Analog and Digital I/O

PHYS 3420: Data Acquisition and Analysis

© 2013, John E. Sohl, Department of Physics, Weber State University

Revised January 28, 2013

Objectives

- Measure voltages
- Produce controllable analog output voltages
- Create a function generator for a sine wave function
- Control some external device such as an LED
- Measure light with a photodiode

Figure 1.1 Our DAQ board, the miniLAB 1008.

Equipment

Computer with LabVIEW
miniLAB 1008 USB DAQ system
Oscilloscope and DMM
Power supply

Function Generator
LEDs
Resistors
Photodiode

Background

The miniLAB 1008 is a low end USB based DAQ board. It is adequate enough for learning how to collect data, but it is limited in speed, resolution and power. Many of these limitations are common with DAQ boards. For example, the analog output is 0 to 5 volts but it can only produce 30 mA per channel. If your system needs more power then you need to build a transistor or op amp driver.

The interface to the hardware is done with the "Universal Library for LabVIEW" made by MCC (Measurement Computing Corporation). You can find all these VIs in the MC ULx function library as shown in Figure 2. You can also find lots of example files specific for the MCC Universal Library by looking in the following location on your hard drive
C:\Program Files (x86)\National Instruments\LabVIEW 2012\examples\ULx

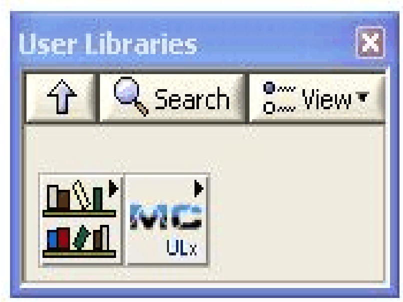

Figure 2. Location of the MCC Universal Library.

Procedure

Simple Sine Wave Generator.

The Measurement Computing Universal Library is the collection of vi's designed to run their data acquisition hardware. The old version was simply known as UL for LabVIEW and was a large collection of pretty basic functions. While easy to use, they were not terribly powerful. The new software is called "ULx" and has some impressive built in power. However, they are not as easy to use without some background. First of all, there is a new "task" line that runs through all the DAQ functions. This has several purposes, first of all it creates a data dependency that forces the set up steps to be completed before the data acquisition hardware can be called. This is a good thing. Secondly, and more important, the "task" data line contains all the information about the set up for the task you are trying to accomplish.

To build the simplest possible sine wave generator we will do the following sequence:

1. Define the hardware (i.e., set up the channels/features needed on the equipment).
2. Create a signal.
3. Write the signal to the hardware.

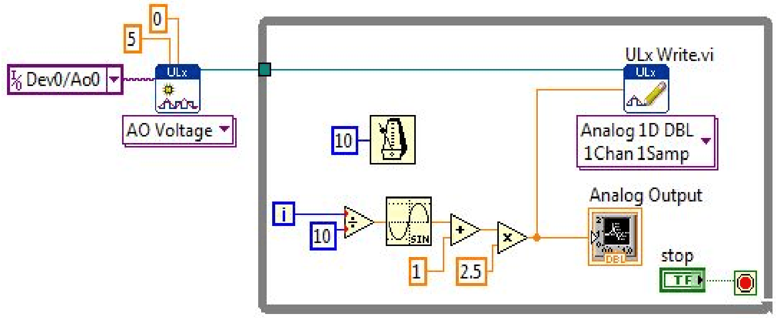

Figure 3. A very simple analog output sine-wave generator.

Your vi should look like Figure 3 when you get done with the following steps. This VI is not very good but it is a nice introduction to the process. Save this as "*yourname* Simple Sine Generator."

4. Start by placing a data acquisition vi found in Functions >> User Libraries >> MC ULx >> Create Virtual Channel. Once you have selected this you need to click on the drop down menu at the bottom of the vi and select Analog Output then select Voltage. Now you need to tell it which pins the signal will be coming from, this is called a "physical channel." Create a constant (or control if you want) and then select device 0, analog output 0 from the drop down menu. On the miniLab 1008 this is the screw terminal labeled "D/A OUT 0."
5. Create a signal. I don't care how you do it, but the signal output must be in the range of 0V to 5V since your analog output device is constrained to that range. Adding a waveform chart is helpful.
6. Write the signal to the hardware. From the MC ULx function pallette select the ULxWrite.vi and configure it to write a single data point to a single channel.

