

# GT32L24F0210 多国文字字库芯片

## 规格书 DATASHEET

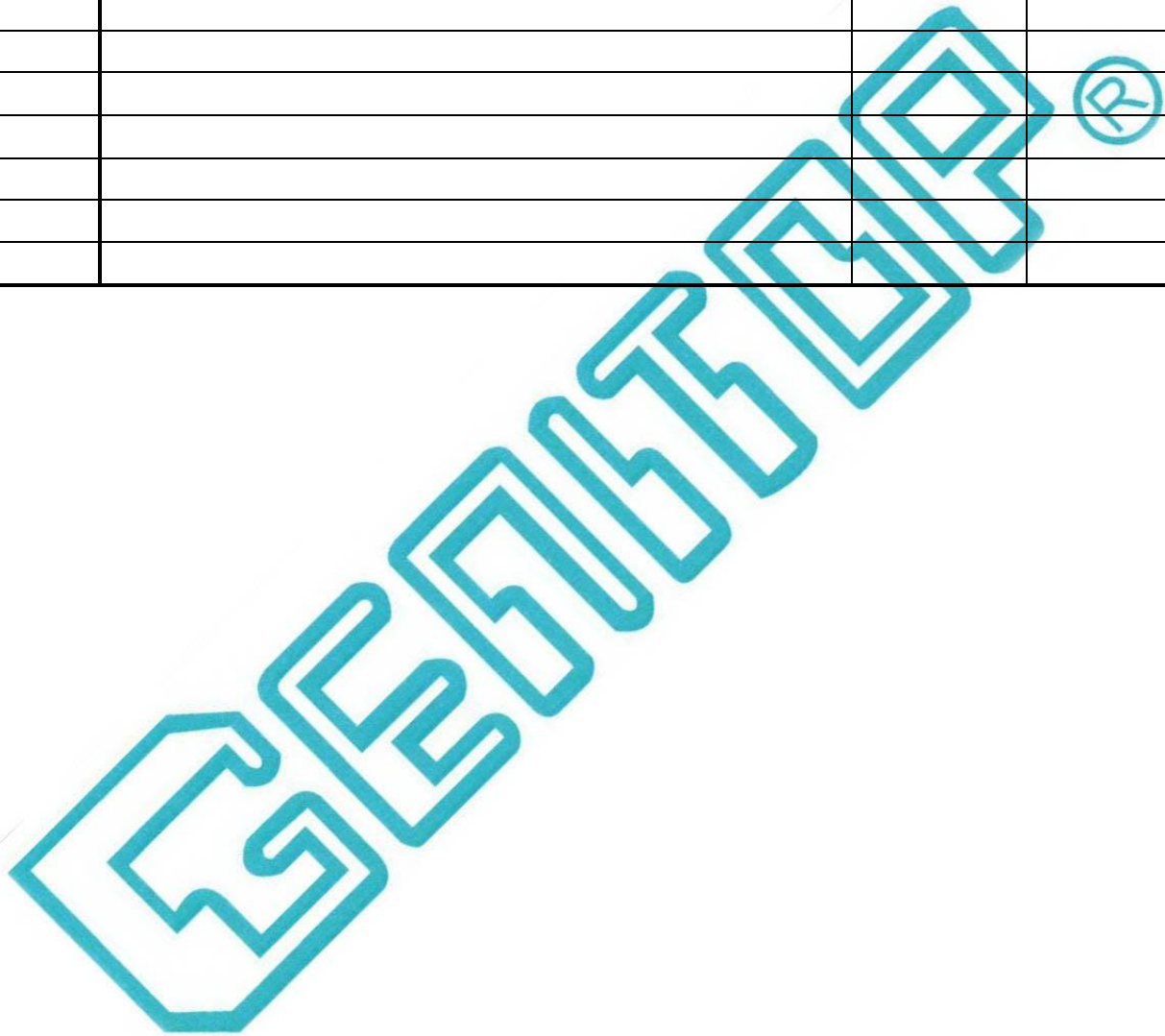
- CODE PAGE 字符集
- ISO8859 字符集
- ASCII 字符集
- LCM 字符集
- UNICODE 多国字符集
- 字号：5x7~24 点阵
- 排置方式：竖置横排及横置横排
- 总线接口：SPI 串行总线
- 封装类型：SOP8-A

V1.1\_I\_B

2013-03

## 版本修订记录

版本号	修改内容	日期	备注
V1.0I_A	字库芯片说明书的制定	2013-01	字库定制
V1.1I_A	增加 CP437 特殊字符算法及样张	2013-01	



## 目 录

<b>1 概述</b> .....	<b>4</b>
1.1 芯片特点.....	4
1.2 芯片内容.....	4
<b>2 操作指令</b> .....	<b>7</b>
2.1 Instruction Parameter(指令参数).....	7
2.2 Read Data Bytes (一般读取).....	7
2.3 Read Data Bytes at Higher Speed (快速读取点阵数据).....	7
<b>3 字符点阵字库地址表</b> .....	<b>9</b>
<b>4 字符点阵数据在芯片中的地址计算方法</b> .....	<b>13</b>
4.1 ASCII 字符的地址计算.....	13
4.2 UNICODE 字符的地址计算.....	15
4.3 ISO 字符的地址计算.....	21
4.4 LCM 字符的地址计算.....	27
4.5 CODE PAGE.....	28
4.6 其他字符.....	41
4.7 UTF-8 转 UTF-16 转换程序.....	46
4.8 5X7 字符 ISO8859 转 UNICODE 程序.....	47
<b>5 引脚描述与电路连接</b> .....	<b>48</b>
5.1 引脚配置.....	48
5.2 引脚描述.....	48
5.3 SPI 接口与主机接口参考电路示意图.....	50
<b>6 电气特性</b> .....	<b>51</b>
6.1 绝对最大额定值.....	51
6.2 DC 特性.....	51
6.3 AC 特性.....	51
<b>7 封装尺寸</b> .....	<b>53</b>
<b>8 附录</b> .....	<b>54</b>
8.1 字型样张.....	54
8.2 173 国外文字库总表.....	87
8.3 173 国外文拼音索引表(汉语拼音排序).....	91
8.4 173 国外文字库索引表(英文字母排序).....	93

# 1 概述

GT32L24F0210是一款支持多国文字的字库芯片，支持25套CODE PAGE、13套ASCII、多国外文UNICODE，同时支持14套ISO8859字符集，该字库支持多达173国文字。24点阵以下排列格式为竖置横排，24点阵的为横置横排。用户通过字符内码，利用本手册提供的方法计算出该字符点阵在芯片中的地址，可从该地址连续读出字符点阵信息。

## 1.1 芯片特点

- 数据总线：SPI 串行总线接口
- 点阵排列方式：竖置横排及横置横排
- 时钟频率：80MHz(max.) @3.3V
- 工作电压：2.7V~3.6V
- 电流：
  - 工作电流：12mA
  - 待机电流：10uA
- 工作温度：-40°C~85°C
- 封装：SOP8-A

## 1.2 芯片内容

分类	字库	字号	字符数	字体	排列方式	备注
ASCII	ASCII	5X7	96	标准	Y-竖置横排	
	ASCII	7X8	96	标准	Y-竖置横排	
	ASCII	6X12	96	标准	Y-竖置横排	
	ASCII	8X16	96	Arial	Y-竖置横排	
	ASCII	8X16	96	粗体	Y-竖置横排	
	ASCII	9X24	96	打印体	W-横置横排	
	ASCII	12点不等宽	96	Arial	Y-竖置横排	
	ASCII	12点不等宽	96	Times New Roman	Y-竖置横排	
	ASCII	16点不等宽	96	Arial	Y-竖置横排	
	ASCII	16点不等宽	96	Times New Roman	Y-竖置横排	
	ASCII	24点不等宽	96	Arial	W-横置横排	
	ASCII	32点不等宽	96	Arial	W-横置横排	
UNICODE	拉丁文	8X16	496	标准	Y-竖置横排	
		12X24	496	标准	W-横置横排	
		12X24	496	打印体	W-横置横排	
		12点不等宽	496	标准	Y-竖置横排	

		16 点不等宽	496	标准	Y-竖置横排
希腊文		8X16	96	标准	Y-竖置横排
		12X24	96	标准	W-横置横排
		12 点不等宽	96	标准	Y-竖置横排
		16 点不等宽	96	标准	Y-竖置横排
基里尔文		8X16	208	标准	Y-竖置横排
		12X24	208	标准	W-横置横排
		12X24	208	打印体	W-横置横排
		12 点不等宽	208	标准	Y-竖置横排
希伯来文		16 点不等宽	208	标准	Y-竖置横排
		8X16	112	标准	Y-竖置横排
泰文		12X24	112	黑体	W-横置横排
		8X16	128	标准	Y-竖置横排
日文假名		24 点不等宽	128	标准	W-横置横排
		8X16	64	标准	W-横置横排
阿拉伯文		12X24	64	黑体	W-横置横排
		16 点不等宽	576	标准	Y-竖置横排
		24 点不等宽	576	标准	W-横置横排
ISO8859	NO.1~NO.16 (Without No.6 and No.12)	5X7	128x14	标准	Y-竖置横排
	NO.1~NO.16 (Without No.6 and No.12)	8X16	128x14	标准	Y-竖置横排
LCM 字符	LCM	5X7	256	标准	Y-竖置横排
		5X10	256x7	标准	Y-竖置横排
	LCM 字符备用区- 1	5X10	256	标准	Y-竖置横排
	LCM 字符备用区- 2	5X10	256	标准	Y-竖置横排
CODE PAGE	437—USA, Standard Europe	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	737—Greek	12x24	256	标准	W-横置横排
	775—Baltic	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	850—Multilingual	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	852—Latin 2	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	855—Cyrillic	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	857—Turkish	12x24	256	标准	W-横置横排
			打印体	W-横置横排	
858—Euro	12x24	128	标准	W-横置横排	
			打印体	W-横置横排	
860—Portuguese	12x24	256	标准	W-横置横排	
			打印体	W-横置横排	
862—Hebrew	12x24	256	标准	W-横置横排	

	863—Canadian French	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	864—Arabic	24 点不等宽	256	标准	W-横置横排
	865—Nordic	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	866—Cyrillic 2	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	1251—Cyrillic	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	1252—Latin 1	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	1253—Greek	12x24	256	标准	W-横置横排
	1254—Turkish	12x24	256	标准	W-横置横排
				打印体	W-横置横排
	1255—Hebrew New	12x24	256	标准	W-横置横排
	1256—Arabic	24 点不等宽	256	标准	W-横置横排
1257—Baltic	12x24	256	标准	W-横置横排	
			打印体	W-横置横排	
928—Greek	12x24	96	标准	W-横置横排	
Hebrew old	12x24	96	标准	W-横置横排	
International character	12X24	132	标准	W-横置横排	
			打印体	W-横置横排	
katakana	12X24	64	打印体	W-横置横排	
数字符号	数字符号	14x28	15	黑体	W-横置横排
		20x40	12	黑体	W-横置横排
		28 点不等宽	15	Arial	W-横置横排
		40 点不等宽	13	Arial	W-横置横排
条形码	EAN13	16x26	60	标准	W-横置横排
	CODE128	16x20	107	标准	W-横置横排
其他符号	天线符号		5	标准	W-横置横排
	电池符号		4	标准	W-横置横排

## 2 操作指令

### 2.1 Instruction Parameter(指令参数)

Instruction	Description	Instruction Code(One-Byte)		Address Bytes	Dummy Bytes	Data Bytes
Read	Read Data Bytes	0000 0011	03 h	3	—	1 to ∞
Fast Read	Read Data Bytes at Higher Speed	0000 1011	0B h	3	1	1 to ∞

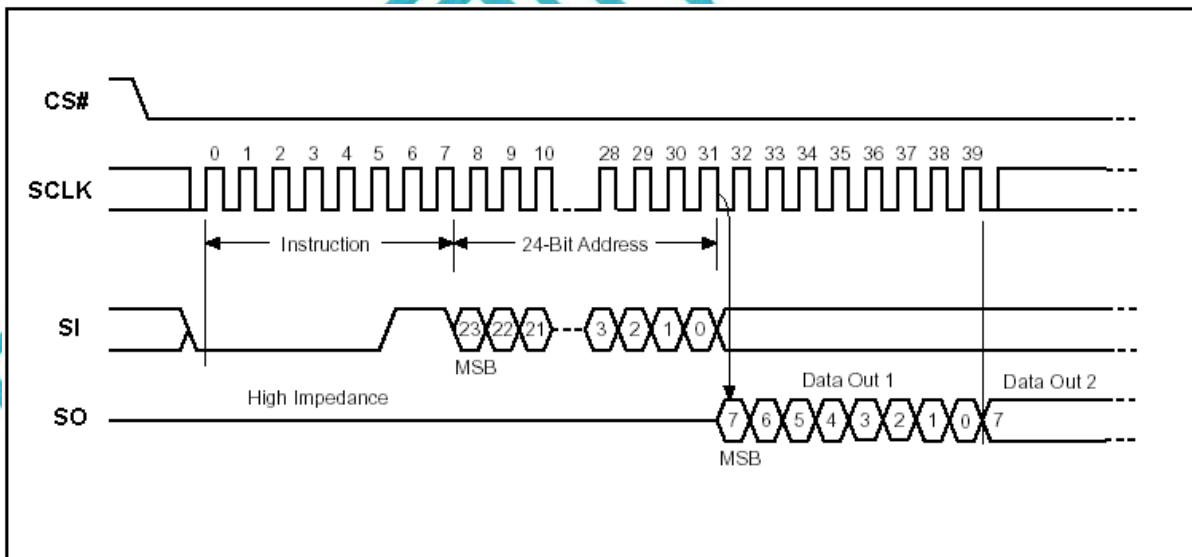
### 2.2 Read Data Bytes (一般读取)

Read Data Bytes 需要用指令码来执行每一次操作。READ 指令的时序如下(图):

- 首先把片选信号 (CS#) 变为低, 紧跟着的是 1 个字节的命令字 (03 h) 和 3 个字节的地址和通过串行数据输入引脚 (SI) 移位输入, 每一位在串行时钟 (SCLK) 上升沿被锁存。
- 然后该地址的字节数据通过串行数据输出引脚 (SO) 移位输出, 每一位在串行时钟 (SCLK) 下降沿被移出。
- 读取字节数据后, 则把片选信号 (CS#) 变为高, 结束本次操作。

如果片选信号 (CS#) 继续保持为底, 则下一个地址的字节数据继续通过串行数据输出引脚 (SO) 移位输出。

图: Read Data Bytes (READ) Instruction Sequence and Data-out sequence:



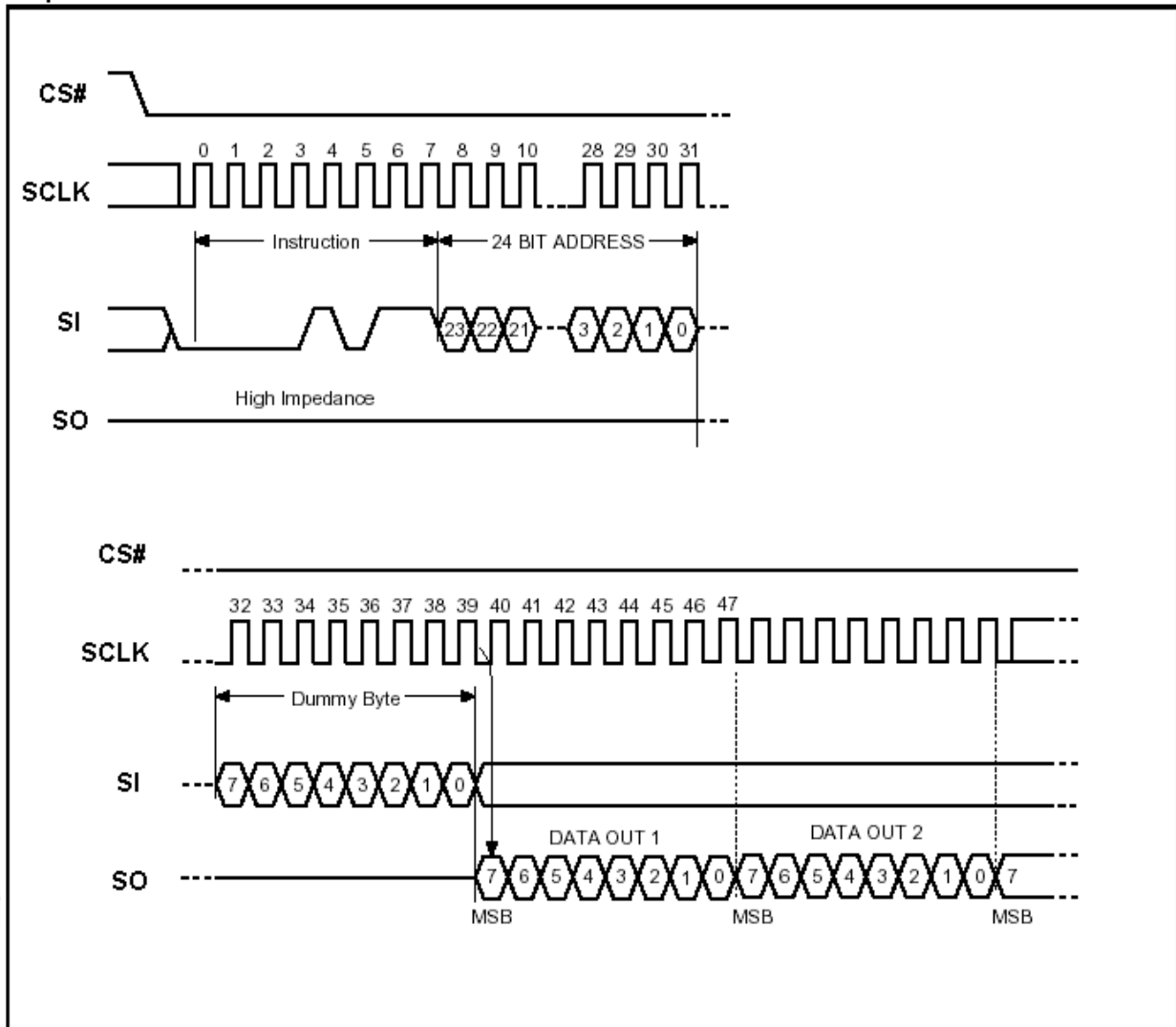
### 2.3 Read Data Bytes at Higher Speed (快速读取点阵数据)

Read Data Bytes at Higher Speed 需要用指令码来执行操作。READ\_FAST 指令的时序如下(图):

- 首先把片选信号 (CS#) 变为低, 紧跟着的是 1 个字节的命令字 (0B h) 和 3 个字节的地址以及一个字节 Dummy Byte 通过串行数据输入引脚 (SI) 移位输入, 每一位在串行时钟 (SCLK) 上升沿被锁存。

- 然后该地址的字节数据通过串行数据输出引脚（SO）移位输出，每一位在串行时钟（SCLK）下降沿被移出。
- 如果片选信号（CS#）继续保持为底，则下一个地址的字节数据继续通过串行数据输出引脚（SO）移位输出。例：读取一个 15x16 点阵汉字需要 32Byte，则连续 32 个字节读取后结束一个汉字的点阵数据读取操作。  
如果不需要继续读取数据，则把片选信号（CS#）变为高，结束本次操作。

图：Read Data Bytes at Higher Speed (READ\_FAST) Instruction Sequence and Data-out sequence:





### 3 字符点阵字库地址表

序号	字库内容	编码体系 ( 字符集 )	字符数	起始地址	参考算法
1	5X7 点 ASCII 字符	ASCII 字符集	96	20000	4.1.1
2	7X8 点 ASCII 字符		96	20300	4.1.2
3	6X12 点 ASCII 字符		96	20600	4.1.3
4	8X16 点 ASCII 字符		96	20C00	4.1.4
5	8X16 点粗体 ASCII 字符		96	21400	4.1.5
6	9X24 ASCII 字符		96	60000	4.1.6
7	12 点阵不等宽 ASCII 方头字符		96	21A00	4.1.7
8	12 点阵不等宽 ASCII 白正字符		96	223C0	4.1.8
9	16 点阵不等宽 ASCII 方头字符		96	22D80	4.1.9
10	16 点阵不等宽 ASCII 白正字符		96	23A40	4.1.10
11	24 点阵不等宽 ASCII 方头字符		96	24700	4.1.11
12	32 点不等宽 ASCII 方头字符		96	61200	4.1.12
13	32 点不等宽 ASCII 白正字符		96	642C0	4.1.13
14	8X16 拉丁文	UNICODE	496	262C0	4.2.1
15	12X24 拉丁文		496	2A3C0	4.2.2
16	12X24 拉丁文 ( Arail)		496	6AAC6	4.2.3
17	12 点不等宽拉丁文		496	3F080	4.2.4
18	16 点不等宽拉丁文		496	38640	4.2.5
19	8X16 希腊文		96	281C0	4.2.6

20	12X24 希腊文		96	300C0	4.2.7	
21	12 点不等宽希腊文		96	422E0	4.2.8	
22	16 点不等宽希腊文		96	3C820	4.2.9	
23	8X16 基里尔文		208	287C0	4.2.10	
24	12X24 基里尔文		208	312C0	4.2.11	
25	12X24 西里尔文 ( Arail)		208	6E6C6	4.2.12	
26	12 点不等宽基里尔文		208	42CA0	4.2.13	
27	16 点不等宽基里尔文		208	3D4E0	4.2.14	
28	8X16 希伯来文		112	294C0	4.2.15	
29	12X24 希伯来文		112	70DC6	4.2.16	
30	8X16 泰文		128	29BC0	4.2.17	
31	24 点不等宽泰文		128	72086	4.2.18	
32	8x16 日文假名		64	7324C	4.2.19	
33	12X24 日文假名		64	7363C	4.2.20	
34	16 点不等宽阿拉伯文		576	339C0	4.2.21	
35	24 点不等宽阿拉伯文		576	54BC0	4.2.22	
36	8x16ISO8859NO.1 ~ NO.16 (Without No.6 and No.12)		ISO 8859	128X14	441C0	4.3.1— 4.3.14
37	5x7ISO8859NO.1 ~ NO.16 (Without No.6 and No.12)			128X14	4B1C0	4.3.15— 4.3.28
38	5x10 LCM 字符		LCM	256X7	4E9C0	4.4
39	5x7 LCM 字符			256	52FC0	

40	LCM 字符备用区- 1 (5x10 )		256	537C0	
41	LCM 字符备用区- 2 (5x10 )		256	541C0	
42	437--USA,Standard Europe)		256	7BDFC	4.5.1
43	737--Greek		256	7BFFC	4.5.2
44	775--Baltic		256	7C1FC	4.5.3
45	850--Multilingual		256	7C3FC	4.5.4
46	852--Latin 2		256	7C5FC	4.5.5
47	855--Cyrillic		256	7C7FC	4.5.6
48	857--Turkish		256	7C9FC	4.5.7
49	858--Euro		128	7CBFC	4.5.8
50	860--Portuguese		256	7CCFC	4.5.9
51	862--Hebrew		256	7CEFC	4.5.10
52	863--Canadian French	CODE PAGE	256	7D0FC	4.5.11
53	864--Arabic		256	7D2FC	4.5.12
54	865--Nordic		256	7D4FC	4.5.13
55	866--Cyrillic 2		256	7D6FC	4.5.14
56	1251--Cyrillic		256	7D8FC	4.5.15
57	1252--Latin I		256	7DAFC	4.5.16
58	1253--Greek		256	7DCFC	4.5.17
59	1254--Turkish		256	7DEFC	4.5.18
60	1255--Hebrew New		256	7E0FC	4.5.19
61	1256--Arabic		256	7E2FC	4.5.20

62	1257--Baltic		256	7E4FC	4.5.21
63	928--Greek		96	7E6FC	4.5.22
64	Hebrew old code		96	7E7BC	4.5.23
65	International character code		132	7E87C	4.5.24
66	12x24Katakana		64	7423C	4.5.25
67	14x28 黑体半角_数字符号	数字符号	15	67380	4.6.1
68	20x40 点黑体半角_数字符号		12	676C8	4.6.2
69	28 点不等宽_数字符号		15	67C68	4.6.3
70	40 点不等宽_数字符号		13	68316	4.6.4
71	条形码字符_EAN13	条形码	60	68C8E	4.6.5
72	条形码字符_CODE128		107	69936	4.6.6
73	天线符号	其他符号	5	6A9EE	4.6.7
74	电池符号		4	6AA66	4.6.8

## 4 字符点阵数据在芯片中的地址计算方法

用户只要知道字符的内码，就可以计算出该字符点阵在芯片中的地址，然后就可从该地址连续读出点阵信息用于显示。

### 4.1 ASCII 字符的地址计算

#### 4.1.1 5x7 点阵 ASCII 标准字符

**参数说明：**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法：**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address = (ASCIICODE - 0x20) * 8 + 0x020000
```

#### 4.1.2 7x8 点阵 ASCII 标准字符

**参数说明：**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法：**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20)*8 + 0x20300
```

#### 4.1.3 6x12 点阵 ASCII 标准字符

**说明：**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法：**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 12 + 0x20600
```

#### 4.1.4 8x16 点阵 ASCII 标准字符

**说明：**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法：**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 16 + 0x20C00;
```

#### 4.1.5 8x16 点阵 ASCII 粗体字符

**说明：**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 16 + 0x21400
```

#### 4.1.6 9x24 点阵 ASCII 字符

参数说明:

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

BaseAdd: 表示点阵数据在字库芯片中的起始地址。

**计算方法:**

```
BaseAdd=0x060000;
if((unicode>=0x20)&&(unicode<=0x7F))
    addr = (unicode-0x20)*48+ BaseAdd;
```

#### 4.1.7 12 点阵不等宽 ASCII 方头 (Arial) 字符

**说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 26 + 0x21A00
```

#### 4.1.8 12 点阵不等宽 ASCII 白正 (Times) 字符

**说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 26 + 0x223C0
```

#### 4.1.9 16 点阵不等宽 ASCII 方头 (Arial) 字符

**说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 34 + 0x22D80
```

#### 4.1.10 16 点阵不等宽 ASCII 白正 (Times) 字符

**说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 34 + 0x23A40
```

#### 4.1.11 24 点阵不等宽 ASCII 方头 (Arial) 字符

**说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

**计算方法:**

```
if(ASCIICODE >=0x20 && ASCIICODE <=0x7F)
    Address =(ASCIICODE-0x20) * 74 + 0x24700
```

#### 4.1.12 32 点不等宽 Arial ASCII 字符

**参数说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

BaseAdd: 表示点阵数据在字库芯片中的起始地址。

**计算方法:**

```
BaseAdd=0x061200;
if((unicode>=0x20)&&(unicode<=0x7F))
    addr = (unicode-0x20)*130+ BaseAdd;
```

#### 4.1.13 32 点不等宽白正 ASCII 字符

**参数说明:**

ASCIICode: 表示 ASCII 码 (8bits)

Address: ASCII 字符点阵在芯片中的字节地址。

BaseAdd: 表示点阵数据在字库芯片中的起始地址。

**计算方法:**

```
BaseAdd=0x0642C0;
if((unicode>=0x20)&&(unicode<=0x7F))
    addr = (unicode-0x20)*130+ BaseAdd;
```

## 4.2 UNICODE 字符的地址计算

### 4.2.1 8x16 点阵拉丁文系字符

**说明:。**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```

if (FontCode>=0x0020 && FontCode<=0x007F)
    Address=(FontCode-0x0020) * 16 +0x262C0;
Else if (FontCode>=0x00A0 && FontCode<=0x017F)
    Address=(FontCode-0x0040) * 16 +0x262C0;
else if (FontCode>=0x01A0 && FontCode<=0x01CF)
    Address=(FontCode-0x01A0+320) * 16 +0x262C0;
else if (FontCode>=0x01F0 && FontCode<=0x01FF)
    Address=(FontCode-0x01F0+368) * 16 +0x262C0;
else if (FontCode>=0x0210 && FontCode<=0x021F)
    Address=(FontCode-0x0210+384) * 16 +0x262C0;
else if (FontCode>=0x1EA0 && FontCode<=0x1EFF)
    Address=(FontCode-0x1EA0+400) * 16 +0x262C0;

```

#### 4.2.2 12x24 点阵拉丁文字符

说明:

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

```

if (FontCode>=0x0020 && FontCode<=0x007F)
    Address=(FontCode-0x0020) * 48 +0x2A3C0;
else if (FontCode>=0x00A0 && FontCode<=0x017F)
    Address=(FontCode-0x00A0+96) * 48 +0x2A3C0;
else if (FontCode>=0x01A0 && FontCode<=0x01CF)
    Address=(FontCode-0x01A0+320) * 48 +0x2A3C0;
else if (FontCode>=0x01F0 && FontCode<=0x01FF)
    Address=(FontCode-0x01F0+368) * 48 +0x2A3C0;
else if (FontCode>=0x0210 && FontCode<=0x021F)
    Address=(FontCode-0x0210+384) * 48 +0x2A3C0;
else if (FontCode>=0x1EA0 && FontCode<=0x1EFF)
    Address=(FontCode-0x1EA0+400) * 48 +0x2A3C0;

```

#### 4.2.3 12x24 点阵拉丁文字符 (Arial)

说明:

FontCode: 表示 unicode 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

```

if (FontCode>=0x0020 && FontCode<=0x007F)
    Address=(FontCode-0x0020) * 48 +0x6AAC6;
else if (FontCode>=0x00A0 && FontCode<=0x017F)
    Address=(FontCode-0x00A0+96) * 48 +0x6AAC6;

```

#### 4.2.4 12 点阵不等宽拉丁文字符

说明:



FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```

if (FontCode>=0x0020 && FontCode<=0x007F)
    Address=(FontCode-0x0020) * 26 +0x3F080;
else if (FontCode>=0x00A0 && FontCode<=0x017F)
    Address=(FontCode-0x00A0+96) * 26 +0x3F080;
else if (FontCode>=0x01A0 && FontCode<=0x01CF)
    Address=(FontCode-0x01A0+320) * 26 +0x3F080;
else if (FontCode>=0x01F0 && FontCode<=0x01FF)
    Address=(FontCode-0x01F0+368) * 26 +0x3F080;
else if (FontCode>=0x0210 && FontCode<=0x021F)
    Address=(FontCode-0x0210+384) * 26 +0x3F080;
else if (FontCode>=0x1EA0 && FontCode<=0x1EFF)
    Address=(FontCode-0x1EA0+400) * 26 +0x3F080;
    
```

#### 4.2.5 16 点阵不等宽拉丁文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```

if (FontCode>=0x0020 && FontCode<=0x007F)
    Address=(FontCode-0x0020) * 34 +0x38640;
else if (FontCode>=0x00A0 && FontCode<=0x017F)
    Address=(FontCode-0x00A0+96) * 34 +0x38640;
else if (FontCode>=0x01A0 && FontCode<=0x01CF)
    Address=(FontCode-0x01A0+320) * 34 +0x38640;
else if (FontCode>=0x01F0 && FontCode<=0x01FF)
    Address=(FontCode-0x01F0+368) * 34 +0x38640;
else if (FontCode>=0x0210 && FontCode<=0x021F)
    Address=(FontCode-0x0210+384) * 34 +0x38640;
else if (FontCode>=0x1EA0 && FontCode<=0x1EFF)
    Address=(FontCode-0x1EA0+400) * 34 +0x38640;
    
```

#### 4.2.6 8x16 点阵希腊文系字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```

if (FontCode>=0x0370 && FontCode<=0x03CF)
    Address=(FontCode-0x0370) * 16 +0x281C0;
    
```

#### 4.2.7 12x24 点阵希腊文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0370 && FontCode<=0x03CF)
    Address=(FontCode-0x0370) * 48 +0x300C0;
```

#### 4.2.8 12 点不等宽希腊文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0370 && FontCode<=0x03CF)
    Address=(FontCode-0x0370) * 26 +0x422E0;
```

#### 4.2.9 16 点阵不等宽希腊文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0370 && FontCode<=0x03CF)
    Address=(FontCode-0x0370) * 34 +0x3C820;
```

#### 4.2.10 8x16 点阵基里尔文系字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0400 && FontCode<=0x045F)
    Address=(FontCode-0x0400) * 16 +0x287C0;
else if (FontCode>=0x0490 && FontCode<=0x04FF)
    Address=(FontCode-0x0490+96) * 16 +0x287C0;
```

#### 4.2.11 12x24 点阵基里尔文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0400 && FontCode<=0x045F)
```

```
Address=(FontCode-0x0400) * 48 +0x312C0;
else if (FontCode>=0x0490 && FontCode<=0x04FF)
    Address=(FontCode-0x0490+96) * 48 +0x312C0;
```

#### 4.2.12 12x24 点阵基里尔文字符 (Arial)

**说明:**

FontCode: 表示 unicode 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0400 && FontCode<=0x045F)
    Address=(FontCode-0x0400) * 48 +0x6E6C6;
else if (FontCode>=0x0490 && FontCode<=0x04FF)
    Address=(FontCode-0x0490+96) * 48 +0x6E6C6;
```

#### 4.2.13 12 点不等宽基里尔文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0400 && FontCode<=0x045F)
    Address=(FontCode-0x0400) * 26 +0x42CA0;
else if (FontCode>=0x0490 && FontCode<=0x04FF)
    Address=(FontCode-0x0490+96) * 26 +0x42CA0;
```

#### 4.2.14 16 点阵不等宽基里尔文字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0400 && FontCode<=0x045F)
    Address=(FontCode-0x0400) * 34 +0x3D4E0;
else if (FontCode>=0x0490 && FontCode<=0x04FF)
    Address=(FontCode-0x0490+96) * 34 +0x3D4E0;
```

#### 4.2.15 8x16 点阵希伯来文系字符

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0590 && FontCode<=0x05FF)
    Address=(FontCode-0x0590) * 16 +0x294C0;
```

#### 4.2.16 12X24 点希伯来文系字符

说明:

FontCode: 表示 unicode 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

if (FontCode>=0x0591 && FontCode<=0x05FF)

Address=(FontCode-0x0591) \* 48 +0x70DC6;

#### 4.2.17 8x16 点阵泰文字符

说明:

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

if (FontCode>=0x0E00 && FontCode<=0x0E5F)

Address=(FontCode-0x0E00) \* 16 +0x29BC0;

#### 4.2.18 24 点等宽泰文

说明:

FontCode: 表示 unicode 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

if (FontCode>=0x0E00 && FontCode<=0x0E5F)

Address=(FontCode-0x0E00) \* 50 +0x72086;

#### 4.2.19 8x16 日文假名

参数说明:

CODE: UNICODE 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07324C ;

if(CODE>=0xFF61 && CODE<=0xFF9F)

Address = (CODE-0xFF61)\*16+BaseAddr;

#### 4.2.20 12x24 日文假名

参数说明:

CODE: UNICODE 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07363C ;
if(CODE>=0xFF61 && CODE<=0xFF9F)
Address = (CODE-0xFF61)*48+BaseAddr;
```

#### 4.2.21 16 点阵不等宽阿拉伯文字符

**说明:**

UNICODE\_alb: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
BaseAdd= 0x0339C0;
if( UNICODE_alb >= 0x0600 && UNICODE_alb <= 0x06FF )//alb 1
    Address = 34*(UNICODE_alb-0x0600)+ BaseAdd;
else if( UNICODE_alb >= 0xfb50 && UNICODE_alb <= 0xfbff )//alb 2
    Address = 34*(16*16+UNICODE_alb-0xfb50)+ BaseAdd;
else if( UNICODE_alb >= 0xfe70 && UNICODE_alb <= 0xfeff )//alb 3
    Address = 34*(16*11+16*16+UNICODE_alb-0xfe70)+ BaseAdd;
```

#### 4.2.22 24 点不等宽阿拉伯文

**说明:**

UNICODE\_alb: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
BaseAdd= 0x54BC0;
if( UNICODE_alb >= 0x0600 && UNICODE_alb <= 0x06FF )//alb 1
    Address = 74 *(UNICODE_alb-0x0600)+ BaseAdd;
else if( UNICODE_alb >= 0xfb50 && UNICODE_alb <= 0xfbff )//alb 2
    Address = 74 *(16*16+UNICODE_alb-0xfb50)+ BaseAdd;
else if( UNICODE_alb >= 0xfe70 && UNICODE_alb <= 0xfeff )//alb 3
    Address = 74 *(16*11+16*16+UNICODE_alb-0xfe70)+ BaseAdd;
```

用户只要知道字符的内码, 就可以计算出该字符点阵在芯片中的地址, 然后就可从该地址连续读出点阵信息用于显示。

## 4.3 ISO 字符的地址计算

### 4.3.1 8x16 ISO8859-1

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FontCode >=0x0080 && FontCode <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x441C0;
```

#### 4.3.2 8x16 ISO8859-2

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x449C0;
```

#### 4.3.3 8x16 ISO8859-3

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x451C0;
```

#### 4.3.4 8x16 ISO8859-4

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x459C0;
```

#### 4.3.5 8x16 ISO8859-5

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x461C0;
```

#### 4.3.6 8x16 ISO8859-7

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x469C0;
```

#### 4.3.7 8x16 ISO8859-8

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x471C0;
```

#### 4.3.8 8x16 ISO8859-9

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x479C0;
```

#### 4.3.9 8x16 ISO8859-10

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x481C0;
```

#### 4.3.10 8x16 ISO8859-11

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x489C0;
```

#### 4.3.11 8x16 ISO8859-13

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x491C0;
```

#### 4.3.12 8x16 ISO8859-14

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x49CC0;
```

#### 4.3.13 8x16 ISO8859-15

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x4A1C0;
```

#### 4.3.14 8x16 ISO8859-16

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x4A9C0;
```

#### 4.3.15 5x7 ISO8859-1

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4B1C0;
```

#### 4.3.16 5x7 ISO8859-2

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4B5C0;
```

#### 4.3.17 5x7 ISO8859-3

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4B9C0;
```

#### 4.3.18 5x7 ISO8859-4

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)



Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 +0x4BDC0;
```

#### 4.3.19 5x7 ISO8859-5

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4C1C0;
```

#### 4.3.20 5x7 ISO8859-7

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4C5C0;
```

#### 4.3.21 5x7 ISO8859-8

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4C9C0;
```

#### 4.3.22 5x7 ISO8859-9

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 16 + 0x4CDC0;
```

#### 4.3.23 5x7 ISO8859-10

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4D1C0;
```

#### 4.3.24 5x7 ISO8859-11

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4D5C0;
```

#### 4.3.25 5x7 ISO8859-13

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4D9C0;
```

#### 4.3.26 5x7 ISO8859-14

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4DDC0;
```

#### 4.3.27 5x7 ISO8859-15

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4E1C0;
```

#### 4.3.28 5x7 ISO8859-16

**说明:**

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if(FONTCODE >=0x0080 && FONTCODE <=0x00FF)
    Address =(FONTCODE-0x80) * 8 + 0x4E5C0;
```

## 4.4 LCM 字符的地址计算

### 4.4.1 5x10 LCM 定制区 1

**说明:**

用于显示屏调用时, 按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode >= 0x0000 && FontCode <= 0x00FF)
    Address = FontCode * 10 + 0x4E9C0;
```

### 4.4.2 5x10 LCM 定制区 2

**说明:**

用于显示屏调用时, 按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode >= 0x0000 && FontCode <= 0x00FF)
    Address = FontCode * 10 + 0x4F3C0;
```

### 4.4.3 5x10 LCM 定制区 3

**说明:**

用于显示屏调用时, 按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode >= 0x0000 && FontCode <= 0x00FF)
    Address = FontCode * 10 + 0x4FDC0;
```

### 4.4.4 5x10 LCM 定制区 8

**说明:**

用于显示屏调用时, 按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode >= 0x0000 && FontCode <= 0x00FF)
    Address = FontCode * 10 + 0x507C0;
```

### 4.4.5 5x10 LCM 定制区 11

**说明:**

用于显示屏调用时, 按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0000 && FontCode<=0x00FF)
    Address=FontCode * 10 +0x511C0;
```

