

ECE 445
Senior Design Laboratory
Design Document

DIY PLANTIFY

A Self-Moving Potted Plant System For Ideal Sunlight Exposure

Team No. 70

Maya Kurup
(mayaek2@illinois.edu)
Joshmita Chintala
(jchint2@illinois.edu)
Hongshang Fan
(hf7@illinois.edu)

TA: Raman Singh

Professor: Olga Mironenko

February 23, 2023

Abstract

This document will provide a more in-depth look into our senior design project, DIY Plantify. This document will consist of our problem statement, solution, design in further detail, cost and schedule, and ethical/safety considerations related to our project.

Table of Contents

1 Introduction	4
1.1 Problem	4
1.2 Solution	4
1.3 Visual Aid	5
1.4 High-Level Requirements	6
2 Design	7
2.1 Block Diagram	7
2.2 Subsystem Overview	8
2.3 Subsystem Requirements	9
2.4 Tolerance Analysis	11
3 Cost and Schedule	12
3.1 Cost Analysis	12
3.2 Schedule	13
4 Ethics & Safety	15
References	16

1 Introduction

1.1 Problem

At the root of every plant, it needs 5 different components for it to grow, survive, and thrive: light, air, water, nutrients, and space to grow. In people's day-to-day lives, there are not many systems put in place to help those individuals understand how much sunlight a plant needs, when the plant needs it, and how much of it they need. As well, there aren't many systems in place to understand how much water a plant needs, when it needs to be watered, and if you are adding enough. So, a solution to resolve these issues can be very beneficial in people's day-to-day lives when growing plants (simple leaf plants, trees, fruits, or even vegetables) on a smaller scale, but can also be extended to a professional/advanced level that farmers and larger industries can use.

1.2 Solution

A solution for this issue is to create a system in which a light sensor is connected to a pot, and this can detect how much sunlight that plant is retaining. Once that sensor sees that the sunlight exposure is too low/high based on what the plant needs, it will alert the system and user. In this system, we also want to implement a chassis with motors/moving robots beneath this pot that can move this pot in different locations around the room (chassis - similar to a Roomba-vacuum moving system). With the combination of this light sensor and a moving chassis, we can feasibly make a product that can be applied and used in people's day-to-day life that will optimize the amount of light a plant receives.

With a proper search procedure to find the right spot of light, we can let the robot record light data along the route every hour during the day, and find the location with the most light every hour, then use this data to control the robot. We would then update this data every month or seasonal change.

Based on the timeline of our project, we can foresee that maybe we will have time to make further implementations of this product. An additional component we would implement is an alert system that would tell the user when moisture levels are too low in a plant. This pot would use another sensor (depending on the route of how we would want to do it - weight measuring sensor, moisture control sensor, etc) to detect how much moisture is in the pot. All in all, we would create a system that optimizes sunlight and also warns the user about low moisture levels for small plants and can further be extended to larger systems, which would help everyone at a local level and professional/worldwide level.

