

Programming Example: Python Power Supply Efficiency Example

June 21, 2021

Power supply efficiency is an important aspect of any power supply design. Using a few standard pieces of test gear, one can quickly build a modular system with enough resolution to measure the most demanding applications.

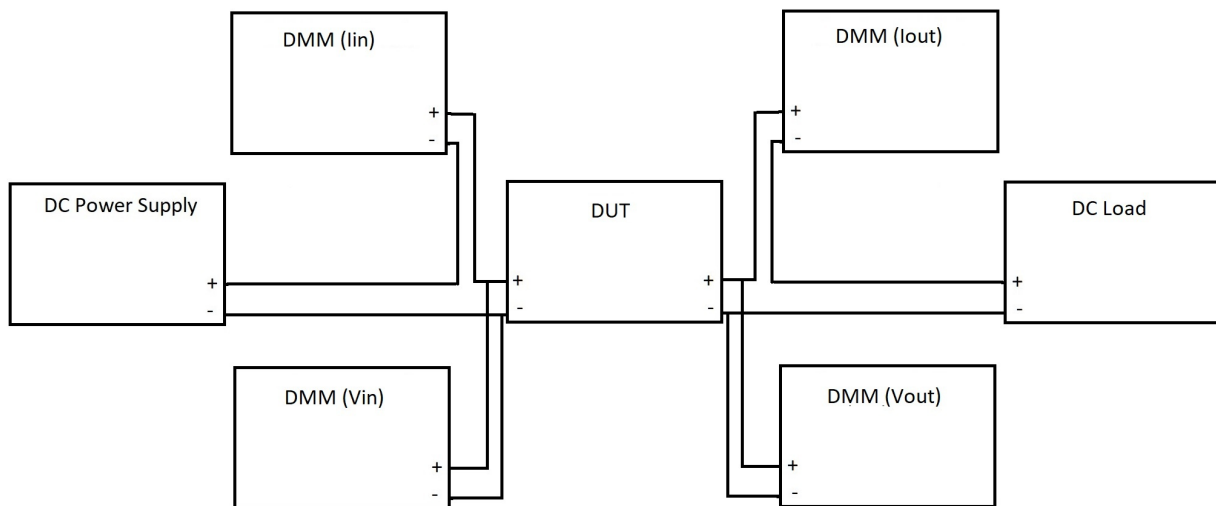
This note describes a Python example that we created as part of an application note which describes the basics of power supply efficiency testing, which is available here: [Power Supply Efficiency App Note](#)

Typical power efficiency curves require 10 or more load current steps to get an accurate representation of efficiency. Manually performing this test can be a bit tedious and prone to error.

For the resolution question, consider that most commercial power supplies and DC loads have 3.5 digit measurement resolution and accuracy values that may not be high enough to characterize the DUT confidently.

To address these two considerations, we added four multimeters: one to measure V_{in} , I_{in} , V_{out} , and I_{out} and created a computer program to automate the instrument configurations, step load current values, and data collection.

Here is a wiring diagram of the setup:



Software

Automating this type of test is a great way to minimize errors as well as speed up the test process.

We put together an example program using Python that performs the following:

- Prompts the user for start, stop, and number of current load steps
- Power Supply settings to deliver proper power to the DUT
- Filename to save the CSV data file

Requirements:

- Qty 1 SIGLENT SPD series power supply
- Qty 1 SIGLENT SDL1000X series DC electronic load
- Qty 4 SIGLENT SDM3000 series digital multimeter
- Qty 1 6 port or more USB hub, USB cables
- Cabling* to connect all instruments and DUT
- Python 3.0 or greater
- tkinter
- PyVISA

*Use the proper gauge wiring and connectors that are rated above the maximum current draw expected to minimize the risk of fire or overheating.

To run the program:

- Download Python 3.6, PyVISA, and tkinter (a module that eases Graphical User Interface creation)
- Download the program [SIGLENT PS Efficiency.ZIP](#)
- Connect instruments and DUT. It may be helpful to put a small label on each DMM to make it easier to visually identify their roles.
- Open the program using a Python editor like IDLE
- Edit the USB addresses for each instrument.