#### 4.4.6 5x10 LCM 定制区 12

**说明:**

用于显示屏调用时,按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0000 && FontCode<=0x00FF)
    Address=FontCode * 10 +0x51BC0;
```

#### 4.4.7 5x10 LCM 定制区 13

**说明:**

用于显示屏调用时,按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0000 && FontCode<=0x00FF)
    Address=FontCode * 10 +0x525C0;
```

#### 4.4.8 5x10 LCM 定制区 0

**说明:**

用于显示屏调用时,按本节所述 8X10 程序调用

FontCode: 表示 UNICODE 内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

**计算方法:**

```
if (FontCode>=0x0000 && FontCode<=0x00FF)
    Address=FontCode * 10 +0x52FC0;
```

### 4.5 CODE PAGE

#### 4.5.1 437—USA,Standard Europe

参数说明:

CODE: CODE PAGE 437 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x05BDFC ;

if(CODE>=0x00 && CODE<=0xFF)

Address = CODE\*2+BaseAddr;

#### 437 special graphic characters

参数说明:

CODE: CODE PAGE 437 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

BaseAddr=0x7E984;

if(CODE>=0x00 && CODE<=0x1F)

Address=(CODE-1)\*48+BaseAddr;

else if(CODE==0x7F)

Address=0x07EF54;

#### 4.5.2 737—Greek

参数说明:

CODE: CODE PAGE 737 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07BFFC ;

if(CODE>=0x00 && CODE<=0xFF)

Address = CODE\*2+BaseAddr;

#### 4.5.3 775—Baltic

参数说明:

CODE: CODE PAGE 775 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07C1FC ;

if(CODE>=0x00 && CODE<=0xFF)

Address = CODE\*2+BaseAddr;

CODE PAGE 850 转 UNICODE 调用算法

参数说明:

#### 4.5.4 850—Multilingual

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07C3FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
Address = CODE*2+BaseAddr;
```

#### 4.5.5 852—Latin 2

参数说明:

CODE: CODE PAGE 852 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07C5FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.6 855—Cyrillic

参数说明:

CODE: CODE PAGE 855 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07C7FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.7 857—Turkish

参数说明:

CODE: CODE PAGE 857 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07C9FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.8 858—Euro

参数说明:

CODE: CODE PAGE 858 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07CBFC ;

```
if(CODE>=0x80 && CODE<=0xFF)
Address = (CODE-0x80)*2+BaseAddr;
```

#### 4.5.9 860—Portuguese

参数说明:

CODE: CODE PAGE 860 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07CCFC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.10 862—Hebrew

参数说明:

CODE: CODE PAGE 862 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07CEFC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.11 863—Canadian French

参数说明:

CODE: CODE PAGE 863 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07D0FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.12 864—Arabic

参数说明:

CODE: CODE PAGE 864 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07D2FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
Address = CODE*2+BaseAddr;
```

#### 4.5.13 865—Nordic

参数说明:

CODE: CODE PAGE 865 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07D4FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.14 866—Cyrillic 2

参数说明:

CODE: CODE PAGE 866 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07D6FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.15 1251—Cyrillic

参数说明:

CODE: CODE PAGE 1251 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07D8FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.16 1252—Latin I

参数说明:

CODE: CODE PAGE 1252 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07DAFC ;



```
if(CODE>=0x00 && CODE<=0xFF)  
Address = CODE*2+BaseAddr;
```

#### 4.5.17 1253—Greek

参数说明:

CODE: CODE PAGE 1253 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07DCFC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.18 1254—Turkish

参数说明:

CODE: CODE PAGE 1254 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07DEFB ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.19 1255—Hebrew New

参数说明:

CODE: CODE PAGE 1255 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E0FC ;

```
if(CODE>=0x00 && CODE<=0xFF)
```

```
Address = CODE*2+BaseAddr;
```

#### 4.5.20 1256—Arabic

参数说明:

CODE: CODE PAGE 1256 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E2FC ;

if(CODE>=0x00 && CODE<=0xFF)  
Address = CODE\*2+BaseAddr;

#### 4.5.21 1257—Baltic

参数说明:

CODE: CODE PAGE 1257 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E4FC ;

if(CODE>=0x00 && CODE<=0xFF)

Address = CODE\*2+BaseAddr;

#### 4.5.22 928—GREEK

参数说明:

CODE: CODE PAGE GREEK 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E6FC ;

if(CODE>=0xA0 && CODE<=0xFF)

Address = (CODE-0xA0)\*2+BaseAddr;

#### 4.5.23 HEBREW

参数说明:

CODE: CODE PAGE HEBREW 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E7BC ;

if(CODE>=0x20 && CODE<=0x80)

Address = (CODE-0x20)\*2+BaseAddr;

#### 4.5.24 International Character

##### 4.5.24.1 INTERNATIONAL CHARACTER CODE USA

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE USA 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E87C ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.2 INTERNATIONAL CHARACTER CODE FRANCE

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE FRANCE 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E894 ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.3 INTERNATIONAL CHARACTER CODE GERMANY

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE GERMANY 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E8AC ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
```

```
break;
}
```

#### 4.5.24.4 INTERNATIONAL CHARACTER CODE U.K

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE U.K 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E8C4 ;

char i;

char CODE[12]={0x23,0x24,0x40,0x5b,0x5c  
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};

for(i=0;i<12;i++)

```
{
  if(INTERCODE==CODE[i])
    Address=i*2+BaseAddr;
  break;
}
```

#### 4.5.24.5 INTERNATIONAL CHARACTER CODE DENMARK I

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE DENMARK I 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E8DC ;

char i;

char CODE[12]={0x23,0x24,0x40,0x5b,0x5c  
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};

for(i=0;i<12;i++)

```
{
  if(INTERCODE==CODE[i])
    Address=i*2+BaseAddr;
  break;
}
```

#### 4.5.24.6 INTERNATIONAL CHARACTER CODE SWEDEN

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE SWEDEN 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E8F4 ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.7 INTERNATIONAL CHARACTER CODE LTALY

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE LTALY 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E90C ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.8 INTERNATIONAL CHARACTER CODE SPAIN

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE SPAIN 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07E924 ;
char i;
char CODE[12]={0x23,0x24,0x40,0x5b,0x5c
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};
for(i=0;i<12;i++)
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.9 INTERNATIONAL CHARACTER CODE JAPAN

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE JAPAN 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E93C ;

char i;

char CODE[12]={0x23,0x24,0x40,0x5b,0x5c  
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};

for(i=0;i<12;i++)

```
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.10 INTERNATIONAL CHARACTER CODE NORWAY

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE NORWAY 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E954 ;

char i;

char CODE[12]={0x23,0x24,0x40,0x5b,0x5c  
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};

for(i=0;i<12;i++)

```
{
    if(INTERCODE==CODE[i])
        Address=i*2+BaseAddr;
    break;
}
```

#### 4.5.24.11 INTERNATIONAL CHARACTER CODE DENMARK II

参数说明:

INTERCODE:INTERNATIONAL CHARACTER CODE DENMARK II 编码。

CODE[]: INTERNATIONAL CHARACTER CODE 编码 TABLE。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAddr=0x07E96C ;

char i;

char CODE[12]={0x23,0x24,0x40,0x5b,0x5c  
0x5d,0x5e,0x60,0x7b,0x7c,0x7d,0x7e};

for(i=0;i<12;i++)

```
{
  if(INTERCODE==CODE[i])
    Address=i*2+BaseAddr;
  break;
}
```

#### 4.5.25 Katakana

参数说明:

CODE: CODE PAGE KATAKANA 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

计算方法:

```
BaseAddr=0x07423C ;
if(CODE>=0x80 && CODE<=0x9F)
  Address = (CODE-0x80)*72+BaseAddr;
else if(CODE>=0xA0 && CODE<=0xDF)
  Address = (CODE-0xA0)*48+0x900+BaseAddr;
else if(CODE>=0xE0 && CODE<=0xFF)
  Address = (CODE-0xE0)*72+0x1500+BaseAddr;
```

COPE PAGE 补丁 UNICODE 调用

CODE: CODE PAGE KATAKANA 编码。

Address 表示点阵在芯片中的字节地址。

BaseAddr: 说明点阵数据在字库芯片中的起始地址。

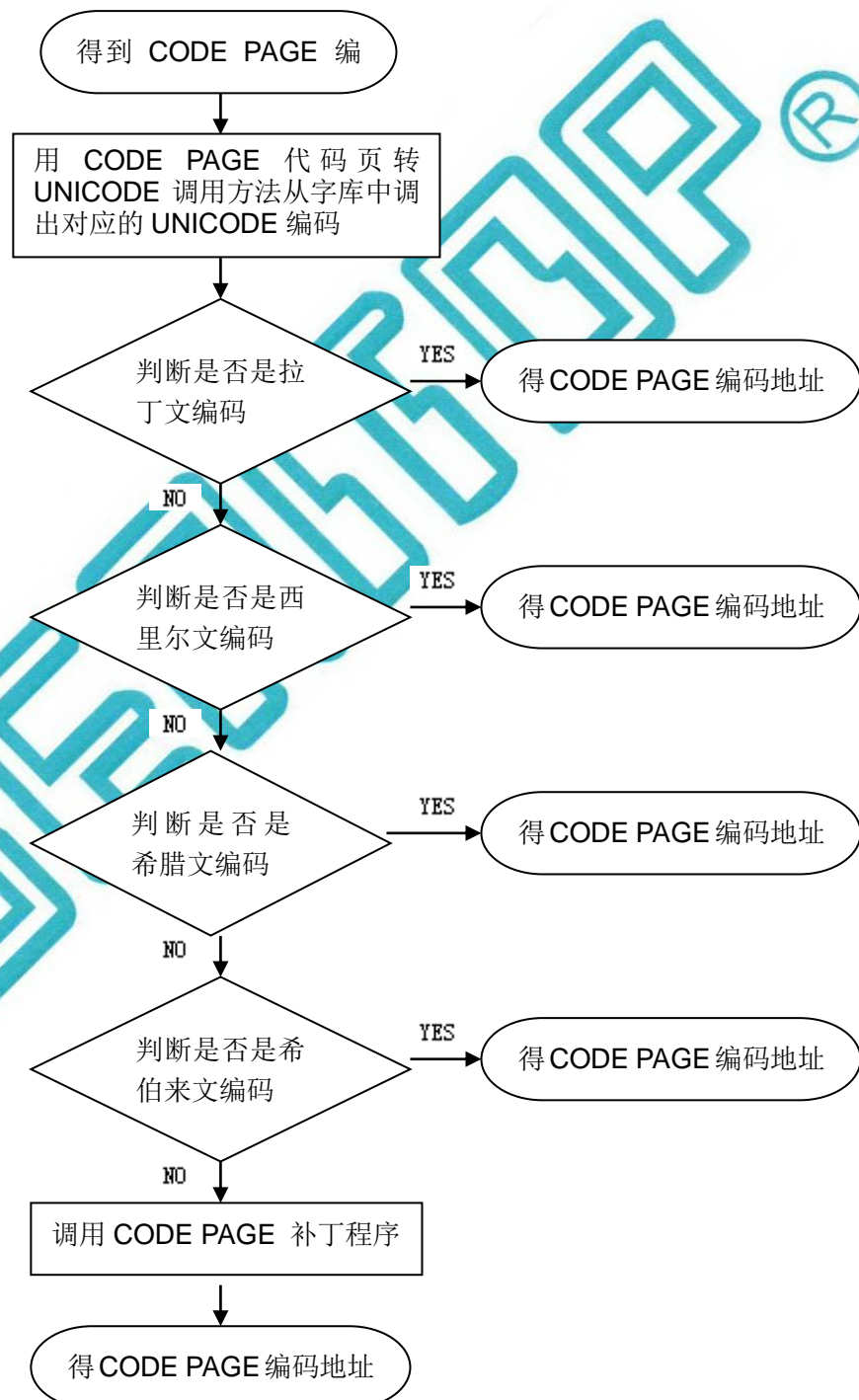
计算方法:

```
BaseAddr=0x07603C ;
if (code==0x0192){
  Address=BaseAddr;
}
else if (code>=0x0393 && code<=0x03c6){
  Address=(code-0x0393+1)*48+BaseAddr;
}
else if (code>=0x2013 && code<=0x20ac){
  Address=(code-0x2013+0x35)*48+BaseAddr;
}
else if (code>=0x2219 && code<=0x2265){
  Address=(code-0x2219+0xCf)*48+BaseAddr;
}
else if (code>=0x2310 && code<=0x2321){
  Address=(code-0x2310+0x11c)*48+BaseAddr;
}
else if (code>=0x2500 && code<=0x25a0){
  Address=(code-0x2500+0x12e)*48+BaseAddr;
}
else if (code>=0x02c6 && code<=0x02dd){
```

```

Address=(code-0x02c6+0x1Cf)*48+BaseAddr;
}
else if (code>=0x2116 && code<=0x2122){
    Address=(code-0x2116+0x1E7)*48+BaseAddr;
}
else
    return 0;
    
```

#### 4.5.26 CODE PAGE 调用流程





## 4.6 其他字符

### 数字符号

#### 4.6.1 14x28 黑体数字符号

说明:

此部分内容为 0 1 2 3 4 5 6 7 8 9 , . ¥ \$ £

Sequence: 表示 字符顺序, 从 0 开始计数。

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: 对应字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x067380;

Address = Sequence \* 56+ BaseAdd;

#### 4.6.2 20x40 黑体数字符号

说明:

此部分内容为 0 1 2 3 4 5 6 7 8 9 . ,

Sequence: 表示 字符顺序, 从 0 开始计数。

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: 对应字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x0676C8;

Address = (Sequence) \* 120+BaseAdd;

#### 4.6.3 28 点不等宽数字符号

说明:

此部分内容为 0 1 2 3 4 5 6 7 8 9 , . ¥ \$ £

Sequence: 表示 字符顺序, 从 0 开始计数。

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: 对应字符点阵在芯片中的字节地址。

注: 前两个字节为宽度信息。

计算方法:

BaseAdd=0x067C68;

Address = Sequence \* 114+ BaseAdd;

#### 4.6.4 40 点不等宽数字符号

说明:

此部分内容为 0 1 2 3 4 5 6 7 8 9 . ,

Sequence: 表示 字符顺序, 从 0 开始计数。

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: 对应字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x068316;

Address = Sequence \* 202 + BaseAddr;

## 条形码

### 4.6.5 EAN13 条形码

函数: DWORD\* BAR\_CODE(int\* BAR\_NUM)

功能: 将数组条形码转为对应条形码图形地址。

参数: int\* BAR\_NUM 条形码数字数组指针, BAR\_NUM[13]数组包含 13 个数字。

返回: 定义 DWORD BAR\_PIC\_ADDR[13];用于存放对应地址, 返回此数组指针。

DWORD BAR\_PIC\_ADDR[13];

DWORD\* BAR\_CODE(int\* BAR\_NUM)

```
{
    DWORD i,BaseAddr=0x068C8E;
    BAR_PIC_ADDR[0]=BAR_NUM[0]*54+540*0+ BaseAddr;
    BAR_PIC_ADDR[1]=BAR_NUM[1]*54+540*1+ BaseAddr;

    switch(BAR_NUM[0])
    {
    case 0:
        for(i=2;i<=6;i++)
        {
            BAR_PIC_ADDR[i]=BAR_NUM[i]*54+540*1+ BaseAddr;
        }
        break;

    case 1:
        BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*1+ BaseAddr;
        BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
        BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*1+ BaseAddr;
        BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*2+ BaseAddr;
        BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*2+ BaseAddr;
        break;

    case 2:
        BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*1+ BaseAddr;
        BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
        BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*2+ BaseAddr;
        BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*1+ BaseAddr;
        BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*2+ BaseAddr;
        break;

    case 3:
        BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*1+ BaseAddr;
        BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
```

```
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*1+ BaseAddr;
break;
```

case 4:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*2+ BaseAddr;
break;
```

case 5:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*2+ BaseAddr;
break;
```

case 6:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*1+ BaseAddr;
break;
```

case 7:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*2+ BaseAddr;
break;
```

case 8:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*1+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*1+ BaseAddr;
break;
```

case 9:

```
BAR_PIC_ADDR[2]=BAR_NUM[2]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[3]=BAR_NUM[3]*54+540*2+ BaseAddr;
BAR_PIC_ADDR[4]=BAR_NUM[4]*54+540*1+ BaseAddr;
```

```

        BAR_PIC_ADDR[5]=BAR_NUM[5]*54+540*2+ BaseAddr;
        BAR_PIC_ADDR[6]=BAR_NUM[6]*54+540*1+ BaseAddr;
        break;

    }

```

```

BAR_PIC_ADDR[7]=BAR_NUM[7]*54+540*3+ BaseAddr;
for(i=8;i<=11;i++)
{
    BAR_PIC_ADDR[i]=BAR_NUM[i]*54+540*4+ BaseAddr;
}
BAR_PIC_ADDR[12]=BAR_NUM[12]*54+540*5+ BaseAddr;
return BAR_PIC_ADDR;
}

```

#### 4.6.6 CODE 128 条形码

函数: DWORD\* BAR\_CODE(int\* BAR\_NUM)

功能: 将数组条形码转为对应条形码图形地址

参数: int\* BAR\_NUM 条形码数字数组指针, BAR\_NUM[4]数组包含 4 个条形码 ASCII 符(数组取值为 0~94)。

返回: 定义 DWORD BAR\_PIC\_ADDR[7] ; 用于存放对应地址, 返回数组指针。

设基地址: BaseAddr=0x069936;

起始符有 3 种模式

当 flag=1 时为 Code-128-A;

当 flag=2 时为 Code-128-B;

当 flag=3 时为 Code-128-C;

```

DWORD flag;
DWORD BAR_PIC_ADDR[7];
DWORD* BAR_CODE(int* BAR_NUM)
{
    int i;
    for(i=0;i<7;i++)
    {
        switch(flag)
        case 1 :
            if(i==0)
            {
                BAR_PIC_ADDR[i]=103*40+BaseAddr;
            }
            else if(i==1||i==2||i==3||i==4)
            {
                BAR_PIC_ADDR[i]=BAR_NUM[i-1]*40+BaseAddr;
            }

```

```

else if(i==5)
{
    BAR_PIC_ADDR[i]=95*40+BaseAddr;
}
else if(i==6)
{
    BAR_PIC_ADDR[i]=106*40+BaseAddr;
}
break;

```

case 2 :

```

if(i==0)
{
    BAR_PIC_ADDR[i]=104*40+BaseAddr;
}
else if(i==1||i==2||i==3||i==4)
{
    BAR_PIC_ADDR[i]=BAR_NUM[i-1]*40+BaseAddr;
}
else if(i==5)
{
    BAR_PIC_ADDR[i]=95*40+BaseAddr;
}
else if(i==6)
{
    BAR_PIC_ADDR[i]=106*40+BaseAddr;
}
break;

```

case 3 :

```

if(i==0)
{
    BAR_PIC_ADDR[i]=105*40+BaseAddr;
}
else if(i==1||i==2||i==3||i==4)
{
    BAR_PIC_ADDR[i]=BAR_NUM[i-1]*40+BaseAddr;
}
else if(i==5)
{
    BAR_PIC_ADDR[i]=95*40+BaseAddr;
}
else if(i==6)
{
    BAR_PIC_ADDR[i]=106*40+BaseAddr;
}

```

```

    }
    break;

    default:
    break;
}
return BAR_PIC_ADDR;
}

```

注：在屏上打点时要按照国家规范进行拼凑。

#### 4.6.7 天线符号

函数：DWORD\* Antenna\_CODE(int\* NUM)

功能：获取 12X12 天线 调用地址。

参数：NUM 0123 带表天线信号强度。

返回：数据地址

```

DWORD* Antenna_CODE(int* NUM)
{
    DWORD BaseAdd=0x06A9EE;
    AddressBaseAdd+NUM*24);
}

```

#### 4.6.8 电池符号

函数：DWORD\* Battery\_CODE(int\* NUM)

功能：获取 12X12 电池 调用地址。

参数：NUM 0123 带表电池电量。

返回：数据地址

