PLC Training Based on a 3D Virtual Maquette Control: An Educational Experience in Automation

José L. Lima[1] and José C. Gonçalves[2]

Abstract **- A virtual automation laboratory is usually used as a stand-alone application aiming an educational experience of process simulation and control where students have no contact with hardware. A better way to let students practice in a real PLC (Programmable Logic Controller) achieving the automation class programming skills is to use a maquette. This paper describes a PC application, behaving as a Virtual Maquette that emulates a three-dimensional real world where an external PLC controls its behavior receiving feedback from the plant. The developed interface connects the PLC to the PC through the parallel port. The Virtual Maquette is composed by a car parking where user requests gate open order while PLC supervises gate open/close command, traffic lights and night lighting. Cars movements captivate student's attention and increases motivation to learn. The Virtual Maquette supports different ways of control allowing several simultaneous student groups to work in different solutions.**

Index Terms – Control, Education, Virtual Laboratories.

INTRODUCTION

The Bologna's declaration introduces, in the European Union Education Space, a significative change in the learning process, by changing the focus from a paradigm oriented to the transmission to a paradigm oriented to the learning process. However, it is necessary to go further by introducing new tools that motivate and involve teachers and students in the learning process [1]. Individual learning is the basic concept and practice is nowadays a common answer to industry requirements [2].

One of automation lessons skills is to prepare students to develop PLC (Programmable Logic Controller) programs. The real PLC applications are applied in several industrial designs [3]. The student PLC programs can usually be tested in laboratories resorting to maquettes. A classical 2D maquette connects to the PLC through wires and basically contains several switches and lights. A 3D maquette is easier to perceive but harder to control and prototype. Objects movements that emphasize the maquette and captivate the student's attention are a difficult task. Even more, usually there are several groups, forcing the use of more than one maquette at the same time. An example of a 3D car parking maquette is presented in Figure 1 where there are some cars, indicator lights and barred gates with its controller [4].

FIGURE 1
3D REAL MAQUETTE – CAR PARKING

A Virtual Maquette is a tridimensional reconstruction of a model (industrial parts, buildings, cars , etc.) inside a computer containing realistic textures, lightning, different views stretching the distance from virtual to real [5].

The developed Virtual Maquette is composed by a PC application, reducing some problems and allowing an attractive visualization [6] The developed car parking maquette scene is presented in Figure 2.
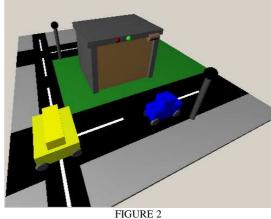

FIGURE 2
DEVELOPED VIRTUAL MAQUETTE

This paper is organized as follows: Initially, the Virtual Maquette and its world behavior and hardware interface are described. In the next section, the PLC programming issues where a first approach is presented and a sample solution is

---

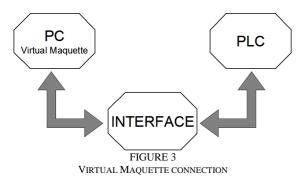[1] José L. Lima, Polytechnic Institute of Bragança, jllima@ipb.pt
[2] José C. Gonçalves, Polytechnic Institute of Bragança, goncalves@ipb.pt

shown. Finally, last section rounds up with conclusions and future work.

VIRTUAL MAQUETTE DESCRIPTION

The developed Virtual Maquette gathers the advantages of a classical maquette and some extra features supporting the students to test their work. The disadvantages presented in a 3D classical maquette such as hardness to prototype and the hard control due to some hardware limitations are not presented in a virtual 3D Maquette. It is also a low cost solution because the software can be easily changed to different requisites and distributed to several work groups. These advantages are not presented in solutions based only on hardware.

The developed Virtual Maquette is composed by three cars, two of them are small cars and another is a big one, a garage, a road, traffic lights, artificial and day lightning providing a more realistic environment than a conventional maquette. The 3D scene, the zoom and the move around the world feature emphasizes the appearance [7]. It is desired to control the gate command (open and close), traffic lights and street illumination by gathering information from the environment. The closed loop is achieved resorting to an external PLC (where students develop the Ladder Program) connected through the PC. The interface joints the PLC and the PC through the parallel port as presented in Figure 3.

FIGURE 3
VIRTUAL MAQUETTE CONNECTION

The real developed interface connected to the PLC is presented in Figure 4. The PLC (at the left) has onboard inputs and outputs where the black wires are connected. Otherwise, an external Input/Output card should be attached. On the right side, the interface connects the PLC to the PC parallel port.
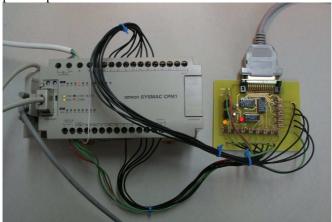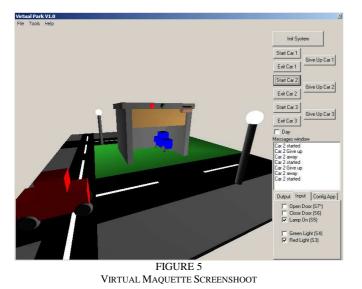

FIGURE 4

PLC – PC INTERFACE

The Virtual Maquette provides to the PLC (CPM1 from Omron [8]) the boolean status of the eight environment variables as presented in Table I. The PLC Addresses with its symbolic names and the corresponding PC Parallel Port bits are also presented. Symbolic names allow programmers to decode the developed software easily as illustrated in the *subsection II* of *PLC Programming Example*.

