Developing An Interdisciplinary Hardware [1 2] Laboratory With CIM Capabilities

Carl Steidley
Mohammad M. Asoodeh

Department of Computer Science/Department of Industrial Technology
Southeastern Louisiana University
Hammond, LA 70402

INTRODUCTION

It has become widely accepted that the computer is an indispensable tool in the practice of science and engineering, thus, in recent years the science, engineering, and engineering technology education communities have been adopting and adapting the computer as a tool in ever new and innovative ways of teaching and research in science, engineering, and technology. Likewise, the importance of the availability of a variety of computing environments to undergraduate students of science and technology is widely agreed upon. This importance is attested to by the accreditation criteria of the Computer Science Accreditation Board as well as that of Industrial Technology.

The Department of Computer Science and the Department of Industrial Technology at Southeastern Louisiana University (SLU) have initiated a joint project to build computing facilities and curricular programs which will provide outstanding educational opportunities for computer science and industrial technology majors. Among the aims of this collaboration is to create a model computer integrated manufacturing facility, built on existing and recently acquired equipment and facilities. Moreover, this project is a part of an Integrated Undergraduate Technology-Rich Curriculum.

In its publication **Report on the National Science Foundation Disciplinary Workshops on Undergraduate Education [2]** the National Science Foundation had this to say about undergraduate computer science laboratory facilities:

> “The laboratory infrastructure is not in place to support undergraduate computer science as a science. ”

In response to this and other criticisms [1,7], the first author chose to develop a laboratory that would support three computer science courses that emphasize computer hardware, instrumentation, interfacing, computer-controlled systems, and real-time physical systems. It was his intent that in this laboratory students would gain hands-on experience with the fundamental hardware building blocks of computing machinery and the

[1] Partial support for this work was provided by the National Science Foundation Instrument and Laboratory Improvement Program, Contract # DUE-9350651.

[2] Partial support for this project was provided by the Louisiana Educational Quality Support Fund, Contract# LEQSF(94-95)-ENH-UG-23.

integration of these building blocks with software to build computing systems. This laboratory would also provide students with facilities to experiment, develop, and run real-time software support for a physical system.

As the project began to develop and space in which to locate the laboratory became difficult to find, the second author joined the faculty of Industrial Technology at Southeastern Louisiana University. The second author was given a laboratory, but the equipment in the laboratory was outdated. The two authors found that they had many mutual interests and thus the project described in this paper was born.

Laboratory Development Plans

Initially, in seeking funding for this project it was planned that this laboratory would support three computer science courses which address both specific guideline subject matter as well as supporting the philosophy of the **Role of the Laboratory** as recommended by **Computing Curricula 1991** a Report of the ACM/IEEE-CS Curriculum Task Force [7]. These courses are: Computer Architecture, Microprocessor Systems, and Robotic Software Systems.

It was not our aim to train electronic engineers or technicians, but rather to impart hardware principles to future computer scientists/software engineers. [5] Our challenge and need is to teach these students how to think and solve problems in terms of digital logic as it functions within and with computer systems. We were not anxious to teach basic electronics nor to have students have to worry about correct power supply wiring, accidentally wiring output-to-output, input-to-input, or otherwise causing electrical short circuits that may damage valuable laboratory equipment or even damage inexpensive integrated circuits which take valuable laboratory time to trouble shoot and replace.

Interdisciplinary Work

When the second author joined the Industrial Technology faculty at Southeastern in early 1995, he discovered that the laboratory in which he was to conduct several of his courses was equipped with mostly outdated equipment much of which was in a state of ill-repair. At this time the first author found that, due to construction and renovation, he was to lose the space in which he had intended to establish his laboratory. The authors began to see ways in which a laboratory could be equipped to support courses that could satisfy the needs of both and possibly improve the education of their students. Three computer science and three industrial technology courses that were to be immediately supported by the lab are: Computer Architecture, Microprocessor Systems, Robotic Software Systems, Industrial Robotics, Computer-Aided Manufacturing, and Materials Science.

