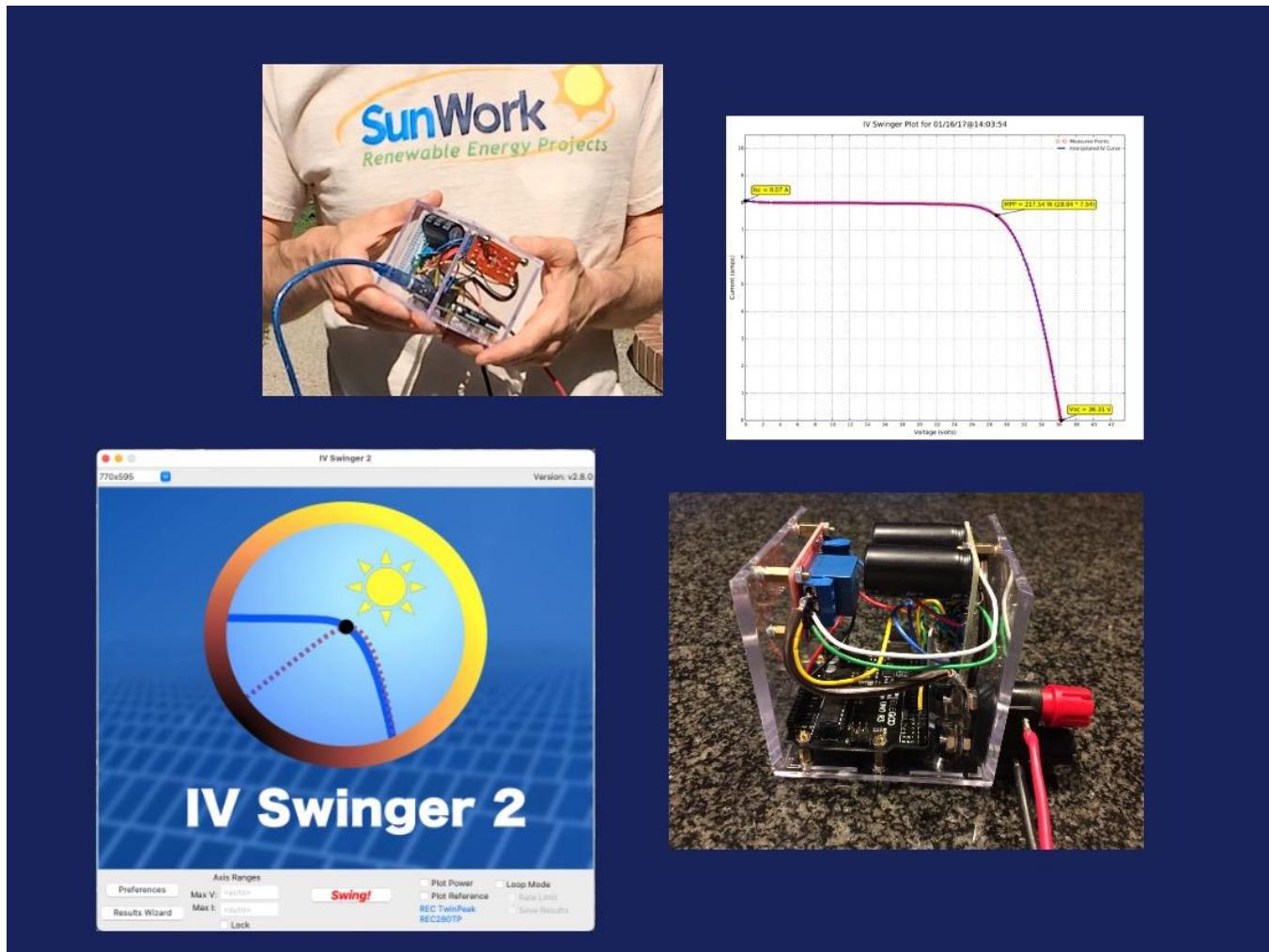


# IV Swinger 2

## User Guide

Document Revision: 1.12 (14-Apr, 2023)  
[Updated to Release v2.8.0]

Chris Satterlee



Copyright (C) 2017-2023 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](https://github.com/csatt/IV_Swinger)

# Table of Contents

<b>1</b>	<b>YouTube Demo Videos / Quick Start .....</b>	<b>6</b>
1.1	YouTube Demo Videos .....	6
1.2	Quick Start.....	6
<b>2</b>	<b>Introduction .....</b>	<b>7</b>
<b>3</b>	<b>Visual Tour of the Hardware .....</b>	<b>9</b>
3.1	Original version.....	9
3.1.1	USB Port.....	9
3.1.2	Binding Posts / PV Connection Cables.....	9
3.1.3	Innards .....	10
3.1.3.1	Circuit Board with Load Capacitors .....	10
3.1.3.2	Arduino UNO.....	11
3.1.3.3	Relay Module .....	12
3.2	Newer hardware variants.....	12
3.3	Optional Environmental Sensors .....	13
<b>4</b>	<b>Using the IV Swinger 2 Software .....</b>	<b>14</b>
4.1	Main Window .....	14
4.1.1	Swing! Button .....	15
4.1.2	Results Wizard Button .....	15
4.1.3	Preferences Button.....	16
4.1.4	Plot Power Button .....	16
4.1.5	Plot Reference Button (advanced feature) .....	17
4.1.6	Loop Mode Controls .....	18
4.1.7	Axis Ranges Control .....	19
4.1.8	Image Size Control.....	20
4.1.9	Version.....	20
4.2	Tooltips .....	21
4.3	Menus .....	21
4.3.1	About Menu (Windows) .....	22
4.3.2	IV Swinger 2 Menu (Mac) .....	22
4.3.3	File Menu.....	22
4.3.3.1	View Log File .....	23
4.3.3.2	View Config File .....	23
4.3.3.3	View Run Info File .....	23
4.3.3.4	Open Run Folder .....	24
4.3.4	USB Port Menu .....	24
4.3.5	Calibrate Menu .....	25
4.3.5.1	Vref (+5V).....	26
4.3.5.2	Voltage – basic .....	27
4.3.5.3	Current – basic .....	28
4.3.5.4	Voltage – advanced .....	28
4.3.5.5	Current – advanced .....	29
4.3.5.5.1	EMR advanced current calibration jumper.....	31
4.3.5.6	Resistors.....	32
4.3.5.7	Pyranometer .....	32
4.3.5.8	Bias Battery .....	32
4.3.5.9	Invalidate Arduino EEPROM .....	32
4.3.5.10	Calibration Help .....	32
4.3.6	Instances Menu .....	33
4.3.7	Help Menu .....	34
4.4	Results Wizard Dialog.....	36

4.4.1	Expanding and Collapsing Date Groups .....	36
4.4.2	Selecting Runs.....	37
4.4.3	Changing the Title.....	37
4.4.4	Overlaying Multiple Runs .....	38
4.4.4.1	Changing Overlay Plotting Options.....	39
4.4.4.2	Saving Overlays.....	39
4.4.5	Viewing the PDF.....	40
4.4.6	Batch Updates .....	40
4.4.7	Deleting Runs.....	41
4.4.8	Copying Runs .....	42
4.4.9	Changing Where to Look for Runs.....	44
4.4.10	Creating a Desktop Shortcut to Runs Folder.....	46
4.4.11	Importing Results from a USB Drive.....	47
<b>4.5</b>	<b>Preferences Dialog .....</b>	<b>48</b>
4.5.1	Plotting Preferences Tab .....	49
4.5.1.1	Line type.....	50
4.5.1.2	Isc, MPP, Voc labels .....	50
4.5.1.3	Font Name .....	50
4.5.1.4	Font scale, Line scale, and Point scale .....	50
4.5.1.5	ADC correction.....	50
4.5.1.5.1	Fix Isc point .....	50
4.5.1.5.2	Fix Voc point.....	51
4.5.1.5.3	Combine =V points.....	51
4.5.1.5.4	Reduce noise .....	51
4.5.1.5.5	Fix overshoot.....	51
4.5.1.6	Battery bias .....	51
4.5.1.7	Series resistance compensation ( $m\Omega$ ) .....	51
4.5.2	Looping Preferences Tab .....	52
4.5.3	Arduino Preferences Tab .....	53
4.5.3.1	SPI clock freq.....	53
4.5.3.2	Max IV points .....	54
4.5.3.3	Min Isc ADC.....	54
4.5.3.4	Max Isc poll .....	54
4.5.3.5	Isc stable ADC.....	54
4.5.3.6	Max discards .....	54
4.5.3.7	Aspect height .....	54
4.5.3.8	Aspect width .....	54
4.5.3.9	Relay is active-high .....	54
4.5.4	PV Model Preferences Tab .....	55
4.5.4.1	Adding a New PV to the List .....	56
4.5.4.2	Testing a New PV Model.....	56
4.5.4.3	Reference Curve Plotting Options .....	57
4.5.4.4	Adding a Reference Curve to an Existing IV Curve .....	57
4.5.5	Remote Command Preferences Tab .....	58
<b>4.6</b>	<b>Folders and Files .....</b>	<b>59</b>
4.6.1	IV_Swinger2.cfg file .....	59
4.6.1.1	IV_Swinger2_starting.cfg file .....	59
4.6.2	“logs” folder.....	59
4.6.3	Run folders and files .....	60
4.6.3.1	IV_Swinger2.cfg (in run folder).....	60
4.6.3.2	iv_swinger2_yymmdd_hh_mm_ss.pdf.....	60
4.6.3.3	iv_swinger2_yymmdd_hh_mm_ss.gif.....	60
4.6.3.4	iv_swinger2_yymmdd_hh_mm_ss.csv.....	60
4.6.3.5	adc_pairs_yymmdd_hh_mm_ss.csv.....	60
4.6.3.6	run_info_yymmdd_hh_mm_ss.txt .....	61

4.6.4	“overlays” folder.....	61
4.6.4.1	overlaid_yymmdd_hh_mm_ss.pdf.....	61
4.6.4.2	overlaid_yymmdd_hh_mm_ss.gif.....	61
4.6.5	“inst” folder .....	61
<b>5</b>	<b>Bugs.....</b>	<b>62</b>
<b>5.1</b>	<b>Known Bugs.....</b>	<b>62</b>
<b>5.2</b>	<b>Unknown Bugs and Enhancement Requests.....</b>	<b>62</b>
<b>6</b>	<b>Ratings and Limitations .....</b>	<b>63</b>
<b>7</b>	<b>Laptop Software Installation.....</b>	<b>64</b>
<b>7.1</b>	<b>Mac.....</b>	<b>64</b>
7.1.1	IV Swinger 2 application (Mac).....	64
<b>7.2</b>	<b>Windows .....</b>	<b>64</b>
7.2.1	IV Swinger 2 application (Windows).....	64
7.2.2	Arduino driver (Windows) .....	64
<b>8</b>	<b>Arduino Software Installation.....</b>	<b>65</b>
<b>9</b>	<b>IV Curve Tracing of PV Cells [DEPRECATED] .....</b>	<b>66</b>
<b>9.1</b>	<b>Cell Version IV Swinger 2.....</b>	<b>66</b>
9.1.1	High Power PV Cell Support .....	67
9.1.2	Low Power PV Cell Support .....	67
<b>9.2</b>	<b>Bias Battery .....</b>	<b>68</b>
<b>9.3</b>	<b>Connections.....</b>	<b>70</b>
<b>9.4</b>	<b>Second Relay .....</b>	<b>71</b>
9.4.1	Side effects of the 2 <sup>nd</sup> Relay .....	73
9.4.2	SSR cell version .....	73
<b>9.5</b>	<b>Application Configuration and Use .....</b>	<b>73</b>
9.5.1	Calibration .....	73
9.5.1.1	Resistor Values.....	73
9.5.1.1.1	Resistor R1 .....	73
9.5.1.1.2	Resistor RF.....	74
9.5.1.2	Bias Battery Calibration .....	74
9.5.1.2.1	Pre-requisite: normal voltage and current calibration .....	74
9.5.1.2.2	Manual bias battery calibration.....	74
9.5.2	Swinging an IV Curve with a Bias Battery .....	75
9.5.2.1	Applying the Battery Bias.....	75
9.5.2.2	Calibrating Based on a Biased IV Curve .....	76
<b>9.6</b>	<b>Consider Using Multiple High Power PV Cells in Series .....</b>	<b>76</b>
<b>10</b>	<b>Simulation.....</b>	<b>77</b>
<b>11</b>	<b>Remote Command Feature.....</b>	<b>78</b>
<b>11.1</b>	<b>Command Syntax.....</b>	<b>78</b>
<b>11.2</b>	<b>Reply Syntax.....</b>	<b>78</b>
<b>11.3</b>	<b>Supported Commands.....</b>	<b>78</b>
11.3.1	Swing Command .....	79
<b>11.4</b>	<b>Client Program Example.....</b>	<b>79</b>
11.4.1	Python .....	80
11.4.2	C.....	81

## Table of Figures

Figure 2-1: Typical IV Curve of Unshaded PV Module.....	7
Figure 3-1: USB Port .....	9
Figure 3-2: Binding Posts and PV Cables.....	10
Figure 3-3: Circuit Board with Load Capacitors .....	11
Figure 3-4: Arduino UNO.....	11
Figure 3-5: Relay Module .....	12
Figure 3-6: SSR version for PV modules .....	13
Figure 4-1: Main Window (annotated) .....	14
Figure 4-2: IV Curve with Power Plotted .....	17
Figure 4-3: IV Curve with Reference Curve Plotted .....	18
Figure 4-4: Results Wizard Dialog .....	36
Figure 4-5: Expanding a Date Group.....	36
Figure 4-6: Selecting a Run .....	37
Figure 4-7: Changing the Title of a Run.....	38
Figure 4-8: Creating an Overlay .....	39
Figure 4-9: Copying Runs to a USB Drive (Mac) .....	43
Figure 4-10: Copying Runs to a USB Drive (Windows) .....	44
Figure 4-11: Path to Results Folder .....	45
Figure 4-12: Changing the Path to a USB Drive .....	45
Figure 4-13: Path Successfully Changed to USB Drive .....	46
Figure 4-14: Making a Desktop Shortcut to Raw Results .....	47
Figure 4-15: Importing Results from USB Drive .....	48
Figure 4-16: Preferences Dialog .....	49
Figure 4-17: Preferences Looping Tab .....	52
Figure 4-18: Preferences Arduino Tab .....	53
Figure 4-19: Preferences PV Model Tab .....	55
Figure 4-20: Overlaid PV Model Test Curves .....	56
Figure 4-21: Preferences Remote Command Tab .....	58
Figure 4-22: Run folder contents .....	60
Figure 9-1: Cell Version IV Swinger 2.....	66
Figure 9-2: High-Power DIP Switch.....	68
Figure 9-3: Bias battery and PV cell connected.....	69
Figure 9-4: IV Curve for 2x2 D-cell bias battery pack.....	70
Figure 9-5: Binding Post Connections With Bias Battery .....	71
Figure 9-6: Connections with 2nd Relay OFF.....	72
Figure 9-7: Connections with 2nd Relay ON .....	72
Figure 9-8: Bias battery calibration dialog .....	75

# 1 YouTube Demo Videos / Quick Start

## 1.1 YouTube Demo Videos

**Part I:** <https://youtu.be/WhnTWciiNNo> (7:02)

**Part II:** [https://youtu.be/9iPq5AsuU\\_U](https://youtu.be/9iPq5AsuU_U) (6:48)

## 1.2 Quick Start

### 1. Open the IV Swinger 2 application on your Mac or Windows laptop

Software installation instructions are on page 64. There should be a “Not connected” message below the button labeled “Swing!”

### 2. Connect the USB cable from the IV Swinger 2 hardware to the laptop

The “Swing!” button on your screen should change to red text and the “Not connected” message should disappear. If this does not happen, see section 4.3.4 on page 24.

### 3. Connect the PV panel to the cables

Take care to connect the positive (+) cable from the panel to the female IV Swinger cable (RED binding post) and the negative (-) cable from the panel to the male IV Swinger cable (BLACK binding post).

### 4. Establish the desired PV panel conditions

i.e. angle, shading, etc. Also at this point, take any other desired measurements such as irradiance and temperature if standalone sensors are being used.

### 5. Click on the “Swing!” Button

The IV curve will be displayed on the screen. Repeat as many times and as often as desired. All results are saved and can be viewed, copied, combined, modified, etc. later.

- "Plot Power" can be checked to include the power curve on the graph. This can be done after the fact too.
- "Loop Mode" can be checked to repeatedly swing IV curves. "Rate Limit" can be checked to slow down the looping rate to a specified interval. Since looping can generate a lot of data and image files, the default is to not save the results. "Save Results" can be checked to override this default.

### 6. Click on the “Results Wizard” Button

A dialog window will open with options to:

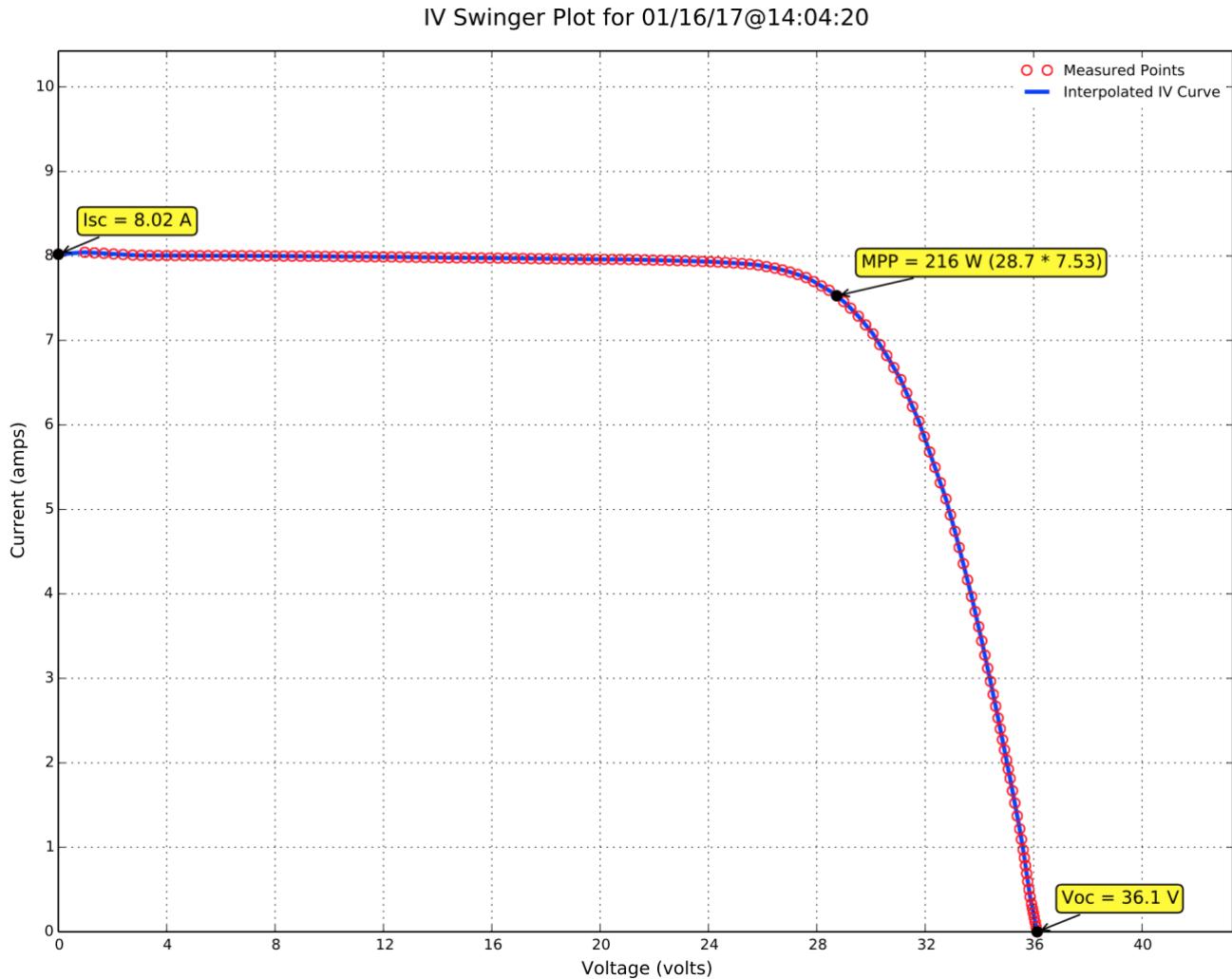
- View results of previous runs
- Combine multiple curves on the same plot (overlays)
- Modify the title and appearance of curves and overlays
- Copy them to a USB drive (or elsewhere)
- View the PDF

The Results Wizard does not require the hardware to be connected. In fact, it can be run on results that were collected by a different computer and copied to a USB drive.

## 2 Introduction

Like the first-generation IV Swinger, IV Swinger 2 automates the process of tracing the IV curve of a single photovoltaic (PV) solar panel.

The IV curve is a graph of current versus voltage for various load values driven by the PV panel. A typical IV curve of an unshaded PV panel looks like this:



**Figure 2-1: Typical IV Curve of Unshaded PV Module**

At one extreme there is the short-circuit current ( $I_{sc}$ ), which is the current delivered by the PV when the leads are directly connected to each other (in which case the load is 0 ohms and the voltage is 0 volts). At the other extreme is the open-circuit voltage ( $V_{oc}$ ), which is the voltage across the PV terminals when there is no connection between them (in which case the load resistance is infinite and the current is 0 amps). Between these two points is a curve dependent on the load (resistance) connected between the PV terminals. As seen above, the typical IV curve is a fairly horizontal line declining slightly from  $I_{sc}$  as the voltage increases and then declining steeply when the voltage nears  $V_{oc}$ . Power is the product of current and voltage. No power is delivered when the circuit is shorted (since the voltage is 0) and no power is delivered when the circuit is open (since the current is 0). The maximum power is delivered for the load value at the “knee” of the curve. One purpose of generating an IV curve is to determine the maximum power that a given panel can generate under a given set of conditions (such as irradiance and

temperature). Plotting the IV curves for different conditions including shading is an important lab exercise in courses teaching principles of photovoltaic operation

Manual generation of an IV curve is a fairly tedious (albeit educational) process. First the  $I_{sc}$  and  $V_{oc}$  values are measured with a multimeter. Then the PV panel is connected to a variable load. This can be a bank of light bulbs that may be switched into or out of the circuit. Or it can be a heavy-duty rheostat. The load value is incrementally changed, and at each point the meter is used to measure the current and voltage, which are recorded manually. Then the values are typed into a spreadsheet and plotted. Performing this process by hand at least once is a good way to get a real feel for what is going on. But it is time-consuming, subject to human error, subject to changing conditions, etc.

The IV Swinger is named for the process of “swinging out an IV curve”, which is how Gil Masters described it in his CEE176B class at Stanford. The first-generation IV Swinger uses immersion heating coils for the loads, and a Raspberry Pi single-board computer to control relays that switch the loads into and out of the circuit. It has built-in voltmeter and ammeter hardware that are read by the software at each load value. The results are written to a USB thumb drive for later viewing.

IV Swinger 2 is a completely different design. It uses a capacitor as the load. A discharged capacitor “looks” like a short circuit (zero resistance). But as it charges up, its apparent resistance increases. When it is fully charged, it “looks” like an open circuit (infinite resistance). A typical PV module in full sun will charge the capacitor up to the  $V_{oc}$  voltage in a matter of milliseconds. An Arduino UNO microcontroller board is used to control a single relay that switches between bleeding the charge from the capacitor to connecting the PV module across it. After switching the PV module into the circuit, the Arduino very quickly reads from the voltmeter and ammeter hardware repeatedly to record points on the IV curve as the capacitor charges up.

IV Swinger 2 requires a Mac or Windows laptop to be connected to it by a USB cable. The power for the Arduino and other circuitry comes through the USB cable. An application program running on the laptop communicates with the Arduino via the USB cable and provides the interface to the user.

Although IV Swinger 2 is not quite as easy to relate to the manual method as the original IV Swinger, it has many advantages:

- Real time results
- Much better resolution (>100 points per curve vs. ~20)
- Much lower time to swing a curve (milliseconds vs seconds)
- Much smaller size and lower weight
- Simpler user interface
- Simpler, more reliable and repairable design
- More flexible connectivity (binding posts)
- Much lower cost to build (\$50 vs \$330 in 2017)
- Much less time and effort to build

IV Swinger 2 also has a variant that works with high-power PV cells. That version is now deprecated, but using different component values with the “module” version works well for most smaller PV cells. Please refer to the document [“IV Swinger 2: Hardware Scaling.”](#)

### 3 Visual Tour of the Hardware

#### 3.1 Original version

There are now several variants of the hardware. This section describes the original version. Many of these exist and still work fine with the latest software. The newer variants are described in Section 3.2 below.

##### 3.1.1 USB Port

On the front of the IV Swinger 2 is a single “Type B” USB port/jack. This how the IV Swinger 2 connects to the host laptop computer both for power and communication. Note that when the USB cable is plugged into this port, the lid is “locked” onto the case and when the cable is unplugged, the lid may be removed.

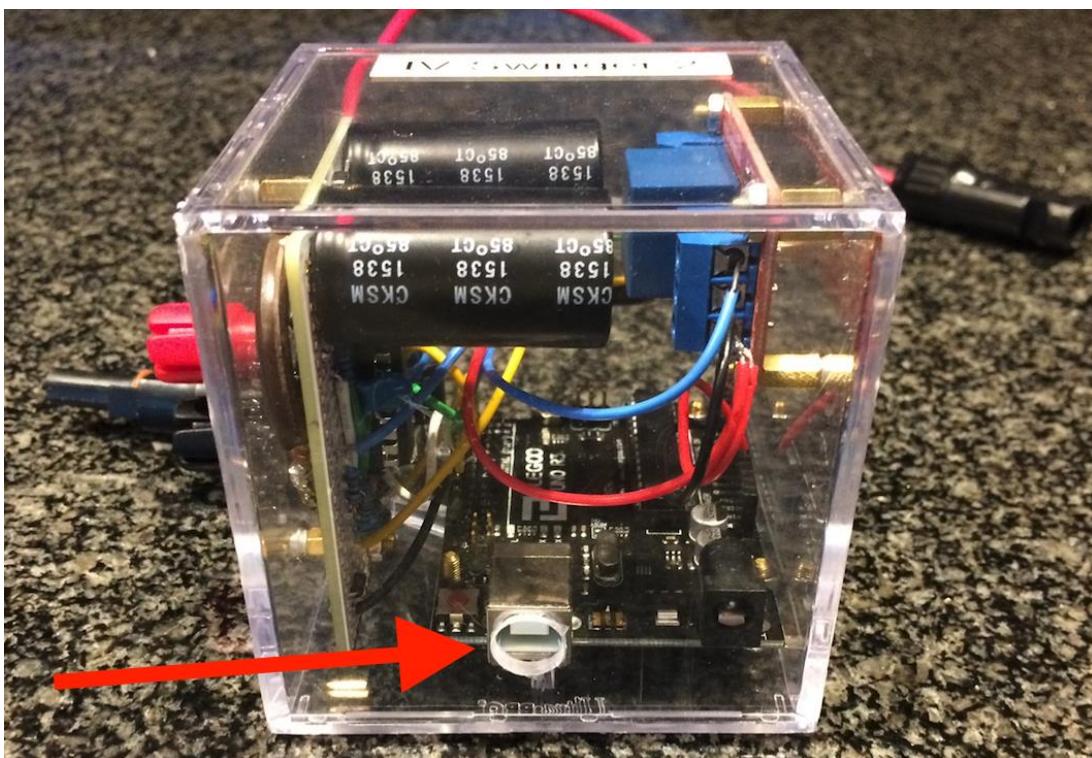


Figure 3-1: USB Port

##### 3.1.2 Binding Posts / PV Connection Cables

On the lower left side, at the rear, are the RED and BLACK binding posts. The RED post connects to the positive module cable (PV+) and the BLACK post connects to the negative module cable (PV-). The PV connection cables are inserted into the holes in the binding posts at the back, and the knobs tightened down. Make sure the connection cable with the female MC4 connector is connected to the RED binding post and the cable with the male MC4 connector is connected to the BLACK binding post. The binding posts allow for the flexibility of using different length PV connection cables or even non-MC4 connections such as banana plugs, but you have to be careful to get the polarity correct.



