IMPROVING SOLAR MODULES DURING WILDFIRES

Final Report

John Olson (Lead), and Anthony Sandoval

Faculty Advisor: Dr. Nagy Bengiamin

May 10, 2021

EXECUTIVE SUMMARY (John)

In this paper, a methodology is presented to improve and optimize solar modules with an automated system, which includes a battery and an air compressor. These two components of the system improve and optimize solar modules by backing up the decrease in generation and cleaning the surface of the solar panel. Background information on the various parts of the project is contained in the first portion of this paper, as well as an introduction to the design. Following that, a narrative and analysis are given on the design procedure and data. Included in this narrative are alternative solutions, schematics, pictures of hardware used, an economic and safety analysis, and results received to date. Various references, standards, and appendices can be found at the end of the paper.

TABLE OF CONTENTS

INTRODUCTION (John)

The purpose of this project is to improve and optimize solar modules with an automated system. The improvements are aimed at introducing greater resilience for solar generation during wildfires. The smoke layer that develops in the sky during wildfires can cover large areas over several states, decreasing the amount of sunlight that gets to the solar panels in that area. In addition to this, ash drifts down and covers up solar panels, further decreasing their output [4,5]. These decreases can cause a need for rolling blackouts, as much of our power generation in California comes from solar panels [2,6].

One model for increasing the resilience of power generation has been to use a combination of photovoltaic (PV) arrays with battery systems, thus reducing interruptions like an imbalance of supply and demand causing the need for a power shutoff [7, 8]. In our system, this is accomplished through a battery whose output is controlled by an Arduino sensor and a MOSFET switch. The Arduino will be keeping track of the maximum power point tracking from the solar panel by observing the duty cycle and making the necessary changes for the most efficient output. In addition to supplementing the PV generation, another effective way to add resilience to the system has been to ensure that the solar module is automatically cleaned. This can be done in a variety of ways, but it has been demonstrated that removing dust and ash from solar modules helps to keep the generation up to expectations [9]. For our system, this is done through the use of an optical sensor in combination with an air compressor, both controlled by an Arduino through its analog inputs and a MOSFET switch. The optical sensor will constantly check if there is an interruption in its reflection which means that there is ash, which will be substituted with diatomaceous earth, on the solar panel. It will then send the data which will notify it to turn on the air compressor to get rid of the ash.

A project like this could be valuable to a utility company that wishes to implement a solution to the decrease in photovoltaic generation caused by wildfires. From a societal standpoint, a solar panel setup that works well in the harshest of conditions can be relied upon and trusted, especially when customers can see the effects of good design such as a lack of rolling blackouts, as well as consistency in communication. The final product of this project may serve to improve upon the reliability of solar panels, as well as identify the measurable effects of smoke and ash on their output.

BACKGROUND

(John)

How Solar Panels Work and the Problem of Relying on Them

Solar cells are pn-junction devices with no directly applied voltage at the junction. Solar energy is converted into electrical energy when light hits the space-charge region and electrons and holes are generated. These are then separated quickly and the electric field sweeps them out of the space-charge region, generating a photocurrent. This photocurrent produces a voltage across the load, which means power has been produced. The effective range of wavelengths of light is from about 400 nm to 800 nm, which includes most white LEDs like ours. Normally, solar cells are fabricated from silicon, but III-V compound semiconductors like GaAs can also be used [11].

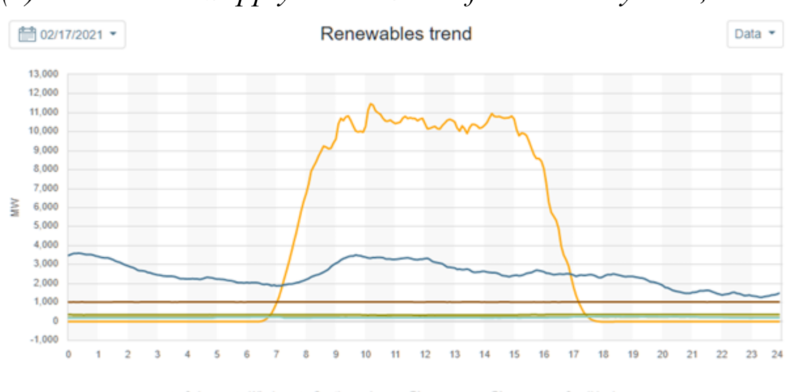
The solar monitoring company “Solar Analytics” monitors the energy consumption and solar PV performance of 35,000 sites across Australia, and found that PV output in Sydney and Canberra dropped by 15 - 45% on heavy smoke days during the December 2019 wildfire season. Analysis of PV sites in Sydney on December 21, 2019, which was very smoky, but a cloud-free day, discovered a 27% drop in generation compared to a clear day at the same time of year in 2018. Different semiconductor usage in various types of solar panels respond differently to the haze from smoke, with the standard silicon version showing the best results [3].

Here in California, the decrease in solar generation is the main cause of the need for rolling blackouts, and will increase in frequency as California nears its renewable power goals. Steve Berberich, CEO of the California ISO, rebuked California state regulators at the California Public Utilities Commission for not ensuring adequate generation in a statement from August 18: “The situation we are in could have been avoided. For many years we have pointed out to the procurement authorizing authorities that there was inadequate power available.” In September, Former Energy Secretary Ernest Moniz stated that backing up solar and wind with natural gas is unavoidable with our current technology, describing a “shortage of capacity”. With California’s demand for less natural gas generation and more solar generation, the state relies more heavily on the ability of the solar panels to generate enough power consistently, which does not happen when the state is covered in smoke [1].

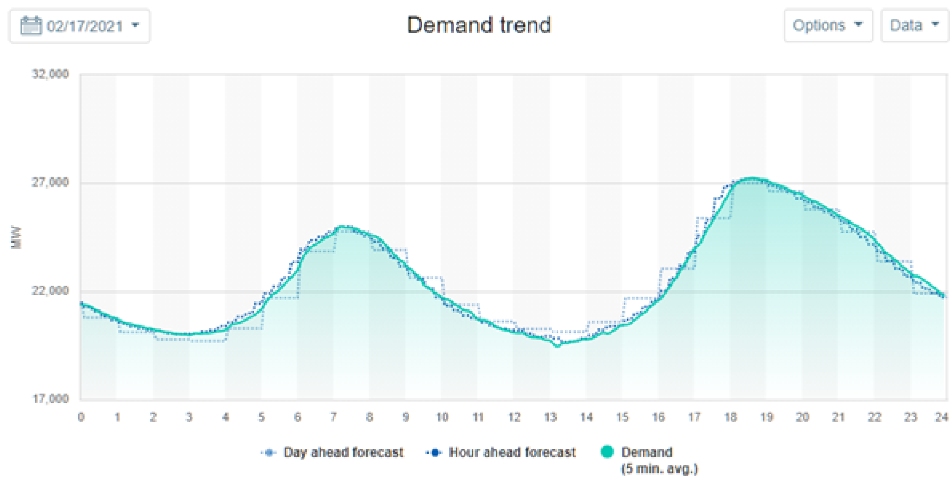
To be more specific, the problem with simply relying on solar generation, to a degree where it cannot be supplemented by something like a natural gas power plant, is that solar generation hits its peak at a different time of day than the peak electricity usage. Solar generation is typically at its highest at midday to early afternoon. The curve for electricity usage looks almost the opposite, declining at midday and increasing as afternoon turns to evening. To illustrate this, Figure 1 shows the renewables supply (a) and demand (b) curves for February 17th, 2021, from CAISO.

Figure 1

(a) Renewables Supply Trend Curve for February 17th, 2021.



(b) Demand Trend Curve for February 17th, 2021.



As millions of acres of land burned across California in a few short weeks of September 2020, solar power generation declined by thirty percent compared to July earlier that year [5, 10]. According to data from the time, presented by Stephen York of the U.S. Energy Information Administration, smoke concentration levels reached peaks of 659 micrograms per cubic meter, the highest on record. At the same time, PV generation declined to its lowest all summer, only generating fifty gigawatt hours on September 11th, 2020. Data from summer 2020 is shown in Figure 2. Solar generation was 13.4% lower in September 2020 than it was a year prior, despite an at least 5.3% increase in capacity. While it is true that most of the solar generation takes place in Southern California, while the fires were mostly concentrated in Northern and Central California. Coastal winds tend to push most wildfire smoke south, which is why there was such a dramatic decrease in generation there despite the lack of local fires.

Buck-Boost Converters

In order to utilize the power generated by a solar panel, its output must be converted to a usable form. While a DC-AC inverter is beyond the scope of this project, it would be necessary for the solar panel to power the 60 Hz 120V-AC that is used by most household devices. In this project, the solar panel is used to power a 12V-DC air compressor, so it is necessary to convert the output to that 12V, which requires a Buck-Boost converter. A Buck-Boost converter is a DC to DC converter where the output voltage is constant while the input voltage can be higher or lower than the output voltage, and the output power is equal to the input power because of low conversion energy needed. It combines the principles of a Buck converter and of a Boost converter. Figure 3 shows the circuit for a Buck-Boost converter. When operating as a Buck Converter, transistor 2 is turned off, and a high frequency square wave is applied to the gate of transistor 1. When this voltage is high, current flows through the inductor, charging it and the capacitor, while supplying the load, and the Schottky diode D1 is off. When the voltage is low at the gate of transistor 1, it is also off and the magnetic field in the inductor begins to collapse. The back e.m.f. generated by this collapsing field switches the polarity across the inductor, which turns on the Schottky diode D1, causing current to flow through the load still. As this current decreases, the charge in the capacitor during the on period of transistor 1 also adds to the current flowing through the load. This means that V_{out} will be relatively constant. When operating as a Boost converter, transistor 1 is now turned on continually, and the high frequency square wave is now applied to the gate of transistor 2. During the time that transistor 2 is on, the input current flows through the inductor and through transistor 2. Current will not flow through the Schottky diode D2 at this time and the load is being supplied by the charge on the capacitor C, and the load is at a potential of approximately $V_S + V_L$. During the off period of transistor 2, the inductor L is charged and the capacitor C is partially discharged. The inductor generates a back e.m.f. dependent on the derivative of the current as transistor 2 changes states and the inductance of the inductor L. The polarity across L has also now reversed, which adds to the input voltage, generating the same output voltage of $V_S + V_L$ [12].

Figure 2

Smoke and PV data from Summer 2020

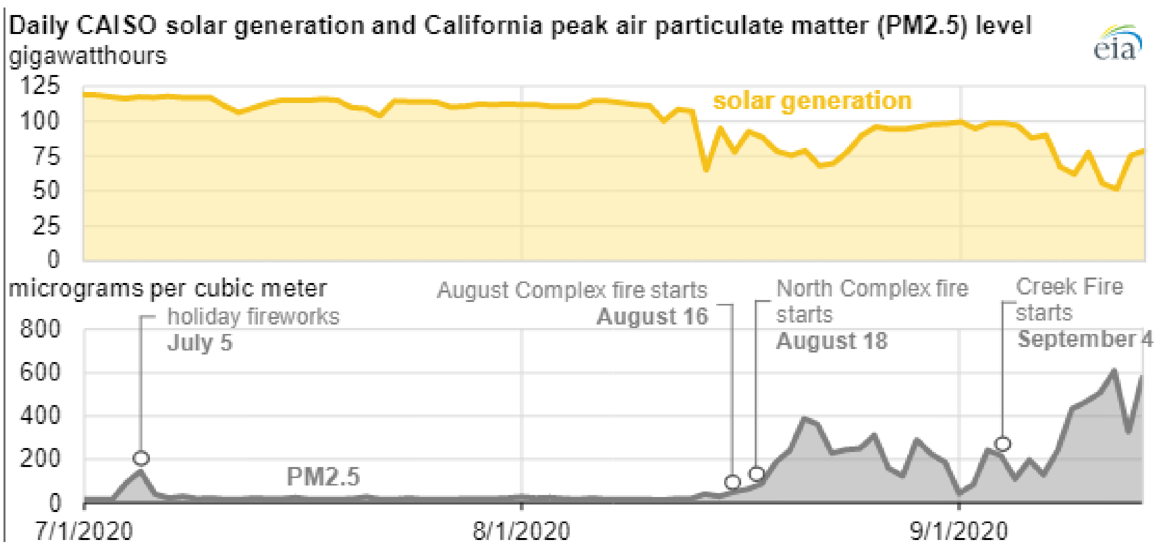
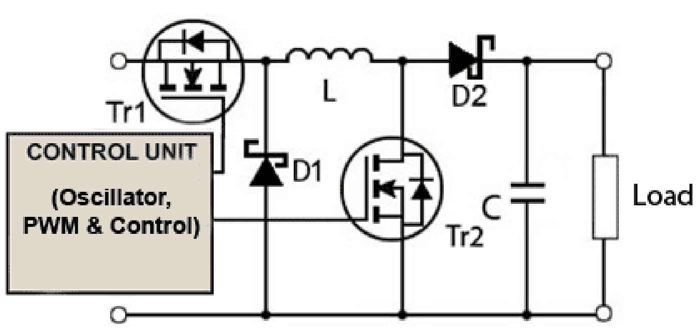


Figure 3

Buck-Boost converter.



