

高通<sup>®</sup>字库  
GENTOP<sup>®</sup>

# GT31L16M1Y80 标准点阵汉字库芯片

— 产品规格书 —

V1.01\_L  
2024-08



[www.gaotongfont.cn](http://www.gaotongfont.cn)

## 版本修订记录

版本号	修改内容	日期	备注
V 1.0I_A	字库芯片说明书的制定	2012-08	字库定制
V1.0I_B	增加状态寄存器的说明	2012-03	
V1.0I_C	增加自由读写空间对应烧录器中的型号	2013-06	
V1.0I_D	字库芯片说明书格式修改	2015-05	
V1.0I_E	更新字库 AC/DC 参数	2017-03	
V1.0I_F	添加上电时序、删除 sop8 系列封装	2019-06	
V1.0I_G	时钟频率由 50MHZ 更新为 45MHZ	2019-07	
V1.0I_H	更新规格书样张 、 验证数据	2020-03	
V1.0I_I	单线更新为四线，更新引脚配置及增加四线指令、更新电气特性	2020-12	
V1.0I_J	更新地址、网址、电气特性	2022-06	
V1.0I_K	更新企业网站网址	2023-04	
V1.0I_L	更新概述	2024-08	

# 目 录

<b>1 概述</b> .....	<b>5</b>
1.1 芯片特点.....	5
1.2 芯片内容.....	6
1.3 字型样张.....	7
<b>2 操作指令</b> .....	<b>8</b>
2.1 Instruction Parameter(指令参数).....	8
2.2 Read Data Bytes (一般读取) .....	8
2.3 Read Data Bytes at Higher Speed (快速读取点阵数据) .....	9
2.4 四线 I/O 快速读取 (EBH) .....	10
2.5 四线 I/O 字快速读 (E7H) .....	11
2.6 四线页写入指令 (32H) .....	12
2.8 Write Enable (写使能) .....	13
2.9 Write Disable (写非能) .....	13
2.10Page Program (页写入) .....	13
2.11Sector Erase (扇区擦除) .....	14
2.12 深度睡眠模式指令 (B9H) .....	14
2.13 唤醒深度睡眠模式指令 (ABH) .....	14
2.14 读芯片状态时序.....	15
2.15 芯片状态寄存器以及说明.....	15
2.16 读取芯片状态寄存器的命令说明.....	15
<b>3 自由可读写空间描述</b> .....	<b>16</b>
3.1 存储组织.....	16
3.2 存储块、扇区结构.....	16
<b>4 引脚描述与电路连接</b> .....	<b>17</b>
4.1 引脚配置.....	17
4.2 引脚描述.....	17
4.3SPI 接口与主机接口参考电路示意图.....	19
<b>5 电气特性</b> .....	<b>20</b>
5.1 绝对最大额定值.....	20
5.2 DC 特性.....	20

5.3 AC 特性 .....	20
5.4 上电时序 .....	22
<b>6 封装尺寸 .....</b>	<b>23</b>
<b>7 字库排置（竖置横排） .....</b>	<b>25</b>
7.1 点阵排列格式 .....	25
7.2 15X16 点汉字排列格式举例 .....	25
7.3 16 点阵不等宽 ASCII 圆角字体字符排列格式 .....	25
<b>8 点阵数据验证（客户参考用） .....</b>	<b>27</b>



# 1 概述

GT31L16M1Y80是一款16x16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，利用我司所提供库文件内的函数接口可直接读取该内码的点阵信息。

GT31L16M1Y80除含有上述字库以外，还提供1024KB字节的可自由读写空间，包括256个扇区，每个扇区4K字节或16页，每页256字节，可自由读写空间地址范围为：0x000000~0xfffff，可重复擦写10万次以上。

## 重要提示：

高通字库芯片开发资料及库文件获取步骤

- 1: 下载 MindCraft 软件并注册
- 2: 打开应用->自动库 Lib 助手
- 3: 根据界面提示选择相应的配置或者参数
- 4: 点击生成文件，自动生成开发资料及 API 函数库

高通字库芯片开发资料及库文件获取

<https://www.hmi.gaotongfont.cn/gtzkxpkfzl>

高通字库在 32 位 MCU 上的使用视频教程

[https://www.bilibili.com/video/BV1aG41117uH/?spm\\_id\\_from=333.999.0.0](https://www.bilibili.com/video/BV1aG41117uH/?spm_id_from=333.999.0.0)

## 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：45MHz (max.) @3.3V
- 工作电压：2.7V~3.6V
- 电流：
  - 工作电流：读电流 20mA(max)
  - 写电流 30mA(max)
  - 睡眠电流：8uA(Max)
- 工作温度：-40°C~85°C
- 封装：DFN2X3/SOP8-B
- 字符集：
  - 中文 GB18030
  - 兼容 UNICODE
- 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	7x10	96	打印机字体	Y-竖置横排
	ASCII	7x12	96	打印机字体	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	16x32	96	标准	Y-竖置横排
	ASCII	16x32	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
GB18030 字符集	GB18030 汉字	16x16	27533+1038	宋体	Y-竖置横排
Unicode -> GBK 转码表					
BIG5-> GBK 转码表					

## 1.3 字型样张

### 1.3.1 汉字字符

点阵	字体	标准	中文字体样张
16 点	宋体	国标	高通字库,绽放文字之美

### 1.3.2 ASCII 点阵字符

点阵大小	字体	字符集	ASCII 字体样张
5X7	标准	ASCII	AaBbCcDdEe12345
7X8	粗体		<b>AaBbCcDdEe12345</b>
7X10	打印		AaBbCcDdEe12345
7X12	打印		AaBbCcDdEe12345
8X16	标准		AaBbCcDdEe12345
8X16	粗体		<b>AaBbCcDdEe12345</b>
16 点	圆角		<b>AaBbCcDdEe12345</b>
16X32	宋体		AaBbCcDdEe12345
16X32	粗体		<b>AaBbCcDdEe12345</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 ∞
Fast Read	Read Data Bytes at Higher Speed	0000 1011	0B h	3	1
WREN	Write Enable	0000 0110	06 h	—	—
WRDI	Write Disable	0000 0100	04 h	—	—
PP	Page Program	0000 0010	02 h	3	1 to 256
SE	Sector Erase	0010 0000	20 h	3	—
BE	Block Erase(64K)	1101 1000	D8 h	3	—
CE	Chip Erase	0110 0000/ 1100 0111	60 H/ C7 H	—	—

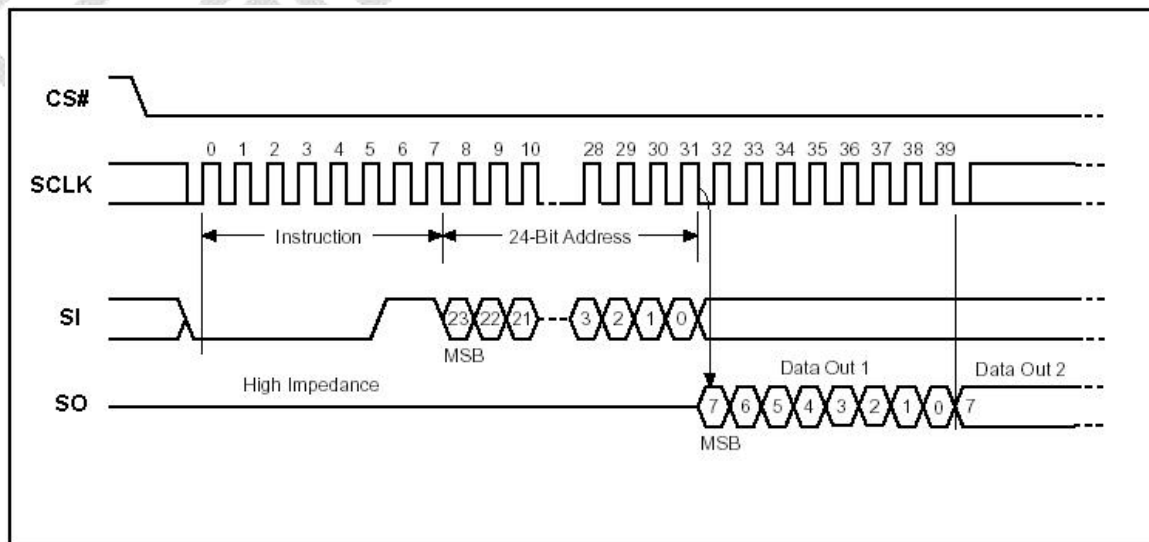
### 2.2 Read Data Bytes (一般读取)

