Enhancement of PLC programming learning based on a virtual laboratory

Wen-Jye Shyr

National Changhua University of Education
Changhua, Taiwan

ABSTRACT: This study develops a novel method using a virtual laboratory for Programmable Logic Controller (PLC) programming learning. Students use the Multiprog software and the initial program to develop their own applications. Some of the virtual tools that are available to design and integrate the different parts of a mechatronics design are also discussed. Analytical results indicate that experiments supported by Multiprog virtual laboratory user friendly control interfaces generate positive results with respect to knowledge and understanding among the students of the material.

INTRODUCTION

Engineering education places a heavy priority on laboratory experience. Given that several objectives of engineering education have been discussed [1][2], laboratories must orient students on how to perform experiments in real or simulated environments.

Virtual learning environments are widespread in higher education, typically used to deliver instructional materials and facilitate communication within a course [3]. Computer-based education programs have been increasingly adopted in schools to supplement or replace traditional teaching methods. Teachers have performed many studies to discern the impacts of new teaching techniques on students learning abilities. Computer-based virtual simulation is being widely used for the purposes of engineering education. A paper by Georgiev et al presents experiences in building virtual laboratories and provides a discussion of important and relevant issues with regard to the pedagogy, software and equipment utilised [4]. Hassapis and Pavlidou present an exercise that addresses all the practical issues encountered in a real-life application of an industrial control [5]. The exercise is based on the use of the Multiprog software engineering workbench, which is a suite of tools that professionals use to develop applications for this class of PLC programming learning. Morgil et al performed an experiment to determine the academic successes of students who are given computer-aided conventional education [6]. Visualisation and conduct of laboratory experiments are the most effective ways to simplify and clarify the comprehension of complex theory [7]. Gomis et al used commercial PLC to automate a manufacturing cell and communicate with other systems [8]. A diverse array of various programming methods has been adopted to write programs for industrial control applications and for programmable logic controller (PLC) programming learning [9].

Mechatronics is a field of technology that requires the integration of mechanical, electrical, electronics and control systems to achieve control and automation [10]. The aim of mechatronics is to have a flexible system that can make fit into its environment, detect critical operation situations and optimise processes that would be dangerous or difficult to control otherwise. This is generally achieved using software and electronic devices that implement a control system to respond to external changes [11][12].

In a virtual laboratory, students are frequently provided with control tools that enable multiple solutions to a given design problem. A broad-based approach, involving student-built projects controlled with a computer, encourages creativity and excitement about the subject [13]. Both of the studies on the use of virtual experiments in electronics, microelectronics, nano-science [14], and biomechanics [15] have been positive, and extolled the advantages and potential. This virtual laboratory addresses the following pedagogical issues: 1) introducing mechatronics; 2) learning how to plan a mechatronics project; 3) implementing easy PLC programs using the various Multiprog languages;

4) learning how to monitor processes; 5) grasping PLC automation concepts and noticing the importance of standards, and 6) understanding how virtual laboratory processes work.

This study addresses the influence of teaching technologies involved in the subjects related to mechatronics with PLC on students' academic success and permanence in learning. A simulator is integrated into the Multiprog software, and the course students are permitted to program the PLC in order to control the virtual process. With respect to its contributions, in addition to designing software architecture to perform virtual laboratory experiments, this work undertakes one experiment in structured environments.

THE MULTIPROG MAIN CONCEPTS

Multiprog is a standard programming system for international Electro-Technical Commission (IEC) designed PLC and conventional PLC. It is based on the standard IEC61131-3, and includes the full range of IEC features. Multiprog makes all functions easily accessible via a menu, and can generate a project with a few dialogues. Having finished that, the students can immediately start developing a program. The IEC61131-3 standard is the modern approach for a wide range of programmable automation and control systems, not just PLC. Because the programming is now standardised, the teaching of programming becomes largely independent of the actual hardware being utilised. Moreover, the techniques learned and programs that are subsequently developed can be transferred between systems from different manufacturers.

THE IEC61131-3 STANDARD

The need for a new standard for PLC is widely recognised in the international industry community. A working group within the IEC is set up to address the complete design of PLC, including the hardware design, installation, testing, documentation programming and communications. The IEC recognises that industrial instrumentation and control systems need an open system approach to construct large systems by equipment from different manufacturers. The IEC61131-3 standard, published in March 1993, was the first standard to define the programming of control systems such as PLC. Ladder Programming has become a popular graphical language for programming PLC, but unfortunately has several problems.

