

1. 简介

HD35-I2C 是 I2C 接口 3.5 寸 TFT 彩色显示模块，自带 ASCII 字库和中文字库。模块带 64M bit 的内存，可存储 BMP 图片文件，通过 I2C 接口与 MCU 连接，实现彩色图形信息显示。

(1) 集成 4 个 ASCII 字库和 4 个汉字字库，字库类型：

ASCII 字库：8x16, 12x24, 16x32, 32x64

GB2312 中文字库：16x16, 24x24, 32x32, 64x64

(2) 内置内存容量：64Mbit

字库占用 16Mbit，剩余空间 48Mbit 用于存储图片数据。

(3) 集成绘图功能，向 LCDs 简单地发送命令和坐标参数，可实现点，直线，方形，圆形，按键，窗体，编辑框显示。

2. 规格说明

序号	项目	规格标准值	单位	备注
1	显示点阵数	480*RGB*320	Dots	
2	LCD 尺寸	3.5	inch	
3	LCM 外型尺寸		mm	
4	动态显示区	46(L)*72(W)	mm	
5	像素尺寸	0.15(L)*0.15(W)	mm	
6	像素成份	a-Si TFT		
7	LCD 模式	65 k TFT		16 位色彩
8	视角	12 o'clock		
9	背光	白色 led		
10	模块供电	3.3~5	V	

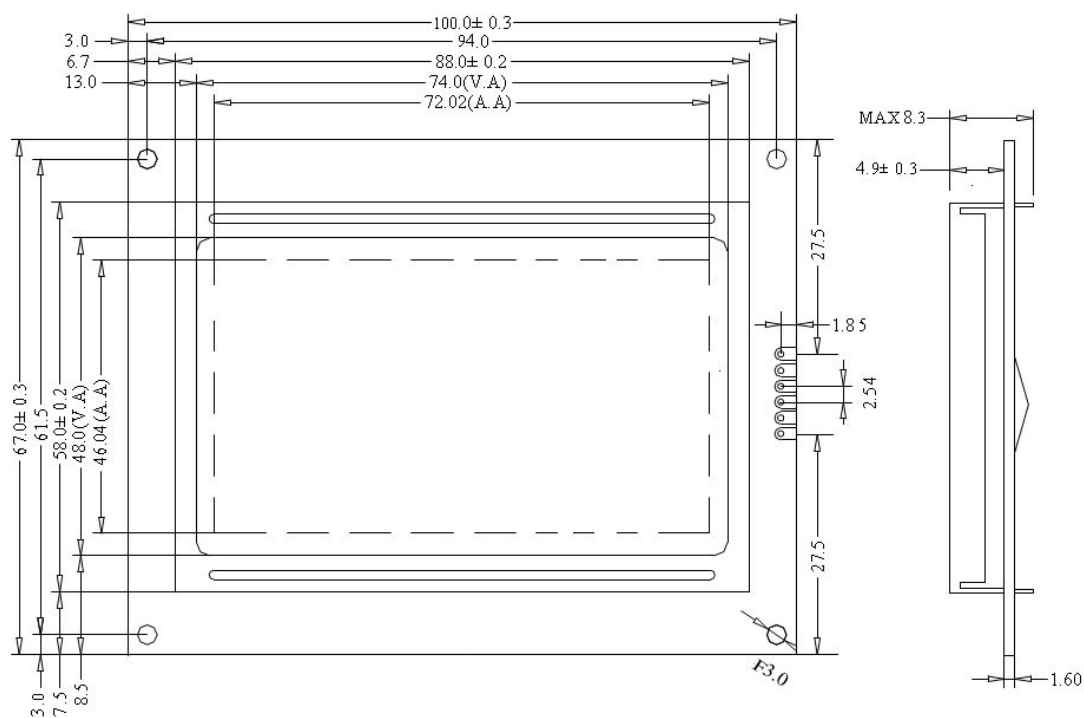
产品最大额定值：

项目	符号	最小值	典型值	最大值	单位	备注
操作温度	Top	-20	-	+70	°C	
存储温度	Tst	-30	-	+80	°C	
供电电压	VDD	-0.3	-	5.2	V	
输入电压	Vin	-0.3	-	VDD+0.3	V	根据输入电压确定

产品电气特性：

项目	符号	最小值	典型值	最大值	单位	备注
工作电压	VDD	3.3	5.0	5.4	V	3.3V/5V 供电
工作电流	I	43	46	50	mA	背光开
	I	20	28	35		背光关

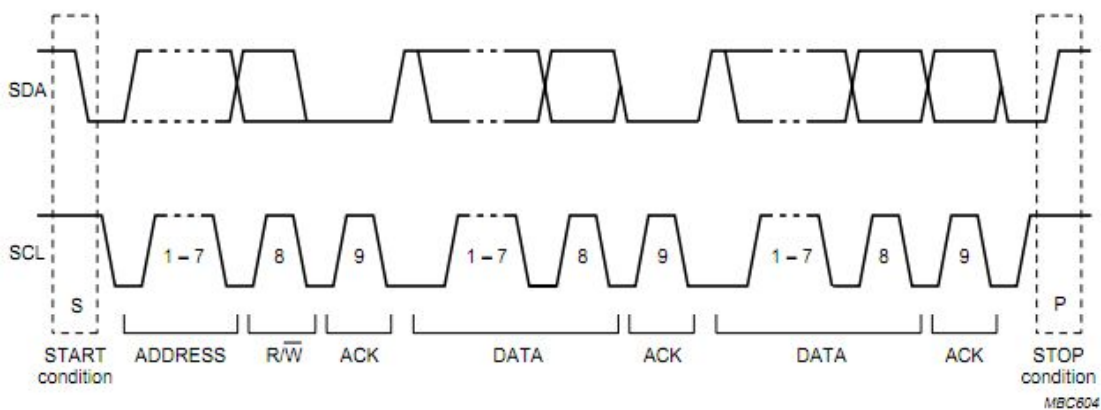
3. 结构图与引脚



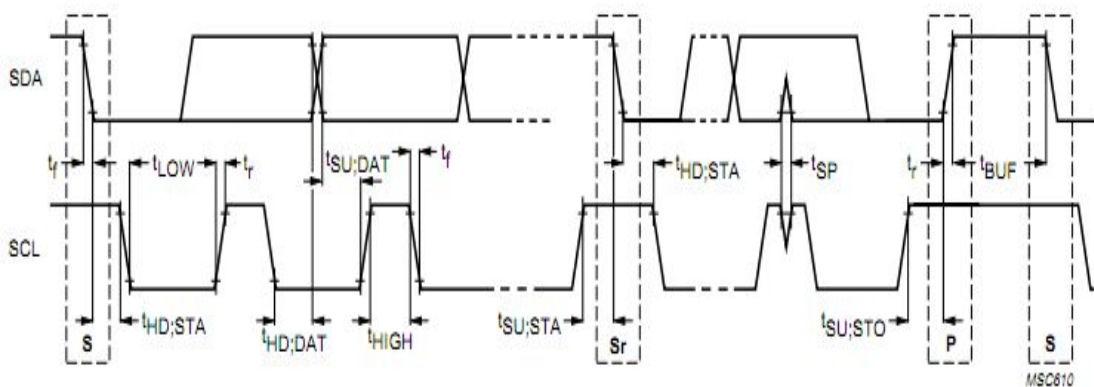
I2C 控制接口说明:

接口序号	接口名称
1	VDD 电源
2	GND 地
3	SDA 串行数据接口
4	SCL 串行时钟接口
5	NC
6	RST 复位控制口

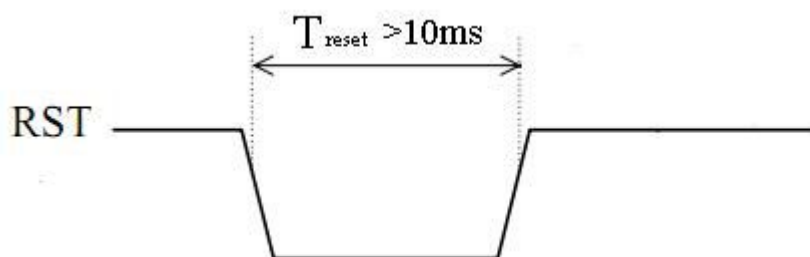
I2C 接口时序图:



A complete data transfer.



RST 复位引脚的时序图:



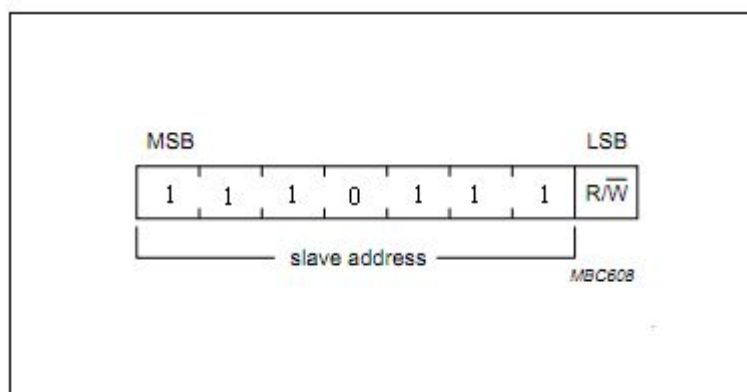
显示模块的复位时间最少为 10ms.

