

Projektbeschreibung

FeurIO – Interaktive Feuerschale



Team

Duy Quang, Chau, Matr.Nr 171 39 30

Doris, Ebenschwanger, Matr.Nr. 168 20 74

Konzept

FeurIO ist eine interaktive Feuerschale, die nicht nur mit Design punktet, sondern dem Benutzer auch die Möglichkeit gibt, selbst sein individuelles Feuer je nach Gefühlslage zusammenzustellen. FeurIO wird, wie ein echtes Feuer, mit einer Streichholzschachtel „angefacht“. Sofort sorgen kleine Ventilatoren, LED-Streifen und weißes Krepppapier für das typische Flackern der „Flammen“. In der Mitte kann der Benutzer z.B. einen Vulkanstein auf die Wärmeplatte platzieren und diesen nach ein paar Minuten als portablen Wärmespeicher in seiner Jackentasche verstauen. Zudem hat der Benutzer die Möglichkeit mit individuellen separaten Holzklötzchen die Farbe der Flammen seines FeurIOs einzustellen. Blau wird als beruhigend empfunden, Rot als anregend und Grün als vitalisierend. Zudem passt sich FeurIO zusätzlich der Umgebungshelligkeit an und leuchtet heller bei dunklen Räumen und weniger hell bei viel Lichteinfluss. Um FeurIO auszuschalten benötigt der Benutzer nur ein Streichholz, das durch eine Lichtschranke im Inneren die Komponenten deaktiviert. FeurIO ist somit eine leicht zu bedienende, interaktive Feuerschale, die den Benutzer mit Licht, Wärme und garantierter Entspannung versorgt.

Implementierung

Der Prototyp besteht aus einem Kunststoffbehälter, der mit einem dicken Pappdeckel verschlossen wurde. In diesem Deckel befinden sich 4 kleine 4cm Ventilatoren, jeweils umrandet mit einem LED-Streifen und weißem Krepppapier, das die Flammen symbolisieren soll. In der Mitte des Deckels befindet sich außerdem eine Wärmeplatte, auf der ein Peltier Element installiert wurde um ausreichend Wärme zu produzieren. Dieses wird mit 1,25 V versorgt. Im Inneren des Kunststoffbehälters befindet sich, neben der notwendigen Elektronik und des Arduino Micro-Controllers, eine Art Schiene, auf der die Holzelemente durch das FeurIO geschoben werden können. Diese Schiene ist mit Kupferfolie und Litzen bestückt um die Widerstände an den Holzklötzchen messen zu können.

An der Außenseite des Kunststoffbehälters befinden sich ein Fotosensor, der die Umgebungshelligkeit misst, ein Thermometer, das die Flackerbewegung der LEDs steuert, die Lichtschranke für die Deaktivierung und die Löcher für das „Feuerholz“. Um eine nachträgliche Verwertung der Elemente möglich zu machen, wurden alle zusammengehörigen Kabel in Spiralschläuche verstaut und ordnungsgemäß beschriftet.

Status, Erweiterungsmöglichkeiten

Unser Prototyp demonstriert das grundlegende Prinzip, kann allerdings noch ausgebaut werden.

Ästhetische Veränderungen:

- Stärkere Lüfter oder ein großer Lüfter in der Mitte wären leistungsfähiger und würden das Flackern noch besser unterstreichen
- Anstatt viele kleine Krepppapierstücke wären größere „Flammen“ ansprechender, allerdings nur in Verbindung mit stärkeren Lüftern möglich
- Ein kleinerer Behälter in der sich die Elektronik verbirgt würde die Schalenform besser unterstreichen

Technische Änderungen:

- Das Thermometer wird im Moment nur durch extreme Wertveränderungen angesprochen, somit sieht man die Veränderung der Blinkgeschwindigkeit der LED-Streifen im Moment eher ungenau
- Implementierung einer genaueren, besseren Lüfter Steuerung, da die momentan verbauten Lüfter mit geringerer Spannung nicht direkt anspringen.

