

GT21L16T1W 标准汉字字库芯片

用户手册 DATASHEET

- 字型： 15X16 点阵
- 汉字字符集： GB12345、BIG5 基本集
- 日文字符集： JIS0208、兼容 Unicode 内码
- 多国文字： 拉丁文、基里尔文、阿拉伯文等 150 国
- 排置方式： 横置横排
- 总线接口： SPI 串行总线
- 芯片形式： SO8 封装

VER 3.6

2010-Q3

版本修订记录

版本号	修改内容	日期	备注
V35	1. Big5 字库算法部分。	2010-7	
	2. 8*16 希腊算法部分。	2010-7	
	3. 12 点不等宽拉丁字符算法部分。	2010-7	
	4. 12 点不等宽希腊字符算法部分。	2010-7	
	5. 12 点不等宽基里尔字符算法部分。	2010-7	
	6. 16 点不等宽拉丁字符算法部分。	2010-7	
	7. 16 点不等宽希腊字符算法部分。	2010-7	
	8. 16 点不等宽基里尔字符算法部分。	2010-7	
V36	9. 12 点不等宽基里尔字符算法部分。	2010-8	
	10. 16 点不等宽基里尔字符算法部分。	2010-8	
	11. 12 点不等宽希腊字符算法部分。	2010-8	
	12. 16 点不等宽希腊字符算法部分。	2010-8	
	13. 附录中增加补丁文件列表。	2010-8	

GT21L16T1W

繁日外

字库芯片

标准字库

繁日外字库

目 录

第一部分：硬件部分

1 概述	4
1.1 芯片特点	4
1.2 芯片内容	4
2 引脚描述与接口连接	6
2.1 引脚配置	6
2.2 引脚描述	6
2.3 HOST CPU 主机 SPI 接口电路示意图	7
3 操作指令	8
3.1 指令参数	8
3.2 Read Data Bytes (一般读取)	8
3.3 Read Data Bytes at Higher Speed (快速读取点阵数据)	9
4 电气特性	10
4.1 绝对最大额定值	10
4.2 DC 特性	10
4.3 AC 特性	10
5 封装尺寸	12

第二部分：软件部分

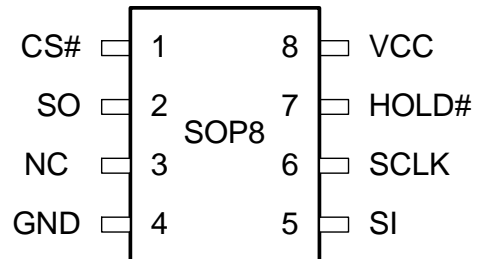
6 字库调用方法	13
6.1 字符点阵排列格式	13
6.2 点阵字库地址表	17
6.3 字符在芯片中的地址计算方法	17
7 附录	28
7.1 GB12345 1 区字符 (846 字符)	28
7.2 Unicode 字符区字符	31
7.3 国家地区语言文系 Unicode 对照表 (150 国)	40

1 概述

GT21L16T1W是一款15X16点阵字库芯片，支持GB12345国标繁体汉字（含国家信标委合法授权）、BIG5基本集汉字、JIS0208 日文字符集（兼容Unicode内码）及150国文字。排列格式为横置横排。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：字节横置横排
- 时钟频率：30MHz(max.) @3.3V
- 工作电压：2.7V~3.6V
- 电流：
 - 工作电流：12mA
 - 待机电流：10uA
- 封装：SOP8
- 尺寸（SOP8）：4.90mmX3.90mm（193milX154mil）
- 工作温度：-20℃~85℃



1.2 芯片内容

分类	字库内容	编码体系（字符集）	字符数
汉字及字符	16 点阵 GB12345 标准点阵字库	GB12345	6866+846
	16 点阵 BIG5 基本集点阵字库	BIG5	5401+408
日文及字符	16 点阵 JIS0208 点阵字库	JIS0208	6398+1644
ASCII 字符	5X7 点 ASCII 字符	ASCII	96
	7X8 点 ASCII 字符	ASCII	96
	6X12 点 ASCII 字符	ASCII	96
	8X16 点 ASCII 字符	ASCII	96
	8X16 点粗体 ASCII 字符	ASCII	96
	12 点阵不等宽 ASCII 方头（Arial）字符	ASCII	96
	16 点阵不等宽 ASCII 方头（Arial）字符	ASCII	96
Unicode 多国字符	8X16 点拉丁文系字符	Unicode	376
	8X16 点希腊文系字符	Unicode	96
	8X16 点基里尔文系字符	Unicode	250
	12 点阵不等宽 Unicode 字符（拉丁文系、希腊文系、基里尔文系）	Unicode	555
	16 点阵不等宽 Unicode 字符（拉丁文系、希腊文系、基里尔文系）	Unicode	555
	12 点阵不等宽阿拉伯文系字符	Unicode	250
	12 点阵不等宽阿拉伯文系变体字符	自定义	498
	16 点阵不等宽阿拉伯文系字符	Unicode	250
内码索引表	16 点阵不等宽阿拉伯文系变体字符	自定义	498
	BIG5 汉字内码字符索引表	BIG5	
	JIS0208 日文内码字符索引表	JIS0208	
	Unicode 日文内码字符索引表	Unicode	

150 国文字简表

文系	语言	国家	拉丁文国家	文系国家
拉丁文系	英语	英国、美国等	39 国	112 国
	法语	法国、尼日尔等	22 国	
	西班牙语	西班牙、墨西哥等	22 国	
	葡萄牙语	葡萄牙、巴西等	7 国	
	德语	德国、奥地利等	5 国	
	意大利语	意大利、圣马力诺等	3 国	
	马来语	马来西亚、文莱等	2 国	
	斯瓦希里语	坦桑尼亚、肯尼亚等	2 国	
	其他拉丁语	荷兰、瑞典等	10 国	
阿拉伯文系	阿拉伯语	埃及、约旦等		21 国
基里尔文系	12 种语言	俄罗斯、哈萨克等		15 国
希腊文系	希腊语	希腊、塞浦路斯		2 国
				合计 150 国

字型样张

GB12345 汉字

啊阿埃挨哎唉哀皑癌藹矮艾礙愛隘鞍
 俺按暗岸胺案骛昂盎凹敖熬翱澳傲奥
 芭捌扒叭吧芭八疤巴拔跋靶把耙霸霸
 白柏百摆佰败拜裨斑班搬扳般颁板版

BIG5 汉字

一乙丁七乃九了二人儿入八几刀刁力
 卜又三下丈上丫丸凡久么也乞于亡兀
 千叉口土士夕大女子孑寸小尢尸山
 己巳巳巾干井弋弓才丑丐不中丰丹之

JIS 日文

◆□■△▲※→※〒→←↑↓＝€
 ああいううええおおかがきぎく
 亜啞娃阿哀愛挨始逢葵茜穉惡握渥
 院陰隱韻吋右宇鳥羽迂雨卯鶉窺丑

拉丁文

Ü ü Á á Ä ä Å å Æ æ Ø ø Æ æ ; ' °
 Z H Θ I K Λ M N Ξ O Π P
 β γ δ ε ζ η θ ι κ λ μ ν

希腊文

Δ Ε Ζ Η Θ Ι Κ Λ Ξ Υ Ω Ι
 é ñ í ü α β γ δ φ χ ψ Ω
 φ χ ψ ω ι ü ó ú ο π ρ σ

基里尔文

Ё Ъ Ѓ Є С І І Ј Љ Њ Ы К
 Н О П Р С Т У Ф Х Ц Ч Ш
 й к л м н о п р с т у ф

阿拉伯文

، * ' أ ا إ ا ء ا
 ج د د د د د د د د د د د د
 ك ك ك ك ك ك ك ك ك ك ك ك

6X12 点 ASCII 字符

!"#\$%&'()*+,-./0123456789:;
 =>?@ABCDEFGHIJKLMN OPQRSTU V
 WXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
 {|}~āâãäēëèéîíïðóôõ

8X16 点 ASCII 字符

!"#\$%&'()*+,-./012345
 6789:;<=>?@ABCDEFGHIJK
 LMNOPQRSTUVWXYZ[\]^_`a

5x7 点 ASCII 字符

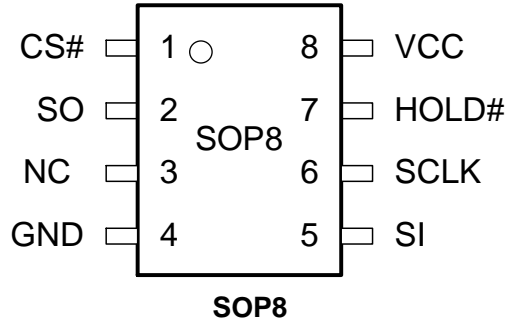
!"#\$%&'()*+,-./0123456789:
 =>?@ABCDEFGHIJKLMN OPQRSTU V
 WXYZ[\]^_`abcdefghijklmnopqrstuvwxyz

7x8 点 ASCII 字符

!"#\$%&'()*+,-./01234
 6789:;<=>?@ABCDEFGHIJ
 LMNOPQRSTUUVWXYZ[\]^_`
 bcdefghijklmnopqrstuv
 6789:;<=>?@ABCDEFGHIJ

2 引脚描述与接口连接

2.1 引脚配置



2.2 引脚描述

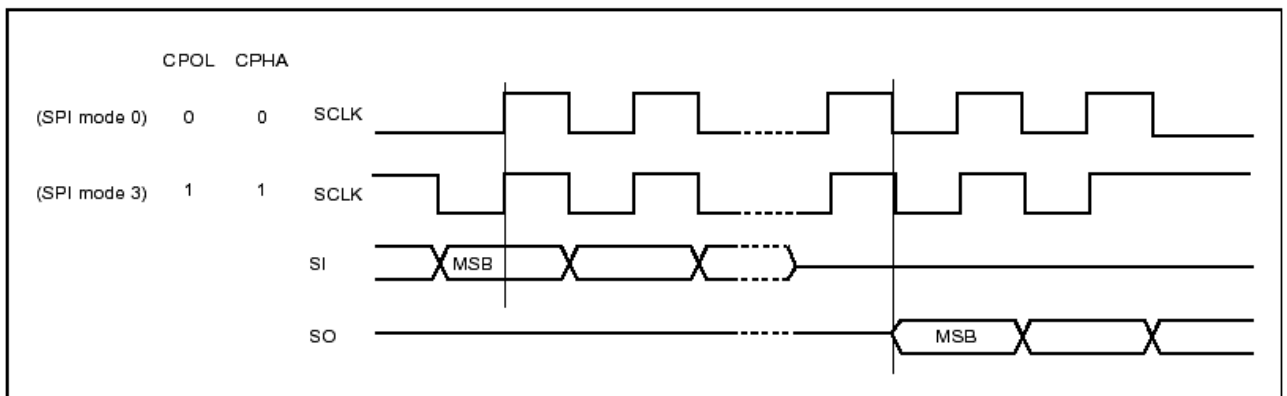
SOP8	名称	I/O	描述
1	CS#	I	片选输入 (Chip enable input)
2	SO	O	串行数据输出 (Serial data output)
3	NC		悬空
4	GND		地(Ground)
5	SI	I	串行数据输入 (Serial data input)
6	SCLK	I	串行时钟输入 (Serial clock input)
7	HOLD#	I	总线挂起 (Hold, to pause the device without)
8	VCC		电源(+ 3.3V Power Supply)

