Rachel Cherrey, Michael Gertz, Brandon Martinez, Christian Sieck

System 2 Final Report

Abstract: Our project was to create a system which could accurately state the best angle for a PV array in order to position it to get the most efficiency out of it as possible, and to report the Power Data of the PV's to ThingSpeak. We were able to successfully create the system which functioned when the latitude was inputted into it, and the power readings were sent to ThingSpeak, but were not always accurate.

Introduction to Grand Challenge: The grand challenge our team selected is making solar energy economical. Solar electricity is a growing multibillion industry as there is a need to switch to more sustainable energy sources. Fossil fuels are depleting and unable to match growing energy demands. Furthermore, these old energy sources largely contribute to pollution and carbon dioxide within the atmosphere [2]. In light of the climate crisis occurring, the need for a long-term sustainable energy source is more imperative than ever. Solar energy technology currently is able to convert sunlight into electricity with an efficiency of 10-20%, but new experiential multilayer cells have been able to reach 40% efficiency [2]. While technological advancements continue to be made with solar energy cells, a growing issue is making solar energy an economically viable option. Costs can be reduced through alternate materials for solar cells, or changes in how materials are being organized. Ultimately, great improvements in efficiency need to be made in order to make solar energy systems feasible from an economical standpoint.

The Problem Story: A homeowner in Wyoming, Liam, makes an average salary of $120k a year and wants to invest in solar energy. Liam resides in an agricultural sector of Wyoming, has a home that is 2100 square feet, and a property that is 2,430 acres. On the property, there are minimal trees or large obstructions that block the sun's light, so there is plenty of direct sun exposure for ground solar panels. There are no neighborhood restrictions on installing solar panels, as long as the necessary town work permit is requested. He lives with his partner and three dogs. In efforts to reduce carbon pollution and cut their electricity bill, they want to make the switch to renewable energy sources. Investing in solar panels is quite expensive, having an average installation cost between $10,258 to $12,537 [1] in Wyoming, so it's important to Liam to maximize sun exposure throughout the day to produce as much energy as possible. He has done some research in automatic sun-tracking solar panels, but concluded the return on investment was far too small, as they cost anywhere from $500-$1000 per panel[3], which will at least double the cost of installation and materials. Current automated systems available create a premium of 57% to 100% over the cost of fixed solar panels (depending on the type of tracking system) for only a maximum of 35% higher solar output. Liam would prefer a system that informs him how to manually shift his solar panels, to avoid paying an even more expensive installation fee, while also maximizing the solar panels sun exposure.

Problem Statement: Our group is tasked with creating a system that would be able to maximize sun exposure throughout the day, comparable to a single-axis solar tracker, by the user manually adjusting the solar panels' position. This would increase the amount of energy output by about 15% [9], as well as reduce installation and maintenance costs over time. New solar tracking systems are extremely expensive to install, and often this premium makes it unrealistic for single households.

System Overview: The purpose of the system is to provide a way for a homeowner with an adjustable solar panel to adjust their panel to get the most possible efficiency out of it. The system is an LCD display that reports what angle to adjust the panel to for maximum power output. The system also reports the power generated by the panel to the internet, so it can be accessed away from home.

System Shall List:

- The system shall enclose all necessary parts securely inside of it
- The system shall display the correct angle to be adjusted on LCD
- The system shall report the power readings to the internet

System Level Representations:

