ER3304-1
DATASHEET

- **Character Set：GB2312**
- **Font Size：12x12、16x16、24x24、32x32**
- **Data Arrangement：horizontal byte, horizontal string**
- **Bus Interface：SPI**
- **Package：SOP8-B**

V 1.0 I _A

2013-04

INDEX

1 General

ER3304-1 contains 12x12、16x16、24x24、32x32 dot matrix Chinese font，supporting GB2312 Simplified Chinese and ASCII.The data arrangement format is horizontal byte, horizontal string. The user may obtain the address of certain character dot matrix with the calculation method given by this data sheet, which enables the user to access to more character data by continually reading from the address already obtained.

1.1 Chip Feature

- Bus Interface: SPI
- Data Arrangement: horizontal byte, horizontal string
- Frequency: 80MHz(max.)@3.3V
- Operating Voltage: 2.7V~3.6V
- Current:
  Operating：13mA
  Standby：1uA
- Package：SOP8-B
- Package Size：7.90mmX5.23mm
- Operating Temperature: -40℃~85℃ Storage Temperature: -55~125℃

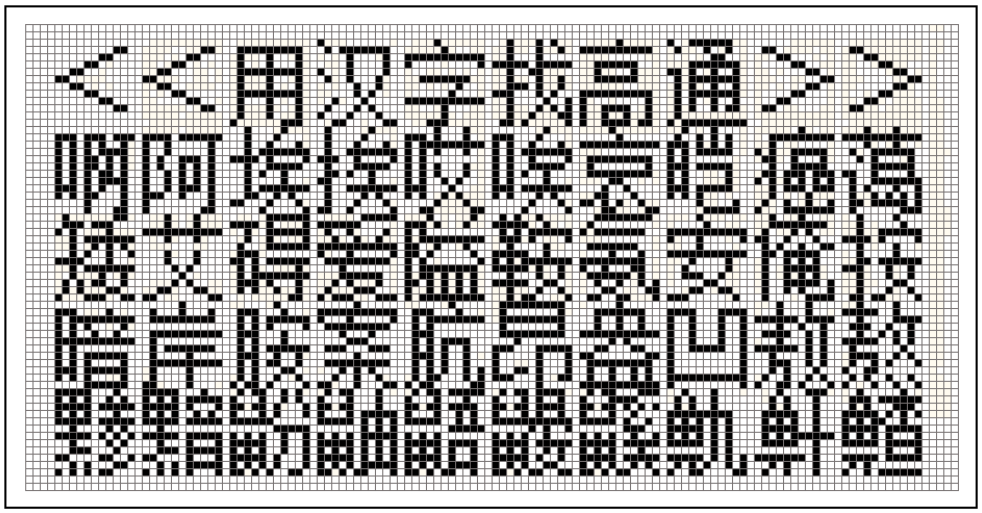
1.2 Font Content

Character Set	Content	Font Size	Number of Characters	Font Type	Data Arrangement	Remark
ASCII Character Set	ASCII	5x7	96	Basic	W-horizontal byte, horizontal string	
	ASCII	7x8	96	Basic	W	
	ASCII	6x12	96	Basic	W	
	ASCII	8x16	96	Basic	W	
	ASCII	12x24	96	Basic	W	
	ASCII	16x32	96	Basic	W	
	ASCII	12 dots proportional adjusted	96	Arial	W	
	ASCII	12 dots proportional adjusted	96	Times new Roman	W	
	ASCII	16 dots proportional adjusted	96	Arial	W	
	ASCII	16 dots proportional adjusted	96	Times new Roman	W	
	ASCII	24 dots proportional adjusted	96	Arial	W	
	ASCII	24 dots proportional adjusted	96	Times new Roman	W	
	ASCII	32 dots proportional adjusted	96	Arial	W	
	ASCII	32 dots proportional adjusted	96	Times new Roman	W	
Chinese Character Set	GB2312 Chinese Character	12x12	6763	Song typeface	W	
		16x16	6763	Song typeface	W	
		24x24	6763	Song typeface	W	
		32x32	6763	Song typeface	W	
	GB2312 Character	12x12	846	Song typeface	W	
		16x16	846	Song typeface	W	
		24x24	846	Song typeface	W	
		32x32	846	Song typeface	W	
	National standard extended character	6x12	126	Song typeface	W	
		8x16	126	Song typeface	W	
		12x24	126	Song typeface	W	
		16x32	126	Song typeface	W	

