

EECS 1021 MINOR PROJECT REPORT

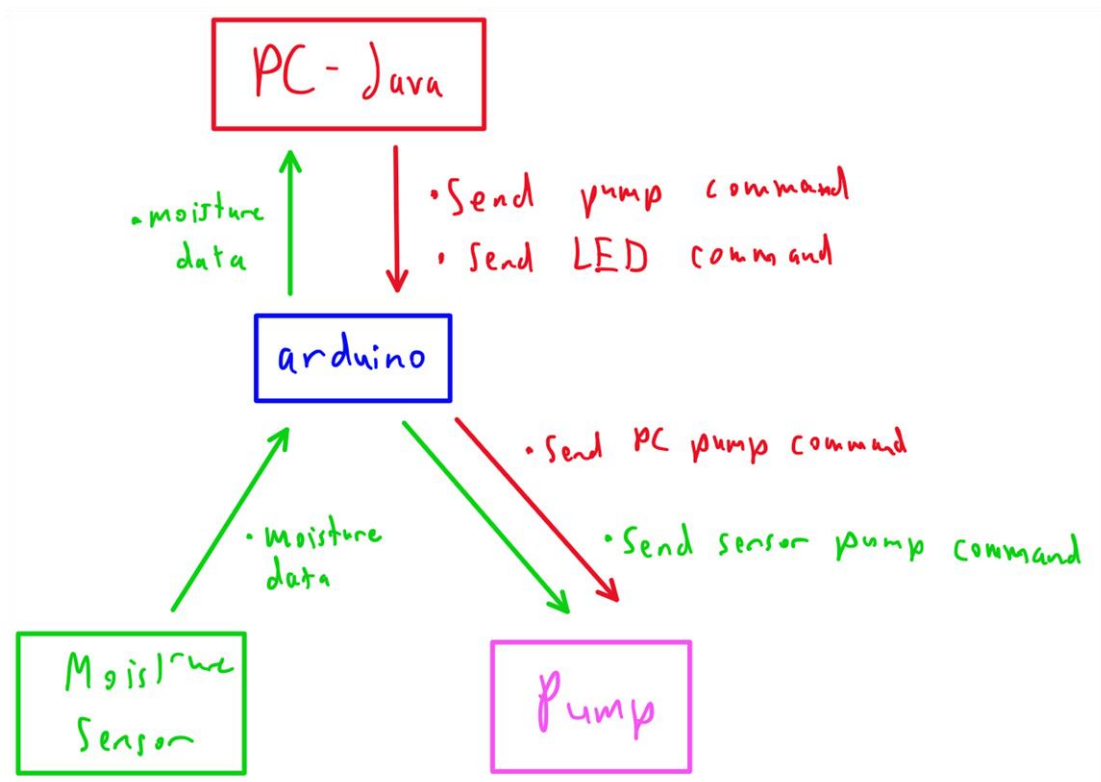
Kiet Le, 218052787

Introduction

The minor project that I worked on is an automated plant watering system that uses an Arduino microprocessor, a moisture sensor, and a pump as its components. Along side the hardware I used Arduino IDE code in tandem with Java code. This allowed me to create a GUI where I could control the pump and see moisture data from my computer.

This project is a good introduction to me about combining object-oriented programming with things like sensors and actuators to have a useful function.

