SMART FACTORY

# Smart Factory 2 & Industry 4.0

CP4902

www.matrixtsl.com

# Table of Contents

## TABLE OF CONTENTS


**Introduction 2**

**Worksheets 3**

*Worksheet 1 – Understanding Sensors 4*

*Worksheet 2 – Reject Mechanisms 8*

*Worksheet 3 – Understanding the Conveyor 9*

*Worksheet 4 – Sorting Counters 11*

*Worksheet 5 – Driving the Stepper Motor 13*

*Worksheet 6 – Understanding the Plunger 15*

*Worksheet 7 – Delivering Counters 17*

*Worksheet 8 – Robot Arm 19*

*Worksheet 9 – Commissioning The Cell 20*

*Worksheet 10 – Completing the Smart Factory 22*

*Worksheet 11 – Defects & Reset Sequence 24*

*Worksheet 12 – Vision System 26*

*Worksheet 13 – RFID 28*

*Worksheet 14 – Network & Communications 30*

*Worksheet 15 – Data Logging 32*

*Worksheet 16 – Analytics 34*

*Worksheet 17 – IO Link 36*

*Worksheet 18 – Predictive Maintenance 38*

**Student Handbook 40**

**Safety Note 60**

**Version Control 62**

## Introduction

## INTRODUCTION

Welcome to the **Industrial Maintenance for Closed Loop Systems Curriculum**. This program is designed to provide a comprehensive understanding of industrial automation and control within a modern manufacturing environment. Whether you are a student, technician, or engineer, this curriculum equips you with the practical skills to design, program, and troubleshoot an integrated automated system using real industrial components.

### Curriculum

This curriculum is divided into a series of worksheets, each focusing on a specific topic such as PLC programming, HMI design, sensor integration, motion control, and pneumatic operation. The activities progress from fundamental control principles to advanced Industry 4.0 concepts.

Automated smart factories are central to modern industry. Understanding how PLCs, HMIs, sensors, and actuators work together to monitor and control production processes is a valuable and transferable skill.

We encourage you to approach each worksheet with curiosity and attention to detail. The practical exercises will help you apply what you learn, developing both your technical competence and your confidence in managing intelligent, data-driven automation systems. Let’s get started!

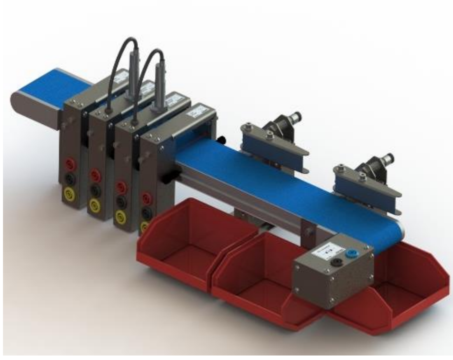
# WORKSHEETS

## Worksheets

## Worksheet 1 – Understanding Sensors

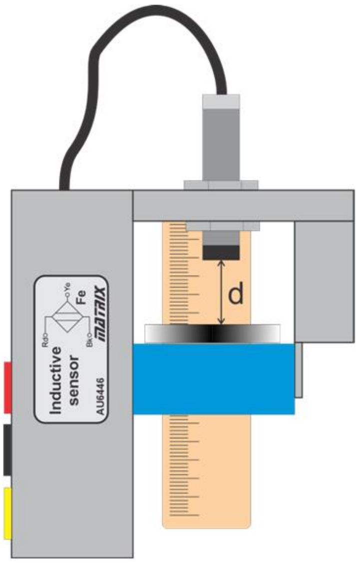
Modern production systems use sensors to detect materials and control automation. Some act as simple switches, while others convert analogue signals to digital data. In this Smart Factory, inductive sensors detect metals, capacitive sensors detect materials, and photoelectric sensors detect passing objects. Understanding each sensor’s behaviour is key to accurate sorting.



### Inductive Sensor

*Over To You*

- Take the inductive sensor and apply power and ground to the red and black connectors respectively.




