

Mechatronics Hands-on Training through the Development of an Internet-based Automatic Control Laboratory

J. C. MARTÍNEZ-GARCÍA and R. GARRIDO

Departamento de Control Automático/Centro de Investigación y de Estudios Avanzados del IPN
A.P. 14-740, 07300 México D.F., México.

E-mail: {martinez,garrido}@ctrl.cinvestav.mx

Abstract— This paper summarizes the authors experience from Mechatronics and Automatic Control Teaching at the undergraduated level. A hands-on remote Internet-based control laboratory project is discussed here as way for training.

Keywords— Mechatronics and Automatic Control Education, Internet-Based Experimental Facilities, Hands-On Projects.

I. INTRODUCTION

It is clear that modern industry needs a new kind of engineering approach mainly for the design of technical systems operating mechanically with respect to at least some kernel functions but with more or less electronics supporting them decisively, i.e., modern industry needs Mechatronics engineers. Because of the nature of Mechatronics, defined as a synergistic combination of precision mechanical engineering, electronic control, and systems thinking in the design of products and processes, a learning methodology based on hands-on projects has been usually considered as an ideal way to train engineers at both the undergraduate and the graduate levels (see for instance [1]). By other way, because of the increasing availability of Internet services (mainly due to the reduction of cost in computers and connections), there exists now an international effort concerning the use of Internet for educative purposes. As far as Mechatronics and Control Applications are concerned, the theoretical basis can be enhanced through Internet-based remote courses and virtual experimentations (see for instance [2]). Moreover, Internet can be used for performing real-time experimentation (nothing is more educative than experimenting with real systems). For the above reason, the development of Internet-based real-time experimental services is naturally justified. Internet increases in a spectacular way the availability of expensive experimental resources, which is always important, to a wide audience.

This paper summarizes the experiences resulting from the development of a project concerning a remote Internet-based automatic control laboratory. The aim of the project was the implementation of a remote laboratory in order to

provide experimental facilities to the Internet community and also to perform both basic and advanced research in Mechatronics and in Automatic Control. The strategy followed for the development of the project was heavily oriented to the training of engineers.

First of all, we briefly present in Section II the profile of the students team charged of the project. Section III is devoted to describe the Internet-based laboratory.

We present in Section IV, following a sequential description, the development of the project. We summarize the main results in Section V and we finish in Section VI with some perspectives and concluding remarks.

II. THE STUDENTS

The students concerned with the experience reported here were undergraduate ones from Escuela Superior de Cómputo (ESCOM). This school, located at the north of Mexico City, belongs to the National Polytechnic Institute of Mexico (IPN), which is mainly specialized in industrial necessities-based engineering education.

The students followed a four year academic program, which was oriented to computer engineering and electronics. In fact, the program was heavily focused in computer software design. Our concerned team was composed of five students (one woman and four men). The implementation of the remote laboratory was the graduation project of the team and was conceived as a way to reinforce their theoretical basis and their skills to solve practical problems.

III. THE PROJECT

The especification of the remote Internet-based control laboratory project was presented to the students as follows (see Figure 1):

Consider a PC networked to Internet (via an Ethernet card) with a MS-Windows NT Server 4.0 [3] as its operative system. Also, consider a DC-motor including an optical incremental shaft encoder and a power amplifier to be given (DC-Servosystem). Develop a Worl Wide Web site and implement, as a Hyper Text Transfer Protocol (HTTP) service, a basic Automatic Control Laboratory based on the DC-motor. The desired acceleration, velocity and position

