

## Inhaltsfelder Informatiksysteme & Algorithmen

# Mit dem Arduino Sensoren zur Wettermessung programmieren – ein Projekt

Nach einer Idee von Thomas Rosenthal  
Mit Illustrationen von Dr. Wolfgang Zettlmeier



© RAABE 2021

© Paul Fleet/Stock/Getty Images Plus

In diesem Projekt mit dem Arduino programmieren die Schülerinnen und Schüler nach einer kurzen Wiederholung zu Wetter und Wetterelementen sowie den Grundbefehlen des Arduinos selbstständig einzelne Sensoren zur Messung von Temperatur, Luftfeuchtigkeit und Luftdruck, ein Display und eine Echtzeituhr für eine Wettermessung.

### KOMPETENZPROFIL

**Klassenstufe:** 7–10

**Dauer:** 8 Unterrichtsstunden (Minimalplan: 7)

**Lernziele:** Die Lernenden ... 1. programmieren einzelne Sensoren mithilfe des Arduino-Boards, 2. erstellen Schaltpläne, 3. führen ein Projekt selbstständig durch.

**Thematische Bereiche:** Informatiksysteme, Algorithmen, weiterführende Projekte

**Kompetenzbereiche:** Modellieren, Implementieren, Darstellen und Interpretieren, Kommunizieren und Kooperieren



## Auf einen Blick

### Benötigte Materialien

- 1 PC/Laptop pro Lernenden bzw. pro Schülerpaar
- 1 rote LED und ein 10-k $\Omega$ -Widerstand
- 11 kurze (male-male), 11 lange (male-male) und 4 lange (male-female) Jumperkabel
- ein 10-k $\Omega$ -Widerstand
- I<sup>2</sup>C-Display, RTC-Modul und LDR
- Sensor DHT 11 und Sensor BMP 180



### 1./2. Stunde

**Thema:** Grundlagen zum Thema Wetter und den Wetterelementen

**M 1** Welche Faktoren bestimmen das Wetter?

**M 1** Wetterelemente

### 3./4. Stunde

**Thema:** Ausgabe von Daten über den seriellen Monitor und über das I<sup>2</sup>C-Display.

**M 2** Ausgabe von Informationen über den seriellen Monitor

**M 3** Angabe der Helligkeit über einen lichtabhängigen Widerstand

**M 4** Einbindung von Bibliotheken und Ausgabe von Informationen über das I<sup>2</sup>C-Display

### 5./6. Stunde

**Thema:** Programmieren von Temperatur, Luftfeuchtigkeit und Luftdruck.

**M 6** Temperatur und Luftfeuchtigkeit programmieren

**M 7** Luftdruck programmieren

### 7./8. Stunde

**Thema:** Programmieren von Datum und Uhrzeit.

**M 8** Datum und Uhrzeit programmieren

**M 9** Wettermessung programmieren

## Angabe der Helligkeit über einen lichtabhängigen Widerstand

M 4

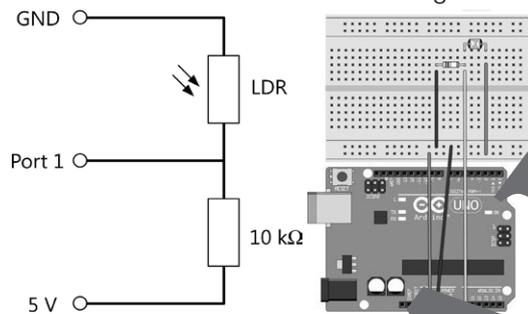
In modernen Häusern oder auch bei neueren Autos wird die Beleuchtung automatisch gesteuert. Bei Dämmerung möchte man das Licht noch nicht voll aufgedreht haben. Vielmehr soll die Helligkeit der Beleuchtung stufenweise den Lichtverhältnissen draußen angepasst werden. Um den Lichteinfall zu messen, verwendet man einen lichtabhängigen Widerstand, auch LDR (engl. Light Dependent Resistor) genannt. Der Widerstandswert eines LDR ändert sich, je nachdem, wie viel Licht auf ihn einfällt, d. h.: Je mehr Licht einfällt, desto geringer ist der Widerstandswert. Diese Werte können über einen sogenannten seriellen Monitor ausgegeben werden. Diese Eigenschaft lässt sich nun dazu verwenden, einen LDR als Lichtsensor zu verwenden. Dazu muss man mit ihm einen sogenannten Spannungsteiler aufbauen. Unser Lichtsensor ist ein solcher Spannungsteiler, der aus dem LDR als veränderlichem Widerstand und einem Festwiderstand besteht, wie es die obige Abbildung verdeutlicht (Port 1 = A0).