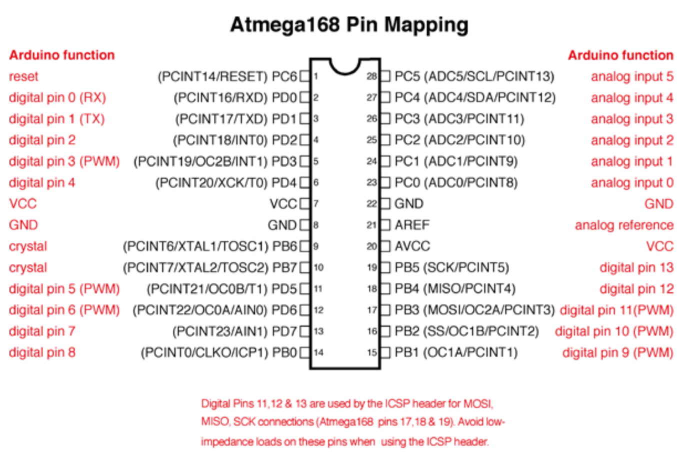
For my portion of the project, I need to have learned about solar generation, batteries, power conversion, power electronics, and power systems protection and control. The classes offered at Fresno State that cover these topics include the following: ECE 90, ECE 102, ECE 121, ECE 128, ECE 151, and ECE 153.

(Anthony)

Two of the main components that we used are a microcontroller and a microprocessor. We needed multiple inputs to read voltages and sensors so the Arduino Uno R3 was perfect for our design specifications. An Arduino is a microcontroller based around an ATmega328P which is run by RISC architecture. It also has SPI and USART interfaces as well as the most important factor for our project, 14 Digital I/O pins with 6 of them being able to provide a Pulse Width Modulation output. This specific Arduino is the most common microcontroller used in projects due to how easy it is to implement in software. It comes with its own integrated development environment and it only weighs 25g and can fit in your hand.

Figure 4

ATmega328P, same as Atmega168



The sensor we used can be directly connected to the Arduino through any of the I/O pins in order to work correctly. It works by sending and receiving a signal and once that signal is interrupted it will notify the Arduino. The Raspberry Pi 4 is a microprocessor that will be reading the data from the Arduino in order to verify that our project is working as intended. It will be connected through the USB port from the Arduino and into a monitor to see the results in real time.

The last part is the Maximum Power Point Tracking (MPPT) which is a DC to DC converter that matches the solar panel and the battery in order to produce the most efficient output. This is done due to the battery while it is charging has a different required voltage then what is being outputted by the solar panel. It's more of a controller that is constantly keeping track of the output of the solar panel and figuring out what the best power it can output in order to charge the battery as efficiently as possible.

To complete this project an understanding of how solar panels, voltage and current regulation, microprocessors, and design flows worked in relation to what was learned in previous courses : ECE 90, ECE 128, ECE 176, and ECE 178.

DESIGN

Procedure

(John)

My portion of the project generally deals with the overall design of the system, and with handling adjustments that need to be made to ensure it is in proper working condition. Keeping that in mind, the procedure for designing this project was as follows:

To start, there were three goals that I needed to accomplish. Goal 1: simulate the conditions properly, Goal 2: supplement the decreased generation, and Goal 3: clean the surface of the panel. Goal 1 was accomplished by containing smoke generated by a device in a tank, which is positioned in front of a very bright LED spot light. The light is filtered through the smoke and onto the solar panel. This simulates the smoke in the air filtering the sunlight. The ash that settles on solar panels during wildfires was simulated through the use of diatomaceous earth, which is a powdery substance with a similar particle size to ash. The solar panel has a maximum power output of 100 W (17.9 V, 5.72 A), and it reaches the maximum voltage when the LED spot light is shone onto it without obstruction, but only reaches 50 mA. The amount of decrease in generation was monitored and the concentration of smoke was adjusted accordingly, with a margin of a 25% to 60% decrease as the target. Since the current measurement was so low to begin with, and stayed at the 50 mA value, the parameter that was measured was the voltage. The measurement for current was done in series with a 2 Ω resistor. The target range for the smoke was reached easily and lasted for 2 minutes. Goal 2 was accomplished through the use of a battery, which was able to be charged by the solar module. When the output of the solar panel is 12.5 V or lower, the Arduino sends a signal to a MOSFET which allows the battery to supplement the generation. This Arduino control was handled by my partner, Anthony. The output from the solar panel and battery is connected to a buck-boost converter device and converted to 12 V-DC. Goal 3 was accomplished through the use of a 12 V DC powered air compressor, powered by generation from the solar array and the battery. The air compressor forces air through a thin, flexible polyethylene tubing with caps at the ends and small holes in the sides, which is attached to the side of the solar panel. The control of the operation of the air compressor, which is dependent on an optical sensor detecting the “ash” on the panel, was handled by my partner, Anthony. When the optical sensor detects ash on the panel, it outputs a signal to a second MOSFET which allows the system to power the air compressor. Schematics, box diagrams, and pictures of the project will follow in the next section. A full list of components used in this project is given in Appendix A. Before that, however, Anthony will share his portion of the design process.

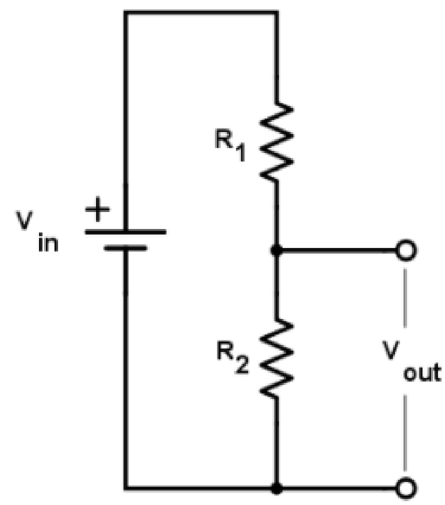
(Anthony)

My portion of the project was understanding the software and how to combine all the parts together to get a functioning project. The first step was understanding how the Maximum Power Point Tracker could be implemented with a buck-boost converter and a battery. The main goal for me was to reduce the amount of power that was being wasted. We wanted to optimize and get the best possible efficiency than just a stand alone solar panel. To start off I needed to figure out how to constantly calculate the open circuit voltage and short circuit current in order to get the maximum power point. I knew I could use the Arduino I/O pins to read the voltage but it first needed to be scaled down as the voltage limit exceeded the limit on the Arduino. To go more in depth into the Maximum Power Point Tracker (MPPT) I realized that there were many ways to implement this algorithm but I chose the perturb and observe (P&O) method as it aligned perfectly with what we needed[15]. I also had to find a way to send the data from the MPPT in relation to the Buck/Boost converter in order to step up.or step down the voltage. The duty cycle plays a huge role in what we will be using as the load is what will be the connection to the other things we will have connected. I then decided to test what I had on simulink using a PV model as the base and created my own Maximum Power Point Tracker function using the duty cycle to compare previous and current power on Matlab. It worked as intended and the next step was to implement the design with the Arduino, all while connecting the sensor and the air compressor at their correct voltages.

The first physical portion of the project that I started on was resolving the issue of the specifications of the Arduino. The Arudino has very specific voltages that it can output and input and since I was going to be working directly with the output of the solar panel I knew this would be the first issue to tackle. The solar panel can output up to 21.6V and 5.72A and an Arduino will only be able to read from the range of 0-5V according to the datasheet. To solve this problem I knew that the Arduino had an analog input with 5 pins that I could use to convert to a digital value. The way it works is the Arduino has 1023 integer values that you can multiply with the input value in order to get the voltage. This still didn't solve the problem of what would happen if the voltage was anything over 5V and according to the solar panel we could reach up to 21.6V so what I decided to do was linearly scale down the voltage from what the solar panel could output. I used a multiple a 5 and decided to round up the 21.6V to 25V just in case we did see a jump of voltage and that way we wouldn't destroy the Arduino. Also since I didn't use a capacitor to remove any spikes or smoothen the output, the extra range in voltage I left should work in correlation to the scaled down version safety of the 5V max I want to ensure. To actually reduce the voltage in order to be properly read I built a voltage divider circuit that would scale down the voltage by at least factor of 5. I used the voltage divider equation of Vout=Vin(R2/(R1+R2)). The scaling factor I wanted was .2 so I got the values of 100kiloOhm and 20kOhm. With these resistor values I now knew the output voltage would be safe in order to use with the Arduino. Vin would be the solar panel voltage input, R1 is the 100kiloOhm resistor, R2 is the 20kiloOhm resistor, and Vout is where the Arduino analog pin will be connected according to Figure 5.

Figure 5

Voltage divider circuit.



The input voltages will be scaled down according to the ratio and the table belows shows some examples.

Table 1

Voltage divider ratio.

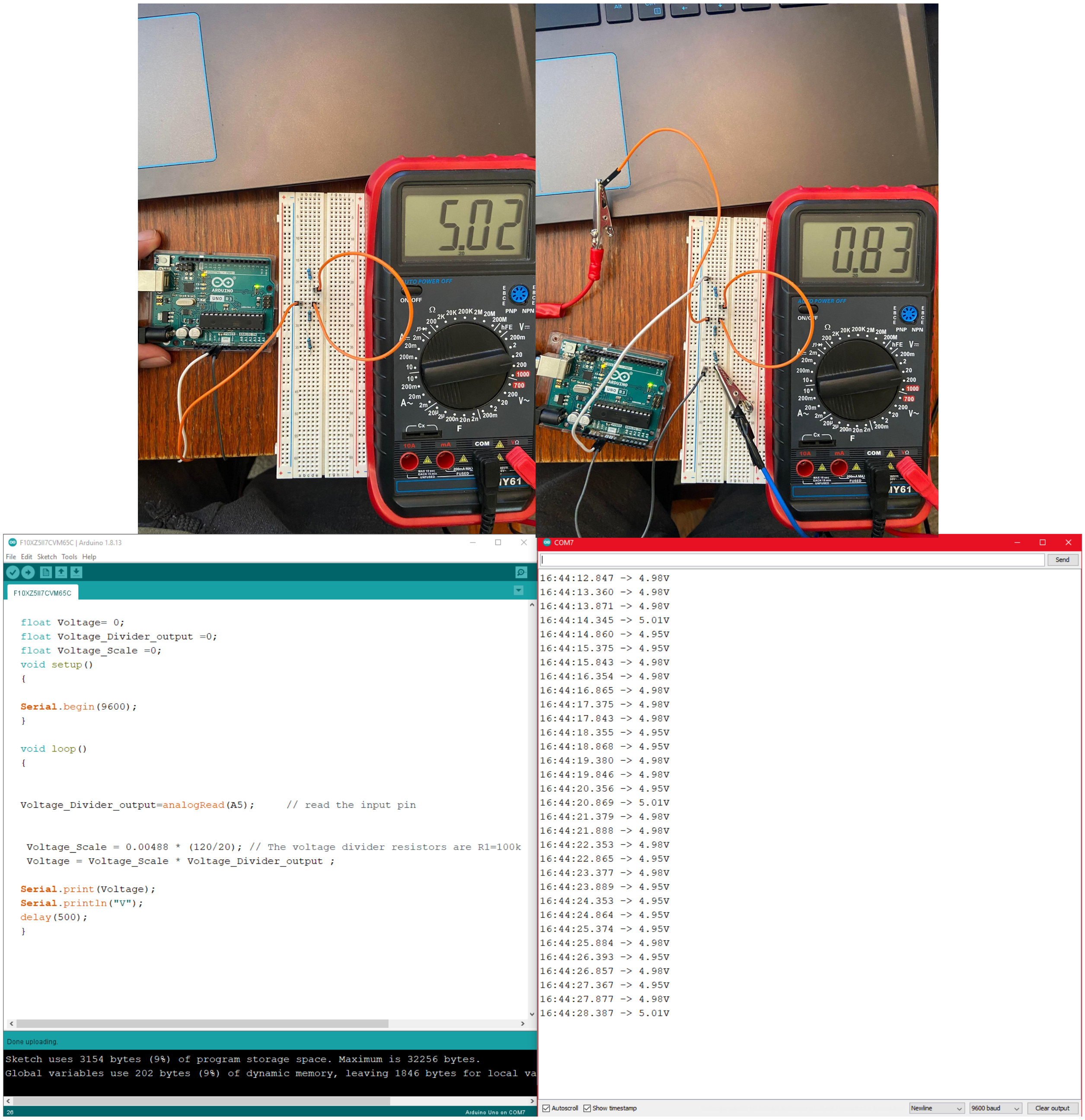
Input Voltage From Solar Panel (V)	Resistor R1 (kΩ)	Resistor R2 (kΩ)	Output Voltage To Arduino (V)
25	100	20	4.166
22.8	100	20	3.8
20.6	100	20	3.433
18.4	100	20	3.066
16.2	100	20	2.7
14	100	20	2.333
11.8	100	20	1.966
9.6	100	20	1.6
7.4	100	20	1.233
5.2	100	20	.866
3	100	20	.5
.8	100	20	.133
.1	100	20	.016
0	100	20	0

To test it out I built the voltage divider circuit and used the 5V output from the Arduino to supply a constant voltage and measured it to be 5.02V directly out of the Arduino as it should. I then used the exact voltage and put it through the voltage divider circuit and according to the equation

I should have got .836 and I got .83 which is almost exactly what I expected as shown in Figure 6.

