# Design Document: Soil Moisture Controller (Pitched Project)

*This project is a pitched project idea by the U.S. Department of Agriculture's research laboratory on campus. It will be performed in partnership with a capstone team of students from the Department of Agricultural and Biological Engineering.*

## Introduction

### Problem

One of the biggest limiting factors for gains in agricultural productivity is the ability to provide sufficient moisture in the soil for the growth of crops. In particular, arid regions face the possibility of the occurrence of droughts that reduces the crop yields in dryland agriculture. To manage this issue, various water management strategies have been developed to ensure that there is sufficient water being applied over these crop lands. These irrigation systems have to provide control over the amount of water that is being applied over these crop lands such that optimal agricultural productivity is achieved while ensuring maximum water use efficiency.
Currently, the measurement of soil moisture content in pots are performed manually with individuals monitoring the moisture level based on weight, or the use of gravimetric sensors.

Upon irrigation, the weight or load of the pot would be at its maximum, and due to evapotranspiration over time, this weight would be lowered. When it eventually crosses a threshold set by the sensor, irrigation of the pots would be triggered again. However, due to the many different components that make up the weight of the pot, it is difficult to measure the exact proportion of increase in plant mass to the change in soil moisture content to obtain an accurate indication of when the irrigation has to be activated. As a result, there is a need for a more precise method to measure and maintain the soil moisture conditions in these pots through the use of soil moisture sensors. These soil moisture sensors would allow for the moisture that exists in the pot to be read so that sufficient irrigation is provided for consistent moisture.

### Solution

Our solution consists of a cheaper yet more effective device that provides constant moisture monitoring and water irrigation as needed to different types of soil. When the moisture within the substrate is below the predetermined target level (e.g. 25, 50, 75, or 100 percent), the water valve will be triggered to an extent where the moisture can be maintained at that level. In addition, there is also an interface so users are also allowed to check the current status of each pot, that is whether the substrate moisture is desirable as indicated by LED lights, and control the target level in the pot through a keypad.

We will work alongside the team of ABE students to construct the irrigation system and ensure that our solution could be scaled up for high-throughput of at least 50 plants in the future.

## Visual Aid/Physical Design

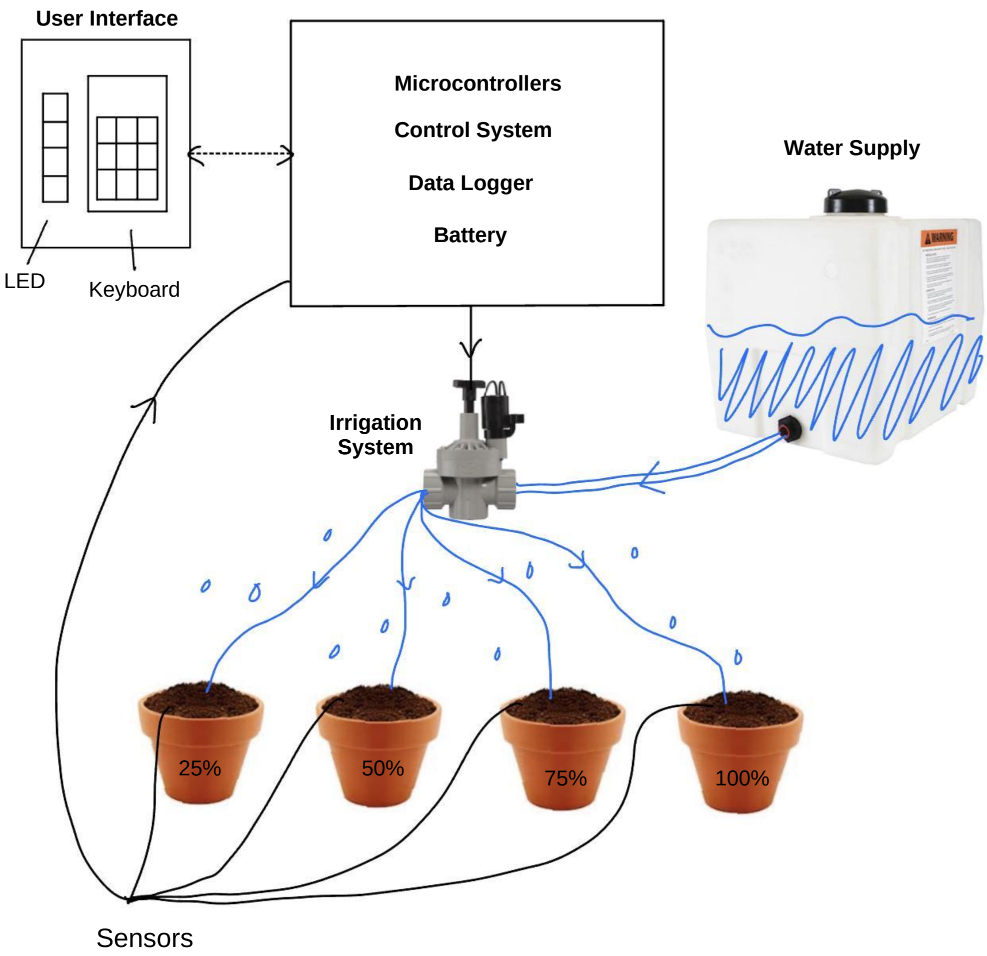

*Figure 1: Visual aid of our project*

## High Level Requirements List

These criteria for success are based upon the requirements articulated by the client (USDA) and jointly developed with the ABE team working together on this project.

