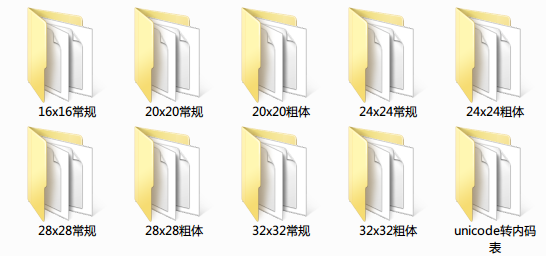
# AC54xx字库使用说明

### 添加字库

请先确定字号以及字形，再根据所选的多国语言选择相应的字库文件以及TAB文件拷贝到资源目录(res文件夹)。(注：若只需要显示内码，\*.TAB文件可不拷贝)



以下为各国语言所对应的字库文件以及Unicode转内码表

Chinese\_Simplified(简体中文):

F\_ASCII.PIX F\_GB2312.PIX F\_GB2312.TAB (GB2312字库)

F\_ASCII.PIX F\_GBK.PIX F\_GBK.TAB(GBK字库)

Chinese\_Traditional(繁体中文):

F\_ASCII.PIX F\_BIG5.PIX F\_BIG5.TAB

Japanese(日语):

F\_SJISASC.PIX F\_SJIS.PIX F\_SJIS.TAB

Korean(韩语):

F\_ASCII.PIX F\_KSC.PIX F\_KSC.TAB

English(英语):

F\_CP1252.PIX F\_CP1252.TAB

French(法语):

F\_CP1252.PIX F\_CP1252.TAB

German(德语):

F\_CP1252.PIX F\_CP1252.TAB

Italian(意大利语):

F\_CP1252.PIX F\_CP1252.TAB

Dutch(荷兰语):

F\_CP1252.PIX F\_CP1252.TAB

Portuguese(葡萄牙语):

F\_CP1252.PIX F\_CP1252.TAB

Spanish(西班牙语):

F\_CP1252.PIX F\_CP1252.TAB

Swedish(瑞典语):

F\_CP1252.PIX F\_CP1252.TAB

Czech(捷克语):

F\_CP1250.PIX F\_CP1250.TAB

Danish(丹麦语):

F\_CP1252.PIX F\_CP1252.TAB

Polish(波兰语):

F\_CP1250.PIX F\_CP1250.TAB

Russian(俄罗斯语):

F\_CP1251.PIX F\_CP1251.TAB

Turkey(土耳其语):

F\_CP1254.PIX F\_CP1254.TAB

Hebrew(希伯来语):

F\_CP1255.PIX F\_CP1255.TAB

Thai(泰语):

F\_CP874.PIX F\_CP874.TAB

Hungarian(匈牙利语):

F\_CP1250.PIX F\_CP1250.TAB

Romanian(罗马尼亚语):

F\_CP1250.PIX F\_CP1250.TAB

Arabic(阿拉伯语):

F\_CP1256.PIX F\_CP1256.TAB

### 2.创建文本控件

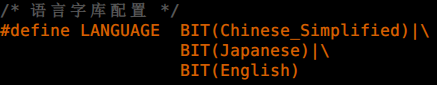
编码格式填**”text”**



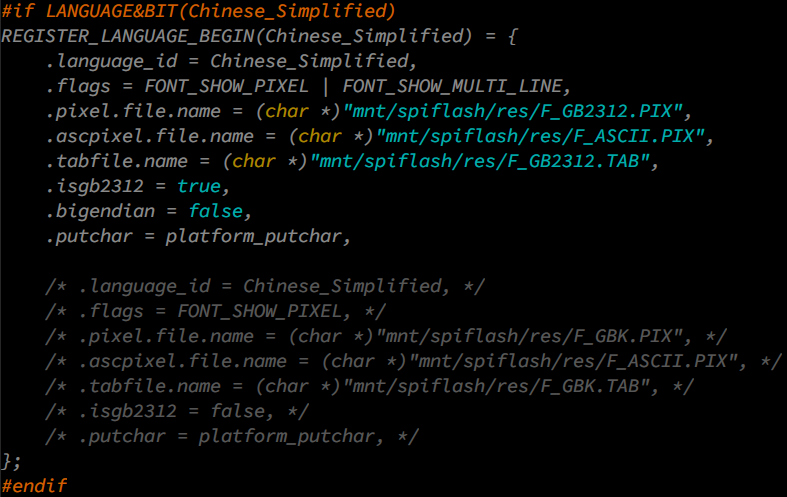
### 3.初始化字库

修改fontinit.c文件，配置需要的语言

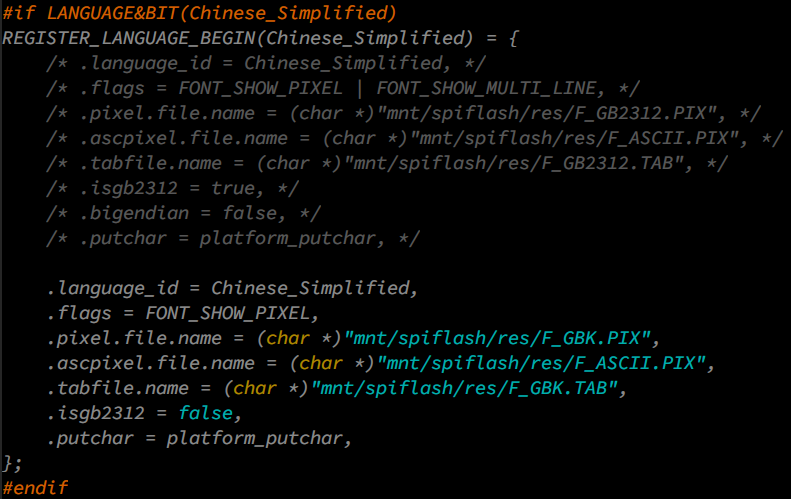
注:简体中文包含GB2312字库以及GBK字库,GB2312是GBK的精简版,适用于spiflash空间比较小的应用.