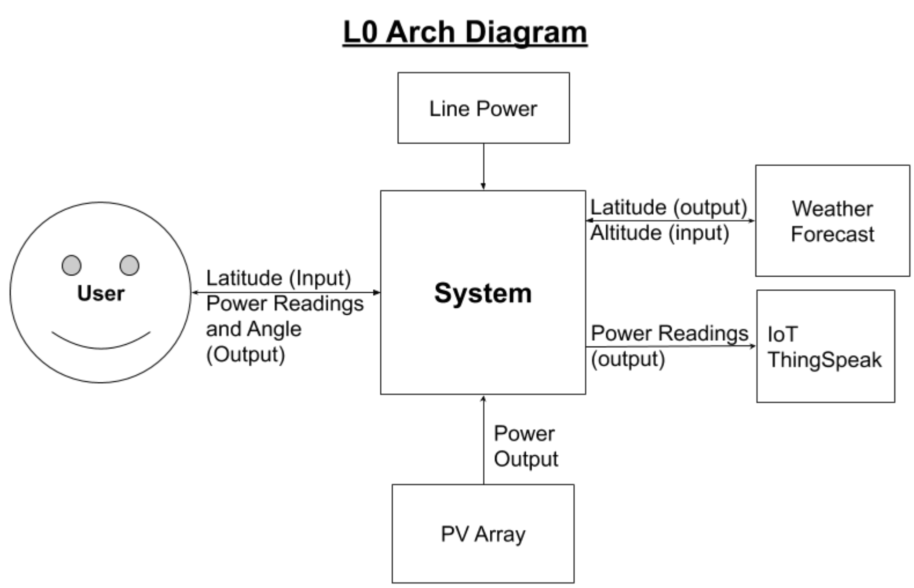


Figure 1: L0 Diagram of the system

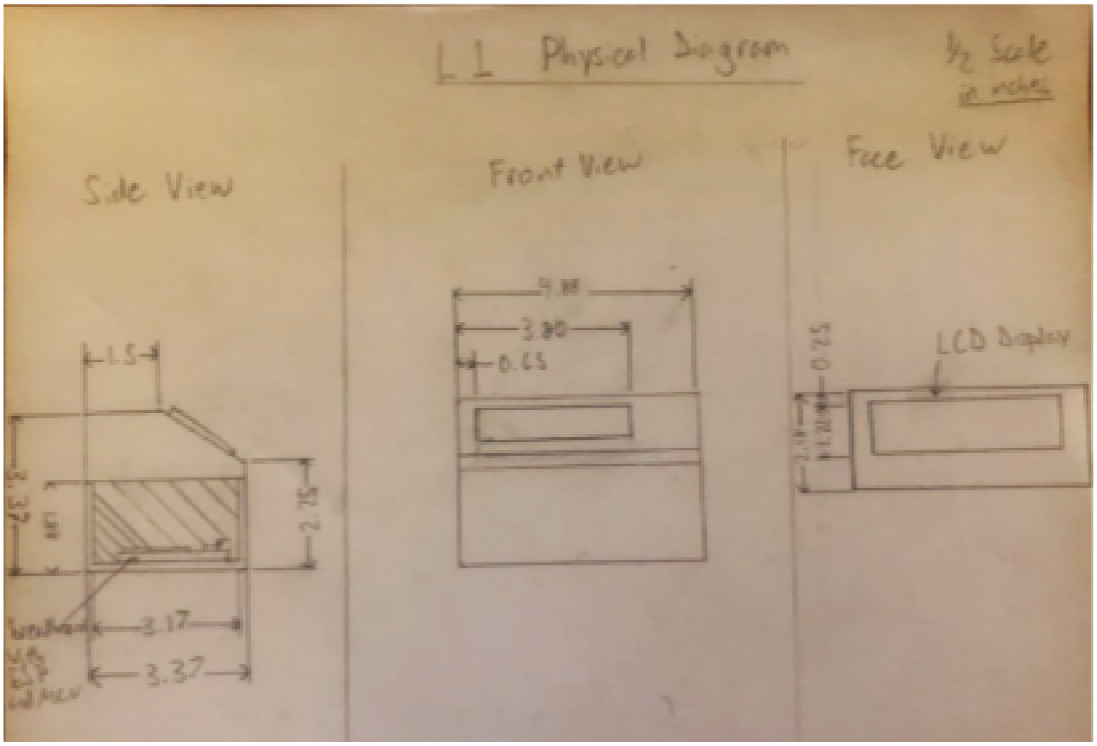


Figure 2: Physical L1 Diagram

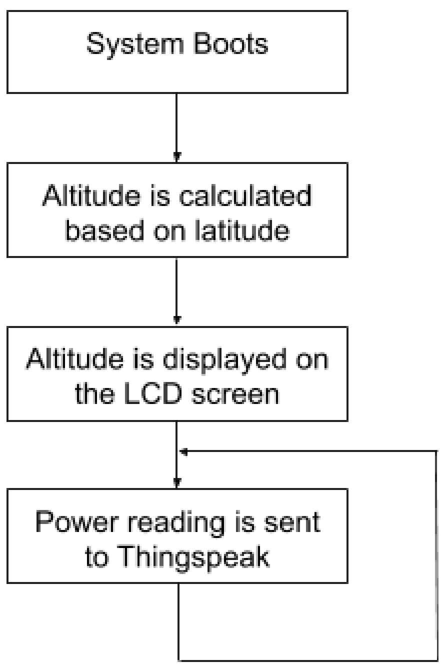


Figure 3: Behavioral L1 Diagram

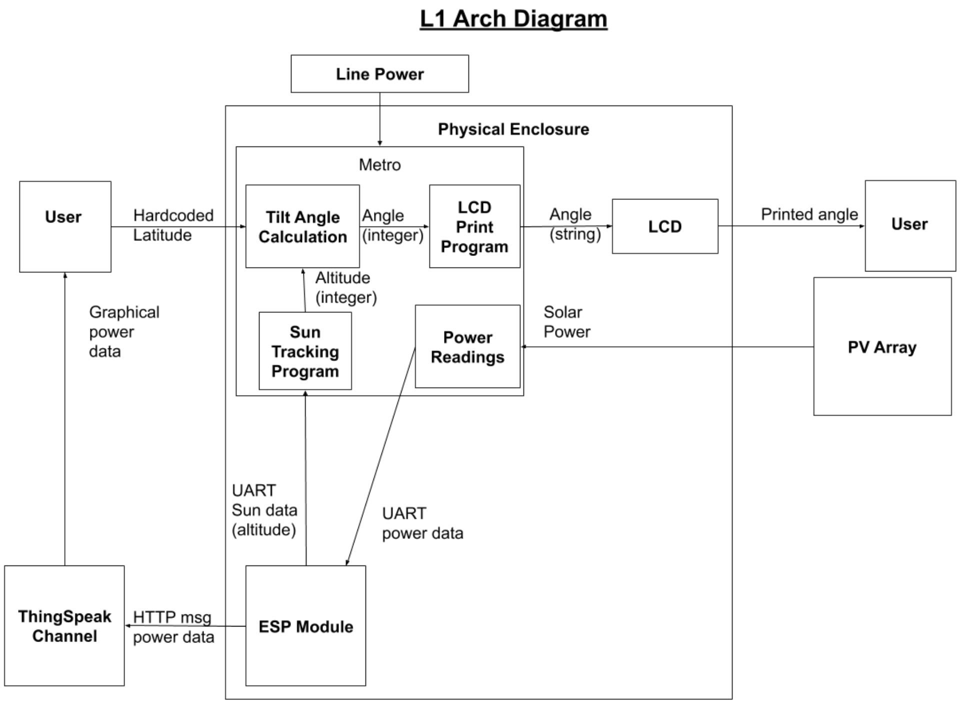


Figure 4: L1 Arch Diagram

Subsystems (and evolution):

Physical Subsystem - Christian:

My first diagram of this design was during our first meeting where we had decided the general problem statement and had a few ideas on how the system would solve that problem.

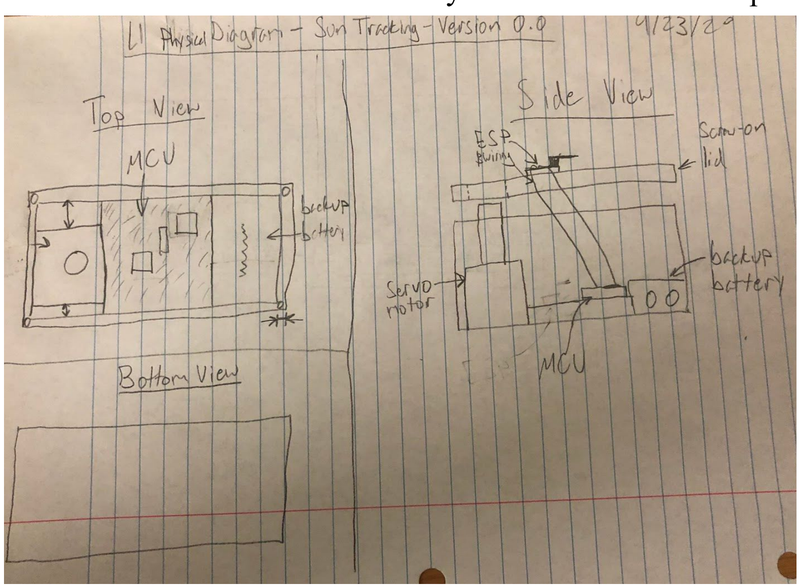


Figure 5: Initial Physical Diagram for the Project

After we had a clear intention for our project, though, this diagram was more or less scrapped, but the general design of the enclosure with a removable lid containing a micro-controller and an ESP was kept. The first real iteration of the enclosure was very similar to the final, but without the 3D printed face which held the LCD. The face of the enclosure was initially supposed to be laser cut, but due to the laser cutter not working, I had to switch to a 3D printed version, which was less sturdy and the area for the LCD pins had to be much larger. Figure 6 shows the enclosure before any of the components were added to it, and the two smaller holes in the face were put there for a button which was later removed from the system.