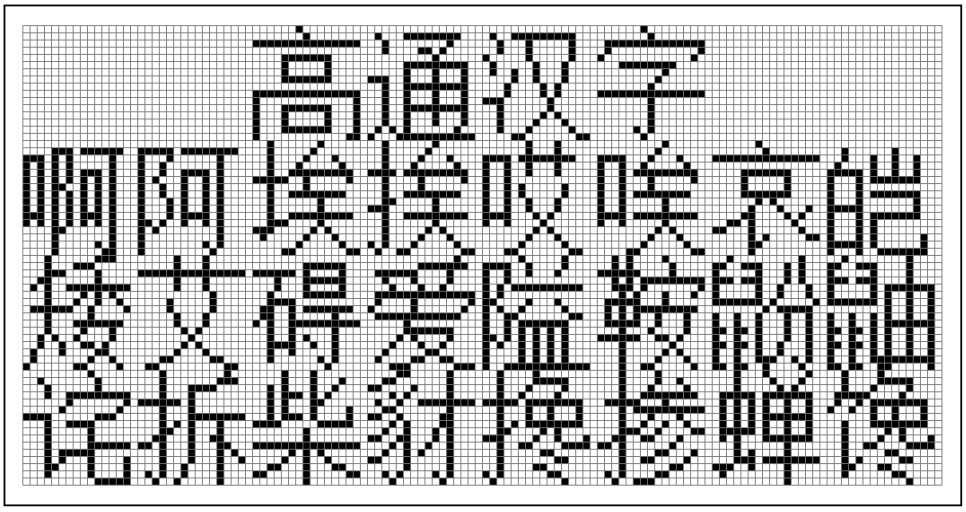
1.3 Font Sample

1.3.1 **GB2312 Chinese Character**

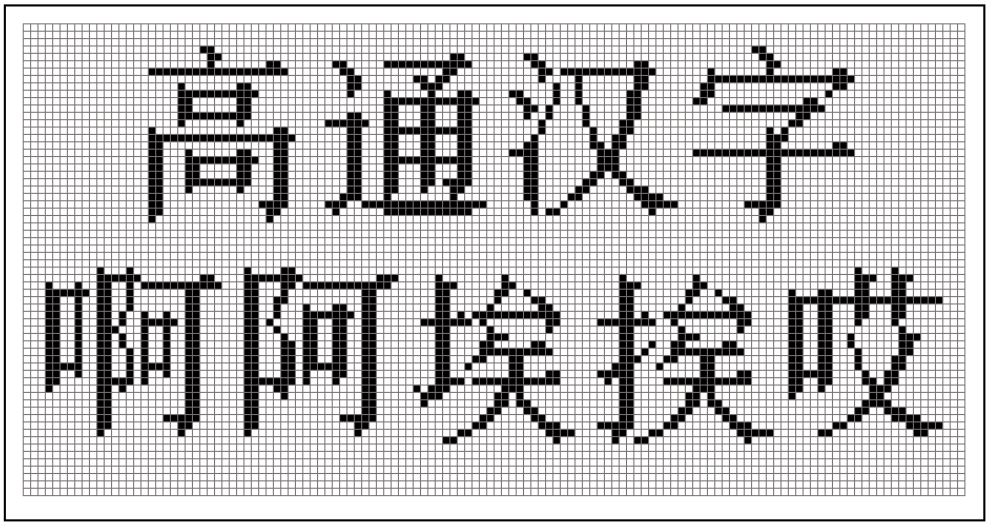
12x12 GB2312 Chinese Character

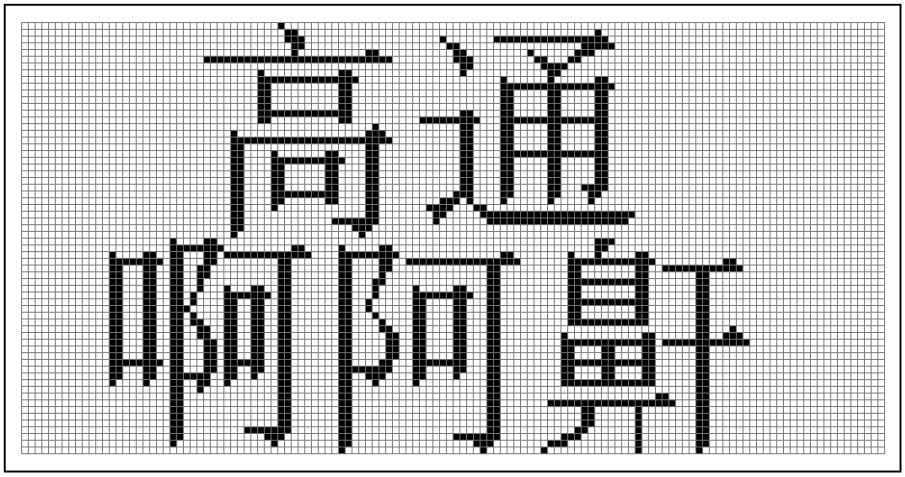


16x16 GB2312 Chinese Character

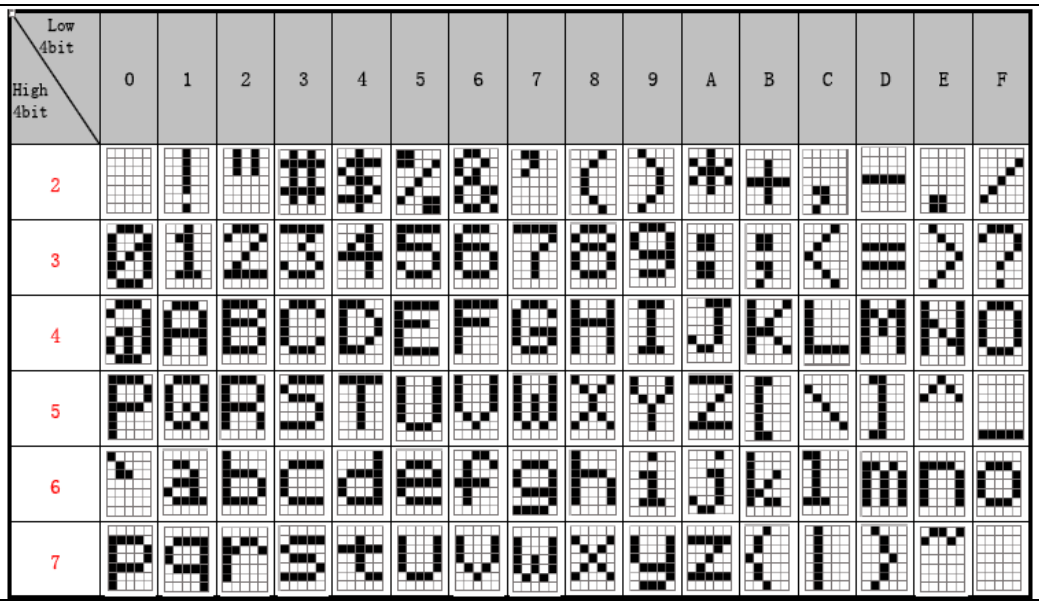


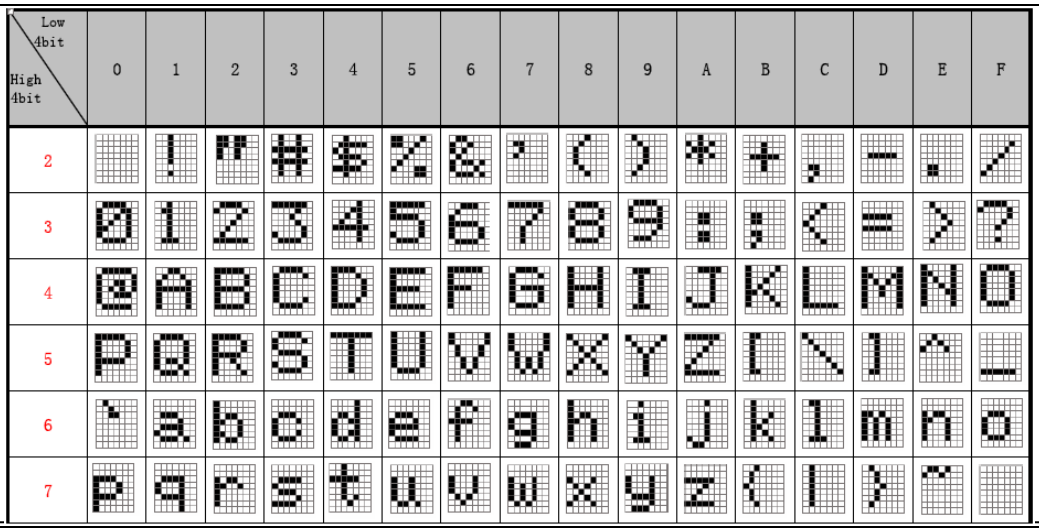
24x24 GB2312 Chinese Character



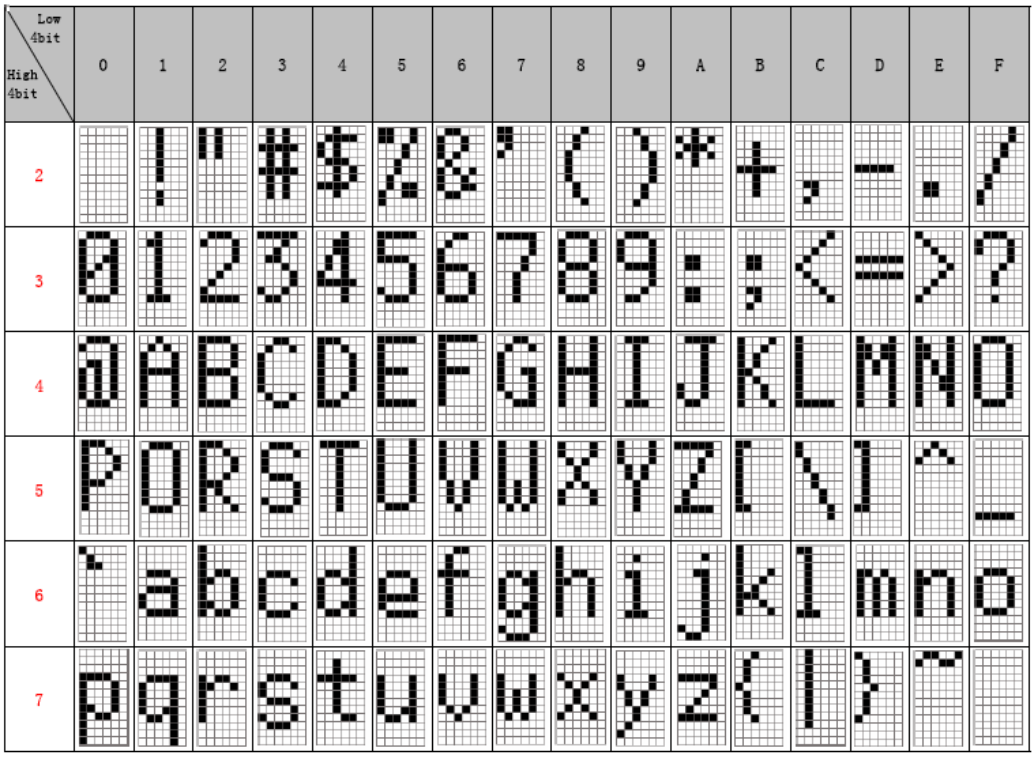
32x32 GB2312 Chinese Character


1.3.2 **Other dot matrix characters**

