Telematics applications for a
body feed-positioning station

POPESCU DORIN, VAVA SORIN,
MORARET ADRIAN, MANTA FLORIN, POPESCU LIVIA
Department of Mechatronics
University of Craiova
107 Blvd. Decebal, 200440, Craiova
ROMANIA

Abstract: - The aim of the work was to develop the hardware and software structures to be used in an e-teaching context. A web based application has been developed, which links a programmable logic controller (PLC) with a computer for programming via Internet. The main requirement of our application is to provide a friendly user interface that allows PLC application developers to control the devices from anywhere in the world via a thin client that does not need the installation of any additional software on the user side. After considering several alternatives, it has been opted for web based open-source technologies that considerably reduced the implementation costs while accomplishing the other project requirements. The user has possibility to see the environment of process station via images from a network video camera.

Key-Words: - telematics, web based application, programmable logic controller, open-source technologies.

1   Introduction

The high cost necessary to carry out experiments (for didactical purposes) with manipulators and robots in various environments led to the development of remote facilities where the physical system, sensors and operating environment can be at a great distance.

Some researchers achieved an analysis of the current challenges in Internet tele-programming and tried to find the possible solutions under the current environment, where there are some unsolved common problems associated with this new technology such as limited bandwidth and unreliable signaling [2], [6].

Remote laboratories are laboratory experiments that run remotely via a web interface. Usually, either the student can set some parameters on the web, then a software interface converts those parameters to a form that is accepted by the local computer running the experiment or he can tele-operate an equipment [7], [8], [15], [17], [18].

We proposed more, namely the student can achieve a program, which can run on the PLC. The student has thorough freedom to choose what experiment wants to do. The goal is remote programming of a PLC, which controls a process station for use in tele-education. The process station is controlled through PLC and Internet connection.

Web-based tele-robotic systems have only become available on the Internet in the last decade. In the beginning they utilized a CGI interface to access the robots [2], [18]. With the introduction of Java and its integration into web browsers, developers could create tele-operated systems that sustained an interactive link to the robot during its execution. Many new tele-robotic systems were created that gave the user much more control and provided functionality, which under CGI could have never been possible [1], [3], [4], [12].

The Internet lab technology offers the students the opportunity to work with sophisticated equipment, of the kind they are more likely to find in an industrial setting, and which may be too expensive for most faculties to purchase.

This paper gives a solution for tele-programming of a certain PLC. It tries to show an aspect of higher education and training based on Information Technology. The final purpose of the remote laboratory is to allow the student to take full control of the equipment, in order to fulfill the task required by the teacher.

2   PLC and process station

The FX2N Mitsubishi PLC is a micro-PLC and it has 16 digital inputs, 16 digital outputs, 235 counters and 256 timers, timed and programmed interrupts, internal PID, high-speed counter and 0.08 μs instruction time. It incorporates an RS-422 port for programming and an RS-485 port for communication with other PLCs or PCs [14], [16].

The GX Developer editing software for FX2N Mitsubishi PLC permits writing programs in statement list or ladder diagram. Our achieved application permits writing programs in statement list and it has a syntax checking system and a help menu about instructions of FX2N Mitsubishi PLC and about the way to write programs.

Process station (body feed-positioning station) belongs to Flexible Manufacturing System FMS-200.

Body feed-positioning station (Fig. 1a and Fig. 1b.) feeds in the body that is the support for the turning mechanism, and moves it to the pallet located in the transfer system. This operation begins when the pallet is opposite the station, held in a determined location by a stop on the transfer.

This station may be divided into a series of modules. Each sub-division has been made by considering it as a set of components, which performs a specific operation within the process completed at the station.

Body feed: The feeder which supplies the body is of the gravity feed type, in that the bodies are stored in a stack so that when the bottom one is removed, the next falls into place under the effect of its own weight and that of the ones above it. The body is extracted by a pneumatic cylinder that pushes against a pusher shaped to match the profile of the body.

Position verification: The body contains a housing in which the other components are fitted. This housing must always be facing upwards when the base is placed on the pallet. To check correct body orientation, a check is made by a cylinder that advances and inserts a cylindrical part in the body housing. If the body is inverted, this part cannot enter the housing, the cylinder cannot complete its stroke and the magnetic detector on the cylinder is not activated. A signal to this effect enters the PLC, resulting in an indication that the position of the body is incorrect.

Movement to transfer point: A cylinder with a pusher at its end is used to situate the body at the point from which it is loaded onto the pallet. The cylinder is rectangular, to prevent the pusher from turning.

Rejection of incorrect body: If the verification process shows the position of the body to be incorrect, a single-acting cylinder moves it towards a ramp, leaving the space clear for a new body.

