AC 2011-2059: ENGAGING COMPUTER SCIENCE STUDENTS IN ELECTRO-MECHANICAL ENGINEERING PROJECTS

Masoud Naghedolfeizi, Fort Valley State University

Dr. Masoud Naghedolfeizi is a professor in the Deaprtment of Mathematics and Computer Science at Fort Valley State University. His research interests include computer-based measurement and instrumentation systems, applied artificial networks, and engineering education.

Prof. Sanjeev Arora, Fort Valley State College
Prof. Ramana Gosukonda, Fort Valley State University

Dr. Gosukonda is an Associate Professor of Computer Science at Fort Valley State University. His research interests include Bioinformatics, Applications of Artificial Neural Networks in Biotechnology, and Pedagogical research in computer science curriculum for undergraduate students.

Mr. Nabil A Yousif, Fort Valley State University

Mr. Yousif is an assistant professor in the Mathematic and Computer Science Depadtment, at Fort Valley State University. His interest includes System Business Application Software design and database design.

Engaging Computer Science Students in Electro-Mechanical Engineering Projects

Abstract

This paper describes two engineering projects that were implemented in a capstone computer science course. The projects were designed to engage students in cross-disciplinary activities and to enhance their career opportunities in the job market. The first project was concerned with the design of a data acquisition software system and the second project involved a standard data acquisition system for condition monitoring of computing equipment. The students who worked on these projects were involved in hands-on activities and gained knowledge and skills that were cross-disciplinary in nature.

Introduction

In recent years, there has been an increasing trend in industry to recruit college graduates with interdisciplinary as well as cross-disciplinary skills[1]. Many educators have also emphasized that it is essential to engage students in activities across the curriculum to enhance their problem-solving skills and broaden their perspective and understanding of real-life problems[1-5]. This could be attributed to the fact that many significant real-world problems are interlinked in nature, involving for instance mathematics, computer science, and engineering. As a result, many colleges and universities have offered various courses or activities across the curriculum[5, 6, & 8]. For instance, in the field of computer science bio-informatics has currently become very popular and courses or minor programs in this field are offered by many computer science departments[5]. However, it should be also recognized that the standard and sometimes rigid curricula of many majors including computer science do not leave enough room for interdisciplinary/cross-disciplinary courses that could be important for the future careers of the graduates. Further, the lack of facilities such as appropriate laboratories and trained faculty could markedly hamper the inclusion of well-structured cross-disciplinary topics in the curriculum. Nevertheless, efforts should be made to somehow incorporate such activities in the program of study [5, 7, &8]. For example, students could be engaged in cross-disciplinary projects or case studies at the junior and senior levels.

The survey of pertinent literature suggests an increased need to equip computing professionals with interdisciplinary/cross-disciplinary skills and some solutions have been offered to the problem. Several approaches involving requiring interdisciplinary projects in capstone or upper level division courses with traditionally closely related fields such as electrical and computer engineering have been implemented in many computer science programs and discussed in several scholarly publications[9-11]. Additionally, the computer science major has several common courses with electrical and computer engineering majors such as digital fundamentals, computer

networks, artificial intelligence, and computer organization. These have helped alleviate the problem to some extent.

However, little attention has been paid to engaging computer science students with *other* engineering fields such as mechanical and measurement/ instrumentation. Many problems associated with these engineering fields require computer solutions that often demand computing professionals with some kind of relevant engineering knowledge and skills. For instance, computer science professionals working in the field of robotics could greatly benefit from having additional skill-sets in mechanical and sensory devices that are extensively used in this area. This is also the case for computer professionals working in other engineering fields that need computer programs/databases/intelligent systems to optimize and/or control the operation of various industrial plant components. Additionally, this knowledge and skill sets could help reduce the technical communication gap between computing and engineering professionals in the industrial world.

As a first step in this direction, the computer science/physics/engineering faculty at our institution in spring 2009 decided to conduct a pilot test of this new approach to cross-disciplinary activities by incorporating electro-mechanical engineering-based projects in the capstone course, Senior Seminar.

