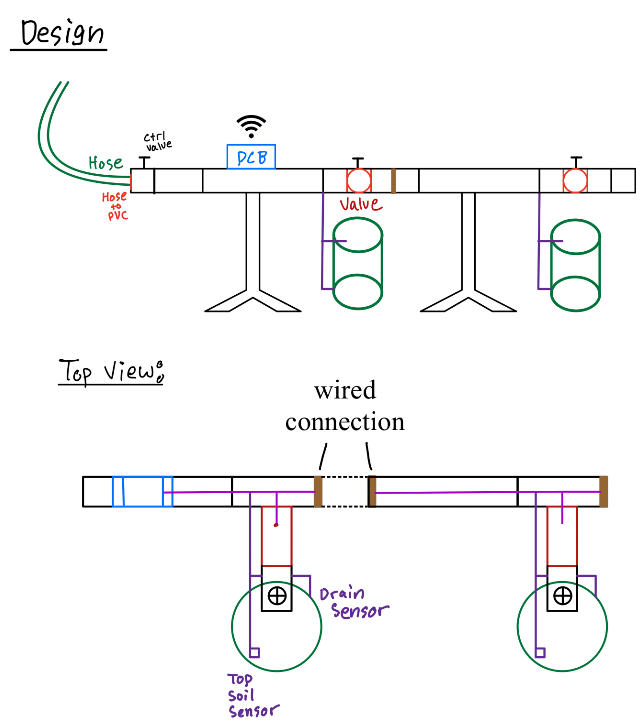
Introduction

Problem: Gardening is a skill that takes a lot of intensive care and effort as each individual plant has its respective living condition it must meet. These living conditions such as required sunlight, minimum amount of water, and climate vary from plant to plant and it can be very difficult to be attentive to all these details in keeping your plants in the best possible condition as we are occupied with our busy lives or simply lack the skill. Watering outdoor plants can be very tedious and a task often forgotten.

Solution: Our solution to this is to micro-manage the watering aspect of home gardening, taking input from soil sensors to form a smart irrigation system. This system will help the user monitor a single plant or more. In terms of current competitors on the market, other similar products are limited to the number of plants that can be monitored and require a water pump. This system will be modular and can be linked together to build a larger system. Other systems only measure the moisture content within the first couple inches of the surface and do not connect directly to a water hose. Our solution will water the plant till the whole pot is moist and fully watered. Using Solenoid valves in connection to a garden hose for irrigation, a single plant can be configured to have a minimal moisture level, providing the most desirable conditions for your plant, or a connect system can be created via daisy chaining.

There is a similar project from Spring 2023 but there are significant differences. The Project i and referring to is the "Don't Kill My Plant" Habit Tracker. Their project converts phone habits to watering / environmental changes. Our project aims to care for the plant in an outdoor setting and allows for multiple plants to be taken care of. Another similar project is DIY Plantify from Spring 2023. This project moves plants away from light if it is too intense and tests moisture levels based on weight. Again, very different from our water irrigation system.

