Development and Validation of a Less Expensive and Portable PLC Module for Students Training in Industrial Automation

Article *in* Electronics ETF · December 2022

DOI: 10.53314/ELS2226039P

1 author:

Joel Palomino
Universidad Tecnológica del Perú

Development and Validation of a Less Expensive and Portable PLC Module for Students Training in Industrial Automation

Joel Fernando Palomino Masco

Abstract— **Programmable logic controllers (PLC) devices play an important role in automation systems; therefore, technicians must have adequate training with these devices. If a student wants to learn about PLCs, they will find two problems. First, training programs are expensive, and second, PLC modules are not suitable for gaining hands-on skills. This study proposes the implementation and validation of a less expensive PLC module that is accessible to students interested in learning about this technology. Two prototypes were developed using a PIC16F877 and an ATMEGA328P microcontroller. This study describes the enhancement made in the second prototype over the first prototype. Finally, a comparison test between the two prototypes and two commercial PLCs is demonstrated. A MyRIO programmed as a data logger monitored the PLC operation for 12h. The program recorded the response time of the PLC output each time a 24 volts signal was sent to its input. The results show that commercial PLC has a fast and more stable response time. Nevertheless, the developed modules proved to work effectively for 12h, demonstrating their efficiency for training.**

Index Terms— **Interactive, Less Expensive, PLC Trainer, Portable.**

Original Research Paper
DOI: 10.53314/ELS2226039P

I. Introduction

BECAUSE of the rapid pace of technological development, professionals must keep their knowledge up to date. This is true for automation systems, with the introduction of programmable logic controllers (PLC). These devices were designed in the 1970s to replace the relay control devices used in the industry to control machines and processes. The device includes a wiring interface for connecting to external devices and a programmable interface for developing a control algorithm [1]-[2].

PLC devices are used in industry to reduce costs. Production lines originally operated by employees are now replaced with PLCs. Therefore, industries require employees with skills to operate and maintain PLCs in automation environments. Therefore, students in institutes and universities must be trained in PLCs programming and installation.

The availability of specialized courses as new PLC devices emerge and the cost associated makes it difficult for students to find adequate training [1]. Therefore, institutions prefer a theoretical and simulated approach, which hinders student practical training [3]. However, PLC training kits are not suitable for students to acquire hands-on skills, because the prefabricated casing provides almost no access to the PLC wiring [4]. Therefore, it is difficult to find appropriate training for this purpose.

Commercially available PLCs devices can cost around $300 or more; however, related research indicates that economical devices can be designed using commercial devices. For example, [5] designed a PLC module using the Atmega8L that had a cost of around $20. Similarly, [6] describes the design of a less expensive PLC module based on the PIC18F4580. Finally, [1] focuses on the design of a training module based on the DVP14SS2 PLC, the reported cost is around $214.

According to [7], the features of a PLC training module should emphasize hardware connection of the inputs and outputs. Additionally, [3] states that PLC training modules must be portable and easily accessible rather than modules that fit in large spaces. Their design consisted of a small training kit with dimensions of 47 x 37 x 27 cm and a weight of 10 kg. Similarly, [8] developed a training PLC module with dimensions of 100 x 44.1 x 92.7. Although the design is not portable, his main objective was to solve the lack of training kits.

A questionnaire to evaluate the effectiveness of a PLC module designed with an ATMEGA256A3U microcontroller was created in [9]. The questions were applied to 60 laboratories, and the main topic was the ease of use of the module. The results showed that 51 laboratories evaluated the module with good to excellent remarks. Only 3 laboratories sent 3 poor remarks.

Less expensive PLC modules can also be implemented using development boards and adding the required power circuit for the I/O pins. Parab [10] describes the use of an Arduino UNO to create a temperature control system with relay logic. Similarly, Comlan [11] uses Arduino to execute programs developed from a GRAFCET free editor in Java. Another platform used is Raspberry Pi, Vieira [12] describes the use of these module to implement a PLC designing an industrial shield for the I/O pins. In the same way, Ahmad [13] uses the Raspberry Pi 3 B as a PLC to control the sedimentation process of water treatment plant. For the modules the described hardware consists of a voltage divider with an isolation circuit for the inputs; and the outputs pins control a relay that turns on/off the loads connected to it.

The use of raspberry as a PLC can go further for real-time applications. Ogorodnyk [14] describes the use of a Raspberry Pi to collect data in real-time from sensors of molding machines. Additionally, Mastang [15] employs a Raspberry Pi to

Manuscript received 02 December 2021. Received in revised form 07 March 2022. Accepted for publication 17 May 2022.

Joel Fernando Palomino Masco is with the Faculty of Mechanical Engineering, Universidad Tecnologica del Perú, Lima, Perú (e-mail: C19338@utp.edu.pe).

collect data from a PLC and deliver information to a database for real-time monitoring. The synchronization of the device is done from applications that acquire time from a computer. For the case of stand-alone applications an external real-time clock can be added.

