

IV Swinger 2

Optional Environmental Sensors

Document Revision: 2.00 (26-Mar, 2019)

Chris Satterlee

The collage consists of five images arranged in a grid-like layout:

- Top Left:** A person wearing a "SunWork Renewable Energy Projects" t-shirt holds the IV Swinger 2 device, which is a clear acrylic enclosure containing electronic components and a blue cable.
- Top Right:** An IV Plot titled "IV Swinger Plot for 01/16/17@14:03:54". The plot shows Current (amps) on the y-axis (0 to 10) versus Voltage (volts) on the x-axis (0 to 40). It displays two curves: a red line for "Measured Points" and a blue line for the "Interpolated IV Curve". Annotations include $I_{sc} = 8.07 \text{ A}$, $V_{oc} = 36.31 \text{ V}$, and $P_{mp} = 211.54 \text{ W}$ (26.84×7.54).
- Bottom Left:** A screenshot of the "IV Swinger 2" software interface. It features a circular logo with a sun and a blue gradient background, and the text "IV Swinger 2" at the bottom. The interface includes buttons for "Preferences", "Results Wizard", "Axis Ranges", "Max V:", "Max I:", and "Lock". A central button labeled "Swing!" is highlighted in red. To the right are checkboxes for "Loop Mode", "Plot Power", "Rate Limit", and "Save Results".
- Bottom Right:** A photograph of the physical hardware, showing the clear acrylic case, the Arduino Uno microcontroller board, and various wires connected to it. A red alligator clip is attached to one of the wires.

Copyright (C) 2018, 2019 Chris Satterlee

IV Swinger and IV Swinger 2 are open source hardware and software projects.

Permission to use the hardware design is granted under the terms of the TAPR Open Hardware License Version 1.0 (May 25, 2007) - <http://www.tapr.org/OHL>

Permission to use the software is granted under the terms of the GNU General Public License v3 - <http://www.gnu.org/licenses>.

Current versions of the license files, documentation, hardware design files, and software can be found at:

https://github.com/csatt/IV_Swinger

Table of Contents

1	Introduction.....	4
2	DS18B20 Temperature Sensors	5
2.1	Connections	6
2.1.1	Connecting One DS18B20 to IV Swinger 2 (PermaProto-based).....	6
2.1.2	Connecting One DS18B20 to IV Swinger 2 (PCB-based).....	6
2.1.3	Connecting Multiple DS18B20s to IV Swinger 2.....	7
2.2	Software Installation	8
2.2.1	Installing Arduino Libraries	8
2.2.2	Modifying IV_Swinger2.ino	8
2.3	Swinging IV Curves with Temperature Sensors	9
2.3.1	Temperature Scale	9
2.3.2	Precision.....	9
2.3.3	Where Are the Temperatures Recorded?	9
3	Modified InstESRE Pyranometer (Irradiance Sensor)	11
3.1	Modifications to the InstESRE Pyranometer	11
3.2	Connections	11
3.3	Building the Modified InstESRE Pyranometer	12
3.4	Installing the 3.5mm Jack in the IV Swinger 2 Enclosure	12
3.5	Software Installation	13
3.5.1	Installing Arduino Library	13
3.5.2	Modifying IV_Swinger2.ino	13
3.6	Swinging IV Curves with a Pyranometer	14
3.6.1	Placement / Mounting of the Pyranometer.....	14
3.6.2	Irradiance Units.....	14
3.6.3	Precision.....	14
3.6.4	Where Is the Irradiance Recorded?	15
3.7	Calibrating the Pyranometer	15
3.7.1	Reference Pyranometer	15
3.7.2	Using a PV Module as a Pseudo Reference Pyranometer.....	16
3.7.2.1	Rated Isc	16
3.7.2.2	Temperature Coefficient of Isc	16
3.7.2.3	Calculating Irradiance from Measured Isc	16
3.7.3	Performing the Calibration	17

Table of Figures

Figure 2-1: DS18B20 Sensor.....	5
Figure 2-2: DS18B20 Waterproof Probe with Cable	5
Figure 2-3: Connections for a Single DS18B20.....	6
Figure 2-4: PCB connections.....	7
Figure 2-5: Connecting Additional DS18B20s	7
Figure 3-1: Pyranometer connections.....	12

1 Introduction

This document describes how to build and add optional environmental sensors to the IV Swinger 2 curve tracer.

The currently supported sensors are:

- DS18B20 temperature sensors
- Modified InstESRE pyranometer (irradiance sensor)

If support for additional sensors (inclinometer? compass?) is ever added, those will also be covered in this document. However, there are currently no plans for sensors other than the two listed above.

2 DS18B20 Temperature Sensors

The DS18B20 is an inexpensive temperature sensor that can be purchased for less than \$2 on Amazon, eBay, and other places. It comes in a TO-92-3 package that looks just like a discrete transistor as seen in Figure 2-1 below.

