Linnæus University
Sweden

Assignment 3

Fault tolerance of Embedded System

Assignment 3

Author: your surname, your name
Supervisor: Mehdi Saman Azari
Semester: HT24
Course code: 2DT303

Deadline:

The deadline to submit the assignment is **20 Oct 2024**.

Rules:

1) You have to submit a report for Assignment 3, with the LNU template.
2) You have to submit a file in ***.pdf*** format.
3) You **must** use the LNU template for your report. You are allowed to use this file or a new one. In both cases, the student has to adopt the LNU template.
4) The file you submit **must be renamed** as follows <ASGM3 -surname_name.pdf>
5) Deadline is **20 Oct 2024**. Each day of delay over the submission deadline will cause a 2 points penalization.
6) The evaluation for this assignment is in points. Maximum points 50pts.

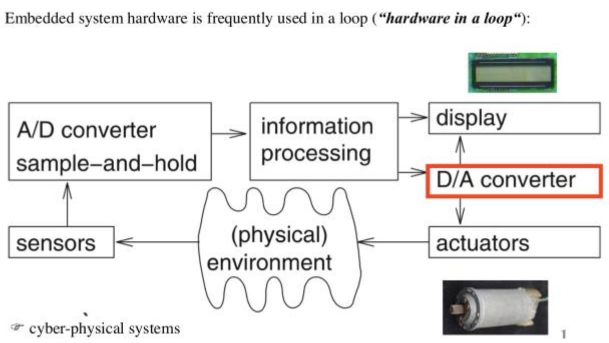
Exercise 1 (50 points)

The objective of this assignment is to investigate and understand the fault tolerance capabilities of the Raspberry Pi Pico microcontroller, identify potential failure scenarios, and develop strategies to mitigate or recover from these faults.

Initial phase:

- Conduct an initial Raspberry Pi project that includes hardware in a loop, featuring at least one sensor and one actuator. Please note that the complexity of the baseline project will influence the execution of this assignment and will also affect the final grade.

Embedded System Hardware



Embedded system hardware is frequently used in a loop (***"hardware in a loop"***):

☞ cyber-physical systems

- Ensure that the Raspberry Pi Pico functions correctly under normal operating conditions.

Project Tasks:

Task 1: Introduction to Fault Tolerance

a) Provide a brief overview of fault tolerance, its importance in real-world applications.
b) Explain the concept of fault tolerance in embedded systems, specifically in the context of Raspberry Pi Pico.
c) List the common types of faults that can occur in hardware in loop component, especially in embedded systems.

Task 2: Identifying Fault Scenarios

In Task 2, you need explore potential fault scenarios that could affect your baseline project based on Raspberry Pi Pico, aiming to gain a deeper understanding of the various risks and vulnerabilities that an embedded system may encounter. This task is designed to encourage you to think critically about the potential challenges in fault tolerance and prepare you for developing relevant strategies in the subsequent tasks.

a) **Brainstorm Fault Scenarios**: you should brainstorm a list of fault scenarios. These scenarios should encompass a wide range of potential issues that could affect the operation of your baseline project. Examples include:
   - Power supply interruptions or voltage fluctuations.
   - Code errors, such as infinite loops or software crashes.
   - Hardware faults like a malfunctioning sensor or actuator or a broken connection.
   - Environmental factors like extreme temperatures or humidity.
   - Communication failures with other devices or network issues.

b) **Describe Each Scenario:** For each identified fault scenario, you should provide a detailed description, explaining what the scenario entails, its potential consequences, and how it might impact the operation of the Raspberry Pi Pico.

c) **Prioritize Scenarios:** Once you have a list of fault scenarios, you should prioritize them based on factors such as likelihood, severity of impact, and relevance to their specific application. This step helps in focusing on the most critical scenarios for further analysis.

By the end of Task 2, you will have a comprehensive list of fault scenarios, with detailed descriptions and an understanding of the potential impact of these faults on your Raspberry Pi Pico projects. This groundwork will be essential for developing fault tolerance strategies in Task 3.

Task 3: Developing Fault Tolerance Strategies

In Task 3, you need focus on developing strategies to address the fault scenarios that identified in Task 2. you need explore various fault tolerance mechanisms that can be applied to the Raspberry Pi Pico and decide on a strategy to implement for one specific fault scenario. This task encourages creative problem-solving and hands-on experimentation.

a) **Review Fault Scenarios**: Begin by revisiting the list of fault scenarios identified in Task 2. Select two specific scenario to work on during this task. Ensure that the chosen scenario is relevant to the project or application under consideration.

b) **Research Fault Tolerance Mechanisms:** you should propose suitable fault tolerance mechanisms for embedded systems. Some common mechanisms include:

- Code redundancy: Implementing backup or redundant code paths.
- Error handling: Implementing error-checking and error-handling routines in the code.
- Backup power sources: Integrating backup power supplies (e.g., batteries or supercapacitors).
- Watchdog timers: Using hardware or software timers to monitor system health.
- Sensor redundancy: Employing multiple sensors for the same measurement.

c) **Resource Requirements:** Identify any additional components or resources required for implementing the chosen strategy (e.g., extra sensors, capacitors, hardware watchdog timer, etc.). Ensure these resources are readily available.

By the end of Task 3, you will have a clear understanding of the fault tolerance strategy they plan to implement and a structured plan for its integration into the Raspberry Pi Pico project. This will pave the way for practical implementation and testing in the subsequent tasks.

Task 4: Practical Implementation

In Task 4, you need put your chosen fault tolerance strategy into action by implementing it on the Raspberry Pi Pico. This hands-on task provides an opportunity for you to gain practical experience in incorporating fault tolerance mechanisms into embedded systems.

a) **Code Implementation**: you need write or modify the code to integrate the selected fault tolerance strategy into their Raspberry Pi Pico project. This may involve incorporating redundancy, error handling, watchdog timers, or other mechanisms, as chosen in Task 3.

b) **Hardware Integration**: If the chosen fault tolerance strategy involves additional hardware components (e.g., backup power sources, sensors), you should connect and integrate these components with the Raspberry Pi Pico.

c) **Testing**: After implementing their chosen strategies, you should carry out tests to evaluate the effectiveness of their fault tolerance mechanisms.

d) **Analysis**: you should document the results of the tests and analyse how well their strategy mitigated or recovered from the fault.

By the end of Task 4, you will have successfully implemented your chosen fault tolerance strategy on the Raspberry Pi Pico.

Task 5: Report

you should prepare a report that summarizes your findings and experiences based on the order of tasks.

# Evaluation Criteria

This assignment will be assessed based on the following criteria:

- Complexity of the baseline project.
- Creativity and effectiveness of fault tolerance strategies.
- Quality of implementation.
- Comprehensive testing and evaluation.
- Clarity and organization of the report.

# Notes

- This assignment can be completed individually or in a group (maximum of two participants).