The IEC61131-3 standard defines a software model for industrial controllers. It stipulates the composition and construction of programs that can be adopted in the model, and how such programs interact with the host machine and with other programs.

PROGRAM ORGANISATION UNITS

Program Organisation Units (POUs) are the crux of program development based on IEC61131-3. They can be categorised into three types, Functions (FUN), Function Blocks (FB) and Programs (PROG). A FUN is defined as a program organisation unit that yields exactly one data element when executed. A FUN does not contain internal memory. The same arguments always yield the same output. Figure 1 illustrates the standard function in FUN. An FB can process several outputs, and contains internal state information. Every function block instance has a structure with internal data, the inputs or default input values, and output, last output or default output values. Figure 2 illustrates the standard function in FB. A PROG combines the FUN and FB with access to the I/O variables in POU.

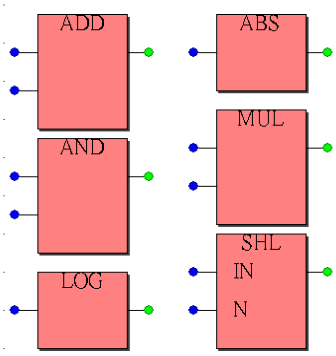


Figure 1: The standard function in FUN.

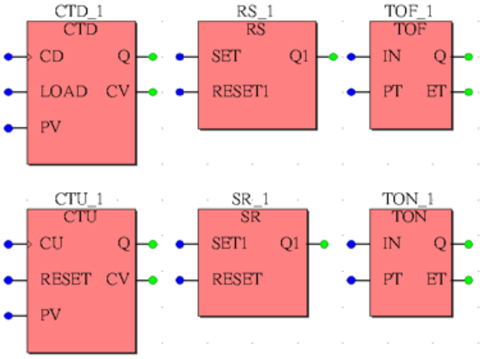


Figure 2: The standard function in FB.

The use of blocks has two advantages. First, the interface must be defined exactly, which means that the block operation and behaviour also have precise definitions. The second is linked to the programmer's skills. The code can be written with any of the five languages that the standard defines. The standard contains five different languages, which can be mixed within an application. There are three graphical languages: 1) Sequential Function Chart (SFC); 2) Function Block Diagram (FDB); and 3) Ladder Diagram (LD); and two text-based languages: 1) Structured Text (ST) and 2) Instruction List (IL).

The SFC is an effective means of representing the key elements of a sequential process, i.e. the conditions required for passing from one state to another and the effects (physical outputs) present while in a particular state. The FDB looks rather like a circuit diagram of logic elements, which may be primitive (e.g. AND gates) or more sophisticated (RS flip-flops). The LD resembles a wiring diagram of a relay control system, whereas the ST resembles a Pascal program and the IL resembles an assembly language program [16].

EXPERIMENT

The virtual laboratory is based on the elevator flowchart illustrated in Figure 3. Table 1 shows the PLC contact point planning for the elevator. The circuit for elevator is shown as Figure 4.

