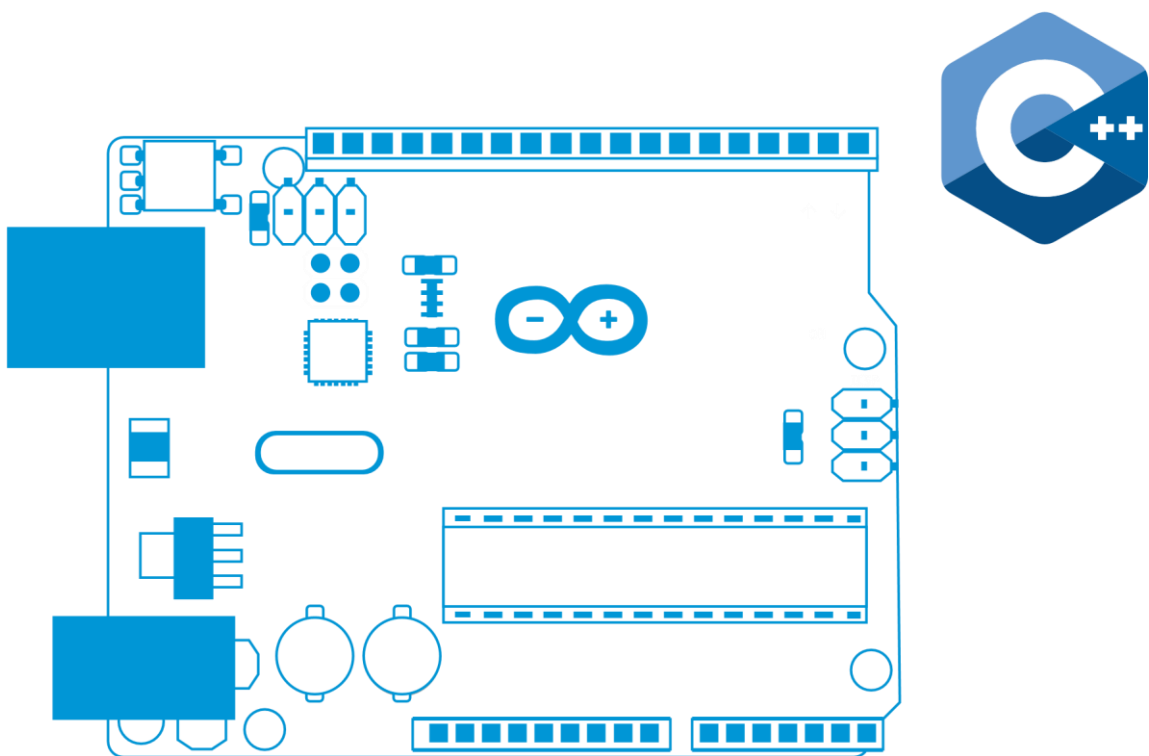
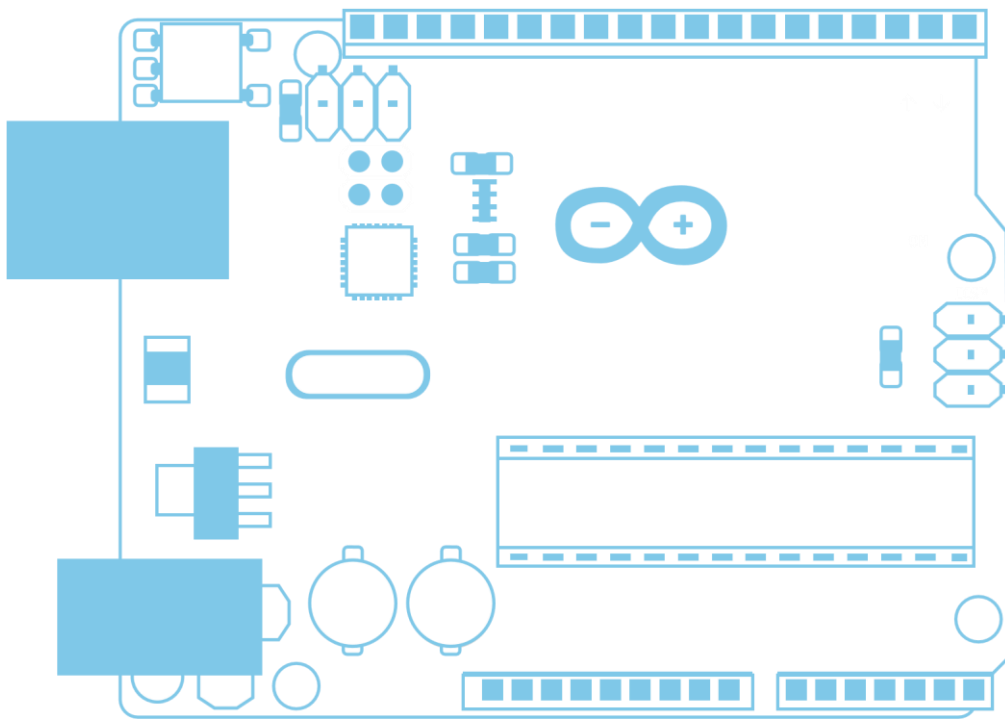


Lehren mit dem AI

→ Lernen mit ARDUINO!

Einführung in die Datenverarbeitung mit Arduino in C++





Aufgabe 0: Erste Schritte	Seite 5
Aufgabe 1: Mach mich an!	Seite 8
Aufgabe 2: Signalisiere ein S.O.S.!	Seite 11
Aufgabe 3: Temperaturmessung	Seite 12
Aufgabe 4: Druckmessung	Seite 16
Aufgabe 5: Höhenmessung	Seite 18

Lehren mit dem All – Lernen mit Arduino | T04.1
www.esa.int/education

Das ESA Education Office freut sich über Feedback und Kommentare
teachers@esa.int

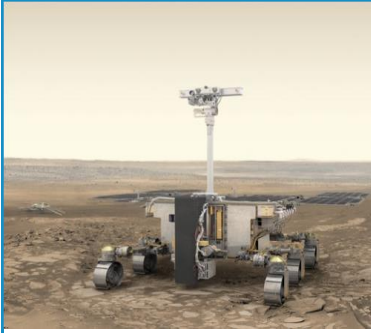
Eine Produktion von ESA Education
Copyright 2018 © European Space Agency

Eine Übersetzung von ESERO Germany

→ Lernen mit ARDUINO!

