

GT22L16M1Y

标准汉字字库芯片

简要说明

BRIEF

- **GB18030 字符集 (27484 汉字): 15x16 点阵**
 - **兼容 Unicode V1.0 (20902 汉字): 15x16 点阵**
 - **ASCII 字符集 (4 套): 5x7~ 16 点阵**
-
- **排置方式: 竖置横排**
 - **总线接口: SPI 串行总线**
 - **芯片形式: DFN 封装 SOP-16 封装**

VER 1.6_ I A

2012-3

版本修订记录

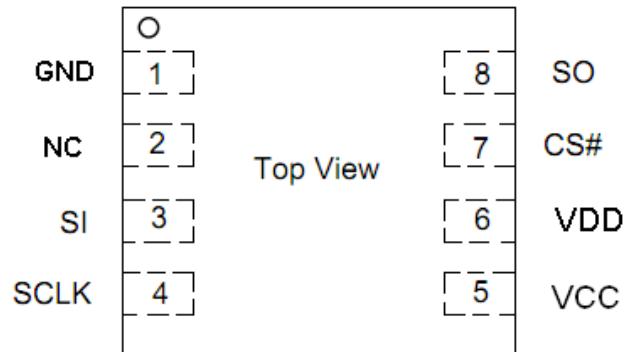
版本号	修改内容	日期	备注
V1.3. II.B	升级 code 转码表-增加 DFN 封装	2011-11-17	
V1.3. II.C	删除 SOP 封装	2011-11-23	
V1.3. II.D	字型样张修改	2012-1-7	
V1.4. I.A	升级 code, 增加新 ASCII 与 BIG5 转码表	2012-1-13	
V1.4. I.B	增加 SOP-16 封装, 修改电流参数, 修改引脚描述	2012-1-30	
V1.4B. I.B	增加 24748+尾字(软字库)	2012-3-12	
V1.5. I.A	增加了 16X32ASCII 字符, 8X16 ASCII 字符位置下调	2012-2-15	
V1.6.IA	替换 8X16 ASCII 字符,增加 24748+尾字 (ROM), 增加 8X16 ASCII 粗体字符	2012-3-12	

1 概述

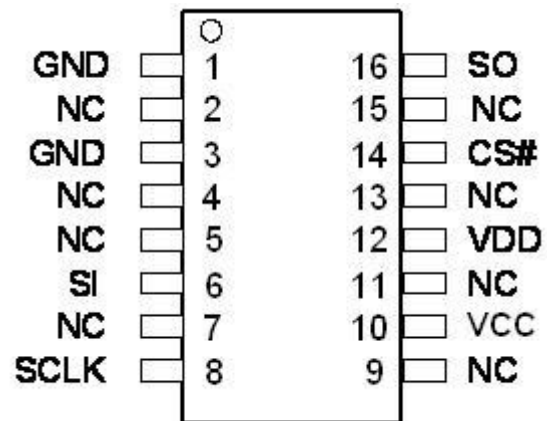
GT22L16M1Y是一款15X16点阵字库芯片，支持GB18030国标汉字（同时支持UNICODE编码）。排列格式为竖置横排。用户通过字符内码，利用用户手册提供的方法计算出该字符点阵在芯片中的地址，可从该地址连续读出字符点阵信息。

1.1 芯片特点

- 封装：DFN-8
- 尺寸：4.0mm x 4.0mm (158milX158mil)
- 数据总线：SPI 串行总线接口
- 点阵排列方式：字节竖置横排
- 时钟频率：30MHz(max.) @3.3V
- 工作电压：2.7V~3.6V
- 电流：
 - 工作电流：8mA
 - 待机电流：8uA
- 工作温度：-20°C~70°C



- 封装：SOP-16
- 尺寸：10.0mm x 4.4mm (394milX173mil)
- 数据总线：SPI 串行总线接口
- 点阵排列方式：字节竖置横排
- 时钟频率：30MHz(max.) @3.3V
- 工作电压：2.7V~3.6V
- 电流：
 - 工作电流：8mA
 - 待机电流：8uA
- 工作温度：-20°C~70°C



