19xx- and 29xx-C/R[1]: How to Write Basic Code with Power Meter Labview Drivers

Step 1: Install Computer Interface software from the web

- The latest version can be found under the product page for respective power meter you own
- Software Downloads are found in the "Literature & Downloads" section
- Download and Install *Computer Interface Software*
- Note: Do not connect the device to your computer before the install and make sure you select the proper version to install (i.e. 32bit, 64bit or 32bit on 64bit)

Step 2: Power on and connect Power Meter

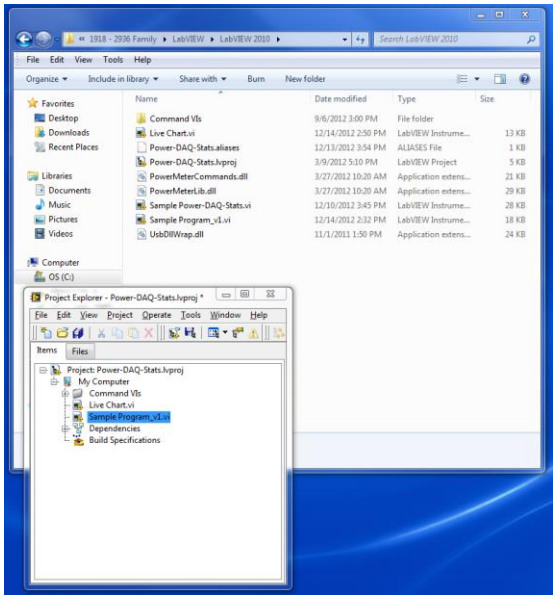
- Connect the desired detector to power meter
- Turn on power meter
- Connect the power meter via USB to your computer

Step 3: Check the connectivity

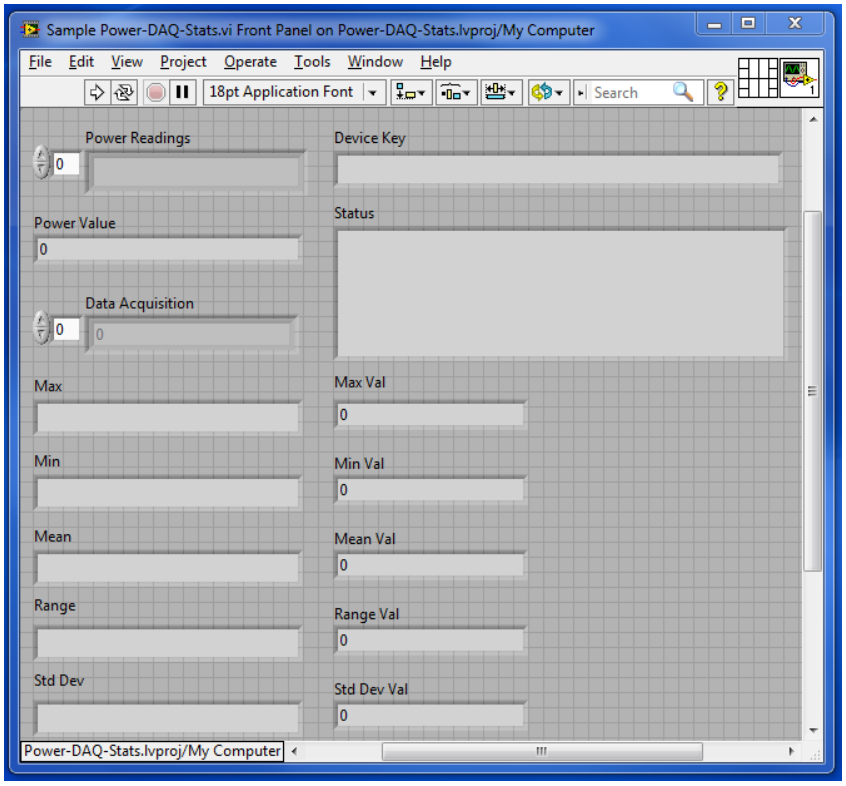
- Easiest way to do this is to run the Power Meter Application EXE
  - This can be found under Programs>Newport>Power Meter Application
- As long as the power meter application connects and runs properly, you have all of the drivers installed properly to run the Labview drivers.

