

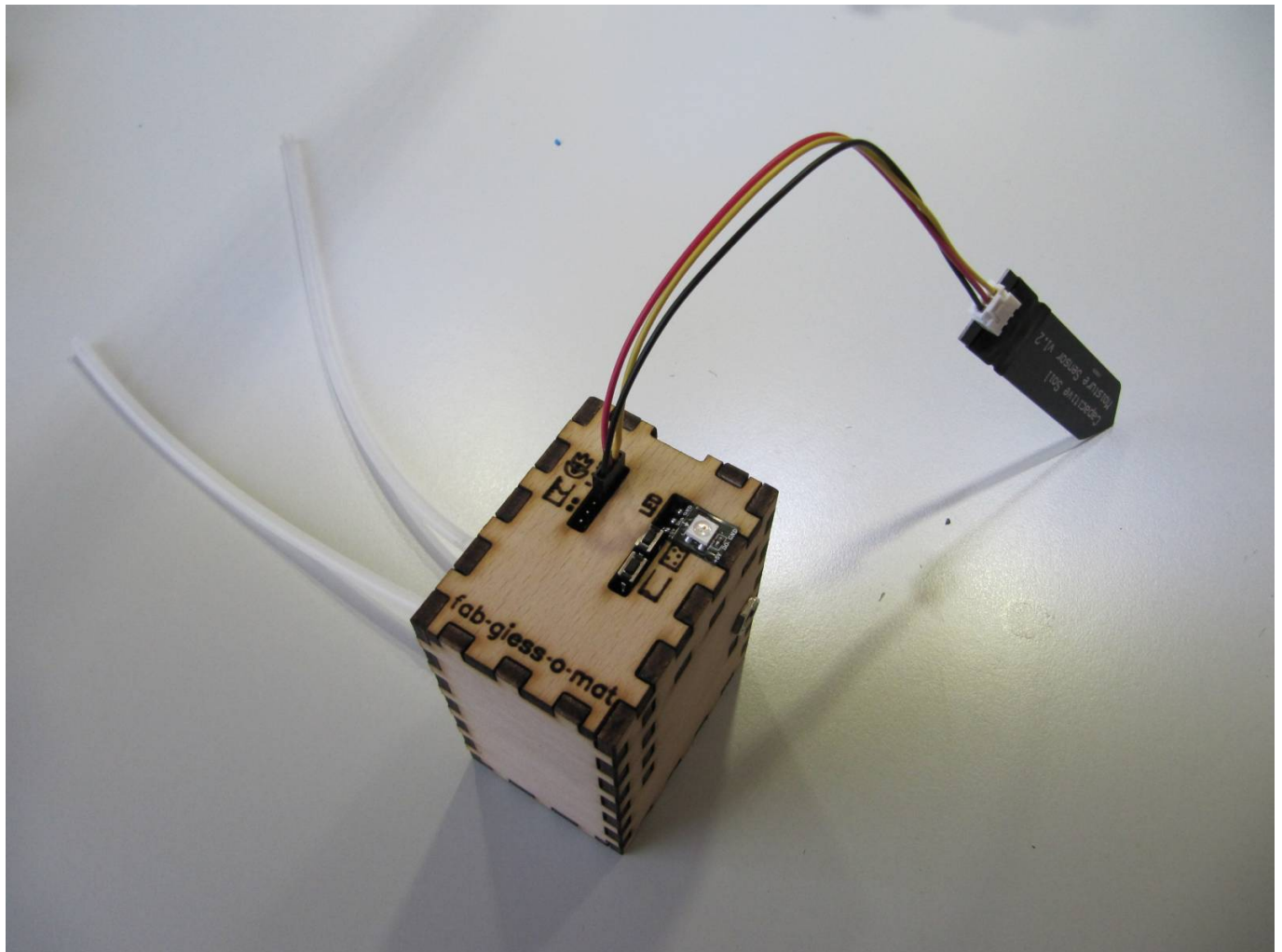
# fab-giess-o-mat

Eine Mikrocontroller-gesteuerte Bewässerungsanlage für Zimmerpflanzen

## Features

- Feuchtigkeitsmessung
- Steuerung einer oder mehrerer Pumpen
- Wassertank-Füllstandsmessung
- LED Anzeige
- USB Schnittstelle zum konfigurieren
- optionales WiFi Modul mit Webinterface zum konfigurieren und Loggen der Messwerte

**Projektstatus:** Gerät ist einsatzbereit, aber die Software für das Wifi Modul ist noch nicht fertig.





## Motivation

Es gibt schon einige DIY Bewässerungssysteme, die aber alle so ihre Macken haben. zu kompliziert, zu frickelig, zu teuer, zu wenig features, schlechte Doku, veraltete Hardware. Ich möchte mein altes [Giess-o-mat](#) Projekt modernisieren.

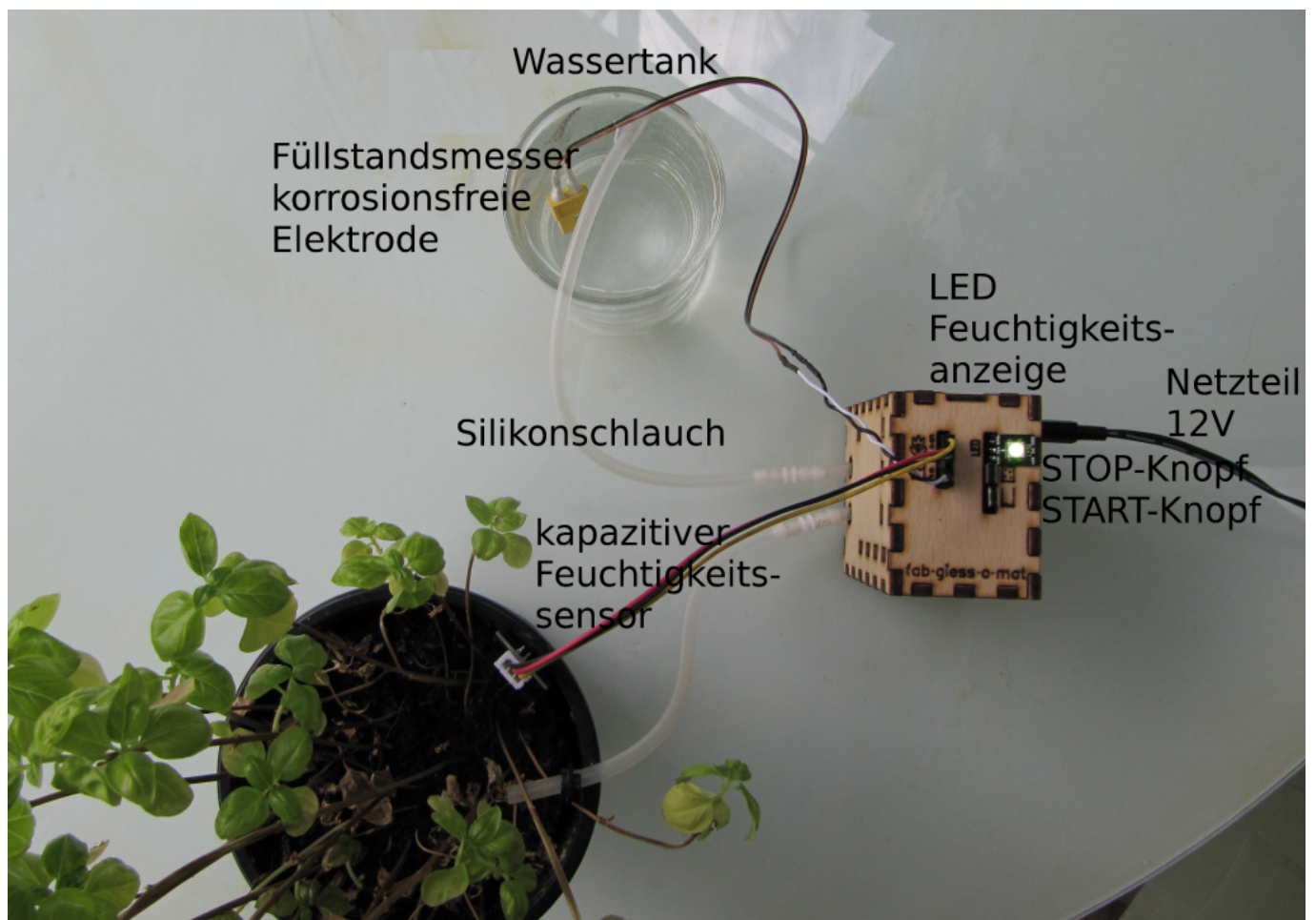
Es gibt da dieses paar lustige Projekt [Windowfarm](#), das beliebt ist zum Nachbauen. Ein Beispiel (Prinzip ohne Pumpe):

