 9:
66 / |x| > 1
67 fstp %st(0) / x
68 fsub %st,%st(0) / +/-0 or NaN+invalid
69 fdiv %st,%st(0) / NaN+invalid or NaN
70 ret
71 .align 4
72 SET_SIZE(acosl)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/asin.s

1

```
*****
1853 Tue Nov 25 12:59:08 2014
new/usr/src/lib/libm/i386/src/asin.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "asin.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(asin,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

36 #undef fabs

35     ENTRY(asin)
36     fldl    4(%esp)           / push x
37     fldl           / push 1
38     fld     %st(1)           / x , 1 , x
39     fabs           / |x| , 1 , x
40     fucomp
41     ftsw %ax
42     sahf
43     ja     .ERR
44     fadd   %st(1),%st           / 1+x,x
45     fldl   / 1,1+x,x
46     fsub   %st(2),%st           / 1-x,1+x,x
47     fmulp  %st,%st(1)          / (1-x)*(1+x),x
48     fsqrt  / sqrt((1-x)/(1+x)),x
49     fpatan / atan(x/sqrt((1-x)/(1+x)))
50     ret

52 .ERR:
53     / |x| > 1
54     pushl  %ebp
55     movl   %esp,%ebp
```

new/usr/src/lib/libm/i386/src/asin.s

2

```
56     PIC_SETUP(1)
57     fstp   %st(0)           / x
58     fstp   %st(0)           / empty NPX stack
59     pushl  $2
60     pushl  12(%ebp)         / high x
61     pushl  8(%ebp)          / low x
62     pushl  12(%ebp)         / high x
63     pushl  8(%ebp)          / low x
64     call   PIC_F(_SVID_libm_err) / report SVID result/error
65     addl   $20,%esp
66     PIC_WRAPUP
67     leave
68     ret
69     .align 4
70     SET_SIZE(asin)

_____ unchanged portion omitted
```


new/usr/src/lib/libm/i386/src/asinf.s

1

1567 Tue Nov 25 12:59:08 2014

new/usr/src/lib/libm/i386/src/asinf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "asinf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(asinf,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

36 #undef fabs

35     ENTRY(asinf)
36     flds    4(%esp)           / push x
37     fldl    / push 1
38     fld     %st(1)           / x , 1 , x
39     fabs    / |x| , 1 , x
40     fucomp
41     ftsw   %ax
42     sahf
43     ja     .ERR
44     fadd   %st(1),%st        / 1+x,x
45     fldl   / 1,1+x,x
46     fsub   %st(2),%st        / 1-x,1+x,x
47     fmulp  %st,%st(1)       / (1-x)*(1+x),x
48     fsqrt  / sqrt((1-x)*(1+x)),x
49     fpatan / atan(x/sqrt((1-x)*(1+x)))
50     ret

52 .ERR:
53     / |x| > 1
54     fstp   %st(0)           / x
55     fstp   %st(0)           / empty NPX stack
```

new/usr/src/lib/libm/i386/src/asinf.s

2

```
56     fldz
57     fdiv   %st(0),%st      / 0/0
58     ret
59     .align 4
60     SET_SIZE(asinf)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/asinl.s

1

```
*****
1539 Tue Nov 25 12:59:09 2014
new/usr/src/lib/libm/i386/src/asinl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "asinl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(asinl,function)
33 #include "libm_synonyms.h"

35 #undef fabs

34     ENTRY(asinl)
35     fldt     4(%esp)           / push x
36     fldl     / push 1
37     fld      %st(1)           / x , 1 , x
38     fabs     / |x| , 1 , x
39     fucomp
40     fstsw   %ax
41     sahf
42     ja      9f
43     fadd    %st(1),%st        / 1+x,x
44     fldl    / 1,1+x,x
45     fsub    %st(2),%st        / 1-x,1+x,x
46     fmulp   %st,%st(1)       / (1-x)*(1+x),x
47     fsqrt   / sqrt((1-x)*(1+x)),x
48     fpatan  / atan(x/sqrt((1-x)*(1+x)))
49     ret

50 9:
51     / |x| > 1
52     fstp    %st(0)           / x
53     fsub    %st,%st(0)       / +/-0 or NaN+invalid
54     fdiv    %st,%st(0)       / NaN+invalid or NaN
55     ret
```

new/usr/src/lib/libm/i386/src/asinl.s

2

```
56     .align 4
57     SET_SIZE(asinl)
_____unchanged_portion_omitted_
```

new/usr/src/lib/libm/i386/src/atan.s

1

1187 Tue Nov 25 12:59:09 2014

new/usr/src/lib/libm/i386/src/atan.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file      "atan.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(atan,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(atan)
```

```
35     fldl    4(%esp)           / push arg
```

```
36     fldl           / push 1.0
```

```
37     fpatan           / atan(arg/1.0)
```

```
38     ret
```

```
39     .align 4
```

```
40     SET_SIZE(atan)
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/atan2.s

1

```
*****
1782 Tue Nov 25 12:59:10 2014
new/usr/src/lib/libm/i386/src/atan2.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "atan2.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(atan2,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(atan2)
36     movl    4(%esp),%eax        / low part of y
37     movl    12(%esp),%ecx       / low part of x
38     orl     %eax,%ecx
39     jz      .maybe_0s

41     / not both x and y are 0's
42 1:
43     fldl    4(%esp)            / push y
44     fldl    12(%esp)           / push x
45     fpatan                                / return atan2(y,x)
46     ret

48 .maybe_0s:
49     movl    8(%esp),%eax        / high part of y
50     movl    16(%esp),%ecx       / high part of x
51     orl     %eax,%ecx
52     andl    $0x7fffffff,%ecx   / clear sign
53     jnz    1b
54     / both x and y are 0's
55     pushl   %ebp
56     movl    %esp,%ebp
57     PIC_SETUP(1)
```

new/usr/src/lib/libm/i386/src/atan2.s

2

```
58     pushl   $3
59     pushl   12(%ebp)           / high y
60     pushl   8(%ebp)            / low y
61     pushl   20(%ebp)           / high x
62     pushl   16(%ebp)           / low x
63     call    PIC_F(_SVID_libm_err) / report SVID result/error
64     addl    $20,%esp
65     PIC_WRAPUP
66     leave
67     ret
68     .align 4
69     SET_SIZE(atan2)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/atan2f.s

1

1234 Tue Nov 25 12:59:10 2014

new/usr/src/lib/libm/i386/src/atan2f.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "atan2f.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(atan2f,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"
```

```
35     ENTRY(atan2f)
36     flds    4(%esp)        / push y
37     flds    8(%esp)        / push x
38     fpatan          / return atan2(y,x)
39     ret
40     .align 4
41     SET_SIZE(atan2f)
```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/atan21.s

1

1209 Tue Nov 25 12:59:10 2014

new/usr/src/lib/libm/i386/src/atan21.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "atan21.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(atan21,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(atan21)
```

```
35     fldt    4(%esp)           / push y
```

```
36     fldt   16(%esp)          / push x
```

```
37     fpatan                                / return atan2(y,x)
```

```
38     ret
```

```
39     .align 4
```

```
40     SET_SIZE(atan21)
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/atanl.s

1

1191 Tue Nov 25 12:59:11 2014

new/usr/src/lib/libm/i386/src/atanl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "atanl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(atanl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(atanl)
```

```
35     fldt    4(%esp)        / push arg
```

```
36     fldl                    / push 1.0
```

```
37     fpatan                    / atan(arg/1.0)
```

```
38     ret
```

```
39     .align 4
```

```
40     SET_SIZE(atanl)
```

```
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/ceil.s

1

```
*****
1440 Tue Nov 25 12:59:11 2014
new/usr/src/lib/libm/i386/src/ceil.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "ceil.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ceil,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(ceil)
35 subl $8,%esp
36 fstcw (%esp)
37 fldl 12(%esp)
38 movw (%esp),%cx
39 orw $0x0c00,%cx
40 xorw $0x0400,%cx
41 movw %cx,4(%esp)
42 fldcw 4(%esp) / set RD = up
43 frndint
44 fstcw 4(%esp) / restore RD
45 movw 4(%esp),%dx
46 andw $0xf3ff,%dx
47 movw (%esp),%cx
48 andw $0x0c00,%cx
49 orw %dx,%cx
50 movw %cx,(%esp)
51 fldcw (%esp) / restore RD
52 addl $8,%esp
53 ret
54 .align 4
55 SET_SIZE(ceil)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/copysign.s

1

1733 Tue Nov 25 12:59:12 2014

new/usr/src/lib/libm/i386/src/copysign.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file    "copysign.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(copysign,function)
33 #include "libm_synonyms.h"

34     ENTRY(copysign)
35     movl    8(%esp),%eax        / eax <-- hi_32(x)
36     movl    16(%esp),%ecx       / ecx <-- hi_32(y)
37     andl    $0x7fffffff,%eax   / eax <-- hi_32(abs(x))
38     andl    $0x80000000,%ecx    / ecx[31] <-- sign_bit(y)
39     orl     %ecx,%eax          / eax <-- hi_32(copysign(x,y))
40     movl    4(%esp),%ecx       / ecx <-- lo_32(x)
41                                     /      = lo_32(copysign(x,y))
42     subl    $8,%esp           / set up loading dock for result
43     movl    %ecx,(%esp)        / copy lo_32(result) to loading dock
44     movl    %eax,4(%esp)       / copy hi_32(result) to loading dock
45     fldl    (%esp)            / load copysign(x,y)
46     fwait                                / in case fldl causes exception
47     addl    $8,%esp           / restore stack-pointer for return
48     ret
49     .align 4
50     SET_SIZE(copysign)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/copysignf.s

1

1572 Tue Nov 25 12:59:12 2014

new/usr/src/lib/libm/i386/src/copysignf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file      "copysignf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(copysignf,function)
33 #include "libm_synonyms.h"

34     ENTRY(copysignf)
35     movl    4(%esp),%eax        / eax <-- x
36     movl    8(%esp),%ecx        / ecx <-- y
37     andl    $0x7fffffff,%eax   / eax <-- abs(x)
38     andl    $0x80000000,%ecx   / ecx[31] <-- sign_bit(y)
39     orl     %ecx,%eax          / eax <-- copysign(x,y)
40     subl    $4,%esp           / set up loading dock for result
41     movl    %eax,(%esp)        / copy result to loading dock
42     flds   (%esp)             / load copysign(x,y)
43     fwait                                / in case fldl causes exception
44     addl   $4,%esp            / restore stack-pointer for return
45     ret
46     .align 4
47     SET_SIZE(copysignf)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/copysignl.s

1

1821 Tue Nov 25 12:59:12 2014

new/usr/src/lib/libm/i386/src/copysignl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "copysignl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(copysignl,function)
33 #include "libm_synonyms.h"

34     ENTRY(copysignl)
35     movl    12(%esp),%eax        / sign and bexp of x
36     movl    24(%esp),%ecx        / sign and bexp of y
37     andl    $0x00007fff,%eax    / eax <-- bexp(x)
38     andl    $0x00008000,%ecx    / ecx <-- sign(y)
39     orl     %ecx,%eax           / eax <-- bexp(x) with sign(y)
40     movl    8(%esp),%ecx        / ecx <-- hi_32(sgnfcnd(x))
41     movl    4(%esp),%edx        / edx <-- lo_32(sgnfcnd(x))
42     subl    $12,%esp           / set up loading dock for result
43     movl    %edx,(%esp)        / copy lo_32(result's sgnfcnd)
44                                     / to loading dock
45     movl    %ecx,4(%esp)       / copy hi_32(result's sgnfcnd)
46                                     / to loading dock
47     movl    %eax,8(%esp)       / copy sign&bexp(result)
48                                     / to loading dock
49     fldt    (%esp)            / load copysign(x,y)
50     addl    $12,%esp          / restore stack-pointer for return
51     ret
52     .align 4
53     SET_SIZE(copysignl)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/cos.s

1

1322 Tue Nov 25 12:59:13 2014

new/usr/src/lib/libm/i386/src/cos.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "cos.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(cos,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"
```

```
35     ENTRY(cos)
36     PIC_SETUP(1)
37     call    PIC_F(__reduction)
38     PIC_WRAPUP
39     cmpl    $1,%eax
40     jl     .cos0
41     je     .cos1
42     cmpl    $2,%eax
43     je     .cos2
44     fsin
45     ret
46 .cos2:
47     fcos
48     fchs
49     ret
50 .cos1:
51     fsin
52     fchs
53     ret
54 .cos0:
55     fcos
56     ret
57     .align 4
```

new/usr/src/lib/libm/i386/src/cos.s

2

58 SET_SIZE(cos)

unchanged_portion_omitted

```

*****
4254 Tue Nov 25 12:59:13 2014
new/usr/src/lib/libm/i386/src/exp.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "exp.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(exp)
36     movl    8(%esp),%ecx        / ecx <-- hi_32(x)
37     andl    $0x7fffffff,%ecx   / ecx <-- hi_32(|x|)
38     cmpl    $0x3fe62e42,%ecx   / Is |x| < ln(2)?
39     jb     .shortcut          / If so, take a shortcut.
40     je     .check_tail        / |x| may be only slightly < ln(2)
41     cmpl    $0x7ff00000,%ecx   / hi_32(|x|) >= hi_32(INF)?
42     jae     .not_finite        / if so, x is not finite
43 .finite_non_special:
44     fldl    4(%esp)            / push x
45     subl    $8,%esp
46
47     fstcw   (%esp)              /// overhead of RP save/restore; 63/15
48     movw   (%esp),%ax           /// ; 15/3
49     movw   %ax,4(%esp)          /// ; 4/1
50     orw   $0x0300,%ax          /// save old RP; 2/1
51     movw   %ax,(%esp)           /// force 64-bit RP; 2/1
52     fldcw  (%esp)              /// ; 2/1
53     fldl2e %st,%st(1)          /// ; 19/4
54     fmulp  %st,%st(1)          / push log2e }not for xtndd_dbl
55     fld   %st(0)               / z = x*log2e }not for xtndd_dbl
56     frndint %st(0)            / duplicate stack top
57     fucom  [z],z               / [z],z
58                                     / This and the next 3 instructions

```

```

58     fstsw  %ax                 / add 10 clocks to runtime of the
59     sahf   %eax                / main branch, but save about 265
60     je     .z_integral         / upon detection of integral z.
61     / [z] != z, compute exp(x)
62     fxch   %eax                / z,[z]
63     fsb   %st(1),%st          / z-[z],[z]
64     f2xml  %eax,%eax           / 2***(z-[z])-1,[z]
65     fldl   %eax                / 1,2***(z-[z])-1,[z]
66     faddp  %st,%st(1)         / 2***(z-[z]) ,[z]
67 .merge:
68     fscale %eax,%eax           / exp(x) ,[z]
69     fstp   %st(1)              / restore RD
70     fstcw  (%esp)
71     movw   (%esp),%dx
72     andw   $0xfcff,%dx
73     movw   4(%esp),%cx
74     andw   $0x0300,%cx
75     orw   %dx,%cx
76     movw   %cx,(%esp)
77     fldcw  (%esp)              /// restore old RP; 19/4
78     fstpl  (%esp)              / round to double
79     fldl   (%esp)              / exp(x) rounded to double
80     fxam   %eax,%eax           / determine class of exp(x)
81     add    $8,%esp
82     fstsw  %ax                 / store status in ax
83     andw   $0x4500,%ax
84     cmpw   $0x0500,%ax
85     je     .overflow
86     cmpw   $0x4000,%ax
87     je     .underflow
88     ret

90 .overflow:
91     fstp   %st(0)              / stack empty
92     push  %ebp
93     mov   %esp,%ebp
94     PIC_SETUP(1)
95     pushl $6
96     jmp   .error

98 .underflow:
99     fstp   %st(0)              / stack empty
100    push  %ebp
101    mov   %esp,%ebp
102    PIC_SETUP(2)
103    pushl $7

105 .error:
106    pushl 12(%ebp)              / high x
107    pushl 8(%ebp)               / low x
108    pushl 12(%ebp)              / high x
109    pushl 8(%ebp)               / low x
110    call  PIC_F(_SVID_libm_err)
111    addl  $20,%esp
112    PIC_WRAPUP
113    leave
114    ret

116 .z_integral:
117    fstp   %st(0)              / here, z is integral
118    fldl   %eax                / ,z
119    jmp   .merge

121 .check_tail:
122    movl   4(%esp),%edx         / edx <-- lo_32(x)
123    cmpl   $0xfefa39ef,%edx    / Is |x| slightly < ln(2)?

```

```

124     ja      .finite_non_special    / branch if |x| slightly > ln(2)
125 .shortcut:
126     / Here, |x| < ln(2), so |z| = |x*log2(e)| < 1,
127     / whence z is in f2xml's domain.
128     fldl   4(%esp)                / push x
129     fldl2e %st,%st(1)             / push log2e }not for xtndd_dbl
130     fmulp  %st,%st(1)             / z = x*log2e }not for xtndd_dbl
131     f2xml  %st,%st(1)             / 2**(x*log2(e))-1 = e**x - 1
132     fldl   %st,%st(1)             / 1,2**(z)-1
133     faddp  %st,%st(1)             / 2**(z) = e**x
134     ret

136 .not_finite:
137     / Here, flags still have settings from execution of
138     / cmpl   $0x7ff00000,%ecx      / hi_32(|x|) > hi_32(INF)?
139     ja     .NaN_or_pinf           / if not, x may be +/- INF
140     movl  4(%esp),%edx            / edx <-- lo_32(x)
141     cmpl  $0,%edx                / lo_32(x) = 0?
142     jne   .NaN_or_pinf           / if not, x is NaN
143     movl  8(%esp),%eax            / eax <-- hi_32(x)
144     andl  $0x80000000,%eax       / here, x is infinite, but +/-?
145     jz    .NaN_or_pinf           / branch if x = +INF
146     fldz  %st,%st(1)            / Here, x = -inf, so return 0
147     ret

149 .NaN_or_pinf:
150     / Here, x = NaN or +inf, so load x and return immediately.
151     fldl  4(%esp)
152     fwait
153     ret
154     .align 4
155     SET_SIZE(exp)
_____unchanged_portion_omitted_____

```

```

*****
3528 Tue Nov 25 12:59:14 2014
new/usr/src/lib/libm/i386/src/exp10.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "exp10.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp10,function)
33 #include "libm_synonyms.h"

33     ENTRY(exp10)
34     movl    8(%esp),%ecx          / ecx <-- hi_32(x)
35     andl   $0x7fffffff,%ecx     / ecx <-- hi_32(|x|)
36     cmpl   $0x3fd34413,%ecx     / Is |x| < log10(2)?
37     jb     .shortcut           / If so, take a shortcut.
38     je     .check_tail         / maybe |x| only slightly < log10(2)
39     cmpl   $0x7ff00000,%ecx     / hi_32(|x|) >= hi_32(INF)?
40     jae    .not_finite         / if so, x is not finite
41 .finite_non_special:         / Here, log10(2) < |x| < INF
42     fldl   4(%esp)             / push x (=arg)

44     subl   $8,%esp             / save RP and set round-to-64-bits
45     fstcw  (%esp)
46     movw   (%esp),%ax
47     movw   %ax,4(%esp)
48     orw   $0x0300,%ax
49     movw   %ax,(%esp)
50     fldcw  (%esp)

52     fldl2t                / push log2(10) }NOT for xtndd_dbl
53     fmulpl %st,%st(1)     / z = x*log2(10) }NOT for xtndd_dbl
54     fld   %st(0)          / duplicate stack top
55     frndint                / [z],z
56     fucom                / z integral?

```

```

57     fstsw  %ax
58     sahf
59     je     .z_integral       / branch if z integral
60     fxch
61     fsb   %st(1),%st        / z-[z], [z]
62     f2xm1
63     fldl
64     faddpl %st,%st(1)       / 2***(z-[z])-1, [z]
65     fscale
66     fstpl  %st(1)           / 2***(z-[z]), [z]
67
68     fstcw  (%esp)           / restore old RP
69     movw   (%esp),%dx
70     andw   $0xfcff,%dx
71     movw   4(%esp),%cx
72     andw   $0x0300,%cx
73     orw   %dx,%cx
74     movw   %cx,(%esp)
75     fldcw  (%esp)
76     add    $8,%esp

78     ret

80 .z_integral:                / here, z is integral
81     fstpl  %st(0)
82     fldl
83     fscale
84     fstpl  %st(1)           / l = 2**0, z
85
86     fstcw  (%esp)           / restore old RP
87     movw   (%esp),%dx
88     andw   $0xfcff,%dx
89     movw   4(%esp),%cx
90     andw   $0x0300,%cx
91     orw   %dx,%cx
92     movw   %cx,(%esp)
93     fldcw  (%esp)
94     add    $8,%esp

96     ret

98 .check_tail:
99     movl   4(%esp),%edx       / edx <-- lo_32(x)
100    cmpl   $0x509f79fe,%edx   / Is |x| slightly > log10(2)?
101    ja     .finite_non_special / branch if |x| slightly > log10(2)
102 .shortcut:
103    / Here, |x| < log10(2), so |z| = |x*log2(10)| < 1
104    / whence z is in f2xm1's domain.
105    fldl   4(%esp)           / push x (=arg)
106    fldl2t                / push log2(10) }NOT for xtndd_dbl
107    fmulpl %st,%st(1)       / z = x*log2(10) }NOT for xtndd_dbl
108    f2xm1
109    fldl
110    faddpl %st,%st(1)       / 2**z - 1
111    ret

113 .not_finite:
114    cmpl   $0x7ff00000,%ecx   / hi_32(|x|) > hi_32(INF)?
115    ja     .NaN_or_pinf
116    movl   4(%esp),%edx       / if so, x is NaN
117    cmpl   $0,%edx           / edx <-- lo_32(x)
118    jne    .NaN_or_pinf       / lo_32(x) = 0?
119    movl   .NaN_or_pinf
120    movl   8(%esp),%eax       / if not, x is NaN
121    andl   $0x80000000,%eax   / eax <-- hi_32(x)
122    jz     .NaN_or_pinf       / here, x is infinite, but +/-?
123    jz     .NaN_or_pinf       / branch if x = +INF
124    fldz
125    / Here, x = -inf, so return 0

```

new/usr/src/lib/libm/i386/src/exp10.s

3

```
123         ret

125 .NaN_or_pinf:
126     / Here, x = NaN or +inf, so load x and return immediately.
127     fldl    4(%esp)
128     fwait
129     ret
130     .align 4
131     SET_SIZE(exp10)
_____unchanged_portion_omitted_____
```



```

*****
3069 Tue Nov 25 12:59:14 2014
new/usr/src/lib/libm/i386/src/exp10f.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "exp10f.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp10f,function)
33 #include "libm_synonyms.h"

33     ENTRY(exp10f)
34     movl    4(%esp),%ecx        / ecx <-- x
35     andl    $0x7fffffff,%ecx   / ecx <-- |x|
36     cmpl    $0x3e9a209a,%ecx   / Is |x| < log10(2)?
37     jbe     .shortcut         / If so, take a shortcut.
38     cmpl    $0x7f800000,%ecx   / |x| >= INF?
39     jae     .not_finite       / if so, x is not finite
40     flds    4(%esp)           / push x (=arg)

42     subl    $8,%esp           / save RP and set round-to-64-bits
43     fstcw   (%esp)
44     movw   (%esp),%ax
45     movw   %ax,4(%esp)
46     orw    $0x0300,%ax
47     movw   %ax,(%esp)
48     fldcw  (%esp)

50     fldl2t                / push log2(10) }NOT for xtndd_dbl
51     fmulp   %st,%st(1)     / z = x*log2(10) }NOT for xtndd_dbl
52     fld     %st(0)         / duplicate stack top
53     frndint                / [z],z
54     fucom   %ax             / z integral?
55     fstsw   %ax
56     sahf

```

```

57     je     .z_integral      / branch if z integral
58     fxch   %z              / z, [z]
59     fsb    %st(1),%st      / z-[z], [z]
60     f2xml                / 2**(z-[z])-1, [z]
61     fldl   %st(1)          / 1,2**(z-[z])-1, [z]
62     faddp   %st,%st(1)     / 2**(z-[z]), [z]
63     fscale                / 2**z = 10**(arg), [z]
64     fstp   %st(1)

66     fstcw   (%esp)         / restore old RP
67     movw   (%esp),%dx
68     andw   $0xfcff,%dx
69     movw   4(%esp),%cx
70     andw   $0x0300,%cx
71     orw    %dx,%cx
72     movw   %cx,(%esp)
73     fldcw  (%esp)
74     add    $8,%esp

76     ret

78     .z_integral:         / here, z is integral
79     fstp   %st(0)        / ,z
80     fldl   %st(1)        / 1 = 2**0, z
81     fscale                / 2**(0 + z) = 2**z = 10**(arg), z
82     fstp   %st(1)        / 10**(arg)

84     fstcw   (%esp)         / restore old RP
85     movw   (%esp),%dx
86     andw   $0xfcff,%dx
87     movw   4(%esp),%cx
88     andw   $0x0300,%cx
89     orw    %dx,%cx
90     movw   %cx,(%esp)
91     fldcw  (%esp)
92     add    $8,%esp

94     ret

96     .shortcut:
97     / Here, |x| < log10(2), so |z| = |x*log2(10)| < 1
98     / whence z is in f2xml's domain.
99     flds   4(%esp)         / push x (=arg)
100    fldl2t                / push log2(10) }NOT for xtndd_dbl
101    fmulp   %st,%st(1)     / z = x*log2(10) }NOT for xtndd_dbl
102    f2xml                / 2**z - 1
103    fldl   %st(1)         / 1,2**z - 1
104    faddp   %st,%st(1)     / 2**z = 10**x
105    ret

107    .not_finite:
108    ja     .NaN_or_pinf     / branch if x is NaN
109    movl   4(%esp),%eax     / eax <-- x
110    andl   $0x80000000,%eax / here, x is infinite, but +/-?
111    jz     .NaN_or_pinf     / branch if x = +INF
112    fldz                / Here, x = -inf, so return 0
113    ret

115    .NaN_or_pinf:
116    / Here, x = NaN or +inf, so load x and return immediately.
117    flds   4(%esp)
118    fwait
119    ret
120    .align 4
121    SET_SIZE(exp10f)

```

unchanged_portion_omitted

```

*****
3473 Tue Nov 25 12:59:15 2014
new/usr/src/lib/libm/i386/src/exp101.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "exp101.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp101,function)
33 #include "libm_synonyms.h"

33     .data
34     .align 4
35 lt2_hi: .long 0xfbd00000, 0x9a209a84, 0x00003ffd
36 lt2_lo: .long 0x653f4837, 0x8677076a, 0x0000bfc9

38     ENTRY(exp101)
39     movl 12(%esp),%ecx      / cx <-- sign&bexp(x)
40     andl $0x00007fff,%ecx  / ecx <-- zero_xtnd(bexp(x))
41     cmpl $0x00003ffd,%ecx  / Is |x| < log10(2)?
42     jb   .shortcut        / If so, take a shortcut.
43     je   .check_tail      / maybe |x| only slightly < log10(2)
44     cmpl $0x00007fff,%ecx  / bexp(|x|) = bexp(INF)?
45     je   .not_finite      / if so, x is not finite
46     cmpl $0x0000400e,%ecx  / |x| < 32768 = 2^15?
47     jb   .finite_non_special / if so, proceed with argument reduction
48     fldt 4(%esp)          / x
49     fldl %st(0)           / 1, x
50     jmp 1f
51 .finite_non_special:     / Here, log10(2) < |x| < 2^15
52     fldt 4(%esp)          / x
53     fld  %st(0)           / x, x
54     fldl2t                / log2(10), x, x
55     fmulp                 / z := x*log2(10), x
56     frndint               / [z], x

```

```

57     fst  %st(2)           / [z], x, [z]
58     PIC_SETUP(1)
59     fldt PIC_L(lt2_hi)    / lt2_hi, [z], x, [z]
60     fmulp                 / [z]*lt2_hi, x, [z]
61     fsubrp %st,%st(1)    / x-[z]*lt2_hi, [z]
62     fldt PIC_L(lt2_lo)    / lt2_lo, x-[z]*lt2_hi, [z]
63     PIC_WRAPUP
64     fmul  %st(2),%st      / [z]*lt2_lo, x-[z]*lt2_hi, [z]
65     fsubrp %st,%st(1)    / r := x-[z]*log10(2), [z]
66     fldl2t                / log2(10), r, [z]
67     fmulp                 / f := r*log2(10), [z]
68     f2xml                 / 2^f-1, [z]
69     fldl                  / 1, 2^f-1, [z]
70     faddp %st,%st(1)     / 2^f, [z]
71 1:
72     fscale                / 10^x, [z]
73     fstp %st(1)
74     ret

76 .check_tail:
77     movl 8(%esp),%ecx     / ecx <-- hi_32(sgnfcnd(x))
78     cmpl $0x9a209a84,%ecx / Is |x| < log10(2)?
79     ja   .finite_non_special
80     jb   .shortcut
81     movl 4(%esp),%edx     / edx <-- lo_32(sgnfcnd(x))
82     cmpl $0xfbcff798,%edx / Is |x| slightly > log10(2)?
83     ja   .finite_non_special / branch if |x| slightly > log10(2)
84 .shortcut:
85     / Here, |x| < log10(2), so |z| = |x/log10(2)| < 1
86     / whence z is in f2xml's domain.
87     fldt 4(%esp)          / x
88     fldl2t                / log2(10), x
89     fmulp                 / z := x*log2(10)
90     f2xml                 / 2^z-1
91     fldl                  / 1, 2^z-1
92     faddp %st,%st(1)     / 10^x
93     ret

95 .not_finite:
96     movl 8(%esp),%ecx     / ecx <-- hi_32(sgnfcnd(x))
97     cmpl $0x80000000,%ecx / hi_32(sgnfcnd(x)) = hi_32(sgnfcnd(INF))
98     jne .NaN_or_pinf     / if not, x is NaN or unsp.
99     movl 4(%esp),%edx     / edx <-- lo_32(sgnfcnd(x))
100    cmpl $0,%edx          / lo_32(sgnfcnd(x)) = 0?
101    jne .NaN_or_pinf     / if not, x is NaN
102    movl 12(%esp),%eax    / ax <-- sign&bexp(x)
103    andl $0x00008000,%eax / here, x is infinite, but +/-?
104    jz   .NaN_or_pinf     / branch if x = +INF
105    fldz
106    ret

108 .NaN_or_pinf:
109     / Here, x = NaN or +inf, so load x and return immediately.
110     fldt 4(%esp)
111     ret
112     .align 4
113     SET_SIZE(exp101)
_____ unchanged_portion_omitted_____

```

new/usr/src/lib/libm/i386/src/exp2.s

1

2756 Tue Nov 25 12:59:15 2014

new/usr/src/lib/libm/i386/src/exp2.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "exp2.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp2,function)
33 #include "libm_synonyms.h"

34     ENTRY(exp2)
35     movl    8(%esp),%ecx        / ecx <-- hi_32(x)
36     andl   $0x7fffffff,%ecx   / ecx <-- hi_32(|x|)
37     cmpl   $0x3ff00000,%ecx   / Is |x| < 1?
38     jb     .shortcut          / If so, take a shortcut.
39     je     .check_tail        / |x| may be only slightly < ln(2)
40     cmpl   $0x7ff00000,%ecx   / hi_32(|x|) >= hi_32(INF)?
41     jae    .not_finite        / if so, x is not finite
42 .finite_non_special:        / Here, 1 < |x| < INF
43     fldl   4(%esp)            / push arg
44     fld   %st(0)              / duplicate stack top
45     frndint                    / [x],x
46     fucom                    / x integral?
47     fstsw  %ax
48     sahf
49     je     .x_integral        / branch if x integral
50     fxch
51     fsub  %st(1),%st          / x-[x], [x]
52     f2xml                    / 2**(x-[x])-1, [x]
53     fldl   %st(1)             / 1,2**(x-[x])-1, [x]
54     faddp  %st,%st(1)         / 2**(x-[x]), [x]
55     fscale                    / 2**x = 2**(arg), [x]
56     fstp  %st(1)
57     ret
```

new/usr/src/lib/libm/i386/src/exp2.s

2

```
59 .x_integral:
60     fstp  %st(0)              / ,x
61     fldl   %st(0)             / 1 = 2**0, x
62     fscale                    / 2**(0 + x) = 2**x, x
63     fstp  %st(1)             / 2**x
64     ret

66 .check_tail:
67     movl   4(%esp),%edx        / edx <-- lo_32(x)
68     cmpl   $0x00000000,%edx   / Is |x| slightly > 1?
69     ja     .finite_non_special / branch if |x| slightly > 1
70 .shortcut:
71     / Here, |x| <= 1,
72     / whence x is in f2xml's domain.
73     fldl   4(%esp)            / push x
74     f2xml                    / 2**x - 1
75     fldl   %st(1)             / 1,2**x - 1
76     faddp  %st,%st(1)         / 2**x
77     ret

79 .not_finite:
80     cmpl   $0x7ff00000,%ecx   / hi_32(|x|) > hi_32(INF)?
81     ja     .NaN_or_pinf       / if so, x is NaN
82     movl   4(%esp),%edx        / edx <-- lo_32(x)
83     cmpl   $0,%edx            / lo_32(x) = 0?
84     jne    .NaN_or_pinf       / if not, x is NaN
85     movl   8(%esp),%eax        / eax <-- hi_32(x)
86     andl   $0x80000000,%eax   / here, x is infinite, but +/-?
87     jz     .NaN_or_pinf       / branch if x = +INF
88     fldz                    / Here, x = -inf, so return 0
89     ret

91 .NaN_or_pinf:
92     / Here, x = NaN or +inf, so load x and return immediately.
93     fldl   4(%esp)
94     fwait
95     ret
96     .align 4
97     SET_SIZE(exp2)

    unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/exp2f.s

1

2348 Tue Nov 25 12:59:16 2014
new/usr/src/lib/libm/i386/src/exp2f.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

29 .file "exp2f.s"

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp2f,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(exp2f)
35 movl 4(%esp),%ecx / ecx <-- x
36 andl $0x7fffffff,%ecx / ecx <-- |x|
37 cmpl $0x3f800000,%ecx / Is |x| <= 1?
38 jbe .shortcut / If so, take a shortcut.
39 cmpl $0x7f800000,%ecx / |x| >= INF?
40 jae .not_finite / if so, x is not finite
41 flds 4(%esp) / push arg
42 fld %st(0) / duplicate stack top
43 frndint / [x],x
44 fucom %ax / x integral?
45 fstsw %ax
46 sahf
47 je .x_integral / branch if x integral
48 fxch / x, [x]
49 fsub %st(1),%st / x-[x], [x]
50 f2xml / 2**(x-[x])-1, [x]
51 fldl / 1,2**(x-[x])-1, [x]
52 faddp %st,%st(1) / 2**(x-[x]), [x]
53 fscale / 2**x = 2**(arg), [x]
54 fstp %st(1)
55 ret
```

57 .x_integral: / here, x is integral

new/usr/src/lib/libm/i386/src/exp2f.s

2

```
58 fstp %st(0) / ,x
59 fldl / 1 = 2**0, x
60 fscale / 2**(0 + x) = 2**x, x
61 fstp %st(1) / 2**x
62 ret

64 .shortcut:
65 / Here, |x| <= 1,
66 / whence x is in f2xml's domain.
67 flds 4(%esp) / push x
68 f2xml / 2**x - 1
69 fldl / 1,2**x - 1
70 faddp %st,%st(1) / 2**x
71 ret

73 .not_finite:
74 ja .NaN_or_pinf / branch if x is NaN
75 movl 4(%esp),%eax / eax <-- x
76 andl $0x80000000,%eax / here, x is infinite, but +/-?
77 jz .NaN_or_pinf / branch if x = +INF
78 fldz / Here, x = -inf, so return 0
79 ret

81 .NaN_or_pinf:
82 / Here, x = NaN or +inf, so load x and return immediately.
83 flds 4(%esp)
84 fwait
85 ret
86 .align 4
87 SET_SIZE(exp2f)
unchanged_portion_omitted
```

```

*****
2928 Tue Nov 25 12:59:16 2014
new/usr/src/lib/libm/i386/src/exp21.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "exp21.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(exp21,function)
33 #include "libm_synonyms.h"

34     ENTRY(exp21)
35     movl    12(%esp),%ecx        / cx <-- sign&bexp(x)
36     andl    $0x00007fff,%ecx    / ecx <-- zero_xtnd(bexp(x))
37     cmpl    $0x00003fff,%ecx    / Is |x| <= 1?
38     jb      .shortcut          / If so, take a shortcut.
39     je      .check_tail        / |x| may be slightly > 1
40     cmpl    $0x00007fff,%ecx    / bexp(|x|) = bexp(INF)?
41     je      .not_finite        / if so, x is not finite
42 .finite_non_special:
43     fldt    4(%esp)            / Here, 1 < |x| < INF
44     fld     %st(0)             / push arg
45     frndint                               / duplicate stack top
46     fucom                               / [x],x
47     fnstsw %ax                 / x integral?
48     sahf
49     je      .x_integral        / branch if x integral
50     fxch
51     fsub    %st(1),%st         / x-[x], [x]
52     f2xml
53     fldl    %st(1)             / 2**(x-[x])-1, [x]
54     faddp   %st,%st(1)         / 1,2**(x-[x])-1, [x]
55     fscale
56     fstp    %st(1)             / 2**(x-[x]), [x]
57     ret

```

```

59 .x_integral:
60     fstp    %st(0)            / ,x
61     fldl    %st(0)            / 1 = 2**0, x
62     fscale
63     fstp    %st(1)            / 2**(0 + x) = 2**x, x
64     ret

66 .check_tail:
67     movl    8(%esp),%ecx      / ecx <-- hi_32(sgnfcnd(x))
68     cmpl    $0x80000000,%ecx  / Is |x| <= 1?
69     ja      .finite_non_special
70     movl    4(%esp),%edx      / edx <-- lo_32(sgnfcnd(x))
71     cmpl    $0x00000000,%edx  / Is |x| slightly > 1?
72     ja      .finite_non_special / branch if |x| slightly > 1
73 .shortcut:
74     / Here, |x| < 1,
75     / whence x is in f2xml's domain.
76     fldt    4(%esp)            / push x
77     f2xml
78     fldl    %st(1)            / 2**x - 1
79     faddp   %st,%st(1)        / 1,2**x - 1
80     ret

82 .not_finite:
83     movl    8(%esp),%ecx      / ecx <-- hi_32(sgnfcnd(x))
84     cmpl    $0x80000000,%ecx  / hi_32(|x|) = hi_32(INF)?
85     jne     .NaN_or_pinf      / if not, x is NaN
86     movl    4(%esp),%edx      / edx <-- lo_32(x)
87     cmpl    $0,%edx           / lo_32(x) = 0?
88     jne     .NaN_or_pinf      / if not, x is NaN
89     movl    12(%esp),%eax     / ax <-- sign&bexp(x)
90     andl    $0x00008000,%eax  / here, x is infinite, but +/-?
91     jz      .NaN_or_pinf      / branch if x = +INF
92     fldz
93     ret

95 .NaN_or_pinf:
96     / Here, x = NaN or +inf, so load x and return immediately.
97     fldt    4(%esp)
98     ret
99     .align 4
100    SET_SIZE(exp21)

```

unchanged portion omitted

```

*****
3663 Tue Nov 25 12:59:16 2014
new/usr/src/lib/libm/i386/src/expl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "expl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expl,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 4
36 ln2_hi: .long 0xd1d00000, 0xb17217f7, 0x00003ffe
37 ln2_lo: .long 0x4c67fc0d, 0x8654361c, 0x0000bfce

39     ENTRY(expl)
40     movl 12(%esp),%ecx      / cx <-- sign&bexp(x)
41     andl $0x7fff,%ecx      / ecx <-- zero_xtnd(bexp(x))
42     cmpl $0x3ffe,%ecx      / Is |x| < 0.5?
43     jb 2f                  / If so, see which shortcut to take
44     je .check_tail        / More checking if 0.5 <= |x| < 1
45     cmpl $0x00007fff,%ecx  / bexp(|x|) = bexp(INF)?
46     je .not_finite        / if so, x is not finite
47     cmpl $0x0000400e,%ecx  / |x| < 32768 = 2^15?
48     jb .finite_non_special / if so, proceed with argument reduction
49     fldt 4(%esp)          / x
50     fldl 1f                / 1, x
51     jmp 1f
52 .finite_non_special:      / Here, ln(2) < |x| < 2^15
53     fldt 4(%esp)          / x
54     fld  %st(0)            / x, x
55     fldl2e                / log2(e), x, x
56     fmulp                 / z := x*log2(e), x
57     frndint               / [z], x

```

```

58     fst  %st(2)            / [z], x, [z]
59     PIC_SETUP(1)
60     fldt PIC_L(ln2_hi)    / ln2_hi, [z], x, [z]
61     fmulp                 / [z]*ln2_hi, x, [z]
62     fsubrp %st,%st(1)     / x-[z]*ln2_hi, [z]
63     fldt PIC_L(ln2_lo)    / ln2_lo, x-[z]*ln2_hi, [z]
64     PIC_WRAPUP
65     fmul  %st(2),%st      / [z]*ln2_lo, x-[z]*ln2_hi, [z]
66     fsubrp %st,%st(1)     / r := x-[z]*ln(2), [z]
67     fldl2e                / log2(e), r, [z]
68     fmulp                 / f := r*log2(e), [z]
69     f2xml                 / 2^f-1, [z]
70     fldl 1, 2^f-1, [z]
71     faddp %st,%st(1)     / 2^f, [z]
72 1:
73     fscale                / e^x, [z]
74     fstp %st(1)
75     ret

77 2:
78     cmpl $0x3fbe,%ecx     / Here, |x| < 0.5
79     jae .shortcut        / Is |x| >= 2^-65?
80     fldt 4(%esp)         / If so, take a shortcut
81     fldl 1, x
82     faddp %st,%st(1)     / x
83     ret

85 .check_tail:
86     movl 8(%esp),%ecx     / ecx <-- hi_32(sgnfncd(x))
87     cmpl $0xb17217f7,%ecx / Is |x| < ln(2)?
88     ja .finite_non_special
89     jb .shortcut
90     movl 4(%esp),%edx     / edx <-- lo_32(x)
91     cmpl $0xd1cf79ab,%edx / Is |x| slightly < ln(2)?
92     ja .finite_non_special / branch if |x| slightly > ln(2)
93 .shortcut:
94     / Here, |x| < ln(2), so |z| = |x/ln(2)| < 1,
95     / whence z is in f2xml's domain.
96     fldt 4(%esp)         / x
97     fldl2e                / log2(e), x
98     fmulp                 / x*log2(e)
99     f2xml                 / 2^(x*log2(e))-1 = e^x-1
100    fldl 1, e^x-1
101    faddp %st,%st(1)     / 1, e^x-1
102    ret

104 .not_finite:
105    movl 8(%esp),%ecx     / ecx <-- hi_32(sgnfncd(x))
106    cmpl $0x80000000,%ecx  / hi_32(|x|) = hi_32(INF)?
107    jne .NaN_or_pinf      / if not, x is NaN
108    movl 4(%esp),%edx     / edx <-- lo_32(x)
109    cmpl $0,%edx          / lo_32(x) = 0?
110    jne .NaN_or_pinf      / if not, x is NaN
111    movl 12(%esp),%eax     / ax <-- sign&bexp(x)
112    andl $0x00008000,%eax  / here, x is infinite, but +/-?
113    jz .NaN_or_pinf       / branch if x = +INF
114    fldz                  / Here, x = -inf, so return 0
115    ret

117 .NaN_or_pinf:
118    / Here, x = NaN or +inf, so load x and return immediately.
119    fldt 4(%esp)
120    fadd %st(0),%st      / quiet SNaN
121    ret
122    .align 4
123    SET_SIZE(expl)

```

new/usr/src/lib/libm/i386/src/expml.s

1

```
*****
3708 Tue Nov 25 12:59:17 2014
new/usr/src/lib/libm/i386/src/expml.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "expml.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expml,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 4
36 .mhundred:    .float -100.0

38     ENTRY(expml)
39     movl    8(%esp),%ecx        / ecx <-- hi_32(x)
40     andl   $0x7fffffff,%ecx    / ecx <-- hi_32(|x|)
41     cmpl   $0x3fe62e42,%ecx    / Is |x| < ln(2)?
42     jb     .shortcut          / If so, take a shortcut.
43     je     .check_tail        / |x| may be only slightly < ln(2)
44     cmpl   $0x7ff00000,%ecx    / hi_32(|x|) >= hi_32(INF)?
45     jae    .not_finite        / if so, x is not finite
46 .finite_non_special:        / Here, ln(2) < |x| < INF
47     fldl   4(%esp)            / push x

49     subl   $8,%esp            / save RP and set round-to-64-bits
50     fstcw  (%esp)
51     movw   (%esp),%ax
52     movw   %ax,4(%esp)
53     orw   $0x0300,%ax
54     movw   %ax,(%esp)
55     fldcw  (%esp)

57     fldl2e        / push log2e }not for xtndd_dbl
```

new/usr/src/lib/libm/i386/src/expml.s

2

```
58     fmulpl %st,%st(1)        / z = x*log2e }not for xtndd_dbl
59     fld   %st(0)            / duplicate stack top
60     frndint        / [z],z
61     / [z] != 0, compute exp(x) and then subtract one to get expml(x)
62     fxch        / z,[z]
63     fsub   %st(1),%st        / z-[z],[z]
64     f2xml        / 2**(z-[z])-1,[z]
65     / avoid spurious underflow when scaling to compute exp(x)
66     PIC_SETUP(1)
67     flds   PIC_L(.mhundred)
68     PIC_WRAPUP
69     fucom   %st(2)            / if -100 !< [z], then use -100
70     fstsw  %ax
71     sahf
72     jb     .got_int_part
73     fxch   %st(2)
74 .got_int_part:
75     fstp   %st(0)            / 2**(z-[z])-1,max([z],-100)
76     fldl   / 1,2**(z-[z])-1,max([z],-100)
77     faddpl %st,%st(1)        / 2**(z-[z]),max([z],-100)
78     fscale / exp(x),max([z],-100)
79     fldl   / 1,exp(x),max([z],-100)
80     fxch   / exp(x),1,max([z],-100)
81     fsubpl %st,%st(1)        / exp(x)-1,max([z],-100)
82     fstp   %st(1)

84     fstcw  (%esp)            / restore old RP
85     movw   (%esp),%dx
86     andw   $0xfcff,%dx
87     movw   4(%esp),%cx
88     andw   $0x0300,%cx
89     orw   %dx,%cx
90     movw   %cx,(%esp)
91     fldcw  (%esp)
92     add    $8,%esp

94     ret

96 .check_tail:
97     movl   4(%esp),%edx        / edx <-- lo_32(x)
98     cmpl   $0xfefa39ef,%edx    / Is |x| slightly < ln(2)?
99     ja     .finite_non_special / branch if |x| slightly > ln(2)
100 .shortcut:
101     / Here, |x| < ln(2), so |z| = |x*log2(e)| < 1,
102     / whence z is in f2xml's domain.
103     fldl   4(%esp)            / push x
104     fldl2e / push log2e }not for xtndd_dbl
105     fmulpl %st,%st(1)        / z = x*log2e }not for xtndd_dbl
106     f2xml        / 2**(x*log2(e))-1 = e**x - 1
107     ret

109 .not_finite:
110     / Here, flags still have settings from execution of
111     /
112     ja     .NaN_or_pinf        / if not, x may be +/- INF
113     movl   4(%esp),%edx        / edx <-- lo_32(x)
114     cmpl   $0,%edx            / lo_32(x) = 0?
115     jne    .NaN_or_pinf        / if not, x is NaN
116     movl   8(%esp),%eax        / eax <-- hi_32(x)
117     andl   $0x80000000,%eax    / here, x is infinite, but +/-?
118     jz     .NaN_or_pinf        / branch if x = +INF
119     fldl   / Here, x = -inf, so return -1
120     fchs
121     ret

123 .NaN_or_pinf:
```

new/usr/src/lib/libm/i386/src/expml.s

3

```
124      / Here, x = NaN or +inf, so load x and return immediately.
125      fldl    4(%esp)
126      fwait
127      ret
128      .align 4
129      SET_SIZE(expml)
unchanged_portion_omitted
```



```

*****
3982 Tue Nov 25 12:59:17 2014
new/usr/src/lib/libm/i386/src/expmlf.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "expmlf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expmlf,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 4
36 .mhundred:    .float  -100.0

38     ENTRY(expmlf)
39     movl    4(%esp),%ecx        / ecx <-- x
40     andl    $0x7fffffff,%ecx   / ecx <-- |x|
41     cmpl    $0x3f317217,%ecx   / Is |x| < ln(2)?
42     jbe     .shortcut          / If so, take a shortcut.
43     cmpl    $0x7f800000,%ecx   / |x| >= INF?
44     jae     .not_finite       / if so, x is not finite
45     flds   4(%esp)            / push x

47     subl   $8,%esp            / save RP and set round-to-64-bits
48     fstcw  (%esp)
49     movw   (%esp),%ax
50     movw   %ax,4(%esp)
51     orw   $0x0300,%ax
52     movw   %ax,(%esp)
53     fldcw  (%esp)

55     fldl2e    %st,%st(1)      / push log2e }not for xtndd_dbl
56     fmulp    %st,%st(1)      / z = x*log2e }not for xtndd_dbl
57     fld     %st(0)           / duplicate stack top

```

```

58     frndint    / [z],z
59     fucom     / This and the next 3 instructions
60     fstsw    %ax          / add 10 clocks to runtime of the
61     sahf     / main branch, but save about 265
62     je      .z_integral   / upon detection of integral z.
63     / [z] != 0, compute exp(x) and then subtract one to get expml(x)
64     fxch    / z,[z]
65     fsub    %st(1),%st    / z-[z],[z]
66     f2xml   / 2**(z-[z])-1,[z]
67     / avoid spurious underflow when scaling to compute exp(x)
68     PIC_SETUP(1)
69     flds    PIC_L(.mhundred)
70     PIC_WRAPUP
71     fucom   %st(2)        / if -100 !< [z], then use -100
72     fstsw  %ax
73     sahf
74     jnb    .got_int_part
75     fxch   %st(2)
76 .got_int_part:
77     fstp   %st(0)        / 2**(z-[z])-1,max([z],-100)
78     fldl   / 1,2**(z-[z])-1,max([z],-100)
79     faddp  %st,%st(1)    / 2**(z-[z]) ,max([z],-100)
80     fscale / exp(x) ,max([z],-100)
81     fldl   / 1,exp(x) ,max([z],-100)
82     fsubrp %st,%st(1)    / exp(x)-1 ,max([z],-100)
83     fstp   %st(1)

85     fstcw  (%esp)        / restore old RP
86     movw   (%esp),%dx
87     andw   $0xfcff,%dx
88     movw   4(%esp),%cx
89     andw   $0x0300,%cx
90     orw   %dx,%cx
91     movw   %cx,(%esp)
92     fldcw  (%esp)
93     add    $8,%esp

95     ret

97 .z_integral:           / here, z is integral
98     fstp   %st(0)        / ,z
99     / avoid spurious underflow when scaling to compute exp(x)
100    PIC_SETUP(2)
101    flds   PIC_L(.mhundred)
102    PIC_WRAPUP
103    fucom  %st(1)        / if -100 !< [z], then use -100
104    fstsw  %ax
105    sahf
106    jnb   .scale_wont_ovfl
107    fxch  %st(1)
108 .scale_wont_ovfl:
109    fstp  %st(0)        / max([z],-100)
110    fldl  / 1,max([z],-100)
111    fscale / exp(x) ,max([z],-100)
112    fldl  / 1,exp(x) ,max([z],-100)
113    fsubrp %st,%st(1)  / exp(x)-1 ,max([z],-100)
114    fstp  %st(1)

116    fstcw  (%esp)        / restore old RP
117    movw   (%esp),%dx
118    andw   $0xfcff,%dx
119    movw   4(%esp),%cx
120    andw   $0x0300,%cx
121    orw   %dx,%cx
122    movw   %cx,(%esp)
123    fldcw  (%esp)

```

```
124     add     $8,%esp
126     ret

128 .shortcut:
129     / Here,  $|x| < \ln(2)$ , so  $|z| = |x*\log_2(e)| < 1$ ,
130     / whence  $z$  is in  $f2xml$ 's domain.
131     flds    4(%esp)           / push x
132     fldl2e           / push  $\log_2 e$  }not for xtndd_dbl
133     fmulp    %st,%st(1)     /  $z = x*\log_2 e$  }not for xtndd_dbl
134     f2xml           /  $2^{*(x*\log_2(e))-1} = e^{*x} - 1$ 
135     ret

137 .not_finite:
138     ja     .NaN_or_pinf     / branch if x is NaN
139     movl   4(%esp),%eax     / eax <-- x
140     andl   $0x80000000,%eax / here, x is infinite, but +/-?
141     jz     .NaN_or_pinf     / branch if x = +INF
142     fldl           / Here, x = -inf, so return -1
143     fchs
144     ret

146 .NaN_or_pinf:
147     / Here, x = NaN or +inf, so load x and return immediately.
148     flds    4(%esp)
149     fwait
150     ret
151     .align 4
152     SET_SIZE(expmlf)
unchanged portion omitted
```

```

*****
3682 Tue Nov 25 12:59:18 2014
new/usr/src/lib/libm/i386/src/expm1.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "expm1.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(expm1,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 4
36 ln2_hi: .long 0xd1d00000, 0xb17217f7, 0x00003ffe
37 ln2_lo: .long 0x4c67fc0d, 0x8654361c, 0x0000bffe

39     ENTRY(expm1)
40     movl 12(%esp),%ecx      / cx <-- sign&bexp(x)
41     movl %ecx,%eax        / ax <-- sign&bexp(x)
42     andl $0x00007fff,%ecx / ecx <-- zero_xtnd(bexp(x))
43     cmpl $0x00003ffe,%ecx / Is |x| < ln(2)?
44     jb   .shortcut       / If so, take a shortcut.
45     je   .check_tail     / |x| may be only slightly < ln(2)
46     cmpl $0x00007fff,%ecx / bexp(|x|) = bexp(INF)?
47     je   .not_finite     / if so, x is not finite
48     andl $0x0000ffff,%eax / eax <-- sign&bexp(x)
49     cmpl $0x0000c006,%eax / x <= -128?
50     jae  1f              / if so, simply return -1
51     cmpl $0x0000400d,%ecx / |x| < 16384 = 2^14?
52     jb   .finite_non_special / if so, proceed with argument reduction
53     fldt 4(%esp)        / x >= 16384; x
54     fldl 1, x           / 1, x
55     fscale / +Inf, x
56     fstp %st(1)        / +Inf
57     ret

```

```

59 .finite_non_special: / -128 < x < -ln(2) || ln(2) < x < 2^14
60     fldt 4(%esp)      / x
61     fld  %st(0)       / x, x
62     fldl2e / log2(e), x, x
63     fmulp / z := x*log2(e), x
64     frndint / [z], x
65     fst  %st(2)      / [z], x, [z]
66     PIC_SETUP(1)
67     fldt PIC_L(ln2_hi) / ln2_hi, [z], x, [z]
68     fmulp / [z]*ln2_hi, x, [z]
69     fsubrp %st,%st(1) / x-[z]*ln2_hi, [z]
70     fldt PIC_L(ln2_lo) / ln2_lo, x-[z]*ln2_hi, [z]
71     PIC_WRAPUP
72     fmul %st(2),%st / [z]*ln2_lo, x-[z]*ln2_hi, [z]
73     fsubrp %st,%st(1) / r := x-[z]*ln(2), [z]
74     fldl2e / log2(e), r, [z]
75     fmulp / f := r*log2(e), [z]
76     f2xml / 2^f-1,[z]
77     fldl 1, 2^f-1, [z]
78     faddp %st,%st(1) / 2^f, [z]
79     fscale / e^x, [z]
80     fstp %st(1) / e^x
81     fldl 1, e^x
82     fsubrp %st,%st(1) / e^x-1
83     ret

85 .check_tail:
86     movl 8(%esp),%ecx / ecx <-- hi_32(sgnfcnd(x))
87     cmpl $0xb17217f7,%ecx / Is |x| < ln(2)?
88     ja   .finite_non_special
89     jb   .shortcut
90     movl 4(%esp),%edx / edx <-- lo_32(x)
91     cmpl $0xd1cf79ab,%edx / Is |x| slightly < ln(2)?
92     ja   .finite_non_special / branch if |x| slightly > ln(2)
93 .shortcut:
94     / Here, |x| < ln(2), so |z| = |x/ln(2)| < 1,
95     / whence z is in f2xml's domain.
96     fldt 4(%esp)      / x
97     fldl2e / log2(e), x
98     fmulp / z := x*log2(e)
99     f2xml / 2^(x*log2(e))-1 = e^x-1
100    ret

102 .not_finite:
103    movl 8(%esp),%ecx / ecx <-- hi_32(sgnfcnd(x))
104    cmpl $0x80000000,%ecx / hi_32(|x|) = hi_32(INF)?
105    jne  .NaN_or_pinf / if not, x is NaN
106    movl 4(%esp),%edx / edx <-- lo_32(x)
107    cmpl $0,%edx / lo_32(x) = 0?
108    jne  .NaN_or_pinf / if not, x is NaN
109    movl 12(%esp),%eax / ax <-- sign&bexp(x)
110    andl $0x00008000,%eax / here, x is infinite, but +/-?
111    jz   .NaN_or_pinf / branch if x = +INF

112 1:
113    fldl / Here, x = -inf, so return -1
114    fchs
115    ret

117 .NaN_or_pinf:
118    / Here, x = NaN or +inf, so load x and return immediately.
119    fldt 4(%esp)
120    ret
121    .align 4
122    SET_SIZE(expm1)

```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/fabs.s

1

1136 Tue Nov 25 12:59:18 2014

new/usr/src/lib/libm/i386/src/fabs.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "fabs.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fabs,function)
33 #include "libm_synonyms.h"

34     ENTRY(fabs)
35     fldl     4(%esp)
37 #undef     fabs
36     fabs
37     ret
38     .align  4
39     SET_SIZE(fabs)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/fabsf.s

1

1141 Tue Nov 25 12:59:19 2014

new/usr/src/lib/libm/i386/src/fabsf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "fabsf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fabsf,function)
33 #include "libm_synonyms.h"

34     ENTRY(fabsf)
35     flds     4(%esp)
37 #undef     fabs
36     fabs
37     ret
38     .align  4
39     SET_SIZE(fabsf)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/fabsl.s

1

1141 Tue Nov 25 12:59:19 2014

new/usr/src/lib/libm/i386/src/fabsl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "fabsl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fabsl,function)
33 #include "libm_synonyms.h"

34     ENTRY(fabsl)
35     fldt     4(%esp)
37 #undef     fabs
36     fabs
37     ret
38     .align  4
39     SET_SIZE(fabsl)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/finitfef.s

1

1373 Tue Nov 25 12:59:20 2014

new/usr/src/lib/libm/i386/src/finitfef.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file "finitfef.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(finitfef,function)
33 #include "libm_synonyms.h"

34     ENTRY(finitfef)
35     movl    4(%esp),%eax        / eax <-- x
36     notl   %eax                / not(bexp) = 0 iff bexp = all 1's
37     andl   $0x7f800000,%eax    / ZF <-- 1    iff not(bexp) = 0
38     jz     .done               / no jump if arg. is finite
39     movl   $1,%eax            / %ax was 0; ansi needs %eax = 1
40     .done:
41     ret
42     .align 4
43     SET_SIZE(finitfef)
44 unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/finitel.s

1

1705 Tue Nov 25 12:59:20 2014

new/usr/src/lib/libm/i386/src/finitel.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "finitel.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(finitel,function)
33 #include "libm_synonyms.h"

33     ENTRY(finitel)
34     movl    12(%esp),%eax        / %ax <-- sign&bexp(x)
35     testl  $0x80000000,8(%esp)  / ZF = 1 iff hi_32(sgnfnd(x))'s msb = 0
36     jz     .chk_denormal_or_0
37     notl  %eax                 / not(bexp) = 0 iff bexp = all 1's
38     andl  $0x00007fff,%eax     / ZF <-- 1 iff not(bexp) = 0
39     jz     .done               / no jump if arg. is finite
40     movl  $1,%eax              / ansi needs %eax = 1
41 .done:
42     ret

44 .chk_denormal_or_0:
45     andl  $0x00007fff,%eax     / ZF <-- 1 iff bexp = 0 iff denormal or
46     jnz   .unsupported        / jump if arg has unsupported format
47     movl  $1,%eax              / ansi needs %eax = 1
48     ret

50 .unsupported:
51     movl  $0,%eax              / unsupported format does not represent
52     ret                          / a finite number
53     .align 4
54     SET_SIZE(finitel)
unchanged portion omitted
```


new/usr/src/lib/libm/i386/src/floor.s

1

```
*****
1446 Tue Nov 25 12:59:20 2014
new/usr/src/lib/libm/i386/src/floor.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "floor.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(floor,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(floor)
35 subl $8,%esp
36 fstcw (%esp)
37 fldl 12(%esp)
38 movw (%esp),%cx
39 orw $0x0c00,%cx
40 xorw $0x0800,%cx
41 movw %cx,4(%esp)
42 fldcw 4(%esp) / set RD = down
43 frndint
44 fstcw 4(%esp) / restore RD
45 movw 4(%esp),%dx
46 andw $0xf3ff,%dx
47 movw (%esp),%cx
48 andw $0x0c00,%cx
49 orw %dx,%cx
50 movw %cx,(%esp)
51 fldcw (%esp) / restore RD
52 addl $8,%esp
53 ret
54 .align 4
55 SET_SIZE(floor)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/floorl.s

1

```
*****
1863 Tue Nov 25 12:59:21 2014
new/usr/src/lib/libm/i386/src/floorl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "floorl.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ceil, function)
33 LIBM_ANSI_PRAGMA_WEAK(floorl, function)
34 #include "libm_synonyms.h"
```

```
35 ENTRY(ceil)
36 subl $8,%esp
37 fstcw (%esp)
38 fldt 12(%esp)
39 movw (%esp),%cx
40 orw $0x0c00,%cx
41 xorw $0x0400,%cx
42 movw %cx,4(%esp)
43 fldcw 4(%esp) / set RD = up
44 frndint / restore RD
45 fstcw 4(%esp) / restore RD
46 movw 4(%esp),%dx
47 andw $0xf3ff,%dx
48 movw (%esp),%cx
49 andw $0x0c00,%cx
50 orw %dx,%cx
51 movw %cx,(%esp)
52 fldcw (%esp) / restore RD
53 addl $8,%esp
54 ret
55 .align 4
56 SET_SIZE(ceil)
```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/fmod.s

1

```
*****
1772 Tue Nov 25 12:59:21 2014
new/usr/src/lib/libm/i386/src/fmod.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmod.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmod,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

35     ENTRY(fmod)
36     movl    16(%esp),%eax        / eax <-- hi_32(y)
37     andl    $0x7fffffff,%eax    / eax <-- hi_32(|y|)
38     orl     12(%esp),%eax        / eax <-- lo_32(y)|hi_32(|y|)
39     je     .zero

41     fldl    12(%esp)            / load arg y
42     fldl    4(%esp)            / load arg x
43 .mod_loop:
44     fprem                                 / partial fmod
45     fstsw   %ax                 / store status word
46     andw   $0x400,%ax          / check for incomplete reduction
47     jne    .mod_loop           / while incomplete, do fprem again
48     fstp   %st(1)
49     ret

50 .zero:
51     pushl   %ebp
52     movl   %esp,%ebp
53     PIC_SETUP(1)
54     pushl   $27                 / case 27 in _SVID_libm_err
55     pushl   20(%ebp)            / pass x
56     pushl   16(%ebp)
57     pushl   12(%ebp)            / pass y
```

new/usr/src/lib/libm/i386/src/fmod.s

2

```
58     pushl   8(%ebp)
59     call   PIC_F(_SVID_libm_err)
60     addl   $20,%esp
61     PIC_WRAPUP
62     leave
63     ret
64     .align 4
65     SET_SIZE(fmod)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/fmodf.s

1

1366 Tue Nov 25 12:59:22 2014

new/usr/src/lib/libm/i386/src/fmodf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmodf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmodf,function)
33 #include "libm_synonyms.h"

34     ENTRY(fmodf)
35     flds    8(%esp)           / load arg y
36     flds    4(%esp)           / load arg x
37 .mod_loop:
38     fprem                               / partial fmod
39     fstsw   %ax                 / store status word
40     andw   $0x400,%ax           / check for incomplete reduction
41     jne    .mod_loop           / while incomplete, do fprem again
42     fstp   %st(1)
43     ret
44     .align 4
45     SET_SIZE(fmodf)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/fmodl.s

1

1366 Tue Nov 25 12:59:22 2014

new/usr/src/lib/libm/i386/src/fmodl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "fmodl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(fmodl,function)
33 #include "libm_synonyms.h"

34     ENTRY(fmodl)
35     fldt    16(%esp)        / load arg y
36     fldt    4(%esp)        / load arg x
37 .mod_loop:
38     fprem                                / partial fmod
39     fstsw   %ax              / store status word
40     andw   $0x400,%ax       / check for incomplete reduction
41     jne    .mod_loop        / while incomplete, do fprem again
42     fstp   %st(1)
43     ret
44     .align 4
45     SET_SIZE(fmodl)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/hypot.s

1

```
*****
3195 Tue Nov 25 12:59:23 2014
new/usr/src/lib/libm/i386/src/hypot.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file "hypot.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(hypot,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

36 #undef fabs

35     .data
36     .align 4
37 inf:
38     .long 0x7f800000

40     ENTRY(hypot)
41     movl 8(%esp),%eax      / eax <-- hi_32(x)
42     andl $0x7fffffff,%eax / eax <-- hi_32(|x|)
43     jz   .x_maybe_0      / if x = +/-0, return |y|
44     subl $0x7ff00000,%eax / eax <-- hi_32(|x|) - hi_32(INF)
45     jz   .x_maybe_inf

46 .check_y:
47     movl 16(%esp),%eax    / eax <-- hi_32(y)
48     andl $0x7fffffff,%eax / eax <-- hi_32(|y|)
49     jz   .y_maybe_0      / if y = +/-0, return |x|
50     subl $0x7ff00000,%eax / eax <-- hi_32(|y|) - hi_32(INF)
51     jz   .y_maybe_inf

52 .do_hypot:
53     fldl 12(%esp)        / ,y
54     fmul %st(0),%st     / ,y*y
55     fldl 4(%esp)        / x,y*y
```

new/usr/src/lib/libm/i386/src/hypot.s

2

```
56     fmul %st(0),%st     / x*x,y*y
57     faddp %st,%st(1)   / x*x+y*y
58     fsqrt                                / sqrt(x*x+y*y)
59     subl $8,%esp
60     fstpl (%esp)        / round to double
61     fldl (%esp)        / sqrt(x*x+y*y) rounded to double
62     PIC_SETUP(1)
63     flds PIC_L(Inf)     / Inf , sqrt(x*x+y*y)
64     PIC_WRAPUP
65     addl $8,%esp
66     fucomp
67     fstsw %ax           / store status in %ax
68     sahf                / 80387 flags in %ah to 80386 flags
69     jz   .maybe_ovflw
70     ret

72 .maybe_ovflw:
73     jnp .ovflw
74     ret

76 .ovflw:
77     / overflow occurred
78     fstp %st(0)        / stack empty
79     pushl %ebp
80     movl %esp,%ebp
81     PIC_SETUP(2)
82     pushl $4
83     pushl 20(%ebp)     / high y
84     pushl 16(%ebp)     / low y
85     pushl 12(%ebp)     / high x
86     pushl 8(%ebp)      / low x
87     call PIC_F(_SVID_libm_err)
88     addl $20,%esp
89     PIC_WRAPUP
90     leave
91     ret

93 .x_maybe_0:
94     movl 4(%esp),%ecx   / ecx <-- lo_32(x)
95     orl %ecx,%eax      / is x = +/-0?
96     jnz .check_y       / branch if x is denormal
97     / x = +/-0, so return |y|
98     fldl 12(%esp)
99     fabs
100    ret

102 .x_maybe_inf:
103    movl 4(%esp),%ecx   / ecx <-- lo_32(x)
104    orl %ecx,%eax      / is x = +/-INF?
105    jnz .check_y       / branch if x is NaN
106    / push&pop y in case y is a SNaN
107    fldl 12(%esp)
108    fstp %st(0)
109    / x = +/-INF, so return |x|
110    fldl 4(%esp)
111    fabs
112    ret

114 .y_maybe_0:
115    movl 12(%esp),%ecx  / ecx <-- lo_32(y)
116    orl %ecx,%eax      / is y = +/-0?
117    jnz .do_hypot      / branch if y is denormal
118    / y = +/-0, so return |x|
119    fldl 4(%esp)
120    fabs
121    ret
```

```
123 .y_maybe_inf:
124     movl    12(%esp),%ecx        / ecx <-- lo_32(y)
125     orl    %ecx,%eax           / is y = +/-INF?
126     jnz    .do_hypot          / branch if y is NaN
127     / push&pop x in case x is a SNaN
128     fldl   4(%esp)
129     fstp   %st(0)
130     / y = +/-INF, so return |y|
131     fldl   12(%esp)
132     fabs
133     ret
134     .align 4
135     SET_SIZE(hypot)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/hypotf.s

1

1873 Tue Nov 25 12:59:23 2014

new/usr/src/lib/libm/i386/src/hypotf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "hypotf.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(hypotf,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"
```

```
36 #undef fabs
```

```
35 ENTRY(hypotf)
36 movl 4(%esp),%eax / eax <-- x
37 andl $0x7fffffff,%eax / eax <-- |x|
38 jz .return_abs_y / if x = +/-0, return |y|
39 subl $0x7f800000,%eax / eax <-- |x| - INF
40 jz .return_abs_x / if x = +/-INF, return |x|
41 movl 8(%esp),%eax / eax <-- y
42 andl $0x7fffffff,%eax / eax <-- |y|
43 jz .return_abs_x / if y = +/-0, return |x|
44 subl $0x7f800000,%eax / eax <-- |y| - INF
45 .return_abs_y:
46 flds 8(%esp) / y
47 jz .take_abs / if y = +/-INF, return |y|
48 fmul %st(0),%st / y*y
49 flds 4(%esp) / x,y*y
50 fmul %st(0),%st / x*x,y*y
51 faddp %st,%st(1) / x*x+y*y
52 fsqrt / sqrt(x*x+y*y)
53 subl $4,%esp
54 fstps (%esp) / round to single
55 flds (%esp)
```

new/usr/src/lib/libm/i386/src/hypotf.s

2

```
56 fwait
57 addl $4,%esp
58 ret
```

```
60 .return_abs_x:
61 / returns |x|
62 flds 4(%esp)
63 .take_abs:
64 fabs
65 ret
66 .align 4
67 SET_SIZE(hypotf)
```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/ieee_funcl.s

1

```
*****
3188 Tue Nov 25 12:59:23 2014
new/usr/src/lib/libm/i386/src/ieee_funcl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

29 .file "ieee_funcl.s"

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(isinfl,function)
33 LIBM_ANSI_PRAGMA_WEAK(isnormall,function)
34 LIBM_ANSI_PRAGMA_WEAK(issubnormall,function)
35 LIBM_ANSI_PRAGMA_WEAK(iszerol,function)
36 LIBM_ANSI_PRAGMA_WEAK(signbitl,function)
37 #include "libm_synonyms.h"
```

```
33 ENTRY(isinfl)
34 movl 12(%esp),%eax / ax <-- sign and bexp of x
35 notl %eax
36 andl $0x00007fff,%eax
37 jz .L6
38 movl $0,%eax
39 .not_inf:
40 ret

42 .L6:
43 movl 8(%esp),%ecx / here, (eax) = 0.0
44 xorl $0x80000000,%ecx / handle unsupported implicitly
45 orl 4(%esp), %ecx
46 jnz .not_inf
47 movl $1,%eax
48 ret
49 .align 4
50 SET_SIZE(isinfl)
unchanged portion omitted
```

new/usr/src/lib/libm/i386/src/ilogb.s

1

2337 Tue Nov 25 12:59:24 2014

new/usr/src/lib/libm/i386/src/ilogb.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "ilogb.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ilogb,function)
33 #include "libm_synonyms.h"
33 #include "xpg6.h"

35     .data
36     .align 8
37 two52: .long 0x0,0x43300000 / 2**52

39     ENTRY(ilogb)
40     movl 8(%esp),%eax / eax <-- hi_32(x)
41     andl $0x7fffffff,%eax / eax <-- hi_32(abs(x))
42     testl $0x7ff00000,%eax / is bexp(x) 0?
43     jz .bexp_0 / jump if x is 0 or subnormal
44 / biased exponent is non-zero
45     cmpl $0x7ff00000,%eax / is bexp(x) 0x7ff?
46     jae .bexp_all_1 / jump if x is NaN or Inf
47     shrl $20,%eax / eax <-- bexp(x)
48     subl $1023,%eax / unbias exponent by 1023
49     ret

51 .bexp_all_1:
52     movl $0x7fffffff,%eax / x is NaN or inf, so return 0x7fffffff
53     jmp 0f

55 .bexp_0:
56     orl 4(%esp),%eax / test whether x is 0
57     jnz .ilogb_subnorm
```

new/usr/src/lib/libm/i386/src/ilogb.s

2

```
58     movl $0x80000001,%eax / x is +/-0, so return 0x80000001
59 0:
60     PIC_SETUP(0)
61     PIC_G_LOAD(movzwl, __xpg6,ecx)
62     PIC_WRAPUP
63     andl $_C99SUSv3_ilogb_0InfNaN_raises_invalid,%ecx
64     cmpl $0,%ecx
65     je 1f
66     fldz
67     fdivp %st,%st(0) / raise invalid as per SUSv3
68 1:
69     ret

71 .ilogb_subnorm: / subnormal input
72     fldl 4(%esp) / push x
73     PIC_SETUP(1)
74     fmul PIC_L(two52) / x*2**52
75     PIC_WRAPUP
76     subl $8,%esp / set up storage area
77     fstpl (%esp) / store x*2**52 in storage are
78     movl $0x7ff00000,%eax
79     andl 4(%esp),%eax
80     shrl $20,%eax / extract exponent of x*2**52
81     subl $1075,%eax / unbias it by 1075 (= 1023 + 52)
82     addl $8,%esp
83     ret
84     .align 4
85     SET_SIZE(ilogb)

    _____
    unchanged_portion_omitted_
```

new/usr/src/lib/libm/i386/src/ilogbf.s

1

2607 Tue Nov 25 12:59:25 2014

new/usr/src/lib/libm/i386/src/ilogbf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "ilogbf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ilogbf,function)
33 #include "libm_synonyms.h"
34 #include "xpg6.h"

35     .data
36     .align 8
37 two23: .long 0x4b000000        / 2**23

39     ENTRY(ilogbf)
40     movl 4(%esp),%eax        / eax <-- x
41     testl $0x7f800000,%eax   / is bexp(x) 0?
42     jz .bexp_0              / jump if x is 0 or subnormal
43                             / here, biased exponent is non-zero
44     andl $0x7fffffff,%eax   / eax <-- abs(x)
45     cmpl $0x7f800000,%eax   / is bexp(x) 0xff?
46     jae .bexp_all_1        / jump if x is NaN or Inf
47     shrl $23,%eax          / eax <-- zero_xtnd(bexp(x))
48     subl $127,%eax         / unbias exponent by 127
49     ret

51 .bexp_all_1:
52     movl $0x7fffffff,%eax   / x is NaN or inf, so return 0x7fffffff
53     jmp 0f

55 .bexp_0:
56     andl $0x7fffffff,%eax   / eax <-- abs(x), and
57                             / ZF = 1 iff x = 0.0
```

new/usr/src/lib/libm/i386/src/ilogbf.s

2

```
58     jnz .ilogb_subnorm      / subnormal input
59     movl $0x80000001,%eax   / x is +/-0, so return 0x80000001
60 0:
61     PIC_SETUP(0)
62     PIC_G_LOAD(movzwl, __xpg6,ecx)
63     PIC_WRAPUP
64     andl $_C99SUSv3_ilogb_0InfNaN_raises_invalid,%ecx
65     cmpl $0,%ecx
66     je 1f
67     fldz
68     fdivp %st,%st(0)       / raise invalid as per SUSv3
69 1:
70     ret

72 .ilogb_subnorm:
73     flds 4(%esp)          / push x
74     PIC_SETUP(1)
75     fmulb PIC_L(two23)    / x*2**23; rebias x by 127+23,
76                             / instead of 127
77     PIC_WRAPUP
78     subl $4,%esp         / set up storage area
79     fstps (%esp)         / store x*2**23 in storage area
80     fwait                / (shouldn't raise exception, but
81                             / just in case)
82     movl $0x7f800000,%eax / eax <-- single_bexp_mask
83     andl (%esp),%eax     / eax[23..30] <-- bexp(x*2**23),
84                             / rest_of(eax) <-- 0
85     shrl $23,%eax        / eax <-- zero_xtnd(bexp(x*2**23))
86     subl $150,%eax       / unbias rebias x by 150 (= 127 + 23)
87     addl $4,%esp        / restore stack for caller
88     ret
89     .align 4
90     SET_SIZE(ilogbf)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/ilogbl.s

1

```
*****
2421 Tue Nov 25 12:59:25 2014
new/usr/src/lib/libm/i386/src/ilogbl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "ilogbl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(ilogbl,function)
33 #include "libm_synonyms.h"
34 #include "xpg6.h"

35     .data
36     .align 8
37 two63: .long 0x0,0x43d00000    / 2**63

39     ENTRY(ilogbl)
40     movl 12(%esp),%eax        / eax <-- sign and bexp of x
41     andl $0x00007fff,%eax    / eax <-- bexp(x)
42     jz   .bexp_0             / jump iff x is 0 or subnormal
43     / here, biased exponent is non-zero
44     testl $0x80000000,8(%esp) / test msb of hi_32(sgnfcnd(x))
45     jz   .ilogbl_not_finite  / jump if unsupported format
46     cmpl $0x00007fff,%eax    /
47     je   .ilogbl_not_finite  /
48     subl $16383,%eax         / unbias exponent by 16383 = 0x3fff
49     ret

51 .ilogbl_not_finite:
52     movl $0x7fffffff,%eax    / x is NaN/inf/unsup
53     jmp 0f

55 .bexp_0:
56     movl 8(%esp),%eax         / eax <-- hi_32(sgnfcnd(x))
57     orl 4(%esp),%eax         / test whether x is 0
```

new/usr/src/lib/libm/i386/src/ilogbl.s

2

```
58     jnz  .ilogbl_subnorm    / jump iff x is subnormal
59     movl $0x80000001,%eax    / x is +/-0, so return 0x80000001
60 0:
61     PIC_SETUP(0)
62     PIC_G_LOAD(movzwl, __xpg6, ecx)
63     PIC_WRAPUP
64     andl $_C99SUSv3_ilogb_0InfNaN_raises_invalid,%ecx
65     cmpl $0,%ecx
66     je   1f
67     fldz
68     fdivp %st,%st(0)        / raise invalid as per SUSv3
69 1:
70     ret