交流参数:

PARAMETER	SYMBOL	STANDARD-MODE		UNIT
		MIN.	MAX.	
SCL clock frequency	f_{SCL}	0	100	kHz
Hold time (repeated) START condition. After this period, the first clock pulse is generated	$t_{HD;STA}$	4.0	–	μs
LOW period of the SCL clock	t_{LOW}	4.7	–	μs
HIGH period of the SCL clock	t_{HIGH}	4.0	–	μs
Set-up time for a repeated START condition	$t_{SU;STA}$	4.7	–	μs
Data hold time: for CBUS compatible masters (see NOTE, Section 10.1.3) for I ² C-bus devices	$t_{HD;DAT}$	5.0 0 ⁽²⁾	– 3.45 ⁽³⁾	μs μs
Data set-up time	$t_{SU;DAT}$	250	–	ns
Rise time of both SDA and SCL signals	t_r	–	1000	ns
Fall time of both SDA and SCL signals	t_f	–	300	ns
Set-up time for STOP condition	$t_{SU;STO}$	4.0	–	μs
Bus free time between a STOP and START condition	t_{BUF}	4.7	–	μs
Capacitive load for each bus line	C_b	–	400	pF
Noise margin at the LOW level for each connected device (including hysteresis)	V_{nL}	$0.1V_{DD}$	–	V
Noise margin at the HIGH level for each connected device (including hysteresis)	V_{nH}	$0.2V_{DD}$	–	V

I2C 地址:

设备地址为: 0x77



图片下载 Uart 接口:



ISP 图片下载接口说明:

接口序号	接口名称
1	VDD 电源
2	TX1 串行发送接口
3	RX1 串行接收接口
4	GND 地

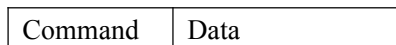
注: ISP 图片下载串口的默认通讯波特率为: 256000 KBPS



4. 控制命令格式和指令集

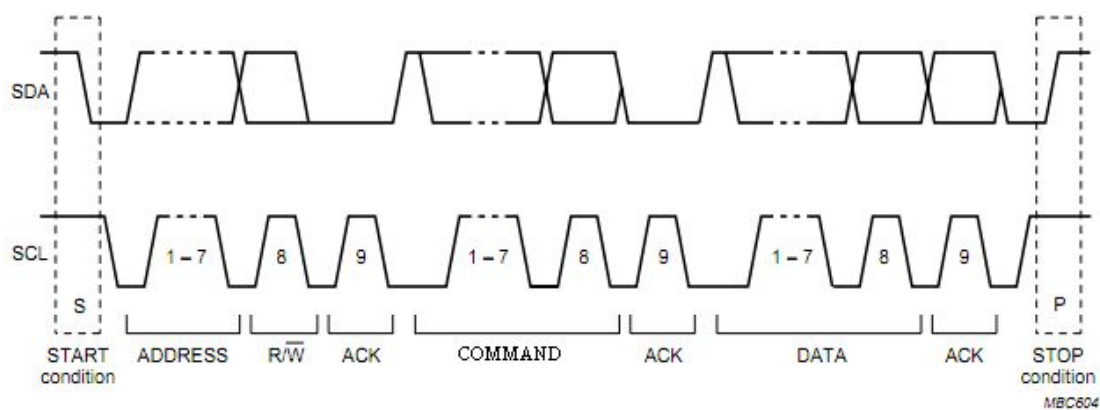
通过 I2C 接口向 LCD 模块发送命令和数据，数据包格式由：
命令 (Command) + 数据 (Data) 组成。

4.1. 控制指令格式



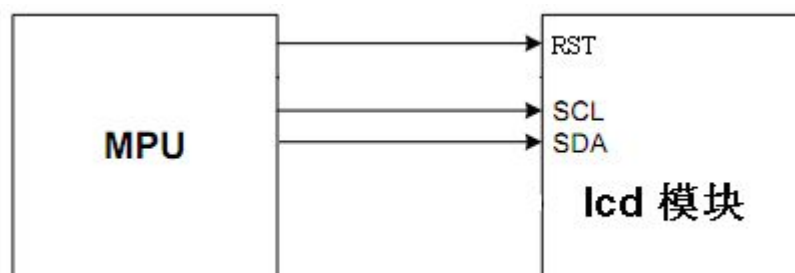
在标准 I2C 的通讯流程，先发地址，再发命令 Command，后发数据 Data。

以下是 I2C 接口数据传送时序图：



A complete data transfer.

与 MCU 控制连接图：



4.2. 控制命令集

功能 function	命令 Command	数据 Data
清整个 显示屏	0x80	NULL
设置背 景色	0x81	Data1:背景色高八位。 Data2:背景色低八位。
设置前 景色	0x82	Data1: 字符颜色高八位。 Data2: 字符颜色低八位。
设置背 光亮度	0x83	Data1: 背光亮度, 范围 0x00~0x09 0x00 关背光 0x01~0x09 九级可调背光
ASCII 字符显 示	0x85	Data1: 字符左上角 X 轴坐标高八位 Data2: 字符左上角 X 轴坐标低八位 Data3: 字符左上角 Y 轴坐标高八位 Data4: 字符左上角 Y 轴坐标低八位 Data5: 高 4 位字符显示方式, 低 4 位字符类型 Data6: ASCII 码
ASCII 字符串 显示	0x86	Data1: 左上角 X 轴坐标高八位 Data2: 左上角 X 轴坐标低八位 Data3: 左上角 Y 轴坐标高八位 Data4: 左上角 Y 轴坐标低八位 Data5: bit7~bit4 字符串长度 bit3 字符显示方式 bit2~bit0 字符类型 Data6~DataN: 要显示的 ASCII 数据
汉字显 示	0x87	Data1: 字符左上角 X 轴坐标高八位 Data2: 字符左上角 X 轴坐标低八位 Data3: 字符左上角 Y 轴坐标高八位 Data4: 字符左上角 Y 轴坐标低八位 Data5: 高 4 位字符显示方式, 低 4 位字符类型 Data6: 二级汉字库 GB 码高八位 Data7: 二级汉字库 GB 码低八位
汉字字 符串显 示	0x88	Data1: 左上角 X 轴坐标高八位 Data2: 左上角 X 轴坐标低八位 Data3: 左上角 Y 轴坐标高八位 Data4: 左上角 Y 轴坐标低八位 Data5: bit7~bit4 字符串长度 bit3 字符显示方式 bit2~bit0 字符尺寸类型 Data6~DataN: 要显示的中文 GB 码数据

		中文 GB 码的顺序按照先高字节后低字节
图片显示	0x89	Data1: 图标左上角 X 轴坐标高八位 Data2: 图标左上角 X 轴坐标低八位 Data3: 图标左上角 Y 轴坐标高八位 Data4: 图标左上角 Y 轴坐标低八位 Data5~ Data8: 图标地址共 4 个字节, 顺序按照先高字节后低字节
画点	0x8A	Data1: 点 X 轴坐标高八位 Data2: 点 X 轴坐标低八位 Data3: 点 Y 轴坐标高八位 Data4: 点 Y 轴坐标低八位
画任意直线	0x8B	Data1: 直线起点 X 轴坐标高八位 Data2: 直线起点 X 轴坐标低八位 Data3: 直线起点 Y 轴坐标高八位 Data4: 直线起点 Y 轴坐标低八位 Data5: 直线终点 X 轴坐标高八位 Data6: 直线终点 X 轴坐标低八位 Data7: 直线终点 Y 轴坐标高八位 Data8: 直线终点 Y 轴坐标低八位
画矩形框	0x8C	Data1: 左上角 X 轴坐标高八位 Data2: 左上角 X 轴坐标低八位 Data3: 左上角 Y 轴坐标高八位 Data4: 左上角 Y 轴坐标低八位 Data5: 矩形长度高八位 Data6: 矩形长度低八位 Data7: 矩形高度高八位 Data8: 矩形高度低八位
画实心矩形	0x8D	Data1: 左上角 X 轴坐标高八位 Data2: 左上角 X 轴坐标低八位 Data3: 左上角 Y 轴坐标高八位 Data4: 左上角 Y 轴坐标低八位 Data5: 矩形长度高八位 Data6: 矩形长度低八位 Data7: 矩形高度高八位 Data8: 矩形高度低八位
画圆	0x8E	Data1: 圆心 X 轴坐标高八位 Data2: 圆心 X 轴坐标低八位 Data3: 圆心 Y 轴坐标高八位 Data4: 圆心 Y 轴坐标低八位 Data5: 圆半径高八位 Data6: 圆半径低八位
画实心	0x8F	Data1: 圆心 X 轴坐标高八位

圆		Data2: 圆心 X 轴坐标低八位 Data3: 圆心 Y 轴坐标高八位 Data4: 圆心 Y 轴坐标低八位 Data5: 圆半径高八位 Data6: 圆半径低八位
画窗体 box	0x9a	Data1: 窗体起点 X 轴坐标高八位 Data2: 窗体起点 X 轴坐标低八位 Data3: 窗体起点 Y 轴坐标高八位 Data4: 窗体起点 Y 轴坐标低八位 Data5: 窗体终点 X 轴坐标高八位 Data6: 窗体终点 X 轴坐标低八位 Data7: 窗体终点 Y 轴坐标高八位 Data8: 窗体终点 Y 轴坐标低八位 Data9: 显示模式
画按钮 button	0x9b	Data1: 按钮起点 X 轴坐标高八位 Data2: 按钮起点 X 轴坐标低八位 Data3: 按钮起点 Y 轴坐标高八位 Data4: 按钮起点 Y 轴坐标低八位 Data5: 按钮终点 X 轴坐标高八位 Data6: 按钮终点 X 轴坐标低八位 Data7: 按钮终点 Y 轴坐标高八位 Data8: 按钮终点 Y 轴坐标低八位 Data9: 显示模式
画 ASCII 字符串 编辑栏 edit	0x9c	Data1: 编辑栏起点 X 轴坐标高八位 Data2: 编辑栏起点 X 轴坐标低八位 Data3: 编辑栏起点 Y 轴坐标高八位 Data4: 编辑栏起点 Y 轴坐标低八位 Data5: 编辑栏终点 X 轴坐标高八位 Data6: 编辑栏终点 X 轴坐标低八位 Data7: 编辑栏终点 Y 轴坐标高八位 Data8: 编辑栏终点 Y 轴坐标低八位 Data9: bit7~bit4 字符串长度 bit2~bit0 字符类型 Data6~DataN: 要显示的 ASCII 数据
画汉字 字符串编辑 栏 edit	0x9d	Data1: 编辑栏起点 X 轴坐标高八位 Data2: 编辑栏起点 X 轴坐标低八位 Data3: 编辑栏起点 Y 轴坐标高八位 Data4: 编辑栏起点 Y 轴坐标低八位 Data5: 编辑栏终点 X 轴坐标高八位 Data6: 编辑栏终点 X 轴坐标低八位 Data7: 编辑栏终点 Y 轴坐标高八位 Data8: 编辑栏终点 Y 轴坐标低八位