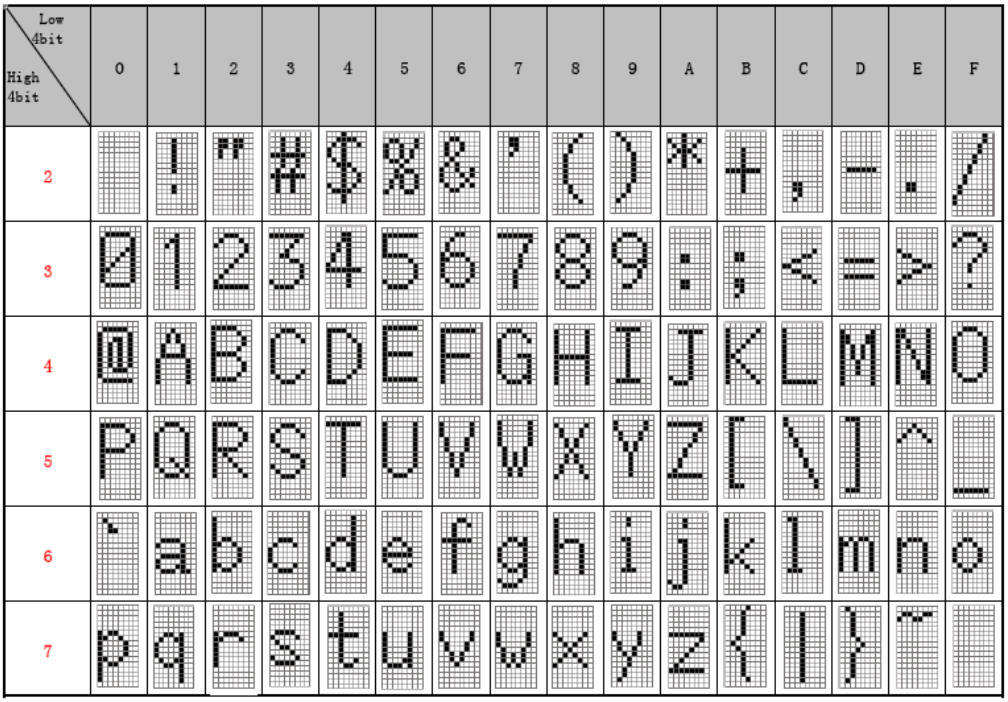
5x7 ASCII


7x8 ASCII


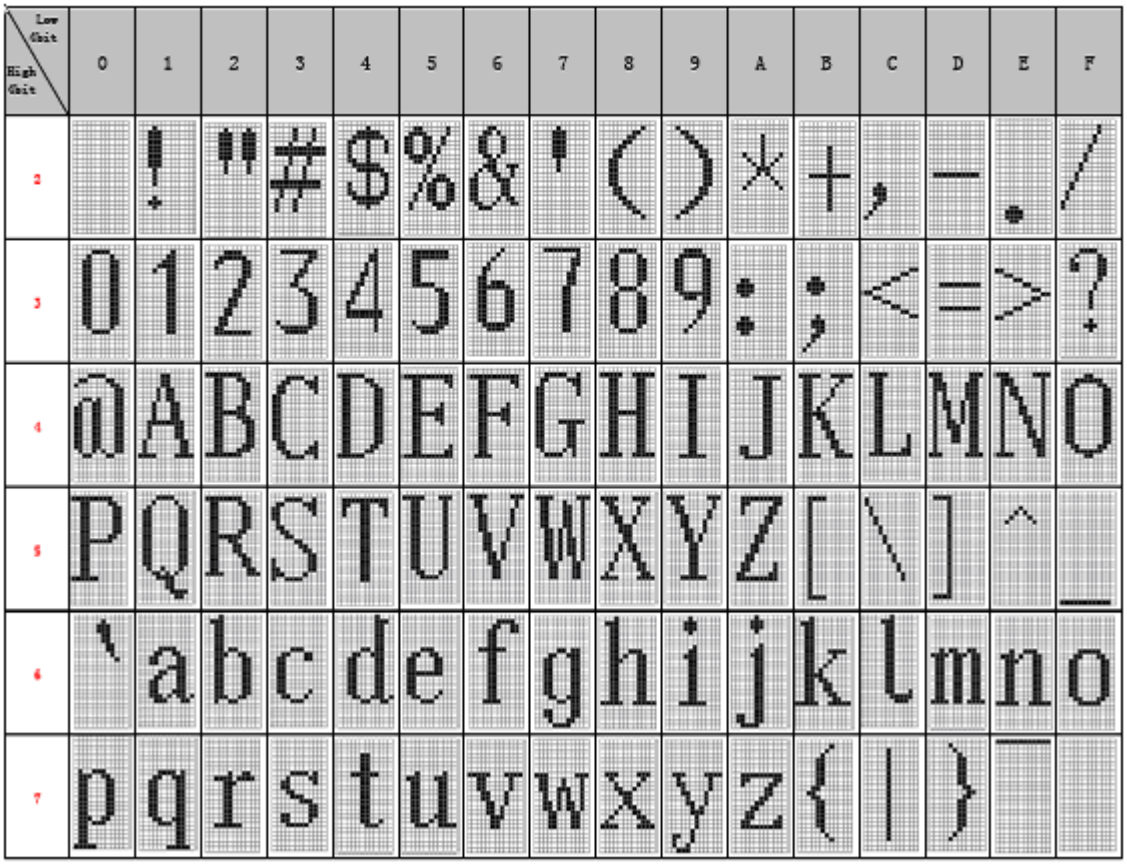
6x12 ASCII



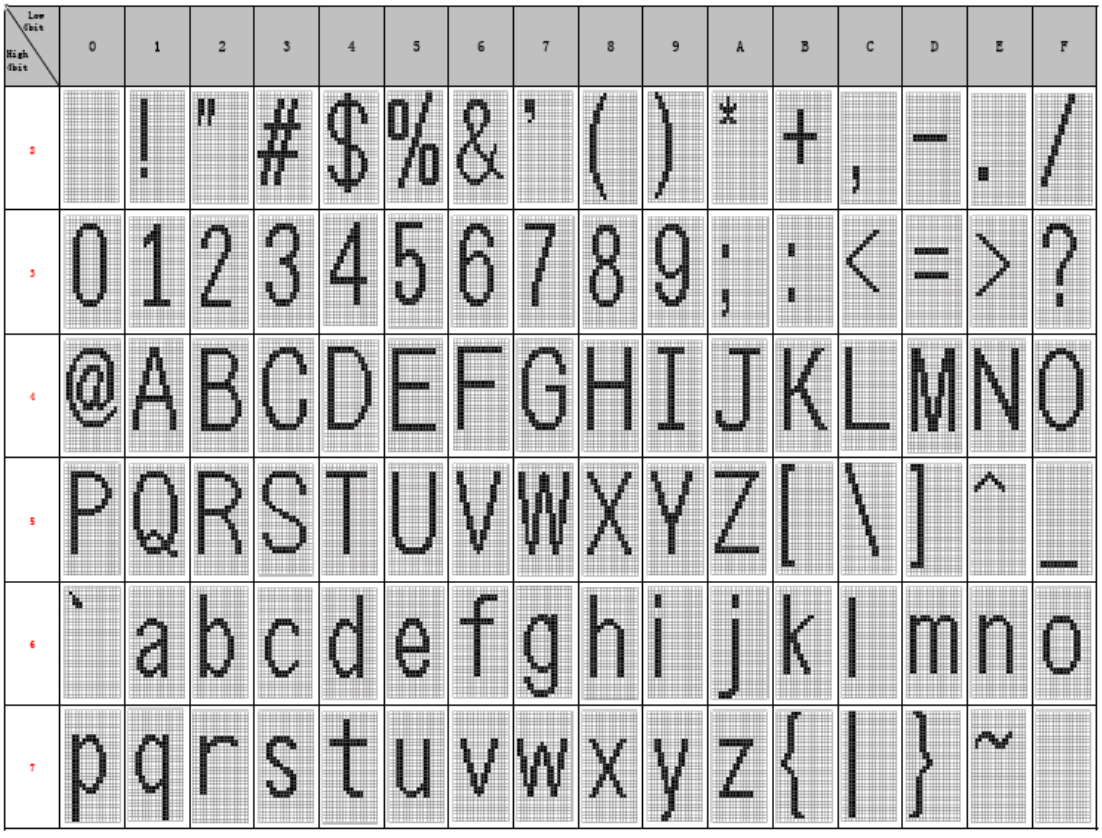
8x16 ASCII



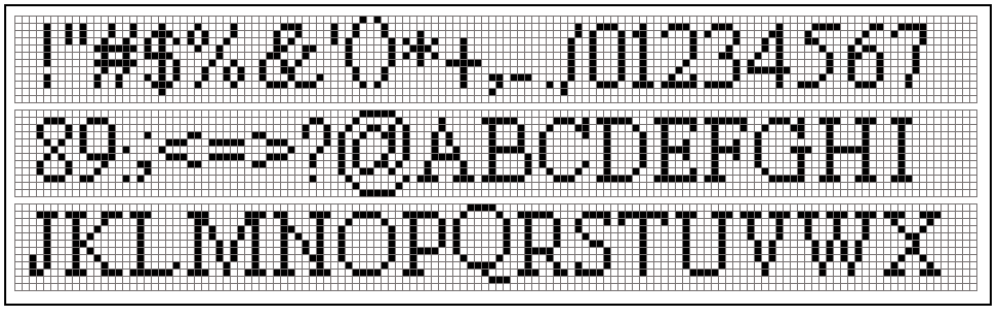
12x24 ASCII



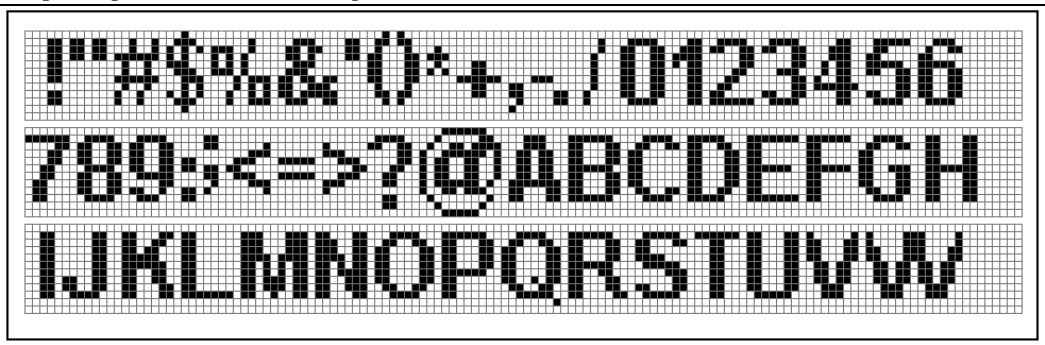
16x32 ASCII



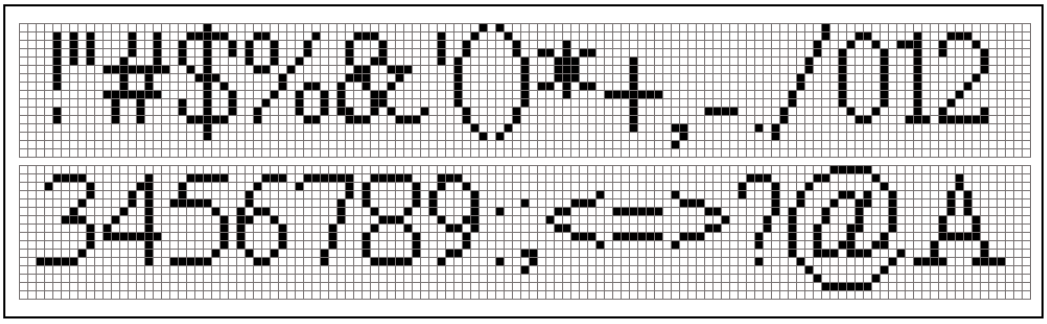
12 dots proportional adjusted ASCII (Times new Roman)



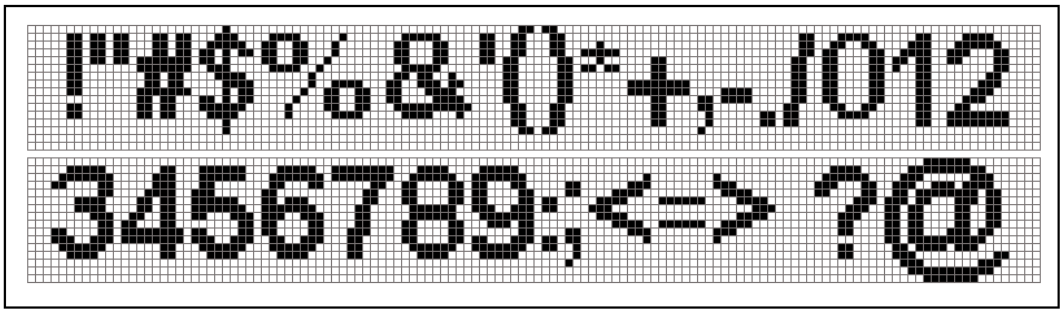
12 dots proportional adjusted ASCII （Arial）



