Al-Khwarizmi Engineering Journal
ISSN (printed): 1818 – 1171, ISSN (online): 2312 – 0789
Vol. 21, No. 2, June (2025), pp. 53-71

Al-Khwarizmi Engineering Journal

Development and Implementation of an Automated Production Line Training System

Ali Noori[1*], Faiz Fawzi[2], Hussain Tabina[3], and Egorov Igor Nikolaevich[4]

[1,2]*Department of Automated Manufacturing Engineering, Al-Khwarizmi collage of Engineering, University of Baghdad, Baghdad, Iraq*
[3]*Ministry of Higher Education and Scientific Research, Baghdad, Iraq*
[4]*Vladimir State University, Vladimir, Russia*
Corresponding Author's Email: ali.nouri2204@kecbu.uobaghdad.edu.iq

(Received 16 May 2024; Revised 9 January 2025; Accepted 16 February 2025; Published 1 June 2025)
https://doi.org/10.22153/kej.2025.02.003

Abstract

Industrial automation has significantly enhanced productivity, efficiency, and product quality by integrating advanced technologies such as robotics, artificial intelligence, and IoT. This study focuses on rehabilitating and developing the DL-MS6000 automatic industrial production line training system at the Ministry of Higher Education and Scientific Research in Baghdad, Iraq. The system, comprising six stations controlled by six PLCs, had issues, including hardware failures, the need for periodic updates, and a lack of SCADA control. The development process replaced the six PLCs with a single CYMON PLC and eliminated the need for CC-Link network devices, reducing system complexity and cost. The system was optimized using a new control algorithm. The system achieved high precision, zero steady-state error, and a settling time of 40 ms by using the Root Locus method in the MATLAB program. Additionally, a SCADA/HMI system was implemented to enhance control and monitoring. The newly developed system offers increased reliability, cost-efficiency, and ease of use, making it a suitable tool for educational purposes in university training rooms.

Keywords: *Automated production line; Programmable Logic Controller (PLC); Training system; SCADA.*

1. Introduction

Over the past few years, industrial automation has been increasingly integrated into companies' work, increasing efficiency and productivity. Industrial automation refers to utilizing control systems, machines, and information technology to automate traditional human tasks. This paradigm shift aims to simplify processes, increase efficiency, and improve product quality by integrating robotics, artificial intelligence, machine learning, and Internet of Things (IoT) technologies[1]. Control applications are deployed in automated industries through diverse hardware and software designs and methodologies. Product categories, including computer numeric control (CNC), programmable logic controller (PLC), and distributed control system (DCS), tackle specific issues and exhibit significant differences in programming and functionality. [2]. Research and industrial practice pay attention to the training and support system for industrial workers, which addresses the growing complications of industrial machinery. Skill or a procedure were taught in the training systems. These systems allow a person to become familiar with the process before training on the actual machine [3].

Helal and Yahya [4] study utilized human-machine interfaces to regulate and oversee an automated line of production. The system utilized TIA PORTAL software for HMI creation, LabVIEW for data trending, and FACTORY IO for the design, simulation, and implementation of the production line. The S7-1200 PLC controlled all functions, with ladder logic diagrams created by TIA PORTAL V13 software. The system permitted dual human-machine interfaces for the control and monitoring of the pneumatic system's physical parameters. facilitating monitoring of equipment performance. The operationalized system can be used in laboratories to bridge the gap between theoretical understanding and practical implementation, fostering innovation and practical implementation. Jingyu Li. [5] designed automatic material sorting control systems using Mitsubishi FX3U series PLC controllers and vision sensors for training purposes. The platform combines various technologies, including vision sensors, electrical and electronic, mechanical, frequency converter, motor drive, and human-machine interface, to teach the detection, transmission, and processing processes. Wan et al. [6] created a multifunctional sorting training platform using Siemens S7-200 PLC control. The platform consists of mechanical, electronics, electricals, microelectronics, sensors, interfaces, and signal transformation. It provides practical training experiences for teaching electromechanical integration. The platform uses modular thinking to make sorting modules more flexible, resulting in a greater educational effect compared to traditional training platforms. Sasidhar et al. [7] developed a low-cost automation (LCA) system that sorts objects based on their color using Allen Bradley MICROLOGIX 1200 PLC controller. The system consists of two parts: software, which uses ladder logic programming to control processes, and hardware, which includes conveyor belts, sensors, an electronic system for sorting, and a motor. The automated sorting machine improves efficiency, practicality, and safety for operators, ensuring exceptional processing capacity and performance, particularly in color identification. Additional functionalities include sorting operations based on dimensions, form, and mass, and simple sorting operations using a piston configuration. YE Li. [8] developed a practice training system using PLC programmable logic control, integrating mechanical and electrical laboratory equipment. The system includes a training platform, electromechanical integration equipment, PLC modules, touch screens, and sensors. It helps students practice and simulate production, developing their brains and increasing their professional knowledge and skills. Joanna Baroro et al. [9] studied the replacement of manual packaging systems in the industry using a proposed automated system. The system uses a Mitsubishi FX series programmable controller, sensors, motors, pneumatic components, and solenoid coils. Experiments showed a 50-75% reduction in time spent at filling, weighing, and capping stations, and approximately 90% complete automation without human intervention. Liang and Xiao. [10] developed a practical training system to simulate automatic production lines using a Siemens PLC (S7-200). The system comprises sensors, switches, motors, lights, pneumatic elements, and mechanical frameworks. It encompasses mechanical concepts, pneumatic controls, detectors, electric control engineering, and programmable logic controllers. The system is open, providing students with a complete overview of internal wiring situations. It can be used in single-station systems or the entire system, and can be reconstructed according to innovation needs.

The primary focus of the studies was on employing one or more Programmable Logic Controllers (PLCs) to manage various operations, including color detection, sorting, packaging, and handling. This was achieved by integrating expensive Human-Machine Interface (HMI) software and touch panel screen packages with PLCs provided by the PLC manufacturer. A few research papers specifically addressed the control and monitoring of pneumatic systems utilizing SCADA systems.

The motivation for this study is to rehabilitate and develop the automated industrial production line training system (DL-MS6000) in the Ministry of Higher Education and Scientific Research / Baghdad / Iraq. The system comprises six stations (distribution, testing, processing, handling, assembly, and storing) used to simulate the production line. The system is controlled by six PLCs (one for each station), and six network devices called CC-Links that connect the six PLCS to achieve a production line. This production line has technical problems in terms of the hardware, number of PLCs (6 PLCs), and network devices, which need a periodic costly update, which makes it unworking as a production line, and there is no SCADA system to control and monitor the production line. To develop this industrial line, reduce these problems, and facilitate control, maintenance, and operation, the CYMON PLC CPU CM1-CP4A PLC regulates six production line training system stations. A pneumatic system drives the system, creating a SCADA system for control and monitoring.

2. The Automated Production Line Training System

The automated production line training system (DL-MS6000) consists of six stations. The system is installed on an aluminum board with grooves, as shown in Figure (1). All stations can be combined easily to build an automatic product line. The stations consist of modular units that are easy to build, disassemble, and maintain. One Mitsubishi PLC controls each station. CC link is used as a network device to communicate and transfer data between the stations. The first three stations (Distribution station, testing station, and processing station) will be discussed in this study. Figure (2) illustrates the overall system block diagram.

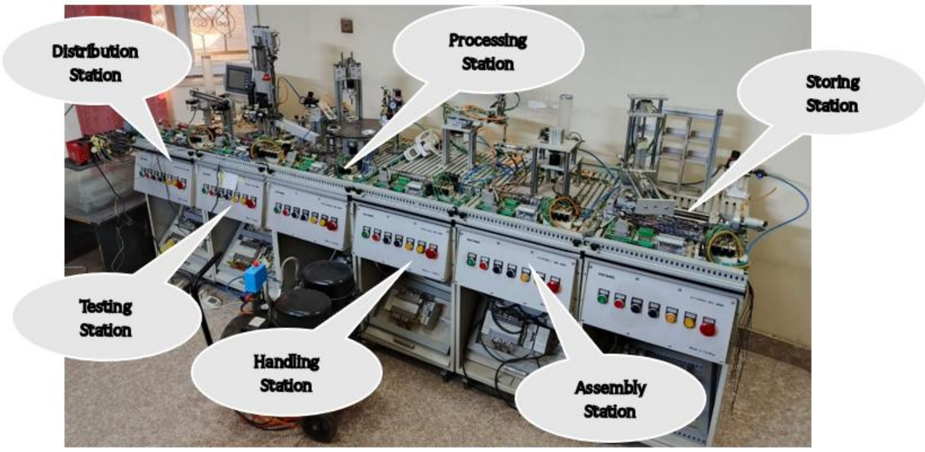


Fig. 1. Automated production line training system

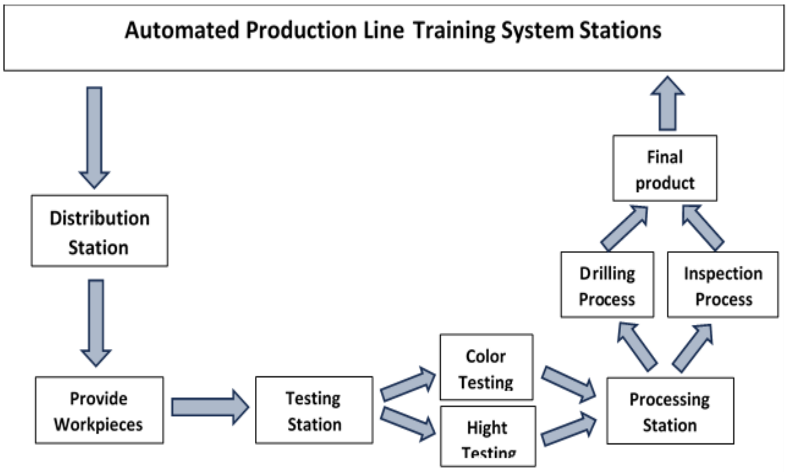


Fig. 2. The overall system block diagram.

