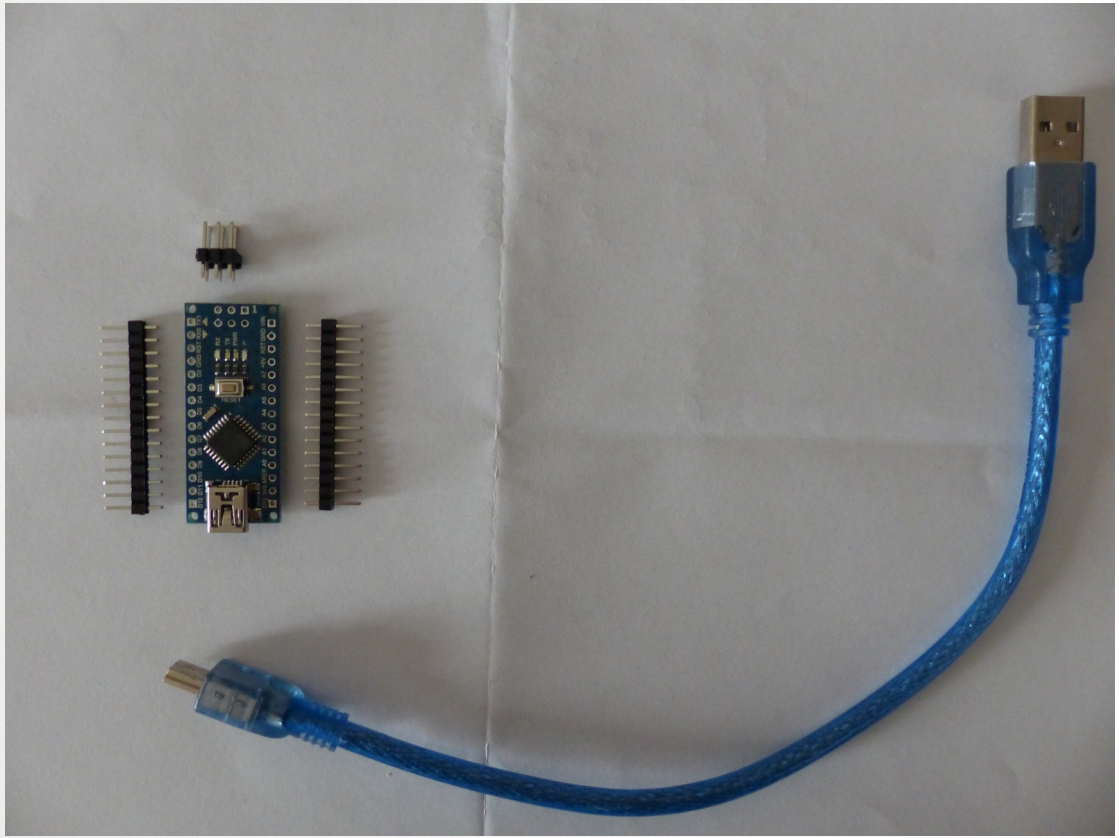
Arduino UNO

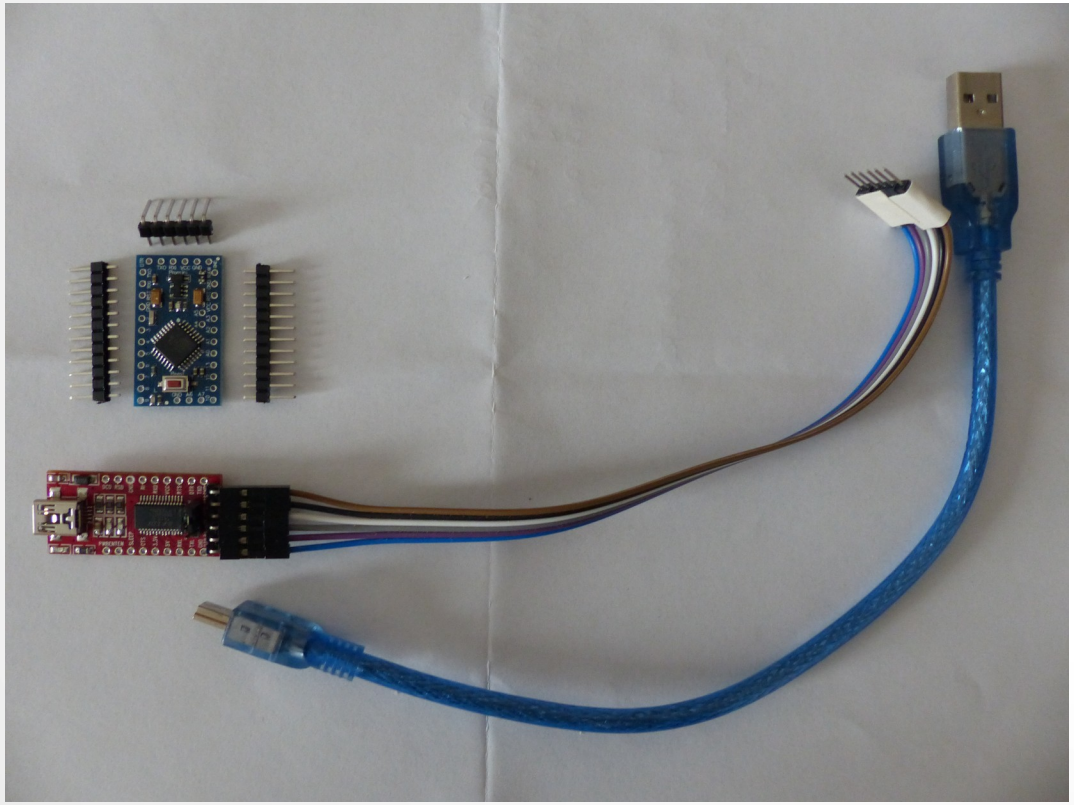
- Für unsere Zwecke:
  zu groß, zu schwer, zu teuer