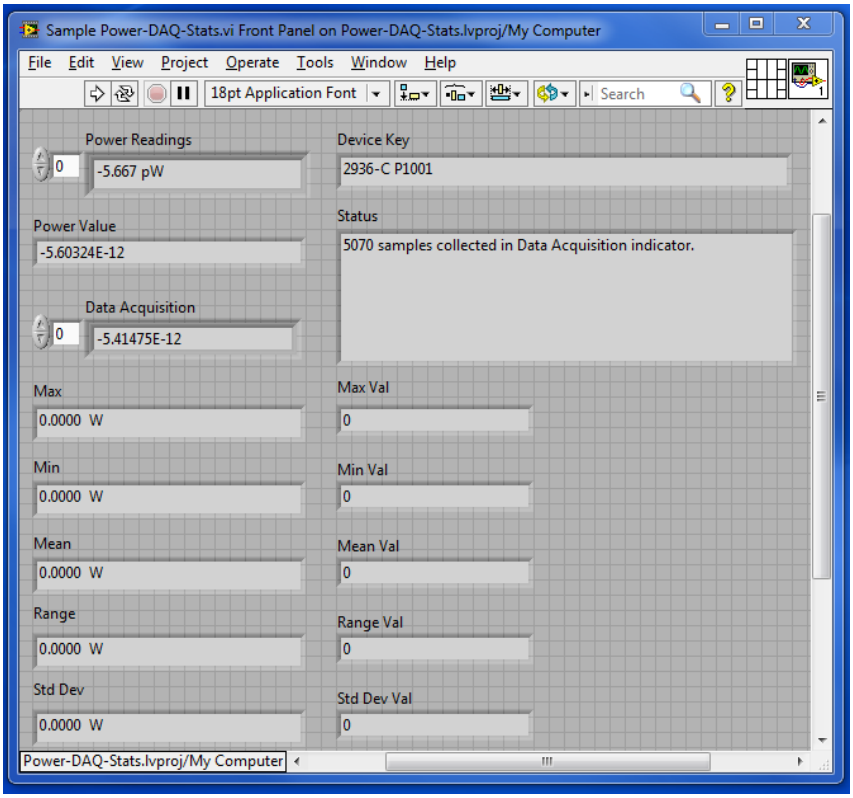
Step 4: Run Sample Labview Driver

- The sample drivers can be found here:
  - C:\Program Files\Newport\Newport Power Meter Application\Samples
  - Select model number family folder
  - Select Labview
    - Note C# Drivers are also located here
  - Select Labview Version you are running

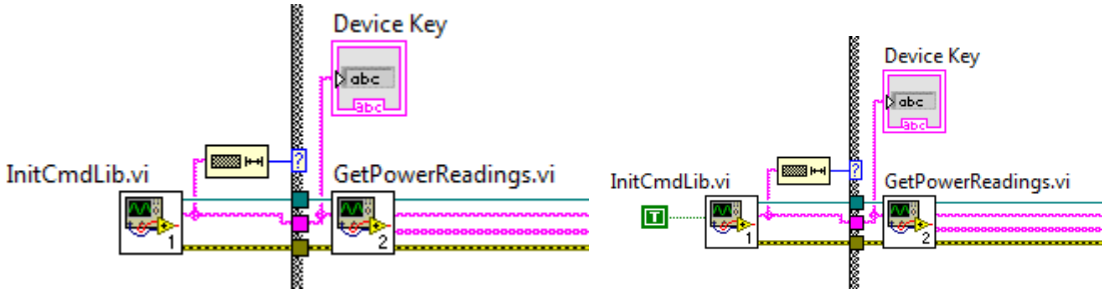


- Open: *Power-DAQ-Stats.lvproj*
- Open: *Sample Power-DAQ-Stats.vi* from this project
- Run: *Sample Power-DAQ-Stats.vi* from this project
  - The program will auto-detect the power meter
  - No other parameters need to be entered


