

DS18B20 mit Arduino auslesen

Vom Drei-Draht-Anschluss bis zum ersten Temperaturwert und Mehrsensorbetrieb.

DS18B20 Temperatur-Sensor Wasserdicht



MAKERSHOP ONLINE

www.makershop.de

<https://www.makershop.de/sensoren/temperatur/ds18b20-wasserdicht/>

Das lernst du in diesem Tutorial

Farbbelegung pruefen, Pull-up einsetzen und den Sensor per 1-Wire auslesen.



DIGITALER 1-WIRE-TEMPERATURFUEHLER

Wir verbinden die wasserdichte DS18B20-Sonde im stabilen Drei-Draht-Betrieb mit einem Arduino Uno.

- Etwa 1 m Anschlusskabel
- Vergossener Sensor in Edelstahluhse
- Mehrere Sensoren an einem Datenbus
- Programmierbare Aufloesung von 9 bis 12 Bit
- Kompatibel mit Arduino, ESP32, ESP8266 und Raspberry Pi

Zeitbedarf: etwa 15 Minuten

Du benoetigst einen Arduino Uno, die DS18B20-Sonde, Steckkabel und einen 4,7-kOhm-Widerstand fuer den Daten-Pull-up.

1. Sensor anschliessen



4,7-kOhm-Pull-up zwischen VDD und DQ

Bei 3,3-V-Boards den Pull-up an 3,3 V legen

Rotes Kabel - VDD	Arduino 5 V
Gelbes Kabel - DATA	Arduino Digitalpin 2
Schwarzes Kabel - GND	Arduino GND
Pull-up	4,7-kOhm-Widerstand zwischen 5 V und DATA

Vor dem Einschalten kontrollieren

Eine vertauschte Versorgung kann Sensor oder Controller beschadigen. Bei ESP32, ESP8266 und Raspberry Pi DATA mit Pull-up an 3,3 V betreiben.

2. Bibliotheken installieren

Bibliotheksverwalter	Arduino IDE oeffnen und Werkzeuge > Bibliotheken verwalten waehlen.
OneWire	Nach 'OneWire' von Paul Stoffregen suchen und installieren.
DallasTemperature	Nach 'DallasTemperature' von Miles Burton suchen und installieren.
Port	Board und seriellen Port in der Arduino IDE auswaehlen.

3. Ersten Messwert ausgeben

DS18B20 an Digitalpin 2

```
#include <OneWire.h>
#include <DallasTemperature.h>

#define ONE_WIRE_BUS 2

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

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

void loop() {
  sensors.requestTemperatures();
  float tempC = sensors.getTempCByIndex(0);

  if (tempC == DEVICE_DISCONNECTED_C) {
    Serial.println("Sensor nicht gefunden");
  } else {
    Serial.print("Temperatur: ");
    Serial.print(tempC, 2);
    Serial.println(" C");
  }

  delay(1000);
}
```

4. Messung pruefen

- Seriellen Monitor auf 115200 Baud stellen.
- Bei 12 Bit kann die Wandlung bis zu 750 ms dauern.
- Der Fehlerwert -127 Grad C weist meist auf Verbindung oder Pull-up hin.
- 85 Grad C direkt nach dem Start kann ein noch nicht aktualisierter Einschaltwert sein.
- Sonde und Vergleichsthermometer vor dem Vergleich ausreichend angleichen lassen.

Mehrere Sensoren am selben Bus

Jeder DS18B20 besitzt eine eindeutige 64-Bit-Adresse. Dadurch lassen sich mehrere Fuehler parallel an derselben Datenleitung betreiben. Fuer eine feste Zuordnung die Adressen einmal auslesen und im Sketch speichern.

Bus statt Sternverkabelung

Bei laengeren Leitungen kurze Abzweige und eine saubere gemeinsame Masse bevorzugen. Anzahl, Leitung und Pull-up immer am realen Aufbau testen.

Typische Fehler schnell loesen

-127 Grad C	Kabelbelegung, gemeinsamen GND und 4,7-kOhm-Pull-up pruefen.
85 Grad C	Nach requestTemperatures auf abgeschlossene Wandlung warten.
Sensor fehlt	Richtiges GPIO im Sketch und OneWire-Bibliothek kontrollieren.
Unruhige Werte	Leitungsfuehrung, Versorgung und Bus-Topologie verbessern.
ESP32 / ESP8266	Daten-Pull-up an 3,3 V legen und einen geeigneten GPIO waehlen.
Mehrere Sensoren	Nicht nur per Index, sondern ueber die eindeutige Adresse zuordnen.

Einbauhinweise

- Nur die Edelstahlsonde und den vorgesehenen Kabelbereich Feuchtigkeit aussetzen.
- Offene Kabelenden, Klemmstellen und Controller trocken unterbringen.
- Mechanische Zugbelastung am Uebergang von Kabel zu Sonde vermeiden.
- Bei extremen Temperaturen die Grenzen des gesamten Fuehlers beachten, nicht nur die des Chips.

[DS18B20-Fuehler bei MAKERSHOP.DE](#)

[Offizielle DS18B20-Dokumentation](#)

[DallasTemperature Arduino-Bibliothek](#)