Hardware-in-the-loop Simulation of a Programmable Logic Controller, Industrial Robots and Conveyor Systems using RoboGuide

Shivayogi Akki
Dept. of Applied Computing
Michigan Technological University
Houghton, MI, USA
sakki@mtu.edu

Rahul Bondalapati
Dept. of Applied Computing
Michigan Technological University
Houghton, MI, USA
rbondala@mtu.edu

Punyokti Patil
Dept. of Applied Computing
Michigan Technological University
Houghton, MI, USA
pvpatil1@mtu.edu

GurveetSingh Ajmani
Dept. of Applied Computing
Michigan Technological University
Houghton, MI, USA
gajmani@mtu.edu

Nathir Rawashdeh
Dept. of Applied Computing
Michigan Technological University
Houghton, MI, USA
narawash@mtu.edu

***Abstract*—This paper presents an evaluation method for accessing a RoboGuide virtual robot cell and establishing control over a PLC-integrated robot system, with the primary objective of enhancing industrial automation capabilities. The experimental setup encompasses the simulation of a virtual robot cell featuring four Fanucs robots and two part conveyor belts, all controlled through Allen-Bradley PLC-based systems and a Human Machine Interface (HMI). This technological framework enables the testing and validation of control code on simulated industrial systems, ultimately contributing to increased operational efficiency and productivity. The study not only outlines the essential components involved in various industrial sectors but also assesses their efficacy, offering valuable insights into practical applications. Emphasizing the critical role of quality control and assurance in the automation sector for ensuring product standards and safety, the paper describes a simulation created within RoboGuide. In this simulation, the robot and PLC establish full-duplex communication over an Ethernet TCP/IP protocol. A custom control panel is developed, connecting the HMI to the PLC, thereby enabling the execution of robot actions or teaching pendant programs within the simulation environment.**

The core objective of this research is to assess the functionality of an assembly line using hardware-in-the-loop (HIL) simulation prior to the implementation of Factory Acceptance Tests (FAT) and Site Acceptance Tests (SAT) in a real-world industrial setting.

***Index Terms*—Industrial automation, Robot simulation, Programmable Logic Controller (PLCs), Hardware-in-the-loop (HIL).**

I. Introduction

The concepts of PLCs and DCSs are blending in with the trend of industrial automation, which connects multi robotic cell with integrated controllers to enable them to complete tasks on their own. The Master/ root PLC, for instance, manages the entire production process, whereas robots have their own control systems. Industrial flexibility and productivity are increased by multi-robot work cells. An increase in research and demand for a multi-robot system has lately occurred. The study topics for multi-robot systems include collision avoidance, task planning, communication, and performance evaluation. Assembling, welding, and material handling are now more versatile and complex tasks that robots can now perform using advanced sensors, technology, and control systems. Simulation software is used to replicate real-life tasks in a computer program, and simulations are used to test a process virtually and collect the data to predict its working when applied in real life [2]. This step helps prevent flaws in a process and could help increase its efficiency.

Fig. 1. PLC hardware-in-the-loop communication

This research paper delves into an implementation and evaluation study of a mechanism designed to enable a Programmable Logic Controller (PLCs) to access and control a virtual robot system that has been integrated in a simulation software, to advance the system's industrial automation capabilities [3]. The system has been designed and prophecies to simulate and control the robot with the aim of enhancing the efficiency and productivity of any available process automation line. The paper elaborates on the hardware and software components used in the device assembly and thoroughly evaluates the mechanism's effectiveness in meeting its objectives. Holistically, this study provides valuable insights into the real-world industrial application by accessing and simulating various