73 .ilogbl_subnorm:
74     fldt 4(%esp)            / push x, setting D-flag
75     PIC_SETUP(1)
76     fnull PIC_L(two63)     / x*2**63
77     PIC_WRAPUP
78     subl $12,%esp
79     fstpt (%esp)
80     movl $0x00007fff,%eax
81     andl 8(%esp),%eax      / eax <-- sign and bexp of x*2**63
82     subl $16445,%eax      / unbias it by (16,383 + 63)
83     addl $12,%esp
84     ret
85     .align 4
86     SET_SIZE(ilogbl)

unchanged portion omitted
```

new/usr/src/lib/libm/i386/src/isnan.s

1

```
*****
1852 Tue Nov 25 12:59:26 2014
new/usr/src/lib/libm/i386/src/isnan.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "isnan.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(isnan,function)
33     .weak _isnan
34     .type _isnan,@function
35 _isnan = __isnan
36     .weak _isnanand
37     .type _isnanand,@function
38 _isnanand = __isnanand
39     .weak isnand
40     .type isnand,@function
41 isnand = __isnanand
42 #include "libm_synonyms.h"

43     ENTRY(isnan)
44     movl    8(%esp),%eax        / eax <-- hi_32(x)
45     andl   $0x7fffffff,%eax    / eax <-- hi_32(abs(x))
46     subl   $0x7ff00000,%eax    / weed out finite values
47     jae    .nan_or_inf        / no jump if arg. is finite
48     movl   $0,%eax           / ansi needs (eax) = 0
49     ret

50 .nan_or_inf:
51     ja     .got_nan          / no jump if arg. may be infinite;
52                               / let nan waste time
53                               / (eax) = 0 here
54     testl  $0xffffffff,4(%esp) / ZF <-- 1 iff lo_frac. = 0
55                               /     iff arg. is infinite
56     jnz   .got_nan          / no jump if arg. is infinite;
57     ret
```

new/usr/src/lib/libm/i386/src/isnan.s

2

```
58 .got_nan:
59     movl   $1,%eax          / %eax was 0, must be made 1 to
60                               / indicate TRUE
61     ret
62     .align 4
63     SET_SIZE(isnan)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/isnanf.s

1

1575 Tue Nov 25 12:59:26 2014

new/usr/src/lib/libm/i386/src/isnanf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "isnanf.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(isnanf,function)
33     .weak _isnanf
34     .type _isnanf,@function
35 _isnanf = __isnanf
36 #include "libm_synonyms.h"
```

```
37     ENTRY(isnanf)
38     movl    4(%esp),%eax        / eax <-- x
39     andl    $0x7fffffff,%eax   / eax <-- abs(x)
40     subl    $0x7f800000,%eax   / ZF <-- 1 iff x is infinite
41     jae     .nan_or_inf        / no jump iff arg. is finite
42     movl    $0,%eax
43     ret
44 .nan_or_inf:
45     jnz     .got_nan           / no jump if arg. infinite;
46                                     / let nan waste time
47     ret                         / %eax = 0 here
48 .got_nan:
49     movl    $1,%eax           / %eax was 0, must be made 1 to
50                                     / indicate TRUE
51     ret
52     .align 4
53     SET_SIZE(isnanf)
_____ unchanged_portion_omitted_
```

new/usr/src/lib/libm/i386/src/isnanl.s

1

1705 Tue Nov 25 12:59:27 2014

new/usr/src/lib/libm/i386/src/isnanl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "isnanl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(isnanl,function)
33 #include "libm_synonyms.h"

34     ENTRY(isnanl)
35     movl    12(%esp),%eax        / ax <-- sign bit and exp
36     andl    $0x00007fff,%eax
37     jz     .not_nan            / jump if exp is all 0
38     xorl    $0x00007fff,%eax
39     jz     .nan_or_inf         / jump if exp is all 1
40     testl   $0x80000000,8(%esp)
41     jz     .got_nan           / jump if leading bit is 0
42     movl    $0,%eax
43 .not_nan:
44     ret
45 .nan_or_inf:
46     cmpl    $0x80000000,8(%esp) / note that %eax = 0 from before
47     jnz    .got_nan           / what is first half of significand?
48     testl   $0xffffffff,4(%esp) / jump if not equal to 0x80000000
49     jnz    .got_nan           / is second half of significand 0?
50     ret
51 .got_nan:
52     movl    $1,%eax
53     ret
54     .align 4
55     SET_SIZE(isnanl)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/llrint.s

1

1262 Tue Nov 25 12:59:27 2014

new/usr/src/lib/libm/i386/src/llrint.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "llrint.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(llrint,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(llrint)
35     movl    %esp,%ecx
36     subl    $8,%esp
37     fldl   4(%ecx)           / load x
38     fistpll -8(%ecx)        / [x]
39     fwait
40     movl   -8(%ecx),%eax
41     movl  -4(%ecx),%edx
42     addl   $8,%esp
43     ret
44     .align 4
45     SET_SIZE(llrint)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/llrintf.s

1

1266 Tue Nov 25 12:59:28 2014

new/usr/src/lib/libm/i386/src/llrintf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "llrintf.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(llrintf,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(llrintf)
35     movl    %esp,%ecx
36     subl    $8,%esp
37     flds   4(%ecx)           / load x
38     fistpll -8(%ecx)        / [x]
39     fwait
40     movl    -8(%ecx),%eax
41     movl    -4(%ecx),%edx
42     addl   $8,%esp
43     ret
44     .align 4
45     SET_SIZE(llrintf)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/llrintl.s

1

1266 Tue Nov 25 12:59:28 2014

new/usr/src/lib/libm/i386/src/llrintl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "llrintl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(llrintl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(llrintl)
35     movl    %esp,%ecx
36     subl    $8,%esp
37     fldt   4(%ecx)           / load x
38     fistpll -8(%ecx)        / [x]
39     fwait
40     movl    -8(%ecx),%eax
41     movl    -4(%ecx),%edx
42     addl   $8,%esp
43     ret
44     .align 4
45     SET_SIZE(llrintl)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/log.s

1

```
*****
2282 Tue Nov 25 12:59:28 2014
new/usr/src/lib/libm/i386/src/log.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "log.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(log)
36     fldln2                / loge(2)
37     movl    8(%esp),%eax  / eax <-- hi_32(x)
38     testl  $0x80000000,%eax
39     jnz    .maybe_0_or_less
40     testl  $0x7fffffff,%eax
41     jz     .maybe_0
42     fldl  4(%esp)        / arg, loge(2)
43     fyl2x                / loge(2)*log2(arg); ln(arg)
44     ret

46 .maybe_0:
47     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
48     cmpl  $0,%ecx
49     je    .zero         / no branch if x is +denormal
50 .neg_nan_reentry:
51     fldl  4(%esp)        / arg, loge(2)
52     fyl2x                / loge(2)*log2(arg); ln(arg)
53     ret

55 .zero_or_less:
56     / x =< 0
57     testl  $0x7fffffff,%eax
```

new/usr/src/lib/libm/i386/src/log.s

2

```
58     jnz    .less_than_0
59     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
60     cmpl  $0,%ecx
61     jne    .less_than_0  / branch if x is -denormal
62 .zero:
63     / x = +/-0
64     pushl %ebp
65     movl  %esp,%ebp
66     PIC_SETUP(1)
67     pushl $16
68     jmp   .merge
69
70 .maybe_0_or_less:
71     cmpl  $0xffff0000,%eax / -INF below hi_32(x)?
72     ja    .neg_nan_reentry
73     jb    .zero_or_less
74     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
75     cmpl  $0,%ecx
76     jne    .neg_nan_reentry / branch if x is NaN with signbit = 1
77     / x = -INF
78 .less_than_0:
79     pushl %ebp
80     movl  %esp,%ebp
81     PIC_SETUP(2)
82     pushl $17
83 .merge:
84     fstp  %st(0)         / stack empty
85     pushl 12(%ebp)
86     pushl 8(%ebp)
87     pushl 12(%ebp)
88     pushl 8(%ebp)
89     call  PIC_F(_SVID_libm_err)
90     addl  $20,%esp
91     PIC_WRAPUP
92     leave
93     ret
94     .align 4
95     SET_SIZE(log)

_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/log10.s

1

```
*****
2300 Tue Nov 25 12:59:29 2014
new/usr/src/lib/libm/i386/src/log10.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "log10.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log10,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(log10)
36     fldlg2                / log10(2)
37     movl    8(%esp),%eax   / eax <-- hi_32(x)
38     testl  $0x80000000,%eax
39     jnz    .maybe_0_or_less
40     testl  $0x7fffffff,%eax
41     jz     .maybe_0
42     fldl  4(%esp)         / arg, log10(2)
43     fyl2x                / log10(2)*log2(arg); log10(arg)
44     ret

46 .maybe_0:
47     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
48     cmpl  $0,%ecx
49     je    .zero         / no branch if x is +denormal
50 .neg_nan_reentry:
51     fldl  4(%esp)         / arg, log10(2)
52     fyl2x                / log10(2)*log2(arg); log10(arg)
53     ret

55 .zero_or_less:
56     / x =< 0
57     testl  $0x7fffffff,%eax
```

new/usr/src/lib/libm/i386/src/log10.s

2

```
58     jnz    .less_than_0
59     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
60     cmpl  $0,%ecx
61     jne    .less_than_0  / branch if x is -denormal
62 .zero:
63     / x = +/-0
64     pushl %ebp
65     movl  %esp,%ebp
66     PIC_SETUP(1)
67     pushl $18
68     jmp   .merge

70 .maybe_0_or_less:
71     cmpl  $0xffff0000,%eax / -INF below hi_32(x)?
72     ja    .neg_nan_reentry
73     jb    .zero_or_less
74     movl  4(%esp),%ecx    / ecx <-- lo_32(x)
75     cmpl  $0,%ecx
76     jne    .neg_nan_reentry / branch if x is NaN with signbit = 1
77     / x = -INF
78 .less_than_0:
79     pushl %ebp
80     movl  %esp,%ebp
81     PIC_SETUP(2)
82     pushl $19
83 .merge:
84     fstp  %st(0)         / stack empty
85     pushl 12(%ebp)
86     pushl 8(%ebp)
87     pushl 12(%ebp)
88     pushl 8(%ebp)
89     call  PIC_F(_SVID_libm_err)
90     addl  $20,%esp
91     PIC_WRAPUP
92     leave
93     ret
94     .align 4
95     SET_SIZE(log10)

_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/log10f.s

1

1256 Tue Nov 25 12:59:29 2014

new/usr/src/lib/libm/i386/src/log10f.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "log10f.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log10f,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(log10f)
36     fldlg2
37     flds     4(%esp)           / st = arg, st(1) = log10(2)
38     fyl2x           / st = log10(arg) = log10(2)*log2(arg)
39     ret
40     .align 4
41     SET_SIZE(log10f)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/log101.s

1

1231 Tue Nov 25 12:59:30 2014

new/usr/src/lib/libm/i386/src/log101.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file "log101.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log101,function)
33 #include "libm_synonyms.h"

34     ENTRY(log101)
35     fldlg2
36     fldt    4(%esp)          / st = arg, st(1) = log10(2)
37     fyl2x          / st = log10(arg) = log10(2)*log2(arg)
38     ret
39     .align 4
40     SET_SIZE(log101)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/log2.s

1

```
*****
1186 Tue Nov 25 12:59:30 2014
new/usr/src/lib/libm/i386/src/log2.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "log2.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log2,function)
33 #include "libm_synonyms.h"

34     ENTRY(log2)
35     fldl                / push 1.0
36     fldl     4(%esp)    / push x
37     fyl2x               / st = 1.0*log2(arg)
38     ret
39     .align 4
40     SET_SIZE(log2)

_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/log2f.s

1

1190 Tue Nov 25 12:59:31 2014

new/usr/src/lib/libm/i386/src/log2f.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "log2f.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(log2f,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(log2f)
```

```
35     fldl                / push 1.0
```

```
36     flds      4(%esp)    / push x
```

```
37     fyl2x              / st = 1.0*log2(arg)
```

```
38     ret
```

```
39     .align 4
```

```
40     SET_SIZE(log2f)
```

```
_____unchanged_portion_omitted_____
```


new/usr/src/lib/libm/i386/src/log2l.s

1

1190 Tue Nov 25 12:59:31 2014

new/usr/src/lib/libm/i386/src/log2l.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "log2l.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(log2l,function)
33 #include "libm_synonyms.h"

34     ENTRY(log2l)
35     fldl                    / push 1.0
36     fldt     4(%esp)        / push x
37     fyl2x                   / st = 1.0*log2(arg)
38     ret
39     .align 4
40     SET_SIZE(log2l)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/logl.s

1

1218 Tue Nov 25 12:59:32 2014

new/usr/src/lib/libm/i386/src/logl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "logl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(logl,function)
33 #include "libm_synonyms.h"

34     ENTRY(logl)
35     fldln2
36     fldt    4(%esp)          / st = arg, st(1) = loge(2)
37     fyl2x          / st = ln(arg) = loge(2)*log2(arg)
38     ret
39     .align 4
40     SET_SIZE(logl)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/lrint.s

1

1237 Tue Nov 25 12:59:32 2014

new/usr/src/lib/libm/i386/src/lrint.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "lrint.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(lrint,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(lrint)
35     movl   %esp,%ecx
36     subl   $8,%esp
37     fldl   4(%ecx)           / load x
38     fistpl -8(%ecx)         / [x]
39     fwait
40     movl   -8(%ecx),%eax
41     addl   $8,%esp
42     ret
43     .align 4
44     SET_SIZE(lrint)
```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/lrintf.s

1

1241 Tue Nov 25 12:59:33 2014

new/usr/src/lib/libm/i386/src/lrintf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "lrintf.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(lrintf,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(lrintf)
35     movl   %esp,%ecx
36     subl   $8,%esp
37     flds   4(%ecx)           / load x
38     fistpl -8(%ecx)         / [x]
39     fwait
40     movl   -8(%ecx),%eax
41     addl   $8,%esp
42     ret
43     .align 4
44     SET_SIZE(lrintf)
```

```
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/lrintl.s

1

1241 Tue Nov 25 12:59:33 2014

new/usr/src/lib/libm/i386/src/lrintl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "lrintl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(lrintl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(lrintl)
35     movl    %esp,%ecx
36     subl    $8,%esp
37     fldt   4(%ecx)           / load x
38     fistpl -8(%ecx)         / [x]
39     fwait
40     movl   -8(%ecx),%eax
41     addl   $8,%esp
42     ret
43     .align 4
44     SET_SIZE(lrintl)
```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/lround.s

1

2158 Tue Nov 25 12:59:34 2014

new/usr/src/lib/libm/i386/src/lround.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file    "lround.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(lround,function)
33 #include "libm_synonyms.h"
34 #undef fabs
```

```
34     .section .rodata
35     .align 4
36 .Lhalf: .float 0.5
```

```
38     ENTRY(lround)
39     movl    %esp,%ecx
40     subl    $8,%esp
41     fstcw  -8(%ecx)
42     fldl   4(%ecx)
43     movw   -8(%ecx),%dx
44     andw  $0xf3ff,%dx
45     movw   %dx,-4(%ecx)
46     fldcw -4(%ecx)          / set RD = to_nearest
47     fld   %st(0)
48     frndint                / [x],x
49     fstcw -4(%ecx)
50     movw   -4(%ecx),%dx
51     andw  $0xf3ff,%dx
52     movw   -8(%ecx),%ax
53     andw  $0x0c00,%ax
54     orw   %dx,%ax
55     movw   %ax,-8(%ecx)
56     fldcw -8(%ecx)          / restore RD
```

new/usr/src/lib/libm/i386/src/lround.s

2

```
57     fucom                                / check if x is already an integer
58     fstsw  %ax
59     sahf
60     jp     0f
61     je     0f
62     fxch
63     fsub   %st(1),%st                    / x,[x]
64     fabs                                / x-[x],[x]
65     PIC_SETUP(1)
66     fcoms PIC_L(.Lhalf)
67     PIC_WRAPUP
68     fnstsw %ax
69     sahf
70     jae   2f                            / if |x-[x]| = 0.5 goto halfway,
71                                           / most cases will not take branch.
72 0:
73     fstp   %st(0)
74 1:
75     fistpl -8(%ecx)
76     fwait
77     movl   -8(%ecx),%eax
78     addl   $8,%esp
79     ret
80 2:
81     / x = n+0.5, recompute lround(x) as x+sign(x)*0.5
82     fldl   4(%ecx)                    / x, 0.5, [x]
83     movl   8(%ecx),%eax              / high part of x
84     andl   $0x80000000,%eax
85     jnz   3f
86     faddp
87     fstp   %st(1)
88     jmp    1b
89 3:
90     / here, x is negative, so return x-0.5
91     fsubp  %st,%st(1)                / x-0.5,[x]
92     fstp   %st(1)
93     jmp    1b
94     .align 4
95     SET_SIZE(lround)
```

_____ unchanged portion omitted _____

new/usr/src/lib/libm/i386/src/lroundl.s

1

```
*****
2182 Tue Nov 25 12:59:34 2014
new/usr/src/lib/libm/i386/src/lroundl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "lroundl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(lroundl,function)
33 #include "libm_synonyms.h"
34 #undef fabs

34     .section .rodata
35     .align 4
36 .Lhalf: .float 0.5

38     ENTRY(lroundl)
39     movl    %esp,%ecx
40     subl   $8,%esp
41     fstcw  -8(%ecx)
42     fldt   4(%ecx)
43     movw   -8(%ecx),%dx
44     andw   $0xf3ff,%dx
45     movw   %dx,-4(%ecx)
46     fldcw  -4(%ecx)          / set RD = to_nearest
47     fld   %st(0)
48     frndint          / [x],x
49     fstcw  -4(%ecx)
50     movw   -4(%ecx),%dx
51     andw   $0xf3ff,%dx
52     movw   -8(%ecx),%ax
53     andw   $0x0c00,%ax
54     orw   %dx,%ax
55     movw   %ax,-8(%ecx)
56     fldcw  -8(%ecx)          / restore RD
```

new/usr/src/lib/libm/i386/src/lroundl.s

2

```
57     fucom          / check if x is already an integer
58     fstsw   %ax
59     sahf
60     jp      0f
61     je      0f
62     fxch
63     fsub   %st(1),%st          / x,[x]
64     fabs          / x-[x],[x]
65     PIC_SETUP(1)
66     fcoms PIC_L(.Lhalf)
67     PIC_WRAPUP
68     fnstsw %ax
69     sahf
70     jae    2f          / if |x-[x]| = 0.5 goto halfway,
71                                / most cases will not take branch.
72 0:
73     fstp   %st(0)
74 1:
75     fistpl -8(%ecx)
76     fwait
77     movl   -8(%ecx),%eax
78     addl   $8,%esp
79     ret
80 2:
81     / x = n+0.5, recompute lroundl(x) as x+sign(x)*0.5
82     fldt   4(%ecx)          / x, 0.5, [x]
83     movw   12(%ecx),%ax     / sign+exp part of x
84     andw   $0x8000,%ax     / look at sign bit
85     jnz   3f
86     faddp
87     fstp   %st(1)
88     jmp    1b
89 3:
90     / here, x is negative, so return x-0.5
91     fsubp  %st,%st(1)      / x-0.5,[x]
92     fstp   %st(1)
93     jmp    1b
94     .align 4
95     SET_SIZE(lroundl)
_____unchanged_portion_omitted_____
```

```

*****
2706 Tue Nov 25 12:59:35 2014
new/usr/src/lib/libm/i386/src/nextafter.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "nextafter.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(nextafter,function)
33     .weak _nextafter
34     .type _nextafter,@function
35 _nextafter    = __nextafter
36 #include "libm_synonyms.h"
37 #include "libm_protos.h"

38     .data
39     .align 8
40 Fmin:    .long 0x1,0x0
41 ftmp:    .long 0,0           /// WILL WRITE INTO

44     ENTRY(nextafter)
45     pushl %ebp
46     movl %esp,%ebp
47     fldl 16(%ebp)           / y
48     subl $8,%esp
49     fldl 8(%ebp)           / load x
50     fucom                / x : y
51     fstsw %ax
52     sahf
53     jp    .NaN
54     je    .equal
55     fstp %st(1)           / x
56     ja    .bigger
57     / x < y

```

```

58     ftst
59     movl $1,%ecx           /// Fmin
60     movl %ecx,-8(%ebp)
61     movl $0,%ecx           /// Fmin+4
62     movl %ecx,-4(%ebp)
63     fnstsw %ax
64     sahf
65     je    .final
66     ja    .addulp
67     jb    .subulp
68 .bigger:
69     / x > y
70     ftst
71     movl $1,%ecx           /// Fmin
72     movl %ecx,-8(%ebp)
73     movl $0,%ecx           /// Fmin+4
74     xorl $0x80000000,%ecx
75     movl %ecx,-4(%ebp)
76     fnstsw %ax
77     sahf
78     je    .final
79     jb    .addulp
80 .subulp:
81     movl 8(%ebp),%eax      / low x
82     movl 12(%ebp),%ecx     / high x
83     subl $1,%eax          / low x - ulp
84     movl %eax,-8(%ebp)
85     sbbl $0x0,%ecx
86     movl %ecx,-4(%ebp)
87     jmp   .final
88 .addulp:
89     movl 8(%ebp),%eax      / low x
90     movl 12(%ebp),%ecx     / high x
91     addl $1,%eax          / low x + ulp
92     movl %eax,-8(%ebp)
93     adcl $0x0,%ecx
94     movl %ecx,-4(%ebp)

96 .final:
97     fstp %st(0)
98     fldl -8(%ebp)
99     andl $0x7ff00000,%ecx
100    jz    .underflow
101    cmpl $0x7ff00000,%ecx
102    je    .overflow
103    jmp   .return
104 .overflow:
105    PIC_SETUP(1)
106    pushl $46
107    fstp %st(0)           / stack empty
108    pushl -4(%ebp)
109    pushl -8(%ebp)
110    pushl -4(%ebp)
111    pushl -8(%ebp)
112    call PIC_F(_SVID_libm_err)
113    addl $20,%esp
114    PIC_WRAPUP
115    jmp   .return
116 .underflow:
117    PIC_SETUP(2)
118    fldl PIC_L(Fmin)
119    fmul %st(0),%st
120    fstpl PIC_L(ftmp)     / create underflow signal
121    PIC_WRAPUP
122    jmp   .return
123 .equal:

```


new/usr/src/lib/libm/i386/src/nextafter.s

3

```
124      fstp   %st(0)      / C99 says to return y when x == y
125      jmp    .return
126 .NaN:
127      faddp  %st,%st(1)  / x+y,x
128 .return:
129      fwait
130      leave
131      ret
132      .align 4
133      SET_SIZE(nextafter)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/nextafterf.s

1

```
*****
2460 Tue Nov 25 12:59:35 2014
new/usr/src/lib/libm/i386/src/nextafterf.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "nextafterf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(nextafterf,function)
33 #include "libm_synonyms.h"

34     .data
35     .align 4
36 Fmaxf: .long 0x7f7fffff
37 Fminf: .long 0x1
38 ftmpf: .long 0

41     ENTRY(nextafterf)
42     pushl   %ebp
43     movl   %esp,%ebp
44     movl   $0,%eax        /// upper half of %eax must be initialized
45     flds  12(%ebp)        / y
46     subl   $4,%esp
47     flds  8(%ebp)         / x, y
48     fucom %eax            / x : y
49     fstsw %ax
50     sahf
51     jp     .NaN
52     je     .equal
53     fstp  %st(1)         / x
54     ja     .bigger
55     / x < y
56     ftst
57     movl   $0x1,-4(%ebp) / -4(%ebp) contains Fminf
```

new/usr/src/lib/libm/i386/src/nextafterf.s

2

```
58     fnstsw %ax
59     sahf
60     je     .final
61     ja     .addulp
62     jb     .subulp
63 .bigger:
64     / x > y
65     ftst
66     movl   $0x80000001,-4(%ebp) / -4(%ebp) contains -Fminf
67     fnstsw %ax
68     sahf
69     je     .final
70     jb     .addulp
71 .subulp:
72     movl   8(%ebp),%eax / x
73     subl   $1,%eax / x - ulp
74     movl   %eax,-4(%ebp)
75     jmp    .final
76 .addulp:
77     movl   8(%ebp),%eax / x
78     addl   $1,%eax / x + ulp
79     movl   %eax,-4(%ebp)

81 .final:
82     fstp  %st(0) / empty
83     flds  -4(%ebp) / z
84     andl  $0x7f800000,%eax
85     jz    .underflow
86     cmpl  $0x7f800000,%eax
87     je    .overflow
88     jmp   .return
89 .overflow:
90     PIC_SETUP(1)
91     flds  PIC_L(Fmaxf) / Fmaxf, z
92     fmul  %st(0),%st / overflow-to-Inf, z
93     fstps PIC_L(ftmpf) / z & create overflow signal
94     PIC_WRAPUP
95     jmp   .return
96 .underflow:
97     PIC_SETUP(2)
98     flds  PIC_L(Fminf) / Fminf, z
99     fmul  %st(0),%st / underflow-to-0, z
100    fstps PIC_L(ftmpf) / z & create underflow signal
101    PIC_WRAPUP
102    jmp   .return
103 .equal:
104    fstp  %st(0) / C99 says to return y when x == y
105    jmp   .return
106 .NaN:
107    faddp %st,%st(1) / x+y
108 .return:
109    fwait
110    leave
111    ret
112    .align 4
113    SET_SIZE(nextafterf)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/nextafter1.s

1

4201 Tue Nov 25 12:59:35 2014

new/usr/src/lib/libm/i386/src/nextafter1.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "nextafter1.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(nextafter1,function)
33 #include "libm_synonyms.h"
```

```
34 .section .rodata
35 .align 4
36 .LFmaxl: .long 0xffffffff,0xffffffff,0x00007ffe
37 .LFminl: .long 0x1,0x0,0x0
```

```
40 ENTRY(nextafter1)
41 pushl %ebp
42 movl %esp,%ebp
43 fldt 20(%ebp) / y
44 subl $12,%esp
45 fldt 8(%ebp) / load x
46 fucom %eax / x : y
47 fstsw %ax
48 sahf
49 jp .LNaN
50 je .Lequal
51 fstp %st(1) / x
52 ja .Lbigger
53 / x < y
54 ftst
55 movl $1,-12(%ebp) /// -12(%ebp) contains Fminl
56 movl $0,-8(%ebp)
57 movl $0,%ecx /// final needs this
```

new/usr/src/lib/libm/i386/src/nextafter1.s

2

```
58 movl %ecx,-4(%ebp)
59 fnstsw %ax
60 sahf
61 je .Lfinal
62 ja .Laddulp
63 jb .Lsubulp
64 .Lbigger:
65 / x > y
66 ftst
67 movl $1,-12(%ebp) /// -12(%ebp) contains -Fminl
68 movl $0,-8(%ebp)
69 movl $0x00008000,%ecx /// final needs this
70 movl %ecx,-4(%ebp)
71 fnstsw %ax
72 sahf
73 je .Lfinal
74 jb .Laddulp
75 .Lsubulp:
76 movl 12(%ebp),%edx / high word of significand of x
77 movl 16(%ebp),%ecx / x's exponent
78 andl $0x0000ffff,%ecx
79 movl %edx,%eax
80 not %eax
81 andl $0x80000000,%eax / look at explicit leading bit
82 orl %ecx,%eax
83 andl $0x80007fff,%eax
84 jnz .Lnot_pseudonormal / zero value implies pseudonormal
85 addl $1,%ecx / if pseudonormal, turn into equivalent normal
86 .Lnot_pseudonormal:
87 movl 8(%ebp),%eax / low x
88 subl $1,%eax / low x - ulp
89 movl %eax,-12(%ebp)
90 cmpl $0xffffffff,%eax / this means low x was 0
91 jz .Lborrow
92 movl %edx,-8(%ebp)
93 movl %ecx,-4(%ebp)
94 jmp .Lfinal
95 .Lborrow:
96 cmpl $0x80000000,%edx / look at high x
97 je .Lsecond_borrow
98 subl $1,%edx
99 movl %edx,-8(%ebp)
100 movl %ecx,-4(%ebp)
101 jmp .Lfinal
102 .Lsecond_borrow:
103 movl %ecx,%eax
104 andl $0x7fff,%eax / look at exp x without sign bit
105 cmpl $1,%eax
106 jbe .Lsubnormal_result / exp > 1 ==> result will be normal
107 movl $0xffffffff,-8(%ebp)
108 subl $1,%ecx
109 movl %ecx,-4(%ebp)
110 jmp .Lfinal
111 .Lsubnormal_result:
112 movl $0x7fffffff,-8(%ebp)
113 movl %ecx,%eax
114 andl $0x8000,%eax / look at sign bit
115 jz .Lpositive
116 movl $0x8000,%ecx
117 movl %ecx,-4(%ebp)
118 jmp .Lfinal
119 .Lpositive:
120 movl $0,%ecx
121 movl %ecx,-4(%ebp)
122 jmp .Lfinal
123 .Laddulp:
```

```

124     movl    12(%ebp),%edx    / high x
125     movl    16(%ebp),%ecx    / x's exponent
126     andl    $0x0000ffff,%ecx
127     movl    %edx,%eax
128     not     %eax
129     andl    $0x80000000,%eax    / look at explicit leading bit
130     orl     %ecx,%eax
131     andl    $0x80007fff,%eax
132     jnz     .Lnot_pseudonormal_2    / zero value implies pseudonormal
133     addl    $1,%ecx
134 .Lnot_pseudonormal_2:
135     movl    8(%ebp),%eax    / low x
136     addl    $1,%eax    / low x + ulp
137     movl    %eax,-12(%ebp)
138     jz     .Lcarry    / jump if the content of %eax is 0
139     movl    %edx,-8(%ebp)
140     movl    %ecx,-4(%ebp)
141     jmp     .Lfinal
142 .Lcarry:
143     movl    %edx,%eax
144     andl    $0x7fffffff,%eax
145     cmpl    $0x7fffffff,%eax    / look at high x
146     je     .Lsecond_carry
147     addl    $1,%edx
148     movl    %edx,-8(%ebp)
149     movl    %ecx,-4(%ebp)
150     jmp     .Lfinal
151 .Lsecond_carry:
152     movl    $0x80000000,-8(%ebp)
153     addl    $1,%ecx
154     movl    %ecx,-4(%ebp)
155 .Lfinal:
156     fstp    %st(0)
157     fldt    -12(%ebp)
158     andl    $0x00007fff,%ecx
159     jz     .Lunderflow
160     cmpw    $0x7fff,%cx
161     je     .Loverflow
162     jmp     .Lreturn
163 .Loverflow:
164     PIC_SETUP(1)
165     fldt    PIC_L(.LFmaxl)
166     PIC_WRAPUP
167     fmulp   %st,%st(0)    / create overflow signal
168     jmp     .Lreturn
169 .Lunderflow:
170     PIC_SETUP(2)
171     fldt    PIC_L(.LFminl)
172     PIC_WRAPUP
173     fmulp   %st,%st(0)    / create underflow signal
174     jmp     .Lreturn
175 .Lequal:
176     fstp    %st(0)    / C99 says to return y when x == y
177     jmp     .Lreturn
178 .LNaN:
179     faddp   %st,%st(1)    / x+y,x
180 .Lreturn:
181     fwait
182     leave
183     ret
184     .align 4
185     SET_SIZE(nextafter1)

```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/nexttowardl.s

1

4205 Tue Nov 25 12:59:36 2014

new/usr/src/lib/libm/i386/src/nexttowardl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "nexttowardl.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(nexttowardl,function)
33 #include "libm_synonyms.h"
```

```
34     .section .rodata
35     .align 4
36 .LFmaxl:    .long  0xffffffff,0xffffffff,0x00007ffe
37 .LFminl:    .long  0x1,0x0,0x0
```

```
40     ENTRY(nexttowardl)
41     pushl   %ebp
42     movl   %esp,%ebp
43     fldt  20(%ebp)    / y
44     subl  $12,%esp
45     fldt  8(%ebp)     / load x
46     fucom  %eax       / x : y
47     fstsw  %ax
48     sahf
49     jp     .LNaN
50     je     .Lequal
51     fstp  %st(1)     / x
52     ja     .Lbigger
53     / x < y
54     ftst
55     movl  $1,-12(%ebp)  /// -12(%ebp) contains Fminl
56     movl  $0,-8(%ebp)
57     movl  $0,%ecx      /// final needs this
```

new/usr/src/lib/libm/i386/src/nexttowardl.s

2

```
58     movl   %ecx,-4(%ebp)
59     fnstsw %ax
60     sahf
61     je     .Lfinal
62     ja     .Laddulp
63     jb     .Lsubulp
64 .Lbigger:
65     / x > y
66     ftst
67     movl  $1,-12(%ebp)  /// -12(%ebp) contains -Fminl
68     movl  $0,-8(%ebp)
69     movl  $0x00008000,%ecx  /// final needs this
70     movl  %ecx,-4(%ebp)
71     fnstsw %ax
72     sahf
73     je     .Lfinal
74     jb     .Laddulp
75 .Lsubulp:
76     movl  12(%ebp),%edx  / high word of significand of x
77     movl  16(%ebp),%ecx  / x's exponent
78     andl  $0x0000ffff,%ecx
79     movl  %edx,%eax
80     not  %eax
81     andl  $0x80000000,%eax  / look at explicit leading bit
82     orl  %ecx,%eax
83     andl  $0x80007fff,%eax
84     jnz  .Lnot_pseudonormal  / zero value implies pseudonormal
85     addl  $1,%ecx        / if pseudonormal, turn into equivalent normal
86 .Lnot_pseudonormal:
87     movl  8(%ebp),%eax  / low x
88     subl  $1,%eax      / low x - ulp
89     movl  %eax,-12(%ebp)
90     cmpl  $0xffffffff,%eax  / this means low x was 0
91     jz     .Lborrow
92     movl  %edx,-8(%ebp)
93     movl  %ecx,-4(%ebp)
94     jmp   .Lfinal
95 .Lborrow:
96     cmpl  $0x80000000,%edx  / look at high x
97     je     .Lsecond_borrow
98     subl  $1,%edx
99     movl  %edx,-8(%ebp)
100    movl  %ecx,-4(%ebp)
101    jmp   .Lfinal
102 .Lsecond_borrow:
103    movl  %ecx,%eax
104    andl  $0x7fff,%eax  / look at exp x without sign bit
105    cmpl  $1,%eax
106    jbe  .Lsubnormal_result  / exp > 1 ==> result will be normal
107    movl  $0xffffffff,-8(%ebp)
108    subl  $1,%ecx
109    movl  %ecx,-4(%ebp)
110    jmp   .Lfinal
111 .Lsubnormal_result:
112    movl  $0x7fffffff,-8(%ebp)
113    movl  %ecx,%eax
114    andl  $0x8000,%eax  / look at sign bit
115    jz     .Lpositive
116    movl  $0x8000,%ecx
117    movl  %ecx,-4(%ebp)
118    jmp   .Lfinal
119 .Lpositive:
120    movl  $0,%ecx
121    movl  %ecx,-4(%ebp)
122    jmp   .Lfinal
123 .Laddulp:
```

```

124     movl    12(%ebp),%edx    / high x
125     movl    16(%ebp),%ecx    / x's exponent
126     andl    $0x0000ffff,%ecx
127     movl    %edx,%eax
128     not     %eax
129     andl    $0x80000000,%eax    / look at explicit leading bit
130     orl     %ecx,%eax
131     andl    $0x80007fff,%eax
132     jnz     .Lnot_pseudonormal_2    / zero value implies pseudonormal
133     addl    $1,%ecx
134 .Lnot_pseudonormal_2:
135     movl    8(%ebp),%eax    / low x
136     addl    $1,%eax    / low x + ulp
137     movl    %eax,-12(%ebp)
138     jz     .Lcarry    / jump if the content of %eax is 0
139     movl    %edx,-8(%ebp)
140     movl    %ecx,-4(%ebp)
141     jmp     .Lfinal
142 .Lcarry:
143     movl    %edx,%eax
144     andl    $0x7fffffff,%eax
145     cmpl    $0x7fffffff,%eax    / look at high x
146     je     .Lsecond_carry
147     addl    $1,%edx
148     movl    %edx,-8(%ebp)
149     movl    %ecx,-4(%ebp)
150     jmp     .Lfinal
151 .Lsecond_carry:
152     movl    $0x80000000,-8(%ebp)
153     addl    $1,%ecx
154     movl    %ecx,-4(%ebp)
155 .Lfinal:
156     fstp    %st(0)
157     fldt    -12(%ebp)
158     andl    $0x00007fff,%ecx
159     jz     .Lunderflow
160     cmpw    $0x7fff,%cx
161     je     .Loverflow
162     jmp     .Lreturn
163 .Loverflow:
164     PIC_SETUP(1)
165     fldt    PIC_L(.LFmaxl)
166     PIC_WRAPUP
167     fmulp    %st,%st(0)    / create overflow signal
168     jmp     .Lreturn
169 .Lunderflow:
170     PIC_SETUP(2)
171     fldt    PIC_L(.LFminl)
172     PIC_WRAPUP
173     fmulp    %st,%st(0)    / create underflow signal
174     jmp     .Lreturn
175 .Lequal:
176     fstp    %st(0)    / C99 says to return y when x == y
177     jmp     .Lreturn
178 .LNaN:
179     faddp    %st,%st(1)    / x+y,x
180 .Lreturn:
181     fwait
182     leave
183     ret
184     .align 4
185     SET_SIZE(nexttowardl)

```

unchanged portion omitted

```

*****
11151 Tue Nov 25 12:59:36 2014
new/usr/src/lib/libm/i386/src/pow.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "pow.s"

31 / Note: 0^NaN should not signal "invalid" but this implementation
32 / does because y is placed on the NPX stack.

34 / Special cases:
35 /
36 / x ** 0 is 1
37 / 1 ** y is 1
38 / x ** NaN is NaN
39 / NaN ** y (except 0) is NaN
40 / x ** 1 is x
41 / +-(|x| > 1) ** +inf is +inf
42 / +-(|x| > 1) ** -inf is +0
43 / +-(|x| < 1) ** +inf is +0
44 / +-(|x| < 1) ** -inf is +inf
45 / (-1) ** +-inf is +1
46 / +0 ** +y (except 0, NaN) is +0
47 / -0 ** +y (except 0, NaN, odd int) is +0
48 / -0 ** +y (odd int) is -0
49 / +-0 ** -y (except 0, NaN) _SVID_libm_err
50 / +inf ** +y (except 0, NaN) is +inf
51 / +inf ** -y (except 0, NaN) is +0
52 / -inf ** +-y (except 0, NaN) is -0 ** -+y (NO z flag)
53 / x ** -1 is 1/x
54 / x ** 2 is x*x
55 / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
56 / x ** y (x negative & y not integer) _SVID_libm_err
57 / if x and y are finite and x**y = 0 _SVID_libm_err (underflow)
58 / if x and y are finite and x**y = inf _SVID_libm_err (overflow)

```

```

60 #include "libm.h"
61 LIBM_ANSI_PRAGMA_WEAK(pow,function)
62 #include "libm_synonyms.h"
63 #include "libm_protos.h"
64 #include "xpg6.h"

66 #undef fabs

65     .data
66     .align 4
67 negzero:
68     .float -0.0
69 one:
70     .float 1.0
71 negone:
72     .float -1.0
73 two:
74     .float 2.0
75 Snan:
76     .long 0x7f800001
77 pinfinity:
78     .long 0x7f800000
79 ninfinitiy:
80     .long 0xff800000

83     ENTRY(pow)
84     pushl %ebp
85     movl %esp,%ebp
86     PIC_SETUP(1)

88     fldl 8(%ebp) / x
89     fxam / determine class of x
90     fnstsw %ax / store status in %ax
91     movb %ah,%dh / %dh <- condition code of x

93     fldl 16(%ebp) / y , x
94     fxam / determine class of y
95     fnstsw %ax / store status in %ax
96     movb %ah,%dl / %dl <- condition code of y

98     call .pow_main /// LOCAL
99     PIC_WRAPUP
100    leave
101    ret

103 .pow_main:
104 / x ** 0 is 1 unless x is 0 or a NaN
105 movb %dl,%cl
106 andb $0x45,%cl
107 cmpb $0x40,%cl / C3=1 C2=0 C1=? C0=0 when +-0
108 jne 1f
109 movb %dh,%cl
110 andb $0x45,%cl
111 cmpb $0x40,%cl / C3=1 C2=0 C1=? C0=0 when +-0
112 jne 2f
113 / 0^0
114 pushl $20
115 jmp .SVIDerr / SVID error handler

116 2:
117 cmpb $0x01,%cl /// C3=0 C2=0 C1=? C0=1 when +-NaN
118 jne 2f
119 / NaN^0
120 pushl $42
121 jmp .SVIDerr

```

```

122 2:
123     / (not 0 or NaN)^0
124     fstp    %st(0)          / x
125     fstp    %st(0)          / stack empty
126     fldl    / 1
127     ret

129 1:
130     PIC_G_LOAD(movzwl, __xpg6, eax)
131     andl    $_C99SUSv3_pow_treats_Inf_as_an_even_int, %eax
132     cmpl    $0, %eax
133     je      lf

135     / C99: 1 ** anything is 1
136     fldl    / 1, y, x
137     fucomp  %st(2)          / y, x
138     fnstsw  %ax             / store status in %ax
139     sahf    / 80387 flags in %ax to 80386 flags
140     jp      lf              / so that pow(NaN1, NaN2) returns NaN2
141     jne     lf
142     fstp    %st(0)          / x
143     ret

145 1:
146     / x ** NaN is NaN
147     movb    %dl, %cl
148     andb    $0x45, %cl
149     cmpb    $0x01, %cl      / C3=0 C2=0 C1=? C0=1 when +-NaN
150     jne     lf
151     fstp    %st(1)          / y
152     ret

154 1:
155     / y is not NaN
156     / NaN ** y (except 0) is NaN
157     movb    %dh, %cl
158     andb    $0x45, %cl
159     cmpb    $0x01, %cl      / C3=0 C2=0 C1=? C0=1 when +-NaN
160     jne     lf
161     fstp    %st(0)          / x
162     ret

163 1:
164     / x is not NaN
165     / x ** 1 is x
166     fcoms   PIC_L(one)      / y, x
167     fnstsw  %ax             / store status in %ax
168     sahf    / 80387 flags in %ax to 80386 flags
169     jne     lf
170     fstp    %st(0)          / x
171     ret

172 1:
173     / y is not 1
174     / +-(x > 1) ** +inf is +inf
175     / +-(x > 1) ** -inf is +0
176     / +-(x < 1) ** +inf is +0
177     / +-(x < 1) ** -inf is +inf
178     / +-(x = 1) ** +-inf is NaN
179     movb    %dl, %cl
180     andb    $0x47, %cl
181     cmpb    $0x05, %cl      / C3=0 C2=1 C1=0 C0=1 when +inf
182     je      .yispinf
183     cmpb    $0x07, %cl      / C3=0 C2=1 C1=1 C0=1 when -inf
184     je      .yisninf

185     / +0 ** +y (except 0, NaN)          is +0
186     / -0 ** +y (except 0, NaN, odd int) is +0
187     / +0 ** -y (except 0, NaN)          is +inf (z flag)

```

```

188     / -0 ** -y (except 0, NaN, odd int) is +inf (z flag)
189     / -0 ** y (odd int)                 is - (+0 ** x)
190     movb    %dh, %cl
191     andb    $0x47, %cl
192     cmpb    $0x40, %cl      / C3=1 C2=0 C1=0 C0=0 when +0
193     je      .xispzero
194     cmpb    $0x42, %cl      / C3=1 C2=0 C1=1 C0=0 when -0
195     je      .xisnzero

197     / +inf ** +y (except 0, NaN)        is +inf
198     / +inf ** -y (except 0, NaN)        is +0
199     / -inf ** +-y (except 0, NaN)       is -0 ** +-y (NO z flag)
200     movb    %dh, %cl
201     andb    $0x47, %cl
202     cmpb    $0x05, %cl      / C3=0 C2=1 C1=0 C0=1 when +inf
203     je      .xispinf
204     cmpb    $0x07, %cl      / C3=0 C2=1 C1=1 C0=1 when -inf
205     je      .xisninf

207     / x ** -1 is 1/x
208     fcoms   PIC_L(negone)   / y, x
209     fnstsw  %ax             / store status in %ax
210     sahf    / 80387 flags in %ax to 80386 flags
211     jne     lf
212     fld     %st(1)          / x, y, x
213     fdivrs  PIC_L(one)      / 1/x, y, x
214     jmp     .signok         / check for over/underflow

216 1:
217     / y is not -1
218     / x ** 2 is x*x
219     fcoms   PIC_L(two)      / y, x
220     fnstsw  %ax             / store status in %ax
221     sahf    / 80387 flags in %ax to 80386 flags
222     jne     lf
223     fld     %st(1)          / x, y, x
224     fld     %st(0)          / x, x, y, x
225     fmulp   / x^2, y, x
226     jmp     .signok         / check for over/underflow

227 1:
228     / y is not 2
229     / make copies of x & y
230     fld     %st(1)          / x, y, x
231     fld     %st(1)          / y, x, y, x

232     / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
233     / x ** y (x negative & y not integer) is NaN
234     movl    $0, %ecx        / track whether to flip sign of result
235     fld     %st(1)          / x, y, x, y, x
236     ftst    / compare %st(0) with 0
237     fnstsw  %ax             / store status in %ax
238     sahf    / 80387 flags in %ax to 80386 flags
239     fstp    %st(0)          / y, x, y, x
240     ja      .merge         / x > 0
241     / x < 0
242     call    .yis_int
243     cmpl    $0, %ecx
244     jne     lf
245     / x < 0, y is non-integral
246     fstp    %st(0)          / x, y, x
247     fstp    %st(0)          / y, x
248     pushl   $24
249     jmp     .SVIDerr        / SVID error handler

251 1:
252     / x < 0 & y = int
253     fxch    / x, y, y, x
254     fchs    / px = -x, y, y, x

```



```

254      fxch                                / y , px , y , x
255 .merge:
256      / px > 0
257      fxch                                / px , y , y , x

259      / x**y = exp(y*ln(x))
260      fyl2x                                / t=y*log2(px) , y , x
261      fld      %st(0)                      / t , t , y , x
262      frndint                                / [t] , t , y , x
263      fxch                                    / t , [t] , y , x
264      fucom
265      fnstsw  %ax                          / store status in %ax
266      sahf                                    / 80387 flags in %ax to 80386 flags
267      je      1f                            / t is integral
268      fsub   %st(1),%st                    / t-[t] , [t] , y , x
269      f2xmul                                / 2*(t-[t])-1 , [t] , y , x
270      fadds  PIC_L(one)                    / 2*(t-[t]) , [t] , y , x
271      fscale                                / 2**t = px**y , [t] , y , x
272      jmp
273 1:
274      fstp   %st(0)                          / t=[t] , y , x
275      fldl   / 1 , t , y , x
276      fscale                                / 1*2**t = x**y , t , y , x
277 2:
278      fstp   %st(1)                          / x**y , y , x
279      cmpl  $1,%ecx
280      jne   .signok
281      fchs
282 .signok:
283      subl  $8,%esp
284      fstpl (%esp)                          / round to double precision
285      fldl  (%esp)                          / place result on NPX stack
286      addl  $8,%esp

288      fxam                                    / determine class of x**y
289      fnstsw %ax                             / store status in %ax
290      andw  $0x4500,%ax
291      / check for overflow
292      cmpw  $0x0500,%ax                    / C0=0 C1=1 C2=? C3=1 then +-inf
293      jne   1f
294      / x*y overflows
295      fstp  %st(0)                          / y , x
296      pushl $21
297      jmp  .SVIDerr
298 1:
299      / check for underflow
300      cmpw  $0x4000,%ax                    / C0=1 C1=0 C2=? C3=0 then +-0
301      jne   1f
302      / x*y underflows
303      fstp  %st(0)                          / y , x
304      pushl $22
305      jmp  .SVIDerr
306 1:
307      fstp  %st(2)                          / y , x**y
308      fstp  %st(0)                          / x**y
309      ret

```

311 / -----

```

313 .xispinf:
314      ftst                                / compare %st(0) with 0
315      fnstsw %ax                          / store status in %ax
316      sahf                                    / 80387 flags in %ax to 80386 flags
317      ja    .retpinf                       / y > 0
318      jmp   .retpzzero                     / y < 0

```

```

320 .xisninf:
321      / -inf ** +-y is -0 ** +-y
322      fchs                                / -y , x
323      flds  PIC_L(negzero)                 / -0 , -y , x
324      fstp  %st(2)                          / -y , -0
325      jmp   .xisnzzero

327 .yispinf:
328      fld   %st(1)                          / x , y , x
329      fabs                                / |x| , y , x
330      fcomps PIC_L(one)                     / y , x
331      fnstsw %ax                             / store status in %ax
332      sahf                                    / 80387 flags in %ax to 80386 flags
333      je    .retponeorinvalid               / x == -1      C99
334      ja    .retpinf                         / |x| > 1
335      jmp   .retpzzero                       / |x| < 1

337 .yisninf:
338      fld   %st(1)                          / x , y , x
339      fabs                                / |x| , y , x
340      fcomps PIC_L(one)                     / y , x
341      fnstsw %ax                             / store status in %ax
342      sahf                                    / 80387 flags in %ax to 80386 flags
343      je    .retponeorinvalid               / x == -1      C99
344      ja    .retpzzero                       / |x| > 1
345      jmp   .retpinf                         / |x| < 1

347 .xispzzero:
348      / y cannot be 0 or NaN ; stack has      y , x
349      ftst                                / compare %st(0) with 0
350      fnstsw %ax                             / store status in %ax
351      sahf                                    / 80387 flags in %ax to 80386 flags
352      ja    .retpzzero                       / y > 0
353      / x = +0 & y < 0
354      jmp   .SVIDzerotoneg

356 .xisnzzero:
357      / y cannot be 0 or NaN ; stack has      y , x
358      call  .y_is_int
359      cmpl  $1,%ecx
360      jne   1f                               / y is not an odd integer
361      / y is an odd integer
362      ftst                                / compare %st(0) with 0
363      fnstsw %ax                             / store status in %ax
364      sahf                                    / 80387 flags in %ax to 80386 flags
365      ja    .retnzzero                       / y > 0
366      / x = -0 & y < 0 (odd int)
367      / x = -inf & y != 0 or NaN
368      movb  %dh,%cl
369      andb  $0x45,%cl
370      cmpb  $0x05,%cl                       / C3=0 C2=1 C1=? C0=1 when +-inf
371      jne   .SVIDzerotoneg
372      fstp  %st(0)                          / x
373      fstp  %st(0)                          / stack empty
374      flds  PIC_L(ninfinity)                 / -inf
375      ret

377 1:
378      ftst                                / compare %st(0) with 0
379      fnstsw %ax                             / store status in %ax
380      sahf                                    / 80387 flags in %ax to 80386 flags
381      ja    .retpzzero                       / y > 0
382      / x = -0 & y < 0 (not odd int)
383      / x = -inf & y not 0 or NaN
384      movb  %dh,%cl
385      andb  $0x45,%cl

```

```

386      cmpb    $0x05,%c1          / C3=0 C2=1 C1=? C0=1 when +-inf
387      jne     .SVIDzerotoneg     /
388      jmp     .retpinf          / return +inf (NO z flag)

390 .retpzero:
391      fstp    %st(0)             / x
392      fstp    %st(0)             / stack empty
393      fldz    / +0
394      ret

396 .retnzero:
397      fstp    %st(0)             / x
398      fstp    %st(0)             / stack empty
399      flds    PIC_L(negzero)     / -0
400      ret

402 .retponeorinvalid:
403      PIC_G_LOAD(movzwl,__xpg6,eax)
404      andl    $_C99SUSv3_pow_treats_Inf_as_an_even_int,%eax
405      cmpl    $0,%eax
406      je     lf
407      fstp    %st(0)             / x
408      fstp    %st(0)             / stack empty
409      fldl    / 1
410      ret

412 1:
413      fstp    %st(0)             / x
414      fstp    %st(0)             / stack empty
415      flds    PIC_L(Snan)        / Q NaN (i flag)
416      fwait
417      ret

419 .retpinf:
420      fstp    %st(0)             / x
421      fstp    %st(0)             / stack empty
422      flds    PIC_L(pinfinity)   / +inf
423      ret

425 .SVIDzerotoneg:
426      pushl   $23
427 .SVIDerr:
428      / At this point the fp stack contains y , x and the number
429      / of the error case has been pushed on the memory stack.
430      subl    $16,%esp
431      fstpl   8(%esp)            / push y
432      fstpl   (%esp)            / push x; NPX stack empty
433      call   PIC_F(_SVID_libm_err) / report result/error according to SVID
434      addl   $20,%esp
435      ret

437 / Set %ecx to 2 if y is an even integer, 1 if y is an odd integer,
438 / 0 otherwise. Assume y is not zero. Do not raise inexact or modify
439 / %edx.
440 .y_is_int:
441      movl    20(%ebp),%eax
442      andl    $0x7fffffff,%eax   / |y|
443      cmpl    $0x43400000,%eax
444      jae    lf                  / |y| >= 2^53, an even int
445      cmpl    $0x3ff00000,%eax
446      jb     2f                  / |y| < 1, can't be an int
447      movl    %eax,%ecx
448      sarl    $20,%ecx
449      subl    $0x433,%ecx
450      negl    %ecx                / 52 - unbiased exponent of y
451      movl    16(%ebp),%eax

```

```

452      bsfl    %eax,%eax          / index of least sig. 1 bit
453      jne    3f                  / jump if 1 bit found
454      movl    20(%ebp),%eax
455      bsfl    %eax,%eax
456      addl    $32,%eax           / 32 + index of least sig. 1 bit
457 3:
458      cmpl    %ecx,%eax
459      jb     2f
460      ja     lf
461      movl    $1,%ecx
462      ret
463 1:
464      movl    $2,%ecx
465      ret
466 2:
467      xorl    %ecx,%ecx
468      ret
469      .align 4
470      SET_SIZE(pow)
unchanged portion omitted

```

new/usr/src/lib/libm/i386/src/powf.s

1

```
*****
10472 Tue Nov 25 12:59:37 2014
new/usr/src/lib/libm/i386/src/powf.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "powf.s"

31 / Note: 0^SNaN should not signal "invalid" but this implementation
32 / does because y is placed on the NPX stack.

34 / Special cases:
35 /
36 / x ** 0 is 1
37 / 1 ** y is 1                (C99)
38 / x ** NaN is NaN
39 / NaN ** y (except 0) is NaN
40 / x ** 1 is x
41 / +-(|x| > 1) ** +inf is +inf
42 / +-(|x| > 1) ** -inf is +0
43 / +-(|x| < 1) ** +inf is +0
44 / +-(|x| < 1) ** -inf is +inf
45 / (-1) ** +-inf is +1        (C99)
46 / +0 ** +y (except 0, NaN)    is +0
47 / -0 ** +y (except 0, NaN, odd int) is +0
48 / +0 ** -y (except 0, NaN)    is +inf (z flag)
49 / -0 ** -y (except 0, NaN, odd int) is +inf (z flag)
50 / -0 ** y (odd int)           is - (+0 ** x)
51 / +inf ** +y (except 0, NaN)   is +inf
52 / +inf ** -y (except 0, NaN)   is +0
53 / -inf ** +-y (except 0, NaN)  is -0 ** -y (NO z flag)
54 / x ** -1 is 1/x
55 / x ** 2 is x*x
56 / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
57 / x ** y (x negative & y not integer) is NaN (i flag)
```

new/usr/src/lib/libm/i386/src/powf.s

2

```
59 #include "libm.h"
60 LIBM_ANSI_PRAGMA_WEAK(powf,function)
61 #include "libm_synonyms.h"
62 #include "libm_protos.h"
63 #include "xpg6.h"

65 #undef fabs

64     .data
65     .align 4
66 negzero:
67     .float -0.0
68 half:
69     .float 0.5
70 one:
71     .float 1.0
72 negone:
73     .float -1.0
74 two:
75     .float 2.0
76 Snan:
77     .long 0x7f800001
78 pinfinity:
79     .long 0x7f800000
80 ninfinity:
81     .long 0xff800000

84     ENTRY(powf)
85     pushl %ebp
86     movl %esp,%ebp
87     PIC_SETUP(1)

89     flds 8(%ebp)                / x
90     fxam                        / determine class of x
91     fnstsw %ax                  / store status in %ax
92     movb %ah,%dh                / %dh <- condition code of x

94     flds 12(%ebp)               / y , x
95     fxam                        / determine class of y
96     fnstsw %ax                  / store status in %ax
97     movb %ah,%dl                / %dl <- condition code of y

99     call .pow_main                /// LOCAL
100    PIC_WRAPUP
101    leave
102    ret

104 .pow_main:
105     / x ** 0 is 1
106     movb %dl,%cl
107     andb $0x45,%cl
108     cmpb $0x40,%cl                / C3=1 C2=0 C1=? C0=0 when +-0
109     jne lf
110     fstp %st(0)                    / x
111     fstp %st(0)                    / stack empty
112     fldl %dl                        / 1
113     ret

115 1:
116     / y is not zero
117     PIC_G_LOAD(movzwl, __xpg6,eax)
118     andl $_C99SUSv3_pow_treats_Inf_as_an_even_int,%eax
119     cmpl $0,%eax
120     je lf

121     / C99: 1 ** anything is 1
```

```

122 fldl          / 1, y, x
123 fucomp      %st(2) / y, x
124 fnstsw     %ax      / store status in %ax
125 sahf       / 80387 flags in %ax to 80386 flags
126 jp        lf      / so that pow(NaN1,NaN2) returns NaN2
127 jne       lf
128 fstp     %st(0)   / x
129 ret

131 1:
132 / x ** NaN is NaN
133 movb     %dl,%cl
134 andb     $0x45,%cl
135 cmpb     $0x01,%cl / C3=0 C2=0 C1=? C0=1 when +-NaN
136 jne     lf
137 fstp     %st(1)   / y
138 ret

140 1:
141 / y is not NaN
142 / NaN ** y (except 0) is NaN
143 movb     %dh,%cl
144 andb     $0x45,%cl
145 cmpb     $0x01,%cl / C3=0 C2=0 C1=? C0=1 when +-NaN
146 jne     lf
147 fstp     %st(0)   / x
148 ret

149 1:
150 / x is not NaN
151 / x ** 1 is x
152 fcoms    PIC_L(one) / y, x
153 fnstsw   %ax      / store status in %ax
154 sahf     / 80387 flags in %ax to 80386 flags
155 jne     lf
156 fstp     %st(0)   / x
157 ret

158 1:
159 / y is not 1
160 / ++-(x > 1) ** +inf is +inf
161 / ++-(x > 1) ** -inf is +0
162 / ++-(x < 1) ** +inf is +0
163 / ++-(x < 1) ** -inf is +inf
164 / ++-(x = 1) ** +-inf is NaN
165 movb     %dl,%cl
166 andb     $0x47,%cl
167 cmpb     $0x05,%cl / C3=0 C2=1 C1=0 C0=1 when +inf
168 je      .yispinf
169 cmpb     $0x07,%cl / C3=0 C2=1 C1=1 C0=1 when -inf
170 je      .yisninf

171 / +0 ** +y (except 0, NaN) is +0
172 / -0 ** +y (except 0, NaN, odd int) is +0
173 / +0 ** -y (except 0, NaN) is +inf (z flag)
174 / -0 ** -y (except 0, NaN, odd int) is +inf (z flag)
175 / -0 ** y (odd int) is - (+0 ** x)
176 movb     %dh,%cl
177 andb     $0x47,%cl
178 cmpb     $0x40,%cl / C3=1 C2=0 C1=0 C0=0 when +0
179 je      .xispzero
180 cmpb     $0x42,%cl / C3=1 C2=0 C1=1 C0=0 when -0
181 je      .xisnzero

183 / +inf ** +y (except 0, NaN) is +inf
184 / +inf ** -y (except 0, NaN) is +0
185 / -inf ** +y (except 0, NaN) is -0 ** -y (NO z flag)
186 movb     %dh,%cl
187 andb     $0x47,%cl

```

```

188 cmpb     $0x05,%cl / C3=0 C2=1 C1=0 C0=1 when +inf
189 je      .xispinf
190 cmpb     $0x07,%cl / C3=0 C2=1 C1=1 C0=1 when -inf
191 je      .xisninf

193 / x ** -1 is 1/x
194 fcoms    PIC_L(negone) / y, x
195 fnstsw   %ax      / store status in %ax
196 sahf     / 80387 flags in %ax to 80386 flags
197 jne     lf
198 fld      %st(1)   / x, y, x
199 fdivrs   PIC_L(one) / 1/x, y, x
200 jmp      .signok / check for over/underflow

202 1:
203 / y is not -1
204 / x ** 2 is square(x)
205 fcoms    PIC_L(two) / y, x
206 fnstsw   %ax      / store status in %ax
207 sahf     / 80387 flags in %ax to 80386 flags
208 jne     lf
209 fld      %st(1)   / x, y, x
210 fld      %st(0)   / x, x, y, x
211 fmulp    / x^2, y, x
212 jmp      .signok / check for over/underflow

213 1:
214 / y is not 2
215 / x ** 1/2 is sqrt(x)
216 fcoms    PIC_L(half) / y, x
217 fnstsw   %ax      / store status in %ax
218 sahf     / 80387 flags in %ax to 80386 flags
219 jne     lf
220 fld      %st(1)   / x, y, x
221 fsqrt    / sqrt(x), y, x
222 jmp      .signok / check for over/underflow

223 1:
224 / y is not 2
225 / make copies of x & y
226 fld      %st(1)   / x, y, x
227 fld      %st(1)   / y, x, y, x

228 / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
229 / x ** y (x negative & y not integer) is NaN
230 movl     $0,%ecx / track whether to flip sign of result
231 fld      %st(1)   / x, y, x, y, x
232 ftst    / compare %st(0) with 0
233 fnstsw   %ax      / store status in %ax
234 sahf     / 80387 flags in %ax to 80386 flags
235 fstp     %st(0)   / y, x, y, x
236 ja      .merge   / x > 0
237 / x < 0
238 call    .y_is_int
239 cmpl    $0,%ecx
240 jne     lf
241 / x < 0 & y != int so x**y = NaN (i flag)
242 fstp     %st(0)   / x, y, x
243 fstp     %st(0)   / y, x
244 fstp     %st(0)   / y, x
245 fstp     %st(0)   / y, x
246 fldz
247 fdiv    %st,%st(0) / 0/0
248 ret

250 1:
251 / x < 0 & y = int
252 fxch
253 fxch / x, y, y, x
254 fxch / px = -x, y, y, x
255 fxch / y, px, y, x

```

```

254 .merge:
255     / px > 0
256     fxch                / px , y , y , x

258     / x**y = exp(y*ln(x))
259     fyl2x
260     fld    %st(0)
261     frndint
262     fxch                / t=y*log2(px) , y , x
263     fucom
264     fnstsw %ax
265     sahf                / t , t , y , x
266     je     1f
267     fsub  %st(1),%st
268     f2xm1
269     fadds PIC_L(one)
270     fscale
271     jmp   2f
272 1:
273     fstp  %st(0)
274     fldl
275     fscale                / [t] , t , y , x
276 2:
277     fstp  %st(1)
278     cmpl $1,%ecx
279     jne  .signok
280     fchs                / t , [t] , y , x
281 .signok:
282     subl $4,%esp
283     fstps (%esp)
284     flds (%esp)
285     addl $4,%esp
286     fstp %st(2)
287     fstp %st(0)
288     ret

290 / -----
292 .xispinf:
293     ftst                / compare %st(0) with 0
294     fnstsw %ax
295     sahf                / store status in %ax
296     ja     .retpinf
297     jmp   .retpzero

299 .xisninf:
300     / -inf ** +-y is -0 ** -+y
301     fchs                / 80387 flags in %ax to 80386 flags
302     flds PIC_L(negzero)
303     fstp %st(2)
304     jmp   .xisnzero

306 .yispinf:
307     fld    %st(1)
308     fabs
309     fcomps PIC_L(one)
310     fnstsw %ax
311     sahf                / store status in %ax
312     je     .retponeorinvalid
313     ja     .retpinf
314     jmp   .retpzero

316 .yisninf:
317     fld    %st(1)
318     fabs
319     fcomps PIC_L(one)

```

```

320     fnstsw %ax
321     sahf                / store status in %ax
322     je     .retponeorinvalid
323     ja     .retpzero
324     jmp   .retpinf

326 .xispzero:
327     / y cannot be 0 or NaN ; stack has y , x
328     ftst                / compare %st(0) with 0
329     fnstsw %ax
330     sahf                / store status in %ax
331     ja     .retpzero
332     / x = +0 & y < 0 so x**y = +inf
333     jmp   .retpinfzflag

335 .xisnzero:
336     / y cannot be 0 or NaN ; stack has y , x
337     call .y_is_int
338     cmpl $1,%ecx
339     jne  1f
340     / y is an odd integer
341     ftst                / y is not an odd integer
342     fnstsw %ax
343     sahf                / compare %st(0) with 0
344     ja     .retnzero
345     / store status in %ax
346     / x = -0 & y < 0 (odd int) return -inf (z flag)
347     / x = -inf & y != 0 or NaN return -inf (NO z flag)
348     movb %dh,%cl
349     andb $0x45,%cl
350     cmpb $0x05,%cl
351     je     2f
352 2:
353     fdiv %st,%st(1)
354     / C3=0 C2=1 C1=? C0=1 when +-inf
355     / y / x, x (raise z flag)
356     / stack empty
357     / -inf

358 1:
359     / y is not an odd integer
360     ftst                / compare %st(0) with 0
361     fnstsw %ax
362     sahf                / store status in %ax
363     ja     .retpzero
364     / 80387 flags in %ax to 80386 flags
365     / y > 0
366     / x = -0 & y < 0 (not odd int) return +inf (z flag)
367     / x = -inf & y not 0 or NaN return +inf (NO z flag)
368     movb %dh,%cl
369     andb $0x45,%cl
370     cmpb $0x05,%cl
371     jne  .retpinfzflag
372     / ret +inf & divide-by-0 flag
373     jmp   .retpinf

371 .retpzero:
372     fstp %st(0)
373     / x
374     / stack empty
375     / +0

377 .retnzero:
378     fstp %st(0)
379     / x
380     / stack empty
381     / -0

383 .retponeorinvalid:
384     PIC_G_LOAD(movzwl,__xpg6,eax)
385     andl $_C99SUSv3_pow_treats_Inf_as_an_even_int,%eax

```

```
386      cmpl   $0,%eax
387      je     1f
388      fstp  %st(0)      / x
389      fstp  %st(0)      / stack empty
390      fldl  %st(0)      / 1
391      ret

393 1:
394      fstp  %st(0)      / x
395      fstp  %st(0)      / stack empty
396      flds  PIC_L(Snan) / Q NaN (i flag)
397      fwait
398      ret

400 .retpinf:
401      fstp  %st(0)      / x
402      fstp  %st(0)      / stack empty
403      flds  PIC_L(pinfinity) / +inf
404      ret

406 .retpinfzflag:
407      fstp  %st(0)      / x
408      fstp  %st(0)      / stack empty
409      fldz
410      fdivrs PIC_L(one) / 1/0
411      ret

413 / Set %ecx to 2 if y is an even integer, 1 if y is an odd integer,
414 / 0 otherwise. Assume y is not zero. Do not raise inexact or modify
415 / %edx.
416 .y_is_int:
417      movl  12(%ebp),%eax
418      andl  $0x7fffffff,%eax      / |y|
419      cmpl  $0x4b800000,%eax
420      jae  1f                      / |y| >= 2^24, an even int
421      cmpl  $0x3f800000,%eax
422      jb   2f                      / |y| < 1, can't be an int
423      movl  %eax,%ecx
424      sarl  $23,%ecx
425      subl  $150,%ecx
426      negl  %ecx                    / 23 - unbiased exponent of y
427      bsfl  %eax,%eax              / index of least sig. 1 bit
428      cmpl  %ecx,%eax
429      jb   2f
430      ja   1f
431      movl  $1,%ecx
432      ret
433 1:
434      movl  $2,%ecx
435      ret
436 2:
437      xorl  %ecx,%ecx
438      ret
439      .align 4
440      SET_SIZE(powf)
unchanged_portion_omitted
```



```

122     jp     lf                / so that pow(NaN1,NaN2) returns NaN2
123     jne     lf
124     fstp    %st(0)          / x
125     ret

127 1:
128     / x ** NaN is NaN
129     movb    %dh,%cl
130     andb    $0x45,%cl
131     cmpb    $0x01,%cl      / C3=0 C2=0 C1=? C0=1 when +-NaN
132     jne     lf
133     fstp    %st(1)          / y
134     ret

136 1:
137     / NaN ** y (except 0) is NaN
138     movb    %dh,%cl
139     andb    $0x45,%cl
140     cmpb    $0x01,%cl      / C3=0 C2=0 C1=? C0=1 when +-NaN
141     jne     lf
142     fstp    %st(0)          / x
143     ret

145 1:
146     / x is not NaN
147     / x ** 1 is x
148     fcoms   PIC_L(one)      / y , x
149     fnstsw  %ax             / store status in %ax
150     sahf    %ax             / 80387 flags in %ax to 80386 flags
151     jne     lf
152     fstp    %st(0)          / x
153     ret

154 1:
155     / y is not 1
156     / +-(x > 1) ** +inf is +inf
157     / +-(x > 1) ** -inf is +0
158     / +-(x < 1) ** +inf is +0
159     / +-(x < 1) ** -inf is +inf
160     / +-(x = 1) ** +-inf is NaN
161     movb    %dh,%cl
162     andb    $0x47,%cl
163     cmpb    $0x05,%cl      / C3=0 C2=1 C1=0 C0=1 when +inf
164     je      .yispinf
165     cmpb    $0x07,%cl      / C3=0 C2=1 C1=1 C0=1 when -inf
166     je      .yisninf

167     / +0 ** +y (except 0, NaN)      is +0
168     / -0 ** +y (except 0, NaN, odd int) is +0
169     / +0 ** -y (except 0, NaN)      is +inf (z flag)
170     / -0 ** -y (except 0, NaN, odd int) is +inf (z flag)
171     / -0 ** y (odd int)             is - (+0 ** x)
172     movb    %dh,%cl
173     andb    $0x47,%cl
174     cmpb    $0x40,%cl      / C3=1 C2=0 C1=0 C0=0 when +0
175     je      .xispzero
176     cmpb    $0x42,%cl      / C3=1 C2=0 C1=1 C0=0 when -0
177     je      .xisnzero

179     / +inf ** +y (except 0, NaN)    is +inf
180     / +inf ** -y (except 0, NaN)    is +0
181     / -inf ** +y (except 0, NaN)    is -0 ** -+y (NO z flag)
182     movb    %dh,%cl
183     andb    $0x47,%cl
184     cmpb    $0x05,%cl      / C3=0 C2=1 C1=0 C0=1 when +inf
185     je      .xispinf
186     cmpb    $0x07,%cl      / C3=0 C2=1 C1=1 C0=1 when -inf
187     je      .xisninf

```

```

189     / x ** -1 is 1/x
190     fcoms   PIC_L(negone)      / y , x
191     fnstsw  %ax             / store status in %ax
192     sahf    %ax             / 80387 flags in %ax to 80386 flags
193     jne     lf
194     fld     %st(1)           / x , y , x
195     fdivrs  PIC_L(one)        / 1/x , y , x
196     jmp     .signok          / check for over/underflow

198 1:
199     / y is not -1
200     / x ** 2 is x*x
201     fcoms   PIC_L(two)        / y , x
202     fnstsw  %ax             / store status in %ax
203     sahf    %ax             / 80387 flags in %ax to 80386 flags
204     jne     lf
205     fld     %st(1)           / x , y , x
206     fld     %st(0)           / x , x , y , x
207     fmulp   %st(0)           / x^2 , y , x
208     jmp     .signok          / check for over/underflow

209 1:
210     / y is not 2
211     / x ** 1/2 is sqrt(x)
212     fcoms   PIC_L(half)       / y , x
213     fnstsw  %ax             / store status in %ax
214     sahf    %ax             / 80387 flags in %ax to 80386 flags
215     jne     lf
216     fld     %st(1)           / x , y , x
217     fsqrt   %st(1)           / sqrt(x) , y , x
218     jmp     .signok          / check for over/underflow

219 1:
220     / y is not 1/2
221     / make copies of x & y
222     fld     %st(1)           / x , y , x
223     fld     %st(1)           / y , x , y , x

224     / -x ** y (an integer) is (-1)**(y) * (+x)**(y)
225     / x ** y (x negative & y not integer) is NaN
226     movl    $0,%ecx          / track whether to flip sign of result
227     fld     %st(1)           / x , y , x , y , x
228     ftst    %st(0)           / compare %st(0) with 0
229     fnstsw  %ax             / store status in %ax
230     sahf    %ax             / 80387 flags in %ax to 80386 flags
231     fstp    %st(0)           / y , x , y , x
232     ja      .merge          / x > 0
233     / x < 0
234     call    .y_is_int
235     cmpl    $0,%ecx
236     jne     lf
237     / x < 0 & y != int so x**y = NaN (i flag)
238     fstp    %st(0)           / x , y , x
239     fstp    %st(0)           / y , x
240     fstp    %st(0)           / x
241     fstp    %st(0)           / stack empty
242     fldz
243     fdiv    %st,%st(0)       / 0/0
244     ret

246 1:
247     / x < 0 & y = int
248     fxch
249     fchs
250     fxch
251     / px > 0
252     fxch

```



```

254      / x**y = exp(y*ln(x))
255      fyl2x          / t=y*log2(px) , y , x
256      fld          %st(0) / t , t , y , x
257      frndint       / [t] , t , y , x
258      fxch          / t , [t] , y , x
259      fucom
260      fnstsw %ax      / store status in %ax
261      sahf          / 80387 flags in %ax to 80386 flags
262      je           / t is integral
263      fsub %st(1),%st / t-[t] , [t] , y , x
264      f2xm1       / 2**(t-[t])-1 , [t] , y , x
265      fadds PIC_L(one) / 2**(t-[t]) , [t] , y , x
266      fscale      / 2**t = px**y , [t] , y , x
267      jmp
268 1:
269      fstp %st(0)    / t=[t] , y , x
270      fldl          / 1 , t , y , x
271      fscale      / 1*2**t = x**y , t , y , x
272 2:
273      fstp %st(1)    / x**y , y , x
274      cmpl $1,%ecx
275      jne .signok
276      fchs          / change sign since x<0 & y=-int
277 .signok:
278      fstp %st(2)    / y , x**y
279      fstp %st(0)    / x**y
280      ret

282 / -----

284 .xispinf:
285      ftst          / compare %st(0) with 0
286      fnstsw %ax    / store status in %ax
287      sahf          / 80387 flags in %ax to 80386 flags
288      ja           / y > 0
289      jmp .retpzzero / y < 0

291 .xisninf:
292      / -inf ** +-y is -0 ** -+y
293      fchs          / -y , x
294      flds PIC_L(negzero) / -0 , -y , x
295      fstp %st(2)    / -y , -0
296      jmp .xisnzero

298 .yispinf:
299      fld %st(1)    / x , y , x
300      fabs          / |x| , y , x
301      fcomps PIC_L(one) / y , x
302      fnstsw %ax    / store status in %ax
303      sahf          / 80387 flags in %ax to 80386 flags
304      je .retponeorinvalid / x == -1 C99
305      ja .retpinf   / |x| > 1
306      jmp .retpzzero / |x| < 1

308 .yisninf:
309      fld %st(1)    / x , y , x
310      fabs          / |x| , y , x
311      fcomps PIC_L(one) / y , x
312      fnstsw %ax    / store status in %ax
313      sahf          / 80387 flags in %ax to 80386 flags
314      je .retponeorinvalid / x == -1 C99
315      ja .retpzzero / |x| > 1
316      jmp .retpinf  / |x| < 1

318 .xispzzero:
319      / y cannot be 0 or NaN ; stack has y , x

```

```

320      ftst          / compare %st(0) with 0
321      fnstsw %ax    / store status in %ax
322      sahf          / 80387 flags in %ax to 80386 flags
323      ja           / y > 0
324      / x = +0 & y < 0 so x**y = +inf
325      jmp .retpinfzflag / ret +inf & z flag

327 .xisnzero:
328      / y cannot be 0 or NaN ; stack has y , x
329      call .y_is_int
330      cmpl $1,%ecx
331      jne lf        / y is not an odd integer
332      / y is an odd integer
333      ftst          / compare %st(0) with 0
334      fnstsw %ax    / store status in %ax
335      sahf          / 80387 flags in %ax to 80386 flags
336      ja           / y > 0
337      / x = -0 & y < 0 (odd int) return -inf (z flag)
338      / x = -inf & y != 0 or NaN return -inf (NO z flag)
339      movb %dh,%cl
340      andb $0x45,%cl
341      cmpb $0x05,%cl / C3=0 C2=1 C1=? C0=1 when +-inf
342      je 2f
343      fdiv %st,%st(1) / y / x , x (raise z flag)
344 2:
345      fstp %st(0)    / x
346      fstp %st(0)    / stack empty
347      flds PIC_L(ninfinity) / -inf
348      ret

350 1:
351      / y is not an odd integer
352      ftst          / compare %st(0) with 0
353      fnstsw %ax    / store status in %ax
354      sahf          / 80387 flags in %ax to 80386 flags
355      ja           / y > 0
356      / x = -0 & y < 0 (not odd int) return +inf (z flag)
357      / x = -inf & y not 0 or NaN return +inf (NO z flag)
358      movb %dh,%cl
359      andb $0x45,%cl
360      cmpb $0x05,%cl / C3=0 C2=1 C1=? C0=1 when +-inf
361      jne .retpinfzflag / ret +inf & divide-by-0 flag
362      jmp .retpinf  / return +inf (NO z flag)

363 .retpzzero:
364      fstp %st(0)    / x
365      fstp %st(0)    / stack empty
366      fldz          / +0
367      ret

369 .retnzero:
370      fstp %st(0)    / x
371      fstp %st(0)    / stack empty
372      flds PIC_L(negzero) / -0
373      ret

375 .retponeorinvalid:
376      PIC_G_LOAD(movzwl,__xpg6,eax)
377      andl $_C99SUSv3_pow_treats_Inf_as_an_even_int,%eax
378      cmpl $0,%eax
379      je lf
380      fstp %st(0)    / x
381      fstp %st(0)    / stack empty
382      fldl          / 1
383      ret

385 1:

```

```

386     fstp   %st(0)           / x
387     fstp   %st(0)           / stack empty
388     flds   PIC_L(Snan)      / Q NaN (i flag)
389     fwait
390     ret

392 .retpinf:
393     fstp   %st(0)           / x
394     fstp   %st(0)           / stack empty
395     flds   PIC_L(pinfinity)  / +inf
396     ret

398 .retpinfzflag:
399     fstp   %st(0)           / x
400     fstp   %st(0)           / stack empty
401     fldz
402     fdivrs PIC_L(one)       / 1/0
403     ret

405 / Set %ecx to 2 if y is an even integer, 1 if y is an odd integer,
406 / 0 otherwise. Assume y is not zero. Do not raise inexact or modify
407 / %edx.
408 .y_is_int:
409     movl   28(%ebp),%eax
410     andl   $0x7fff,%eax      / exponent of y
411     cmpl   $0x403f,%eax
412     jae   1f                 / |y| >= 2^64, an even int
413     cmpl   $0x3fff,%eax
414     jb    2f                 / |y| < 1, can't be an int
415     movl   %eax,%ecx
416     subl   $0x403e,%ecx
417     negl   %ecx              / 63 - unbiased exponent of y
418     movl   20(%ebp),%eax
419     bsfl   %eax,%eax         / index of least sig. 1 bit
420     jne   3f                 / jump if 1 bit found
421     movl   24(%ebp),%eax
422     bsfl   %eax,%eax
423     addl   $32,%eax          / 32 + index of least sig. 1 bit
424 3:
425     cmpl   %ecx,%eax
426     jb    2f
427     ja    1f
428     movl   $1,%ecx
429     ret
430 1:
431     movl   $2,%ecx
432     ret
433 2:
434     xorl   %ecx,%ecx
435     ret
436     .align 4
437     SET_SIZE(powl)
unchanged portion omitted

```

new/usr/src/lib/libm/i386/src/remainder.s

1

```
*****
2101 Tue Nov 25 12:59:38 2014
new/usr/src/lib/libm/i386/src/remainder.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainder.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainder,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(remainder)
36     pushl    %ebp
37     movl    %esp,%ebp
38     fldl   16(%esp)           / load arg y
39     fldl   8(%esp)           / load arg x
40     fucom
41     fnstsw %ax
42     sahf
43     jp     .rem_loop         / if x or y is NaN, use fpreml

45     movl   20(%esp),%eax     / eax <-- hi_32(y)
46     andl   $0x7fffffff,%eax / eax <-- hi_32(|y|)
47     orl   16(%esp),%eax     / eax <-- lo_32(y)|hi_32(|y|)
48     je    .yzero_or_xinf

50     movl   12(%esp),%eax     / eax <-- hi_32(x)
51     andl   $0x7fffffff,%eax / eax <-- hi_32(|x|)
52     cmpl   $0x7ff00000,%eax
53     jne   .rem_loop
54     cmpl   $0,8(%esp)
55     je    .yzero_or_xinf

57 .rem_loop:
```

new/usr/src/lib/libm/i386/src/remainder.s

2

```
58     fpreml                / partial remainder
59     fstsw    %ax           / store status word
60     andw    $0x400,%ax    / check for incomplete reduction
61     jne     .rem_loop     / while incomplete, do fpreml again
62     fstp    %st(1)
63     leave
64     ret

66 .yzero_or_xinf:
67     PIC_SETUP(1)
68     fstp    %st(0)        / x
69     fstp    %st(0)        / empty NPX stack
70     pushl   $28           / case 28 in _SVID_libm_err
71     pushl   20(%ebp)      / pass y
72     pushl   16(%ebp)
73     pushl   12(%ebp)      / pass x
74     pushl   8(%ebp)
75     call   PIC_F(_SVID_libm_err)
76     addl   $20,%esp
77     PIC_WRAPUP
78     leave
79     ret
80     .align 4
81     SET_SIZE(remainder)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/remainderf.s

1

1399 Tue Nov 25 12:59:38 2014

new/usr/src/lib/libm/i386/src/remainderf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainderf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainderf,function)
33 #include "libm_synonyms.h"

34     ENTRY(remainderf)
35     flds    8(%esp)           / load arg y
36     flds    4(%esp)           / load arg x
37 .rem_loop:
38     fprem1                    / partial remainder
39     fstsw   %ax               / store status word
40     andw   $0x400,%ax         / check whether reduction complete
41     jne    .rem_loop          / while reduction incomplete, do fprem1
42     fstp   %st(1)
43     ret
44     .align 4
45     SET_SIZE(remainderf)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/remainder1.s

1

1399 Tue Nov 25 12:59:38 2014

new/usr/src/lib/libm/i386/src/remainder1.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remainder1.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remainder1,function)
33 #include "libm_synonyms.h"

34     ENTRY(remainder1)
35     fldt    16(%esp)          / load arg y
36     fldt    4(%esp)          / load arg x
37 .rem_loop:
38     fprem1                    / partial remainder
39     fstsw   %ax               / store status word
40     andw   $0x400,%ax        / check whether reduction complete
41     jne    .rem_loop         / while reduction incomplete, do fprem1
42     fstp   %st(1)
43     ret
44     .align 4
45     SET_SIZE(remainder1)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/remquo.s

1

1904 Tue Nov 25 12:59:39 2014

new/usr/src/lib/libm/i386/src/remquo.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remquo.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remquo,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(remquo)
36     fldl    12(%esp)           / load arg y
37     fldl    4(%esp)           / load arg x
38 .Lrem_loop:
39     fpreml                / partial remainder
40     fstsw   %ax              / store status word
41     andw   $0x400,%ax        / check whether reduction complete
42     jne   .Lrem_loop         / while reduction incomplete, do fpreml
43     fstsw   %ax
44     fwait
45     fstp   %st(1)
46     movw   %ax,%dx
47     andw   $0x4000,%dx       / get C3
48     sarw   $13,%dx
49     movw   %ax,%cx
50     andw   $0x100,%cx        / get C0
51     sarw   $6,%cx
52     addw   %cx,%dx
53     andw   $0x200,%ax        / get C1
54     sarw   $9,%ax
55     addw   %dx,%ax
56     cwtl
57     movl   8(%esp),%edx       / sign and bexp of x
```

new/usr/src/lib/libm/i386/src/remquo.s

2

```
58     movl   16(%esp),%ecx     / sign and bexp of y
59     andl   $0x80000000,%edx   / edx <- sign(x)
60     andl   $0x80000000,%ecx   / ecx <- sign(y)
61     cmpl   %edx,%ecx
62     je     .pos
63     negl   %eax              / negative n
64 .pos:
65     movl   20(%esp),%ecx
66     movl   %eax,0(%ecx)      / last 3 significant bits of quotient
67     ret
68     .align 4
69     SET_SIZE(remquo)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/remquoof.s

1

1909 Tue Nov 25 12:59:40 2014

new/usr/src/lib/libm/i386/src/remquoof.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file "remquoof.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remquoof,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

35     ENTRY(remquoof)
36     flds    8(%esp)           / load arg y
37     flds    4(%esp)           / load arg x
38 .Lremf_loop:
39     fpreml           / partial remainder
40     fstsw    %ax             / store status word
41     andw    $0x400,%ax       / check whether reduction complete
42     jne     .Lremf_loop      / while reduction incomplete, do fpreml
43     fstsw    %ax
44     fwait
45     fstp    %st(1)
46     movw    %ax,%dx
47     andw    $0x4000,%dx      / get C3
48     sarw    $13,%dx
49     movw    %ax,%cx
50     andw    $0x100,%cx      / get C0
51     sarw    $6,%cx
52     addw    %cx,%dx
53     andw    $0x200,%ax      / get C1
54     sarw    $9,%ax
55     addw    %dx,%ax
56     cwtl
57     movl    4(%esp),%edx     / sign and bexp of x
```

new/usr/src/lib/libm/i386/src/remquoof.s

2

```
58     movl    8(%esp),%ecx     / sign and bexp of y
59     andl    $0x80000000,%edx  / edx <- sign(x)
60     andl    $0x80000000,%ecx  / ecx <- sign(y)
61     cmpl    %edx,%ecx
62     je      .pos
63     negl    %eax             / negative n
64 .pos:
65     movl    12(%esp),%ecx
66     movl    %eax,0(%ecx)     / last 3 significant bits of quotient
67     ret
68     .align 4
69     SET_SIZE(remquoof)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/remquol.s

1

```
*****
1911 Tue Nov 25 12:59:40 2014
new/usr/src/lib/libm/i386/src/remquol.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "remquol.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(remquol,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"

35     ENTRY(remquol)
36     fldt    16(%esp)           / load arg y
37     fldt    4(%esp)           / load arg x
38 .Lremml_loop:
39     fpreml                / partial remainder
40     fstsw    %ax              / store status word
41     andw    $0x400,%ax        / check whether reduction complete
42     jne     .Lremml_loop      / while reduction incomplete, do fpreml
43     fstsw    %ax
44     fwait
45     fstp    %st(1)
46     movw    %ax,%dx
47     andw    $0x4000,%dx        / get C3
48     sarw    $13,%dx
49     movw    %ax,%cx
50     andw    $0x100,%cx        / get C0
51     sarw    $6,%cx
52     addw    %cx,%dx
53     andw    $0x200,%ax        / get C1
54     sarw    $9,%ax
55     addw    %dx,%ax
56     cwtl
57     movl    12(%esp),%edx      / sign and bexp of x
```

new/usr/src/lib/libm/i386/src/remquol.s

2

```
58     movl    24(%esp),%ecx      / sign and bexp of y
59     andl    $0x00008000,%edx    / edx <- sign(x)
60     andl    $0x00008000,%ecx    / ecx <- sign(y)
61     cmpl    %edx,%ecx
62     je     .pos
63     negl    %eax                / negative n
64 .pos:
65     movl    28(%esp),%ecx
66     movl    %eax,0(%ecx)        / last 3 significant bits of quotient
67     ret
68     .align 4
69     SET_SIZE(remquol)

unchanged_portion_omitted
```


new/usr/src/lib/libm/i386/src/rint.s

1

1369 Tue Nov 25 12:59:41 2014

new/usr/src/lib/libm/i386/src/rint.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "rint.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(rint,function)
33 #include "libm_synonyms.h"

34     ENTRY(rint)
35     fldl    4(%esp)          / load x
36     movl   8(%esp),%eax     / eax <-- hi_32(x)
37     andl  $0x7fffffff,%eax / eax <-- hi_32(|x|)
38     cmpl  $0x43300000,%eax / is |x| >= 2**52?
39     jae   .done            / if so, branch (already integral)
40     frndint                / [x], per rounding mode
41 .done:
42     fwait
43     ret
44     .align 4
45     SET_SIZE(rint)

unchanged portion omitted
```

new/usr/src/lib/libm/i386/src/rintf.s

1

1359 Tue Nov 25 12:59:41 2014

new/usr/src/lib/libm/i386/src/rintf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "rintf.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(rintf,function)
33 #include "libm_synonyms.h"

34     ENTRY(rintf)
35     flds    4(%esp)           / load x
36     movl   4(%esp),%eax      / eax <-- x
37     andl  $0x7fffffff,%eax  / eax <-- |x|
38     cmpl  $0x4b000000,%eax  / is |x| >= 2**23?
39     jae   .done             / if so, branch (already integral)
40     frndint                    / |x|, per rounding mode
41 .done:
42     fwait
43     ret
44     .align 4
45     SET_SIZE(rintf)

unchanged portion omitted
```

new/usr/src/lib/libm/i386/src/rintl.s

1

1186 Tue Nov 25 12:59:42 2014

new/usr/src/lib/libm/i386/src/rintl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "rintl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(rintl,function)
33 #include "libm_synonyms.h"

34     ENTRY(rintl)
35     fldt     4(%esp)           / load x
36     frndint           / [x], per rounding mode
37     fwait
38     ret
39     .align 4
40     SET_SIZE(rintl)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/rndintl.s

1

```
*****
2892 Tue Nov 25 12:59:42 2014
new/usr/src/lib/libm/i386/src/rndintl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

28
29     .file "rndintl.s"

30
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(aintl,function)
33 LIBM_ANSI_PRAGMA_WEAK(irintl,function)
34 LIBM_ANSI_PRAGMA_WEAK(anintl,function)
35 LIBM_ANSI_PRAGMA_WEAK(nintl,function)
36 #include "libm_synonyms.h"
37 #undef fabs

33     ENTRY(aintl)
34     movl    %esp,%eax
35     subl    $8,%esp
36     fstcw  -8(%eax)
37     fldt   4(%eax)
38     movw   -8(%eax),%cx
39     orw    $0x0c00,%cx
40     movw   %cx,-4(%eax)
41     fldcw  -4(%eax)           / set RD = to_zero
42     frndint
43     fstcw  -4(%eax)
44     movw   -4(%eax),%dx
45     andw   $0xf3ff,%dx
46     movw   -8(%eax),%cx
47     andw   $0x0c00,%cx
48     orw    %dx,%cx
49     movw   %cx,-8(%eax)
50     fldcw  -8(%eax)           / restore RD
51     addl   $8,%esp
52     ret
```

new/usr/src/lib/libm/i386/src/rndintl.s

2

```
53     .align 4
54     SET_SIZE(aintl)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libm/i386/src/round.s

1

```
*****
2128 Tue Nov 25 12:59:42 2014
new/usr/src/lib/libm/i386/src/round.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "round.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(round,function)
33 #include "libm_synonyms.h"
34 #undef fabs

34     .section .rodata
35     .align 4
36 .Lhalf: .float 0.5

38     ENTRY(round)
39     movl    %esp,%ecx
40     subl   $8,%esp
41     fstcw  -8(%ecx)
42     fldl   4(%ecx)
43     movw   -8(%ecx),%dx
44     andw   $0xf3ff,%dx
45     movw   %dx,-4(%ecx)
46     fldcw  -4(%ecx)          / set RD = to_nearest
47     fld    %st(0)
48     frndint          / [x],x
49     fstcw  -4(%ecx)
50     movw   -4(%ecx),%dx
51     andw   $0xf3ff,%dx
52     movw   -8(%ecx),%ax
53     andw   $0x0c00,%ax
54     orw   %dx,%ax
55     movw   %ax,-8(%ecx)
56     fldcw  -8(%ecx)          / restore RD
```

new/usr/src/lib/libm/i386/src/round.s

2

```
57     fucom          / check if x is already an integer
58     fstsw   %ax
59     sahf
60     jp      0f
61     je      0f
62     fxch
63     fsub    %st(1),%st          / x,[x]
64     fabs
65     PIC_SETUP(1)
66     fcoms  PIC_L(.Lhalf)
67     PIC_WRAPUP
68     fnstsw %ax
69     sahf
70     jae    2f          / if |x-[x]| = 0.5 goto halfway,
71                                / most cases will not take branch.
72 0:
73     addl   $8,%esp
74     fstp  %st(0)
75     ret
76 2:
77     / x = n+0.5, recompute round(x) as x+sign(x)*0.5
78     fldl   4(%ecx)          / x, 0.5, [x]
79     movl   8(%ecx),%eax          / high part of x
80     andl   $0x80000000,%eax
81     jnz   3f
82     faddp
83     addl   $8,%esp
84     fstp  %st(1)
85     ret
86 3:
87     / here, x is negative, so return x-0.5
88     fsubp  %st,%st(1)          / x-0.5,[x]
89     addl   $8,%esp
90     fstp  %st(1)
91     ret
92     .align 4
93     SET_SIZE(round)

_____unchanged_portion_omitted_____
```

```

*****
2147 Tue Nov 25 12:59:43 2014
new/usr/src/lib/libm/i386/src/roundl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "roundl.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(roundl,function)
33 #include "libm_synonyms.h"
34 #undef fabs

34     .section .rodata
35     .align 4
36 .Lhalf: .float 0.5

38     ENTRY(roundl)
39     movl    %esp,%ecx
40     subl   $8,%esp
41     fstcw  -8(%ecx)
42     fldt   4(%ecx)
43     movw  -8(%ecx),%dx
44     andw  $0xf3ff,%dx
45     movw  %dx,-4(%ecx)
46     fldcw -4(%ecx)          / set RD = to_nearest
47     fld   %st(0)
48     frndint                / [x],x
49     fstcw -4(%ecx)
50     movw  -4(%ecx),%dx
51     andw  $0xf3ff,%dx
52     movw  -8(%ecx),%ax
53     andw  $0x0c00,%ax
54     orw  %dx,%ax
55     movw  %ax,-8(%ecx)
56     fldcw -8(%ecx)          / restore RD

```

```

57     fucom                                / check if x is already an integer
58     fstsw  %ax
59     sahf
60     jp     0f
61     je     0f
62     fxch
63     fsub   %st(1),%st                    / x,[x]
64     fabs
65     PIC_SETUP(1)
66     fcoms PIC_L(.Lhalf)
67     PIC_WRAPUP
68     fnstsw %ax
69     sahf
70     jae   2f                            / if |x-[x]| = 0.5 goto halfway,
71                                           / most cases will not take branch.
72 0:
73     addl   $8,%esp
74     fstp  %st(0)
75     ret
76 2:
77     / x = n+0.5, recompute roundl(x) as x+sign(x)*0.5
78     fldt  4(%ecx)                       / x, 0.5, [x]
79     movw  12(%ecx),%ax                   / sign+exp of x
80     andw  $0x8000,%ax                    / look at sign bit
81     jnz   3f
82     faddp
83     addl   $8,%esp
84     fstp  %st(1)
85     ret
86 3:
87     / here, x is negative, so return x-0.5
88     fsubp  %st,%st(1)                    / x-0.5,[x]
89     addl   $8,%esp
90     fstp  %st(1)
91     ret
92     .align 4
93     SET_SIZE(roundl)

```

unchanged_portion_omitted

new/usr/src/lib/libm/i386/src/scalbn.s

1

1212 Tue Nov 25 12:59:43 2014

new/usr/src/lib/libm/i386/src/scalbn.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalbn.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalbn,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalbn)
35     fldl    12(%esp)           / convert N to extended
36     fldl    4(%esp)           / push x
37     fscale
38     fstp    %st(1)
39     ret
40     .align 4
41     SET_SIZE(scalbn)
```

```
_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/scalblnf.s

1

1216 Tue Nov 25 12:59:44 2014

new/usr/src/lib/libm/i386/src/scalblnf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalblnf.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalblnf,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalblnf)
```

```
35     fldl    8(%esp)           / convert N to extended
```

```
36     flds    4(%esp)           / push x
```

```
37     fscale
```

```
38     fstp    %st(1)
```

```
39     ret
```

```
40     .align 4
```

```
41     SET_SIZE(scalblnf)
```

```
_____ unchanged_portion_omitted
```


new/usr/src/lib/libm/i386/src/scalblnl.s

1

1245 Tue Nov 25 12:59:45 2014

new/usr/src/lib/libm/i386/src/scalblnl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalblnl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalblnl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalblnl)
35     fildl    16(%esp)           / convert 32-bit integer N
36                                     / to extended-double
37     fldt    4(%esp)           / push x
38     fscale
39     fstp    %st(1)
40     ret
41     .align  4
42     SET_SIZE(scalblnl)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/scalbn.s

1

1208 Tue Nov 25 12:59:45 2014

new/usr/src/lib/libm/i386/src/scalbn.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "scalbn.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(scalbn,function)
33 #include "libm_synonyms.h"

34     ENTRY(scalbn)
35     fldl    12(%esp)          / convert N to extended
36     fldl    4(%esp)          / push x
37     fscale
38     fstp    %st(1)
39     ret
40     .align  4
41     SET_SIZE(scalbn)

unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/scalbnf.s

1

1212 Tue Nov 25 12:59:45 2014

new/usr/src/lib/libm/i386/src/scalbnf.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalbnf.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalbnf,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalbnf)
35     fildl    8(%esp)          / convert N to extended
36     flds    4(%esp)          / push x
37     fscale
38     fstp    %st(1)
39     ret
40     .align  4
41     SET_SIZE(scalbnf)
```

```
_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/scalbnl.s

1

1241 Tue Nov 25 12:59:46 2014

new/usr/src/lib/libm/i386/src/scalbnl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file     "scalbnl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(scalbnl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(scalbnl)
35     fildl    16(%esp)           / convert 32-bit integer N
36                                     / to extended-double
37     fldt    4(%esp)           / push x
38     fscale
39     fstp    %st(1)
40     ret
41     .align  4
42     SET_SIZE(scalbnl)
```

```
    unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/sin.s

1

1322 Tue Nov 25 12:59:46 2014

new/usr/src/lib/libm/i386/src/sin.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "sin.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(sin,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"
```

```
35     ENTRY(sin)
36     PIC_SETUP(1)
37     call    PIC_F(__reduction)
38     PIC_WRAPUP
39     cmpl    $1,%eax
40     jl     .sin0
41     je     .sin1
42     cmpl    $2,%eax
43     je     .sin2
44     fcos
45     fchs
46     ret
47 .sin2:
48     fsin
49     fchs
50     ret
51 .sin1:
52     fcos
53     ret
54 .sin0:
55     fsin
56     ret
57     .align 4
```

new/usr/src/lib/libm/i386/src/sin.s

2

```
58     SET_SIZE(sin)
59     unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/sincos.s

1

```
*****
1685 Tue Nov 25 12:59:47 2014
new/usr/src/lib/libm/i386/src/sincos.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file "sincos.s"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(sincos,function)
33 #include "libm_synonyms.h"
34 #include "libm_protos.h"

35     ENTRY(sincos)
36     PIC_SETUP(1)
37     call    PIC_F(__reduction)
38     PIC_WRAPUP
39     fsincos
40     cmpl    $1,%eax
41     jl     .sincos0
42     je     .sincos1
43     cmpl    $2,%eax
44     je     .sincos2
45     / n=3
46     fchs
47     movl    12(%esp),%eax
48     fstpl  0(%eax)
49     movl    16(%esp),%eax
50     fstpl  0(%eax)
51     fwait
52     ret
53 .sincos2:
54     / n=2
55     fchs
56     movl    16(%esp),%eax
57     fstpl  0(%eax)
```

new/usr/src/lib/libm/i386/src/sincos.s

2

```
58     fchs
59     movl    12(%esp),%eax
60     fstpl  0(%eax)
61     fwait
62     ret
63 .sincos1:
64     / n=1
65     movl    12(%esp),%eax
66     fstpl  0(%eax)
67     fchs
68     movl    16(%esp),%eax
69     fstpl  0(%eax)
70     fwait
71     ret
72 .sincos0:
73     / n=0
74     movl    16(%esp),%eax
75     fstpl  0(%eax)
76     movl    12(%esp),%eax
77     fstpl  0(%eax)
78     fwait
79     ret
80     .align 4
81     SET_SIZE(sincos)
unchanged_portion_omitted
```

new/usr/src/lib/libm/i386/src/sqrtl.s

1

1145 Tue Nov 25 12:59:47 2014

new/usr/src/lib/libm/i386/src/sqrtl.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2005 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "sqrtl.s"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(sqrtl,function)
```

```
33 #include "libm_synonyms.h"
```

```
34     ENTRY(sqrtl)
35     fldt    4(%esp)
36     fsqrt
37     ret
38     .align 4
39     SET_SIZE(sqrtl)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/tan.s

1

1293 Tue Nov 25 12:59:47 2014

new/usr/src/lib/libm/i386/src/tan.s

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29     .file "tan.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(tan,function)
33 #include "libm_synonyms.h"
33 #include "libm_protos.h"
```

```
35     ENTRY(tan)
36     PIC_SETUP(1)
37     call    PIC_F(__reduction)
38     PIC_WRAPUP
39     andl    $1,%eax
40     cmpl   $0,%eax
41     je     .tan1
42     fptan
43     fdivp  %st,%st(1)
44     fchs
45     ret
46 .tan1:
47     fptan
48     fstp   %st(0)
49     ret
50     .align 4
51     SET_SIZE(tan)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/trunc.s

1

```
*****
1443 Tue Nov 25 12:59:48 2014
new/usr/src/lib/libm/i386/src/trunc.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "trunc.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(trunc,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(trunc)
35 movl %esp,%eax
36 subl $8,%esp
37 fstcw -8(%eax)
38 fldl 4(%eax)
39 movw -8(%eax),%cx
40 orw $0x0c00,%cx
41 movw %cx,-4(%eax)
42 fldcw -4(%eax) / set RD = to_zero
43 frndint
44 fstcw -4(%eax)
45 movw -4(%eax),%dx
46 andw $0xf3ff,%dx
47 movw -8(%eax),%cx
48 andw $0x0c00,%cx
49 orw %dx,%cx
50 movw %cx,-8(%eax)
51 fldcw -8(%eax) / restore RD
52 addl $8,%esp
53 ret
54 .align 4
55 SET_SIZE(trunc)
```

unchanged portion omitted

new/usr/src/lib/libm/i386/src/truncl.s

1

```
*****
1447 Tue Nov 25 12:59:48 2014
new/usr/src/lib/libm/i386/src/truncl.s
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "truncl.s"
```

```
31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(truncl,function)
33 #include "libm_synonyms.h"
```

```
34 ENTRY(truncl)
35 movl %esp,%eax
36 subl $8,%esp
37 fstcw -8(%eax)
38 fldt 4(%eax)
39 movw -8(%eax),%cx
40 orw $0x0c00,%cx
41 movw %cx,-4(%eax)
42 fldcw -4(%eax) / set RD = to_zero
43 frndint
44 fstcw -4(%eax)
45 movw -4(%eax),%dx
46 andw $0xf3ff,%dx
47 movw -8(%eax),%cx
48 andw $0x0c00,%cx
49 orw %dx,%cx
50 movw %cx,-8(%eax)
51 fldcw -8(%eax) / restore RD
52 addl $8,%esp
53 ret
54 .align 4
55 SET_SIZE(truncl)
```

unchanged portion omitted

new/usr/src/lib/libm/sparc/src/copysign.S

1

1239 Tue Nov 25 12:59:49 2014

new/usr/src/lib/libm/sparc/src/copysign.S

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "copysign.S"
```

```
31 #include "libm.h"
```

```
32 LIBM_ANSI_PRAGMA_WEAK(copysign,function)
```

```
33 #include "libm_synonyms.h"
```

```
34 ENTRY(copysign)
35 sethi    %hi(0x80000000),%o3
36 andn    %o0,%o3,%o0
37 and     %o2,%o3,%o2
38 or      %o2,%o0,%o0
39 std     %o0,[%sp+0x48]
40 retl
41 ldd     [%sp+0x48],%f0
```

```
43 SET_SIZE(copysign)
```

```
unchanged_portion_omitted
```

new/usr/src/lib/libm/sparc/src/fabs.S

1

1220 Tue Nov 25 12:59:50 2014

new/usr/src/lib/libm/sparc/src/fabs.S

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */
```

```
29 .file "fabs.S"
```

```
31 #include "libm.h"
```

```
32 LIB_ANSI_PRAGMA_WEAK(fabs,function)
```

```
33 #include "libm_synonyms.h"
```

```
34 ENTRY(fabs)
35 sethi    %hi(0x80000000),%o2
36 andn    %o0,%o2,%o0
37 std     %o0,[%sp+0x48]
38 nop
39 nop
40 nop
41 nop
42 nop
43 nop
44 retl
45 ldd     [%sp+0x48],%f0
```

```
47 SET_SIZE(fabs)
```

```
_____ unchanged_portion_omitted
```

new/usr/src/lib/libm/sparc/src/nextafter.S

1

2545 Tue Nov 25 12:59:50 2014

new/usr/src/lib/libm/sparc/src/nextafter.S

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26  * Use is subject to license terms.
27 */

29     .file     "nextafter.S"

31 #include "libm.h"
32 LIBM_ANSI_PRAGMA_WEAK(nextafter,function)
33     .weak    _nextafter
34     .type    _nextafter,#function
35 _nextafter = __nextafter
36 #include "libm_synonyms.h"
37 #include "libm_protos.h"

39 #if defined(LIBM_BUILD) && !defined(ELFOBJ)
40 #define mENTRY(x)      ENTRY(__libm/**/x)
41 #define mName(x)      NAME(__libm/**/x)
42 #else
43 #define mENTRY(x)      ENTRY(x)
44 #define mName(x)      NAME(x)
45 #endif

38     RO_DATA
39     .align  8
40 .lconstant:
41 two54     = 0x00
42     .word   0x43500000,0x0          ! 2**54
43 twom54    = 0x08
```

new/usr/src/lib/libm/sparc/src/nextafter.S

2

```
44     .word   0x3c900000,0x0          ! 2**-54
45 tiny      = 0x10
46     .word   0x00100000,0x0          ! tiny

48 ! variable using fp
49 x         = -0x8
50 y         = -0x10

52     ENTRY(nextafter)
53     save   %sp,-128,%sp
54     PIC_SETUP(17)
55     std    %i0,[%fp+x]
56     or     %g0,%i0,%o0              ! save original arguments
57     or     %g0,%i1,%o1
58     std    %i2,[%fp+y]
59     or     %g0,%i2,%o2
60     or     %g0,%i3,%o3
61     ldd    [%fp+x],%f2              ! x
62     ldd    [%fp+y],%f0              ! y
63     fcmpd  %f2,%f0                  ! x:y
64     PIC_SET(17,.lconstant,10)
65     sethi  %hi(0x80000000),%l1
66     andn   %i0,%l1,%l4
67     fbe    9f                       ! next_return
68     nop
69     fbu,a  9f                       ! next_return
70     fmuld  %f2,%f0,%f0              ! + -> * for Cheetah
71     orcc   %i1,%l4,%g0              ! see if x is zero
72     bne    1f
73     tst    %i0
74     ! x is zero, return sign(y)*min
75     and    %i2,%l1,%i0
76     ba     4f                       ! next_final
77     mov    1,%i1
78 1:      bge    2f
79     nop
80     ! x is negative
81     fbl    1f                       ! next_subulp
82     nop
83     fbg    3f                       ! next_addulp
84     nop
85 2:      fbl    3f                       ! next_addulp
86     nop
87     nop
88 1:      subcc  %i1,1,%i1              ! next_subulp
89     ba     4f                       ! next_final
90     subx   %i0,0,%i0
91     subx   %i0,0,%i0
92 3:      addcc  %i1,1,%i1              ! next_addulp
93     addx   %i0,0,%i0
94     addx   %i0,0,%i0
95 4:      sethi  %hi(0x7ff00000),%l3    ! next_final
96     std    %i0,[%fp+x]
97     andcc  %i0,%l3,%i2
98     be,a   1f                       ! xflow
99     ldd    [%l0+tiny],%f2
100    cmp    %i2,%l3
101    bne,a  9f                       ! next_return
102    ldd    [%fp+x],%f0
103    call   NAME(_SVID_libm_err)      ! overflow
104    call   mName(_SVID_libm_err)     ! overflow
105    or     %g0,46,%o4
106    ba     9f
107    nop
108 1:      nop                       ! xflow
```

new/usr/src/lib/libm/sparc/src/nextafter.S

3

```
109      fmuld   %f2,%f2,%f2
110      ldd     [%fp+x],%f0
111 9:      ! next_return
112      ret
113      restore

115      SET_SIZE(nextafter)
unchanged_portion_omitted
```

new/usr/src/lib/libmvec/Makefile.com

1

5589 Tue Nov 25 12:59:50 2014

new/usr/src/lib/libmvec/Makefile.com

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011 Nexenta Systems, Inc. All rights reserved.
14 #
```

16 LIBMDIR = \$(SRC)/lib/libm

```
18 mvecOBSJS = \
19     _vTBL_atan1.o \
20     _vTBL_atan2.o \
21     _vTBL_rsqrto.o \
22     _vTBL_sincos.o \
23     _vTBL_sincos2.o \
24     _vTBL_sqrtfo.o \
25     _vatan.o \
26     _vatan2.o \
27     _vatan2fo.o \
28     _vatanfo.o \
29     _vc_abs.o \
30     _vc_exp.o \
31     _vc_log.o \
32     _vc_pow.o \
33     _vcos.o \
34     _vcosbig.o \
35     _vcosbigfo.o \
36     _vcosfo.o \
37     _vexp.o \
38     _vexpfo.o \
39     _vhypot.o \
40     _vhypotfo.o \
41     _vlog.o \
42     _vlogfo.o \
43     _vpow.o \
44     _vpowfo.o \
45     _vrem_pio2m.o \
46     _vrhypot.o \
47     _vrhypotfo.o \
48     _vrsqrto.o \
49     _vrsqrtofo.o \
50     _vsin.o \
51     _vsinbig.o \
52     _vsinbigfo.o \
```

new/usr/src/lib/libmvec/Makefile.com

2

```
53     _vsincos.o \
54     _vsincosbig.o \
55     _vsincosbigfo.o \
56     _vsincosfo.o \
57     _vsinfo.o \
58     _vsqrto.o \
59     _vsqrtofo.o \
60     _vz_abs.o \
61     _vz_exp.o \
62     _vz_log.o \
63     _vz_pow.o \
64     vatan2.o \
65     vatan2fo.o \
66     vatan.o \
67     vatanfo.o \
68     vc_abs.o \
69     vc_exp.o \
70     vc_log.o \
71     vc_pow.o \
72     vcos.o \
73     vcosfo.o \
74     vexp.o \
75     vexpfo.o \
76     vhypot.o \
77     vhypotfo.o \
78     vlog.o \
79     vlogfo.o \
80     vpow.o \
81     vpowfo.o \
82     vrhypot.o \
83     vrhypotfo.o \
84     vrsqrto.o \
85     vrsqrtofo.o \
86     vsin.o \
87     vsincos.o \
88     vsincosfo.o \
89     vsinfo.o \
90     vsqrto.o \
91     vsqrtofo.o \
92     vz_abs.o \
93     vz_exp.o \
94     vz_log.o \
95     vz_pow.o \
96     #end
```

```
97 mvecvisCOBJS = \
98     _vTBL_atan1.o \
99     _vTBL_atan2.o \
100     _vTBL_rsqrto.o \
101     _vTBL_sincos.o \
102     _vTBL_sincos2.o \
103     _vTBL_sqrtfo.o \
104     _vcosbig.o \
105     _vcosbigfo.o \
106     _vrem_pio2m.o \
107     _vsinbig.o \
108     _vsinbigfo.o \
109     _vsincosbig.o \
110     _vsincosbigfo.o \
111     _vsincosbigfo.o \
112     #end
```

```
112 mvecvisSOBJS = \
113     _vatan.o \
114     _vatan2.o \
```

```

115     __vatan2f.o \
116     __vatanf.o \
117     __vcos.o \
118     __vcosf.o \
119     __vexp.o \
120     __vexpf.o \
121     __vhypot.o \
122     __vhypotf.o \
123     __vlog.o \
124     __vlogf.o \
125     __vpow.o \
126     __vpowf.o \
127     __vrhypot.o \
128     __vrhypotf.o \
129     __vrsqrt.o \
130     __vrsqrtf.o \
131     __vsin.o \
132     __vsincos.o \
133     __vsincosf.o \
134     __vsinf.o \
135     __vsqrt.o \
136     __vsqrtf.o
138     __vsqrtf.o \
139 #end

138 mvecvis2COBJS = \
139     __vTBL_sincos.o \
140     __vTBL_sincos2.o \
141     __vTBL_sqrtf.o \
142     __vcosbig.o \
143     __vcosbig_ultra3.o \
144     __vrem_pio2m.o \
145     __vsinbig.o \
146     __vsinbig_ultra3.o
149     __vsinbig_ultra3.o \
150 #end

148 mvecvis2SOBJS = \
149     __vcos_ultra3.o \
150     __vlog_ultra3.o \
151     __vsin_ultra3.o \
152     __vsqrtf_ultra3.o
156     __vsqrtf_ultra3.o \
157 #end

154 include $(SRC)/lib/Makefile.lib
155 include $(SRC)/lib/Makefile.rootfs
156 include $(LIBMDIR)/Makefile.libm.com

158 LIBS = $(DYNLIB)
159 SRCDIR = ../common/
160 DYNFLAGS += $(ZIGNORE)
165 DYNFLAGS += -zignore

162 LINTERROFF = -erroff=E_FP_DIVISION_BY_ZERO
163 LINTERROFF += -erroff=E_FP_INVALID
164 LINTERROFF += -erroff=E_BAD_PTR_CAST_ALIGN
165 LINTERROFF += -erroff=E_ASSIGNMENT_CAUSE_LOSS_PREC
166 LINTERROFF += -erroff=E_FUNC_SET_NOT_USED

168 LINTFLAGS += $(LINTERROFF)
169 LINTFLAGS64 += $(LINTERROFF)
170 LINTFLAGS64 += -errchk=longptr64

172 CLAGS += $(LINTERROFF)
173 CFLAGS64 += $(LINTERROFF)

```

```

180 ASDEF += -DLIBMVEC_SO_BUILD

175 FLTRPATH_sparc = $$ORIGIN/cpu/$$ISALIST/libmvec_isa.so.1
176 FLTRPATH_sparcv9 = $$ORIGIN/./cpu/$$ISALIST/sparcv9/libmvec_isa.so.1
177 FLTRPATH_i386 = $$ORIGIN/libmvec/$$HWCAP
178 FLTRPATH = $(FLTRPATH_$(TARGET_ARCH))

180 sparc_CFLAGS += -_cc=-W0,-xintrinsic
181 sparcv9_CFLAGS += -_cc=-W0,-xintrinsic
182 CPPFLAGS_i386 += -Dfabs=__fabs

184 SRCS_mvec_i386 = ../common/_vsqrtf.c

186 SRCS_mvec_sparc = $(SRCS_mvec_i386)

188 SRCS_mvec_sparcv9 = $(SRCS_mvec_i386)
191 CPPFLAGS += -DLIBMVEC_SO_BUILD

193 SRCS_mvec_i386 = \
194     ../common/_vsqrtf.c \
195     #end

197 SRCS_mvec_sparc = \
198     $(SRCS_mvec_i386) \
199     #end
200 SRCS_mvec_sparcv9 = \
201     $(SRCS_mvec_i386) \
202     #end

191 SRCS_mvec = \
192     $(SRCS_mvec_$(TARGETMACH)) \
193     ../common/_vTBL_atan1.c \
194     ../common/_vTBL_atan2.c \
195     ../common/_vTBL_rsqrt.c \
196     ../common/_vTBL_sincos.c \
197     ../common/_vTBL_sincos2.c \
198     ../common/_vTBL_sqrtf.c \
199     ../common/_vatan.c \
200     ../common/_vatan2.c \
201     ../common/_vatan2f.c \
202     ../common/_vatanf.c \
203     ../common/_vc_abs.c \
204     ../common/_vc_exp.c \
205     ../common/_vc_log.c \
206     ../common/_vc_pow.c \
207     ../common/_vcos.c \
208     ../common/_vcosbig.c \
209     ../common/_vcosbigf.c \
210     ../common/_vcosf.c \
211     ../common/_vexp.c \
212     ../common/_vexpf.c \
213     ../common/_vhypot.c \
214     ../common/_vhypotf.c \
215     ../common/_vlog.c \
216     ../common/_vlogf.c \
217     ../common/_vpow.c \
218     ../common/_vpowf.c \
219     ../common/_vrem_pio2m.c \
220     ../common/_vrhypot.c \
221     ../common/_vrhypotf.c \
222     ../common/_vrsqrt.c \
223     ../common/_vrsqrtf.c \
224     ../common/_vsin.c \
225     ../common/_vsinbig.c \
226     ../common/_vsinbigf.c \

```



```
227 ../common/___vsincos.c \
228 ../common/___vsincosbig.c \
229 ../common/___vsincosbigf.c \
230 ../common/___vsincosf.c \
231 ../common/___vsinf.c \
232 ../common/___vsqrt.c \
233 ../common/___vz_abs.c \
234 ../common/___vz_exp.c \
235 ../common/___vz_log.c \
236 ../common/___vz_pow.c \
237 ../common/vatan2.c \
238 ../common/vatan2f.c \
239 ../common/vatan.c \
240 ../common/vatanf.c \
241 ../common/vc_abs.c \
242 ../common/vc_exp.c \
243 ../common/vc_log.c \
244 ../common/vc_pow.c \
245 ../common/vcos.c \
246 ../common/vcosf.c \
247 ../common/vexp.c \
248 ../common/vexpf.c \
249 ../common/vhypot.c \
250 ../common/vhypotf.c \
251 ../common/vlog.c \
252 ../common/vlogf.c \
253 ../common/vpow.c \
254 ../common/vpowf.c \
255 ../common/vrhypot.c \
256 ../common/vrhypotf.c \
257 ../common/vrsqrt.c \
258 ../common/vrsqrtf.c \
259 ../common/vsin.c \
260 ../common/vsincos.c \
261 ../common/vsincosf.c \
262 ../common/vsinf.c \
263 ../common/vsqrt.c \
264 ../common/vsqrtf.c \
265 ../common/vz_abs.c \
266 ../common/vz_exp.c \
267 ../common/vz_log.c \
268 ../common/vz_pow.c \
281 ../common/vz_pow.c \
282 #end
```

```
270 .KEEP_STATE:
```

```
272 all: $(LIBS)
```

```
274 lint: lintcheck
```

```
276 pics/%.o: ../$(TARGET_ARCH)/src/%.S
```

```
277 $(COMPILE.s) -o $@ $<
```

```
278 $(POST_PROCESS_O)
```

```
280 pics/%.o: ../common/$(CHIP)/%.S
```

```
281 $(COMPILE.s) -o $@ $<
```

```
282 $(POST_PROCESS_O)
```

new/usr/src/lib/libmvec/common/__vhypot.c

1

```
*****
8456 Tue Nov 25 12:59:51 2014
new/usr/src/lib/libmvec/common/__vhypot.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #include <sys/isa_defs.h>
31 #include "libm_synonyms.h"
31 #include "libm_inlines.h"

33 #ifdef _LITTLE_ENDIAN
34 #define HI(x)      *(1+(int*)x)
35 #define LO(x)      *(unsigned*)x
36 #else
37 #define HI(x)      *(int*)x
38 #define LO(x)      *(1+(unsigned*)x)
39 #endif

41 #ifdef __RESTRICT
42 #define restrict _Restrict
43 #else
44 #define restrict
45 #endif

47 /* double hypot(double x, double y)
48  *
49  * Method :
50  *   1. Special cases:
51  *       x or y is +Inf or -Inf           => +Inf
52  *       x or y is NaN                   => QNaN
53  *   2. Computes hypot(x,y):
54  *       hypot(x,y) = m * sqrt(xnm * xnm + ynm * ynm)
55  *   Where:
56  *       m = max(|x|, |y|)
57  *       xnm = x * (1/m)
```

new/usr/src/lib/libmvec/common/__vhypot.c

2

```
58 *          ynm = y * (1/m)
59 *
60 *   Compute xnm * xnm + ynm * ynm by simulating
61 *   multi-precision arithmetic.
62 *
63 * Accuracy:
64 *   Maximum error observed: less than 0.872 ulp after 16.777.216.000
65 *   results.
66 */

69 #define sqrt __sqrt

68 extern double sqrt(double);
69 extern double fabs(double);

71 static const unsigned long long LCONST[] = {
72 0x41b0000000000000ULL, /* D2ON28 = 2 ** 28      */
73 0x0010000000000000ULL, /* D2ONM1022 = 2 ** -1022 */
74 0x7fd0000000000000ULL /* D2ONP1022 = 2 ** 1022  */
75 };
__unchanged_portion_omitted__
```

new/usr/src/lib/libmvec/common/__vhypotf.c

1

```
*****
4145 Tue Nov 25 12:59:52 2014
new/usr/src/lib/libmvec/common/__vhypotf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 #include "libm_synonyms.h"
30 #include "libm_inlines.h"
31
32 #ifdef __RESTRICT
33 #define restrict _Restrict
34 #else
35 #define restrict
36 #endif
37
38 #define sqrt __sqrt
39
40 extern double sqrt(double);
41
42 void
43 __vhypotf(int n, float * restrict x, int stridex, float * restrict y,
44           int stridey, float * restrict z, int stridez)
45 {
46     float      x0, x1, x2, y0, y1, y2, z0, z1, z2, *pz0, *pz1, *pz2;
47     unsigned   hx0, hx1, hx2, hy0, hy1, hy2;
48     int        i, j0, j1, j2;
49
50     do
51     {
52         LOOP0:
53             hx0 = *(unsigned*)x & ~0x80000000;
54             hy0 = *(unsigned*)y & ~0x80000000;
55             *(unsigned*)&x0 = hx0;
56             *(unsigned*)&y0 = hy0;
57             if (hy0 > hx0)
```

new/usr/src/lib/libmvec/common/__vhypotf.c

2

```
56     {
57         i = hy0 - hx0;
58         j0 = hy0 & 0x7f800000;
59         if (hx0 == 0)
60             i = 0x7f800000;
61     }
62     else
63     {
64         i = hx0 - hy0;
65         j0 = hx0 & 0x7f800000;
66         if (hy0 == 0)
67             i = 0x7f800000;
68         else if (hx0 == 0)
69             i = 0x7f800000;
70     }
71     if (i >= 0x0c800000 || j0 >= 0x7f800000)
72     {
73         z0 = x0 + y0;
74         if (hx0 == 0x7f800000)
75             z0 = x0;
76         else if (hy0 == 0x7f800000)
77             z0 = y0;
78         else if (hx0 > 0x7f800000 || hy0 > 0x7f800000)
79             z0 = *x + *y;
80         *z = z0;
81         x += stridex;
82         y += stridey;
83         z += stridez;
84         i = 0;
85         if (--n <= 0)
86             break;
87         goto LOOP0;
88     }
89     pz0 = z;
90     x += stridex;
91     y += stridey;
92     z += stridez;
93     i = 1;
94     if (--n <= 0)
95         break;
96
97 LOOP1:
98     hx1 = *(unsigned*)x & ~0x80000000;
99     hy1 = *(unsigned*)y & ~0x80000000;
100     *(unsigned*)&x1 = hx1;
101     *(unsigned*)&y1 = hy1;
102     if (hy1 > hx1)
103     {
104         i = hy1 - hx1;
105         j1 = hy1 & 0x7f800000;
106         if (hx1 == 0)
107             i = 0x7f800000;
108     }
109     else
110     {
111         i = hx1 - hy1;
112         j1 = hx1 & 0x7f800000;
113         if (hy1 == 0)
114             i = 0x7f800000;
115         else if (hx1 == 0)
116             i = 0x7f800000;
117     }
118     if (i >= 0x0c800000 || j1 >= 0x7f800000)
119     {
120         z1 = x1 + y1;
121         if (hx1 == 0x7f800000)
```

```

122         z1 = x1;
123     else if (hy1 == 0x7f800000)
124         z1 = y1;
125     else if (hx1 > 0x7f800000 || hy1 > 0x7f800000)
126         z1 = *x + *y;
127     *z = z1;
128     x += stridex;
129     y += stridey;
130     z += stridez;
131     i = 1;
132     if (--n <= 0)
133         break;
134     goto LOOP1;
135 }
136 pz1 = z;
137 x += stridex;
138 y += stridey;
139 z += stridez;
140 i = 2;
141 if (--n <= 0)
142     break;

144 LOOP2:
145 hx2 = *(unsigned*)x & ~0x80000000;
146 hy2 = *(unsigned*)y & ~0x80000000;
147 *(unsigned*)&x2 = hx2;
148 *(unsigned*)&y2 = hy2;
149 if (hy2 > hx2)
150 {
151     i = hy2 - hx2;
152     j2 = hy2 & 0x7f800000;
153     if (hx2 == 0)
154         i = 0x7f800000;
155 }
156 else
157 {
158     i = hx2 - hy2;
159     j2 = hx2 & 0x7f800000;
160     if (hy2 == 0)
161         i = 0x7f800000;
162     else if (hx2 == 0)
163         i = 0x7f800000;
164 }
165 if (i >= 0x0c800000 || j2 >= 0x7f800000)
166 {
167     z2 = x2 + y2;
168     if (hx2 == 0x7f800000)
169         z2 = x2;
170     else if (hy2 == 0x7f800000)
171         z2 = y2;
172     else if (hx2 > 0x7f800000 || hy2 > 0x7f800000)
173         z2 = *x + *y;
174     *z = z2;
175     x += stridex;
176     y += stridey;
177     z += stridez;
178     i = 2;
179     if (--n <= 0)
180         break;
181     goto LOOP2;
182 }
183 pz2 = z;

185 z0 = sqrt(x0 * (double)x0 + y0 * (double)y0);
186 z1 = sqrt(x1 * (double)x1 + y1 * (double)y1);
187 z2 = sqrt(x2 * (double)x2 + y2 * (double)y2);

```

```

188         *pz0 = z0;
189         *pz1 = z1;
190         *pz2 = z2;

192         x += stridex;
193         y += stridey;
194         z += stridez;
195         i = 0;
196     } while (--n > 0);

198     if (i > 0)
199     {
200         if (i > 1)
201         {
202             z1 = sqrt(x1 * (double)x1 + y1 * (double)y1);
203             *pz1 = z1;
204         }
205         z0 = sqrt(x0 * (double)x0 + y0 * (double)y0);
206         *pz0 = z0;
207     }
208 }
_____unchanged_portion_omitted_____

```

```

*****
11558 Tue Nov 25 12:59:52 2014
new/usr/src/lib/libmvec/common/__vrhypot.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */

30 #include <sys/isa_defs.h>
31 #include "libm_synonyms.h"
31 #include "libm_inlines.h"

33 #ifdef _LITTLE_ENDIAN
34 #define HI(x)      *(1+(int*)x)
35 #define LO(x)     *(unsigned*)x
36 #else
37 #define HI(x)     *(int*)x
38 #define LO(x)    *(1+(unsigned*)x)
39 #endif

41 #ifdef __RESTRICT
42 #define restrict _Restrict
43 #else
44 #define restrict
45 #endif

47 /* double rhypot(double x, double y)
48  *
49  * Method :
50  * 1. Special cases:
51  *    x or y = Inf          => 0
52  *    x or y = NaN        => QNaN
53  *    x and y = 0         => Inf + divide-
54  * 2. Computes rhypot(x,y):
55  *    rhypot(x,y) = m * sqrt(1/(xnm * xnm + ynm * ynm))
56  * Where:
57  *    m = 1/max(|x|,|y|)

```

```

58  *          xnm = x * m
59  *          ynm = y * m
60  *
61  * Compute 1/(xnm * xnm + ynm * ynm) by simulating
62  * multi-precision arithmetic.
63  *
64  * Accuracy:
65  * Maximum error observed: less than 0.869 ulp after 1.000.000.000
66  * results.
67 */

70 #define sqrt __sqrt

69 extern double sqrt(double);

70 extern double fabs(double);

72 static const int __vlibm_TBL_rhypot[] = {
73 /* i = [0,127]
74  * TBL[i] = 0x3fff00000 + *(int*)&(1.0 / *(double*)&(0x3fff000000000000ULL + (i <
75 0x7fe00000, 0x7fdffc07f, 0x7fdf81f8, 0x7fdf4465,
76 0x7fdf07c1, 0x7fdce07, 0x7fde9131, 0x7fde573a,
77 0x7fde1e1e, 0x7fdd5d6, 0x7fddae60, 0x7fdd77b6,
78 0x7fdd41d4, 0x7fdd0cb5, 0x7fdcd856, 0x7fdca4b3,
79 0x7fdc71c7, 0x7fdc3f8f, 0x7fdc0e07, 0x7fdbdd2b,
80 0x7fdbacf9, 0x7fdb7d6c, 0x7fdb4e81, 0x7fdb2036,
81 0x7fdaf286, 0x7fdac570, 0x7fda98ef, 0x7fda6d01,
82 0x7fda41a4, 0x7fda16d3, 0x7fd9ec8e, 0x7fd9c2d1,
83 0x7fd99999, 0x7fd970e4, 0x7fd948b0, 0x7fd920fb,
84 0x7fd8f9c1, 0x7fd8d301, 0x7fd8abc9, 0x7fd886e5,
85 0x7fd86186, 0x7fd83c97, 0x7fd81818, 0x7fd7f405,
86 0x7fd7d05f, 0x7fd7ad22, 0x7fd78a4c, 0x7fd767dc,
87 0x7fd745d1, 0x7fd72428, 0x7fd702e0, 0x7fd6e1f7,
88 0x7fd6c16c, 0x7fd6a13c, 0x7fd68168, 0x7fd661ec,
89 0x7fd642c8, 0x7fd623fa, 0x7fd60581, 0x7fd5e75b,
90 0x7fd5c988, 0x7fd5ac05, 0x7fd58ed2, 0x7fd571ed,
91 0x7fd55555, 0x7fd53909, 0x7fd51d07, 0x7fd50150,
92 0x7fd4e5e0, 0x7fd4cab8, 0x7fd4afd6, 0x7fd49539,
93 0x7fd47ae1, 0x7fd460cb, 0x7fd446f8, 0x7fd42666,
94 0x7fd41414, 0x7fd3fb01, 0x7fd3e22c, 0x7fd3c995,
95 0x7fd3b13b, 0x7fd3991c, 0x7fd38138, 0x7fd3698d,
96 0x7fd3521c, 0x7fd33ae4, 0x7fd323e3, 0x7fd30d19,
97 0x7fd2f684, 0x7fd2e025, 0x7fd2c9fb, 0x7fd2b404,
98 0x7fd29e41, 0x7fd288b0, 0x7fd27350, 0x7fd25e22,
99 0x7fd24924, 0x7fd23456, 0x7fd21fb7, 0x7fd20b47,
100 0x7fd1f704, 0x7fd1e2ef, 0x7fd1cfc06, 0x7fd1bb4a,
101 0x7fd1a7b9, 0x7fd19453, 0x7fd18118, 0x7fd16e06,
102 0x7fd15b1e, 0x7fd1485f, 0x7fd135c8, 0x7fd12358,
103 0x7fd11111, 0x7fd0fef0, 0x7fd0ecf5, 0x7fd0db20,
104 0x7fd0c971, 0x7fd0b7e6, 0x7fd0a681, 0x7fd0953f,
105 0x7fd08421, 0x7fd07326, 0x7fd0624d, 0x7fd05197,
106 0x7fd04104, 0x7fd03091, 0x7fd02040, 0x7fd01010,
107 };
unchanged_portion_omitted

```

```

*****
15231 Tue Nov 25 12:59:52 2014
new/usr/src/lib/libmvec/common/__vrhypotf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #include <sys/isa_defs.h>
31 #include "libm_synonyms.h"
32 #include "libm_inlines.h"

33 #ifdef _LITTLE_ENDIAN
34 #define HI(x)      *(1+(int*)x)
35 #define LO(x)     *(unsigned*)x
36 #else
37 #define HI(x)     *(int*)x
38 #define LO(x)    *(1+(unsigned*)x)
39 #endif

41 #ifdef __RESTRICT
42 #define restrict __Restrict
43 #else
44 #define restrict
45 #endif

47 /* float rhypotf(float x, float y)
48  *
49  * Method :
50  *     1. Special cases:
51  *         for x or y = Inf          => 0;
52  *         for x or y = NaN         => QNaN;
53  *         for x and y = 0          => +Inf + divide-by-zero
54  *     2. Computes d = x * x + y * y;
55  *     3. Computes reciprocal square root from:
56  *         d = m * 2**n
57  *     Where:

```

```

58  *         m = [0.5, 2],
59  *         n = ((exponent + 1) & ~1).
60  *
61  *     Then:
62  *         rsqrtf(d) = 1/sqrt( m * 2**n ) = (2 ** (-n/2)) * (1/sqrt(m))
63  *     4. Computes 1/sqrt(m) from:
64  *         1/sqrt(m) = (1/sqrt(m0)) * (1/sqrt(1 + (1/m0)*dm))
65  *     Where:
66  *         m = m0 + dm,
67  *         m0 = 0.5 * (1 + k/64) for m = [0.5, 0.5+127/256], k = [0
68  *         m0 = 1.0 * (0 + k/64) for m = [0.5+127/256, 1.0+127/128], k = [6
69  *     Then:
70  *         1/sqrt(m0), 1/m0 are looked up in a table,
71  *         1/sqrt(1 + (1/m0)*dm) is computed using approximation:
72  *         1/sqrt(1 + z) = ((a3 * z + a2) * z + a1) * z + a0
73  *         where z = [-1/64, 1/64].
74  *
75  *     Accuracy:
76  *         The maximum relative error for the approximating
77  *         polynomial is 2**(-27.87).
78  *         Maximum error observed: less than 0.535 ulp after 3.000.000.000
79  *         results.

81 #pragma align 32 (__vlibm_TBL_rhypotf)

83 static const double __vlibm_TBL_rhypotf[] = {
84  /*
85  i = [0,63]
86  TBL[2*i+0] = 1.0 / (*(double*)&(0x3ff0000000000000LL + (i << 46)));
87  TBL[2*i+1] = (double)(0.5/sqrtl(2) / sqrtl(*(double*)&(0x3ff0000000000000LL + (
88  TBL[128+2*i+0] = 1.0 / (*(double*)&(0x3ff0000000000000LL + (i << 46)));
89  TBL[128+2*i+1] = (double)(0.25 / sqrtl(*(double*)&(0x3ff0000000000000LL + (i <<
90  */
91  1.000000000000000000000000e+00, 3.5355339059327378637e-01,
92  9.8461538461538467004e-01, 3.5082320772281166965e-01,
93  9.6969696969696972388e-01, 3.4815531191139570399e-01,
94  9.5522388059701490715e-01, 3.4554737023254405992e-01,
95  9.4117647058823528106e-01, 3.4299717028501769400e-01,
96  9.2753623188405798228e-01, 3.4050261230349943009e-01,
97  9.1428571428571425717e-01, 3.3806170189140660742e-01,
98  9.0140845070422537244e-01, 3.3567254331867563133e-01,
99  8.8888888888888883955e-01, 3.3333333333333331483e-01,
100 8.7671232876712323900e-01, 3.3104235544094717802e-01,
101 8.6486486486486491287e-01, 3.2879797461071458287e-01,
102 8.5333333333333338810e-01, 3.2659863237109043599e-01,
103 8.4210526315789469010e-01, 3.2444284226152508843e-01,
104 8.3116883116883122362e-01, 3.2232918561015211356e-01,
105 8.2051282051282048435e-01, 3.2025630761017426229e-01,
106 8.1012658227848100001e-01, 3.1822291367029204023e-01,
107 8.0000000000000004441e-01, 3.1622776601683794118e-01,
108 7.9012345679012341293e-01, 3.1426968052735443360e-01,
109 7.8048780487804880757e-01, 3.1234752377721214378e-01,
110 7.7108433734939763049e-01, 3.1046021028253312224e-01,
111 7.6190476190476186247e-01, 3.0860669992418382490e-01,
112 7.5294117647058822484e-01, 3.0678599553894819740e-01,
113 7.4418604651162789665e-01, 3.0499714066520933198e-01,
114 7.3563218390804596680e-01, 3.0323921743156134756e-01,
115 7.2727272727272729291e-01, 3.015113445776362918e-01,
116 7.1910112359550559802e-01, 2.9981267559834456904e-01,
117 7.1111111111111113825e-01, 2.981423969997197031e-01,
118 7.0329670329670335160e-01, 2.9649972666444046610e-01,
119 6.9565217391304345895e-01, 2.9488391230979427160e-01,
120 6.8817204301075274309e-01, 2.9329423004270660513e-01,
121 6.8085106382978721751e-01, 2.9172998299578911663e-01,
122 6.7368421052631577428e-01, 2.9019050004400465115e-01,
123 6.666666666666662966e-01, 2.8867513459481286553e-01,

```

124 6.5979381443298967813e-01, 2.8718326344709527165e-01,
125 6.5306122448979586625e-01, 2.8571428571428569843e-01,
126 6.4646464646464651960e-01, 2.8426762180748055275e-01,
127 6.4000000000000001332e-01, 2.8284271247461900689e-01,
128 6.3366336633663367106e-01, 2.8143901789211672737e-01,
129 6.2745098039215685404e-01, 2.8005601680560193723e-01,
130 6.2135922330097081989e-01, 2.7869320571664707442e-01,
131 6.1538461538461541878e-01, 2.7735009811261457369e-01,
132 6.0952380952380957879e-01, 2.7602622373694168934e-01,
133 6.0377358490566035432e-01, 2.7472112789737807015e-01,
134 5.9813084112149528249e-01, 2.7343437080986532361e-01,
135 5.9259259259259255970e-01, 2.7216552697590867815e-01,
136 5.8715596330275232617e-01, 2.7091418459143856712e-01,
137 5.8181818181818178992e-01, 2.6967994498529684888e-01,
138 5.7657657657657657158e-01, 2.6846242208560971987e-01,
139 5.7142857142857139685e-01, 2.6726124191242439654e-01,
140 5.6637168141592919568e-01, 2.6607604209509572168e-01,
141 5.6140350877192979340e-01, 2.6490647141300877054e-01,
142 5.5652173913043478937e-01, 2.6375218935831479250e-01,
143 5.5172413793103447510e-01, 2.6261286571944508772e-01,
144 5.4700854700854706358e-01, 2.6148818018424535570e-01,
145 5.4237288135593220151e-01, 2.6037782196164771520e-01,
146 5.3781512605042014474e-01, 2.5928148942086576278e-01,
147 5.333333333333332593e-01, 2.5819888974716115326e-01,
148 5.2892561983471075848e-01, 2.5712973861329002645e-01,
149 5.2459016393442625681e-01, 2.5607375986579195004e-01,
150 5.2032520325203257539e-01, 2.5503068522533534068e-01,
151 5.16129303225806450180e-01, 2.5400025400038100942e-01,
152 5.120000000000001066e-01, 2.5298221281347033074e-01,
153 5.0793650793650790831e-01, 2.5197631533948483540e-01,
154 5.0393700787401574104e-01, 2.5098232205526344041e-01,
155 1.000000000000000000e+00, 2.500000000000000000e-01,
156 9.8461538461538467004e-01, 2.4806946917841690703e-01,
157 9.6969696969696972388e-01, 2.4618298195866547551e-01,
158 9.5522388059701490715e-01, 2.4433888871261044695e-01,
159 9.4117647058823528106e-01, 2.4253562503633296910e-01,
160 9.2753623188405798228e-01, 2.4077170617153839660e-01,
161 9.1428571428571425717e-01, 2.3904572186687872426e-01,
162 9.0140845070422537244e-01, 2.3735633163877067897e-01,
163 8.8888888888888883955e-01, 2.3570226039551583908e-01,
164 8.7671232876712323900e-01, 2.3408229439226113655e-01,
165 8.6486486486486491287e-01, 2.3249527748763856860e-01,
166 8.5333333333333338810e-01, 2.3094010767585029797e-01,
167 8.4210526315789469010e-01, 2.2941573387056177213e-01,
168 8.3116883116883122362e-01, 2.2792115291927589338e-01,
169 8.2051282051282048435e-01, 2.2645540682891915352e-01,
170 8.1012658227848100001e-01, 2.2501758018520479077e-01,
171 8.0000000000000004441e-01, 2.2360679774997896385e-01,
172 7.9012345679012341293e-01, 2.222222222222220989e-01,
173 7.8048780487804880757e-01, 2.2086305214969309541e-01,
174 7.7108433734939763049e-01, 2.1952851997938069295e-01,
175 7.6190476190476186247e-01, 2.1821789023599238999e-01,
176 7.5294117647058822484e-01, 2.1693045781865616384e-01,
177 7.4418604651162789665e-01, 2.1566554640687682354e-01,
178 7.3563218390804596680e-01, 2.1442250696755896233e-01,
179 7.2727272727272729291e-01, 2.1320071635561044232e-01,
180 7.1910112359550559802e-01, 2.1199957600127200541e-01,
181 7.111111111111113825e-01, 2.1081851067789195153e-01,
182 7.0329670329670335160e-01, 2.0965696734438366011e-01,
183 6.9565217391304345895e-01, 2.0851441405707477061e-01,
184 6.8817204301075274309e-01, 2.0739033894608505104e-01,
185 6.8085106382978721751e-01, 2.0628424925175867233e-01,
186 6.7368421052631577428e-01, 2.0519567041703082322e-01,
187 6.6666666666666662966e-01, 2.0412414523193150862e-01,
188 6.5979381443298967813e-01, 2.0306923302672380549e-01,
189 6.5306122448979586625e-01, 2.0203050891044216364e-01,

190 6.4646464646464651960e-01, 2.0100756305184241945e-01,
191 6.4000000000000001332e-01, 2.0000000000000001110e-01,
192 6.3366336633663367106e-01, 1.9900743804199783060e-01,
193 6.2745098039215685404e-01, 1.9802950859533485772e-01,
194 6.2135922330097081989e-01, 1.9706585563285863860e-01,
195 6.1538461538461541878e-01, 1.9611613513818404453e-01,
196 6.0952380952380957879e-01, 1.9518001458970662965e-01,
197 6.0377358490566035432e-01, 1.9425717247145282696e-01,
198 5.9813084112149528249e-01, 1.9334729780913270658e-01,
199 5.9259259259259255970e-01, 1.9245008972987526219e-01,
200 5.8715596330275232617e-01, 1.9156525704423027490e-01,
201 5.8181818181818178992e-01, 1.9069251784911847580e-01,
202 5.7657657657657657158e-01, 1.8983159915049979682e-01,
203 5.7142857142857139685e-01, 1.8898223650461362655e-01,
204 5.6637168141592919568e-01, 1.8814417367671945613e-01,
205 5.6140350877192979340e-01, 1.8731716231633879777e-01,
206 5.5652173913043478937e-01, 1.8650096164806276300e-01,
207 5.5172413793103447510e-01, 1.8569533817705186074e-01,
208 5.4700854700854706358e-01, 1.8490006540840969729e-01,
209 5.4237288135593220151e-01, 1.8411492357966466327e-01,
210 5.3781512605042014474e-01, 1.8333969940564226464e-01,
211 5.333333333333332593e-01, 1.8257418583505535814e-01,
212 5.2892561983471075848e-01, 1.8181818181818182323e-01,
213 5.2459016393442625681e-01, 1.8107149208503706128e-01,
214 5.2032520325203257539e-01, 1.8033392693348646030e-01,
215 5.1612903225806450180e-01, 1.7960530202677491007e-01,
216 5.120000000000001066e-01, 1.7888543819998317663e-01,
217 5.0793650793650790831e-01, 1.7817416127494958844e-01,
218 5.0393700787401574104e-01, 1.7747130188322274291e-01,
219 };
222 #define fabsf __fabsf
221 extern float fabsf(float);
223 static const double
224 A0 = 9.99999997962321453275e-01,
225 A1 = -4.99999998166077580600e-01,
226 A2 = 3.75066768969515586277e-01,
227 A3 = -3.12560092408808548438e-01;
229 static void
230 __vrhypotf_n(int n, float * restrict px, int stridex, float * restrict py,
231 int stridey, float * restrict pz, int stridez);
233 #pragma no_inline(__vrhypotf_n)
235 #define RETURN(ret)
236 {
237 *pz = (ret);
238 pz += stridez;
239 if (n_n == 0)
240 {
241 spx = px; spy = py; spz = pz;
242 ay0 = *(int*)py;
243 continue;
244 }
245 n--;
246 break;
247 }
unchanged_portion_omitted

```

*****
10451 Tue Nov 25 12:59:53 2014
new/usr/src/lib/libmvec/common/_vrsqrt.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #include <sys/isa_defs.h>
31 #include "libm_synonyms.h"
31 #include "libm_inlines.h"

33 #ifdef _LITTLE_ENDIAN
34 #define HI(x)      *(1+(int*)x)
35 #define LO(x)     *(unsigned*)x
36 #else
37 #define HI(x)     *(int*)x
38 #define LO(x)    *(1+(unsigned*)x)
39 #endif

41 #ifdef __RESTRICT
42 #define restrict _Restrict
43 #else
44 #define restrict
45 #endif

47 /* double rsqrt(double x)
48  *
49  * Method :
50  * 1. Special cases:
51  *     for x = NaN           => QNaN;
52  *     for x = +Inf         => 0;
53  *     for x is negative, -Inf => QNaN + invalid;
54  *     for x = +0          => +Inf + divide-by-zero
55  *     for x = -0         => -Inf + divide-by-zero
56  * 2. Computes reciprocal square root from:
57  *     x = m * 2**n

```

```

58  * Where:
59  *     m = [0.5, 2),
60  *     n = ((exponent + 1) & ~1).
61  * Then:
62  *     rsqrt(x) = 1/sqrt(m * 2**n) = (2 ** (-n/2)) * (1/sqrt(m))
63  * 2. Computes 1/sqrt(m) from:
64  *     1/sqrt(m) = (1/sqrt(m0)) * (1/sqrt(1 + (1/m0)*dm))
65  * Where:
66  *     m = m0 + dm,
67  *     m0 = 0.5 * (1 + k/64) for m = [0.5, 0.5+127/256), k = [0
68  *     m0 = 1.0 * (0 + k/64) for m = [0.5+127/256, 1.0+127/128), k = [6
69  *     m0 = 2.0 for m = [1.0+127/128, 2.0), k = 12
70  * Then:
71  *     1/sqrt(m0) is looked up in a table,
72  *     1/m0 is computed as (1/sqrt(m0)) * (1/sqrt(m0)).
73  *     1/sqrt(1 + (1/m0)*dm) is computed using approximation:
74  *     1/sqrt(1 + z) = (((a6 * z + a5) * z + a4) * z + a3)
75  *                   * z + a2) * z + a1) * z + a0
76  *     where z = [-1/128, 1/128].
77  *
78  * Accuracy:
79  *     The maximum relative error for the approximating
80  *     polynomial is 2**(-56.26).
81  *     Maximum error observed: less than 0.563 ulp after 1.500.000.000
82  *     results.
83 */

86 #define sqrt __rsqrt

85 extern double sqrt (double);
86 extern const double __vlibm_TBL_rsqrt[];

88 static void
89 __vrsqrt_n(int n, double * restrict px, int stridex, double * restrict py, int s

91 #pragma no_inline(__vrsqrt_n)

93 #define RETURN(ret)
94 {
95     *py = (ret);
96     py += stridex;
97     if (n_n == 0)
98     {
99         spx = px; spy = py;
100        hx = HI(px);
101        continue;
102    }
103    n--;
104    break;
105 }

_____unchanged_portion_omitted_____

```


new/usr/src/lib/libmvec/common/__vrsqrtf.c

1

16770 Tue Nov 25 12:59:53 2014
new/usr/src/lib/libmvec/common/__vrsqrtf.c
5261 libm should stop using synonyms.h
5298 fabs is 0-sized, confuses dis(1) and others
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2 * CDDL HEADER START
3 *
4 * The contents of this file are subject to the terms of the
5 * Common Development and Distribution License (the "License").
6 * You may not use this file except in compliance with the License.
7 *
8 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27 * Use is subject to license terms.
28 */
29
30 #include "libm_synonyms.h"
30 #include "libm_inlines.h"
31
32 #ifdef __RESTRICT
33 #define restrict _Restrict
34 #else
35 #define restrict
36 #endif
37
38 /* float rsqrtf(float x)
39 *
40 * Method :
41 * 1. Special cases:
42 *     for x = NaN                => QNaN;
43 *     for x = +Inf              => 0;
44 *     for x is negative, -Inf  => QNaN + invalid;
45 *     for x = +0                => +Inf + divide-by-zero
46 *     for x = -0                => -Inf + divide-by-zero
47 * 2. Computes reciprocal square root from:
48 *     x = m * 2**n
49 * Where:
50 *     m = [0.5, 2),
51 *     n = ((exponent + 1) & ~1).
52 * Then:
53 *     rsqrtf(x) = 1/sqrt( m * 2**n ) = (2 ** (-n/2)) * (1/sqrt(m))
54 * 2. Computes 1/sqrt(m) from:
55 *     1/sqrt(m) = (1/sqrt(m0)) * (1/sqrt(1 + (1/m0)*dm))
56 * Where:
57 *     m = m0 + dm,
```

new/usr/src/lib/libmvec/common/__vrsqrtf.c

2

58 * m0 = 0.5 * (1 + k/64) for m = [0.5, 0.5+127/256), k = [0
59 * m0 = 1.0 * (0 + k/64) for m = [0.5+127/256, 1.0+127/128), k = [6
60 * Then:
61 * 1/sqrt(m0), 1/m0 are looked up in a table,
62 * 1/sqrt(1 + (1/m0)*dm) is computed using approximation:
63 * 1/sqrt(1 + z) = ((a3 * z + a2) * z + a1) * z + a0
64 * where z = [-1/64, 1/64].
65 *
66 * Accuracy:
67 * The maximum relative error for the approximating
68 * polynomial is 2**(-27.87).
69 * Maximum error observed: less than 0.534 ulp for the
70 * whole float type range.
71 */

74 #define sqrtf __sqrtf

73 extern float sqrtf(float);

```
75 static const double __TBL_rsqrtf[] = {
76 /*
77 i = [0,63]
78 TBL[2*i ] = 1 / (*(double*)&(0x3fe0000000000000ULL + (i << 46))) * 2**-24;
79 TBL[2*i+1] = 1 / sqrtl(*(double*)&(0x3fe0000000000000ULL + (i << 46)));
80 i = [64,127]
81 TBL[2*i ] = 1 / (*(double*)&(0x3fe0000000000000ULL + (i << 46))) * 2**-23;
82 TBL[2*i+1] = 1 / sqrtl(*(double*)&(0x3fe0000000000000ULL + (i << 46)));
83 */
84 1.1920928955078125000e-07, 1.4142135623730951455e+00,
85 1.1737530048076923728e-07, 1.4032928308912466786e+00,
86 1.1559688683712121533e-07, 1.3926212476455828160e+00,
87 1.1387156016791044559e-07, 1.3821894809301762397e+00,
88 1.1219697840073529256e-07, 1.3719886811400707760e+00,
89 1.1057093523550724772e-07, 1.3620104492139977204e+00,
90 1.0899135044642856803e-07, 1.3522468075656264297e+00,
91 1.0745626100352112918e-07, 1.3426901732747025253e+00,
92 1.0596381293402777190e-07, 1.333333333333332593e+00,
93 1.0451225385273972023e-07, 1.3241694217637887121e+00,
94 1.0309992609797297870e-07, 1.3151918984428583315e+00,
95 1.0172526041666667320e-07, 1.3063945294843617440e+00,
96 1.0038677014802631022e-07, 1.2977713690461003537e+00,
97 9.9083045860389616921e-08, 1.2893167424406084542e+00,
98 9.7812750400641022247e-08, 1.2810252304406970492e+00,
99 9.6574614319620251657e-08, 1.2728916546811681609e+00,
100 9.5367431640625005294e-08, 1.2649110640673517647e+00,
101 9.4190055941358019463e-08, 1.2570787221094177344e+00,
102 9.3041396722560978838e-08, 1.2493900951088485751e+00,
103 9.1920416039156631290e-08, 1.2418408411301324890e+00,
104 9.0826125372023804482e-08, 1.2344267996967352996e+00,
105 8.9757582720588234048e-08, 1.2271439821557927896e+00,
106 8.8713889898255812722e-08, 1.2199885626608373279e+00,
107 8.7694190014367814875e-08, 1.2129568697262453902e+00,
108 8.6697665127840911497e-08, 1.2060453783110545167e+00,
109 8.5723534058988761666e-08, 1.1992507023933782762e+00,
110 8.4771050347222225457e-08, 1.1925695879998878812e+00,
111 8.3839500343406599951e-08, 1.1859989066577618644e+00,
112 8.2928201426630432481e-08, 1.1795356492391770864e+00,
113 8.2036500336021511923e-08, 1.1731769201708264205e+00,
114 8.1163771609042551220e-08, 1.1669199319831564665e+00,
115 8.030946118421050820e-08, 1.1607620001760186046e+00,
116 7.9472859700520828922e-08, 1.1547005383792514621e+00,
117 7.865351868556699530e-08, 1.1487330537883810866e+00,
118 7.7850964604591830522e-08, 1.1428571428571427937e+00,
119 7.7064591224747481298e-08, 1.137070487229922110e+00,
120 7.6293945312500001588e-08, 1.1313708498984760276e+00,
121 7.5538559715346535571e-08, 1.1257560715684669095e+00,
```

122 7.4797985600490195040e-08, 1.1202240672224077489e+00,
123 7.4071791565533974158e-08, 1.1147728228665882977e+00,
124 7.3359562800480773303e-08, 1.1094003924504582947e+00,
125 7.2660900297619054173e-08, 1.1041048949477667573e+00,
126 7.1975420106132072725e-08, 1.0988845115895122806e+00,
127 7.1302752628504667579e-08, 1.0937374832394612945e+00,
128 7.0642541956018514597e-08, 1.0886621079036347126e+00,
129 6.9994445240825691959e-08, 1.0836567383657542685e+00,
130 6.9358132102272723904e-08, 1.0787197799411873955e+00,
131 6.8733284065315314719e-08, 1.0738496883424388795e+00,
132 6.8119594029017853361e-08, 1.0690449676496975862e+00,
133 6.7516765763274335346e-08, 1.0643041683803828867e+00,
134 6.6924513432017540145e-08, 1.0596258856520350822e+00,
135 6.6342561141304348632e-08, 1.0550087574332591700e+00,
136 6.5770642510775861156e-08, 1.0504514628777803509e+00,
137 6.5208500267094023655e-08, 1.0459527207369814228e+00,
138 6.4655885858050847233e-08, 1.04151128784645908608e+00,
139 6.4112559086134451001e-08, 1.0371259576834630511e+00,
140 6.3578287760416665784e-08, 1.032795589886446131e+00,
141 6.3052847365702481089e-08, 1.0285189544531601058e+00,
142 6.2536020747950822927e-08, 1.0242950394631678002e+00,
143 6.2027597815040656970e-08, 1.0201227409013413627e+00,
144 6.1527375252016127325e-08, 1.0160010160015240377e+00,
145 6.1035156250000001271e-08, 1.0119288512538813229e+00,
146 6.0550750248015869655e-08, 1.0079052613579393416e+00,
147 6.0073972687007873182e-08, 1.0039292882210537616e+00,
148 1.1920928955078125000e-07, 1.000000000000000000e+00,
149 1.1737530048076923728e-07, 9.9227787671366762812e-01,
150 1.1559688683712121533e-07, 9.8473192783466190203e-01,
151 1.1387156016791044559e-07, 9.773555485044178781e-01,
152 1.1219697840073529256e-07, 9.7014250014533187638e-01,
153 1.1057093523550724772e-07, 9.6308682468615358641e-01,
154 1.0899135044642856803e-07, 9.5618288746751489704e-01,
155 1.0745626100352112918e-07, 9.4942532655508271588e-01,
156 1.0596381293402777190e-07, 9.4280904158206335630e-01,
157 1.0451225385273972023e-07, 9.3632917756904454620e-01,
158 1.0309992609797297870e-07, 9.2998110995055427441e-01,
159 1.0172526041666667320e-07, 9.2376043070340119190e-01,
160 1.0038677014802631022e-07, 9.1766293548224708854e-01,
161 9.9083045860389616921e-08, 9.1168461167710357351e-01,
162 9.7812750400641022247e-08, 9.0582162731567661407e-01,
163 9.6574614319620251657e-08, 9.0007032074081916306e-01,
164 9.5367431640625005294e-08, 8.9442719099991585541e-01,
165 9.4190055941358019463e-08, 8.8888888888888883955e-01,
166 9.3041396722560978838e-08, 8.8345220859877238162e-01,
167 9.1920416039156631290e-08, 8.7811407991752277180e-01,
168 9.0826125372023804482e-08, 8.7287156094396955996e-01,
169 8.9757582720588234048e-08, 8.6772183127462465535e-01,
170 8.8713889898255812722e-08, 8.6266218562750729415e-01,
171 8.7694190014367814875e-08, 8.5769002787023584933e-01,
172 8.6697665127840911497e-08, 8.5280286542244176928e-01,
173 8.5723534058988761666e-08, 8.4799830400508802164e-01,
174 8.477105034722225457e-08, 8.4327404271156780613e-01,
175 8.3839500343406599951e-08, 8.3862786937753464045e-01,
176 8.2928201426630432481e-08, 8.3405765622829908246e-01,
177 8.2036500336021511923e-08, 8.2956135578434020417e-01,
178 8.1163771609042551220e-08, 8.2513699700703468931e-01,
179 8.0309416118421050820e-08, 8.2078268166812329287e-01,
180 7.9472859700520828922e-08, 8.1649658092772603446e-01,
181 7.8653551868556699530e-08, 8.1227693210689522196e-01,
182 7.7850964604591830522e-08, 8.0812203564176865456e-01,
183 7.7064591224747481298e-08, 8.0403025220736967782e-01,
184 7.6293945312500001588e-08, 8.0000000000000004441e-01,
185 7.5538559715346535571e-08, 7.9602975216799132241e-01,
186 7.4797985600490195040e-08, 7.9211803438133943089e-01,
187 7.4071791565533974158e-08, 7.8826342253143455441e-01,

