LeafIt!

An Autonomous Watering System

University of Central Florida

EEL 4914: Senior Design 1 Summer 2022

Divide and Conquer 1.0

Group 9

Kyle Eaton: Electrical Engineering

Adam Keller: Computer Engineering

Francis Olearcyzk: Electrical Engineering

Matthew Ongcapin: Computer Engineering

Table Of Contents

Section 1: Motivation, objectives, goals, and function

1.1 Motivation

Everyone loves to go on vacation or travel, however if you grow plants at your house this can be troublesome if you are gone for an extended period. You either must rely on a neighbor or friend to water your plants or hope that the rain waters them enough while you are gone. Our project, the autonomous watering system: LeafIt! solves this issue. You no longer need to worry about your plants while traveling.

Another use case could be for elderly or disabled people, where plant care might not be the easiest thing to do. Our project will water and monitor your plants for you, all you need to do is the initial setup and refill the water reservoir.

1.2 Project goals and objectives

The primary goal of this project is to create a system to monitor/ water your plants while you are away on vacation. The system will use sensors to gather the temperature, moisture levels, water levels, etc. Any of these sensors can trigger a notification sent to the user. For example, if the water reservoir is low a notification will be sent to the user to refill the water. We will also make a phone app as an interface to use the watering system. The phone app will alert you when the water reservoir is low, a plant needs more sunlight, the moisture or temperature is off, etc.

1.3 Discussion of the function of the project

The function of this project is to create a system to monitor and take care of plants autonomously. All the user needs to do is refill the water reservoir. We will make a phone app for easy User Interface and ease of use for when you are away from your house.

1.4 Product Research

SPlant Automatic Drip Irrigation Kit

This is an example of a commercial product that we came across in our product research. Most of the commercial versions of this product have this kind of design. Like a house's sprinkler system, the user pushes buttons to set watering times for all the plants.

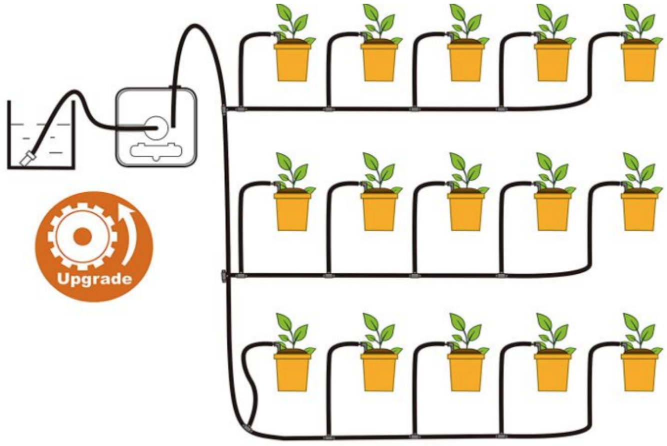

Figure 1: Example overview

IoT Automatic Plant Watering System

Other automatic plant watering systems seem to be mostly DIY projects, especially in the field of IoT. This project from Instructables showed us the possibility of using technology to connect the task of plant watering to smart phones since in this project, the act of watering can be controlled by a smart phone. The product research inspired us to utilize the basic design of the commercial plant watering systems combined with the technology and smaller scale of the DIY projects that have yet to break into the market.

