Lab 7 Final Design Project

Reference Resource (*optional and available in the GWC lab*):
Practical Electronics for Inventors, Paul Scherz and Simon Monk, 2016, ISBN 978-1-25-958754-2

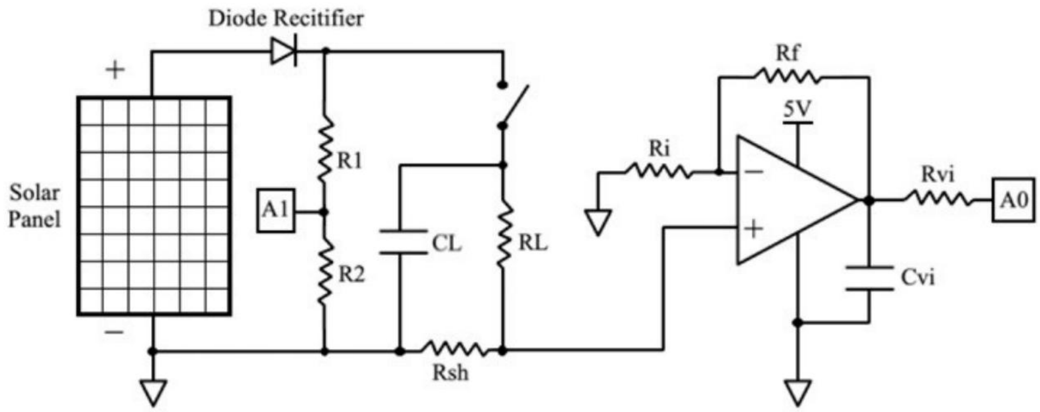
Design Project Overview:
Sparky Solar is a major designer of solar panel systems. Users and installers of Sparky Solar panels can take field measurements of the solar panel voltages and currents, but being the innovative company that it is, Sparky Solar CEO, Melanie Waters, wants to build a more automated tool that will provide a digital readout of the voltage, current, and power of the solar panels. This automated system will be a much better experience for the end-user and will prevent user calculation errors when determining the power output of the panels. The goal of the design is to create something portable, easy to use, and accurate for the end-users and something that shows the innovative nature of Sparky Solar.

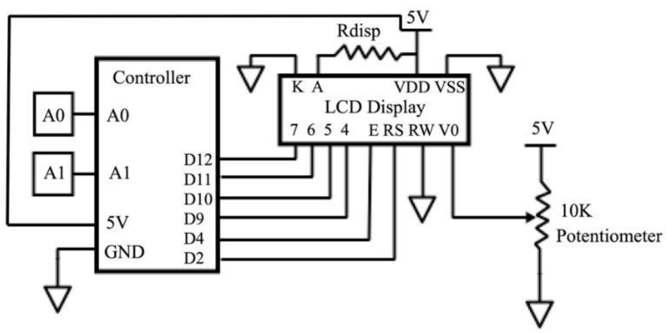
Sparky Solar has hired your team of circuit specialists to design and implement the automated solar panel power meter. The power meter must be portable and must output the result through an LCD digital display. For the prototype of the design, only single solar panels with a maximum output voltage of ~9V will be tested. All components needed for this design will be provided as indicated in the next table. The evaluation of the final prototypes will include technical competency, proof of understanding, and value added for the stakeholders. The engineering team will demonstrate the working solar power meter to the Sparky Solar Representative (grader) via screenshots, photos, their measurements and (optionally) a video-recorded demonstration (provide a link to your video in your Data Sheet).

Inventory List:
The following is a list of available inventory for this project:

Part	Details
Resistors	10Ω (10W), 47Ω, 100Ω, 220Ω, 330Ω, 510Ω, 1kΩ, 1.5kΩ, 2.2kΩ, 3.3kΩ, 4.7kΩ, 10kΩ, 22kΩ, 47kΩ,100kΩ, 220kΩ470kΩ,1MΩ,2MΩ
Capacitors	0.1μF, 1μF, 10μF, 22μF, 47μF, 100μF, 220μF
Diode Rectifiers	1N4001 (1N4148 in TinkerCAD and LTSPICE)
Op-Amp	Microchip MCP6022 or Analog Devices OP484
Alligator clips	Connectors
Assorted jumper wire	Red, black, white, green, blue, orange
Potentiometer	10K
Slide Switch	Breadboard-friendly SPDT Slide Switch (optional)
Solar Panels	6V, 167mA Solar Panel, Mini Polycrystalline (or a 9V battery)
DC Supply	Analog Discovery 2/3 Kit
Breadboard	830 Tie Points
Digital Multimeter	Portable Multimeter MS8217, DT-830B, DT838, or similar
Oscilloscope	Analog Discovery 2/3 Kit
UNO Controller Board	Elegoo or Arduino
USB Cable	Controller board to USB
LCD Display	LCD1602 Module with Pin Header
LEDS, buttons, buzzers, photoresistors	Assorted