Arduino Nano



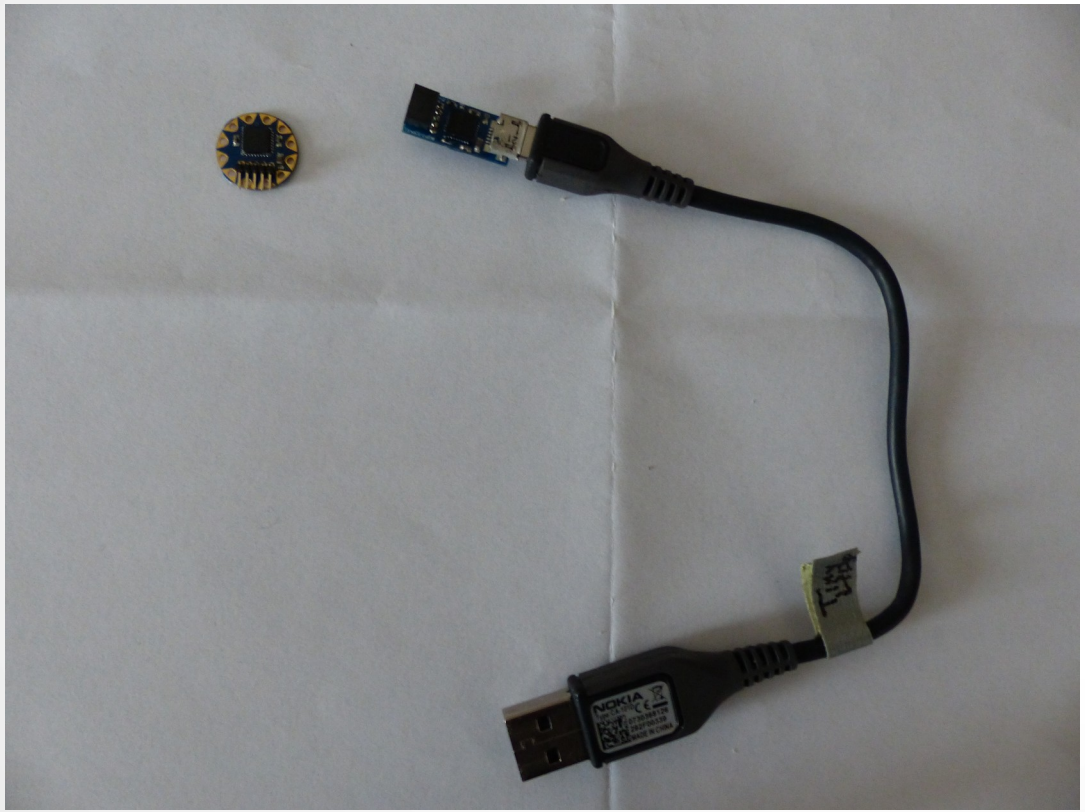
- Einfach, damit zu testen, da direkt am PC anschließ-bar
- 32 KB Programmspeicher

Arduino ProMini



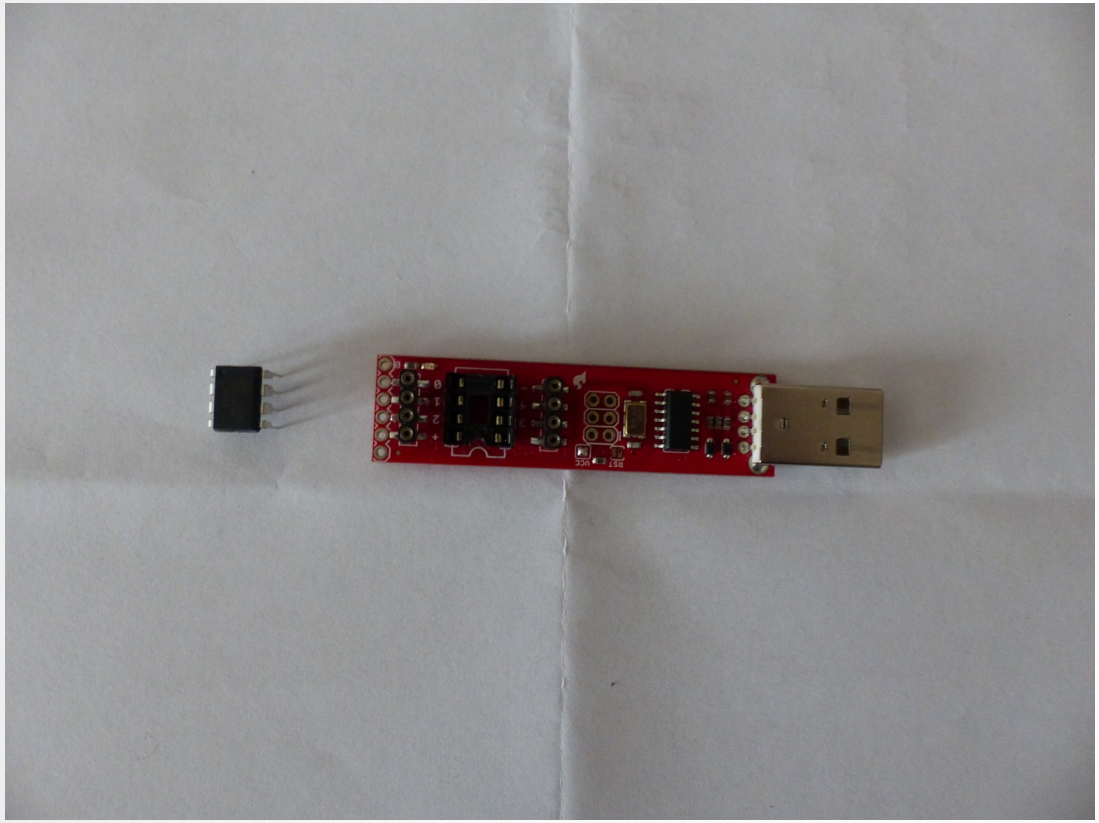
- Kein unnötiger Ballast wegen Wegfall der USB-Schnittstelle
- 32 KB Programmspeicher
- Preiswert (e-bay)
- Programmierung über USB-Adapter
- Achtung: 3,3V- und 5V-Version!

