

REMOTE LABORATORY FOR AUTOMATION EDUCATION

**José M. Azorín, Luis Payá, Luis M. Jiménez,
Nicolás M. García, José M. Sabater**

*Dpto. Ingeniería de Sistemas Industriales, Universidad
Miguel Hernández, Elche (Alicante), 03202 Spain*

Abstract: In this paper, advances in the implementation of a remote laboratory that allows students to perform automation experiments through Internet are presented. Nowadays a supervision and control web tool of an electropneumatic manipulator managed by a PLC (Programmable Logic Controller) has been developed. In the paper, new characteristics that are being added to use this tool in control and automation courses at the Miguel Hernández University (Spain) are explained. *Copyright ©2004 IFAC*

Keywords: Remote control, virtual laboratory, Internet, control education, process control.

1. INTRODUCTION

The experimental formation of students has always been a very important subject in control and automation education. Habitually, experimental systems, e.g. servomotors, inverted pendulums, temperature control systems or coupled water tanks apparatus, have been used in control courses. On the other hand, PLCs that control different physical processes, e.g. classification systems or manipulators, are usually employed in automation courses.

However, there are some factors that degrade the experimental formation of the student:

- Scheduling. The students can only perform experiments in the timetables fixed by the teacher, so the student can not experiment freely and he can not analyze with a bigger emphasis and longer time the aspects that he considers more important.
- Limited number of equipments. Usually the experimental systems are expensive, therefore universities only purchase a limited number of these systems and each system is used

by several students simultaneously. This produces that the student, as in the previous item, can not experiment freely to improve his knowledge.

- Evaluation process. The traditional evaluation systems of experiments performed by students are reports that must be checked by the teacher to verify if the student has reached the desired level of knowledge. However, in this process the student does not know the result of the evaluation until the professor feedbacks to him the result, and, in addition, in that moment the student can not modify the report delivered to improve the evaluation.

The expansion of Internet from decade 90 has affected a different scopes of the society, including the educational scope. In this area, Internet offers multiple possibilities, as the remote learning through online courses (Poindexter and Heck, 1999), or the remote access to laboratories (Schmid, 1999) (Ko *et al.*, 2001).

The remote access through Internet to perform experiments solves the factors previously mentioned that degraded the experimental formation of the student:

- Scheduling. If the student can freely access to the laboratory through Internet, he will be able to dedicate all the desired time to perform the experiments, so he will be able to improve his knowledge.
- Limited number of equipments. The limitation of the equipments number will not affect the experimental formation of students, because all students can access individually to the same experimental system.
- Evaluation process. Through an online evaluation system it is possible to feedback to the student the evaluation result in real time. This way, the student can modify his report before delivering it for his final evaluation.

We have developed at Miguel Hernández University (Spain) a tool that allows the supervision and control through Internet of a electropneumatic manipulator managed by a PLC (Paya *et al.*, 2003). In this paper, new characteristics that are being incorporated to this tool with the purpose of using the tool in control and automation courses are described.

The paper is organized as follows. Section 2 describes the system architecture that allows the supervision and control of a electropneumatic manipulator through Internet. In Section 3 the characteristics that are going to be incorporated to develop a remote laboratory for automation education are presented. Section 4 explains the web system that allows the execution and evaluation of the automation experiments. Finally, Section 5 summarizes the conclusions of this paper.

2. ARCHITECTURE OF THE SYSTEM

The architecture of the system has been developed trying to reach the maximum simplicity on its use for the student, but allowing him to have a full control of the process. This way, the main goal was to make accesible all the PLC inputs and outputs through any web browser and make possible the downloading of new programs to the PLC in a remote way, even if the student does not have the specific PLC software.

A client/server architecture has been implemented to reach these goals. This architecture allows remote users to visualize in any web browser a Java applet where the process variables are accesible. The Java applet has been included in a HTML page that resides in a server with Windows NT 4.0 powered by the Internet Information Server (IIS) software of Microsoft. Nowadays we are mi-

grating the web server from IIS to Apache server (Apache, 2004).

A Visual Basic application that is located on the server side has been developed to access the PLC variables from the applet. This application and the PLC are directly communicated. The application and the applet exchange the information through the TCP/IP protocol using sockets. This architecture has been developed using a scaled electropneumatic manipulator. However, the architecture can be applicable to the monitoring and control of any industrial process.

Next, the hardware and software architectures are detailed.

2.1 Hardware Architecture

Fig. 1 shows the distribution of the elements used in the laboratory setup. The laboratory is composed by a physical process, a PLC, a server PC and an image server.

