RQF LEVEL 5

CSASA501

COMPUTER SYSTEM AND ARCHITECTURE

System Automation with PLC

TRAINER'S MANUAL

October, 2024

SYSTEM AUTOMATION WITH PLC

KOICA Korea International Cooperation Agency

TQUM TVET Quality Management Project

2024

AUTHOR'S NOTE PAGE (COPYRIGHT)

Copies available from:

- *HQs: Rwanda TVET Board-RTB*
- *Web: www.rtb.gov.rw*
- **KIGALI-RWANDA**

Original published version: October 2024

ACKNOWLEDGEMENTS

The publisher would like to thank the following for their assistance in the elaboration of this training manual:

Rwanda TVET Board (RTB) extends its appreciation to all parties who contributed to the development of the trainer's and trainee's manuals for the TVET Certificate V in in Computer System and Architecture, specifically for the module "**CSA501: System Automation with PLC**"

We extend our gratitude to KOICA Rwanda for its contribution to the development of these training manuals and for its ongoing support of the TVET system in Rwanda.

We extend our gratitude to the TQUM Project for its financial and technical support in the development of these training manuals.

We would also like to acknowledge the valuable contributions of all TVET trainers and industry practitioners in the development of this training manual.
The management of Rwanda TVET Board extends its appreciation to both its staff and the staff of the TQUM Project for their efforts in coordinating these activities.

This training manual was developed:

Under Rwanda TVET Board (RTB) guiding policies and directives

Under Financial and Technical support of

COORDINATION TEAM

RWAMASIRABO Aimable

MARIA Bernadette M. Ramos

MUTIJIMA Asher Emmanuel

Production Team

Authoring and Review

NGENDAHIMANA THEOGENE

TUYISENGE JEAN LUC

Validation

UWANYIRIGIRA JANVIERE

NTAZINDA SYLVERIEN

Conception, Adaptation and Editorial works

HATEGEKIMANA Olivier

GANZA Jean Francois Regis

HARELIMANA Wilson

NZABIRINDA Aimable

DUKUZIMANA Therese

NIYONKURU Sylvestre

KWIZERA INGABIRE Diane

Formatting, Graphics, Illustrations, and infographics

YEONWOO Choe

SUA Lim

SAEM Lee

SOYEON Kim

WONYEONG Jeong

MANISHIMWE Marc

Financial and Technical support

KOICA through TQUM Project

TABLE OF CONTENT

ACRONYMS

CPU: Central Processing Unit

DCS: Distributed Control System

FBD: Function Block Diagram

HMI: Human-Machine Interface

I/O: Input/output

IEC: International Electro Technical Commission

IL: Instruction List

IOP: Input/output Processor

LAD: Ladder Diagram

PID: Proportional-Integral-Derivative (control)

PLC: Programmable Logic Controller

PPE: Personal Protective Equipment

RP: Rwanda Polytechnic

RTB: Rwanda TVET Board

RTU: Remote Terminal Unit

SCADA - Supervisory Control and Data Acquisition

SCL: Structured Control Language

ST: Structured Text

TQUM: TVET Quality management project

TVET: Technical and Vocational Education and Training

VFD: Variable Frequency Drive

INTRODUCTION

This trainer's manual includes all the methodologies required to effectively deliver the module titled **"System Automation with PLC."** Trainees enrolled in this module will engage in practical activities designed to develop and enhance their competencies.

The development of this training manual followed the Competency-Based Training and Assessment (CBT/A) approach, offering ample practical opportunities that mirror real-life situations.

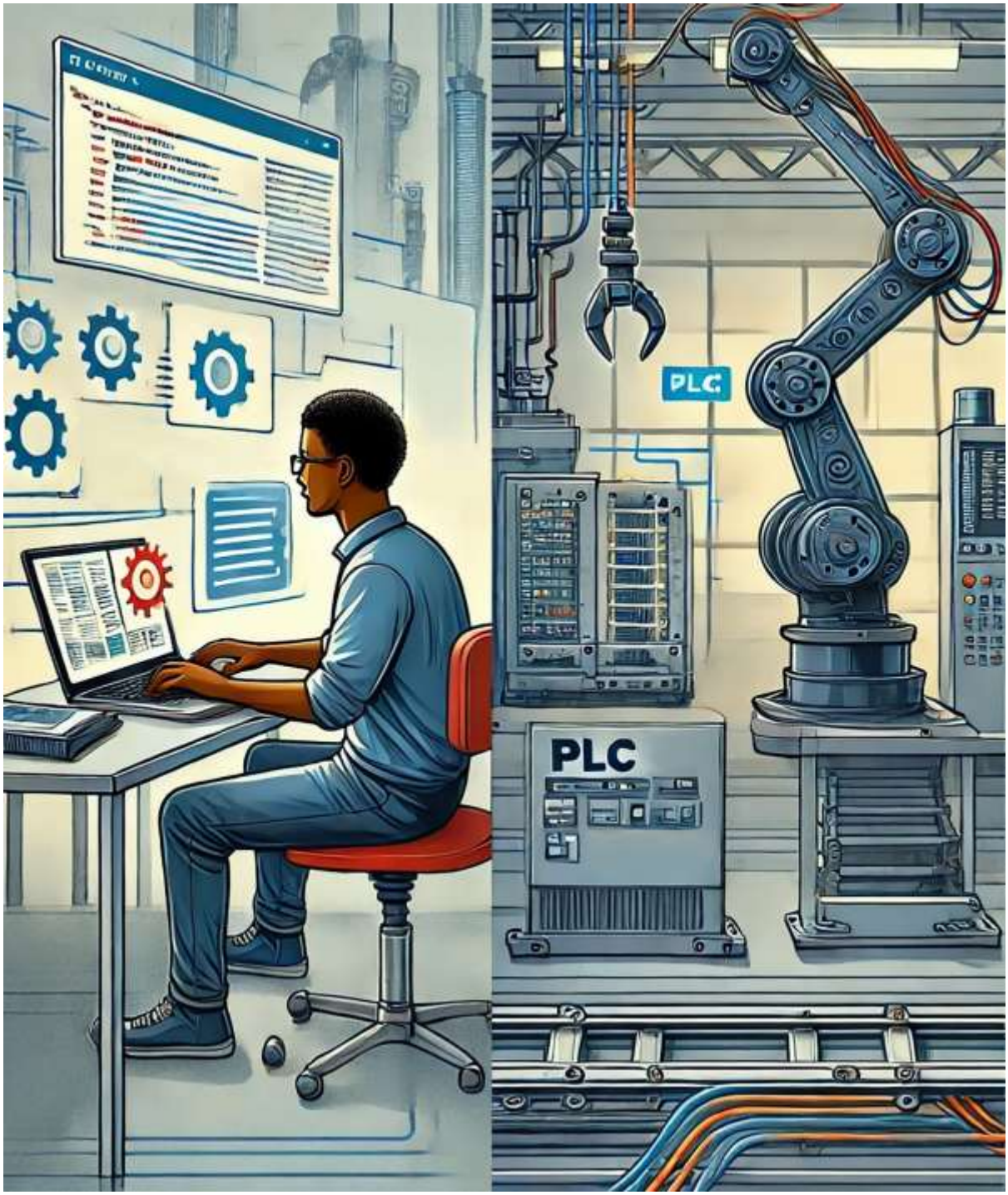
The trainer's manual is organized into Learning Outcomes, which is broken down into indicative content that includes both theoretical and practical activities. It provides detailed information on the key competencies required for each learning outcome, along with the objectives to be achieved.

As a trainer, you will begin by asking questions related to the activities to encourage critical thinking and guide trainees toward real-world applications in the labor market. The manual also outlines essential information such as learning hours, didactic materials, and suggested methodologies.

This manual outlines the procedures and methodologies for guiding trainees through various activities as detailed in their respective trainee manuals. The activities included in this training manual are designed to offer students opportunities for both individual and group work. Upon completing all activities, you will assist trainees in conducting a formative assessment known as the end learning outcome assessment. Ensure that students review the key reading and the points to remember section.

MODULE CODE AND TITLE: CSASA501 SYSTEM AUTOMATION WITH PLC

Learning Outcome 1: Prepare automation system
Learning Outcome 2: Develop PLC Program
Learning Outcome 3: Deploy PLC program


PLC
PLC

Indicative contents
1.1 Identification of automation system requirements
1.2 Analysing Automation system
1.3 Selecting of tools, materials and equipment
1.4 Designing automation system
1.5 Installation of system hardware

Key Competencies for Learning Outcome 1: Prepare automation system

Knowledge	Skills	Attitudes
• Description of system automation • Description of Programmable Logic Controller (PLC) • Description of System analysis techniques • Description of system components • Identification of System requirements • Identification of symbols and notation • Description drawing templates	• Analysing Automation system • Selecting of tools, materials and equipment • Preparation of drawing environment • Installation of a drawing tool • Applying colour scheme and styling • Drawing system architecture • Installation of system hardware • Connecting hardware components • Testing hardware connection	• Being analyst. • Having Curiosity to explore and improve existing processes. • Being problem solver. • Being Resourcefulness • Having Precision in ensuring compatibility between selected components. • Having Confidence in making decisions. • Having Creativity in system design to optimize efficiency. • Being Patient and attention to detail when installing sensitive equipment.

Duration: 20 hrs

Learning outcome 1 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Describe properly the concept of system automation as used in various industries.
2. Describe properly programmable logic controllers (PLC) as used in system automation.
3. Analyse correctly system requirements based on existing design and user needs
4. Select properly tools, materials and equipment according to automation system requirements.
5. Design properly automation system based on analysis findings.
6. Install correctly system hardware according to manufacturer guidelines and automation system design.

Resources

Equipment	Tools	Materials
● Personal computer ● PPE ● PLC ● Input and output modules ● Power supply components ● Multimeter ● Contactors	● TIA Portal software ● Screwdriver set ● Wire strippers/cutters ● Crimping pliers	● Electrical wires ● Labels ● Sensors ● Actuators ● Relays ● Conduit ● Zip ties ● Internet

Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Avail existing System drawings to analyse
- Avail computer lab
- Avail workshop

Indicative content 1.1: Identification of Automation System Requirements

Duration: 4 hrs

Theoretical Activity 1.1.1: Introduction to system automation

Notes to the trainer:

- Trainer may use small group for introducing system automation.
- The use of videos as didactic materials is required

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask trainees to answer the following questions:

i. What do you understand by system automation?
ii. Describe the types of system automation
iii. Identify he application of system automation
iv. Explain the working principles of system automation

Step 2: Ask trainees to write answers provided on flipchart/paper
Step 3: Asks trainees to present their findings
Step 4: Provides expert view
Step 5: Address any questions or concerns
Step 6: Ask trainees to read the key reading1.1.1 in the trainee manual.

Points to Remember

- System automation refers to the use of control systems, such as computers or robots, to handle processes or machines in industries to minimize human intervention.
- Automation systems can be categorized into Fixed Automation, Programmable Automation, Flexible Automation and Integrated Automation Systems.
- Automation should be used across various industries, including manufacturing, Logistics and Warehousing, Building Management, Agriculture, Healthcare, etc. to improve productivity, quality, and consistency

- Automation systems work by integrating control mechanisms, sensors, actuators, and communication networks. These systems operate based on real-time feedback and control logic to achieve high efficiency, safety, and scalability in industrial or commercial setting
- By automating systems, industries can optimize productivity, enhance safety, reduce operational costs, and increase precision, creating a more reliable and efficient environment.

Theoretical Activity 1.1.2: Description of Programmable Logic Controller (PLC)

Notes to the trainer:

- While delivering this content, a small group can be used for describing Programmable Logic Controller (PLC)

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask trainees to answer the following questions:

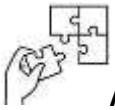
i. What do you understand by Programmable Logic Controller?
ii. Describe the types of Programmable Logic Controller
iii. Identify he application of Programmable Logic Controller
iv. Identify the Characteristics of Programmable Logic Controller

Step 2: Ask trainees to write answers provided on flipchart/paper
Step 3: Asks trainees to discuss the provided answer and choose correct answers
Step 4: Provides expert view and clarifies ideas
Step 5: Address any questions or concerns
Step 6: Ask trainees to read the key reading 1.1.2 in the trainee manual.

Points to Remember

- A Programmable Logic Controller (PLC) is an industrial computer-based device used for controlling processes or machines in the manufacturing and production industry
- PLCs can be classified based on various factors such as size, input/output capacity, functionality, and structure. Some types include Compact PLC, Modular PLC, Rack-Mounted PLC, Soft PLC, Safety PLCs, etc.

- PLCs are versatile devices used in many industries to automate processes, control machinery, and ensure efficient, safe, and precise operation of systems. Some key application areas of PLCs are Manufacturing and Assembly Lines, Process Control in Chemical and Food Industries, Water Treatment and Wastewater Management, Energy Management and Power Generation, Automotive Industry, etc.
- PLCs possess several key characteristics that make them essential in industrial automation and process control applications. These characteristics are crucial for ensuring that PLCs can operate efficiently and reliably in demanding environments. Most important characteristics of PLCs include Programmability, Real-Time Operation, Reliability, Scalability and Flexibility, Communication Capabilities, etc.



