AC 2007-1363: INTERNET-BASED ROBOTICS AND MECHATRONICS EXPERIMENTS FOR REMOTE LABORATORY DEVELOPMENT

Richard Chiou, Drexel University

Yongjin Kwon, Drexel University

Shreepud Rauniar, Drexel University

Horacio Sosa, Drexel University

Internet-Based Robotics and Mechatronics Experiments for Remote Laboratory Development

Abstract

This paper describes a series of laboratory experiments in Internet-based robotics and mechatronics, as well as the design, development, and evaluation of an Internet-based laboratory facility to be used to deliver an undergraduate laboratory course for engineering and engineering technology education. Internet-based robotics and mechatronics can be utilized in a wide range of curricula in both engineering and engineering technology programs. Educators are faced with the challenge of providing students with an adequate laboratory experience that will better prepare students for a corporate world where the need for engineers in the quality, service, and information technology industries is increasing. The uniqueness of the newly developed facility is its modularity in design and the use of commercially available hardware and software technologies. The laboratory consists of Yamaha Robots and machine vision systems, Allen Bradley PLC modules, Webcams and sensors, a data acquisition system, mechanical systems, and software applications for monitoring and control. Using this system, one can quickly use an experimental setup for an application problem, view and program the robots, and control robotic and mechanical motions remotely through the World Wide Web. The use of modern sensors and data acquisition instrumentation for monitoring and control of such an application is also beneficial as laboratory practices for undergraduate classes on Web-based gauging, measurement, inspection, diagnostic system, and quality control.

1. Introduction

The use of Internet-based technologies by industry has grown enormously in recent years[1-5]. Practically all modern manufacturing organizations use the World Wide Web in one form or another: to design and test product ideas, control industrial processes, inspect parts, automate material handling and certain repetitive tasks, and integrate processes and systems. Engineers are challenged to understand these technologies and their strengths and weaknesses, and apply them in a cost-effective manner. It is also very important that these technologies be integrated via the Internet to maximize their effectiveness[6-14].

The Internet-based approach for laboratory development and educational enhancement has been introduced in universities[15-20]. Drexel University's Applied Engineering Technology Program received a NSF CCLI grant (2004-2007) to develop laboratory courses in Internet-based robotics and automation. In conjunction with the grant, the course, MET 205 Robotics & Mechatronics, has been developed and offered at Drexel University. This course provides a requisite understanding of Internet-based robotics/automation/machine vision for students to progress to the advanced level in the curriculum. The course also serves as a means for students to gain exposure to advanced industrial automation concepts before partaking in their required senior design project. The course has an applied learning focus, offering flexibility to the students through an open laboratory philosophy. Since the concepts of Internet-based robotics and mechatronics are best conveyed through application-based learning, the course is divided into two components: a classroom lecture component and an associative laboratory component. The laboratory component is central to the course and is available to the students outside of normal class time. The Web-based system allows robotics and automation to communicate, share design data, information and knowledge through the Internet[21-22]. This allows the students the freedom to explore the concepts of each lesson without time constraints inhibiting learning. In order to provide an enhanced laboratory experience, the students work with real world industrial components. Table 1 provides an overview of lecture and laboratory series in MET 205 Robotics and Mechatronics.

Table 1: Overview of lecture and laboratory series

WEEK	Lecture and Laboratory Topic
1	Introduction to robotics and automation
2	PLC ladder logic programming
3	Robot programming and implementation
4	PLC components and languages
5	Midterm exam
6	Robot sensors and applications
7	Advanced PLC functions and programming
8	Integration of sensors and robots
9	Remote Web operations and vision calibrations
10	E-Manufacturing and Internet-based robotics
11	Final exam

2. Course Development and Laboratory Assignments

The course provides the students with a comprehensive knowledge of Internet-based manufacturing automation using industrial robots and other common machinery and components as shown in Figure 1. Laboratory assignments were used to emphasize important technological issues and provided hands-on design experience with the Internet-based technologies. The specific Internet-based technologies chosen were: computers and networks, robotics, automated inspection and vision systems, PLCs, sensors, automated assembly systems, and automated material handling. Six laboratory assignments were used to reinforce lecture information and to give hands-on design experience. It is believed that hands-on experience is required before learning about Internet-based technologies.

Laboratory 1: Web Based Robotics and Mechatronics Systems

Students learn the introduction of the Web-based robotics and mechatronics system during the first week. Specially designed workstations have been developed for the course as a part of this practicum and are necessary to complete many of the exercises in the course. The laboratory exercises are closely tied to the classroom lecture topics and the students are required to explore and use supplemental resources to complete the exercises. During the laboratory exercises, the students have an opportunity to apply their knowledge by integrating several components together to develop an integrated solution to a manufacturing problem. To instill the teamwork skills desired by industry, the students are required to use a collaborative team approach while completing the exercises.

