11-2022

An Industrial Pneumatic and Servo Four-axis Robotic Gripper System: Description and Unitronics Ladder Logic Programming

Zongguang Liu
Michigan Technological University, zliu9@mtu.edu

Chrispin Johnston
Michigan Technological University, cjohnsto@mtu.edu

Aleksi Leino
Michigan Technological University, afleino@mtu.edu

Travis Winter
Donald Engineering

Aleksandr Sergeyev
Michigan Technological University, avsergue@mtu.edu

See next page for additional authors

Recommended Citation
Z. Liu, C. Johnston, A. Leino, T. Winter, A. Sergeyev, M. Gauthier, N. Rawashdeh, "An Industrial Pneumatic and Servo Four-axis Robotic Gripper System: Description and Unitronics Ladder Logic Programming", 2022 American Society for Engineering Education (ASEE) Conference for Industry and Education Collaboration (CIEC), Tempe, AZ Feb. 4-11 2022, ASEE. Retrieved from: https://digitalcommons.mtu.edu/michigantech-p/16522

Follow this and additional works at: https://digitalcommons.mtu.edu/michigantech-p

Part of the Computer Engineering Commons, Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Authors

Zongguang Liu, 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

Zongguang Liu[a], Chrispin Johnston[a], Aleksi Leino[a], Travis Winter[b], Aleksandr Sergeyev[a], Mark Gauthier[b], Nathir Rawashdeh[a]

[a] Michigan Technological University, [b] Donald Engineering

Abstract

As part of the advanced programmable logic controllers (PLC) course at Michigan Tech, this class project is performed on a mechatronics system gifted by Donald Engineering, a Michigan-based supplier of industrial automation systems and components. This paper explores the functionality and ladder programming of a four-axis robot enclosed in a cage with one side guarded by an optical fence. The robot has pneumatically actuated X-Y linear motion and a pneumatic gripper. Furthermore, the Z-axis motion and gripper wrist rotation are controlled by servo motors. A human machine interface (HMI) is also present, and it allows for easy manipulation and programming of the robot. This type robot can be used to transfer small components between conveyer belts or for light assembly functions. This paper details of the system's components, operation, and custom programming.

Literature Review

A programmable logic controller (PLC) is a rugged computer used to track and automate manufacturing processes in a variety of industries [1]. A PLC consists of four main parts: input and output (I/O) modules, a power source, and a CPU. Through software that has access to I/O modules, a PLC provides control and monitoring of equipment and enters the programming application via addresses. PLCs are utilized in a variety of different applications, especially in industry, and offer several benefits, the most significant of which are the ability to reduce costs and lower the time for production. In theory, a PLC is necessary for any application that needs electrical control [2]. Automation has improved productivity while also reducing variances in manufactured parts, resulting in higher quality standards. Flexible and adaptable manufacturing concepts that merge imaging and motion with industrial robots are an ideal teaching tool for the field of "Mechatronics," which encompasses mechanism design and analysis, soft computing, sensing, and electromechanical systems [3].

Controlling and programming the movements of a robot's mobile elements is critical, particularly when the application calls for constructive simplicity and ease of implementation [4]. Each input's status is read at the beginning of the program [5]. The current value of each input is read and stored during the operating cycle, i.e., a scan, of the PLC. It collects data from input mechanisms or sensors that are connected and uses that information to trigger an output from parameters that are pre-programmed [1, 6]. There needs to be a proper alignment of all the component parts in order to achieve accurate operating and sequencing of the device [7]. A function block diagram (FBD), an instruction list (IL), a ladder diagram (LD), structured text (ST), and a sequential function map (SFC) comprise the various languages for programming that are utilized for PLCs up to the current time [8]. The LD, the most well-known among these various programming languages, is a graphical symbolic language [8]. Examining the system's proper operation in accordance with the LD design requirements is challenging. More specifically, the LD is difficult to comprehend and check for correctness, which results in an absence of versatility in control systems found in industrial environments. In this regard, research has been conducted to develop the processes that use LDs in order to make them more versatile and compatible [8]. Discrete event models (DEMs) such as Petri nets, state diagrams, and others have been used in several of these methods [8, 9]. A PLC can track and log run-time data, e.g., system efficiency or operating temperature, and automate the initiation and termination of processes-based output and input data. PLCs are versatile and reliable control systems that can be adapted to most applications. Usually, after writing the program for a PLC on a computer the program is subsequently downloaded to the controller. Ladder logic programs are available in most programming applications used by PLCs. The programming language that has been commonly used is ladder logic.

Robot Station

The Pneumatic Robot is designed to pick and place objects. The dimensions that the Pneumatic Robot works within are limited, so it cannot be used in tandem with other devices that require a lot of workspace to complete their tasks. An additional limitation for this Pneumatic Robot is that it has fixed movements so that it cannot stop at any random location within the working area. The most appropriate circumstance to use this robot would be to transfer parts from one conveyor to another conveyor.

