**Lab Manual**

Period: P8

Module: ESP (Embedded System Project)

Author: Freddy Hurkmans

Date: November 12, 2016

(Update: Jesús Ravelo / November 2016)

**Table of Contents**

[1 Introduction 1](#_Toc360630012)

[2 Schedule 1](#_Toc360630013)

[3 The laundry machine (LM) 1](#_Toc360630014)

[3.1 Software design 2](#_Toc360630015)

[3.2 Unit tests 3](#_Toc360630016)

[3.3 Flat cable 4](#_Toc360630017)

[3.4 Centipede 7](#_Toc360630018)

[4 Requirements 7](#_Toc360630019)

[4.1 Washing programs 7](#_Toc360630020)

[4.2 More requirements 8](#_Toc360630021)

[5 Simulator 9](#_Toc360630022)

# Introduction

This document gives additional information about the washing machine project ProC++.

# Schedule

|  |  |
| --- | --- |
| **week** | **delivery** |
| 1 | (start up)  Test First demonstration |
| 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 |

## Advances and final submission

Before the final submission, two partial advances must be submitted:

* 1st advance, at the end of week 3, for 20% of the final mark;
* 2nd advance, at the end of week 5, for 30% of the final mark.

The final submission will then hold for the remaining 50% of the final mark.

In all three submissions, apart from all the regular contents that the team will present in a project (advance/final) document, there must be a clear explanation of how work/responsibilities were divided among the team members.

The schedule presented above must be used as a guide, but each team can propose variations to it and discuss them with the teacher.

# The laundry machine (LM)

When you study the requirements as listed below, you will realize that this system is very suited for an object oriented approach. Make sure that your system design is properly written in UML, and that you apply the object oriented techniques optimally.

## Software design

In this paragraph, a possible UML diagram is given of the application. Of course, other solutions are possible, perhaps better than this one.

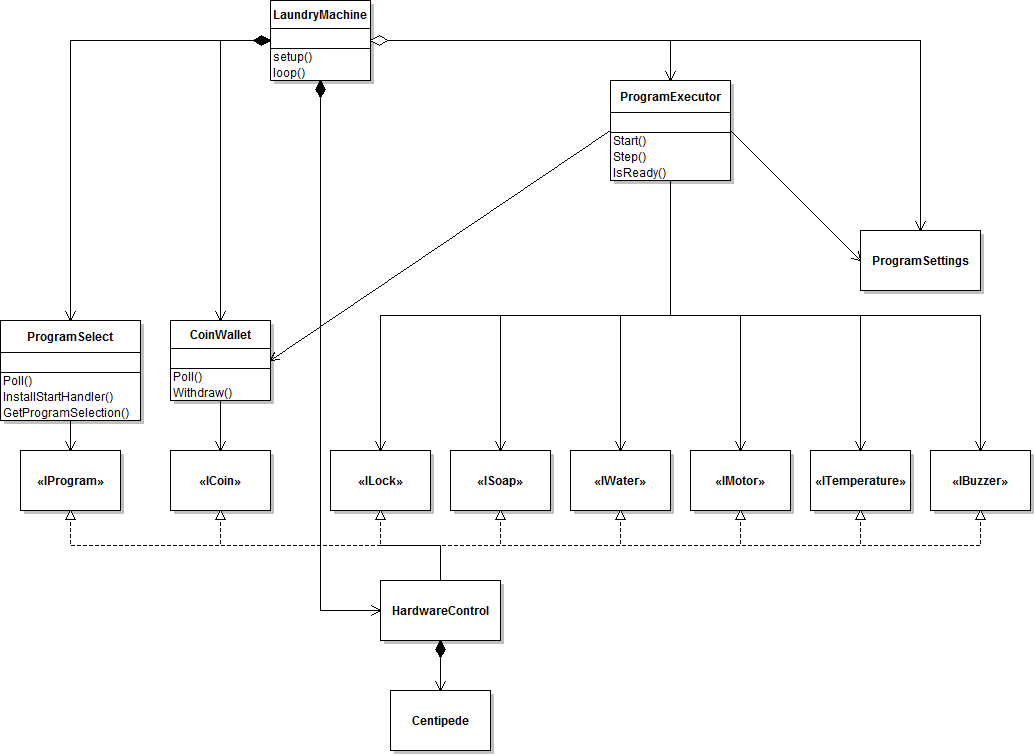
You can realize your project with this class diagram, or you can design your own class diagram. On SharePoint, template classes are available.