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。

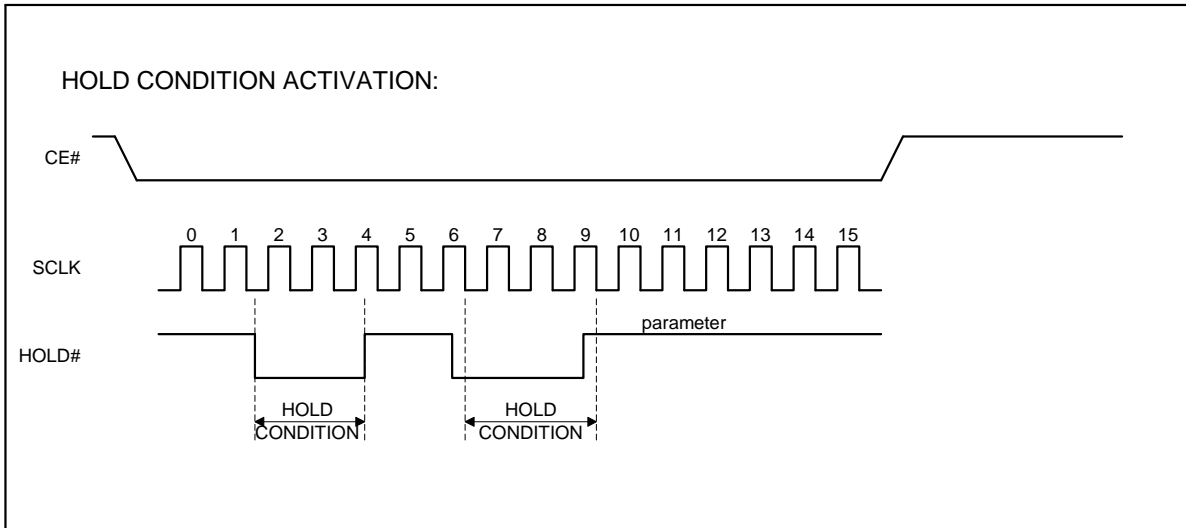


总线挂起输入 (HOLD#):

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。

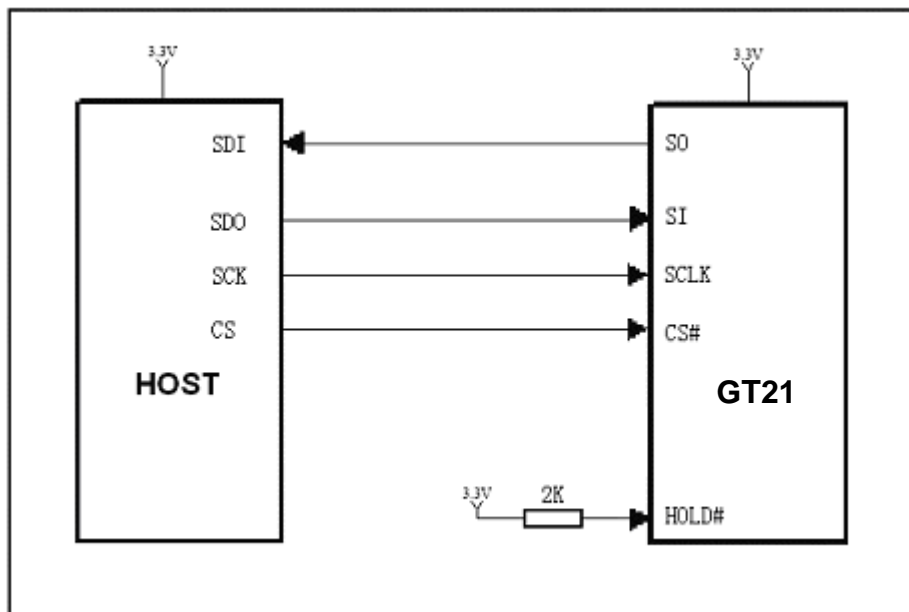
当HOLD#信号变为低并且串行时钟信号 (SCLK) 处于低电平时，进入总线挂起状态。

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。



2.3 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

3 操作指令

3.1 指令参数

Instruction Set

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 ∞

所有对本芯片的操作只有 2 个，那就是 Read Data Bytes (READ “一般读取”)和 Read Data Bytes at Higher Speed (FAST_READ “快速读取点阵数据”)。

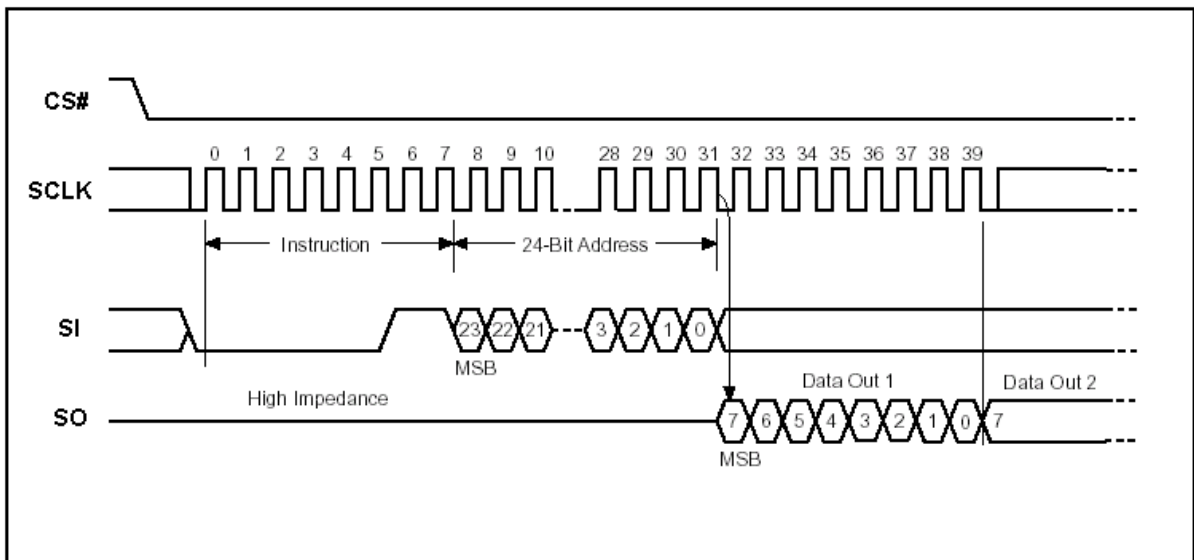
3.2 Read Data Bytes（一般读取）

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，紧接着的是 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:



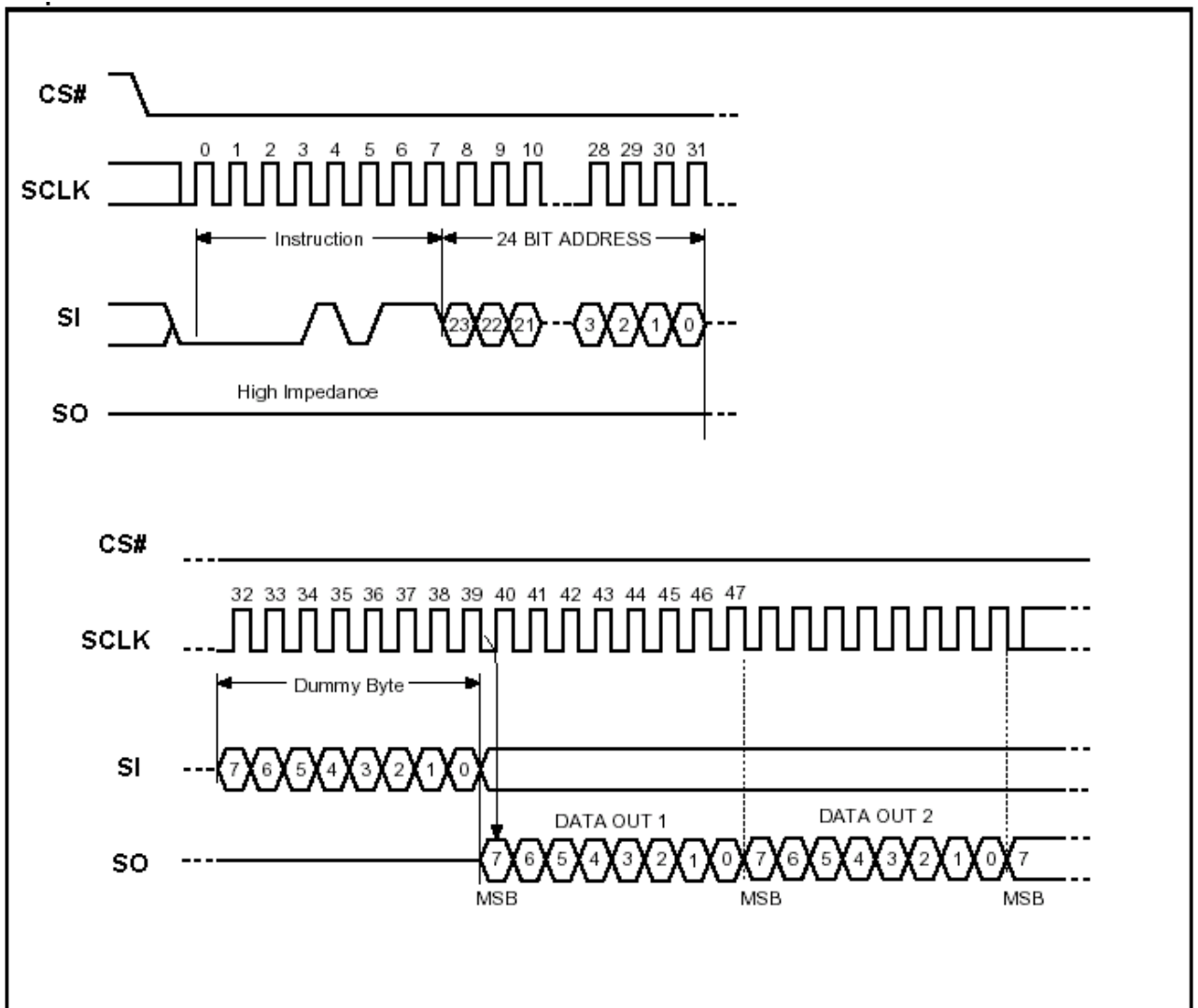
3.3 Read Data Bytes at Higher Speed (快速读取点阵数据)