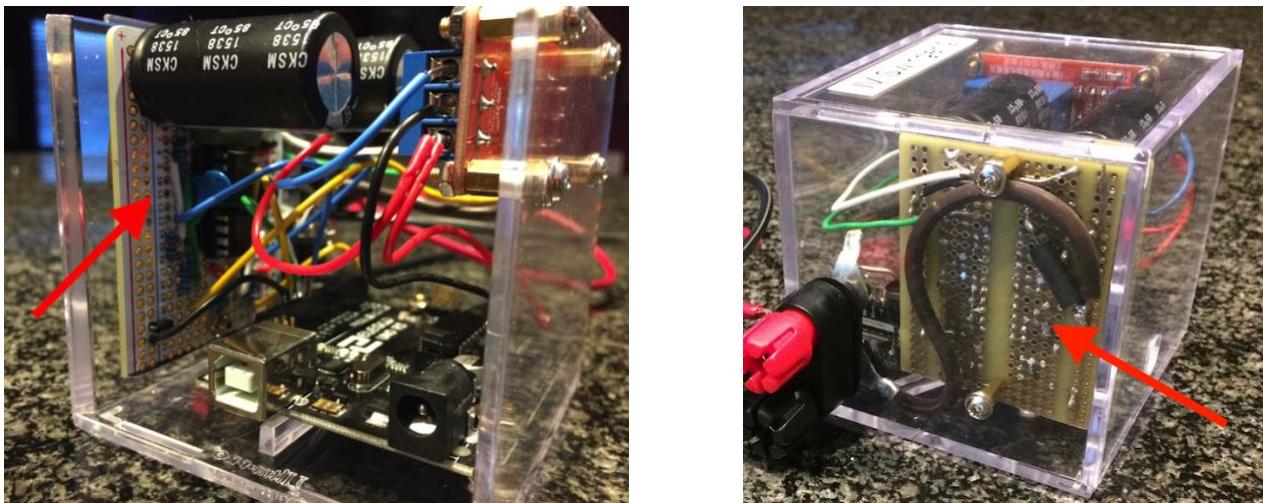
**Figure 3-2: Binding Posts and PV Cables**

### 3.1.3 Innards

#### 3.1.3.1 Circuit Board with Load Capacitors

The white “Perma-Proto” board on the side with the binding posts contains the following:

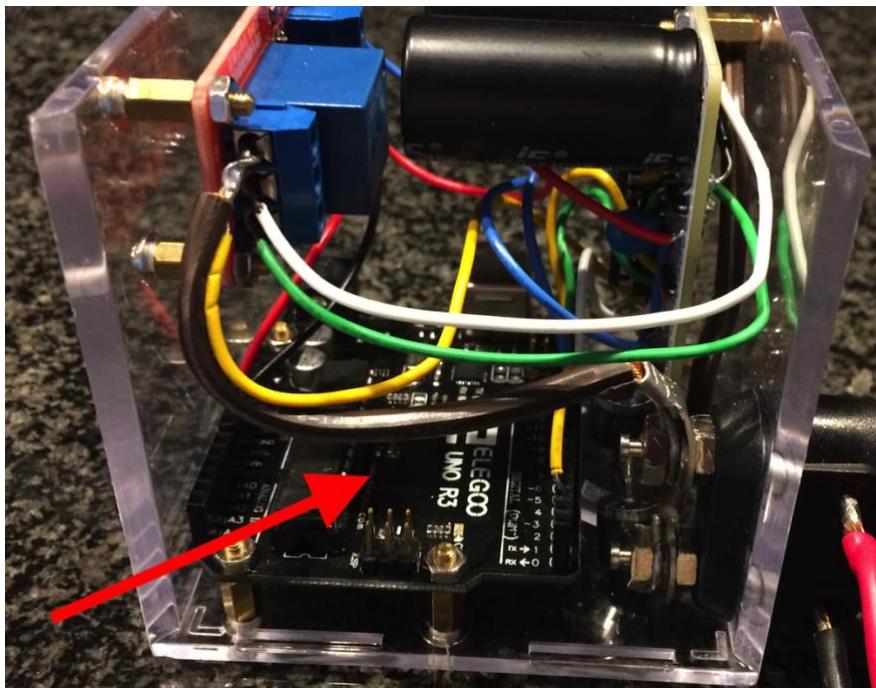
- 2000 $\mu$ F load capacitance (two 1000 $\mu$ F capacitors in parallel)
- Bleed resistor. Between tests, the relay connects this resistor across the capacitors to drain them of all charge.
- Analog-to-Digital Converter (ADC) for voltmeter and ammeter
- Shunt resistor (on back). This is used for the ammeter. It is a precise low-resistance (.005 $\Omega$ ) resistor that is in the load circuit. The voltage across it is measured by the ADC (after amplification by the op amp circuit). Ohm’s Law is then used to calculate the measured current.
- Voltage divider circuit to scale the PV voltage down to the range supported by the ADC
- Op amp circuit to scale the voltage across the ammeter shunt resistor up to the range supported by the ADC



**Figure 3-3: Circuit Board with Load Capacitors**

### 3.1.3.2 Arduino UNO

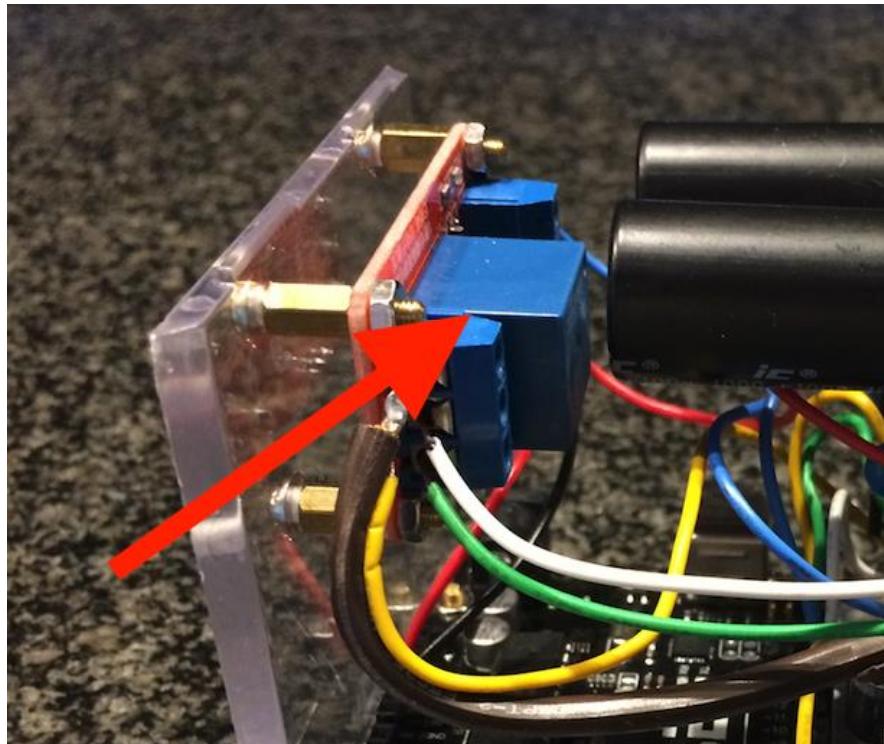
The Arduino UNO is the board (with the USB port) mounted to the bottom of the case. This is a very inexpensive (~\$10) and ubiquitous microcontroller that is a staple of the “maker” community. It runs software that communicates with the host laptop computer over USB. When instructed by the laptop, it runs the algorithm for swinging the IV curve. This algorithm consists of activating the relay and repeatedly reading the ammeter and voltmeter values from the ADC while the capacitor charges up. When the points have all been measured, the values are uploaded to the laptop.



**Figure 3-4: Arduino UNO**

### 3.1.3.3 Relay Module

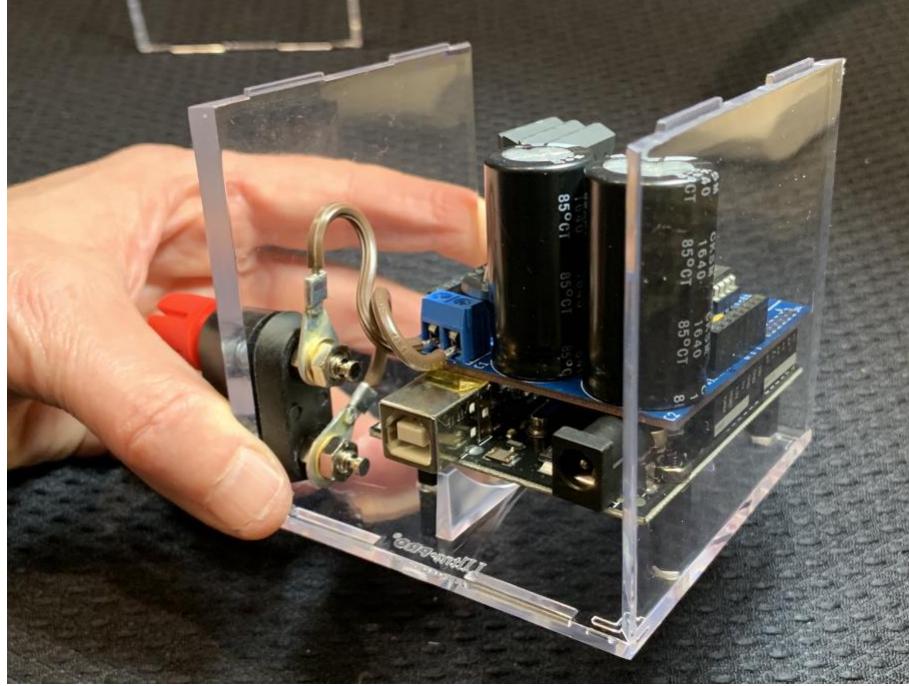
The relay is controlled by the Arduino software to switch what is across the capacitor load: the bleed resistor or the PV module. The bleed resistor is selected most of the time. The PV module is selected only for the brief amount of time that the IV curve is being traced. A relay is a physical switch operated by an electromagnet. You can hear it “click” when it switches. The two clicks (representing the beginning and end of the IV curve) will be very close together in most cases (“click-click”). But in low sun cases, there can be several seconds between the two clicks since the capacitor load takes a lot longer to charge up to the Voc voltage.



**Figure 3-5: Relay Module**

## 3.2 Newer hardware variants

Newer IV Swinger 2's are constructed differently from what is described and pictured in Section 3.1 above. Instead of using a Perma-Proto board, they use a custom printed circuit board (PCB). The PCB is an “Arduino shield”, meaning it has pins that mate directly with the sockets on the top of the Arduino. This makes building an IV Swinger 2 much easier and mistake proof. Furthermore, there are options to use solid-state relays (SSRs) or field-effect transistors (FETs) instead of the electromechanical relay (EMR) described in Section 3.1.3.3 above. The only wires in SSR/FET models are from the PCB to the binding posts. SSRs and FETs are silent, but more importantly, can handle a higher voltage and should never wear out like an EMR (the downside is that they are more expensive). Figure 3-6 below shows the SSR version for PV modules. There is also an EMR version and a FET version of the PCB for PV modules and [deprecated] EMR and SSR versions for high-power PV cells. The current software works with all IV Swinger 2 variants, including the older ones. Most of what is in this document applies regardless of which variant is being used. However, there are a few cases where it is relevant.



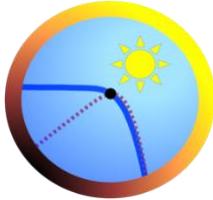
**Figure 3-6: SSR version for PV modules**

### 3.3 Optional Environmental Sensors

There is also support for optional temperature and irradiance sensors. A separate document entitled “[IV Swinger 2: Optional Environmental Sensors](#)” describes how to build those sensors and how to use them. For the most part, this document does not cover the usage of these sensors; however, there are several places where they are mentioned.

## 4 Using the IV Swinger 2 Software

The IV Swinger 2 software application runs on a Mac or Windows laptop. If it is already installed, you can recognize it from its icon:



On a Mac, the program will be in the Applications folder. On Windows, it will be in the standard place that your version of Windows keeps programs (e.g. the “Program Files (x86)” folder).

If the application is not already installed, see the *Laptop Software Installation* section on page 64 of this document.

### 4.1 Main Window

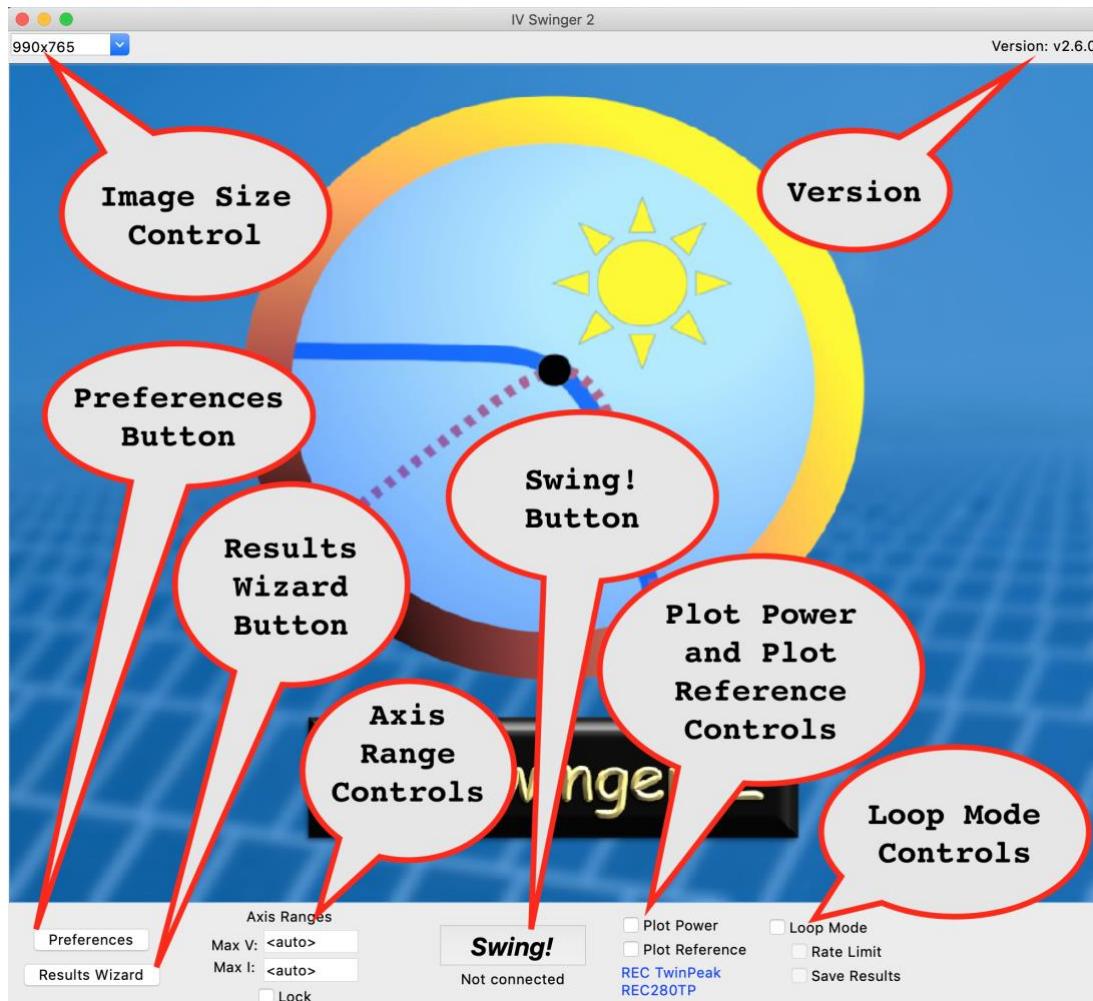


Figure 4-1: Main Window (annotated)

When the application is first opened, a “splash screen” with the logo and title is displayed. The annotated image in Figure 4-1 above is from the Mac version. The Windows version is the same other than the standard window control buttons and the fact that Windows includes the menus on the window itself<sup>1</sup>. In order of importance, the annotated items are as follows:

#### 4.1.1 Swing! Button

The Swing! button is used to initiate the tracing of an IV curve. Before the IV Swinger 2 hardware is connected, the button looks like this:



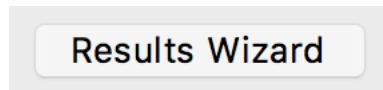
The button cannot be pressed until the hardware is connected and the button changes its appearance to this:



If this does not happen, see section 4.3.4 on page 24.

Pressing the button can be done either by clicking it with the mouse or by pressing the Enter key.

#### 4.1.2 Results Wizard Button



Click this button to open the Results Wizard, which is a dialog that provides options to:

- View results of previous runs
- Combine multiple curves on the same plot (overlays)
- Modify the title and appearance of curves and overlays
- Copy them to a USB drive (or elsewhere)
- View the PDF

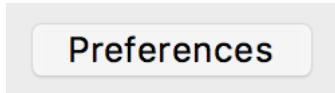
The Results Wizard does not require the hardware to be connected. In fact, it can be run on results that were collected by a different computer and copied to a USB drive.

---

<sup>1</sup> Most of the screen captures in this document are from the Mac version. There are minor differences in the “look and feel” of the Windows equivalents, but no functional differences.

Detailed usage instructions are provided in the Results Wizard Dialog section on page 36.

#### 4.1.3 Preferences Button



Click this button to open the Preferences Dialog, which provides options to:

- Change the appearance of the plotted graph (font size, line size, etc.)
- Change the behavior of the looping option
- Change the behavior of the low-level Arduino code
- Specify a PV model for the Plot Reference feature
- Configure the Remote Command feature

Detailed usage instructions are provided in the Preferences Dialog section on page 48.

#### 4.1.4 Plot Power Button



Check this box to display the power curve overlaid on the IV curve.

IV Swinger Plot for 01/16/17@14:04:20

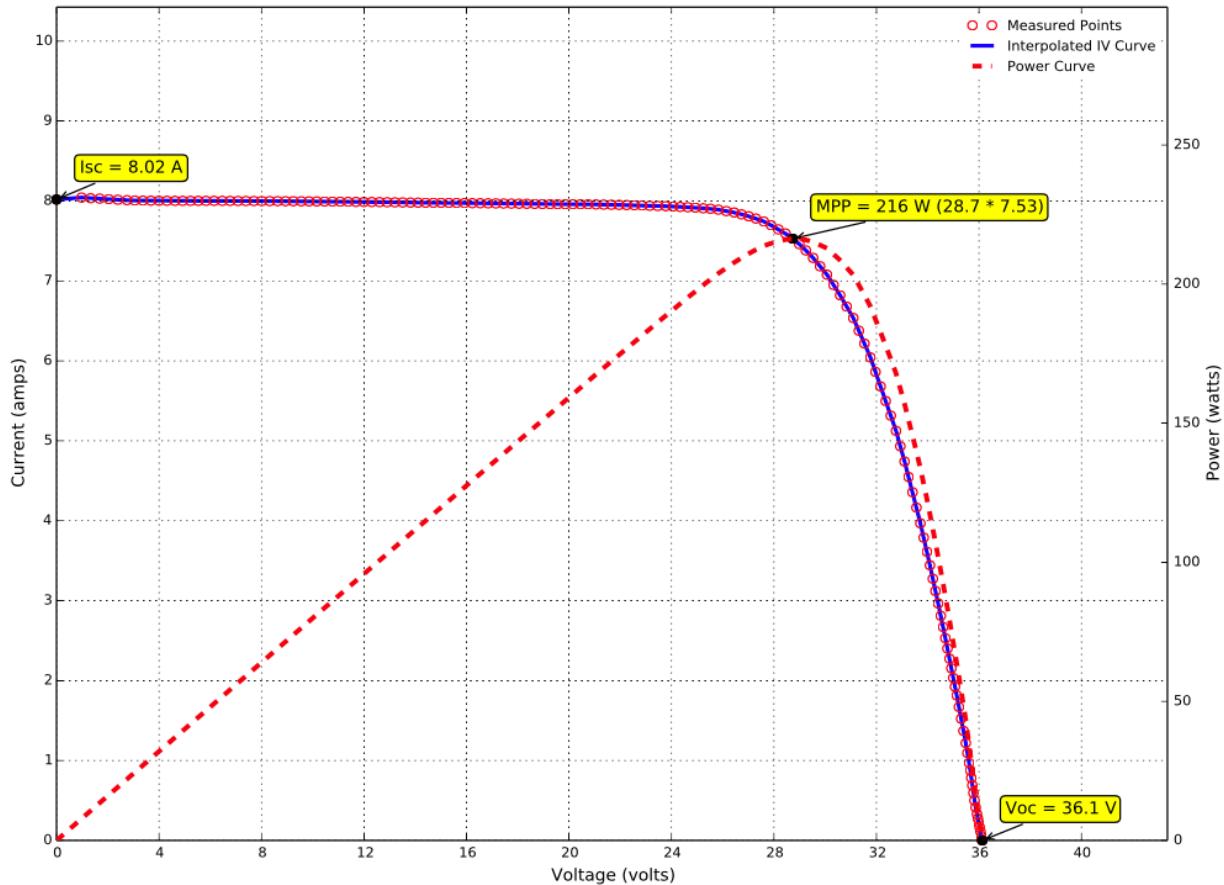
**Figure 4-2: IV Curve with Power Plotted**

Figure 4-2 above is an example. The graph now has a secondary vertical axis on the right indicating power in watts. This axis is scaled such that the power curve (dashed red line) intersects the IV curve at the Maximum Power Point (MPP).

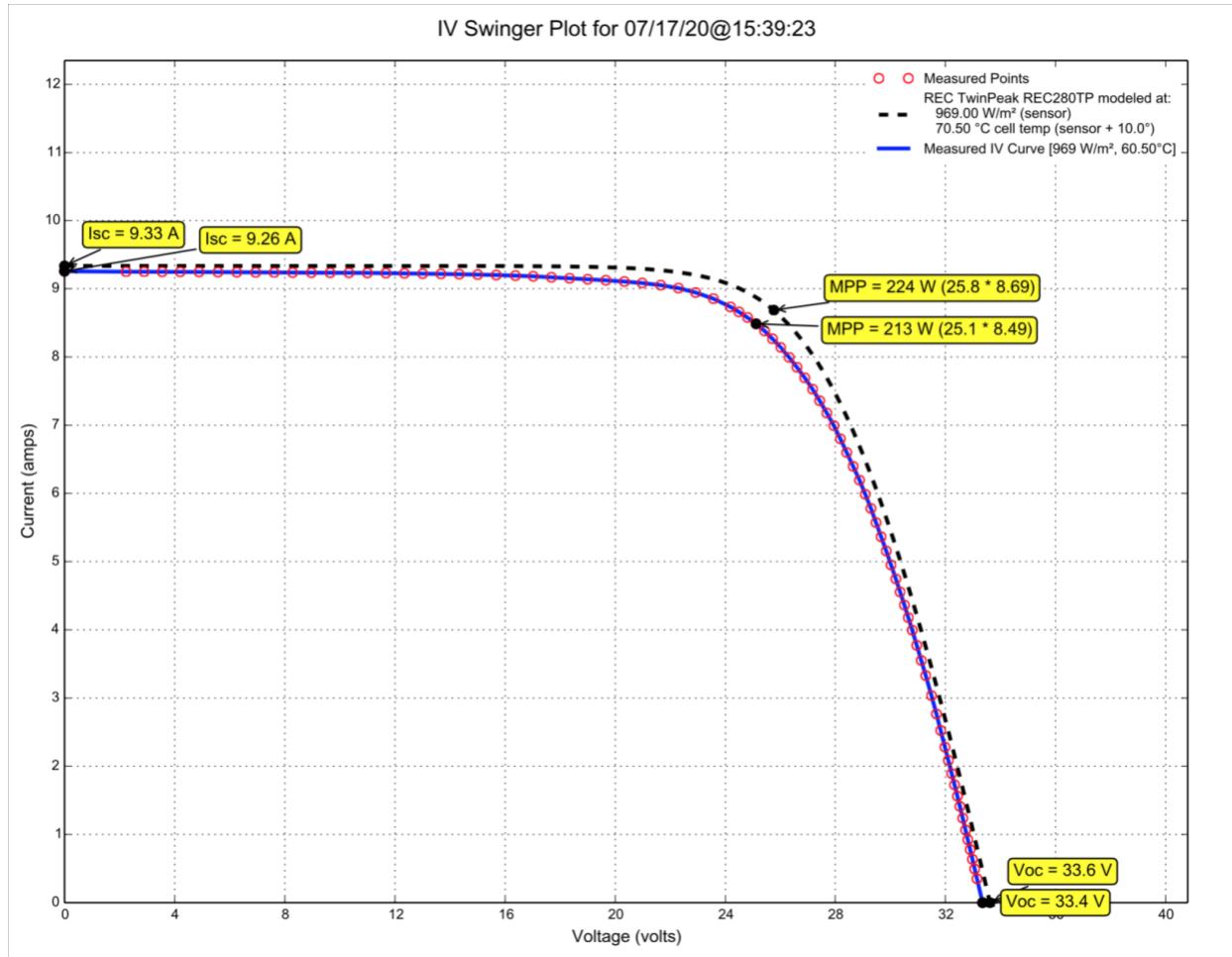
#### 4.1.5 Plot Reference Button (advanced feature)

Plot Reference

REC TwinPeak

REC280TP

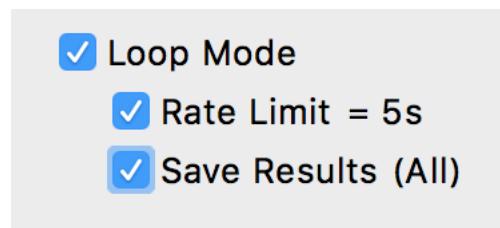
Check this box to display the PV model reference curve overlaid on the IV curve. This button is disabled unless a PV has been selected on the "PV Model" tab of Preferences.



**Figure 4-3: IV Curve with Reference Curve Plotted**

Figure 4-3 above is an example. Generating the reference curve is performed using a mathematical model of the PV from the specifications contained in the datasheet for the particular PV under test. The irradiance and cell temperature are needed, which implies the use of optional sensors. However, it is possible to estimate the irradiance and cell temperature from the measured  $I_{SC}$  and  $V_{OC}$  values. See Section 4.5.4 “PV Model Preferences Tab” on page 55 for more information.

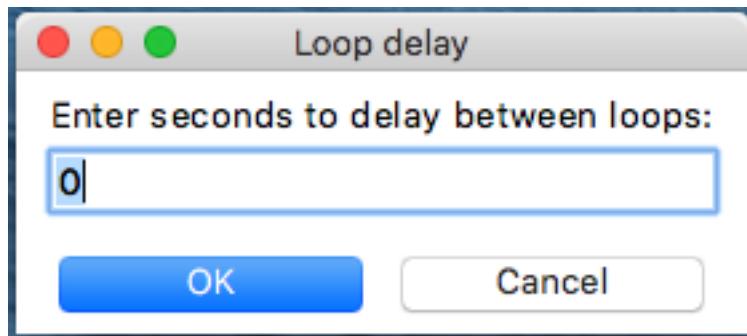
#### 4.1.6 Loop Mode Controls



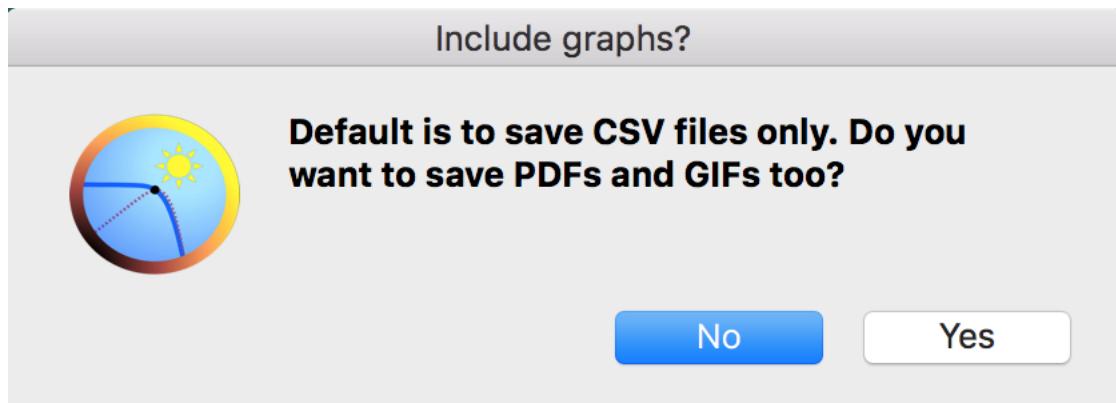
By default, pressing the Swing! button generates a single IV curve. Checking the “Loop Mode” box enables automatic repetition of the curve tracing. When this box is checked, pressing the Swing! button starts the looping. The Swing! button changes to a STOP button and the looping continues until that button is pressed.

If neither of the other two boxes are checked, a maximum of one IV curve is traced every one second and none of the results are saved. This mode can be useful to observe the effect of different types of shading in real time without cluttering your disk with the all of the saved results.

If the Rate Limit box is checked, a small dialog pops up requesting the number of seconds to delay between repetitions<sup>2</sup>:

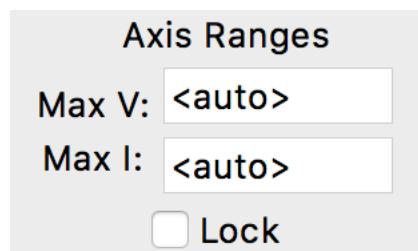


If the Save Results box is checked, the following dialog pops up:



If you choose “No”, only the (very small) CSV files are saved and the graph files are not saved. However, if the Results Wizard is used to browse these results later, it will re-generate the graph files. If you choose “Yes”, the graph files will be saved too.

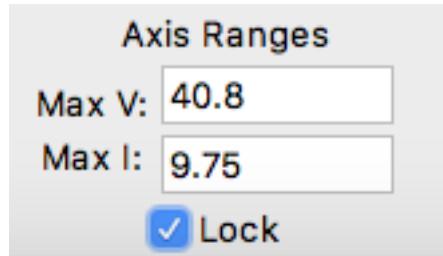
#### 4.1.7 Axis Ranges Control



By default, the axis ranges are chosen automatically, i.e. the maximum voltage on the X-axis and the maximum current on the Y-axis are chosen so the displayed IV curve fits nicely on the graph. The axis range controls can be used to override this behavior.

<sup>2</sup> Older, slower laptops may not be able to keep up with the specified rate if the delay is small.

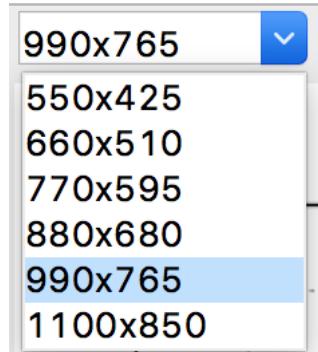
The Lock box can be checked to lock the ranges to the values of the first (or most recent) run or to values entered by the user (just type the desired value and hit Enter). If the values are changed when a plot is currently displayed, it will be redrawn using the new values.



A typical use case for this would be if you want to swing a series of IV curves and see them all at the same scale. After swinging the first one in the series, check the box and then swing the others.

Note that when Loop Mode is enabled, the Lock box is automatically checked. This is so all iterations of the loop are plotted at the same scale so the changes from each iteration to the next are properly visualized relative to each other. However, for long running tests where the irradiance is expected to change a lot, it may make sense to uncheck the Lock box so the curves don't go out of range (this override was not possible prior to v2.8.0.)

#### 4.1.8 Image Size Control



The image size control in the upper left of the main window controls the size of the displayed image in pixels (width x height). If the default size isn't a good fit for your laptop screen, use this to change it. The drop-down menu has several choices. You may also just type in a size, but it must be in the ratio of 11.0 x 8.5 (actually, just type in the width, and the height will be calculated for you).

Note that this only affects the size of the GIF image that is used to display the graph on your laptop screen. The PDF is always the same size: 11.0" x 8.5".

