

3 Stufen Intervallscheibenwischer im Saporoshez

Im Saporoshez 968 wird der Scheibenwischer über einen Drehschalter im Armaturenbrett eingeschaltet. Der normale Drehschalter hat 3 Stellungen: aus, langsam und schnell.

Dieser Drehschalter wird durch einen Drehschalter mit 6 Stellungen ersetzt. Dadurch sind folgende Stellungen möglich:

1. aus
2. langsam
3. Intervall mit 4 Sekunden kurzer Pause
4. Intervall mit 6 Sekunden mittlerer Pause
5. Intervall mit 8 Sekunden langer Pause
6. schnell

Der Originalknopf wird auf dem neuen Drehschalter mit 6 Stellungen weitergenutzt, so dass die Veränderung im Armaturenbrett erst auf dem zweiten Blick auffällt.

Beim Saporoshez wird der Scheibenwischermotor dauernd über eine Sicherung mit Strom versorgt, geschaltet wird der Scheibenwischermotor über zwei Anschlüsse auf der Minus-Seite (Masse).

Die Schaltung wurde mit einem Arduino Uno gebaut. Auf dem Arduino Uno ist ein so genanntes „Arduino Uno Screw Shield“ gesteckt, mit dem sich Kabel gut und sicher am Arduino Uno anschließen lassen. Genau auf dieses Screw Shield passt wiederum ein Modul mit zwei Relais, die vom Arduino Uno angesteuert werden. Die Stromversorgung des Arduino Uno erfolgt über einen Step-Down-Wandler. Als Gehäuse dient eine günstige Snackbox 11,5x8x5,7cm.

Der Sketch selbst im Arduino Uno ist einfach, er basiert auf einem Vorschlag vom Nutzer combie für das Problem „blink without delay“.

```
//3stufen_intervallscheibenwischer_im_sapo
```

```
//Fuer Arduino UNO Stand 23.12.2022
```

```
//Den Code zum Blinken hat der Nutzer combie vorgestellt, vielen Dank
```

```
const unsigned long onzeit = 1000; // ms Einschaltzeit    Schalterposition 2, 3 und 4  
const unsigned long offzeit1 = 4000; // ms kurze  Ausschaltzeit Schalterposition 2  
const unsigned long offzeit2 = 6000; // ms mittlere Ausschaltzeit Schalterposition 3  
const unsigned long offzeit3 = 8000; // ms lange  Ausschaltzeit Schalterposition 4
```

```
const int relais1 = 2; //Signal zum Ansteuern Relais 1 am Digitalpin 2 Kabel grüngelb  
const int relais2 = 3; //Signal zum Ansteuern Relais 2 am Digitalpin 3 Kabel hellblau
```

