

## 第17章 Shift-JIS支持

---

最通用的日文编码方法是 Shift-JIS。Shift-JIS 编码使 8 位字符得到丰富的使用，第一个字节的值用来区别单字节和多字节字符。

你不需要调用特殊函数来显示一个 Shift-JIS 字符串。最主要的必要条件是要有一种包含有用于显示的 Shift-JIS 字符的字体。

## 17.1 创建 Shift-JIS 字体

字体转换器能够从任何的 Windows 字体产生一种 Shift-JIS 字体用于  $\mu$ C/GUI。当使用一种 Shift-JIS 字体时，用于显示显示 Shift-JIS 字符的函数会自动地与库链接。

关于如何创建 Shift-JIS 字体的详细信息，请联系 Micrium 公司 (info@micrium.com)。一份单独的字体转换器的技术资料描述了所有你需要在你的  $\mu$ C/GUI 工程中实现 Shift-JIS 的有效方法。

## 17.2 例子

下面的例子定义了一种包括 6 个字符的小字体：“A”，“B”，“C” 和 Shift-JIS 字符 0x8350（片假名字母 KE），0x8351（片假名字母 GE）和 0x8352（片假名字母 KO）。然后在显示屏上绘出一个复合字符串。该例子的源代码是 Sample\Misc\ShiftJIS.c。

```

/*-----
文件:      ShiftJIS.c
目的:      展示  $\mu$ C/GUI 的 ShiftJIS 性能
-----*/

#include "GUI.H"

/*-----
*                      ShiftJIS 字体的定义                      *
*-----*/

/* 拉丁文大写字母 A */
static const unsigned char acFontSJIS13_0041[13] = { /* 编码 0041 */
    _____,
    _____,
    __X_____,
    __X_____,
    __X_X_____,
    __X_X_____,
    __X_X_____,
    _XXXXX_____,
    _X__X_____,
    _X__X_____,
    XXX_XXX_,
    _____,
    _____};

```

```
/* 拉丁文大写字母 B */
```

```
static const unsigned char acFontSJIS13_0042[13] = { /* 编码 0042 */
    _____,
    _____,
    XXXXXX____,
    _X__X__,
    _X__X__,
    _X__X__,
    _XXXX____,
    _X__X__,
    _X__X__,
    _X__X__,
    XXXXXX____,
    _____,
    _____};
```

```
/* 拉丁文大写字母 C */
```

```
static const unsigned char acFontSJIS13_0043[13] = { /* 编码 0043 */
    _____,
    _____,
    __XX_X__,
    _X__XX__,
    X__X__,
    X_____,
    X_____,
    X_____,
    X_____,
    _X__X__,
    _XXX____,
    _____,
    _____};
```

```
/* 片假名字母 KE */
```

```
static const unsigned char acFontSJIS13_8350[26] = { /* 编码 8350 */
    __XX____, _____,
    __X____, _____,
    __X____, _____,
    _____, XXXXX____,
    __X__X__, _____,
    __X__X__, _____,
    X__X__, _____,
    _____X__, _____,
```

```

_____X_, _____,
_____X_, _____,
_____X_, _____,
____XX_____, _____,
_____ , _____ } ;

```

*/\* 片假名字母 GE \*/*

```

static const unsigned char acFontSJIS13_8351[26] = { /* 编码 8351 */
____XX_____, X_X_____,
_____X_____, _X_X_____,
_____X_____, _____,
_____XXXXXXX, XXX_____,
_____X_____X, _____,
_____X_____X, _____,
X_____X, _____,
_____X_____, _____,
_____X_____, _____,
_____X_____, _____,
_____X_____, _____,
____XX_____, _____,
_____ , _____ } ;

```

*/\* 片假名字母 KO \*/*

```

static const unsigned char acFontSJIS13_8352[26] = { /* 编码 8352 */
_____, _____,
_____, _____,
_____XXXXXXX, XX_____,
_____, _X_____,
_____, _X_____,
_____, _X_____,
_____, _X_____,
_____, _X_____,
_____, _X_____,
_____XXXXXXX, XXXX_____,
_____, _____,
_____, _____,
_____ , _____ } ;

```

```

static const GUI_CHARINFO GUI_FontSJIS13_CharInfo[6] = {
    {7, 7, 1, (void *)&acFontSJIS13_0041} /* 编码 0041 */
    , { 7, 7, 1, (void *)&acFontSJIS13_0042} /* 编码 0042 */
    , { 7, 7, 1, (void *)&acFontSJIS13_0043} /* 编码 0043 */

```

```

    , { 14, 14, 2, (void *)&acFontSJIS13_8350}      /* 编码 8350 */
    , { 14, 14, 2, (void *)&acFontSJIS13_8351}      /* 编码 8351 */
    , { 14, 14, 2, (void *)&acFontSJIS13_8352}      /* 编码 8352 */
};

static const GUI_FONT_PROP GUI_FontSJIS13_Prop2 = {
    0x8350          /* 第一个字符 */
    , 0x8352        /* 最后一个字符 */
    , &GUI_FontSJIS13_CharInfo[3] /* 第一个字符的地址 */
    , (void *)0     /* 指向下一个 GUI_FONT_PROP 的指针 */
};

static const GUI_FONT_PROP GUI_FontSJIS13_Prop1 = {
    0x0041          /* 第一个字符 */
    , 0x0043        /* 最后一个字符 */
    , &GUI_FontSJIS13_CharInfo[ 0] /* 第一个字符的地址 */
    , (void *)&GUI_FontSJIS13_Prop2 /* 指向下一个 GUI_FONT_PROP 的指针 */
};

static const GUI_FONT GUI_FontSJIS13 = {
    GUI_FONTTYPE_PROP_SJIS /* 字体类型 */
    , 13                   /* 字体高度 */
    , 13                   /* 字体 Y 轴方向的间隔 */
    , 1                    /* X 轴方向放大系数 */
    , 1                    /* Y 轴方向放大系数 */
    , (void *)&GUI_FontSJIS13_Prop1
};

/*****
 *          包含 ASCII 字符和 ShiftJIS 字符的字符串的定义          *
*****/

static const char aSJIS[] = {
    "ABC\x83\x50\x83\x51\x83\x52\x0"
};

/*****
 *          展示 ShiftJIS 字符的输出          *
*****/

void DemoShiftJIS(void)

```

```
{
    GUI_SetFont(&GUI_Font13HB_1);
    GUI_DispStringHCenterAt("µC/GUI-sample: ShiftJIS characters", 160, 0);
    /* 设置 ShiftJIS 字体 */
    GUI_SetFont(&GUI_FontSJIS13);
    /* 显示字符串 */
    GUI_DispStringHCenterAt(aSJIS, 160, 40);
}

/*****
*                                     *
******/

void main(void)
{
    GUI_Init();
    DemoShiftJIS();
    while(1)
        GUI_Delay(100);
}
```

范例程序运行结果的屏幕截图

**µC/GUI sample: UNICODE characters**

ABCだぢぢ