

# Az-Delivery

## Willkommen!

Vielen Dank, dass Sie sich für unseren AZ-Delivery DS18B20 Temperatursensor entschieden haben. Auf den folgenden Seiten zeigen wir Ihnen, wie Sie dieses praktische Gerät bedienen und einrichten können.

## Viel Spaß!



## Anwendungsbereiche

Bildung und Lehre: Einsatz in Schulen, Hochschulen und Ausbildungseinrichtungen zur Vermittlung von Grundlagen der Elektronik, Programmierung und eingebetteten Systemen. Forschung und Entwicklung: Verwendung in Forschungs- und Entwicklungsprojekten zur Erstellung von Prototypen und Experimenten in den Bereichen Elektronik und Informatik. Prototypenentwicklung: Einsatz in der Entwicklung und Erprobung neuer elektronischer Schaltungen und Geräte. Hobby und Maker-Projekte: Verwendung durch Elektronikenthusiasten und Hobbyisten zur Entwicklung und Umsetzung von DIY-Projekten.

## Erforderliche Kenntnisse und Fähigkeiten

Grundlegendes Verständnis der Elektronik und Elektrotechnik. Kenntnisse in der Programmierung, insbesondere in der Programmiersprache C/C++. Fähigkeit, Schaltpläne zu lesen und einfache Schaltungen zu entwerfen. Erfahrung im Umgang mit elektronischen Komponenten und Löten.

## Betriebsbedingungen

Das Produkt darf nur mit den im Datenblatt spezifizierten Spannungen betrieben werden, um Beschädigungen zu vermeiden. Eine stabilisierte Gleichstromquelle ist zum Betrieb erforderlich. Bei der Verbindung mit anderen elektronischen Komponenten und Schaltungen sind die maximalen Strom- und Spannungsgrenzen zu beachten, um Überlastungen und Schäden zu vermeiden.

## Umweltbedingungen

Das Produkt sollte in einer sauberen, trockenen Umgebung verwendet werden, um Schäden durch Feuchtigkeit oder Staub zu vermeiden. Schützen Sie das Produkt vor direkter Sonneneinstrahlung (UV)

## Bestimmungsgemäße Verwendung

Das Produkt ist für den Einsatz in Bildungs-, Forschungs- und Entwicklungsumgebungen konzipiert wurde. Es dient zur Entwicklung, Programmierung und Prototypenentwicklung von elektronischen Projekten und Anwendungen. Das Sensor Produkt ist nicht als fertiges Verbraucherprodukt gedacht, sondern als Werkzeug für technisch versierte Nutzer, darunter Ingenieure, Entwickler, Forscher und Studenten.

## Nicht bestimmungsgemäße vorhersehbare Verwendung

Das Produkt eignet sich nicht für den industriellen Einsatz oder sicherheitsrelevante Anwendungen. Eine Verwendung des Produkts in Medizingeräten oder für Zwecke der Luft- und Raumfahrt ist nicht zulässig

## Entsorgung

Nicht mit dem Hausmüll entsorgen! Ihr Produkt ist entsprechend der europäischen Richtlinie über Elektro- und Elektronik-Altgeräte umweltgerecht zu entsorgen. Die darin enthaltenen, wertvollen Rohstoffe können so der Wiederverwendung zugeführt werden. Die Anwendung dieser Richtlinie trägt zum Umwelt- und Gesundheitsschutz bei. Nutzen Sie die von Ihrer Kommune eingerichtete Sammelstelle zur Rückgabe und Verwertung elektrischer und elektronischer Altgeräte. WEEE-Reg.-Nr.: DE 62624346

## Elektrostatische Entladung

Achtung: Elektrostatische Entladungen können das Produkt beschädigen. Hinweis: Erden Sie sich, bevor Sie das Produkt berühren, indem Sie beispielsweise ein antistatisches Armband tragen oder eine geerdete Metalloberfläche berühren.

## Sicherheitshinweise

Obwohl unser Produkt den Anforderungen der RoHS-Richtlinie (2011/65/EU) entspricht und keine gefährlichen Stoffe in über den Grenzwerten zulässigen Mengen enthält, können dennoch Rückstände vorhanden sein. Beachten Sie die folgenden Sicherheitshinweise, um chemische Gefährdungen zu vermeiden: Achtung: Beim Löten können Dämpfe entstehen, die gesundheitsschädlich sein können. Hinweis: Verwenden Sie einen Lötdampfabsauger oder arbeiten Sie in einem gut belüfteten Bereich. Tragen Sie gegebenenfalls eine Atemschutzmaske. Achtung: Einige Personen könnten empfindlich auf bestimmte Materialien oder Chemikalien reagieren, die im Produkt enthalten sind. Hinweis: Sollten Hautreizungen oder allergische Reaktionen auftreten, unterbrechen Sie die Nutzung und suchen Sie gegebenenfalls einen Arzt auf. Achtung: Halten Sie das Produkt außerhalb der Reichweite von Kindern und Haustieren, um versehentlichen Kontakt und Verschlucken von Kleinteilen zu vermeiden. Hinweis: Bewahren Sie das Produkt in einem sicheren, geschlossenen Behälter auf, wenn es nicht verwendet wird. Achtung: Vermeiden Sie den Kontakt des Produkts mit Nahrungsmitteln und Getränken. Hinweis: Lagern und verwenden Sie das Produkt nicht in der Nähe von Lebensmitteln, um eine Kontamination zu verhindern. Obwohl unser Produkt den