Einführung in die Datenverarbeitung mit C++

Wusstest du schon ...?



Die für 2020 angesetzte Mission des ExoMars Programms wird einen europäischen Rover sowie eine russische, wissenschaftliche Nutzlast zur Oberfläche des Mars bringen, um herauszufinden, ob dort jemals Leben existiert hat!

→ Aufgabe 0: Erste Schritte

Jeder nutzt täglich Technologie! Bei Weltraummissionen verwenden Ingenieure Technologie, um Roboter zu steuern und mit ihnen zu kommunizieren und so wird unser wissenschaftliches Wissen stetig erweitert. Bei dieser Übungsreihe wirst du zum Ingenieur. Du untersuchst den roten Planeten mithilfe eines technologischen Tools, dem Arduino. Und nun lass uns herausfinden, wie man die Komponenten des Arduino Baukastens identifiziert!

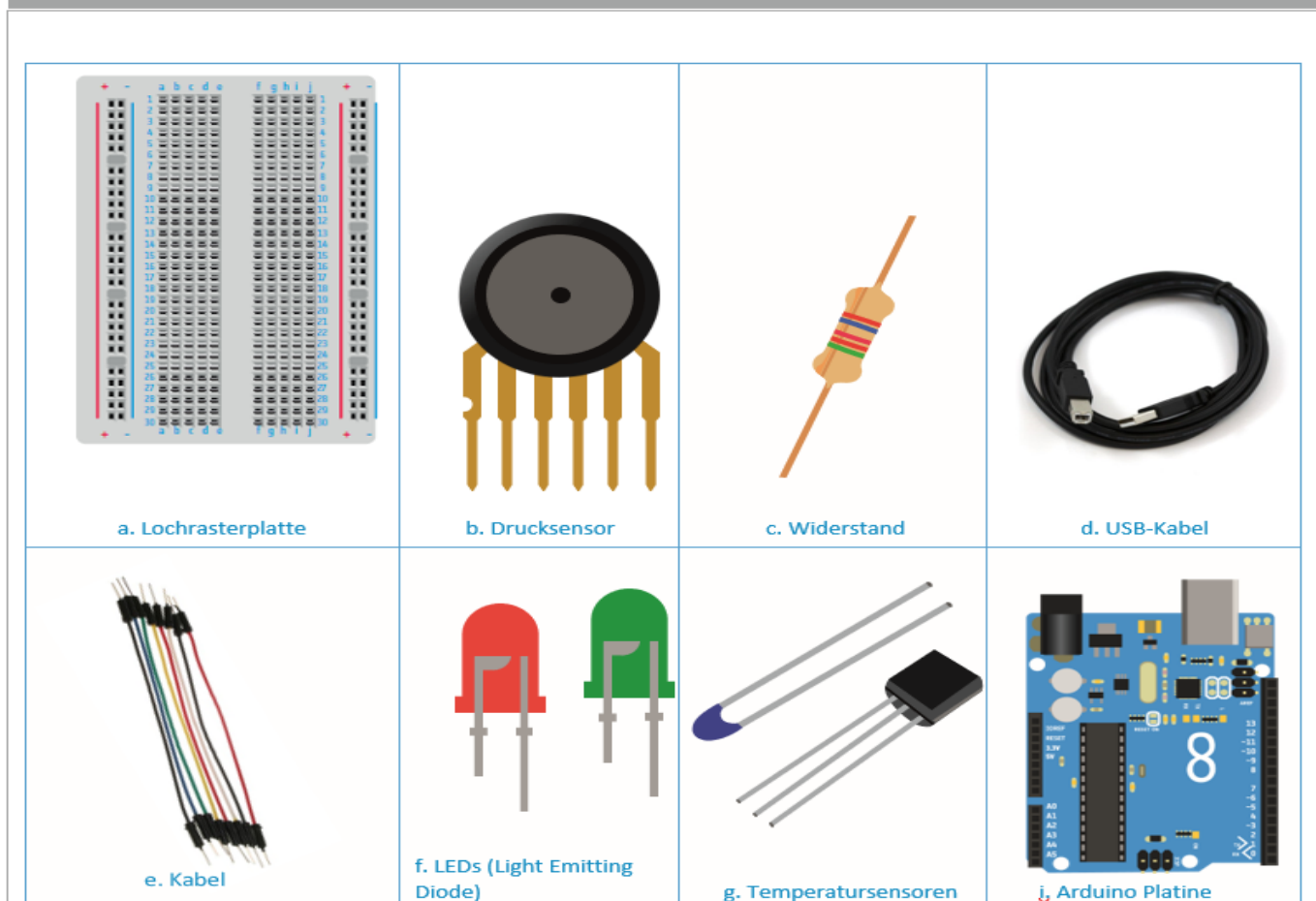
Equipment

- Der Arduino Basis-Baukasten

Bodenstation-Komponenten:

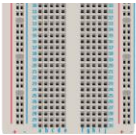
1. Arbeitet in Zweierteams. Öffnet eure Arduino -Bausätze und vergleicht die Komponenten mit den Beschreibungen auf der nächsten Seite.

Abbildung A1

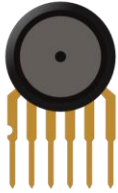


↑ Arduino Grundausrüstung

KOMPONENTEN



Lochrasterplatte



Drucksensor



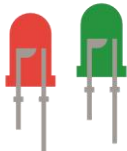
Widerstand



USB-Kabel



Kabel



LEDs (Light Emitting Diode)



Temperatursensoren



Arduino Platine

BESCHREIBUNG

Eine weiße Trägerplatte, mit der man einfach elektronische Komponenten miteinander verbinden kann.

Hat eine runde Form und misst den Umgebungsdruck.

Zylindrische Form mit farbigen Bändern. Reduziert die Spannung in einem Schaltkreis.

Ein langes, schwarzes Kabel, um den Arduino mit einem Computer zu verbinden.