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:



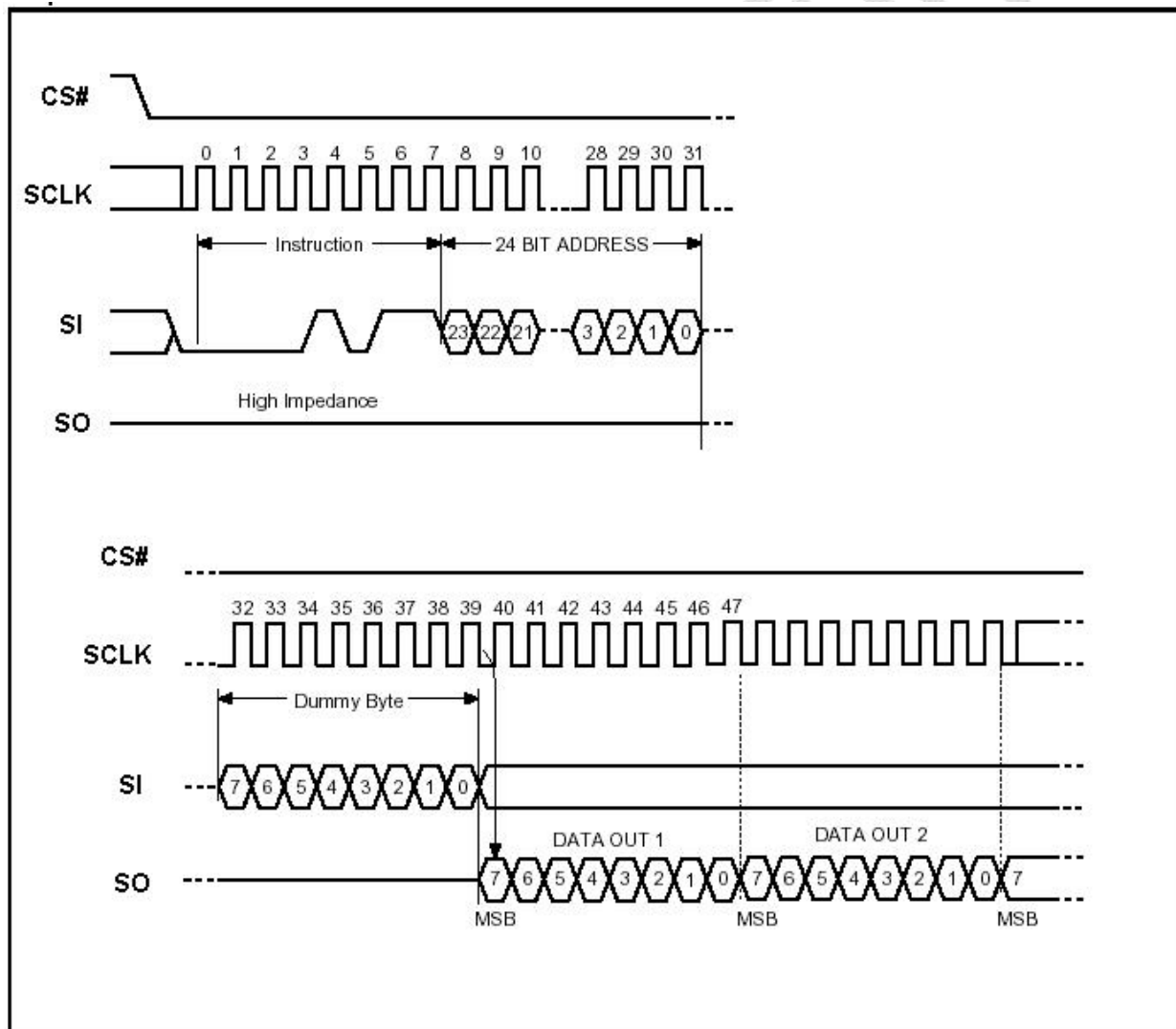


### 2.3 Read Data Bytes at Higher Speed (快速读取点阵数据)

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:

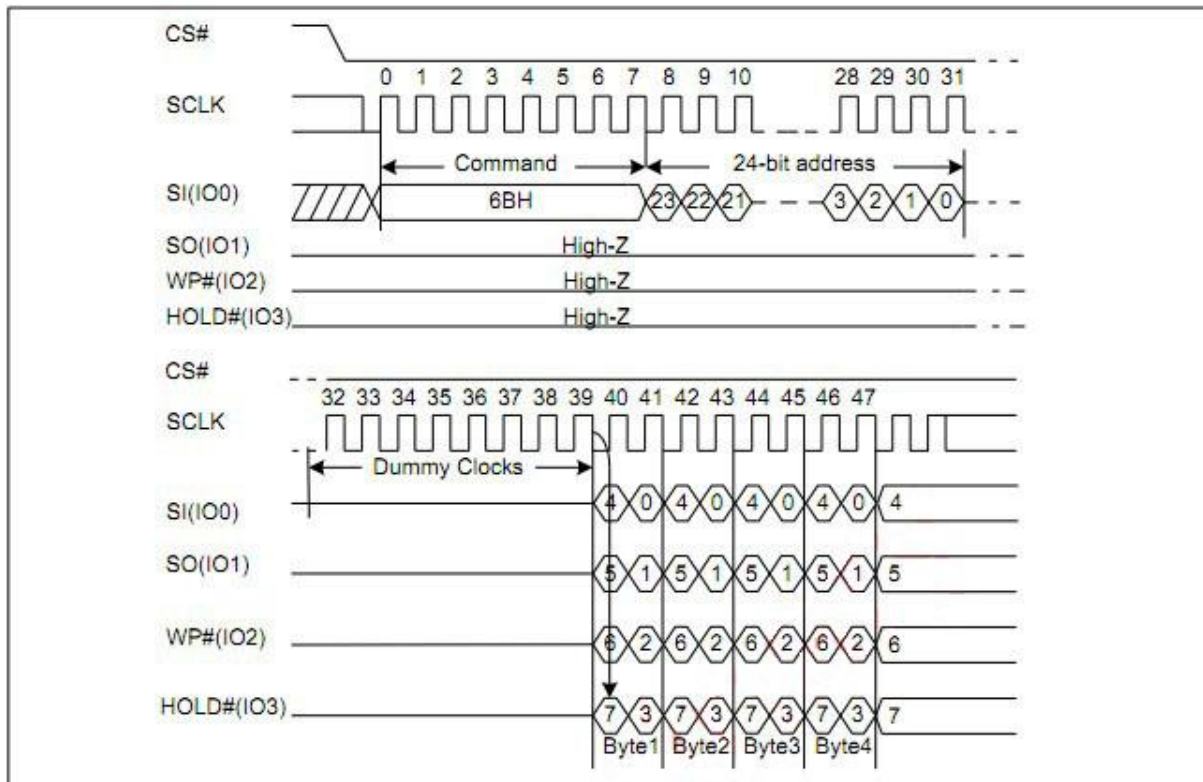


#### 四线输出快速读取 (6BH)

四线输出快速读取命令后面是 3 字节地址 (A23-A0) 和一个空字节, 每个位在 SCLK 的上升沿锁存, 然后存储器内容从 IO3, IO2, IO1, IO0 每个时钟周期移出 4 位。命令序列如图所示。所寻址的第

