

Arduino-Bauanleitung: Luftdruckmessgerät

Die Mars-Station benötigt ein Luftdruckmessgerät. Mit der nachstehenden Anleitung lässt sich ein solches Gerät kostengünstig herstellen.

Für den Bau eines **Luftdruckmessgerätes** mit einem **Arduino Uno** werden folgende **Materialien** benötigt:

1. Arduino Uno
2. Luftdruckmesser BMP 180
3. Experimentier-Steckboard
4. LCD-Display 16x2
5. Steckbrückenkabel
6. Drehpotentiometer
7. Computer/ Laptop/ iPad mit einem USB

Einrichtung des Arduino Uno

1. Arduino IDE öffnen

Lade die Arduino-Software von der offiziellen Website herunter und installiere sie auf deinem Computer. Unter folgendem Link kann man das Arduino Programm herunterladen:

<https://www.arduino.cc/en/software>

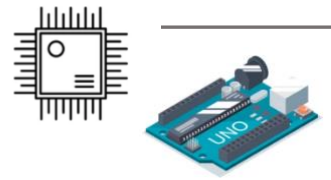


2. Verbindung zum Computer herstellen

Verbinde das Arduino Uno Board mit deinem Computer über das USB-Kabel.

3. Board und Port auswählen

Du musst in der Arduino IDE das richtige Board und den richtigen Port auswählen. Um das Board auszuwählen, gehe zu Werkzeuge --> Board --> Arduino AVR Boards und wähle Arduino Uno. Um den Port auszuwählen, gehe zu Werkzeuge --> Port und wähle den Port aus, an den dein Arduino Uno angeschlossen ist. Er wird in der Regel als "COM" angezeigt. Teste die Verbindung, indem du ein Beispielprogramm wie "Blink" aus dem Menü "File" --> "Examples" öffnest und auf das Board hochlädst.



4. Bibliotheken installieren:

Installiere die benötigten Bibliotheken in der Arduino IDE. Das funktioniert, indem du in der IDE auf "Sketch" -> "Bibliothek einbinden" -> "Bibliotheken verwalten" gehst und dann nach "Wire", "LiquidCrystal" und "Adafruit BMP085" suchst. Installiere diese Bibliotheken.

5. Code einfügen

Wenn du einen eigenen Code schreiben willst, findest du hilfreiche Tipps auf Seite 5. Andernfalls kannst du den nachfolgenden Code problemlos kopieren und in die Arduino IDE einfügen.

6. Code hochladen

Jetzt bist du bereit, deinen Code auf den Arduino Uno hochzuladen. Öffne den Sketch (dein Arduino-Programm) in der Arduino IDE und klicke auf den Hochladen-Button, der wie ein Pfeil nach rechts aussieht und sich in der oberen linken Ecke des Fensters befindet. Die IDE wird deinen Code kompilieren und auf den Arduino Uno hochladen. Sobald der Upload abgeschlossen ist, sollte dein Arduino Uno beginnen, den hochgeladenen Code auszuführen.

7. Zusammenbau des Arduino Uno

Zunächst wird das Display wie in der folgenden Skizze an den Arduino angeschlossen

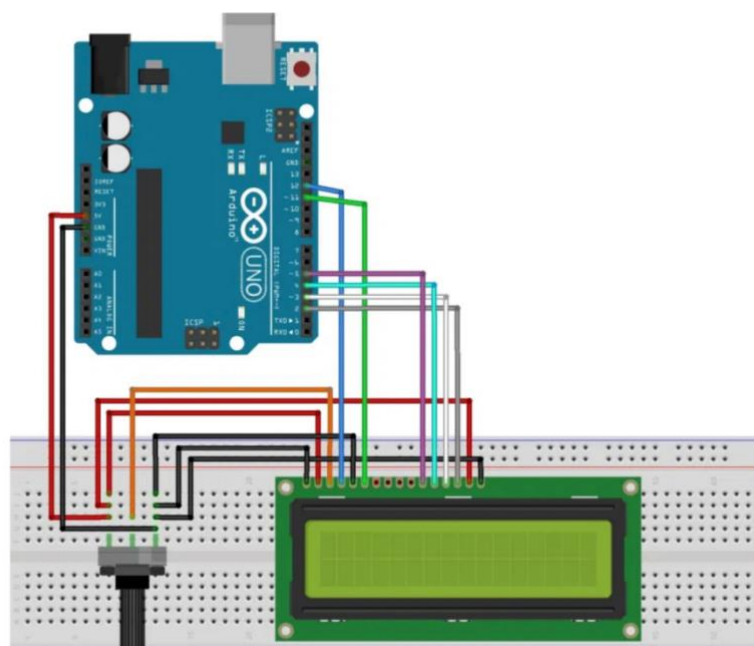
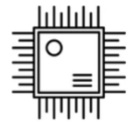


Abbildung 1: Quelle: <https://funduino.de/nr-13-lcd-display>



Anschließend muss noch der Luftdrucksensor wie auf der Abbildung 2 (unten) mit dem Arduino verbunden werden.

