Incorporating LabVIEW with Traditional Circuit Analysis

Randy K. Buchanan¹ and Giancarlo Milano²

<u>Abstract</u>

Due to the continual growth of technology in the industrial world, corresponding changes in courses within Electronics Engineering Technology curricula are imperative. Improvement of course topics and different approaches to laboratory experimentation are essential methods of relating lecture and hands-on experience in laboratories to present-day, real-world applications. The Electronics Engineering Technology program at the University of Southern Mississippi has recently integrated Laboratory Virtual Instrument Engineering Workbench (LabVIEW) into its instrumentation course. Fundamental LabVIEW exercises were performed in the lab, as well as exercises that used LabVIEW incorporated with traditional instrumentation.

Initially, LabVIEW exercises were dedicated to establishing a basic understanding of the software functionality. Once students became familiar with the software, they began to view LabVIEW as an extension of electronic instrumentation. Although most students appeared skeptical at the beginning of the semester, ultimately they expressed not only an understanding of the role software such as LabVIEW plays in the realm of instrumentation, but also an awareness of the power such tools may hold. In following semesters, some students within the instrumentation class chose to use LabVIEW as an integral component of their senior design projects.

Introduction

Electronic instrumentation is ever changing in accordance with advancements in technology. During the spring 2003 semester, in an effort to respond to this changing technology, an initiative was undertaken to revise an instrumentation course within the Electronics Engineering Technology (EET) curriculum at the University of Southern Mississippi (USM). Due to the growing prominence of computerized instrumentation in industry, it was deemed applicable to incorporate this trend into the electronics curriculum. Prevalent instrumentation software called Laboratory Virtual Instrumentation Workbench (LabVIEW) was chosen to be used to accomplish this task. LabVIEW was introduced to sophomore students in the program by combining the existing traditional concepts in the course with the new capabilities offered by the instrumentation software. LabVIEW was introduced to the students within the laboratory sessions of the course.

As a requirement of the course, students were required to purchase their own versions of the software with accompanying text. Initially students were simply encouraged to become familiar with the software, and learn its basic functionality. LabVIEW exercises and traditional instrumentation labs were preformed both individually and then together toward the end of the course. In some instances, several established laboratory exercises were combined in order to allow time to perform the new exercises. The ultimate goal was the integration of the traditional instrumentation with LabVIEW. To accomplish this integration, real-time simulation instrumentation was designed, manipulated, and from it data was collected. Simulation was the chosen as the goal due to only one computer having the data acquisition (DAQ) card necessary at the time to collect data. Therefore, the students ran their Virtual Instrumentation (VI) programs on the single station at various stages and alternate schedules.

The LabVIEW Environment

LabVIEW is a graphical programming language, as opposed to a text-based language, and is used to create programs in a graphical block diagram form. Within the program is an extensive library of functions including data acquisition libraries, communication, control, and methods for data acquisition, presentation, and storage.

¹ Electronics Engineering Technology, University of Southern Mississippi, Hattiesburg, MS 39406

² Electronics Engineering Technology, University of Southern Mississippi, Hattiesburg, MS 39406

LabVIEW programs are referred to as virtual instruments (VIs). This label indicates that virtual instruments mirror the operation and appearance of actual instruments. These programs are comprised of a user interface known as the front panel, and a diagram area that contains the actual graphical program. The panel typically simulates the front panel of a physical instrument and provides a method of inputting and displaying data and information about the process. Programmed instructions that are given to the virtual instrument are in the form of a graphical block diagram. Within the block diagram, virtual wires connect objects that send or receive data, perform predefined functions, and control the overall flow of execution.

Students Introduced to LabVIEW

Students were required to purchase the LabVIEW Student Edition textbook and software for the Instrumentation Laboratory Course. During the laboratory sessions, LabVIEW was slowly introduced with introductory level exercises. Basic concepts such as an introduction to menus, editing, subroutines, structures, arrays and clusters, charts and graphs, data acquisition, strings and files, etc. were covered. Students soon learned that LabVIEW was a tool they could use to easily design their own virtual instruments to collect and manipulate data. In order for the students to become familiar with the new software and adapt it to the instrumentation course, the semester was divided in three major components. A portion of the semester focused on the familiarization with the LabVIEW environment, another on familiarization with standard instrumentation applications, and the remainder with integration of standard instrumentation with LabVIEW virtual instrument applications.

