



# EECS 1011

## Major Project: Auto Smart Plant Watering & Alarm System Using Arduino on MATLAB

### Abstract

This lab consists of creating a 'state' machine. This refers to the core of a program that takes actions based on things that are sensed or measured. A state machine is created to monitor and water a plant, while also monitoring a plant's climate & light exposure, and react to it.

Ali Saheb

alisaheb@my.yorku.ca

218791376

December 7, 2021

## Report

**INTRODUCTION:** This project consists of creating a MATLAB script using different Arduino hardware to code a functioning self-sustainable smart plant alarm system. The system is developed around a spearmint plant but can easily be tweaked by any user, even with little coding knowledge, to operate any type plant's needs. **CONTEXT:** The objective of this project is to create an independent smart plant alarm system. With a smart plant alarm system, the plant can water itself, as well as warn the user when the surrounding temperatures or light exposure varies from what is set to be 'pristine' (meaning perfect condition). This is to decrease the loss of plant life around a household, as well as prevent the user from forgetting to manage their plant. **REQUIREMENTS:** For the smart plant alarm system to be a viable option to caring for your plants, the system must run on a constant loop so that the plant is able to monitor itself 24/7. The most important feature must be for the plant to successfully water itself based on moisture readings and stop watering once the amount of water is sufficient. The plant must also read its surrounding temperature and react based on these readings (If too warm enable fan to cool down, if too cold enable alarm to inform user that the plant must be moved to a warmer climate). A light sensor must also be integrated as certain plants prefer different amounts of sunlight; therefore, an alarm must sound if the preferred conditions are not met. There must also be a live moisture graph reading for the user to have a visual understanding of the moisture fluctuations in the soil before and after watering, as well as overtime. **COMPONENTS LIST:** This project consists of the following components:

- Arduino Grove Beginner Kit
  - Used to control the different modules (via MATLAB).
- Moisture Sensor
  - Provides different voltage values depending on the amount of water present (resistance).
- Pump
  - Used to pump water from the water source into the plant (via clear tubing).
- MOSFET Board
  - Used to power the pump (contains 9V battery connection).
- Buzzer (D5), LED (D4), LED (D13).
  - Buzzer (D5) used as board communication and alarm system along with red LED (D4) to trigger alerts/warnings; green LED (D13) used to signify a healthy system.
- Light Sensor
  - Used to provide different voltage values based on surrounding light exposure.
- Temperature Sensor
  - Used to read surrounding temperature to determine if it is too hot/cold and communicate with the user on whether the plant should be relocated or not (also sends signal to 5V fan to cool the plant down if needed).
- 5V Fan
  - Receives signal to cool system down if temperature reading is warmer than preferred.

**PROCEDURE:** MATLAB is used to create a code that reads the moisture sensor, and different voltage values for wet, dry, and moist soil were assigned based on previous testing. The code then reacts depending on if the soil is wet or dry and pumps a measured amount of water to the plant. A live graph is created to visualize how the soil reacts to being watered. If the soil moisture is determined to be

sufficient, the system moves on to temperature where preferred temperatures are also determined and assigned based on previous testing. Depending on the temperature, the system then suggests to the user what is best for the plant and enables a fan for a short amount of time to cool the system. Following this comes the light sensor, where different voltage values outputted depending on the amount of light exposure. Predetermined voltage values were created based on previous testing, and variables such as sunlight and dark were assigned. The system reads the current light exposure and informs the user if action is required. The system is then repeated on a loop and tests the soil moisture once again. **TEST:** To test the system, I was able to remove the moisture sensor from the board and use the built-in potentiometer to set different voltage values. This allows for testing of any voltage value, allowing me to visualize how the system would react under any circumstance. For the temperature sensor I was able to test the system by setting the 'temperature\_celsius' variable to the temperature I wanted to test. This way I was able to test all temperature conditions and see how the system reacts without freezing/burning myself! To test the light sensor, I was able to turn off all surrounding light sources to determine what a 'dark' voltage reading would be. I then recorded the voltage reading after shining a flashlight directly at the sensor. This simulates sunlight and gives an idea of what the value would be if it were directly under a light source. However, spearmint plants prefer shade therefore the system reacts to the surrounding light exposure and prompts the user on what they should do to have a full healthy plant (move to shade/move to sunlight). Using different live graphs and voltage readings I was able to determine what the ideal conditions for the spearmint plant are. **CONTINGENCY:** Another idea I had in mind for the major project involved the use of an air quality sensor to determine the quality of the surrounding air. The issue with this idea was that the module is a lot more complicated than any other module we have used in the past, and so I was unable to create a functioning air quality monitor within the time limit. In the future I can give myself more time to figure out how to use new modules and components to push myself, strengthening my creative problem-solving skills and forcing me to become more of a critical thinker (related to ENG). **ADDITIONAL MATERIAL:** The reasoning behind the creation of this smart plant alarm system was because in my household (as well as many others) I noticed that many of our plants were sometimes forgotten about, resulting in dehydration and in turn the end of the plant's life. The location of the plants was also to blame. Places like the front/back door's temperature is constantly fluctuating as people enter/exit, creating an unsustainable environment for the plants. Light deprivation caused the plants to grow slower, restricting the plant from fully blooming. Even when caring for the plant to the best of your ability, you may simply misjudge the location of the plant or the water you give it. For these reasons, the smart plant alarm system is a phenomenal solution to allowing your plants to live with the utmost efficiency, without the need to constantly be monitored. **CONCLUSION:** The system successfully monitors a plant without the need for supervision, allowing the plant to be entirely independent in the sense that it will monitor and look after itself. Anyone struggling to raise plants of their own may see this as an efficient and easy method to providing the best care for their plant possible.