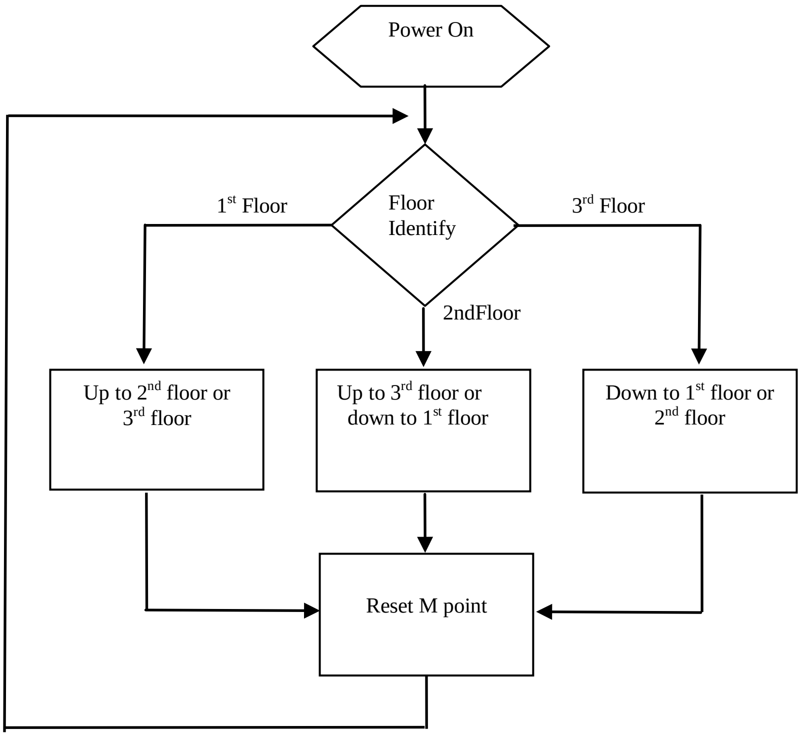

Power On
Floor Identify
1st Floor
3rd Floor
2ndFloor
Up to 2nd floor or 3rd floor
Up to 3rd floor or down to 1st floor
Down to 1st floor or 2nd floor
Reset M point

Figure 3: Flowchart of elevator processing.

Table 1: PLC contact point planning for elevator.

PLC Contact Point	Function	DemoIO address
X0	Power Switch	DemoIO_B00C000_I
X1	1F Limit Switch	DemoIO_B00C001_I
X2	2F Limit Switch	DemoIO_B00C002_I
X3	3F Limit Switch	DemoIO_B00C003_I
X9	1F Push Button (PB)	DemoIO_B00C009_I
X10	2F Push Button (PB)	DemoIO_B00C0010_I
X11	3F Push Button (PB)	DemoIO_B00C0011_I
Y0	Motor forward (Elevator up)	DemoIO_B00C000_O
Y1	Motor reverse (Elevator down)	DemoIO_B00C001_O

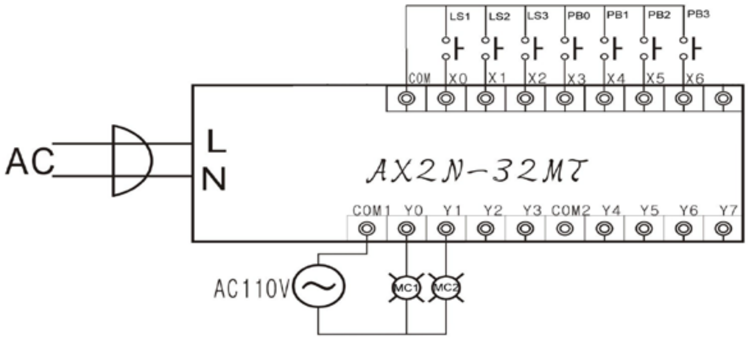


Figure 4: The circuit for elevator.

VIRTUAL LABORATORY SETUP

More than familiarising students with the virtual platform, this session involves an important part of the work. Although free to select the language and the approach to solving, students must defend their choice to the other students and the teacher.

PROCESS

The elevator position should be detected in advance when the elevator is in the Power ON state (Figure 5). If the elevator position is at the 1st floor, then the user can push the 2nd or 3rd floor button (Figure 6). If the elevator position is at the 2nd floor, then the user can push the 1st or 3rd floor button (Figure 7). If the elevator position is at the 3rd floor, then the user can push the 1st or 2nd floor button (Figure 8). The *M* point is reset when the elevator reached the wanted floor (Figure 9).

This study develops PLC to control this elevator process, because PLC is flexible and robust, and easy for maintenance and engineering staff to modify and debug. The programs are developed using Multiprog software, using various IEC61131-3 languages, human machine interface (HMI) capability and simulator.

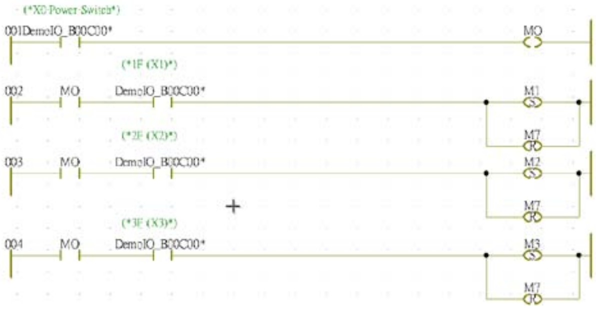


Figure 5: Elevator position detect.

