REMOTE LABORATORY FOR PLC TRAINING

Rodrigo Pita Rolle*, Lucas Belório de Castro*, Larissa Agnes Pereira Ronqui*, Wagner Endo*, Marcos Banheti Rabello Vallim*

*UTFPR - Universidade Tecnológica Federal do Paraná - Câmpus Cornélio Procópio
Avenida Alberto Carazzai, 1640, 86300-000
Cornélio Procópio, PR, Brasil

Emails: rprolle@live.com, lucas.bc.lyb@gmail.com, larironqui@hotmail.com, wendo@utfpr.edu.br, mvallim@utfpr.edu.br

Abstract— This paper presents a web-based remote laboratory designed to provide experimental activities of automation and control of discrete event systems. The purpose of the project is to provide remotely through the Internet an experimentation interface that allows the practical application of concepts related to the automation of discrete systems and the training of the user in process controllers utilized in industrial plants. Complementarily, the platform has auxiliary documentation in order to provide the user the necessary information for the execution of the proposed activities. Finally, are discussed the main constructive and functional aspects of this platform and is also shown an usability analysis of the system.

Keywords— Discrete Event Systems, Discrete Control, Remote Laboratory, PLC Training.

1 Introduction

The remote operation of automation and control systems is a relevant functionality in the current technological context. It is possible because of the development of computing and Internet communication. In this context, there is also a large interest in using this kind of tools in the academic scope, especially for facilitating the access of the students to laboratory practices (Cooper, 2000).

There are other works, such as the Remote Laboratory for Learning of Energetic Efficiency in a Water Pumping System (Loureiro, 2009), in which a water pumping plant is controlled through different methods of flow adjustment. The purpose is to provide a learning platform where the user can compare the energetic efficiency of each adjustment strategy used in the experience, what can be done with a synoptic screen in which the system operator can vary operation parameters. Other related work is the Project of a Remote Laboratory for Automation of Industrial Processes (Martins, 2013), that presents a remote laboratory for automation of a pneumatic system using a Programmable Logic Controller (PLC). In this case, the user accesses the PLC and performs the configuration of the controller, verifying afterwards evolution of the system through a video stream provided for a monitoring webcam.

The purpose of this paper is to create a remote access platform for PLC training in a way that does not require from the user the need to configure IPs and network infrastructure – that is the major issue identified in the literature – and to emphasize the development of an efficient control strategy for the proposed problem, providing to the remote user the full software environment provided by the PLC manufacturer. In the developed platform, the user performs all the controller configuration procedure and development of the programming logic, with full access to resources such as error debugging, on-line Input/Output monitoring, PROFIBUS networks programming and others.

Another particular characteristic of this laboratory is the kind of process that is controlled. It is an automation process simulation software that interacts with the PLC through a data acquisition module. This device provides discrete voltage levels to the controller's Input/Output bus, what allows the PLC to interpret the system answers and, likewise, allows the software to interpret the control action imposed by the controller. Additionally, the simulator allows the user a manual operation option of actuators for functional learning and a fail simulation option, in which the user can cause fails in one or more sensors in order to develop fail-safe strategies.

The basic procedure for the user includes the system modelling with Petri Nets, that will ease the development of the PLC programming logic and also motivate the user to practice modelling techniques.

In general terms, the focus of this access platform is to provide the remote user the same resources available for the students that use the physical discrete systems learning laboratory, through a friendly interface that shows all the fundamental information for the execution of the proposed task and provides full access to additional simulation options.

2 Theoretical base

2.1 Control of discrete event systems

The basic concept to be explored with the proposed laboratory is the control of Discrete Event Systems (DES). In this kind of control, the sys-

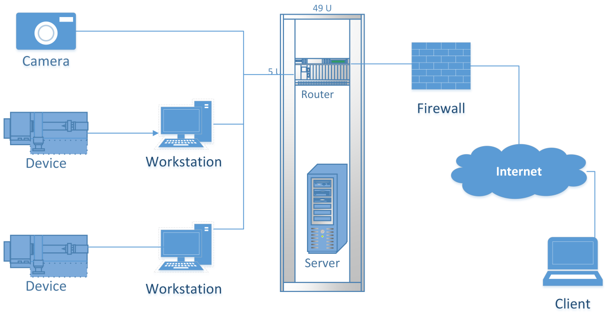

Figure 1: Block diagram of a remote laboratory.

tem variables can assume pre-estabilished values belonging to a discrete set. Its behavior is, therefore, time-independent and event-driven.