- Referring to the diagram: using a ruler measure the minimum distance, d, from the sensor at which the inductive sensor triggers for each metal counter: aluminium and steel.
    - On some models of sensor there is an LED indicator.
    - On others you will need to use a multimeter on the output to see when the output voltage changes.

| | Capacitive sensor | Inductive sensor |
|---|---|---|
| Steel counter | | |
| Aluminium counter | | |

- Add the inductive sensor to the conveyor belt using the t-blots to secure the sensor to the sides of the conveyor. Set up the sensor so that the height of the sensor above the conveyor belt is 5.5 mm more than the height you measured for the aluminium counter. This is 1.5mm plus the height of the counter (4mm). This gives some small margin for error.
- Add power to the conveyor manually so that it runs.
- Place the aluminium and stainless steel on the conveyor and make sure that the sensor reliably detects the steel counter but not the aluminium one.

**So what?**

- The height of the conveyor varies slightly as it turns: there is a join in the belt that raises the counter slightly . You may need to increase the distance slightly between the sensor and the counter.
- The inductive sensor allows you to differentiate between different types of metal. So, the inductive sensor needs setting up with some precision.

**Capacitive Sensor**

*Over To You*

- Set up the capacitive sensor so that the height above the conveyor is 2mm more than the trigger height you measured for the steel and aluminium counters. The counter thickness is 4mm so this should give a reliable indication of the presence of a metal counter.
- Add power to the conveyor manually so that it runs.
- Place the aluminium and stainless steel on the conveyor and make sure that the sensor reliably detects the presence of a metal counter beneath it.

*So What?*

You now have two sensors: the inductive sensor allows you to differentiate between steel and aluminium counters. The capacitive sensor allows you to differentiate between plastic and metal sensors. With a little programming logic you should be able to develop a programme that allows you to sort the counters.

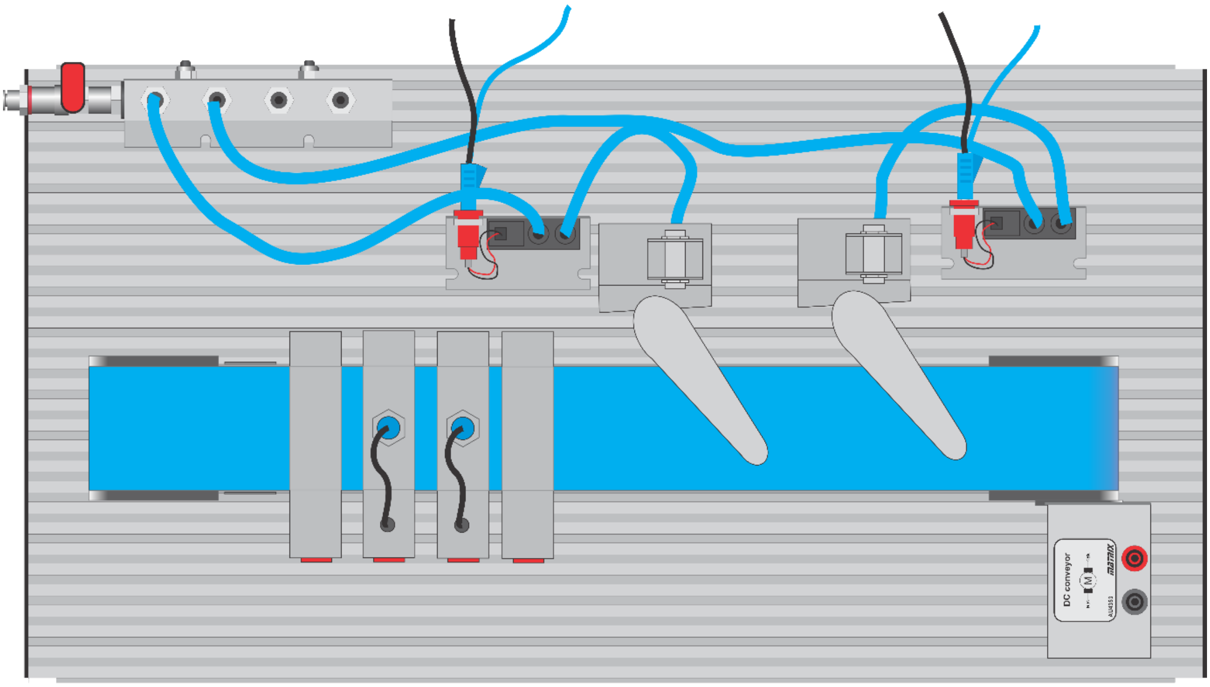
 

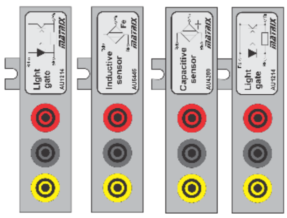
*Over To You*

- Add a light gate to the conveyor using the T bolts.
- Add power to the conveyor manually so that it runs.
- Make sure that the light gate is triggered by a counter

