Winkekatze

Inventarnummer	g????
Ort	Elektrowerkstatt
Eigentümer	Marco

Ansprechpartner

- Daniel Stingl



Entstehung

Hierbei handelt es sich um eine modifizierte Winkekatze. Die Idee hierzu hatte ich, als sich der Begriff China-shop bezüglich Marcos doch beträchtliche Auswahl an China Elektronik eingebürgert hat.

Ursprünglich hatte ich geplant einen Sensor an die Schublade wo Marco die Teile lagert anzubringen. Nachdem ich aber dann bemerkt hatte, dass mit dieser Lösung die Katze nur einige Minuten/Jahr winken würde, habe ich die Katze mit einem PIR Bewegungssensor ausgestattet.

Seither winkt die Katze, wenn Marco (oder jemand anderes) auf ,,seinen*„ Platz sitzt.

* Im Hackerspace gibt es natürlich keine feste Sitzordnung, es handelt sich lediglich um seinen Lieblingsplatz!

Technische Umsetzung

Die Steuerung des Winkens wurde ursprünglich mit AA Zellen und einem an/aus Schalter umgesetzt .

Ich habe die Stromversorgung um Wartungsfreien/armen Betrieb zu gewährleisten mit einer 18615 Li-Ion zelle und einem Charging modul ausgestattet.

Das Winken steuert ein Ardoino pro mini 3,3 V 8 MHZ.

Bei Bewegungsmeldung des PIR's schaltet nach einer kurzen Verzögerung ein Mosfet die Winkekatze an.

Wenn eine gewisse Zeit keine Bewegung gemeldet wurde, stellt die Katze das Winken wieder ein.

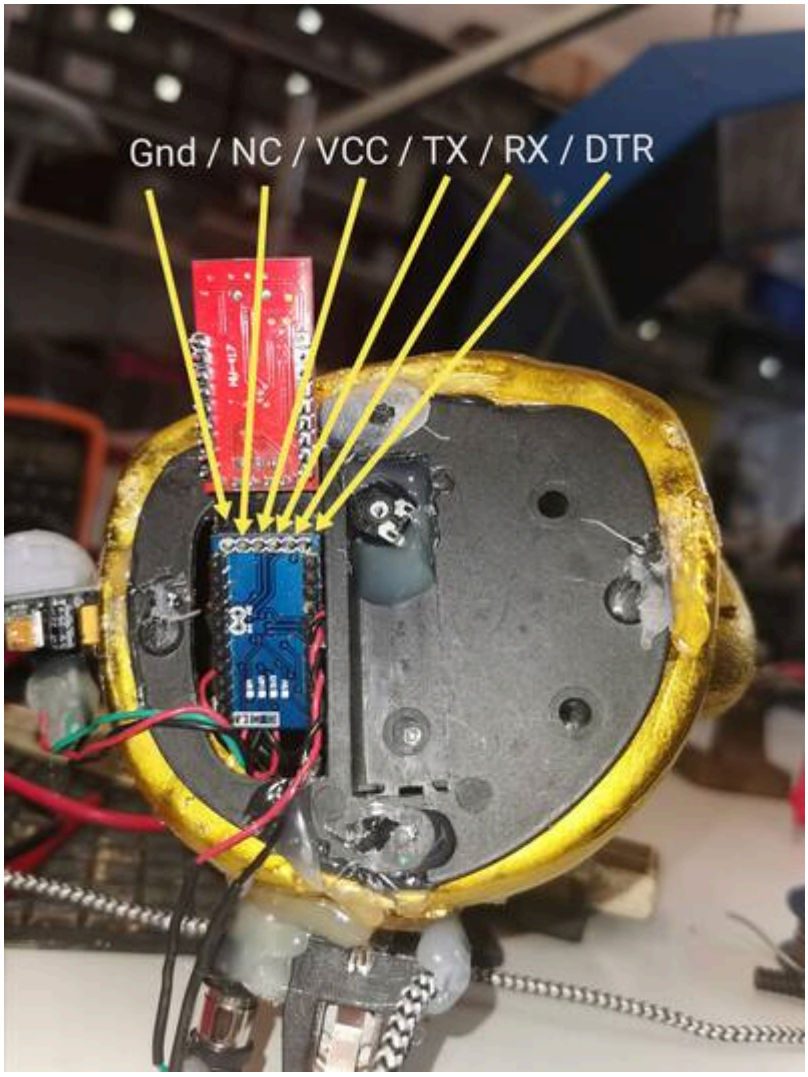
Die Winkstärke kann am Poti (Boden) eingestellt werden, leider ist die Poti wahl nicht so gut meinerseits ausgefallen, so kann man nicht sehr fein zwischen stark und schwach Winken einstellen.