Lab Instructions:
Sparky Solar has provided a basic schematic for the design as shown below:





As the design team, you are permitted to adjust this design as you require, but you must meet the basic requirements of Sparky Solar. This solar panel power meter must be designed to do the following:

1. Step down the voltage from the solar panel to a voltage value that can be an input into the controller (A1 = 5V maximum) for measurement. *(Lab 2 Voltage Dividers)*

2. Sample of the load current through a shunt resistor (Rsh), measure the voltage across the shunt resistor using the non-inverting Op-Amp, multiply the value of that voltage to a value that can be input into the controller (0V < Op-Amp output < 5V → Op-Amp Gain ~ 10-100, keep in mind that this gain range is merely a guideline and you are allowed to use other values as you deem fit), convert that voltage back into a current through Rvi and input into the controller (A0). (Lab 2 Resistor Networks and Lab 5 Op-Amps)

3. Use the Controller to measure and calculate the voltage, current, and power supplied to the load resistor (RL) from the solar panel. *(Lab 4 Arduino)*

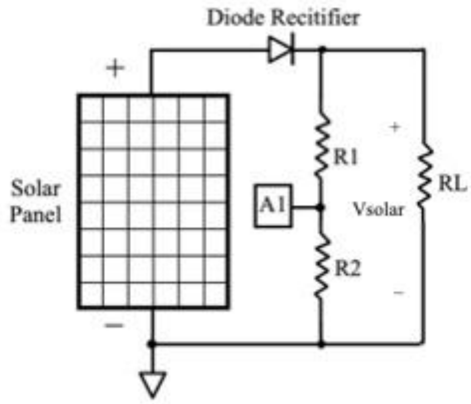
4. Output the voltage, current, and power values to the LCD display (or serial port). *(Lab 4 Arduino)*

5. Select a load capacitor (CL) so that the discharge time of the load (with a supply of ~9V and ~50mA) is about 0.1 second. The single pole dual throw switch (SPDT) is used to disconnect the load for discharging. *(Lab 6 Frequency Domain and Bode Plots)*

All resistors selected for this design will be single resistors, not resistor networks. The following instructions are the detailed procedures of the summarized five steps above.

The following are detailed procedure of the summarized five steps above.

Step 1: Design the Voltage Divider



Using your understanding of **voltage dividers**, design and test a circuit that will step down the voltage from the solar panel. The maximum input voltage for the Controller pin (A1) is 5V, and your design must work for any light condition. You can uses a 9V DC supply (battery) or a solar panel and high power lamp as a light source in the lab for testing, or you can take this outside to test with the actual solar panel. This replacement is, in general, a standard procedure in the design process especially when dealing with solar panels since they are highly susceptible to light condition changes, something that you do not want to encounter while testing your circuit.

Data Sheet Part 1

Answer all the questions in the data sheet.

A diode is a device that allows current to only flow in one direction *(be sure to install the diode in the correct direction)*. What do you think the purpose of the diode rectifier is in this design*? (In TinkerCAD or LTSpice, you can use the 1N4148 diode if the 1N4007 is not available).*

Connect the load resistor (RL = 1 kΩ) to your solar panel and **measure the voltage** the solar panel can produce over different light conditions using a multimeter. While you're outside, be sure to also **measure the current** the solar panel can produce over different light conditions. *Reminder – voltage is measured with the multimeter in parallel with the resistor and current is measured with the multimeter in series with the resistor (refer to Lab 2 for how to measure voltage and current)*. **Collect the voltage and current measurements for the listed light conditions on the datasheet.**

The CEO of Sparky Solar, Melanie Waters has told you that the design criteria for the voltage divider (in order of importance) are:

1. **Availability, 2. Lower cost, 3. Accuracy.**

When designing the voltage divider circuit, select which group of resistors you will use:

Resistors	Cost	Accuracy	Availability
Set 1 (1Ω-100Ω)	$1.00/each	+/- 1%	2-week delay
Set 2 (1kΩ - 100kΩ)	$2.00/each	+/- 2%	Available now
Set 3 (1MΩ - 100MΩ)	$3.00/each	+/- 5%	Available now

From your selected resistor set and inventory, **select R1 and R2** to ensure that the value of A1 does not exceed 5V for <u>all</u> light conditions. Assume a maximum solar panel voltage of ~9V, **what is the voltage into pin A1?**

Which light condition should be considered when determining the values or R1 and R2 and why?