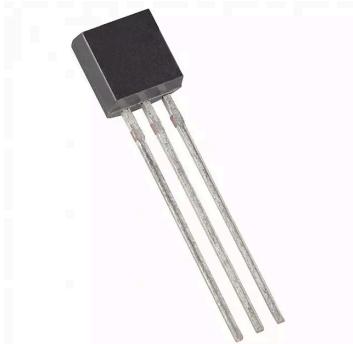


Figure 2-1: DS18B20 Sensor

It may also be purchased encapsulated in a waterproof enclosure with a 1-meter, 2-meter or 3-meter cable for not much more money than the device alone. These are abundant on Amazon and eBay. See Figure 2-2 below.



Figure 2-2: DS18B20 Waterproof Probe with Cable

In addition to being inexpensive and readily available, the DS18B20 has the following desirable characteristics:

- Only one Arduino signal pin is needed
- Multiple temperature sensors may be connected to the same one-wire interface
- Arduino library code is available
- Accuracy is more than adequate

2.1 Connections

2.1.1 Connecting One DS18B20 to IV Swinger 2 (PermaProto-based)

The connections that must be made in order for one DS18B20 to be detected and its temperature read by the IV Swinger 2 software are shown in Figure 2-3 below.

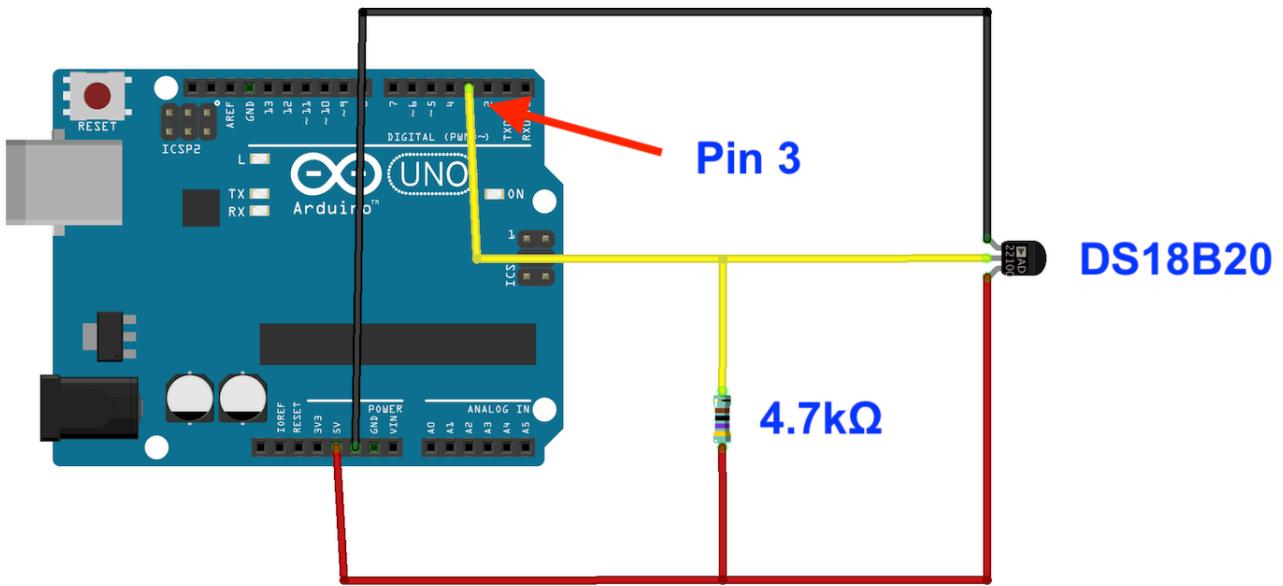


Figure 2-3: Connections for a Single DS18B20

The data (yellow) wire must connect directly to Pin 3 on the Arduino.

The power (red) wire can connect anywhere that is connected to the +5V pin on the Arduino, e.g. it can be soldered to the +5V rail on the PermaProto board. Optionally, it may be connected to the +3.3V pin on the Arduino.

The ground (black) wire can connect anywhere that is connected to the GND pin(s) on the Arduino, e.g. it can be soldered to the ground rail on the PermaProto board. Or it can connect directly to the unused GND pin on the Arduino itself.

The 4.7k Ω resistor must be connected between the data (yellow) wire and the power (red) wire. This resistor may be soldered to the PermaProto board, but that is not necessary. If you are adding the DS18B20 to an already-built IV Swinger 2, it will be easier to just solder this resistor to the two wires themselves – just make sure it is protected from shorting to anything (electrical tape, shrink tubing, etc.).

2.1.2 Connecting One DS18B20 to IV Swinger 2 (PCB-based)

If your IV Swinger 2 is built using a PCB instead of a PermaProto board, the PCB is plugged into the top of the Arduino. Electrically, the connections are the same as shown in Figure 2-3 above, but the data (yellow) wire is plugged into pin 5 of the A2 stacking connector on the PCB, and the power (red)

and ground (black) wires can be plugged into the female header, FH as shown in the figure below.

