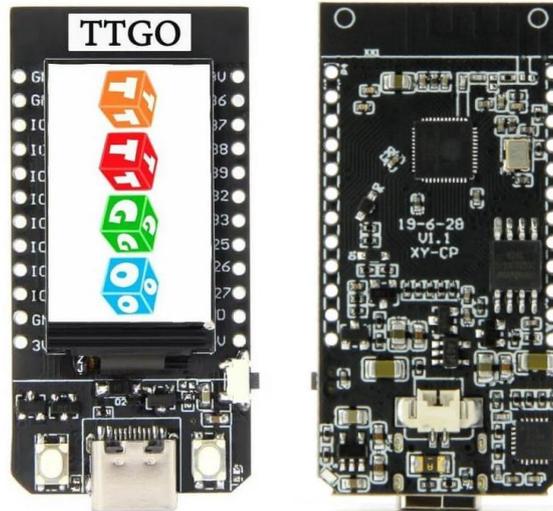


REF: B11-TTGO-ESP32

## TGO T-Display ESP32 Wi-Fi Bluetooth module 1.14-inch LCD development board



### Description

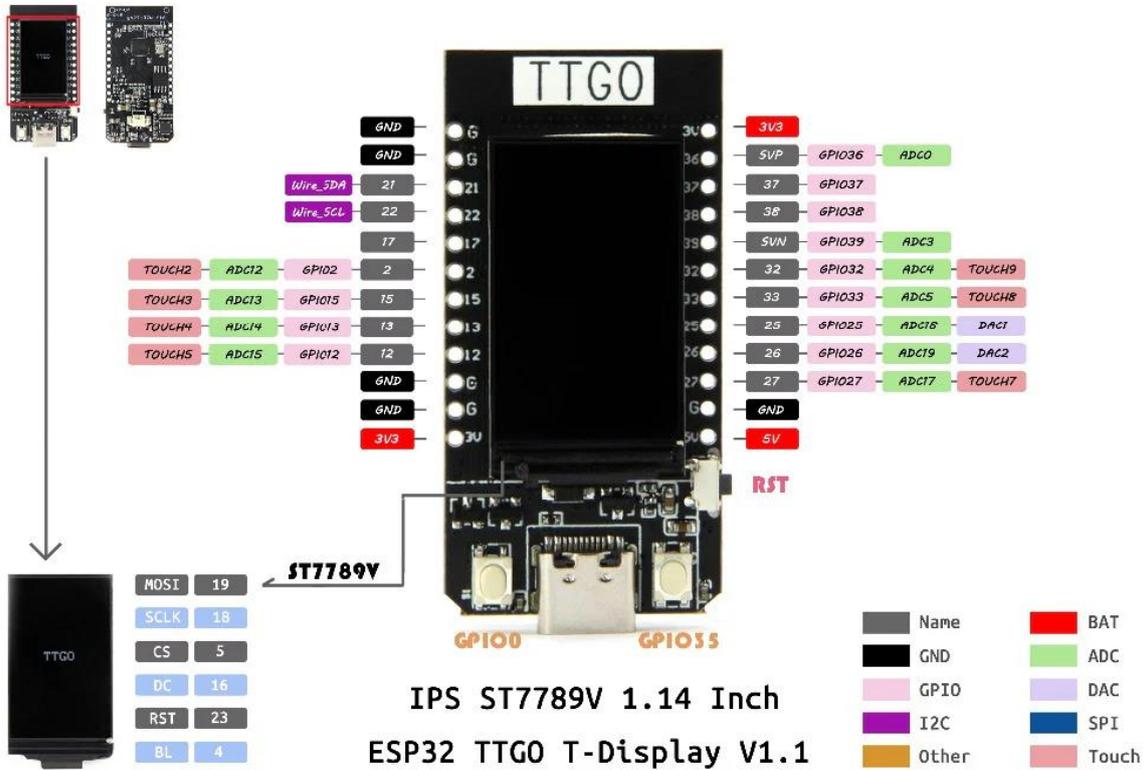
The TTGO T-Display is a development board based on the ESP32 that includes a 1.14-inch colour display. So, on this small board it has Wi-Fi, Bluetooth, MCU, Display. If you want to start an IoT project, this is a good one to start with. The OLED display on board is a 135x240 pixels TFT IPS display based on ST7789 driver IC and connects to ESP32's SPI interface. On its back side you can connect a li battery as a power supply.

