

```

*****
67270 Wed May 1 07:06:41 2019
new/usr/src/uts/common/io/bscbus.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed 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 2009 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 *
25 * Copyright 2019 Joyent, Inc.
26 *
27 * The "bscbus" driver provides access to the LOMlite2 virtual registers,
28 * so that its clients (children) need not be concerned with the details
29 * of the access mechanism, which in this case is implemented via a
30 * packet-based protocol over a Xbus (similar to ebus) parallel link to the
31 * H8 host interface registers.
32 *
33 * On the other hand, this driver doesn't generally know what the virtual
34 * registers signify - only the clients need this information.
35 */

38 #include <sys/note.h>
39 #include <sys/types.h>
40 #include <sys/conf.h>
41 #include <sys/debug.h>
42 #include <sys/errno.h>
43 #include <sys/file.h>

45 #if defined(__sparc)
46 #include <sys/intr.h>
47 #include <sys/membar.h>
48 #endif

50 #include <sys/kmem.h>
51 #include <sys/modctl.h>
52 #include <sys/note.h>
53 #include <sys/open.h>
54 #include <sys/poll.h>
55 #include <sys/spl.h>
56 #include <sys/stat.h>
57 #include <sys/strlog.h>
58 #include <sys/atomic.h>

60 #include <sys/ddi.h>
61 #include <sys/sunddi.h>

```

```

62 #include <sys/sunddi.h>
63
64 #include <sys/bscbus.h>
65
66 #if defined(NDI_ACC_HDL_V2)
67
68 /*
69  * Compiling for Solaris 10+ with access handle enhancements
70  */
71 #define HANDLE_TYPE ndi_acc_handle_t
72 #define HANDLE_ADDR(hdlp) (hdlp->ah_addr)
73 #define HANDLE_FAULT(hdlp) (hdlp->ah_fault)
74 #define HANDLE_MAPLEN(hdlp) (hdlp->ah_len)
75 #define HANDLE_PRIVATE(hdlp) (hdlp->ah_bus_private)
76
77 #else
78
79 /*
80  * Compatibility definitions for backport to Solaris 8/9
81  */
82 #define HANDLE_TYPE ddi_acc_impl_t
83 #define HANDLE_ADDR(hdlp) (hdlp->ahi_common.ah_addr)
84 #define HANDLE_FAULT(hdlp) (hdlp->ahi_fault)
85 #define HANDLE_MAPLEN(hdlp) (hdlp->ahi_common.ah_len)
86 #define HANDLE_PRIVATE(hdlp) (hdlp->ahi_common.ah_bus_private)
87
88 #define ddi_driver_major(dip) ddi_name_to_major(ddi_binding_name(dip))
89
90 #endif /* NDI_ACC_HDL_V2 */

93 /*
94  * Local definitions
95  */
96 #define MYNAME "bscbus"
97 #define NOMAJOR (~major_t)0
98 #define DUMMY_VALUE ~(int8_t)0

100 #define BSCBUS_INST_TO_MINOR(i) (i)
101 #define BSCBUS_MINOR_TO_INST(m) (m)

103 #define BSCBUS_MAX_CHANNELS (4)

105 #define BSCBUS_DUMMY_ADDRESS ((caddr_t)0x0CADD1ED)
106 #define ADDR_TO_OFFSET(a, hdlp) ((caddr_t)(a) - HANDLE_ADDR(hdlp))
107 #define ADDR_TO_VREG(a) ((caddr_t)(a) - BSCBUS_DUMMY_ADDRESS)
108 #define VREG_TO_ADDR(v) (BSCBUS_DUMMY_ADDRESS + (v))

110 #ifndef DEBUG
111 #define BSCBUS_LOGSTATUS
112 #endif /* DEBUG */

114 #ifdef BSCBUS_LOGSTATUS
115 /*
116  * BSC command logging routines.
117  * Record the data passing to and from the BSC
118  */

120 typedef enum {
121     BSC_CMD_BUSY = 1, /* bsc reports busy */
122     BSC_CMD_CLEARING = 2, /* clearing bsc busy */
123     BSC_CMD_CLEARED = 3, /* cleared bsc busy */
124     BSC_CMD_SENDING = 4, /* sending next byte */
125     BSC_CMD_SENT = 5, /* sending last byte */
126     BSC_CMD_PENDING = 6, /* got sent byte ack */
127     BSC_CMD_REPLY = 7, /* got reply byte */

```

```

128     BSC_CMD_COMPLETE = 8,           /* command complete */
129     BSC_CMD_ERROR_SEQ = 9,          /* error status */
130     BSC_CMD_ERROR_STATUS = 10,      /* error status */
131     BSC_CMD_ERROR_OFLOW = 11,       /* error status */
132     BSC_CMD_ERROR_TOUT = 12,        /* error status */
133
134     BSC_CMD_PROCESS = 13,           /* async intr */
135     BSC_CMD_VLINTR = 14,            /* vl intr */
136     BSC_CMD_VLINTRUNCL = 15,        /* vl intr unclaim */
137     BSC_CMD_DOGPAT = 17,            /* watchdog pat */
138 } bsc_cmd_stamp_t;
    unchanged portion omitted
408 #else /* DEBUG */
409 #define bscbus_trace(...) (void)(0)
407 #define bscbus_trace
410 #endif /* DEBUG */

412 static struct bscbus_state *
413 bscbus_getstate(dev_info_t *dip, int instance, const char *caller)
414 {
415     struct bscbus_state *ssp = NULL;
416     dev_info_t *sdip = NULL;
417     major_t dmaj = NOMAJOR;

419     if (dip != NULL) {
420         /*
421          * Use the instance number from the <dip>; also,
422          * check that it really corresponds to this driver
423          */
424         instance = ddi_get_instance(dip);
425         dmaj = ddi_driver_major(dip);
426         if (bscbus_major == NOMAJOR && dmaj != NOMAJOR)
427             bscbus_major = dmaj;
428         else if (dmaj != bscbus_major) {
429             cmn_err(CE_WARN,
430                  "%s: major number mismatch (%d vs. %d) in %s(),",
431                  "probably due to child misconfiguration",
432                  MYNAME, bscbus_major, dmaj, caller);
433             instance = -1;
434         }
435     }

437     if (instance >= 0)
438         ssp = ddi_get_soft_state(bscbus_statep, instance);
439     if (ssp != NULL) {
440         sdip = ssp->dip;
441         if (dip == NULL && sdip == NULL)
442             ssp = NULL;
443         else if (dip != NULL && sdip != NULL && sdip != dip) {
444             cmn_err(CE_WARN,
445                  "%s: devinfo mismatch (%p vs. %p) in %s(),",
446                  "probably due to child misconfiguration",
447                  MYNAME, (void *)dip, (void *)sdip, caller);
448             ssp = NULL;
449         }
450     }

452     return (ssp);
453 }
    unchanged portion omitted

```

```

*****
168308 Wed May 1 07:06:42 2019
new/usr/src/uts/common/io/bscv.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed 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 2009 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
26 /*
27  * Copyright 2019 Joyent, Inc.
28  */
30 /*
31  * bscv.c - multi-threaded lom driver for the Stiletto platform.
32  */
34 /*
35  * Included files.
36  */
38 #include <sys/note.h>
39 #include <sys/types.h>
40 #include <sys/param.h>
41 #include <sys/uio.h>
42 #include <sys/open.h>
43 #include <sys/cred.h>
44 #include <sys/stream.h>
45 #include <sys/system.h>
46 #include <sys/conf.h>
47 #include <sys/reboot.h>
48 #include <sys/modctl.h>
49 #include <sys/mkdev.h>
50 #include <sys/errno.h>
51 #include <sys/debug.h>
52 #include <sys/kmem.h>
53 #include <sys/consdev.h>
54 #include <sys/file.h>
55 #include <sys/stat.h>
56 #include <sys/disp.h>
57 #include <sys/ddi.h>
58 #include <sys/sunddi.h>
59 #include <sys/stream.h>
60 #include <sys/strlog.h>
61 #include <sys/log.h>

