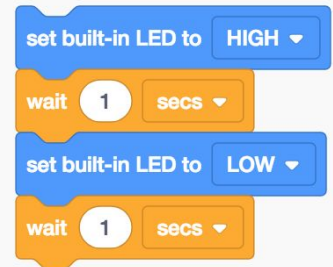


Standard Code:



Info

Beim Arduino handelt es sich um eine Plattform, die es jedem Menschen ermöglichen soll, seine eigenen technischen Ideen in die Tat umzusetzen.

Tipp

Teste dein Programm zunächst im Simulator und anschließend auf deinem Arduino UNO Gerät.

Hinweis

Falls du noch Fragen hast oder etwas nicht ganz verstanden hast, findest du auf arduino-tutorial.de weitere Informationen.

Aufgabe

Wir lernen wie ein Arduino aufgebaut ist, wie er funktioniert und wie wir ihn in unserem Tool simulieren.

Schritt 1:

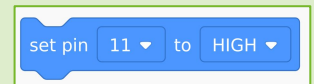
- Ziehe den Arduino UNO R3 auf deine Arbeitsfläche und klicke [Code](#)
- Schaue dir den Code an und überlege was passieren wird. Starte nun die Simulation

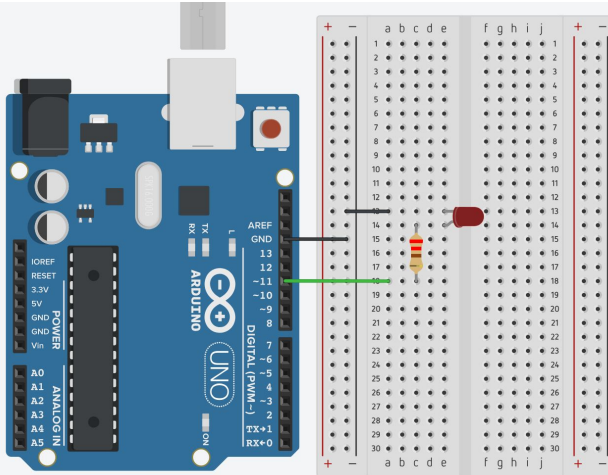
Schritt 2

- Versuche nun den Code so zu verändern, dass die LED schneller blinkt.

Nächste Aufgabe

- Steuere eine externe LED, indem du einen Schaltkreis mit einem 220 Ohm Widerstand und einer LED mit den Pins GPIO 11 und GND verbindest.





Tipp

Wenn du auf den Widerstand klickst, lässt sich die Größe des Widerstands ändern.

Resistor

Name

Resistance Ω

Info

Was passiert wenn man den Block “set pin 11 to HIGH” ausführt? Pin 11 wird auf digital “HIGH” gesetzt.

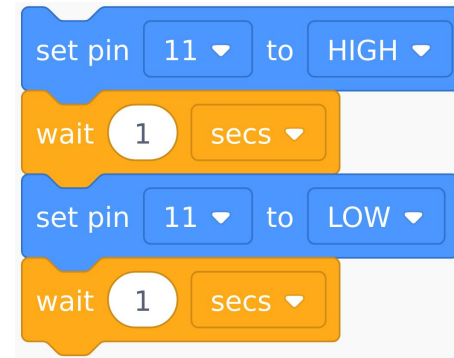
“HIGH” bedeutet dass der Arduino eine 5V Spannung an PIN 11 anlegt. Das führt dazu, dass die LED anfängt zu leuchten.

“LOW” bedeutet, dass der Pin mit GND verbunden wird und dadurch keine Spannung an der LED anliegt.

Aufgabe

- Versuche nun den Arduino zu nutzen, um eine externe LED zum Blinken zu bringen. Nutze hierzu eine Breadboard wie im Bild auf der linken Seite.
- Tipp: Der Widerstand muss **220 Ohm** groß sein.