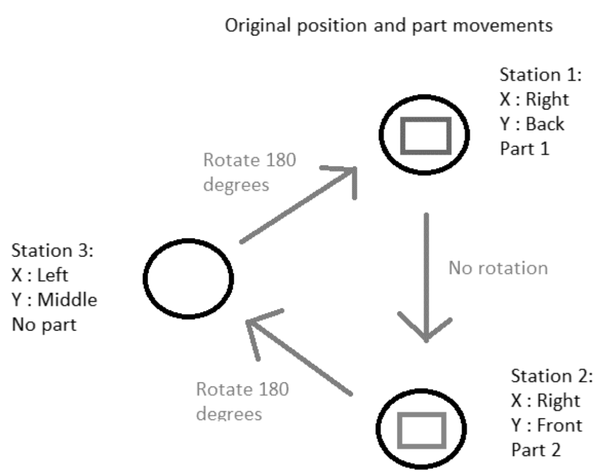

Figure 1. Pick and place project spatial summary

The Pneumatic Robot was designed to move two parts (Part 1, Part 2), piece by piece, around three designated points (Station 1, Station 2, Station 3) as shown in Fig. 1. The Pneumatic Robot will move Part 2 from Station 2 (X: Right/Y: Front) to Station 3 (X: Left/Y: Middle). The Pneumatic Robot will move Part 1 from Station 1 (X: Right/Y: Back) to Station 2. Part 2 will be moved from Station 3 to Station 1, Part 1 will be moved from Station 2 to Station 3, Part 2 will be moved from Station 1 to Station 2, and finally Part 1 will be moved from Station 3 to Station 1. After the program finishes executing, both Part 1 and Part 2 should be in their original positions with the original orientation. To complete a full rotation of Part 1 and Part 2 through the three stations requires a total of six movements.

System Components

Sensors

The pneumatic robot system included the following sensors:

- Ultrasonic sensors for x-and y-axis
- Inductive proximity sensor for z-axis
- Magnetic sensor for gripper rotation

Robot Motion

The Pneumatic Robot has 4 degrees of freedom: left-to-right motion in the x-axis, forward/backward motion in the y-axis, upward/downward motion in the z-axis, and the end effector has the ability to rotate 180 degrees clockwise (CW) and counterclockwise (CCW). Fig. 2 illustrates the robot moving parts. The motion in the x-axis, the y-axis, and the end effector is pneumatically driven. The motion in the z-axis is electrically driven. If the robot is programed to move from the forward position to the middle position in the y-axis, it will reach the rear position first and then move to the middle position. The Pneumatic Robot can achieve only fixed positions (2-X, 3-Y, 2-Z, CW/CCW, gripper closed/open) and no positions in-between. Table 1 summarizes the robot's motion patters.

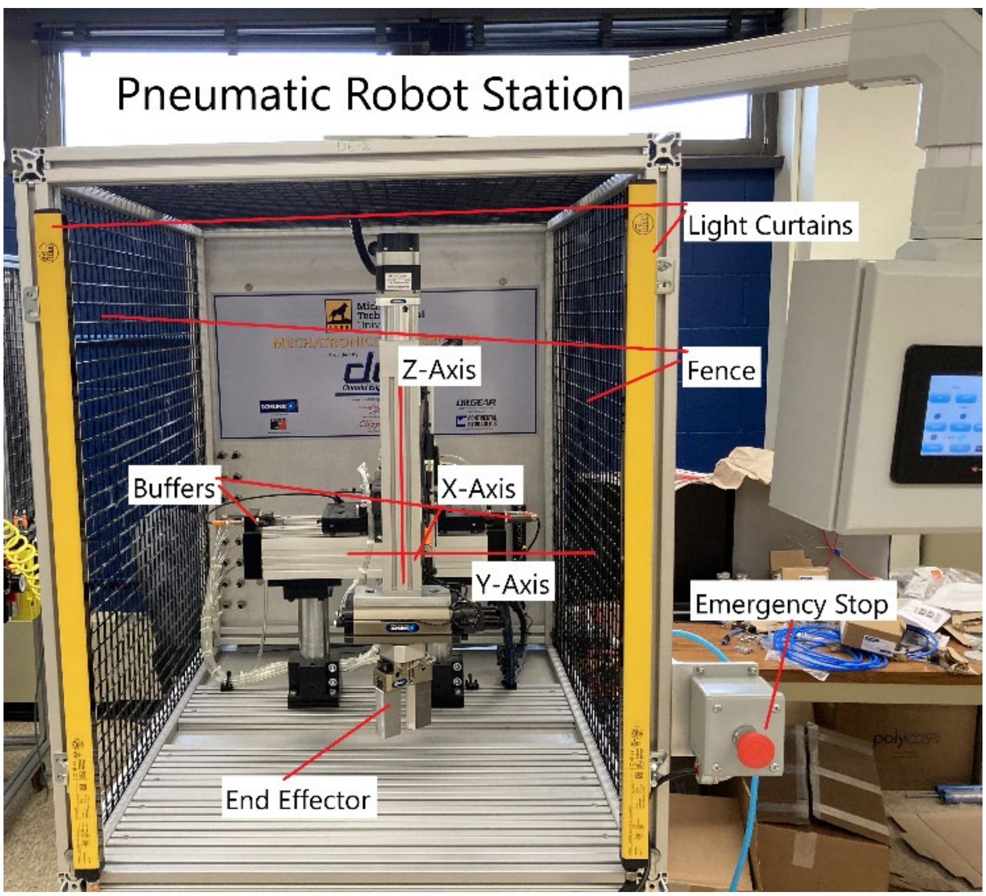

Figure 2. Pneumatic robot station components

Table I. Motion Summary

X-axis	Left	Right	
Y-axis	Rear	Middle	Forward
Z-axis	Up	Down	
End-effector motion	Rotation (CW, CCW) for 180 deg	Close/open	

Safety Features

Industrial robots contain several features designed to protect operator safety on the factory floor. Some feature may not be obvious, such as infrared light curtains. The robot system used in this work has to following safety features:

- Aire fence: Located on both sides, top, and at the back of the robot station.
- Light curtain: Located at the very front of the robot station. If the light barrier is broken, the safety exhaust valve located on the right of the robot station will release the air out of the system. All pneumatic motion will cease in the x and y axes, as well as in the end effector's ability to rotate.
- Emergency stop button: Locate on the front right side of the robot station. When the emergency button is pressed, the power supply will be cut, and all motion will cease. Light curtain invasion only stops motion in the axes that are pneumatically driven (x & y axes) but not in the axis that is electrically driven (z-axis).
- Buffers on either end of the x-axis: Rapidly decelerate the robot's movement and aid in absorbing the impact of the robot when it crashes into the ends of the rail that allows the robot to move side to side.

