

Teaching Data Acquisition Using Laptop Computers

John A. Gumaer
Northern Michigan University

Abstract

This paper presents an inexpensive technique of using laptop computers to teach data acquisition skills to technology students. These students possess a laptop computer and have completed one semester of Visual Basic programming. The approach presented in this paper allows students to develop and integrate their understanding of both hardware and software concepts related to data acquisition. The main tools used are a laptop PC, parallel port adapter, and Visual Basic software. The parallel port adapter provides access to the parallel port of a laptop computer and is easily built by students. Students construct circuits for basic digital and analog input and output sources. These circuits interface to the laptop parallel port through the parallel port adapter. Students create Visual Basic applications to access and control the parallel port to acquire data. Using this setup, students are able to perform a variety of data acquisition and control experiments. The benefits of this approach are that students are able to develop their own data acquisition system and understand how the hardware and software work together. Used in conjunction with laptop computers, this technique is very portable and allows students to perform experiments in class and across campus without being tied to a data acquisition laboratory. By avoiding custom data acquisition hardware and software, the costs for this technique are minimal.

Introduction

This paper presents an inexpensive technique of using laptop computers to teach data acquisition skills to technology students. Traditionally, data acquisition was taught in a dedicated lab using desktop computers equipped with specialized data acquisition adapter cards. These adapter cards permitted interfacing analog and digital signal sources to the desktop computer. Beginning with the Fall 1999 semester, Northern Michigan University introduced a program in which all incoming freshmen were equipped with laptop computers. By 2002, students in upper division engineering technology classes all had laptop computers. The laptops in use were Intel Pentium-class machines running either Microsoft Windows Me or Microsoft Windows XP as the operating system. All laptops had the Microsoft Office suite and Microsoft Visual Basic 6.0 installed on them.

One goal was to utilize these computers to improve the student learning experience in a data acquisition course without significantly increasing the cost of course delivery. Another goal was

to involve the students in the "hands-on" development of a data acquisition system. A third goal was to minimize per seat cost due to budget constraints. Table 1 lists several options that were considered to satisfy these goals. A common technique is to use commercial data acquisition cards in a PCMCIA format coupled with dedicated data acquisition software. This approach is ideal for those wishing to gather and analyze data without worrying about the implementation details. These were table options 1 and 2 considered from different vendors. The main problem with these two options was that the cost per seat exceeded \$500. Options 3 and 4 involved using commercial PCMCIA data acquisition cards but have students develop the software. Although this reduced the per seat cost, it still exceeded \$500. Option 5 was to use a USB-based data acquisition card and have students develop the application software. As shown in Table 1, the cost per seat for this was approximately \$270. Option 6 was to utilize the laptop parallel port and have students develop the software. This technique had the lowest cost per seat and required the most student participation in developing the system. Due to its low cost and high student involvement, this was the approach chosen to implement in the course.

Table 1: Comparison of available data acquisition equipment and costs					
Option	Interface	Vendor	Hardware Adaptor	Software Application	Cost
1	PCMCIA slot	National Instruments	6024E	LabVIEW 7	\$ 1,690
2	PCMCIA slot	Measurement Computing	DAS16/12AO	SoftWIRE	\$ 598
3	PCMCIA slot	National Instruments	6024E	Drivers Only	\$ 695
4	PCMCIA slot	Measurement Computing	DAS16/12AO	Drivers Only	\$ 549
5	USB port	B & B Electronics	UD128A8D	BB-DAC	\$ 270
6	Parallel port	None	None	None	\$ 25
Source: www.ni.com , www.measurementcomputing.com , www.bb-elec.com					

Hardware

The laptop PC parallel port consists of 25 signal lines housed in a DB-25 shell connector¹. Eight of these lines are the signal ground return, 12 lines are for digital data output to a printer, and five lines are for digital input from the printer to the laptop. The details and functions of these signal lines are described in many references on the parallel port². A screw terminal adapter board was built by the students which had a parallel printer cable at one end to connect to the laptop parallel printer port. The other end of the printer cable was terminated in a block of 16 screw terminals, one for each signal line. Students used the screw terminals to interface their sensors and circuits to the laptop. Three of the digital output and six of the ground lines were not used in this implementation