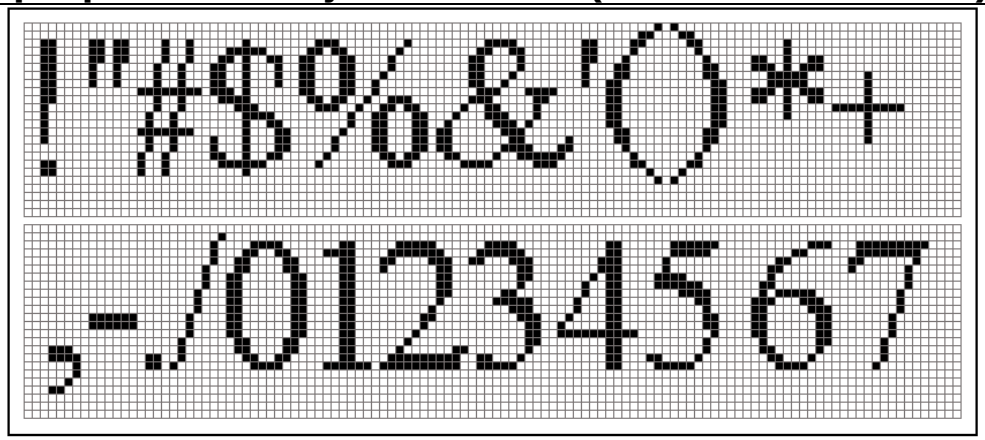
16 dots proportional adjusted ASCII (Times new Roman)



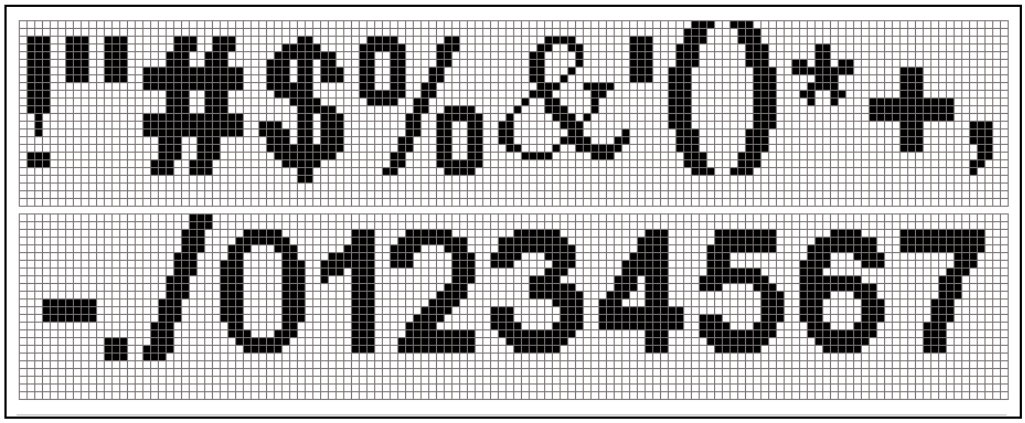
16 dots proportional adjusted ASCII （Arial）



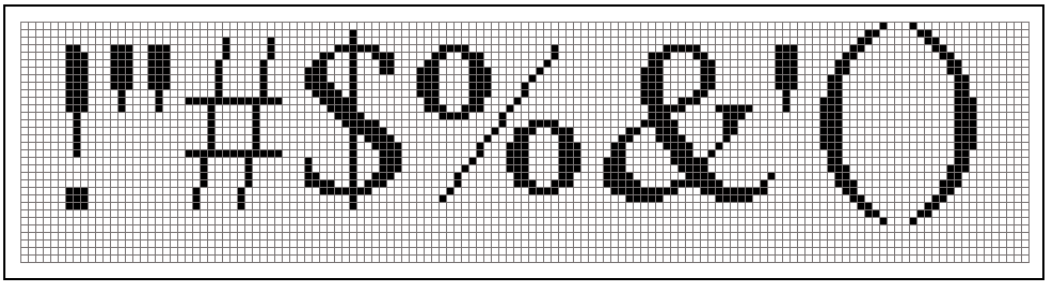
24 dots proportional adjusted ASCII (Times new Roman)



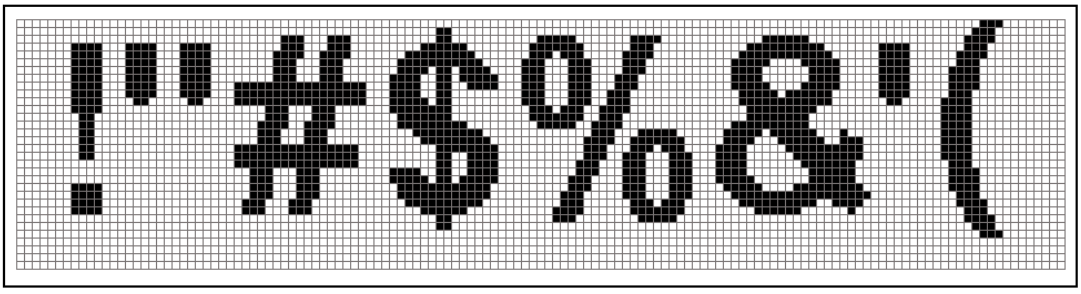
24 dots proportional adjusted ASCII （Arial）



32 dots proportional adjusted ASCII (Time new Roman)



32 dots proportional adjusted ASCII （Arial）



2 Operating Instruction

2.1 Instruction Parameter

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 ∞

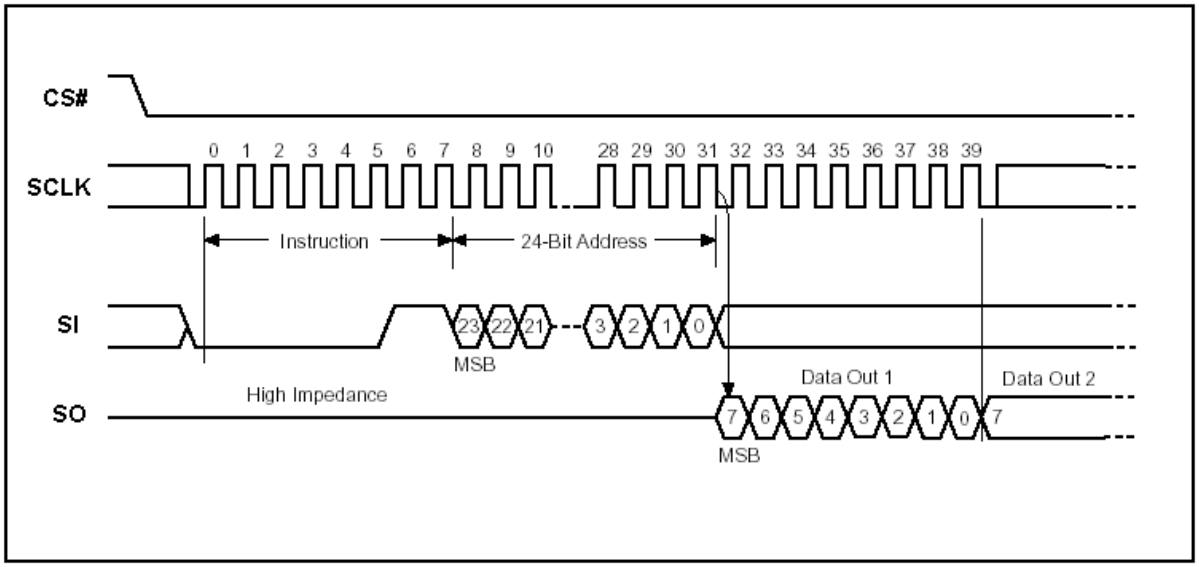
2.2 Read Data Bytes

Read Data Bytes need Instruction Code to carry out every operation. READ instruction process by below timing sequence (picture):

- First Chip-Selection signal(CS#) remains active low,1 byte command word(03 h) and 3 bytes address be shifted into Pin (SI) by bit, every bit be locked during serial clock (SCLK) remains active high.
- Then the data of this address be shifted out from Pin(SO) by bit during serial clock (SCLK) remains active low.
- After read those date bytes,Chip-Selection signal(CS#) remains active high, finish this operation.

  If Chip-Selection signal（CS#）remains active low，the data of next address be shifted out from Pin(SO) by bit