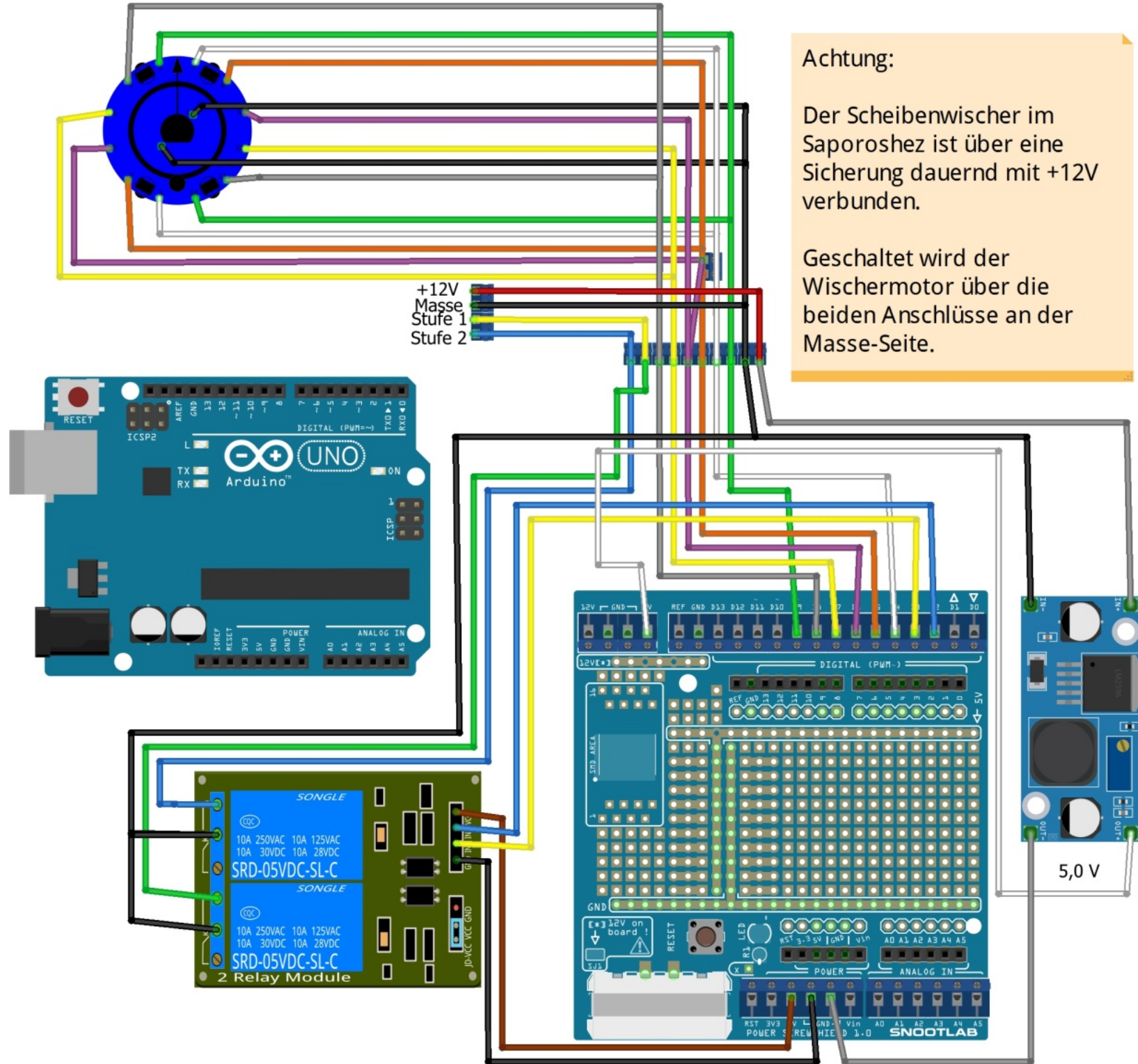
```
int schapo0 = HIGH; //Variable für Schalterposition 0 am Digitalpin 4 Kabel weiß  
int schapo1 = HIGH; //Variable für Schalterposition 1 am Digitalpin 5 Kabel orange  
int schapo2 = HIGH; //Variable für Schalterposition 2 am Digitalpin 6 Kabel lila  
int schapo3 = HIGH; //Variable für Schalterposition 3 am Digitalpin 7 Kabel gelb  
int schapo4 = HIGH; //Variable für Schalterposition 4 am Digitalpin 8 Kabel grau  
int schapo5 = HIGH; //Variable für Schalterposition 5 am Digitalpin 9 Kabel grün
```

```
void setup()
```

```
{  
pinMode(relais1, OUTPUT);  
pinMode(relais2, OUTPUT);  
pinMode(4, INPUT_PULLUP);  
pinMode(5, INPUT_PULLUP);  
pinMode(6, INPUT_PULLUP);  
pinMode(7, INPUT_PULLUP);  
pinMode(8, INPUT_PULLUP);  
pinMode(9, INPUT_PULLUP);  
digitalWrite(relais1, HIGH); // beim Einschalten der Autobatterie erst einmal alles ausschalten  
digitalWrite(relais2, HIGH); // beim Einschalten der Autobatterie erst einmal alles ausschalten  
int schapo0 = HIGH; //Variable für Schalterposition 0 am Digitalpin 4 Kabel weiß  
int schapo1 = HIGH; //Variable für Schalterposition 1 am Digitalpin 5 Kabel orange  
int schapo2 = HIGH; //Variable für Schalterposition 2 am Digitalpin 6 Kabel lila  
int schapo3 = HIGH; //Variable für Schalterposition 3 am Digitalpin 7 Kabel gelb  
int schapo4 = HIGH; //Variable für Schalterposition 4 am Digitalpin 8 Kabel grau  
int schapo5 = HIGH; //Variable für Schalterposition 5 am Digitalpin 9 Kabel grün  
}
```

```
void loop()
```

```
{  
schapo0 = digitalRead(4);  
schapo1 = digitalRead(5);  
schapo2 = digitalRead(6);  
schapo3 = digitalRead(7);  
schapo4 = digitalRead(8);  
schapo5 = digitalRead(9);  
if (schapo0 == LOW)  
{  
digitalWrite(relais1, LOW);  
digitalWrite(relais2, LOW);  
}  
if (schapo1 == LOW)  
{  
digitalWrite(relais1, HIGH);  
digitalWrite(relais2, HIGH);  
}  
if (schapo2 == LOW)  
{  
digitalWrite(relais1, (millis()%(onzeit+offzeit1))>offzeit1); //Vom Nutzer Combie vorgestellt  
digitalWrite(relais2, (millis()%(onzeit+offzeit1))>offzeit1);  
}  
if (schapo3 == LOW)  
{  
digitalWrite(relais1, (millis()%(onzeit+offzeit2))>offzeit2);  
digitalWrite(relais2, (millis()%(onzeit+offzeit2))>offzeit2);  
}  
if (schapo4 == LOW)  
{  
digitalWrite(relais1, (millis()%(onzeit+offzeit3))>offzeit3);  
digitalWrite(relais2, (millis()%(onzeit+offzeit3))>offzeit3);  
}  
if (schapo5 == LOW)  
{  
digitalWrite(relais1, HIGH);  
digitalWrite(relais2, LOW);  
}  
delay(30);}  
}
```