Benutzerspezifische Änderungen:

- Zusatzfunktion für das „Feuerholz“, das nicht nur die Farbe der LED-Streifen anspricht, sondern auch Wärme durch die Lüfter abgibt
- Ausschalten durch „Auspusten“, da im Moment die Lichtschranke nur durch Unterbrechung mit einem relativ dicken Gegenstand (Streichholz) möglich ist
- Ebenso könnte man an der Oberseite zusätzlich noch Abstandsmesser einbauen, welche die LED Helligkeit reguliert, wenn der Benutzer seine Hand über diese Stelle hält

Bedienungsanleitung

Feurio – Interaktive Feuerschale

Setup

1. Stromzufuhr per USB – Port des Computers (bei einem Netz USB – Hub können die Ergebnisse abweichen)
2. Zusätzlich Stromzufuhr per Netzteil

Bedienungsanleitung

Der Benutzer kann mit dem Feurio wie folgt interagieren:

- Einschalten mit einer Streichholzschachtel (Drucksensor)
- Änderung der Farben durch „Feuerholz nachlegen“
- Auflegen eines portablen Wärmespeichers auf die Wärmeplatte
- Änderung der Helligkeit durch Abdunkelung des Raumes oder zusätzlicher Lichtquellen
- Ausschalten durch Streichholz (Lichtschanke)

Interaktionskonzept

Unser Konzept besteht daraus, dass wir alle natürlichen Interaktionen mit dem Feuer abdecken wollen. Dazu gehören sowohl aktive Interaktionen wie z.B. das Ein- und Ausschalten des Feuers, das nachlegen von Feuerholz als auch passive Interaktionen wie z.B. die Abgabe von Wärme, die Reaktionen auf Wärme und Licht.

Ausführlichere Beschreibung des Interaktionskonzeptes

Aktive Interaktionen:

- **Einschalten:** Durch das Streichen über die Streichholzschachtel wird der Prototyp eingeschaltet.
- **Ausschalten:** Durch das Einführen eines kleinen Objektes (hier: Streichholz) in die dafür vorgesehene Öffnung, kann man den Prototyp ausschalten.
- **Hinzufügen von „Feuerholz“:** Durch das Einführen der Holzblöcke kann die Farbe des Feuers geändert werden. Dazu haben wir drei verschiedene Holzblöcke bereitgestellt (Änderung der Farbe auf Rot, Grün oder Blau).

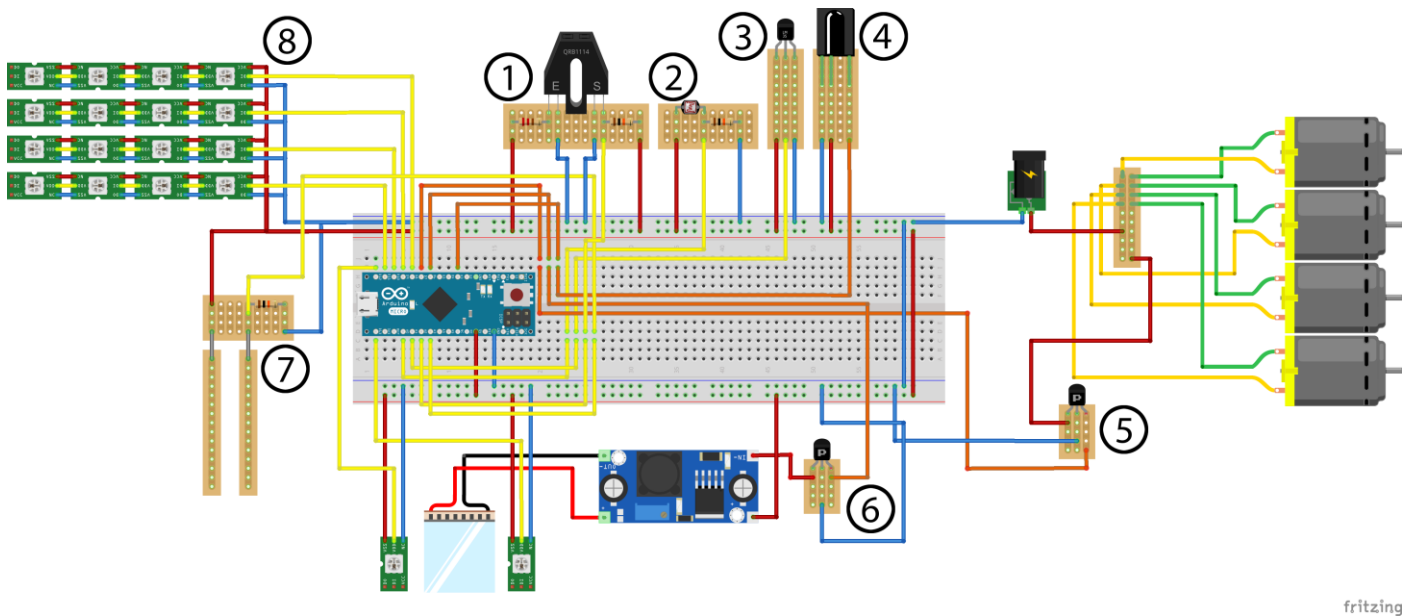
Passive Interaktionen:

- **Abgabe von Wärme:** Die Abgabe von Wärme funktioniert über das Peltierbauteil. Dieses erzeugt bei geringer Spannung viel Wärme. Diese kann unter anderem für die Erwärmung eines portablen Wärmespeichers genutzt werden (z.B. Lavastein).
- **Wärme:** Der Prototyp reagiert auf die Zimmertemperatur (wenn die Handfläche aufgelegt ist auch auf die Körpertemperatur). Dies wird durch die Blinkgeschwindigkeit der LEDs dargestellt, dabei gilt jedoch, je kälter es ist desto langsamer ist die Blinkgeschwindigkeit.
- **Licht:** Der Prototyp reagiert auch auf die Umgebungshelligkeit, dabei gilt, je dunkler der Raum, desto heller leuchten die LEDs. Die Umgebungshelligkeit kann auch von den Handflächen beeinflusst werden, indem man die Hand auf den Photosensor legt.

FeurIO – Interaktive Feuerschale

Implementierung

Das FeurIO lässt sich in acht Unterbauteile einteilen (siehe Bild).



fritzing

- 1) Dieses Bauteil stellt eine Lichtschranke dar. Sie läuft über zwei Stromkreise. Der erste Stromkreis (linke Hälfte) betreibt eine kleine LED in dem Bauteil. Um zu verhindern, dass die LED durchbrennt, wird ein 220 OHM Vorwiderstand eingesetzt.
In dem zweiten Stromkreis befindet sich ein Photosensor, dieser ändert seinen Widerstandswert, wenn sich ein Objekt zwischen der LED und dem Photosensor befindet. In dem Stromkreis befindet sich ein 10k OHM Widerstand, der einen Kurzschluss verhindern soll.
Signal Input: A2
- 2) Hierbei handelt es sich um einen Photosensor, der dazu dient die Umgebungshelligkeit zu messen. Auch hier wird zusätzlich ein 10k OHM Widerstand in Reihe geschaltet um einen Kurzschluss zu verhindern.
Signal Input: A0
- 3) Bei diesem Bauteil handelt es sich um ein Thermometer. Er soll die Raumtemperatur oder die Handwärme (wenn diese aufgelegt wird) messen und diese für die Verarbeitung an den Arduino weitergeben.
Signal Input: A1
- 4) Dieses Bauteil stellt einen Infrarot Empfänger dar. Durch diesen erhält der Arduino das Signal Befehle auszuführen. Wir verwenden ihn nur für das Ein- und Ausschalten des FeurIOs.
Signal Input: 3
- 5) Bei den Motoren handelt es sich um vier Lüfter, die die Flammenteile zum Bewegen bringen. Diese werden über eine externe Stromquelle mit 12 V betrieben. In der Schaltung befindet sich ein NPN-Transistor, der für das Ansteuern der Lüfter über den Arduino zuständig ist.
Signal Output: 7

- 6) Das Peltierbauteil erzeugt in unserem Fall Wärme. Dieser wird ebenfalls über ein NPN-Transistor geschaltet. Vor dem Peltierbauteil befindet sich ein Spannungswandler, der die Spannung von 5 Volt auf 1,25 Volt runterregelt. Neben dem Peltierbauteil befinden sich noch zwei LEDs, welche über weitere PINs angesteuert werden. Diese werden angeschaltet, wenn das Peltierbauteil aktiv ist.