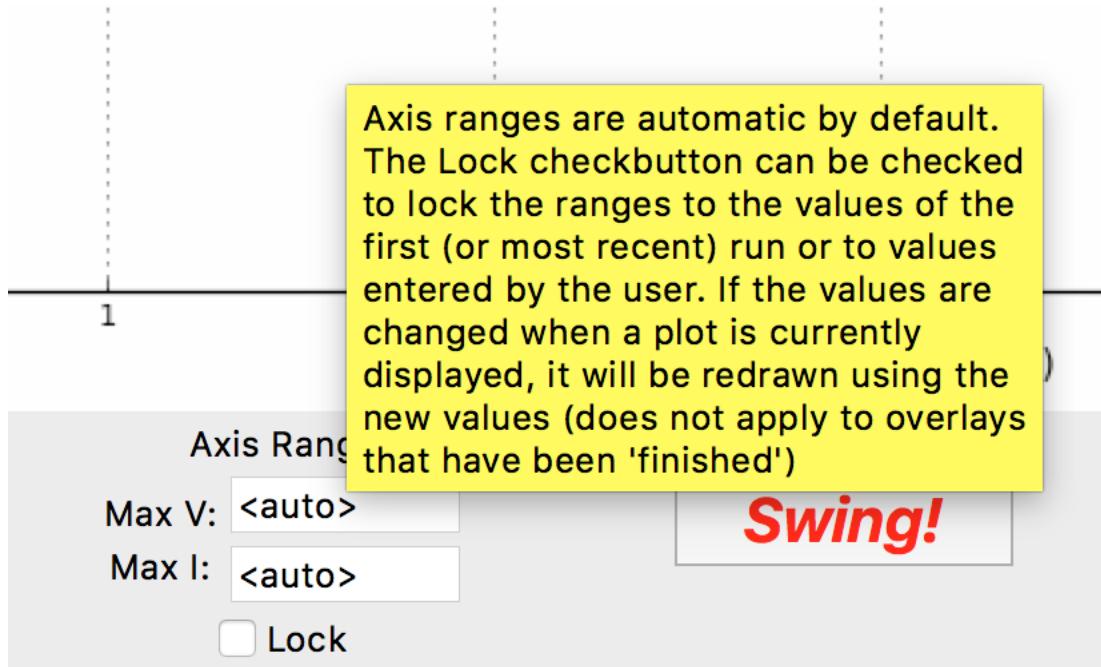
#### 4.1.9 Version

Version: v2.1.3

The version number of the software is displayed in the upper right corner of the main window.

## 4.2 Tooltips

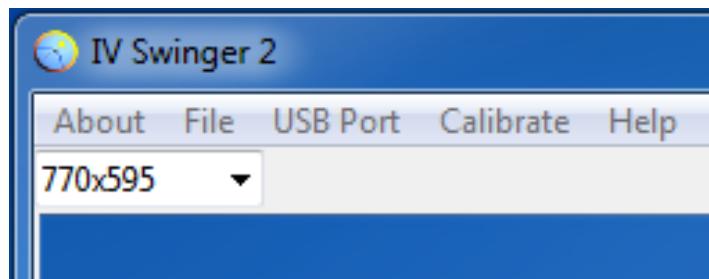
Most of the user interface tools (e.g. buttons, checkboxes, drop-downs, etc.) have “tooltips”, which are little yellow pop-up help windows that appear when the mouse is hovered over the tool. For example:



The tooltip disappears after you've had enough time to read it. But if you weren't finished, just move the mouse off and then back onto the tool and it will reappear.

## 4.3 Menus

As mentioned earlier, the Windows version has the menus along the top of the application window (like all Windows applications):



The Mac version puts the menus at the top of the screen when the application has focus (like all Mac applications):



Two other minor differences are that the first menu is named “About” in the Windows version and is named “IV Swinger 2” in the Mac version, and there is a “Window” menu in the Mac version. Otherwise, they are identical.

#### 4.3.1 About Menu (Windows)

This menu has only one item: “About IV Swinger 2”. This brings up a dialog with the software version number, open-source license info and copyright.

#### 4.3.2 IV Swinger 2 Menu (Mac)

This menu is the typical Mac “application” menu:

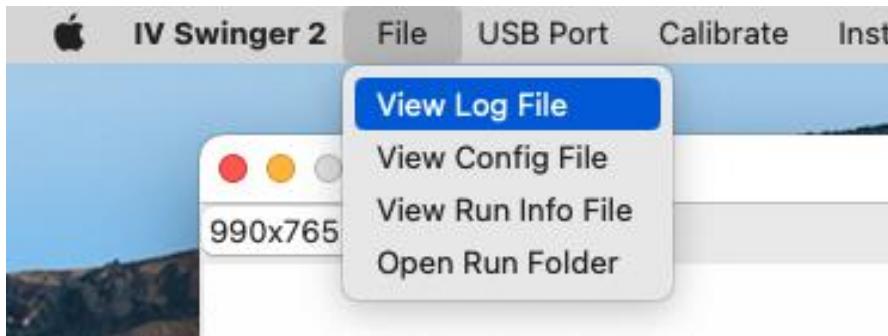


The “About IV Swinger 2” item brings up a dialog with the software version number, open-source license info and copyright.

The others are standard for all Mac applications, but not necessarily useful. The Preferences item brings up the Preferences Dialog and is equivalent to pressing the Preferences button.

#### 4.3.3 File Menu

The File menu has four items: View Log File, View Config File, View Run Info File, and Open Run Folder.



#### 4.3.3.1 View Log File

The log file contains debug information logged by the software. This can be useful for troubleshooting software problems, and you may be asked to send this file to the developer if you run into a bug. There is a single log file for each session (i.e. from the time you start the application until you close it), so it contains the information for all runs that were performed during that session. Selecting this menu item brings up a file dialog allowing you to select which log file to view. The default selection is the current log file unless you are looking at an older run from a previous session using the Results Wizard, in which case the default selection is the log file for that older session. The editor that is used depends on how you have your computer set up to open text files (.txt extension).

#### 4.3.3.2 View Config File

Most users will never need to use this. The Config file contains the preferences and other values that are retained between invocations of the application. A copy of the Config file is also saved for each run so that the values applicable to that run are saved (and possibly modified later). When the Results Wizard is in use and a previous run is being viewed, the Config file for that run is displayed when “View Config File” is selected.

#### 4.3.3.3 View Run Info File

The Run Info file is a text file that may be used to capture additional information about a particular run. If the optional temperature and/or irradiance sensors are used for a given run, those measurements are written to this file by the software. The user may also add temperature and irradiance measurements manually, if those were taken using independent sensors. And finally, the user may add any additional info that they want to, in any format - for example, module (or cell) type, tilt angle, notes on shading, notes on sky conditions, etc.

If the Results Wizard is not in use, the Run Info file for the current run is opened in your text editor when this menu item is selected. When the Results Wizard is in use and a previous run is being viewed, the Run Info file for that run is opened. Therefore, you may either add your info immediately after the run, or you may add it later. The editor that is used depends on how you have your computer set up to open text files (.txt extension).

If the optional temperature and/or irradiance sensors are used, the measurement values that the software writes to this file are also displayed in the legend of the IV curve plot. If independent sensors are used for these measurements, those values will also be displayed in the legend of the plot if the user enters them in the same format that the software uses, e.g.:

Irradiance: 774 W/m<sup>2</sup>  
Temperature at sensor #1 is 52.00 degrees Celsius

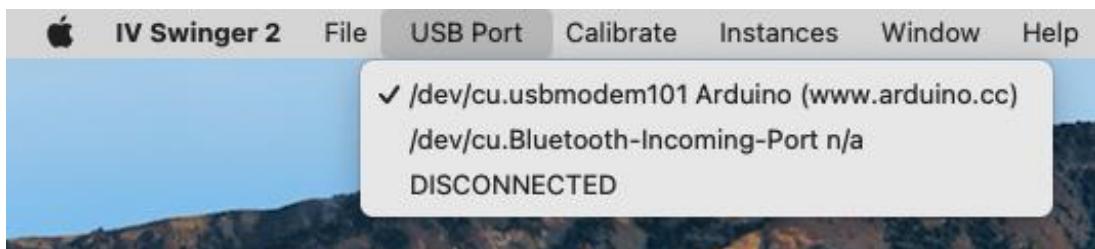
The irradiance value must be a positive integer (no decimal point). The temperature value may be positive or negative, and may be either integer or floating point. If there are multiple temperature sensors, each must be listed on a new line. They are displayed in the order they appear in the file.

#### 4.3.3.4 Open Run Folder

If the Results Wizard is not in use, the folder for the current run is opened when this menu item is selected. When the Results Wizard is in use and a previous run is being viewed, the folder for that run is opened. The default file manager application is used (e.g. Finder for Mac, Explorer for Windows). See Section 4.6.3 on page 60 for descriptions of the files in the run folder. Most users will have no need for this. It can be useful, however, to open the CSV files to see the actual measurement values. Just be warned that any changes made to the ADC CSV file (and saved) will cause the loss of the recorded values. If you want to make changes to the CSV files, it is recommended to first use the Results Wizard “Copy” button to create a copy of the run directory.

#### 4.3.4 USB Port Menu

The USB Port menu lists each serial port that is connected to a device. It can be used (if necessary) to tell the software what USB port the IV Swinger 2 is connected to.



Normally the software figures this out by itself and automatically selects the correct port, so you may never need to use this menu. However, if the “Swing!” button does not turn red when you plug the IV Swinger 2 USB cable into the laptop, you need to figure out which port is the correct one and select it manually. It may be obvious from the name. If not, do this:

- Close the IV Swinger 2 application and disconnect the IV Swinger 2 USB cable from the laptop
- Re-open the IV Swinger 2 application (leave the cable disconnected)
- Pull down the USB Port menu and take note of the listed ports
- Connect the USB cable from the IV Swinger 2 hardware to the laptop
- Pull down the USB Port menu and select the port that is new to the list

Note that if you unplug the IV Swinger 2 from one USB port and plug it into a different one while the application is running, you may need to manually select the new USB port using the menu (depends on the OS and version). You may even need to close the application and open it again. It is best to choose one USB port and stick with it.

The DISCONNECTED menu entry may be selected to force the software to not connect to any USB port. This can be needed when using the “instances” feature (see Instances Menu on p.33.)

### 4.3.5 Calibrate Menu

For the original target users of IV Swinger 2 (students first learning about PV technology), a high degree of accuracy is not critical. Without calibration, the voltage and/or current measurements may be inaccurate by 10% or more, but that will have virtually no impact on the learning value because the IV curve shapes will be correct and differences will be accurately represented. More sun will always result in higher current, higher temperature will always result in lower voltage, etc. IV Swinger 2 can be extremely useful with no calibration whatsoever.

For some users, however, the ability to calibrate to a higher level of accuracy is desirable. The Calibrate menu provides this ability.

Users who require the **highest degree of accuracy** should do the following:

- **Power the hardware** using a DC power supply<sup>3</sup>

Normally the Arduino is powered from the USB port. This bypasses the Arduino's voltage regulator, so it is only as accurate and stable as the laptop provides. When the Arduino is powered via its barrel jack or Vin pin (7 – 12 volts), it regulates that to a very accurate +5V and the USB power is not used. The IV Swinger 2 ammeter/voltmeter circuitry uses the power from the Arduino and can be more accurate with the regulated +5V.

NOTE: An improvement in the Vref (+5V) calibration (see section 4.3.5.1 below) introduced in software release v2.6.0 has substantially narrowed the gap between powering the Arduino via USB and powering it using a DC power supply. The software now essentially measures Vref on every IV curve, which mostly solves the issue of USB voltage variability. The advantage of using a separate DC supply is minimal, and it is certainly much less convenient.

- Use the **advanced voltage** and **advanced current** calibration options

The advanced calibration uses two points instead of just one for each calibration. This results in better accuracy across the whole range. The advanced current calibration can be performed with a bench DC power supply, which not only is more convenient but also much more controllable. As of v2.8.0, current calibration of EMR-based IV Swinger 2 hardware may be performed with a bench DC power supply (previously this only worked for the SSR/FET design.)

Accuracy of better than  $\pm 0.5\%$  across most of the range (i.e. excluding very low voltages and currents) can be achieved when all of the above guidelines are followed.

---

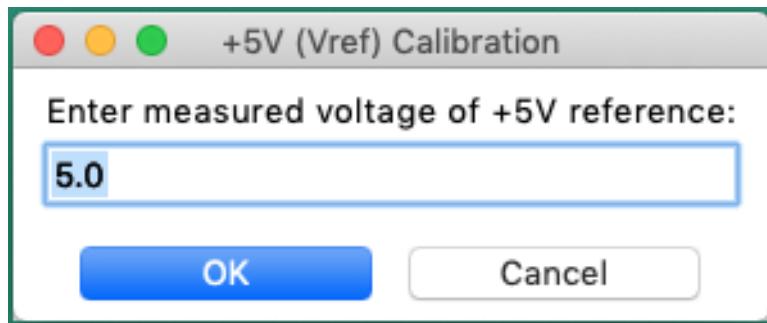
<sup>3</sup> Search for “Arduino power supply” on Amazon or elsewhere. Using a 9V battery is not recommended.

The Calibrate menu looks like this:

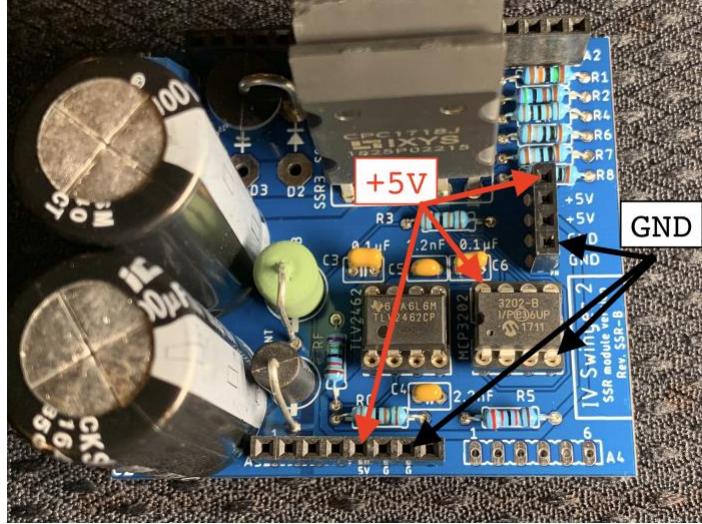


#### 4.3.5.1 Vref (+5V)

As mentioned above, using an external DC power supply results in the most accurate and stable measurements. Normally, however, the IV Swinger 2 hardware power comes from the laptop via the USB cable. This is much more convenient than using an external power supply. USB nominally provides +5V, but this voltage can vary from one laptop to another and even on the same laptop at different times. This voltage is used as the reference voltage (Vref) for the ammeter and voltmeter circuitry; if it is not exactly 5V, the measurements will be inaccurate until a Vref calibration is performed. As of software release v2.6.0 there is an algorithm that compensates for variations in the Vref voltage after the calibration, both due to using a different laptop and due to changes on the same laptop. The first item on the Calibrate menu, Vref (+5V) allows the user to input the measured value of the +5V Vref, which the software then uses in its calculations. When this menu item is selected, the following dialog comes up:



To measure this voltage, you need to have the lid off with the USB cable connected to the laptop. The IV Swinger 2 application should be running. Use a digital multimeter (DMM) to measure the voltage between pin 4 (V<sub>ss</sub>) and pin 8 (V<sub>DD/V<sub>REF</sub></sub>) of the MCP3202 chip. Alternately, the black DMM lead may be connected to the black binding post, which is GND. If your IV Swinger 2 is built with a PCB, there is a small female header on the board that has two ground pins and two +5V pins that may be more convenient (using male-to-male jumpers and alligator clip DMM probes). There are also +5V and GND pins on the Arduino power header. Measurement between any one of the GND points to any one of the +5V points is fine.



Enter the measured voltage in the box and click OK.

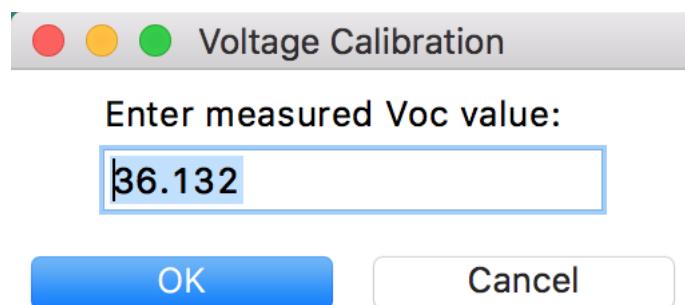
Starting with software release v2.6.0, the calibration is stored in non-volatile memory on the IV Swinger 2 device (Arduino EEPROM). This means that a calibrated IV Swinger 2 can be used by multiple laptops, and the results should be the same even if the Vref voltage is different. This is possible by comparing the Vref voltage to an internal (1.1V nominal) reference voltage in the Arduino. The internal reference voltage differs slightly from one Arduino to another, but is very stable on each and does not depend on the +5V Vref. The Vref (+5V) calibration is technically now a misnomer; it is actually determining the precise value of the Arduino's internal (1.1V) reference voltage, which is then used to measure the +5V Vref voltage just before swinging each IV curve<sup>4</sup>.

If you do use an external power supply, the voltage should be very close to 5.0 V. It is still a good idea to measure it to make sure.

#### 4.3.5.2 Voltage – basic

The “Voltage – basic” item on the Calibrate menu allows you to “correct” the open circuit voltage ( $V_{oc}$ ) value of a given IV curve with the value that is measured with a digital multimeter (DMM). The curve will be regenerated with  $V_{oc}$  calibrated to the measured value, and future curves will be generated using the new calibration. This is a 1-point calibration, and has been supported since the original v2.0.1 release.

When the “Voltage – basic” item is selected, the following dialog comes up:



<sup>4</sup> Thank you to Leland Mayne for this enhancement!

You then overtype the value with the measured value and press OK.

Note that the “Voltage – basic” menu item is only available immediately after you have swung an IV curve. It is not possible to perform a calibration using an older curve being viewed with the Results Wizard.

**The Calibration Help menu item has detailed instructions on how to perform the basic voltage calibration with a DMM. Those instructions will not be repeated here, but are important to follow carefully.**

The voltage calibration value is stored in non-volatile memory on the IV Swinger 2 device (Arduino EEPROM). This means that a calibrated IV Swinger 2 can be used by multiple laptops, and the results should be the same.

#### **4.3.5.3 Current – basic**

The “Current – basic” item on the Calibrate menu allows you to “correct” the short circuit current ( $I_{sc}$ ) value of a given IV curve with the value that is measured with a digital multimeter (DMM). The curve will be regenerated with  $I_{sc}$  calibrated to the measured value, and future curves will be generated using the new calibration. The basic Current Calibration dialog is very similar to the basic Voltage Calibration dialog. This is a 1-point calibration, and has been supported since the original v2.0.1 release.

**The Calibration Help menu item has detailed instructions on how to perform the basic current calibration with a DMM. Those instructions will not be repeated here, but are important to follow carefully.**

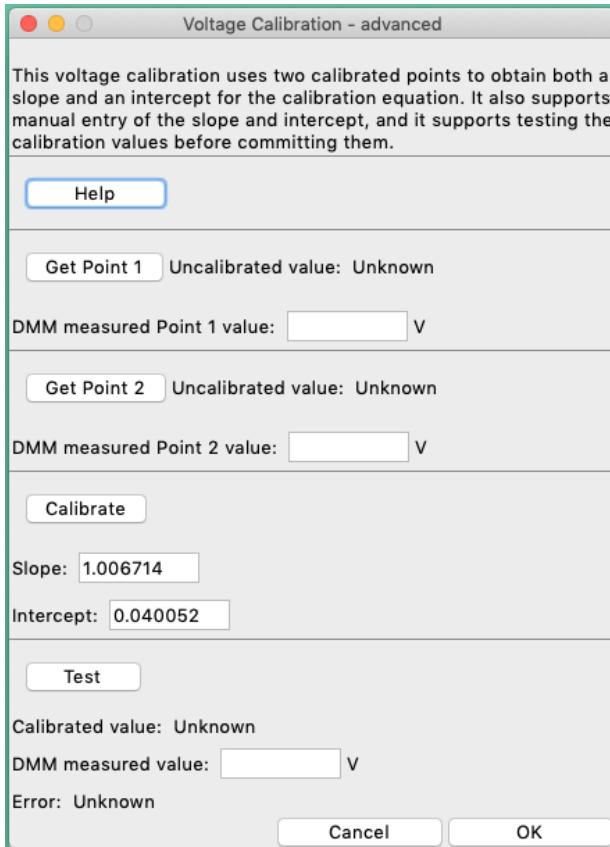
The current calibration value is stored in non-volatile memory on the IV Swinger 2 device (Arduino EEPROM). This means that a calibrated IV Swinger 2 can be used by multiple laptops, and the results should be the same.

#### **4.3.5.4 Voltage – advanced**

The “Voltage – advanced” item on the Calibrate menu allows you to perform a more accurate 2-point voltage calibration. Supported for advanced calibration was added in the v2.5.0 release.

A bench DC power supply is required to perform this calibration. This is because the  $V_{oc}$  of a PV module or cell is not controllable. The DC power supply should be able to generate voltages up to about 80% of the maximum  $V_{oc}$  that you expect to be measuring.

The dialog that comes up when you select this menu item looks like this:



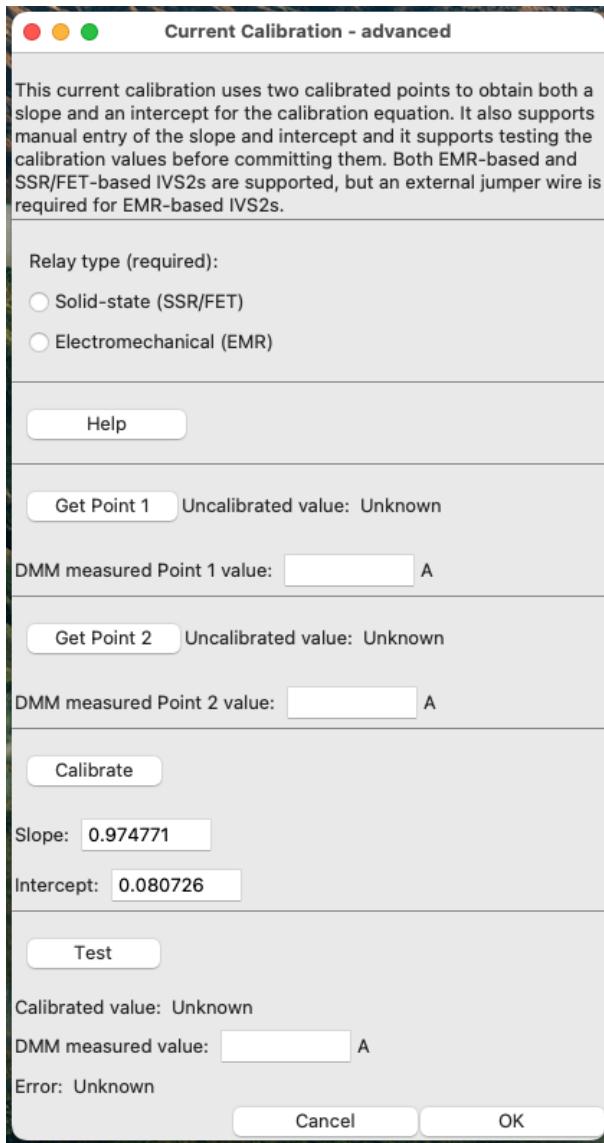
**Click on the Help button for detailed instructions on how to perform the advanced voltage calibration with a DMM and DC power supply. Those instructions will not be repeated here, but are important to follow carefully.**

Note that the Test button feature in this dialog may be used at any time, and can be useful to determine if a calibration is necessary.

#### 4.3.5.5 Current – advanced

The “Current – advanced” item on the Calibrate menu allows you to perform a more accurate 2-point current calibration. Supported for advanced calibration was added in the v2.5.0 release.

The dialog that comes up when you select this menu item looks like this:



The first thing you must do is select which type of relay the IV Swinger 2 hardware has: SSR/FET or EMR.

The hardware can (and should) be calibrated with a bench DC power supply. EMR-based hardware requires an external jumper (see section 4.3.5.5.1 below).

**Click on the Help button for detailed instructions on how to perform the advanced current calibration with a DMM and DC power supply. Those instructions will not be repeated here, but are important to follow carefully.**

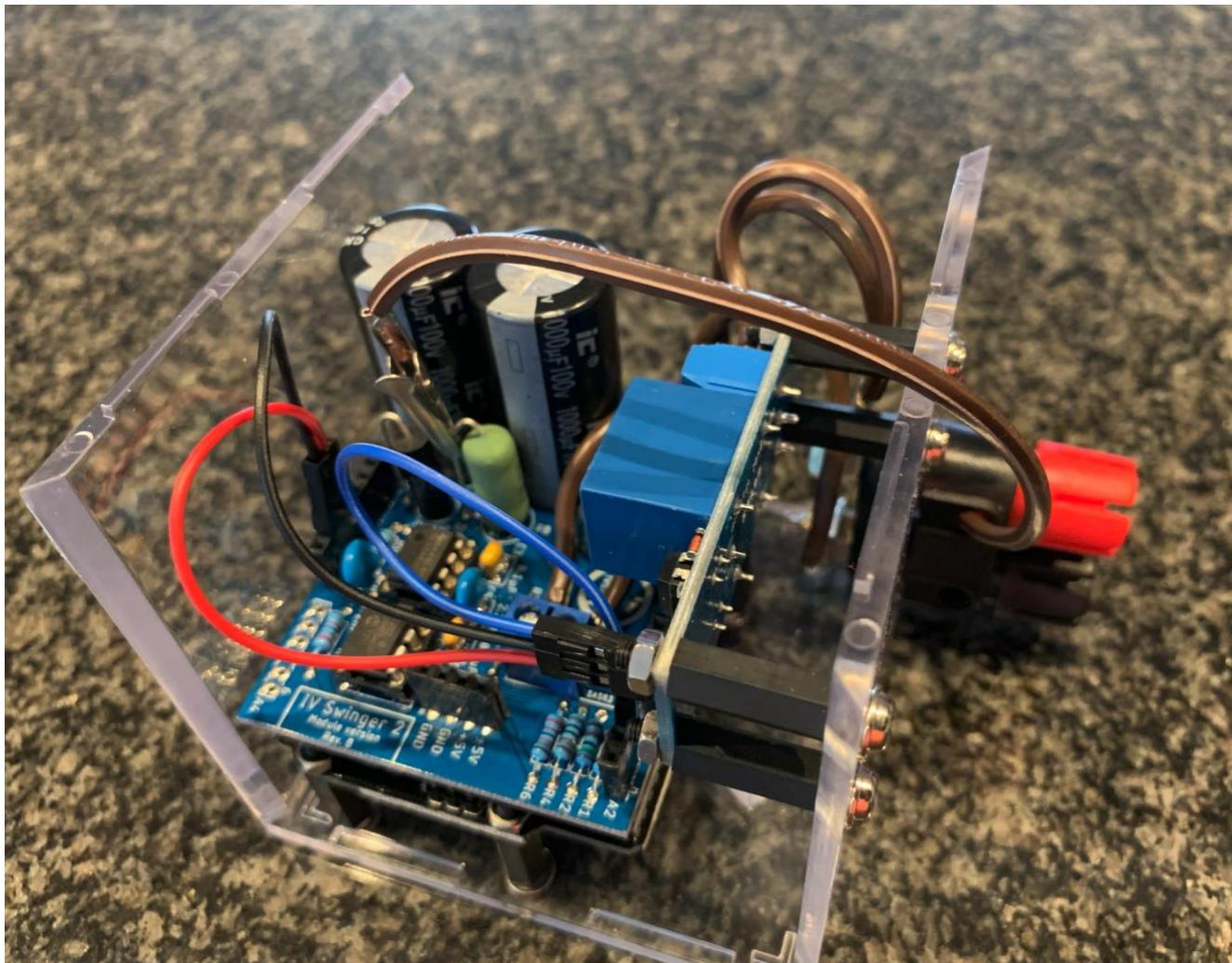
Note that the Test button feature in this dialog may be used at any time<sup>5</sup>, and can be useful to determine if a calibration is necessary.

---

<sup>5</sup> The external jumper must be connected if the hardware is EMR-based.

#### **4.3.5.5.1 EMR advanced current calibration jumper**

The EMR design has no way for software to create a path around the load capacitors as do the SSR and FET designs, so it requires the manual connection of a jumper wire. **This jumper is connected between the red binding post (PV+) and the exposed leg of the bleed resistor (Rb).** This wire needs an alligator clip on at least one end, to connect to the Rb lead. The other end can be screwed into the hole on the side of the red binding post (or an alligator clip can be used to clip it to the post.) Here is a photo:

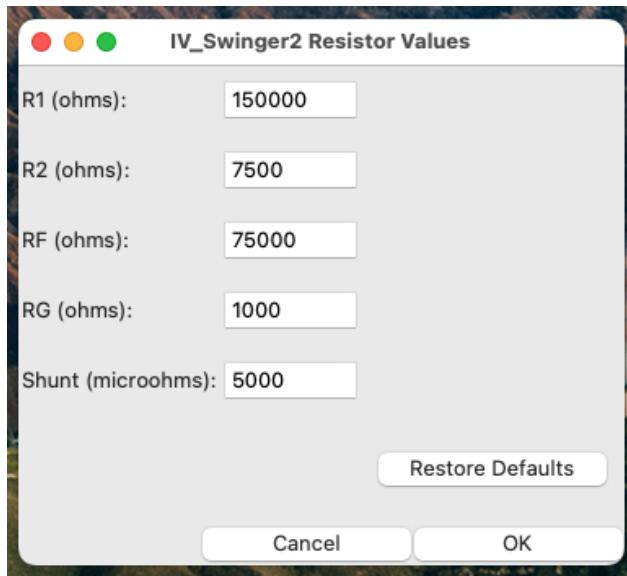


The exposed leg of Rb is the LOAD\_CAP- signal, which is also connected to the top of the shunt resistor. That creates a path between PV+ and PV- that bypasses the load capacitors but goes through the shunt. It is safe for this path to carry 10A of current (or more) indefinitely.