		Data5: bit7~bit4 字符串长度 bit2~bit0 字符类型 Data6~DataN: 要显示的汉字数据
十进制 数字显示	0x9e	Data1: 数字左上角 X 轴坐标高八位 Data2: 数字左上角 X 轴坐标低八位 Data3: 数字左上角 Y 轴坐标高八位 Data4: 数字左上角 Y 轴坐标低八位 Data5: 高 4 位数字显示方式，低 4 位数字显示大小 Data6: 要显示十进制数字（范围 0~255）

4.4 控制命令详述

4.4.1 清整个显示屏（0x80）

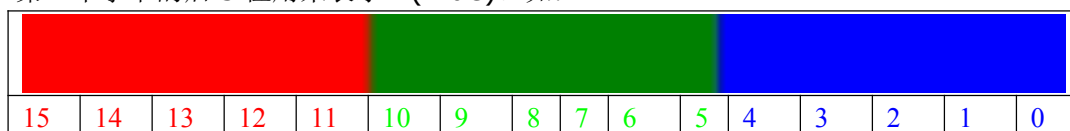
把整个显示屏设为背景色，背景色默认为蓝色。

发送数据：

Command	0x80
Data	NULL

4.4.2 设置背景色（0x81）

设置在 LCD 上显示的背景色，主要与清整个显示屏命令配合使用，默认颜色为蓝色。本 LCD 模块采用 RGB565 彩色模式，一个像素占两个字节，其中：第一个字节的前 5 位用来表示 R(Red)，第一个字节的后三位+第二个字节的前三位用来表示 G(Green)，第二个字节的后 5 位用来表示 B(Blue)。如：



常用颜色的 RGB565 数值：

颜色	RGB565 值
红	0xF800
蓝	0x001F
绿	0x07E0
白	0xFFFF
黑	0x0000

发送数据：

Command	0x81
Data	ColorH ColorL

注：ColorH 背景颜色高 8 位

ColorL 背景颜色低 8 位

4.4.3 设置前景颜色（0x82）

设置在 LCD 上显示的字符和图形颜色

发送数据:

Command	0x82
Data	ColorH ColorL

注: ColorH 前景颜色高 8 位
ColorL 前景颜色低 8 位

4.4.4 设置背光亮度 (0x83)

设置 LCD 模块背光亮度。

发送数据:

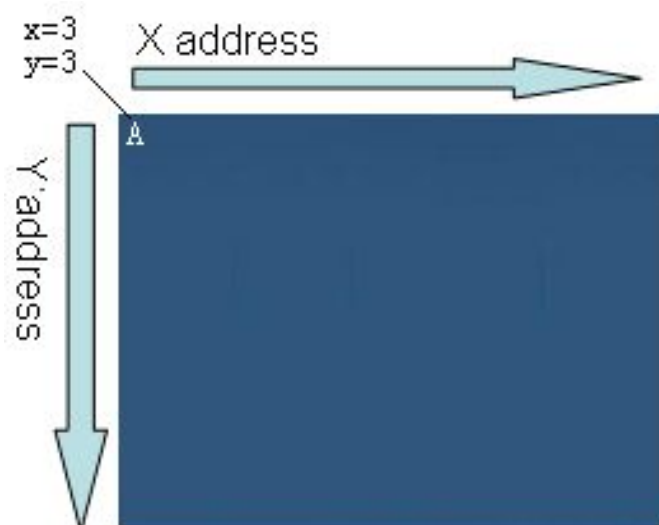
Command	0x83
Data	BLValue

注: BLValue 背光亮度设置值, 范围从 1~9。

亮度	数值
最暗	1
可调范围	1~9
最高	9
关闭背光	0x00

4.4.5 ASCII 字符显示 (0x85)

在屏幕上显示 ASCII 字符, 先是指定要显示的 X 轴坐标和 Y 轴坐标, 然后发送要显示字符的 ASCII 码。如下图示, 要显示的字符坐标是, X=3, Y=3。要显示字符是 'A'。



发送数据:

Command	0x85
---------	------

Data	XH XL YH YL FT ASCII
------	----------------------

注:

FT:

高 4 位: 0xf 透明写字符 0x0 带背景写字符

低 4 位: ASCII 显示字库类型

字库尺寸	数值
8x16	1
12x24	2
16x32	3
32x64	0

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

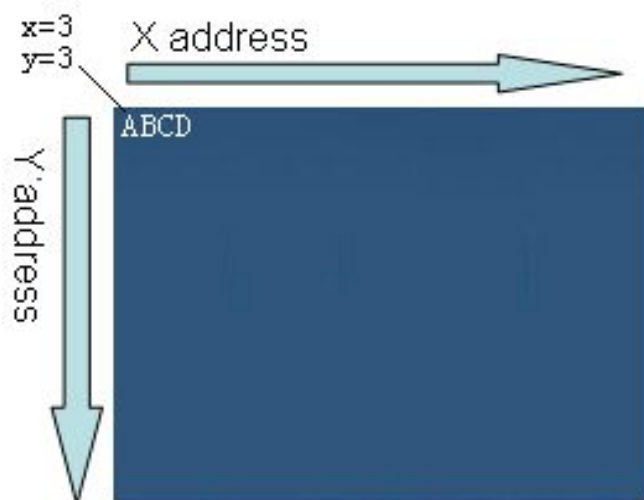
YL: Y 轴地址低 8 位

ASCII: ASCII 码

4.4.6 ASCII 字符串显示 (0x86)

在屏幕上显示 ASCII 字符串, 先是指定要显示的 X 轴坐标和 Y 轴坐标, 然后发送要显示字符串的 ASCII 码。

如下图示, 要显示的字符坐标是, X=3, Y=3。要显示字符是串 'ABCD'。



发送数据:

Command	0x86
Data	XH XL YH YL FT ASCIIString

注:

FT:

bit7~bit4 字符串长度

bit3 字符显示方式 (1 透明写字符 0 带背景写字符)

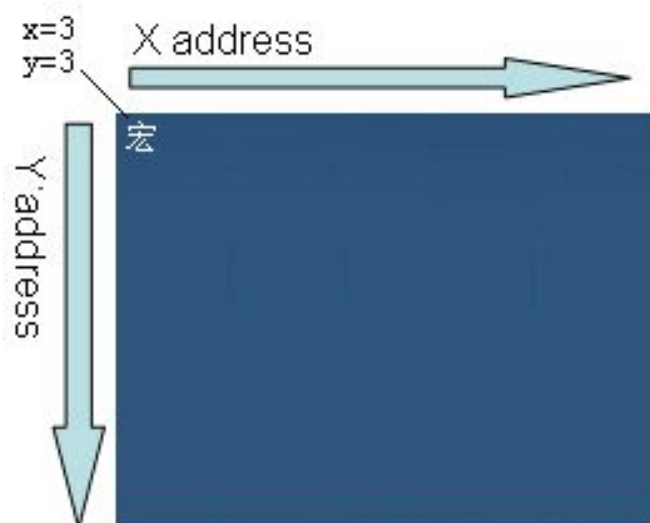
bit2~bit0 字库尺寸类型

字库尺寸	数值
8x16	1
12x24	2
16x32	3
32x64	0

ASCIIString 最长为 15 个字符。

4.4.7 汉字显示 (0x87)

在屏幕上显示汉字字符，先是指定要显示的 X 轴坐标和 Y 轴坐标，然后发送要显示汉字的区位码。如下图所示，要显示的字符坐标是，X=3，Y=3。要显示字符是‘宏’。



发送数据：

Command	0x87
Data	XH XL YH YL FT HZH HZL

注：

FT:

高 4 位：0xf 透明写字符 0x0 带背景写字符

低 4 位：汉字显示字库类型

字库尺寸	数值
16x16	4
24x24	5
32x32	6
64x64	7

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

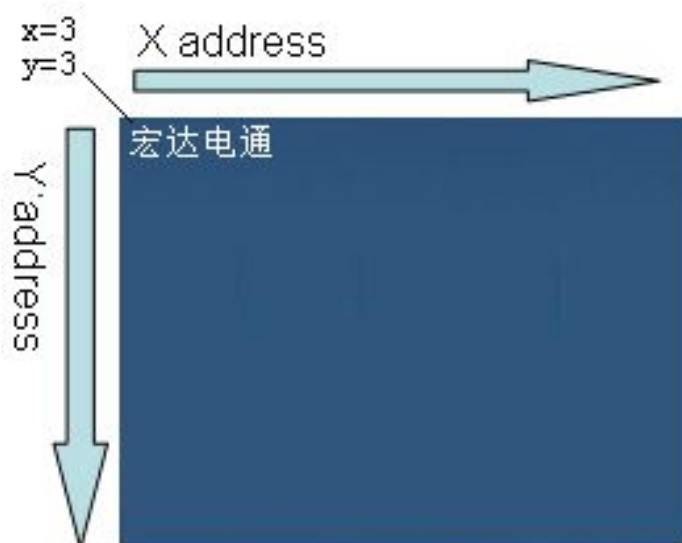
YL: Y 轴地址低 8 位

HZH: 汉字区位码高 8 位

HZL: 汉字区位码低 8 位

4.4.8 汉字字符串显示 (0x88)