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



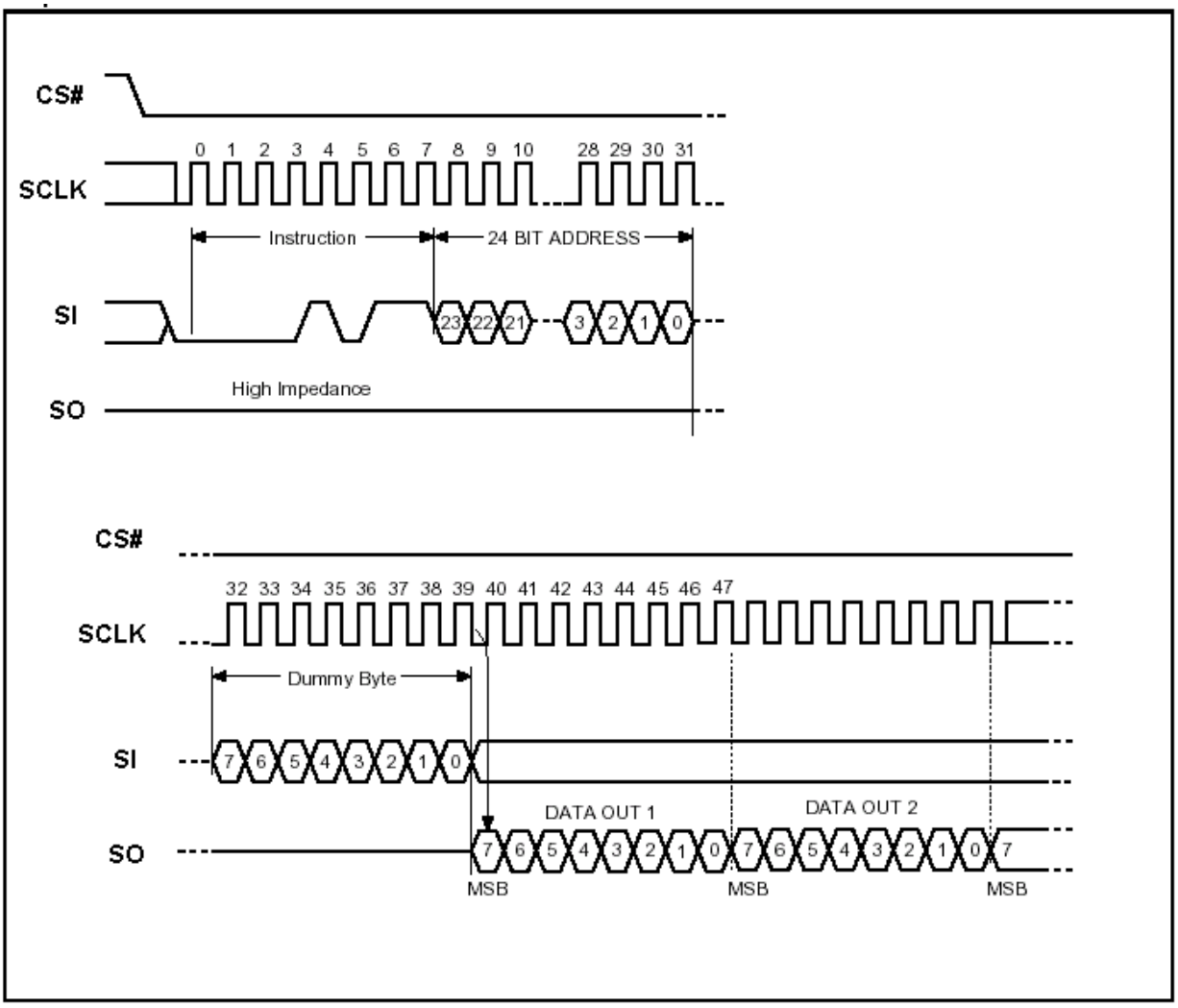
2.3 Read Data Bytes at Higher Speed

Read Data Bytes at Higher Speed need Instruction Code to carry out every operation. READ instruction process by below timing sequence (picture)：

- First Chip-Selection signal(CS#) remains active low, 1 byte command word(0Bh), 3 bytes address and 1 byte Dummy be shifted into Pin(SI) by bit,every bit be locked during serial clock (SCLK) remains active high.
- Then the data of this address be shifted out from Pin(SO) by bit during serial clock remains active low.
- If Chip-Selection signal(CS#) keeps active low, data of next address will be shifted out. Example:Read one 15x16 Chinese character need 32 Bytes, after read 32 bytes data, finish reading one Chinese character data.

  If no need to read next data,then remains Chip-Selection signal(CS#) active high, finish this operation.

Picture: Read Data Bytes at Higher Speed (READ_FAST) Instruction Sequence and Data-out sequence



3 Dot Matrix Font Address Table

NO.	Font Content	Character Set	Number of Characters	Base address	Reference algorithm
1	12x12 GB2312	GB2312	6763+846	0x00000	4.1.1
2	15x16 GB2312	GB2312	6763+846	0x2C9D0	4.1.2
3	24x24 GB2312	GB2312	6763+846	0x68190	4.1.3
4	32x32 GB2312	GB2312	6763+846	0xEDF00	4.1.4
5	6x12 National standard extended character	GB2312	126	0x1DBE0C	4.1.5
6	6x12 ASCII	ASCII	96	0x1DBE00	4.2.3
7	12 dots proportional adjusted ASCII (Arial)	ASCII	96	0x1DC400	4.2.7
8	12 dots proportional adjusted ASCII (Time)	ASCII	96	0x1DCDC0	4.2.8
9	8x16 National standard extended character	GB2312	126	0x1DD790	4.1.6
10	8x16 ASCII	ASCII	96	0x1DD780	4.2.4
11	5x7 ASCII	ASCII	96	0x1DDF80	4.2.1
12	7x8 ASCII	ASCII	96	0x1DE280	4.2.2
13	16 dots proportional adjusted ASCII (Arial)	ASCII	96	0x1DE580	4.2.9
14	16 dots proportional adjusted ASCIII (Time)	ASCII	96	0x1DF240	4.2.10
15	12x24 National standard extended character	GB2312	126	0x1DFF30	4.1.8
16	12x24 ASCII	ASCII	96	0x1DFF00	4.2.5
17	24 dots proportional adjusted ASCII (Arial)	ASCII	96	0x1E22D0	4.2.11
18	24 dots proportional adjusted ASCII (Time)	ASCII	96	0x1E3E90	4.2.12
19	16x32 National standard extended character	GB2312	126	0x1E5A90	4.1.9
20	16x32 ASCII	ASCII	96	0x1E5A50	4.2.6
21	32 dots proportional adjusted ASCII (Arial)	ASCII	96	0x1E99D0	4.2.13
22	32 dots proportional adjusted ASCII (Time)	ASCII	96	0x1ECA90	4.2.14
23	Service area			0x1EFB50	
24	8x16 GB2312 Special character	GB2312	64	0x1F2880	4.1.7
25	Service area			0x1F7CC8	

4 Calculation of Character Address

With certain calculation method, the user may obtain certain character dots address using internal code.

4.1 Calculation of GB2312 Chinese Character Address

4.1.1 12x12 GB2312

Parameter：
GBCode: Character code.
MSB: High byte of GB code.
LSB: Low byte of GB code.
Address: Address of character data in chip.
BaseAdd: Base address of font in chip.
Calculation of character address:

```
BaseAdd=0x0;
if(MSB >=0xA1 && MSB <= 0Xa9 && LSB >=0xA1)
    Address =( (MSB - 0xA1) * 94 + (LSB - 0xA1))*24+ BaseAdd;
else if(MSB >=0xB0 && MSB <= 0xF7 && LSB >=0xA1)
    Address = ((MSB - 0xB0) * 94 + (LSB - 0xA1)+ 846)*24+ BaseAdd;
```

4.1.2 15x16 GB2312

Parameter：
GBCode: Character code.
MSB: High byte of GB code.
LSB: Low byte of GB code.
Address: Address of character data in chip.
BaseAdd: Base address of font in chip.
Calculation of character address:

```
BaseAdd=0x2C9D0;
if(MSB >=0xA1 && MSB <= 0Xa9 && LSB >=0xA1)
    Address =( (MSB - 0xA1) * 94 + (LSB - 0xA1))*32+ BaseAdd;
else if(MSB >=0xB0 && MSB <= 0xF7 && LSB >=0xA1)
    Address = ((MSB - 0xB0) * 94 + (LSB - 0xA1)+ 846)*32+ BaseAdd;
```

4.1.3 24x24 GB2312

Parameter：
GBCode: Character code.
MSB: High byte of GB code.
LSB: Low byte of GB code.
Address: Address of character data in chip.
BaseAdd: Base address of font in chip.

Calculation of character address:

```
BaseAdd=0x68190;
if(MSB >=0xA1 && MSB <= 0Xa9 && LSB >=0xA1)
    Address =( (MSB - 0xA1) * 94 + (LSB - 0xA1))*72+ BaseAdd;
else if(MSB >=0xB0 && MSB <= 0xF7 && LSB >=0xA1)
    Address = ((MSB - 0xB0) * 94 + (LSB - 0xA1)+ 846)*72+ BaseAdd;
```

4.1.4 32x32 GB2312

Parameter：

GBCode: Character code.
MSB: High byte of GB code.
LSB: Low byte of GB code.
Address: Address of character data in chip.
BaseAdd: Base address of font in chip.

Calculation of character address:

```
BaseAdd=0XEDF00;
if(MSB >=0xA1 && MSB <= 0Xa9 && LSB >=0xA1)
    Address =( (MSB - 0xA1) * 94 + (LSB - 0xA1))*128+ BaseAdd;
else if(MSB >=0xB0 && MSB <= 0xF7 && LSB >=0xA1)
    Address = ((MSB - 0xB0) * 94 + (LSB - 0xA1)+ 846)*128+ BaseAdd;
```

4.1.5 6x12 National standard extended character

Parameter：

BaseAdd: Base address of font in chip.
FontCode: Character code （16bits）
ByteAddress: Address of character data in chip.

Calculation of character address:

```
BaseAdd=0x1DBE0C
if (FontCode>= 0xAAA1 && FontCode<=0xAAFE )
    ByteAddress = (FontCode–0xAAA1 ) * 12+BaseAdd;
Else if(FontCode>= 0xABA1 && FontCode<=0xABC0 )
    ByteAddress = (FontCode–0xABA1 + 95) * 12+BaseAdd;
```

4.1.6 8x16 National standard extended character

Parameter：

BaseAdd: Base address of font in chip.
FontCode: Character code（16bits）
ByteAddress: Address of character data in chip.