Control and Power Components

The robot included the following electrical and mechanical components highlighted in Fig 3 and listed here:

- Unitronic PLC (USC-B5-B1): 16+16 digital input/output; 16 analog input; 16 relay input.
- Power supply
- Pneumatic control/supply
- Mechanical gripper
- Stepper motor drivers

Robot HMI

On the human machine interface control touch panel, shown in Fig. 3, the operator can move the robot by pressing the related commands. The M35 must be pressed in order to energize the pneumatic supply.

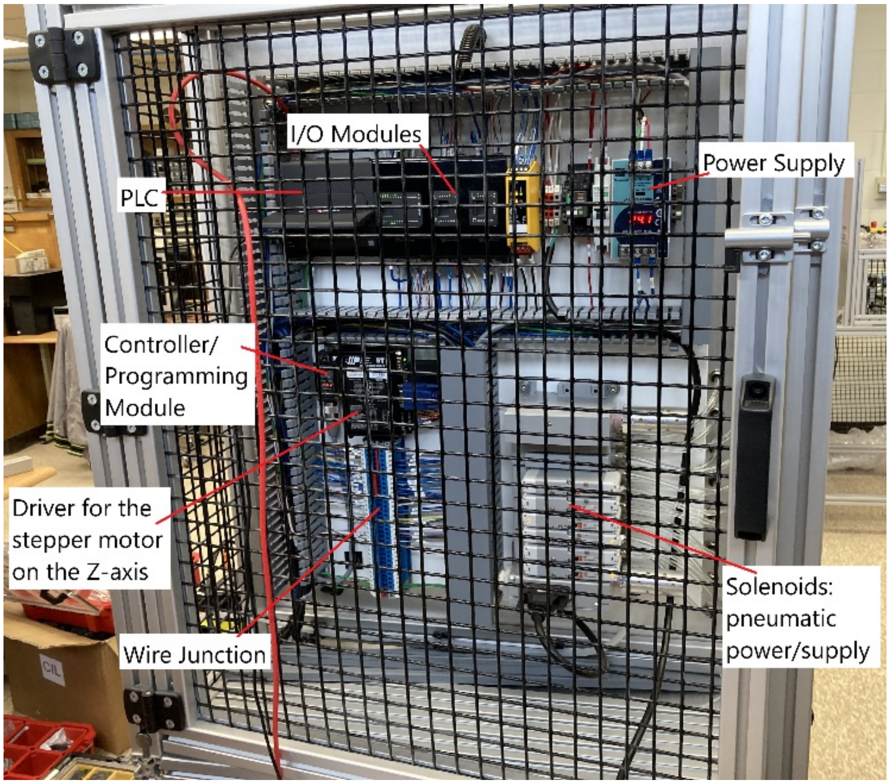

Figure 3. Pneumatic robot station control components.

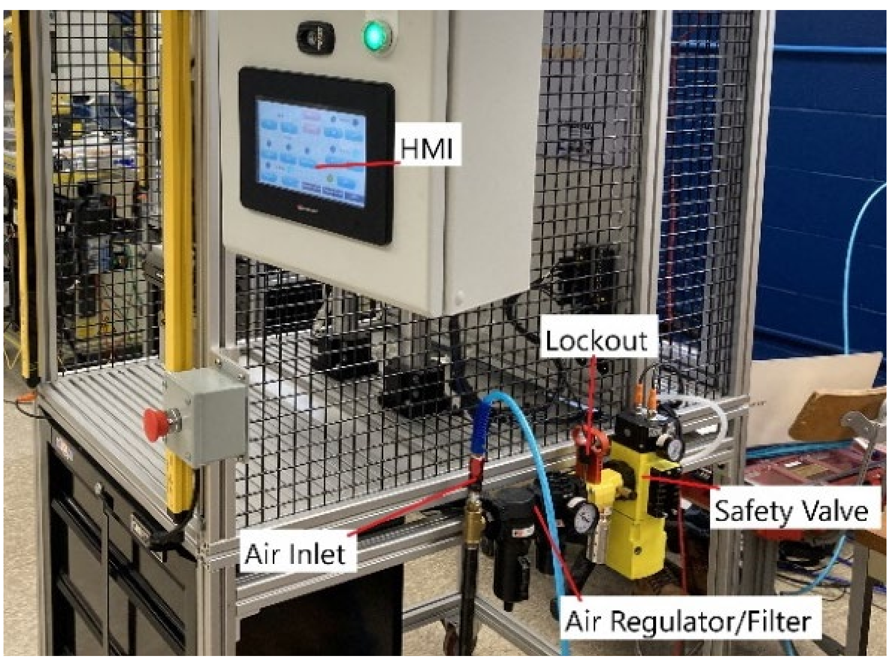

Figure 4. Robot station pneumatic components.