Of course, this jumper must be disconnected after the calibration, before swinging IV curves.

#### 4.3.5.6 Resistors

When the Resistors menu entry is selected, a dialog that looks like the following comes up:



This allows the precise measured values for these resistors to be specified. Also, some IV Swinger 2 variants use resistor values that are different in order to “scale up” or “scale down” the voltage and/or current range, and this allows those values to be specified.

The resistor values are all stored in non-volatile memory on the IV Swinger 2 device (Arduino EEPROM).

#### 4.3.5.7 Pyranometer

The pyranometer (irradiance sensor) is one of the optional sensors described in the “[IV Swinger 2: Optional Environmental Sensors](#)” document. Refer to that document for more details, including calibration. Note that this calibration value is stored on the laptop, not on the IV Swinger 2 device. Ideally, it would be stored on the pyranometer hardware, but there is nowhere to store it.

#### 4.3.5.8 Bias Battery

The “Bias Battery” calibration is used only for the cell version of IV Swinger 2, which sometimes requires a bias battery in series with the PV cell. See Chapter 9 on page 66 for more information.

#### 4.3.5.9 Invalidate Arduino EEPROM

The “Invalidate Arduino EEPROM” menu entry can be used to clear out the calibration values and start from the default calibration. This applies only to the Vref, Current, Voltage, and Resistor values.

#### 4.3.5.10 Calibration Help

The Calibration Help item brings up a dialog with more information on how to perform calibrations.

### 4.3.6 Instances Menu

The Instances menu was introduced in v2.8.0. Initially, it has only two entries:



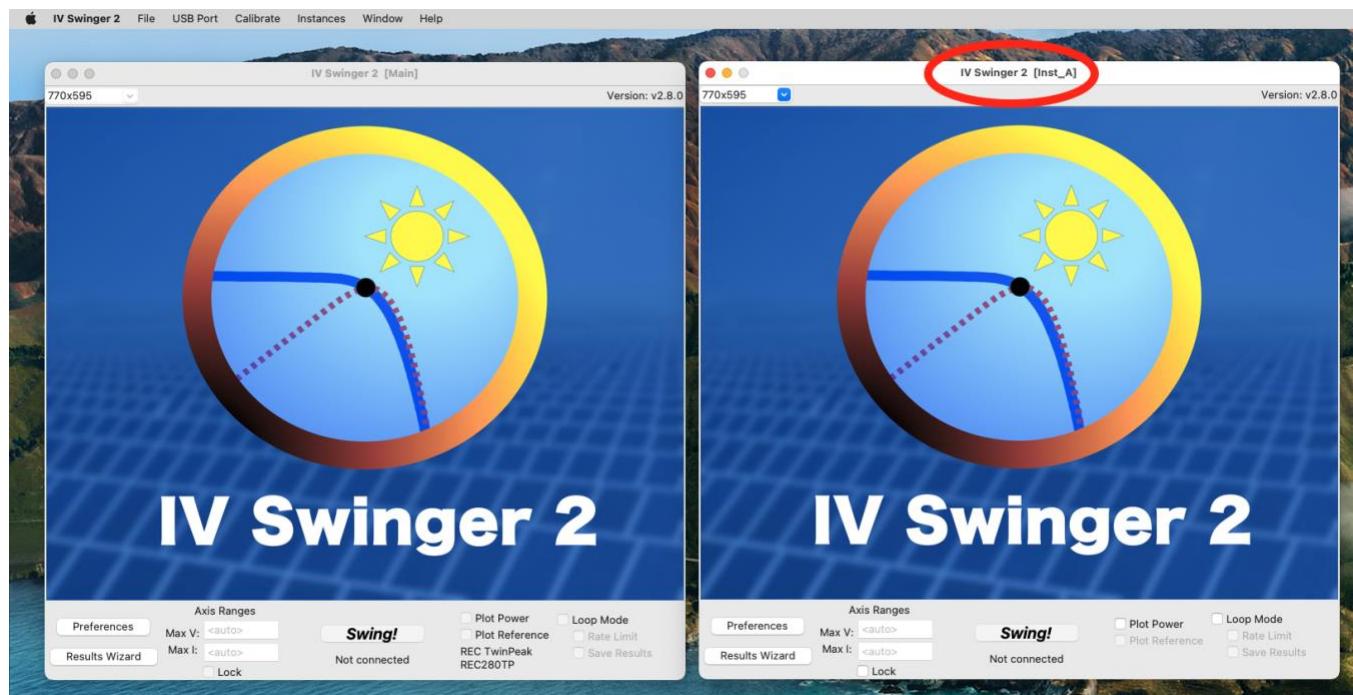
The "instances" feature provides the ability to concurrently control more than one IV Swinger 2 from a single laptop. Previously this required running the application from multiple user accounts.

When an instance is created, a dedicated subdirectory (folder) is created for its configuration and data (see section 4.6.5 on page 61.) This prevents runtime collisions and conflicts and also makes it much easier for the user to identify which results are for which PV module or cell under test.

To create a new instance, select "Add New Instance" in the menu. This will prompt you to enter a name for the instance:



Clicking OK will launch a new GUI window<sup>6</sup> to control that instance:



<sup>6</sup> The main GUI window and all instance GUI windows all run under the same process (PID).

Mac note: The menu bar looks identical for all GUI windows, but the menu bar is associated only with the window that has focus (is selected.) For this reason, it is important to click on the correct window before using the menu or else the menu actions may be applied to the wrong one. This is not an issue on Windows because each window has its own menu bar.

The USB port for a new instance must be assigned manually from the USB port menu. Each GUI window must have a different USB port selected in order to prevent contention issues. The ports that are already in use will be disabled (grayed out) in the USB Port menu. If a previously created instance which uses a USB port that is already in use is launched, it will come up with its USB port set to DISCONNECTED.

Instances that have been previously created will be listed in the menu and can be launched by selecting those menu entries:



The main GUI window remains active even when one or more instance GUI windows are active. Killing an instance GUI window doesn't affect the other GUIs, but killing the main GUI kills all GUIs. If the main GUI is not going to be used to control an IV Swinger 2, its USB port should be set to DISCONNECTED in the USB Port menu.

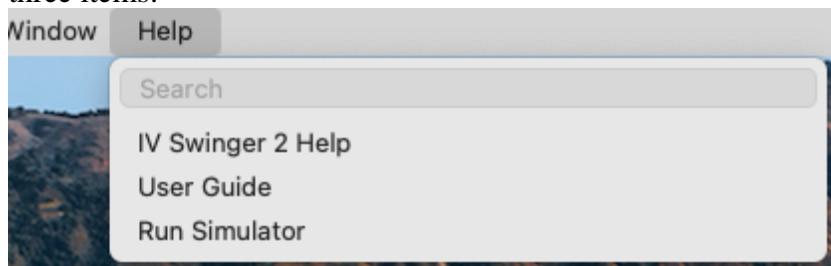
Each instance has its own preferences that are initially the defaults - they are not inherited from the main GUI preferences or from any other instance.

The Results Wizard for each instance lists that instance's runs only.

Each instance GUI generates its own log file, viewable from its File menu.

#### 4.3.7 Help Menu

The Help menu has three items:



- “IV Swinger 2 Help” brings up a dialog with a very coarse overview of how to use the application.
- “User Guide” opens the “IV Swinger 2 User Guide PDF”

- “Run Simulator” helps choose components for “scaled” versions of the hardware that are capable of measuring IV curves for PV modules or cells that are smaller or larger than the standard design supports. The vast majority of users will have no reason to use it. There is another document entitled “[IV Swinger2: Hardware Scaling](#)” that describes how to use the simulator.

## 4.4 Results Wizard Dialog

The Results Wizard dialog is shown in Figure 4-4 below. It provides the tools for viewing results of previous runs, combining multiple curves on the same plot, modifying their title and appearance, copying them to a USB drive (or elsewhere), deleting them, etc.

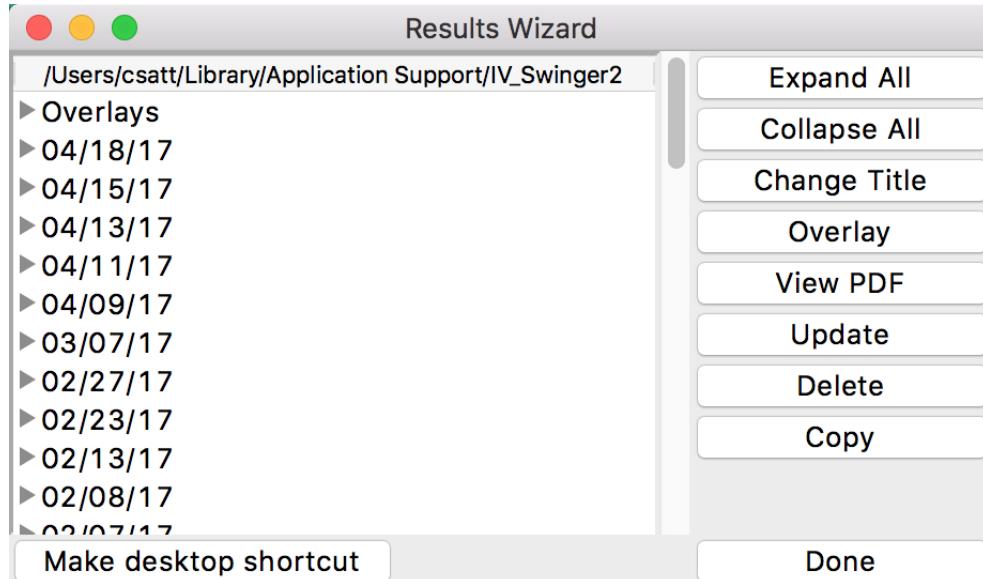


Figure 4-4: Results Wizard Dialog

### 4.4.1 Expanding and Collapsing Date Groups

The left-hand side of the dialog is a “tree view” that groups results by the date they were captured. Each date group can be expanded or collapsed (by default all come up collapsed, as above). To expand a date group, click on the small triangle (plus sign in Windows) to its left, for example:

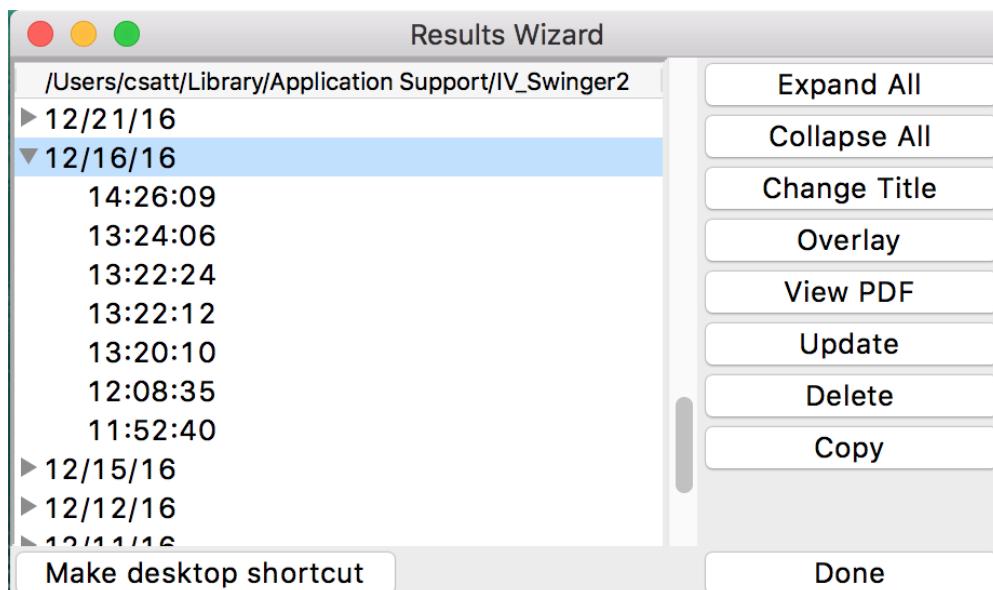


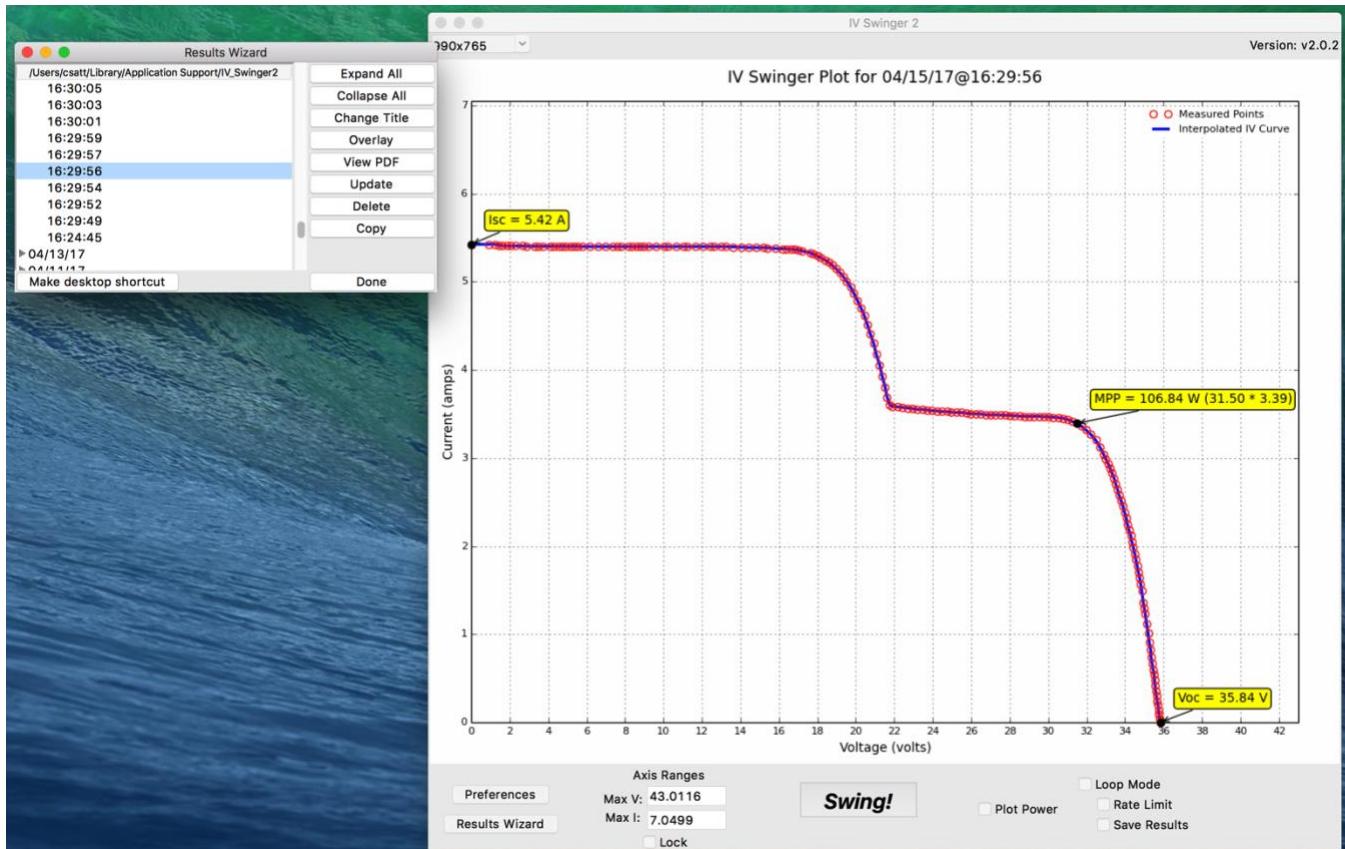
Figure 4-5: Expanding a Date Group

Within each date group are the runs that were captured on that date, labeled by the time that they were captured. Note that all runs (and date groups) are sorted from newest to oldest.

The Expand All and Collapse All buttons at the upper right expand and collapse all date groups at once.

#### 4.4.2 Selecting Runs

Click on a single run to select it. The IV curve will be displayed in the main window:



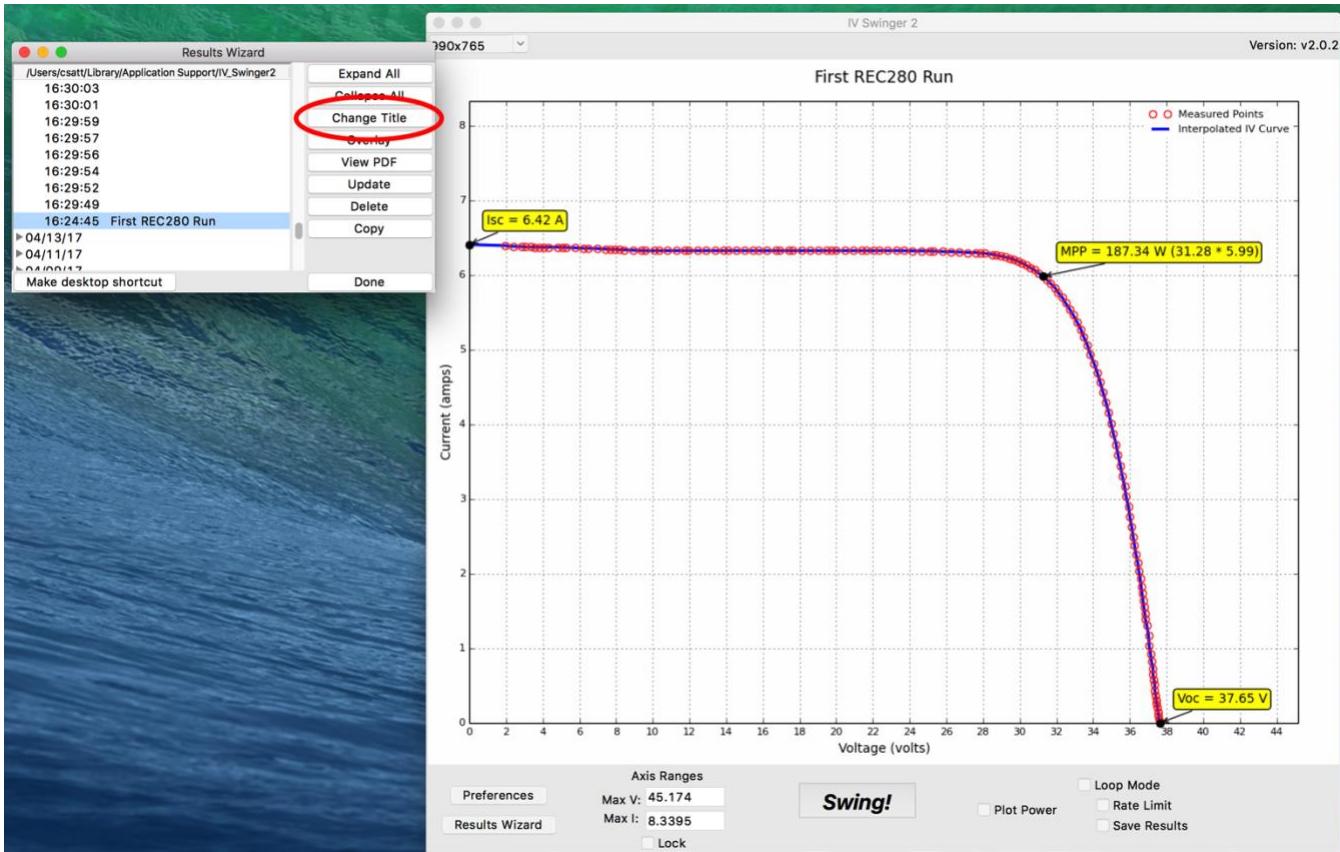
**Figure 4-6: Selecting a Run**

The up and down arrow keys can be used to move up and down the list. This can be very useful to quickly “flip through” the results. In fact, it can be used to create a sort of “animation” if there’s a series of runs that were generated with the axes locked.

Some actions support multiple selected runs. Use “Shift-click” to select a contiguous range of runs and “Control-click” to select non-contiguous runs, just as you would in other Mac and Windows applications. Note that when multiple runs are selected, the oldest one is displayed.

#### 4.4.3 Changing the Title

The default title at the top of each graph is “IV Swinger Plot for mm/dd/yy@hh:mm:ss”. This can be changed to something more descriptive. Select a single run and click the Change Title button. This opens a dialog where you can change the title. The new title is also listed in the tree view:



**Figure 4-7: Changing the Title of a Run**

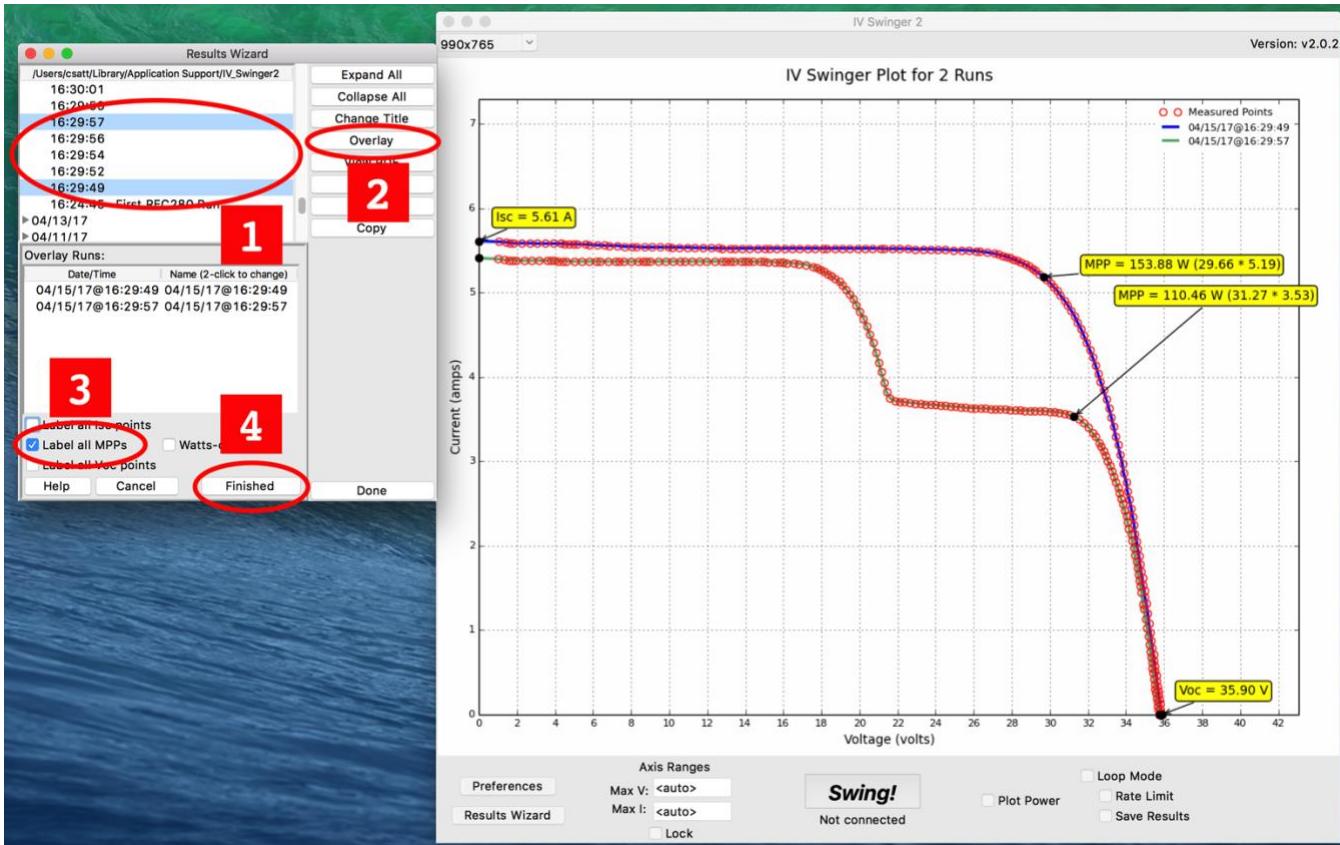
#### 4.4.4 Overlaying Multiple Runs

Combining multiple IV curves onto a single graph can be very useful for seeing their differences. The Results Wizard supports overlaying up to eight IV curves on the same graph.

The basic sequence for using this feature is:

1. Select up to 8 runs (using shift-click and/or control-click)
2. Click on the Overlay button
3. Change plotting options, if desired
4. Click the Finished button

These steps are illustrated in Figure 4-8 below. This screenshot was taken between step #3 and step #4. Notice that the dialog has now sprouted the section in the lower left corner containing a list of the overlay runs, some option check boxes, and Help, Cancel, and Finished buttons.



**Figure 4-8: Creating an Overlay**

#### 4.4.4.1 Changing Overlay Plotting Options

The overlay example shown in Figure 4-8 shows one simple change to the plotting options made in step #3: the “Label all MPPs” box is checked. Many other changes may be made at this point (in addition to the other similar check boxes):

- The title can be changed by clicking on the Title button
- The order can be changed by dragging and dropping the runs listed in the “Overlay Runs” box (or by clicking on the Date/Time heading to sort chronologically)
- The names of the runs on the graph can be changed by double-clicking on them in the “Overlay Runs” box
- The Plot Power box (in the main window) can be checked or unchecked
- Many of the Plotting preferences in the Preferences Dialog (see section 4.5 on page 48) can be changed (not the ADC corrections, Battery bias, or Series resistance compensation)

#### 4.4.4.2 Saving Overlays

When the Finished button is pressed, the overlay is saved. If the Cancel button is pressed, the overlay is discarded. The saved overlays are viewable in the “Overlays” group at the top of the tree view list in the Results Wizard.

**IMPORTANT:** Saved overlays are “final”. They cannot be “tweaked” later. Of course, you can always recreate them from scratch.

#### 4.4.5 Viewing the PDF

The graph shown in the main window is a GIF format image. A PDF of each graph is also generated for each run (or overlay), and this is much more suitable for printing. It is also possible to zoom into the PDF without the image becoming pixelated<sup>7</sup>. The View PDF button invokes the viewer that your computer is set up to use by default for PDF files (e.g. Preview on Mac and Acrobat Reader on Windows).

Once the PDF is open in the viewer, you can print it from there or do any of the other things that the viewer allows you to do (including saving it somewhere else).

#### 4.4.6 Batch Updates

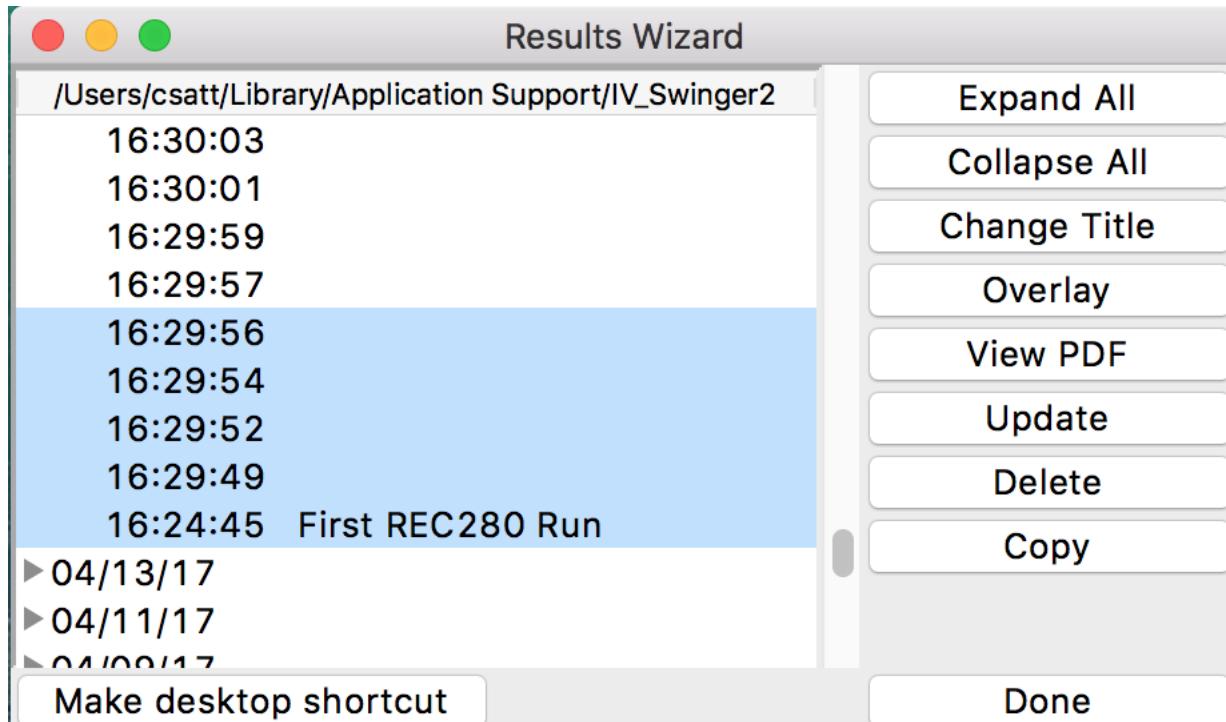
The Update button is used for updating a batch of runs to conform to the same set of plotting preferences. The sequence is as follows:

1. Select multiple runs
2. Change plotting option
3. Click Update button

The order of the first two steps is important.

For example, suppose you want to add the power curve to a batch of runs.