```

DWORD* Battery_CODE(int* NUM)
{
    DWORD BaseAdd=0x06AA66;
    AddressBaseAdd+NUM*24);
}

```

### 4.7 UTF-8 转 UTF-16 转换程序

```

//-----
//函数名称: UFT8toUTF16(unsigned char *pUTF8,unsigned short *pUTF16 )
//函数说明: UTF8 转 UTF16 码函数
//参数说明: pUTF8 UTF8 数据指针; pUTF16 获得 UTF16 的指针
//返回说明: 该 UTF8 字符的占用字节数。
//-----
unsigned char UFT8toUTF16(unsigned char *pUTF8,unsigned short *pUTF16 )
{
    unsigned char bytes[3];
    unsigned short UNICODE16;

```

```

bytes[0] = *pUTF8++;
if( bytes[0] < 0x80 )
{
    *pUTF16 = bytes[0];
    return(1);
}
bytes[1] = *pUTF8++;
if( bytes[0] >= 0xC0 && bytes[0] < 0xE0 )
{
    UNICODE16 = 0x1f&bytes[0];
    *pUTF16 = (UNICODE16<<6)+(0x3f&bytes[1]);
    return(2);
}
bytes[2] = *pUTF8++;
if( bytes[0] >= 0xE0 && bytes[0] < 0xF0 )
{
    UNICODE16 = 0x0f&bytes[0];
    UNICODE16 = (UNICODE16<<6)+(0x3f&bytes[1]);
    *pUTF16 = (UNICODE16<<6)+(0x3f&bytes[2]);
    return(3);
}
else
    return(0x00);
}
    
