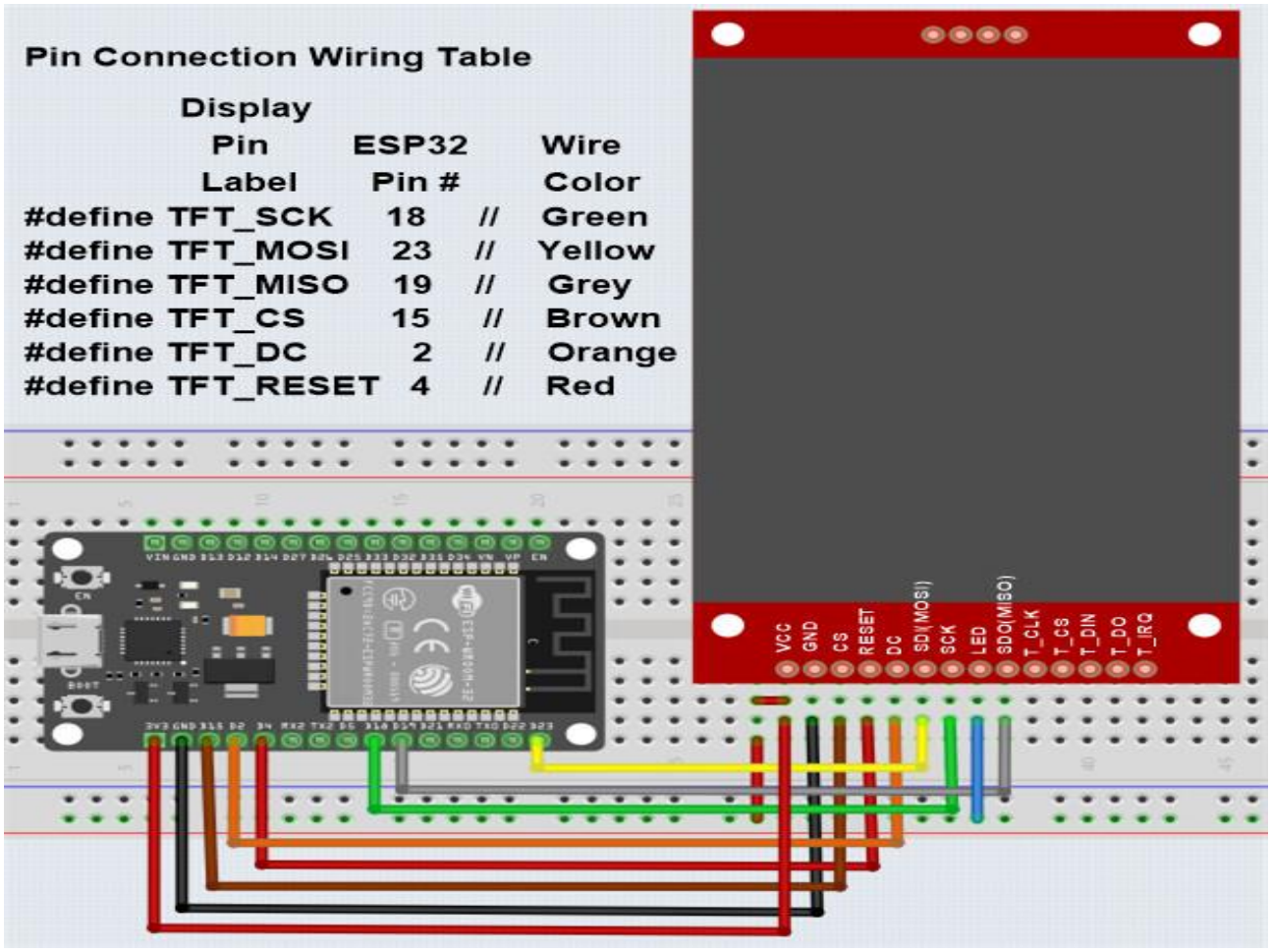
Display 3.5 TFT SPI 480X320 V1.0 esp32

1.Installation 3.5 TFT SPI 480X320 V1.0: The TFT_eSPI by Bodmer library should first be installed under the library manager in the Arduino IDE.



2. Configuration 3.5 TFT SPI 480X320 V1.0

2.1 Driver

under the file **../Adruino/librairies/TFT_eSPI** find the **User.Setup.h** double click on the file.

There are several drivers and only one of them should be selected, #**define ILI9341_DRIVER** is used by default, but it didn't work for me.

Therefore, the default driver must be commented out and **#define ILI9486_DRIVER** selected as the driver, the others must be commented out.

Example of my User_Setup.h

```
// Only define one driver, the other ones must be commented out
//#define ILI9341_DRIVER      // Generic driver for common displays
//#define ILI9341_2_DRIVER    // Alternative ILI9341 driver, see https://github.com/Bodmer/TFT_eSPI/issues/1172
//#define ST7735_DRIVER      // Define additional parameters below for this display
//#define ILI9163_DRIVER     // Define additional parameters below for this display
//#define S6D02A1_DRIVER
//#define RPI_ILI9486_DRIVER // 20MHz maximum SPI
//#define HX8357D_DRIVER
//#define ILI9481_DRIVER // is working
#define ILI9486_DRIVER
//#define ILI9488_DRIVER     // WARNING: Do not connect ILI9488 display SDO to MISO if other devices share the SPI bus (TFT SDO does NOT tristate when CS is high)
//#define ST7789_DRIVER      // Full configuration option, define additional parameters below for this display
//#define ST7789_2_DRIVER    // Minimal configuration option, define additional parameters below for this display
//#define R61581_DRIVER
//#define RM68140_DRIVER
//#define ST7796_DRIVER
//#define SSD1351_DRIVER
//#define SSD1963_480_DRIVER
//#define SSD1963_800_DRIVER
//#define SSD1963_800ALT_DRIVER
//#define ILI9225_DRIVER
//#define GC9A01_DRIVER

```

2.2 pin numbers

These are selected by default, because we are working with esp32 we should comment them out and use the pin number of eps32

Step 1: comment the default out

```
// ###### EDIT THE PIN NUMBERS IN THE LINES FOLLOWING TO SUIT YOUR ESP8266 SETUP ######

// For NodeMCU - use pin numbers in the form PIN_Dx where Dx is the NodeMCU pin designation
//#define TFT_CS   PIN_D8  // Chip select control pin D8
//#define TFT_DC   PIN_D3  // Data Command control pin
//#define TFT_RST  PIN_D4  // Reset pin (could connect to NodeMCU RST, see next line)
//#define TFT_RST  -1    // Set TFT_RST to -1 if the display RESET is connected to NodeMCU RST or 3.3V
```

Step 2: remove comments for ESP32 Dev board

```
203 // For ESP32 Dev board (only tested with ILI9341 display)
204 // The hardware SPI can be mapped to any pins
205
206 #define TFT_MISO 19
207 #define TFT_MOSI 23
208 #define TFT_SCLK 18
209 #define TFT_CS   15  // Chip select control pin
210 #define TFT_DC    2  // Data Command control pin
211 #define TFT_RST   4  // Reset pin (could connect to RST pin)
212
213 //#define TFT_RST  -1  // Set TFT_RST to -1 if display RESET
```

Have fun