**Tip:** Informiere dich anhand des Infomaterials zum seriellen Monitor und zum Spannungsteiler.



### Aufgabe

Erkläre die Funktionsweise des LDR in eigenen Worten



Grafik: Dr. Wolfgang Zettlmeier

### //So geht's

- **Programmiere** die Helligkeitsschaltung und speichere den Sketch unter „01\_Helligkeit\_seriell“.

```

1 int LDR = 0; //Sensoreingang
2 int Helligkeit; //Wert des LDR
3
4 void setup()
5   Serial.begin(9600); //Starten der seriellen Ausgabe
6
7
8 void loop()
9   Helligkeit = analogRead(LDR);
10  //Lesen der Messwerte
11  Serial.println(Helligkeit); //Anzeigen der Werte im seriellen
12 //Monitor
13  delay(200);
14 }
```



## M 5

## Einbindung von Bibliotheken und Ausgabe von Informationen über das I<sup>2</sup>C-Display

Buchstaben und Ziffern lassen sich über ein LCD-Display ausgeben. Dies kann in vielen Anwendungen sehr nützlich sein, um zum Beispiel Messwerte darzustellen.

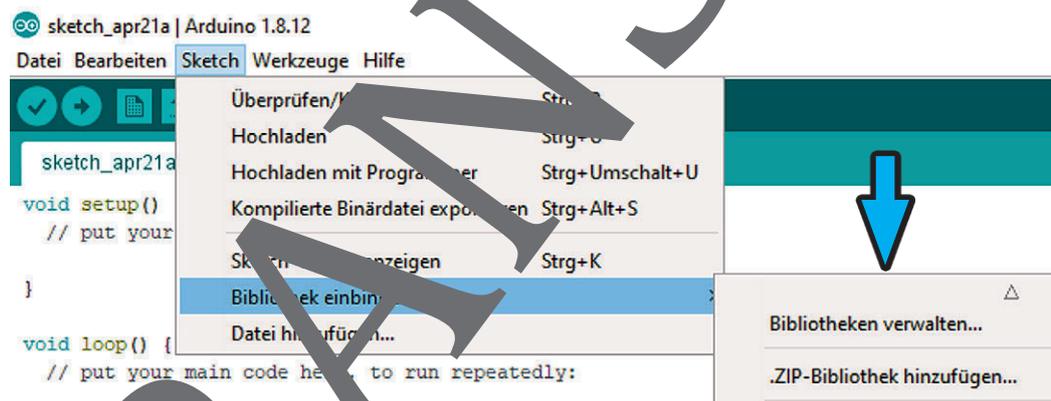
Mithilfe des LCD-Displays lassen sich darüber hinaus auch Daten darstellen, wenn beispielsweise der Arduino nicht mit einem PC verbunden ist. Das LCD-Display mit einem aufgelöteten I<sup>2</sup>C-Bus ermöglicht eine einfache Verkabelung, was bei komplexeren Projekten vorteilhaft ist.

Auf der Rückseite des Displays befindet sich ein Drehregler, mit dem sich der Kontrast regulieren lässt. Die vom Programm her anzusteuernde I<sup>2</sup>C-Adresse, eine HEX-Zahl, kann unterschiedlich sein und muss im Sketch entsprechend angegeben werden.

Die typischen I<sup>2</sup>C-HEX-Adressen sind „0x27“ und „0x3F“. Für die Verwendung des I<sup>2</sup>C-Displays ist das Einbinden einer Bibliothek in die Arduino-Software notwendig. Bei Bibliotheken handelt es sich um Programmbausteine, die die Programmierung von Komponenten wesentlich vereinfachen, indem Sie den Umfang des Programmes deutlich reduzieren können.

### Einbinden einer Bibliothek

Wähle nun „Arduino > LoadCrystal-I2C-library-master“ vom USB-Stick aus und lade sie ein.



# Sie wollen mehr für Ihr Fach?

## Bekommen Sie: Ganz einfach zum Download im RAABE Webshop.



**Über 5.000 Unterrichtseinheiten**  
sofort zum Download verfügbar



**Webinare und Videos**  
für Ihre fachliche und  
persönliche Weiterbildung



**Attraktive Vergünstigungen**  
für Referendar:innen  
mit bis zu 15% Rabatt



**Käuferschutz**  
mit Trusted Shops



Jetzt entdecken:  
**www.raabe.de**