Figure 1. Parallel Port Terminal Adaptor

For 8-bit digital output, no additional hardware was required as this is the native mode of printer port in which data is being sent to the printer for printing. For 8-bit digital input, the data must be multiplexed four bits at a time since there are four digital input lines available in the printer port. One parallel port output line is used to control the multiplexer. An ADC0804 8-bit analog to digital converter (A/D) was used to provide analog input capability for input signals in the range of 0-5 volts³. In order to generate analog output signals in a 0-5 volt range, a DAC0832 8-bit digital to analog converter (D/A) was used⁴. Students developed the circuits to provide this digital and analog functionality on a solderless protoboard. The complete system containing laptop, screw terminal adapter, and protoboard is shown in Figure 2.

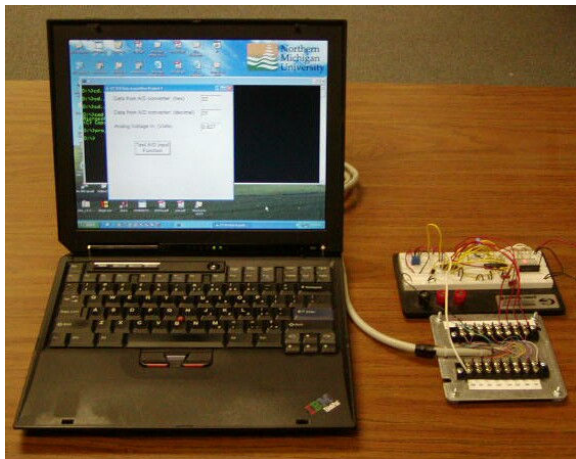


Figure 2. Complete Data Acquisition Setup

Software

The language chosen to interface to the data acquisition hardware was Microsoft Visual Basic. This language was chosen since all students in this data acquisition class had previously taken a one semester Visual Basic programming course. The principal advantage of using Visual Basic is the ease with which the user interface and graphics could be displayed on the screen. Overall application development time is also fairly short with Visual Basic so students could quickly see the results of their efforts. A serious limitation of Visual Basic is the lack of low-level I/O

functions to access devices. Typically this limitation is overcome by making calls to a DLL written in a language that can perform low-level I/O access such as C or C++. There are a number of proprietary and shareware DLLs available for this purpose. The DLL chosen for this project was freely available shareware that provided general PC port access under both Windows Me and Windows XP⁵. This latter requirement was necessary since students could have either operating system on their laptop.

Applications

The students developed a variety of applications. These applications increased in complexity as the students became more familiar with the system and enhanced it. At each stage, students would first complete an exercise to verify basic functionality of their system before proceeding to an application that would include sensor interfacing or signal conditioning. The applications and the data acquisition functions used are listed in Table 2. The first set of three exercises involved developing a functional digital input/output system. The next three exercises added the capability of analog signal acquisition. The last two exercises developed the ability to generate an analog signal from a digital input value. Throughout the exercises, data acquisition, signal conditioning, and control system concepts were discussed and applied where appropriate.

The first application involved using digital output to flash one or more LEDs connected to the parallel port as specified by the user. The purpose of this exercise was to enable the students to become familiar with using the hardware and software together to provide low-level access to the parallel port. The next project involved using the Visual Basic timer and a solid-state relay connected to the parallel port to control a table lamp. Through the graphical user interface, the user could specify the times and duration that the user wished to have the lamp turned on. This project demonstrated the basic characteristics of an open-loop control system. The next project was to add the capability of digital input to the data acquisition system. In order to support 8-bit digital input data using four digital input lines, students had to multiplex the data. An eight-position DIP switch was used to provide digital input that was processed by the Visual Basic application to give the binary, decimal, and hexadecimal representations of the number on-screen.