188 7.3359562800480773303e-08, 7.8446454055273617811e-01,
189 7.2660900297619054173e-08, 7.8072005835882651859e-01,
190 7.1975420106132072725e-08, 7.7702868988581130782e-01,
191 7.1302752628504667579e-08, 7.7338919123653082632e-01,
192 7.0642541956018514597e-08, 7.6980035891950104876e-01,
193 6.9994445240825691959e-08, 7.6626102817692109959e-01,
194 6.9358132102272723904e-08, 7.6277007139647390321e-01,
195 6.8733284065315314719e-08, 7.5932639660199918730e-01,
196 6.8119594029017853361e-08, 7.5592894601845450619e-01,
197 6.7516765763274335346e-08, 7.5257669470687782454e-01,
198 6.6924513432017540145e-08, 7.4926864926535519107e-01,
199 6.6342561141304348632e-08, 7.4600384659225105199e-01,
200 6.5770642510775861156e-08, 7.4278135270820744296e-01,
201 6.5208500267094023655e-08, 7.3960026163363878915e-01,
202 6.4655885858050847233e-08, 7.3645969431865865307e-01,
203 6.4112559086134451001e-08, 7.3335879762256905856e-01,
204 6.3578287760416665784e-08, 7.3029674334022143256e-01,
205 6.3052847365702481089e-08, 7.2727272727272729291e-01,
206 6.2536020747950822927e-08, 7.2428596834014824513e-01,
207 6.2027597815040656970e-08, 7.2133570773394584119e-01,
208 6.1527375252016127325e-08, 7.1842120810709964029e-01,
209 6.1035156250000001271e-08, 7.1554175279993270653e-01,
210 6.0550750248015869655e-08, 7.1269664509979835376e-01,
211 6.0073972687007873182e-08, 7.0988520753289097165e-01,
212 ;

unchanged_portion_omitted

new/usr/src/lib/libmvec/common/__vsqrt.c

1

1296 Tue Nov 25 12:59:54 2014

new/usr/src/lib/libmvec/common/__vsqrt.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #include "libm_synonyms.h"
30 #include "libm_inlines.h"

32 #ifdef __RESTRICT
33 #define restrict _Restrict
34 #else
35 #define restrict
36 #endif

39 #define sqrt __sqrt

38 extern double sqrt(double);

40 void
41 __vsqrt(int n, double * restrict x, int stridex, double * restrict y, int stride
42 {
43     for(; n > 0 ; n--)
44     {
45         *y = sqrt(*x);
46         x += stridex;
47         y += stridey;
48     }
49 }
```

new/usr/src/lib/libmvec/common/__vsqrtf.c

1

1295 Tue Nov 25 12:59:54 2014

new/usr/src/lib/libmvec/common/__vsqrtf.c

5261 libm should stop using synonyms.h

5298 fabs is 0-sized, confuses dis(1) and others

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 #ifdef __RESTRICT
31 #define restrict _Restrict
32 #else
33 #define restrict
34 #endif

36 #include "libm_synonyms.h"
36 #include "libm_inlines.h"

39 #define sqrtf __sqrtf

38 extern float sqrtf(float);

40 void
41 __vsqrtf(int n, float * restrict x, int stridex, float * restrict y, int stridey)
42 {
43     for(; n > 0 ; n--)
44     {
45         *y = sqrtf(*x);
46         x += stridex;
47         y += stridey;
48     }
49 }
```

new/usr/src/lib/libmvec/common/vcos_.c

1

```
*****
1235 Tue Nov 25 12:59:55 2014
new/usr/src/lib/libmvec/common/vcos_.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 extern void __vcos(int, double *, int, double *, int);

32 #if !defined(LIBMVEC_SO_BUILD)
33 #if defined(ARCH_v8plusa) || defined(ARCH_v8plusb) || defined(ARCH_v9a) || defin
34 #define CHECK_ULTRA3
35 #endif
36 #endif /* !defined(LIBMVEC_SO_BUILD) */

38 #ifdef CHECK_ULTRA3
39 #include <strings.h>
40 #define sysinfo _sysinfo
41 #include <sys/systeminfo.h>

43 #define BUFLLEN 257

45 static int use_ultra3 = 0;

47 extern void __vcos_ultra3(int, double *, int, double *, int);
48 #endif

32 #pragma weak vcos_ = __vcos_

34 /* just invoke the serial function */
35 void
36 __vcos_(int *n, double *x, int *stridex, double *y, int *stridey)
37 {
56 #ifdef CHECK_ULTRA3
```

new/usr/src/lib/libmvec/common/vcos_.c

2

```
57     int         u;
58     char        buf[BUFLLEN];

60     u = use_ultra3;
61     if (!u) {
62         /* use __vcos_ultra3 on Cheetah (and ???) */
63         if (sysinfo(SI_ISALIST, buf, BUFLLEN) > 0 && !strncmp(buf, "sparc
64             u = 3;
65         else
66             u = 1;
67         use_ultra3 = u;
68     }
69     if (u & 2)
70         __vcos_ultra3(*n, x, *stridex, y, *stridey);
71     else
72 #endif
38     __vcos(*n, x, *stridex, y, *stridey);
39 }
    unchanged_portion_omitted
```

new/usr/src/lib/libmvec/common/vis/_vcos_ultra3.S

1

```
*****
49892 Tue Nov 25 12:59:55 2014
new/usr/src/lib/libmvec/common/vis/_vcos_ultra3.S
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "_vcos_ultra3.S"

31 #include "libm.h"
32 #if defined(LIBMVEC_SO_BUILD)
33     .weak    __vcos
34     .type    __vcos, #function
35     __vcos = __vcos_ultra3
36 #endif

36     RO_DATA
37     .align  64
38 constants:
39     .word   0x42c80000,0x00000000    ! 3 * 2^44
40     .word   0x43380000,0x00000000    ! 3 * 2^51
41     .word   0x3fe45f30,0x6dc9c883    ! invpio2
42     .word   0x3ff921fb,0x54442c00    ! pio2_1
43     .word   0x3d318469,0x898cc400    ! pio2_2
44     .word   0x3a71701b,0x839a2520    ! pio2_3
45     .word   0xbfc55555,0x55555533    ! pp1
46     .word   0x3f811111,0x10e7d53b    ! pp2
47     .word   0xbf2a0167,0xe6b3cf9b    ! pp3
48     .word   0xbfdfffff,0xffffffff    ! qq1
49     .word   0x3fa55555,0x54f88ed0    ! qq2
50     .word   0xbf56c12c,0xdd185f60    ! qq3

52 ! local storage indices

54 #define xsave          STACK_BIAS-0x8
```

new/usr/src/lib/libmvec/common/vis/_vcos_ultra3.S

2

```
55 #define ysave          STACK_BIAS-0x10
56 #define nsave          STACK_BIAS-0x14
57 #define xsxsave        STACK_BIAS-0x18
58 #define sysave         STACK_BIAS-0x1c
59 #define biguns         STACK_BIAS-0x20
60 #define nk3            STACK_BIAS-0x24
61 #define nk2            STACK_BIAS-0x28
62 #define nk1            STACK_BIAS-0x2c
63 #define nk0            STACK_BIAS-0x30
64 #define junk           STACK_BIAS-0x38
65 ! sizeof temp storage - must be a multiple of 16 for V9
66 #define tmps           0x40

68 ! register use

70 ! i0  n
71 ! i1  x
72 ! i2  stridex
73 ! i3  y
74 ! i4  stridex
75 ! i5  0x80000000

77 ! l0  hx0
78 ! l1  hx1
79 ! l2  hx2
80 ! l3  hx3
81 ! l4  k0
82 ! l5  k1
83 ! l6  k2
84 ! l7  k3

86 ! the following are 64-bit registers in both V8+ and V9

88 ! g1  __vlibm_TBL_sincos2
89 ! g5  scratch

91 ! o0  py0
92 ! o1  py1
93 ! o2  py2
94 ! o3  py3
95 ! o4  0x3e400000
96 ! o5  0x3fe921fb,0x4099251e
97 ! o7  scratch

99 ! f0  hx0
100 ! f2
101 ! f4
102 ! f6
103 ! f8  hx1
104 ! f10
105 ! f12
106 ! f14
107 ! f16 hx2
108 ! f18
109 ! f20
110 ! f22
111 ! f24 hx3
112 ! f26
113 ! f28
114 ! f30
115 ! f32
116 ! f34
117 ! f36
118 ! f38

120 #define c3two44 %f40
```

```

121 #define c3two51 %f42
122 #define invpio2 %f44
123 #define pio2_1 %f46
124 #define pio2_2 %f48
125 #define pio2_3 %f50
126 #define ppl %f52
127 #define pp2 %f54
128 #define pp3 %f56
129 #define qq1 %f58
130 #define qq2 %f60
131 #define qq3 %f62

133 ENTRY(__vcos_ultra3)
134 save %sp,-SA(MINFRAME)-tmpls,%sp
135 PIC_SETUP(17)
136 PIC_SET(17,constants,o0)
137 PIC_SET(17,__vlibm_TBL_sincos2,o1)
138 mov %o1,%g1
139 wr %g0,0x82,%asi ! set %asi for non-faulting loads
140 #ifdef __sparcv9
141 stx %i1,[%fp+xsave] ! save arguments
142 stx %i3,[%fp+ysave]
143 #else
144 st %i1,[%fp+xsave] ! save arguments
145 st %i3,[%fp+ysave]
146 #endif
147 st %i0,[%fp+nsave]
148 st %i2,[%fp+sxsave]
149 st %i4,[%fp+sysave]
150 st %g0,[%fp+biguns] ! biguns = 0
151 ldd [%o0+0x00],c3two44 ! load/set up constants
152 ldd [%o0+0x08],c3two51
153 ldd [%o0+0x10],invpio2
154 ldd [%o0+0x18],pio2_1
155 ldd [%o0+0x20],pio2_2
156 ldd [%o0+0x28],pio2_3
157 ldd [%o0+0x30],ppl
158 ldd [%o0+0x38],pp2
159 ldd [%o0+0x40],pp3
160 ldd [%o0+0x48],qq1
161 ldd [%o0+0x50],qq2
162 ldd [%o0+0x58],qq3
163 sethi %hi(0x80000000),%i5
164 sethi %hi(0x3e400000),%o4
165 sethi %hi(0x3fe921fb),%o5
166 or %o5,%lo(0x3fe921fb),%o5
167 sllx %o5,32,%o5
168 sethi %hi(0x4099251e),%o7
169 or %o7,%lo(0x4099251e),%o7
170 or %o5,%o7,%o5
171 sll %i2,3,%i2 ! scale strides
172 sll %i4,3,%i4
173 add %fp,junk,%o1 ! loop prologue
174 add %fp,junk,%o2
175 add %fp,junk,%o3
176 ld [%i1],%i0 ! *x
177 ld [%i1],%f0
178 ld [%i1+4],%f3
179 andn %i0,%i5,%i0 ! mask off sign
180 add %i1,%i2,%i1 ! x += stridex
181 ba .loop0
182 nop

184 ! 16-byte aligned
185 .align 16
186 .loop0:

```

```

187 lda [%i1]%asi,%i1 ! preload next argument
188 sub %i0,%o4,%g5
189 sub %o5,%i0,%o7
190 fabss %f0,%f2

192 lda [%i1]%asi,%f8
193 orcc %o7,%g5,%g0
194 mov %i3,%o0 ! py0 = y
195 bl,pn %icc,.range0 ! hx < 0x3e400000 or hx > 0x4099251e

197 ! delay slot
198 lda [%i1+4]%asi,%f11
199 addcc %i0,-1,%i0
200 add %i3,%i4,%i3 ! y += stridey
201 ble,pn %icc,.last1

203 ! delay slot
204 andn %i1,%i5,%i1
205 add %i1,%i2,%i1 ! x += stridex
206 faddd %f2,c3two44,%f4
207 st %f15,[%o1+4]

209 .loop1:
210 lda [%i1]%asi,%i2 ! preload next argument
211 sub %i1,%o4,%g5
212 sub %o5,%i1,%o7
213 fabss %f8,%f10

215 lda [%i1]%asi,%f16
216 orcc %o7,%g5,%g0
217 mov %i3,%o1 ! py1 = y
218 bl,pn %icc,.range1 ! hx < 0x3e400000 or hx > 0x4099251e

220 ! delay slot
221 lda [%i1+4]%asi,%f19
222 addcc %i0,-1,%i0
223 add %i3,%i4,%i3 ! y += stridey
224 ble,pn %icc,.last2

226 ! delay slot
227 andn %i2,%i5,%i2
228 add %i1,%i2,%i1 ! x += stridex
229 faddd %f10,c3two44,%f12
230 st %f23,[%o2+4]

232 .loop2:
233 lda [%i1]%asi,%i3 ! preload next argument
234 sub %i2,%o4,%g5
235 sub %o5,%i2,%o7
236 fabss %f16,%f18

238 lda [%i1]%asi,%f24
239 orcc %o7,%g5,%g0
240 mov %i3,%o2 ! py2 = y
241 bl,pn %icc,.range2 ! hx < 0x3e400000 or hx > 0x4099251e

243 ! delay slot
244 lda [%i1+4]%asi,%f27
245 addcc %i0,-1,%i0
246 add %i3,%i4,%i3 ! y += stridey
247 ble,pn %icc,.last3

249 ! delay slot
250 andn %i3,%i5,%i3
251 add %i1,%i2,%i1 ! x += stridex
252 faddd %f18,c3two44,%f20

```

```

253      st      %f31,[%o3+4]

255 .loop3:
256      sub     %l3,%o4,%g5
257      sub     %o5,%l3,%o7
258      fabss  %f24,%f26
259      st      %f5,[%fp+nk0]

261      orcc   %o7,%g5,%g0
262      mov     %i3,%o3          ! py3 = y
263      bl,pn  %icc,.range3     ! hx < 0x3e400000 or > hx 0x4099251e
264 ! delay slot
265      st      %f13,[%fp+nk1]

267 !!! DONE?
268 .cont:
269      srlx   %o5,32,%o7
270      add     %i3,%i4,%i3     ! y += stridey
271      fmovs  %f3,%f1
272      st      %f21,[%fp+nk2]

274      sub     %o7,%l0,%l0
275      sub     %o7,%l1,%l1
276      faddd  %f26,c3two44,%f28
277      st      %f29,[%fp+nk3]

279      sub     %o7,%l2,%l2
280      sub     %o7,%l3,%l3
281      fmovs  %f11,%f9

283      or     %l0,%l1,%l0
284      or     %l2,%l3,%l2
285      fmovs  %f19,%f17

287      fmovs  %f27,%f25
288      fmuld  %f0,invpio2,%f6  ! x * invpio2, for medium range

290      fmuld  %f8,invpio2,%f14
291      ld     [%fp+nk0],%l4

293      fmuld  %f16,invpio2,%f22
294      ld     [%fp+nk1],%l5

296      orcc   %l0,%l2,%g0
297      bl,pn  %icc,.medium
298 ! delay slot
299      fmuld  %f24,invpio2,%f30
300      ld     [%fp+nk2],%l6

302      ld     [%fp+nk3],%l7
303      sll   %l4,5,%l4          ! k
304      fcmpd  %fcc0,%f0,pio2_3  ! x < pio2_3 iff x < 0

306      sll   %l5,5,%l5
307      ldd   [%l4+%g1],%f4
308      fcmpd  %fcc1,%f8,pio2_3

310      sll   %l6,5,%l6
311      ldd   [%l5+%g1],%f12
312      fcmpd  %fcc2,%f16,pio2_3

314      sll   %l7,5,%l7
315      ldd   [%l6+%g1],%f20
316      fcmpd  %fcc3,%f24,pio2_3

318      ldd   [%l7+%g1],%f28

```

```

319      fsubd  %f2,%f4,%f2      ! x -= __vlibm_TBL_sincos2[k]

321      fsubd  %f10,%f12,%f10

323      fsubd  %f18,%f20,%f18

325      fsubd  %f26,%f28,%f26

327      fmuld  %f2,%f2,%f0      ! z = x * x

329      fmuld  %f10,%f10,%f8

331      fmuld  %f18,%f18,%f16

333      fmuld  %f26,%f26,%f24

335      fmuld  %f0,qq3,%f6

337      fmuld  %f8,qq3,%f14

339      fmuld  %f16,qq3,%f22

341      fmuld  %f24,qq3,%f30

343      faddd  %f6,qq2,%f6
344      fmuld  %f0,pp2,%f4

346      faddd  %f14,qq2,%f14
347      fmuld  %f8,pp2,%f12

349      faddd  %f22,qq2,%f22
350      fmuld  %f16,pp2,%f20

352      faddd  %f30,qq2,%f30
353      fmuld  %f24,pp2,%f28

355      fmuld  %f0,%f6,%f6
356      faddd  %f4,pp1,%f4

358      fmuld  %f8,%f14,%f14
359      faddd  %f12,pp1,%f12

361      fmuld  %f16,%f22,%f22
362      faddd  %f20,pp1,%f20

364      fmuld  %f24,%f30,%f30
365      faddd  %f28,pp1,%f28

367      faddd  %f6,qq1,%f6
368      fmuld  %f0,%f4,%f4
369      add    %l4,%g1,%l4

371      faddd  %f14,qq1,%f14
372      fmuld  %f8,%f12,%f12
373      add    %l5,%g1,%l5

375      faddd  %f22,qq1,%f22
376      fmuld  %f16,%f20,%f20
377      add    %l6,%g1,%l6

379      faddd  %f30,qq1,%f30
380      fmuld  %f24,%f28,%f28
381      add    %l7,%g1,%l7

383      fmuld  %f2,%f4,%f4

```



```

385      fmuld    %f10,%f12,%f12
387      fmuld    %f18,%f20,%f20
389      fmuld    %f26,%f28,%f28
391      fmuld    %f0,%f6,%f6
392      faddd    %f4,%f2,%f4
393      ldd      [%14+16],%f32
395      fmuld    %f8,%f14,%f14
396      faddd    %f12,%f10,%f12
397      ldd      [%15+16],%f34
399      fmuld    %f16,%f22,%f22
400      faddd    %f20,%f18,%f20
401      ldd      [%16+16],%f36
403      fmuld    %f24,%f30,%f30
404      faddd    %f28,%f26,%f28
405      ldd      [%17+16],%f38
407      fmuld    %f32,%f6,%f6
408      ldd      [%14+8],%f2
410      fmuld    %f34,%f14,%f14
411      ldd      [%15+8],%f10
413      fmuld    %f36,%f22,%f22
414      ldd      [%16+8],%f18
416      fmuld    %f38,%f30,%f30
417      ldd      [%17+8],%f26
419      fmuld    %f2,%f4,%f4
421      fmuld    %f10,%f12,%f12
423      fmuld    %f18,%f20,%f20
425      fmuld    %f26,%f28,%f28
427      fsubd    %f6,%f4,%f6
428      lda      [%i1]%asi,%10      ! preload next argument
430      fsubd    %f14,%f12,%f14
431      lda      [%i1]%asi,%f0
433      fsubd    %f22,%f20,%f22
434      lda      [%i1+4]%asi,%f3
436      fsubd    %f30,%f28,%f30
437      andn    %10,%i5,%10
438      add     %i1,%i2,%i1
440      faddd    %f6,%f32,%f6
441      st      %f6,[%o0]
443      faddd    %f14,%f34,%f14
444      st      %f14,[%o1]
446      faddd    %f22,%f36,%f22
447      st      %f22,[%o2]
449      faddd    %f30,%f38,%f30
450      st      %f30,[%o3]

```

```

451      addcc    %i0,-1,%i0
453      bg,pt    %icc,.loop0
454 ! delay slot
455      st      %f7,[%o0+4]
457      ba,pt    %icc,.end
458 ! delay slot
459      nop
462      .align   16
463 .medium:
464      faddd    %f6,c3two51,%f4
465      st      %f5,[%fp+nk0]
467      faddd    %f14,c3two51,%f12
468      st      %f13,[%fp+nk1]
470      faddd    %f22,c3two51,%f20
471      st      %f21,[%fp+nk2]
473      faddd    %f30,c3two51,%f28
474      st      %f29,[%fp+nk3]
476      fsubd    %f4,c3two51,%f6
478      fsubd    %f12,c3two51,%f14
480      fsubd    %f20,c3two51,%f22
482      fsubd    %f28,c3two51,%f30
484      fmuld    %f6,pio2_1,%f2
485      ld      [%fp+nk0],%10      ! n
487      fmuld    %f14,pio2_1,%f10
488      ld      [%fp+nk1],%11
490      fmuld    %f22,pio2_1,%f18
491      ld      [%fp+nk2],%12
493      fmuld    %f30,pio2_1,%f26
494      ld      [%fp+nk3],%13
496      fsubd    %f0,%f2,%f0
497      fmuld    %f6,pio2_2,%f4
498      add     %10,1,%10
500      fsubd    %f8,%f10,%f8
501      fmuld    %f14,pio2_2,%f12
502      add     %11,1,%11
504      fsubd    %f16,%f18,%f16
505      fmuld    %f22,pio2_2,%f20
506      add     %12,1,%12
508      fsubd    %f24,%f26,%f24
509      fmuld    %f30,pio2_2,%f28
510      add     %13,1,%13
512      fsubd    %f0,%f4,%f32
514      fsubd    %f8,%f12,%f34
516      fsubd    %f16,%f20,%f36

```

```

518     fsubd    %f24,%f28,%f38
520     fsubd    %f0,%f32,%f0
521     fcmlpe32 %f32,pio2_3,%14      ! x <= pio2_3 iff x < 0
523     fsubd    %f8,%f34,%f8
524     fcmlpe32 %f34,pio2_3,%15
526     fsubd    %f16,%f36,%f16
527     fcmlpe32 %f36,pio2_3,%16
529     fsubd    %f24,%f38,%f24
530     fcmlpe32 %f38,pio2_3,%17
532     fsubd    %f0,%f4,%f0
533     fmuld    %f6,pio2_3,%f6
534     sll     %14,30,%14          ! if (x < 0) n = -n ^ 2
536     fsubd    %f8,%f12,%f8
537     fmuld    %f14,pio2_3,%f14
538     sll     %15,30,%15
540     fsubd    %f16,%f20,%f16
541     fmuld    %f22,pio2_3,%f22
542     sll     %16,30,%16
544     fsubd    %f24,%f28,%f24
545     fmuld    %f30,pio2_3,%f30
546     sll     %17,30,%17
548     fsubd    %f6,%f0,%f6
549     sra     %14,31,%14
551     fsubd    %f14,%f8,%f14
552     sra     %15,31,%15
554     fsubd    %f22,%f16,%f22
555     sra     %16,31,%16
557     fsubd    %f30,%f24,%f30
558     sra     %17,31,%17
560     fsubd    %f32,%f6,%f0      ! reduced x
561     xor     %10,%14,%10
563     fsubd    %f34,%f14,%f8
564     xor     %11,%15,%11
566     fsubd    %f36,%f22,%f16
567     xor     %12,%16,%12
569     fsubd    %f38,%f30,%f24
570     xor     %13,%17,%13
572     fabsd    %f0,%f2
573     sub     %10,%14,%10
575     fabsd    %f8,%f10
576     sub     %11,%15,%11
578     fabsd    %f16,%f18
579     sub     %12,%16,%12
581     fabsd    %f24,%f26
582     sub     %13,%17,%13

```

```

584     faddd    %f2,c3two44,%f4
585     st      %f5,[%fp+nk0]
586     and     %14,2,%14
588     faddd    %f10,c3two44,%f12
589     st      %f13,[%fp+nk1]
590     and     %15,2,%15
592     faddd    %f18,c3two44,%f20
593     st      %f21,[%fp+nk2]
594     and     %16,2,%16
596     faddd    %f26,c3two44,%f28
597     st      %f29,[%fp+nk3]
598     and     %17,2,%17
600     fsubd    %f32,%f0,%f4
601     xor     %10,%14,%10
603     fsubd    %f34,%f8,%f12
604     xor     %11,%15,%11
606     fsubd    %f36,%f16,%f20
607     xor     %12,%16,%12
609     fsubd    %f38,%f24,%f28
610     xor     %13,%17,%13
612     fzero   %f38
613     ld      [%fp+nk0],%14
615     fsubd    %f4,%f6,%f6      ! w
616     ld      [%fp+nk1],%15
618     fsubd    %f12,%f14,%f14
619     ld      [%fp+nk2],%16
621     fnegd   %f38,%f38
622     ld      [%fp+nk3],%17
623     sll     %14,5,%14      ! k
625     fsubd    %f20,%f22,%f22
626     sll     %15,5,%15
628     fsubd    %f28,%f30,%f30
629     sll     %16,5,%16
631     fand    %f0,%f38,%f32      ! sign bit of x
632     ldd    [%14+%g1],%f4
633     sll     %17,5,%17
635     fand    %f8,%f38,%f34
636     ldd    [%15+%g1],%f12
638     fand    %f16,%f38,%f36
639     ldd    [%16+%g1],%f20
641     fand    %f24,%f38,%f38
642     ldd    [%17+%g1],%f28
644     fsubd    %f2,%f4,%f2      ! x -= __vlibm_TBL_sincos2[k]
646     fsubd    %f10,%f12,%f10
648     fsubd    %f18,%f20,%f18

```

```

649      nop
651      fsubd   %f26,%f28,%f26
652      nop

654 ! 16-byte aligned
655      fmuld   %f2,%f2,%f0          ! z = x * x
656      andcc   %l0,1,%g0
657      bz,pn   %icc,.case8
658 ! delay slot
659      fxor    %f6,%f32,%f32

661      fmuld   %f10,%f10,%f8
662      andcc   %l1,1,%g0
663      bz,pn   %icc,.case4
664 ! delay slot
665      fxor    %f14,%f34,%f34

667      fmuld   %f18,%f18,%f16
668      andcc   %l2,1,%g0
669      bz,pn   %icc,.case2
670 ! delay slot
671      fxor    %f22,%f36,%f36

673      fmuld   %f26,%f26,%f24
674      andcc   %l3,1,%g0
675      bz,pn   %icc,.case1
676 ! delay slot
677      fxor    %f30,%f38,%f38

679 !.case0:
680      fmuld   %f0,qq3,%f6          ! cos(x0)

682      fmuld   %f8,qq3,%f14        ! cos(x1)

684      fmuld   %f16,qq3,%f22       ! cos(x2)

686      fmuld   %f24,qq3,%f30       ! cos(x3)

688      faddd   %f6,qq2,%f6
689      fmuld   %f0,pp2,%f4

691      faddd   %f14,qq2,%f14
692      fmuld   %f8,pp2,%f12

694      faddd   %f22,qq2,%f22
695      fmuld   %f16,pp2,%f20

697      faddd   %f30,qq2,%f30
698      fmuld   %f24,pp2,%f28

700      fmuld   %f0,%f6,%f6
701      faddd   %f4,pp1,%f4

703      fmuld   %f8,%f14,%f14
704      faddd   %f12,pp1,%f12

706      fmuld   %f16,%f22,%f22
707      faddd   %f20,pp1,%f20

709      fmuld   %f24,%f30,%f30
710      faddd   %f28,pp1,%f28

712      faddd   %f6,qq1,%f6
713      fmuld   %f0,%f4,%f4
714      add     %l4,%g1,%l4

```

```

716      faddd   %f14,qq1,%f14
717      fmuld   %f8,%f12,%f12
718      add     %l5,%g1,%l5

720      faddd   %f22,qq1,%f22
721      fmuld   %f16,%f20,%f20
722      add     %l6,%g1,%l6

724      faddd   %f30,qq1,%f30
725      fmuld   %f24,%f28,%f28
726      add     %l7,%g1,%l7

728      fmuld   %f2,%f4,%f4

730      fmuld   %f10,%f12,%f12

732      fmuld   %f18,%f20,%f20

734      fmuld   %f26,%f28,%f28

736      fmuld   %f0,%f6,%f6
737      faddd   %f4,%f32,%f4
738      ldd     [%l4+16],%f0

740      fmuld   %f8,%f14,%f14
741      faddd   %f12,%f34,%f12
742      ldd     [%l5+16],%f8

744      fmuld   %f16,%f22,%f22
745      faddd   %f20,%f36,%f20
746      ldd     [%l6+16],%f16

748      fmuld   %f24,%f30,%f30
749      faddd   %f28,%f38,%f28
750      ldd     [%l7+16],%f24

752      fmuld   %f0,%f6,%f6
753      faddd   %f4,%f2,%f4
754      ldd     [%l4+8],%f32

756      fmuld   %f8,%f14,%f14
757      faddd   %f12,%f10,%f12
758      ldd     [%l5+8],%f34

760      fmuld   %f16,%f22,%f22
761      faddd   %f20,%f18,%f20
762      ldd     [%l6+8],%f36

764      fmuld   %f24,%f30,%f30
765      faddd   %f28,%f26,%f28
766      ldd     [%l7+8],%f38

768      fmuld   %f32,%f4,%f4

770      fmuld   %f34,%f12,%f12

772      fmuld   %f36,%f20,%f20

774      fmuld   %f38,%f28,%f28

776      fsubd   %f6,%f4,%f6

778      fsubd   %f14,%f12,%f14

780      fsubd   %f22,%f20,%f22

```

```

782      fsubd   %f30,%f28,%f30
784      faddd   %f6,%f0,%f6
786      faddd   %f14,%f8,%f14
788      faddd   %f22,%f16,%f22
790      faddd   %f30,%f24,%f30
791      mov     %l0,%l4
793      fnegd   %f6,%f4
794      lda     [%l1]%asi,%l0          ! preload next argument
796      fnegd   %f14,%f12
797      lda     [%l1]%asi,%f0
799      fnegd   %f22,%f20
800      lda     [%l1+4]%asi,%f3
802      fnegd   %f30,%f28
803      andn    %l0,%i5,%l0
804      add     %l1,%i2,%l1
806      andcc   %l4,2,%g0
807      fmovdnz %icc,%f4,%f6
808      st      %f6,[%o0]
810      andcc   %l1,2,%g0
811      fmovdnz %icc,%f12,%f14
812      st      %f14,[%o1]
814      andcc   %l2,2,%g0
815      fmovdnz %icc,%f20,%f22
816      st      %f22,[%o2]
818      andcc   %l3,2,%g0
819      fmovdnz %icc,%f28,%f30
820      st      %f30,[%o3]
822      addcc   %i0,-1,%i0
823      bg,pt   %icc,.loop0
824 ! delay   slot
825      st      %f7,[%o0+4]
827      ba,pt   %icc,.end
828 ! delay   slot
829      nop
831      .align  16
832 .case1:
833      fmuld   %f24,pp3,%f30          ! sin(x3)
835      fmuld   %f0,qq3,%f6          ! cos(x0)
837      fmuld   %f8,qq3,%f14         ! cos(x1)
839      fmuld   %f16,qq3,%f22        ! cos(x2)
841      faddd   %f30,pp2,%f30
842      fmuld   %f24,qq2,%f28
844      faddd   %f6,qq2,%f6
845      fmuld   %f0,pp2,%f4

```

```

847      faddd   %f14,qq2,%f14
848      fmuld   %f8,pp2,%f12
850      faddd   %f22,qq2,%f22
851      fmuld   %f16,pp2,%f20
853      fmuld   %f24,%f30,%f30
854      faddd   %f28,qq1,%f28
856      fmuld   %f0,%f6,%f6
857      faddd   %f4,pp1,%f4
859      fmuld   %f8,%f14,%f14
860      faddd   %f12,pp1,%f12
862      fmuld   %f16,%f22,%f22
863      faddd   %f20,pp1,%f20
865      faddd   %f30,pp1,%f30
866      fmuld   %f24,%f28,%f28
867      add     %l7,%g1,%l7
869      faddd   %f6,qq1,%f6
870      fmuld   %f0,%f4,%f4
871      add     %l4,%g1,%l4
873      faddd   %f14,qq1,%f14
874      fmuld   %f8,%f12,%f12
875      add     %l5,%g1,%l5
877      faddd   %f22,qq1,%f22
878      fmuld   %f16,%f20,%f20
879      add     %l6,%g1,%l6
881      fmuld   %f24,%f30,%f30
883      fmuld   %f2,%f4,%f4
885      fmuld   %f10,%f12,%f12
887      fmuld   %f18,%f20,%f20
889      fmuld   %f26,%f30,%f30
890      ldd     [%l17+8],%f24
892      fmuld   %f0,%f6,%f6
893      faddd   %f4,%f32,%f4
894      ldd     [%l4+16],%f0
896      fmuld   %f8,%f14,%f14
897      faddd   %f12,%f34,%f12
898      ldd     [%l5+16],%f8
900      fmuld   %f16,%f22,%f22
901      faddd   %f20,%f36,%f20
902      ldd     [%l6+16],%f16
904      fmuld   %f24,%f28,%f28
905      faddd   %f38,%f30,%f30
907      fmuld   %f0,%f6,%f6
908      faddd   %f4,%f2,%f4
909      ldd     [%l4+8],%f32
911      fmuld   %f8,%f14,%f14
912      faddd   %f12,%f10,%f12

```

```

913      ldd      [%15+8],%f34

915      fmuld   %f16,%f22,%f22
916      faddd   %f20,%f18,%f20
917      ldd      [%16+8],%f36

919      faddd   %f26,%f30,%f30
920      ldd      [%17+16],%f38

922      fmuld   %f32,%f4,%f4

924      fmuld   %f34,%f12,%f12

926      fmuld   %f36,%f20,%f20

928      fmuld   %f38,%f30,%f30

930      fsubd   %f6,%f4,%f6

932      fsubd   %f14,%f12,%f14

934      fsubd   %f22,%f20,%f22

936      faddd   %f30,%f28,%f30

938      faddd   %f6,%f0,%f6

940      faddd   %f14,%f8,%f14

942      faddd   %f22,%f16,%f22

944      faddd   %f30,%f24,%f30
945      mov     %10,%14

947      fnegd   %f6,%f4
948      lda     [%i1]%asi,%10      ! preload next argument

950      fnegd   %f14,%f12
951      lda     [%i1]%asi,%f0

953      fnegd   %f22,%f20
954      lda     [%i1+4]%asi,%f3

956      fnegd   %f30,%f28
957      andn    %10,%i5,%10
958      add     %i1,%i2,%i1

960      andcc   %14,2,%g0
961      fmovdnz %icc,%f4,%f6
962      st      %f6,[%o0]

964      andcc   %11,2,%g0
965      fmovdnz %icc,%f12,%f14
966      st      %f14,[%o1]

968      andcc   %12,2,%g0
969      fmovdnz %icc,%f20,%f22
970      st      %f22,[%o2]

972      andcc   %13,2,%g0
973      fmovdnz %icc,%f28,%f30
974      st      %f30,[%o3]

976      addcc   %i0,-1,%i0
977      bg,pt   %icc,.loop0
978 ! delay slot

```

```

979      st      %f7,[%o0+4]

981      ba,pt   %icc,.end
982 ! delay slot
983      nop

985      .align  16
986 .case2:
987      fmuld   %f26,%f26,%f24
988      andcc   %13,1,%g0
989      bz,pn   %icc,.case3
990 ! delay slot
991      fxor    %f30,%f38,%f38

993      fmuld   %f16,pp3,%f22      ! sin(x2)

995      fmuld   %f0,qq3,%f6      ! cos(x0)

997      fmuld   %f8,qq3,%f14     ! cos(x1)

999      faddd   %f22,pp2,%f22
1000     fmuld   %f16,qq2,%f20

1002     fmuld   %f24,qq3,%f30     ! cos(x3)

1004     faddd   %f6,qq2,%f6
1005     fmuld   %f0,pp2,%f4

1007     faddd   %f14,qq2,%f14
1008     fmuld   %f8,pp2,%f12

1010     fmuld   %f16,%f22,%f22
1011     faddd   %f20,qq1,%f20

1013     faddd   %f30,qq2,%f30
1014     fmuld   %f24,pp2,%f28

1016     fmuld   %f0,%f6,%f6
1017     faddd   %f4,pp1,%f4

1019     fmuld   %f8,%f14,%f14
1020     faddd   %f12,pp1,%f12

1022     faddd   %f22,pp1,%f22
1023     fmuld   %f16,%f20,%f20
1024     add     %16,%g1,%16

1026     fmuld   %f24,%f30,%f30
1027     faddd   %f28,pp1,%f28

1029     faddd   %f6,qq1,%f6
1030     fmuld   %f0,%f4,%f4
1031     add     %14,%g1,%14

1033     faddd   %f14,qq1,%f14
1034     fmuld   %f8,%f12,%f12
1035     add     %15,%g1,%15

1037     fmuld   %f16,%f22,%f22

1039     faddd   %f30,qq1,%f30
1040     fmuld   %f24,%f28,%f28
1041     add     %17,%g1,%17

1043     fmuld   %f2,%f4,%f4

```

```

1045      fmuld    %f10,%f12,%f12
1047      fmuld    %f18,%f22,%f22
1048      ldd      [%16+8],%f16
1050      fmuld    %f26,%f28,%f28
1052      fmuld    %f0,%f6,%f6
1053      faddd    %f4,%f32,%f4
1054      ldd      [%14+16],%f0
1056      fmuld    %f8,%f14,%f14
1057      faddd    %f12,%f34,%f12
1058      ldd      [%15+16],%f8
1060      fmuld    %f16,%f20,%f20
1061      faddd    %f36,%f22,%f22
1063      fmuld    %f24,%f30,%f30
1064      faddd    %f28,%f38,%f28
1065      ldd      [%17+16],%f24
1067      fmuld    %f0,%f6,%f6
1068      faddd    %f4,%f2,%f4
1069      ldd      [%14+8],%f32
1071      fmuld    %f8,%f14,%f14
1072      faddd    %f12,%f10,%f12
1073      ldd      [%15+8],%f34
1075      faddd    %f18,%f22,%f22
1076      ldd      [%16+16],%f36
1078      fmuld    %f24,%f30,%f30
1079      faddd    %f28,%f26,%f28
1080      ldd      [%17+8],%f38
1082      fmuld    %f32,%f4,%f4
1084      fmuld    %f34,%f12,%f12
1086      fmuld    %f36,%f22,%f22
1088      fmuld    %f38,%f28,%f28
1090      fsubd    %f6,%f4,%f6
1092      fsubd    %f14,%f12,%f14
1094      faddd    %f22,%f20,%f22
1096      fsubd    %f30,%f28,%f30
1098      faddd    %f6,%f0,%f6
1100      faddd    %f14,%f8,%f14
1102      faddd    %f22,%f16,%f22
1104      faddd    %f30,%f24,%f30
1105      mov      %10,%14
1107      fnegd    %f6,%f4
1108      lda      [%11]%asi,%10      ! preload next argument
1110      fnegd    %f14,%f12

```

```

1111      lda      [%11]%asi,%f0
1113      fnegd    %f22,%f20
1114      lda      [%11+4]%asi,%f3
1116      fnegd    %f30,%f28
1117      andn     %10,%i5,%10
1118      add      %i1,%i2,%i1
1120      andcc    %14,2,%g0
1121      fmovdnz  %icc,%f4,%f6
1122      st       %f6,[%o0]
1124      andcc    %11,2,%g0
1125      fmovdnz  %icc,%f12,%f14
1126      st       %f14,[%o1]
1128      andcc    %12,2,%g0
1129      fmovdnz  %icc,%f20,%f22
1130      st       %f22,[%o2]
1132      andcc    %13,2,%g0
1133      fmovdnz  %icc,%f28,%f30
1134      st       %f30,[%o3]
1136      addcc    %i0,-1,%i0
1137      bg,pt    %icc,.loop0
1138      ! delay slot
1139      st       %f7,[%o0+4]
1141      ba,pt    %icc,.end
1142      ! delay slot
1143      nop
1145      .align   16
1146      .case3:
1147      fmuld    %f16,pp3,%f22      ! sin(x2)
1149      fmuld    %f24,pp3,%f30      ! sin(x3)
1151      fmuld    %f0,qq3,%f6        ! cos(x0)
1153      fmuld    %f8,qq3,%f14      ! cos(x1)
1155      faddd    %f22,pp2,%f22
1156      fmuld    %f16,qq2,%f20
1158      faddd    %f30,pp2,%f30
1159      fmuld    %f24,qq2,%f28
1161      faddd    %f6,qq2,%f6
1162      fmuld    %f0,pp2,%f4
1164      faddd    %f14,qq2,%f14
1165      fmuld    %f8,pp2,%f12
1167      fmuld    %f16,%f22,%f22
1168      faddd    %f20,qq1,%f20
1170      fmuld    %f24,%f30,%f30
1171      faddd    %f28,qq1,%f28
1173      fmuld    %f0,%f6,%f6
1174      faddd    %f4,pp1,%f4
1176      fmuld    %f8,%f14,%f14

```

```

1177      fadd    %f12,pp1,%f12
1179      fadd    %f22,pp1,%f22
1180      fmuld   %f16,%f20,%f20
1181      add     %16,%g1,%16
1183      fadd    %f30,pp1,%f30
1184      fmuld   %f24,%f28,%f28
1185      add     %17,%g1,%17
1187      fadd    %f6,qq1,%f6
1188      fmuld   %f0,%f4,%f4
1189      add     %14,%g1,%14
1191      fadd    %f14,qq1,%f14
1192      fmuld   %f8,%f12,%f12
1193      add     %15,%g1,%15
1195      fmuld   %f16,%f22,%f22
1197      fmuld   %f24,%f30,%f30
1199      fmuld   %f2,%f4,%f4
1201      fmuld   %f10,%f12,%f12
1203      fmuld   %f18,%f22,%f22
1204      ldd     [%16+8],%f16
1206      fmuld   %f26,%f30,%f30
1207      ldd     [%17+8],%f24
1209      fmuld   %f0,%f6,%f6
1210      fadd    %f4,%f32,%f4
1211      ldd     [%14+16],%f0
1213      fmuld   %f8,%f14,%f14
1214      fadd    %f12,%f34,%f12
1215      ldd     [%15+16],%f8
1217      fmuld   %f16,%f20,%f20
1218      fadd    %f36,%f22,%f22
1220      fmuld   %f24,%f28,%f28
1221      fadd    %f38,%f30,%f30
1223      fmuld   %f0,%f6,%f6
1224      fadd    %f4,%f2,%f4
1225      ldd     [%14+8],%f32
1227      fmuld   %f8,%f14,%f14
1228      fadd    %f12,%f10,%f12
1229      ldd     [%15+8],%f34
1231      fadd    %f18,%f22,%f22
1232      ldd     [%16+16],%f36
1234      fadd    %f26,%f30,%f30
1235      ldd     [%17+16],%f38
1237      fmuld   %f32,%f4,%f4
1239      fmuld   %f34,%f12,%f12
1241      fmuld   %f36,%f22,%f22

```

```

1243      fmuld   %f38,%f30,%f30
1245      fsubd   %f6,%f4,%f6
1247      fsubd   %f14,%f12,%f14
1249      fadd    %f22,%f20,%f22
1251      fadd    %f30,%f28,%f30
1253      fadd    %f6,%f0,%f6
1255      fadd    %f14,%f8,%f14
1257      fadd    %f22,%f16,%f22
1259      fadd    %f30,%f24,%f30
1260      mov     %10,%14
1262      fnegd   %f6,%f4
1263      lda     [%i1]%asi,%10      ! preload next argument
1265      fnegd   %f14,%f12
1266      lda     [%i1]%asi,%f0
1268      fnegd   %f22,%f20
1269      lda     [%i1+4]%asi,%f3
1271      fnegd   %f30,%f28
1272      andn    %10,%i5,%10
1273      add     %i1,%i2,%i1
1275      andcc   %14,2,%g0
1276      fmovdnz %icc,%f4,%f6
1277      st      %f6,[%o0]
1279      andcc   %11,2,%g0
1280      fmovdnz %icc,%f12,%f14
1281      st      %f14,[%o1]
1283      andcc   %12,2,%g0
1284      fmovdnz %icc,%f20,%f22
1285      st      %f22,[%o2]
1287      andcc   %13,2,%g0
1288      fmovdnz %icc,%f28,%f30
1289      st      %f30,[%o3]
1291      addcc   %i0,-1,%i0
1292      bg,pt   %icc,.loop0
1293      ! delay slot
1294      st      %f7,[%o0+4]
1296      ba,pt   %icc,.end
1297      ! delay slot
1298      nop
1300      .align  16
1301      .case4:
1302      fmuld   %f18,%f18,%f16
1303      andcc   %12,1,%g0
1304      bz,pn   %icc,.case6
1305      ! delay slot
1306      fxor   %f22,%f36,%f36
1308      fmuld   %f26,%f26,%f24

```

```

1309      andcc    %l3,1,%g0
1310      bz,pn    %icc,.case5
1311 ! delay slot
1312      fxor     %f30,%f38,%f38

1314      fmuld   %f8,pp3,%f14      ! sin(x1)
1316      fmuld   %f0,qq3,%f6      ! cos(x0)

1318      faddd   %f14,pp2,%f14
1319      fmuld   %f8,qq2,%f12

1321      fmuld   %f16,qq3,%f22    ! cos(x2)
1323      fmuld   %f24,qq3,%f30    ! cos(x3)

1325      faddd   %f6,qq2,%f6
1326      fmuld   %f0,pp2,%f4

1328      fmuld   %f8,%f14,%f14
1329      faddd   %f12,qq1,%f12

1331      faddd   %f22,qq2,%f22
1332      fmuld   %f16,pp2,%f20

1334      faddd   %f30,qq2,%f30
1335      fmuld   %f24,pp2,%f28

1337      fmuld   %f0,%f6,%f6
1338      faddd   %f4,pp1,%f4

1340      faddd   %f14,pp1,%f14
1341      fmuld   %f8,%f12,%f12
1342      add     %l5,%g1,%l5

1344      fmuld   %f16,%f22,%f22
1345      faddd   %f20,pp1,%f20

1347      fmuld   %f24,%f30,%f30
1348      faddd   %f28,pp1,%f28

1350      faddd   %f6,qq1,%f6
1351      fmuld   %f0,%f4,%f4
1352      add     %l4,%g1,%l4

1354      fmuld   %f8,%f14,%f14

1356      faddd   %f22,qq1,%f22
1357      fmuld   %f16,%f20,%f20
1358      add     %l6,%g1,%l6

1360      faddd   %f30,qq1,%f30
1361      fmuld   %f24,%f28,%f28
1362      add     %l7,%g1,%l7

1364      fmuld   %f2,%f4,%f4

1366      fmuld   %f10,%f14,%f14
1367      ldd     [%l5+8],%f8

1369      fmuld   %f18,%f20,%f20

1371      fmuld   %f26,%f28,%f28

1373      fmuld   %f0,%f6,%f6
1374      faddd   %f4,%f32,%f4

```

```

1375      ldd     [%l4+16],%f0

1377      fmuld   %f8,%f12,%f12
1378      faddd   %f34,%f14,%f14

1380      fmuld   %f16,%f22,%f22
1381      faddd   %f20,%f36,%f20
1382      ldd     [%l6+16],%f16

1384      fmuld   %f24,%f30,%f30
1385      faddd   %f28,%f38,%f28
1386      ldd     [%l7+16],%f24

1388      fmuld   %f0,%f6,%f6
1389      faddd   %f4,%f2,%f4
1390      ldd     [%l4+8],%f32

1392      faddd   %f10,%f14,%f14
1393      ldd     [%l5+16],%f34

1395      fmuld   %f16,%f22,%f22
1396      faddd   %f20,%f18,%f20
1397      ldd     [%l6+8],%f36

1399      fmuld   %f24,%f30,%f30
1400      faddd   %f28,%f26,%f28
1401      ldd     [%l7+8],%f38

1403      fmuld   %f32,%f4,%f4

1405      fmuld   %f34,%f14,%f14

1407      fmuld   %f36,%f20,%f20

1409      fmuld   %f38,%f28,%f28

1411      fsubd   %f6,%f4,%f6

1413      faddd   %f14,%f12,%f14

1415      fsubd   %f22,%f20,%f22

1417      fsubd   %f30,%f28,%f30

1419      faddd   %f6,%f0,%f6

1421      faddd   %f14,%f8,%f14

1423      faddd   %f22,%f16,%f22

1425      faddd   %f30,%f24,%f30
1426      mov     %l0,%l4

1428      fnegd   %f6,%f4
1429      lda     [%l1]%asi,%l0      ! preload next argument

1431      fnegd   %f14,%f12
1432      lda     [%l1]%asi,%f0

1434      fnegd   %f22,%f20
1435      lda     [%l1+4]%asi,%f3

1437      fnegd   %f30,%f28
1438      andn    %l0,%i5,%l0
1439      add     %i1,%i2,%i1

```



```

1441      andcc    %l4,2,%g0
1442      fmovdnz  %icc,%f4,%f6
1443      st       %f6,[%o0]

1445      andcc    %l1,2,%g0
1446      fmovdnz  %icc,%f12,%f14
1447      st       %f14,[%o1]

1449      andcc    %l2,2,%g0
1450      fmovdnz  %icc,%f20,%f22
1451      st       %f22,[%o2]

1453      andcc    %l3,2,%g0
1454      fmovdnz  %icc,%f28,%f30
1455      st       %f30,[%o3]

1457      addcc    %i0,-1,%i0
1458      bg,pt    %icc,.loop0
1459 ! delay    slot
1460      st       %f7,[%o0+4]

1462      ba,pt    %icc,.end
1463 ! delay    slot
1464      nop

1466      .align   16
1467 .case5:
1468      fmuld    %f8,pp3,%f14          ! sin(x1)

1470      fmuld    %f24,pp3,%f30        ! sin(x3)

1472      fmuld    %f0,qq3,%f6          ! cos(x0)

1474      faddd    %f14,pp2,%f14
1475      fmuld    %f8,qq2,%f12

1477      fmuld    %f16,qq3,%f22        ! cos(x2)

1479      faddd    %f30,pp2,%f30
1480      fmuld    %f24,qq2,%f28

1482      faddd    %f6,qq2,%f6
1483      fmuld    %f0,pp2,%f4

1485      fmuld    %f8,%f14,%f14
1486      faddd    %f12,qq1,%f12

1488      faddd    %f22,qq2,%f22
1489      fmuld    %f16,pp2,%f20

1491      fmuld    %f24,%f30,%f30
1492      faddd    %f28,qq1,%f28

1494      fmuld    %f0,%f6,%f6
1495      faddd    %f4,pp1,%f4

1497      faddd    %f14,pp1,%f14
1498      fmuld    %f8,%f12,%f12
1499      add      %l5,%g1,%l5

1501      fmuld    %f16,%f22,%f22
1502      faddd    %f20,pp1,%f20

1504      faddd    %f30,pp1,%f30
1505      fmuld    %f24,%f28,%f28
1506      add      %l7,%g1,%l7

```

```

1508      faddd    %f6,qq1,%f6
1509      fmuld    %f0,%f4,%f4
1510      add      %l4,%g1,%l4

1512      fmuld    %f8,%f14,%f14

1514      faddd    %f22,qq1,%f22
1515      fmuld    %f16,%f20,%f20
1516      add      %l6,%g1,%l6

1518      fmuld    %f24,%f30,%f30

1520      fmuld    %f2,%f4,%f4

1522      fmuld    %f10,%f14,%f14
1523      ldd      [%l5+8],%f8

1525      fmuld    %f18,%f20,%f20

1527      fmuld    %f26,%f30,%f30
1528      ldd      [%l7+8],%f24

1530      fmuld    %f0,%f6,%f6
1531      faddd    %f4,%f32,%f4
1532      ldd      [%l4+16],%f0

1534      fmuld    %f8,%f12,%f12
1535      faddd    %f34,%f14,%f14

1537      fmuld    %f16,%f22,%f22
1538      faddd    %f20,%f36,%f20
1539      ldd      [%l6+16],%f16

1541      fmuld    %f24,%f28,%f28
1542      faddd    %f38,%f30,%f30

1544      fmuld    %f0,%f6,%f6
1545      faddd    %f4,%f2,%f4
1546      ldd      [%l4+8],%f32

1548      faddd    %f10,%f14,%f14
1549      ldd      [%l5+16],%f34

1551      fmuld    %f16,%f22,%f22
1552      faddd    %f20,%f18,%f20
1553      ldd      [%l6+8],%f36

1555      faddd    %f26,%f30,%f30
1556      ldd      [%l7+16],%f38

1558      fmuld    %f32,%f4,%f4

1560      fmuld    %f34,%f14,%f14

1562      fmuld    %f36,%f20,%f20

1564      fmuld    %f38,%f30,%f30

1566      fsubd    %f6,%f4,%f6

1568      faddd    %f14,%f12,%f14

1570      fsubd    %f22,%f20,%f22

1572      faddd    %f30,%f28,%f30

```

```

1574      fadd    %f6,%f0,%f6
1576      fadd    %f14,%f8,%f14
1578      fadd    %f22,%f16,%f22
1580      fadd    %f30,%f24,%f30
1581      mov     %l0,%l4
1583      fnegd   %f6,%f4
1584      lda     [%i1]%asi,%l0      ! preload next argument
1586      fnegd   %f14,%f12
1587      lda     [%i1]%asi,%f0
1589      fnegd   %f22,%f20
1590      lda     [%i1+4]%asi,%f3
1592      fnegd   %f30,%f28
1593      andn    %l0,%i5,%l0
1594      add     %i1,%i2,%i1
1596      andcc   %l4,2,%g0
1597      fmovdnz %icc,%f4,%f6
1598      st      %f6,[%o0]
1600      andcc   %l1,2,%g0
1601      fmovdnz %icc,%f12,%f14
1602      st      %f14,[%o1]
1604      andcc   %l2,2,%g0
1605      fmovdnz %icc,%f20,%f22
1606      st      %f22,[%o2]
1608      andcc   %l3,2,%g0
1609      fmovdnz %icc,%f28,%f30
1610      st      %f30,[%o3]
1612      addcc   %i0,-1,%i0
1613      bg,pt   %icc,.loop0
1614 ! delay   slot
1615      st      %f7,[%o0+4]
1617      ba,pt   %icc,.end
1618 ! delay   slot
1619      nop
1621      .align  16
1622 .case6:
1623      fmuld   %f26,%f26,%f24
1624      andcc   %l3,1,%g0
1625      bz,pn   %icc,.case7
1626 ! delay   slot
1627      fxor   %f30,%f38,%f38
1629      fmuld   %f8,pp3,%f14      ! sin(x1)
1631      fmuld   %f16,pp3,%f22      ! sin(x2)
1633      fmuld   %f0,qq3,%f6      ! cos(x0)
1635      fadd    %f14,pp2,%f14
1636      fmuld   %f8,qq2,%f12
1638      fadd    %f22,pp2,%f22

```

```

1639      fmuld   %f16,qq2,%f20
1641      fmuld   %f24,qq3,%f30      ! cos(x3)
1643      fadd    %f6,qq2,%f6
1644      fmuld   %f0,pp2,%f4
1646      fmuld   %f8,%f14,%f14
1647      fadd    %f12,qq1,%f12
1649      fmuld   %f16,%f22,%f22
1650      fadd    %f20,qq1,%f20
1652      fadd    %f30,qq2,%f30
1653      fmuld   %f24,pp2,%f28
1655      fmuld   %f0,%f6,%f6
1656      fadd    %f4,pp1,%f4
1658      fadd    %f14,pp1,%f14
1659      fmuld   %f8,%f12,%f12
1660      add     %l5,%g1,%l5
1662      fadd    %f22,pp1,%f22
1663      fmuld   %f16,%f20,%f20
1664      add     %l6,%g1,%l6
1666      fmuld   %f24,%f30,%f30
1667      fadd    %f28,pp1,%f28
1669      fadd    %f6,qq1,%f6
1670      fmuld   %f0,%f4,%f4
1671      add     %l4,%g1,%l4
1673      fmuld   %f8,%f14,%f14
1675      fmuld   %f16,%f22,%f22
1677      fadd    %f30,qq1,%f30
1678      fmuld   %f24,%f28,%f28
1679      add     %l7,%g1,%l7
1681      fmuld   %f2,%f4,%f4
1683      fmuld   %f10,%f14,%f14
1684      ldd     [%l5+8],%f8
1686      fmuld   %f18,%f22,%f22
1687      ldd     [%l6+8],%f16
1689      fmuld   %f26,%f28,%f28
1691      fmuld   %f0,%f6,%f6
1692      fadd    %f4,%f32,%f4
1693      ldd     [%l4+16],%f0
1695      fmuld   %f8,%f12,%f12
1696      fadd    %f34,%f14,%f14
1698      fmuld   %f16,%f20,%f20
1699      fadd    %f36,%f22,%f22
1701      fmuld   %f24,%f30,%f30
1702      fadd    %f28,%f38,%f28
1703      ldd     [%l7+16],%f24

```

```

1705      fmuld   %f0,%f6,%f6
1706      faddd   %f4,%f2,%f4
1707      ldd     [%14+8],%f32

1709      faddd   %f10,%f14,%f14
1710      ldd     [%15+16],%f34

1712      faddd   %f18,%f22,%f22
1713      ldd     [%16+16],%f36

1715      fmuld   %f24,%f30,%f30
1716      faddd   %f28,%f26,%f28
1717      ldd     [%17+8],%f38

1719      fmuld   %f32,%f4,%f4

1721      fmuld   %f34,%f14,%f14

1723      fmuld   %f36,%f22,%f22

1725      fmuld   %f38,%f28,%f28

1727      fsubd   %f6,%f4,%f6

1729      faddd   %f14,%f12,%f14

1731      faddd   %f22,%f20,%f22

1733      fsubd   %f30,%f28,%f30

1735      faddd   %f6,%f0,%f6

1737      faddd   %f14,%f8,%f14

1739      faddd   %f22,%f16,%f22

1741      faddd   %f30,%f24,%f30
1742      mov     %10,%14

1744      fnegd   %f6,%f4
1745      lda     [%i1]%asi,%10          ! preload next argument

1747      fnegd   %f14,%f12
1748      lda     [%i1]%asi,%f0

1750      fnegd   %f22,%f20
1751      lda     [%i1+4]%asi,%f3

1753      fnegd   %f30,%f28
1754      andn    %10,%i5,%10
1755      add     %i1,%i2,%i1

1757      andcc   %14,2,%g0
1758      fmovdnz %icc,%f4,%f6
1759      st      %f6,[%o0]

1761      andcc   %11,2,%g0
1762      fmovdnz %icc,%f12,%f14
1763      st      %f14,[%o1]

1765      andcc   %12,2,%g0
1766      fmovdnz %icc,%f20,%f22
1767      st      %f22,[%o2]

1769      andcc   %13,2,%g0
1770      fmovdnz %icc,%f28,%f30

```

```

1771      st      %f30,[%o3]

1773      addcc   %i0,-1,%i0
1774      bg,pt   %icc,.loop0
1775 ! delay slot
1776      st      %f7,[%o0+4]

1778      ba,pt   %icc,.end
1779 ! delay slot
1780      nop

1782      .align  16
1783 .case7:
1784      fmuld   %f8,pp3,%f14          ! sin(x1)

1786      fmuld   %f16,pp3,%f22        ! sin(x2)

1788      fmuld   %f24,pp3,%f30        ! sin(x3)

1790      fmuld   %f0,qq3,%f6          ! cos(x0)

1792      faddd   %f14,pp2,%f14
1793      fmuld   %f8,qq2,%f12

1795      faddd   %f22,pp2,%f22
1796      fmuld   %f16,qq2,%f20

1798      faddd   %f30,pp2,%f30
1799      fmuld   %f24,qq2,%f28

1801      faddd   %f6,qq2,%f6
1802      fmuld   %f0,pp2,%f4

1804      fmuld   %f8,%f14,%f14
1805      faddd   %f12,qq1,%f12

1807      fmuld   %f16,%f22,%f22
1808      faddd   %f20,qq1,%f20

1810      fmuld   %f24,%f30,%f30
1811      faddd   %f28,qq1,%f28

1813      fmuld   %f0,%f6,%f6
1814      faddd   %f4,pp1,%f4

1816      faddd   %f14,pp1,%f14
1817      fmuld   %f8,%f12,%f12
1818      add     %15,%g1,%15

1820      faddd   %f22,pp1,%f22
1821      fmuld   %f16,%f20,%f20
1822      add     %16,%g1,%16

1824      faddd   %f30,pp1,%f30
1825      fmuld   %f24,%f28,%f28
1826      add     %17,%g1,%17

1828      faddd   %f6,qq1,%f6
1829      fmuld   %f0,%f4,%f4
1830      add     %14,%g1,%14

1832      fmuld   %f8,%f14,%f14

1834      fmuld   %f16,%f22,%f22

1836      fmuld   %f24,%f30,%f30

```

```

1838      fmuld    %f2,%f4,%f4
1840      fmuld    %f10,%f14,%f14
1841      ldd      [%15+8],%f8
1843      fmuld    %f18,%f22,%f22
1844      ldd      [%16+8],%f16
1846      fmuld    %f26,%f30,%f30
1847      ldd      [%17+8],%f24
1849      fmuld    %f0,%f6,%f6
1850      faddd    %f4,%f32,%f4
1851      ldd      [%14+16],%f0
1853      fmuld    %f8,%f12,%f12
1854      faddd    %f34,%f14,%f14
1856      fmuld    %f16,%f20,%f20
1857      faddd    %f36,%f22,%f22
1859      fmuld    %f24,%f28,%f28
1860      faddd    %f38,%f30,%f30
1862      fmuld    %f0,%f6,%f6
1863      faddd    %f4,%f2,%f4
1864      ldd      [%14+8],%f32
1866      faddd    %f10,%f14,%f14
1867      ldd      [%15+16],%f34
1869      faddd    %f18,%f22,%f22
1870      ldd      [%16+16],%f36
1872      faddd    %f26,%f30,%f30
1873      ldd      [%17+16],%f38
1875      fmuld    %f32,%f4,%f4
1877      fmuld    %f34,%f14,%f14
1879      fmuld    %f36,%f22,%f22
1881      fmuld    %f38,%f30,%f30
1883      fsubd    %f6,%f4,%f6
1885      faddd    %f14,%f12,%f14
1887      faddd    %f22,%f20,%f22
1889      faddd    %f30,%f28,%f30
1891      faddd    %f6,%f0,%f6
1893      faddd    %f14,%f8,%f14
1895      faddd    %f22,%f16,%f22
1897      faddd    %f30,%f24,%f30
1898      mov      %10,%14
1900      fnegd    %f6,%f4
1901      lda      [%i1]asi,%10          ! preload next argument

```

```

1903      fnegd    %f14,%f12
1904      lda      [%i1]asi,%f0
1906      fnegd    %f22,%f20
1907      lda      [%i1+4]asi,%f3
1909      fnegd    %f30,%f28
1910      andn     %10,%i5,%10
1911      add      %i1,%i2,%i1
1913      andcc    %14,2,%g0
1914      fmovdnz   %icc,%f4,%f6
1915      st       %f6,[%o0]
1917      andcc    %11,2,%g0
1918      fmovdnz   %icc,%f12,%f14
1919      st       %f14,[%o1]
1921      andcc    %12,2,%g0
1922      fmovdnz   %icc,%f20,%f22
1923      st       %f22,[%o2]
1925      andcc    %13,2,%g0
1926      fmovdnz   %icc,%f28,%f30
1927      st       %f30,[%o3]
1929      addcc    %i0,-1,%i0
1930      bg,pt    %icc,.loop0
1931      ! delay slot
1932      st       %f7,[%o0+4]
1934      ba,pt    %icc,.end
1935      ! delay slot
1936      nop
1938      .align   16
1939      .case8:
1940      fmuld    %f10,%f10,%f8
1941      andcc    %11,1,%g0
1942      bz,pn    %icc,.case12
1943      ! delay slot
1944      fxor     %f14,%f34,%f34
1946      fmuld    %f18,%f18,%f16
1947      andcc    %12,1,%g0
1948      bz,pn    %icc,.case10
1949      ! delay slot
1950      fxor     %f22,%f36,%f36
1952      fmuld    %f26,%f26,%f24
1953      andcc    %13,1,%g0
1954      bz,pn    %icc,.case9
1955      ! delay slot
1956      fxor     %f30,%f38,%f38
1958      fmuld    %f0,pp3,%f6          ! sin(x0)
1960      faddd    %f6,pp2,%f6
1961      fmuld    %f0,qq2,%f4
1963      fmuld    %f8,qq3,%f14        ! cos(x1)
1965      fmuld    %f16,qq3,%f22      ! cos(x2)
1967      fmuld    %f24,qq3,%f30      ! cos(x3)

```

```

1969      fmuld   %f0,%f6,%f6
1970      faddd   %f4,qq1,%f4

1972      faddd   %f14,qq2,%f14
1973      fmuld   %f8,pp2,%f12

1975      faddd   %f22,qq2,%f22
1976      fmuld   %f16,pp2,%f20

1978      faddd   %f30,qq2,%f30
1979      fmuld   %f24,pp2,%f28

1981      faddd   %f6,pp1,%f6
1982      fmuld   %f0,%f4,%f4
1983      add     %l4,%g1,%l4

1985      fmuld   %f8,%f14,%f14
1986      faddd   %f12,pp1,%f12

1988      fmuld   %f16,%f22,%f22
1989      faddd   %f20,pp1,%f20

1991      fmuld   %f24,%f30,%f30
1992      faddd   %f28,pp1,%f28

1994      fmuld   %f0,%f6,%f6

1996      faddd   %f14,qq1,%f14
1997      fmuld   %f8,%f12,%f12
1998      add     %l5,%g1,%l5

2000      faddd   %f22,qq1,%f22
2001      fmuld   %f16,%f20,%f20
2002      add     %l6,%g1,%l6

2004      faddd   %f30,qq1,%f30
2005      fmuld   %f24,%f28,%f28
2006      add     %l7,%g1,%l7

2008      fmuld   %f2,%f6,%f6
2009      ldd    [%l4+8],%f0

2011      fmuld   %f10,%f12,%f12

2013      fmuld   %f18,%f20,%f20

2015      fmuld   %f26,%f28,%f28

2017      fmuld   %f0,%f4,%f4
2018      faddd   %f32,%f6,%f6

2020      fmuld   %f8,%f14,%f14
2021      faddd   %f12,%f34,%f12
2022      ldd    [%l5+16],%f8

2024      fmuld   %f16,%f22,%f22
2025      faddd   %f20,%f36,%f20
2026      ldd    [%l6+16],%f16

2028      fmuld   %f24,%f30,%f30
2029      faddd   %f28,%f38,%f28
2030      ldd    [%l7+16],%f24

2032      faddd   %f2,%f6,%f6
2033      ldd    [%l4+16],%f32

```

```

2035      fmuld   %f8,%f14,%f14
2036      faddd   %f12,%f10,%f12
2037      ldd    [%l5+8],%f34

2039      fmuld   %f16,%f22,%f22
2040      faddd   %f20,%f18,%f20
2041      ldd    [%l6+8],%f36

2043      fmuld   %f24,%f30,%f30
2044      faddd   %f28,%f26,%f28
2045      ldd    [%l7+8],%f38

2047      fmuld   %f32,%f6,%f6

2049      fmuld   %f34,%f12,%f12

2051      fmuld   %f36,%f20,%f20

2053      fmuld   %f38,%f28,%f28

2055      faddd   %f6,%f4,%f6

2057      fsubd   %f14,%f12,%f14

2059      fsubd   %f22,%f20,%f22

2061      fsubd   %f30,%f28,%f30

2063      faddd   %f6,%f0,%f6

2065      faddd   %f14,%f8,%f14

2067      faddd   %f22,%f16,%f22

2069      faddd   %f30,%f24,%f30
2070      mov     %l0,%l4

2072      fnegd   %f6,%f4
2073      lda     [%l1]%asi,%l0          ! preload next argument

2075      fnegd   %f14,%f12
2076      lda     [%l1]%asi,%f0

2078      fnegd   %f22,%f20
2079      lda     [%l1+4]%asi,%f3

2081      fnegd   %f30,%f28
2082      andn    %l0,%i5,%l0
2083      add     %l1,%i2,%l1

2085      andcc   %l4,2,%g0
2086      fmovdnz %icc,%f4,%f6
2087      st      %f6,[%o0]

2089      andcc   %l1,2,%g0
2090      fmovdnz %icc,%f12,%f14
2091      st      %f14,[%o1]

2093      andcc   %l2,2,%g0
2094      fmovdnz %icc,%f20,%f22
2095      st      %f22,[%o2]

2097      andcc   %l3,2,%g0
2098      fmovdnz %icc,%f28,%f30
2099      st      %f30,[%o3]

```

```

2101      addcc    %i0,-1,%i0
2102      bg,pt    %icc,.loop0
2103 ! delay slot
2104      st       %f7,[%o0+4]

2106      ba,pt    %icc,.end
2107 ! delay slot
2108      nop

2110      .align  16
2111 .case9:
2112      fmuld   %f0,pp3,%f6          ! sin(x0)

2114      fmuld   %f24,pp3,%f30       ! sin(x3)

2116      faddd   %f6,pp2,%f6
2117      fmuld   %f0,qq2,%f4

2119      fmuld   %f8,qq3,%f14       ! cos(x1)

2121      fmuld   %f16,qq3,%f22      ! cos(x2)

2123      faddd   %f30,pp2,%f30
2124      fmuld   %f24,qq2,%f28

2126      fmuld   %f0,%f6,%f6
2127      faddd   %f4,qq1,%f4

2129      faddd   %f14,qq2,%f14
2130      fmuld   %f8,pp2,%f12

2132      faddd   %f22,qq2,%f22
2133      fmuld   %f16,pp2,%f20

2135      fmuld   %f24,%f30,%f30
2136      faddd   %f28,qq1,%f28

2138      faddd   %f6,pp1,%f6
2139      fmuld   %f0,%f4,%f4
2140      add     %l4,%g1,%l4

2142      fmuld   %f8,%f14,%f14
2143      faddd   %f12,pp1,%f12

2145      fmuld   %f16,%f22,%f22
2146      faddd   %f20,pp1,%f20

2148      faddd   %f30,pp1,%f30
2149      fmuld   %f24,%f28,%f28
2150      add     %l17,%g1,%l17

2152      fmuld   %f0,%f6,%f6

2154      faddd   %f14,qq1,%f14
2155      fmuld   %f8,%f12,%f12
2156      add     %l15,%g1,%l15

2158      faddd   %f22,qq1,%f22
2159      fmuld   %f16,%f20,%f20
2160      add     %l16,%g1,%l16

2162      fmuld   %f24,%f30,%f30

2164      fmuld   %f2,%f6,%f6
2165      ldd    [%l14+8],%f0

```

```

2167      fmuld   %f10,%f12,%f12

2169      fmuld   %f18,%f20,%f20

2171      fmuld   %f26,%f30,%f30
2172      ldd    [%l17+8],%f24

2174      fmuld   %f0,%f4,%f4
2175      faddd   %f32,%f6,%f6

2177      fmuld   %f8,%f14,%f14
2178      faddd   %f12,%f34,%f12
2179      ldd    [%l15+16],%f8

2181      fmuld   %f16,%f22,%f22
2182      faddd   %f20,%f36,%f20
2183      ldd    [%l16+16],%f16

2185      fmuld   %f24,%f28,%f28
2186      faddd   %f38,%f30,%f30

2188      faddd   %f2,%f6,%f6
2189      ldd    [%l14+16],%f32

2191      fmuld   %f8,%f14,%f14
2192      faddd   %f12,%f10,%f12
2193      ldd    [%l15+8],%f34

2195      fmuld   %f16,%f22,%f22
2196      faddd   %f20,%f18,%f20
2197      ldd    [%l16+8],%f36

2199      faddd   %f26,%f30,%f30
2200      ldd    [%l17+16],%f38

2202      fmuld   %f32,%f6,%f6

2204      fmuld   %f34,%f12,%f12

2206      fmuld   %f36,%f20,%f20

2208      fmuld   %f38,%f30,%f30

2210      faddd   %f6,%f4,%f6

2212      fsubd   %f14,%f12,%f14

2214      fsubd   %f22,%f20,%f22

2216      faddd   %f30,%f28,%f30

2218      faddd   %f6,%f0,%f6

2220      faddd   %f14,%f8,%f14

2222      faddd   %f22,%f16,%f22

2224      faddd   %f30,%f24,%f30
2225      mov    %l10,%l14

2227      fnegd   %f6,%f4
2228      lda    [%l11]asi,%l10          ! preload next argument

2230      fnegd   %f14,%f12
2231      lda    [%l11]asi,%f0

```

```

2233      fnegd    %f22,%f20
2234      lda      [%i1+4]%asi,%f3

2236      fnegd    %f30,%f28
2237      andn    %i0,%i5,%i0
2238      add     %i1,%i2,%i1

2240      andcc   %i4,2,%g0
2241      fmovdnz %icc,%f4,%f6
2242      st      %f6,[%o0]

2244      andcc   %i1,2,%g0
2245      fmovdnz %icc,%f12,%f14
2246      st      %f14,[%o1]

2248      andcc   %i2,2,%g0
2249      fmovdnz %icc,%f20,%f22
2250      st      %f22,[%o2]

2252      andcc   %i3,2,%g0
2253      fmovdnz %icc,%f28,%f30
2254      st      %f30,[%o3]

2256      addcc  %i0,-1,%i0
2257      bg,pt  %icc,.loop0
2258 ! delay  slot
2259      st      %f7,[%o0+4]

2261      ba,pt  %icc,.end
2262 ! delay  slot
2263      nop

2265      .align 16
2266 .case10:
2267      fmuld  %f26,%f26,%f24
2268      andcc  %i3,1,%g0
2269      bz,pn  %icc,.case11
2270 ! delay  slot
2271      fxor   %f30,%f38,%f38

2273      fmuld  %f0,pp3,%f6          ! sin(x0)

2275      fmuld  %f16,pp3,%f22       ! sin(x2)

2277      faddd  %f6,pp2,%f6
2278      fmuld  %f0,qq2,%f4

2280      fmuld  %f8,qq3,%f14       ! cos(x1)

2282      faddd  %f22,pp2,%f22
2283      fmuld  %f16,qq2,%f20

2285      fmuld  %f24,qq3,%f30     ! cos(x3)

2287      fmuld  %f0,%f6,%f6
2288      faddd  %f4,qq1,%f4

2290      faddd  %f14,qq2,%f14
2291      fmuld  %f8,pp2,%f12

2293      fmuld  %f16,%f22,%f22
2294      faddd  %f20,qq1,%f20

2296      faddd  %f30,qq2,%f30
2297      fmuld  %f24,pp2,%f28

```

```

2299      faddd  %f6,pp1,%f6
2300      fmuld  %f0,%f4,%f4
2301      add     %i4,%g1,%i4

2303      fmuld  %f8,%f14,%f14
2304      faddd  %f12,pp1,%f12

2306      faddd  %f22,pp1,%f22
2307      fmuld  %f16,%f20,%f20
2308      add     %i6,%g1,%i6

2310      fmuld  %f24,%f30,%f30
2311      faddd  %f28,pp1,%f28

2313      fmuld  %f0,%f6,%f6

2315      faddd  %f14,qq1,%f14
2316      fmuld  %f8,%f12,%f12
2317      add     %i5,%g1,%i5

2319      fmuld  %f16,%f22,%f22

2321      faddd  %f30,qq1,%f30
2322      fmuld  %f24,%f28,%f28
2323      add     %i7,%g1,%i7

2325      fmuld  %f2,%f6,%f6
2326      ldd    [%i4+8],%f0

2328      fmuld  %f10,%f12,%f12

2330      fmuld  %f18,%f22,%f22
2331      ldd    [%i6+8],%f16

2333      fmuld  %f26,%f28,%f28

2335      fmuld  %f0,%f4,%f4
2336      faddd  %f32,%f6,%f6

2338      fmuld  %f8,%f14,%f14
2339      faddd  %f12,%f34,%f12
2340      ldd    [%i5+16],%f8

2342      fmuld  %f16,%f20,%f20
2343      faddd  %f36,%f22,%f22

2345      fmuld  %f24,%f30,%f30
2346      faddd  %f28,%f38,%f28
2347      ldd    [%i7+16],%f24

2349      faddd  %f2,%f6,%f6
2350      ldd    [%i4+16],%f32

2352      fmuld  %f8,%f14,%f14
2353      faddd  %f12,%f10,%f12
2354      ldd    [%i5+8],%f34

2356      faddd  %f18,%f22,%f22
2357      ldd    [%i6+16],%f36

2359      fmuld  %f24,%f30,%f30
2360      faddd  %f28,%f26,%f28
2361      ldd    [%i7+8],%f38

2363      fmuld  %f32,%f6,%f6

```

```

2365      fmuld   %f34,%f12,%f12
2367      fmuld   %f36,%f22,%f22
2369      fmuld   %f38,%f28,%f28
2371      faddd   %f6,%f4,%f6
2373      fsubd   %f14,%f12,%f14
2375      faddd   %f22,%f20,%f22
2377      fsubd   %f30,%f28,%f30
2379      faddd   %f6,%f0,%f6
2381      faddd   %f14,%f8,%f14
2383      faddd   %f22,%f16,%f22
2385      faddd   %f30,%f24,%f30
2386      mov     %l0,%l14
2388      fnegd   %f6,%f4
2389      lda     [%l1]%asi,%l0      ! preload next argument
2391      fnegd   %f14,%f12
2392      lda     [%l1]%asi,%f0
2394      fnegd   %f22,%f20
2395      lda     [%l1+4]asi,%f3
2397      fnegd   %f30,%f28
2398      andn   %l0,%i5,%l0
2399      add     %i1,%i2,%i1
2401      andcc   %l4,2,%g0
2402      fmovdnz %icc,%f4,%f6
2403      st      %f6,[%o0]
2405      andcc   %l1,2,%g0
2406      fmovdnz %icc,%f12,%f14
2407      st      %f14,[%o1]
2409      andcc   %l2,2,%g0
2410      fmovdnz %icc,%f20,%f22
2411      st      %f22,[%o2]
2413      andcc   %l3,2,%g0
2414      fmovdnz %icc,%f28,%f30
2415      st      %f30,[%o3]
2417      addcc   %i0,-1,%i0
2418      bg,pt  %icc,.loop0
2419 ! delay  slot
2420      st      %f7,[%o0+4]
2422      ba,pt  %icc,.end
2423 ! delay  slot
2424      nop
2426      .align 16
2427 .casell:
2428      fmuld   %f0,pp3,%f6      ! sin(x0)
2430      fmuld   %f16,pp3,%f22    ! sin(x2)