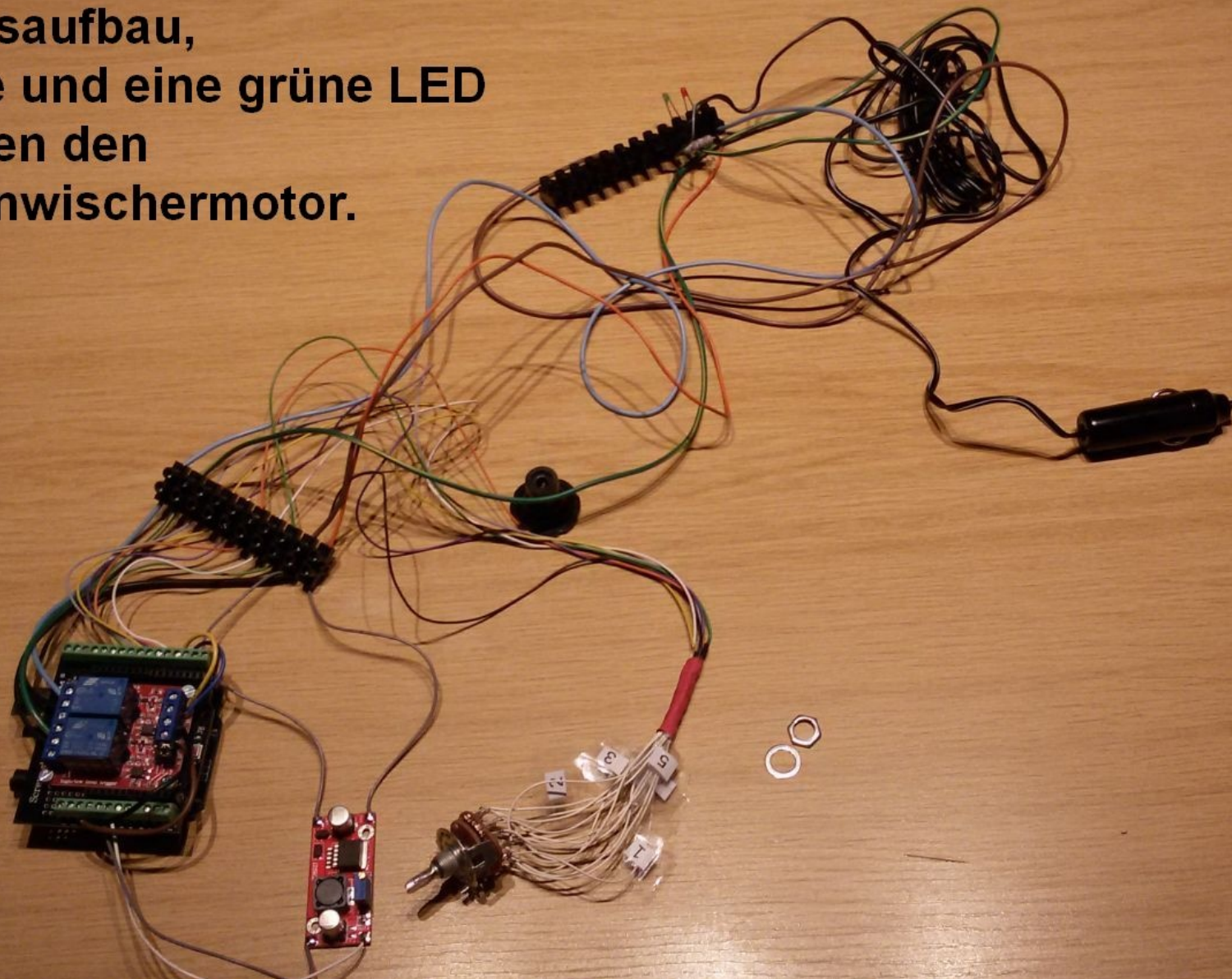


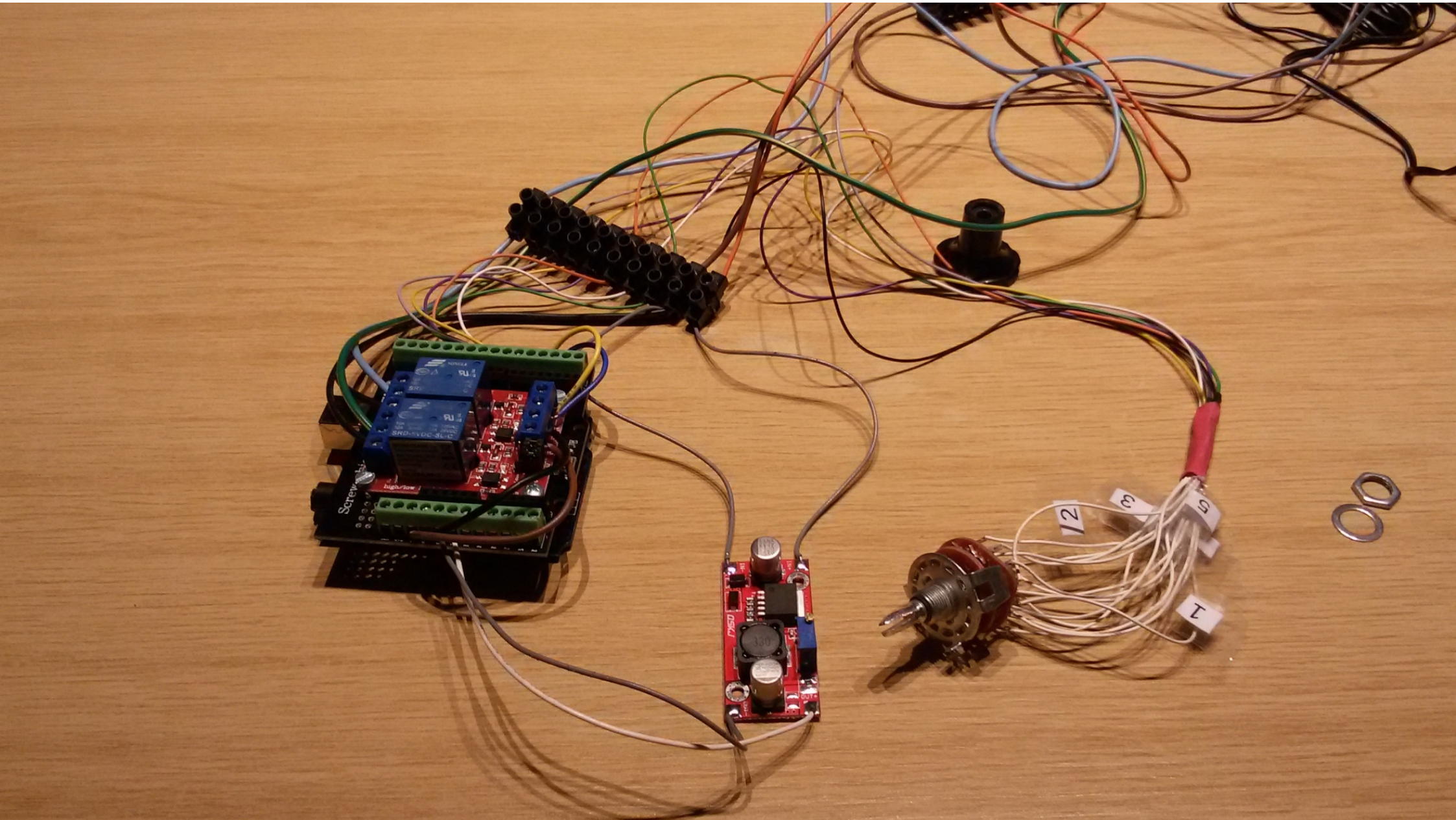
Achtung:

Der Scheibenwischer im Saporoshez ist über eine Sicherung dauernd mit +12V verbunden.

