

# Raspberry-Pi Tetris Projekt

## Raspberry-Pi Tetris Projekt

# Inhalt

- [S. 3] ----- ( *Aufbau* ) -----

- benötigte Materialien
- Aufsetzen des Raspberry Pi
- Einrichtung des Betriebssystems
- Aufbau des Raspberry Pi

- [S. 8] ----- ( *Starten des Projektes* ) -----

- Klonen des Projektes
- Umgang mit der Virtuellen Umgebung

- [S. 9] ----- ( *einblick in den Code* ) -----

- Projekt Struktur
- Beschreibung der Inhalte

# Aufbau

## benötigte Materialien

die Folgenden Gegenstände werden benötigt:

- Raspberry Pi 4 Model B
- Micro-SC Karte mit Raspberry Pi OS
- Breadboard
- 3x Led
- 3x 220 resistor
- Jump-Wires
- Knopf-Modul
- Joystick-Modul
- PCF8591 ADC-Modul
- HDMI-Micro HDMI Kabel
- USB-C Kabel
- Maus
- Tastatur
- Monitor

# Aufsetzen des Raspberry

Lade den [Raspberry-Pi Imager](#) herunter

1. wähle bei Raspberry Pi Modell: [Raspberry Pi 4](#) aus
2. wähle bei Betriebssystem (OS): [Raspberry Pi OS \(64-Bit\)](#) aus
3. wähle deine micro-SD Karte aus und schreibe diese

schließe Tastatur, Maus und Monitor an den Raspberry an und stecke die Micro-SD Karte ein

## Einrichtung des Betriebssystems

wenn sie sich im Raspberry Pi OS befindest, rufe das Terminal auf und updaten sie ihr System

```
sudo apt update  
sudo apt upgrade
```

nun installiere die Notwendigen Pakete:

```
sudo apt install python3 python3-virtualenv
```

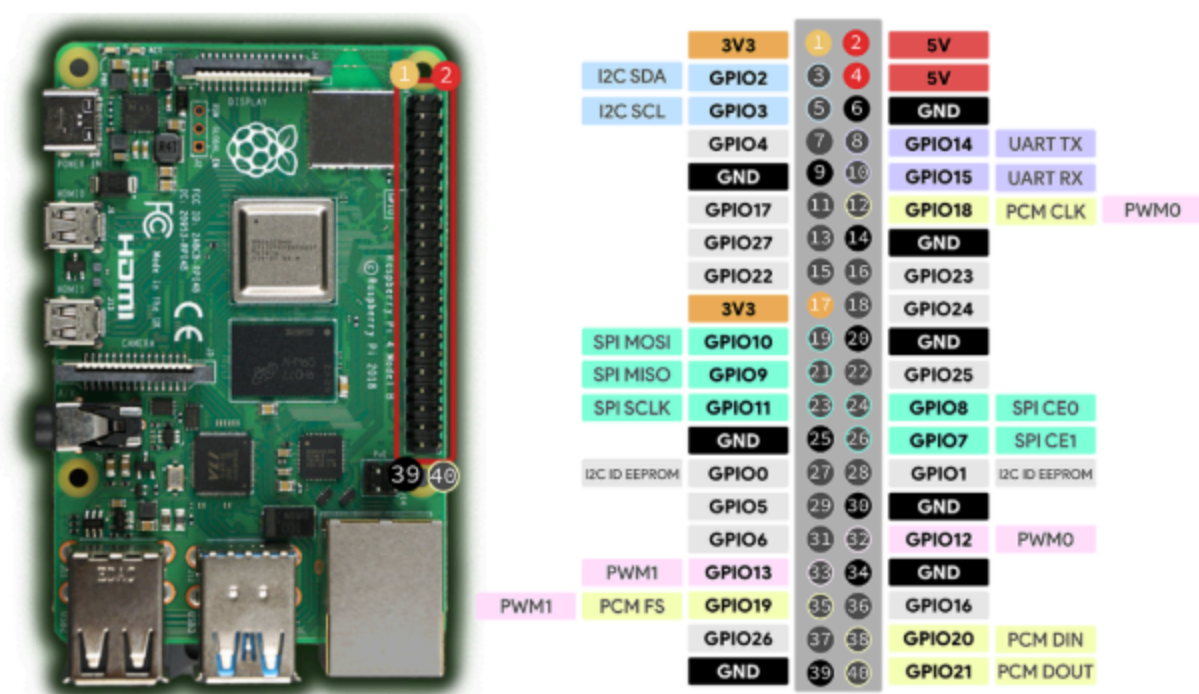
jetzt müssen sie nur noch I2C auf den Raspberry Aktivieren  
rufe die raspberry pi config TUI auf

```
sudo raspi-config
```