```

```

2432      fmuld   %f24,pp3,%f30      ! sin(x3)
2434      faddd   %f6,pp2,%f6
2435      fmuld   %f0,qq2,%f4
2437      fmuld   %f8,qq3,%f14      ! cos(x1)
2439      faddd   %f22,pp2,%f22
2440      fmuld   %f16,qq2,%f20
2442      faddd   %f30,pp2,%f30
2443      fmuld   %f24,qq2,%f28
2445      fmuld   %f0,%f6,%f6
2446      faddd   %f4,qq1,%f4
2448      faddd   %f14,qq2,%f14
2449      fmuld   %f8,pp2,%f12
2451      fmuld   %f16,%f22,%f22
2452      faddd   %f20,qq1,%f20
2454      fmuld   %f24,%f30,%f30
2455      faddd   %f28,qq1,%f28
2457      faddd   %f6,pp1,%f6
2458      fmuld   %f0,%f4,%f4
2459      add     %l4,%g1,%l4
2461      fmuld   %f8,%f14,%f14
2462      faddd   %f12,pp1,%f12
2464      faddd   %f22,pp1,%f22
2465      fmuld   %f16,%f20,%f20
2466      add     %l6,%g1,%l6
2468      faddd   %f30,pp1,%f30
2469      fmuld   %f24,%f28,%f28
2470      add     %l7,%g1,%l7
2472      fmuld   %f0,%f6,%f6
2474      faddd   %f14,qq1,%f14
2475      fmuld   %f8,%f12,%f12
2476      add     %l5,%g1,%l5
2478      fmuld   %f16,%f22,%f22
2480      fmuld   %f24,%f30,%f30
2482      fmuld   %f2,%f6,%f6
2483      ldd     [%l4+8],%f0
2485      fmuld   %f10,%f12,%f12
2487      fmuld   %f18,%f22,%f22
2488      ldd     [%l6+8],%f16
2490      fmuld   %f26,%f30,%f30
2491      ldd     [%l7+8],%f24
2493      fmuld   %f0,%f4,%f4
2494      faddd   %f32,%f6,%f6
2496      fmuld   %f8,%f14,%f14

```



```

2497      fadd    %f12,%f34,%f12
2498      ldd     [%15+16],%f8

2500      fmuld   %f16,%f20,%f20
2501      fadd    %f36,%f22,%f22

2503      fmuld   %f24,%f28,%f28
2504      fadd    %f38,%f30,%f30

2506      fadd    %f2,%f6,%f6
2507      ldd     [%14+16],%f32

2509      fmuld   %f8,%f14,%f14
2510      fadd    %f12,%f10,%f12
2511      ldd     [%15+8],%f34

2513      fadd    %f18,%f22,%f22
2514      ldd     [%16+16],%f36

2516      fadd    %f26,%f30,%f30
2517      ldd     [%17+16],%f38

2519      fmuld   %f32,%f6,%f6

2521      fmuld   %f34,%f12,%f12

2523      fmuld   %f36,%f22,%f22

2525      fmuld   %f38,%f30,%f30

2527      fadd    %f6,%f4,%f6

2529      fsubd   %f14,%f12,%f14

2531      fadd    %f22,%f20,%f22

2533      fadd    %f30,%f28,%f30

2535      fadd    %f6,%f0,%f6

2537      fadd    %f14,%f8,%f14

2539      fadd    %f22,%f16,%f22

2541      fadd    %f30,%f24,%f30
2542      mov     %10,%14

2544      fnegd   %f6,%f4
2545      lda     [%11]%asi,%10      ! preload next argument

2547      fnegd   %f14,%f12
2548      lda     [%11]%asi,%f0

2550      fnegd   %f22,%f20
2551      lda     [%11+4]%asi,%f3

2553      fnegd   %f30,%f28
2554      andn    %10,%i5,%10
2555      add     %i1,%i2,%i1

2557      andcc   %14,2,%g0
2558      fmovdnz %icc,%f4,%f6
2559      st      %f6,[%0]

2561      andcc   %11,2,%g0
2562      fmovdnz %icc,%f12,%f14

```

```

2563      st      %f14,[%0]

2565      andcc   %12,2,%g0
2566      fmovdnz %icc,%f20,%f22
2567      st      %f22,[%02]

2569      andcc   %13,2,%g0
2570      fmovdnz %icc,%f28,%f30
2571      st      %f30,[%03]

2573      addcc   %i0,-1,%i0
2574      bg,pt   %icc,.loop0
2575      ! delay slot
2576      st      %f7,[%0+4]

2578      ba,pt   %icc,.end
2579      ! delay slot
2580      nop

2582      .align  16
2583      .case12:
2584      fmuld   %f18,%f18,%f16
2585      andcc   %12,1,%g0
2586      bz,pn   %icc,.case14
2587      ! delay slot
2588      fxor    %f22,%f36,%f36

2590      fmuld   %f26,%f26,%f24
2591      andcc   %13,1,%g0
2592      bz,pn   %icc,.case13
2593      ! delay slot
2594      fxor    %f30,%f38,%f38

2596      fmuld   %f0,pp3,%f6      ! sin(x0)

2598      fmuld   %f8,pp3,%f14     ! sin(x1)

2600      fadd    %f6,pp2,%f6
2601      fmuld   %f0,qq2,%f4

2603      fadd    %f14,pp2,%f14
2604      fmuld   %f8,qq2,%f12

2606      fmuld   %f16,qq3,%f22   ! cos(x2)

2608      fmuld   %f24,qq3,%f30   ! cos(x3)

2610      fmuld   %f0,%f6,%f6
2611      fadd    %f4,qq1,%f4

2613      fmuld   %f8,%f14,%f14
2614      fadd    %f12,qq1,%f12

2616      fadd    %f22,qq2,%f22
2617      fmuld   %f16,pp2,%f20

2619      fadd    %f30,qq2,%f30
2620      fmuld   %f24,pp2,%f28

2622      fadd    %f6,pp1,%f6
2623      fmuld   %f0,%f4,%f4
2624      add     %14,%g1,%14

2626      fadd    %f14,pp1,%f14
2627      fmuld   %f8,%f12,%f12
2628      add     %15,%g1,%15

```

```

2630      fmuld   %f16,%f22,%f22
2631      fadddd  %f20,pp1,%f20

2633      fmuld   %f24,%f30,%f30
2634      fadddd  %f28,pp1,%f28

2636      fmuld   %f0,%f6,%f6

2638      fmuld   %f8,%f14,%f14

2640      fadddd  %f22,qq1,%f22
2641      fmuld   %f16,%f20,%f20
2642      add     %l6,%g1,%l6

2644      fadddd  %f30,qq1,%f30
2645      fmuld   %f24,%f28,%f28
2646      add     %l7,%g1,%l7

2648      fmuld   %f2,%f6,%f6
2649      ldd    [%l14+8],%f0

2651      fmuld   %f10,%f14,%f14
2652      ldd    [%l15+8],%f8

2654      fmuld   %f18,%f20,%f20

2656      fmuld   %f26,%f28,%f28

2658      fmuld   %f0,%f4,%f4
2659      fadddd  %f32,%f6,%f6

2661      fmuld   %f8,%f12,%f12
2662      fadddd  %f34,%f14,%f14

2664      fmuld   %f16,%f22,%f22
2665      fadddd  %f20,%f36,%f20
2666      ldd    [%l16+16],%f16

2668      fmuld   %f24,%f30,%f30
2669      fadddd  %f28,%f38,%f28
2670      ldd    [%l17+16],%f24

2672      fadddd  %f2,%f6,%f6
2673      ldd    [%l14+16],%f32

2675      fadddd  %f10,%f14,%f14
2676      ldd    [%l15+16],%f34

2678      fmuld   %f16,%f22,%f22
2679      fadddd  %f20,%f18,%f20
2680      ldd    [%l16+8],%f36

2682      fmuld   %f24,%f30,%f30
2683      fadddd  %f28,%f26,%f28
2684      ldd    [%l17+8],%f38

2686      fmuld   %f32,%f6,%f6

2688      fmuld   %f34,%f14,%f14

2690      fmuld   %f36,%f20,%f20

2692      fmuld   %f38,%f28,%f28

2694      fadddd  %f6,%f4,%f6

```

```

2696      fadddd  %f14,%f12,%f14

2698      fsubd   %f22,%f20,%f22

2700      fsubd   %f30,%f28,%f30

2702      fadddd  %f6,%f0,%f6

2704      fadddd  %f14,%f8,%f14

2706      fadddd  %f22,%f16,%f22

2708      fadddd  %f30,%f24,%f30
2709      mov     %l0,%l4

2711      fnegd   %f6,%f4
2712      lda     [%i1]%asi,%l0      ! preload next argument

2714      fnegd   %f14,%f12
2715      lda     [%i1]%asi,%f0

2717      fnegd   %f22,%f20
2718      lda     [%i1+4]%asi,%f3

2720      fnegd   %f30,%f28
2721      andn    %l0,%i5,%l0
2722      add     %i1,%i2,%i1

2724      andcc   %l4,2,%g0
2725      fmovdnz %icc,%f4,%f6
2726      st      %f6,[%o0]

2728      andcc   %l1,2,%g0
2729      fmovdnz %icc,%f12,%f14
2730      st      %f14,[%o1]

2732      andcc   %l2,2,%g0
2733      fmovdnz %icc,%f20,%f22
2734      st      %f22,[%o2]

2736      andcc   %l3,2,%g0
2737      fmovdnz %icc,%f28,%f30
2738      st      %f30,[%o3]

2740      addcc   %i0,-1,%i0
2741      bg,pt   %icc,.loop0
2742      ! delay slot
2743      st      %f7,[%o0+4]

2745      ba,pt   %icc,.end
2746      ! delay slot
2747      nop

2749      .align  16
2750      .case13:
2751      fmuld   %f0,pp3,%f6      ! sin(x0)

2753      fmuld   %f8,pp3,%f14    ! sin(x1)

2755      fmuld   %f24,pp3,%f30   ! sin(x3)

2757      fadddd  %f6,pp2,%f6
2758      fmuld   %f0,qq2,%f4

2760      fadddd  %f14,pp2,%f14

```

```

2761      fmuld    %f8,qq2,%f12
2763      fmuld    %f16,qq3,%f22          ! cos(x2)
2765      faddd    %f30,pp2,%f30
2766      fmuld    %f24,qq2,%f28
2768      fmuld    %f0,%f6,%f6
2769      faddd    %f4,qq1,%f4
2771      fmuld    %f8,%f14,%f14
2772      faddd    %f12,qq1,%f12
2774      faddd    %f22,qq2,%f22
2775      fmuld    %f16,pp2,%f20
2777      fmuld    %f24,%f30,%f30
2778      faddd    %f28,qq1,%f28
2780      faddd    %f6,pp1,%f6
2781      fmuld    %f0,%f4,%f4
2782      add      %l4,%g1,%l4
2784      faddd    %f14,pp1,%f14
2785      fmuld    %f8,%f12,%f12
2786      add      %l5,%g1,%l5
2788      fmuld    %f16,%f22,%f22
2789      faddd    %f20,pp1,%f20
2791      faddd    %f30,pp1,%f30
2792      fmuld    %f24,%f28,%f28
2793      add      %l7,%g1,%l7
2795      fmuld    %f0,%f6,%f6
2797      fmuld    %f8,%f14,%f14
2799      faddd    %f22,qq1,%f22
2800      fmuld    %f16,%f20,%f20
2801      add      %l6,%g1,%l6
2803      fmuld    %f24,%f30,%f30
2805      fmuld    %f2,%f6,%f6
2806      ldd     [%l4+8],%f0
2808      fmuld    %f10,%f14,%f14
2809      ldd     [%l5+8],%f8
2811      fmuld    %f18,%f20,%f20
2813      fmuld    %f26,%f30,%f30
2814      ldd     [%l7+8],%f24
2816      fmuld    %f0,%f4,%f4
2817      faddd    %f32,%f6,%f6
2819      fmuld    %f8,%f12,%f12
2820      faddd    %f34,%f14,%f14
2822      fmuld    %f16,%f22,%f22
2823      faddd    %f20,%f36,%f20
2824      ldd     [%l6+16],%f16
2826      fmuld    %f24,%f28,%f28

```

```

2827      faddd    %f38,%f30,%f30
2829      faddd    %f2,%f6,%f6
2830      ldd     [%l4+16],%f32
2832      faddd    %f10,%f14,%f14
2833      ldd     [%l5+16],%f34
2835      fmuld    %f16,%f22,%f22
2836      faddd    %f20,%f18,%f20
2837      ldd     [%l6+8],%f36
2839      faddd    %f26,%f30,%f30
2840      ldd     [%l7+16],%f38
2842      fmuld    %f32,%f6,%f6
2844      fmuld    %f34,%f14,%f14
2846      fmuld    %f36,%f20,%f20
2848      fmuld    %f38,%f30,%f30
2850      faddd    %f6,%f4,%f6
2852      faddd    %f14,%f12,%f14
2854      fsubd    %f22,%f20,%f22
2856      faddd    %f30,%f28,%f30
2858      faddd    %f6,%f0,%f6
2860      faddd    %f14,%f8,%f14
2862      faddd    %f22,%f16,%f22
2864      faddd    %f30,%f24,%f30
2865      mov     %l0,%l4
2867      fnegd    %f6,%f4
2868      lda     [%l1]%asi,%l0          ! preload next argument
2870      fnegd    %f14,%f12
2871      lda     [%l1]%asi,%f0
2873      fnegd    %f22,%f20
2874      lda     [%l1+4]%asi,%f3
2876      fnegd    %f30,%f28
2877      andn    %l0,%i5,%l0
2878      add     %l1,%i2,%l1
2880      andcc   %l4,2,%g0
2881      fmovdnz  %icc,%f4,%f6
2882      st      %f6,[%o0]
2884      andcc   %l1,2,%g0
2885      fmovdnz  %icc,%f12,%f14
2886      st      %f14,[%o1]
2888      andcc   %l2,2,%g0
2889      fmovdnz  %icc,%f20,%f22
2890      st      %f22,[%o2]
2892      andcc   %l3,2,%g0

```

```

2893      fmovdnz  %icc,%f28,%f30
2894      st      %f30,[%o3]

2896      addcc   %i0,-1,%i0
2897      bg,pt   %icc,.loop0
2898 ! delay slot
2899      st      %f7,[%o0+4]

2901      ba,pt   %icc,.end
2902 ! delay slot
2903      nop

2905      .align  16
2906 .case14:
2907      fmuld   %f26,%f26,%f24
2908      andcc   %l3,1,%g0
2909      bz,pn   %icc,.case15
2910 ! delay slot
2911      fxor    %f30,%f38,%f38

2913      fmuld   %f0,pp3,%f6          ! sin(x0)

2915      fmuld   %f8,pp3,%f14        ! sin(x1)

2917      fmuld   %f16,pp3,%f22       ! sin(x2)

2919      faddd   %f6,pp2,%f6
2920      fmuld   %f0,qq2,%f4

2922      faddd   %f14,pp2,%f14
2923      fmuld   %f8,qq2,%f12

2925      faddd   %f22,pp2,%f22
2926      fmuld   %f16,qq2,%f20

2928      fmuld   %f24,qq3,%f30      ! cos(x3)

2930      fmuld   %f0,%f6,%f6
2931      faddd   %f4,qq1,%f4

2933      fmuld   %f8,%f14,%f14
2934      faddd   %f12,qq1,%f12

2936      fmuld   %f16,%f22,%f22
2937      faddd   %f20,qq1,%f20

2939      faddd   %f30,qq2,%f30
2940      fmuld   %f24,pp2,%f28

2942      faddd   %f6,pp1,%f6
2943      fmuld   %f0,%f4,%f4
2944      add     %l4,%g1,%l4

2946      faddd   %f14,pp1,%f14
2947      fmuld   %f8,%f12,%f12
2948      add     %l5,%g1,%l5

2950      faddd   %f22,pp1,%f22
2951      fmuld   %f16,%f20,%f20
2952      add     %l6,%g1,%l6

2954      fmuld   %f24,%f30,%f30
2955      faddd   %f28,pp1,%f28

2957      fmuld   %f0,%f6,%f6

```

```

2959      fmuld   %f8,%f14,%f14

2961      fmuld   %f16,%f22,%f22

2963      faddd   %f30,qq1,%f30
2964      fmuld   %f24,%f28,%f28
2965      add     %l7,%g1,%l7

2967      fmuld   %f2,%f6,%f6
2968      ldd     [%l4+8],%f0

2970      fmuld   %f10,%f14,%f14
2971      ldd     [%l5+8],%f8

2973      fmuld   %f18,%f22,%f22
2974      ldd     [%l6+8],%f16

2976      fmuld   %f26,%f28,%f28

2978      fmuld   %f0,%f4,%f4
2979      faddd   %f32,%f6,%f6

2981      fmuld   %f8,%f12,%f12
2982      faddd   %f34,%f14,%f14

2984      fmuld   %f16,%f20,%f20
2985      faddd   %f36,%f22,%f22

2987      fmuld   %f24,%f30,%f30
2988      faddd   %f28,%f38,%f28
2989      ldd     [%l7+16],%f24

2991      faddd   %f2,%f6,%f6
2992      ldd     [%l4+16],%f32

2994      faddd   %f10,%f14,%f14
2995      ldd     [%l5+16],%f34

2997      faddd   %f18,%f22,%f22
2998      ldd     [%l6+16],%f36

3000      fmuld   %f24,%f30,%f30
3001      faddd   %f28,%f26,%f28
3002      ldd     [%l7+8],%f38

3004      fmuld   %f32,%f6,%f6

3006      fmuld   %f34,%f14,%f14

3008      fmuld   %f36,%f22,%f22

3010      fmuld   %f38,%f28,%f28

3012      faddd   %f6,%f4,%f6

3014      faddd   %f14,%f12,%f14

3016      faddd   %f22,%f20,%f22

3018      fsubd   %f30,%f28,%f30

3020      faddd   %f6,%f0,%f6

3022      faddd   %f14,%f8,%f14

3024      faddd   %f22,%f16,%f22

```

```

3026      fadd    %f30,%f24,%f30
3027      mov     %l0,%l14

3029      fnegd   %f6,%f4
3030      lda     [%i1]%asi,%l0      ! preload next argument

3032      fnegd   %f14,%f12
3033      lda     [%i1]%asi,%f0

3035      fnegd   %f22,%f20
3036      lda     [%i1+4]%asi,%f3

3038      fnegd   %f30,%f28
3039      andn    %l0,%i5,%l0
3040      add     %i1,%i2,%i1

3042      andcc   %l4,2,%g0
3043      fmovdnz %icc,%f4,%f6
3044      st      %f6,[%o0]

3046      andcc   %l1,2,%g0
3047      fmovdnz %icc,%f12,%f14
3048      st      %f14,[%o1]

3050      andcc   %l2,2,%g0
3051      fmovdnz %icc,%f20,%f22
3052      st      %f22,[%o2]

3054      andcc   %l3,2,%g0
3055      fmovdnz %icc,%f28,%f30
3056      st      %f30,[%o3]

3058      addcc   %i0,-1,%i0
3059      bg,pt   %icc,.loop0
3060 ! delay  slot
3061      st      %f7,[%o0+4]

3063      ba,pt   %icc,.end
3064 ! delay  slot
3065      nop

3067      .align  16
3068 .case15:
3069      fmuld   %f0,pp3,%f6      ! sin(x0)

3071      fmuld   %f8,pp3,%f14    ! sin(x1)

3073      fmuld   %f16,pp3,%f22   ! sin(x2)

3075      fmuld   %f24,pp3,%f30   ! sin(x3)

3077      fadd    %f6,pp2,%f6
3078      fmuld   %f0,qq2,%f4

3080      fadd    %f14,pp2,%f14
3081      fmuld   %f8,qq2,%f12

3083      fadd    %f22,pp2,%f22
3084      fmuld   %f16,qq2,%f20

3086      fadd    %f30,pp2,%f30
3087      fmuld   %f24,qq2,%f28

3089      fmuld   %f0,%f6,%f6
3090      fadd    %f4,qq1,%f4

```

```

3092      fmuld   %f8,%f14,%f14
3093      fadd    %f12,qq1,%f12

3095      fmuld   %f16,%f22,%f22
3096      fadd    %f20,qq1,%f20

3098      fmuld   %f24,%f30,%f30
3099      fadd    %f28,qq1,%f28

3101      fadd    %f6,pp1,%f6
3102      fmuld   %f0,%f4,%f4
3103      add     %l4,%g1,%l4

3105      fadd    %f14,pp1,%f14
3106      fmuld   %f8,%f12,%f12
3107      add     %l5,%g1,%l5

3109      fadd    %f22,pp1,%f22
3110      fmuld   %f16,%f20,%f20
3111      add     %l6,%g1,%l6

3113      fadd    %f30,pp1,%f30
3114      fmuld   %f24,%f28,%f28
3115      add     %l7,%g1,%l7

3117      fmuld   %f0,%f6,%f6

3119      fmuld   %f8,%f14,%f14

3121      fmuld   %f16,%f22,%f22

3123      fmuld   %f24,%f30,%f30

3125      fmuld   %f2,%f6,%f6
3126      ldd    [%l4+8],%f0

3128      fmuld   %f10,%f14,%f14
3129      ldd    [%l5+8],%f8

3131      fmuld   %f18,%f22,%f22
3132      ldd    [%l6+8],%f16

3134      fmuld   %f26,%f30,%f30
3135      ldd    [%l7+8],%f24

3137      fmuld   %f0,%f4,%f4
3138      fadd    %f32,%f6,%f6

3140      fmuld   %f8,%f12,%f12
3141      fadd    %f34,%f14,%f14

3143      fmuld   %f16,%f20,%f20
3144      fadd    %f36,%f22,%f22

3146      fmuld   %f24,%f28,%f28
3147      fadd    %f38,%f30,%f30

3149      fadd    %f2,%f6,%f6
3150      ldd    [%l4+16],%f32

3152      fadd    %f10,%f14,%f14
3153      ldd    [%l5+16],%f34

3155      fadd    %f18,%f22,%f22
3156      ldd    [%l6+16],%f36

```

```

3158      fadd    %f26,%f30,%f30
3159      ldd     [%17+16],%f38

3161      fmuld   %f32,%f6,%f6

3163      fmuld   %f34,%f14,%f14

3165      fmuld   %f36,%f22,%f22

3167      fmuld   %f38,%f30,%f30

3169      fadd    %f6,%f4,%f6

3171      fadd    %f14,%f12,%f14

3173      fadd    %f22,%f20,%f22

3175      fadd    %f30,%f28,%f30

3177      fadd    %f6,%f0,%f6

3179      fadd    %f14,%f8,%f14

3181      fadd    %f22,%f16,%f22

3183      fadd    %f30,%f24,%f30
3184      mov     %10,%14

3186      fnegd   %f6,%f4
3187      lda     [%i1]%asi,%10      ! preload next argument

3189      fnegd   %f14,%f12
3190      lda     [%i1]%asi,%f0

3192      fnegd   %f22,%f20
3193      lda     [%i1+4]%asi,%f3

3195      fnegd   %f30,%f28
3196      andn   %10,%i5,%10
3197      add     %i1,%i2,%i1

3199      andcc   %14,2,%g0
3200      fmovdnz %icc,%f4,%f6
3201      st      %f6, [%o0]

3203      andcc   %11,2,%g0
3204      fmovdnz %icc,%f12,%f14
3205      st      %f14, [%o1]

3207      andcc   %12,2,%g0
3208      fmovdnz %icc,%f20,%f22
3209      st      %f22, [%o2]

3211      andcc   %13,2,%g0
3212      fmovdnz %icc,%f28,%f30
3213      st      %f30, [%o3]

3215      addcc   %i0,-1,%i0
3216      bg,pt   %icc,.loop0
3217 ! delay slot
3218      st      %f7, [%o0+4]

3220      ba,pt   %icc,.end
3221 ! delay slot
3222      nop

```

```

3225      .align 16
3226 .end:
3227      st      %f15, [%o1+4]
3228      st      %f23, [%o2+4]
3229      st      %f31, [%o3+4]
3230      ld      [%fp+biguns],%i5
3231      tst     %i5                      ! check for huge arguments remaining
3232      be,pt   %icc,.exit
3233 ! delay slot
3234      nop
3235 #ifdef __sparcv9
3236      ldx     [%fp+xsave],%o1
3237      ldx     [%fp+ysave],%o3
3238 #else
3239      ld      [%fp+xsave],%o1
3240      ld      [%fp+ysave],%o3
3241 #endif
3242      ld      [%fp+nsave],%o0
3243      ld      [%fp+sxsave],%o2
3244      ld      [%fp+sysave],%o4
3245      sra     %o2,0,%o2                ! sign-extend for V9
3246      sra     %o4,0,%o4
3247      call   __vlibm_vcos_big_ultra3
3248      sra     %o5,0,%o5                ! delay slot

3250 .exit:
3251      ret
3252      restore

3255      .align 16
3256 .last1:
3257      fadd    %f2,c3two44,%f4
3258      st      %f15, [%o1+4]
3259 .last1_from_range1:
3260      mov     0,%11
3261      fzeros  %f8
3262      fzero   %f10
3263      add     %fp,junk,%o1
3264 .last2:
3265      fadd    %f10,c3two44,%f12
3266      st      %f23, [%o2+4]
3267 .last2_from_range2:
3268      mov     0,%12
3269      fzeros  %f16
3270      fzero   %f18
3271      add     %fp,junk,%o2
3272 .last3:
3273      fadd    %f18,c3two44,%f20
3274      st      %f31, [%o3+4]
3275      st      %f5, [%fp+nk0]
3276      st      %f13, [%fp+nk1]
3277 .last3_from_range3:
3278      mov     0,%13
3279      fzeros  %f24
3280      fzero   %f26
3281      ba,pt   %icc,.cont
3282 ! delay slot
3283      add     %fp,junk,%o3

3286      .align 16
3287 .range0:
3288      cmp     %10,%o4

```

```

3289     bl,pt    %icc,1f                ! hx < 0x3e400000
3290 ! delay slot, harmless if branch taken
3291     sethi   %hi(0x7ff00000),%o7
3292     cmp     %l0,%o7
3293     bl,a,pt %icc,2f                ! branch if finite
3294 ! delay slot, squashed if branch not taken
3295     st      %o4,[%fp+biguns]      ! set biguns
3296     fzero   %f0
3297     fmuld   %f2,%f0,%f2
3298     st      %f2,[%o0]
3299     ba,pt   %icc,2f
3300 ! delay slot
3301     st      %f3,[%o0+4]
3302 1:
3303     fdtoi   %f2,%f4                ! raise inexact if not zero
3304     sethi   %hi(0x3ff00000),%o7
3305     st      %o7,[%o0]
3306     st      %g0,[%o0+4]
3307 2:
3308     addcc   %i0,-1,%i0
3309     ble,pn  %icc,.end
3310 ! delay slot, harmless if branch taken
3311     add     %i3,%i4,%i3            ! y += stridey
3312     andn   %l1,%i5,%l0            ! hx &= ~0x80000000
3313     fmovs   %f8,%f0
3314     fmovs   %f11,%f3
3315     ba,pt   %icc,.loop0
3316 ! delay slot
3317     add     %i1,%i2,%i1            ! x += stridex

3320     .align 16
3321 .range1:
3322     cmp     %l1,%o4
3323     bl,pt   %icc,1f                ! hx < 0x3e400000
3324 ! delay slot, harmless if branch taken
3325     sethi   %hi(0x7ff00000),%o7
3326     cmp     %l1,%o7
3327     bl,a,pt %icc,2f                ! branch if finite
3328 ! delay slot, squashed if branch not taken
3329     st      %o4,[%fp+biguns]      ! set biguns
3330     fzero   %f8
3331     fmuld   %f10,%f8,%f10
3332     st      %f10,[%o1]
3333     ba,pt   %icc,2f
3334 ! delay slot
3335     st      %f11,[%o1+4]
3336 1:
3337     fdtoi   %f10,%f12             ! raise inexact if not zero
3338     sethi   %hi(0x3ff00000),%o7
3339     st      %o7,[%o1]
3340     st      %g0,[%o1+4]
3341 2:
3342     addcc   %i0,-1,%i0
3343     ble,pn  %icc,.last1_from_range1
3344 ! delay slot, harmless if branch taken
3345     add     %i3,%i4,%i3            ! y += stridey
3346     andn   %l2,%i5,%l1            ! hx &= ~0x80000000
3347     fmovs   %f16,%f8
3348     fmovs   %f19,%f11
3349     ba,pt   %icc,.loop1
3350 ! delay slot
3351     add     %i1,%i2,%i1            ! x += stridex

3354     .align 16

```

```

3355 .range2:
3356     cmp     %l2,%o4
3357     bl,pt   %icc,1f                ! hx < 0x3e400000
3358 ! delay slot, harmless if branch taken
3359     sethi   %hi(0x7ff00000),%o7
3360     cmp     %l2,%o7
3361     bl,a,pt %icc,2f                ! branch if finite
3362 ! delay slot, squashed if branch not taken
3363     st      %o4,[%fp+biguns]      ! set biguns
3364     fzero   %f16
3365     fmuld   %f18,%f16,%f18
3366     st      %f18,[%o2]
3367     ba,pt   %icc,2f
3368 ! delay slot
3369     st      %f19,[%o2+4]
3370 1:
3371     fdtoi   %f18,%f20             ! raise inexact if not zero
3372     sethi   %hi(0x3ff00000),%o7
3373     st      %o7,[%o2]
3374     st      %g0,[%o2+4]
3375 2:
3376     addcc   %i0,-1,%i0
3377     ble,pn  %icc,.last2_from_range2
3378 ! delay slot, harmless if branch taken
3379     add     %i3,%i4,%i3            ! y += stridey
3380     andn   %l3,%i5,%l2            ! hx &= ~0x80000000
3381     fmovs   %f24,%f16
3382     fmovs   %f27,%f19
3383     ba,pt   %icc,.loop2
3384 ! delay slot
3385     add     %i1,%i2,%i1            ! x += stridex

3388     .align 16
3389 .range3:
3390     cmp     %l3,%o4
3391     bl,pt   %icc,1f                ! hx < 0x3e400000
3392 ! delay slot, harmless if branch taken
3393     sethi   %hi(0x7ff00000),%o7
3394     cmp     %l3,%o7
3395     bl,a,pt %icc,2f                ! branch if finite
3396 ! delay slot, squashed if branch not taken
3397     st      %o4,[%fp+biguns]      ! set biguns
3398     fzero   %f24
3399     fmuld   %f26,%f24,%f26
3400     st      %f26,[%o3]
3401     ba,pt   %icc,2f
3402 ! delay slot
3403     st      %f27,[%o3+4]
3404 1:
3405     fdtoi   %f26,%f28             ! raise inexact if not zero
3406     sethi   %hi(0x3ff00000),%o7
3407     st      %o7,[%o3]
3408     st      %g0,[%o3+4]
3409 2:
3410     addcc   %i0,-1,%i0
3411     ble,pn  %icc,.last3_from_range3
3412 ! delay slot, harmless if branch taken
3413     add     %i3,%i4,%i3            ! y += stridey
3414     ld      [%i1],%l3
3415     ld      [%i1],%f24
3416     ld      [%i1+4],%f27
3417     andn   %l3,%i5,%l3            ! hx &= ~0x80000000
3418     ba,pt   %icc,.loop3
3419 ! delay slot
3420     add     %i1,%i2,%i1            ! x += stridex

```

new/usr/src/lib/libmvec/common/vis/__vcos_ultra3.S

53

3422 SET_SIZE(__vcos_ultra3)
_____unchanged_portion_omitted_

new/usr/src/lib/libmvec/common/vis/_vlog_ultra3.S

1

84349 Tue Nov 25 12:59:56 2014

new/usr/src/lib/libmvec/common/vis/_vlog_ultra3.S

5262 libm needs to be carefully undef'd

5268 libm doesn't need to hide symbols which are already local

Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>

Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>

Reviewed by: Gordon Ross <gwr@nexenta.com>

Approved by: Gordon Ross <gwr@nexenta.com>

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file      "_vlog_ultra3.S"

31 #include "libm.h"
32 #if defined(LIBMVEC_SO_BUILD)
33     .weak      __vlog
34     .type      __vlog, #function
35     __vlog = __vlog_ultra3
36 #endif

37 /*
38  * ELEVENBIT table and order 5 POLYNOMIAL no explicit correction t
39  */

40     RO_DATA
41     .align    64
42 !! this is a new 11 bit table.
43 TBL:
44     .word    0xbfd522ae,    0x0738a000
45     .word    0xbd2ebe70,    0x8164c759
46     .word    0xbfd5178d,    0x9ab55000
47     .word    0xbd35c153,    0x0fe963b3
48     .word    0xbfd50c6f,    0x1d11b000
49     .word    0xbd42f8ca,    0x40bec1ea
50     .word    0xbfd50152,    0x8dal1f000
51     .word    0xbd42cfac,    0x6d29f4d7
52     .word    0xbfd4f637,    0xebba9000
53     .word    0xbd401f53,    0x9a676da3
54     .word    0xbfd4eb1f,    0x36b07000
```

new/usr/src/lib/libmvec/common/vis/_vlog_ultra3.S

2

```
55     .word    0xbd184047,    0x46e5797b
56     .word    0xbfd4e008,    0x6dd8b000
57     .word    0xbd4594b6,    0xaf0ddc3c
58     .word    0xbfd4d4f3,    0x90890000
59     .word    0xbd19fd79,    0x3a9f1441
60     .word    0xbfd4c9e0,    0x9e172000
61     .word    0xbd4877dd,    0xb93d49d7
62     .word    0xbfd4becf,    0x95d97000
63     .word    0xbd422662,    0x6ffee2c8
64     .word    0xbfd4b3c0,    0x77267000
65     .word    0xbd4d3497,    0x2fdf5a8c
66     .word    0xbfd4a8b3,    0x41552000
67     .word    0xbd46127e,    0x3d0dc8d1
68     .word    0xbfd49da7,    0xf3bcc000
69     .word    0xbd307b33,    0x4daf4b9a
70     .word    0xbfd4929e,    0x8db4e000
71     .word    0xbd3b9056,    0x556c70de
72     .word    0xbfd48797,    0x0e958000
73     .word    0xbd3dc1b8,    0x465cf25f
74     .word    0xbfd47c91,    0x75b6f000
75     .word    0xbd05acd1,    0x7009e35b
76     .word    0xbfd4718d,    0xc271c000
77     .word    0xbd306c18,    0xfb4c14c5
78     .word    0xbfd4668b,    0xf41ef000
79     .word    0xbd432874,    0x4e9d2b85
80     .word    0xbfd45b8c,    0xa017d000
81     .word    0xbd4e26ed,    0xf182f57b
82     .word    0xbfd4508e,    0x03b61000
83     .word    0xbd40ef1c,    0x2579199c
84     .word    0xbfd44591,    0xe0539000
85     .word    0xbd4e916a,    0x76d6dc28
86     .word    0xbfd43a97,    0x9f4ac000
87     .word    0xbd23ee07,    0x6a81f88e
88     .word    0xbfd42f9f,    0x3ff62000
89     .word    0xbd390644,    0x0f7d3354
90     .word    0xbfd424a8,    0xc1b0c000
91     .word    0xbd2dc57c,    0x99ae2a25
92     .word    0xbfd419b4,    0x23d5e000
93     .word    0xbd418e43,    0x6ec90e0a
94     .word    0xbfd40ec1,    0x65c13000
95     .word    0xbd3f59a8,    0xa01757f6
96     .word    0xbfd403d0,    0x86cea000
97     .word    0xbd3e6ef5,    0x74487308
98     .word    0xbfd3f8e1,    0x865a8000
99     .word    0xbd26f338,    0x912773e3
100    .word    0xbfd3edf4,    0x63c16000
101    .word    0xbd407cc1,    0xeb4069e1
102    .word    0xbfd3e309,    0x1e604000
103    .word    0xbd43f634,    0xa2afb68d
104    .word    0xbfd3d81f,    0xb5946000
105    .word    0xbd4b74e0,    0xf558b217
106    .word    0xbfd3cd38,    0x28bb6000
107    .word    0xbd489faf,    0xb06c8342
108    .word    0xbfd3c252,    0x77333000
109    .word    0xbd183b54,    0xb606bd5c
110    .word    0xbfd3b76e,    0xa059f000
111    .word    0xbd47b5cf,    0x9912c7cb
112    .word    0xbfd3ac8c,    0xa38e5000
113    .word    0xbd48bd04,    0x10ff506d
114    .word    0xbfd3a1ac,    0x802f3000
115    .word    0xbd398ecf,    0x399abd8d
116    .word    0xbfd396ce,    0x359bb000
117    .word    0xbd4ea7c6,    0x3a99c99c
118    .word    0xbfd38bf1,    0xc3370000
119    .word    0xbd4ce9e9,    0x41e9516d
120    .word    0xbfd38117,    0x28564000
```

```

121 .word 0xbd496386, 0xdb17e3f5
122 .word 0xbfd3763e, 0x64645000
123 .word 0xbd318b1f, 0x291dcb56
124 .word 0xbfd36b67, 0x76be1000
125 .word 0xbd116ecd, 0xb0f177c8
126 .word 0xbfd36092, 0x5ec44000
127 .word 0xbd4eb929, 0xf344bbd1
128 .word 0xbfd355bf, 0x1bd82000
129 .word 0xbd491599, 0x1da6c3c6
130 .word 0xbfd34aed, 0xad5b1000
131 .word 0xbd3a2aac, 0xf2be1fdd
132 .word 0xbfd3401e, 0x12aec000
133 .word 0xbd4741c6, 0x5548eb71
134 .word 0xbfd33550, 0x4b355000
135 .word 0xbd446efc, 0x89cefce92
136 .word 0xbfd32a84, 0x56512000
137 .word 0xbd04f928, 0x139af5d6
138 .word 0xbfd31fba, 0x3364c000
139 .word 0xbd4a08d8, 0x6ce5a16e
140 .word 0xbfd314f1, 0xe1d35000
141 .word 0xbd49c761, 0x4b37b0d2
142 .word 0xbfd30a2b, 0x61001000
143 .word 0xbd4a53e9, 0x6290ef5b
144 .word 0xbfd2ff66, 0xb04ea000
145 .word 0xbd43a896, 0xd5f0c8e9
146 .word 0xbfd2f4a3, 0xcf22e000
147 .word 0xbd4b8693, 0xf85f2705
148 .word 0xbfd2e9e2, 0xbce12000
149 .word 0xbd24300c, 0x128d1dc2
150 .word 0xbfd2df23, 0x78edd000
151 .word 0xbce292b7, 0xcd95c595
152 .word 0xbfd2d466, 0x02adc000
153 .word 0xbd49dcbc, 0x88caaf9b
154 .word 0xbfd2c9aa, 0x59863000
155 .word 0xbd4a7f90, 0xe829d4d2
156 .word 0xbfd2bef0, 0x7cdc9000
157 .word 0xbd2a9cfa, 0x4a5004f4
158 .word 0xbfd2b438, 0x6c168000
159 .word 0xbd4e1827, 0x3a343630
160 .word 0xbfd2a982, 0x269a3000
161 .word 0xbd4b7e9c, 0x6aa35e8c
162 .word 0xbfd29ecd, 0xabcdf000
163 .word 0xbd44073b, 0x3bdc2243
164 .word 0xbfd2941a, 0xfbb186000
165 .word 0xbd46f79e, 0xa4678ebb
166 .word 0xbfd2896a, 0x13e08000
167 .word 0xbd3a8ed0, 0x27e16952
168 .word 0xbfd27eba, 0xf58d8000
169 .word 0xbd49399d, 0xfdd2d096
170 .word 0xbfd2740d, 0x9f870000
171 .word 0xbd45f660, 0x0b9a802a
172 .word 0xbfd26962, 0x1134d000
173 .word 0xbd4724f0, 0x77d6ecce
174 .word 0xbfd25eb8, 0x49ff2000
175 .word 0xbd310c25, 0x03f76b8e
176 .word 0xbfd25410, 0x494e5000
177 .word 0xbd3b1d7a, 0xc0ef77f2
178 .word 0xbfd2496a, 0x0e8b3000
179 .word 0xbd003238, 0x687cfe2e
180 .word 0xbfd23ec5, 0x991eb000
181 .word 0xbd44920d, 0xdbae8d6f
182 .word 0xbfd23422, 0xe8724000
183 .word 0xbd40708a, 0x931c895b
184 .word 0xbfd22981, 0xfbef7000
185 .word 0xbd42f5ef, 0x4fb53f93
186 .word 0xbfd21ee2, 0xd3003000

```

```

187 .word 0xbd40382e, 0x41be00e3
188 .word 0xbfd21445, 0x6d0eb000
189 .word 0xbd41a87d, 0xeba46baf
190 .word 0xbfd209a9, 0xc9857000
191 .word 0xbd45b053, 0x3ba9c94d
192 .word 0xbfd1ff0f, 0xe7cf4000
193 .word 0xbd3e9d5b, 0x513ff0c1
194 .word 0xbfd1f477, 0xc7573000
195 .word 0xbd26d6d4, 0x010d751a
196 .word 0xbfd1e9e1, 0x67889000
197 .word 0xbd43e8a8, 0x961ba4d1
198 .word 0xbfd1df4c, 0xc7cf2000
199 .word 0xbd30b43f, 0x0455f7e4
200 .word 0xbfd1d4b9, 0xe796c000
201 .word 0xbd222a66, 0x7c42e56d
202 .word 0xbfd1ca28, 0xc64ba000
203 .word 0xbd4ca760, 0xf7a15533
204 .word 0xbfd1bf99, 0x635a6000
205 .word 0xbd4729bb, 0x5451ef6e
206 .word 0xbfd1b50b, 0xbe2fc000
207 .word 0xbd38ecd7, 0x3263201f
208 .word 0xbfd1aa7f, 0xd638d000
209 .word 0xbd29f60a, 0x9616f7a0
210 .word 0xbfd19ff5, 0xaae2f000
211 .word 0xbce69fd9, 0x9ec05ba8
212 .word 0xbfd1956d, 0x3b9bc000
213 .word 0xbd27d2f7, 0x3ad1aa14
214 .word 0xbfd18ae6, 0x87d13000
215 .word 0xbd43a034, 0x64df39ff
216 .word 0xbfd18061, 0x8ef18000
217 .word 0xbd45be80, 0x1bc9638d
218 .word 0xbfd175de, 0x506b3000
219 .word 0xbd30c07c, 0x4da5752f
220 .word 0xbfd16b5c, 0xcbacf000
221 .word 0xbd46e6b3, 0x7de945a0
222 .word 0xbfd160dd, 0x0025e000
223 .word 0xbd4ba5c1, 0xc499684a
224 .word 0xbfd1565e, 0xed455000
225 .word 0xbd4f8629, 0x48125517
226 .word 0xbfd14be2, 0x927ae000
227 .word 0xbd49a817, 0xc85685e2
228 .word 0xbfd14167, 0xef367000
229 .word 0xbd3e0c07, 0x824daaf5
230 .word 0xbfd136ef, 0x02e82000
231 .word 0xbd4217d3, 0xe78d3ed8
232 .word 0xbfd12c77, 0xcd007000
233 .word 0xbd13b294, 0x8a11f797
234 .word 0xbfd12202, 0x4c000000
235 .word 0xbd38fdd9, 0x76fabda5
236 .word 0xbfd1178e, 0x8227e000
237 .word 0xbd31ef78, 0xce2d07f2
238 .word 0xbfd10d1c, 0x6c194000
239 .word 0xbd4cb3de, 0x00324ee4
240 .word 0xbfd102ac, 0x0a35c000
241 .word 0xbd483810, 0x88080a5e
242 .word 0xbfd0f83d, 0x5bef2000
243 .word 0xbd475fa0, 0x37a37ba8
244 .word 0xbfd0edd0, 0x60b78000
245 .word 0xbd0019b5, 0x2d8435f5
246 .word 0xbfd0e365, 0x18012000
247 .word 0xbd2a5943, 0x8bbdca93
248 .word 0xbfd0d8fb, 0x813eb000
249 .word 0xbd1ee8c8, 0x8753fa35
250 .word 0xbfd0ce93, 0x9be30000
251 .word 0xbd4e8266, 0xd788ddf1
252 .word 0xbfd0c42d, 0x67616000

```

```

253 .word 0xbd27188b, 0x163ceae9
254 .word 0xbfd0b9c8, 0xe32d1000
255 .word 0xbd42224e, 0x89208f94
256 .word 0xbfd0af66, 0x0eb9e000
257 .word 0xbd23c7c3, 0xf528d80a
258 .word 0xbfd0a504, 0xe97bb000
259 .word 0xbd303094, 0xe6690c44
260 .word 0xbfd09aa5, 0x72e6c000
261 .word 0xbd3b50a1, 0xe1734342
262 .word 0xbfd09047, 0xaa6f9000
263 .word 0xbd3f18e8, 0x3ce75c0e
264 .word 0xbfd085eb, 0xf8ae000
265 .word 0xbd3e5d51, 0x3f45fe7b
266 .word 0xbfd07b91, 0x21adb000
267 .word 0xbd4520ba, 0x8e9b8a72
268 .word 0xbfd07138, 0x604d5000
269 .word 0xbd40c4e6, 0xd8b76a75
270 .word 0xbfd066e1, 0x4adf4000
271 .word 0xbd47f6bb, 0x351a4a71
272 .word 0xbfd05c8b, 0xe0d96000
273 .word 0xbd2ad0f1, 0xc77ccb58
274 .word 0xbfd05238, 0x21b1a000
275 .word 0xbd4ec752, 0xd39776ce
276 .word 0xbfd047e6, 0x0cde8000
277 .word 0xbd2bdf1, 0x0d397f3c
278 .word 0xbfd03d95, 0xa1d67000
279 .word 0xbd3a1788, 0x0f236109
280 .word 0xbfd03346, 0xe0106000
281 .word 0xbcf89ff8, 0xa966395c
282 .word 0xbfd028f9, 0xc7035000
283 .word 0xbd483851, 0x858333c0
284 .word 0xbfd01eae, 0x5626c000
285 .word 0xbd3a43dc, 0xfade85ae
286 .word 0xbfd01464, 0x8cf23000
287 .word 0xbd4d082a, 0x567b45ed
288 .word 0xbfd00a1c, 0x6adda000
289 .word 0xbd31cd8d, 0x688b9e18
290 .word 0xbfcfffab, 0xddec23000
291 .word 0xbd236a1a, 0xdb4a75a4
292 .word 0xbfcfeb22, 0x33ea0000
293 .word 0xbd2f3418, 0xde00938b
294 .word 0xbfcfd69b, 0xd4240000
295 .word 0xbd3641a8, 0xf2ccc45
296 .word 0xbfcfc218, 0xbe620000
297 .word 0xbd34bba4, 0x6f1cf6a0
298 .word 0xbfcfad98, 0xf1965000
299 .word 0xbd16ee92, 0x73d7c2de
300 .word 0xbfcf991c, 0x6cb3b000
301 .word 0xbd1bcbec, 0xca0cdf30
302 .word 0xbfcf84a3, 0x2ead7000
303 .word 0xbd386af1, 0xd33d9e37
304 .word 0xbfcf702d, 0x36777000
305 .word 0xbd3bdf9a, 0xba663077
306 .word 0xbfcf5bba, 0x83060000
307 .word 0xbd341b25, 0x4a43da63
308 .word 0xbfcf474b, 0x134df000
309 .word 0xbd1146d8, 0x8821289
310 .word 0xbfcf32de, 0xe6448000
311 .word 0xbd2efb83, 0x625f1609
312 .word 0xbfcf1e75, 0xfadf9000
313 .word 0xbd37bcea, 0x6d13e04a
314 .word 0xbfcf0a10, 0x50157000
315 .word 0xbd3dad5f, 0x7347f55b
316 .word 0xbfcf5ad, 0xe4dcf000
317 .word 0xbd3fcbbd, 0xd53488e4
318 .word 0xbfcfee14e, 0xb82d6000

```

```

319 .word 0xbd39d172, 0x6f4de261
320 .word 0xbfccecf2, 0xc8fe9000
321 .word 0xbd104e71, 0x7062a6fe
322 .word 0xbfcceb89a, 0x1648b000
323 .word 0xbd32e26f, 0x74808b80
324 .word 0xbfccea444, 0x9f04a000
325 .word 0xbd35e916, 0x63732a36
326 .word 0xbfcce8ff2, 0x622ba000
327 .word 0xbd378e13, 0xd33981e5
328 .word 0xbfcce7ba3, 0x5eb77000
329 .word 0xbd3c5422, 0x3b90d937
330 .word 0xbfcce6757, 0x93a26000
331 .word 0xbd01dc8e, 0xc0554762
332 .word 0xbfcce530e, 0xffe71000
333 .word 0xbccc21227, 0x6041f430
334 .word 0xbfcce3ec9, 0xa280c000
335 .word 0xbd14bd96, 0x3fb80bff
336 .word 0xbfcce2a87, 0x7a6b2000
337 .word 0xbd382381, 0x7787081a
338 .word 0xbfcce1648, 0x86a27000
339 .word 0xbd36ce95, 0xba645527
340 .word 0xbfcce020c, 0xc6235000
341 .word 0xbd356a7f, 0xa92375ee
342 .word 0xbfcdeedd4, 0x37eae000
343 .word 0xbd3e0125, 0x53595898
344 .word 0xbfcdd99e, 0xda6d000
345 .word 0xbd2fa273, 0x2c71522a
346 .word 0xbfcddc56c, 0xae452000
347 .word 0xbd3eb37a, 0xa24e1817
348 .word 0xbfcddb13d, 0xb0d48000
349 .word 0xbd32806a, 0x847527e6
350 .word 0xbfcdd9d11, 0xe1a3f000
351 .word 0xbd19da04, 0xfa9fa4c6
352 .word 0xbfcdd8e9, 0x3fb2f000
353 .word 0xbd2141af, 0xfb96815e
354 .word 0xbfcdd74c3, 0xca018000
355 .word 0xbd393e4c, 0xfa17dce1
356 .word 0xbfcdd60a1, 0x7f903000
357 .word 0xbd24523f, 0x207be58e
358 .word 0xbfcdd4c82, 0x5f5fd000
359 .word 0xbd3e3f04, 0x21df291e
360 .word 0xbfcdd3866, 0x6871f000
361 .word 0xbd21935e, 0x98ed9a88
362 .word 0xbfcdd244d, 0x99c85000
363 .word 0xbd29cfb0, 0xc890770
364 .word 0xbfcdd1037, 0xf2655000
365 .word 0xbd3cf6b0, 0x31492124
366 .word 0xbfcddc25, 0x714bd000
367 .word 0xbd39fbd3, 0x34e03910
368 .word 0xbfcce816, 0x157f1000
369 .word 0xbd330faa, 0x2efb357e
370 .word 0xbfcdd409, 0xde02d000
371 .word 0xbd132115, 0x39f1dec5
372 .word 0xbfcce000, 0xc9db3000
373 .word 0xbd38a4a9, 0xe8aa1402
374 .word 0xbfccabfa, 0xd80d0000
375 .word 0xbd11e253, 0x70a10e3e
376 .word 0xbfcce97f8, 0x079d4000
377 .word 0xbd23b161, 0xa8c6e6c5
378 .word 0xbfcce83f8, 0x57919000
379 .word 0xbd358740, 0x00c94a0f
380 .word 0xbfcce6ffb, 0xc6f00000
381 .word 0xbd3ee138, 0xd3a69d43
382 .word 0xbfcce5c02, 0x54bf2000
383 .word 0xbd1d2f55, 0x73da163b
384 .word 0xbfcce480c, 0x0005c000

```

```

385 .word 0xbd39a294, 0xd5e44e76
386 .word 0xbfcc3418, 0xc7cb7000
387 .word 0xbd234b5d, 0xe46e0516
388 .word 0xbfcc2028, 0xab17f000
389 .word 0xbd3368f8, 0x8d51c29d
390 .word 0xbfcc0c3b, 0xa8f3a000
391 .word 0xbd3ac339, 0x48e7f56a
392 .word 0xbfcbf851, 0xc0675000
393 .word 0xbd257be3, 0x67ef56a7
394 .word 0xbfcb4e6a, 0xf07c2000
395 .word 0xbd350591, 0x910f505a
396 .word 0xbfcbd087, 0x383bd000
397 .word 0xbd315a1d, 0xd355f6a5
398 .word 0xbfcbba6, 0x96b07000
399 .word 0xbd3d0045, 0xea3f2624
400 .word 0xbfcb8c9, 0xae4a000
401 .word 0xbd3a32e7, 0xf44432da
402 .word 0xbfcb94ee, 0x93e36000
403 .word 0xbd2f2a06, 0xe2db48a3
404 .word 0xbfcb8117, 0x30b82000
405 .word 0xbd1e9068, 0x3b9cd768
406 .word 0xbfcb6d42, 0xe06ec000
407 .word 0xbd302afe, 0x254869ba
408 .word 0xbfcb5971, 0xa213a000
409 .word 0xbd39b50e, 0x83aa91df
410 .word 0xbfcb45a3, 0x74b39000
411 .word 0xbd3701df, 0x22138fc3
412 .word 0xbfcb31d8, 0x575bc000
413 .word 0xbd3c794e, 0x562a63cb
414 .word 0xbfcb1e10, 0x4919e000
415 .word 0xbd3fa006, 0x2597f33a
416 .word 0xbfcb0a4b, 0x48fc1000
417 .word 0xbd3368c69, 0x51e3338a
418 .word 0xbfcafb689, 0x5610d000
419 .word 0xbd375beb, 0xba042b64
420 .word 0xbfcae2ca, 0x6f672000
421 .word 0xbd37a8d5, 0xae54f550
422 .word 0xbfccacf0e, 0x940e7000
423 .word 0xbd2800e3, 0xa7e64e07
424 .word 0xbfcaabb55, 0xc3169000
425 .word 0xbd1d6694, 0xd43acc9f
426 .word 0xbfcaa79f, 0xfb8fc000
427 .word 0xbd3a8bf1, 0x1c0d8aaa
428 .word 0xbfca93ed, 0x3c8ad000
429 .word 0xbd33c6de, 0x57d4ef4c
430 .word 0xbfca803d, 0x8518d000
431 .word 0xbd3e09d1, 0x87f293cc
432 .word 0xbfca6c90, 0xd44b7000
433 .word 0xbce38901, 0xf909e74b
434 .word 0xbfca58e7, 0x29348000
435 .word 0xbd3e867d, 0x504551b1
436 .word 0xbfca4540, 0x82e6a000
437 .word 0xbd360a77, 0xc81f7171
438 .word 0xbfca319c, 0xe074a000
439 .word 0xbcbdb7dba, 0xe650d5b3
440 .word 0xbfca1dfc, 0x40f1b000
441 .word 0xbd2fc3e1, 0xff6190fe
442 .word 0xbfca0a5e, 0xa371a000
443 .word 0xbd322191, 0x988b2e31
444 .word 0xbfca9f6c4, 0x07089000
445 .word 0xbd29904d, 0x6865817a
446 .word 0xbfca9e32c, 0x6acb0000
447 .word 0xbd3e5e8d, 0xbcb0fb4ac
448 .word 0xbfca9cf97, 0xcdce0000
449 .word 0xbd3d862f, 0x10c414e3
450 .word 0xbfca9bc06, 0x2f26f000

```

```

451 .word 0xbd3874d8, 0x1809e6d5
452 .word 0xbfca9a877, 0x8deba000
453 .word 0xbd3470fa, 0x3efec390
454 .word 0xbfca994eb, 0xe9325000
455 .word 0xbd2a9c9d, 0x28bcbe25
456 .word 0xbfca98163, 0x4011a000
457 .word 0xbd34eadd, 0x9e9045e2
458 .word 0xbfca96ddd, 0x91a0b000
459 .word 0xbd32ac6b, 0x11cf6f2b
460 .word 0xbfca95a5a, 0xdcf70000
461 .word 0xbd07f228, 0x58a0ff6f
462 .word 0xbfca946db, 0x212c6000
463 .word 0xbd36cf76, 0x74ca02ba
464 .word 0xbfca9335e, 0x5d594000
465 .word 0xbd33115c, 0x3abd47da
466 .word 0xbfca91fe4, 0x90965000
467 .word 0xbd30369c, 0xf30a1c32
468 .word 0xbfca90c6d, 0xb9fcb000
469 .word 0xbd39b282, 0xa239ca0d
470 .word 0xbfca8ff9, 0xd8a60000
471 .word 0xbd2af16c, 0x8230ceca
472 .word 0xbfca8e588, 0xebac2000
473 .word 0xbd3b7d5c, 0xab2d1140
474 .word 0xbfca8d21a, 0xf2299000
475 .word 0xbd14d652, 0x74757226
476 .word 0xbfca8beaf, 0xeb38f000
477 .word 0xbd3d1855, 0x6aa2da66
478 .word 0xbfca8ab47, 0xd5f5a000
479 .word 0xbd187eb8, 0x505d468f
480 .word 0xbfca897e2, 0xb17b1000
481 .word 0xbd334a64, 0x63f9a0b1
482 .word 0xbfca88480, 0x7ce56000
483 .word 0xbd1c77ce, 0xf4a8712c
484 .word 0xbfca87121, 0x3750e000
485 .word 0xbd3328eb, 0x42f9af75
486 .word 0xbfca85ac4, 0xdfda7000
487 .word 0xbd3785ab, 0x048301ba
488 .word 0xbfca84a6b, 0x759f5000
489 .word 0xbd02ebfe, 0xa903cfb8
490 .word 0xbfca83714, 0xf7bd0000
491 .word 0xbd2ed83a, 0xf85a2ced
492 .word 0xbfca823c1, 0x6551a000
493 .word 0xbd1e0ddb, 0x9a631e83
494 .word 0xbfca81070, 0xbd7b9000
495 .word 0xbfcafe80a, 0x682e646
496 .word 0xbfca7fd22, 0xff599000
497 .word 0xbd3a9d05, 0x02ea120c
498 .word 0xbfca7e9d8, 0x2a0b0000
499 .word 0xbd116849, 0xfa40e4f0
500 .word 0xbfca7d690, 0x3caf5000
501 .word 0xbd359fca, 0x741e7f15
502 .word 0xbfca7c34b, 0x3666a000
503 .word 0xbd3175c9, 0x81b45e10
504 .word 0xbfca7b009, 0x16515000
505 .word 0xbd146280, 0xd3e606a3
506 .word 0xbfca79cc9, 0xdb902000
507 .word 0xbd1e00d0, 0x375e70bd
508 .word 0xbfca7898d, 0x85444000
509 .word 0xbd38e67b, 0xe3dbaf3f
510 .word 0xbfca77654, 0x128f6000
511 .word 0xbd0274ba, 0xdf268e7c
512 .word 0xbfca7631d, 0x82935000
513 .word 0xbd350c41, 0x1c1d060f
514 .word 0xbfca74fe9, 0xd4729000
515 .word 0xbd249736, 0xd91da11e
516 .word 0xbfca73cb9, 0x074fd000

```

```

517 .word 0xbd04cab7, 0x97ffd2cc
518 .word 0xbfc7298b, 0x1a4e3000
519 .word 0xbd15aacc, 0xe43ce383
520 .word 0xbfc71660, 0x0c914000
521 .word 0xbce51b15, 0x7cec3838
522 .word 0xbfc70337, 0xdd3ce000
523 .word 0xbd206a17, 0x8a5eab9c
524 .word 0xbfc6f012, 0x8b756000
525 .word 0xbd357739, 0x0d31ef0f
526 .word 0xbfc6dcf0, 0x165f8000
527 .word 0xbd1b9566, 0x9a33e4c6
528 .word 0xbfc6c9d0, 0x7d203000
529 .word 0xbd3f8e30, 0x14099349
530 .word 0xbfc6b6b3, 0xbedd1000
531 .word 0xbd1a8f73, 0xa64d3813
532 .word 0xbfc6a399, 0xdabbd000
533 .word 0xbd1c1b2c, 0x6657a967
534 .word 0xbfc69082, 0xcfe2b000
535 .word 0xbd2dale7, 0x20b79662
536 .word 0xbfc67d6e, 0x9d785000
537 .word 0xbd2dc2ef, 0x9eb1f25a
538 .word 0xbfc66a5d, 0x42a3a000
539 .word 0xbd3a6893, 0x3aa00298
540 .word 0xbfc6574e, 0xbe8c1000
541 .word 0xbd19cf8b, 0x2c3c2e78
542 .word 0xbfc64443, 0x10594000
543 .word 0xbd22f605, 0xb0281916
544 .word 0xbfc6313a, 0x37335000
545 .word 0xbd3aec82, 0xac378565
546 .word 0xbfc61e34, 0x3242d000
547 .word 0xbd32bb2d, 0x97ecd861
548 .word 0xbfc60b31, 0x00b09000
549 .word 0xbd21d752, 0x6cee0fd8
550 .word 0xbfc5f830, 0xa1a5c000
551 .word 0xbd352268, 0x98ffc1bc
552 .word 0xbfc5e533, 0x144c1000
553 .word 0xbd2c63e8, 0x189ade2b
554 .word 0xbfc5d238, 0x57cd7000
555 .word 0xbd23530a, 0x5ba6e7ac
556 .word 0xbfc5bf40, 0x6b543000
557 .word 0xbd3b63f7, 0x0525d9f9
558 .word 0xbfc5ac4b, 0x4e0b2000
559 .word 0xbd351709, 0xd7275f36
560 .word 0xbfc59958, 0xff1d5000
561 .word 0xbd178be9, 0xa258d7eb
562 .word 0xbfc58669, 0x7db62000
563 .word 0xbd39e26c, 0x65e8cb44
564 .word 0xbfc5737c, 0xc9018000
565 .word 0xbd39baa7, 0xa6b887f6
566 .word 0xbfc56092, 0xe02ba000
567 .word 0xbd245850, 0x06899d98
568 .word 0xbfc54dab, 0xc2610000
569 .word 0xbd2746fe, 0xe5c8d0d8
570 .word 0xbfc53ac7, 0x6ece9000
571 .word 0xbd39ca8a, 0x2a8725d5
572 .word 0xbfc527e5, 0xe4a1b000
573 .word 0xbd2633e8, 0xe5697dc7
574 .word 0xbfc51507, 0x2307f000
575 .word 0xbd306b11, 0xecc0d77b
576 .word 0xbfc5022b, 0x292f6000
577 .word 0xbd348a05, 0xff36a25b
578 .word 0xbfc4ef51, 0xf6466000
579 .word 0xbd3bc83d, 0x21c8cd53
580 .word 0xbfc4dc7b, 0x897bc000
581 .word 0xbd0c79b6, 0x0ae1ff0f
582 .word 0xbfc4c9a7, 0xe1fe8000

```

```

583 .word 0xbcff39f7, 0x50dbbb30
584 .word 0xbfc4b6d6, 0xfe2000
585 .word 0xbd1522ec, 0xf56e7952
586 .word 0xbfc4a408, 0xdfaa7000
587 .word 0xbd33b41f, 0x86e5dd72
588 .word 0xbfc4913d, 0x8333b000
589 .word 0xbd258379, 0x54fdb678
590 .word 0xbfc47e74, 0xe8ca5000
591 .word 0xbd3ef836, 0xa48fdcf
592 .word 0xbfc46baf, 0x0f9f5000
593 .word 0xbd3b6d8c, 0xbe1bdef9
594 .word 0xbfc458eb, 0xf6e3f000
595 .word 0xbcf5c0fe, 0x1f2b8094
596 .word 0xbfc4462b, 0x9dc9b000
597 .word 0xbd1ede9d, 0x63b93e7a
598 .word 0xbfc4336e, 0x03829000
599 .word 0xbd3ac363, 0xa859c2af
600 .word 0xbfc420b3, 0x2740f000
601 .word 0xbd3ba75f, 0x4de97ddf
602 .word 0xbfc40dfb, 0x08378000
603 .word 0xbcc9bb453, 0xc4f7b685
604 .word 0xbfc3fb45, 0xa5992000
605 .word 0xbd319713, 0xc0cae559
606 .word 0xbfc3e892, 0xfe995000
607 .word 0xbd2b6aad, 0x914d5249
608 .word 0xbfc3d5e3, 0x126bc000
609 .word 0xbd13fb2f, 0x85096c4b
610 .word 0xbfc3c335, 0xe0447000
611 .word 0xbd3ae77d, 0x114a8b5f
612 .word 0xbfc3b08b, 0x6757f000
613 .word 0xbd15485c, 0x35b37c15
614 .word 0xbfc39de3, 0xa6dae000
615 .word 0xbd284fc7, 0x32ce95f1
616 .word 0xbfc38b3e, 0x9e027000
617 .word 0xbd21e21f, 0x5747d00e
618 .word 0xbfc3789c, 0x4c041000
619 .word 0xbd19b4f4, 0x44d31e60
620 .word 0xbfc365fc, 0xb0159000
621 .word 0xbcc62fa8, 0x234b7289
622 .word 0xbfc3535f, 0xc96d1000
623 .word 0xbd013f1c, 0x3b1fab68
624 .word 0xbfc340c5, 0x97411000
625 .word 0xbd20b846, 0x104c58f3
626 .word 0xbfc32e2e, 0x18c86000
627 .word 0xbd3e6220, 0x6c327115
628 .word 0xbfc31b99, 0x4d3a4000
629 .word 0xbd3f098e, 0xe3a50810
630 .word 0xbfc30907, 0x33ce3000
631 .word 0xbd33f323, 0x7c4d853e
632 .word 0xbfc2f677, 0xcbbc0000
633 .word 0xbd352b30, 0x2160f40d
634 .word 0xbfc2e3eb, 0x143bf000
635 .word 0xbd218910, 0x2710016e
636 .word 0xbfc2d161, 0xc8680000
637 .word 0xbd039d6c, 0xcb81b4a1
638 .word 0xbfc2bed9, 0xb3d49000
639 .word 0xbd095245, 0x4a40d26b
640 .word 0xbfc2ac55, 0x095f5000
641 .word 0xbd38b2e6, 0x4bce4dd6
642 .word 0xbfc299d3, 0xc6060000
643 .word 0xbd3d4d00, 0x79dc08d9
644 .word 0xbfc28753, 0xbcl1a000
645 .word 0xbd37494e, 0x359302e6
646 .word 0xbfc274d7, 0x17ad4000
647 .word 0xbd38a65b, 0xa0967592
648 .word 0xbfc2625d, 0x1e6dd000

```

```

649 .word 0xbd3ead69, 0xd0f61c28
650 .word 0xbfc24fe5, 0xcfe8e4000
651 .word 0xbd318f96, 0x26b10d30
652 .word 0xbfc23d71, 0x2a49c000
653 .word 0xbd100d23, 0x8fd3df5c
654 .word 0xbfc22aff, 0x2ddbd000
655 .word 0xbd32e1ea, 0xca7cb4f0
656 .word 0xbfc2188f, 0xd9807000
657 .word 0xbd131786, 0x02bce3fb
658 .word 0xbfc20623, 0x2c73c000
659 .word 0xbd2351a5, 0x02bb95f5
660 .word 0xbfc1f3b9, 0x25f25000
661 .word 0xbd3a822c, 0x593df273
662 .word 0xbfc1e151, 0xc5391000
663 .word 0xbd38e5f5, 0xf578d80e
664 .word 0xbfc1ceed, 0x09853000
665 .word 0xbd2d47c7, 0x8dcdaae
666 .word 0xbfc1bc8a, 0xf2143000
667 .word 0xbd2acd64, 0xf955458
668 .word 0xbfc1aa2b, 0x7e23f000
669 .word 0xbd2ca78e, 0x44389934
670 .word 0xbfc197ce, 0xacf2a000
671 .word 0xbd31abl14, 0x4caf6736
672 .word 0xbfc18574, 0x7dbec000
673 .word 0xbd3e6744, 0x45bd9b49
674 .word 0xbfc1731c, 0xefc74000
675 .word 0xbfcfde27c, 0xd98317fd
676 .word 0xbfc160c8, 0x024b2000
677 .word 0xbd2ec2d2, 0xa9009e3d
678 .word 0xbfc14e75, 0xb489f000
679 .word 0xbd33fd84, 0x66dfe192
680 .word 0xbfc13c26, 0x05c39000
681 .word 0xbd318501, 0x13584d7c
682 .word 0xbfc129d8, 0xf5381000
683 .word 0xbd1d77cc, 0x415a172e
684 .word 0xbfc1178e, 0x8227e000
685 .word 0xbd21ef78, 0xce2d07f2
686 .word 0xbfc10546, 0xabd3d000
687 .word 0xbd00189b, 0x51d162e8
688 .word 0xbfc0f301, 0x17cf000
689 .word 0xbcff64bb, 0xe51793b4
690 .word 0xbfc0e0be, 0xd264a000
691 .word 0xbd3baf2, 0x3aeb549c
692 .word 0xbfc0ce7e, 0xcdccc000
693 .word 0xbd14652d, 0xabff5447
694 .word 0xbfc0bc41, 0x62f73000
695 .word 0xbd36ca04, 0x73bd9c29
696 .word 0xbfc0aa06, 0x91267000
697 .word 0xbd2755cc, 0x51f9bdae
698 .word 0xbfc097ce, 0x579d2000
699 .word 0xbce33742, 0xda652881
700 .word 0xbfc08598, 0xb59e3000
701 .word 0xbd340d11, 0x47fb37ea
702 .word 0xbfc07365, 0xaa6d1000
703 .word 0xbd16e172, 0x43f1226a
704 .word 0xbfc06135, 0x354d4000
705 .word 0xbd363046, 0x28340ee9
706 .word 0xbfc04f07, 0x5582d000
707 .word 0xbd1a3d31, 0x4c780403
708 .word 0xbfc03cdc, 0xa51e000
709 .word 0xbd381a9c, 0xf169fc5c
710 .word 0xbfc02ab3, 0x52ff2000
711 .word 0xbd27ce63, 0x5d569b2b
712 .word 0xbfc0188d, 0x2ecf6000
713 .word 0xbd03f965, 0x1cff9dfe
714 .word 0xbfc00669, 0x9d07c000

```

```

715 .word 0xbd3b8775, 0x304686e1
716 .word 0xbfbfe891, 0x39dbd000
717 .word 0xbd159653, 0x60bdea07
718 .word 0xbfbfc454, 0x5b8f0000
719 .word 0xbd29cba7, 0xd5591204
720 .word 0xbfbfa01c, 0x9db57000
721 .word 0xbd29c32b, 0x816dd634
722 .word 0xbfbf7be9, 0xfedbf000
723 .word 0xbd2bcbe8, 0xb535310e
724 .word 0xbfbf57bc, 0x7d900000
725 .word 0xbd176a6c, 0x9ea8b04e
726 .word 0xbfbf3394, 0x185fa000
727 .word 0xbd1ea383, 0x09d097b7
728 .word 0xbfbf0f70, 0xcd990000
729 .word 0xbd0718fb, 0x613960ee
730 .word 0xbfbbeb52, 0x9c8d1000
731 .word 0xbd0b6260, 0x903c8f99
732 .word 0xbfbec739, 0x830a1000
733 .word 0xbcf1fcba, 0x80cdd0fe
734 .word 0xbfbfa325, 0x7fe10000
735 .word 0xbd2ef30d, 0x47e4627a
736 .word 0xbfbf7f16, 0x91a32000
737 .word 0xbd2a7c74, 0xc871080d
738 .word 0xbfbf5b0c, 0xb6e22000
739 .word 0xbd109021, 0x3b34d95f
740 .word 0xbfbf3707, 0xee304000
741 .word 0xbd20f684, 0xe6766abd
742 .word 0xbfbf1308, 0x36208000
743 .word 0xbd21aeaa, 0xf90019f9
744 .word 0xbfbdef0d, 0x8d466000
745 .word 0xbd2b715f, 0x7da2cb17
746 .word 0xbfbfdbcb17, 0xf2361000
747 .word 0xbd226a0a, 0x5ba47956
748 .word 0xbfbda727, 0x63844000
749 .word 0xbd1a8940, 0x1fa71733
750 .word 0xbfbdb833b, 0xdfc64000
751 .word 0xbd24805c, 0x07408695
752 .word 0xbfbfd5f55, 0x65921000
753 .word 0xbceec4739, 0x830a8d2a
754 .word 0xbfbdb3b73, 0xf37e1000
755 .word 0xbd2f3501, 0x33da5007
756 .word 0xbfbdb1797, 0x88219000
757 .word 0xbd0b219d, 0xaf7df76b
758 .word 0xbfbfbcf3c0, 0x22142000
759 .word 0xbce9d2b6, 0x6ddd996f
760 .word 0xbfbccfed, 0xbfee1000
761 .word 0xbd0d4119, 0xf3892ad
762 .word 0xbfbcac20, 0x60484000
763 .word 0xbd2d53ed, 0xcc4f420b
764 .word 0xbfbcb8858, 0x01bc4000
765 .word 0xbd2646d1, 0xc65aacd3
766 .word 0xbfbcb6494, 0xa2e41000
767 .word 0xbd214bd1, 0x564189cb
768 .word 0xbfbcb40d6, 0x425a5000
769 .word 0xbd296224, 0x3a3261b9
770 .word 0xbfbcb1d1c, 0xdeb5000
771 .word 0xbd02f7e7, 0x23a02373
772 .word 0xbfbbbf968, 0xf69fc000
773 .word 0xbd24218c, 0x8d824283
774 .word 0xbfbdbd5b9, 0x08a72000
775 .word 0xbd2236aa, 0x3ae84f31
776 .word 0xbfbbbb20e, 0x936d6000
777 .word 0xbd22e8af, 0x9574c8e4
778 .word 0xbfbbb8e69, 0x15901000
779 .word 0xbd22bef7, 0xf208fbd9
780 .word 0xbfbbb6ac8, 0x8dad5000

```

```

781 .word 0xbd2637bf, 0xea044b8d
782 .word 0xbfbb472c, 0xfa63e000
783 .word 0xbd1246f5, 0xc7f4588b
784 .word 0xbfbb2396, 0x5a52f000
785 .word 0xbd2e009b, 0x115ec8f8
786 .word 0xbfbb0004, 0xacla8000
787 .word 0xbdl1aaf97, 0x037f2b35
788 .word 0xbfbadc77, 0xee5ae000
789 .word 0xbd25189b, 0xec79cdf7
790 .word 0xbfbab8f0, 0x1fb52000
791 .word 0xbd27f69d, 0xd23d3ac2
792 .word 0xbfba956d, 0x3ecad000
793 .word 0xbd2cc6f2, 0x9805895f
794 .word 0xbfba71ef, 0x4a3e2000
795 .word 0xbdl1bbc94, 0x7b201fbf
796 .word 0xbfba4e76, 0x40b1b000
797 .word 0xbd286f52, 0x51aeefe0e
798 .word 0xbfba2b02, 0x20c8e000
799 .word 0xbdl17d329, 0x8e6b7dbf
800 .word 0xbfba0792, 0xe9277000
801 .word 0xbd2958c6, 0x4d94ab90
802 .word 0xbfbb9e428, 0x9871e000
803 .word 0xbd22c483, 0xd0942b9c
804 .word 0xbfbb9c0c3, 0x2d4d2000
805 .word 0xbdl1520fd, 0x85f1e661
806 .word 0xbfbb99d62, 0xa65eb000
807 .word 0xbd22dd17, 0xd834450a
808 .word 0xbfbb97a07, 0x024cb000
809 .word 0xbd2ce867, 0xd19bed86
810 .word 0xbfbb956b0, 0x3fbd000
811 .word 0xbd286fb6, 0x03fe1b67
812 .word 0xbfbb9335e, 0x5d594000
813 .word 0xbd23115c, 0x3abd47da
814 .word 0xbfbb91011, 0x59c6c000
815 .word 0xbd27af17, 0x9df80b59
816 .word 0xbfbb8ecc9, 0x33aeb000
817 .word 0xbdl1ba18c, 0x833010ab
818 .word 0xbfbb8c985, 0xe9b9e000
819 .word 0xbd290791, 0x0379ff94
820 .word 0xbfbb8a647, 0x7a91d000
821 .word 0xbd285181, 0x5f37adbf
822 .word 0xbfbb8830d, 0xe4e08000
823 .word 0xbd05f60b, 0x79c8f66a
824 .word 0xbfbb85fd9, 0x27506000
825 .word 0xbd248fcf, 0xccd1e7c7
826 .word 0xbfbb83ca9, 0x408ca000
827 .word 0xbd2326c8, 0xd744c7d1
828 .word 0xbfbb8197e, 0x2f40e000
829 .word 0xbd0f80dc, 0xf96ffdf7
830 .word 0xbfbb7f657, 0xf2194000
831 .word 0xbd21bef9, 0x43faf4d2
832 .word 0xbfbb7d336, 0x87c29000
833 .word 0xbd0e4461, 0xf3833832
834 .word 0xbfbb7b019, 0xeeea0000
835 .word 0xbd275649, 0xaee848d4
836 .word 0xbfbb78d02, 0x263d8000
837 .word 0xbd069b57, 0x94b69fb7
838 .word 0xbfbb769ef, 0x2c6b5000
839 .word 0xbdl1a35d8, 0xc73b6a55
840 .word 0xbfbb746e1, 0x0226000
841 .word 0xbd2db25d, 0x23c3bc5b
842 .word 0xbfbb723d7, 0xa0123000
843 .word 0xbd2c3cbb, 0x84fef08e
844 .word 0xbfbb700d3, 0x0aeac000
845 .word 0xbcec1e8d, 0xa99ded32
846 .word 0xbfbb6ddd3, 0x3f5c7000

```

```

847 .word 0xbd2aeb06, 0x82906a06
848 .word 0xbfbb6bad8, 0x3c188000
849 .word 0xbd0daf3c, 0xc08926ae
850 .word 0xbfbb697e1, 0xffd06000
851 .word 0xbd296c57, 0x15a12bb6
852 .word 0xbfbb674f0, 0x89365000
853 .word 0xbd24f332, 0x993a6604
854 .word 0xbfbb65203, 0xd6fcf000
855 .word 0xbdl1ea006, 0x8199326b
856 .word 0xbfbb62f1b, 0xe7d77000
857 .word 0xbdl1d0cd5, 0x02538764
858 .word 0xbfbb60c38, 0xba799000
859 .word 0xbdl172c4, 0x3aec129e
860 .word 0xbfbb5e95a, 0xd979000
861 .word 0xbbcfcb7ce, 0x1d171711
862 .word 0xbfbb5c680, 0x9fe63000
863 .word 0xbd23c479, 0x935581b6
864 .word 0xbfbb5a3ab, 0xb01ad000
865 .word 0xbd2c4ae9, 0x3cd5f430
866 .word 0xbfbb580db, 0x7ceb5000
867 .word 0xbdl1c07f6, 0xcbe60d53
868 .word 0xbfbb55e10, 0x050e0000
869 .word 0xbd0c1d74, 0xc53c72e
870 .word 0xbfbb53b49, 0x4739c000
871 .word 0xbd221868, 0x5306aaa5
872 .word 0xbfbb51887, 0x42261000
873 .word 0xbd0850ec, 0xb12c59ec
874 .word 0xbfbb4f5c9, 0xf48ad000
875 .word 0xbd0580c1, 0x2c81f8fd
876 .word 0xbfbb4d311, 0x5d207000
877 .word 0xbd2d58bb, 0x4fa163c2
878 .word 0xbfbb4b05d, 0x7aa01000
879 .word 0xbd07029c, 0x6ef93715
880 .word 0xbfbb48dae, 0x4bc31000
881 .word 0xbcb85b20, 0x8c200bea
882 .word 0xbfbb46b03, 0xcf437000
883 .word 0xbd2787a5, 0x2f0f629e
884 .word 0xbfbb4485e, 0x03dbd000
885 .word 0xbd2f5a8d, 0xd1a4d56e
886 .word 0xbfbb425bc, 0xe8474000
887 .word 0xbd2365ac, 0x5219daef
888 .word 0xbfbb40320, 0x7b414000
889 .word 0xbd26fd84, 0xaa8157c0
890 .word 0xbfbb3e088, 0xbb85f000
891 .word 0xbd248068, 0xbdc331fa
892 .word 0xbfbb3bdf5, 0xa7d1e000
893 .word 0xbd2cc85e, 0xa5db4ed7
894 .word 0xbfbb39b67, 0x3ee24000
895 .word 0xbd0a759b, 0xa99f5667
896 .word 0xbfbb378dd, 0xf749000
897 .word 0xbdl1c5044, 0xa3c7eb28
898 .word 0xbfbb35658, 0x68470000
899 .word 0xbd2464d7, 0x0035b508
900 .word 0xbfbb333d7, 0xf8183000
901 .word 0xbd2e96d4, 0x957e477c
902 .word 0xbfbb3115c, 0x2da75000
903 .word 0xbd25bc37, 0x0651448
904 .word 0xbfbb2eee5, 0x07b40000
905 .word 0xbd08081e, 0xdd77c860
906 .word 0xbfbb2cc72, 0x84fe5000
907 .word 0xbd2e38bd, 0x0cb32a28
908 .word 0xbfbb2aa04, 0xa4471000
909 .word 0xbdl1e922e, 0xa2c72d06
910 .word 0xbfbb2879b, 0x644f5000
911 .word 0xbdl1752b6, 0xf65943ec
912 .word 0xbfbb26536, 0xc3d8c000

```

