

## 1 概述

SD16S1Y是一款内含16x16点阵的汉字库芯片，支持GB2312国标简体汉字（含有国家信标委合法授权）、ASCII字符。排列格式为竖置横排。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。

SD16S1Y除含有上述字库以外，还提供16个扇区，每个扇区4K字节或16页，每页256字节，可自由写入空间地址范围为：6FFFF-7FFFF。仅支持上位机烧录，可重复擦写10万次以上。

### 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
- 时钟频率：80MHz @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：-40℃~85℃
- 封装：SOT23-6
- 字符集：
  - GB2312
  
- 字号：16x16 点阵

## 1.2 芯片内容

字符集	字库	字号	字符数	字体	排列方式
ASCII 字符集	ASCII	5x7	96	标准	Y-竖置横排
	ASCII	7x8	96	标准	Y-竖置横排
	ASCII	8x16	96	标准	Y-竖置横排
	ASCII	8x16	96	粗体	Y-竖置横排
	ASCII	16点阵不等宽	96	圆角字体	Y-竖置横排
	ASCII	16点阵不等宽	96	线型字体	Y-竖置横排
数字 字符集	数字	16点阵不等宽	15	线型字体(粗体)	Y-竖置横排
	数字	16点阵不等宽	15	圆角字体(粗体)	Y-竖置横排
	数字	24点阵不等宽	15	线型字体(粗体)	Y-竖置横排
	数字	24点阵不等宽	15	圆角字体(粗体)	Y-竖置横排
	数字	32点阵不等宽	15	线型字体(粗体)	Y-竖置横排
	数字	32点阵不等宽	15	圆角字体(粗体)	Y-竖置横排
	数字	48点阵不等宽	15	圆角字体(粗体)	Y-竖置横排
	数字	48点阵不等宽	15	线型字体(粗体)	Y-竖置横排
	数字	64点阵不等宽	15	圆角字体(粗体)	Y-竖置横排
	数字	64点阵不等宽	15	线型字体(粗体)	Y-竖置横排
	数字	16点阵不等宽	11	方块体	Y-竖置横排
	数字	24点阵不等宽	11	方块体	Y-竖置横排
	数字	32点阵不等宽	11	方块体	Y-竖置横排
	数字	48点阵不等宽	11	方块体	Y-竖置横排
	数字	64点阵不等宽	11	方块体	Y-竖置横排
	数字	24点阵不等宽	12	时钟体	Y-竖置横排
	数字	32点阵不等宽	12	时钟体	Y-竖置横排
	数字	48点阵不等宽	12	时钟体	Y-竖置横排
	数字	64点阵不等宽	12	时钟体	Y-竖置横排
	汉字 字符集	GB2312 汉字	16x16	6763	宋体
GB2312 字符		16x16	376	宋体	Y-竖置横排
扩展字符		16x16	126	宋体	Y-竖置横排
UNICODE- GB 转码表					

