

Zwischenpräsentation 11. Oktober 2017

Drehteller revisited

Tobias H.

Nico B.

1. Inbetriebnahme und Analyse des Drehtellers auf dem Stand der Vorgängergruppe
2. Schritte in Richtung der Projekt-Ziele
3. Fazit / Skizzierung des weiteren Vorgehens

Projektziele



Projektziele:

- ESP8266 soll ersetzt werden durch ESP32
- LED-Ring zur Beleuchtung soll angeschlossen werden
- Elektronik soll komplett ins „Innere“ des Drehtellers integriert werden
- Je nach Zeit evtl. andere Weiterentwicklungen...

Eigenschaften

- Drehteller
 - für Objekt oder Kamera
 - Antrieb über Schrittmotor
- Elektronik
 - auf Steckplatine
 - Stromversorgung über USB
 - ESP8266
 - Motortreiber
 - 2 Taster
 - Kondensator (Spannungstabilisierung)
- Steuerung über WiFi/Webserver



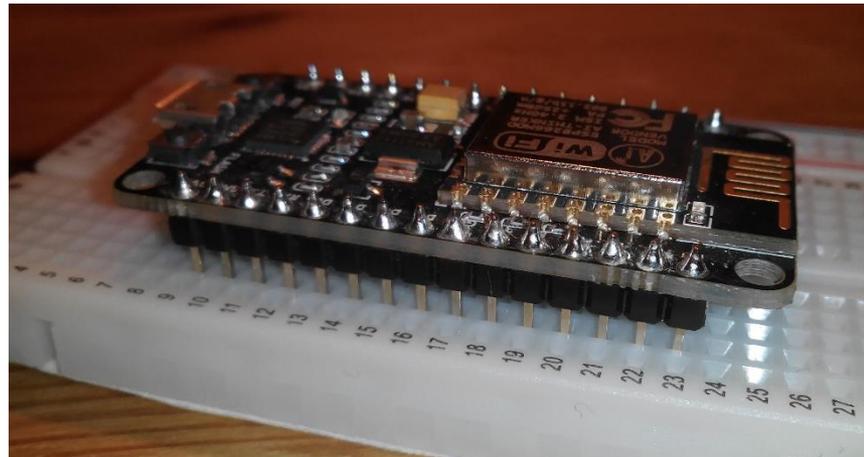
Bild der Vorgängergruppe

Problem:

ESP8266 verbindet sich nicht mit Hotspot (Smartphone)

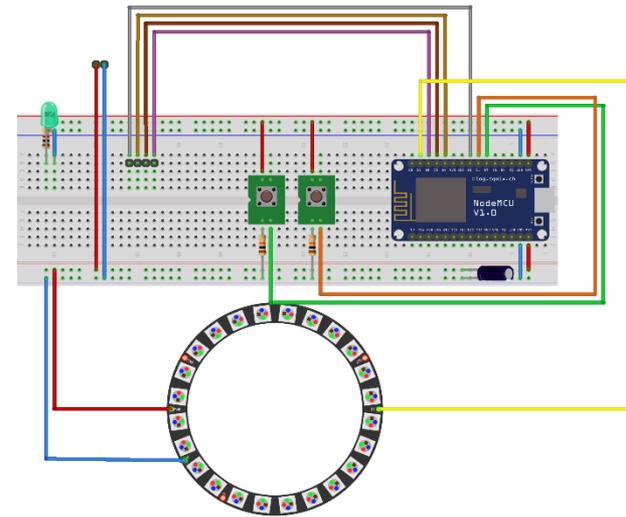
Lösung:

ESP8266 darf nicht ganz in die Steckplatine gesteckt werden
→ sonst wahrscheinlich Abschirmung!