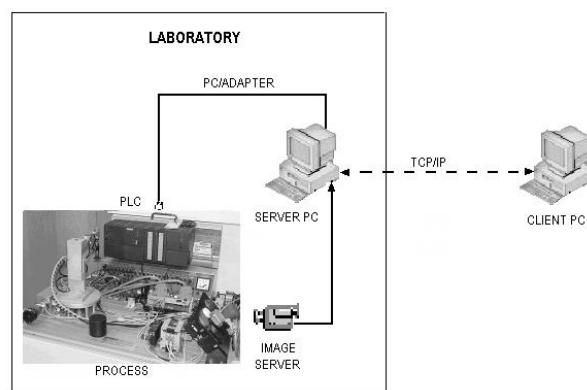


Fig. 1. Hardware architecture of the system.

The physical process is an scaled electropneumatic manipulator composed by two axis and a holder vacuum system. The first axis is a rotational joint actuated by a 24V DC servomotor. The joint position can be known with three microswitches and one encoder. The second axis consists on a double effect pneumatic cylinder. Besides, there is a control panel with start and emergency push-buttons, and a manual/automatic selector. The manipulator is controlled by a Siemens S7-314 IFM PLC. The server PC communicates with the PLC through the PC-Adapter cable, that connects the PLC MPI interface with any serial port of the PC.

An AXIS image server with a Sony EVI D30 video camera has been installed to provide to the remote user the real images of the process. This image server has a embebed web server, therefore it does not require any additional software to operate.

Any PC with a web browser and the Java plug-in (version 1.3 or higher) may be a client PC that can access to the remote laboratory through Internet. The student only needs a text editor to create his control programs for the PLC.

2.2 Software Architecture

As it has been appointed previously, a client/server architecture has been implemented. Next, the software applications developed are described.

Server Application. The server application has been developed using the Microsoft Visual Basic 6.0 programming environment. It uses OCX controls and functions provided by the Simatic Computing and Command Interface libraries of Siemens (Siemens, 2000). Using these libraries, the application can access to the PLC inputs and outputs, read or write PLC internal variables, and download new programs to the PLC. A graphic user interface (GUI) allows to use the server application as a local SCADA in case the remote access through Internet was not necessary.

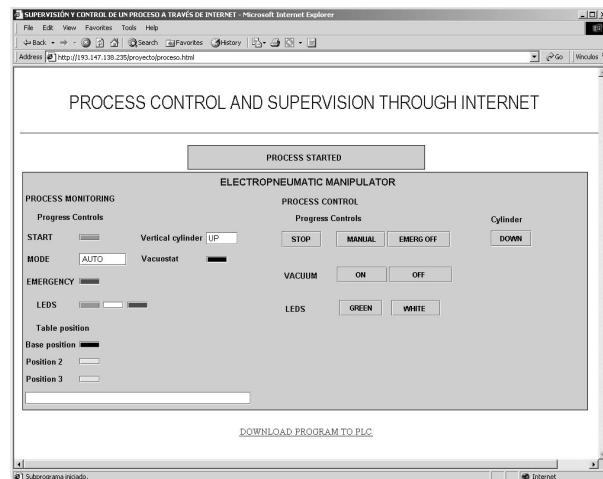


Fig. 2. Java applet developed running in a PC client.

Client Application. The client application has been developed as a Java applet (Java, 2004). The GUI has been built using Swing components. This GUI is very similar to the server application GUI, and it is divided in two parts, Fig. 2. The first one is the process monitoring area, where the student can observe online the value of the process variables. The second one is the process control area, where the user can act on the remote PLC pressing buttons of the application. The applet communicates with the server application using sockets, through the TCP/IP protocol, to access all the process variables.

When any process variable changes its value (e.g. the vertical cylinder goes down), the server application sends a sequence of bytes through the socket that indicates the variable that has changed and its new value. Then, the associated Swing component in the GUI will change its appearance to show the new state of the process variable. On the other hand, an event is generated when the student acts on a button of the applet to carry out an action over the process. This event sends to the server application a sequence of variables that indicates the PLC variable to modify and its new value. In Fig. 3 the information flow of the monitoring and control process is shown.