Calculation of character address:

```
BaseAdd=0x1DD790
if (FontCode>= 0xAAA1 && FontCode<=0xAAFE )
```

ByteAddress = (FontCode–0xAAA1) * 16+BaseAdd;
Else if(FontCode>= 0xABA1 && FontCode<=0xABC0)
ByteAddress = (FontCode–0xABA1 + 95) * 16+BaseAdd;

4.1.7 8x16 GB2312 Special Character

Parameter：
BaseAdd: Base address of font in chip.
FontCode: Character code（16bits）
ByteAddress: Address of character data in chip.
Calculation of character address:
BaseAdd=0x1F2880
if (FontCode >= 0xACA1 && FontCode <=0xACDF)
ByteAddress = (FontCode–0xACA1) * 16+BaseAdd;

4.1.8 12x24 National standard extended character

Parameter：
BaseAdd: Base address of font in chip.
FontCode: Character code（16bits）
ByteAddress: Address of character data in chip.
Calculation of character address:
BaseAdd=0x1DFF30
if (FontCode>= 0xAAA1 && FontCode<=0xAAFE)
ByteAddress = (FontCode–0xAAA1) * 48+BaseAdd;
Else if(FontCode>= 0xABA1 && FontCode<=0xABC0)
ByteAddress = (FontCode–0xABA1 + 95) * 48+BaseAdd;

4.1.9 16x32 National standard extended character

Parameter：
BaseAdd: Base address of font in chip.
FontCode: Character code（16bits）
ByteAddress: Address of character data in chip.
Calculation of character address:
BaseAdd=0x1E5A90
if (FontCode>= 0xAAA1 && FontCode<=0xAAFE)
ByteAddress = (FontCode–0xAAA1) * 64+BaseAdd;
Else if(FontCode>= 0xABA1 && FontCode<=0xABC0)
ByteAddress = (FontCode–0xABA1 + 95) * 64+BaseAdd;

4.2 Calculation of ASCII Character Address

4.2.1 5x7 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DDF80
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 8+BaseAdd;

4.2.2 7x8 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DE280
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 8+BaseAdd;

4.2.3 6x12 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DBE00
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 12+BaseAdd;

4.2.4 8x16 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DD780
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)

Address = (ASCIICode –0x20) * 16+BaseAdd;

4.2.5 12x24 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DFF00
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 48+BaseAdd;

4.2.6 16x32 ASCII

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1E5A50
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 64+BaseAdd;

4.2.7 12 dots proportional adjusted ASCII（Arial）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DC400
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 26 + BaseAdd;

4.2.8 12 dots proportional adjusted ASCII（Times New Roman）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DCDC0
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)

Address = (ASCIICode –0x20) *26 + BaseAdd;

4.2.9 16 dots proportional adjusted ASCII（Arial）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DE580
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 34 + BaseAdd;

4.2.10 16 dots proportional adjusted ASCII（Times New Roman）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1DF240
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 34 + BaseAdd;

4.2.11 24 dots proportional adjusted ASCII（Arial）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1E22D0
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 74 + BaseAdd;

4.2.12 24 dots proportional adjusted ASCII（Times New Roman）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1E3E90
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)

Address = (ASCIICode –0x20) * 74+ BaseAdd;

4.2.13 32 dots proportional adjusted ASCII（Arial）

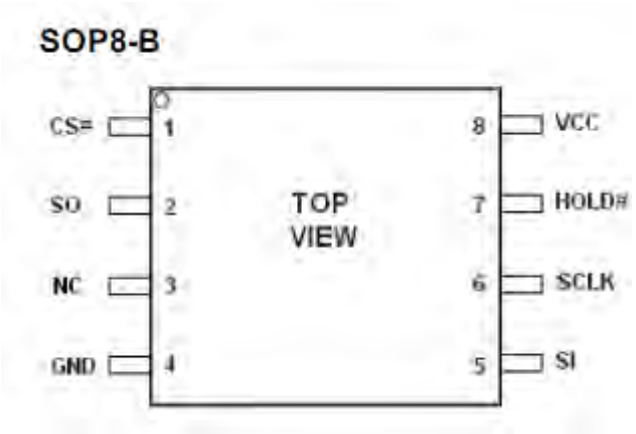
Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1E99D0
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 130 + BaseAdd;

4.2.14 32 dots proportional adjusted ASCII（Times New Roman）

Parameter：
ASCIICode: ASCII character code（8bits）
BaseAdd: Base address of font in chip.
Address: Address of ASCII character data in chip.
Calculation of character address:
BaseAdd=0x1ECA90
if (ASCIICode >= 0x20 && ASCIICode <= 0x7E)
Address = (ASCIICode –0x20) * 130 + BaseAdd;

5 Pin Description and Circuit Connection

5.1 Pin Configuration



SOP8-B

NO.	Name	I/O	Description
1	CS#	I	Chip enable input
2	SO	O	Serial data output
3	NC		Empty pin
4	GND		Ground
5	SI	I	Serial data input
6	SCLK	I	Serial clock input
7	HOLD#	I	Hold, to pause the device without
8	VCC		+ 3.3V Power Supply

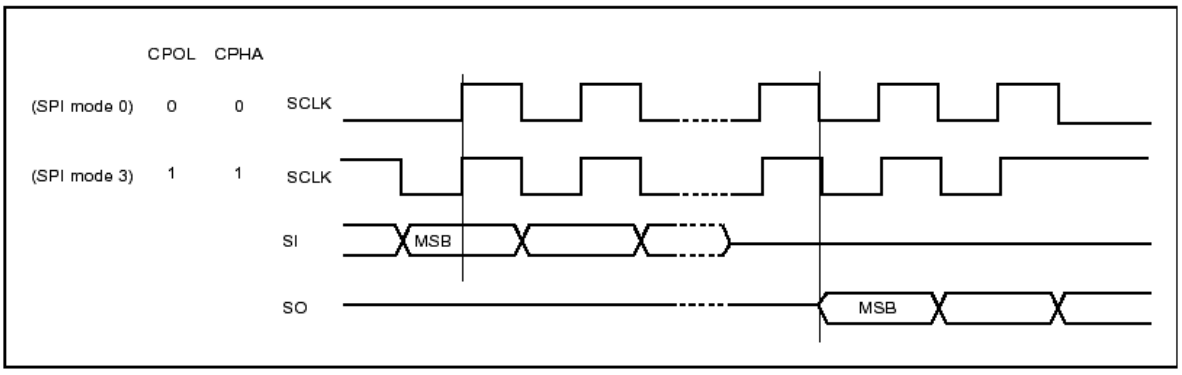
5.2 Pin Description

Serial Data Output (SO): Data shift-out on the falling edge of the serial clock.

Serial Data Input (SI): Data shift-in on the rising edge of the serial clock.

Serial Clock Input (SCLK): Data shift-out on the falling edge of the serial clock, shift-in on the rising edge of the serial clock.

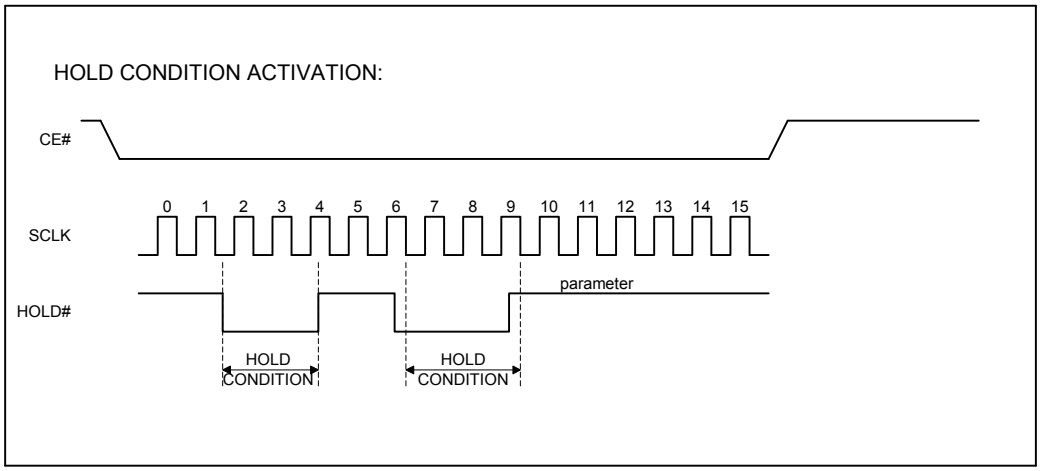
Chip Enable Input (CS#): The device is enabled by a high to low transition on CE#. CE# must remain low for the duration of any command sequence



HOLD#: To temporarily stop serial communication with SPI memory without resetting the device.

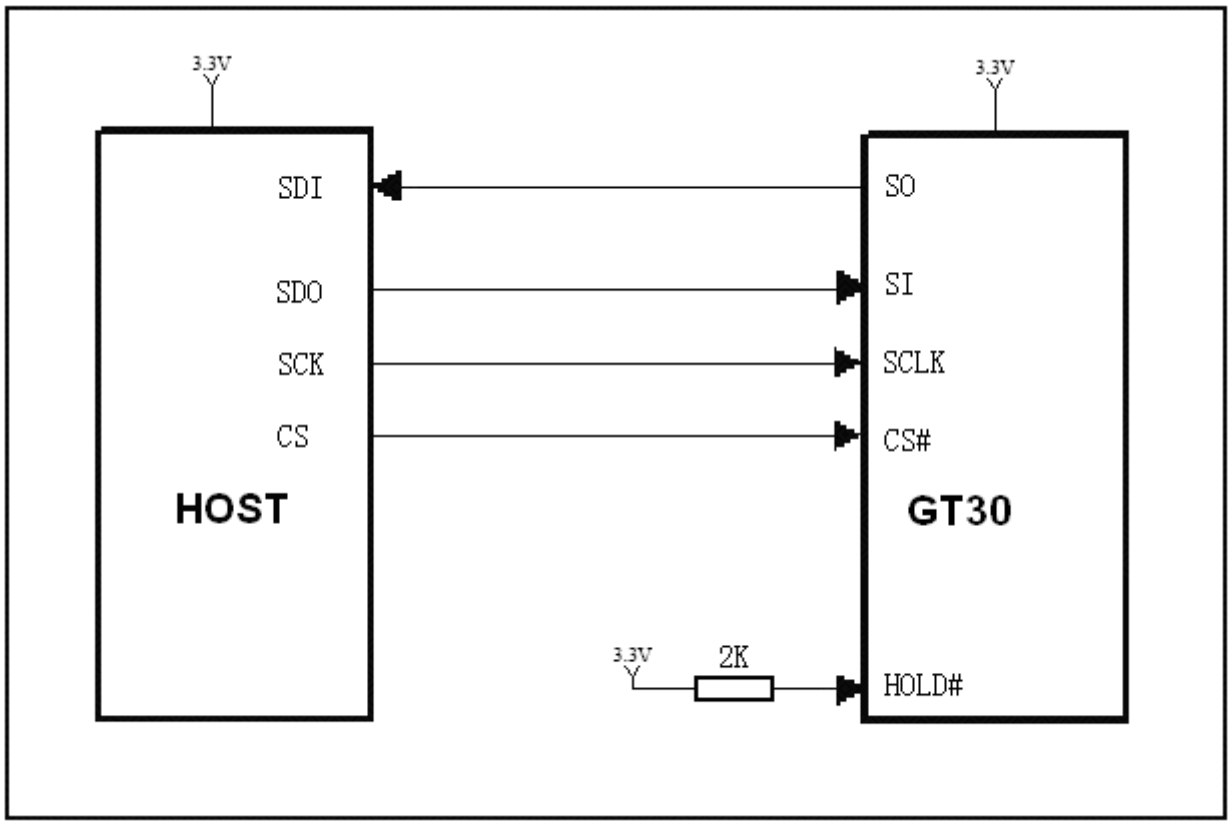
The HOLD# mode begins when the SCK active low state coincides with the falling edge of the HOLD# signal.

