

1. Introduction to Testing Platform

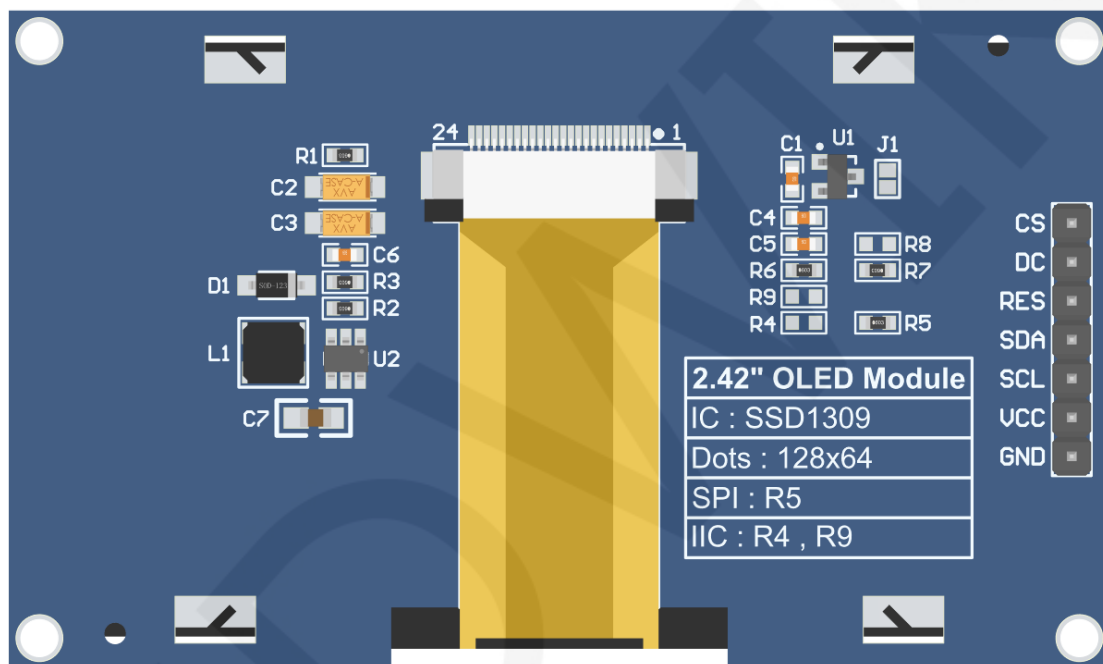
Development board: STC89/STC12 development board

MCU : STC89C52RC、STC12C5A60S2

Frequency : 11.0592MHZ

2. Pin connection instructions

The display module is connected to the microcontroller using a DuPont cable, with specific instructions as follows:



Module Back Pins

NOTE:

- Connect to a 5V microcontroller, which can short circuit J1 to keep the IO voltage and IO high level consistent;
- R8 is not soldered by default. If there is no need to control the CS pin, R8 solders the 0R resistor to keep the CS signal grounded;
- If SPI communication mode is selected, R5 will weld 0R resistor, and R4 and R9 will be disconnected;
- If IIC communication mode is selected, R4 and R9 will be welded with 0R resistor, and R5 will be disconnected;

STC89C52RC and STC12C5A60S2 Microcontrollers SPI Test Program Wiring Instructions

Number	Module pins	Corresponding STC89/STC12 development board wiring pins	Remarks
1	GND	GND	OLED screen power supply ground
2	VCC	5V/3.3V	OLED screen power supply positive
3	SCL	P17	SPI bus clock signal
4	SDA	P15	SPI bus write data signal
5	RES	P33	OLED screen reset control signal, low-level reset
6	DC	P12	OLED screen command/data selection control signal High level: data, low level: command
7	CS	P13	OLED screen chip selection control signal, effective at low level (if welding R8, CS pin may not be connected)