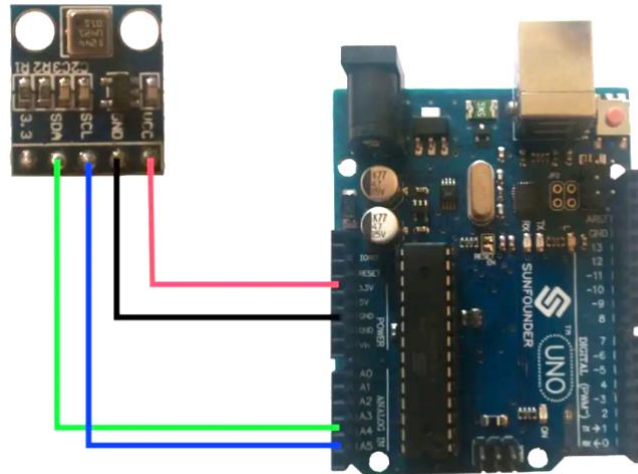


Abbildung 2: Quelle: <https://draeger-it.blog/sainsmart-lektion-10-bmp180/>

Funktionsweise des Luftdruckmessgerätes

Stell dir den **BMP085-Sensor** als winzige **Federwaage** vor, die sich je nach dem umgebenden **Luftdruck** bewegt. Wenn der **Luftdruck** hoch ist, drückt er die Feder zusammen. Wenn der Luftdruck niedrig ist, entspannt sich die Feder.

Dieser **Sensor** misst die "Stärke" der Feder und wandelt diese in eine **elektrische Spannung** um. Diese **Spannung** ändert sich je nach dem **Luftdruck**. Aber der **Arduino** kann diese **Spannung** nicht direkt lesen. Deshalb muss die **Spannung** in ein digitales **Signal umgewandelt** werden, das der **Arduino** verstehen kann.

Nachdem das **Signal digitalisiert** wurde, wird es über eine spezielle "Kommunikationslinie" (genannt I2C) an den **Arduino** gesendet. Dann kann der **Arduino** dieses **digitale Signal** lesen und in einen **Druckwert umwandeln**, den wir verstehen können.

Da der Sensor den Druck in Pascal misst und wir den Druck normalerweise in **Hektopascal** (hPa) angeben, teilen wir den gemessenen Wert einfach durch 100, um ihn in hPa umzurechnen. Dieser **Druckwert** wird dann auf dem **LCD-Bildschirm** angezeigt und auch an den **Computer gesendet**. Das Ganze passiert etwa **einmal pro Sekunde**, sodass wir immer den **aktuellen Druckwert** sehen können.



Mars-Erforschung mit dem selbstgebauten Luftdruckmessgerät

Das **selbstgebaute Luftdruckmessgerät** ermöglicht es uns, das **Mars-Modell** zu analysieren. Ein Experiment untersucht, warum es auf dem **Mars kein flüssiges Wasser** gibt. Dafür wird ein mit **heißem Wasser** (80°C) gefülltes Becherglas unter eine **Glasglocke** gestellt. Durch den Einsatz eines **Kompressors** wird der **Luftdruck** unter der Glocke **reduziert**, wodurch das Wasser zu **kochen** beginnt. Unser eigenes **Luftdruckmessgerät** erlaubt es uns, den stetig **sinkenden Luftdruck** unter der Glocke zu beobachten.

Der Selbstbau eines Luftdruckmessgeräts bietet diverse Vorteile. Herkömmliche **Messgeräte** sind in der Regel **teuer**, unser selbstgebautes Messgerät ist eine **kostengünstige Alternative**. Zusätzlich ermöglicht der eigenständige Bau des Messgeräts eine **Verknüpfung** verschiedener **Unterrichtsfächer**, was zum Konzept des **fächerübergreifenden Unterrichts** beiträgt.

Dieser Unterrichtsansatz weist viele Vorteile auf, unter anderem fördert er bei den Schüler:innen die Fähigkeit, **komplexe Probleme** durch **vernetzendes Denken und fachübergreifende Strukturierung** zu lösen. Ein weiterer Vorteil besteht darin, dass er **realitätsbezogenes** Handeln ermöglicht, das nicht an Fachgrenzen gebunden ist und somit fächerübergreifend es Fachwissen verknüpft wird.