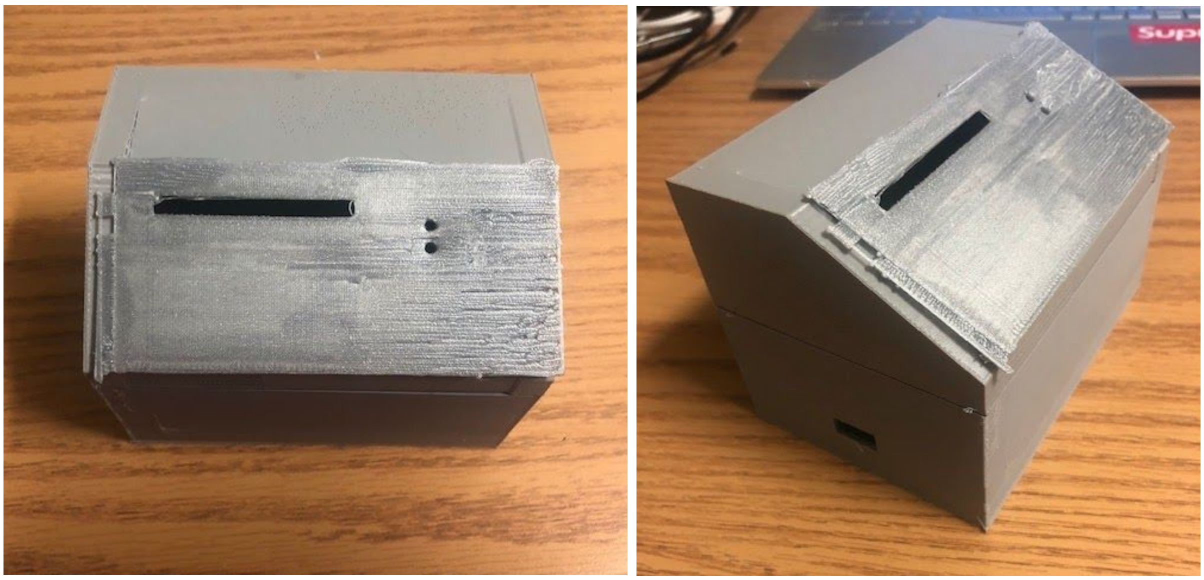


Figure 6: Pictures of the Physical enclosure without any electrical components

LCD Subsystem - Rachel:

My first iteration of the L1 architectural diagram for my subsystem exhibits how the Metro M0 Express and LCD are working together. My subsystem takes in the appropriate angle of adjustment and displays it onto the Liquid Crystal Display as a string. It's powered by plugging the Metro M0 Express to a power supply (either a computer or DC power outlet).

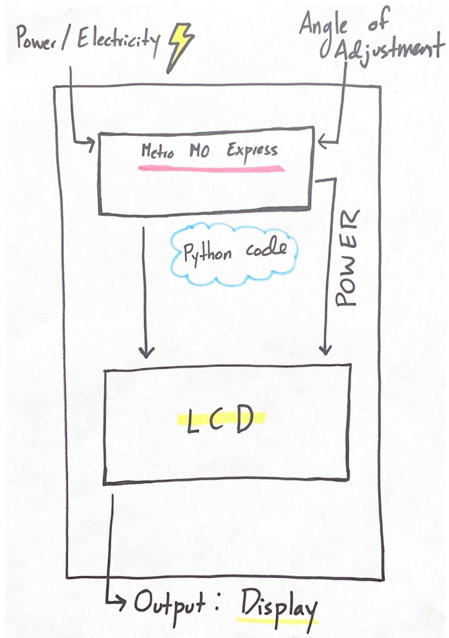


Figure 7: L1 Architectural Diagram for LCD Subsystem

The evolution of my subsystem largely centered on the communication protocol that I was using to communicate between the Metro M0 Express and the LCD. Originally, I was using a I2C protocol which was the only available interface compatible with the LCD (Newhaven Display NHD-0216K3Z-NSW-BBW-V3) I was initially using. The Character_LCD Library, which is necessary for the characterization on the display, was not compatible with the I2C protocol. It was for this reason I switched to a different LCD (Adafruit 181) which is capable of parallel interface and communication protocol. Once I made this change, I was able to run the necessary code and have strings displayed on the LCD, as pictured in Figure 8.

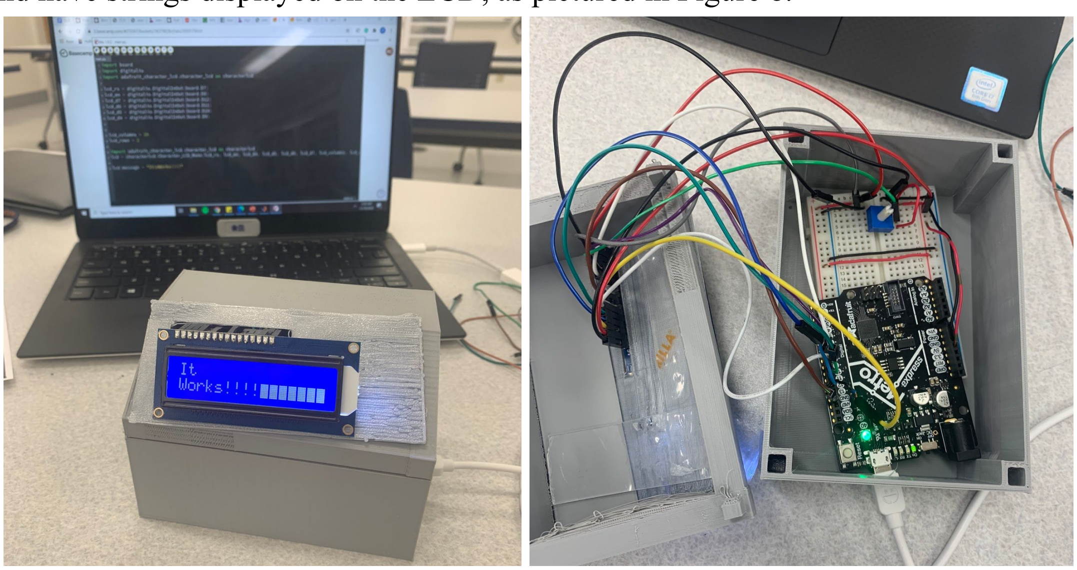


Figure 8: Final Functioning Implementation of LCD Subsystem

Altitude Data Subsystem - Michael:
My subsystem is designed to take input parameters from the user and use those data points to calculate and output the maximum angle (altitude) the sun will be in a given day. This angle is shown in the figure below.