Figure 1. A workstation with a Yamaha robot, a computer, and a controller, mechanical devices, and Allen Bradley PLC 1756 Series.

All the devices, such as the robot, Web camera, and PLC, are linked using an Ethernet connection. This reduces the wire maze needed to link every device and enables students to operate/control the equipment remotely. All the devices mentioned are connected to a LAN network as shown in Figure 2. These devices are connected by an Internet Protocol (IP) address over the network. The term “network” refers to the connection between the devices in the work area as well as with the users. The network used here is local area network (LAN), which is connected to a Drexel University server. The Web server is connected to this LAN and also from outside, the LAN can have access to the devices on the network. Every device connected to the network has a unique IP address, which is used to connect to them and also recognize them on the network. The IP address is separated into network address and host address sections. The network address section is extracted from the IP address by *and* processing with the subnet mask. The remaining portion is the host address section. Devices belonging to the same network must be set to have the same network address. The host address, however, should be different for every device so that no two devices have the same number. The first and the last host address numbers are reserved for the system (0 and 255), so it should be made sure not to set these as the IP address.

In the process of communication, the IP address of the device must first be specified for the connection. After the connection is made, the actual data is exchanged between the devices. For security reasons all those devices are password protected. This paper focuses on the Internet-based automation technology in communication with a

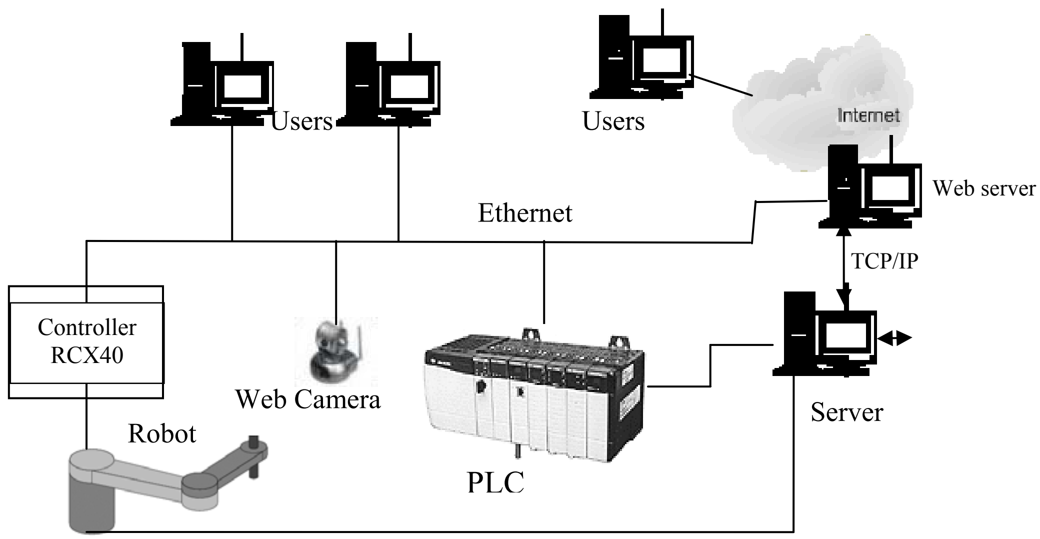


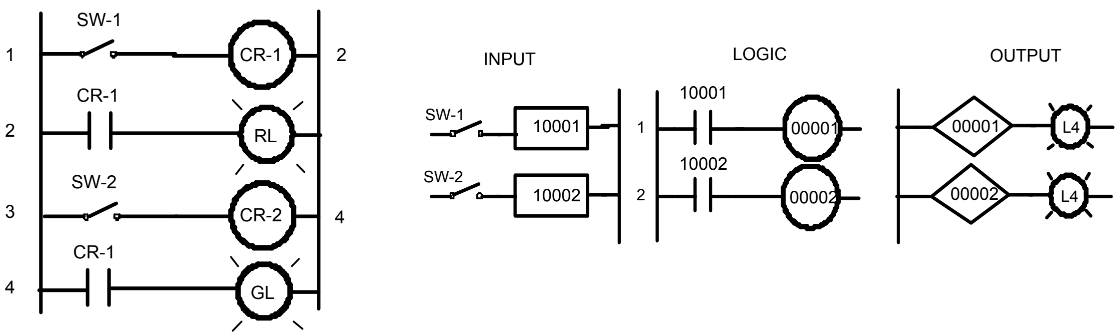
Figure 2. Web-based control architecture.