FIRST, you would select all of the desired runs (using shift-click and/or control-click):

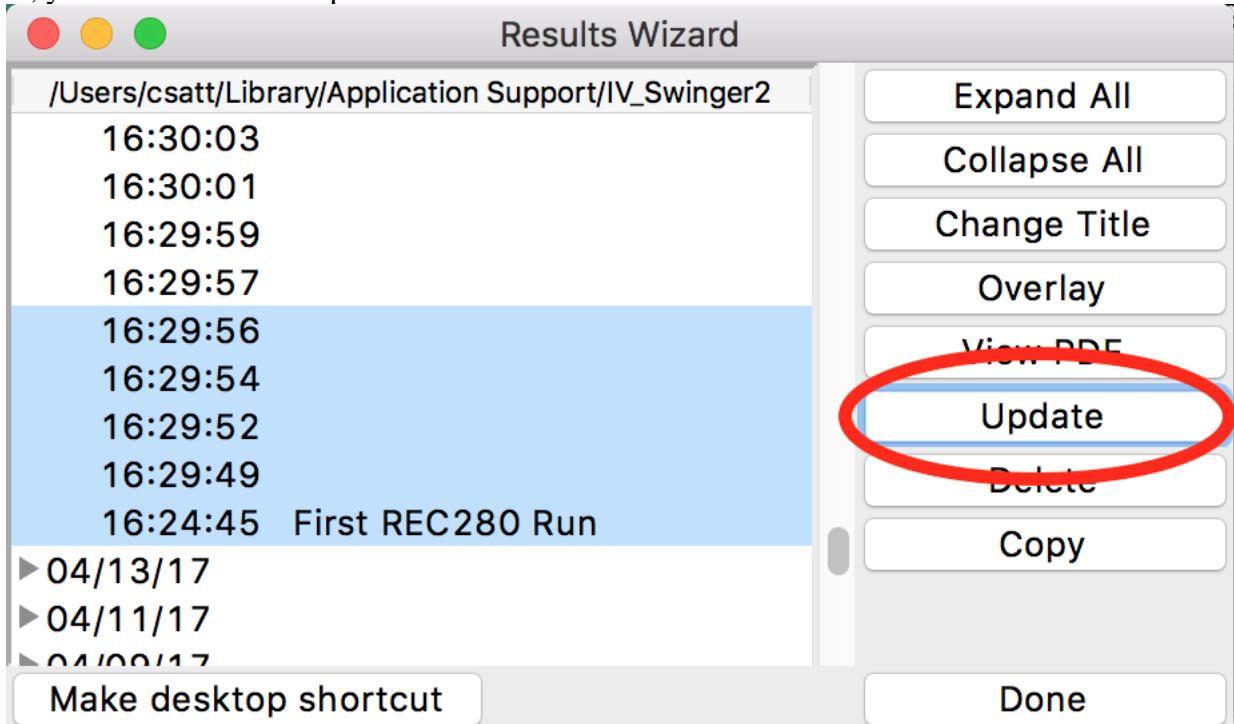


SECOND, you would check the Plot Power box:

<sup>7</sup> Unfortunately, for technical reasons the PDF cannot be used in the image pane of the main window.



LAST, you would click the Update button:



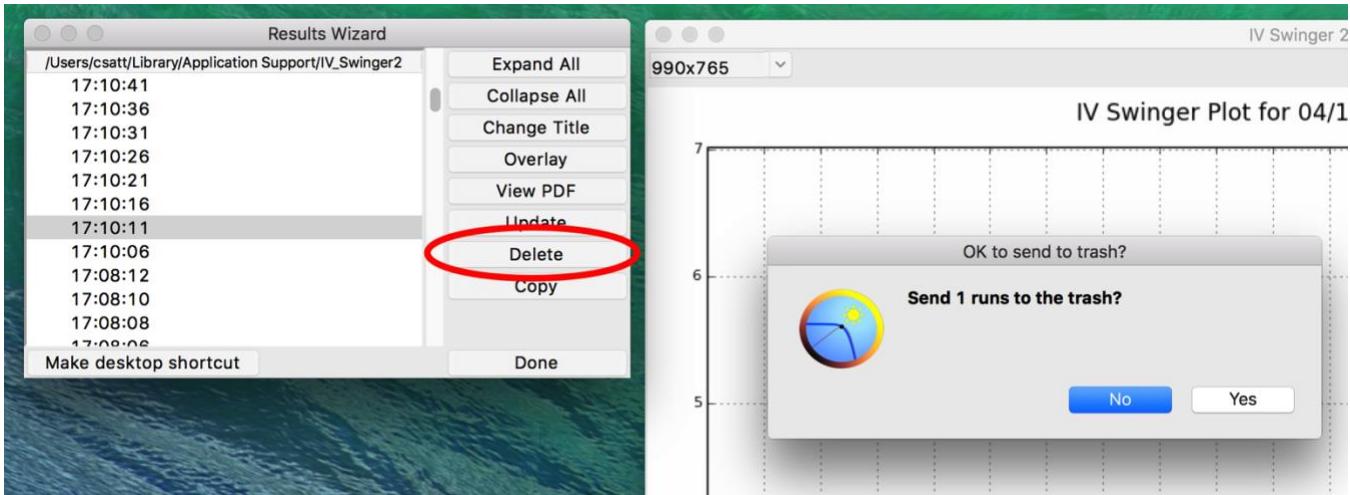
The batch update can also be used to redraw all of the selected runs using the same axis ranges. To do that, step #2 would be to check the Lock box in the Axis Ranges controls (and possibly type in the Max V and Max I values).

Similarly, the display image size can be updated for a batch of runs. To do that, step #2 would be to change the value in the Image Size control (see section 4.1.8 on page 20).

And finally, step #2 can be changing any of the Plotting preferences in the Preferences Dialog (see section 4.5 on page 48).

#### 4.4.7 Deleting Runs

To delete runs (actually send them to the trash), select them and then click the Delete button. This will bring up a dialog asking for your OK to move it/them to the trash:



#### 4.4.8 Copying Runs

To copy runs to a USB drive (or anywhere else), select them and then click the Copy button. This will bring up a standard Finder (Mac) or Explorer (Windows) dialog for you to select the destination for the copy. See Figure 4-9: Copying Runs to a USB Drive (Mac) and Figure 4-10: Copying Runs to a USB Drive (Windows) below for what this looks like on both platforms.

If it doesn't already exist, a folder named IV\_Swinger2 will be created at the selected destination, and the selected run or runs will be copied under that folder. Each run consists of multiple files in a folder named for the date and time of the run (see section 4.5.4 on page 55). If the IV\_Swinger2 folder already exists at the selected destination, the selected run or runs will be added to it.

Note that runs copied to a USB drive on a Mac may be viewed on a Windows machine and vice versa.

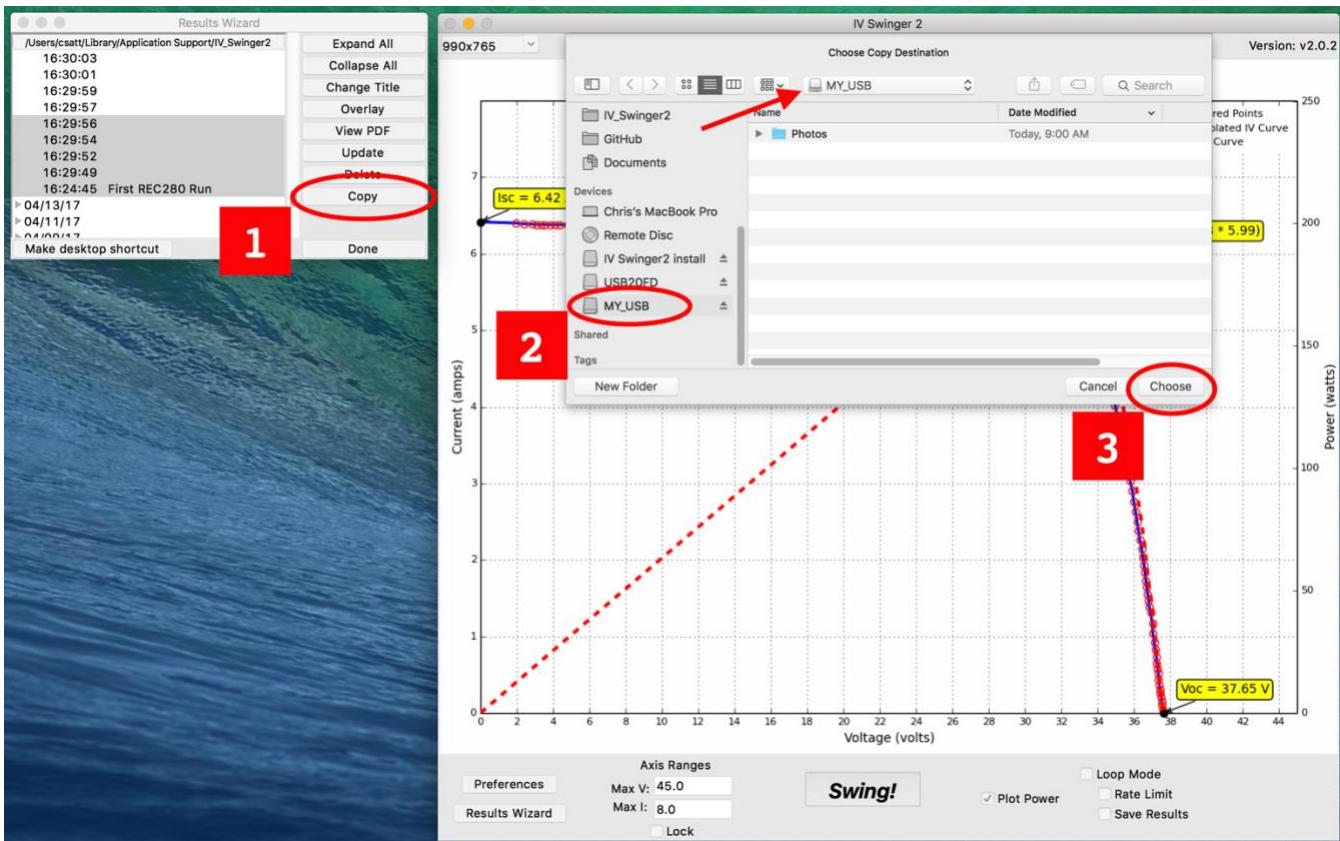
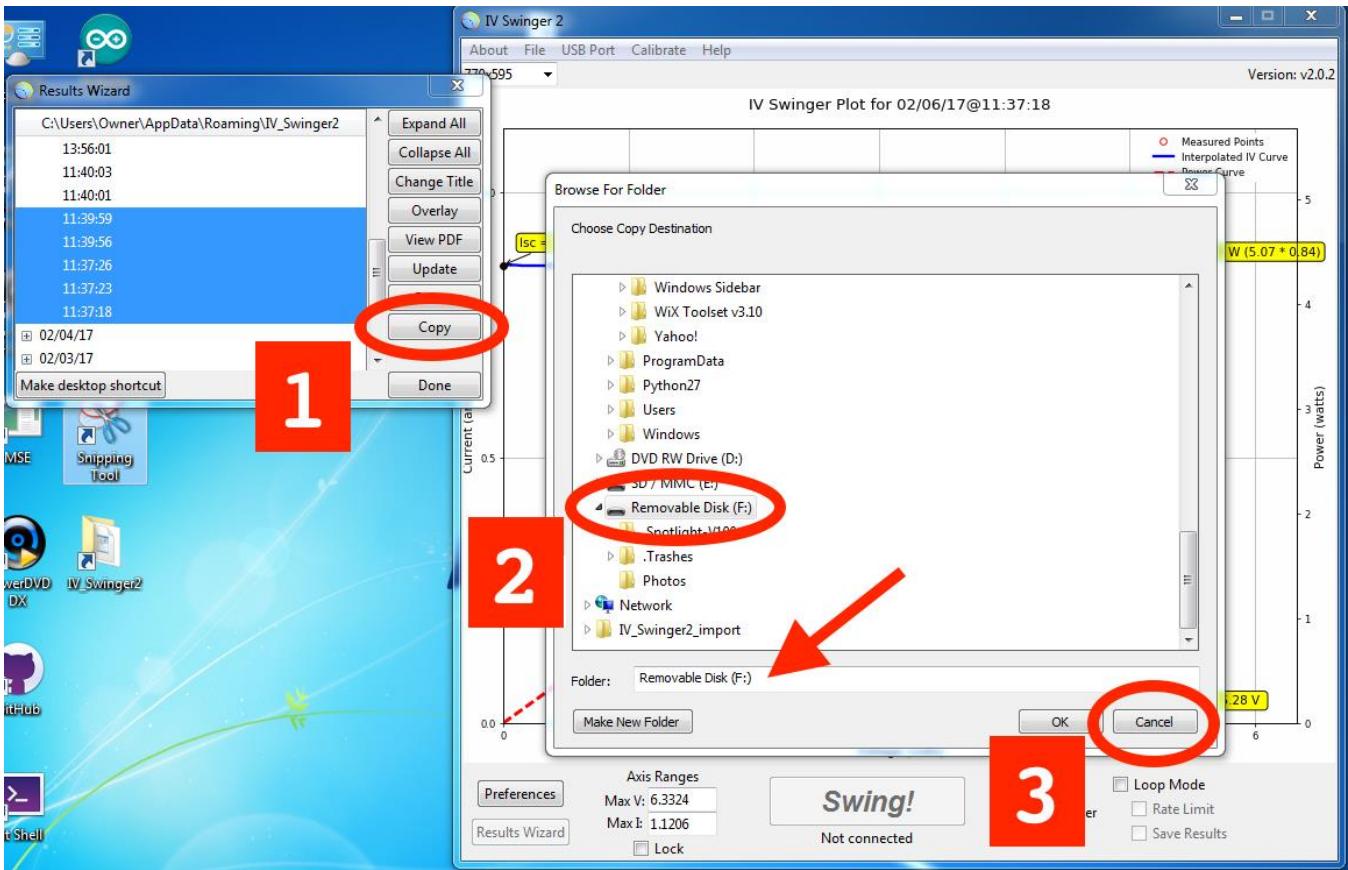


Figure 4-9: Copying Runs to a USB Drive (Mac)



**Figure 4-10: Copying Runs to a USB Drive (Windows)<sup>8</sup>**

#### 4.4.9 Changing Where to Look for Runs

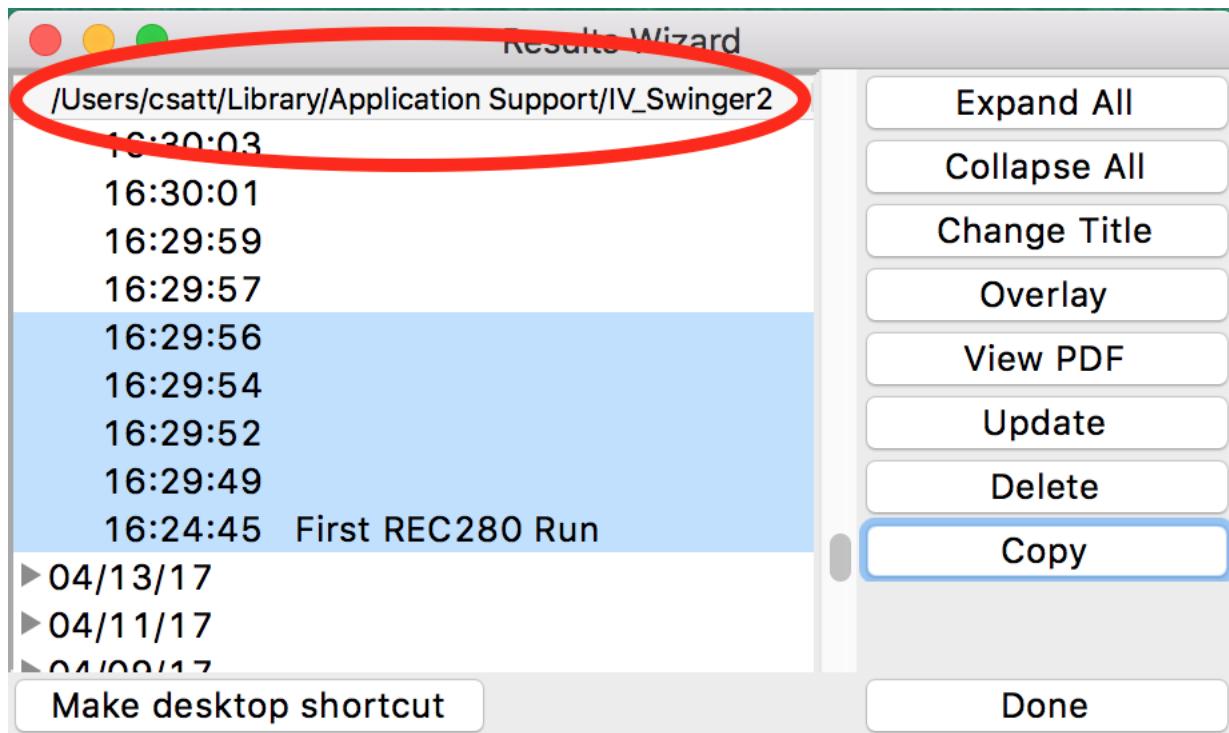
The runs are stored in a folder named IV\_Swinger2 in a standard, but somewhat difficult-to-find location on your computer. This location may be different depending on the version of MacOS or Windows that you are running. When you open the Results Wizard, this location is listed at top of the tree view pane (see Figure 4-11 below).

If you want to use the Results Wizard to browse runs on a USB drive (e.g. copied from a different computer), you can point it to that new location by clicking on the path at the top of the tree view pane. This will bring up a standard Finder (Mac) or Explorer (Windows) dialog for you to select the location of the results.

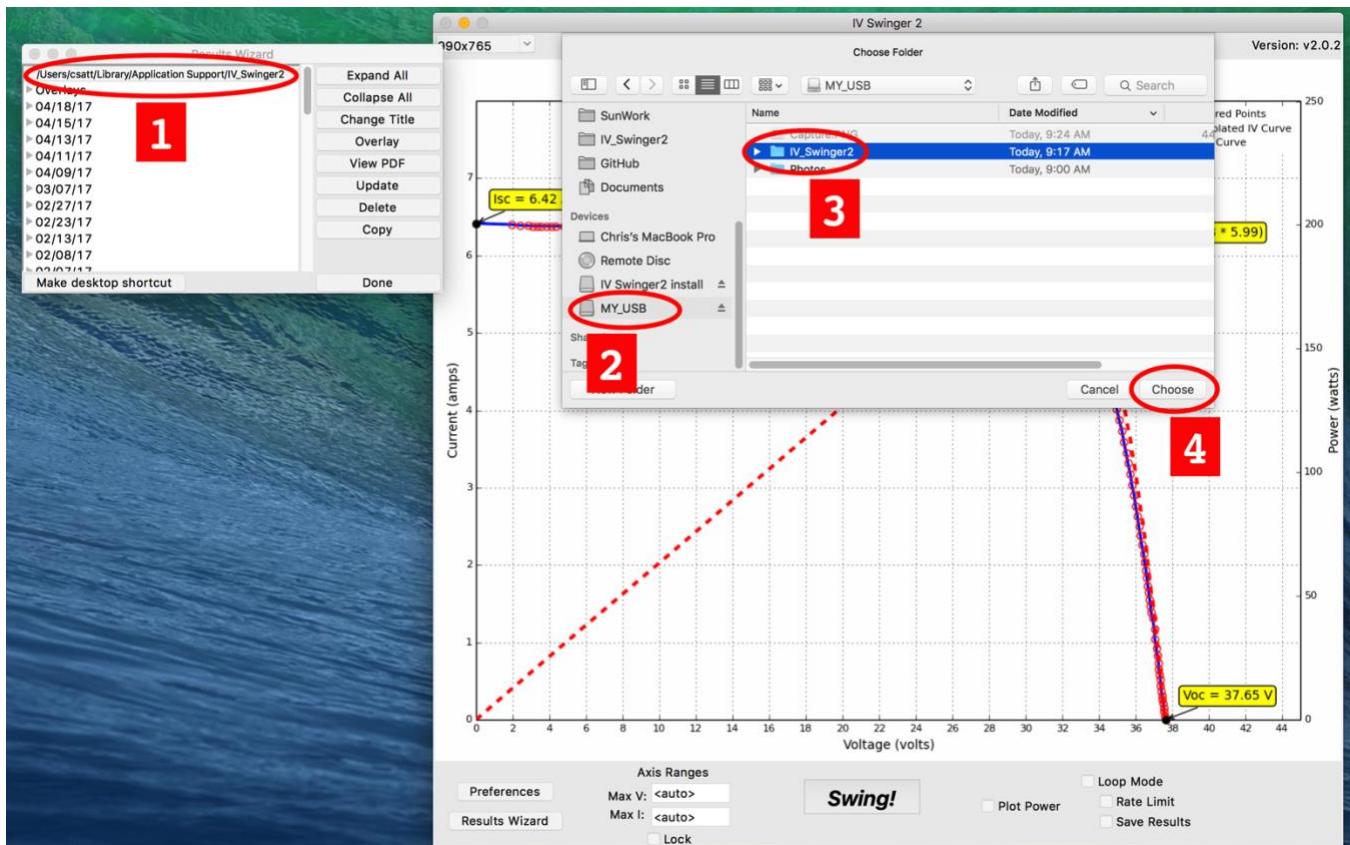
Figure 4-12 shows this process on a Mac. Windows is very similar. Figure 4-13 shows what it looks like after the path has successfully been changed to point to the USB drive.

---

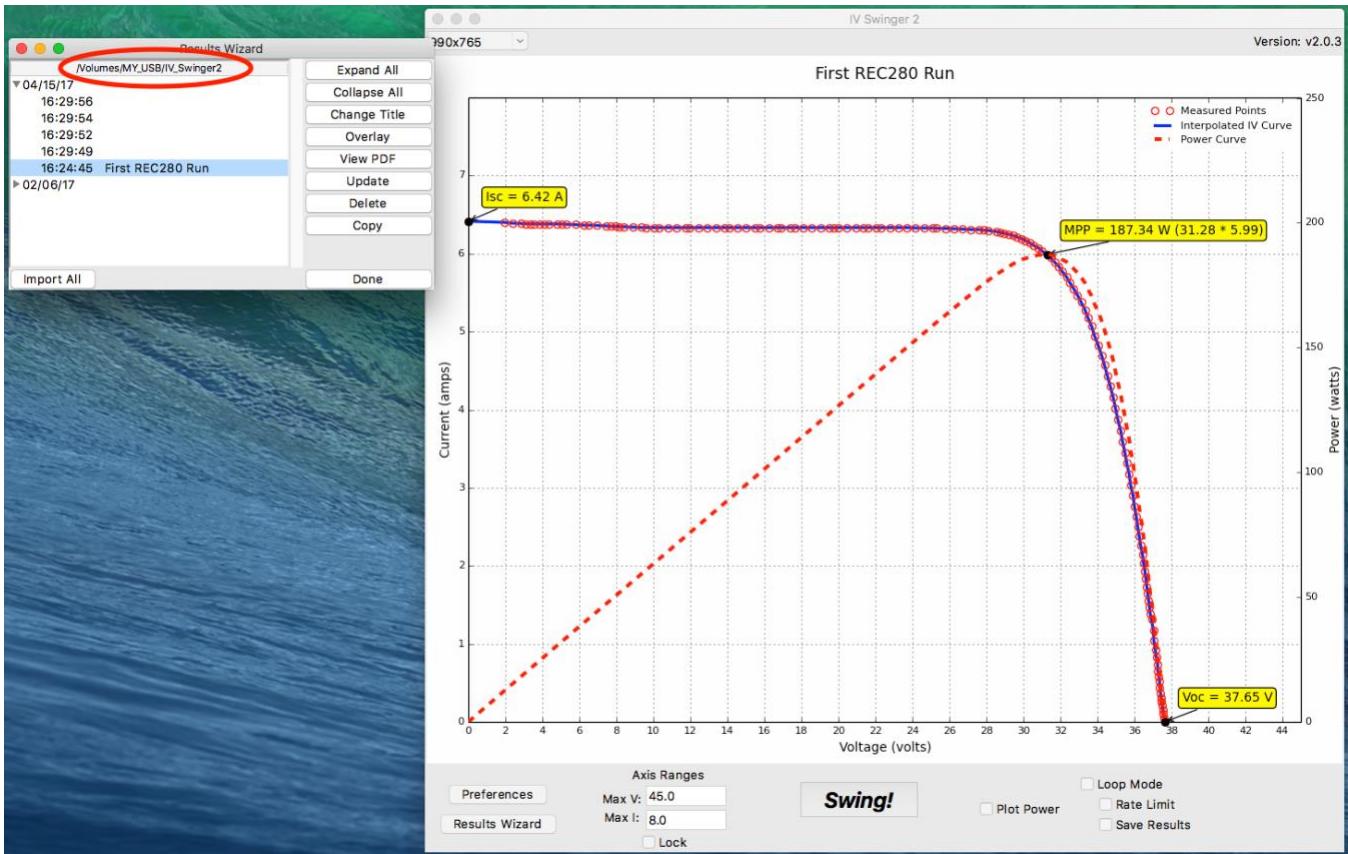
<sup>8</sup> Click OK, not Cancel!



**Figure 4-11: Path to Results Folder**



**Figure 4-12: Changing the Path to a USB Drive**



**Figure 4-13: Path Successfully Changed to USB Drive**

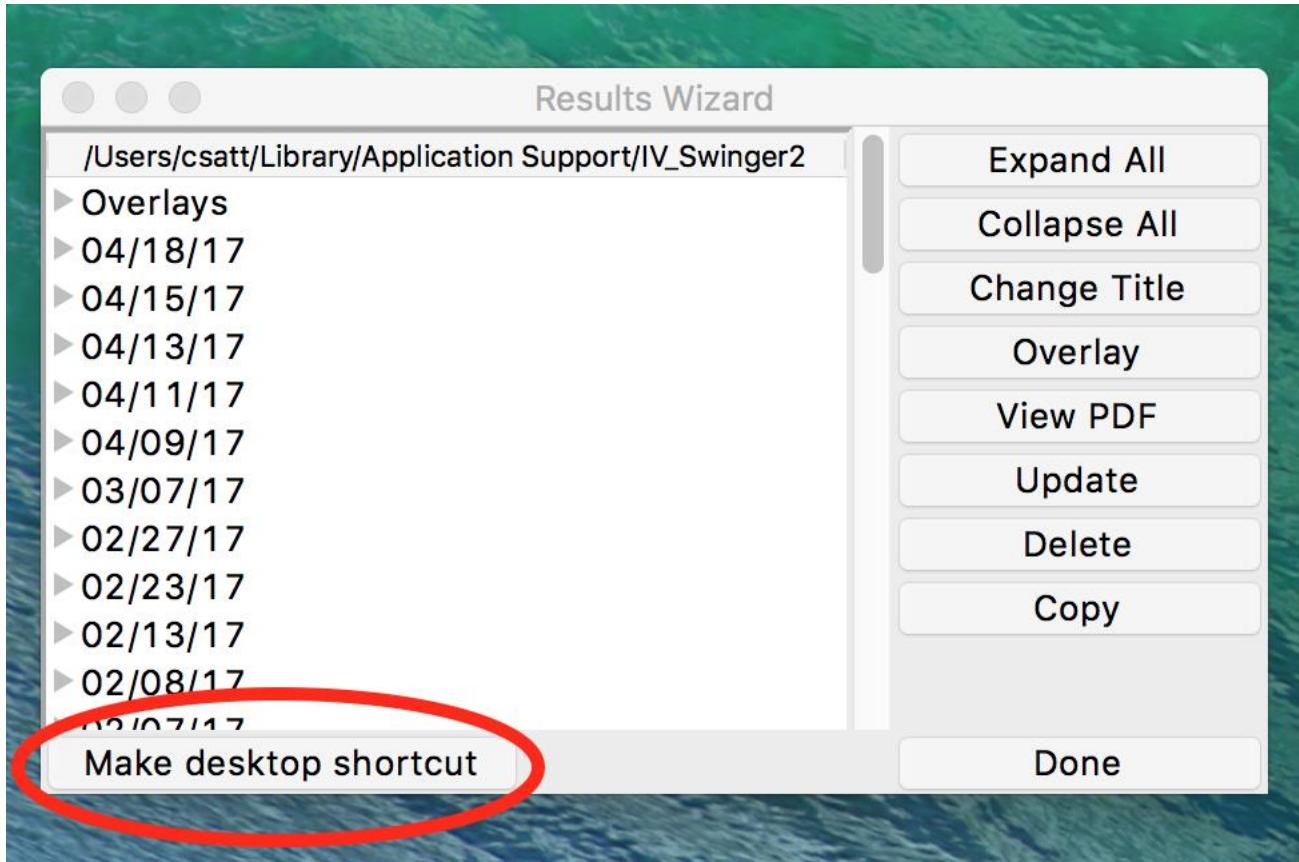
#### 4.4.10 Creating a Desktop Shortcut to Runs Folder

The Results Wizard provides the tools necessary for most browsing, modifying, combining, printing, etc. It “hides” the details of what files are included in the results folders.

However, some users may want to directly access the raw results folders and their files. For example, you may want to open the CSV file(s) in a spreadsheet program. You might also want to know how much disk space is being used, etc. As described in the previous section, the path to the results folder is shown at the top of the Results Wizard tree view pane. But this may be a “hidden” folder that is difficult to navigate to in Finder/Explorer.

The Make Desktop Shortcut button at the bottom of the Results Wizard can be used to create a desktop shortcut to the results folder to make it easier to locate in Finder/Explorer. The shortcut will be named IV\_Swinger2.

See section 4.5.4 on page 55 for a description of the files.



**Figure 4-14: Making a Desktop Shortcut to Raw Results**

#### 4.4.11 Importing Results from a USB Drive

Section 4.4.9 above describes how to view the results on a USB drive (or elsewhere). In this case, the button in the lower left corner of the dialog changes to an “Import” button, as shown below in Figure 4-15. Pressing this button will copy the selected results from the USB drive to the standard runs folder. If nothing is selected, it will copy ALL of the results. If any of the same runs or overlays already exist, a dialog comes up asking whether to overwrite all of them. If you answer “no”, then the import will skip all of the duplicates and only copy the different ones. If you answer “yes” the import will overwrite all of the duplicates as well as copying the different ones.

The typical use case for this is that Person A has captured results on a laptop and wants to share them with Person B (e.g. a lab partner). The first step is for Person A to copy the results to a USB drive (see section 4.4.8 on page 42). Next, Person A gives the USB drive to Person B, who imports the results to his/her laptop and then gives the USB drive back to Person A.