Here is a screenshot of the code showing the addresses:

```

PS Efficiency GUI.py - C:\Users\jayre\Documents\Application Notes\Power Supply Efficiency\GUI Version\GUI Version\PS Eff...
File Edit Format Run Options Window Help
UserOutput3_Label.pack()
UserOutput4_Label.pack()

return StepOutput

def StartButtonAction(inboxes, outlabels):

    ConfigDel = 1 #Configuration delay in seconds
    SetupDel = 0.2 #Setup delay in seconds
    MeasDel = 0.01 #Configure measure delay in seconds

    rm = visa.ResourceManager()
    SPDadd = 'USB0::0xF4EC::0x1410::SPD1XDAD1R0011::INSTR'
    SDLadd = 'USB0::0xF4EC::0x1621::SDL13FBQ3R0152::INSTR'

    SDMVinadd = 'USB0::0xF4EC::0xEE38::SDM35FAX3R0637::INSTR'
    SDMIinadd = 'USB0::0xF4EC::0xEE38::SDM36FAX1R0084::INSTR'

    SDMVoutadd = 'USB0::0xF4EC::0xEE38::SDM35FAC3R0135::INSTR'
    SDMIoutadd = 'USB0::0xF4EC::0xEE38::SDM36FAX1R0083::INSTR'

    SPD = rm.open_resource(SPDadd)
    SDL = rm.open_resource(SDLadd)

    SDMVin = rm.open_resource(SDMVinadd)
    SDMIin = rm.open_resource(SDMIinadd)

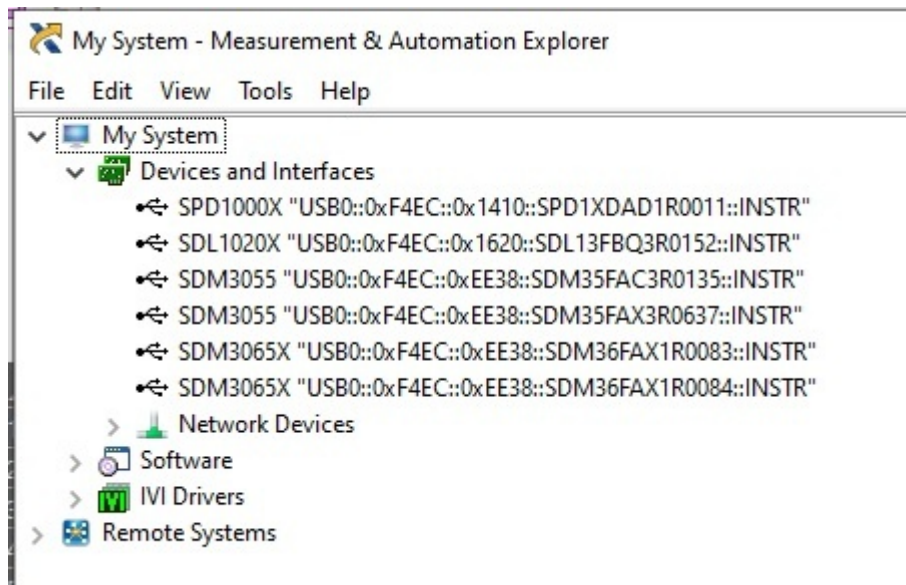
    SDMVout = rm.open_resource(SDMVoutadd)
    SDMIout = rm.open_resource(SDMIoutadd)

```

Where:

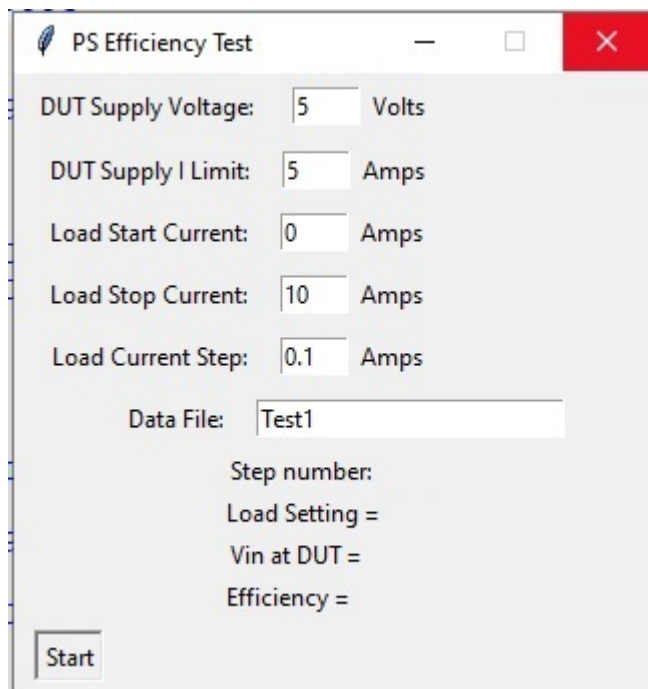
- SPDadd is the power supply address
- SDLadd is the electronic load address
- SDMVinadd is the address of the DMM measuring the input voltage
- SDMIinadd is the address of the DMM measuring the input current
- SDMVoutadd is the address of the DMM measuring the output voltage
- SDMIoutadd is the address of the DMM measuring the output current