Anforderungen der RoHS-Richtlinie (2011/65/EU) entspricht und keine gefährlichen Stoffe in über den Grenzwerten zulässigen Mengen enthält, können dennoch Rückstände vorhanden sein. Beachten Sie die folgenden Sicherheitshinweise, um chemische Gefährdungen zu vermeiden: Achtung: Beim Löten können Dämpfe entstehen, die gesundheitsschädlich sein können. Hinweis: Verwenden Sie einen Lötdampfabsauger oder arbeiten Sie in einem gut belüfteten Bereich. Tragen Sie gegebenenfalls eine Atemschutzmaske. Achtung: Einige Personen könnten empfindlich auf bestimmte Materialien oder Chemikalien reagieren, die im Produkt enthalten sind. Hinweis: Sollten Hautreizungen oder allergische Reaktionen auftreten, unterbrechen Sie die Nutzung und suchen Sie gegebenenfalls einen Arzt auf. Achtung: Halten Sie das Produkt außerhalb der Reichweite von Kindern und Haustieren, um versehentlichen Kontakt und Verschlucken von Kleinteilen zu vermeiden. Hinweis: Bewahren Sie das Produkt in einem sicheren, geschlossenen Behälter auf, wenn es nicht verwendet wird. Achtung: Vermeiden Sie den Kontakt des Produkts mit Nahrungsmitteln und Getränken. Hinweis: Lagern und verwenden Sie das Produkt nicht in der Nähe von Lebensmitteln, um eine Kontamination zu verhindern. Das Produkt enthält empfindliche elektronische Komponenten und scharfe Kanten. Unsachgemäßer Umgang oder Montage kann zu Verletzungen oder Beschädigungen führen. Beachten Sie die folgenden Sicherheitshinweise, um mechanische Gefährdungen zu vermeiden: Achtung: Die Platine und die Anschlüsse des Produkts können scharfe Kanten aufweisen. Gehen Sie vorsichtig vor, um Schnittverletzungen zu vermeiden. Hinweis: Tragen Sie bei der Handhabung und Montage des Produkts geeignete Schutzhandschuhe. Achtung: Vermeiden Sie übermäßigen Druck oder mechanische Belastung der Platine und der Komponenten. Hinweis: Montieren Sie das Produkt nur auf stabilen und ebenen Oberflächen. Verwenden Sie geeignete Abstandshalter und Gehäuse, um mechanische Belastungen zu minimieren. Achtung: Stellen Sie sicher, dass das Produkt sicher befestigt ist, um unbeabsichtigtes Verrutschen oder Herunterfallen zu verhindern. Hinweis: Verwenden Sie passende Unterlage oder eine sichere Befestigung in Gehäusen oder auf Montageplatten. Achtung: Achten Sie darauf, dass alle Kabelverbindungen sicher und korrekt angeschlossen sind, um Zugbelastungen und versehentliches Herausziehen zu vermeiden. Hinweis: Führen Sie Kabel so, dass sie nicht unter Spannung stehen und keine Stolpergefahr darstellen. Das Produkt arbeitet mit elektrischen Spannungen und Strömen, die bei unsachgemäßem Gebrauch zu elektrischen Schlägen, Kurzschlüssen oder anderen Gefahren führen können. Beachten Sie die folgenden Sicherheitshinweise, um elektrische Gefährdungen zu vermeiden: Achtung: Verwenden Sie das Produkt nur mit den spezifizierten Spannungen. Hinweis: Die Leistungsgrenzen des Produkts finden Sie im dazugehörigen Datenblatt Achtung: Vermeiden Sie Kurzschlüsse zwischen den Anschlüssen und Komponenten des Produkts Hinweis: Achten Sie darauf, dass keine leitenden Objekte die Platine berühren oder überbrücken. Verwenden Sie isolierte Werkzeuge und beachten Sie die Anordnung der Verbindungen. Achtung: Führen Sie keine Arbeiten am Produkt durch, wenn es mit einer Stromquelle verbunden ist. Hinweis: Trennen Sie das Produkt von der Stromversorgung, bevor Sie Änderungen an der Schaltung vornehmen oder Komponenten anschließen bzw. entfernen. Achtung: Überschreiten Sie nicht die spezifizierten Stromstärken für die Ein- und Ausgänge des Produkts. Hinweis: Die Leistungsgrenzen des Produkts finden sich in den technischen Spezifikationen oder im Datenblatt Achtung: Stellen Sie sicher, dass die verwendeten Stromquellen stabil und korrekt dimensioniert sind. Hinweis: Verwenden Sie nur geprüfte und geeignete Netzteile, um Spannungsschwankungen und Überlastungen zu vermeiden. Achtung: Halten Sie ausreichenden Abstand zu spannungsführenden Teilen ein, um unabsichtlichen Kontakt zu vermeiden. Hinweis: Sorgen Sie entsprechend der verwendeten Spannung für eine sichere und übersichtliche Anordnung der Verkabelung. Achtung: Verwenden Sie isolierende Gehäuse oder Schutzabdeckungen, um das Produkt vor direktem Kontakt zu schützen. Hinweis: Setzen Sie das Produkt in ein nicht leitendes Gehäuse ein, um versehentliche Berührungen und Kurzschlüsse zu vermeiden. Das Produkt und die darauf befindlichen Komponenten können sich während des Betriebs erwärmen. Unsachgemäßer Umgang oder eine Überlastung des Produkts kann zu Verbrennungen, Beschädigungen oder Bränden führen. Beachten Sie die folgenden Sicherheitshinweise, um thermische Gefährdungen zu vermeiden: Achtung: Stellen Sie sicher, dass das Produkt innerhalb der empfohlenen Betriebstemperaturen verwendet wird. Hinweis: Der empfohlene Betriebstemperaturbereich liegt typischerweise zwischen -40°C und +85°C. Überprüfen Sie die spezifischen Angaben im Datenblatt des Produkts. Achtung: Platzieren Sie das Produkt nicht in der Nähe von externen Wärmequellen wie Heizkörpern oder direkter Sonneneinstrahlung. Hinweis: Sorgen Sie dafür, dass das Produkt in einem kühlen und gut belüfteten Bereich betrieben wird. Achtung: Stellen Sie sicher, dass das Produkt gut belüftet ist, um eine Überhitzung zu vermeiden. Hinweis: Verwenden Sie Lüfter oder Kühlkörper, wenn das Produkt in einem geschlossenen Gehäuse betrieben wird oder in einer Umgebung mit eingeschränkter Luftzirkulation. Achtung: Montieren Sie das Produkt auf hitzebeständigen Oberflächen und in hitzebeständigen Gehäusen. Hinweis: Verwenden Sie Materialien für Gehäuse, die hohe Temperaturen aushalten können, um eine Beschädigung oder Feuergefahr zu vermeiden. Achtung: Implementieren Sie eine Überwachung der Temperatur bei Verwendung eines Gehäuses und gegebenenfalls Schutzmechanismen, die das Produkt abschalten, wenn es überhitzt. Hinweis: Verwenden Sie Temperaturfühler und entsprechende Software, um die Temperatur des Produkts zu überwachen und das System bei Bedarf abzuschalten. Achtung: Vermeiden Sie Überlastungen, die zu übermäßiger Erwärmung der Komponenten führen können. Hinweis: Überschreiten Sie nicht die spezifizierten Grenzwerte für Strom und Spannung, um eine Überhitzung zu verhindern. Achtung: Kurzschlüsse können erhebliche Hitze entwickeln und Brände verursachen. Hinweis: Stellen Sie sicher, dass alle Verbindungen korrekt und sicher sind und dass keine leitenden Objekte unbeabsichtigt Kurzschlüsse verursachen können.

## Inhaltsverzeichnis

Einführung .....	3
Die Pinbelegung.....	4
Einrichten der Arduino IDE.....	5
Wie man den Raspberry Pi und Python einrichtet .....	9
Verbinden des Sensors mit dem Atmega328P Board .....	10
Bibliothek für Arduino IDE .....	11
Skizzenbeispiel .....	13
Verbinden des Sensors mit dem Raspberry Pi.....	19
Freischaltung der Schnittstelle .....	20
Aktivieren mehrerer Schnittstellen.....	23
Python-Skript für die Verwendung mehrerer DS18B20-Sensoren .....	24

## Einführung

Der DS18B20 ist ein digitaler Temperatursensor, der 9 bis 12 Bits digitale Temperaturmessungen liefert und eine Alarmfunktion mit nichtflüchtigen, vom Benutzer programmierbaren oberen und unteren Auslösepunkten besitzt. Der Sensor kommuniziert über einen Bus, der nur einen Datenpin, einen Stromversorgungspin und einen Erdungspin für die Kommunikation mit einem Mikrocontroller benötigt.

Jeder Sensor hat eine 64-Bit-Serienadresse, so dass mehrere Sensoren auf demselben Bus arbeiten können. Somit ist es einfach, einen Mikrocontroller zur Steuerung vieler Sensoren zu verwenden, die über ein großes Gebiet verteilt sind.

Der Sensor benötigt keinen Standby-Strom, d.h. wenn keine Temperaturdaten gelesen werden, verbraucht der Sensor keinen Strom. Der Temperaturmessbereich reicht von  $-55^{\circ}\text{C}$  bis  $+125^{\circ}\text{C}$  ( $67^{\circ}\text{F}$  bis  $257^{\circ}\text{F}$ ), mit einer Genauigkeit von  $\pm 0,5^{\circ}\text{C}$  (9 Bit);  $\pm 0,25^{\circ}\text{C}$  (10 Bit);  $\pm 0,125^{\circ}\text{C}$  (11 Bit); und  $\pm 0,0625^{\circ}\text{C}$  (12 Bit) Auflösung.

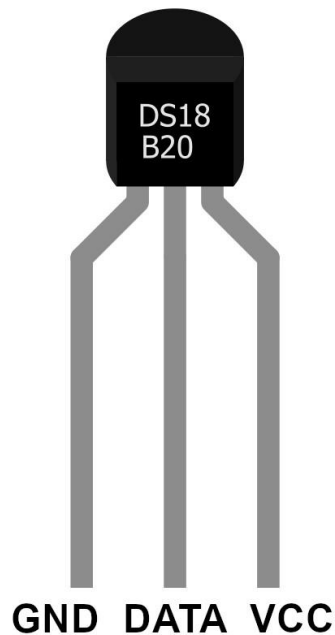
**HINWEIS:** Sollten Kommunikationsprobleme auftreten, versuchen Sie, einen  $4,7\text{k}\Omega$ -Pull-up-Widerstand an den DATA-Pin anzuschließen.

**HINWEIS:** In den technischen Daten der Schnittstelle wird die maximale Anzahl der Sensoren, die mit derselben Schnittstelle verbunden werden können, nicht angegeben. Schnittstelle verbunden werden können, aber in der praktischen Anwendung ist diese Zahl nicht so hoch, und Sie sollten darauf achten.

**HINWEIS:** Es gibt eine Begrenzung der Kabellänge, die bei der Verwendung von Langstrecken-Kommunikation berücksichtigt werden sollte. Achten Sie auf die verteilte Kapazität und den Widerstand des Kabels.

**HINWEIS:** Der DS18B20 und gewöhnliche Transistoren sehen ähnlich aus, daher sollte man ihn nicht für einen Transistor halten, um Schäden zu vermeiden!

## Die Pinbelegung



**VCC pin** - versorgt den Sensor mit Strom. Obwohl die Versorgungsspannung zwischen 3,3V und 5,5V liegen kann, wird eine 5V-Versorgung empfohlen. Im Falle einer 5V-Versorgung. Ein Kabel, das Sensor und Mikrocontroller verbindet, kann bis zu 20 Meter lang sein. Bei einer Versorgungsspannung von 3,3 V sollte die Kabellänge jedoch nicht länger als ein Meter sein, da sonst der Spannungsabfall in der Leitung zu Messfehlern führt.