```

```

62 #include <sys/utsname.h>
63 #include <sys/callb.h>
64 #include <sys/sysevent.h>
65 #include <sys/nvpair.h>
66 #include <sys/sysevent/eventdefs.h>
67 #include <sys/sysevent/domain.h>
68 #include <sys/sysevent/env.h>
69 #include <sys/sysevent/dr.h>
71 #include <sys/lom_io.h>
72 #include <sys/bscbus.h>
73 #include <sys/bscv_impl.h>
75 /*
76  * Variables defined here and visible internally only
77  */
79 static void *bscv_statep = NULL;
81 /*
82  * Forward declarations
83  */
85 static int bscv_getinfo(dev_info_t *, ddi_info_cmd_t, void *, void **);
86 static int bscv_attach(dev_info_t *, ddi_attach_cmd_t);
87 static int bscv_detach(dev_info_t *, ddi_detach_cmd_t);
88 static int bscv_quiesce(dev_info_t *);
89 static int bscv_map_regs(bscv_soft_state_t *);
90 static void bscv_unmap_regs(bscv_soft_state_t *);
91 static void bscv_map_chan_logical_physical(bscv_soft_state_t *);
93 static int bscv_open(dev_t *, int, int, cred_t *);
94 static int bscv_close(dev_t, int, int, cred_t *);
95 static void bscv_full_stop(bscv_soft_state_t *);
97 static void bscv_enter(bscv_soft_state_t *);
98 static int bscv_tryenter(bscv_soft_state_t *ssp);
99 static void bscv_exit(bscv_soft_state_t *);
100 #ifdef DEBUG
101 static int bscv_held(bscv_soft_state_t *);
102 #endif /* DEBUG */
104 static void bscv_put8(bscv_soft_state_t *, int, bscv_addr_t, uint8_t);
105 static void bscv_put16(bscv_soft_state_t *, int, bscv_addr_t, uint16_t);
106 static void bscv_put32(bscv_soft_state_t *, int, bscv_addr_t, uint32_t);
107 static uint8_t bscv_get8(bscv_soft_state_t *, int, bscv_addr_t);
108 static uint16_t bscv_get16(bscv_soft_state_t *, int, bscv_addr_t);
109 static uint32_t bscv_get32(bscv_soft_state_t *, int, bscv_addr_t);
110 static void bscv_setclear8(bscv_soft_state_t *, int,
111     bscv_addr_t, uint8_t, uint8_t);
112 static void bscv_setclear8_volatile(bscv_soft_state_t *, int,
113     bscv_addr_t, uint8_t, uint8_t);
114 static void bscv_rep_rw8(bscv_soft_state_t *, int,
115     uint8_t *, bscv_addr_t, size_t, uint_t, boolean_t);
116 static uint8_t bscv_get8_cached(bscv_soft_state_t *, bscv_addr_t);
118 static uint8_t bscv_get8_locked(bscv_soft_state_t *, int, bscv_addr_t, int *);
119 static void bscv_rep_get8_locked(bscv_soft_state_t *, int,
120     uint8_t *, bscv_addr_t, size_t, uint_t, int *);
122 static boolean_t bscv_faulty(bscv_soft_state_t *);
123 static void bscv_clear_fault(bscv_soft_state_t *);
124 static void bscv_set_fault(bscv_soft_state_t *);
125 static boolean_t bscv_session_error(bscv_soft_state_t *);
126 static int bscv_retcode(bscv_soft_state_t *);
127 static int bscv_should_retry(bscv_soft_state_t *);

```

```

128 static void bscv_locked_result(bscv_soft_state_t *, int *);

130 static void bscv_put8_once(bscv_soft_state_t *, int, bscv_addr_t, uint8_t);
131 static uint8_t bscv_get8_once(bscv_soft_state_t *, int, bscv_addr_t);
132 static uint32_t bscv_probe(bscv_soft_state_t *, int, uint32_t *);
133 static void bscv_resync_comms(bscv_soft_state_t *, int);

135 static boolean_t bscv_window_setup(bscv_soft_state_t *);
136 static int bscv_eerw(bscv_soft_state_t *, uint32_t, uint8_t *,
137     unsigned, boolean_t);

139 static int bscv_ioctl(dev_t, int, intptr_t, int, cred_t *, int *);
140 static int bscv_ioc_dogstate(bscv_soft_state_t *, intptr_t, int);
141 static int bscv_ioc_psustate(bscv_soft_state_t *, intptr_t, int);
142 static int bscv_ioc_fanstate(bscv_soft_state_t *, intptr_t, int);
143 static int bscv_ioc_fledstate(bscv_soft_state_t *, intptr_t, int);
144 static int bscv_ioc_ledstate(bscv_soft_state_t *, intptr_t, int);
145 static int bscv_ioc_info(bscv_soft_state_t *, intptr_t, int);
146 static int bscv_ioc_mread(bscv_soft_state_t *, intptr_t, int);
147 static int bscv_ioc_volts(bscv_soft_state_t *, intptr_t, int);
148 static int bscv_ioc_stats(bscv_soft_state_t *, intptr_t, int);
149 static int bscv_ioc_temp(bscv_soft_state_t *, intptr_t, int);
150 static int bscv_ioc_cons(bscv_soft_state_t *, intptr_t, int);
151 static int bscv_ioc_eventlog2(bscv_soft_state_t *, intptr_t, int);
152 static int bscv_ioc_info2(bscv_soft_state_t *, intptr_t, int);
153 static int bscv_ioc_test(bscv_soft_state_t *, intptr_t, int);
154 static int bscv_ioc_mprog2(bscv_soft_state_t *, intptr_t, int);
155 static int bscv_ioc_mread2(bscv_soft_state_t *, intptr_t, int);

157 static void bscv_event_daemon(void *);
158 static void bscv_start_event_daemon(bscv_soft_state_t *);
159 static int bscv_stop_event_daemon(bscv_soft_state_t *);
160 static int bscv_pause_event_daemon(bscv_soft_state_t *);
161 static void bscv_resume_event_daemon(bscv_soft_state_t *);
162 static void bscv_event_process(bscv_soft_state_t *ssp, boolean_t);
163 static int bscv_event_validate(bscv_soft_state_t *, uint32_t, uint8_t);
164 static void bscv_event_process_one(bscv_soft_state_t *, lom_event_t *);
165 static void bscv_build_eventstring(bscv_soft_state_t *,
166     lom_event_t *, char *, char *);
167 static int bscv_level_of_event(lom_event_t *);
168 static void bscv_status(bscv_soft_state_t *, uint8_t, uint8_t);
169 char *bscv_get_label(char [][MAX_LOM2_NAME_STR], int, int);
170 static void bscv_generic_sysevent(bscv_soft_state_t *, char *, char *, char *,
171     char *, int32_t, char *);
172 static void bscv_sysevent(bscv_soft_state_t *, lom_event_t *);

174 static int bscv_prog(bscv_soft_state_t *, intptr_t, int);
175 static int bscv_prog_image(bscv_soft_state_t *, boolean_t,
176     uint8_t *, int, uint32_t);
177 static int bscv_prog_receive_image(bscv_soft_state_t *, lom_prog_t *,
178     uint8_t *, int);
179 static void bscv_leave_programming_mode(bscv_soft_state_t *, boolean_t);
180 static int bscv_prog_stop_lom(bscv_soft_state_t *);
181 static int bscv_prog_start_lom(bscv_soft_state_t *);

183 static int bscv_attach_common(bscv_soft_state_t *);
184 static int bscv_cleanup(bscv_soft_state_t *);
185 static void bscv_setup_capability(bscv_soft_state_t *);
186 static int bscv_probe_check(bscv_soft_state_t *);
187 static void bscv_setup_hostname(bscv_soft_state_t *);
188 static void bscv_read_hostname(bscv_soft_state_t *, char *);
189 static void bscv_write_hostname(bscv_soft_state_t *, char *, uint8_t);
190 static void bscv_setup_static_info(bscv_soft_state_t *);
191 static uint8_t bscv_read_env_name(bscv_soft_state_t *, uint8_t,
192     uint8_t, uint8_t, char [][MAX_LOM2_NAME_STR], int);
193 static void bscv_setup_events(bscv_soft_state_t *);