The term "event" is used to describe the abrupt and instantaneous occurrence of a phenomenon in the system (e.g. the trigger of a sensor). In each occurrence of an event, the system can adopt a new comportment or to execute a new function, namely the system can adopt a new "state" – which is equivalent to an "operation status": stopped, moving, filling, discharging, among others – of one or more elements of a plant.

There are many mathematical formalism that attend to different classes of problems that can be formulated for DES. The major mathematical formalisms currently used are: Supervisory Control, Petri Nets, Dioid Algebra, Markov Chains, Queoing Theory and others.

2.1.1 Interpreted Petri nets

Petri Nets is a DES formalism used for description and analysis of synchronization, communication and sharing among simultaneous processes. They are composed by two types of nodes, named "transitions" and "places". The places are symbolized by circles and are associated to conditions that may be satisfied so the transitions can happen, while the transitions have a bar as representation and are normally associated to events. The connection of these two nodes is made with arcs, that can start from a place to a transition or from a transition to a place (Cassandras and Lafortune, 2009).

Interpreted Petri Nets (IPNs) are non-autonomous Petri Nets that have synchronization with external events and/or timing. These nets own this denomination because of the diversified applications or interpretations that they can have, according to the area in which is intended to use them and their adaptation to these areas (David and Alla, 2010).

2.2 Remote laboratories

Remote laboratories are systems constituted of a physical plant composed of real equipments, that are operated remotely through a graphical interface using the Internet as communication mean. By definition, it consists in a modality of experiment driven and controlled remotely. The remote laboratories are more suited to handle a bigger number of users, especially in situations of small physical installations or limited in availability of experiments (Chen et al., 2010).

The block diagram of a remote laboratory is shown in Figure 1, adapted from Chen et al. (2010). The function of the server is to handle the information flow, such as scheduling, log management, connection requests, among others. It also may be ready to work as a webserver for the pages that will be made available to the user. The workstations are utilized to execute user's requirements and to control the laboratory devices during the realization of the experiences. It is also necessary to provide real-time monitoring of the process through a camera or similar way.

2.3 Programmable logic controllers

The element of control chose for this platform and usually applied to control DES is the PLC. This machine is a microprocessor-based controller that uses a programmable memory to store instructions and to implement functions such as logics, sequencing, counting, timing and arithmetical operations in order to control processes and machines. The work principle of this equipment are electrical I/O signals, that can be either analog or digital. Its development is focused on providing a viable operation for engineers that can have little knowledge of computing science and programming languages, aiming for a quick learning of its operation procedures (Bolton, 2009).

2.4 Programming languages

The IEC61131-3 standard, one of the major reference models for programmable electronic systems, establishes five programming languages for PLCs: Ladder (LD), Function Block Diagrams (FBD), Sequential Function Charts (SFC), Instruction Lists (IL) and Structured Text (ST). The three first are graphical languages, while the last two are textual ones.

The LD consists basically in "contacts" and "coils", that are similar to old hardwired relay circuits. According to Molina et al. (2007), seventy five percent of PLC programs utilize this programming logic. The IL standard is a similar approach to Assembler languages and is a good alternative to intense data processing. The ST language is similar to Pascal, it is especially useful to solve mathematical problems. The languages FBD and SFC are both useful to structure the main program, that is defined as a logical compilation of elements of programming languages (Molina et al., 2007).