**DATA pin** - ist der Daten-Pin und wird für die Kommunikation zwischen dem Sensor und dem Mikrocontroller verwendet (kann an die Ein-Draht-Schnittstelle angeschlossen werden)

**GND pin** - ist der Masse-Pin und sollte mit der gemeinsamen Masse oder 0V (auf Atmega328P Board oder Raspberry Pi) verbunden werden.

## Einrichten der Arduino IDE

Wenn Sie Arduino IDE noch nicht installiert haben, folgen Sie dem Link: <https://www.arduino.cc/en/Main/Software> und laden Sie die Installationsdatei für Ihr Betriebssystem herunter.

### Download the Arduino IDE



Für *Windows*-Benutzer: Doppelklicken Sie auf die heruntergeladene .exe-Datei und folgen Sie den Anweisungen im Installationsfenster..

Für *Linux*-Benutzer laden Sie eine Datei mit der Erweiterung *.tar.xz* herunter, die Sie entpacken müssen. Gehen Sie nach dem Entpacken in das entpackte Verzeichnis und öffnen Sie das Terminal in diesem Verzeichnis. Sie müssen zwei *.sh*-Skripte ausführen, das erste heißt *arduino-linux-setup.sh* und das zweite heißt *install.sh*.



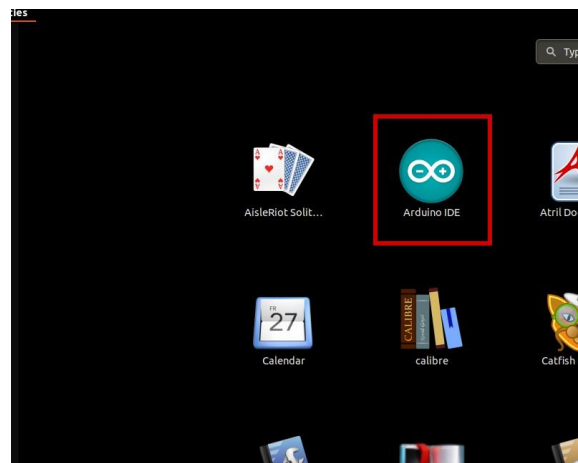
# Az-Delivery

Um das erste Skript im Terminal auszuführen, führen Sie den folgenden Befehl aus:

`sh arduino-linux-setup.sh user_name user_name` - ist der Name eines Superusers im Linux-Betriebssystem. Sie werden aufgefordert, das Passwort für den Superuser einzugeben. Warten Sie ein paar Minuten, bis das Skript alles abgeschlossen hat.

Nach der Installation des ersten Skripts müssen Sie das zweite Skript namens `install.sh` ausführen. Führen Sie im Terminal den folgenden Befehl aus: `sh install.sh`

Nach der Installation dieser Skripte gehen Sie zu Alle Apps, wo Sie die Arduino-IDE installiert finden können..

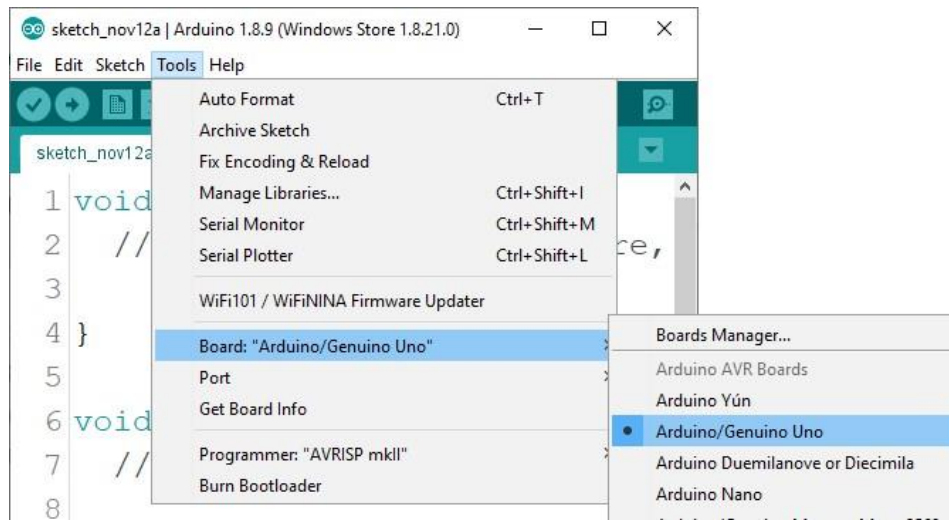


Auf fast allen Betriebssystemen ist ein Texteditor vorinstalliert (z. B. auf Windows das Notepad, auf Linux Ubuntu das Gedit, auf Linux Raspbian das Leafpad usw.). Alle diese Texteditoren sind für den Zweck des Ebooks vollkommen ausreichend. Als Nächstes müssen Sie überprüfen, ob Ihr PC Ihr Mikrocontroller-Board erkennen kann.



Öffnen Sie die frisch installierte Arduino IDE und gehen Sie zu:  
*Werkzeuge > Board > {Ihr Boardname hier}*

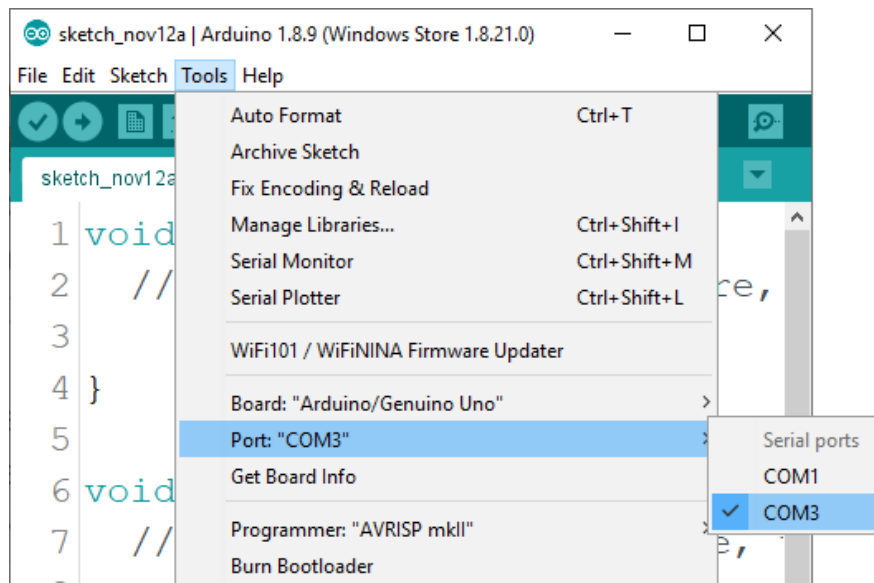
*{Ihr Boardname hier}* sollte der *Arduino/Genuino Uno* sein, wie Sie auf dem folgenden Bild sehen können:



Sie müssen den Port auswählen, an den das Mikrocontroller-Board angeschlossen ist. Gehen Sie zu: *Tools > Port > {Port name goes here}* und wenn Sie das Mikrocontroller-Board an den USB-Port angeschlossen haben, sollten Sie einen Portnamen sehen.

Wenn Sie die Arduino IDE unter Windows verwenden, lauten die Portnamen wie folgt:

# Az-Delivery



Für Linux-Benutzer lautet der Portname zum Beispiel `/dev/ttyUSBx`, wobei x eine ganze Zahl zwischen 0 und 9 ist, zum Beispiel.