Connect the miniLab 1008 to the USB port. Run your signal generator. If you are having problems it might be that your hardware is not registered to the computer. If so, run the Measurement Computing Instacal program and configure your hardware. **Connect an oscilloscope to the output and capture a display to show your instructor.**

Notice that a curious thing happens when you stop the vi. The output holds at the last value. This can be good or bad depending on your needs. You have to remember that if you want it to output zero or some other voltage when stopped, you need to program that in.

Once your vi is running correctly, ruin it and see what happens. Change the minimum value on the "Create Virtual Channel.vi" to -5V and run the program. Fix that and change the signal you generated to run outside the 0V to 5V range. Notice the helpful popup windows. This is a huge improvement over the previous Universal Library that would simply crash and not tell you why unless you set up error message indicators.

Note that the miniLAB 1008 is limited in speed. The analog out channel is specified for a max speed of 100 samples per second (single channel mode, only 50 S/s in dual channel mode). So don't get rid of your real bench top function generator just yet...

N.B., by using a function node in place of the sine VI you can get any arbitrary function to come out of the analog output. (If you try this, an easy and quick function to try is a sum or product of sine waves.)

Control Something

It is time to have a bit of fun. Place a red LED in series with a 150 Ω resistor. Place this series circuit between one of the analog out channel outputs and ground. Select various periods for your sine wave generator. Your LED should flash with a ramping up and down brightness. (Think about the orientation of the LED, if it isn't flashing you might have it in backwards.)

If you wish, you can connect a different color of LED, watch your choice of resistor so that you don't run more than about 20 mA through the LED. (You can use the same resistor value as you did for the red LED, it just won't be as bright.)

Another fast and fun optional thing to try is to use both analog output channels zero and one. Flash the two LEDs at different rates.

Measure Something

Create the VI shown in Figure 4.

Your DAQ board (the miniLAB) is configured for differential mode input. Thus, the input for channel 0 is actually pins 1 and 2 which are labeled as "CH0 IN" and "CH1 IN" on the board. Pin 1 is the "HI" or plus side and pin 2 is the "LO" or negative side like you would have on a digital multimeter.

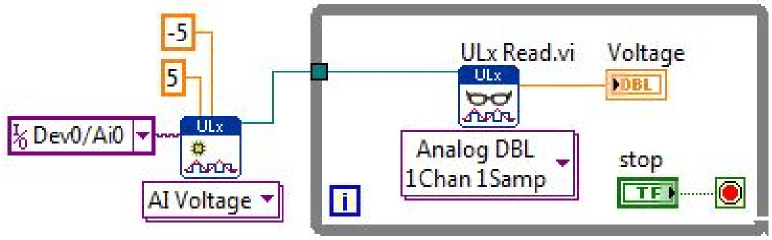

Figure 4. A simple analog input vi.

For starters connect the bench power supply to Channel 0 of the DAQ board. Connect the +V to pin 1 and ground to pin 2. Play with the supply while taking data and watching the screen. Change the input voltage range and see what happens. You might want to look at the datasheet for the miniLAB to see what ranges are available. (Try a range that is NOT available.)

Now replace the power supply with a photodiode connected between the two pins of the Analog-to-Digital input. Measure the room light and block it with your hand. Adjust the input range as needed for a good signal.

Measure the light output of a lamp

We will use the photodiode to measure the light from a lamp. This is especially interesting when turning the lamp on and off. Place the photodiode such that it faces directly towards an **incandescent** lamp.

Modify your VI from Figure 4 to save the data in a spreadsheet file. Try to make it run as fast as you can, that is, don't put any extra stuff in the While or For loop that you don't need. Save the data file as "*yourname*-Light.xls" Name the tab in your Excel file as "Slow."