```

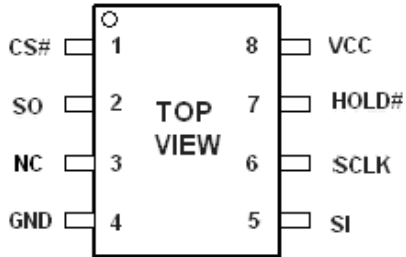
#### 4.8 5X7 字符 ISO8859 转 UNICODE 程序

序号	补丁描述	修补方式	对应文档	版本号	发布时间
1	5X7 ISO8859 转 UNICODE 参考程 序	程序	<<ISO8859_UNICODE.c>>	V1.00	2011-1-12

## 5 引脚描述与电路连接

### 5.1 引脚配置

#### SOP8-A



### 5.2 引脚描述

#### SOP8-A

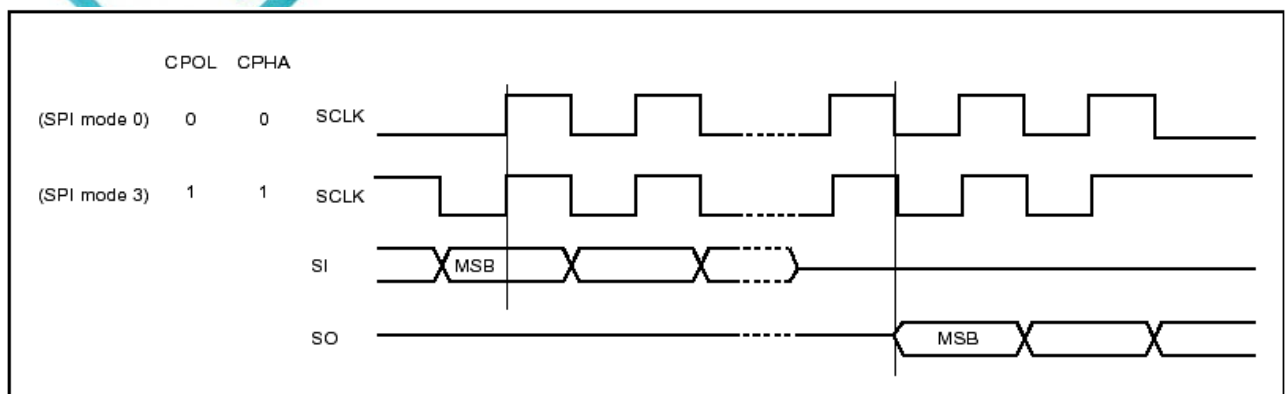
NO.	名称	I/O	描述
1	CS#	I	片选输入 (Chip enable input)
2	SO	O	串行数据输出 (Serial data output)
3	NC		悬空
4	GND		地(Ground)
5	SI	I	串行数据输入 (Serial data input)
6	SCLK	I	串行时钟输入 (Serial clock input)
7	HOLD#	I	总线挂起 (Hold, to pause the device without)
8	VCC		电源(+ 3.3V Power Supply)

**串行数据输出 (SO):** 该信号用来把数据从芯片串行输出, 数据在时钟的下降沿移出。

**串行数据输入 (SI):** 该信号用来把数据从串行输入芯片, 数据在时钟的上升沿移入。

**串行时钟输入 (SCLK):** 数据在时钟上升沿移入, 在下降沿移出。

**片选输入 (CS#):** 所有串行数据传输开始于CS#下降沿, CS#在传输期间必须保持为低电平, 在两条指令之间保持为高电平。



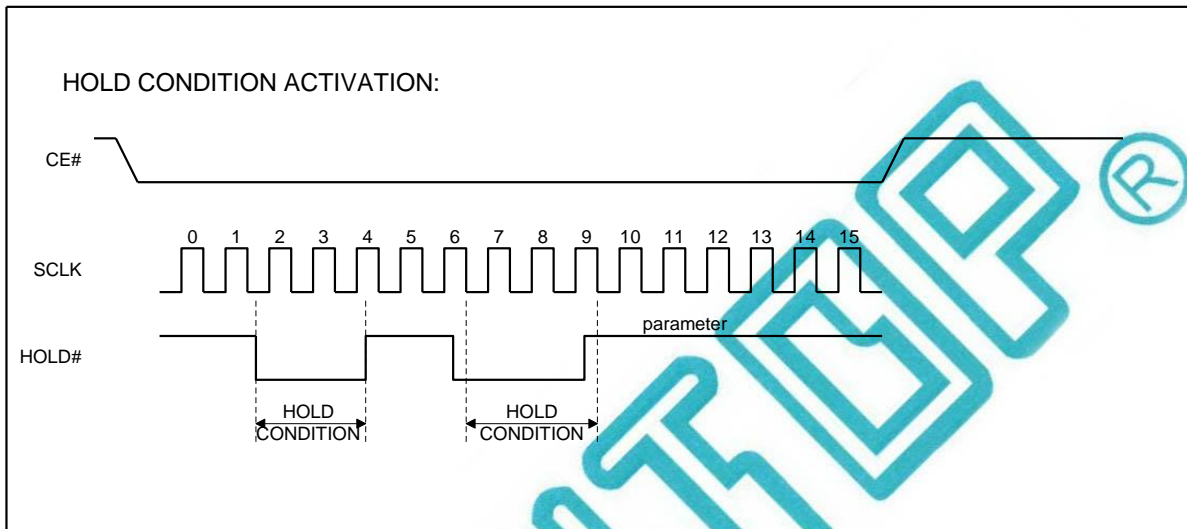


**总线挂起输入 (HOLD#):**

该信号用于片选信号有效期间暂停数据传输，在总线挂起期间，串行数据输出信号处于高阻态，芯片不对串行数据输入信号和串行时钟信号进行响应。

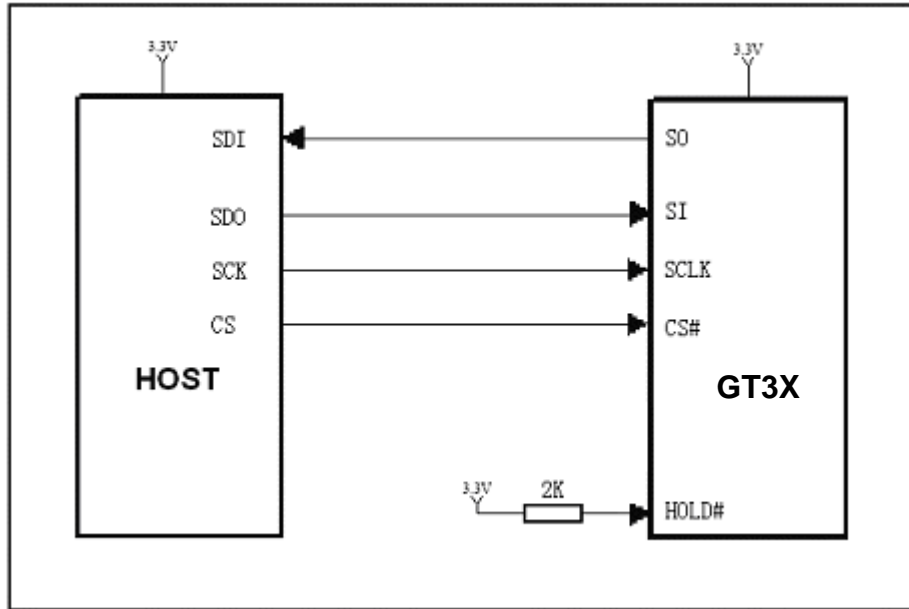
当HOLD#信号变为低并且串行时钟信号 (SCLK) 处于低电平时，进入总线挂起状态。

当HOLD#信号变为高并时串行时钟信号 (SCLK) 处于低电平时，结束总线挂起状态。



### 5.3 SPI 接口与主机接口参考电路示意图

SPI 与主机接口电路连接可以参考下图（#HOLD 管脚建议接 2K 电阻 3.3V 拉高）。



SPI 接口与主机接口参考电路示意图

## 6 电气特性

### 6.1 绝对最大额定值

Symbol	Parameter	Min.	Max.	Unit	Condition
T <sub>OP</sub>	Operating Temperature	-40	85	°C	
T <sub>STG</sub>	Storage Temperature	-65	150	°C	
VCC	Supply Voltage	-0.3	3.6	V	
V <sub>IN</sub>	Input Voltage	-0.3	VCC+0.3	V	
GND	Power Ground	-0.3	0.3	V	

### 6.2 DC 特性

Condition: T<sub>OP</sub> = -20°C to 70°C, GND=0V

Symbol	Parameter	Min.	Max.	Unit	Condition
I <sub>DD</sub>	VCC Supply Current(active)		12	mA	VCC=2.2~3.6V
I <sub>SB</sub>	VCC Standby Current		10	uA	
V <sub>IL</sub>	Input LOW Voltage	-0.3	0.3VCC	V	
V <sub>IH</sub>	Input HIGH Voltage	0.7VCC	VCC+0.4	V	
V <sub>OL</sub>	Output LOW Voltage		0.4 (I <sub>OL</sub> =1.6mA)	V	
V <sub>OH</sub>	Output HIGH Voltage	0.8VCC (I <sub>OH</sub> =-100uA)		V	
I <sub>LI</sub>	Input Leakage Current	0	2	uA	
I <sub>LO</sub>	Output Leakage Current	0	2	uA	

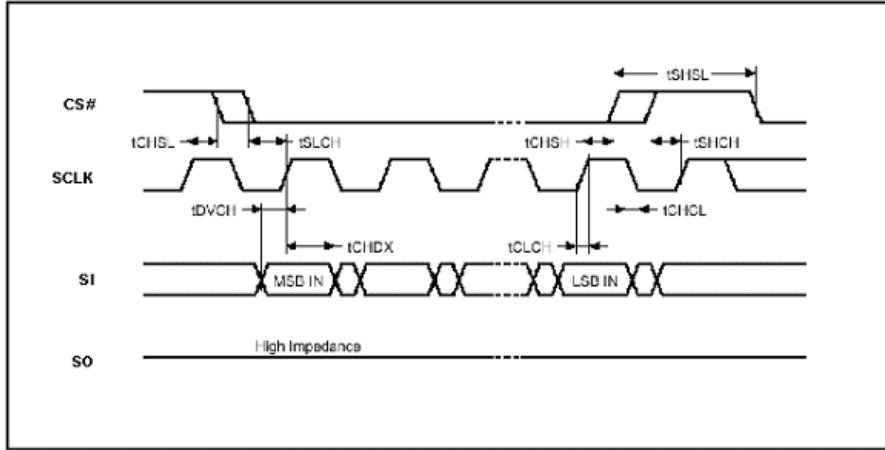
Note: I<sub>IL</sub>: Input LOW Current, I<sub>IH</sub>: Input HIGH Current,  
 I<sub>OL</sub>: Output LOW Current, I<sub>OH</sub>: Output HIGH Current,

### 6.3 AC 特性

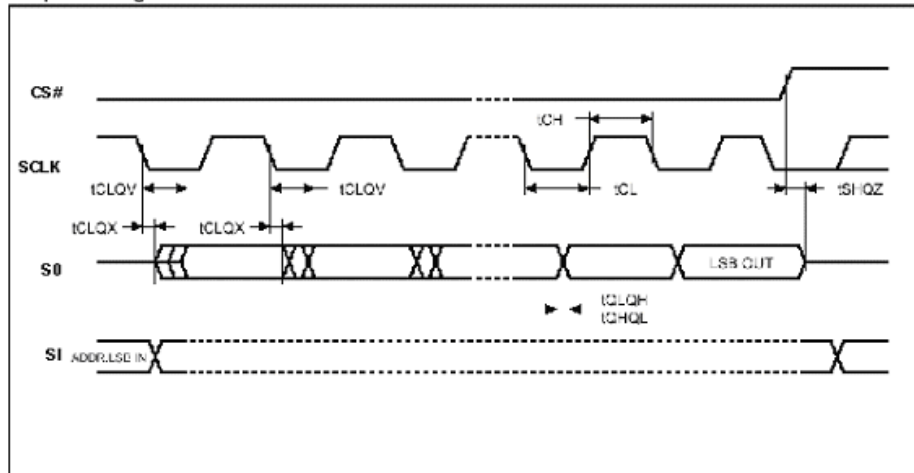
Symbol	Alt.	Parameter	Min.	Max.	Unit
F <sub>c</sub>	F <sub>c</sub>	Clock Frequency	D.C.	90	MHz
t <sub>CH</sub>	t <sub>CLH</sub>	Clock High Time	15		ns
t <sub>CL</sub>	t <sub>CLL</sub>	Clock Low Time	15		ns
t <sub>CLCH</sub>		Clock Rise Time(peak to peak)	0.1		V/ns
t <sub>CHCL</sub>		Clock Fall Time (peak to peak)	0.1		V/ns
t <sub>SLCH</sub>	t <sub>CSS</sub>	CS# Active Setup Time (relative to SCLK)	5		ns
t <sub>CHSL</sub>		CS# Not Active Hold Time (relative to SCLK)	5		ns
t <sub>DVCH</sub>	t <sub>DSU</sub>	Data In Setup Time	2		ns
t <sub>CHDX</sub>	t <sub>DH</sub>	Data In Hold Time	5		ns
t <sub>CHSH</sub>		CS# Active Hold Time (relative to SCLK)	5		ns
t <sub>SHCH</sub>		CS# Not Active Setup Time (relative to SCLK)	5		ns
t <sub>SHSL</sub>	t <sub>CSH</sub>	CS# Deselect Time	100		ns
t <sub>SHQZ</sub>	t <sub>DIS</sub>	Output Disable Time		9	ns

$t_{CLQV}$	tv	Clock Low to Output Valid		9	ns
$t_{CLQX}$	tho	Output Hold Time	0		ns

Serial Input Timing



Output Timing

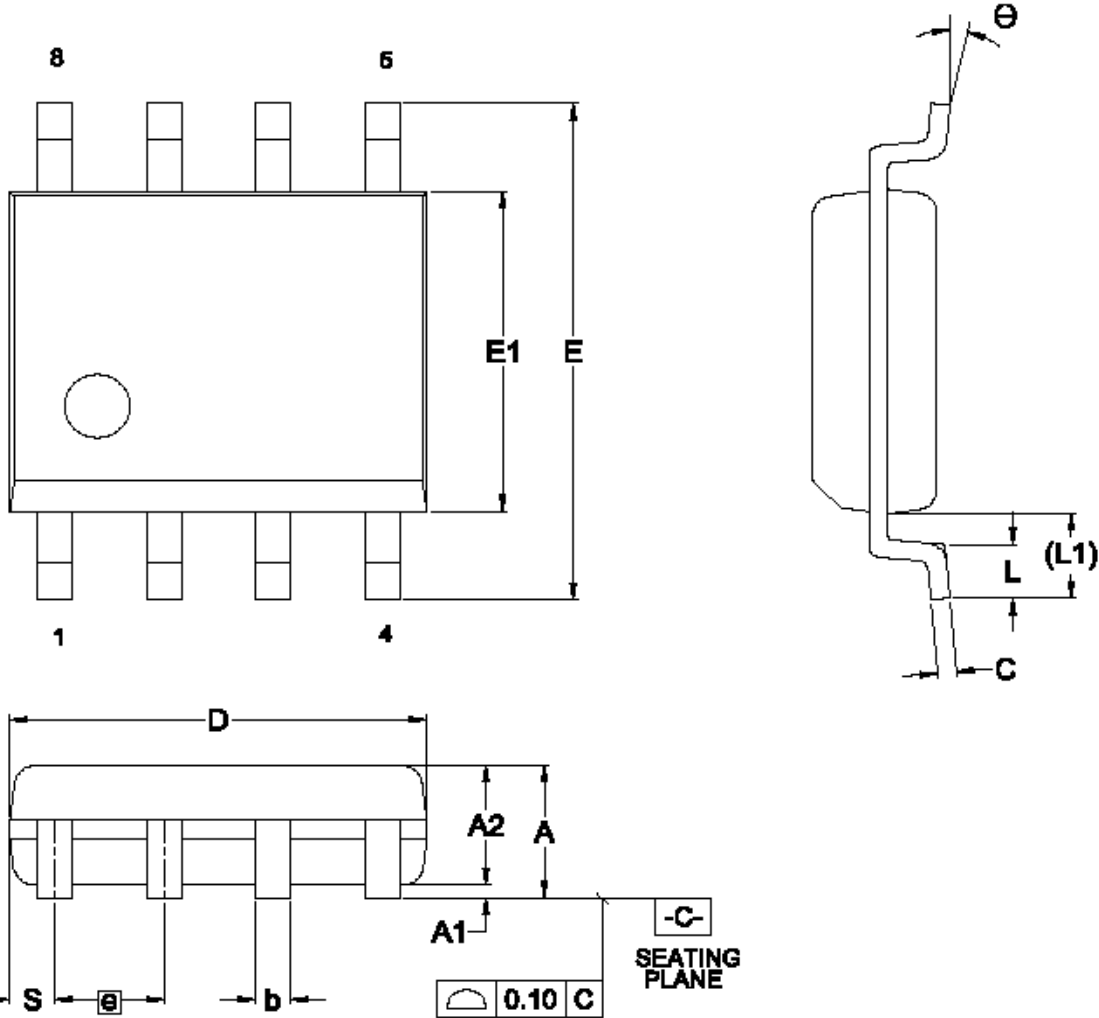


## 7 封装尺寸

封装类型	封装尺寸
SOP8-A	4.90mmX3.90mm (193milX154mil)

Package  
SOP8-A

Unit :mm



Dimensions (inch dimensions are derived from the original mm dimensions)

		A	A1	A2	b	C	D	E	E1	⊙	L	L1	S	θ
Mm	Min.	-	0.10	1.35	0.36	0.15	4.77	5.80	3.60		0.46	0.65	0.41	0
	Norm.	-	0.15	1.45	0.41	0.20	4.90	5.99	3.90	1.27	0.66	1.05	0.54	5
	Max.	1.75	0.20	1.55	0.51	0.25	5.03	6.20	4.00		0.86	1.25	0.67	8
inch	Min.	-	0.004	0.053	0.014	0.006	0.188	0.228	0.150		0.018	0.033	0.016	0
	Norm.	-	0.006	0.057	0.016	0.008	0.193	0.236	0.154	0.050	0.026	0.041	0.021	5
	Max.	0.069	0.008	0.061	0.020	0.010	0.198	0.244	0.156		0.034	0.049	0.026	8

## 8 附录

### 8.1 字型样张

#### 8.1.1 ASCII 字符

5x7 点阵 ASCII 标准字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

7x8 点阵 ASCII 标准字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

6x12 点阵 ASCII 标准字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

8x16 点阵 ASCII 标准字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

8x16 点阵 ASCII 粗体字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

9x24 点阵 ASCII 标准字符

Low 4bit / High 4bit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	



12x24 点阵 ASCII 标准字符

Low Gbit High Gbit	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

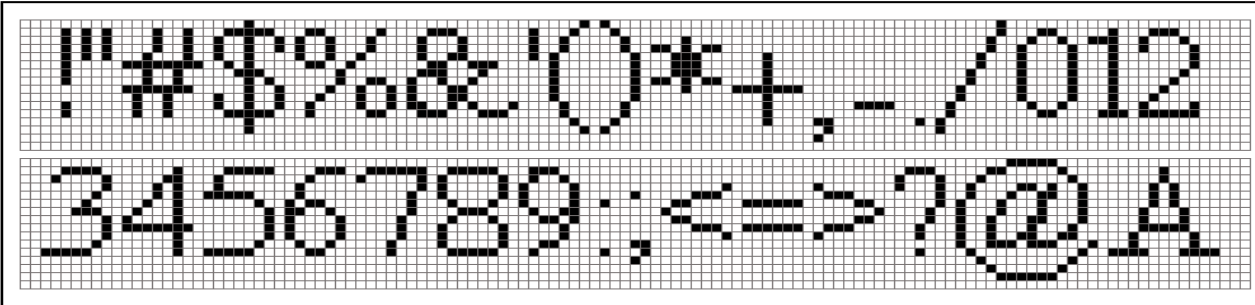
12 点阵不等宽 ASCII 白正(Time new Roman)

!"#\$%&'()\*+,-./01234567  
89:;<=>?@ABCDEFGHI  
JKLMNOPQRSTUVWXYZ

12 点阵不等宽 ASCII 方头 (Arial)

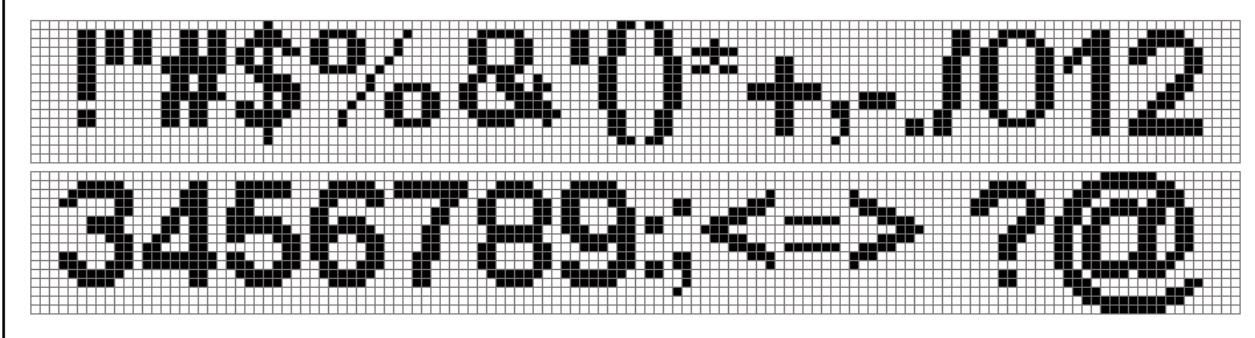
!"#\$%&'()\*+,-./0123456  
789:;<=>?@ABCDEFGH  
JKLMNOPQRSTUVWXYZ

16 点阵不等宽 ASCII 白正(Time new Roman)



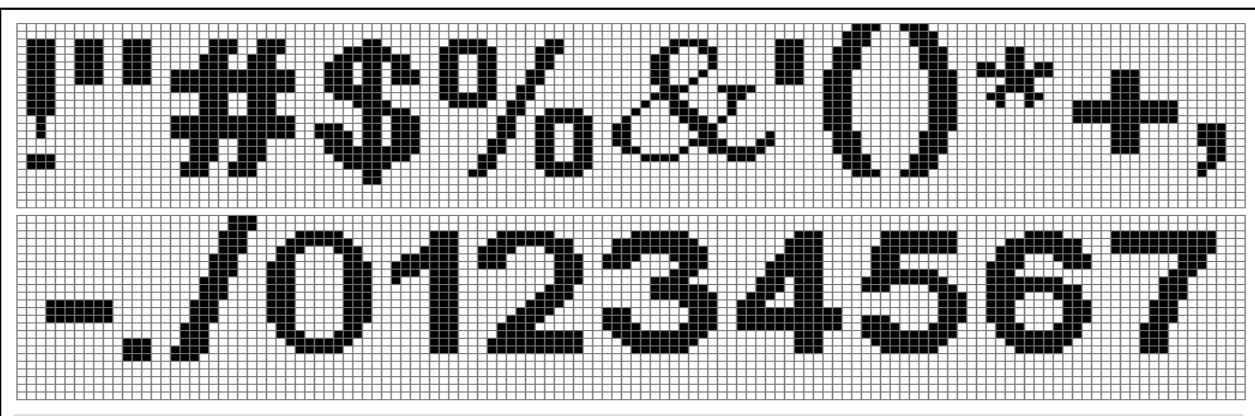
!"#\$%&'()\*+,-./012  
3456789:;<=>?@A

16 点阵不等宽 ASCII 方头 (Arial)



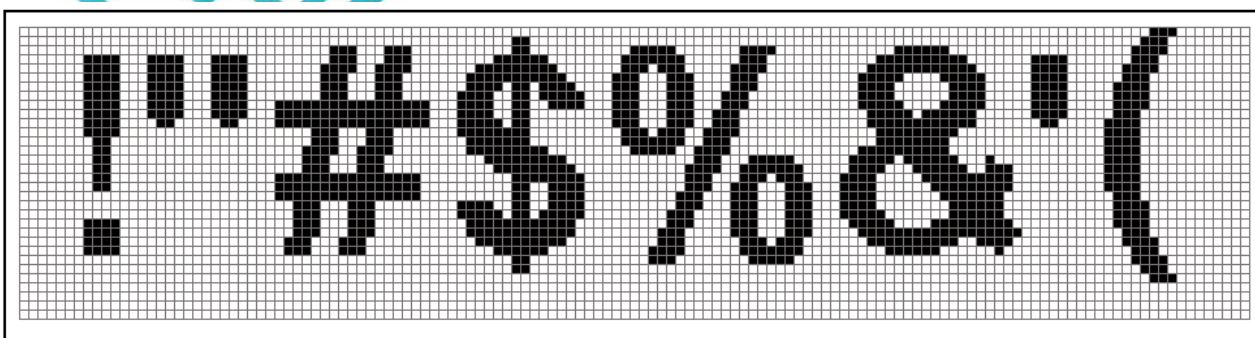
!"#\$%&'()\*+,-./012  
3456789:;<=>?@

24 点阵不等宽 ASCII 方头 (Arial)



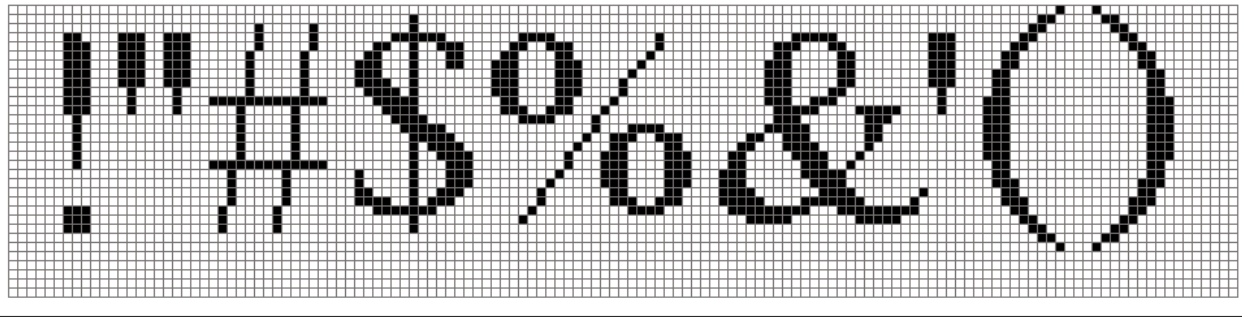
!"#\$%&'()\*+,-./01234567

32 点阵不等宽 ASCII 方头 (Arial)



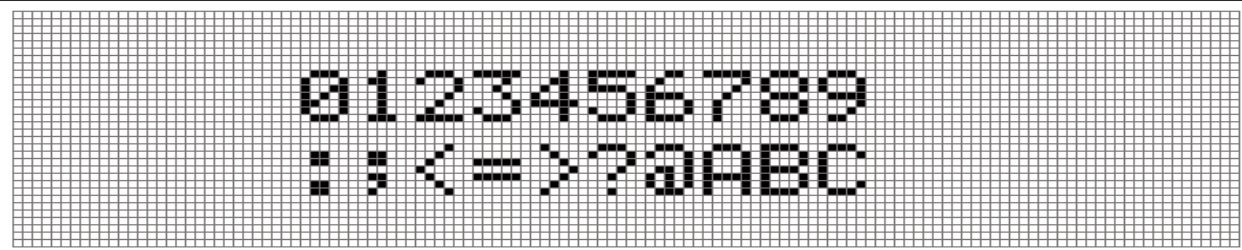
!"#\$%&'()

32 点阵不等宽 ASCII 白正 (Times New Roman)

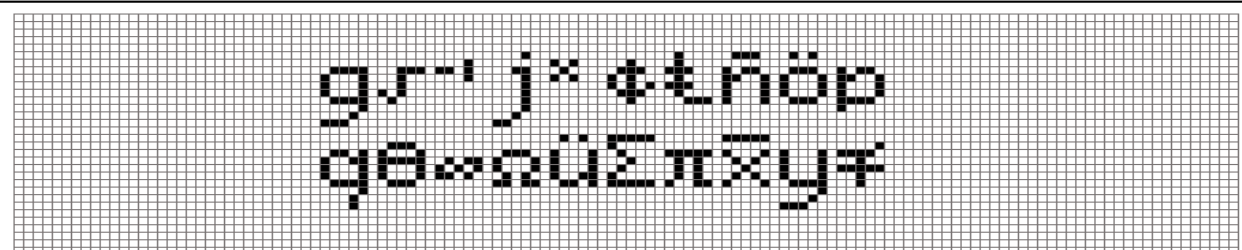


8.1.2 LCM 字符

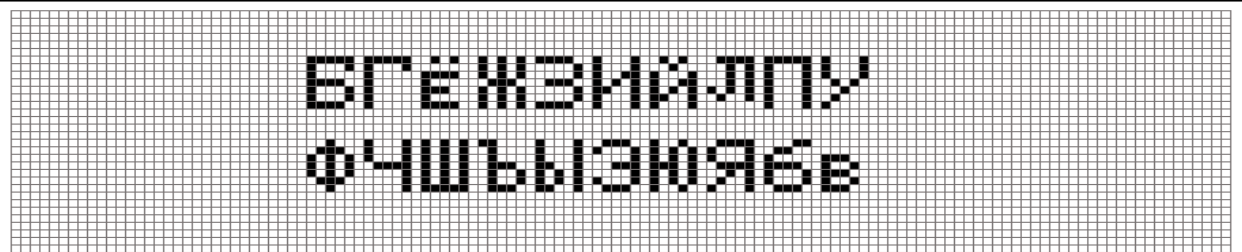
LCM0 5x7 点阵



LCM1 5x10 点阵



LCM2 5x10 点阵



LCM3 5x10 点阵

ВГЕЖЗИАЛПУ  
ФЧШЬЫІЭЮЯБВ

LCM4 5x10 点阵

ИЛ◊ККЭДРААА  
ААААЕЕЕЕЕЕІІ

LCM5 5x10 点阵

АААБВВІІІІІІІІІІ  
ККККЭЭУУУУУУУУУУ

LCM6 5x10 点阵

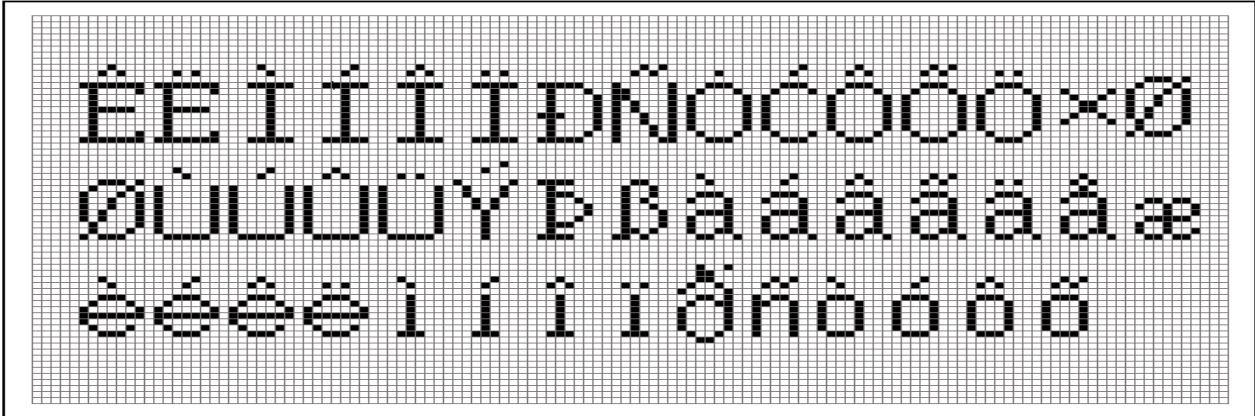
рqr'j\*φtñö  
pqθ\*ωΩUΣπXy

LCM7 5x10 点阵

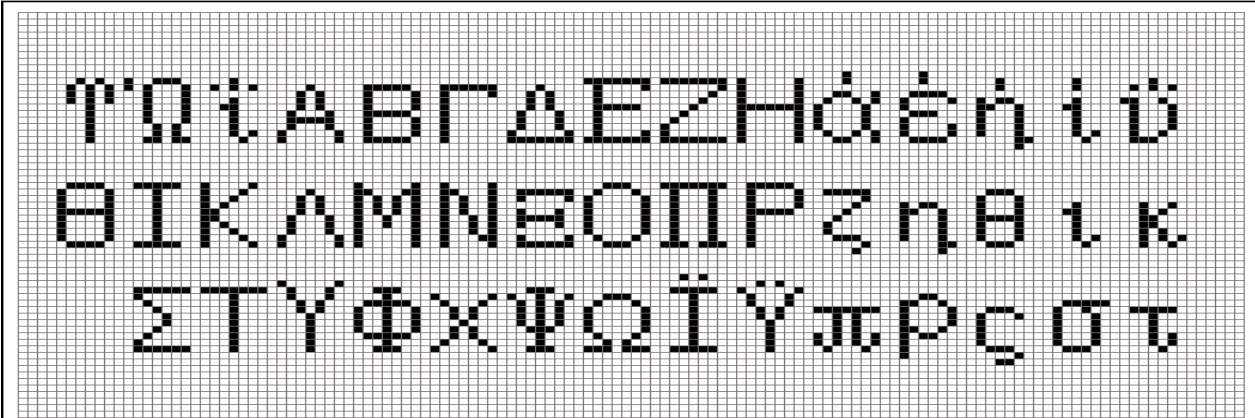
本ノヒワヘホマミム  
×モトヨウリルロ

8.1.3 UNICODE 字库

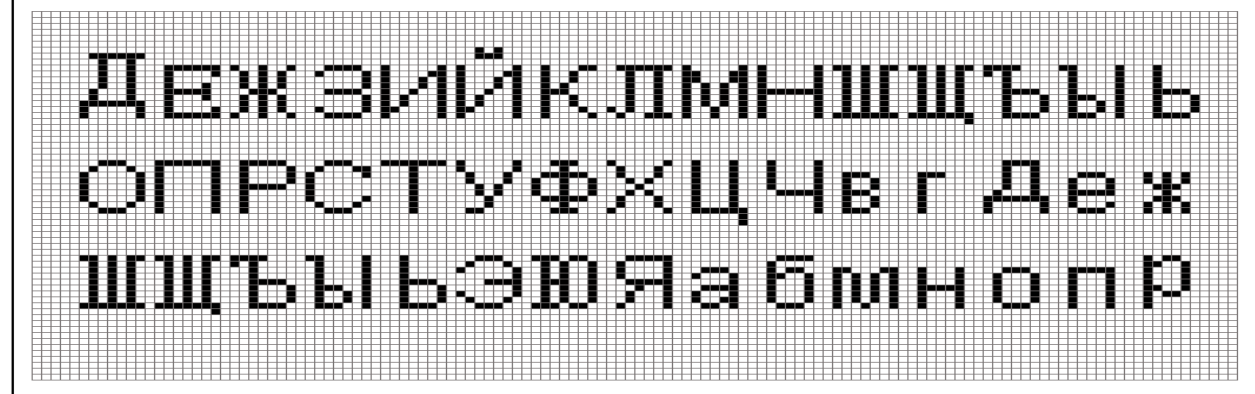
8x16 点阵拉丁文



8x16 点阵希腊文



8x16 点阵基里尔文



8x16 点阵希伯来文

אבגדה טו זח טי כלהמנ  
ספ קר שרע פז צק ןאבג

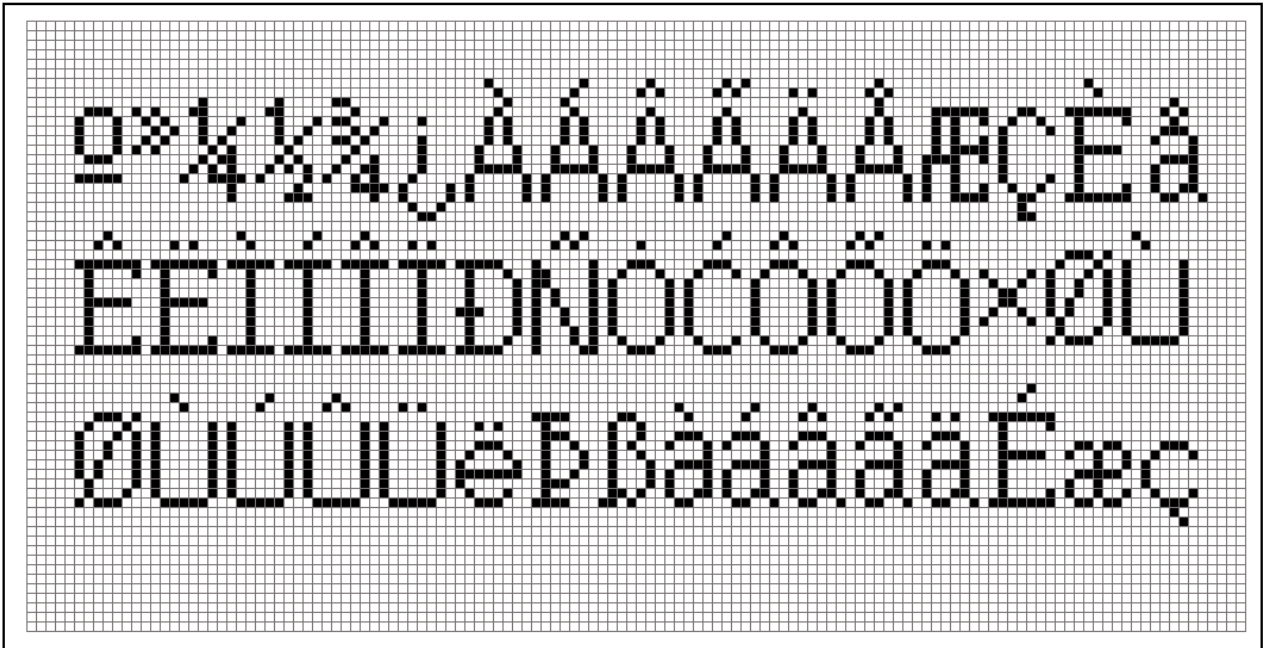
8x16 点阵泰文

ก ข ค ด อ อ ย ฟ ก ข ย ร ฌ  
ช ฉ ล ผ ฝ ห อ ฮ อ ฮ อ ฮ อ  
ต ถ ท ธ น บ ป ผ ฝ พ ฟ

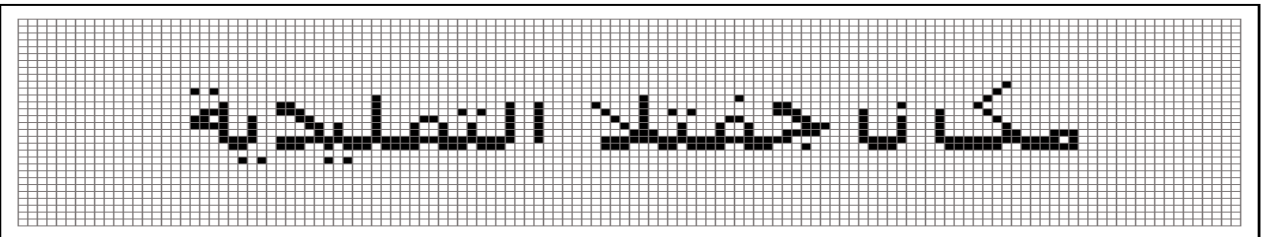
24 点阵不等宽泰文

ก ข ค ด อ อ ย ฟ ก ข ย ร ฌ  
ช ฉ ล ผ ฝ ห อ ฮ อ ฮ อ ฮ อ

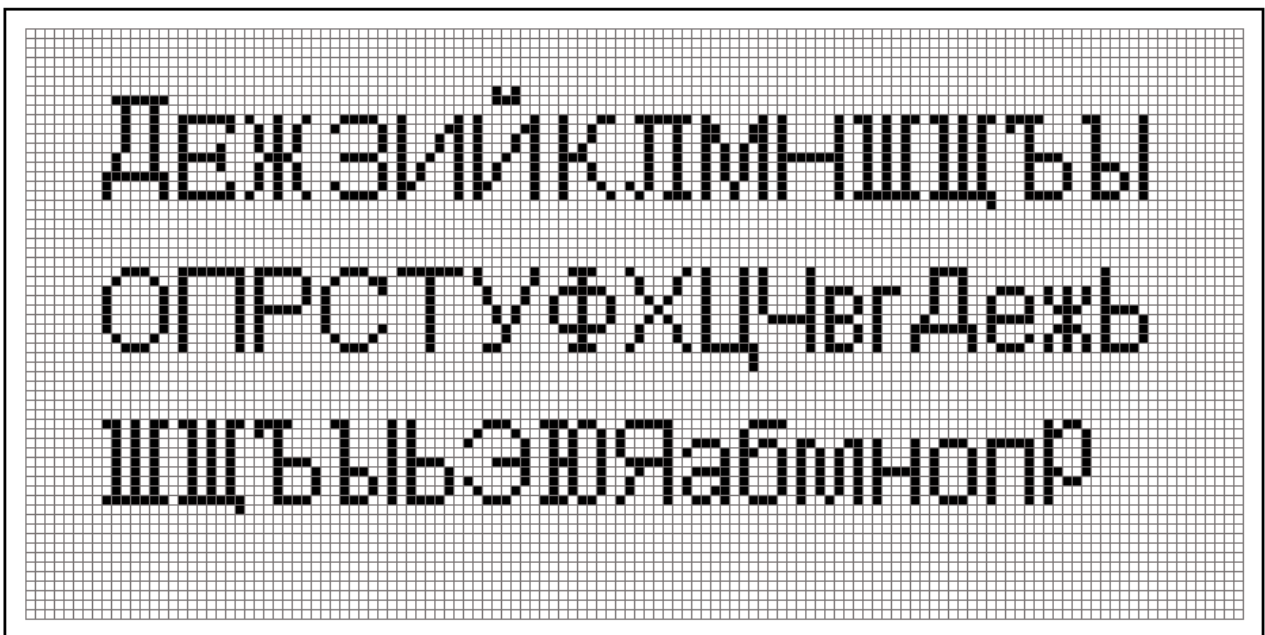
16 点阵不等宽拉丁文



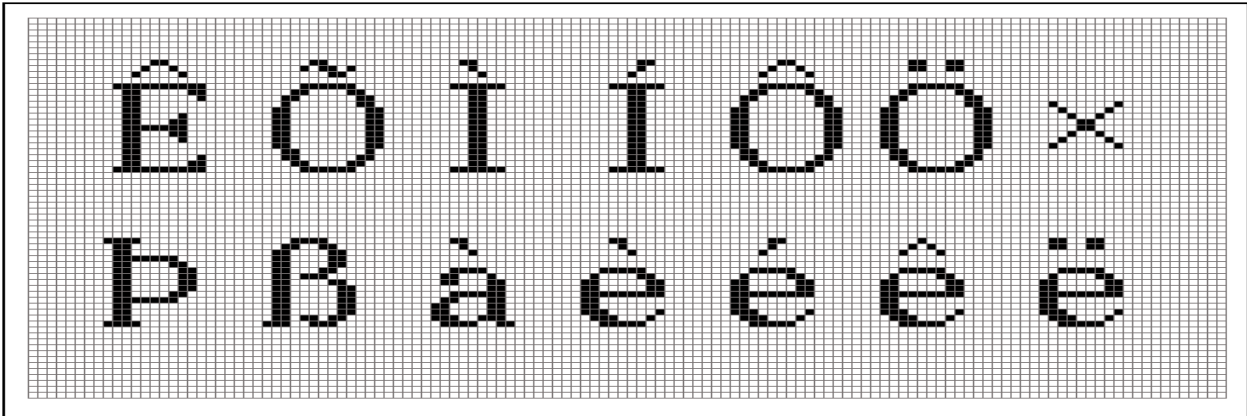
16 点阵不等宽阿拉伯文



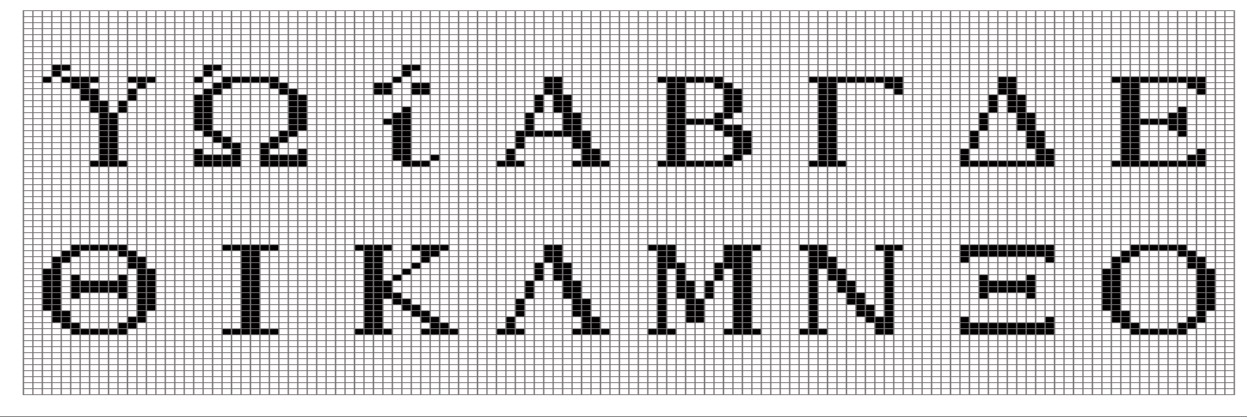
16 点阵不等宽基里尔文



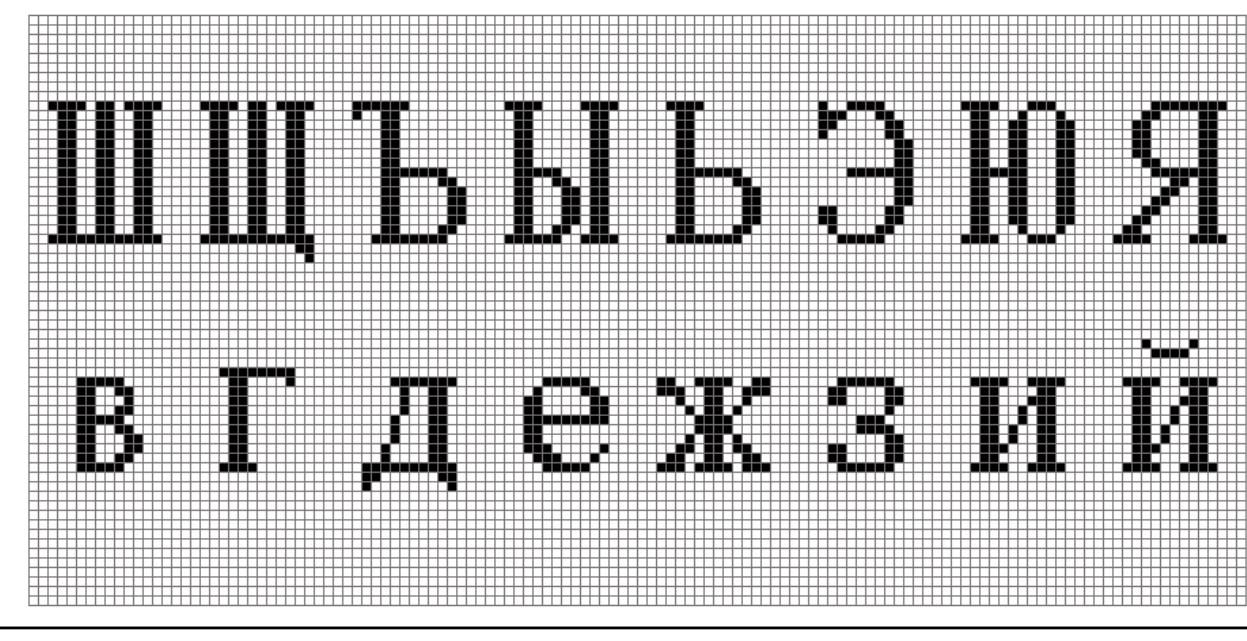
12x24 点阵拉丁文



12x24 点阵希腊文



12x24 点阵基里尔文







ISO8859-3

€ Ð Ž ŋ Ā Ā Ā Ā Ā Ā Ū Ū Ū  
 Ů Ů Č É Ě Ě Ě Ě Í Î ï ä å æ  
 Đ Ñ Ò Ķ Ô Õ Ö × Ø Ů ï ï đ



ISO8859-4

Ё Ж Э И Й К Л М Н О Г Д В  
 П Р С Т У Ф Х Ц Ч Ш Н О П  
 Щ Ъ Ы Ь Э Ю Я а б в ч ш щ



ISO8859-5

Ÿ Ÿ А В В Г Д В Ж Э Ь Э Ю  
 И Й К Л М Н О П Р С Ж Э И  
 Т У Ф Х Ц Ч Ш Щ Ъ Ы Р С Т

ISO8859-7

'EHI \*'O&'P'Q'IAΦΧΨ  
 ΒΓΔΕΖΗΘΙΚΛΑϚαβ  
 ΜΝΞΟΠΡ πΣΤΤ κ λ μ

ISO8859-8

ηĀĀĀĀĀĀĀĀĀĀĀĀĀĀĀĀ  
 ÉĚĚĚĚĚĚĚĚĚĚĚĚĚĚĚĚ  
 ÓÔÕÖ×ØÙÚÛÜŋōó

ISO8859-9

ĺĀĀĀĀĀĀĀĀĀĀĀĀĀĀĀĀ  
 ÉÊËĬĨĪĪĪĪĪĪĪĪĪĪĪĪĪ  
 ÓÔÕÖ×ØÙÚÛÜŋòóôõ

ISO8859-10

Å Ä Å Å Æ Ç È É Ê Ë à á â  
 ì í î ï ð Ñ Ò Ó Ô Õ è é ì  
 Ò Ø Ù Ú Û Ü Ý Þ ß ô õ

ISO8859-11

ì í î ï ð Ñ Ò Ó Ô Õ à á â  
 Ò × Ø Ù Ú Û Ü Ý Þ ß è é ì  
 à á â ã ä å æ ç è é ô õ

ISO8859-13

ì í î ï ð Ñ Ò Ó Ô Õ à á â  
 Ò Š Ÿ Ù Ú Û Ü Ě Ě ß è é ì  
 à á â ã ä é æ ç è é ô õ

ISO8859-14

Æ Ç È É Ê Ë à á â ã ä å æ  
 Đ Ñ Ò Ó Ô Õ ê ë ì í î ï ð  
 ù ú ý ß ô õ ö ÷ ø ù ú

ISO8859-15

À Ç è ç È É ê ë ì í â ã ä  
 î ñ ò Ñ Ò Ó ô õ ö ß ù ú ï  
 Ő Œ Ÿ à á ä å ß Ÿ

ISO8859-16

ÿ z À á ç â ã Á Ç Ò Ë Ï  
 È É Ê Ë ê ì í î ï Đ Ñ æ ç è  
 ò ó ô õ ö ù ú Ÿ ù ú ù ù đ ñ ò

8.1.5 CODE PAGE 字符

12X24 点阵 437--USA,Standard Europe (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢	£	¥	℔	ƒ
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	¬	½	¼	¡	«	»	
B	⌘	⌘	⌘		†	‡	‡	π	ƒ	‡		π	⌘	⌘	⌘	⌘
C	L	⌘	⌘	†	†	†	†	†	†	†	†	†	†	†	†	†
D	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
F	≡	±	≥	≤	∫	∫	÷	≈	°	•	•	√	∞	φ <sup>2</sup>	■	

437-Special graphic characters

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		☺	☹	♥	♦	♣	♠	•	◐	◑	♂	♀	♠	♠	♠	♠