nun wähle zuerst **Interface Options** aus und dann I2C und aktiviere es  
führen sie nun ein neustart durch

```
sudo reboot
```

# Aufbau des Raspberry Pi



## 1. Breadboard

- **Pin 6 (GND)** → Minus-Schiene des Breadboards
- jede **LED** hat eine Verbindung zur **Minus-reihe**
- der **Joystick** (GND-Pin) hat eine Verbindung zur **Minus-reihe**
- das **Button-Modul** (Minus Pin) hat eine Verbindung zur **Minus-reihe**
- das **PCF8591** (GND Pin) Modul hat eine Verbindung zur **Minus-reihe**

## 1. LEDs anschließen (GPIO Output)

LED	Raspberry Pi GPIO-Pin
LED 1	GPIO 17 (Pin 11)
LED 2	GPIO 27 (Pin 13)
LED 3	GPIO 22 (Pin 15)

- Jede LED benötigt einen **220  $\Omega$  Widerstand** vor **GND**.

## 3. PCF8591 anschließen

PCF8591 Pin	Raspberry Pi Pin
<b>VCC</b>	3.3V (Pin 1)
<b>SDA</b>	GPIO 2 (Pin 3)
<b>SCL</b>	GPIO 3 (Pin 5)

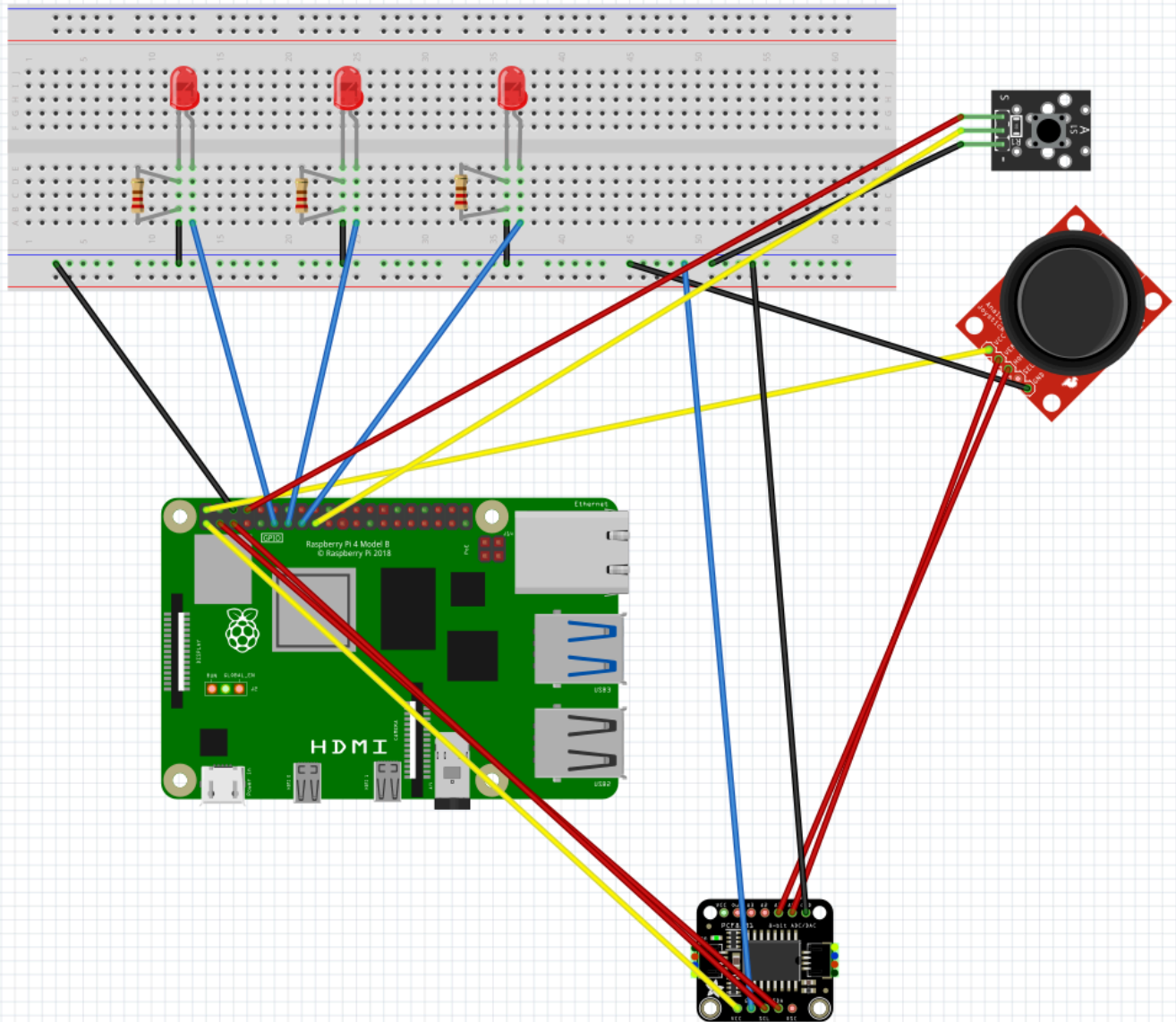
## 4. Joystick an den PCF8591 anschließen

