

3.5inch Arduino 16BIT Module MAR3513 用户手册

产品概述

该款 Arduino Mega2560 模块为 3.5 寸 TFT LCD 模块，拥有 480x320 分辨率，可显示 65K 彩色。其采用 16 位线并口通信方式，驱动 IC 为 ILI9486。该模块包含有 LCD 显示屏，5V~3.3V 电平转换电路，可以直插到 Arduino mega2560 开发板上使用，还支持 SD 卡和 SPI Flash 功能扩展。

产品特点

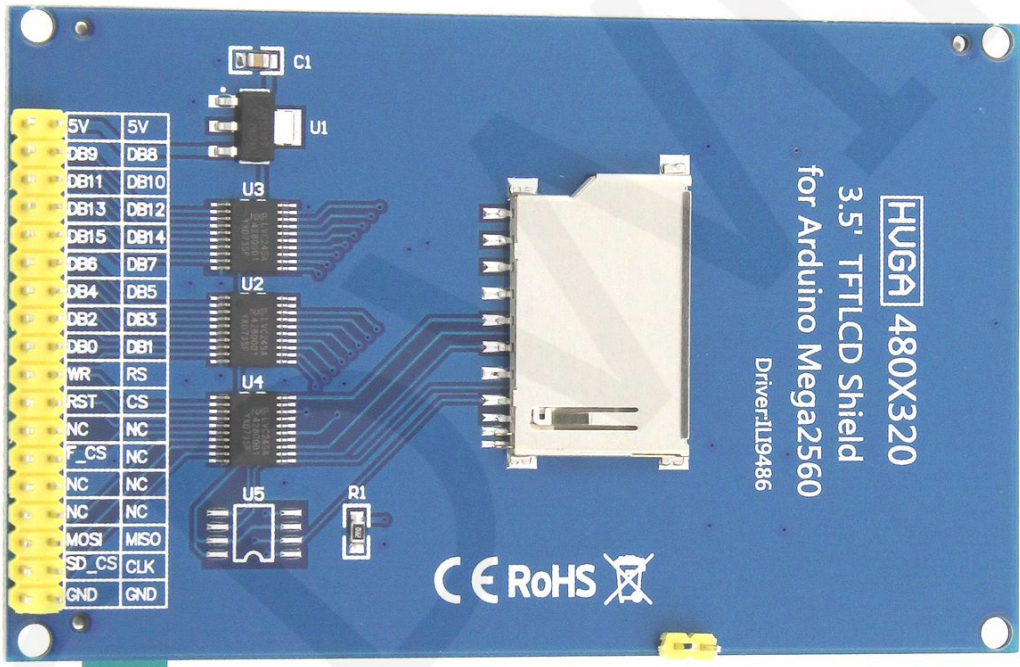
- 3.5 寸彩屏，支持 16BIT RGB 65K 色显示，显示色彩丰富
- 320x480 高清分辨率，显示效果清晰
- 采用 16 位并行总线传输，传输速度快
- 板载 5V/3.3V 电平转换 IC，兼容 5V/3.3V 工作电压
- 支持 Arduino Mage2560 直插式使用
- 提供 Arduino 库和丰富的示例程序
- 支持 SD 卡和 SPI Flash 功能扩展
- 军工级工艺标准,长期稳定工作
- 提供底层驱动技术支持

产品参数

名称	描述
显示颜色	RGB 65K 彩色
SKU	MAR3513
尺寸	3.5(inch)
类型	TFT
驱动芯片	ILI9486
分辨率	480*320 (Pixel)
模块接口	16Bit parallel interface
有效显示区域	48.96x73.44 (mm)

模块尺寸	60.30x96.60 (mm)
背光	6 chip HighLight white LEDs
工作温度	-10℃~60℃
存储温度	-20℃~70℃
工作电压	3.3V / 5V
功耗	TDB
产品重量	约 45(g)

接口说明



模块引脚丝印图

序号	模块引脚	引脚说明
1	5V	电源引脚
2	DB0	数据总线低8位引脚
3	DB1	
4	DB2	
5	DB3	
6	DB4	

7	DB5	
8	DB6	
9	DB7	
10	DB8	数据总线高8位引脚
11	DB9	
12	DB10	
13	DB11	
14	DB12	
15	DB13	
16	DB14	
17	DB15	
18	RS	液晶屏寄存器/数据选择引脚
19	WR	液晶屏写控制引脚
20	CS	液晶屏片选控制引脚
21	RST	液晶屏复位控制引脚
22	NC	无定义，保留
23	F_CS	扩展引用：SPI flash片选引脚
24	MISO	SPI总线输入引脚（扩展应用）
25	MOSI	SPI总线输出引脚（扩展应用）
26	CLK	SPI总线时钟引脚（扩展应用）
27	SD_CS	扩展引用：SD卡片选引脚
28	GND	电源地

