1. Introduction to Testing Platform

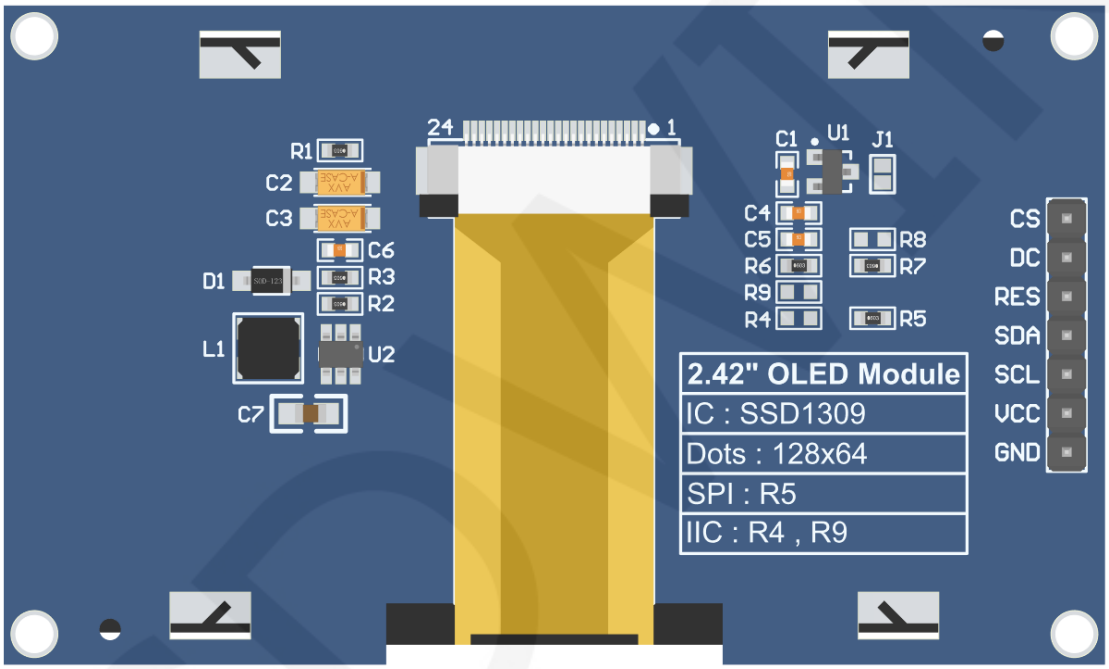
Development board: Arduino UNO/MEGA2560

MCU: AVR_ATmega328P/AVR_ATmega2560

Frequency: 16MHz/16MHz

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:

A. Connect to a 5V microcontroller, which can short circuit J1 to keep the IO voltage and IO high level consistent;

B. 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;

C. If SPI communication mode is selected, R5 will weld 0R resistor, and R4 and R9 will be disconnected;

D. If IIC communication mode is selected, R4 and R9 will be welded with 0R resistor, and R5 will be disconnected;

Arduino UNO and Mega2560 microcontroller SPI test program wiring instructions					
Number	Module Pin	Corresponding to development board wiring pin			Remarks
		Hardware SPI		SoftWare SPI	
		UNO	Mega 2560		
1	GND	GND			OLED screen power supply ground
2	VCC	5V/3.3V			OLED screen power supply positive
3	SCL	13	52	13	SPI bus clock signal
4	SDA	11	51	11	SPI bus write data signal
5	RES	8			OLED screen reset control signal, low-level reset
6	DC	9			OLED screen command/data selection control signal High level: data, low level: command
7	CS	10			OLED screen chip selection control signal, effective at low level (if welding R8, CS pin may not be connected)

Arduino UNO and Mega2560 microcontroller IIC test program wiring instructions					
Number	Module Pin	Corresponding to development board wiring pin			Remarks
		Hardware IIC		SoftWare IIC	
		UNO	Mega 2560		
1	GND	GND			OLED screen power supply ground
2	VCC	5V/3.3V			OLED screen power supply positive
3	SCL	A5	21	A5	IIC bus clock signal
4	SDA	A4	20	A4	IIC bus data signal