The software of the control unit is organized in the following way:

* the business logic where all behavior is realized (washing programs, safety);   
  this business logic contains several classes:
  + LaundryMachine  
    the main loop of the application (because there is no real operating system)  
    it polls the classes ProgramSelect and CoinWallet to see if something has happened, and it lets the class ProgramExecutor perform one step (e.g. 1/10 of a second) of a washing program
  + ProgramSelect  
    when polled, it checks if the program select button has been pressed (and if yes, it updates the program indicator leds (A, B, or C)); furthermore, when the start button has been pressed, the installed StartHandler will be called
  + CoinWallet  
    when polled, it checks if a coin button has been pressed (and if yes, it updates the coin indicator leds)  
    when a withdrawal has been done, the coin indicator leds are updated accordingly
  + ProgramExecutor  
    it executs (in steps) a complete washing program, with given settings
  + ProgramSettings  
    only containing settings for an actual washing program
* IProgram, ICoin, ILock (etc, see class diagram)  
  interfaces for dedicated parts of the hardware
* the HardwareControl class, where the hardware can be controlled on a high level (like setting the motor speed or rotation direction)

The HardwareControl class can be implemented in two different ways: either to access a real hardware LM, or a simulator. In this way, you can develop and test your business logic application on a PC which increases the development speed; especially when you don't have access to a LM all the time.

The real hardware is controlled via the Centipede class which accesses the Centipede shield, which controls the IO-lines of a flat cable (see the paragraphs below).



## Unit tests

The SharePoint project has been prepared for Unit Tests, currently running in Windows (with CodeBlocks) and Linux (with make). It is expected that the system can easily be made operational in other environments.

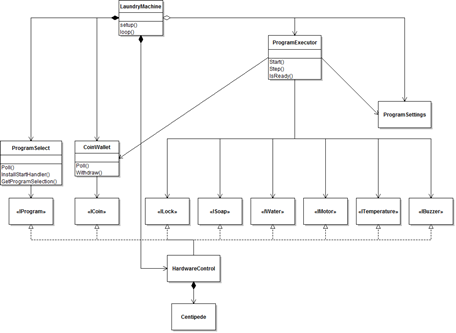
It contains only one test for one class (CoinWallet). Other tests for CoinWallet, and tests for other classes need to be implemented in your project.

Google Test is used for the unit tests.

A schematic view of the test harness:

Google

Test



StubHardwareControl

TestCoinWallet

*TestProgramExecutor*

*TestProgramSelect*

As you can see, some classes are replaced in this test setup (compared with the previous class diagram).

Description of the files (directory Test):

* TestCoinWallet.cpp: implementation of unit tests for class CoinWallet
* (at this moment absent) TestProgramSelect.cpp, TestProgramExecutor.cpp: to be created in a similar way when the corresponding classes need to be tested
* StubHardwareControl.cpp: stub of the class HardwareControl (you can also use mocks)
* ArduinoWrapper.h, DeviceUnderTest.cpp: wrappers for the .ino files (wrappers are needed to compile ino files in CodeBlocks)
* Google Test (directory gtest): several header- and source files to offer the unit test functionality. Normally, you don't need to access them
* LaundryMachineTest.cbp (directory CodeBlocks): project file for compiling an executable (for Windows)
* Makefile (directory Linux): makefile for compiling a Linux binary

## Flat cable

The control unit *controls* the LM in the following way:

* lock/unlock the door
* open/close the water drain
* open/close the water sink
* heating the water
* start/stop the motor, clockwise and anti-clockwise, at different speeds