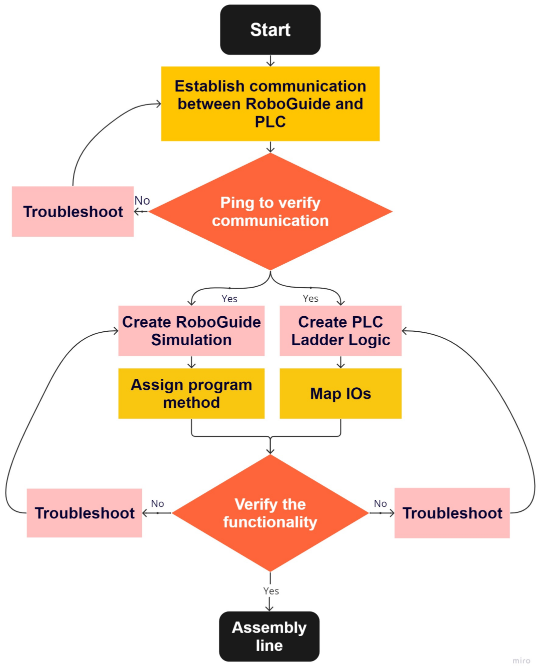

Fig. 2. Project Workflow

parameters for improving industrial automation processes that involve the integration of PLCs with robots (Fig.2). Previous projects published under the same education settings include gripper control [4] [5], robot arm control and interfacing [6] [7] [8], supervisory control and quality assurance [9] [10].

II. Components

A. Programmable Logic Controller

PLC, which stands for Programmable Logic Controller (Fig. 3), is a digital computer that automates industrial processes in harsh environments, such as manufacturing, production lines, and machinery. They are programmed using specialized software and languages, such as ladder logic and function block diagram, and can control and monitor a wide range of industrial applications. PLCs are capable of sequencing, timing, counting, and monitoring, and can communicate with other devices and systems through various communication protocols, including Ethernet, RS-232, and Modbus. Overall, PLCs are an essential component in modern industrial automation, improving the efficiency, reliability, and safety of industrial processes.

B. RoboGuide

RoboGuide (Fig.3) is a powerful simulation and offline programming software developed by FANUC for industrial robots. It allows users to create, simulate, and validate robot programs in a virtual environment, without the need for a physical robot. The software supports a wide range of FANUC robots and third-party devices and includes a programming tool that allows users to create, edit, and debug robot programs in various programming languages. Users can also use

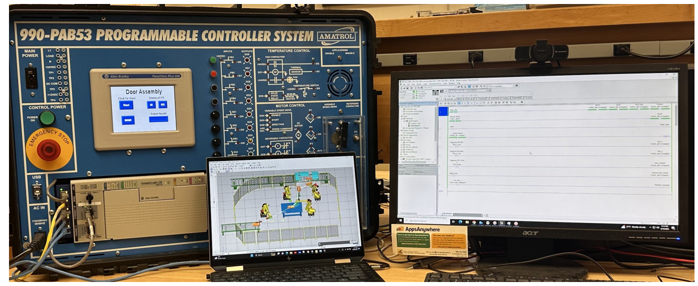

Fig. 3. Project Setup with PLC, HMI and Simulator

RoboGuide to generate code for the robot controller, which can then be transferred to the physical robot. Overall, RoboGuide is a valuable tool for improving the efficiency and productivity of industrial robot applications by allowing users to validate robot programs and optimize robot performance in a safe and cost-effective manner.

C. Ethernet IP

Ethernet is a widely used computer networking technology that allows devices to communicate with each other over a local area network (LAN) or wide area network (WAN). It was developed in the 1970s by Xerox Corporation and later standardized by the Institute of Electrical and Electronics Engineers (IEEE). Ethernet uses a packet-switched protocol to transmit data between devices. Each device on the network is assigned a unique Media Access Control (MAC) address, which is used to identify and route data packets. Ethernet also uses a carrier sense multiple access with collision detection (CSMA/CD) algorithm to manage network traffic and prevent collisions between packets.

Ethernet supports a variety of transmission speeds, ranging from 10 megabits per second (Mbps) to 100 gigabits per second (Gbps), depending on the type of Ethernet cable and network equipment used. The most commonly used Ethernet cables are twisted pair copper cables and fiber optic cables. Ethernet is widely used in both commercial and residential settings to connect devices such as computers, printers, routers, and servers to a network. It is also used in industrial applications to connect machines and equipment, and in telecommunications to provide high-speed data transfer over long distances. Overall, Ethernet is a reliable and scalable technology that has become a fundamental component of modern computer networking.

D. Ethernet HUB

An Ethernet hub is a basic networking device that operates at the physical layer of the OSI model and allows multiple devices to connect to a network. It has multiple Ethernet ports, which can be used to connect computers, printers, servers, and other devices to a network. However, when a device sends data to the hub, the hub broadcasts the data to all other connected devices, regardless of whether they are the intended recipient or not. This can lead to network congestion and reduced network performance, especially in larger networks with many connected devices. Ethernet hubs have largely been replaced by Ethernet switches, which provide more advanced features such as traffic management, quality of service (QoS), and virtual LAN (VLAN) support. However, hubs can still be useful in certain situations where a simple and inexpensive way to expand a network is needed.