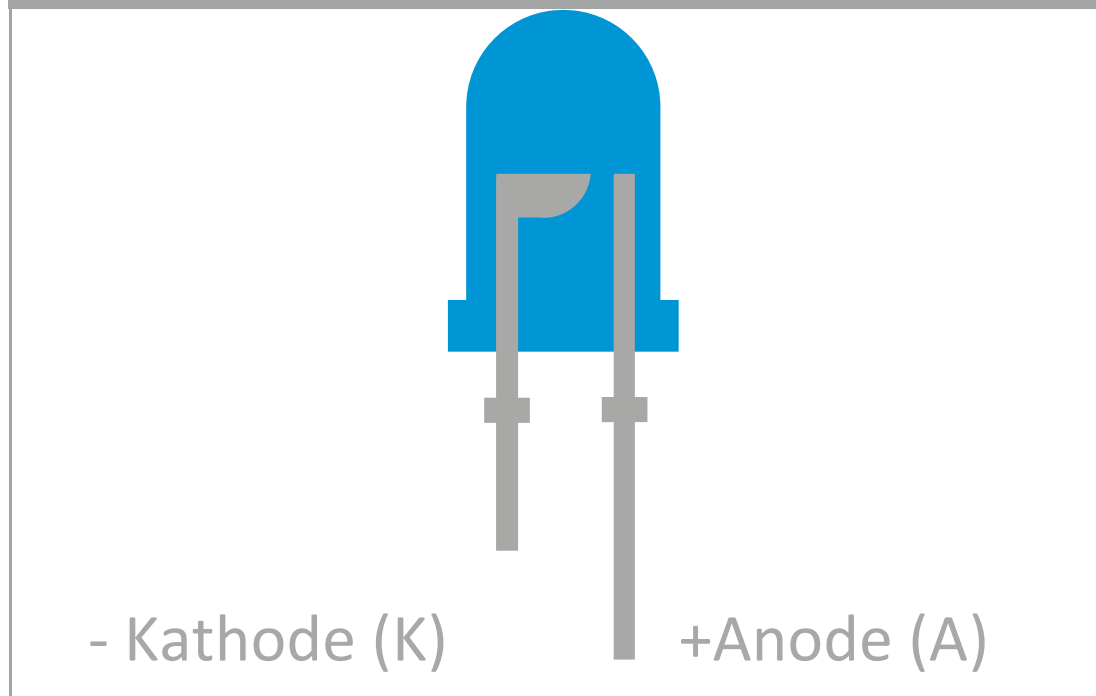
Haben unterschiedliche Farben und werden dazu verwendet, um Komponenten miteinander zu verbinden und mit Strom zu versorgen.

Sind rot und grün und strahlen Licht ab, wenn Strom durch sie hindurchfließt.

Hat zwei lange Metallstifte, die mit einem blauen Kopf verbunden sind. Misst Temperatur.

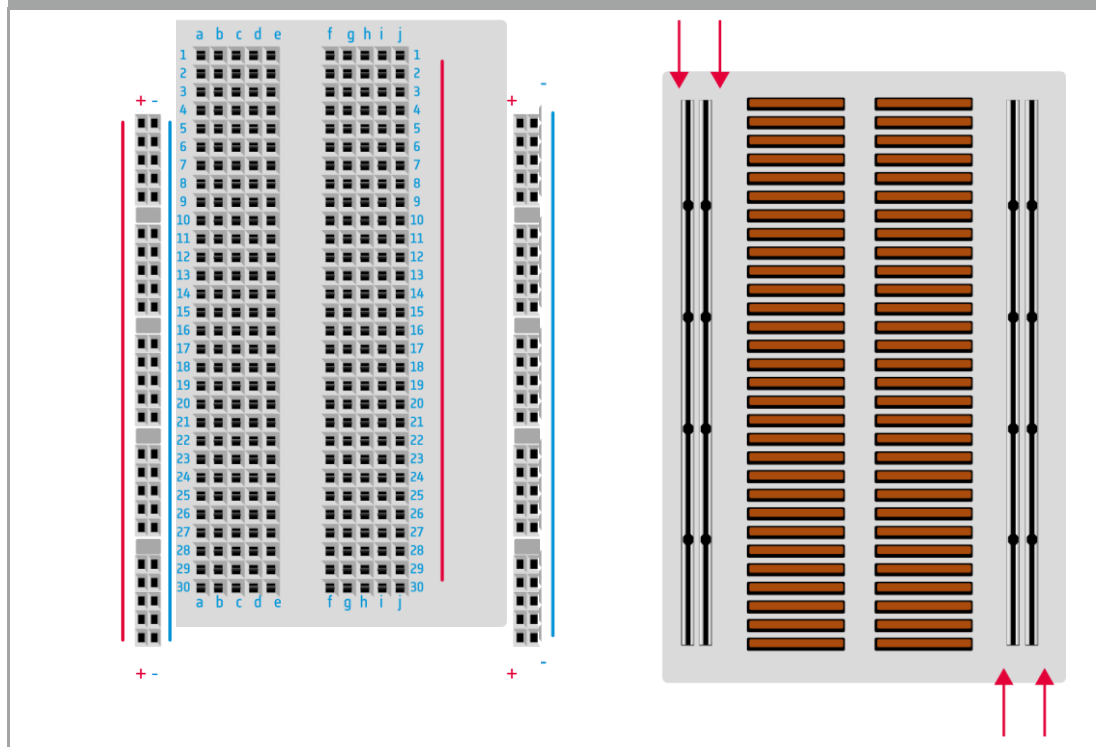
Hat Input- und Output-Pins und funktioniert wie ein einfacher Computer.

Abbildung A3



↑ LED: Beachte, dass die LED einen langen (Anode) und einen kurzen Metallstift (Kathode) hat.

Abbildung A3



↑ Lochrasterplatte: Beachte, dass alle Pins in jeder Reihe miteinander verbunden sind.

→ Aufgabe 1: Mach mich an!

Um interaktive Objekte zu steuern, müsst ihr ein Tool wie z.B. das Arduino verwenden, das aus mehreren elektronischen Komponenten besteht. Da ihr die Ingenieure seid, die dafür verantwortlich sind, den roten Planeten zu studieren, müsst ihr in der Lage sein, mit dem Rover zu kommunizieren, um seine Experimente zu koordinieren und die von ihm erhobenen Daten auszulesen. In der vorliegenden Aufgabe werdet ihr die C++ Programmiersprache erlernen, damit ihr mit eurem Arduino Uno kommunizieren könnt und die Leuchten am Arduino steuern könnt!