Computer Architecture

To support the Computer Architecture course we wanted a rugged, “student proofed” digital logic system which contained all of the components essential for a complete introduction to the digital building blocks of computer systems. That is, the intended sequence of laboratory exercises and experiments would lead the student through a series of basic combinational logic exercises in which the student would become familiar with discrete AND, OR, NAND, NOR, and Inverter logic. This would be followed by exercises in which flip-flops, the half, full and parallel adders would be constructed from discrete logic gates. This series of projects would be followed by exercises in which counter, shift register, and encoders/decoders are constructed from discrete sequential and combinational components. [5]

Microprocessor Systems

To support the Microprocessor Systems course we wanted to provide students with a series of interfacing and control exercises and experiments in which we could build on the fundamental hardware principles established in the Computer Architecture course. Further, we wanted these experiences to begin to approach “real world" conditions, that is, we wanted to provide students with access to the “inner workings” of a commercially available microcomputing system. However, we were constrained by our clientele in that these are “liberal arts” software engineering/computer science majors who have experience writing programs, have solid mathematics backgrounds, but have little or no electronics background save the Computer Architecture course mentioned above. [3,5]

Robotic Software Systems

During the original planning stages for this laboratory we envisioned using the lab to support a Real-Time Software Systems course. While we may use this lab for this purpose, we have been successful in finding external support for a laboratory to support only that course [6] thus, we will use this lab to support Robotic Software Systems in addition to the architecture, microprocessor, robotics, CAM, and materials science courses.

Our plans for support of the Robotic Software Systems course involved the acquisition of “smart” robotic systems, that is, we anticipated acquiring robots that have on-board microprocessor systems and the ability to be controlled from a teaching pendant. These characteristics would provide for quick “start-up” time for students to become familiar with the system by learning to guide, control, and store trajectory programs for the robot system’s manipulator arm, thus, learning the fundamentals of this new type of 1/0 system prior to delving into instrumentation, interfacing, and programming projects for the system. These characteristics also provide an opportunity for teaching students the fundamentals of stepper motors and digital control of such motors in a physical system.

We were concerned, during the planning stages for this laboratory, that the interfacing of the control computer and the robot system work cell could be very difficult for our “non-engineering” majors.

We were also concerned that the laboratory projects not be "cookbookish" while providing solid hardware principles, thus, it was imperative to find just the right equipment. **[3,4]**

Computer-Aided Manufacturing

This course was originally designed to teach students to use the computer to control the operation of a variety of manufacturing equipment; such as, lathes, drilling, and milling machines. However, the course has been expanded to cover concepts and procedures associated with numerical control (NC) and computer numerical control (CNC) technologies used in today’s manufacturing industries. In this course, the students learn about automated system processes in which machine tools follow an ordered sequence of operations under the control of a set of coded instructions. This would require theoretical and practical knowledge of automated systems.

The laboratory equipment to support this course consisted of three Dyna Mechanics Dyna Myte 3000 CNC lathes and three 486 control microcomputers for these machines. However, little had been done, prior to this project, to integrate these systems in any but the most rudimentary fashion, that is, as simple control computers.

Our plans for support for this course included the integration of the student's prerequisite computer-aided design experience, that is, students will utilize CAD drawings from AutoCAD converted to a DXF file and imported into a CAM program, such as MasterCAM, to be used for numerical control of the manufacture of the designed piece.

Industrial Robotics

This course was originally designed to present the electronic and mechanical construction of representative types of industrial, personal and educational robots. The course was also designed to cover the industrial and engineering applications as well as the economic and human factors of robots in manufacturing systems.

Again, our plans for support of this course involved the acquisition of "smart" robotic systems, that is, we anticipated acquiring robots that have on-board microprocessor systems and the ability to be controlled from a teaching pendant. These characteristics would provide for quick "start-up" time for students to become familiar with the system by learning to guide, control, and store trajectory programs for the robot system's manipulator arm, thus, learning the fundamentals of this new type of 1/0 system.

Materials Science

This course was originally designed to be a study of the principal materials used in industry, as well as the structure, properties, treatment, selection criteria, and common methods of testing these materials. The course has grown to cover wider areas such as finite element analysis (FEA), finite element modeling (FEM), strain gage testing, and photo elastic analysis. These new areas require intensive use of computers and communications networks for the gathering and analysis of required information.

