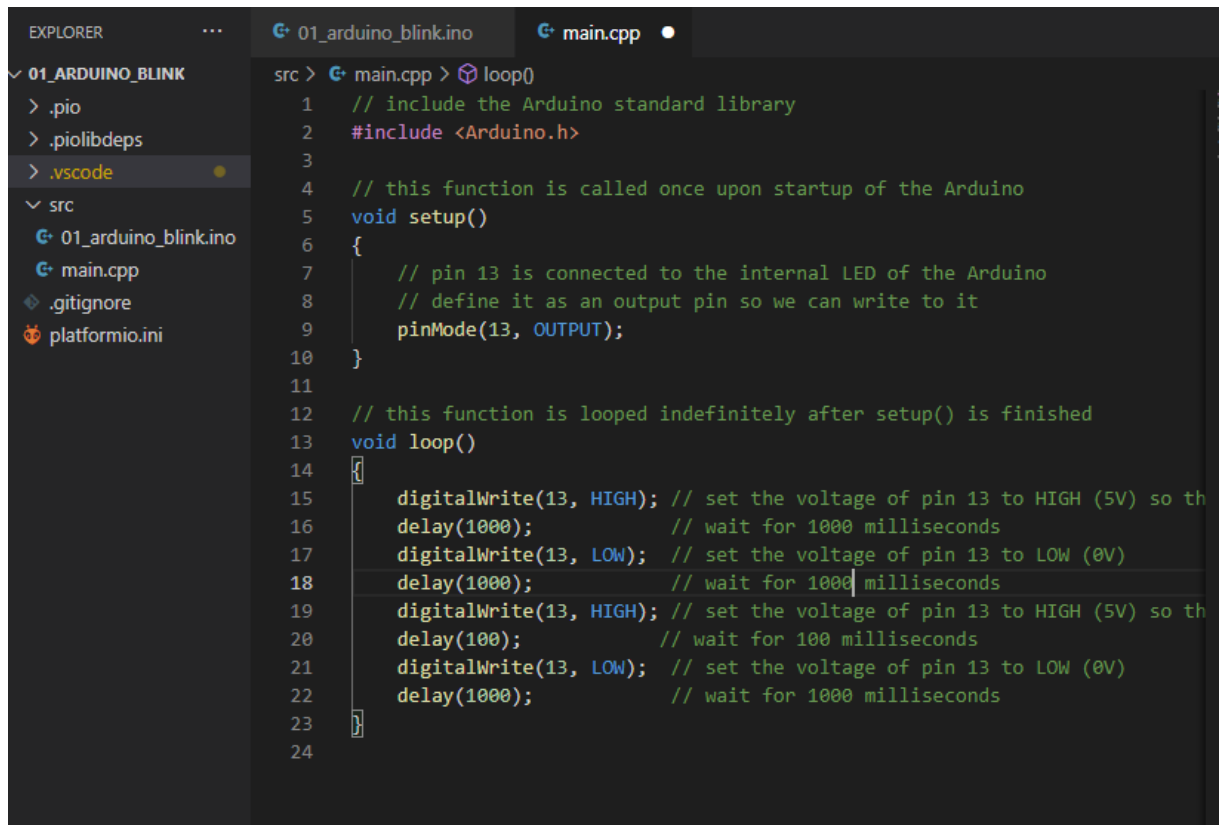


Übung 4

1. Die passenden Videos zu den Aufgaben 1.2/1.3/1.4 sind im Anhang.

1.1. Code:

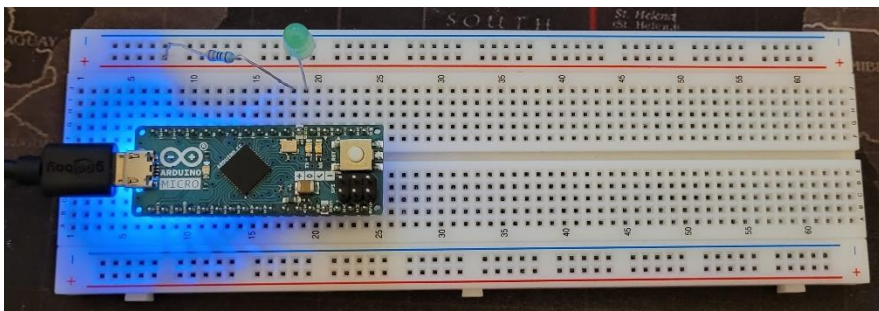


```
EXPLORER  ...  01_arduino_blink.ino  main.cpp

01_ARDUINO_BLINK
> .pio
> .pio/libdeps
> .vscode
src
  01_arduino_blink.ino
  main.cpp
  .gitignore
  platformio.ini

src > main.cpp > loop()
1  // include the Arduino standard library
2  #include <Arduino.h>
3
4  // this function is called once upon startup of the Arduino
5  void setup()
6  {
7      // pin 13 is connected to the internal LED of the Arduino
8      // define it as an output pin so we can write to it
9      pinMode(13, OUTPUT);
10 }
11
12 // this function is looped indefinitely after setup() is finished
13 void loop()
14 {
15     digitalWrite(13, HIGH); // set the voltage of pin 13 to HIGH (5V) so th
16     delay(1000);           // wait for 1000 milliseconds
17     digitalWrite(13, LOW);  // set the voltage of pin 13 to LOW (0V)
18     delay(1000);           // wait for 1000 milliseconds
19     digitalWrite(13, HIGH); // set the voltage of pin 13 to HIGH (5V) so th
20     delay(100);            // wait for 100 milliseconds