Sample Power-DAQ-Stats.vi Front Panel on Power-DAQ-Stats.lvproj/My Computer
File Edit View Project Operate Tools Window Help
18pt Application Font
Search
Power Readings
Device Key
Power Value
Status
Data Acquisition
Max
Max Val
Min
Min Val
Mean
Mean Val
Range
Range Val
Std Dev
Std Dev Val
Power-DAQ-Stats.lvproj/My Computer


Sample Power-DAQ-Stats.vi Front Panel on Power-DAQ-Stats.lvproj/My Computer
File Edit View Project Operate Tools Window Help
18pt Application Font
Search
Power Readings
-5.667 pW
Device Key
2936-C P1001
Power Value
-5.60324E-12
Status
5070 samples collected in Data Acquisition indicator.
Data Acquisition
-5.41475E-12
Max
0.0000 W
Max Val
Min
0.0000 W
Min Val
Mean
0.0000 W
Mean Val
Range
0.0000 W
Range Val
Std Dev
0.0000 W
Std Dev Val
Power-DAQ-Stats.lvproj/My Computer

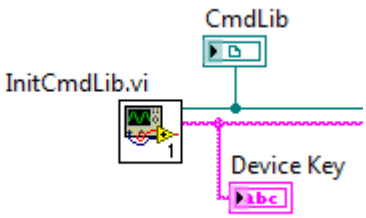
- If the program runs with no faults you are ready to customize you own software. It will look like the above image
- Troubleshooting:
    - You can enable logging from the *InitCmdLib.vi* SubVI to trouble shoot
        - The log files can be found in My Documents>Log



    - If the Labview doesn't work but the Power Meter Application Works, ensure you installed to proper version that matches your Labview (i.e. 64bit, 32 bit, or 32bit on 64bit)

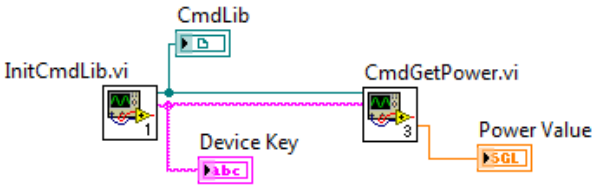
Step 5: Customize your own program

- The Command Vis folder contains the SubVIs you will need for integration
- These can be simply copied from the folder into your Block Diagram to build you VI.
- The folder path could look like this:
    - C:\Program Files\Newport\Newport Power Meter Application\Samples\1918 - 2936 Family\LabVIEW\LabVIEW 2010\Command Vis
- Every code will need the *InitCmdLib.vi* to get the power meter initialized
- From there you will need to string the *CmdLib* out and the *Device Key* out to the input on the other SubVIs you want to use