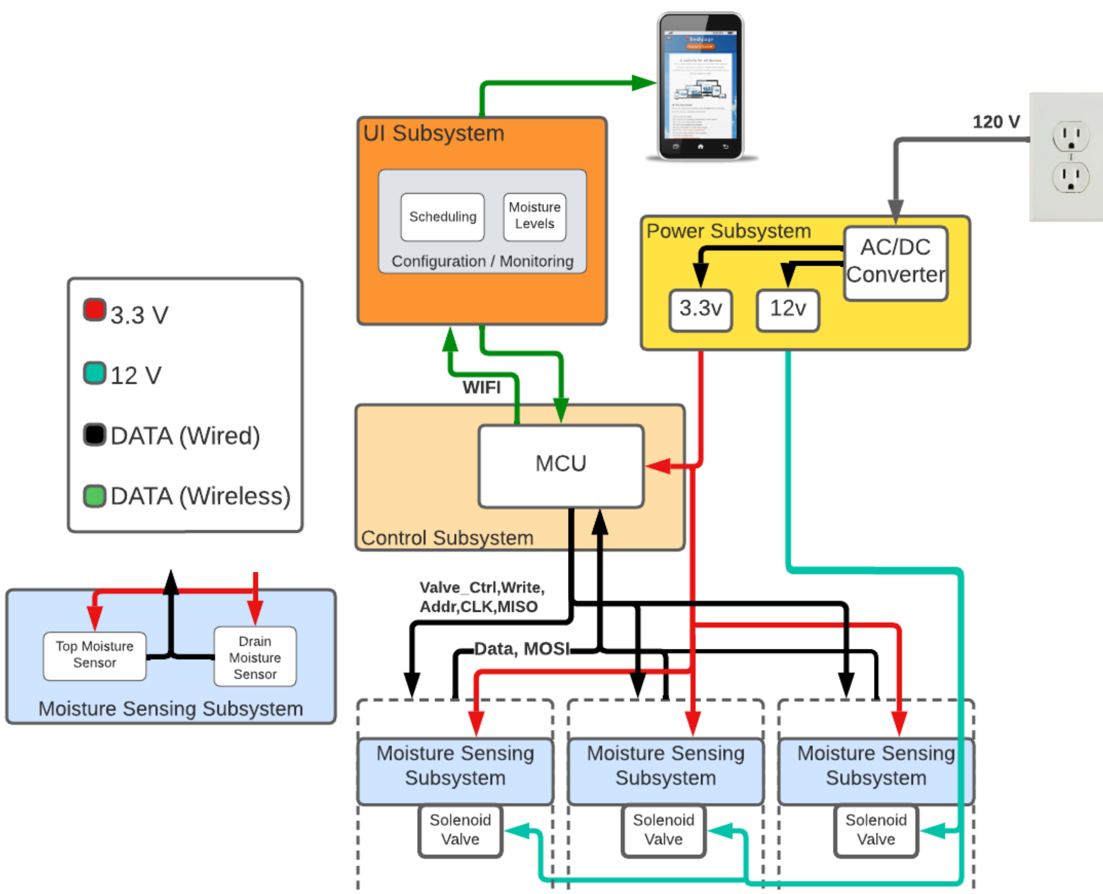


High Level Requirements:

- Requirement 1: The system controls moisture levels in individual plants, performing standard watering on schedule until drainage occurs, and continually monitors and waters plants if moisture deviates by more than ± 10% from user-defined values.
- Requirement 2:The system starts watering within ± 15 minutes of the user-defined time, respects a ± 15-minute permissible window, skips plants within ± 10% of their minimum moisture level, and stays within a ± 5% daily water usage limit. level for that plant. The system should not water if doing so would exceed ±5% of the user-defined maximum water usage per day.
- Requirement 3: The system can control multiple plants at once via master-slave communication. The system can manage the moisture levels of each plant individually and modulate their valves independently.

Design

Block Diagram



Subsystem Overview: A brief description of the function of each subsystem in the block diagram and explain how it connects with the other subsystems. Every subsystem in the block diagram should have its own paragraph.

A detailed explanation of the communication between master and slaves can be found in Appendix 1.

UI Subsystem: This subsystem is the user interface through which parameters can be set for operation of the system. It will communicate via bluetooth or WiFi with the Master Control subsystem and will relay input information from the user about permissible water usage, permissible watering hours, and the minimum moisture levels for each individual plant. Additionally, the subsystem must be able to receive information from the master control subsystem about the moisture level of each plant and the water consumption information.

Requirements:
Can be installed and operated on consumer mobile devices i.e. smartphone, tablet, etc.
Can communicate via bluetooth/WiFi with the master control unit.
Can take in user parameters, store them, and communicate them to the master control unit.
Can receive plant moisture and water usage information from the master control unit and display it in a legible manner to the user.

Control Subsystem: This subsystem is a microcontroller which acts as the master control of the system. It will communicate via bluetooth/WiFi with the mobile app to take in the user's parameters. Also, it will receive wired data from the slave subsystems which encodes the state of each plant's two sensors. Using the information from the state of the plant and the restrictions from the user's inputs on whether watering is allowed or not, it will send out a signal to open or close the valve for each individual plant. Master to slave wired communication will follow a basic Read/Write protocol. The master will cycle through every address in the system and will first have the Write signal low and perform SPI communication with the slave and make its determination of whether to open or close that slave's valve. Then it will raise the Write signal as well as output the open/close signal on the specified wire for a set amount of time before cycling to the next address.

Detailed Requirements:
Can operate off of a 3.3V with ~200mA current draw.
Can communicate via Bluetooth/WiFi with the UI subsystem.
Can store the input watering restriction parameters internally.
Can send/read information from the slave wired data signals.
Can correctly manage the communication protocol of cycling through each address, reading sensor data, calculating the valve control decision, and writing this decision out to the slave.

Moisture Sensing Subsystem: This subsystem will be replicated n times for n plants and contains the sensors and valve for its individual plant. It will have a sensor at the bottom of the plant to detect whether water is draining through the bottom of the plant or not, and it will have a moisture sensor within the top few inches of the plant's soil. It will contain the solenoid valve which is positioned to allow/disallow water flow to the plant. It will contain digital logic to implement the correct control/communication of the system. At a high level, it will have its unique address saved on chip and will compare with the address data lines to determine if it is being communicated with by the master or not. If it is, and the Write signal is low, it will allow data from its two sensors to be transmitted via SPI protocol to the Control Subsystem. If the Write signal is high, it will store the valve control signal value into a flip flop. The output of this flip flop is used as the control of whether the solenoid valve is open or closed.