Application of learning 1.1

As an automation technician at a food processing plant, your primary responsibility is to automate the packaging line to reduce waste and enhance accuracy. You start by thoroughly analyzing the existing packaging process to identify inefficiencies. Based on your findings, you recommend integrating Programmable Logic Controllers (PLCs) to improve control over the packaging machinery. You then define key requirements for the new system, focusing on accessibility and maintainability to minimize downtime and facilitate quick troubleshooting by operators

Checklist:

SN	Criteria	Indicators	Yes	No
1	Process Analysis is well followed	1.1 Existing packaging process is thoroughly analyzed for inefficiencies.		
		1.2 Key inefficiencies in the current system are documented.		
		1.3 Stakeholder input is gathered to identify critical issues.		
2	Integration of PLCs are well planed	2.1 Recommendations for PLC integration are based on the analysis of the existing process.		
		2.2 Selected PLC model is appropriate for controlling packaging machinery.		
		2.3 Required I/O points for the new system are identified and sufficient.		

3	System Requirements are properly identified	3.1 Accessibility requirements are defined to ensure ease of use by operators.		
		3.2 Maintainability requirements are established to minimize downtime.		
		3.3 Performance targets for speed and accuracy are set for the new system.		
4	Implementation Planning are well established	4.1 Detailed implementation plan is developed for integrating the new system.		
		4.2 Timeline for system integration is established.		
		4.3 Resources (personnel, equipment) for implementation are allocated.		
5	Testing and Validation are properly done	5.1 Initial tests are planned to verify the new system's performance against defined targets.		
		5.2 System is validated to ensure it meets accessibility and maintainability criteria.		
		5.3 Adjustments are made based on test results and operator feedback.		

Indicative content 1.2: Analysing Automation System

Duration: 4 hrs

Theoretical Activity 1.2.1: Description of automation system analysis

Notes to the trainer:

- While delivering this content, a small group can be used for Analyzing the existing System
- You need to have Flowchart, Data Flow Diagrams, Fault Tree for same existing system

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask trainees to answer the following questions:

i. What does automation system analysis means?
ii. Identify automation system analysis techniques
iii. List automation system components
iv. Analyze system requirements

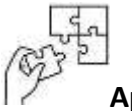
Step 2: Ask trainees to write answers provided on flipchart/paper.
Step 3: Asks trainees to discuss the provided answer and choose correct answers
Step 4: Provides expert view and clarifies ideas
Step 5: Address any questions or concerns
Step 6: Ask trainees to read the key reading 1.2.1 in the trainee manual.

Points to Remember

- System analysis is essential for understanding the current functionality and identifying areas for automation improvement. In PLC-based system automation, the following techniques are commonly used: Flowcharting, Data Flow Diagrams (DFD), Process Mapping, Fault Tree Analysis.
- In any PLC-based automated system, understanding the components is critical to analysing and improving the system. The key components include: Input Devices like

sensors, push buttons, and switches and central control unit**(PLC),** Output Devices like motors, actuators, lights, or valves and Human-Machine Interface (HMI), Communication Modules.

- In system automation using PLCs, identifying the user's needs helps ensure that the automated system is designed to solve real operational challenges. Key steps in identifying user needs include Process Improvement, Safety Requirements, Performance and Efficiency, Flexibility and Scalability, Integration Needs



Application of learning 1.2:

You are tasked with designing a PLC-based automation system for a packaging plant that moves products along a conveyor belt.

i. Which adjustments could you make to ensure the system meets the performance requirement of adjusting speed without products falling off the belt?

ii. **Answer:** The PLC could be programmed to gradually accelerate or decelerate the conveyor belt rather than making sudden speed changes. This would prevent products from falling off.

iii. How would you design the HMI to make sure the system is easy to use for operators who need to start, stop, and adjust the speed?

iv. **Answer:** The HMI could have simple buttons for start, stop, and speed adjustment, clearly labelled in multiple languages. It should also display real-time data, such as current speed and error messages, for easy operation.

v. What features would you implement to ensure the system can recover from an unexpected motor stop?

Answer: The PLC should store the current state of the conveyor belt (e.g., speed and position) so that it can resume where it left off after the error is resolved. An alarm should notify operators if the system does not recover within the 10-second window.

Indicative content 1.3: Selecting of Tools, Materials and Equipment

Duration: 4hrs

Theoretical Activity 1.3.1: Identification of tools, materials and equipment

Notes to the trainer:

- Trainer must avail all tools, materials and equipment that will be shown to the student as didactic material

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request trainees to answer the following questions:

i. Define the following terms:
   a. Tool
   b. Material
   c. Equipment

ii. Identify the examples and functions of tool material and equipment used in automation system

Step 2: Asks any trainees to write answers provided on flipchart/paper.
Step 3: Asks trainees to discuss the provided answer and choose correct answers.
Step 4: Provides expert view and clarifies ideas by using didactic materials.
Step 5: Address any questions or concerns.
Step 6: Ask trainees to read the key reading 1.3.1 in the trainee manual

Points to Remember

- When working with PLC systems, various tools are required for installation, programming, testing, and maintenance. Some essential tools include:
  - **Programming Tools:** PLC Programming Software, PC or Laptop
  - **Wiring and Electrical Tools:** Screwdrivers, Wire Strippers, Multimeter, Crimping Tool

- **Diagnostic Tools:** PLC Simulator or Tester, Logic Probes and Oscilloscopes

- Materials refer to consumables and components that are used to build and maintain a PLC-based automation system. Key materials include:
  - **Wiring and Cables:** Control Wiring, Power Cables, Network Cables (Ethernet, RS-485)
  - **Connectors and Terminals:** Terminal Blocks, Wire Connectors
  - **Electrical Components:** Relays and Contactors, Fuses and Circuit Breakers
- In addition to tools and materials, specialized equipment is required to ensure the successful operation and control of PLC-based automation systems. Important equipment includes:
  - **Programmable Logic Controller (PLC)**
  - **Input Devices:** Sensors, Push Buttons and Switches
  - **Output Devices:** Actuators, Motors, Valves
  - **Human-Machine Interface (HMI)**
  - **Power Supply Units (PSUs)**
  - **Communication Modules:** Network Routers, Hubs, and Switches

Practical Activity 1.3.2: Selecting tools, materials and equipment

Notes to the trainer

- The trainer may avail tools, materials and equipment used in automation system.
- Avail workshop.

Key steps:

While delivering this content pass through the following steps

Step 1: Introduce the topic and ask trainees to perform the task described below should
As an automation technician, you have been assigned to select tools, materials and equipment that will be used in designing automation system.

Step 2: Demonstrate and explain each procedure while selecting tools, materials and equipment

Step 3: Ask trainee to select tools, materials and equipment.

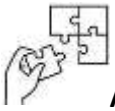
Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

Step 6: Lead them to read key reading 1.3.2 in their manual

Points to Remember

- Selecting the Right Tools for Automation, one of the most important decisions in prototype automation is choosing the right tools for the job. There are many factors to consider, such as the type of prototype, the complexity of the tasks, the budget, the skills of the team, and the desired outcomes.



Application of learning 1.3:

Imagine you are assigned to automate a water pump system in a small irrigation project. The goal is to automatically start and stop the pump based on soil moisture levels detected by a sensor. The system will also allow manual control through a push button and provide a status display for the operator. You need to select the appropriate tools, materials, and equipment to build and maintain this PLC-based system.

Checklist:

SN	Criteria	Indicators	Yes	No
1	Tools, material and equipment are properly selected	1.1 Tools are selected		
		1.2 Materials are selected		
		1.3 Equipment are selected		

Indicative content 1.4: Designing Automation System

Duration: 4 hrs

Theoretical Activity 1.4.1: Description of drawing environment

Notes to the trainer:

- Small group can be used for describing drawing environment.

Key steps:

While delivering this content pass through the following steps:

Step 1: Introduce the activity and ask trainees to answer the following questions:

i. What does drawing environment means?
ii. Identify standard symbols and notations for drawing system diagrams
iii. What do you understand by the term drawing templates?
iv. Explain how to apply colour scheme and styling

Step 2: Ask trainees to write answers provided on flipchart/paper.
Step 3: Asks trainees to discuss the provided answer and choose correct answers.
Step 4: Provides expert view and clarifies ideas.
Step 5: Address any questions or concerns.
Step 6: Ask trainees to read the key reading 1.4.1 in the trainee manual.

Points to Remember

- Select a Suitable Tool: like Microsoft Visio, Lucidchart, Draw.io, EPLAN, or AutoCAD
- Using standard symbols and notations in diagrams ensures that they are universally understood by other engineers and technicians. So you have to know Hardware Block Diagrams, Flowcharts, Architecture Diagrams, Sequence Diagrams symbols
- Using templates helps standardize your diagrams and speeds up the design process. Most diagramming tools offer ready-to-use templates specific to system automation, PLC programming, and control systems.

- Adding colors and styles to your diagrams can improve clarity and readability. It also helps highlight different parts of the system, making it easier to understand complex processes.

Practical Activity 1.4.2: Installing a drawing tool

Notes to the trainer

- Avail one of the drawing software related to the tasks
- Avail computer lab

Key steps:

While delivering this activity, pass through the following steps

Step 1: Introduce the activity and ask trainees to read the task described below:
As computer system architecture trainee, you are asked to install Draw.io
Step 2: Explain the task and provide clear work instruction.
Step 3: Demonstrate how to install Draw.io, explain the steps to follow.
Step 4: Ask trainee to perform a task by following the demonstrated steps verify if the draw.io is properly installed
Step 4: Ask trainees to read key reading 1.4.1.

Points to Remember

- **Steps to install Draw.io**
    - Download the Draw.io Installer
    - Run the Draw.io Installer
    - Launch Draw.io

Practical Activity 1.4.3: Drawing system architecture

Notes to the trainer

- The trainer may use individuals in Drawing system architecture.
- Avail of software and tutorials as didactic materials is required.
- Avail computer Lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask trainees to perform the task described below should be done individually:

You are an automation technician tasked with enhancing the packaging line in a food processing plant. The goal is to streamline operations, reduce waste, and improve the accuracy of packaging.

Tasks:

i. Identification of System Components
ii. Create Hardware Block Diagram
iii. Create Functional Flow Diagram
iv. Create an Architecture Diagram
v. Dataflow and Communication (Sequence Diagram)

Step 2: Demonstrate and explain each procedure while performing a Drawing system architecture.

Step 3: Distributes individual PLC hardware to the students.

Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

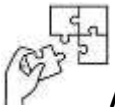
Step 6: Lead them to read key reading 1.4.2 in their manuals

Points to remember

- Identification of System Components:

- Identify all essential components involved in the packaging line. This may include sensors, actuators, controllers, conveyors, human-machine interfaces (HMIs), and packaging machines.
- Create Hardware Block Diagram:
- Develop a hardware block diagram representing the primary components of the system and their connections. This visual will help clarify the physical setup of the automation hardware.
- Create Functional Flow Diagram:
- Construct a functional flow diagram to illustrate the sequence of operations in the packaging line. Include steps such as product detection, conveyor movement, packaging, quality control, and any error-handling mechanisms.
- Create an Architecture Diagram:

Design an architecture diagram to show how different hardware and software components interact within the system. This diagram will highlight the structural layout and connectivity of components and systems.



Application of learning 1.4.

As an automation technician, you are assigned to enhance the bottling line in a beverage production facility. Your primary objective is to streamline operations, minimize product spillage, and improve the accuracy of bottle filling and labelling.

Checklist:

SN	Criteria	Indicators	Yes	No
1	Components are properly identified	1.1 All system components (conveyor belts, sensors, actuators, PLC, etc.) are identified and documented.		
		1.2 Specifications for each component (capacity, compatibility) are gathered.		
		1.3 Components are categorized based on their function in the packaging process.		
2	Hardware Block Diagram is correctly created	2.1 A block diagram illustrating the relationship between the PLC and other components is created.		
		2.2 All connections (wiring, communication protocols) are accurately represented in the diagram.		

		2.3 The diagram is reviewed for clarity and completeness.		
3	Functional Flow Diagram is correctly created	3.1 A flow diagram outlining the packaging process steps is developed.		
		3.2 Decision points (e.g., quality checks) are included in the flowchart.		
		3.3 The flow diagram is validated against the current process for accuracy.		
4	Architecture Diagram is correctly created	4.1 An architecture diagram depicting the overall system setup is created.		
		4.2 The diagram includes the integration of various subsystems and communication pathways.		
		4.3 The architecture diagram is reviewed by team members for feedback.		
5	Dataflow and Communication (Sequence Diagram) is correctly created	5.1 A sequence diagram mapping data flow and communication between components is constructed.		
		5.2 The diagram illustrates the sequence of messages exchanged during the packaging process.		
		5.3. The sequence diagram is checked for completeness and alignment with functional requirements.		