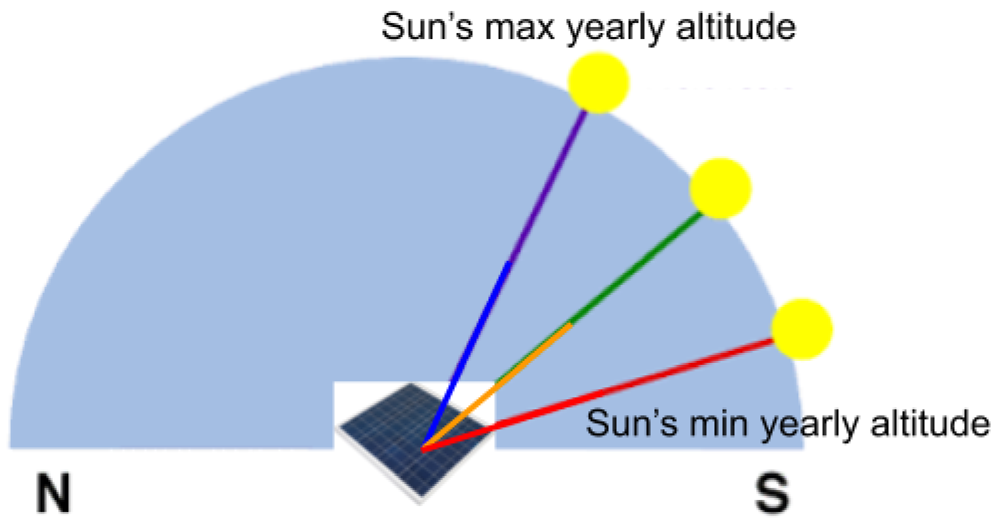


Figure 9: Visualization of Solar altitude throughout the year

My original idea was to use a website called suncalc.org to find the altitude of the sun, using the given input parameters of the users latitude and longitude from a google maps search, as well as the current date. Using these parameters, as well as a standard time of 12:10, the average time of the altitude being at its max throughout the year, a very accurate altitude could be captured from this website and output by my subsystem. After most of coding work was finished for this subsystem, I realized that the microcontrollers we were using in this iteration of the system only utilize a specialized variation of python that doesn't include the specific python libraries I was using to access information from the website. After realizing this, I had to rethink my plan on how to get the altitude data as an output depending on various inputs. I decided to do this by manually creating a table of data that uses numerical estimation to associate latitudes with altitudes. I ended up doing this with only a small amount of time frames so the system will really only be accurate for a few days of the year. My plan for the next iteration was to be able to access the suncal.org website without using complex libraries. This would have allowed the subsystem to be accurate throughout the entire year as opposed to to just a few parts of it.

Power Data Subsystem - Brandon:
The Power Reading system focuses on the accuracy of the amount of power being consumed by the load whether it is a lamp, TV, refrigerator etc. My design didn't measure those high impedances or power levels. However, I tried to model the bigger system into a smaller system where I can use a shunt resistor that can tolerate small voltage levels between 3V and 7V. For measuring the current, there were different options to consider like the implementation of the

clamp ammeter. However, such devices have a high cost which opposes our goal for affordable solar energy. Using the shunt resistor circuit, the shunt resistor had a large resistance (5,600 Ohms) for my first test and it evolved into having a smaller resistance (3,300 Ohms) for my second test. The load has a large proportion resistance compared to the shunt resistor. Making it larger than 3,300 Ohms would be ideal. I used the AD2 as a power source to replace the solar panel to test the circuit and code. Then, I changed the source to the microcontroller’s digital output. I intended to use the solar panel, but couldn’t be achieved due to logistical limitations.

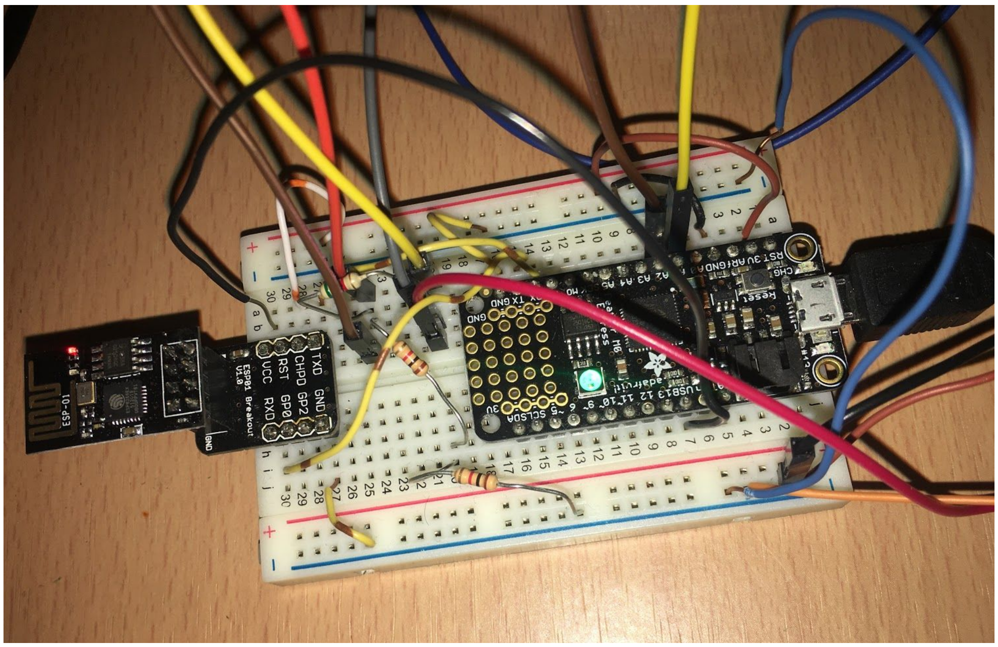
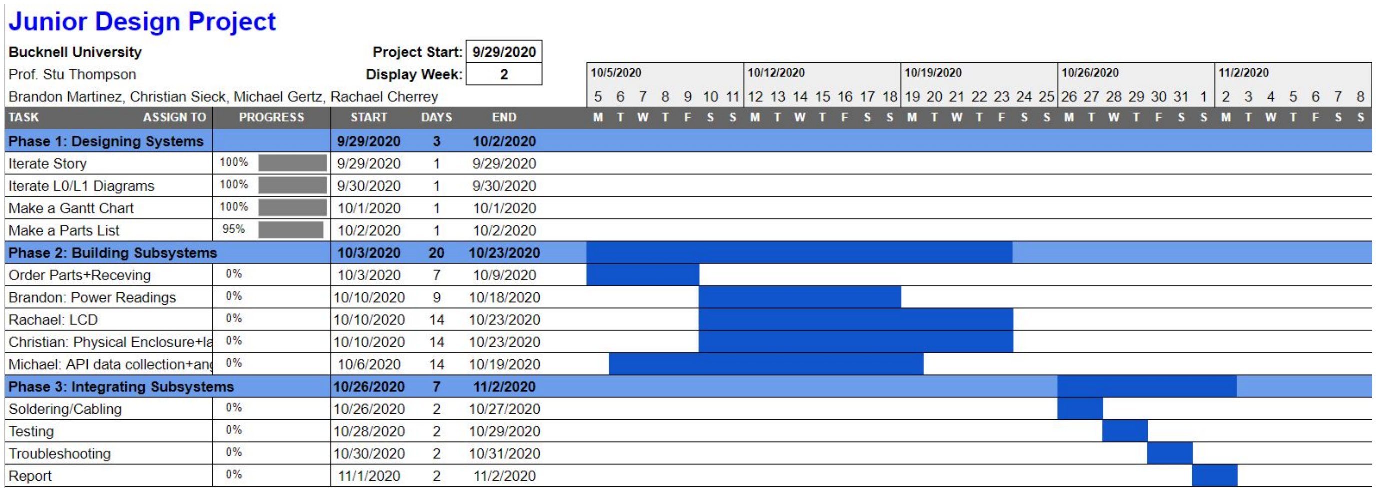