- The moisture sensors should be able to detect the current level of moisture in the soil to an accuracy of +- 5% for the moisture level data to be logged on an SD card and displayed on an LED bar graph on an hourly basis

- The system should be able to provide irrigation when the moisture level falls beyond a set threshold level as inputted using a keypad by the user and maintain it to an accuracy of +-5%
- The system should be scalable to four different pots and the moisture level maintained at 100%, 75%, 50% and 25% in each of the respective pots

## Design

### Block Diagram

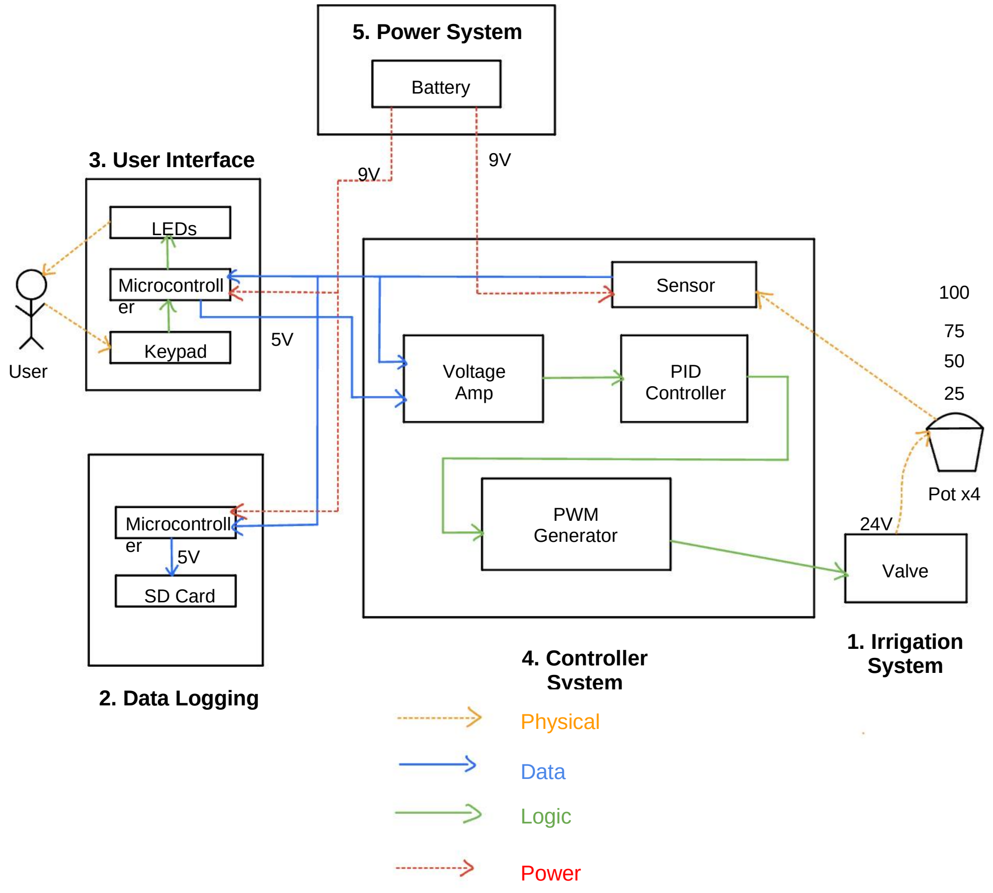

*Figure 2: Block Diagram*

As shown in *Figure 2*, our project can be broken down into five subsystems: irrigation, data logging, user interface, controller, and power. The irrigation system consists of a valve, pipes, and water supply that will be used to water 4 different pots of soil in order to maintain a specific moisture level in each. We will work with the ABE team to design this. The data logging system will collect and store data such as temperature, weight, etc. every hour, and consists of a microcontroller and SD card. The user interface allows users to type on a keypad to manually control the watering/moisture level monitoring of a pot (if necessary) and LEDs as a display to see the moisture level of each pot. The controller system consists of a sensor, voltage amp, PI controller, and PWM generator. It will take in moisture level and user data as input, and output a signal that tells the water valve how much water should be given to each pot. Finally, the power subsystem is made up of a 9V battery, but we are open to exploring other options to supply power to our system.

A further breakdown of our block diagram tied to our visual aid can be found below in *Figure 3*.