[http://stadtmachtsatt.de/wp-content/uploads/2013/12/Bauanleitung-Windowfarm\\_DinA4.pdf](http://stadtmachtsatt.de/wp-content/uploads/2013/12/Bauanleitung-Windowfarm_DinA4.pdf)

Eine weitere Inspiration ist das [fabfarmer](#) Projekt

Das Projekt sollte auch von Leuten ohne besondere technische Vorkenntnisse zusammengebaut werden können. Deshalb möchten wir günstige Standard Hardware verwenden, keine selbstgebasteten Platinen, soweit das möglich ist. Der Löt-Aufwand sollte nicht zu hoch sein.

## Aufbau



## Bedienung und LED Anzeige

Beim Druck auf den **START-Knopf** wird der aktuelle Feuchtigkeits-Messwert als „Schaltschwelle

trocken“ gespeichert und die Pumpe eingeschaltet.

Beim Druck auf den **STOP-Knopf** wird der aktuelle Feuchtigkeits-Messwert als „Schaltschwelle nass“ gespeichert und die Pumpe ausgeschaltet. Außerdem wird die Zeit zwischen ein- und ausschalten gemessen und als Einschaltzeit gespeichert. Die LED wird grün.

Bei einem unkonfiguriertem System, oder wenn die Schaltschwelle trocken größer oder gleich der Schaltschwelle nass ist, leuchtet die LED blau.

Die Farbe der LED ändert sich von grün übergelb und orange zu rot, wenn die Schaltschwelle trocken erreicht wird. Dann wird die Pumpe automatisch eingeschaltet. Die Pumpe bleibt so lange an, bis die Einschaltzeit abgelaufen ist.

Der **Füllstandssensor** ist optional. Es handelt sich um zwei einfache Elektroden, die Stromfluss durch das Wasser im Tank messen. Wenn einmal der Kontakt geschlossen wurde, ist der Füllstandssensor aktiv. Wird der Kontakt geöffnet, weil der Tank leer ist, blinkt die LED blau. **Achtung:** keine Kupferkontakte verwenden. Die oxidieren schnell.

## Konfiguration über PC

Der fab-giess-o-mat kann über einen Mini-USB Kabel an einen PC angeschlossen werden. Es wird ein Terminalprogramm wie TeraTerm oder minicom benötigt.

\* fab-giess-o-mat \*

Pumpe an Pin D2

Sensor an Pin A0

Füllstandssensor an Pin A5

Taster Start an Pin D3

Taster Stop an Pin D6

RGB LED an Pin D8

vorletzte Pump-Zeit: 0.0.1970 00:00:00

letzte Pump-Zeit: 0.0.1970 00:00:00

\* fab-giess-o-mat Hauptmenue \*

u - Uhr stellen

e - maximale Einschaltzeit [sek]: 6

a - minimale Ausschaltzeit [std:min]: 16:00

t - Schaltschwelle trocken: 565

n - Schaltschwelle nass: 427

s - Sensor lesen

i - Pumpe ein

o - Pumpe aus

## Umfeld

Es wäre cool, für das Fablab einen Prototypen zu bauen, leider ist dort zu wenig Platz.

# Technische Beschreibung / Details

## Mikrokontroller

**Arduino Nano** (klein, kompakt und gibts für ~2€ beim Chinesen deines Vertrauens)

Man kann sich mit dem PC an die USB Schnittstelle anflanschen und den Giess-o-mat über ein Menü in einem [Terminalprogramm](#) steuern.

## Wifi Modul

ESP32 zum Aufstecken - soll auch ohne funktionieren. Mit ESPs habe ich schon experimentiert und die Erfahrung gemacht, dass die nicht besonders zuverlässig sind. Für eine Bewässerungssteuerung ungeeignet, aber ok um übers Internet ein paar Messwerte abrufbar zu machen oder Einstellungen ändern

Den ESP kann man verwenden, um die Messwerte über einen längeren Zeitpunkt zu loggen und eine Graphik im Webinterface generieren.

## Feuchtigkeitssensoren

Die aktuelle Version benutzt kapazitive Sensoren mit ADC Ausgang (bei Aliexpress suchen nach capacitive moisture sensor). Der Sensor besteht aus 2 Kondensatorflächen. Es fließt kein Strom durch die blumenerde. Je feuchter die Erde, desto höher die Kapazität. Der Messwert wird einfach mit `analogRead()` eingelesen.

Alternative: ein einfacher kapazitiven Sensor nur aus 2 Kondensatorplatten und einem hochohmigen Widerstand. Der Arduino schaltet zuerst einen Ausgang auf high und lädt den Kondensator auf. Dann wird der Pin auf ADC-Input umgeschaltet und nach gemessen, wie lange es dauert, bis der Kondensator entladen ist. Der Sourcecode für die Ansteuerung befindet sich in der Datei `sensor.ino`, wird aber nicht benutzt.

## Pumpe

Wir benutzen eine [Schlauchpumpe](#) (Peristaltikpumpen), die über einen Transistor eingeschaltet werden kann.

## Gehäuse

Das Gehäuse wird aus einer 4mm starken Platte ausgeschnitten.

Hier ist das Design für den Lasercutter:

<https://github.com/chbergmann/fab-giess-o-mat/blob/master/cad/giessomat-gehauese.svg>