Wenn du alles richtig gemacht hast, sollten deine Programmierblöcke so aussehen:



Nächste Aufgabe

Verwende nun für das Ein-/Ausschalten der LED einen Taster (“Button”) und verbinde diesen mit Pin 8, GND und 5V. Die LED sollte nur blinken, wenn die Taste gedrückt ist.

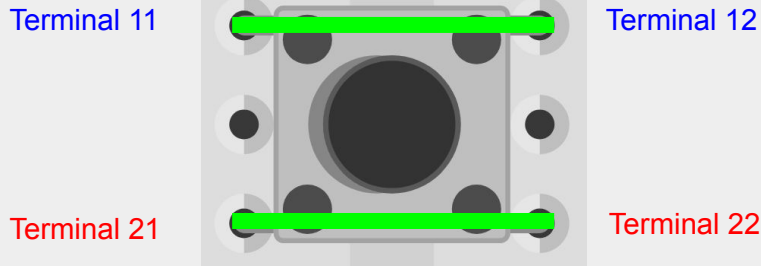
Tipp

Für die nächste Aufgabe:

- Benutze einen 1KΩ Widerstand zwischen GND <-> Taster schalten, um ein “Low” Signal zu erhalten.
- Benutze den “read digital pin” Block um zu lesen ob PIN 8 auf “HIGH” oder “LOW” gesetzt ist.
- Mehr Information wie man einen Taster benutzt, findest du auf der nächsten Seite

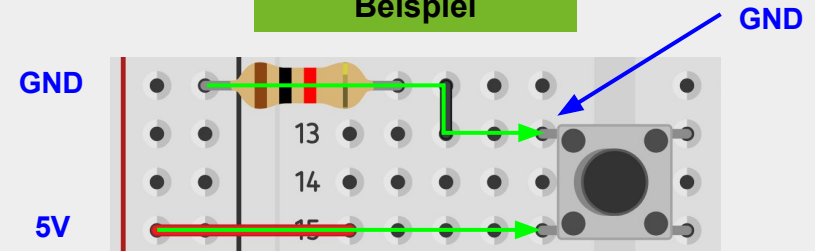
Wie funktioniert ein Taster?

Nicht Gedrückt



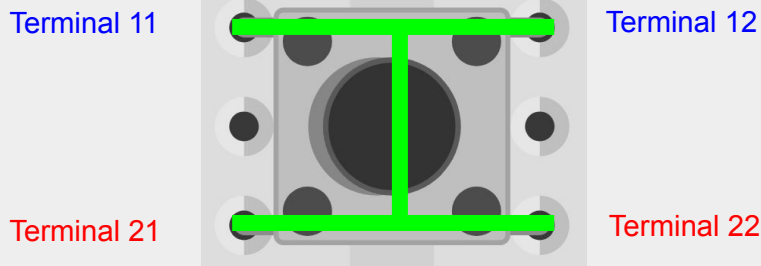
Es gibt zwei Seiten, **Terminal 1X** und **Terminal 2X**.
Terminal 11 und **12** sind immer verbunden genauso wie
Terminal 21 und **22**.

Beispiel



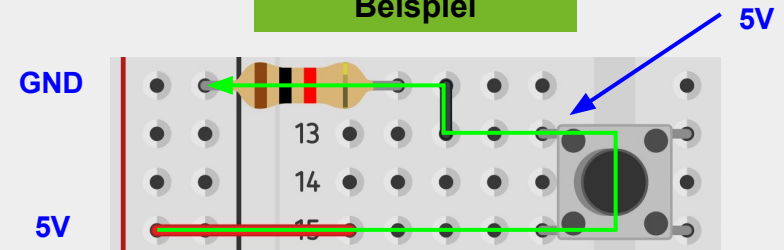
In diesem Beispiel ist **Terminal 11** auf GND (LOW) gesetzt,
 wenn der Taster nicht gedrückt ist.