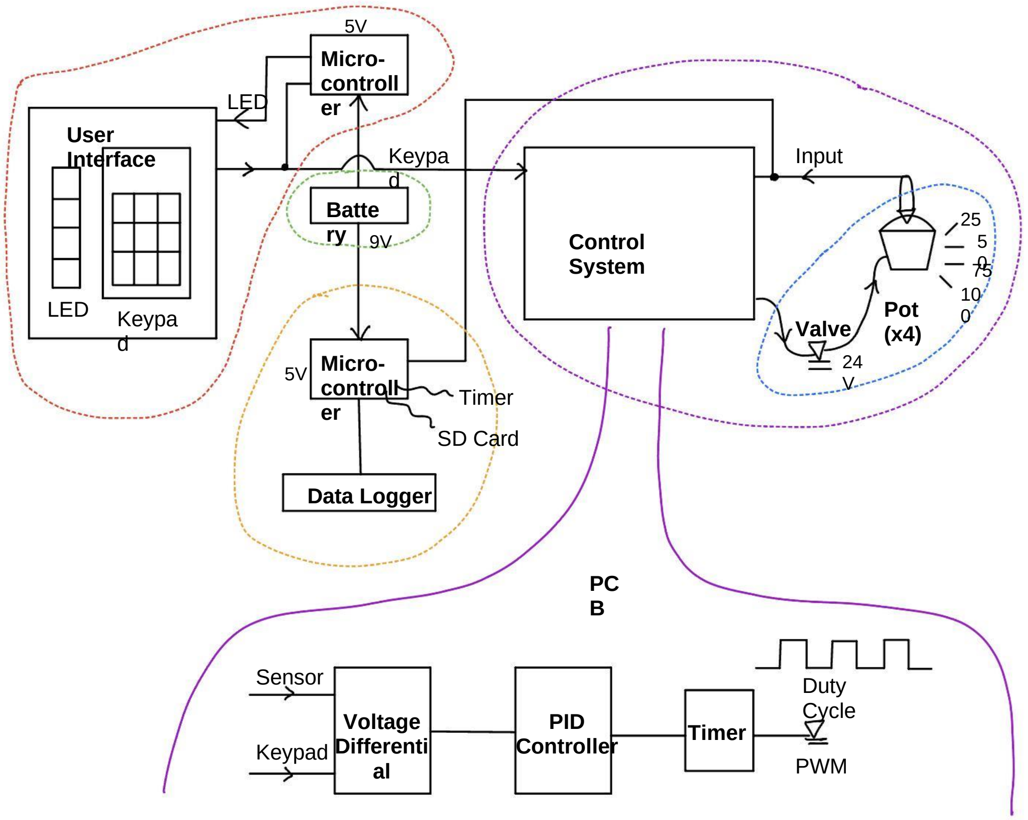

*Figure 3: Further breakdown of block diagram and visual aid*

## Physical Design

The purpose of the system is to monitor the volumetric water content (VWC) of the soil in each plant pot during the course of Matthew Brooks' drought experiments. This system will supply irrigation to each pot individually in order to keep it at the VWC desired during the course of the experiment. The system favors precision, in which all pots with the same desired VWC have very similar actual VWCs, over accuracy, in which all pots of the same desired VWC are within an acceptable range but may vary in actual VWC between individual pots.

The experiments this system will serve will involve plants growing in pots of the same size within a Conviron PGC Flex growth chamber. Each plant will be grown in the same substrate. Plants will begin at the same VWC at the beginning of the experiment, then will be divided randomly into four categories, each category at a different VWC. The plants will be held at this VWC for the remainder of the experiment. Pots within each category will be distributed across the growth chamber and will not be bunched together. Within each category, plants will be held at four different nutrient levels through weekly applications of fertilizer.

Each pot will have one ECH2O EC-5 soil moisture sensor in it. The sensors will be calibrated to the specific substrate being used in the experiment. This calibration will involve using the sensor in the desired substrate at known VWCs and using those readings and known VWCs to create a calibration curve. This calibration curve, based on the measured capacitance of the soil, can be used to determine the VWC in each pot with that same substrate. Should it be deemed necessary through further research, there will be four different calibrations done for the varying nutrient concentrations in each pot to avoid any discrepancies based on differences in ion concentrations. The exact orientation of the probe in each pot will be determined through further research.

There will be solenoid valves connected to each pot that will allow irrigation to be fed to pots individually. The valves will be connected to a drip emitter or similar device that will allow irrigation to be fed evenly and in such a way that soil compaction is minimized. The evenly fed water will minimize the chances of the sensors being located in uncharacteristically dry or wet patches within the substrate, preventing falsely inflated or deflated VWC readings. The minimization of soil compaction will help prevent any changes in the sensors' accuracy over time, as compaction may alter the measured electrical conductivity of the soil at a given moisture level. The exact method of water delivery will be determined through further research.

The sensors will check the VWC in each pot regularly. The exact time interval will be decided after further research. Should a pot require more water to reach its target VWC, the solenoid valve for that pot will be opened for enough time to deliver the necessary amount of water.

The soil sensors and water valves will be housed within the growth chambers as well as any wiring and pipes that must connect to them. These wires and pipes will be fed through the two 2" instrumentation ports located on the growth chamber doors. The water pipes will likely connect to an overhead water source currently in place in the room housing the growth chambers. The wiring will connect to the rest of the control system, which will be housed elsewhere in the room containing the growth chambers.

This system must be able to be moved between growth chambers between experiments. Ideally, this system will allow for plants to be removed from the system temporarily during the course of an experiment, though this is not a requirement.

Together with the ABE team and USDA, our team paid a visit to the growth chambers in the Edward R. Madigan Laboratory (ERML). The pictures of the empty growth chambers where the pots and the system would be placed are shown below. A more complete picture of the physical design will be obtained when the ABE team sets up the pots and the system is incorporated into the chambers.