```

```

195 static void bscv_trace(bscv_soft_state_t *, char, const char *,
196     const char *, ...);

198 #ifdef __sparc
199 static void bscv_idi_init();
200 static void bscv_idi_fini();
201 static void bscv_idi_new_instance(dev_info_t *dip);
202 static void bscv_idi_clear_err();
203 void bscv_idi_set(struct bscv_idi_info info);
204 static boolean_t bscv_idi_err();
205 static boolean_t bscv_nodename_set(struct bscv_idi_info info);
206 static boolean_t bscv_sig_set(struct bscv_idi_info info);
207 static boolean_t bscv_wdog_pat(struct bscv_idi_info info);
208 static boolean_t bscv_wdog_cfg(struct bscv_idi_info info);
209 static void bscv_write_sig(bscv_soft_state_t *ssp, bscv_sig_t s);
210 #endif /* __sparc */

212 static void bscv_setup_watchdog(bscv_soft_state_t *ssp);
213 static void bscv_write_wdog_cfg(bscv_soft_state_t *,
214     uint_t, boolean_t, uint8_t);

216 #if defined(__i386) || defined(__amd64)
217 static void bscv_inform_bsc(bscv_soft_state_t *, uint32_t);
218 static void bscv_watchdog_pat_request(void *);
219 static void bscv_watchdog_cfg_request(bscv_soft_state_t *, uint8_t);
220 static uint_t bscv_set_watchdog_timer(bscv_soft_state_t *, uint_t);
221 static void bscv_clear_watchdog_timer(bscv_soft_state_t *);

223 static boolean_t bscv_panic_callback(void *, int);
224 static void bscv_watchdog_cyclic_add(bscv_soft_state_t *);
225 static void bscv_watchdog_cyclic_remove(bscv_soft_state_t *);

227 static uint8_t wdog_reset_on_timeout = 1;

229 #define WDOG_ON 1
230 #define WDOG_OFF 0
231 #define CLK_WATCHDOG_DEFAULT 10 /* 10 seconds */
232 #define WATCHDOG_PAT_INTERVAL 100000000 /* 1 second */

234 static int bscv_watchdog_enable;
235 static int bscv_watchdog_available;
236 static int watchdog_activated;
237 static uint_t bscv_watchdog_timeout_seconds;
238 #endif /* __i386 || __amd64 */

240 #ifdef __sparc
241 struct bscv_idi_callout bscv_idi_callout_table[] = {
242     {BSCV_IDI_NODENAME, &bscv_nodename_set},
243     {BSCV_IDI_SIG, &bscv_sig_set},
244     {BSCV_IDI_WDOG_PAT, &bscv_wdog_pat},
245     {BSCV_IDI_WDOG_CFG, &bscv_wdog_cfg},
246     {BSCV_IDI_NULL, NULL}
247 };
    unchanged_portion_omitted
248 #endif

404 #ifdef DEBUG
405 /* Tracing is enabled if value is non-zero. */
406 static int bscv_trace_flag = 1;

408 #define BSCV_TRACE if (bscv_trace_flag != 0) bscv_trace
409 #else
410 #define BSCV_TRACE(...) (void)(0)
407 #define BSCV_TRACE
411 #endif

```

```
413 /*
414  * kernel accessible routines. These routines are necessarily global so the
415  * driver can be loaded, and unloaded successfully
416  */

418 /*
419  * function      - _init
420  * description   - initialize the driver state structure and installs the
421  *                driver module into the kernel
422  * inputs        - none
423  * outputs       - success or failure of module installation
424  */

426 int
427 _init(void)
428 {
429     register int e;

431     if ((e = ddi_soft_state_init(&bscv_statep,
432     sizeof (bscv_soft_state_t), 1)) != 0) {
433         return (e);
434     }

436     if ((e = mod_install(&modlinkage)) != 0) {
437         ddi_soft_state_fini(&bscv_statep);
438     }

440 #ifdef __sparc
441     if (e == 0) bscv_idi_init();
442 #endif /* __sparc */
443     return (e);
444 }
_____unchanged_portion_omitted_____
```

new/usr/src/uts/common/io/mwl/mwl.c

1

```
*****
114894 Wed May 1 07:06:42 2019
new/usr/src/uts/common/io/mwl/mwl.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
3  * Use is subject to license terms.
4  */
6 /*
7  * Copyright (c) 2007-2009 Sam Leffler, Errno Consulting
8  * Copyright (c) 2007-2008 Marvell Semiconductor, Inc.
9  * All rights reserved.
10 *
11 * Redistribution and use in source and binary forms, with or without
12 * modification, are permitted provided that the following conditions
13 * are met:
14 * 1. Redistributions of source code must retain the above copyright
15 * notice, this list of conditions and the following disclaimer,
16 * without modification.
17 * 2. Redistributions in binary form must reproduce at minimum a disclaimer
18 * similar to the "NO WARRANTY" disclaimer below ("Disclaimer") and any
19 * redistribution must be conditioned upon including a substantially
20 * similar Disclaimer requirement for further binary redistribution.
21 *
22 * NO WARRANTY
23 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
24 * 'AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
25 * LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY
26 * AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
27 * THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR SPECIAL, EXEMPLARY,
28 * OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
29 * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
30 * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
31 * IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
32 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
33 * THE POSSIBILITY OF SUCH DAMAGES.
34 */
36 /*
37  * Copyright 2019 Joyent, Inc.
38  */
40 /*
41  * Driver for the Marvell 88W8363 Wireless LAN controller.
42  */
43 #include <sys/stat.h>
44 #include <sys/dlpi.h>
45 #include <inet/common.h>
46 #include <inet/mi.h>
47 #include <sys/stream.h>
48 #include <sys/errno.h>
49 #include <sys/stropts.h>
50 #include <sys/stat.h>
51 #include <sys/sunddi.h>
52 #include <sys/strsubr.h>
53 #include <sys/strsun.h>
54 #include <sys/pci.h>
55 #include <sys/mac_provider.h>
56 #include <sys/mac_wifi.h>
57 #include <sys/net80211.h>
58 #include <inet/wifi_ioctl.h>
60 #include "mwl_var.h"
```

new/usr/src/uts/common/io/mwl/mwl.c

2

```
62 static int mwl_attach(dev_info_t *devinfo, ddi_attach_cmd_t cmd);
63 static int mwl_detach(dev_info_t *devinfo, ddi_detach_cmd_t cmd);
64 static int mwl_quiesce(dev_info_t *devinfo);
66 DDI_DEFINE_STREAM_OPS(mwl_dev_ops, nulldev, nulldev, mwl_attach, mwl_detach,
67     nodev, NULL, D_MP, NULL, mwl_quiesce);
69 static struct modldrv mwl_modldrv = {
70     &mod_driverops, /* Type of module. This one is a driver */
71     "Marvell 88W8363 WiFi driver v1.1", /* short description */
72     &mwl_dev_ops /* driver specific ops */
73 };
unchanged portion omitted
117 #define MWL_DBG_ATTACH (1 << 0)
118 #define MWL_DBG_DMA (1 << 1)
119 #define MWL_DBG_FW (1 << 2)
120 #define MWL_DBG_HW (1 << 3)
121 #define MWL_DBG_INTR (1 << 4)
122 #define MWL_DBG_RX (1 << 5)
123 #define MWL_DBG_TX (1 << 6)
124 #define MWL_DBG_CMD (1 << 7)
125 #define MWL_DBG_CRYPT0 (1 << 8)
126 #define MWL_DBG_SR (1 << 9)
127 #define MWL_DBG_MSG (1 << 10)
129 uint32_t mwl_dbg_flags = 0x0;
131 #ifdef DEBUG
132 #define MWL_DBG \
133     mwl_debug
134 #else
135 #define MWL_DBG(...) (void)(0)
136 #endif
138 /*
139  * PIO access attributes for registers
140  */
141 static ddi_device_acc_attr_t mwl_reg_accattr = {
142     DDI_DEVICE_ATTR_V0,
143     DDI_STRUCTURE_LE_ACC,
144     DDI_STRICTORDER_ACC,
145     DDI_DEFAULT_ACC
146 };
unchanged portion omitted
```