Indicative content 1.5: Installation of System Hardware

Duration: 4 hrs

- **Practical Activity 1.5.1: Installing system hardware**

Notes to the trainer

- The trainer may use individuals in installing system hardware
- Avail of software, PLC hardware and tutorials as didactic materials is required.
- Avail computer Lab.

While delivering this activity, pass through the following steps:

Step1. Introduce the topic and ask trainees to perform the task described below should be done individually:

As an automation technician, you have been assigned to install a robotic arm in an assembly line of a manufacturing facility to enhance production efficiency and accuracy. Your tasks involve preparing the installation site, connecting the necessary hardware components, and testing the hardware connections to ensure seamless integration with the existing system.

Step 2: Demonstrate and explain each procedure while installing system hardware
Step 3: Distributes individual PLC hardware to the students.
Step 4: Assess their performance and provide feedback.
Step 5: Allow trainees to ask questions or clarifications
Step 6: Ask them to read key reading 1.5.1 in their manual

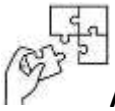
Points to Remember

- **Installing System Hardware follow these steps:**

✓ Prepare Installation Site

✓ Connect Hardware Components

✓ Test Hardware Connection



Application of learning 1.5.

As an automation technician, you are responsible for installing a new automated sorting system in a warehouse aimed at improving inventory management and order fulfilment efficiency. Your tasks include preparing the installation site, connecting hardware components, and testing the system to ensure it operates correctly.

Check list

SN	Criteria	Indicators	Yes	No
1	Installation Site is properly prepared	1.1 The installation area is free of debris and obstacles.		
		1.2 Sufficient space is available for equipment and operation.		
		1.3 The floor is level and strong enough to support the equipment.		
		1.4 Proper ventilation is provided in the installation area.		
		1.5 Adequate lighting is available to ensure visibility during installation.		
		1.6 Safety measures (emergency stops, barriers) are in place.		
2	Hardware Components are correctly connected	2.1 All hardware components are unpacked and organized.		
		2.2 Wiring diagrams are reviewed before making connections.		
		2.3 Power supply is properly routed to all devices.		
		2.4 All sensors, actuators, and communication lines are securely connected.		
		2.5 Color codes and labeling on wiring are followed correctly.		
3	System Operation is properly tested	3.1 The system is powered on successfully without errors.		

		3.2 Each component of the sorting system operates as expected.		
		3.3 Sensors accurately detect items for sorting.		
		3.4 Data transfer between the control system and sorting mechanism is verified.		
		3.5 Any issues encountered are documented for troubleshooting.		
4	Final Review	4.1 The system meets operational specifications and performance criteria.		
		4.2 Installation is complete, and all components are securely mounted.		
		4.3 User manuals and documentation are provided to relevant personnel.		
		4.4 Staff training is scheduled for using and troubleshooting the new system.		

Learning outcome 1 end assessment

Theoretical assessment

A. Multiple-Choice Questions

Read carefully the following statements and choose by encircling the letter corresponding to the right answer

a. Which of the following best defines automation?
   a) Increasing the need for human labor
   b) Implementing systems that operate automatically
   c) Decreasing productivity in manufacturing
   d) Focusing solely on aesthetic improvements

   Answer: b) Implementing systems that operate automatically

b. Which type of automation is ideal for high-volume, repetitive tasks?
   a) Flexible automation
   b) Programmable automation
   c) Fixed automation
   d) Batch automation

   Answer: c) Fixed automation

c. What is the primary purpose of a Programmable Logic Controller (PLC)?
   a) Controlling lighting in residential homes
   b) Managing office tasks
   c) Automating industrial machinery
   d) Enhancing manual control of systems

   Answer: c) Automating industrial machinery

d. Which system requirement focuses on the system's ability to recover after a malfunction?
   a) Accessibility
   b) Usability
   c) Maintainability
   d) Recoverability

Answer: d) Recoverability

e. In system automation, the "usability requirement" primarily refers to:
    a) System affordability
    b) Ease of use for users
    c) Recovery options after failure
    d) System speed

    Answer: b) Ease of use for users

f. When analyzing an existing system, which of the following is commonly identified?
    a) System marketing strategies
    b) User needs and system components
    c) Product pricing
    d) Employee training needs

    Answer: b) User needs and system components

g. Which type of system diagram shows the sequence of data and interactions between components?
    a) Functional flow diagram
    b) Sequence diagram
    c) Hardware block diagram
    d) Architecture diagram

    Answer: b) Sequence diagram

h. What is the first step when designing an automation system drawing environment?
    a) Choosing a color scheme
    b) Selecting tools and equipment
    c) Identifying symbols and notations
    d) Installing a drawing tool

    Answer: d) Installing a drawing tool

i. In selecting tools, materials, and equipment for automation, which is considered a "tool"?
    a) PLC
    b) Flowchart software
    c) Circuit wiring
    d) Sensors

Answer: b) Flowchart software

j. What is the main purpose of testing hardware connections during system installation**?**
   a) To ensure all components are properly connected and functioning
   b) To improve system appearance
   c) To finalize system documentation
   d) To increase system cost

Answer: a) To ensure all components are properly connected and functioning

B. Fill-in-the-gap questions

Read the following statement and complete them by using correct terms

1. The main objective of identifying **automation system requirements** is to ensure that the system meets the user's __ **needs** _____ and performs effectively.
2. A **Programmable Logic Controller (PLC)** is commonly used in industrial applications to control ____ **automated** ____ and ___ **mechanical** _____ processes.
3. In system analysis, understanding the **working principles** helps determine how the automation system will operate and ___ **respond** _____ in different scenarios.
4. System **performance requirements** are aimed at ensuring that the automation system can meet the necessary ___ **efficiency** _____ standards.
5. The **usability requirement** of an automation system ensures that it is easy for users to _____ **understand** ___ and operate.
6. When **selecting tools, materials, and equipment** for an automation project, ___ **tools** _such as PLC programming software are necessary for system design and control.
7. In designing an automation system, installing a drawing tool allows the designer to create accurate ___ **flow** _____ and ____ **architecture** ____ diagrams.
8. The **hardware block diagram** in an automation system is used to represent the system's ___ **physical** _____ components and their connections.
9. **System analysis techniques** are used to evaluate the existing system's ___ **capabilities** __, limitations, and potential improvements.
10. Testing hardware connections during the installation phase ensures that all components are properly connected and ____ **functioning** ____ as expected.

C. OPEN QUESTIONS

1. What are the primary benefits of implementing automation in industrial processes, and how does it affect productivity and efficiency?

Answer: Automation enhances productivity by increasing speed, consistency, and accuracy in industrial processes. It reduces human error, improves efficiency by maintaining steady output, and enables systems to operate continuously, often leading to increased production and reduced operational costs.

2. Explain the different types of automation systems and provide examples of where each type might be most effectively applied.
   Answer: There are three main types of automation:

- **Fixed automation** is used for high-volume, repetitive tasks, such as in assembly lines for manufacturing cars.
- **Programmable automation** is more flexible and can be reprogrammed for different tasks, suitable for batch production, like clothing manufacturing.
- **Flexible automation** can handle varying tasks without reprogramming, ideal for custom or small-batch production, like robotic arms in advanced manufacturing.

3. Describe the role of a Programmable Logic Controller (PLC) in automation. How does it differ from traditional control systems?
   Answer: A PLC is used to automate industrial processes by monitoring input devices and making real-time decisions to control output devices. Unlike traditional control systems, PLCs are highly durable, programmable, and adaptable to different applications, which makes them ideal for complex and evolving industrial tasks.
4. When analyzing an existing automation system, what system analysis techniques would you use to evaluate its effectiveness?
   Answer: System analysis techniques include flowcharting to map processes, data analysis to evaluate performance, and gap analysis to identify areas for improvement. Stakeholder interviews and user feedback can provide insights into user needs, while benchmarking helps compare system performance against industry standards.
5. What are the key factors to consider when identifying the performance requirements of an automation system?
   Answer: Performance requirements should consider processing speed, accuracy, reliability, and scalability. Factors like response time, throughput, and efficiency are critical in ensuring that the system meets the operational demands effectively and consistently.
6. How do usability, recoverability, and maintainability requirements contribute to the overall success of an automation system?
   Answer: Usability ensures that operators can easily interact with the system, minimizing errors and training needs. Recoverability allows the system to quickly return to normal

operations after failures, reducing downtime. Maintainability ensures that the system is easy to repair and upgrade, which is crucial for long-term sustainability.

7. What criteria would you use to select tools, materials, and equipment for designing and implementing an automation system?
   Answer: Selection criteria should include compatibility with system requirements, durability, cost, ease of use, and the ability to support scalability. For example, tools should allow precise configuration, materials should be reliable, and equipment should be compatible with the automation processes to ensure efficiency and longevity.
8. Discuss the importance of creating an accurate hardware block diagram during the system design phase. What information should it convey?
   Answer: A hardware block diagram provides a visual representation of the system's components and their interactions, helping to ensure correct component selection and layout. It should convey the connections between hardware components, power sources, signal pathways, and control modules, offering a clear guide for assembly and troubleshooting.
9. How does the use of dataflow and communication diagrams (e.g., sequence diagrams) enhance the understanding of an automation system's functionality?
   Answer: Dataflow and sequence diagrams illustrate the flow of data and communication within the system, clarifying how components interact and in what sequence. This enhances understanding of system behavior, helps identify bottlenecks, and assists in debugging and optimizing communication processes within the system.
10. Explain the process and importance of testing hardware connections during the installation of an automation system. What steps would you take to troubleshoot any issues?
    Answer: Testing ensures all components are properly connected and communicating. Key steps include visually inspecting connections, running diagnostic tests, and measuring signal levels. If issues arise, troubleshooting steps may involve isolating components to identify the faulty area, checking for loose connections, or reviewing compatibility between connected devices.

Practical assessment

As an automation technician, you are tasked with Designing and install a PLC-based control system for a conveyor belt system in a manufacturing facility. The system must include the following functionalities:

1. Control the conveyor belt motor based on inputs from proximity sensors.
2. Stop the conveyor when a part is detected at the end of the line.
3. Include emergency stop functionality.

SN	Criteria	Indicators	Yes	No
1	**Installation Site is properly prepared**	Site is cleared of debris and obstacles.		
		Location is accessible for maintenance.		
		Proximity to existing power sources and wiring is verified.		
		Safety protocols are in place for personnel.		
2	**Hardware Components are correctly connected**	Control panel is securely mounted		
		Wiring connections are made according to the manufacturer's diagram.		
		All connections are insulated to prevent short circuits.		
		Power supply is properly routed to the control panel.		
		Communication cables are linked correctly to the embedded system.		
3	**Hardware Connections are properly tested**	Power is turned on, and the control system initializes without errors.		
		Sensors read input data (e.g., temperature, proximity) accurately.		
		Actuators function correctly based on sensor input.		
		PLC communicates effectively with all connected hardware.		
		System operates as intended based on input conditions.		
		Documented test results, including any troubleshooting steps taken.		

Further information to the trainer

Books

Alves, T. R., Buratto, M., De Souza, F. M., & Rodrigues, T. V. (2014, October). OpenPLC: An open source alternative to automation. In *IEEE Global Humanitarian Technology Conference (GHTC 2014)* (pp. 585-589). IEEE.

Francesco, Pasquale Chiacchio, and Diego Gerbasio. "On the implementation of industrial automation systems based on PLC." *IEEE Transactions on Automation Science and Engineering* 10, no. 4 (2012): 990-1003.

Ye, L. (2024). Design of PLC Electrical Control System. *Journal of Theory and Practice of Engineering Science*, *4*(03), 134-162.