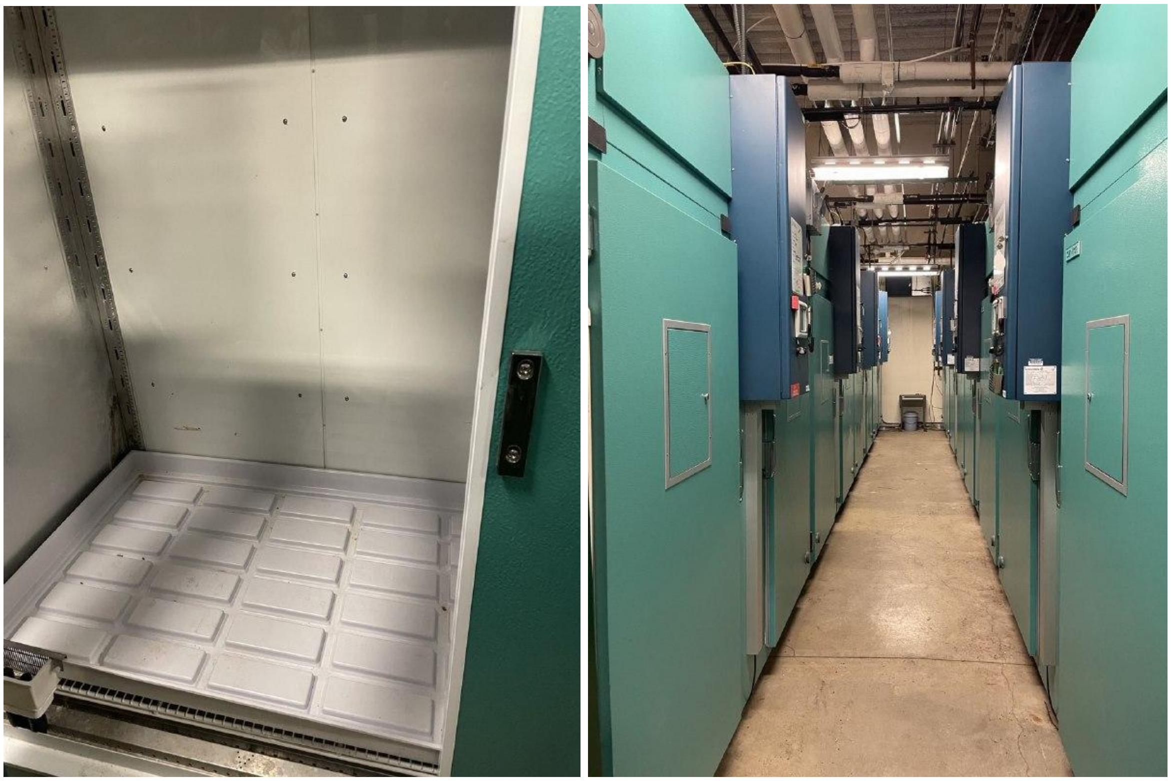

*Figure 4. Growth chambers in the ERML*

### Block Descriptions

Subsystem 1: Irrigation Subsystem

Irrigation is the process of artificially applying controlled amounts of water to land or crops. This is done by using valves as well as a system of tubes and pumps to bring in water from pipes, canals, sprinklers, and other mean-made water sources, instead of relying on rainfall. For this project, the irrigation subsystem for each soil pot would consist of a valve that would open and close based on the moisture level measured, in order to maintain a desired set of moisture conditions for different soil and soilless substrate mixes. Irrigation is needed in a given pot if it is sensed that the moisture level falls below a certain value (for example, below 75 for fine soil). When this happens, relay switches activated by a microcontroller, such as an Arduino, will operate the irrigation valves (likely 24V) that correspond to each sensor-controlled pot, and water will flow out until the soil reaches an ideal value again.

| **Requirement** | **Verification** |
|---|---|
| Valves will open/close to dispense a correct amount of water to the appropriate pot based on the signal received from the controller | - When the 24V valve for a pot receives a signal that irrigation is needed, it should flip open like a lid and water flows out. Once moisture level is reached, the valve should close within 1 second. Overall, when irrigation is not needed, valve should remain closed.<br>- Valves are powered when closed, non-powered when open. Use a multimeter to measure if there is any power (>0.5W) in a valve currently dispensing water versus in a valve that does not need to dispense water (should be 0W). |
| The irrigation system should be scalable so that at least 4 pots can be watered in by respective amounts at the same time (within 5 minutes for 4 pots) | - Set up 4 pots with different soils that require 4 different moisture levels (eg. 25%, 50%, 75%, 100%). Make sure all are under-watered and need irrigation<br>- Plug in SD card and run program<br>- All 4 pots should be watered at the same time so the display reads their correct desired moisture level within 5 minutes (start to finish)<br>- The irrigation will either happen (1) simultaneously through 4 different valves, or (2) consecutively with a queue based on delta of water needed. Decision TBD by ABE team (Eg for scenario 2: Pot 1 is at 60% and needs 75% moisture is higher priority and would be fully watered before Pot 2 at 45% that needs 50% moisture) |
| Water will be pumped from a supply tank using a series of tubes until a predetermined desired moisture level between 0 and 100 is reached for a given pot | - Connect the control system to the irrigation system. Irrigation system includes pipes and valves, and water is pumped from an external source<br>- Test using one pot of soil that is below its desired moisture level, for example, 50%. Water should continue being pumped into the pot until the sensor detects that it has reached 50% again<br>- For a sanity check, the user should be able to |

| | |
|---|---|
| | see the moisture level for that Pot say 50% on the LCD display |

Subsystem 2: Data Logging Subsystem

The data acquisition subsystem will consist of a data logger, an instrument that monitors and records changes in conditions over time. Most data loggers can accept two or more types of input, so we would program ours to take inputs such as voltage, current, temperature, etc. The data logger will ultimately communicate the need for irrigation by measuring and recording calculated factors like volumetric water content for each soil, and generating a list of plants that require irrigation. Then, this list of plants will be sent to the microcontroller that carries out the irrigation process for the relevant plants by using a pulsing I/O signal of either 0 or 5V to communicate whether or not irrigation is needed. There are many expensive existing data loggers such as the CR100, but we would want to buy or build one that is still battery-powered and effective for a cheaper price. One option that nicely interfaces with an Arduino microcontroller would be to create a data logger from scratch using a data-logging shield, coin battery, and SD card [1].