### 1.3 字型样张

#### 1.3.1 汉字字符

16x16 点阵 GB2312 汉字



#### 1.3.2 其它点阵字符

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	{		}	~	

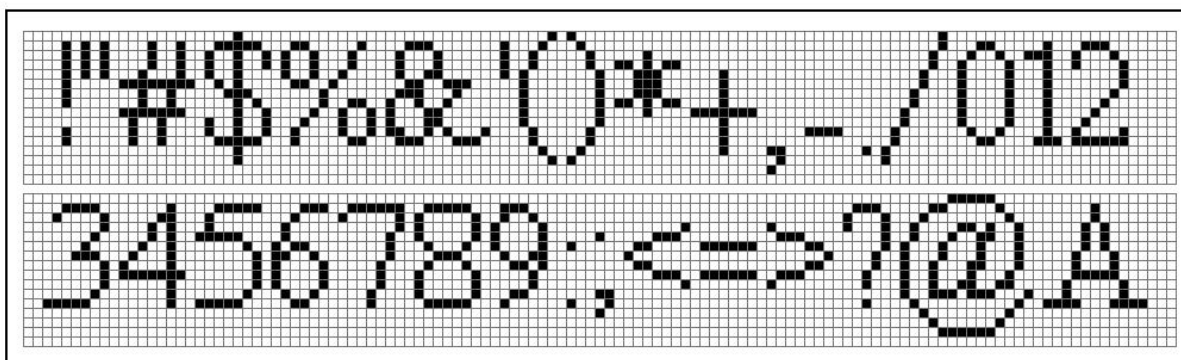
**8x16 点阵 ASCII 标准字符**

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	{		}	~	

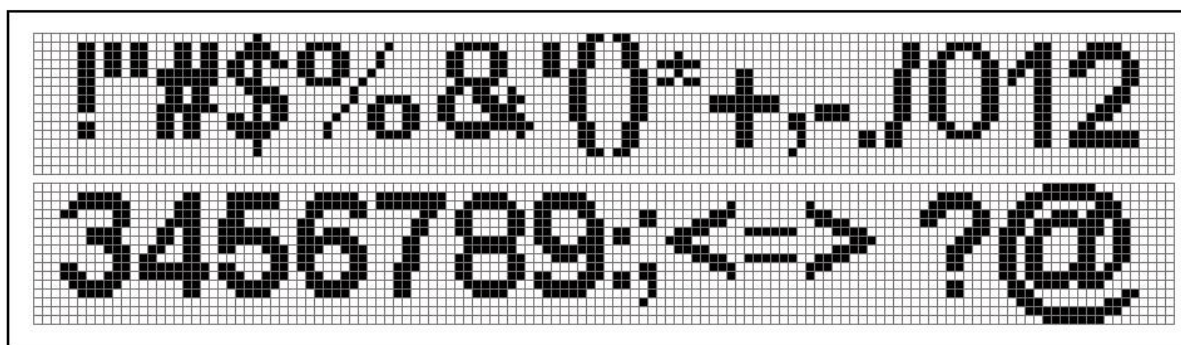
8x16 点阵 ASCII 粗体字符

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	{		}	~	

16 点阵不等宽 ASCII(线型字体)



16 点阵不等宽 ASCII (圆角字体)



48 点阵不等宽（圆角字体）

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	.	,	:	"	'	

48 点阵不等宽（线型字体）

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	.	,	:	"	'	

48 点阵不等宽（时钟体）

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	:					

48 点阵不等宽（方块体）

Low High	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	:					

## 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 $\infty$
FAST_READ	Read Data Bytes at Higher Speed	0000 1011	0B h	3	1	1 to $\infty$

所有对本芯片 SPI 接口的操作只有 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”)。

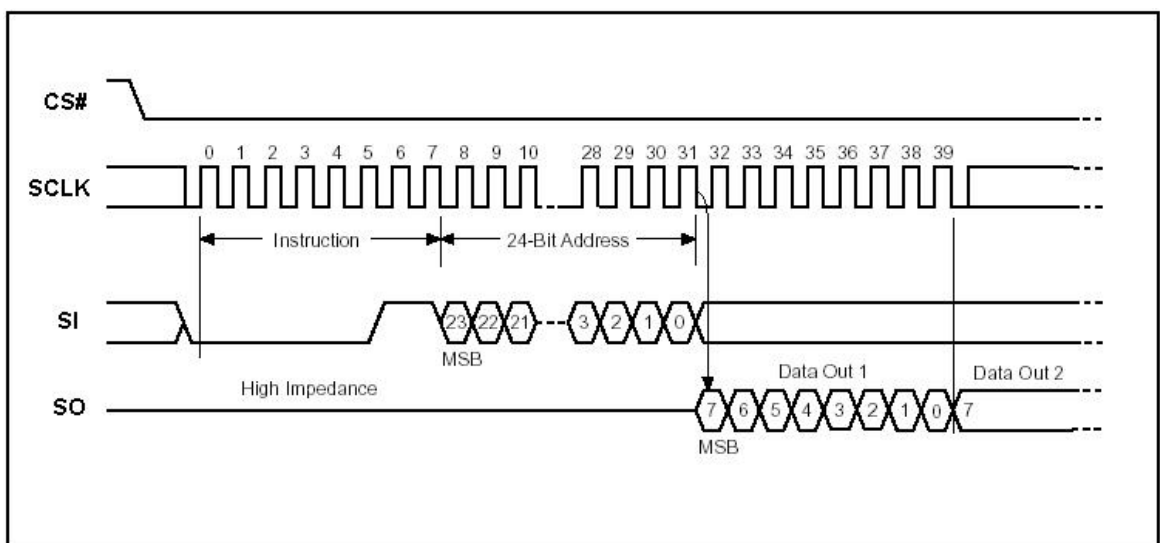
### 2.2 Read Data Bytes（一般读取）

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:



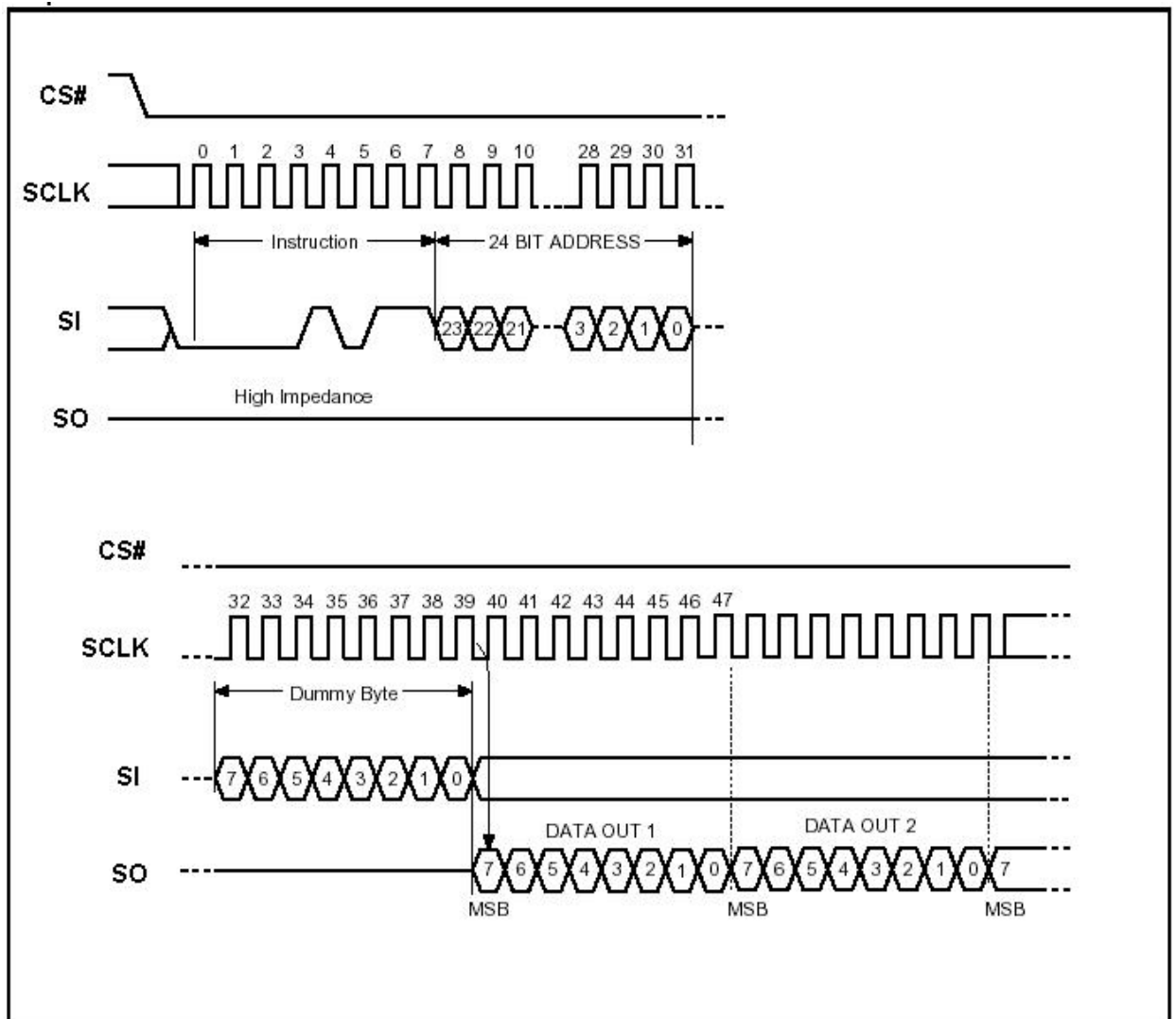
### 2.3 Read Data Bytes at Higher Speed (快速读取点阵数据)

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:

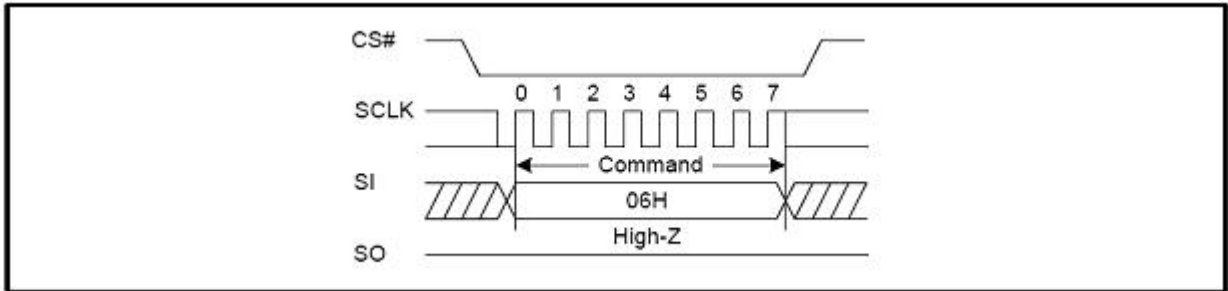




## 2.4 Write Enable (写使能)

Write Enable 指令的时序如下(图):

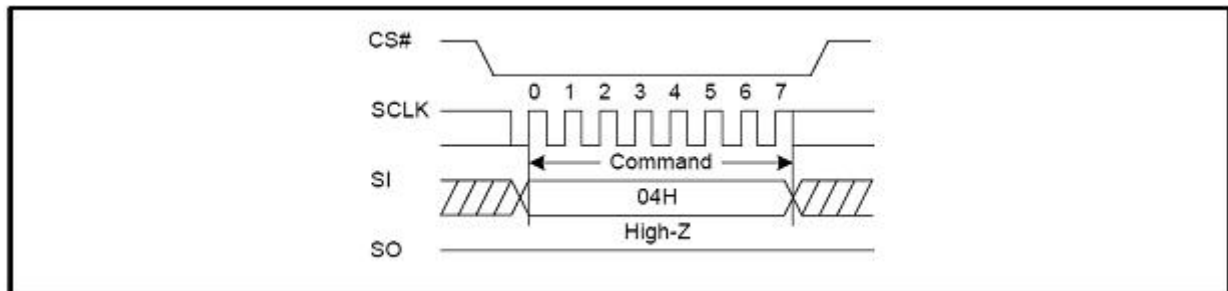
CS#变低→发送 Write Enable 命令→CS#变高



## 2.5 Write Disable (写非能)

Write Disable 指令的时序如下(图):