2.1 Distribution Station

The distribution station is the first station, which plays a feeding role in supplying workpieces to the next station. It consists of two modules: a distribution module and a rotation transmission module. The distribution module can automatically deliver workpieces from the silo to the transfer stage so that the rotating cylinder can pick them up and transport them to another station. The distribution station is shown in Figure (3).

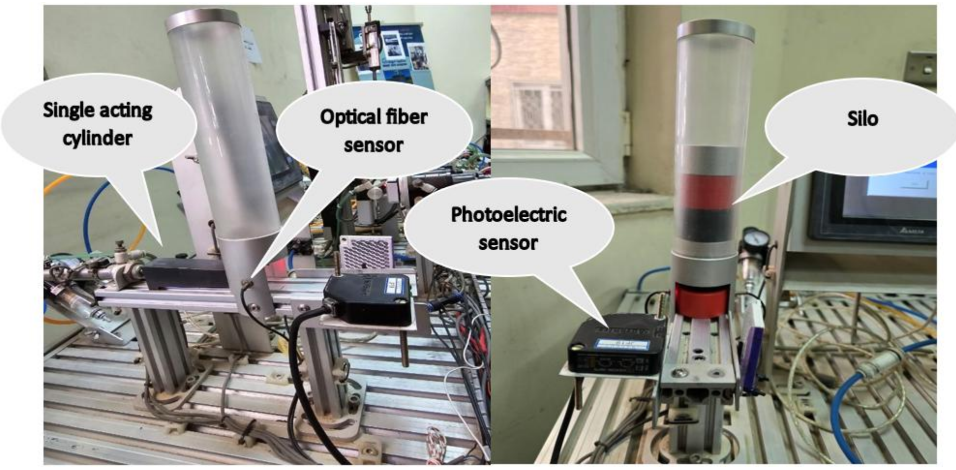


Fig. 3. The distribution module

The rotation transmission module consists of a rotating cylinder, the vacuum generator, and suction cups combined, and the workpiece is transmitted from the distribution module to the next station.

2.2 Testing Station

In this station, the testing process worked in two steps. The first step is testing the color (Red, Black, and silver) and material (plastic or metallic) of the workpiece, while the second step is to measure the height of the workpiece (high, low, or medium). The testing station has five parts: detection module, measuring module, Rodless cylinder, Defective loaded Plunger, and Swing module. The detection module consists of three types of sensors (Inductive sensor, Photoelectric sensor, and Capacitance sensor), sensor tray, and workpiece launch cylinder, as illustrates in Figure (4). The inductive sensor detects metal using the reflection principle; the capacitive sensor tests the existence of the workpiece, while the photoelectric sensor detects the color.

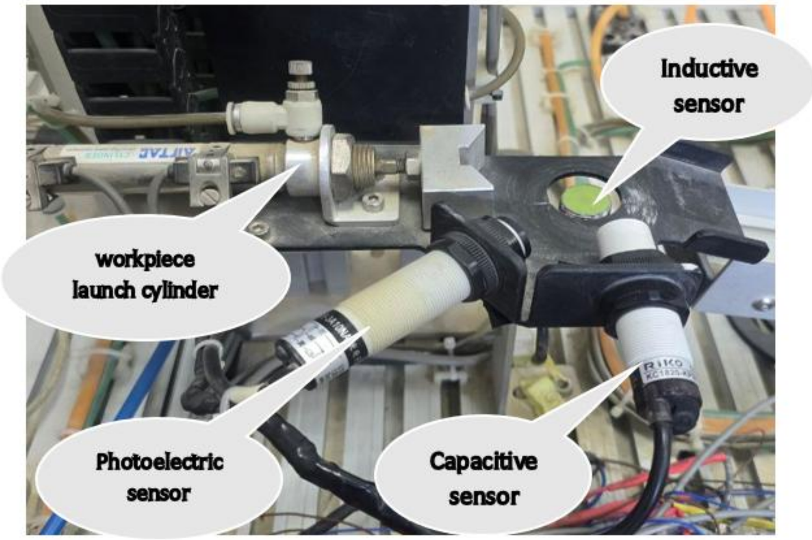


Fig. 4. The detection module

Judged the Material type according to different combinations of detection signals. The material state is determined according to the signal from other parts of each sensor, as shown in Table (1).

Table 1,
Detection signals

Workpiece's Material / Sensor Type	Black Workpiece	Colored Workpiece (Black, Red and Silver)	Metal Workpiece
Photoelectric Sensor	○	●	●
Capacitance Sensor	●	●	●
Inductive Sensor	○	○	●

The Measuring Module consists of a height Measuring Device, Analog Sensor, and Height Measuring Transfer Cylinder, used to measure the Height of the workpiece. The rodless cylinder is used to rise and down the testing platform, while the main role of the defective loaded Plunger is to load the faulty workpieces.

The Swing module consists of a swing cylinder, a rise cylinder, and a vacuum sucker. The swing module works as a link to transport the workpiece between the testing station and the processing station. As shown in Figure (5).

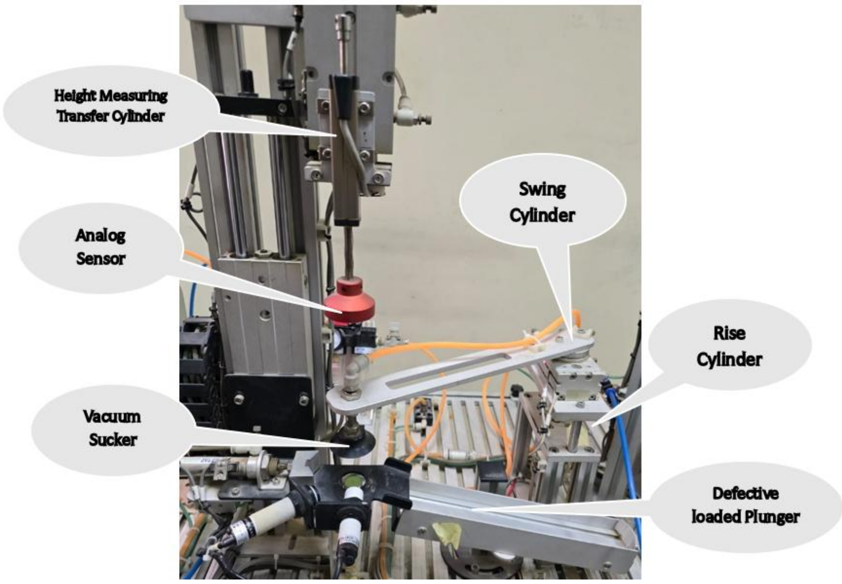


Fig. 5. Testing Station

2.3 Processing Station

This station has three modules (turntable module, drill module, and hole detect module), as shown in Figure (6). The turntable module consists of a DC gear motor, capacitive sensor, and inductive sensor. It has four workstations. The swing module draws the workpiece to the initial workstation. Upon detection of the workpiece by the capacitive sensor, the turntable rotates 90 degrees, transferring the workpiece to the drilling module for the inner diameter drilling. Subsequently, it is moved to the hole detection module workstation to ascertain its qualification before being sent to the fourth workstation.

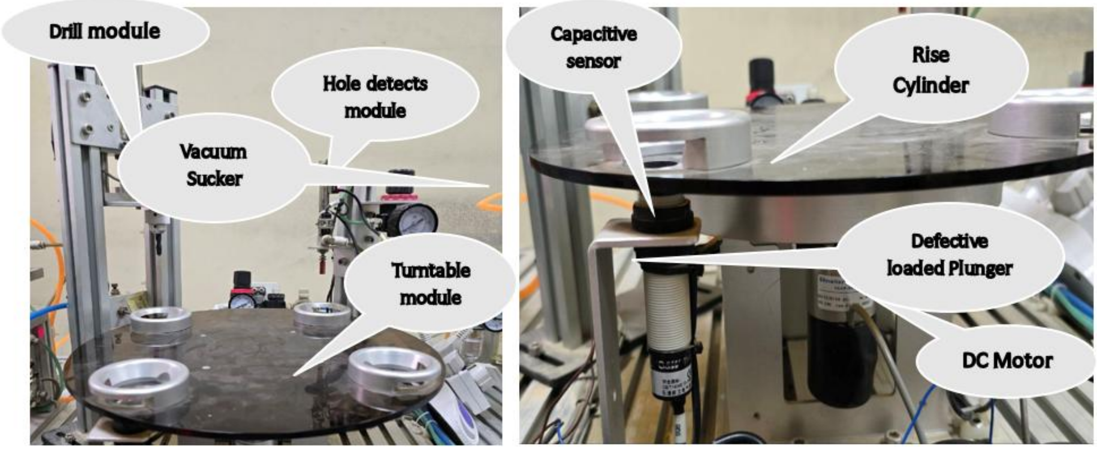


Fig 6. The Processing Station and the turntable module

The drilling module consists of a DC motor and a double-acting cylinder; when the workpiece reaches the drill workstation, the cylinder moves down to start the drilling process, and then the turntable rotates 90 degrees to move the workpiece to the detection workstation. The hole detects module is a double-acting cylinder that works as a detection cylinder to judge whether the workpiece has a hole or not and to test the qualification of the inner hole.