The ability to process analog input signals was added next. The ADC0804 8-bit A/D converter was used in free-running mode. This converter is easily interfaced to a digital system and accepts analog input signals in the range 0-5 V. The students tested the basic functionality of their A/D system in Project 4 by monitoring the applied input voltage using a DVM and comparing that with the value calculated in a VB application which they created. A solid-state temperature sensor was added to the A/D converter and the students created a data logging thermometer application. The thermometer provided a display of instantaneous and average temperature in both Fahrenheit and Celsius. The Visual Basic application also time-stamped the data and stored it to a log file. The students then were able to use a spreadsheet application, Microsoft Excel, to graph and analyze the data. The next project involved creating a closed loop control system to maintain a uniform temperature in a process. The temperature of a small room was monitored using the temperature sensing system from the previous project. A 1500W electric heater was controlled by a solid-state relay connected to the parallel port. The students were able to adjust

and monitor their system through the Visual Basic application that they created. They were also able to see how hysteresis affected the behavior of their system.

A DAC0832 D/A converter was added to the system to provide an analog voltage in the range 0-5V proportional to an 8-bit digital value. The students built and verified the functionality of their D/A converter system in Project 7. In Project 8, the students built an I-V curve tracer for testing the forward conduction characteristics of diodes. The D/A converter was used to generate a voltage that was used to forward bias a diode. The current being conducted through the diode was sampled using the A/D converter. The students created a Visual Basic application that controlled the operation of the curve tracer and provided a graphical display of the data in real time.

This selection of applications provided technology students with a basic and an applied exposure to digital input, digital output, analog input, and analog output. Students also learned how to integrate hardware and software development, and to use other applications to analyze collected data. The cost of components to implement these applications was less than \$25.

Table 2: Data Acquisition Applications						
Project	Description	Digital Input	Digital Output	Analog Input	Analog Output	Type
1	LED flasher		X			Fundamental
2	Appliance Timer		X			Applied
3	Number Converter	X	X			Fundamental
4	A/D range	X	X	X		Fundamental
5	Fahrenheit/Celsius Thermometer	X	X	X		Applied
6	Greenhouse heater control	X	X	X		Applied
7	D/A range	X	X		X	Fundamental
8	Diode I-V curve tracer	X	X	X	X	Applied

Conclusion

The goal of this project was to leverage the software and hardware available on the student's laptop computers to enhance their understanding of data acquisition skills and techniques at a minimal cost. This technique of teaching data acquisition skills by interfacing a laptop PC parallel port to various signal sources was successful on several levels. Students received exposure to common forms of data acquisition hardware, signal conditioning, and control systems. The basic suite of software applications delivered with the laptops was sufficient for the tasks at hand. The student response to this technique was quite favorable. Feedback from student evaluations of the course showed that the opportunity to create their own data acquisition hardware and software applications was appreciated by the students. In addition, the interfacing skills used encouraged students to review their prior coursework and integrate that knowledge. One unexpected benefit of this approach was that students took their data acquisition systems out of the electronics lab and into other environments where there was data to collect or devices to control. The portability of the laptop encouraged this. The cost of this technique was also quite

low, less than \$25 per seat. This low cost means that it is quite feasible for students to buy or build their own systems that they can use for experiments in other classes.

The main limitation of this technique is the fragility of the student's data acquisition circuits since they were simply wired on a solderless protoboard. A more permanent solution would be for the students to construct a printed circuit board version of their system. Two recent developments that may also be helpful are the availability of low cost USB and parallel port data acquisition modules suitable for laptop interfacing^{6,7}. Finally, the applications developed in this course were targeted towards electrical engineering technology students. These applications could be adapted to be meaningful for students in other technology disciplines.

Bibliography

1. Description of parallel port connector pins, www.doc.ic.ac.uk/~ih/doc/par/.
2. Description of parallel port signals, www.beyondlogic.org/spp/parallel.htm.
3. Datasheet for ADC0804, www.national.com/pf/AD/ADC0804.html.
4. Datasheet for DAC0832, www.national.com/pf/DA/DAC0832.html.
5. Software driver documentation, www.driverlinux.com/Download/DIPortIO.htm.
6. Product description of USB-based data acquisition module, www.labjack.com/labjack_u12.html.
7. Product description of parallel port module, www.circuitspecialists.com/prod.html/icOid/279.

JOHN A. GUMAER

John A. Gumaer is an assistant professor of Engineering Technology at Northern Michigan University. He earned a BSEE from the University of Texas at San Antonio (1986) and a MSEE from the University of Texas at Austin (1990). He is a registered professional engineer in Michigan and has worked in hardware and software development.