Senior Seminar Projects

The projects in Senior Seminar course are traditionally related to standard business environment and research in emerging technologies. Prime examples include database system development and research, artificial intelligence, computer networks, computer games, and information security. Students in this course are required to complete two types of projects. The first kind (Type I) is a moderately large scale project and the student normally needs the whole semester to complete it. At the end of the project, the student submits a detailed report on the project and makes an oral presentation before the class and invited faculty. The second (Type II) is a relatively small scale project where the student needs to finish it within one and half month. At the conclusion of the project, the student makes a poster presentation for faculty judges and other participants during previously announced dates in the Auditorium of College of Arts and Sciences.

In 2009, for the first time, it was decided to add engineering-based projects to the list of Senior Seminar projects. The purpose of adding these projects was primarily to further enhance career opportunities for students since current trends in technical job market indicate that many jobs require the knowledge of both computer science and engineering.

With the aid of faculty with engineering and physics background, two projects with real-world applications and hands-on activities were designed. One project from Type I category and the other for Type II. The projects were briefly discussed in the class and the interested students were asked to see the project supervisors for further discussions. Five students showed interest and the project were discussed with them in details. After reviewing the student technical background in physics, mathematics, and engineering fields, two students were selected. The student who was selected for Type I project had an associate degree in electronics engineering,

excellent programming skills, and a very keen interest. He also had a solid background in physics. The other student who was selected to work on the Type II project had a very good mathematical background and appetite for hands-on activities.

Type I project: Design of a Data Acquisition System

In this project, the student was required to design a data acquisition system for condition monitoring of industrial machineries. The data acquisition system recorded both time and frequency domains of three-axis vibration signatures of rotating machineries. It also recorded three temperature signatures. The project was divided into two parts and the student completed the first part before proceeding to the second. The entire project was developed under LabVIEW programming environment[12 &13], which is the industrial standard software and is also extensively used in many scientific organizations. Similar to Visual Basic (VB), LabVIEW has two panels: front panel and Wiring Diagram corresponding to Form Window and Procedure Window in VB, respectively. However, unlike the VB there is no scripting in LabVIEW. Every programming function/ procedure (e.g.; arithmetic operations, looping, arrays, and conditional and logical statements) is in form of a graphical object.

To familiarize the student with LabVIEW, the project's supervisor provided the student with a brief introduction to LabVIEW programming and demonstrated how to develop a simple program under LabVIEW. Then, the student was given pertinent literature and tutorials to gain further knowledge and skills in LabVIEW programming. To complete the project, the student carried out three main tasks: Design of Front Panel, Design of Wiring Diagram, as well as Testing and Debugging.

1. Design of Front Panel: This phase was relatively simple and the student needed to identify the appropriate controls and indicators in the front panel that provided the graphical user interface. The controls were used to configure data acquisition parameters such as sampling rate, filter selection, and data acquisition card settings. In addition, they provided command buttons and text boxes for recording and displaying data from one or more vibration axis and temperature sensors. The indicators were used to show graphically/numerically the acquired data in time domain and/or frequency domain.

   It should be noted that prior to designing the front panel, student gained a good understanding of the operation of data acquisition system and user specifications for data recording and displaying. Thus, the student went through the full programming life cycle system including interviewing the user.

2. Design of Wiring Diagram: In this phase student implemented a LabVIEW program for the objects designed in the front panel. This phase was relatively complex and the student had to implement the program in form of graphical objects. The data flow from one object to another was in form of wires connecting the two objects. In addition, the student needed to utilize LabVIEW data acquisition functions in the program to configure the data acquisition board and acquire sampled data in the time domain from selected sensors at a specified sampling rate set by the user in the front panel. The sampled data along with its power spectrum in frequency domain were displayed on the front panel.

The power spectrum analysis was only used for vibration measurements. The program also allowed the user to record time domain measurements with time stamped and user messages (entered in the front panel) in spreadsheet format.

3. Testing and Debugging: In this phase, the student ran the data acquisition software to collect data from all sensors and performed debugging as needed.

