

RaspberryPi Zero Install

einrichten als WS2812b Controller

von thorsten kattaneck, 01.11.2021

Grundsystem installieren

- neustes Image von <https://www.raspberrypi.com/software/operating-systems/> runter laden
- ich nutze hier die Lite Version ohne Desktop
- image auf SD kopieren (unter Linux): **sudo dd if=./image of=/dev/sd? bs=1M**
- WLAN zugriff einrichten
 - Boot Partition der SD Karte einhängen und eine Datei mit dem Namen wpa_supplicant.conf erstellen mit folgendem Inhalt:

```
country=DE
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
    ssid="xxx"                <= statt xxx muss die SSID deines WLAN
    psk="xxx"                <= statt xxx muss dein WLAN Password rein
    key_mgmt=WPA-PSK
}
```

- um den SSH Zugriff zu aktivieren muss lediglich eine leere Datei namens ssh in die boot partition eingefügt werden
- nun kann der raspberrypi mit der SD Karte gestartet werden, nach 1-2 min sollte man den raspberry an pinggen können → ping raspberrypi.local
- nun kann man sich per ssh auf dem raspberry anmelden → ssh pi@raspberrypi.local
- das Passwort lautet Standard „raspberry“
- nun könne wir den Pi erst mal konfigurieren → sudo raspi-config
- als erstes das Passwort ändern ! (System Options → Password)
- dann den hostname ändern ! (System Options → Hostname) z.B. „led-ctrl-01“
- dann Filesystem maximieren ! (Advanced Options → Expand Filesystem)
- dann Zeitzone einstellen ! (Localisation Options → Timezone)
- dann kann erst mal alles auf dem neusten Stand gebracht werden: sudo apt update und sudo apt upgrade
- Audio deaktivieren
 - sudo nano /etc/modprobe.d/raspi-blacklist.conf
 - dort fügen wir **blacklist snd_bcm2835** ein
 - sudo nano /boot/config.txt
 - dtparam=audio=on einfach auskommentieren
 - nach einem Neustart ist das Audio deaktiviert

Library erstellen und installieren

- sudo apt-get install gcc make cmake build-essential git
- git clone https://github.com/jgarff/rpi_ws281x
- cd rpi_ws281x
- mkdir build
- cmake ..
- make

- `sudo make install` → Installiert die Library nach `usr/local/lib` und `usr/local/include/ws2811`

Software erstellen

- `git clone https://github.com/ThKattanek/window_neopixel_matrix.git`
- `cd window_neopixel_matrix/raspi_neopixel_player`
- `mkdir build`
- `cd build`
- `cmake ..`
- `make`
- `sudo make install`
- `sudo ./neopixel_player` startet den player

Testanwendung automatisch beim Start ausführen

- Folgendes Skript mit dem Namen „neopixel_player“ unter `/etc/init.d` speichern.

```
#!/bin/sh
### this skript running the ws2812 controller while pi start
# Actions
case "$1" in
  start)
    neopixel_player
    ;;
  stop)
    # STOP
    ;;
  restart)
    # RESTART
    ;;
  esac

exit 0
```

- `chmod +x /etc/init.d/ neopixel_player`
- `sudo update-rc.d neopixel_player defaults`
- mit `sudo update-rc.d neopixel_player remove` kann es wieder entfernt werden

Nützliche Linux Befehle

- ✓ `top`
- ✓ `htop`
- ✓ `shutdown now`

Quellen:

- <https://medium.com/@larsongb/using-a-raspberry-pi-to-control-leds-38731b9db4>
- <https://developer-blog.net/raspberry-pi-zero-als-led-strip-controller/>
- <https://www.computerhilfen.de/info/raspberry-pi-zero-w-ohne-monitor-einrichten-wlan-und-ssh-aktivieren.html>
- <https://www.raspberrypi.com/software/operating-systems/>
- <https://blog.heimetli.ch/raspberry-led-strip-c.html>
- https://github.com/jgarff/rpi_ws281x
- <https://jankarres.de/2014/07/raspberry-pi-autostart-von-programmen-einrichten/>