One easy way to establish the USB addresses is to use the National Instruments MAX application. With Max, you can connect the instruments via USB and check all of the addresses:



NOTE: The USB resource ID contains the instrument serial number, which can be used to help identify which instrument you are communicating with. As noted above, it may be helpful to put a small label on each instrument including the serial number to make it easier to visually identify each instrument and ensure they are wired correctly for their role in the test.

- After wiring everything up, you can run the module.
- The first thing you should see is the User Interface (UI) as shown below:



- Enter the proper values for each entry and press Start

- As the process begins, you should see the output indicator on the power supply turn on, the DC Load input light should turn on, and the displayed values for current and voltage should meet your programmed values
- The program prints each step value measurement on the display as shown here:

```

*Python 3.6.5 Shell*
File Edit Shell Debug Options Window Help
Load: Siglent Technologies,SDL1020X,SDL13FBQ3R0152,1.1.1.19R1
DMM V In: Siglent Technologies,SDM3055,SDM35FAX3R0637,1.01.01.20R2
DMM I In: Siglent Technologies,SDM3065X,SDM36FAX1R0084,3.01.01.07
DMM V Out: Siglent Technologies,SDM3055,SDM35FAC3R0135,1.01.01.20R2
DMM I Out: Siglent Technologies,SDM3065X,SDM36FAX1R0083,3.01.01.07
Total Steps: 101
Start: 0.0
Stop: 10.0
Increment: 0.1
Start-->:SOURce:CURRent:LEVEL 0.0
Load: 0.000000
Vin Setting: 5.0
Vin at DUT: +4.84649260E+00
DUT Vin Error: 0.153507399999999963
Adjusted Vin: 5.1535074
SDMVIn: +5.00046023E+00
SDMIIn: +1.35293625E-01
Pin: 0.6765303911850338
SDMVoutMeas: +1.49570581E+00
SDMIoutMeas: +7.38562773E-06
Pout: 1.1046726306258111e-05
Pd: 0.6765193444587275
Eff: 0.00%

Step 2 -->:SOURce:CURRent:LEVEL 0.1
Load: 0.100000
Vin Setting: 5.1535074
Vin at DUT: +4.96860100E+00
DUT Vin Error: 0.031399000000000004
Adjusted Vin: 5.1849064
SDMVIn: +5.00074402E+00
SDMIIn: +1.63302482E-01
Pin: 0.8166339103126576
SDMVoutMeas: +1.49536316E+00
SDMIoutMeas: +9.33043810E-02
Pout: 0.13952393401400395
Pd: 0.6771099762986537
Eff: 17.09%

Step 3 -->:SOURce:CURRent:LEVEL 0.2
Load: 0.200000
Vin Setting: 5.1849064
Vin at DUT: +4.96677129E+00
DUT Vin Error: 0.033228710000000038
Adjusted Vin: 5.21813511
SDMVIn: +5.00083737E+00
SDMIIn: +1.93319630E-01
Pin: 0.9667600300585731
SDMVoutMeas: +1.49495282E+00

```

This program uses the auto-ranging measurement function of each instrument, which can add time

to each measurement. For proper DC measurements, there needs to be some time delay at each step value to ensure a stable reading. We have programmed sufficient delays for most DUTs, but you may need to tweak delays to meet your specific test needs.