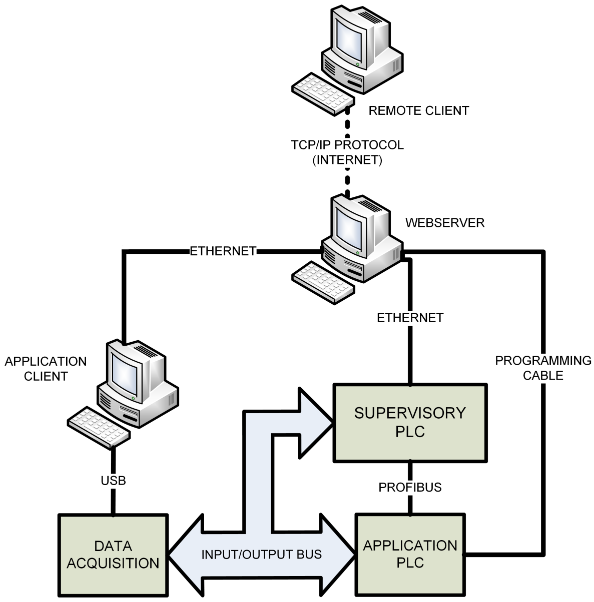

Figure 2: Remote laboratory architecture.

This last one was initially a french standard called “Grafcet”, a graphical language for modelling and control of DES established by IEC 60848 standard and included in IEC 848 standard as “SFC” (Roussel and Lesage, 1996). SFC is applied to solve problems whose command actions are sequential or time-dependent, problems in which the solution based in Ladder logic could be confusing or even not viable (David, 1995). The SFC graphical representation consists of “steps” represented by squares and “Transitions” represented by horizontal bars. The steps and transitions are interconnected by arcs.

3 Development

3.1 Remote access tool

In order to provide the user the unrestricted access to the PLC programming environment, aiming to offer remotely the same resources available for the user situated at the physical laboratory, it was chosen the VNC (acronym for Virtual Network Computing). It is a system that allows a remote user to operate a computer in a different location through a network. This is a communication strategy based in the concept of remote frame buffer, that consists basically of the server computer to receive input events from the client and to return compressed screenshots to this one. (Haraikawa et al., 2001).

Aiming to simplify the remote access, it was utilized an Java application to perform the connection to the remote desktops. Thus, the VNC server is installed in the laboratory computers and the remote user has access to the application, that is already configured and made available in the web interface.

Figure 3: Computational process simulator.

3.2 Remote laboratory development

The work platform developed was based in the architecture shown in Figure 2. It can be interpreted as a simplification of the structure presented in Figure 1. Since this system may host only one process, it does not require such a complex network infrastructure. It is notable that the computer shown as Webserver executes by itself the functions of host of web pages and Workstation that is utilized to configure the Application PLC. It is used a Supervisory PLC with Ethernet interface and an Application PLC that is configured by the remote user.

Major PLC brands as Siemens do not provide programming software for free, so it is not possible to make its distribution to the users. Thus, the proposed solution to allow the user to work directly in the PLC programming even without owning the software is the remote access to the server located in the physical laboratory, which owns a licensed copy of the PLC programming suite. So, the user has full access to the resources of the suite even without having it installed in his computer.

It is used ITS PLC Professional Edition, a computational simulator especially developed for discrete control training, shown in Figure 3, that is interconnected to the PLC through a Data Acquisition hardware (DAQ). In this way, the communication PLC-Process is performed by electrical signals, exactly as it would occur in a real industrial process (CURZEL and LÜDERS, n.d.). This program owns five different plants and can be used with any PLC. Once the process evolution happens in the virtual environment, it is not necessary to use an independent device to capture images for remote monitoring. It was chosen to use the application client computer integrated to the network in order to provide desktop sharing, in which the simulator shows the application plant in real time.