Regarding the programing software, most commercial PLC require a license for its use. Another option is to employ open source software. Rodriguez [16] in 2014 presents OpenPLC, a software that allows to write PLC programs according to the IEC 61131-3 standard. This software has been used to test the PLC security against anormal operation [17], and against cyber attacks [18]. These tests demonstrate that the software is a valid platform to evaluate PLC functionality. Having access to the code enables to perform various studies on the PLC operation in search of faults, which is difficult with a commercial PLC software. Finally, another software is LDmicro, Vishnu [19] uses this software to develop an accessible PLC module for developing countries.

Previous research, show that PLC modules can be developed from microcontrollers and commercial development board; by adding additional circuitry to operate with higher voltages as used in the industry. These studies focus more on the development of portable PLC modules, with the results focusing on their effectiveness as relay controllers. However, these studies do not compare their performance with that of commercial PLC. Therefore, this study is novel in that it compares the performance of a modular PLC with that of a commercial Siemens PLC. The ATMEGA328P was used to develop the modular PLC and the validation test was performed with National Instruments MyRio-1900 [20]. This device has configurable I/O that are configured with LabView software and is used for developing applications that work with onboard FPGA and microprocessor. For this research the MyRio device was configured as a data logger. The contents of this study are as follows; the first section describes the design and implementation of the PLC module using two different microcontroller and explains how the first prototype was improved; the second section is about the programming interface; the third section shows the tests performed to compare the module performance with a commercial PLC followed by a discussion of the results; finally, the conclusions and future research are stated.

II. Design of PLC Module

A. Hardware Design

The PLC module uses commercially available components; the ladder programming language was used, which is simple for electricians to understand, and it works with 24 volts inputs and outputs. Two prototypes were developed in this step using two different microcontrollers, the PIC16F877 and ATMEGA328P. Because the PLC works with high-power devices, the control unit must be protected. Therefore, the input is designed around an optocoupler, and the output is designed around a relay, to electrically isolate the power circuit from the control circuit.

Fig. 1. shows the input circuit for the first prototype. To activate the LED inside the optocoupler, a 24-volts input is connected to a voltage divider. This signals the phototransistor, which activates the microcontroller input through an inverting gate and resistor R1. Additionally, an external LED is connected to the phototransistor as a signaling device.

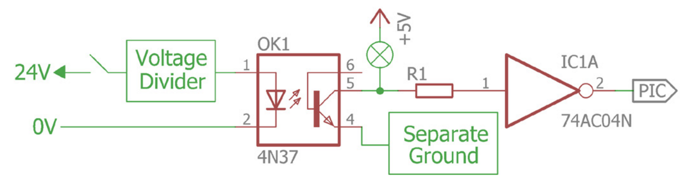

Fig.1. Electronic diagram of PLC module input for the first prototype.

In the second prototype, the inverting gate was eliminated. The external LED was connected in parallel with the optocoupler input to allow the user to diagnose the input circuit in case the optocoupler failed. Finally, the output of the phototransistor is connected directly to the microcontroller through resistor R2. The circuit for the input is shown in Fig. 2.

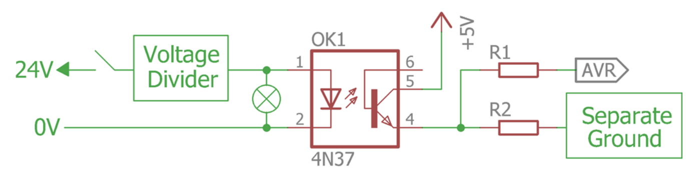

Fig. 2. Circuit diagram of PLC module input for the second prototype.

Fig. 3. shows the output circuit for the first prototype. The digital pin of the microcontroller activates the LED of an optocoupler through an op-amp wired as a buffer. The transistor on the other side is connected to the coil of a relay. The relay normally open contacts are the exit of the PLC. This configuration enables the user to control actuators that operate on different DC or AC voltages. Finally, a LED is connected in parallel with the relay coil to indicate that the output is activated.

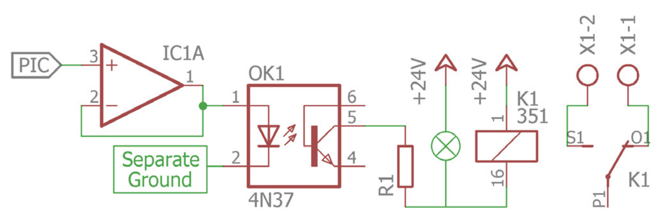

Fig. 3. Electronic diagram of PLC module output for the first prototype.

In the second prototype, the optocoupler and op-amp are replaced by a transistor. The transistor is connected directly to the relay coil. As in the previous case, the normally open contacts of the relay are the PLC output, and a LED is connected in parallel with the relay coil to indicate an activation of the output. The circuit for the output is shown in Fig. 4.