Eine Verbesserung wäre, das Poti gegen ein geeigneteres zu ersetzten,

Wer das und/oder Softwareänderungen vornehmen möchte, kann den nachfolgenden Quellcode benutzten/anpassen:

Programmiert werden, kann die Katze wie folgt:

Mittels USB-Serial Wandler (TLL), den Anschluss (am Boden) verbinden.



Source Code

```
#include "LowPower.h"

<note>
// Winkekatze
// promini Nano 3,3 V

const int WinkePin = 3;
const int Sensor = 2;          // Der Taster wurde an Pin 2 angeschlossen
volatile int PIRStatus = 0;         // Variable die den Status des Taster
ausliest, wird der sog. "Vektor" 0 zugewiesen
volatile int WinkePinStatus = 0;
int winkOnDelay = 5;
int winktimeout = 30;
int winken = 0;

int winkOnDelaycounter = 0;
int winktimeoutcounter = 0;

// Taster Anschluss an Pin 2 entspricht dem Vektor 0   (hier der Fall)
```

```
// Taster Anschluss an Pin 3 entspricht dem Vektor 1

void setup() {
  pinMode(Sensor, INPUT);       // Hier wird der Sensor als INPUT definiert
  pinMode(WinkePin, OUTPUT);
  Serial.begin(9600);
  Serial.println("Winken aus");
  digitalWrite(WinkePin, LOW);
}

void loop() {

  PIRStatus = digitalRead(Sensor);      // ... wird der PIRStatus neu
definiert ("volatile"). Die neue Definition geschieht durch das Auslesen des
Sensors an Pin 2. ...
  WinkePinStatus = digitalRead(WinkePin);

  if (PIRStatus == HIGH) {
    winktimeoutcounter = 0;
    if (winkOnDelaycounter >= winkOnDelay && winken == 0) {
      Serial.println("Winken an");
      winkOnDelay = 0;
      digitalWrite(WinkePin, HIGH);
      winken = 1;
    } else {
      if (winkOnDelaycounter < winkOnDelay) {
        winkOnDelaycounter++;
      }
    }
  } else {
    if (winktimeoutcounter < winktimeout) {
      winktimeoutcounter++;
      if (winkOnDelaycounter > 0) {
        winkOnDelaycounter--;
      }
    }
  }

  if (winktimeoutcounter >= winktimeout && winken == 1) {
    Serial.println("Winken aus");
    digitalWrite(WinkePin, LOW);
    winken = 0;
  }

  Serial.print("WinkePinStatus:");
  Serial.println(WinkePinStatus);
  Serial.print("winken:");
  Serial.println(winken);
  Serial.print("PIRStatus:");
  Serial.println(PIRStatus);
```

```
  Serial.print("winkOnDelaycounter:");
  Serial.println(winkOnDelaycounter);
  Serial.print("winktimeoutcounter:");
  Serial.println(winktimeoutcounter);

  delay(100);
  Serial.println("deepsleep");
  delay(100);
  LowPower.powerDown(SLEEP_8S, ADC_OFF, BOD_OFF);
  delay(100);

}
```

From:
https://wiki.hackerspace-bremen.de/ - **Hackerspace Bremen e.V.**

Permanent link:
https://wiki.hackerspace-bremen.de/geraetschaften/elektronik/winkekatze

Last update: **2022-11-17 22:34**