Figure 10: First Power Reader circuit

Logistics Model and Reflection:
As expected from the beginning, we ran behind schedule. The individual subsystems were not fully completed until about a month after the October 23rd deadline, and we didn't have the system fully integrated until the end of November. We planned for the testing, troubleshooting to take about half the time that they did, especially the troubleshooting, and the report itself took about 3 days, where we had originally planned for 1. Of course, our team was inexperienced with long design tasks and some of the areas we were designing in, and with the parts taking a long time to ship and not being the correct parts for the job, our project took a lot longer than expected. We should have allotted more time during subsystem creation for parts ordering as well. If we were to do it again, we could have expanded the time taken for each of the subsystems as well as the Phase 3 things. Dividing up the work was an effective strategy, but it would have been nicer to have been able to work with a team member more often so that we could keep each other accountable for getting parts of the project done faster. Overall, I think it was a great learning experience, and next time, we will be more realistic about the deadlines and try to make them more attainable so that we can stick to them.



Junior Design Project

Bucknell University — **Project Start:** 9/29/2020
Prof. Stu Thompson — **Display Week:** 2
Brandon Martinez, Christian Sieck, Michael Gertz, Rachael Cherrey

TASK	ASSIGN TO	PROGRESS	START	DAYS	END
Phase 1: Designing Systems			**9/29/2020**	**3**	**10/2/2020**
Iterate Story		100%	9/29/2020	1	9/29/2020
Iterate L0/L1 Diagrams		100%	9/30/2020	1	9/30/2020
Make a Gantt Chart		100%	10/1/2020	1	10/1/2020
Make a Parts List		95%	10/2/2020	1	10/2/2020
Phase 2: Building Subsystems			**10/3/2020**	**20**	**10/23/2020**
Order Parts+Receving		0%	10/3/2020	7	10/9/2020
Brandon: Power Readings		0%	10/10/2020	9	10/18/2020
Rachael: LCD		0%	10/10/2020	14	10/23/2020
Christian: Physical Enclosure+la		0%	10/10/2020	14	10/23/2020
Michael: API data collection+an		0%	10/6/2020	14	10/19/2020
Phase 3: Integrating Subsystems			**10/26/2020**	**7**	**11/2/2020**
Soldering/Cabling		0%	10/26/2020	2	10/27/2020
Testing		0%	10/28/2020	2	10/29/2020
Troubleshooting		0%	10/30/2020	2	10/31/2020
Report		0%	11/1/2020	2	11/2/2020

Figure 11: Gantt Chart

System Implementation:

Click [here]() for a video demo of our system working. In the recording, the altitude inputted is 40.3° which results in the solar panel needing to be adjusted by 49° above the horizontal, as displayed on the LCD.

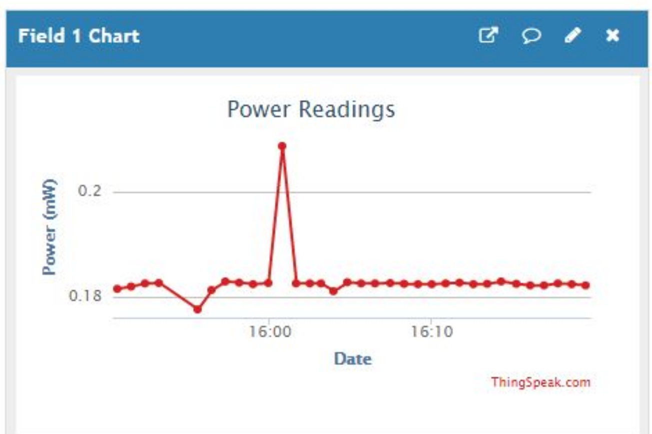


Figure 12: Power Readings received by Thingspeak

The power readings for the amount of power produced by the solar panels (simulated) are uploaded on Thingspeak shown in Figure 5. The microcontroller receives a digital output of 3.3V in an analog input and measures the current passing through a 3.3kΩ load and a 3.3kΩ shunt resistor, connecting to another analog input. Power is calculated by multiplying the voltage and current.

System Evaluation:

Test Setup:

Test 1 - Physical Test: This test will determine if the system is able to function in an indoor environment. For this test, the system should be plugged in. The system should be moved around and jostled, but not dropped. A passing grade will be given if the system continues to function after it has been jostled.

Test 2 - Behavioral Test: This test will determine if the system is able to report the correct angle of the sun to maximize power output. The system should be turned on and reporting the angle. Then, the operator should check if the value reported on the LCD matches the value given online. A passing grade will be given if the LCD displays the correct value and the value is within 1 degree of the online data.