Ladder Logic Programming

The Unitronic programmable logic controller was programmed using ladder logic, as shown in Figures 5 through 10. Programs were written using UniLogic's ladder logic to perform simple tasks, nevertheless became quite complex to write and follow. The program had to be written in such a way that all output functions were used only once in the program, which meant that single rungs became very complex. UniLogic's's ladder logic was appropriate for the purpose of programming a pneumatic pick-and-place robot but other programming ladder logic languages should be considered as complexity of tasks increases.

Figure 5 illustrates the start rung including lap counters. The button M35 is used as the start button for the program. Two different positions determine the lap counting. Two conditions are defined for lap counting. P is positive transition contact which works like one-shot rising. If the condition satisfied once, the output will remains true.

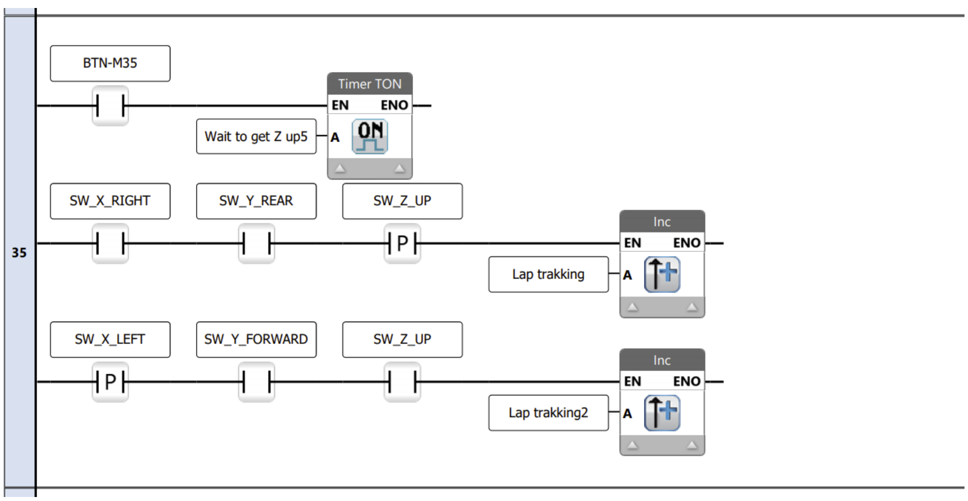

Figure 5. Ladder logic section 35: start and counters.

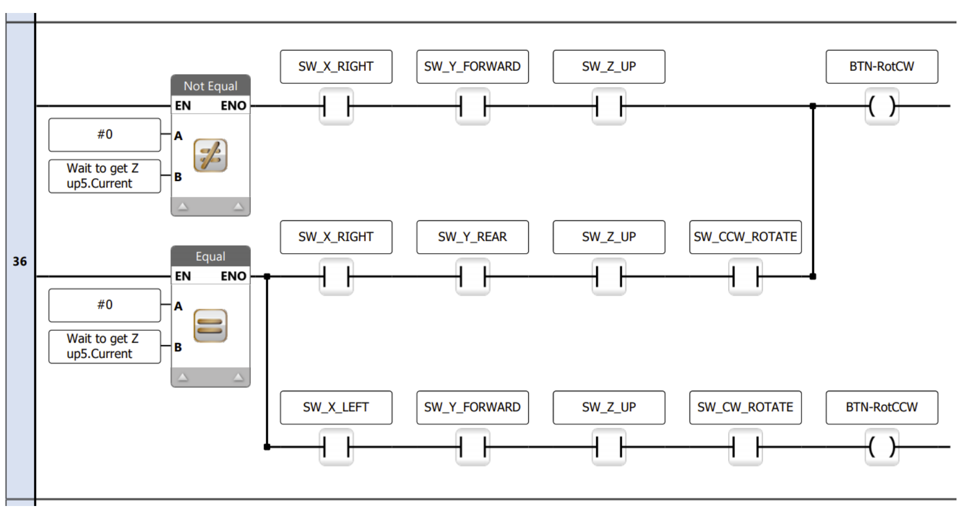

Figure 6. Ladder logic section 36: rotations.

In fig. 6, the ladder logic for determining the rotation of the gripper by different conditions, is shown. All sub-rungs with the *not equal* block are used for redirecting the robot to the home position (right forward up CW open) when the M35 button pressed.

Fig.7 shows the ladder logic section for gripper open and close conditions. The reason for using increment and decrement at the end of each rung is that the gripper should change the condition for the next lap. For example, the current gripper will pick up (close) item from station 1 but it should drop off (open) item to station 1 in the next lap. At the end of rungs, another counter gets involved for the robot to make decision on gripper conditions in different laps. If the gripper open once the counter will minus 1 and if the gripper closed, the counter will plus 1.