Figure 6

Input/Output voltage and Input/Output reading.



Then I decided to test out higher than 5V with the power supply for a constant voltage and to see if the voltage divider could handle the voltage of anything higher than 5V. I connected the voltage supply and set it to 15.3V and grounded the power supply with the Arduino ground. I was worried that the resistors wouldn't be able to handle the voltage or if the Arduino would even read it. I ran the code and on Arduino it read that it was reading 15.25V according to Figure 7 .

Figure 7

Output voltage using power supply.

Then I decided to use the solar panel instead of the power supply to see how the Arduino would react with the rapid change of voltages. I grounded the Arduino with the solar panel, power supply, and multimeter. I used the multimeter in order to verify the voltage reading from the

Arduino was accurate. When the solar panel had an output voltage of 14.24 according to the multimeter the Arduino had a reading of 14.26 which is what I expected according to Figure 8.

Figure 8

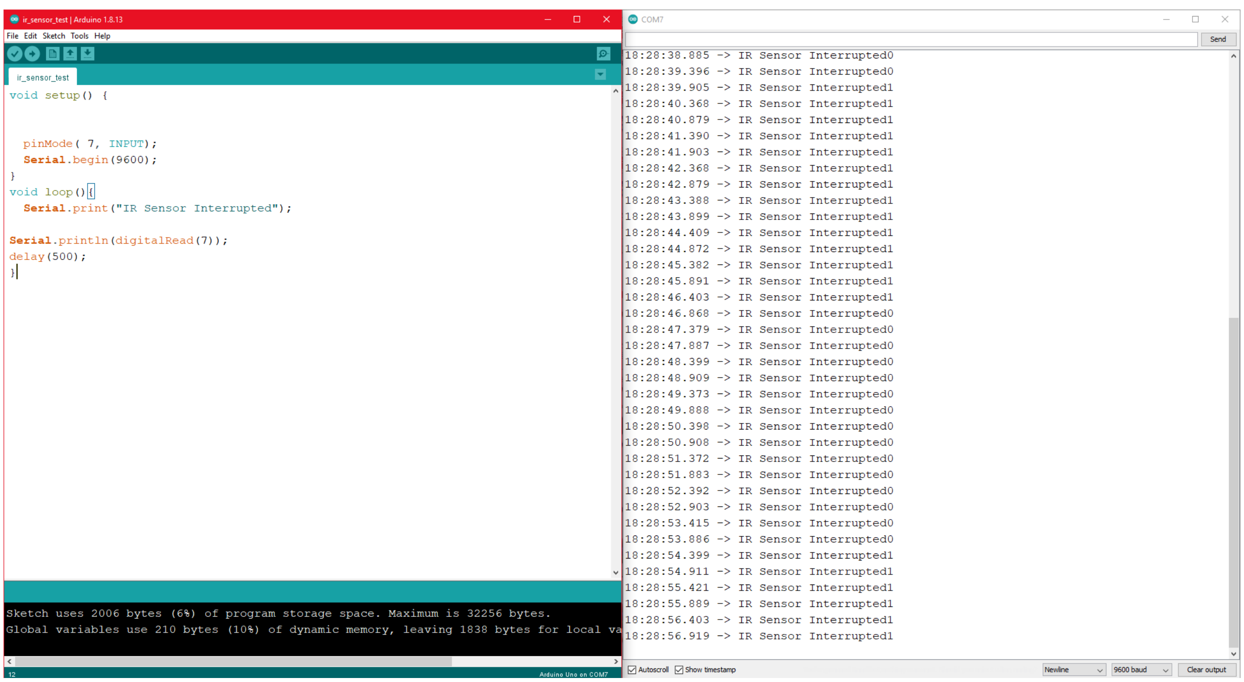
Output Voltage using solar panel.

After I finished the voltage divider portion, I realized that I can comfortably read any voltage with the Arduino that the solar panel or battery can output.

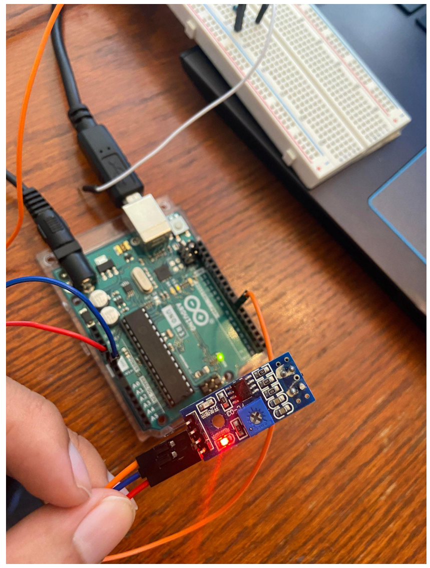
Next was the IR sensor as shown in Figure 9. The IR sensor we used was a SMAKN TCRT5000 and the way it works is that it sends a light signal, and it is reflected and received by the receiver. If the signal is interrupted by the light not being reflected correctly it will let me know that the signal was interrupted. This is what I am using to detect the ash on the solar panel as the solar panel is a straight surface and it can reflect light really well. I connected the IR sensor to the Arduino by using the VCC from the sensor to the 5V output of the Arduino, the GND to the GND, and the D0 to the 7th Digital pin. The sensor has a calibration tool that seems like it is just a potentiometer that regulates the voltage in order to get a different precision. I turned the knob with a screwdriver in order to get the correct distance and sensitivity in relation to the ash on the solar panel. In order to see that the Arduino was reading correctly I set the 7th Pin to be an input as that is where I connected the IR sensor and then used the digitalRead syntax as it returns either a HIGH or LOW which in my case means that there is no ash or ash on the panel as shown in Figure 9.

Figure 9

(a) IR sensor reading.



(b) IR sensor connected.

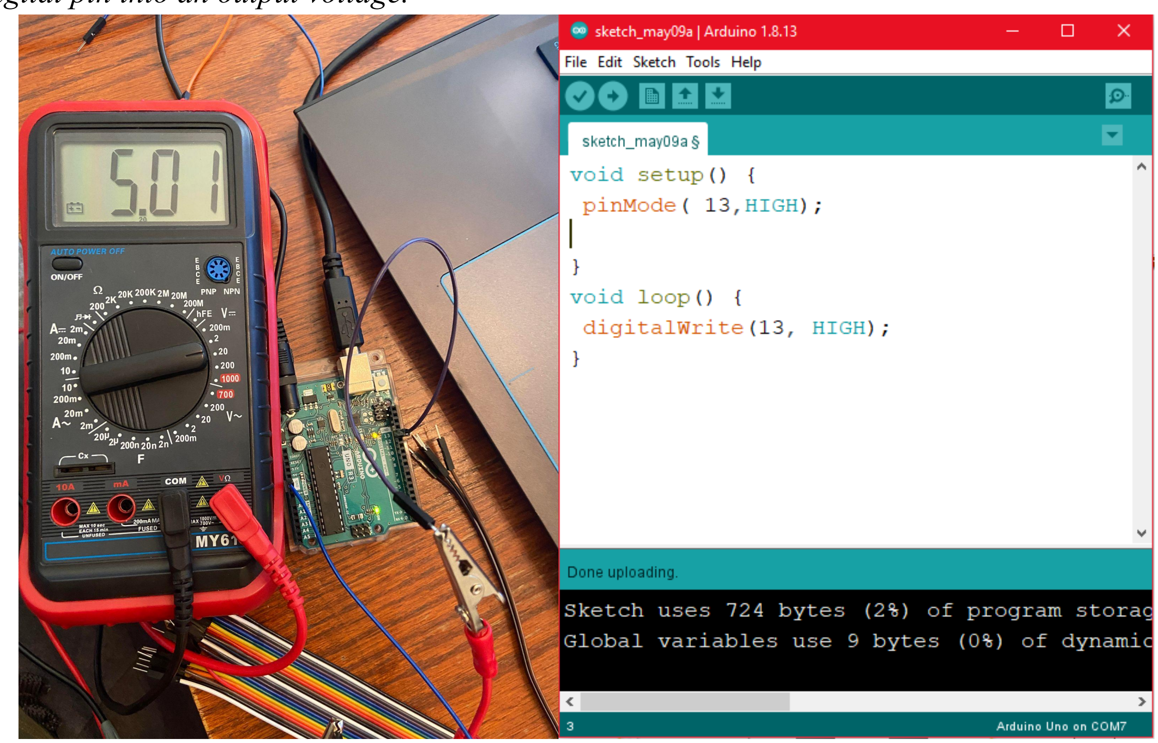


After the IR sensor was working as I wanted it to, I had to move on to the next task which was sending out the data to the air compressor in order to turn it on.

The air compressor can only be activated by 12V and there was no way for the Arduino to send out 12V. Thankfully the Arduino allows some of its digital pins to work as outputs as shown in Figure 10 . The way the output digital pins work on the Arduino is that by setting the pin to an output it goes to a low-impedance state which lets high current go through and low voltage.

Figure 10

Digital pin into an output voltage.

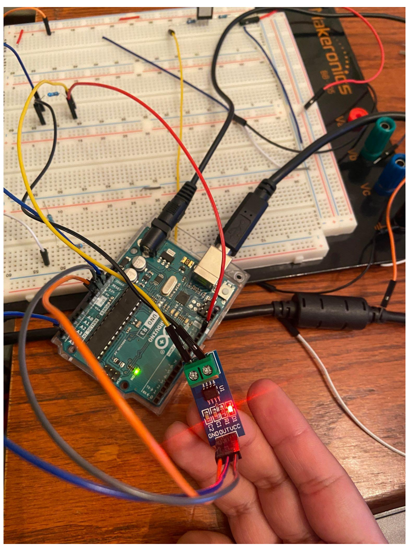


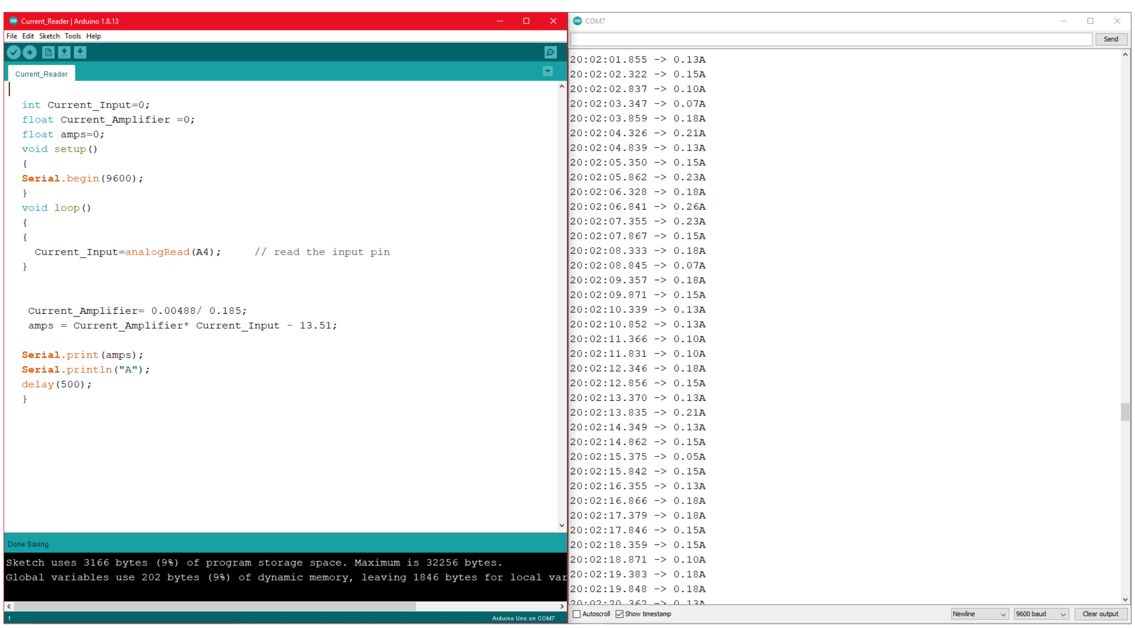
After figuring out and completing this portion of the project I moved on to how I would get this 5V signal to activate when the ash would be detected and then how to get that 5V to get to 12V. I then designed a Buck Boost converter according to the example on TEXAS INSTRUMENTS but when building it on the breadboard it did not work as intended and even with a potentiometer the accuracy wasn't what I wanted. I then looked at some Buck Boost converters online and we purchased one with an adjustable output of up to 30V. This solved the issue on how the Arduino would be able to output enough power in order to turn on the air compressor.