硬件配置

该 LCD 模块硬件电路包含三大部分：LCD 显示控制电路、电平转换电路、SD 卡控制电路。

LCD 显示控制电路用于控制 LCD 的引脚，包括控制引脚和数据传输引脚。

电平转换电路用于进行 5V/3.3V 转换，使模块可以兼容 3.3V/5V 电源。

SD 卡控制电路用于 SD 卡功能扩展，控制 SD 卡的识别，读取及写入。

工作原理

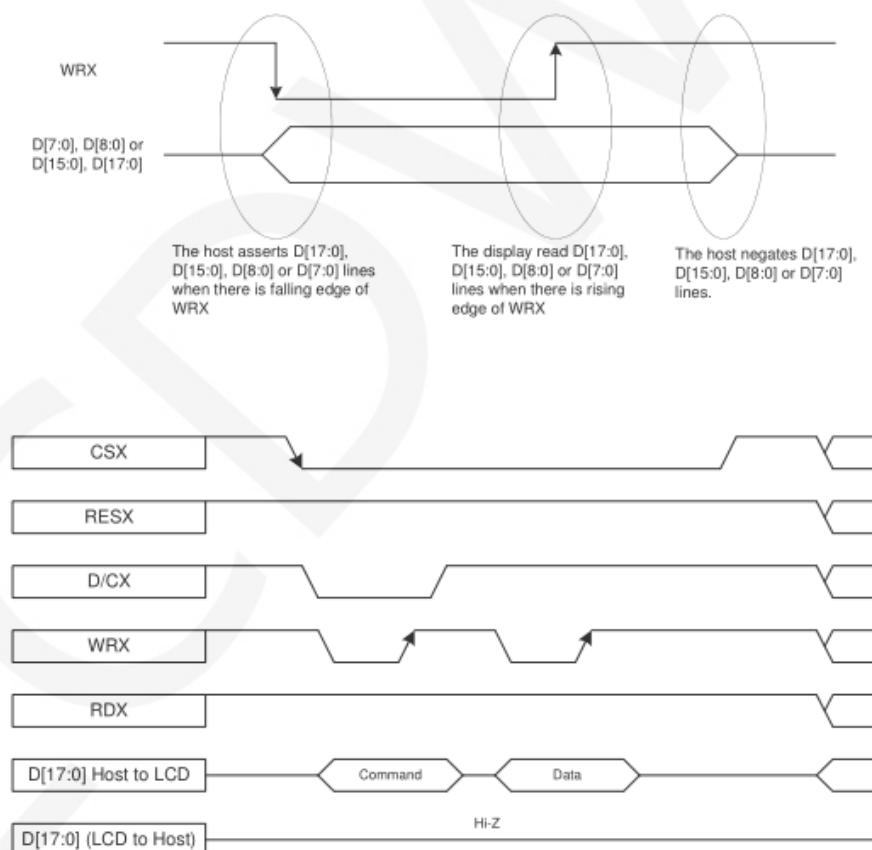
1、ITI9486 控制器简介

ITI9486 控制器支持的最大分辨率为 320*480, 拥有一个 345600 字节大小的 GRAM。同时支持 8 位、9 位、16 位、18 位并口数据总线, 还支持 3 线制和 4 线制 SPI 串口。由于支持的分辨率比较大, 传输的数据量大, 所以采用并口传输, 传输速度快。ITI9486 还支持 65K、262K RGB 颜色显示, 显示色彩很丰富, 同时支持旋转显示和滚动显示以及视频播放, 显示方式多样。

ITI9486 控制器使用 16bit (RGB565) 来控制一个像素点显示, 因此可以每个像素点显示颜色多达 262K 种。像素点地址设置按照行列的顺序进行, 递增递减方向由扫描方式决定。ITI9486 显示方法按照先设置地址再设置颜色值进行。

2、并口通信简介

并口通信写模式时序如下图所示:



CSX 为片选信号, 用于开启和禁止并口通信, 低电平有效

RESX 为外部复位信号，低电平有效

D/CX 为数据或者命令选择信号，1-写数据或者命令参数，0-写命令

WRX 为写数据控制信号

D[X:0]为并口数据位，共有 8 位、9 位、16 位、18 位四种类型

当进行写入操作时，在已经复位的基础上，先设置数据或者命令选择信号，然后将片选信号拉低，接下来从主机输入需要写入的内容，然后将写数据控制信号拉低再拉高，数据在写控制信号的上升沿会被写入到液晶屏控制 IC，最后将片选信号拉高，一次数据写入操作完成。

使用说明

1、Arduino 使用说明

接线说明：

引脚标注见接口说明。

此模块可以直接插入 Mega2560 中使用，不需要再手动接线。

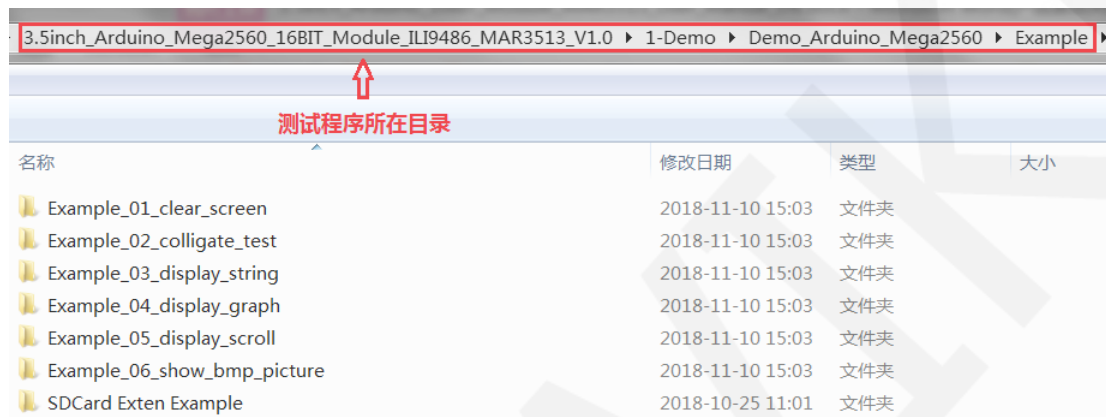


与Mega2560直插图

Arduino MEGA2560单片机测试程序直插说明		
序号	模块引脚	对应MEGA2560开发板直插引脚
1	5V	5V
2	DB0	37
3	DB1	36
4	DB2	35
5	DB3	34
6	DB4	33
7	DB5	32
8	DB6	31
9	DB7	30
10	DB8	22
11	DB9	23
12	DB10	24
13	DB11	25
14	DB12	26
15	DB13	27
16	DB14	28
17	DB15	29
18	RS	38
19	WR	39
20	CS	40
21	RST	41
22	NC	不需要接
23	F_CS	45
24	MISO	50
25	MOSI	51
26	CLK	52
27	SD_CS	53
28	GND	GND

操作步骤:

- A、按照上述接线说明将 LCD 模块直插到 Arduino 单片机上，并上电；
- B、将测试程序包中 **Install libraries** 目录下的依赖库拷贝到 Arduino 工程目录的 **libraries** 文件夹下（如果不需要依赖库，则不需要拷贝）；
- C、打开 Arduino 测试程序所在目录，选择需要测试的示例，如下图所示：
（测试程序说明请查阅测试程序包中测试程序说明文档）



- D、打开所选的示例工程，进行编译和下载。

关于 Arduino 测试程序依赖库拷贝、编译和下载的具体操作方法见如下文档：

http://www.lcdwiki.com/res/PublicFile/Arduino_IDE_Use_Illustration_CN.pdf

- E、LCD 模块如果正常显示字符和图形，则说明程序运行成功；

2、C51 使用说明**接线说明:**

引脚标注见接口说明。

STC89C52RC单片机测试程序接线说明		
序号	模块引脚	对应STC89开发板接线引脚
1	5V	5V
2	DB0	P30
3	DB1	P31
4	DB2	P32
5	DB3	P33
6	DB4	P34

7	DB5	P35
8	DB6	P36
9	DB7	P37
10	DB8	P20
11	DB9	P21
12	DB10	P22
13	DB11	P23
14	DB12	P24
15	DB13	P25
16	DB14	P26
17	DB15	P27
18	RS	P12
19	WR	P11
20	CS	P13
21	RST	P14
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