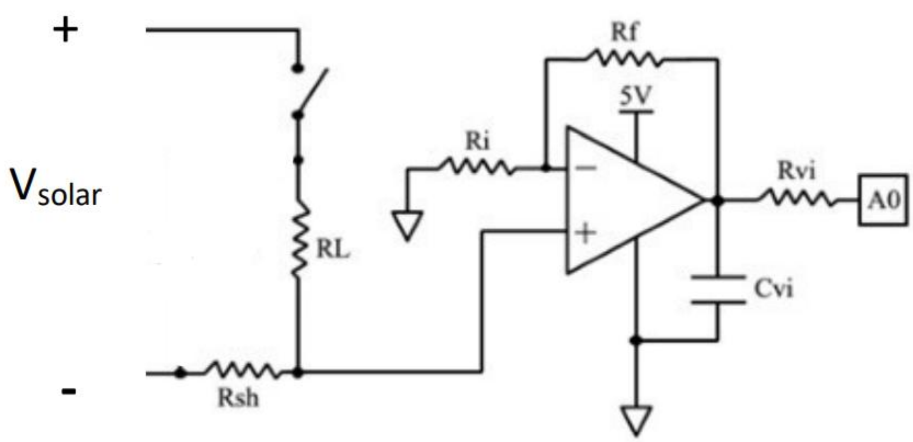
Your resistor divider is designed such that the maximum voltage on pin A1 is 5V (*a requirement of the Arduino*). Determine the **maximum voltage** across each resistor to stay below the **1/4W power rating** for each. *If the voltage exceeds 5V across either, you will need to reselect your R1 and R2 values*. In engineering design, it is important to know the limits of your design. This step is meant to determine how robust your design is and to make sure that the resistors you selected can handle the amount of energy that could possibly dissipate through them.

Create a **simulation of your circuit in LTSPICE or TinkerCAD** and include a **screenshot with the data sheet**. You can replicate the Solar Panel with a DC supply in simulation. *Be sure to set you supply current high enough in the simulation to maintain the solar panel voltage.*

Physically build and test your circuit. You can use a DC supply (battery) or solar panel and light source in the lab to replicate the solar panel outside for troubleshooting and testing your design.

Do your measured results from your built circuit match your simulated results? How much error are you seeing? What are some possible reasons for the difference in the real and theoretical results?

<u>Step 2: Design the Op-Amp Circuit</u>



Using your understanding of **Op-Amp circuits**, design and test a circuit that will measure the current across the load resistor (RL). For this design, you can make the following assumptions:

- The op-amp is ideal ($V^+ = V^-$ and $I^+ = I^- = 0A$)
- The load resistance (RL) is **1 kΩ**
- This circuit is in steady state (DC supply) so for your design you can assume the capacitor looks like an <u>open</u> for this part of the design
- To convert the load current into an input voltage for the Op-Amp, a shunt resistor (Rsh) of **5Ω** is used
- The output voltage of the Op-Amp will be the input for the Controller calculation (A0)
- Read and understand the data sheet and pin diagram for the Op-Amp you are using (*see Lab 5*)
- Although other values for the resistor (Rvi) and capacitor (Cvi) would work, use an **Rvi of 1 kOhm** and a **Cvi of 1 µF.**

Data Sheet Part 2

Answer all the questions in the data sheet.

a) Determine the **maximum current** across your load resistor (RL). The resistors provided are ¼ W resistors – will your resistors exceed this power rating (*compare with the currents from the solar panels in Part 1*)? *(The shunt resistor is small enough that you can omit it from your calculation).*

b) A small shunt resistor is used to sample the load current and convert it into a voltage for the op-amp. The shunt resistor must be very small so that almost no energy intended for the load resistor is wasted in measuring. The op-amp uses that small voltage, amplifies it, and sends it to the Arduino. Why do you think the current measurement needs to be done using an **op-amp** rather than how you measured the voltage in Part 1? *(Think about the reasons why op-amps are used).*

Given this change in the circuit of Part 2 compared to Part 1, do your resistor values of R1, R2 and RL still hold? Do your power rating values of R1, R2 and RL still hold?
Hints:

1. If the added subcircuit draws current from the solar panel, then recalculations need to be made.
2. The Rsh resistor can be neglected since it is very small

c) Using the highest load current from Part b, determine the **voltage into the + terminal** of the op-amp (*note:* $V^+ = I_{load}R_{sh}$). *The shunt resistor is very small because you don't want to draw extra current from the solar panel supply to do the measurements. However, this means that the op-amp input voltage will also be very small.*

d) The op-amp used for this design works rail-to-rail with a 5V supply. This means that the range of the op-amp output voltage (V_{out}) is about 0V to 5V and anything outside of those values will be "cut-off". For the op-amp feedback resistor design, the CEO of Sparky Solar, Melanie Waters, has told you that the op-amp circuit needs to function at temperatures as high as 155˚C, should be available soon for testing, and a lower price point, if possible. The design criteria for the op-amp circuit (in order of importance) are:
1. Higher operating temp, 2. Lower cost, 3. Sooner availability.

When designing the op-amp circuit, select which group of resistors you will use for Ri and Rf to amplify the magnitude of the op-amp output voltage (*note: this is a non-inverting configuration*).