- Upon completion, the program will indicate that it is complete and write a CSV file to the folder where the Python program is located.


```
*Python 3.6.5 Shell*
File Edit Shell Debug Options Window Help

Step 99 -->:SOURce:CURRent:LEVEL 9.8
Load: 9.800000
Vin Setting: 5.301566419999999
Vin at DUT: +4.99945203E+00
DUT Vin Error: 0.0005479700000003973
Adjusted Vin: 5.302114389999999
SDMVIn: +5.00010549E+00
SDMIIn: +3.10342924E+00
Pin: 15.51747358075053
SDMVoutMeas: +1.46351300E+00
SDMIoutMeas: +9.10822651E+00
Pout: 13.33000790432963
Pd: 2.1874656764208993
Eff: 85.90%

Step 100 -->:SOURce:CURRent:LEVEL 9.9
Load: 9.900000
Vin Setting: 5.302114389999999
Vin at DUT: +5.00011296E+00
DUT Vin Error: -0.0001129600000000508
Adjusted Vin: 5.302001429999999
SDMVIn: +5.00005695E+00
SDMIIn: +3.10398565E+00
Pin: 15.520105021982767
SDMVoutMeas: +1.46344890E+00
SDMIoutMeas: +9.10971299E+00
Pout: 13.331599454531212
Pd: 2.1885055674515552
Eff: 85.90%

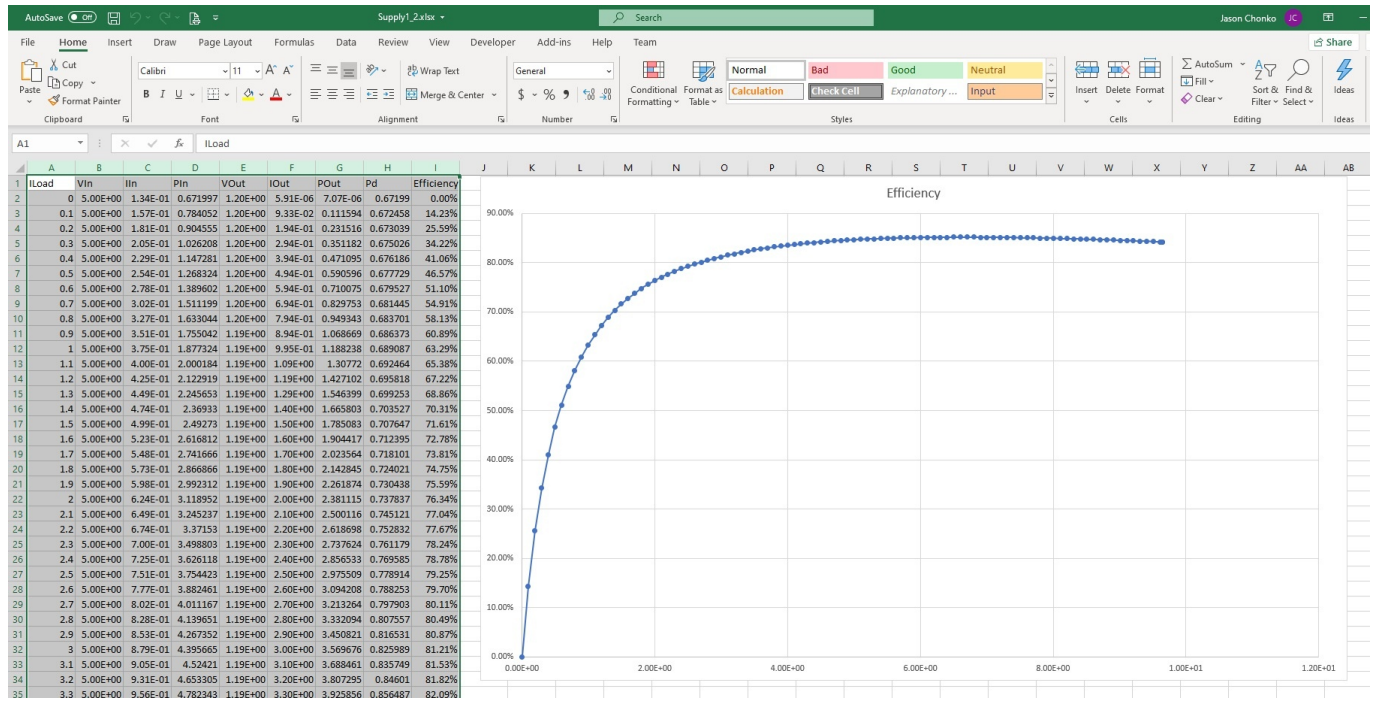
Step 101 -->:SOURce:CURRent:LEVEL 10.0
Load: 10.000000
Vin Setting: 5.302001429999999
Vin at DUT: +5.00005321E+00
DUT Vin Error: -5.3209999999914714e-05
Adjusted Vin: 5.301948219999999
SDMVIn: +5.00001587E+00
SDMIIn: +3.10438393E+00
Pin: 15.52196891657297
SDMVoutMeas: +1.46350731E+00
SDMIoutMeas: +9.11028471E+00
Pout: 13.33296826926623
Pd: 2.18900064730674
Eff: 85.90%

Sweep Complete. Data saved in Test1.csv
```