Signal Output (Peltier): 6

Signal Output (LED): 13, 12

- 7) Hierbei handelt es sich um ein Ohmmeter. Mit diesem wird der Widerstand der einzelnen Holzblöcke gemessen. Dadurch können wir bestimmen um welchen Holzblock es sich handelt. Die zwei Leiterbahnen sollen jeweils ein Kupferstreifen darstellen. Über weitere Berechnungen erhalten wir den unbekannten Widerstand:

```
buffer = rawOhmmeter * Vin;  
Vout = (buffer) / 1024.0;  
buffer = (Vin / Vout) - 1;  
R2 = R1 * buffer;
```

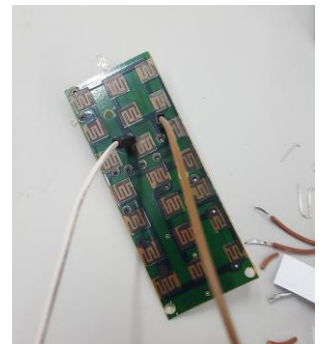
rawOhmmeter: vom Pin abgegriffener Wert
buffer: Zwischenspeicher
Vin: Fester Wert (hier 5)
Vout: Berechneter Wert
R1: Fester Wert (10k)
R2: Unbekannter Widerstand

Signal Input: A1

- 8) Bei diesen Bauteilen handelt es sich um vier separate LED – Strips mit jeweils vier LEDs. Diese dienen zur Beleuchtung der Flammen und werden einzeln angesteuert.

Signal Output: 11, 10, 9, 8

- 9) Zusätzlich zu der Schaltung haben wir eine IR-Fernbedienung, diese wurde in eine Streichholzschachtel verbaut und wir über einen Drucksensor, der außerhalb der Schachtel befestigt ist.



Anhang

FeurIO – Interaktive Feuerschale

Code: FeurIO.ino

```
#include <IRLib.h>
#include <Adafruit_NeoPixel.h>

//PIN Zuweisung OUTPUT
uint8_t LED_PIN_PEL = 12;
uint8_t LED_PIN_TIER = 13;
uint8_t LED_STRIP [4] = {11, 10, 9, 8};
uint8_t FAN_PIN = 7;
uint8_t PELTIER_PIN = 6;

//PIN Zuweisungen INPUT
uint8_t PHOTO_INPUT = A0;
uint8_t TEMP_INPUT = A1;
uint8_t SCHRANKE = A2;
uint8_t OHMMETER = A3;
uint8_t IR_INPUT = 3;

//VARS
uint8_t LED_STRIP_ELEMENT = 3;
Adafruit_NeoPixel strip [4];
Adafruit_NeoPixel strip_pel;
Adafruit_NeoPixel strip_tier;
uint8_t photoValue = 0;
uint8_t tempValue = 0;
uint16_t rawOhmmeter = 0;
uint8_t Vin = 5;
float Vout = 0;
float R1 = 10000;
float R2 = 0;
float buffer = 0;
bool state = false;
IRrecv My_Receiver(IR_INPUT);
IRdecode My_Decoder;

void setup() {
  pinMode(FAN_PIN, OUTPUT);
  pinMode(PELTIER_PIN, OUTPUT);
  strip_pel = Adafruit_NeoPixel(1, LED_PIN_PEL, NEO_GRB + NEO_KHZ800);
  strip_tier = Adafruit_NeoPixel(1, LED_PIN_TIER, NEO_GRB + NEO_KHZ800);
  strip[0] = Adafruit_NeoPixel(LED_STRIP_ELEMENT, LED_STRIP[0], NEO_GRB + NEO_KHZ800);
  strip[1] = Adafruit_NeoPixel(LED_STRIP_ELEMENT, LED_STRIP[1], NEO_GRB + NEO_KHZ800);
  strip[2] = Adafruit_NeoPixel(LED_STRIP_ELEMENT, LED_STRIP[2], NEO_GRB + NEO_KHZ800);
  strip[3] = Adafruit_NeoPixel(LED_STRIP_ELEMENT, LED_STRIP[3], NEO_GRB + NEO_KHZ800);
  for (uint8_t i = 0; i <= LED_STRIP_ELEMENT; i++) {
    strip[i].begin();
    strip[i].show();
  }
  My_Receiver.enableIRIn();
}
```

```

void loop() {
  strip_pel.setPixelColor(0, strip_pel.Color(255, 255, 255));
  strip_tier.setPixelColor(0, strip_tier.Color(255, 255, 255));
  strip_pel.begin(); strip_tier.begin();
  strip_pel.show(); strip_tier.show();
  digitalWrite(LED_PIN_PEL, HIGH);
  digitalWrite(LED_PIN_TIER, HIGH);
  rawOhmmeter = analogRead(OHMMETER);
  photoValue = analogRead(PHOTO_INPUT);
  tempValue = analogRead(TEMP_INPUT);
  if (photoValue >= 254) {
    photoValue = 254;
  }
  if (tempValue >= 254) {
    tempValue = 254;
  }
  for (uint8_t i = 0; i <= LED_STRIP_ELEMENT; i++) {
    strip[i].setBrightness(photoValue);
  }
  if (state) {
    analogWrite(FAN_PIN, 255);
    digitalWrite(PELTIER_PIN, HIGH);
  }
  for (uint8_t i = 0; i <= LED_STRIP_ELEMENT; i++) {
    if (R2 > 2000 && R2 < 2300) { //ORANGE
      fireLight(strip[i].Color(242, 164, 14), strip[i].Color(238, 56, 14), i, tempValue);
    } else if (R2 > 5500 && R2 < 7000) { //ROT
      fireLight(strip[i].Color(238, 56, 14), strip[i].Color(160, 10, 10), i, tempValue);
    } else if (R2 < 1400 && R2 > 1000) { //GRUEN
      fireLight(strip[i].Color(124, 252, 0), strip[i].Color(0, 128, 0), i, tempValue);
    } else if (R2 > 1400 && R2 < 2000) { //TUERKIS
      fireLight(strip[i].Color(14, 231, 231), strip[i].Color(14, 231, 101), i, tempValue);
    } else if (state) { //Default ON
      fireLight(strip[i].Color(255, 0, 0), strip[i].Color(255, 130, 0), i, tempValue);
    }
  }
}
if (rawOhmmeter) {
  buffer = rawOhmmeter * Vin;
  Vout = (buffer) / 1024.0;
  buffer = (Vin / Vout) - 1;
  R2 = R1 * buffer;
}
if (My_Receiver.GetResults(&My_Decoder)) {
  My_Decoder.decode();
  if (My_Decoder.value == 16617583) {
    state = true;
  }
  My_Receiver.resume();
}
if (analogRead(SCHRANKE) < 540) {
  for (uint8_t i = 0; i <= LED_STRIP_ELEMENT; i++) {
    ledOff(i);
  }
  digitalWrite(FAN_PIN, LOW);
  digitalWrite(PELTIER_PIN, LOW);
  state = false;
}
}