After completing this step, I realized that I was missing the current reading for the solar panel and the output of the Arduino. I used a current sensor that can be attached to the Arduino called the ACS712 as shown in Figure 11.

Figure 11

Current reading using ACS712.





Schematics

(John)

Figure 12 shows a schematic diagram of the project. This schematic identifies the individual components and how they are connected to each other. Figure 13 is the PSIM implementation of my part of the project, with labeled signals and measurements in place of the microelectronics from Anthony's part. Figures 14 and 15 show the simulation of the maximum power point tracking and buck boost generation and corresponding result in PSIM. The capabilities of PSIM to accurately simulate the project are limited, but the results show that the Maximum Power Point Tracker is capable of adjusting the voltage when the output of the solar module is decreased. Figure 16 shows various pictures during the testing of the decreased generation due to smoke and ash. Figure 17 shows various pictures of the system when testing indoors with no smoke or ash.

(Anthony)

Figure 18 shows the interaction between the IR sensor receiving the data and constantly sending it to the Arduino. After the Arduino notices that the data has changed due to ash covering the panel it will notify the compressor to turn on causing the ash to be blown off the solar panel. Figure 19 shows the Maximum Power Point Tracking(MPPT) flowchart that will be taking care of finding the most optimal power by looking at its duty cycle and calculating the necessary information from the current and voltage as explained in the Maximum Power Point Tracking down below.

Figure 12

Schematic diagram of the project.

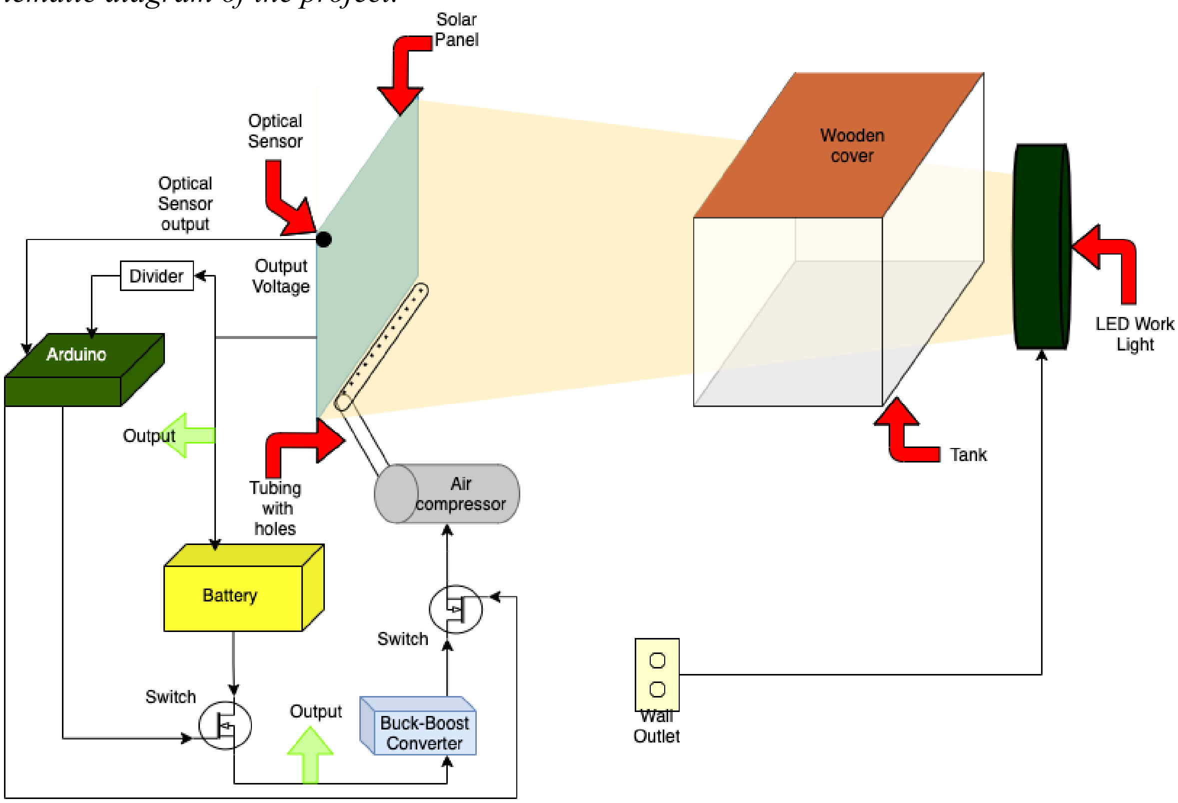


Figure 13

PSIM system.

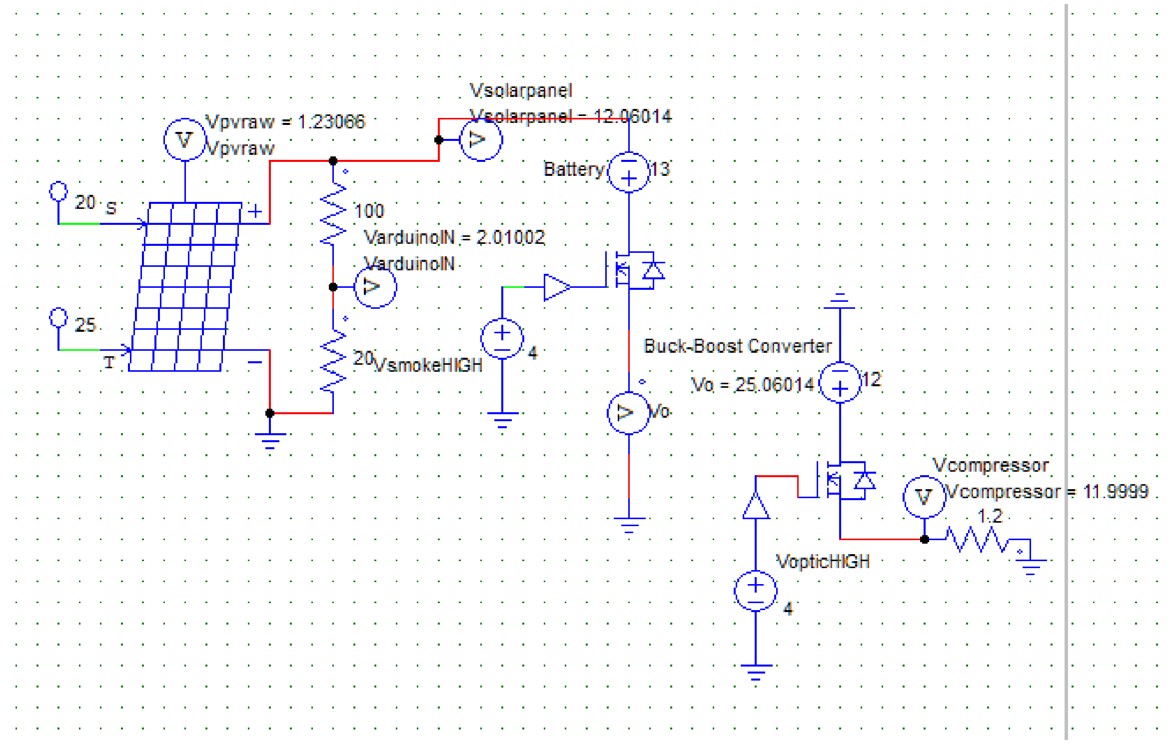


Figure 14

PSIM MPPT and Buck-Boost circuit.

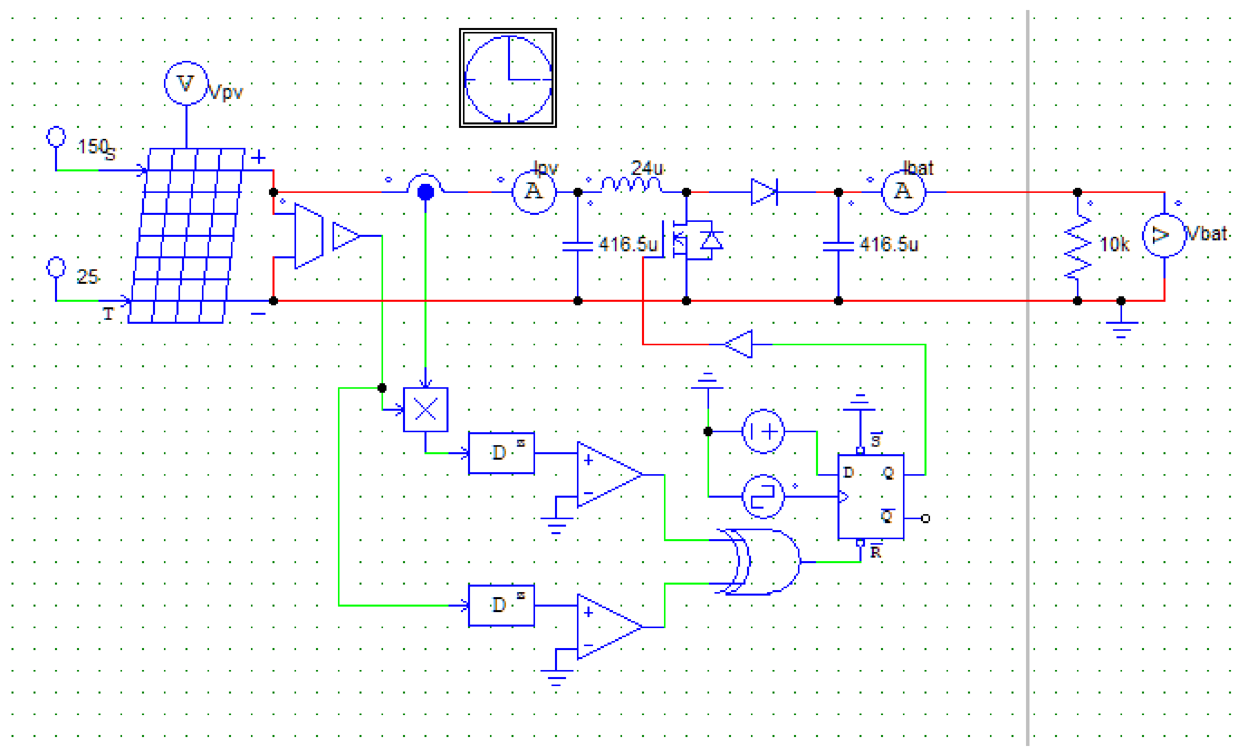


Figure 15

PSIM MPPT and Buck-Boost results.