Type 2 project: Condition Monitoring of Computing Devices

In this project, the student used a portable data acquisition/logger system to collect data to monitor the health of typical computing devices such as computers, printers, servers, and copiers. Standard data loggers are capable of collecting time domain vibration and temperature data using appropriate sensors. The data logger used in this project was designed for academic purposes and it was capable of recording and displaying time domain data. After training the student with the data logger, he was directed to collect vibration data from several computers, printers, and servers in computer labs of the mathematics and computer science department. The data from the computers were collected during saving large data files in order to pick up vibration signature from the computer hard disk, cooling fan, and CPU fan. For the printers, the data were collected during a print job. The sampling rate and the duration of data collection were set in data logger before the start of each data collection.

The student was provided with background information about frequency domain analysis and its applications to condition monitoring of rotating equipment. The student was then guided to study discrete and fast Fourier transforms in order to obtain frequency domain signatures of the collected data using spreadsheet software system.

Learning Objectives

1. Student should be able to apply programming concepts learned in computer science programming classes (JAVA in our department) to a new programming environment (LabVIEW)
2. Student should be able to design a GUI-based data acquisition software system that follows the programming life cycle.
3. Student will gain cross-disciplinary knowledge and skills that could be useful for his/her future career
4. Students will gain hands-on experience in using industry-type data acquisition systems and learn data collection, processing, and analysis techniques to measure the health of computing equipment.

Objectives 1-3 are directly related to the Project I and objective 3 and 4 are associated with Project II.

Project Implementation and Results

LabVIEW 6.1 was used to implement Project I. Figure 1 displays the Input/Output window, known as front panel in LabVIEW. It includes two tabs: waveform and configuration settings. In Waveform tab, the user can select up to three vibration sensors by clicking on push-button switches (labeled X-Axis, Y-Axis, and Z-Axis) under Input Control section. The sampling rate

and the total number of samples per waveform can be set by typing the required values in the Sample Setting section. The time domain vibration wave forms of these selected sensors are displayed in the Accelerometer Input chart. The power spectra (frequency domain signatures) of these waveforms are also displayed in Vibration Power Spectrum chart. The user can also select the type of filter needed for the power spectrum from a drop-down menu labeled Filter Selections. The filters are used in FFT of power spectrum calculations[14-15]. The user can also collect measurement data from other types of sensors such temperature and voltage by clicking on sensor buttons labeled sensor 1-3. In addition, the user can save the time domain data and provide data log description (in text box shown in the Figure 1) by clicking on the CAPTURE button under Data Logging in Front Panel. In the Configuration Settings tab, the user can assign a data acquisition channel to each measurement sensor described above. The channels are categorized as accelerometers and temperatures.