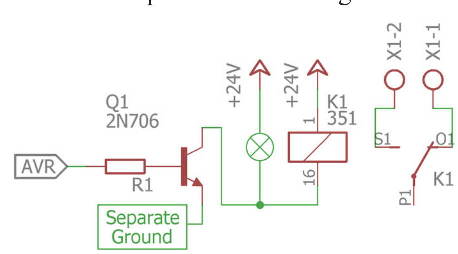

Fig. 4. Circuit diagram of PLC module output for the second prototype.

A 4-pin connector was used to connect two analog pins on both prototypes. The 24-volts energy supply for the module was accessible through two pins. A step-down converter was used to supply 5 volts to the microcontroller. A LED was connected near the voltage supply port to indicate that the device is powered on.

B. Programming Interface

To program the PLC module, "LDmicro" software was used [21]. It is an open-source code that allows users to write ladder diagrams and generate the HEX files to program microcontrollers. The PICKIT is commonly used to program the PIC microcontroller; however, for this case, the BK precision programmer 844USB device programmer was used for its availability in the institution [22]. The first prototype (Fig. 5.) has a 40-pin zif socket for the microcontroller. An external 14 MHz crystal was used to match the clock requirements established by the LD Micro software. Finally, the board dimensions are 17 x 14 cm, including the space for 4 PCB spacers.

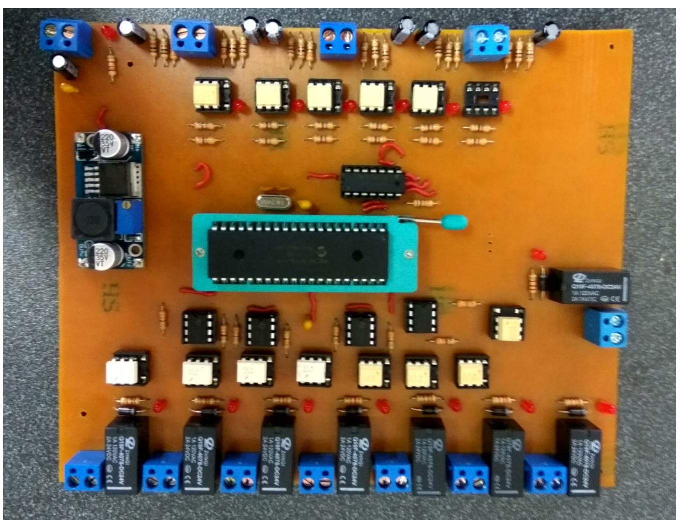

Fig. 5. Developed PLC prototype using the PIC16F877.

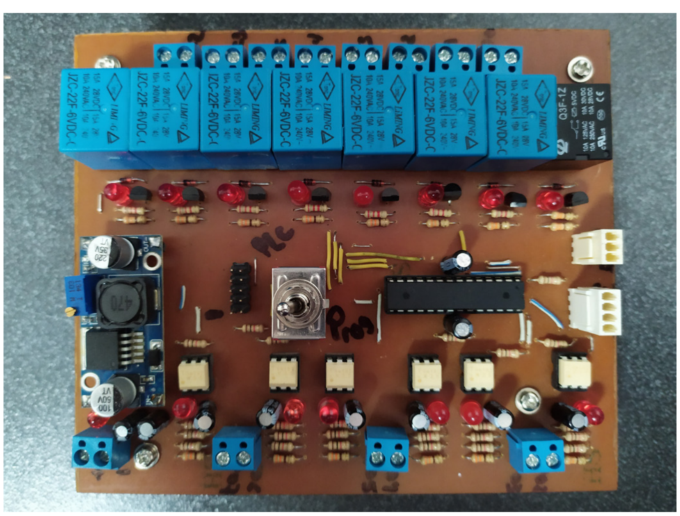

Fig. 6. Developed PLC prototype using the ATMEGA328P.

For the second prototype, a commercial version of the USBasp programmer was used for its low price [23]. A 10-pin male connector located near the microcontroller was used to connect to the programmer. According to the requirements of the LD Micro software, the Atmel internal clock was adequate for proper operation. The developed board is shown in Fig. 6., its dimensions are 14.1 x 11.3 cm, which are smaller than the previous version.

C. Price of Developed PLC Module

The components of the first PLC module, excluding the PIC programmer, cost approximately $60. This price includes a printed circuit board (PCB) and the required tools for PCB etching. For the second module, the price for acquiring the components was also approximately $60. This price includes the AVR programmer, the PCB, and the tools for etching.

D. Corrections made in the second module

With the first PLC module, students demonstrated a bad habit of connecting the supply voltage in reverse polarity, which causes the module step-down converter to explode. To avoid this issue an N channel MOSFET with a 24-volts input was used as reverse supply protection.