```

913 .word 0xbd0b4bac, 0x097c5ba3
914 .word 0xbfb242d6, 0xc1a58000
915 .word 0xbd24b838, 0xac648481
916 .word 0xbfb2207b, 0x5c785000
917 .word 0xbd127633, 0xf0431efb
918 .word 0xbfb1fe24, 0x93144000
919 .word 0xbd27a374, 0xela7c696
920 .word 0xbfb1dbd2, 0x643d1000
921 .word 0xbd221649, 0xb2ef8928
922 .word 0xbfb1b984, 0xceb6e000
923 .word 0xbd121a31, 0x2f307601
924 .word 0xbfb1973b, 0xd1465000
925 .word 0xbd159b45, 0x53e4c2cb
926 .word 0xbfb174f7, 0x6ab09000
927 .word 0xbcf71031, 0x7ee2e483
928 .word 0xbfb152b7, 0x99bb3000
929 .word 0xbd299135, 0xbe3f3df6
930 .word 0xbfb1307c, 0x5d2c7000
931 .word 0xbd2357c9, 0xfa3dbf1f
932 .word 0xbfb10e45, 0xb3cae000
933 .word 0xbd20612d, 0xaf6b9737
934 .word 0xbfb0ec13, 0x9c5da000
935 .word 0xbd180247, 0xe54ebd73
936 .word 0xbfb0c9e6, 0x15ac4000
937 .word 0xbd2c2da8, 0x0974d976
938 .word 0xbfb0a7bd, 0x1e7ef000
939 .word 0xbd20f926, 0xcdf8dfb4
940 .word 0xbfb08598, 0xb59e3000
941 .word 0xbd240d11, 0x47fb37ea
942 .word 0xbfb06378, 0xd9d32000
943 .word 0xbd104990, 0x672b0729
944 .word 0xbfb0415d, 0x89e74000
945 .word 0xbd1111c0, 0x5cfl7d53
946 .word 0xbfb01f46, 0xc4a4a000
947 .word 0xbd11157c, 0x89ecf845
948 .word 0xbfaffa69, 0x11ab9000
949 .word 0xbcf80464, 0xc1c0d47a
950 .word 0xbfafb64d, 0xaa8b6000
951 .word 0xbd13830d, 0xaeb373e0
952 .word 0xbfaf723b, 0x517fc000
953 .word 0xbd048a79, 0x154f796a
954 .word 0xbfaf2e32, 0x04209000
955 .word 0xbcfb9ba8, 0x2f4d6e7f
956 .word 0xbfaeea31, 0xc006b000
957 .word 0xbd10f760, 0xd81b6242
958 .word 0xbfaea63a, 0x82cc0000
959 .word 0xbd19f144, 0x08e210e7
960 .word 0xbfae624c, 0x4a0b5000
961 .word 0xbd1c368e, 0x2e6265dd
962 .word 0xbfae1e67, 0x13606000
963 .word 0xbd1a0d3c, 0xb7b141db
964 .word 0xbfaadda8a, 0xdc67e000
965 .word 0xbd1c9ca7, 0x364c37a2
966 .word 0xbfad96b7, 0xa2bf8000
967 .word 0xbd12eb81, 0xf49d3d78
968 .word 0xbfad52ed, 0x6405d000
969 .word 0xbd10de8b, 0x575910a6
970 .word 0xbfad0f2c, 0x1dda6000
971 .word 0xbd0c6fc7, 0x04385ddf
972 .word 0xbfacb73, 0xcdddb000
973 .word 0xbcf65c36, 0xe09f5fe2
974 .word 0xbfac87c4, 0x71b12000
975 .word 0xbd13799a, 0xf29d923d
976 .word 0xbfac441e, 0x06f72000
977 .word 0xbd153c7d, 0x26143455
978 .word 0xbfac0080, 0x8b530000

```

```

979 .word 0xbd003c05, 0x63baea2e
980 .word 0xbfabbcceb, 0xfc68f000
981 .word 0xbd0080f2, 0xe79d07ab
982 .word 0xbfab7960, 0x57de2000
983 .word 0xbd0f5af1, 0xf7b24d0f
984 .word 0xbfab35dd, 0x9b58b000
985 .word 0xbd1559d3, 0x5b3d5639
986 .word 0xbfaaf263, 0xc47fb000
987 .word 0xbd085458, 0x172a97ad
988 .word 0xbfaaaef2, 0xd0fb1000
989 .word 0xbcdf8346, 0xa77685c1
990 .word 0xbfaa6b8a, 0xbe73a000
991 .word 0xbd1e988d, 0x46e25c90
992 .word 0xbfaa282b, 0xa936000
993 .word 0xbce70a67, 0xf10371d7
994 .word 0xbfa9e4d5, 0x3304e000
995 .word 0xbcfec4a6, 0x991acef2
996 .word 0xbfa9a187, 0xb573d000
997 .word 0xbd1cf746, 0xc4ec9bca
998 .word 0xbfa95e43, 0x0f8ce000
999 .word 0xbd01774c, 0x225e2c8d
1000 .word 0xbfa91b07, 0x3efd7000
1001 .word 0xbcf8a0eb, 0x0224d5a9
1002 .word 0xbfa8d7d4, 0x4173f000
1003 .word 0xbcf24a7b, 0x7a089116
1004 .word 0xbfa894aa, 0x149fb000
1005 .word 0xbcf19a8, 0xbe97660a
1006 .word 0xbfa85188, 0xb630f000
1007 .word 0xbcca0544, 0x165f80aa
1008 .word 0xbfa80e70, 0x23d8c000
1009 .word 0xbd1988fa, 0x435d02ec
1010 .word 0xbfa7cb60, 0x5b495000
1011 .word 0xbcf8af3, 0x69d6d0f4
1012 .word 0xbfa78859, 0x5a357000
1013 .word 0xbd0ee9e5, 0xef898b68
1014 .word 0xbfa7455b, 0x1e511000
1015 .word 0xbcfb28ce, 0xb91e296d
1016 .word 0xbfa70265, 0xa550e000
1017 .word 0xbd0ddc83, 0xb80a8c63
1018 .word 0xbfa6bf78, 0xcea9000
1019 .word 0xbd163cc0, 0x0f16f7e9
1020 .word 0xbfa67c94, 0xf2d4b000
1021 .word 0xbd16b082, 0x09f3282f
1022 .word 0xbfa639b9, 0xb4c6b000
1023 .word 0xbd14f37b, 0x6b7f9673
1024 .word 0xbfa5f6e7, 0x3078e000
1025 .word 0xbd1f6f4a, 0xffdb6d69
1026 .word 0xbfa5b41d, 0x63a49000
1027 .word 0xbd0abcc4, 0x7e8a0c20
1028 .word 0xbfa5715c, 0x4c03c000
1029 .word 0xbd1ddd8c, 0x80ee2760
1030 .word 0xbfa52ea3, 0xe7519000
1031 .word 0xbd16ff79, 0x68012363
1032 .word 0xbfa4ebf4, 0x3349e000
1033 .word 0xbcf37578, 0x4620c465
1034 .word 0xbfa4a94d, 0x2da96000
1035 .word 0xbd18ace0, 0x8a56ed78
1036 .word 0xbfa466ae, 0xd42de000
1037 .word 0xbcff4c64, 0x521016be
1038 .word 0xbfa42419, 0x2495d000
1039 .word 0xbd05f329, 0x88dd64a6
1040 .word 0xbfa3e18c, 0x1ca0a000
1041 .word 0xbd1d23b4, 0xfdb8de39
1042 .word 0xbfa39f07, 0xba0eb000
1043 .word 0xbd1ac4a7, 0x590b95de
1044 .word 0xbfa35c8b, 0xfaa13000

```



```

1045 .word 0xbccabeaf, 0x7cf59aac
1046 .word 0xbfa31a18, 0xdc1a1000
1047 .word 0xbd07dd58, 0xd860ceab
1048 .word 0xbfa2d7ae, 0x5c3c5000
1049 .word 0xbd175b1a, 0xe989664c
1050 .word 0xbfa2954c, 0x78cbc000
1051 .word 0xbd1c3526, 0x570c1572
1052 .word 0xbfa252f3, 0x2f8d1000
1053 .word 0xbd107d35, 0xc0436cf5
1054 .word 0xbfa210a2, 0x7e45c000
1055 .word 0xbcf8ceca, 0x131bef9c
1056 .word 0xbfalce5a, 0x62bc3000
1057 .word 0xbd04e63c, 0x6c6fccc5
1058 .word 0xbfa18c1a, 0xdab7b000
1059 .word 0xbcf22af4, 0xd32f2ac0
1060 .word 0xbfa149e3, 0xe4005000
1061 .word 0xbd1519d5, 0x96fa5c0c
1062 .word 0xbfa107b5, 0x7c5f2000
1063 .word 0xbd152b81, 0xe94af0a6
1064 .word 0xbfa0c58f, 0xa19df000
1065 .word 0xbd155317, 0x53a74377
1066 .word 0xbfa08372, 0x51877000
1067 .word 0xbd1cc91e, 0xb2004222
1068 .word 0xbfa0415d, 0x89e74000
1069 .word 0xbd0111c0, 0x5cfd1d753
1070 .word 0xbf9ffea2, 0x91136000
1071 .word 0xbd04dd01, 0xd7640dc2
1072 .word 0xbf9f7a9b, 0x16782000
1073 .word 0xbd00ab64, 0x9c6f9f5c
1074 .word 0xbf9ef6a4, 0x9f98f000
1075 .word 0xbd0671e4, 0xe8f151a3
1076 .word 0xbf9e72bf, 0x2813c000
1077 .word 0xbd0ca2ba, 0xda22cae5
1078 .word 0xbf9deeea, 0xab883000
1079 .word 0xbd0c6e1d, 0x7741b591
1080 .word 0xbf9d6b27, 0x25979000
1081 .word 0xbd000425, 0x79723e3d
1082 .word 0xbf9ce774, 0x91e4d000
1083 .word 0xbd00d7ce, 0xf3d25198
1084 .word 0xbf9c63d2, 0xec14a000
1085 .word 0xbd05e318, 0xfe7acbca
1086 .word 0xbf9be042, 0x2fcd6000
1087 .word 0xbd01ec42, 0x87f2c9ca
1088 .word 0xbf9b5cc2, 0x58b71000
1089 .word 0xbd01cc23, 0x715f7fd0
1090 .word 0xbf9ad953, 0x627b6000
1091 .word 0xbd0ab5a1, 0x1a805efd
1092 .word 0xbf9a55f5, 0x48c5c000
1093 .word 0xbcf0fc7b, 0x0697e1b5
1094 .word 0xbf99d2a8, 0x07432000
1095 .word 0xbcf7cf80, 0x538b441e
1096 .word 0xbf994f6b, 0x99a24000
1097 .word 0xbcf1d5ef, 0x96cf7f51
1098 .word 0xbf98cc3f, 0xfb937000
1099 .word 0xbd050394, 0x323f2c7a
1100 .word 0xbf984925, 0x28c8c000
1101 .word 0xbd057d17, 0x3697cf30
1102 .word 0xbf97c61b, 0x1cf5d000
1103 .word 0xbd0dc0dc, 0x1ed96ee4
1104 .word 0xbf974321, 0xd3d00000
1105 .word 0xbcfb4a69, 0x0fe94778
1106 .word 0xbf96c039, 0x490e3000
1107 .word 0xbcff7b34, 0x02fd59ca
1108 .word 0xbf963d61, 0x78690000
1109 .word 0xbd07abf3, 0x89596542
1110 .word 0xbf95ba9a, 0x5d9ac000

```

```

1111 .word 0xbcacbb84, 0xe08d78ac
1112 .word 0xbf9537e3, 0xf45f3000
1113 .word 0xbcf592ce, 0x96bf9299
1114 .word 0xbf94b53e, 0x3873e000
1115 .word 0xbd0b6ee9, 0xbca265c1
1116 .word 0xbf9432a9, 0x25980000
1117 .word 0xbd098139, 0x928637fe
1118 .word 0xbf93b024, 0xb78c5000
1119 .word 0xbcf9a5e2, 0x3a02f82a
1120 .word 0xbf932db0, 0xea132000
1121 .word 0xbd0c432c, 0x4c2257ef
1122 .word 0xbf92ab4d, 0xb8f09000
1123 .word 0xbcf82c84, 0xa532c74c
1124 .word 0xbf9228fb, 0x1fea2000
1125 .word 0xbd0c4f8c, 0xa12647f9
1126 .word 0xbf91a6b9, 0x1ac73000
1127 .word 0xbcec30e9, 0xb54e2dd6
1128 .word 0xbf912487, 0xa5507000
1129 .word 0xbd0edf2f, 0xf6a59c94
1130 .word 0xbf90a266, 0xbb508000
1131 .word 0xbcf9a5be1, 0x7c2ec500
1132 .word 0xbf902056, 0x58935000
1133 .word 0xbd008e93, 0xe47420b7
1134 .word 0xbf8f3cac, 0xf1cd3000
1135 .word 0xbcf64d83, 0xc9a6875d
1136 .word 0xbf8e38ce, 0x30333000
1137 .word 0xbcc0bbae, 0x12ebf308
1138 .word 0xbf8d3510, 0x63fa4000
1139 .word 0xbcea8d92, 0xdf00beb
1140 .word 0xbf8c3173, 0x84c75000
1141 .word 0xbcf9e0cc0, 0x31046026
1142 .word 0xbf8b2df7, 0x8a428000
1143 .word 0xbcf4c647, 0xa5d4542f
1144 .word 0xbf8a2a9c, 0x6c170000
1145 .word 0xbce18876, 0x525971bc
1146 .word 0xbf892762, 0x21f33000
1147 .word 0xbcd456ba, 0x9344a27f
1148 .word 0xbf882448, 0xa388a000
1149 .word 0xbcd55104, 0xb16137f1
1150 .word 0xbf87214f, 0xe88c0000
1151 .word 0xbcf27275, 0xd7338080
1152 .word 0xbf861e77, 0xe8b53000
1153 .word 0xbcf8c11, 0x507150cb
1154 .word 0xbf851bc0, 0x9bbf4000
1155 .word 0xbcd9e1ea, 0x5258a3c6
1156 .word 0xbf841929, 0xf9683000
1157 .word 0xbcd77c75, 0x5d013688
1158 .word 0xbf8316b3, 0xf9714000
1159 .word 0xbcfb8dcc, 0x8ba5563d
1160 .word 0xbf82145e, 0x939ef000
1161 .word 0xbcce891c, 0x6274ffda
1162 .word 0xbf811229, 0xbfb89000
1163 .word 0xbcf50ee4, 0x5fd053b1
1164 .word 0xbf801015, 0x7588d000
1165 .word 0xbcfce251, 0x998b505f
1166 .word 0xbf7e1c43, 0x59bad000
1167 .word 0xbce9f504, 0xdabb6021
1168 .word 0xbf7c189c, 0xbb0e2000
1169 .word 0xbcdf9ebb, 0x69dea7ed
1170 .word 0xbf7a1536, 0xfeb35000
1171 .word 0xbcecb8e8, 0x91b69c25
1172 .word 0xbf781212, 0x14586000
1173 .word 0xbce6a81c, 0x14b9f937
1174 .word 0xbf760f2d, 0xebb16000
1175 .word 0xbcb86835, 0x84891753
1176 .word 0xbf740c8a, 0x74787000

```

```

1177 .word 0xbce1c38e, 0xf838000c
1178 .word 0xbf720a27, 0x9e6e0000
1179 .word 0xbce34d96, 0x922727aa
1180 .word 0xbf700805, 0x59588000
1181 .word 0xbce66afc, 0xb31c67b2
1182 .word 0xbf6c0c47, 0x2a092000
1183 .word 0xbc657d36, 0x31cacba0
1184 .word 0xbf680904, 0x82898000
1185 .word 0xbcc701a5, 0xa9c30314
1186 .word 0xbf640642, 0x9be3c000
1187 .word 0xbcccf0de, 0xc26e96f3
1188 .word 0xbf600401, 0x55d58000
1189 .word 0xbcd13bce, 0x0ce3ddd8
1190 .word 0xbf580481, 0x20511000
1191 .word 0xbcc0a8ce, 0x7ceb0de6
1192 .word 0xbf500200, 0x55655000
1193 .word 0xbcc11266, 0xaf9afc3f
1194 .word 0xbf400100, 0x15575000
1195 .word 0xbca62237, 0x79c0dc11
1196 .word 0x00000000, 0x00000000
1197 .word 0x00000000, 0x00000000
1198 .word 0x3f4ffc00, 0xaa8ab000
1199 .word 0x3c80fbc0, 0x4d051925
1200 .word 0x3f5ff802, 0xa9ab1000
1201 .word 0x3c8ccf14, 0xf1d0a9f2
1202 .word 0x3f67f704, 0x7d798000
1203 .word 0x3cbcd344, 0xeb43240a
1204 .word 0x3f6ff00a, 0xa2b10000
1205 .word 0x3cd78094, 0x10d6ad37
1206 .word 0x3f73f38a, 0x60f06000
1207 .word 0x3cd22569, 0x3c937494
1208 .word 0x3f77ee11, 0xebd82000
1209 .word 0x3ced274f, 0x0b48e81d
1210 .word 0x3f7be79c, 0x70058000
1211 .word 0x3ced91f3, 0x4d808088
1212 .word 0x3f7fe02a, 0x6b106000
1213 .word 0x3cde23f0, 0xdda40e47
1214 .word 0x3f81ebde, 0x2d199000
1215 .word 0x3cef97c0, 0x0b723c9a
1216 .word 0x3f83e729, 0x5d25a000
1217 .word 0x3cef63e0, 0xd65eabc
1218 .word 0x3f85e1f7, 0x03ecb000
1219 .word 0x3cfca09f, 0x585da1b5
1220 .word 0x3f87dc47, 0x5f810000
1221 .word 0x3cf4edba, 0x4a25e0b1
1222 .word 0x3f89d61a, 0xadc6b000
1223 .word 0x3cfb1963, 0x27b4256d
1224 .word 0x3f8bcf71, 0x2c743000
1225 .word 0x3cf09782, 0x5ef65dc3
1226 .word 0x3f8dc84b, 0x19123000
1227 .word 0x3cf02950, 0x78e96cc1
1228 .word 0x3f8fc0a8, 0xb0fc0000
1229 .word 0x3cdf1e7c, 0xf6d3a69c
1230 .word 0x3f90dc45, 0x18afc000
1231 .word 0x3d090f43, 0x1ff3b010
1232 .word 0x3f91d7f7, 0xeb9ee000
1233 .word 0x3d07cd8a, 0xf80670b5
1234 .word 0x3f92d36c, 0xfbf55000
1235 .word 0x3cff0bb3, 0x41706c38
1236 .word 0x3f93cea4, 0x4346a000
1237 .word 0x3cf5d3bc, 0xd295bf53
1238 .word 0x3f94c99e, 0x04901000
1239 .word 0x3d0bd98c, 0xbbebe949
1240 .word 0x3f95c45a, 0x51b8d000
1241 .word 0x3cec449d, 0xe927827c
1242 .word 0x3f96bed9, 0x48d1b000

```

```

1243 .word 0x3cff43be, 0x9f5bc086
1244 .word 0x3f97b91b, 0x07d5b000
1245 .word 0x3cd1aa92, 0x7f54c717
1246 .word 0x3f98b31f, 0xaca9b000
1247 .word 0x3c8c3ab4, 0x8db4decf
1248 .word 0x3f99ace7, 0x551cc000
1249 .word 0x3cf45134, 0x09c1df81
1250 .word 0x3f9aa672, 0x1ee83000
1251 .word 0x3cf6a75a, 0xe2d7a49d
1252 .word 0x3f9b9fc0, 0x27af9000
1253 .word 0x3cd97fbd, 0x465b7589
1254 .word 0x3f9c98d1, 0x8d00c000
1255 .word 0x3d0027ab, 0xe9d883c3
1256 .word 0x3f9d91a6, 0x6c543000
1257 .word 0x3d0987c5, 0x9633ee68
1258 .word 0x3f9e8a3e, 0x3cd0000
1259 .word 0x3d095817, 0x086b1c01
1260 .word 0x3f9f829b, 0x0e783000
1261 .word 0x3ce80267, 0xc7e09e3e
1262 .word 0x3fa03d5d, 0x85e73000
1263 .word 0x3d1dde25, 0x83b4a73b
1264 .word 0x3fa0b94f, 0x7c196000
1265 .word 0x3ce76769, 0xfdd87d3
1266 .word 0x3fa13523, 0x78597000
1267 .word 0x3cef29e2, 0x4702d328
1268 .word 0x3falb0d9, 0x8923d000
1269 .word 0x3d12ff85, 0x945dd915
1270 .word 0x3fa22c71, 0xbcea8000
1271 .word 0x3cfd2818, 0xf87f888f
1272 .word 0x3fa2a7ec, 0x2214e000
1273 .word 0x3d10e631, 0x0add3804
1274 .word 0x3fa32348, 0xc7001000
1275 .word 0x3d0a5b6e, 0x42c7927d
1276 .word 0x3fa39e87, 0xb9feb000
1277 .word 0x3d1labf52, 0x02b64055
1278 .word 0x3fa419a9, 0x09593000
1279 .word 0x3d0ae6e3, 0x3ea4753a
1280 .word 0x3fa494ac, 0xc34d9000
1281 .word 0x3ce1c78a, 0x56fd2473
1282 .word 0x3fa50f92, 0xf60f9000
1283 .word 0x3d12d9f6, 0x1523ffc6
1284 .word 0x3fa58a5b, 0xafc8e000
1285 .word 0x3d035231, 0xaa3d4b1d
1286 .word 0x3fa60506, 0xfe98d000
1287 .word 0x3d1516fd, 0xf9ac7f28
1288 .word 0x3fa67f94, 0xf094b000
1289 .word 0x3d1b307c, 0xf9f93b5b
1290 .word 0x3fa6fa05, 0x93c7b000
1291 .word 0x3d0a0af2, 0x0eb1a504
1292 .word 0x3fa77458, 0xf632d000
1293 .word 0x3d19f88c, 0x69e543dd
1294 .word 0x3fa7ee8f, 0x25cd4000
1295 .word 0x3ce7bd3d, 0xcb47c2e4
1296 .word 0x3fa868a8, 0x3083f000
1297 .word 0x3d0b3b8b, 0xd96a72db
1298 .word 0x3fa8e2a4, 0x243a1000
1299 .word 0x3d173dd6, 0x0284c920
1300 .word 0x3fa95c83, 0xec8e000
1301 .word 0x3cff5beb, 0x41d00a41
1302 .word 0x3fa9d644, 0xfdffa000
1303 .word 0x3cf3c905, 0x39a473b6
1304 .word 0x3faa4fe9, 0xffa3d000
1305 .word 0x3cfla7b5, 0xfbfd6db2
1306 .word 0x3faac972, 0x21711000
1307 .word 0x3d1f1a7d, 0xe0264459
1308 .word 0x3fab42dd, 0x71197000

```

```

1309 .word 0x3cebec28, 0xd14c7d9f
1310 .word 0x3fabbc2b, 0xfc44f000
1311 .word 0x3d005cf2, 0xdd7d04a2
1312 .word 0x3fac355d, 0xd0921000
1313 .word 0x3d1e5999, 0x357f0710
1314 .word 0x3facae72, 0xfb95c000
1315 .word 0x3cf0540d, 0xfda4e418
1316 .word 0x3fad276b, 0x8adb0000
1317 .word 0x3d16a423, 0xc78a64b0
1318 .word 0x3fada047, 0x8be39000
1319 .word 0x3cf2963d, 0x8fb7f02b
1320 .word 0x3fae1907, 0x0c276000
1321 .word 0x3ca5b99b, 0x9d617a09
1322 .word 0x3fae91aa, 0x1914f000
1323 .word 0x3d10beaf, 0xf119cac5
1324 .word 0x3faf0a30, 0xc0116000
1325 .word 0x3cf5330b, 0xe64b8b77
1326 .word 0x3faf829b, 0x0e783000
1327 .word 0x3cf80267, 0xc7e09e3e
1328 .word 0x3faffae9, 0x119b9000
1329 .word 0x3cf819ba, 0x13162a9c
1330 .word 0x3fb0398d, 0x6b622000
1331 .word 0x3d153ac8, 0xd000cc01
1332 .word 0x3fb07598, 0x3598e000
1333 .word 0x3d11c4c0, 0x6d2999e2
1334 .word 0x3fb0b194, 0xee0d1000
1335 .word 0x3d199ba9, 0x3da7b72e
1336 .word 0x3fb0ed83, 0x9b552000
1337 .word 0x3d1bf82e, 0x4add5131
1338 .word 0x3fb12964, 0x4402e000
1339 .word 0x3d056224, 0x572ac464
1340 .word 0x3fb16536, 0xeea37000
1341 .word 0x3d25c1d0, 0xc4b82e7c
1342 .word 0x3fb1a0fb, 0xa1bf8000
1343 .word 0x3d24a3fc, 0xc319d6dc
1344 .word 0x3fb1dcb2, 0x63db1000
1345 .word 0x3d22889e, 0xbd3d1303
1346 .word 0x3fb2185b, 0x3b75a000
1347 .word 0x3cfce760, 0x70cdcfc5
1348 .word 0x3fb253f6, 0x2f0a1000
1349 .word 0x3d105be3, 0xeda69c04
1350 .word 0x3fb28f83, 0x450ed000
1351 .word 0x3d251aeb, 0x54232ed1
1352 .word 0x3fb2cb02, 0x83f5d000
1353 .word 0x3d2c3dc5, 0x94cae043
1354 .word 0x3fb30673, 0xf22c8000
1355 .word 0x3d24c9e2, 0x9dcf0ba5
1356 .word 0x3fb341d7, 0x961bd000
1357 .word 0x3cfd0929, 0x98376105
1358 .word 0x3fb37d2d, 0x76283000
1359 .word 0x3cfcfaab, 0x2400751e
1360 .word 0x3fb3b875, 0x98b1b000
1361 .word 0x3d1bb7d4, 0xd6a6b9db
1362 .word 0x3fb3f3b0, 0x04140000
1363 .word 0x3cee2474, 0xacdfcec5
1364 .word 0x3fb42edc, 0xbea64000
1365 .word 0x3d1bc0ee, 0xea7c9acd
1366 .word 0x3fb469fb, 0xcabb5000
1367 .word 0x3d26cc78, 0x9e4ae327
1368 .word 0x3fb4a50d, 0x3aa1b000
1369 .word 0x3cd003d9, 0xeed183bb
1370 .word 0x3fb4e011, 0x08a35000
1371 .word 0x3d25cb9f, 0xbe58b5c9
1372 .word 0x3fb51b07, 0x3f061000
1373 .word 0x3d207ed2, 0x4f1cd0d4
1374 .word 0x3fb555ef, 0xe40b5000

```

```

1375 .word 0x3ce692f1, 0x90d1c46b
1376 .word 0x3fb590ca, 0xfd01000
1377 .word 0x3d28509e, 0xae455754
1378 .word 0x3fb5cb98, 0x92ed4000
1379 .word 0x3d17be44, 0xa64fc52f
1380 .word 0x3fb60658, 0xa9375000
1381 .word 0x3ce8763b, 0xdd389ef2
1382 .word 0x3fb6410b, 0x46fe7000
1383 .word 0x3d256038, 0x61a13976
1384 .word 0x3fb67bb0, 0x726ec000
1385 .word 0x3cef724b, 0x69ef5912
1386 .word 0x3fb6b648, 0x31afe000
1387 .word 0x3d1033d7, 0xb22085b8
1388 .word 0x3fb6f0d2, 0x8ae56000
1389 .word 0x3d269737, 0xc93373da
1390 .word 0x3fb72b4f, 0x842ea000
1391 .word 0x3d21f666, 0x7fe6c45a
1392 .word 0x3fb765bf, 0x23a6b000
1393 .word 0x3d2c2687, 0xf9477b53
1394 .word 0x3fb7a021, 0x6f649000
1395 .word 0x3d2c2499, 0x430831ff
1396 .word 0x3fb7da76, 0x6d7b1000
1397 .word 0x3d066422, 0x240644d8
1398 .word 0x3fb814be, 0x23f8c000
1399 .word 0x3ccb2381, 0xda82fdfd
1400 .word 0x3fb84ef8, 0x98e82000
1401 .word 0x3d205465, 0xb72d106e
1402 .word 0x3fb88925, 0xd24fa000
1403 .word 0x3d2c55f5, 0x76088ff3
1404 .word 0x3fb8c345, 0xd6319000
1405 .word 0x3d2641eb, 0x596854cc
1406 .word 0x3fb8fd58, 0xaa8c2000
1407 .word 0x3cf136fe, 0x4348da4e
1408 .word 0x3fb9375e, 0x55595000
1409 .word 0x3d2dbb86, 0xe70186c9
1410 .word 0x3fb97156, 0xdc8f6000
1411 .word 0x3d0f01f3, 0x28123425
1412 .word 0x3fb9ab42, 0x46203000
1413 .word 0x3d0d66df, 0x661e3e7b
1414 .word 0x3fb9e520, 0x97f9c000
1415 .word 0x3d235fac, 0xb52dd050
1416 .word 0x3fbalef1, 0xd8061000
1417 .word 0x3d29a82e, 0xdbf2f796
1418 .word 0x3fba58b6, 0x0c2b2000
1419 .word 0x3d091c65, 0x1d1b06b1
1420 .word 0x3fba926d, 0x3a4ad000
1421 .word 0x3d158d94, 0x2f48aa71
1422 .word 0x3fbacc17, 0x68433000
1423 .word 0x3d0561f1, 0x7d2016d1
1424 .word 0x3fbb05b4, 0x9bee4000
1425 .word 0x3d0ff22c, 0x18f84a5e
1426 .word 0x3fbb3f44, 0xdb221000
1427 .word 0x3d2fa2a7, 0xb1bc135d
1428 .word 0x3fbb78c8, 0x2bb0e000
1429 .word 0x3d2b4210, 0x878cf032
1430 .word 0x3fbbb23e, 0x9368e000
1431 .word 0x3d22e9cf, 0x954c48ea
1432 .word 0x3fbbba8, 0x18146000
1433 .word 0x3d1d921d, 0x248382a6
1434 .word 0x3fbc2504, 0xbf79d000
1435 .word 0x3d1c5f13, 0x43bd2b70
1436 .word 0x3fbc5e54, 0x8f5bc000
1437 .word 0x3d1d0c57, 0x585f5e06
1438 .word 0x3fbc9797, 0x8d78e000
1439 .word 0x3d223fde, 0xd105cef9
1440 .word 0x3fbcd0cd, 0xbf8c1000

```

```

1441 .word 0x3d0f0a6d, 0xa86eba18
1442 .word 0x3fbd09f7, 0x2b4c4000
1443 .word 0x3d2048c0, 0x00354e33
1444 .word 0x3fbd4313, 0xd66cb000
1445 .word 0x3d0aeaf2, 0x1bb2a3b2
1446 .word 0x3fbd7c23, 0xc69cb000
1447 .word 0x3d0a046c, 0x8b35e23e
1448 .word 0x3fbd527, 0x0187d000
1449 .word 0x3d224ef0, 0xad5c303f
1450 .word 0x3fbdee1d, 0x8cd5e000
1451 .word 0x3d2ae4bf, 0x1ac200ee
1452 .word 0x3fbe2707, 0x6e2af000
1453 .word 0x3d072f4f, 0x543fff10
1454 .word 0x3fbe5fe4, 0xab272000
1455 .word 0x3d240a2c, 0x11600366
1456 .word 0x3fbe98b5, 0x49671000
1457 .word 0x3d119dd2, 0x27143a5b
1458 .word 0x3fbed179, 0x4e837000
1459 .word 0x3d20175e, 0x45b17dbe
1460 .word 0x3fbf0a30, 0xc0116000
1461 .word 0x3d05330b, 0xe64b8b77
1462 .word 0x3fbf42db, 0xa3a22000
1463 .word 0x3d29da91, 0x9a4127e6
1464 .word 0x3fbf7b79, 0xfec37000
1465 .word 0x3d2bbd9e, 0x05da04c0
1466 .word 0x3fbfb40b, 0xd6ff4000
1467 .word 0x3d2c0bec, 0xb7b53b5b
1468 .word 0x3fbfec91, 0x31dbe000
1469 .word 0x3d257554, 0x5ca333f2
1470 .word 0x3fc01285, 0x0a6df000
1471 .word 0x3d395e79, 0xadfe901b
1472 .word 0x3fc02ebb, 0x42bf3000
1473 .word 0x3d3a95c1, 0x68c7fc69
1474 .word 0x3fc04aeb, 0x449f6000
1475 .word 0x3d2afa90, 0x65ccd35c
1476 .word 0x3fc06715, 0x12ca5000
1477 .word 0x3d32dc54, 0x3191fae2
1478 .word 0x3fc08338, 0xaffa2000
1479 .word 0x3d30533c, 0xac823e27
1480 .word 0x3fc09f56, 0xlee71000
1481 .word 0x3d33867d, 0x4754172c
1482 .word 0x3fc0bb6d, 0x6247a000
1483 .word 0x3d35464f, 0x3ccd04b3
1484 .word 0x3fc0d77e, 0x7cd08000
1485 .word 0x3d3cb2cd, 0x2ee2f482
1486 .word 0x3fc0f389, 0x7134b000
1487 .word 0x3d02e530, 0xbb6149cf
1488 .word 0x3fc10f8e, 0x42253000
1489 .word 0x3d336263, 0xde634e7c
1490 .word 0x3fc12b8c, 0xf2518000
1491 .word 0x3d348a4a, 0x13c0a0fc
1492 .word 0x3fc14785, 0x84674000
1493 .word 0x3d156345, 0x1027c750
1494 .word 0x3fc16377, 0xf124000
1495 .word 0x3d091e1a, 0xbf41763e
1496 .word 0x3fc17f64, 0x58fca000
1497 .word 0x3d2843fa, 0xd093c8dc
1498 .word 0x3fc19b4a, 0xa0ced000
1499 .word 0x3d03bedb, 0x4ef663a7
1500 .word 0x3fc1b72a, 0xd52f6000
1501 .word 0x3d2e80a4, 0x1811a396
1502 .word 0x3fc1d304, 0xf8c35000
1503 .word 0x3d164aec, 0x82ebbf7
1504 .word 0x3fc1eed9, 0xe2dc000
1505 .word 0x3d161563, 0x7097648f
1506 .word 0x3fc20aa7, 0x18102000

```

```

1507 .word 0x3d3f2c94, 0x348552fe
1508 .word 0x3fc2266f, 0x190a5000
1509 .word 0x3d3596fa, 0xa3df8c05
1510 .word 0x3fc24231, 0x13ba5000
1511 .word 0x3cfc5ff8, 0x71162641
1512 .word 0x3fc25ded, 0x0abc6000
1513 .word 0x3d35a385, 0x4f176449
1514 .word 0x3fc279a3, 0x00ab4000
1515 .word 0x3d3ef432, 0xb3235108
1516 .word 0x3fc29552, 0xf81ff000
1517 .word 0x3d248d30, 0x1771c408
1518 .word 0x3fc2b0fc, 0xf3b1a000
1519 .word 0x3d177ca3, 0xe30a59ea
1520 .word 0x3fc2cca0, 0xf5f5f000
1521 .word 0x3d128439, 0xb9403b82
1522 .word 0x3fc2e83f, 0x0180d000
1523 .word 0x3cee7aa7, 0xaf63c632
1524 .word 0x3fc303d7, 0x18e47000
1525 .word 0x3d3fa5fd, 0x28c704d4
1526 .word 0x3fc31f69, 0x3eb19000
1527 .word 0x3d32cc6c, 0x8d2e3482
1528 .word 0x3fc33af5, 0x75770000
1529 .word 0x3d3c9ecc, 0xa2fe72a5
1530 .word 0x3fc3567b, 0xbfc22000
1531 .word 0x3d3250d2, 0x53991a1f
1532 .word 0x3fc371fc, 0x201e8000
1533 .word 0x3d3ee877, 0x9b2d8abc
1534 .word 0x3fc38d76, 0x99164000
1535 .word 0x3d1844a5, 0x9e39bb70
1536 .word 0x3fc3a8eb, 0x2d31a000
1537 .word 0x3d1bafb7, 0x7d5d503e
1538 .word 0x3fc3c459, 0xdef76000
1539 .word 0x3d3edc86, 0xf6b70d33
1540 .word 0x3fc3dfc2, 0xb0ecc000
1541 .word 0x3d28a72a, 0x62b8c13f
1542 .word 0x3fc3fb25, 0xa5952000
1543 .word 0x3d3195be, 0x6b358ff7
1544 .word 0x3fc41682, 0xbf727000
1545 .word 0x3d377fdc, 0x7bf03db2
1546 .word 0x3fc431da, 0x01050000
1547 .word 0x3d304837, 0x836e0391
1548 .word 0x3fc44d2b, 0x6ccb7000
1549 .word 0x3d3a3ccf, 0xa7b2a1f1
1550 .word 0x3fc46877, 0x0542f000
1551 .word 0x3d03f5d0, 0x3957bc10
1552 .word 0x3fc483bc, 0xcce6e000
1553 .word 0x3d1lea52, 0x723f6369
1554 .word 0x3fc49efc, 0xc6313000
1555 .word 0x3d3cde14, 0xcc15551b
1556 .word 0x3fc4ba36, 0xf39a5000
1557 .word 0x3d279568, 0x981bcc36
1558 .word 0x3fc4d56b, 0x5798e000
1559 .word 0x3d380580, 0x15a96555
1560 .word 0x3fc4f099, 0xf4a23000
1561 .word 0x3cf640d0, 0x50150d92
1562 .word 0x3fc50bc2, 0xcd29c000
1563 .word 0x3d1lada57, 0x28db8d4f
1564 .word 0x3fc526e5, 0xe3a1b000
1565 .word 0x3d20de8b, 0x90075b8f
1566 .word 0x3fc54203, 0x3a7a8000
1567 .word 0x3d268d68, 0xed855f0e
1568 .word 0x3fc55d1a, 0xd4232000
1569 .word 0x3d3add94, 0xdda647e8
1570 .word 0x3fc5782c, 0xb3091000
1571 .word 0x3d28b739, 0x5d0d777d
1572 .word 0x3fc59338, 0xd9982000

```

```

1573 .word 0x3cf0ba68, 0xb7555d4a
1574 .word 0x3fc5ae3f, 0x4a3aa000
1575 .word 0x3d21ea25, 0xf012a8b9
1576 .word 0x3fc5c940, 0x07597000
1577 .word 0x3d15c9ad, 0xccb7337a
1578 .word 0x3fc5e43b, 0x135bd000
1579 .word 0x3d278a96, 0x6224c79e
1580 .word 0x3fc5ff30, 0x70a79000
1581 .word 0x3d1e9e43, 0x9f105039
1582 .word 0x3fc61a20, 0x21a0e000
1583 .word 0x3d3dd9dd, 0x1bdf3cdd
1584 .word 0x3fc6350a, 0x28aaa000
1585 .word 0x3d2d5ec0, 0xab8163af
1586 .word 0x3fc64fee, 0x8825f000
1587 .word 0x3d3896fc, 0xa298884b
1588 .word 0x3fc66acd, 0x4272a000
1589 .word 0x3d3aa1bd, 0xbf6c785
1590 .word 0x3fc685a6, 0x59eef000
1591 .word 0x3d3706ab, 0x49f7e6f6
1592 .word 0x3fc6a079, 0xd0f7a000
1593 .word 0x3d35a3f8, 0x448d14f5
1594 .word 0x3fc6bb47, 0xa9e80000
1595 .word 0x3d19f64d, 0x23ea3296
1596 .word 0x3fc6d60f, 0xe719d000
1597 .word 0x3d10e46a, 0xa3b2e266
1598 .word 0x3fc6f0d2, 0x8ae56000
1599 .word 0x3d369737, 0xc93373da
1600 .word 0x3fc70b8f, 0x97a1a000
1601 .word 0x3d34ea64, 0xf6a95bef
1602 .word 0x3fc72647, 0x0fa3f000
1603 .word 0x3d211641, 0xe3178b76
1604 .word 0x3fc740f8, 0xf5403000
1605 .word 0x3d2e9326, 0xcdfceabe
1606 .word 0x3fc75ba5, 0x4ac8e000
1607 .word 0x3d3ddca5, 0x8bc4a7c0
1608 .word 0x3fc7764c, 0x128f2000
1609 .word 0x3d027490, 0x3479e3d1
1610 .word 0x3fc790ed, 0x4ee26000
1611 .word 0x3d199bbd, 0x4e7746f6
1612 .word 0x3fc7ab89, 0x0210d000
1613 .word 0x3d321237, 0xc6d65ad4
1614 .word 0x3fc7c61f, 0x2e673000
1615 .word 0x3d2b8da4, 0x99c82e40
1616 .word 0x3fc7e0af, 0xd630c000
1617 .word 0x3d139e7c, 0x1d8f1034
1618 .word 0x3fc7fb3a, 0xfbb75000
1619 .word 0x3d204815, 0xb73ec551
1620 .word 0x3fc815c0, 0xa1435000
1621 .word 0x3d2fab5a, 0x0dbfc630
1622 .word 0x3fc83040, 0xc91bc000
1623 .word 0x3d3e5b71, 0xc6e66f32
1624 .word 0x3fc84abb, 0x75865000
1625 .word 0x3d0392a9, 0x058ea173
1626 .word 0x3fc86530, 0xa8c70000
1627 .word 0x3d398bb0, 0xc4ea3e3
1628 .word 0x3fc87fa0, 0x6520c000
1629 .word 0x3d322120, 0x401202fc
1630 .word 0x3fc89a0a, 0xacd4e000
1631 .word 0x3d2c0bfb, 0xda8f5a72
1632 .word 0x3fc8b46f, 0x82236000
1633 .word 0x3d12d9f2, 0x102dd7c9
1634 .word 0x3fc8cece, 0xe74ad000
1635 .word 0x3d16917d, 0x56f5912d
1636 .word 0x3fc8e928, 0xde886000
1637 .word 0x3d3a8154, 0xb13d72d5
1638 .word 0x3fc9037d, 0x6a180000

```

```

1639 .word 0x3d230dea, 0x57c1c8d9
1640 .word 0x3fc91dcc, 0x8c340000
1641 .word 0x3d37bc6a, 0xbdddef46
1642 .word 0x3fc93816, 0x47159000
1643 .word 0x3d267385, 0x2b8b8c4f
1644 .word 0x3fc9525a, 0x9cf45000
1645 .word 0x3d2ad1d9, 0x04c1d4e3
1646 .word 0x3fc96c99, 0x9006a000
1647 .word 0x3d2a88d5, 0x9cbb452c
1648 .word 0x3fc986d3, 0x22818000
1649 .word 0x3cf93b56, 0x4dd44000
1650 .word 0x3fc9a107, 0x56988000
1651 .word 0x3d264aa6, 0x242cd098
1652 .word 0x3fc9bb36, 0x2e7df000
1653 .word 0x3d3706ab, 0xaf18f802
1654 .word 0x3fc9d55f, 0xac62d000
1655 .word 0x3ce732c0, 0x789487af
1656 .word 0x3fc9ef83, 0xd2769000
1657 .word 0x3d3467a4, 0x26031900
1658 .word 0x3fca09a2, 0xa2e79000
1659 .word 0x3d311331, 0x195f76e6
1660 .word 0x3fca23bc, 0x1fe2b000
1661 .word 0x3d258c64, 0xdc46c1ea
1662 .word 0x3fca3dd0, 0x4b938000
1663 .word 0x3d297da1, 0x366e2c5a
1664 .word 0x3fca57df, 0x28244000
1665 .word 0x3d3b99c8, 0xca1d9abb
1666 .word 0x3fca71e8, 0xb7bdf000
1667 .word 0x3d377a9a, 0xc887d66f
1668 .word 0x3fca8bec, 0xfc882000
1669 .word 0x3d3e3185, 0xcf21b9cf
1670 .word 0x3fcaa5eb, 0xf8a93000
1671 .word 0x3d2abead, 0x92d5cae2
1672 .word 0x3fcabfe5, 0xae461000
1673 .word 0x3d125c2b, 0x1a83b18e
1674 .word 0x3fcad9da, 0x1f827000
1675 .word 0x3d1df520, 0xdf03e6e
1676 .word 0x3fcacf3c9, 0x4e80b000
1677 .word 0x3d3fe5b1, 0x9cc03270
1678 .word 0x3fcb0db3, 0x3d620000
1679 .word 0x3d3fee14, 0x38eab906
1680 .word 0x3fcb2797, 0xee463000
1681 .word 0x3d105dd5, 0xbe4bfd5c
1682 .word 0x3fcb4177, 0x634ba000
1683 .word 0x3d355d01, 0x5666069f
1684 .word 0x3fcb5b51, 0x9e8fb000
1685 .word 0x3d2691ba, 0x27fdc19e
1686 .word 0x3fcb7526, 0xa22e4000
1687 .word 0x3d2c0dbf, 0x2e785490
1688 .word 0x3fcb8ef6, 0x70420000
1689 .word 0x3d387533, 0x321788e0
1690 .word 0x3fcb8c1, 0xae46000
1691 .word 0x3d3a32e2, 0x9eee9d85
1692 .word 0x3fcbcc286, 0x742d8000
1693 .word 0x3d39ac53, 0xf39d121c
1694 .word 0x3fcbdc46, 0xae344000
1695 .word 0x3d3625b4, 0x023d6505
1696 .word 0x3fcbf601, 0xbb0e4000
1697 .word 0x3d2386a9, 0x47c378b5
1698 .word 0x3fcc0fb7, 0x9ccfd000
1699 .word 0x3d272000, 0xcc2eb551
1700 .word 0x3fcc2968, 0x558c1000
1701 .word 0x3d318146, 0x108e3ae0
1702 .word 0x3fcc4313, 0xe754e000
1703 .word 0x3d3279be, 0x74cad7d6
1704 .word 0x3fcc5cba, 0x543ae000

```

```

1705 .word 0x3d20929d, 0xecb454fc
1706 .word 0x3fcc765b, 0x9e4d6000
1707 .word 0x3d31ab6b, 0x36976f6c
1708 .word 0x3fcc8ff7, 0xc79a9000
1709 .word 0x3d344358, 0x4bb03de6
1710 .word 0x3fcc98e, 0xd22f5000
1711 .word 0x3d3e9673, 0xe735df63
1712 .word 0x3fccc320, 0xc0176000
1713 .word 0x3d240903, 0x9a653794
1714 .word 0x3fccdcad, 0x935d1000
1715 .word 0x3d3cbe01, 0xf966cb77
1716 .word 0x3fccf635, 0x4e09c000
1717 .word 0x3d277123, 0x9a07d55b
1718 .word 0x3fcd0fb7, 0xf2255000
1719 .word 0x3d3ca15a, 0x9bf3989b
1720 .word 0x3fcd2935, 0x81b6b000
1721 .word 0x3d1f363f, 0xb5d55685
1722 .word 0x3fcd42ad, 0xfec35000
1723 .word 0x3d3a28ff, 0xc09fef63
1724 .word 0x3fcd5c21, 0x6b4fb000
1725 .word 0x3d3722b7, 0x221acb2f
1726 .word 0x3fcd758f, 0xc95ef000
1727 .word 0x3d3a97bd, 0x5d2fa755
1728 .word 0x3fcd8ef9, 0x1af31000
1729 .word 0x3d3abbe8, 0x0f26ce1f
1730 .word 0x3fcd85d, 0x620ce000
1731 .word 0x3d240194, 0xc16cc7ec
1732 .word 0x3fcdclbc, 0xa0abe000
1733 .word 0x3d38fac1, 0xa628ccc6
1734 .word 0x3fcddeb16, 0xd8ce9000
1735 .word 0x3d384421, 0xa3bed1d1
1736 .word 0x3fcdcf46c, 0x0c722000
1737 .word 0x3d3a5e82, 0xb0b79039
1738 .word 0x3fce0dbc, 0x3d92a000
1739 .word 0x3d359233, 0xf0529bf1
1740 .word 0x3fce2707, 0x6e2af000
1741 .word 0x3d172f4f, 0x543fff10
1742 .word 0x3fce404d, 0xa034b000
1743 .word 0x3d2cf022, 0x3ecbb0ce
1744 .word 0x3fce598e, 0xd5a87000
1745 .word 0x3d3c5d96, 0x861c2cec
1746 .word 0x3fce72cb, 0x107da000
1747 .word 0x3d1dd48c, 0xcdf5471c
1748 .word 0x3fce8c02, 0x52aa5000
1749 .word 0x3d34bfd2, 0x3f8b8c80
1750 .word 0x3fcea534, 0x9e23a000
1751 .word 0x3d381b93, 0x4c73ccb5
1752 .word 0x3fcebe61, 0xf4dd7000
1753 .word 0x3d3615d6, 0x67811ada
1754 .word 0x3fced78a, 0x58ca8000
1755 .word 0x3d16f1b5, 0x3793387e
1756 .word 0x3fcef0ad, 0xcbbdc5000
1757 .word 0x3d326ca4, 0x31bca86e
1758 .word 0x3fcf09cc, 0x50036000
1759 .word 0x3d3da094, 0x18d999db
1760 .word 0x3fcf22e5, 0xe72f1000
1761 .word 0x3ce7561d, 0x7d037c19
1762 .word 0x3fcf3bfa, 0x934d6000
1763 .word 0x3d2d9f2a, 0x937b903b
1764 .word 0x3fcf550a, 0x564b7000
1765 .word 0x3d366e0e, 0x2fb6fe81
1766 .word 0x3fcf6e15, 0x32153000
1767 .word 0x3d0b2b44, 0x29d89c5c
1768 .word 0x3fcf871b, 0x28955000
1769 .word 0x3ce14052, 0xb5b2204b
1770 .word 0x3fcfa01c, 0x3bb57000

```

```

1771 .word 0x3d397823, 0x81478a1f
1772 .word 0x3fcfb918, 0x6d5e3000
1773 .word 0x3d3c551a, 0xaa8cd86f
1774 .word 0x3fcfd20f, 0xbf76f000
1775 .word 0x3d3b8ea9, 0x234e4064
1776 .word 0x3fcfeb02, 0x33e60000
1777 .word 0x3d2f316e, 0x32d5e8c7
1778 .word 0x3fd001f7, 0xe6484000
1779 .word 0x3d38a957, 0x40c9abbc
1780 .word 0x3fd00e6c, 0x45ad5000
1781 .word 0x3cdcc68d, 0x52e01203
1782 .word 0x3fd01ade, 0x39139000
1783 .word 0x3d4deed9, 0xe6647d5c
1784 .word 0x3fd0274d, 0xc16c2000
1785 .word 0x3d2979e8, 0x9c835c2
1786 .word 0x3fd033ba, 0xdfa74000
1787 .word 0x3d0c30bc, 0x1485bdfc
1788 .word 0x3fd04025, 0x94b4d000
1789 .word 0x3cf036b8, 0x9ef42d7f
1790 .word 0x3fd04c8d, 0xe1841000
1791 .word 0x3d4c0328, 0xb5da628f
1792 .word 0x3fd058f3, 0xc703e000
1793 .word 0x3d478bcc, 0xa196e4a9
1794 .word 0x3fd06557, 0x46227000
1795 .word 0x3d0131df, 0xb4868d6a
1796 .word 0x3fd071b8, 0x5fcd5000
1797 .word 0x3d421a3a, 0x2e0ff2f8
1798 .word 0x3fd07e17, 0x14f1c000
1799 .word 0x3d40819c, 0xd863da16
1800 .word 0x3fd08a73, 0x667c5000
1801 .word 0x3d3ebc1d, 0x40c5a329
1802 .word 0x3fd096cd, 0x55591000
1803 .word 0x3d3f998d, 0x20550a31
1804 .word 0x3fd0a324, 0xe2739000
1805 .word 0x3d0c6bee, 0x7ef4030e
1806 .word 0x3fd0af7a, 0x0eb6c000
1807 .word 0x3d23ccf9, 0x4945adaa
1808 .word 0x3fd0bbcc, 0xdb0d2000
1809 .word 0x3d32f32c, 0xcc5dcdfb
1810 .word 0x3fd0c81d, 0x4860a000
1811 .word 0x3d40d218, 0x5ff17467
1812 .word 0x3fd0d46b, 0x579ab000
1813 .word 0x3d3d2c81, 0xf640e1e6
1814 .word 0x3fd0e0b7, 0x09a43000
1815 .word 0x3d32a038, 0xa7862f2a
1816 .word 0x3fd0ed00, 0x5f657000
1817 .word 0x3d4b48e2, 0xb5e955ff
1818 .word 0x3fd0f947, 0x59c66000
1819 .word 0x3d4356cf, 0x407bf3a5
1820 .word 0x3fd1058b, 0xf9ae4000
1821 .word 0x3d45aa31, 0x3f415699
1822 .word 0x3fd111ce, 0x4003e000
1823 .word 0x3d4c99b9, 0x1ed29693
1824 .word 0x3fd11e0e, 0x2dad9000
1825 .word 0x3d496e01, 0xdc0cc691
1826 .word 0x3fd12a4b, 0xc3911000
1827 .word 0x3d452c57, 0xcf5c66d4
1828 .word 0x3fd13687, 0x0293a000
1829 .word 0x3d4160bd, 0xb314c76f
1830 .word 0x3fd142bf, 0xeb9a0000
1831 .word 0x3d31ce61, 0x85b58a9e
1832 .word 0x3fd14ef6, 0x7f886000
1833 .word 0x3d40b42c, 0xd101b436
1834 .word 0x3fd15b2a, 0xbf428000
1835 .word 0x3d489c71, 0x2d927594
1836 .word 0x3fd1675c, 0xababa000

```

```

1837 .word 0x3d38380e, 0x731f55c4
1838 .word 0x3fd1738c, 0x45a66000
1839 .word 0x3d431c8b, 0x7fe69f45
1840 .word 0x3fd17fb9, 0x8e150000
1841 .word 0x3d42baba, 0x2c5aeebe
1842 .word 0x3fd18be4, 0x85d93000
1843 .word 0x3d3c167f, 0x6f3604ab
1844 .word 0x3fd1980d, 0x2dd442000
1845 .word 0x3d2b7b3a, 0x7a361c9a
1846 .word 0x3fd1a433, 0x86e67000
1847 .word 0x3d4e857a, 0xf9cb1f55
1848 .word 0x3fd1b057, 0x91f07000
1849 .word 0x3d46915c, 0xc91d50e9
1850 .word 0x3fd1bc79, 0x4fd1c000
1851 .word 0x3d419879, 0xc5c22c21
1852 .word 0x3fd1c898, 0xc1699000
1853 .word 0x3d43f5f7, 0x8d1cea80
1854 .word 0x3fd1d4b5, 0xe796a000
1855 .word 0x3d222a5b, 0xd197bac2
1856 .word 0x3fd1e0d0, 0xc3371000
1857 .word 0x3d3af8f2, 0xa9b0d4a0
1858 .word 0x3fd1ece9, 0x5528a000
1859 .word 0x3d4cf630, 0x9ec96b89
1860 .word 0x3fd1f8ff, 0x9e48a000
1861 .word 0x3d27946c, 0x040cbe77
1862 .word 0x3fd20513, 0x9f73b000
1863 .word 0x3cf6e15e, 0x1609e0a4
1864 .word 0x3fd21125, 0x59861000
1865 .word 0x3d382e78, 0xba2950c4
1866 .word 0x3fd21d34, 0xcd5b9000
1867 .word 0x3d3b552f, 0xb28badaa
1868 .word 0x3fd22941, 0xfbcf7000
1869 .word 0x3d42cb44, 0x850a7b4f
1870 .word 0x3fd2354c, 0xe5bc8000
1871 .word 0x3d414389, 0x7cfeacce
1872 .word 0x3fd24155, 0x8bfd1000
1873 .word 0x3d300fff, 0x3228fcad
1874 .word 0x3fd24d5b, 0xef6ae000
1875 .word 0x3d44ff114, 0x3f81b02a
1876 .word 0x3fd25960, 0x10df7000
1877 .word 0x3d38e7bc, 0x224ea3e3
1878 .word 0x3fd26561, 0xf1338000
1879 .word 0x3d38b488, 0x66faa45f
1880 .word 0x3fd27161, 0x913f8000
1881 .word 0x3d34f4f1, 0xf61564b4
1882 .word 0x3fd27d5e, 0xf1db5000
1883 .word 0x3d4e6dc8, 0xb8735361
1884 .word 0x3fd2895a, 0x13de8000
1885 .word 0x3d3a8d7a, 0xd24c13f0
1886 .word 0x3fd29552, 0xf81ff000
1887 .word 0x3d348d30, 0x1771c408
1888 .word 0x3fd2a149, 0x9f762000
1889 .word 0x3d479220, 0x57062a92
1890 .word 0x3fd2ad3e, 0x0ab73000
1891 .word 0x3d2b972e, 0x488c359f
1892 .word 0x3fd2b930, 0x3ab89000
1893 .word 0x3d4a493b, 0x4a5013d7
1894 .word 0x3fd2c520, 0x304f8000
1895 .word 0x3d230852, 0x8c342f39
1896 .word 0x3fd2d10d, 0xec508000
1897 .word 0x3d360c61, 0xf7088353
1898 .word 0x3fd2dcf9, 0x6f8fd000
1899 .word 0x3d20b4a2, 0x8e33c9ce
1900 .word 0x3fd2e8e2, 0xbae11000
1901 .word 0x3d4a6138, 0x5992350a
1902 .word 0x3fd2f4c9, 0xcf17a000

```

```

1903 .word 0x3d371f04, 0x9374b87b
1904 .word 0x3fd300ae, 0xad063000
1905 .word 0x3d342f56, 0x8b75fcac
1906 .word 0x3fd30c91, 0x557f1000
1907 .word 0x3d4d7ad4, 0xebd75d15
1908 .word 0x3fd31871, 0xc9544000
1909 .word 0x3d184fab, 0x94cecf9
1910 .word 0x3fd32450, 0x09570000
1911 .word 0x3d3d271b, 0x9bdae59d
1912 .word 0x3fd3302c, 0x16586000
1913 .word 0x3d36217d, 0xc2a3e08b
1914 .word 0x3fd33c05, 0xf128d000
1915 .word 0x3d4b51be, 0x71fc7961
1916 .word 0x3fd347dd, 0x9a987000
1917 .word 0x3d4aa9ac, 0x8ace9fdc
1918 .word 0x3fd353b3, 0x1376d000
1919 .word 0x3d4d99ca, 0x0327b24d
1920 .word 0x3fd35f86, 0x5c932000
1921 .word 0x3d427c10, 0xd8af2d5b
1922 .word 0x3fd36b57, 0x76bc1000
1923 .word 0x3d116978, 0x5a9c223f
1924 .word 0x3fd37726, 0x62bfd000
1925 .word 0x3d40b5e4, 0xa9d627ef
1926 .word 0x3fd382f3, 0x216c4000
1927 .word 0x3d4df3c5, 0xbc5cb012
1928 .word 0x3fd38ebd, 0xb38ed000
1929 .word 0x3d290582, 0xe67d4ca0
1930 .word 0x3fd39a86, 0x19f45000
1931 .word 0x3d18ee51, 0x937354f5
1932 .word 0x3fd3a64c, 0x55694000
1933 .word 0x3d37a71c, 0xbcd735d0
1934 .word 0x3fd3b210, 0x66b9b000
1935 .word 0x3d461f09, 0x33f754f9
1936 .word 0x3fd3bd22, 0x4eb14000
1937 .word 0x3d46d425, 0xb478c893
1938 .word 0x3fd3c992, 0x0e1b2000
1939 .word 0x3d141c28, 0xaa680b76
1940 .word 0x3fd3d54f, 0xa5e1f000
1941 .word 0x3d3c3e1c, 0xd9a395e3
1942 .word 0x3fd3e10b, 0x16701000
1943 .word 0x3d3f3bcf, 0x145429c7
1944 .word 0x3fd3ecc4, 0x60ef5000
1945 .word 0x3d4e9fd7, 0x9d83ecff
1946 .word 0x3fd3f87b, 0x86093000
1947 .word 0x3d451014, 0x55d3b3bc
1948 .word 0x3fd40430, 0x8686a000
1949 .word 0x3d3f8ef4, 0x3049f7d3
1950 .word 0x3fd40fe3, 0x63303000
1951 .word 0x3d3e5c5f, 0xe79f05c6
1952 .word 0x3fd41b94, 0x1cce0000
1953 .word 0x3d47dcb7, 0xf60de01c
1954 .word 0x3fd42742, 0xb427d000
1955 .word 0x3d433c6c, 0x7ea3ecc5
1956 .word 0x3fd432ef, 0x2a04e000
1957 .word 0x3d40276b, 0x3674752a
1958 .word 0x3fd43e99, 0x7f2c1000
1959 .word 0x3d1c3f72, 0x40c41a04
1960 .word 0x3fd44a41, 0xb463c000
1961 .word 0x3d31ee28, 0xf37cf612
1962 .word 0x3fd455e7, 0xca720000
1963 .word 0x3d1ad8c6, 0x36629aed
1964 .word 0x3fd4618b, 0xc21c5000
1965 .word 0x3d4d84fa, 0x16f66f66
1966 .word 0x3fd46d2d, 0x9c280000
1967 .word 0x3d359b27, 0x5f67f75a
1968 .word 0x3fd478cd, 0x5959b000

```

```

1969 .word 0x3d2ec89b, 0xf0c8d098
1970 .word 0x3fd44846a, 0xfa75b000
1971 .word 0x3d4a7057, 0x47219c8d
1972 .word 0x3fd49006, 0x80400000
1973 .word 0x3d43a198, 0x00f2f83a
1974 .word 0x3fd49b9f, 0xeb7c1000
1975 .word 0x3d3dac1c, 0x58ab60d7
1976 .word 0x3fd4a737, 0x3cecf000
1977 .word 0x3d432ee5, 0x8a0655db
1978 .word 0x3fd4b2cc, 0x75555000
1979 .word 0x3d43f81a, 0x1c3a02db
1980 .word 0x3fd4be5f, 0x95777000
1981 .word 0x3d4141b6, 0x993293ee
1982 .word 0x3fd4c9f0, 0x9e152000
1983 .word 0x3d487888, 0x63c7f488
1984 .word 0x3fd4d57f, 0x8fefef00
1985 .word 0x3d23f926, 0x7fd06868
1986 .word 0x3fd4e10c, 0x6bc8a000
1987 .word 0x3cf8283f, 0x1636f061
1988 .word 0x3fd4ec97, 0x32600000
1989 .word 0x3d234d7a, 0xaf04d104
1990 .word 0x3fd44f81f, 0xe4763000
1991 .word 0x3d4a00c2, 0x6f2c03dd
1992 .word 0x3fd503a6, 0x82cb1000
1993 .word 0x3d4965cd, 0xc3a41929
1994 .word 0x3fd50f2b, 0x0e1e0000
1995 .word 0x3d3a0940, 0x8c47b8d8
1996 .word 0x3fd51aad, 0x872df000
1997 .word 0x3d405a13, 0x927ac19f
1998 .word 0x3fd5262d, 0xeeb98000
1999 .word 0x3d40f230, 0x47bb5b00
2000 .word 0x3fd531ac, 0x457ee000
2001 .word 0x3d3df83b, 0x7d931501
2002 .word 0x3fd53d28, 0x8c3bd000
2003 .word 0x3d4dd48d, 0x029240a7
2004 .word 0x3fd548a2, 0xc3add000
2005 .word 0x3d23167e, 0x63081cf7
2006 .word 0x3fd5541a, 0xec91b000
2007 .word 0x3d4f3f4a, 0xa91c688a
2008 .word 0x3fd55f91, 0x07a43000
2009 .word 0x3d4dc337, 0x10e416b4
2010 .word 0x3fd56b05, 0x15a18000
2011 .word 0x3d29247b, 0xbc4a23fc
2012 .word 0x3fd57677, 0x17455000
2013 .word 0x3d44d8a9, 0x356d941b
2014 .word 0x3fd581e7, 0x0d4b2000
2015 .word 0x3d4c19c3, 0xc9da4e1c
2016 .word 0x3fd58d54, 0xf86e0000
2017 .word 0x3d2791f3, 0x0a795215
2018 .word 0x3fd598c0, 0xd9687000
2019 .word 0x3d43d05b, 0x4793492e
2020 .word 0x3fd5a42a, 0xb0f4c000
2021 .word 0x3d4fc338, 0xala4108b
2022 .word 0x3fd5af92, 0x7fccd000
2023 .word 0x3d4c7f9a, 0x01400711
2024 .word 0x3fd5baf8, 0x46aa1000
2025 .word 0x3d46328b, 0x83c602e0
2026 .word 0x3fd5c65c, 0x06459000
2027 .word 0x3d4300fc, 0xff3f88cd
2028 .word 0x3fd5d1bd, 0xbf580000
2029 .word 0x3d4394a1, 0x1b1c1ee4
2030 .word 0x3fd5dd1d, 0x7299b000
2031 .word 0x3d43a84f, 0x3bf518f5
2032 .word 0x3fd5e87b, 0x20c29000
2033 .word 0x3d3527d1, 0x8f7738fa
2034 .word 0x3fd5f3d6, 0xca8a2000

```

```

2035 .word 0x3d37af84, 0x8e19cc75
2036 .word 0x3fd5ff30, 0x70a79000
2037 .word 0x3d2e9e43, 0x9f105039
2038 .word 0x3fd60a88, 0x13d1a000
2039 .word 0x3d36e9b9, 0xc879af55
2040 .word 0x3fd615dd, 0xb4bec000
2041 .word 0x3d13c7ca, 0x90bc04b2
2042 .word 0x3fd62131, 0x5424e000
2043 .word 0x3d463e81, 0xdaacbcc
2044 .word 0x3fd62c82, 0xf2b9c000
2045 .word 0x3d3e54bd, 0xbd7c8a98
2046 .word 0x3fd637d2, 0x91329000
2047 .word 0x3d450450, 0x865165ea
2048 .word 0x3fd64320, 0x30444000
2049 .word 0x3d3efe02, 0x7a01d7df
2050 .word 0x3fd64e6b, 0xd0a35000
2051 .word 0x3d2afe80, 0x69d61295
2052 .word 0x3fd659b5, 0x7303e000
2053 .word 0x3d1f281d, 0xb0af8efc
2054 .word 0x3fd664fd, 0x1819b000
2055 .word 0x3d418e55, 0xe463b5fe
2056 .word 0x3fd67042, 0xc0983000
2057 .word 0x3d4c6148, 0xdbdcf10d
2058 .word 0x3fd67b86, 0x6d327000
2059 .word 0x3d438fd6, 0x3ea11c64
2060 .word 0x3fd686c8, 0x1e9b1000
2061 .word 0x3d32bb11, 0x0af84054
2062 .word 0x3fd69207, 0xd5845000
2063 .word 0x3d43a44f, 0x4861e4ab
2064 .word 0x3fd69d45, 0x92a03000
2065 .word 0x3d38b1bd, 0xbf97ffa6
2066 .word 0x3fd6a881, 0x56a03000
2067 .word 0x3d420e9b, 0xd9d37351
2068 .word 0x3fd6b3bb, 0x22359000
2069 .word 0x3d30f625, 0x7a933268
2070 .word 0x3fd6bef2, 0xf6111000
2071 .word 0x3d48f8fc, 0x947d5965
2072 .word 0x3fd6ca28, 0xd2e34000
2073 .word 0x3d430ad0, 0xb8c49166
2074 .word 0x3fd6d55c, 0xb95c3000
2075 .word 0x3d39b9c8, 0xae9a6ee2
2076 .word 0x3fd6e08e, 0xaa2ba000
2077 .word 0x3d1e38c1, 0x39318d71
2078 .word 0x3fd6ebbe, 0xa600e000
2079 .word 0x3d4cce14, 0xc7dd17dd
2080 .word 0x3fd6f6ec, 0xad8b2000
2081 .word 0x3d249058, 0xdf08376
2082 .word 0x3fd70218, 0xc178e000
2083 .word 0x3d42a947, 0x0e225428
2084 .word 0x3fd70d42, 0xe2789000
2085 .word 0x3d21aeaa, 0x337ee287
2086 .word 0x3fd7186b, 0x11381000
2087 .word 0x3d1934e2, 0x677d272b
2088 .word 0x3fd72391, 0x4e650000
2089 .word 0x3d0c1d52, 0xbdc87d8a
2090 .word 0x3fd72eb5, 0x9aac9000
2091 .word 0x3d4dd010, 0xd08a7a15
2092 !! TBL - end

2094 ! constants:
2095 .align 64
2096 CONSTANTS:
2097 .word 0x40000000,0x00000000
2098 .word 0x3fe55555,0x555571da
2099 .word 0x3fd99999,0x8702be3a
2100 .word 0x3fd24af7,0x3f4569b1

```



```

2101 .word 0x3ea62e42,0xfee00000 ! scaled by 2**(-20
2102 .word 0x3caa39ef,0x35793c76 ! scaled by 2**(-20
2103 .word 0xfffffc00,0x00000000 ! ELEVENBIT
2104 .word 0x43200000
2105 .word 0xfff00000
2106 .word 0xc0190200 ! ELEVENBIT
2107 .word 0x0200 ! ELEVENBIT

```

```

2109 #define two 0x00
2110 #define A1 0x08
2111 #define A2 0x10
2112 #define A3 0x18
2113 #define ln2hi 0x20
2114 #define ln2lo 0x28
2115 #define mask 0x30
2116 #define ox43200000 0x38
2117 #define oxfff00000 0x3c
2118 #define ox0194000 0x40
2119 #define ox4000 0x44

```

2122 ! local storage indices

```

2124 #define jnk STACK_BIAS-0x8
2125 #define tmp2 STACK_BIAS-0x10
2126 #define tmp1 STACK_BIAS-0x18
2127 #define tmp0 STACK_BIAS-0x20
2128 #define tmp3 STACK_BIAS-0x28
2129 #define tmp4 STACK_BIAS-0x30
2130 #define tmp5 STACK_BIAS-0x38
2131 #define tmp6 STACK_BIAS-0x40
2132 ! sizeof temp storage - must be a multiple of 16 for V9
2133 #define tmps 0x40

```

2135 ! register use

```

2137 ! i0 n
2138 ! i1 x
2139 ! i2 stridex
2140 ! i3 y
2141 ! i4 stridey
2142 ! i5

```

2144 ! g1 TBL

```

2146 ! l0 j0
2147 ! l1 j1
2148 ! l2 j2
2149 ! l3
2150 ! l4 0x94000
2151 ! l5 CONSTANTS
2152 ! l6 0x000fffff
2153 ! l7 0x7ff00000

```

```

2155 ! o0 py0
2156 ! o1 py1
2157 ! o2 py2
2158 ! o3 used in primary range bounds check
2159 ! o4 used in primary range bounds check
2160 ! o5 used in .range1 check section as temporary
2161 ! o7 NOT USED

```

```

2163 ! f0 u0,q0
2164 ! f2 v0,(two-v0)-u0,z0
2165 ! f4 n0,f0,q0
2166 ! f6 s0

```

```

2167 ! f8 q
2168 ! f10 u1,q1
2169 ! f12 v1,(two-v1)-u1,z1
2170 ! f14 n1,f1,q1
2171 ! f16 s1
2172 ! f18 t ! now tmp0 storage
2173 ! f20 u2,q2
2174 ! f22 v2,(two-v2)-u2,q2
2175 ! f24 n2,f2,q2
2176 ! f26 s2
2177 ! f28 0xffff00000
2178 ! f29 0x43200000
2179 ! f30 0x4000
2180 ! f31 0xc0194000
2181 ! f32 t0
2182 ! f34 h0,f0-(c0-h0)
2183 ! f36 c0
2184 ! f38 A1
2185 ! f40 two
2186 ! f42 t1
2187 ! f44 h1,f1-(c1-h1)
2188 ! f46 c1
2189 ! f48 A2
2190 ! f50 0xffff8000... or 0xfffffc00 for 6 or 11 bit tbl resp
2191 ! f52 t2
2192 ! f54 h2,f2-(c2-h2)
2193 ! f56 c2
2194 ! f58 A3 now tmp1 storage
2195 ! f60 ln2hi
2196 ! f62 ln2lo
2197 ! -----
2198 ! -----
2199 ! PREFETCH info
2200 #define PREFETCH_MULT_READS 0
2201 ! -----
2202 ! -----
2203 ! define pipes for easier reading

```

2205 #define ICNT %i0

```

2207 #define XPTR %i1
2208 #define XSTR %i2
2209 #define YPTR %i3
2210 #define YSTR %i4

```

```

2212 #define RANGE_LO %l6
2213 #define RANGE_HI %l7

```

```

2215 #define P0_x1 %f0
2216 #define P0_f1 %f1
2217 #define P0_f2 %f2
2218 #define P0_f3 %f3
2219 #define P0_f4 %f4
2220 #define P0_f5 %f5
2221 #define P0_f6 %f6
2222 #define P0_f7 %f7
2223 !#define P0_f8 %f8
2224 #define T0_f8 %f8
2225 #define P0_f9 %f9

```

```

2227 #define P1_x2 %f10
2228 #define P1_f11 %f11
2229 #define P1_f12 %f12
2230 #define P1_f13 %f13
2231 #define P1_f14 %f14
2232 #define P1_f15 %f15

```



```

2761      ba,pt    %icc,.cont0
2762      ! delay  slot
2763      fpsub32s P0_f2,CONSTE432_f29,P0_f2      ! n -= 0x4320000
2764      1:
2765          fdivs  CONSTE432_f29,P0_f1,P0_f2      ! raise div-by-z
2766          ba,pt  %icc,3f
2767      ! delay  slot
2768          st     INF_f28,[YPTR]      ! store -inf
2769      2:
2770          sll   %l0,1,%l0      ! lop off sign bit
2771          add   XPTR,XSTR,XPTR      ! x += stridex
2772          orcc  %l0,%o5,%g0
2773          be,pn %icc,1b      ! if x == -0
2774      ! delay  slot
2775          add   YPTR,YSTR,YPTR      ! y += stridey
2776          fzero P0_f2      ! *y = (x < 0.0? 0.0 : x) * inf
2777          fcmpd %fcc0,P0_X1,P0_f2
2778          fmovdl %fcc0,P0_f2,P0_X1
2779          fand  INF_f28,FP50_MASK,P0_f2
2780          fnegd P0_f2,P0_f2
2781          fmuld P0_X1,P0_f2,P0_X1
2782          st    P0_X1,[YPTR]
2783      3:
2784          addcc ICNT,-1,ICNT
2785          ble,pn %icc,.endloop2
2786      ! delay  slot
2787          st    P0_f1,[YPTR+4]
2788          ld    [XPTR],%l0      ! get next argument
2789          sub   %l0,RANGE_HI,%o3      ! bnds chk x1
2790          sub   RANGE_LO,%l0,%o4      ! bounds chk x1
2791          ldd   [XPTR],P0_X1
2792          fpadd32s P0_X1,TTOPMSK,P0_f2      ! n=(ix+0xc0194000)&0xffff0000
2793          ba,pt %icc,.loop0
2794      ! delay  slot
2795          fands P0_f2,INF_f28,P0_f2      !58

2798      .align 16
2799      .range1:
2800          cmp   %l1,RANGE_HI
2801          bgeu,pn %icc,2f      ! if (unsigned) ix >= 0x7ff00000
2802      ! delay  slot
2803          ld    [XPTR+4],%o5
2804          fxtod P1_X2,P1_X2      ! scale by 2**1074 w/o trapping
2805          st    P1_X2,[%fp+tmp1]
2806          add   XPTR,XSTR,XPTR      ! x += stridex
2807          orcc  %l1,%o5,%g0
2808          be,pn %icc,1f      ! if x == 0
2809      ! delay  slot
2810          add   YPTR,YSTR,YPTR      ! y += stridey
2811          fpadd32s P1_X2,TTOPMSK,P1_f12      ! n = (ix + 0xc0194000)
2812          fands P1_f12,INF_f28,P1_f12
2813          fpsub32s P1_X2,P1_f12,P1_X2      ! u.l[0] -= n
2814          ld    [%fp+tmp1],%l1
2815          ba,pt %icc,.cont1
2816      ! delay  slot
2817          fpsub32s P1_f12,CONSTE432_f29,P1_f12      ! n -= 0x4320000
2818      1:
2819          fdivs  CONSTE432_f29,P1_f11,P1_f12      ! raise div-by-z
2820          ba,pt  %icc,3f
2821      ! delay  slot
2822          st     INF_f28,[YPTR]      ! store -inf
2823      2:
2824          sll   %l1,1,%l1      ! lop off sign bit
2825          add   XPTR,XSTR,XPTR      ! x += stridex
2826          orcc  %l1,%o5,%g0

```

```

2827      be,pn    %icc,1b      ! if x == -0
2828      ! delay  slot
2829          add   YPTR,YSTR,YPTR      ! y += stridey
2830          fzero P1_f12      ! *y = (x < 0.0? 0.0 : x) * inf
2831          fcmpd %fcc0,P1_X2,P1_f12
2832          fmovdl %fcc0,P1_f12,P1_X2
2833          fand  INF_f28,FP50_MASK,P1_f12
2834          fnegd P1_f12,P1_f12
2835          fmuld P1_X2,P1_f12,P1_X2
2836          st    P1_X2,[YPTR]
2837      3:
2838          addcc ICNT,-1,ICNT
2839          ble,pn %icc,.endloop0
2840      ! delay  slot
2841          st    P1_f11,[YPTR+4]
2842          ld    [XPTR],%l1      ! get next argument
2843          ldd   [XPTR],P1_X2
2844          fpadd32s P1_X2,TTOPMSK,P1_f12      ! X + TTOP
2845          ba,pt %icc,.loop1
2846      ! delay  slot
2847          fands P1_f12,INF_f28,P1_f12      ! & INF