The HOLD mode ends when the HOLD# signal's rising edge coincides with the SCK active low state.



5.3 SPI Connection Block Diagram

Refer to below picture(# HOLD pin is recommended to pick up 2K resistor 3.3V pulled)



SPI Connection Block Diagram

6 Electrical Characteristic

6.1 Absolute Maximum Rating

Symbol	Parameter	Min.	Max.	Unit	Condition
T_{OP}	Operating Temperature	-40	85	℃	SPI mode
T_{STG}	Storage Temperature	-55	125	℃	
VCC	Supply Voltage	-0.5	4.0	V	
V_{IN}	Input Voltage	-0.5	4.0	V	
GND	Power Ground	0	0	V	

6.2 DC Characteristic

Condition：T_{OP} = -40℃ to 85℃，GND=0V in SPI mode；

Symbol	Parameter	Min.	Max.	Unit	Condition
I_{DD}	VCC Supply Current(active)		20	mA	
I_{SB}	VCC Standby Current		5	uA	
V_{IL}	Input LOW Voltage	-0.3	0.6	V	VCC=2.7V-3.6V
V_{IH}	Input HIGH Voltage	0.7VCC	VCC+0.3	V	
V_{OL}	Output LOW Voltage		0.4 (I_{OL}=1.6mA)	V	
V_{OH}	Output HIGH Voltage	VCC-0.2 (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，

6.3 AC Characteristic

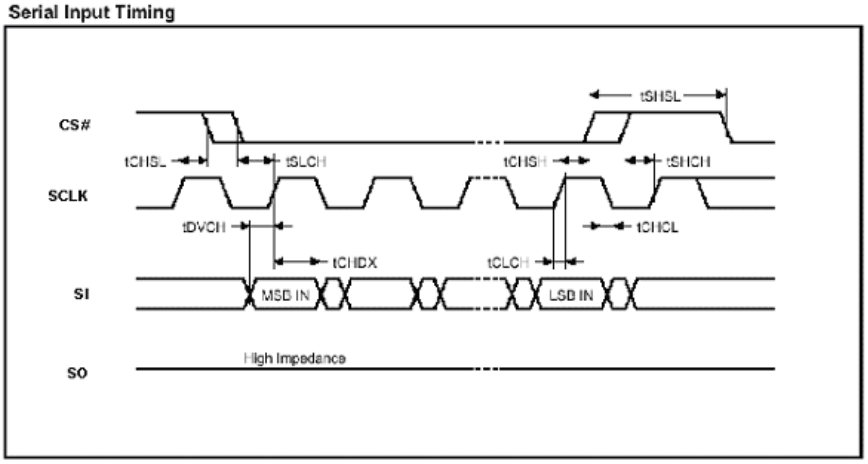
6.3.1 AC Characteristic under SPI

Condition：T_{OP} = -20℃ to 85℃，VCC= 2.7V to 3.6V

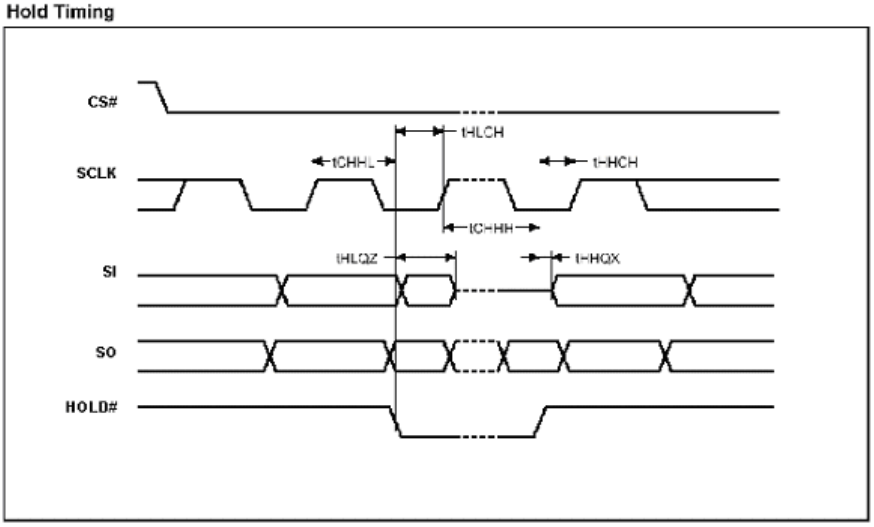
Symbol	Alt.	Parameter	Min.	Max.	Unit
Fc	Fc	Clock Frequency	D.C.	80	MHz
t_{CH}	t_{CLH}	Clock High Time	4		ns
t_{CL}	t_{CLL}	Clock Low Time	4		ns
t_{CLCH}		Clock Rise Time(peak to peak)	0.2		V/ns
t_{CHCL}		Clock Fall Time (peak to peak)	0.2		V/ns
t_{SLCH}	t_{CSS}	CS# Active Setup Time (relative to SCLK)	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	2		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		6	ns
t_{CLQV}	t_{V}	Clock Low to Output Valid		7	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		6	ns
t HLQZ	tHZ	HOLD# to Output High-Z		6	ns