Placing body on pallet: A two-axis manipulator is used to place the body on the pallet lying on the belt conveyor. Each axis carries a parallel rod cylinder. The terminal element is a vacuum holding platform with four suction cups to absorb possible lack of alignment in height. Suction is created by a vacuum ejector fitted with a vacuum switch that sends a signal to the PLC indicating that the part has been correctly clamped.

Fig.1a.

Fig.1b.

3   Telematics applications

The goal of this project in the remote programming area is to discover and develop the system by combining network technology with capabilities of PLCs and proces station. The use of Internet technology for remote programming application offers the advantage of low-cost deployment. There is no longer a requirement for expensive purpose built equipment at each operator's location. Almost every computer connected to the Internet can be used to control a tele-operable device. The downside is the limitation of varying bandwidth and unpredictable time delays. These Internet features should be defined and considered before designing an efficient remote programming system. Besides that, several functional requirements should also be defined before designing any tele-operable system [5].

This web-based application allows users to perform own experiments remotely from another computer. Using a standard web browser and a connection to the Internet, the user can write his program for the PLC.

After this, the user uploads his program and the results can be viewed on screen. The experiments have video streams available for visual monitoring.

3.1 Performance requirements and technologies used

The main requirement of this application is to provide a friendly user interface that allows PLC application developers to control the devices from anywhere in the world via a thin client that does not need the installation of any additional software on the user side. Another requirement is related to the costs that are involved during the project development (e.g. development tools, hardware etc.).

After considering several alternatives, it has been opted for web based open-source technologies that considerably reduced the implementation costs while accomplishing the other project requirements.

Thus, Java were chosen for the application development [9], [10], [11], [13]. In order to speed up the development process and eliminate software bugs, additional open source libraries and APIs have also been used. The role of each one within the whole system will be explained in the next paragraphs.

3.2 Application architecture

The general system structure is described in Fig. 2.
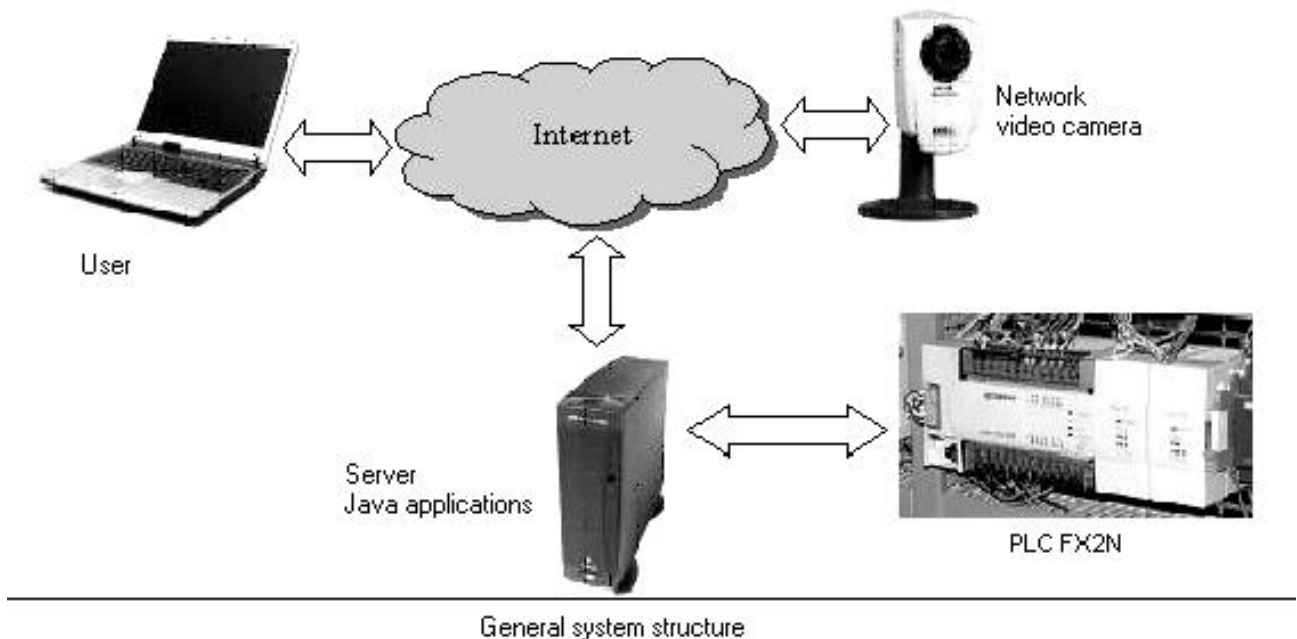

General system structure

Fig. 2.

The only software needed on the user side is a conventional web browser that can run JavaScript scripts and Java applets. This application has been tested and optimized to work on the most popular web browsers including Internet Explorer and FireFox, while other ones are currently being tested.

The Java server application is responsible for the application administration module, that involves users management, communication with PLC, PLC program editor etc.