Plot your data in Excel and look closely at the period. Chances are good that the image will have a sine-wave like shape at the top of a DC offset light curve. Measure the period and note it on your Excel graph.

Higher Speed Measurements

Change your analog in VI to one that runs high-speed scans in the foreground. Running in the foreground means that everything else pretty much stops while the device collects data.

Create a vi and save it as "*yourname*-Light-NSamples.vi" where you start with what is shown in Figure 5. Read through the "detailed help" file on the function of the various inputs and the settings that

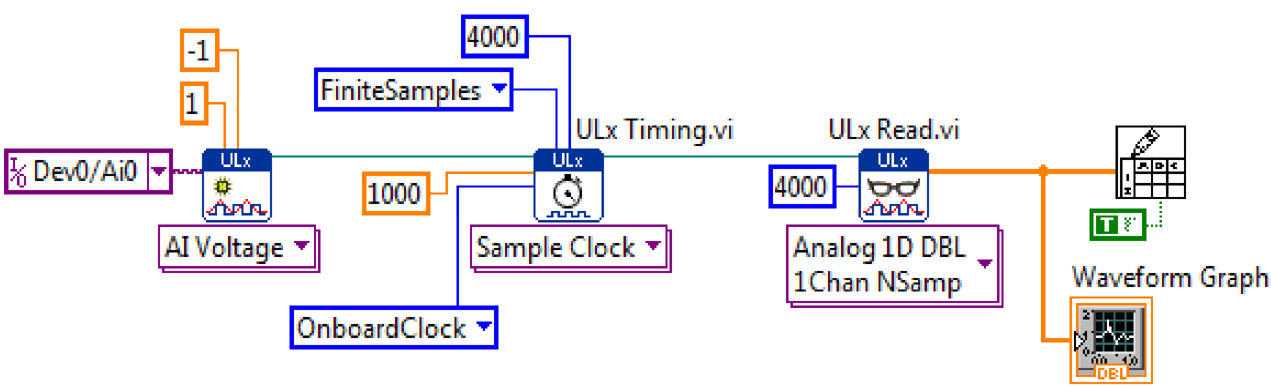

Figure 5. Starting point for having the miniLab 1008 run at a faster rate to collect data on the turn on of a lamp. The top right vi is Programming >> File I/O >> Write to Spreadsheet File.vi.

are available. Here is an overview of what is happening in this vi.

The first function sets up the type of reading you are making (Analog Input, Voltage). The next function sets up the clock that will be used to pace the data collection, the OnboardClock uses the clock that is built into the miniLab 1008. The "1000" is the number of samples per second (recall that the miniLab 1008 can take data no faster than a rate of 1.2kS/s =1200 samples per second). The next function (ULx Read) will collect 4000 samples and then output that result to the spreadsheet and waveform graph. Keep in mind that the miniLab 1008 has a 4k buffer, so it cannot take more than 4000 data points with this configuration.

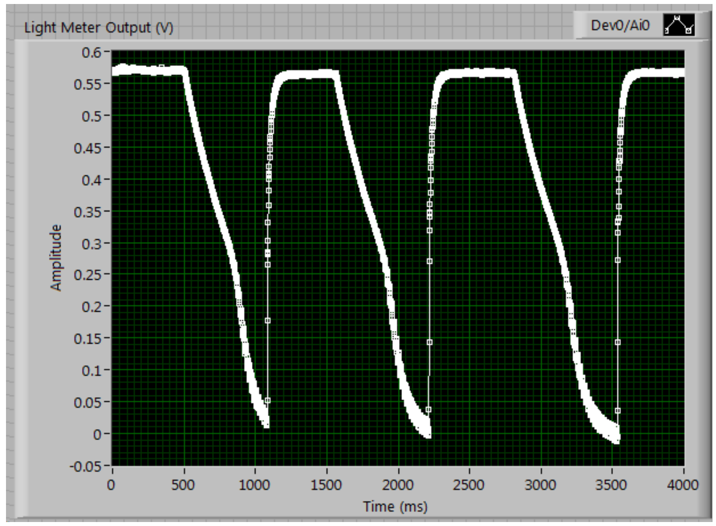

Figure 6. The simple front panel for the vi in Figure 5.

