EESTEC Challenge 2025

Improving the PID controller for a Self-Balancing Robot

23.04.2025

EESTEC Challenge 2024

1. Optimize the PID Controller (Balance)
2. Enable Movement (Mobility)

We brought:

Sensor & actuator

Code Examples

Support Team

Task 1: Optimize the PID Controller (Balance)

Optimize the PID controller so that the self-balancing robot (SBR) remains stable in a fixed position without falling over.
The goal is to achieve precise and robust balance, even when exposed to minor external disturbances (such as nudges or a sloped surface).

Evaluation Criteria:

Stability: How well does the robot maintain its balance?

Response Behavior: How quickly and smoothly does it react to disturbances?

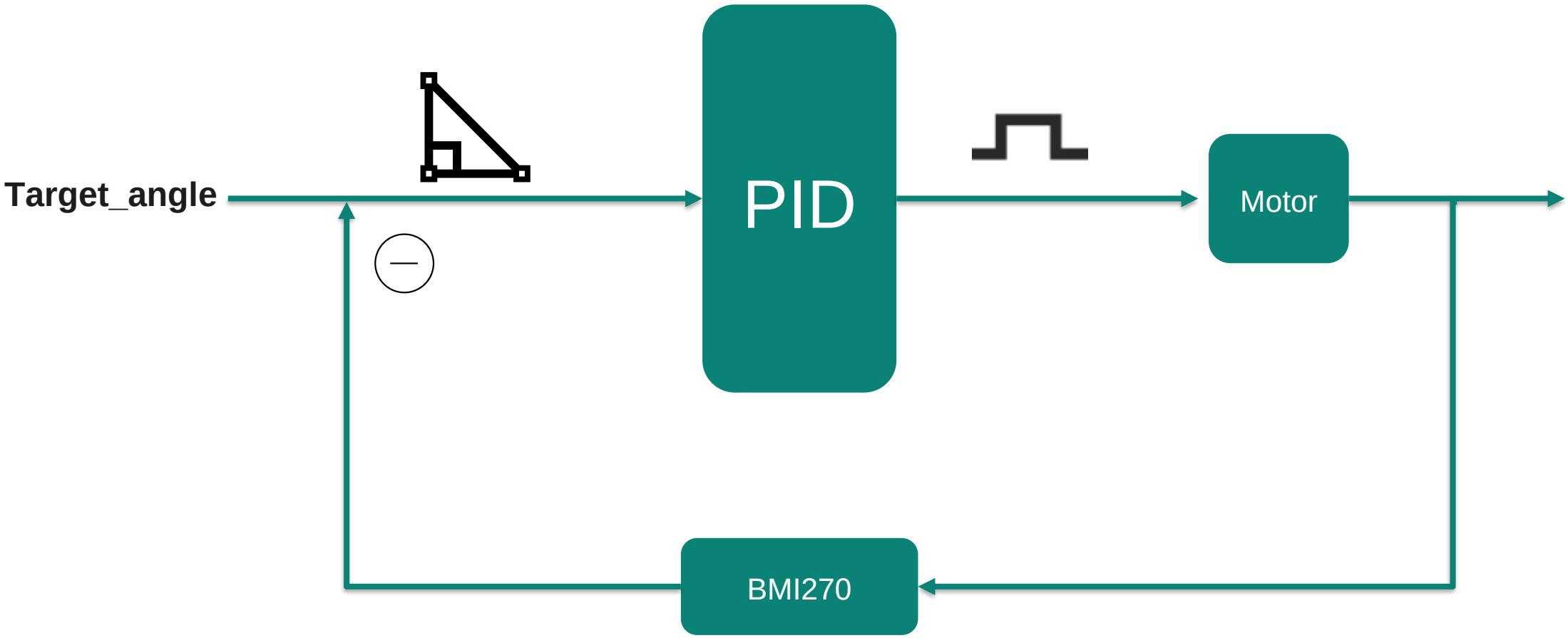
PID Tuning: Clear and well-thought-out parameterization of the PID controller.