An AXIS web camera is used for sending live video of the process to the user's browser. This camera model includes a built-in WEB server running under Embedded Linux that transmits the video stream either via an ActiveX control or a Java applet. While the ActiveX transmission method gives better results, the

Java applet is a good choice for browsers that don't support ActiveX technology (such as Opera or Konqueror).

3.3 Implementation details

Application is achieved using Java and contain two modules: the client and the server. The client application is an applet hosted by website department. The server application is hosted by a computer connected to Mitsubishi PLC via RS232 connection. The server application uses COMM API for connection with PLC.

3.3.1 The client

The client GUI is presented in Fig. 3.
The client application offers buttons and an editor window for instruction list.
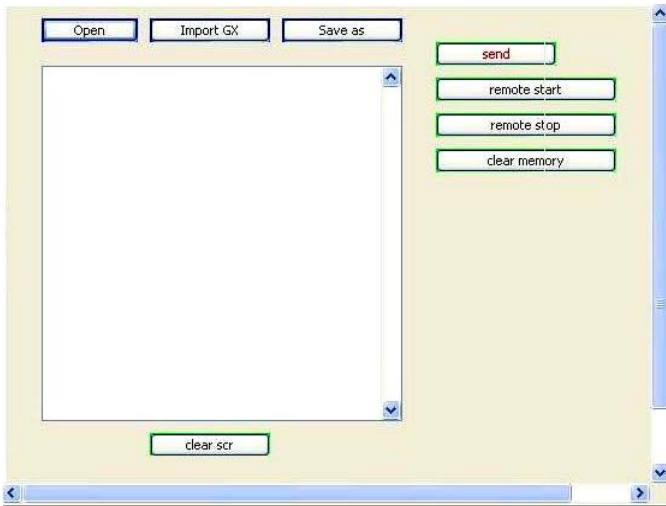
Fig. 3.

Fig. 4.

The buttons are used for next functions:

- **Open**: to open a programm which it was achieved using editor window of applet and saved through Save as button
- **Save as**: to save an instruction list programm edited in the editor window as a text file
- **Import GX**: because the GX Developer editing software for FX2N Mitsubishi PLC permits writing programs in statement list, it was creted the facility to import programms edited using GX Developer
- **Send**: after the programm was written in the editor window or open or imported, it can be uploaded on the server using thie button; the server will translate the programm fom text to machine-code instructions and then it will sent to Mitsubishi PLC
- **Remote start**: the user can change the PLC status, for example he can put the PLC in RUN status
- **Remote stop**: the user can change the PLC status, for example he can put the PLC in STOP status and the execution of PLC programm is stoped
- **Clear memory**: to delete the PLC memory before to upload a new programm
- **Clear scr**: to clear the editor window

The editor window permits to write the Mitsubishi PLC programm as instruction list. A syntax checking system was developed. If the user write a wrong instruction mnemonic or a wrong variable, he will receive a error message. An example is presented in Fig. 4.
The error message window indicates the number of line where it is the error and the wrong instruction mnemonic or the wrong variable. In addition, in the editor window the wrong instruction mnemonic or the wrong variable will be marked by red colour. In this way it will be more easier for user to correct the programm.
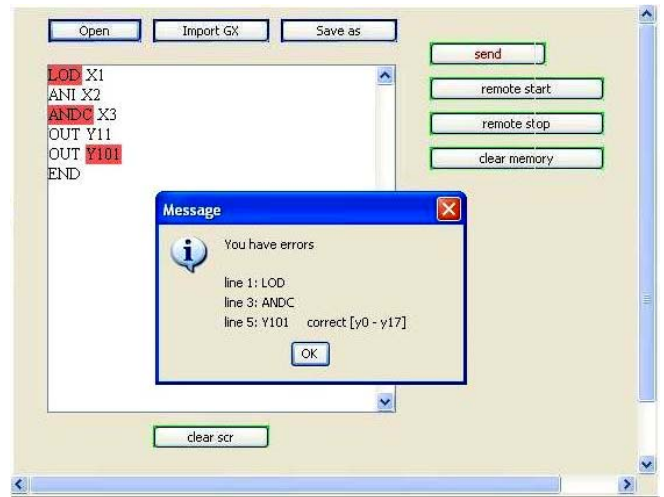
3.3.2 The server
The Java server application is responsible for the application administration module, that involves users management, communication with PLC, PLC status, etc. Only one user can be connected to server application (it means to Mitsubishi PLC).
The server translates the programm from text to machine-code instructions and then it will send the programm to Mitsubishi PLC at the transmision rate of 9600 bps.
The server has a window (fig. 5) that permits to see the uploaded user programm and the user command (Remote start, Remote stop or Clear memory).

Fig. 5.