E. Human-Machine Interface

HMI or Human-Machine Interface (Fig.4) is a user interface that allows humans to interact with machines and control systems, commonly used in a variety of applications from industrial automation and control to consumer electronics. It typically consists of a display screen and input devices such as touchscreens or buttons, and software that enables users to monitor and control the system. In industrial automation, HMIs are used to monitor and control machines and processes, display real-time data, adjust machine settings, and respond to alarms and alerts. In consumer electronics, HMIs are used in products such as smartphones and tablets, allowing users to interact with the device through touchscreens and voice commands. HMIs have become an essential component of modern control and automation systems, providing users with an intuitive and easy-to-use interface that improves efficiency and productivity.

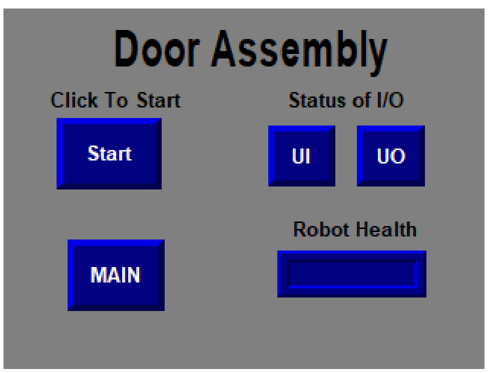

Fig. 4. Human Machine Interface

III. Communication

Communication needs to be established between the PLC and RoboGuide using Ethernet/IP cable. In order to establish strong communication, the PLC and RoboGuide need to be configured. Step one, connect the PLC with the RoboGuide, and the PLC with RSLogix 5000 using Ethernet/IP Cables and configure the IPv4 address (usually, 192.168.10.xxx) of all systems to have the same Network ID but different Host ID while the Subnet Mask is 255.255.255.0.

A. With respect to the PLC

To establish communication between the PLC and the robot, first, create an Ethernet/IP module in RSLogix 5000 and define it as a generic Ethernet/IO module. Next, enter the IP address of the robot in the RoboGuide and define the assembly instances and size of the input and output configurations. This will create input and output tags on the PLC end, allowing for communication with the robot, refer to Fig.5.

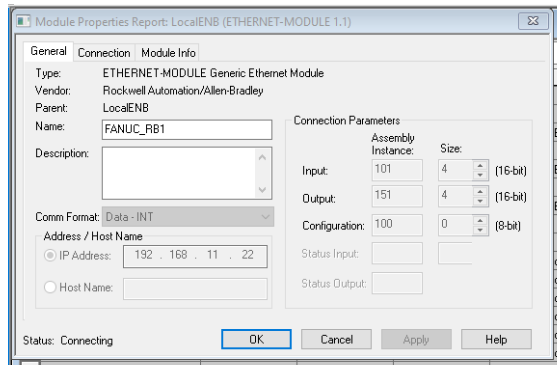

Fig. 5. RSLogix Creating Ethernet Module [15]

B. With respect to RoboGuide

In order to add a robot to the simulation and establish a connection with the PLC, the robot's IPv4 address should be configured to the same PC's IPv4 address. This can be done in the teach pendant by going to Menu >Setup >Host communication >Configure. Next, in RoboGuide, the input and output sizes from the robot end should be defined by going to Menu >I/O >Ethernet/IP >Adaptor configuration. Then, in Menu >System >Program select >Config, the UI signals should be changed to true, I/O to Simulated I/O, and the operator panel setting to remote. Additionally, the program select mode should be changed to OTHER and the start method to OTHER. For Ethernet connection, the default rack is 89 and the slot number is set to 1, since the I/O's are sent and received via Ethernet/IP cable. Finally, the I/O ranges should be set according to the requirement. In Menu >select 0 >System >Variables >select Item and enter 600 >change $ RMT_Master to "0" from "2" to make the slave because the PLC is the master. Performing these steps establishes a connection between the PLC and the robot in the RoboGuide software.