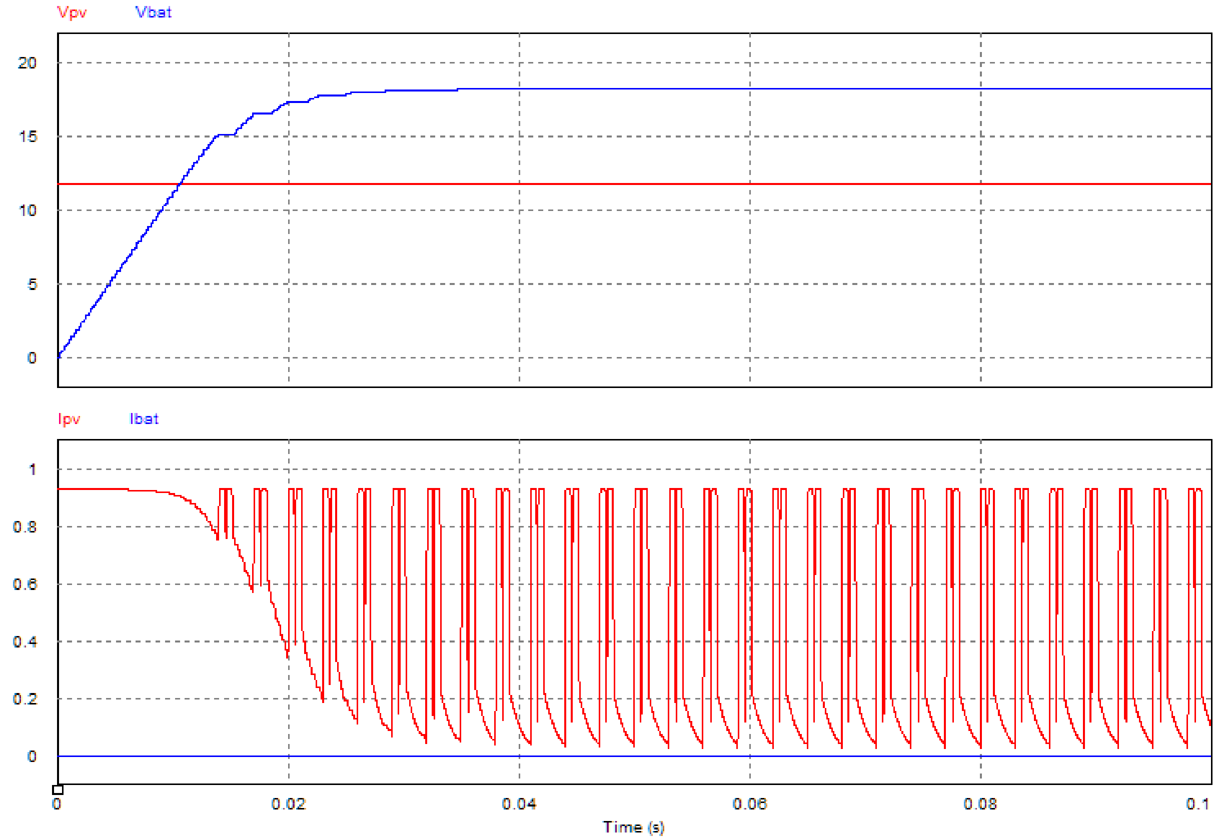
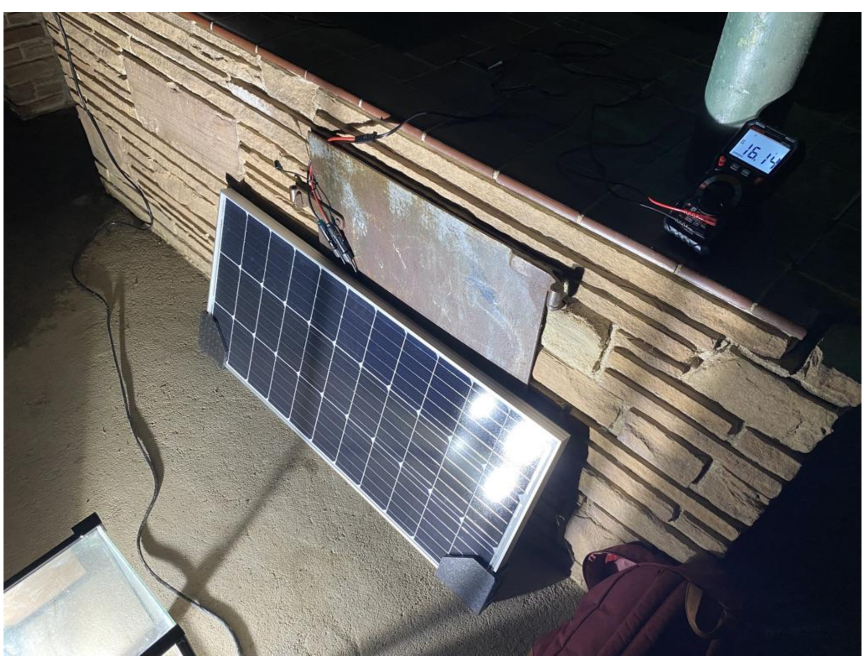


Figure 16

(a) Testing at no smoke and result.



(b) Testing at 25% - 60% reduction.

(c) Testing with sunlight and ash.

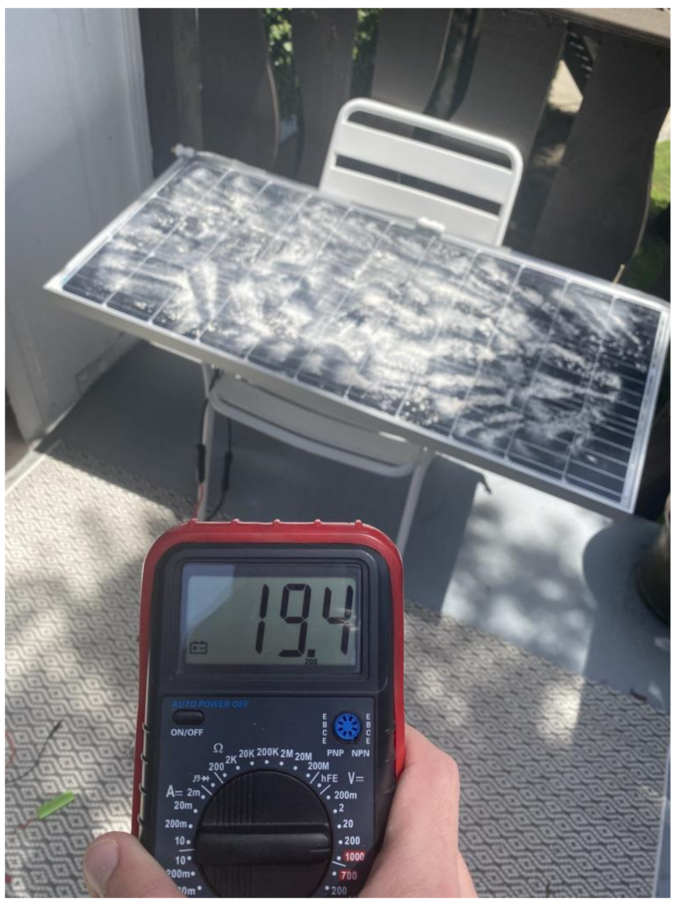
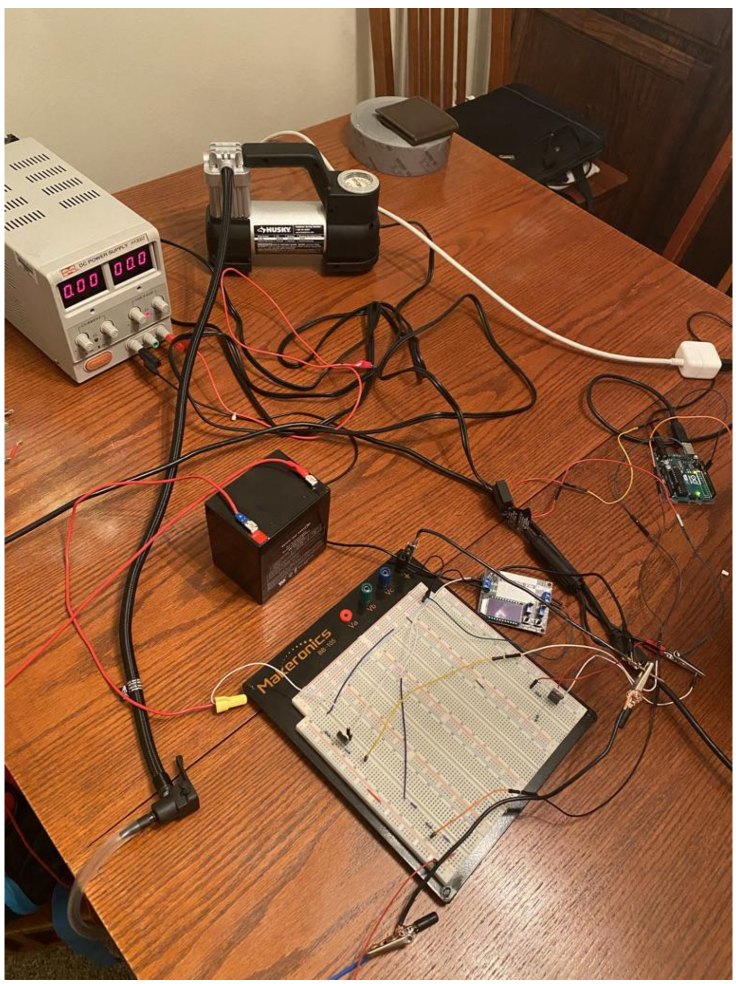


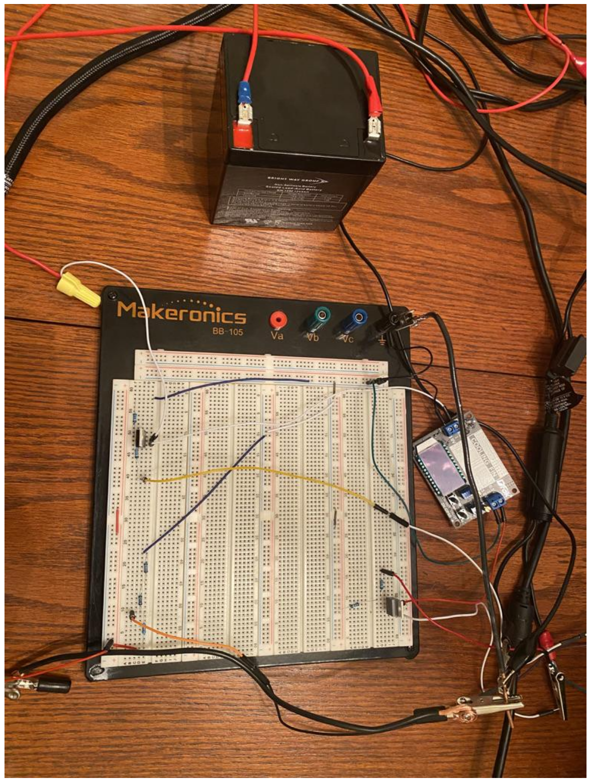
Figure 17

(a) Solar panel with tubing and optical sensor attached.

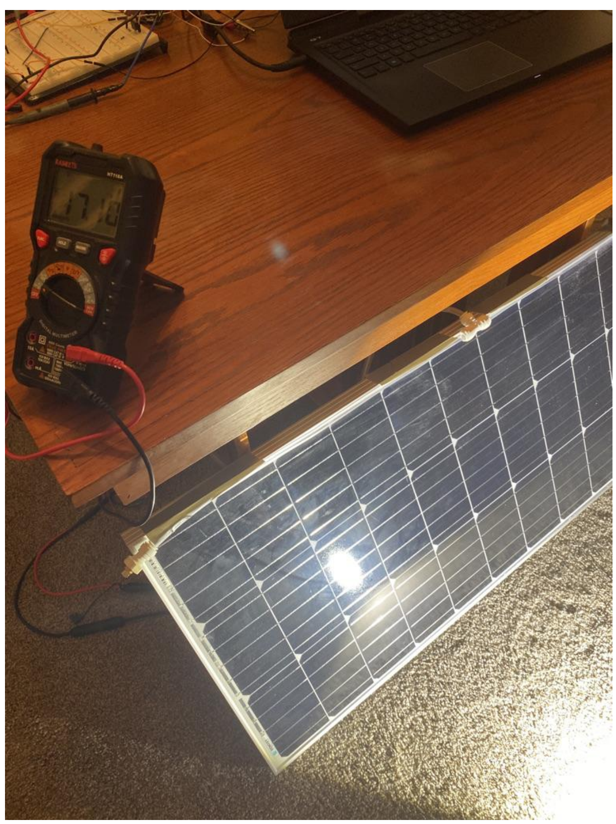
(b) Breadboard, battery, air compressor.



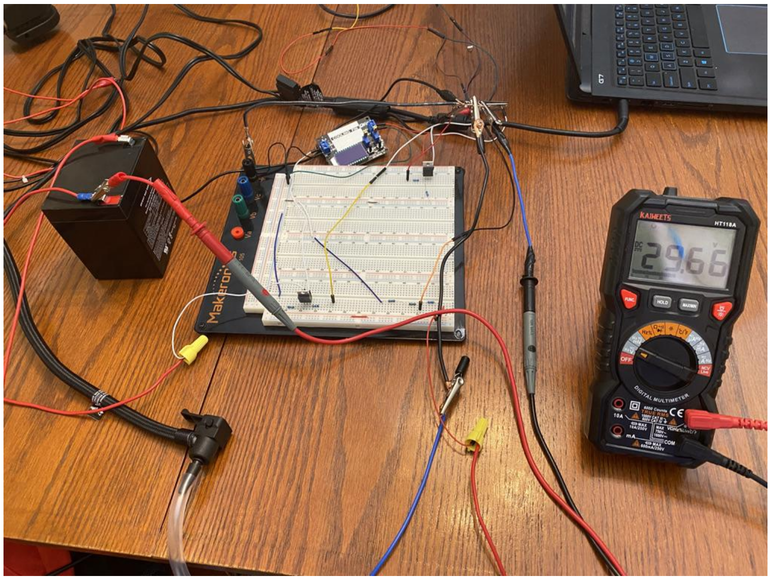
(c)Closer view of breadboard and battery, with buck-boost converter on the side.