new/usr/src/uts/common/io/rtw/rtwvar.h

1

```
*****
13745 Wed May 1 07:06:43 2019
new/usr/src/uts/common/io/rtw/rtwvar.h
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * Copyright 2009 Sun Microsystems, Inc. All rights reserved.
3  * Use is subject to license terms.
4  *
5  * Copyright 2019 Joyent, Inc.
6  */
7 /*
8  * Copyright (c) 2004, 2005 David Young. All rights reserved.
9  *
10 * Driver for the Realtek RTL8180 802.11 MAC/BBP by David Young.
11 *
12 * Redistribution and use in source and binary forms, with or without
13 * modification, are permitted provided that the following conditions
14 * are met:
15 * 1. Redistributions of source code must retain the above copyright
16 * notice, this list of conditions and the following disclaimer.
17 * 2. Redistributions in binary form must reproduce the above copyright
18 * notice, this list of conditions and the following disclaimer in the
19 * documentation and/or other materials provided with the distribution.
20 * 3. The name of David Young may not be used to endorse or promote
21 * products derived from this software without specific prior
22 * written permission.
23 *
24 * THIS SOFTWARE IS PROVIDED BY David Young ``AS IS'' AND ANY
25 * EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
26 * THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
27 * PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL David
28 * Young BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
29 * EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
30 * TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
31 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
32 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
33 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
34 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
35 * OF SUCH DAMAGE.
36 */
37 #ifndef _RTWVAR_H_
38 #define _RTWVAR_H_
39
40 #ifdef __cplusplus
41 extern "C" {
42 #endif
43
44 #include <sys/list.h>
45 #include <sys/net80211.h>
46
47 #ifndef __func__
48 #define __func__ ""
49 #endif
50
51 extern void rtw_dbg(uint32_t dbg_flags, const int8_t *fmt, ...);
52
53 #define RTW_DEBUG_TUNE 0x000001
54 #define RTW_DEBUG_PKTFLT 0x000002
55 #define RTW_DEBUG_XMIT 0x000004
56 #define RTW_DEBUG_DMA 0x000008
57 #define RTW_DEBUG_NODE 0x000010
58 #define RTW_DEBUG_PWR 0x000020
59 #define RTW_DEBUG_ATTACH 0x000040
60 #define RTW_DEBUG_REGDUMP 0x000080
61 #define RTW_DEBUG_ACCESS 0x000100
```

new/usr/src/uts/common/io/rtw/rtwvar.h

2

```
62 #define RTW_DEBUG_RESET 0x000200
63 #define RTW_DEBUG_INIT 0x000400
64 #define RTW_DEBUG_PKT_DUMP 0x000800
65 #define RTW_DEBUG_RECV 0x001000
66 #define RTW_DEBUG_RECV_DESC 0x002000
67 #define RTW_DEBUG_IOSTATE 0x004000
68 #define RTW_DEBUG_INTR 0x008000
69 #define RTW_DEBUG_PHY 0x010000
70 #define RTW_DEBUG_PHYIO 0x020000
71 #define RTW_DEBUG_PHYBITIO 0x040000
72 #define RTW_DEBUG_TIMEOUT 0x080000
73 #define RTW_DEBUG_BUGS 0x100000
74 #define RTW_DEBUG_BEACON 0x200000
75 #define RTW_DEBUG_WIFICFG 0x400000
76 #define RTW_DEBUG_80211 0x800000
77 #define RTW_DEBUG_MAX 0xfffff

79 #ifdef DEBUG
80 #define RTW_DPRINTF \
81     rtw_dbg
82 #else /* DEBUG */
83 #define RTW_DPRINTF(...) (void)(0)
84 #endif /* DEBUG */

86 enum rtw_locale {
87     RTW_LOCALE_USA = 0,
88     RTW_LOCALE_EUROPE,
89     RTW_LOCALE_JAPAN,
90     RTW_LOCALE_UNKNOWN
91 };
unchanged_portion_omitted
```

new/usr/src/uts/common/io/rwd/rt2661.c

1

```
*****
83101 Wed May 1 07:06:43 2019
new/usr/src/uts/common/io/rwd/rt2661.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
3  * Use is subject to license terms.
4  */

6 /*
7  * Copyright (c) 2006
8  * Damien Bergamini <damien.bergamini@free.fr>
9  *
10 * Permission to use, copy, modify, and distribute this software for any
11 * purpose with or without fee is hereby granted, provided that the above
12 * copyright notice and this permission notice appear in all copies.
13 *
14 * THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
15 * WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
16 * MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
17 * ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
18 * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
19 * ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
20 * OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
21 */

23 /*
24  * Ralink Technology RT2561, RT2561S and RT2661 chipset driver
25  * http://www.ralinktech.com/
26  */

28 /*
29  * Copyright 2019 Joyent, Inc.
30  */

32 #include <sys/types.h>
33 #include <sys/byteorder.h>
34 #include <sys/conf.h>
35 #include <sys/cmn_err.h>
36 #include <sys/stat.h>
37 #include <sys/ddi.h>
38 #include <sys/sunddi.h>
39 #include <sys/strsubr.h>
40 #include <sys/ethernet.h>
41 #include <inet/common.h>
42 #include <inet/nd.h>
43 #include <inet/mi.h>
44 #include <sys/note.h>
45 #include <sys/stream.h>
46 #include <sys/strsun.h>
47 #include <sys/modctl.h>
48 #include <sys/devops.h>
49 #include <sys/dlpi.h>
50 #include <sys/mac_provider.h>
51 #include <sys/mac_wifi.h>
52 #include <sys/net80211.h>
53 #include <sys/net80211_proto.h>
54 #include <sys/varargs.h>
55 #include <sys/policy.h>
56 #include <sys/pci.h>
57 #include <sys/crypto/common.h>
58 #include <sys/crypto/api.h>
59 #include <inet/wifi_ioctl.h>

61 #include "rt2661_reg.h"
```

new/usr/src/uts/common/io/rwd/rt2661.c

2

```
62 #include "rt2661_var.h"
63 #include "rt2661_ucode.h"

65 #define RT2661_DBG_80211 (1 << 0)
66 #define RT2661_DBG_DMA (1 << 1)
67 #define RT2661_DBG_EEPROM (1 << 2)
68 #define RT2661_DBG_FW (1 << 3)
69 #define RT2661_DBG_HW (1 << 4)
70 #define RT2661_DBG_INTR (1 << 5)
71 #define RT2661_DBG_RX (1 << 6)
72 #define RT2661_DBG_SCAN (1 << 7)
73 #define RT2661_DBG_TX (1 << 8)
74 #define RT2661_DBG_RADIO (1 << 9)
75 #define RT2661_DBG_RESUME (1 << 10)
76 #define RT2661_DBG_MSG (1 << 11)

78 uint32_t rt2661_dbg_flags = 0;

80 #ifdef DEBUG
81 #define RWD_DEBUG \
82     rt2661_debug
83 #else
84 #define RWD_DEBUG(...) (void)(0)
85 #define RWD_DEBUG
86 #endif

87 static void *rt2661_soft_state_p = NULL;

89 static const uint8_t *ucode = NULL;
90 int usize;

92 static const struct {
93     uint32_t reg;
94     uint32_t val;
95 } rt2661_def_mac[] = {
    unchanged_portion_omitted
```