Implementation Tips:

Start with small P and D values to observe basic behavior.

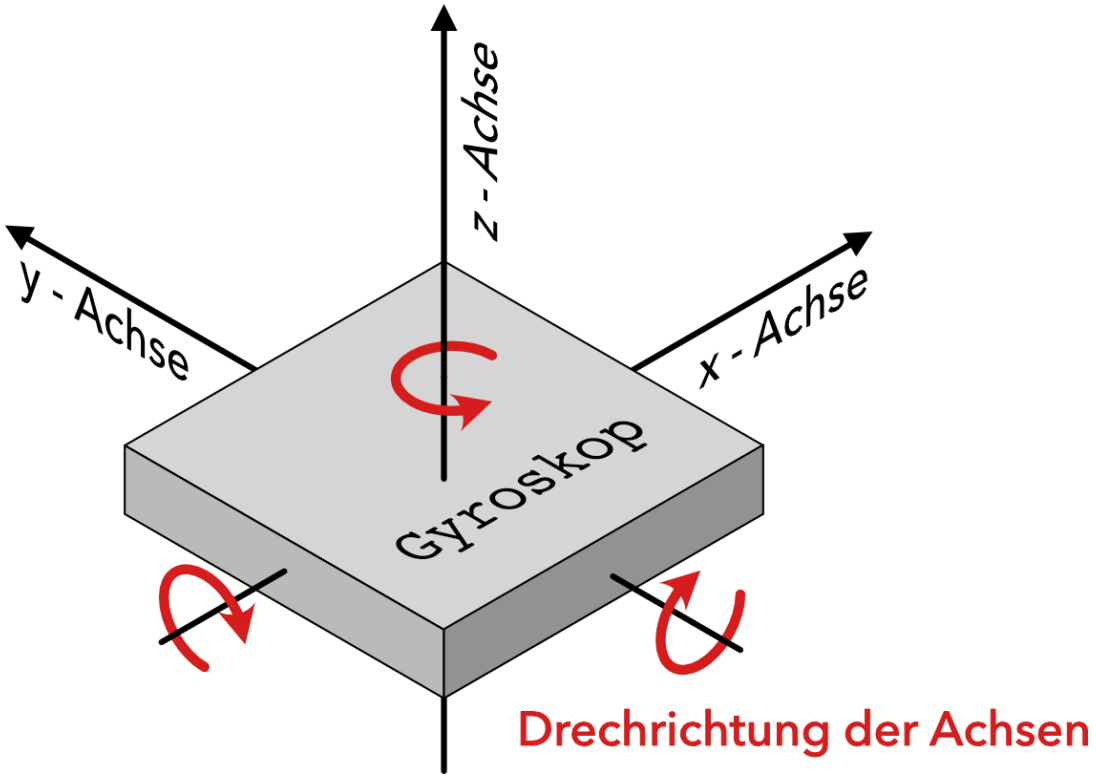
- Use logging (e.g., via serial output) to analyze the controller’s behavior.

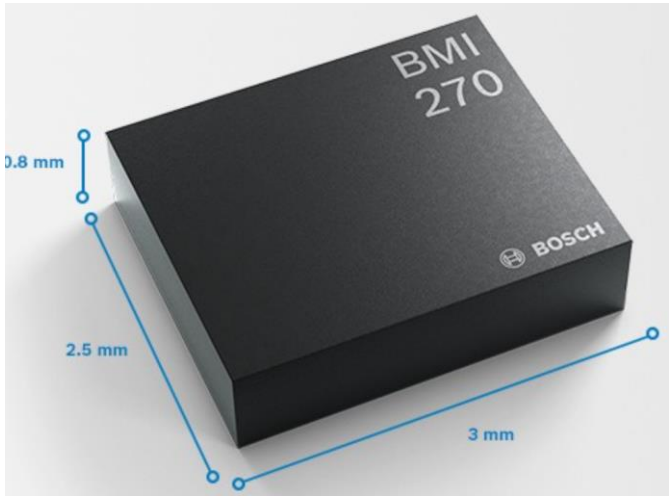
Task 1: Optimize the PID Controller (Balance)



Task 1: Optimize the PID Controller (Balance).