An example of the front panel, with some data collected from my desk lamp at home, is shown in Figure 6. A more advanced front panel is shown in Figure 7. In that version I had a waveform chart that gives you the live output so you can adjust your system. Once you are happy with the behavior you can press the "Collect Data" button and a vi very similar to that shown in Figure 5 runs which will then actually collect and save your data. I then swap out the waveform chart for a waveform graph to display the data that were collected. You don't need to go to this much effort, but it was fun.

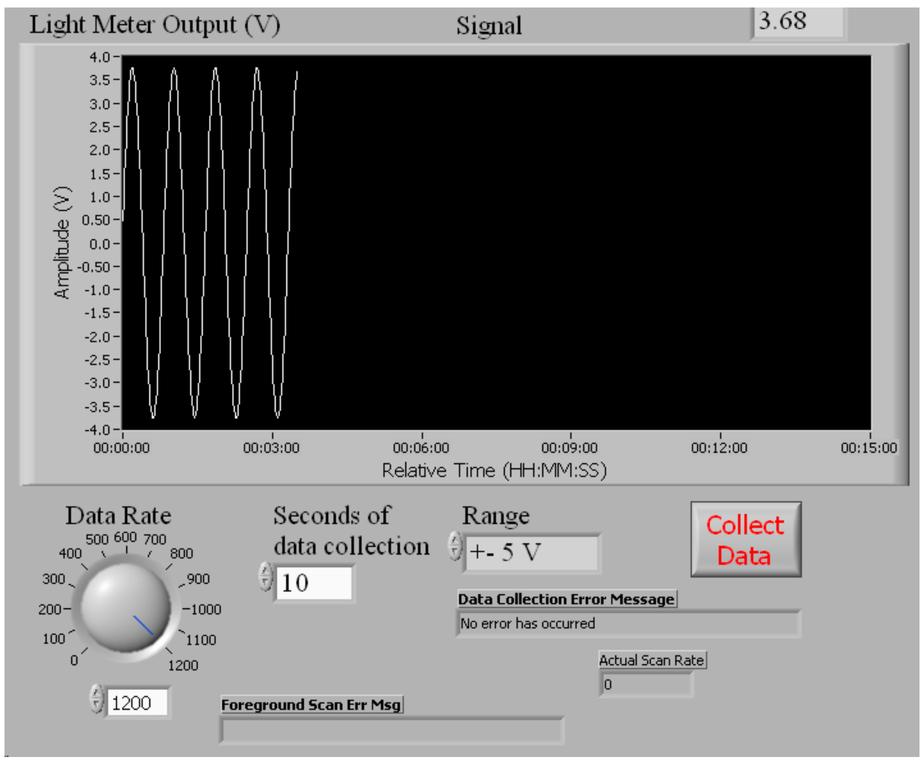

Figure 7. My front panel for the foreground scan. Your panel does not need to look like this to accomplish the task at hand.

Run the program and collect data for 4 seconds and save your data to an Excel file. Once you get something you are happy with save that data and associated plots in the Excel file you created earlier with the slower data collection rate. Place this new data in its own worksheet tab. Label this tab, "fast."

Your data should look something like the plots shown in Figure 8.