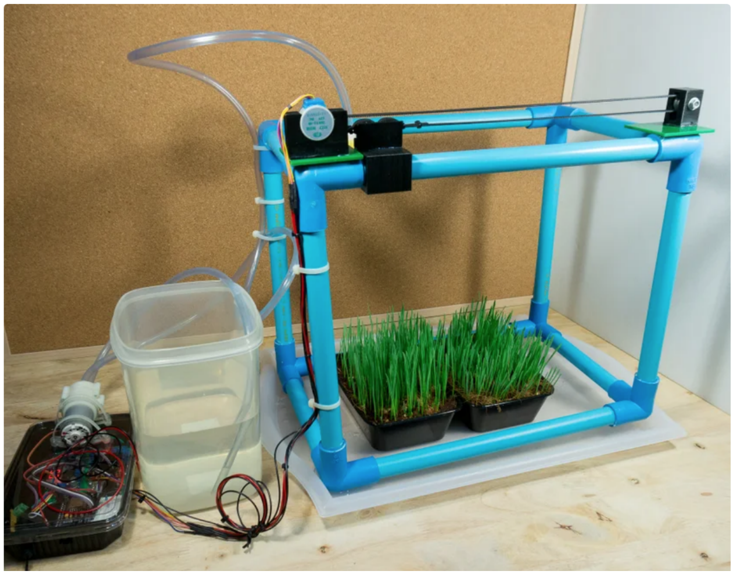

Figure 2: DIY example

Section 2: Requirement Specification

The self-watering plant system that will have these specifications:

- The system shall automatically water one outdoor plant using a water pump with amounts of water based on one of the three possible types of plants that the user inputs, the temperature, and the moisture present in the soil of the plant
- The system shall have an interface for users to choose soil moisture and temperature settings based on three types of plants
- The system shall have two sensors to monitor soil moisture of the plant and temperature
- The system shall use a water pump to draw water from a reservoir of water outside of the system i.e., a bucket or pot of water
- The system shall alert the user’s Android phone of inclement weather, unfavorable amounts of moisture in their plant as well as updates on the plant when it has been watered
- The user shall be able to input into the system “vacation days”, prompting the system to calculate if the water reservoir is passable given the type of plant and the number of vacation days.
- The system components shall be a power system powered by batteries, a temperature sensor with conditioning circuit and ADC, a soil moisture sensor with conditioning circuit, a water pump, a microcontroller, a development board, an Android phone, and elastic pipe

- The system's watering and temperature settings shall be based on three types of plants: Tropical, Temperate, and Cactus/Succulent
- The system shall not weigh more than 20 pounds
- The system shall support a container of water at least 0.5 Gallons and at most 3-5 Gallons
- The system shall be no taller than 2-3 feet
- The system shall be no longer than 1-2 feet
- The power system shall have current leakage protection
- The power system shall be capable of delivering at least 12V DC
- The power supply shall be capable of powering the sensors
- The total cost of the system shall not exceed $200
- The microcontroller must be able to produce a PWM signal that has the amplitude of 1V max and a minimum of 1V
- The frequency that the microcontroller must produce on the PWM signal must be able to operate on a range of 50Hz up to 100Hz
- The minimum operation voltage of the microcontroller must be at least 3.3 volts
- The program of the system must be able to compile and correctly output the expected actions deemed by the user input
- The system shall have weatherproof protection for electronic components
- The system shall have adequate heat sinks for device and user protection

Section 3: Prototype Representations (Project prototype illustration, block diagrams)

Project Prototype Illustration

The figure below shows all the components working together in the system. After inputting the user settings, the MCU prompts the water pump, submerged in a container of water, to water the plant. Sensors check the temperature and soil moisture in order to prompt continued watering. An Android phone receives updates on the plant's health from the hardware.