*So What?*

The light gates allow you to detect the presence of a counter reliably. Two light gates allow you to measure the speed of a counter on the conveyor belt. With a little programming you can now predict where a counter is on the conveyor belt once it has been through the sensors. This allows you to effectively programme the rejection mechanisms to direct the counters to the appropriate sorting bins.



## Key Takeaways

Now that you have understood the function of the different types of sensor you can set them up together on the conveyor properly.

Referring to the diagram above set them up as near to the start of the conveyor as you can in the following order:

1. Light gate
2. Inductive sensor
3. Capacitive sensor
4. Light gate

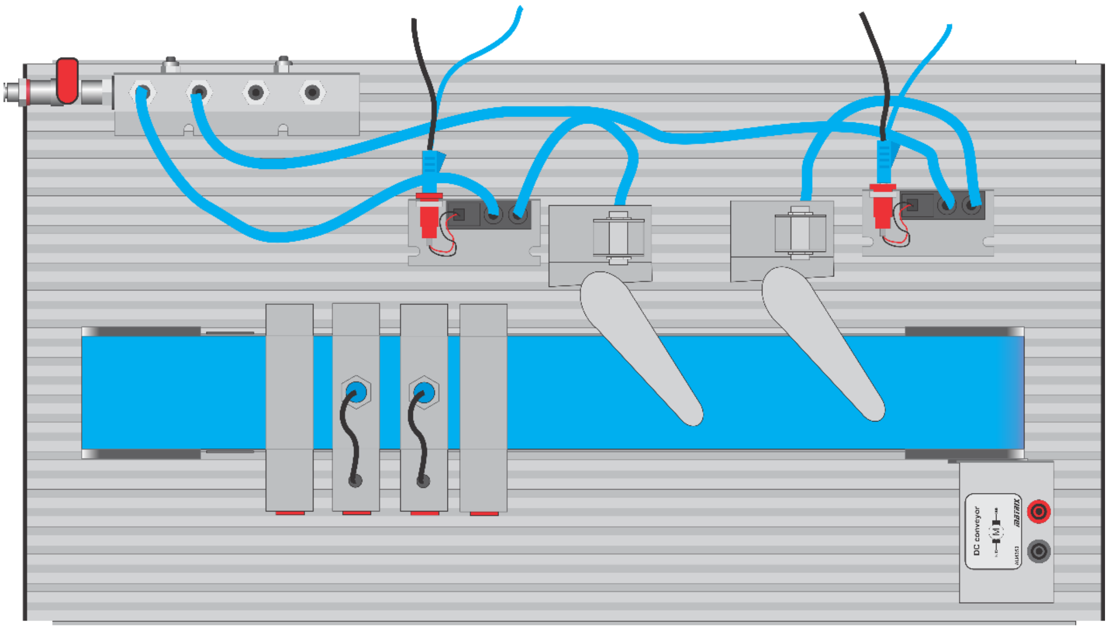
You are now ready to look at the reject mechanisms.

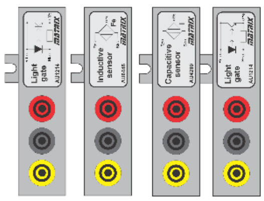
## Worksheets

## Worksheet 2 – Reject Mechanisms

Rotating machines provide much of the power needed in automation. However pneumatics are often used because of their longevity, their power, their speed of operation and their ability to provide linear movement. The rejection components on the conveyor are a good example of this.

Photograph shows a part on a conveyor belt.