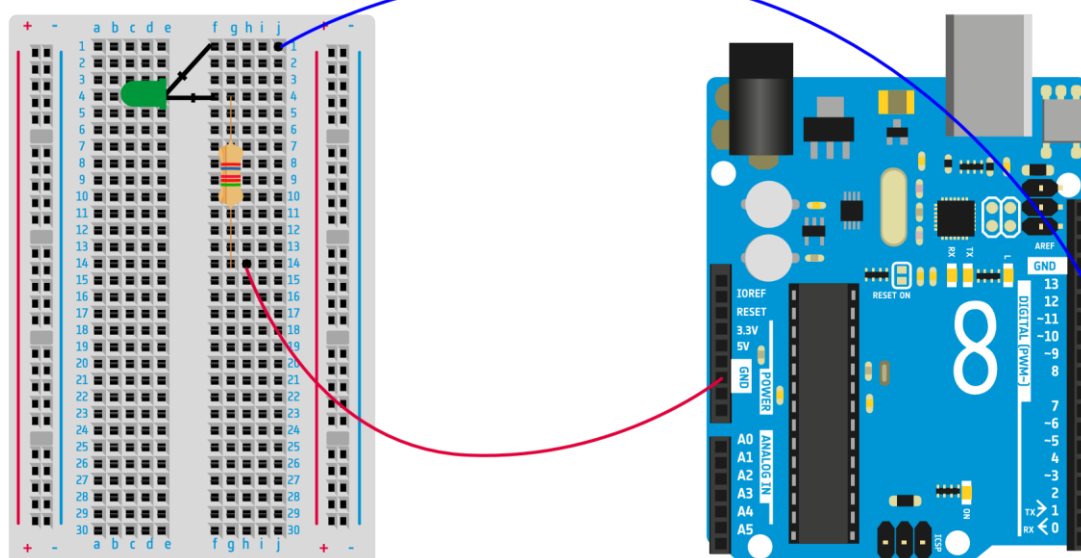
Komponenten

- 1 Arduino Uno
- 1 Lochrasterplatte
- 1 grüne LED
- 1 Widerstand 220Ω (rot/rot/schwarz/schwarz/braun)
- 2 Kabel

Übung

1. Befolgt die folgende Anleitung sorgfältig, um die Lochrasterplatte richtig aufzubauen, damit ihr die grüne LED kontrollieren könnt:
 - a. Verbindet die grüne LED (durchsichtiges Plastik) mit der Lochrasterplatte so wie in Abb. A4.
 - b. Verbindet den 220Ω Widerstand (rot/rot/schwarz/schwarz/braun) mit dem kurzen Metallstift der LED wie in Abb. A4.
 - c. Verbindet den langen Metallstift der LED mithilfe eines Kabels mit Pin **13** auf dem Arduino Uno.
 - d. Verbindet den Widerstand mit dem **GND** (ground) Pin des Arduino Uno mithilfe eines Kabels.
 - e. Verbindet das Arduino Uno mithilfe des USB-Kabels mit einem Computer.

Abbildung A4



↑ Arduino Schaltkreis zum Einschalten der grünen LED

2. Zeichnet in der unteren Box den elektrischen Schaltkreis auf, der den Verbindungen in Abb. A4 entspricht. Verwendet dabei die folgenden Symbole:



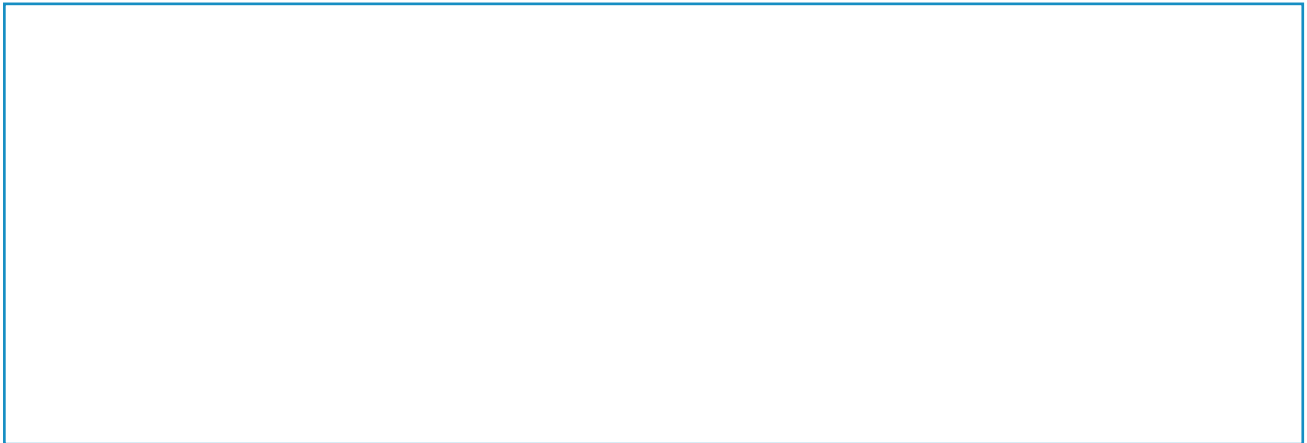
LED



Resistor



Ground



3. Jetzt, da die Lochrasterplatte bereit ist, müsst ihr die Befehle an das Arduino Uno schicken, um die grüne LED aufleuchten zu lassen.

Öffnet die Arduino Software (IDE) auf dem Computer und klickt auf File → Examples → Basics → Blink. Der Blink-Quelltext wird nun auf dem Bildschirm, so wie in der untenstehenden Abbildung, angezeigt. Lest die Erklärung gründlich, um zu verstehen wie das funktioniert:

Ladet das Programm auf den Arduino.

```

1  /*
2  Blink
3  Turns on an LED on for one second, then off for one second, repeatedly.
4
5  Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
6  it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN takes care
7  of use the correct LED pin whatever is the board used.
8  If you want to know what pin the on-board LED is connected to on your Arduino model, check
9  the Technical Specs of your board at https://www.arduino.cc/en/Main/Products
10
11 This example code is in the public domain.
12
13 modified 8 May 2014
14 by Scott Fitzgerald
15
16 modified 2 Sep 2016
17 by Arturo Guadalupi
18 */
19
20
21 // the setup function runs once when you press reset or power the board
22 void setup() {
23   // initialize digital pin LED_BUILTIN as an output.
24   pinMode(LED_BUILTIN, OUTPUT);
25 }
26
27 // the loop function runs over and over again forever
28 void loop() {
29   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
30   delay(1000); // wait for a second
31   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
32   delay(1000); // wait for a second
33 }
34

```

Der graue Text sind Kommentare, die erklären, was der Quelltext macht.


