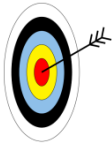


## Station 0 – Einstieg

### Herzlich Willkommen zum Workshop rund um das Thema Licht!

Bevor ihr euch auf eure Projekte stürzt, ist es wichtig, dass ihr euer Material kennt. Hier bekommt ihr einen Überblick über den Arduino-Mikrocontroller, die Bauteile und die Programmierumgebung!

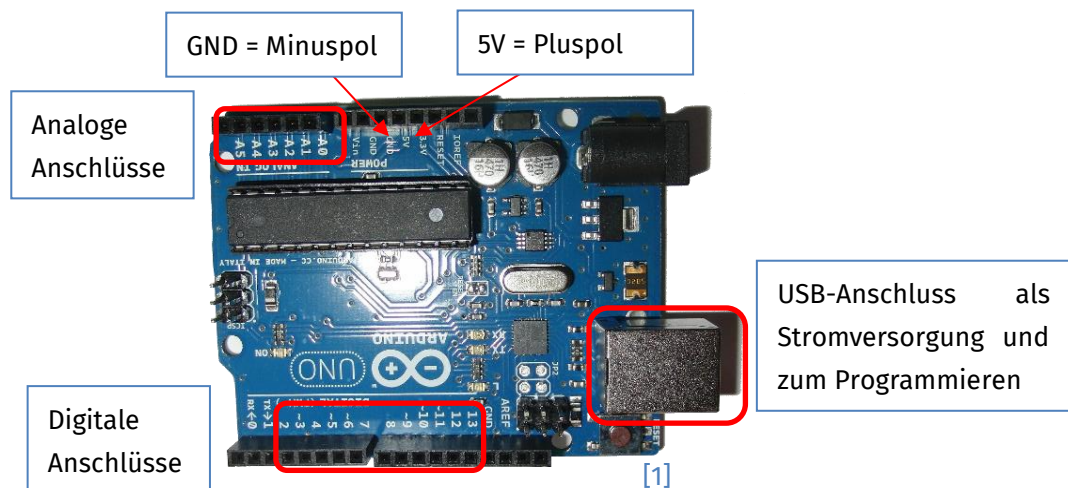


In diesem Arbeitsblatt werdet ihr...

- X eine erste Schaltung aufbauen,
- X eine LED zum Leuchten und Blinken bringen
- X und sie mit einem Taster ein- und ausschalten.

### Der Arduino

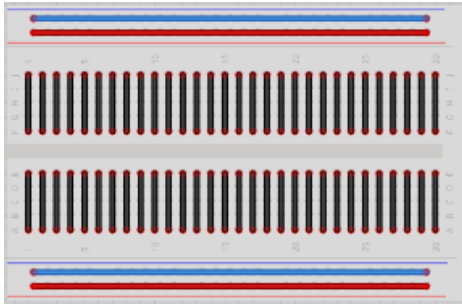
Als Erstes solltet ihr euch den Arduino anschauen, der heute im Mittelpunkt jeder Schaltung steht. Wenn ihr genauer hinsieht, dann gibt es auf eurem Arduino verschiedene Anschlüsse. Sucht auf eurem Arduino die gekennzeichneten Anschlüsse aus der Abbildung:



Die digitalen Anschlüssen 0 und 1 sind für besondere Aufgaben vorgesehen und sollten von euch nicht benutzt werden!

## Station 0 – Einstieg

### Das Steckbrett



[2]

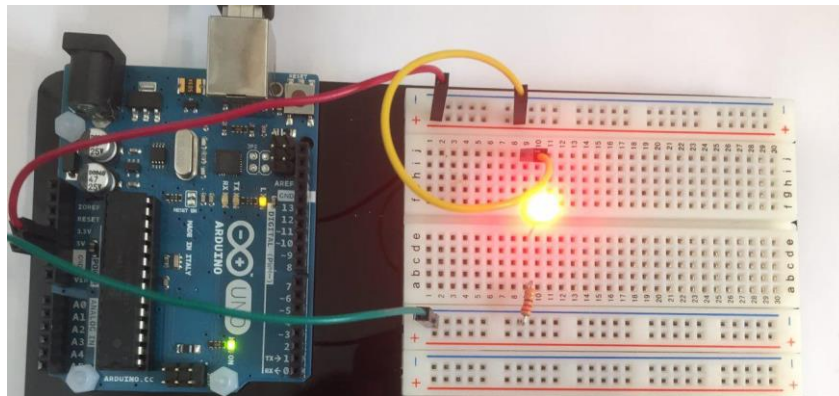
Auf dem Steckbrett werdet ihr gleich all eure Schaltungen stecken. Das Steckbrett ist, wie in der Abbildung links zu sehen, intern verbunden. Das bedeutet, dass der blaue und der rote Streifen wie ein langes Kabel funktionieren. Genauso sieht es bei den einzelnen Spalten aus (hier in schwarz).