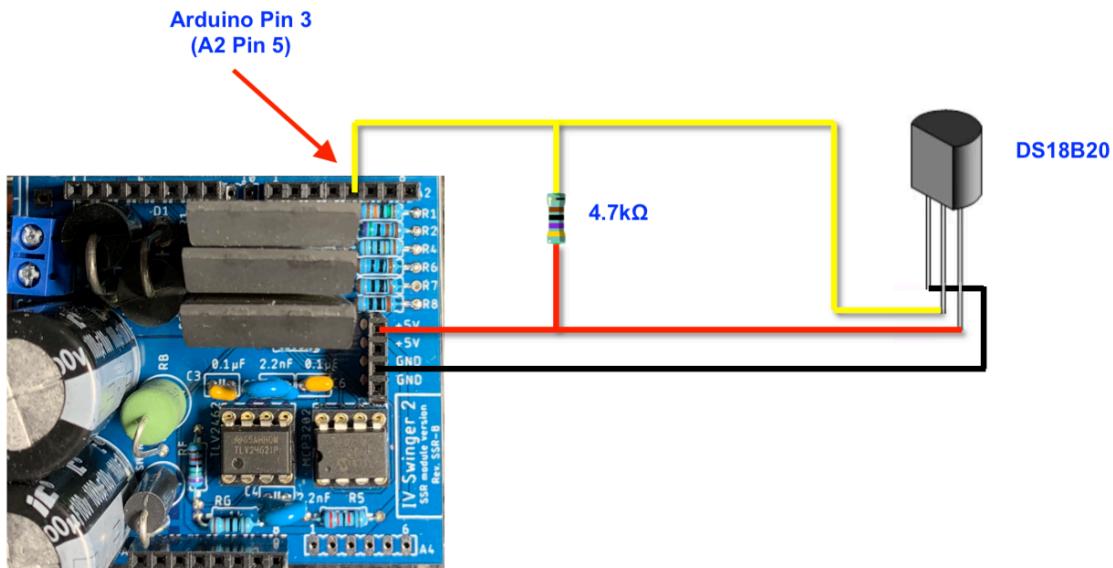


Figure 2-4: PCB connections

2.1.3 Connecting Multiple DS18B20s to IV Swinger 2

A nice feature of the DS18B20 is that each one has a unique ROM code, which allows multiple sensors to be connected to the same data wire. The software can identify which sensor is which, and read each of their temperatures. This could be useful if you want to measure the temperature at multiple places on the module, for example.

Figure 2-5 below shows the addition of a second DS18B20.

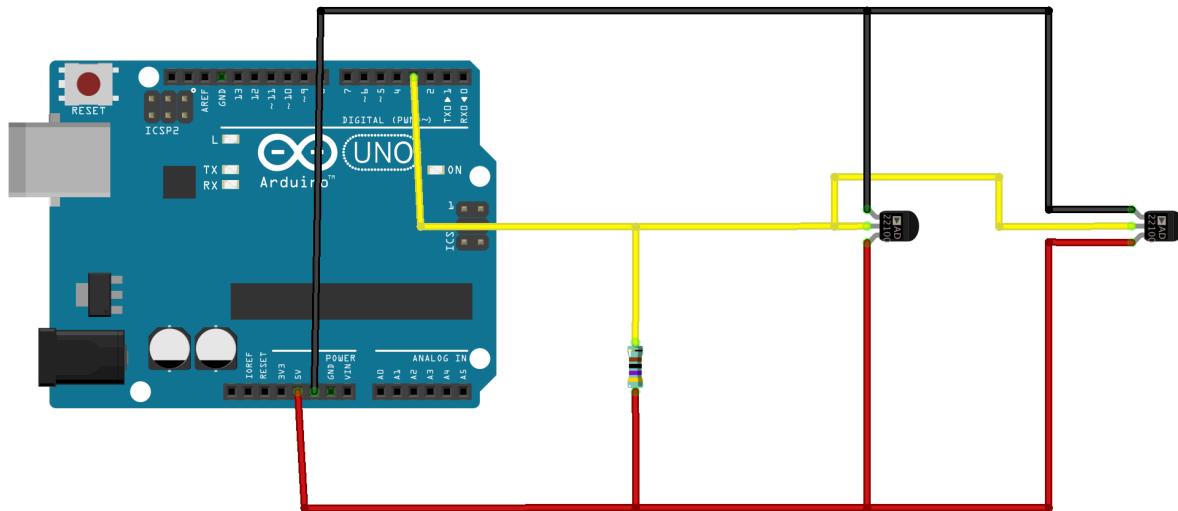


Figure 2-5: Connecting Additional DS18B20s

Note that there is only one 4.7kΩ resistor regardless of how many DS18B20s there are.