Gedrückt

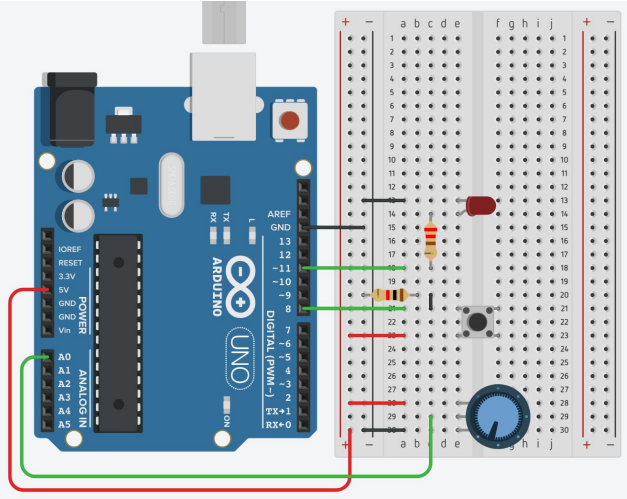


Wenn man den Taster drückt, verbinden sich **Terminal 1X**
 und **Terminal 2X** miteinander, also sind alle 4 Pins
 miteinander verbunden.

Beispiel



In diesem Beispiel ist **Terminal 11** auf 5V (HIGH) gesetzt, weil
Terminal 21 auf 5V liegt und beide miteinander verbunden sind.
 So kann man die Spannung auf **Terminal 11** ändern.



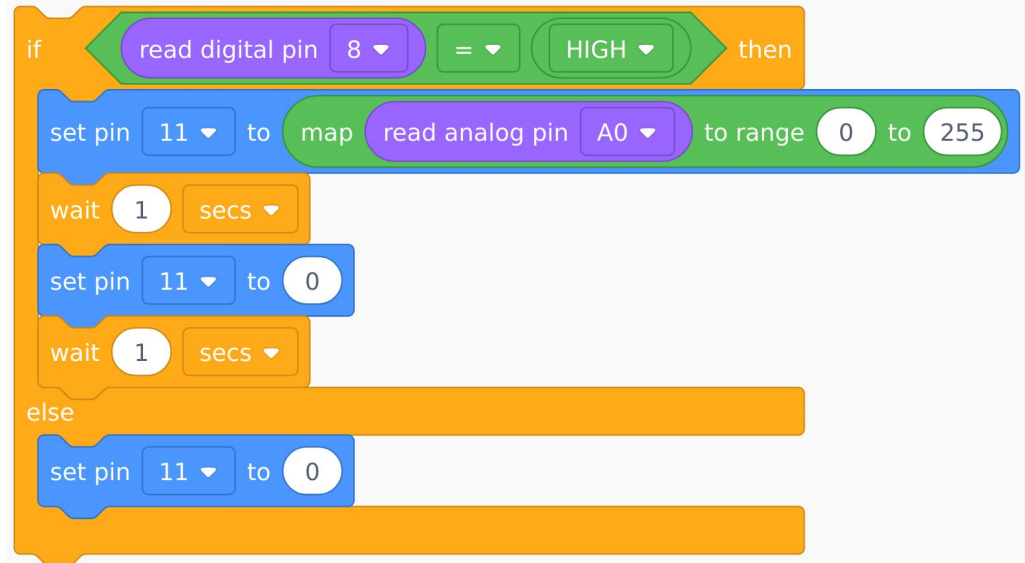
Info

PIN 11 ist besonders und kann nicht nur einen Wert von 5V oder GND ausgeben, sondern auch einen Wert zwischen GND und 5V. Man setzt ihn, indem man einen Wert von 0 bis 255 an den pin setzt (mit "set pin _ to" Block). 0 entspricht dem GND (Minus-Pol), 255 entspricht 5V, Zwischenwerte bedeuten eine Spannung zwischen 0 und 5V. Damit kann man die Helligkeit der LED regulieren.