21     digitalWrite(13, LOW);  // set the voltage of pin 13 to LOW (0V)
22     delay(1000);           // wait for 1000 milliseconds
23 }
24
```

1.2. Breadboard-Aufbau:



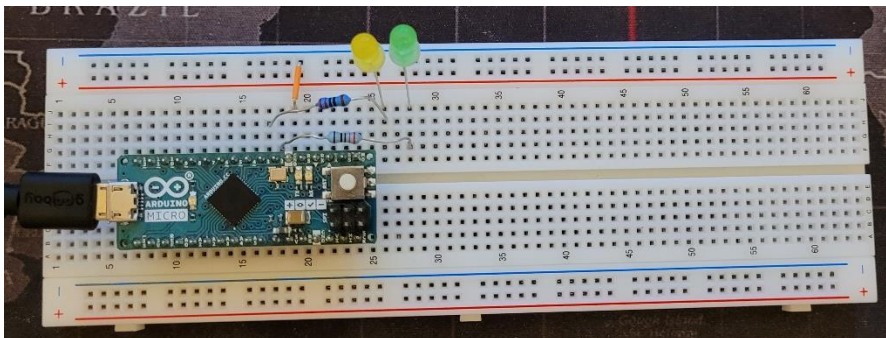
Code:

```
EXPLORER  ...  01_arduino_blink.ino  main.cpp X

01_ARDUINO_BLINK
├── .pio
├── .pio\libdeps
├── .vscode
└── src
    ├── 01_arduino_blink.ino
    ├── main.cpp
    ├── .gitignore
    └── platformio.ini

src > 01_arduino_blink.ino > loop()
1  // include the Arduino standard library
2  #include <Arduino.h>
3
4  // this function is called once upon startup of the Arduino
5  void setup()
6  {
7      // pin 2 is connected to the external led with pre resistor
8      // define it as an output pin so we can write to it
9      pinMode(2, OUTPUT);
10 }
11
12 // this function is looped indefinitely after setup() is finished
13 void loop()
14 {
15
16     digitalWrite(2, HIGH); // set the voltage of pin 2 to HIGH (5V) so the
17     delay(1000);           // wait for 1000 milliseconds
18     digitalWrite(2, LOW);  // set the voltage of pin 2 to LOW (0V)
19     delay(1000);           // wait for 1000 milliseconds
20 }
21
22
```