2.2 Software Installation

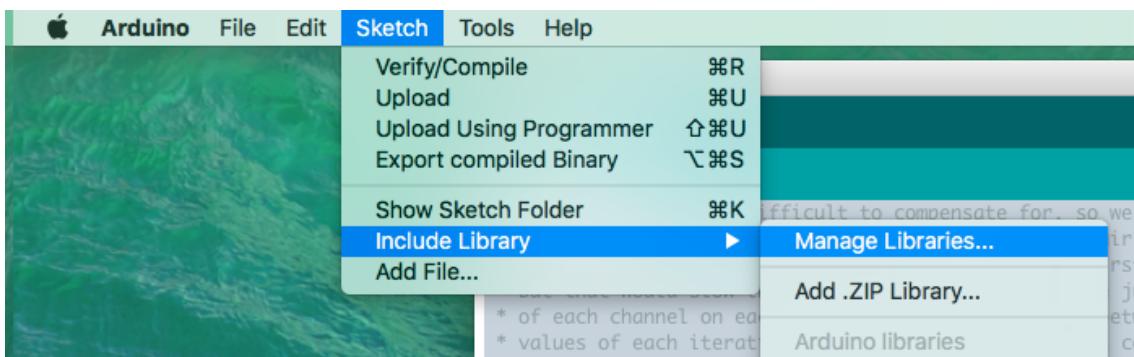
The latest releases of the IV Swinger 2 Arduino sketch and laptop application have support for the DS18B20. However, the Arduino sketch makes use of the OneWire and DallasTemperature libraries, which must be installed. Also, the IV_Swinger2.ino file must be edited to uncomment one line.

2.2.1 Installing Arduino Libraries

Two Arduino libraries must be installed:

- OneWire
- DallasTemperature

To do this from the Arduino IDE, use Sketch->Include Library->Manage Libraries...



Search for “OneWire”. Click on the one with the name “OneWire” and then click on the Install button.

Search for “DallasTemperature”. Click on the one with the name “DallasTemperature” and then click on the Install button.

2.2.2 Modifying IV_Swinger2.ino

Since temperature sensing is an optional feature, it was decided not to burden all users with having to install the OneWire and DallasTemperature libraries. The tradeoff is that users who do want to use this feature need to modify one line in the IV Swinger2 Arduino sketch.

You must manually change (“uncomment”) this line in IV_Swinger2.ino:

```
//#define DS18B20_SUPPORTED
```

to:

```
#define DS18B20_SUPPORTED
```

You may make this change using any text editor before starting up the Arduino IDE, or you may use the editor built into the Arduino IDE. If you have installed the OneWire and DallasTemperature libraries, the sketch should upload without errors.

Note that running code with this line uncommented will work fine even if there are no DS18B20s connected.

2.3 Swinging IV Curves with Temperature Sensors

When you have one or more DS18B20s connected as described in Section 2.1 above and you have uploaded IV_Swinger2.ino with `#define DS18B20_SUPPORTED` uncommented, you don't need to do anything else to get temperature readings.

2.3.1 Temperature Scale

Temperatures are reported in degrees Celsius. There is no option to report in degrees Fahrenheit.

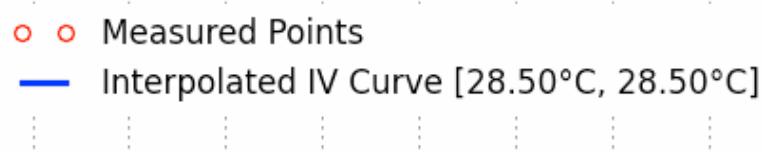
2.3.2 Precision

Temperatures are reported in **increments of 0.25°C**. The DS18B20 supports programmable precisions from 9 to 12 bits, and this is 10-bit precision. 12-bit precision readings take 3/4 second for each device, which is too slow. 10-bit readings take less than 1/4 second. This is currently hardcoded in the Arduino sketch, and there are no plans to make it configurable.

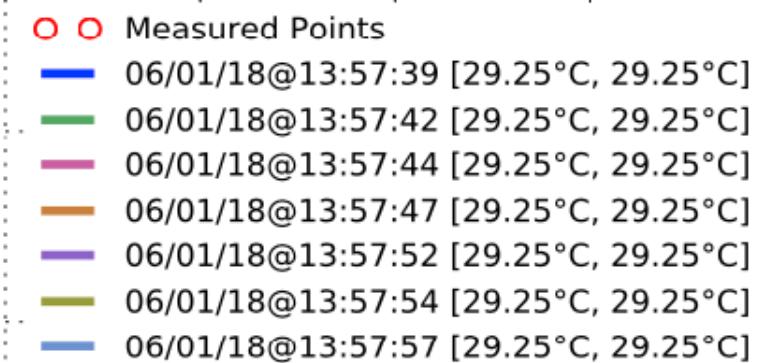
2.3.3 Where Are the Temperatures Recorded?

The laptop application records the temperature(s) in a “run info” file (which will also contain values from other environmental sensors if they exist). There is a run info file saved for each run and it is saved in the same directory/folder as the CSV, PDF and other files for the run.