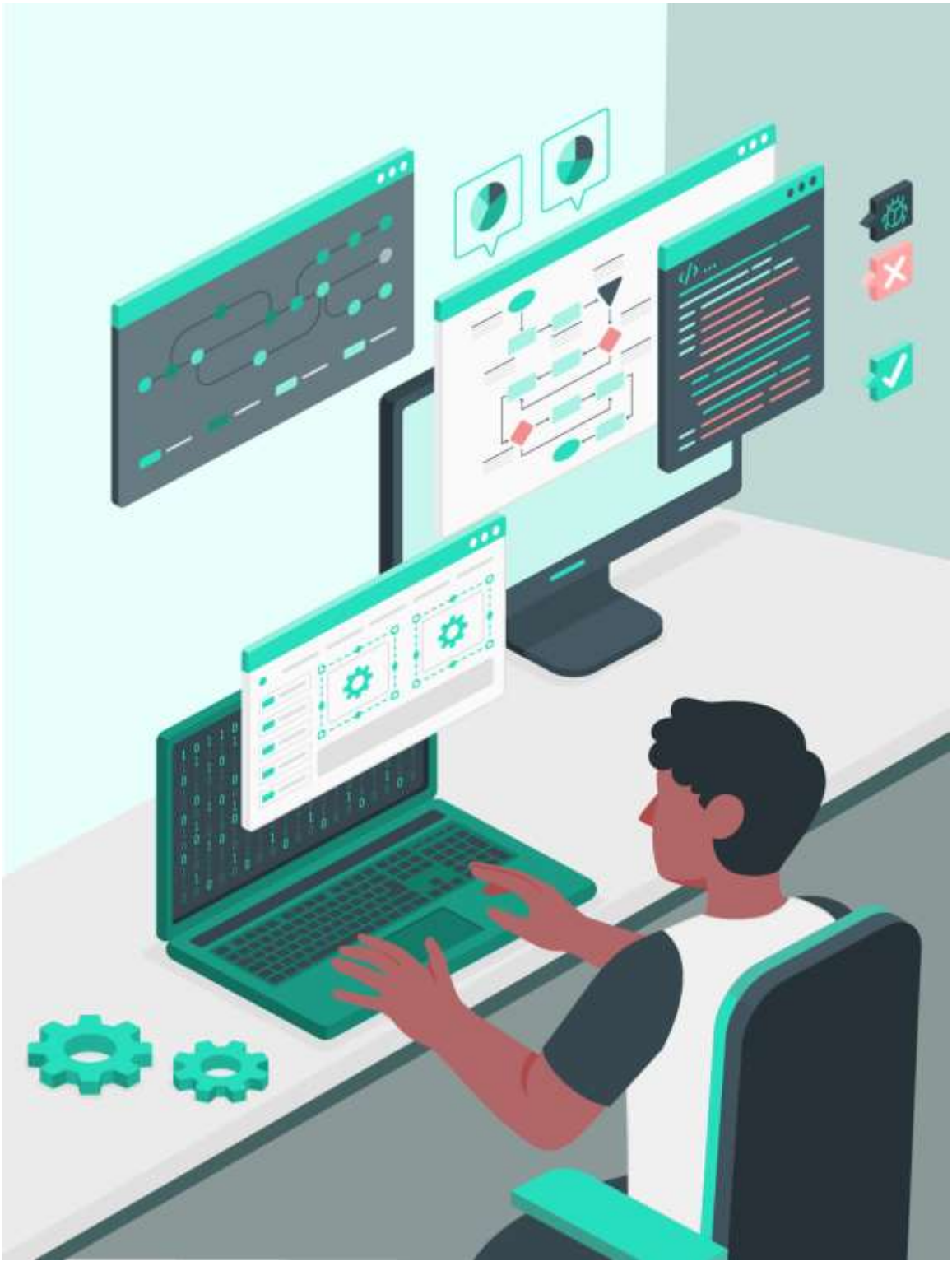
Yuanyushkin AS, Lobanov DV, Rychkov DA. Automation tool preparation in the conditions of production. Applied mechanics and materials. 2015 Jul 20; 770:739-43.

Web links

Https://www.solisplc.com/tutorials.

https://instrumentationtools.com/plc-scan-time/#:~:text=Definition%20of%20PLC%20Scan%20Time,to%20a%20large%20till%201000ms.

.



Indicative contents

2.1 Installation of PLC simulation software

2.2 Selecting PLC programming languages

2.3 Writing PLC program

2.4 Converting PLC programming language

Key Competencies for Learning Outcome 2: Develop PLC Program

Knowledge	Skills	Attitudes
• Description of PLC Architecture • Description of Programming Languages (Ladder Logic, FBD, ST) • Description of Control System Concepts	• Installing PLC software • Programming using Ladder Logic, FDB and ST • Using PLC Software tool • Wiring PLC program Blocks • Configuring PLC program Blocks • Testing PLC Program • Simulating PLC Program	• Being Attentive to Detail • Being a Problem-Solver • Being Adaptive • Being Safety-Oriented • Being Patience and Persistence

Duration: 40 hrs

Learning outcome 2 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Describe properly PLC architecture based on PLC type
2. Install correctly PLC simulation software based on the system to be automated
3. Select properly PLC programming languages based on the system to be automated
4. Write correctly PLC program based on automated system
5. Convert properly PLC programming language based on automated system required output

Resources

Equipment	**Tools**	**Materials**
• Computer • Input and output modules • Power supply components • PLC • Multimeter • Personal computer • PPE • Contactors • PLC modules (e.g., Siemens S7-1200)	• TIA Portal software • Automation Studio • Screwdriver set • Wire strippers/cutters • Crimping pliers	• Internet bundle • Electrical wires • Labels • Sensors • Actuators • Relays • Conduit • Zip ties • Internet • White board • Marker

Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Ensure all necessary tools and equipment for Developing PLC Program are available and in working condition.
- Review the safety procedures and protocols to be followed during the PLC program Development.
- Set up a clean, organized workspace to learning and practice.

Indicative content 2.1: Installation of PLC Simulation Software

Duration: 7hrs

Practical Activity 2.1.1: Installing PLC simulation software

Notes to the trainer

- The trainer may use individuals in Drawing system architecture.
- Avail of software and tutorials as didactic materials is required.
- Avail computer Lab.

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask trainees to perform the task described below should be done individually:

You are asked to go in computer lab and perform the following tasks:

i. Prepare simulation software installation (TIA Portal).
ii. Install simulation software (TIA Portal)
iii. Perform system compatibility check up

Step 2: Demonstrate and explain each procedure while performing a Drawing system architecture.

Step 3: Ask trainees to perform the given task and monitor them

Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

Step 6: Ask them to read key reading 2.1.1 in their manual

Points to Remember

- **Steps of installing Simulation software (TIA Portal)**

  Step 1. Create account on Siemens

  Step 2. Click Register and create your account

Step 3: Fill the form and create your account.

Step 4: After downloading the files, move them all into a single directory and then double-click on the "TIA_Portal_STEP7_Prof_ Safety_WINCC_Adv_Unified_V17.exe" to start the installation

Step 5 – Install PLCSIM

Step 6 – Write your first PLC program

i. Performing a system compatibility check for TIA Portal involves several steps to ensure that your hardware and software meet the necessary requirements. Here's a step-by-step guide:

- **Steps for System Compatibility Check:**

  Step 1: Check System Requirements

  Step 2: Check Operating System Compatibility

  Step 3: Install Required Software

  Step 4: Perform Compatibility Check

  Step 5: Analyze Results

  Step 6: Post-Installation Verification

- **Configure Software Packages:**

  After installation, open TIA Portal, and configure the environment for your simulation or project development.

  - ✓ Create a New Project
  - ✓ Configure PLC Simulator (PLCSIM):
  - ✓ Configure HMI (WinCC)
  - ✓ Install Updates and Add-ons:
  - ✓ Run the Simulation:
  - ✓ Test the Configuration:
  - ✓ Save and Document Configuration

Application of learning 2.1.

The XYZ Logistics Warehouse is upgrading its automated conveyor system to increase efficiency in packaging and sorting products. The existing system uses a Siemens S7-1200 PLC for control, and the new upgrade involves integrating an automated barcode scanner and a new conveyor belt drive**.** The barcode scanner comes with its own software, while the conveyor belt requires precise speed control through a Variable Frequency Drive (VFD), also connected to the PLC. Your task is to prepare simulation software installation (TIA Portal) and after perform system compatibility check up by analysing the current setup and ensuring the new barcode scanner software and conveyor VFD can communicate with the existing Siemens S7-1200 PLC

Checklist:

SN	Criteria	Indicators	Yes	No
1	PLC simulation software is properly installed	1.1 Steps of installing Simulation software (TIA Portal) are followed		
		1.2 Steps for System Compatibility Check are followed		
		1.3 Software Packages are Configured		

Indicative content 2.2: Selecting PLC Programming Languages

Duration : 10 hrs

Theoretical Activity 2.2.1: Description of PLC programming languages

Notes to the trainer:

- The small groups may be used to Describe PLC programming languages.
- The images, videos, and illustrations may also be used as didactic materials.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and Invite the trainees to answer the following questions in the small groups:

i. What are the types PLC programming languages?
ii. What are the application of PLC programming languages?
iii. What are the characteristics of PLC programming languages?
iv. What are the symbols and notations of PLC programming languages?

Step 2: Ask the trainees to presents findings from their small groups

Step 3: Refers to the trainee's answers, provide and clarify the expert view.

Step 4: Direct trainees to read the key reading 2.2.1 in their manuals for more details PLC programming languages.

Points to Remember

PLC programming languages are used to instruct PLCs (Programmable Logic Controllers) on how to control machinery, processes, and other automated systems.

Types PLC programming languages

- **Ladder Diagram (LD**)

Application:

Ladder diagrams are widely used in industrial control systems, especially in the field of automation and electrical engineering. Here are some key applications of ladder diagrams:

- ✓ Motor Control
- ✓ Conveyor Systems Temperature Control
- ✓ Pressure Control Emergency Stop Circuits
- ✓ Sequential Control

Symbols and Notations:

- **Contact (--| |--)**: Represents input conditions such as switches or sensors.
- **Coil (-- () --)**: Represents output devices like motors or actuators.
- **Timers (TON, TOF)**: For delay or time-based operations.
- **Counters (CTU, CTD)**: For counting up or down events.

- **Function Block Diagram (FBD)**

AND **Gate**: Symbol: A rectangle labeled AND.
OR **Gate**: Symbol: A rectangle labeled OR.
NOT **Gate (Inverter)**: Symbol: A rectangle labeled NOT with a single input.
XOR **Gate**: Symbol: A rectangle labeled XOR.
NAND**/NOR Gates**: Symbols: Rectangles labeled NAND or NOR.

- **Structured Text (ST)**

Application:

- Used in process automation, continuous processes, and control systems involving analog signals and PID controllers.
- Popular in industries like chemical, food processing, and water treatment, temperature regulation, **Monitoring and Diagnostics**.

Symbols and Notations:

IF, THEN, ELSE: Conditional statements.

FOR, WHILE: Loop control.

+ - * /: Arithmetic operators for mathematical operations.

=: Assignment operator for variable storage.

- **Sequential Function Chart (SFC)**

Application:

- Typically used in batch processing, sequential operations (e.g., automated assembly lines), or any process that follows a defined sequence.
- Ideal for controlling processes that have distinct states or stages.

Symbols and Notations:

- **Step (□)**: Represents a discrete phase in the process.
- **Transition (→)**: Represents conditions that must be met to move to the next step.
- **Actions**: Associated with each step, defining what actions are to be taken.

- **Instruction List**

Application:

- Suitable for simple operations, particularly in legacy systems or when performance optimization is critical.
- Less commonly used in modern PLC systems but still found in certain applications requiring compact code.

Symbols and Notations:

- **LD**: Load instruction.
- **ST**: Store instruction.
- **AND, OR, NOT**: Logical operations.
- **ADD, SUB, MUL**: Arithmetic instructions.

Theoretical Activity 2.2.2: Description of selecting criteria for PLC Programming language

Notes to the trainer:

- The small groups may be used to describe criteria for selecting PLC programming languages.

- The images, videos, and illustrations may also be used as didactic materials.

 **Key steps:**

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and Invite the trainees to answer the following questions in the small groups:
Describe the following selecting criteria for PLC Programming language

i. Language features
ii. Compatibility and interoperability
iii. Scalability
iv. Learning curve
v. Security
vi. Advantages and disadvantages

Step 2: Ask the trainees to presents findings from their small groups
Step 3: Refers to the trainee's answers, provide and clarify the expert view.
Step 4: Direct trainees to read the key reading 2.2.2 in their manuals for more details on PLC programming languages**.**

 **Points to Remember**

- **Key Considerations for Language Features:**

  ✓ Ease of Logic Representation:
  ✓ Complexity Handling:
  ✓ Libraries and Function Blocks
  ✓ Visual vs. Text-Based

- Compatibility refers to how well different programming languages, software, hardware, and PLC components can work together without causing conflicts or requiring significant modifications
- Interoperability refers to the ability of systems, components, or devices to work together seamlessly, even if they are using different programming languages or technologies

- Scalability is essential when designing systems that may expand over time or require changes during operation.
- The learning curve in PLC programming refers to the progression of skill and knowledge development as a person becomes proficient in writing, debugging, and understanding PLC programs
- Key Considerations for Security in **PLC programming** language?
    - ❖ Code Integrity
    - ❖ Access Control
    - ❖ Validation and Verification
    - ❖ Error Handling and Robustness

Each programming language has its own strengths and weaknesses, making them more or less suitable depending on the application.

Application of learning 2.2.

As an automation technician, you are tasked with selecting appropriate PLC programming languages for XYZ Manufacturing's automated packaging line. The packaging system involves several operations, including barcode scanning, speed control using VFDs, and robotic arm coordination. You need to ensure that each function is handled with the most efficient programming approach for optimal system performance.

