QUICK REFERENCE · TK03

Traffic Light

Red, yellow and green, one pin each.

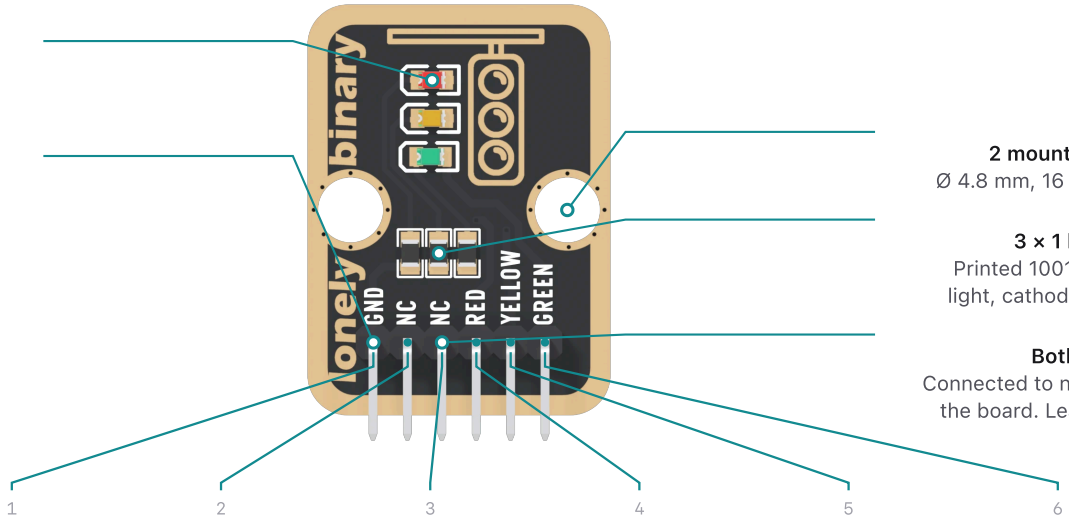
Keep the resistors Each light's 1 kΩ resistor is the only thing limiting its current. **Never bridge them**: without one, a pin runs far past its rating and fails, sometimes weeks later.

Six pins, four wires

LEDs up, header at the bottom. Wire GND, RED, YELLOW and GREEN; the two NC holes are connected to nothing on the board.

Red, yellow, green
Three 0805 LEDs, one header pin each.

GND is the square pad
Count from it. The back prints no pin names.



2 mounting holes
Ø 4.8 mm, 16 mm apart

3 × 1 kΩ, fitted
Printed 1001. One per light, cathode to GND.

Both NC pins
Connected to nothing on the board. Leave them.

Front	GND	NC	NC	RED	YELLOW	GREEN
On / off	GND	●	●	RED	YELLOW	GREEN

A sequence ≠ a state delay() lights the right colours and stops everything else. Keep the phase in a variable and watch millis().

Three lights, three currents

Red

3.1 mA from a 5 V pin

1.4 mA from 3.3 V. Forward voltage 1.5–2.6 V.

Yellow

3.1 mA from a 5 V pin

1.4 mA from 3.3 V. Forward voltage 1.8–2.4 V.

Green

2.3 mA from a 5 V pin

0.6 mA from 3.3 V. Forward voltage 2.6–3.1 V.

The green needs almost a volt more than red or yellow, so it gets the least current, and on a 3.3 V board less than half of theirs. Its datasheet rates it far brighter per milliamp, so it still reads as lit. All three are well inside what a pin may supply, together or apart.

Currents are worked out, not measured: the pin voltage less each LED's forward voltage at a few mA (red and yellow 1.9 V, green 2.7 V), over 1 kΩ.

A light that remembers

Arduino IDE · Board: your board · ESP32-S3: USB CDC On Boot: Enabled

```
// Uno: 9, 10, 11. ESP32: 25, 26, 27. S3: 4, 5, 6. Pico: 13–15.
const int RED_PIN = 4, YELLOW_PIN = 5, GREEN_PIN = 6;

struct Phase { bool red, yellow, green; unsigned long ms; };
const Phase PHASES[] = {        // one row per phase, in order
  { true,  false, false, 5000 },   // red
  { false, false, true,  5000 },   // green
  { false, true,  false, 2000 },   // yellow
};
int phase = 0;
unsigned long phaseStart = 0;

void show(int p) {
  digitalWrite(RED_PIN, PHASES[p].red ? HIGH : LOW);
  digitalWrite(YELLOW_PIN, PHASES[p].yellow ? HIGH : LOW);
  digitalWrite(GREEN_PIN, PHASES[p].green ? HIGH : LOW);
}

void setup() {
  pinMode(RED_PIN, OUTPUT);
  pinMode(YELLOW_PIN, OUTPUT);
  pinMode(GREEN_PIN, OUTPUT);
  show(phase);
  phaseStart = millis();
}

void loop() {
  if (millis() - phaseStart >= PHASES[phase].ms) {
    phase = (phase + 1) % 3;
    show(phase);
    phaseStart = millis();
  }
  // free to read a button or the serial port here
}
```

Wiring

	FRONT	UNO	ESP32	S3	PICO
1	GND	GND	GND	GND	GND
2	NC				
3	NC				
4	RED	D9	GPIO 25	GPIO 4	GP13
5	YELLOW	D10	GPIO 26	GPIO 5	GP14
6	GREEN	D11	GPIO 27	GPIO 6	GP15

Hole 1 is the square pad. The sketch runs red 5 s, green 5 s, yellow 2 s, one common pattern; add a red-and-yellow row for the UK order. On a classic ESP32 never use GPIO 6 to 11: they belong to the flash chip.

delay() or millis()

A sequence of digitalWrite and delay() lights correctly and does nothing else: while it waits, a button press or a serial command goes unanswered. The sketch above keeps the phase in a variable and checks the clock, so loop() comes round thousands of times a second.

When it does not work

One light never comes on

That wire is on a different pin from the sketch, or one hole over: count from the square pad.

Nothing lights at all

GND is not connected. The three LEDs share it and have no path home without it.

A button is ignored

The sketch is inside a delay(). Use the millis() version and read the button in loop().

The lights are in the wrong order

RED, YELLOW and GREEN are crossed between the header and the board's pins. Check each against the table.

The green is dimmer on 3.3 V

Normal: it needs about 2.7 V, so a 3.3 V pin leaves it little. Red and yellow keep more headroom.

A light stays on after a reset

Set every pin in setup(), or show() the first phase before loop() runs.

Specifications

LEDs	Red, yellow, green, 0805	**Resistors**	3 × 1 kΩ, one per light, fitted
At 5 V	Red 3.1, Yellow 3.1, Green 2.3 mA	**At 3.3 V**	Red 1.4, Yellow 1.4, Green 0.6 mA
Logic	Active high, one pin per light	**Header**	6 pins, right-angle, 2.54 mm
Size	22.4 × 30.4 mm	**Mounting**	2 × Ø 4.8 mm, 16 mm apart