The control unit *monitors* the LM for the following issues:

* locked door
* open drain
* closed soap compartment
* motor status
* water temperature
* water level
* program indicator
* insertion of new coins

At a low level, detailed information about the pinning can be found in the schematics document. Note: maybe you have to do your own investigations to make everything clear.

|  |  |  |  |
| --- | --- | --- | --- |
| **pin** | **name** | **remarks** | **in/out (from control unit's point of view)** |
| 1 | group2 | see below (I) | out |
| 3 | group1 |
| 5 | strobe |
| 7 | keyselect | see below (II) |
| 9 | buzzer |  |
| 11 | heater |  |
| 13 | speed2 | 11=off  10=low speed  01=medium speed  00=high speed |
| 15 | speed1 |
| 17 | dataC | see below (I) |
| 19 | dataB |
| 21 | dataA |
| 23 | motor L/R | 0=turn left  1=turn right |
| 25 | soap1 |  |
| 27 | sink |  |
| 29 | drain |  |
| 31 | lock |  |
| 33 | water2 | 00=empty  01=33%  10=66%  11=100% filled | in |
| 35 | water1 |
| 37 | temperature2 | 00=cold  01=cold  10=medium  11=hot |
| 39 | temperature1 |
| 41 | in3 | see below (II) |
| 43 | in2 |
| 45 | in1 |
| 47 | in0 |
| 49 | 5V |  | |
| remaining pins | ground |

Due to hardware limitations, some lines are multiplexed. The usage is described in the remarks below.

Remarks:

1. with the group and data bits, various indicators can be set  
   it is only handled by the LM when the strobe line is inactive (for at least 80ms) and immediately active (for at least 10 ms)  
   see the following table:

|  |  |  |
| --- | --- | --- |
| **group2+group1** | **dataC+dataB+dataA** | **indicator** |
| 00 | 001  010  100 | coin 10 #1  #2  #3 |
| 01 | 001  010  100 | coin 50 #1  #2  #3 |
| 10 | 001  010  100 | coin 200 #1  #2  soap 2 |
| 11 | 001  010  100 | program #1 (A)  #2 (B)  #3 (C) |

1. with the keyselect, different status information of the LM can be read, according to the following table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **keyselect** | **in3** | **in2** | **in1** | **in0** | **remarks** |
| 0 | doorlock | soap2 | soap1 | pressure | switches |
| 1 | coin 10 | coin 50 | coin 200 | start | push buttons |
| clear | clear | clear |  |
| program |  |  | program |

Pressing the three coin buttons at the same time is the same as pressing the clear button. Also, start and coin 10 is the same as pressing the program key.

## Centipede

With the Centipede shield, the Arduino has enough IO-lines to control the LM.

More information about the Centipede can be found at <http://docs.macetech.com/doku.php/centipede_shield> (the contents can also be found on Sharepoint). Before you can use the Centipede shield, extract the Centipede library and copy it to the Arduino\libraries directory and restart the Arduino IDE.

In Sharepoint, there is a sample Arduino sketch (BasicLaundryMachine) which uses the Centipede shield to control the LM (of course not all functionality is implemented; the rest is left for you).

# Requirements

## Washing programs

When the user has selected one of the programs A, B or C, and inserted enough coins and pressed the start button, the control unit has to fulfill the following activities:

**Program A:**

* Cost: 360
* Lock
* Prewash
  + take water (fill 50%), no heating, add soap 1, rotate clockwise, regular speed, 1 minute, rotate counter clockwise, regular speed, 1 minute, sink (dirty) water
* main wash
  + take water (fill 50%), heat (50%), add soap 2, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 1 time, sink (dirty) water
  + take water (fill 50%), no heating, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 1 time, sink (dirty) water
  + Centrifugation, keep sinking the water, rotate clockwise, highest speed, 30 seconds, rotate counterclockwise highest speed, 30 seconds, repeat rotating 1 times
* Unlock