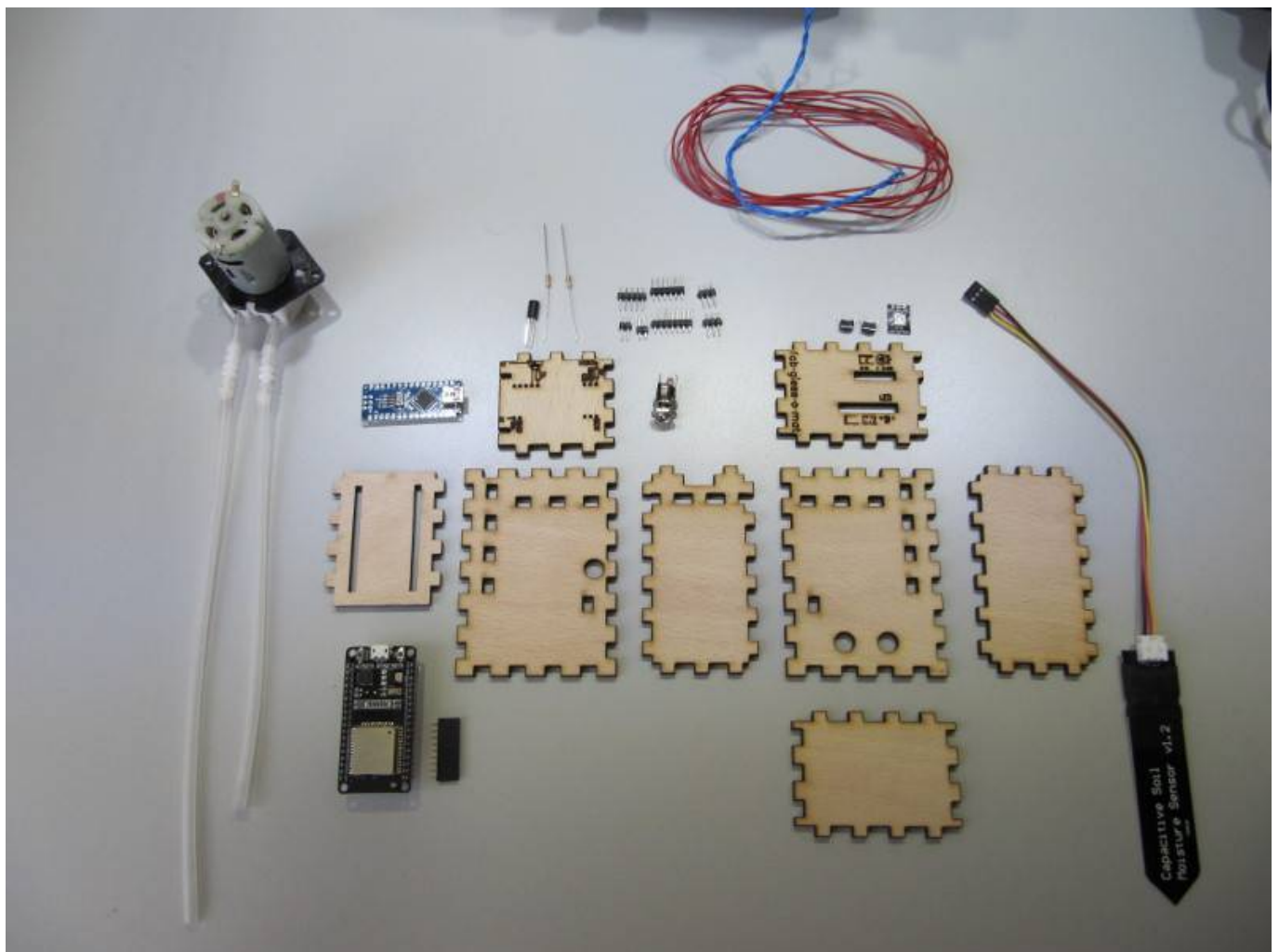
Einstellungen:

grün: gravieren



braun: ignorieren. Ist nur zur Dokumentation da

- 1 Schlauchpumpe 12V
- 1 Arduino Nano
- 2 taktile Taster
- 1 RGB LED WS2812
- 1 Transistor NPN, min. 12V, 1A
- 1 Hohlstecker-Buchse 5,5mm 2,1mm
- 1 Netzteil 12V, min. 1A
- 1 Feuchtigkeitssensor
- Silikonschlauch 5x3mm
- 1 Widerstand 1k
- Stiftleisten 2,54mm. Stückelung: 1x1, 1x2, 2x3, 1x5, 1x6, 1x7
- Draht, 2 Farben
- Schrumpfschlauch 2mm
- 1 Holz oder Kunststoffplatte 4 x 240 x 150 mm (Lastercutter-Design)



# Konfiguration über PC

Der fab-giess-o-mat kann über einen Mini-USB Kabel an einen PC angeschlossen werden. Es wird ein Terminalprogramm wie TeraTerm oder minicom benötigt.

\* fab-giess-o-mat \*

Pumpe an Pin D2

Sensor an Pin A0

Füllstandssensor an Pin A5

Taster Start an Pin D3

Taster Stop an Pin D6

RGB LED an Pin D8

vorletzte Pump-Zeit: 0.0.1970 00:00:00

letzte Pump-Zeit: 0.0.1970 00:00:00

\* fab-giess-o-mat Hauptmenue \*

u - Uhr stellen

e - maximale Einschaltzeit [sek]: 6

a - minimale Ausschaltzeit [std:min]: 16:00

t - Schaltschwelle trocken: 565

n - Schaltschwelle nass: 427

s - Sensor lesen

i - Pumpe ein

o - Pumpe aus

## Umfeld

Es wäre cool, für das Fablab einen Prototypen zu bauen, leider ist dort zu wenig Platz.

## Technische Beschreibung / Details

### Mikrokontroller

**Arduino Nano** (klein, kompakt und gibts für ~2€ beim Chinesen deines Vertrauens)

Man kann sich mit dem PC an die USB Schnittstelle anflanschen und den Giess-o-mat über ein Menü in einem [Terminalprogramm](#) steuern.

### Wifi Modul

ESP32 zum Aufstecken - soll auch ohne funktionieren. Mit ESPs habe ich schon experimentiert und die Erfahrung gemacht, dass die nicht besonders zuverlässig sind. Für eine Bewässerungssteuerung ungeeignet, aber ok um übers Internet ein paar Messwerte abrufbar zu machen oder Einstellungen ändern

Den ESP kann man verwenden, um die Messwerte über einen längeren Zeitpunkt zu loggen und eine Graphik im Webinterface generieren.

## Feuchtigkeitssensoren

Die aktuelle Version benutzt kapazitive Sensoren mit ADC Ausgang (bei Aliexpress suchen nach capacitive moisture sensor). Der Sensor besteht aus 2 Kondensatorflächen. Es fließt kein Strom durch die blumenerde. Je feuchter die Erde, desto höher die Kapazität. Der Messwert wird einfach mit analogRead() eingelesen.

Alternative: ein einfacher kapazitiven Sensor nur aus 2 Kondensatorplatten und einem hochohmigen Widerstand. Der Arduino schaltet zuerst einen Ausgang auf high und lädt den Kondensator auf. Dann wird der Pin auf ADC-Input umgeschaltet und nach gemessen, wie lange es dauert, bis der Kondensator entladen ist. Der Sourcecode für die Ansteuerung befindet sich in der Datei sensor.ino, wird aber nicht benutzt.

## Pumpe

Wir benutzen eine [Schlauchpumpe](#) (Peristaltikpumpen), die über einen Transistor eingeschaltet werden kann.

## Gehäuse

Das Gehäuse wird aus einer 4mm starken Platte ausgeschnitten.

Hier ist das Design für den Lasercutter:

<https://github.com/chbergmann/fab-giess-o-mat/blob/master/cad/giessomat-gehauese.svg>

Einstellungen:

grün: gravieren

rot: schneiden

schwarz: schneiden

braun: ignorieren. Ist nur zur Dokumentation da

## Bauteile-Liste

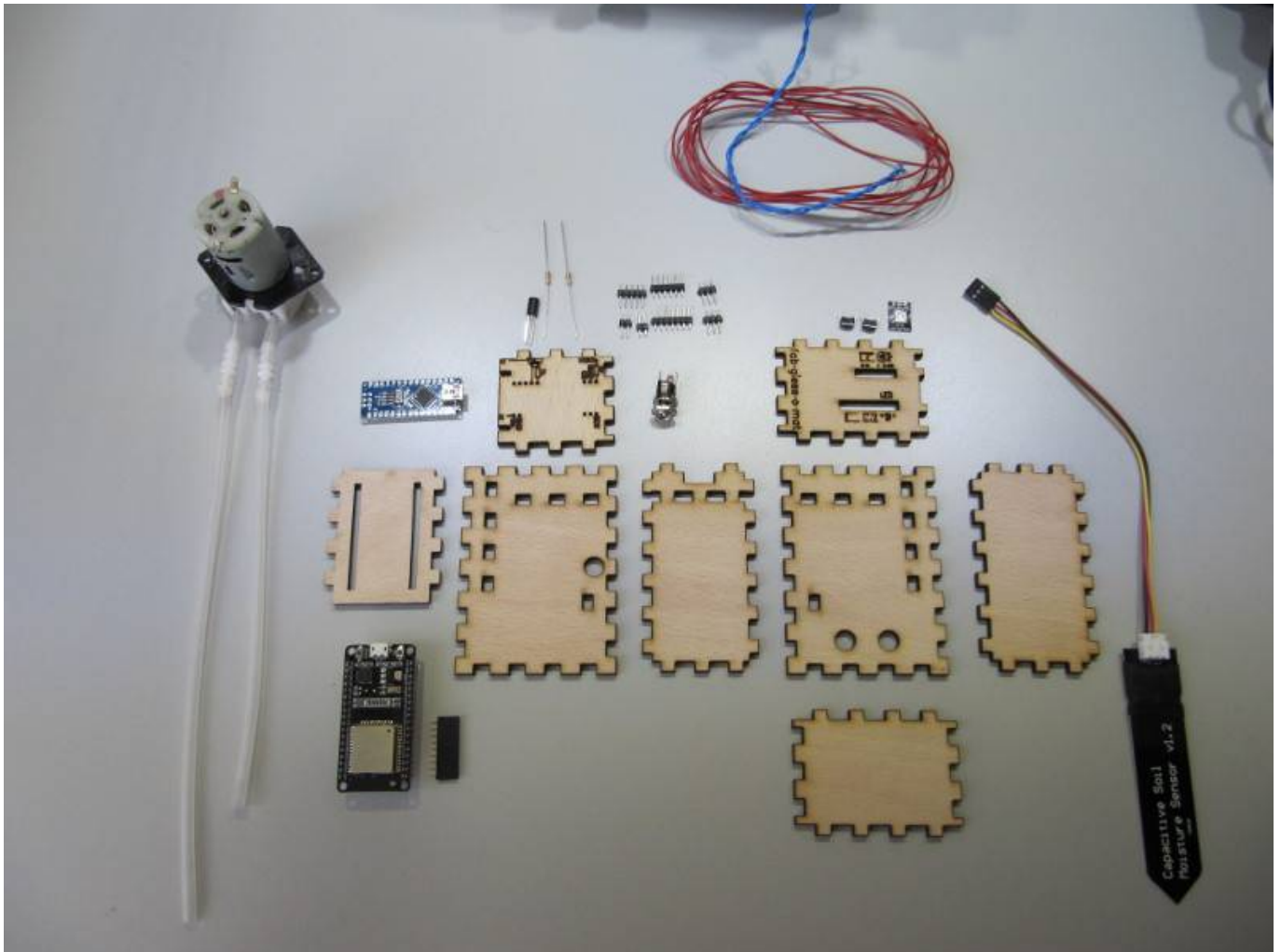
- [1 Schlauchpumpe 12V](#)
- [1 Arduino Nano](#)
- [2 taktile Taster](#)
- [1 RGB LED WS2812](#)
- [1 Transistor NPN, min. 12V, 1A](#)
- [1 Hohlstecker-Buchse 5,5mm 2,1mm](#)
- [1 Netzteil 12V, min. 1A](#)
- [1 Feuchtigkeitssensor](#)
- [Silikonschlauch 5x3mm](#)
- [1 Widerstand 1k](#)
- [Stiftleisten 2,54mm. Stückelung: 1x1, 1x2, 2x3, 1x5, 1x6, 1x7](#)



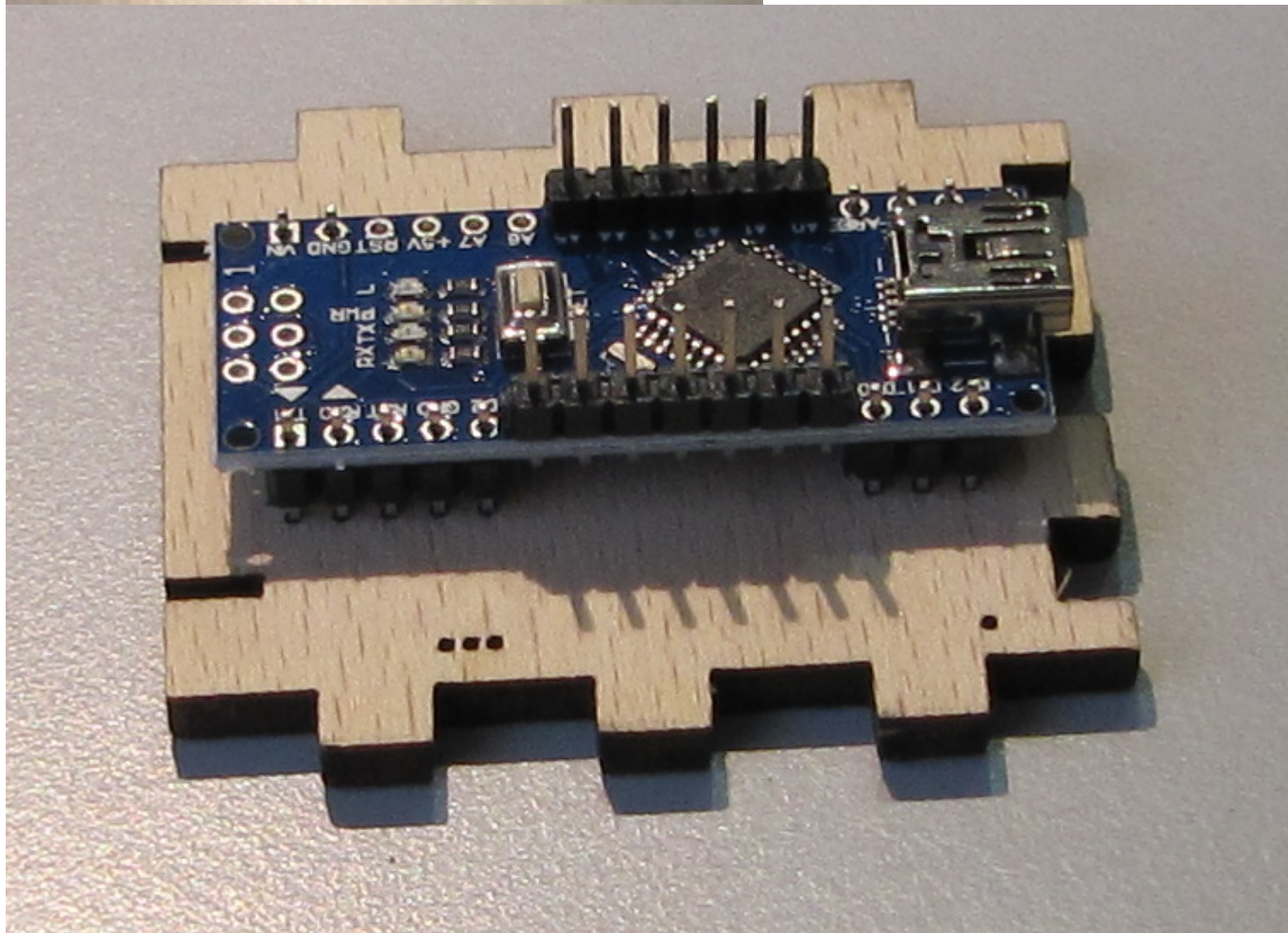
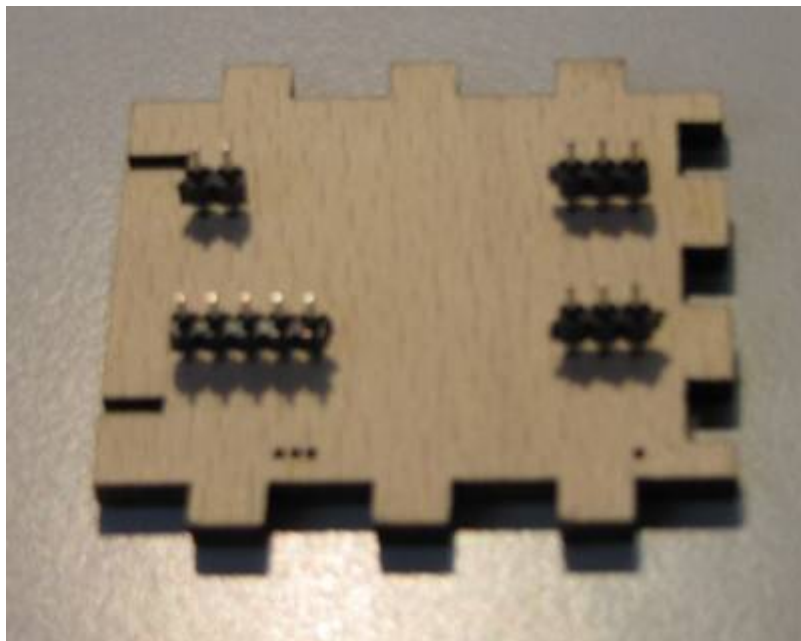
- Draht, 2 Farben
- Schrumpfschlauch 2mm
- 1 Holz oder Kunststoffplatte 4 x 240 x 150 mm ([Lastercutter-Design](#))