Each time the supply voltage is reversed, the MOSFET will automatically shut down the power that supplies the PLC module, protecting all the components.

Another problem was discovered after students disconnected and reconnected the wire several times from the screw-type terminal blocks. Each time the screwdriver was used, the pressure applied slowly broke the solder used in the terminal blocks. After several tests, the PCB tracks and the solder used made false contacts, which cause the program to not work accurately. One solution was to increase the thickness of the PCB board from 1.5 to 3 mm, but this makes it more difficult to cut the board. Another solution was to increase the number of PCB spacers place them near the screw-type terminal blocks.

E. Comparison with Open-Source Modules

This section presents a comparison between the developed module and other commercial open source PLC modules. Table I summarizes the principal technical characteristics of the developed module against the following modules:

1) Controllino [24] mini designed with the ATMEGA328P microcontroller.
2) PICPLC16 [25] designed with the PIC18F97J60 microcontroller.
3) AVRPLC16 [26] is designed with the ATMEGA32 microcontroller.

The information shows that the developed module hardware is similar to other commercial modules. The main difference is in the price. Commercial modules are designed to comply with industrial standards, which increases the price as more precise components are required. The developed module application is intended for education purposes; therefore, protection against harsh, dirty and electrically noise environments is not developed.

Table I
Comparison of Technical Characteristics of Develop Module Against Commercial Open-Source PLC Modules

	Developed module	Controllino Mini	PICPLC16 V7	AVRPLC16 V6
Inputs	6	8	16	8
Outputs	8	8	16	16
Output Type	Relay	Digital and Relay	Relay	Relay
Analog Inputs	2	5	3	8
Supply Voltage	24V DC	12V DC or 24V DC	7 - 23V AC or 9 - 32 V DC	12 - 22 V AC or 16 - 30 V DC
Programming Software	LDmicro	Any Arduino compatible software	mikroC PRO, mikro BASIC PRO and mikro PASCAL PRO	Any software that generates hex files for AVR microcontrollers.
Serial Communication	UART	SPI, UART and I2C	SPI, UART and I2C	UART
Price	$ 60	$ 149	$ 179	$ 159

F. Comparison with commercial PLC Modules

Table II shows a comparison between the developed module and the following industrial PLC module:

1) Siemens S7-1200 CPU1212C [27]
2) Siemens LOGO 24CEO [28]
3) Micro 810 2080-LC10-12QWB [29]

Table II
Comparison of Technical Characteristics of Develop Module Against Commercial PLC Modules

	Developed module	S7 1200 CPU 1212C	LOGO 24CEO	Micro 810 2080-LC10-12QWB
Inputs	6	8	8	8
Outputs	8	6	4	4
Output Type	Relay	Digital	Digital	Relay
Analog Inputs	2	2	4	4
Supply Voltage	24 V DC	20.4 – 28.8 V DC	20.4 – 28.8 V DC	24 V DC/ V AC
Price	$ 60	$ 360	$ 100	$ 130

Table II summarizes the principal hardware characteristics. Where the input and output capability of the developed PLC module is comparable with an industrial PLC. Other characteristics are not shown, because it was not designed to match industrial standards.

III. Validation of PLC Module

A. Proposed Methodology

To test the developed PLC module performance a comparative test with two PLC Siemens was performed, the selected PLC were the Siemens S7-1200 CPU 1211C AC/DC/RLY and the Siemens LOGO. The National Instruments MyRio-1900, programmed as a data logger, was used for the test. Since the purpose of the developed module purpose is for training, the operation time usually doesn't exceed an hour. Nevertheless, the test duration was programmed for 12h to determine whether the developed module can work as a commercial PLC without failing.

The data logger was programmed to send a signal every 5 seconds to the PLC input and count the time ticks until the PLC exited the corresponding output. The PLC program logic is simple, with an input connected to an exit (Fig. 7.).

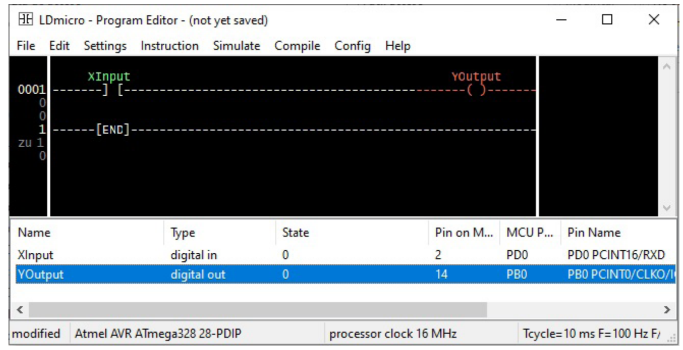

Fig. 7. Ladder Diagram to test the PLC module.