STC12C5A60S2单片机测试程序接线说明

序号	模块引脚	对应STC12开发板接线引脚
1	5V	5V
2	DB0	P00
3	DB1	P01
4	DB2	P02
5	DB3	P03

6	DB4	P04
7	DB5	P05
8	DB6	P06
9	DB7	P07
10	DB8	P20
11	DB9	P21
12	DB10	P22
13	DB11	P23
14	DB12	P24
15	DB13	P25
16	DB14	P26
17	DB15	P27
18	RS	P12
19	WR	P11
20	CS	P13
21	RST	P33
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

操作步骤:

- 按照上述接线说明将 LCD 模块和 C51 单片机连接起来，并上电；
- 选择需要测试的 C51 测试程序，如下图所示：

3.5inch ▶ 3.5inch_Arduino_Mega2560_16BIT_Module_ILI9486_MAR3513_V1.0 ▶ 1-Demo ▶ Demo_C51 ▶			
测试程序所在目录			
名称	修改日期	类型	大小
C51_Demo_STC12C5A60S2_16BIT	2018-11-10 15:04	文件夹	
C51_Demo_STC89C52RC_16BIT	2018-11-10 15:04	文件夹	
3.5inch_16BIT_Module_C51_Demo_Instructions.pdf	2018-11-20 17:07	看图王 PDF 文件	

C、打开所选的测试程序工程，进行编译和下载；

关于 C51 测试程序编译和下载的详细说明见如下文档：

http://www.lcdwiki.com/res/PublicFile/C51_Keil%26stc-isp_Use_Illustration_CN.pdf

D、LCD 模块如果正常显示字符和图形，则说明程序运行成功；

3、STM32 使用说明

接线说明：

引脚标注见接口说明。

STM32F103RCT6单片机测试程序接线说明		
序号	模块引脚	对应MiniSTM32开发板接线引脚
1	5V	5V
2	DB0	PB0
3	DB1	PB1
4	DB2	PB2
5	DB3	PB3
6	DB4	PB4
7	DB5	PB5
8	DB6	PB6
9	DB7	PB7
10	DB8	PB8
11	DB9	PB9
12	DB10	PB10
13	DB11	PB11
14	DB12	PB12
15	DB13	PB13
16	DB14	PB14

17	DB15	PB15
18	RS	PC8
19	WR	PC7
20	CS	PC9
21	RST	PC10
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

STM32F103ZET6单片机测试程序接线说明

序号	模块引脚	对应Elite STM32开发板接线引脚
1	5V	5V
2	DB0	PF0
3	DB1	PF1
4	DB2	PF2
5	DB3	PF3
6	DB4	PF4
7	DB5	PF5
8	DB6	PF6
9	DB7	PF7
10	DB8	PF8
11	DB9	PF9
12	DB10	PF10
13	DB11	PF11
14	DB12	PF12
15	DB13	PF13
16	DB14	PF14
17	DB15	PF15

18	RS	PC8
19	WR	PC7
20	CS	PC9
21	RST	PC10
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

STM32F407ZGT6单片机测试程序接线说明

序号	模块引脚	对应Explorer STM32F4开发板接线引脚
1	5V	5V
2	DB0	PG0
3	DB1	PG1
4	DB2	PG2
5	DB3	PG3
6	DB4	PG4
7	DB5	PG5
8	DB6	PG6
9	DB7	PG7
10	DB8	PG8
11	DB9	PG9
12	DB10	PG10
13	DB11	PG11
14	DB12	PG12
15	DB13	PG13
16	DB14	PG14
17	DB15	PG15
18	RS	PC8

19	WR	PC7
20	CS	PC9
21	RST	PC10
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

STM32F429IGT6单片机测试程序接线说明

序号	模块引脚	对应Apollo STM32F4/F7开发板接线引脚
1	5V	5V
2	DB0	PE0
3	DB1	PE1
4	DB2	PE2
5	DB3	PE3
6	DB4	PE4
7	DB5	PE5
8	DB6	PE6
9	DB7	PE7
10	DB8	PE8
11	DB9	PE9
12	DB10	PE10
13	DB11	PE11
14	DB12	PE12
15	DB13	PE13
16	DB14	PE14
17	DB15	PE15
18	RS	PC8
19	WR	PC7