3. Undeveloped system

The undeveloped production line training system has many problems:

1. High operating cost due to the need for continuous and expensive periodic updating of the connection devices.
2. Using six PLCs to control the system makes it a high-cost system.
3. The need for six connection devices (CC-Link) to link the stations together to simulate the synchronous production line increases the system's cost.
4. The complex, huge, and closed-source software program used to operate the system is difficult for trainers to understand.
5. There is no SCADA system and the possibility of switching to manual operation and turning on and off the connection units may cause some interruptions and challenges in the system's work.

4. Methodology to Develop the System

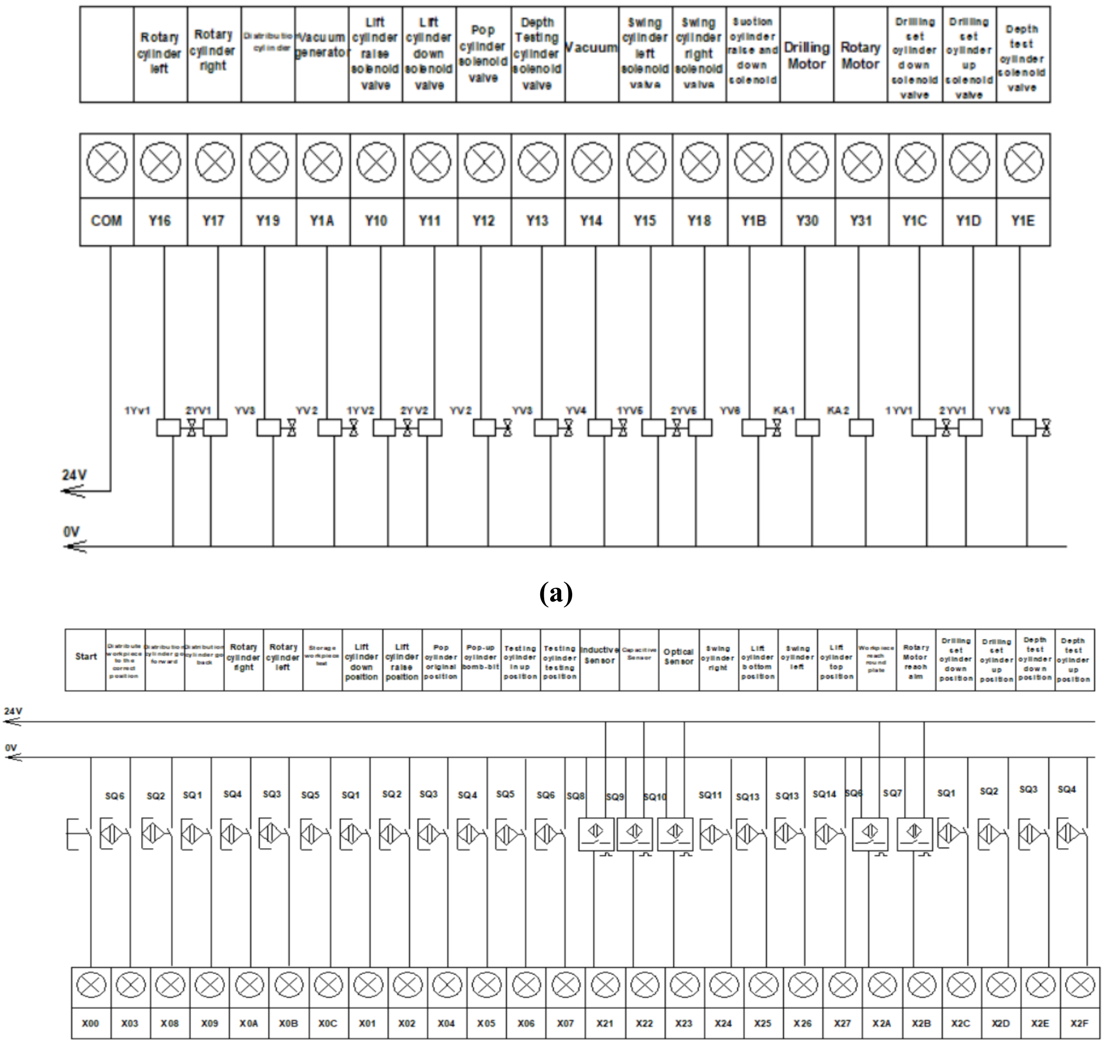
Many steps are followed to rehabilitate and develop the automated production line training system, starting with diagnosing and discovering the system's problems and obstacles. After determining the main issue of the system, a new automatic control system was designed and implemented, using CYMON PLC CPU CM1-CP4A PLC to regulate six stations, which cancels the six PLCs therefore, there is no need for using six connection devices (CC-Link) to link the stations together to simulate the synchronous production line. Building a new, simple, and smooth program with few steps makes the system less complex and more manageable for students and trainees to understand. Cancel the manual operation option and create a SCADA system for control and monitoring. Hardware and software components selected depend on the system functions and cost avoidance.

4.1 Hardware Development

The major hardware elements that are utilized to execute the creation of the automated production line training system will be outlined in the following:

4.1.1 Wiring system

A new automatic control system was designed and implemented to control the whole automated production line training system using one Programmable logic controller (PLC), which needed to be made into a new wiring system, especially when most of the system's wires were damaged. The system uses 0.25mm control wires with 0.5mm terminals, which are compatible with the PLC's I/O ports and the system terminals. Figure (7) shows the new wiring diagram.



(a)

(b)

Fig. 7. The wiring diagram. (a) Inputs Circuit, (b) Outputs Circuit

4.1.2 Photoelectric Sensor

BEN5M-MFR/ BEN3M-PFR Photoelectric Sensor Retroreflective type used in the first station instead of an optical sensor, which didn't work properly; that gives the needed function and the economical price made this sensor perfect to use

Table 2,
Fiber-Optic Photoelectric and Photoelectric Sensor properties.[11], [12], [13], [14]

Sensor Type	Fiber-Optic Photoelectric Sensor E3X-A11	Photoelectric Sensor Retroreflective BEN5M-MFR
Amplifier	High Performance Amplifier	Amplifier Built-In Type with Universal Voltage
Sensitivity adjustment	8 turn potentiometer	Adjustment VR
Sensing distance	Extended sensing distance	0.1 to 5m
Light source	Pulse-modulated red LED (660 nm)	Infrared LED (850nm)
Response time	200 µs max	Max. 20ms
Current consumption	35 mA max	Max. 4VA
Working principle	The sensor detects objects by transmitting or reflecting light, causing a change in light intensity. This signal intensity is processed and converted into a logic or electrical output.	The photoelectric sensor consists of an LED transmitter and a phototransistor, a solid-state semiconductor emitting light when electric current travels through it. The output signal changes based on object reflectivity.

4.1.3 Capacitive Proximity Sensor:

The capacitive proximity Sensor KC1820-KP2K replaces the capacitive proximity Sensor CR18-8DN, which did not work. This sensor used in the testing station to detect the workpiece. Because the Sensing distance of the old sensor (CR18-8DN) was minimal (0 ~ 5.6mm), the sensor fixed to be so close to the workpiece to detect it, and when the swing module transport the workpiece between the testing station and the processing station the workpiece hit the sensor which caused a sensor failure, therefor replace the old sensor with a new model (KC1820-KP2K) with more considerable Sensing distance (2 ~ 30mm) make it suitable to detect the workpiece with reasonable distance from the workpiece to protect the sensor from any friction with the Swing Module

Table 3,
Capacitive Proximity Sensor CR18-8DN and Capacitive Proximity Sensor KC1820-KP2K properties.[15], [16], [17]

Sensor Type	Capacitive Proximity Sensor CR18-8DN	Capacitive Proximity Sensor KC1820-KP2K
Sensing materials	Iron, Metal, Plastic, Water, Stone, Wood etc.	Metal, Plastic, Water, Oil
Sensitivity adjustment	15±3 -turn potentiometer (VR).	20-turn potentiometer (VR).
Sensing distance	0 ~ 5.6mm	2 ~ 30mm
Hysteresis	MAX 20 % of sensing distance	≦ 10 % of sensing distance
Current consumption	MAX. 15 mA	≦ 22 mA no-load
Working principle	Establishing an electric field by applying an alternating current (AC) voltage to a transmitting electrode detects the approach of an object or material by a slight change in capacitance value	Establishing an electric field by applying an alternating current (AC) voltage to a transmitting electrode detects the approach of an object or material by a slight change in capacitance value

4.1.4 Programmable Logic Controller (PLC)

The automated production line training system had three MITSUBISHI FX2N-32MRCCL PLCs to control the system and three CC-Links to link the stations to communicate. The main issue in this system was the CC-Links, which needed periodic updates, it was expensive, to provide the necessary communication between the stations to simulate a production line. To solve this problem, The CYMON CPU CM1-CP4A PLC is used to control the whole system, and there is no need for communication or network devices. The MITSUBISHI FX2N-32MRCCL PLC has only 16 inputs and 16 output ports, and the system has more than 45 digital and analogue inputs and outputs; therefore, it cannot be used as a core controller. The CYMON CPU CM1-CP4A PLC controller provides flexibility and power to control various devices. The small design, adaptable setup, and robust instruction set render the CYMON PLC CPU CM1-CP4A an effective choice for system control.