The data logger was programmed to accurately record the PLC operation time. The scan time, which was set to 10 msec was used to determine the operation time of the PLC. The data logger has two programming modes; Real-Time and FPGA modes. In Real-Time mode the smallest clock tick is 100 msec, which is longer than the PLC operation time. However, the FPGA minimum clock tick is 25 ns, which is less than the PLC operation time.

Fig. 8. shows the connection between the data logger and PLC. The module sent a signal every 5 s to the input and record the time required to generate and output. The device operated autonomously and could detect any failure using this method.

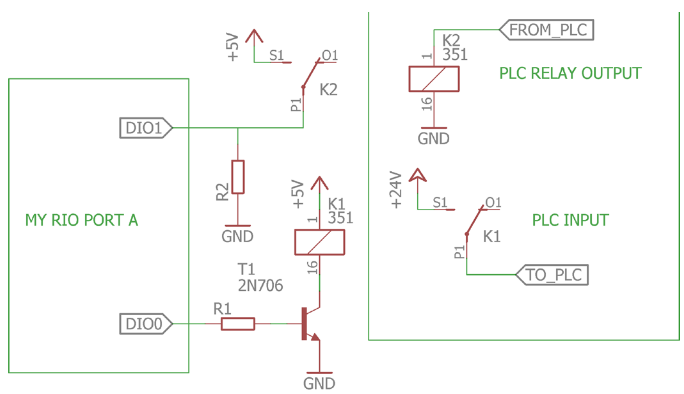

Fig. 8. The connection between MyRIO and PLC module.

As shown in Fig. 8, a relay was used to connect a 5-volts output to the data logger with a 24-volts input of the PLC. Because the relay adds a small delay, this time had to be subtracted from the data registered by the data logger. The relay model is the JZC-22F-6VDC from Ningbo Forward Relay Corporation. According to the datasheet, the operation time is less than 15 msec, which is not enough to estimate the correct timing.

Therefore, the selected relay was tested using an oscilloscope. The result is shown in Fig. 9.

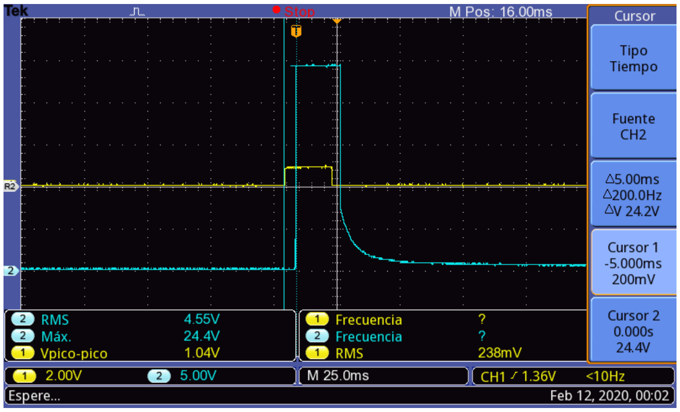

Fig. 9. Operation time of the relay used for Datalogging.

The delay time for the command signal to arrive at the PLC module from the data logger is around 5 ms. This value was used to correct the data obtained from the data logger. Finally, the temperature sensor TCN75A from Microchip, was programmed to visualize if the temperature variation affects the results. The PLC module is connected to the data logger for the described test procedure is shown in Fig. 10.

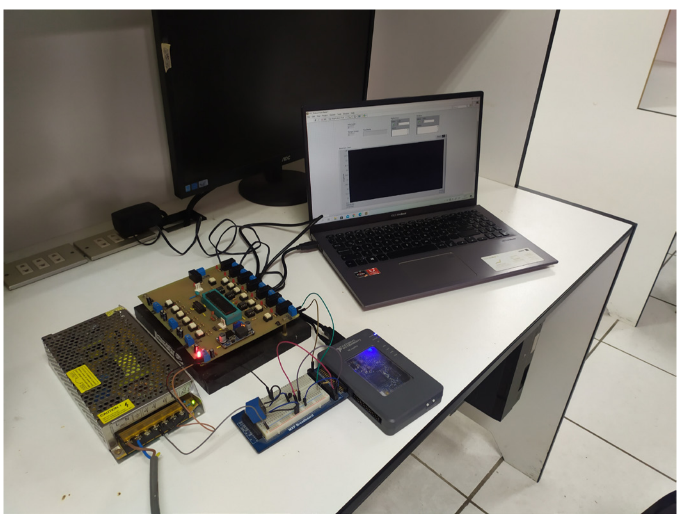

Fig. 10. PLC module connected to the data logger to test its operation for 24 h.

B. Results

Four different PLC models were tested, including the first PLC module, the second PLC module, the Siemens LOGO PLC, and the Siemens S7-1200. The scan time for each PLC was set to 10 msec. Each PLC was subjected to a 12-h test to determine its response time. The following graphs show the results obtained by the data logger, also temperature readings plotted along each dataset. The datasets shown correspond to one input and output, similar results can be obtained when performed on other pins of the PLC. Additionally, the interquartile range of each dataset is shown in tables beneath each graph.

