

WANKER DEFENSE SYSTEM - BAUANLEITUNG

Benötigte Teile

[Teensy LC](#)

[Relay Modul](#)

[Reed Switch](#)

[Pushbutton](#)

[Rote LED](#)

[330 Ohm Widerstand](#)

[Verlängerungskabel](#)

[Micro USB Kabel](#)

[Kabel zum Verbinden der Komponenten](#)

[Gehäuse](#)

Magnet

Aufbau

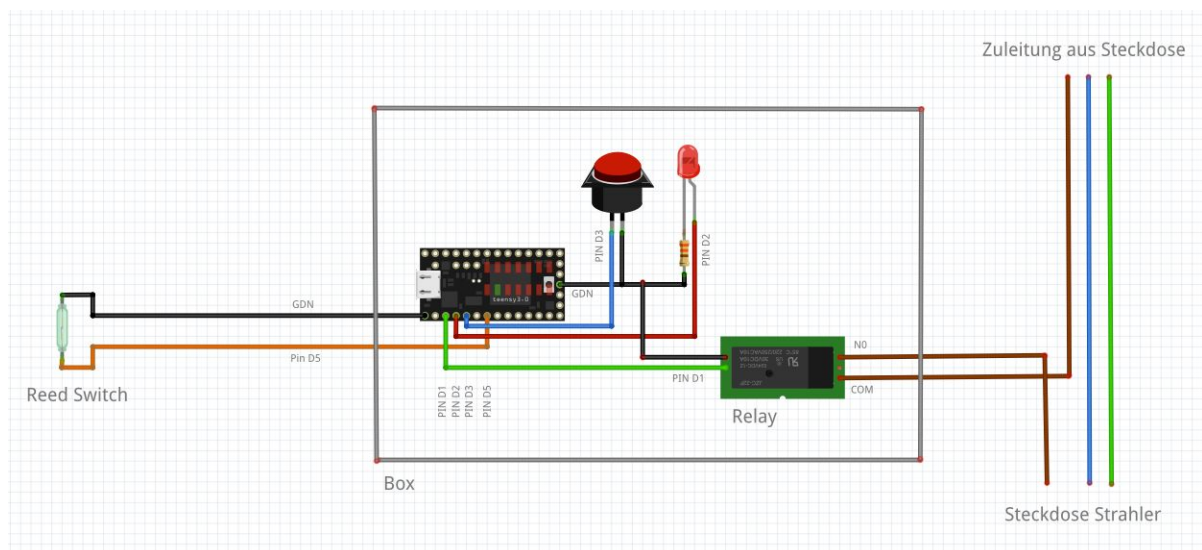
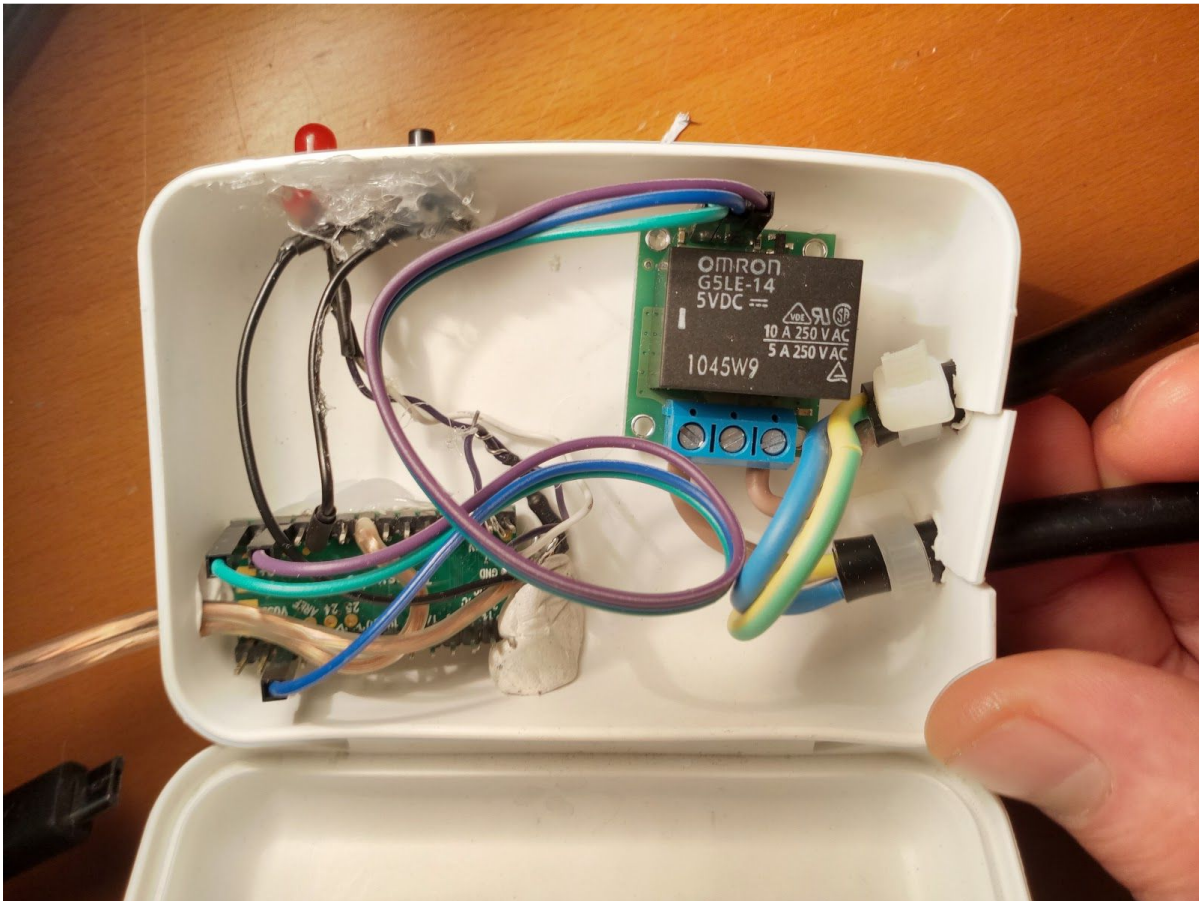
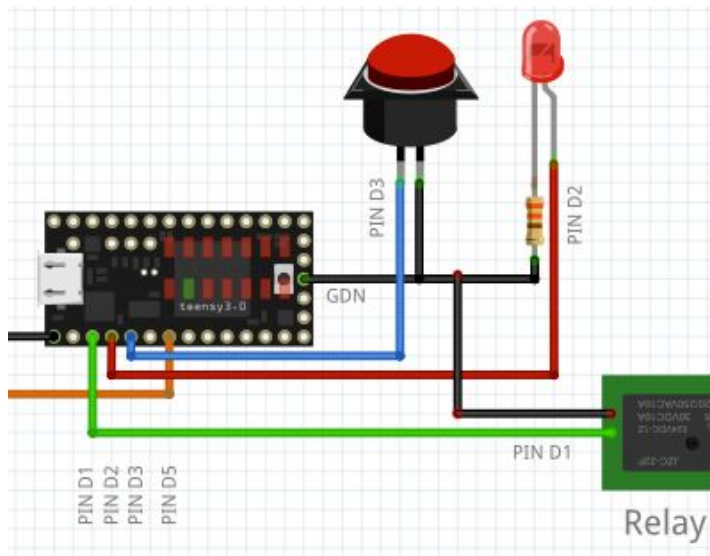