一字节可以在任何位置。地址在每个字节的数据被移出后自动递增到下一个更高的地址。

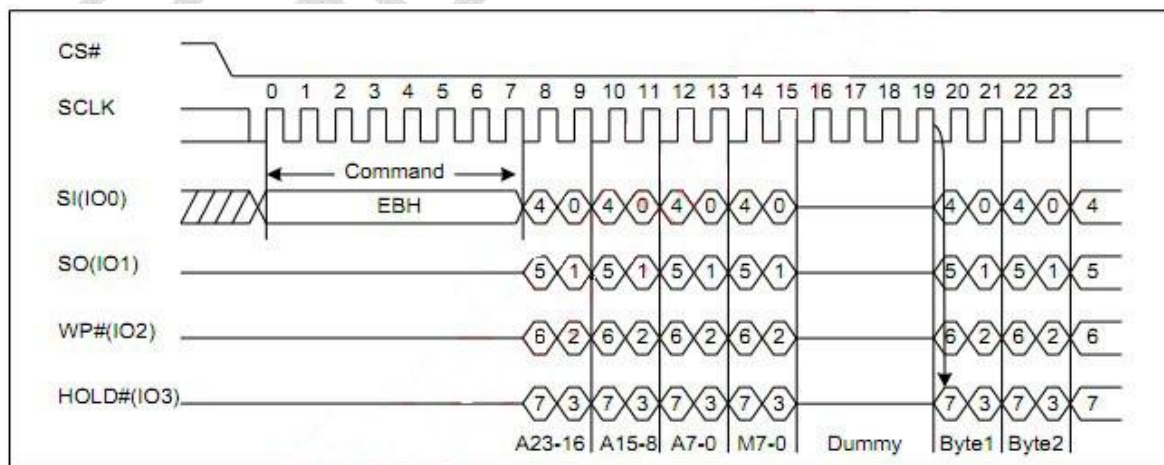
四线输出快速读取序列图



### 2.4 四线 I/O 快速读取 (EBH)

四线 I/O 快速读取命令，输入 3 个字节地址 (A23-0) 和“连续读取模式”字节和 4 个空循环时钟，每个时钟 4 位每个 IO3, IO2, IO1, IO0 位在 SCLK 的上升沿期间被锁存，则存储器内容从 IO3, IO2, IO1, IO0 每个时钟周期移出 4 位，命令序列如图所示。所寻址的第一字节可以在任何位置。每个字节的数据移出后，地址自动递增到下一个更高的地址。四线使能位必须将状态寄存器 (S9) 的状态 (QE) 设置为使能。

四线 I/O 快速读取序列图 (M7-0 = 0XH 或非 AXH)

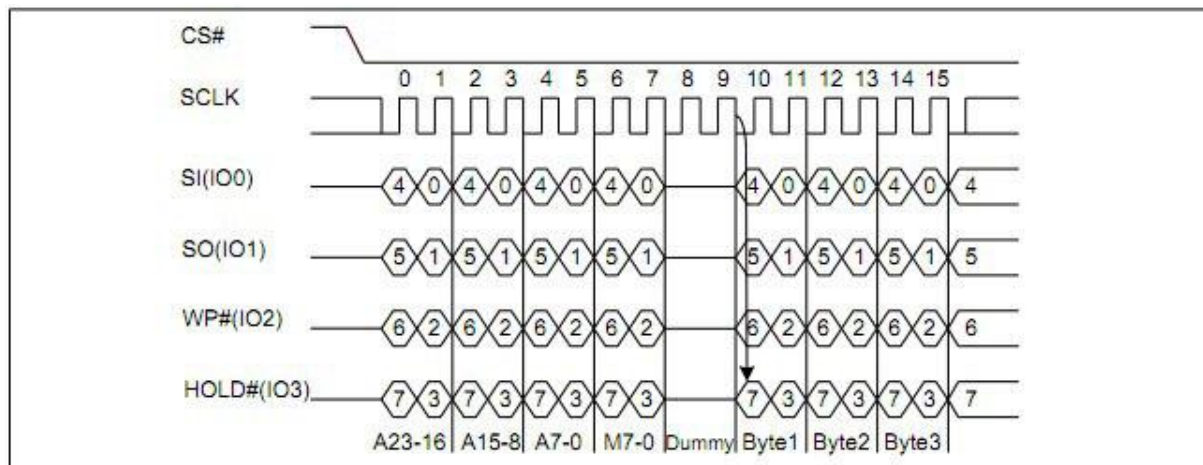


四线 I/O 快速读取序列图 (M7-0 = AXH)





四线 I / O 字快速读取序列图 (M7-0 = AXH)



## 2.6 四线页写入指令 (32H)

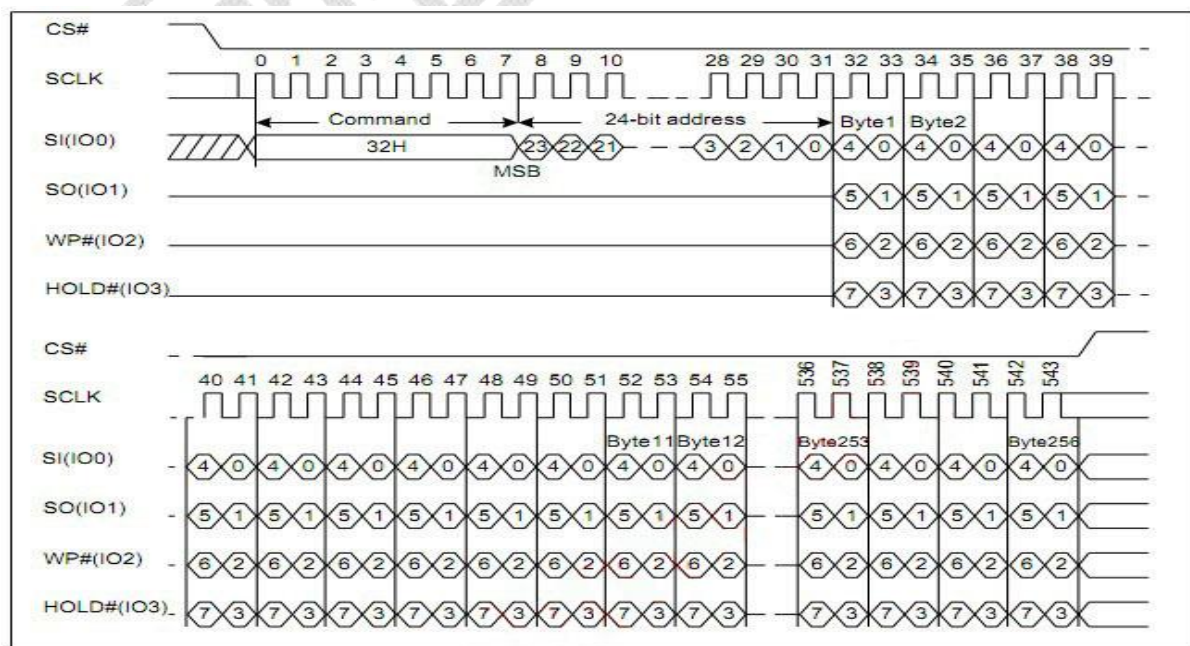
四线页写入命令用于使用四个引脚 (IO0, IO1, IO2 和 IO3) 对存储器进行编程。要使用四线页写入功能, 必须设置状态寄存器 Bit9 中的四态使能 (QE = 1)。在发送页编程命令之前, 必须先执行写使能 (WREN) 命令来设置写使能锁存 (WEL) 位。通过驱动 CS# 低电平, 然后是命令代码 (32H), 三个地址字节和 IO 引脚上至少一个数据字节, 输入四线页写入命令。