new/usr/src/uts/common/io/rwn/rt2860.c

1

```
*****
85657 Wed May 1 07:06:44 2019
new/usr/src/uts/common/io/rwn/rt2860.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * Copyright 2017 Gary Mills
3  * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
4  * Use is subject to license terms.
5  * Copyright 2019 Joyent, Inc.
5  * Copyright (c) 2018, Joyent, Inc.
6  */

8 /*
9  * Copyright (c) 2007, 2008
10  * Damien Bergamini <damien.bergamini@free.fr>
11  *
12  * Permission to use, copy, modify, and distribute this software for any
13  * purpose with or without fee is hereby granted, provided that the above
14  * copyright notice and this permission notice appear in all copies.
15  *
16  * THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
17  * WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
18  * MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
19  * ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
20  * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
21  * ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
22  * OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
23  */

25 /*
26  * Ralink Technology RT2860 chipset driver
27  * http://www.ralinktech.com/
28  */

30 #include <sys/types.h>
31 #include <sys/byteorder.h>
32 #include <sys/conf.h>
33 #include <sys/cmn_err.h>
34 #include <sys/stat.h>
35 #include <sys/ddi.h>
36 #include <sys/sunddi.h>
37 #include <sys/strsubr.h>
38 #include <inet/common.h>
39 #include <sys/note.h>
40 #include <sys/stream.h>
41 #include <sys/strsun.h>
42 #include <sys/modctl.h>
43 #include <sys/devops.h>
44 #include <sys/mac_provider.h>
45 #include <sys/mac_wifi.h>
46 #include <sys/net80211.h>
47 #include <sys/net80211_proto.h>
48 #include <sys/varargs.h>
49 #include <sys/pci.h>
50 #include <sys/crypto/common.h>
51 #include <sys/crypto/api.h>
52 #include <inet/wifi_ioctl.h>

54 #include "rt2860_reg.h"
55 #include "rt2860_var.h"

57 #define RT2860_DBG_80211 (1 << 0)
58 #define RT2860_DBG_DMA (1 << 1)
59 #define RT2860_DBG_EEPROM (1 << 2)
60 #define RT2860_DBG_FW (1 << 3)
```

new/usr/src/uts/common/io/rwn/rt2860.c

2

```
61 #define RT2860_DBG_HW (1 << 4)
62 #define RT2860_DBG_INTR (1 << 5)
63 #define RT2860_DBG_RX (1 << 6)
64 #define RT2860_DBG_SCAN (1 << 7)
65 #define RT2860_DBG_TX (1 << 8)
66 #define RT2860_DBG_RADIO (1 << 9)
67 #define RT2860_DBG_RESUME (1 << 10)
68 #define RT2860_DBG_MSG (1 << 11)

70 uint32_t rt2860_dbg_flags = 0x0;

72 #ifdef DEBUG
73 #define RWN_DEBUG \
74     rt2860_debug
75 #else
76 #define RWN_DEBUG(...) (void)(0)
76 #define RWN_DEBUG
77 #endif

79 static void *rt2860_soft_state_p = NULL;
80 static uint8_t rt2860_fw_bin [] = {
81 #include "fw-rt2860/rt2860.ucode"
82 };
    unchanged portion omitted
```

new/usr/src/uts/common/io/uath/uath.c

1

```
*****
89556 Wed May 1 07:06:44 2019
new/usr/src/uts/common/io/uath/uath.c
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * Copyright 2010 Sun Microsystems, Inc. All rights reserved.
3  * Use is subject to license terms.
4  */

6 /*
7  * Copyright 2019 Joyent, Inc.
8  */

10 /*
11  * Copyright (c) 2006 Sam Leffler, Errno Consulting
12  * Copyright (c) 2008-2009 Weongyo Jeong <weongyo@freebsd.org>
13  * All rights reserved.
14  *
15  * Redistribution and use in source and binary forms, with or without
16  * modification, are permitted provided that the following conditions
17  * are met:
18  * 1. Redistributions of source code must retain the above copyright
19  * notice, this list of conditions and the following disclaimer,
20  * without modification.
21  * 2. Redistributions in binary form must reproduce at minimum a disclaimer
22  * similar to the "NO WARRANTY" disclaimer below ("Disclaimer") and any
23  * redistribution must be conditioned upon including a substantially
24  * similar Disclaimer requirement for further binary redistribution.
25  *
26  * NO WARRANTY
27  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
28  * 'AS IS' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
29  * LIMITED TO, THE IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY
30  * AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL
31  * THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR SPECIAL, EXEMPLARY,
32  * OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
33  * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
34  * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
35  * IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
36  * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
37  * THE POSSIBILITY OF SUCH DAMAGES.
38  */

40 /*
41  * This driver is distantly derived from a driver of the same name
42  * by Damien Bergamini. The original copyright is included below:
43  *
44  * Copyright (c) 2006
45  * Damien Bergamini <dami.bergamini@free.fr>
46  *
47  * Permission to use, copy, modify, and distribute this software for any
48  * purpose with or without fee is hereby granted, provided that the above
49  * copyright notice and this permission notice appear in all copies.
50  *
51  * THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
52  * WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
53  * MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
54  * ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
55  * WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
56  * ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
57  * OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
58  */
```

61 #include <sys/types.h>

new/usr/src/uts/common/io/uath/uath.c

2

```
62 #include <sys/cmn_err.h>
63 #include <sys/strsubr.h>
64 #include <sys/strsun.h>
65 #include <sys/modctl.h>
66 #include <sys/devops.h>
67 #include <sys/byteorder.h>
68 #include <sys/mac_provider.h>
69 #include <sys/mac_wifi.h>
70 #include <sys/net80211.h>

72 #define USBDRV_MAJOR_VER 2
73 #define USBDRV_MINOR_VER 0
74 #include <sys/usb/usba.h>
75 #include <sys/usb/usba/usba_types.h>

77 #include "uath_reg.h"
78 #include "uath_var.h"

80 static void *uath_soft_state_p = NULL;

82 /*
83  * Bit flags in the ral_dbg_flags
84  */
85 #define UATH_DBG_MSG 0x000001
86 #define UATH_DBG_ERR 0x000002
87 #define UATH_DBG_USB 0x000004
88 #define UATH_DBG_TX 0x000008
89 #define UATH_DBG_RX 0x000010
90 #define UATH_DBG_FW 0x000020
91 #define UATH_DBG_TX_CMD 0x000040
92 #define UATH_DBG_RX_CMD 0x000080
93 #define UATH_DBG_ALL 0x0000ff

95 uint32_t uath_dbg_flags = 0;

97 #ifdef DEBUG
98 #define UATH_DEBUG \
99     uath_debug
100 #else
101 #define UATH_DEBUG(...) (void)(0)
102 #define UATH_DEBUG
103 #endif

104 /*
105  * Various supported device vendors/products.
106  * UB51: AR5005UG 802.11b/g, UB52: AR5005UX 802.11b/g
107  */
108 #define UATH_FLAG_PRE_FIRMWARE (1 << 0)
109 #define UATH_FLAG_ABG (1 << 1)
110 #define UATH_FLAG_ERR (1 << 2)
111 #define UATH_DEV(v, p, f) \
112     { { USB_VENDOR_#v, USB_PRODUCT_#v##_##p }, (f) }, \
113     { { USB_VENDOR_#v, USB_PRODUCT_#v##_##p##_NF }, \
114     (f) | UATH_FLAG_PRE_FIRMWARE }
115 #define UATH_DEV_UG(v, p) UATH_DEV(v, p, 0)
116 #define UATH_DEV_UX(v, p) UATH_DEV(v, p, UATH_FLAG_ABG)

118 struct uath_devno {
119     uint16_t vendor_id;
120     uint16_t product_id;
121 };
_____unchanged_portion_omitted_____
```

new/usr/src/uts/common/sys/fibre-channel/impl/fp.h

1

```
*****
18631 Wed May 1 07:06:45 2019
new/usr/src/uts/common/sys/fibre-channel/impl/fp.h
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed 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 2008 Sun Microsystems, Inc. All rights reserved.
23 * Use is subject to license terms.
24 */
25
26 /*
27  * Copyright 2019 Joyent, Inc.
28  */
29
30 #ifndef _FP_H
31 #define _FP_H
32
33 #include <sys/note.h>
34
35 #ifdef __cplusplus
36 extern "C" {
37 #endif
38
39 /*
40  * Debugging, Error reporting, and tracing
41  */
42 #define FP_LOG_SIZE          1024 * 1024
43
44 #define FP_LEVEL_1          0x00001    /* attach/detach PM CPR */
45 #define FP_LEVEL_2          0x00002    /* startup */
46 #define FP_LEVEL_3          0x00004    /* state change, discovery */
47 #define FP_LEVEL_4          0x00008    /* statec/devc to ULPs */
48 #define FP_LEVEL_5          0x00010    /* FCA UB callbacks */
49 #define FP_LEVEL_6          0x00020    /* Name Server */
50 #define FP_LEVEL_7          0x00040    /* RSCN */
51 #define FP_LEVEL_8          0x00080    /* I/O tracing */
52 #define FP_LEVEL_9          0x00100    /* Failure messages */
53
54
55 /*
56  * Log contents to system messages file
57  */
58 #define FP_MSG_LEVEL_1      (FP_LEVEL_1 | FC_TRACE_LOG_MSG)
59 #define FP_MSG_LEVEL_2      (FP_LEVEL_2 | FC_TRACE_LOG_MSG)
60 #define FP_MSG_LEVEL_3      (FP_LEVEL_3 | FC_TRACE_LOG_MSG)
```