2850      .align 16
2851      .range2:
2852          cmp   %l2,RANGE_HI
2853          bgeu,pn %icc,2f      ! if (unsigned) ix >= 0x7ff00000
2854      ! delay  slot
2855          ld    [XPTR+4],%o5
2856          fxtod P2_X3,P2_X3      ! scale by 2**1074 w/o trapping
2857          st    P2_X3,[%fp+tmp2]
2858          add   XPTR,XSTR,XPTR      ! x += stridex
2859          orcc  %l2,%o5,%g0
2860          be,pn %icc,1f      ! if x == 0
2861      ! delay  slot
2862          add   YPTR,YSTR,YPTR      ! y += stridey
2863          fpadd32s P2_X3,TTOPMSK,P2_f22      ! n = (ix + 0xc0194000) & 0xffff0
2864          fands P2_f22,INF_f28,P2_f22
2865          fpsub32s P2_X3,P2_f22,P2_X3      ! u.l[0] -= n
2866          ld    [%fp+tmp2],%l2
2867          ba,pt %icc,.cont2
2868      ! delay  slot
2869          fpsub32s P2_f22,CONSTE432_f29,P2_f22      ! n -= 0x43200000
2870      1:
2871          fdivs  CONSTE432_f29,P2_f21,P2_f22      ! raise div-by-zero
2872          ba,pt  %icc,3f
2873      ! delay  slot
2874          st     INF_f28,[YPTR]      ! store -inf
2875      2:
2876          sll   %l2,1,%l2      ! lop off sign bit
2877          add   XPTR,XSTR,XPTR      ! x += stridex
2878          orcc  %l2,%o5,%g0
2879          be,pn %icc,1b      ! if x == -0
2880      ! delay  slot
2881          add   YPTR,YSTR,YPTR      ! y += stridey
2882          fzero P2_f22      ! *y = (x < 0.0? 0.0 : x) * inf
2883          fcmpd %fcc0,P2_X3,P2_f22
2884          fmovdl %fcc0,P2_f22,P2_X3
2885          fand  INF_f28,FP50_MASK,P2_f22
2886          fnegd P2_f22,P2_f22
2887          fmuld P2_X3,P2_f22,P2_X3
2888          st    P2_X3,[YPTR]
2889      3:
2890          addcc ICNT,-1,ICNT
2891          ble,pn %icc,.endloop1
2892      ! delay  slot

```

```
2893      st      P2_f21,[YPTR+4]
2894      ld      [XPTR],%12      ! get next argument
2895      ldd     [XPTR],P2_X3
2896      fpadd32s P2_X3,TTOPMSK,P2_f22 ! X + TTOP
2897      ba,pt   %icc,.loop2
2898 ! delay slot
2899      fands   P2_f22,INF_f28,P2_f22      !      X3
2900      nop !ld [XPTR+4],P2_f21

2902      SET_SIZE(__vlog_ultra3)
unchanged_portion_omitted_
```

new/usr/src/lib/libmvec/common/vis/_vsin_ultra3.S

1

```
*****
49907 Tue Nov 25 12:59:56 2014
new/usr/src/lib/libmvec/common/vis/_vsin_ultra3.S
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
```

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file      "_vsin_ultra3.S"

31 #include "libm.h"
32 #if defined(LIBMVEC_SO_BUILD)
33     .weak      __vsin
34     .type      __vsin, #function
35     __vsin = __vsin_ultra3
36 #endif

36     RO_DATA
37     .align    64
38 constants:
39     .word     0x42c80000,0x00000000    ! 3 * 2^44
40     .word     0x43380000,0x00000000    ! 3 * 2^51
41     .word     0x3fe45f30,0x6dc9c883    ! invpio2
42     .word     0x3ff921fb,0x54442c00    ! pio2_1
43     .word     0x3d318469,0x898cc400    ! pio2_2
44     .word     0x3a71701b,0x839a2520    ! pio2_3
45     .word     0xbfc55555,0x55555533    ! pp1
46     .word     0x3f811111,0x10e7d53b    ! pp2
47     .word     0xbf2a0167,0xe6b3cf9b    ! pp3
48     .word     0xbfdfffff,0xfffff65    ! qq1
49     .word     0x3fa55555,0x54f88ed0    ! qq2
50     .word     0xbf56c12c,0xdd185f60    ! qq3

52 ! local storage indices

54 #define xsave      STACK_BIAS-0x8
```

new/usr/src/lib/libmvec/common/vis/_vsin_ultra3.S

2

```
55 #define ysave      STACK_BIAS-0x10
56 #define nsave      STACK_BIAS-0x14
57 #define xsxsave    STACK_BIAS-0x18
58 #define sysave     STACK_BIAS-0x1c
59 #define biguns     STACK_BIAS-0x20
60 #define nk3        STACK_BIAS-0x24
61 #define nk2        STACK_BIAS-0x28
62 #define nk1        STACK_BIAS-0x2c
63 #define nk0        STACK_BIAS-0x30
64 #define junk       STACK_BIAS-0x38
65 ! sizeof temp storage - must be a multiple of 16 for V9
66 #define tmps       0x40

68 ! register use

70 ! i0  n
71 ! i1  x
72 ! i2  stridex
73 ! i3  y
74 ! i4  stridex
75 ! i5  0x80000000

77 ! l0  hx0
78 ! l1  hx1
79 ! l2  hx2
80 ! l3  hx3
81 ! l4  k0
82 ! l5  k1
83 ! l6  k2
84 ! l7  k3

86 ! the following are 64-bit registers in both V8+ and V9

88 ! g1  __vlibm_TBL_sincos2
89 ! g5  scratch

91 ! o0  py0
92 ! o1  py1
93 ! o2  py2
94 ! o3  py3
95 ! o4  0x3e400000
96 ! o5  0x3fe921fb,0x4099251e
97 ! o7  scratch

99 ! f0  hx0
100 ! f2
101 ! f4
102 ! f6
103 ! f8  hx1
104 ! f10
105 ! f12
106 ! f14
107 ! f16 hx2
108 ! f18
109 ! f20
110 ! f22
111 ! f24 hx3
112 ! f26
113 ! f28
114 ! f30
115 ! f32
116 ! f34
117 ! f36
118 ! f38

120 #define c3two44 %f40
```



```

121 #define c3two51 %f42
122 #define invpio2 %f44
123 #define pio2_1 %f46
124 #define pio2_2 %f48
125 #define pio2_3 %f50
126 #define ppl %f52
127 #define pp2 %f54
128 #define pp3 %f56
129 #define qq1 %f58
130 #define qq2 %f60
131 #define qq3 %f62

133 ENTRY(__vsin_ultra3)
134 save %sp,-SA(MINFRAME)-tmpls,%sp
135 PIC_SETUP(17)
136 PIC_SET(17,constants,o0)
137 PIC_SET(17,__vlibm_TBL_sincos2,o1)
138 mov %o1,%g1
139 wr %g0,0x82,%asi ! set %asi for non-faulting loads
140 #ifdef __sparcv9
141 stx %i1,[%fp+xsave] ! save arguments
142 stx %i3,[%fp+ysave]
143 #else
144 st %i1,[%fp+xsave] ! save arguments
145 st %i3,[%fp+ysave]
146 #endif
147 st %i0,[%fp+nsave]
148 st %i2,[%fp+sxsave]
149 st %i4,[%fp+sysave]
150 st %g0,[%fp+biguns] ! biguns = 0
151 ldd [%o0+0x00],c3two44 ! load/set up constants
152 ldd [%o0+0x08],c3two51
153 ldd [%o0+0x10],invpio2
154 ldd [%o0+0x18],pio2_1
155 ldd [%o0+0x20],pio2_2
156 ldd [%o0+0x28],pio2_3
157 ldd [%o0+0x30],ppl
158 ldd [%o0+0x38],pp2
159 ldd [%o0+0x40],pp3
160 ldd [%o0+0x48],qq1
161 ldd [%o0+0x50],qq2
162 ldd [%o0+0x58],qq3
163 sethi %hi(0x80000000),%i5
164 sethi %hi(0x3e400000),%o4
165 sethi %hi(0x3fe921fb),%o5
166 or %o5,%lo(0x3fe921fb),%o5
167 sllx %o5,32,%o5
168 sethi %hi(0x4099251e),%o7
169 or %o7,%lo(0x4099251e),%o7
170 or %o5,%o7,%o5
171 sll %i2,3,%i2 ! scale strides
172 sll %i4,3,%i4
173 add %fp,junk,%o1 ! loop prologue
174 add %fp,junk,%o2
175 add %fp,junk,%o3
176 ld [%i1],%l0 ! *x
177 ld [%i1],%f0
178 ld [%i1+4],%f3
179 andn %l0,%i5,%l0 ! mask off sign
180 ba .loop0
181 add %i1,%i2,%i1 ! x += stridex

183 ! 16-byte aligned
184 .align 16
185 .loop0:
186 lda [%i1]%asi,%l1 ! preload next argument

```

```

187 sub %l0,%o4,%g5
188 sub %o5,%l0,%o7
189 fabss %f0,%f2

191 lda [%i1]%asi,%f8
192 orcc %o7,%g5,%g0
193 mov %i3,%o0 ! py0 = y
194 bl,pn %icc,.range0 ! hx < 0x3e400000 or hx > 0x4099251e

196 ! delay slot
197 lda [%i1+4]%asi,%f11
198 addcc %i0,-1,%i0
199 add %i3,%i4,%i3 ! y += stridey
200 ble,pn %icc,.last1

202 ! delay slot
203 andn %l1,%i5,%l1
204 add %i1,%i2,%i1 ! x += stridex
205 fadd %f2,c3two44,%f4
206 st %f15,[%o1+4]

208 .loop1:
209 lda [%i1]%asi,%l2 ! preload next argument
210 sub %l1,%o4,%g5
211 sub %o5,%l1,%o7
212 fabss %f8,%f10

214 lda [%i1]%asi,%f16
215 orcc %o7,%g5,%g0
216 mov %i3,%o1 ! py1 = y
217 bl,pn %icc,.range1 ! hx < 0x3e400000 or hx > 0x4099251e

219 ! delay slot
220 lda [%i1+4]%asi,%f19
221 addcc %i0,-1,%i0
222 add %i3,%i4,%i3 ! y += stridey
223 ble,pn %icc,.last2

225 ! delay slot
226 andn %l2,%i5,%l2
227 add %i1,%i2,%i1 ! x += stridex
228 fadd %f10,c3two44,%f12
229 st %f23,[%o2+4]

231 .loop2:
232 lda [%i1]%asi,%l3 ! preload next argument
233 sub %l2,%o4,%g5
234 sub %o5,%l2,%o7
235 fabss %f16,%f18

237 lda [%i1]%asi,%f24
238 orcc %o7,%g5,%g0
239 mov %i3,%o2 ! py2 = y
240 bl,pn %icc,.range2 ! hx < 0x3e400000 or hx > 0x4099251e

242 ! delay slot
243 lda [%i1+4]%asi,%f27
244 addcc %i0,-1,%i0
245 add %i3,%i4,%i3 ! y += stridey
246 ble,pn %icc,.last3

248 ! delay slot
249 andn %l3,%i5,%l3
250 add %i1,%i2,%i1 ! x += stridex
251 fadd %f18,c3two44,%f20
252 st %f31,[%o3+4]

```

```

254 .loop3:
255     sub     %l3,%o4,%g5
256     sub     %o5,%l3,%o7
257     fabss  %f24,%f26
258     st     %f5,[%fp+nk0]

260     orcc   %o7,%g5,%g0
261     mov     %i3,%o3           ! py3 = y
262     bl,pn  %icc,.range3     ! hx < 0x3e400000 or > hx 0x4099251e
263 ! delay slot
264     st     %f13,[%fp+nk1]

266 !!! DONE?
267 .cont:
268     srlx   %o5,32,%o7
269     add     %i3,%i4,%i3     ! y += stridey
270     fmovs  %f3,%f1
271     st     %f21,[%fp+nk2]

273     sub     %o7,%l0,%l10
274     sub     %o7,%l1,%l11
275     faddd  %f26,c3two44,%f28
276     st     %f29,[%fp+nk3]

278     sub     %o7,%l2,%l12
279     sub     %o7,%l3,%l13
280     fmovs  %f11,%f9

282     or     %l0,%l1,%l10
283     or     %l2,%l3,%l12
284     fmovs  %f19,%f17

286     fmovs  %f27,%f25
287     fmuld  %f0,invpio2,%f6   ! x * invpio2, for medium range

289     fmuld  %f8,invpio2,%f14
290     ld     [%fp+nk0],%l14

292     fmuld  %f16,invpio2,%f22
293     ld     [%fp+nk1],%l15

295     orcc   %l0,%l2,%g0
296     bl,pn  %icc,.medium
297 ! delay slot
298     fmuld  %f24,invpio2,%f30
299     ld     [%fp+nk2],%l16

301     ld     [%fp+nk3],%l17
302     sll   %l4,5,%l14       ! k
303     fcmpd  %fcc0,%f0,pio2_3 ! x < pio2_3 iff x < 0

305     sll   %l5,5,%l15
306     ldd   [%l4+%g1],%f4
307     fcmpd  %fcc1,%f8,pio2_3

309     sll   %l6,5,%l16
310     ldd   [%l5+%g1],%f12
311     fcmpd  %fcc2,%f16,pio2_3

313     sll   %l7,5,%l17
314     ldd   [%l6+%g1],%f20
315     fcmpd  %fcc3,%f24,pio2_3

317     ldd   [%l7+%g1],%f28
318     fsubd  %f2,%f4,%f2     ! x -= __vlibm_TBL_sincos2[k]

```

```

320     fsubd  %f10,%f12,%f10
322     fsubd  %f18,%f20,%f18
324     fsubd  %f26,%f28,%f26
326     fmuld  %f2,%f2,%f0     ! z = x * x
328     fmuld  %f10,%f10,%f8
330     fmuld  %f18,%f18,%f16
332     fmuld  %f26,%f26,%f24
334     fmuld  %f0,pp3,%f6
336     fmuld  %f8,pp3,%f14
338     fmuld  %f16,pp3,%f22
340     fmuld  %f24,pp3,%f30
342     faddd  %f6,pp2,%f6
343     fmuld  %f0,qq2,%f4
345     faddd  %f14,pp2,%f14
346     fmuld  %f8,qq2,%f12
348     faddd  %f22,pp2,%f22
349     fmuld  %f16,qq2,%f20
351     faddd  %f30,pp2,%f30
352     fmuld  %f24,qq2,%f28
354     fmuld  %f0,%f6,%f6
355     faddd  %f4,qq1,%f4
357     fmuld  %f8,%f14,%f14
358     faddd  %f12,qq1,%f12
360     fmuld  %f16,%f22,%f22
361     faddd  %f20,qq1,%f20
363     fmuld  %f24,%f30,%f30
364     faddd  %f28,qq1,%f28
366     faddd  %f6,pp1,%f6
367     fmuld  %f0,%f4,%f4
368     add    %l4,%g1,%l14
370     faddd  %f14,pp1,%f14
371     fmuld  %f8,%f12,%f12
372     add    %l5,%g1,%l15
374     faddd  %f22,pp1,%f22
375     fmuld  %f16,%f20,%f20
376     add    %l6,%g1,%l16
378     faddd  %f30,pp1,%f30
379     fmuld  %f24,%f28,%f28
380     add    %l7,%g1,%l17
382     fmuld  %f0,%f6,%f6
383     ldd   [%l4+8],%f0

```

```

385    fmuld    %f8,%f14,%f14
386    ldd     [%15+8],%f8

388    fmuld    %f16,%f22,%f22
389    ldd     [%16+8],%f16

391    fmuld    %f24,%f30,%f30
392    ldd     [%17+8],%f24

394    fmuld    %f2,%f6,%f6

396    fmuld    %f10,%f14,%f14

398    fmuld    %f18,%f22,%f22

400    fmuld    %f26,%f30,%f30

402    faddd    %f6,%f2,%f6
403    fmuld    %f0,%f4,%f4
404    ldd     [%14+16],%f2

406    faddd    %f14,%f10,%f14
407    fmuld    %f8,%f12,%f12
408    ldd     [%15+16],%f10

410    faddd    %f22,%f18,%f22
411    fmuld    %f16,%f20,%f20
412    ldd     [%16+16],%f18

414    faddd    %f30,%f26,%f30
415    fmuld    %f24,%f28,%f28
416    ldd     [%17+16],%f26

418    fmuld    %f2,%f6,%f6

420    fmuld    %f10,%f14,%f14

422    fmuld    %f18,%f22,%f22

424    fmuld    %f26,%f30,%f30

426    faddd    %f6,%f4,%f6

428    faddd    %f14,%f12,%f14

430    faddd    %f22,%f20,%f22

432    faddd    %f30,%f28,%f30

434    faddd    %f6,%f0,%f6

436    faddd    %f14,%f8,%f14

438    faddd    %f22,%f16,%f22

440    faddd    %f30,%f24,%f30

442    fnegd    %f6,%f4
443    lda     [%i1]%asi,%i0        ! preload next argument

445    fnegd    %f14,%f12
446    lda     [%i1]%asi,%f0

448    fnegd    %f22,%f20
449    lda     [%i1+4]%asi,%f3

```

```

451    fnegd    %f30,%f28
452    andn    %i0,%i5,%i10
453    add     %i1,%i2,%i1

455    fmovd1   %fcc0,%f4,%f6        ! (hx < -0)? -s : s
456    st      %f6, [%o0]

458    fmovd1   %fcc1,%f12,%f14
459    st      %f14, [%o1]

461    fmovd1   %fcc2,%f20,%f22
462    st      %f22, [%o2]

464    fmovd1   %fcc3,%f28,%f30
465    st      %f30, [%o3]
466    addcc    %i0,-1,%i0

468    bg,pt   %icc,.loop0
469 ! delay   slot
470    st      %f7, [%o0+4]

472    ba,pt   %icc,.end
473 ! delay   slot
474    nop

477    .align   16
478 .medium:
479    faddd    %f6,c3two51,%f4
480    st      %f5, [%fp+nk0]

482    faddd    %f14,c3two51,%f12
483    st      %f13, [%fp+nk1]

485    faddd    %f22,c3two51,%f20
486    st      %f21, [%fp+nk2]

488    faddd    %f30,c3two51,%f28
489    st      %f29, [%fp+nk3]

491    fsubd    %f4,c3two51,%f6

493    fsubd    %f12,c3two51,%f14

495    fsubd    %f20,c3two51,%f22

497    fsubd    %f28,c3two51,%f30

499    fmuld    %f6,pio2_1,%f2
500    ld      [%fp+nk0],%i0        ! n

502    fmuld    %f14,pio2_1,%f10
503    ld      [%fp+nk1],%i1

505    fmuld    %f22,pio2_1,%f18
506    ld      [%fp+nk2],%i2

508    fmuld    %f30,pio2_1,%f26
509    ld      [%fp+nk3],%i3

511    fsubd    %f0,%f2,%f0
512    fmuld    %f6,pio2_2,%f4

514    fsubd    %f8,%f10,%f8
515    fmuld    %f14,pio2_2,%f12

```

```

517     fsubd    %f16,%f18,%f16
518     fmuld    %f22,pio2_2,%f20

520     fsubd    %f24,%f26,%f24
521     fmuld    %f30,pio2_2,%f28

523     fsubd    %f0,%f4,%f32

525     fsubd    %f8,%f12,%f34

527     fsubd    %f16,%f20,%f36

529     fsubd    %f24,%f28,%f38

531     fsubd    %f0,%f32,%f0
532     fcmple32 %f32,pio2_3,%14      ! x <= pio2_3 iff x < 0

534     fsubd    %f8,%f34,%f8
535     fcmple32 %f34,pio2_3,%15

537     fsubd    %f16,%f36,%f16
538     fcmple32 %f36,pio2_3,%16

540     fsubd    %f24,%f38,%f24
541     fcmple32 %f38,pio2_3,%17

543     fsubd    %f0,%f4,%f0
544     fmuld    %f6,pio2_3,%f6
545     sll      %14,30,%14          ! if (x < 0) n = -n ^ 2

547     fsubd    %f8,%f12,%f8
548     fmuld    %f14,pio2_3,%f14
549     sll      %15,30,%15

551     fsubd    %f16,%f20,%f16
552     fmuld    %f22,pio2_3,%f22
553     sll      %16,30,%16

555     fsubd    %f24,%f28,%f24
556     fmuld    %f30,pio2_3,%f30
557     sll      %17,30,%17

559     fsubd    %f6,%f0,%f6
560     sra      %14,31,%14

562     fsubd    %f14,%f8,%f14
563     sra      %15,31,%15

565     fsubd    %f22,%f16,%f22
566     sra      %16,31,%16

568     fsubd    %f30,%f24,%f30
569     sra      %17,31,%17

571     fsubd    %f32,%f6,%f0      ! reduced x
572     xor     %10,%14,%10

574     fsubd    %f34,%f14,%f8
575     xor     %11,%15,%11

577     fsubd    %f36,%f22,%f16
578     xor     %12,%16,%12

580     fsubd    %f38,%f30,%f24
581     xor     %13,%17,%13

```

```

583     fabsd    %f0,%f2
584     sub     %10,%14,%10

586     fabsd    %f8,%f10
587     sub     %11,%15,%11

589     fabsd    %f16,%f18
590     sub     %12,%16,%12

592     fabsd    %f24,%f26
593     sub     %13,%17,%13

595     faddd    %f2,c3two44,%f4
596     st      %f5,[%fp+nk0]
597     and     %14,2,%14

599     faddd    %f10,c3two44,%f12
600     st      %f13,[%fp+nk1]
601     and     %15,2,%15

603     faddd    %f18,c3two44,%f20
604     st      %f21,[%fp+nk2]
605     and     %16,2,%16

607     faddd    %f26,c3two44,%f28
608     st      %f29,[%fp+nk3]
609     and     %17,2,%17

611     fsubd    %f32,%f0,%f4
612     xor     %10,%14,%10

614     fsubd    %f34,%f8,%f12
615     xor     %11,%15,%11

617     fsubd    %f36,%f16,%f20
618     xor     %12,%16,%12

620     fsubd    %f38,%f24,%f28
621     xor     %13,%17,%13

623     fzero   %f38
624     ld      [%fp+nk0],%14

626     fsubd    %f4,%f6,%f6      ! w
627     ld      [%fp+nk1],%15

629     fsubd    %f12,%f14,%f14
630     ld      [%fp+nk2],%16

632     fnegd   %f38,%f38
633     ld      [%fp+nk3],%17
634     sll     %14,5,%14        ! k

636     fsubd    %f20,%f22,%f22
637     sll     %15,5,%15

639     fsubd    %f28,%f30,%f30
640     sll     %16,5,%16

642     fand    %f0,%f38,%f32      ! sign bit of x
643     ldd    [%14+%g1],%f4
644     sll    %17,5,%17

646     fand    %f8,%f38,%f34
647     ldd    [%15+%g1],%f12

```

```

649      fand    %f16,%f38,%f36
650      ldd     [%16+%g1],%f20

652      fand    %f24,%f38,%f38
653      ldd     [%17+%g1],%f28

655      fsubd   %f2,%f4,%f2          ! x -= __vlibm_TBL_sincos2[k]

657      fsubd   %f10,%f12,%f10

659      fsubd   %f18,%f20,%f18
660      nop

662      fsubd   %f26,%f28,%f26
663      nop

665 ! 16-byte aligned
666      fmuld   %f2,%f2,%f0          ! z = x * x
667      andcc   %l0,1,%g0
668      bz, pn  %icc,.case8
669 ! delay slot
670      fxor    %f6,%f32,%f32

672      fmuld   %f10,%f10,%f8
673      andcc   %l1,1,%g0
674      bz, pn  %icc,.case4
675 ! delay slot
676      fxor    %f14,%f34,%f34

678      fmuld   %f18,%f18,%f16
679      andcc   %l2,1,%g0
680      bz, pn  %icc,.case2
681 ! delay slot
682      fxor    %f22,%f36,%f36

684      fmuld   %f26,%f26,%f24
685      andcc   %l3,1,%g0
686      bz, pn  %icc,.case1
687 ! delay slot
688      fxor    %f30,%f38,%f38

690 !.case0:
691      fmuld   %f0,qq3,%f6          ! cos(x0)

693      fmuld   %f8,qq3,%f14        ! cos(x1)

695      fmuld   %f16,qq3,%f22      ! cos(x2)

697      fmuld   %f24,qq3,%f30      ! cos(x3)

699      faddd   %f6,qq2,%f6
700      fmuld   %f0,pp2,%f4

702      faddd   %f14,qq2,%f14
703      fmuld   %f8,pp2,%f12

705      faddd   %f22,qq2,%f22
706      fmuld   %f16,pp2,%f20

708      faddd   %f30,qq2,%f30
709      fmuld   %f24,pp2,%f28

711      fmuld   %f0,%f6,%f6
712      faddd   %f4,pp1,%f4

714      fmuld   %f8,%f14,%f14

```

```

715      faddd   %f12,pp1,%f12

717      fmuld   %f16,%f22,%f22
718      faddd   %f20,pp1,%f20

720      fmuld   %f24,%f30,%f30
721      faddd   %f28,pp1,%f28

723      faddd   %f6,qq1,%f6
724      fmuld   %f0,%f4,%f4
725      add     %l4,%g1,%l4

727      faddd   %f14,qq1,%f14
728      fmuld   %f8,%f12,%f12
729      add     %l5,%g1,%l5

731      faddd   %f22,qq1,%f22
732      fmuld   %f16,%f20,%f20
733      add     %l6,%g1,%l6

735      faddd   %f30,qq1,%f30
736      fmuld   %f24,%f28,%f28
737      add     %l7,%g1,%l7

739      fmuld   %f2,%f4,%f4

741      fmuld   %f10,%f12,%f12

743      fmuld   %f18,%f20,%f20

745      fmuld   %f26,%f28,%f28

747      fmuld   %f0,%f6,%f6
748      faddd   %f4,%f32,%f4
749      ldd     [%l4+16],%f0

751      fmuld   %f8,%f14,%f14
752      faddd   %f12,%f34,%f12
753      ldd     [%l5+16],%f8

755      fmuld   %f16,%f22,%f22
756      faddd   %f20,%f36,%f20
757      ldd     [%l6+16],%f16

759      fmuld   %f24,%f30,%f30
760      faddd   %f28,%f38,%f28
761      ldd     [%l7+16],%f24

763      fmuld   %f0,%f6,%f6
764      faddd   %f4,%f2,%f4
765      ldd     [%l4+8],%f32

767      fmuld   %f8,%f14,%f14
768      faddd   %f12,%f10,%f12
769      ldd     [%l5+8],%f34

771      fmuld   %f16,%f22,%f22
772      faddd   %f20,%f18,%f20
773      ldd     [%l6+8],%f36

775      fmuld   %f24,%f30,%f30
776      faddd   %f28,%f26,%f28
777      ldd     [%l7+8],%f38

779      fmuld   %f32,%f4,%f4

```

```

781      fmuld   %f34,%f12,%f12
783      fmuld   %f36,%f20,%f20
785      fmuld   %f38,%f28,%f28
787      fsubd   %f6,%f4,%f6
789      fsubd   %f14,%f12,%f14
791      fsubd   %f22,%f20,%f22
793      fsubd   %f30,%f28,%f30
795      faddd   %f6,%f0,%f6
797      faddd   %f14,%f8,%f14
799      faddd   %f22,%f16,%f22
801      faddd   %f30,%f24,%f30
802      mov     %l0,%l14
804      fnegd   %f6,%f4
805      lda     [%l1]%asi,%l0      ! preload next argument
807      fnegd   %f14,%f12
808      lda     [%l1]%asi,%f0
810      fnegd   %f22,%f20
811      lda     [%l1+4]%asi,%f3
813      fnegd   %f30,%f28
814      andn   %l0,%i5,%l0
815      add     %i1,%i2,%i1
817      andcc   %l4,2,%g0
818      fmovdnz %icc,%f4,%f6
819      st      %f6, [%o0]
821      andcc   %l1,2,%g0
822      fmovdnz %icc,%f12,%f14
823      st      %f14, [%o1]
825      andcc   %l2,2,%g0
826      fmovdnz %icc,%f20,%f22
827      st      %f22, [%o2]
829      andcc   %l3,2,%g0
830      fmovdnz %icc,%f28,%f30
831      st      %f30, [%o3]
833      addcc   %i0,-1,%i0
834      bg,pt  %icc,.loop0
835 ! delay  slot
836      st      %f7, [%o0+4]
838      ba,pt  %icc,.end
839 ! delay  slot
840      nop
842      .align  16
843 .case1:
844      fmuld   %f24,pp3,%f30      ! sin(x3)
846      fmuld   %f0,qq3,%f6       ! cos(x0)

```

```

848      fmuld   %f8,qq3,%f14      ! cos(x1)
850      fmuld   %f16,qq3,%f22     ! cos(x2)
852      faddd   %f30,pp2,%f30
853      fmuld   %f24,qq2,%f28
855      faddd   %f6,qq2,%f6
856      fmuld   %f0,pp2,%f4
858      faddd   %f14,qq2,%f14
859      fmuld   %f8,pp2,%f12
861      faddd   %f22,qq2,%f22
862      fmuld   %f16,pp2,%f20
864      fmuld   %f24,%f30,%f30
865      faddd   %f28,qq1,%f28
867      fmuld   %f0,%f6,%f6
868      faddd   %f4,pp1,%f4
870      fmuld   %f8,%f14,%f14
871      faddd   %f12,pp1,%f12
873      fmuld   %f16,%f22,%f22
874      faddd   %f20,pp1,%f20
876      faddd   %f30,pp1,%f30
877      fmuld   %f24,%f28,%f28
878      add     %l7,%g1,%l7
880      faddd   %f6,qq1,%f6
881      fmuld   %f0,%f4,%f4
882      add     %l4,%g1,%l4
884      faddd   %f14,qq1,%f14
885      fmuld   %f8,%f12,%f12
886      add     %l5,%g1,%l5
888      faddd   %f22,qq1,%f22
889      fmuld   %f16,%f20,%f20
890      add     %l6,%g1,%l6
892      fmuld   %f24,%f30,%f30
894      fmuld   %f2,%f4,%f4
896      fmuld   %f10,%f12,%f12
898      fmuld   %f18,%f20,%f20
900      fmuld   %f26,%f30,%f30
901      ldd     [%l17+8],%f24
903      fmuld   %f0,%f6,%f6
904      faddd   %f4,%f32,%f4
905      ldd     [%l14+16],%f0
907      fmuld   %f8,%f14,%f14
908      faddd   %f12,%f34,%f12
909      ldd     [%l15+16],%f8
911      fmuld   %f16,%f22,%f22
912      faddd   %f20,%f36,%f20

```

```

913      ldd      [%16+16],%f16

915      fmuld   %f24,%f28,%f28
916      faddd   %f38,%f30,%f30

918      fmuld   %f0,%f6,%f6
919      faddd   %f4,%f2,%f4
920      ldd      [%14+8],%f32

922      fmuld   %f8,%f14,%f14
923      faddd   %f12,%f10,%f12
924      ldd      [%15+8],%f34

926      fmuld   %f16,%f22,%f22
927      faddd   %f20,%f18,%f20
928      ldd      [%16+8],%f36

930      faddd   %f26,%f30,%f30
931      ldd      [%17+16],%f38

933      fmuld   %f32,%f4,%f4

935      fmuld   %f34,%f12,%f12

937      fmuld   %f36,%f20,%f20

939      fmuld   %f38,%f30,%f30

941      fsubd   %f6,%f4,%f6

943      fsubd   %f14,%f12,%f14

945      fsubd   %f22,%f20,%f22

947      faddd   %f30,%f28,%f30

949      faddd   %f6,%f0,%f6

951      faddd   %f14,%f8,%f14

953      faddd   %f22,%f16,%f22

955      faddd   %f30,%f24,%f30
956      mov     %10,%14

958      fnegd   %f6,%f4
959      lda     [%i1]%asi,%10      ! preload next argument

961      fnegd   %f14,%f12
962      lda     [%i1]%asi,%f0

964      fnegd   %f22,%f20
965      lda     [%i1+4]%asi,%f3

967      fnegd   %f30,%f28
968      andn   %10,%i5,%10
969      add     %i1,%i2,%i1

971      andcc   %14,2,%g0
972      fmovdnz %icc,%f4,%f6
973      st      %f6,[%o0]

975      andcc   %11,2,%g0
976      fmovdnz %icc,%f12,%f14
977      st      %f14,[%o1]

```

```

979      andcc   %12,2,%g0
980      fmovdnz %icc,%f20,%f22
981      st      %f22,[%o2]

983      andcc   %13,2,%g0
984      fmovdnz %icc,%f28,%f30
985      st      %f30,[%o3]

987      addcc   %i0,-1,%i0
988      bg,pt   %icc,.loop0
989 ! delay    slot
990      st      %f7,[%o0+4]

992      ba,pt   %icc,.end
993 ! delay    slot
994      nop

996      .align  16
997 .case2:
998      fmuld   %f26,%f26,%f24
999      andcc   %13,1,%g0
1000     bz,pn   %icc,.case3
1001 ! delay    slot
1002     fxor   %f30,%f38,%f38

1004     fmuld   %f16,pp3,%f22      ! sin(x2)

1006     fmuld   %f0,qq3,%f6      ! cos(x0)

1008     fmuld   %f8,qq3,%f14     ! cos(x1)

1010     faddd   %f22,pp2,%f22
1011     fmuld   %f16,qq2,%f20

1013     fmuld   %f24,qq3,%f30     ! cos(x3)

1015     faddd   %f6,qq2,%f6
1016     fmuld   %f0,pp2,%f4

1018     faddd   %f14,qq2,%f14
1019     fmuld   %f8,pp2,%f12

1021     fmuld   %f16,%f22,%f22
1022     faddd   %f20,qq1,%f20

1024     faddd   %f30,qq2,%f30
1025     fmuld   %f24,pp2,%f28

1027     fmuld   %f0,%f6,%f6
1028     faddd   %f4,pp1,%f4

1030     fmuld   %f8,%f14,%f14
1031     faddd   %f12,pp1,%f12

1033     faddd   %f22,pp1,%f22
1034     fmuld   %f16,%f20,%f20
1035     add     %16,%g1,%16

1037     fmuld   %f24,%f30,%f30
1038     faddd   %f28,pp1,%f28

1040     faddd   %f6,qq1,%f6
1041     fmuld   %f0,%f4,%f4
1042     add     %14,%g1,%14

1044     faddd   %f14,qq1,%f14

```

```

1045      fmuld   %f8,%f12,%f12
1046      add     %l5,%g1,%l5

1048      fmuld   %f16,%f22,%f22

1050      faddd   %f30,%qg1,%f30
1051      fmuld   %f24,%f28,%f28
1052      add     %l7,%g1,%l7

1054      fmuld   %f2,%f4,%f4

1056      fmuld   %f10,%f12,%f12

1058      fmuld   %f18,%f22,%f22
1059      ldd     [%l16+8],%f16

1061      fmuld   %f26,%f28,%f28

1063      fmuld   %f0,%f6,%f6
1064      faddd   %f4,%f32,%f4
1065      ldd     [%l14+16],%f4

1067      fmuld   %f8,%f14,%f14
1068      faddd   %f12,%f34,%f12
1069      ldd     [%l15+16],%f8

1071      fmuld   %f16,%f20,%f20
1072      faddd   %f36,%f22,%f22

1074      fmuld   %f24,%f30,%f30
1075      faddd   %f28,%f38,%f28
1076      ldd     [%l17+16],%f24

1078      fmuld   %f0,%f6,%f6
1079      faddd   %f4,%f2,%f4
1080      ldd     [%l14+8],%f32

1082      fmuld   %f8,%f14,%f14
1083      faddd   %f12,%f10,%f12
1084      ldd     [%l15+8],%f34

1086      faddd   %f18,%f22,%f22
1087      ldd     [%l16+16],%f36

1089      fmuld   %f24,%f30,%f30
1090      faddd   %f28,%f26,%f28
1091      ldd     [%l17+8],%f38

1093      fmuld   %f32,%f4,%f4

1095      fmuld   %f34,%f12,%f12

1097      fmuld   %f36,%f22,%f22

1099      fmuld   %f38,%f28,%f28

1101      fsubd   %f6,%f4,%f6

1103      fsubd   %f14,%f12,%f14

1105      faddd   %f22,%f20,%f22

1107      fsubd   %f30,%f28,%f30

1109      faddd   %f6,%f0,%f6

```

```

1111      faddd   %f14,%f8,%f14

1113      faddd   %f22,%f16,%f22

1115      faddd   %f30,%f24,%f30
1116      mov     %l0,%l4

1118      fnegd   %f6,%f4
1119      lda     [%i1]%asi,%l0          ! preload next argument

1121      fnegd   %f14,%f12
1122      lda     [%i1]%asi,%f0

1124      fnegd   %f22,%f20
1125      lda     [%i1+4]%asi,%f3

1127      fnegd   %f30,%f28
1128      andn    %l0,%i5,%l0
1129      add     %i1,%i2,%i1

1131      andcc   %l4,2,%g0
1132      fmovdnz %icc,%f4,%f6
1133      st      %f6,[%o0]

1135      andcc   %l1,2,%g0
1136      fmovdnz %icc,%f12,%f14
1137      st      %f14,[%o1]

1139      andcc   %l2,2,%g0
1140      fmovdnz %icc,%f20,%f22
1141      st      %f22,[%o2]

1143      andcc   %l3,2,%g0
1144      fmovdnz %icc,%f28,%f30
1145      st      %f30,[%o3]

1147      addcc   %i0,-1,%i0
1148      bg,pt   %icc,.loop0
1149      ! delay slot
1150      st      %f7,[%o0+4]

1152      ba,pt   %icc,.end
1153      ! delay slot
1154      nop

1156      .align 16
1157      .case3:
1158      fmuld   %f16,pp3,%f22          ! sin(x2)

1160      fmuld   %f24,pp3,%f30          ! sin(x3)

1162      fmuld   %f0,qq3,%f6          ! cos(x0)

1164      fmuld   %f8,qq3,%f14          ! cos(x1)

1166      faddd   %f22,pp2,%f22
1167      fmuld   %f16,qq2,%f20

1169      faddd   %f30,pp2,%f30
1170      fmuld   %f24,qq2,%f28

1172      faddd   %f6,qq2,%f6
1173      fmuld   %f0,pp2,%f4

1175      faddd   %f14,qq2,%f14
1176      fmuld   %f8,pp2,%f12

```



```

1178    fmuld    %f16,%f22,%f22
1179    fadddd   %f20,qq1,%f20

1181    fmuld    %f24,%f30,%f30
1182    fadddd   %f28,qq1,%f28

1184    fmuld    %f0,%f6,%f6
1185    fadddd   %f4,pp1,%f4

1187    fmuld    %f8,%f14,%f14
1188    fadddd   %f12,pp1,%f12

1190    fadddd   %f22,pp1,%f22
1191    fmuld    %f16,%f20,%f20
1192    add     %l6,%g1,%l6

1194    fadddd   %f30,pp1,%f30
1195    fmuld    %f24,%f28,%f28
1196    add     %l17,%g1,%l17

1198    fadddd   %f6,qq1,%f6
1199    fmuld    %f0,%f4,%f4
1200    add     %l14,%g1,%l14

1202    fadddd   %f14,qq1,%f14
1203    fmuld    %f8,%f12,%f12
1204    add     %l15,%g1,%l15

1206    fmuld    %f16,%f22,%f22

1208    fmuld    %f24,%f30,%f30

1210    fmuld    %f2,%f4,%f4

1212    fmuld    %f10,%f12,%f12

1214    fmuld    %f18,%f22,%f22
1215    ldd     [%l16+8],%f16

1217    fmuld    %f26,%f30,%f30
1218    ldd     [%l17+8],%f24

1220    fmuld    %f0,%f6,%f6
1221    fadddd   %f4,%f32,%f4
1222    ldd     [%l14+16],%f0

1224    fmuld    %f8,%f14,%f14
1225    fadddd   %f12,%f34,%f12
1226    ldd     [%l15+16],%f8

1228    fmuld    %f16,%f20,%f20
1229    fadddd   %f36,%f22,%f22

1231    fmuld    %f24,%f28,%f28
1232    fadddd   %f38,%f30,%f30

1234    fmuld    %f0,%f6,%f6
1235    fadddd   %f4,%f2,%f4
1236    ldd     [%l14+8],%f32

1238    fmuld    %f8,%f14,%f14
1239    fadddd   %f12,%f10,%f12
1240    ldd     [%l15+8],%f34

1242    fadddd   %f18,%f22,%f22

```

```

1243    ldd     [%l16+16],%f36

1245    fadddd   %f26,%f30,%f30
1246    ldd     [%l17+16],%f38

1248    fmuld    %f32,%f4,%f4

1250    fmuld    %f34,%f12,%f12

1252    fmuld    %f36,%f22,%f22

1254    fmuld    %f38,%f30,%f30

1256    fsubd    %f6,%f4,%f6

1258    fsubd    %f14,%f12,%f14

1260    fadddd   %f22,%f20,%f22

1262    fadddd   %f30,%f28,%f30

1264    fadddd   %f6,%f0,%f6

1266    fadddd   %f14,%f8,%f14

1268    fadddd   %f22,%f16,%f22

1270    fadddd   %f30,%f24,%f30
1271    mov     %l10,%l14

1273    fnegd    %f6,%f4
1274    lda     [%l11]asi,%l10      ! preload next argument

1276    fnegd    %f14,%f12
1277    lda     [%l11]asi,%f0

1279    fnegd    %f22,%f20
1280    lda     [%l11+4]asi,%f3

1282    fnegd    %f30,%f28
1283    andn     %l10,%i5,%l10
1284    add     %l11,%i2,%l11

1286    andcc    %l14,2,%g0
1287    fmovdnz  %icc,%f4,%f6
1288    st       %f6,[%o0]

1290    andcc    %l11,2,%g0
1291    fmovdnz  %icc,%f12,%f14
1292    st       %f14,[%o1]

1294    andcc    %l12,2,%g0
1295    fmovdnz  %icc,%f20,%f22
1296    st       %f22,[%o2]

1298    andcc    %l13,2,%g0
1299    fmovdnz  %icc,%f28,%f30
1300    st       %f30,[%o3]

1302    addcc    %i0,-1,%i0
1303    bg,pt    %icc,.loop0
1304 ! delay  slot
1305    st       %f7,[%o0+4]

1307    ba,pt    %icc,.end
1308 ! delay  slot

```

```

1309      nop

1311      .align 16
1312 .case4:
1313      fmuld   %f18,%f18,%f16
1314      andcc   %l2,1,%g0
1315      bz, pn   %icc,.case6
1316 ! delay slot
1317      fxor    %f22,%f36,%f36

1319      fmuld   %f26,%f26,%f24
1320      andcc   %l3,1,%g0
1321      bz, pn   %icc,.case5
1322 ! delay slot
1323      fxor    %f30,%f38,%f38

1325      fmuld   %f8,pp3,%f14      ! sin(x1)

1327      fmuld   %f0,qq3,%f6      ! cos(x0)

1329      faddd   %f14,pp2,%f14
1330      fmuld   %f8,qq2,%f12

1332      fmuld   %f16,qq3,%f22    ! cos(x2)

1334      fmuld   %f24,qq3,%f30    ! cos(x3)

1336      faddd   %f6,qq2,%f6
1337      fmuld   %f0,pp2,%f4

1339      fmuld   %f8,%f14,%f14
1340      faddd   %f12,qq1,%f12

1342      faddd   %f22,qq2,%f22
1343      fmuld   %f16,pp2,%f20

1345      faddd   %f30,qq2,%f30
1346      fmuld   %f24,pp2,%f28

1348      fmuld   %f0,%f6,%f6
1349      faddd   %f4,pp1,%f4

1351      faddd   %f14,pp1,%f14
1352      fmuld   %f8,%f12,%f12
1353      add     %l5,%g1,%l5

1355      fmuld   %f16,%f22,%f22
1356      faddd   %f20,pp1,%f20

1358      fmuld   %f24,%f30,%f30
1359      faddd   %f28,pp1,%f28

1361      faddd   %f6,qq1,%f6
1362      fmuld   %f0,%f4,%f4
1363      add     %l4,%g1,%l4

1365      fmuld   %f8,%f14,%f14

1367      faddd   %f22,qq1,%f22
1368      fmuld   %f16,%f20,%f20
1369      add     %l6,%g1,%l6

1371      faddd   %f30,qq1,%f30
1372      fmuld   %f24,%f28,%f28
1373      add     %l7,%g1,%l7

```

```

1375      fmuld   %f2,%f4,%f4

1377      fmuld   %f10,%f14,%f14
1378      ldd     [%l5+8],%f8

1380      fmuld   %f18,%f20,%f20

1382      fmuld   %f26,%f28,%f28

1384      fmuld   %f0,%f6,%f6
1385      faddd   %f4,%f32,%f4
1386      ldd     [%l4+16],%f0

1388      fmuld   %f8,%f12,%f12
1389      faddd   %f34,%f14,%f14

1391      fmuld   %f16,%f22,%f22
1392      faddd   %f20,%f36,%f20
1393      ldd     [%l6+16],%f16

1395      fmuld   %f24,%f30,%f30
1396      faddd   %f28,%f38,%f28
1397      ldd     [%l7+16],%f24

1399      fmuld   %f0,%f6,%f6
1400      faddd   %f4,%f2,%f4
1401      ldd     [%l4+8],%f32

1403      faddd   %f10,%f14,%f14
1404      ldd     [%l5+16],%f34

1406      fmuld   %f16,%f22,%f22
1407      faddd   %f20,%f18,%f20
1408      ldd     [%l6+8],%f36

1410      fmuld   %f24,%f30,%f30
1411      faddd   %f28,%f26,%f28
1412      ldd     [%l7+8],%f38

1414      fmuld   %f32,%f4,%f4

1416      fmuld   %f34,%f14,%f14

1418      fmuld   %f36,%f20,%f20

1420      fmuld   %f38,%f28,%f28

1422      fsubd   %f6,%f4,%f6

1424      faddd   %f14,%f12,%f14

1426      fsubd   %f22,%f20,%f22

1428      fsubd   %f30,%f28,%f30

1430      faddd   %f6,%f0,%f6

1432      faddd   %f14,%f8,%f14

1434      faddd   %f22,%f16,%f22

1436      faddd   %f30,%f24,%f30
1437      mov     %l0,%l4

1439      fnegd   %f6,%f4
1440      lda     [%l1]%asi,%l0      ! preload next argument

```

```

1442      fnegd    %f14,%f12
1443      lda      [%i1]asi,%f0

1445      fnegd    %f22,%f20
1446      lda      [%i1+4]asi,%f3

1448      fnegd    %f30,%f28
1449      andn    %i0,%i5,%i10
1450      add     %i1,%i2,%i1

1452      andcc   %i4,2,%g0
1453      fmovdnz %icc,%f4,%f6
1454      st      %f6,[%o0]

1456      andcc   %i11,2,%g0
1457      fmovdnz %icc,%f12,%f14
1458      st      %f14,[%o1]

1460      andcc   %i12,2,%g0
1461      fmovdnz %icc,%f20,%f22
1462      st      %f22,[%o2]

1464      andcc   %i13,2,%g0
1465      fmovdnz %icc,%f28,%f30
1466      st      %f30,[%o3]

1468      addcc   %i0,-1,%i0
1469      bg,pt   %icc,.loop0
1470 ! delay   slot
1471      st      %f7,[%o0+4]

1473      ba,pt   %icc,.end
1474 ! delay   slot
1475      nop

1477      .align  16
1478 .case5:
1479      fmuld   %f8,pp3,%f14          ! sin(x1)

1481      fmuld   %f24,pp3,%f30        ! sin(x3)

1483      fmuld   %f0,qq3,%f6         ! cos(x0)

1485      faddd   %f14,pp2,%f14
1486      fmuld   %f8,qq2,%f12

1488      fmuld   %f16,qq3,%f22        ! cos(x2)

1490      faddd   %f30,pp2,%f30
1491      fmuld   %f24,qq2,%f28

1493      faddd   %f6,qq2,%f6
1494      fmuld   %f0,pp2,%f4

1496      fmuld   %f8,%f14,%f14
1497      faddd   %f12,qq1,%f12

1499      faddd   %f22,qq2,%f22
1500      fmuld   %f16,pp2,%f20

1502      fmuld   %f24,%f30,%f30
1503      faddd   %f28,qq1,%f28

1505      fmuld   %f0,%f6,%f6
1506      faddd   %f4,pp1,%f4

```

```

1508      faddd   %f14,pp1,%f14
1509      fmuld   %f8,%f12,%f12
1510      add     %i15,%g1,%i15

1512      fmuld   %f16,%f22,%f22
1513      faddd   %f20,pp1,%f20

1515      faddd   %f30,pp1,%f30
1516      fmuld   %f24,%f28,%f28
1517      add     %i17,%g1,%i17

1519      faddd   %f6,qq1,%f6
1520      fmuld   %f0,%f4,%f4
1521      add     %i14,%g1,%i14

1523      fmuld   %f8,%f14,%f14

1525      faddd   %f22,qq1,%f22
1526      fmuld   %f16,%f20,%f20
1527      add     %i16,%g1,%i16

1529      fmuld   %f24,%f30,%f30

1531      fmuld   %f2,%f4,%f4

1533      fmuld   %f10,%f14,%f14
1534      ldd     [%i15+8],%f8

1536      fmuld   %f18,%f20,%f20

1538      fmuld   %f26,%f30,%f30
1539      ldd     [%i17+8],%f24

1541      fmuld   %f0,%f6,%f6
1542      faddd   %f4,%f32,%f4
1543      ldd     [%i14+16],%f0

1545      fmuld   %f8,%f12,%f12
1546      faddd   %f34,%f14,%f14

1548      fmuld   %f16,%f22,%f22
1549      faddd   %f20,%f36,%f20
1550      ldd     [%i16+16],%f16

1552      fmuld   %f24,%f28,%f28
1553      faddd   %f38,%f30,%f30

1555      fmuld   %f0,%f6,%f6
1556      faddd   %f4,%f2,%f4
1557      ldd     [%i14+8],%f32

1559      faddd   %f10,%f14,%f14
1560      ldd     [%i15+16],%f34

1562      fmuld   %f16,%f22,%f22
1563      faddd   %f20,%f18,%f20
1564      ldd     [%i16+8],%f36

1566      faddd   %f26,%f30,%f30
1567      ldd     [%i17+16],%f38

1569      fmuld   %f32,%f4,%f4

1571      fmuld   %f34,%f14,%f14

```

```

1573      fmuld   %f36,%f20,%f20
1575      fmuld   %f38,%f30,%f30
1577      fsubd   %f6,%f4,%f6
1579      faddd   %f14,%f12,%f14
1581      fsubd   %f22,%f20,%f22
1583      faddd   %f30,%f28,%f30
1585      faddd   %f6,%f0,%f6
1587      faddd   %f14,%f8,%f14
1589      faddd   %f22,%f16,%f22
1591      faddd   %f30,%f24,%f30
1592      mov     %10,%14
1594      fnegd   %f6,%f4
1595      lda     [%11]%asi,%10      ! preload next argument
1597      fnegd   %f14,%f12
1598      lda     [%11]%asi,%f0
1600      fnegd   %f22,%f20
1601      lda     [%11+4]%asi,%f3
1603      fnegd   %f30,%f28
1604      andn    %10,%i5,%10
1605      add     %i1,%i2,%i1
1607      andcc   %14,2,%g0
1608      fmovdnz %icc,%f4,%f6
1609      st      %f6,[%o0]
1611      andcc   %11,2,%g0
1612      fmovdnz %icc,%f12,%f14
1613      st      %f14,[%o1]
1615      andcc   %12,2,%g0
1616      fmovdnz %icc,%f20,%f22
1617      st      %f22,[%o2]
1619      andcc   %13,2,%g0
1620      fmovdnz %icc,%f28,%f30
1621      st      %f30,[%o3]
1623      addcc   %i0,-1,%i0
1624      bg,pt   %icc,.loop0
1625 ! delay   slot
1626      st      %f7,[%o0+4]
1628      ba,pt   %icc,.end
1629 ! delay   slot
1630      nop
1632      .align  16
1633 .case6:
1634      fmuld   %f26,%f26,%f24
1635      andcc   %13,1,%g0
1636      bz,pn   %icc,.case7
1637 ! delay   slot
1638      fxor    %f30,%f38,%f38

```

```

1640      fmuld   %f8,pp3,%f14      ! sin(x1)
1642      fmuld   %f16,pp3,%f22     ! sin(x2)
1644      fmuld   %f0,qq3,%f6      ! cos(x0)
1646      faddd   %f14,pp2,%f14
1647      fmuld   %f8,qq2,%f12
1649      faddd   %f22,pp2,%f22
1650      fmuld   %f16,qq2,%f20
1652      fmuld   %f24,qq3,%f30     ! cos(x3)
1654      faddd   %f6,qq2,%f6
1655      fmuld   %f0,pp2,%f4
1657      fmuld   %f8,%f14,%f14
1658      faddd   %f12,qq1,%f12
1660      fmuld   %f16,%f22,%f22
1661      faddd   %f20,qq1,%f20
1663      faddd   %f30,qq2,%f30
1664      fmuld   %f24,pp2,%f28
1666      fmuld   %f0,%f6,%f6
1667      faddd   %f4,pp1,%f4
1669      faddd   %f14,pp1,%f14
1670      fmuld   %f8,%f12,%f12
1671      add     %15,%g1,%15
1673      faddd   %f22,pp1,%f22
1674      fmuld   %f16,%f20,%f20
1675      add     %16,%g1,%16
1677      fmuld   %f24,%f30,%f30
1678      faddd   %f28,pp1,%f28
1680      faddd   %f6,qq1,%f6
1681      fmuld   %f0,%f4,%f4
1682      add     %14,%g1,%14
1684      fmuld   %f8,%f14,%f14
1686      fmuld   %f16,%f22,%f22
1688      faddd   %f30,qq1,%f30
1689      fmuld   %f24,%f28,%f28
1690      add     %17,%g1,%17
1692      fmuld   %f2,%f4,%f4
1694      fmuld   %f10,%f14,%f14
1695      ldd     [%15+8],%f8
1697      fmuld   %f18,%f22,%f22
1698      ldd     [%16+8],%f16
1700      fmuld   %f26,%f28,%f28
1702      fmuld   %f0,%f6,%f6
1703      faddd   %f4,%f32,%f4
1704      ldd     [%14+16],%f0

```

```

1706    fmuld    %f8,%f12,%f12
1707    faddd    %f34,%f14,%f14

1709    fmuld    %f16,%f20,%f20
1710    faddd    %f36,%f22,%f22

1712    fmuld    %f24,%f30,%f30
1713    faddd    %f28,%f38,%f28
1714    ldd     [%17+16],%f24

1716    fmuld    %f0,%f6,%f6
1717    faddd    %f4,%f2,%f4
1718    ldd     [%14+8],%f32

1720    faddd    %f10,%f14,%f14
1721    ldd     [%15+16],%f34

1723    faddd    %f18,%f22,%f22
1724    ldd     [%16+16],%f36

1726    fmuld    %f24,%f30,%f30
1727    faddd    %f28,%f26,%f28
1728    ldd     [%17+8],%f38

1730    fmuld    %f32,%f4,%f4

1732    fmuld    %f34,%f14,%f14

1734    fmuld    %f36,%f22,%f22

1736    fmuld    %f38,%f28,%f28

1738    fsubd    %f6,%f4,%f6

1740    faddd    %f14,%f12,%f14

1742    faddd    %f22,%f20,%f22

1744    fsubd    %f30,%f28,%f30

1746    faddd    %f6,%f0,%f6

1748    faddd    %f14,%f8,%f14

1750    faddd    %f22,%f16,%f22

1752    faddd    %f30,%f24,%f30
1753    mov     %l0,%l4

1755    fnegd    %f6,%f4
1756    lda     [%i1]%asi,%l0        ! preload next argument

1758    fnegd    %f14,%f12
1759    lda     [%i1]%asi,%f0

1761    fnegd    %f22,%f20
1762    lda     [%i1+4]%asi,%f3

1764    fnegd    %f30,%f28
1765    andn    %l0,%i5,%l0
1766    add     %i1,%i2,%i1

1768    andcc    %l4,2,%g0
1769    fmovdnz  %icc,%f4,%f6
1770    st      %f6,[%o0]

```

```

1772    andcc    %l1,2,%g0
1773    fmovdnz  %icc,%f12,%f14
1774    st      %f14,[%o1]

1776    andcc    %l2,2,%g0
1777    fmovdnz  %icc,%f20,%f22
1778    st      %f22,[%o2]

1780    andcc    %l3,2,%g0
1781    fmovdnz  %icc,%f28,%f30
1782    st      %f30,[%o3]

1784    addcc    %i0,-1,%i0
1785    bg,pt   %icc,.loop0
1786 ! delay  slot
1787    st      %f7,[%o0+4]

1789    ba,pt   %icc,.end
1790 ! delay  slot
1791    nop

1793    .align  16
1794 .case7:
1795    fmuld    %f8,pp3,%f14        ! sin(x1)

1797    fmuld    %f16,pp3,%f22      ! sin(x2)

1799    fmuld    %f24,pp3,%f30     ! sin(x3)

1801    fmuld    %f0,qq3,%f6       ! cos(x0)

1803    faddd    %f14,pp2,%f14
1804    fmuld    %f8,qq2,%f12

1806    faddd    %f22,pp2,%f22
1807    fmuld    %f16,qq2,%f20

1809    faddd    %f30,pp2,%f30
1810    fmuld    %f24,qq2,%f28

1812    faddd    %f6,qq2,%f6
1813    fmuld    %f0,pp2,%f4

1815    fmuld    %f8,%f14,%f14
1816    faddd    %f12,qq1,%f12

1818    fmuld    %f16,%f22,%f22
1819    faddd    %f20,qq1,%f20

1821    fmuld    %f24,%f30,%f30
1822    faddd    %f28,qq1,%f28

1824    fmuld    %f0,%f6,%f6
1825    faddd    %f4,pp1,%f4

1827    faddd    %f14,pp1,%f14
1828    fmuld    %f8,%f12,%f12
1829    add     %l5,%g1,%l5

1831    faddd    %f22,pp1,%f22
1832    fmuld    %f16,%f20,%f20
1833    add     %l6,%g1,%l6

1835    faddd    %f30,pp1,%f30
1836    fmuld    %f24,%f28,%f28

```

```

1837      add     %l7,%g1,%l7
1839      faddd   %f6,%q1,%f6
1840      fmuld   %f0,%f4,%f4
1841      add     %l4,%g1,%l4
1843      fmuld   %f8,%f14,%f14
1845      fmuld   %f16,%f22,%f22
1847      fmuld   %f24,%f30,%f30
1849      fmuld   %f2,%f4,%f4
1851      fmuld   %f10,%f14,%f14
1852      ldd    [%l5+8],%f8
1854      fmuld   %f18,%f22,%f22
1855      ldd    [%l6+8],%f16
1857      fmuld   %f26,%f30,%f30
1858      ldd    [%l7+8],%f24
1860      fmuld   %f0,%f6,%f6
1861      faddd   %f4,%f32,%f4
1862      ldd    [%l4+16],%f0
1864      fmuld   %f8,%f12,%f12
1865      faddd   %f34,%f14,%f14
1867      fmuld   %f16,%f20,%f20
1868      faddd   %f36,%f22,%f22
1870      fmuld   %f24,%f28,%f28
1871      faddd   %f38,%f30,%f30
1873      fmuld   %f0,%f6,%f6
1874      faddd   %f4,%f2,%f4
1875      ldd    [%l4+8],%f32
1877      faddd   %f10,%f14,%f14
1878      ldd    [%l5+16],%f34
1880      faddd   %f18,%f22,%f22
1881      ldd    [%l6+16],%f36
1883      faddd   %f26,%f30,%f30
1884      ldd    [%l7+16],%f38
1886      fmuld   %f32,%f4,%f4
1888      fmuld   %f34,%f14,%f14
1890      fmuld   %f36,%f22,%f22
1892      fmuld   %f38,%f30,%f30
1894      fsubd   %f6,%f4,%f6
1896      faddd   %f14,%f12,%f14
1898      faddd   %f22,%f20,%f22
1900      faddd   %f30,%f28,%f30
1902      faddd   %f6,%f0,%f6

```

```

1904      faddd   %f14,%f8,%f14
1906      faddd   %f22,%f16,%f22
1908      faddd   %f30,%f24,%f30
1909      mov     %l0,%l4
1911      fnegd   %f6,%f4
1912      lda     [%i1]%asi,%l0      ! preload next argument
1914      fnegd   %f14,%f12
1915      lda     [%i1]%asi,%f0
1917      fnegd   %f22,%f20
1918      lda     [%i1+4]%asi,%f3
1920      fnegd   %f30,%f28
1921      andn    %l0,%i5,%l0
1922      add     %i1,%i2,%i1
1924      andcc   %l4,2,%g0
1925      fmovdnz %icc,%f4,%f6
1926      st      %f6,[%o0]
1928      andcc   %l1,2,%g0
1929      fmovdnz %icc,%f12,%f14
1930      st      %f14,[%o1]
1932      andcc   %l2,2,%g0
1933      fmovdnz %icc,%f20,%f22
1934      st      %f22,[%o2]
1936      andcc   %l3,2,%g0
1937      fmovdnz %icc,%f28,%f30
1938      st      %f30,[%o3]
1940      addcc   %i0,-1,%i0
1941      bg,pt   %icc,.loop0
1942 ! delay   slot
1943      st      %f7,[%o0+4]
1945      ba,pt   %icc,.end
1946 ! delay   slot
1947      nop
1949      .align  16
1950 .case8:
1951      fmuld   %f10,%f10,%f8
1952      andcc   %l1,1,%g0
1953      bz,pn   %icc,.case12
1954 ! delay   slot
1955      fxor   %f14,%f34,%f34
1957      fmuld   %f18,%f18,%f16
1958      andcc   %l2,1,%g0
1959      bz,pn   %icc,.case10
1960 ! delay   slot
1961      fxor   %f22,%f36,%f36
1963      fmuld   %f26,%f26,%f24
1964      andcc   %l3,1,%g0
1965      bz,pn   %icc,.case9
1966 ! delay   slot
1967      fxor   %f30,%f38,%f38

```

```

1969      fmuld   %f0,pp3,%f6          ! sin(x0)
1971      faddd   %f6,pp2,%f6
1972      fmuld   %f0,qq2,%f4
1974      fmuld   %f8,qq3,%f14        ! cos(x1)
1976      fmuld   %f16,qq3,%f22       ! cos(x2)
1978      fmuld   %f24,qq3,%f30       ! cos(x3)
1980      fmuld   %f0,%f6,%f6
1981      faddd   %f4,qq1,%f4
1983      faddd   %f14,qq2,%f14
1984      fmuld   %f8,pp2,%f12
1986      faddd   %f22,qq2,%f22
1987      fmuld   %f16,pp2,%f20
1989      faddd   %f30,qq2,%f30
1990      fmuld   %f24,pp2,%f28
1992      faddd   %f6,pp1,%f6
1993      fmuld   %f0,%f4,%f4
1994      add     %14,%g1,%14
1996      fmuld   %f8,%f14,%f14
1997      faddd   %f12,pp1,%f12
1999      fmuld   %f16,%f22,%f22
2000      faddd   %f20,pp1,%f20
2002      fmuld   %f24,%f30,%f30
2003      faddd   %f28,pp1,%f28
2005      fmuld   %f0,%f6,%f6
2007      faddd   %f14,qq1,%f14
2008      fmuld   %f8,%f12,%f12
2009      add     %15,%g1,%15
2011      faddd   %f22,qq1,%f22
2012      fmuld   %f16,%f20,%f20
2013      add     %16,%g1,%16
2015      faddd   %f30,qq1,%f30
2016      fmuld   %f24,%f28,%f28
2017      add     %17,%g1,%17
2019      fmuld   %f2,%f6,%f6
2020      ldd    [%14+8],%f0
2022      fmuld   %f10,%f12,%f12
2024      fmuld   %f18,%f20,%f20
2026      fmuld   %f26,%f28,%f28
2028      fmuld   %f0,%f4,%f4
2029      faddd   %f32,%f6,%f6
2031      fmuld   %f8,%f14,%f14
2032      faddd   %f12,%f34,%f12
2033      ldd    [%15+16],%f8

```

```

2035      fmuld   %f16,%f22,%f22
2036      faddd   %f20,%f36,%f20
2037      ldd    [%16+16],%f16
2039      fmuld   %f24,%f30,%f30
2040      faddd   %f28,%f38,%f28
2041      ldd    [%17+16],%f24
2043      faddd   %f2,%f6,%f6
2044      ldd    [%14+16],%f32
2046      fmuld   %f8,%f14,%f14
2047      faddd   %f12,%f10,%f12
2048      ldd    [%15+8],%f34
2050      fmuld   %f16,%f22,%f22
2051      faddd   %f20,%f18,%f20
2052      ldd    [%16+8],%f36
2054      fmuld   %f24,%f30,%f30
2055      faddd   %f28,%f26,%f28
2056      ldd    [%17+8],%f38
2058      fmuld   %f32,%f6,%f6
2060      fmuld   %f34,%f12,%f12
2062      fmuld   %f36,%f20,%f20
2064      fmuld   %f38,%f28,%f28
2066      faddd   %f6,%f4,%f6
2068      fsubd   %f14,%f12,%f14
2070      fsubd   %f22,%f20,%f22
2072      fsubd   %f30,%f28,%f30
2074      faddd   %f6,%f0,%f6
2076      faddd   %f14,%f8,%f14
2078      faddd   %f22,%f16,%f22
2080      faddd   %f30,%f24,%f30
2081      mov     %10,%14
2083      fnegd   %f6,%f4
2084      lda     [%i1]%asi,%10          ! preload next argument
2086      fnegd   %f14,%f12
2087      lda     [%i1]%asi,%f0
2089      fnegd   %f22,%f20
2090      lda     [%i1+4]%asi,%f3
2092      fnegd   %f30,%f28
2093      andn    %10,%i5,%10
2094      add     %i1,%i2,%i1
2096      andcc   %14,2,%g0
2097      fmovdnz %icc,%f4,%f6
2098      st      %f6,[%o0]
2100      andcc   %i1,2,%g0

```

```

2101      fmovdnz  %icc,%f12,%f14
2102      st      %f14,[%o1]

2104      andcc   %l2,2,%g0
2105      fmovdnz %icc,%f20,%f22
2106      st      %f22,[%o2]

2108      andcc   %l3,2,%g0
2109      fmovdnz %icc,%f28,%f30
2110      st      %f30,[%o3]

2112      addcc   %i0,-1,%i0
2113      bg,pt   %icc,.loop0
2114 ! delay  slot
2115      st      %f7,[%o0+4]

2117      ba,pt   %icc,.end
2118 ! delay  slot
2119      nop

2121      .align  16
2122 .case9:
2123      fmuld   %f0,pp3,%f6          ! sin(x0)

2125      fmuld   %f24,pp3,%f30       ! sin(x3)

2127      faddd   %f6,pp2,%f6
2128      fmuld   %f0,qq2,%f4

2130      fmuld   %f8,qq3,%f14       ! cos(x1)

2132      fmuld   %f16,qq3,%f22      ! cos(x2)

2134      faddd   %f30,pp2,%f30
2135      fmuld   %f24,qq2,%f28

2137      fmuld   %f0,%f6,%f6
2138      faddd   %f4,qq1,%f4

2140      faddd   %f14,qq2,%f14
2141      fmuld   %f8,pp2,%f12

2143      faddd   %f22,qq2,%f22
2144      fmuld   %f16,pp2,%f20

2146      fmuld   %f24,%f30,%f30
2147      faddd   %f28,qq1,%f28

2149      faddd   %f6,pp1,%f6
2150      fmuld   %f0,%f4,%f4
2151      add     %l4,%g1,%l4

2153      fmuld   %f8,%f14,%f14
2154      faddd   %f12,pp1,%f12

2156      fmuld   %f16,%f22,%f22
2157      faddd   %f20,pp1,%f20

2159      faddd   %f30,pp1,%f30
2160      fmuld   %f24,%f28,%f28
2161      add     %l7,%g1,%l7

2163      fmuld   %f0,%f6,%f6

2165      faddd   %f14,qq1,%f14
2166      fmuld   %f8,%f12,%f12

```

```

2167      add     %l5,%g1,%l5

2169      faddd   %f22,qq1,%f22
2170      fmuld   %f16,%f20,%f20
2171      add     %l6,%g1,%l6

2173      fmuld   %f24,%f30,%f30

2175      fmuld   %f2,%f6,%f6
2176      ldd     [%l4+8],%f0

2178      fmuld   %f10,%f12,%f12

2180      fmuld   %f18,%f20,%f20

2182      fmuld   %f26,%f30,%f30
2183      ldd     [%l7+8],%f24

2185      fmuld   %f0,%f4,%f4
2186      faddd   %f32,%f6,%f6

2188      fmuld   %f8,%f14,%f14
2189      faddd   %f12,%f34,%f12
2190      ldd     [%l5+16],%f8

2192      fmuld   %f16,%f22,%f22
2193      faddd   %f20,%f36,%f20
2194      ldd     [%l6+16],%f16

2196      fmuld   %f24,%f28,%f28
2197      faddd   %f38,%f30,%f30

2199      faddd   %f2,%f6,%f6
2200      ldd     [%l4+16],%f32

2202      fmuld   %f8,%f14,%f14
2203      faddd   %f12,%f10,%f12
2204      ldd     [%l5+8],%f34

2206      fmuld   %f16,%f22,%f22
2207      faddd   %f20,%f18,%f20
2208      ldd     [%l6+8],%f36

2210      faddd   %f26,%f30,%f30
2211      ldd     [%l7+16],%f38

2213      fmuld   %f32,%f6,%f6

2215      fmuld   %f34,%f12,%f12

2217      fmuld   %f36,%f20,%f20

2219      fmuld   %f38,%f30,%f30

2221      faddd   %f6,%f4,%f6

2223      fsubd   %f14,%f12,%f14

2225      fsubd   %f22,%f20,%f22

2227      faddd   %f30,%f28,%f30

2229      faddd   %f6,%f0,%f6

2231      faddd   %f14,%f8,%f14

```



```

2233      fadd    %f22,%f16,%f22
2235      fadd    %f30,%f24,%f30
2236      mov     %l0,%l4

2238      fnegd   %f6,%f4
2239      lda     [%i1]%asi,%l0      ! preload next argument

2241      fnegd   %f14,%f12
2242      lda     [%i1]%asi,%f0

2244      fnegd   %f22,%f20
2245      lda     [%i1+4]%asi,%f3

2247      fnegd   %f30,%f28
2248      andn    %l0,%i5,%l0
2249      add     %i1,%i2,%i1

2251      andcc   %l4,2,%g0
2252      fmovdnz %icc,%f4,%f6
2253      st      %f6,[%o0]

2255      andcc   %l1,2,%g0
2256      fmovdnz %icc,%f12,%f14
2257      st      %f14,[%o1]

2259      andcc   %l2,2,%g0
2260      fmovdnz %icc,%f20,%f22
2261      st      %f22,[%o2]

2263      andcc   %l3,2,%g0
2264      fmovdnz %icc,%f28,%f30
2265      st      %f30,[%o3]

2267      addcc   %i0,-1,%i0
2268      bg,pt   %icc,.loop0
2269      ! delay slot
2270      st      %f7,[%o0+4]

2272      ba,pt   %icc,.end
2273      ! delay slot
2274      nop

2276      .align 16
2277      .case10:
2278      fmuld   %f26,%f26,%f24
2279      andcc   %l3,1,%g0
2280      bz,pn   %icc,.case11
2281      ! delay slot
2282      fxor    %f30,%f38,%f38

2284      fmuld   %f0,pp3,%f6      ! sin(x0)

2286      fmuld   %f16,pp3,%f22    ! sin(x2)

2288      fadd    %f6,pp2,%f6
2289      fmuld   %f0,qq2,%f4

2291      fmuld   %f8,qq3,%f14    ! cos(x1)

2293      fadd    %f22,pp2,%f22
2294      fmuld   %f16,qq2,%f20

2296      fmuld   %f24,qq3,%f30    ! cos(x3)

2298      fmuld   %f0,%f6,%f6

```

```

2299      fadd    %f4,qq1,%f4

2301      fadd    %f14,qq2,%f14
2302      fmuld   %f8,pp2,%f12

2304      fmuld   %f16,%f22,%f22
2305      fadd    %f20,qq1,%f20

2307      fadd    %f30,qq2,%f30
2308      fmuld   %f24,pp2,%f28

2310      fadd    %f6,pp1,%f6
2311      fmuld   %f0,%f4,%f4
2312      add     %l4,%g1,%l4

2314      fmuld   %f8,%f14,%f14
2315      fadd    %f12,pp1,%f12

2317      fadd    %f22,pp1,%f22
2318      fmuld   %f16,%f20,%f20
2319      add     %l6,%g1,%l6

2321      fmuld   %f24,%f30,%f30
2322      fadd    %f28,pp1,%f28

2324      fmuld   %f0,%f6,%f6

2326      fadd    %f14,qq1,%f14
2327      fmuld   %f8,%f12,%f12
2328      add     %l5,%g1,%l5

2330      fmuld   %f16,%f22,%f22

2332      fadd    %f30,qq1,%f30
2333      fmuld   %f24,%f28,%f28
2334      add     %l7,%g1,%l7

2336      fmuld   %f2,%f6,%f6
2337      ldd     [%l4+8],%f0

2339      fmuld   %f10,%f12,%f12

2341      fmuld   %f18,%f22,%f22
2342      ldd     [%l6+8],%f16

2344      fmuld   %f26,%f28,%f28

2346      fmuld   %f0,%f4,%f4
2347      fadd    %f32,%f6,%f6

2349      fmuld   %f8,%f14,%f14
2350      fadd    %f12,%f34,%f12
2351      ldd     [%l5+16],%f8

2353      fmuld   %f16,%f20,%f20
2354      fadd    %f36,%f22,%f22

2356      fmuld   %f24,%f30,%f30
2357      fadd    %f28,%f38,%f28
2358      ldd     [%l7+16],%f24

2360      fadd    %f2,%f6,%f6
2361      ldd     [%l4+16],%f32

2363      fmuld   %f8,%f14,%f14
2364      fadd    %f12,%f10,%f12

```

```

2365     ldd     [%15+8],%f34
2367     faddd   %f18,%f22,%f22
2368     ldd     [%16+16],%f36
2370     fmuld   %f24,%f30,%f30
2371     faddd   %f28,%f26,%f28
2372     ldd     [%17+8],%f38
2374     fmuld   %f32,%f6,%f6
2376     fmuld   %f34,%f12,%f12
2378     fmuld   %f36,%f22,%f22
2380     fmuld   %f38,%f28,%f28
2382     faddd   %f6,%f4,%f6
2384     fsubd   %f14,%f12,%f14
2386     faddd   %f22,%f20,%f22
2388     fsubd   %f30,%f28,%f30
2390     faddd   %f6,%f0,%f6
2392     faddd   %f14,%f8,%f14
2394     faddd   %f22,%f16,%f22
2396     faddd   %f30,%f24,%f30
2397     mov     %10,%14
2399     fnegd   %f6,%f4
2400     lda     [%i1]%asi,%10      ! preload next argument
2402     fnegd   %f14,%f12
2403     lda     [%i1]%asi,%f0
2405     fnegd   %f22,%f20
2406     lda     [%i1+4]%asi,%f3
2408     fnegd   %f30,%f28
2409     andn   %10,%i5,%10
2410     add    %i1,%i2,%i1
2412     andcc  %14,2,%g0
2413     fmovdnz %icc,%f4,%f6
2414     st     %f6,[%o0]
2416     andcc  %11,2,%g0
2417     fmovdnz %icc,%f12,%f14
2418     st     %f14,[%o1]
2420     andcc  %12,2,%g0
2421     fmovdnz %icc,%f20,%f22
2422     st     %f22,[%o2]
2424     andcc  %13,2,%g0
2425     fmovdnz %icc,%f28,%f30
2426     st     %f30,[%o3]
2428     addcc  %i0,-1,%i0
2429     bg,pt  %icc,.loop0
2430 ! delay slot

```

```

2431     st     %f7,[%o0+4]
2433     ba,pt  %icc,.end
2434 ! delay slot
2435     nop
2437     .align 16
2438 .case11:
2439     fmuld   %f0,pp3,%f6      ! sin(x0)
2441     fmuld   %f16,pp3,%f22   ! sin(x2)
2443     fmuld   %f24,pp3,%f30   ! sin(x3)
2445     faddd   %f6,pp2,%f6
2446     fmuld   %f0,qq2,%f4
2448     fmuld   %f8,qq3,%f14    ! cos(x1)
2450     faddd   %f22,pp2,%f22
2451     fmuld   %f16,qq2,%f20
2453     faddd   %f30,pp2,%f30
2454     fmuld   %f24,qq2,%f28
2456     fmuld   %f0,%f6,%f6
2457     faddd   %f4,qq1,%f4
2459     faddd   %f14,qq2,%f14
2460     fmuld   %f8,pp2,%f12
2462     fmuld   %f16,%f22,%f22
2463     faddd   %f20,qq1,%f20
2465     fmuld   %f24,%f30,%f30
2466     faddd   %f28,qq1,%f28
2468     faddd   %f6,pp1,%f6
2469     fmuld   %f0,%f4,%f4
2470     add    %14,%g1,%14
2472     fmuld   %f8,%f14,%f14
2473     faddd   %f12,pp1,%f12
2475     faddd   %f22,pp1,%f22
2476     fmuld   %f16,%f20,%f20
2477     add    %16,%g1,%16
2479     faddd   %f30,pp1,%f30
2480     fmuld   %f24,%f28,%f28
2481     add    %17,%g1,%17
2483     fmuld   %f0,%f6,%f6
2485     faddd   %f14,qq1,%f14
2486     fmuld   %f8,%f12,%f12
2487     add    %15,%g1,%15
2489     fmuld   %f16,%f22,%f22
2491     fmuld   %f24,%f30,%f30
2493     fmuld   %f2,%f6,%f6
2494     ldd     [%14+8],%f0
2496     fmuld   %f10,%f12,%f12