1.3 Visual Aid

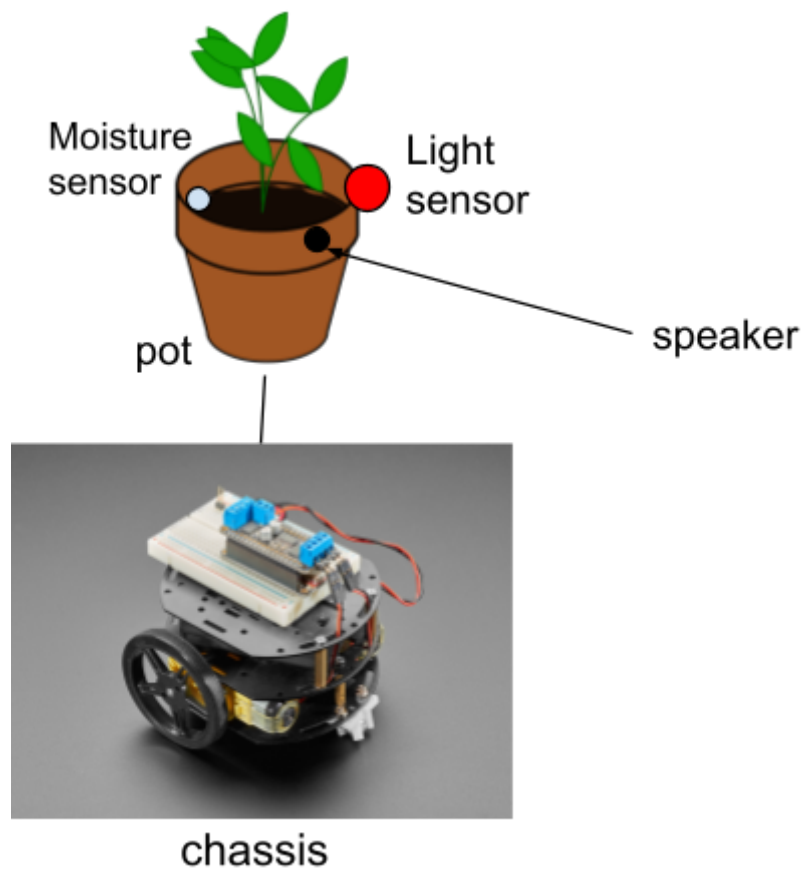


Figure 1: Overview of the DIY Plantify System

1.4 High-Level Requirements

In order to successfully obtain a fully working product by the end of this course, we plan to reach the following goals.

1. The system should be able to successfully track the light sensor data. With initial setup testing, we should be able to understand where the greatest amount of sunlight and the smallest amount of sunlight can be found in a certain room. This can be calculated by UV/heat indexes (0-11+) provided during each time of each day, specific luminosity required by each plant (75-250 fc), and the degree at which shadows form on a surface. Data gathered should be recorded in a table, and graphs/figures can be made from the results.
 - a. Determine what kind of plant would be ideal in this scenario (indoor vs outdoor, small vs big; flower vs leaf vs vegetable plant)
2. Create a moving robot/chassis able to move to a location (on both an X & Y axis) with the most light in any given room. Create different testable variables to determine how fast (constant rate between 0.5 -1.2 mph), when (times of sunlight exposure), and where (set range limits for how far it can move in the X or Y axis) this chassis should move.
3. A notification system should be configured so that the user will be notified when the plant needs to be watered. A sound alarm (able to be heard from 15-20 feet away) system will be implemented into the microprocessor so that the user will be notified when the plant needs to be watered.