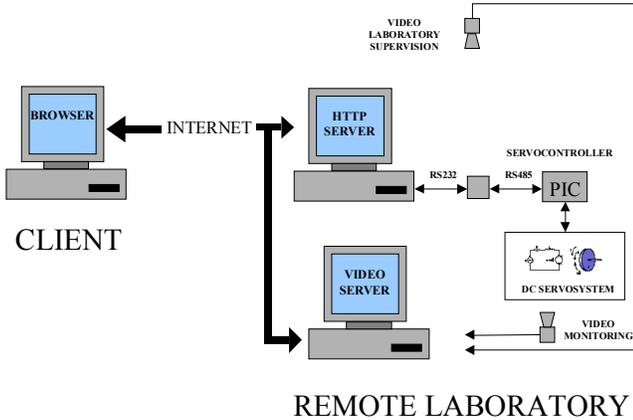


Fig. 1. Architecture of the Internet-based laboratory.

of the axis of the DC-motor, linked to the PC which supports a HTTP server, must be specified by a Client accessing the service via a standard Internet connection, using any standard Browser, available freely on the World Wide Web. The operation of the DC-Servosystem must be monitored by the Client via streaming video. The Client behaviour must be monitored by the administrator of the remote laboratory through the World Wide Web.

Figure 1 shows a scheme of the remote laboratory. We took the DC-motor as our initial brick because this system is interesting from at least two points of view: its control is a standard topic in Mechatronics and Automatic Control education and the DC-motor is an essential component for the implementation of more complex systems, like robot manipulators, which will be added to the remote laboratory in the near future.

As far as the connection between the DC-motor and the HTTP server was concerned, we did not propose any particular technology to the students, i.e., we did not specify the control hardware. Moreover, we did not propose any particular HTTP server. A full academic year was dedicated to the project, which was supported by the mexican government, through its Council of Science and Technology (Conacyt).

IV. THE DEVELOPMENT OF THE PROJECT

A. The DC-Servosystem

A 24 volts DC-motor, a power amplifier and a shaft incremental optoelectronic encoder were chosen. This system is presented in Figure 2.

B. The Controller Application

We present here the procedure followed for the implementation of the Internet-based application which controls the DC-motor. This procedure has been mainly proposed by the students.

1. First of all, the students participate in a 20 hours-course concerning Internet communication protocols, HTML and HTTP servers.

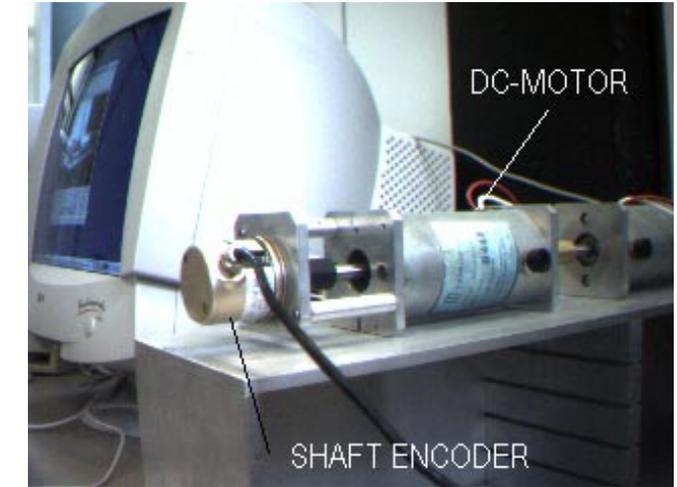


Fig. 2. DC-Servosystem.

2. A HTTP server was chosen. The first choice was the public-domain HTTP server Apache (see [4]). This server was tested and a basic WWW site was implemented. In order to control the DC-motor, two alternatives were considered: a PC bus-based controller board, or, a microcontroller linked to the PC as a peripheral.

The second option was chosen since a PC-based controller board requires its specific software to be adapted to the HTTP server, which is very consuming in terms of development time. Thus, a microcontroller linked to the PC supporting the HTTP server was taken, in order to implement the control system.

3. A commercial board (see Figure 3) which includes a Proportional-Integral-Derivative (PID) position and velocity controller was chosen: the PIC-SERVO board by J R Kerr (see [5]), which has a PID controller implemented in a Microchip PIC16C73 microcontroller (see [6] for the details). This board receives the PID parameters, as well as the position, velocity and acceleration settings through a serial port (RS-232 protocol). Several microcontrollers can be linked to the same serial port using a RS-485 adapter. The microcontroller was then linked to the PC supporting the HTTP server via the COM 1 serial port.