Tiny Lily Mini



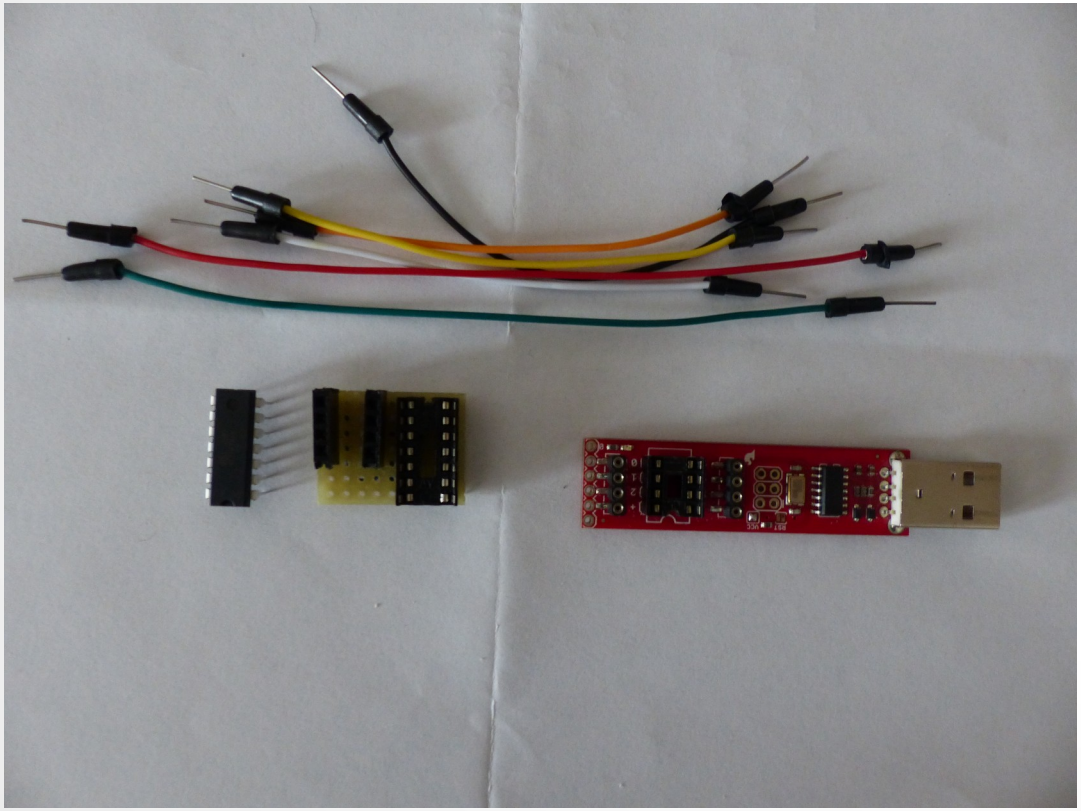
- Prozessor ATMega328P 3,3 V, 8 MHz (Watterott)
- 32 KB Programmspeicher
- Programmierung mit speziellem Adapter
- ca. 10 €
- etwas Fingerspitzengefühl zum Einlöten nötig

ATtiny85



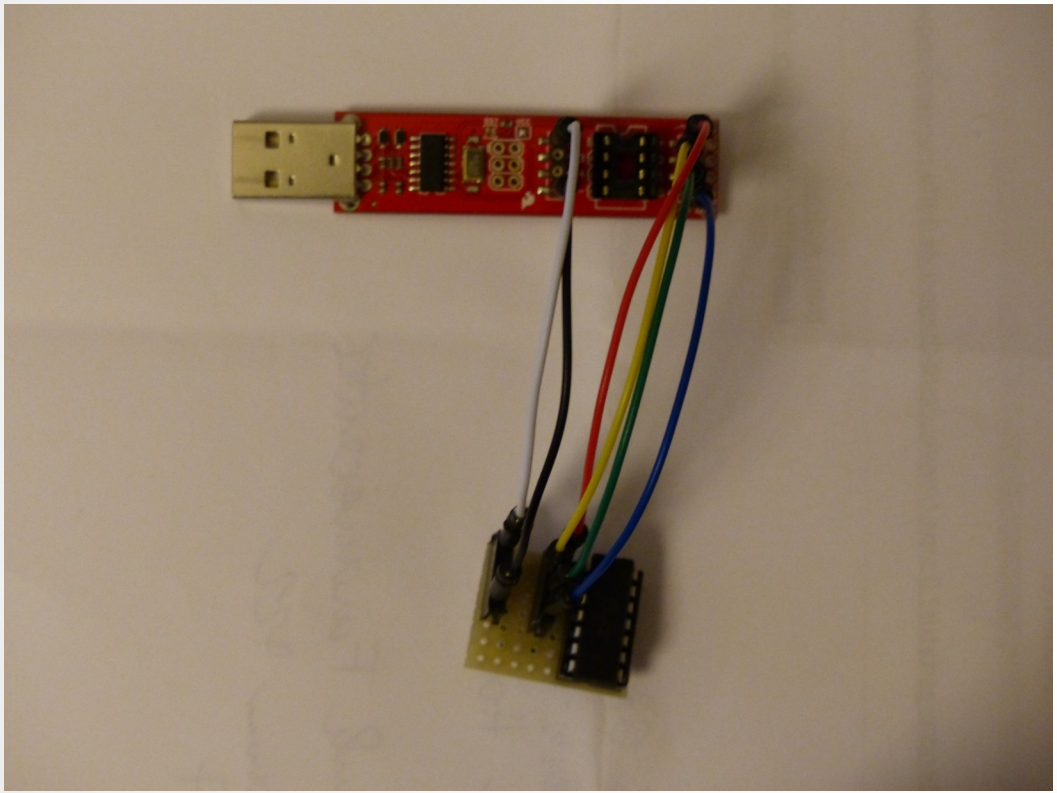
- 2,7 V .. 5,5 V
- Nur 8 KB Programmspeicher!
- 5 freie Pins
- Programmierbar
  mit Tiny USB Adapter von Sparkfun
  (z. B. Watterott)
- keine In-Circuit-Programmierung:
  Prozessor aus Sockel nehmen
- 8-bit-Timer-Libs (Servo, Serial, ...)
- Preiswert: ca. 1 €