Yamaha SCARA robot. The controller of the robot (RCX40) can be connected to the Internet directly or to a computer using RS 232C cable on COM 1 port. The software used for communication is VIP for Windows Version 1.6.0 developed by the Yamaha Co. Ltd. The experimental setup includes the following items: ROCKWELL RSLogix 5000, Yamaha SCARA robot, RCX40 robot controller with optional on-board Ethernet card, Yamaha I/O checker, DLink DCS-5300, and HP m1050e PCs. The system also consists of power supplies, DC motors, fans, buzzers, limit switches, relays, and lights. The RCX40 controller can be connected to the Ethernet and controlled using a PC/server. Two Web cameras are used constantly view the robot's movement.

Laboratory 2 - PLC Ladder Logic Programming

In this experiment, students learn the basics of using the ROCKWELL Automation "RSLogix 5000" Enterprise series ladder logic software. Students also learn how to set up uncomplicated ladder logic programs, how to save them on a USB jump drive, and how to run them on the Allen Bradley PLC to control robots and electric actuators. RSLogix 5000 is a powerful industrial software package, which is very similar to many other industrial ladder logic programs in use in industry today. Students program PLC ladder logic, as along with relay ladder logic, although the differences between them are fairly slight. As shown in Figure 3, a typical relay ladder logic control circuit is used for sequence control. Switch 1 (SW-1) turns on control relay 1 (CR-1) which in turn energizes a red light (RL) and, similarly, switch 2 (SW-2) turns on control relay 2 (CR-2) which in turn energizes a green light (GL).

In this laboratory, students create a program for the ControlLogix processor and then verify it is operating correctly, as shown in Figure 4. The ControlLogix processor uses RSLogix 5000 programming software. Students learn that the tag names on the left identify the tag as referring to a local (versus remote) module, in slot 0 or 1, and data type of input (I), or output (O). These tags all refer to the input and output modules. Students learn how to enter a rung that is used to monitor the bit 0 input (the first input) of the input module in slot 0. Therefore, the inputs from slot 0 can be selected. Students click on the "+" next to "Local:0:I" to examine the input data and the module in slot 0. The following screen appears. Notice that in addition to the actual data, students can get information such as a timestamp, and open-wire detection. These are all advanced features of the module that can be used in diagnosing problems in the system.



(a) (b)

Figure 3. (a) Relay ladder logic diagram in comparison with (b) PLC ladder logic diagram.

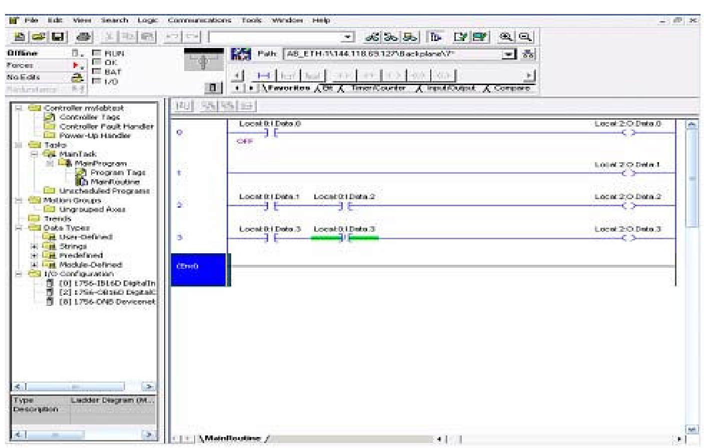


Figure 4: Programmable logic operations with inputs and outputs.

Laboratory 3 - Advanced PLC Functions for Programming

Both TON and TOF relay timers are widely used in most relay manufacturers for engineer designing a relay-control circuit. The TOF function is often confusing to work with and typically is not often found in PLC control languages, since many engineers prefer to use combinations of TON timers, N/O contacts and N/C contacts. Additionally, the terms “Upcounter” (counts up from zero), “Downcounter” (counts down to zero), “Preset Value” (either the initial value or the target value of the counter), and “Accumulated Count” (the variable input, counted at a given time) became commonly used in industry.

These PLC timers are mainly of two types. Most common were the “Timing On” (TON) devices—that is, they produce a time delay after the relay coil is energized. A relay variation of this is the “Timing Off” (TOF) time delay, which is mechanically arranged to give a time delay after the power to the relay coil is cut off. Before leaving the tag database, students have to create a tag called “myTimer” and select the “Edit Tags” tab. For the tag name, the students type “myTimer.” Under the “Type” column, they then click the ellipses and select “Timer.” This will create all of the Timer tags, such as myTimer.acc and myTimer.dn.