Anleitung zur Programmierung des Arduino Uno

1. Erlernen der Grundlagen von C++

Arduino-Code basiert auf C++, daher wäre es hilfreich, dich mit den Grundlagen dieser Programmiersprache vertraut zu machen. Es gibt viele Ressourcen online, einschließlich Tutorials und Videos, die du verwenden kannst.

2. Verständnis der Arduino-Grundlagen

Nimm dir etwas Zeit, um die Grundlagen von Arduino zu lernen, einschließlich dessen, was die `setup()` und `loop()` Funktionen bedeuten, wie du mit digitalen und analogen Pins umgehst, und wie du Sensorwerte liest.

3. Bibliotheken installieren

Für dieses spezielle Projekt benötigst du drei Bibliotheken: `Wire`, `LiquidCrystal` und `Adafruit_BMP085`. Du kannst diese in der Arduino IDE installieren, indem du zu `Sketch --> Include Library --> Manage Libraries` gehst und nach den entsprechenden Bibliotheken suchst.

4. Programmieren des Drucksensors

Jetzt kannst du anfangen, den Code für dein Projekt zu schreiben. Beginne mit der Einbindung der benötigten Bibliotheken und der Initialisierung des Sensors und des LCDs. Dann konzentriere dich auf die `setup()` Funktion, in der du den Sensor initialisierst und überprüfst, ob er korrekt funktioniert.

5. Luftdruck lesen und anzeigen

In der `loop()` Funktion solltest du den Luftdruck vom Sensor lesen, ihn in Hektopascal umrechnen und auf dem LCD-Display ausgeben. Du kannst auch den Wert auf der seriellen Konsole ausgeben, um zu überprüfen, ob alles korrekt funktioniert.

6. Code überprüfen und hochladen

Sobald du deinen Code fertig geschrieben hast, überprüfe ihn auf Fehler und lade ihn dann auf dein Arduino-Board hoch. Du kannst dies tun, indem du das Board mit einem USB-Kabel an deinen Computer anschließt und dann den Upload-Button in der Arduino IDE drückst.



7. Testen und Fehlerbehebung

Nachdem dein Code auf das Board hochgeladen wurde, solltest du überprüfen, ob alles wie erwartet funktioniert. Wenn etwas nicht funktioniert, gehe zurück und überprüfe deinen Code und deine Verdrahtung.

Tipp: Es kann hilfreich sein, den Code in kleine Teile zu zerlegen und jeden Teil einzeln zu testen. Dies kann dir dabei helfen, Fehler leichter zu finden und zu beheben.

Viel Spaß beim Programmieren!

Fehlersuche

Sollte das fertiggestellte **Luftdruckmessgerät nicht** ordnungsgemäß **funktionieren**, wäre es ratsam, den Programmcode erneut zu **überprüfen**. Darüber hinaus könnten **fehlerhafte Steckverbindungen** ebenfalls ein Problem darstellen, daher lohnt es sich, diese noch einmal gründlich zu kontrollieren. Achte auch darauf, das **Potentiometer** korrekt **einzustellen**, damit du den angezeigten Text deutlich erkennen kannst.

Falls das Gerät weiterhin keine Messwerte liefert, besteht die Möglichkeit, sich die Daten **digital anzeigen** zu lassen. Der dafür notwendige Code wurde bereits eingefügt. Du musst nur den **Serieller Monitor** öffnen das gibt auf der **Lupe oben rechts**. Solltest du keine Daten auf deinem Computer erhalten, könnte der **Sensor defekt** sein oder es besteht ein Problem mit den **Steckverbindungen**. Stelle sicher, dass du alle Verbindungen nochmals überprüft hast, bevor du den Sensor als defekt einstuft.

Wenn dir jedoch **digitale Werte** angezeigt werden, ist möglicherweise das **Display fehlerhaft**. Überprüfe noch einmal alle **Steckverbindungen** zum Display. Beim Aufbau eines solchen Systems kann leicht ein Fehler passieren. Falls das Display nicht leuchtet, könnte es beschädigt sein und muss möglicherweise ersetzt werden.

Diese Fehlersuche soll dir helfen, mögliche Probleme zu erkennen und zu beheben. Mit **Geduld** und **Sorgfalt** beim Aufbau und bei der Fehlersuche sollte dein Luftdruckmessgerät bald einwandfrei **funktionieren**.