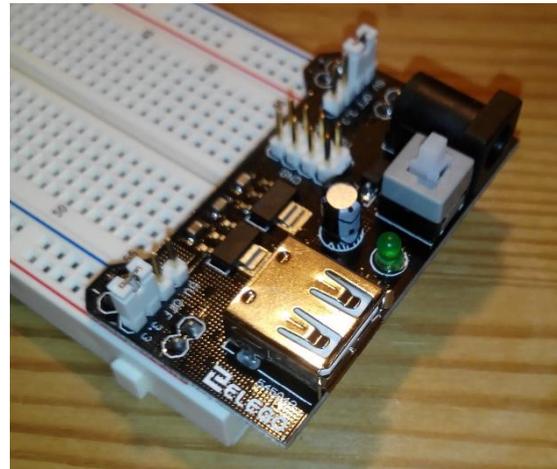


Inbetriebnahme

- Anschließen bzw. anlöten des LED-Rings nach Fritzing-Plan

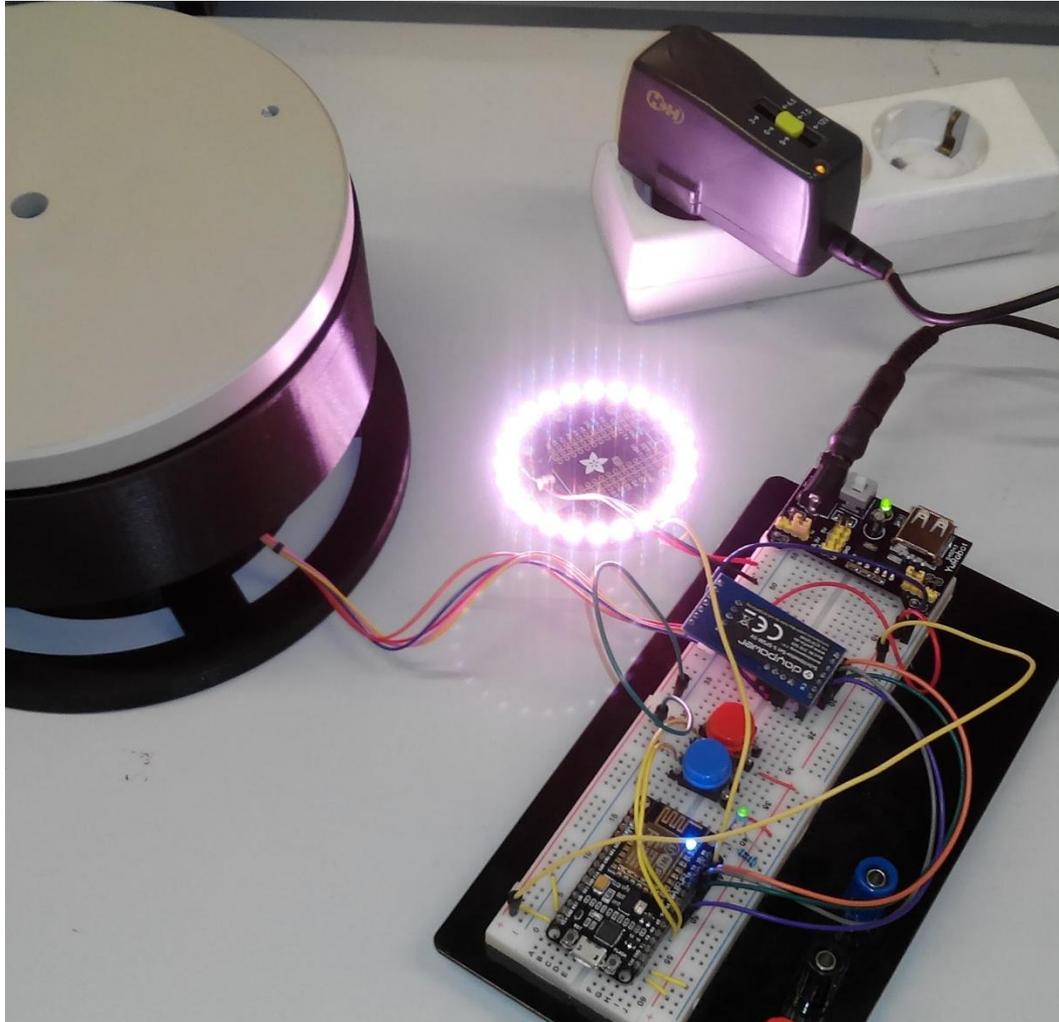


fritzing



- Inbetriebnahme der Platine zur Stromversorgung über Netzteil

Inbetriebnahme



← → ↻ 🏠 ⓘ 192.168.43.37

Drehteller Steuerung

Projektteam:
lajo1020
esti1011

RGB-LED:

CC66FF

R - Rot:

G - Gelb:

B - Blau:

Helligkeit:

Motor:

Left Stop Right

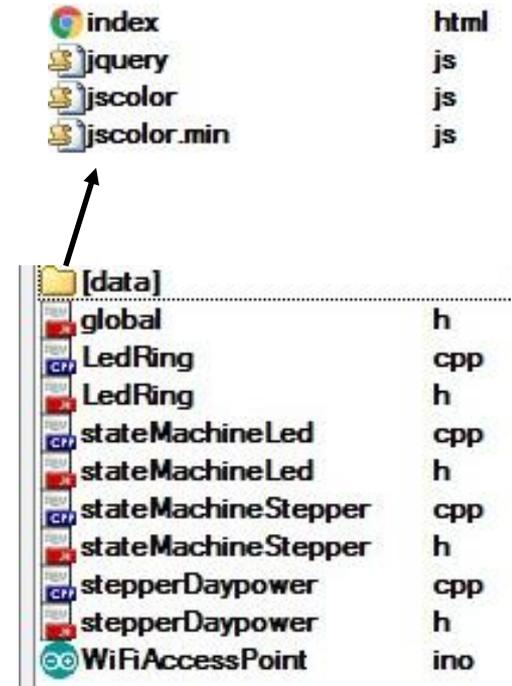
Speed:

Inbetriebnahme



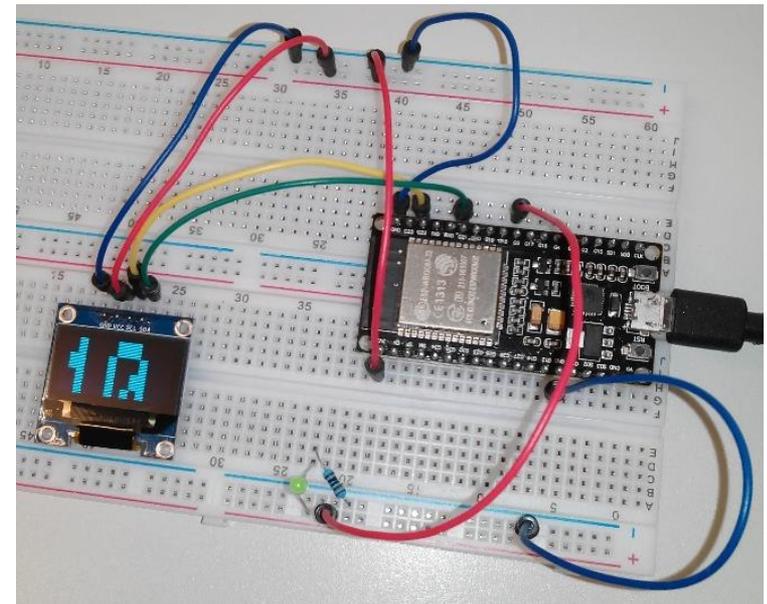
Analyse

- Arduino IDE einrichten für ESP8266
- Ziel: Quellcode der Vorgänger selber flashen
 - `WiFiAccessPoint.ino`
 - Dateien für Web-Oberfläche im Ordner `data`
→ SPIFFS (SPI Flash File System)
 - Einfache Übertragung bei ESP8266 durch Plugin möglich
- Flashen funktioniert → Drehteller läuft
- `jquery.js` nicht benötigt und nur eine von `jscolor.js` und `jscolor.min.js` wird benötigt



Beschäftigung mit ESP32

- Problem: deutlich neuer als ESP8266
→ weniger Hilfe im Internet
- Arduino IDE einrichten für ESP32
- Beispiele ausprobieren:
 - `Blink.ino`
 - `ESP32_OLED_Simple.ino`
 - `HallSensor.ino` (siehe nächste Folie)
 - `SimpleWebServerWiFi.ino`
- Wie funktioniert SPIFFS auf auf ESP32?
→ kein Plugin verfügbar