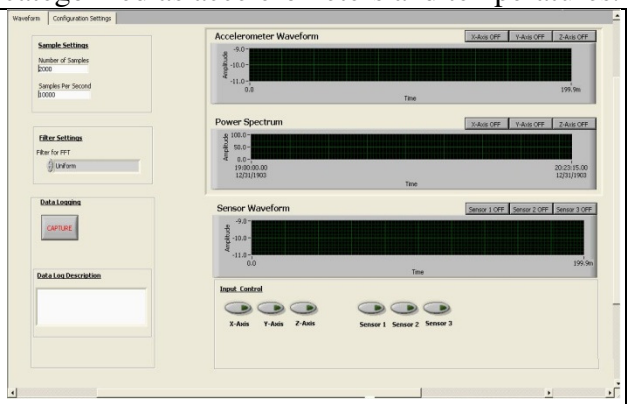

Figure 1: Front panel of the data acquisition system

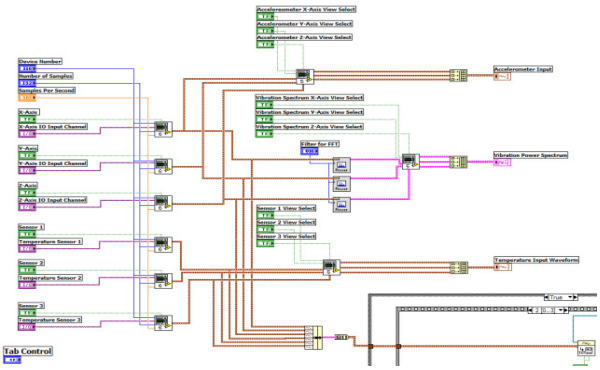

Figure 2: Portion of wiring diagram of the data acquisition software

In the main wiring diagram, the student used numerous LabVIEW programming structures such as loops, selection, subprograms (known as sub VIs in LabVIEW), arrays, and various data types as well as LabVIEW data acquisition functions to build the data acquisition system. Figure 2 displays a portion of the wiring diagram. While developing the diagram, student learned how to logically connect programming structures together. For example, to collect the data, the data acquisition functions have to be placed in a continuous loop structure that can be terminated by the user on the front panel.

Since data is processed as soon as they become available in LabVIEW programming environment, the programming structures may not be processed sequentially as is the case in traditional programming languages. Upon being informed of this feature, the student learned how to process the data sequentially using Sequence Structures in LabVIEW. This approach enhanced student's critical thinking needed in developing a program under a new programming environment that requires the student to anticipate which portion of the program needs sequential processing. Sequence structure was implemented in logging the data.

During the project, the student applied the principles of software life cycle system to develop his program. He worked with the users (two faculty members who would use the software) to understand problem specifications. In the solution design phase, he set up his algorithm and

quickly understood that the complexity of the problem requires the subdividing of the problem into smaller tasks. In the implementation phase, he coded the algorithm in LabVIEW program. Finally, he tested and debugged the program and requested the users to examine the software. After examining the software, the users provided the student with suggestions and comments. For instance, the users needed to record the data with time stamp and description provided through the front panel. They also requested some changes to the charts used to display the data. The student made the necessary modifications to satisfy the users' needs.

In the second project, the student used Vernier LabQuest data acquisition hardware and software systems[16] as well as a three-axis accelerometer to collect vibration data from computing devices. The schematic diagram of data acquisition and the ensuing analysis is shown in Figure 3. The data was collected from a healthy computer (as baseline), a computer with cooling fan problem, a computer with noisy hard disk, and a server. In addition, data was collected from a laser jet printer (Figure 4) and a copy machine. Since data was collected in the time domain, the student had to upload the data to a PC computer equipped with MS Excel software to carry out frequency domain analysis. Although this software has only the basic FFT function for frequency analysis, it was sufficient for this project for two main reasons: the student was very proficient in using Excel and did not have to learn another language; and the data from the Vernier equipment could be easily exported to Excel. Furthermore, a full-range frequency analysis (available in other software systems such as Mathlab) was not required, since this was a small-scale project.

Prior to frequency analysis, we provided the student with information about frequency domain analysis using Fast Fourier Transforms[15]. Concepts such as aliasing, Nyquist frequency, folding frequency, and spectrum resolution were explained and discussed in details. Once the student gained sufficient knowledge, he was able to carry out the frequency domain analysis by utilizing Excel FFT function. Since the rotating speeds of the cooling fans and hard disks were known, the student was able to identify these components by inspecting the frequency domain spectra. By looking at the spectra, it became very clear to the student why it was important to perform frequency domain analysis verses time domain analysis for equipment containing rotating or vibrating components. He noticed from visual inspection of time domain spectra, it is impossible to identify various frequencies containing in a complex waveform in the time domain but these can be easily identified in the frequency domain spectra. He also compared the frequency spectra of a healthy computer with those of computers with bad cooling fan and hard disk. He concluded that changes in the shapes of fundamental frequencies could be indicative of bad cooling fan and hard disk. Sample spectra are shown in Figures 5a and 5b.