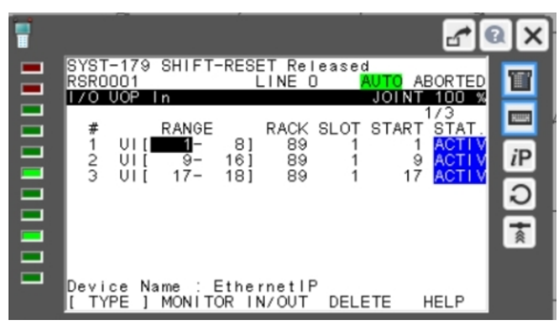

Fig. 6. RoboGuide Setup

IV. Programming

A. I/O Mapping

After assigning the bits (Fig.6), the first 20 bits go for the UOP which are the default bits that are assigned by Fanuc. The remaining bits are assigned to the Digital Inputs. These are the inputs for RoboGuide and the Outputs of the PLC. **UO-** Like the UIs, the UOs have 20 predefined bits from Fanuc which are the outputs of RoboGuide and the inputs to the PLC. The Remaining bits are assigned to the Digital Inputs as per the requirements. In UI, I MSTP, Hold, SFSPD, and Enable are 4 bits that should always be ON in order for the simulation to work.

B. Production start Methodologies

1. PNS - PNS stands for Program Number Selection. The PNS function allows for the selection or examination of a program through the utilization of both the program number selection signals (PNS1 to PNS8 PNSTROBF) and the START signal, as shown in Fig.7. During the temporary halt or execution of a program, these signals are disregarded.

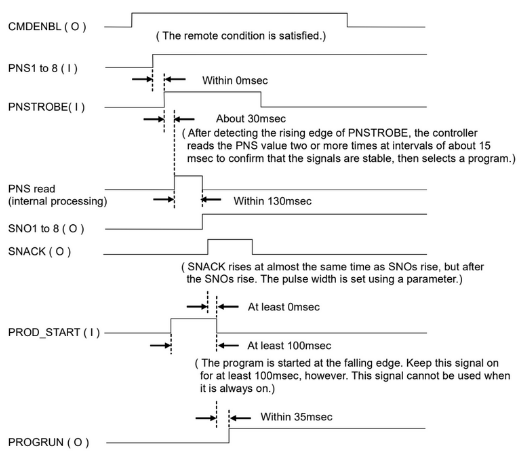

Fig. 7. The sequence of automatic operation by PNS

2. RSR - The RSR function utilizes the robot service request signals (RSR1 to RSR8 inputs) to choose and initiate a program. If a different program is already in the process of execution or has been temporarily halted, the selected program will wait until the currently running program is complete before starting, as shown in Fig.8.

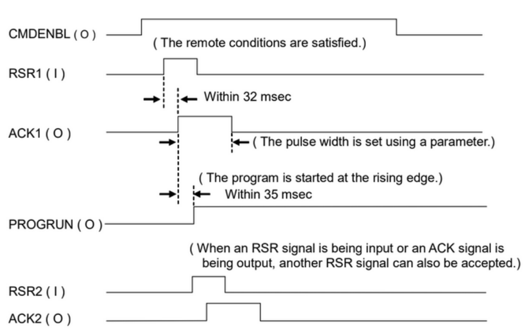

Fig. 8. The sequence of automatic operation by RSR

3. Style - To choose or organize a program, the remote controller employs the STYLE function. By utilizing the input signals STYLE1 to STYLE8, a specific program number can be designated for the desired STYLE. Prior to utilizing the STYLE function, programs must be assigned to each STYLE number in advance. Unlike PNS and RSR, program names are not limited when using the STYLE function.

Fig. 9. Car Door Assembly Line

3. OTHER - This option is best when the requirement is to run one main program repetitively. This simplifies the process of assigning the program and enables the use of DI and DO.