4. In order to send the PID parameters and the desired position, velocity and acceleration settings to the microcontroller, it was necessary to implement a C-language application. At this level, it was considered that the best option to link the application with the HTTP server was through the so-called CGI-BIN modality (see for instance [7]), which allows an application to be executed by a specific Client request. Because of the lack of clear information of the HTTP server Apache, as far as the CGI-BIN modality is concerned, it was decided to switch to a commercial HTTP server. The O'Really's Website was then considered (see [8]). It was a good choice, because it was very easy to configure.

5. To implement the CGI-BIN application concerning the serial-based microcontroller, a MS-DOS application developed by the microcontroller's manufacturer was first tested

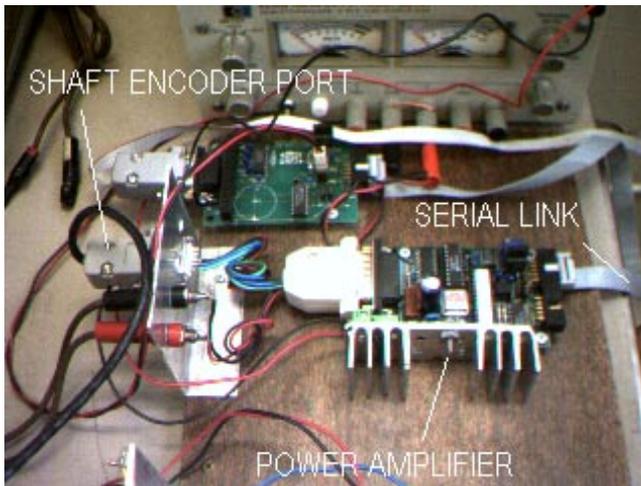


Fig. 3. The PIC-SERVO board.

in order to master the in-board protocol. Once the application was tested, a new modified 32-bits version was designed and successfully tested under the MS-Windows NT Server 4.0 operative system. We shall call this application in the sequel the *Controller Application*.

6. Since the PID parameters, as well as the position, velocity and acceleration settings, can be stocked as ASCII code in a simple text file, the setting values can be sent by the Client in an easy way. Indeed, the Client introduces the values in an interactive Form accessed through the browser and then, it sends them via the POST HTTP modality. The values are then stocked in a ASCII text file, in order to be recovered by the controller application executed by a Client request.

7. In order to check and track the behaviour of the Client, an ASP (MS - Active Server Page)-based data-base was conceived, implemented and linked to an automatic administrative service. This service assign an identification login and a password to any unregistered Client (which ask for the service) and checks the corresponding information concerning a registered Client. The automatic administrative service notifies the Administrator the data assigned to the Client, which he can eventually modify.

V. MAIN RESULTS

In four months the students implemented the basic Internet-based Automatic Control Laboratory kernel. Indeed, a DC-servosystem can be now reached by a Client connected to Internet. In Figure 4 we show the WWW form which receives the settings corresponding to the position, velocity and acceleration especifications. The Client receives as an answer to its request a message which shows the especificed settings, in order to be validated. If the Client validates the information, he (or she) push a botton and the Controller Application is executed by the HTTP server. This application recovers then the new position of the DC-motor axis and transmit it to the Client.

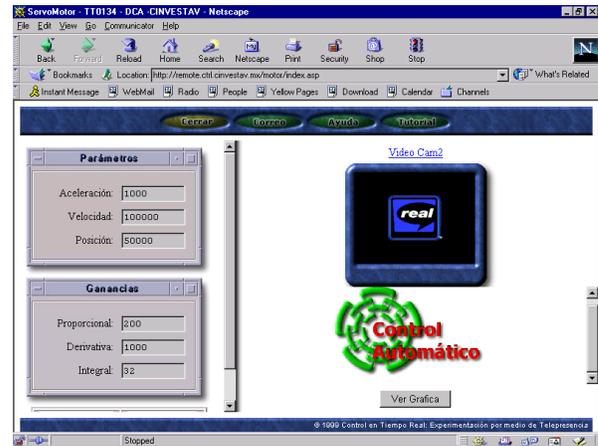


Fig. 4. WWW Form for the operation of the DC-Servosystem.