Discussions and Concluding Remarks

In this paper, two engineering projects used in our computer science senior seminar (capstone) course were described. Each of these projects dealt with vibration measurements but from different perspective. In the first project the focus was software development to be used for data collection and analysis; whereas, in the second project the purpose was to perform data collection and analysis. Although the students were initially uneasy when they embarked on the projects, they were provided with ample documentation, guidance, personal assistance, and support. Once they gained a certain level of knowledge and skills, they carried out the projects

with a great deal of self-confidence and enthusiasm. After completing the projects, both students understood that they could apply the knowledge and skills gained in computer science classes to

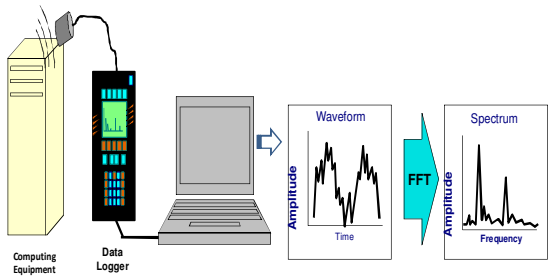

Figure 3: Schematic diagram of data collection and ensuing analysis

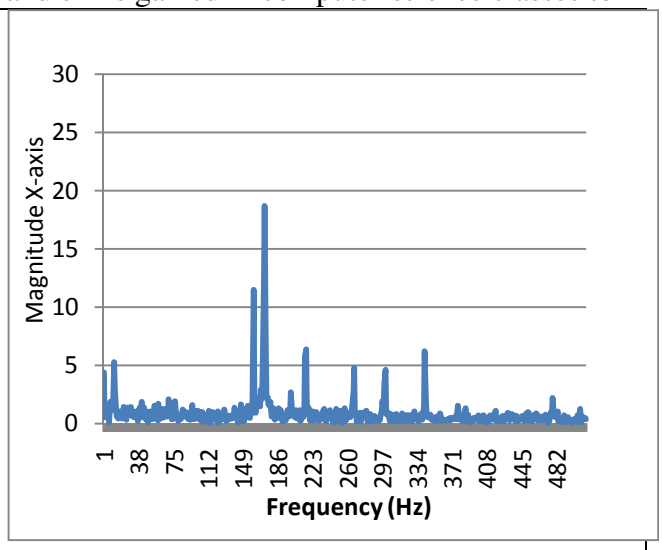

Figure 4: Frequency spectrum of a LaserJet printer while printing

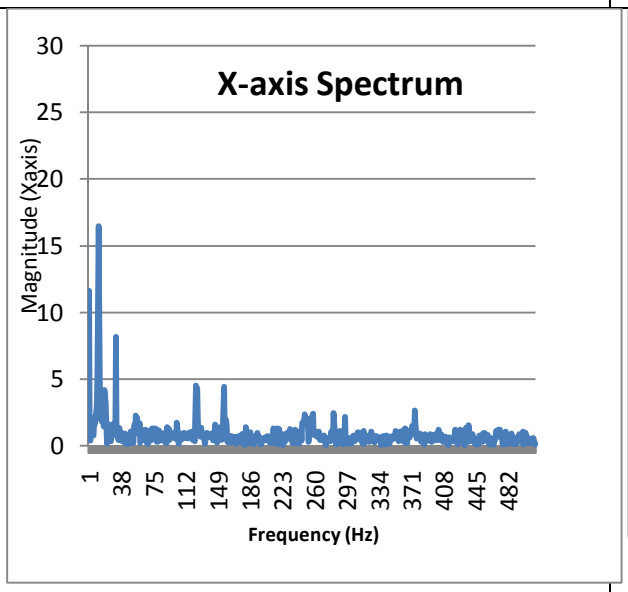

Figure 5a: Frequency spectrum of a healthy computer

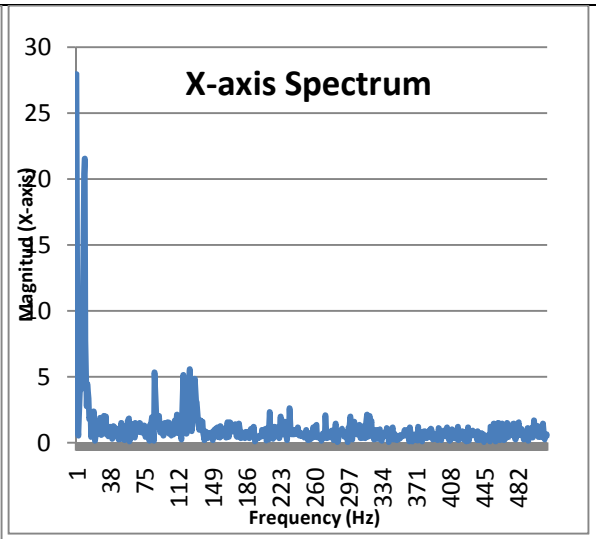

Figure 5b: Frequency spectrum of a computer with a noisy hard disk

engineering applications by learning some basic concepts and methods. They were especially delighted that these projects provided them with additional skills to enhance their career opportunities in the job market.

One might argue a major project in a capstone course should be designed to further increase the knowledge and skills of students in materials they already learned in the classroom rather than to

be used for new concepts and skills. To address this concern, a cross-disciplinary project should be designed with the focus on enhancing the major field knowledge and skills of students while they gain cross-disciplinary knowledge and skills. This was exactly the case in the project one where the student gained cross-disciplinary knowledge and skills and at the same time, enhanced his previous knowledge in software development and life cycle.

References

1 Bjarne Stroustrup, "Viewpoint: What Should We Teach New Software Developers? Why?" Communications of the ACM, vol. 53, No.01, pp: 40-43, 01/2010

2 Bijan Mashaw, "Information technology approach in computer science," Journal of Computing Sciences in Colleges,Volume 24 , Issue 5, pp: 191-197, May 2009