在屏幕上显示汉字字符串，先是指定要显示的 X 轴坐标和 Y 轴坐标，然后发送要显示汉字串的区位码。如下图示，要显示的字符坐标是，X=3，Y=3。要显示字符是‘宏达电通’。



发送数据：

Command	0x88
Data	XH XL YH YL FT HZString

注：

FT:

bit7~bit4 字符串长度

bit3 字符显示方式 (1 透明写字符 0 带背景写字符)

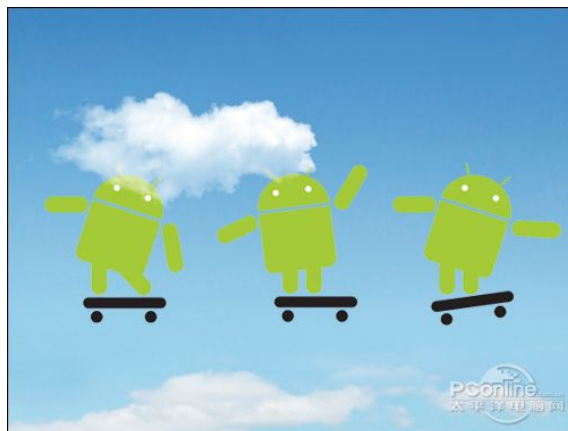
bit2~bit0 字库尺寸类型

字库尺寸	数值
16x16	4
24x24	5
32x32	6
64x64	7

HZString 最长为 12 个汉字 = 24 个 byte

4.4.9 图片显示 (0x89)

在屏幕上显示图标，先是指定要显示的 X 轴坐标和 Y 轴坐标，然后发送要显示图标在 flash 的存储地址。如下图所示，要显示的字符坐标是，X=0，Y=0。



发送数据：

Command	0x89
Data	XH XL YH YL Icon3 Icon2 Icon1 Icon0

注：

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

YL: Y 轴地址低 8 位

Icon3: icon 地址 24~31 位

Icon2: icon 地址 16~23 位

Icon1: icon 地址 8~15 位

Icon0: icon 地址 0~7 位

4.4.10 画点 (0x8A)

在 LCD 任意区域根据前景色画一点

发送数据：

Command	0x8A
Data	XH XL YH YL

注：

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

YL: Y 轴地址低 8 位

4.4.11 画任意线 (0x8B)

在 LCD 任意区域画线

发送数据：

Command	0x8B
Data	XSH XSL YSH YSL XEH XEL YEH YEL

注：

XSH: 直线 X 轴起始地址高 8 位

XSL: 直线 X 轴起始地址低 8 位

YSH: 直线 Y 轴起始地址高 8 位

YSL: 直线 Y 轴起始地址低 8 位

XEH: 直线 X 轴结束地址高 8 位

XEL: 直线 X 轴结束地址低 8 位

YEH: 直线 Y 轴结束地址高 8 位

YEL: 直线 Y 轴结束地址低 8 位

4.4.12 画矩形框 (0x8C)

在 LCD 任意区域画矩形框

发送数据：

Command	0x8C
Data	XH XL YH YL WH WL HH HL

注：

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

YL: Y 轴地址低 8 位

WH: 矩形框宽度高 8 位

WL: 矩形框宽度低 8 位

HH: 矩形框高度高 8 位

HL: 矩形框高度低 8 位

4.4.13 画实心矩形 (0x8D)

在 LCD 任意区域画矩形框

发送数据：

Command	0x8D
Data	XH XL YH YL WH WL HH HL

注：

XH: X 轴地址高 8 位

XL: X 轴地址低 8 位

YH: Y 轴地址高 8 位

YL: Y 轴地址低 8 位

WH: 矩形宽度高 8 位

WL: 矩形宽度低 8 位

HH: 矩形高度高 8 位

HL: 矩形高度低 8 位

4.4.14 画圆形 (0x8E)

在 LCD 任意区域画圆形框

发送数据:

Command	0x8E
Data	XH XL YH YL RH RL

注:

XH: 圆心 X 轴地址高 8 位

XL: 圆心 X 轴地址低 8 位

YH: 圆心 Y 轴地址高 8 位

YL: 圆心 Y 轴地址低 8 位

RH: 圆形半径高 8 位

RL: 圆形半径低 8 位

4.4.15 画实心圆形 (0x8F)

在 LCD 任意区域画实心圆形

发送数据:

Command	0x8F
Data	XH XL YH YL RH RL

注:

XH: 圆心 X 轴地址高 8 位

XL: 圆心 X 轴地址低 8 位

YH: 圆心 Y 轴地址高 8 位

YL: 圆心 Y 轴地址低 8 位

RH: 实心圆形半径高 8 位

RL: 实心圆形半径低 8 位

4.4.16 画窗体 box (0x9A)

在 LCD 任意区域画窗体

发送数据:

Command	0x9A
Data	XSH XSL YSH YSL XEH XEL YEH YEL MODE

注:

XSH: 窗体 X 轴起始地址高 8 位

XSL: 窗体 X 轴起始地址低 8 位

YSH: 窗体 Y 轴起始地址高 8 位

YSL: 窗体 Y 轴起始地址低 8 位

XEH: 窗体 X 轴结束地址高 8 位

XEL: 窗体 X 轴结束地址低 8 位

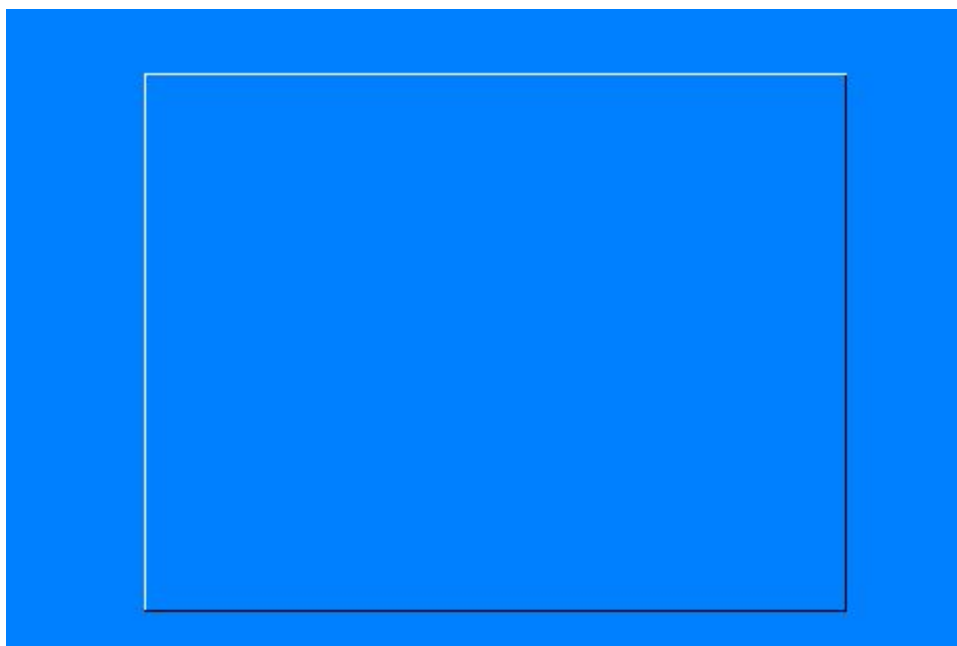
YEH: 窗体 Y 轴结束地址高 8 位

YEL: 窗体 Y 轴结束地址低 8 位

MODE:

0x00 窗体上凸

0xff 窗体下凹



4.4.17 画按键 button (0x9B)

在 LCD 任意区域画按键

发送数据:

Command	0x9B
Data	XSH XSL YSH YSL XEH XEL YEH YEL MODE

注:

XSH: 按键 X 轴起始地址高 8 位

XSL: 按键 X 轴起始地址低 8 位

YSH: 按键 Y 轴起始地址高 8 位

YSL: 按键 Y 轴起始地址低 8 位

XEH: 按键 X 轴结束地址高 8 位

XEL: 按键 X 轴结束地址低 8 位

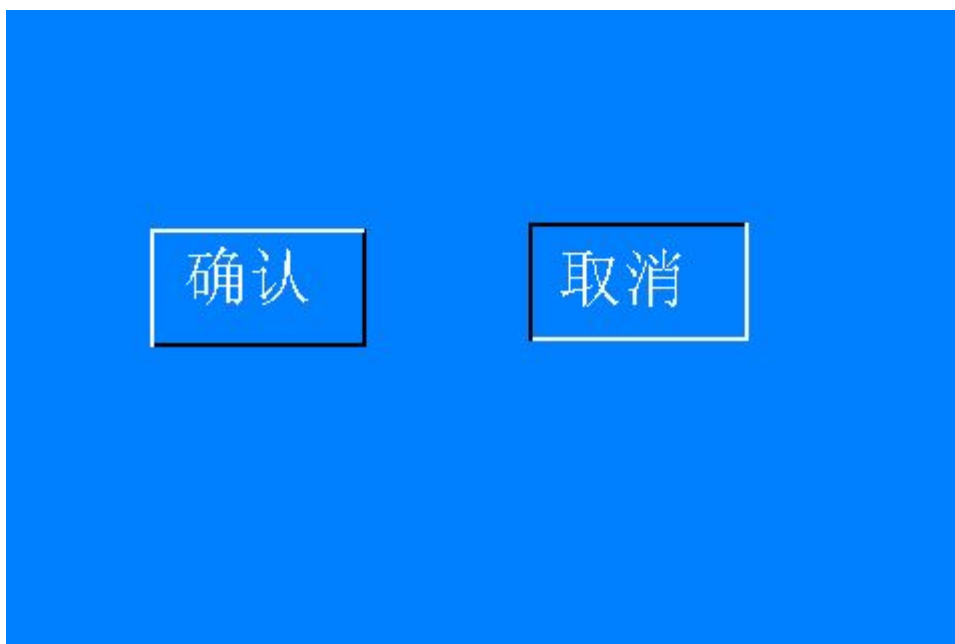
YEH: 按键 Y 轴结束地址高 8 位

YEL: 按键 Y 轴结束地址低 8 位

MODE:

0x00 按键上凸

0xff 按键下凹



4.4.18 画 ASCII 字符显示编辑栏 edit (0x9C)

在 LCD 任意区域画编辑栏

发送数据:

Command	0x9C
Data	XSH XSL YSH YSL XEH XEL YEH YEL FT ASCIIString

注:

XSH: 编辑栏 X 轴起始地址高 8 位

XSL: 编辑栏 X 轴起始地址低 8 位

YSH: 编辑栏 Y 轴起始地址高 8 位

YSL: 编辑栏 Y 轴起始地址低 8 位

XEH: 编辑栏 X 轴结束地址高 8 位

XEL: 编辑栏 X 轴结束地址低 8 位

YEH: 编辑栏 Y 轴结束地址高 8 位

YEL: 编辑栏键 Y 轴结束地址低 8 位

FT:

bit7~bit4 字符串长度

bit2~bit0 字库尺寸类型

字库尺寸	数值
8x16	1
12x24	2
16x32	3
32x64	0



4.4.19 画汉字字符显示编辑栏 edit (0x9D)

在 LCD 任意区域画编辑栏

发送数据:

Command	0x9D
Data	XSH XSL YSH YSL XEH XEL YEH YEL FT HZString

注:

XSH: 编辑栏 X 轴起始地址高 8 位

XSL: 编辑栏 X 轴起始地址低 8 位

YSH: 编辑栏 Y 轴起始地址高 8 位

YSL: 编辑栏 Y 轴起始地址低 8 位

XEH: 编辑栏 X 轴结束地址高 8 位

XEL: 编辑栏 X 轴结束地址低 8 位

YEH: 编辑栏 Y 轴结束地址高 8 位

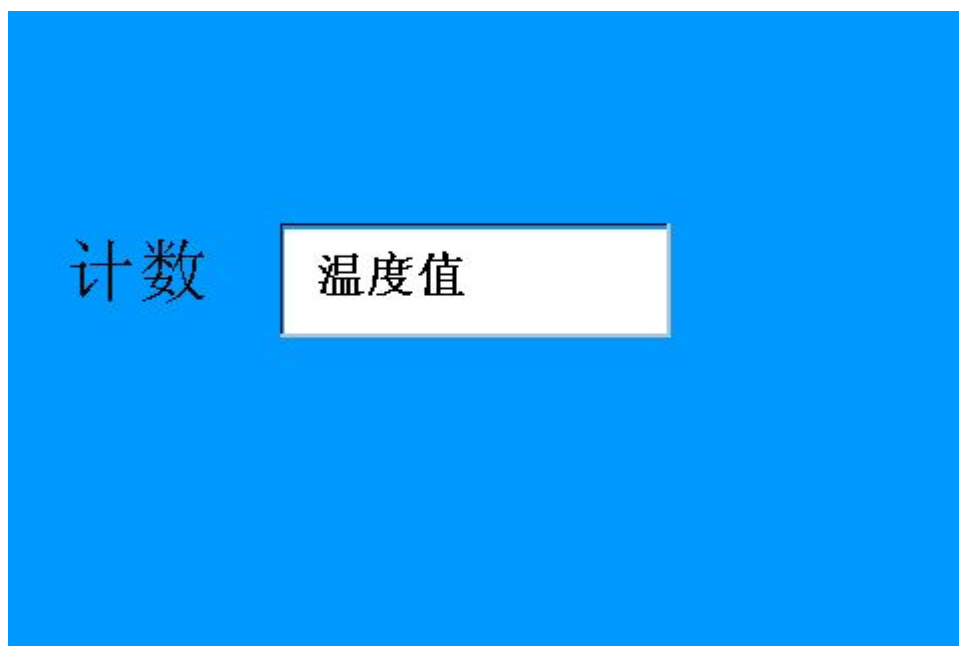
YEL: 编辑栏键 Y 轴结束地址低 8 位

FT:

bit7~bit4 汉字串长度, 汉字长度为汉字个数, 最长为 9 个汉字

bit2~bit0 字库尺寸类型

字库尺寸	数值
16x16	4
24x24	5
32x32	6
64x64	7



4.4.20 十进制数字显示（0x9E）

在屏幕上指定 X 轴坐标和 Y 轴坐标显示十进制数字

发送数据：

Command	0x9E
Data	XH XL YH YL FT Number

注：

FT：

高 4 位：0xf 透明写字符 0x0 带背景写字符

低 4 位：数字显示字库类型

字库尺寸	数值
8x16	1
12x24	2
16x32	3
32x64	0

XH：X 轴地址高 8 位

XL：X 轴地址低 8 位

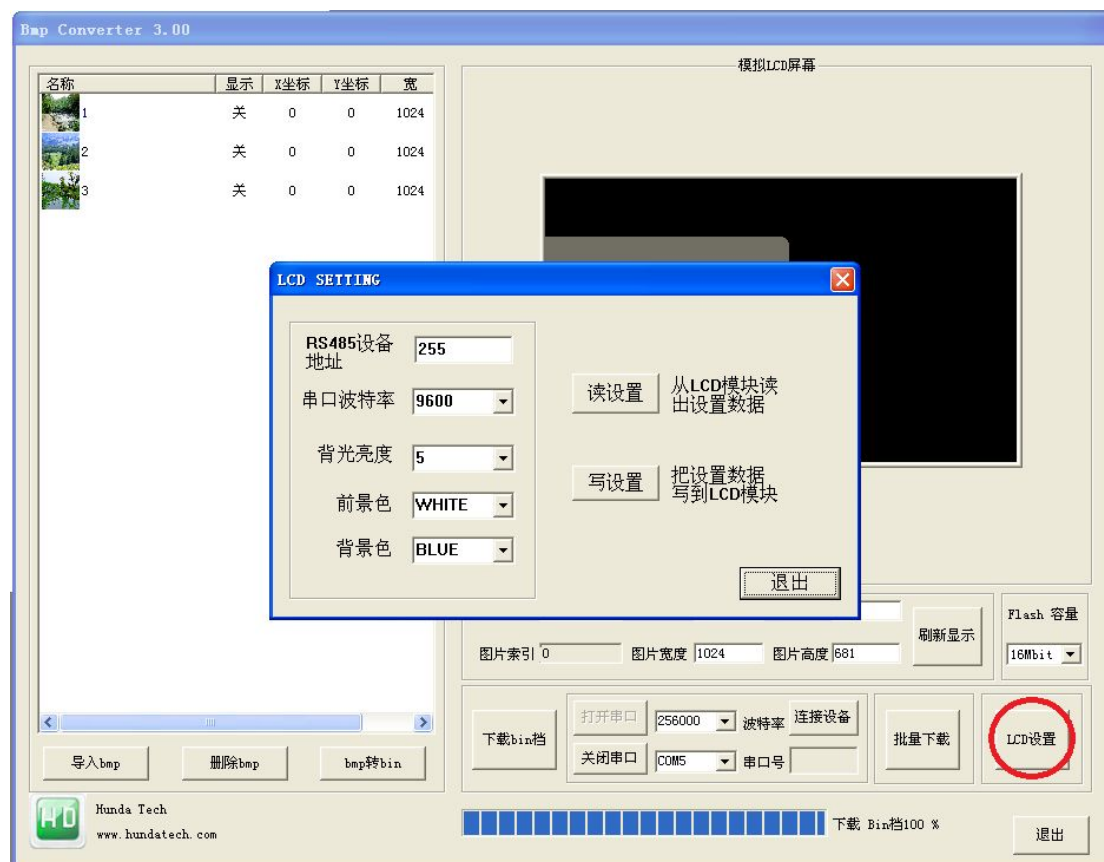
YH：Y 轴地址高 8 位

YL：Y 轴地址低 8 位

Number：十进制数字（范围 0~255）

5.修改液晶显示模块默认设置

5.1 使用 BmpConverter 通过串口链接液晶模块



设置说明:

- (1)RS485 设备地址（范围 0~255，在 RS485 网络里面，设备地址不能相同）
- (2)串口波特率（默认波特率 9600）
- (3)背光亮度（范围 0~9，0 为关背光，9 亮度最高。默认值为 5）
- (4)前景颜色（默认为白色）
- (5)背景颜色（默认为蓝色）