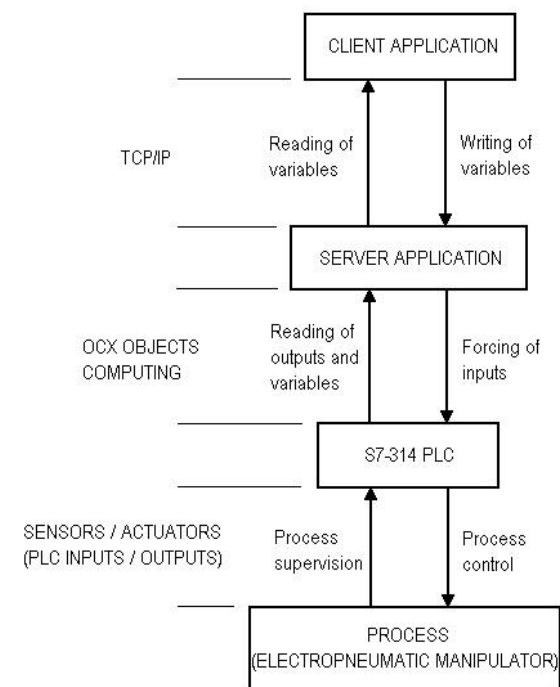


Fig. 3. Information flow of the monitoring and control process.

Additional Applications. Several additional applications have been implemented to allow the interaction of the remote user with the process.

An application that controls the automatic wake-up and stop of the server application has been developed to avoid that the server would be always running, waiting for connections of remote users. An auxiliar applet implemented in the client side requests a connection to an auxiliar application that is always running in the server side without any graphic interface. This application checks if the server application is running to decide if it starts the server application or it informs to the auxiliar applet that the server application is busy.

Several applications are necessary for download programs from the client side to the PLC. A

HTML form and a CGI (Common Gateway Interface) application have been implemented to transfer the program from the hard disk of the client to the hard disk of the server. The user indicates the program files in the HTML form. The CGI application receives the data from the form and saves the program files in the server hard disk. The additional functions necessary to download programs have been implemented in the server application using the Siemens Command Interface library. These functions allow to create an empty Simatic project, to import the source file and symbol table to the new project, to compile the project, to download the blocks compiled to the PLC and to run the user program. The information flow of the program downloading to the PLC is shown in Fig. 4.

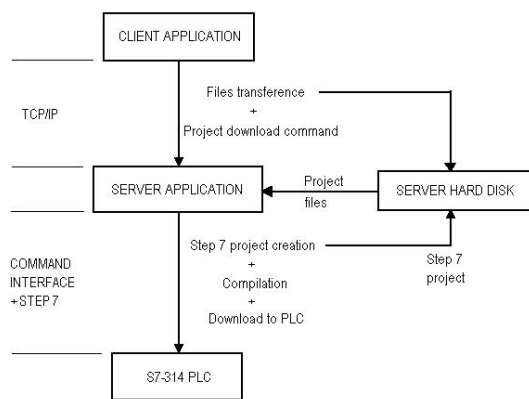


Fig. 4. Information flow of the PLC program downloading.

Furthermore a DLL library has been implemented using Microsoft Visual C++ 6.0 compiler to close the windows that appear in the server side during the downloading program process.

3. REMOTE LABORATORY DEVELOPMENT

In this section the new characteristics that must be added to the tool explained in last section to develop a remote laboratory for automation education are explained.

The characteristics that are going to be incorporated to the tool with the purpose that the students of automation courses can use it to perform experimental exercises are the next:

- Three dimensional representation of the electropneumatic manipulator. In addition to supervise online the variables of the process, the visualization of the manipulator evolution through a 3D virtual model can provide to the student a better information. The API Java 3D is going to be used to implement

this 3D virtual model. Java 3D is an optional package that can be downloaded separately from the Java standard distribution. It allows to develop applications and applets that show 3D graphics. This package has been used in other works to develop remote laboratories (Xiang *et al.*, 2001).

- Management of multiple users. Nowadays the developed system only allows that a user could access to the remote manipulator in each instant. The concurrency is not allowed, as it is logical. However, a FIFO system must be implemented to manage the users that are waiting to use the remote laboratory.
- Development of the automation laboratory web. This web must control the identified access of the student, and must contain:
 - Tutorials about the electropneumatic manipulator,
 - tutorials about PLC programming, and
 - collection of experimental exercises to be performed using the remote laboratory.

Furthermore the web must allow the delivery and automatic evaluation of the exercises performed by students, providing to students the evaluation result in real time. This web is being developed using PHP and it is explained in the next section.

4. WEB SYSTEM FOR REMOTE EDUCATION OF AUTOMATION

A web application powered by Apache centralizes the access to the different aspects of the automation remote laboratory, Fig. 5. This application is coded in PHP (a very flexible and powerful language to write web server applications) (PHP, 2004). This application is the responsible of the resources access, the common user interface and the automatic evaluation of the student.

