

Phần 3: Màn hình

Ok, đầu tiên mình muốn chọn lắp màn hình trước.

Màn hình là phần giúp hiển thị trực quan nhất. Mình có thể thấy được thành quả của mình hiển thị ngay trên màn hình chứ không phải chỉ ở mỗi `Serial.print`

Màn hình cũng giúp mình kiểm tra được những lỗi phần cứng mà chỉ dùng `Serial.print` thì không thể tìm ra hết được.

Lắp đặt

Màn hình	ESP32-C3	Chắc năng	Lý do	Ghi Chú
VCC	3.3V hoặc 5V	Cấp nguồn cho màn hình.	Nguồn vào từ USB qua ESP32 có thể đẩy ra thành 2 nguồn 3.3V hoặc 5V.	Với những thiết bị nhẹ điện. Có thể sử dụng nguồn từ ESP32. Chọn nguồn phù hợp với thiết bị
GND	GND	Nối đất	Tạo một điểm tham chiếu chung cho toàn bộ thiết bị trên hệ thống	Toàn bộ thiết bị đều cần kết nối GND
SCK	GPIO4	Clock (SPI)	Màn hình sử dụng giao thức SPI nên cần sử dụng chân clock GPIO4 là chân clock của mạch ESP32-C3 này	Chân clock có thể thay đổi với những board mạch khác nhau. Kiểm tra datasheet để biết thêm chi tiết.
SDA	GPIO6	Data out (MOSI)	Màn hình nhận dữ liệu từ cổng ra của ESP32	MOSI cũng là một chân đặc biệt trên ESP32. Chú ý kiểm tra datasheet
RES	GPIO1	Reset	Là chân điều khiển không yêu cầu tốc độ cao. Chọn bất kỳ chân GPIO còn trống thuận tiện cho đi dây	Tránh chọn các chân đặc biệt Cần chú ý khai báo chính xác trong code
DC	GPIO2	Data/Command	Tương tự như RES. Có thể chọn chân GPIO thuận tiện	

BLK	GPIO3	Điều khiển nền của màn hình	Tương tự như RES và DC. Có thể chọn chân thuận tiện cho việc đi dây	
-----	-------	-----------------------------	---	--

Kiểm tra

Cài đặt thư viện TFT_eSPI

“ Arduino and PlatformIO IDE compatible TFT library optimised for the Raspberry Pi Pico (RP2040), STM32, ESP8266 and ESP32 that supports different driver chips

Để thêm thư viện chúng ta có thể thực hiện 1 trong 2 cách sau:

- Truy cập PIO Home → Library → Search TFT_eSPI và cài đặt library TFT_eSPI của Bodmer
- Hoặc thêm `bodmer/TFT_eSPI` vào phần `lib_deps` trong file `platformio.ini`

Khi đó file `platformio.ini` sẽ trông như thế này

```
[env:esp32-c3-devkitm-1]
platform = platformio/espressif32 @ 6.6.0
board = esp32-c3-devkitm-1
framework = arduino
platform_packages =
    framework-arduinoespressif32 @ ~3.20014.0
lib_deps =
    SPI
    bodmer/TFT_eSPI@^2.5.43
monitor_speed = 115200
build_flags =
    -D ARDUINO_USB_MODE=1
    -D ARDUINO_USB_CDC_ON_BOOT=1
```

Tiếp theo cần cấu hình thư viện để nó tương thích với đúng phần cứng.

Truy cập `.pio/libdeps/esp32-c3-devkitm-1/TFT_eSPI/User_Setup.h` và thực hiện config tương ứng với cấu hình phần cứng.

```
// Driver của màn hình
#define ST7789_DRIVER
#define ESP32_C3_DMA_ERROR_WORKAROUND
#define TFT_RGB_ORDER TFT_RGB // Nếu màu in ra bị sai Blue và Red thì đổi lại thành TFT_BGR

// Kích thước của màn hình
#define TFT_WIDTH 240
#define TFT_HEIGHT 240

// Các chân kết nối đang sử dụng
#define TFT_MOSI 6
#define TFT_SCLK 4
#define TFT_CS -1
```

```

#define TFT_DC    2
#define TFT_RST   1
#define TFT_BL    3

// Tín hiệu b? t? t ? ?n n?n
#define TFT_BACKLIGHT_ON HIGH

// Các font s? ???c compile
#define LOAD_GLCD    // Font 1 (built-in), required for basic text
#define LOAD_FONT2    // Small 16 px font
#define LOAD_FONT4    // Medium 26 px font
#define LOAD_FONT6
#define LOAD_FONT7
#define LOAD_FONT8
#define LOAD_GFXFF    // FreeFonts (optional)
#define SMOOTH_FONT // Anti-aliased fonts (needs FS)

// T?c ?? SPI
#define SPI_FREQUENCY    40000000
#define SPI_READ_FREQUENCY 20000000

```

Cách này có điểm yếu là sẽ bị **reset** khi Full Clean hoặc Update/Re-install thư viện. Nhưng trong quá trình test các các setup khác đều gặp lỗi bootloop do config sai. Nên hiện tại vẫn sử dụng cách này để setup.

"Hello World!"

Tiếp theo để test xem màn hình đã được setup đúng và phần cứng còn hoạt động tốt hay không. Chúng ta sẽ thử với đoạn code hiển thị "Hello world!" trên màn hình.

Truy cập `src ? main.cpp` và sử dụng đoạn code dưới đây:

```

#include <Arduino.h>
#include <TFT_eSPI.h>

// T?o ??i t??ng tft t? th? vi?n
TFT_eSPI tft = TFT_eSPI();

void setup() {
    tft.init(); // Kh?i t?i tft controller

    tft.setRotation(0); // Set h??ng xoay màn hình // S? d?ng các tham s? 0, 1, 2, 3
    tft.fillScreen(TFT_BLACK); // Fill toàn b? màn hình b?ng màu ?en
    tft.setTextColor(TFT_GREEN); // Set màu ch?
    tft.setTextSize(2); // Set c? ch?
    tft.setCursor(0, 0); // Set t?a ?? ?i?m b?t ??u vi?t ch?
    tft.println("Hello World!"); // Hi?n th? ch?
}

void loop() {}

```

Sau đó thực hiện Upload. Quá trình Upload có thể có warning (console màu vàng) nhưng không ảnh hưởng gì.

Sau khi upload xong kiểm tra màn hình nếu hiển thị như hình dưới là đã chạy **thành công** sẵn sàng cho bước tiếp theo thôi.



Trong trường hợp chưa hiển thị được "Hello World!" trên màn hình hãy kiểm tra cửa sổ terminal trên VSCode:

- Nếu **có** log chạy liên tục thì rất có thể bạn bị bootloop.
- Nếu **không** có hãy nhấn nút **Reset** trên ESP32 để ép ESP32 khởi động lại.

Nếu vẫn không được thực hiện check list sau:

- Kiểm tra các chân kết nối đảm bảo không chân nào bị lỏng.
- Kiểm tra lại file `User_Setup.h` đảm bảo đã config đúng file, đúng tham số.
- Đảm bảo đã khai báo đúng và đủ file `platformio.ini`
- Thực hiện Clean và Upload lại.

Nếu vẫn không được nữa. Thì nên thử với một cái màn hình khác để đảm bảo phần cứng không bị hỏng.

Revision #3
Created 2025-10-11 12:02:09 UTC by ThanhDV
Updated 2025-10-26 07:00:00 UTC by ThanhDV