*注意事项：前景颜色和背景颜色不能相同，会导致分不出显示。

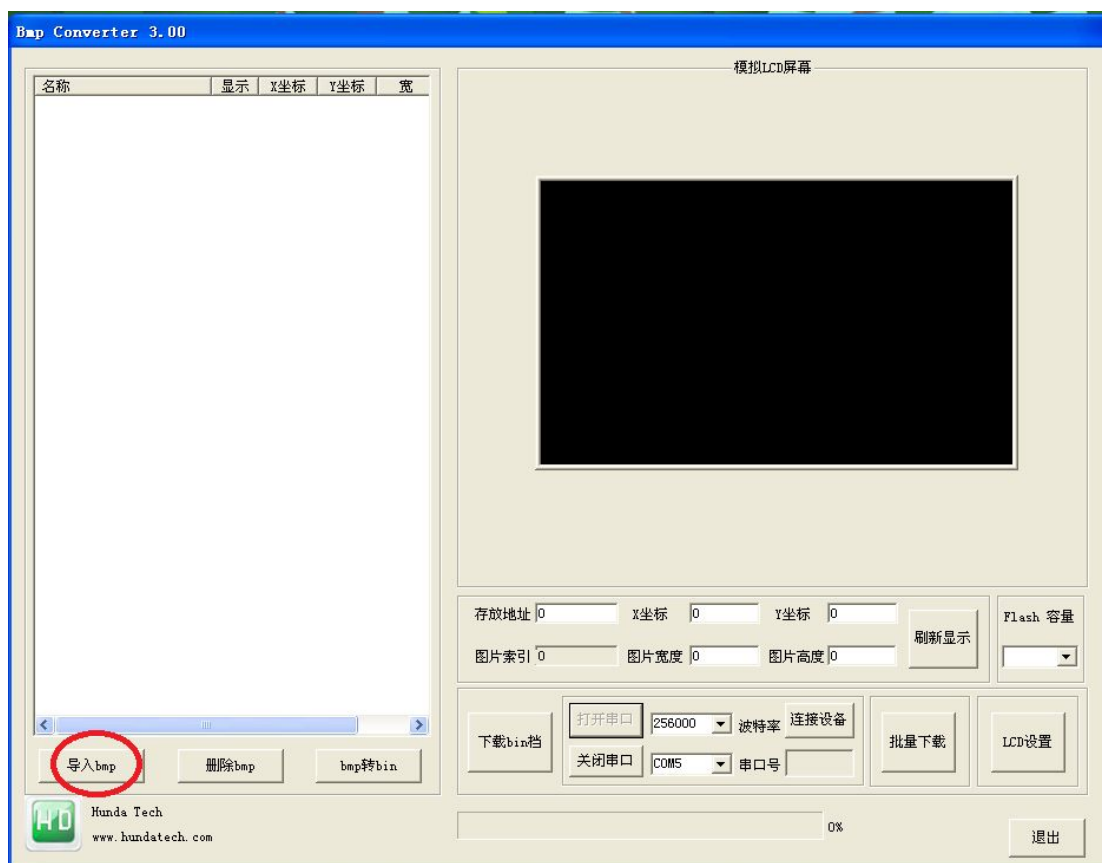
修改默认设置后按“写设置”按键，把默认设置写到液晶模块。

6. 使用 BmpConverter 做 UI 设计和图片下载:

6.1 开始选择 LCD 尺寸:

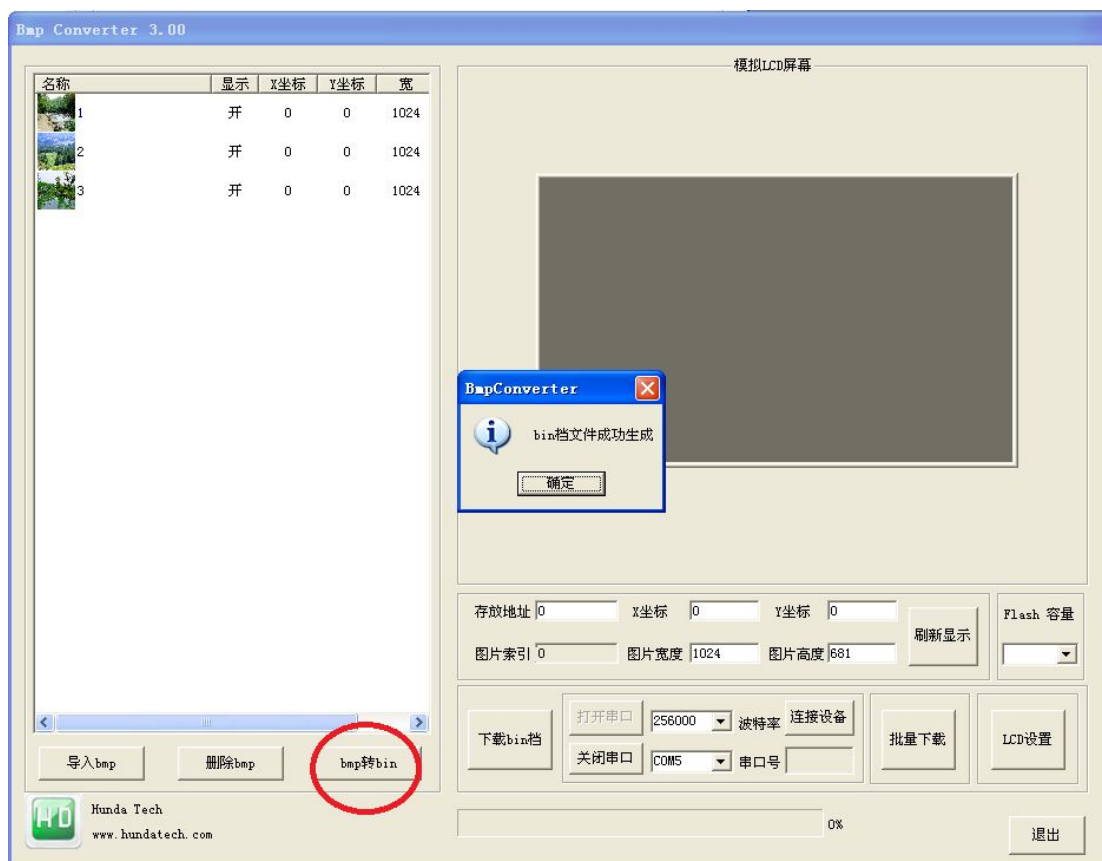
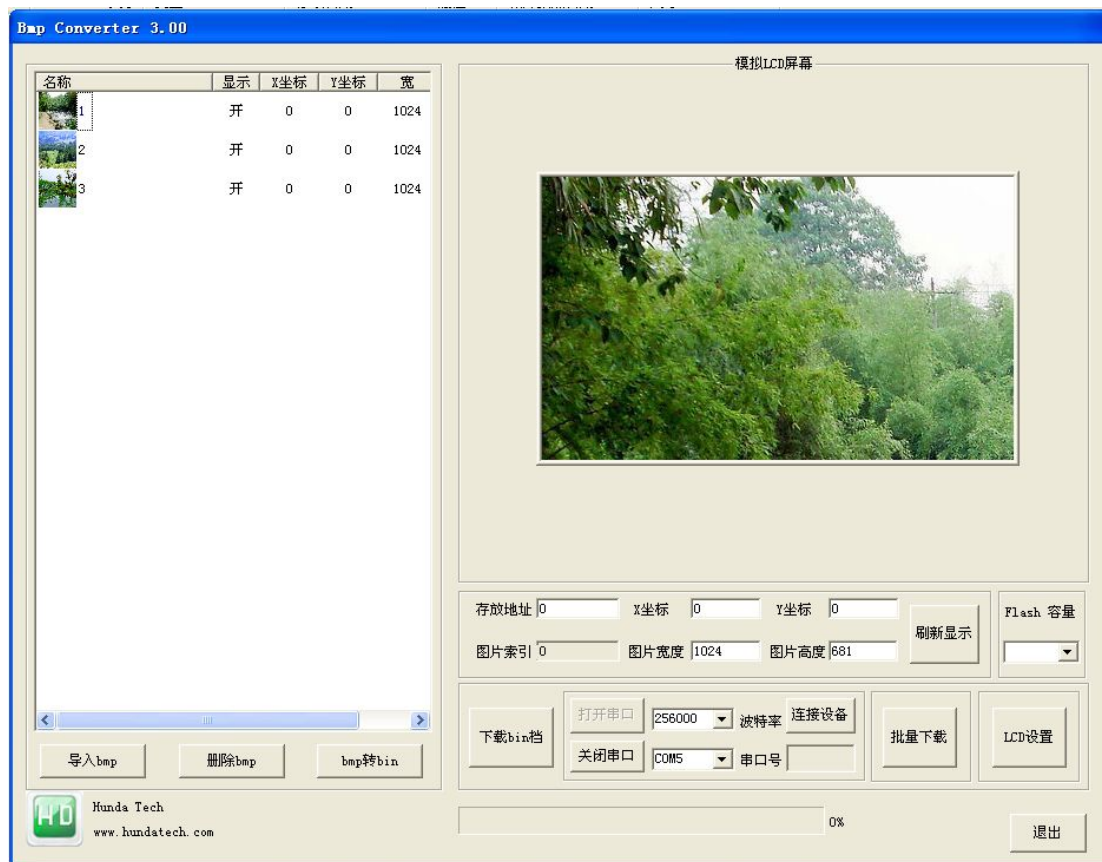


6.2.按导入 bmp,导入 bmp 图片:

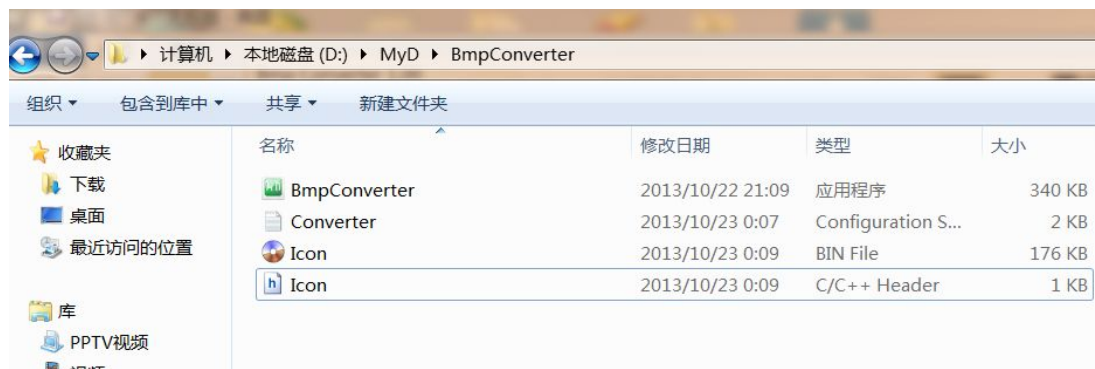


我们可以通过选择左边的图片后，在模拟 lcd 屏里拖动选择的图片，记录对应图片的存放位置和图片长宽。

6.3.按 bmp 转 bin 按键，把图片转成 bin 档和.h 文件：



6.4.生成 bin 档: Icon.bin 和 Icon.h:



Icon.bin 是相应的图片数据。

Icon.h 里记录 bmp 存储地址，通过该地址让 bmp 在 LCD 液晶模块上显示图片：

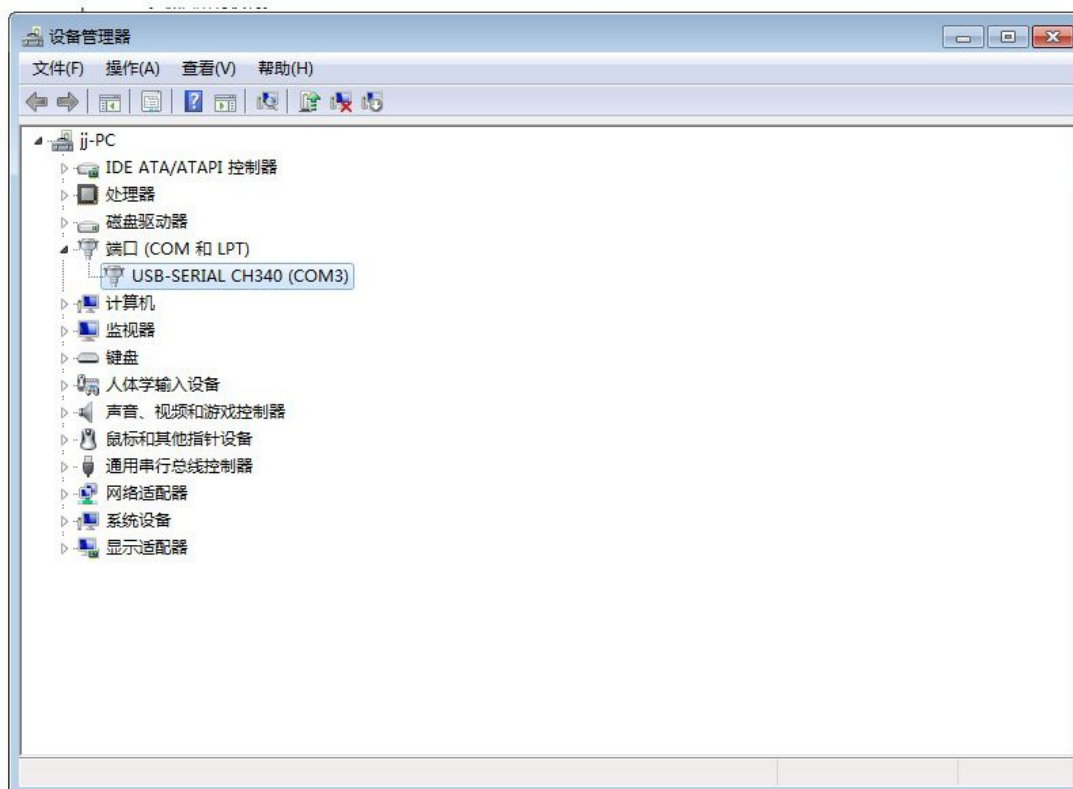
```

Icon.h
// Model Name:COLOR_RGB
// www.hundatech.com
// Generation Time: 2013/10/23 AM 12:09:38

#define IM_BLUE_SELECT 0x0
#define IM_DIALOG_BUTTON_SEL 0x12
#define IM_FIRST_ANIMATION 0x1a
#define IM_GREEN_SELECT 0x147
#define IM_WARNING_POPWIN_BUTTON 0x159

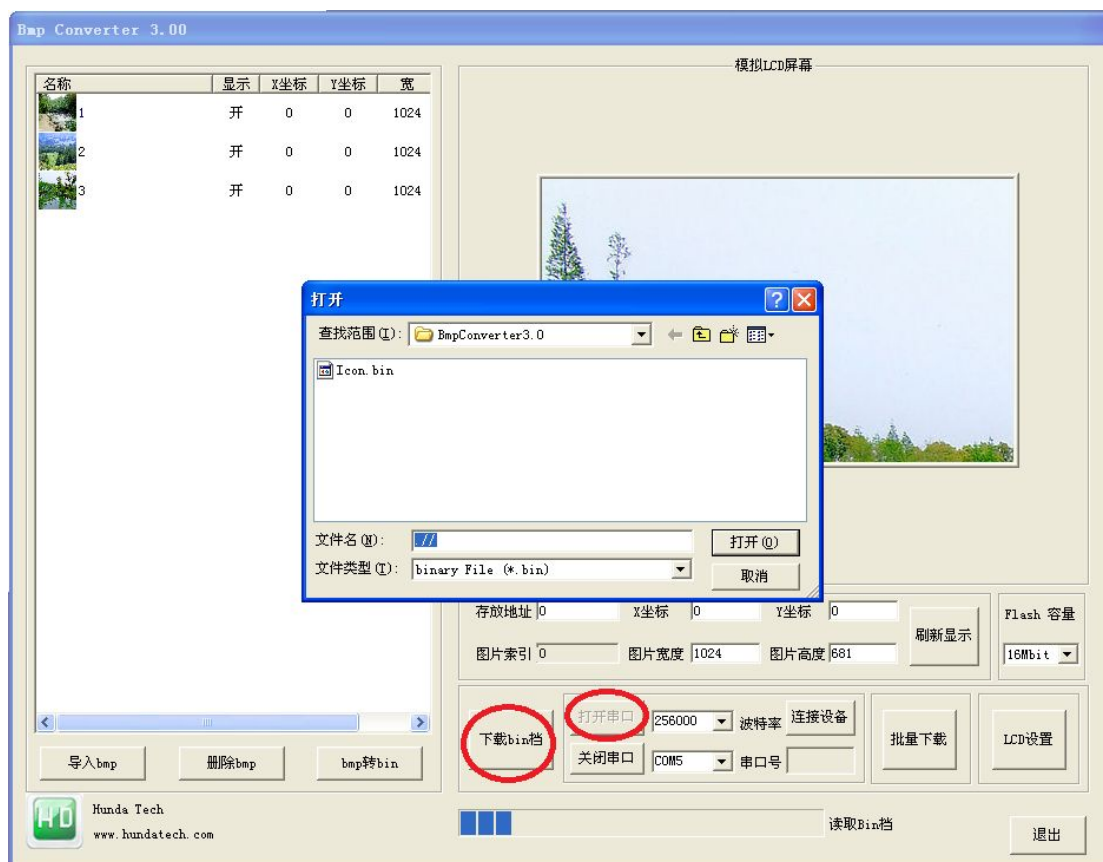
```

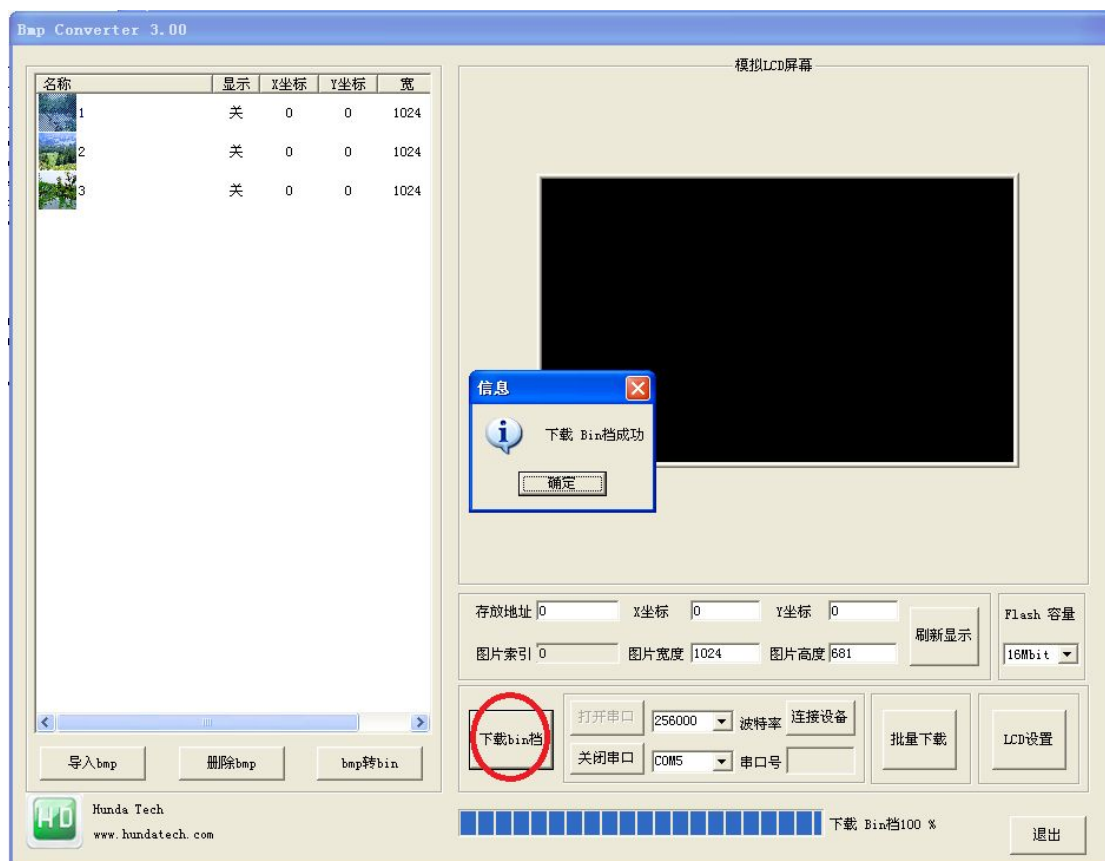
6.5 通过 USB 下载线，LCD 液晶模块链接 PC 后记录对应的串口号：



按打开串口按键，链接 LCD 液晶模块：

6.6. 按下载 bin 档按键，下载 bin 档到 LCD 液晶模块的内存中。





注意事项:

ISP 图片下载串口的默认通讯波特率为: 256000 KBPS

7.开发流程

7.1 整理素材

BmpConverter 只支持 BMP 图片。

7.2 使用 BmpConverter 把 BMP 图片转为 BIN 档。

7.3 使用串口把图片 BIN 下载到显示模块的 flash 中。

7.4 上电后就可与单片机通讯，并显示彩色图片。



8.LCDs 模块的测试

8.1 使用 SPI 测试板测试

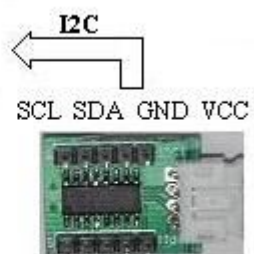


8.2 测试板连接方式

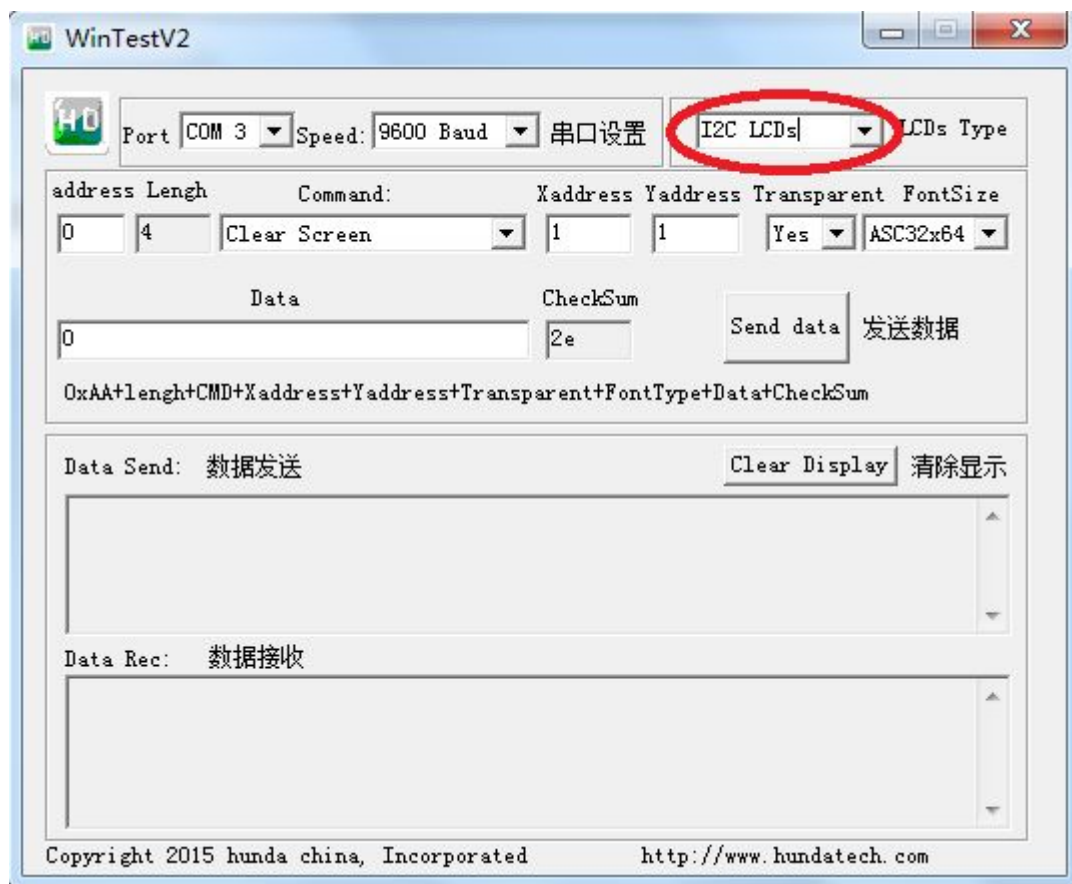
把测试小板直接连到液晶

测试小板的连接标注:

1	VCC
2	GND
3	SDA
4	SCL
5	NC



8.3 使用 Wintest 软件进行测试



注意事项:

1. 串口波特率默认为 9600
2. LCDs Type 选择为 I2C LCDs

附录部分

附录 1: ASCII 码表

H/L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
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	{		}	~	Δ

16*8 半宽字符表

附录 2：汉字码址表

GB 2312 中文字型码表

Table listing characters and their codes for GB 2312. Rows include A1A0 to B0C0. Columns show character sets like numbers, punctuation, and basic Chinese characters.

Table listing characters and their codes for GB 2312, rows B0D0 to B9F0. Contains a wide variety of Chinese characters including rare and complex ones.

BAC0 豪盒亨吼
 BADO 毫貉横厚
 BAE0 郝河恒后
 BAFO 耗涸轰呼
 BBA0 号赫哄乎
 BBEO 浩揭烘忽
 BBFO 呵鹤虹瓠
 BC00 喝贺洪胡
 BC10 荷嘿宏胡
 BC20 核痕弘胡
 BC30 禾很红狐
 BC40 和很喉糊
 BC50 何恨候湖
 BC60 合吟猗
 BC70 化蒙幌贻
 BC80 伏集伎榭
 BC90 奸险蓂
 BD00 蒋娇较竭
 BD10 斤警究鞠
 BD20 具揪巨绢
 BD30 峻慨烤课
 BD40 哭傀唢唢
 BD50 唢唢唢唢
 BD60 唢唢唢唢
 BD70 唢唢唢唢
 BD80 唢唢唢唢
 BD90 唢唢唢唢
 CE00 唢唢唢唢
 CE10 唢唢唢唢
 CE20 唢唢唢唢
 CE30 唢唢唢唢
 CE40 唢唢唢唢
 CE50 唢唢唢唢
 CE60 唢唢唢唢
 CE70 唢唢唢唢
 CE80 唢唢唢唢
 CE90 唢唢唢唢
 CF00 唢唢唢唢
 CF10 唢唢唢唢
 CF20 唢唢唢唢
 CF30 唢唢唢唢
 CF40 唢唢唢唢
 CF50 唢唢唢唢
 CF60 唢唢唢唢
 CF70 唢唢唢唢
 CF80 唢唢唢唢
 CF90 唢唢唢唢
 D000 唢唢唢唢
 D010 唢唢唢唢
 D020 唢唢唢唢
 D030 唢唢唢唢
 D040 唢唢唢唢
 D050 唢唢唢唢
 D060 唢唢唢唢
 D070 唢唢唢唢
 D080 唢唢唢唢
 D090 唢唢唢唢
 D100 唢唢唢唢
 D110 唢唢唢唢
 D120 唢唢唢唢
 D130 唢唢唢唢
 D140 唢唢唢唢
 D150 唢唢唢唢
 D160 唢唢唢唢
 D170 唢唢唢唢
 D180 唢唢唢唢
 D190 唢唢唢唢
 D200 唢唢唢唢
 D210 唢唢唢唢
 D220 唢唢唢唢
 D230 唢唢唢唢
 D240 唢唢唢唢
 D250 唢唢唢唢
 D260 唢唢唢唢
 D270 唢唢唢唢
 D280 唢唢唢唢
 D290 唢唢唢唢
 D300 唢唢唢唢
 D310 唢唢唢唢
 D320 唢唢唢唢
 D330 唢唢唢唢
 D340 唢唢唢唢
 D350 唢唢唢唢
 D360 唢唢唢唢
 D370 唢唢唢唢
 D380 唢唢唢唢
 D390 唢唢唢唢
 D400 唢唢唢唢

C4B0 目南倪娘
 C4C0 木奈霓念
 C4D0 慕耐妮捻
 C4E0 慕奶能撵
 C5A0 慕乃嫩碾
 C5B0 慕叙内年
 C5C0 慕纳馁拈
 C5D0 母娜呢萼
 C5E0 姆那冲溜
 C6A0 山纳闹逆
 C6B0 牡呐恼腻
 C6C0 拘哪恼匿
 C6D0 某拿挠你
 C6E0 牟穆囊拟
 C7A0 车穆囊拟
 C7B0 陌陌男泥
 C7C0 陌陌男泥
 C7D0 陌陌男泥
 C7E0 陌陌男泥
 C8A0 陌陌男泥
 C8B0 陌陌男泥
 C8C0 陌陌男泥
 C8D0 陌陌男泥
 C8E0 陌陌男泥
 C8F0 陌陌男泥
 C9A0 陌陌男泥
 C9B0 陌陌男泥
 C9C0 陌陌男泥
 C9D0 陌陌男泥
 C9E0 陌陌男泥
 C9F0 陌陌男泥
 CAA0 陌陌男泥
 CAB0 陌陌男泥
 CAC0 陌陌男泥
 CAD0 陌陌男泥
 CAE0 陌陌男泥
 CAF0 陌陌男泥
 CBA0 陌陌男泥
 CBB0 陌陌男泥
 CBC0 陌陌男泥
 CBD0 陌陌男泥
 CBE0 陌陌男泥
 CBF0 陌陌男泥
 CCA0 陌陌男泥
 CCB0 陌陌男泥
 CCC0 陌陌男泥
 CCD0 陌陌男泥
 CCE0 陌陌男泥
 CCF0 陌陌男泥
 CDA0 陌陌男泥
 CDB0 陌陌男泥
 CDC0 陌陌男泥
 CDD0 陌陌男泥
 CDE0 陌陌男泥
 CDF0 陌陌男泥