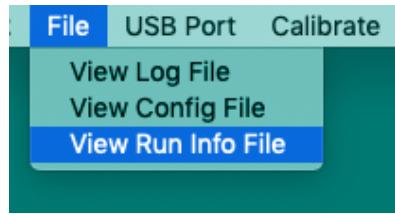
The temperature or temperatures are also included on the graph itself, in the legend. Here is an example with two temperature sensors:



Each curve on an overlay will have the temperature(s) for that run:



If you have only one sensor, that is all the information you need. But if you have more than one, you need to know which temperature is from which sensor. The File menu in the laptop application has an entry, “View Run Info File”:



The contents of the run info file might look like:

```
ROM code of DS18B20 temp sensor #1 is 0xEF000007F9CE1628
ROM code of DS18B20 temp sensor #2 is 0xBF000007F8D09328
Temperature at sensor #1 is 29.25 degrees Celsius
Temperature at sensor #2 is 29.25 degrees Celsius
```

The ROM codes uniquely identify each sensor, but that is unlikely to be of much help. Probably the best way to identify which sensor is which is to use freeze spray, an ice cube, cold drink, or something like that to artificially cool one sensor at a time to see which one it is. Fortunately, the sensors will always be in the same order, so you only have to go through this identification process once.

The temperatures recorded in the legend entries on the graph are listed in order [#1, #2, ...]

3 Modified InstESRE Pyranometer (Irradiance Sensor)

Irradiance is the power/area (W/m^2) value of the sunlight hitting the surface of the PV module or cell at a given point in time. “Full sun” is about 1000 W/m^2 . An instrument that measures irradiance is called a pyranometer. A “true” pyranometer is horrendously expensive. This one costs about \$2700:

<https://www.hukseflux.com/products/solar-radiation-sensors/pyranometers/sr30-d1-pyranometer>

A true pyranometer uses a thermopile sensor. It accurately measures the solar radiation across its full spectral range. It also has a directional response that is very close to the ideal cosine characteristic.

Fortunately, it is possible to build a “surrogate” pyranometer for less than the cost of the IV Swinger 2. Dr. David Brooks of the Institute for Earth Science Research and Education (InstESRE) has a kit that he sells for \$25 (which includes US delivery). His website explains how this DIY pyranometer differs from a true pyranometer:

<http://www.instesre.org/construction/pyranometer/pyranometer.htm>

An interesting observation is that the sensor in this pyranometer is a silicon photodiode, which is essentially just a minuscule solar cell. This tiny solar cell has its own IV characteristic. The irradiance is determined by measuring the current generated by the photodiode through a load resistor. Irradiance is directly proportional to the current, or the current is directly proportional to the irradiance, depending on which way you want to look at it. This, of course, is also true for the PV module or cell that is the device under test (DUT) for the IV Swinger 2. Ironically, the DUT is, or could be, a better pyranometer than the pyranometer! In fact, Section 3.7.2 describes how to use a PV module to calibrate the pyranometer once you have built it.

The photodiode pyranometer can be useful, however, to compare the IV curves of different PV modules or cells if it is calibrated to a reference pyranometer. Commercial IV curve tracers use a similar sensor for their irradiance sensors.

3.1 Modifications to the InstESRE Pyranometer

IV Swinger 2 supports the InstESRE pyranometer with the following modifications:

- Addition of an ADS1115 analog-to-digital converter (ADC)
- (Optional) addition of an internal TMP36 sensor for photodiode temperature compensation

3.2 Connections

Figure 3-1 below shows the electrical connections between all of the components of the modified InstESRE pyranometer and between the pyranometer and the IV Swinger 2. The figure shows a PCB-based IV Swinger 2.