ATtiny84



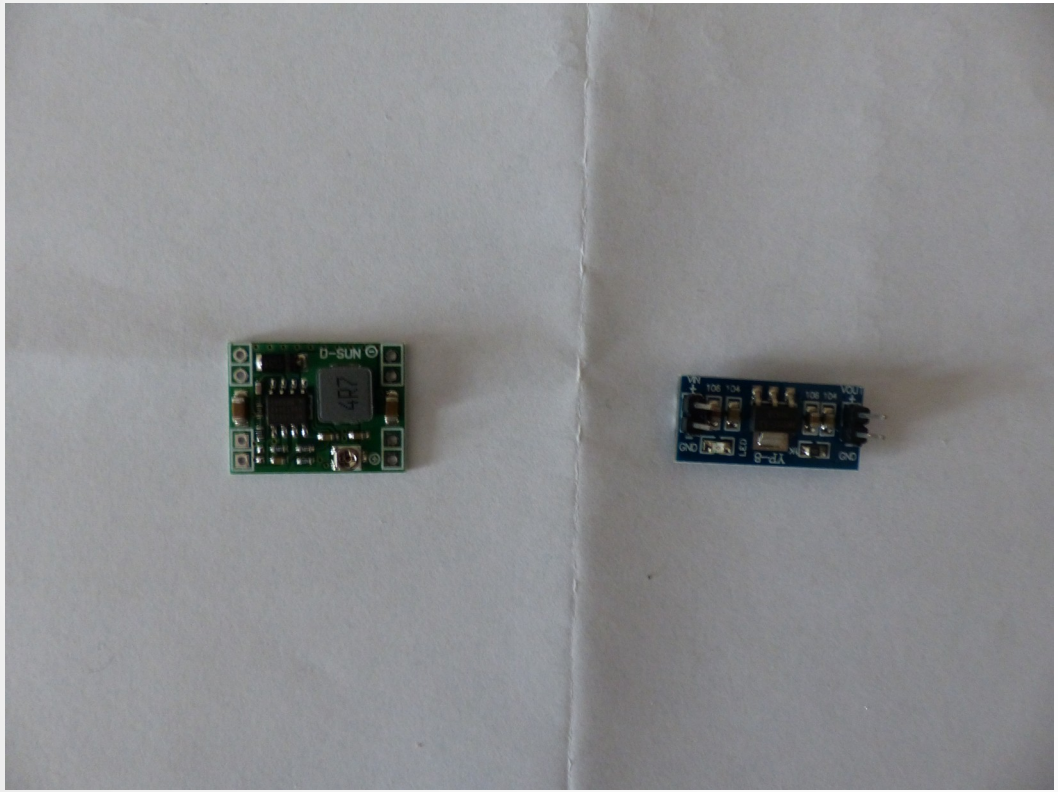
- 2,7 .. 5,5 V
- Nur 8 KB Programmspeicher!
- 11 freie Pins
- Programmierbar
  mit Tiny USB Adapter (Sparkfun)
  und proprietärem Adapter
  ATtiny85 → ATtiny84
- keine In-Circuit-Programmierung:
  Prozessor aus Sockel nehmen
- 8-bit-Timer-Libs (Servo, Serial, ...)
- ca. 1 € bis 3 €

ATtiny84 Programmieren



- Verbinden
  - VCC
  - GND
  - Reset
  - SCK
  - MISO
  - MOSI
- Auswählen: USBtinyISP
- Zuerst: Bootloader programmieren!

Wandler



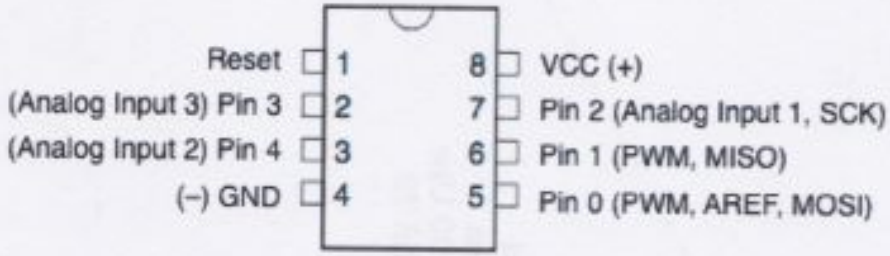
Manchmal sinnvoll:

- links: Stepdown-Wandler
  z. B. auf 5 V (e-bay)
- Rechts: Wandler 5 V auf 3,3 V
  (e-bay)

