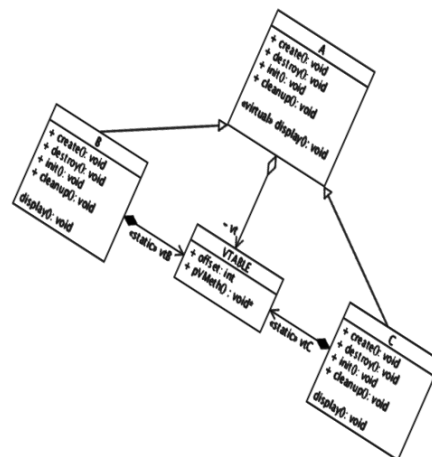


# A Software Engineering Semester Project with an Embedded System

Medard Rieder

Thomas Sterren

School of Engineering - Infotronics, Inf2 / SS 2023



## Introduction

This document is the task setting for your semester project in Informatics II. The task combines the application of the knowledge you have gained in the Informatics II course:

- Imperative programming using C / C++
- Object- oriented programming using C / C++
- Software modelling using UML
- Software patterns for embedded systems
- Stateful programming using C / C++
- Execution framework

The task will also bring some new knowledge which is:

- Application of C / C++ programming on an embedded system
- Using the PICEBS development board with a PIC18F87K22
- Testing of firmware

## System description

The PICEBS board has 3 buttons and a led bar with ten leds. You have to develop a firmware, the responds to the following specifications:

- Click the first button: led1 starts blinking
- Click the first button again: led1 stops blinking
- Click the second button: led10 starts blinking
- Click the second button again: led10 stops blinking
- Double click the first button: led1 blinks three times
- Double click the second button: led10 blinks three times
- Long click either the first or the second button: led1 and led10 start blinking
- Long click either the first or the second button again: led1 and led10 stop blinking

## Task description

### **Analysis**

- Develop a physical (deployment) diagram of the above specified system with the nodes  $\mu$ Controller, 2 leds and two buttons.
- Develop a use case diagram with the use cases click, double click, and long click.
- Develop the sunny day sequence diagrams for each use case. Use the objects user, buttonX, controller, and ledX. buttonX means either 1 or 2. ledX means either 1 or 10.

### **Design**

- Define packages as relevant for this system.
- Derive classes and methods from the sequence diagrams.
- Develop a class diagram with classes, packages, and relations among classes. For each class, give its attributes and methods.

### **Implementation**

- Develop the state machine of the Button class. It must be a polling state machine. This means the buttons do not generate interruptions, but the state machine is reading the button state each 20 ms or similar.
- Develop the state machine of a class which handles click, double click, and long click.
- Develop the state machine of the main controller class.
- Implement all classes in object-oriented C for the PIC18F87K22  $\mu$ Controller.

### **Test**

- Define a test prescription for the click scenarios.
- Define a test prescription for the double click scenarios.
- Define a test prescription for the long click scenarios.
- Run the tests.
- Document the test results. Make conclusions on what should be changed or optimized.

## Deliverables

The following deliverables are required:

- All UML diagrams.
- The MPLAB project.
- The test prescriptions and results.

Diagrams and test stuff are delivered exclusively in pdf format. Pack your entire deliverables in a zip file with the name: namePrenamSemProj.zip. A delivery will be opened in Cyberlearn as usual. Do not miss the deadline. We will not accept any deliveries by email.