Major Project Report – Walid AlDari – 218 375 162

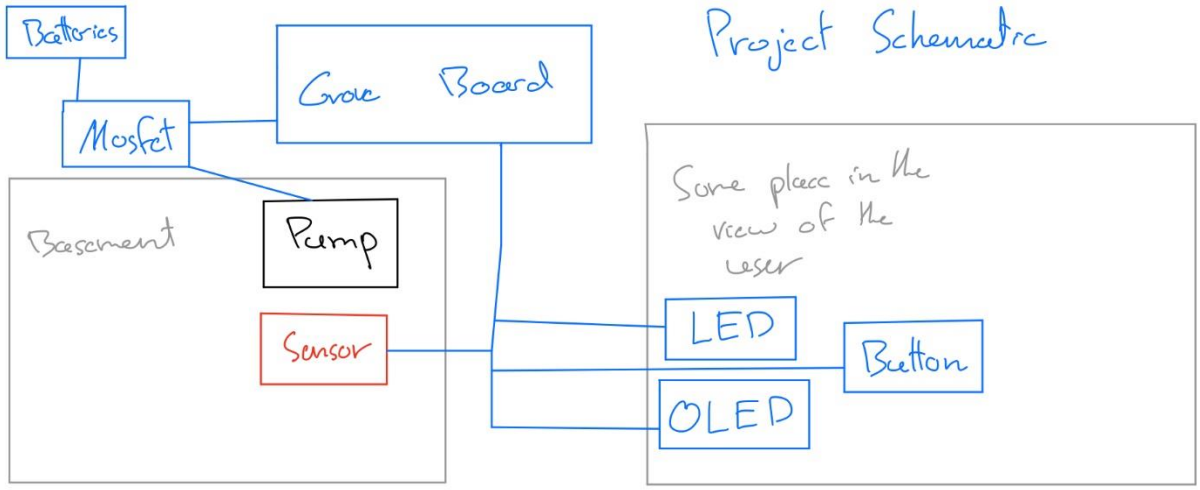
INTRODUCTION

My major project is basically an anti-flooding system that has 2 adjustable modes. An automatic mode that will pump out any flooding water as soon as the sensor detects water, and a manual mode that will only alert the individual about the flood but will not pump until the person does it manually.

CONTEXT

The anti-flooding system is a very useful system that can be used in third world countries that suffer from flooding a lot, as its very easy to build and integrate and can be set up by almost anyone. Its also very effective and cheap, and so mass-producing the project won't be expensive and won't require tons amounts of funding which is hard to get nowadays.

TECHNICAL REQUIREMENTS/SPECIFICATIONS



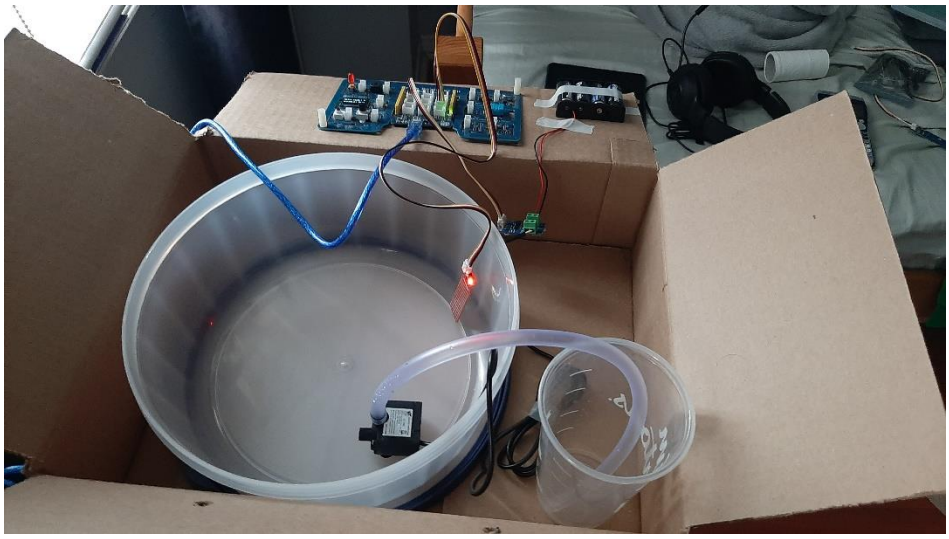
General Requirements:

- Required Arduino components
- A basement or an area that suffers from flooding
- Electricity

COMPONENTS LIST

- Grove board: brain of the system

- MOSFET: connected to the grove board, controls the supply of power from the batteries to the pump
- Water Pump: connected to the MOSFET, moves water
- Batteries: connected to the MOSFET, supplies power
- Water Level sensor: connected to the board, detects water and sends its level
- LED: connected to the board, lights up when the system is in manual mode, and the water sensor detected flooding
- OLED: connected to the board, displays the current water level, the current buzzer volume, and the type of mode the system is set to (auto or manual)
- Button: connected to the board, activates the pump when pressed if the system is in manual mode



PROCEDURE

Simply modified the minor project by replacing the moisture sensor with the water level sensor, I then added a Tupperware acting as the basement. I then tested the water level sensor to check whether its working or not and determined the “dry” level of the sensor, which was any value below 400 is considered either dry or a bit wet, and any value above that means there is actual water, and any value above 650 implies there is a large amount of water.

After testing out the components, I started writing the Arduino code for the project, which consisted of if statements and multiple functions. Afterwards I started working on the java, by setting up the app to contain 2 labels, 2 buttons and a slider. I then established a connection on the com port between the Java program and the Arduino, I then binded the buttons to change the mode of the Arduino (auto or manual), afterwards I binded the slider to control the buzzer volume/frequency.

I then had all the info displayed on the OLED. Afterwards it was just debugging. Some of my problems during development included: the button not functioning when the slider value is modified, pump not activating when the java program is running, and finally the last error I fixed was that the slider value was not only changing the buzzer volume but also the mode.

TEST

I tested that the system worked by going through most of the settings combination, meaning I set the mode to automatic and made sure that only the pump was the active component while the others should not be doing anything. I then set the mode to manual and made sure that the pump did not activate unless the button was pressed, I also checked to see whether the slider changed the buzzer frequency. And made sure the LED and buzzer were activated when the sensor was in water.

LEARNING OUTCOMES

Yes, my project addresses the 5 learning outcomes.

- LO 1: My project wouldn’t have worked if I never tested it or debugged it, and so I did encounter multiple problems when testing and through debugging I was able to solve them all.
- LO 2: To address the problem of basement flooding, I’ve built an application that utilizes the jSerialComm API to establish a connection between the java program and the Arduino which are 2 vital components of my application/project.
- LO 3: A few parts of my major project were completed during the labs, and so I’ve adopted that ready-made code into my program.

- LO 4: My entire project is event-driven, part of it is automatic and the other part has the button controlling the pumping event, and has the java slider controlling the volume of the buzzer.
- LO 5: The project required a mix of coding knowledge and very subtle electrical engineering knowledge, that when used with the objected oriented language of Java resulted in my major project. Arduino isn't object oriented however I used Java which is object oriented.

CONTINGENCY

There is a minor problem in my project that lies in the sensor itself, when its wet but not fully submerged in water, the system thinks that its still in water. To fix that problem I thought of adding a servo connected to the sensor and have it shake off any water stuck on the sensor. However, my current servo was broken and the province went into lockdown for the 1500th time and so I couldn't purchase a new part, so I couldn't implement that.

Next time, I would start much earlier. Looking forward to eng 4000, I think working in a team would be the perfect thing to do as in a team we can build a complex project that'll take less time to make compared to working alone.

ADDITIONAL MATERIAL

Nothing else to add

CONCLUSION

In the end, I have enjoyed working on this project and I've learned a lot. And I look forward to using and building on this knowledge that I've gained to pursue better projects with larger impacts.