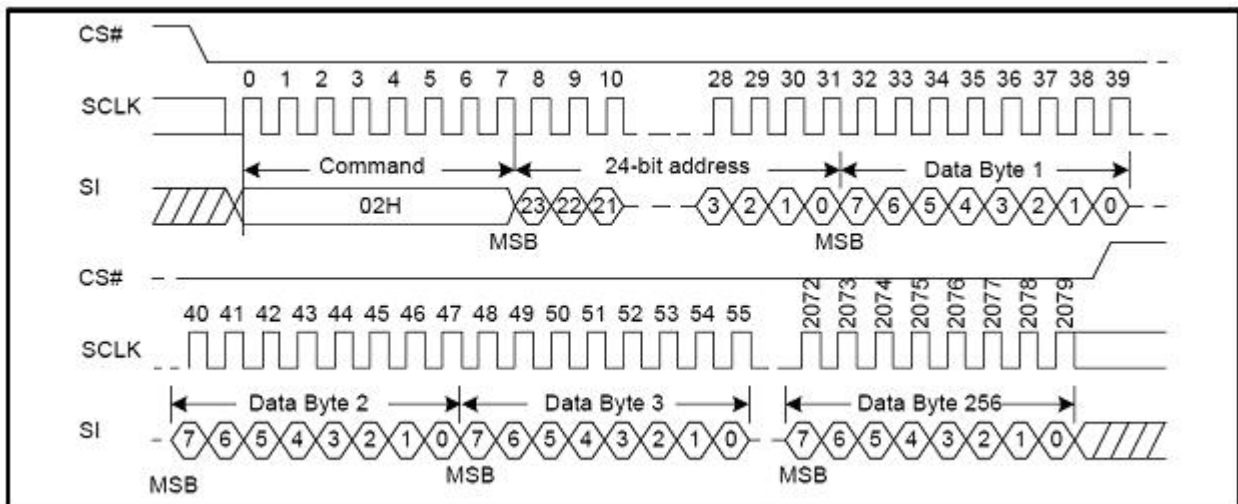
CS#变低→发送 Write Disable 命令→CS#变高



## 2.6 Page Program (页写入)

Page Program 指令的时序如下(图):

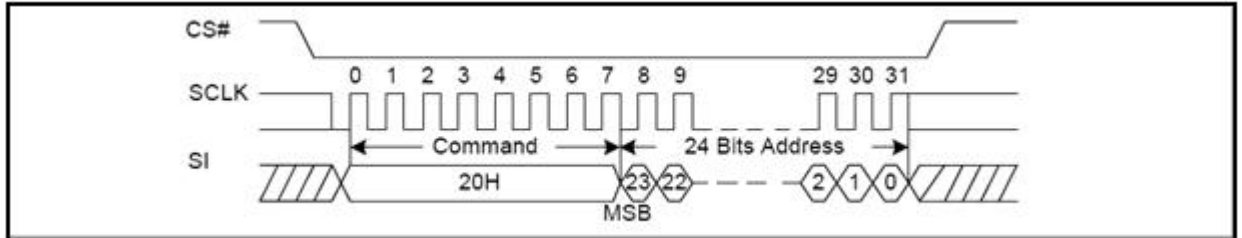
CS#变低→发送 Page Program 命令→发送 3 字节地址→发送数据→CS#变高



## 2.7 Sector Erase (扇区擦除)

Sector Erase 指令的时序如下(图):

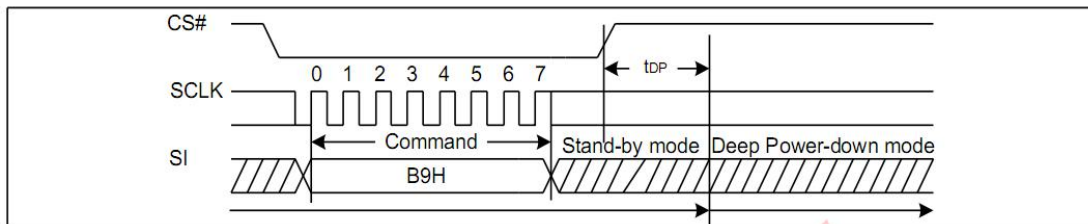
CS#变低→发送 Sector Erase 命令→发送 3 字节地址→CS#变高



## 2.8 深度睡眠模式指令 (B9H)

一旦字库芯片进入深度睡眠模式，所有的命令将被忽略，除了唤醒深度睡眠模式指令，首先 CS#为低电平，输入 B9H 命令，然后 CS#变为高电平并持续 TDP 的时间(TDP=25us)，在 TDP 的持续时间内，字库芯片进入深层关机模式。