命令序列如图所示。如果超过 256 字节发送到器件, 先前锁存的数据被丢弃, 最后 256 个数据字节被保证在同一页内被正确编程。如果小于 256 个数据字节发送到设备, 它们在请求的地址被正确编程, 而不会对同一页的其他字节产生任何影响。在最后一个数据字节的第 8 位被锁存后, CS# 必须被驱动为高电平; 否则不执行四线页写入 (PP) 命令。

一旦 CS# 驱动为高电平, 自定时四页面编程周期 (其持续时间为  $t_{pp}$ ) 被启动。当四页面编程周期正在进行时, 可以读取状态寄存器以检查写入进行 (WIP) 位的值。在自定时四页面编程周期中, 写入进行 (WIP) 位为 1, 完成后为 0。在周期完成前的某个未指定时间, 写使能锁存器 (WEL) 位复位。

一个不执行应用于由块保护 (BP4, BP3, BP2, BP1 和 BP0) 保护的页的四页编程命令。

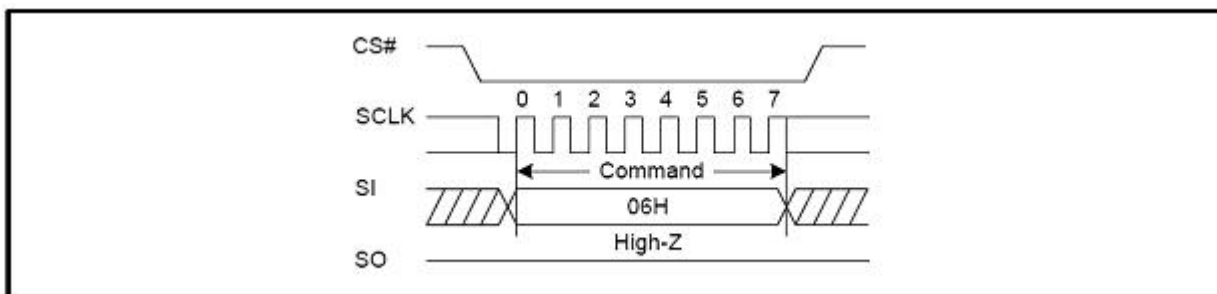
Quad 页面程序序列图



## 2.8 Write Enable (写使能)

Write Enable 指令的时序如下(图):

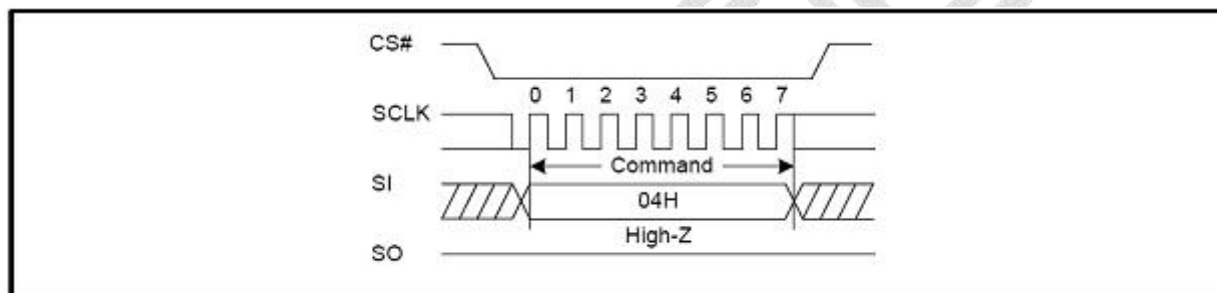
CS#变低->发送 Write Enable 命令->CS#变高



## 2.9 Write Disable (写非能)

Write Disable 指令的时序如下(图):

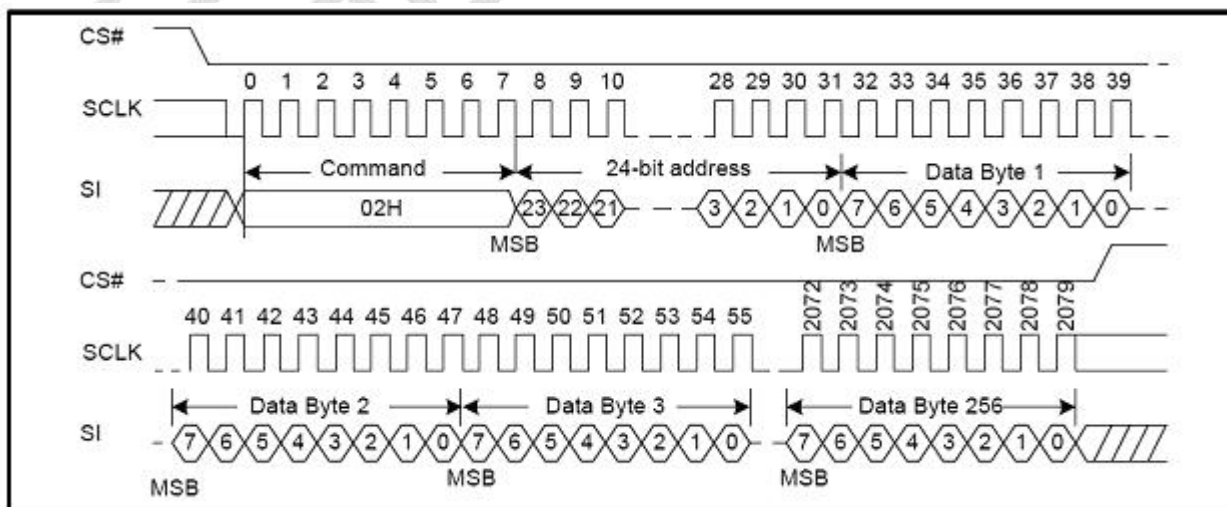
CS#变低->发送 Write Disable 命令->CS#变高



## 2.10 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

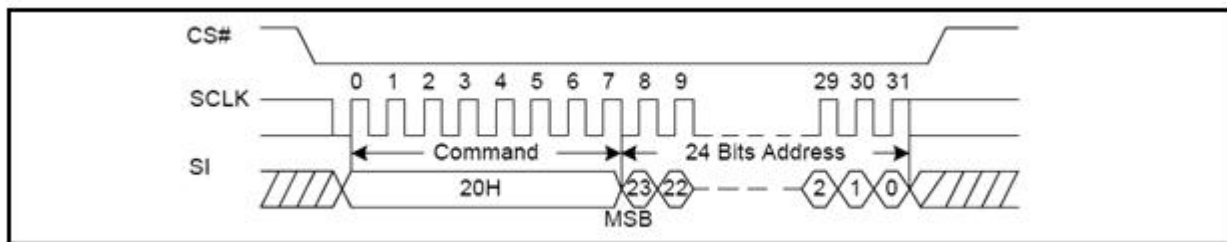


注：写入指令发送 CS#变高后需进行忙状态判断，等待芯片内部完成写入后，才可以对芯片进行下一步操作，判断忙状态请参考该型号相应的库文件，如无库文件请与我司索要。

## 2.11 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

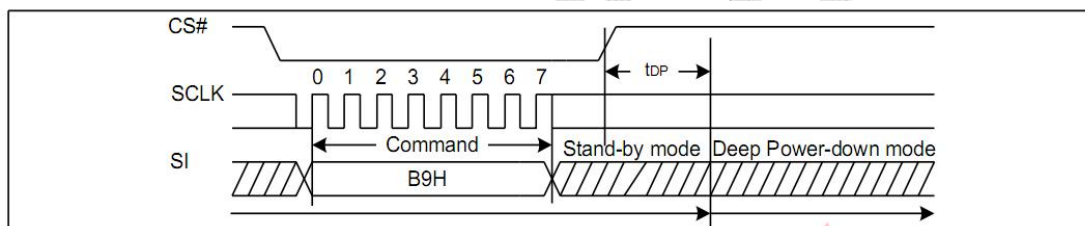


注：擦除指令发送 CS#变高后需进行忙状态判断，等待芯片内部完成擦除后，才可以对芯片进行下一步操作，判断忙状态请参考该型号相应的库文件，如无库文件请与我司索要。

## 2.12 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，首先首先 CS#为低电平，输入 B9H 命令，然后然后 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。