Test 3 - Behavioral Test 2: This test will determine if the system is able to report values to an online location. The system should be turned on and reporting some angle and direction on the LCD display. The operator should then open Thingspeak on their computer and check the value given. A passing grade will be given if Thingspeak is displaying the power output value which was most recently updated to the system. Also, the readings should be within a certain range to its actual value. The readings should be about 10% accurate to be considered reliable.

Test 4 - Electronics Test: This test will determine if the system is able to function after having its internal components poked or prodded at, or if certain objects could be able to fit inside the physical enclosure and cause the system to stop working. For this test, we decided to use a No. 2 pencil, as that is a common object that could possibly cause damage to the system. A passing grade will be given if the user is unable to get a pencil into the enclosure and make contact with the Metro board or ESP-01.

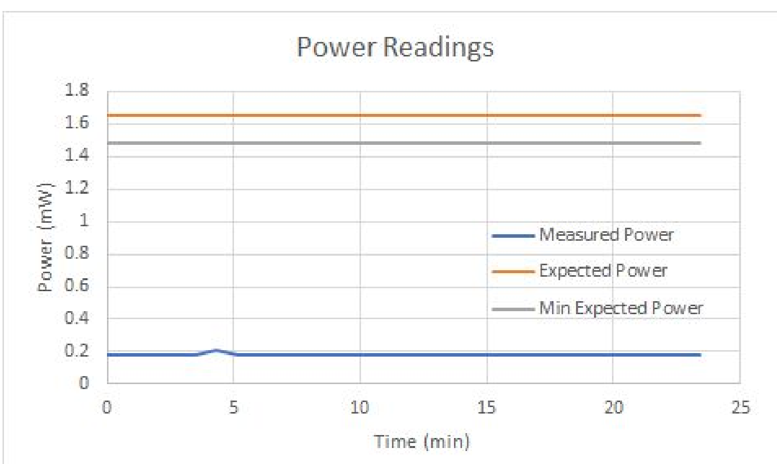


Figure 13: Measured and Expected Power Reading

The power readings were measured by monitoring the Thingspeak channel. The expected value was calculated by solving the power dissipated by two 3.3kΩ resistors with an input voltage of 3.3 V. The minimum value derives from the context to have 10% accuracy to allow the user to monitor previous power consumption for budgeting. 10% is enough where the error cannot cause the user to misinterpret the data by more than $50 off each month. Also, if the readings were to be used to monitor the power consumption of the user's house. The data should be updated each day on Thingspeak. The testing procedure endured over several minutes.

Table I: System Evaluation Testing

	Pass/Fail	**Explanation**
Test 1	Pass	The system was fully functional after being shaken and moved around
Test 2	Pass	The system accurately matches angle to the data from online for specific times of year.
Test 3	Fail	88.8% accuracy of Power Readings. The error can cause the user to misread savings up to $400
Test 4	Test not Completed	

Judgement and Conclusion:
Based on the system evaluations conducted, and successfully passing two of the three of the tests performed, it is fair to conclude that we may continue with our current approach as soon as we can improve upon the power readings. The strength of our system is largely related to its functionality. Our system is able to use any inputted altitude to determine what angle the solar panel needs to be adjusted to, which is outputted onto the LCD. Further, power readings from the "solar panel" (we used the digital output pins on the Metro M0 Express) were reported on Thingspeak using a WiFi module (ESP-01). The readings are not as accurate as we want them to be. Given the context that the user needs the data to save energy and money, the readings would fail to provide that service. If the accuracy of the altitude data can be improved as days change, we believe this could change, however the current system does not succeed.

Weaknesses of our system are:

- Accuracy of Power Readings
- Physical Components not fitting snugly into the enclosure, especially the LCD because of the flimsy face part

- Accurate angles of solar panel based more parts of the year.

We can improve by:

- Making the components out of a stronger material, or by making the whole enclosure larger so that components have more space, also integrating everything onto a single chip could help get rid of a lot of the clutter
- Changing the way we get the current altitude to incorporate more input data for a more accurate altitude reading
- Using more accurate, existing systems to measure power output from the PV array

Interviews:

<u>Context Interview: Marc Del Vecchio</u>

- Impractical to expect homeowners to adjust the solar panel daily
- Likely errors in adjustment by the user/homeowner
- Locations of solar panels problematic
  - Roof - extremely dangerous to be trying to adjust the solar panels
  - Ground - any efficiency gained by adjustment will be negated by the shading on the ground and lack of sun exposure
- There is already technology available to homeowners, such as the Tesla app, that will break down the energy produced throughout the day by time

<u>Tech Interview: Prof. Kabalan</u>

- Could change scope since panels on the roof wouldn't be easily adjusted
- Might work better in commercial/agricultural sectors
- Might also be more efficient to have it automated
- In the future, moving it to a totally digital system would be useful and do away with the physical aspect
- Research could be done to check if the system is necessary in the residential sector

Individual Reflection (Brandon)

1) What are you most proud of and why?
   I am proud to accomplish a course of design where I confronted many obstacles like being remote. I had to get out of my comfort zone to learn new things in different ways.
2) How do you feel you contributed the most to the team and project?
   I contributed to the team by doing my part of the system. I could have played a better role in the team by being in Bucknell. However, I could only do my piece and conduct interviews remotely. Information was also a key deliverable I gave to my team.
3) What are your biggest takeaways from the experience? (up to 3)
   I have a general understanding of design.
   Teamwork is fundamental to a project's success.
   Communication is easier when you are in a meeting rather than a chat room.

4) How did the experience change your conception of engineering design?
   This experience gave me a better understanding of design. Picking a choice to solve a problem in a particular way embarks a path of new knowledge. Asking questions and finding answers takes you to Atlantis.

Individual Reflection (Michael)

1) What are you most proud of and why?
   I am very proud that my team was able to design a system ourselves and implement all of our parts that were built separately into one cohesive system. It was really cool to be able to see a whole semester of work come together and have a final product we can point to and say “We made that”.

2) How do you feel you contributed the most to the team and project?
   I feel I contributed the most in a research and design idea capacity. I feel I helped generate ideas on what system we could build or potential ways to solve an issue when they came up, whether they were in my own subsystem, someone else's, or a conceptual problem from the whole group.

3) What are your biggest takeaways?
   Start our work on the next step early and ask lots of questions.
   You can't communicate too much so share everything you can that could have any impact on your other team members.

