QUICK REFERENCE · TK01

XL LED

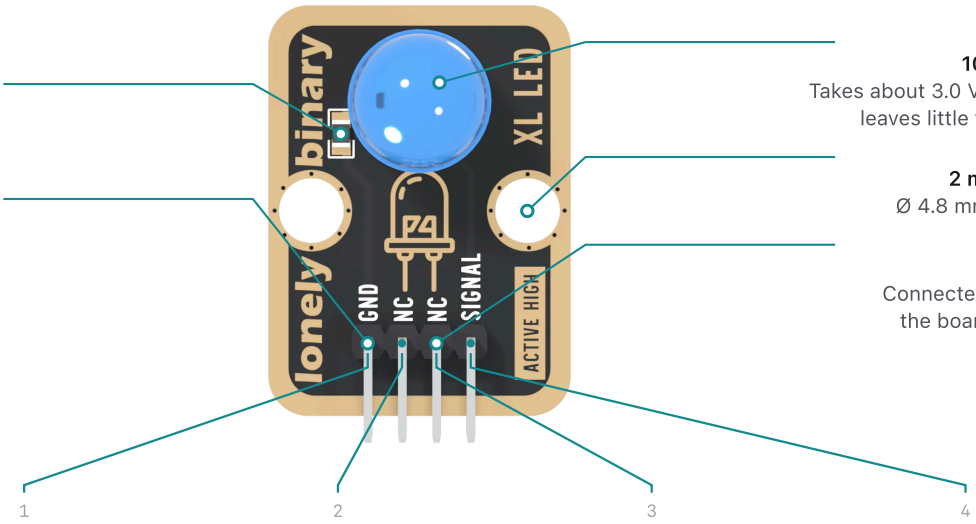
One blue LED, its resistor fitted, driven by one pin.

Keep the resistor

The 150 Ω resistor on the board is the only thing limiting the current. Without it a pin runs far past its rating and fails, sometimes weeks later. **Never wire a bare LED to a pin**, and never bridge or replace this resistor with a lower value.

Four pins, two wires

LED side up, header at the bottom. Wire GND and SIGNAL; the two NC holes are connected to nothing on the board.



150 Ω resistor, fitted
Printed 1500. It keeps the pin inside its rating. Add none.

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

10 mm blue LED
Takes about 3.0 V, so a 3.3 V pin leaves little for the resistor.

2 mounting holes
Ø 4.8 mm, 16 mm apart

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

Front	GND	NC	NC	SIGNAL
On / off	GND	●	●	SIGNAL
Brightness	GND	●	●	SIGNAL

Dim on 3.3 V ≠ faulty

There is no VCC pin: the pin driving SIGNAL is the LED's power. From 3.3 V the LED gets about 2 mA, a sixth of what a 5 V board gives. Plainly lit, and correct.

How bright, on which board

Arduino Uno

13 mA from a 5 V pin

SIGNAL to D9. PWM at 490 Hz.

ESP32-S3

2 mA from a 3.3 V pin

SIGNAL to GPIO 4. PWM at 1000 Hz.

Raspberry Pi Pico

2 mA from a 3.3 V pin

SIGNAL to GP15. PWM at 1000 Hz.

The LED takes about 3.0 V whatever the pin gives, and only the rest drives current through the 150 Ω resistor: 2.0 V from a 5 V pin, 0.3 V from a 3.3 V one. A third less voltage gives about a sixth of the current. It is dimmer on an ESP32 or a Pico, and still plainly lit.

Current is the voltage measured across the resistor on the bench, divided by 150 Ω: 2.0 V with SIGNAL at 5 V, 0.3 V with SIGNAL at 3.3 V. One board; the LED has no datasheet, so treat both as approximate. The ESP32 behaves as the ESP32-S3.

The first blink, no library

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

```
// SIGNAL's GPIO number. Uno: 9. ESP32, ESP32-S3: 4. Pico: 15.
const int LED_PIN = 4;

void setup() {
  Serial.begin(115200);
  pinMode(LED_PIN, OUTPUT);   // without it, HIGH is a weak pull-up
}

void loop() {
  digitalWrite(LED_PIN, HIGH);  // the pin supplies the current
  Serial.println("on");
  delay(500);

  digitalWrite(LED_PIN, LOW);   // 0 V on the anode: nothing flows
  Serial.println("off");
  delay(500);
}
```

```
on
off
on
```

Serial Monitor at 115200, one line every half second, and the LED blinking with it.

Wiring

	FRONT	UNO	ESP32, S3	PICO	WIRE
1	GND	GND	GND	GND	✓
2	NC				–
3	NC				–
4	SIGNAL	D9	GPIO 4	GP15	✓

Hole 1 is the square pad. Both NC holes are connected to nothing, so wiring them changes nothing. Every pin in the table can also do PWM. On an Uno only 3, 5, 6, 9, 10 and 11 can; on the ESP32 boards and the Pico nearly any output pin can. Keep an ESP32 off its strapping pins.

Brightness is a duty cycle

```
// Brightness: 0 is off, 255 is fully on. A PWM pin only.
analogWrite(LED_PIN, 64);     // a quarter of the time HIGH

// A fade that looks even: step through
// 255 * (i / 31) ^ 2.2 rather than 0, 8, 16 ...
```

analogWrite switches the pin fully on and off, 490 times a second on an Uno's pin 9 and 1000 on the ESP32 and Pico cores, and the LED shows the average. It can only make the LED dimmer than digitalWrite HIGH, never brighter. Equal steps of duty do not look like equal steps of light; send the 2.2 power curve instead.

When it does not work

It never lights

SIGNAL is not on the pin the sketch names: LED_PIN is the GPIO number printed beside the pin. Then GND. Then count from the square pad.

It is barely visible on an Uno

pinMode(LED_PIN, OUTPUT) is missing, so HIGH only switches on the pin's weak internal pull-up.

analogWrite does nothing

Not a PWM pin. On an Uno use 3, 5, 6, 9, 10 or 11. On an ESP32 use board package 2.0 or later.

It is dim on an ESP32 or a Pico

Normal. About 2 mA from 3.3 V against 13 mA from 5 V. For more light, use a 5 V board or switch 5 V through a transistor.

It will not turn off

The pin is held HIGH by something else. A serial TX pin idles HIGH; a supply pin is always on. Move SIGNAL to a plain GPIO.

It flickers

A pin lifting out of the breadboard row first. On an Uno, D0 and D1 flicker with serial data, and D13 flashes after a reset.

Specifications

LED	10 mm, blue, through-hole, 12.5 mm tall	**Resistor**	150 Ω, 0805, printed 1500, fitted
Current	About 13 mA from 5 V, about 2 mA from 3.3 V	**Logic**	Active high: HIGH lights it
Supply	None of its own. The SIGNAL pin powers the LED	**Header**	4 pins, right-angle, 2.54 mm: GND, NC, NC, SIGNAL
Size	22.4 × 30.4 mm	**Mounting**	2 × Ø 4.8 mm holes, 16 mm apart