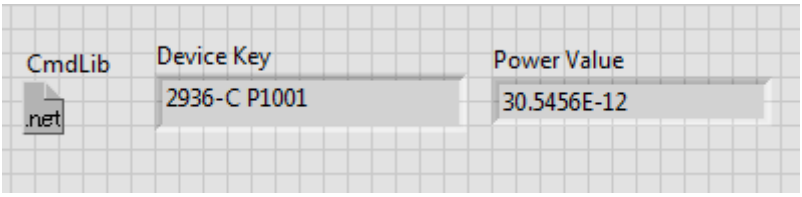


Step 6: Example of Custom Live Chart VI

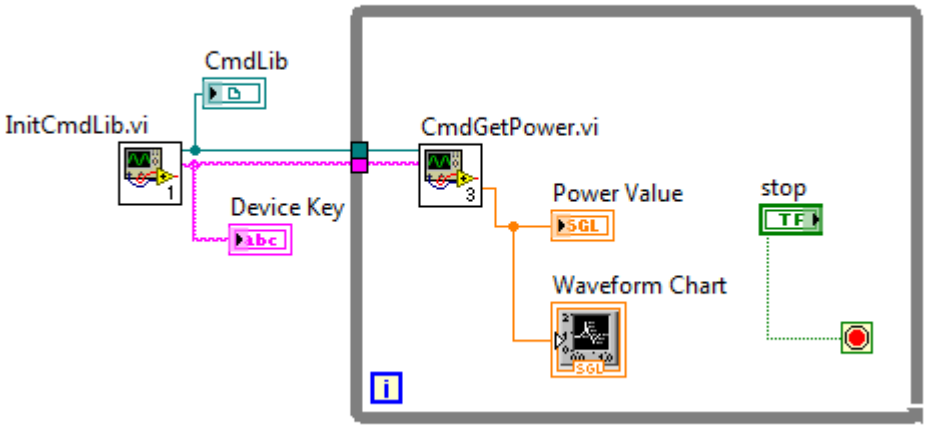
- Wire the *CmdLib* and *Device Key* strings to the *CmdGetPower.vi* from the *InitCmdLib.vi*.
- Display an Indicator for the *Power Value* to get a basic power display whenever you want to run the program. Should look like this:



- The Front Panel would look like this after running:

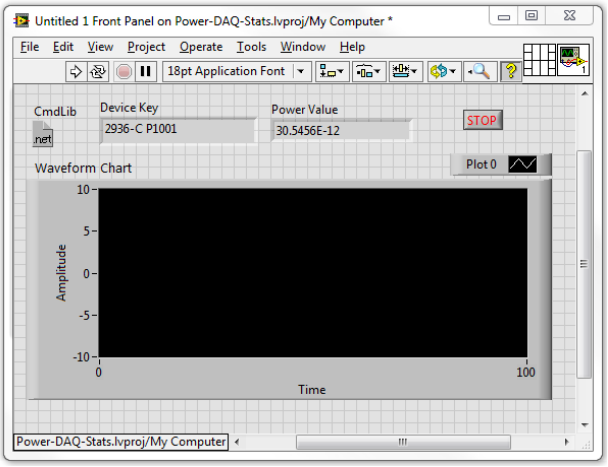


- Now, inserting a Waveform Chart and a While Loop will give you a live chart while the VI is running.
    - Note: You only need to Initialize once
        - After Inizilizing, you can simply use the strings the InitCmdLib.vi SubVI creates directly in your program
    - This is why the While Loop is only around the CmdGetPower.vi and Waveform Chart

