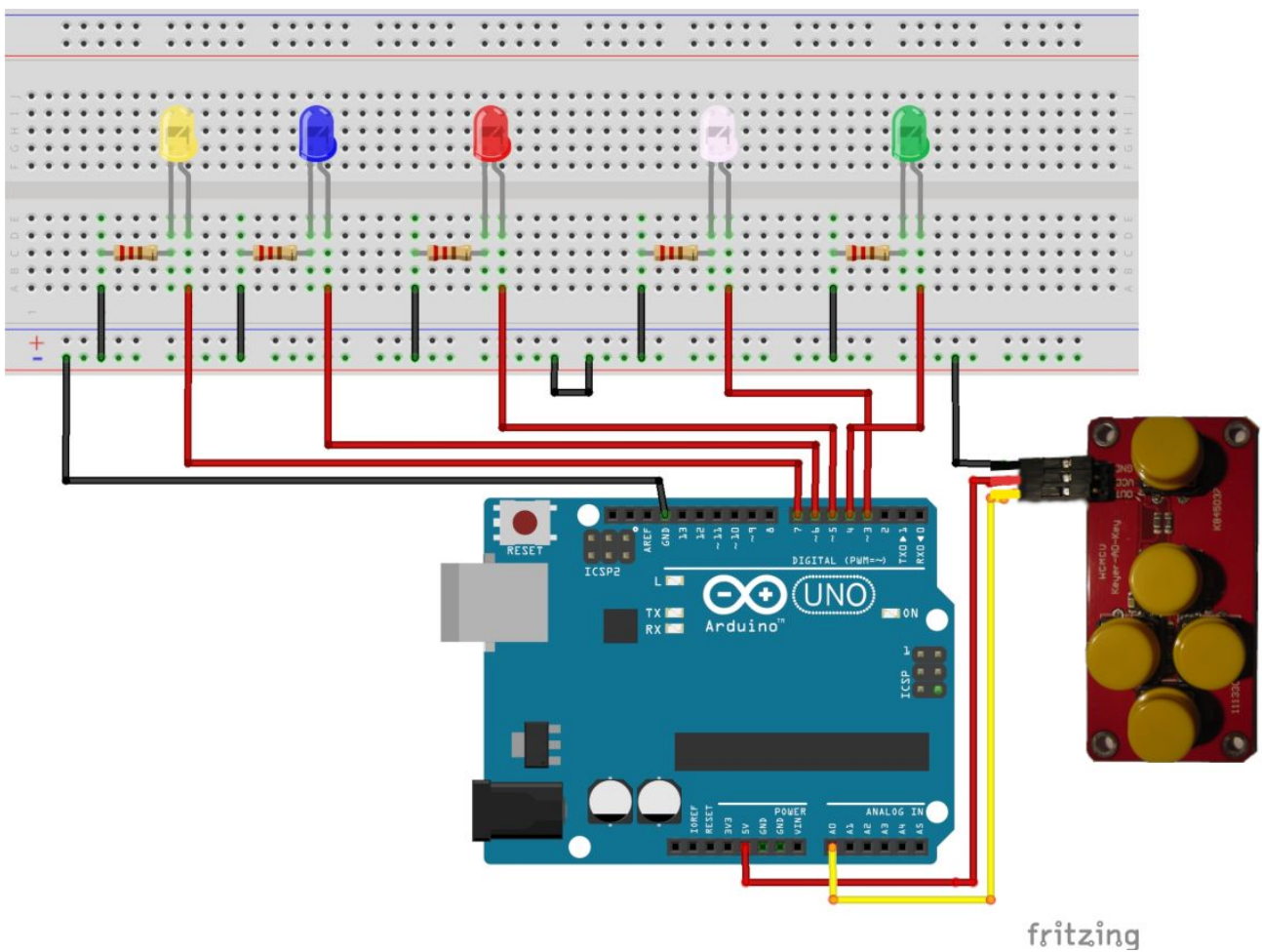


LEDs mit einem Tastenpad schalten

Benötigte Bauteile:

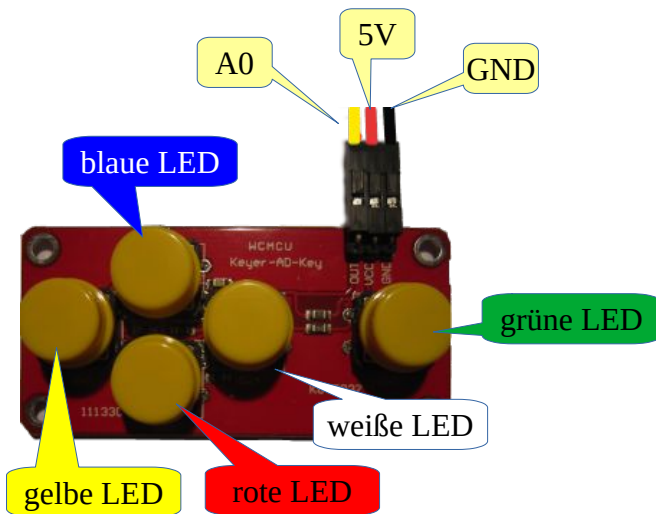
- 5 LEDs
- 5 Widerstände > 100 Ω
- Tastenpad
- Leitungsdrähte

Baue die Schaltung auf



fritzing

Die gedrückte Taste wird durch die Messung des analogen Wertes ermittelt. Leider kann dieser Wert nur ungefähr angegeben werden, so dass es beim laufenden Programm gelegentlich zu nicht erwünschten Effekten kommen kann.



Taster	Ungefährer analoger Wert
links	0 - 5
oben Mitte	30 - 40
unten Mitte	85 - 95
rechts	160 - 170
außen	330 - 380
nicht gedrückt	1023

Ein Tastendruck schaltet die entsprechende LED ein, erneuter Tastendruck schaltet sie wieder aus.

Für die Abfrage von Wertebereichen muss ...

```
# include <stdio.h>
```

... im Kopf eingebunden werden

Definiere ein Array für die LEDs ...

```
int LED[5] = {3, 4, 5, 6, 7};
```

... und eins für den aktuellen Status der LEDs:

```
bool Status[5] = {false, false, false, false, false};
```

false → aus, true → an

Für das Auslesen der Taster sind die Variablen ...

```
int Taster;  
int Analogwert;
```

... zuständig.



Eine int-Funktion erledigt das Auslesen des analogen Werts. Eine Besonderheit ist hier das Abfragen des Wertes mit switch mit einem Wertebereich.



Denke daran, dass die Funktion Tasterabfrage außerhalb des setup- und des loop-Teils steht.

```
int Tasterabfrage()
{
    Analogwert = analogRead(A0);
    delay(200);
    Serial.println(Analogwert);

    /*
    A0 gibt je nach gedrückten Taster einen Wert aus
    über den Seriellen Monitor wird dieser Wert angezeigt
    und kann dann eventuell angepasst werden
    Taster 1 → links
    Taster 2 → oben Mitte
    Taster 3 → unten Mitte
    Taster 4 → rechts
    Taster 5 → außen
    */

    switch (Analogwert)
    {
        case 0 ... 20:
            Taster = 1;
            break;
        case 30 ... 60:
            Taster = 2;
            break;
        case 70 ... 120:
            Taster = 3;
            break;
        case 150 ... 200:
            Taster = 4;
            break;
        case 300 ... 400:
            Taster = 5;
            break;
        default:
            return 0;
    }

    // gedrückten Taster zurückgeben
    return Taster;
}
```

Der setup-Teil weist mit einer for-Schleife jeder LED des Arrays den Wert OUTPUT zu.

```
void setup()
{
  Serial.begin(9600);
  for (int i = 0; i <= 4; i++)
  {
    pinMode(LED[i], OUTPUT);
  }
}
```

Der loop-Teil fragt mit Hilfe der Funktion Tasterabfrage den gedrückten Taster ab:

```
void loop()
{
  Taster = Tasterabfrage();

  // Taster links
  if (Taster == 1)
  {
    /*
     * gelbe LED Pin 7, LED[4] Status[4]
     * Status umdrehen, aus true wird false
     * aus false wird true
     */
    Status[4] = !Status[4];
    digitalWrite(LED[4], Status[4]);
  }

  // Taster oben Mitte
  if (Taster == 2)
  {
    // blaue LED Pin 6, LED[3] Status[3]
    Status[3] = !Status[3];
    digitalWrite(LED[3], Status[3]);
  }

  // Taster unten Mitte
  if (Taster == 3)
  {
    // rote LED Pin 5, LED[2] Status[2]
    . . .
    . . .
  }

  // Taster rechts
  if (Taster == 4)
  {
    // weiße LED Pin 4, LED[1] Status[1]
    . . .
    . . .
  }
}
```

```
// Taster rechts außen
if (Taster == 3)
{
  // grüne LED Pin 3, LED[0] Status[0]
  . . .
  . . .
}
}
```