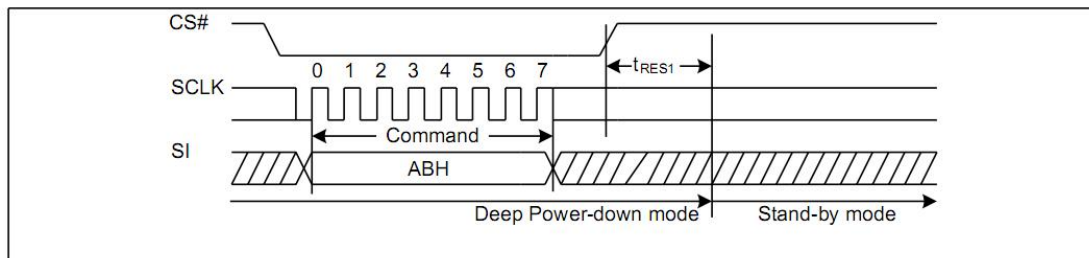
深度睡眠模式指令的时序波形图



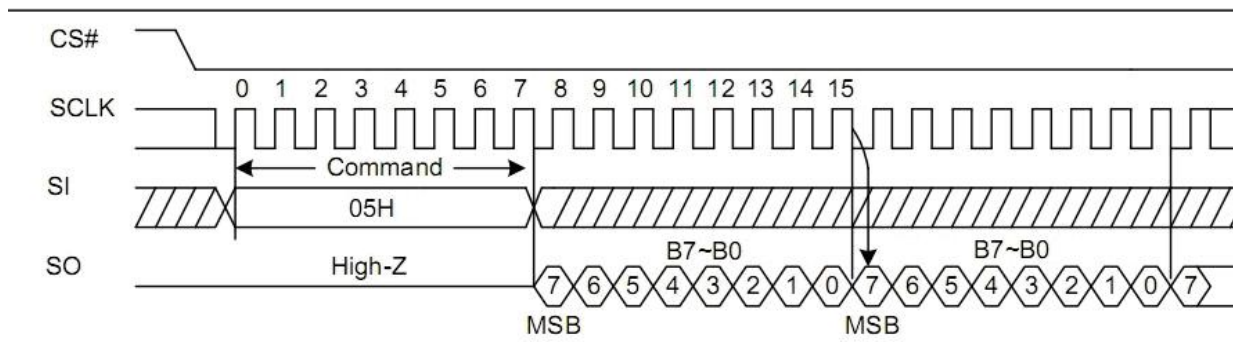
## 2.13 唤醒深度睡眠模式指令 (ABH)

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



## 2.14 读芯片状态时序



## 2.15 芯片状态寄存器以及说明

### Status Register

B7	B6	B5	B4	B3	B2	B1	B0
BP0	SP4	SP3	SP2	SP1	SP0	WSL	WIP

判断芯片是否在忙状态，使用寄存器 B0,当 B0 位的 WIP 位为 1 的时候，为忙状态，当 WIP 位为 0 的时候芯片处于空闲状态。

## 2.16 读取芯片状态寄存器的命令说明

发送命令 05H，然后读取芯片状态寄存器的 B7-B0 位。判断 WIP 位的状态来判断芯片是否在忙状态。

### 3 自由可读写空间描述

#### 3.1 存储组织

每设备	每块	每扇区	每页	
1M	64K	4K	256	字节
4K	256	16		页
256	16			扇区
16				块

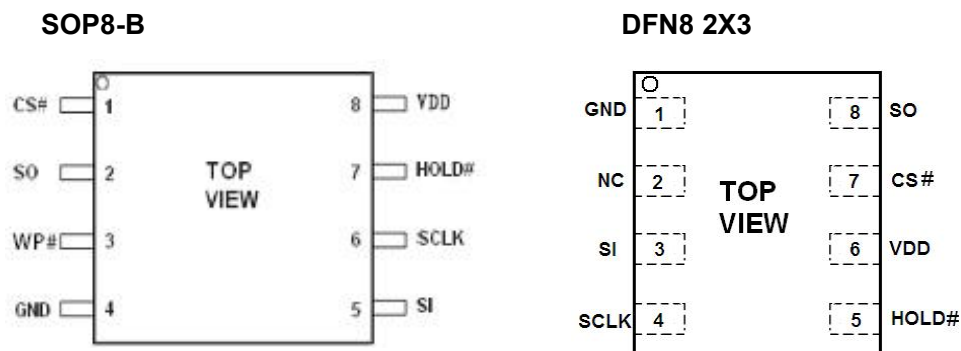
#### 3.2 存储块、扇区结构

块	扇区	地址范围	
15	255	0x0FF000	0x0FFFFFF
	.....	.....	.....
	240	0x0F0000	0x0F0FFF
14	239	0x0EF000	0x0EFFFF
	.....	.....	.....
	224	0x0E0000	0x0E0FFF
.....	.....	.....	.....
	.....	.....	.....
	.....	.....	.....
.....	.....	.....	.....
	.....	.....	.....
	.....	.....	.....
.....	.....	.....	.....
	.....	.....	.....
	.....	.....	.....
2	47	0x02F000	0x02FFFF
	.....	.....	.....
	32	0x020000	0x020FFF
1	31	0x01F000	0x01FFFF
	.....	.....	.....
	16	0x010000	0x010FFF
0	15	0x00F000	0x00FFFF
	.....	.....	.....
	0	0x000000	0x000FFF



## 4 引脚描述与电路连接

### 4.1 引脚配置



### 4.2 引脚描述

#### SOP8-B

NO.	名称	I/O	描述
1	CS#	I	片选输入 (Chip enable input)
2	SO	O	1. 串串行数据输出 (Serial data output) 2. 数据输入输出脚1 (Data input output1)
3	WP#		1. 串串行数据输出 (Serial data output) 2. 数据输入输出脚2 (Data input output2)
4	GND		地(Ground)
5	SI	I	1. 串行数据输入 (Serial data input) 2. 数据输入输出脚0 (Data input output0)
6	SCLK	I	串行时钟输入 (Serial clock input)
7	HOLD#	I	1. 总线挂起 (Hold, to pause the device without) 2. 数据输入输出脚3 (Data input output3)
8	VDD		电源(+ 3.3V Power Supply)

#### DFN8 2X3

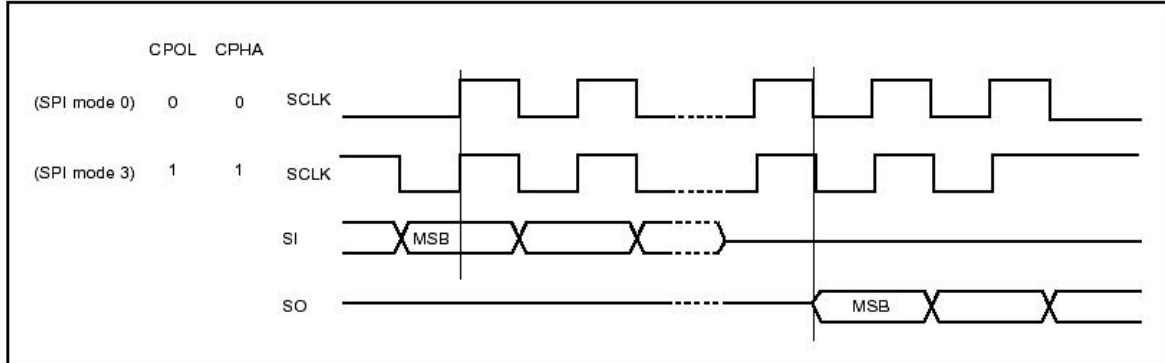
NO.	名称	I/O	描述
1	GND		地(Ground)
2	NC		悬空
3	SI	I	串行数据输入 (Serial data input)
4	SCLK	I	串行时钟输入 (Serial clock input)
5	HOLD#	I	总线挂起 (Hold, to pause the device without)
6	VDD		电源(+ 3.3V Power Supply)
7	CS#	I	片选输入 (Chip enable input)
8	SO	O	串行数据输出 (Serial data output)