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:



4 电气特性

4.1 绝对最大额定值

Symbol	Parameter	Min.	Max.	Unit	Condition
T _{OP}	Operating Temperature	-20	85	°C	
T _{STG}	Storage Temperature	-65	150	°C	
VCC	Supply Voltage	-0.3	3.6	V	
V _{IN}	Input Voltage	-0.3	VCC+0.3	V	
GND	Power Ground	-0.3	0.3	V	

4.2 DC 特性

Condition: T_{OP} = -20°C to 85°C, GND=0V

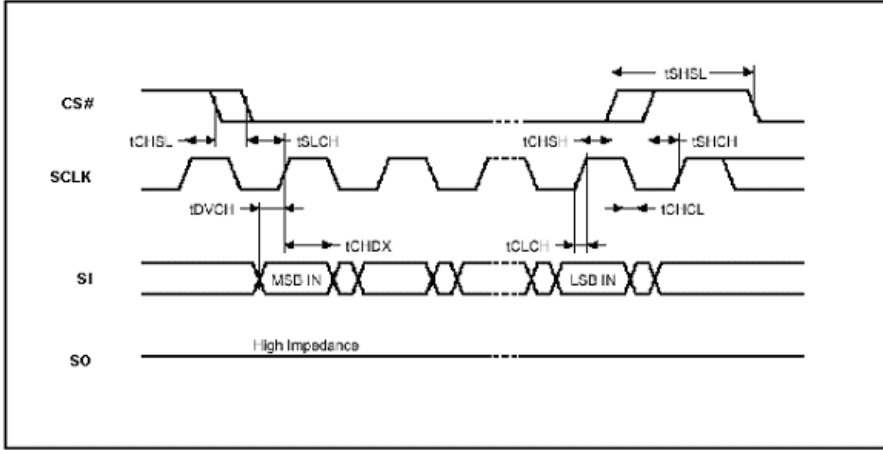
Symbol	Parameter	Min.	Max.	Unit	Condition
I _{DD}	VCC Supply Current(active)		12	mA	
I _{SB}	VCC Standby Current		10	uA	
V _{IL}	Input LOW Voltage	-0.3	0.3VCC	V	VCC=2.7-3.6V
V _{IH}	Input HIGH Voltage	0.7VCC	VCC+0.4	V	
V _{OL}	Output LOW Voltage		0.4 (I _{OL} =1.6mA)	V	
V _{OH}	Output HIGH Voltage	0.8VCC (I _{OH} =-100uA)		V	
I _{LI}	Input Leakage Current	0	2	uA	
I _{LO}	Output Leakage Current	0	2	uA	

Note: I_{IL}: Input LOW Current, I_{IH}: Input HIGH Current,
I_{OL}: Output LOW Current, I_{OH}: Output HIGH Current,

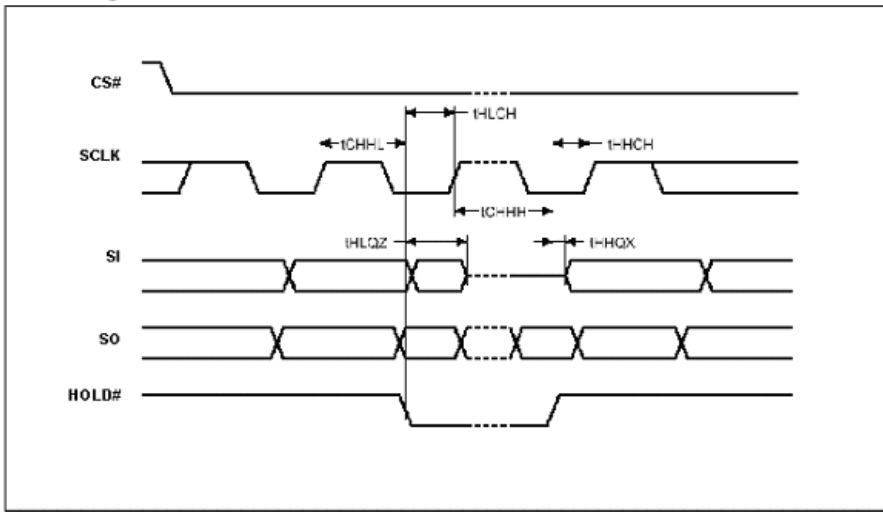
4.3 AC 特性

Symbol	Alt.	Parameter	Min.	Max.	Unit
Fc	Fc	Clock Frequency	D.C.	30	MHz
t _{CH}	t _{CLH}	Clock High Time	15		ns
t _{CL}	t _{CLL}	Clock Low Time	15		ns
t _{CLCH}		Clock Rise Time(peak to peak)	0.1		V/ns
t _{CHCL}		Clock Fall Time (peak to peak)	0.1		V/ns
t _{SLCH}	t _{css}	CS# Active Setup Time (relative to SCLK)	5		ns
t _{CHSL}		CS# Not Active Hold Time (relative to SCLK)	5		ns
t _{DVCH}	t _{dsu}	Data In Setup Time	2		ns
t _{CHDX}	t _{dh}	Data In Hold Time	5		ns
t _{CHSH}		CS# Active Hold Time (relative to SCLK)	5		ns
t _{SHCH}		CS# Not Active Setup Time (relative to SCLK)	5		ns
t _{SHSL}	t _{CSH}	CS# Deselect Time	100		ns
t _{SHQZ}	t _{dis}	Output Disable Time		9	ns
t _{CLQV}	t _v	Clock Low to Output Valid		9	ns
t _{CLQX}	t _{ho}	Output Hold Time	0		ns
t _{HLCH}		HOLD# Setup Time (relative to SCLK)	5		ns
t _{CHHH}		HOLD# Hold Time (relative to SCLK)	5		ns
t _{HHCH}		HOLD Setup Time (relative to SCLK)	5		ns
t _{CHHL}		HOLD Hold Time (relative to SCLK)	5		ns
t _{HHQX}	t _{lz}	HOLD to Output Low-Z		9	ns
t _{HLQZ}	t _{hz}	HOLD# to Output High-Z		9	ns