Checklist:

SN	Criteria	Indicators	Yes	No
1	PLC programming languages is properly selected	1.1 Application of PLC programming languages is considered		
		1.2 Characteristics of PLC programming languages is considered		
		1.3 Language features of PLC programming languages is considered		
		1.4 Compatibility and interoperability of PLC programming languages is considered		
		1.5 Scalability of PLC programming languages is considered		
		1.6. Security of PLC programming languages is considered		

Indicative content 2.3: Writing PLC Program

Duration : 15hrs

Practical Activity 2.3.1: Using Ladder Diagram (LAD) programming language

Notes to the trainer

- The trainer may avail tools, materials and equipment used in automation system.
- Avail computer Lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduces the topic for the day and ask trainees to read the following task individually: You are asked to go in computer lab and write PLC program by using Ladder Diagram (LAD) programming language

Step 2: Explain the task and provide clear instructions to be followed

Step 3: Demonstrate how to perform the task by explaining the steps to be followed

Step 4: Ask trainees to perform the given task individual and monitor them

Step 5: Assess their performance and provide feedback.

Step 6: Allow trainees to ask questions or clarifications

Step 7: Lead them to read key reading 2.3.1 in their manuals

Points to Remember

- Ladder Diagram (LAD) is a graphical programming language widely used in Programmable Logic Controllers (PLCs) for industrial automation
- Developing a logic ladder diagram involves a structured approach to visualize logical operations and control processes

- Translating logic into rungs involves converting a control logic diagram (like a flowchart or a logical description) into a Ladder Diagram (LAD) format. Each rung of a ladder diagram represents a logical operation or sequence, typically corresponding to relay logic
- Testing a program is conducted to verify a software system's functionality, performance, and reliability to identify defects or errors
- Debugging a program is investigating and resolving those defects, aiming to eliminate issues and ensure smooth operation. As you see at the bottom, all errors are corrected well
- Documentation: It is crucial to document the debugging process, including the observed symptoms, root cause, applied fixes, and testing outcomes. Documentation is a reference for future debugging efforts, aids knowledge sharing, and helps maintain a record of resolved issues.
- Maintenance refers to the process of ensuring that the TIA Portal project remains functional and up to date throughout its lifecycle

Practical Activity 2.3.2: Using Function Block Diagram (FBD) programming language

Notes to the trainer

- The trainer may avail tools, materials and equipment used in automation system.
- Avail computer Lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduces the topic for the day and ask trainees to read the following task individually:
You are asked to go in computer lab and write PLC program by usingFunction Block Diagram (FBD) programming language

Step 2: Explain the task and provide clear instructions to be followed

Step 3: Demonstrate how to perform the task by explaining the steps to be followed

Step 4: Ask trainees to perform the given task individual and monitor them

Step 5: Assess their performance and provide feedback.

Step 6: Allow trainees to ask questions or clarifications

Step 7: Lead them to read key reading 2.3.2 in their manuals

Points to Remember

- System requirements on Function Block Diagram (FBD)
  - ✓ Understand the Process or System
  - ✓ Identify Inputs and Outputs
  - ✓ Determine Control Logic
  - ✓ Set Timing and Safety Requirements
- Logical Blocks Example: Use AND, OR, and NOT blocks to handle control conditions (e.g. Start AND Temperature OK).
- Program Testing and Debugging
  - ✓ Simulate the Program
  - ✓ Step-by-Step Testing
  - ✓ Monitor Variables
  - ✓ Fix Issues and Debug
  - ✓ Deploy the Program

Practical Activity 2.3.2: Using Structured Control Language

Notes to the trainer

- The trainer may avail tools, materials and equipment used in automation system.
- Avail computer Lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduces the topic for the day and ask trainees to read the following task individually:

You are asked to go in computer lab and write PLC program by using Structured Control Language (SCL)

Step 2: Explain the task and provide clear instructions to be followed

Step 3: Demonstrate how to perform the task by explaining the steps to be followed

Step 4: Ask trainees to perform the given task individual and monitor them

Step 5: Assess their performance and provide feedback.

Step 6: Allow trainees to ask questions or clarifications

Step 7: Lead them to read key reading 2.3.3 in their manuals

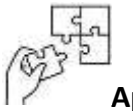
Points to Remember

- The system requirements of Structured Control Language (SCL)
  - ✓ Understanding the Application
  - ✓ Identifying Inputs and Outputs
  - ✓ Operational Logic
  - ✓ Performance Metrics
- Program Structure in SCL

An SCL program typically consists of the following main components

1. Header Section
2. Variable Declaration Section
3. Function and Function Block Declarations
4. Main Program Body
5. End of Program

- Coding logic with control statements in Structured Control Language (SCL) is essential for implementing decision-making processes and repetitive tasks in automation systems.
- Interfacing with Input/output (I/O) devices in Structured Control Language (SCL) is essential for enabling communication between the control program and the physical world.
- Testing and debugging are critical stages in the development of programs in Structured Control Language (SCL) to ensure that the code functions as intended and meets the specified requirements. Here's an overview of the testing and debugging processes in SCL.
- Debugging in PLC simulation software is the process of identifying, isolating, and resolving errors or malfunctions in a simulated PLC program or configuration.



Application of learning 2.3.

A storage tank requires an automated control system to maintain water levels efficiently. The system should activate the water pump when the water level falls below a certain threshold, as indicated by a low-level sensor, and keep it running until the water reaches the high-level threshold. Once the high level is reached, the pump should continue to operate for an additional 15 seconds to stabilize the level before shutting off. As automation technician Your task is to develop a ladder logic program for this system. The program should ensure that the pump starts when the low-level sensor detects a low water level and continues to run until the high-level sensor detects a high water level, at which point a 15-second delay is triggered before the pump stops. Please verify that the sensors and timer function as intended, allowing for consistent water level control within the tank.

Checklist:

SN	Criteria	Indicators	Yes	No
1	System Analysis is properly initiated	1.1 System requirements and control tasks are clearly defined (pump activation, high/low-level control, timing delay).		
		1.2 Input/output wiring of low-level sensor (I0.0), high-level sensor (I0.1), and pump (Q0.0**)** are confirmed.		
		1.3 Timer (T0) settings are reviewed for accuracy (15-second delay).		
		1.4 Ladder logic flow is reviewed to ensure proper order of operations.		
2	Ladder Logic Programming is correctly performed	2.1 Programmed low-level sensor to activate the pump when the water level is below the low threshold.		
		2.3 High-level sensor (I0.1) triggers the timer for pump delay shutdown.		

		2.4 Timer (T0) is set for 15 seconds to delay pump shutdown after high level is reached.		
		2.5 Ladder logic is checked to ensure the pump shuts off after the timer completes.		
3	System is properly tested	3.1 Tested low-level sensor activation to confirm it starts the pump.		
		3.2 High-level sensor tested to confirm it initiates the 15-second delay.		
		3.3 Timer delay verified to confirm 15-second functionality before stopping the pump.		
		3.4 Full system test completed, verifying pump starts, fills, pauses, and stops as programmed.		
4	Safety and Reset Functionality are well done	4.1 Confirmed system reset functionality if sensors or timer need adjustments.		
		4.2 Verified standby mode is resumed if water level drops below low threshold after pump stops.		
5	Documentation and Review are well done	5.1 Documented all sensor and output configurations for reference.		
		5.2 Test results logged and adjustments noted as required.		
		5.3 Verified that system meets design specifications and operates consistently before deployment.		

Indicative content 2.4: Converting PLC Programming Language

Duration: 8 hrs

Practical Activity 2.4.1: Converting ladder diagram to structure control language (LAD to SCL)

Notes to the trainer

- The trainer may use individuals to perform ladder diagram to structure control language (LAD to SCL)
- Avail PLC software (TIA portal) and PLC hardware is required.
- Avail Computer lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask learners read the task described below individually:
As automation technician in our warehouse, you have been tasked with upgrading the conveyor belt sorting system, which currently relies on a Ladder Diagram (LAD) program to sort packages by size. The existing LAD program controls sensors, motors, and actuators but lacks the flexibility and data tracking needed to meet increasing demands. Management has assigned you to convert this LAD program into Structured Control Language (SCL) to unlock more advanced functionality. Your task involves translating LAD rungs—including start/stop functions, package size detection, diversion mechanisms, and package counting—into structured SCL code.

Step 1: Explain the task and provide clear instructions to be followed

Step 2: Demonstrate how to perform the task by explaining the steps to be followed

Step 3: Ask trainees to perform the given task individual and monitor them

Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

Step 6: Lead them to read key reading 2.4.1 in their manuals

Points to Remember

- The steps for converting a Ladder Diagram (LAD) program into Structured Control Language (SCL) for PLC programming:
  1. Understand and Analyze the Ladder Diagram (LAD)
  2. Define SCL Variables
  3. Translate Contacts to Boolean Expressions
  4. Convert Series and Parallel Logic
  5. Translate Coils as Output Assignments
  6. Translate Timers and Counters
  7. Combine and Test the Code
  8. Document the Code

Practical Activity 2.4.2: Converting Ladder Diagram to Function Block Diagram (LAD to FBD)

Notes to the trainer

- The trainer may use individuals to perform ladder diagram to function block diagram (LAD to FBD) programming language.
- Avail PLC software (TIA portal) and PLC hardware is required.
- Avail Computer lab.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduces the topic for the day and ask trainees to read the following task individually:

You are a technician tasked with upgrading a motor control system to improve its readability and modularity. Currently, the system operates with a Ladder Diagram (LAD) where pressing a "Start" button turns the motor on, and pressing a "Stop" button turns it off. To make this system easier to troubleshoot, you need to convert the control logic to a Function Block

Diagram (FBD). Your task is to design the FBD by selecting appropriate function blocks (such as Set/Reset latches or logic gates) to replicate the start/stop functionality of the motor.

Step 2: Explain the task and provide clear instructions to be followed

Step 3: Demonstrate how to perform the task by explaining the steps to be followed

Step 4: Ask trainees to perform the given task individual and monitor them

Step 5: Assess their performance and provide feedback.

Step 6: Allow trainees to ask questions or clarifications

Step 7: Lead them to read key reading 2.4.2 in their manuals

Points to Remember

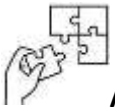
- The steps for Converting Ladder Diagram (LAD) to Function Block Diagram (FBD)

Step 1: Understand the Ladder Logic

Step 2: Create a New Function Block Diagram (FBD)

Step 3: Translate Ladder Logic into FBD

Step 4: Assign Inputs and Outputs



Application of learning 2.4.

As an industrial automation technician, you are tasked with upgrading an existing motor control system for a production line. The current control logic uses a Ladder Diagram (LAD) and allows operators to start and stop the motor with two buttons: A Start button to initiate motor operation and a Stop button to halt it. To modernize this system, you need to convert the Ladder Diagram into both Structured Control Language (SCL) and Function Block Diagram (FBD) formats.

Checklist:

SN	Criteria	Indicators	Yes	No
1	System Analysis is properly initiated	1.1 System requirements and control tasks are clearly defined (motor start/stop operation).		
		1.2 Inputs and outputs identified (Start button, Stop button, Motor Running indicator)		
		1.3 LAD logic reviewed for series and parallel connection configuration.		

		1.4 Motor start/stop sequence verified in the original LAD logic.		
2	Structured Control Language (SCL) Programming) is correctly Designed	2.1 Defined Boolean variables for Start button, Stop button, and Motor Running state.		
		2.2 SCL code created to set Motor Running state based on Start and Stop button conditions.		
		2.3 IF statements verified to match LAD control flow.		
		2.4 Motor start and stop actions tested successfully in SCL simulation.		
3	Function Block Diagram (FBD) is correctly Designed	3.1 AND block set up for series conditions, OR block for parallel conditions.		
		3.2 SR latch created for motor state control (Set/Reset based on Start and Stop buttons).		
		3.3 All blocks connected properly (inputs to SR latch, output to Motor Running).		
		3.4 FBD logic matches LAD sequence and operates correctly in simulation.		
4	Simulation and Testing are well performed	4.1 Start button activates motor in both SCL and FBD implementations.		
		4.2 Stop button halts motor operation in both SCL and FBD implementations.		
		4.3 Motor maintains state until Stop button is pressed, confirming latch behavior.		
		4.4 No discrepancies found between SCL and FBD outputs during testing.		
5	Documentation is well done	5.1 Initial LAD setup and modifications are documented for future reference.		
		5.2 SCL and FBD logic fully documented with comments and descriptions.		

		5.3 Troubleshooting guide prepared for operators based on new logic formats.		
6	Finalization is well done	3.1 Both SCL and FBD meet operational requirements after testing.		
		6.2 System documentation and logic diagrams updated.		
		6.3 Operators and maintenance staff trained on upgraded system.		

Learning outcome 2 end assessment

Theoretical assessment

A. Open questions

Read carefully the following statements and choose by encircling the letter corresponding to the right answer

1. What does a normally open (NO) contact in a ladder logic program represent?

   Answer: A normally open (NO) contact represents a switch that is open in its default state and closes when activated.