**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。

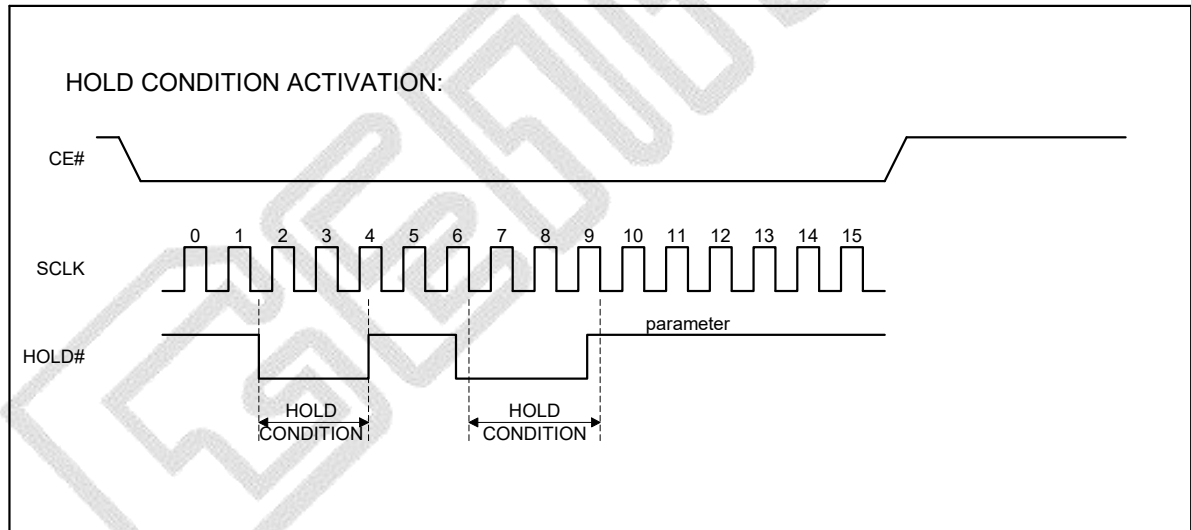


**总线挂起输入 (HOLD#):**

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。

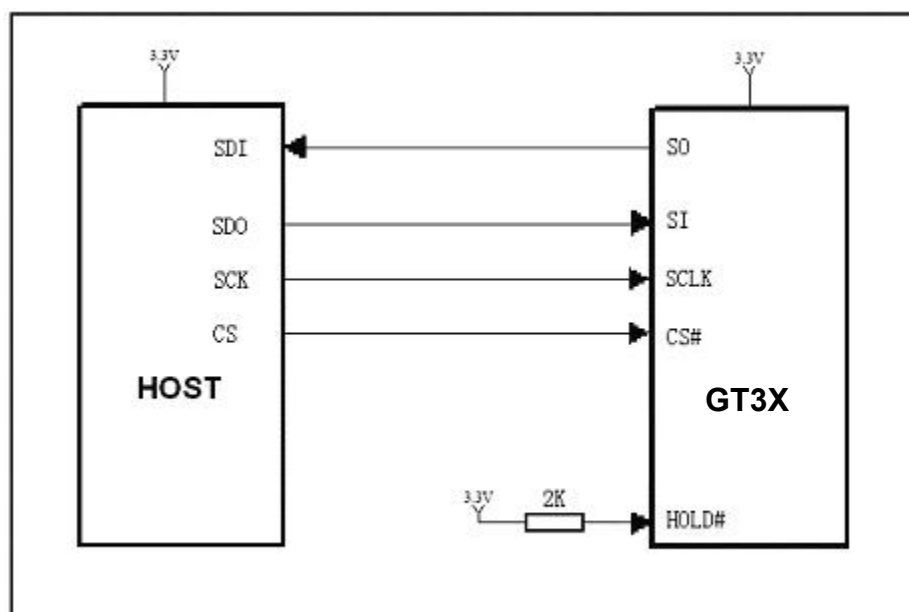
当HOLD#信号变为低并且串行时钟信号 (SCLK) 处于低电平时, 进入总线挂起状态。

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



### 4.3 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

## 5 电气特性

### 5.1 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	
V <sub>DD</sub>	Supply Voltage	-0.3	3.6	V	
V <sub>IN</sub>	Input Voltage	-0.3	V <sub>DD</sub> +0.3	V	

### 5.2 DC 特性

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

Symbol	Parameter	Min.	Typ.	Max.	Unit	Condition
I <sub>DD</sub>	VDD Supply Current(active)	-	-	30	mA	
I <sub>SB</sub>	VDD Standby Current	-	-	40	uA	/CS=VDD, VIN=VDD orVSS
I <sub>CC2</sub>	Deep Power-Down Current	-	-	8	uA	/CS=VDD, VIN=VDD or VSS
V <sub>IL</sub>	Input LOW Voltage	-0.5	-	0.16VCC	V	VDD=2.7~3.6 V
V <sub>IH</sub>	Input HIGH Voltage	0.8VCC	-	VCC+0.4	V	
V <sub>OL</sub>	Output LOW Voltage		-	0.2 (I <sub>OL</sub> =1.6mA)	V	
V <sub>OH</sub>	Output HIGH Voltage	VDD-0.2 (I <sub>OH</sub> =-100uA)	-		V	
I <sub>LI</sub>	Input Leakage Current	-	-	±2	uA	
I <sub>LO</sub>	Output Leakage Current	-	-	±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,

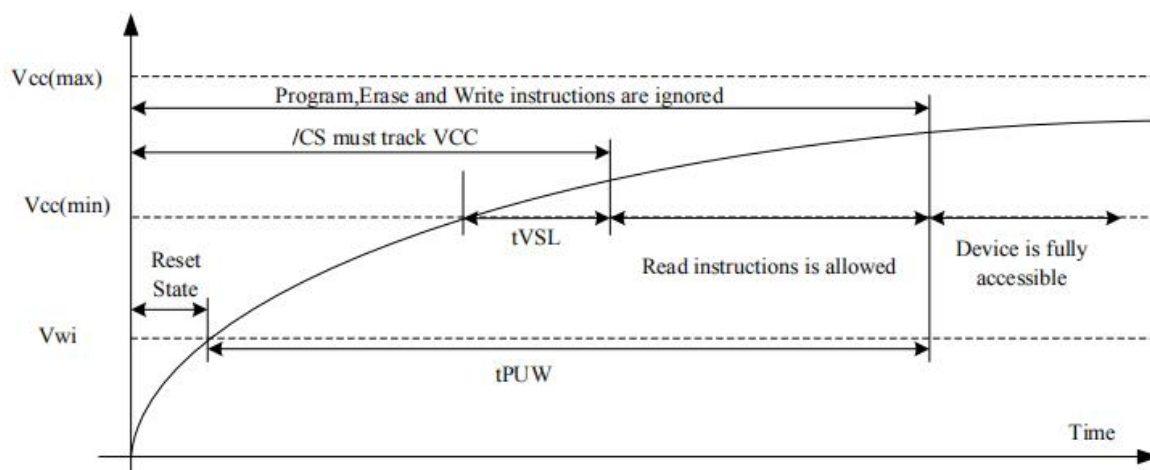
### 5.3 AC 特性

Symbol	Alt.	Parameter	Min.	Typ.	Max.	Unit
F <sub>c</sub>	F <sub>c</sub>	Clock Frequency	D.C.		45	MHz
t <sub>CH</sub>	t <sub>CLH</sub>	Clock High Time	4.5			ns
t <sub>CL</sub>	t <sub>CLL</sub>	Clock Low Time	4.5			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)	7			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