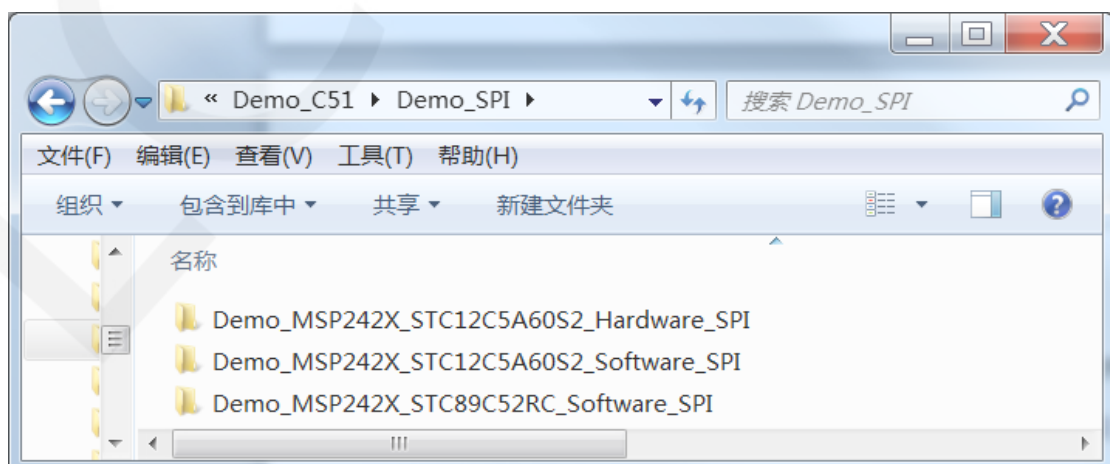
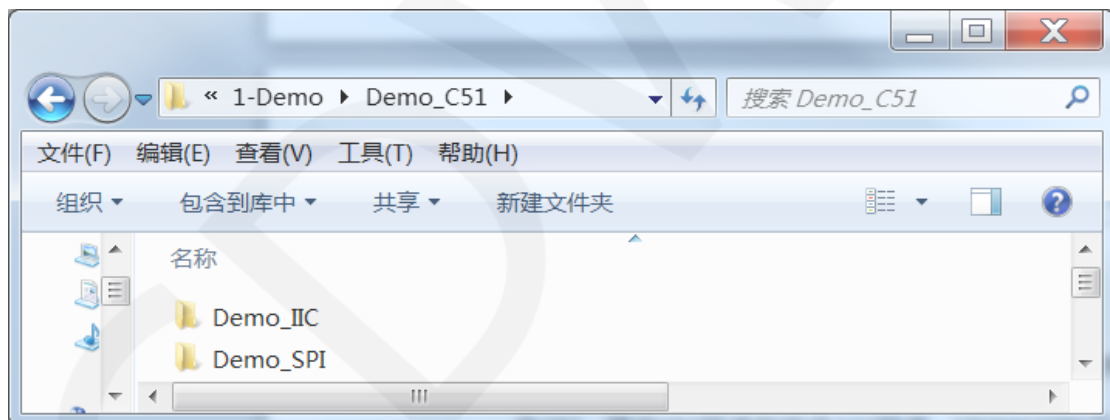
STC89C52RC and STC12C5A60S2 Microcontrollers IIC Test Program Wiring Instructions