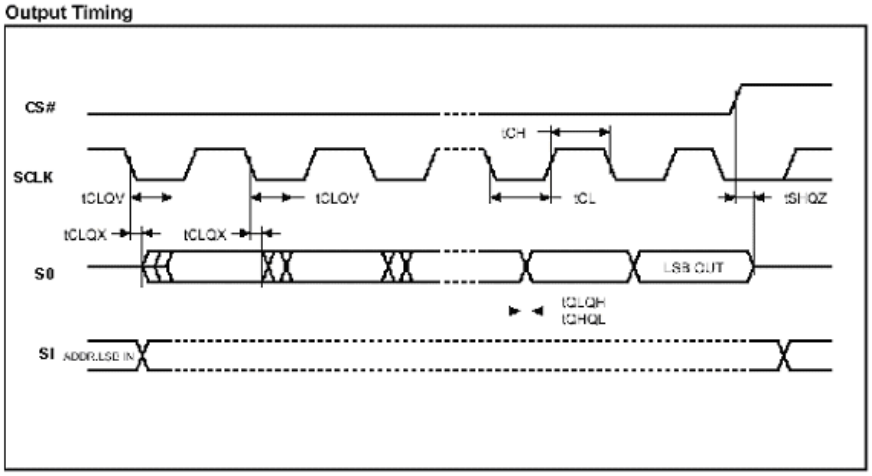
Serial Input Timing



Hold Timing

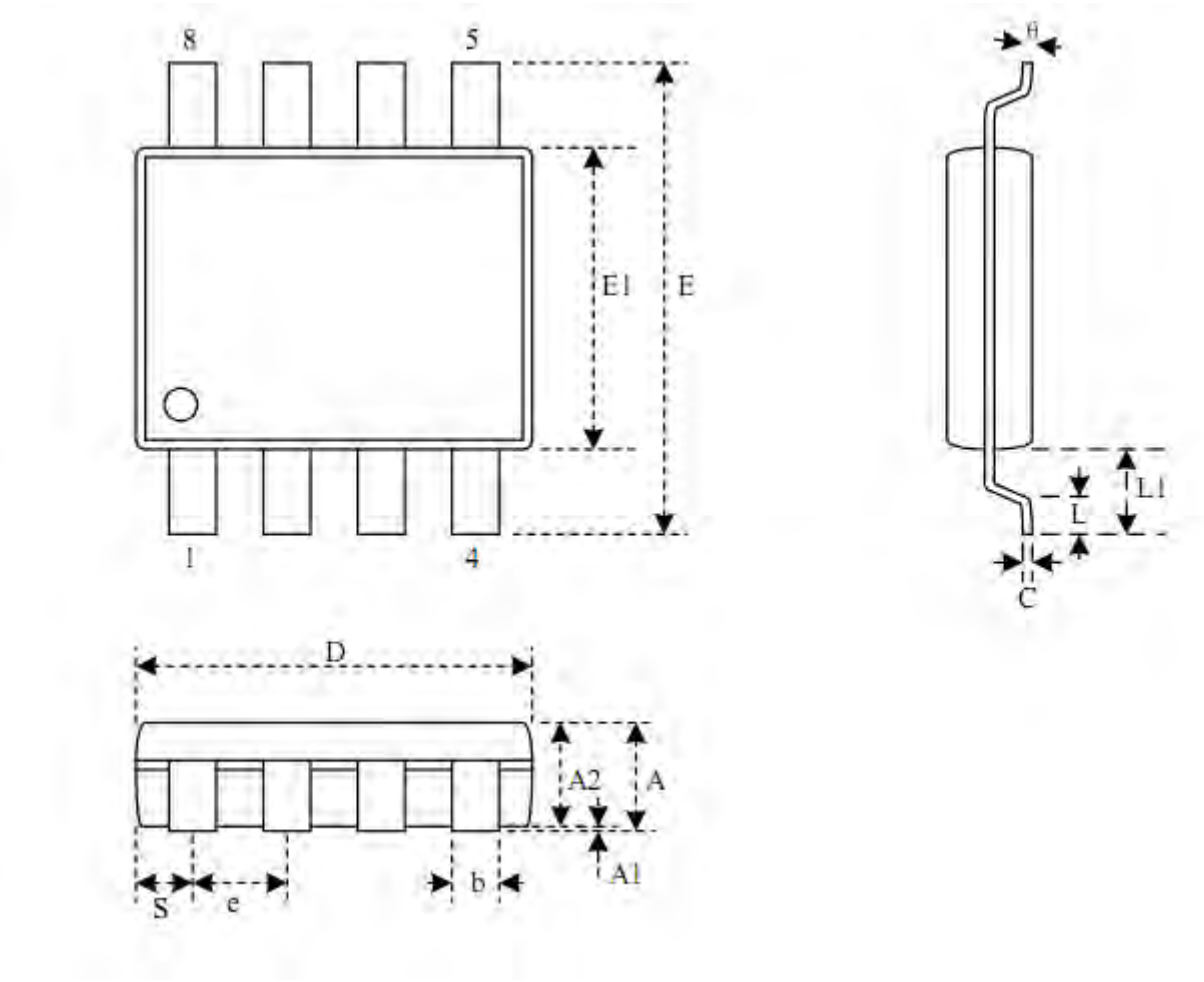


Output Timing



7 Package Size: SOP8-B

unit: mm



Dimensions

Symbol / Unit		A	A1	A2	b	C	D	E	E1	e	L	L1	S	θ
mm	Min		0.05	1.70	0.36	0.19	5.13	7.70	5.18		0.50	1.21	0.62	0
	Nom		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
	Nom		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

8 Appendix

8.1 Section 1 of GB2312 (846 characters)

Section 1 of GB2312 include 846 characters corresponding to code position A1A1~A9EF

GB2312 section1

A1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A			、	。	·	ˉ	ˇ	¨	〃	々	—	～	‖	…	‘	’
B	“	”	〔	〕	〈	〉	《	》	「	」	『	』	〖	〗	【	】
C	±	×	÷	∶	∧	∨	∑	∏	∪	∩	∈	∷	√	⊥	∥	∠
D	⌒	⊙	∫	∮	≡	≌	≈	∽	∝	≠	≮	≯	≤	≥	∞	∵
E	∴	♂	♀	°	′	″	℃	$	¤	¢	£	‰	§	№	☆	★
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	€		㈠	㈡	㈢	㈣	㈤	㈥	㈦	㈧	㈨	㈩	
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	｛	｜	｝	￣	

GB2312　section1

A4	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		あ	あ	ぃ	い	う	う	ぇ	え	お	お	か	が	き	ぎ	く
B	ぐ	け	げ	こ	ご	さ	ざ	し	じ	す	ず	せ	ぜ	そ	ぞ	た
C	だ	ち	ぢ	っ	つ	づ	て	で	と	ど	な	に	ぬ	ね	の	は
D	ば	ぱ	ひ	び	ぴ	ふ	ぶ	ぷ	へ	べ	ぺ	ほ	ぼ	ぽ	ま	み
E	む	め	も	ゃ	や	ゅ	ゆ	ょ	よ	ら	り	る	れ	ろ	ゎ	わ
F	ゐ	ゑ	を	ん												

A5	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		ァ	ア	ィ	イ	ウ	ウ	エ	エ	オ	オ	カ	ガ	キ	ギ	ク
B	グ	ケ	ゲ	コ	ゴ	サ	ザ	シ	ジ	ス	ズ	セ	ゼ	ソ	ゾ	タ
C	ダ	チ	ヂ	ッ	ツ	ヅ	テ	デ	ト	ド	ナ	ニ	ヌ	ネ	ノ	ハ
D	バ	パ	ヒ	ビ	ピ	フ	ブ	プ	ヘ	ベ	ペ	ホ	ボ	ポ	マ	ミ
E	ム	メ	モ	ャ	ヤ	ュ	ユ	ョ	ヨ	ラ	リ	ル	レ	ロ	ヮ	ワ
F	ヰ	ヱ	ヲ	ン	ヴ	ヵ	ヶ									

A6	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
B	Π	Ρ	Σ	Τ	Υ	Φ	Χ	Ψ	Ω							
C		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	'	°	`	:	;	!	?
E	ˆ	ˇ	ˆ	ˇ	ˆ	ˇ	≪	≫	ˍ	˩	˧	˥	˩	˧	▬	▬
F	ˆ	ˇ	∣	⋮	∣	⦚										

GB2312 section1

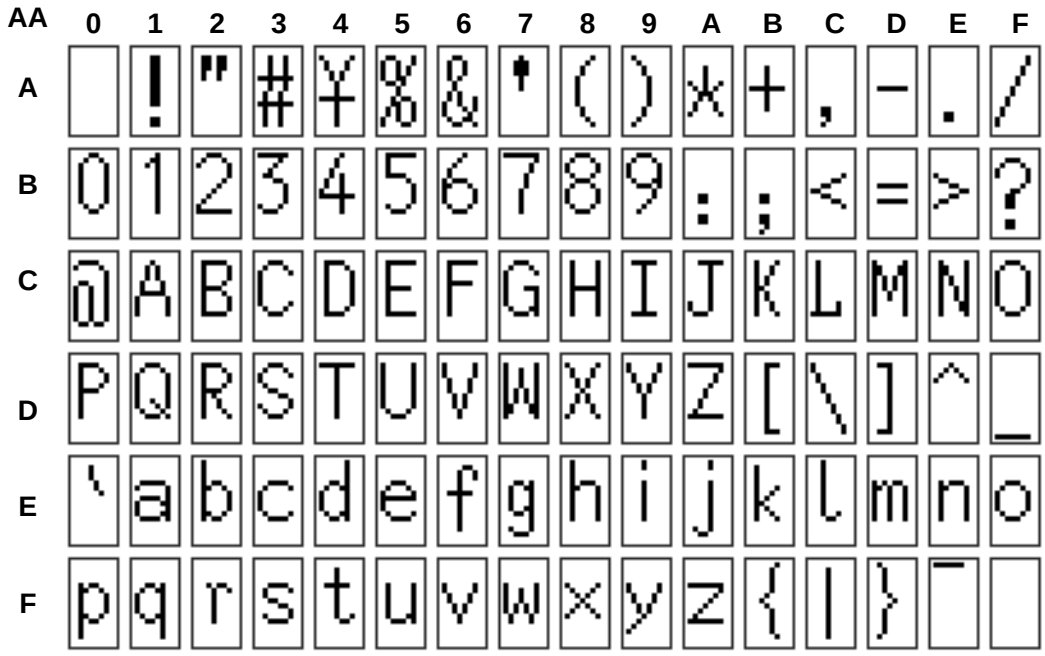
A7	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н
B	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э
C	Ю	Я														
D		а	б	в	г	д	е	ё	ж	з	и	й	к	л	м	н
E	о	п	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э
F	ю	я														

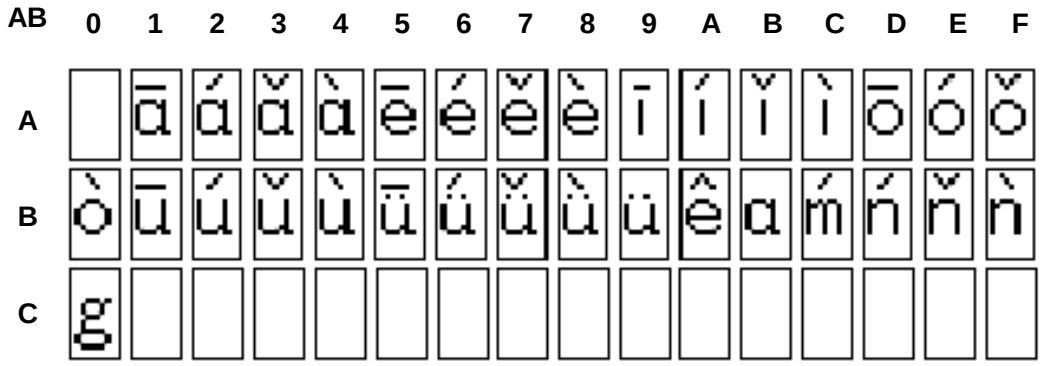
A8	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
A		ā	á	ǎ	à	ē	é	ě	è	ī	í	ǐ	ì	ō	ó	ǒ
B	ò	ū	ú	ǔ	ù	ǖ	ǘ	ǚ	ǜ	ü	ê	ɑ	ḿ	ń	ň	ǹ
C	ɡ					ㄅ	ㄆ	ㄇ	ㄈ	ㄉ	ㄊ	ㄋ	ㄌ	ㄍ	ㄎ	ㄏ
D	ㄐ	ㄑ	ㄒ	ㄓ	ㄔ	ㄕ	ㄖ	ㄗ	ㄘ	ㄙ	ㄚ	ㄛ	ㄜ	ㄝ	ㄞ	ㄟ
E	ㄠ	ㄡ	ㄢ	ㄣ	ㄤ	ㄥ	ㄦ	ㄧ	ㄨ	ㄩ						
F																

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 8×16 National standard extended character（126 character）

Include 126 characters with internal code AAA1~ABC0





8.3 8×16 Special character（64 characters）

Include 64 characters with internal code ACA1~ACDF