```

```

2498    fmuld    %f18,%f22,%f22
2499    ldd      [%16+8],%f16

2501    fmuld    %f26,%f30,%f30
2502    ldd      [%17+8],%f24

2504    fmuld    %f0,%f4,%f4
2505    faddd    %f32,%f6,%f6

2507    fmuld    %f8,%f14,%f14
2508    faddd    %f12,%f34,%f12
2509    ldd      [%15+16],%f8

2511    fmuld    %f16,%f20,%f20
2512    faddd    %f36,%f22,%f22

2514    fmuld    %f24,%f28,%f28
2515    faddd    %f38,%f30,%f30

2517    faddd    %f2,%f6,%f6
2518    ldd      [%14+16],%f32

2520    fmuld    %f8,%f14,%f14
2521    faddd    %f12,%f10,%f12
2522    ldd      [%15+8],%f34

2524    faddd    %f18,%f22,%f22
2525    ldd      [%16+16],%f36

2527    faddd    %f26,%f30,%f30
2528    ldd      [%17+16],%f38

2530    fmuld    %f32,%f6,%f6

2532    fmuld    %f34,%f12,%f12

2534    fmuld    %f36,%f22,%f22

2536    fmuld    %f38,%f30,%f30

2538    faddd    %f6,%f4,%f6

2540    fsubd    %f14,%f12,%f14

2542    faddd    %f22,%f20,%f22

2544    faddd    %f30,%f28,%f30

2546    faddd    %f6,%f0,%f6

2548    faddd    %f14,%f8,%f14

2550    faddd    %f22,%f16,%f22

2552    faddd    %f30,%f24,%f30
2553    mov     %10,%14

2555    fnegd    %f6,%f4
2556    lda     [%i1]%asi,%10        ! preload next argument

2558    fnegd    %f14,%f12
2559    lda     [%i1]%asi,%f0

2561    fnegd    %f22,%f20
2562    lda     [%i1+4]%asi,%f3

```

```

2564    fnegd    %f30,%f28
2565    andn    %10,%i5,%10
2566    add     %i1,%i2,%i1

2568    andcc    %14,2,%g0
2569    fmovdnz  %icc,%f4,%f6
2570    st      %f6,[%o0]

2572    andcc    %11,2,%g0
2573    fmovdnz  %icc,%f12,%f14
2574    st      %f14,[%o1]

2576    andcc    %12,2,%g0
2577    fmovdnz  %icc,%f20,%f22
2578    st      %f22,[%o2]

2580    andcc    %13,2,%g0
2581    fmovdnz  %icc,%f28,%f30
2582    st      %f30,[%o3]

2584    addcc    %i0,-1,%i0
2585    bg,pt   %icc,.loop0
2586 ! delay  slot
2587    st      %f7,[%o0+4]

2589    ba,pt   %icc,.end
2590 ! delay  slot
2591    nop

2593    .align  16
2594    .case12:
2595    fmuld    %f18,%f18,%f16
2596    andcc    %12,1,%g0
2597    bz,pn   %icc,.case14
2598 ! delay  slot
2599    fxor    %f22,%f36,%f36

2601    fmuld    %f26,%f26,%f24
2602    andcc    %13,1,%g0
2603    bz,pn   %icc,.case13
2604 ! delay  slot
2605    fxor    %f30,%f38,%f38

2607    fmuld    %f0,pp3,%f6                ! sin(x0)

2609    fmuld    %f8,pp3,%f14              ! sin(x1)

2611    faddd    %f6,pp2,%f6
2612    fmuld    %f0,qq2,%f4

2614    faddd    %f14,pp2,%f14
2615    fmuld    %f8,qq2,%f12

2617    fmuld    %f16,qq3,%f22            ! cos(x2)

2619    fmuld    %f24,qq3,%f30            ! cos(x3)

2621    fmuld    %f0,%f6,%f6
2622    faddd    %f4,qq1,%f4

2624    fmuld    %f8,%f14,%f14
2625    faddd    %f12,qq1,%f12

2627    faddd    %f22,qq2,%f22
2628    fmuld    %f16,pp2,%f20

```

```

2630      fadd    %f30,qq2,%f30
2631      fmuld   %f24,pp2,%f28

2633      fadd    %f6,pp1,%f6
2634      fmuld   %f0,%f4,%f4
2635      add     %l4,%g1,%l4

2637      fadd    %f14,pp1,%f14
2638      fmuld   %f8,%f12,%f12
2639      add     %l5,%g1,%l5

2641      fmuld   %f16,%f22,%f22
2642      fadd    %f20,pp1,%f20

2644      fmuld   %f24,%f30,%f30
2645      fadd    %f28,pp1,%f28

2647      fmuld   %f0,%f6,%f6

2649      fmuld   %f8,%f14,%f14

2651      fadd    %f22,qq1,%f22
2652      fmuld   %f16,%f20,%f20
2653      add     %l6,%g1,%l6

2655      fadd    %f30,qq1,%f30
2656      fmuld   %f24,%f28,%f28
2657      add     %l7,%g1,%l7

2659      fmuld   %f2,%f6,%f6
2660      ldd    [%l4+8],%f0

2662      fmuld   %f10,%f14,%f14
2663      ldd    [%l5+8],%f8

2665      fmuld   %f18,%f20,%f20

2667      fmuld   %f26,%f28,%f28

2669      fmuld   %f0,%f4,%f4
2670      fadd    %f32,%f6,%f6

2672      fmuld   %f8,%f12,%f12
2673      fadd    %f34,%f14,%f14

2675      fmuld   %f16,%f22,%f22
2676      fadd    %f20,%f36,%f20
2677      ldd    [%l6+16],%f16

2679      fmuld   %f24,%f30,%f30
2680      fadd    %f28,%f38,%f28
2681      ldd    [%l7+16],%f24

2683      fadd    %f2,%f6,%f6
2684      ldd    [%l4+16],%f32

2686      fadd    %f10,%f14,%f14
2687      ldd    [%l5+16],%f34

2689      fmuld   %f16,%f22,%f22
2690      fadd    %f20,%f18,%f20
2691      ldd    [%l6+8],%f36

2693      fmuld   %f24,%f30,%f30
2694      fadd    %f28,%f26,%f28

```

```

2695      ldd    [%l7+8],%f38

2697      fmuld   %f32,%f6,%f6

2699      fmuld   %f34,%f14,%f14

2701      fmuld   %f36,%f20,%f20

2703      fmuld   %f38,%f28,%f28

2705      fadd    %f6,%f4,%f6

2707      fadd    %f14,%f12,%f14

2709      fsubd   %f22,%f20,%f22

2711      fsubd   %f30,%f28,%f30

2713      fadd    %f6,%f0,%f6

2715      fadd    %f14,%f8,%f14

2717      fadd    %f22,%f16,%f22

2719      fadd    %f30,%f24,%f30
2720      mov     %l0,%l4

2722      fnegd   %f6,%f4
2723      lda     [%l1]asi,%l0      ! preload next argument

2725      fnegd   %f14,%f12
2726      lda     [%l1]asi,%f0

2728      fnegd   %f22,%f20
2729      lda     [%l1+4]asi,%f3

2731      fnegd   %f30,%f28
2732      andn    %l0,%i5,%l0
2733      add     %i1,%i2,%i1

2735      andcc   %l4,2,%g0
2736      fmovdnz %icc,%f4,%f6
2737      st      %f6,[%o0]

2739      andcc   %l1,2,%g0
2740      fmovdnz %icc,%f12,%f14
2741      st      %f14,[%o1]

2743      andcc   %l2,2,%g0
2744      fmovdnz %icc,%f20,%f22
2745      st      %f22,[%o2]

2747      andcc   %l3,2,%g0
2748      fmovdnz %icc,%f28,%f30
2749      st      %f30,[%o3]

2751      addcc   %i0,-1,%i0
2752      bg,pt   %icc,.loop0
2753      ! delay slot
2754      st      %f7,[%o0+4]

2756      ba,pt   %icc,.end
2757      ! delay slot
2758      nop

2760      .align 16

```

```

2761 .case13:
2762     fmuld    %f0,pp3,%f6          ! sin(x0)

2764     fmuld    %f8,pp3,%f14       ! sin(x1)

2766     fmuld    %f24,pp3,%f30      ! sin(x3)

2768     faddd    %f6,pp2,%f6
2769     fmuld    %f0,qq2,%f4

2771     faddd    %f14,pp2,%f14
2772     fmuld    %f8,qq2,%f12

2774     fmuld    %f16,qq3,%f22     ! cos(x2)

2776     faddd    %f30,pp2,%f30
2777     fmuld    %f24,qq2,%f28

2779     fmuld    %f0,%f6,%f6
2780     faddd    %f4,qq1,%f4

2782     fmuld    %f8,%f14,%f14
2783     faddd    %f12,qq1,%f12

2785     faddd    %f22,qq2,%f22
2786     fmuld    %f16,pp2,%f20

2788     fmuld    %f24,%f30,%f30
2789     faddd    %f28,qq1,%f28

2791     faddd    %f6,pp1,%f6
2792     fmuld    %f0,%f4,%f4
2793     add     %l4,%g1,%l4

2795     faddd    %f14,pp1,%f14
2796     fmuld    %f8,%f12,%f12
2797     add     %l5,%g1,%l5

2799     fmuld    %f16,%f22,%f22
2800     faddd    %f20,pp1,%f20

2802     faddd    %f30,pp1,%f30
2803     fmuld    %f24,%f28,%f28
2804     add     %l7,%g1,%l7

2806     fmuld    %f0,%f6,%f6

2808     fmuld    %f8,%f14,%f14

2810     faddd    %f22,qq1,%f22
2811     fmuld    %f16,%f20,%f20
2812     add     %l6,%g1,%l6

2814     fmuld    %f24,%f30,%f30

2816     fmuld    %f2,%f6,%f6
2817     ldd     [%l4+8],%f0

2819     fmuld    %f10,%f14,%f14
2820     ldd     [%l5+8],%f8

2822     fmuld    %f18,%f20,%f20

2824     fmuld    %f26,%f30,%f30
2825     ldd     [%l7+8],%f24

```

```

2827     fmuld    %f0,%f4,%f4
2828     faddd    %f32,%f6,%f6

2830     fmuld    %f8,%f12,%f12
2831     faddd    %f34,%f14,%f14

2833     fmuld    %f16,%f22,%f22
2834     faddd    %f20,%f36,%f20
2835     ldd     [%l6+16],%f16

2837     fmuld    %f24,%f28,%f28
2838     faddd    %f38,%f30,%f30

2840     faddd    %f2,%f6,%f6
2841     ldd     [%l4+16],%f32

2843     faddd    %f10,%f14,%f14
2844     ldd     [%l5+16],%f34

2846     fmuld    %f16,%f22,%f22
2847     faddd    %f20,%f18,%f20
2848     ldd     [%l6+8],%f36

2850     faddd    %f26,%f30,%f30
2851     ldd     [%l7+16],%f38

2853     fmuld    %f32,%f6,%f6

2855     fmuld    %f34,%f14,%f14

2857     fmuld    %f36,%f20,%f20

2859     fmuld    %f38,%f30,%f30

2861     faddd    %f6,%f4,%f6

2863     faddd    %f14,%f12,%f14

2865     fsubd    %f22,%f20,%f22

2867     faddd    %f30,%f28,%f30

2869     faddd    %f6,%f0,%f6

2871     faddd    %f14,%f8,%f14

2873     faddd    %f22,%f16,%f22

2875     faddd    %f30,%f24,%f30
2876     mov     %l0,%l4

2878     fnegd    %f6,%f4
2879     lda     [%l1]asi,%l0          ! preload next argument

2881     fnegd    %f14,%f12
2882     lda     [%l1]asi,%f0

2884     fnegd    %f22,%f20
2885     lda     [%l1+4]asi,%f3

2887     fnegd    %f30,%f28
2888     andn    %l0,%i5,%l0
2889     add     %l1,%i2,%l1

2891     andcc   %l4,2,%g0
2892     fmovdzn %icc,%f4,%f6

```

```

2893      st      %f6, [%o0]

2895      andcc   %l1, 2, %g0
2896      fmovdnz %icc, %f12, %f14
2897      st      %f14, [%o1]

2899      andcc   %l2, 2, %g0
2900      fmovdnz %icc, %f20, %f22
2901      st      %f22, [%o2]

2903      andcc   %l3, 2, %g0
2904      fmovdnz %icc, %f28, %f30
2905      st      %f30, [%o3]

2907      addcc   %i0, -1, %i0
2908      bg, pt  %icc, .loop0
2909 ! delay slot
2910      st      %f7, [%o0+4]

2912      ba, pt  %icc, .end
2913 ! delay slot
2914      nop

2916      .align  16
2917 .case14:
2918      fmuld   %f26, %f26, %f24
2919      andcc   %l3, 1, %g0
2920      bz, pn  %icc, .case15
2921 ! delay slot
2922      fxor   %f30, %f38, %f38

2924      fmuld   %f0, pp3, %f6          ! sin(x0)

2926      fmuld   %f8, pp3, %f14        ! sin(x1)

2928      fmuld   %f16, pp3, %f22       ! sin(x2)

2930      faddd   %f6, pp2, %f6
2931      fmuld   %f0, qq2, %f4

2933      faddd   %f14, pp2, %f14
2934      fmuld   %f8, qq2, %f12

2936      faddd   %f22, pp2, %f22
2937      fmuld   %f16, qq2, %f20

2939      fmuld   %f24, qq3, %f30       ! cos(x3)

2941      fmuld   %f0, %f6, %f6
2942      faddd   %f4, qq1, %f4

2944      fmuld   %f8, %f14, %f14
2945      faddd   %f12, qq1, %f12

2947      fmuld   %f16, %f22, %f22
2948      faddd   %f20, qq1, %f20

2950      faddd   %f30, qq2, %f30
2951      fmuld   %f24, pp2, %f28

2953      faddd   %f6, pp1, %f6
2954      fmuld   %f0, %f4, %f4
2955      add     %l4, %g1, %l4

2957      faddd   %f14, pp1, %f14
2958      fmuld   %f8, %f12, %f12

```

```

2959      add     %l5, %g1, %l5

2961      faddd   %f22, pp1, %f22
2962      fmuld   %f16, %f20, %f20
2963      add     %l6, %g1, %l6

2965      fmuld   %f24, %f30, %f30
2966      faddd   %f28, pp1, %f28

2968      fmuld   %f0, %f6, %f6

2970      fmuld   %f8, %f14, %f14

2972      fmuld   %f16, %f22, %f22

2974      faddd   %f30, qq1, %f30
2975      fmuld   %f24, %f28, %f28
2976      add     %l7, %g1, %l7

2978      fmuld   %f2, %f6, %f6
2979      ldd     [%l4+8], %f0

2981      fmuld   %f10, %f14, %f14
2982      ldd     [%l5+8], %f8

2984      fmuld   %f18, %f22, %f22
2985      ldd     [%l6+8], %f16

2987      fmuld   %f26, %f28, %f28

2989      fmuld   %f0, %f4, %f4
2990      faddd   %f32, %f6, %f6

2992      fmuld   %f8, %f12, %f12
2993      faddd   %f34, %f14, %f14

2995      fmuld   %f16, %f20, %f20
2996      faddd   %f36, %f22, %f22

2998      fmuld   %f24, %f30, %f30
2999      faddd   %f28, %f38, %f28
3000      ldd     [%l7+16], %f24

3002      faddd   %f2, %f6, %f6
3003      ldd     [%l4+16], %f32

3005      faddd   %f10, %f14, %f14
3006      ldd     [%l5+16], %f34

3008      faddd   %f18, %f22, %f22
3009      ldd     [%l6+16], %f36

3011      fmuld   %f24, %f30, %f30
3012      faddd   %f28, %f26, %f28
3013      ldd     [%l7+8], %f38

3015      fmuld   %f32, %f6, %f6

3017      fmuld   %f34, %f14, %f14

3019      fmuld   %f36, %f22, %f22

3021      fmuld   %f38, %f28, %f28

3023      faddd   %f6, %f4, %f6

```

```

3025      fadd    %f14,%f12,%f14
3027      fadd    %f22,%f20,%f22
3029      fsubd   %f30,%f28,%f30
3031      fadd    %f6,%f0,%f6
3033      fadd    %f14,%f8,%f14
3035      fadd    %f22,%f16,%f22
3037      fadd    %f30,%f24,%f30
3038      mov     %l0,%l4
3040      fnegd   %f6,%f4
3041      lda     [%i1]%asi,%l0      ! preload next argument
3043      fnegd   %f14,%f12
3044      lda     [%i1]%asi,%f0
3046      fnegd   %f22,%f20
3047      lda     [%i1+4]%asi,%f3
3049      fnegd   %f30,%f28
3050      andn    %l0,%i5,%l0
3051      add     %i1,%i2,%i1
3053      andcc   %l4,2,%g0
3054      fmovdnz %icc,%f4,%f6
3055      st      %f6,[%o0]
3057      andcc   %l1,2,%g0
3058      fmovdnz %icc,%f12,%f14
3059      st      %f14,[%o1]
3061      andcc   %l2,2,%g0
3062      fmovdnz %icc,%f20,%f22
3063      st      %f22,[%o2]
3065      andcc   %l3,2,%g0
3066      fmovdnz %icc,%f28,%f30
3067      st      %f30,[%o3]
3069      addcc   %i0,-1,%i0
3070      bg,pt   %icc,.loop0
3071      ! delay slot
3072      st      %f7,[%o0+4]
3074      ba,pt   %icc,.end
3075      ! delay slot
3076      nop
3078      .align  16
3079      .case15:
3080      fmuld   %f0,pp3,%f6      ! sin(x0)
3082      fmuld   %f8,pp3,%f14     ! sin(x1)
3084      fmuld   %f16,pp3,%f22   ! sin(x2)
3086      fmuld   %f24,pp3,%f30   ! sin(x3)
3088      fadd    %f6,pp2,%f6
3089      fmuld   %f0,qq2,%f4

```

```

3091      fadd    %f14,pp2,%f14
3092      fmuld   %f8,qq2,%f12
3094      fadd    %f22,pp2,%f22
3095      fmuld   %f16,qq2,%f20
3097      fadd    %f30,pp2,%f30
3098      fmuld   %f24,qq2,%f28
3100      fmuld   %f0,%f6,%f6
3101      fadd    %f4,qq1,%f4
3103      fmuld   %f8,%f14,%f14
3104      fadd    %f12,qq1,%f12
3106      fmuld   %f16,%f22,%f22
3107      fadd    %f20,qq1,%f20
3109      fmuld   %f24,%f30,%f30
3110      fadd    %f28,qq1,%f28
3112      fadd    %f6,pp1,%f6
3113      fmuld   %f0,%f4,%f4
3114      add     %l4,%g1,%l4
3116      fadd    %f14,pp1,%f14
3117      fmuld   %f8,%f12,%f12
3118      add     %l5,%g1,%l5
3120      fadd    %f22,pp1,%f22
3121      fmuld   %f16,%f20,%f20
3122      add     %l6,%g1,%l6
3124      fadd    %f30,pp1,%f30
3125      fmuld   %f24,%f28,%f28
3126      add     %l7,%g1,%l7
3128      fmuld   %f0,%f6,%f6
3130      fmuld   %f8,%f14,%f14
3132      fmuld   %f16,%f22,%f22
3134      fmuld   %f24,%f30,%f30
3136      fmuld   %f2,%f6,%f6
3137      ldd     [%l4+8],%f0
3139      fmuld   %f10,%f14,%f14
3140      ldd     [%l5+8],%f8
3142      fmuld   %f18,%f22,%f22
3143      ldd     [%l6+8],%f16
3145      fmuld   %f26,%f30,%f30
3146      ldd     [%l7+8],%f24
3148      fmuld   %f0,%f4,%f4
3149      fadd    %f32,%f6,%f6
3151      fmuld   %f8,%f12,%f12
3152      fadd    %f34,%f14,%f14
3154      fmuld   %f16,%f20,%f20
3155      fadd    %f36,%f22,%f22

```

```

3157      fmuld   %f24,%f28,%f28
3158      faddd   %f38,%f30,%f30

3160      faddd   %f2,%f6,%f6
3161      ldd     [%14+16],%f32

3163      faddd   %f10,%f14,%f14
3164      ldd     [%15+16],%f34

3166      faddd   %f18,%f22,%f22
3167      ldd     [%16+16],%f36

3169      faddd   %f26,%f30,%f30
3170      ldd     [%17+16],%f38

3172      fmuld   %f32,%f6,%f6

3174      fmuld   %f34,%f14,%f14

3176      fmuld   %f36,%f22,%f22

3178      fmuld   %f38,%f30,%f30

3180      faddd   %f6,%f4,%f6

3182      faddd   %f14,%f12,%f14

3184      faddd   %f22,%f20,%f22

3186      faddd   %f30,%f28,%f30

3188      faddd   %f6,%f0,%f6

3190      faddd   %f14,%f8,%f14

3192      faddd   %f22,%f16,%f22

3194      faddd   %f30,%f24,%f30
3195      mov     %10,%14

3197      fnegd   %f6,%f4
3198      lda     [%i1]%asi,%10      ! preload next argument

3200      fnegd   %f14,%f12
3201      lda     [%i1]%asi,%f0

3203      fnegd   %f22,%f20
3204      lda     [%i1+4]%asi,%f3

3206      fnegd   %f30,%f28
3207      andn    %10,%i5,%10
3208      add     %i1,%i2,%i1

3210      andcc   %14,2,%g0
3211      fmovdnz %icc,%f4,%f6
3212      st      %f6,[%o0]

3214      andcc   %11,2,%g0
3215      fmovdnz %icc,%f12,%f14
3216      st      %f14,[%o1]

3218      andcc   %12,2,%g0
3219      fmovdnz %icc,%f20,%f22
3220      st      %f22,[%o2]

3222      andcc   %13,2,%g0

```

```

3223      fmovdnz %icc,%f28,%f30
3224      st      %f30,[%o3]

3226      addcc   %i0,-1,%i0
3227      bg,pt   %icc,.loop0
3228 ! delay   slot
3229      st      %f7,[%o0+4]

3231      ba,pt   %icc,.end
3232 ! delay   slot
3233      nop

3236      .align  16
3237 .end:
3238      st      %f15,[%o1+4]
3239      st      %f23,[%o2+4]
3240      st      %f31,[%o3+4]
3241      ld      [%fp+biguns],%i5
3242      tst     %i5      ! check for huge arguments remaining
3243      be,pt   %icc,.exit
3244 ! delay   slot
3245      nop
3246 #ifdef   __sparcv9
3247      ldx     [%fp+xsave],%o1
3248      ldx     [%fp+ysave],%o3
3249 #else
3250      ld      [%fp+xsave],%o1
3251      ld      [%fp+ysave],%o3
3252 #endif
3253      ld      [%fp+nsave],%o0
3254      ld      [%fp+sxsave],%o2
3255      ld      [%fp+sysave],%o4
3256      sra     %o2,0,%o2      ! sign-extend for V9
3257      sra     %o4,0,%o4
3258      call    __vlibm_vsin_big_ultra3
3259      sra     %o5,0,%o5      ! delay slot

3261 .exit:
3262      ret
3263      restore

3266      .align  16
3267 .last1:
3268      faddd   %f2,c3two44,%f4
3269      st      %f15,[%o1+4]
3270 .last1_from_range1:
3271      mov     0,%l1
3272      fzeros  %f8
3273      fzero   %f10
3274      add     %fp,junk,%o1
3275 .last2:
3276      faddd   %f10,c3two44,%f12
3277      st      %f23,[%o2+4]
3278 .last2_from_range2:
3279      mov     0,%l2
3280      fzeros  %f16
3281      fzero   %f18
3282      add     %fp,junk,%o2
3283 .last3:
3284      faddd   %f18,c3two44,%f20
3285      st      %f31,[%o3+4]
3286      st      %f5,[%fp+nk0]
3287      st      %f13,[%fp+nk1]
3288 .last3_from_range3:

```



```

3289      mov     0,%l3
3290      fzeros  %f24
3291      fzero   %f26
3292      ba,pt   %icc,.cont
3293 ! delay slot
3294      add     %fp,junk,%o3

3297      .align  16
3298 .range0:
3299      cmp     %l0,%o4
3300      bl,pt   %icc,1f          ! hx < 0x3e400000
3301 ! delay slot, harmless if branch taken
3302      sethi   %hi(0x7ff00000),%o7
3303      cmp     %l0,%o7
3304      bl,a,pt %icc,2f          ! branch if finite
3305 ! delay slot, squashed if branch not taken
3306      st      %o4,[%fp+biguns] ! set biguns
3307      fzero   %f0
3308      fmuld   %f2,%f0,%f2
3309      st      %f2,[%o0]
3310      ba,pt   %icc,2f
3311 ! delay slot
3312      st      %f3,[%o0+4]
3313 1:
3314      fdtoi   %f2,%f4          ! raise inexact if not zero
3315      st      %f0,[%o0]
3316      st      %f3,[%o0+4]
3317 2:
3318      addcc   %i0,-1,%i0
3319      ble,pn  %icc,.end
3320 ! delay slot, harmless if branch taken
3321      add     %i3,%i4,%i3      ! y += stridey
3322      andn    %l1,%i5,%l0      ! hx &= ~0x80000000
3323      fmovs   %f8,%f0
3324      fmovs   %f11,%f3
3325      ba,pt   %icc,.loop0
3326 ! delay slot
3327      add     %i1,%i2,%i1      ! x += stride

3330      .align  16
3331 .rangel:
3332      cmp     %l1,%o4
3333      bl,pt   %icc,1f          ! hx < 0x3e400000
3334 ! delay slot, harmless if branch taken
3335      sethi   %hi(0x7ff00000),%o7
3336      cmp     %l1,%o7
3337      bl,a,pt %icc,2f          ! branch if finite
3338 ! delay slot, squashed if branch not taken
3339      st      %o4,[%fp+biguns] ! set biguns
3340      fzero   %f8
3341      fmuld   %f10,%f8,%f10
3342      st      %f10,[%o1]
3343      ba,pt   %icc,2f
3344 ! delay slot
3345      st      %f11,[%o1+4]
3346 1:
3347      fdtoi   %f10,%f12        ! raise inexact if not zero
3348      st      %f8,[%o1]
3349      st      %f11,[%o1+4]
3350 2:
3351      addcc   %i0,-1,%i0
3352      ble,pn  %icc,.last1_from_rangel
3353 ! delay slot, harmless if branch taken
3354      add     %i3,%i4,%i3      ! y += stridey

```

```

3355      andn    %l2,%i5,%l1      ! hx &= ~0x80000000
3356      fmovs   %f16,%f8
3357      fmovs   %f19,%f11
3358      ba,pt   %icc,.loop1
3359 ! delay slot
3360      add     %i1,%i2,%i1      ! x += stride

3363      .align  16
3364 .range2:
3365      cmp     %l2,%o4
3366      bl,pt   %icc,1f          ! hx < 0x3e400000
3367 ! delay slot, harmless if branch taken
3368      sethi   %hi(0x7ff00000),%o7
3369      cmp     %l2,%o7
3370      bl,a,pt %icc,2f          ! branch if finite
3371 ! delay slot, squashed if branch not taken
3372      st      %o4,[%fp+biguns] ! set biguns
3373      fzero   %f16
3374      fmuld   %f18,%f16,%f18
3375      st      %f18,[%o2]
3376      ba,pt   %icc,2f
3377 ! delay slot
3378      st      %f19,[%o2+4]
3379 1:
3380      fdtoi   %f18,%f20        ! raise inexact if not zero
3381      st      %f16,[%o2]
3382      st      %f19,[%o2+4]
3383 2:
3384      addcc   %i0,-1,%i0
3385      ble,pn  %icc,.last2_from_range2
3386 ! delay slot, harmless if branch taken
3387      add     %i3,%i4,%i3      ! y += stridey
3388      andn    %l3,%i5,%l2      ! hx &= ~0x80000000
3389      fmovs   %f24,%f16
3390      fmovs   %f27,%f19
3391      ba,pt   %icc,.loop2
3392 ! delay slot
3393      add     %i1,%i2,%i1      ! x += stride

3396      .align  16
3397 .range3:
3398      cmp     %l3,%o4
3399      bl,pt   %icc,1f          ! hx < 0x3e400000
3400 ! delay slot, harmless if branch taken
3401      sethi   %hi(0x7ff00000),%o7
3402      cmp     %l3,%o7
3403      bl,a,pt %icc,2f          ! branch if finite
3404 ! delay slot, squashed if branch not taken
3405      st      %o4,[%fp+biguns] ! set biguns
3406      fzero   %f24
3407      fmuld   %f26,%f24,%f26
3408      st      %f26,[%o3]
3409      ba,pt   %icc,2f
3410 ! delay slot
3411      st      %f27,[%o3+4]
3412 1:
3413      fdtoi   %f26,%f28        ! raise inexact if not zero
3414      st      %f24,[%o3]
3415      st      %f27,[%o3+4]
3416 2:
3417      addcc   %i0,-1,%i0
3418      ble,pn  %icc,.last3_from_range3
3419 ! delay slot, harmless if branch taken
3420      add     %i3,%i4,%i3      ! y += stridey

```

```
3421      ld      [%i1],%l3
3422      ld      [%i1],%f24
3423      ld      [%i1+4],%f27
3424      andn   %l3,%i5,%l3      ! hx &= ~0x80000000
3425      ba,pt  %icc,.loop3
3426 ! delay slot
3427      add    %i1,%i2,%i1      ! x += stridex

3429      SET_SIZE(__vsin_ultra3)
_____unchanged_portion_omitted_____
```

new/usr/src/lib/libmvec/common/vis/_vsqrtf_ultra3.S

1

```
*****
25030 Tue Nov 25 12:59:57 2014
new/usr/src/lib/libmvec/common/vis/_vsqrtf_ultra3.S
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21 /*
22 * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
23 */
24 /*
25 * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
26 * Use is subject to license terms.
27 */

29     .file     "_vsqrtf_ultra3.S"

31 #include "libm.h"
32 #if defined(LIBMVEC_SO_BUILD)
33     .weak     __vsqrtf
34     .type     __vsqrtf, #function
35     __vsqrtf = __vsqrtf_ultra3
36 #endif

36     RO_DATA
37     .align   64

39 .CONST_TBL:
40     .word   0x3fe00001, 0x80007e00 ! K1 = 5.00000715259318464227e-01
41     .word   0xbfc00003, 0xc0017a01 ! K2 = -1.25000447037521686593e-01
42     .word   0x000fffff, 0xffffffff ! DC0 = 0x000fffffffffffff
43     .word   0x3ff00000, 0x00000000 ! DC1 = 0x3ff0000000000000
44     .word   0x7ffff000, 0x00000000 ! DC2 = 0x7ffff00000000000

46 #define DC0          %f6
47 #define DC1          %f4
48 #define DC2          %f2
49 #define K2           %f38
50 #define K1           %f36
51 #define TBL          %l2
52 #define stridex      %l3
53 #define stridey      %l4
54 #define _0x1fff      %l5
```

new/usr/src/lib/libmvec/common/vis/_vsqrtf_ultra3.S

2

```
55 #define counter      %l6
56 #define _0x00800000  %l7
57 #define _0x7f800000  %o0

59 #define tmp_px       STACK_BIAS-0x40
60 #define tmp_counter  STACK_BIAS-0x38
61 #define tmp0         STACK_BIAS-0x30
62 #define tmp1         STACK_BIAS-0x28
63 #define tmp2         STACK_BIAS-0x20
64 #define tmp3         STACK_BIAS-0x18
65 #define tmp4         STACK_BIAS-0x10

67 ! sizeof temp storage - must be a multiple of 16 for V9
68 #define tmps         0x40

70 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
71 !      !!!!  algorithm  !!!!
72 !
73 !   x0 = *px;
74 !   ax = *(int*)px;
75 !   px += stridex;
76 !
77 ! if( ax >= 0x7f800000 )
78 ! {
79 !     *py = sqrtf(x0);
80 !     py += stridey;
81 !     continue;
82 ! }
83 ! if( ax < 0x00800000 )
84 ! {
85 !     *py = sqrtf(x0);
86 !     py += stridey;
87 !     continue;
88 ! }
89 !
90 ! db0 = (double)x0;
91 ! iexp0 = ax >> 24;
92 ! iexp0 += 0x3c0;
93 ! lexp0 = (long long)iexp0 << 52;
94 !
95 ! db0 = vis_fand(db0,DC0);
96 ! db0 = vis_for(db0,DC1);
97 ! hi0 = vis_fand(db0,DC2);
98 !
99 ! ax >>= 11;
100 ! si0 = ax & 0x1fff;
101 ! dtmp0 = ((double*)((char*)TBL + si0))[0];
102 ! xx0 = (db0 - hi0);
103 ! xx0 *= dtmp0;
104 ! dtmp0 = ((double*)((char*)TBL + si0))[1]
105 ! res0 = K2 * xx0;
106 ! res0 += K1;
107 ! res0 *= xx0;
108 ! res0 += DC1;
109 ! res0 = dtmp0 * res0;
110 ! dtmp1 = *((double*)&lexp0);
111 ! res0 *= dtmp1;
112 ! fres0 = (float)res0;
113 ! *py = fres0;
114 ! py += stridey;
115 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

117     ENTRY(__vsqrtf_ultra3)
118     save    %sp, -SA(MINFRAME)-tmps, %sp
119     PIC_SETUP(17)
120     PIC_SET(17, .CONST_TBL, o2)
```

```

121 PIC_SET(17, __vlibm_TBL_sqrtf, 12)
123 st    %i0, [%fp+tmp_counter]
124 sll  %i2, 2, stridex
125 or   %g0, 0xff8, %i5
127 stx  %i1, [%fp+tmp_px]
128 sll  %i5, 1, _0x1fff0
130 ldd  [%o2], K1
131 sll  %i4, 2, stridey
133 ldd  [%o2+8], K2
134 or   %g0, %i3, %g5
136 ldd  [%o2+16], DC0
137 sethi %hi(0x7f800000), %o0
139 ldd  [%o2+24], DC1
140 sethi %hi(0x00800000), %i17
142 ldd  [%o2+32], DC2
144 .begin:
145 ld   [%fp+tmp_counter], counter
146 ldx  [%fp+tmp_px], %i1
147 st   %g0, [%fp+tmp_counter]
148 .beginl:
149 cmp  counter, 0
150 ble, pn %icc, .exit
152 lda  [%i1]0x82, %o2      ! (2_0) ax = *(int*)px;
154 or   %g0, %i1, %o7
155 lda  [%i1]0x82, %f25    ! (2_0) x0 = *px;
157 cmp  %o2, _0x7f800000  ! (2_0) ax ? 0x7f800000
158 bge, pn %icc, .spec    ! (2_0) if( ax >= 0x7f800000 )
159 nop
161 cmp  %o2, _0x00800000  ! (2_0) ax ? 0x00800000
162 bl, pn %icc, .spec     ! (2_0) if( ax < 0x00800000 )
163 nop
165 fstod %f25, %f56      ! (2_0) db0 = (double)x0;
167 lda  [stridex+%o7]0x82, %o1 ! (3_0) ax = *(int*)px;
169 sra  %o2, 24, %i1      ! (2_0) iexp0 = ax >> 24;
171 add  %o7, stridex, %i1  ! px += stridex
172 add  %i1, 960, %i10     ! (2_0) iexp0 += 0x3c0;
173 lda  [stridex+%o7]0x82, %f0 ! (3_0) x0 = *px;
174 fand %f56, DC0, %f60    ! (2_0) db0 = vis_fand(db0, DC0);
176 cmp  %o1, _0x7f800000  ! (3_0) ax ? 0x7f800000
177 bge, pn %icc, .update0 ! (3_0) if( ax >= 0x7f800000 )
178 nop
179 .cont0:
180 sllx %i0, 52, %o3      ! (2_0) lexp0 = (long long)iexp0 << 52;
182 sra  %o2, 11, %i2      ! (2_0) ax >>= 11;
183 stx  %o3, [%fp+tmp0]   ! (2_0) dtmpl = *((double*)&lexp0);
184 for  %f60, DC1, %f40   ! (2_0) db0 = vis_for(db0, DC1);
186 cmp  %o1, _0x00800000  ! (3_0) ax ? 0x00800000

```

```

187 bl, pn %icc, .update1  ! (3_0) if( ax < 0x00800000 )
188 nop
189 .cont1:
190 fstod %f0, %f48        ! (3_0) db0 = (double)x0;
192 and  %i2, _0x1fff0, %o3 ! (2_0) si0 = ax & 0x1fff0;
193 lda  [%i1+stridex]0x82, %o2 ! (4_0) ax = *(int*)px;
195 add  %i1, stridex, %i1  ! px += stridex
196 add  %o3, TBL, %i2      ! (2_0) (char*)TBL + si0
197 fand %f40, DC2, %f46   ! (2_0) hi0 = vis_fand(db0, DC2);
199 sra  %o1, 24, %o4      ! (3_0) iexp0 = ax >> 24;
201 lda  [%i1]0x82, %f13   ! (4_0) x0 = *px;
202 fand %f48, DC0, %f58   ! (3_0) db0 = vis_fand(db0, DC0);
204 add  %o4, 960, %i0     ! (3_0) iexp0 += 0x3c0;
206 cmp  %o2, _0x7f800000  ! (4_1) ax ? 0x7f800000
207 bge, pn %icc, .update2 ! (4_1) if( ax >= 0x7f800000 )
208 nop
209 .cont2:
210 fsubd %f40, %f46, %f44 ! (2_1) xx0 = (db0 - hi0);
211 sllx %i0, 52, %g1      ! (3_1) lexp0 = (long long)iexp0 << 52;
212 ldd  [%i2], %f40       ! (2_1) dtmp0 = ((double*)((char*)TBL +
214 sra  %o1, 11, %i10     ! (3_1) ax >>= 11;
215 stx  %g1, [%fp+tmp1]   ! (3_1) dtmpl = *((double*)&lexp0);
216 for  %f58, DC1, %f48   ! (3_1) db0 = vis_for(db0, DC1);
218 cmp  %o2, _0x00800000  ! (4_1) ax ? 0x00800000
219 bl, pn %icc, .update3  ! (4_1) if( ax < 0x00800000 )
220 nop
221 .cont3:
222 fstod %f13, %f50      ! (4_1) db0 = (double)x0;
224 fmuld %f44, %f40, %f46 ! (2_1) xx0 * = dtmp0;
225 and  %i0, _0x1fff0, %i0 ! (3_1) si0 = ax & 0x1fff0;
226 lda  [%i1+stridex]0x82, %i1 ! (0_0) ax = *(int*)px;
228 add  %i0, TBL, %i10    ! (3_1) (char*)TBL + si0
229 fand %f48, DC2, %f62   ! (3_1) hi0 = vis_fand(db0, DC2);
231 sra  %o2, 24, %o7      ! (4_1) iexp0 = ax >> 24;
233 add  %i1, stridex, %o4  ! px += stridex
234 add  %o7, 960, %o7      ! (4_1) iexp0 += 0x3c0;
235 lda  [%i1+stridex]0x82, %f17 ! (0_0) x0 = *px;
236 fand %f50, DC0, %f54   ! (4_1) db0 = vis_fand(db0, DC0);
238 fmuld K2, %f46, %f52   ! (2_1) res0 = K2 * xx0;
239 cmp  %i1, _0x7f800000  ! (0_0) ax ? 0x7f800000
240 bge, pn %icc, .update4 ! (0_0) if( ax >= 0x7f800000 )
241 fsubd %f48, %f62, %f42 ! (3_1) xx0 = (db0 - hi0);
242 .cont4:
243 sllx %o7, 52, %o1      ! (4_1) lexp0 = (long long)iexp0 << 52;
244 ldd  [%i0+TBL], %f40   ! (3_1) dtmp0 = ((double*)((char*)TBL +
246 sra  %o2, 11, %i5      ! (4_1) ax >>= 11;
247 stx  %o1, [%fp+tmp2]   ! (4_1) dtmpl = *((double*)&lexp0);
248 for  %f54, DC1, %f34   ! (4_1) db0 = vis_for(db0, DC1);
250 cmp  %i1, _0x00800000  ! (0_0) ax ? 0x00800000
251 bl, pn %icc, .update5  ! (0_0) if( ax < 0x00800000 )
252 nop

```

```

253 .cont5:
254     fstod    %f17,%f56                ! (0_0) db0 = (double)x0;

256     fmuld   %f42,%f40,%f42          ! (3_1) xx0 *= dtmp0;
257     lda     [stridex+%04]0x82,%i0    ! (1_0) ax = *(int*)px;
258     fadd    %f52,K1,%f52            ! (2_1) res0 += K1;

260     sra     %l1,24,%g1               ! (0_0) iexp0 = ax >> 24;
261     and     %i5,_0x1fff0,%i5        ! (4_1) si0 = ax & 0x1fff0;
262     fand    %f34,DC2,%f62          ! (4_1) hi0 = vis_fand(db0,DC2);

264     add     %04,stridex,%i1         ! px += stridex

266     add     %g1,960,%o5             ! (0_0) iexp0 += 0x3c0;
267     add     %i5,TBL,%i3             ! (4_1) (char*)TBL + si0
268     lda     [stridex+%04]0x82,%f21  ! (1_0) x0 = *px;
269     fand    %f56,DC0,%f32          ! (0_0) db0 = vis_fand(db0,DC0);

271     fmuld   K2,%f42,%f50            ! (3_1) res0 = K2 * xx0;
272     cmp     %i0,_0x7f800000         ! (1_0) ax ? 0x7f800000
273     bge,pn %icc,.update6           ! (1_0) if( ax >= 0x7f800000 )
274     fsubd   %f34,%f62,%f54         ! (4_1) xx0 = (db0 - hi0);
275 .cont6:
276     fmuld   %f52,%f46,%f52          ! (2_1) res0 *= xx0;
277     sllx    %o5,52,%o7             ! (0_0) lexp0 = (long long)iexp0 << 52;
278     ldd     [TBL+%i5],%f62         ! (4_1) dtmp0 = ((double*)((char*)TBL +

280     sra     %l1,11,%i4             ! (0_0) ax >>= 11;
281     stx     %o7,[%fp+tmp3]          ! (0_0) dtmpl = *((double*)&lexp0);
282     for     %f32,DC1,%f48          ! (0_0) db0 = vis_for(db0,DC1);

284     cmp     %i0,_0x00800000         ! (1_0) ax ? 0x00800000
285     bl,pn  %icc,.update7           ! (1_0) if( ax < 0x00800000 )
286     nop
287 .cont7:
288     fstod    %f21,%f56                ! (1_0) db0 = (double)x0;

290     fmuld   %f54,%f62,%f46          ! (4_1) xx0 *= dtmp0;
291     and     %i4,_0x1fff0,%g1        ! (0_0) si0 = ax & 0x1fff0;
292     lda     [%i1+stridex]0x82,%o2    ! (2_0) ax = *(int*)px;
293     fadd    %f50,K1,%f62            ! (3_1) res0 += K1;

295     add     %g1,TBL,%i5             ! (0_0) (double*)((char*)TBL + si0
296     fand    %f48,DC2,%f32          ! (0_0) hi0 = vis_fand(db0,DC2);

298     sra     %i0,24,%o4             ! (1_0) iexp0 = ax >> 24;
299     ldd     [%i2+8],%f60            ! (2_1) dtmp0 = ((double*)((char*)TBL +
300     fadd    %f52,DC1,%f58          ! (2_1) res0 += DC1;

302     add     %i1,stridex,%o7         ! px += stridex
303     add     %o4,960,%i2             ! (1_0) iexp0 += 0x3c0;
304     lda     [%i1+stridex]0x82,%f25  ! (2_0) x0 = *px;
305     fand    %f56,DC0,%f34          ! (1_0) db0 = vis_fand(db0,DC0);

307     fmuld   K2,%f46,%f50            ! (4_1) res0 = K2 * xx0;
308     cmp     %o2,_0x7f800000         ! (2_0) ax ? 0x7f800000
309     bge,pn %icc,.update8           ! (2_0) if( ax >= 0x7f800000 )
310     fsubd   %f48,%f32,%f52         ! (0_0) xx0 = (db0 - hi0);
311 .cont8:
312     fmuld   %f62,%f42,%f54          ! (3_1) res0 *= xx0;
313     sllx    %i2,52,%o4             ! (1_0) lexp0 = (long long)iexp0 << 52;
314     ldd     [TBL+%g1],%f32         ! (0_0) dtmp0 = ((double*)((char*)TBL +

316     fmuld   %f60,%f58,%f60          ! (2_1) res0 = dtmp0 * res0;
317     sra     %i0,11,%g1             ! (1_0) ax >>= 11;
318     stx     %o4,[%fp+tmp4]          ! (1_0) dtmpl = *((double*)&lexp0);

```

```

319     for     %f34,DC1,%f48          ! (1_0) db0 = vis_for(db0,DC1);

321     cmp     %o2,_0x00800000         ! (2_0) ax ? 0x00800000
322     bl,pn  %icc,.update9           ! (2_0) if( ax < 0x00800000 )
323     ldd     [%fp+tmp0],%f40         ! (2_1) dtmpl = *((double*)&lexp0);
324     fstod   %f25,%f56                ! (2_0) db0 = (double)x0;
325 .cont9:
326     fmuld   %f52,%f32,%f42          ! (0_0) xx0 *= dtmp0;
327     and     %g1,_0x1fff0,%o5        ! (1_0) si0 = ax & 0x1fff0;
328     lda     [stridex+%o7]0x82,%o1    ! (3_0) ax = *(int*)px;
329     fadd    %f50,K1,%f34            ! (4_1) res0 += K1;

331     add     %o5,TBL,%i4             ! (1_0) (char*)TBL + si0
332     fand    %f48,DC2,%f62          ! (1_0) hi0 = vis_fand(db0,DC2);

334     fmuld   %f60,%f40,%f32          ! (2_1) res0 *= dtmpl;
335     sra     %o2,24,%l1             ! (2_0) iexp0 = ax >> 24;
336     ldd     [%i0+8],%f40            ! (3_1) dtmp0 = ((double*)((char*)TBL +
337     fadd    %f54,DC1,%f58          ! (3_1) res0 += DC1;

339     add     %o7,stridex,%i1         ! px += stridex
340     add     %l1,960,%l0             ! (2_0) iexp0 += 0x3c0;
341     lda     [stridex+%o7]0x82,%f0    ! (3_0) x0 = *px;
342     fand    %f56,DC0,%f60          ! (2_0) db0 = vis_fand(db0,DC0);

344     fmuld   K2,%f42,%f50            ! (0_0) res0 = K2 * xx0;
345     cmp     %o1,_0x7f800000         ! (3_0) ax ? 0x7f800000
346     bge,pn %icc,.update10         ! (3_0) if( ax >= 0x7f800000 )
347     fsubd   %f48,%f62,%f54         ! (1_0) xx0 = (db0 - hi0);
348 .cont10:
349     fmuld   %f34,%f46,%f52          ! (4_1) res0 *= xx0;
350     sllx    %l0,52,%o3             ! (2_0) lexp0 = (long long)iexp0 << 52;
351     ldd     [TBL+%o5],%f56         ! (1_0) dtmp0 = ((double*)((char*)TBL +

353     fmuld   %f40,%f58,%f34          ! (3_1) res0 = dtmp0 * res0;
354     sra     %o2,11,%i2             ! (2_0) ax >>= 11;
355     stx     %o3,[%fp+tmp0]          ! (2_0) dtmpl = *((double*)&lexp0);
356     for     %f30,DC1,%f40          ! (2_0) db0 = vis_for(db0,DC1);

358     cmp     %o1,_0x00800000         ! (3_0) ax ? 0x00800000
359     bl,pn  %icc,.update11         ! (3_0) if( ax < 0x00800000 )
360     ldd     [%fp+tmp1],%f62         ! (3_1) dtmpl = *((double*)&lexp0);
361     fstod   %f0,%f48                ! (3_0) db0 = (double)x0;
362 .cont11:
363     fmuld   %f54,%f56,%f30          ! (1_0) xx0 *= dtmp0;
364     and     %i2,_0x1fff0,%o3        ! (2_0) si0 = ax & 0x1fff0;
365     lda     [%i1+stridex]0x82,%o2    ! (4_0) ax = *(int*)px;
366     fadd    %f50,K1,%f56            ! (0_0) res0 += K1;

368     add     %i1,stridex,%i1         ! px += stridex
369     add     %o3,TBL,%i2             ! (2_0) (char*)TBL + si0
370     fand    %f40,DC2,%f46          ! (2_0) hi0 = vis_fand(db0,DC2);

372     fmuld   %f34,%f62,%f28          ! (3_1) res0 *= dtmpl;
373     sra     %o1,24,%o4             ! (3_0) iexp0 = ax >> 24;
374     ldd     [%i3+8],%f50            ! (4_1) dtmp0 = ((double*)((char*)TBL +
375     fadd    %f52,DC1,%f54          ! (4_1) res0 += DC1;

377     lda     [%i1]0x82,%f13          ! (4_0) x0 = *px;
378     fand    %f48,DC0,%f58          ! (3_0) db0 = vis_fand(db0,DC0);

380     or      %g0,%g5,%i3
381     cmp     counter,5
382     bl,pn  %icc,.tail
383     add     %o4,960,%g5             ! (3_0) iexp0 += 0x3c0;

```

```

385     ba     .main_loop
386     sub     counter,5,counter      ! counter

388     .align 16
389 .main_loop:
390     fmuld  K2,%f30,%f60          ! (1_1) res0 = K2 * xx0;
391     cmp    %o2,_0x7f800000      ! (4_1) ax ? 0x7f800000
392     bge,pn %icc,.update12      ! (4_1) if( ax >= 0x7f800000 )
393     fsubd  %f40,%f46,%f44      ! (2_1) xx0 = (db0 - hi0);
394 .cont12:
395     fmuld  %f56,%f42,%f52      ! (0_1) res0 *= xx0;
396     sllx   %g5,52,%g5          ! (3_1) lexp0 = (long long)iexp0 << 52;
397     ldd    [%i2],%f40          ! (2_1) dtmp0 = ((double*)((char*)TBL +
398     fdtos  %f32,%f15          ! (2_2) fres0 = (float)res0;

400     fmuld  %f50,%f54,%f42      ! (4_2) res0 = dtmp0 * res0;
401     sra    %o1,11,%i0          ! (3_1) ax >>= 11;
402     stx    %g5,[%fp+tmp1]      ! (3_1) dtmp1 = *((double*)&lexp0);
403     for    %f58,DC1,%f48      ! (3_1) db0 = vis_for(db0,DC1);

405     cmp    %o2,_0x00800000      ! (4_1) ax ? 0x00800000
406     bl,pn  %icc,.update13      ! (4_1) if( ax < 0x00800000 )
407     ldd    [%fp+tmp2],%f56     ! (4_2) dtmp1 = *((double*)&lexp0);
408     fstod  %f13,%f50          ! (4_1) db0 = (double)x0;
409 .cont13:
410     fmuld  %f44,%f40,%f46      ! (2_1) xx0 *= dtmp0;
411     and    %i0,_0x1fff,%i0     ! (3_1) si0 = ax & 0x1fff;
412     lda    [%i1+strindex]0x82,%i1 ! (0_0) ax = *(int*)px;
413     faddd  %f60,K1,%f32        ! (1_1) res0 += K1;

415     add    %i0,TBL,%i0         ! (3_1) (char*)TBL + si0
416     add    %i3,stridey,%o3      ! py += stridey
417     st     %f15,[%i3]         ! (2_2) *py = fres0;
418     fand   %f48,DC2,%f62      ! (3_1) hi0 = vis_fand(db0,DC2);

420     fmuld  %f42,%f56,%f44      ! (4_2) res0 *= dtmp1;
421     sra    %o2,24,%o7          ! (4_1) iexp0 = ax >> 24;
422     ldd    [%i5+8],%f58       ! (0_1) dtmp0 = ((double*)((char*)TBL +
423     faddd  %f52,DC1,%f34      ! (0_1) res0 += DC1;

425     add    %i1,stridex,%o4     ! px += stridex
426     add    %o7,960,%o7         ! (4_1) iexp0 += 0x3c0;
427     lda    [%i1+stridex]0x82,%f17 ! (0_0) x0 = *px;
428     fand   %f50,DC0,%f54      ! (4_1) db0 = vis_fand(db0,DC0);

430     fmuld  K2,%f46,%f52        ! (2_1) res0 = K2 * xx0;
431     cmp    %i1,_0x7f800000      ! (0_0) ax ? 0x7f800000
432     bge,pn %icc,.update14      ! (0_0) if( ax >= 0x7f800000 )
433     fsubd  %f48,%f62,%f42      ! (3_1) xx0 = (db0 - hi0);
434 .cont14:
435     fmuld  %f32,%f30,%f48      ! (1_1) res0 *= xx0;
436     sllx   %o2,52,%o1          ! (4_1) lexp0 = (long long)iexp0 << 52;
437     ldd    [%i0+TBL],%f40      ! (3_1) dtmp0 = ((double*)((char*)TBL +
438     fdtos  %f28,%f19          ! (3_2) fres0 = (float)res0;

440     fmuld  %f58,%f34,%f32      ! (0_1) res0 = dtmp0 * res0;
441     sra    %o2,11,%i5          ! (4_1) ax >>= 11;
442     stx    %o1,[%fp+tmp2]      ! (4_1) dtmp1 = *((double*)&lexp0);
443     for    %f54,DC1,%f34      ! (4_1) db0 = vis_for(db0,DC1);

445     cmp    %i1,_0x00800000      ! (0_0) ax ? 0x00800000
446     bl,pn  %icc,.update15      ! (0_0) if( ax < 0x00800000 )
447     ldd    [%fp+tmp3],%f60     ! (0_1) dtmp1 = *((double*)&lexp0);
448     fstod  %f17,%f56          ! (0_0) db0 = (double)x0;
449 .cont15:
450     fmuld  %f42,%f40,%f42      ! (3_1) xx0 *= dtmp0;

```

```

451     add    %o3,stridey,%g5      ! py += stridey
452     lda    [stridex+%o4]0x82,%i0 ! (1_0) ax = *(int*)px;
453     faddd  %f52,K1,%f52        ! (2_1) res0 += K1;

455     sra    %i1,24,%g1          ! (0_0) iexp0 = ax >> 24;
456     and    %i5,_0x1fff,%i5     ! (4_1) si0 = ax & 0x1fff;
457     st     %f19,[%o3]         ! (3_2) *py = fres0;
458     fand   %f34,DC2,%f62      ! (4_1) hi0 = vis_fand(db0,DC2);

460     fmuld  %f32,%f60,%f40      ! (0_1) res0 *= dtmp1;
461     add    %o4,stridex,%i1     ! px += stridex
462     ldd    [%i4+8],%f60       ! (1_1) dtmp0 = ((double*)((char*)TBL +
463     faddd  %f48,DC1,%f58      ! (1_1) res0 += DC1;

465     add    %g1,960,%o5        ! (0_0) iexp0 += 0x3c0;
466     add    %i5,TBL,%i3        ! (4_1) (char*)TBL + si0
467     lda    [stridex+%o4]0x82,%f21 ! (1_0) x0 = *px;
468     fand   %f56,DC0,%f32      ! (0_0) db0 = vis_fand(db0,DC0);

470     fmuld  K2,%f42,%f50        ! (3_1) res0 = K2 * xx0;
471     cmp    %o2,_0x7f800000      ! (1_0) ax ? 0x7f800000
472     bge,pn %icc,.update16      ! (1_0) if( ax >= 0x7f800000 )
473     fsubd  %f34,%f62,%f54      ! (4_1) xx0 = (db0 - hi0);
474 .cont16:
475     fmuld  %f52,%f46,%f52      ! (2_1) res0 *= xx0;
476     sllx   %o5,52,%o7          ! (0_0) lexp0 = (long long)iexp0 << 52;
477     ldd    [TBL+%i5],%f62     ! (4_1) dtmp0 = ((double*)((char*)TBL +
478     fdtos  %f44,%f23          ! (4_2) fres0 = (float)res0;

480     fmuld  %f60,%f58,%f44      ! (1_1) res0 = dtmp0 * res0;
481     sra    %i0,11,%i4         ! (0_0) ax >>= 11;
482     stx    %o7,[%fp+tmp3]      ! (0_0) dtmp1 = *((double*)&lexp0);
483     for    %f32,DC1,%f48      ! (0_0) db0 = vis_for(db0,DC1);

485     cmp    %i0,_0x00800000      ! (1_0) ax ? 0x00800000
486     bl,pn  %icc,.update17      ! (1_0) if( ax < 0x00800000 )
487     ldd    [%fp+tmp4],%f34     ! (1_1) dtmp1 = *((double*)&lexp0);
488     fstod  %f21,%f56          ! (1_0) db0 = (double)x0;
489 .cont17:
490     fmuld  %f54,%f62,%f46      ! (4_1) xx0 *= dtmp0;
491     and    %i4,_0x1fff,%g1     ! (0_0) si0 = ax & 0x1fff;
492     lda    [%i1+stridex]0x82,%o2 ! (2_0) ax = *(int*)px;
493     faddd  %f50,K1,%f62        ! (3_1) res0 += K1;

495     add    %g1,TBL,%i5        ! (0_0) (double*)((char*)TBL + si0
496     add    %g5,stridey,%g5      ! py += stridey
497     st     %f23,[stridex+%o3]  ! (4_2) *py = fres0;
498     fand   %f48,DC2,%f32      ! (0_0) hi0 = vis_fand(db0,DC2);

500     fmuld  %f44,%f34,%f44      ! (1_1) res0 *= dtmp1;
501     sra    %i0,24,%o4         ! (1_0) iexp0 = ax >> 24;
502     ldd    [%i2+8],%f60       ! (2_1) dtmp0 = ((double*)((char*)TBL +
503     faddd  %f52,DC1,%f58      ! (2_1) res0 += DC1;

505     add    %i1,stridex,%o7     ! px += stridex
506     add    %o4,960,%i2        ! (1_0) iexp0 += 0x3c0;
507     lda    [%i1+stridex]0x82,%f25 ! (2_0) x0 = *px;
508     fand   %f56,DC0,%f34      ! (1_0) db0 = vis_fand(db0,DC0);

510     fmuld  K2,%f46,%f50        ! (4_1) res0 = K2 * xx0;
511     cmp    %o2,_0x7f800000      ! (2_0) ax ? 0x7f800000
512     bge,pn %icc,.update18      ! (2_0) if( ax >= 0x7f800000 )
513     fsubd  %f48,%f32,%f52      ! (0_0) xx0 = (db0 - hi0);
514 .cont18:
515     fmuld  %f62,%f42,%f54      ! (3_1) res0 *= xx0;
516     sllx   %i2,52,%o4         ! (1_0) lexp0 = (long long)iexp0 << 52;

```

```

517      ldd      [TBL+%g1],%f32      ! (0_0) dtmp0 = ((double*)((char*)TBL +
518      fdtos    %f40,%f27          ! (0_1) fres0 = (float)res0;

520      fmuld    %f60,%f58,%f60      ! (2_1) res0 = dtmp0 * res0;
521      sra     %i0,11,%g1           ! (1_0) ax >>= 11;
522      stx     %o4,[%fp+tmp4]       ! (1_0) dtmp1 = *((double*)&lexp0);
523      for     %f34,DC1,%f48       ! (1_0) db0 = vis_for(db0,DC1);

525      cmp     %o2,_0x00800000      ! (2_0) ax ? 0x00800000
526      bl,pn   %icc,.update19      ! (2_0) if( ax < 0x00800000 )
527      ldd     [%fp+tmp0],%f40      ! (2_1) dtmp1 = *((double*)&lexp0);
528      fstod   %f25,%f56          ! (2_0) db0 = (double)x0;
529 .cont19:
530      fmuld    %f52,%f32,%f42      ! (0_0) xx0 *= dtmp0;
531      and     %g1,_0x1fff0,%o5     ! (1_0) si0 = ax & 0x1fff0;
532      lda     [%stridx+%o7]0x82,%o1 ! (3_0) ax = *(int*)px;
533      faddd    %f50,K1,%f34        ! (4_1) res0 += K1;

535      add     %o5,TBL,%i4          ! (1_0) (char*)TBL + si0
536      add     %g5,stridey,%g1      ! py += stridey
537      st      %f27,[%g5]          ! (0_1) *py = fres0;
538      fand    %f48,DC2,%f62       ! (1_0) hi0 = vis_fand(db0,DC2);

540      fmuld    %f60,%f40,%f32      ! (2_1) res0 *= dtmp1;
541      sra     %o2,24,%i1          ! (2_0) iexp0 = ax >> 24;
542      ldd     [%i0+8],%f40        ! (3_1) dtmp0 = ((double*)((char*)TBL +
543      faddd    %f54,DC1,%f58       ! (3_1) res0 += DC1;

545      add     %o7,stridx,%i1      ! px += stridx
546      add     %i1,960,%i0         ! (2_0) iexp0 += 0x3c0;
547      lda     [%stridx+%o7]0x82,%f0 ! (3_0) x0 = *px;
548      fand    %f56,DC0,%f60       ! (2_0) db0 = vis_fand(db0,DC0);

550      fmuld    K2,%f42,%f50        ! (0_0) res0 = K2 * xx0;
551      cmp     %o1,_0x7f800000      ! (3_0) ax ? 0x7f800000
552      bge,pn  %icc,.update20      ! (3_0) if( ax >= 0x7f800000 )
553      fsubd   %f48,%f62,%f54      ! (1_0) xx0 = (db0 - hi0);
554 .cont20:
555      fmuld    %f34,%f46,%f52      ! (4_1) res0 *= xx0;
556      sllx    %i0,52,%o3          ! (2_0) lexp0 = (long long)iexp0 << 52;
557      ldd     [TBL+%o5],%f56      ! (1_0) dtmp0 = ((double*)((char*)TBL +
558      fdtos    %f44,%f8           ! (1_1) fres0 = (float)res0;

560      fmuld    %f40,%f58,%f34      ! (3_1) res0 = dtmp0 * res0;
561      sra     %o2,11,%i2          ! (2_0) ax >>= 11;
562      stx     %o3,[%fp+tmp0]       ! (2_0) dtmp1 = *((double*)&lexp0);
563      for     %f60,DC1,%f40       ! (2_0) db0 = vis_for(db0,DC1);

565      cmp     %o1,_0x00800000      ! (3_0) ax ? 0x00800000
566      bl,pn   %icc,.update21      ! (3_0) if( ax < 0x00800000 )
567      ldd     [%fp+tmp1],%f62      ! (3_1) dtmp1 = *((double*)&lexp0);
568      fstod   %f0,%f48           ! (3_0) db0 = (double)x0;
569 .cont21:
570      fmuld    %f54,%f56,%f30      ! (1_0) xx0 *= dtmp0;
571      and     %i2,_0x1fff0,%o3     ! (2_0) si0 = ax & 0x1fff0;
572      lda     [%i1+stridx]0x82,%o2 ! (4_0) ax = *(int*)px;
573      faddd    %f50,K1,%f56        ! (0_0) res0 += K1;

575      add     %i1,stridx,%i1      ! px += stridx
576      add     %o3,TBL,%i2         ! (2_0) (char*)TBL + si0
577      st      %f8,[%stridx+%g5]   ! (1_1) *py = fres0;
578      fand    %f40,DC2,%f46       ! (2_0) hi0 = vis_fand(db0,DC2);

580      fmuld    %f34,%f62,%f28      ! (3_1) res0 *= dtmp1;
581      sra     %o1,24,%o4          ! (3_0) iexp0 = ax >> 24;
582      ldd     [%i3+8],%f50        ! (4_1) dtmp0 = ((double*)((char*)TBL +

```

```

583      faddd    %f52,DC1,%f54      ! (4_1) res0 += DC1;

585      add     %g1,stridey,%i3      ! py += stridey
586      subcc   counter,5,counter    ! counter
587      lda     [%i1]0x82,%f13      ! (4_0) x0 = *px;
588      fand    %f48,DC0,%f58       ! (3_0) db0 = vis_fand(db0,DC0);

590      bpos,pt %icc,.main_loop
591      add     %o4,960,%g5         ! (3_0) iexp0 += 0x3c0;

593      add     counter,5,counter
594 .tail:
595      subcc   counter,1,counter
596      bneg,a  .begin
597      or     %g0,%i3,%g5

599      fmuld    %f56,%f42,%f52      ! (0_1) res0 *= xx0;
600      fdtos    %f32,%f15         ! (2_2) fres0 = (float)res0;

602      fmuld    %f50,%f54,%f42      ! (4_2) res0 = dtmp0 * res0;

604      ldd     [%fp+tmp2],%f56     ! (4_2) dtmp1 = *((double*)&lexp0);

606      add     %i3,stridey,%o3     ! py += stridey
607      st      %f15,[%i3]         ! (2_2) *py = fres0;

609      subcc   counter,1,counter
610      bneg,a  .begin
611      or     %g0,%o3,%g5

613      fmuld    %f42,%f56,%f44      ! (4_2) res0 *= dtmp1;
614      ldd     [%i5+8],%f58        ! (0_1) dtmp0 = ((double*)((char*)TBL +
615      faddd    %f52,DC1,%f34       ! (0_1) res0 += DC1;

617      fdtos    %f28,%f19         ! (3_2) fres0 = (float)res0;

619      fmuld    %f58,%f34,%f32      ! (0_1) res0 = dtmp0 * res0;

621      ldd     [%fp+tmp3],%f60     ! (0_1) dtmp1 = *((double*)&lexp0);

623      add     %o3,stridey,%g5     ! py += stridey

625      st      %f19,[%o3]         ! (3_2) *py = fres0;

627      subcc   counter,1,counter
628      bneg,a  .begin
629      nop

631      fmuld    %f32,%f60,%f40      ! (0_1) res0 *= dtmp1;

633      fdtos    %f44,%f23         ! (4_2) fres0 = (float)res0;

635      add     %g5,stridey,%g5     ! py += stridey
636      st      %f23,[%stridx+%o3] ! (4_2) *py = fres0;

638      subcc   counter,1,counter
639      bneg,a  .begin
640      nop

642      fdtos    %f40,%f27         ! (0_1) fres0 = (float)res0;

644      st      %f27,[%g5]         ! (0_1) *py = fres0;

646      ba     .begin
647      add     %g5,stridey,%g5

```

```

649     .align 16
650 .spec:
651     fsqrts    %f25,%f25
652     sub      counter,1,counter
653     add      %i1, stridex,%i1
654     st       %f25,[%g5]
655     ba      .begin1
656     add      %g5, stridey,%g5

658     .align 16
659 .update0:
660     cmp      counter,1
661     ble     .cont0
662     fzeros  %f0

664     stx     %i1,[%fp+tmp_px]
665     sethi   %hi(0x7f800000),%o1

667     sub     counter,1,counter
668     st      counter,[%fp+tmp_counter]

670     ba     .cont0
671     or     %g0,1,counter

673     .align 16
674 .update1:
675     cmp     counter,1
676     ble     .cont1
677     fzeros  %f0

679     stx     %i1,[%fp+tmp_px]
680     clr     %o1

682     sub     counter,1,counter
683     st      counter,[%fp+tmp_counter]

685     ba     .cont1
686     or     %g0,1,counter

688     .align 16
689 .update2:
690     cmp     counter,2
691     ble     .cont2
692     fzeros  %f13

694     stx     %i1,[%fp+tmp_px]
695     sethi   %hi(0x7f800000),%o2

697     sub     counter,2,counter
698     st      counter,[%fp+tmp_counter]

700     ba     .cont2
701     or     %g0,2,counter

703     .align 16
704 .update3:
705     cmp     counter,2
706     ble     .cont3
707     fzeros  %f13

709     stx     %i1,[%fp+tmp_px]
710     clr     %o2

712     sub     counter,2,counter
713     st      counter,[%fp+tmp_counter]

```

```

715     ba     .cont3
716     or     %g0,2,counter

718     .align 16
719 .update4:
720     cmp     counter,3
721     ble     .cont4
722     fzeros  %f17

724     stx     %o4,[%fp+tmp_px]
725     sethi   %hi(0x7f800000),%l1

727     sub     counter,3,counter
728     st      counter,[%fp+tmp_counter]

730     ba     .cont4
731     or     %g0,3,counter

733     .align 16
734 .update5:
735     cmp     counter,3
736     ble     .cont5
737     fzeros  %f17

739     stx     %o4,[%fp+tmp_px]
740     clr     %l1

742     sub     counter,3,counter
743     st      counter,[%fp+tmp_counter]

745     ba     .cont5
746     or     %g0,3,counter

748     .align 16
749 .update6:
750     cmp     counter,4
751     ble     .cont6
752     fzeros  %f21

754     stx     %i1,[%fp+tmp_px]
755     sethi   %hi(0x7f800000),%i0

757     sub     counter,4,counter
758     st      counter,[%fp+tmp_counter]

760     ba     .cont6
761     or     %g0,4,counter

763     .align 16
764 .update7:
765     cmp     counter,4
766     ble     .cont7
767     fzeros  %f21

769     stx     %i1,[%fp+tmp_px]
770     clr     %i0

772     sub     counter,4,counter
773     st      counter,[%fp+tmp_counter]

775     ba     .cont7
776     or     %g0,4,counter

778     .align 16
779 .update8:
780     cmp     counter,5

```



```

781     ble     .cont8
782     fzeros  %f25

784     stx     %o7, [%fp+tmp_px]
785     sethi   %hi(0x7f800000), %o2

787     sub     counter, 5, counter
788     st      counter, [%fp+tmp_counter]

790     ba     .cont8
791     or      %g0, 5, counter

793     .align  16
794 .update9:
795     cmp     counter, 5
796     ble     .cont9
797     fzeros  %f25

799     stx     %o7, [%fp+tmp_px]
800     clr     %o2

802     sub     counter, 5, counter
803     st      counter, [%fp+tmp_counter]

805     ba     .cont9
806     or      %g0, 5, counter

808     .align  16
809 .update10:
810     cmp     counter, 6
811     ble     .cont10
812     fzeros  %f0

814     stx     %i1, [%fp+tmp_px]
815     sethi   %hi(0x7f800000), %o1

817     sub     counter, 6, counter
818     st      counter, [%fp+tmp_counter]

820     ba     .cont10
821     or      %g0, 6, counter

823     .align  16
824 .update11:
825     cmp     counter, 6
826     ble     .cont11
827     fzeros  %f0

829     stx     %i1, [%fp+tmp_px]
830     clr     %o1

832     sub     counter, 6, counter
833     st      counter, [%fp+tmp_counter]

835     ba     .cont11
836     or      %g0, 6, counter

838     .align  16
839 .update12:
840     cmp     counter, 2
841     ble     .cont12
842     fzeros  %f13

844     stx     %i1, [%fp+tmp_px]
845     sethi   %hi(0x7f800000), %o2

```

```

847     sub     counter, 2, counter
848     st      counter, [%fp+tmp_counter]

850     ba     .cont12
851     or      %g0, 2, counter

853     .align  16
854 .update13:
855     cmp     counter, 2
856     ble     .cont13
857     fzeros  %f13

859     stx     %i1, [%fp+tmp_px]
860     clr     %o2

862     sub     counter, 2, counter
863     st      counter, [%fp+tmp_counter]

865     ba     .cont13
866     or      %g0, 2, counter

868     .align  16
869 .update14:
870     cmp     counter, 3
871     ble     .cont14
872     fzeros  %f17

874     stx     %o4, [%fp+tmp_px]
875     sethi   %hi(0x7f800000), %i1

877     sub     counter, 3, counter
878     st      counter, [%fp+tmp_counter]

880     ba     .cont14
881     or      %g0, 3, counter

883     .align  16
884 .update15:
885     cmp     counter, 3
886     ble     .cont15
887     fzeros  %f17

889     stx     %o4, [%fp+tmp_px]
890     clr     %i1

892     sub     counter, 3, counter
893     st      counter, [%fp+tmp_counter]

895     ba     .cont15
896     or      %g0, 3, counter

898     .align  16
899 .update16:
900     cmp     counter, 4
901     ble     .cont16
902     fzeros  %f21

904     stx     %i1, [%fp+tmp_px]
905     sethi   %hi(0x7f800000), %i0

907     sub     counter, 4, counter
908     st      counter, [%fp+tmp_counter]

910     ba     .cont16
911     or      %g0, 4, counter

```

```

913     .align 16
914 .update17:
915     cmp     counter,4
916     ble     .cont17
917     fzeros %f21

919     stx    %i1,[%fp+tmp_px]
920     clr    %i0

922     sub    counter,4,counter
923     st     counter,[%fp+tmp_counter]

925     ba     .cont17
926     or     %g0,4,counter

928     .align 16
929 .update18:
930     cmp     counter,5
931     ble     .cont18
932     fzeros %f25

934     stx    %o7,[%fp+tmp_px]
935     sethi  %hi(0x7f800000),%o2

937     sub    counter,5,counter
938     st     counter,[%fp+tmp_counter]

940     ba     .cont18
941     or     %g0,5,counter

943     .align 16
944 .update19:
945     cmp     counter,5
946     ble     .cont19
947     fzeros %f25

949     stx    %o7,[%fp+tmp_px]
950     clr    %o2

952     sub    counter,5,counter
953     st     counter,[%fp+tmp_counter]

955     ba     .cont19
956     or     %g0,5,counter

958     .align 16
959 .update20:
960     cmp     counter,6
961     ble     .cont20
962     fzeros %f0

964     stx    %i1,[%fp+tmp_px]
965     sethi  %hi(0x7f800000),%o1

967     sub    counter,6,counter
968     st     counter,[%fp+tmp_counter]

970     ba     .cont20
971     or     %g0,6,counter

973     .align 16
974 .update21:
975     cmp     counter,6
976     ble     .cont21
977     fzeros %f0

```

```

979     stx    %i1,[%fp+tmp_px]
980     clr    %o1

982     sub    counter,6,counter
983     st     counter,[%fp+tmp_counter]

985     ba     .cont21
986     or     %g0,6,counter

988 .exit:
989     ret
990     restore
991     SET_SIZE(__vsqrtf_ultra3)
unchanged_portion_omitted

```

new/usr/src/lib/libmvec/common/vlog_.c

1

```
*****
1235 Tue Nov 25 12:59:58 2014
new/usr/src/lib/libmvec/common/vlog_.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 extern void __vlog(int, double *, int, double *, int);

32 #if !defined(LIBMVEC_SO_BUILD)
33 #if defined(ARCH_v8plusa) || defined(ARCH_v8plusb) || defined(ARCH_v9a) || defin
34 #define CHECK_ULTRA3
35 #endif
36 #endif /* !defined(LIBMVEC_SO_BUILD) */

38 #ifdef CHECK_ULTRA3
39 #include <strings.h>
40 #define sysinfo _sysinfo
41 #include <sys/systeminfo.h>

43 #define BUFLLEN 257

45 static int use_ultra3 = 0;

47 extern void __vlog_ultra3(int, double *, int, double *, int);
48 #endif

32 #pragma weak vlog_ = __vlog_

34 /* just invoke the serial function */
35 void
36 __vlog_(int *n, double *x, int *stridex, double *y, int *stridey)
37 {
56 #ifdef CHECK_ULTRA3
```

new/usr/src/lib/libmvec/common/vlog_.c

2

```
57 int u;
58 char buf[BUFLLEN];

60 u = use_ultra3;
61 if (!u) {
62     /* use __vlog_ultra3 on Cheetah (and ???) */
63     if (sysinfo(SI_ISALIST, buf, BUFLLEN) > 0 && !strncmp(buf, "sparc
64         u = 3;
65     else
66         u = 1;
67     use_ultra3 = u;
68 }
69 if (u & 2)
70     __vlog_ultra3(*n, x, *stridex, y, *stridey);
71 else
72 #endif
38     __vlog(*n, x, *stridex, y, *stridey);
39 }
    unchanged_portion_omitted
```

```

*****
1235 Tue Nov 25 12:59:58 2014
new/usr/src/lib/libmvec/common/vsin_
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */

30 extern void __vsin(int, double *, int, double *, int);

32 #if !defined(LIBMVEC_SO_BUILD)
33 #if defined(ARCH_v8plusa) || defined(ARCH_v8plusb) || defined(ARCH_v9a) || defin
34 #define CHECK_ULTRA3
35 #endif
36 #endif /* !defined(LIBMVEC_SO_BUILD) */

38 #ifdef CHECK_ULTRA3
39 #include <strings.h>
40 #define sysinfo _sysinfo
41 #include <sys/systeminfo.h>

43 #define BUFLLEN 257

45 static int use_ultra3 = 0;

47 extern void __vsin_ultra3(int, double *, int, double *, int);
48 #endif

32 #pragma weak vsin_ = __vsin_

34 /* just invoke the serial function */
35 void
36 __vsin_(int *n, double *x, int *stridex, double *y, int *stridey)
37 {
56 #ifdef CHECK_ULTRA3

```

```

57     int         u;
58     char        buf[BUFLLEN];

60     u = use_ultra3;
61     if (!u) {
62         /* use __vsin_ultra3 on Cheetah (and ???) */
63         if (sysinfo(SI_ISALIST, buf, BUFLLEN) > 0 && !strncmp(buf, "sparc
64             u = 3;
65         else
66             u = 1;
67         use_ultra3 = u;
68     }
69     if (u & 2)
70         __vsin_ultra3(*n, x, *stridex, y, *stridey);
71     else
72 #endif
38     __vsin(*n, x, *stridex, y, *stridey);
39 }
    unchanged_portion_omitted

```

new/usr/src/lib/libmvec/common/vsqrtrf.c

1

```
*****
1241 Tue Nov 25 12:59:58 2014
new/usr/src/lib/libmvec/common/vsqrtrf.c
5262 libm needs to be carefully undef'd
5268 libm doesn't need to hide symbols which are already local
Reviewed by: Josef 'Jeff' Sipek <jeffpc@josefsipek.net>
Reviewed by: Igor Kozhukhov <ikozhukhov@gmail.com>
Reviewed by: Gordon Ross <gwr@nexenta.com>
Approved by: Gordon Ross <gwr@nexenta.com>
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */
21
22 /*
23  * Copyright 2011 Nexenta Systems, Inc. All rights reserved.
24 */
25 /*
26  * Copyright 2006 Sun Microsystems, Inc. All rights reserved.
27  * Use is subject to license terms.
28 */
29
30 extern void __vsqrtrf(int, float *, int, float *, int);
31
32 #if !defined(LIBMVEC_SO_BUILD)
33 #if defined(ARCH_v8plusa) || defined(ARCH_v8plusb) || defined(ARCH_v9a) || defin
34 #define CHECK_ULTRA3
35 #endif
36 #endif /* !defined(LIBMVEC_SO_BUILD) */
37
38 #ifdef CHECK_ULTRA3
39 #include <strings.h>
40 #define sysinfo _sysinfo
41 #include <sys/systeminfo.h>
42
43 #define BUFLLEN 257
44
45 static int use_ultra3 = 0;
46
47 extern void __vsqrtrf_ultra3(int, float *, int, float *, int);
48 #endif
49
50 #pragma weak vsqrtrf_ = __vsqrtrf_
51
52 /* just invoke the serial function */
53 void
54 __vsqrtrf(int *n, float *x, int *stridex, float *y, int *stridey)
55 {
56 #ifdef CHECK_ULTRA3
```

new/usr/src/lib/libmvec/common/vsqrtrf.c

2

```
57     int         u;
58     char        buf[BUFLLEN];
59
60     u = use_ultra3;
61     if (!u) {
62         /* use __vsqrtrf_ultra3 on Cheetah (and ???) */
63         if (sysinfo(SI_ISALIST, buf, BUFLLEN) > 0 && !strncmp(buf, "sparc
64             u = 3;
65         else
66             u = 1;
67         use_ultra3 = u;
68     }
69     if (u & 2)
70         __vsqrtrf_ultra3(*n, x, *stridex, y, *stridey);
71     else
72 #endif
73         __vsqrtrf(*n, x, *stridex, y, *stridey);
74 }
75
76 unchanged_portion_omitted
```