```

```
void ledOff(uint8_t pin) {  
  for (uint8_t i = 0; i < LED_STRIP_ELEMENT; i++) {  
    strip[pin].setPixelColor(i, 0);  
  }  
  strip[pin].show();  
}  
  
void fireLight(uint32_t c, uint32_t f, uint8_t pin, uint8_t wait) {  
  strip[pin].setPixelColor(0, c);  
  strip[pin].setPixelColor(1, f);  
  strip[pin].setPixelColor(2, c);  
  strip[pin].show();  
  delay(255-wait);  
  strip[pin].setPixelColor(0, f);  
  strip[pin].setPixelColor(1, c);  
  strip[pin].setPixelColor(2, f);  
  strip[pin].show();  
}
```


Anhang

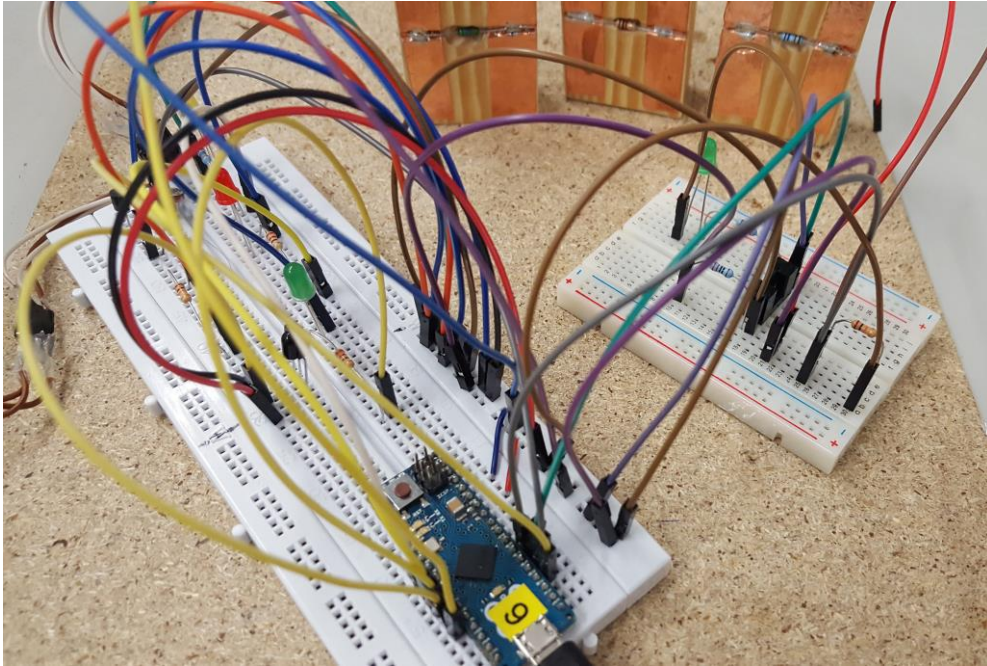


Abbildung 1 – Testschaltung aller Sensoren (im Hintergrund befinden sich die Feuerholzblöcke)