2 Design

2.1 Block Diagram

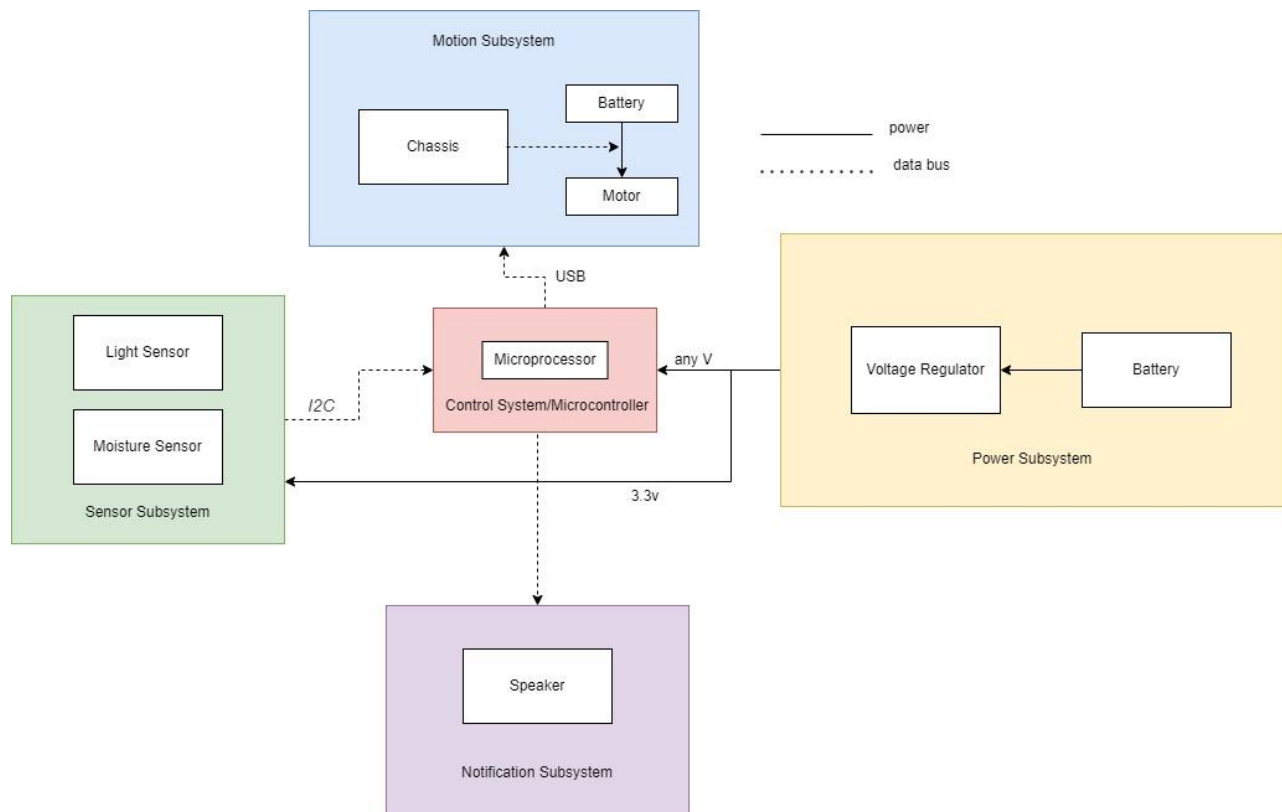


Figure 2: Block Diagram for the DIY Plantify System

2.2 Subsystem Overview

Sensor subsystem

Sensor subsystem consists of a light sensor and moisture sensor. The light sensor will record comparative light intensity data during the initial setup and will continue to track results based on time of day, obstruction due to shadows, and seasonal changes. The moisture sensor will watch out for low moisture levels and will alert (Notification subsystem) the user when the plant needs to be watered with the sound of an alarm.

Requirements	Verification
The moisture sensor must be able to detect dryness within 2 days, and never send an alarm when not dry.	Verify the moisture sensor is working properly by taking moisture level measurements consistently throughout a week, and fact-check those levels with online resources and verifications of the plant itself.
The light sensor should find the location with the most light within 15 cm of error.	Continuously record the light sensor data over a weekly period, and make sure measurements being recorded are consistent and accurate with that daily sunlight emissions.

Motion subsystem

The motion system is a chassis robot carrying the pot. Microcontroller will send commands to it to move the plant to the desired position.

Requirements	Verification
The chassis must be able to move at a constant speed(constant rate between 0.5 -1.2 mph) to ensure the plant does not fall from an accelerated force. As well, measurements can be recorded at a constant rate.	Record the speed at which the chassis moves using IO Lab or some sort of measurement tool that defines velocity and acceleration.
The wheels should have an ideal amount of torque based on the size of the plant, in order for the plant to not fall.	Calculate the torque using the equation: $Torque = rF\sin(\theta)$, and solve for Force using IO Lab.

The platform should be large and durable enough to hold a small/regular sized potted plant.	Platform carrying plant can be pre-determined, so make sure the plant being bought fits the size of this platform (diameter of potted-plant base should cover maximum 95% and minimum 45% of the surface area.)
---	---

Notification subsystem

The notification subsystem will consist of a speaker. The speaker will sound an alarm when the moisture sensor thinks it's too dry.

Requirements	Verification
Must be able to make some noise (audible to the user) when receiving a signal of dryness.	Connect the speaker to the processor and let the processor send the signal. See if the speaker makes some noise.

Power subsystem

Power subsystem consists of a battery and voltage regulator. It handles power regulating and powering other subsystems.

Requirements	Verification
The system must provide over-current, over-voltage, under-voltage, and short-circuit protection.	Connect the voltage regulator to various circuits with varying conditions, see if the regulator can shut down in time.
It must provide a stable 3.3 ± 0.1 V power source.	Check voltage using a multimeter, and ensure that the proper amount of power is being supplied.

Microcontroller

Requirements	Verification
Able to store 10 sets of data gathered for each hour during initial setup.	Check the memory space after an experimental initial setup.
Able to send signal of current desired location to chassis.	Give the chassis a set of data and check whether it carries desired location according to data.
Able to analyze data from moisture sensor and send a signal to the speaker when indicated by the data.	Put the moisture sensor into a pile of dry paper and check if the speaker will beep.

2.4 Tolerance Analysis

Light Sensor: The light sensor may run into errors during testing; data points measured may have %error, based on shadows, time of day, and direct sunlight obstruction may also propose issues. As mentioned in the high-level requirements, we should be able to understand where the greatest amount of sunlight and the smallest amount of sunlight can be found in a certain room. This can be calculated by UV/heat indexes (0-11+) provided during each time of each day, specific luminosity required by each plant (75-250 fc), and the degree at which shadows form on a surface. Data gathered should be recorded in a table, and graphs/figures can be made from the results.

Moving Chassis: We may run into errors determining when this chassis should move, how far it should move (and in what direction), and how it will move without running into obstructing objects. This can all be done by using code, so obviously, errors that arise in the program will be determined as we go.