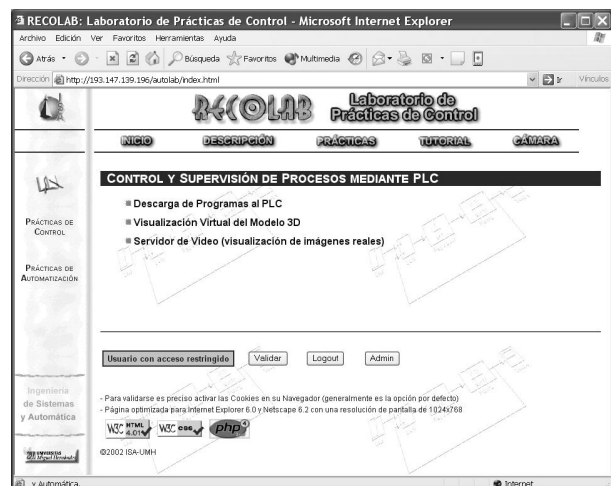


Fig. 5. Web application for automation education.

The site shares a common interface with other laboratories for control education developed in our department included in the so called RECOLAB system (REmote Control LABoratory)(Puerto *et al.*, 2001).

Most of RECOLAB site can be accessed from Internet without any restriction as far it is a open teaching tool that could be used by any student interested in an automation course. The most critical point in RECOLAB is the real time execution of program files in the PLC, acting over the physical system. In order to have a more efficient use of the laboratory resources this processing needs to be done under a validated access.

The used validation system is not based in an standard Apache server supported system, because these methods are focused in controlling the web resources, not the application resources. The validation system is coded in PHP so it permits to be integrated with the main application. Options rendered by the PHP code are based on the validation done by the user. In this way there is no need to maintain different PHP code for validated or no validated users. If an user requests an option that needs validation, the PHP code itself request for user validation.

In order to keep security, any validated access is made under a SSL connection using the `mod_ssl` module in the Apache server. To store the validation data, the application can be configure to use cookies o sessions. The validation process is based in a user name and a password. The control access to the resources is based on a user name and a priority level. Most of the resources in RECOLAB use an access control by level, being similar to the group access used in other validation systems.

Also, a management application has been developed in order to administrate the users and to keep the access logs. The access to this application is controlled by the same validation system.

At the moment the automatic evaluation system is been developed. The automatic evaluation system will use a data base of tests related with the practical exercises that can be done with the system. The students will fill a form with the responses to the questions for each exercise and the system will evaluates how well the exercise has been performed. The evaluation system will coded in PHP and MySQL will be used as data base engine (MySQL, 2004).

5. CONCLUSIONS

In this paper, the new advances in the development of a remote laboratory for control and

automation courses at the Miguel Hernández University (Spain) have been presented. The remote laboratory will allow that students do not need to follow any rigid schedule, and it will allow also a flexible learning and a fast self-evaluation.

Nowadays a client/server architecture that allows to supervise and control a electropneumatic manipulator through Internet has been developed. The advantage of the architecture implemented is that minimal requirements are necessary to use the remote laboratory: a PC, Internet, a text editor, a browser web and the Java plug-in.

The system will be used in teaching at our university when the management of multiple users and the automatic evaluation system will be finished.

REFERENCES

- Apache (2004). The apache software foundation. <http://www.apache.org>.
- Java (2004). Java technology. <http://java.sun.com>.
- Ko, C.C., B.M. Chen, J. Chen, Y. Zhuang and K.C. Tan (2001). Development of a web-based laboratory for control experiments on a coupled tank apparatus. *IEEE Transactions on Education* **44**(1), 76–86.
- MySQL (2004). Mysql. <http://www.mysql.org>.
- Paya, L., J.M. Azorin, N.M. Garcia, J.M. Sabater, C. Perez and R.P. Neco (2003). Process control and supervision through internet. *Proceedings of the International Conference on Engineering Education, Valencia, Spain*.
- PHP (2004). Hypertext preprocessor. <http://www.php.net>.
- Poindexter, S.E. and B.S. Heck (1999). Using the web in your courses: What can you do? what should you do?. *IEEE Control System* pp. 83–92.
- Puerto, R., O. Reinoso, R.P. Neco, N.M. Garcia and L.M. Jimenez (2001). Remote lab for control applications using matlab. *Internet Based Control Education 2001. A proceedings volume from the IFAC Workshop Madrid*.
- Schmid, C. (1999). Virtual laboratory for engineering education. *Proceedings of the 19th World Conference on Open Learning and Distance Education*.
- Siemens (2000). User manual. *SIMATIC Computing*.
- Xiang, X., C.D. Cheng, C.C. Ko, B.M. Chen and S.J. Lu (2001). Api for virtual laboratory instrument using java3d. *Proceedings of the 3rd International Conference on Control Theory and Applications, Pretoria, South Africa*.