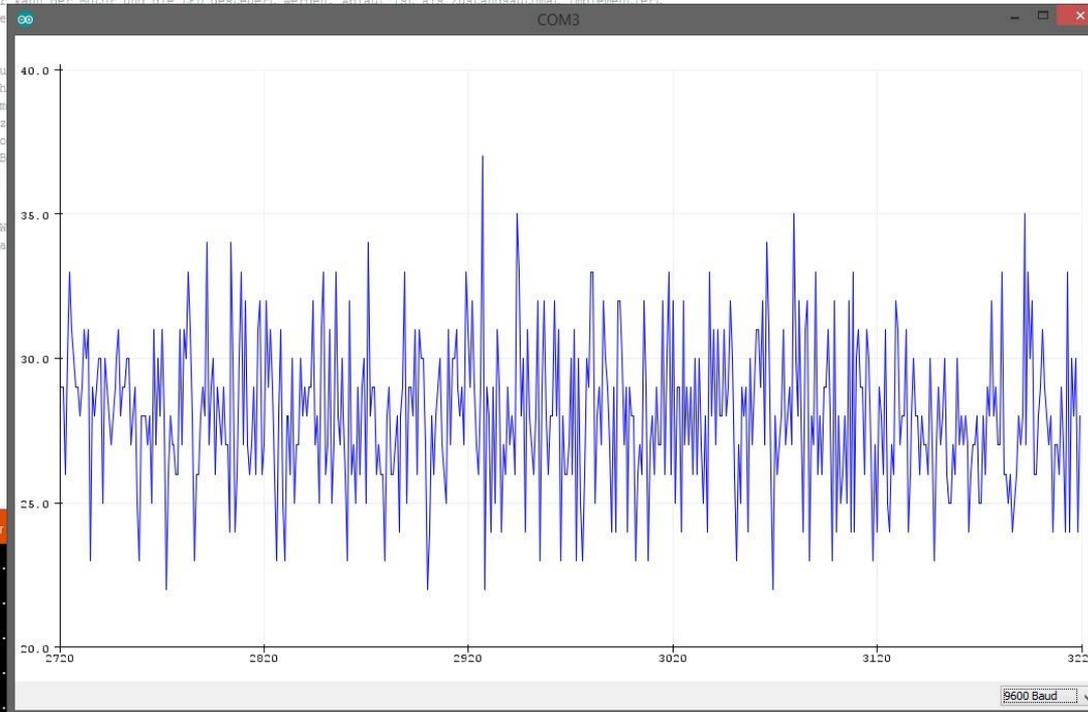
Hall-Sensor auf ESP32

det sich zu einem Wifi-Accesspoint. Zugangsdaten sind im Code mit ssid und password deklariert
e NodeMCU verbunden hat kann man im Browser die Adresse <http://192.168.43.27> aufrufen. Dies wurde bei einem Android-Hotspot getestet
ann dann die Farbe der RGB-LED verändert werden und der Motor angesteuert werden
lien Handschalter kann der Motor und die LED gesteuert werden. Ablauf ist als Zustandsautomat implementiert
lle Schnittstelle

Web-Dateien:
-IDE muss das Flu
.müssen die die h
ich die Dateien m
zu kurzen Latenz
Ändern der html c
mit dem Laptop(B
r unten) wird im

Wifi:
odeMCU auch als W
wahrscheinlich da

```
>  
Fi.h>  
FiMulti.h>  
sServer.h>  
bServer.h>  
NS.h>
```



d der serielle Monitor

Fehlern

```
..... [ 94% ]  
..... [ 96% ]  
..... [ 99% ]  
..... [ 100% ]
```

Beschäftigung mit ESP32

- Partitionstabelle (unter Arduino IDE) in
...\\esp32\\tools\\partitions\\
default.csv
- Erstelle SPIFFS-Image mit dem tool
mkspiffs (Achtung: Version 0.1.3 wählen, die
aktuelle 0.2.0 macht Probleme):
mkspiffs -c [src_folder] -b 4096 -
p 256 -s 0x16F000 spiffs.bin
- Flashen an die richtige Stelle mit esptool:
esptool.exe ... write-flash 0x291000
spiffs.bin
- Letzter Stand: read funktioniert noch nicht...
- Alternative falls kein Erfolg
→ keine Kapselung, sondern
string in der *.ino

#	Name,	Type,	SubType,	Offset,	Size,	Flags
2	nvs,	data,	nvs,	0x9000,	0x5000,	
3	otadata,	data,	ota,	0xe000,	0x2000,	
4	app0,	app,	ota_0,	0x10000,	0x140000,	
5	app1,	app,	ota_1,	0x150000,	0x140000,	
6	eeeprom,	data,	0x99,	0x290000,	0x1000,	
7	spiffs,	data,	spiffs,	0x291000,	0x16F000,	

```
Listing directory: /
                FILE: /index.html  SIZE:
FILE: /jscolor.js  SIZE: 53709
FILE: /zzz.txt    SIZE: 43
FILE: /hello.txt  SIZE: 6
Reading file: /zzz.txt
                Read from file:
```

- Intensive Beschäftigung mit ESP32
 - Große Fortschritte beim allgemeinen Verständnis
 - wir sind optimistisch
- Nächste Schritte
 - Portierung vorantreiben, evtl. ohne SPIFFS
 - nicht durch Kleinigkeiten aufhalten lassen
 - Ziel nicht aus dem Auge verlieren
 - Die Integration kann erst danach in Angriff genommen werden (richtig?)
 - Onlinedokumentation muss überarbeitet werden



Vielen Dank
für Ihre Aufmerksamkeit