    1 /*
    2 * CDDL HEADER START
    3 *
    4 * The contents of this file are subject to the terms of the
    5 * Common Development and Distribution License, Version 1.0 only
    6 * (the "License"). You may not use this file except in compliance
    7 * with the License.
    8 *
    9 * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
   10 * or http://www.opensolaris.org/os/licensing.
   11 * See the License for the specific language governing permissions
   12 * and limitations under the License.
   13 *
   14 * When distributing Covered Code, include this CDDL HEADER in each
   15 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
   16 * If applicable, add the following below this CDDL HEADER, with the
   17 * fields enclosed by brackets "[]" replaced with your own identifying
   18 * information: Portions Copyright [yyyy] [name of copyright owner]
   19 *
   20 * CDDL HEADER END
   21 */
   22 /*
   23 * Copyright 2014 Garrett D'Amore <garrett@damore.org>
   24 *
   25 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
   26 * Use is subject to license terms.
   27 */

   29 /*   Copyright (c) 1986 AT&T */
   30 /*   All Rights Reserved   */

   33 /*   This module is created for NLS on Jun.04.86   */

   35 #ifndef _WIDEC_H
   36 #define _WIDEC_H

   38 #include <sys/feature_tests.h>

   40 #include <stdio.h>   /* For definition of FILE */
   41 #include <euc.h>
   42 #include <wchar.h>

   44 #ifdef __cplusplus
   45 extern "C" {
   46 #endif

   48 /* Character based input and output functions */
   49 extern wchar_t *getws(wchar_t *);
   50 extern int   putws(const wchar_t *);

   52 #if !defined(__lint)
   53 #define getwc(p)   fgetwc(p)
   54 #define putwc(x, p)   fputwc((x), (p))
   55 #define getwchar()   getwc(stdin)
   56 #define putwchar(x)   putwc((x), stdout)
   57 #endif

   59 /* wchar_t string operation functions */
   60 extern wchar_t *strtows(wchar_t *, char *);
   61 extern wchar_t *wscpy(wchar_t *, const wchar_t *);

   62 extern wchar_t *wsncpy(wchar_t *, const wchar_t *, size_t);
   63 extern wchar_t *wscat(wchar_t *, const wchar_t *);
   64 extern wchar_t *wsncat(wchar_t *, const wchar_t *, size_t);
   65 extern wchar_t *wschr(const wchar_t *, wchar_t);
   66 extern wchar_t *wsrchr(const wchar_t *, wchar_t);
   67 extern wchar_t *wspbrk(const wchar_t *, const wchar_t *);
   68 extern wchar_t *wstok(wchar_t *, const wchar_t *);
   69 extern char   *wstostr(char *, wchar_t *);

   71 extern int   wscmp(const wchar_t *, const wchar_t *);
   72 extern int   wsncmp(const wchar_t *, const wchar_t *, size_t);
   73 extern size_t   wslen(const wchar_t *);
   74 extern size_t   wsspn(const wchar_t *, const wchar_t *);
   75 extern size_t   wscspn(const wchar_t *, const wchar_t *);
   76 extern int   wscoll(const wchar_t *, const wchar_t *);
   77 extern size_t   wsxfrm(wchar_t *, const wchar_t *, size_t);

   79 **#if !defined(_STRICT_SYMBOLS)**
   79 *#if !defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX)*

   81 extern wchar_t *wsdup(const wchar_t *);
   82 extern int   wscol(const wchar_t *);
   83 extern double   wstod(const wchar_t *, wchar_t **);
   84 extern long   wstol(const wchar_t *, wchar_t **, int);
   85 extern int   wscasecmp(const wchar_t *, const wchar_t *);
   86 extern int   wsncasecmp(const wchar_t *, const wchar_t *, size_t);
   87 extern int   wsprintf(wchar_t *, const char *, ...);
   88 #if defined(_LONGLONG_TYPE)
   89 extern long long   wstoll(const wchar_t *, wchar_t **, int);
   90 #endif /* defined(_LONGLONG_TYPE) */

   92 **#endif /* !defined(_STRICT_SYMBOLS) */**
   92 *#endif /* !defined(_STRICT_STDC) && !defined(__XOPEN_OR_POSIX) */*

   94 /* Returns the code set number for the process code c. */
   95 #define WCHAR_SHIFT   7
   96 #define WCHAR_S_MASK   0x7f
   97 #define wcsetno(c) \
   98   (((c)&0x20000000)?(((c)&0x10000000)?1:3):(((c)&0x10000000)?2:0))

  100 /* Aliases... */
  101 #define windex   wschr
  102 #define wrindex   wsrchr

  104 #define watol(s)   wstol((s), (wchar_t **)0, 10)
  105 #if defined(_LONGLONG_TYPE) && !defined(__lint)
  106 #define watoll(s)   wstoll((s), (wchar_t **)0, 10)
  107 #endif /* defined(_LONGLONG_TYPE) && !defined(__lint) */
  108 #define watoi(s)   ((int)wstol((s), (wchar_t **)0, 10))
  109 #define watof(s)   wstod((s), (wchar_t **)0)

  111 /*
  112 * other macros.
  113 */
  114 #define WCHAR_CSMASK   0x30000000
  115 #define EUCMASK   0x30000000
  116 #define WCHAR_CS0   0x00000000
  117 #define WCHAR_CS1   0x30000000
  118 #define WCHAR_CS2   0x10000000
  119 #define WCHAR_CS3   0x20000000
  120 #define WCHAR_BYTE_OF(wc, i) (((wc&~0x30000000)>>(7*(3-i)))&0x7f)

  122 #ifdef __cplusplus
  123 }
_____**unchanged_portion_omitted_**