Next, students add a rung which divides the timer’s accumulated value by 1,000 to scale the value down to a number in the range of 0 to 10 that they can send to the analog output module. The analog card, by default, is looking for a value between 0 and 10, which it will output to the meter as 0 to 10 volts with the following steps: (1) Add another rung to the program, (2) Scroll to the “Compute/Math” tab and select the “DIV” instruction, (3) Once finishing the rung and right clicking on “meter0,” select New “meter0” and create the tag as follows. The PLC program is shown in Figure 5. Once the timer’s accumulated value is divided by 1,000, the value is sent directly to the first channel (Channel 0) on the analog output module. The program is then saved and downloaded. When the program is in “run” mode, the meter (AO0) moves smoothly from 0 to 10 volts and starts over at 0 volts again.

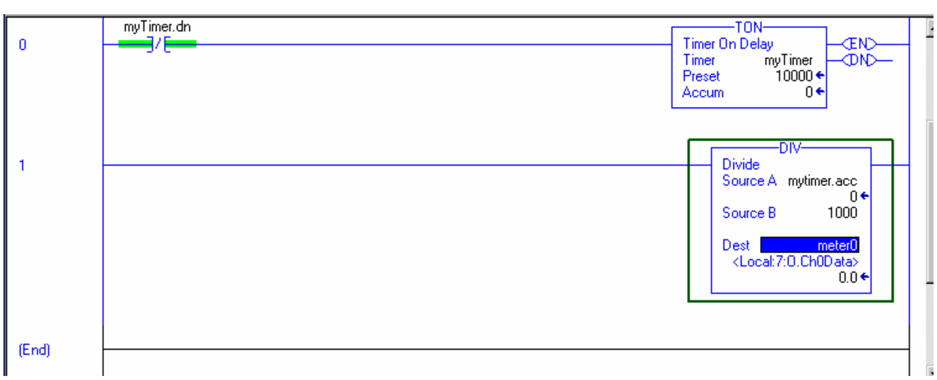


Figure 5. PLC program with Timer and Math Functions

Laboratory 4 - Visual Basic Programming for Internet-Based Robotic Control

Visual Basic (VB) programming for applications in Internet-based robotic control is an attractive alternative to students. VB, with its control objectives and internet library files, make computer programming a very interesting subject. The availability of VB in library functions for Internet-based control enables students to perform sophisticated Internet-based robotics. The TCP/IP is a two way communication protocol. For every instruction issued by the user, the controller sends back an acknowledgement. This ensures that the issued command is executed. The Visual Basic programming language provides Winsock library (winsck.dll) for Internet programming. Basic methods provided by this library that are mainly required for a successful control and monitoring of the robot are connect(), state(), getData(), sendData(), and disconnect(). The system integration with the methods enables the Internet-based robotic control to perform the sequence of tasks outlined in the figure automatically when the students start the experiment from a remote site.

The first step is to establish a connection with the controller when the application starts. The IP address, port number, username, and password need to be provided for the communication channel. After the authentication is completed, the next step is to acquire the robot system information such as current arm position, status of motors, mode, or loaded program. This information is acquired by sending the respective commands and receiving the responses from the controller. The following code illustrates how to recognize the current servo motor status:

```
Public Sub checkServo()
   Dim sBuffer As String
      wxWinsock1.SendData "@?SERVO" & vbCrLf
      wxWinsock1.GetData sBuffer
      If sBuffer = "ON" Then
         cmdServoOn.Caption = "Turn-Off Servo"
      End If
      If sBuffer = "OFF" Then
         cmdServoOn.Caption = "Turn-On Servo"
      End If
   Return
End Sub
```

This function sends the command “@?SERVO & vbCrLf” to the controller and receives the response using wxWinsock1.GetData. The sent command requests the servo status in the controller. This command ends with “vbCrLf” which stands for Visual Basic Carriage Return Line Feed. It is used as the shortcut for the “Return” key. The received data is stored in a variable sBuffer. There can be two responses for the sent command: “ON” or “OFF.” Thus, the required operation can be completed by comparing the responses. If the response is “ON,” the caption on the button “cmdServoOn” is set to

"Turn-Off Servo." Similarly, if the response is "OFF," the caption on the button "cmdServoOn" is set to "Turn-ON Servo." The other commands can also be created to acquire the various status of the robot as shown in the Table 2.