Our plans for the laboratory to support this course included the use of automated instrumentation that would allow us to set up information about strain gauge instrumentation to observe material properties, gauge sensitivities, etc. Using these parameters the system could automatically output the results of testing in predetermined engineering units. This information could also be retained for off-line display or for export to databases, wordprocessors, and spreadsheets.

EQUIPMENT ACQUISITIONS

Computer Architecture

We chose CES Industries Ed Lab 702 Digital Trainer to support the Computer Architecture course in our laboratory. This is a rugged digital logic development system containing all the discrete components essential for a complete introduction to the digital hardware of computer systems. This system meets this requirement without requiring students to understand even the basics of power supplies or other electronics. That is, students will be able to construct digital logic circuits almost as easily as is done in the lecture with pencil and paper and yet a physical system which will perform a physical function will be realized. This system includes: AND, OR, NAND, NOR, exclusive OR, NOT/Inverter combinational circuits; RS, JK, D, and T sequential circuits; input switches with complementary outputs and momentary switches for logic level input; a clock circuit; and individual display lamps and hexadecimal display for output. The inputs and outputs of each component are brought out to a heavy aluminum panel via banana jacks, thus allowing for rapid symbolic assembly of fictional circuitry.

Using this development system in the laboratory, the student will reinforce and learn the concepts, applications, and operations of discrete logic circuits. Most importantly however, the student will develop the knowledge of how these devices interact with each other and how they are put together to form fundamental building block integrated circuitry.

Microprocessor Systems

We chose the CES Industries PC Interfacing System to support the Microprocessor Systems course in the laboratory. This system allows the user to bring the data, address, and control buses of an attached PC out to a panel via ribbon cable and banana jacks. The system is ruggedly constructed and fully protects the associated control computer. This protection is provided by buffering all connections to the control computer. This system is filly compatible with the digital development system used to support the architecture course.

We anticipate a series of exercises in which students will learn to build address decoders, scan and input external data from sensors and instrumentation, provide latching and control and then provide scanned output after processing the input data.

Materials Science

To support the laboratory for this course we chose National Instrument's Data Acquisition Class Pack. This package includes: A full development system of LabVIEW, a termination breadboard with silk screened component locations, a DAQ interface, and an 1/0 DAQ unit for communication through the parallel port.

LabVIEW, for those unfamiliar with this package, is a graphical programming system for data acquisition and control, data analysis, and data presentation. This package provides an alternative to cumbersome text-based programming by providing an innovative programming methodology in which the user graphically assembles software modules called VIS (virtual instruments).

Robotic Software Systems, Industrial Robotics, and Computer-Aided Manufacturing

For laboratory support of the Robotic Software Systems, Industrial Robotics, and Computer-Aided Manufacturing courses we chose several systems manufactured by Eshed Robotec Inc. The First of these systems is the Scorbot ER III Point-to-Point Robot System. The Scorbot ER III consists of a highly reliable mechanical arm and multi-function controller. The ER III manipulator arm is an open, vertically articulated structure. It has 6 D. C. servo motors with closed loop control. The manipulator arm has five axes plus gripper with optical encoders on all axes. It has a load capacity of 1 kilogram and a repeatability of 0.5 mm and 10 different speeds with a maximum speed of 330 mm or 13 inches per second. The gripper is fitted with sensors to allow for measurement of objects as well as touch control. The controller provides for simultaneous control of 8 axes (7 plus gripper). The Scorbot can be directly controlled by a hand-held teach pendant. The controller provides 8 inputs as well as 8 outputs thus providing easy integration with the system's environment.

The software for the robotic system includes a high-level, menu-driven robot programming language (Scorbase), an animated, graphic-based simulation language, and utilities for programming in assembly as well as high-level procedural languages.

The second Eshed Robotec system to be used to support these courses is the Scorbot ER V Continuous Path robot system. This is a fast, accurate, flexible and reliable robotic system designed for laboratory applications, research, and/or training. This manipulator arm also has an open structure and is vertically articulated with five degrees of freedom (plus gripper). A variety of grippers allows for a wide range of flexibility. The ER V controller is a stand-alone, real-time, multitasking controller which allows for simultaneous and independent operation of several programs, grouping of axes for multiple device control and editing programs while others are running. The basic controller is configured with 8 axes of control and is expandable to 11 axes and is capable of handling both 16 inputs and outputs.

