ESP32S3 displays double eye expressions on dual circular 240x240 1.8-inch GC9A01 SPI displays

Using ESP32-S3 development board, connect the circular 240x240 1.8-inch GC9A01 SPI display, and use Arduino TFT-E SPI library to realize double eye expression display:



Step

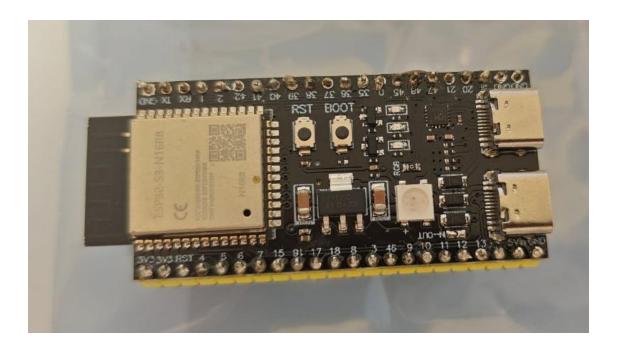
1. Circuit connection

| GC9A01 240x240 1.28 inch display | ESP32-S3-DevKitC-1 N16R8 development board |
|----------------------------------|--|
| VCC | 3V3 |
| GND | GND |
| SCL | 10 |
| SDA | 11 |
| DC | 8 |
| CS | 9 |
| RST | 4 |
| | |

GC9A01 240x240 1.28 inch display



ESP32-S3-DevKitC-1 N16R8 development board

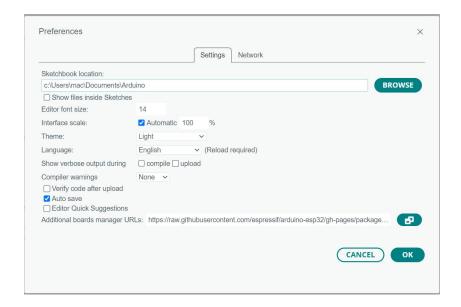


2. Install the development environment

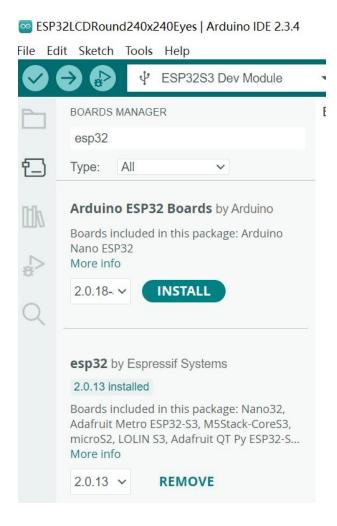
- 1. Install Aruidno IDE
- 2. Install Aruidno ESP32 chip pack
- (1) Open Arduino IDE, select File-> Options-> Settings.

Paste the following link into the address of the development board manager:

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_dev_index.json



(2) Open the Development Board Manager and search for "esp32". Locate the "ESP32 by Espressif Systems" entry. Select the version (choose 2.0.13 here as it has been tested without issues; higher versions may cause problems). Click Install to proceed with the installation process. Wait until the download and installation are complete. (If installation fails, try clicking Install again.)



Offline installation

If the download fails and the installation fails, you can install it offline.

Download the package directly:

File shared through the web disk: ESP32 chip package for Arduino

Link: https://pan.baidu.com/s/1rcNSqkbexJ5-YNmNXmt-dA?pwd=rd5h Extraction code: rd5h

Select the decompression path. It should be placed in the corresponding user's Arduino device package directory. The following is the installation path for Arduino version:



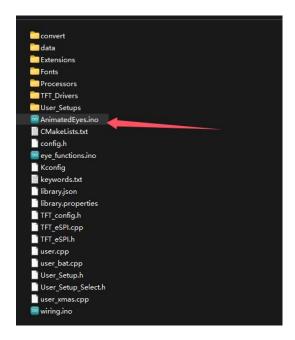
After the decompression to the corresponding folder is completed, close the software, open arduino again, click the development board manager, and you will see that esp32-arduino has been installed.

3. Download the code and compile it

Download the code

git clone https://github.com/iEmoBot/OpenEmo.git

After downloading the code, the Arduino code displayed in the eyes is located below AnimatedEyes



Open the code and burn it with Arduino IDE

```
AnimatedEyes | Arduino IDE 2.3.6
File Edit Sketch Tools Help
                 ₹ ESP32S3 Dev Module
                                    AnimatedEyes.ino TFT_eSPI.cpp TFT_eSPI.h TFT_config.h config.h User_Setup.h
       SKETCHBOOK
               7
                                      89
                                                                     // One-per-eye structure
                                            struct {
                                      90
                                              int16_t tft_cs;
                                                                    // Chip select pin for each display
13
                                              eyeBlink blink;
                                                                    // Current blink/wink state
                                      91
                                             int16_t xposition; // x position of eye image
                                      92
                                      93
                                            } eye[NUM_EYES];
                                      94
                                      95
                                            uint32_t startTime; // For FPS indicator
$ >
                                      96
                                            // INITIALIZATION -- runs once at startup -----
                                      97
                                            void setup(void) {
                                      98
                                      99
                                             Serial.begin(115200);
                                     100
                                              //while (!Serial);
                                             Serial.println("Starting");
                                     101
                                     102
                                            #if defined(DISPLAY_BACKLIGHT) && (DISPLAY_BACKLIGHT >= 0)
                                     103
                                     104
                                             // Enable backlight pin, initially off
                                     105
                                              Serial.println("Backlight turned off");
                                             pinMode(DISPLAY_BACKLIGHT, OUTPUT);
                                     106
                                             digitalWrite(DISPLAY_BACKLIGHT, LOW);
                                     107
                                     108
                                            #endif
                                     109
                                     110
                                              // User call for additional features
                                             user_setup();
                                     111
                                     112
                                              // Initialise the eye(s), this will set all chip selects low
                                     113
                                     114
                                              initEyes();
                                     115
                                              // Initialise TFT
                                     116
                                    Output
                                      Writing at 0x00010000... (3 %)
                                     Writing at 0x00015e0b... (7 %)
Writing at 0x00019f2b... (11 %)
                                     Writing at 0x0001f4b0... (15 %)
Writing at 0x0003567b... (19 %)
                                     Writing at 0x00041151... (23 %)
                                     Writing at 0x00046508... (26 %)
                                     Writing at 0x0004b401... (30 %)
                                     Writing at 0x000506b7... (34 %)
             NEW SKETCH
                                     Writing at 0x00056013... (38 %)
```

The software shows up after burning