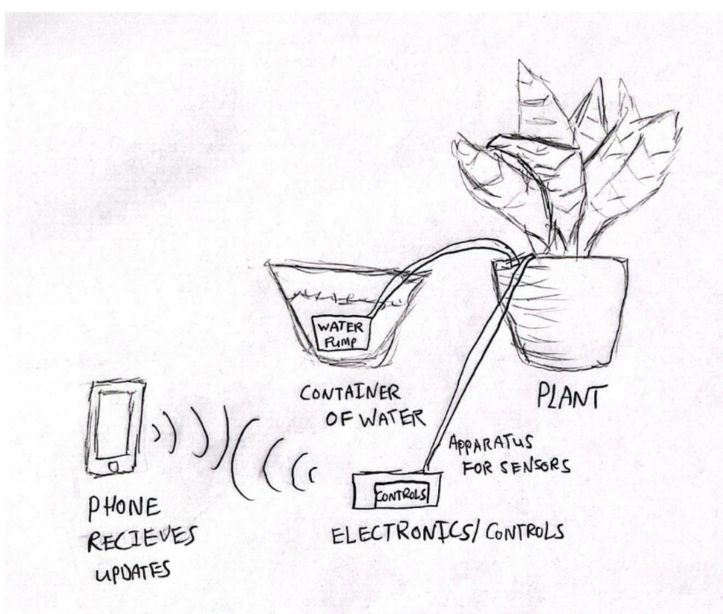

Figure 3: Concept Illustration

Project Block Diagram

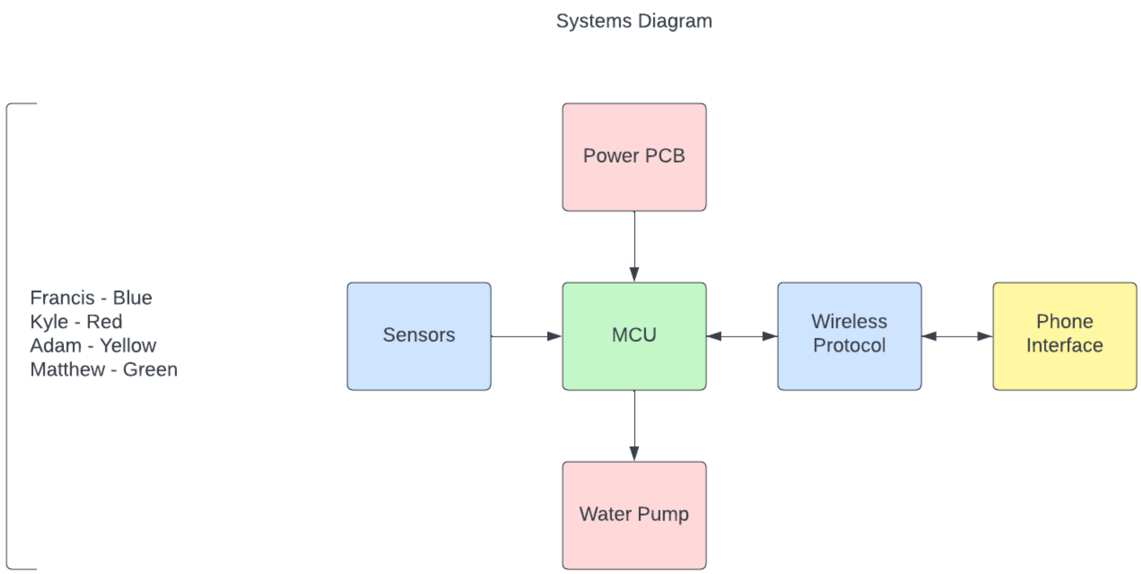

Figure 4: This is the overall system diagram that shows who is in charge of what parts of the project