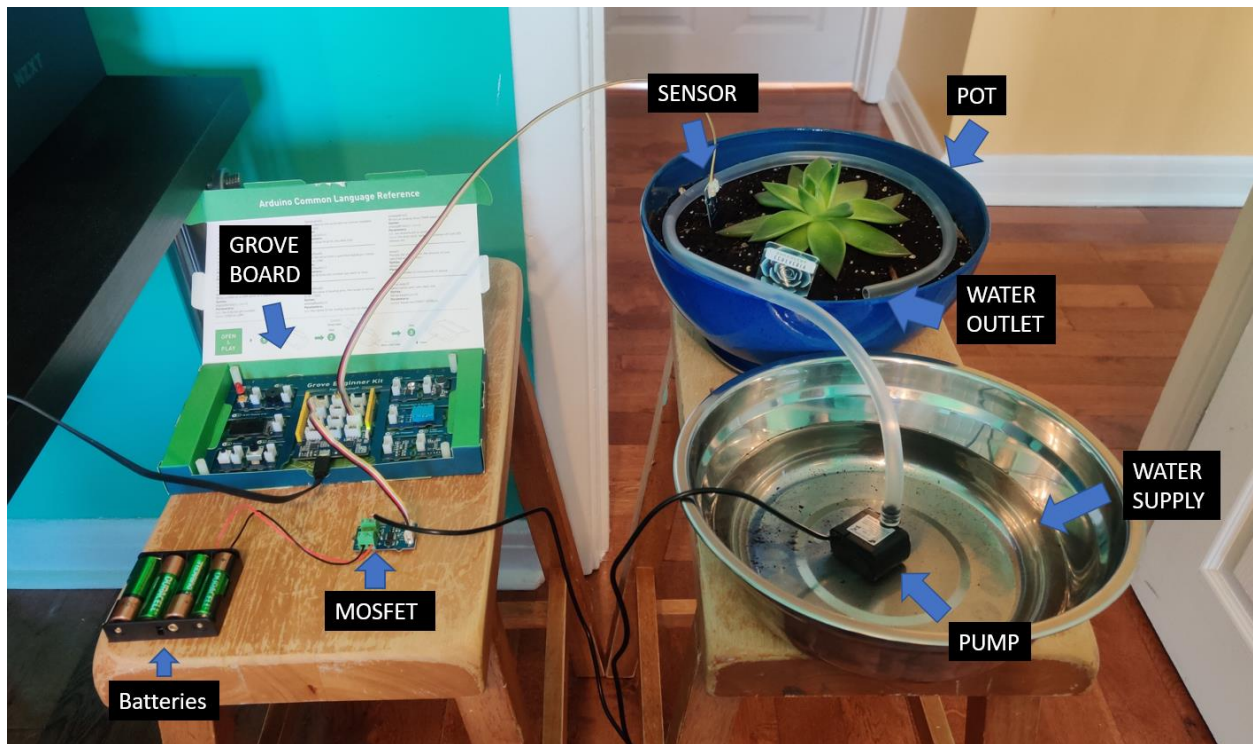
Technical Requirements/Specifications



Functions:

- Automatically pump water when moisture sensor is detecting above the wet threshold.
- While the moisture sensor is detecting a moisture below the wet threshold and the pump is inactive, the user can manually activate pump for a few seconds with a button from the GUI.
- GUI displays voltage graph which shows the relative moisture values with respect to time
- GUI displays a voltage T-chart which shows the relative moisture values with the time it was taken.
- GUI has a slider which can be used to turn on and off a red LED on the board.

Components List



- Grove Arduino board
 - Grove board holds the Arduino microprocessor.
 - It also has the red LED that I use.
- Moisture sensor
- Pump
- MOSFET
- Computer (with Java)

Procedure

The process I used for creating my project was to cut the different function implementation into sections. That is, every feature that I was to add, I would work on that feature exclusively before moving to the next. So, I started with the automatic moisture pump, the feature where the pump will automatically turn on when the moisture value is above a certain point. After that I moved onto the moisture data graph, where the Java would get data from the pump and then graph it with respect to time. Then, I moved onto the manual pump feature, where when the sensor is below the moisture threshold and not on, I could press a button in the Java GUI and activate the pump. My last function I worked on was the slider LED function. When the slider is between [1,2), it would activate the red LED on the grove board. When the slider was on 2 then the red LED would turn off.

Test

For the automatic pumping feature, I tested that by dipping the sensor into some water. When it would be in water the pump would stop. When it was removed it would start pumping again. Testing the manual pump would just consist of getting the pump turned off by putting the sensor in water. When it

is turned off, I can click the button in the GUI and if it started pumping then I know it works. Testing out the graph would consist of me putting the sensor in and out with water. If the graph moved in tandem with the state of the sensor then I can confirm that it is functional.

Contingency

One idea I had in mind was to make the slider change the brightness of the LED. This couldn't really work since I couldn't find a way to control the voltage sent to the LED. Maybe next time I will find a way to adjust the voltage sent to the LED. I may need more components that I don't have on hand at the time. Some lessons that I learned that I could apply in the future is probably the fact that you can have so many possibilities from the components that you have. Using your computer with things like microcontrollers and actuators and sensors opens up so many possibilities for innovation and creativity.

Conclusion

Overall, I had fun with the project. Getting to create something that had features like a GUI that controlled the pump and displayed the sensor values was a really cool experience. This will inspire me to work on cooler things moving forward so that I may expand my tools and allow me to become a better engineer.

Video Link: <https://youtu.be/BeJWmJbCEqM>