In addition to a graphics-animation simulation package like that described above for the ER III, software for the ER V includes an internal language for the controller called the Advanced Control Language (ACL) and a five-level robot programming language called SCOREBASE which is a menu driven high-level language. SCOREBASE Level 5 provides tools for immediate integration of the robot into an automated cell with cartesian, linear, and circular robot control provided.

A third Eshed Robotec system that will be used to support these courses is the Eshed ROBOTVISIONplus system. This machine vision system includes an updated vision software package, with a real-time frame grabber (512 x 512 pixels, 256 gray levels) and multiframe/multicamera capabilities. Multiple vision parameters such as camera gain/offset, threshold, etc. can be set for each application. Explicit shape analysis and measurements can be conducted.

To outfit our CIM cell, we purchased two Eshed Robotec ER VII robotic systems. The ER VII is vertically articulated, 5 axes robot with D. C. servo-motor actuators, harmonic drive transmission, utilizing optical encoder feedback. The manipulator has a maximum payload of 2.2 kg, maximum composite speed of 1000 mrn/see, and a repeatability of 0.2 mm.

The ER VII control unit is a stand-alone, real-time, multi-tasking controller based on the Motorola 68020, which allows simultaneous and independent operation of several programs, grouping of axes for multiple device control and program editing while other are running. The controller is configured with 8 axes of control, 128Kb of battery backed-up user RAM, 16 inputs and 16 outputs. The controller's 250 parameters include: proportional, differential and integral constants for each of the axes; thermic, impact and limit protections; tool center point offset; kinematic calculation factors; and velocity profile and speed limits.

The continuous path control, exceptional working volume and absolute speed/time between control points will enable us to design spraying or dispensing exercises and experiments in which students can expect that the robots will perform in a very efficient and accurate manner. Due to the high repeatability factor of these systems we will be able to design sorting, palletizing, loading/ unloading CNC machines or other material handling application in which we can expect high quality results. Since these robotic systems have both open or closed loop control with various velocity profiles (paraboloid, trapezoid, free running, acceleration and deceleration), proportional-integral-differential control the difference between the fine tuned control and insufficient gain, overshoots, position and error speed can all be demonstrated.

Salvageable Equipment

Equipment that was deemed salvageable and useful to this project included: three Dyna Mechanics Dyna Myte 3000 CNC lathes and a Dyna Mechanics Dyna Myte 2800CNC milling machine.

The Dyna Myte 3000 is an industrial grade, bench-top CNC lathe designed for small, precision parts manufacturing. Its precision turning specifications include a position accuracy of +/-0.0006", a repeatability of +/-0.000 15", and both a roundness and cylindricity of +/-0.0002". It has an RS-232 interface for use with CAD/CAM software as well as an external synchronization interface.

The Dyna Myte 2800 is an industrial grade bed-type precision CNC milling, drilling, and boring machine. The 2800's performance specifications include 60"/minute traverse rate, 0.1 "/minute to 30"/minute cutting feed rate, position accuracy of 0.0005", and a position repeatability of +/-0,0001", It has an RS-232 interface for download or upload execution.

In order to begin the design of our computer integrated manufacturing cell, we purchased several pieces of hardware and software packages including: a pneumatic centering vise for use with the lathe; pneumatic, CNC automatic door opening systems for both the lathe and the mill,; CNC integration packages for both the lathe and the mill; and a 3D machining bundle for use with both the lathe and the mill.

INTERDISCIPLINARY INSTRUCTION

Currently, our plans are to involve students of industrial technology and computer science in team projects in the laboratory. Moreover, we plan that computer science students will teach industrial technology students the fundamentals of the SCOREBASE programming language, as well as the fundamentals of computer simulation of actual robot movement via the graphical simulation package. Likewise, we plan that industrial technology students will teach computer science students the fundamentals of computer aided design as well as the fundamentals of industrial CNC machinery and their uses in manufacturing processes.