### 5.4 上电时序

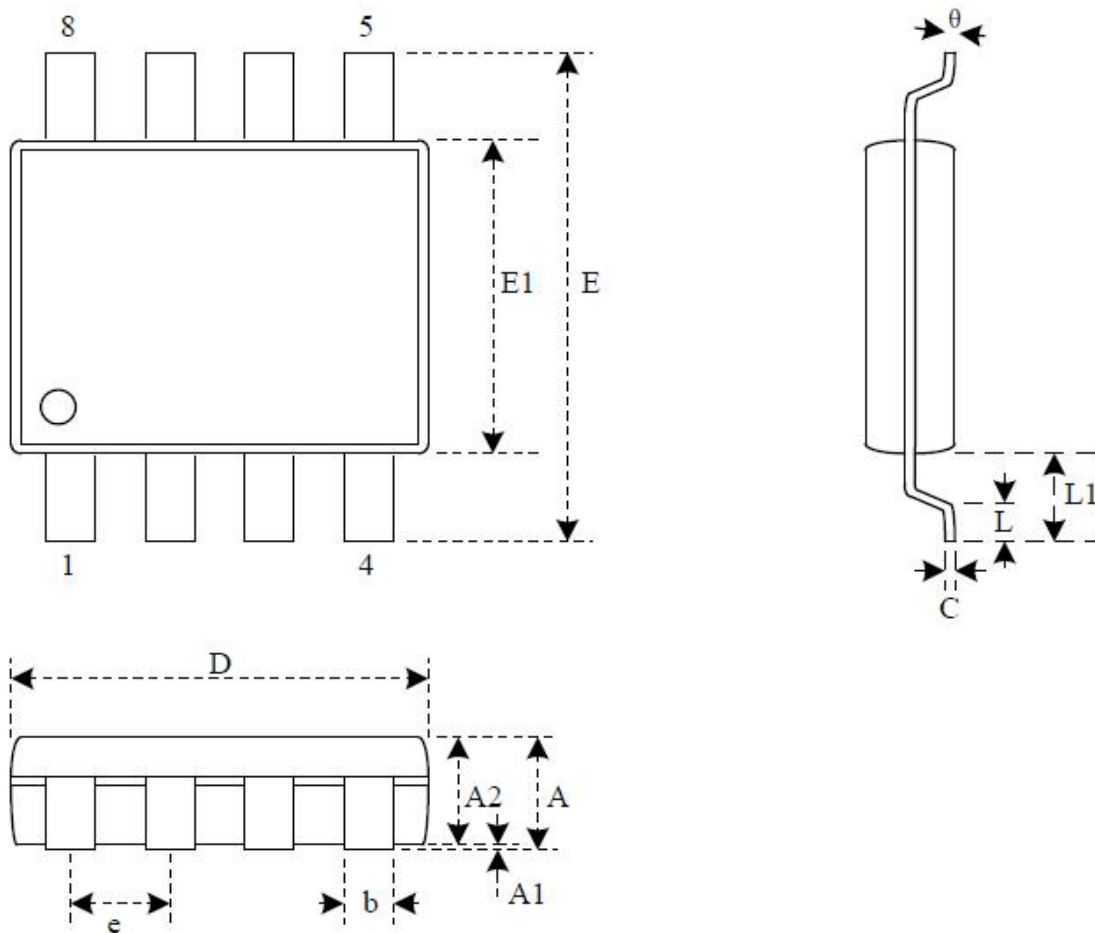
Symbol	Parameter	Min	Max	unit
T VSL	VCC(min)To/CS Low	10	-	us
T PUW	Time Delay From VCC(min)To Write Instruction	4	10	ms
VWI	Write Inhibit Voltage VCC(min)	1.5	2.5	v



## 6 封装尺寸

封装类型	封装尺寸
SOP8-B	5.28mmX7.90mm (208milX311mil)
DFN8 2X3	2.0mmx 3.0mm (79milX118mil)

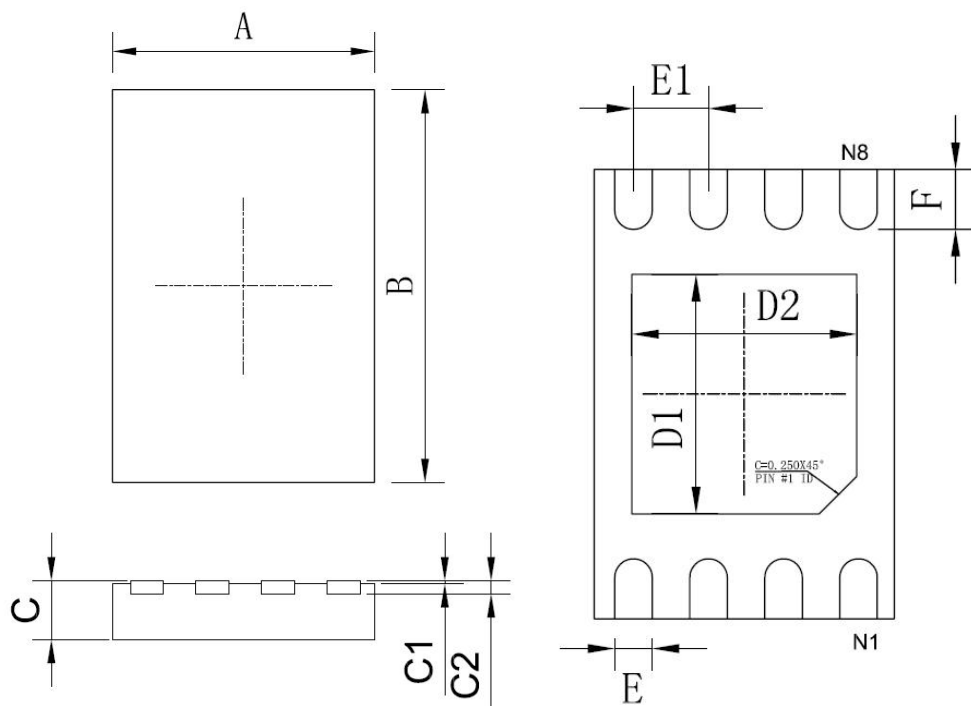
### SOP8-B



Dimensions

Symbol		A	A1	A2	b	C	D	E	E1		L	L1	S		
Unit															
Mm	Min.	-	0.05	1.70	0.36	0.19	5.13	7.70	5.18		0.50	1.21	0.62	0	
	Norm	-	0.15	1.80	0.41	0.20	5.23	7.90	5.28	1.27	0.65	1.31	0.74	5	
	Max.	2.16	0.25	1.91	0.51	0.25	5.33	8.10	5.38		0.80	1.41	0.88	8	
inch	Min.	-	0.002	0.067	0.014	0.007	0.202	0.303	0.204		0.020	0.048	0.024	0	
	Norm	-	0.006	0.071	0.016	0.008	0.206	0.311	0.208	0.050	0.026	0.052	0.029	5	
	Max.	0.085	0.010	0.075	0.020	0.010	0.210	0.319	0.212		0.031	0.056	0.035	8	

DFN8-2X3



DIMENSION LABEL 尺寸 标注	MIN (mm) 最小 (mm)	MAX (mm) 最大 (mm)	DIMENSION LABEL 尺寸 标注	MIN (mm) 最小 (mm)	MAX (mm) 最大 (mm)
A	2.0±0.1		D1	1.60TYP	
B	3.0±0.1		D2	1.50TYP	
C	0.70	0.80	E	0.250TYP	
C1	0~0.050		E1	0.500TYP	
C2	0.203TYP		F	0.400TYP	