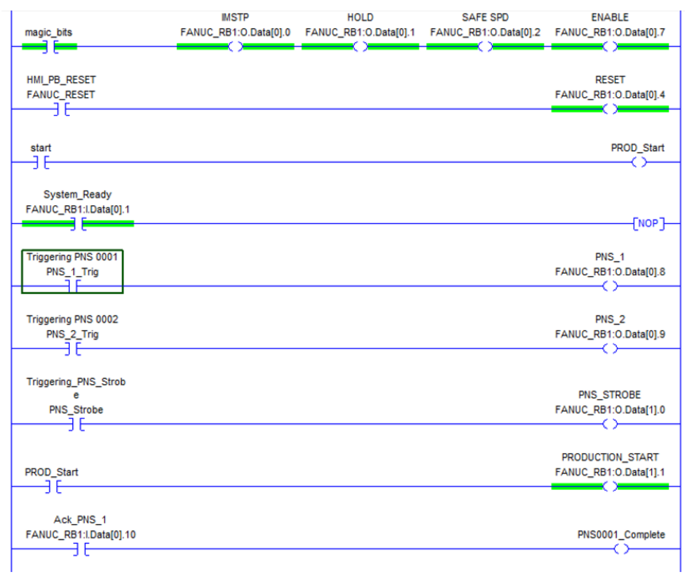

Fig. 10. Ladder Logic program

C. Simulation

Car door assembly line simulation, shown in Fig.9 is developed in RSLogix 5000. This assembly line comprises three robots, two conveyor lines, and a protected work cell. 2 major forms of action take place: Pick and place, and welding. The robot used is R-2000iB/165F. Note: As a safety precaution, the fault reset is not bypassed to the PLC. Shift+reset needs to be manually pressed in the Teach pendant in the initial stage of the operation.

D. Ladder Logic and HMI

The ladder logic program is developed to operate the simulation using the Production_Start command. Rung 0 is used to switch on the 4 main bits (IMSTP, HOLD, SAFE SPD, and ENABLE bit), refer to Fig.10. The program also has the option to switch from OTHER to PNS if required. Human-Machine Interface is created using Factory Talk. A basic HMI is to be created with consideration to the Home screen, Emergency Stop Button, Safety Alarms, and Program Start Button.

V. Results and Future Scope

This paper advances upon the implementation of a Hardware-in-Loop assembly line simulation utilizing Programmable Logic Controllers (PLC) and RoboGuide, with a primary focus on evaluating the efficiency and safety parameters of the assembly line. The findings generated by this simulation analysis not only hold the potential for substantial cost-effective improvements but also features the system's practical applicability, as evidenced in our successful implementation within the car door assembly line.

In anticipation of future advancements in this domain, several crucial features can be considered for integration. A paramount consideration is the configuration of PLC and Human Machine Interface (HMI) tags, facilitating remote access. This would include the use of "consumed" and "produced" tags on the PLC, streamlining program navigation and HMI screen usage, thereby offering enhanced troubleshooting capabilities for the entire cell. This innovation paves the way for the implementation of additional features, including the integration of a manual mode to facilitate manual cell movements via the HMI interface, an exhaustive examination of multiple operational modes (Auto Mode, Manual Mode, Semi-automatic, and Dry Cycle), and the inclusion of data acquisition into Excel sheets to bolster data analysis and management.

Furthermore, ensuring the utmost safety and operational integrity, a pivotal aspect is the comprehensive evaluation of robot safety through the HMI interface, accompanied by real-time monitoring through an overview screen. The ability to address robot warnings directly from the HMI is a testament to the system's robustness, offering quick solutions to issues such as missed part pickups leading to production halts. Keeping production on track is achieved through a count screen (FIS screen), while a bypass mechanism is essential for maintaining workflow in situations involving problematic robots.

The functionality should encompass the assessment of recovery operations in both auto and manual modes using a dedicated Recovery Push Button (PB). Manipulating part data, including the determination of the appropriate part style to run within the cell, adds a layer of versatility and adaptability. Monitoring the robot's status directly through the HMI's robot control screen is imperative. The complete control of the entire cell's functions and recovery processes via the HMI interface forms the core of an efficient and responsive system.

To further elevate the system's safety credentials, an Emergency Stop (E-stop) functionality on the HMI screen should be incorporated, enabling the simulation and testing of E-stop scenarios, a critical component of safety tests. Additionally, the ability to manipulate sensor and safety component data for the evaluation of the cell's safety functions is a hallmark of a well-rounded and secure operational environment.

In summary, the proposed enhancements not only bolster the functionality of the assembly line simulation but also underscore its relevance and potential in a broader industrial context, representing a significant step toward safer, more efficient manufacturing processes.

Acknowledgment

This work was financially supported by the College of Computing and the Institute of Computing and Cybersystems (ICC) at Michigan Technological University.

References

[1] Park, Sang C., and Minho Chang. "Hardware-in-the-loop simulation for a production system." International Journal of Production Research 50, no. 8 (2012): 2321-2330.

[2] Suresh, Jayasuriya. "Integration of SCARA Robot for Pick and Place Application using PLC." In Proceedings of the First International Conference on Combinatorial and Optimization, ICCAP 2021, December 7-8 2021, Chennai, India. 2021.