Quell-Code:

```
#include <Wire.h>
// Es werden mehrere Bibliotheken eingebunden: Wire.h für die I2C- Kommunikation,

#include <LiquidCrystal.h> // LiquidCrystal.h zur Steuerung des LCD-Displays
#include <Adafruit_BMP085.h>
// und Adafruit_BMP085.h für die Kommunikation mit dem BMP085-Sensor.

Adafruit_BMP085 bmp;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int hPa = 1;
// Eine Instanz des BMP085-Sensors (bmp) und des LCD-Displays (lcd) wird erstellt. Das LCD-Display ist
// so konfiguriert, dass es an den Pins 12, 11, 5, 4, 3 und 2 angeschlossen ist.

void setup(){
  lcd.begin(16, 2);
  //beginn der seriellen Kommunikation mit 9600 Baud
  Serial.begin(9600);
  lcd.setCursor(0, 0);
  //Wenn kein Sensor gefunden wurde dann...
  if (!bmp.begin()) {
    Serial.println("Es wurde kein Sensor gefunden, bitte prüfe die Verkabelung!");
    lcd.print("Fehler");
    //eine Endlosschleife
    //Dieses sorgt dafür das die Loop nicht erreicht wird!
    while (1) {}
  }
}
//Im setup() wird das LCD-Display initialisiert und die serielle Kommunikation mit 9600 Baud gestartet.
//Dann wird versucht, eine Verbindung zum BMP085-Sensor herzustellen. Wenn keine Verbindung
//möglich ist, wird eine Fehlermeldung über die serielle Schnittstelle ausgegeben und auf dem LCD-
//Display angezeigt. Danach bleibt das Programm in einer Endlosschleife stecken, um die loop()-Funktion
//zu verhindern.

void loop(){

  hPa = bmp.readPressure()/100;
  //Ausgabe des Luftdrucks auf der seriellen Schnittstelle
  lcd.setCursor(0, 0);
  lcd.print("Luftdruck:");
  lcd.setCursor(0, 1);
  lcd.print(hPa);
  lcd.setCursor(3, 1);
  lcd.print("hPa");
  Serial.print("Luftdruck = ");
  Serial.print(hPa);
  Serial.println(" hPa");
  delay(1000); //Kurze Pause von 1 sek.
}
```



//In der loop()-Funktion wird der Luftdruck vom Sensor gelesen und in Hektopascal (hPa) umgerechnet, indem der Wert durch 100 geteilt wird. Dieser Wert wird dann auf dem LCD-Display angezeigt und über die serielle Schnittstelle ausgegeben.

//Die loop()-Funktion wartet dann eine Sekunde (1000 Millisekunden) bevor sie erneut ausgeführt wird, was dazu führt, dass die Druckmessung und Ausgabe einmal pro Sekunde erfolgt.

Unter den Folgenden Links findet man alle benötigten Baumaterialien :

<https://www.reichelt.de/d1-shield-temperatur-luftdrucksensor-bmp180-d1z-sens-bmp180-p266072.html>

<https://www.reichelt.de/arduino-kompatibles-uno-r3-board-dip-variante-atmega328-usb-arduino-uno-dip2-p161029.html>

<https://www.reichelt.de/entwicklerboards-display-16-x-2-zeichen-blau-debo-lcd16x2-bl-p335002.html>

<https://www.reichelt.de/entwicklerboards-steckbrueckenkabel-40-pole-m-m-15-cm-debo-kabelset11-p282693.html>

<https://www.reichelt.de/drehpotentiometer-10-kohm-linear-6-mm-pih-pc16ip061033-p232701.html>

<https://www.reichelt.de/experimentier-steckboard-640-200-kontakte-steckboard-1k2v-p67678.html>

Quellen:

https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.shutterstock.com%2Fsearch%2Farduino-icon&psig=AOvVaw1VYc-WtxdaJwOm5B6XqCxi&ust=1686165027474000&source=images&cd=vfe&ved=0CBMQjhxqFwoTCNiPwa2sr_8CFQAAAAAdAAAAABAE

https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.shutterstock.com%2Fde%2Fsearch%2Farduino&psig=AOvVaw1VYc-WtxdaJwOm5B6XqCxi&ust=1686165027474000&source=images&cd=vfe&ved=0CBMQjhxqFwoTCNiPwa2sr_8CFQAAAAAdAAAAABAO

<https://funduino.de/nr-13-lcd-display>

https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.flaticon.com%2Ffree-icon%2Fmars_1146292&psig=AOvVaw3ZQOhfvfOObyW7QqfG-Z5Y&ust=1686167357582000&source=images&cd=vfe&ved=0CBMQjhxqFwoTCIDOXIS1r_8CFQAAAAAdAAAAABAE

<https://draeger-it.blog/sainsmart-lektion-10-bmp180/>

https://www.flaticon.com/de/kostenloses-icon/programmierung_291056