Geschaltet wird der Wischermotor über die beiden Anschlüsse an der Masse-Seite.

**Versuchsaufbau,
eine rote und eine grüne LED
simulieren den
Scheibenwischermotor.**





A Raspberry Pi 4 board with a blue relay module mounted on top. The relay module has two relays and is labeled "Relay" and "12VDC 3A". It is connected to a green terminal block. The board is also connected to a black terminal block with many wires.

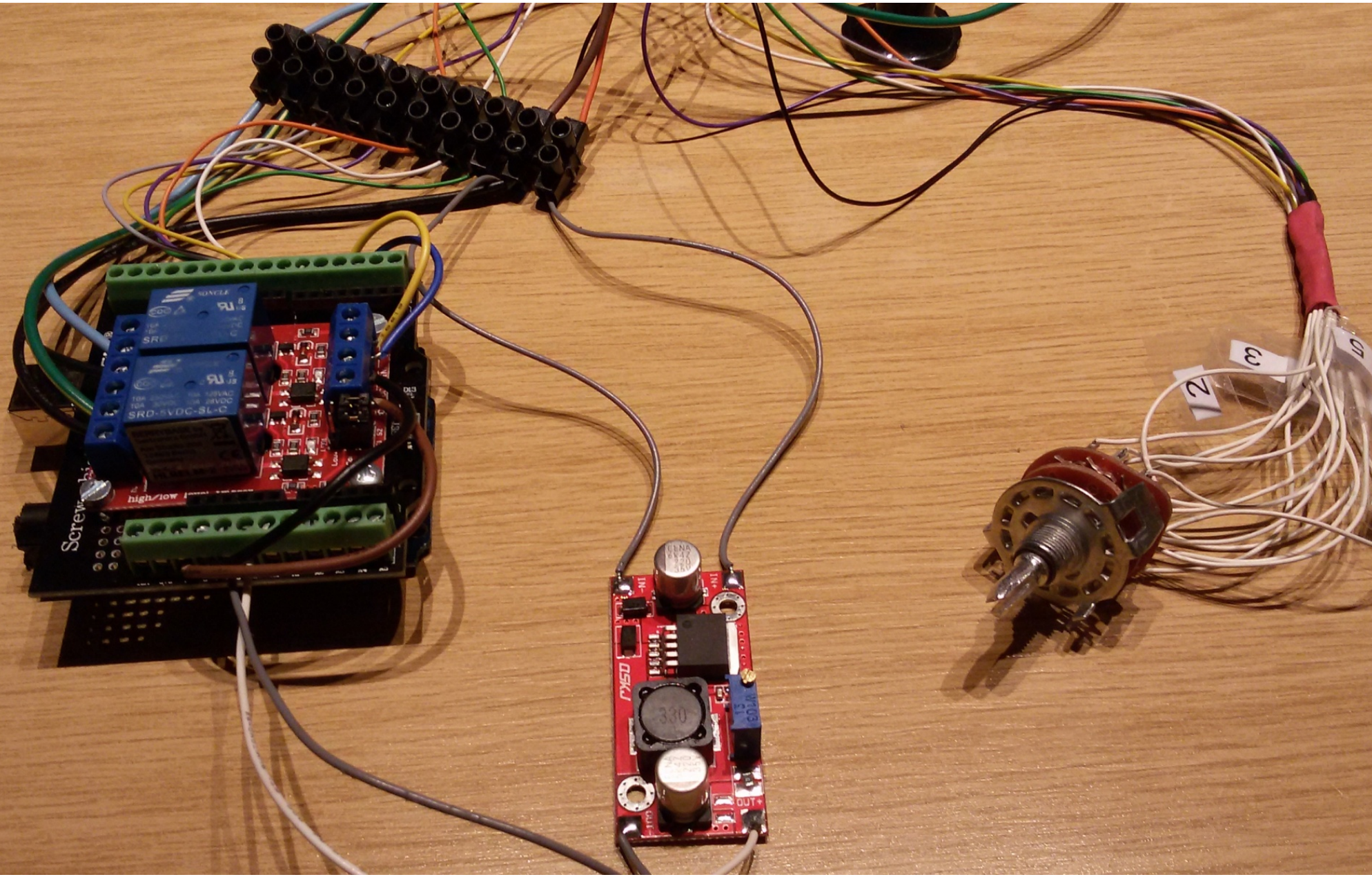
A red DC-DC converter module with two electrolytic capacitors. It is labeled "1.5A" and "5V". It is connected to a black terminal block.