Table 4,
MITSUBISHI FX2N-32MRCCL PLC and CYMON CPU CM1-CP4A PLC specification.

PLC Type	MITSUBISHI FX2N-32MRCCL	CYMON CPU CM1-CP4A
Operation Method	Cyclic operation by stored program., performed by dedicated LSI	Stored Program, Cyclic Operation, Time Driven Interrupt
I/O control Method	Batch processing when END instruction is executed. I/0 refresh instructions available	Indirect, Direct by Instructions
Program Language	Relay symbolic language + Stepladder. SFC expression possible	IL (Instruction List), LD (Ladder Diagram)
Data Processing Method	32 bit	16 bit
Counter	Up Counter, Up-Down Counter, High Speed Counter	Up Counter, Down Counter, Up-Down Counter, Ring Counter
Operation Mode	RUN, STOP	RUN, STOP, PAUSE, DEBUG

4.2 The New Automatic Control System

The new algorithm was built to perform the processes and system functions, as shown in Figure (8). The CICON software managed all units of the automated production line, using ladder languish to build the program to perform the required actions for this project according to the algorithm.

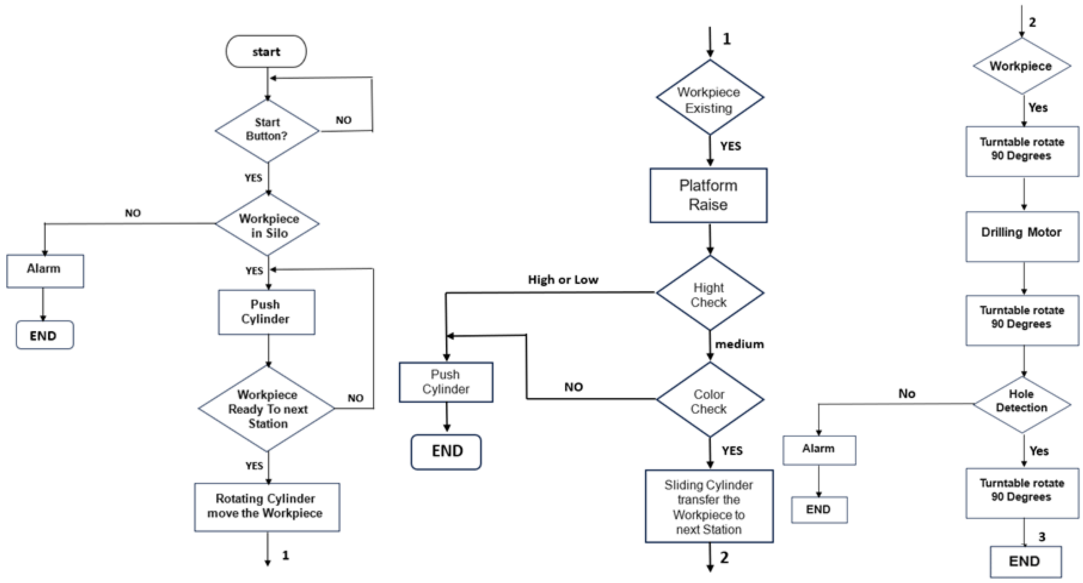


Fig. 8. The flowchart of the first three Stations.

4.3 System Modelling of DC Motor Position of the Turntable

During the industrial line's operation and after completing the electrical wiring using one PLC instead of six PLCs, a problem appeared in the third stage. According to Table (5), this problem is summed up by the fact that the turntable deviates with a different error rate while moving to the virtual location corresponding to the drilling process.

Table 5,
Error rate for the turntable location

Exp. No	Desired theta	Actual theta	Error rate
1	230°	272°	42°
2	230	274°	44°
3	230°	275°	45°
4	230°	275.8°	45.8°
5	230°	276.5°	46.5°

The white paper was divided into 360-degree sections to represent the turntable's total angular movement and the DC motor's actual position during the automatic process, as shown in Figure (9).

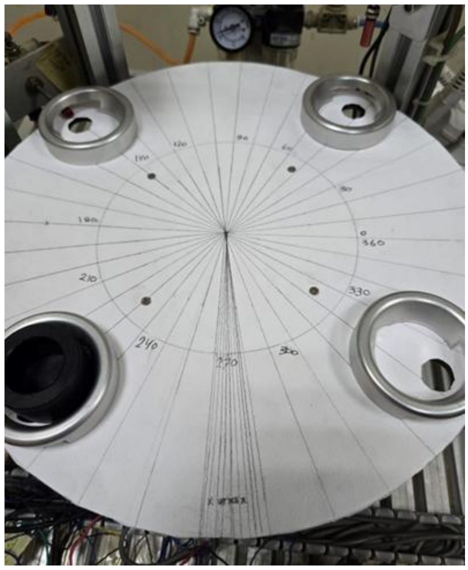


Fig. 9. Turntable

Figure (10) Depicts the electrical circuit counterpart of the armature with the free-body diagram (FBD) of the rotor. The transfer function is constructed using both components and assessed under no-load and full-load situations.

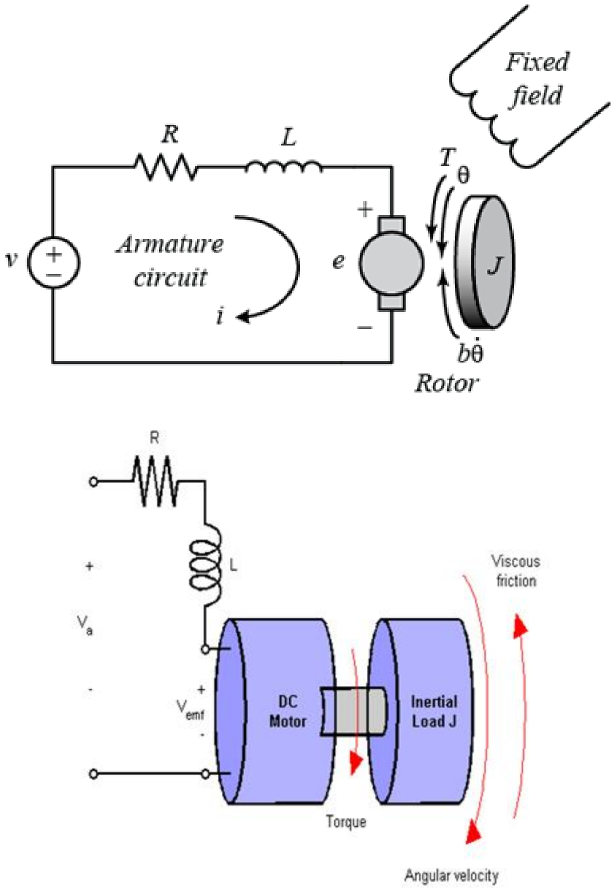


Fig. 10. The electrical circuit counterpart of the armature with the free-body diagram (FBD) of the rotor.

For this simulation, the subsequent values were assumed for the physical parameters APPENDIX1 for DC motor series SG371 with spur gearbox, which is shown in Figure (11).

Fig. 11. DC motor of the turntable

Hypothesized that the system's input is the voltage source (V) applied to the motor's armature, while the output is the shaft's position (theta). The rotor and shaft are presumed to be inflexible. A model of viscosity friction was proposed, in which the frictional force is directly proportional to the shaft angular speed.

The torque generated by the DC motor is directly proportional to the current through the armature and the strength of the magnetic field. This magnetic field is assumed to be uniform in this context. Thus, the motor force is directly correlated to the armature current by a factor that remains constant, as illustrated in the equation below. This is known as an armature-controlled motor.

$T = k * i$ …(1)

k = the armature coil constant, i = the armature current.

By using Kirchhoff's law

$V = L\,(di / dt) + R * i + e$ ...(2)

L = Inductance, R = resistance, e = back emf

By applying Laplace transform,

$i = (v - k2\, s\theta) / (Ls + R)$ …(3)

Also,

$T = J\,(\ddot{\theta}) + F\,(\dot{\theta}) = k * i$ …(4)

J = inertial torque, F = coefficient of friction.