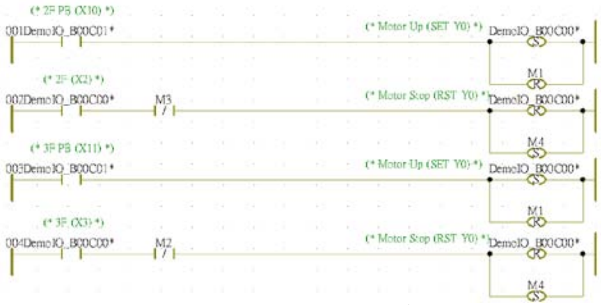
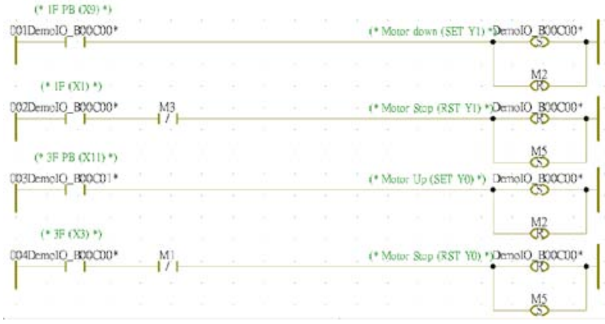


Figure 6: Elevator position is at 1st floor.



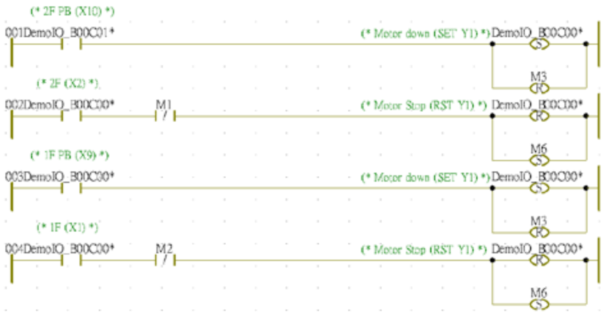
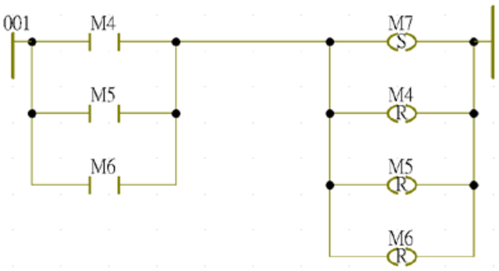


Figure 7: Elevator position is at 2nd floor.

Figure 8: Elevator position is at 3rd floor.



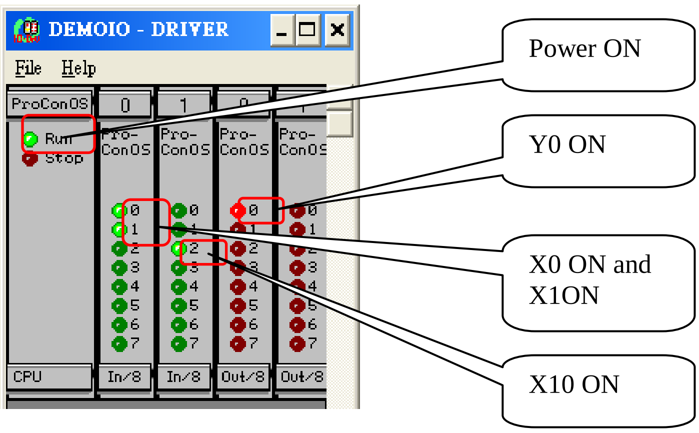


Figure 9: *M* point.

Figure 10: All indicators show PLC signal.

SIMULATION

The simulation capability of the Multiprog software enables virtual laboratories to be constructed without the requirement for any other external software or hardware. The program has a virtual process section (created by the teacher), where the real system is simulated. Therefore, the new program sections created by the students read the sensors' information from the monitor, and write the actuation of the motors and valves. Changes in the states of the motors and valves change the sensor information. The sensor, motor and valve states are defined as global variables, and hence can be utilised anywhere in the program without any problem.