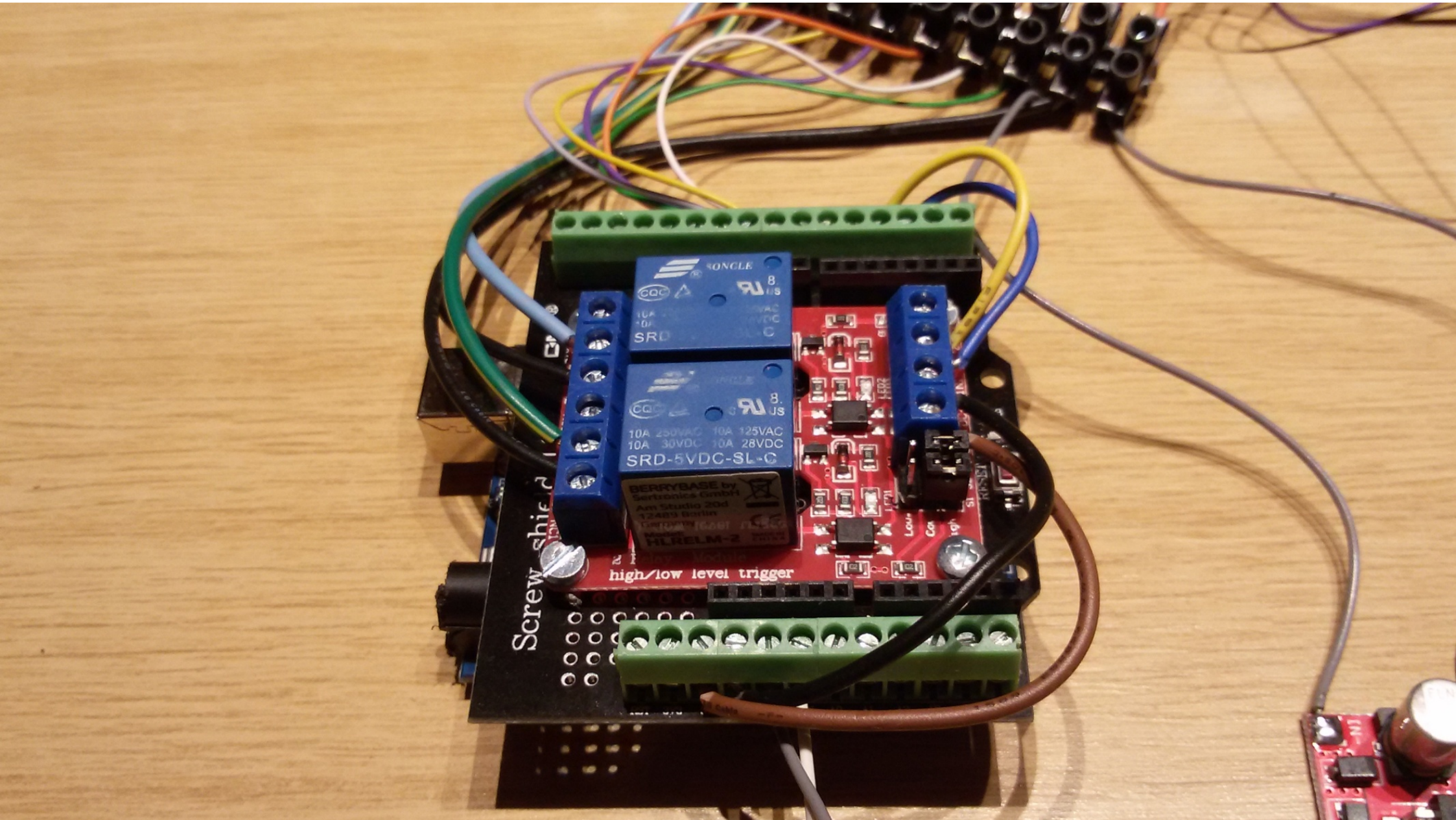
A stepper motor with a metal shaft and a red plastic housing. It is connected to a bundle of white wires.