VI. PERSPECTIVES AND CONCLUDING REMARKS

Nowadays a new team of undergraduate students are extending the Controller Application in order to include several DC-servosystems. A two degrees-of-freedom planar manipulator has been designed and will be build in the near future. The manipulator will be controlled with two DC-servosystems linked to the Controller Application. A movil robot will also be included in the remote laboratory.

A JAVA based human-machine interface is being conceived. At this level, the former team of students followed a course on JAVA programming as a part of the project.

As far as the monitoring is concerned, a comercial streaming video and audio server was chosen (the Basic Server Plus software by RealNetworks [9]) and was integrated to the Internet-based automatic control laboratory. It is always interesting to see what real systems do, but some times it is not possible to obtain a qualitative comprehension of the system's dynamic behaviour via video information. For that reason a more classic 2-D visual interface was added to the service. This tool (in fact a JAVA Applet) allows the Client to follow the behaviour of the servosystem in response to the specified commands (see Figure 5).

We evaluated on-line the behaviour of the former engineering students team, and we can conclude that the hands-on approach was a very good one. Since the students team was obligated to use very efficiently its mechanics, electronics and informatics skills. Initially, the students have not any knowledge on Internet programming and now they have a very good one. However, it was necessary to increase their level of knowledge in basic Automatic Control theory. Currently, the students who were implied in the project work mainly in Internet-based enterprises.

Acknowledgement 1: We would like to thank Ivonne Praxedis, Eduardo Pérez, Guillermo Pérez, Daniel Paredes and Alejandro Rada (our former engineering students team), because of their effort and the good quality of their work.

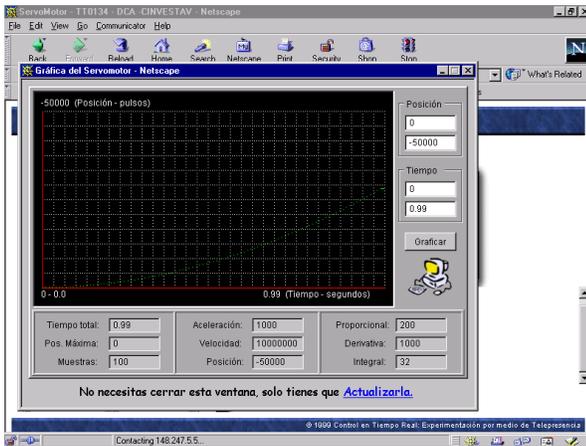


Fig. 5. 2-D monitoring tool.

REFERENCES

- [1] M. K. Ramasubramanian: *Teaching mechatronics design at North Carolina State University*. Proceedings of Mechatronics 96, Cal. Poly, June 13-15, 1996.
- [2] G. Salazar-Silva, J.C. Martínez-García and R. Garrido-Moctezuma: *Enhancing basic robotics education on the Web*. Proceedings of the American Control Conference (ACC'99). San Diego, CA, 1999.
- [3] C. Russel and S. Crawford: *Running Microsoft Windows NT Server 4.0*. Microsoft Press, Redmond, Washington, 1997.
- [4] The HTTP Apache Server web site:
- [5] J. R. Kerr: <http://www.jrkerr.com>.
- [6] The Microchip Inc. web site: <http://www.microchip.com>.
- [7] M. Felton: *CGI Internet Programming with C++ and C*. Prentice Hall, New Jersey, 1997.
- [8] O'Really Website: <http://software.oreally.com>.
- [9] Real Networks: <http://www.real.com>.