20	CS	PC9
21	RST	PC10
22	NC	不需要接
23	F_CS	不需要接
24	MISO	不需要接
25	MOSI	不需要接
26	CLK	不需要接
27	SD_CS	不需要接
28	GND	GND

操作步骤:

- 按照上述接线说明将 LCD 模块和 STM32 单片机连接起来，并上电；
- 选择需要测试的 STM32 测试程序，如下图所示：



- 打开所选的测试程序工程，进行编译和下载；

关于 STM32 测试程序编译和下载的详细说明见如下文档：

http://www.lcdwiki.com/res/PublicFile/STM32_Keil_Use_Illustration_CN.pdf

- LCD 模块如果正常显示字符和图形，则说明程序运行成功；

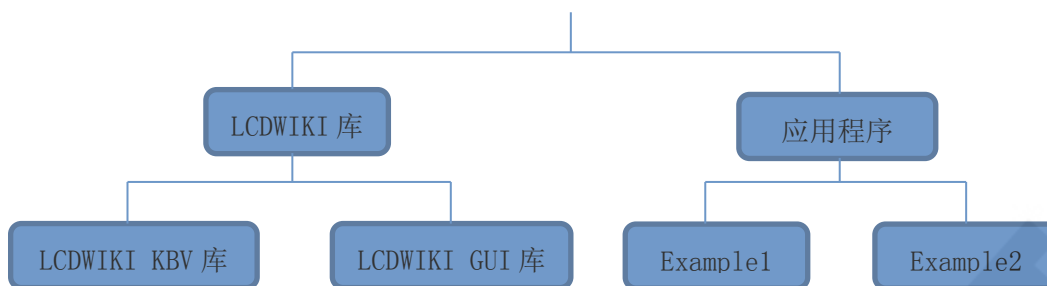
软件说明

1、代码架构

A、Arduino 代码架构说明

代码架构如下图所示：

示例代码



Arduino 的测试程序代码由两部分组成：LCDWIKI 库和应用代码。

LCDWIKI 库包含两部分内容：LCDWIKI_KBV 库和 LCD_GUI 库。

应用程序包含几个测试示例，每个测试示例包含不同的测试内容

LCDWIKI_KBV 为底层库，和硬件有关联，主要负责操作寄存器，包括硬件模块初始化，数据和命令传输，像素点坐标和颜色设置，显示方式配置等。

LCDWIKI_GUI 为中间层库，主要负责使用底层库提供的 API 实现图形的绘制，字符显示等操作。

应用程序是利用 LCDWIKI 库提供的 API，编写一些测试示例，实现某方面的测试功能。

B、C51 和 STM32 代码架构说明

代码架构如下图所示：



主程序运行时的 Demo API 代码包含在 test 代码中；

LCD 初始化以及相关的斌并口写数据操作都包含在 LCD 代码中；

画点、线、图形以及中英文字符显示相关的操作都包含在 GUI 代码中；

主函数实现应用程序运行；

平台代码因平台而异；

2、GPIO 定义说明

A、Arduino 测试程序 GPIO 定义说明

模块是直插到 Arduino mega2560 上使用的，所以不允许修改 GPIO 口定义。

B、C51 测试程序 GPIO 定义说明

C51 测试程序 GPIO 定义放在 lcd.h 文件里，如下图所示：

```
//Io连接
#define LCD_DataPortH P2 //高8位数据口,8位模式下只使用高8位
#define LCD_DataPortL P0 //低8位数据口,8位模式下低8位可以不
sbit LCD_RS = P1^2; //数据/命令切换
sbit LCD_WR = P1^1; //写控制
//sbit LCD_RD = P1^0; //读控制
sbit LCD_CS = P1^3; //片选
sbit LCD_RESET = P3^3; //复位
//sbit LCD_BL=P3^2; //背光控制, 如果不需要控制, 接3.3V
```