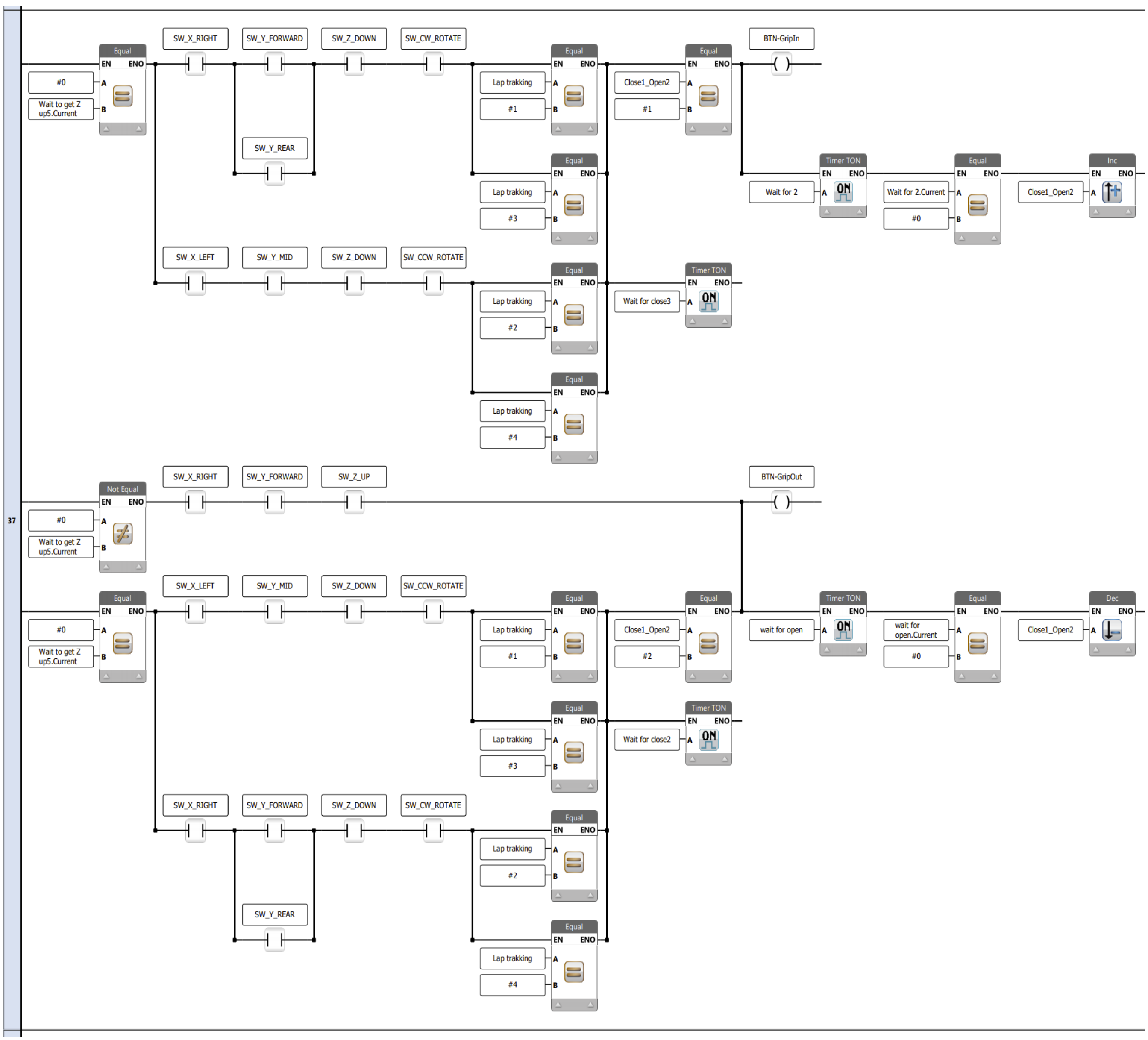

Figure 7. Ladder logic section 37: gripper in/out (+ activating Z up timers)