## **Wie man den Raspberry Pi und Python einrichtet**

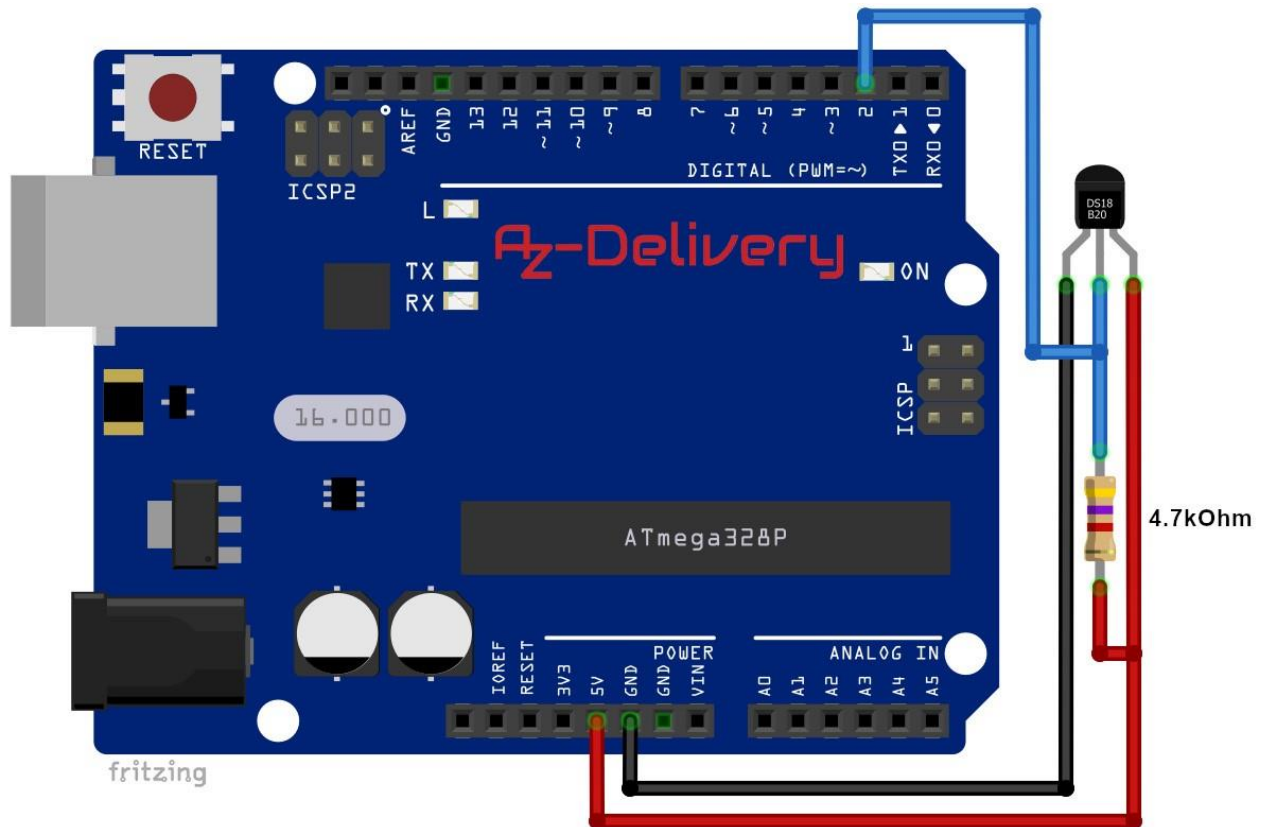
Auf dem Raspberry Pi müssen Sie zunächst das Betriebssystem installieren und dann alles so einrichten, dass Sie ihn im Headless-Modus verwenden können. Der Headless-Modus ermöglicht es Ihnen, sich aus der Ferne mit dem Raspberry Pi zu verbinden, ohne dass Sie einen PC-Monitor, eine Maus oder eine Tastatur benötigen. Die einzigen Dinge, die Sie für diesen Modus benötigen, sind der Raspberry Pi, eine Stromversorgung und eine Internetverbindung. All dies wird im Detail im kostenlosen eBook Raspberry Pi Quick Startup Guide erklärt, das Sie auf unserer Website finden:

<https://www.az-delivery.de/products/raspberry-pi-kostenfreies-e-book?ls=en>

Auf dem Raspbian-Betriebssystem ist Python bereits vorinstalliert.

## Verbinden des Sensors mit dem Atmega328P Board

Verbinden Sie den DS18B20-Sensor mit dem Mikrocontroller-Board wie im folgenden Anschlussplan dargestellt:



**DS18B20 pin** > **Mikrocontroller-Pin**

DATA > D2

VCC > 5V

GND > GND

**Blauer Draht**

**Rote Leitung**

**Schwarzer Draht**

**HINWEIS:** Zwischen dem OUT-Pin und dem VCC-Pin ist ein 4,7kΩ Pull-up-Widerstand angeschlossen.

## Bibliothek für Arduino IDE

Um den DS18B20 Sensor mit dem Atmega328P Board zu verwenden, müssen wir zuerst eine Bibliothek dafür herunterladen. Gehe zu:  
Werkzeuge > Bibliotheken verwalten

Wenn sich ein neues Fenster öffnet, geben Sie Dallas in das Suchfeld ein und laden Sie die Bibliothek DallasTemperature von Miles Burton, Tim Newsome, Guil Barros und Rob Tillaart herunter, wie auf dem folgenden Bild gezeigt:

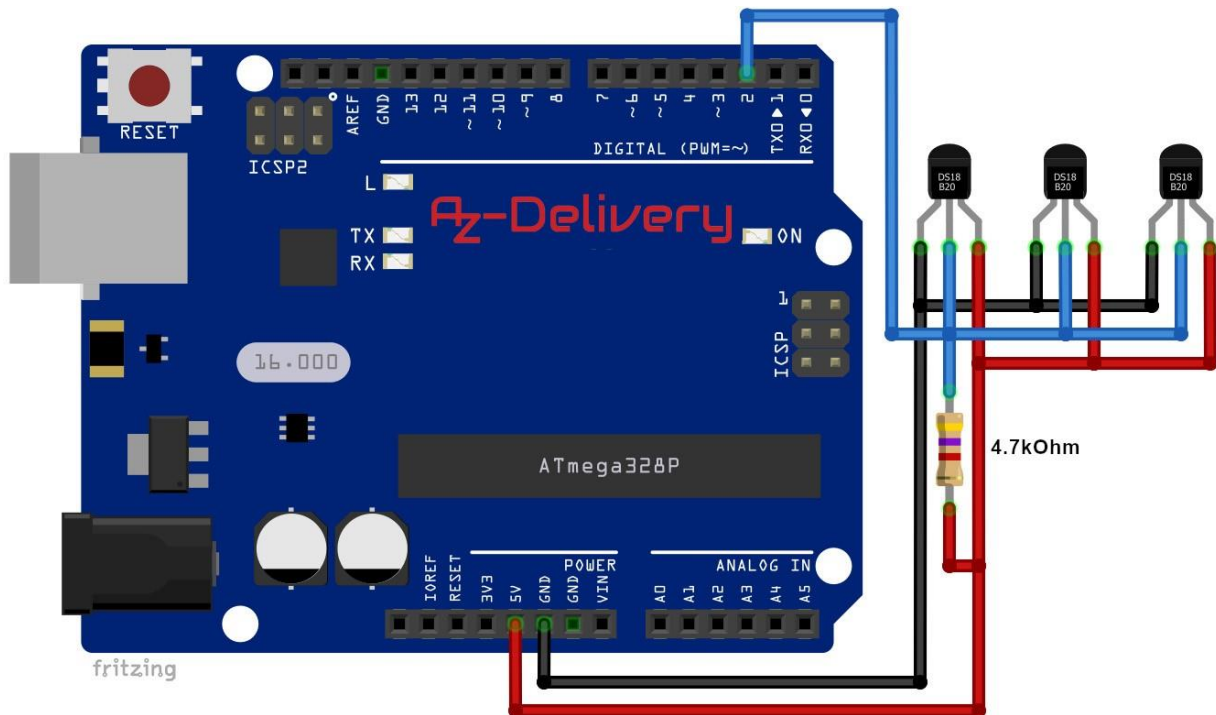


Gehen Sie nun zu: Datei > Beispiele > DallasTemperature > ... und Sie werden viele Skizzenbeispiele finden. Wir haben einen Sketch namens Multiple verwendet und modifiziert, um Temperaturdaten von drei verschiedenen DS18B20-Sensoren zu lesen. Datei > Beispiele > DallasTemperature > Multiple

Wenn Sie nur einen DS18B20-Sensor verwenden, ist ein Sketch namens Simple ausreichend:

Datei > Beispiele > DallasTemperature > Simple

Verbinden Sie drei DS18B20-Sensoren mit dem Atmega328P-Board, wie im folgenden Anschlussplan dargestellt:



## Skizzenbeispiel

Das folgende Skizzenbeispiel bezieht sich auf drei DS18B20-Sensoren an derselben One-Wire-Schnittstelle:

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2 // Data wire is plugged into D2 pin
#define TEMPERATURE_PRECISION 12
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
DeviceAddress one, two, three;

void setup() {    Serial.begin(9600);    Serial.println("Dallas
Temperature IC Control Library Demo");    sensors.begin();
Serial.print("Locating devices...");
    Serial.print("Found ");
    Serial.print(sensors.getDeviceCount(), DEC);
    Serial.println(" devices.");
Serial.print("Parasite power is: ");
if(sensors.isParasitePowerMode()) {
Serial.println("ON");    }    else {
Serial.println("OFF");
    }
    // one tab    if(!sensors.getAddress(one, 0)) {
Serial.println("Unable to find address for Device 0"); }
if(!sensors.getAddress(two, 2)) {    Serial.println("Unable to
find address for Device 2"); }    if(!sensors.getAddress(three,
1)) {    Serial.println("Unable to find address for Device 1");
    }
}
```



# Az-Delivery

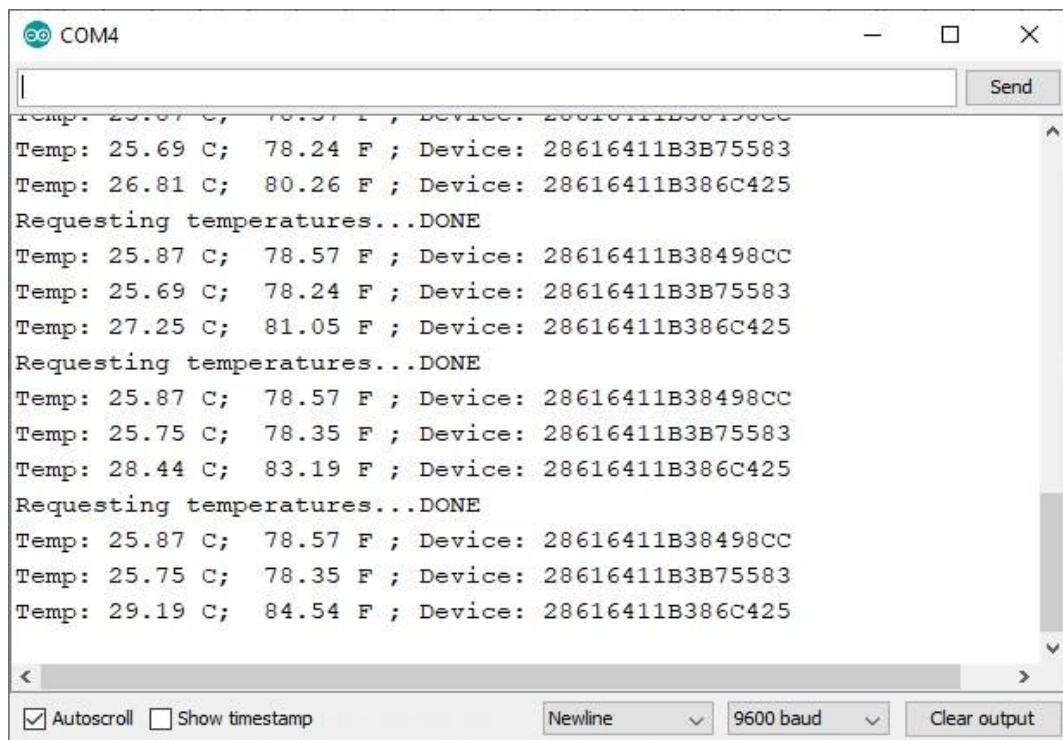
```
    Serial.print("Device 0 Address: ");    printAddress(one);
Serial.println();    Serial.print("Device 1 Address: ");
printAddress(two);    Serial.println();
Serial.print("Device 2 Address: ");    printAddress(three);
Serial.println();    sensors.setResolution(one,
TEMPERATURE_PRECISION);    sensors.setResolution(two,
TEMPERATURE_PRECISION);    sensors.setResolution(three,
TEMPERATURE_PRECISION);    Serial.print("Device 0
Resolution: ");

    Serial.print(sensors.getResolution(one), DEC);
    Serial.println();
    Serial.print("Device 1 Resolution: ");
    Serial.print(sensors.getResolution(two), DEC);
    Serial.println();
    Serial.print("Device 2 Resolution: ");
    Serial.print(sensors.getResolution(three), DEC);
    Serial.println();
}
void printAddress(DeviceAddress deviceAddress) {    for(uint8_t
i = 0; i < 8; i++) {        if(deviceAddress[i] < 16)
Serial.print("0");        Serial.print(deviceAddress[i], HEX);
    }    }    void printTemperature(DeviceAddress deviceAddress) {
float        tempC        =        sensors.getTempC(deviceAddress);
Serial.print("Temp: ");
    Serial.print(tempC);
    Serial.print(" C; ");
    Serial.print(DallasTemperature::toFahrenheit(tempC));
    Serial.print(" F");    }
void printResolution(DeviceAddress deviceAddress) {
Serial.print("Resolution: ");
```

# Az-Delivery

```
Serial.println(sensors.getResolution(deviceAddress)); } void  
printData(DeviceAddress deviceAddress) {  
  printTemperature(deviceAddress);  Serial.print(" ; Device: ");  
  printAddress(deviceAddress);  Serial.println();  
}  
void loop() {  Serial.print("Requesting  
temperatures...");  
  sensors.requestTemperatures();  
  Serial.println("DONE");  printData(one);  
  printData(two);  printData(three);  
  delay(1000); }
```

Laden Sie den Sketch auf das Mikrocontroller-Board hoch und öffnen Sie Serial Monitor (Tools > Serial Monitor). Das Ergebnis sollte so aussehen, wie in der Abbildung unten gezeigt:



# Az-Delivery

Wir verwenden die Variable `ONE_WIRE_BUS`, um zu definieren, an welchen digitalen Pin wir die One-Wire-Schnittstelle anschließen. Für dieses eBook wird der Wert der `ONE_WIRE_BUS`-Variable auf D2 gesetzt, aber Sie können jeden anderen digitalen Pin des Mikrocontroller-Boards verwenden, mit Ausnahme derjenigen, die in der seriellen Schnittstelle verwendet werden, D0 und D1 (es ist eine Empfehlung, Sie können sie verwenden, aber Sie müssen sicher sein, dass diese Pins getrennt sind, wenn Sie Skizzen hochladen).

Wir haben die Variable `TEMPERATURE_PRECISION` verwendet, um die Genauigkeit für DS18B20-Sensoren einzustellen. Die in dieser Variablen gespeicherte Zahl ist eine digitale Umwandlungszahl in Bits und kann im Bereich von 9 bis 12 liegen, jede andere Zahl führt zu einem Fehler. Für dieses eBook haben wir die Variable auf den Maximalwert (12) gesetzt.

Wir haben die folgende Codezeile verwendet: `DeviceAddress` eins, zwei, drei um Variablen für Sensoradressen zu erstellen, und in unserem Beispiel haben wir drei erstellt.

Wir haben ein `OneWire`-Objekt definiert und erstellt, das für die Busschnittstelle verwendet wird: `OneWire oneWire(ONE_WIRE_BUS);`

Dann haben wir das Objekt `oneWire` verwendet, um das Objekt `sensors` zu definieren und zu erstellen, das für alle angeschlossenen Sensoren verwendet wird:  
`DallasTemperature sensors(&oneWire))`

Um das Objekt `sensors` zu initialisieren, haben wir die folgende Codezeile verwendet: `sensors.begin()`

Mit dieser Codezeile erkennt das `sensors`-Objekt alle an der Busschnittstelle angeschlossenen Sensoren. Es erkennt auch alle Adressen der Sensoren.

# Az-Delivery

Jetzt können wir überprüfen, ob die Sensoren richtig funktionieren, indem wir die folgenden Codezeilen für jeden Sensor verwenden, den wir an die One-Wire-Schnittstelle anschließen: `if(!sensors.getAddress(one, 0)) { Serial.println("Unable to find address: Device 0"); }` wobei `one` die Adresse des ersten Sensors ist.

Um die Genauigkeit der Analog-Digital-Wandlung für einen bestimmten Sensor einzustellen, verwenden wir die folgende Codezeile:  
`sensors.setResolution(one, TEMPERATURE_PRECISION)`

Wenn Sie die Analog-Digital-Wandlungsgenauigkeit des jeweiligen Sensors auslesen möchten, können Sie die folgende Codezeile verwenden:  
`sensors.getResolution(one)`

Die Funktion gibt einen hexadezimalen Wert zurück, und um ihn in einen dezimalen Wert umzuwandeln, verwenden wir die folgende Codezeile:  
`Serial.print(sensors.getResolution(one), DEC);`

Um die Temperaturdaten zu lesen, müssen wir zunächst alle Daten von allen Sensoren abfragen, indem wir die folgende Codezeile verwenden:  
`sensors.requestTemperatures();`

Erst nach dieser Codezeile können wir die Daten eines bestimmten Sensors mit der folgenden Codezeile auslesen:

`float tempC = sensors.getTempC(deviceAddress);`

wobei wir das Argument `deviceAddress` an die Funktion übergeben, um Temperaturdaten von einem bestimmten Sensor zu lesen. Diese Daten sind

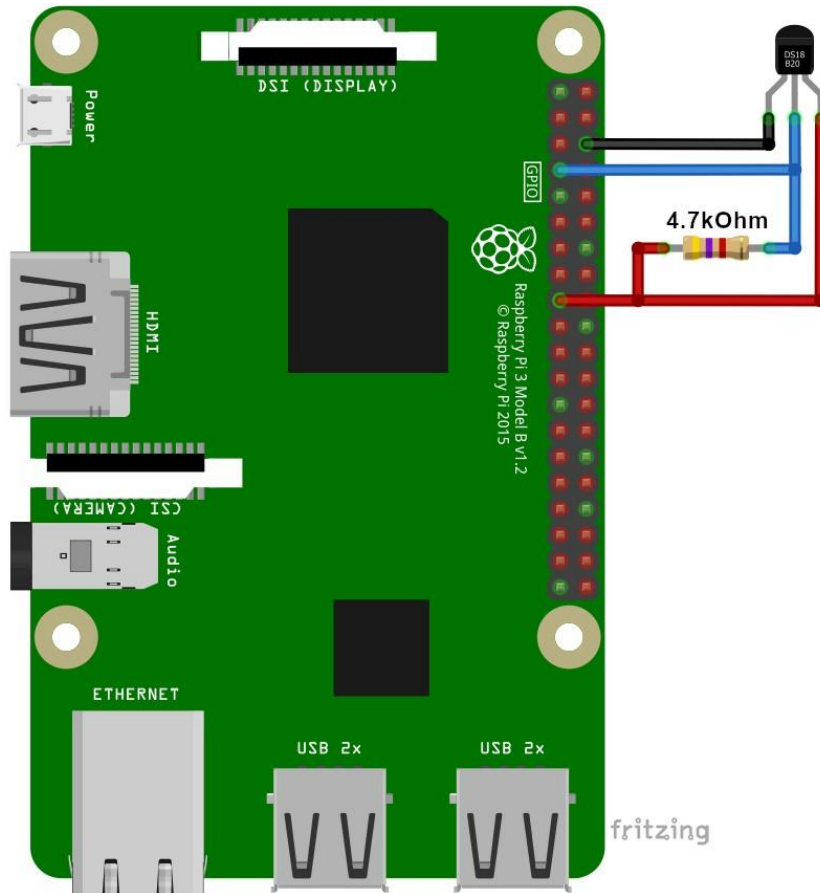


Temperaturwerte in Celsius, und um sie in Fahrenheit umzuwandeln, verwenden wir die folgende Codezeile:

```
DallasTemperature::toFahrenheit(tempC)
```

## Verbinden des Sensors mit dem Raspberry Pi

Verbinden Sie den DS18B20-Sensor mit dem Raspberry Pi wie im folgenden Anschlussplan dargestellt:



DS18B20 pin	>	Raspberry Pi pin
GND	>	GND [pin 6]
DATA	>	GPIO4 [pin 7]
VCC	>	3V3 [pin 17]

**Schwarzes Kabel**

**Blaues Kabel**

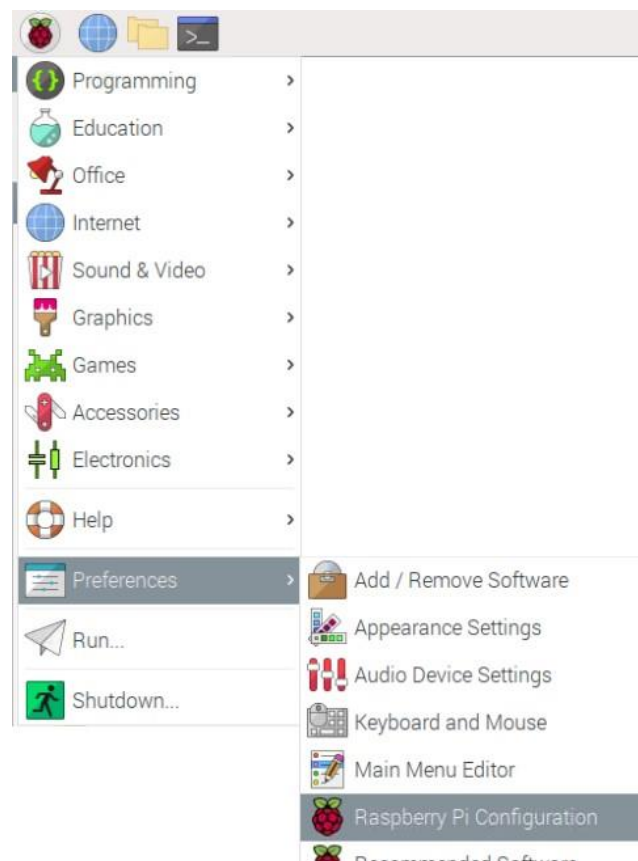
**Rotes Kabel**

**HINWEIS:** Zwischen dem OUT-Pin und dem 3V3-Pin ist ein Pull-up-Widerstand von 4,7kΩ angeschlossen.

## Freischaltung der Schnittstelle

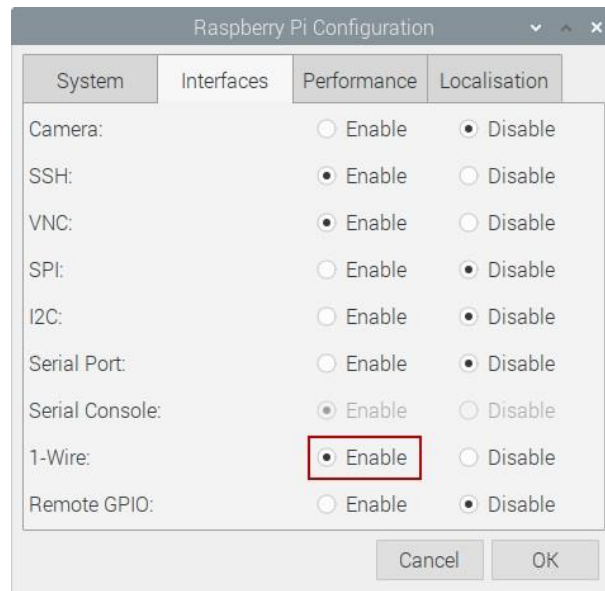
Bevor wir den DS18B20-Sensor mit dem Raspberry Pi verwenden können, müssen wir zunächst die Busschnittstelle in Raspbian aktivieren. Standardmäßig ist die Hardware-Bus-Schnittstelle auf Pin GPIO4 (Pin 7), aber wir müssen sie zuerst aktivieren. Um die Busschnittstelle zu aktivieren, gehen Sie zu:

*Applications Menu > Preferences > Raspberry Pi Configuration,*  
wie in der folgenden Abbildung gezeigt:

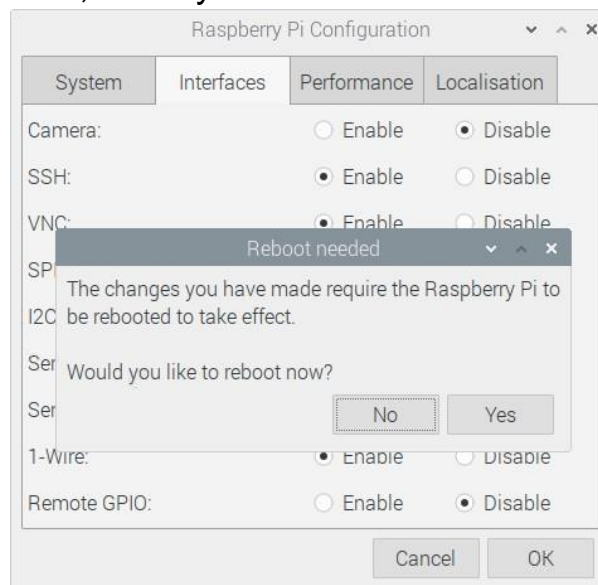




Wenn das neue Fenster angezeigt wird, öffnen Sie die Registerkarte Schnittstellen und suchen Sie nach den Optionsschaltflächen mit der Bezeichnung 1-Wire, und wählen Sie die Optionsschaltfläche Enable (Aktivieren), wie in der folgenden Abbildung dargestellt:



Sie werden aufgefordert, das System neu zu starten.



# Az-Delivery

Wenn Raspbian wieder hochgefahren ist, öffnen Sie das Terminal und führen Sie die folgenden Befehle nacheinander aus: `sudo modprobe w1-gpio` `sudo modprobe w1-therm` `cd /sys/bus/w1/devices/`

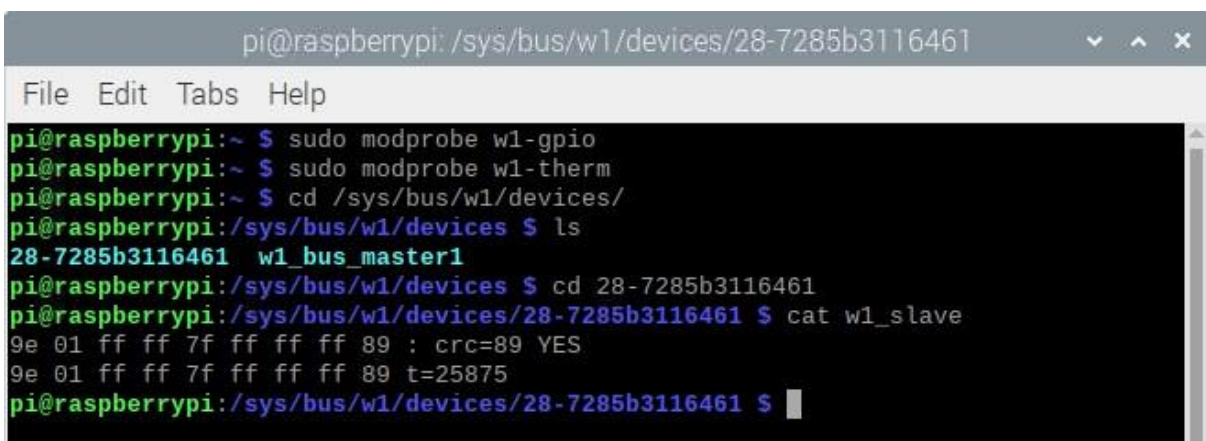
und wenn Sie den folgenden Befehl: `ls` ausführen, sollte die Ausgabe im Terminal wie folgt aussehen:

28-7285b3116461 und w1\_bus\_master1

Die erste Zahl 28-7285b3116461 wird für Sie unterschiedlich sein, da dies die serielle Adresse des spezifischen Sensors ist, und jeder Sensor hat seine eigene eindeutige serielle Adresse. Um zu testen, ob alles funktioniert, führen Sie diese beiden Befehle aus:

`cd 28-7285b3116461` - eine Nummer oder Serienadresse von der letzten Seite `cat w1_slave`

Das Ergebnis sollte so aussehen, wie auf dem folgenden Bild zu sehen ist:



```
pi@raspberrypi:~ $ sudo modprobe w1-gpio
pi@raspberrypi:~ $ sudo modprobe w1-therm
pi@raspberrypi:~ $ cd /sys/bus/w1/devices/
pi@raspberrypi:/sys/bus/w1/devices $ ls
28-7285b3116461 w1_bus_master1
pi@raspberrypi:/sys/bus/w1/devices $ cd 28-7285b3116461
pi@raspberrypi:/sys/bus/w1/devices/28-7285b3116461 $ cat w1_slave
9e 01 ff ff 7f ff ff ff 89 : crc=89 YES
9e 01 ff ff 7f ff ff ff 89 t=25875
pi@raspberrypi:/sys/bus/w1/devices/28-7285b3116461 $
```

t=25875 - dies sind die Temperaturdaten in °C (Celsius) = 25.875°C.



## Aktivieren mehrerer Schnittstellen

Um die Busschnittstelle ohne grafische Benutzeroberfläche (GUI) zu aktivieren, bevor Sie Ihren Raspberry Pi neu starten, müssen Sie in der Datei auf:

/boot/config.txt die folgende Zeile einfügen: `dtoverlay=w1-gpio` oder

`dtoverlay=w1-gpio,gpiopin=x`

wobei x ein benutzerdefinierter Pin ist, wenn Sie ihn verwenden möchten (Standard ist GPIO4 [Pin 7], wie im vorherigen Kapitel erwähnt).

Neuere Kernel (4.9.28 und später) erlauben die Verwendung von dynamischem Overlay-Laden, einschließlich der Erstellung mehrerer One-Wire-Schnittstellen, die gleichzeitig verwendet werden können: `sudo dtoverlay w1-gpio gpiopin=4 pullup=0 # pin 7` `sudo dtoverlay w1-gpio gpiopin=17 pullup=0 # pin 11` `sudo dtoverlay w1-gpio gpiopin=27 pullup=0 # pin 13`

Sobald einer der oben genannten Schritte durchgeführt wurde und die Erkennung abgeschlossen ist, können Sie die Geräte auflisten, die der Raspberry Pi über alle One-Wire-Schnittstellen erkannt hat, indem Sie den folgenden Befehl im Terminal ausführen:

`ls /sys/bus/w1/devices/`

**HINWEIS:** Die Verwendung von w1-gpio auf dem Raspberry Pi erfordert normalerweise einen 4,7kΩ Pull-up-Widerstand zwischen dem GPIO-Pin und einer 3,3V-Versorgung.

## Python-Skript für die Verwendung mehrerer DS18B20-Sensoren

Wir haben uns entschieden, den Code in zwei Skripte aufzuteilen, da er so besser lesbar ist. Im Folgenden finden Sie den Code für die Klasse subscript:

```
import os
import glob
import time
class DS18B20:

    def __init__(self):
        os.system('modprobe w1-gpio')
        os.system('modprobe w1-therm') base_dir =
        '/sys/bus/w1/devices/' device_folder =
        glob.glob(base_dir + '28*')
        self._count_devices = len(device_folder)
        self._devices = list() i = 0 while i <
        self._count_devices:
            self._devices.append(device_folder[i] + '/w1_slave')
            i += 1

    def device_names(self):
        names = list() for i in
        range(self._count_devices):
            names.append(self._devices[i]) temp
            = names[i][20:35] names[i] = temp

        return names
```

# Az-Delivery

```
# (one tab) def _read_temp(self,
    index):
    f = open(self._devices[index], 'r') lines
    = f.readlines()
    f.close() return
    lines

def tempC(self, index = 0):
    lines = self._read_temp(index) retries = 5 while
    (lines[0].strip()[-3:] != 'YES') and (retries > 0):
        time.sleep(0.1) lines =
    self._read_temp(index) retries -= 1
    if retries == 0:
        return 998

    equals_pos = lines[1].find('t=') if
    equals_pos != -1:
        temp = lines[1][equals_pos + 2:]
    return float(temp) / 1000 else:
    return 999 # error

def device_count(self): return
    self._count_devices
```

(The most of code in the script is modified from the script on the adafruit site)

Speichern Sie das Skript unter dem Namen DS18B20classfile.py.

# Az-Delivery

Im Folgenden finden Sie den Code für das Hauptskript:

```
import time from DS18B20classFile
import DS18B20

degree_sign = u'\xb0' # degree sign
devices = DS18B20() count =
devices.device_count() names =
devices.device_names()

print('[Press CTRL+C to end the script]') try: # Main
program loop while True: i = 0 print('\nReading
temperature, number of sensors: {}'.format(count))

while i < count:
    container = devices.tempC(i) print('{} Temp:
{:0.3f}C, {:0.3f}F of the device {}'.format(i+1, container, degree_sign,
container * 9.0 / 5.0 + 32.0, degree_sign,
names[i]))