Table 2: Robotic operation commands

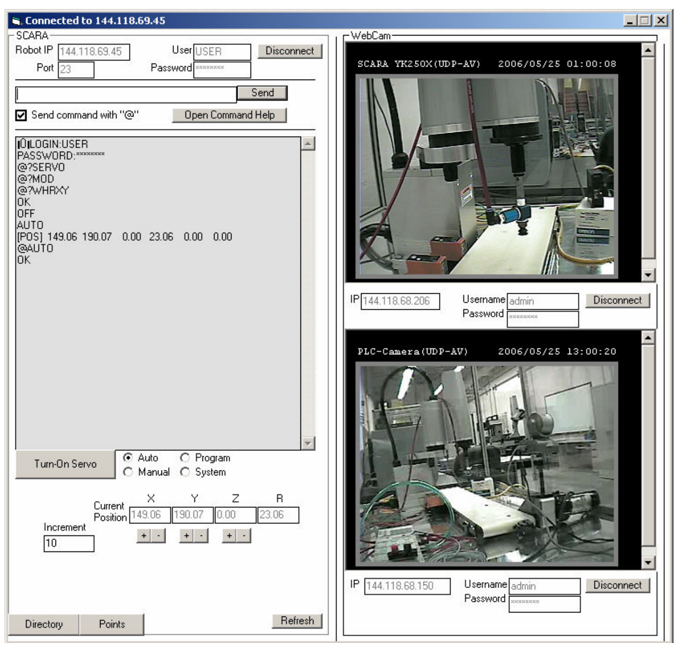
Command	Description
@?SPEED	Check the current speed of the robot
@?WHRXY	Check the current robot position
@?MOD	Check the status (auto/manual)
@?EMG	Emergency stop status
@?MEM	Remaining memory capacity
@?DI(22)	Check the input state of data-input pin 22
@READ DIR	Read the program directory of the controller

The command "WebRauto" application enables streaming of video images from Webcams installed in the robot work cell. Most of the Webcams are provided with the Web-based interface for its control. The images from this camera can be viewed using a Web browser by entering the IP address in the browser. Since the command "WebRauto" is a complete control package, it is necessary to have a video feedback embedded with it. For most Webcams, ActiveX component is used as the control mechanism. This can be incorporated into a VB application for displaying image by using the browser library Microsoft Internet Control component. Below is the part of code used in the "WebRauto" application for creating a website and loading the video image:

```
Private Sub cmdWebcamConnect_Click()
   Dim fso, txtfile, myPath
            If cmdWebcamConnect.Caption = "Connect" Then
            Set fso = CreateObject("Scripting.FileSystemObject")
            Set txtfile = fso.createtextfile("dLinkConnect.htm", True)
            txtfile.write Trim(dlinkTxtStart) & dlinkTxtMiddle & "http://" &
txtWebcamUser & ":" & txtWebcamPassword & "@" & txtWebcamIp &
Trim(dlinktxtEnd)
            txtfile.Close
            myPath = CurDir()
            axBrowser1.Navigate (myPath & "\dLinkConnect.htm")
            controlButton cmdWebcamConnect, "Disconnect"
         …………….
End Sub
```

The button "cmdWebcamConnect" is used for loading the video image from the camera. When it is pressed, a website is composed using the "File system object" provided in VB. The commands *dlinkTxtStart*, *dlinkTxtMiddle,* and *dlinktxtEnd* are the variables that contain the script to be written to the website. The name of the website created is dLinkConnect.htm. This website consists of commands that load the ActiveX component by providing the object ID and class ID, IP address, port address, username, and password.

Figure 6 shows remote controlling and monitoring of a SCARA robot connected to the YAMAHA RCX40 controller via the Internet. When a series of buttons in the Internet-based control system are pressed, they send commands via an Ethernet connection using the methods provided by Microsoft Winsock component library. This program is designed to connect to the controller remotely, receive the system information, send the user input to the robot, and disconnect once the operation is finished. It also provides the students to view the operations at the robot area using two video cameras which can be controlled for pan and tilt motions. Once the controller (RCX40) is connected to Internet, it is treated as a server as it awaits commands from the user/client and sends them to the controller. For viewing the workspace, two Webcams are used as shown in the figure. This program enables to run the robot manually by typing specific destination points or automatically by writing a program. The cameras can be accessed by typing in its IP address. For security reasons, the camera is password protected. It can be accessed in two modes: demonstration mode and complete access mode.



Robotic Control Webcams

Figure 6. The robot is controlled and viewed simultaneously using two Webcams by a connection to the server is established using Visual Basic Software.