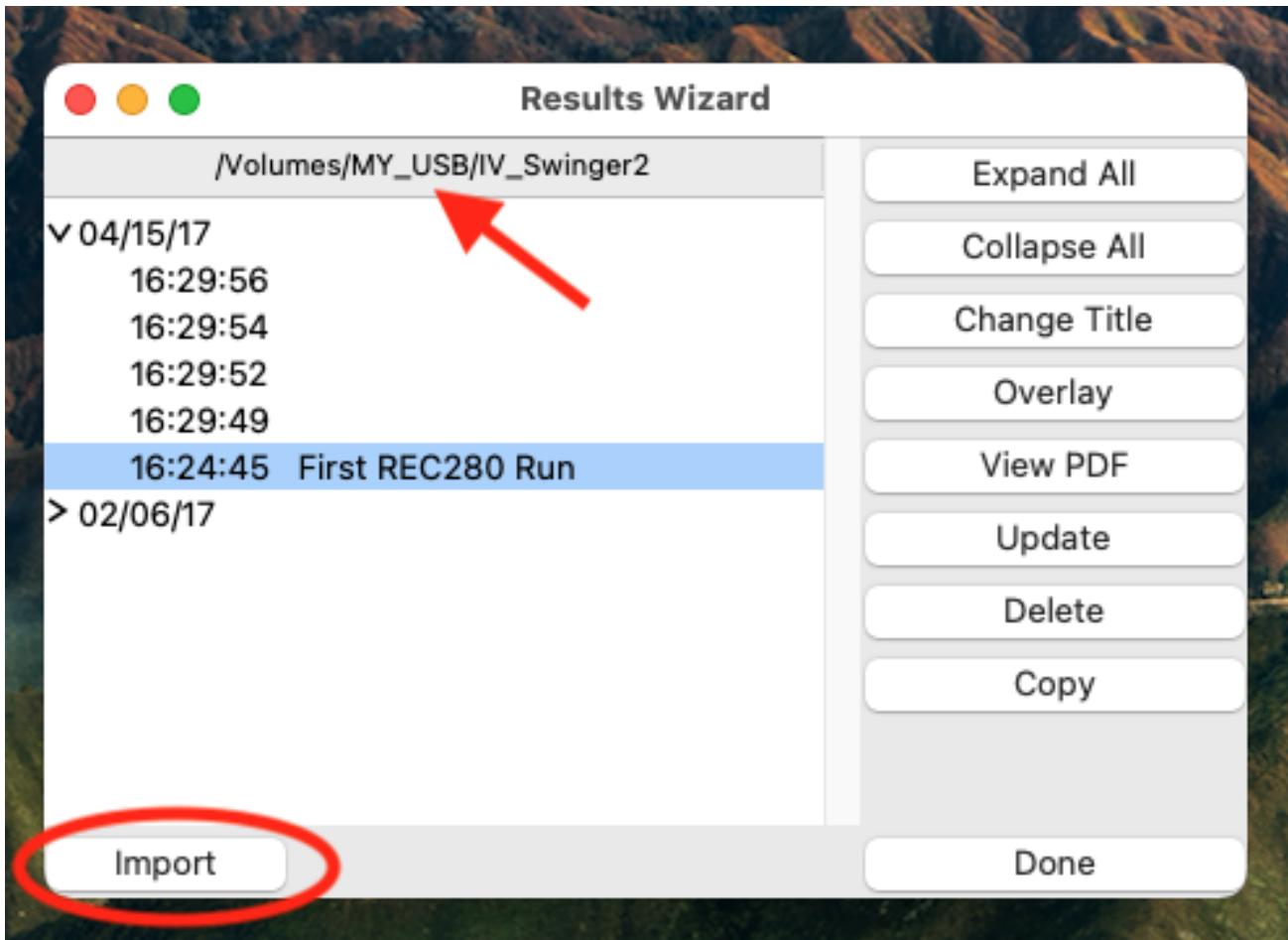
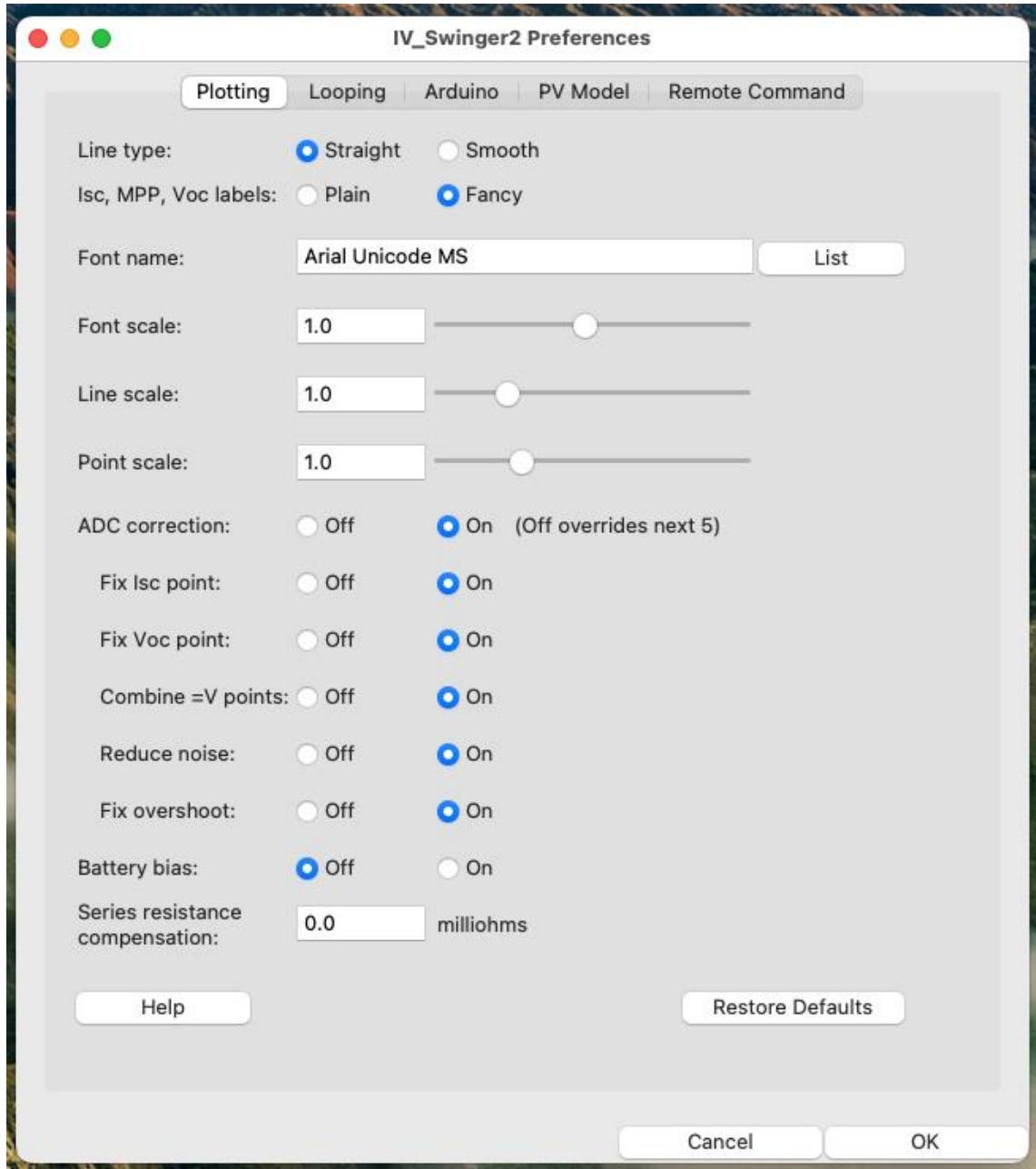


Figure 4-15: Importing Results from USB Drive

## 4.5 Preferences Dialog

Figure 4-16 below shows the dialog that comes up when the Preferences button is pressed. This is the Mac version but the Windows version is very similar. At the top are five “tabs”: Plotting, Looping, Arduino, PV Model, and Remote Command. The figure shows the Plotting tab selected.

Sections 4.5.1, 4.5.2, 4.5.3, 4.5.4, and 4.5.5 below discuss each tab.



**Figure 4-16: Preferences Dialog**

#### 4.5.1 Plotting Preferences Tab

Figure 4-16 above shows the Plotting tab.

The configuration options on the Plotting tab control the appearance of the IV curve plot, both on screen and in the generated PDF. Changes made to these options are applied immediately to the current plot (if there is one), so you can experiment and see the effect of your changes. Clicking OK will save the new values, and they will be used from that point forward. Clicking Cancel will revert to the previous values. These changes only affect the post-processing of the raw measurements from the hardware, which are never changed. Following are brief explanations of each option.

#### **4.5.1.1 Line type**

The curve interpolated between the measured points may either be "Straight" or "Smooth". The default is "Straight" because most IV Swinger 2 runs have so many points that there is virtually no advantage to a smooth curve, which takes longer to generate. When "Smooth" is selected, Catmull-Rom spline interpolation is used<sup>9</sup>.

#### **4.5.1.2 Isc, MPP, Voc labels**

Two styles are available for the labels: "Plain" and "Fancy". Plain labels are text only. Fancy labels are enclosed in a box with a yellow background, and there are arrows indicating the point.

#### **4.5.1.3 Font Name**

The name of the font to use for all text (title, legends, labels, etc.) The default is "Arial Unicode MS" which supports character sets for many languages. If that font is not available, the plotting will SILENTLY revert to a default font. The "List" button generates a list of fonts in a dialog from which you may copy and paste. The list is also written to the log file. However, it is not guaranteed that all of those fonts will behave correctly. Some (Wingdings, for example) may even mess things up to the point where the application must be closed and reopened if they are used.

#### **4.5.1.4 Font scale, Line scale, and Point scale**

The default font size, line thickness, and size of the "dots" indicating the measured points are chosen automatically based on the size/resolution of the graph. These options allow the user to scale those up or down, according to taste. The sliders can be used, or the values can be typed in manually. If Line scale is set to 0.0, the interpolated curve is not plotted. If Point scale is set to 0.0, the measured points are not plotted.

#### **4.5.1.5 ADC correction**

By default, several "corrections" are made to the raw values read by the analog to digital converter (ADC) chip to correct for noise and other effects that affect the proper rendering of the IV curve and the calculation of the maximum power. If "Off" is selected, these corrections are not performed. Note that the values written to the ADC CSV file are the uncorrected values, regardless of this setting. Also note that the calibration settings are used regardless of this setting.

Additionally, there are individual controls for each of the ADC corrections. These options are only relevant if the "ADC correction" option is "On".

##### **4.5.1.5.1 Fix Isc point**

If "ADC correction" is "On", this controls whether the first point of the curve should be modified (or even removed in some cases). This point is the Arduino code's attempt at approximating the Isc point, but its simplistic algorithm often gets it wrong. When this control is "On", the Isc point is removed if the

---

<sup>9</sup> The original IV Swinger captured only about 30 points, so this was important. The feature is still in the code.

first measured point after it has a voltage value more than 20% of the Voc voltage value (in which case implying that we know Isc is misleading). Otherwise, it is extrapolated from the beginning of the curve using a more sophisticated algorithm than the Arduino code uses.

#### **4.5.1.5.2 Fix Voc point**

If "ADC correction" is "On", this controls whether the last point of the curve should be modified. The Arduino code records the actual ADC values for both channels. Due to noise, however, the value on the channel measuring current is usually not zero. This correction simply zeros out that value.

#### **4.5.1.5.3 Combine =V points**

If "ADC correction" is "On", this controls whether consecutive points that have equal ADC values on the voltage channel will be combined to a single point using the average of the ADC values on the current channel for those points.

#### **4.5.1.5.4 Reduce noise**

If "ADC correction" is "On", this controls whether the noise reduction algorithm is applied.

#### **4.5.1.5.5 Fix overshoot**

If "ADC correction" is "On", this controls whether the curve is corrected to fix the phenomenon where the tail of the IV curve overshoots the voltage that was measured when the circuit was actually open, before charging the capacitor. The voltage of all points is scaled such that the tail of the curve hits the I=0 point at the measured Voc voltage. This phenomenon was a mystery. It is now understood to be due to the +5V supply (from USB) drooping when the relay is active. The reduced reference voltage to the ADC results in voltage measurements that are too high. Negative overshoot (i.e. undershoot) is also corrected; this can be due to SSR/FET current draw in the SSR or FET version.

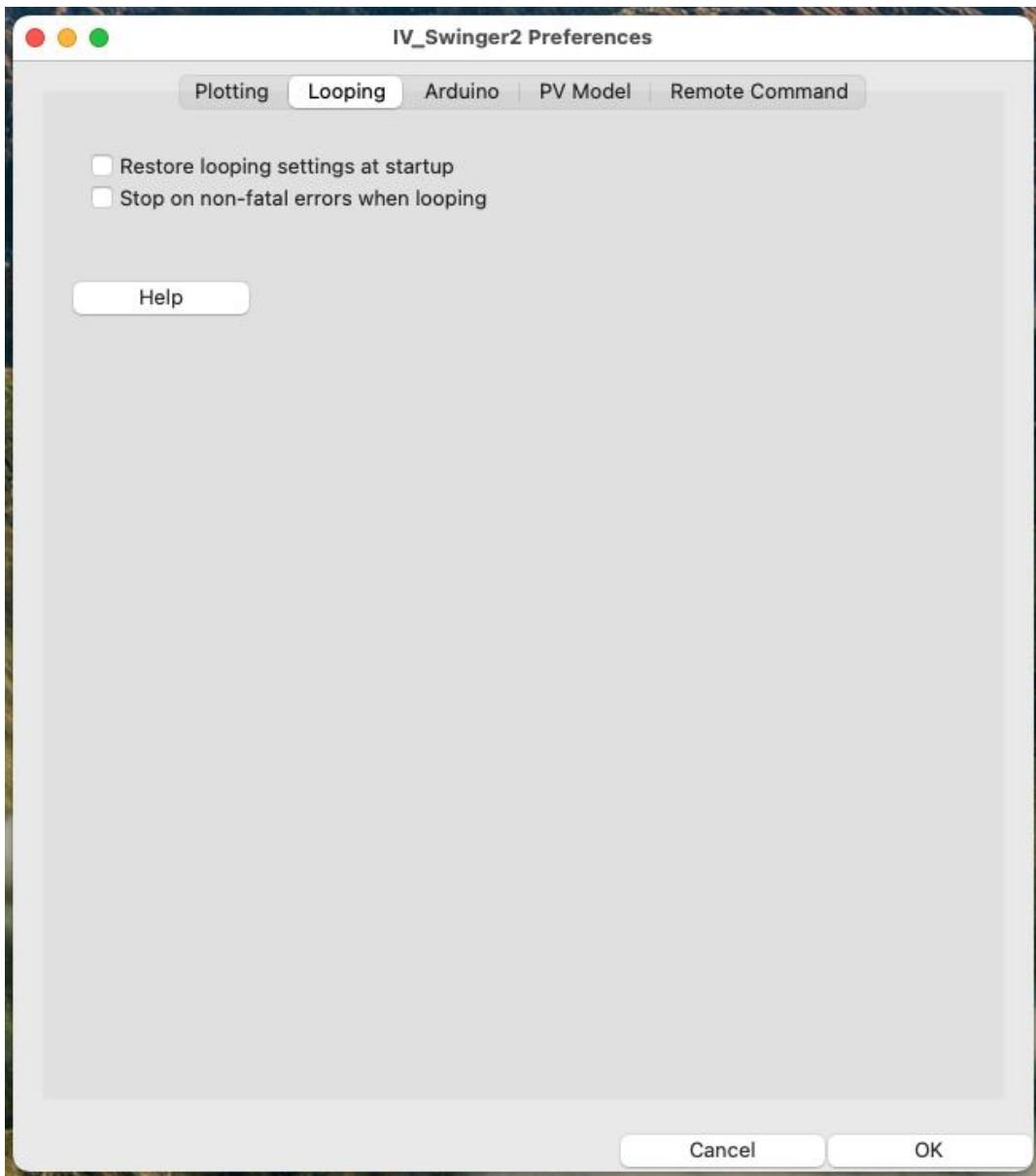
### **4.5.1.6 Battery bias**

The cell version of IV Swinger 2 may require a bias battery to be placed in series with the PV cell in order to trace the curve properly. A calibration of the bias battery must be performed before swinging a cell IV curve using the bias battery. Once that has been done, this control enables the software to "subtract" the bias such that the rendered IV curve is that of the PV cell alone. See Section 9.5.2 on page 75 for more detail.

#### **4.5.1.7 Series resistance compensation ( $m\Omega$ )**

This control may be used to negate the effect of any resistance that is in series with the PV module or cell under test upstream from the point where the voltage is measured. If this value is positive, the voltage at each plotted point is increased by an amount equal to  $I * \text{series\_res\_comp}$ . This could be used, for example, to factor out the effect of a long cable with known resistance. The resulting curve will have a steeper slope (and higher power MPP), than it would without the long cable. A negative value has the opposite effect. Different values may be specified for the normal and battery bias modes.

#### 4.5.2 Looping Preferences Tab

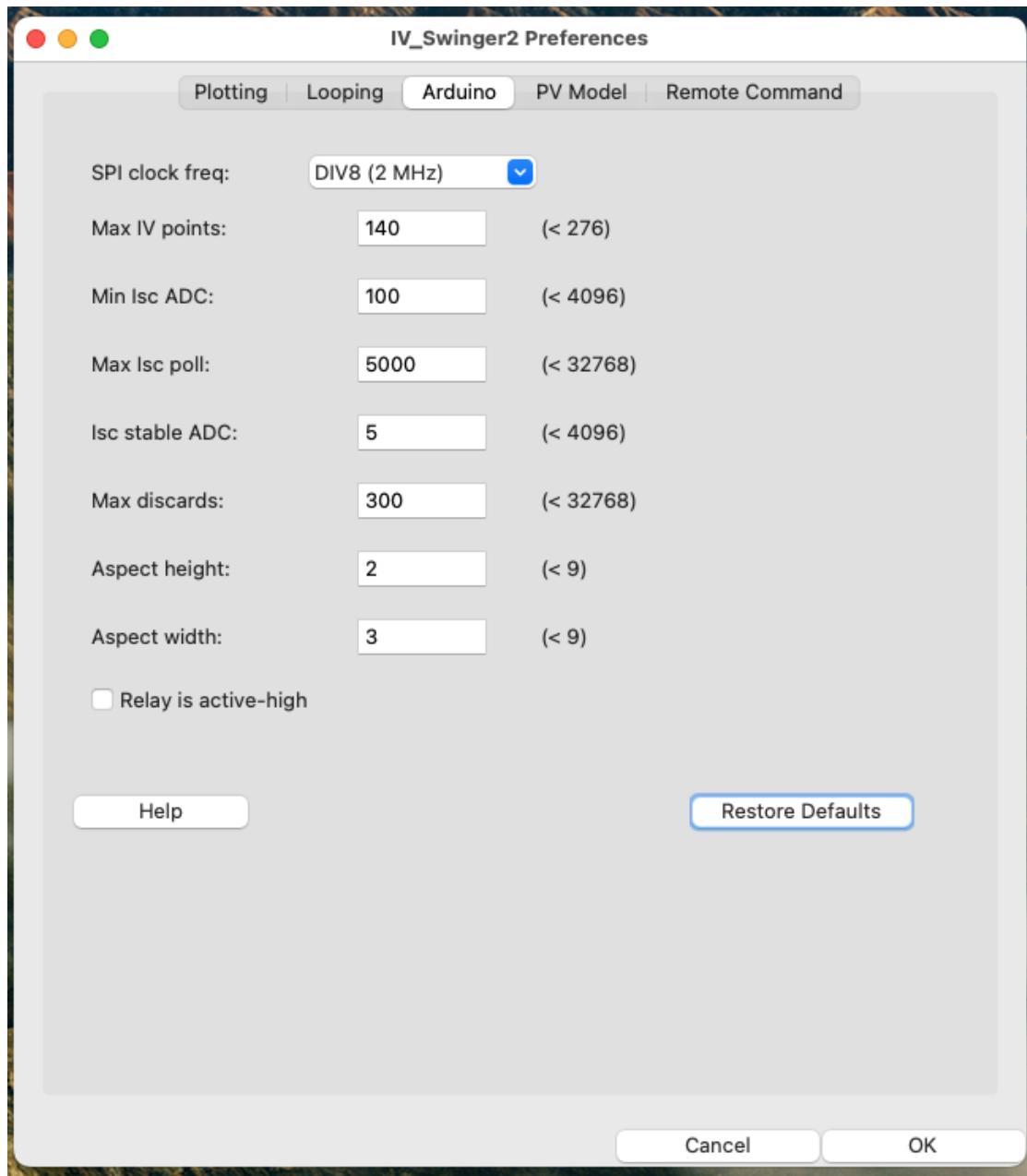


**Figure 4-17: Preferences Looping Tab**

There are currently only two options on the Preferences Looping tab. The main options controlling looping behavior are on the main IV Swinger 2 window to the right of the "Swing!" button (see section 4.1.5 on page 17). The first option is to choose whether the settings on the main screen should be retained after the program is closed and restored the next time it is opened. If this box is not checked, the application always comes up with Loop Mode turned off.

The second option is to choose whether or not looping should stop on non-fatal errors. As of v2.8.0, the default is to not stop on all errors; some errors will display their message on the main screen and the looping will continue. A good example of where this can be useful is the “Timed out polling for stable Isc” error, which is usually caused by insufficient sunlight. That may be intentional, such as during a sweep test for shading or tilt angle, or an unattended multi-day test. If the box is checked and one of these errors is detected, a dialog is displayed and the user must dismiss the dialog and restart the looping by clicking on the Swing! button.

#### 4.5.3 Arduino Preferences Tab



**Figure 4-18: Preferences Arduino Tab**

The configuration options on the Arduino tab are for advanced users who are familiar with the code that runs on the IV Swinger 2 Arduino microcontroller. Normally the default values should be used. Any changes made here are sent to the Arduino when the OK button is pressed and take effect starting with the next run. Following are very brief descriptions of each option.

##### 4.5.3.1 SPI clock freq

Clock frequency of the SPI bus. The SPI bus is used to communicate with the MCP3202 ADC. At 5V, the MCP3202 is specified to work up to 1.8 MHz, but current parts have been shown to operate fine with faster clocks. A higher SPI clock frequency results in more closely spaced points on the IV curve. The default is 2 MHz.

#### **4.5.3.2 Max IV points**

Maximum number of I/V pairs to capture. The actual number will always be less (sometimes substantially less).

#### **4.5.3.3 Min Isc ADC**

Minimum ADC value the channel measuring current must reach before starting to poll for stable Isc.

#### **4.5.3.4 Max Isc poll**

Maximum number of loops waiting for Isc to stabilize before giving up.

#### **4.5.3.5 Isc stable ADC**

Three consecutive measurements must vary less than this amount for Isc to be considered stable.

#### **4.5.3.6 Max discards**

Maximum consecutive points that may be discarded because they are too close together before recording a point anyway.

#### **4.5.3.7 Aspect height**

Height of graph's aspect ratio (max 8). Used for "distance" calculation in the discard algorithm.

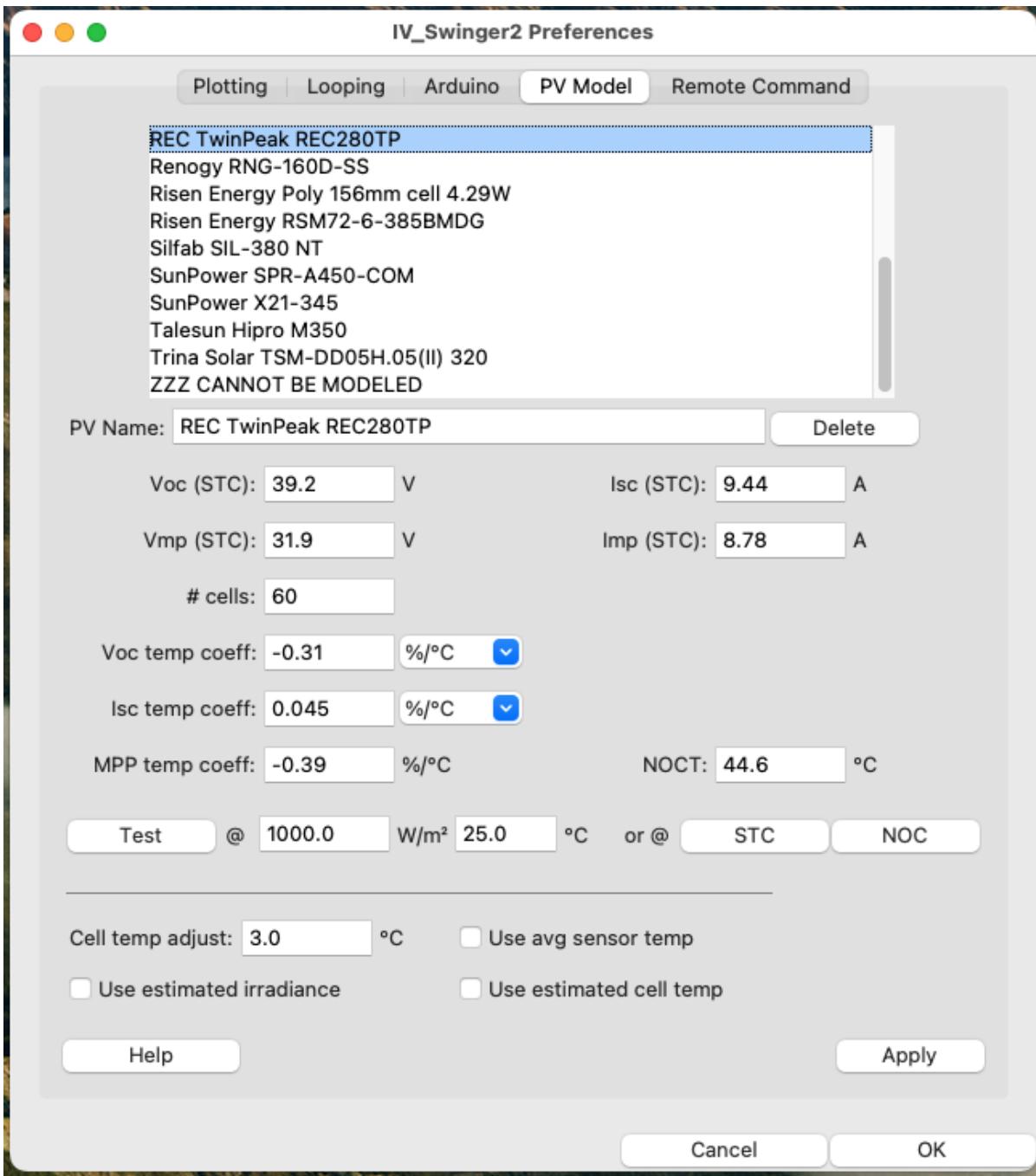
#### **4.5.3.8 Aspect width**

Width of graph's aspect ratio (max 8). Used for "distance" calculation in the discard algorithm.

#### **4.5.3.9 Relay is active-high**

Check ONLY if the IV Swinger 2 was constructed with a (non-standard) relay module that has an active-high trigger pin. This value will be saved in the Arduino EEPROM so the hardware "remembers" what type of relay it has. **This option should NEVER be selected for an IV Swinger 2 that uses SSRs (or FETs), or damage could result.**

#### 4.5.4 PV Model Preferences Tab



**Figure 4-19: Preferences PV Model Tab**

Figure 4-19 above shows the PV Model tab.

The configuration controls on the PV Model tab specify the characteristics of the PV under test so that reference curves may be generated and overlaid onto measured IV curves for comparison. The “Plot Reference” button on the main window (see section 4.1.5 on page 17) is enabled only when a PV has been selected on this tab. Generating the reference curve is performed using a mathematical model of the PV which requires the specifications contained in the datasheet for the particular PV under test. The list at the top is used to select the applicable PV. By default, the entry “NONE” at the top of the list is selected, which disables the “Plot Reference” button. The other entries are pre-populated with several

examples of real PV modules. Of course, it is unlikely that you will find the PV module(s) you are testing in this list, so it will be necessary to manually add new entries.

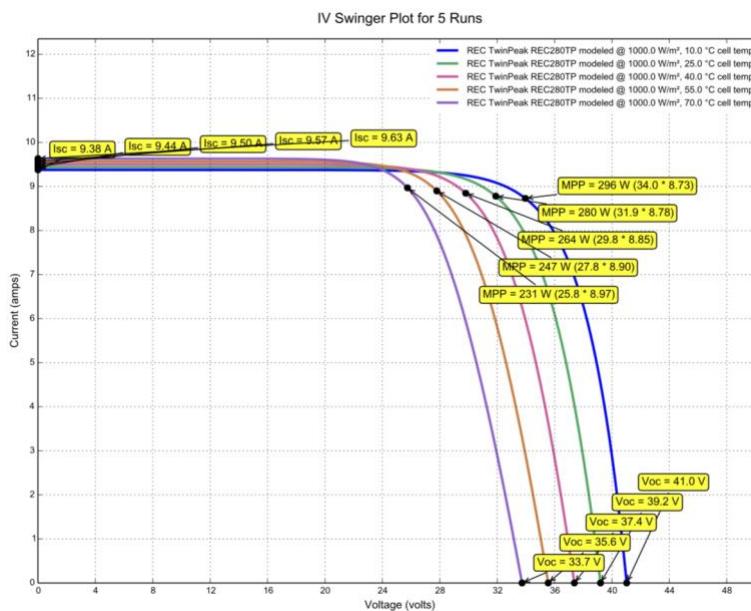
#### 4.5.4.1 Adding a New PV to the List

The process for adding a new PV is:

- Select an existing PV from the list
- Overtype the new PV's name and spec values
- Use the Test button(s) to verify that the PV can be modeled
- Remove unneeded PVs using the Delete button (optional)
- Click on OK to permanently add the new PV to the list

#### 4.5.4.2 Testing a New PV Model

It is not possible to model all PV modules and cells, especially if the datasheet information is entered incorrectly or it is actually incorrect in the datasheet itself. Clicking the Test button runs the model for the specified PV and generates a graph of its expected IV curve at the specified irradiance and cell temperature. An error dialog is generated if the modeling fails, or if it succeeds but is imperfect. The STC button generates the curve at standard test conditions. The NOC button generates the curve at normal operating conditions. The Results Wizard may be used to view the test curves and potentially overlay several of them. This can be useful to visualize the expected effects of varying irradiance with constant cell temperature, or varying cell temperature with constant irradiance. Many datasheets contain graphs like the one in Figure 4-20 below (STC irradiance, various cell temperatures).