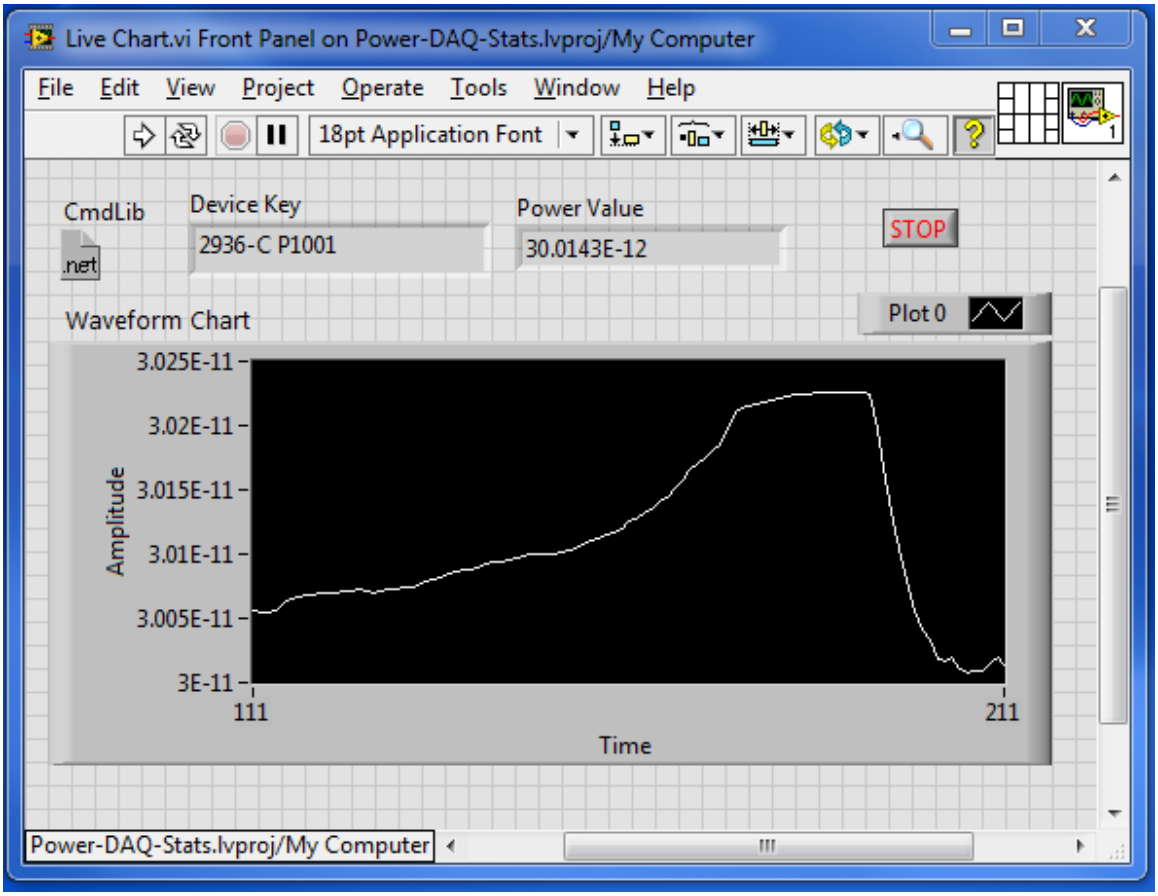


- To stop the program, click the stop button linked to the stop button in your VI.

- The Front Panel will look something like this:



- Running the program will display a nice live chart like this after some time:



- This is just the basics of building a custom program. Continue adding SubVIs to develop more advanced custom programs. Integrate these VIs with drivers from some of your other instruments to create even more powerful programs.

Appendix, List of Commands for Power Meters:


CmdGetAcquisitionMode.vi


CmdGetAnalogFilter.vi


CmdGetAttenuatorEnable.vi


CmdGetAutoRangeEnable.vi


CmdGetDigitalFilter.vi


CmdGetFilterType.vi


CmdGetIdentification.vi


CmdGetPower.vi


CmdGetStatsMaxMinusMinValue.vi


CmdGetStatsMaxValue.vi


CmdGetStatsMeanValue.vi


CmdGetStatsMinValue.vi


CmdGetStatsStdDevValue.vi


CmdGetUnits.vi


CmdGetWavelength.vi


CmdGetZeroValue.vi


CmdSetAcquisitionMode.vi


CmdSetAnalogFilter.vi


CmdSetAttenuatorEnable.vi


CmdSetAutoRangeEnable.vi


CmdSetDigitalFilter.vi


CmdSetFilterType.vi


CmdSetRange.vi


CmdSetUnits.vi


CmdSetWavelength.vi


CmdSetZeroValue.vi


GetFirstDeviceKey.vi


GetFrequency.vi


GetFrequencyValue.vi


GetNumChannels.vi


GetPowerReadings.vi


GetSampleCount.vi


GetStatsMax.vi


GetStatsMean.vi


GetStatsMin.vi


GetStatsRange.vi


GetStatsStdDev.vi


GetZeroOffset.vi


InitCmdLib.vi


PerformDAQ.vi


SetChannel.vi