Laboratory 5 - Sequential Control Using VIP PLC via Web

A Yamaha RCX40 controller is used for communicating robot input/output signals to and from sensors and actuators. It is a collection of switches, each with a specific function. The operating voltage is 24V and is provided to pin numbers 47-48

(+24V DC) and 49-50 (GND). The connection to the controller is made using a MR-50LM (50 pin) connector. Notation used for input data is DI and for output DO followed by two digit numbers (e.g., DI01, DI11, DO33). Pins 11 (DI20) through 26 (DI37) are used for general purpose input signals. Pins 39 (DO20) through 46 (DO27) are used for general purpose output signals. Besides normal robot programs, this controller can execute sequence programs in response to the robot input/output (DI, DO, MO, LO, TO, SI, SO) signals in fixed cycles. The cycle is determined by the program capacity. This means that two different programs, a robot program and a sequence program, can be executed simultaneously. Using this function allows the controller to operate as if it had a built-in PLC (programmable logic controller). The logic functions performed by the program are determined by the contact configurations used to obtain the AND, OR, and NOT logic functions as shown in the following examples:

DO(20)=~DO(20)

DO(25)=DI(21) AND DI(22)

MO(26)=DO(26) OR DO(25)
:

The PLC flip-flop timer sequencing technique can be programmed for stepping a program they design through a typical process industry cycle, using the electrical test bench for practical demonstrations. Industries which consist of the processing of liquids and/or gases into useful products are generally known as the process industries. Not only does the PLC can be performed the sequencing from one step to another, but with the use of Yamaha Robotics digital I/O, it also can control temperature, pressure, and the flow rate of fluids. Because of the precise nature of this type of control, a batch processing requires highly sophisticated PLC programming and engineering.

Figure 7 shows the control circuits for integration of PLC sequence with a robot and mechanical systems for a simulated process through the Internet. The control is planned for a logical system by labeling inputs and outputs. The students have to make sure that the sequence programming is in agreement with the logical system and the required electric circuit is connected with the interconnect wiring to the Yamaha RCX40 controller, as well as the input and output modules. The material handling system uses a diaphragm-type vacuum cup as it provides large suction force for small suction volumes. The FESTO suction cup, with a diameter of 15 mm, is used in the material handling system.

The system integration enables the computer to automatically perform the sequence of tasks outlined by a user when the students start the equipment from a remote site. The sequence control is primarily accomplished by the use of Yamaha VIP programming. Yamaha VIP Windows is applied as assistant software for multiple axes robot controller and robots. It can be used to create and edit programs, point, parameter, shift and/or hand data directory used with robot controller. The Yamaha VIP software for robotic programs enables information exchange between the various levels of the control architecture. For the pick and place program, students check for a part at position 1 OR position 2 and check for a robot to pick the part and drop it at position 3. The cycle is repeated.

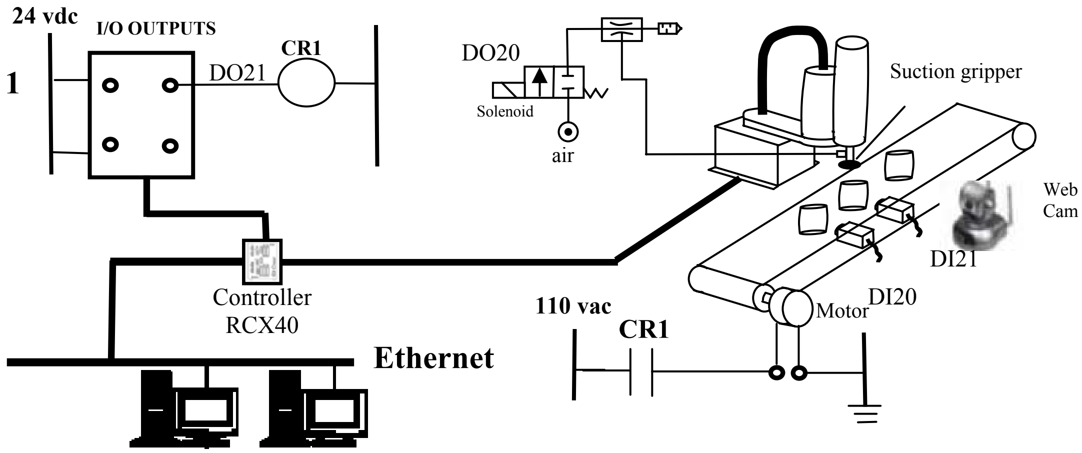


Figure 7. Control circuits for integration of PLC sequence with a robot, and mechanical systems for a simulated process through Internet.

```
*Top:                              'Defines the label Top
WAIT DI2(1)=1 OR DI2(0)=1,5000     'Waits for DI21 or DI20 to turn on
GOSUB *GetPart:                    'Executes subroutine GetPart
GOTO *Top                          'Go to the label Top
*GetPart:                          'Defines the label GetPart
IF DI2(1)=1 THEN                   'If DI21 is on…
        MOVE P,P2,Z=0.0            'Move to point 2 with a Z arch of 0mm
        SET DO2(0)                 'Turns on DO20
        GOSUB *DropOff             'Executes the subroutine DropOff
ELSE                               'If DI21 is not on…
        IF DI2(0)=1 THEN           'If DI20 is on…
                MOVE P,P1,Z=0.0    'Move to P1 with a Z arch of 0mm
                SET DO2(0)         'Turn on DO20
                GOSUB *DropOff     'Execute the subroutine DropOff
RETURN                             'Return to the GOSUB
*DropOff:                          'Defines the label DropOff
MOVE P,P3,Z=0.0                    'Move to P3 with a Z arch of 0mm
RESET DO2(0)                       'Turn off DO20
RETURN                             'Return to the GOSUB
```