new/usr/src/uts/common/sys/fibre-channel/impl/fp.h

2

```
62 #define FP_MSG_LEVEL_4      (FP_LEVEL_4 | FC_TRACE_LOG_MSG)
63 #define FP_MSG_LEVEL_5      (FP_LEVEL_5 | FC_TRACE_LOG_MSG)
64 #define FP_MSG_LEVEL_6      (FP_LEVEL_6 | FC_TRACE_LOG_MSG)
65 #define FP_MSG_LEVEL_7      (FP_LEVEL_7 | FC_TRACE_LOG_MSG)
66 #define FP_MSG_LEVEL_8      (FP_LEVEL_8 | FC_TRACE_LOG_MSG)
67 #define FP_MSG_LEVEL_9      (FP_LEVEL_9 | FC_TRACE_LOG_MSG)
68
69
70 /*
71  * Log contents to trace buffer
72  */
73 #define FP_BUF_LEVEL_1      (FP_LEVEL_1 | FC_TRACE_LOG_BUF)
74 #define FP_BUF_LEVEL_2      (FP_LEVEL_2 | FC_TRACE_LOG_BUF)
75 #define FP_BUF_LEVEL_3      (FP_LEVEL_3 | FC_TRACE_LOG_BUF)
76 #define FP_BUF_LEVEL_4      (FP_LEVEL_4 | FC_TRACE_LOG_BUF)
77 #define FP_BUF_LEVEL_5      (FP_LEVEL_5 | FC_TRACE_LOG_BUF)
78 #define FP_BUF_LEVEL_6      (FP_LEVEL_6 | FC_TRACE_LOG_BUF)
79 #define FP_BUF_LEVEL_7      (FP_LEVEL_7 | FC_TRACE_LOG_BUF)
80 #define FP_BUF_LEVEL_8      (FP_LEVEL_8 | FC_TRACE_LOG_BUF)
81 #define FP_BUF_LEVEL_9      (FP_LEVEL_9 | FC_TRACE_LOG_BUF)
82
83
84 /*
85  * Log contents to both system messages file and trace buffer
86  */
87 #define FP_MSG_BUF_LEVEL_1  (FP_LEVEL_1 | FC_TRACE_LOG_BUF | \
88                             FC_TRACE_LOG_MSG)
89 #define FP_MSG_BUF_LEVEL_2  (FP_LEVEL_2 | FC_TRACE_LOG_BUF | \
90                             FC_TRACE_LOG_MSG)
91 #define FP_MSG_BUF_LEVEL_3  (FP_LEVEL_3 | FC_TRACE_LOG_BUF | \
92                             FC_TRACE_LOG_MSG)
93 #define FP_MSG_BUF_LEVEL_4  (FP_LEVEL_4 | FC_TRACE_LOG_BUF | \
94                             FC_TRACE_LOG_MSG)
95 #define FP_MSG_BUF_LEVEL_5  (FP_LEVEL_5 | FC_TRACE_LOG_BUF | \
96                             FC_TRACE_LOG_MSG)
97 #define FP_MSG_BUF_LEVEL_6  (FP_LEVEL_6 | FC_TRACE_LOG_BUF | \
98                             FC_TRACE_LOG_MSG)
99 #define FP_MSG_BUF_LEVEL_7  (FP_LEVEL_7 | FC_TRACE_LOG_BUF | \
100                             FC_TRACE_LOG_MSG)
101 #define FP_MSG_BUF_LEVEL_8  (FP_LEVEL_8 | FC_TRACE_LOG_BUF | \
102                             FC_TRACE_LOG_MSG)
103 #define FP_MSG_BUF_LEVEL_9  (FP_LEVEL_9 | FC_TRACE_LOG_BUF | \
104                             FC_TRACE_LOG_MSG)
105
106 /*
107  * Log contents to system messages file, console and trace buffer
108  */
109 #define FP_MSG_BUF_CONSOLE_LEVEL_1 (FP_LEVEL_1 | FC_TRACE_LOG_BUF | \
110                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
111 #define FP_MSG_BUF_CONSOLE_LEVEL_2 (FP_LEVEL_2 | FC_TRACE_LOG_BUF | \
112                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
113 #define FP_MSG_BUF_CONSOLE_LEVEL_3 (FP_LEVEL_3 | FC_TRACE_LOG_BUF | \
114                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
115 #define FP_MSG_BUF_CONSOLE_LEVEL_4 (FP_LEVEL_4 | FC_TRACE_LOG_BUF | \
116                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
117 #define FP_MSG_BUF_CONSOLE_LEVEL_5 (FP_LEVEL_5 | FC_TRACE_LOG_BUF | \
118                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
119 #define FP_MSG_BUF_CONSOLE_LEVEL_6 (FP_LEVEL_6 | FC_TRACE_LOG_BUF | \
120                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
121 #define FP_MSG_BUF_CONSOLE_LEVEL_7 (FP_LEVEL_7 | FC_TRACE_LOG_BUF | \
122                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
123 #define FP_MSG_BUF_CONSOLE_LEVEL_8 (FP_LEVEL_8 | FC_TRACE_LOG_BUF | \
124                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
125 #define FP_MSG_BUF_CONSOLE_LEVEL_9 (FP_LEVEL_9 | FC_TRACE_LOG_BUF | \
126                                     FC_TRACE_LOG_MSG | FC_TRACE_LOG_CONSOLE)
127 #ifndef DEBUG
```

```

129 #define FP_DTRACE                fc_trace_debug
131 #else
133 #define FP_DTRACE(...)            (void)(0)
129 #define FP_DTRACE
135 #endif
137 #define FP_TRACE                  fc_trace_debug
140 #ifdef  DEBUG
142 #define FP_TRACE_DEFAULT          (FC_TRACE_LOG_MASK | FP_LEVEL_1 |\
143 FP_LEVEL_2 | FP_LEVEL_3 |\
144 FP_LEVEL_4 | FP_LEVEL_5 |\
145 FP_LEVEL_6 | FP_LEVEL_7 | FP_LEVEL_9)
147 #else
149 #define FP_TRACE_DEFAULT          (FC_TRACE_LOG_MASK | FP_LEVEL_1 |\
150 FP_LEVEL_2 | FP_LEVEL_3 |\
151 FP_LEVEL_4 | FP_LEVEL_5 |\
152 FP_LEVEL_6 | FP_LEVEL_7 | FP_LEVEL_9)
153 #endif
155 #define FP_THREAD(x, y, z)        fp_logq, x->fp_ibuf, fp_trace, y, z
157 #define FP_NHEAD1(x, y)          FP_THREAD(port, FP_BUF_LEVEL_##x, y)
159 #define FP_NHEAD2(x, y)          FP_THREAD(port, FP_MSG_BUF_LEVEL_##x, y)
161 #define FP_NHEAD3(x, y)          FP_THREAD(port, FP_MSG_BUF_CONSOLE_LEVEL_##x, y)
164 /* This is used in about a dozen or so places in fp.c */
165 #define FP_IS_PKT_ERROR(pkt)      (((pkt)->pkt_state != FC_PKT_SUCCESS) ||\
166 ((pkt)->pkt_state == FC_PKT_SUCCESS &&\
167 (pkt)->pkt_resp_resid != 0))
170 /*
171 * This is only used in fp_ns_init() and fp_fabric_online().
172 */
173 #define FP_MAX_DEVICES            255
176 /*
177 * Software restoration bit fields while doing (PM)SUSPEND/(PM)RESUME
178 * Used with the fp_restore field in the fc_local_port_t struct.
179 */
180 #define FP_RESTORE_WAIT_TIMEOUT   0x01
181 #define FP_RESTORE_OFFLINE_TIMEOUT 0x02
182 #define FP_ELS_TIMEOUT            (20)
183 #define FP_NS_TIMEOUT            (120)
184 #define FP_IS_F_PORT(p)          ((p) & 0x1000)
185 #define FP_RETRY_COUNT           (5)
186 #define FP_RETRY_DELAY           (3)                /* E_D_TOV + 1 second */
187 #define FP_OFFLINE_TICKER        (90)             /* seconds */
188 #define FP_DEFAULT_SID           (0x0000AE)      /* Used once */
189 #define FP_DEFAULT_DID           (0x0000EA)      /* Used once */
190 #define FP_PORT_IDENTIFIER_LEN   (4)
191 #define FP_UNSQL_BUF_COUNT       (20)
192 #define FP_UNSQL_BUF_SIZE        (sizeof (la_els_logi_t))

