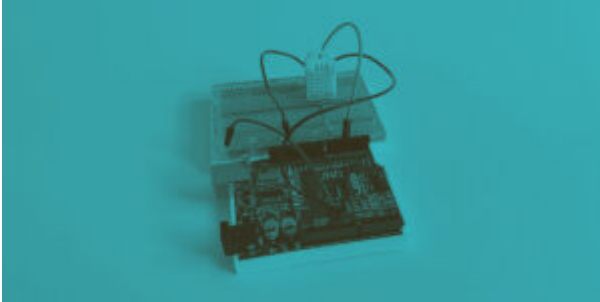


Mit dem DHT11 und DHT22 Temperatur und Luftfeuchtigkeit messen

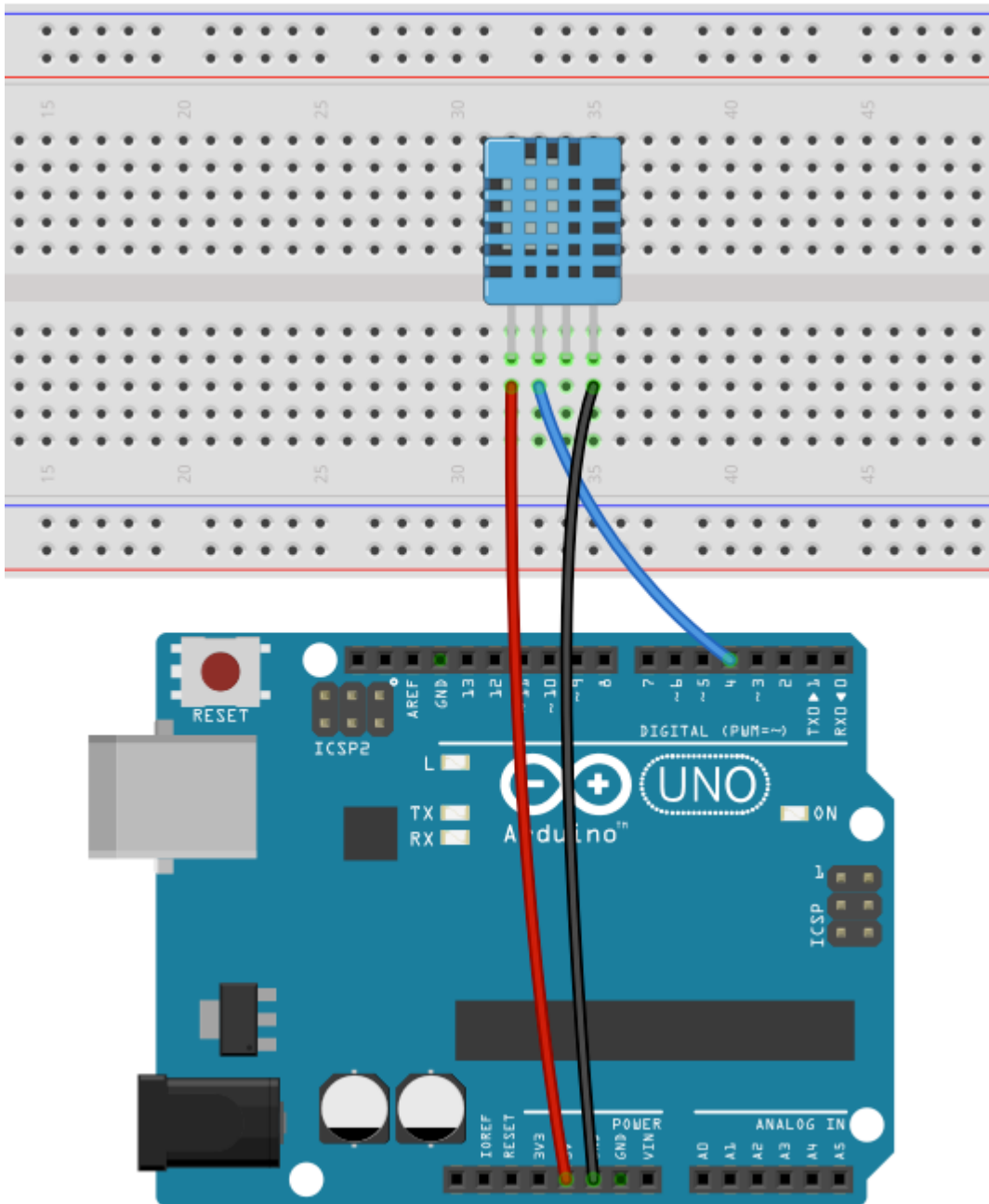


Für den Arduino und andere Microcontroller gibt es eine Vielzahl von Sensoren, die die [Temperatur](#) messen. Zwei davon sind der DHT11 und der DHT22, die darüber hinaus auch die Luftfeuchtigkeit ermitteln können.