The controller (RCX40) is connected to a computer (we treat it as a server as it awaits commands from user/client and sends them to the controller). The connection between the controller and the server is made by using RS 232C cable connected to the COM 1 port of the server. For viewing the workspace, two D-Link Webcams, which have pan/zoom/tilt functions, have been used as shown in Figure 8. The software used for controlling the robot is VIP Window software developed by Yamaha Robotics. This software enables the user to run the robot manually by typing specific destination points or automatically by writing a program. The D-Link cameras can be accessed using

Microsoft Internet Explorer and typing in their IP address. In demonstration mode, features like pan, tilt, and zoom are disabled. The VIP Software on the server can be accessed from anywhere using Remote Desktop.

Figure 8. Connection to the server is established using RDC, and VIP Windows software is used for controlling the robot. The robot is viewed simultaneously using two Webcams.

Laboratory 6 - Online Experiments for Engineering Students

The authors have been developing a strong relationship with other universities for the purpose of developing online experiments for applied engineering students. The team has tested and developed effective online laboratory courses, which emphasize the use of

(a) (b)

Figure 9. (a) A robot control window, machine vision system, and Webcam for a remote inspection. (b) Two viewing windows (live robot images from Webcams) and the robot control interface for online experiments.

the Internet-based robotics, automation, and machine vision techniques, with the Industrial Engineering department in Arizona State University. Both Drexel University and Arizona State University had a joint course offering (MET 205 Robotics & Mechatronics at Drexel University and IEE 563 Distributed Information Systems at Arizona State University) for online robotic experiments in the fall of 2005. The schools formulated geographically separated virtual teams and collaborated on the laboratory experiments using the Web-enabled robotic systems over the Internet. Throughout the 10-week term, students at Drexel University spent eight weeks on laboratory experiments to get familiar with topics in robot workings, operations, programming, and sensor integration. The students at Arizona State University programmed, debugged, uploaded, and tested the robotic systems over the Internet. Students from Arizona State University successfully implemented Internet applications to remotely operate the robot in the form of information interface as shown in Figure 9. The last two weeks were allocated to the specifically designed online robotic experiments for both schools. Such online laboratories enable multiple institutions to share expensive laboratory resources, hence providing engineering and engineering technology students access to more sophisticated concepts and laboratory experiences as shown in Figure 10.

Figure 10. Students worked on the laboratory projects in the MET 205 Robotics and Mechatronics offered at Drexel University.

3. Evaluations

The Internet-based laboratory course is a new concept, and evaluation of the facility for its usefulness provided an understanding in terms of students' point of view. With this in mind, a weekly survey was incorporated within the facility that students need to complete at the end each laboratory session. In addition to the weekly surveys, the facility is equipped with collecting facility usage data. These data allowed the facilitator to know the level and timings of use of the facility and hence provided a broader understanding of the students' behavior in terms of use of the facility. Course reviews by students were very positive. The benefits of an active learning model are derived. Some students complained about too much time involved with the laboratory assignments, but many commented positively about the knowledge they gained. Students commented that they enjoyed working in the laboratory. The evaluations were adaptations of forms developed by our team, a respected authority on student-centered learning methods in engineering technology education. Student evaluation was conducted at the last week of the class. A customized set of questionnaires were given to students, and the evaluation results were compiled. The results showed the highly supportive evidence toward the intended course outcomes.

4. Conclusions

The paper presents the development of a laboratory course for teaching undergraduate introductory Internet-based technology on robotics and mechatronics. Offering a laboratory course over the Internet as a part of a regular program is a unique initiative. Such an advanced technological curriculum has offered a unique career pathway to students interested in advanced engineering technology through the full-time program at Drexel, dual degree programs with community colleges, and 2+2+2 programs incorporating also high schools in the Greater Philadelphia Region. Furthermore, the project will reinforce the ongoing initiatives to revitalize the regional manufacturing sector by providing highly skilled graduates to meet the demands of new advanced technologies.

Acknowledgement

The authors would like to thank the National Science Foundation (Grant No. NSF-DUE-CCLI- 0410719) for its financial support of the project and Yamaha Robotics for its in-kind gift of robot equipment.