2. Explain the function of a normally closed (NC) contact in a ladder logic program.
   Answer: A normally closed (NC) contact represents a switch that is closed in its default state and opens when activated.
3. What is the purpose of a relay coil in a ladder logic program?
   Answer: A relay coil is used to control an output device by energizing or deenergizing it based on the rung conditions.
4. Define an On-Delay Timer (TON) in a ladder logic program.
   Answer: An On-Delay Timer (TON) is a timer that starts counting when its input condition is true, and its output becomes true after the preset time elapses.
5. What is a Counter (CTU) in a ladder logic program?
   Answer: A Counter (CTU) is used to count the number of occurrences of an input event and increment its count value.

B. Multiple Choice Questions:

Read the following statement and answer by circling a letter corresponding to the correct answer

1. In Ladder Logic, what does a normally open (NO) contact represent?
   a. A closed switch when active
   b. An open switch when inactive
   c. A relay coil
   d. A condition that is always true

   Answer: b) An open switch when inactive

2. Which type of PLC memory is used to store the user program?
   a. RAM
   b. ROM
   c. EPROM
   d. Flash Memory

Answer: d) Flash Memory

3. What is the purpose of a timer in a PLC program?
    a. To count the number of events
    b. To measure the time between events
    c. To delay actions in the program
    d. To control the program flow

Answer: c) To delay actions in the program

4. Which software is commonly used for programming Siemens PLCs?
    a. TIA Portal
    b. Visual Studio
    c. MATLAB
    d. AutoCAD

Answer: **a) TIA Portal**

5. Which of the following is a typical output device in a PLC system?
    a. Sensor
    b. Push Button
    c. Actuator
    d. Thermocouple

Answer: c) Actuator

6. Which software is used for PLC simulation in this context?

    a. AutoCAD
    b. TIA Portal
    c. MATLAB
    d. Proteus
    Answer: b) TIA Portal

7. **Before installing PLC software, what check should be performed?**

    a. Network connection check
    b. System compatibility check
    c. Calibration check
    d. Firmware update check
    Answer: b) System compatibility check

8. **Which programming language represents system logic with rungs?**

a. Function Block Diagram (FBD)
b. Structured Control Language (SCL)
c. Ladder Diagram (LAD)
d. Sequential Function Chart (SFC)
Answer: c) Ladder Diagram (LAD)

9. **In PLC programming, which language uses function blocks to construct logic?**
    a. SCL
    b. FBD
    c. LAD
    d. STL
    Answer: b) FBD
10. **Which criterion evaluates how easily a programming language can be learned by new users?**
    a. Scalability
    b. Interoperability
    c. Security
    d. Learning curve
    Answer: d) Learning curve
11. **Which of the following statements is true about SCL?**
    a. It uses graphical symbols for programming.
    b. It is based on structured control statements.
    c. It is primarily used for creating ladder diagrams.
    d. It lacks debugging capabilities.
    Answer: b) It is based on structured control statements.
12. **What is the primary goal of testing and debugging a PLC program?**
    a. To ensure system compatibility
    b. To improve scalability
    c. To detect and correct program errors
    d. To update system documentation
    Answer: c) To detect and correct program errors
13. **In Ladder Diagram programming, what is used to represent the process logic?**
    a. Function blocks
    b. Code structures
    c. Rungs
    d. Statements
    Answer: c) Rungs

14. **Which PLC programming language requires defining both system requirements and block parameters for configuring program logic?**
    a. Ladder Diagram (LAD)
    b. Structured Control Language (SCL)
    c. Function Block Diagram (FBD)
    d. Sequential Function Chart (SFC)
    Answer: c) Function Block Diagram (FBD)
15. **When converting from Ladder Diagram to Structured Control Language, the logic is translated into _____.**
    a. rungs
    b. control statements
    c. functions
    d. block diagrams
    Answer: b) control statements

C. For the following questions, Fill-in-the-Blank

Read the following statement and complete them by using correct terms: **Ladder Diagram (LAD), Function Block Diagram (FBD), scalability, parameters, system compatibility, control, TIA Portal, control, rungs, function blocks, structured, Debugging,**

1. The main software used to simulate PLC programs in this setup is called _____.
   Answer: **TIA Portal**
2. To ensure compatibility, it is necessary to perform a _____ check-up before installing PLC simulation software.
   Answer: **system compatibility**
3. The two main PLC programming languages used to define system requirements and develop logic are _____ and _____.
   Answer: Ladder Diagram (LAD), Function Block Diagram (FBD)
4. In Ladder Diagram programming, system logic is represented in _____ that simulate electrical relay logic.
   Answer: **rungs**
5. The SCL programming language is known for using _____ statements to control logic and flow.
   Answer: **control**
6. One advantage of Function Block Diagram (FBD) is its use of _____ for building logic, making it suitable for process-oriented applications.
   Answer: **function blocks**

7. One of the selection criteria for a PLC programming language is _____, which refers to the language's adaptability to expanding applications.
   Answer: **scalability**
8. _____ is a method of testing a PLC program to find and fix any errors before deploying it in a live system.
   Answer: Debugging
9. Converting a Ladder Diagram to Structured Control Language (SCL) requires translating rungs into _____ logic statements.
   Answer: **structured**
10. In FBD, configuring block _____ allows the program to be tailored to specific input and output requirements.
    Answer: **parameters**

Practical assessment

As an automation technician, you are tasked with setting up PLC simulation software and program a new production line at a manufacturing plant. You must install and configure TIA Portal software to simulate the system, which will control conveyor belts, robotic arms, and sorting sensors. First, perform a system compatibility check to ensure the software runs smoothly on the available computer system. Once installed, run and configure the software packages according to the plant's specifications. After setting up the simulation environment, choose appropriate PLC programming languages to define the control logic for each part of the production line, taking into account the complexity of each task.

Checklist:

SN	Criteria	Indicators	Yes	No
1	System Compatibility is properly Checked	1.1 Verify that the computer meets the minimum hardware requirements for TIA Portal.		
		1.2 Check that the operating system is compatible with the TIA Portal software version.		
		1.3 Ensure necessary software dependencies (e.g., .NET Framework) are installed.		
		1.4 Confirm that there is sufficient disk space for installation and operation.		

2	TIA Portal is properly Installed	2.1 Download the latest version of TIA Portal from the official source.		
		2.2 Run the installation wizard and follow the prompts to complete the installation.		
		2.3 Check for any installation errors and troubleshoot if necessary.		
		2.4 Ensure TIA Portal is properly registered and activated.		
3	Configuration of Software is properly done	3.1 Open TIA Portal and create a new project for the production line simulation.		
		3.2 Configure project settings (e.g., PLC type, communication settings).		
		3.3 Import or create required hardware configurations for conveyor belts, robotic arms, and sorting sensors.		
		3.4 Verify that the project environment is set up correctly (e.g., I/O mapping, address allocation).		
4	Selection of Programming Languages is properly done	4.1 Review system requirements and task complexity for each component in the production line.		
		4.2 Select appropriate PLC programming languages for control logic (e.g., LAD, FBD).		
		4.3 Justify language selection based on system requirements and team expertise.		
5	Define Control Logic is properly done	5.1 Create flowcharts for each part of the production line to outline control logic.		
		5.2 Translate flowcharts into the selected programming languages (LAD, FBD) within TIA Portal.		
		5.3 Ensure all necessary I/O devices are integrated into the control logic.		
6	Testing the Simulation is properly done	6.1 Run the simulation to check for errors in the programming logic.		
		6.2 Validate that the conveyor belts, robotic arms, and sorting sensors operate as intended under various scenarios.		
		6.3 Document any issues encountered during simulation and make necessary adjustments to the programming logic.		

7	Documentation and Training is properly done	7.1 Document the setup process, programming logic, and any special configurations made during the installation.		
		7.2 Create a user guide for operators explaining how to use the new system.		
		7.3 Provide training sessions for staff on how to navigate and utilize TIA Portal effectively.		

Further information to the trainer

Books

Dzinic, J., & Yao, C. (2014). Simulation-based verification of PLC programs.

Frey, G., & Litz, L. (2000, October). Formal methods in PLC programming. In Smc 2000 conference proceedings. 2000 ieee international conference on systems, man and cybernetics.'cybernetics evolving to systems, humans, organizations, and their complex interactions'(cat. no. 0 (Vol. 4, pp. 2431-2436). IEEE.

Park, C. M., Bajimaya, S. M., Park, S. C., Wang, G. N., Kwak, J. G., Han, K. H., & Chang, M. (2006, November). Development of virtual simulator for visual validation of PLC program. In *2006 International Conference on Computational Inteligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06)* (pp. 32-32). IEEE.

Park, S. C., Park, C. M., & Wang, G. N. (2008). A PLC programming environment based on a virtual plant. *The international journal of advanced manufacturing technology*, *39*, 1262-1270.

Web links

https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbU9qYmdrMVVnUnFMZXlNMjJ0ZVo1MDJxdTM5UXxBQ3Jtc0tucXJDOGhyTFRCXzVJdGJwNW00SHJGUi1OYUVkaW53X0Zla19Hbi0wM25LMWM2Tkx0b2ZlRTBfM1ZvX0lLa29kRy1CeHlZRUliQk5faVVRd2xmNlpTTlpyVUFsM1VWTER1R3U0akRIRmpxeDd3S3h1VQ&q=https%3A%2F%2Fdrive.google.com%2Ffile%2Fd%2F16F65N2xPyqaCC5y1p53q6nwcviejRL-0%2Fview%3Fusp%3Dsharing&v=XhpJBh5ctpk

(https://www.amazon.com/PLC-Programming-Industrial-Automation-Collins/dp/0071810455)

Indicative contents
3.1 Downloading PLC program
3.2 Testing automation system
3.3 Documenting automation system
3.4 Maintaining automation system

Key Competencies for Learning Outcome 3: Deploy PLC program

Knowledge	Skills	Attitudes
● Description of PLC programming concepts ● Description of testing technics	● Downloading PLC program ● Flash the program file to PLC ● Testing automation system ● Documenting automation system ● Maintaining automation system	● Being patient ● Being Analytical and details oriented ● Being adaptable ● Being Accurate

Duration: 20 hrs

Learning outcome 3 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Download correctly PLC program based on downloading techniques.
2. Test correctly automation system according to intended use.
3. Document properly automation system according to system operation.
4. Maintain properly automation system according to system functionality.

Resources

Equipment	Tools	Materials
● Conveyor belt system with motor and drive mechanism ● PLC modules (e.g., Siemens S7-1200) ● PLC input and output modules ● PLC Power Supply ● Contactors ● Safety gloves (e.g., Mechanix Wear, Ansell) ● Protective clothing (e.g.,	● Pliers ● Wire stripper ● Allen wrench ● PLC programming software (TIA Portal) ● Screw drivers	● Electrical wires (various gauges) ● Relay ● Cable ties ● Sensors ● Labels ● Ethernet cables/USB Cable ● Power cables

coveralls, safety vests) • HMI devices (e.g Siemens wincc) • Multimeter • Personal computer		

Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Ensure all necessary tools and equipment for Deploying PLC Program are available and in working condition.
- Review the safety procedures and protocols to be followed during the PLC program Deployment.
- Set up a clean, organized workspace to learning and practice.

Indicative content 3.1: Downloading PLC Program

Duration: 4 hrs

Theoretical Activity 3.1.1: Introduction to program file preparation

Notes to the trainer:

- The small groups may be used to introduce program file preparation
- The images, videos, and illustrations may also be used as didactic materials.
- Avail of workshop

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and invite the trainees to answer the following questions in the small groups

i. What do you understand by program file preparation?
ii. Describes the steps required in PLC program file preparation.
iii. Differentiate method of connection and depth connection used in PLC

Step 2: Ask the trainees to present findings from their small groups

Step 3: Refers to the trainee's answers, provide and clarify the expert view

Step 4: Direct trainees to read the key reading 3.1.1 in their manuals for more details

Points to Remember

- Program file preparation is a critical stage in working with a PLC (Programmable Logic Controller) system. It involves developing, organizing, and testing the program before it is transferred to the PLC for controlling an industrial process or machinery.
- The steps in PLC program file preparation

✓ Understand the Control Requirements
✓ Selecting the Right Programming Software
✓ Writing the Control Logic

- ✓ Organizing the Program Structure.
- ✓ Configuring I/O Mapping
- ✓ Error Checking and Simulation
- ✓ Communication Settings
- ✓ Testing the Program in Real-Time
- ✓ Finalizing and Saving the Program

- Method of Connection is concerned with the physical and network-based ways to connect the programming device to the PLC, like whether you use USB, Ethernet, or Wi-Fi. Depth of Connection refers to the level of access you have once connected, ranging from simply transferring a program to being able to fully control and monitor the PLC's operations

Practical Activity 3.1.2: Flashing PLC Program

Notes to the trainer

- The trainer may use individuals in PLC Flashing program.
- Avail of PLC hardware and tutorials as didactic materials is required.
- Avail workshop.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask trainees to perform the task described below should be done individually:
You are an automation technician at a beverage packaging plant. The production line is currently controlled by an outdated PLC program that doesn't account for new packaging requirements. Your task is to flash a new program into the PLC that includes updated logic for handling new bottle sizes and improving overall efficiency.