(d) Solar panel measurement.



(e) Solar panel and battery output added



(f) Buck-Boost input and output

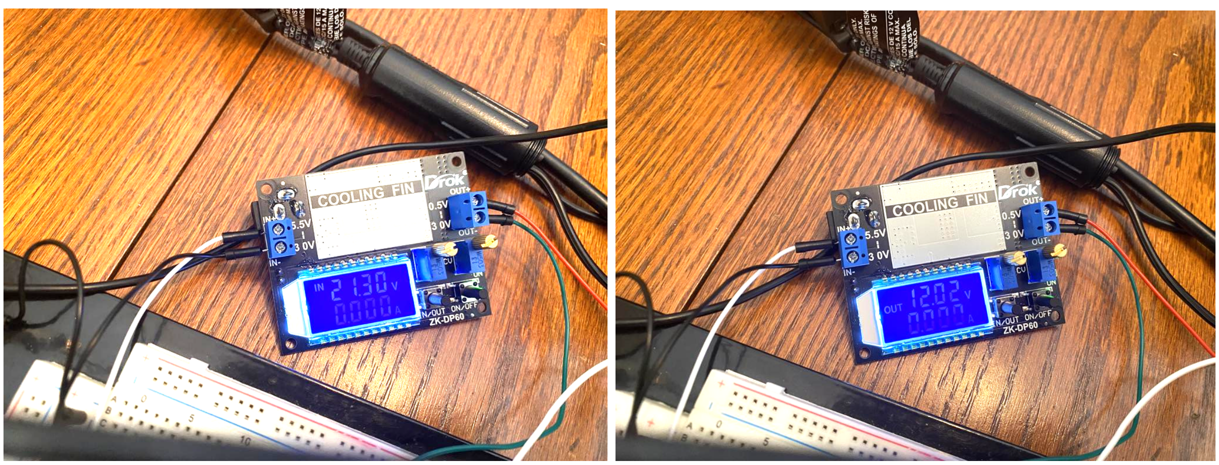


Figure 18

Flowchart to activate air compressor.

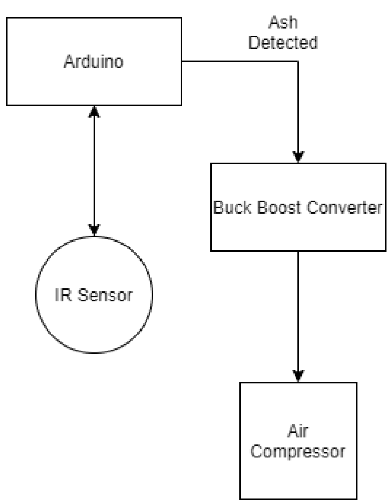
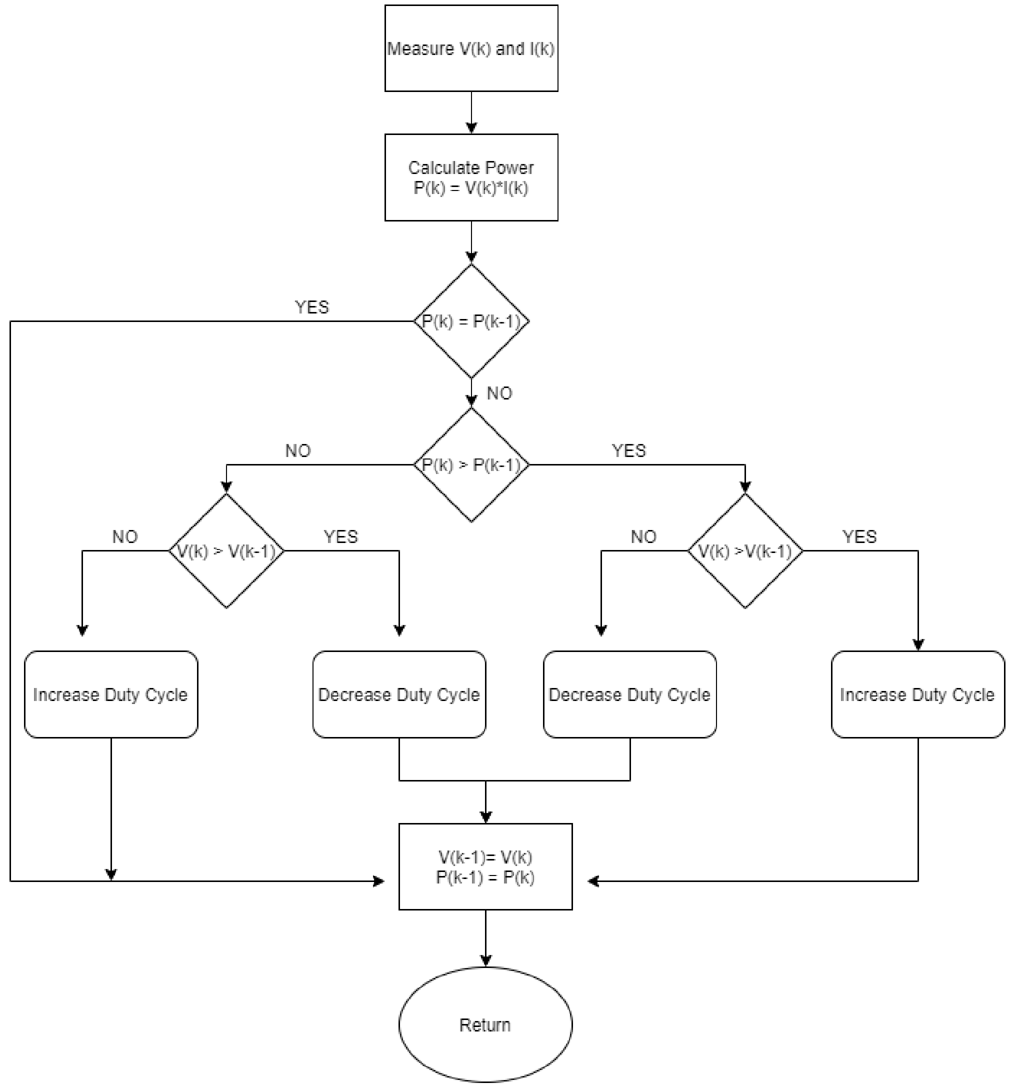


Figure 19

Flowchart for Maximum Power Point Tracking.



The Maximum Power Point Tracking is another way of having a smart battery charger. A normal battery charger just takes in what it can while not optimizing the extra voltage or current in order to charge at its greatest efficiency. The Perturb and Observe algorithm uses a buck boost converter and an Arduino in order to deliver the maximum amount of power to the battery. The way the tracking works is through constant measuring and comparing of the solar panel output voltage and increasing or decreasing the power to extract the maximum power.

Figure 20

Matlab Maximum Power Point Tracker code.

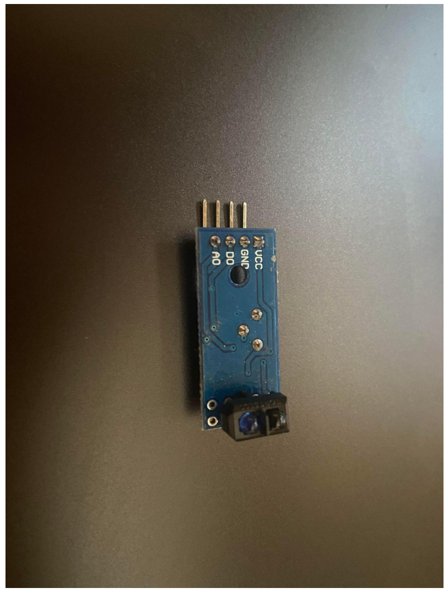
```
function d= dutyCycle(Vpv,Ipv)
persistent Vpre Ppre dpre   %retain values
if isempty(dpre)
    Vpre = 10;
    Ppre = 20;
    dpre = 0.3;
end
Ppv = Vpv*Ipv;     %calculate power of current inputs
DeltaD = 0.01;      %Initialize
    if(Ppv == Pre) %Compare the current Power with the previous Power
        d= dpre;   %If they are equal to each other still then the duty cycle remains the same
    else
        if(Ppv>Ppre)%If the current Power is greater then the previous, check if current voltage is greater than previous also
            if(Vpv >Vpre)%If it is greater the duty cycle must increase
                d= dpre +DeltaD;
            else
                d= dpre-DeltaD; %If not greater then decrease the Duty Cycle
            end
        else
            if(Vpv>Vpre) %If Ppv<Ppre, check the current adn previous voltages again
                d=dpre -DeltaD; %Power is increase with voltage rise to riase voltage more
            else
                d= dpre +DeltaD; %Power is decreased with voltage rise to decrease voltage more
            end
        end
    end
    Vpre= Vpv;
    Ppre= Ppv;
    dpre = d;
```

Figure 21

Arduino UNO R3.

Figure 22

Ash detection sensor.



Data

(John)

Table 2

Solar panel testing data

Result #	Distance (feet)	Angle (degrees)	Light Source	Smoke	Ash	Voltage (V)	Current (A)	Power (W)
1	2	90	LED	No	No	17.54	0.05	0.877
2	4	90	LED	No	No	16.2	0.04	0.648
3	2	45	LED	No	No	17.3	0.05	0.865
4	2	15	LED	No	No	16.4	0.04	0.656
5	N/A	90	Sun	No	No	20.6	3.29	67.774
6	N/A	90	Sun	No	Yes	19.4	2.14	41.516
7	N/A	80	Sun	No	No	20.2	2.78	56.156
8	3	75	LED	No	No	16.14	0.05	0.807
9	3	75	LED	Yes	No	7.17	(0.05)	(0.359)
10	3	75	LED	Yes	No	10.81	0.05	0.541
11	3	75	LED	Yes	No	11.62	(0.05)	(0.581)

(Anthony)

Voltage Divider Circuit

Vout=Vin*(R2/(R1+R2))

5V=25V(R2/(R1+R2))

R1 = 100Kohm

R2= 20Kohm

Voltage Divider Arduino

The Arduino has an Analog to Digital Converter installed that converts analog voltage to a specific digital value. The way it works is through a ratiometric value to relate the Analog and Digital value.

(ADC Resolution/ System Voltage) = (ADC Reading/Analog Voltage Measured)

Since I am using the 10-bit ADC of the Arduino I can use (5/1024) which is equivalent to .00488. This is how I will convert back to the original value as I can do the opposite of the voltage divider equation to get the original voltage value back. In the voltage divider equation, I used (20/120) for the value of the resistance so in order to get the original voltage I will do the opposite and multiply but by the reciprocal so (120/20). I can then multiply the ratiometric value by the reciprocal of the resistance in my voltage divider circuit. This new value is the ratio that will be multiplied by the analog input that the Arduino is reading in order to give the correct output of what Voltage we are measuring.

Voltage = (input analog voltage) * (120/20) * (5/1024)

Voltage Divider Code

```
float Voltage= 0;
float Voltage_Divider_output =0;
float Voltage_Scale =0;
void setup()
{
Serial.begin(9600);
}
void loop()
{
 Voltage_Divider_output=analogRead(A5);     // read the input pin
 Voltage_Scale = 0.00488 * (120/20); // The voltage divider resistors are R1=100k and R2=20k
// 5/1024 =0.00488
 Voltage = Voltage_Scale * Voltage_Divider_output ;   // output original voltage
Serial.print(Voltage);
Serial.println("V");
delay(500);
}
```

Current Sensor(ACS712)

The current sensor has a 5.0V, single supply operation with a 66 to 185 mV/A output sensitivity according to the datasheet in figure 23. It also has a built in ratiometric output from the supply voltage which will make the calculations around the same as what I did for the Voltage divider.

Figure 23

Data sheet for ACS712.

FEATURES AND BENEFITS
- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 µs output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at T_A = 25°C
- Small footprint, low-profile SOIC8 package
- 1.2 mΩ internal conductor resistance
- 2.1 kVRMS minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratiometric output from supply voltage

I soon realized that the data sheet from the first page was different from the following pages depending on the current amount. Since I had the 20A current sensors then the sensitivity should be 100 mV/A instead of 185 which would completely change the output according to figure 24.

Figure 24

5/20/30 A sensitivity.