Resistors	Cost	Physical Size	Max Operating Temp	Availability
Set 1 (1Ω-100Ω)	$1.00/each	3mm x 1 mm	155˚C	2-week delay
Set 2 (1kΩ - 100kΩ)	$2.00/each	4mm x 3mm	155˚C	Available now
Set 3 (200kΩ - 10MΩ)	$2.00/each	2mm x 2mm	135˚C	Available now

e) As noted above, V+ will be small, so you need to increase the output voltage of the op-amp so that the possible output voltages for all light conditions are better distributed in the 0-5V range. You still need to ensure that the output value of the op-amp does not exceed its supply voltage (5V) for the highest possible V^+. Select **values for Ri and Rf that ensure the output voltage of the op-amp doesn't exceed the 5V limit** of the Arduino pin.

f) Why is a **rail-to-rail Op-Amp** required for this design? A rail-to-rail op-amp costs more than a standard op-amp. What value to Sparky Solar does this extra feature add to the design? Refer to *Lab 5: Op-Amps* to recall what a rail-to-rail op-amp is and why we use them. (Hint: *the voltage we sample from the shunt resistor is very small).*

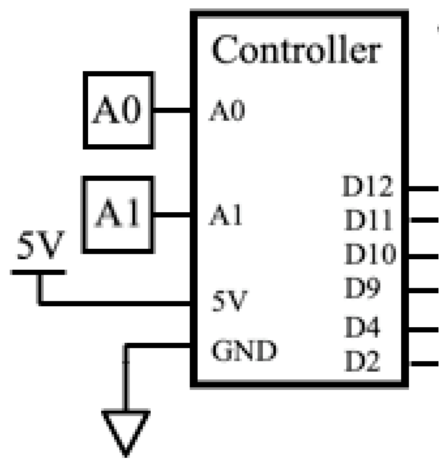
g) Do the values of **Ri and Rf need to be extremely accurate** for your **current measurement to be precise**? Explain why or why not.

h) **Create a simulation of your circuit in LTSPICE or TinkerCAD** and include a screenshot with the data sheet. *Be sure to set you supply current in your simulation high enough to maintain the solar panel voltage.*

i) **Physically build and test your circuit.** You can use a DC supply to replicate the solar panel for troubleshooting and testing your design. Do your measured results from your built circuit match your simulated results? How much error are you seeing? What are some possible reasons for the difference in the real and theoretical results?

Step 3: Design the Controller Circuit

For the intermediate building and testing steps of the controller circuit you will have a voltage supply (DC supply/9V Battery/solar panel) as well as the supply for your Arduino board (laptop/desktop USB). It is very important that you separate these two supplies - don't accidentally supply your Arduino with more than 9V. It is also **very, very important** that you **never power the Arduino board with a battery AND the USB** from your laptop – this could fry your USB port. Always have either the battery cable connected to the board OR the laptop USB, never both. Also make sure your solar panel/9V Battery/DC supply are never directly connected to the supply of your Controller board, especially when the USB cable is connected. **Keep all supplies isolated!!! Share the grounds, isolate the different supply voltages.**



Using your understanding of **Arduino circuits**, design and test a circuit that has two analog inputs for your current and voltage measurements (A0 and A1, respectively), a 5V output supply (that will go from the Arduino to supply the op-amp), and will display the measured current, voltage, and power from the input pins and display on the Serial Monitor.

The basic set-up for the Controller code can be found below (*note: the resistor values you have selected need to be added to the code*). You can also find the actual Lab 7 Arduino sketch folder and .ino file on canvas.

Note: Don't be afraid to play with the code and determine what each section is doing. Try different things and you may find more efficient ways to meet the objectives.

Additional Information regarding the Controller:

- The Controller is supplied with energy (voltage, current, power) from either the USB port or the 9V battery port, **not the 5V pin**
- The 5V pin is an **output pin** that can be used to supply other circuits (low current supply)
- Be sure to connect the Controller ground to your other grounds in the circuit to avoid floating nodes *(if you get odd voltage values, check your grounds*)
- The A0 and A1 pins are used in this design as **Analog inputs**. Analog means that the value you are inputting can range from 0-5V (*digital means the input is either 0 or 5V*). Since the Controller works

digitally, the analog input is converted to a digital input inside the Controller (A/D converter). Each input has 1024 digital bits available to “store” the analog value, so in the code below, when we calculate you’ll see a multiplier of 5/1024. We are simply dividing the 5V maximum value by 1024 bits to create a digital value that the Controller understands.
For example: if the analog voltage into pin A0 is 1V, the A/D converter will calculate that 1V out of a maximum value of 5V is the same thing as 205 out of 1024 bits, so the ADCV (analog to digital conversion for voltage) is 205.