Step 2: Demonstrate and explain each procedure while flashing plc program.

Step 3: Distributes individual PLC kits to the students.

Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

Step 6: Lead them to read key reading 3.2.2 in their manuals

Points to Remember

- **The steps process to flash a program to a PLC:**
  1. Verify Program Integrity
  2. Connect the Programming Device to the PLC
  3. Put the PLC in the Correct Mode
  4. Select the Flash/Download Option
  5. Start the Flashing Process
  6. Verify the Flashing Process
  7. Put the PLC in RUN Mode
  8. Perform Testing and Debugging
  9. Save the Program Configuration

Application of learning 3.1.

As an automation technician at a building management system company, you are tasked with updating the PLC program that controls the HVAC system for a large commercial building. The existing system has been underperforming, leading to increased energy usage and inconsistent temperature control. To address these issues, you begin by conducting an initial assessment, reviewing the current system documentation, and analysing historical performance data to identify patterns and inefficiencies.

Checklist:

SN	Criteria	Indicators	Yes	No
1	Initial Assessment is properly done	1.1 Current HVAC system documentation is reviewed (wiring diagrams, PLC code, and device specs).		
		1.2 Historical performance data is analysed for patterns (e.g., temperature fluctuations, energy usage).		

		1.3 Inefficiencies in temperature control and energy usage are identified.		
		1.4 All sensors and actuators (e.g., temperature sensors, dampers, fans) are confirmed operational.		
		1.5 Communication between PLC and HVAC components (sensors, actuators) is verified.		
2	PLC Program is properly reviewed	2.1 Existing PLC program logic is analysed for inefficiencies or errors.		
		2.2 Control strategies (e.g., PID loops, set points) are checked for optimization potential.		
		2.3 PLC memory and processor capacity are reviewed for system expansion or updates.		
		2.4 Ladder logic or function block programming of the current system is examined for improvements.		
3	Control Logic are correctly updated	3.1 New control sequences (e.g., optimized temperature set points, fan control logic) are designed.		
		3.2 Control logic is implemented for energy-efficient HVAC operation (e.g., variable fan speed, timed modes).		
		3.3 The updated PLC program is simulated and tested offline for functionality.		
		3.4 Changes are documented and shared with relevant stakeholders (maintenance team, management).		
4	System is properly tested	4.1 Updated PLC program is uploaded to the live system.		

		4.2 HVAC system response is tested with new control logic (adjusting temperature, fan speeds).		
		4.3 All sensors and actuators (fans, dampers) respond correctly to new program commands.		
		4.4 HVAC performance under different load conditions (peak/off-peak) is monitored for stability.		
5	Performance Evaluation is properly checked	5.1 Energy consumption is tracked over a set period (post-update comparison to historical data).		
		5.2 Temperature control and comfort levels across the building are analysed for consistency.		
		5.3 Any remaining inefficiencies or deviations from the expected performance are identified.		
		5.4 Final program adjustments are made based on post-update performance evaluation.		

Indicative content 3.2: Testing Automation System

Duration: 6 hrs

Theoretical Activity 3.2.1: Description of automation system testing techniques

Notes to the trainer:

- The small groups may be used to read automation system testing techniques.
- The images, videos, and illustrations may also be used as didactic materials.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and invite the trainees to answer the following questions in the small groups.

i. What are automation system testing techniques?
ii. What are the Advantages and disadvantages automation system testing techniques?
iii. Explain STLC (Software Testing Life Cycle).

Step 2: Ask the trainees to present findings from their small groups

Step 3: Refers to the trainee's answers, provide and clarify the expert view

Step 4: Direct trainees to read the key reading 3.2.1 in their manuals for more details

Points to Remember

- Automation system testing, also known as automated testing, is a software testing technique that uses tools and scripts to automatically perform tests on a system.
- Types of automation system testing techniques may include: Functional testing, Unit Testing, Integration Testing, and Simulation Testing.
- Automation system testing has many advantages, including:
  - ✓ Improved Reliability
  - ✓ Early Bug Detection
  - ✓ Enhanced Safety

- ✓ Cost Efficiency
- ✓ Comprehensive Coverage
- ✓ Documentation and Audit Trail

- However, there are also some disadvantages to automation system testing, including
  - ✓ Initial Investment Costs
  - ✓ Time-Consuming Process
  - ✓ Maintenance Effort
  - ✓ Complexity of Testing
  - ✓ Incomplete Scenario Coverage
  - ✓ Dependence on Tools and Technologies
- The Software Testing Life Cycle (STLC) is a structured process that outlines the various stages of software testing. It encompasses all testing-related activities, from planning and designing to execution and closure, ensuring a systematic approach to validating software quality.

Practical Activity 3.2.2: Applying automation system testing techniques

Notes to the trainer

- The trainer may use small group for applying automation system testing techniques.
- Avail of PLC hardware and tutorials as didactic materials is required.
- Avail workshop.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask learners read the task described below individually:

As an automation engineer at a manufacturing plant, you are tasked with upgrading the PLC system that controls the conveyor system used for transporting materials between different production stages. The current system has been experiencing issues, such as inconsistent conveyor speeds and frequent jams, leading to production delays. To address these problems, you have developed a new PLC program designed to improve the control logic for conveyor speed regulation, material detection, and jam management.

Task:

Apply automation testing techniques to ensure the updated PLC program functions properly, effectively controlling the conveyor system's speed, detecting jams, and maintaining smooth material flow before the system is deployed in the live production environment.

Step 2: Demonstrate and explain each procedure while applying automation system testing techniques.

Step 3: Distributes individual tools kits to the trainees

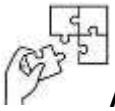
Step 4: Assess their performance and provide feedback.

Step 5: Allow trainees to ask questions or clarifications

Step 6: Lead them to read key reading 3.2.2 in their manuals

Points to Remember

- Unit testing involves testing individual components or modules of the PLC program to ensure they function as intended.
- Steps for applying unit testing includes: Isolate the Unit, Simulate Inputs, Check Outputs, Document Results and Iterate
- System testing evaluates the complete and integrated PLC program as a whole, ensuring that all components work together as intended in the actual operating environment
- Steps for applying system testing includes: prepare the testing environment, execute functional tests, stress testing, monitor performance and document and analyze results



Application of learning 3.2.

As an automation technician at a water treatment plant, you are tasked with upgrading the existing manual system that controls various processes such as filtration, chemical dosing, and water level management. The current system relies on manual operations, which has led to inefficiencies, including inconsistent chemical dosing and frequent overflow issues in the water storage tanks. To resolve these challenges, you are hired to develop a new PLC program that will automate and improve the control logic for each stage of the water treatment process

Task:

Apply automation testing techniques to verify that the updated PLC program works as intended, ensuring the correct chemical dosing, efficient water filtration, and accurate water level management before deploying it to the plant.

Checklist:

SN	Criteria	Indicators	Yes	No
1	Initial Assessment is properly done	Current manual system documentation is reviewed (system processes, equipment specifications).		
		Historical performance data is analysed for inefficiencies (e.g., chemical dosing inconsistencies, overflow frequency).		
		Inefficiencies in chemical dosing, filtration, and water level management are identified.		
		Existing sensors and actuators are confirmed operational (e.g., flow meters, chemical dosing pumps, water level sensors).		
		Communication between PLC and existing plant components (sensors, pumps) is verified.		
2	PLC Program is properly reviewed	Existing manual process logic is reviewed for inefficiencies and manual controls that need automation.		
		Control strategies (e.g., dosing accuracy, water flow management) are analysed for automation potential.		
		PLC memory and processor capacity are reviewed for expansion needs and updates.		
		Control logic for filtration, chemical dosing, and water levels is examined for improvements.		
3	Control Logic is properly updated	New control sequences (e.g., automated chemical dosing, water level management) are designed.		
		Control logic is implemented for optimized filtration, dosing, and water level management.		
		The updated PLC program is simulated and tested offline for functionality.		
		Changes are documented and shared with relevant stakeholders (operations team, maintenance).		

4	Unit Testing is properly performed	Individual control functions (e.g., chemical dosing, water flow management) are tested in isolation.		
		Edge cases for chemical dosing (e.g., minimum and maximum dosing levels) are verified.		
		Water level management logic is tested with simulated overflow conditions to ensure accurate response.		
5	Integration Testing is properly performed	Integrated tests verify that the PLC program correctly interacts with all connected components (sensors, pumps).		
		Communication protocols between PLC and sensors/pumps are validated.		
6	System Testing is properly performed	Updated PLC program is uploaded to the live system for testing.		
		Water treatment system response is tested with new control logic (chemical dosing, filtration).		
		All sensors and actuators (pumps, flow meters, water level sensors) respond correctly to new program commands.		
		Water treatment system performance under different conditions (peak/off-peak) is monitored for stability.		
7	Performance Evaluation is properly performed	Chemical dosing consistency is tracked over a set period (post-update comparison to historical data).		
		Overflow occurrence rate is monitored and compared to historical data for improvement.		
		Final adjustments are made based on post-update performance evaluation.		

Indicative content 3.3: Documenting Automation System

Duration: 4 hrs

Theoretical Activity 3.3.1: Description of technical documentation

Notes to the trainer:

- The small groups may be used to document automation system.
- The images, videos, and illustrations may also be used as didactic materials.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and Invite the trainees to answer the following questions in the small groups.

i. What is Hardware configuration and Software configuration of automation system?
ii. What are the key aspects of hardware and software configuration?
iii. Discuss Control logic of automation system.
iv. Explain the Network architecture of automation system.
v. What are the key components of network architecture?
vi. What is the automation system user manual?

Step 2: Ask the trainees to presents findings from their small groups

Step 3: Refers to the trainee's answers, provide and clarify the expert view

Step 4: Direct trainees to read the key reading 3.3.1 in their manuals for more details

Points to Remember

1. **Hardware configuration** refers to the arrangement and setup of physical components in a PLC system, ensuring that all devices communicate effectively and operate correctly.

Software configuration involves setting up the programming environment, defining parameters, and developing control logic for a Programmable Logic Controller (PLC). This process is essential for ensuring that the PLC operates effectively within an automation system

2. **Key aspects of hardware configuration**

- PLC Selection
- I/O Modules Configuration
- Communication Ports and Protocols
- Power Supply Configuration
- Peripheral Devices and Accessories
- Environmental Considerations
- Documentation and Labelling
- Testing and Validation

Key steps involved in software configuration, using TIA Portal as an example:

- Create a New Project
- Configure the Hardware
- Set Up Network Configuration
- Programming the PLC
- Set Up HMI (if applicable)
- Parameter Configuration
- Testing and Simulation
- Download the Program to the PLC
- Monitoring and Maintenance
- Documentation

3. **Control logic** refers to the set of instructions and rules that dictate how a Programmable Logic Controller (PLC) responds to inputs from various sensors and devices, ultimately determining how outputs (like motors, lights, and other actuators) are activated
4. **Network architecture** in automation systems refers to the structured layout of hardware, software, communication protocols, and data flows that enable devices to communicate and operate together effectively.

Key Components of Network Architecture

- Controllers:
- Network Topologies
- Communication Protocols
- Network Security
- Monitoring and Maintenance

5. Creating an automation system user manual involves providing comprehensive instructions and guidelines to help users operate, maintain, and troubleshoot the system effectively.

Practical Activity 3.3.2: Performing of hardware and software configuration in automation system

Notes to the trainer

- The trainer may use individuals in hardware and software configuration.
- Avail of PLC hardware and tutorials as didactic materials is required.
- Avail workshop.

Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the topic and ask learners read the task described below individually:

As an automation technician in a logistics company, you are tasked with configuring a new automated warehouse management system (WMS) designed to streamline inventory management and improve order fulfilment processes. This system consists of automated guided vehicles (AGVs), RFID scanners, conveyor belts, and a centralized control system that oversees operations. The goal is to ensure that all hardware components are correctly installed and configured, and that the software is programmed to manage the flow of goods efficiently within the warehouse, from receiving shipments to storing and picking items for dispatch.

Step 1: Demonstrate and explain each procedure while applying hardware and software configuration in automation system.

Step 2: Distributes individual tools kits to the students

Step 3: Assess their performance and provide feedback.

