

Door Sensor

A magnet near the sensor reads HIGH. Set the pull-down in setup: the board has none.



VCC is your board's logic voltage

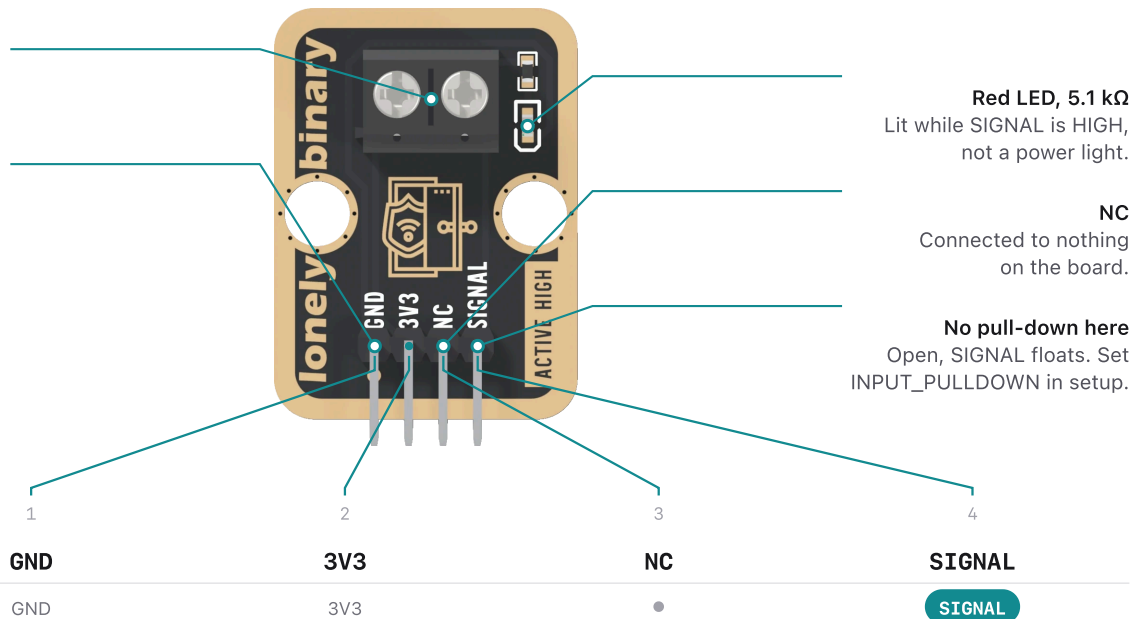
A closed sensor joins SIGNAL straight to VCC, so SIGNAL sees whatever VCC is. **On an ESP32, ESP32-S3 or Pico wire VCC to 3V3:** 5 V there puts 5 V on a 3.3 V pin each time the door closes. Only an Uno takes 5V. The header prints 3V3, and that is a label, not a rule.

Pins, and which to wire

Parts up, header at the bottom. Wire GND, VCC and SIGNAL; NC is connected to nothing on the board. The sensor's two wires go under the screws, either way round.

Screw terminal, 2 way
5.0 mm pitch. Wires go in either way: no polarity.

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



The LED is not a power light

It hangs off SIGNAL, so it is dark with the door open however well the block is powered. And with the door open nothing pulls SIGNAL anywhere, so a pin with no pull-down reads HIGH or LOW from one read to the next. Set one in setup(), or on an Uno fit 10 kΩ from SIGNAL to GND. Never INPUT_PULLUP.

What it draws, and when

Open, sensor apart

0 mA

VCC reaches only the terminal and the header. Nothing on the block runs.

Closed, VCC on 3V3

0.3 mA, worked out

0.25 mA in the LED, 0.07 mA in an ESP32's own 45 kΩ.

Closed, VCC on 5V

1.1 mA, worked out

Uno only. 0.6 mA in the LED, 0.5 mA in the outside 10 kΩ.

None of it comes from your pin, which only reads the voltage. On a battery, hang the sensor so the state that lasts longest is the open one.

Arithmetic on the net list and a red LED's usual 2 V drop, not measured. The sensors' sizes and distances are not on this sheet: they have not been measured here, so watch the LED as you move the magnet. The sketch compiles for an ESP32-S3 and an Uno; the block has not been bench-tested for this sheet.

The first read, on an ESP32-S3

Arduino IDE · Board: ESP32S3 Dev Module · USB CDC On Boot: Enabled · No library

```
// The pin SIGNAL is wired to.
// Uno: 2. ESP32: 25. ESP32-S3: 4. Pico: 15.
const int DOOR_PIN = 4;

int last = -1; // nothing read yet

void setup() {
  Serial.begin(115200);

  // No pull-down on the board: set one. Never INPUT_PULLUP.
  #ifdef INPUT_PULLDOWN
    pinMode(DOOR_PIN, INPUT_PULLDOWN);
  #else
    // Uno or Nano: none inside. Fit 10k from SIGNAL to GND.
    pinMode(DOOR_PIN, INPUT);
  #endif
}

void loop() {
  int level = digitalRead(DOOR_PIN);

  if (level != last) {
    if (level == HIGH) {
      Serial.println("door closed (magnet near)");
    } else {
      Serial.println("door open");
    }
    last = level;
  }

  delay(10);
}
```

door open
door closed (magnet near)
Serial Monitor at 115200, one line each time the door changes, with the red LED lit while it is closed.

An Uno has no pull-down, and a door that chatters

An Uno or Nano has no internal pull-down: fit 10 kΩ from SIGNAL to GND, VCC to 5V, and read with INPUT. ESP32 GPIO 34 to 39 have no internal pull-down either: use another pin. A change that has not held for 50 ms is a wire or a contact settling, so count it only after that.

When it does not work

LED dark with the wires touched

Wire caught by its insulation, not its copper, or VCC or GND is off.
Count from the square pad.

Random HIGH and LOW, door open

The pull-down is missing. Check setup(), and on an Uno the outside 10 kΩ.

Serial Monitor empty on an S3

Set USB CDC On Boot to Enabled, then upload again.

Specifications

Type	Screw terminal for a reed sensor	Terminal	2-way, 5 mm pitch, no polarity
SIGNAL	Closed sensor puts VCC on it. Floats when open	Pull-down	None on the board: set one
Indicator	Red LED, 5.1 kΩ, lit while closed	Supply	Your board's logic voltage
Current	0 open; 0.3 mA closed at 3.3 V	Board	22.4 × 30.4 mm, 4-pin right-angle header

Wiring

	FRONT	S3	UNO	ESP32	PICO
1	GND	GND	GND	GND	GND
2	3V3	3V3	5V	3V3	3V3
3	NC				
4	SIGNAL	GPIO 4	D2	GPIO 25	GP15

Hole 1 is the square pad. VCC is each board's own logic voltage: 3V3 on the ESP32 boards and the Pico, 5V on the Uno. The pins are the Push Button and Reed Switch blocks' pins.