$i = ((Js^2\, \theta) + Fs\theta) / k1$ …(5)

By combining 3.3 & 3.5,

$\theta / v = k /((Ls + R)(Js + F) + k\_1*k\_2)\, s$ …(6)

k1 and k2 = constants

The motor must be accurately aligned to guarantee that the steady-state error in its position is null when a designated position is specified. The steady-state error induced by a constant perturbation must also equal to (zero). The supplementary performance criterion is that the motor reaches its final position swiftly without considerable the overshoot. In this situation, the system must attain a settle time of 40 milliseconds and the overshoot of under 16%.

Simulating a reference input using a unit step input should provide the subsequent motor position output:

- Settling time below 0.04 seconds
- Do not exceed 16 per cent overshoot
- No steady-state error arises, even when exposed to a step disturbance input.

4.3.1 The Representation in MATLAB

The transfer function: The motor's open-loop transfer function can be represented in MATLAB by defining the transfer function and the following parameters. Executing the following script in the command line yields the output displayed below.

Table 6,
The parameters of the DC motor

Symple	Paremeter name	Value
(J)	Moment of inertia of the rotor	9.6852 E-6 kg.m^2
(b)	Motor viscous friction constant	10.5231 E-6 N.m.s
(Kb)	Electromotive force constant	0.0822 V/rad/sec
(Kt)	Motor torque constant	0.0844 N.m/Amp
(R)	Electric resistance	15 Ohm
(L)	Electric resistance	8.25E-6H

Transfer function in continuous time: The controller will be developed using the Root Locus approach by converting an analog DC motor model, the sample data for the model of the DC motor can be obtained. The position of the DC motor open-loop transfer function in the Laplace domain is as follows.

$$P_{(S)} = \frac{\theta_{(S)}}{V_{(S)}} = \frac{K}{S\,(\,(J_S+b)(L_S+R)+K^2)} \quad \left[\frac{rad}{V}\right]$$

Furthermore, the design specifications of the 1-radian stepped reference are as follows, as indicated on the main page.

- The settling time within 40 milliseconds.
- An overshoot below 16 percent.
- No error in the steady-state occurs, even when subjected to a step disruption input.

4.3.2 Developing a Plant Sampled-Data Model.

The first step in designing a digital control system is to generate a sampled data model of the plant. Therefore, choosing a frequency with which the continuous-time plant is sampled is necessary. In choosing a sampling period, it is desired that the sampling frequency be fast compared to the system's dynamics so that the sampled output of the system captures the system's full behavior so that significant inter-sample behavior is not missed.
Create a continuous-time model of the plant. Create a new m-file and add the following MATLAB code (refer to the main problem for the details of getting these commands). Running the m-file within the MATLAB command window will generate the below output.

Using the zpk command above transforms the transfer function into a form where the zeros, poles, and gain can be seen explicitly. Examining the poles of the plant (or its frequency response), it is clear that the pole at -1.0287e+09 contributes very little to the plant's response. The gain crossover frequency of the plant is approximately 5 Hz. Therefore, choosing a sampling period of 0.001 seconds (frequency of 1000 Hz) is significantly faster than the dynamics of the plant. This sampling period is also fast compared to the speed that will be achieved by the resultant closed-loop system. A sample time of 0.001 seconds is specifically 1/100 of the required time constant and 1/40 of the required settling time.
The specified transfer function will be transformed from a continuous Laplace domain into a discontinuous z-domain. The c2d command in MATLAB facilitates this translation, and it necessitates three parameters:

- A model of a system.
- The sample time (Ts).
- The hold circuit classification.

Assuming a zero-order hold (ZOH) circuit, including the subsequent commands into the m-file and executing them in the MATLAB command window produces the sampled data model below.
Observe the presence of both the pole and the zero near z equal to zero, which effectively negates one another. The cancellation in the transfer function (TF) can be performed using the minreal command with an acceptable margin of 0.001. Removing those poles and zeros will reduce the transfer function (TF) order and alleviate numerical difficulties in MATLAB. The utilization of the mineral command produces the resulting reduced transfer function order. Note the absence of the pole and the zero at z equal to 0.

Analyse the system's closed-loop response without further compensation. Initially, it must close the transfer function loop by employing the feedback instruction. The closed-loop response to steps was analysed utilizing the zero-order of hold technique after loop closure. This can be accomplished by using the steps and stairs instructions. When provided with a discrete model, the step command will produce a vector of discrete samples at the sampling time Ts corresponding to the model. The following syntax indicates the simulation of the step response for 0.5 seconds. The steps command produces discrete data points in a stairstep manner, similar to the zero-order of hold circuit output. Adding the following code to the last of the previous m-file and re-executing it generates a graph similar to the one illustrated below.

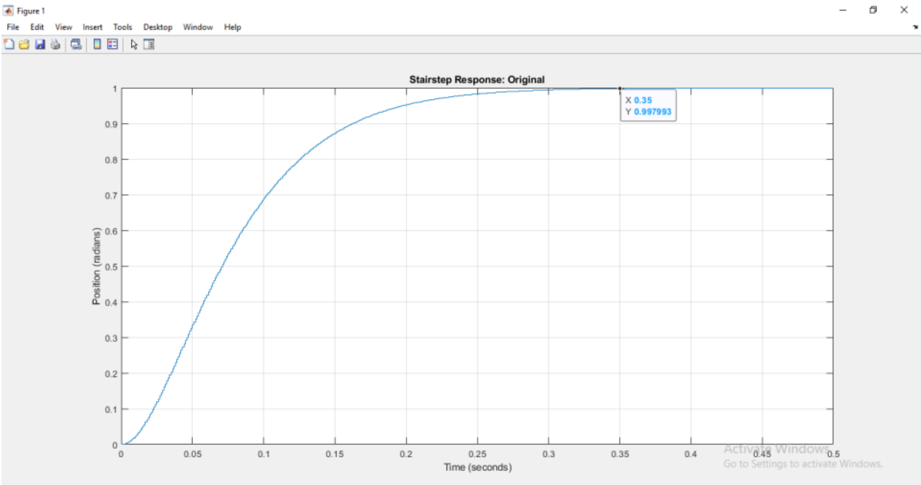


Fig. 12. Stair step response

4.3.3 Root Locus Design

The root locus methodology in design aims to adjust the closed-loop response by positioning the system's closed-loop poles at specified locations. This methodology applies to discrete-time models as well. The controller is designed using the Control System Designer graphical user interface (GUI).

Initially, a window titled Control System Designer will open, displaying the form illustrated in Figure (13). This window displays the root locus of the transfer function dP motor processed by the Control System Designer function.

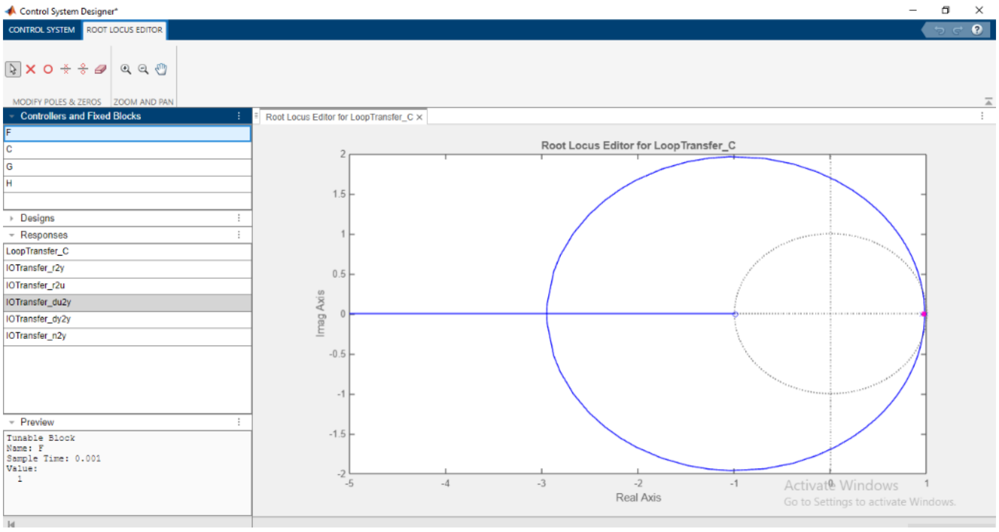


Fig. 13. The root locus of TF.

In the Control System Designer, an integrator is incorporated into the compensator to mitigate the impact of a constant disturbance in a steady state. The integral control in continuous-time domains is denoted as 1 / s. The Control System Designer (CSD) utilizes a forward variance estimation to transition from the s-plane for the z-plane, expressed as s Equals (z-1) / Ts, where Ts denotes the sample time. Including an integrator introduces a pole at one on the root locus diagram. The standard compensator format is the time constant form; however, the compensator will be formulated in zero/pole/gain form for this case.