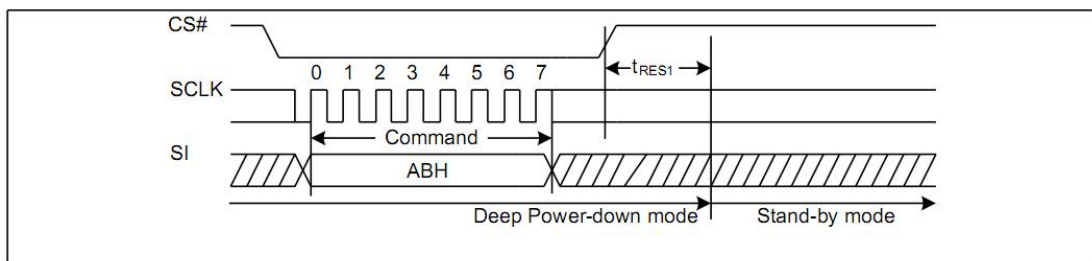
深度睡眠模式指令的时序波形图



## 2.9 唤醒深度睡眠模式指令 (ABH)

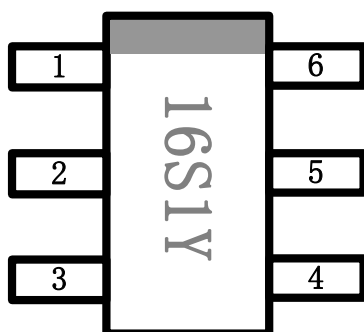
首先 CS#为低电平，向字库芯片发送 ABH 指令，然后 CS#变为高电平并持续 Tres1 的时间 (Tres1=25us)，字库芯片将恢复正常运行，CS#引脚必须在 Tres1 时间内保持高电平。

唤醒深度睡眠模式指令的时序波形图



### 3 引脚描述与电路连接

#### 3.1 引脚配置



#### 3.2 引脚描述

##### SOT23-6

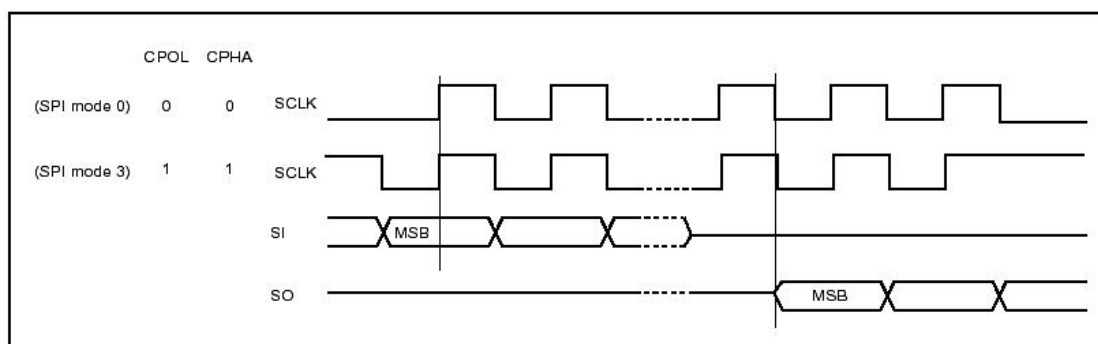
NO.	名称	I/O	描述
1	SCLK	I	串行时钟输入 (Serial clock input)
2	GND		地(Ground)
3	CS#	I	片选输入 (Chip enable input)
4	VCC		电源(+ 3.3V Power Supply)
5	SO	O	串行数据输出 (Serial data output)
6	SI	I	串行数据输入 (Serial data input)

**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。

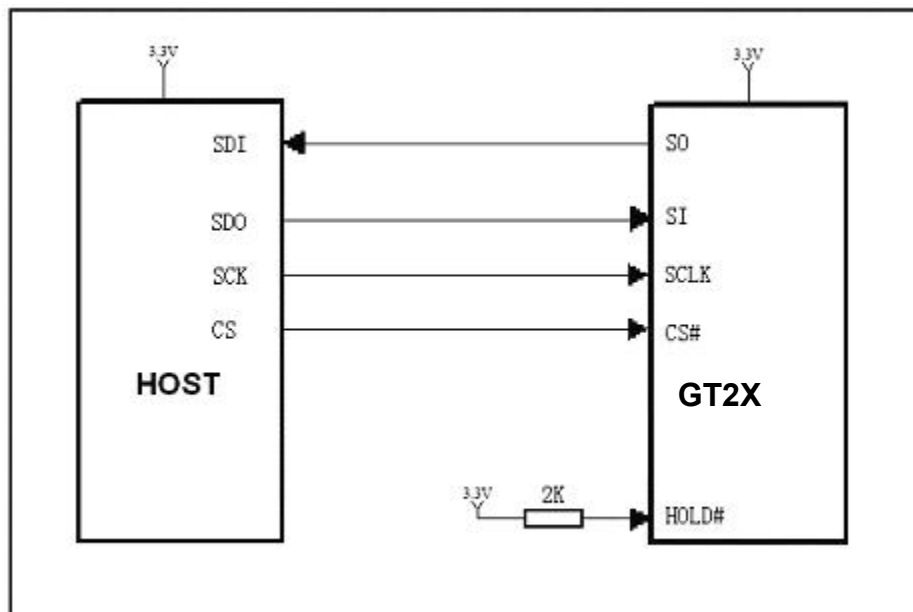
**串行时钟输入 (SCLK):** 数据在时钟上升沿移入，在下降沿移出。

**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。



### 3.3 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

## 4 电气特性

### 4.1 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	-55	125	°C	
V <sub>CC</sub>	Supply Voltage	-0.3	3.6	V	
V <sub>IN</sub>	Input Voltage	-0.3	V <sub>CC</sub> +0.3	V	
GND	Power Ground	-0.3	0.3	V	

