AIP Conference Proceedings

RESEARCH ARTICLE | JULY 16 2025

HMI SCADA using Python and QML programming integrated to outseal PLC ✓

Muhammad Husni Muttaqin; Eko Budi Utomo; Diky Zakaria ✉; Ade Gafar Abdullah

AIP Conf. Proc. 3141, 050005 (2025)
https://doi.org/10.1063/5.0265590

View Online

Export Citation

Articles You May Be Interested In

Algorithm for creating an epizootic situation monitoring system for parasitic diseases of animals using a geoinformation system

AIP Conf. Proc. (October 2022)

Monitoring systems design and data acquisition on powerhouse and utility using MES interface based on programmable logic controller

AIP Conf. Proc. (December 2023)

Application of the internet of things for motor speed control and temperature monitoring based on raspberry and programmable logic controllers

AIP Conf. Proc. (February 2025)

AIP Publishing

18 July 2025 15:17:36

HMI SCADA using Python and QML Programming Integrated to Outseal PLC

Muhammad Husni Muttaqin[1,a)], Eko Budi Utomo[2,b)], Diky Zakaria[3,c)] and Ade Gafar Abdullah[4,d)]

[1]*PT. Syergie Indoprima, Bandung, Indonesia*
[2]*Program Studi Teknik Mekatronika, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia*
[3]*Program Studi Mekatronika dan Kecerdasan Buatan, Universitas Pendidikan Indonesia, Bandung, Indonesia*
[4]*Electrical Engineering Study Program, Universitas Pendidikan Indonesia, Bandung, Indonesia*

[a)] husnimuttaqin@student.upi.edu
[b)] ekobudi_u@pens.ac.id
[c)] Corresponding author: dikyzak@upi.edu
[d)] ade_gaffar@upi.edu

18 July 2025 15:17:36

Abstract. Python is a free and open-source language. Python is a versatile language, as it can run on various platforms, including Windows, Linux/Unix, and MacOS. Moreover, Python's open-source nature encourages the development and dissemination of open-source applications. This research focuses on the development of a Human-Machine Interface (HMI) SCADA system using Python and QML programming languages. The aim is to integrate this system with Outseal PLC, a programmable logic controller commonly used in industrial automation. To achieve this goal, an experimental method is employed. The first step in the experimental process involves designing the system wiring, which establishes the connections between the various components of the HMI SCADA system. Once the wiring is complete, the user interface development phase begins using Python's QML. QML, or Qt Meta-Object Language, is a declarative language that allows developers to design interactive user interfaces with ease. After developing the user interface, the next step is to integrate it with the Outseal PLC. This integration enables the system to monitor input and output status of PLC, providing real-time data on the industrial processes it controls. The developed system can send control signals to the PLC using the Modbus protocol. To validate the effectiveness of the developed system, it was tested with users during a training session held at PENS Surabaya. The participants were given hands-on experience in creating a simple HMI SCADA system that is connected to the Outseal PLC. Through this training, they gained a better understanding of HMI SCADA development using Python and QML.

INTRODUCTION

In recent times, Supervisory Control and Data Acquisition (SCADA) systems have assumed a pivotal role across various industries. These systems play a vital role in monitoring and regulating multiple processes, resulting in improved operational efficiency and effectiveness. The Human-Machine Interface (HMI) holds particular significance within SCADA systems as it offers users a visual representation of monitored data and enables interactive engagement with the system. Due to their versatility, user-friendliness, and the availability of open-source libraries, programming languages like Python and QML have gained popularity for constructing HMI SCADA systems [1].

Python, being a free and open-source programming language, can run on multiple platforms, including Windows, Linux/Unix, and Mac OS X. Consequently, Python promotes the creation and dissemination of open-source

Proceedings of the 4th International Symposium on Materials and Electrical Engineering (ISMEE 2023)
AIP Conf. Proc. 3141, 050005-1–050005-7; https://doi.org/10.1063/5.0265590
Published under an exclusive license by AIP Publishing. 978-0-7354-5196-4/$30.00

applications [2]. Programmable Logic Controllers (PLCs) are a category of embedded systems used for controlling machines and processes. Initially introduced in the early 1970s to replace outdated and costly relay control logic, PLCs have provided increased flexibility, enhanced reliability, improved communication capabilities, quicker response times, and simplified troubleshooting. Industrial control engineers have shown consistent interest in PLCs, leading to the development and standardization of their design techniques and programming languages [3].

