**Wahlpflichtmodul-Labor Wintersemester 2023/24**

**Laborbericht**

über das Projekt „Gleichgewichtstrainer“

im Modul „Internet of Things – Smart Sensor Systems”

Eingereicht durch:

Anh Quoc Nguyen  
1397466

Betreuung von Prof. Dr. Bergbauer

A logo for a university

Description automatically generated

Frankfurt University of Applied Sciences

Fachbereich 2: Informatik- & Ingenieurwissenschaften

Studiengang: Informatik (B.Sc.)

# EINLEITUNG

In diesem Projekt wird ein Fitnesstrainer entwickelt, der spezielle Funktionen und Spiele anbietet. Im Gegensatz zu normalen Fitnessuhren und Gadgets, die mehr für Bewegung und Aktiv-Leben fördern, fordert dieser Fitnesstrainer die Spieler dazu auf, komplett still zu halten, um ihre Körperspannung zu verbessern. Dieses Gadget bietet den Sportlern zwei Grundfunktionen: **„don’t move“** und **„keep your balance“.**

Bei der ersten Grundfunktion **„don’t move“** ist der Spieler verpflichtet, möglichst still zu bleiben. Jede kleine Bewegung kostet einen Punktabzug von insgesamt zehn Punkten. Das Spiel ist beendet, wenn alle zehn Punkte abgezogen sind. Während des Spiels wird die Stoppuhr automatisch aktiviert und speichert am Ende jedes Spiels nur das beste Ergebnis.

Alternativ zur ersten Funktion kann der Spieler eine beliebige Position wählen und sein Gleichgewicht halten. Bei diesem Modus hat der Spieler die Freiheit, zwischen fünf Schwierigkeitsniveaus zu wählen. Jedes Niveau hat einen Toleranzwinkel für Auslenkung. Wenn der Spieler seine Position nicht halten kann und sich innerhalb des Toleranzbereichs bewegt (nähert sich dem maximale zugelassene Ablenkungswinkel), warnt der Gleichgewichtstrainer den Spieler durch ein Tonsignal. So kann der Spieler sofort zu seiner ursprünglichen Position zurückfinden. Je höher das Niveau, desto kleiner ist die zugelassene Ablenkung und desto kürzer der Zeitabschnitt, in dem der Spieler reagieren muss. Wenn er nicht rechtzeitig reagieren kann, verliert er sofort einen Punkt.

|  |  |  |
| --- | --- | --- |
| Niveau/ Toleranz | Winkel in grad | Zeit in Sekunden |
| Noob | Von 20° bis 60° | 5s |
| Athletic | Von 20° bis 50° | 4s |
| Champion | Von 15° bis 30° | 3s |
| Legend | Von 10° bis 25° | 2s |
| God-like | Von 5° bis 15° | 1s |

Tabelle 1: Toleranz-Winkel und Zeitabschnitt für die einzelnen Niveaus



Abbildung 1: Mögliche Positionen, mit den man üben kann

# PROJEKT

## Vorstellung der Hardware

### Calliope mini Platine

Die Calliope mini Platine ist ein vielseitiges und benutzerfreundliches Microcontroller-Board, das speziell für den Einsatz im Bildungsbereich entwickelt wurde. Es eignet sich hervorragend, um Kindern und Jugendlichen die Grundlagen der Programmierung und Elektronik näherzubringen

### Übersicht

OnBoard Hardware:

Nordic nRF51822 Multi-protocol Bluetooth® 4.0 low energy/ 2.4GHz RF SoC

32-bit ARM Cortex M0 processor (16MHz)

16kB RAM 256 kB Flash

Bluetooth Low Energy

5 x 5 LED-Matrix-Bildschirm

Beschleunigungssensor, Gyroskop, Magnetometer (Bosch BMX055)

MEMS-Mikrofon

DC-Motortreiber (TI DRV8837)

Piezo-Lautsprecher

Programmierbare RGB-LED (WS2812b)

2 programmierbare Taster

Serielle Schnittstelle (USB + konfigurierbare Anschlüsse)

PWM-Ausgabe

4 Bananenstecker-/Krokodilklemmenanschlüsse

4 analoge Eingänge

8-11 Ein-/Ausgangsanschlüsse (je nach Softwarekonfiguration)

SPI + I2C

USB-Micro-B-Anschluss (Programmierung und Stromversorgung)

JST-Batterieanschluss (3.3V)

4 Bananen-/Krokodilklemmenanschluss für 3.3V (Ausgang)

2 Grove-Steckverbinder (I2C + Seriell/Analog)

NXP KL26 (USB und Stromversorgung)

Flash-Programmspeicher (optional)

#### Eingabe und Ausgabe

5x5 LED-Matrix: Zeigt Bilder und Symbole, unterstützt Programmieren und visuelle Anzeige.

Lautsprecher: Gibt Töne und Musik wieder.

Knöpfe/ Tasten: Zwei programmierbare Tasten für Befehle oder Steuerung.

Touch-Pins: Berührungssensitive Eingabegeräte.

Logo Touch-Pin: Zusätzlicher Touch-Pin im Logo für weitere Eingaben.

#### Sensoren

Lagesensor: Erkennt Ausrichtung und Bewegungen, ideal für Bewegungserkennung und Spiele.

Lichtsensor: Misst Umgebungshelligkeit, reagiert auf Lichtveränderungen.

Temperatursensor: Erfasst Umgebungstemperatur.

Kompass: Bestimmt Himmelsrichtung, unterstützt Navigation.

Funk: Drahtlose Kommunikation zwischen Calliope mini Geräten.