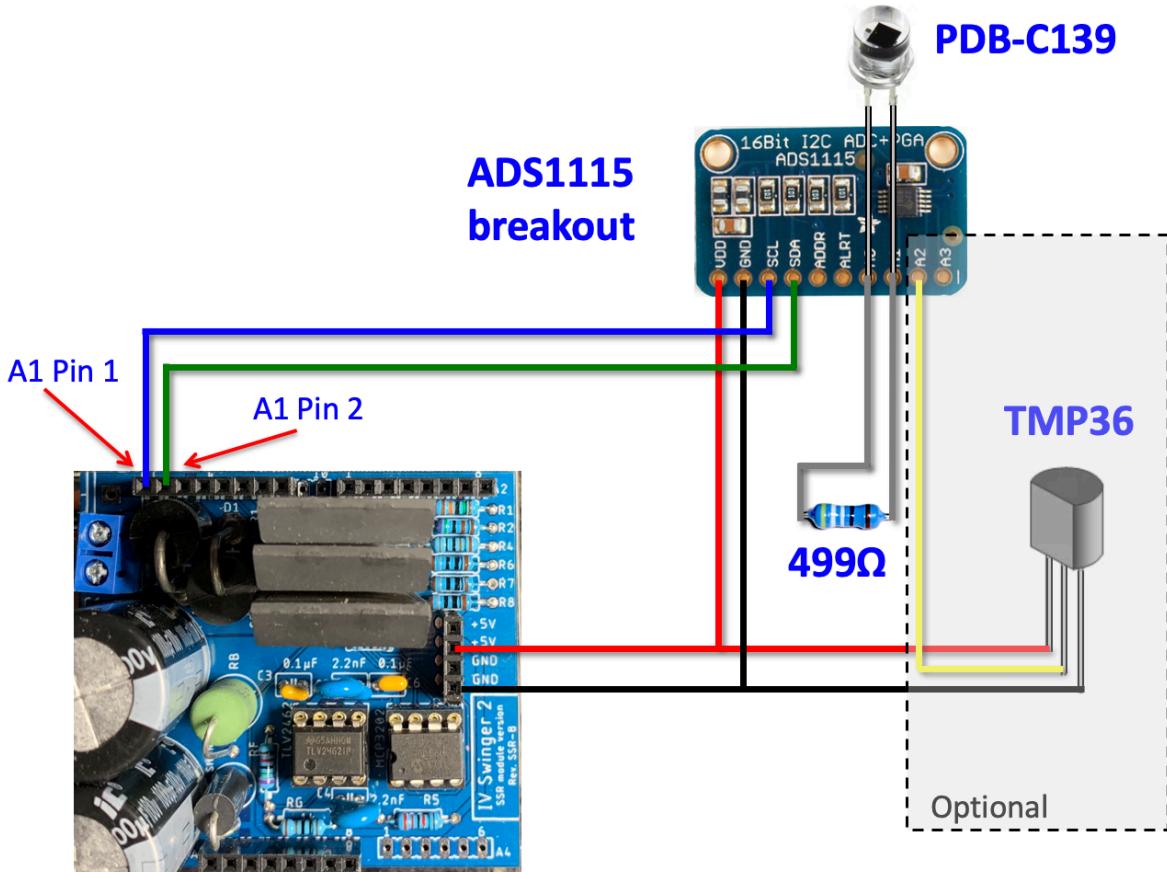


Figure 3-1: Pyranometer connections

3.3 Building the Modified InstESRE Pyranometer

Although the modifications were designed for use with IV Swinger 2, this version of the InstESRE pyranometer is also useful for other applications where interfacing with an Arduino and temperature compensation are desired. Therefore, the documentation on how to build one has been placed in an independent GitHub repository:

https://github.com/csatt/ADS1115_InstESRE_Pyranometer

There you will find a document and a standalone Arduino test sketch. The document contains additional descriptions of the hardware, instructions for how to order the kit and other parts, and a step-by-step description of the construction. There is also an Instructable, with photos of each step:

<https://www.instructables.com/id/ADS1115-InstESRE-Pyranometer>

Once you have built and tested the modified InstESRE pyranometer according to the Instructable and document, you may proceed with the following sections in this document.

3.4 Installing the 3.5mm Jack in the IV Swinger 2 Enclosure

The 4-conductor 3.5mm female panel mount jack that you soldered the red, black, blue, and green hookup wires to should be installed in the IV Swinger 2 enclosure by drilling the appropriate size hole and fastening it in place with the provided nut. The four hookup wires should be connected to the IV

Swinger 2 PCB as shown in Figure 2-4 on page 7. If your IV Swinger 2 is PermaProto-based, the red and black wires should be soldered to the power and ground rails of the PermaProto board and the blue and black wires connected to the Arduino SCL and SDA pins.

3.5 Software Installation

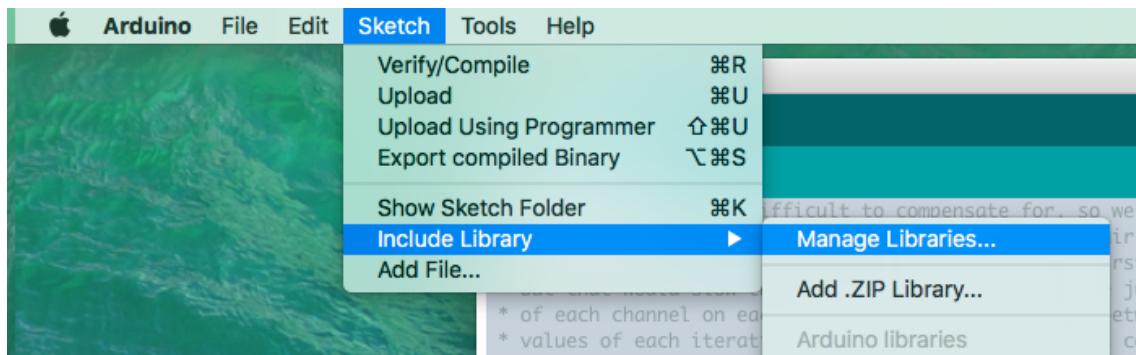
The latest releases of the IV Swinger 2 Arduino sketch and laptop application have support for the modified InstESRE pyranometer. However, the Arduino sketch makes use of the Wire and Adafruit ADS1X15 libraries. The Wire library is built in, but the Adafruit library must be installed. Also, the IV_Swinger2.ino file must be edited to uncomment one line.