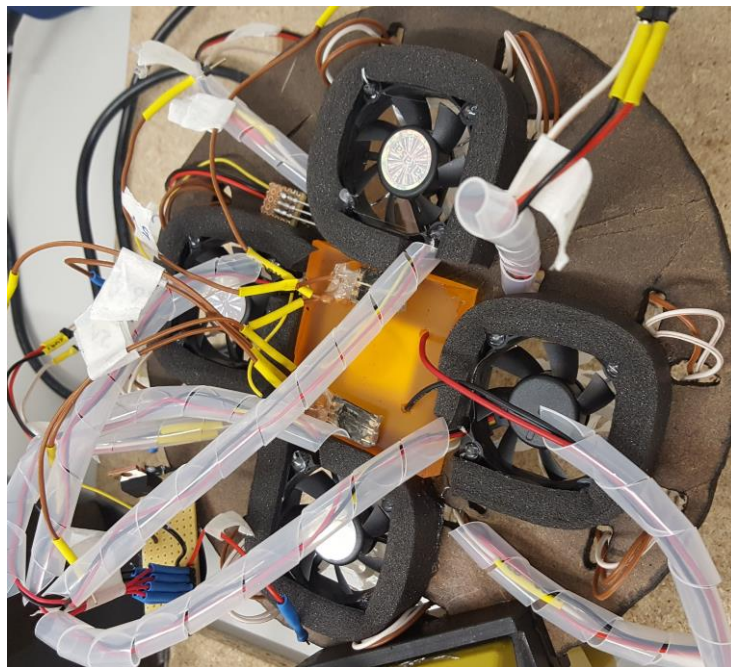


Abbildung 2 – Lüfter, LED, Peltier Bauteile fixiert an den Deckel

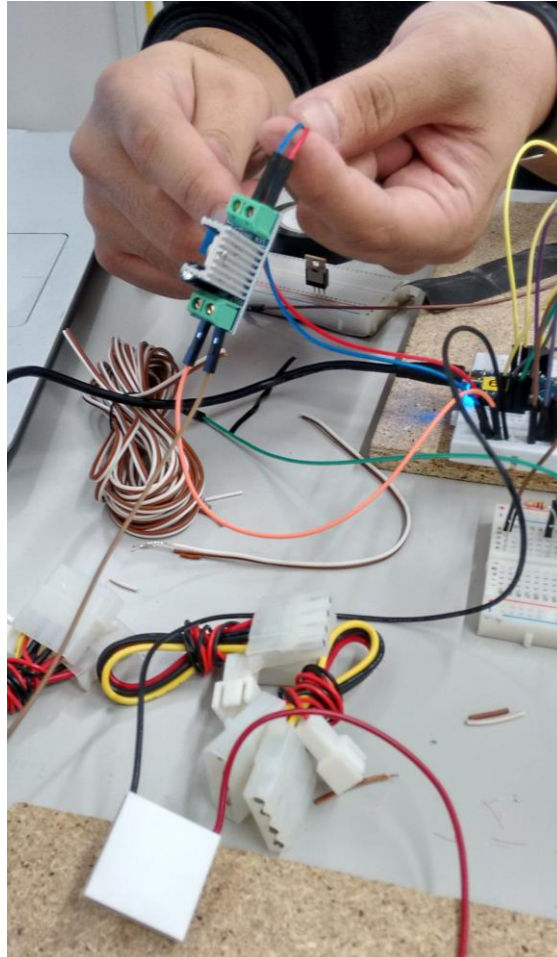


Abbildung 3 – Unten Peltier Bauteil, Oben Spannungswandler