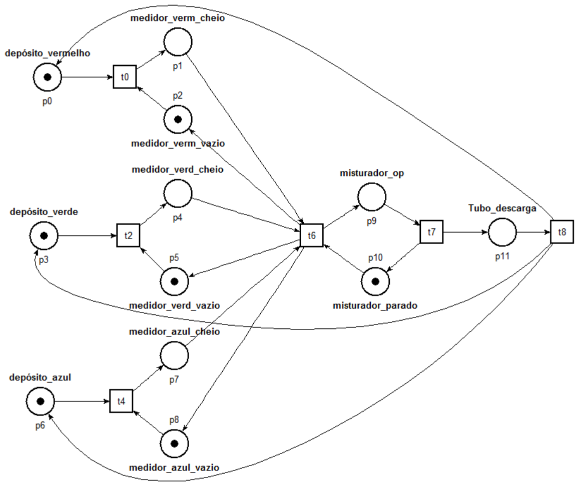

Figure 4: Interpreted Petri Net for white ink production.

The central data flow processing between the internal network and the Internet is performed by the central Webserver available at the University. Although the pages of the website are not incorporated in this server, it works as a bridge between the internal network and the Internet, warranting accessibility for the platform, providing security conditions and also controlling the data flow.

3.2.1 Remote access structure

In order to provide the remote access to the laboratory, it was developed a Web interface that allows the access to applications in a friendly environment. All necessary information for the task development is available in this site, as well as access structures to programming environment and process monitoring. The developed structure requires the remote user just Internet access with a browser and updated Java Runtime Environment.

By accessing the programming environment, the remote user is capable of performing the configuration and programming steps of the application PLC, inserting the strategy created with the DES control concepts in the programming languages available in the controller's programming suite.

Once this stage is done, the user may access the viewing area. In this environment, there are four virtual buttons that allow the user-process interaction, according to the function the buttons are programmed to execute in the control logic.

3.2.2 Learning tutorials

In order to accelerate the adaptation of the users to the experimentation environment, there were developed tutorials that include instructions about the virtual plant and the application PLC/programming suite.

These tutorials present the necessary information for the experiment conduction in a summarized way, also introducing the user in an environment in which are used the standarts of automation plants documentation, such as Piping and Instrumentation Diagrams, Input/Output Devices List and PLC Input/Output Addressing Lists.

3.2.3 Platform validation

In order to demonstrate the utilizing of the remote laboratory, it was conceived an experimentation scenario in which are approached the concepts of DES modelling with Petri Nets and the PLC programming in SFC language. The basic procedure, prescribed in five stages, is the following:

1. To access the remote laboratory available in the address `http://www.lremotosed.cp.utfpr.edu.br` and to read the instructions;
2. To analyse the simulation plant and to verify the proposed objective;
3. To model the plant using IPN;
4. To convert the IPN model to SFC;
5. To execute the control of the plant through the PLC and to check the evolution of the system in the simulation software.

The simulation scenario chosen for this demonstration was the ink dosage plant named "Batching", that simulates the mixing of three ink colors (red, green and blue) in order to obtain a batch of ink with desired color. The objective is to model the system and to develop a control strategy that results in the production of a batch of white ink.

Based in the observation of the presented system and in the defined objectives, the user may model the plant and to start the construction of the control logic. The modelling can be made using the software TINA (Time Petri Net Analyzer) (Berthomieu* et al., 2004), available for free download to the user and that provides important analysis tools for the verification of essential characteristics of the developed model. An example of IPN model for this system is presented in Figure 4.

Since the model is obtained as an IPN, it is necessary to convert it to SFC and to execute the PLC programming using the toolsuite of its configuration software. An example of SFC diagram for the control of the production of a batch of white ink is shown in Figure 5.