### Specifications

- Chipset: ESPRESSIF-ESP32 240MHz Xtensa® single-/dual-core 32-bit LX6 microprocessor
- FLASH Memory: QSPI flash 4MB
- SRAM: 520 KB SRAM
- Button: Reset
- USB to TTL IC: CP2104
- Modular interface: UART, SPI, SDIO, I2C, LED PWM, TV PWM, I2S, IRGPIO, ADC, capacitor touch sensor, DACLNA pre-amplifier
- Display: IPS ST7789V 1.14 Inch
- Working Voltage: 2.7V-4.2V
- Working Current: About 60MA
- Sleep Current: about 120uA
- Working temperature range: -40°C ~ +85°C
- Size & Weight: 51.5 x 25.3 x 8.5mm (7.8g)

## Pin diagram

The manufacturer provides this information about the board. We will use the pinout for the ST7789 screen. The screen is 135 pixels wide and 240 pixels in height.



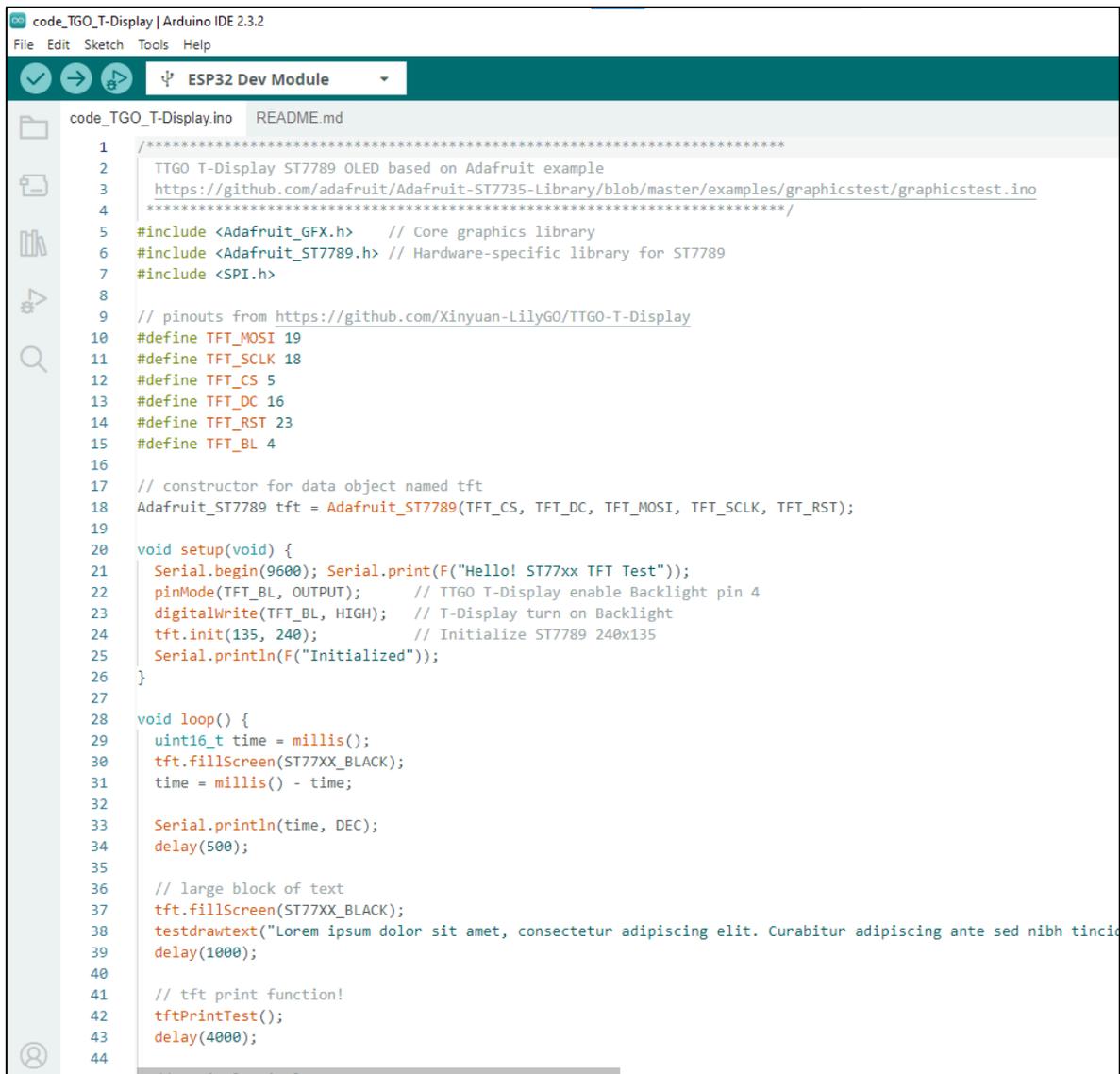
## Library

Install the library below as image shown below.

<p><b>Adafruit GFX Library</b> by Adafruit</p> <p>1.11.11 installed</p> <p>Adafruit GFX graphics core library, this is the 'core' class that all our other graphics libraries derive from. Install this library in...</p> <p><a href="#">More info</a></p> <p>1.11.11 <input type="button" value="REMOVE"/></p>	<p><b>Adafruit ST7735 and ST7789 Library</b> by Adafruit</p> <p>1.11.0 installed</p> <p>This is a library for the Adafruit ST7735, ST7789, ST7796S SPI displays. This is a library for the Adafruit ST7735, ST7789, ST7796S SPI...</p> <p><a href="#">More info</a></p> <p>1.11.0 <input type="button" value="REMOVE"/></p>
---	---

## CODING

Adafruit libraries are included that work with the ST7789 display. You can look in the documents at methods that make shapes work on the 135x240 pixel screen. Pins are defined according to the diagram from the vendor. A constructor uses these definitions to toggle the physical pins for chip select, data/command, mosi, clock and reset. Setup turns on pin 4 backlight lamp for the screen. Loop will repeat the patterns.