LED_BUILTIN ist der Name des Pins, mit dem die LED verbunden ist

1000ms ist die Latenz beim Einschalten der LED

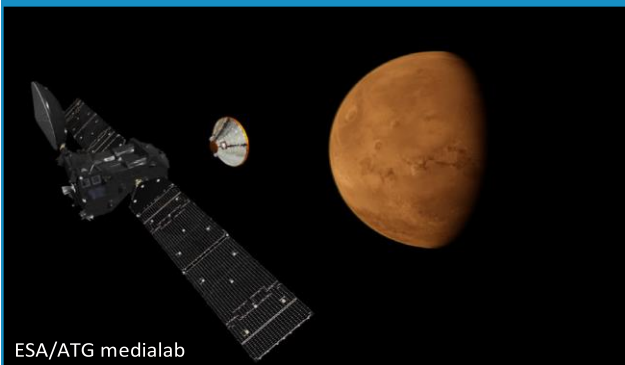
HIGH = LED ist an
LOW = LED ist aus

- a. Bevor der Quelltext auf das Arduino Uno hochgeladen werden kann, müsst ihr den Namen des Pins in dem Quelltext mit der Nummer des Pins abgleichen, an dem die grüne LED mit dem Arduino Uno verbunden ist.

Erklärt, was ihr im Quelltext verändert habt, um die grüne LED steuern zu können:

- b. Ladet den Quelltext auf das Arduino, indem ihr auf das  oben links auf dem Bildschirm klickt. Glückwunsch! Ihr habt die grüne LED angemacht!

Wusstest du schon ...?



ESA/ATG medialab

Der Mars ist so weit von der Erde entfernt, dass es ziemlich lange dauert, bis Signale von einem Raumschiff oder vom Rover zu uns zurückgekehrt sind. Diese entfernungsbedingte Latenz macht die Kommunikation mit dem Rover sowie die Reaktionen auf plötzliche Probleme zu großen Herausforderungen.

→ Aufgabe 2: Signalisiere ein S.O.S.!

Wie man eine LED mithilfe des Quelltexts zum Leuchten bringt, wisst ihr bereits. In der folgenden Aufgabe lernt ihr, wie man mit einer LED kommuniziert und wie man eine Nachricht zu einem Rover auf dem Mars schickt. Tatsächlich ist es so, dass der Rover von einem Sandsturm bedroht wird! Um Schaden am Rover zu vermeiden, müsst ihr dafür sorgen, dass der Rover ein S.O.S. signalisiert!

Wusstest du schon ...?



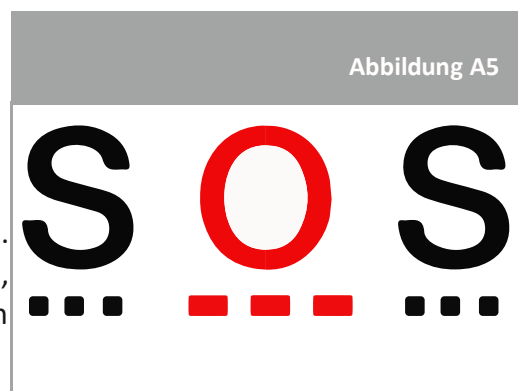
Dieses Bild eines Sandsturms auf dem Mars zeigt Wolken, die sich aus verschiedenen Gründen bilden können: Veränderungen im atmosphärischen Druck, in der Höhe und Temperatur sowie Vertikalverschiebungen, wenn z.B. der Wind über einen Berg oder Krater bläst.

Komponenten

- Der Stromkreis, der in Aufgabe 1 gebaut wurde.

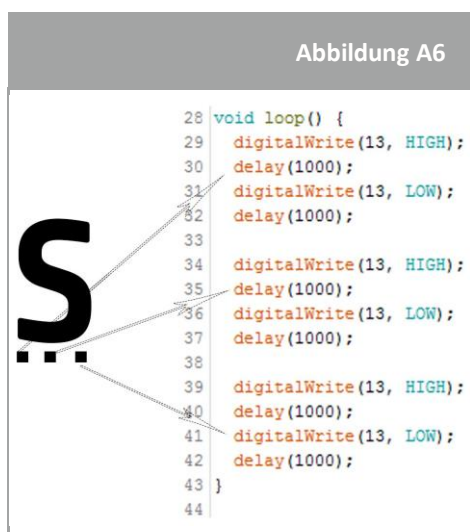
Übung

Verwendet den Morse-Code, und lasst die grüne LED aufleuchten. Wie ihr in Abbildung 5 seht, bedeutet ein Punkt ein kurzes Signal, der die LED kurz aufleuchten lässt. Ein Strich bedeutet ein längeres Signal, das die LED länger aufleuchten lässt.



↑ SOS in Morse-Code

In Abb. A6 findet ihr ein Beispiel für den Quelltext, um ein ‚S‘ zu signalisieren.



↑ Quelltext, um ein ‚S‘ in Morse-Code zu signalisieren

Versucht zu erklären, was ihr im Quelltext verändern müsst, um statt des ‚S‘ ein ‚O‘ zu signalisieren:

Verändert den Quelltext so, dass ein S.O.S. signalisiert wird. Ladet den Quelltext anschließend auf das Arduino und testet ihn!

→ Aufgabe 3: Temperaturmessung

Die Arduino Technologie erlaubt es euch, mithilfe eines auf einem Laptop geschriebenen Quelltexts eine LED zu steuern, die mit einem Arduino verbunden ist. Wenn ihr den Quelltext hochladet, schickt dieser Befehle an das Arduino und aktiviert die elektronischen Komponenten. In der ExoMars-Mission wird Technologie verwendet, die mithilfe einiger Sensoren die Umwelt des roten Planeten untersucht. In dieser Aufgabe lernt ihr wie man die Temperatur im Klassenzimmer misst und wie man die Temperatur auf dem Mars simuliert!