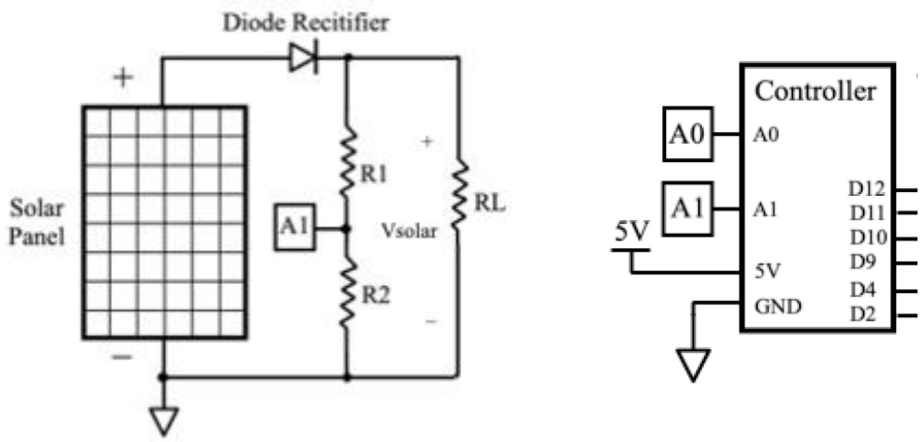
```
const long Rf = ; //enter value of feedback resistor in Ohms
const long Ri = ; //enter value of bias resistor in Ohms
const long R1 = ; //enter value of series resistor in Ohms
const long R2 = ; //enter value of load resistor in Ohms
const long Rsh = 50; //enter value of shunt resistor in Ohms
float Read_Vtemp = A1;//value read from ADC on pin A1
float Read_Itemp  = A0;//value read from ADC on pin A0
float Gain; //Gain of the Op-Amp Circuit
float Vtemp; //setting temporary analog voltage input
float Itemp; //setting temporary analog current input (actually a voltage reading)
float Voltage = 0.0; //initializing voltage reading
float Current = 0.0; //initializing current reading
float Power = 0.0;   //initializing power calculation

void setup()
{
  Serial.begin(9600); //set up serial monitor

}

void loop()
{

 Vtemp = analogRead(Read_Vtemp);
 Itemp = analogRead(Read_Itemp);

 // Voltage Measurement
 // 1. Converts the analog input (max 5V) into a digital value (max 1024)
 // 2. Accounts for the voltage divider calculation
 Voltage = Vtemp * (5.0/1024.0) * (R1+R2) / R2; //in Volts

 // Current Measurement
 // 1. Calculates the Op-Amp Gain
 Gain = 1+(Rf/Ri); //in V/V
 // 2. Converts the analog input (5V max) into a digital value (max 1024)
 // 3. Accounts for the op-amp gain, current to voltage conversion (Rsh), and units (mA=1/1000)
 Current = Itemp*(5.0/1024.0)/(Gain*Rsh)*1000; //in mA


 // Power Measurement
 Power = Voltage * Current;//Calculation of power from solar panel in mW

 // Print values to the serial monitor for verification
 Serial.print("ADCV reads: ");
 Serial.print(Vtemp);
 Serial.print("   Solar Panel Voltage is: ");
 Serial.print(Voltage);
 Serial.print(" Volts");
 Serial.println("");
 Serial.println("  ");
 delay(1000);

 Serial.print("ADCI reads: ");
 Serial.print(Itemp);
 Serial.print("   Solar Panel Current is: ");
 Serial.print(Current);
 Serial.print(" milliAmps");
 Serial.println("");
 Serial.println("  ");
 delay(1000);

 Serial.print("Solar Panel Power is: ");
 Serial.print(Power);
 Serial.print(" milliWatts");
 Serial.println("  ");
 Serial.println("__________________");
 delay(1000);

}
```

Data Sheet Part 3

Answer all the questions in the data sheet.

1. [Optional] Create a simulation of your Controller circuit in TinkerCAD and include a screenshot with the data sheet.
2. **Physically build and test your Controller circuit.** You can use a DC supply to feed in values for the A0 and A1 inputs for troubleshooting and testing your design. Using the values provided for pins A0 and A1, verify your voltage, current, and power measurements on the Serial Monitor and record your results. Using your R1, R2, Ri, Rf values, calculate the expected output voltage, current and power results and then record your simulated results from the Serial Monitor. Hint: *the equations you need are in the Arduino Code that you've been provided on Canvas. However, you will need to provide the code with the resistance values you calculated for all of the resistors.*

3. . [Optional] Create a simulation of your Voltage Divider circuit (from Part 1) and Controller circuit together in TinkerCAD and include a screenshot on the data sheet (you can use a DC voltage supply in TinkerCAD to replicate the solar panel).