并口引脚定义需要选择整套 GPIO 口组，如 P1，P2 等，这样传输数据时，操作方便。

其他引脚可以定义成任何空闲的 GPIO。

C、STM32 测试程序 GPIO 定义说明

STM32 测试程序 GPIO 定义放在 lcd.h 文件里，如下图所示：

```
//////////////////////////////////////
//-----LCD端口定义-----
#define GPIO_TYPE GPIOC //GPIO组类型
//#define LED 4 //背光控制引脚 PC4
#define LCD_CS 9 //片选引脚 PC9
#define LCD_RS 8 //寄存器/数据选择引脚 PC8
#define LCD_RST 5 //复位引脚 PC5
#define LCD_WR 7 //写引脚 PC7

//QDtech全系列模块采用了三极管控制背光亮灭，用户也可以接PWM调节背光亮度
//#define LCD_LED PBout(LED) //LCD背光 PC4
//如果使用官方库函数定义下列底层，速度将会下降到14帧每秒，建议采用我司推荐方法
//以下Io定义直接操作寄存器，快速Io操作，刷屏速率可以达到28帧每秒！

//GPIO置位（拉高）
#define LCD_CS_SET GPIO_TYPE->BSRR=1<<LCD_CS //片选端口
#define LCD_RS_SET GPIO_TYPE->BSRR=1<<LCD_RS //数据/命令
#define LCD_RST_SET GPIO_TYPE->BSRR=1<<LCD_RST //复位
#define LCD_WR_SET GPIO_TYPE->BSRR=1<<LCD_WR //写

//GPIO复位（拉低）
#define LCD_CS_CLR GPIO_TYPE->BRR=1<<LCD_CS //片选端口
#define LCD_RS_CLR GPIO_TYPE->BRR=1<<LCD_RS //数据/命令
#define LCD_RST_CLR GPIO_TYPE->BRR=1<<LCD_RST //复位
#define LCD_WR_CLR GPIO_TYPE->BRR=1<<LCD_WR //写

//PB0~15,作为数据线
//注意：如果使用8位模式数据总线，则液晶屏的数据高8位是接到MCU的高8位总线上
//举例：如果接8位模式则本示例接线为液晶屏DB10-DB17对应接至单片机GPIOB_Pin8-GPIOB
//举例：如果是16位模式：DB0-DB7分别接GPIOB_Pin0-GPIOB_Pin7,DB10-DB17对应接至单片机
#define DATAOUT(x) GPIOB->ODR=x; //数据输出
```

数据并口引脚定义需要选择整套 GPIO 口组，如 PB，PE 等，传输数据时，操作方便。

其他引脚可以定义成任何空闲的 GPIO。

3、并口通信代码实现

A、Arduino 测试程序并口通信代码实现

相关的代码在 LCDWIKI_KBV 库的 mcu_16bit_magic.h 文件里实现，如下图所示：

```
// Data write strobe, ~2 instructions and always inline
#define WR_STROBE { WR_ACTIVE; WR_IDLE; }
#define RD_STROBE { RD_IDLE; RD_ACTIVE; RD_ACTIVE; RD_ACTIVE; }
#define write16(x) { write_16(x) }
#define read16(dst) { read_16(dst) }
#define writeCmd8(x){ CD_COMMAND; write8(x); CD_DATA; }
#define writeData8(x){ write8(x) }
#define writeCmd16(x){ CD_COMMAND; write16(x); CD_DATA; }
#define writeData16(x){ write16(x) }

#define write_16(x)    { PORTA = (x) >> 8; PORTC = x; WR_STROBE; }
#define write8(x)     { PORTC = x; WR_STROBE; }
```

实现了 8、16 位命令以及 8、16 位数据传输。

B、C51 和 STM32 测试程序并口通信代码实现

相关的代码在 LCD.c 文件里实现，如下图所示：

```
void LCD_write(u16 VAL)
{
    LCD_CS_CLR;
    DATAOUT(VAL);
    LCD_WR_CLR;
    LCD_WR_SET;
    LCD_CS_SET;
}
```

实现了 8、16 位命令以及 8、16 位数据传输。

常用软件

本套测试示例需要显示中英文、符号以及图片，所以要用到取模软件。取模软件有两种：Image2Lcd 和 PCtoLCD2002。这里只针对该套测试程序说明一下取模软件的设置。

PCtoLCD2002 取模软件设置如下：

点阵格式选择**阴码**

取模方式选择**逐行式**

取模走向选择**顺向（高位在前）**

输出数制选择**十六进制数**

自定义格式选择**C51 格式**

具体设置方法见如下网页：

<http://www.lcdwiki.com/zh/%E3%80%90%E6%95%99%E7%A8%8B%E3%80%91%E4%B8%AD%E8%8B%B1%E6%96%87%E6%98%BE%E7%A4%BA%E5%8F%96%E6%A8%A1%E8%AE%BE%E7%BD%AE>

Image2Lcd 取模软件设置如下图所示：



Image2Lcd 软件需要设置为水平、自左向右、自上向下、低位在前扫描方式。