## Measured Values/Constants

### Moisture Sensor Voltage Values

Not immersed in anything (air): 3.7 Volts.

Immersed in dry soil: 3.5 Volts.

Immersed in water-saturated soil 3.2 Volts.

Immersed in a clear glass of water 2.5 Volts.

### Pump Timing Value

Pump: approx. 25mL / 5s → 5mL/s

Therefore, 1L in 200s (3min 33s)

### Temperature Sensor Values (spearmint plant preference)

Hot: 21°C

Cold: 13°C

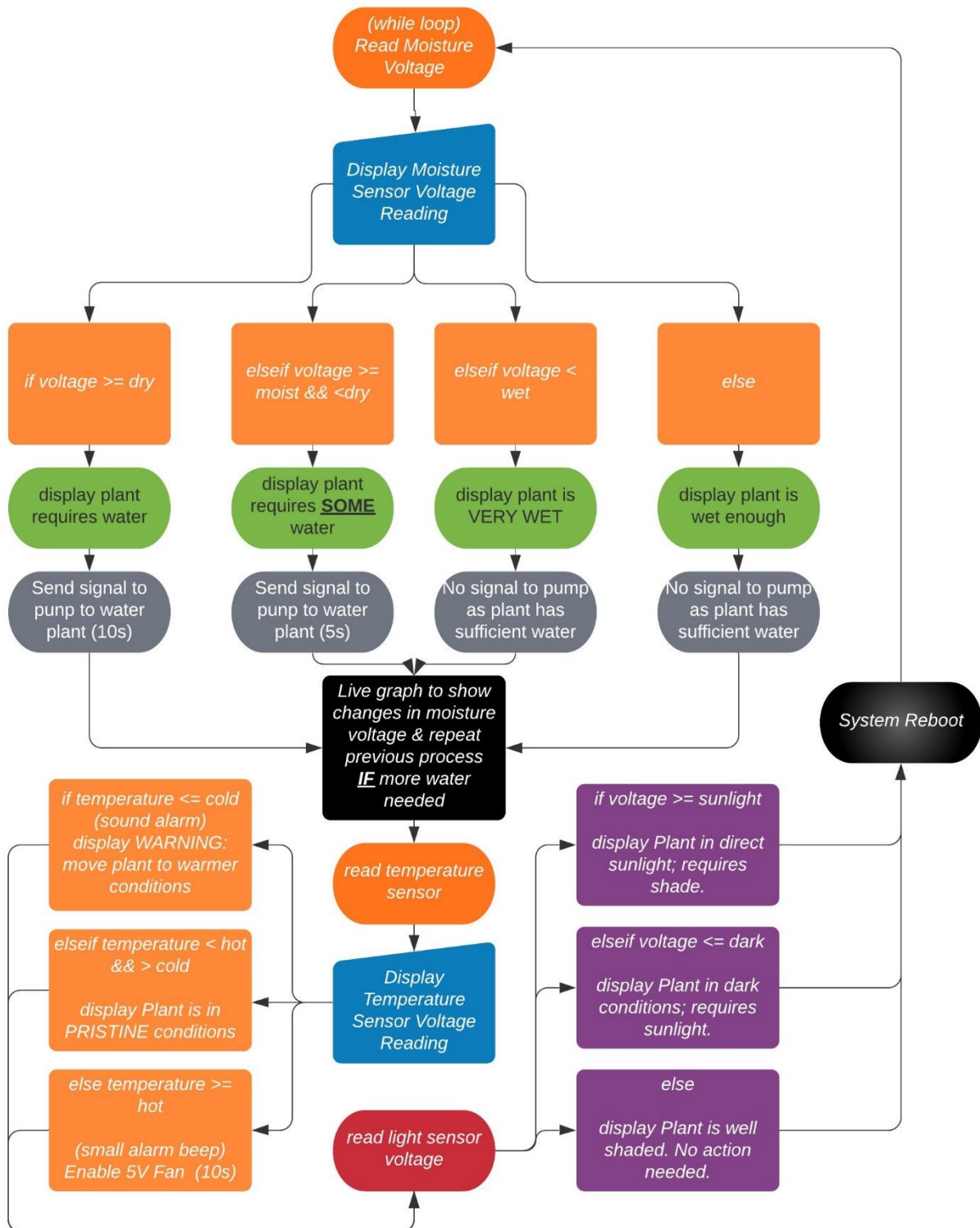
Pristine: 17°C

### Light Sensor Voltage Values

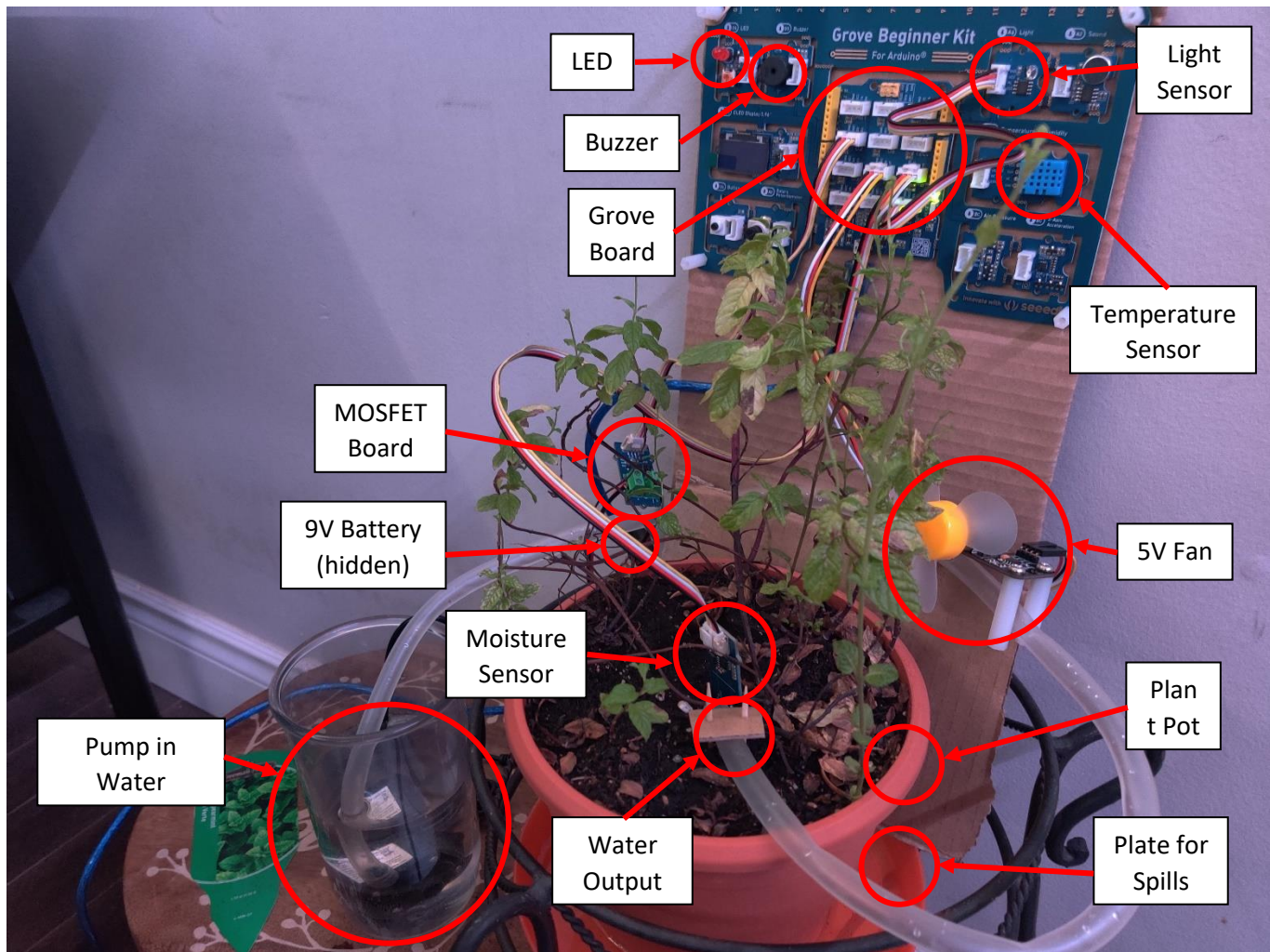
Sunlight: 1.5 Volts.

Dark: 0.5 Volts.

## Flowchart of Code



## Plant Hardware Setup



## Video Demonstration of Major Project

[Click Here for Link!](#)

(In case above link doesn't work):

[https://drive.google.com/file/d/1gBWI-F5hiy5KHiYuEh1f3v3\\_URT535\\_X/view?usp=sharing](https://drive.google.com/file/d/1gBWI-F5hiy5KHiYuEh1f3v3_URT535_X/view?usp=sharing)

<https://youtu.be/F8PaIb0cIRc>

**P.S** I apologize if the video seems rushed as it was difficult to fit the entire setup and demonstration into a 5-minute video. I demonstrated all features to the best of my ability.