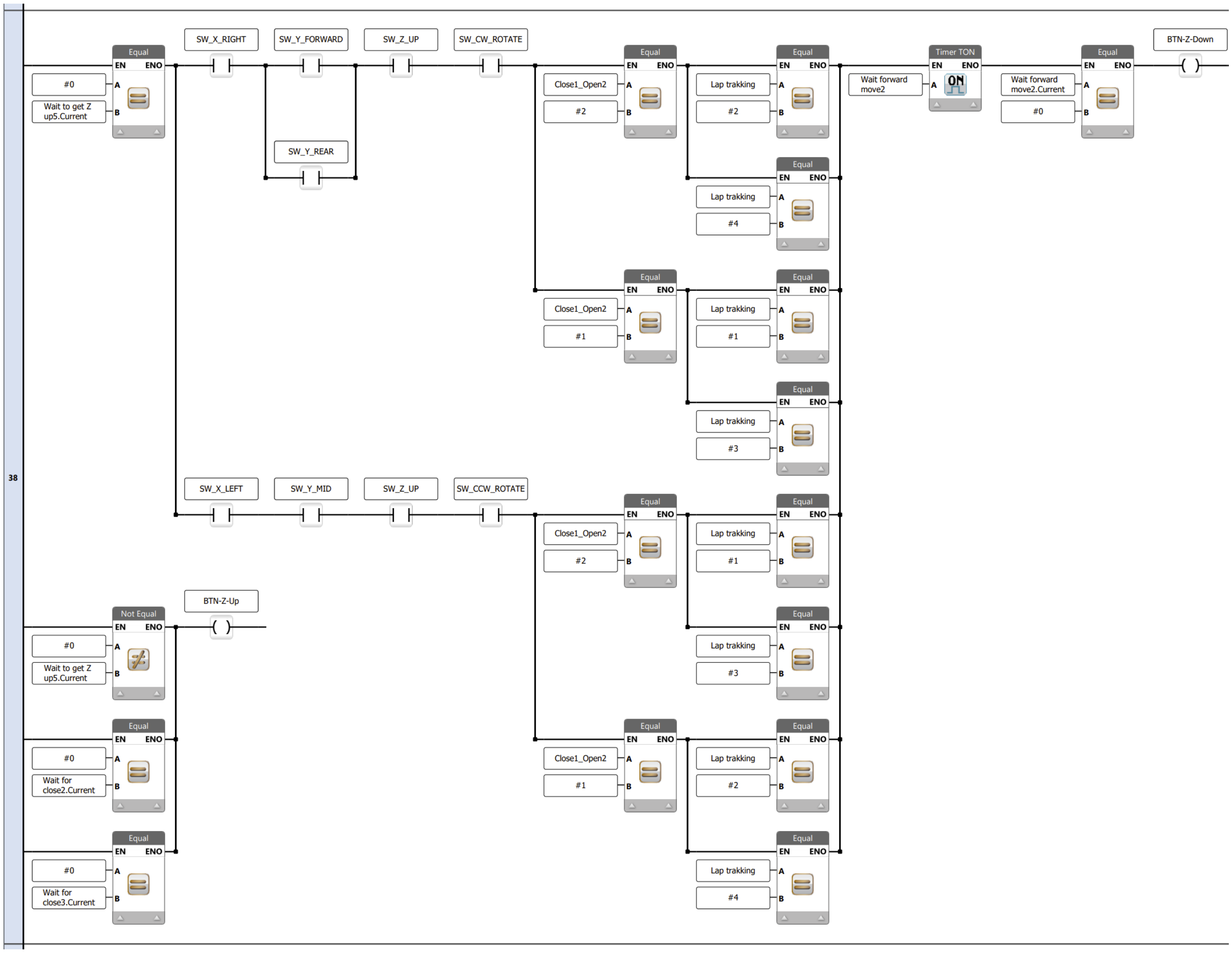

Figure 8. Ladder logic section 38: Z-Movements

The logic for programming the conditions for Z movements is shown in Fig. 8. The timer at the end of sub-rungs is waiting for the robot to move forward, otherwise the robot arm may move down when the position is *rear* and satisfy other conditions and keep moving up and down in the *rear* position.

In Fig. 9, he logic defining all conditions for robot's movements on the X-axis is shown. All the *equal* blocks are used for letting the robot determine movements based on the lap and the gripper condition.

Similarly, the definition of all conditions for robot's movements on Y-axis are shown in Fig. 10. All comparison function for the whole program, are used for identifying different laps since the robot will perform differently in different laps even in the same station.