1	▶	◀	↑	!!	¶	§	■	‡	↑	↓	→	←	↔	▲	▼	
7																△

12X24 点阵 437--USA,Standard Europe (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8	ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ÿ	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	ö	ü	¢	£	¥	℞	f
A	á	í	ó	ú	ñ	ñ	ª	º	¿	¡	½	¼	⅓	¼	«	»
B	⌘	⌘	⌘		†	‡	‡	π	₯	₯	∥	π	¼	¼	¼	¼
C	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
F	≡	±	≥	≤	∫	∫	÷	≈	°	•	•	√	n	²	■	

12X24 点阵 737—Greek (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	A	B	Γ	Δ	E	Z	H	Θ	I	K	Λ	M	N	Ξ	O	Π
9	P	Σ	T	Υ	Φ	X	Ψ	Ω	α	β	γ	δ	ε	ζ	η	θ
A	ι	κ	λ	μ	ν	ξ	ο	π	ρ	σ	ς	τ	υ	φ	χ	ψ
B	⌘	⌘	⌘		†	‡	‡	π	₯	₯	∥	π	¼	¼	¼	¼
C	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	ω	ά	έ	ή	ï	í	ó	ύ	ü	ώ	À	É	Η	Ì	Ó	Υ
F	Ω	±	≥	≤	ÿ	ÿ	÷	≈	°	•	•	√	n	²	■	

12X24 点阵 775—Baltic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ć	ü	é	ā	ä	ǰ	ǎ	ć	ł	ē	Ŕ	ŕ	ī	Ź	Ä	Å
9	É	æ	Æ	ō	ö	Ǧ	ç	Ś	ś	Ö	Ü	ø	ƒ	Ø	×	×
A	Ā	Ī	ó	Ź	ż	ż	”		©	®	¬	½	¼	Ł	«	»
B	⋮	⋮	⋮		†	Ą	Č	Ę	É	∥	∥	ŋ	∥	Į	Š	ŀ
C	Ł	⊥	⊥	†	—	†	Ū	Ū	Ł	ŋ	∥	∥	∥	=	∥	Ź
D	ą	č	ę	è	ı	š	ų	ū	ż	ł	ŀ	■	■	■	■	■
E	Ó	β	Ō	Ń	õ	Õ	μ	ń	Ŧ	ķ	Ł	ı	ŋ	Ē	Ņ	,
F	-	±	“	¾	¶	§	÷	”	°	•	.	¹	³	²	■	

12X24 点阵 775—Baltic (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ć	Ü	é	ā	ä	ǰ	ǎ	ć	ł	ē	Ŕ	ŕ	ī	Ź	Ä	Å
9	É	æ	Æ	ō	ö	Ǧ	ç	Ś	ś	Ö	Ü	ø	ƒ	Ø	×	×
A	Ā	Ī	ó	Ź	ż	ż	”		©	®	¬	½	¼	Ł	«	»
B	⋮	⋮	⋮		†	Ą	Č	Ę	É	∥	∥	ŋ	∥	Į	Š	ŀ
C	Ł	⊥	⊥	†	—	†	Ū	Ū	Ł	ŋ	∥	∥	∥	=	∥	Ź
D	ą	č	ę	è	ı	š	ų	ū	ż	ł	ŀ	■	■	■	■	■
E	Ó	β	Ō	Ń	õ	Õ	μ	ń	Ŧ	ķ	Ł	ı	ŋ	Ē	Ņ	,
F	-	±	“	¾	¶	§	÷	”	°	•	.	¹	³	²	■	

12X24 点阵 850—Multilingual (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	à	ǎ	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	ƒ	Ø	×	ƒ
A	á	í	ó	ú	ñ	Ñ	ª	°	¿	®	¬	½	¼	ı	«	»
B	⋮	⋮	⋮		†	Á	Â	À	©	∥	∥	ŋ	∥	ç	¥	ŀ
C	Ł	⊥	⊥	†	—	†	ã	Ă	Ł	ŋ	∥	∥	∥	=	∥	×
D	ð	Đ	Ê	Ë	È	ı	Í	Î	Ï	ł	ŀ	■	■	ı	İ	■
E	Ó	β	Ô	Ò	õ	Õ	μ	þ	Ɔ	Ú	Û	Ù	ÿ	Ý	—	,
F	-	±	=	¾	¶	§	÷	ˆ	°	¨	.	¹	³	²	■	



12X24 点阵 850—Multilingual (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	Ü	é	â	ä	à	ã	ç	ê	ë	è	ï	î	ï	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	ö	ü	ø	£	∅	×	f
A	á	í	ó	ú	ñ	ñ	ª	º	¿	®	¬	½	¼	¡	«	»
B	⌘	⌘	⌘		†	Á	Â	À	©	¶		¶	¶	¢	¥	¡
C	L	⊥	T	†	-	+	ã	Ã	ℓ	¶	≡	¶		=	¶	∞
D	ö	ð	ê	ë	è	í	î	ï	↓	↑	■	■	↓	↑	■	■
E	Ó	β	Ô	Ò	Õ	Ö	μ	ρ	ρ	Ú	Û	Ü	Ý	Ý	-	'
F	-	±	=	¾	π	§	÷	,	°	¨	.	¹	³	²	■	

12X24 点阵 852--Latin 2 (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	û	ć	ç	ł	ë	Ń	ń	î	ž	Ä	Ć
9	É	Ł	Í	ô	ö	Ł	ł	Ś	ś	Ö	Ü	Ť	ť	Ł	×	č
A	á	í	ó	ú	Ą	ą	Ź	ź	Ę	ę	¬	ż	Č	š	«	»
B	⌘	⌘	⌘		†	Á	Â	Ě	Ş	¶		¶	¶	ž	ž	¡
C	L	⊥	T	†	-	+	Ă	ă	ℓ	¶	≡	¶		=	¶	∞
D	đ	Đ	Ď	Ě	đ	Ń	Í	Î	ě	↓	↑	■	■	Ť	Ů	■
E	Ó	β	Ô	Ń	ń	ň	Š	š	Ř	Ú	ř	Ů	ý	Ý	ť	'
F	-	˝	˘	˘	˘	§	÷	,	°	¨	.	ú	Ř	ř	■	

12X24 点阵 852--Latin 2 (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	Ü	é	â	ä	û	ć	ç	ʃ	ë	ö	ó	î	ž	Ä	Ć
9	É	Ł	í	ô	ö	ł	ŧ	ś	ś	ö	ü	ř	ť	ł	×	č
A	á	í	ó	ú	À	à	Ž	ž	É	é	˘	ž	č	š	«	»
B	⌘	⌘	⌘		†	Á	Â	Ě	Š	ǃ		ŋ	ǂ	ž	ž	˘
C	L	⊥	⊥	†	—	†	Ǻ	ǻ	Š	ǃ	ǂ	ŋ	ǂ	=	ǃ	ǂ
D	đ	Đ	Ď	Ě	đ	Ň	Í	Î	ě	ǃ	ǂ	■	■	J	Û	■
E	Ó	ß	Ô	Ń	ń	ň	Š	š	Ř	Ú	ř	Ü	ý	ý	ı	˘
F	-	˘	˘	˘	˘	š	÷	˘	˘	˘	˘	ü	ř	ř	■	

12X24 点阵 855—Cyrillic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ђ	Ђ	ѓ	Ѓ	ё	Ё	є	Є	ѕ	Ѕ	і	І	ї	Ї	ј	Ј
9	љ	Љ	њ	Њ	ћ	Ћ	ќ	Ќ	ў	Ў	ц	Ц	ю	Ю	ъ	Ъ
A	а	А	б	Б	ц	Ц	д	Д	е	Е	ф	Ф	г	Г	«	»
B	⌘	⌘	⌘		†	х	Х	и	И	ǃ		ŋ	ǂ	й	Й	˘
C	L	⊥	⊥	†	—	†	к	К	л	Л	ǃ	ǂ	ŋ	ǂ	=	ǃ
D	л	Л	м	М	н	Н	о	О	п	П	ǃ	ǂ	■	■	П	я
E	Я	р	Р	с	С	т	Т	у	У	ж	Ж	в	В	ь	Ь	№
F	-	ы	Ы	з	З	ш	Ш	э	Э	щ	Щ	ч	Ч	§	■	

12X24 点阵 855—Cyrillic (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ђ	Ђ	ѓ	Ѓ	ё	Ё	є	Є	ѕ	Ѕ	і	І	ї	Ї	ј	Ј
9	љ	Љ	њ	Њ	ћ	Ћ	ќ	Ќ	ў	Ў	ц	Ц	ю	Ю	ъ	Ъ
A	а	А	б	Б	ц	Ц	д	Д	е	Е	ф	Ф	г	Г	«	»
B	⌘	⌘	⌘		†	х	Х	и	И	ǃ		ŋ	ǂ	й	Й	˘
C	L	⊥	⊥	†	—	†	к	К	л	Л	ǃ	ǂ	ŋ	ǂ	=	ǃ
D	л	Л	м	М	н	Н	о	О	п	П	ǃ	ǂ	■	■	П	я
E	Я	р	Р	с	С	т	Т	у	У	ж	Ж	в	В	ь	Ь	№
F	-	ы	Ы	з	З	ш	Ш	э	Э	щ	Щ	ч	Ч	§	■	

12X24 点阵 857—Turkish (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ı	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	İ	Ö	Ü	ø	£	Ø	Ş	ş
A	á	í	ó	ú	ñ	Ñ	Ğ	ğ	¿	®	¬	½	¼	ı	«	»
B	⌘	⌘	⌘		†	Á	Â	À	©	‡		¶	⌘	¢	¥	⌘
C	L	⊥	⊥	†	—	+	ã	Ã	ℒ	℞	±	π		=	≠	⊗
D	°	ª	Ê	Ë	È		Í	Î	Ï	↓	↑	■	■		ì	■
E	Ó	β	Ô	Ò	õ	Õ	μ		×	Ú	Û	Ù	ì	ÿ	—	,
F	-	±		¾	¶	§	÷	,	°	¨	.	¹	³	²	■	

12X24 点阵 857—Turkish (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	Ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ı	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	İ	Ö	Ü	ø	£	Ø	Ş	ş
A	á	í	ó	ú	ñ	Ñ	Ğ	ğ	¿	®	¬	½	¼	ı	«	»
B	⌘	⌘	⌘		†	Á	Â	À	©	‡		¶	⌘	¢	¥	⌘
C	L	⊥	⊥	†	—	+	ã	Ã	ℒ	℞	±	π		=	≠	⊗
D	°	ª	Ê	Ë	È		Í	Î	Ï	↓	↑	■	■		ì	■
E	Ó	β	Ô	Ò	õ	Õ	μ		×	Ú	Û	Ù	ì	ÿ	—	,
F	-	±		¾	¶	§	÷	,	°	¨	.	¹	³	²	■	

12X24 点阵 858—Euro (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	®	¬	½	¼	¡	«	»
B	⌘	⌘	⌘		†	Á	Â	À	©	¶		¶	¶	¢	¥	γ
C	L	⊥	⊥	⊥	—	+	ã	Ã	ℒ	π	≡	π		=	≠	∞
D	ð	Ð	Ê	Ë	È	€	Í	Î	Ï	↓	↑	■	■	—	Ì	■
E	Ó	β	Ô	Ò	õ	Õ	μ	ρ	ρ	Ú	Û	Ù	ý	Ý	—	'
F	-	±	=	¾	¶	§	÷	˘	°	¨	.	¹	³	²	■	

12X24 点阵 858—Euro (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	®	¬	½	¼	¡	«	»
B	⌘	⌘	⌘		†	Á	Â	À	©	¶		¶	¶	¢	¥	γ
C	L	⊥	⊥	⊥	—	+	ã	Ã	ℒ	π	≡	π		=	≠	∞
D	ð	Ð	Ê	Ë	È	€	Í	Î	Ï	↓	↑	■	■	—	Ì	■
E	Ó	β	Ô	Ò	õ	Õ	μ	ρ	ρ	Ú	Û	Ù	ý	Ý	—	'
F	-	±	=	¾	¶	§	÷	˘	°	¨	.	¹	³	²	■	

12X24 点阵 860—Portuguese (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ã	à	Á	ç	ê	Ê	è	Í	Ô	ì	Ã	Â
9	É	À	È	ô	õ	ò	Ú	ù	Ì	Õ	Ü	¢	£	Ù	℞	Ó
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	Ò	¬	½	¼	¡	«	»
B	⌘	⌘	⌘		†	‡		π	¶	¶		¶	¶		¶	γ
C	L	⊥	⊥	⊥	—	+	†		ℒ	π	≡	π		=	≠	±
D	≡	π	π	≡	≡	F	π	≠	↓	↑	■	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
F	≡	±	≥	≤	∫	J	÷	≈	°	•	•	√	n	²	■	

12X24 点阵 860—Portuguese (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	Ü	É	â	ã	à	Á	Ç	Ê	Ê	È	Í	Ô	Ï	Ã	Â
9	É	À	È	Ô	Õ	Ò	Ú	Ù	Ì	Õ	Ü	Ç	£	Ù	ß	Ó
A	á	í	ó	ú	ñ	ñ	ª	º	¿	Ò	¬	½	¼	¡	«	»
B	☼	☼	☼		†	‡	‡	π	¶	¶		π	⊥	⊥	⊥	⊥
C	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	ο	μ	τ	Φ	Θ	Ω	δ	ω	φ	ε	∩
F	≡	±	≥	≤	∫	∫	÷	≈	°	•	.	√	n	²	■	

12X24 点阵 862—Hebrew (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ז	ב	ג	ד	ה	ו	ז	ח	ט	י	ך	כ	ל	מ	נ	ס
9	ז	ב	ג	ד	ה	ו	ז	ח	ט	י	ך	כ	ל	מ	נ	ס
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	Ò	¬	½	¼	¡	«	»
B	☼	☼	☼		†	‡	‡	π	¶	¶		π	⊥	⊥	⊥	⊥
C	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	ο	μ	τ	Φ	Θ	Ω	δ	ω	φ	ε	∩
F	≡	±	≥	≤	∫	∫	÷	≈	°	•	.	√	n	²	■	

12X24 点阵 863--Canadian French (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	Â	à	¶	ç	ê	ë	è	ï	î	=	À	§
9	É	È	Ê	ô	Ë	ï	û	ù	ø	Ô	Ü	ø	£	Ù	Û	f
A		'	ó	ú	¨	´	°	—	î	ˆ	˘	½	¼	¾	«	»
B	▒	▒	▒		†	‡	‡	¶	¶	¶	¶	¶	¶	¶	¶	¶
C	L	⊥	⊥	⊥	—	†	‡	‡	‡	‡	‡	‡	‡	‡	‡	‡
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	⊞	φ	ε	∩
F	≡	±	≥	≤	∫	J	÷	≈	°	•	·	√	n	²	■	

12X24 点阵 863--Canadian French (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	Â	à	¶	ç	ê	ë	è	ï	î	=	À	§
9	É	È	Ê	ô	Ë	ï	û	ù	ø	Ô	Ü	ø	£	Ù	Û	f
A		'	ó	ú	¨	´	°	—	î	ˆ	˘	½	¼	¾	«	»
B	▒	▒	▒		†	‡	‡	¶	¶	¶	¶	¶	¶	¶	¶	¶
C	L	⊥	⊥	⊥	—	†	‡	‡	‡	‡	‡	‡	‡	‡	‡	‡
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	⊞	φ	ε	∩
F	≡	±	≥	≤	∫	J	÷	≈	°	•	·	√	n	²	■	

12X24 点阵 864—Arabic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	°	•	•	√	▒	—		†	†	⊥	⊥	⊥	⊥	⊥	⊥	⊥
9	β	⊞	φ	±	½	¼	≈	«	»	لا	لا			لا	لا	
A		-	آ	£	ø	أ		ا	ب	ت	ث	،	ج	ح	خ	خ
B	•	ا	ب	ج	د	هـ	و	ز	ح	ط	ظ	ع	ف	غ	ص	؟
C	ø	ع	آ	أ	ؤ	ح	ن	ا	ب	ة	ت	ث	ج	ح	خ	د
D	ذ	ر	ز	ط	ظ	ظ	ظ	ظ	ظ	ظ	ظ	ظ	ظ	ظ	ظ	ع
E	•	ف	ق	ك	ل	م	ن	هـ	و	ي	ي	ض	ع	غ	غ	م
F	•	="	ن	هـ	هـ	ي	ي	غ	ق	لا	لا	ل	ك	بي	■	

12X24 点阵 865—Nordic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	ℙ	f
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	¬	½	¼	¡	«	»	
B	⌘	⌘	⌘		†	‡	‡	π	ƒ	‡	‡	π	‡	‡	‡	‡
C	L	⊥	⊥	⊥	—	+	†	‡	‡	‡	‡	‡	‡	‡	‡	‡
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	θ	φ	ε	∩
F	≡	±	≥	≤	∫	J	÷	≈	°	•	·	√	η	²	■	

12X24 点阵 865—Nordic (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ç	Ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	ℙ	f
A	á	í	ó	ú	ñ	Ñ	ª	º	¿	¬	½	¼	¡	«	»	
B	⌘	⌘	⌘		†	‡	‡	π	ƒ	‡	‡	π	‡	‡	‡	‡
C	L	⊥	⊥	⊥	—	+	†	‡	‡	‡	‡	‡	‡	‡	‡	‡
D	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	■	■	■	■	■
E	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	θ	φ	ε	∩
F	≡	±	≥	≤	∫	J	÷	≈	°	•	·	√	η	²	■	

12X24 点阵 866--Cyrillic 2 (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
9	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
A	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
B	⌘	⌘	⌘		†	‡	‡	π	¶	‡		π	⌘	⌘	‡	‡
C	L	⊥	⊥	⊥	—	†	‡	‡	⊥	π	⊥	π	‡	=	‡	⊥
