

Instrumentation & Interfacing: "LabVIEW" (Phys 363) Spring 2012



What am I doing here? LabVIEW is a graphical programming language used to create data acquisition and control interfaces. In other words, it is used to enable a computer to control other real-world hardware. As a result of completing this course, you will be able to create control and acquisition programs using LabVIEW, you will be able to interface the LabVIEW computer to external hardware using at least three protocols. We are using LabVIEW 2010.

Where can I get a textbook? The LabVIEW software comes with excellent documentation, which we will consult in place of a textbook. There are also a number of LabVIEW manuals and texts available in ISC 226 for your use. Please do not remove them from that room. If you want a book of your own, you might this one, which includes an actual working copy of LabVIEW:
LabVIEW 2009 Student Edition, by R. H. Bishop ISBN 978-0132141291

If you choose one of the first two options, you can buy it directly from National Instruments, the maker of LabVIEW, at www.ni.com.

How will I be graded? Your grade will be determined by:

Weekly Assignments and quizzes	60%
Project Stages	10%
Final Project Submission	30%

Note: Submission of a functional final project is mandatory to pass this class.

How will I submit assignments? You must drag/copy the code for your weekly assignments into my inbox: \\files\Inbox\Physics\Pogo. A link to this address can also be found on my home page. Or, you can get there from the RUN... menu on your PC. Assignments **must** be titled **##-LastnameFirstInitial.vi**. So, my third assignment would be called "03-PogoE.vi". Sometimes, you may have to add extra files, too. Be careful: once an assignment has been placed into the inbox, it cannot be retrieved, deleted, or changed! Emailed assignments will not be accepted.

What is required for the final project? Final projects must involve interfacing a computer with external device(s). The LabVIEW control system must include active feedback: it must make some measurement(s), and then adjust the timing or nature of the next measurement based on the result(s). Outputs should be non-trivial; in the ideal case, they will involve motion. Simple automation of data-acquisition is not sufficient. State machines are also unacceptable. Also, some portion of the final output of the system must be permanent (i.e., stored in a file rather than merely displayed on-screen). Finals projects should be debugged and "idiot-proof" (i.e., they shouldn't crash if the user makes an input error).

What are these “stages” for the project? Projects will be completed and submitted in stages:

- Stage 1: An abstract of your project’s main idea. It is *your* responsibility to find an idea, not mine!
- Stage 2: A complete written proposal. The project proposal must be a short professional report describing the scope of your project. It must include the following sections:
 - a. The finalized abstract.
 - b. A detailed list of all the hardware you expect to use (including specifics for simple stuff such as tape, wires and cables!).
 - c. Sketches of any custom hardware you will construct or assemble.
 - d. A detailed summary of the intended user input. A sketch or screen shot of the user interface should be included.
 - e. A detailed summary of the project output. Again, a screen shot and an example listing from any output text files to be generated should be included.
 - f. A description of the measurement(s) the system will make, including their frequency (e.g., twice each minute), and any unit scales (e.g., 1 volt ↔ 10cm).
 - g. A flowchart diagramming the flow of information in the system, including an **explicit** indication of the necessary feedback.
- Stage 3: A detailed checklist (see also stage 2b) verifying that all the equipment has been obtained.
- Stage 4: An in-class demonstration that each element of external hardware can, at a minimum, send or receive data from the computer, as appropriate.
- Stage 5: A prototype of the final project and front panel. Everything must work to some extent, but some things will not yet be smooth and perfect.

If any of the original sections of the proposal are found to require modifications, an explanation of the changes and their reason is required simultaneously with the next stage. Any stage may be completed early.

Schedule of Assignments

Date	Assignments Due	Format
Tuesday, January 24, 2012	Assignment #1	inbox
Tuesday, January 31, 2012	Assignment #2	inbox
Tuesday, February 7, 2012	Assignment #3	inbox
Thursday, February 16, 2012	Assignment #4	inbox
Thursday, February 23, 2012	Assignment #5	inbox
Thursday, March 8, 2012	Assignment #6	inbox
Friday, March 9, 2012	Project Stage 1: Abstract	inbox
Thursday, March 22, 2012	Assignment #7	inbox
Tuesday, March 27, 2012	Project Stage 2: Full Proposal	inbox & hardcopy
Tuesday, April 3, 2012	Project Stage 3: Equipment Inventory	hardcopy
Tuesday, April 10, 2012	Project Stage 4: Signal Check	in-class demo
Thursday, April 26, 2012	Project Stage 5: Prototype Demonstration	in-class demo
Tuesday, May 1, 2012	Completed, debugged Final Projects	inbox

Learning Outcomes

At the end of this course, students will:

- Be able to design, create, and debug instrument control and acquisition programs using LabVIEW
- Be able to use LabVIEW to interface your computer to instruments using at least three protocols (analog DAQ, digital DAQ, and GPIB). In addition, you may learn other protocols (e.g., serial).
- Design, create, and debug complex systems of instruments combining data input and output to work with your computer.