Step 4: Allow trainees to ask questions or clarifications

Step 5: Lead them to read key reading 3.3.2 in their manuals

Points to Remember

1. Hardware Configuration

1.1 System Inspection

- Verify the Components**.**
- Check the Wiring Diagram

1.2 Mounting the Equipment

- Mount the Controller (e.g., PLC or DCS)
- Install Sensors and Actuators

1.3 Electrical Connections

- Power Supply
- input/output (I/O) Wiring
- Communication Cables

1.4 Powering Up

- Initial Power-Up
- Check Sensor and Actuator Operation

2. Software Configuration

2.1 Controller (PLC/DCS) Configuration

- Install Programming Software
- Configure Communication Settings

2.2 Creating the Logic Program

- Input/output Mapping
- Program Development
- Timers and Counters

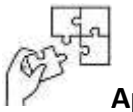
2.3 Testing and Simulation

- **Offline Simulation:**
- **Download Program to Controller:**

2.4 Human-Machine Interface (HMI) Configuration

- **HMI Software Installation:**
- **Create Screen Layouts:**

Link HMI to Controller:



Application of learning 3.3.

As an automation technician at a food processing company, you are tasked with configuring a new automated packaging system designed to streamline the packaging process for various food products. The system consists of multiple components, including conveyor belts, packaging machines, sensors, and a programmable logic controller (PLC) that coordinates the entire operation.

The packaging process must be efficient and adaptable to handle different product sizes and packaging types. You need to ensure that the hardware components are correctly configured and that the software is programmed to operate seamlessly with the hardware. Your goal is to minimize downtime during the setup and ensure that the system meets production requirements.

Checklist

SN	Criteria	Indicators	Yes	No
1	Hardware Configuration	1.1 All hardware components (conveyor belts, packaging machines, sensors) are assembled correctly.		
		1.2 Conveyor belts are aligned and speed settings are configured appropriately.		
		1.3 Packaging machines are set up for different packaging types and sizes.		
		1.4 Sensors are installed for product detection, position monitoring, and quality control.		
		1.5 Electrical connections are made according to wiring diagrams and safety standards.		
		1.6 PLC is properly connected to all hardware components and powered on.		
		1.7 Physical inspection confirms that all components are undamaged and operational.		
2	Software Configuration	2.1 PLC program is developed and uploaded according to the packaging process requirements.		
		2.2 Control logic includes sequences for product detection, packaging, and operational control.		
		2.3 HMI design is intuitive and allows for easy navigation by operators.		
		2.4 Data logging features are implemented for tracking production metrics.		

		2.5 Software configurations are tested in a simulated environment and all functionalities operate correctly.		
		2.6 All configuration settings, program details, and documentation are completed and stored for future reference.		

Indicative content 3.4: Maintaining Automation System

Duration: 6hrs

Practical Activity 3.4.1: Apply preventive maintenance, diagnosing and troubleshooting in automation system

Notes to the trainer

- Facilitate a group discussion, asking students to share their experiences, challenges from the activity.
- Image and didactic may be used if any.

Key steps:

While delivering this activity, pass through the following steps:

Step1. Introduce the topic and ask trainees read the task described below individually:

The beverage manufacturing plant has identified intermittent failures in the automated bottling line, particularly with the bottle detection sensors and the programmable logic controller (PLC) controlling the mixer. As the automation technician, please apply preventive maintenance**,** diagnosing and troubleshooting in automation system by inspecting the bottling line, lubricating moving parts, calibrating sensors, and updating the control system software.

Step2. Explain the task and provide clear instructions to be followed

Step3. Demonstrate how apply preventive maintenance, diagnosing and troubleshooting in automation system.

Step4. Assess their performance and provide feedback.

Step5. Allow learners to ask questions or clarifications

Step6. Lead them to read key reading 3.4.2 in their manuals

Points to Remember

- **Preventive maintenance** is a proactive approach aimed at keeping automation systems in optimal condition by performing regular checks and maintenance tasks
- **System diagnosing** is the process of identifying and analyzing faults or issues within an automation system
- **System troubleshooting** is the process of diagnosing and resolving issues within an automation system to restore its normal operation

Steps for Preventive Maintenance in an Automation System

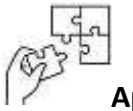
- Develop a Maintenance Schedule
- Conduct Regular Inspections
- Lubricate Moving Parts
- Calibrate Sensors
- Clean Equipment
- Check Electrical Connections
- Update Software and Firmware
- Test System Performance
- Document Maintenance Activities
- Review and Adjust Maintenance Plans

Steps for Diagnosing in an Automation System

- Identify Symptoms:
- Check System Documentation
- Verify Power Supply
- Inspect Physical Components
- Test Inputs and Outputs
- Monitor System Performance
- Check for Alarms and Error Codes
- Isolate the Problem
- Review Control Logic
- Consult Manufacturer Support
- Implement Solutions
- Document Findings

Steps for Troubleshooting in an Automation System

1. Define the Problem
2. Prioritize the Issue
3. Review System Documentation
4. Check for Recent Changes
5. Inspect Physical Components
6. Test Electrical Connections
7. Evaluate Inputs and Outputs
8. Analyze Control Logic
9. Use Diagnostic Tools
10. Isolate and Test Components
11. Implement Solutions
12. Document the Process



Application of learning 3.4.

In a food packaging facility, an automated packaging system that includes conveyors, filling machines, labelling systems, and sensors has been experiencing inconsistent performance, leading to frequent stoppages and occasional mislabelling of products. To address these issues, the automation technician is tasked with applying preventive maintenance, diagnosing the underlying problems, and troubleshooting the system.

Checklist

SN	Criteria	Indicators	Yes	No
1	Preventive Maintenance	1.1 Preventive maintenance schedule is established and followed for all equipment.		
		1.2 Lubrication of moving parts is performed regularly according to manufacturer specifications.		
		1.3 Calibration of sensors is conducted routinely to ensure accuracy.		
		1.4 Regular inspections are performed to identify wear and potential issues.		

		1.5 Documentation of maintenance activities is up to date and includes dates and tasks performed.		
2	Diagnosing	2.1 Performance issues are documented, including specific symptoms and conditions.		
		2.2 System logs are reviewed for error messages and unusual patterns.		
		2.3 Sensor data is analyzed for discrepancies or anomalies affecting performance.		
		2.4 Mechanical components are inspected for signs of failure, such as jammed labels or misalignments.		
		2.5 Control logic and programming are reviewed for errors or misconfigurations.		
3	Troubleshooting	3.1 Input signals from sensors are tested for proper operation.		
		3.2 Output signals to actuators are verified for correct functionality.		
		3.3 Simulation of operational conditions is performed to replicate issues for analysis.		
		3.4 Communication between components is verified for interruptions or faults.		
		3.5 Faulty components are identified and replaced as necessary.		
		3.6 Adjustments to control logic are made based on findings, followed by retesting of the system.		
		3.7 All findings and actions taken during diagnosis and troubleshooting are documented.		

Learning outcome 3 end assessment

Theoretical assessment

I. Open Questions

1. What steps are involved in preparing a program file for downloading to a PLC?
    - **Answer:** The steps include defining the program requirements, writing the program code, and saving the file in a compatible format for the PLC.
2. Explain the significance of selecting the appropriate connection method when downloading a PLC program**.**

Answer: Selecting the correct connection method is crucial for ensuring data integrity, communication speed, and successful transfer of the program to the PLC.

3. What are the different methods of connecting to a PLC, and how do they differ?

Answer: Common methods include USB, Ethernet, RS-232, and wireless connections. They differ in speed, distance limitations, and compatibility with various PLC models.

4. Describe the process of flashing a program file to a PLC. What precautions should be taken?

Answer: The process involves connecting to the PLC, selecting the program file, and initiating the transfer. Precautions include verifying the file compatibility and ensuring the PLC is in the correct mode for downloading.

5. What are the key techniques used for testing automation systems, and how do they contribute to system reliability?

Answer: Key techniques include functional testing, performance testing, and regression testing. They help identify issues early, ensure all components work correctly, and confirm the system meets specified requirements.

6. Discuss the advantages and disadvantages of various testing techniques applied in automation systems.

Answer: Advantages may include thoroughness and early error detection, while disadvantages could involve high costs, and the time required to conduct extensive testing.

7. What is the Software Testing Life Cycle (STLC), and what are its key phases?

Answer: The STLC is a process that outlines the various phases of testing software, including requirement analysis, test planning, test case design, test execution, and test closure.

8. Differentiate between unit testing and system testing in the context of automation systems.

Answer: Unit testing focuses on individual components or modules, whereas system testing evaluates the entire system's compliance with specified requirements.

9. What elements should be included in the technical documentation for an automation system?

Answer: Elements include hardware configuration, software configuration, control logic, network architecture, and user manuals.

10. Explain the role of preventive maintenance in maintaining an automation system.

Answer: Preventive maintenance involves regular inspections and servicing to prevent potential failures, ensuring the system operates efficiently and reducing downtime.

II.Multiple-Choice Questions

1. **What is the first step in program file preparation for a PLC?**

A) Flashing the program

B) Writing the program code

C) Selecting connection method

D) Testing the program

Answer: B) Writing the program code

2. **Which of the following is NOT a method of connection for downloading a PLC program?**

A) USB

B) Ethernet

C) HDMI

D) RS-232

Answer: C) HDMI

3. **What is the primary advantage of using Ethernet for PLC connections?**

A) Higher compatibility

B) Greater distance coverage

C) Simplicity of use

D) Lower cost

Answer: B) Greater distance coverage

4. **Which technique is primarily used for identifying issues in individual modules?**

A) System testing

B) Unit testing

C) Integration testing

D) Performance testing

Answer: B) Unit testing

5. **In the STLC, what phase follows test planning?**

A) Test case design

B) Test execution

C) Requirement analysis

D) Test closure

Answer: A) Test case design

6. **What is a primary disadvantage of exhaustive testing techniques?**

A) They are time-consuming

B) They are cost-effective

C) They guarantee bug-free software

D) They are easy to implement

Answer: A) They are time-consuming

7. **What should technical documentation include regarding hardware?**

A) Programming languages used

B) Hardware configuration

C) User experience

D) Testing results

Answer: **B) Hardware configuration**

8. **What is the purpose of a user manual in an automation system?**

A) To outline the maintenance schedule

B) To provide end-users with operational guidance

C) To describe testing procedures

D) To detail programming techniques

Answer: B) To provide end-users with operational guidance

9. **Which of the following is a method of diagnosing system issues in an automation system?**

A) Preventive maintenance

B) Fault tree analysis

C) User training

D) User manual review

Answer: B) Fault tree analysis

10. **What is the main focus of preventive maintenance?**

A) Fixing failures as they occur

B) Enhancing system performance

C) Preventing failures before they happen

D) Upgrading software regularly

Answer: C) Preventing failures before they happen

Practical assessment

You are tasked with designing and implementing an automation system for a small manufacturing facility that uses a conveyor belt to transport products from one workstation to another. The system will be controlled by a Programmable Logic Controller (PLC) and will include sensors to detect the presence of products on the conveyor.

Check list

SN	Criteria	Indicators	Yes	No
1	System Design	1.1 The design specifications for the automation system are documented and approved.		
		1.2 The layout of the conveyor system is optimized for efficient product flow between workstations.		
		1.3 All necessary components (conveyor belts, PLC, sensors) are selected based on operational requirements.		
		1.4 Electrical and control diagrams are created to outline connections and functionalities.		
	Implementation	2.1 The installation of the conveyor system is completed according to design specifications.		
		2.2 All hardware components (PLC, sensors) are installed correctly and securely.		

2		2.3 Wiring and electrical connections are made according to safety standards and wiring diagrams.		
		2.4 The PLC is programmed with the correct logic for controlling the conveyor and sensors.		
3	Testing and Validation	3.1 Initial testing is conducted to verify that the conveyor system operates as intended.		
		3.2 Sensors are tested to ensure they accurately detect the presence of products.		
		3.3 The PLC program is tested for correct execution of control logic during operation.		
		3.4 System performance is validated under various operational conditions to ensure reliability.		
4	Training and Documentation	4.1 Operators are trained on the use of the new automation system, including safety procedures.		
		4.2 Documentation for the system, including user manuals and maintenance guides, is provided.		
		4.3 A preventive maintenance schedule is established and communicated to the maintenance team.		

END

Further information to the trainer

Books

Darvas, D., Blanco, E., & Molnár, S. V. (2019). PLCverif re-engineered: An open platform for the formal analysis of PLC programs. In *Proceedings of the 17th International Conference on Accelerator and Large Experimental Physics Control Systems, JACoW* (pp. 21-27).

Ghaleb, A., Zhioua, S., & Almulhem, A. (2018). On PLC network security. *International Journal of Critical Infrastructure Protection*, *22*, 62-69.

Mellado, J., & Núñez, F. (2022). Design of an IoT-PLC: A containerized programmable logical controller for the industry 4.0. *Journal of Industrial Information Integration*, *25*, 100250.

Schofield, B., Borrego, J., & Blanco Viñuela, E. (2022). Continuous Integration for PLC-based Control System Development. *JACoW*,478-483.

Web links

https://www.mathworks.com/help/plccoder/ug/deploy-ladder-diagram.html

https://www.electroniclinic.com/plc-timers-and-counters-their-types-and-practical-uses/

https://www.realpars.com/blog/plc-timer

RTB | RWANDA TVET BOARD

October, 2024