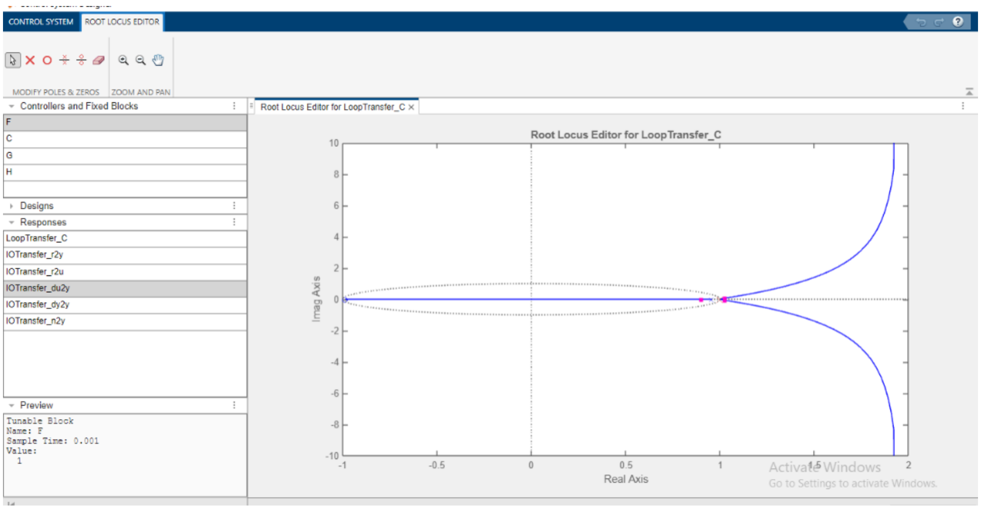


Fig. 14. Adding compensating pole at z=1.

The Figure (14) illustrates a root-locus chart of a stable closed-loop system with three poles near 1, suggesting increased instability. A zero is added at one inside a unit circle to remove one of the poles and direct the inward of the root locus. The zero, located at z = 0.95, is included in the compensator in a manner analogous to that of the integrator. The Real Zero is selected from the context menu and positioned at 0.95 in the Location cell. The root locus diagram should resemble that depicted in the Figure (15).

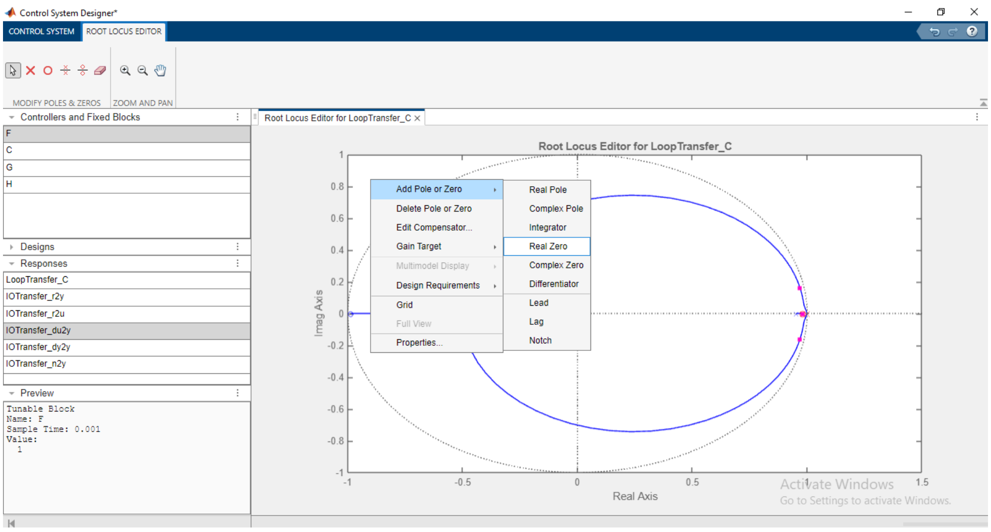


Fig. 15. Adding Real Zero at 0.95 real axis

The integrator has improved disturbance rejection requirements; However, the closed-loop system may not achieve robust stability by additional adjustments to loop gain. Additional design requirements remain unaddressed. To resolve these issues, ascertain the area of the complicated plane in which closed-loop poles have to be situated. Incorporate these areas into the root locus by selecting the design requirement and configuring the overshoot to less than 16%. Reiterate this procedure to incorporate the time to settle criterion under 40 milliseconds. The unshaded region identifies the closed-loop poles for transient requirements, assuming a canonical second-order system.

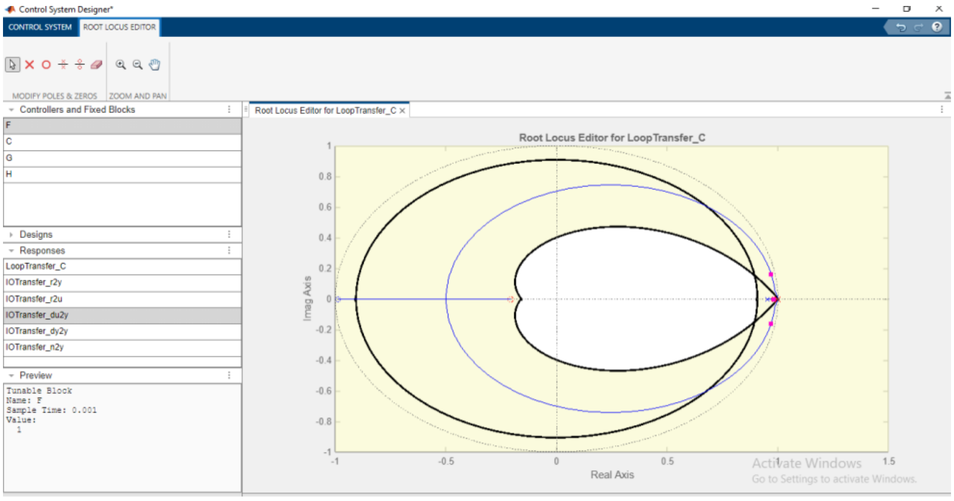


Fig. 18. Pole zeros compensator results

Figure (18) reveals that the root locus' two dominant branches don't pass through the desired region, requiring additional poles or zeros for the compensator to bend it. To cancel the zero at -0.98, add a pole nearby, as it adds overshoot into a step response. This will yield two branches from a root locus extending to the right and beyond the unit circle.

Two supplementary zeros can be incorporated near the desired closed-loop pole positions through the Compensator Editor window (CEW) to augment the branches of a control system. A graphical tweaking methodology is employed in this instance. Choose a solitary zero from the ROOT LOCUS EDITOR (RLE) tab, which is on the top of the Control System Designer (CSD) window, and click the resultant "pointer" on the real axis to position the zero. Reiterate the same procedure for the second zero. The effect of moving these two zeros can be visualized by moving the pointer over the corresponding zero on the root locus and clicking on it. An additional pole will need to be added to make the controller causal. Using a graphical approach, an actual pole can be added by choosing an x from the ROOT LOCUS EDITOR (RLE) tab. Through trial and error, the arrival was at a pole position of 0.6. The resulted root locus is shown below.

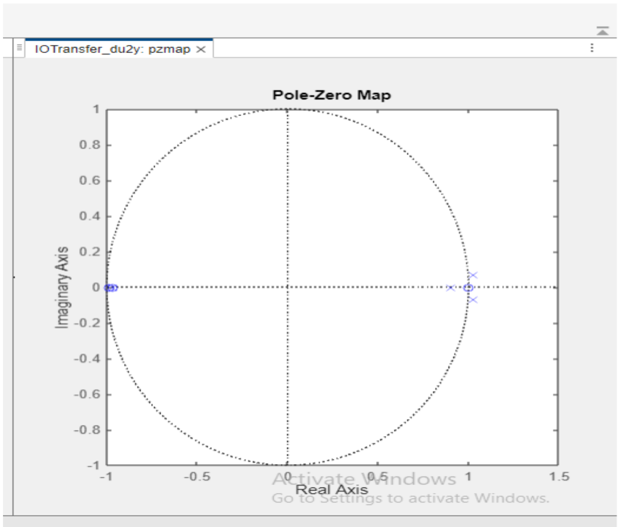


Fig. 19. Try and error adding pole and zeros

To move closed-loop pole positions along a root locus, a loop gain must be chosen. Due to the non-canonical second-order closed-loop system, trial and error are necessary to identify pole locations. A plot for the closed-loop step response can help observe the effect of gain changes on the actual step response without relying on second-order idealization.

To check the system's closed-loop step response, navigate to the IOTransfer_r2y: step tab. If closed accidentally, it can still be open. From the Control System Designer window, click on the New Plot menu, select New Step, and then choose IOTransfer_r2y from the Select Responses to Plot menu. The closed-loop step response will appear in figure (20).