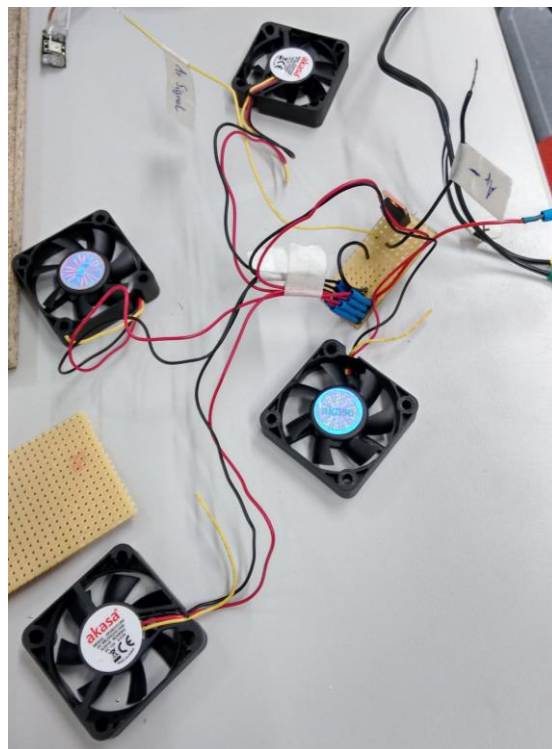


Abbildung 4 – Lüfter Bauteil



Abbildung 5 – Einbau aller Elemente in die „Schale“

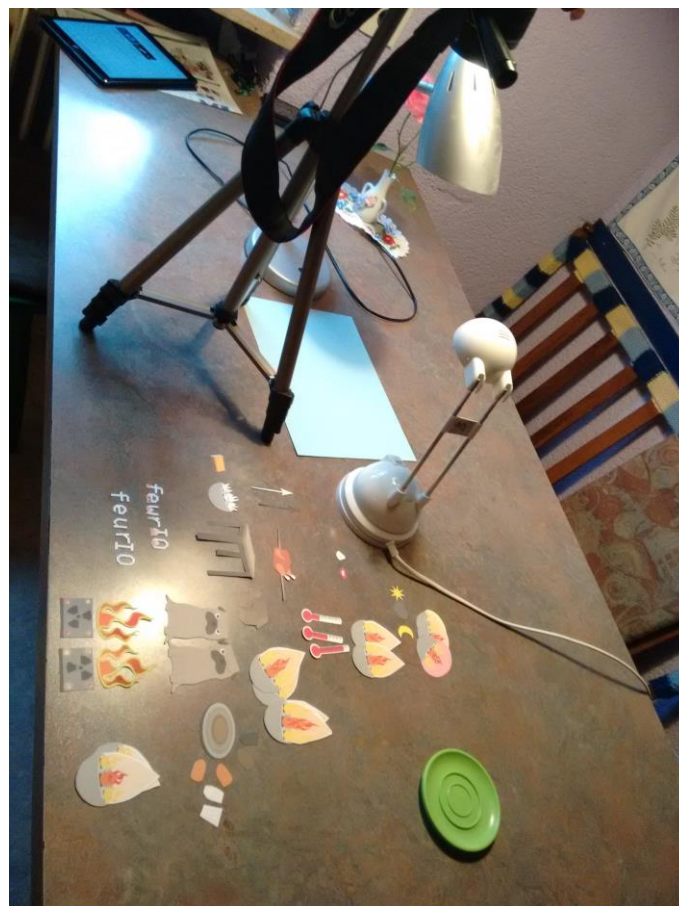


Abbildung 6 – Setup für Konzeptvideo