The feasibility of our project depends on how well these two components are able to work together. The best way to understand if it's feasible or not is to take different examples from the internet, and explore their problem-solving steps to resolve the issues that they faced.

The most important aspect of our design is the light and moisture sensor circuit. The moisture contains a $10k\Omega$ and 100Ω resistor in parallel, and the light sensor contains double $4.7k\Omega$ in series. With 3.3v power, total power dissipation $P=V^2/R$ would be 0.111W, within the range of the power regulator.

3 Cost and Schedule

3.1 Cost Analysis

The total cost of our project is estimated to be around \$59.49 before shipping. After adding the 5% shipping cost, and 10% sales tax, our total amount will come near \$68.41. We can expect a salary of $\$40/\text{hour} \times 8\text{hrs}/\text{week} \times 10\text{ weeks} = \$3,200$ per team member. Therefore, since there are 3 members on our teams, the cost of labor would amount to \$9,600. Which would give us a final total of \$9668.41 for the entire scope of our project.

Description	Manufacturer	Quantity	Price	Link?
Chassis	Sparkfun	1	\$30	Link
Light Sensor	Sparkfun	1	\$5.95	Link
Moisture Sensor	Sparkfun	1	\$6.50	Link
Attiny85	DEVMO	1	\$15.99	Link
Speaker	Sparkfun	1	\$1.05	Link
Voltage regulator	Sparkfun	1	17.5	Link

Figure 3: Itemized List of the Components and Costs

3.2 Schedule

Week	Task	Person
February 20-27	Order parts for prototyping	Everyone
	Research plant & buy it	Maya
	Research chassis and figure out best way to implement it	Joshmita
	Start basic structure of circuit setup (digital)	Hongshang
February 27 - March 6	Finalize all components being used	Everyone
	Research the best way to acquire all data (light sensor, moisture sensor, movement), determine how the data will be presented, adapted, and used, and understand which data points are going to be taken	Everyone
March 6-13 (Spring Break)	Make sure all the physical components are ready to be put together and implemented	Everyone
March 13-20	Build Mock-up Prototype and start testing data	Everyone
March 20-27	Refine Prototype	Everyone
March 27 - April 3	Test Integrated System	Everyone

April 3-10	Finish all testing	Everyone
April 10-17 (Mock Demo - TA)	Make updates based on feedback	Everyone
April 17-24 (Final Demo - Instructor & TAs)	Make updates based on feedback	Everyone
April 24 - May 1 (Final Presentation - Instructor & TAs)	Be prepared :)	Everyone

Figure 4: Schedule for Project Progression

4 Ethics & Safety

Addressing Ethical Issues:

- In relation to the IEEE Code of Ethics
 - We must comply with ethical design and sustainable development practices and disclose factors that might endanger the public or environment. In relation to our project, since we plan to make an automated robot, we must ensure that it doesn't endanger the environment it is in, in any way.
 - We must mention the societal implications and capabilities of our robot, making sure the end-user is well-informed.
 - Since there is a group doing something similar in terms of measuring values relating to plants (they are measuring moisture values), we must make sure all our ideas are our own and original.
 - We must make sure all our claims and estimates are accurate and realistic, and accept honest feedback and criticism from our TA to make our project as precise as possible.
 - We must credit any sources, code, data, or information we use in the process of making our project.
 - We must use any equipment only if trained or experienced to use them.
 - We must work well with our team, and treat each other fairly and with respect.
- In relation to the ACM Code of Ethics
 - We must consider and provide evaluations of any associated risks of our project.
 - We must design and develop a safe and secure project.
 - We must make sure that the user is the main focus when making and producing this project. The end-user/public should be considered throughout the entire process, and should be made with them in mind.

Addressing Safety Issues:

- Make sure safety protocols are followed while in the lab soldering, using our PCB, and testing our sensors.
- Make sure to be aware when something can put ourselves at risk (when testing and building)
- Set boundaries, and make sure to work collaboratively so that not one person is more at risk for injury than another.
- Make sure that the end-product is safe and does not harm the user of the device, the environment, and the public. Users should be warned to be careful around the operating area despite the low speed of robot.

References

[1] “ACM Code of Ethics and Professional Conduct,” *ACM*. [Online]. Available:

<https://www.acm.org/code-of-ethics>. [Accessed: 09-Feb-2023].

[2] “IEEE Code of Ethics,” *IEEE*. [Online]. Available:

<https://www.ieee.org/about/corporate/governance/p7-8.html>. [Accessed: 09-Feb-2023].