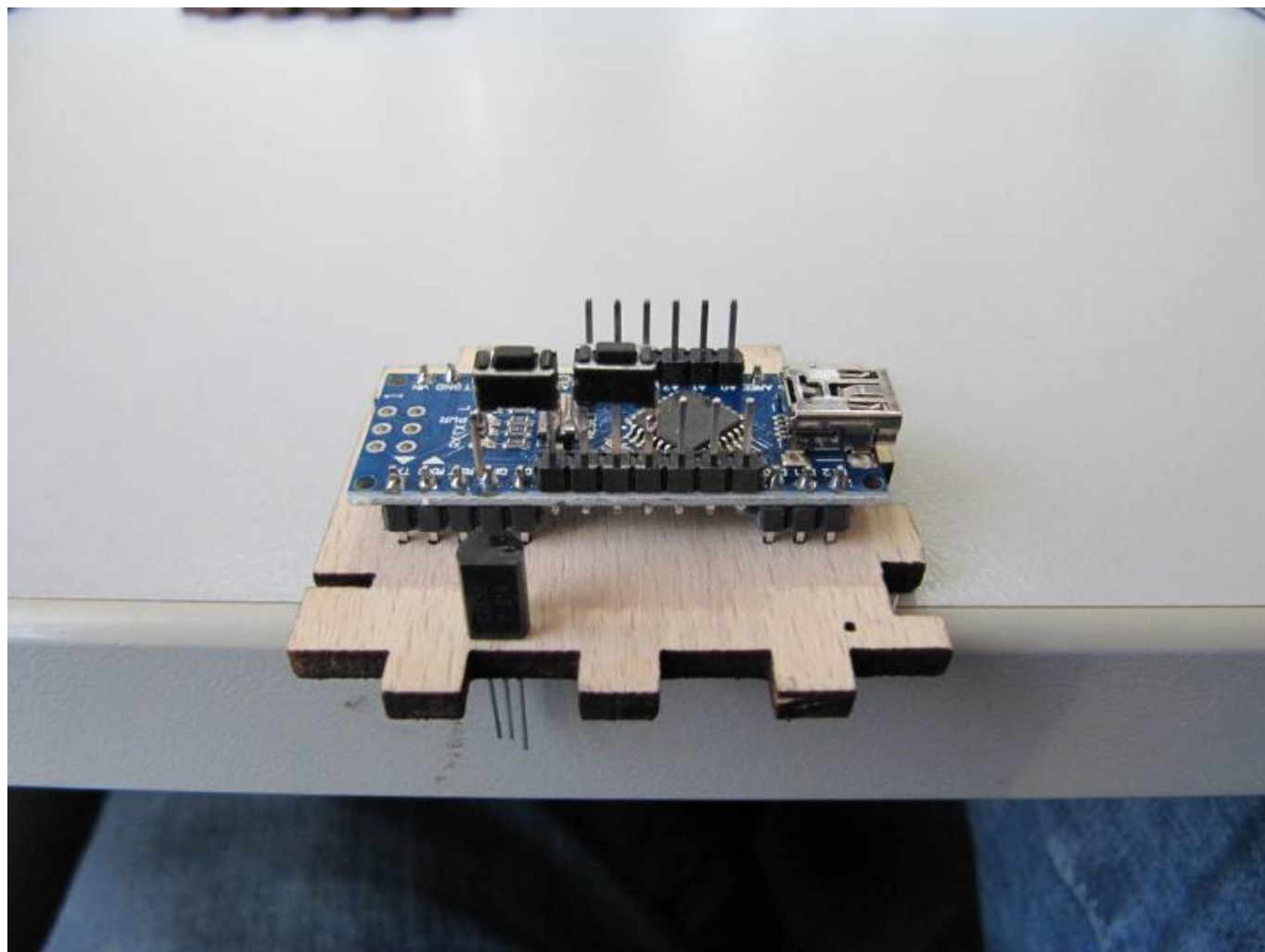
### optionale Wifi-Erweiterung:

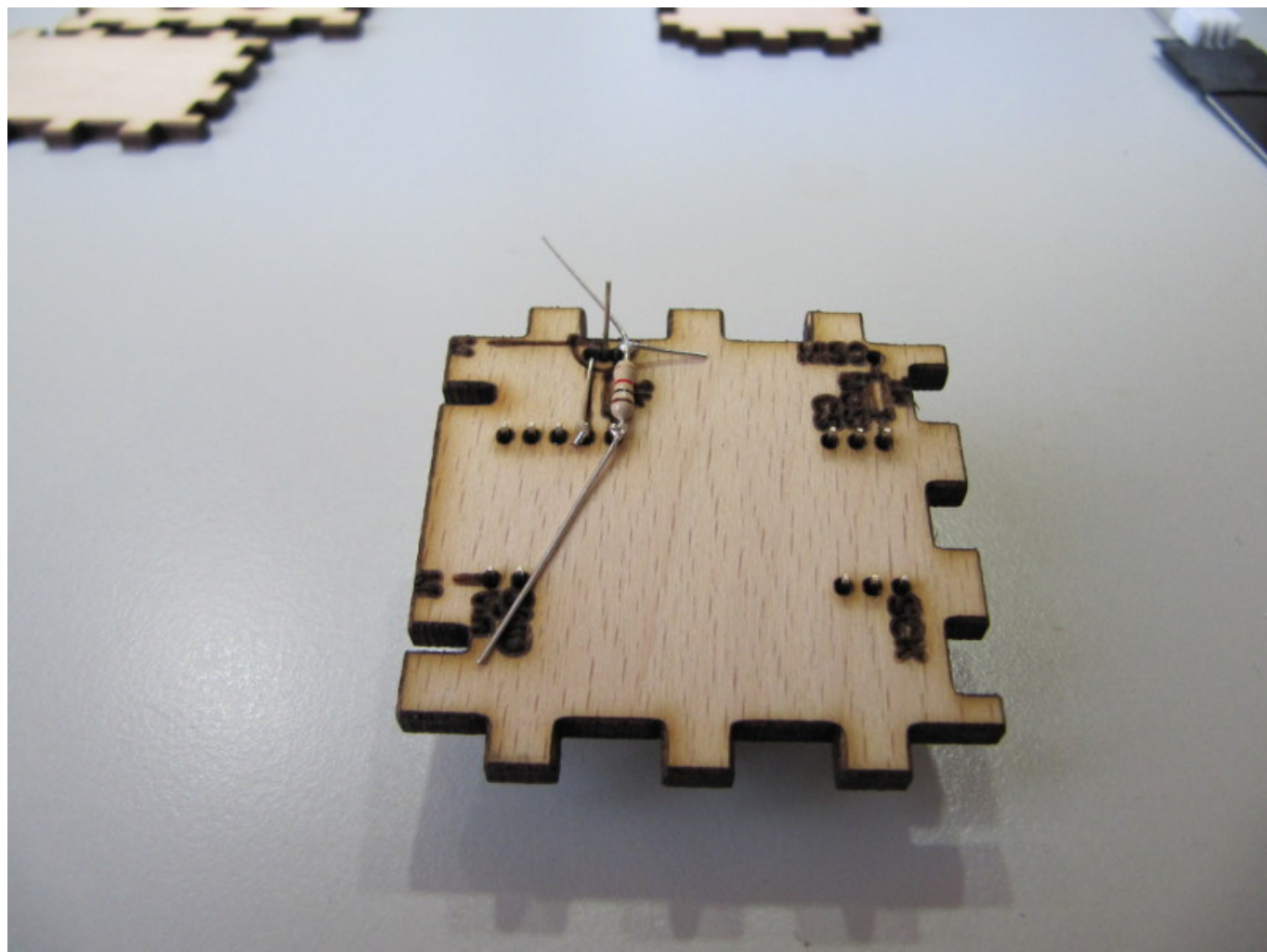
- 1 ESP32 Wifi Modul
- 1 Widerstand 1k
- Buchsenleiste 2,54mm, 1x8