Joystick Pin	PCF8591 Pin
<b>VRX (X-Achse)</b>	AIN0 (A0)
<b>VRY (Y-Achse)</b>	AIN1 (A1)

Joystick Pin	Raspberry Pi Pin
<b>VCC</b>	3.3V (Pin 1)

am ende sollte es wie folgt aussehen

Schwarz:	Ground
Blau:	GPIO Output
Rot:	GPIO input
Gelb:	Strom





## Starten des Projektes

um mit dem Projekt zu beginnen solltest du zuerst das Projekt auf deinen Raspberry Pi klonen

```
git clone https://github.com/Blitzyanic/RaspberryPi-Game-Project.git
```

nun gehe in das Projekt erstelle ein virtuelles environment und installiere die Pakete

### wechsel zum Projekt

```
cd RaspberryPi-Game-Project
```

### erstellen eines venv

```
python -m venv .venv
```

### starten des venv

```
source .venv/bin/activate
```

### Installation der Pakete

```
pip install -r requirements.txt
```

### Projekt Starten

```
python src/main.py
```

es sollte sich nun ein Fenster öffnen die Fallenden Blöcke können mit dem Joystick bewegt werden und mit dem Knopf ist es möglich diese zu drehen

wenn sie die decke erreichen verlieren sie ein Leben (Lampe geht aus)











wenn alle leben verbraucht sind können sie mit ein druck auf den Knopf das spiel neustarten

# Einblick in den Code

## Projekt Struktur

```
.
├── docs
├── LICENSE
├── main.py
├── README.md
├── requirements.txt
├── src
├──   ├── game
├──   │   ├── Figure.py
├──   │   └── Tetris.py
├──   └── raspberry
├──   └── Rpi.py
```

## Beschreibung der Inhalte

- **docs/** → Beinhaltet die Dokumentation wie diese hier 
- **LICENSE** → Das Projekt läuft unter der GPL v3 Lizenz 
- **main.py** → Hauptdatei des Projekts, verbindet das Tetris-Spiel mit dem Raspberry Pi 
- **README.md** → Einführung & Infos zum Projekt 
- **requirements.txt** → Enthält die benötigten Python-Pakete mit Versionen 
- **src/** → Quellcode des Projekts 
  - **game/** → Enthält den Code für das Tetris-Spiel 
    - **Figure.py** → Klasse für die Tetris-Blöcke 
    - **Tetris.py** → Enthält die Spiellogik 
  - **raspberry/** → Enthält den Code zur Steuerung des Raspberry Pi 
    - **Rpi.py** → Klasse zum Auslesen der Raspberry-Module 