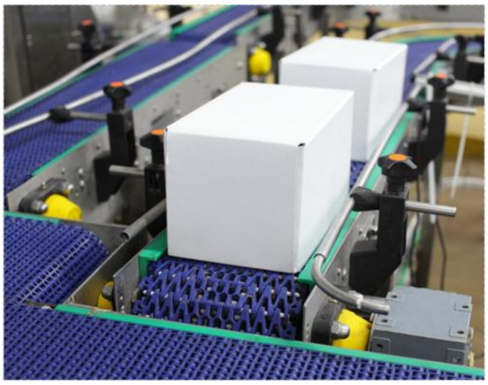
### Over To You

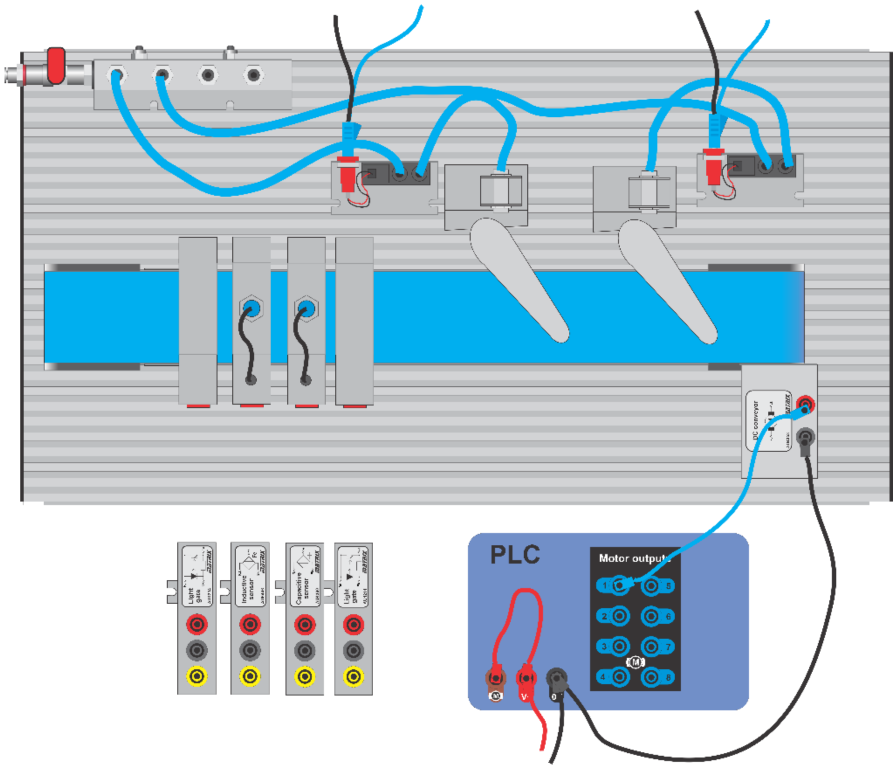
- Add the reject mechanisms, the pneumatic manifold, and the two 3/2 electronic valves to your system.
- Connect the 3/2 valves in turn to the 24V line and make sure that the reject mechanisms are working satisfactorily with the metal and plastic counters.

## Worksheet 3 – Understanding the Conveyor

Conveyor belts are at the heart of many production lines moving items from one location to another allowing mass production of items from pills to drinks.

Photograph shows boxes on a system of conveyor belts that move and merge goods in a factory.






### Over To You

- Wire up the conveyor to your PLC or controller. You will need to use a motor output - these are also sometimes called 'transistor' outputs - for the conveyor.
- On some PLCs the supply line for the transistors is a separate input terminal to the PLC. This allows the transistor outputs to switch a different voltage to the PLC power supply if needed. On the diagram this is labelled as 'M' for motor. You need to put 24V into that terminal - as shown above.
- Write a simple program that switches 24V to the conveyor to make sure the conveyor works.

## Worksheets

- Vary the mark space output of the transistor output between 10% and 100% in 10% steps and make a note of the sped of the conveyor. To do this you will need to use a ruler to measure the position of a counter and a stopwatch. Make a note of your answers in the table.