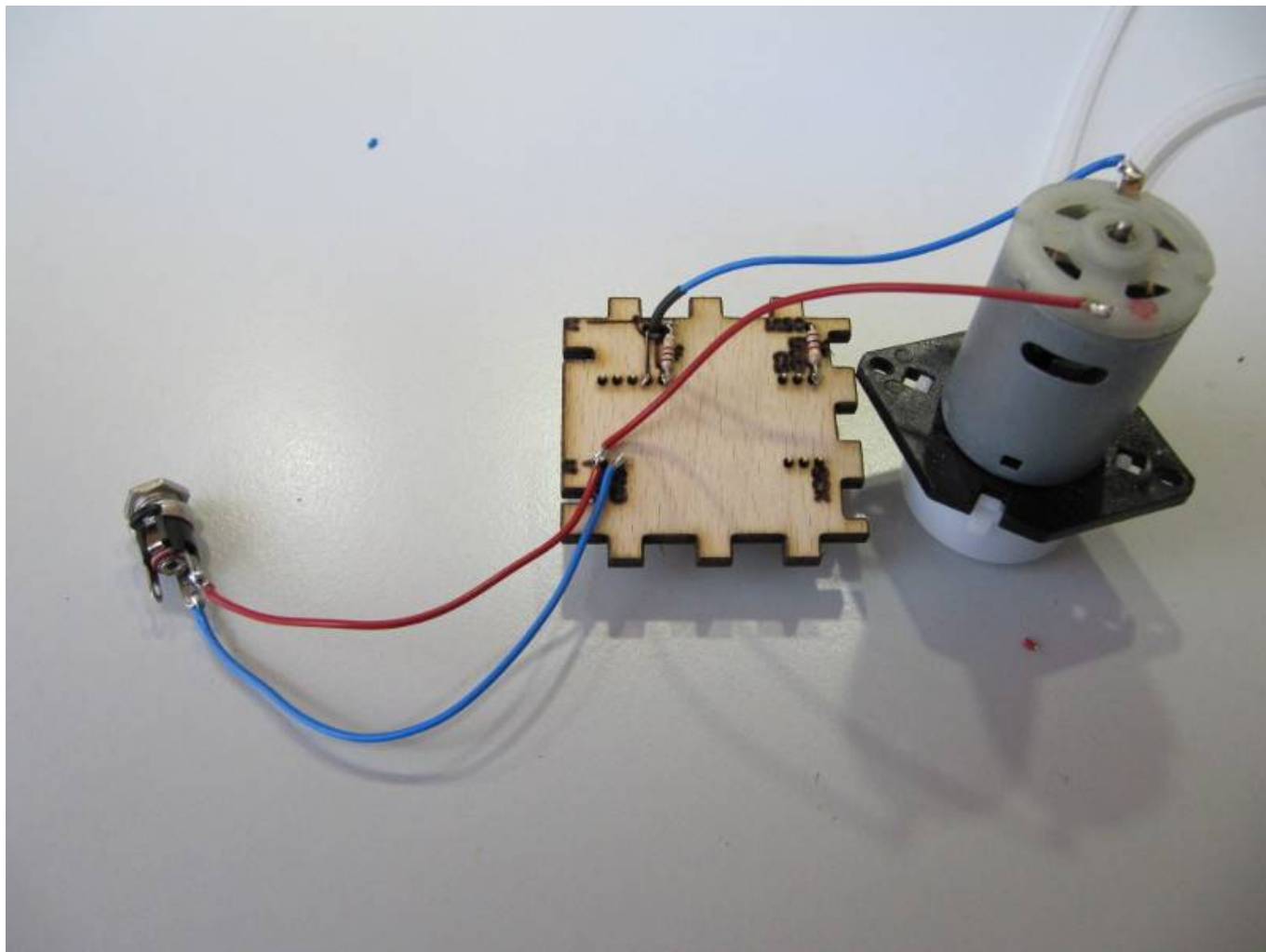
## Bauanleitung





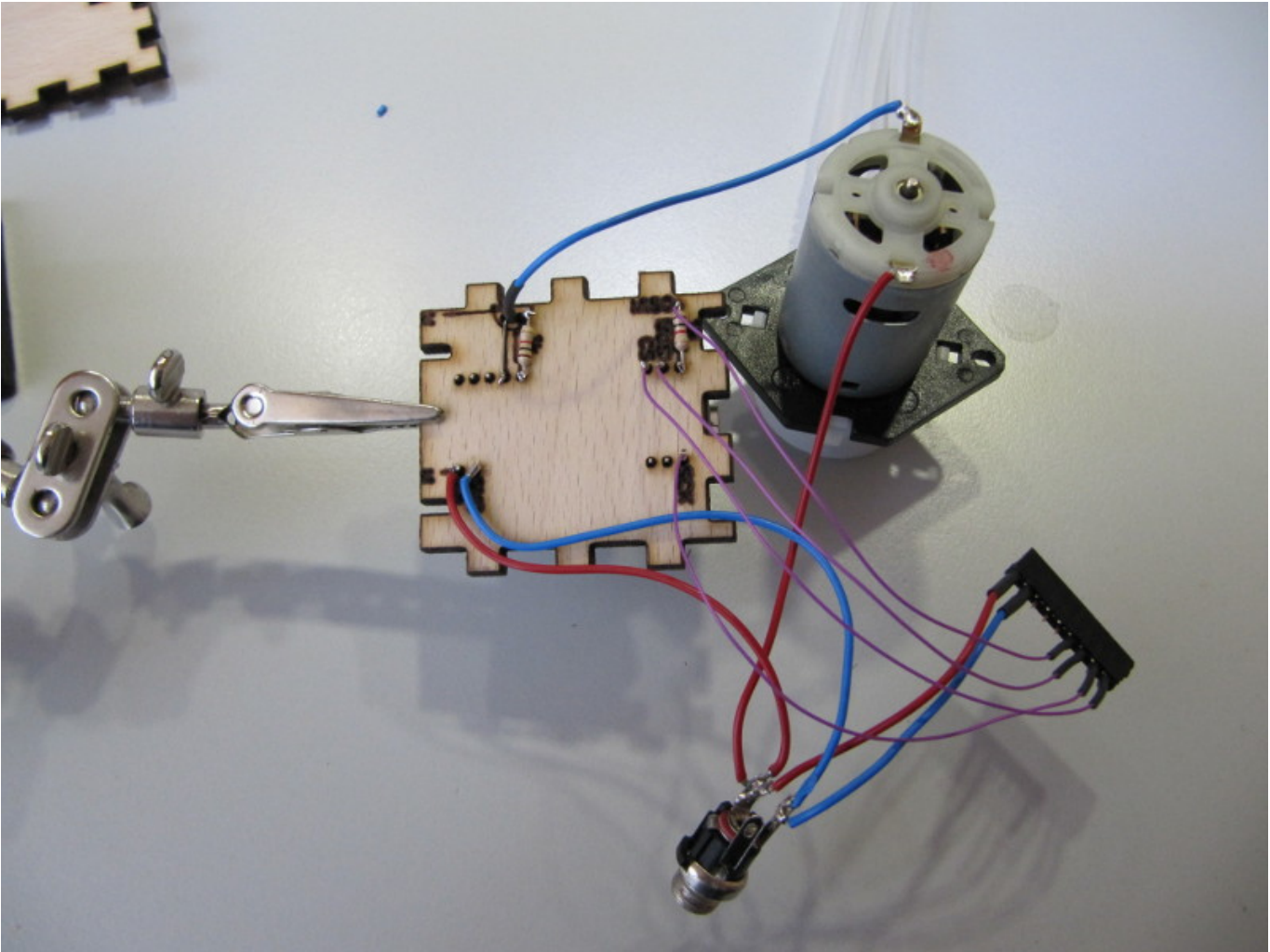




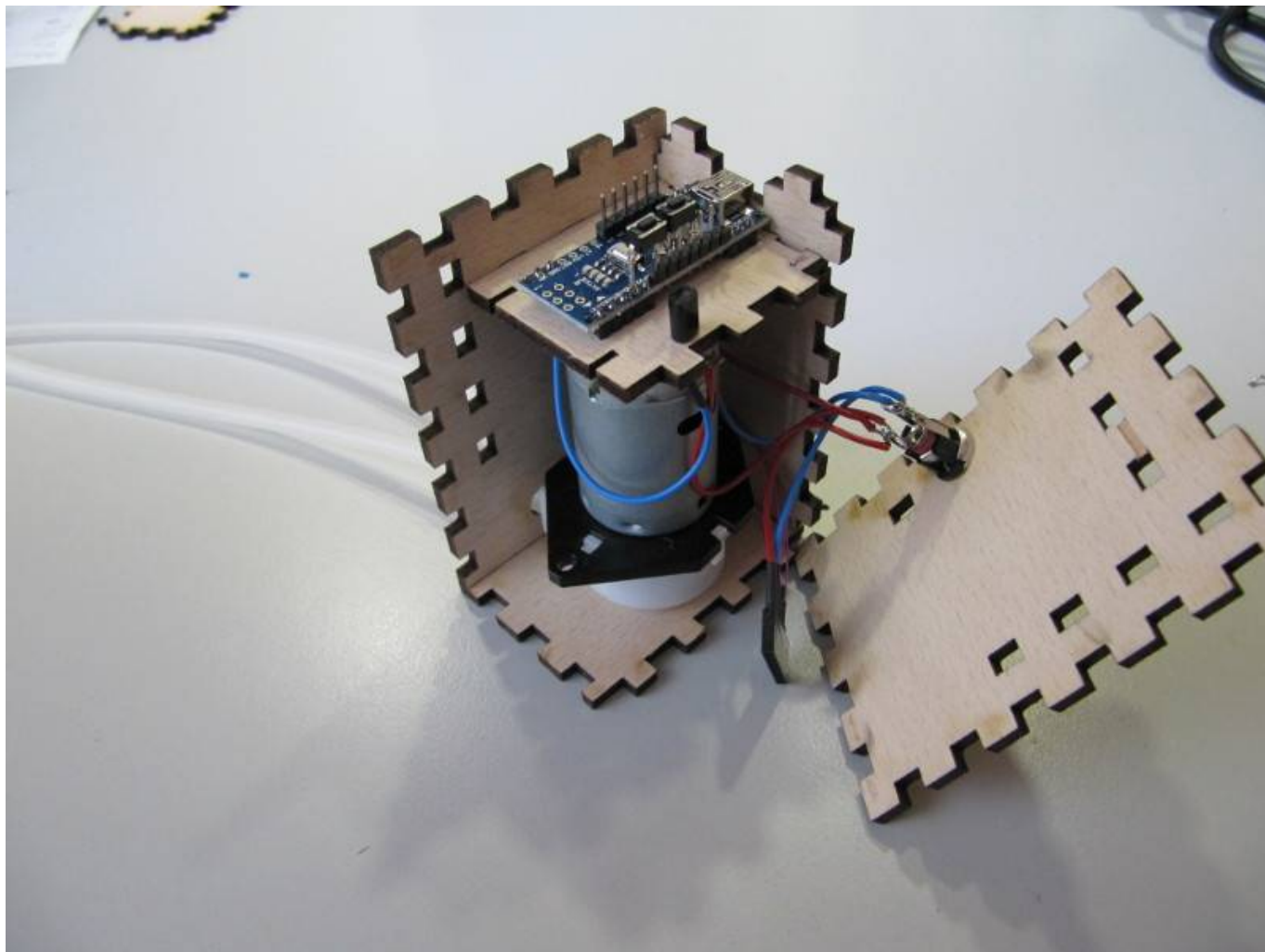


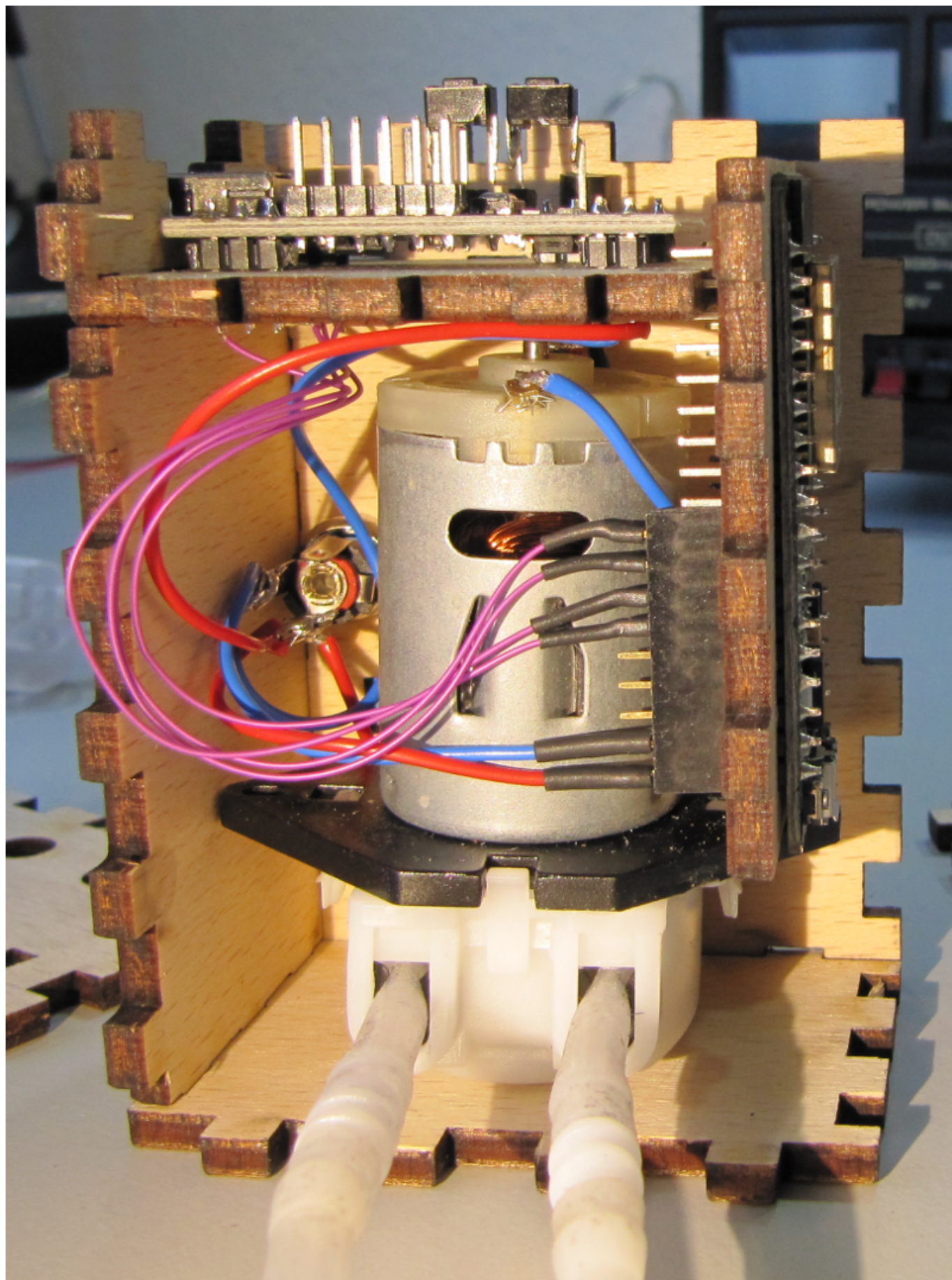
Den Widerstand oben rechts und die Buchsenleiste mit den lila Kabeln dran braucht man für den Anschluß eines Wifi Moduls.

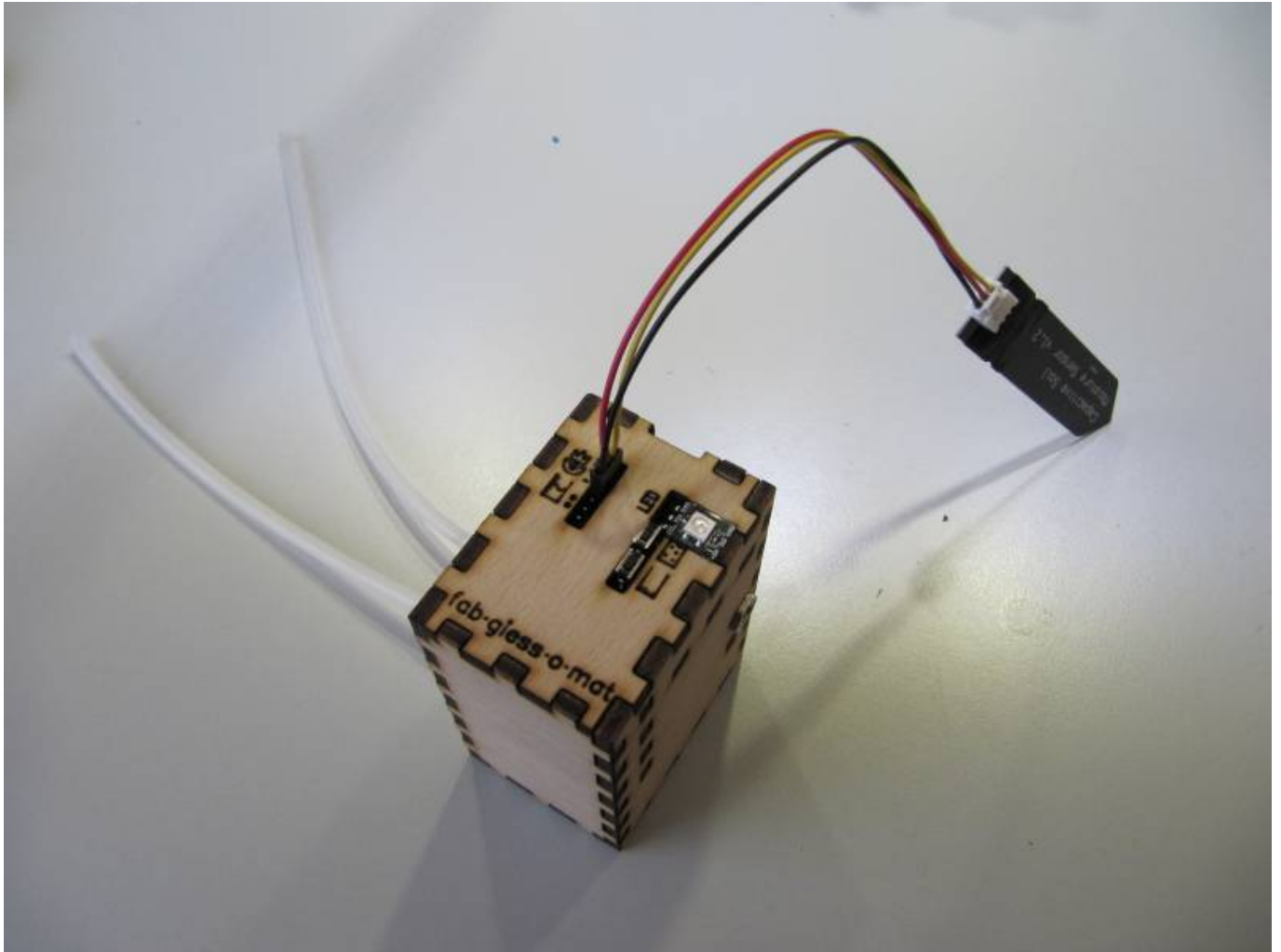
**Achtung:** Es besteht die Gefahr, das das Wifi-Modul in dem Gehäuse den Hitzetot stirbt. Vorerst nicht bestücken !











## Sourcecode

<https://github.com/chbergmann/fab-giess-o-mat>