Wenn man also ein Bauteil mit beiden Beinchen in eine verbundene Reihe steckt, dann ist das so, als ob es gar nicht da wäre...

### Eine LED zum Leuchten bringen

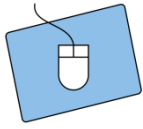
Jetzt könnt ihr auch schon mit eurer ersten Schaltung loslegen und so eine **LED** (Lampe) zum Leuchten bringen. Dazu braucht ihr nur die kleine rote LED und einen Widerstand aus dem Station-0-Kästchen. Die Widerstände verhindern, dass die LEDs kaputtgehen. Je nach Stärke haben sie unterschiedliche Farbringe. Ihr braucht einen mit **rot-rot-braun-goldenen** Ringen.



[3]

Damit die LED leuchtet, muss man sie richtig herum in das Steckbrett stecken. Das lange Beinchen muss später mit dem Plus-, das kurze Beinchen mit dem Minuspol verbunden werden.

## Station 0 – Einstieg



1. Verbindet den 5V-Anschluss am Arduino mit der roten Leiste auf dem Steckbrett.
2. Verbindet den GND-Anschluss am Arduino mit der blauen Leiste auf dem Steckbrett.
3. Fügt nun ein gelbes Kabel, einen Widerstand und eine LED, wie auf dem Bild oben zu sehen, in die Schaltung ein.
4. Verbindet den Arduino über das USB-Kabel mit dem Laptop.

*Sehr gut, ihr habt die LED zum Leuchten gebracht. Jetzt könnt ihr anfangen zu programmieren.*

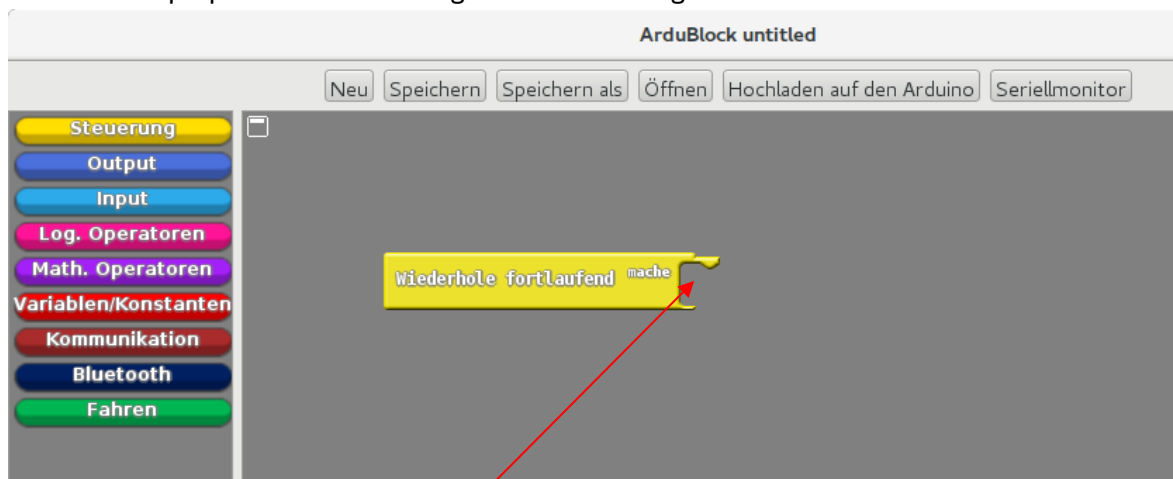
### Eine LED zum Blinken bringen

Ihr wisst nun, wie man die LED zum Leuchten bringt. Doch wie schafft man es, dass sie blinkt? Hier kommt der Arduino ins Spiel. Hierzu müsst ihr aber zunächst ein klein wenig umbauen.



Steckt eure Schaltung so um, dass das gelbe Kabel vom Plus-Beinchen der LED zu einem der digitalen Eingänge (nicht 0 oder 1) führt.

Auf eurem Laptop ist bereits das Programm ArduBlock geöffnet:



[4]



[5]

Mit Hilfe von ArduBlock soll zunächst die LED zum Leuchten gebracht, also eingeschaltet, werden.

Dazu braucht ihr den Befehl **digitalWrite** aus **Output**. Zieht ihn in den mache-Bereich des

## Station 0 – Einstieg

**Wiederhole-fortlaufend-Befehls.** Dann müsst ihr anpassen, an welchem Ausgang (Pin) eure LED angeschlossen ist. Der Wert **HIGH** bedeutet, dass die LED eingeschaltet wird.



1. Schreibt in ArduBlock ein Programm, das eure LED zum Leuchten bringt.
2. Testet euer Programm, indem ihr auf „Hochladen auf den Arduino“ klickt.