[3] Iacob, Mihai, and Gheorghe-Daniel Andreescu. "Real-time hardware-in-the-loop test platform for thermal power plant control systems." In 2011 IEEE 9th International Symposium on Intelligent Systems and Informatics, pp. 495-500. IEEE, 2011.

[4] Kocher, Erik, Chukwuemeka George Ochieze, Ahmat Oumar, Travis Winter, Aleksandr Sergeyev, Mark Gauthier, and Nathir Rawashdeh. "A Smart Parallel Gripper Industrial Automation System for Measurement of Gripped Work Piece Thickness." In Conference for Industry and Education Collaboration 2022. 2022.

[5] Piechocki, Brian, Chelsey Spitzner, Namratha Karanam, Travis Winter, Aleksandr Sergeyev, Mark Gauthier, and Nathir Rawashdeh. "Operation of a Controllable Force-sensing Industrial Pneumatic Parallel Gripper System." In Conference for Industry and Education Collaboration 2022. 2022.

[6] Liu, Zongguang, Chrispin Johnston, Aleksi Leino, Travis Winter, Aleksandr Sergeyev, Mark Gauthier, and Nathir Rawashdeh. "An Industrial Pneumatic and Servo Four-axis Robotic Gripper System: Description and Unitronics Ladder Logic Programming." In Conference for Industry and Education Collaboration 2022. 2022.

[7] Reyes, Abel A., Skylar Reinhardt, Tony Wise, Nathir Rawashdeh, and Sidike Paheding. "Gesture Controlled Collaborative Robot Arm and Lab Kit." In Conference for Industry and Education Collaboration 2022. 2022.

[8] Salameen, Laith, Abdelkarim Estatieh, Salman Darbisi, Tarek A. Tutunji, and Nathir A. Rawashdeh. "Interfacing Computing Platforms for Dynamic Control and Identification of an Industrial KUKA Robot Arm." In 2020 21st International Conference on Research and Education in Mechatronics (REM), pp. 1-5. IEEE, 2020.

[9] Valaboju, Charan, Manikanth Amuda, Sri Charan NCh, Sushvan Reddy Gopireddy, Paniz K. Hazaveh, and Nathir A. Rawashdeh. "A Supervisory Control System for a Mechatronics Assembly Station." In 2023 15th International Conference on Computer and Automation Engineering (ICCAE), pp. 503-507. IEEE, 2023.

[10] Nandre, Chetan, Edwin Yazbec, Prathamesh Urunkar, Sourish Motey, Paniz Hazaveh, and Nathir A. Rawashdeh. "Robot Vision-based Waste Recycling Sorting with PLC as Centralized Controller." In 2023 15th International Conference on Computer and Automation Engineering (ICCAE), pp. 381-384. IEEE, 2023.

[11] Bonfe, Marcello, Matteo Vignali, and Mario Fiorini. "PLC-based control of a robot manipulator with closed kinematic chain." In 2009 IEEE International Conference on Robotics and Automation, pp. 1262-1267. IEEE, 2009.

[12] Anterszlak, Damian. "PLC controllers in robotic arc welding systems." PhD diss., Instytut Technik Wytwarzania, 2022.

[13] Kajan, Martin, Ján Šovčik, František Duchoň, Leo Mrafko, Martin Florek, and Peter Beňo. "Sensoric subsystem of automated guided vehicle: TCP communication between SIMATIC S7 PLC and Arduino." In 2014 23rd international conference on robotics in Alpe-Adria-Danube Region (RAAD), pp. 1-6. IEEE, 2014.
[14] Hashim, Hairulazwan, and Zainal Alam Haron. "A study on industrial communication networking: ethernet based implementation." In 2007 International Conference on Intelligent and Advanced Systems, pp. 1111-1114. IEEE, 2007.
[15] Wendt, Arne, and Thorsten Schüppstuhl. "A Solution to the Generalized ROS Hardware IO Problem-A Generic Modbus/TCP Device Driver for PLCs, Sensors and Actuators." In 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1-8. IEEE, 2021.
[16] Rajapakse, B., and N. Hildreth. "Remote access toolkit for industrial robots." Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 218, no. 8 (2004): 1017-1022.
[17] Gang, Huang, Yu Zhilou, Chen Naikuo, Feng Guirong, Zhang Lei, Hu Boyi, and Qin Shiming. "Development of OPC UA based centralized server Fieldbus data high efficiency transmit architecture." In 2020 39th Chinese Control Conference (CCC), pp. 4580-4585. IEEE, 2020.