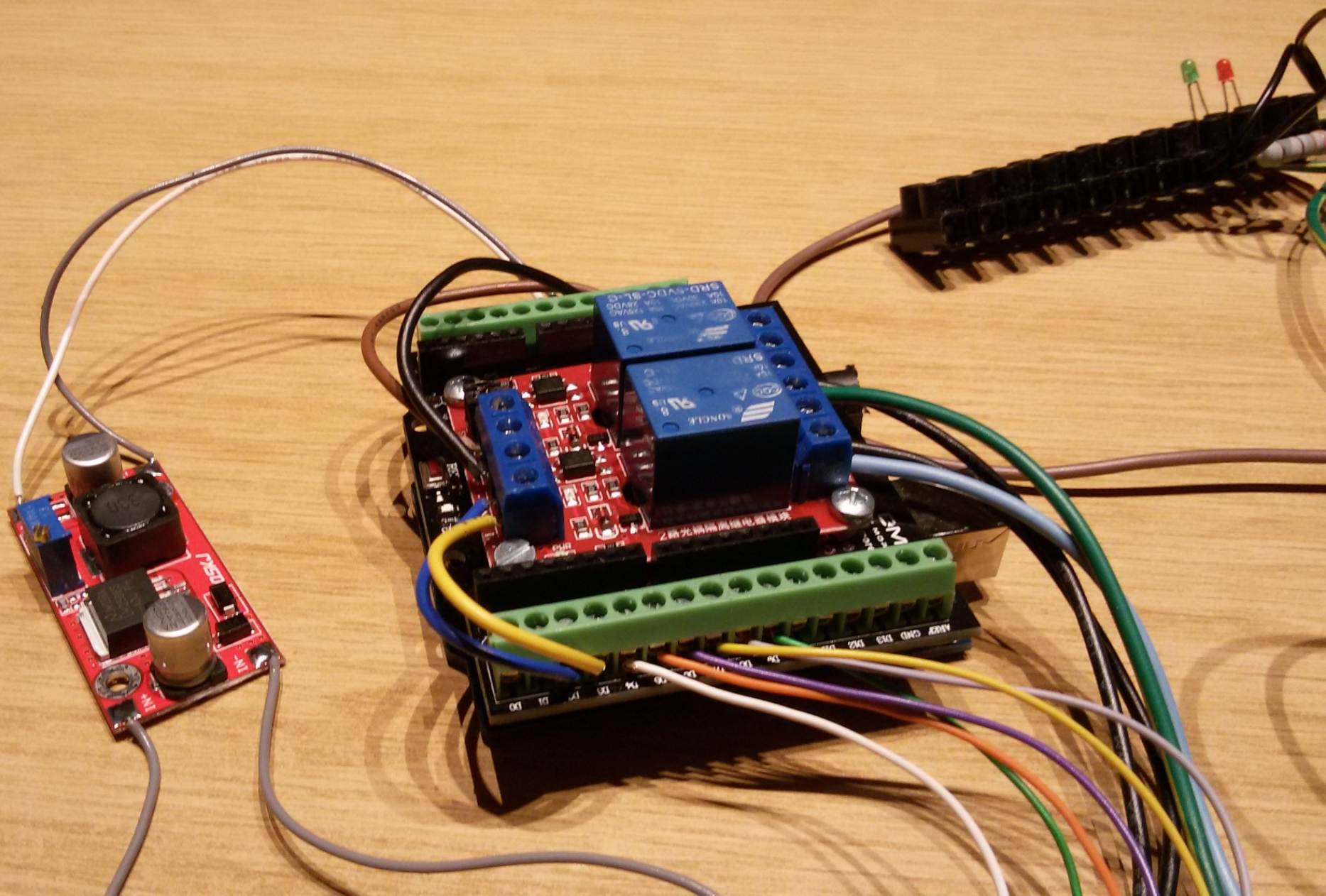
12
3
5
1



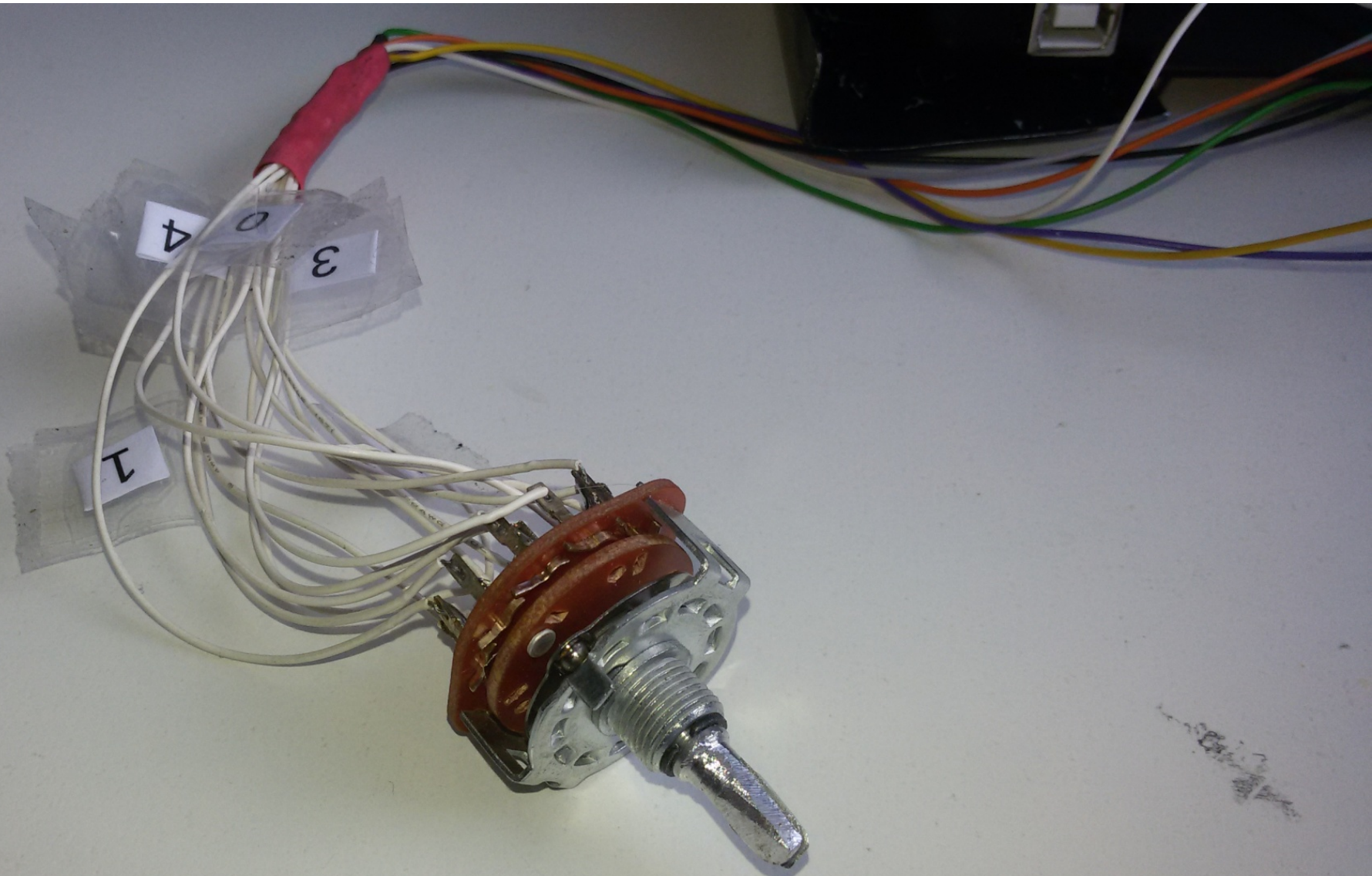


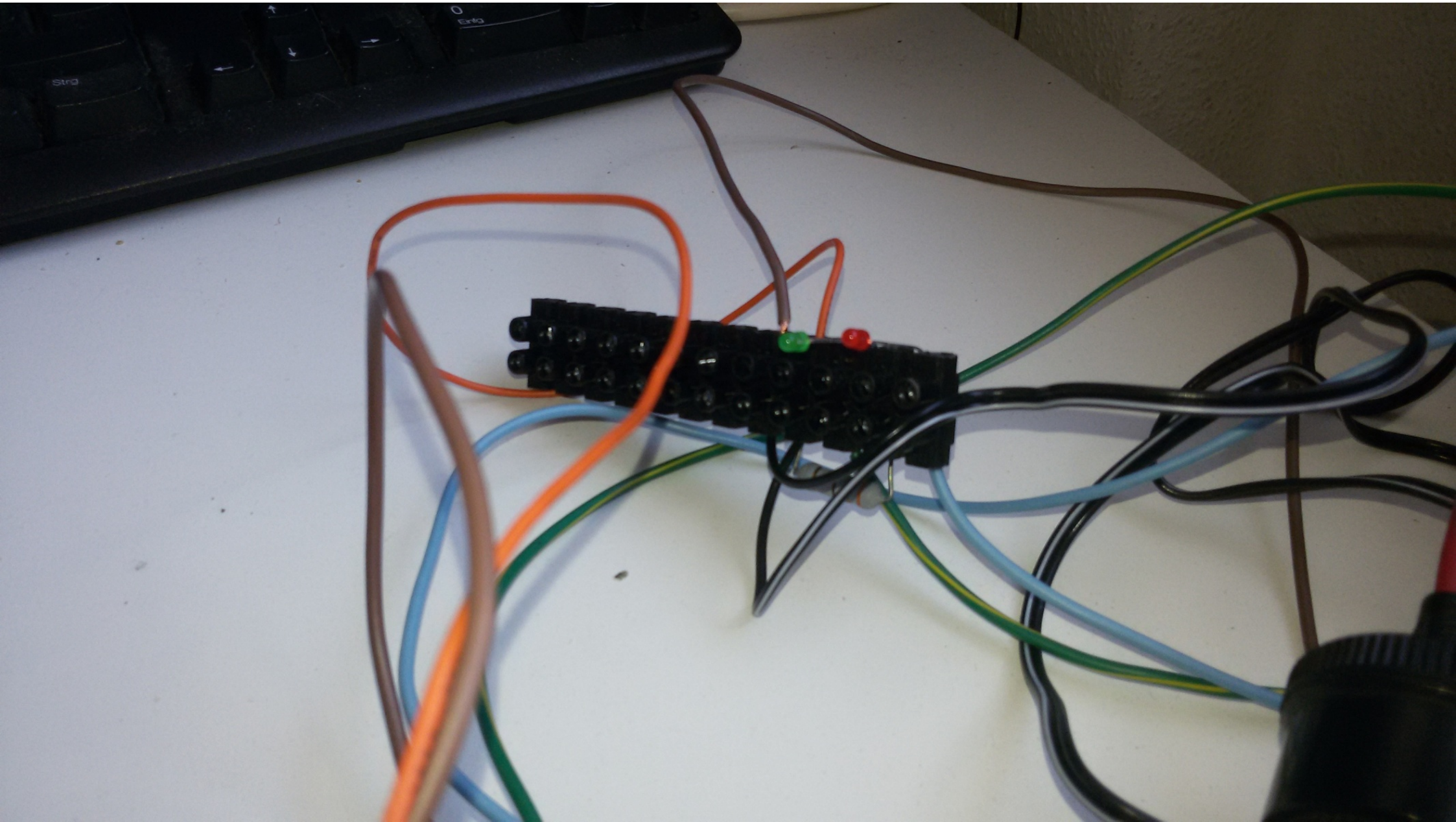