Komponenten

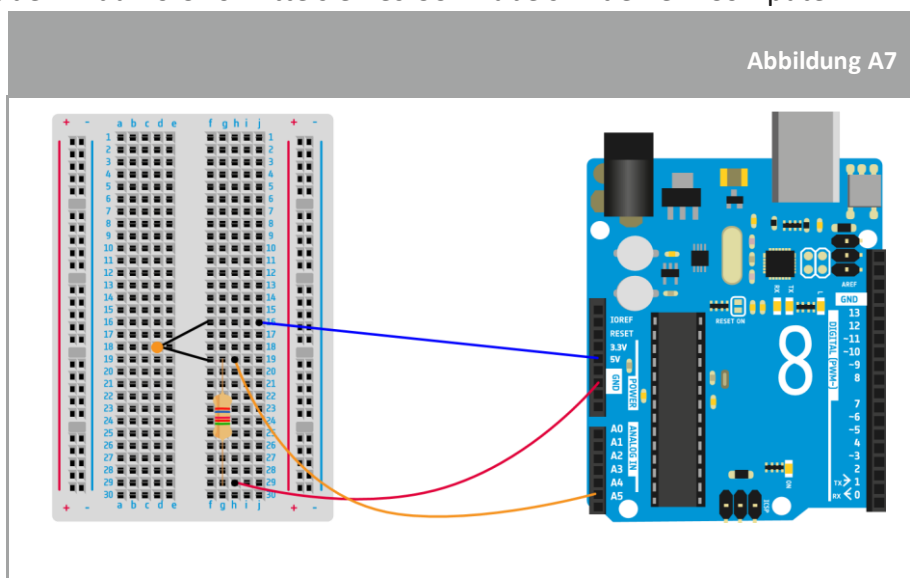
- 1 Arduino Uno
- 1 Lochrasterplatte
- 1 Temperatursensor (Thermistor)
- 1 10kΩ Widerstand (braun/schwarz/schwarz/rot/braun)
- 3 Kabel

Übung

1. Um die Temperatur im Klassenraum zu messen, müsst ihr einen Schaltkreis auf die Lochrasterplatte bauen, wie in Abb. A7.

Befolgt hierzu die untenstehenden Instruktionen:

- a. Baut den Temperatursensor auf die Lochrasterplatte wie in Abb. A7.
- b. Verbindet den 10kΩ Widerstand (braun/schwarz/schwarz/rot/braun) mit dem Temperatursensor wie in Abb. A7.
- c. Verbindet den Temperatursensor mittels eines Kabels mit dem **5V** Pin auf dem Arduino Uno.
- d. Verbindet den Widerstand mittels eines Kabels mit dem **GND** Pin des Arduino Uno.
- e. Verbindet den Temperatursensor und den Widerstand mittels eines Kabels mit dem **A5** Pin des Arduino Uno.
- f. Verbindet den Arduino Uno mittels eines USB-Kabels mit einem Computer.



↑ Arduino Schaltkreis zum Messen von Temperatur

2. Abb. A8 veranschaulicht einen lückenhaften Quelltext zur Temperaturmessung. In die drei Kästchen (rot, grün und blau) müssen drei fehlende Werte eingetragen werden, damit der Arduino die richtigen Befehle für die Temperaturmessung erhält. Finden wir nun heraus, wie man den Quelltext vervollständigen kann.

```

1
2 void setup() {
3     Serial.begin(9600);
4 }
5
6 void loop() {
7
8     float ;
9     float ;
10
11      = float(analogRead());
12      = (-40000/) + 100;
13
14     Serial.println();
15     Serial.print("Temperature in classroom: ");
16     Serial.print();
17     Serial.print("C");
18     delay(1000);
19
20 }

```

[↑ Arduino Quelltext zum Messen von Temperatur](#)

- a. Welche Einheit wird in Europa zur Temperaturmessung verwendet?


Der mit dem Arduino Uno verbundene Temperatursensor liest die Temperatur zunächst in einer anderen Einheit ab, nämlich Volt. Um diesen Wert von Volt in Grad Celsius umzurechnen, könnt ihr die Formel benutzen, die ihr in Abb. A8 in Zeile 12 findet. Schließlich könnt ihr euch die Temperatur mittels der Serial.print-Funktion in eurem Quelltext auf dem Bildschirm anzeigen lassen.


- b. Definiert die Variable, die die Temperatur in Volt beinhaltet:

Definiert die Variable, die die Temperatur in Grad Celsius beinhaltet:

Welcher Pin auf dem Arduino ist der, der mit dem Thermistor verbunden ist?

- c. Ihr seid nun bereit, den Quelltext auf einem Laptop zu schreiben. Öffnet die Arduino Software und schreibt den Quelltext aus Abb. A8 ab. Verwendet hierzu die Variablen aus den farbigen Kästchen. Achtet dabei auf die korrekte Zuordnung der Farben!

Ladet den Quelltext auf das Arduino hoch, indem ihr  klickt.

- d. Um die Temperaturmessung anzuzeigen, müsst ihr ein spezielles Fenster in der Arduino Software öffnen, welches Serial Monitor genannt wird. Haltet auf der Menüleiste Ausschau nach diesem  Symbol und klickt darauf.

Diskussion

1. Diskutiert die Relevanz der Temperaturmessung mit euren Mitschülerinnen und Mitschülern. Wie könnt ihr verifizieren, dass der Temperatursensor eine genaue Messung der Temperatur im Klassenraum vornimmt?