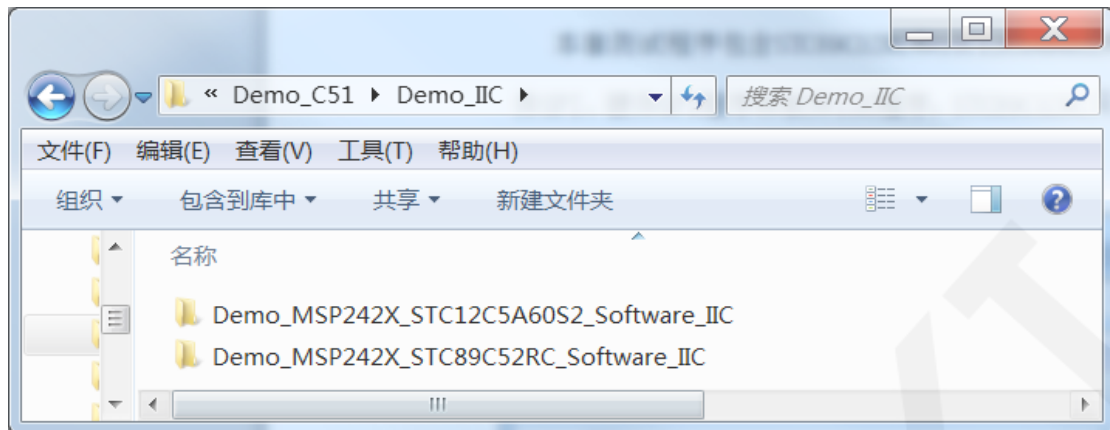
Number	Module pins	Corresponding STC89/STC12 development board wiring pins	Remarks
1	GND	GND	OLED screen power supply ground
2	VCC	5V/3.3V	OLED screen power supply positive
3	SCL	P17	IIC bus clock signal
4	SDA	P15	IIC bus data signal
5	RES	P33/3.3V	OLED screen reset control signal, low-level reset (if no control is required, the RES pin can be connected to a high-level (3.3V))
6	DC	P12/GND/3.3V	IIC bus selects signal from device address When connecting to the P12 pin, set it to low level: 0x78, and set it to high level:

			0x7A Low level (connected to GND): 0x78, high level (connected to 3.3V): 0x7A
7	CS	P13/GND	OLED screen chip selection control signal, effective at low levels When using IIC communication, there is no need for control. When connecting to P13, P13 must be set to low level or GND can be connected (if welding R8, CS pin can not be connected)

3. Demo Function Description

This testing program includes two types of MCU programs: STC89C52RC and STC12C5A60S2. STC12C5A60S2 includes software SPI, hardware SPI program, and software IIC program. STC89C52RC only has software SPI program and software IIC program, which are located in **Demo_C51** directory, as shown in the following figure:





✧ Description of sample program content

The sample program includes the following content:

- A. Home screen display;
- B. Single color screen brushing
- C. English display;
- D. Display of numbers and symbols
- E. Chinese display;
- F. BMP monochrome image display;

✧ Example program display direction switching instructions

Found macro definition **USE_HORIZONTAL** and **COLOR_STATE** in **HARDWARE\OLED\oled.h** file, as shown in the following:

```
#define USE_HORIZONTAL 0 // 设置显示方向: 0-正常, 1-旋转180度  
#define COLOR_STATE 0 // 设置显示模式: 0-正常显示, 1-反色显示
```

Modify **USE_HORIZONTAL** and **COLOR_STATE** macro according to the following definition:

```
#define USE_HORIZONTAL 0 //0 ° rotation (Default value)  
#define USE_HORIZONTAL 1 //180 ° rotation  
#define COLOR_STATE 0 //Black background, monochrome display  
content(Default value)  
#define COLOR_STATE 1 //Monochrome background, black display content
```

✧ Example program IIC slave device address modification instructions (only for IIC test programs)

First, locate the macro definition **IIC_SLAVE_ADDR** in the **HARDWARE\IIC\iic.h** file, as shown in the following figure:

```
//定义IIC从设备地址
#define IIC_SLAVE_ADDR 0x78 //0x7A
```

Modify **IIC_SLAVE_ADDR** macro definition according to the following definition is sufficient to:

```
#define IIC_SLAVE_ADDR 0x78 //Slave device address is 0x78 (default value)
```

```
#define IIC_SLAVE_ADDR 0x7A //Slave device address is 0x7A
```