An important note for calculations is that a standard microSD card has an operating voltage of 3.3V, so we cannot connect it directly to circuits that use 5V logic or the card can be permanently damaged. The microcontroller we will have for this subsystem will include an onboard ultra-low dropout voltage regulator and logic level shifter chip that creates safe communication. It will also include a microSD card socket on the front where our memory card should be able to interface with it easily.

| **Requirement** | **Verification** |
|---|---|
| Data logger should send a pulse I/O of 0 or 5V to signal to the controller whether or not irrigation is needed for each pot | - Test using an LED (controls is the part that actually activates irrigation valve opening/closing, and irrigation is dependent on the ABE team, so we just need a way to simulate if a pulse of 0 or 5V is being sent for data logger)<br>- Using a multimeter, measure the voltage of sensor for one pot when it is at an ideal irrigation level (should 0V if no irrigation needed)<br>- After water is removed so that moisture is below a level and irrigation should be needed, measure if the voltage of signal is 5V<br>- First: LED should light up if around 5V or stay off if it’s not<br>- Second: Data logger should show on LCD |

| | |
|---|---|
| | display what irrigation level is sensed for each pot (eg. Pot 1: 70%) |
| Record data every 60 minutes* (sample rate) for each pot and store on an SD card<br>*Note: Our sample rate is subject to change, based on the USDA's requests | - Allocate a few hours of time to be able to check the data logger each hour (eg. 4pm-6pm, 12-9pm, etc)<br>- Make sure SD card is plugged into the card socket on the front of the Arduino<br>- Start running Arduino program from scratch<br>- Plug in SD card to computer to extract the data then read data from SD card with Arduino → file.read() method → results get printed in a text or CSV file<br>- Verify if data is actually printed with timestamps 60 minutes apart |
| Data logger should be able to read more than 2 inputs such as voltage, current, temperature, etc. at once | - Set up data logger to take readings every 10 seconds of 2 inputs from the sensor, such as temperature and voltage<br>- Over 24 hours, starting on a specific day, begin running the system. We know the logger should produce 8640 data points (1 day = 86400 seconds)<br>- Logger should produce data for each point (1 rows/point)<br>- Extract data into CSV file with temperature and voltage as the columns if correct |

Subsystem 3: User Interface Subsystem

The user interface subsystem would consist of two main components. The first component would be a 10-segment LED bar graph that shows the soil moisture level as detected by the sensors in each pot. This would give the user visibility of the soil moisture level in the pot at any point in time instead of having to check on the data log in the SD card.

The second component would be a 4x4 keypad where the user is able to input the desired soil moisture level that the user would like the pot to be maintained at. The compact size of the keypad would be beneficial in a greenhouse setting where the user would not have to rely on inputting this data via a laptop.

An Arduino Uno microcontroller can be used for interfacing between the soil moisture sensors, 10-segment LED bar graph and 4x4 keypad.

| **Requirement** | **Verification** |
|---|---|
| The 10-segment LED bar graph must receive and display data from the microcontroller for visualization of soil moisture level data collected by the sensors in the pot. | Connect the 10-segment LED bar graph to the Arduino microcontroller. Verify that the 10-segment LED bar graph is able to display the correct amount of soil moisture within a 10% range. |
| The user must be able to input any percentage value of soil moisture level from 0 to 100 using 4x4 keypad. | Connect the 4x4 keypad to the Arduino microcontroller together with the 10-segment LED bar graph. Verify that the value that is entered in the keypad is reflected accurately by the 10-segment LED bar graph. |
| The microcontroller must be able to send and receive data to and from the voltage comparator in the controller subsystem. | Test the situation where the user enters a soil moisture value that is below the current soil moisture level detected by the sensors. Ensure that the water valves in the irrigation subsystem does not open.<br>Test the situation where the user enters a soil moisture value that is above the current soil moisture level detected by the sensors. Ensure that the water valves in the irrigation subsystem opens within one minute and subsequently maintains the soil moisture level at the level set by the user. |

Subsystem 4: Controller Subsystem

In order to efficiently gain the desired substrate moisture level, we decide to implement a PID controller which takes the feedback input from the moisture sensor, compares the measured value with the desired value, and triggers the water valve if the measured value is below the desired value.

The value from both the moisture sensor and user input will be sent to a differential amplifier that outputs a voltage proportional to the voltage difference. The filtered voltage will then be inputted to the PID controller which consists of potentiometers for tuning the controller, inverting op-amps for amplification, and capacitors for implementing the integrator, and derivative circuits.

Finally, the output of the controller will be amplified and connected to a LM555 Timer chip in order to generate a PWM signal to the water valve so that the amount of water being given is sufficient to each pot.

| **Requirement** | **Verification** |
|---|---|
| The voltages from the moisture sensor and user input must be calibrated to the same scale in order to acquire a reasonable voltage difference in a linear range from 0-5V. | Connect the scaled outputs of user input and moisture sensor to a digital multimeter to check the output voltage. Verify that the voltage of both inputs can be linearly converted to moisture percentage. The characteristics between the readings and voltage can be obtained from manually measuring the voltage at each reading and plotting the graph between the two variables. An additional amplifier and voltage divider would be needed to adjust the voltage offset and range from both sources. |
| The control system must be able to remain stable. | Test the stability of the control system by connecting the power supply as a temporary input to the system and oscilloscope to the output of the system to check that the current and voltage does not become unstable. As a safety measure, connect the voltage clipper circuit to the system output so that the voltage is only limited to 5V. |
| The controller must be able to send the information on the amount of water needed for each pot so that the relative error would be less than 5% within ten minutes. | Obtain the amount of water being transmitted to the substrate by connecting the power supply and timer chip to the valve input and varying the duty cycle of the signal into the valve. Collect the data for the water needed to fill the pot at each moisture baseline (25%, 50%, 75%, and 100%) to gather an estimated amount of water to increase the moisture percentage by one percent. Verify by inputting the desired duty cycle and compare the moisture percentage of the substrate after ten minutes have passed. |

Subsystem 5: Power System

The power subsystem would consist of 9V batteries that are rechargeable and provide power to the other subsystems like user interface subsystem, data log subsystem and controller subsystems. The power subsystem must be capable of outputting sufficient current such that all devices can be powered consistently.