### 4.2 DC 特性

Condition: T<sub>OP</sub> = -40°C to 85°C, GND=0V

Symbol	Parameter	Min.	Max.	Unit	Condition
I <sub>DD</sub>	V <sub>CC</sub> Supply Current(active)		8	mA	
I <sub>SB</sub>	V <sub>CC</sub> Standby Current		13	uA	
I <sub>DP</sub>	DeepPower-Down Current		2	uA	
V <sub>IL</sub>	Input LOW Voltage	-0.3	0.3V <sub>CC</sub>	V	V <sub>CC</sub> =2.7~3.6V
V <sub>IH</sub>	Input HIGH Voltage	0.7V <sub>CC</sub>	V <sub>CC</sub> +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	V <sub>CC</sub> -0.2 (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	

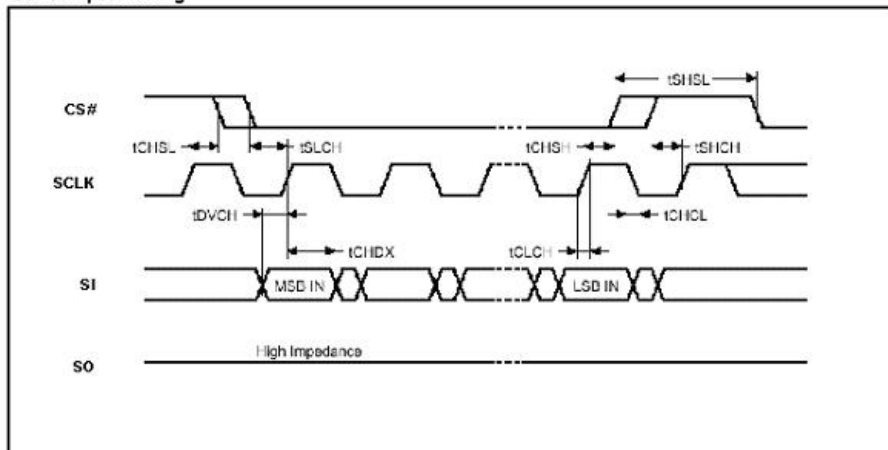
Note: I<sub>IL</sub>: Input LOW Current, I<sub>IH</sub>: Input HIGH Current,  
I<sub>OL</sub>: Output LOW Current, I<sub>OH</sub>: Output HIGH Current,

### 4.3 AC 特性

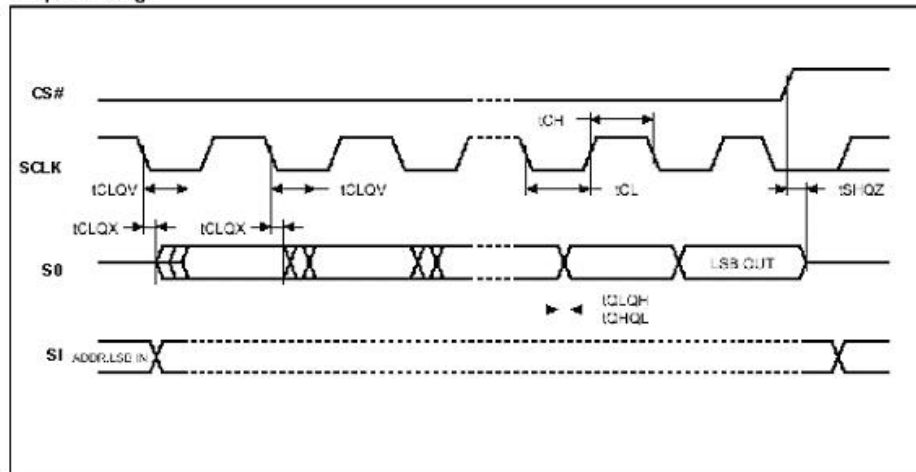
Symbol	Alt.	Parameter	Min.	Max.	Unit
F <sub>c</sub>	F <sub>c</sub>	Clock Frequency	D.C.	80	MHz
t <sub>CH</sub>	t <sub>CLH</sub>	Clock High Time	4		ns
t <sub>CL</sub>	t <sub>CLL</sub>	Clock Low Time	4		ns
t <sub>CLCH</sub>		Clock Rise Time(peak to peak)	0.2		V/ns
t <sub>CHCL</sub>		Clock Fall Time (peak to peak)	0.2		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	2		ns
t <sub>CHSH</sub>		CS# Active Hold Time (relative to SCLK)	5		ns