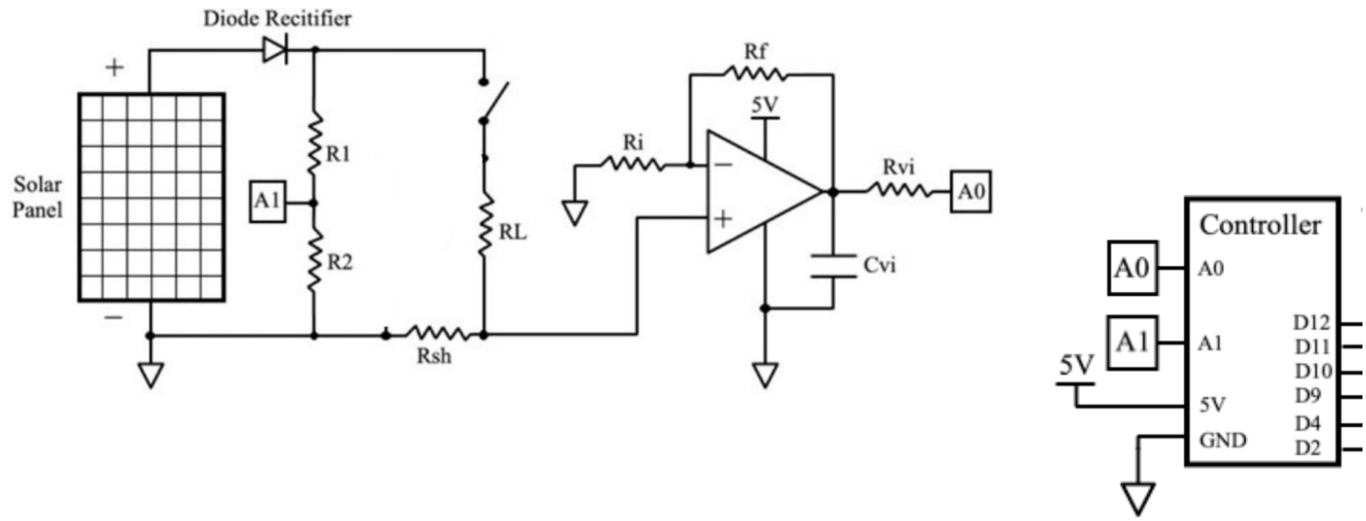
Set your solar panel voltage to the maximum expected voltage (from Part 1). Record the measured voltage from the Serial Monitor.

Do your **simulated** results from your TinkerCAD circuit match your **theoretical** results? How much error are you seeing? What are some possible reasons for the difference?

4. **Physically build and test your two-part subcircuit shown above (solar panel with voltage divider and controller).** Make sure to connect the voltage divider to the A1 pin on the Arduino. Do not forget to connect the grounds of both subcircuits together as well.

Do your measured results from your built circuit match your simulated results? How much error are you seeing? What are some possible reasons for the difference in the real and theoretical results?

5. [Optional] Create a **simulation of your Voltage Divider circuit (from Part 1) and Op-Amp circuit (from Part 2) and Controller circuit (Part 3) together in TinkerCAD** and include a screenshot on the data sheet (*you can use a DC voltage supply in TinkerCAD to replicate the solar panel*).



Set your solar panel voltage and current to the maximum expected voltage and current (from Parts 1 & 2). Record the measured **voltage**, **current**, and **power** from the Serial Monitor.

Do your **simulated results** from your TinkerCAD circuit match your **theoretical results**? How much error are you seeing? What are some possible reasons for the difference?

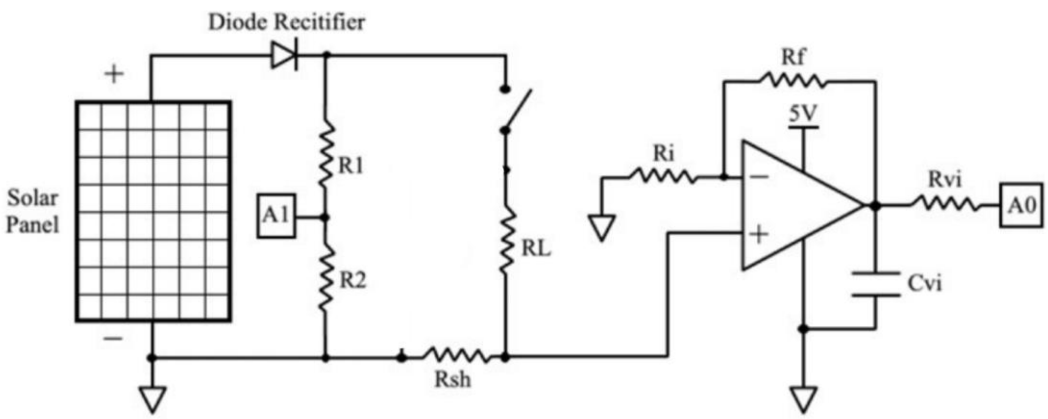
6. Physically build and test your three-part subcircuit (Voltage Divider circuit, Op-Amp and Controller). You can use a DC supply to replicate the solar panel for troubleshooting and testing your design. Do not forget to do the appropriate connections of A0 and all circuit grounds.