TABLE I
VIRTUAL MAQUETTE OUTPUTS MAP

Virtual Maquette Status	PLC Address	PC Parallel Port bit	Symbolic Name
Big car open door request	0.0	D0	BCODR
Small car open door request	0.1	D1	SCODR
Gate Opened	0.2	D2	SGO
Gate Closed	0.3	D3	SGC
Sunlight presence	0.4	D4	Sunlight_P
Small car 1 parked	0.5	D5	SC1P
Small car 2 parked	0.6	D6	SC2P
Big car parked	0.7	D7	BCP

The PLC must control the Boolean status of the five actuators in the Virtual Maquette as presented in Table II. The PLC Addresses with its symbolic names and the corresponding PC Parallel Port bits are also presented.

TABLE II
VIRTUAL MAQUETTE INPUTS MAP

Virtual Maquette Actuator	PLC Address	PC Parallel Port bit	Symbolic Name
Red traffic light	10.3	S3	R
Green traffic light	10.4	S4	G
Street illumination switch	10.5	S5	Street_light
Close door command	10.6	S6	GC
Open door command	10.7	S7	GO

The Virtual Maquette application software is presented in Figure 5. It was developed in Delphi language resorting to GLScene, an OpenGL 3D library for Delphi. It provides visual components and objects allowing description and rendering of 3D scenes [9].

FIGURE 5
VIRTUAL MAQUETTE SCREENSHOOT
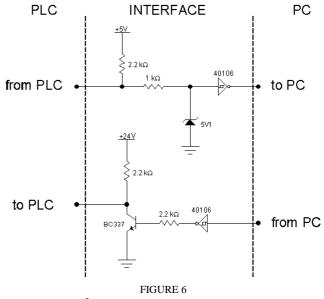
I. World Behavior

The Virtual Maquette control must obey to some requisites. The garage capacity allows either two small cars or just a big one. The red traffic light must be turned on in the following circumstances: A small car is parked and the big one requests the open door command or the big car is parked and a small one requests the open door command. Otherwise, the green traffic light should be turned on when a car can be parked and the gate should be opened. The gate status is closed by default. If a car is getting in or leaving the park, the gate must be completely opened. When the gate is opened, the open command should be turned off avoiding motor damage, just as well in closing operation. The street light must be turned on at nightfall.

The cars movement is autonomous. User starts-up the desired car, which moves until the garage entrance and remains static waiting for the gate to be opened. Once opened, the car gets into the garage with an autonomous movement too. Otherwise, user can cancel the parking operation guiding the car to move away.

These actions must be provided by the PLC, based on the developed software as a challenge student's task. If a PLC program malfunction, the Virtual Maquette remains still and presents a visual warning informing the fault to the student.

II. Hardware Interface

The basic objective of this interface is to adapt two different Voltage levels. On the one hand, the PLC standard Voltages are 24V. On the other hand, the PC parallel port Voltages are 5V. It is basically composed by digital logic circuitry and voltage limiters as presented in Figure 6.
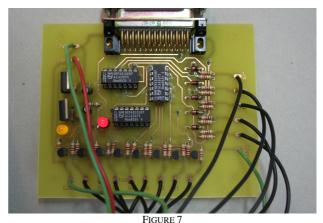

FIGURE 6
INTERFACE ELECTRICAL SCHEMATIC

The PC parallel port is protected by a 5V1 Zenner diode. Then, a buffer gate is applied allowing by this way the conversion of 0V and 24V inputs (from the PLC) to 0V and 5V outputs to be applied in the status address of the PC parallel port [10].

The PC parallel port output data address bits connect to a buffer to achieve the base current of the NPN transistor, working as an open-collector switch. In both situations a pull-up resistance is used. On the one hand, for the interface output, the pull-up resistance places 24V to the PLC when the transistor is switched off and 0 V when in saturation. On the other hand, a pull-up resistance defines the default state supporting the PLC relay and open collector types [11]. The impedance adaptation is guaranteed applying a NOT gate for both input and output signals.

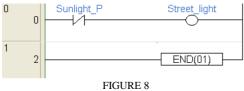The developed interface circuitry is shown in Figure 7.