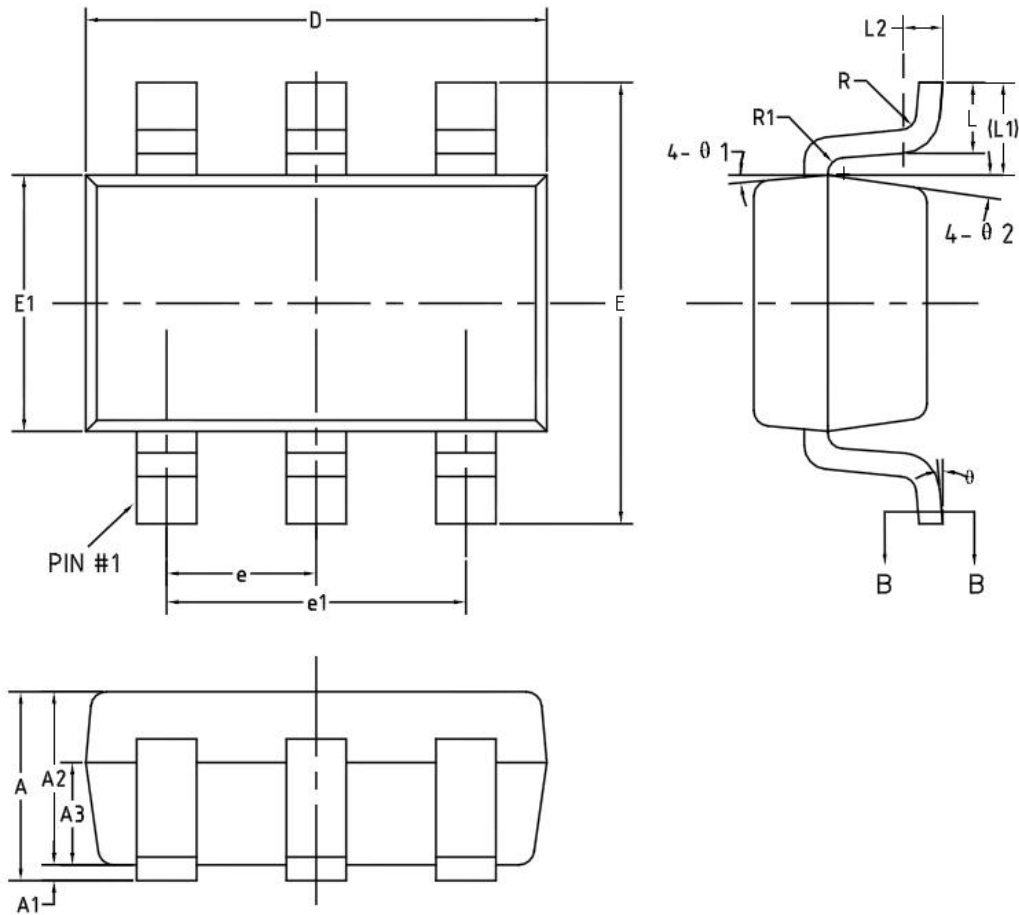
t SHCH		CS# Not Active Setup Time (relative to SCLK)	5		ns
t SHSL	tCSH	CS# Deselect Time	20		ns
t SHQZ	tDIS	Output Disable Time		6	ns
t CLQV	tV	Clock Low to Output Valid		6	ns
t CLQX	tHO	Output Hold Time	0		ns
t HLCH		HOLD# Setup Time (relative to SCLK)	5		ns
t CHHH		HOLD# Hold Time (relative to SCLK)	5		ns
t HHCH		HOLD Setup Time (relative to SCLK)	5		ns
t CHHL		HOLD Hold Time (relative to SCLK)	5		ns
t HHQX	tLZ	HOLD to Output Low-Z		9	ns
t HLQZ	tHZ	HOLD# to Output High-Z		9	ns

Serial Input Timing



Output Timing



**5 封装尺寸** SOT23-6 Package


SYMBOL	MIN	NOM	MAX
A	—	—	1.45
A1	0	—	0.15
A2	0.90	1.10	1.30
A3	0.60	0.65	0.70
b	0.39	—	0.49
b1	0.38	0.40	0.45
c	0.12	—	0.19
c1	0.11	0.13	0.15
D	2.85	2.95	3.05
E	2.60	2.80	3.00
E1	1.55	1.65	1.75
e	0.85	0.95	1.05
e1	1.80	1.90	2.00
L	0.35	0.45	0.60
L1	0.59REF		
L2	0.25BSC		
R	0.05	—	—
R1	0.05	—	0.20
θ	0°	—	8°
θ 1	8°	10°	12°
θ 2	8°	10°	12°

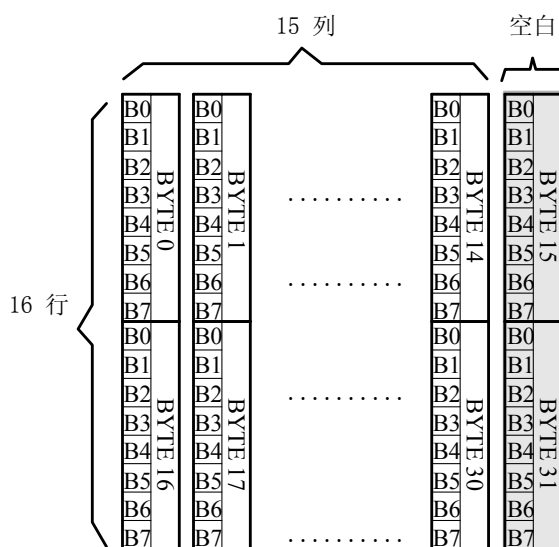
## 6 字库排置（竖置横排）

### 6.1 点阵排列格式

每个汉字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：即一个字节的高位表示下面的点，低位表示上面的点（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，则将出现对应的汉字。