Detailed Requirements:
Can operate off a 3.3V power supply +/- 5%.
Draws less than 20mA +/- 5% for digital logic components per slave.
Can communicate via bus with the control unit for sending sensor data and receiving valve control data.
Can communicate top level moisture data with +/- 10% accuracy.
Can modulate valve within 10 seconds of a change in control signal and correctly retain the valve state until the control information changes once again.

Power Subsystem: This subsystem delivers appropriate voltage levels to the other modules. It will consist of a connection to a 120V standard residential power system and a voltage converter to generate 3.3V for the microcontroller and the combinational logic. It will also have a voltage converter to generate 6V for the valves themselves.

Detailed Requirements:
Capable of using a standard US power system to drive all the logical components at 3.3V +/- 5%.
Capable of delivering 300mA at 3.3V +/- 5% for the microcontroller + digital logic components.
Capable of using a standard US power system to supply all the valves at 6V +/- 5%.
Capable of delivering 160mA * n for n valves +/- 5%. A reasonable bound on n is still being determined; for now we will aim for being able to power at least 4 slaves and determine the power/cost demands of higher numbers of plants. Thus for now we seek to be able to supply 600mA at 6V +/- 5%.

Tolerance Analysis

The most critical component of our system is getting accurate moisture data from the plants to the master control subsystem. If we are not getting reasonably correct data, our control decisions may be often incorrect and thus the system may operate seemingly randomly. We will aim to have the data received by the microcontroller to have less than 10% error from the actual value read by the sensor. Our sensors and our 8 bit ADC will both run on a 3.3V supply voltage.

However, online reading suggests that the max output of the sensor will be 80% of Vcc = 3.3V * .8 = 2.64V. So our analog sensor output will have an operating range of 0 - 2.64V.
Our 8 bit DAC sourced at 3.3V will detect levels every 3.3V / (2^8) = 0.129V step. Sampling an analog signal with an 8 bit DAC introduces quantization error while going from the "infinite precision" analog signal to the 8 bit precision DAC. Effectively, our DAC will map input [0, 0.129V) to 0V, [.129, .258V) to 0.129V, etc. The quantization error can be considered the actual analog signal minus the voltage it is mapped to. Thus, we can see that the max quantization error is one voltage step- here, 0.129V (ie, worst case is the actual signal is o[n] + 0.128999999V which gets mapped to o[n] where o[n] is the output of the DAC). Our max error in the signal will therefore be .129V / 2.64V = 0.0489 = 4.89%, comfortably within our desired 10% error margin for communication of the moisture data.

Ethics and Safety

Safety Standards: Following the IEEE guideline on the safety of our project, it is designed to be easy to use and compatible. It is safe to use and will not have the ability to cause property damage (IEEE Code of Ethics 7.8.9) **[1]**. Any high voltage components will be properly sealed to protect against the elements, to ensure user safety and prevent property damage. Informing the user how to use the device properly will be crucial to ensure safety and functionality.

Water Usage Regulations: In compliance with state regulations (e.g., Illinois watering restrictions), implementing features in our project to help prevent excessive water usage during specific times and seasons. Any rules or regulations on a state and local will be encouraged in the system (ACM 2.3) **[2]**. The user will be notified of these restrictions via push notification.

- Illinois: No watering is allowed between 10 a.m. - 4 p.m. in all areas during the period from May 15 through September 15.
- Champaign Water Restriction levels (Voluntary,Mandatory,Emergency) will be monitored and notify users to limit water use. **[4]**

Intellectual Property and Attribution: When developing unique Wi-Fi technology, it's important to credit others' work appropriately and ensure that our project respects existing patents and intellectual property rights. Properly citing and respecting the work of others helps maintain ethical standards (ACM Code 1.5). [**2]**

Privacy Concerns: Ensuring the privacy of users' data and information transmitted over the Wi-Fi network is crucial. Following privacy standards of ACM Code 1.6 **[2]**, each user has a right to privacy and privacy standards will be followed to ensure information accessible via the internet is protected. This will be done by safeguarding any of the user's personal information by following industry level privacy practices in our code.