For the Arduino microcontrollers in the user interface and data log subsystems, the 9V battery can be used together with a snap-in connector and a DC barrel jack. The Arduino microcontrollers accept 7-12 V DC through the power jack and the onboard voltage regulator steps down the voltage to the required 5 and 3.3 V DC. This would allow for the data log and user interface subsystems to be used as portable devices without a need for mains voltage, which is often the situation in a greenhouse.

For the controller subsystem, it is likely that a higher voltage of around 24 V DC is required for the irrigation valves. To step up this voltage, a voltage booster can be used.

| **Requirement** | **Verification** |
|---|---|
| The batteries should output the required 24V, 15V and 9V (± 0.1V) needed to power the irrigation subsystem, controller subsystem and user interface/data logging subsystem operation respectively | Use an oscilloscope to check that the battery and voltage converter output is 24V, 15V and 9V (± 0.1V) respectively |
| The batteries should be able to recharge to its full capacity within 5 hours. | Place the rechargeable battery with the charger and check if full-capacity charge is achieved after 5 hours. |

## Tolerance/Risk Analysis

The most challenging part of the project is to construct a closed-loop control system that satisfies all the subsystem requirements, which are the ability to remain stable within 5% from the desired level. Since the PID controller will be implemented with operational amplifiers (op-amps), resistors, and capacitors, majority of the tuning will be done by adjusting the resistance, capacitance, and ratios between resistance on each side of the amplifier. In order to test the feasibility of the control system itself, a simulation of the closed-loop system in response to a unit step function is run on LTSpice. All op-amps in the simulation are in the LM741 model.