SCADA systems are technically defined as systems that facilitate data acquisition from remote locations and supervisory control over various decisions based on the collected data. SCADA integrates data acquisition and telemetry, encompassing data gathering, transmission to a central location, critical control and analysis, as well as data presentation on screens and displays for operators and decision-makers [4].

Modbus is an open protocol standard extensively employed in industrial equipment for monitoring and control applications. Some Modbus-compliant devices utilize an RS-485 asynchronous serial interface for communication. Due to the protocol's minimal hardware and software complexity, Modbus devices and applications are more straightforward and cost-effective compared to other industrial networks [5].

Throughout this study, we investigated the integration of the open-source Python programming language as an HMI/SCADA system to establish communication with the Outseal PLC. The use of Python's adaptability allowed us to effectively design a customized HMI/SCADA system tailored to our specific application needs. Incorporating Python not only enhanced system functionality but also provided the advantages associated with open-source software. Overall, the selection of Python as the HMI/SCADA system proved to be a pragmatic and efficient choice for our application. Then, we have done the training in PENS Surabaya to validate the effectiveness of the developed system.

METHODS

The method of the research is experimental method [6] with the stage detailed in Fig. 1.

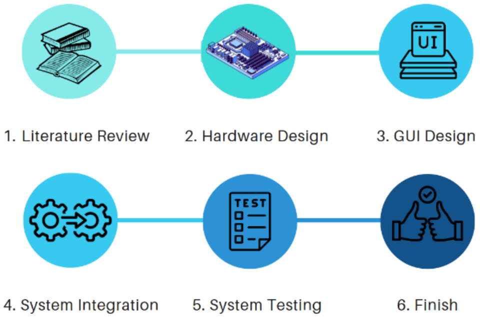

FIGURE 1. Research methods

As shown in Fig. 1, the research began with a review of the relevant literature to determine the requirements. Then, we design the hardware and conduct the wiring in Proteus before having the PCB printed. Python and QML are used to develop graphical user interface (GUI). After completing each component, we integrate them until they can communicate with one another. System testing was carried out in a training session held at PENS Surabaya.

18 July 2025 15:17:36

RESULTS AND DISCUSSION

Hardware Design

Hardware design is done in Easyeda. The schematic diagram can be seen in Fig. 2.

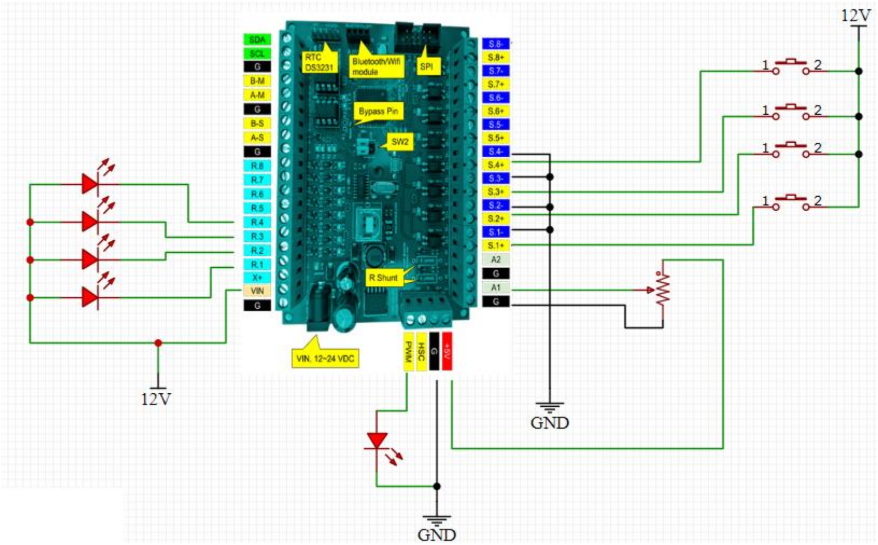

FIGURE 2. Schematic diagram of the system

As shown in Fig. 2, the wiring consists of four sections: digital read, digital write, analog read, and analog write. Digital read is used to determine the button's position, digital write is used to turn the lamp on and off, analog read is used to determine the potentiometer's value, and analog write is used to provide the PWM value for the led.