5	**RES**	8/3.3V	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**	9/GND/3.3V	IIC bus selects signal from device address When connecting to the pin 9, pin 9 is low level: 0x78, and pin 9 is high level: 0x7A Low level (connected to GND): 0x78, high level (connected to 3.3V): 0x7A
7	**CS**	10/GND	OLED screen chip selection control signal, effective at low levels When using IIC communication, there is no need for control. When connecting to pin 10, it must be set to low level or GND can be connected (such as welding R8, CS pin can not be connected)

3. Demo Function Description

This set of testing programs includes two MCU programs, Arduino UNO and Mega2560. Each MCU program includes spi and iic testing programs, and each testing program includes hardware and software functional testing, which is located in **Demo_ UNO_ Mega2560** directory, as shown in the following figure:

✧ Description of sample program content

The testing program includes the following test items:

A. Example01-graph_ Test is a graphical display test

B. Example02 string_ Test is a character display test;

C. Example03 show_ BMP is a BMP bitmap display test;

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

Open any IIC sample program and locate the **setup** function. If using the 0x7A slave device address, there is no need to annotate the two lines of code **digitalWrite(9, HIGH)** and **u8g2.setI2CAddress(0x7A)** (to make them effective). If using the 0x78 slave device address, the two lines of code **digitalWrite(9, HIGH)** and **u8g2.setI2CAddress(0x7A)** need to be annotated (to make them ineffective), as shown in the following figure:

```
void setup()
{
  Serial.begin(9600);
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  digitalWrite(9, LOW);
  digitalWrite(10, LOW);
  /*When using 0x7A slave device address, please use the following definition*/
  //digitalWrite(9, HIGH);
  //u8g2.setI2CAddress(0x7A);
  u8g2.begin();
}
```

4. Demo Usage Instructions

✧ Installing development tool software

Download the installation package from the Arduino official website.

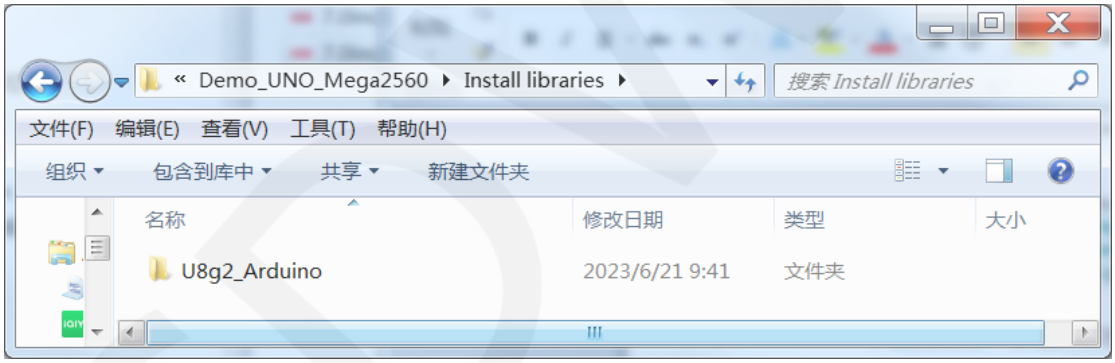
Download address:https://www.arduino.cc/en/software

Download the corresponding installation package according to your PC system, as shown in the following figure (the version in the picture may not be the latest version, and the download interface may not be the latest):

After the download is completed, unzip and click Install.