Aufgabe

Dein Schaltkreis sollte jetzt wie im Bild aussehen. Wenn nicht, baue ihn nach.

Wenn du alles richtig gemacht hast, sollten deine Programmierblöcke so aussehen:

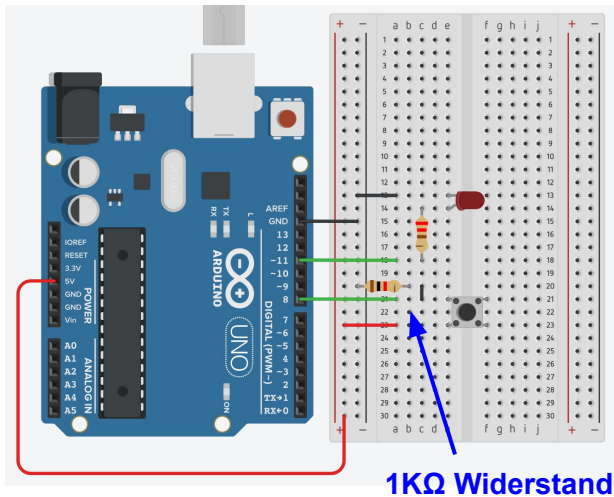


Info

Was macht "map _ to range 0 to 255"? Weil "set pin" einen Wert zwischen 0 und 255 braucht und "read analog pin" einen Wert zwischen 0 und 1023 hat, muss dieser Wert zwischen 0 und 255 angepasst werden. Das macht genau das "map" Block.

Zusatzaufgabe

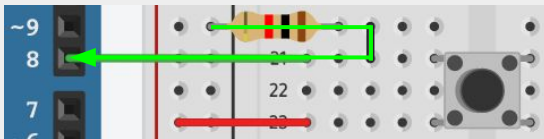
Wenn du schon früher fertig bist, kannst du versuchen eine Ampel zu bauen. Man kann 3 LEDs (Rot, Gelb, Grün) mit Pins 0 - 2 verbinden und eine Ampel bauen, damit die Farbe zwischen Rot, Gelb und Grün wechselt. Benutze noch einmal 220Ω Widerstände.



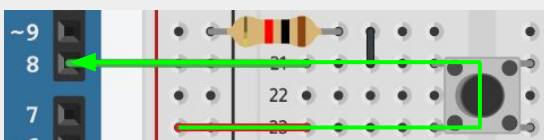
Info

Was passiert, wenn man den Taster drückt?

Nicht gedrückt: Pin 8 ist mit GND(LOW) verbunden.



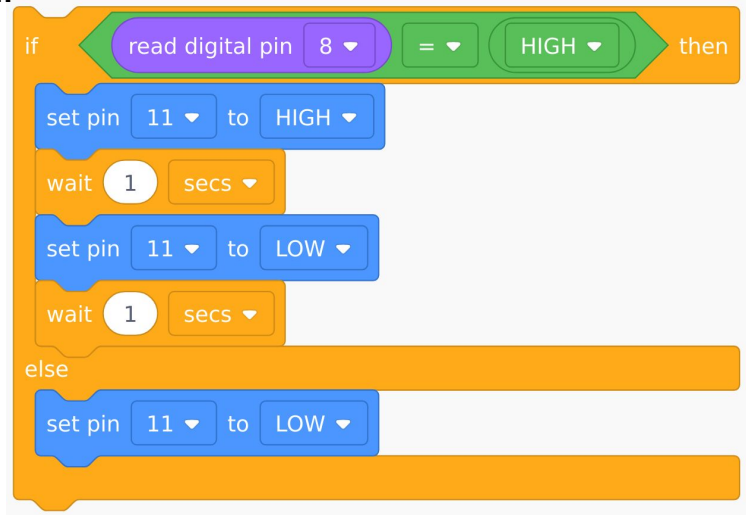
Gedrückt: Pin 8 ist mit 5V(HIGH) verbunden.