3.5.1 Installing Arduino Library

You already did this if you built and tested your own pyranometer. Otherwise, the following Arduino library must be installed:

- Adafruit ADS1X15

From the Arduino IDE, use Sketch->Include Library->Manage Libraries...



Search for “Adafruit”. Click on the one with the name “Adafruit ADS1X15” and then click on the Install button.

3.5.2 Modifying IV_Swinger2.ino

Since the pyranometer is an optional feature, it was decided not to burden all users with having to install the Adafruit ADS1X15 library. The tradeoff is that users who do want to use this feature need to modify one line in the IV Swinger2 Arduino sketch.

You must manually change (“uncomment”) this line in IV_Swinger2.ino:

```
//#define ADS1115_PYRANOMETER_SUPPORTED
```

to:

```
#define ADS1115_PYRANOMETER_SUPPORTED
```

You may make this change using any text editor before starting up the Arduino IDE, or you may use the editor built into the Arduino IDE. If you have installed the Adafruit ADS_1X15 library, the sketch should upload without errors.

Note that running code with this line uncommented will work fine even if there is no pyranometer connected.

3.6 Swinging IV Curves with a Pyranometer

When you have the pyranometer connected as described in Section 3.2 above and you have uploaded IV_Swinger2.ino with `#define ADS1115_PYRANOMETER_SUPPORTED` uncommented, you don't need to do anything else to get irradiance readings. However, in order for them to be at all accurate, you must perform a calibration, as described in Section 3.7 below. If you have included the optional TMP36 sensor, the irradiance values will be adjusted based on the measured temperature of the pyranometer sensor.

3.6.1 Placement / Mounting of the Pyranometer

The pyranometer must be pointed in the same direction as the PV module. One way to do this is to screw the back of the case to a piece of wood that is narrower than the distance between the two screws that hold the back to the case. A 1x3 (3/4 x 2-1/2 inches (19 x 64 mm)) works. This piece of wood can then be clamped to the PV module frame.



3.6.2 Irradiance Units

Irradiance is reported in W/m^2 . There is no option to report in any other units.

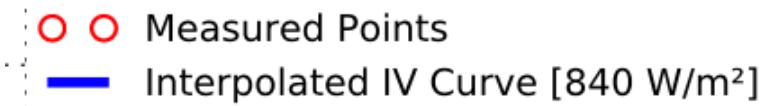
3.6.3 Precision

Irradiance is reported in **increments of 1 W/m^2** . There is no option to change the precision.

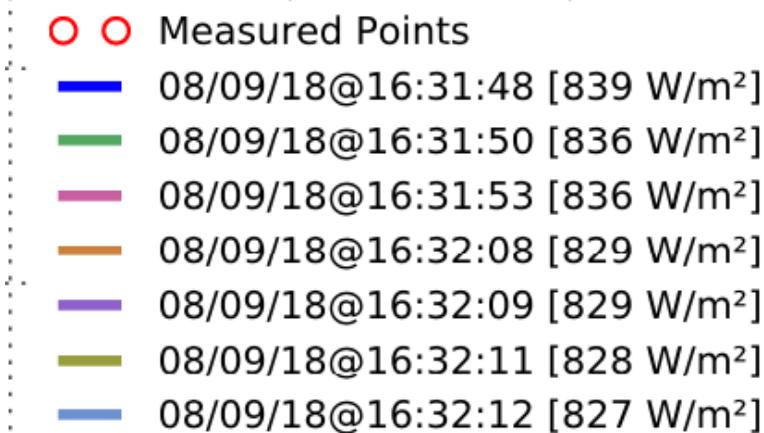
3.6.4 Where Is the Irradiance Recorded?

The laptop application records the irradiance in a “run info” file (which will also contain values from other environmental sensors if they exist). There is a run info file saved for each run and it is saved in the same directory/folder as the CSV, PDF and other files for the run. It can be opened from the File menu (View Run Info File).

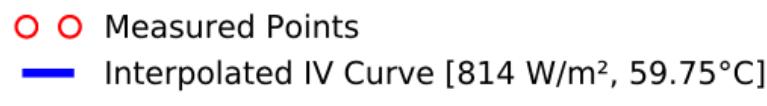
The irradiance is also included on the graph itself, in the legend. Here is an example:



Each curve on an overlay will have the irradiance for that run:



If you also have one or more DS18B20 temperature sensors, the temperature(s) will be listed in the brackets after the irradiance:



3.7 Calibrating the Pyranometer

The irradiance values measured by the pyranometer will be very inaccurate until it is calibrated.