4   Operating instructions
A user with a computer and Internet has to work in the following way after he navigate through our website: he has to ask permission to work with our PLC; he will receive an ID and a password; he will connect to our server and he can read or download our

documentation about system (PLC - process station); he has to answer to some questions and then he can write a user program with the original language of the Mitsubishi PLC; the application points out the syntactic errors; he can send his user program. The user program for PLC will be run and the user can observe the movements and operation of the manipulator by means of a network video camera (Fig. 6).

Fig. 6.

The main learning goals for users are: learning of the structure of Mitsubishi PLC; learning of Mitsubishi PLC statements; learning of the way to implement a flowchart or a ladder diagram by statement list; learning to achieve a program for Mitsubishi PLC in order to control a process station.

To perform an experiment with our system the user has to read the documentation about system and PLC, because before the remote experiment he has to answer some questions. Only if the user answers correctly he can access the user interface, writes a program and run it.

5 Conclusion

This project is currently being used at University of Craiova (in the PLC class). Such remote laboratory experiment method enables the student to use laboratory equipment, which is not usually available to the students.

The application has been tested inside the faculty where a computer has been connected to Internet that the user had used to program the PLC in order to control the process station which was in another laboratory.

Students can watch the process station via the application. For the majority of remote experiments

via Internet in mechatronics, the student can teleoperate a robot or set some parameters on the web and then he can watch the robot movements. We performed more, namely the student can write a program, which can run on the PLC in order to control the process station (that includes manipulators) as he wants.

The application is provided with friendly GUI. The exercises help students to understand and compare different process control methods using a PLC.

References:
[1] S. Amin, M. Zakaria, N. Majid, L. Siong, L. Horvath and J. Tar, Internet-based telerobotics: UTM's experince and future direction, *The 10th International Conference on Advanced Robotics, ICAR 2001*, Budapest, Hungary, 2001, pp. 313-319.

[2] P. Backes, K Tso, J. Norris, G. Tharp, J. Slostad, R. Bonitz and K. All, Internet based operations for the Mars polar lander mission, *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, 2000, pp. 2025-2032.

[3] D. Popescu, I. Dinulescu, L. Popescu, Tele-programming of Programmable Logic Controllers via Internet, *WSEAS Trans. on Advances in Engineering Education*, 2, Vol.2, 2005, pp.118-126.

[4] I. Dinulescu, D. Popescu, G. Terejanu, Web based Telematics Application using Open-Source technologies, *12th Int. Symposium on Systems Theory, SINTES12*, Craiova, 2005, pp. 267-272.

[5] D. Popescu, G. Terejanu, I. Dinulescu, M. Leoveanu, PLC based Telemanipulation via Internet, 14th *Int. Workshop on Robotics in Alpe-Adria-Danube Region*, Bucharest, 2005, pp.441-447.

[6] G. Belmonte Bermudez and M.A. Perez Sanchez, Robots Tele-programming, *1st Workshop on Robotics Education and Training, RET 2001*, Weingarten, Germany, 2001, pp. 19-24.

[7] R. Berntzen, J.O. Strandman, T. Fjeldly and M. Shur, Advanced Solutions for Performing Real Experiments over the Internet, *Int. Conference on Engineering Education*, Oslo, 2001, pp. 21-26.

[8] H. Ewald and G.F. Page, Performing Experiments by Remote Control Using the Internet, *Global J. of Engineering. Education*, Vol. 4, No. 3, 2000.

[9] D. Flanagan, *Java Script – The Definitive Guide*, O'Reilly, 2001.

[10] M. Hall, *Core Servlets and Java Server Pages*, Prentice Hall, 2000.

[11] B. Kurniawan, *Java for the Web with Servlets, JSP, and EJB: A Developer's Guide to J2EE Solutions*, New Riders Publishing, 2002.

[12] R. Marin and P. Sanz (2001) The UJI Telerobotic Training System, *1ˢᵗ Workshop on Robotics Education and Training*, RET 2001, Weingarten, Germany, pp. 25-30.

[13] B. McLaughlin, *Java and XML*, O'Reilly, 2000.

[14] D. Popescu, *Programmable logic controllers*, Ed. Sitech, 2001.

[15] D. Popescu and K. Schilling, TOM – A Remote Laboratory for Mobile Robots Education, *7ᵗʰ International Conference on Intelligent Engineering Systems*, Assiut, Egipt, 2003, pp. 447-453.

[16] ***, FX Programming Manual, revision D, April 2003.

[17] S. Grange, T. Fong and C. Baur, Effective vehicle teleoperation on the world wide web, *IEEE International Conference on Robotics & Automation*, 2007-2012, 2000.

[18] K. Schilling, H. Roth and R. Lieb, Teleoperations of rovers – from Mars to education, *Proceedings IEEE International Symposium on Industrial Electronics*, 1, 1997, 257-262.