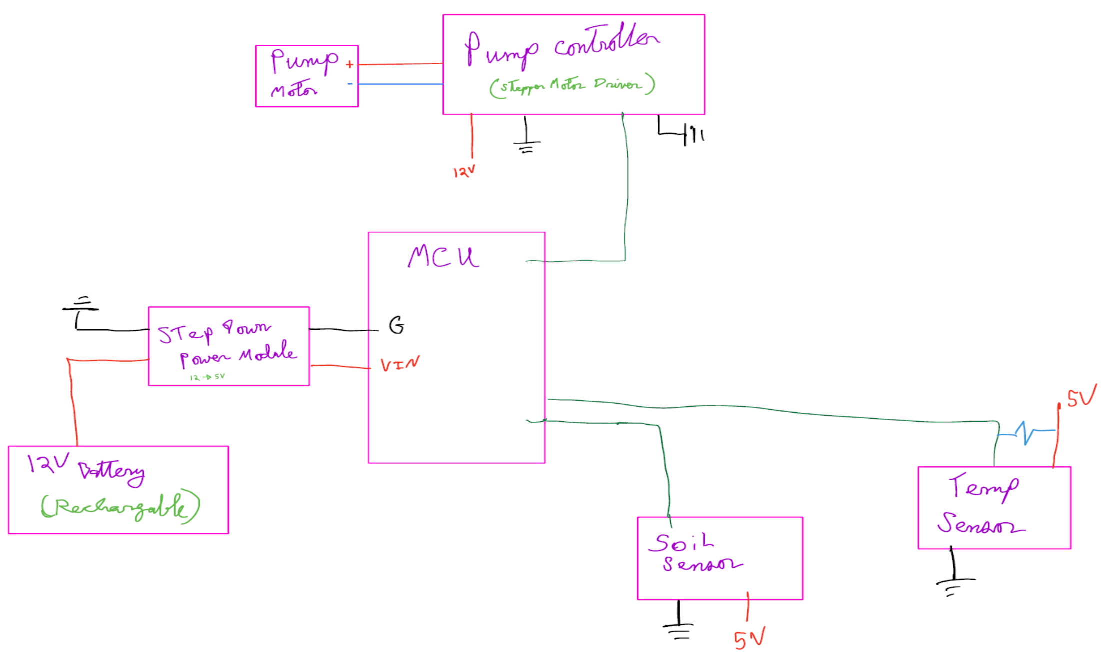

Figure 5: This is the power structure of the system

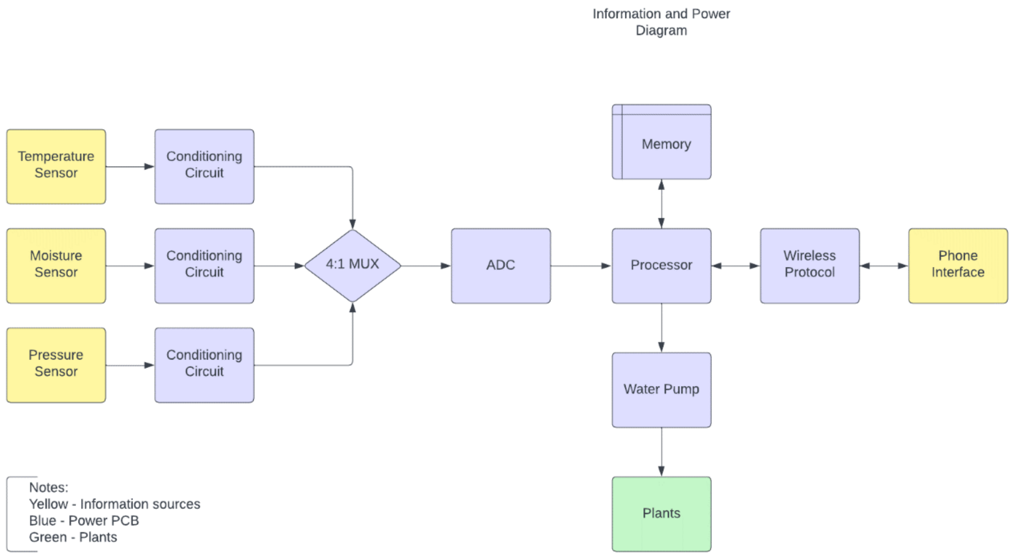

Figure 6: This is a general overview of the information and power flow

Sensors

This system will operate with at least three sensors. There will be a temperature sensor, a moisture sensor, and a pressure sensor. The temperature and moisture sensor will provide information to the MCU about the conditions the plants are in, and the pressure sensor will measure how much water is left in the water reservoir.

All three sensors will feed into one ADC. The input to the ADC will be switched with a multiplexor. Each sensor will have a conditioning circuit to reduce noise and to ensure that the system's resolution is not limited by the ADC. This information can then be processed by the system's MCU. The information flow can be seen in Figure 6.

Transceiver and Receiver + Protocol

This system will need to be able to communicate with the phone interface wirelessly. The two protocols that are currently being investigated are Bluetooth and Wi-Fi. Depending on which one is implemented will greatly affect the scope of the project. Both will most likely need a separate PCB.