References

1. *SME Manufacturing Engineering* – Automation Technology: Robotic Automation Can Cut Costs, pp. 65 - 72, December 2005.
2. *SME Manufacturing Engineering* – Tomorrow's Factory: Manufacturing Industry Takes First Steps Toward Implementing Collaborative E-Manufacturing Systems, pp. 43-60, Nov. 2001.
3. *SME* – Machine tools begin connecting to the Internet Manufacturing, 9/2001.
4. Koc, M., Ni, J. and Lee, J., "Introduction of e-manufacturing," *Proceedings of the International Conference on Frontiers on Design and Manufacturing*, Dalian, China, July 2002.
5. Lee, J., 2003, "E-manufacturing—fundamental, tools, and transformation," *Robotics and Computer-Integrated Manufacturing*, Volume 19, Issue 6, pp. 501-507.
6. Ahn, S. H., Bharadwaj, B., Khalid, H., Liou, S. Y. and Wright, P. K., "Web-Based Design and Manufacturing Systems for Automobile Components: Architectures and Usability Studies," *International Journal of Computer Integrated Manufacturing*, 15, pp.555–563, 2002.
7. Feng, J., "Internet-Based Reverse Engineering," *International Journal of Advanced Manufacturing Technology*, January, 2002.
8. Hu, H., Yu, L., Tsui, P. W. and Zhou, Q, "Internet Based Robotic System for Teleoperation, Assembly, and Automation," *International Journal of Assembly Automation*, Vol.21, No.2, pp.143-151, 2001.

9. Huang, G. Q. and Mak, K. L., "Web-integrated Manufacturing: Recent Developments and Emerging Issues," *International Journal of Computer Integrated Manufacturing*, 14, pp.3–13, 2001.
10. Potgieter, J and Bright, G., "Modular Mechatronics Control System for Internet Manufacturing," *Proceedings of the 18th International Conference on CAD/CAM*, Robotics and Factories of the Future, Porto, Portugal, pp. 529-536, July 2002.
11. Ravindra Thamma, Luke H. Huang, Shi-Jer Lou, and C. Ray Diez, "Controlling Robot Through Internet Using Java," *Journal of Industrial Technology*, Vol. 20, No. 3, June – Aug., 2004.
12. Shaw, M. J., "Information-Based Manufacturing with the Web," *International Journal of Flexible Manufacturing Systems*, 12, pp.115–129, 2000.
13. Taylor, K. and Dalton, B., "Internet Robotics: A New Robotics Niche," *IEEE Robotics and Automation*, Vol.7, No.1, pp.27-34, March 2000.
14. Kwon, Y., Chiou, R., Rauniar, S. & Sosa, H., "Positioning Accuracy Characterization of Precision Micro Robot over the Internet,"*Journal of Advanced Manufacturing Systems,* Vol. 5, No. 5, pp. 45-57, 2006.
15. Chumbley, L. S., Hargrave, C. P., Constant, K., Hand, B., Andre, T., and Thompson, E. A., Project ExCEL: Web-based scanning electron microscopy for K -12 education, *Journal of Engineering Education*, April, 203-210, 2002.
16. Ko, C. C., Chen, B. M., Hu, S., Ramakrishnan, V., Cheng, C. D., Zhuang, Y., and Chen, J., A Web-based virtual laboratory on a frequency modulation experiment, *IEEE Transaction on Systems, Man, and Cybernetics*, - Part-C, Applications and Reviews, 31(3), 295-303, 2001.
17. Azad, Abul and Song, Xueshu, "Internet Based Physical Experiments: Application within a Laboratory Course *Proceedings of the American Society for Engineering Education Annual Conference & Exposition*, June 18-21, 2006, Chicago, Illinois.
18. Hsieh, Sheng-Jen and Patricia Yee Hsieh, "Web Based Programmable Logic Controller Learning System," *32nd ASEE/IEEE Frontiers in Education Conference*, November 6 - 9, 2002, Boston, MA.
19. Richard Chiou, Yongjin Kwon, Shreepud Rauniar, and Horacio Sosa, "Laboratory Development for Robotics and Automation Education Using Internet Based Technology," *Proceedings of the American Society for Engineering Education Annual Conference & Exposition*, June 18-21, 2006, Chicago, Illinois.
20. Paul I-Hai Lin, Hal Broberg, and Aik Mon, "A Web-based Lab for Distance Learning," *Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition.*
21. Richard Chiou, Yongjin Kwon, Shreepud Rauniar, and Horacio Sosa, "Visual Basic Programming for Internet Based Robotic Control," To appear in *Computers in Education Journal,* 2007**.**
22. Kwon, Y., Rauniar, S., Chiou, R. & Sosa, H., "Remote Control of Quality Using Ethernet Vision & Web-enabled Robotic System," *Journal of Concurrent Engineering: Research and Applications,* Vol. 14, No. 1, pp. 35-42, 2006.