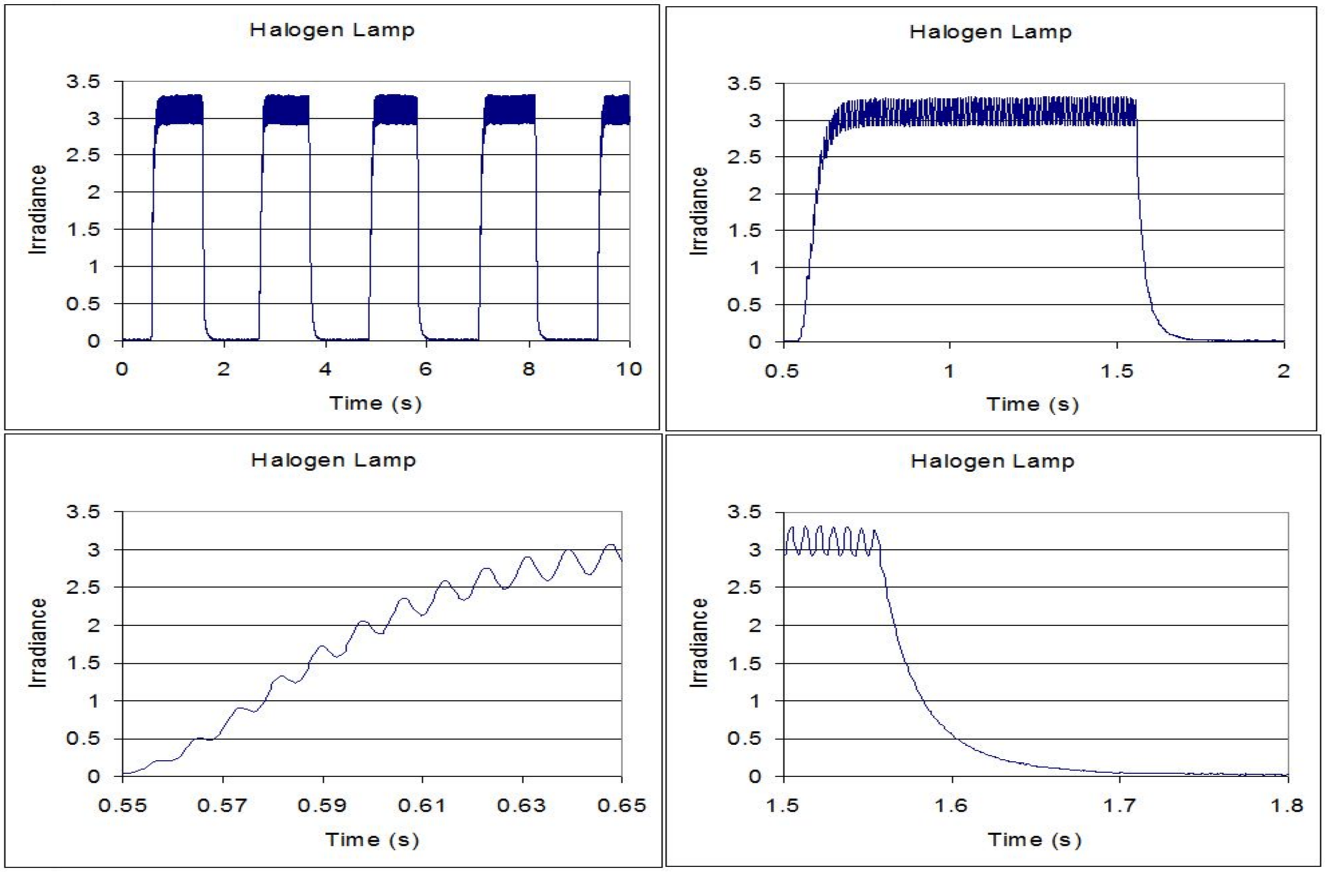

Figure 8. The entire data set is shown in the top left plot. Look carefully at the axes and you'll see that the remaining plots are just zooming in on regions from the first plot. In the bottom left notice the AC power effect while the filament is heating up. In the bottom right plot you can observe Newton cooling. In all cases the irradiance is in arbitrary units. Also, I changed the vertical scale and shifted it to place zero light at zero on the irradiance scale.

Once you have your system working correctly, try measuring the on-off behavior of other lamps such as fluorescent lamps and LED's. A neon lamp is interesting to observe. Try looking at things like laser pointers. (Green laser pointers have a curious and repeatable power up intensity time profile; ask me if you want to know why?)

Questions: What is the frequency of the signal you observe for the LED and fluorescent room lamps? Why is it ***not*** 60 Hz?

Enter your data and do plots on different worksheet tabs inside the same Excel file you created above. Answer the questions above in some obvious place within the Excel file near the top of a worksheet page. Also, locate your plots at the tops of the pages.