QUICK REFERENCE · TK12

NTC Thermistor

A resistance that falls as it warms, as a voltage.

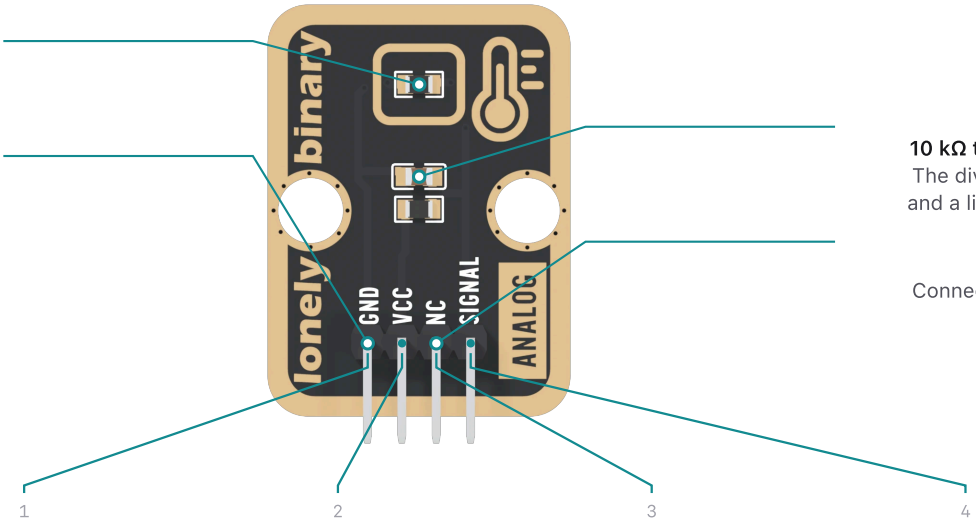
VCC sets the signal SIGNAL is a fraction of VCC, and above half of it below 25 °C. **On an ESP32, ESP32-S3 or Pico, feed VCC from 3V3, never 5V**: from 5 V the signal passes 3.3 V below about 11 °C.

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 thermistor
10 kΩ at 25 °C, B 3950. Touch it and watch it fall.

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



10 kΩ to VCC, 100 nF
The divider's top half, and a little smoothing.

NC
Connected to nothing on the board.

Front	GND	VCC	NC	SIGNAL
Read it	GND	VCC	●	SIGNAL

Warmer = lower voltage The thermistor sits at the bottom of the divider: as it warms its resistance falls, and so does SIGNAL.

What it gives at three temperatures

0 °C

3.85 V from VCC 5 V

The thermistor is 34 kΩ. From 3.3 V: 2.54 V.

25 °C

2.50 V from VCC 5 V

The thermistor is 10 kΩ. From 3.3 V: 1.65 V.

50 °C

1.32 V from VCC 5 V

The thermistor is 3.6 kΩ. From 3.3 V: 0.87 V.

To get degrees back: resistance = 10 kΩ × reading / (full scale − reading), then the B equation. On an Uno the supply cancels out; on an ESP32 read millivolts and divide by 3300.

Values from the part number (10 kΩ ±1 %, B 3950), not a datasheet. Good to around a degree near room temperature. It draws at most 0.5 mA, too little to warm itself.

Read degrees

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

```
// SIGNAL's pin. Uno: A0. ESP32: 34. ESP32-S3: 4. Pico: 26.
const int NTC_PIN = A0;
const int ADC_MAX = 1023;        // Uno, Pico
const float R_FIXED = 10000.0;   // to VCC
const float R25 = 10000.0, B = 3950.0;

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

void loop() {
  float n = analogRead(NTC_PIN);
  float r = R_FIXED * n / (ADC_MAX - n);  // thermistor, ohms
  float k = 1.0 / (1.0 / 298.15 + log(r / R25) / B);
  Serial.print(k - 273.15, 1);
  Serial.println(" C");
  delay(1000);
}
```

```
22.4 C
22.4 C
24.9 C
```

Serial Monitor at 115200, for example; the last line is a fingertip on the thermistor.

Wiring

	FRONT	UNO	ESP32	S3	PICO
1	GND	GND	GND	GND	GND
2	VCC	5V	3V3	3V3	3V3
3	NC				
4	SIGNAL	A0	GPIO 34	GPIO 4	GP26

Hole 1 is the square pad. On an ESP32 or ESP32-S3 use an ADC1 pin, and compute from analogReadMilliVolts() with VCC 3300 mV.

A steadier reading

Average eight or sixteen readings before converting. For an alarm, switch on at one temperature and off a degree or two lower, so a reading on the edge does not flicker the output.

When it does not work

It reads a very low temperature

SIGNAL near full scale: GND is missing, or the wires are crossed. Count from the square pad.

It reads a very high temperature

SIGNAL near 0 V: VCC is missing, or SIGNAL is on a different pin from the sketch.

It goes the wrong way when warmed

The formula has the divider upside down. The thermistor is at the bottom: R = 10 kΩ × n / (max − n).

The ESP32 is a few degrees off

Its ADC is not ratiometric. Use analogReadMilliVolts() and VCC = 3300.

It reads above the room

It is measuring the board it sits on. Move it away from regulators and hands.

It is slow to follow

Normal: the part takes seconds to reach a changed temperature.

Specifications

Thermistor	10 kΩ at 25 °C, ±1 %, B 3950	**Divider**	10 kΩ to VCC, thermistor to GND
Filter	100 nF on SIGNAL	**At 25 °C**	SIGNAL is half of VCC
Supply	3.3 V or 5 V, the same as your board	**Current**	At most 0.5 mA from 5 V
Header	4 pins, right-angle, 2.54 mm	**Size**	22.4 × 30.4 mm