- Find the CSV file, copy columns A – I:

A	B	C	D	E	F	G	H	I
Iload	VIn	IIn	PIn	VOut	IOut	POut	Pd	Efficiency
0	5.00E+00	1.34E-01	0.671997	1.20E+00	5.91E-06	7.07E-06	0.67199	0.00%
0.1	5.00E+00	1.57E-01	0.784052	1.20E+00	9.33E-02	0.111594	0.672458	14.23%
0.2	5.00E+00	1.81E-01	0.904555	1.20E+00	1.94E-01	0.231516	0.673039	25.59%
0.3	5.00E+00	2.05E-01	1.026208	1.20E+00	2.94E-01	0.351182	0.675026	34.22%
0.4	5.00E+00	2.29E-01	1.147281	1.20E+00	3.94E-01	0.471095	0.676186	41.06%
0.5	5.00E+00	2.54E-01	1.268324	1.20E+00	4.94E-01	0.590596	0.677729	46.57%
0.6	5.00E+00	2.78E-01	1.389602	1.20E+00	5.94E-01	0.710075	0.679527	51.10%
0.7	5.00E+00	3.02E-01	1.511199	1.20E+00	6.94E-01	0.829753	0.681445	54.91%
0.8	5.00E+00	3.27E-01	1.633044	1.20E+00	7.94E-01	0.949343	0.683701	58.13%
0.9	5.00E+00	3.51E-01	1.755042	1.19E+00	8.94E-01	1.068669	0.686373	60.89%
1	5.00E+00	3.75E-01	1.877324	1.19E+00	9.95E-01	1.188238	0.689087	63.29%
1.1	5.00E+00	4.00E-01	2.000184	1.19E+00	1.09E+00	1.30772	0.692464	65.38%
1.2	5.00E+00	4.25E-01	2.122919	1.19E+00	1.19E+00	1.427102	0.695818	67.22%
1.3	5.00E+00	4.49E-01	2.245653	1.19E+00	1.29E+00	1.546399	0.699253	68.86%
1.4	5.00E+00	4.74E-01	2.36933	1.19E+00	1.40E+00	1.665803	0.703527	70.31%
1.5	5.00E+00	4.99E-01	2.49273	1.19E+00	1.50E+00	1.785083	0.707647	71.61%
1.6	5.00E+00	5.23E-01	2.616812	1.19E+00	1.60E+00	1.904417	0.712395	72.78%
1.7	5.00E+00	5.48E-01	2.741666	1.19E+00	1.70E+00	2.023564	0.718101	73.81%
1.8	5.00E+00	5.73E-01	2.866866	1.19E+00	1.80E+00	2.142845	0.724021	74.75%
1.9	5.00E+00	5.98E-01	2.992312	1.19E+00	1.90E+00	2.261874	0.730438	75.59%
2	5.00E+00	6.24E-01	3.118952	1.19E+00	2.00E+00	2.381115	0.737837	76.34%
2.1	5.00E+00	6.49E-01	3.245237	1.19E+00	2.10E+00	2.500116	0.745121	77.04%
2.2	5.00E+00	6.74E-01	3.37153	1.19E+00	2.20E+00	2.618698	0.752832	77.67%

- Open a copy of the file Efficiency Template.xls included in the program folder, and paste the results to plot the curve:





North American Headquarters

SIGLENT Technologies NA
6557 Cochran Rd Solon, Ohio 44139
Tel: 440-398-5800
Toll Free: 877-515-5551
Fax: 440-399-1211
info@siglent.com
www.siglentamerica.com/

European Sales Offices

SIGLENT TECHNOLOGIES GERMANY GmbH
Staetzlinger Str. 70
86165 Augsburg, Germany
Tel: +49(0)-821-666 0 111 0
Fax: +49(0)-821-666 0 111 22
info-eu@siglent.com
www.siglenteu.com

Asian Headquarters

SIGLENT TECHNOLOGIES CO., LTD.
Blog No.4 & No.5, Antongda Industrial Zone,
3rd Liuxian Road, Bao'an District,
Shenzhen, 518101, China.
Tel: + 86 755 3661 5186
Fax: + 86 755 3359 1582
sales@siglent.com
www.siglent.com/ens