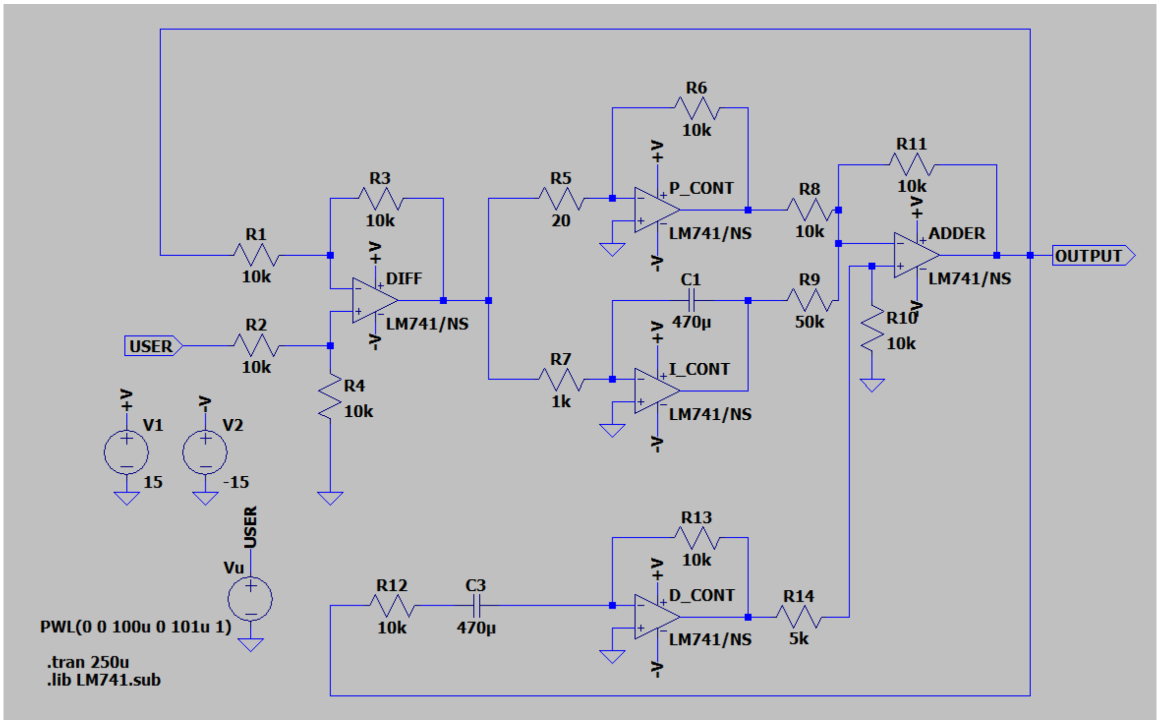

*Figure 5: Closed-loop control system circuit for simulation*

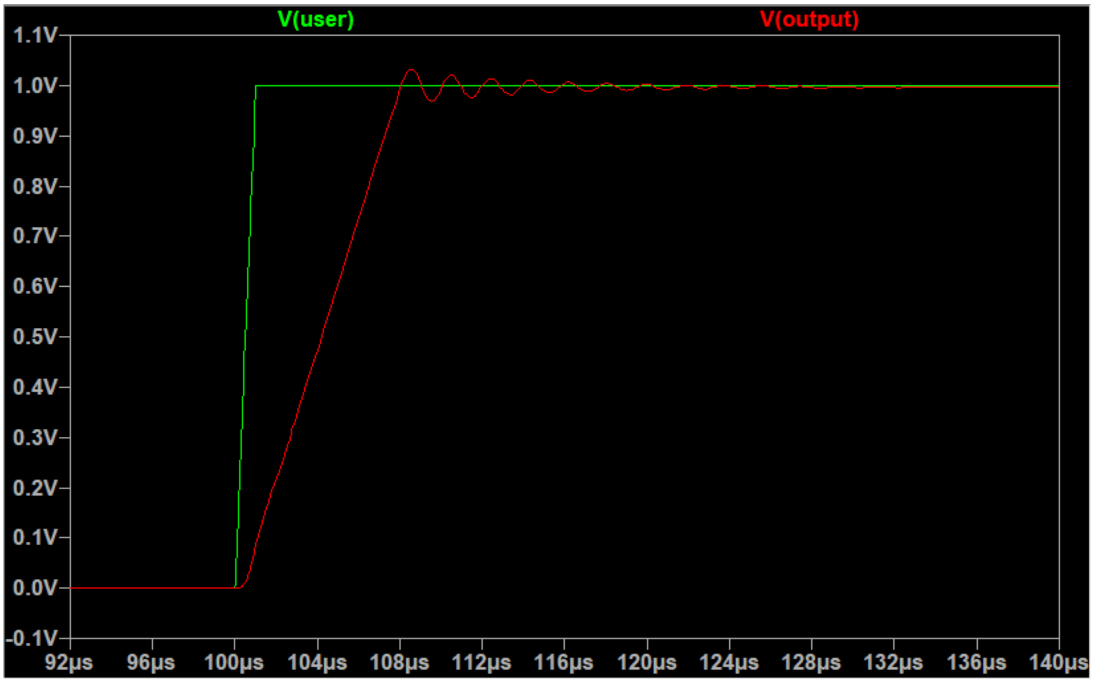

*Figure 6: Simulation Result*

From the simulation result above, it can be seen that the output can track the unit step from the input, and the output remains stable within the required tolerance over 8 microseconds. For this simulation, it is important to note that the moisture sensor and water valve are assumed to be linearly proportional to the output of the control system. Additional tuning would be needed after connecting the system with both components. The methods which will be used to tune the system are as follows:

- If the resistance used in the simulation does not exist, the scaling of the resistances across the op-amp can be done so that the ratio remains constant.
- If the scale of the moisture sensor and user input does not match, the ratio of the resistance in the first op-amp (or voltage differential op-amp) can be adjusted.
- If the offset or range of the control system output does not fit with the valve specifications, additional voltage divider circuit and op-amp can be connected to fix both issues.
- If the steady state of the control system is out of the tolerance limits or the response time is too long, higher proportional gain is needed.
- To fix the instability problem, lower derivative gain is needed. To reduce the oscillation or overshoot of the system, lower proportional and integral gains are needed. The gains can be lowered by increasing the resistance in each part.

## Cost and Schedule

### Cost Analysis

**Labor**

The average starting salary of ECE graduates is $80,000/year, or $40/hour. Assuming that each team member works 100 hours total, this would amount to ($40/hour) x 2.5 x 100 hours to complete = $10,000 per person. Multiplying this by the number of team members, $10,000 x 3, the total cost of labor for our group would be $30,000. Additionally, we will be working alongside a team of 3 ABE students that will be paid separately through their department.

**Parts**

| Description | Manufacturer | Quantity | Unit Price ($) | Cost ($) | Link |
|---|---|---|---|---|---|
| 3/4-inch Solenoid Globe Valve AC 24V | Galcon | 4 | $31.96 | $127.84 | Link |
| 3/4-inch Clear Vinyl Plastic Tubing Flexible (10 feet) | Eastrans | 2 | $11.39 | $22.78 | Link |
| 40 pcs 304 Stainless Steel Hose Clamp | Selizo | 1 | $8.99 | $8.99 | Link |
| ECH20 EC-5 Soil Moisture Sensor | Meter Group | 4 | $145.00 | $580.00 | Link |

| | | | | | |
|---|---|---|---|---|---|
| DS1307 for Arduino Uno Data Logging Shield (3pcs) | AITRIP | 1 | $10.99 | $10.99 | Link |
| 32GB SD Memory Card | LaView | 1 | $8.99 | $8.99 | Link |
| 10-segment LED bar graph | Lite-On Inc. | 4 | $1.38 | $5.52 | Link |
| 4x4 keypad | Parallax Inc. | 1 | $9.27 | $9.27 | Link |
| Arduino Uno R3 ATMEGA328P Eval microcontroller | Arduino | 2 | $27.60 | $55.20 | Link |
| Breadboard | DFRobot | 1 | $2.90 | $2.90 | Link |
| Resistor 10K Ohm 1/4 Watt PTH - 20 pack (Thick Leads) | Sparkfun | 1 | $1.25 | $1.25 | Link |
| LM358DR2G Operational Amplifier | Digikey | 12 | $0.48 | $5.76 | Link |
| NE555DR Timer | Texas Instruments | 4 | $0.46 | $1.84 | Link |

| | | | | | |
|---|---|---|---|---|---|
| 470 µF 16 V Aluminum Electrolytic Capacitors Radial | Rubycon | 8 | $0.48 | $3.84 | Link |
| 10k Ohm Linear Rotary Potentiometer | Sparkfun | 24 | $1.05 | $25.2 | Link |
| 9V rechargeable battery | Energizer Battery Company | 4 | $19.62 | $78.48 | Link |
| 9V Energizer Recharge 4 Battery Black Universal Battery Charger | ACE Hardware Store | 1 | $35.99 | $35.99 | Link |
| Snap in connector for batteries | SparkFun Electronics | 4 | $3.50 | $14.00 | Link |
| 9V to 24V voltage converter | XP Power | 1 | $88.88 | $88.88 | Link |
| 9V to 15V voltage converter | Recom Power | 1 | $28.56 | $28.56 | Link |

*Figure 6: List of estimated parts and costs to complete the project*

**Grand Total**

Labor: $30,000
+ Parts: $1,116.28

The total cost will be $31,116.28.