Although initial assignments were given to lead the student in learning the basic concepts, each chapter of the manual introduced new and more advanced tools. Within each of the sections, several exercises, problems and study cases were given to the students so they could use these lessons in performing useful and fun tasks. For each of the chapters introduced, and for evaluation purposes, an informal report was required for submission one week after the lab meeting took place. Within this report, printouts of the VIs, answers to the questions regarding the exercise section of the book, and a brief discussion and conclusion of the laboratory were required to be included. The discussion and conclusion section of this laboratory report also required several important subsections, such as description of the new tools that were introduced in the chapter, how could these tools become beneficial in real life instrumentation applications, and what they learned from the chapter. Although the various concepts that were introduced were simplistic in nature, they demonstrated how useful and straightforward it was to design practical programs for different applications.

Traditional Instrumentation and LabVIEW

The core of the instrumentation course centered on electronic instrumentation. In realization that LabVIEW was simply an application of existing instrumentation technology and computerization, it was important that the principals of traditional instrumentation not be lost in the process. Upon examination of preexisting laboratory exercises, it was concluded that some of the exercises could be combined, while maintaining a solid coverage of the material. For example, the major components of two separate ac voltmeter lab exercises were combined into one comprehensive lab. Similar consolidation was applied toward other instrumentation topics. The laboratory exercises were rewritten to preserve the important aspects of the labs, while still allowing the exercises to be completed in reasonable time.

Topics such as standard instrumentation circuits, equipment, measurement accuracy, and analog and digital measuring devices were experienced. As the students progressed through the semester, they learned increasingly difficult methods in creating useful virtual instrumentation programs. As more was discovered and learned, students became further motivated and quite enthusiastic about the software. Near the end of the semester a final project was used to coalesce the LabVIEW design knowledge learned throughout the semester with standard instrumentation applications.

Final Integration Project

The purpose of the final project was to bring to life the amalgamation of traditional instrumentation with the world of virtual instrumentation. To allow the student a straightforward way of witnessing LabVIEW work with electronics familiar to them, a dc circuit analysis simulation exercise was assigned. The focus was to create a real-

time simulation application where the student could tangibly observe the method by which data was acquired by LabVIEW and then manipulated based on rules programmed by the designer. In this case, the rules were based on traditional circuit analysis methods. The project assignment, entitled "Instrumental Circuit Analysis using LabVIEW", was divided into three main procedural sections: configuring a data acquisition board to interface with a LabVIEW program and a power supply, creating a virtual electronic circuit in LabVIEW, and producing documentation.

The beginning section required the student to monitor a direct current (dc) voltage level with LabVIEW using a data acquisition board, and display the voltage reading in the graphical user interface (GUI) panel window using an indicator display. The sample rate at which this value can be sampled depends on the DAQ card used, and in this case was a National Instruments 4060; a 5-1/2 Digit Digital Multi-Meter for Peripheral Component Interconnect (PCI). The sampling rate for this unit was 200K samples per second. [National Instruments, 1] Once the signal value was obtained, the student used the various functions within LabVIEW to manipulate the value and obtain the desired format, size, color, and precision of the display.

The second segment of the procedure involved creating a virtual electronic circuit, as shown in Figure 1. This resulted in the student producing a circuit and related mathematical and analytical manipulations needed to calculate desired values. Magnitudes of current through circuit paths within the circuit, as well as voltage drops across components were calculated. In the traditional sense, all that was required to perform the calculations was knowing the relationship between voltage, current, and resistance; or



 $\mathbf{E} = \mathbf{IR}.$

Figure 1. Electrical Circuit Diagram

A basic knowledge of fundamental circuit analysis was also required for a proper analysis of combination series/parallel circuitry. From this knowledge and the familiarity with some of LabVIEW's basic functions, the appropriate LabVIEW program was developed. The intent of this laboratory exercise was to create a simulation program that could be used by any user. Therefore, the circuit resistance values were not given to the students. This required the students to create a program that would allow user input of component values through a user interface panel, as shown in Figure 2. As a user would enter values, it was essential that the total resistance of the circuit, current through each component, and the voltage drop across each of the resistors would indicate the newly calculated value.