Following the schematic in Fig. 2, the hardware is can be seen in Fig. 3.

18 July 2025 15:17:36

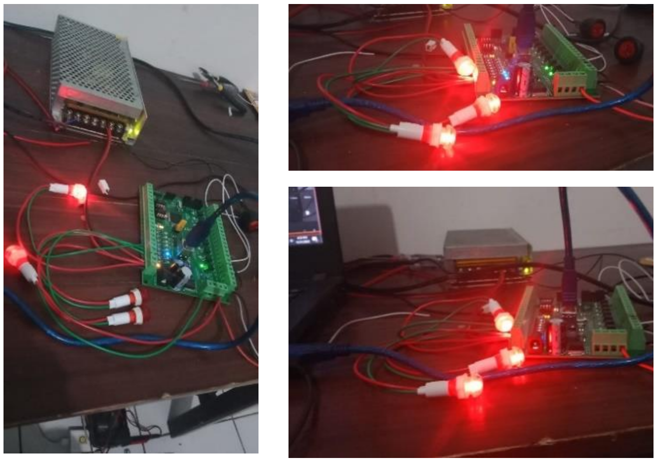

FIGURE 3. Hardware of the system.

GUI Design

The GUI design is done by using Python and QML. There are 3 pages of user interface as we can see in Figs. 4-6.

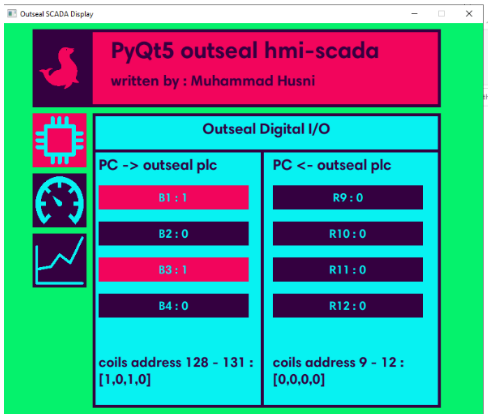

FIGURE 4. Digital I/O user interface.

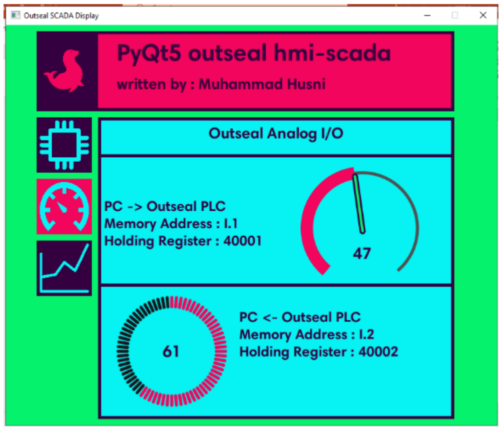

FIGURE 5. Analog I/O user interface.

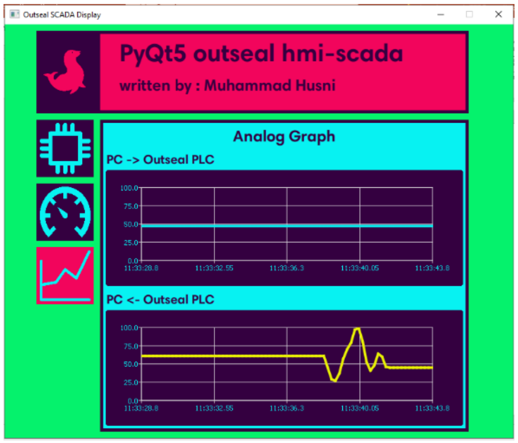

FIGURE 6. Analog I/O graph.