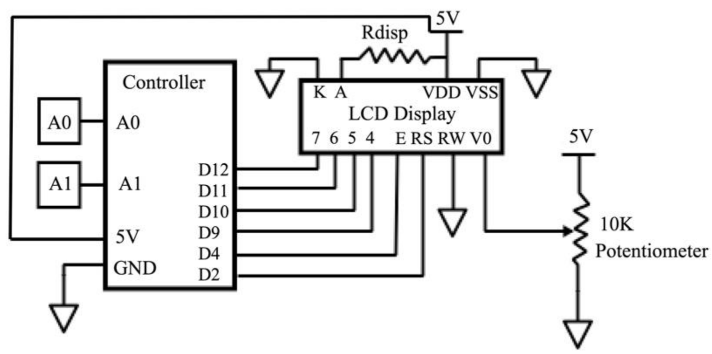
Do your **measured results** from your built circuit match your **simulated results**? How much error are you seeing? What are some possible reasons for the difference in the real and theoretical results?

What **adjustments** can you make to the code to **increase the accuracy** of the voltage, current, and power that display in the serial port?

Bonus Point Opportunity!
Considering the user of this power measuring device (solar installer or electronics enthusiast) and the application (portable, automated, innovative, user-friendly), add any additional automated features to enhance the experience of the user. Include your updated code and schematic attached to the data sheet. Bonus points will be given for successful changes in the code, and/or successful changes in the simulation, and/or successful changes in the physical prototype.

Step 4: Design the LCD Display circuit with your power measuring circuit.





Using your understanding of Arduino circuits and **LCD displays**, design and test the complete automated solar panel voltage, current, and power measurement circuit. Be sure to refer back to Lab 4 for details on the LCD display circuit set-up. For this design, set the display resistor (Rdisp) to **220 Ω**.

For your LCD display, be sure to add to the initialization of your code:

```
#include <LiquidCrystal.h>
const int rs = 2, en = 4, d4 = 9, d5 = 10, d6 = 11, d7 = 12; //set digital output pins for LCD display
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

Add to the set-up of your code:

```
lcd.begin(16, 2);  //set up LCD display

//write to LCD display to verify it's working
lcd.print("   Circuits I    ");
lcd.setCursor(0, 1);
lcd.print("    Wattmeter   ");
delay(2000);
lcd.clear();
```

Add to the main body of your code:

```
//Print values to LCD display
lcd.setCursor(0, 0);
lcd.print("V="); lcd.print(Voltage); //lcd.print("V");
lcd.print(" ");
lcd.print("P="); lcd.print(Power); lcd.print("mW ");
lcd.setCursor(0, 1);
lcd.print("I=");lcd.print(Current); lcd.print("mA");
lcd.print("    ");
delay(1000);
```

Bonus Point Opportunity – use the LCD for additional purposes that somehow benefit the user. Consider the user of this power measuring device (solar installer or electronics enthusiast) and the application (portable, automated, innovative, user-friendly). Include your updated code and schematic attached to the data sheet. Bonus points will be given for successful changes in the code, and/or successful changes in the simulation, and/or successful changes in the physical prototype.

Data Sheet Part 4

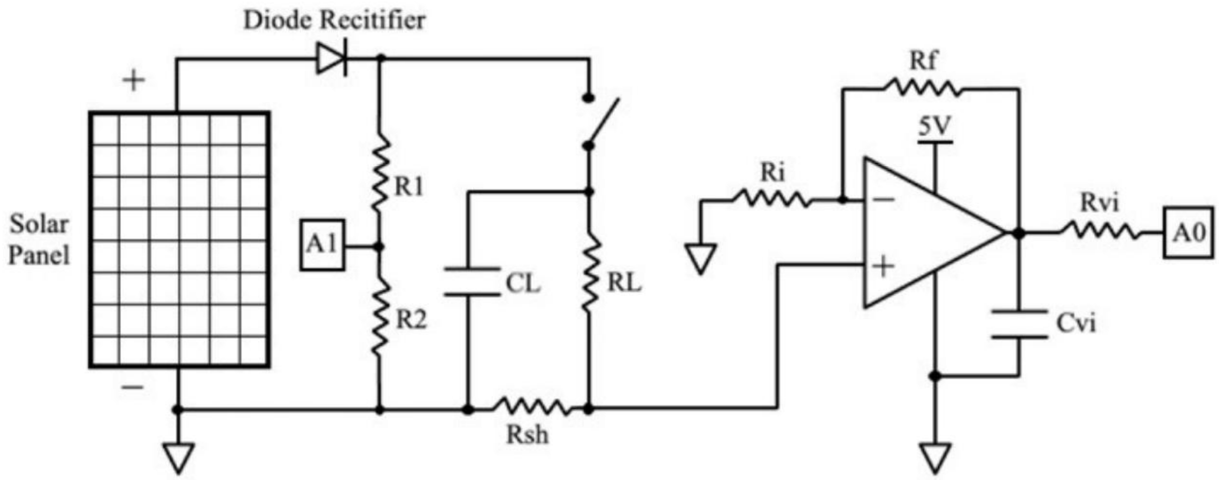
Answer all the questions in the data sheet.

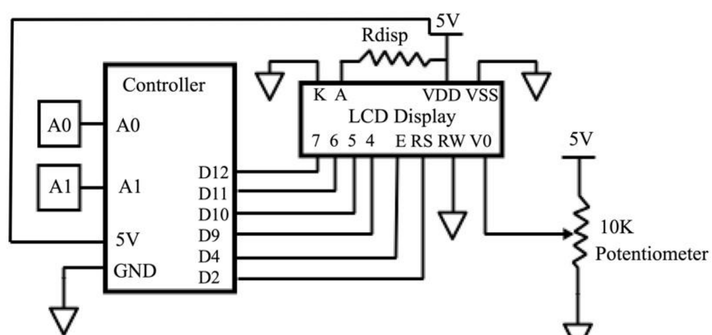
[Optional] Create a simulation of your Automated Solar Panel Power Meter circuit in TinkerCAD and include a screenshot with the data sheet. Test and troubleshoot your simulation (you can use a DC supply to represent the solar panel) over the given voltage supply values. If you need to change any component values, note the changes and the reasons for the changes. Hint: Give the LCD a moment to adjust after changing the input voltage.

Physically build and test your circuit. You can use a DC supply to replicate the solar panel for troubleshooting and testing your design. Take 4 different measurements using 4 different lighting conditions. While changing the lighting conditions, make sure to:

1. Use your analog discovery 2's voltmeter to read the solar panel's voltage
2. Connect your digital multimeter as an ammeter in series with the solar panel to measure its supplied current.

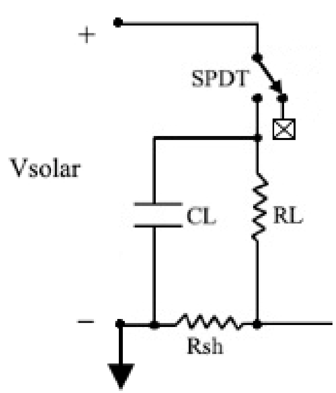
Step 5: Select a Load Capacitor to set the correct discharge time





In our design, we will connect a capacitor in parallel with the RL. The main benefit of this capacitor is to stabilize the voltage on the load resistor in case abrupt changes to the solar panel's voltage occur due to abrupt weather changes. We mimic these changes by a disconnection in the circuit through a switch. Our design should allow for a discharge time that is as high as possible. However, this comes at the expense of the capacitor's capacitance. Sparky Solar has done some measurements and found that a reasonable discharge time for your design is 0.1 seconds.

When operating the Solar Panel Power Meter, the SPDT switch will connect the load to the solar panel. When the SPDT switch is open, the load capacitor (CL) will discharge.



Using your understanding of **First Order Transient Circuits**, determine an appropriate capacitor value (from your available parts) to ensure the load will discharge to **1mV** in about **0.1sec** with a solar panel supply of **9V** and **50mA**. You will still assume the load resistance (RL) of **1 kΩ**. (*You can assume 1mV to be a fully discharged output*).

Data Sheet Part 5

Answer all the questions in the data sheet.

Using the First Order Differential method, First Order Step-by-Step method, or Laplace transforms, determine the size of the load capacitor to meet the design specifications noted above. Show your work on the data sheet.

Data Sheet Part 6

Answer all the questions in the data sheet.

Lab CLOS:

- Understand and apply the application of linear electrical circuits.
- Application of resistive networks, capacitors (in a First Order Transient circuit), and Op-Amps.
- Use SPICE and/or TinkerCAD to analyze a circuit and calculate expected values.
- Proficient in circuit simulation and measurement of electrical systems using a multimeter and an oscilloscope.
- Analysis and design using Ohm’s Law, Kirchhoff’s Laws and the Power Law.
- Understands the difference between calculated and measured results.
- Design of resistive networks and an Op-Amp closed loop circuit.
- Explores multiple solution paths.
- Applies technical skills/knowledge to the development of a technology/product.
- Focuses on understanding the value proposition of a discovery.
- Engages in actions with the understanding that they have potential to lead to both gains or losses.
- Persuades why a discovery adds value from multiple perspectives (technological, societal, financial, environmental, etc.).
- Identifies and works with individuals with complementary skill sets, expertise, etc.
- Integrates/synthesizes different kinds of knowledge.