**Figure 4-20: Overlaid PV Model Test Curves**

#### **4.5.4.3 Reference Curve Plotting Options**

In order to plot a reference curve, the irradiance and cell temperature are required. If the optional IV Swinger 2 pyranometer and/or temperature sensors are implemented (see Section 3.3 on page 13), those sensor values are used. The controls near the bottom (below the line) determine this behavior. The cell temperature is higher than the temperature measured at the back of the PV, so an adjustment value is specified. If there are multiple temperature sensors, the user may choose whether to use their average or the value of the first sensor only.

If irradiance and/or temperature are not measured, the reference curve is generated using estimated values based on the measured  $V_{oc}$  and  $I_{sc}$  of the PV under test. Of course, this assumes that the  $I_{sc}$  and/or  $V_{oc}$  are "correct" and that all out-of-spec effects are manifested on the curve between those two points only<sup>10</sup>. The "Use estimated irradiance" and "Use estimated cell temp" buttons force estimation when measured values ARE present. When measured values are not present, estimation is always used and those controls are not relevant.

#### **4.5.4.4 Adding a Reference Curve to an Existing IV Curve**

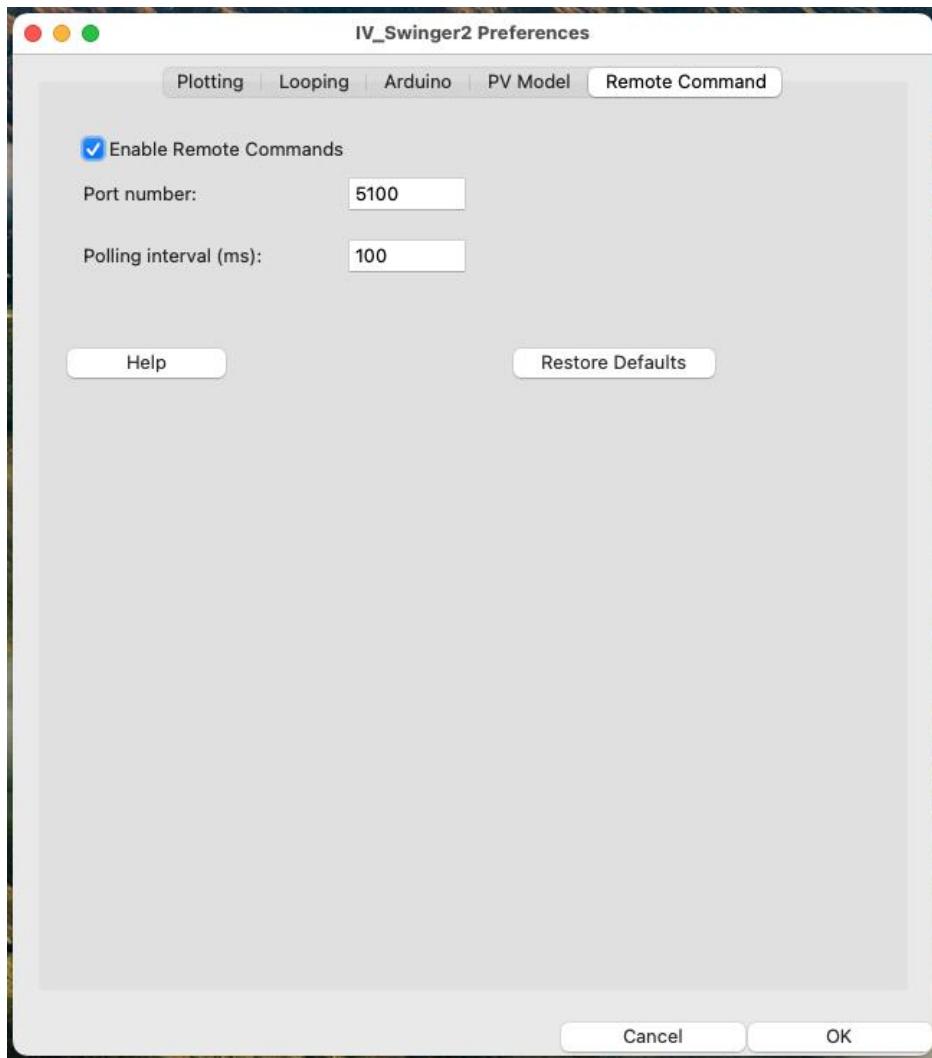
The PV model preferences may be applied to existing IV curves (including those recorded before this feature was available). To do this:

- Open the Results Wizard
- Select the run
- Open Preferences
- Select the PV Model tab
- Choose the appropriate PV and other options
- Click OK button to save
- Check Plot Reference on main window

---

<sup>10</sup> There are many reasons this may not be the case. For example, even a brand new "perfect" PV module will have a low  $I_{sc}$  if it is dirty. Nevertheless, this feature can be useful in some cases.

#### 4.5.5 Remote Command Preferences Tab



**Figure 4-21: Preferences Remote Command Tab**

The Remote Command feature was introduced in v2.8.0. See section 11 on page 78 for a description of the feature and examples.

Note that if the "instances" feature is used, each instance is configured independently and can receive and execute remote commands concurrently with the other instances. They just have to use different ports.

The Remote Command tab in Preferences has three controls:

- Enable Remote Commands:

When checked, the remote command monitor will start when the OK button is clicked. Like other preferences, this is preserved across runs, i.e. the next time the app (or instance) is run, it will come up monitoring for remote commands.

- Port number:

This is the TCP port number that will be used for communicating with the client program. The default value should work in most cases, but the user is free to change it. Each instance must use a different port number.

- Polling interval (ms):

This is how often (in milliseconds) the message queue is checked for incoming commands. The default is 100 ms. A smaller value will decrease latency but will increase CPU load and a larger value will reduce CPU load but will increase latency.

All three controls take effect only when the OK button is clicked.

## 4.6 Folders and Files

Most users should not need or want to access the raw folders files that are used by the IV Swinger 2 software application. For those who do, this section describes them.

The top-level folder is named IV\_Swinger2 and, as described earlier, is in a standard, but somewhat difficult-to-find location on your computer. This location may be different depending on the version of MacOS or Windows that you are running. When you open the Results Wizard, this location is listed at top of the tree view pane. The “Make desktop shortcut” button can be used to make it easier to navigate to.

The IV\_Swinger2 folder contains the following:

- A file named IV\_Swinger2.cfg
- A file named IV\_Swinger2\_starting.cfg
- A folder named “logs”
- A folder for each run named for the date and time of the run
- A folder named “overlays” (after at least one overlay is created)
- A folder named “inst” (after at least one instance is created)

### 4.6.1 IV\_Swinger2.cfg file

This file saves all configuration values such as image size, calibration, and all of the preferences from the Preferences Dialog. These values are restored from this file the next time the application is started. It should not be edited unless you know what you are doing. You may, however, remove it and it will be re-created (with default values) when the application starts.

#### 4.6.1.1 IV\_Swinger2\_starting.cfg file

This file is a copy of the IV\_Swinger2.cfg file when the application most recently started. Differences between this file and the IV\_Swinger2.cfg file represent changes made in the current (or most recent) session.

### 4.6.2 “logs” folder

This folder contains one file for each session that the application is run (possibly spanning many runs). The log file contains debug information logged by the software. If the software detects a software bug, it

may ask the user to send the log file to the developer; otherwise, this is not likely to be useful for general users.

### 4.6.3 Run folders and files

Each run generates its own folder with a name like 170415\_16\_24\_45 (yymmdd hh\_mm\_ss).

▼ 170415_16_24_45	Today, 8:54 AM	--
IV_Swinger2.cfg	Today, 11:01 AM	612 bytes
iv_swinger2_170415_16_24_45.pdf	Today, 8:54 AM	20 KB
iv_swinger2_170415_16_24_45.gif	Today, 8:54 AM	152 KB
iv_swinger2_170415_16_24_45.csv	Yesterday, 11:09 PM	4 KB
adc_pairs_170415_16_24_45.csv	Apr 15, 2017, 4:24 PM	1 KB

**Figure 4-22: Run folder contents**

The “Open Run Folder” entry in the File menu may be used to bring up your computer’s file manager (Finder/Explorer) to locate and access these files (see Section 4.3.3.4 on page 24). The files in the run folder are the following:

#### 4.6.3.1 IV\_Swinger2.cfg (in run folder)

This is a copy of the IV\_Swinger2.cfg file (see section 4.6.1 above) as it existed when the run was performed OR when modifications were made using the Results Wizard.

#### 4.6.3.2 iv\_swinger2\_yymmdd\_hh\_mm\_ss.pdf

This is the PDF of the graph.

#### 4.6.3.3 iv\_swinger2\_yymmdd\_hh\_mm\_ss.gif

This is the GIF of the graph (used for the on-screen display).

#### 4.6.3.4 iv\_swinger2\_yymmdd\_hh\_mm\_ss.csv

This is the CSV file containing the voltage, current, power, and load resistance of each point on the curve. It is in exactly the same format as the original IV Swinger CSV files. It may be read into Excel or another spreadsheet program if desired. Since it is a text file, it can also be easily read by any other program that an advanced user might want to write for the purpose of analyzing the data. This file is regenerated when the Results Wizard Update button is used.

#### 4.6.3.5 adc\_pairs\_yymmdd\_hh\_mm\_ss.csv

This is the CSV file that contains the raw values from the Analog to Digital Converter (ADC). These values are converted to the voltage and current values in the iv\_swinger2\_yymmdd\_hh\_mm\_ss.csv file based on the resistor values in the voltmeter and ammeter circuitry and on the calibration values. They

are also “corrected” to remove some other artifacts. It is unlikely that this file would be useful to the end user. Do not modify this file without creating a copy first because it cannot be re-generated.

#### **4.6.3.6 run\_info\_yymmdd\_hh\_mm\_ss.txt**

This file does not exist by default. However, if one or more of the optional temperature or irradiance sensors are used, it will be created and will contain the measured value(s) for the run. It will be also be created if the user selects the Run Info File entry on the File menu (see Section 4.3.3.3 on page 23).

#### **4.6.4 “overlays” folder**

This folder contains one sub-folder for each overlay that has been created using the Results Wizard. The sub-folders are named with the same yymmdd\_hh\_mm\_ss format as the run folders, but the time refers to the time that the overlay was created, not the time of any of its runs. There are only two files in each sub-folder. Note that these are only the image files; no information about how the overlay was constructed is saved. This is why overlays cannot be “tweaked” after they are finalized.

##### **4.6.4.1 overlaid\_yymmdd\_hh\_mm\_ss.pdf**

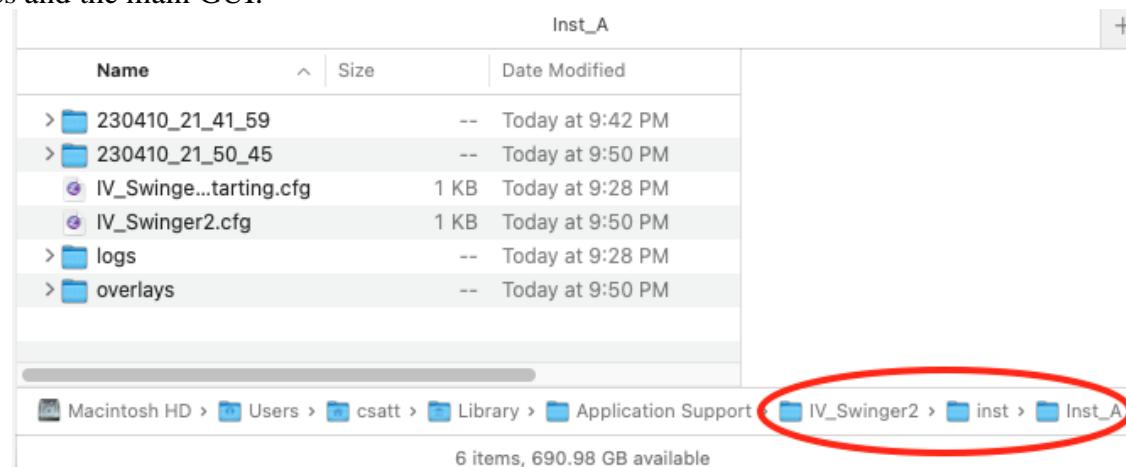
This is the PDF of the overlay graph.

##### **4.6.4.2 overlaid\_yymmdd\_hh\_mm\_ss.gif**

This is the GIF of the overlay graph (used for the on-screen display).

#### **4.6.5 “inst” folder**

This folder comes into existence when the Instances menu is first used to create an “instance” (see section 4.3.6 on page 33.) Each instance has a subfolder under the “inst” folder. Each instance’s subfolder contains all of the same files and folders described above. In other words, each instance has everything that it needs to operate as a separate entity and to keep its results segregated from other instances and the main GUI.



## 5 Bugs

### 5.1 Known Bugs

This section describes bugs that have not yet been understood and/or fixed.

The bugs listed in previous versions of this User Guide have been identified and fixed. There are no known bugs in application version v2.8.0 and Arduino sketch version 1.4.6 (latest versions as of this writing).

Bugs are tracked on GitHub, using “Issues”. For the current list of open and closed bugs, see:

[https://github.com/csatt/IV\\_Swinger/issues](https://github.com/csatt/IV_Swinger/issues)

Note that the GitHub issues are closed when the code with the fix is committed. But those fixes are not available in the pre-built executable Mac and Windows apps until they have been included in a “release”.

It is easy to tell from the GitHub “releases” page ([https://github.com/csatt/IV\\_Swinger/releases](https://github.com/csatt/IV_Swinger/releases)) which commits have been made since the latest release.

### 5.2 Unknown Bugs and Enhancement Requests

Please report any bugs that you find in either of the following two ways:

- Send e-mail to [csatt1@gmail.com](mailto:csatt1@gmail.com)
- File an Issue in GitHub

Enhancement requests are welcome too.

## 6 Ratings and Limitations

1. Number of times the IV Swinger 2 can be dropped and survive: **ZERO**

IV Swinger 2 is a lot more likely to survive a drop than the original IV Swinger. But it's still an acrylic case, and there are connections that are not soldered. So still – don't drop it.

2. Maximum rated PV panel  $I_{sc}$ : **10A**

It is not recommended to connect the IV Swinger 2 to two or more PV panels in parallel. But if you do, remember that the currents of parallel PV panels ADD, so make sure the sum of their rated  $I_{sc}$  values is less than 10A.

3. Maximum rated PV panel  $V_{oc}$ : **80V**

It is not recommended to connect the IV Swinger 2 to two or more PV panels in series. But if you do, remember that the voltages of series PV panels ADD, so make sure the sum of their rated  $V_{oc}$  values is less than 80V.

NOTE: It is probably wishful thinking that the EMR-based designs can withstand 80V. The SSR-based and FET-based designs in theory can handle up to 100V, but that would be right at the specified limit of the SSRs, FETs, load capacitors, and bypass diode.

4. Maximum temperature: **unknown**

This is much less likely to be an issue for IV Swinger 2 than it was for the original IV Swinger since swinging the curve doesn't generate any appreciable heat. Also, the Arduino should be able to run at a higher temperature than the Raspberry Pi.

5. Lifetime (number of IV traces until it dies): **unknown**

The IV Swinger 2 should be much longer lived than the original IV Swinger. The one relay is much less stressed when it switches. On the other hand, it is so easy and quick to swing an IV curve now (especially with loop mode) that the additional longevity may be outweighed by the number of curves attempted. Fortunately, the relay is easy and inexpensive to replace. Nothing else should wear out.

The SSR-based and FET-based designs eliminate that one weak link and should be very robust as long as the current and voltage limits are not exceeded.

6. Number of times you can take the IV Swinger 2 to an airport and not get arrested: **ZERO**

OK, it's much less scary looking than the original. But you still might have some explaining to do.

## **7 Laptop Software Installation**

The IV Swinger 2 application runs on either Mac or Windows

### **7.1 Mac**

#### **7.1.1 IV Swinger 2 application (Mac)**

Requirement: MacOS version 10.14 (Mojave) or higher (Monterey for Apple Silicon)

Download latest release from:

[https://github.com/csatt/IV\\_Swinger/releases/latest](https://github.com/csatt/IV_Swinger/releases/latest)

Under “Assets” click on the link that starts with “iv\_swinger2” and ends with “\_mac.dmg” (Intel) or “\_mac\_arm64.dmg” (Apple Silicon). Open it and follow the instructions on the screen.

### **7.2 Windows**

#### **7.2.1 IV Swinger 2 application (Windows)**

Requirement: Windows 7 or higher

Download latest release from:

[https://github.com/csatt/IV\\_Swinger/releases/latest](https://github.com/csatt/IV_Swinger/releases/latest)

Under “Assets” click on the link that starts with “iv\_swinger2” and ends with “\_win.msi”. Open it and follow the instructions on the screen.

#### **7.2.2 Arduino driver (Windows)**

This is only necessary if you are actually connecting to the IV Swinger 2 hardware. If you are just using the software to look at the results from a different computer, you don’t need this.

Download and install latest release from:

<https://www.arduino.cc/en/Main/Software> (under “Download the Arduino IDE”)

You can choose to install the whole application or only the driver. The next section describes why you might need to use the application, so that is recommended. But only the driver is needed to use IV Swinger 2 hardware that already has its Arduino code installed.

## 8 Arduino Software Installation

The Arduino software that runs on the IV Swinger 2 hardware is called a “sketch”. It is installed when the IV Swinger 2 is constructed, but newer versions with bug fixes and other improvements may become available at a later time.

The IV Swinger 2 application detects the version of the Arduino sketch when it connects. If the Arduino sketch is not up to date, a warning dialog is displayed. This dialog includes instructions for updating the sketch. These instructions are also below.

It is a goal that newer versions of the application work with older versions of the Arduino sketch and vice versa (of course without the benefit of any bug fixes or new features that require both to be updated). However, this may not be possible in all cases.

Here are the instructions for installing the latest IV Swinger 2 Arduino sketch:

- Install the Arduino application (IDE) from:

<https://www.arduino.cc/en/Main/Software>

- Open the Arduino application on your computer
- Find where the Arduino software looks for sketches:

Arduino->Preferences->Settings Tab->Sketchbook location

In the next step, replace <Sketchbook location> with this path.

- Copy the following file to a folder named IV\_Swinger2 in the Sketchbook location<sup>11</sup>:

[https://raw.githubusercontent.com/csatt/IV\\_Swinger/master/Arduino/IV\\_Swinger2/IV\\_Swinger2.ino](https://raw.githubusercontent.com/csatt/IV_Swinger/master/Arduino/IV_Swinger2/IV_Swinger2.ino)

- If you have “wget”, you can do this with:

% wget -P <Sketchbook location>/IV\_Swinger2

[https://raw.githubusercontent.com/csatt/IV\\_Swinger/master/Arduino/IV\\_Swinger2/IV\\_Swinger2.ino](https://raw.githubusercontent.com/csatt/IV_Swinger/master/Arduino/IV_Swinger2/IV_Swinger2.ino)

- Otherwise, use your browser and copy/paste the file contents to a new folder and file: <Sketchbook location>/IV\_Swinger2/IV\_Swinger2.ino

- Go back to the Arduino application and find the IV\_swinger2.ino sketch using:

File->Open

- Click on the arrow button or select “Upload” from the “Sketch” menu

---

<sup>11</sup> This is the latest version. The version that was in the release with the application is [https://raw.githubusercontent.com/csatt/IV\\_Swinger/v2.x.x/Arduino/IV\\_Swinger2/IV\\_Swinger2.ino](https://raw.githubusercontent.com/csatt/IV_Swinger/v2.x.x/Arduino/IV_Swinger2/IV_Swinger2.ino), where v2.x.x is the release name, e.g., v2.8.0.

## 9 IV Curve Tracing of PV Cells [DEPRECATED]

[Smaller PV cells do not require the bias battery that is used for the “Cell Version IV Swinger 2” described in this section. It is only needed for very large PV cells such as the ones used in typical rooftop modules. For smaller cells, the standard “module” design works best, but requires different values of some components to scale it down. This is described in the document [“IV Swinger 2: Hardware Scaling”](#), which also describes how to use the simulator feature of the software to determine the component values. The bias battery scheme works, but not terribly well. Very few “Cell Version” IV Swinger 2’s have ever been built. In theory, the software still supports it, but it is no longer being tested.]

IV Swinger 2 was designed for tracing IV curves of a full PV module (panel). There are now “Cell Versions” of IV Swinger 2 that are capable of tracing IV curves of a single PV cell – which is actually much more challenging. This section of this document describes how to use a Cell Version IVS2. Since it is very similar to how to use the module version, only the differences will be described.

Note: this section was written before PCBs were available for the Cell Version. Like the module version PCBs, there are two variants of the Cell Version: one that uses a 2-channel EMR module and one that uses four SSRs. The photos are of the Perma-Proto-based Cell Version.

### 9.1 Cell Version IV Swinger 2

From the outside, a Cell Version IVS2 looks nearly identical to the module version. The primary difference is that it has a second pair of binding posts. Figure 9-1 below is a photo of a Cell Version IVS2 with the lid removed. You may also notice that the relay module has two relays instead of one.

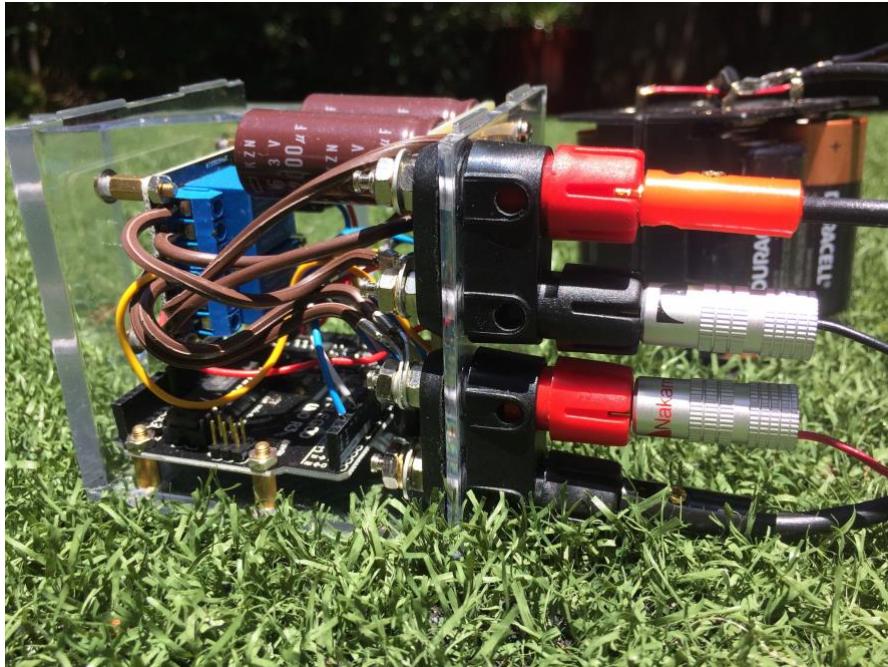


Figure 9-1: Cell Version IV Swinger 2

### **9.1.1 High Power PV Cell Support**

The second pair of binding posts is for the purpose of connecting a bias battery. The bias battery is connected in series with the PV cell in order to boost the voltage to a value that produces a viable IV curve for “high power” PV cells ( $I_{sc} > \sim 4A$ ). The IV curve of the PV cell is rendered by “subtracting” the IV curve of the bias battery. It is necessary to capture a “calibration” IV curve of the battery alone before capturing the combined curve. It is not only inconvenient to have to manually change the connections to swing these two curves, but very slight changes to the battery curve (due to temperature, etc.) have a magnified effect on the final PV-only curve, so the time it takes to manually change the connections often results in poor results. The second relay and second pair of binding posts allow the software to swing the two curves sequentially without human intervention.

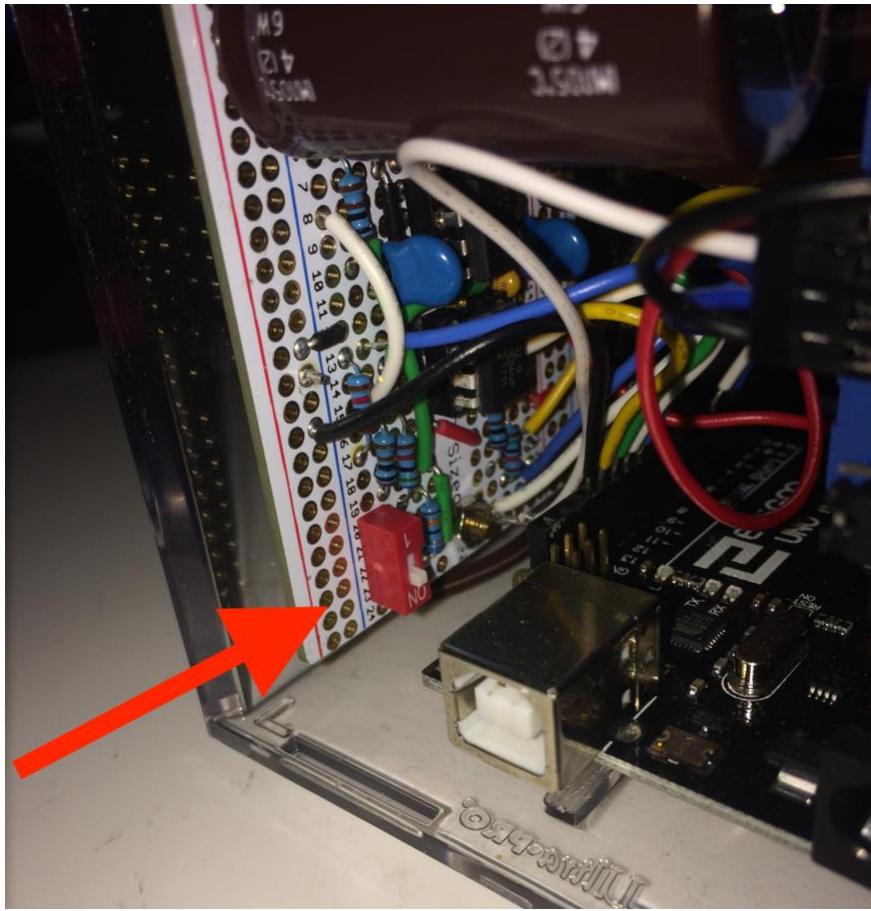
### **9.1.2 Low Power PV Cell Support**

[See note above. Low power PV cells are much better served by a scaled-down “module” version IV Swinger2.]

Low power PV cells (e.g. 250 mW) are physically smaller than the normal high-power cells used in most PV modules. Their  $V_{oc}$  is comparable, but their  $I_{sc}$  is proportionally lower.

A bias battery is not needed for swinging IV curves of low power cells. However, the resolution of the default ammeter circuitry is not adequate to obtain a good IV curve.

The Cell Version IVS2 includes a small DIP switch on the PermaProto board (or PCB) that, when switched “off”, reduces the maximum current that the ammeter can measure by a factor of ~10 (from 13A down to 1.3A). This improves the resolution of the current measurements by a factor of 10. Figure 9-2 below shows the DIP switch in the ON (normal, high-power) position.



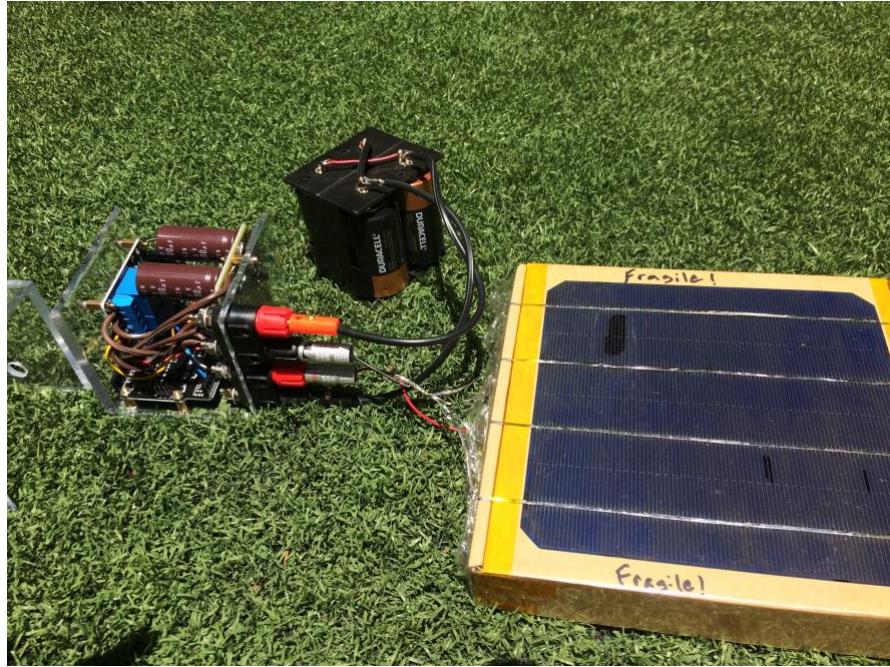
**Figure 9-2: High-Power DIP Switch**

## 9.2 Bias Battery

The Cell Version IVS2 does not have a voltage divider on the input to the voltmeter circuitry, so the maximum supported  $V_{oc}$  voltage is 5 V. Since the bias battery and PV cell are connected in series, this means that the bias battery voltage must be less than about 4.3 V. The bias battery must also have a low internal resistance, i.e. its voltage must not drop too low at the PV cell's  $I_{sc}$  value of current.

The recommended bias battery is a **2x2 arrangement of 1.5V alkaline D-cells**. The series arrangement puts the open circuit voltage at 3V, but doubles the internal resistance. The parallel arrangement halves that. So the 2x2 arrangement has the same internal resistance a single D-cell, but double the voltage. The voltage at 10A is about 1.4V. This satisfies the requirements using inexpensive consumer batteries.