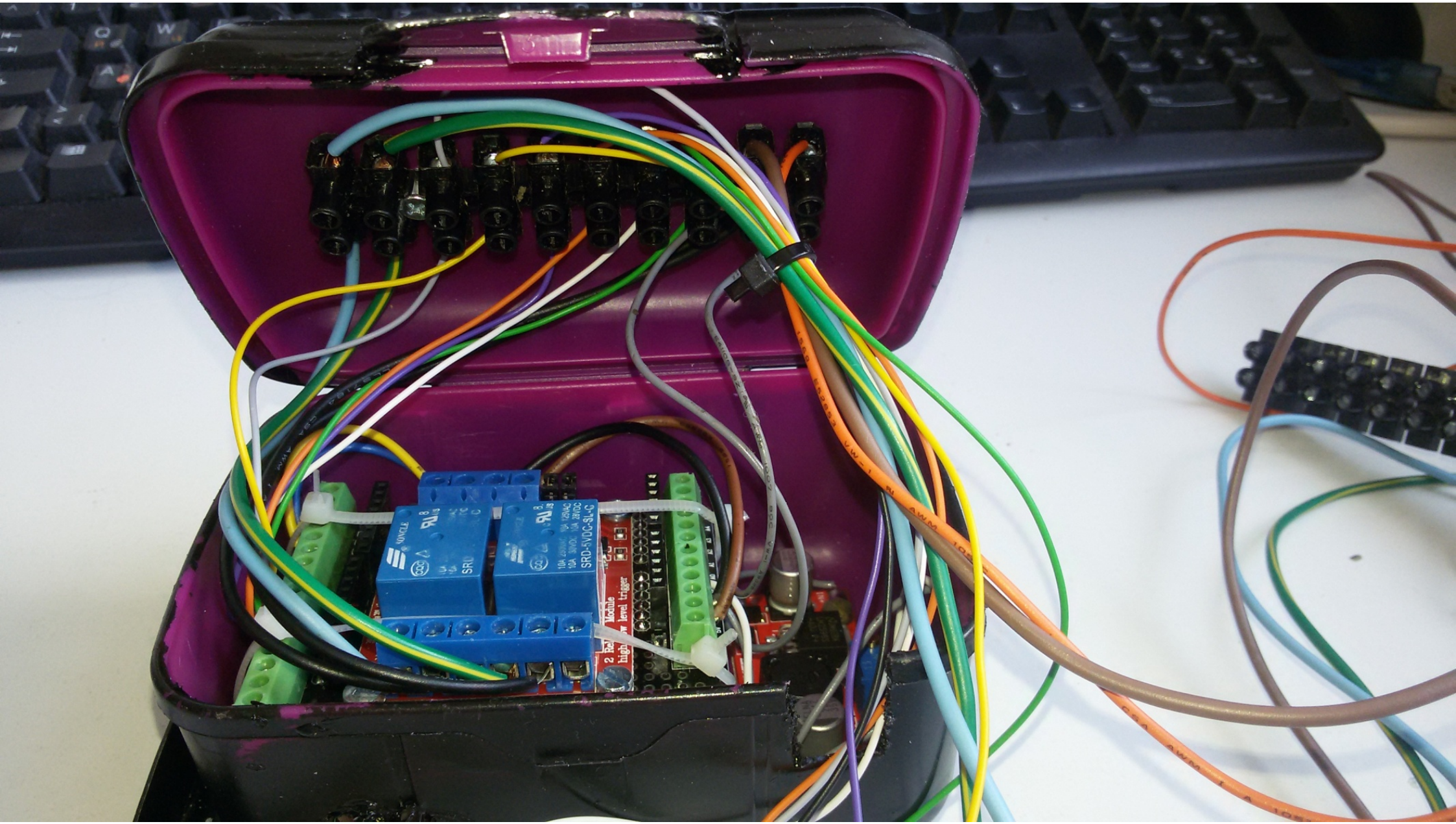


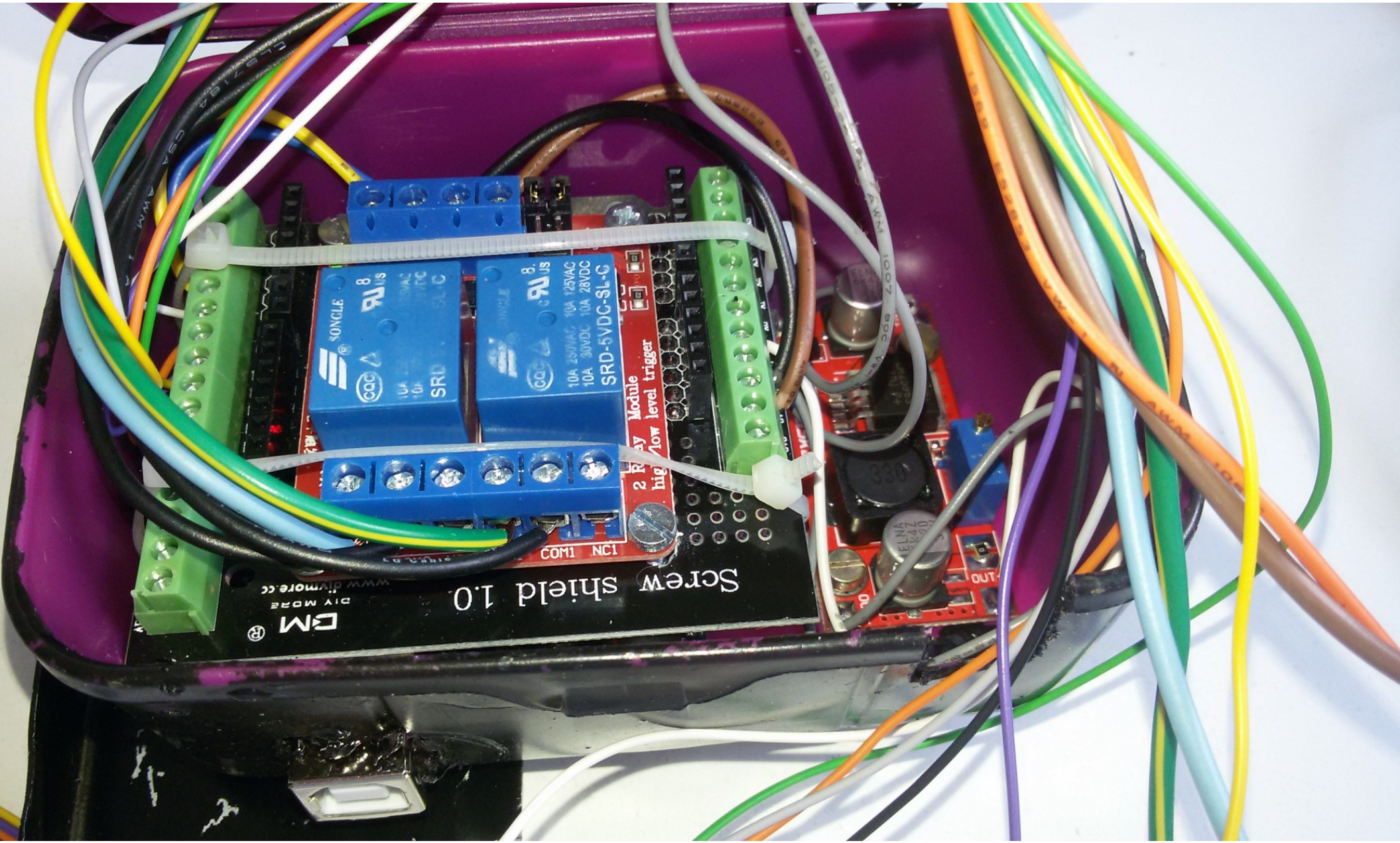












DM[®] Screw shield 1.0
DIY MORE
www.diy-more.cc

8
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
SRD
SIL-C
CQC

3
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
SRD
SIL-C
CQC

2 Relay Module
high/low level trigger

COM1 NC1

OUT