1.2 引脚描述

DFN-8	名称	I/O	描述
1	GND		地(Ground)
2	NC		未定义
3	SI	I	串行数据输入 (Serial data input)
4	SCLK		串行时钟输入
5	VCC		电源(+ 3.3V Power Supply)需串2K电阻
6	VDD		电源(+ 3.3V Power Supply)
7	CS#		片选输入 (Chip enable input)
8	SO	O	串行数据输出 (Serial data output)

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。

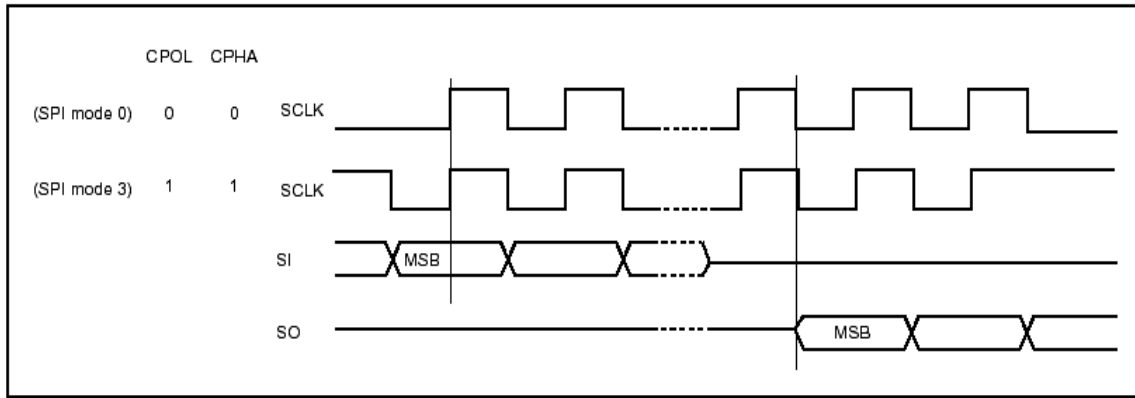
SOP16	名称	I/O	描述
1	GND		地(Ground)
2	NC		空脚
3	GND		地(Ground)
4	NC		空脚
5	NC		空脚
6	SI	I	串行数据输入 (Serial data input)
7	NC		
8	SCLK	I	串行时钟输入 (Serial clock input)
9	NC		空脚
10	VCC		电源(+ 3.3V Power Supply) 需串2K电阻
11	NC		空脚
12	VDD		电源(+ 3.3V Power Supply)
13	NC		空脚
14	CS#	I	片选输入 (Chip enable input)
15	NC		空脚
16	SO	O	串行数据输出 (Serial data output)

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。

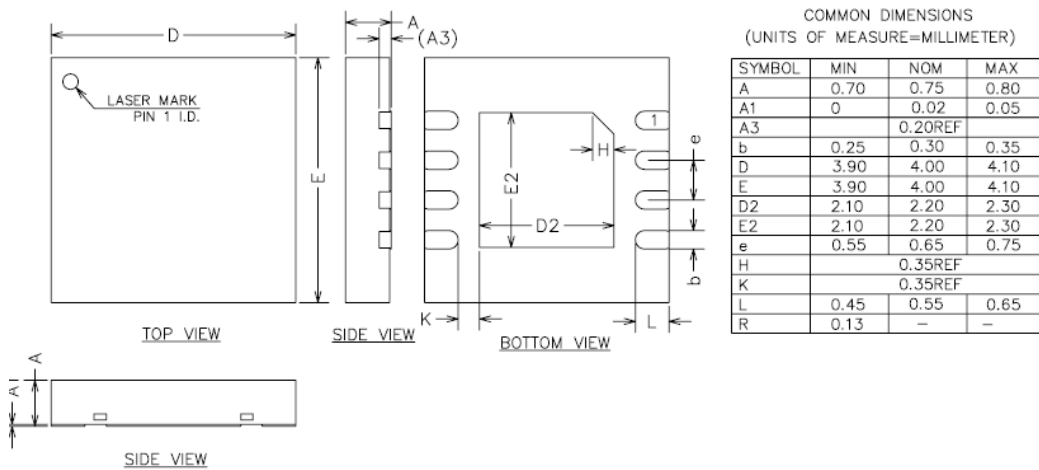
串行数据输入 (SI): 该信号用来把数据从串行输入芯片, 数据在时钟的上升沿移入。