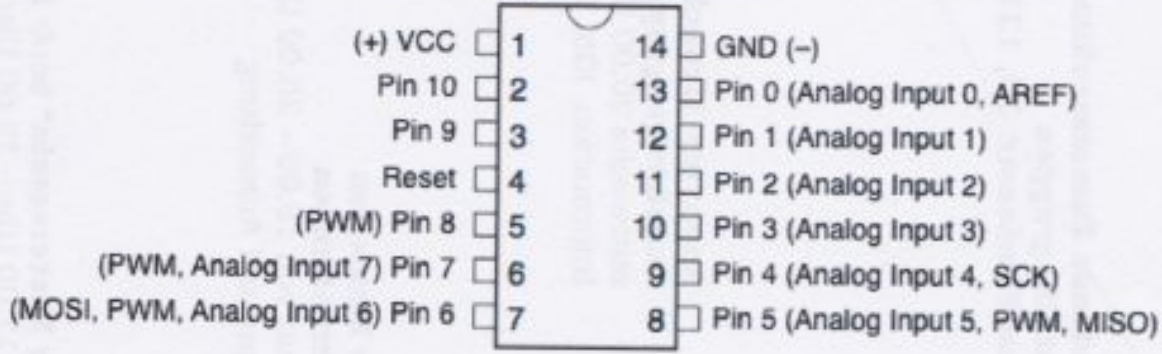
ATtinyx5 / x4 Vergleich

ATtiny Microcontroller Pin-Outs

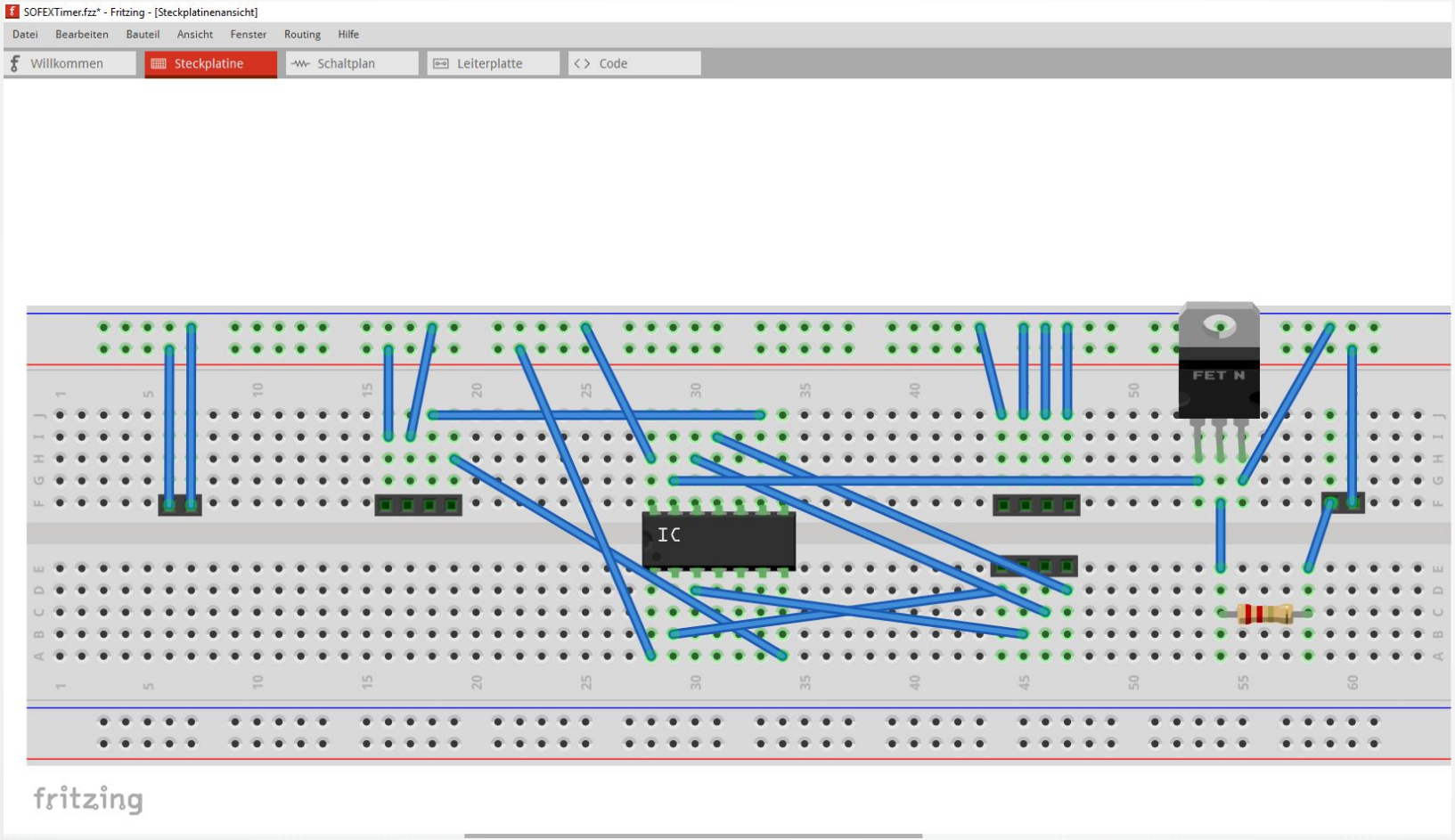
ATtiny45 / ATtiny85



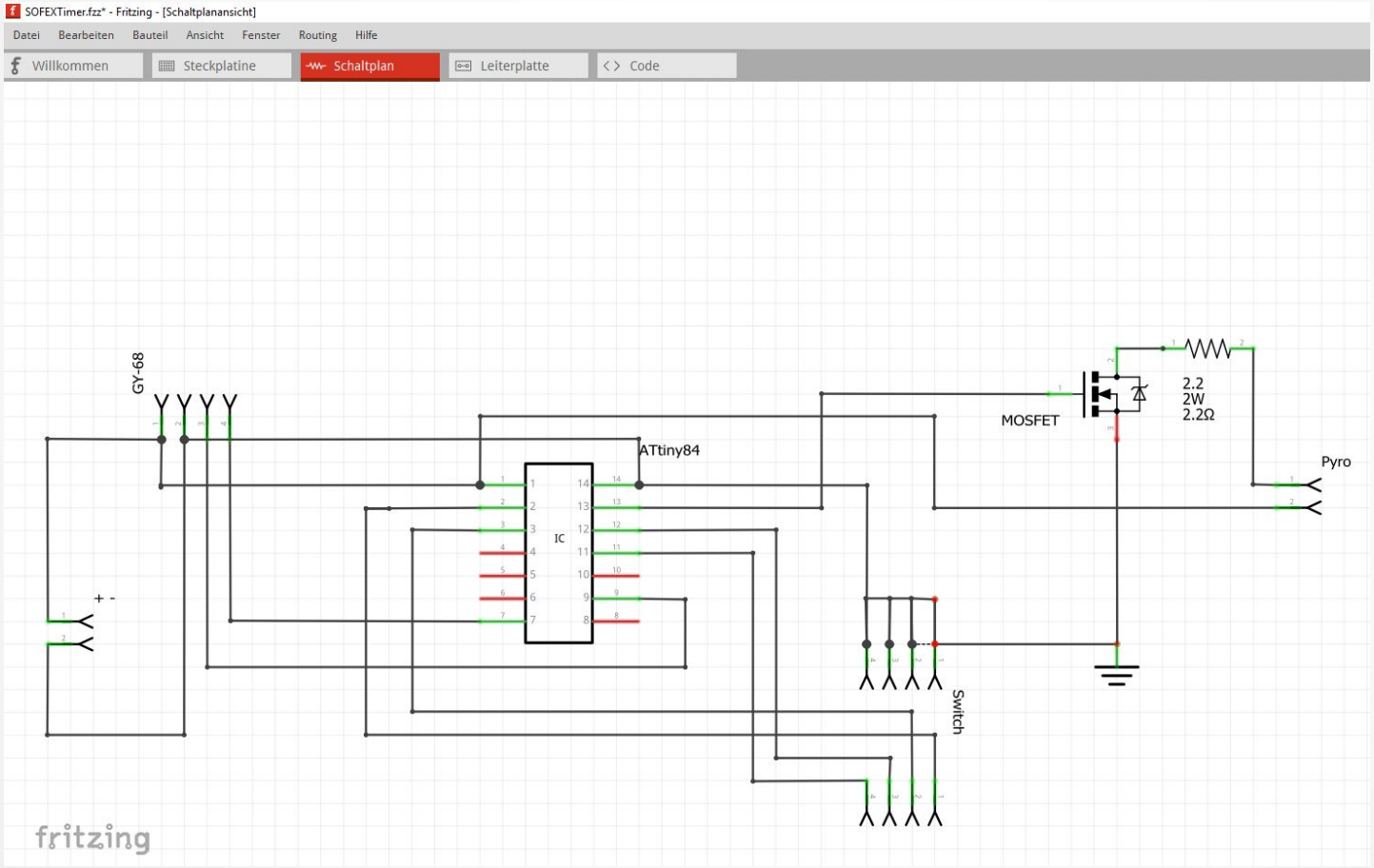