Next, find **OLED_Init** function in the **HARDWARE\OLED\oled.c** file. If using the 0x7A slave device address, there is no need to annotate the two lines of code **P1&=0xF7** and **P1|=0x04** (to make them effective). If using the 0x78 slave device address, the two lines of code **P1&=0xF7** and **P1|=0x04** need to be annotated (to make them ineffective), as shown in the following figure:

```
void OLED_Init(void)
{
    P1 &= 0xF3;
    //slave address is 0x7A, select the follow define:
    //P1 &= 0xF7;
    //P1 |= 0x04;
    OLED_Reset(); //复位OLED
```

4. Demo Usage Instructions

✧ Installing development tool software

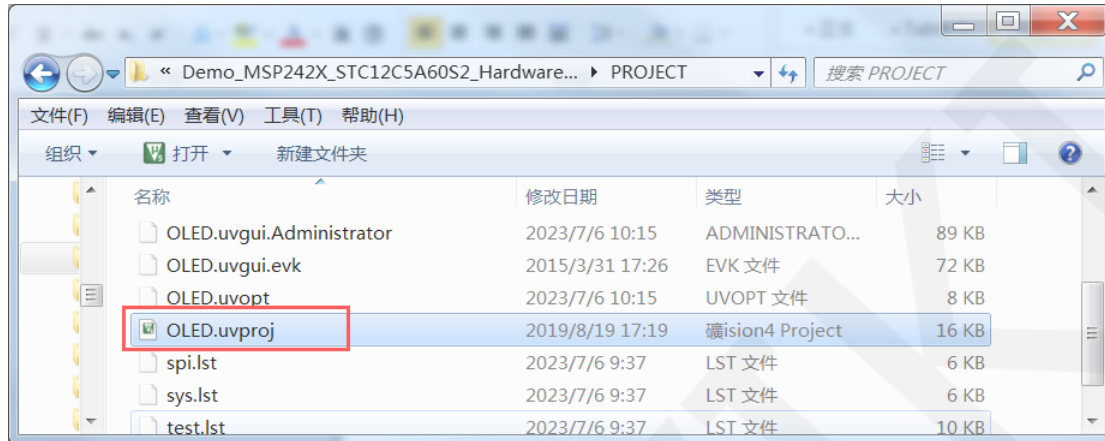
Firstly, you need to install the development tool software. Keil5 and stc-isp software are used here, where Keil5 is used for code editing and compilation, and stc-isp is used for download. Please refer to the online download and installation methods for both software.

✧ Installing chip packages

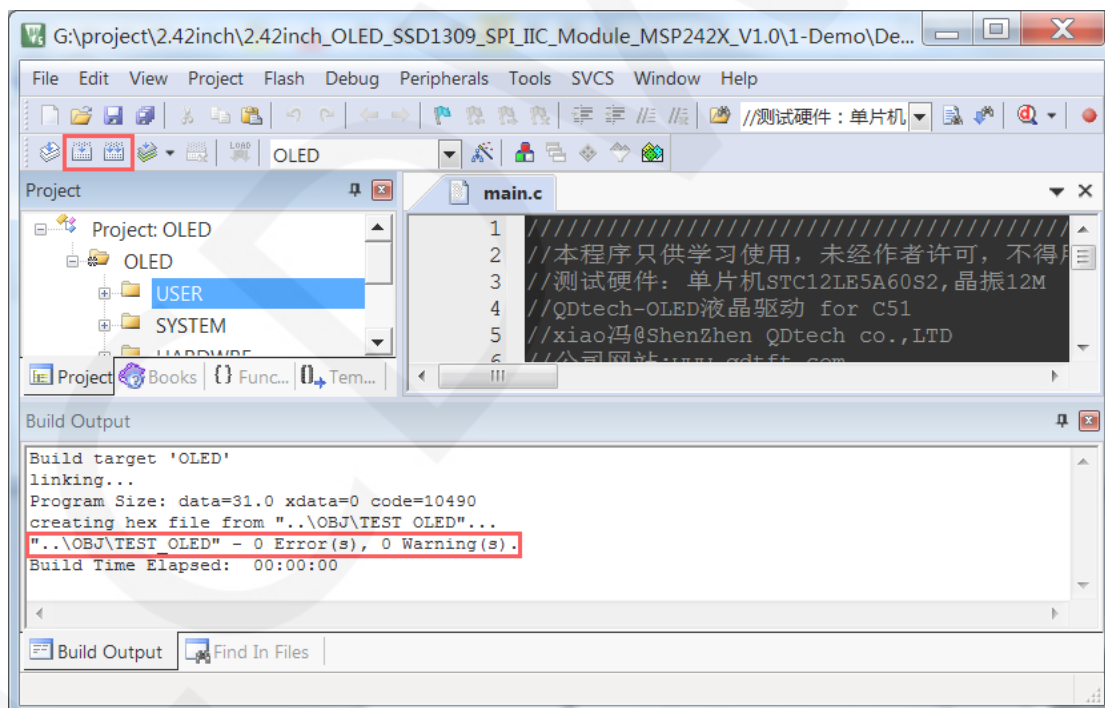
After installing keil5, it is necessary to install the C51 chip package, otherwise the C51 chip cannot be found and the C51 project cannot be created. Please consult online for specific installation methods.