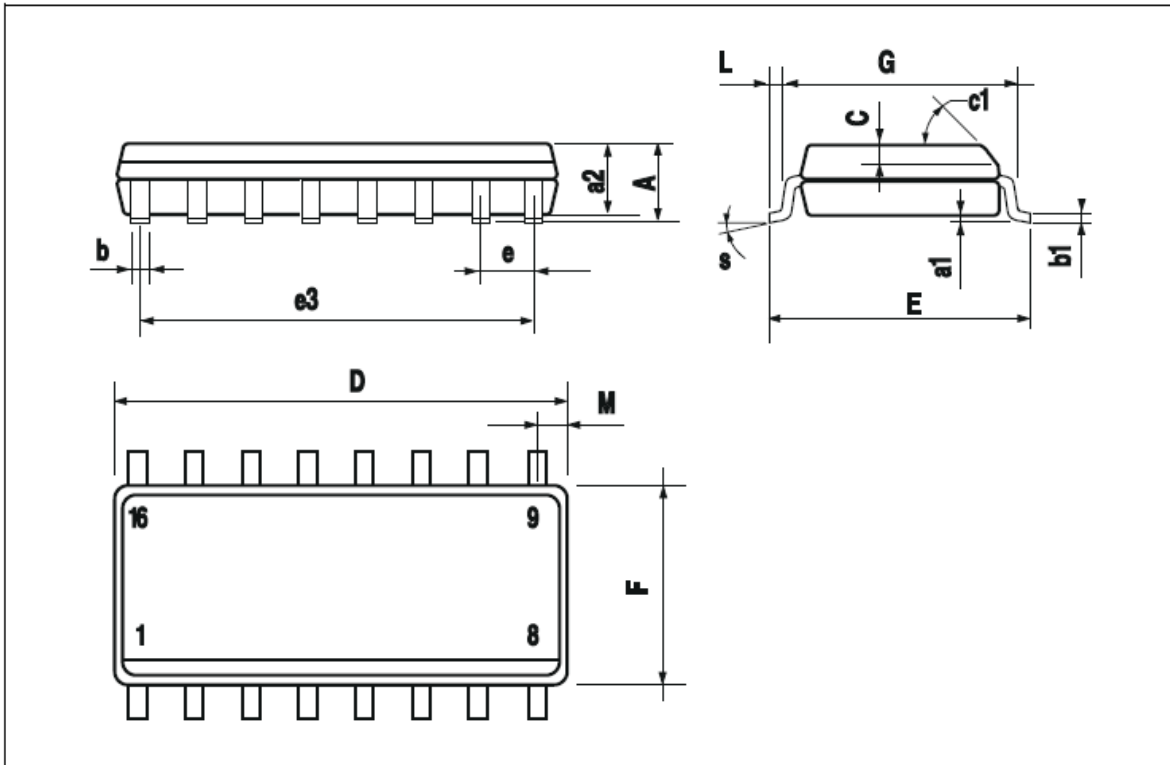
串行时钟输入 (SCLK): 数据在时钟上升沿移入, 在下降沿移出。

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。



1.3 封装尺寸





Dim.	Millimeters			Inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.75			0.069
a1	0.1		0.2	0.004		0.008
a2			1.6			0.063
b	0.35		0.46	0.014		0.018
b1	0.19		0.25	0.007		0.010
C		0.5			0.020	
c1	45° (typ.)					
D	9.8		10	0.386		0.394
E	5.8		6.2	0.228		0.244
e		1.27			0.050	
e3		8.89			0.350	
F	3.8		4.0	0.150		0.157
G	4.6		5.3	0.181		0.209
L	0.5		1.27	0.020		0.050
M			0.62			0.024
S	8° (max.)					

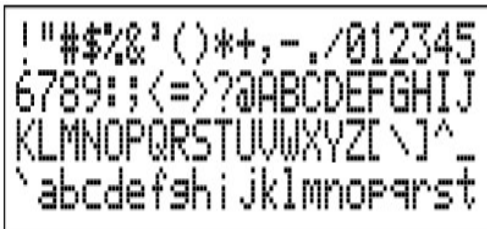
1.4 芯片内容

点阵 字符集	等宽字符								不等宽字符
	5X7	7X8	7X10 打印字体	7X12 打印字体	8X16	8X16 粗体	16X32	15X16 汉字	16 点阵 Arial
ASCII	96	96	96	96	96	96	96		
GB18030								27484+ 1038	
Unicode→GB 转码表	20902								
BIG5→GB 转码表	13060 汉字 (5401+7652+7) +408 字符								