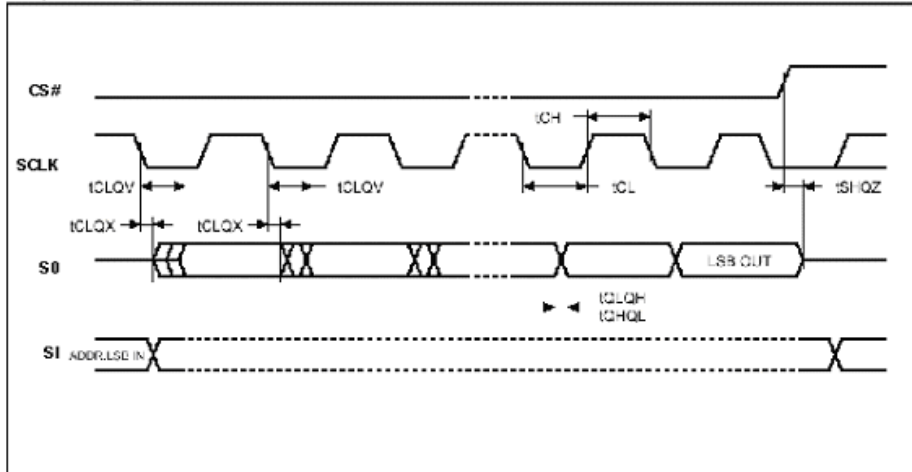
Serial Input Timing



Hold Timing



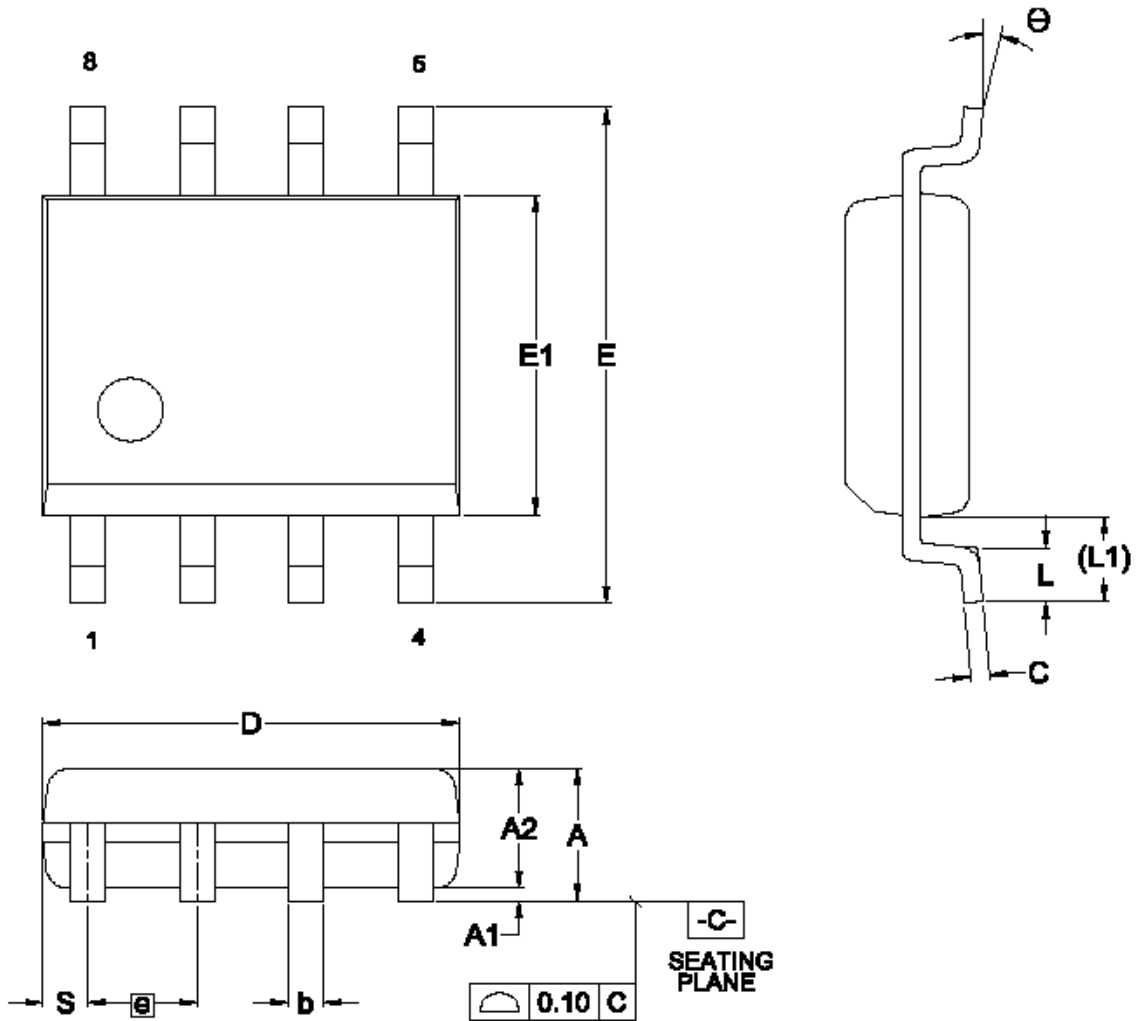
Output Timing



5 封装尺寸

SOP8 Package

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

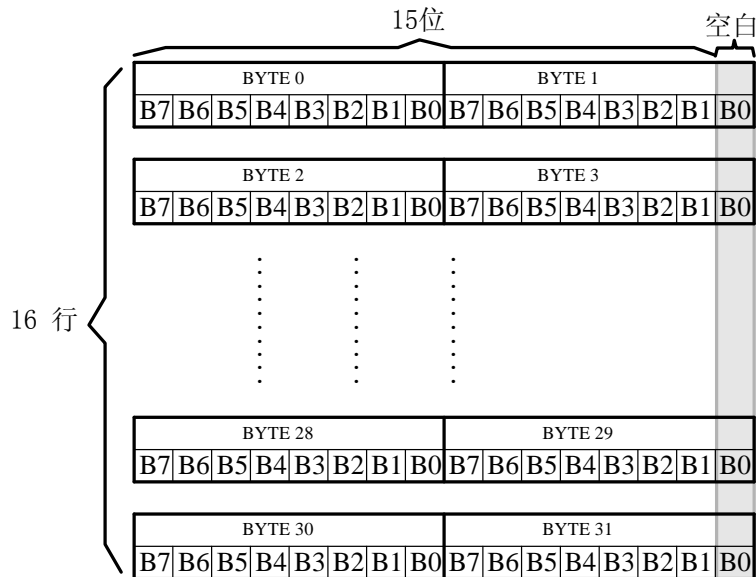
6 字库调用方法

6.1 字符点阵排列格式

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。点阵排列格式为横置横排：即一个字节的低位表示左面的点，高位表示右面的点，排满一行的点后再排下一行。这样把点阵信息用来直接在显示器上按上述规则显示，则将出现对应的汉字。

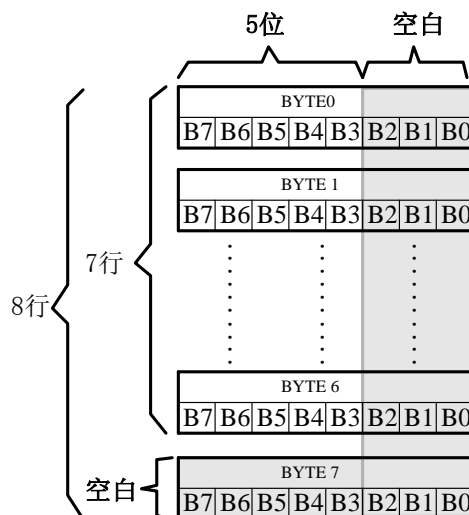
6.1.1 15X16 点字符排列格式

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



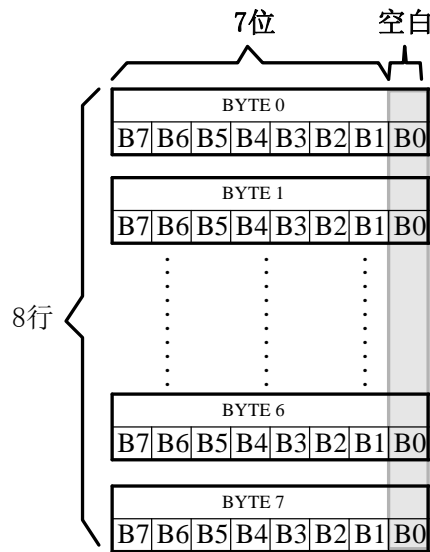
6.1.2 5X7 点 ASCII 字符排列格式

5X7 点 ASCII 的信息需要 8 个字节 (BYTE 0 – BYTE 7) 来表示。该 ASCII 点阵数据是横置横排的，其具体排列结构如下图：



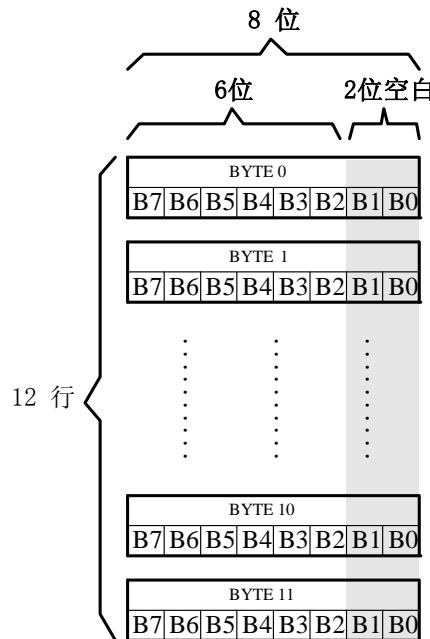
6.1.3 7X8 点 ASCII 字符排列格式

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



6.1.4 6X12 点 ASCII 字符排列格式

6X12 点 ASCII 的信息需要 12 个字节 (BYTE 0 – BYTE11) 来表示。该 ASCII 点阵数据是横置横排的，其具体排列结构如下图：



6.1.5 8X16 点字符排列格式

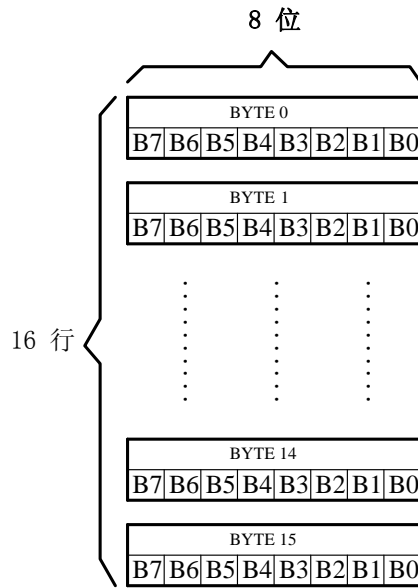
适用于此种排列格式的字体有：

8X16 点 ASCII 字符

8X16 点粗体 ASCII 字符

8X16 点拉丁文、希腊文、基里尔文字符

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



6.1.6 12 点阵不等宽字符排列格式

适用于此种排列格式的字体有：

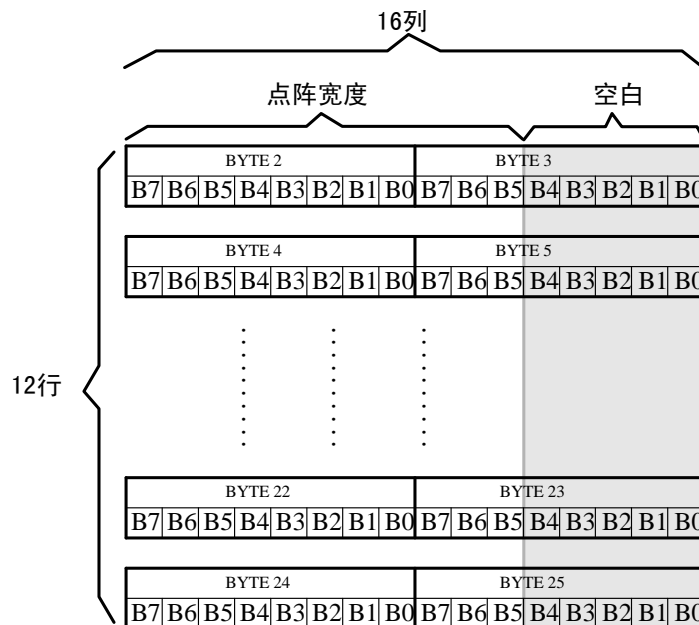
12 点阵不等宽 ASCII 方头 (Arial) 字符

12 点阵不等宽 Unicode 字符

12 点阵不等宽字符的信息需要 26 个字节 (BYTE 0 – BYTE25) 来表示。

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，BYTE2-25 存放横置横排点阵数据。

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。



6.1.7 16 点阵不等宽字符排列格式

适用于此种排列格式的字体有：

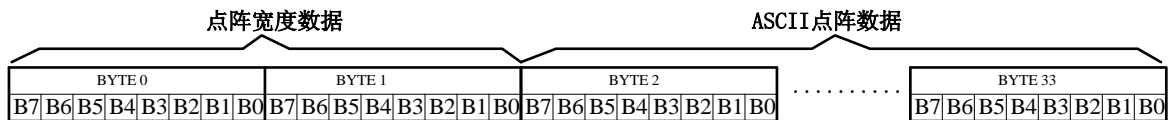
16 点阵不等宽 ASCII 方头（Arial）字符

16 点阵不等宽 Unicode 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。

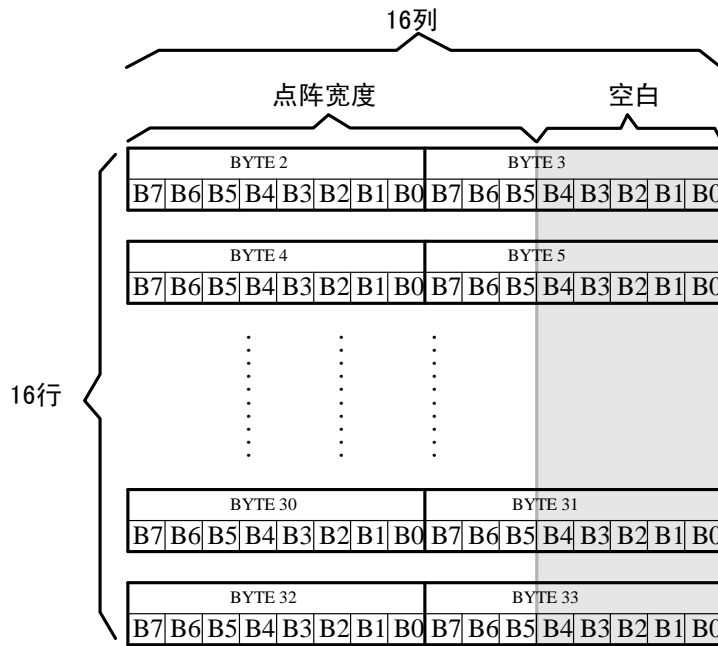
■ 存储格式

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 存放横置横排点阵数据。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 00 00 00 00 00 00 7F 80 7F C0 60 C0 60 C0 60 C0 7F 80 7F C0 60 E0 60 60 60 60 7F C0 7F 80 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 00 00 00 00 00 7F 80 7F C0 60 C0 60 C0 60 C0 7F 80 7F C0 60 E0 60 60 60 60 7F C0 7F 80 00 00 为 ASCII 方头字符 B 的点阵数据。