**Program B:**

* Cost: 480
* Lock
* Prewash
  + take water (fill 50%), heat (50%), add soap 1, rotate clockwise, regular speed, 1 minute, rotate counter clockwise, regular speed, 1 minute, sink (dirty) water
* Main wash
  + take water (fill 50%), heat (50%), add soap 2, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 1 time, sink (dirty) water
  + take water (fill 50%), no heating, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 1 time, sink (dirty) water
  + Centrifugation, keep sinking the water, rotate clockwise, highest speed, 30 seconds, rotate counterclockwise highest speed, 30 seconds, repeat rotating 1 times
* Unlock

**Program C:**

* Cost: 510
* Lock
* Prewash
  + take water (fill 50%), heat (50%), add soap 1, rotate clockwise, regular speed, 1 minute, rotate counter clockwise, regular speed, 1 minute, sink (dirty) water
* Main wash
  + take water (fill 100%), heat (100%), add soap 2, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 3 times, sink (dirty) water
  + take water (fill 50%), no heating, rotate clockwise, regular speed, 1 minute, rotate counterclockwise regular speed, 1 minute, repeat rotating 3 times sink (dirty) water
  + Centrifugation, keep sinking the water, rotate clockwise, highest speed, 30 seconds, rotate counterclockwise highest speed, 30 seconds, repeat rotating 2 times
* Unlock

## More requirements

During operation of the LM, the control unit must realize the following requirements:

**Door and soap:**

* Program can only start when door is closed and sufficient soap is provided
* When program starts, tank door and soap compartment are locked
* Door and soap compartment can only be opened when unlocked
* Normally door and soap compartment will be unlocked when program is ready

**The water:**

* When water drain opens water is pumped into the tank
* When water sink is opened the water is pumped out of the tank
* Four levels of water can be distinguished (empty, 33%, 67%, filled)
* When both the drain and the sink are open, water level lowers slower.

**Water pressure sensor:**

* When no water pressure, the tank cannot be filled (any further); without water no new programs can be started.
* When pressure falls down during a program:
  + When taking water is still required during the rest of the program. In that case the program pauses in a safe mode (no heating) for about 10 minutes. During this time a blinking code is shown on the program led.
    - When pressure returns in this time period, the program resumes
    - When pressure does not return in this time period, the tank is drained and the program stops (door is unlocked) with an error code
  + When taking water is not required during the remainder of the program, the program finishes as usual

**Heating/temperature:**

* Heating is allowed only when at least one level of water is in the tank
* Heating without water causes immediate damage
* Four different temperature levels can be distinguished (environment, warm, warmer, hot)
* While heating temperature will raise one step in a few seconds
* When heating stops, temperature will lower one step each few seconds

**The motor:**

* The motor can turn left and right
* Direction is regularly changed during the washing program or centrifugation to get improved clothing distribution in the tank
* Motor must be stopped prior to changing direction, this increases lifetime considerably. When switched off, stopping takes 1, 2, 5 seconds depending on the speed
* The motor can turn on low, normal, high speeds
* Low to be careful, normal for regular washing and high to dry the clothes (centrifugation)
* During centrifugation there must be no water in the tank and the sink must be open
* Centrifugation with a filled tank cause immediate damage to the machine

**The cashbox:**

* Three coins of 10, three of 50 and two of 200 can be entered
* All price combinations of 10,20,30 and 50, 100, 150 and 200, 400 are possible
* When program has not yet started, CLR returns all money
* Program starts only if there is enough money in the cash box.
* When program starts, the required amount of money is taken
* When clear is pressed once (change) money is returned
* The clear indicator shows that there is change to be taken
* When clear is pressed again, it is assumed that the money has been taken

# Simulator

Some information about the behavior of the LM simulator:

* when the heater becomes active, the heater LED is on, and the temperature starts to increase (LEDs T1, T2 and T3)
* when the heater is inactive, the water cools down
* the actual temperature can be read via two input lines
* when the drain is open, and the sink is closed, the water level increases
* when the drain is closed and the sink is open, the water level decreases
* when both drain and sink are open, the water level remains constant