1.3. Breadboard:



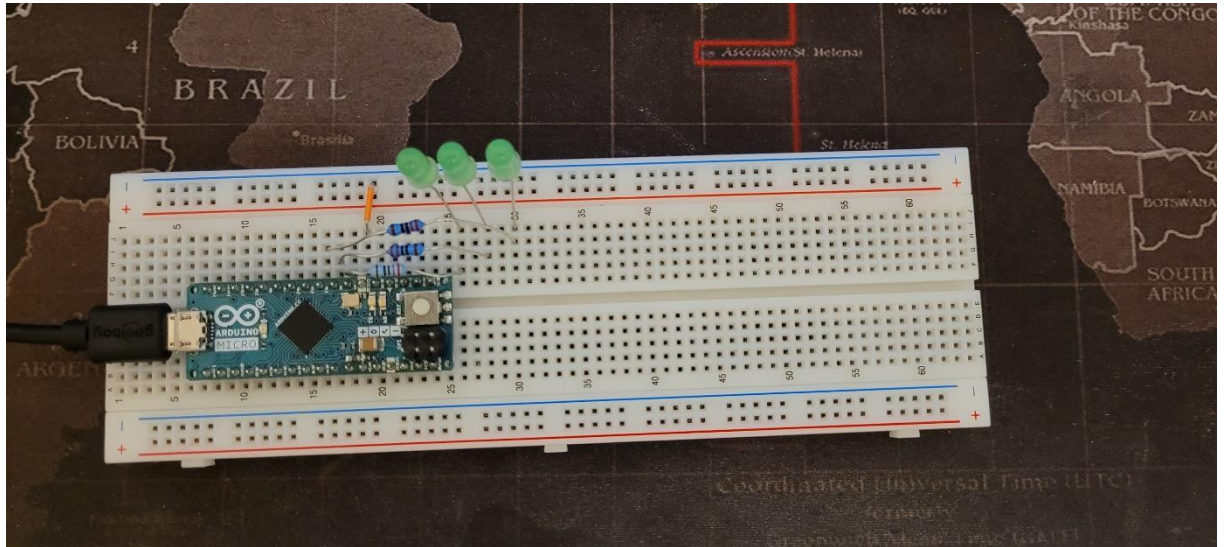
Code:

```
EXPLORER  ...  main.cpp X

01_ARDUINO_BLINK
├── .pio
├── .pio\libdeps
├── HID-Project_ID523
├── Mouse_ID890
├── .vscode
└── src
    ├── 01_arduino_blink.ino
    ├── main.cpp
    ├── .gitignore
    └── platformio.ini

src > main.cpp > loop()
4  // this function is called once upon startup of the Arduino
5  void setup()
6  {
7      // pin 2 is connected to the external LED with a preresistor
8      // define it as an output pin so we can write to it
9      pinMode(2, OUTPUT);
10     pinMode(3, OUTPUT);
11 }
12
13 // this function is looped indefinitely after setup() is finished
14 void loop()
15 {
16     digitalWrite(2, HIGH); // set the voltage of pin 2 to HIGH (5V) so the L
17     delay(1000);           // wait for 1000 milliseconds
18     digitalWrite(2, LOW);  // set the voltage of pin 2 to LOW (0V)
19     delay(1000);           // wait for 1000 milliseconds
20
21     digitalWrite(3, HIGH); // set the voltage of pin 3 to HIGH (5V) so the L
22     delay(1000);           // wait for 1000 milliseconds
23     digitalWrite(3, LOW);  // set the voltage of pin 3 to LOW (0V)
24     delay(1000);           // wait for 1000 milliseconds
25 }
26
```