Appendix 1

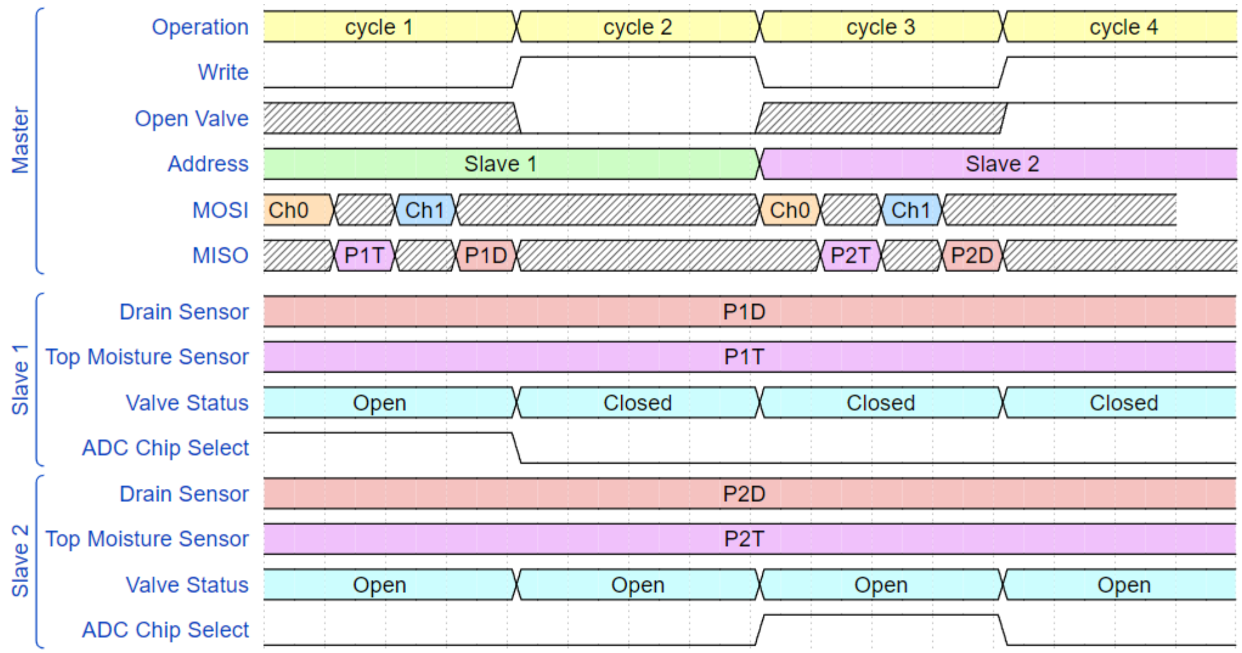
In the effort to make the system as scalable as possible, we will be using digital hardware to control the slaves instead of a microcontroller per slave. To implement this, we will be using a custom communication protocol outlined below. The system will operate by using digital logic to implement the following pseudocode for each slave:

```
if (address == myAddress):
        if (Write == 0):
                myChipSelect <- 1
        if (Write == 1):
                myValveState <- Open Valve
```

The chip select signal will be used to initiate communication between the master and slave via SPI communication protocol. Since we want to actually perform reads from two different channels (the top level moisture sensor and the drain sensor) during our overall read operation, each read operation will contain a brief “untoggle” where the control unit sets the address to a reserved non-operational value (ie 1111) which will ensure all slaves chip select signals are 0. The address will then be set back to the slave’s actual address and a new SPI conversation will be had, this time with Data In instructing the ADC to output data from the second channel. This was not shown on the waveform as it would require going down to the SPI clock level and get needlessly messy.

The valve state will be stored in a Flip Flop clocked by the AND of the address comparator and Write signal, so it will update when the master is writing to the particular slave and hold its value constant otherwise.

The master will be responsible for raising and lowering the Write signal, cycling through the slaves’ addresses, performing SPI communication protocol, reading the data on the buses when the Write signal is low, using the bus data and the user’s inputs to make a decision on opening or closing the particular slave’s valve, and outputting that decision when the Write signal is high. We will have to take care of the specific timings of when the master cycles, reads information, and outputs information after consideration of the worst case set-up and hold times of the digital hardware. A high-level waveform is shown on the next page.



Credit to Aliaksei Chapyzhenka (github: https://github.com/drom) for the waveform creator.

4 Citations

[1] "IEEE Code of Ethics." IEEE (Institute of Electrical and Electronics Engineers). https://ieee.org/about/corporate/governance/p7-8.html

[2] "ACM Code of Ethics and Professional Conduct." *ACM (Association for Computing Machinery)*. 2018. https://www.acm.org/code-of-ethics

[3] "Naturescape Blog: Regional Water Restrictions." *Naturescape Blog | Regional Water Restrictions.* 2015. www.naturescapedesigninc.com/regional-watering-restrictions.html#:~:text=It%20is%20unlawful%20for%20any,plants%2C%20or%20any%20other%20vegetation

[4] “Model Water Use Restriction Ordinance.” *Champaign County Regional Planning Commission*, CCRPC , 2013,
https://ccrpc.org/documents/model-water-use-restriction-ordinance/