November 24, 2016

Thanh Huynh – 2688093

Bilal Butt – 2688700

Michel Clerger – 2694646

Zirui Song - 2652528

GROUP MEMBERS

process report

# General plan and working environment set up

We are planning to do this project basing on the suggestion on LAB Manual. However, at the first week, we experience more with the class diagram and state diagram. Every week, the tasks for individuals will be set up.

|  |  |
| --- | --- |
| week | delivery |
| 1 | (start up)  Test First demonstration  Class diagram (basic)  State diagrams (basic) |
| 2 | Project Initiation Document (including detailed planning) |
| 3 | Specification & implementation Hardware class |
| 4 | Software system design:   * class diagram * state diagrams * sequence diagrams |
| 5 | intermediate demo:   * connection Arduino + Centipede + Laundry Machine demonstrate that the Arduino controls all Laundry Machine features, and that the Arduino monitors all Laundry Machine events * unit tests |
| 6 |  |
| 7 | intermediate demo:   * unit tests |
| 8 |  |
| 9 | final delivery:   * product presentation + demonstration * process description * project documentation |

To sharing and manage code and documents, GIT is used. The repository is set up at: <https://github.com/bilal614/ESP>

# Week 1:

## Group work:

Generally discuss about the group working environment. Investigating the tools for drawing the diagrams such as Microsoft Visio, Astah professional and online tool called https://www.draw.io/. The draw.io is selected since it is an online tools and free, therefore; it is more convenient to use without any installation required

We mainly work on designing the main state diagram and attributes individual tasks.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Use draw.io to draw all components of class diagram | Thanh | Done |
| Doing description of the class | Bilal | Done |
| Doing the description for the state diagram and keep track of status document/group discussion. | Michel | Done |
| Redraw the state diagram on draw.io tool | Zirui | Done |

## Assumptions:

We assume that first the user has to select the program that he wants then he can do the payment.

## Questions:

- What should we do for the specification & implementation of Hardware class?

- Should we have a general state diagram or should we also have a very detailed state diagram for each

- Should we have an end state? Or after the main wash it can just go to initial state?

- Should the association between the LaundryMachine class, ProgramExecutor and Program setting are composition associations?

# Week 2&3:

## Group work:

We discuss about some doubts that we had about the interfaces. Then we make sure that everyone have the hardwares that are necessary for our work. We separate the workload between each other.

We mainly work on designing the main state diagram and attributes individual tasks.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Implementation and testing of interfaces called “IProgram” and “ICoin” | Thanh | Done |
| Implementation and testing of interfaces called “ILock” and “ISoap” | Bilal | Done |
| Implementation and testing of interfaces called “IWater” and “IMotor” | Michel | Done |
| Implementation and testing of interfaces called “ITemperature” and “IBuzzer” | Zirui | Done |

## Questions:

- Question are indicated on the design document.

# Week 4:

## Group work:

We discuss about what we want to include in the Class Diagram and which additional variable might be necessary for our classes. We have decided to work for now on three classes: ProgramSelect, ProgramSettings and CoinWallet. We start discussing on the ProgramExecutor class and what will be the function of each function that it contains. In the end, we separate the workload between each other.

We mainly work on designing the main state diagram and attributes individual tasks.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Implementation of the Final Class Diagram and work on the Class CoinWallet | Thanh | Done |
| Work on the Class ProgramSettings | Bilal | Done for program indicator and cost |
| Modified IWater and IMotor | Michel | Done |
| Modified Temperature and IBuzzer | Zirui | Done |

## Questions:

- Do we have to display the amount of money left? If so, how can we indicate a remainder of 40, if we have only 3 LED of 10 coin?

- What is expected from the InstallStartHandler() function?