D	⊥	π	π	⊥	⊥	π	π	‡	⊥	π	■	■	■	■	■	■
E	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F	Ë	ë	€	€	Ï	ï	Û	û	°	•	·	√	№	∞	■	

12X24 点阵 866--Cyrillic 2 (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
9	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
A	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
B	⌘	⌘	⌘		†	‡	‡	π	¶	‡		π	⌘	⌘	‡	‡
C	L	⊥	⊥	⊥	—	†	‡	‡	⊥	π	⊥	π	‡	=	‡	⊥
D	⊥	π	π	⊥	⊥	π	π	‡	⊥	π	■	■	■	■	■	■
E	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
F	Ë	ë	€	€	Ï	ï	Û	û	°	•	·	√	№	∞	■	

12X24 点阵 1251—Cyrillic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ъ	Ѓ	,	ѓ	„	...	†	‡	€	%	Љ	<	Њ	Ќ	Ѝ	Ў
9	ђ	‘	’	“	”	•	—	—		™	љ	>	њ	ќ	ћ	џ
A		ÿ	ÿ	Ј	∞	Г		§	Ё	©	€	<<	¬	-	®	İ
B	°	±	І	і	г	μ	¶	·	ё	№	е	>>	ј	ѕ	ѕ	ї
C	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
D	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
E	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
F	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я



12X24 点阵 1251—Cyrillic (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Ъ	Ґ	ґ	ґ	”	...	†	‡	€	‰	Љ	<	Њ	Ќ	Ћ	Ў
9	ђ	‘	’	“	”	•	-	-		™	љ	>	њ	ќ	ћ	ў
A		ÿ	ÿ	Ј	Ѡ	Г	І	Ѕ	Ё	©	Є	«	¬	-	®	İ
B	°	±	І	і	Г	μ	π	·	ё	№	Е	»	ј	Ѕ	Ѕ	İ
C	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
D	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
E	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
F	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я

12X24 点阵 1252--Latin I (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡	^	‰	Š	<	œ		Ž	
9		‘	’	“	”	•	-	-	~	™	š	>	œ		ž	ÿ
A		ı	ç	£	¥	ı	§	”	”	©	ª	«	¬	-	®	—
B	°	±	²	³	´	μ	π	·	˘	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

12X24 点阵 1252--Latin I (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡	^	‰	Š	<	Œ		Ž	
9		‘	’	“	”	•	-	-	~	™	š	>	œ		ž	ÿ
A		ı	ç	£	¤	¥		§	¨	©	ª	«	¬	-	®	-
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

12X24 点阵 1253—Greek (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡		‰		<				
9		‘	’	“	”	•	-	-		™		>				
A		“	À	£	¤	¥		§	¨	©		«	¬	-	®	-
B	°	±	²	³	´	µ	¶	·	È	Η	Ι	»	Ό	½	Υ	Ω
C	ί	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
D	Π	Ρ		Σ	Τ	Υ	Φ	Χ	Ψ	Ω	Ϊ	Ϋ	ά	έ	ή	ί
E	ύ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
F	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ϊ	ϋ	ό	ύ	ώ	

12X24 点阵 1254—Turkish (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡	^	‰	Š	<	Œ			
9		‘	’	“	”	•	-	-	~	™	š	>	œ			ÿ
A		ı	ç	£	¤	¥		§	¨	©	ª	«	¬	-	®	-
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ğ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	İ	Ş	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ğ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı	ş	ÿ

12X24 点阵 1254—Turkish (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡	^	‰	Š	<	Œ			
9		‘	’	“	”	•	-	-	~	™	š	>	œ			ÿ
A		ı	ç	£	¤	¥		§	¨	©	ª	«	¬	-	®	—
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ğ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	İ	Ş	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ğ	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı	ş	ÿ

12X24 点阵 1255--Hebrew New (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		,	f	”	...	†	‡	^	‰		<				
9		‘	’	“	”	•	-	-	~	™		>				
A		ı	ç	£	¤	¥		§	¨	©	×	«	¬	-	®	—
B	°	±	²	³	´	µ	¶	·	¸	¹	÷	»	¼	½	¾	¿
C	:	;	:	:	.	..	ˆ	˜	˘	.	˙	.	.	.	.	.
D		·	·	:				ˆ	˜	˘	˙	.	.	.	.	.
E	א	ב	ג	ד	ה	ו	ז	ח	ט	י	ך	כ	ל	ם	מ	ן
F	נ	ו	ז	ח	ט	י	כ	ל	מ	נ	ס	ע	פ	צ	ק	ר

12X24 点阵 1256—Arabic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€	‚	‚	ƒ	”	…	†	‡	ˆ	‰	£	<	œ	ç	ž	š
9	€	‚	‚	“	”	•	-	-	ˆ	™	£	>	œ			š
A		•	¢	£	¤	¥		§	¨	©	®	«	¬	-	®	—
B	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C	ˆ	€	Ā	Á	Â	Ã	Ä	Å	Ā	É	Ê	Ë	Ĝ	Ķ	Ī	Ļ
D	ˆ	€	Ā	Á	Â	Ã	Ä	Å	Ā	É	Ê	Ë	Ĝ	Ķ	Ī	Ļ
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ˆ	€	Ā	Á	Â	Ã	Ä	Å	Ā	É	Ê	Ë	Ĝ	Ķ	Ī	Ļ

12X24 点阵 1257—Baltic (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		‚		”	…	†	‡		‰		<		¨	˘	˙
9		‚	‚	“	”	•	-	-		™		>		—	˘	˙
A			¢	£	¤			§	Ø	©	Ŕ	«	¬	-	®	Æ
B	°	±	²	³	´	µ	¶	·	ø	¹	ŕ	»	¼	½	¾	æ
C	Ą	Į	Ā	Ć	Ä	Å	Ę	Ē	Č	É	Ż	È	Ğ	Ķ	Ī	Ļ
D	Š	Ń	Ņ	Ó	Ō	Õ	Ö	×	Ū	Ł	Ś	Ū	Ū	Ž	Ž	ß
E	ą	į	ā	ć	ä	å	ę	ē	č	é	ż	è	ğ	ķ	ī	ļ
F	š	ń	ņ	ó	ō	õ	ö	÷	ų	ł	ś	ū	ū	ž	ž	·

12X24 点阵 1257—Baltic (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	€		‚		”	…	†	‡		‰		<		¨	˘	˙
9		‚	‚	“	”	•	-	-		™		>		—	˘	˙
A			¢	£	¤			§	Ø	©	Ŕ	«	¬	-	®	Æ
B	°	±	²	³	´	µ	¶	·	ø	¹	ŕ	»	¼	½	¾	æ
C	Ą	Į	Ā	Ć	Ä	Å	Ę	Ē	Č	É	Ż	È	Ğ	Ķ	Ī	Ļ
D	Š	Ń	Ņ	Ó	Ō	Õ	Ö	×	Ū	Ł	Ś	Ū	Ū	Ž	Ž	ß
E	ą	į	ā	ć	ä	å	ę	ē	č	é	ż	è	ğ	ķ	ī	ļ
F	š	ń	ņ	ó	ō	õ	ö	÷	ų	ł	ś	ū	ū	ž	ž	·

12X24 点阵 928—Greek (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		`	´	£				§	¨	©		«	¬			¯
B	°	±	²	³	´	µ	Α	·	Ε	Η	Ι	»	Ό	½	Υ	Ω
C	İ	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
D	Π	Ρ		Σ	Τ	Υ	Φ	Χ	Ψ	Ω	İ	ÿ	Α	Ε	Η	ί
E	ύ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
F	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ï	ü	ó	ú	ώ	

12X24 点阵 Hebrew old (标准)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	א	ב	ג	ד	ה	ו	ז	ח	ט	י	ך	כ	ל	מ	נ	ן
7	נ	ו	ז	ח	ט	י	ך	כ	ל	מ	נ	{		}	~	

12X24 点阵 International character (标准)

Country	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U. S. A.	#	\$	@	[	\	]	^	‘	{		}	~
France	#	\$	à	°	Ç	§	^	‘	é	Ù	è	¨
Germany	#	\$	§	Ä	Ö	Ü	^	‘	ä	ö	ü	ß
U. K.	£	\$	@	[	\	]	^	‘	{		}	~
Denmark I	#	\$	@	Æ	Ø	Å	^	‘	æ	ø	å	~
Sweden	#	*	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italy	#	\$	@	°	\	é	^	ù	à	ò	è	ì
Spain	Pt	\$	@	ı	Ñ	ı	^	‘	¨	ñ	}	~
Japan	#	\$	@	[	¥	]	^	‘	{		}	~
Norway	#	*	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Denmark II	#	\$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü

12X24 点阵 International character (打印体)

Country	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U. S. A.	#	\$	@	[	\	]	^	‘	{		}	~
France	#	\$	à	°	Ç	Š	^	‘	é	Ù	è	..
Germany	#	\$	Š	Ä	Ö	Ü	^	‘	ä	Ö	ü	ß
U. K.	£	\$	@	[	\	]	^	‘	{		}	~
Denmark I	#	\$	@	Æ	Ø	Å	^	‘	æ	ø	å	~
Sweden	#	¤	É	Ä	Ö	Å	Ü	é	ä	Ö	å	ü
Italy	#	\$	@	°	\	é	^	ù	à	ò	è	ì
Spain	¤	\$	@	í	ñ	¿	^	‘	·	ñ	}	~
Japan	#	\$	@	[	¥	]	^	‘	{		}	~
Norway	#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Denmark II	#	\$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü

12x24 点阵 Katakana (打印体)

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	—	—	—	—	—	—	—	—			■	■	■	■	■	+
9	┌	└	┌	└	—	—			┌	└	┌	└	┌	└	┌	└
A		。	「	」	、	・	ヲ	ア	イ	ウ	エ	オ	ヤ	ユ	ヨ	ツ
B	-	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C	タ	チ	ツ	テ	ト	ナ	ニ	ヌ	ネ	ノ	ハ	ヒ	フ	ヘ	ホ	マ
D	ミ	ム	メ	モ	ヤ	ユ	ヨ	ラ	リ	ル	レ	ロ	ワ	ン	。	。
E	二	ト	キ	コ	▲	▲	▼	▼	♠	♥	◆	♣	●	○	／	＼
F	×	円	年	月	日	時	分	秒	〒	市	区	町	村	人	■	■

## 8.2 173 国外文字库总表

### 173 国外文字库总表

FOR GT20L24F6Y UNICODE&amp;ISO8859

文系	区域	序号	国家		语言	ISO-8859		
拉丁文 (英语)	欧洲	1	英国	Britain 或 United Kingdom	英语	ISO8859-1		
		2	爱尔兰	Ireland				
	北美洲	3	美国	USA	英语	ISO8859-1		
		4	加拿大	Canada	英语、法语	ISO8859-1		
		5	伯利兹	Belize	英语	ISO8859-1		
		6	牙买加	Jamaica				
		7	特立尼达和多巴哥	Trinidad and Tobago				
		8	巴哈马	Bahamas				
		9	安提瓜和巴布达	Antigua and Barbuda				
		10	多米尼加	Dominica				
		11	圣文森特	St.Vincent				
		12	圣卢西亚岛	St.Lucia				
		13	格林纳达	Grenada				
		14	圣基茨—尼维斯	St.Kitts-Nevis				
	南美洲	15	圭亚那	Guyana			英语	ISO8859-1
	大洋洲	16	澳大利亚	Australia			英语	ISO8859-1
		17	新西兰	New Zealand				
		18	汤加	Tonga				
		19	斐济	Fiji				
		20	帕劳	Palau				
		21	所罗门	Solomon				
		22	瓦努阿图	Vanuatu				
		23	基里巴斯	Kiribati				
		24	瑙鲁	Nauru				
		25	马绍尔群岛	Marshall Islands				
	非洲	26	南非	South Africa	英语、南非荷兰语	ISO8859-1		
		27	津巴布韦	Zimbabwe	英语	ISO8859-1		
		28	冈比亚	Gambia				
		29	塞拉利昂	Sierra Leone				
		30	利比里亚	Liberia				
		31	加纳	Ghana				
		32	尼日利亚	Nigeria				
		33	乌干达	Uganda				
		34	赞比亚	Zambia				
		35	马拉维	Malawi				
		36	塞舌尔	Seychelles				

		37	毛里求斯	Mauritius		
		38	博茨瓦纳	Botswana		
		39	纳米比亚	Namibia		
		40	莱索托	Lesotho		
拉丁文 (法语)	欧洲	41	法国	FrUNICODEe	法语	ISO8859-15
		42	比利时	Belgium	法语、荷兰语	ISO8859-15
		43	摩纳哥	Monaco	法语、意大利语	ISO8859-15
	北美洲	44	海地	Haiti	法语	ISO8859-15
	非洲	45	塞内加尔	Senegal	法语	ISO8859-15
		46	马里	Mali	法语	
		47	布基纳法索	Burkina Faso	法语	
		48	几内亚	Guinea	法语	
		49	科特迪瓦	cote dlvoire	法语	
		50	多哥	Togo	法语	
		51	贝宁	Benin	法语	
		52	尼日尔	Niger	法语	
		53	喀麦隆	Cameroon	法语	
		54	乍得	Chad	法语	
55		中非	Central African Republic	法语		