ATtiny44 / ATtiny84



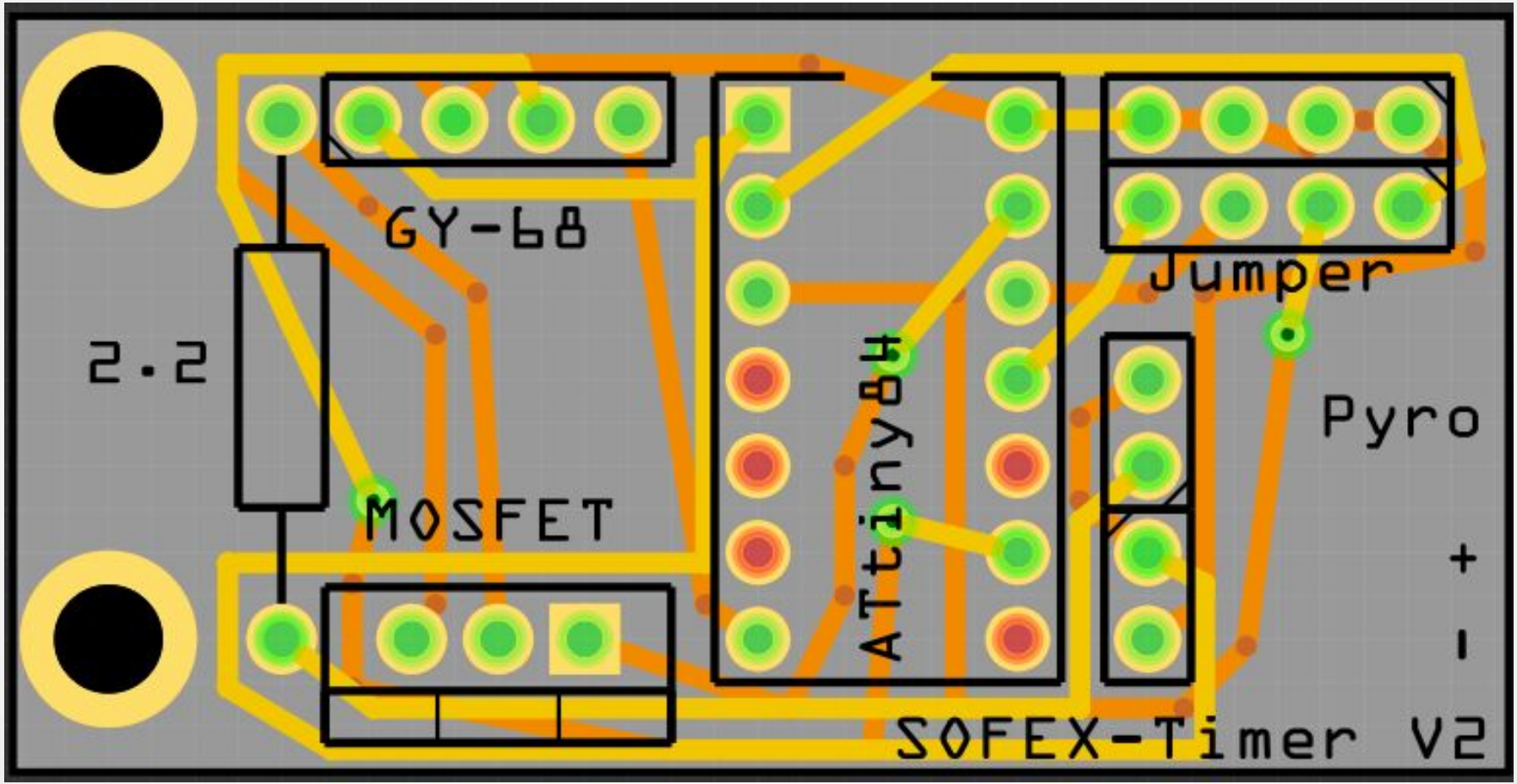
Fritzing: SOFEX-Timer Steckbrett



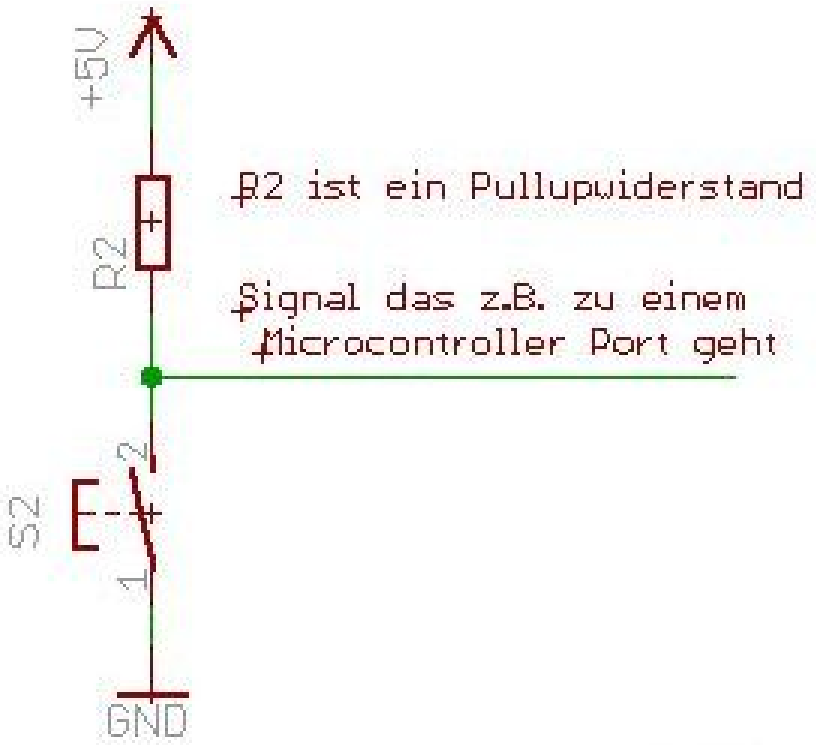
Fritzing: SOFEX-Timer Schaltplan