# Week 5:

## Group work:

We discuss about what the intermediary classes. We defined the intermediary classes that we will need, and then we define the different functions that will be needed to make those classes correspond to the requirements of the washing machine. In the end, we separate the workload between each other, everyone should implement the intermediate classes that relates to the interfaces he/she previously created.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Implementation the intermediate class related to the interfaces called “IProgram” and “ICoin”, and update the Class Diagram | Thanh | Done with ICoin  Working on IProgram |
| Implementation the intermediate class related to the interfaces called “ILock” and “ISoap” | Bilal | Done |
| Implementation the intermediate class related to the interfaces called “IWater” and “IMotor” | Michel | Done |
| Implementation the intermediate class related to the interfaces called “ITemperature” and “IBuzzer” | Zirui | Doing or Done? |

## NOTES:

- For the washing machine, should it be able to return change or not?

- Enumeration type for the program? Where should it be? It could be in the ProgramSettting?

- Is it possible to declare an object of interface class in C++? ILock lock or ILock \*lock? We are planning to do it in the pointer way.

- What does the ‘add soap 1’ mean? Is it different from the ‘closed soap compartment’?

- Why in the ProgrameSelect Constructor we need to be provided a pointer to IProgram in its argument

-Why is there an output pin for lock when there is no corresponding LED or other hardware component for it?

# Week 6:

## Group work:

* Compiled CoinWallet, ProgramSelect and Soap and Door via the Program executors and debugging the problems
* Discuss about InstallStartHandler to get the working concepts of it.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Implementation the IProgame class. Investigate and implement the StubHardwareControl.cpp functions used for testing the CoinWallet Class  Sequence diagram for IsReady() function in ProgramExecutors  State diagram for CoinWallet class | Thanh | Having problems with setting environments for running the test.  Done with the rest of tasks |
| Work on the ProgramExecutor class  Making descriptions of Soap and Lock classes | Bilal | Doing ProgramExecutor  Done with description |
| Continuing the intermediate classes Motor and Water. Introducing state diagram in case of it is necessary.  Making descriptions of Motor and Water class | Michel | Done |
| Continuing implement the ProgramSetting classes.  Making descriptions of Temperature and Buzzer classes. | Zirui | Done |

## NOTES:

- Try to install Codeblocks to run the unit test on Window environment. However there was a compile a error when doing the compilation is done. This problem is solved by install mingw.org <http://www.mingw.org/category/wiki/download>

- Then it can be compiled, but some data types like char or byte are not recognized by Codeblocks even the compiler is selected is AVR GCC

- When we try to run functions of CoinWallet class, the soap and lock together in the program executor via the main loop(). It turns out some unknown behaviors.

# Week 7:

## Group work:

* Compiled Motor, Temperature, Water into ProgramExecutor
* Debugging integrated problems.
* Testing of final product.

## Tasks division for implementation

|  |  |  |
| --- | --- | --- |
| Tasks | Responsible members | Status |
| Changing the base code’s data type to make sure it works for basic environment.  Continuing on the google test: StubHardwareControl, CoinWalletTest and ProgramSelectTest.  Updating the changes on the design documents: class diagrams, google test | Thanh | Done |
| Work on the ProgramExecutor class  Finished documentation for ProgramExecutor class  Worked on design document major decisions reflection part | Bilal | Done |
| Debug and fix Motor and Water.  Work on ProgramExecutor class, implementing and testing the different programs recipes. | Michel | Done |
| Debug and fix the temperature polling problems | Zirui | Done |

## NOTES:

- Introduce the static variable in the hardware class to keep track the sharing data between the controlling of LEDs for coin 200 and the soap 2.

-Do not use the Install handler in the final implementation.

# PERSONAL VIEWS

## Thanh

This project is a good opportunity for me to experience my knowledge in C/C++ together with an embedded device. It gave me a chance to understand how to organize an embedded system project. In previous blocks, when I worked with Arduino environments, the code was put in only one file and sometimes I divided them into several files but I never put them into separated classes for hardware handling and intelligent logic as the one we worked on this project. I realize that separated the code for handling the hardware and the code for handling the working principles of the program itself will make the debugging process become easier. For example, when we know for sure that the hardware working well by building up a stub-hardware as we’ve done in google test, then if there are errors, I will know for sure that it comes from my code.