Figure 2. LabVIEW User Interface

Within the panel of the user interface, there were several parameters that were required for inclusion. For example, user control of component values such as the desired value of each resistor in the circuit was required. With this feature, the user could select or change the value of each of the resistors at any time during an experiment. The related indicators for each of the resistors displayed the current and voltage values for the given circuit configuration. Additionally, the total resistance of the circuit was displayed, as well as the value of the externally supplied dc voltage. An indicator was used to indicate the measured voltage value and a graph was created to plot the graphical representation of input voltage versus time.

To adequately represent the circuit, an actual schematic diagram was inserted inside the user panel window. Consequently, the corresponding current and voltage values for each of the resistors were placed near the appropriate circuit element. The near real-time graph depicting voltage as a function of time was also created for monitoring purposes.

In the process of creating controls and indicators on the front panel, corresponding programming icons were simultaneously created within the programming diagram window of the program. When the student created the resistance control variable, a small icon on representation of that control was created in the diagram window, as shown in Figure 3.



Figure 3. Front Panel Control and Corresponding Diagram Icon

The diagram window contains the programming necessary to perform the desired task at hand. It may be considered a block diagram of sorts, with its graphical block representation of programming functions and the various signal path types that are used to connect each of the blocks. The signal path types are called 'wires' and connect the blocks by carrying different types of information represented as different colors. The colors indicate the data type such as integer, floating point, Boolean, string, etc. The data type used for this experiment was a double precision integer, as indicated in Figure 3. Within the diagram, the quantities entered from the front panel, such as resistance values, were applied to the mathematical equations within the diagram to be used to calculate circuit values, as shown in Figure 4.

The student was required to perform manual analytical calculations to obtain theoretical reference values of the circuit. Once a simplified equation was derived that described the variables at given locations in the circuit, a value was assigned for each of the resistors and for the voltage supply. From these established values, the voltage and current for each of the elements were calculated and recorded. At a later time, the same resistor values for the circuit were entered into LabVIEW, and the student verified that the answers matched those values calculated in the traditional manner. A comparison of obtained values is shown is Table 1. Differences in values were calculated to be between nil and several percent, which typically equated to tens of millivolts or milliamperes.

While many students derived the same values, others obtained values from their program that differed from the calculated values. At this point, the student was instructed to go back and analyze both their program and the analytical calculations. The students were encouraged to assign reasonable values to the elements that would result in predictable outcomes when running the solutions in LabVIEW. By using this method, the student would think through the process and determine which of the solution values was not within the realm of acceptability. This provided an indication for the student as to where the problem with the program resided.

Once all values were determined to be correct, the students advanced to the third section of the laboratory exercise, which included writing a formal laboratory report. Within this report, all of the basic procedures, resulting data, discussion, and conclusions were required to be stated. Students were required to relate the virtual instrumentation experience with traditional circuitry and instrumentation. By requiring this analysis, it was hoped that students would reflect on the significance of what they learned, rather than just stepping through an experiment.



Figure 4. Programming Diagram Window

Table 1	L Com	parison	of An	alvtical	Values	vs.	LabVIEW	Values
I dole		Julibon	01 1 111	aryticar	, and op	· D• ·		, araco

Resistance	ve Voltage (V)					Current (mA)					
RT	VR1	VR2	VR4	VR5	IR2	IR3	IR7	IR4	IR5		
60.810	2.070	0.460	3.280	5.180	0.030	0.013	0.042	0.160	0.210		
			Tł	neoretical C	Calculation						
Resistance	Voltage (V)				Current (mA)						
RT	VR1	VR2	VR4	VR5	IR2	IR3	IR7	IR4	IR5		
60.810	2.055	0.451	3.249	5.138	0.030	0.013	0.043	0.162	0.206		
				Percent Di	fference						
			% Diff	$=\frac{Meas_1}{meas}$	$-Meas_2$ s_Avg	*100%					
			% Diff	$=\frac{Meas_1}{meas}$	$-Meas_2$ s_Avg	*100%					
Resistance		Volta	% Diff	$=\frac{Meas_1}{meas}$	- Meas ₂ s_Avg	*100%	Current (mA)			
Resistance RT	VR1	Volta VR2	% Diff ge (V) VR4	$= \frac{Meas_1}{meas}$	- Meas ₂ s _ Avg	*100%	Current (mA) IR4	IR5		

