

# Dokumentace k semestrálnímu projektu

**Téma:**

## **Dálkové ovládání pomocí pohybu (Vzdálené ovládání pohybem)**

Projekt rozpoznává gesta ruky v reálném čase pomocí senzoru pohybu (IMU) a mikrokontroléru Raspberry Pi Pico. Každé gesto odpovídá jiné barvě LED diody (RGB). Rozpoznávání probíhá pomocí modelu strojového učení, který běží na počítači.

---

## **1. Hardwarové komponenty**

**Použité součástky:**

- **Raspberry Pi Pico WH** (s piny)
- **WaveShare 10 DOF IMU Sensor v2** (MPU9250)
- **RGB LED dioda** (společný katoda)
- **Rezistory:**  $3 \times 220 \Omega$  pro každý kanál LED
- **Breadboard (kontaktní pole)**
- **Dupont kabely** (samec-samec)
- **MicroUSB kabel** (pro napájení a UART komunikaci s PC)

**Zapojení:**

- **IMU senzor → I2C:**
    - SDA → GP0
    - SCL → GP1
  - **RGB LED:**
    - Červená (R) → GP15 + rezistor
    - Zelená (G) → GP14 + rezistor
    - Modrá (B) → GP13 + rezistor
    - Katoda → GND
- 

## **2. Softwarová struktura**

## 2.1 Složka hw\_testing/

Testovací skripty:

- imu\_reading\_test.py – ověření, že IMU funguje (vypisuje data)
- led\_test.py – test LED, barvy a PWM
- module\_finding\_test.py – ověření I2C adresy senzoru

## 2.2 Složka pico\_scripts/

- mpu9250.py, mpu\_6050.py – knihovny pro čtení z IMU
- imu\_stream.py – posílá data z IMU do PC přes UART
- main.py – finální skript na Pico: čte IMU, posílá data, přijímá povely, ovládá LED

## 2.3 Složka final\_python\_scripts/ (PC strana)

- log\_data.py – skript pro záznam CSV souborů se značkou gesta
- run\_inference.py – hlavní skript: čte IMU data z Pico, dává je modelu, pošle zpět barvu jako povel

## 2.4 Kořenový adresář:

- model.pkl – uložený model RandomForestClassifier (scikit-learn)
  - jupyter\_skripty.py – Jupyter Notebook: zpracování CSV, tvorba oken, trénink modelu
  - main\_old.py – první testovací varianta, bez ML
- 

## 3. Pořadí kroků při nasazení

### Krok 1: Zapojení hardware

### Krok 2: Test funkcí:

- led\_test.py → LED funguje
- imu\_reading\_test.py → IMU dává data

### Krok 3: Sběr dat:

- Nahrát imu\_stream.py na Pico
- Na PC spustit log\_data.py a sbírat pohyby s ručním popisem

### Krok 4: Trénink modelu:

- V jupyter\_skripty.py provést:
  - Načtení CSV
  - Tvorba oken
  - Trénink RandomForest
  - Export do model.pkl

**Krok 5: Běhové propojení:**

- Nahrát main.py na Pico
- Spustit run\_inference.py na PC
- Systém funguje v reálném čase