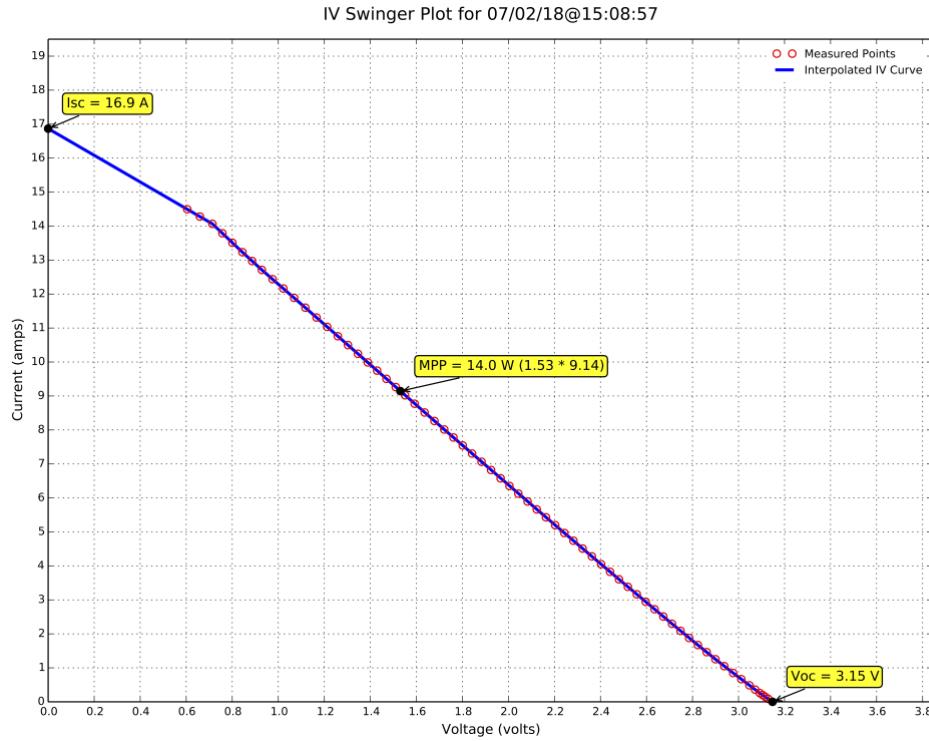
Two Philmore BH121 (or equivalent) 2-cell battery holders can be wired in parallel to securely hold the 4 D-cells. Figure 9-3 below shows the bias battery and a PV cell connected to the Cell Version IVS2. Note the red wire connecting the positive solder lugs and the black wire connecting the two negative solder lugs of the two BH121 holders, which are screwed together back-to-back.



**Figure 9-3: Bias battery and PV cell connected**

Figure 9-4 below is the IV curve for a 2x2 D-cell bias battery. Note that it is a fairly straight line up to a current of nearly 14 A. Smaller batteries (e.g. C, AA, AAA) have higher internal resistance and cannot drive this much current – don't try them. Even D-cells may not work if they are old and tired.

Rechargeable batteries probably won't work (unknown). The requirement is that at the  $I_{sc}$  of the PV cell, the bias battery voltage must be at least 1.0 V. Four new alkaline D-cells in the 2x2 arrangement should accommodate PV cells with an  $I_{sc}$  up to at least 10 A. Note that very little energy is lost swinging the curves, so your D-cell bias battery should last a long time (as long as you take care to never short the leads).

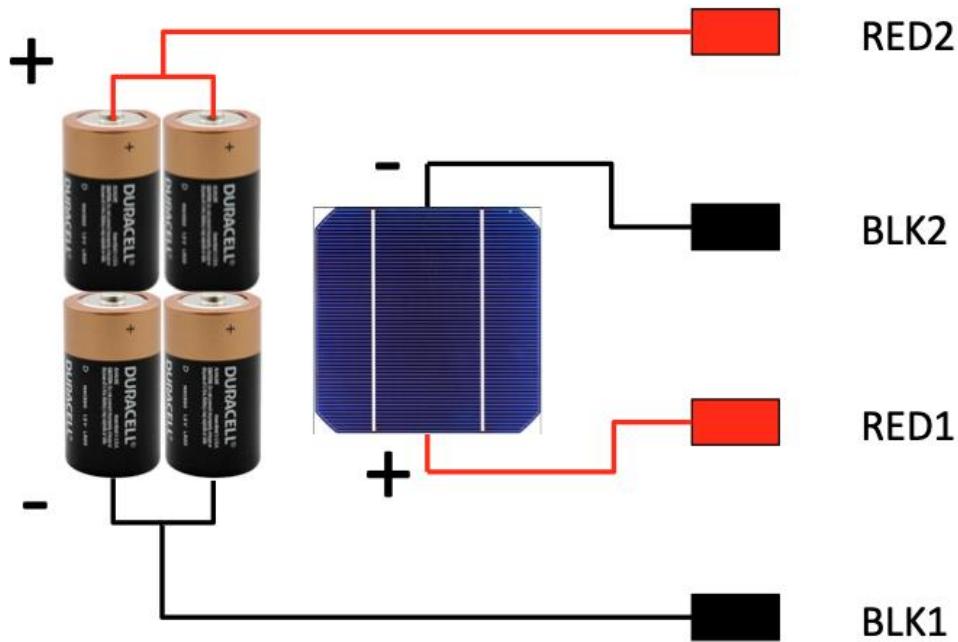


**Figure 9-4: IV Curve for 2x2 D-cell bias battery pack**

### 9.3 Connections

As shown in Figure 9-5 below, the connections to the Cell Version IVS2 are:

- Bias battery:
  - Negative (-) connected to bottom-most black binding post (BLK1)
  - Positive (+) connected to top-most red binding post (RED2)
- PV Cell:
  - Negative (-) connected to middle black binding post (BLK2)
  - Positive (+) connected to middle red binding post (RED1)



**Figure 9-5: Binding Post Connections With Bias Battery**

It is not always necessary to use the bias battery. In particular, if the  $I_{sc}$  of the PV cell is low enough (either its rated value or because the irradiance is low), then the PV cell can be connected to the bottom two binding posts and the upper two binding posts left unconnected.

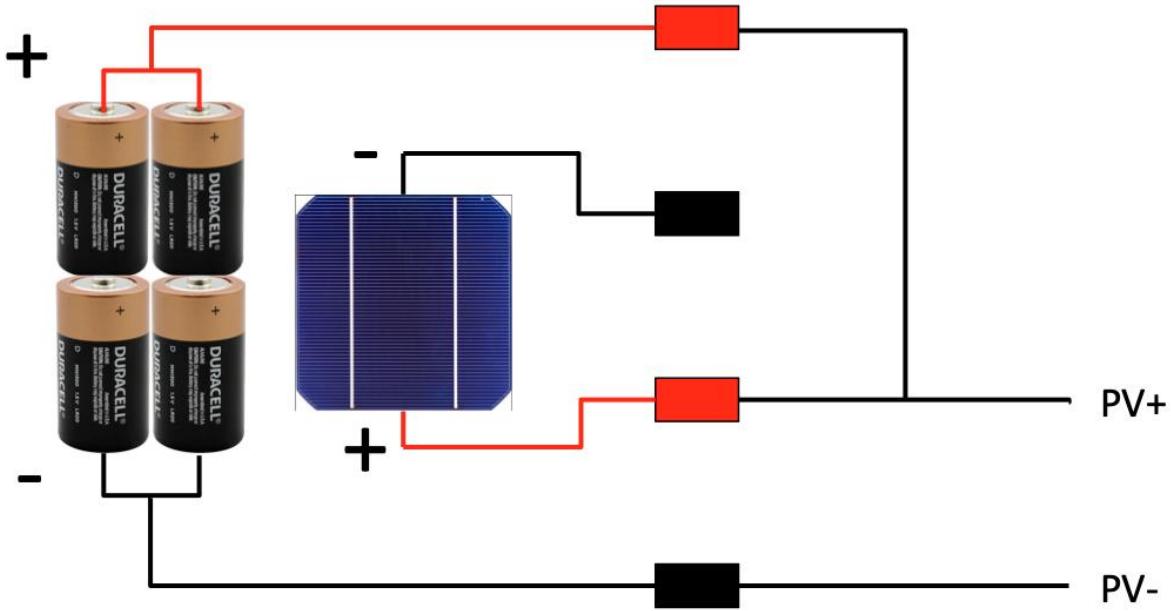
## 9.4 Second Relay

The second relay provides a way for the software to dynamically switch between:

- Bias battery alone (2<sup>nd</sup> relay OFF)
- Bias battery in series with PV cell (2<sup>nd</sup> relay ON)

Note that the bottom two binding posts are connected to the PV+ and PV- inputs to the IV Swinger 2, exactly as they are in the module IV Swinger 2.

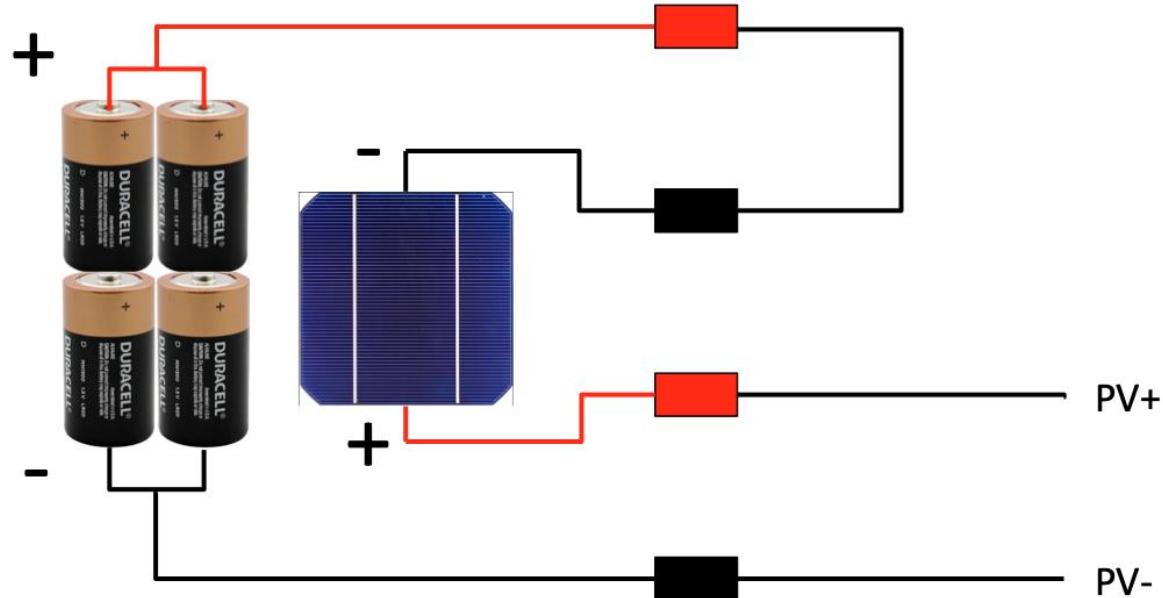
The binding post connections when the 2<sup>nd</sup> relay is inactive (OFF) are shown in Figure 9-6 below.



**Figure 9-6: Connections with 2nd Relay OFF**

Note that the upper black binding post is not connected to anything in this case (floating), so even if the PV cell is connected between the middle two binding posts, it has no effect, and the curve is swung for the bias battery alone, which is connected between the very bottom black binding post and the very top red binding post.

The binding post connections when the 2<sup>nd</sup> relay is active (ON) are shown in Figure 9-7 below.



**Figure 9-7: Connections with 2nd Relay ON**

Therefore:

- When the 2<sup>nd</sup> relay is OFF, the IV curve will be for whatever is connected between the bottom two binding posts OR whatever is connected between the bottom-most black and top-most red binding post.
- When the 2<sup>nd</sup> relay is ON, the IV curve will be for whatever is connected between the bottom-most black binding post and the top-most red binding post IN SERIES with whatever is between the upper black binding post and the lower red binding post.

#### 9.4.1 Side effects of the 2<sup>nd</sup> Relay

There are two side effects of the 2<sup>nd</sup> relay:

- When it is ON, it is drawing current. This causes voltage and current measurements to be higher than they should be due to a droop in the reference voltage to the ADC chip.
- It adds series resistance. This is true for both the OFF state and the ON state, but the amount of resistance may be different for each.

The software attempts to compensate for these side effects, but only if the calibration and usage procedures are followed as described below.

#### 9.4.2 SSR cell version

In the SSR cell version, the SSRs take the place of the two EMRs. The internal details are somewhat different, but the external connections are the same.

### 9.5 Application Configuration and Use

The IV Swinger 2 application has support for the Cell Version of the hardware. Initially, however, it assumes that the hardware is the standard module version.

#### 9.5.1 Calibration

##### 9.5.1.1 Resistor Values

The non-standard values for the R1 and Rf resistors must be communicated to the application using the “Resistors” item on the “Calibrate” menu. When they are changed, these values are stored in the Arduino EEPROM on the IV Swinger 2 hardware so they will stay in effect even if a different laptop is used.

###### 9.5.1.1.1 Resistor R1

There is no voltage divider in the Cell Version IVS2, so the value of R1 must be set to 0.0 ohms. [You can leave the default value for R2 – do not set it to zero.]

### **9.5.1.1.2 Resistor RF**

The value of RF must be the default of 75000 ohms (or its exact measured value) when the DIP switch is in its normal “high power” ON position.

When the DIP switch is moved to the “low power” OFF position, an additional  $680\text{k}\Omega$  resistor, RF1 is placed in series with the  $75\text{k}\Omega$  resistor. The value of RF when the DIP switch is in this position must be changed to 755000 ohms (or the exact measured sum of RF + RF1) in the resistor calibration.

### **9.5.1.2 Bias Battery Calibration**

When a bias battery is used, a calibration is required to capture the IV curve of the bias battery itself. This can be done “manually”, in which case the same bias battery curve is used for all subsequent PV cell curves until another manual calibration is performed. However, this does not produce good results and is not recommended. Instead “dynamic bias calibration” is the only recommended method. When this is enabled, the software captures a bias battery calibration curve every time it swings a biased PV cell IV curve. This is what the second relay is for; it selects one of the following:

- Bias battery by itself
- Bias battery in series with PV cell

#### **9.5.1.2.1 Pre-requisite: normal voltage and current calibration**

It is very important to perform accurate voltage and current calibrations before performing the bias battery calibration (manual or dynamic).

These calibrations must be done with the bias battery and the PV cell in series, connected as follows:

- Battery negative (-) connected to bottom black binding post
- Battery positive (+) connected to PV cell negative (-)
- PV cell positive (+) connected to middle red binding post
- Top two binding posts unconnected

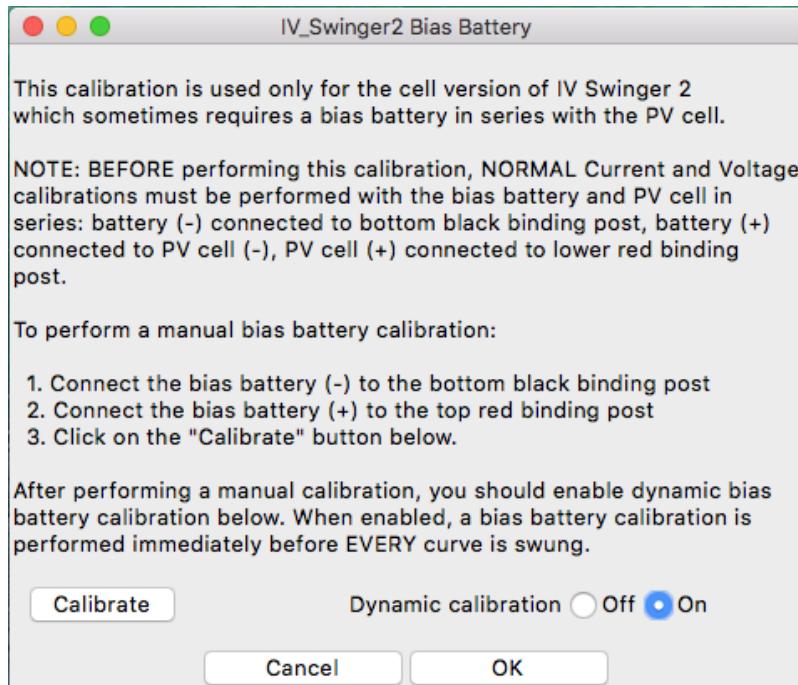
“Battery bias” must be set to “Off” in Preferences (Figure 4-16 on page 49 and Section 4.5.1.6 on page 51). The cell must be kept stationary, and the sun must be constant for the current calibration. Perform the voltage and current calibrations as described in 4.3.5 on page 25 and in the calibration help dialog.

#### **9.5.1.2.2 Manual bias battery calibration**

After the pre-requisite normal current and voltage calibrations have been performed, a manual bias battery calibration should be performed at least once:

- Connect the bias battery and PV cell to the binding posts as specified in 9.3 on page 70
- Select the “Bias Battery” item from the Calibrate menu
- Press the “Calibrate” button

There will be some visual checks of the bias battery IV curve that it will ask you to perform. As long as those look good, you may now enable dynamic bias battery calibration. Figure 9-8 below shows the Bias Battery calibration dialog with dynamic calibration enabled.



**Figure 9-8: Bias battery calibration dialog**

## 9.5.2 Swinging an IV Curve with a Bias Battery

After one manual bias battery calibration has been performed and dynamic calibration is enabled, you are ready to swing biased PV cell IV curves.

### 9.5.2.1 Applying the Battery Bias

With the default preferences, the 2<sup>nd</sup> relay will remain OFF when the Swing! button is pressed, and the resulting curve will be of whatever is connected between the bottom two binding posts OR between the bottom black and top red binding posts. If the bias battery and PV cell are connected as shown in Figure 9-3, you will get the IV curve of the bias battery itself, which is probably not what you want.

You need to turn “Battery bias” to “On” on the Plotting tab of Preferences (see Figure 4-16 on page 49).

If dynamic bias battery calibration is enabled (recommended), when you press Swing!, the software will first swing a calibration curve for the battery itself with the 2<sup>nd</sup> relay OFF. It will then turn the second relay ON and swing the combined curve of the bias battery and PV cell in series. However, what is displayed has the battery curve subtracted from the combined curve, leaving the curve of the PV cell itself.

If dynamic bias battery calibration is not enabled (not recommended), when you press Swing!, the software will turn the second relay ON and swing the combined curve of the bias battery and PV cell in series. What is displayed has the battery curve subtracted from the combined curve, leaving the curve of the PV cell itself. But in this case, the battery curve it subtracts is “old”, i.e. the one from the most recent manual (or dynamic) calibration.

Note that when you use the Results Wizard to look at these IV curves, you can bring up Preferences and “play” with the “Battery bias” button to toggle back and forth between seeing the combined battery+PV curve that was actually measured and the PV-only curve that was derived. You can also see the bias battery calibration curve itself by selecting the “Battery” folder (see Section 4.4.9 on page 44).

### **9.5.2.2 Calibrating Based on a Biased IV Curve**

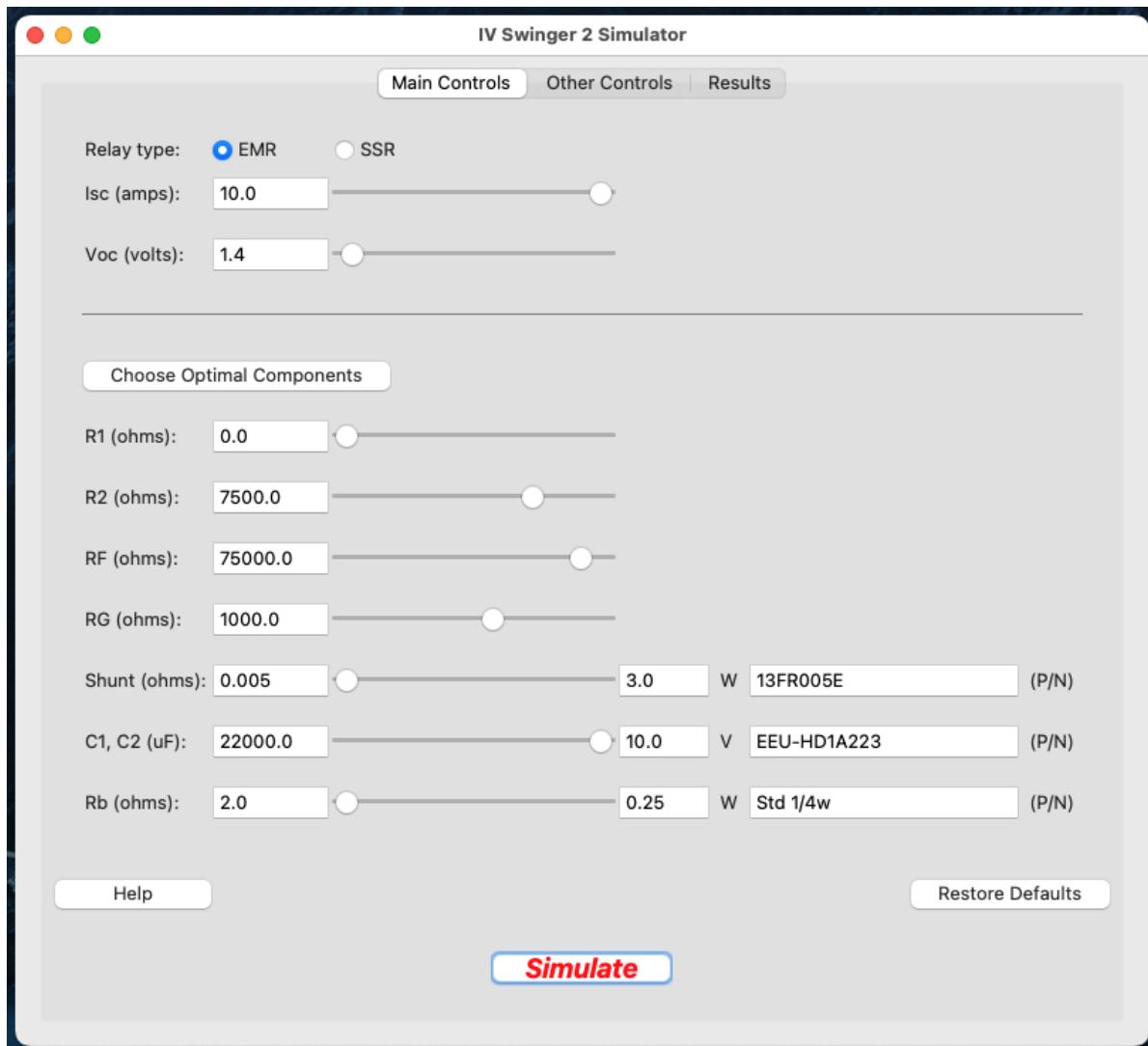
You may perform a voltage calibration after swinging an IV curve with “Battery bias” set to On, using the normal procedure. You may not, however, perform a current calibration in this case. The voltage calibration is only calibrating the “subtraction”, and that is only relevant to voltage.

## **9.6 Consider Using Multiple High Power PV Cells in Series**

It is possible to get good IV curves for single high power PV cells using a bias battery (and a lot of work has gone into making this possible). However, if it is feasible to test 3 or more “identical” PV cells in series, then a bias battery is not needed. This simpler configuration is always going to produce more reliable results than the bias battery configuration, and is recommended. Just remember that the total voltage must be less than 5V.

## 10 Simulation

The IV Swinger 2 application includes a hardware simulator that is useful to choose components for “scaled” versions that are capable of measuring IV curves for PV modules or cells that are smaller or larger than the standard design supports. The simulator is invoked from the Run Simulator entry on the Help menu. This is an intentionally inconspicuous place since the vast majority of users will have no reason to use it. There is another document entitled “[IV Swinger2: Hardware Scaling](#)” that describes how to use the simulator.



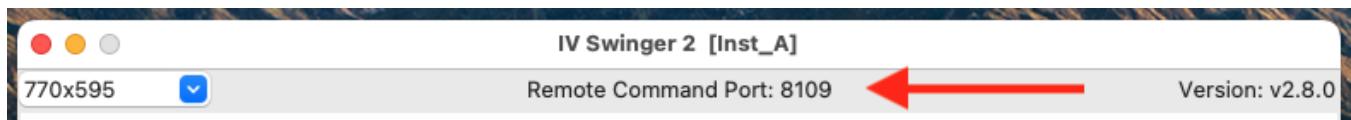
# 11 Remote Command Feature

The remote command feature was introduced in v2.8.0. It allows external client programs (running on the same computer or on a different computer) to send commands to the IV Swinger 2 application. A received command is executed and a reply is sent back to the requester.

The feature uses the open-source ZeroMQ messaging library, which the client program must also use. Many (currently 28) programming languages are supported by ZeroMQ. The simple request/reply (REQ/REP) client/server pattern is used.

The Remote Command tab of Preferences is used to configure and enable the feature. See section 4.5.5 on page 58. If the instances feature (page 33) is used, each instance may be configured to independently accept remote commands on its own port concurrently with the others.

If the remote command monitor is active, the port that it is listening on is displayed on the top line of the GUI window:



## 11.1 Command Syntax

Commands are strings, with the following syntax:

```
<command name>[;<arg>[;<arg>]...]
```

In other words, a command consists of a command name followed by a zero or more arguments separated by semicolons. If there are no arguments, the first semicolon is not necessary.

## 11.2 Reply Syntax

Replies are strings, with the following syntax:

```
SUCCESS[;<info>[;<info>]...]  
or  
ERROR[;<info>[;<info>]...]
```

In other words, a reply consists of either SUCCESS or ERROR followed by a zero or more information fields separated by semicolons.

## 11.3 Supported Commands

Currently, there is only one supported command. More may be added in the future.

### 11.3.1 Swing Command

The Swing command (currently) has no arguments, so it may be sent simply as the string “Swing”. However, it may include one or more “comments”, which are strings starting with the “#” character after a semicolon. These will be ignored by the remote command monitor but will show up in the log file and could be useful for debugging.

The reply to a successful Swing command is:

SUCCESS; Isc = <val> A, Voc = <val> V, MPP = <val> W (<val> V \* <val> A);<run dir path>  
e.g.

SUCCESS; Isc = 5.131696 A, Voc = 20.061550 V, MPP = 67.431723 W (18.797499 V \* 3.587271 A);/Users/csatt/Library/Application Support/IV\_Swinger2/inst/Inst\_A/230331\_16\_25\_46

The reply to a failed Swing command is:

ERROR; rc=<return code name>  
e.g.

ERROR; rc=RC\_ZERO\_VOC

When a Swing command is received, it is handled as if the user clicked on the Swing! Button, with some minor differences:

- Loop Mode is ignored
- Errors (other than exceptions) are displayed on the main screen so no user action is required to dismiss an error dialog

The IV curve is displayed if the command is successful. All result files are saved as usual.

## 11.4 Client Program Example

This section has a simple “starting point” example of a client program – one version in Python and one version in C. For more information on ZeroMQ including how to install it and examples in other languages, visit <https://zeromq.org/>.

This simple example has hardcoded values for the IP address and TCP port to connect to.

### IP address

The IP address value “localhost” is sufficient if the client program is running on the same computer as the IV Swinger 2 application. If the client program is running on a different computer on the same network, the local IP address of the computer running the application must be explicitly specified. Typically, this will be a DHCP address (192.168.x.x). It is even possible to send remote commands through the Internet, but this requires the ability to configure the NAT firewall on the network with the IV Swinger 2 laptop to use *port forwarding* to allow the packets through. In that case, the address is the public IP address of the firewall.

### Port

The port is the TCP port that the remote command monitor is listening on. It is specified on the Remote Command tab of Preferences and is displayed on the top line of the GUI window when the monitor is active. If port forwarding is configured on the NAT firewall, this is the port that needs to be forwarded.

## 11.4.1 Python

This script can be found here:

[https://raw.githubusercontent.com/csatt/IV\\_Swinger/master/python3/sample\\_ivs2\\_rcmd\\_client.py](https://raw.githubusercontent.com/csatt/IV_Swinger/master/python3/sample_ivs2_rcmd_client.py)

```
#!/usr/bin/env python
#
# This is a very simple example of a client that sends a "Swing" remote
# command to the IV Swinger 2 application.
#
"""Simple IV Swinger 2 remote command client example"""

import zmq # pip install pyzmq

IP_ADDRESS = "localhost"      # This computer
# IP_ADDRESS = "192.168.1.102"    # DHCP IP address
# IP_ADDRESS = "99.95.164.127"    # Public IP address (needs port forwarding)

PORT = 5100

COMMAND = "Swing"

# Create ZMQ context
context = zmq.Context()

# Create socket
socket = context.socket(zmq.REQ)

# Connect socket to port
print(f"Connecting to IP address {IP_ADDRESS} port {PORT}")
socket.connect(f"tcp://:{IP_ADDRESS}:{PORT}")

# Send command
print(f"Sending command: {COMMAND}")
socket.send_string(COMMAND)

# Get the reply.
reply = socket.recv_string()
print(f"Received reply: {reply}")
```

## 11.4.2 C

This program can be found here:

[https://raw.githubusercontent.com/csatt/IV\\_Swinger/master/c/sample\\_ivs2\\_rcmd\\_client.c](https://raw.githubusercontent.com/csatt/IV_Swinger/master/c/sample_ivs2_rcmd_client.c)

```
#include <zmq.h>
#include <string.h>

#define IP_ADDRESS "localhost"      // This computer
//#define IP_ADDRESS "192.168.1.102" // DHCP IP address
//#define IP_ADDRESS "99.95.164.127" // Public IP address (needs port forwarding)

#define PORT 5100

#define COMMAND "Swing"

#define MAX_REPLY_LENGTH 512

int main(void) {

    char socket_str[32];
    char reply[MAX_REPLY_LENGTH];

    // Create ZMQ context
    void *context = zmq_ctx_new();

    // Create socket
    void *socket = zmq_socket(context, ZMQ_REQ);

    // Connect socket to port
    printf("Connecting to IP address %s port %d\n", IP_ADDRESS, PORT);
    sprintf(socket_str, "tcp://%s:%d", IP_ADDRESS, PORT);
    zmq_connect(socket, socket_str);

    // Send command
    printf("Sending command: %s\n", COMMAND);
    zmq_send(socket, COMMAND, strlen(COMMAND), 0);

    // Get the reply
    zmq_recv(socket, reply, MAX_REPLY_LENGTH, 0);
    printf("Received reply: %s\n", reply);

    // Clean up
    zmq_close(socket);
    zmq_ctx_destroy(context);

    return 0;
}
```