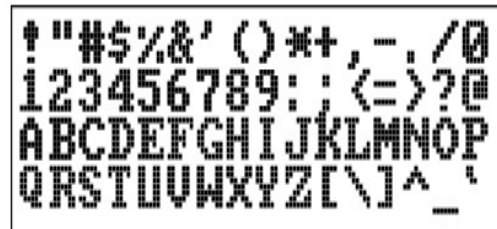
1.5 字型样张

1) ASCII 字库

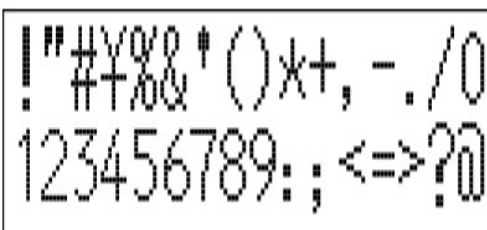
5x7 点 ASCII



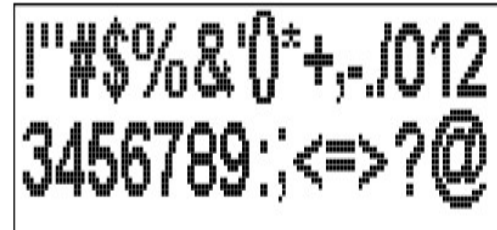
7x8 点 ASCII



8X16 点 ASCII



16 点 ASCII 方头字符



15x16点汉字GB18030

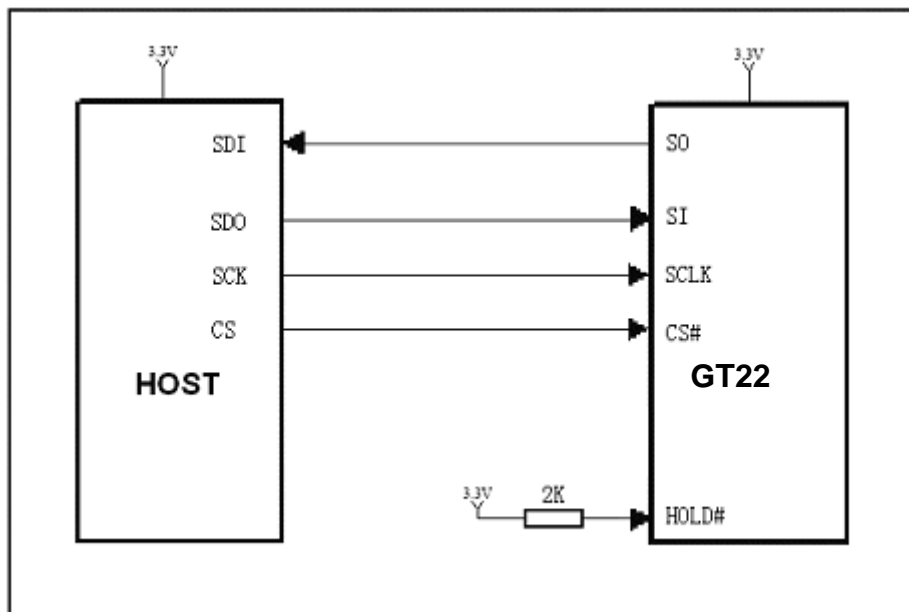


15x16汉字Unicode



1.6 HOST CPU 主机 SPI 接口电路示意图

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）。



HOST CPU 主机 SPI 接口电路示意图

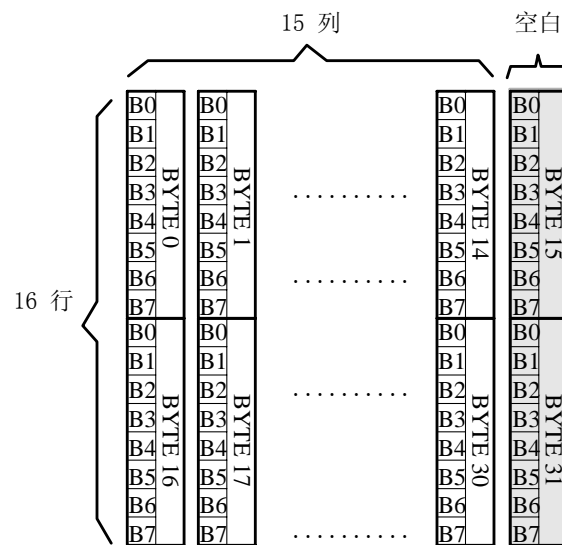
2 字库调用方法

2.1 汉字点阵排列格式