6.2 点阵字库地址表

	字库内容	编码体系	码位范围	字符数	起始地址	参考算法
1	15X16 点 GB12345 点阵字库	GB12345	A1A1-F9A9	6866+846	00000	6.3.1.1
2	5X7 点 ASCII 字符	ASCII	20~7F	96	51980	6.3.2.1
3	7X8 点 ASCII 字符	ASCII	20~7F	96	51C80	6.3.2.2
4	6X12 点 ASCII 字符	ASCII	20~7F	96	51F80	6.3.2.3
5	8X16 点 ASCII 字符	ASCII	20~7F	96	52580	6.3.2.4
6	8X16 点粗体 ASCII 字符	ASCII	20~7F	96	52D80	6.3.2.5
7	12 点阵不等宽 ASCII 方头 (Arial) 字符	ASCII	20~7F	96	53380	6.3.2.6
8	12 点阵不等宽 Unicode 字符 (拉丁文系、希腊文系、基里尔文系)	Unicode	0020-04E9	555	53380	6.3.3.4- 6.3.3.6
9	16 点阵不等宽 ASCII 方头 (Arial) 字符	ASCII	20~7F	96	56BDE	6.3.2.7
10	16 点阵不等宽 Unicode 字符 (拉丁文系、希腊文系、基里尔文系)	Unicode	0020-04E9	555	56BDE	6.3.3.7- 6.3.3.9
11	8X16 点拉丁文系字符	Unicode	00A0-0217	376	5B594	6.3.3.1
12	8X16 点希腊文系字符	Unicode	0370-03CF	96	5CD14	6.3.3.2
13	8X16 点基里尔文系字符	Unicode	0400-04F9	250	5D314	6.3.3.3
14	JIS 索引表	JIS0208	0101-8794	6398+1644	5E2B4	6.3.1.3
15	BIG5 基本集索引表	BIG5	A140-C67E	5401+408	62410	6.3.1.2
16	Unicode 索引表	Unicode	A0-FF50		651F0	6.3.1.4
17	16 点阵不等宽阿拉伯文系字符	Unicode	0600~06F9	250	715E0	6.3.3.12
18	16 点阵不等宽阿拉伯文系变体字符	自定义	B000-B1F1	498	73714	6.3.3.13
19	12 点阵不等宽阿拉伯文系字符	Unicode	0600~06F9	250	78790	6.3.3.10
20	12 点阵不等宽阿拉伯文系变体字符	自定义	B000-B1F1	498	7A0F4	6.3.3.11
21	保留区				7DE80	

6.3 字符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。

6.3.1 汉字及日文字符地址计算

6.3.1.1 15X16 点 GB12345 汉字字符

参数说明:

GBCode表示汉字内码。

MSB 表示汉字内码GBCode 的高8bits。

LSB 表示汉字内码GBCode 的低8bits。

Address 表示汉字或ASCII字符点阵在芯片中的字节地址。

BaseAdd: 说明点阵数据在字库芯片中的起始地址。

计算方法:

BaseAdd=0x00;

if(MSB >=0xA1 && MSB <= 0xA9 && LSB >=0xA1)

Address = ((MSB - 0xA1) * 94 + (LSB - 0xA1))*32+ BaseAdd;

else if(MSB >=0xB0 && MSB <= 0xF9 && LSB >=0xA1)

Address = ((MSB - 0xB0) * 94 + (LSB - 0xA1)+ 1038)*32+ BaseAdd;

6.3.1.2 15X16 点 BIG5 基本集汉字字符

参数说明:

FontCode表示汉字内码。

MSB 表示汉字内码FontCode的高8bits。

LSB 表示汉字内码FontCode的低8bits。

Address 表示汉字点阵在芯片中的字节地址。

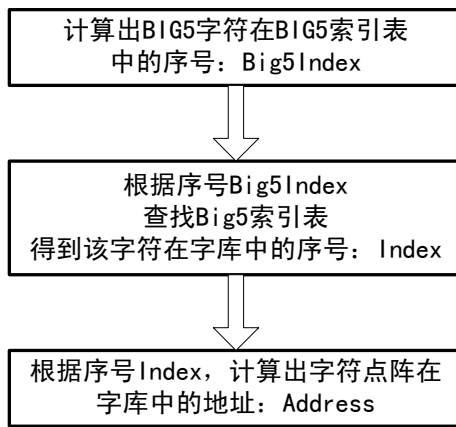
BaseAdd: 说明点阵数据在字库芯片中的起始地址。

Big5Table: 表示 BIG5 索引表。表的起始地址为 0x62410

Big5Index: 表示该内码汉字在 BIG5 索引表中的序号。由该序号可查 BIG5 索引表得出汉字在字库中的序号

Index: 表示该字符在字库中的序号。

计算方法:



```

BaseAdd=0x00;
if(MSB >=0xA1 && MSB <= 0XC6)
{
    if(LSB >=0x40 && LSB <= 0X7E)
        Big5Index =(MSB - 0xA1) * 157 + (LSB - 0x40);
    else if(LSB >=0XA1 && LSB <= 0XFE)
        Big5Index =(MSB - 0xA1) * 157 + 63 + (LSB - 0xA1));
}
Index = Big5Table[Big5Index*2] * 256 + Big5Table[Big5Index*2+1];
Address =Index * 32 + BaseAdd;
  
```

6.3.1.3 15X16 点 JIS 日文字符

参数说明:

FontCode表示日文内码。

MSB 表示日文字符的区号。

LSB 表示日文字符的位号。

Address 表示点阵在芯片中的字节地址。

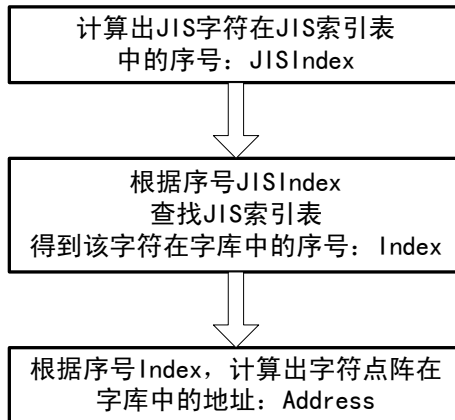
BaseAdd: 说明点阵数据在字库芯片中的起始地址。

JISTable: 表示 JIS 索引表。表的起始地址为 0x5E2B4

JISIndex: 表示该内码日文字符在 JIS 索引表中的序号。由该序号可查 JIS 索引表得出字符在字库中的序号

Index: 表示该字符在字库中的序号。

计算方法:



```

BaseAdd=0x00;
if ((MSB >=1 && MSB <= 94) && (LSB >=1 && LSB <= 94) )
{
    JISIndex = (MSB -1)*94 +LSB-1;
}
Index = JISTable[JISIndex *2] * 256 + JISTable[JISIndex *2+1] ;
Address =Index * 32 + BaseAdd;
    
```

6.3.1.4 15X16 点 Unicode 日文字符

参数说明:

FontCode表示日文内码。

Address 表示点阵在芯片中的字节地址。

BaseAdd: 说明点阵数据在字库芯片中的起始地址。

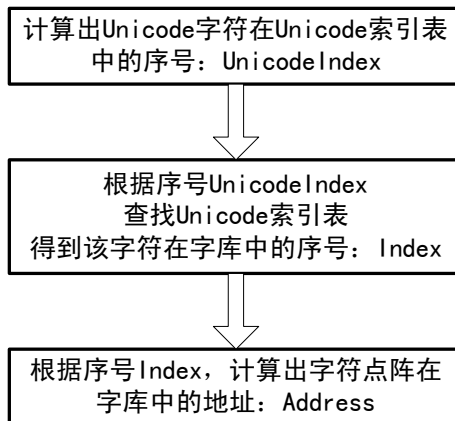
UnicodeTable: 表示 Unicode 索引表。表的起始地址为 0x651F0

UnicodeToIndex(): 根据 Unicode 内码计算出该字符在 Unicode 索引表中的序号。

UnicodeIndex: 表示该内码日文字符在 Unicode 索引表中的序号。由该序号可查 Unicode 索引表得出字符在字库中的序号

Index: 表示该字符在字库中的序号。

计算方法:



```

BaseAdd=0x00;
UnicodeIndex = UnicodeToIndex(FontCode);
Index = UnicodeTable [UnicodeIndex *2] * 256 + UnicodeTable [UnicodeIndex *2+1];
    
```

Address = Index * 32 + BaseAdd;

WORD UnicodeToIndex(WORD code)

```
{
    BYTE result=0;
    WORD h;

    if(code<=0x20) code = 0x3000;
    else if(code<0x7f)
    {
        code=0xfe57-code-0x21;
    }

    if(code<=0xa0) result=1;
    else if(code<=0x0451) h=code-160;
    else if(code< 0x2010) result=1;
    else if(code<=0x2642) h=code-160-7102;
    else if(code< 0x3000) result=1;
    else if(code<=0x33d5) h=code-160-7102-2493;
    else if(code< 0x4e00) result=1;
    else if(code<=0x9fa5) h=code-160-7102-2493-6698;
    else if(code< 0xe76c) result=1;
    else if(code<=0xe864) h=code-160-7102-2493-6698-18374;
    else if(code< 0xf92c) result=1;
    else if(code<=0xfa29) h=code-160-7102-2493-6698-18374-4295;
    else if(code< 0xfe30) result=1;
    else if(code<=0xfe6b) h=code-160-7102-2493-6698-18374-4295-1030;
    else if(code< 0xff01) result=1;
    else if(code<=0xff5e) h=code-160-7102-2493-6698-18374-4295-1030-149;
    else if(code< 0xffe0) result=1;
    else if(code<=0xffe5) h=code-160-7102-2493-6698-18374-4295-1030-149-129;
    else result=1;

    if(result==1)
    {
        h = 0x3000-160-7102-2493;
    }

    return h;
}
```

