

ESP32-C3 OLED Pluggable Terminal Base

Two blocks of eight push-in terminals for the sixteen holes, a battery socket, a charger and a power switch.

✗ Seat the board the right way round, cable out

The board fits end for end as well, and that way its 5V, ground and 3V3 holes land on the base's 3, 4 and 5 ways: the battery rail drives three chip pins. Put V5 on the board and 5V on the base at the same end, with the switch off and the USB cable out.

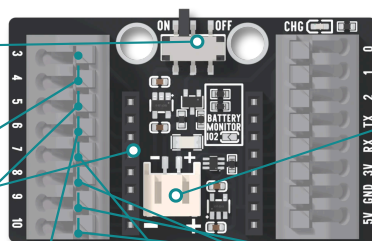
Left block, top to bottom

Slide switch

ON and OFF either side.
OFF takes the cell out of the charger as well.

Socket for the board

Antenna at the switch end.
Match V5 to 5V at one end.



Battery socket

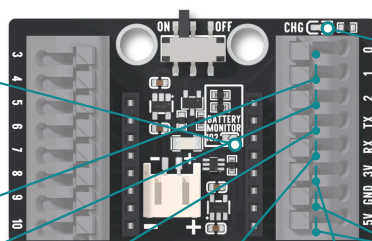
JST PH 2.0. The + on the silkscreen is the positive pin. Check the lead.

	1	2	3	4	5	6	7	8	9	10
Front	3	4	5	6	7	8	9	10		
Digital, analog	GPIO 3	GPIO 4			GPIO 7					GPIO 10
I2C with display			SDA	SCL						
Serial, UART0										

Right block, top to bottom

BATTERY MONITOR

Open jumper. Bridged, it joins half the cell voltage to way 2 for good.



CHG LED

Steady while the cell charges, switch ON.

	1	2	3	4	5	6	7	8
Front	0	1	2	TX	RX	3V	GND	5V
Digital, analog	GPIO 0	GPIO 1	GPIO 2			3V	GND	5V
I2C with display						3V	GND	5V
Serial, UART0				TX, GPIO 21	RX, GPIO 20	3V	GND	5V

The left block has no power

3V, GND and 5V are one way each, all on the right block. A module wired to the left block takes its supply and its ground from there, and a way holds one wire. Ways 5 and 6 are the display's bus; 8 is the blue LED, 9 the BOO button, TX and RX the serial port. A cell needs its own protection circuit, and charging needs the switch ON.

The first program, an LED and a button

Arduino IDE · Board ESP32C3 Dev Module · USB CDC On Boot Enabled · Flash 4MB · an LED, a button, three wires

```
// ESP32-C3 OLED, pluggable base: an LED and a button.
// Wiring is in the table below.
//
// Arduino IDE, Tools menu:
//   Board           ESP32C3 Dev Module
//   USB CDC On Boot Enabled
//   Flash Size      4MB (32Mb)
//   Upload Speed    921600
// No library needed.

const int LED = 0;
const int BUTTON = 1;

void setup() {
  Serial.begin(115200);
  pinMode(LED, OUTPUT);
  pinMode(BUTTON, INPUT_PULLUP); // the pin idles HIGH
  delay(1500); // let the USB port enrol
  Serial.println("LED on GPIO 0, button on GPIO 1");
}

void loop() {
  bool pressed = digitalRead(BUTTON) == LOW;
  bool blink = (millis() / 500) % 2; // flips every half second
  digitalWrite(LED, pressed || blink);
}
```

LED on GPIO 0, button on GPIO 1
Printed once at start-up. The LED then blinks once a second, and stays on while the button is held.

The three wires

WAY	WHERE	WHAT	GOES TO
0	right, 1st down	GPIO 0	330 ohm, then the LED's long leg
1	right, 2nd down	GPIO 1	one side of the button
GND	right, 7th down	ground	a breadboard ground rail

When it does not work

The wire will not go in, or pulls straight out

A push-in way holds a bare conductor, so the stripped length matters: the maker's datasheet gives 8 to 9 mm. Shorter leaves insulation under the spring; longer leaves copper showing next to the neighbouring way, 3.5 mm away. It lists 22 to 18 AWG wire, so a thinner jumper wire may not be held. Tug each wire after it goes in; if it moves, it never seated. To free a wire, press the grey tab on the way and pull.

A wire in the 8 or the 9 way and the board will not upload

Those ways are the blue LED and the BOO button. Manual download mode needs GPIO 8 high and GPIO 9 low at reset, and a wire that holds either one the wrong way at that moment stops it. Take the wire out, upload, put it back.

The LED never lights

Turn the LED round: it only conducts one way, and the long leg goes toward the resistor and the GPIO 0 way. Then check that the resistor shares a breadboard row with the wire from the way, and that the LED's other leg shares a row with the ground rail. If both are right, tug each wire in the base: a wire that moves never seated.

A sensor on the left block has no ground

The left block is GPIO 3 to 10 and nothing else: there is no 3V3 and no GND on it. Ground and 3V3 are on the right block, one way each. Run a wire from the GND way to a breadboard ground rail and take every ground from there, because a way takes one wire.

The 5V way reads nothing with only a battery connected

It is the USB bus voltage, not a regulator output. Nothing on the base raises the cell to 5 V, so on battery alone that way should read nothing. Power a 5 V load from USB, or use the 3V way for a 3.3 V one.

The Serial Monitor is empty

The sketch is running if the LED blinks. Set USB CDC On Boot to Enabled in the Tools menu and upload again, then open the monitor at 115200. Left disabled, Serial.println goes nowhere. The board's USB port also disappears and returns when the chip resets, so reopen the monitor after pressing RST.

Specifications

Fits	ESP32-C3 OLED board, rows 17.78 mm apart	Terminals	2 × 8-way, 3.5 mm pitch, push-in
Wire	22 to 18 AWG, stripped 8 to 9 mm	GPIOs at the edge	13, each once
Power ways	3V, GND, 5V, right block only	Battery	JST PH 2.0, one 3.7 V cell with protection
Charger	TP4054, to 4.2 V, switch ON	Size	53.13 × 32.54 mm, 13.31 mm tall