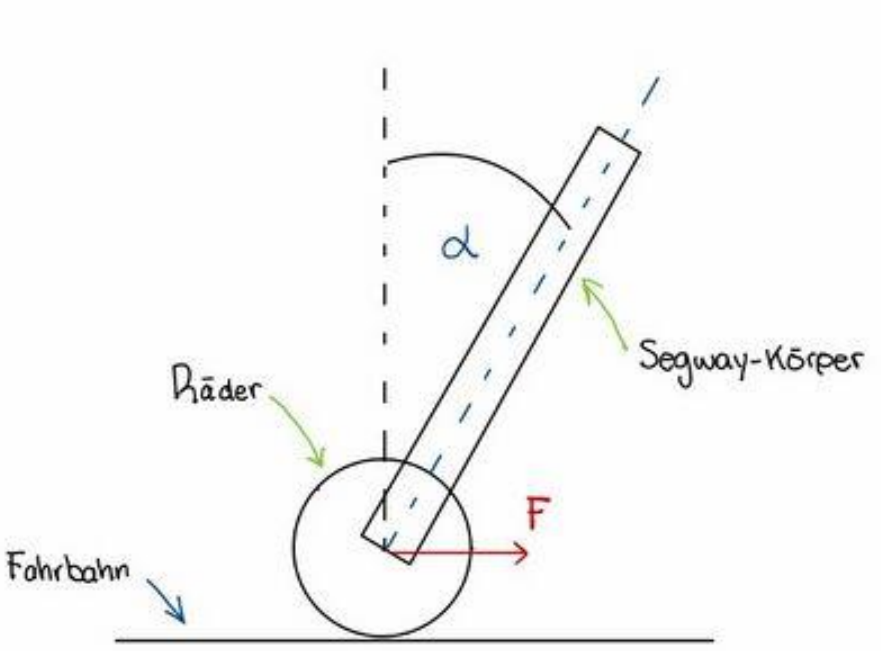
– Gyroscope Sensor: BMI270





Task 1: Optimize the PID Controller (Balance)

– Gyroscope (BMI270) feedback values: ax,ay,az, rx, ry, rz



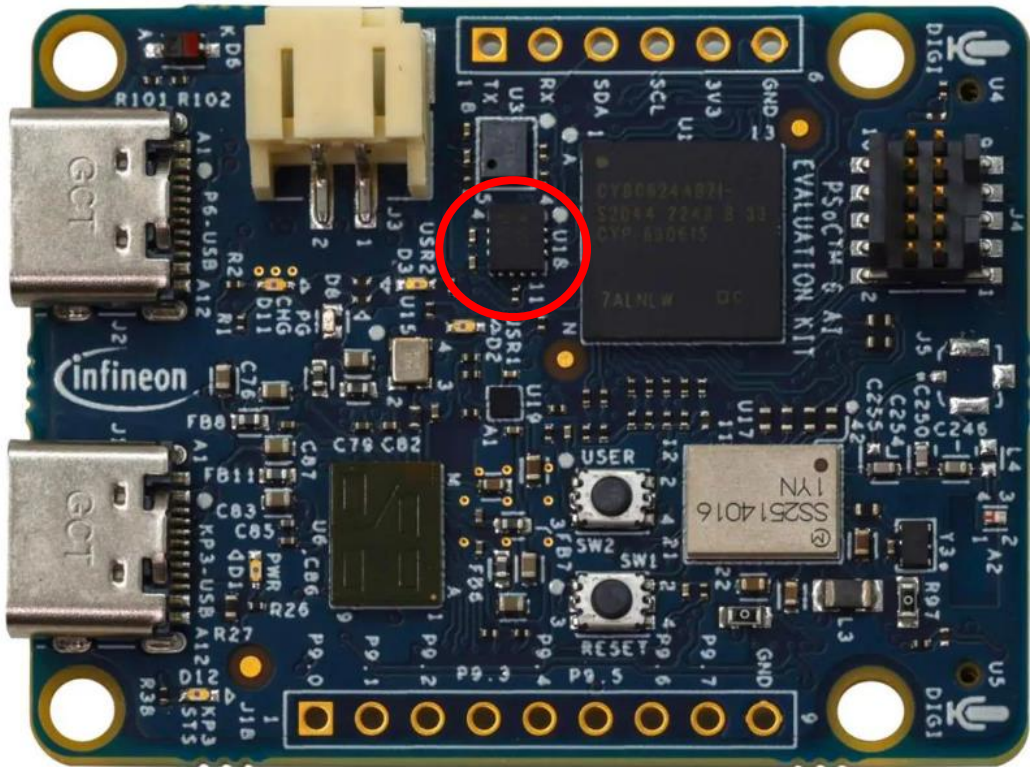
Task 1: Optimize the PID Controller (Balance)