Fritzing: SOFEX-Timer Layout



Pullup-Widerstand



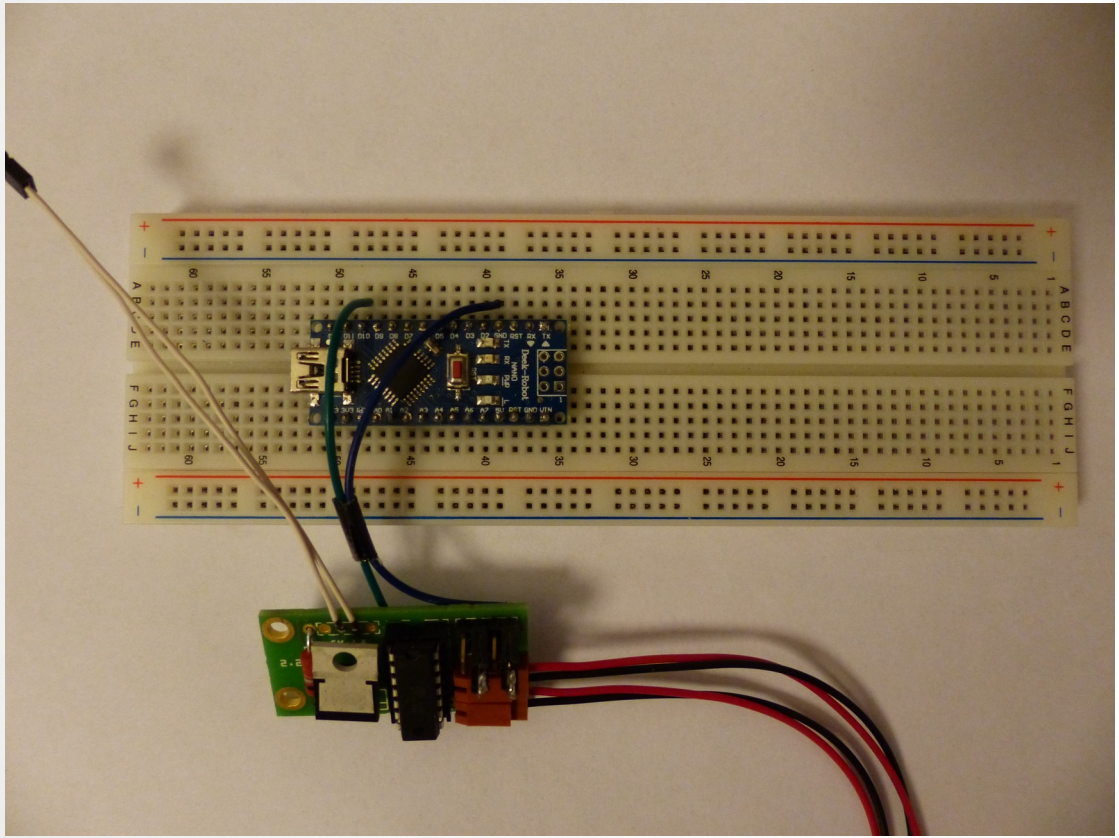
- Mit Pullup-Widerstand:
  pinMode(pinX, INPUT_PULLUP)
- Ohne Pullup-Widerstand:
  pinMode(pinX, INPUT)