3 Suvineetha Herath, et.al "An cross-disciplinary accounting forensics course to improve computer science enrollments," Journal of Computing Sciences in Colleges, Volume 23 , Issue 6, pp: 74-81, June 2008

4 Jamil Saquer , "A data mining course for computer science and non-computer science students," Journal of Computing Sciences in Colleges, Volume 22, Issue 4, pp: 109 – 114, April 2007

5 Ramana M. Gosukonda, and M. Naghedolfeizi, "Initial evaluation of bioinformatics curriculum at Fort Valley State University," Journal of Computing Sciences in Colleges, Volume 23 , Issue 2, pp: 236 – 242, December 2007

6 Karl-Heinz Lampe, et.al "Research between natural and cultural history information: Benefits and IT-requirements for transdisciplinarity," Journal on Computing and Cultural Heritage (JOCCH), Volume 1, Issue 1, Article No.: 4, June 2008

7 Leen-Kiat Soh, et.al "Renaissance computing: an initiative for promoting student participation in computing," Technical Symposium on Computer Science Education, Proceedings of the 40th ACM technical symposium on Computer science education, Pages: 59-63, March 2009

8 Dorothy Deremer, and Katherine G. Herbert "An cross-disciplinary undergraduate science informatics degree in a liberal arts context," Technical Symposium on Computer Science Education, Proceedings of the 37th SIGCSE technical symposium on Computer science education, pp: 179 – 183, March 2006

9 Rover, D.T., Fisher, P.D., "Cross-functional teaming in a capstone engineering design course," 27th Annual Frontiers in Education Conference, pp:215-219, 199.

10 Roppel, T.A, Hung, J.Y. Wentworth, S.W., Hodel, A.S., "An interdisciplinary laboratory sequence in electrical and computer engineering: curriculum design and assessment results," IEEE Transactions on Education, Issue 2, pp: 143-152, May 2000

11 G. H. Bernstein, J. B. Brockman, G. L. Snider, and B. E. Walvoord, " From Bits to Chips: A Multidisciplinary Curriculum for Microelectronics System Design Eductaion," 2003 International Conference on Microelectronics System Education, 2003

12 LabVIEW 6.1 Manuals, National Instruments, www.ni.com

13 John Essick , "Hands On Introduction to LabVIEW for Scientists and Engineers," Oxford University Press, USA ,November 12, 2008

14 E. C. Ifeachor and B. W. Jervis, "Digital Signal Processing: A Practical Approach," Prentice Hall, New Jersey, 2001

15 R. G. Lyons, "Understanding Digital Signal Processing" Prentice Hall, New Jersey, 2004

16 www.vernier.com