18 July 2025 15:17:36

System Integration

The software works by using QML for the display / user interface, Python for the process, and the PLC ladder program for input/output to the PLC. Python and QML are linked together with pyqtSlot, while Python and the PLC ladder program are linked by Pymodbus. The flowchart of the integrated system is in Fig. 7.

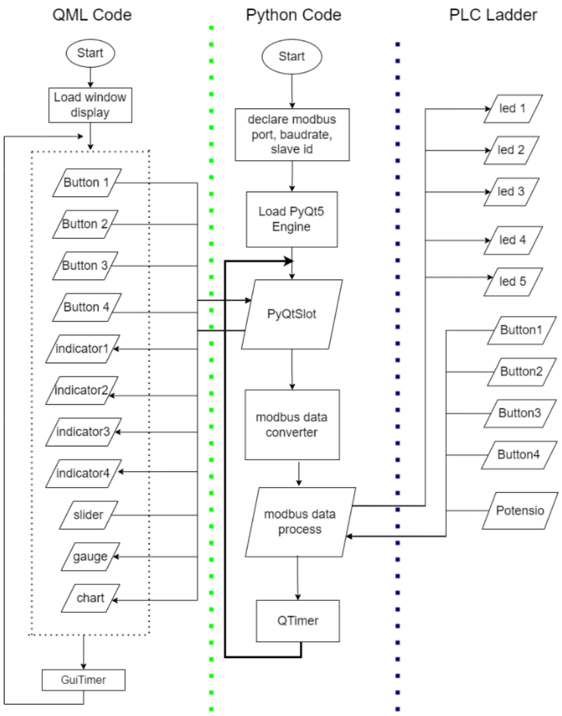

FIGURE 7. Flowchart of HMI SCADA Python system

As shown in figure 7, QML code part represent the GUI, there 4 buttons, 4 indicator, slider, gauge and chart. The python code part is the interface between the GUI and the PLC ladder diagram via Modbus protocol. The ladder diagram can be seen in Fig. 8.

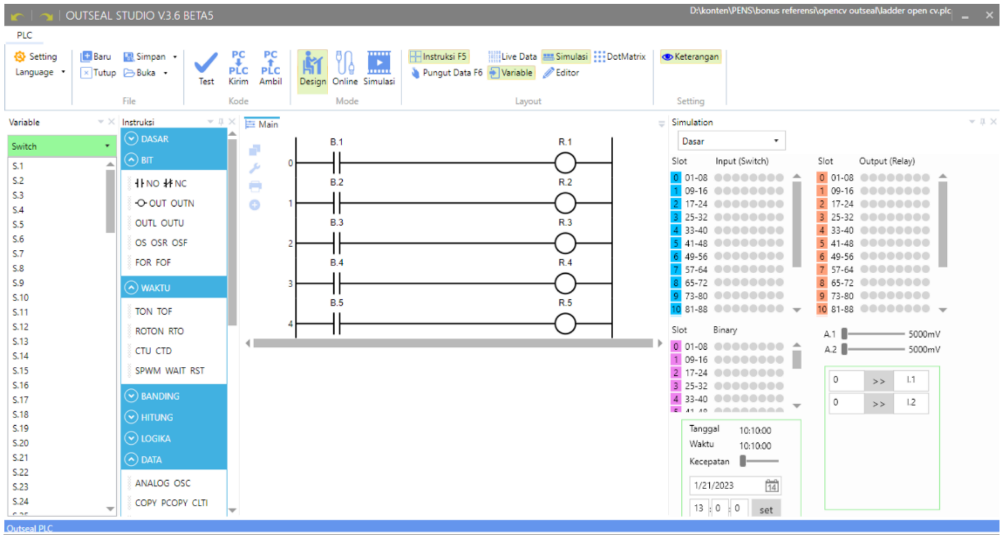

FIGURE 8. Ladder diagram in Outseal Studio.

18 July 2025 15:17:36

Following system testing, the systems are able to communicate with one another. In actual industrial processes, actuators such as push buttons and lights (LEDs) are replaced with industrial-scale actuators. There are 3 ways to issue commands to execute actions (on/off lights):

1. The ladder diagram provides the commands. Python becomes the connection for displaying on the GUI the conditions that occur on the hardware.
2. The GUI (QML code) provides the commands. Using Python as an interface, hardware will respond (lighting will turn on or off) based on the command given.
3. The hardware (push button) issues commands. The lights will switch on / off when the button is pressed. The condition is then sent through Python code, and the results are displayed in real-time on the GUI.