Das Git Repository enthält 2 Projekte:

**fab-giess-o-mat:** Projekt für den Arduino Mikrokontroller

Kann mit der [Arduino IDE](#) bearbeitet und auf den Arduino geladen werden. Alternativen: [Atom.io](#) oder [Eclipse](#) mit dem [PlatformIO](#) Plugin

**wifi-giess-o-mat:** [PlatformIO](#) Projekt für das ESP Wifi Modul.

## Ansprechpartner





christian.bergmann@mailbox.org

## Links zum Thema

### Ähnliche Projekte

[Giess-o-mat](#)

[WaterMe](#)

[fabfarmer](#)

### Feuchtigkeitssensoren

<http://www.n8chteule.de/zentris-blog/erdfeuchtemessung/sensoren-in-der-erdfeuchtebestimmung/>

[https://wwwvs.cs.hs-rm.de/vs-wiki/index.php/Internet\\_der\\_Dinge\\_WS2015/SmartPlant#Messmethode\\_2:\\_Kapazitiv](https://wwwvs.cs.hs-rm.de/vs-wiki/index.php/Internet_der_Dinge_WS2015/SmartPlant#Messmethode_2:_Kapazitiv)

### Verbindungsstücke, Adapter usw. zum ausdrucken

<https://www.3dponics.com/wiki/download-parts-drip-hydroponics/>

## Diskussion

Anregungen, Kritik, dumme Kommentare, etc. könnt ihr im [Forum](#) loswerden.