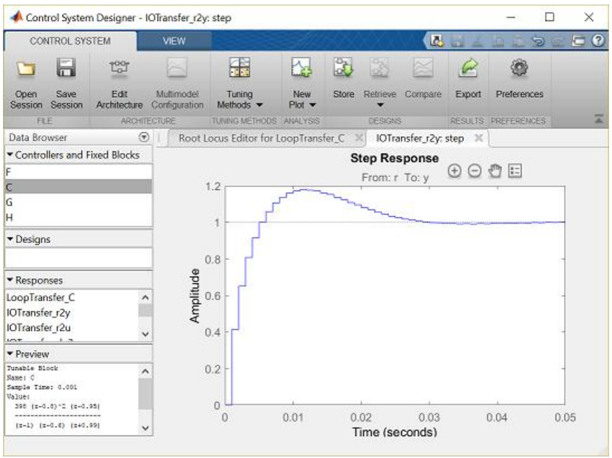


Fig. 20. The closed-loop step response

The step response plot with a loop gain of 1 meets the settling time requirement but not the overshoot requirement. To meet the overshoot requirement, the allowed step response shape is defined by choosing Design Requirements and setting the overshoot to 16% and the settling time to 0.040 seconds. A rise time must be less than 0.040 seconds.

To modify the loop gain, a graphical tuning approach is used by dragging pink boxes on the root locus plot to place a closed-loop pole at a specified location. A loop gains of 800 reduces the overshoot to 10% and achieves a settling time of 0.02 seconds. The steady-state error goes to zero when no disturbance is present. The resulting step response plot is shown in Figure (21).

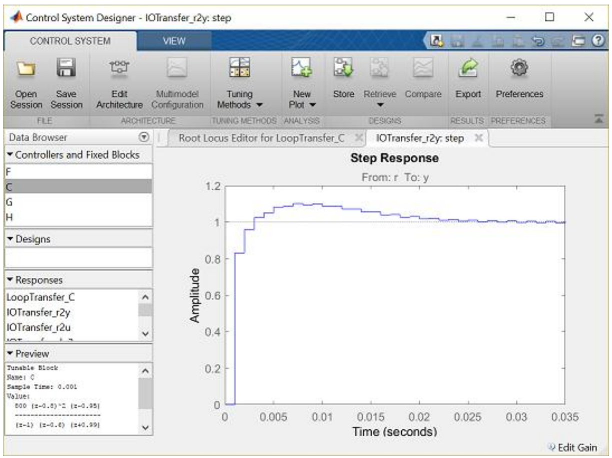


Fig. 21. The resulting step response plot

To verify the system's disturbance rejection properties, generate a disturbance response plot from the Control System Designer window or from the MATLAB command line. Export the designed compensator in the Control System Designer to the workspace by clicking on the Export button in the CONTROL SYSTEM tab. Select compensator C and click Export. Enter the following commands to produce the response plot shown in Figure (22).

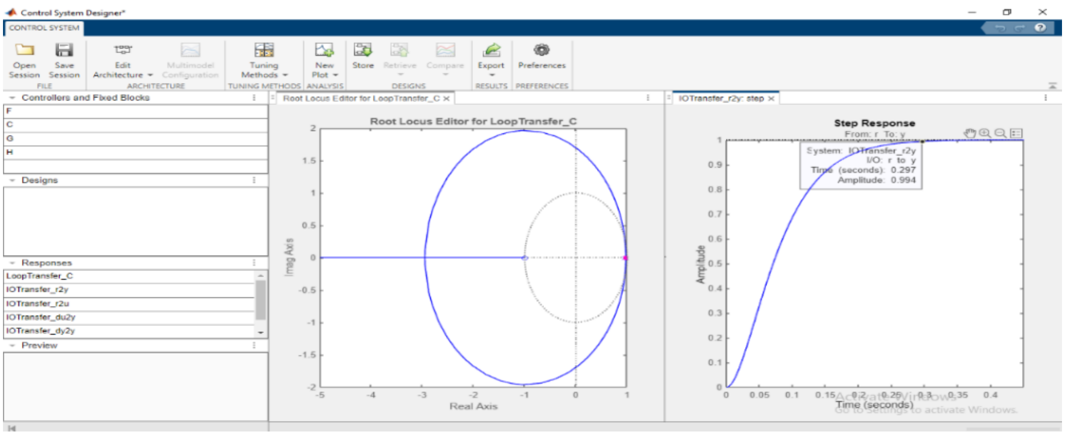
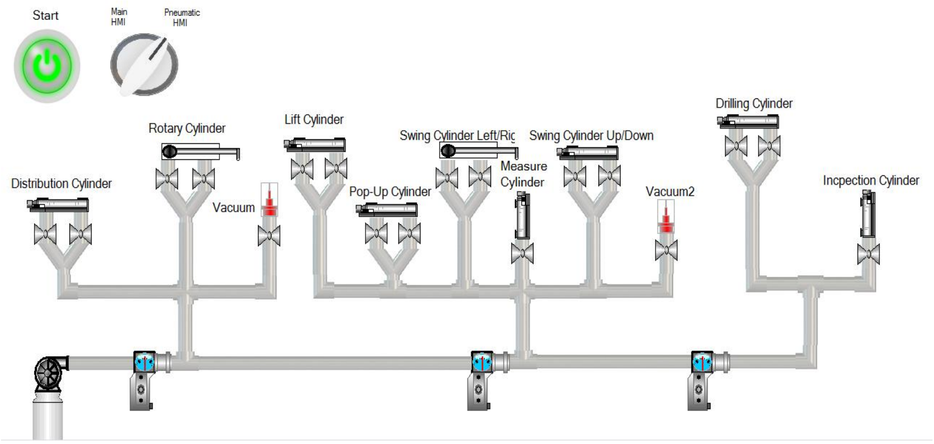


Fig. 22. The response plot.

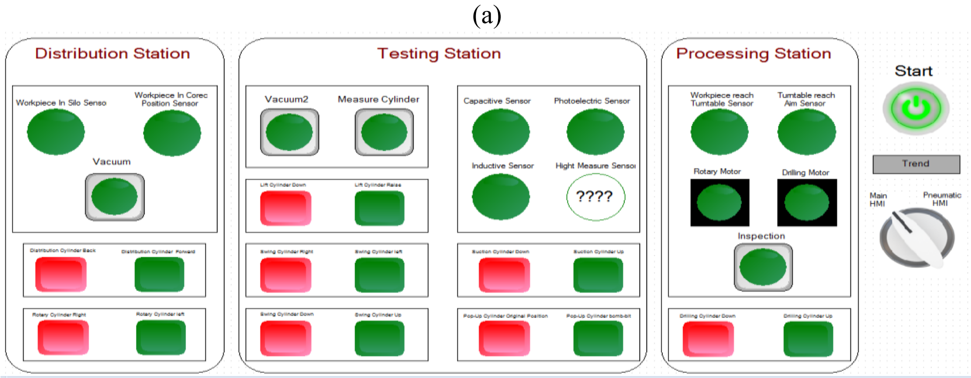
Examination of the above shows that for a step disturbance, the steady-state error will still go to zero. Employing the property of superposition, it is also apparent that the total response will have an overshoot of less than 16% and will reach a steady state well below the 40-millisecond requirement.

4.4 Design and Implementation of SCADA/HMI System

The CICON software was employed to automate the production line and develop the SCADA/HMI system. The initial stage involved accessing the system's project (main ladder program) and incorporating the tags for all digital and analogue inputs, outputs, and timers. Subsequently, the HMI tags were established and associated with PLC tags. These tags facilitated the connection between HMI panels, sensors and actuators on the production line. The sensors and actuators were interfaced using PLC I/O modules. The HMI display was engineered to accurately depict the real system. The HMI facilitated the visualization of the values produced during the process, as illustrated in Figure (23).



(a)



(b)

Fig. 23. HMI in CIMON (a) Pneumatic HMI, (b) Main HMI

5. Results and Discussion

As a result, the realized system worked at high performance after the rehabilitation and development of the hardware components, Enables one PLC to control the automated production line training system without any network or communication device to link the stations, making it work consequently to simulate a real production line and reduce the cost of the system by cancelling the CC-Links devices and 2 PLCs by using one PLC instead of three that means there is no need for the periodic update for the connection devices. Building a new, simple, and smooth program with few steps makes the system less complex and easier for students and trainees to understand than the previous program used to control and manage this system. Solved the problem that appeared in the third stage during the industrial line's operation that the turntable deviates with a different error rate while moving to the virtual location corresponding to the drilling process; the motor must be precisely positioned, with zero steady-state error and constant disturbance error, and achieve a settling time (TS) of 40 milliseconds and less than 16% overshoot. Using the MATLAB Program, the motor's open-loop transfer function (TF) is represented. The controller was designed using the Root Locus method, converting an analogue DC motor model into a sample-data DC motor model based on the Laplace domain transfer function. Root locus methodology adjusts closed-loop response by positioning poles at specified locations. This technique applies to discrete-time models and is designed using the Control System Designer GUI. The loop gain is adjusted using a graphical tuning approach to meet the overshoot requirement by dragging pink boxes on the root locus plot; the loop gain is adjusted to place the closed-loop pole at a specified location. A loop gains of 800 decreases overrun to 10 per cent and attains a settling time (TS) to 0.02 s. The steady-state error is equal to zero in the absence of interruptions. Research indicates that, after a step interruption, the steady-state error will ultimately diminish to zero. Utilizing the principle of overlap, it is clear that the entire reaction will exhibit an overshoot of less than 16 per cent and will reach a steady state well below 0.04 second criterion. Eliminate the manual operating option and develop a SCADA system for control and monitoring. The SCADA/HMI system was installed with CIMON software, designed for constructing HMI interfaces and trends on the touch panel, thereby assisting the operator in controlling and monitoring the manufacturing line, as shown in Figure (24).