FIGURE 7
INTERFACE PRINTED CIRCUIT BOARD

PLC PROGRAMMING EXAMPLE

I. First PLC Programming Approach

Ladder logic is a method of drawing electrical logic schematics. It was originally invented to describe logic made

from relays. Ladder logic is widely used to program PLCs, where sequential control of a process or manufacturing operation is required. Ladder logic is useful for simple but critical control systems, or for reworking old hardwired relay circuits. As programmable logic controllers became more sophisticated, it has also been used in very complex automation systems. An introductory example of a ladder logic program that controls one of the features of the Virtual Maquette is presented in Figure 8 [12].
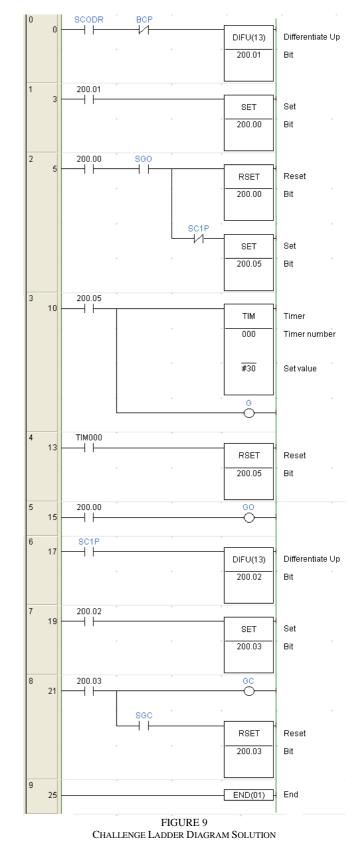

FIGURE 8
INTRODUCTORY LADDER PROGRAMMING EXAMPLE

If Sunlight presence (*Sunlight_P*) is detected then the Street light (*Street_light*) is turned off. This simple Ladder program includes only two '*rungs*' – a Ladder logic line. Nowadays, in real PLC industrial applications, there may be hundreds of them.

II. Programming Challenge

The forward program, presented in Figure 9 and its brief following explanation based on its symbolic names, is an example of one small car parking request [13].

- The SCODR (small car open door request) is only attended if BCP (big car parked) is false.
- If SCODR is attended then the GO (gate open order) is activated.
- The gate is requested to open until it is completely open: SGO (sensor gate open). Then, the green traffic light (G) is turned on for 3 seconds.
- When the car is parked, the gate close request (GC) is activated until it is completely closed (SGC).

This simple program concerns only one small car and has different solutions for the same conduct. For three cars there must be a more complex logic that remains up to students.

FIGURE 9
CHALLENGE LADDER DIAGRAM SOLUTION

CONCLUSIONS AND FUTURE WORK

The developed system, composed by a PC application and a hardware interface, allows students to develop their programs in the PLC, interacting with the Virtual Maquette

while actions are displayed in the PC monitor, based on 3D realistic pictures. The cars movement, the zoom feature, the camera positioning freedom and the easy interact commands captivate students attention and increases their learning motivation. A real PLC student training allows them to program like an industry way.

As a future work, this maquette can be joined in other lectures such as embedded systems allowing students to control the virtual park using a microprocessor or microcontroller programmed in assembly or high level languages. In the PLC lessons context, an auto-valuation can also be created for this Virtual Maquette. Each student has an established time to solve and experiment the PLC program and, at the end, the software application verifies the final behavior punishing each fault and giving a final classification. A statistic data of the tool effectiveness will also be presented.

REFERENCES

[1] Bourne, J. R., Brodersen A. and Dawan M., "The Influence of Technology on Engineering Education", *CRC Press, Inc.*, 1995.

[2] Kroumov, V., Shibayama K. and Inoue A., "Interactive learning tools for enhancing the education in control systems", *33rd ASEE/IEEE Frontiers in Education Conference*, 2003.

[3] Culley S., "Design Applications in Industry and Education", *13th International Conference on Engineering Design*, John Wiley and Sons 2001.

[4] Gallego, F., Tecnoastro Webpage, http://tecnoastro.iespana.es/

[5] Foley, J. D., "Computer graphics: Principles and Practices", *Addison-Wesley Professional*, 1995.

[6] Virtual Automação, http://www.virtualautomacao.com.br, 2007.

[7] Tan, K. C., Wang, L., "Modern Industrial Automation Software Design", *Wiley-IEEE Press*, 2006

[8] Omron Automation Products and Services Webpage, http://www.omron247.com

[9] GLScene Webpage, http://glscene.org/

[10] Gadre, D., V., "Programming the Parallel Port: Interfacing the PC for Data Acquisition & Process Control", *CMP Books*, 1998.

[11] Pallas-Areny, R., Webster, J., G., "Sensors and Signal Conditioning", *John Wiley & Sons*, 2001.

[12] Dunning, G., "Introduction to Programmable Logic Controllers", *Delmar Thomson Learning Inc.*, 2002

[13] Rohner, P., "Plc: Automation with Programmable Logic Controllers", *University of New South Wales Press*, 1996.