Selection Guide

Part Number	Packing*	T_A (°C)	Optimized Range, I_P (A)	Sensitivity, Sens (Typ) (mV/A)
ACS712ELCTR-05B-T	Tape and reel, 3000 pieces/reel	–40 to 85	±5	185
ACS712ELCTR-20A-T	Tape and reel, 3000 pieces/reel	–40 to 85	±20	100
ACS712ELCTR-30A-T	Tape and reel, 3000 pieces/reel	–40 to 85	±30	66

*Contact Allegro for additional packing options.

The calculation for the correct current was to convert the analog value of the current sensor into a digital value like I did for the voltage. I used the same ratiometric as that will always be the same of (5/1024) to get .00488. Next, I divided by the sensitivity of 100 mV/A which is .1 to get the scaling factor that I needed. Then I multiplied the scaling factor by the analog input and subtracted the ratio of the voltage being provided by the Arduino/sensitivity.

Current = (input analog current) * (input voltage / sensitivity) -(half the voltage for polarity from Arduino/sensitivity)

Current = (input analog current) *(.00488)/(.100) - (2.5/.100)

Code for Current Sensor

```
int Analog_Input=0;      //input integer
float Scale = .00488;    // (5/1024) ADC
```

```
 float Sensitivity= .100;       // 100mV/A sensitivity
 void setup()
 {

 Serial.begin(9600);    // data rate 9600 bps
 }

 void loop()
 {
 Analog_Input= analogRead(A1); //read input analog pin
 float Voltage = Analog_Input * Scale; // calculate Voltage
 Voltage = Voltage-2.5;         // Take away polarity since we want positive voltage
 float Current = Voltage / Sensitivity;  // Calculate current
 Serial.print(Current);
 Serial.println("A");

 delay(500);
 }
```

IR Sensor code

```
void setup() {
 pinMode( 7, INPUT);   //pin 7 be input of IR sensor
 Serial.begin(9600);
}
void loop(){
 Serial.print("IR Sensor Interrupted");

Serial.println(digitalRead(7));  //Print 1 or 0 if IR sensor detects ash
delay(500);
}
```

Set Pin output of 5V for transistors code

```
void setup() {
 pinMode( 13,OUTPUT); // set pin 13 to be an output

}
void loop() {
 digitalWrite(13, HIGH); //set pin 13 to output 5V
}
```

Schedule (John)

Table 3

Statements on progress according to project schedule.

Task	**Due**	**Delegation**	**Statement**
-PSIM design -Obtain Microcontrollers	1/15/2021	-John -Anthony	-This task was completed on time -This task was completed early
Research needed DC conv., sensor and battery backup	1/22/2021	Both	This task was completed on time
Obtain all preliminary materials	1/30/2021	Both	This task was completed on time
-Gather simulation data -Program microcontrollers	2/28/2021	-John -Anthony	-This task was completed late (3/8), due to continued effort to accurately represent the system in PSIM
Build and test buck/boost converter	3/8/2021	John	This task is still in progress (delegation has moved to Anthony)
-Test individual components -Test microcontrollers	3/15/2021	-John -Anthony	-This task was completed on time -This task was completed late (3/31)
Put it all together	3/31/2021	Both	This task was completed
Test and gather data	4/16/2021	Both	This task was completed late (5/2)
Prepare Poster/Report	4/23/2021	Both	This task was completed late (5/8)
Prepare Presentation	5/7/2021	Both	This task was completed late (5/9)

ALTERNATIVES

(Anthony)

- Alternative to reptile tank - Build custom plexiglass to surround the whole solar panel with pockets to introduce smoke.
- Alternative Solution - Cool down the solar panel by applying water or mist onto the panels through a tube placed directly on top.
- Alternative Solution - Use different kinds of solar panels such as polycrystalline or thin-film.

(John)

- Alternative to smoke generating devices - Use a pump to aerosolize diatomaceous earth, salt powder, flour, or other fine particulate matter.
- Alternative solution - Apply a superhydrophobic coating to surface of solar panel to clean them easier
- Alternative to buck-boost converter - Transformer

ECONOMIC ANALYSIS (Anthony)

The benefit of our project is around improving the efficiency of the solar panel. In an economic sense the customer will see more power output from the solar panel year-round no matter the weather which makes the idea of a solar panel more attractive. In making the solar panel more efficient it will sway the view into a more attractive purchase.

Table 3

Home Rooftop efficiency in Fresno, California

Rooftop build (6 Panels)	Cost ($ amount)	Efficiency %	Efficiency (Smoke)	Efficiency (Average) watts	Efficiency (Overall) %
Solar Panel (Renogy)	624	21	6.3	8190	13.65
Solar Panel (Our build)	764	31	9.3	12090	20.15

The table above states for a home with 6 panels located on its roof using data from Google's Project Sunroof[13]. The solar panel in the example is one we purchased from Renogy and in terms of efficiency we are stating how much power is converted into the battery itself. The best possible outcome is that all 6 panels produce the same amount at its maximum efficiency but that is impossible due to sunlight not always being present and temperature. These two issues cause the biggest problems and that is why solar panels work best in lower temperatures but with the sun still being present. The two scenarios are the two builds, no installation cost is included as we are only focusing on the solar cell efficiency. Our build has a cost of $140 more due to the inclusion of our equipment that we connected. The build we created overall should have a greater efficiency of 6.5% over just having the solar panels present. Having this efficiency can financially benefit the customer in the long run as it will blow off any built-up dust or ash to keep its efficiency at a higher rate.

SAFETY ANALYSIS (Anthony)

Throughout the testing of our project, we made sure that safety was always a priority as a solar panel can be a fire hazard if not maintained correctly. The proper electrical tape and gloves were used when checking the voltage and current of the solar panels. This was mandatory as the output of the solar panels didn't have the best way for the measurement tools we had. Before connecting anything, we made sure that it was grounded correctly and that we were using the correct connections. Also, before connecting the Arduino, voltage dividers were used in order to reduce the voltage to negate the probability of destroying or causing damage. When using the smoke bombs, we made sure to stand at least 3 feet away and did our first test outside at a park where nobody was present. When releasing the smoke from the container we made sure to step even further away and in the opposite direction that the wind was blowing as the device created so much smoke that traveled quickly in the direction of the wind. When working on the project we made sure to be indoors as the surface of the solar panel gets insanely hot and can cause a shock if not handled properly. The solar panel itself was handled with care and before we used them we made sure to carefully check for damages and even unboxed the safely strapped connections in the back to observe everything was soldered correctly.

RESULTS

(John)

Problems

While the simulation of the project works exactly as intended, and most of the components work perfectly and work together, some crucial pieces of the project did not work when we finally put it all together. Most notably was that the MOSFET switches did not work as expected from the PSIM simulation, passing only about 5 volts from the drain to the source when on. This meant that we could not use the system as designed. The MOSFET switch for the air compressor to turn on had a source node voltage of only .005 V, and so was also malfunctioning. In addition to this, the LED was not able to provide very much current from the solar panel, even though we ended up testing three of them. The expectation that the solar panel could output the proper amount of current came from an early misreading of a measurement, which was off by a factor of 100. This meant that repeated measurements of the current being too low were attributed to a faulty solar panel, but in reality it may have more to do with the specifics of the solar panels and how they interact with the wavelength of light produced by the LED. This is a surprising result because the research on solar panels that was discussed in the background of this report found that the LED work light should have been able to produce nearly as much power as the sun. The useful measurement turned out to be the voltage, which was at its expected value. However, because this was the only useful measurement from the solar panel, the Maximum Power Point Tracker portion of the project (one of Anthony's parts) had to be relegated to theoretical design and calculations rather than actual implementation. We were able to read the small current from the panel but its value was unusable when implemented into the larger system.

Successes

I will now discuss the useful, successful results of my portion of the project, piece by piece. The voltage measurement of the solar panel was used in determining the amount of smoke cover that resulted in the proper 25-60% decrease in generation, so it was actually a 25-60% decrease in voltage. This was because the current measurement did not change drastically when smoke was in the tank. This voltage was able to be tied to the battery's output as well as being taken separately when it was high enough. Both the solar panel and the solar panel plus the battery voltages were able to be converted to 12 V through the use of a buck-boost converter device. This device was used because the Maximum Power Point Tracker was not usable due to the current, and because we were not granted lab access due to the use of smoke in our project. The battery could also be charged with the output from the solar panel, although the total charge in the battery was not measurable other than its voltage and current. The air compressor was able to be powered by the direct output from the Buck-Boost converter with the battery plus solar panel as the input. The air compressor was not quite powerful enough to clear large amounts of the diatomaceous earth from the surface of the panel, but was able to do some, and the connections were well sealed.

Data Analysis

The data used for this section will come from Table 2. Included in this table are: the distance the light source was from the solar panel, the angle the light source took with the solar panel, the light source used, whether there was smoke or not, whether there was ash or not, then the readings for voltage and current, and the calculation of the power. The distance factor made a difference with the LED as the source, and caused a 7.64% decrease in the voltage generated by the panel (Results 1 and 2). The angle at which the solar panel was placed made even less of a difference, with a change of 45 degrees causing only a 1.37% decrease in generated voltage (Results 1 and 3), and only in the extreme example of a 15 degree angle did the decrease become comparable to the change in distance from earlier (Result 1 and 4). When the sun was used as the source, the solar panel generated a much more normal value for the current, but because it would have been too expensive to use a plexiglass tank large enough to cover the whole solar panel that sunlight could have been filtered through, we could not test the smoke condition with the sun as the source. However, we could measure how much a layer of ash decreased the generation and found that the ash layer caused a 38.74% decrease in power generation (Results 5 and 6). Results 8 through 11 were done with the glass tank directly in front of the LED source, and the smoke generating device was lit and the cover opened to control the amount of smoke in the tank based on the voltage generated by the solar panel. When the proper amount of smoke was contained in the tank, it remained there for at least two minutes with the cover completely on, allowing for measurements to be taken.

Conclusion

The original intention of my part in this project was to meet three goals: simulating wildfire conditions, supplementing the generation, and cleaning the surface of the panel. With the results obtained, our hope was to be able to recommend that our design be used in a modular sense on solar panel arrays throughout the state, so that rolling blackouts would become less common. From our results, we can conclude that our design, from the simulation of the smoke filtering sunlight and the ash layer covering the panels, can supplement the decrease in generation caused by these factors with a battery backup and clean the surface with compressed air. More time may be necessary to complete a fully functioning and automatic design, but through simulation and constituent results, we are reasonably confident that the design will meet all three of the goals.

(Anthony)
Problems

Throughout the project a lot of issues came to be just from the Arduino and the parts we got for it. We used Amazon to buy a lot of the components and when testing they would break or just stop working. Everything was so fragile, so we had to be extra careful. Since we were working with high voltage we also had to be careful with the electronics since the buck boost converter could output up to 30V and the Arduino can only handle 5V. The biggest issue we faced was when we were connecting the circuit all together and I sent a 5V output to the transistor from the Arduino to turn on the air compressor the wires melted and the Arduino and my laptop shutoff. We couldn't figure out what happened, but I think we might have not connected something correctly and somehow the voltage exceeded the Arduino's capabilities. I thought we had destroyed my laptop, but it was able to turn back on and the USB port doesn't work anymore where I previously had the Arduino connected. After this situation the Arduino wouldn't read correctly or the pins themselves wouldn't output the correct voltages. This really hurt as I felt we were so close to completing the project that we spent a little over a semester on. Another problem I faced was when trying to measure the current for the Maximum Power Point Tracking it was so low that the power was never even over 1W. This is what caused a lot of confusion for us because we thought the light source we had was the best possible scenario for the highest controlled efficiency so when we had no current we thought we had somehow short circuited the solar panels. Another issue we faced was not having somewhere to test the smoke devices, but we did try a few times at the park on campus. While testing the solar panels with the Arduino the sprinklers came on and that scared me because of all the open electronics we had but we managed to keep everything dry. Since we never got to implementing the MPPT, I was not able to use the Raspberry Pi to give us the real time comparison of what the power output looked like with our build compared to without.

Successes

The main objectives I had to complete were the IR sensor, Buck Boost Converter, Voltage Regulation, Current reading, and the Maximum Power Point Tracking. I completed the IR sensor by calibrating it correctly to the point where it can detect the ash on the surface of the solar panel and then proceed to send data to the Air Compressor to turn on in the form of 5V. I was able to accomplish sending data also to the Buck Boost converter in order to jump whatever voltage input to our desired output voltage. I also got the voltage dividers to work in order to step down the voltages to an input that the Arduino could read and then calibrate it back to the original voltage with an error rate of .17%. I was able to read the current as well with the Arduino by calibrating the sensitivity to where the Arduino could read up to 20A. With the voltage and current I was able to find the power of the solar panel and store it so it can be compared with the previous power in order for the Maximum Power Point Tracking to work theoretically. I ended up writing the code for the Arduino for the P&O method but was never able to implement it but it worked on MATLAB so I see it as a success.