Euer Programm sollte jetzt etwa so aussehen:



[6]

### Und was ist mit dem Blinken?

Es wurde die LED angeschaltet. Zum Blinken müsst ihr sie jedoch auch wieder ausschalten. Damit ihr das Umschalten auch sehen könnt, müsst ihr noch eine Pause einbauen. Den Befehl **Warte Millisekunden** findet ihr unter **Steuerung**.



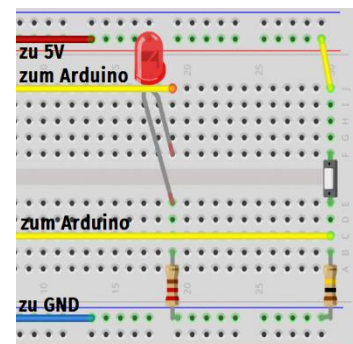
1. Erweitert euer Programm so, dass die LED auch ausgeschaltet wird, indem ihr Wert statt auf **HIGH** auf **LOW** gesetzt wird.
2. Fügt nach **jedem** Schaltvorgang eine Pause von einer Sekunde (1000 ms) ein.
3. Testet euer Programm.
4. Speichert euer Programm unter einem sinnvollen Namen.

### Die LED steuern

Sehr gut, ihr habt die LED nun zum Blinken gebracht. Aber eigentlich wäre es ja nett, wenn man sie auch mit der Hand einschalten könnte. Hierzu braucht ihr einen Taster, also einen kleinen Lichtschalter:



Auch der Taster [7] muss mit einem Widerstand hintereinandergeschaltet werden. Dazu braucht ihr einen Widerstand mit **braun-schwarz-gelb-goldenen** Ringen.



[8]

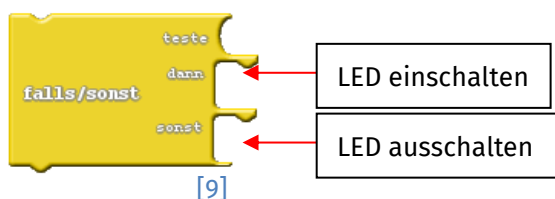
## Station 0 – Einstieg



**Erweitert** eure Schaltung wie in Abbildung 7 zu sehen ist: „Zum Arduino“ bedeutet hier, dass dieses Kabel in einen digitalen Eingang am Arduino gesteckt werden muss. Denkt daran, dass ihr die digitalen Eingänge 0 und 1 nicht nutzen sollt.

*Sehr gut! Dann steht der nächsten Programmerrunde ja nichts mehr im Weg.*

**Falls** der Taster gedrückt ist, **dann** soll die LED angeschaltet werden, **sonst** soll sie ausgeschaltet werden. Unter **Steuerung** findet ihr einen Block, der euch genau dieses Verhalten liefert:



Im **teste**-Teil wird die Voraussetzung für den dann-Block angegeben, also: „**falls** der Taster gedrückt ist“. Um das zu überprüfen, braucht ihr drei weitere Blöcke:

Der **Taster** ist eine Eingabe, seine Befehle findet ihr also unter **Input**. Ihr müsst ihn also **(digital) lesen** und braucht den Block: **digitalRead**.

Nachdem ihr nun auslesen könnt, was am Pin anliegt, müsst ihr es im nächsten Schritt nun vergleichen. Dazu braucht ihr den **==-Operator** aus **Log. Operatoren**.

**Achtung:** Es gibt zwei Varianten. Ihr braucht den mit den runden Lücken.

Vergleichen schön und gut, aber mit was?

Ihr wollt wissen, ob der Taster **an** ist, also **HIGH**. Das findet ihr unter **Variablen/Konstanten**.





1. Sucht die oben genannten Blöcke aus den entsprechenden Menüs.
2. Fügt sie zu einem Programm zusammen, das die LED anschaltet, wenn der Taster gedrückt ist, und ansonsten ausschaltet.
3. Testet euer Programm.

*Klasse! Damit seid ihr nun bestens gerüstet, um euch an die großen Projekte heranzuwagen. Meldet euch, damit ihr das Material dazu erhaltet.*



#### Quellenverzeichnis:

Abb. 1, 3, 7, , , ,  – Quelle: InfoSphere-Team, CC BY-SA 4.0 Attribution-ShareAlike 4.0

## Station 0 – Einstieg

International (<https://creativecommons.org/licenses/by-sa/4.0/>).

**Abb. 2, 8** – Quelle: Screenshots der Fritzing-Software (<http://fritzing.org>), CC-BY-SA (<https://creativecommons.org/licenses/by-sa/3.0/>), erstellt am 31.01.2023.

**Abb. 4-6, 9** – Quelle: Screenshot der Software Ardublock (<https://arduino-basics.com/ardublock/>), GNU General Public License (<https://github.com/taweili/ardublock/blob/master/LICENSE.txt>), erstellt am 31.01.2023.