### 6.2 15X16点汉字排列格式举例

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竖置横排的，其具体排列结构如下图：

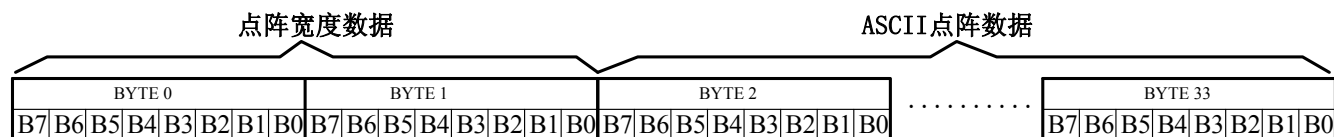


### 6.3 16点阵不等宽ASCII（圆角字体）字符排列格式

16点阵不等宽字符的信息需要34个字节（BYTE 0 – BYTE 33）来表示。

#### ■ 存储格式

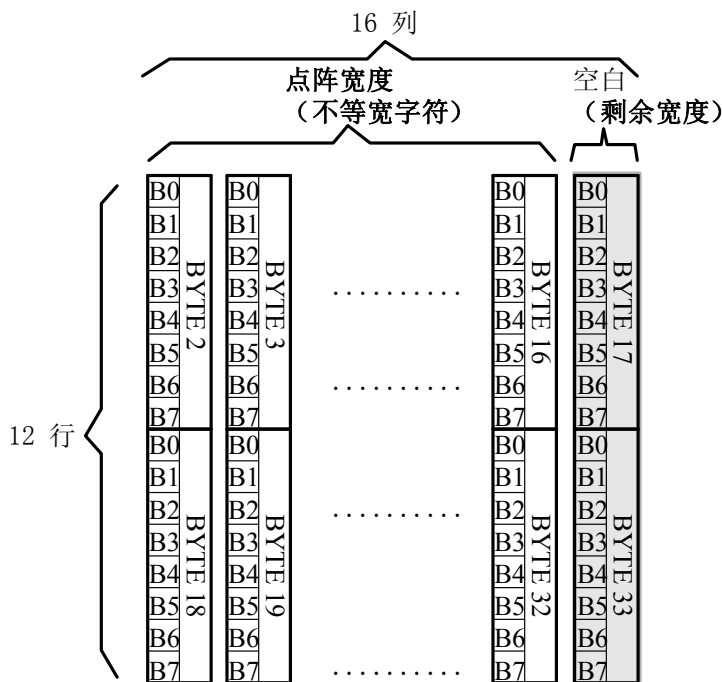
由于字符是不等宽的，因此在存储格式中BYTE 0~ BYTE 1存放点阵宽度数据，BYTE 2-33存放竖置横排点阵数据。具体格式见下图：





■ 存储结构

点阵存储宽度固定为 16，根据不同字符，其实际点阵宽度会小于 16，并会出现相应的空白区。根据 BYTE0~ BYTE1 所存放点阵的宽度数据，可以对还原下一个字的显示或排版留作参考。



例如：ASCII 圆角字符 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 7F 7F  
63 63 63 63 63 67 3E 1C 00 00 00 00 00

其中：

BYTE0~ BYTE1: 00 0C 为 ASCII 圆角字符 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。（见下图）

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 63 67 3E 1C  
00 00 00 00 00 为 ASCII 圆角字符 B 的点阵数据。

## 7 点阵数据验证（客户参考用）

客户将芯片内“A”的数据调出与以下进行对比。若一致，表示 SPI 驱动正常工作；若不一致，请重新编写驱动。

排置：Y（竖置横排）点阵大小 8X16

字母“A”

点阵数据：00 E0 9C 82 9C E0 00 00 0F 00 00 00 00 0F 00

排置：W（横置横排）点阵大小 8X16

字母“A”

点阵数据：00 10 28 28 28 44 44 7C 82 82 82 82 00 00 00 00

## 8 附录

### 8.1 GB2312 1 区 (376 字符)

GB2312 标准点阵字符 1 区对应码位的 A1A1~A9EF 共计 376 个字符;

GB2312 1 区

A1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A			、	。	·	-	√	”	々	一	~		…	‘	’	
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																
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	{		}	—	

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

## 8.2 8x16 点国际扩展字符（126 字符）

内码组成为 AAA1~ABC0 共计 126 个字符

AA	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	{   }	~			

AB	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	ǐ	ì	ō	ó	ǒ
B	ò	ū	ú	ǔ	ù	ǖ	ú	ǘ	ù	ü	ê	à	á	ñ	ñ	ñ
C	g															