每个字符在芯片中是以汉字点阵字模的形式存储的，每个点用一个二进制位表示，存 1 的点，当显示时可以在屏幕上显示亮点，存 0 的点，则在屏幕上不显示。点阵排列格式为竖置横排：即一个字节的低位表示下面的点，高位表示上面的点（如果用户按 16bit 总线宽度读取点阵数据，请注意高低字节的顺序），排满一行后再排下一行。这样把点阵信息用来直接在显示器上按上述规则显示，则将出现对应的字符。

2.1.1 15X16 点 GB2312 汉字排列格式举例

15X16 点汉字的信息需要 32 个字节（BYTE 0 – BYTE 31）来表示。该 15X16 点汉字的点阵数据是横置横排的，其具体排列结构如下图：

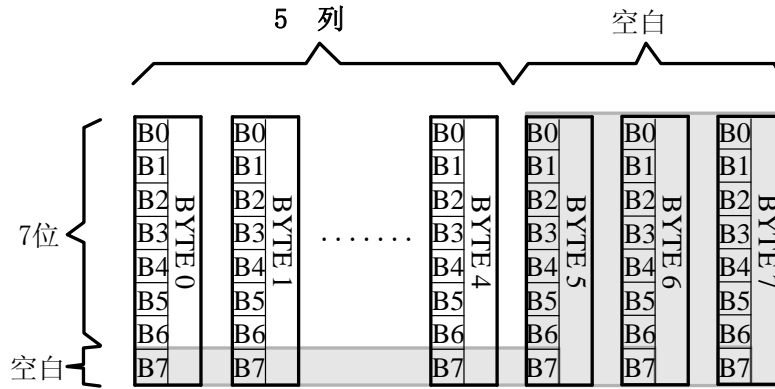


2.2 字符点阵排列格式

每个字符在芯片中是以字符点阵字模的形式存储的，每个点用一个二进制位表示，存 1 的点，当显示时可以在屏幕上显示亮点，存 0 的点，则在屏幕上不显示。点阵排列格式为竖置横排：即一个字节的低位表示下面的点，高位表示上面的点（如果用户按 16bit 总线宽度读取点阵数据，请注意高低字节的顺序），排满一行后再排下一行。这样把点阵信息用来直接在显示器上按上述规则显示，则将出现对应的字符