MONITORING

The process evolution is monitored by the HMI utility in the Multiprog software. All the components, including motors, valves, sensors and buttons are placed on a single screen where the evolution of the process variables can also be noted. Figure 10 shows all indicators of PLC signal. In Figure 10, the input(X) state including X0-X15 (Slot In/8[0] 0-7 represents X0-X7; Slot In/8[1] 0-7 represents X8-X15) and the output(Y) state including Y0-Y15 (Slot Out/8[0] 0-7 represents Y0-Y7; Slot Out/8[1] 0-7 represents Y8-Y15).

PLC SOFTWARE

The PLC software for virtual laboratory based on the elevator is presented in Table 2.

Table 2: The PLC software.

LD X3	SET M5	RST Y1
SET M0	LD M0	RST M3
LD M0	AND X2	LD M4
AND X0	AND X5	SET Y0
AND X5	SET M6	LD M4
SET M1	LD M1	AND X2
LD M0	SET Y0	RST Y0
AND X0	LD M1	RST M4
AND X6	AND X1	LD M5
SET M2	RST Y0	SET Y1
LD M0	RST M1	LD M5
AND X1	LD M2	AND X0
AND X4	SET Y0	RST Y1
SET M3	LD M2	RST M5
LD M0	AND X2	LD M6
AND X1	RST Y0	SET Y1
AND X6	RST M2	LD M6
SET M4	LD M3	AND X1
LD M0	SET Y1	RST Y1
AND X2	LD M3	RST M6
AND X4	AND X0	END

ASSESSMENT

Assessment is utilised in order to elevate the standards in terms of teaching, learning and student achievement. Assessment quality has a marked impact on student willingness to work hard and encourages teachers to focus on ways of improving individual learning attitudes. Assessment occurs continually since judging oneself and others are common practice. The students were requested to answer an anonymous questionnaire and submit it to the teacher. Students responded to questionnaire items on a Likert scale ranging from 1 for *strongly disagree* to 5 for *strongly agree*.

The laboratory course included lectures and laboratory experiments. The course was managed by one teacher and one graduate assistant. Seventeen students, from the population of undergraduate students, participated in the course. Students were required to prepare before the laboratory and write a report upon completing the laboratory. After completing the laboratory exercises, students were given the option of taking a final test or completing a project.

For a system, more than two methods may be combined together, which could give extra confidence regarding the result accuracy by concurrency of data produced from these assessment methods. The methods of questionnaire survey, interview and observation were used in the assessment of the virtual laboratory experiment. A questionnaire survey was used to collect assessment data from the students. Observation and review were used in conjunction with the questionnaire survey as data collection methods.

The survey questions and results are presented in the Table 3. The average response of 3.6 demonstrated that the materials, practices and work were inadequate. In future laboratory work, the teacher should expand upon curricular materials if the budget is sufficient. After evaluating the results, the teacher believes that the virtual laboratory had significantly improved the quality of the course. The course methodology also proved satisfactory.

Table 3: Survey results.

Assessment Items	Average Response
Q1. The course was challenging and interesting.	4.3
Q2. I have learned useful information during the course.	4.2
Q3. The background information was clearly written	4.2
Q4. The materials were helpful in the course.	4.0
Q5. I will recommend this course to other students.	4.1
Q6. The materials were well supplied.	3.6
Q7. There were enough practice sessions.	3.6
Q8. The virtual laboratory may resemble a real industrial process.	4.2
Q9. The virtual laboratory worked appropriately.	3.6

As a result of the interviews realised with the students in unstructured manner, it has been determined that the students can easily use the system in teaching of technologies covered by PLC programming learning presented in this study. It was observed that the students enjoyed using the virtual laboratory experiment. However, some students encountered a problem with operating the virtual objects and required help on how to control the virtual objects at the beginning of the virtual laboratory exercise. Reminding the students to first read the operation rules and giving them a chance to pre-practice the operation, the problem was effectively overcome.

CONCLUSIONS

IEC61131-3 is a reference standard for PLC programming, and has been successfully used in industry. This study develops a virtual laboratory for learning PLC. This virtual laboratory incorporates a simulator. The students can adopt the software during classes and at home, to control the virtual process, and to learn the fundamental automation concepts that will be helpful in their professional careers. In addition to learning how to program a PLC, students are also oriented on how to plan and structure a general automation project.