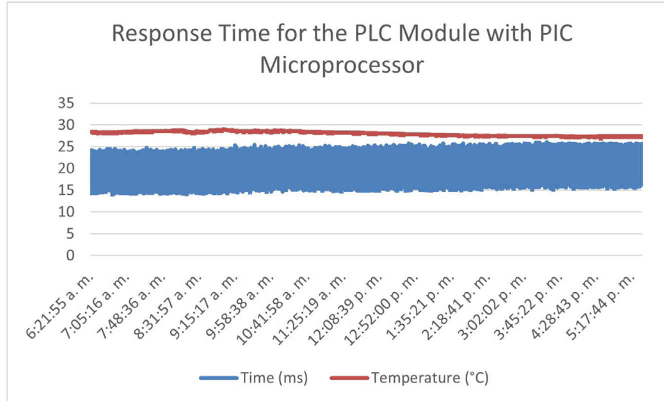

Fig. 11. Dataset obtained from the data logger for the first PLC module.

TABLE III
QUARTILE VALUES FOR THE DATASET OBTAINED FROM THE FIRST PLC MODULE.

Smallest Value	14.2
Q1	17.65
Q2	20.11
Q3	22.59
Largest Value	26.06

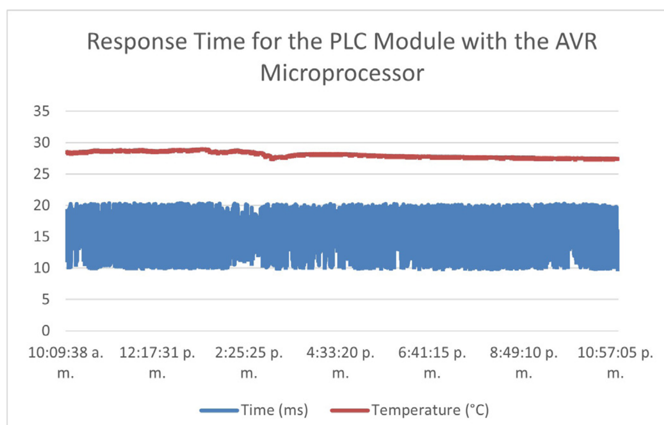

Fig. 12. Dataset obtained from the data logger for the second PLC module.

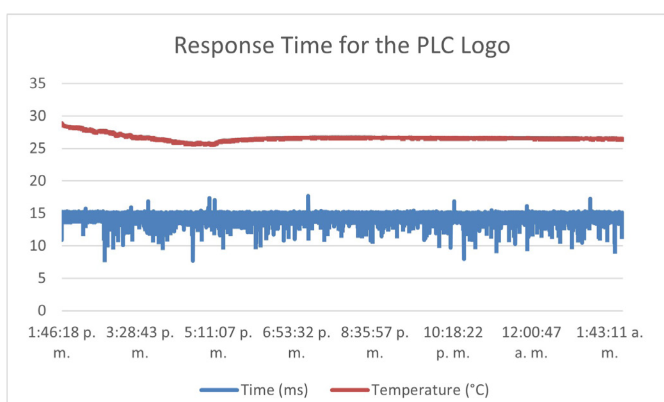

Fig. 13. Dataset obtained from the data logger for the PLC Logo.

Table IV
Quartile Values for the Dataset Obtained from the Second PLC Module.

Smallest Value	9.94
Q1	13.02
Q2	15.26
Q3	17.64
Largest Value	20.36

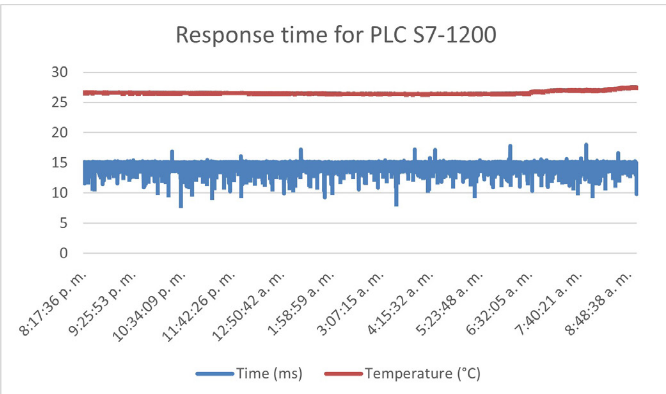

Fig. 14. Dataset obtained from the data logger for the PLC S7-1200.

Table VI
Quartile Values for the Dataset Obtained from the Second PLC S7-1200.

Smallest Value	7.93
Q1	13.9
Q2	14.05
Q3	14.9
Largest Value	17.98

IV. Discussion

As previously stated, the tests were designed to last at least 12h. The objective of the test was to determine the stability of the circuit to response to the input under a controlled time. The datapoints gathered are adequate to validate the functionality of the developed PLC modules.