6.3.2 ASCII 字符的地址计算

6.3.2.1 5X7 点 ASCII 字符

参数说明:

ASCIICode: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x51980

if (ASCIICode >= 0x20) and (ASCIICode <= 0x7E) then

Address = (ASCIICode - 0x20) * 8 + BaseAdd

6.3.2.2 7X8 点 ASCII 字符

参数说明:

ASCIICode: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x51C80

if (ASCIICode >= 0x20) and (ASCIICode <= 0x7E) then

Address = (ASCIICode - 0x20) * 8 + BaseAdd

6.3.2.3 6X12 点 ASCII 字符

说明:

ASCIICode: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x51F80

if (ASCIICode >= 0x20) and (ASCIICode <= 0x7E) then

Address = (ASCIICode - 0x20) * 12 + BaseAdd

6.3.2.4 8X16 点 ASCII 字符

说明:

ASCIICode: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x52580

if (ASCIICode >= 0x20) and (ASCIICode <= 0x7E) then

Address = (ASCIICode - 0x20) * 16 + BaseAdd

6.3.2.5 8X16 点粗体 ASCII 字符

说明:

ASCIICode: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x52D80

if (ASCIICode >= 0x20) and (ASCIICode <= 0x7E) then

$$\text{Address} = (\text{ASCII Code} - 0x20) * 16 + \text{BaseAdd}$$

6.3.2.6 12 点阵不等宽 ASCII 方头 (Arial) 字符

说明:

ASCII Code: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x53380

if (ASCII Code >= 0x20) and (ASCII Code <= 0x7E) then

$$\text{Address} = (\text{ASCII Code} - 0x20) * 26 + \text{BaseAdd}$$

6.3.2.7 16 点阵不等宽 ASCII 方头 (Arial) 字符

说明:

ASCII Code: 表示 ASCII 码 (8bits)

BaseAdd: 说明该套字库在芯片中的起始地址。

Address: ASCII 字符点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x56BDE

if (ASCII Code >= 0x20) and (ASCII Code <= 0x7E) then

$$\text{Address} = (\text{ASCII Code} - 0x20) * 34 + \text{BaseAdd}$$

6.3.3 Unicode 字符的地址计算

6.3.3.1 8X16 点拉丁文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd = 0x5B594

if (FontCode >= 0x00A0) and (FontCode <= 0x0217) then

$$\text{Address} = (\text{FontCode} - 0x00A0) * 16 + \text{BaseAdd}$$

6.3.3.2 8X16 点希腊文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd = 0x5CD14

if (FontCode >= 0x0370) and (FontCode <= 0x03CF) then

$$\text{Address} = (\text{FontCode} - 0x0370) * 16 + \text{BaseAdd}$$

6.3.3.3 8X16 点基里尔文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x5D314

if (FontCode >= 0x0400) and (FontCode <=0x04F9) then

Address = (FontCode-0x0400) * 16+BaseAdd

6.3.3.4 12 点阵不等宽拉丁文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x53380

if (FontCode >= 0x0020) and (FontCode <=0x007F) then

Address = (FontCode-0x0020) * 26+BaseAdd

Else if (FontCode >= 0x00A0) and (FontCode <=0x017F) then

Address = (FontCode-0x0040) * 26+BaseAdd

Else if (FontCode == 0x018F) then

Address = 320*26+BaseAdd

Else if (FontCode == 0x0192) then

Address = 321*26+BaseAdd

Else if (FontCode == 0x01a0) then

Address = 322*26+BaseAdd

Else if (FontCode == 0x01a1) then

Address = 323*26+BaseAdd

Else if (FontCode == 0x01af) then

Address = 324*26+BaseAdd

Else if (FontCode == 0x01b0) then

Address = 325*26+BaseAdd

Else if (FontCode >= 0x01cd) and (FontCode <= 0x01dc) then

Address = (326+FontCode - 0x01cd)*26+BaseAdd

Else if (FontCode == 0x01dd) then

Address = 348*26+BaseAdd

Else if (FontCode >= 0x01e0) and (FontCode <= 0x01e3) then

Address = (342 + FontCode - 0x01e0)*26+BaseAdd

Else if (FontCode == 0x01fe) then

Address = 346*26+BaseAdd

Else if (FontCode == 0x01ff) then

Address = 347*26+BaseAdd

6.3.3.5 12 点阵不等宽希腊文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x53380+349*26

if (FontCode >= 0x037d) and (FontCode <=0x037f) then

Address = (FontCode-0x037d) * 26+BaseAdd

Else if (FontCode >= 0x0386) and (FontCode <=0x03CE) then

Address = (FontCode-0x0383) * 26+BaseAdd

6.3.3.6 12 点阵不等宽基里尔文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x53380+425*26

if (FontCode >= 0x0400) and (FontCode <=0x045F) then

Address = (FontCode-0x0400) * 26+BaseAdd

Else if (FontCode >= 0x0490) and (FontCode <=0x04a3) then

Address = (FontCode-0x 0490+96) * 26+BaseAdd

Else if (FontCode >= 0x04AE) and (FontCode <=0x04B5) then

Address = (FontCode-0x04AE+116) * 26+BaseAdd

Else if (FontCode >= 0x04BA) and (FontCode <=0x04BB) then

Address = (FontCode-0x04BA+124) * 26+BaseAdd

Else if (FontCode >= 0x04D8) and (FontCode <=0x04D9) then

Address = (FontCode-0x04D8+126) * 26+BaseAdd

Else if (FontCode >= 0x04E8) and (FontCode <=0x04E9) then

Address = (FontCode-0x04E8+128) * 26+BaseAdd

6.3.3.7 16 点阵不等宽拉丁文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x56BDE

if (FontCode >= 0x0020) and (FontCode <=0x007F) then

Address = (FontCode-0x0020) * 34+BaseAdd

Else if (FontCode >= 0x00A0) and (FontCode <=0x017F) then

Address = (FontCode-0x0040) * 34+BaseAdd

Else if (FontCode == 0x018F) then

Address = 320*34+BaseAdd


```

Else if (FontCode == 0x0192) then
    Address = 321*34+BaseAdd
Else if (FontCode == 0x01a0) then
    Address = 322*34+BaseAdd
Else if (FontCode == 0x01a1) then
    Address = 323*34+BaseAdd
Else if (FontCode == 0x01af) then
    Address = 324*34+BaseAdd
Else if (FontCode == 0x01b0) then
    Address = 325*34+BaseAdd
Else if (FontCode >= 0x01cd) and (FontCode <= 0x01dc) then
    Address = (326+FontCode - 0x01cd)*34+BaseAdd
Else if (FontCode == 0x01dd) then
    Address = 348*34+BaseAdd
Else if (FontCode >= 0x01e0) and (FontCode <= 0x01e3) then
    Address = (342 + FontCode - 0x01e0)*34+BaseAdd
Else if (FontCode == 0x01fe) then
    Address = 346*34+BaseAdd
Else if (FontCode == 0x01ff) then
    Address = 347*34+BaseAdd

```

6.3.3.8 16 点阵不等宽希腊文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

```

BaseAdd=0x56BDE +349*34
if (FontCode >= 0x037d) and (FontCode <=0x037f) then
    Address = (FontCode-0x037d) * 34+BaseAdd
Else if (FontCode >= 0x0386) and (FontCode <=0x03CE) then
    Address = (FontCode-0x0383) * 34+BaseAdd

```

6.3.3.9 16 点阵不等宽基里尔文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

```

BaseAdd=0x56BDE +425*34
if (FontCode >= 0x0400) and (FontCode <=0x045F) then
    Address = (FontCode-0x0400) * 34+BaseAdd
Else if (FontCode >= 0x0490) and (FontCode <=0x04a3) then
    Address = (FontCode-0x 0490+96) * 34+BaseAdd
Else if (FontCode >= 0x04AE) and (FontCode <=0x04B5) then

```

```

Address = (FontCode-0x04AE+116) * 34+BaseAdd
Else if (FontCode >= 0x04BA) and (FontCode <=0x04BB) then
    Address = (FontCode-0x04BA+124) * 34+BaseAdd
Else if (FontCode >= 0x04D8) and (FontCode <=0x04D9) then
    Address = (FontCode-0x04D8+126) * 34+BaseAdd
Else if (FontCode >= 0x04E8) and (FontCode <=0x04E9) then
    Address = (FontCode-0x04E8+128) * 34+BaseAdd

```

6.3.3.10 12 点阵不等宽阿拉伯文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x78790

```

if (FontCode >= 0x0600) and (FontCode <=0x06F9) then
    Address = (FontCode-0x0600) * 26+BaseAdd

```

6.3.3.11 12 点阵不等宽阿拉伯文系变体字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

FontCode: 表示字符内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x7A0F4

```

if (FontCode >= 0xB000) and (FontCode <=0XB1F1) then
    Address = (FontCode-0xB000) * 26+BaseAdd

```

6.3.3.12 16 点阵不等宽阿拉伯文系字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

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:

BaseAdd=0x715E0

```

if (FontCode >= 0x0600) and (FontCode <=0x06F9) then
    Address = (FontCode-0x0600) * 34+BaseAdd

```

6.3.3.13 16 点阵不等宽阿拉伯文系变体字符

说明:

BaseAdd: 说明本套字库在字库芯片中的起始字节地址。

FontCode: 表示字符内码 (16bits)

Address: 表示汉字点阵在芯片中的字节地址。

计算方法:

BaseAdd=0x73714

```

if (FontCode >= 0xB000) and (FontCode <=0XB1F1) then

```

Address = (FontCode-0xB000) * 34+BaseAdd

7 附录

7.1 GB12345 1 区字符 (846 字符)

GB212345 标准点阵字符 1 区对应码位的 A1A1~A9EF 共计 846 个字符:

GB12345 1 区

A1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A			、	。	·	-	√	”	”	々	—	~		…	‘	’
B	“	”	{	}	<	>	《	》	「	」	『	』	【	】	【	】
C	±	×	÷	:	∧	∨	Σ	Π	U	∩	€	::	√	⊥	#	∠
D	∩	⊙	∫	∫	≡	≈	≈	∞	≠	≠	≠	≠	≠	≠	∞	∴
E	∴	↑	♀	°	'	”	℃	\$	⊗	⊗	£	%	§	No	☆	★
F	○	●	◎	◇	◆	□	■	△	▲	※	→	←	↑	↓	=	