1.4. Breadboard:



Code:

```
EXPLORER  ...  main.cpp x
01_ARDUINO_BLINK
  .pio
  .pio\libdeps
    HID-Project_ID523
    Mouse_JD890
    .vscode
  src
    01_arduino_blinkino
      main.cpp
      editorconfig
      .gitignore
      platformio.ini

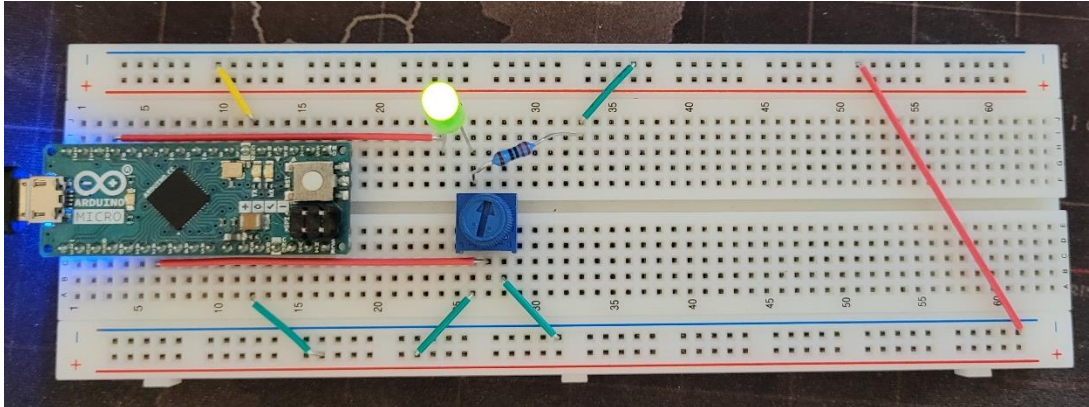
src > main.cpp > loop()
1 // include the Arduino standard library
2 #include <Arduino.h>
3
4 short binRep[3] = {}; // array for the binary representation
5 short num = 0, helperNum = 0; // var to store numbers on each loop
6
7 // this function is called once upon startup of the Arduino
8 void setup()
9 {
10 // pin 2, 3, 4 are connected to external LEDs with a presistor. define it as an output pin so we can write to it
11 pinMode(2, OUTPUT);
12 pinMode(3, OUTPUT);
13 pinMode(4, OUTPUT);
14 }
15
16 // this function is looped indefinitely after setup() is finished
17 void loop()
18 {
19 if (num == 0)
20   num = 0;
21
22   helperNum = num;
23
24   //calculate binary representation
25   for (int j = 0; j < sizeof(binRep); j++)
26   {
27     binRep[j] = helperNum % 2;
28     helperNum = helperNum / 2;
29   }
30
31   if (binRep[0] == 1)
32     digitalWrite(4, HIGH); // set the voltage of pin 2 to HIGH (5V) so the LED turns on
33
34   if (binRep[1] == 1)
35     digitalWrite(3, HIGH); // set the voltage of pin 3 to HIGH (5V) so the LED turns on
36
37   if (binRep[2] == 1)
38     digitalWrite(2, HIGH); // set the voltage of pin 3 to HIGH (5V) so the LED turns on
39
40   delay(1000); // wait for 1000 milliseconds
41   digitalWrite(2, LOW); // set the voltage of pin 2 to LOW (0V)
42   digitalWrite(3, LOW); // set the voltage of pin 3 to LOW (0V)
43   digitalWrite(4, LOW); // set the voltage of pin 4 to LOW (0V)
44   delay(1000); // wait for 1000 milliseconds
45
46   num++;
47 }
```

2. Die Funktion zum Abspielen von Tönen ist `tone(pin, frequency, duration)`. Nach ein bisschen Spielerei, habe ich ein cooles Tutorial zum Abspielen von Songs gefunden: [TUTORIAL](#). Als Star Wars Fan habe ich das Tutorial für eine Umsetzung vom Cantina-Band-Song genutzt. Die Noten habe ich aus der `pitches-Library` kopiert. Das Ergebnis ist auf dem Video zu hören. Der Code (ob der Länge) ist außerdem im Ordner.

3.

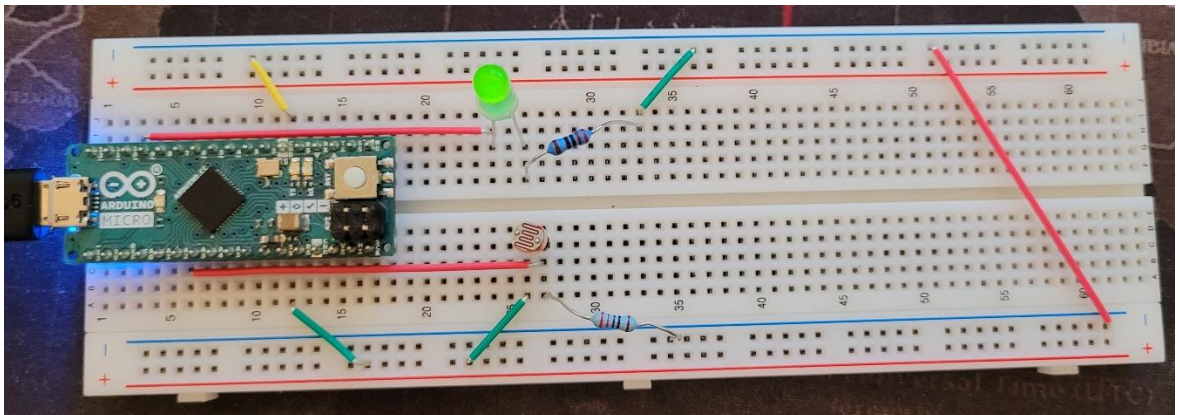
3.1.

Breadboard:



Problem: Im Code Sample fehlte das Mapping von `AnalogRead` (Werte 0-1023) auf `AnalogWrite` (Werte 0-255). Nachträglich anhand der [Dokumentation](#) verbessert und den Wert einfach durch 4 geteilt.

- 3.2. Damit die Schaltung mit Photowiderstand funktioniert, brauchen wir einen Spannungsteiler.
Breadboard:



4.

4.1.

4.2.