3.7.1 Reference Pyranometer

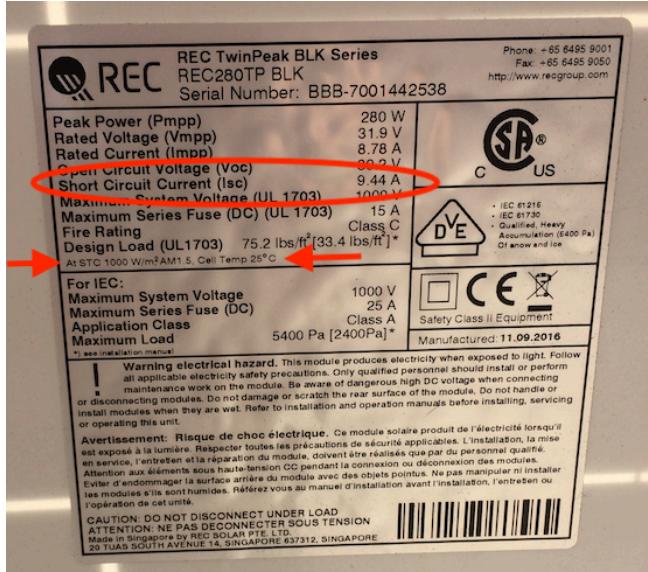
To perform an accurate calibration, you will need access to a reference pyranometer. The best reference pyranometer is a thermopile-based “true” pyranometer. As mentioned earlier, these are very expensive, and you probably do not have access to one. The next best is a commercial silicon sensor pyranometer such as the Kipp & Zonen SP-Lite (which Dr. Brooks uses for calibration). If you can rent or borrow either of these types of reference pyranometer, you will be able to accurately calibrate your pyranometer.

3.7.2 Using a PV Module as a Pseudo Reference Pyranometer

If you cannot obtain a reference pyranometer for the calibration, there is another option. This will not be as accurate, but it will get you in the right ballpark. You must have a PV module that is pretty new, clean, and known to be good. You must also have a clear sunny day, and be able to measure the temperature of the module's cells (with the DS18B20 sensors described in this document, or any other way).

3.7.2.1 Rated Isc

First, look at the label on the back of the PV module to determine the rated Short Circuit Current (Isc).



For this module, the Isc is 9.44 A. The arrows highlight the “Standard Test Conditions” (STC), which are 1000 W/m² with a cell temperature of 25°C.

3.7.2.2 Temperature Coefficient of Isc

What is not on the label is how the Isc is affected by temperature. Knowing this allows you to use the module at a temperature other than 25°C (which is much cooler than a module will be after even a minute or two in full sun). This information will be on the PV module datasheet as the “Temperature coefficient of Isc”. In this case, the datasheet says that it is 0.066 %/°C. That means that the Isc increases with temperature. So, for example, if the cell temperature is 60°C instead of 25°C:

$$\text{Isc increase \% @ } 60^\circ\text{C} = (60^\circ\text{C} - 25^\circ\text{C}) * 0.066 \%/\text{ }^\circ\text{C} = 2.31\%$$

Therefore, for this module the expected Isc at 1000 W/m² at 60°C would be:

$$\text{Isc @ } 60^\circ\text{C} = 1.0231 * 9.44 \text{ A} = 9.66 \text{ A}$$

3.7.2.3 Calculating Irradiance from Measured Isc

If you are fortunate to have a 100% clear day, and can arrange your PV module so the sun is shining exactly perpendicular to the module, you should measure an Isc that is close to the calculated value. It is likely to be somewhat less, however, due to atmospheric aerosols and other factors that mean that the

irradiance is less than 1000 W/m². It is also possible (but less likely) that the irradiance is greater than 1000 W/m².

We know, however, that Isc varies directly with irradiance. Therefore, if we trust the PV module ratings, we can use the measured Isc to calculate the irradiance:

$$\text{Calculated Irradiance} = 1000 \text{ W/m}^2 * (\text{Measured Isc}) / (\text{Isc @ 1000 W/m}^2)$$

Continuing our example, suppose that we've measured the cell temperature at 60°C, and we've measured the Isc at 8.11 A. We calculate the irradiance as:

$$\text{Calculated Irradiance} = 1000 \text{ W/m}^2 * 8.11 \text{ A} / 9.66 \text{ A} = 840 \text{ W/m}^2$$

See what happened there? We just used a PV module as a pyranometer! If that is the only PV module that you will ever be testing with your IV Swinger 2, you may be (rightly) wondering if building a pyranometer was worth all the trouble if your PV module itself is the reference. However, it can be very useful to compare different modules. Of course, if you are concerned about the validity of using your PV module as the reference, you should perform the calibration with an actual reference pyranometer.

3.7.3 Performing the Calibration

The calibration process consists of the following steps:

1. Swing an IV curve with the pyranometer
2. Select the “Pyranometer” entry from the “Calibrate” menu
3. Enter the “actual” (measured or calculated) irradiance in the dialog

If you are using a reference pyranometer, you need to measure the irradiance using that at the same time you swing the IV curve. In Step #3, you enter the value measured with the reference pyranometer.

If you are using the PV module as a pseudo reference pyranometer, you need to use the Isc value from the curve (and the temperature) and calculate the irradiance as described in Section 3.7.2 above, and enter that calculated value in Step #3.