拉丁文 (法语)	非洲	56	吉布提	Djibouti	法语	ISO8859-15
		57	布隆迪	Burundi		
		58	民主刚果	Republic of Democratic Congo		
		59	刚果	Congo		
		60	加蓬	Gabon		
		61	科摩罗	Comoros		
		62	马达加斯加	Madagascar		
拉丁文 (西班牙语)	欧洲	63	西班牙	Spain	西班牙语、加泰隆语	ISO8859-1、-15
		64	安道尔	Andorra	西班牙语	
	北美洲	65	墨西哥	Mexico	西班牙语	ISO8859-1 ISO8859-15
		66	危地马拉	Guatemala		
		67	哥斯达黎加	Costa Rica		
		68	巴拿马	Panama		
		69	多米尼加共和国	Dominican Republic		
		70	萨尔瓦多	El Salvador		
		71	洪都拉斯	Honduras		
		72	尼加拉瓜	Nicaragua		
		73	波罗黎各	Puerto Rico		
		74	古巴	Cuba		
	南美洲	75	委内瑞拉	Venezuela	西班牙语	ISO8859-1



		76	哥伦比亚	Colombia		ISO8859-15
		77	秘鲁	Peru		
		78	阿根廷	Argentina		
		79	厄瓜多尔	Ecuador		
		80	智利	Chile		
		81	乌拉圭	Uruguay		
		82	巴拉圭	Paraguay		
	非洲	83	玻利维亚	Bolivia	西班牙语	ISO8859-1 ISO8859-15
		84	赤道新几内亚	Equatorial New Guinea		
拉丁文 (葡萄牙语)	欧洲	86	葡萄牙	Portugal	葡萄牙语	ISO8859-1 ISO8859-15
	南美洲	87	巴西	Brazil		
	非洲	88	佛得角	Cape Verde		
		89	几内亚比绍	Guinea-Bissau		
		90	圣多美和普林西比	Sao Tome and Principe		
		91	安哥拉	Angola		
92	莫桑比克	Mozambique				
拉丁文 (德语)	欧洲	93	德国	Germany	德语	ISO8859-1、-15
		94	瑞士	Switzerland	德语、法语	ISO8859-1、-15
		95	奥地利	Austria	德语	ISO8859-1、-15
		96	卢森堡	Luxembourg	德语、法语	ISO8859-1、-15
		97	列支敦士登	Liechtenstein	德语	ISO8859-1、-15
拉丁文 (荷兰语)	欧洲	98	荷兰	Holland	荷兰语	ISO8859-1 ISO8859-15
	南美洲	99	苏里南	Surinam		
拉丁文 (北欧)	欧洲	100	丹麦	Denmark	丹麦语	ISO8859-1、-10
		101	挪威	Norway	挪威语	ISO8859-1、-10
		102	瑞典	Sweden	瑞典语	ISO8859-1、-10
		103	法罗群岛	Faroes, The	法罗语	ISO8859-1、-10
		104	格陵兰	Greenland	格陵兰语	ISO8859-1、-10
		105	冰岛	Iceland	冰岛语	ISO8859-1、-10
		106	芬兰	Finland	芬兰语、瑞典语	ISO8859-13、-15
		107	爱沙尼亚	Estonia	爱沙尼亚语	ISO8859-4、-13
		108	拉脱维亚	Latvia	拉脱维亚语	ISO8859-4、-13
		109	立陶宛	Lithuania	立陶宛语	ISO8859-4、-13
拉丁文 (中欧)	欧洲	110	捷克	Czech	捷克语	ISO8859-2
		111	斯洛伐克	Slovakia	斯洛伐克语	ISO8859-2
		112	波兰	Poland	波兰语	ISO8859-2、-16
		113	匈牙利	Hungary	匈牙利语	ISO8859-2、-16
		114	罗马尼亚	Romania	罗马尼亚语	ISO8859-16
拉丁文 (中欧)	欧洲	115	斯洛文尼亚	Slovenia	斯洛文尼亚语	ISO8859-2、-16
		116	克罗地亚	Croatia	克罗地亚语	ISO8859-2、-16
拉丁文	欧洲	117	意大利	Italy	意大利语	ISO8859-1

(南欧)		118	圣马力诺	San Marino		ISO8859-16
		119	梵提冈	Vatican		
		120	土耳其	Turkey	土耳其语	ISO8859-9
		121	马耳他	Malta	马耳他语	ISO8859-3、-9
		122	阿尔巴尼亚	Albania	阿尔巴尼亚语	ISO8859-1、-16
拉丁文 (东南亚)	亚洲	123	越南	Vietnam	越南语	ISO8859-1
		124	马来西亚	Malaysia	马来语	ISO8859-1
		125	文莱	Brunei		
		126	印度尼西亚	Indonesia	印尼语	ISO8859-1
		127	东帝汶	East Timor		
		128	菲律宾	Philippines, The	英语、塔加洛语	ISO8859-1
拉丁文 (非洲)	非洲	129	肯尼亚	Kenya	斯瓦西里语	ISO8859-1
		130	坦桑尼亚	Tanzania		
西里尔文 (东欧)	欧洲	131	俄罗斯	Russia	俄语	ISO8859-5
		132	白俄罗斯	Byelorussia 或 Belarus		
		133	乌克兰	Ukraine	俄语、乌克兰语	ISO8859-5
		134	保加利亚	Bulgaria	保加利亚语	ISO8859-5
		135	摩尔多瓦	Moldova	俄语	ISO8859-5
		136	南斯拉夫联盟	F.R.Yugoslavia	塞尔维亚语	ISO8859-5
		137	波黑	Barbados	塞尔维亚语	ISO8859-5
		138	马其顿	Macedonia	马其顿语	ISO8859-5
西里尔文 (亚洲)	亚洲	139	阿塞拜疆	Azerbaijan	阿塞拜疆语	ISO8859-5
		140	吉尔吉斯斯坦	Kirghizstan	吉尔吉斯斯坦语	ISO8859-5
		141	塔吉克斯坦	Tajikistan	塔吉克斯坦语	ISO8859-5
		142	土库曼斯坦	Turkmenistan	土库曼斯坦语	ISO8859-5
		143	乌兹别克斯坦	Uzbekistan	乌兹别克斯坦语	ISO8859-5
		144	哈萨克斯坦	Kazakhstan	哈萨克斯坦语	ISO8859-5
		145	蒙古	Mongolia	蒙古语	ISO8859-5
希腊文	亚洲	146	希腊	Greece	希腊语	ISO8859-7
		147	塞浦路斯	Cyprus		
阿拉伯文 (非洲)	非洲	148	埃及	Egypt	阿拉伯语	ISO8859-6
		149	突尼斯	Tunisia		
		150	利比亚	Libya		
		151	摩洛哥	Morocco		
		152	阿尔及利亚	Algeria		
		153	苏丹	Sudan, The		
		154	索马里	Somalia		
		155	吉布提	Djibouti		
		156	毛里塔尼亚	Mauritania		
阿拉伯文 (亚洲)	亚洲	157	叙利亚	Syria	阿拉伯语	ISO8859-6
		158	阿联酋	United Arab Emirates, The		
		159	黎巴嫩	Lebanon		
		160	也门	Yemen		

		161	科威特	Kuwait		
		162	卡塔尔	Qatar		
		163	巴林	Bahrain		
		164	阿曼	Oman		
		165	约旦	Jordan		
		166	伊拉克	Iraq		
		167	沙特阿拉伯	Saudi Arabia		
		168	巴勒斯坦	Palestine		
		169	伊朗	Iran	波斯语	
		170	巴基斯坦	Pakistan	乌尔都语、阿拉伯语	
		171	阿富汗	Afghanistan	普什图语	
希伯来文	亚洲	172	以色列	Israel	希伯来语	ISO8859-8
泰文	亚洲	173	泰国	Thailand	泰语	ISO8859-11

### 8.3 173 国外文拼音索引表 (汉语拼音排序)

#### 173 国外文字库索引表 (汉语拼音排序)

首字母	国家	总表序号	首字母	国家	总表序号	首字母	国家	总表序号
A	阿富汗	171		哥斯达黎加	67	P	葡萄牙	86
	阿尔巴尼亚	122		圭亚那	15		帕劳	20
	阿尔及利亚	152	H	海地	44	R	瑞典	102
	阿曼	164		洪都拉斯	71		瑞士	94
	阿根廷	78		荷兰	98	S	萨尔瓦多	70
	阿联酋	158		哈萨克斯坦	144		圣马力诺	118
	阿塞拜疆	139	J	吉尔吉斯斯坦	140		圣多美和普林西比	90
	爱尔兰	2		吉布提	155		圣基茨—尼维斯	14
	爱沙尼亚	107		吉布提	56		圣文森特	11
	奥地利	95		津巴布韦	27		圣卢西亚岛	12
	澳大利亚	16		加蓬	60		塞浦路斯	147
	安道尔	64		加拿大	4		塞内加尔	45
	安提瓜和巴布达	9		加纳	31		塞拉利昂	29
	安哥拉	91		几内亚	48		塞舌尔	36
	埃及	148		几内亚比绍	89		索马里	154
B	巴拿马	68		基里巴斯	23		苏丹	153
	巴黎嫩	159		捷克	110		沙特阿拉伯	167
	巴林	163	K	喀麦隆	53		所罗门	21
	巴基斯坦	170		科摩罗	61		斯洛伐克	111
	巴勒斯坦	168		科特迪瓦	49		斯洛文尼亚	115
	巴拉圭	82		卡塔尔	162		苏里南	99

	巴哈马	8		肯尼亚	129	T	土耳其	120
	巴西	87		克罗地亚	116		土库曼斯坦	142
	白俄罗斯	132		科威特	161		突尼斯	149
	冰岛	105	L	利比里亚	30		特立尼达和多巴哥	7
	贝宁	51		利比亚	150		汤加	18
	伯利兹	5		拉脱维亚	108		塔吉克斯坦	141
	比利时	42		立陶宛	109		泰国	173
	波罗黎各	73		罗马尼亚	114		坦桑尼亚	130
	玻利维亚	83		列支敦士登	97	W	瓦努阿图	22
	波兰	112		卢森堡	96		危地马拉	66
	布隆迪	57		莱索托	40		文莱	125
	波黑	137	M	马耳他	121		乌拉圭	81
	保加利亚	134		马其顿	138		乌克兰	133
	布基纳法索	47		马达加斯加	62		乌干达	33
	博茨瓦纳	38		马来西亚	124		乌兹别克斯坦	143
C	赤道新几内亚	84		马拉维	35		委内瑞拉	75
D	德国	93		马里	46	X	西班牙	63
	丹麦	100		马绍尔群岛	25		休达和梅利亚	85
	多米尼加	10		摩尔多瓦	135		希腊	146
	多米尼加共和国	69		摩纳哥	43		叙利亚	157
	东帝汶	127		摩洛哥	151		新西兰	17
	多哥	50		毛里求斯	37		匈牙利	113
E	俄罗斯	131		毛里塔尼亚	156	Y	牙买加	6
	厄瓜多尔	79		秘鲁	77		也门	160
F	佛得角	88		美国	3		意大利	117
	梵提冈	119		蒙古	145		约旦	165
	法罗群岛	103		墨西哥	65		印度尼西亚	126
	芬兰	106		民主刚果	58		越南	123
	菲律宾	128		莫桑比克	92		伊拉克	166
	法国	41	N	纳米比亚	39		伊朗	169
	斐济	19		南非	26		英国	1
G	冈比亚	28		瑙鲁	24		以色列	172
	古巴	74		尼加拉瓜	72	Z	赞比亚	34
	哥伦比亚	76		尼日利亚	32		中非	55
	格林纳达	13		尼日尔	52		乍得	54
	格陵兰	104		南斯拉夫联盟	136		智利	80
	刚果	59		挪威	101			

## 8.4 173 国外文字库索引表(英文字母排序)

### 173 国外文字库索引表(英文字母排序)

首字母	国家	总表序号	首字母	国家	总表序号	首字母	国家	总表序号
A	Afghanistan	171	G	Gambia	28	O	Oman	164
	Albania	122		Germany	93	P	Palau	20
	Algeria	152		Ghana	31		Pakistan	170
	Andorra	64		Greece	146		Palestine	168
	Angola	91		Greenland	104		Panama	68
	Antigua and Barbuda	9		Grenada	13		Paraguay	82
	Argentina	78		Guatemala	66		Peru	77
	Australia	16		Guinea	48		Philippines, The	128
	Austria	95		Guinea-Bissau	89		Poland	112
	Azerbaijan	139		Guyana	15		Portugal	86
B	Bahamas	8	H	Haiti	44		Puerto Rico	73
	Bahrain	163		Holland	98	Q	Qatar	162
	Barbados	137		Honduras	71		Republic of Democratic Congo	58
	Belgium	42		Hungary	113		Romania	114
	Belize	5	I	Iceland	105		Russia	131
	Benin	51		Indonesia	126	S	San Marino	118
	Bolivia	83		Iran	169		Sao Tome and Principe	90
	Botswana	38		Iraq	166		Saudi Arabia	167
	Brazil	87		Israel	172		Senegal	45
	Britain 或 United Kingdom	1		Italy	117		Seychelles	36
	Brunei	125	J	Jamaica	6		Sierra Leone	29
	Bulgaria	134		Jordan	165		Slovakia	111
	Burkina Faso	47	K	Kazakhstan	144		Slovenia	115
	Burundi	57		Kenya	129		Solomon	21
	Byelorussia 或 Belarus	132		Kirghizstan	140		Somalia	154
C	Cameroon	53		Kiribati	23		South Africa	26
	Canada	4		Kuwait	161		Spain	63
	Cape Verde	88	L	Latvia	108		St.Kitts-Nevis	14
	Central African Republic	55		Lebanon	159		St.Lucia	12

	Ceuta and Melilla	85		Lesotho	40		St.Vincent	11
	Chad	54		Liberia	30		Sudan, The	153
	Chile	80		Libya	150		Surinam	99
	Colombia	76		Liechtenstein	97		Sweden	102
	Comoros	61		Lithuania	109		Switzerland	94
	Congo	59		Ireland	2		Syria	157
	Costa Rica	67		Luxembourg	96	T	Tajikistan	141
	cote d'Ivoire	49	M	Macedonia	138		Tanzania	130
	Croatia	116		Madagascar	62		Thailand	173
	Cuba	74		Malawi	35		Togo	50
	Cyprus	147		Malaysia	124		Tonga	18
	Czech	110		Mali	46		Trinidad and Tobago	7
D	Denmark	100		Malta	121		Tunisia	149
	Djibouti	56		Marshall Islands	25		Turkey	120
	Djibouti	155		Mauritania	156		Turkmenistan	142
	Dominica	10		Mauritius	37	U	Uganda	33
	Dominican Republic	69		Mexico	65		Ukraine	133
E	East Timor	127		Moldova	135		United Arab Emirates, The	158
	Ecuador	79		Monaco	43		Uruguay	81
	Egypt	148		Mongolia	145		USA	3
	El Salvador	70		Morocco	151		Uzbekistan	143
	Equatorial New Guinea	84		Mozambique	92	V	Vanuatu	22
	Estonia	107	N	Namibia	39		Vatican	119
F	F.R.Yugoslavia	136		Nauru	24		Venezuela	75
	Faroese, The	103		New Zealand	17		Vietnam	123
	Fiji	19		Nicaragua	72	Y	Yemen	160
	Finland	106		Niger	52	Z	Zambia	34
	FrUNICODEe	41		Nigeria	32		Zimbabwe	27
G	Gabon	60		Norway	101			