### Schedule

| **Week** | **Task** | **Team Member** |
|---|---|---|

| | | |
|---|---|---|
| **February 27th-March 3rd** | **Design Document Check - 2/27** | Everyone |
| | Start PCB design for the controller | First |
| | Research on how to connect the microcontroller to 4x4 keypad and LED bar graph for user interface | Ren Yi |
| | Research data logger connection to microcontroller | Isabel |
| | Finalize irrigation system design with ABE group; order parts | Everyone |
| **March 6th-10th** | **PCB Order - 3/7** | Everyone |
| | **Teamwork Evaluation I - 3/8** | Everyone |
| | Work on a basic prototype of the user interface | Ren Yi |
| | Test the battery connections to microcontrollers, control system and valves | Ren Yi |
| | Tune the controller for stability | First |
| **March 13th-17th** | Work on a basic prototype of the user interface | Ren Yi |
| | Test the battery connections to microcontrollers, control system and valves | Ren Yi |
| | Work on prototype of data logger, reading mock data from one pot of soil | Isabel |
| | Document the range of input and output of the controller | First |
| | Combine the user interface | Ren Yi, Isabel |

| | | |
|---|---|---|
| **March 20th-24th** | and data logger with moisture sensor | |
| | Combine the irrigation subsystem with the controller | First |
| | Design container for prototype to fit in for protection in plant chambers; order parts | Everyone |
| **March 27th-31st** | **PCB Order - 3/28** | Everyone |
| | **Individual Progress Report Due - 3/29** | Everyone |
| | Combine the sensor with the controller and irrigation system for real moisture input | Everyone |
| | Scale data logger to read from 4 different pots of soil through sensors | Isabel |
| **April 3rd-7th** | Refine prototype to make it more efficient and effective | Everyone |
| | Build container for prototype to fit in for protection in plant chambers | Everyone |
| **April 10th-14th** | Test the soil moisture controller with potted plants | Everyone |
| | Assemble prototype in container | Everyone |
| | Film video of working system for mock demo | Everyone |
| | Fix minor errors | Everyone |
| **April 17th-21st** | **Mock Demo - TBD** | Everyone |
| | Fix any remaining errors | Everyone |
| | Test the soil moisture | Everyone |

| | | |
|---|---|---|
| | controller with potted plants | |
| | Finalize assembly | Everyone |
| | Re-film video for final demo | Everyone |
| **April 24th-28th** | **Final Demo - TBD** | Everyone |
| **May 1st-5th** | **Final Presentation - TBD** | Everyone |
| | **Final Paper Due - 5/3** | Everyone |

**<u>Ethics and Safety</u>**

One of the main ethical standards that should be followed in this project is keeping our project ideas original and different from other ideas that are already found on the market. There should not be any plagiarism of product designs and other applicable sources found during the research process should be properly cited and credited. (IEEE Code of Ethics II.5). In the context of our project, other automated soil moisture-based irrigation control technologies that utilize smart watering to tailor irrigation schedules to meet the water needs of the area are present. Careful consideration should be made to keep our project idea distinct from these existing technologies, in terms of the methods we use to achieve the implementation of such a system. Additionally, the IEEE Code of Ethics I.5 states that the claims and estimates based on available data should be honest and realistic. As our project would be dealing with making soil moisture data available for the user to make decisions on irrigation schedules, our team pledges to ensure the solution is sufficiently tested such that the data shown by the sensor system is accurate and precise. This includes a comparison of the water moisture level detected by the sensor across multiple occasions such that the measurements made by the sensor are repeatable.

Specific safety concerns in our project should also be addressed. Upon review of the laboratory safety regulations as set by the Division of Research Safety in the Office of the Vice Chancellor for Research and Innovation, our team pledges to follow these regulations strictly in the lab sessions that we engage in during the semester. Our team would always work in a team of at least two people, report any broken equipment immediately, clean up after each lab session, and avoid bringing any food into the lab (ECE 445 p.3). One of the potential safety risks that might be encountered during the project would be the possibility of short circuits when the electronic components are wired to a power source resulting in damage to all the combustible crop around the soil moisture controller. Accidental skin contact with the circuit with moisture would also increase the risk of dangerous electric shocks through our body. Special attention would be paid to ensure that electrical systems are kept entirely separate from any external irrigation systems that release water over croplands.

## **Citations and References**

1. Ortiz, Diego, et al. "A Cost-Effective and Customizable Automated Irrigation System for Precise High-Throughput Phenotyping in Drought Stress Studies." *PLOS ONE*, vol. 13, no. 6, 2018, https://doi.org/10.1371/journal.pone.0198546.
2. ECE 445 Lab. Lab : ECE 445 - Senior Design Laboratory. Retrieved February 2, 2023, from https://courses.engr.illinois.edu/ece445/lab/index.asp
3. IEEE code of Ethics. IEEE. (n.d.). Retrieved January 24, 2023, from //www.ieee.org/about/corporate/governance/p7-8.html.
4. Liao, R., Zhang, S., Zhang, X., Wang, M., Wu, H., & Zhangzhong, L. (2021). Development of smart irrigation systems based on real-time soil moisture data in a greenhouse: Proof of concept. *Agricultural Water Management, 245,* 106632.
5. Team, Arduino. "Guide to Arduino & Secure Digital (SD) Storage.: Arduino Documentation." *Arduino Documentation | Arduino Documentation*, https://docs.arduino.cc/learn/programming/sd-guide.
6. Agnihotri, Nikhil. "How to Log Sensor Data Using SD and Micro SD Card with Arduino." *Engineers Garage*, 2023,https://www.engineersgarage.com/arduino-sd-card-micro-sd-card-reader-sensor-data-logging/#:~:text=Reading%20data%20from%20SD%2FMicro%20SD%20card%20involves%20use,number%20of%20lines%20or%20until%20the%20file%20ends.
7. Dahlen, Aaron. (2005) "The PID Controller – Part 1" https://www.nutsvolts.com/magazine/article/the_pid_controller_part_1