4) How did the experience change your conception of engineering design?
   Before this class I never really had a strong conception of what it meant for an engineer to design something. This class taught me that there is a lot more planning involved in the engineering process than I had previously thought. The Design process requires just as much time, if not more, planning and researching before you even start to physically work on your system. Also, I learned that you can make a product and do as much research as you want beforehand, but sometimes you don't know if and how a product will be used until you start to have a prototype that you can put in peoples hands and have them interact with it.

Individual Reflection (Christian)

1) What are you most proud of and why?
   I was most proud of how the system was able to come together at the end very well and complete the task we had in mind. It was very satisfying to be able to use the 3D printer successfully and somewhat effectively.

2) How do you feel you contributed the most to the team and project?
I think I contributed to the team in an organizational role, especially when we were beginning the project and had to keep things developing and moving as we were thinking about possible problems to tackle, and how to divide up the work.

3) What are your biggest takeaways from the experience?
    - Take your time on the design process and take as much time as you can before building
    - Keep in communication with your team and talk about everything
    - Set deadlines and try to not get behind, it's hard to catch up

4) How did the experience change your conception of engineering design?
At first, I knew we had to to design and plan our project, but I never really knew how important it was until we got towards the end. I found myself going back and redesigning everything midway through. Setting up a system shall list and the exact connections you'll have with your teammates' subsystems is very important, because a few missed details could cause huge problems later on.

Individual Reflection (Rachel)

1. I'm most proud of my final implementation of the LCD subsystem. Throughout the process, I encountered a lot of obstacles that required me to change the hardware I was working with and understand how the Metro M0 Express was communicating with the LCD. My subsystem required me to understand its physical wiring implementation, the libraries it's using, and the code that connects it all together. I'm proud I was able to work through the challenges I encountered and get my subsystem fully functional! I'm also really proud of how our team was able to integrate our subsystems together and have the system working as intended - that was really exciting to see.

2. I feel that I contributed the most to the team and project through my own communication and prompting others to do the same. I realized how important it is to be on the same page as my teammates and made an effort to convey the status of my subsystem, issues I was facing, and organize meeting times for the group to discuss progress & what we need to accomplish moving forward. This made a big difference in our group's dynamic as we were able to understand one another's progress and how that impacted our own subsystem, as well as the overall project/system. When individuals are each working on a subsection of an overall system, it's imperative to consistently update one another to prevent issues from arising in the integration process.

3. My biggest takeaways from this experience are the engineering design process, how to approach developing a subsystem, and the importance of communication within your

team. I learned a lot how to approach creating a system from the system level representation with diagrams, to going through numerous iterations of my subsystem.

4. This experience had a tremendous impact on my conception of the engineering design process. Previous to this experience, I hadn't dealt with system level abstractions. I now understand moving through the layers of abstraction, and the differences between the architectural, behavioral, and physical diagrams. These helped me understand and articulate different aspects of the system. Going through numerous iterations of my subsystem taught me about the importance of understanding the details of a system before beginning. I realized that you should fully flesh out all aspects of a design before beginning the fabrication process.

References

[1] “Make Solar Energy Economical,” *Grand Challenges - Make Solar Energy Economical*. [Online]. Available: http://www.engineeringchallenges.org/challenges/solar.aspx. [Accessed: 03-Dec-2020].

[2] “Solar Panel Cost Wyoming: Local prices & online estimator,” SolarReviews. [Online]. Available: https://www.solarreviews.com/solar-panel-cost/wyoming. [Accessed: 03-Dec-2020].

[3] Thesolarnerd.com. 2020. *How Many Solar Panels Are Needed To Run A 3,000 Square Foot House?*. [online] Available at: <https://www.thesolarnerd.com/blog/how-many-solar-panels-3000-sqft-house/> [Accessed 3 December 2020].

[4] 2020. [online] Available at: <https://www.alternative-energy-tutorials.com/solar-power/pv-panel.html#:~:text=An%20individual%20photovoltaic%20cell%20produces,from%20dull%20to%20bright%20sunlight.> [Accessed 3 December 2020].

[5] L. Ada, “Character LCDs,” *Adafruit Learning System*, 29-Jul-2012. [Online]. Available: https://learn.adafruit.com/character-lcds/python-circuitpython. [Accessed: 06-Dec-2020].

[6] “Character_LCD Library,” *CircuitPython Documents*, 2017. [Online]. Available: https://circuitpython.readthedocs.io/projects/charlcd/en/latest/. [Accessed: 06-Dec-2020].

[7] “LCM Module,” *TC1602A-09T Datasheet*, 06-Apr-2009. [Online]. Available: https://cdn-shop.adafruit.com/product-files/181/p181.pdf. [Accessed: 06-Dec-2020].

[8] L. Ada, “Adafruit Learning System,” *Adafruit Metro M0 Express - Designed for CircuitPython*, 26-Oct-2020. [Online]. Available: https://cdn-learn.adafruit.com/downloads/pdf/adafruit-metro-m0-express-designed-for-circuitpython.pdf. [Accessed: 06-Dec-2020].

[9] Gogreensolar.com. 2020. *Does Tilting Solar Panels Make Sense?* [online] Available at: https://blog.gogreensolar.com/2018/10/does-tilting-solar-panels-make-sense.html

[10] L. Ada, “Adding a WiFi Co-Processor to CircuitPython,” *Adafruit Learning System*. [Online]. Available:

https://learn.adafruit.com/adding-a-wifi-co-processor-to-circuitpython-esp8266-esp32/program-esp8266-via-circuitpython. [Accessed: 07-Dec-2020].

[11] L. Teschler, “Measuring current with shunt resistors,” *Power Electronic Tips*. [Online]. Available: https://www.powerelectronictips.com/measuring-current-shunt-resistors/. [Accessed: 07-Dec-2020].