The developed system was evaluated with users during a training session at PENS Surabaya (January 28, 2023) in order to verify its efficacy. The participants were given hands-on experience in developing a simple HMI SCADA system connected to an Outseal PLC. They acquired a greater understanding of HMI SCADA development using Python and QML as a result of this training (Fig. 9).

FIGURE 9. Training session in PENS Surabaya

18 July 2025 15:17:36

After the training was completed, participants were given a questionnaire (Likert scale 1-5, the bigger value, the better) regarding the implementation of the training and the performance of the developed system, the results are shown in table 1.

TABLE 1. Questionnaire results.

No	Questionnaires	Scores				
		1	***2***	***3***	***4***	***5***
1	Participants had the ability to comprehend the material	0%	0%	8.3%	50%	42%
2	This workshop was extremely beneficial to my profession or job.	0%	0%	0%	54%	46%
3	This workshop is highly applicable to the issues I confront at work.	0%	4.2%	12.5%	62.5%	20.8%
4	This workshop taught me how to analyze and solve problems.	4.2%	0%	12.5%	50%	33.3%
5	The instructor mastered the presented material.	0%	0%	0%	58%	41.6%
6	This workshop was presented in a systematic manner.	0%	0%	12.5%	42%	46%
7	Excellent support facilities, including equipment, lodgings, and other technical aspects.	0%	0%	12.5%	50%	37.5%
8	The instructor was attuned to the challenges faced by participants.	0%	0%	0	46%	54%
9	The teamwork and efficacy of learning in this workshop were exceptional.	0%	0%	8.3%	46%	46%

Based on the data in Table 1, the developed system can generally work well and help the participants learn more about HMI SCADA development using Python and QML.

CONCLUSION

Python is a popular programming language that is commonly utilized by technology businesses to construct apps. A Python introduction is required for students and teachers to adapt to technological advances, particularly in the fields of Human Machine Interface (HMI) and Supervisory Control and Data Acquisition (SCADA). This study concentrates on developing an HMI SCADA system using Python and QML. The objective is to integrate this system with Outseal PLC, a commonly used programmable logic controller for industrial automation. The system can provide instructions from hardware, software, and a graphical user interface (GUI), and vice versa. Monitoring system conditions can also be performed in real time. To validate the developed system, it was evaluated with users during a PENS Surabaya training session. The participants were given hands-on experience in developing a simple HMI SCADA system connected to an Outseal PLC. They acquired a greater understanding of HMI SCADA development using Python and QML as a result of this training.

REFERENCES

1. J. Reeser, T. Jankowski, and G.M. Kemper, “Maintaining HMI and SCADA systems through computer virtualization,” 2014 IEEE-IAS/PCA Cem. Ind. Tech. Conf., 1–9 (2014).
2. T.R. Fernandes, L.R. Fernandes, T.R. Ricciardi, L.F. Ugarte, and M.C. de Almeida, in *2018 IEEE PES Transm. Distrib. Conf. Exhib. Am.* (IEEE, 2018), pp. 1–5.
3. P. Pratumsuwan, and W. Pongaen, in *2011 6th IEEE Conf. Ind. Electron. Appl.* (IEEE, 2011), pp. 1477–1481.
4. S.U. Abdi, K. Iqbal, and J. Ahmed, in *2016 IEEE Int. Conf. Ind. Technol.* (IEEE, 2016), pp. 1192–1197.
5. G. Künzel, M.A.C. Ribeiro, and C.E. Pereira, in *2014 12th IEEE Int. Conf. Ind. Informatics* (IEEE, 2014), pp. 446–451.
6. D. Zakaria, T. Gunawan, Y. Somantri, A.G. Abdullah, I. Widiaty, A.B.D. Nandiyanto, N. Amelia, A. Juhana, and W. Arasid, in *IOP Conf. Ser. Mater. Sci. Eng.* (2018).

18 July 2025 15:17:36