Analytical results indicate that the proposed virtual laboratory is successful. A comprehensive virtual laboratory simulates an actual environment for students, who can benefit from a virtual laboratory once becoming familiar with related devices. Besides enabling a teacher to adjust the conventional course curriculum, a virtual laboratory allows students to develop their own programs and identify implementation-related concerns. Survey results reveal a favourable opinion among students participating in the course. Importantly, this work provides insight into the efficacy of a virtual laboratory in enhancing student learning. Based on these encouraging results, it concludes that this positive experience is worth repeating in the classroom.

REFERENCES

1. Gil, L., Blanco, E. and Aul, J.M., The virtual laboratory concept applied to strain measurements. *European J. of Engng. Educ.*, 25, **3**, 243-251 (2000).
2. Nickerson, J.V., Corter, J.E., Esche, S.K. and Chassapis, C., A model for evaluating the effectiveness of remote engineering laboratories and simulations in education. *Computers & Educ.*, 49, 708-725 (2007).
3. McGill, T.J. and Hobbs, V.J., How students and instructors using a virtual learning environment perceive the fit between technology and task. *J. of Computer Assisted Learning*, 24, 191-202 (2008).
4. Georgiev, S., Roth, H., Stefanova, S., Georgiev, T., Stoyanov, E. and Rosch, O., How and why to build and use virtual laboratories. *World Transactions on Engng. and Technol. Educ.*, 1, **2**, 191-195 (2002).
5. Hassapis, G. and Pavlidou, N., A virtual laboratory exercise to teach the development of industrial computer applications. *World Transactions on Engng. and Technol. Educ*, 2, **3**, 411-416 (2003).
6. Morgil, I., Yavuz, S., Oskay, O. and Arda, S., Traditional and computer-assisted learning in teaching acids and bases. *Chemistry Educ. Research and Practice* (2005).
7. Yu, J.Q., Brown, D.J. and Billet, E.E., Development of a virtual laboratory experiment for biology. *European J. of Open, Distance and E-Learning*, 27 (2005).
8. Gomis, O., Montesinos, D., Galceran, S., Sumper, A. and Sudria, A., A distance PLC programming course employing a remote laboratory based on a flexible manufacturing cell. *IEEE Transactions on Educ.*, 49, **2**, 278-284 (2006).
9. Oriol, G.B., Daniel, M.M., Samuel, G.A., Joan, B.J. and Antoni, S.A., A chemical process automation virtual laboratory to teach PLC programming. *Inter. J. of Engng. Educ.*, 3, **2**, 403-410 (2007).
10. Guerra-Zubiaga, D., Ramón-Raygoza, E., Lara-Prieto, V., Parkin, R. and Jackson, M., A Systems approach to a final year mechatronics design course. *Inter. J. of Engng. Educ.*, 26, **1**, 9-17 (2010).
11. Ramasubramanian, M.K. and Noori, M., Evolution of mechatronics into a graduate degree program in the United States: The NC State University Master of Science program with mechatronics concentration. *Inter. J. of Engng. Educ.*, 19, **4**, 519-524 (2003).
12. Smaili, A. and Chehade, S., Effective integration of mechatronics into the mechanical engineering curriculum: A cooperative, project-based learning model with seamless lab/lecture implementation. *Inter. J. of Engng. Educ.*, 21, **4**, 739-744 (2005).

13. James, P., Mechatronics and automotive systems design. *Inter. J. of Electrical Engng. Educ.*, 41, **4**, 307-312 (2004).
14. Gustavsson, I., Thomas, O., Akesson, H., Zackrisson, J. and Hakansson, L., A remote electronics laboratory for physical experiments using virtual breadboards. *Proc. ASEE Annual Conf. and Exposition*, USA (2005).
15. Barr, R., Pandy, M., Petrosino, A., Austin, B. and Goldberg, E., Classroom testing of VaNTH biomechanics learning modules. *Proc. American Society for Engng. Educ. Annual Conf. and Exposition*, USA (2004).
16. Thapa, D., Dangol, S. and Wang, G.N., Transformation from Petri nets model to programmable logic controller using one-to-one mapping technique: Computational intelligence for modeling control and automation. *IEEE Computational Intelligence for Modelling Control and Automation*, 2, 228-233 (2005).