- Motor controller: TLE94112

- Multi-Half-Bridge
- 12 Half bridges

Task 1: Optimize the PID Controller (Balance)

– CY8CKIT-062S2-AI



Task 2 Enable Movement (Mobility)

Extend the robot so that it can move safely and in a controlled way—forward, backward, and in curves—while maintaining its balancing ability at all times.

Evaluation Criteria:

Mobility: Clean, repeatable movements in all directions.

Balance During Movement: No falling over or excessive wobbling.

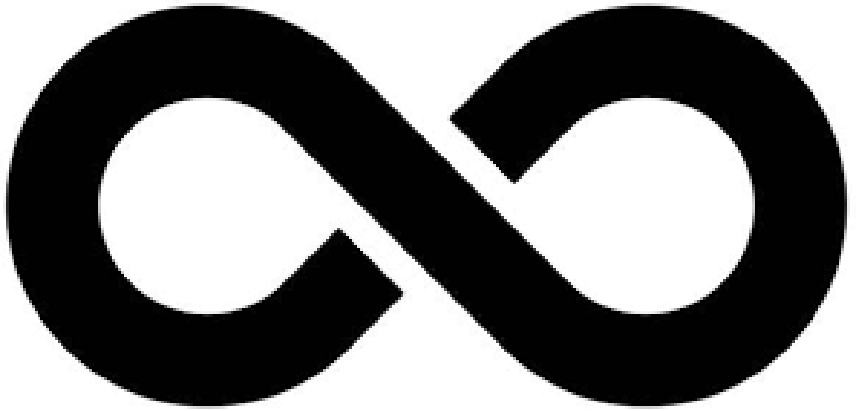
User Control (optional): Possibility for manual control (e.g., via remote control or app).

Implementation Tips:

- Ensure that setpoints (e.g., tilt angle or target position) are dynamically adjusted.

Task 2 Enable Movement (Mobility)

– Move forward, backward, left, right



Some Hints

- You may use any code you find in the Internet, but please provide the source
- You can use Chat GPT, Copilot, ….
- Helpful links:
  https://github.com/Infineon/arduino-core-psoc6
  https://www.hackster.io/Infineon_Team/psoc-6-meets-arduino-the-ultimate-maker-combo-9fc127
  https://www.bosch-sensortec.com/products/motion-sensors/imus/bmi270/
  https://github.com/Infineon/multi-half-bridge

Let‘s Get Started!

github.com/Infineon/hackathon

Submission

- Submission happens on **Google Drive.**
  - You will receive an individual upload link for your team from EESTEC

Deliverables

- A **technical article**, including:
  - **Idea** description
  - Reproducible step-by-step guide on **technical implementation**
  - Overview of **project outcome**
  - **Outlook** for future improvement
  - Format can be **Markdown** or **PDF**
  - Including **graphics/pictures** is appreciated
- Your documented **code**
  - As ZIP archive in Google Drive **or**
  - Private repo on **GitHub** -> invite julian.eder@infineon.com and ericjoerg.schulze@infineon.com
- A short presentation (5 minutes)
- Your **Feedback** about the challenge: **What did you like? What would you do differently?**


Infineon