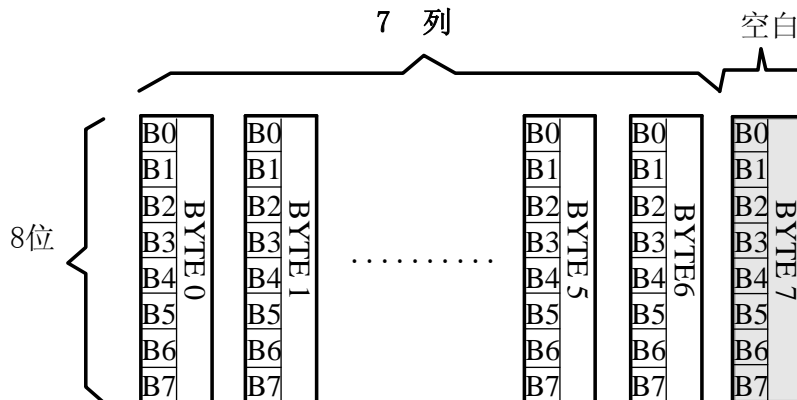
2.2.1 5X7 点 ASCII 字符排列格式

5X7 点 ASCII 的信息需要 8 个字节（BYTE 0 – BYTE7）来表示。该 ASCII 点阵数据是竖置横排的，其具体排列结构如下图：



2.2.2 7X8 点 ASCII 字符排列格式

7X8 点 ASCII 的信息需要 8 个字节 (BYTE 0 – BYTE7) 来表示。该 ASCII 点阵数据是竖置横排的，其具体排列结构如下图：



2.2.3 8X16 点字符排列格式

适用于此种排列格式的字体有：

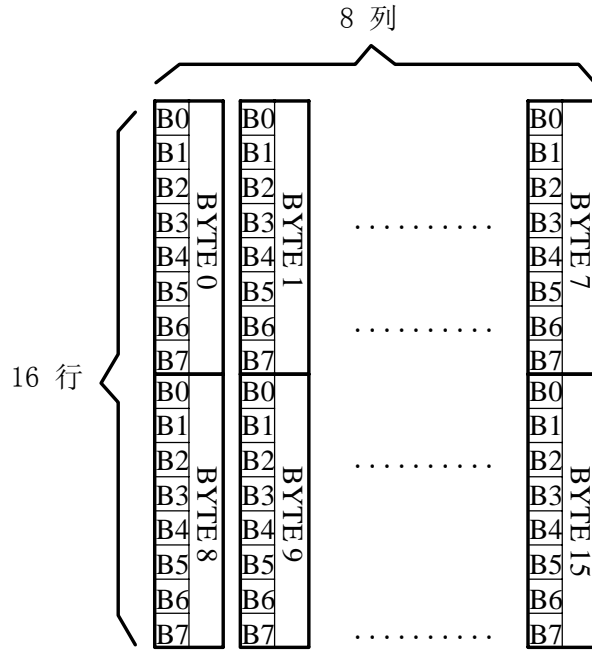
8X16 点 ASCII 字符

8X16 点粗体 ASCII 字符

8X16 点拉丁文、希腊文、西里尔文字符、希伯来文系字符

8X16 点 ISO 8859-1~16 字符

8X16 点字符信息需要 16 个字节 (BYTE 0 – BYTE15) 来表示。该点阵数据是竖置横排的，其具体排列结构如下图：



2.2.4 16 点阵不等宽字符排列格式

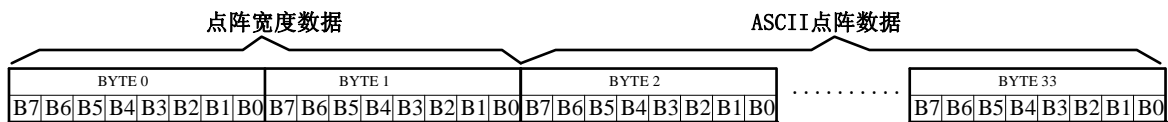
适用于此种排列格式的字有：

- 16 点阵不等宽 ASCII 方头 (Arial) 字符、16 点阵不等宽 ASCII 白正 (Times) 字符
- 16 点阵不等宽 Unicode 字符 (拉丁文系、希腊文系、西里尔文系)
- 16 点阵不等宽阿拉伯文系字符