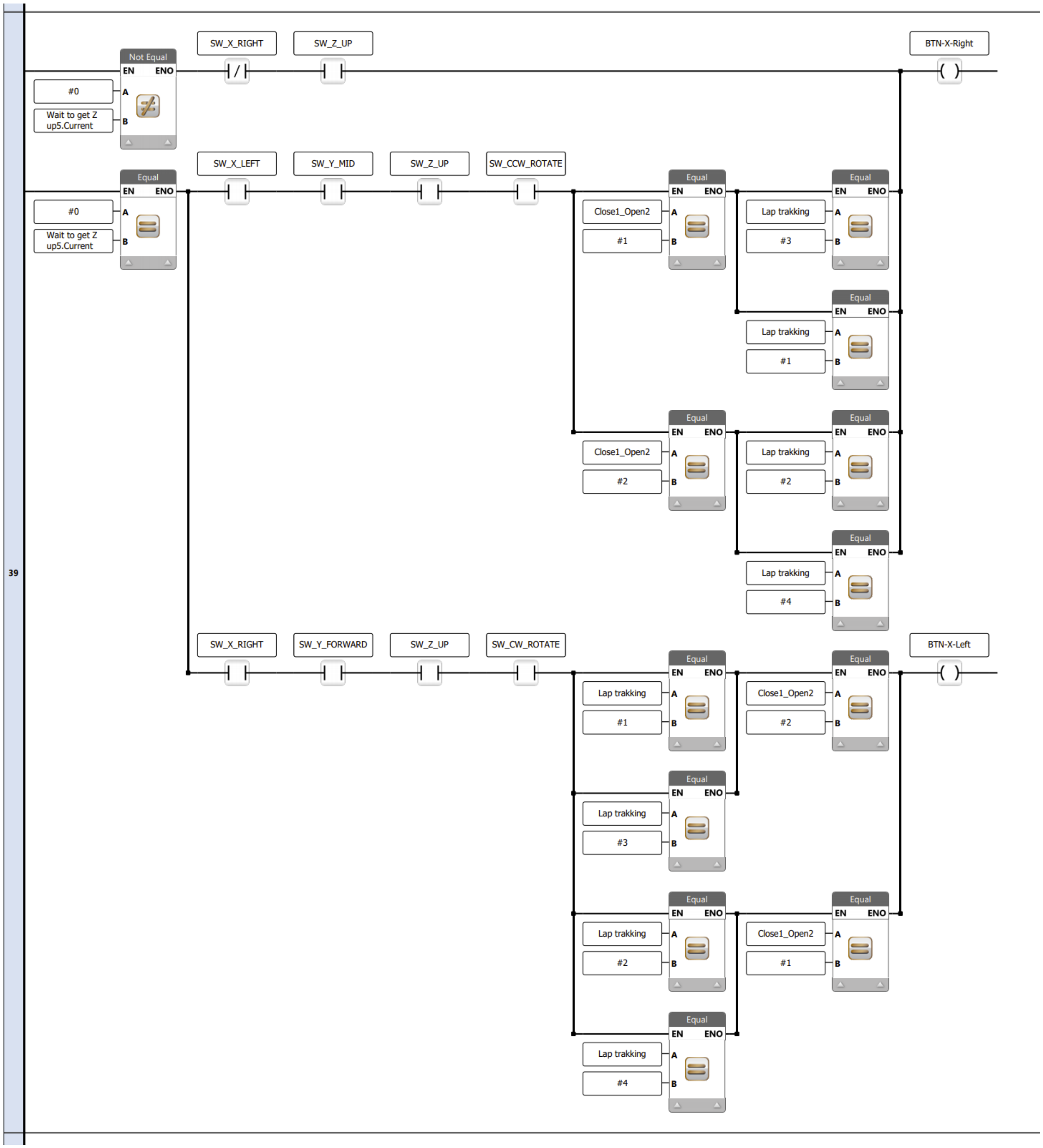

Figure 9. Ladder logic section 39: X-Movements

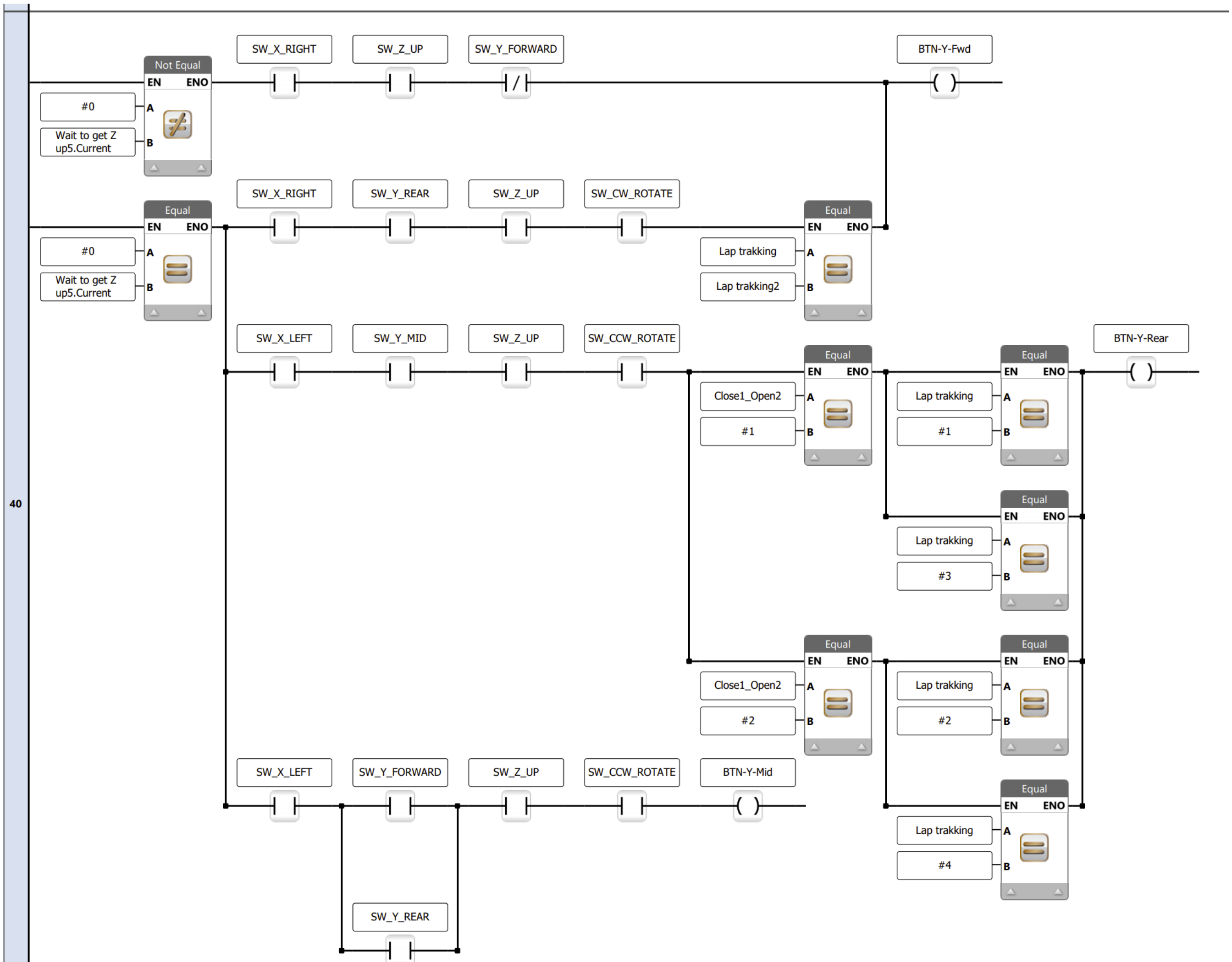

Figure 10. Ladder logic section 40: Y-Movements

Challenges

There were a number of significant challenges that arose during the completion of this project. In the Pneumatic Robot workstation, the original human machine interface (HMI), which was broken, was swapped with a working HMI of the exact same make and model from the neighboring Pneumatic Robot workstation. The ethernet communication between the laptop and the robot station failed to work; however, the use of a special USB cord allowed the project to overcome this obstacle. Using the USB cable was very easy and straight-forward. Another challenge was the lack of control over the speed at which the robot moved from one position to another. The robot would travel through the same range of motions but at varying speeds. When the program was initially downloaded, using the human machine interface (HMI) the robot could not be set to manual mode which ultimately did not allow the HMI to be used to return the robot to its original home position from varying positions within the work area. The program initially downloaded had to be unloaded, the robot could then be moved using the HMI, and then the program downloaded once again in order for the robot to start in the correct home position. Transition Contact conditions were not useful because the robot did not have time to move away from the sensor before the transition contact would go off. By using timers within the program this problem could be solved, but instead position conditions were used and Transition Contacts were used for counters. Little options were available to monitor the position of the robot. The sensors could only monitor when the robot was in a fixed position. The sensors could not tell where the robot is in space while it is transitioning from one position to another. The gripper cannot close all the way but can open all the way. The gripper has no sensors, so in it place a counter was placed in the program. A 1 signifies the gripper is open and a 2 signifies the gripper is closed. One output is assigned to each rung. Before the Pneumatic Robot can execute the UniLogic's's ladder logic program, the UniStream PLC download target must be selected. Once selected the program will be downloaded. The online value of lap tracking and lap tracking two must be set to 1. This should not be reset in the program or else the pneumatic robot will run continuously. Once M35 is pressed on the HMI the robot can execute the program created.