During the implementation of google test, I had problems due to some data types is only introduced by Arduino environments was not able compiled by the other's platform. It gave me the experience to build up the code, it is more useful to use the standard data types introduced by the programming languages than the platform data types. It will reduce the risks when cross-compiler is used.

Also, from the experience of OOD, I had the feeling of there is a big gap between design phase and implementation phase. In OOD project, the design phase took almost the time of the projects and then when the implementation was taken place, it was not corresponding that much to the designs. Therefore, at this project, we started with the implementations sooner. However, it seems to me that sometimes I missed the overview about how things should be coded in the correct and how communications are done between classes when I built my code. Besides, if the design phase is done better, the time of implementations can be reduced. In general, I am wondering about how to reducing the time of design phase but still get the benefits of them to improve the implementation and also use the design document for maintaining the code.

Finally, this project also gave me a chance to practice my team-work skills. I would like to thank all my team-mates for all of their effort and Mr.Sanchez for helping us throughout this project with clear explanations and useful feedback.

## BiLAL

I thoroughly enjoyed doing this project with this group and we all made a very good sustained effort throughout the course of this project. The task at first seemed a little daunting, however as we progressed through all the small steps leading towards our end goal, the vision of the project started to become clearer.

In the previous few blocks we had been acquiring knowledge with respect to C/C++ and embedded devices and this project was a culmination of all that knowledge. I think this project showed us the potential and possibilities of the application of all that knowledge. For me personally that was quite a remarkable exercise to witness and quite satisfactory at the same time.

Also throughout the course of the project we had agreements and disagreements about decisions regarding various planning and implementation details. I think I was privileged enough to work with a set of people who were passionate about the work. Everybody was expressive of their ideas and wanted to incorporate them into the final project. In the end we came up with an amalgamation of all our ideas which resulted in a project that was better than it would have been otherwise. I appreciate the effort that all my team-mates put in this project and am grateful for Mr. Sanchez’s guidance throughout.

## MICHEL

Firstly, this project was an awesome experience; at the beginning, I was a bit skeptical about whether I would be able to manage it properly, but on the way, I realize that it wasn’t that hard if we put ourselves to it, and that also makes me realize that I possess now enough knowledge to build some nice programs.

Secondly, this project also allowed me to realize the importance of some clear and efficient programming. It also showed me the importance of the class diagrams, it also showed me how efficient and useful the functions are in programming, and it allow me to experiment the way that we can incorporate our programs to different pieces of hardware.

Finally, it was a nice experience to be able to work together with my classmates and being able to see the result of all our efforts together. It allowed me to experience more what it means to be working in a group, and to realize everything that come with it.

I will conclude by congratulating all my teammates for their efforts, and by thanking also our teacher Mr. Sanchez, who always tried to make the assignments as clear as possible to us.

## ZIRUI

At the very first when the project is starting with a little bit of chaos, we spent weeks on working with hardware classes and interfaces. It was still completely not like a laundry machine then. However, as the project proceeds, the whole thing becomes clear. It is more or less like that the program of laundry machine is finished very suddenly, with great pleasant and accomplishment of us.

Although I work less than other three teammates, I still work hard on my tasks. By doing that, I have realized the classes and interfaces more profoundly, which is very important because C and C++ is most used in embedded programming. Also because I work less than other three teammates, sometimes I couldn’t catch up or demonstrate something in the group meetings. They are the weaknesses of my personal skills but I didn’t well improved. It would be better if I had more task I did. Nevertheless, I still think it is a very good experience of college studying. My teammates are the most brilliant students in Fontys and I am very glad for working with them. So thanks for the works of all of you and the helps from the teacher Mr. Ravelo Sanchez.