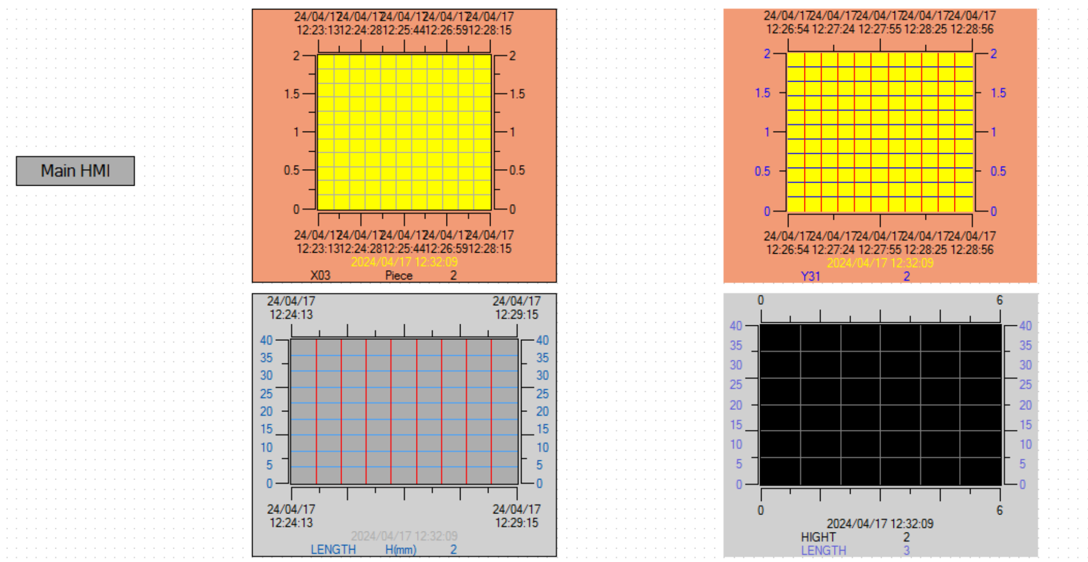


Fig. 24. Trends of height measure sensor and two photoelectric sensors.

6. Conclusion

The developed automatic production line training system can work very well with high accuracy without any downtime or malfunctions due to the problems with the wires, hoses, or the inefficiency of some sensors as were in the old system. That occurred because the entire system was developed, where the damaged hoses were replaced, in addition to implementing a new wiring system and replacing the old damaged wires. The production line can operate continuously, starting from the distribution process and ending with the last step after the inspection process in the processing station, after the control system has been replaced with a new system based on a CYMON CPU CM1-CP4A PLC to control the whole system instead of a PLC in each station and without the need to link the PLCs by using CC-links as network devices which help in overcoming the issue of the costly periodic update needed for the CC-links devices which reduced the total cost of the system by the cancellation of the other PLCs and CC-link devices. The system works in perfect timing without any delay or deviation in the turntable of the processing station by designing a controller using the Root Locus (RL) method with a MATLAB Program to achieve a settling time (TS) of 40 milliseconds and less than 16% overshoot with zero steady-state error and constant disturbance error made the motor stop precisely positioned in the drilling position. A new SCADA system has been created to control and monitor the automatic production line training system. The automatic production line training system can be applied in college and university training rooms because it is open, particularly the controlling cabinet. Students can view the inner wiring overall, and the system is easy to understand. It can work under a single station or a whole system.

References

[1] L. Monostori, "Cyber-physical production systems: Roots, expectations and R&D challenges," in Procedia CIRP, Elsevier B.V., 2014, pp. 9–13. doi: 10.1016/j.procir.2014.03.115.

[2] Industrial Informatics, 2009, INDIN 2009, 7th IEEE International Conference on : date, 23-26 June 2009. IEEE, 2009.

[3] F. Loch, G. Koltun, V. Karaseva, D. Pantförder, and B. Vogel-Heuser, "Model-based training of manual procedures in automated production systems," Mechatronics, vol. 55, pp. 212–223, Nov. 2018, doi: 10.1016/j.mechatronics.2018.05.010.

[4] H. helal Hadi and M. Y. Salloom, "Pneumatic Control System of Automatic Production Line Using Two Method of SCADA/HMI Implement PLC," Al-Khwarizmi Engineering Journal, vol. 15, no. 3, pp. 16–28, Sep. 2019, doi: 10.22153/kej.2019.06.006.

[5] J. Li, "Application Research of Vision Sensor in Material Sorting Automation Control System," in IOP Conference Series: Materials Science and Engineering, Institute of Physics Publishing, Apr. 2020. doi: 10.1088/1757-899X/782/2/022074.

[6] H. Wan, S. Ge, P. Han, F. Li, and S. Zhang, "Design and control of multifunctional sorting and training platform based on PLC control," in AIP Conference Proceedings, American Institute of Physics Inc., May 2018. doi: 10.1063/1.5039035.

[7] K. Sasidhar, S. Faiz Hussain, S. A. Safdar, and A. Uddin, "Design and Development of a PLC Based Automatic Object Sorting," 2017. [Online]. Available: www.rsisinternational.org

[8] Y. E. Li, "The Design of Practice Training System Based on PLC Programmable Automatic Control."

[9] A. Beltran et al., "Automation of Packaging and Material Handling Using Programmable Logic Controller," 2014.

[10] N. Liang and J. Xiao, "Design of Automatic Production Line Training System Based on PLC," 2013. [Online]. Available: http://www.sensorsportal.com

[11] "E3X E3X 2 Fiber-Optic Photoelectric Sensor E3X High Performance Amplifier Has Fast Response Time, Longer Sensing Distance and Self-Diagnostic Functions Ordering Information."

[12] "BEN Series."

[13] S. E. Navarro et al., "Proximity Perception in Human-Centered Robotics: A Survey on Sensing Systems and Applications," Aug. 2021, doi: 10.1109/TRO.2021.3111786.

[14] J. E. Garner, C. E. Lee, L. Huang, J. E. Garner, and L. Huang, "Photoelectric Sensors for Counting and Classifying Vehicles."

[15] "CR Series Cylindrical, Capacitive type proximity sensor."

[16] "Inductive / Capacitive Proximity Sensor."

[17] T. Grosse-Puppendahl et al., "Finding common ground: A survey of capacitive sensing in human-computer interaction," in Conference on Human Factors in Computing Systems - Proceedings, Association for Computing Machinery, May 2017, pp. 3293–3316. doi: 10.1145/3025453.3025808.

تطوير وتنفيذ نظام تدريب خط الإنتاج الآلي

علي نوري[1]*، فائز فوزي[٢]، حسين تبينة[٣]، و إيجوروف إيجور نيكولايفيتش[٤]
[١،٢]قسم هندسة التصنيع المؤتمت، كلية الهندسة الخوارزمي، جامعة بغداد، بغداد، العراق
[٣]وزارة التعليم العالي والبحث العلمي، بغداد، العراق
[٤]جامعة فلاديمير الحكومية، فلاديمير، روسيا
*البريد الالكتروني: ali.nouri2204@kecbu.uobaghdad.edu.iq

المستخلص

لقد حسنت الأتمتة الصناعية الإنتاجية والكفاءة وجودة المنتج بشكل كبير من خلال دمج التقنيات المتقدمة مثل الروبوتات والذكاء الاصطناعي وإنترنت الأشياء. تركز هذه الدراسة على إعادة تأهيل وتطوير نظام تدريب خط الإنتاج الصناعي الأوتوماتيكي DL-MS6000 في وزارة التعليم العالي والبحث العلمي في بغداد، العراق. كان النظام، الذي يتألف من ست محطات يتم التحكم فيها بواسطة ستة وحدات تحكم منطقية قابلة للبرمجة، يعاني من مشكلات، بما في ذلك أعطال الأجهزة والحاجة إلى تحديثات دورية ونقص التحكم في SCADA. استبدلت عملية التطوير وحدات التحكم المنطقية القابلة للبرمجة الستة بوحدة تحكم منطقية قابلة للبرمجة واحدة من نوع CYMON وأزالت الحاجة إلى أجهزة شبكة CC-Link، مما قلل من تعقيد النظام وتكلفته. تم تحسين النظام باستخدام خوارزمية تحكم جديدة. حقق النظام دقة عالية وخطأ حالة مستقرة صفريًا ووقت استقرار ٤٠ مللي ثانية باستخدام طريقة Root Locus في برنامج MATLAB. بالإضافة إلى ذلك، تم تنفيذ نظام SCADA/HMI لتعزيز التحكم والمراقبة. يقدم النظام المطوَّر حديثًا مزيدًا من الموثوقية والفعالية من حيث التكلفة وسهولة الاستخدام، مما يجعله أداة مناسبة للأغراض التعليمية في غرف التدريب الجامعية.