After the implementation of the SFC structure, it is necessary to download the program to the PLC, that will execute the control of the simulated industrial process. The user may verify the evolution of the states of the system through

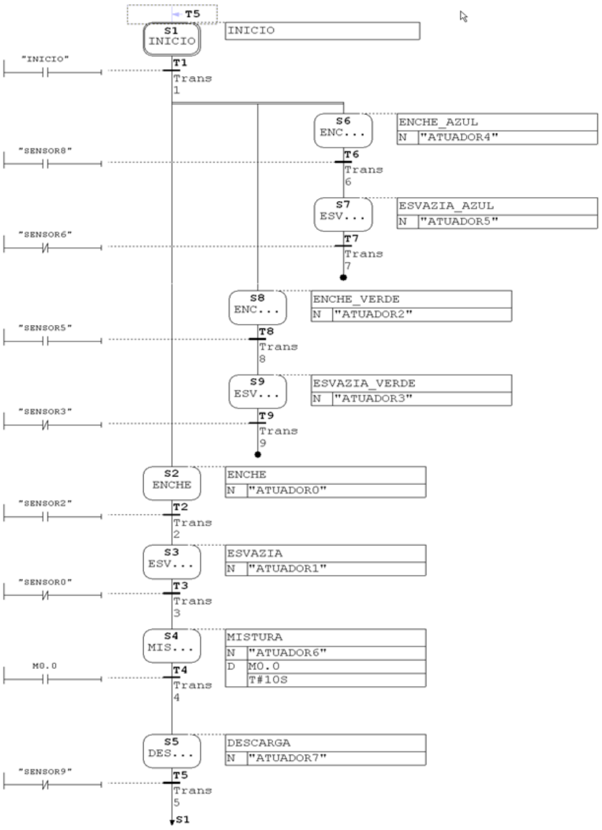

Figure 5: White ink production SFC diagram.

the simulation software, as shown in Figure 6. In this area are arranged the virtual buttons that allow the user-process interaction. According to the programming strategy developed, the user can use these buttons to do actions as "start", "stop" or enter an "emergency mode".

3.2.4 Usability evaluation

In order to evaluate the usability of the developed laboratory, ten Control and Automation Engineering students of the Federal Technological University of Paraná were invited to interact with the platform. It was developed a questionnaire based on the software product usability feature

Figure 6: System operation viewing and virtual buttons on the left side.

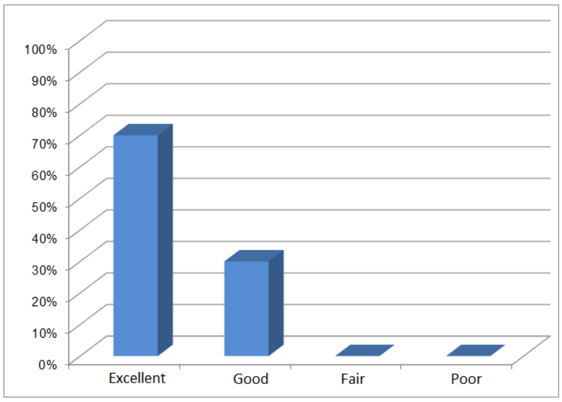

Figure 7: Users' evaluation result.

of the ISO / IEC 9126-1 standard. The usability subcharacteristic is the ability of the software product to be understood, learned, operated and attractive to the user. Each subcharacteristic was addressed in the questionnaire with a question, thus evaluating the ability of the product.

The last part of the questionnaire led to a personal participant's evaluation in relation to the remote laboratory. The obtained results indicate that the platform was approved by the users given the fact that all respondents classified it as "excellent" "or "good", which indicates that, from the user's point of view, the platform was capable to fulfil the proposed objectives. Figure 7 shows a graph illustrating the results obtained.

4 Results and discussions

The developed platform, beyond the approached functional aspects, it also shows implementation flexibility. There is no need for a specific PLC brand or model, as long as the supervisory PLC dispose of an Ethernet interface and the application PLC is able to recognize the desired programming language (in this paper it was used SFC, but any other language can be used according to the managers interests). The computers do not require adaptations or additional accessories and the platform can be expanded as much as required. It is also noticeable that the system can be utilized in LAN (Local Area Network) in situations where it is not desired to make the system available on the Internet.

The remote user can access all software features provided by the PLC manufacturer, including advanced configuration tools, status monitoring of inputs and outputs, debugging and help files. Thus it was overcame the difficulty of providing the remote user unrestricted access to equipment resources without the additional expense of providing programs with paid licenses for programming at distance. The tutorials are presented in standard forms of documents of industrial automation plants. Based on these documents it is also possible to emphasize the documentation of automation and instrumentation projects.