```

```

193 #define FP_CMDWAIT_DELAY         (240) /* Enough time for all cmds to complt */
196 /*
197 * Values and macros used with fp_task and fp_last_task fields in
198 * the fc_local_port_t struct. Also see fp_job_handler() for more info.
199 */
200 #define FP_TASK_IDLE              0
201 #define FP_TASK_PORT_STARTUP     1
202 #define FP_TASK_OFFLINE         2
203 #define FP_TASK_ONLINE          3
204 #define FP_TASK_GETMAP           4
207 /*
208 * cmd_flags
209 */
210 #define FP_CMD_CFLAG_UNDEFINED   (-1)
211 #define FP_CMD_PLOGI_DONT_CARE   0x00
212 #define FP_CMD_PLOGI_RETAIN     0x01 /* Retain LOGIN */
213 #define FP_CMD_DELDEV_ON_ERROR  0x02 /* Remove device on error */
215 /*
216 * cmd_dflags
217 */
218 #define FP_CMD_VALID_DMA_MEM     0x01
219 #define FP_CMD_VALID_DMA_BIND   0x02
220 #define FP_RESP_VALID_DMA_MEM   0x04
221 #define FP_RESP_VALID_DMA_BIND  0x08
224 /* Values for fp_flag field in the fc_local_port_t struct */
225 #define FP_IDLE                  0x00
226 #define FP_OPEN                  0x01
227 #define FP_EXCL                  0x02
228 #define FP_EXCL_BUSY             0x04 /* Exclusive operation in progress */
231 /* message block/unblock'ing */
232 #define FP_WARNING_MESSAGES      0x01
233 #define FP_FATAL_MESSAGES        0x02
236 #define FP_IS_CLASS_1_OR_2(x)    \
237     ((x) == FC_TRAN_CLASS1 || (x) == FC_TRAN_CLASS2)
240 /*
241 * Driver message control
242 */
243 typedef enum fp_msg_dest {
244     FP_CONSOLE_ONLY,
245     FP_LOG_ONLY,
246     FP_LOG_AND_CONSOLE
247 } fp_msg_dest_t;

```

unchanged portion omitted

new/usr/src/uts/common/sys/scsi/targets/sddef.h

1

```
*****
76343 Wed May 1 07:06:45 2019
new/usr/src/uts/common/sys/scsi/targets/sddef.h
10886 smatch debug macro cleanup in usr/src/uts
*****
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://www.opensolaris.org/os/licensing.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed 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 */
24 /*
25 * Copyright 2011 cyril.galibern@opensvc.com
26 * Copyright 2017 Nexenta Systems, Inc. All rights reserved.
27 * Copyright 2019 Joyent, Inc.
28 */

30 #ifndef _SYS_SCSI_TARGETS_SDDEF_H
31 #define _SYS_SCSI_TARGETS_SDDEF_H

33 #include <sys/dktp/fdisk.h>
34 #include <sys/note.h>
35 #include <sys/mhd.h>
36 #include <sys/cmlb.h>

38 #ifdef __cplusplus
39 extern "C" {
40 #endif

43 #if defined(_KERNEL) || defined(_KMEMUSER)

46 #define SD_SUCCESS 0
47 #define SD_FAILURE (-1)

49 #if defined(TRUE)
50 #undef TRUE
51 #endif

53 #if defined(FALSE)
54 #undef FALSE
55 #endif

57 #define TRUE 1
58 #define FALSE 0

60 #if defined(VERBOSE)
61 #undef VERBOSE
```

new/usr/src/uts/common/sys/scsi/targets/sddef.h

2

```
62 #endif

64 #if defined(SILENT)
65 #undef SILENT
66 #endif

69 /*
70 * Fault Injection Flag for Inclusion of Code
71 *
72 * This should only be defined when SDDEBUG is defined
73 * #if DEBUG || lint
74 * #define SD_FAULT_INJECTION
75 * #endif
76 */

78 #if DEBUG || lint
79 #define SD_FAULT_INJECTION
80 #endif
81 #define VERBOSE 1
82 #define SILENT 0

84 /*
85 * Structures for recording whether a device is fully open or closed.
86 * Assumptions:
87 *
88 * + There are only 8 (sparc) or 16 (x86) disk slices possible.
89 * + BLK, MNT, CHR, SWP don't change in some future release!
90 */

92 #if defined(_SUNOS_VTOC_8)

94 #define SDUNIT_SHIFT 3
95 #define SDPART_MASK 7
96 #define NSDMAP NDKMAP

98 #elif defined(_SUNOS_VTOC_16)

100 /*
101 * XXX - NSDMAP has multiple definitions, one more in cmlb_impl.h
102 * If they are coalesced into one, this definition will follow suit.
103 * FDISK partitions - 4 primary and MAX_EXT_PARTS number of Extended
104 * Partitions.
105 */
106 #define FDISK_PARTS (FD_NUMPART + MAX_EXT_PARTS)

108 #define SDUNIT_SHIFT 6
109 #define SDPART_MASK 63
110 #define NSDMAP (NDKMAP + FDISK_PARTS + 1)

112 #else
113 #error "No VTOC format defined."
114 #endif

117 #define SDUNIT(dev) (getminor((dev)) >> SDUNIT_SHIFT)
118 #define SDPART(dev) (getminor((dev)) & SDPART_MASK)

120 /*
121 * maximum number of partitions the driver keeps track of; with
122 * EFI this can be larger than the number of partitions accessible
123 * through the minor nodes. It won't be used for keeping track
124 * of open counts, partition kstats, etc.
125 */
126 #define MAXPART (NSDMAP + 1)
```

```

128 /*
129 * Macro to retrieve the DDI instance number from the given buf struct.
130 * The instance number is encoded in the minor device number.
131 */
132 #define SD_GET_INSTANCE_FROM_BUF(bp)          \
133     (getminor((bp)->b_edev) >> SDUNIT_SHIFT)

137 struct ocinfo {
138     /*
139     * Types BLK, MNT, CHR, SWP,
140     * assumed to be types 0-3.
141     */
142     uint64_t lyr_open[NSDMAP];
143     uint64_t reg_open[OTYPCNT - 1];
144 };
145
146 unchanged portion omitted
147
1140 /*
1141 * Conditional set def
1142 */
1143 #define SD_CONDSET(a, b, c, d)              \
1144     { \
1145     a->c = ((fi_ ## b)->c); \
1146     SD_INFO(SD_LOG_IOERR, un, \
1147             "sd_fault_injection:" \
1148             "setting %s to %d\n", \
1149             d, ((fi_ ## b)->c)); \
1150     }

1152 /* SD FaultInjection ioctls */
1153 #define SDIOC          ('T'<<8)
1154 #define SDIOCSTART    (SDIOC|1)
1155 #define SDIOCSTOP     (SDIOC|2)
1156 #define SDIOCINSERTPKT (SDIOC|3)
1157 #define SDIOCINSERTXB (SDIOC|4)
1158 #define SDIOCINSERTTUN (SDIOC|5)
1159 #define SDIOCINSERTARQ (SDIOC|6)
1160 #define SDIOCPUSH     (SDIOC|7)
1161 #define SDIOCRETRIEVE (SDIOC|8)
1162 #define SDIOCRUN      (SDIOC|9)
1163 #endif

1165 #else

1167 #undef SDDEBUG
1168 #define SD_ERROR(...) (void)(0)
1169 #define SD_TRACE(...) (void)(0)
1170 #define SD_INFO(...) (void)(0)
1171 #define SD_DUMP_MEMORY(...) (void)(0)
1172 #define SD_ERROR { if (0) sd_log_err; }
1173 #define SD_TRACE { if (0) sd_log_trace; }
1174 #define SD_INFO { if (0) sd_log_info; }
1175 #define SD_DUMP_MEMORY { if (0) sd_dump_memory; }
1176 #endif

1177 /*
1178 * Miscellaneous macros
1179 */

1179 #define SD_USECTOHZ(x) (drv_usectohz((x)*1000000))
1180 #define SD_GET_PKT_STATUS(pkt) ((*pkt)->pkt_scbp) & STATUS_MASK

1182 #define SD_BIOERROR(bp, errcode) \