In diesem Tutorial lernst du, wie du den DHT11 und den DHT22 am Arduino und ESP8266 anschließt und verwendest.

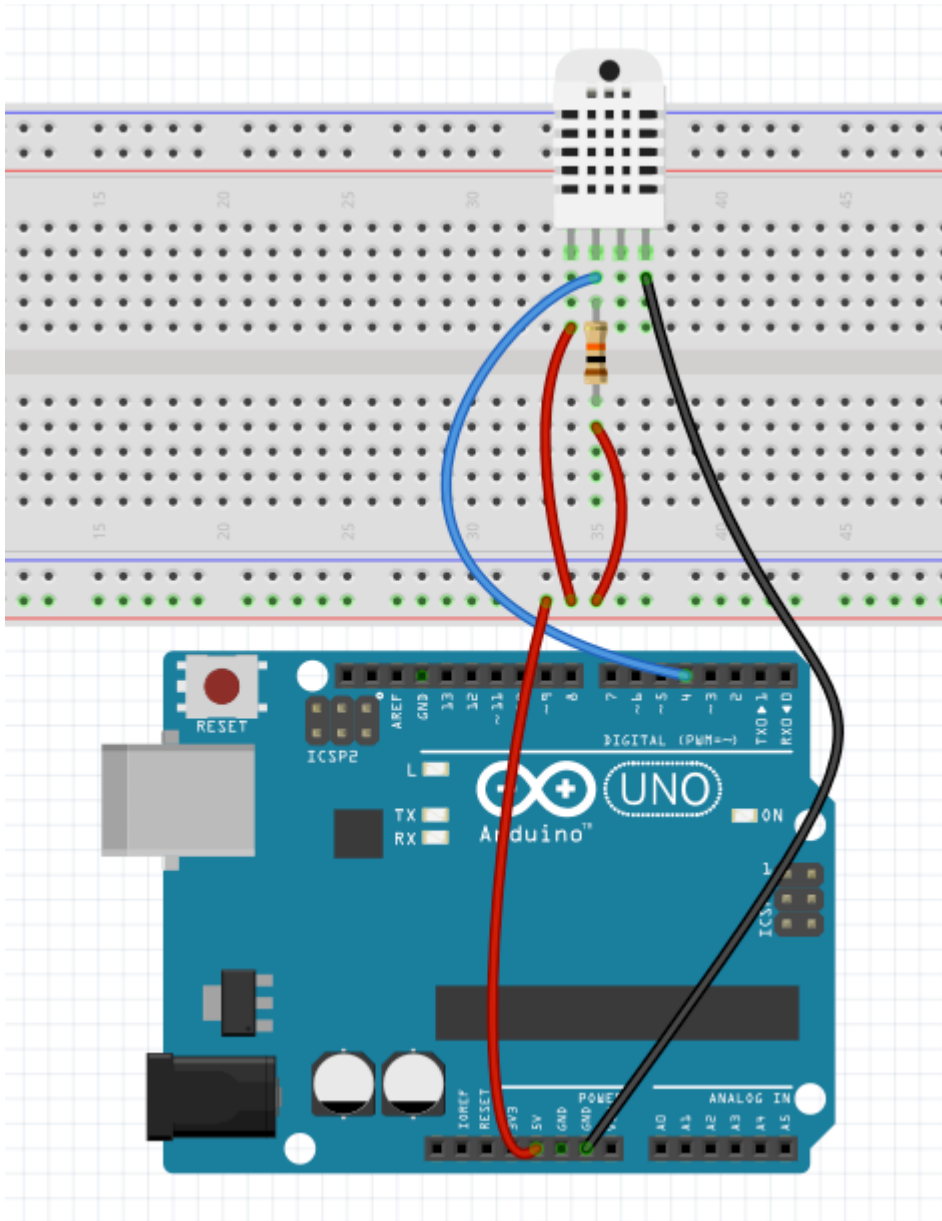
Den Sensor DHT11 anschließen

Der Anschluss des DHT11 am Arduino ist ganz leicht, du benötigst lediglich drei Kabel. Orientiere dich beim Anschluss an folgender Skizze. **An welchen Digital-Pin du deinen Sensor anschließt, steht dir natürlich frei.** Damit der Code dieses Tutorials passt, wähle bitte den Pin 4.