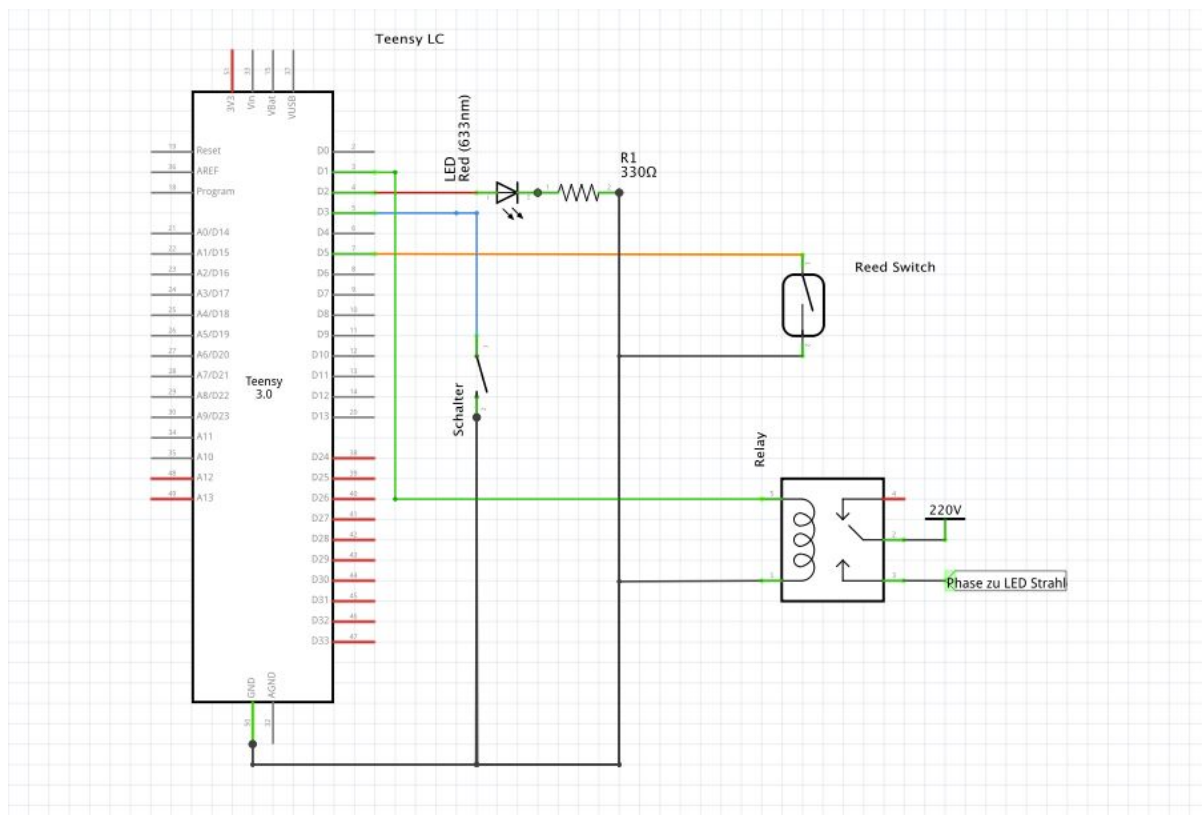
Foto Aufbau



Detail Teensy



Schaltplan



Hardware

1. Hardware nach Schaltplan verbinden. Das Relay Modul hat noch einen VIN Pin der nicht im Schaltplan auftaucht. Diesen mit dem 3V Anschluss vom Teensy verbinden.
2. Das Verlängerungskabel auf einer Länge von 10cm entmanteln und die braune Leitung durchtrennen. Das braune Kabel welches vom Aschlussstecker kommt wird nun mit dem "COM" Eingang am Relay verbunden. Das andere braune Kabel kommt an den "NO" Anschluss. Sobald am Eingang nun eine Spannung vom Teensy Pin D1 kommt schaltet das Relay eine Verbindung zwischen COM und NO und der Strom kann zur Anschlussbuchse fließen.

Software Installation

1. Arduino IDE downloaden, installieren und öffnen:
<http://www.arduino.cc/en/Main/Software>
2. Arduino schließen.

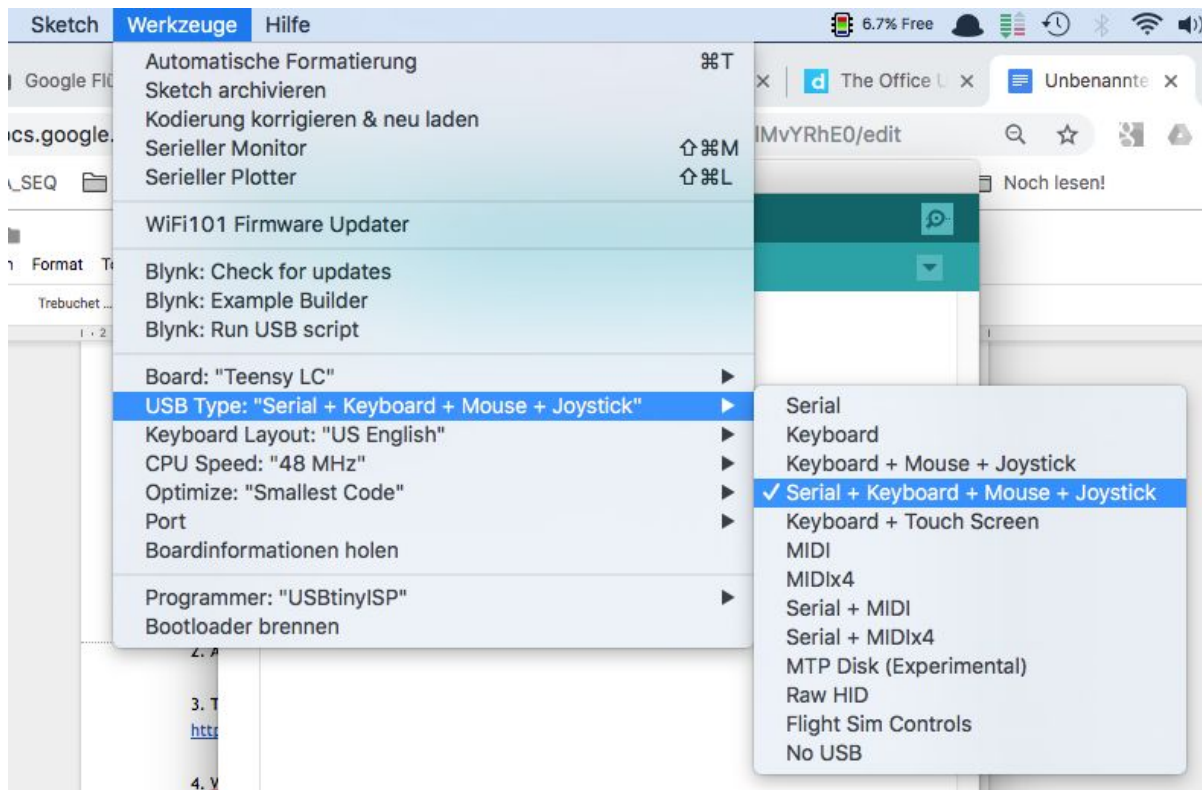
3. Teensyduino downloaden und installieren:

http://www.pjrc.com/teensy/td_download.html

4. Arduino Sketch WFilesDS_v1.3.ino in Arduino öffnen

5. Im Reiter *Werkzeuge* bei *Board* "Teensy LC" auswählen

6. Im Reiter *Werkzeuge* bei *USB Type* "Serial+Keyboard+Mouse+Joystick" auswählen



7. Im Sketch für Windows eine "0" bei "#define MACINTOSH eingeben, für OS X eine 1

```
4 |
id 5 | #define MACINTOSH 1 // set 0 for Windows
6 |
```

8. Im Sketchfenster auf Upload klicken



Programmcode WDS_v1.3.ino

```
// WANKER DEFENSE SYSTEM V1.3 written for BOHEMIAN BROWSER QUARTETT
```

```
#include <Bounce.h>
```

```
#define MACINTOSH 1 // set 0 for Windows
```

```
#define RELAY_PIN 1
```

```
#define LED_PIN 2
```

```
#define BTN_PIN 3
```

```
#define SENS_PIN 5
```

```
#define RELAY_ON 1
```

```
#define RELAY_OFF 0
```

```
boolean isActive = false;
```

```
boolean alarmsOn = false;
```

```
Bounce button0 = Bounce(BTN_PIN, 10); // Debounce Time
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  Serial.println("Start");
```

```
  pinMode(RELAY_PIN, OUTPUT);
```

```
  pinMode(LED_PIN, OUTPUT);
```

```
  pinMode(BTN_PIN, INPUT_PULLUP);
```

```
  pinMode(SENS_PIN, INPUT_PULLUP);
```

```
}
```

```
void loop() {
```

```
  button0.update();
```

```
  read_Button();
```

```
  if(isActive){
```

```
    digitalWrite(LED_PIN,1); //turn Activity Light On
```

```
    if(alarmsOn == false){
```

```
      boolean r = digitalRead(SENS_PIN);
```

```
      if(r == true){turn_Alarm_On();}
```

```
    }
```

```
  }
```

```
}
```

```
void read_Button(){
```

```

if (button0.fallingEdge()) {
  if(isActive) {
    turn_Alarm_Off();
  } else {
    isActive = true;
    Serial.println("Alarm active");
  }
}

}

void turn_Alarm_On(){
  alarmsOn = true;

  // Turn Flashlight On
  Serial.println("Flashlight ON");
  digitalWrite(RELAY_PIN, RELAY_ON);

  // Send Keystroke
  Serial.println("Send Keystroke");
  sendCloseKeystroke();

}

void turn_Alarm_Off(){
  alarmsOn = false;
  isActive = false;

  // Turn Flashlight Off
  Serial.println("Flashlight Off");
  digitalWrite(RELAY_PIN, RELAY_OFF);

  // Turn LED Off
  Serial.println("LED Off");
  digitalWrite(LED_PIN, RELAY_OFF);
}

void sendCloseKeystroke(){

Serial.println("SEND KEYSTROKE");

  if(MACINTOSH){
    // press and hold CTRL
    Keyboard.set_modifier(MODIFIERKEY_GUI);
    Keyboard.send_now();
  }
}

```

```
// press Key W, while cmd/win logo still held
Keyboard.set_key1(KEY_Q);
Keyboard.send_now();

// release all the keys at the same instant
Keyboard.set_modifier(0);
Keyboard.set_key1(0);
Keyboard.send_now();
} else {
// press and hold ALT
Keyboard.set_modifier(MODIFIERKEY_ALT);
Keyboard.send_now();

// press Key F4, while ALT still held
Keyboard.set_key1(KEY_F4);
Keyboard.send_now();

// release all the keys at the same instant
Keyboard.set_modifier(0);
Keyboard.set_key1(0);
Keyboard.send_now();
}
}
```