Testausgaben: SerialRelay

Testausgaben: SerialRelay

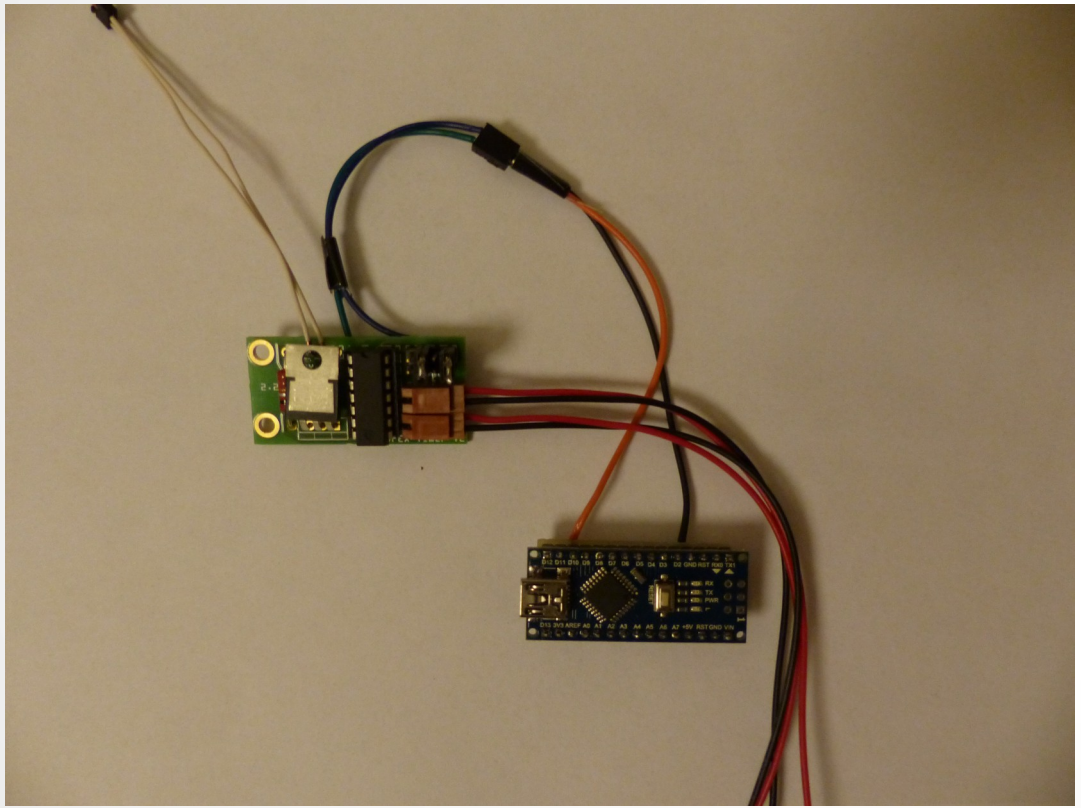
```
#include <SoftwareSerial.h>

SoftwareSerial mySerial(10, 11); // RX, TX

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native USB port only
  }


  Serial.println("Rcv data:");

  // set the data rate for the SoftwareSerial port
  mySerial.begin(9600);
  mySerial.println("Hello, world?");
}

void loop() { // run over and over
  if (mySerial.available()) {
    Serial.write(mySerial.read());
  }
  if (Serial.available()) {
    mySerial.write(Serial.read());
  }
}
```

Testausgaben: SerialRelay



Arduino Nano als Relais
auf Steckbrett

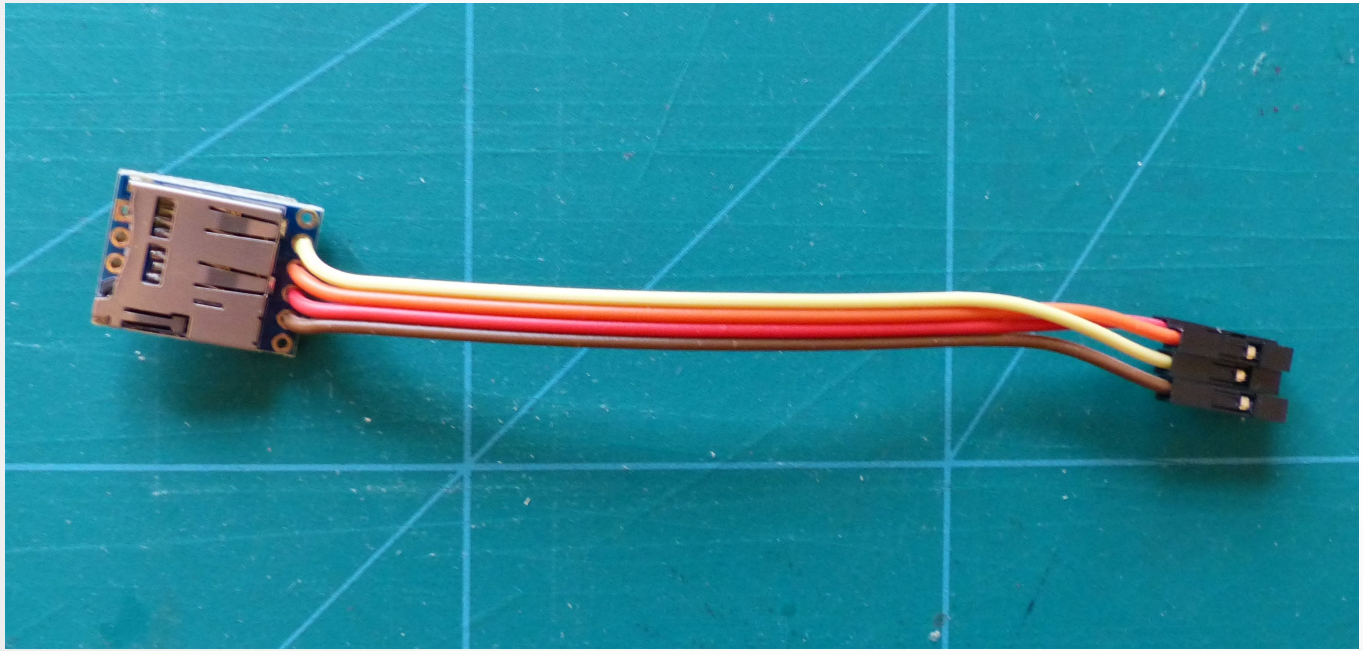
Testausgaben: SerialRelay



Arduino Nano als Relais gelötet

Testausgaben: Display (wenn man hat …)

Testausgaben in Datei schreiben



OpenLog
- Vcc
- Gnd
- Tx
- Rx

SoftwareSerial zur Ausgabe