Conclusions

This paper describes the pneumatic pick and place robot project programmed using an HMI and a Unitronic ladder logic program. The robot system is a give from industry to the mechatronics master's program labs at the university. The robot components and application are described. A pick and pace demonstration program was developed. This project also has a companion video on YouTube.

References

[1] Bhalerao, A., Chopade, K., Doifode, P., & Gaikwad, J. Pick and Place Robotic ARM using PLC. Int. Journal of Engineering Research & Technology, Vol. 08, Issue 08, (Aug. 2019)

[2] Barz, C., Latinovic, T., Erdei, Z., Domide, G., & Balan, A. (2014). PRACTICAL APPLICATION WITH PLC IN MANIPULATION OF A ROBOTIC ARM. Carpathian Journal of Electrical Engineering, 8(1).

[3] Dias, E. P., Grandinetti, F. J., & da Silva Neto, A. M. AUTOMATION OF MANUFACTURING CELL USING AN INDUSTRIAL ROBOT AND CONTROLLED BY PLC.

[4] Laurean, Bogdan. (2009). ANALOG CONTROL AND PROGRAMMING OF A RTT ROBOT WITH A PLC. 10.13140/2.1.1513.8883.

[5] Laurean, B. (2010). PROGRAMMING AND CONTROLLING OF RPP ROBOT BY USING A PLC. ANNALS of the ORADEA UNIVERSITY. Fascicle of Management and Technological Engineering., XIX (IX), 2010/1(1). https://doi.org/10.15660/auofmte.2010-1.1764

[6] Dixit, S. R., Gosavi, D. V., & Chaudhari, H. P. Design of Electro-Pneumatic Robot using PLC and MATLAB with Smart Conveyor Management.

[7] Gaikwad, R., Kharat, S. M. M., & Thakur, J. (2017, September). PLC based automatic car washing system using proximity sensors. In 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI) (pp. 1875-1878). IEEE.

[8] Kang, B. S., & Cho, K. H. (2001, June). Design of PLCs for automated industrial systems based on discrete event models. In ISIE 2001. 2001 IEEE International Symposium on Industrial Electronics Proceedings (Cat. No. 01TH8570) (Vol. 3, pp. 1431-1434). IEEE.

[9] Salameen, L., Estatieh, A., Darbisi, S., Tutunji, T. A., & Rawashdeh, N. A. (2020, December). 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.

Biographies

ZONGGUANG LIU currently is a graduate student in Mechatronics at Michigan Technological University. He graduated from Electrical Engineering Technology at Michigan Technological University in 2019. He was a senior consultant in the math lab at Michigan Technological University since 2017. Interests include autonomous control and robotic system.

CHRISPIN JOHNSTON currently a graduate student in Mechatronics at Michigan Technological University. He graduated with his Bachelor's in Mechanical Engineering Technology at Michigan Technological University in 2019. Interests include additive and subtractive manufacturing, surgical robotic systems, and cyber security of industrial systems and autonomous vehicles.

ALEKSI LEINO is a production engineer at Adapteo in Finland. He obtained a master's degree in mechatronics, robotics and automation from Michigan Tech in 2021. In addition, he holds a master's and bachelor's degrees in mechanical engineering from Tampere University of Technology, in mechanical engineering.

TRAVIS WINTER is the engineering manager at Donald Engineering. He was instrumental in building and commissioning the advanced programmable logic controller systems installed at Michigan Technological University's Mechatronics Playground.

ALEKSANDR SERGEYEV is a Professor of Mechatronics, Electrical, and Robotics Engineering Technology program in the Department of Applied Computing at Michigan Tech. He is a Director of FANUC Authorized Certified Robotic Training Center, and a Director for Master of Science in Mechatronics degree program at Michigan Tech. Dr. Sergeyev is a member of SPIE, ATMAE, IEEE, and ASEE professional organizations, and has mentored numerous undergraduate senior design projects and student publications.

MARK GAUTHIER is the President and Owner of Donald Engineering Co., Inc (DE). located in Western Michigan. After Graduating from Michigan Technological University in 1985, he worked as a technical designer at SDRC in Farmington Hills, Michigan in statistical analysis and assembly review for the automotive industry. In 1988, he joined the family business at Donald Engineering and became power specialist and automation designer. He became president of DE in 1996. In 2019, he consulted and assisted with starting up the Mechatronics program at Michigan Technological University to further education in Automation, Fluid Power, Motion Control, PLCs and Machine safety.

NATHIR RAWASHDEH is an assistant professor at the department of applied computing at Michigan Tech. since August 2019. He instructs introductory and advanced programmable logic controller courses. Prior to this appointment, he was an associate professor in the Mechatronics Engineering Department at the German Jordanian University, where he spent 10 years. His industrial experience includes 5 years with Lexmark International, Inc. Lexington-Kentucky and MathWorks, Inc. in Natick-Massachusetts. Dr. Rawashdeh is a Senior Member of the IEEE and has experience with European-funded research capacity building projects. His research interests include mobile robots, autonomous driving, image processing and sensor fusion.