```
code_TGO_T-Display | Arduino IDE 2.3.2
File Edit Sketch Tools Help
ESP32 Dev Module
code_TGO_T-Display.ino README.md
1 /*****
2 TTGO T-Display ST7789 OLED based on Adafruit example
3 https://github.com/adafruit/Adafruit-ST7735-Library/blob/master/examples/graphicstest/graphicstest.ino
4 *****/
5 #include <Adafruit_GFX.h> // Core graphics library
6 #include <Adafruit_ST7789.h> // Hardware-specific library for ST7789
7 #include <SPI.h>
8
9 // pinouts from https://github.com/Xinyuan-LilyGO/TTGO-T-Display
10 #define TFT_MOSI 19
11 #define TFT_SCLK 18
12 #define TFT_CS 5
13 #define TFT_DC 16
14 #define TFT_RST 23
15 #define TFT_BL 4
16
17 // constructor for data object named tft
18 Adafruit_ST7789 tft = Adafruit_ST7789(TFT_CS, TFT_DC, TFT_MOSI, TFT_SCLK, TFT_RST);
19
20 void setup(void) {
21   Serial.begin(9600); Serial.print(F("Hello! ST77xx TFT Test"));
22   pinMode(TFT_BL, OUTPUT); // TTGO T-Display enable Backlight pin 4
23   digitalWrite(TFT_BL, HIGH); // T-Display turn on Backlight
24   tft.init(135, 240); // Initialize ST7789 240x135
25   Serial.println(F("Initialized"));
26 }
27
28 void loop() {
29   uint16_t time = millis();
30   tft.fillRect(ST77XX_BLACK);
31   time = millis() - time;
32
33   Serial.println(time, DEC);
34   delay(500);
35
36   // large block of text
37   tft.fillRect(ST77XX_BLACK);
38   testdrawtext("Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur adipiscing ante sed nibh tincidunt.",
39               10, 10, 100, 100);
40   delay(1000);
41
42   // tft print function!
43   tftPrintTest();
44   delay(4000);
45 }
```

## Result

Here's what the output looks like, the OLED display will start displaying pattern.