2. Welche Zeile des Quelltexts in Abb. A8 könnt ihr verändern, um den Sensor zu kalibrieren?

3. Um mehr Übung im Umgang mit dem Arduino zu erlangen, kalibriert den Temperatursensor bitte so, als ob er sich auf dem Mars befinden würde. Beachtet, dass die Durchschnittstemperatur auf dem Mars bei -60°C liegt.

a. Durchschnittstemperatur auf der Erde = _____

b. Schreibt die neue Kalibrierungsformel des Temperatursensors auf:

→ Aufgabe 4: Druckmessung

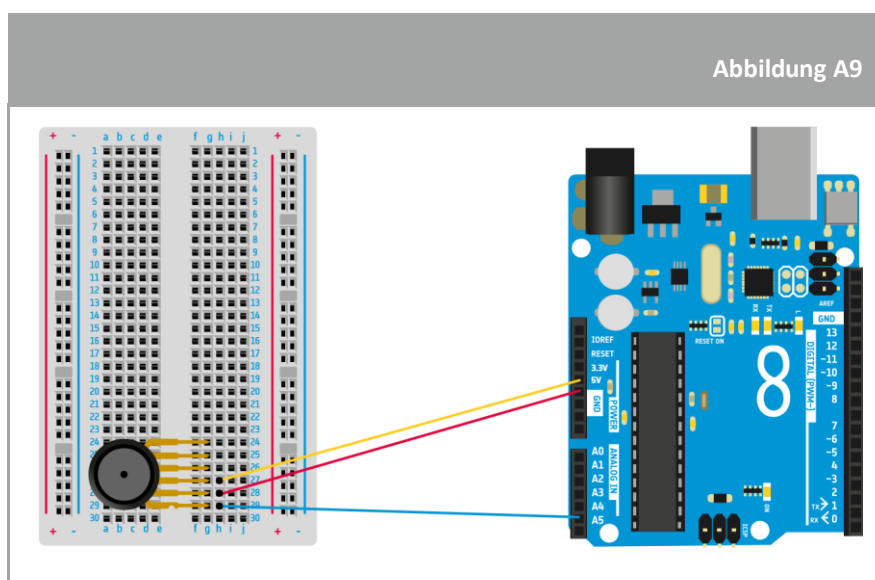
In dieser Aufgabe lernt ihr, wie man mithilfe eines Arduino und eines Drucksensors den atmosphärischen Druck misst.

Komponenten

- 1 Arduino Uno
- 1 Lochrasterkarte
- 1 Drucksensor
- 3 Kabel

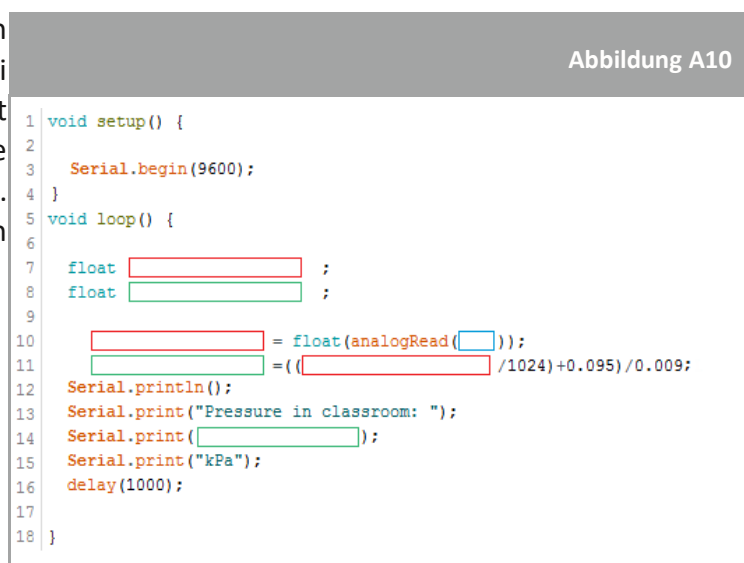
Übung

1. Um den Druck im Klassenraum zu messen, müsst ihr zunächst einen Schaltkreis auf der Lochrasterkarte aufbauen (Abb. A9). Befolgt die Anleitung sorgfältig (Hinweis: Jeder Metallstift an dem Drucksensor macht etwas anderes. Deshalb solltet ihr sicherstellen, dass ihr ihn so wie in Abb. A9 anschließt):



↑ Arduino Schaltkreis zum Messen des Drucks

2. Abb. A10 zeigt einen lückenhaften Quelltext zur Druckmessung. Die drei farbigen Kästchen müssen ausgefüllt werden, damit das Arduino Uno eine korrekte Druckmessung vornehmen kann. Finden wir nun heraus, wie wir den Quelltext ergänzen können.



[↑ Arduino Quelltext zum Messen von Druck](#)

- a. Was ist die physikalische Einheit, die im „International System of Units“ für Druck verwendet wird?

- b. Definiert die Variable für Druck in Volt:

Definiert die Variable für Druck in Kilo Pascal:

Mit welchem Pin des Arduino ist der Drucksensor verbunden?

Der mit dem Arduino Uno verbundene Drucksensor liest den Druck zunächst in einer anderen Einheit ab, nämlich in Volt, einer Einheit für elektrische Spannung. Um diesen Wert in Pascal umzuschreiben, verwenden wir die folgende Formel, die auf dem Datenblatt des Sensors zu finden ist:

$$V_{\text{out}} = V_S (P \times 0.009 - 0.095).$$

Stellt schließlich die Messung auf dem Bildschirm dar, mithilfe der Serial.print-Funktion.

- c. Verwendet nun die Formel $V_{\text{out}} = V_S (P \times 0.009 - 0.095)$; errechnet die Kalibrierungsformel, um Zeile 12 aus Abb. A10 zu vollenden. Weil die Messwertrückmeldung in 1024 Schritte aufgeteilt ist, müsst ihr $V_S = 1024$ setzen.

- d. Nun seid ihr bereit den Quelltext auf einem Laptop zu schreiben. Öffnet die Arduino Software und schreibt den Quelltext aus Abb. 10 zunächst ab und vervollständigt dann die Lücken. Achtet dabei auf die Farben!

Ladet den Quelltext auf das Arduino hoch, indem ihr  klickt.