✧ Installing software library

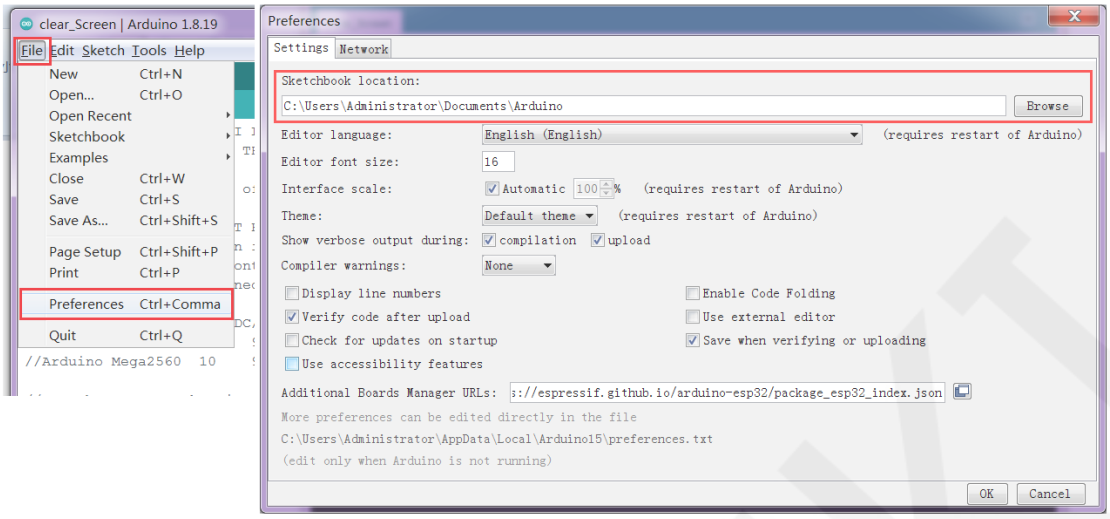
After the development environment is set up, the software library used by the sample program needs to be copied to the project library directory so that the sample program can be called. The software library is located in the **Demo_UNO_Mega2560\Install libraries** directory, as shown in the following figure:



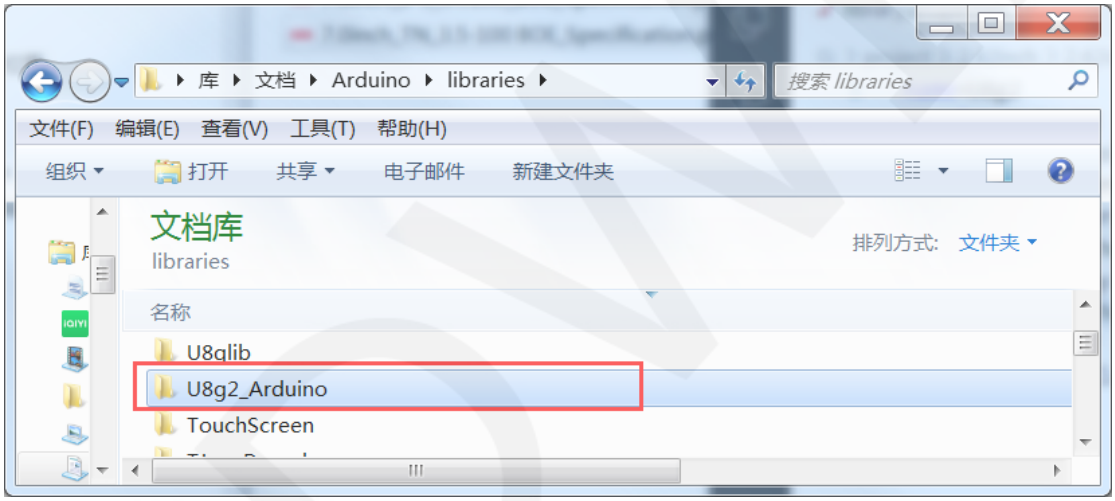
You can also download the latest software library from Github and unzip it (for easy differentiation, you can rename the unzipped folder, as shown in the Install libraries directory), and then copy it to the engineering library directory. The download address is as follows:

https://github.com/olikraus/U8g2_Arduino

The default path for the engineering library directory is **C: Users\Administrator \Documents\Arduino\libraries**. You can also change the project library directory: open the Arduino IDE software, click **File** ->**Preferences**, and reset the **Sketchbook location** in the pop-up interface, as shown in the following figure:



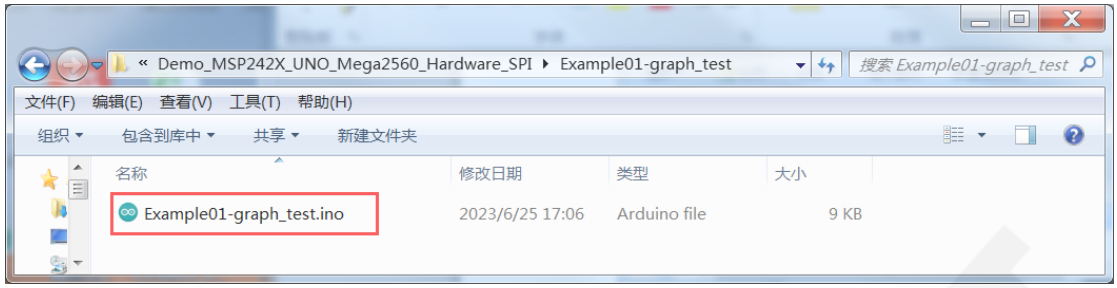
Copy the software library to the project library directory, as shown in the following figure:



✧ Compile and Run Programs

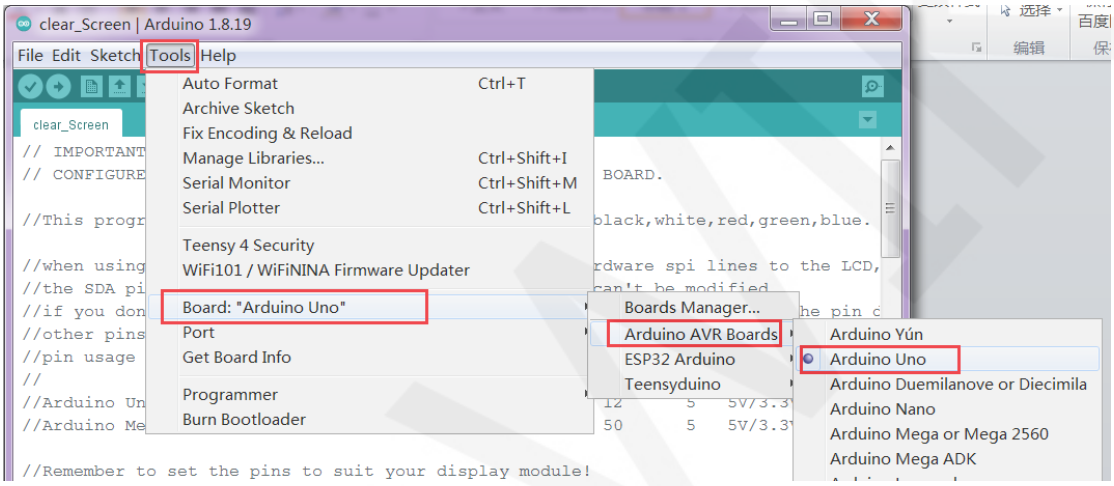
A. Connect the display module to the UNO or Mega2560 development board, and then power up the development board.

B. Open any example under Demo_UNO_Mega2560 directory (Here is the Example01-graph_test of the hardware SPI testing program as an example), as shown in the following figure:

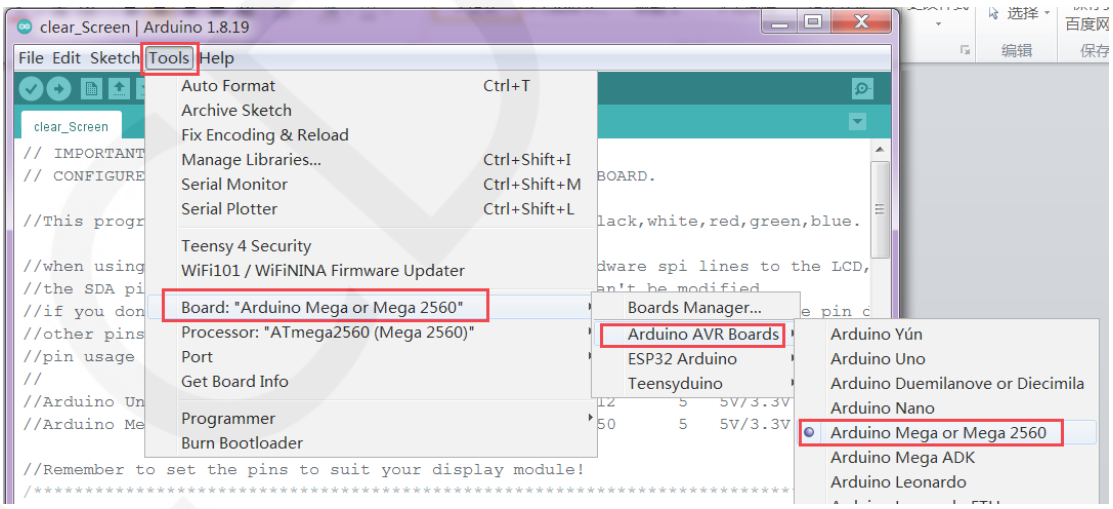


C. After opening the sample project, select the UNO or Mega2560 device, as shown in the following figure:

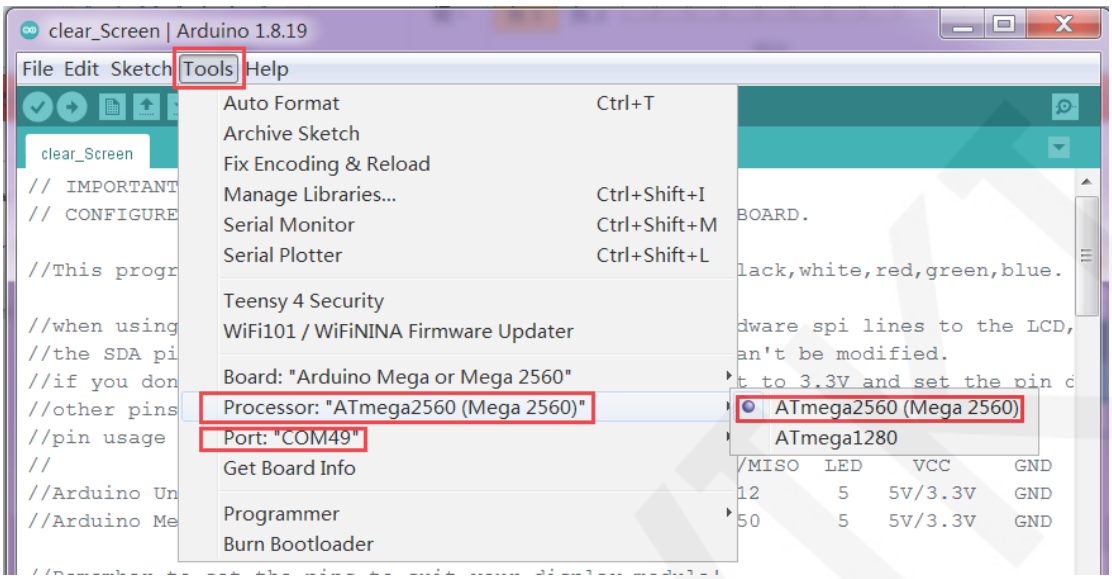
Select UNO:



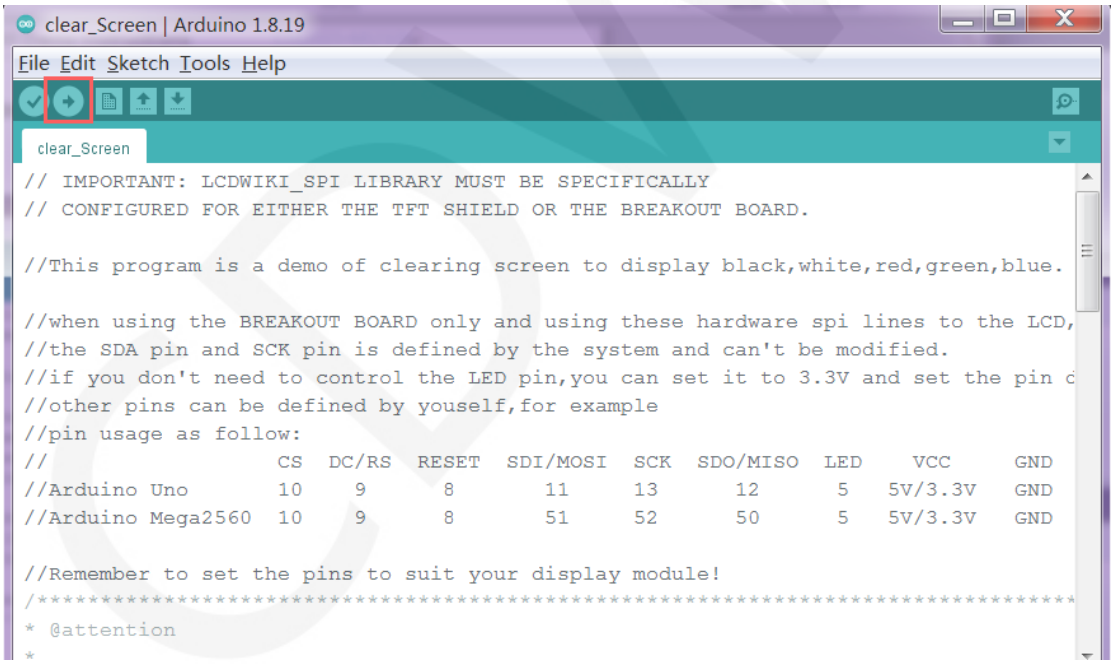
Select Mega2560：



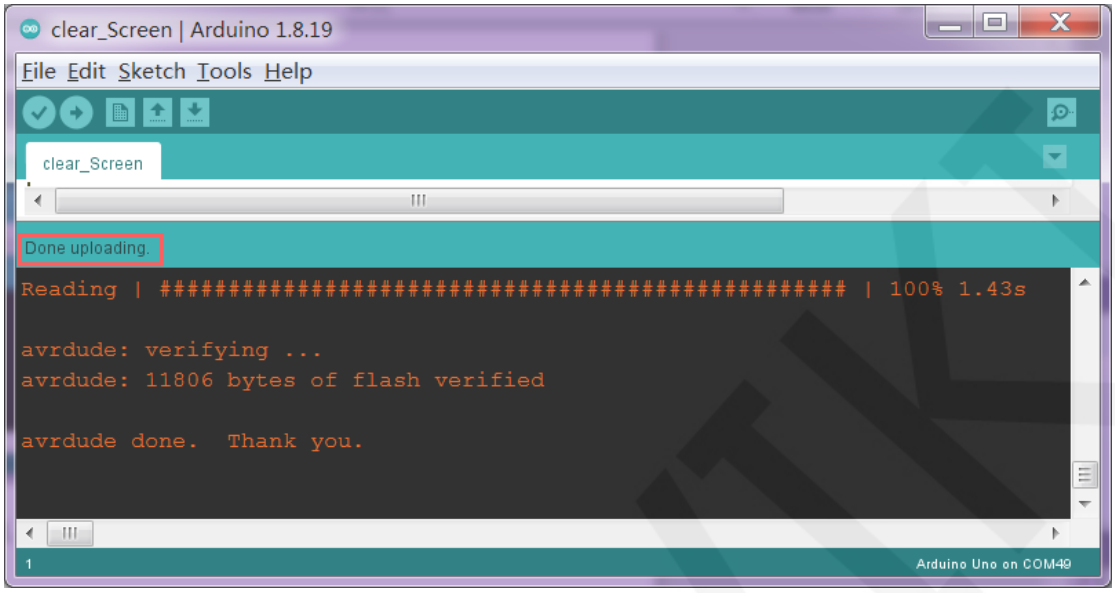
D. Set the **port**. If you choose Mega2560, you also need to set the processor based on the development board used, as shown in the following figure:



E. Click the upload button to compile and download the program, as shown in the following figure:



F. If the following prompt appears, it indicates that the program has been compiled and downloaded successfully, and has already been run:



G. If the display module displays content, it indicates that the program has run successfully.