A2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		i	ii	iii	iv	v	vi	vii	viii	ix	x					
B		1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.
C	16.	17.	18.	19.	20.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
D	(12)	(13)	(14)	(15)	(16)	(17)	(18)	(19)	(20)	①	②	③	④	⑤	⑥	⑦
E	⑧	⑨	⑩	€		(一)	(二)	(三)	(四)	(五)	(六)	(七)	(八)	(九)	(十)	
F		I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII			

A3	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		!	”	#	¥	%	&	'	()	*	+	,	-	.	/
B	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
C	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
D	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
E	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
F	p	q	r	s	t	u	v	w	x	y	z	{		}	—	

GB12345 1 区

A4	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		あ	あ	い	い	う	う	え	え	お	お	か	か	き	き	く
B	ぐ	け	げ	こ	こ	さ	さ	し	し	ず	ず	せ	せ	そ	そ	た
C	だ	ち	ち	っ	っ	づ	づ	て	て	と	と	な	に	ぬ	ぬ	は
D	ば	ば	ひ	ひ	び	ふ	ふ	ぶ	ぶ	へ	へ	べ	べ	ぼ	ぼ	ま
E	む	め	も	や	や	ゆ	ゆ	よ	よ	ら	ら	る	れ	ろ	わ	わ
F	ゐ	ゑ	を	ん												

A5	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		ア	ア	イ	イ	ウ	ウ	エ	エ	オ	オ	カ	カ	キ	キ	ク
B	グ	ケ	ケ	コ	コ	サ	サ	シ	シ	ズ	ズ	セ	セ	ソ	ソ	タ
C	ダ	チ	チ	ツ	ツ	ヅ	ヅ	テ	テ	ト	ト	ナ	ニ	ヌ	ヌ	ハ
D	バ	パ	ヒ	ヒ	ピ	フ	フ	ブ	ブ	ヘ	ヘ	ベ	ベ	ボ	ボ	マ
E	ム	メ	モ	ヤ	ヤ	ユ	ユ	ヨ	ヨ	ラ	リ	ル	レ	ロ	ワ	ワ
F	ヰ	ヱ	ヲ	ン	ヅ	カ	ケ									

A6	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		A	B	Γ	Δ	E	Z	H	Θ	I	K	Λ	M	N	Ξ	Ο
B	Π	P	Σ	T	Ι	Φ	X	Ψ	Ω							
C		α	β	γ	δ	ε	ξ	η	θ	ι	κ	λ	μ	ν	ξ	ο
D	π	ρ	σ	τ	υ	φ	χ	ψ	ω	'	°	`	:	;	!	?
E	∧	∨	∩	∪	∩	∪	≅	≅	∩	∩	∩	∩	∩	∩	∩	∩
F	∩	∩		∩		∩										

GB12345 1 区

A7	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н
B	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э
C	Ю	Я														
D		а	б	в	г	д	е	ё	ж	з	и	й	к	л	м	н
E	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э
F	ю	я														

A8	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		ā	á	ǎ	à	ē	é	ě	è	ī	í	ǎ	ì	ō	ó	ǒ
B	ò	ū	ú	ǔ	ù	ǖ	ú	ǘ	ù	ü	ê	ɑ	ń	ń	ň	ñ
C	g				勺	夕	冂	匚	勹	去	勹	勹	《	丐	厂	
D	ㄣ	ㄨ	ㄒ	ㄒ	尸	冂	冂	ㄣ	厶	冫	冫	冫	世	𠂇	丿	ㄥ
E	纟	又	𠂇	𠂇	尢	厶	儿	丨	乂	冫						
F																

A9	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A					—	—			---	---	!	!	---	---	!	!
B	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌	┌
C	└	└	└	└	└	└	└	└	└	└	└	└	└	└	└	└
D	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘	┘
E	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
F																

7.2 Unicode 字符区字符

Unicode 字符区共收录 拉丁文系 (LATIN)、希腊文系 (GREEK)、基里尔文系 (CYRILLIC)，共计收录字符 456 个。阿拉伯文系 (ARABIC)，共计收录字符 250 个。

7.2.1 8×16 点拉丁文系 (376 字符)

Unicode 字符区-拉丁文系的内码范围为 00A0~0217 共计收录 376 个字符。

*说明：由于拉丁文系 (BASIC LATIN) 的内码范围 0020~007E 与基本 ASCII 编码一致 (可以共用)，因此未列入 Unicode 字符区-拉丁文系中。

Unicode 字符区-拉丁文系

00	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A	□	¡	¢	£	¤	¥	¦	§	¨	©	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

01	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	Ā	ā	Ǻ	ǻ	Ȧ	ȧ	Č	č	Ĉ	ĉ	Ċ	ċ	Č	č	Ď	ď
1	Đ	đ	Ē	ē	Ě	ě	É	é	Ě	ě	Ĝ	ĝ	Ĝ	ĝ	Ğ	ğ
2	Ġ	ġ	Ģ	ģ	Ĥ	ĥ	Ħ	ħ	İ	ı	Ī	ī	Ĭ	ĭ	Į	į
3	İ	ı	Ĵ	ĵ	Ķ	ķ	κ	Ł	ł	Ł	ł	Ł	ł	Ł	ł	Ł
4	Ł	ł	Ń	ń	Ņ	ņ	Ň	ň	Ŋ	ŋ	Ō	ō	Ŏ	ö		
5	Ŏ	ö	Œ	œ	Ŕ	ŕ	Ŗ	ŗ	Ŗ	ŗ	Ś	ś	Ŝ	ŝ	Ş	ş
6	Ŝ	ŝ	Ţ	ţ	Ť	ť	Ŧ	ŧ	Ū	ū	Ū	ū	Ŭ	ŭ	Ů	ů
7	Ů	ů	Ū	ū	Ŵ	ŵ	Ŷ	ŷ	Ÿ	Ź	ź	Ż	ż	Ž	ž	f

01	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ƀ	Ɓ	Ƃ	ƃ	Ƅ	ƅ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə
9	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
A	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
B	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
C	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
D	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
E	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ
F	Ɛ	Ɔ	Ƈ	ƈ	Ɖ	Ɗ	Ƌ	ƌ	ƍ	Ǝ	Ə	Ɛ	Ɔ	Ƈ	ƈ	Ɖ

02	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	Ä ä	Å å	Ë ë	Ê ê	Ï ï	Î î	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï	Ï ï
1	Ŕ ŕ	Ŗ ŗ	Ů ů	Ű ű	Ų ų												

7.2.2 8×16 点基里尔文系（250 字符）

Unicode 字符区-希腊文系的内码范围为 0400~04F9 共计收录 250 个字符。

Unicode 字符区-基里尔文系

04	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		Ё	Ђ	Ѓ	Є	Ѕ	І	Ї	Ј	Љ	Њ	Ћ	Ќ	Й	Ў	Ц
1	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
2	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
3	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
4	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
5		ё	ђ	ѓ	є	ѕ	і	ї	ј	љ	њ	ћ	ќ		џ	ц
6	Ω	ω	Ъ	ь	Ё	Є	Λ	λ	Ѓ	ѓ	Ж	ж	Ѓ	Ѓ	Ѓ	Ѓ
7	Ψ	ψ	Θ	θ	Υ	υ	Ϛ	ϛ	ου	ου	Ο	ο	Ω	ω	Ω	ω

04	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	Q	Q	※	Г	Г	Г	Г									
9	Г	Г	F	f	Б	Б	Ж	Ж	Э	Э	К	к	К	к	К	к
A	К	к	Н	н	Н	н	Ь	ь	Q	Q	С	с	Т	т	У	у
B	Ү	ү	Х	х	Ц	ц	Ч	ч	Ч	ч	н	н	Е	е	Е	е
C	І	Ж	Ж	Б	Б			Н	н			Ч	ч			
D	Ä	ä	Ä	ä	Æ	æ	Ё	ё	Ө	ө	Ө	ө	Ж	ж	Э	э
E	Э	э	Й	й	Й	й	Ö	ö	Ө	ө	Ө	ө			У	у
F	Ü	ü	Ü	ü	Ч	ч			Ы	ы						

7.2.3 8×16 点希腊文系（96 字符）

Unicode 字符区-希腊文系的内码范围为 0370~03CF 共计收录 96 个字符。

Unicode 字符区-希腊文系

03	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
7					'	,					,				;	
8					'	ˆ	Α	·	Ε	Η	Ι		Ο		Τ	Ω
9	ι	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
A	Π	Ρ		Σ	Τ	Υ	Φ	Χ	Ψ	Ω	Ϊ	Ϋ	ά	έ	ή	ί
B	ϐ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
C	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	ϊ	ϋ	ό	ύ	ώ	

7.2.4 16 点阵阿拉伯文系（250 字符）

Unicode 字符区-阿拉伯文系的内码范围为 0600~06F9 共计收录 250 个字符。

06	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0													،			
1												ء				؟
2		ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
3	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ					
4	-	ف	ق	ك	ل	م	ن	ه	و	ى	ي	°	´	ˆ	˙	˚
5	-	˘	˙													
6	*	١	٢	٣	٤	٥	٦	٧	٨	٩	/	,	.	*		
7	'	أ	إ	أ	ء	أ	ؤ	ئ	ى	ث	ن	ب	ب	ن	ب	ن

06	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	ب	ع	خ	ج	ح	خ	ج	ح	ط	ظ	د	ذ	ذ	ذ	ذ	ذ
9	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ	س	ش	ص	ض	ط
A	غ	ف	ق	ك	ل	م	ن	ه	و	ى	ي	ك	ك	ك	ك	ك
B	ه	و	ى	ي	ك	ل	م	ن	ه	و	ى	ي	ك	ل	م	ن
C	ة	ء	آ	أ	ؤ	إ	ئ	ا	ب	ة	ت	ث	ج	ح	خ	د
D	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ	س	ش	ص	ض	ط
E	'	˘	˙	ˆ	˙	˚	°	´	ˆ	˙	˚	°	´	ˆ	˙	˚
F	*	١	٢	٣	٤	٥	٦	٧	٨	٩						

7.2.5 16 点阵阿拉伯文系变体字符（498 字符）

B0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C																
D																
E																
F																

B1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
3	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
4	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
5	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
6	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
7	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
9	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
B	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
C	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
D	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
E	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
F	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