- e. Um die Druckmessungen darzustellen, muss man ein spezifisches Fenster in der Arduino Software öffnen, den Serial Monitor. Haltet Ausschau nach diesem Symbol in der Taskleiste und klickt darauf:



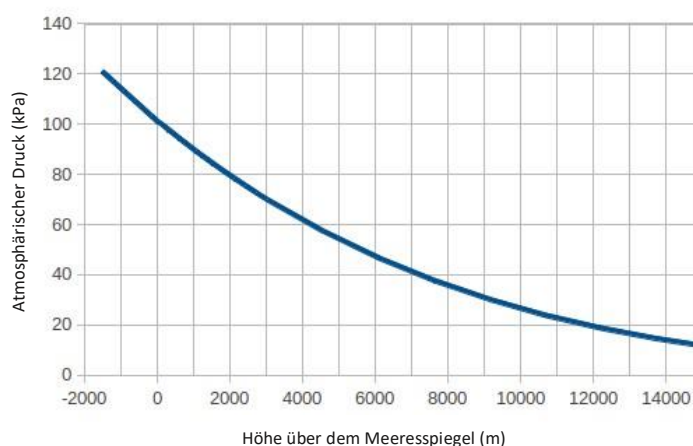
Diskussion

Diskutiert mit euren Mitschülerinnen und Mitschülern die Relevanz der Druckmessung. Wie könnt ihr verifizieren, dass der Sensor den Druck korrekt misst?

Welche Zeile des Quelltexts in Abb. A10 müsst ihr anpassen, um den Sensor zu kalibrieren?

→ Aufgabe 5: Höhenmessung

Der Luftdruck verändert sich mit der Höhe. So ist z.B. der Luftdruck auf See höher als der in den Bergen. Aus dem Graphen ist ersichtlich, wie der Luftdruck variiert. Mit ein wenig Mathematik und Verständnis der zugrundeliegenden Gasgesetze können wir die Beziehung zwischen Druck und Höhe exakt ausrechnen. Doch darum werden wir uns jetzt nicht kümmern. Stattdessen werden wir in der vorliegenden Aufgabe unsere Druckmessungen dazu verwenden, die Höhe zu bestimmen, und zwar mithilfe einer vereinfachten Formel zur Höhenmessung, der Barometrischen Formel.



Komponenten

- Arduino Schaltkreis aus Aufgabe 4
- Quelltext zur Druckmessung aus Aufgabe 4

Übung

1. Öffnet den Quelltext zur Druckmessung, den ihr in Aufgabe 4 in der Arduino Software geschrieben habt. Fügt diesen lückenhaften Quelltext in den Void Loop ein:

Abbildung A11

```

1 float [ ];
2 float [ ];
3
4 [ ] = pow([ ]/101.325, 0.1903);
5 [ ] = (1 - [ ]) * 44300 + 100;
6
7 Serial.println();
8 Serial.print("Altitude in classroom: ");
9 Serial.print([ ]);
10 Serial.print("meters");
11 delay(1000);

```

↑ [Arduino Quelltext zum Messen von Höhe](#)

2. Um die Höhe zu bestimmen, müsst ihr die Messungen des Drucksensors aus Zeile 4 in Abb. A11 verwenden. Wie ihr in den vorangegangenen Aufgaben bereits festgestellt habt, stellt das Arduino die Messungen zunächst in Volt dar. Um diese Werte in Meter umzurechnen, müsst ihr die Formel aus Zeile 5 in Abb. A11 verwenden. Nun können die Höhendaten mithilfe der Serial.print-Funktion im Quelltext dargestellt werden. Sollte eine Kalibrierung von Nöten sein, kann diese in Zeile 5 des Quelltexts vorgenommen werden.

3. a. Definiert eine Variable, die die Höhe in Volt beinhaltet:

Definiert eine Variable, die die Höhe in Metern beinhaltet:

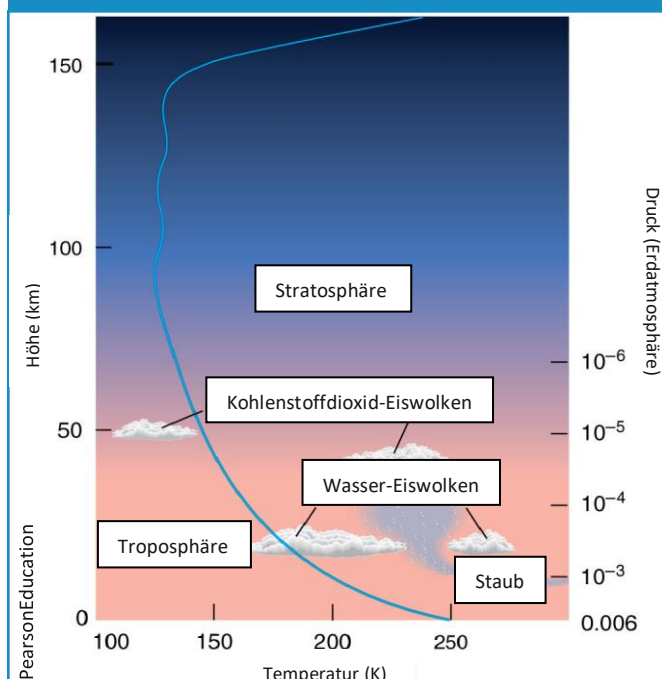
Wie lautet die Bezeichnung für Druck in Pascal (s. Aufgabe 4)?

- b. Vervollständigt den Quelltext mit den Variablen, die ihr für die farbigen Kästchen festgelegt habt. Achtet dabei auf die Farben! Ladet anschließend den Quelltext hoch.
- c. Um die Höhendaten anzeigen zu können, müsst ihr ein spezifisches Fenster in der Arduino Software öffnen, den sogenannten Serial Monitor.
4. Kombiniert den Temperatur- und Höhequelltext in einem einzigen Sketch und ladet diesen auf euer Arduino Uno.

Sammelt Temperatur- und Höhedaten vom Boden des Klassenraums bis hin zur Decke und zeichnet anschließend einen Graphen, der die Ergebnisse darstellt.

Könnt ihr die Form des Graphen erklären?

Wusstest du schon ...?



Die Marsatmosphäre kommt einer extrem dünnen Gasschicht gleich, die hauptsächlich aus Kohlenstoffdioxid besteht und sich von der Planetenoberfläche bis zum Rand des Weltraums erstreckt. Die Sonne heizt die Oberfläche des Mars auf und ein Teil dieser Hitze erwärmt das Gas nahe der Marsoberfläche. Das erhitzte Gas breitet sich dann aus bzw. konvergiert nach oben durch die Atmosphäre. Daher ist die Temperatur des Gases in der Nähe der Oberfläche am höchsten und wird immer niedriger, je höher man aufsteigt.