16 点阵不等宽字符的信息需要 34 个字节 (BYTE 0 – BYTE33) 来表示。

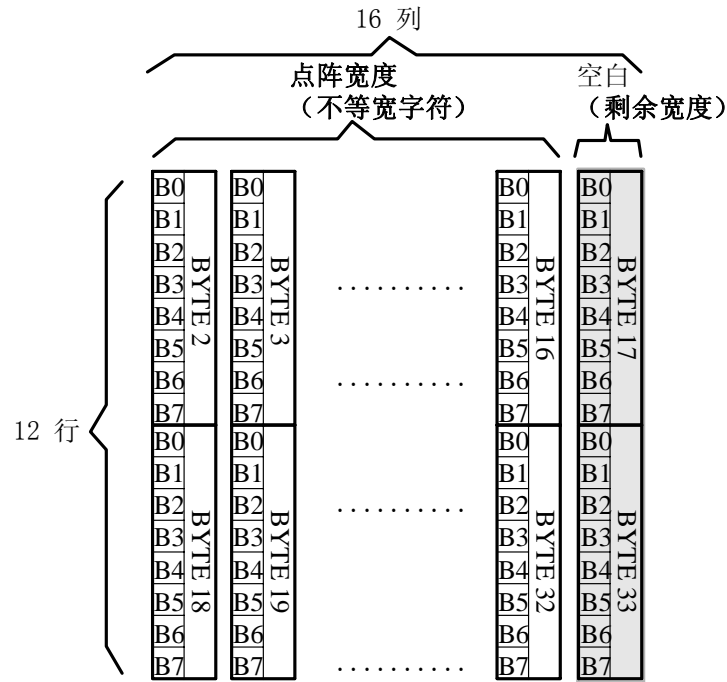
■ 存储格式

由于字符是不等宽的，因此在存储格式中 BYTE0~ BYTE1 存放点阵宽度数据，BYTE2-33 存放竖置横排点阵数据。具体格式见下图：



■ 存储结构

不等宽字符的点阵存储宽度是以 BYTE 为单位取整的，根据不同字符宽度会出现相应的空白区。根据 BYTE0~ BYTE1 所存放点阵的实际宽度数据，可以对还原下一个字的显示或排版留作参考。



例如：ASCII 方头字符 B

0-33BYTE 的点阵数据是： 00 0C 00 F8 F8 18 18 18 18 18 F8 F0 00 00 00 00 00 00 00 7F 7F 63 63 63 63 67 3E 1C 00 00 00 00 00

其中：

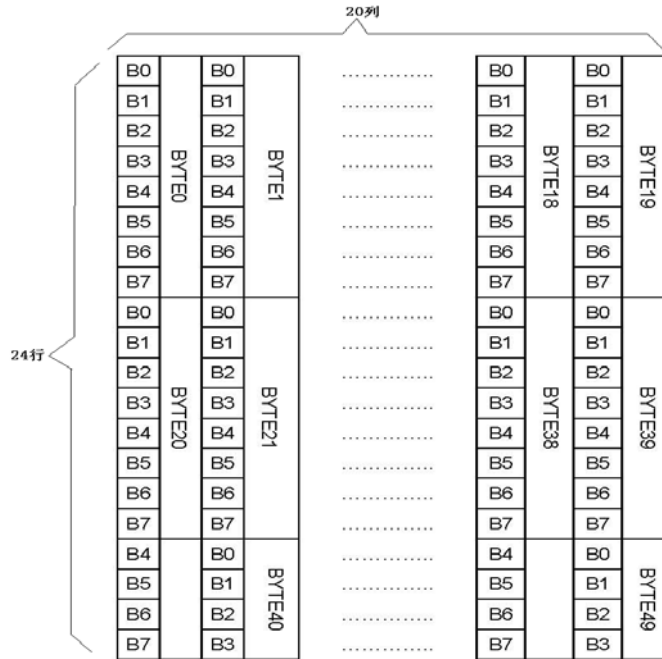
BYTE0~ BYTE1: 00 0C 为 ASCII 方头字符 B 的点阵宽度数据，即：12 位宽度。字符后面有 4 位空白区，可以在排版下一个字时考虑到这一点，将下一个字的起始位置前移。

BYTE2-33: 00 F8 F8 18 18 18 18 18 F8 F0 00 00 00 00 00 00 00 7F 7F 63 63 63 63 67 3E 1C 00 00 00 00 00 为 ASCII 方头字符 B 的点阵数据

2.2.5 GB2312 1 区&3 区压缩字符排列格式

■ 存储格式

此处将 20X24 字符 60 字节压缩成 50 字节存放
如图：



20X24 字符按照正常存储需要占用 60 个字节，第三排每字节高四位补零。

使用压缩存放方式后，第一第二排不变，第三排用 10 个字节代替原先 20 个字节，即 1 个字节的高四位和低四位都用于存放数据，具体存放方法是：

将两个 4bit 数据组成 1 个 byte 存放。

例如：20X24 字节数据原先是：

（前 40 字节）：

```
00 0C 00 F8 F8 18 18 18 18 18 F8 F0 00 00 00 00 00 00 00 7F
7F 63 63 63 63 63 67 3E 1C 00 00 00 00 00 00 00 00 00 00 7F
```

（后 20 字节）：

```
01,02,03,04,05,06,07,08,09,0A,01,02,03,04,05,06,07,08,09,0A,
```

压缩后成为：

（前 40 字节不变）：

```
00 0C 00 F8 F8 18 18 18 18 18 F8 F0 00 00 00 00 00 00 00 7F
7F 63 63 63 63 63 67 3E 1C 00 00 00 00 00 00 00 00 00 00 7F
```

（后 10 字节）：

```
12,34,56,78,9A,12,34,56,78,9A,
```

因此在使用时需要进行移位还原。