```

```

1183     if ((bp)->b_resid == 0) { \
1184         (bp)->b_resid = (bp)->b_bcount; \
1185     } \
1186     if ((bp)->b_error == 0) { \
1187         bioerror(bp, errcode); \
1188     } \
1189     (bp)->b_flags |= B_ERROR; \

1191 #define SD_FILL_SCSI1_LUN_CDB(lump, cdbp) \
1192     if (! (lump)->un_f_is_fibre && \
1193         SD_INQUIRY((lump))->inq_ansi == 0x01) { \
1194         int _lun = ddi_prop_get_int(DDI_DEV_T_ANY, \
1195             SD_DEVINFO((lump)), DDI_PROP_DONTPASS, \
1196             SCSI_ADDR_PROP_LUN, 0); \
1197         if (_lun > 0) { \
1198             (cdbp)->scc_lun = _lun; \
1199         } \
1200     }

1202 #define SD_FILL_SCSI1_LUN(lump, pkt) \
1203     SD_FILL_SCSI1_LUN_CDB((lump), (union scsi_cdb *) (pkt)->pkt_cdbp)

1205 /*
1206 * Disk driver states
1207 */

1209 #define SD_STATE_NORMAL          0
1210 #define SD_STATE_OFFLINE        1
1211 #define SD_STATE_RWAIT          2
1212 #define SD_STATE_DUMPING        3
1213 #define SD_STATE_SUSPENDED      4
1214 #define SD_STATE_PM_CHANGING    5

1216 /*
1217 * The table is to be interpreted as follows: The rows lists all the states
1218 * and each column is a state that a state in each row *can* reach. The entries
1219 * in the table list the event that cause that transition to take place.
1220 * For e.g.: To go from state RWAIT to SUSPENDED, event (d)-- which is the
1221 * invocation of DDI_SUSPEND-- has to take place. Note the same event could
1222 * cause the transition from one state to two different states. e.g., from
1223 * state SUSPENDED, when we get a DDI_RESUME, we just go back to the *last
1224 * state* whatever that might be. (NORMAL or OFFLINE).
1225 *
1226 *
1227 * State Transition Table:
1228 *
1229 *          NORMAL  OFFLINE  RWAIT  DUMPING  SUSPENDED  PM_SUSPENDED
1230 *
1231 * NORMAL      -      (a)      (b)      (c)      (d)      (h)
1232 *
1233 * OFFLINE     (e)      -      (e)      (c)      (d)      NP
1234 *
1235 * RWAIT       (f)      NP      -      (c)      (d)      (h)
1236 *
1237 * DUMPING     NP      NP      NP      -      NP      NP
1238 *
1239 * SUSPENDED   (g)      (g)      (b)      NP*     -      NP
1240 *
1241 * PM_SUSPENDED (i)      NP      (b)      (c)      (d)      -
1242 *
1243 * NP :        Not Possible.
1244 * (a):        Disk does not respond.
1245 * (b):        Packet Allocation Fails
1246 * (c):        Panic - Crash dump
1247 * (d):        DDI_SUSPEND is called.
1248 * (e):        Disk has a successful I/O completed.

```

```

1249 * (f):      sdrunout() calls sdstart() which sets it NORMAL
1250 * (g):      DDI_RESUME is called.
1251 * (h):      Device threshold exceeded pm framework called power
1252 *           entry point or pm_lower_power called in detach.
1253 * (i):      When new I/O come in.
1254 * * :      When suspended, we dont change state during panic dump
1255 */

1258 #define SD_MAX_THROTTLE      256
1259 #define SD_MIN_THROTTLE      8
1260 /*
1261 * Lowest valid max. and min. throttle value.
1262 * This is set to 2 because if un_min_throttle were allowed to be 1 then
1263 * un_throttle would never get set to a value less than un_min_throttle
1264 * (0 is a special case) which means it would never get set back to
1265 * un_saved_throttle in routine sd_restore_throttle().
1266 */
1267 #define SD_LOWEST_VALID_THROTTLE      2

1271 /* Return codes for sd_send_polled_cmd() and sd_scsi_poll() */
1272 #define SD_CMD_SUCCESS      0
1273 #define SD_CMD_FAILURE      1
1274 #define SD_CMD_RESERVATION_CONFLICT      2
1275 #define SD_CMD_ILLEGAL_REQUEST      3
1276 #define SD_CMD_BECOMING_READY      4
1277 #define SD_CMD_CHECK_CONDITION      5

1279 /* Return codes for sd_ready_and_valid */
1280 #define SD_READY_VALID      0
1281 #define SD_NOT_READY_VALID      1
1282 #define SD_RESERVED_BY_OTHERS      2

1284 #define SD_PATH_STANDARD      0
1285 #define SD_PATH_DIRECT      1
1286 #define SD_PATH_DIRECT_PRIORITY      2

1288 #define SD_UNIT_ATTENTION_RETRY      40

1290 /*
1291 * The following three are bit flags passed into sd_send_scsi_TEST_UNIT_READY
1292 * to control specific behavior.
1293 */
1294 #define SD_CHECK_FOR_MEDIA      0x01
1295 #define SD_DONT_RETRY_TUR      0x02
1296 #define SD_BYPASS_PM      0x04

1298 #define SD_GROUP0_MAX_ADDRESS      (0x1fffff)
1299 #define SD_GROUP0_MAXCOUNT      (0xff)
1300 #define SD_GROUP1_MAX_ADDRESS      (0xfffffff)
1301 #define SD_GROUP1_MAXCOUNT      (0xffff)

1303 #define SD_BECOMING_ACTIVE      0x01
1304 #define SD_REMOVAL_ALLOW      0
1305 #define SD_REMOVAL_PREVENT      1

1307 #define SD_GET_PKT_OPCODE(pkt) \
1308      ((union scsi_cdb *)((pkt)->pkt_cdbp))->cdb_un.cmd

1311 #define SD_NO_RETRY_ISSUED      0
1312 #define SD_DELAYED_RETRY_ISSUED      1
1313 #define SD_IMMEDIATE_RETRY_ISSUED      2

```

```

1315 #if defined(__i386) || defined(__amd64)
1316 #define SD_UPDATE_B_RESID(bp, pkt) \
1317      ((bp)->b_resid += (pkt)->pkt_resid + (SD_GET_XBUF(bp)->xb_dma_resid))
1318 #else
1319 #define SD_UPDATE_B_RESID(bp, pkt) \
1320      ((bp)->b_resid += (pkt)->pkt_resid)
1321 #endif

1324 #define SD_RETRIES_MASK      0x00FF
1325 #define SD_RETRIES_NOCHECK      0x0000
1326 #define SD_RETRIES_STANDARD      0x0001
1327 #define SD_RETRIES_VICTIM      0x0002
1328 #define SD_RETRIES_BUSY      0x0003
1329 #define SD_RETRIES_UA      0x0004
1330 #define SD_RETRIES_ISOLATE      0x8000
1331 #define SD_RETRIES_FAILFAST      0x4000

1333 #define SD_UPDATE_RESERVATION_STATUS(un, pkt) \
1334 if (((pkt)->pkt_reason == CMD_RESET) || \
1335      ((pkt)->pkt_statistics & (STAT_BUS_RESET | STAT_DEV_RESET))) { \
1336      if (((un)->un_resvd_status & SD_RESERVE) == SD_RESERVE) { \
1337          (un)->un_resvd_status |= \
1338              (SD_LOST_RESERVE | SD_WANT_RESERVE); \
1339      } \
1340 }
_____unchanged_portion_omitted_____

```