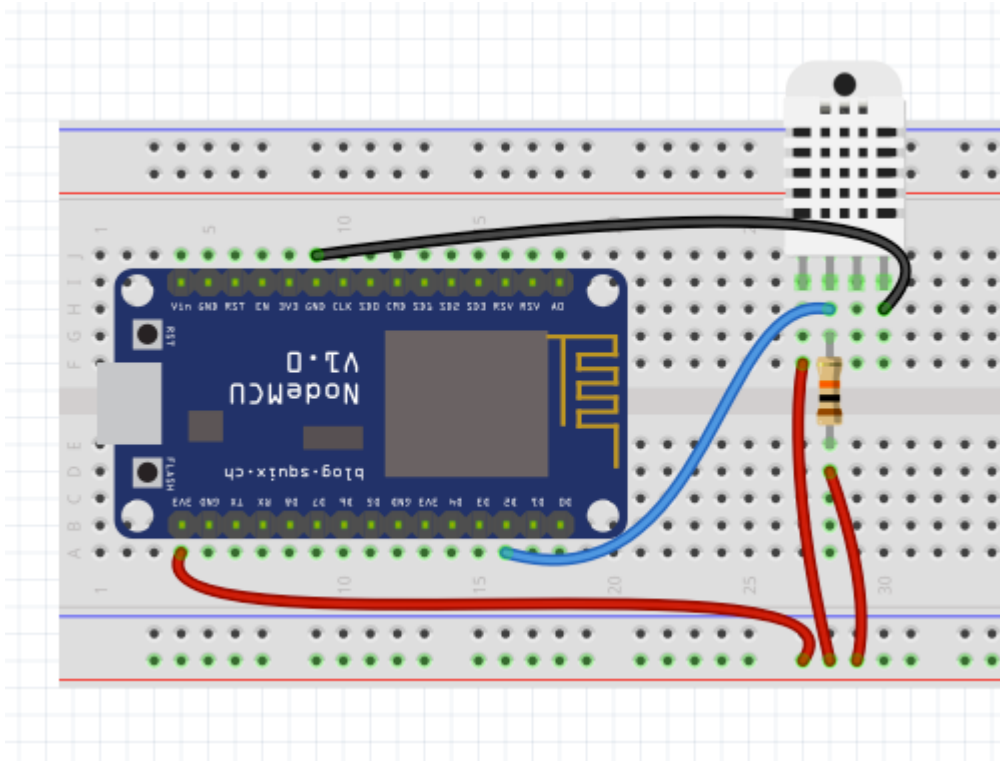
Den Sensor DHT22 anschließen

Um deinen DHT22 am Arduino oder ESP8266 anzuschließen, benötigst du neben Sensor, Microcontroller, Breadboard und Kabeln einen Widerstand mit 10 k Ω . Orientiere dich beim Anschluss am Arduino an folgender Skizze:



Hier ist der Signalpin des DHT22 am **Digital-Pin 4** des Arduinos angeschlossen – so passt der Anschluss zum Sketch unten. **Prinzipiell steht es dir natürlich frei, welchen Digital-Pin du verwendest.**

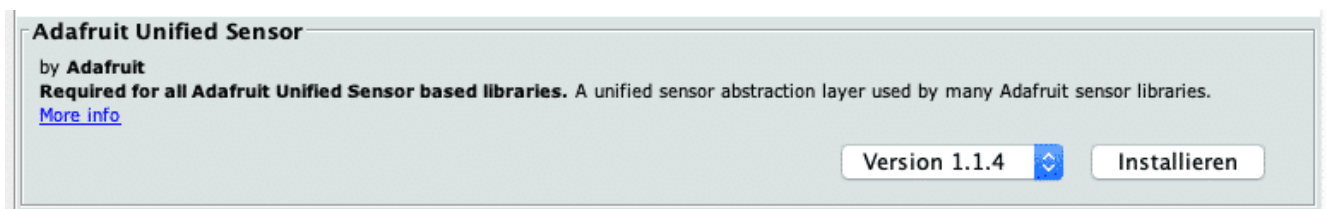
Der Anschluss am ESP8266 sieht folgendermaßen aus:



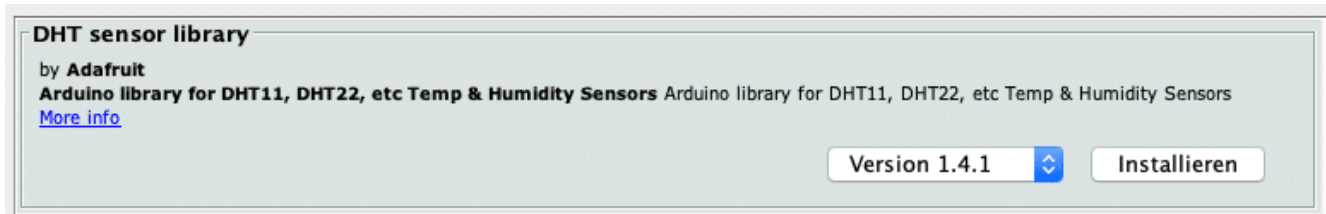
Beim Arduino kannst du den DHT11 oder DHT22 entweder mit 5 Volt oder 3,3 Volt versorgen. Beim ESP8266 stehen dir natürlich nur 3,3 Volt zur Verfügung.

Die passenden Bibliotheken

Du benötigst zwei Bibliotheken, von denen du jedoch nur eine im Sketch einbinden musst. Öffne zunächst deinen Bibliotheksmanager und suche dort nach **Adafruit Unified Sensor**. Installiere anschließend die aktuelle Version.



Suche nun nach **DHT11** oder **DHT22** und installiere die Bibliothek **DHT sensor library**.



Wenn du beide Bibliotheken erfolgreich installiert hast, geht es weiter mit der Messung.

Temperatur und Luftfeuchtigkeit messen

Binde als erstes die Bibliothek für den Sensor zu Beginn deines Sketchs ein:

```
#include "DHT.h"
```

Anschließend definierst du den Pin, an dem der Sensor angeschlossen ist. In der Zeile danach legst du den Typ des DHT fest: In unserem Fall also DHT22 – **wenn du einen DHT11 verwendest, kannst du die Zeile entsprechend abändern.**

```
#define DHTPIN 4  
#define DHTTYPE DHT22
```

Nun fehlen noch zwei Variablen für die Temperatur und die Luftfeuchtigkeit sowie ein Objekt für den Sensor namens **dht**.

```
float temp;  
float humidity;
```

```
DHT dht(DHTPIN, DHTTYPE);
```

In der Setup-Funktion startest du den Sensor mit

```
dht.begin();
```

Die eigentliche Messung findet dann im Loop statt. Diese könnte dank der eingebundenen Bibliothek kaum einfacher sein, da es passende Funktionen gibt. Die Messergebnisse speicherst du in den entsprechenden Variablen:

```
temp = dht.readTemperature();  
humidity = dht.readHumidity();
```

Und das war es schon. Hier nun der vollständige Sketch, in dem du die Messergebnisse alle zwei Sekunden im Seriellen Monitor ausgibst:

```
#include "DHT.h"  
  
#define DHTPIN 4  
#define DHTTYPE DHT22  
  
float temp;  
float humidity;  
  
DHT dht(DHTPIN, DHTTYPE);  
  
void setup() {  
  Serial.begin(9600);  
  dht.begin();  
}  
  
void loop() {  
  
  temp = dht.readTemperature();  
  humidity = dht.readHumidity();  
  Serial.print("Temperatur: ");  
  Serial.print(temp);  
  Serial.println(" *C");  
  Serial.print("Luftfeuchtigkeit: ");  
  Serial.print(humidity);
```

```
Serial.println(" %");  
  
Serial.println();  
delay(2000);  
  
}
```

Wie geht es weiter?

Wie wäre es mit einem eigenen [ESP8266 Web Server](#), über den du die aktuelle Temperatur vom Smartphone aus abrufen kannst?