Bluetooth: Drahtlose Verbindung zu anderen Geräten und Sensoren, ideal für IoT-Projekte.

#### System

Grove: Schnittstellen für einfache Erweiterung mit Sensoren und Aktoren.

Batterie Anschluss: Anschluss für externe Batterie.

USB-Anschluss: Programmierung, Stromversorgung und Datenübertragung.

Prozessor: Leistungsstarker ARM Cortex M0 für Programmausführung und Steuerung.

Status LED: Zeigt Betriebszustand und hilft bei der Diagnose.

Reset Taste: Setzt das Gerät zurück, startet Programme neu oder behebt Fehler.

|  |  |
| --- | --- |
| A diagram of a computer chip  Description automatically generated  Abbildung 2: Allgemein Überblick der Calliope mini Platine | A computer screen shot of a computer chip  Description automatically generated  Abbildung 3: Pinbelegung der Calliope mini Platine |

### Programmierungsumgebung

MakeCode ist eine von Microsoft entwickelte Online-Entwicklungsumgebung, die es ermöglicht, Programme für Mikrocontroller wie den Calliope mini mithilfe einer grafischen Benutzeroberfläche zu erstellen. Sie ist besonders für Anfänger und den Bildungsbereich geeignet, da sie eine intuitive visuelle Programmierung bietet.

Der Calliope mini kann über PC, Android und iOS drahtlos programmiert werden. Nach dem Anschluss über USB oder Bluetooth wird der Calliope mini als USB-Datenträger verwaltet, auf dem Programme übertragen werden können. Diese Programme starten sofort nach dem Einschalten oder Neustart des Systems.

Unterstützte Entwicklungsumgebungen:

Calliope mini-Editor: Einfache Webanwendung basierend auf Scratch.

Microsoft MakeCode: Bietet eine visuelle Programmierung mit JavaScript, C++ und MicroPython.

Open Roberta Lab (NEPO): Eine grafische Programmierumgebung.

Vorteile von MakeCode für den Calliope mini:

Einfache Benutzeroberfläche: Drag-and-Drop-Programmierung.

Integrierter Simulator: Testen von Programmen vor dem Hochladen.

Vielfältige Programmiermöglichkeiten: Unterstützt Blockprogrammierung sowie JavaScript und Python.

Deswegen ist Makecode Blocks bzw. JavaScript für unseren Projekt ausgewählt. (Calliope, Makecode Calliope, n.d.)

A screenshot of a computer

Description automatically generated

Abbildung 4: Die Schnittstelle des Makecode

### Initialisierung des Programms im Simulator

Der Projekt wird mit dem Button „Neues Projekt“ über die Webseite von Makecode gestartet. (Calliope, Makecode Calliope, n.d.)

A screenshot of a computer

Description automatically generated

Abbildung 5: Der Startbildschirm des Makecodes

In diesem Projekt werden zwei Probleme in der Programmiersprache JavaScript gelöst.

A screenshot of a computer

Description automatically generated

Abbildung 6: Wechseln in die textbasierte JavaScript-Ansicht

## Vorstellung der Software

Im folgenden Abschnitt werden nacheinander die beiden Probleme „don’t move“ und „keep your balance“ gelöst. Jede Lösung besteht aus zwei Teilen: Flussdiagramm und Codierung im Makecode.

### Die erste Funktion „Don’t move“

#### Flussdiagram

Bevon zum Programmierteil übergegangen wird, ist es äußert wichtig, ein Flussdiagramm zur Lösung des Problems zu erstellen. Dies erleichtert nicht nur das Verständnis des Programms, sondern auch die Identifikation möglicher Fehlerquellen und die Strukturierung des Codes.

A diagram of a flowchart

Description automatically generated

Abbildung 7: Flussdiagramm von Grundfunktion "Don't move"

#### Programmierung

Um diese Funktion zu programmieren, ist es sehr wichtig, die Dokumentation zu lesen. Einige integrierte Funktionen werden verwendet, die detailliert in der Dokumentation beschrieben sind. Dies hilft, ein besseres Verständnis der verfügbaren Werkzeuge und Methode zu erlangen und ermöglicht eine effektivere Implementierung der Lösung.

Function basic.showNumber (Calliope, Zahl anzeigen, n.d.)

Function input.onGesture (Calliope, Bei Geste, n.d.)

Function basic.showString (Calliope, Text anzeigen, n.d.)

Function input.runningTime (Calliope, Laufzeit, n.d.)

A screenshot of a computer program

Description automatically generated

Abbildung 8: Calliope Mini JavaScript Programm "Don't move"

### Die zweite Funktion „Keep your balance“

# Literaturverzeichnis

Calliope. (kein Datum). *Allgemein der Calliope Mini Platine*. Von https://calliope-mini.github.io/v10/ abgerufen

Calliope. (kein Datum). *Bei Geste*. Von https://makecode.calliope.cc/reference/input/on-gesture abgerufen

Calliope. (kein Datum). *Laufzeit*. Von https://makecode.calliope.cc/reference/input/running-time abgerufen

Calliope. (kein Datum). *Makecode Calliope*. Von https://makecode.calliope.cc/ abgerufen

Calliope. (n.d.). *TECHNICAL DETAILS OF THE CALLIOPE MINI*. Retrieved from https://calliope.cc/en/calliope-mini/tech-facts

Calliope. (kein Datum). *Text anzeigen*. Von https://makecode.calliope.cc/reference/basic/show-string abgerufen

Calliope. (kein Datum). *Zahl anzeigen*. Von https://makecode.calliope.cc/reference/basic/show-number abgerufen