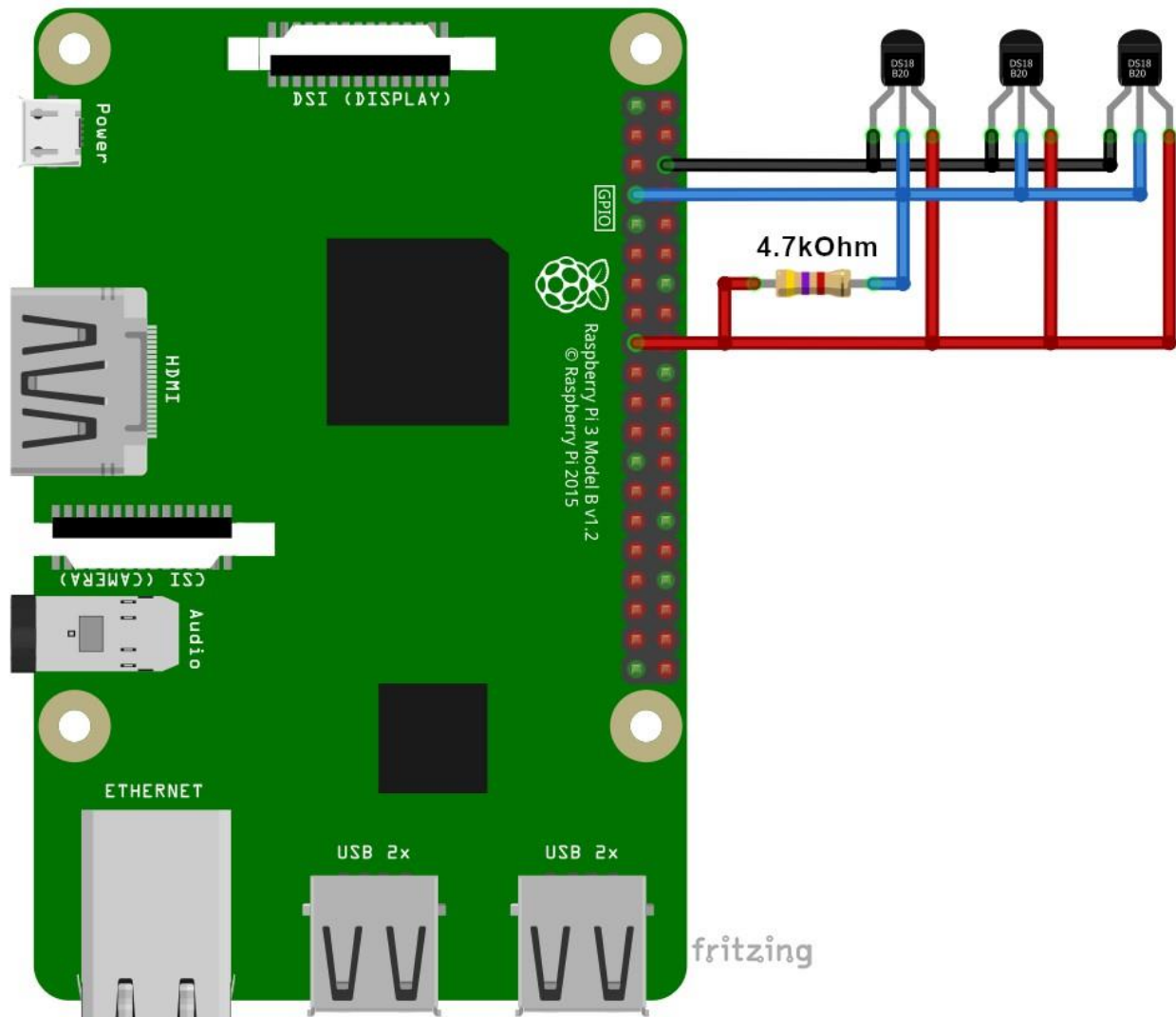
    i = i + 1

time.sleep(1)

# Scavenging work after the end of the program
except KeyboardInterrupt: print('\nScript
end!')
```

Speichern Sie das Skript unter dem Namen DS18B20multiple.py in demselben Verzeichnis, in dem Sie das erste Skript gespeichert haben.

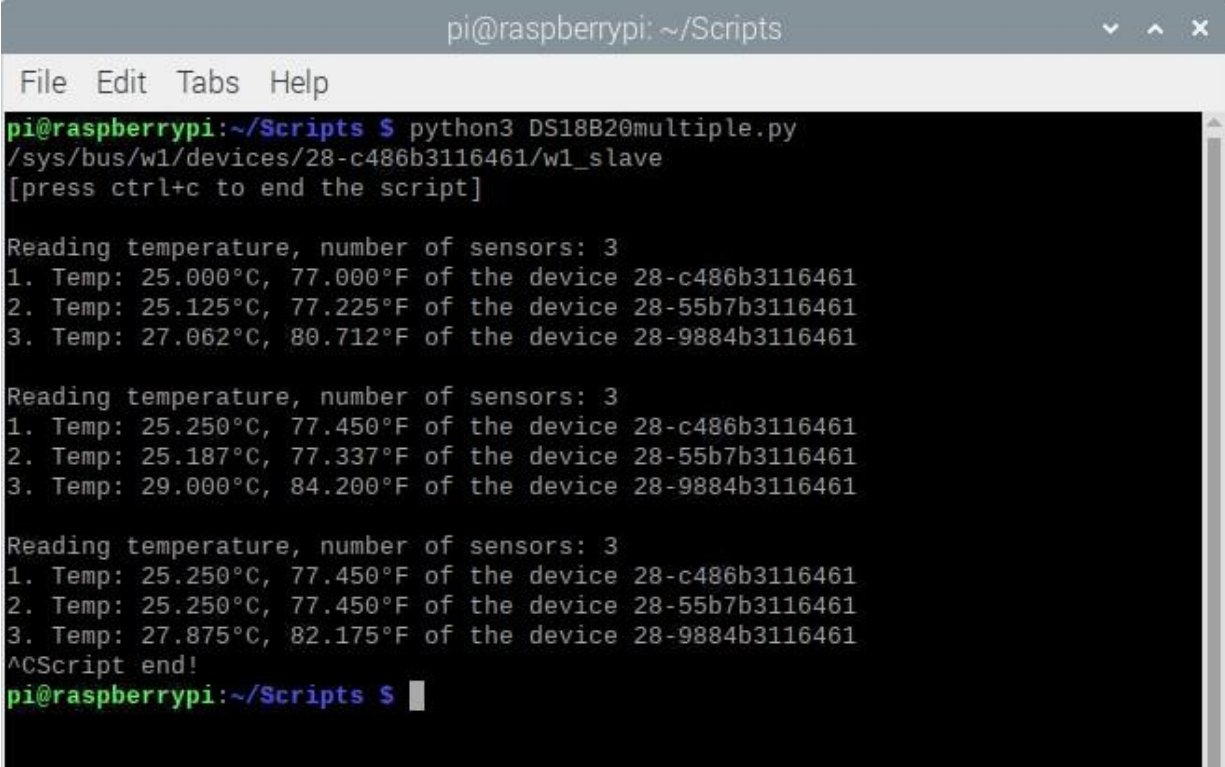
Wir haben zum Beispiel drei DS18B20-Sensoren an dieselbe One-Wire-Schnittstelle des Raspberry Pi angeschlossen, wie im folgenden Anschlussplan dargestellt:



Um das Hauptskript auszuführen, öffnen Sie das Terminal in dem Verzeichnis, in dem Sie beide Skripte gespeichert haben, und führen Sie den folgenden Befehl aus: `python3 DS18B20multiple.py`



Das Ergebnis sollte so aussehen wie in der folgenden Abbildung gezeigt:



```
pi@raspberrypi: ~/Scripts
File Edit Tabs Help
pi@raspberrypi:~/Scripts $ python3 DS18B20multiple.py
/sys/bus/w1/devices/28-c486b3116461/w1_slave
[press ctrl+c to end the script]

Reading temperature, number of sensors: 3
1. Temp: 25.000°C, 77.000°F of the device 28-c486b3116461
2. Temp: 25.125°C, 77.225°F of the device 28-55b7b3116461
3. Temp: 27.062°C, 80.712°F of the device 28-9884b3116461

Reading temperature, number of sensors: 3
1. Temp: 25.250°C, 77.450°F of the device 28-c486b3116461
2. Temp: 25.187°C, 77.337°F of the device 28-55b7b3116461
3. Temp: 29.000°C, 84.200°F of the device 28-9884b3116461

Reading temperature, number of sensors: 3
1. Temp: 25.250°C, 77.450°F of the device 28-c486b3116461
2. Temp: 25.250°C, 77.450°F of the device 28-55b7b3116461
3. Temp: 27.875°C, 82.175°F of the device 28-9884b3116461
^CScript end!
pi@raspberrypi:~/Scripts $
```

Um das Skript zu beenden, drücken Sie STRG + C auf der Tastatur.

Sie können das Skript problemlos für einen oder mehrere DS18B20-Sensoren verwenden.

**Sie haben es geschafft!**  
**Jetzt können Sie Ihr Modul für verschiedene Projekte verwenden.**



Jetzt ist es an der Zeit, zu lernen und die Projekte selbst zu erstellen. Das können Sie mit Hilfe von vielen Beispielskripten und anderen Tutorials, die Sie im Internet finden können.

**Wenn Sie auf der Suche nach hochwertiger Mikroelektronik und Zubehör sind, sind Sie bei der AZ-Delivery Vertriebs GmbH an der richtigen Adresse. Sie erhalten zahlreiche Anwendungsbeispiele, vollständige Installationsanleitungen, eBooks, Bibliotheken und Unterstützung durch unsere technischen Experten.**

<https://az-delivery.de>

Viel Spaß!

Impressum

<https://az-delivery.de/pages/about-us>