✧ Compiling Programs

After the development tool and chip package are successfully installed, open the **PROJECT** directory under the sample program, locate the **uvprojx** file, double-click to open the sample project, as shown in the following figure:



After opening the sample project, you can make modifications to the project code (or not). After the modifications are completed, click the compile button to compile the code. The following prompt appears, indicating successful compilation, as shown in the following figure:

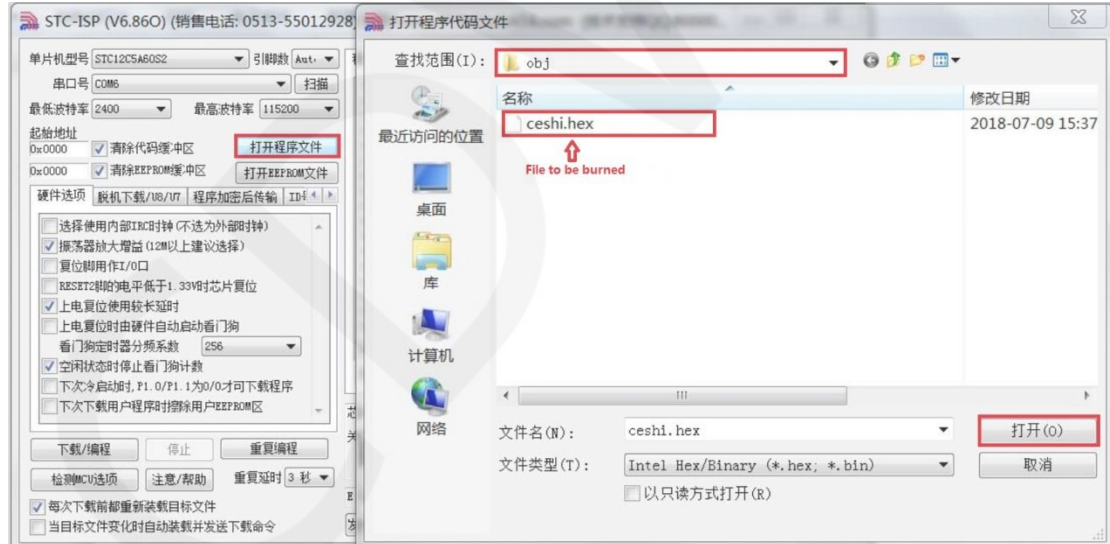


❖ Download and Run Programs

- Open the **STC-ISP** software for program download, first select the correct microcontroller model and baud rate, and set them as shown in the following figure:



B. Click to open the program file ->select the directory where the compiled hex file is located ->select the hex file ->click the open button, as shown in the following figure:



C. Click the **download** button to power on the microcontroller again, and the program will be burned. When the "**Operation successful**" prompt appears, it indicates successful burning. The operation is shown in the following figure:



- D. If the display module displays characters and graphics normally, it indicates that the program has run successfully.