Appendix

Physical Design - Christian:
[see Solid Works files attached]

LCD Subsystem - Rachel:
Schematic:

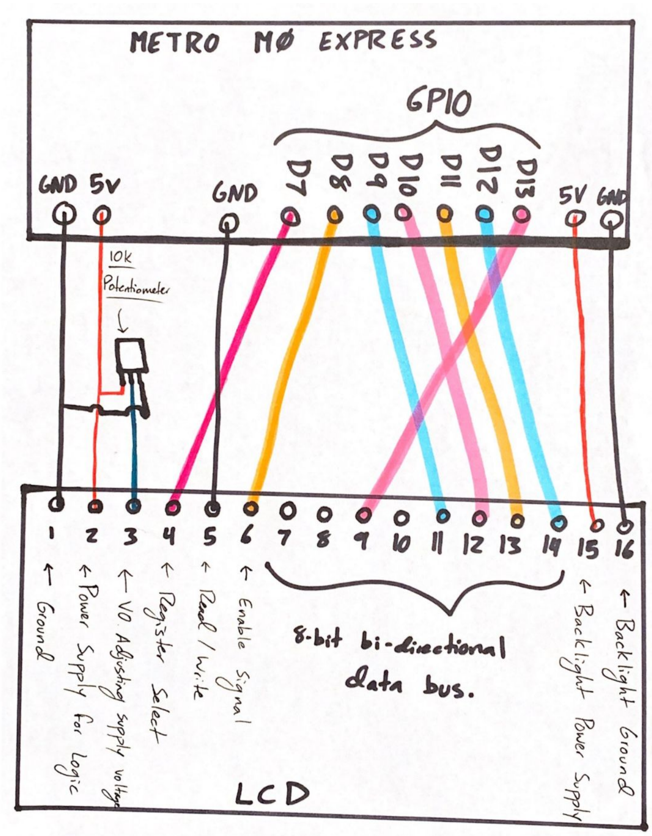


Figure 14: LCD & Metro M0 Express Connections Schematic

Code:

```
import board
import digitalio
import adafruit_character_lcd.character_lcd as characterlcd
```

```
lcd_rs = digitalio.DigitalInOut(board.D7)
lcd_en = digitalio.DigitalInOut(board.D8)
lcd_d7 = digitalio.DigitalInOut(board.D12)
lcd_d6 = digitalio.DigitalInOut(board.D11)
lcd_d5 = digitalio.DigitalInOut(board.D10)
lcd_d4 = digitalio.DigitalInOut(board.D9)


lcd_columns = 16
lcd_rows = 2

import adafruit_character_lcd.character_lcd as characterlcd
lcd = characterlcd.Character_LCD_Mono(lcd_rs, lcd_en, lcd_d4, lcd_d5,
lcd_d6, lcd_d7, lcd_columns, lcd_rows)

lcd.message = "It\nWorks!!!!"
```

References:
Wiring: [5]
Code: [6]
LCD Datasheet: [7]
Metro M0 Express Datasheet: [8]

Power Readings subsystem - Brandon:
Schematic:

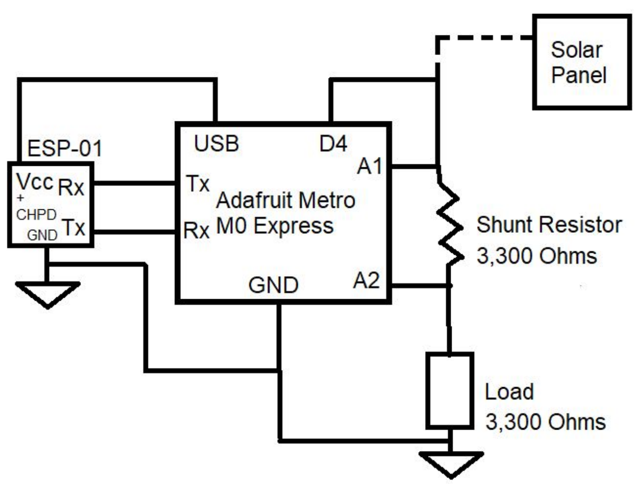


Figure 15: Metro M0 Express Power Reader Schematic (needs PV)

Code:
```
import time
import board
```

```
import busio
import analogio

API_KEY = "U5XA7UEOCYYLE12W"  # ThingsSpeak channel API
SSID = "MySpectrumWiFi88-2G" #wifi setup
PASSWORD = "livelypoodle574"

uart = busio.UART(board.TX, board.RX, baudrate=115200, bits=8,
                  parity=None, stop=1, timeout=1)

analogsensorpin = board.A1  # Change this to your specific pin
analogsensor = analogio.AnalogIn(analogsensorpin)

analogsensorpin2 = board.A2  # Change this to your specific pin
analogsensor2 = analogio.AnalogIn(analogsensorpin2)


print("Resetting the ESP01")
uart.write(b'AT+RST\r\n')
time.sleep(5)
uart.write(b'AT+CWMODE=1\r\n')
time.sleep(5)
uart.write(b'AT+CWJAP="'+bytearray(str(SSID))+b'","'+bytearray(str(PASSWOR
D))+b'"\r\n')
time.sleep(5)

while True:
    #data = uart.read(32)

    #if data is not None:
    #    data_string = ''.join([chr(b) for b in data])
    #    print(data_string,end="")
    #    time.sleep(1)

    Vout = analogsensor.value / 65535 #Volts
    Vsec = analogsensor2.value / 65535

    Cout = (Vout-Vsec)/(3300)
    Pout = Cout*Vout
    watts = Pout*1000 #mW
    print("Sensor Value = "+str(watts)+" mW")
    time.sleep(1)

    uart.write(b"AT+CIPMODE=0\r\n")
    time.sleep(3)
    uart.write(b'AT+CIPSTART="TCP","api.thingspeak.com",80\r\n')
    time.sleep(5)
```

```
    length = len('GET /update?api_key='+API_KEY+'&field1='+str(watts)+
                 ' HTTP/1.1')+29

    uart.write(b'AT+CIPSEND=' + bytearray(str(length)) + b'\r\n')
    time.sleep(2)

    uart.write(b'GET /update?api_key='+bytearray(API_KEY)+b'&field1='+
               bytearray(str(watts))+
               b' HTTP/1.1\r\n')
    time.sleep(10)
    uart.write(b'Host:api.thingspeak.com\r\n\r\n')
    time.sleep(10)
    uart.write(b'AT+CIPCLOSE\r\n')
    time.sleep(20)
```

References:

Current measurement: [11]

Metro M0 Express Datasheet: [8]

ESP-01 Datasheet and Code: [10]

Altitude API - Michael

Code:

```
def setTables():
    lattitudes_list = [[a, a - 2] for a in range(85, 0, -2)]
    return lattitudes_list

def getAltitude(lat):
    lattitudes = setTables()
    b = 5
    alt_list = []
    for j in range(5, 90, 2):
        alt_list.append([j, lattitudes[b - 5]])
        b += 1
    for i in range(len(alt_list)):
        if alt_list[i][1][1] < lat < alt_list[i][1][0]:
            return alt_list[i][0]
        elif alt_list[i][1][0] == lat:
            return alt_list[i][0]
        elif lat == 0:
            return 90
```