11.11.2024

Betreuung:

PROF. DR. WOLFGANG ALBRECHT

Autoren:

Ashraf Yahya

Ahmed sajad Faiz

Studienarbeit

Praktische Anwendungen von Jupyter zur IoT-Steuerung und Datenerfassung: Sensor- und Aktorsteuerung auf dem Raspberry Pi

# Einführung

Diese Studienarbeit dokumentiert verschiedene Projekte zur Steuerung und Datenerfassung mithilfe eines Raspberry Pi. Ein zentrales Werkzeug dabei ist Jupyter Notebook, das in jedem Projekt genutzt wird, um den Code übersichtlich darzustellen und Schritt für Schritt zu erklären. Durch die interaktive Umgebung von Jupyter können theoretische Erläuterungen, Code, Diagramme und Ergebnisse direkt nebeneinander dargestellt werden, was das Lernen und Experimentieren erheblich erleichtert. Jupyter erlaubt es uns, die Funktionsweise jedes Projekts transparent und nachvollziehbar darzustellen, und macht es Anwendern einfach, Anpassungen vorzunehmen und die Ergebnisse sofort zu sehen.

Die Projekte umfassen die Ansteuerung von LEDs, die Steuerung eines Servomotors und die Nutzung von Sensoren zur Erfassung von Temperatur und Bewegung. Jupyter ermöglicht es, diese Projekte dynamisch zu präsentieren, indem Diagramme und Visualisierungen in Echtzeit aktualisiert werden, z.B. durch die Verwendung der Matplotlib-Bibliothek zur Darstellung von Sensordaten.

Darüber hinaus wird die Literatur und Online-Referenzen, die in der Arbeit verwendet wurden, direkt in den Notebooks als Links gespeichert, sodass Leser leicht auf zusätzliche Ressourcen zugreifen können. Jupyter ist daher nicht nur ein Tool zur Programmierung, sondern auch eine Plattform zur Dokumentation und Weiterbildung, die dieses Projekt zu einem wertvollen, interaktiven Lernwerkzeug macht.

# Verzeichnis

## LED-Steuerung (Gemeinsam)

Steuerung von LEDs mit PWM-Signalen auf dem Raspberry Pi. Ziel ist es, die grundlegenden Konzepte der GPIO-Steuerung und PWM-Anwendung für LED-Helligkeitsregulierung zu verstehen.

Notebook: LED\_Projekt

## Temperatursensor (Gemeinsam)

Misst die Umgebungstemperatur und stellt die Daten in Echtzeit dar. Mithilfe der Matplotlib-Bibliothek werden die Daten als Diagramm visualisiert, um Veränderungen anschaulich darzustellen.

Notebook: TempSensor\_Projekt

## Servo Motor (YAHYA)

Ein Servomotor wird über den Raspberry Pi gesteuert und bewegt sich je nach Vorgabe in Winkelstellungen zwischen 0 und 180 Grad. Dieses Projekt vermittelt die Anwendung der Pulsweitenmodulation (PWM) zur präzisen Steuerung von Servomotoren.

Notebook: ServoMotor\_Projekt

## Temperatursensor mit Buzzer und LED (YAHYA)

Mithilfe des BME680-Sensors wird die Umgebungstemperatur überwacht. Erreicht die Temperatur einen bestimmten Wert, sendet eine LED und ein Buzzer ein SOS-Signal, wodurch praxisorientierte Anwendungen der Temperatursteuerung demonstriert werden.

Notebook: BME680\_Buzzer\_LED\_Projekt

## DEBO SENS 9AXIS Sensor (YAHYA)

In diesem Projekt wird der DEBO SENS 9AXIS Sensor genutzt, um Bewegungs- und Lageinformationen zu erfassen und darzustellen. Die aufgezeichneten Daten werden mit geeigneten Funktionen analysiert und visualisiert.

Notebook: DEBO\_SENS\_9AXIS

## LED Display (FAIZ)

Steuerung einer 8x8-LED-Matrix zur Darstellung verschiedener Leuchtmuster. Hierbei wird das Freenove Projects Kit verwendet, um kreative Muster und Animationen auf der Matrix anzuzeigen.

Notebook: LED\_Display

## Motion Sensor (FAIZ)

Mit dem HC-SR501-Bewegungssensor wird Bewegung erkannt und über eine blinkende LED signalisiert. Dieses Projekt verdeutlicht die Grundlagen der Bewegungserkennung mithilfe des Freenove Projects Kit und des Raspberry Pi.

Notebook: Motion\_Sensor

# Allgemeine Notebooks

Einführung in Freenove Kit.ipynb:   
Einführung in die Nutzung des Freenove Kits, das in verschiedenen Projekten zur Steuerung und Sensorik verwendet wird.

Einführung in Matplotlib.ipynb:   
Einführung in die Matplotlib-Bibliothek zur Datenvisualisierung, die in Projekten zur Anzeige von Sensordaten verwendet wird.

Installationen.ipynb:   
Notwendige Installationsanweisungen für Bibliotheken und Abhängigkeiten auf dem Raspberry Pi, um die Projekte auszuführen.

Rasp\_Pi\_Einrichtung.ipynb:   
Einrichtung und Konfiguration des Raspberry Pi für die Nutzung mit Jupyter Notebooks und weiteren Projekten.

Raspberry\_Pi\_Kommunikation.ipynb:   
Einführung in die Kommunikation und Steuerung zwischen dem Raspberry Pi und externen Geräten.

Raspberry\_Pi\_AccessPoint\_Setup.md:   
Anleitung zur Konfiguration des Raspberry Pi als Access Point, um eine drahtlose Netzwerkverbindung für die Projekte bereitzustellen.

RaspberryPi\_Jupyter\_Grundlagen.ipynb:   
Grundlagen zur Nutzung von Jupyter Notebooks auf dem Raspberry Pi, inklusive Tipps zur Code-Entwicklung und -Ausführung.

# Strukturmuster für ein Projekt

Projektname: Servo Motor  
  
1. Einleitung  
 Beschreibung der Ziele und Relevanz des Projekts, z.B. wie die Steuerung eines  
 Servomotors nützlich ist, um die Prinzipien der Pulsweitenmodulation (PWM) zu  
 verstehen.  
  
2. Materialien  
 Liste der erforderlichen Komponenten (Servomotor, Raspberry Pi, Jumper-Kabel).  
  
3. Schaltplan  
 Darstellung des Aufbaus der Schaltung, die den Servomotor mit dem Raspberry Pi   
 verbindet, beispielsweise als Bild oder Diagramm.  
  
4. Setup und Installation  
 Anweisungen zur Installation benötigter Bibliotheken und Software (z.B. GPIO-Bibliothek   
 für Python).  
  
5. Code-Implementierung  
 Detaillierter Python-Code zur Steuerung des Servomotors mit Erklärungen zu den   
 Schritten, wie z.B. dem Einsatz von PWM für die präzise Bewegungssteuerung.  
  
6. Erweiterungen und Variationen  
 Vorschläge zur Projektweiterentwicklung, z.B. Steuerung des Motors basierend auf   
 Eingabewerten oder Sensor-Daten.  
  
7. Fazit  
 Zusammenfassung der Lernergebnisse und Ideen für weitere Anwendungen des erlernten   
 Wissens in anderen Projekten.