We will focus on the creation of an integrated, interdisciplinary, design- oriented, lower-division curricula that emphasize broad concepts, student discovery, cooperative learning, problem-solving processes, and design. We believe that this will drive associated upper-division restructuring. We seek to develop a sense of community during the undergraduate experience, with attention focused on the student/student, student/faculty, and faculty/faculty interfaces. With a interdepartmental commitment to cooperative learning, student/student interaction and student/faculty interaction will change dramatically. Responsibility for learning shifts to students organized into teams. As a coach instead of a lecturer, the faculty member emphasizes positive interdependence among team members and individual accountability. We will increase utilization of new technology-enabled methodologies. High technology classrooms and multimedia laboratories will be implemented throughout the curricula as appropriate.

Again, we plan to utilize the ER III and ER V robotic systems for training purposes, while the ER VII systems will be used in the CIM Cell.

Conclusions

Since the advent of the desktop computer and/or standalone workstation the growth of utilization, in new and innovative ways, of this powerful tool in science and engineering has been phenomenal. The authors believe that we are only scratching the surface. Further, given the increasing percentage of students in engineering and computing sciences seeking graduate degrees, the authors believe that it is increasingly important to introduce students to applied research in the undergraduate curriculum. Likewise, we believe that interdisciplinary training in which students are taught the fundamentals of good practice and policy in computing will enhance and improve the use of this tool to advance science and engineering, thus, students need to be able to use sophisticated, real-world equipment. Students also need to have real-world-like experiences. The work described in this paper is an effort to afford our students the fruit of these beliefs.

The laboratory being equipped and developed, as described in this paper, is meant to provide software engineering/computer science majors with important hardware, instrumentation, and interfacing knowledge that many such majors do not gain during their undergraduate education. Likewise, this laboratory is meant to provide industrial technology students with a modern, automated laboratory in which they can work with future programmer/analysts to learn more about software engineering.

It is hoped that the experience that students will gain from this laboratory will prepare them for becoming members of systems development teams consisting of both hardware and software engineers. Each software engineering team member will be familiar with the hardware engineer's craft, thus allowing for effective, efficient working teams.

At this writing, most of the work described in this paper is still in the early development stages. The authors plan to report an update in their presentation at the Annual Conference.

References

1. "Criteria For Accrediting Programs In Computer Science In the United States", Computing Sciences Accreditation Board, January, 1987.

2. Report on the National Science Foundation Disciplinary Workshops on Undergraduate Education, NSF 89-3, April, 1989.

3. Steidley, Carl, "Robotics: A Closer Look at Microprocessor Systems", SIGCSE Bulletin, Vol 23, No 1, pp 265-272, March, 1991.

4. Steidley, Carl, "Robotics: Self-Directed Applied Research for Undergraduates", Proceedings ACM 19th Annual Computer Science Conference, pg 690, March, 1991.

5. Steidley, Carl, "Hardware: An Introduction to the Engineer's 'Nuts and Bolts' for the Computer Science Student", Proceedings ASEE Annual Conference, June, 1989.

6. Steidley, Carl, "A Laboratory for Real-Time Software Systems Development", Proceedings of the ASEE Annual Conference, pp 1124-1132, June, 1995.

7. Tucker, Allen B., et al, "Computing Curricula 1991 Report of the ACM/IEEE-CS Joint Curriculum Task Force", ACM Press, IEEE Computer Society Press, 1991.

CARL W. STEIDLEY earned his Ph.D. at the University of Oregon. His interests are in the applications of artificial intelligence and robotics. He has taught computer science at Central Washington Univ., and Austin Peay State Univ. Before that he taught mathematics and physics at Oregon Institute of Technology. He has had research and development appointments at NASA Ames Research Ctr., Oak Ridge Natl. Labs and ESI in Portland, OR. Before returning to teaching Steidley held engineering positions with Litton Industries, Magnavox Research Labs, Westinghouse and Philco.

MOHAMMED M. "MIKE" ASOODEH earned his Ph.D. at Texas A&M University. His research interests are in telecommunications, systems engineering, and robotics with current emphasis on multimedia development to support his teaching in these areas. Prior to joining the faculty at Southeastern Louisiana University he taught civil engineering at Texas A&M. Prior to that he was an engineering research associate at Texas A&M.