To analyze the response time, the interquartile range was calculated for each dataset. The results are shown in Table VII. For commercial PLCs the response time is small compared with the developed PLC modules. This can also be confirmed visually in Figs. 10-13.

Table VII
Interquartile Ranges for each PLC Dataset.

Device Name	Interquartile Range
PLC Module with the PIC microcontroller	4.94
PLC Module with the AVR microcontroller	4.62
PLC Logo Siemens	1
PLC S7-1200	1

There are two possible explanations for these results. One is related to the stability of the components used to implement the PLC modules. As stated in section two, the PLC module used commercially available components so that it could be affordable for students. The components used usually have significantly high tolerances and are susceptible to temperature changes. The second explanation is related to the software used to design the ladder diagrams. The LDmicro open source code is a convenient method to program the developed module. However, the ability of the code to produce outputs exactly at the programmed scan time has not been validated. Further tests should be conducted directly in the microcontroller to verify this parameter.

The four PLCs were programmed with a scan time of 10 msec. The median shows how close the PLCs are to this value. The median for Siemens PLC is around 14 msec, with only the second prototype median, which is 15.26 msec is close to this value. The first PLC module median has a value of 20.11 msec, which is 5 msec higher than the second module. This is mainly caused by the number of components used in the first module compared to the second module.

V. Conclusion

A less expensive PLC module was developed using locally available components at a cost of approximately $60. During the design phase, two modules with different components were developed. The second module corrected the errors found in the first module making its response time 5 msec faster when the median values of each dataset were compared. A comparison with other commercial PLC modules shows that the hardware input and output capability are similar.

A comparison test with a SIEMENS Logo PLC and an S7-1200 PLC was performed using a MyRio programmed as a data logger to verify the functionality of the PLC module. Additionally, a temperature sensor was connected to the datalogger, to measure temperature variations that could affect the readings. The duration was programmed to last at 12h. These data show that the PLC can work reliably for a long time without failure.

After analyzing the data collected by the data logger, the results show that the quartile range is small for the SIEMENS PLC compared with the developed PLC. The main reason is the low tolerance of the electronic components used to implement the module. Although it has not been validated, and the scan time of the ladder software may also be responsible for this. This time should be properly validated using direct testing on the microcontroller pins.

VI. Future Research

Further work involves the validation of the PLC module operation under the following environmental conditions:

- Although a temperature sensor was connected to the datalogger, a test to measure the variations of the scan time under different temperatures was not performed. The PLC module can be placed in a test cabin where the environment temperature is controlled.
- Another test is related to the measurement of vibrations. For this test a test bench with controlled oscillations is

required to measure the variation of the scan time under different oscillations conditions.

Finally, a test to verify the effects that the software can have on the scan time is needed. The test consists of measuring the scan time directly in the input and output pins of the microcontroller without additional hardware.

References

[1] R. A. Akparibo, A. Appiah, and O. Fosu-Antwi. "Development of a Programmable Logic Controller Training Platform for the Industrial Control of Processes". *American Scientific Research Journal for Engineering, Technology, and Sciences*. 15(1), 186-196. 2016.

[2] M. Artiyasa, N. Destria, and M. Desima. "Creating kit and plc application with industrial applications (Presented Conference Paper style)". *2nd International Conference on Vocational Education and Technology*. Bali, Indonesia. Nov. 2020.

[3] A. Khairudin,, A. Abu-Samah,, A. Aziz, A. Azlan, H. Karim, and N. Zian. "Design of Portable Industrial Automation Education Training Kit Compatible for IR 4.0 (Presented Conference Paper style)". *7th Conference on Systems, Process and Control (ICSPC 2019)*. Melaka, Malaysia. Dec. 2019.

[4] E. Maarif and S. Suhartinah. "Compact Portable Industrial Automation Kit for Vocational (Presented Conference Paper style)". *International Symposium on Materials and Electrical Engineering (ISMEE) 2017*. Bandung, Indonesia. Nov. 2017.

[5] N. Soares, S. Rogel, and A. Peixoto. "Development of a Modular PLC using low cost microcontroller architecture. (Presented Conference Paper style)". *International Conference on Alive Engineering Education*. Rio de Janeiro, Brazil. Jun. 2017.

[6] U. Sanver, E. Yavuz, C. Eyupoglu and T. Uzun. "Design and implementation of a programmable logic controller using PIC18F4580". *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering*. Moscow, Russia. Jan. 2018.

[7] A. Matthew, A. Adefemi and O. Martins. "Design and Implementation of a PLC Trainer Workstation. Advances in Science". *Technology and Engineering Systems*. 5(4), 755-761. 2020.

[8] S. Soeharto, and S. Wardhana. "Performance of A Programmable Logic Controller". *Journal of Physics: Conference Series*. 2019.