GT21L16T1W

繁日外

字库芯片

标准字库

繁日外字库

7.3 国家地区语言文系 Unicode 对照表（150 国）

序号	国家	区域	语言	文系
1	马来西亚	亚洲	马来语	拉丁文系
2	文莱	亚洲	马来语、英语	拉丁文系
3	印度尼西亚	亚洲	印尼语	拉丁文系
4	菲律宾	亚洲	英语	拉丁文系
5	锡金	亚洲	英语	拉丁文系
6	英国	欧洲	英语	拉丁文系
7	爱尔兰	欧洲	英语	拉丁文系
8	美国	北美洲	英语	拉丁文系
9	加拿大	北美洲	英语、法语	拉丁文系
10	澳大利亚	大洋洲	英语	拉丁文系
11	新西兰	大洋洲	英语	拉丁文系
12	德国	欧洲	德语	拉丁文系
13	瑞士	欧洲	德语、法语	拉丁文系
14	奥地利	欧洲	德语	拉丁文系
15	卢森堡	欧洲	德语、法语	拉丁文系
16	列支敦士登	欧洲	德语	拉丁文系
17	意大利	欧洲	意大利语	拉丁文系
18	梵蒂冈	欧洲	意大利语	拉丁文系
19	圣马力诺	欧洲	意大利语	拉丁文系
20	丹麦	欧洲	丹麦语	拉丁文系
21	冰岛	欧洲	冰岛语	拉丁文系
22	挪威	欧洲	挪威语	拉丁文系
23	瑞典	欧洲	瑞典语	拉丁文系
24	芬兰	欧洲	芬兰语、瑞典语	拉丁文系
25	荷兰	欧洲	荷兰语	拉丁文系
26	苏里南	南美洲	荷兰语	拉丁文系
27	法罗群岛	欧洲	法罗语	拉丁文系
28	葡萄牙	欧洲	葡萄牙语	拉丁文系
29	巴西	南美洲	葡萄牙语	拉丁文系
30	佛得角	非洲	葡萄牙语	拉丁文系
31	几内亚比绍	非洲	葡萄牙语	拉丁文系
32	圣多美和普林西比	非洲	葡萄牙语	拉丁文系
33	安哥拉	非洲	葡萄牙语	拉丁文系
34	莫桑比克	非洲	葡萄牙语	拉丁文系
35	法国	欧洲	法语	拉丁文系
36	比利时	欧洲	法语、荷兰语	拉丁文系
37	摩纳哥	欧洲	法语、意大利语	拉丁文系
38	海地	北美洲	法语	拉丁文系
39	塞内加尔	非洲	法语	拉丁文系
40	马里	非洲	法语	拉丁文系
41	布基纳法索	非洲	法语	拉丁文系
42	几内亚比绍	非洲	法语	拉丁文系
43	科特迪瓦	非洲	法语	拉丁文系

序号	国家	区域	语言	文系
44	多哥	非洲	法语	拉丁文系
45	贝宁	非洲	法语	拉丁文系
46	尼日尔	非洲	法语	拉丁文系
47	喀麦隆	非洲	法语	拉丁文系
48	乍得	非洲	法语	拉丁文系
49	中非	非洲	法语	拉丁文系
50	吉布提	非洲	法语	拉丁文系
51	布隆迪	非洲	法语	拉丁文系
52	民主刚果	非洲	法语	拉丁文系
53	刚果	非洲	法语	拉丁文系
54	加蓬	非洲	法语	拉丁文系
55	科摩多	非洲	法语	拉丁文系
56	马达加斯加	非洲	法语	拉丁文系
57	西班牙	欧洲	西班牙语、加泰隆语	拉丁文系
58	墨西哥	北美洲	西班牙语	拉丁文系
59	危地马拉	北美洲	西班牙语	拉丁文系
60	哥斯达黎加	北美洲	西班牙语	拉丁文系
61	巴拿马	北美洲	西班牙语	拉丁文系
62	多米尼加	北美洲	西班牙语	拉丁文系
63	萨尔瓦多	北美洲	西班牙语	拉丁文系
64	洪都拉斯	北美洲	西班牙语	拉丁文系
65	尼加拉瓜	北美洲	西班牙语	拉丁文系
66	波多黎各	北美洲	西班牙语	拉丁文系
67	古巴	北美洲	西班牙语	拉丁文系
68	委内瑞拉	南美洲	西班牙语	拉丁文系
69	哥伦比亚	南美洲	西班牙语	拉丁文系
70	秘鲁	南美洲	西班牙语	拉丁文系
71	阿根廷	南美洲	西班牙语	拉丁文系
72	厄瓜多尔	南美洲	西班牙语	拉丁文系
73	智利	南美洲	西班牙语	拉丁文系
74	乌拉圭	南美洲	西班牙语	拉丁文系
75	巴拉圭	南美洲	西班牙语	拉丁文系
76	玻利维亚	南美洲	西班牙语	拉丁文系
77	赤道新几内亚	非洲	西班牙语	拉丁文系
78	休达和梅利亚	非洲	西班牙语	拉丁文系
79	牙买加	北美洲	英语	拉丁文系
80	伯利兹	北美洲	英语	拉丁文系
81	特立尼达和多巴哥	北美洲	英语	拉丁文系
82	巴哈马	北美洲	英语	拉丁文系
83	安提瓜和巴布达	北美洲	英语	拉丁文系
84	多米尼加	北美洲	英语	拉丁文系
85	圣文森特	北美洲	英语	拉丁文系
86	格林纳达	北美洲	英语	拉丁文系
87	开曼群岛	北美洲	英语	拉丁文系

序号	国家	区域	语言	文系
88	圣基茨-尼维斯	北美洲	英语	拉丁文系
89	汤加	大洋洲	英语	拉丁文系
90	斐济	大洋洲	英语	拉丁文系
91	所罗门	大洋洲	英语	拉丁文系
92	瓦努阿图	大洋洲	英语	拉丁文系
93	基里巴斯	大洋洲	英语	拉丁文系
94	瑙鲁	大洋洲	英语	拉丁文系
95	马绍尔群岛	大洋洲	英语	拉丁文系
96	津巴布韦	非洲	英语	拉丁文系
97	冈比亚	非洲	英语	拉丁文系
98	塞拉利昂	非洲	英语	拉丁文系
99	利比里亚	非洲	英语	拉丁文系
100	加纳	非洲	英语	拉丁文系
101	尼日利亚	非洲	英语	拉丁文系
102	乌干达	非洲	英语	拉丁文系
103	赞比亚	非洲	英语	拉丁文系
104	马拉维	非洲	英语	拉丁文系
105	塞舌尔	非洲	英语	拉丁文系
106	毛里求斯	非洲	英语	拉丁文系
107	博茨瓦纳	非洲	英语	拉丁文系
108	纳米比亚	非洲	英语	拉丁文系
109	莱索托	非洲	英语	拉丁文系
110	南非	非洲	南非荷兰语、英语	拉丁文系
111	肯尼亚	非洲	斯瓦西里语	拉丁文系
112	坦桑尼亚	非洲	斯瓦西里语	拉丁文系
113	埃及	非洲	阿拉伯语	阿拉伯文系
114	突尼斯	非洲	阿拉伯语	阿拉伯文系
115	利比亚	非洲	阿拉伯语	阿拉伯文系
116	摩洛哥	非洲	阿拉伯语	阿拉伯文系
117	阿尔及利亚	非洲	阿拉伯语	阿拉伯文系
118	苏丹	非洲	阿拉伯语	阿拉伯文系
119	索马里	非洲	阿拉伯语	阿拉伯文系
120	吉布提	非洲	阿拉伯语	阿拉伯文系
121	毛里塔尼亚	非洲	阿拉伯语	阿拉伯文系
122	叙利亚	亚洲	阿拉伯语	阿拉伯文系
123	阿联酋	亚洲	阿拉伯语	阿拉伯文系
124	黎巴嫩	亚洲	阿拉伯语	阿拉伯文系
125	也门	亚洲	阿拉伯语	阿拉伯文系
126	科威特	亚洲	阿拉伯语	阿拉伯文系
127	卡塔尔	亚洲	阿拉伯语	阿拉伯文系
128	巴勒斯坦	亚洲	阿拉伯语	阿拉伯文系
129	巴林	亚洲	阿拉伯语	阿拉伯文系
130	阿曼	亚洲	阿拉伯语	阿拉伯文系
131	约旦	亚洲	阿拉伯语	阿拉伯文系

序号	国家	区域	语言	文系
132	伊拉克	亚洲	阿拉伯语	阿拉伯文系
133	沙特阿拉伯	亚洲	阿拉伯语	阿拉伯文系
134	俄罗斯	欧洲	俄语	基里尔文系
135	白俄罗斯	欧洲	俄语	基里尔文系
136	乌克兰	欧洲	乌克兰语	基里尔文系
137	保加利亚	欧洲	保加利亚语	基里尔文系
138	马其顿	欧洲	马其顿语	基里尔文系
139	南斯拉夫联盟	欧洲	塞尔维亚语	基里尔文系
140	克罗地亚	欧洲	塞尔维亚语	基里尔文系
141	波黑	欧洲	塞尔维亚语	基里尔文系
142	阿塞拜疆	亚洲	阿塞拜疆语	基里尔文系
143	吉尔吉斯斯坦	亚洲	吉尔吉斯语	基里尔文系
144	塔吉克斯坦	亚洲	塔吉克语	基里尔文系
145	土库曼斯坦	亚洲	土库曼语	基里尔文系
146	乌兹别克斯坦	亚洲	乌兹别克语	基里尔文系
147	哈萨克斯坦	亚洲	哈萨克语	基里尔文系
148	蒙古	亚洲	蒙古语	基里尔文系
149	希腊	欧洲	希腊语	希腊文系
150	塞浦路斯	亚洲	希腊语	希腊文系

在 150 国语言文字中，拉丁文系为 112 国；阿拉伯文系为 21 国；基里尔文系为 15 国；希腊文系为 2 国。

其中拉丁文系中，英语为 39 国；法语为 22 国；西班牙语为 22 国；葡萄牙语为 7 国；德语为 5 国；意大利语为 3 国；马来语为 2 国；斯瓦希里语为 2 国，其他各种拉丁语为 10 国。

7.4 补丁文件

序号	补丁描述	修补方式	对应文档	版本号	发布时间
1	16点 UNICODE 对应字库表。	码本对应表	《GT21L16T1W-16-UNICODE 字库码本》	V1.00	2010-8-20
2	16点 GB12345 对应字库表	码本对应表	《GT21L16T1W-16-GB12345 字库码本》	V1.00	2010-8-20
3	16点 JIS0208 对应字库表	码本对应表	《GT21L16T1W-16-JIS0208 字库码本》	V1.00	2010-8-20
4	JIS0208 补丁程序	JIS0208 补丁程序	《GT21L16T1W JIS0208 补丁程序》	V1.00	2010-8-20

GT21L16T1W

繁日外

字库芯片

标准字库

繁日外字库