Bluetooth

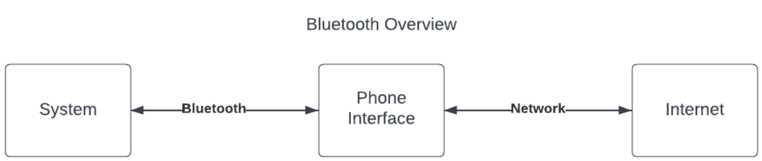

Figure 7: Information flow with Bluetooth

This protocol is simple to implement and will be Peer-to-Peer. It connects the system directly to the user's phone. The range can be about 50-100m. The user would have to locally connect to the watering system. This will change the scope by only allowing the system to communicate with the phone interface when the user is at home. Little information (a few bytes) will need to be transferred to and from the system, so the limited data speed of Bluetooth (1Mbps) is not a limiting factor. This uses much less power than Wi-Fi and will be less taxing on the system's power capacity. This will mean that the features that will use the internet will only work when connected to the phone. Overall, Bluetooth will be easier to implement and will use much less power but will only be able to use the internet when the devices are connected. Bluetooth is also much less secure than Wi-Fi.

Wi-Fi 802.11

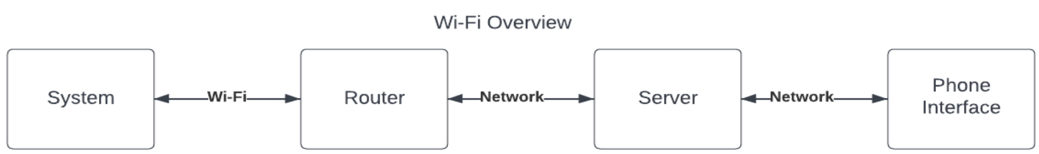

Figure 8: Information flow with Wi-Fi

This protocol will be more difficult to implement and will be Client-Server based. It will allow the device to directly communicate with a server which can provide information from the internet. This means that the system will be able to send live information to a server, and the user can access this information whenever and wherever they are. The watering system would send information to the server 3-4 times a day, and the user can check at any time and send requests. The server is there to minimize power usage and eliminates the issue where both the system and the phone interface have to be on to operate. Possible solutions to avoid the latter will be researched. This protocol uses almost ten times the power of Bluetooth on the device side. This will put strain on the power system and will reduce the operating time of the system. There are alternative low power Wi-Fi protocols, such as 802.11ah which allows for the Wi-Fi module to be turned off when not needed. This protocol is also much more secure but will also be more complicated to implement. Some design challenges would be implementing firmware to store Wi-Fi passwords and maintaining a server.

Potential Software Designs

The structures pictured below show two possibilities for the overall structure of the software. Structure 1 shows the three types of plants that could be inputted (Tropical, Temperate, and Cactus/Succulent), as three separate classes made possible by a "Plant" prototype class. The second potential structure shows a more stripped-down approach, with one "Plant" class with three methods for Tropical, Temperate, and Cactus/Succulent. The movement of the data in each approach is relatively the same in that the user inputs user data which is interpreted by the programming to prompt the system to do work, as well as prompt the system to display status updates to the user's Android phone.

We will utilize techniques such as tight cohesion/low coupling, the 23 software design patterns, clean code philosophy, and Top-Down development with incremental refinement to insure code optimization and cleanliness. Development will take place on a development board and IDE to create an application interface to interface with the system using an Android phone. Decisions made by the system based off of the temperature sensor will be aided by a weather API. The system will be able to connect to the Internet to connect to interface with the phone.