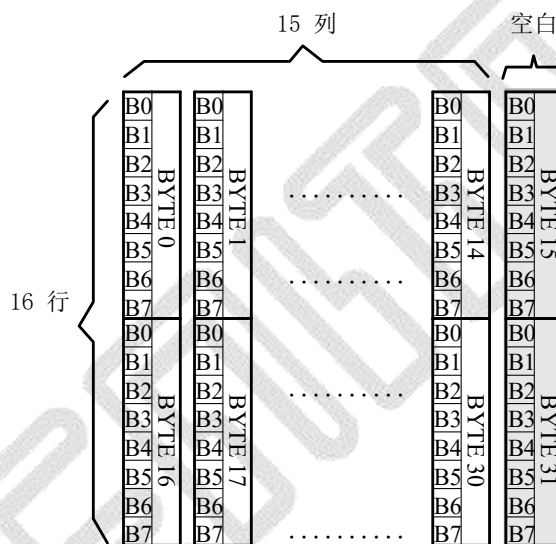
## 7 字库排置 (竖置横排)

### 7.1 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。

### 7.2 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：

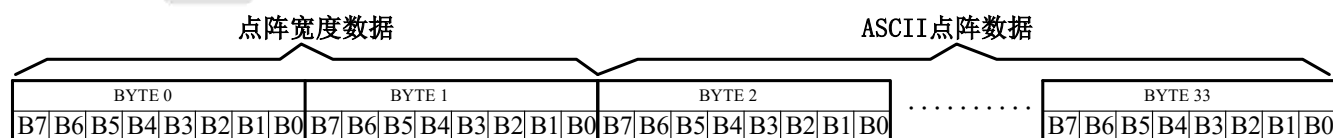


### 7.3 16 点阵不等宽 ASCII 圆角字体字符排列格式

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

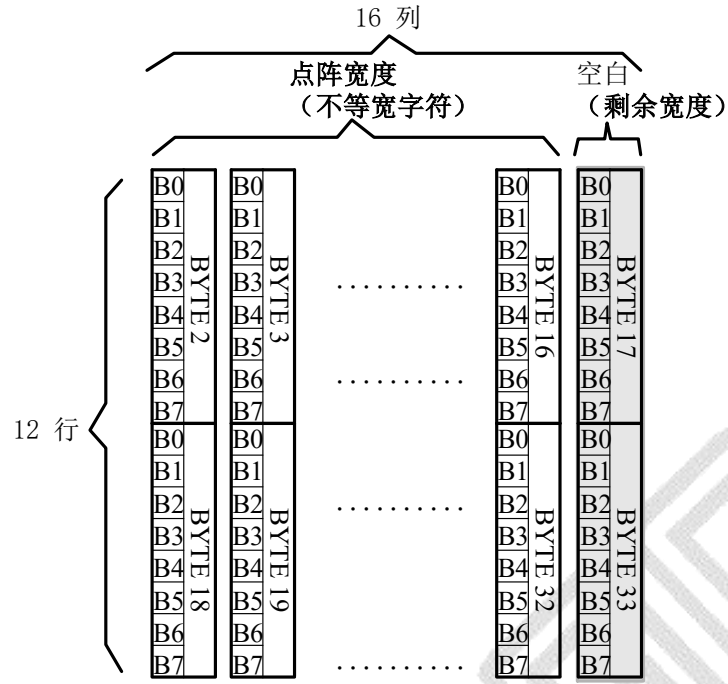
■ 存储格式

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

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。



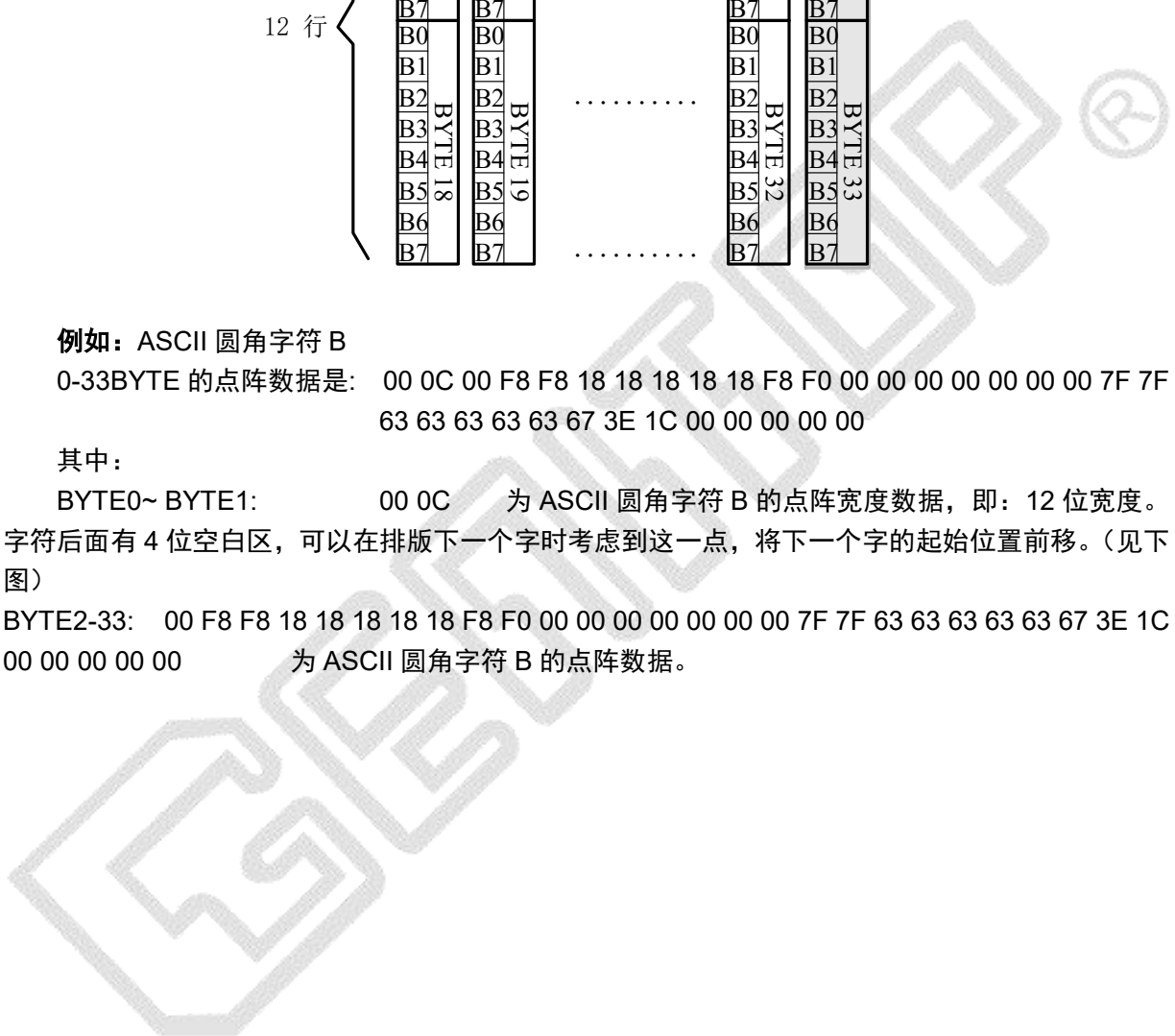
例如：ASCII 圆角字符 B

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。



## 8 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 00 0F 00





创 造 文 明 智 能

**深圳 OFFICE**

地址: 广东省深圳市福田区沙头街道泰然九路金润大厦 12C

电话: 0755-83453881 83453855

传真: 0755-83453855-8004