Aufgabe

Dein Schaltkreis sollte jetzt so wie im Bild aussehen. Wenn nicht, baue ihn nach.

Wenn du alles richtig gemacht hast, sollten deine Programmierblöcke so aussehen:



Nächste Aufgabe

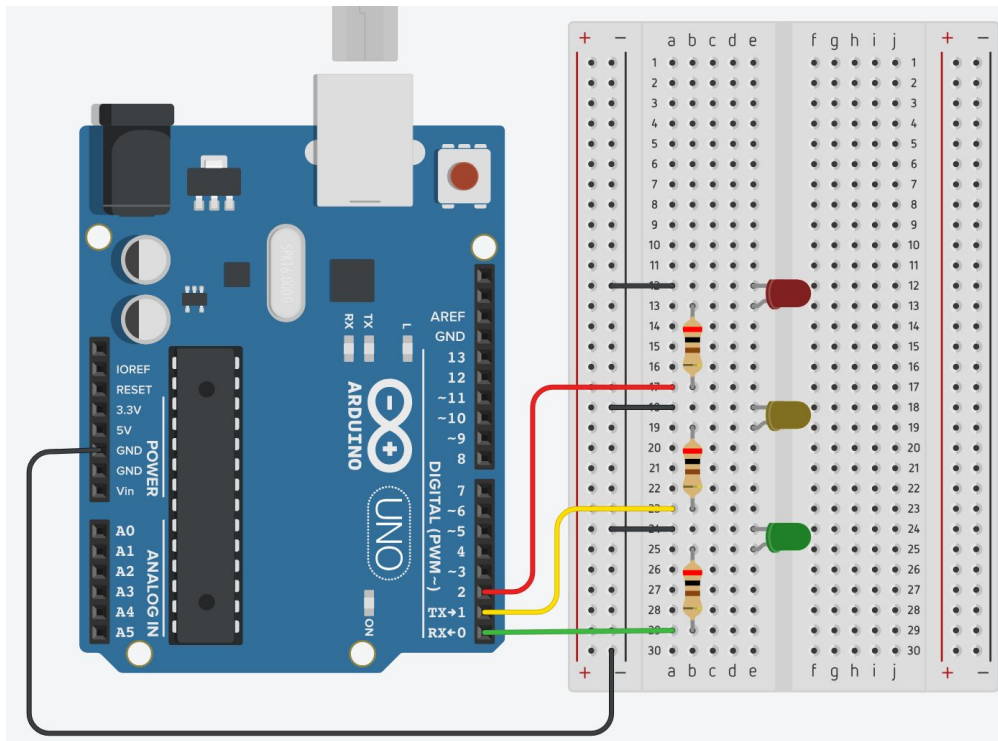
Helligkeit der LED mit einem Potentiometer regulieren: Baue einen Potentiometer ein und verbinde es mit Pin A0, GND und 5V. Benutze den Wert von dem Potentiometer (A0), um die Helligkeit der LED zu steuern.

Tipp

Für die nächste Aufgabe:

- Benutze den "read analog pin" Block um den Wert vom Potentiometer(A0) zu lesen.
- Benutze den "set pin 11 to _" Block, um die Helligkeit zu setzen.
- "read analog pin" hat einen Wert von 0 - 1023 und "set pin _ to" erwartet einen Wert von 0 - 255. Setze den "map _ to range" Block ein, um den Wert von "read analog pin" auf 0 - 255 zu setzen.

Schematischer Aufbau der Ampelschaltung



Code

```

set pin 2 ▼ to HIGH ▼
wait 1 secs ▼
set pin 2 ▼ to LOW ▼
set pin 1 ▼ to HIGH ▼
wait 1 secs ▼
set pin 1 ▼ to LOW ▼
set pin 0 ▼ to HIGH ▼
wait 1 secs ▼
set pin 0 ▼ to LOW ▼
wait 1 secs ▼

```