5 Conclusions

The remote laboratory platform presented in this paper enables the execution of PLC training activities, focusing on the control of DES. There is a remote access interface and an extensible architecture that can also be adapted to the specific needs of managers. It is possible to perform various types of activities according to the availability of simulation scenarios, as well as acquainting the remote user with the PLC programming environments utilized in automation plants, exercising its design capabilities of logic control and manipulation of industrial machinery.

The website structure facilitates the provision of informational content, manuals, tutorials, examples, and any other materials that contribute to the learning of the covered concepts and operation of industrial controllers. It also makes it simple to provide cameras in cases where it is taken the choice of a real experiment. The possibility of choosing PLCs also facilitates the implementation of the platform in places where there is already some equipment in use or it will be made purchase choices according to the financial conditions.

Finally, the flexibility in the simulation scenarios provides this platform larger reach. It is possible to use this structure either for basic testing or for complex applications, as long as it exists a simulation plant, either virtual or real, that corresponds to the intentions of the manager. The data obtained through the usability analysis indicate that the platform presents positive outcomes in the user's point of view.

It is noticeable that the network infrastructure and access the management procedures need to be improved. It would be necessary to develop a database for logs, user registering, scheduling and to pay for professional webhosting service in order to guarantee greater system robustness and safety. However, it is clear that the developed prototype is actually able to contribute to the development of professional skills of its users.

Thereby are fulfilled the objectives of providing a remote experimentation interface that encompasses the parameterization of real industrial controllers, implementation of control logics, operation checking in order to verify the operability of the developed strategy, as well as the availability of instructional didactic materials for experiment execution.

Acknowledgements

The authors would like to thank Federal Technological University of Paraná for the support during the development of this project.

References

Berthomieu*, B., Ribet, P.-O. and Vernadat, F. (2004). The tool tina–construction of abstract state spaces for petri nets and time petri nets, *International Journal of Production Research* **42**(14): 2741–2756.

Bolton, W. (2009). *Programmable logic controllers*, Elsevier.

Cassandras, C. G. and Lafortune, S. (2009). *Introduction to discrete event systems*, Springer Science & Business Media.

Chen, X., Song, G. and Zhang, Y. (2010). Virtual and remote laboratory development: A review, *Proceedings of Earth and Space* **55**: 3843–3852.

Cooper, M. (2000). The challenge of practical work in an euniversity-real, virtual and remote experiments, *Proc. IST2000 The Information Society for All, Nice, France*, pp. 6–8.

CURZEL, J. L. and LÜDERS, R. (n.d.). Modelagem e implementação em fpga do controle discreto de uma planta didática de separação de itens.

David, R. (1995). Grafcet: A powerful tool for specification of logic controllers, *Control Systems Technology, IEEE Transactions on* **3**(3): 253–268.

David, R. and Alla, H. (2010). *Discrete, continuous, and hybrid Petri nets*, Springer Science & Business Media.

Haraikawa, T., Sakamoto, T., Hase, T., Mizuno, T. and Togashi, A. (2001). μvnc: a proposal for internet connectivity and interconnectivity of home appliances based on remote display framework, *IEEE Transactions on Consumer Electronics* **47**(3): 512–519.

Loureiro, I. B. (2009). Laboratório remoto para ensino de eficiência energética em um sistema de bombeamento de água.

Martins, L. M. S. (2013). *Projecto dum laboratório remoto para automação de processos industriais*, PhD thesis, Instituto Superior de Engenharia De Lisboa.

Molina, F., Barbancho, J., Leon, C., Molina, A. and Gomez, A. (2007). Using industrial standards on plc programming learning, *Control & Automation, 2007. MED'07. Mediterranean Conference on*, IEEE, pp. 1–6.

Roussel, J.-M. and Lesage, J.-J. (1996). Validation and verification of grafcets using state machine, *IMACS-IEEE" CESA'96"*, pp. pp–758.