选择GB2312字库时的初始化



选择GBK字库时的初始化



### 4.添加文本控件回调函数

static int text\_txt\_test\_onchange(void \*ctr, enum element\_change\_event e, void \*arg)

{

struct draw\_context \*dc = (struct draw\_context \*)arg;

struct ui\_text \*text = (struct ui\_text \*)ctr;

const char \*ascii = "0123abcd\n";

/\*

注:此处字符串ascii不能直接输入中文,中文的编码由本文件的编码格式决定,如需显示中文,请使用数组的形式.

\*/

switch (e) {

case ON\_CHANGE\_INIT:

/\*此处添加控件默认显示文字,若无显示默认文字的需求可不加\*/

ui\_text\_set\_text\_attrs(text,ascii,strlen(ascii),FONT\_ENCODE\_UTF8,0,FONT\_DEFAULT);

break;

case ON\_CHANGE\_SHOW:

break;

default:

return false;

}

return false;

}

REGISTER\_UI\_EVENT\_HANDLER(TEXT\_TXT\_TEST)

.onchange = text\_txt\_test\_onchange,

.onkey = NULL,

.ontouch = NULL,

};

### 5.字库接口

/\*

函数功能：设置文本的属性,用于文本控件的初始化

输入参数:

Text : ui\_text结构体

str :编码

Strlen :编码长度

Encode :编码类型

Endian :大小端(只针对FONT\_ENCODE\_UNICODE)

flags :标志位

FONT\_DEFAULT //默认显示（单行显示）

FONT\_SHOW\_PIXEL //显示字符

FONT\_SHOW\_MULTI\_LINE //多行显示

返回值:无

\*/

void ui\_text\_set\_text\_attrs(struct ui\_text \*text,const char \*str, int strlen,u8 encode, u8 endian, u32 flags);

/\*

函数功能：内码显示函数

输入参数:

id ：文本控件ID号

str :内码

Strlen :内码长度

flags :标志位

FONT\_DEFAULT //默认显示（单行显示）

FONT\_SHOW\_PIXEL //显示字符

FONT\_SHOW\_MULTI\_LINE //多行显示

返回值:

0:失败

其他:己显示的编码长度

\*/

int ui\_text\_set\_text\_by\_id(int id, const char \*str, int strlen, u32 flags);

/\*

函数功能：unicode编码显示函数

输入参数:

id :文本控件ID号

str :unicode编码

strlen :unicode编码长度

endian :大小端

FONT\_ENDIAN\_BIG //大端

FONT\_ENDIAN\_SMALL //小端

flags :标志位

FONT\_DEFAULT //默认显示（单行显示）

FONT\_SHOW\_PIXEL //显示字符

FONT\_SHOW\_MULTI\_LINE //多行显示

返回值:

0:失败

其他:己显示的编码长度

\*/

int ui\_text\_set\_textw\_by\_id(int id, const char \*str, int strlen, int endian, u32 flags);

/\*

函数功能：utf8编码显示函数

输入参数:

id :文本控件ID号

str :utf8编码

strlen :utf8编码长度

flags :标志位

FONT\_DEFAULT //默认显示（单行显示）

FONT\_SHOW\_PIXEL //显示字符

FONT\_SHOW\_MULTI\_LINE //多行显示

返回值:

0:失败

其他:己显示的编码长度

\*/

int ui\_text\_set\_textu\_by\_id(int id, const char \*str, int strlen, u32 flags);

示例：

/\*内码,直接显示字符串\*/

ui\_text\_set\_text\_by\_id(TEXT\_TXT\_TEST, "abc", -1, FONT\_DEFAULT);

/\*内码,单行显示\*/

ui\_text\_set\_text\_by\_id(TEXT\_TXT\_TEST, ansicode, sizeof(ansicode), FONT\_DEFAULT);

/\*utf16大端编码，多行显示\*/

ui\_text\_set\_textw\_by\_id(TEXT\_TXT\_TEST, utf16\_be, sizeof(utf16\_be), FONT\_ENDIAN\_BIG, FONT\_DEFAULT | FONT\_SHOW\_MULTI\_LINE);

/\*utf16小端编码,单行显示\*/

ui\_text\_set\_textw\_by\_id(TEXT\_TXT\_TEST, utf16\_le, sizeof(utf16\_le), FONT\_ENDIAN\_SMALL, FONT\_DEFAULT);

/\*utf8编码,多行显示\*/

ui\_text\_set\_textu\_by\_id(TEXT\_TXT\_TEST,text\_utf8,sizeof(text\_utf8), FONT\_DEFAULT|FONT\_SHOW\_MULTI\_LINE);