Data Analysis

According to our research, the efficiency of the solar panel output is better with the IR sensor, Buck Boost Converter, air compressor, and Maximum Power Point Tracking that we built than just the solar panel itself. The efficiency should increase by a total of 6% in the best-case scenario but it should always hover in the positive for efficiency compared to the solar panel itself. Using the MPPT in combination with the buck/boost converter and the air compressor the solar panel should be ready for any type of weather, even when it is smokey outside as the ash that resides on the panel will be blown away.

Conclusion

The project itself was very challenging and safety was always our highest priority as we were using a lot of wires and electrical components with the factor of the solar panel being able to produce 100W. Wires and resistors were constantly charred, and a lot of smoke was produced not from the smoke generating device but by our wires not being able to handle the current at times. I completed all my tasks except implementing the Maximum Power Point Tracking due to the extremely low current and failure in transistors working as a switch. This project really did combine everything I have learned throughout my education in Computer Engineering and I want to fix the transistor issue as that is what is holding the voltage to act as a switch. Better solar panel efficiency seems to be the future and I'm glad I got to work on something that resembles my stance for a cleaner future.

ENGINEERING STANDARDS (Anthony)

IEEE 1526-2003 - IEEE Recommended Practice for Testing the Performance of Stand-Alone Photovoltaic Systems

IEEE 1661-2007 - IEEE Guide for Test and Evaluation of Lead-Acid Batteries Used in Photovoltaic (PV) Hybrid Power Systems

IEEE 1561-2019 - IEEE Guide for Optimizing the Performance and Life of Lead-Acid Batteries in Remote Hybrid Power Systems

IEEE 1013-2019 - IEEE Recommended Practice for Sizing Lead-Acid Batteries for Stand-Alone Photovoltaic (PV) Systems

IEEE 1547 - Standard for Interconnecting Distributed Resources with Electric Power Systems

REFERENCES

[1] Behr, Peter. "Solar power plunges as smoke shrouds Calif." *E&E News*, 11 Sept. 2020, https://www.eenews.net/stories/1063713459

[2] "California ISO - Today's Outlook - Supply." *CAISO*, 10 Dec. 2020, http://www.caiso.com/TodaysOutlook/Pages/supply.aspx

[3] Potter, Allison. "How much does smoke haze affect rooftop solar production?" *Solar Analytics*, 13 Jan. 2020, https://www.solaranalytics.com/au/blog/how-much-does-smoke-haze-affect-rooftop-solar-production

[4] "Solar Panel Design and Firefighter Safety." *Solarponics*, 21 Feb. 2017, https://www.solarponics.com/solar-electric/firefighter-safety/#:~:text=The%20ash%20sediment%20can%20reduce,clean%20with%20a%20soft%20sponge.

[5] York, S. (2020, September 30). Smoke from California wildfires decreases solar generation in CAISO. U.S. Energy Information Administration. https://www.eia.gov/todayinenergy/detail.php?id=45336

[6] Sommer, L. (2018, September 10). Why 100 PERCENT clean energy in California is gonna be tricky. KQED. https://www.kqed.org/science/1930972/why-100-percent-clean-energy-in-california-is-gonna-be-tricky

[7] O'Brien, G., & Hope A. (2010). Localism and energy: Negotiating approaches to embedding resilience in energy systems. Energy Policy, 38(12), 7550-7558. https://doi.org/10.1016/j.enpol.2010.03.033

[8] Zhou, J., Tsianikas, S., Birnie, D.P., & Coit, D.W. (2019). Economic and resilience benefit analysis of incorporating battery storage to photovoltaic array generation. Renewable Energy, 135, 652-662. https://doi.org/10.1016/j.renene.2018.12.013

[9] Alghamdi, A. S., Bahaj, A.S., Blunden, L.S. Wu, Y. (2019, August 1). Dust Removal from Solar PV Modules by Automated Cleaning Systems. Energies (Basel), 12(15), 2923. https://doi.org/10.3390/en12152923

[10] California Department of Forestry and Fire Protection (CAL FIRE). (n.d.). 2020 incident archive. https://www.fire.ca.gov/incidents/2020/#:~:text=As%20of%20the%20end%20of,gigafire%22%20as%20the%20area%20burned

[11] Neamen, D. A. (2010). *Microelectronics: Circuit Analysis and Design - Fourth Edition*. McGraw-Hill.

[12] (n.a.). (2020, December 29). Module 3.3: Buck-Boost Converters. *Learn about electronics.* Eric Coates MA BSc. https://learnabout-electronics.org/PSU/psu33.php

[13] Project Sunroof data explorer: a description of methodology and inputs. May. 2017, https://www.google.com/get/sunroof/data-explorer/data-explorer-methodology.pdf

[14] Hauke, Brigitte.(2014, January). Basic Calculation of a Boost Converter's Power Stage. Texas Instruments.
https://www.ti.com/lit/an/slva372c/slva372c.pdf?ts=1618153440401&ref_url=https%253A%252F%252Fwww.google.com%252F

[15] Kamran, Muhammad, et al. "Implementation of Improved Perturb & Observe MPPT Technique with Confined Search Space for Standalone Photovoltaic System." *Journal of King Saud University - Engineering Sciences*, Elsevier, 18 June 2018, www.sciencedirect.com/science/article/pii/S101836391730380X.

APPENDICES
APPENDIX A
Components Used

Solar Panel
- Renogy 100 Watt 12 Volt Monocrystalline Solar Panel, 42.2 x 19.6 x 1.38 in.

Tank
- REPTI ZOO 10 Gallon Reptile Tank Glass Natural Cages Terrarium 20 x 12 x 10 in.
- Particle Board 20 x 10 x ⅜ in.

Air Compressor
- Husky 12 Volt Inflator
- Everbilt Polyethylene tubing ⅜ in O.D./ ¼ in I.D. x 10 ft.
- John Guest ⅜ in. Union Tee & 2X ⅜ in. End Cap

Battery
- Beiter DC Power 12V 5Ah Hi-Capacity SLA Battery

Smoke
- Air-Loc Smoke Generating Device 60 seconds

Light
- Advanced Lighting Systems 6000lm heavy duty LED spot light

Power Supply
- Siglent DC power supply

Optical Sensor
- TCRT 5000 Optical Sensor

Microcontrollers/Microprocessors
- Raspberry Pi 4
- Arduino Uno R3

Ash Substitute
- Diatomaceous Earth

Circuit Design
- Makeronics breadboard
- Wires (breadboard jumper, 16 AWG stranded, 14 AWG solid)
- Alligator clips
- Resistors
- IRFZ44N MOSFETs

APPENDIX B
Solar Module Datasheet

RNG-100D-SS

100W Monocrystalline Solar Panel

Electrical Data

Maximum Power at STC*	100 W
Optimum Operating Voltage (V_{mp})	17.9 V
Optimum Operating Current (I_{mp})	5.72 A
Open Circuit Voltage (V_{oc})	21.6 V
Short Circuit Current (I_{sc})	6.24 A
Cell Efficiency	21.0%
Maximum System Voltage	600 VDC UL
Maximum Series Fuse Rating	15 A

Mechanical Data

Solar Cell Type	Monocrystalline (6.1 x 4 in)
Number of Cells	33 (3 x 11)
Dimensions	42.2 x 19.6 x 1.38in (1074 x 498 x 35mm)
Weight	14.3 lbs (6.5 kg)
Front Glass	Tempered Glass 0.13 in (3.2 mm)
Frame	Anodized Aluminium Alloy
Connectors	Solar Connectors
Fire Rating	Class C

Thermal Characteristics

Operating Module Temperature	-40°F to 176°F
Nominal Operating Cell Temerature (NOCT)	47±2°C
Temperature Coefficient of Pmax	-0.23%/°C
Temperature Coefficient of Voc	-0.33%/°C
Temperature Coefficient of Isc	0.05%/°C

Solar Connectors

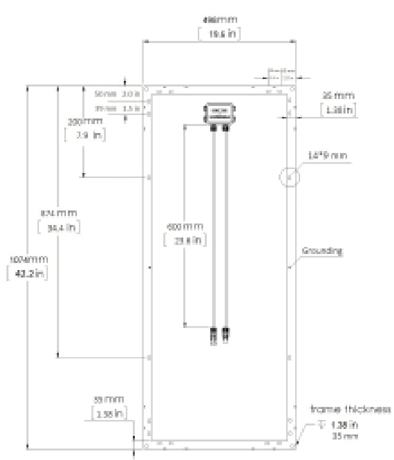
Rated Current	30A
Maximum Voltage	1000VDC
Maximum AWG Size Range	10 AWG
Temperature Range	-40°F to 194°F
IP Rating	IP 67

Junction Box

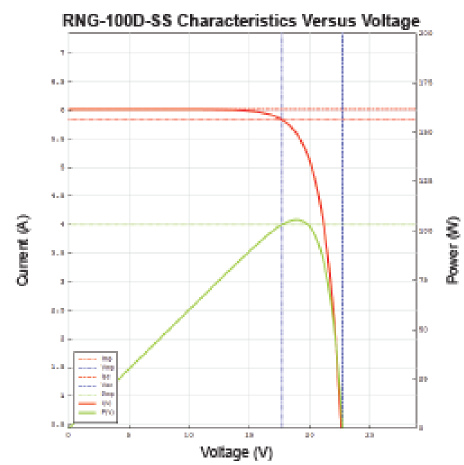
IP Rating	IP 65
Diode Type	HY 10SQ050
Number of Diodes	2 Diode(s)
Output Cables	14 AWG (2.00 ft long)

Certifications

Module Diagram



IV-Curve



*All specifications and data described in this data sheet are tested under Standard Test Conditions (STC - Irradiance: 1000W/m², Temperature: 25°C, Air Mass: 1.5) and may deviate marginally from actual values. Renogy and any of its affiliates has reserved the right to make any modifications to the information on this data sheet without notice. It is our goal to supply our customers with the most recent information regarding our products. These data sheets can be found in the downloads section of our website, www.renogy.com

APPENDIX C
Transistor Datasheet

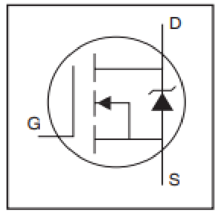
International
IOR Rectifier

PD - 94787B

IRFZ44NPbF

HEXFET® Power MOSFET

- Advanced Process Technology
- Ultra Low On-Resistance
- Dynamic dv/dt Rating
- 175°C Operating Temperature
- Fast Switching
- Fully Avalanche Rated
- Lead-Free



$V_{DSS} = 55V$

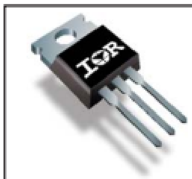
$R_{DS(on)} = 17.5m\Omega$

$I_D = 49A$

Description

Advanced HEXFET® Power MOSFETs from International Rectifier utilize advanced processing techniques to achieve extremely low on-resistance per silicon area. This benefit, combined with the fast switching speed and ruggedized device design that HEXFET power MOSFETs are well known for, provides the designer with an extremely efficient and reliable device for use in a wide variety of applications.

The TO-220 package is universally preferred for all commercial-industrial applications at power dissipation levels to approximately 50 watts. The low thermal resistance and low package cost of the TO-220 contribute to its wide acceptance throughout the industry.



TO-220AB

Absolute Maximum Ratings

	Parameter	Max.	Units
I_D @ $T_C = 25°C$	Continuous Drain Current, V_{GS} @ 10V	49	A
I_D @ $T_C = 100°C$	Continuous Drain Current, V_{GS} @ 10V	35	
I_{DM}	Pulsed Drain Current ①	160	
P_D @$T_C = 25°C$	Power Dissipation	94	W
	Linear Derating Factor	0.63	W/°C
V_{GS}	Gate-to-Source Voltage	± 20	V
I_{AR}	Avalanche Current①	25	A
E_{AR}	Repetitive Avalanche Energy①	9.4	mJ
dv/dt	Peak Diode Recovery dv/dt ③	5.0	V/ns
T_J T_{STG}	Operating Junction and Storage Temperature Range	-55 to + 175	°C
	Soldering Temperature, for 10 seconds	300 (1.6mm from case)	
	Mounting torque, 6-32 or M3 srew	10 lbf•in (1.1N•m)	

Thermal Resistance

	Parameter	Typ.	Max.	Units
$R_{\theta JC}$	Junction-to-Case	—	1.5	°C/W
$R_{\theta CS}$	Case-to-Sink, Flat, Greased Surface	0.50	—	
$R_{\theta JA}$	Junction-to-Ambient	—	62	

09/21/10

APPENDIX D
Arduino UNO R3 Schematic