Structure 1

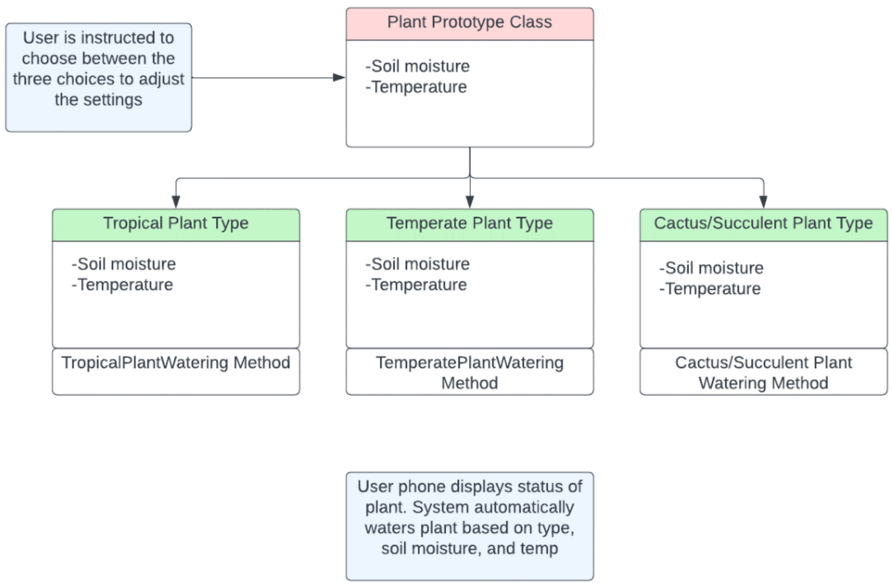

Figure 9: Code Structure 1

Structure 2

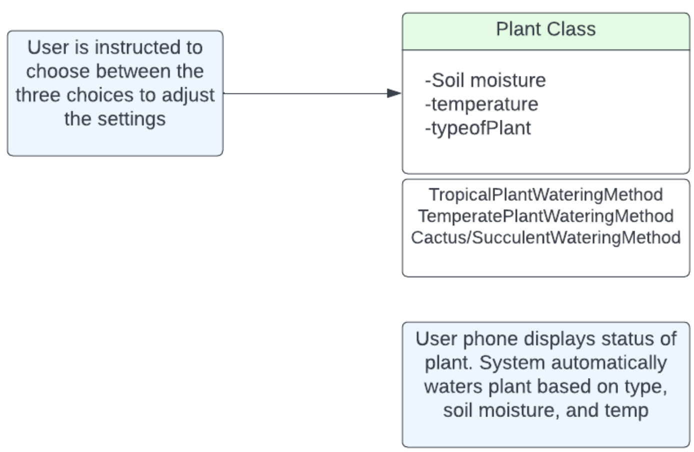

Figure 10: Code Structure 2

Section 4: Budget and Financing

Subsystem	Item	Quantity	Vendor	Estimated Cost
Power Subsystem	12V rechargeable Battery	1	Teneger/Amazon	$22

MCU Subsystem	TBD	TBD	TBD	TBD
Water Reservoir	Tupperware	1	Walmart	$10
Tubing	Aquarium tubing	1	Amazon	$6
Temperature Sensor	Temperature sensor probe	1	GikFun	$3
Moisture Sensor	Soil Sensor	1	Almocn	$3
Water Pump	12V DC Motor	1	Amazon	$16
Voltage Step down	Buck Converter	1	Amazon	$15
Pump Controller	Pump Controller	1	Qunqi	$7

Section 5: Project milestones

Senior Design 1	Task	Start	Due
	Initial Divide and Conquer	May 18, 2022	June 3, 2022
	Updated Divide and Conquer	June 8, 2022	June 17, 2022
	60 Page Draft	June 15, 2022	July 8, 2022
	100 Page Report Submission	July 9, 2022	July 22, 2022
	Final Document	July 23 2022	August 2, 2022
Senior Design 2			
	Assemble Prototype	TBD	TBD
	Testing and Redesign	TBD	TBD
	Finalize Prototype	TBD	TBD
	Peer Report	TBD	TBD
	Final Documentation	TBD	TBD
	Final Presentation	TBD	TBD

References

- https://www.techtarget.com/searchnetworking/tip/Compare-low-power-Wi-Fi-protocols-and-their-roles-in-IoT
- https://www.intuz.com/blog/bluetooth-vs-wifi-connectivity-for-iot-development
- https://www.instructables.com/IoT-Automatic-Plant-Watering-System
- https://www.ipriro.com/products/sPlant-Big-Power-Automatic-Drip-Irrigation-Kit-Indoor-Plants-Self-Watering-System@Tex Watson you dont get this