Conclusions

Electronic instrumentation has changed dramatically over the last several decades, yet from all indications, it appears that many instrumentation courses have not kept pace with the new trends. Changes implemented in the instrumentation course within the USM EET program were designed to help respond to the new trends. LabVIEW is a robust programming platform and is widely utilized throughout many industries. It is a standard for laboratory instrumentation software, and by introducing sophomore students to it, they will be further encouraged to investigate similar new technologies available. In turn, this should help make the students more marketable and valuable upon graduation. So that LabVIEW would be accepted as part of the EET curriculum and a valuable component of the students' overall skills, it was interfaced with the traditional concepts with which they were familiar.

Due to the somewhat flat initial learning curve for LabVIEW, students were encouraged to first become familiar with the software, and simply learn its basic functionality. However, after weeks of using the software and learning the principal functions, students appeared to accelerate at their own pace. The size of the programs as well as the need to learn the meanings of all the various icons could be considered disadvantages when using LabVIEW. However, the easy identification of programming methodologies, the commonsensical layout, and ease of editing proved to be major advantages compared to other programming environments.

To accomplish the integration of LabVIEW into the traditional instrumentation course, a traditional dc circuit proved to be an ideal proving ground. Although some of the preexisting traditional laboratories had to be deleted or combined to make room for the new software, the end result was observed to be worth the sacrifice. When the students started to work through the laboratory process at the beginning of the semester, many appeared not to realize the power of the particular software environment. However, as they worked through the exercises, it was observed that they began to relate their newly acquired knowledge with real-world applications, and they became more familiar and excited about their new undertaking. It was not until the students performed the interfacing lab that they had an opportunity to appreciate the application of LabVIEW to the world around them. Once the basic functionality was learned, the slope of the learning curve began to decrease, and students were observed to be able to learn merely by investigating features within the program.

By using LabVIEW, the students were able to begin to learn the general concepts of computerized instrumentation, data acquisition, control, automation, and system analysis. To have software available for constant updating of system measurements and to be able to view a system's status were observed to be very important and valuable assets for student learning. Near the end of the semester, the students appeared to have obtained a good grasp of LabVIEW's basic functionality, and were able to design, collect, and manipulate several types of data systems within the software. The final lab was success, in that it successfully tied traditional electronics to LabVIEW. Values obtained from running the LabVIEW simulation were equal or near equal the analytical values calculated by the students. These results appeared to bring about a new level of trust in the software by the students. Many of the students, after implementing and learning different concepts of LabVIEW, were so impressed with the new platform that they began to discuss ways to apply their newfound experience to their future senior project designs. This, in turn, added to the sense of accomplishment in the implementation of virtual instrumentation into a traditional instrumentation laboratory course.

<u>References</u>

[1] Retrieved February 04, 2003 from the World Wide Web: <u>http://www.ni.com/pdf/products/us/4mi428-430.pdf</u>

<u>Randy K. Buchanan</u>

Professor Randy Buchanan is Coordinator of Electronics Engineering Technology at the University of Southern Mississippi (USM) in Hattiesburg, MS. He has been on the faculty at 3 universities in 2 states, and directed projects at 4 different NASA centers. He is currently director of the NASA EPSCoR Mississippi research project at the USM Signal Research Center. He is the Chair for the ASEE SE Electrical Engineering Division and an Aerospace Industries division officer with the Instrumentation, Systems, and Automation Society. His areas of research and interests include control systems, scientific instrumentation, laboratory automation, space hardware processing. and planetary & space simulation. He recently received NASA Space Act Software Invention Award and a NASA Space Act Board Action Invention Award for his design and implementation of the Mars Electrostatics Chamber at Kennedy Space Center in Florida.

Giancarlo Milano

Giancarlo Milano received his B.S. Degree in Electronic Engineering Technology in 2003. Giancarlo is a Graduate Assistant in the School of Engineering Technology and currently assists with both research and academic tasks within the electronics program area. His areas of interests include control systems, instrumentation, fluid dynamics, and analytical analysis methods.