[9] E. Mohamed, L. Fuqiang, and J. Yassine. "Design Mini-PLC based on ATxmega256A3U-AU microcontroller (Paper presentation)". *2014 International Conference on Information Science, Electronics and Electrical Engineering*. Sapporo, Japan. Apr. 2014.

[10] R. Parab, and S. Prajapati. "IoT based relay operation". *International Journal of Engineering and Advanced Technology*. 9(1), 6515-6520. 2019.

[11] M. Comlan, D. Delfieu, and N. Assogba. "Grafcet to arduino: Edit and upload grafcets on an arduino boards (Presented Conference Paper style)". *International Conference on Electrical, Computer, Communications and Mechatronics Engineering*. Mauritius, Oct. 2021.

[12] G. Vieira, J. Barbosa P. Leitao, and L. Sakurada. "Low-cost industrial controller based on the raspberry pi platform [Paper presentation]". *21st IEEE International Conference on Industrial Technology*. Buenos Aires, Argentina. Feb. 2020.

[13] E. Ahmad, M. Ridlo, and H. Ash. "Design and Implementation Supervisory Control and Data Acquisition (SCADA) of Sedimentation Process of Water Treatment Plant (WTP) by Using Raspberry PI 3 B (Presented Conference Paper style)". *4th International Conference on Wireless and Telematics*. Bali, Indonesia. Jul. 2018.

[14] O. Ogorodnyk, M. Larsen, K. Martinsen, and O.V. "Lyngstad, Development of application programming interface prototype for injection molding machines (Presented Conference Paper style)". *8th CIRP Conference of Assembly Technology and Systems*. Athens, Greece. Sept. 2020.

[15] Mastang and M. A. Pahmi. "Development of Raspberry Pi applied to Real-Time Monitoring of Overall Equipment Effectiveness (OEE) (Presented Conference Paper style)". *2nd International Conference on Computer, Science, Engineering, and Technology*. Tangerang, Indonesia. Oct. 2019.

[16] T. Rodríguez, M. Buratto, F. Mauricio, T. Virginia, "OpenPLC: An Open Source Alternative to Automation (Presented Conference Paper style)". *IEEE Global Humanitarian Technology Conference (GHTC 2014)*. California, USA. Oct. 2014.

[17] S. Fujita, K. Hata, A. Mochizuki, K. Sawada, S. Shin, and S. Hosokawa. "OpenPLC based control system testbed for PLC whitelisting system". *Artificial Life and Robotics*. 26(1), 149-154. 2021.

[18] T. Alves and T. Morris, "OpenPLC: An IEC 61,131–3 compliant open source industrial controller for cyber security research". *Computers and Security*. 78. 364-379. 2018.

[19] E. Vishnu, R. Kushalkar, N. Tigadi, K. M. Moudgalya, A. Chipkar, A. Zoitl, M. Akshai, T. Alves. "Programmable logic controller: Open source hardware and software for massive training (Presented Conference Paper style)". *Proceedings: IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*. Washington. USA. Oct. 2018.

[20] National Instruments (2022, March 01). myRIO-1900. [Online]. Available: https://www.ni.com/en-us/support/model.myrio-1900.html.

[21] Nehrutsa, I. (2018, Aug 20). LDmicro: Ladder Logic for PIC and AVR (Version 4.4.2). [Online]. Available: https://github.com/LDmicro/LDmicro/releases/tag/4.4.2.0.

[22] BK Precision. (2018, Jul 30). Device Programmer with USB PC Interface. [Online]. Available: https://www.bkprecision.com/products/device-programmers/eprom-programmers/844USB-device-programmer-with-usb-pc-interface.html.

[23] Fischl, T. (2018, July 10). USBasp - USB programmer for Atmel AVR controllers. Thomas Fischl. https://www.fischl.de/usbasp/

[24] Controllino GmbH (2022, February 22). Controllino 100% Arduino compatible PLC. [Online]. Available: https://www.controllino.com/

[25] MikroElektronika d.o.o. (2022 February 24). PICPLC16 V7A. [Online]. Available: https://www.mikroe.com/picplc16-v7a

[26] MikroElektronika d.o.o. (2022 February 24). AVRPLC16 V6. [Online]. Available: https://www.mikroe.com/avrplc16

[27] Siemens (2022 February 26). 6ES7212-1HE40-0XB0. [Online]. Available: https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7212-1HE40-0XB0

[28] Siemens (2022 February 26). 6ED1052-2CC08-0BA1. [Online]. Available: https://mall.industry.siemens.com/mall/es/WW/Catalog/Product/6ED1052-2CC08-0BA1

[29] Rockwell Automation (2022 February 26). Micro810 Programmable Controllers. [Online] Available: https://literature.rockwellautomation.com/idc/groups/literature/documents/um/2080-um001_-en-e.pdf