**So What?**

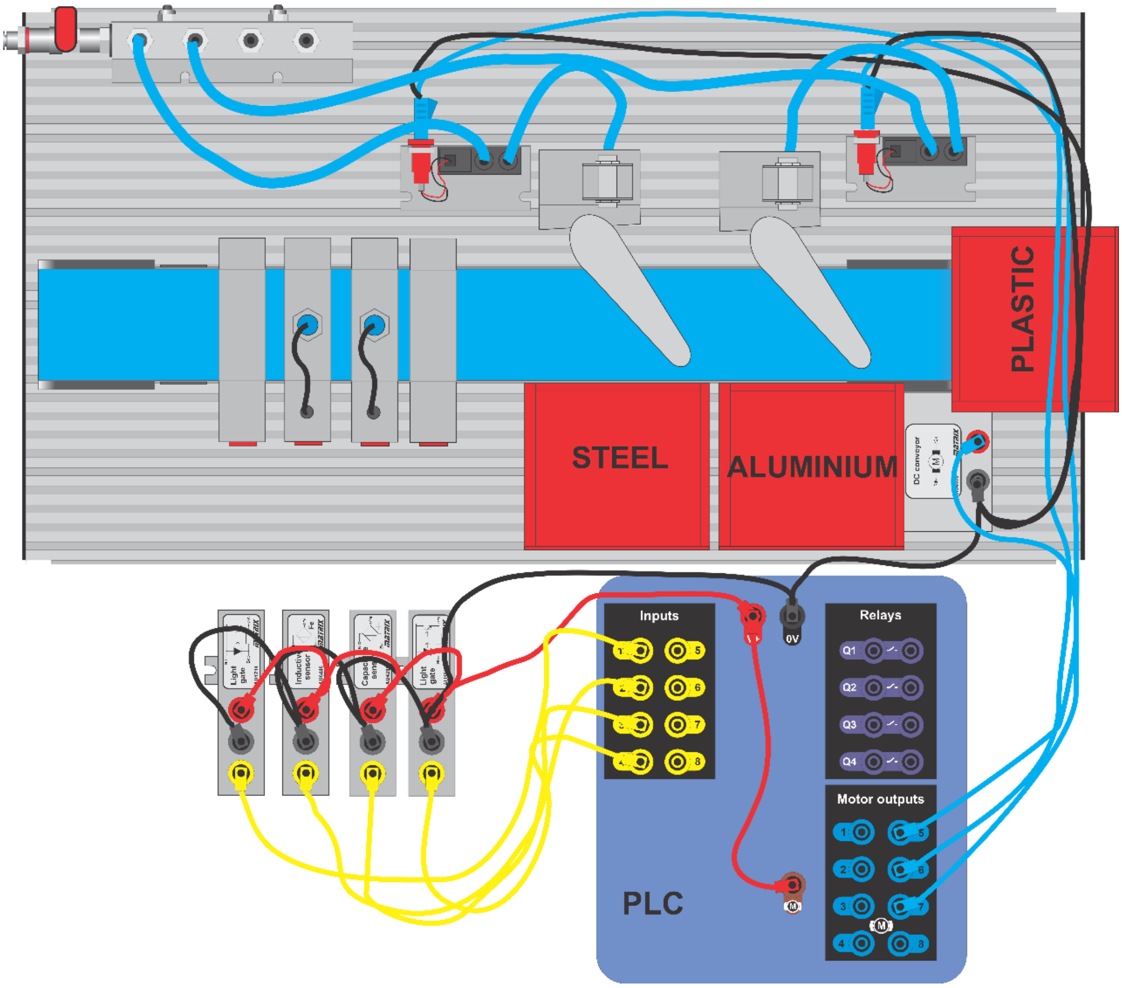
Understanding the speed of the conveyor will be useful later on when you use the robot arm to move counters on and off the conveyor. If you understand the speed of the conveyor then you can accurately predict the position of a counter after it leaves the beam interrupter.

| PWM % | Speed: m/s |
|---|---|
| **10** | |
| **20** | |
| **30** | |
| **40** | |
| **50** | |
| **60** | |
| **70** | |
| **80** | |
| **90** | |
| **100** | |

# Worksheet 4 – Sorting Counters

Sorting components or products is a key part of many automated production lines. The types of component and sensors vary - but the principles are the same: we take in information from sensors and make decisions about what happens to an item on the production line.

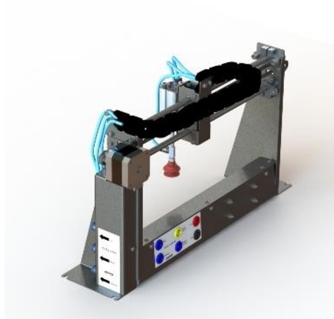
Photograph shows a machine that sorts eggs on their colour.

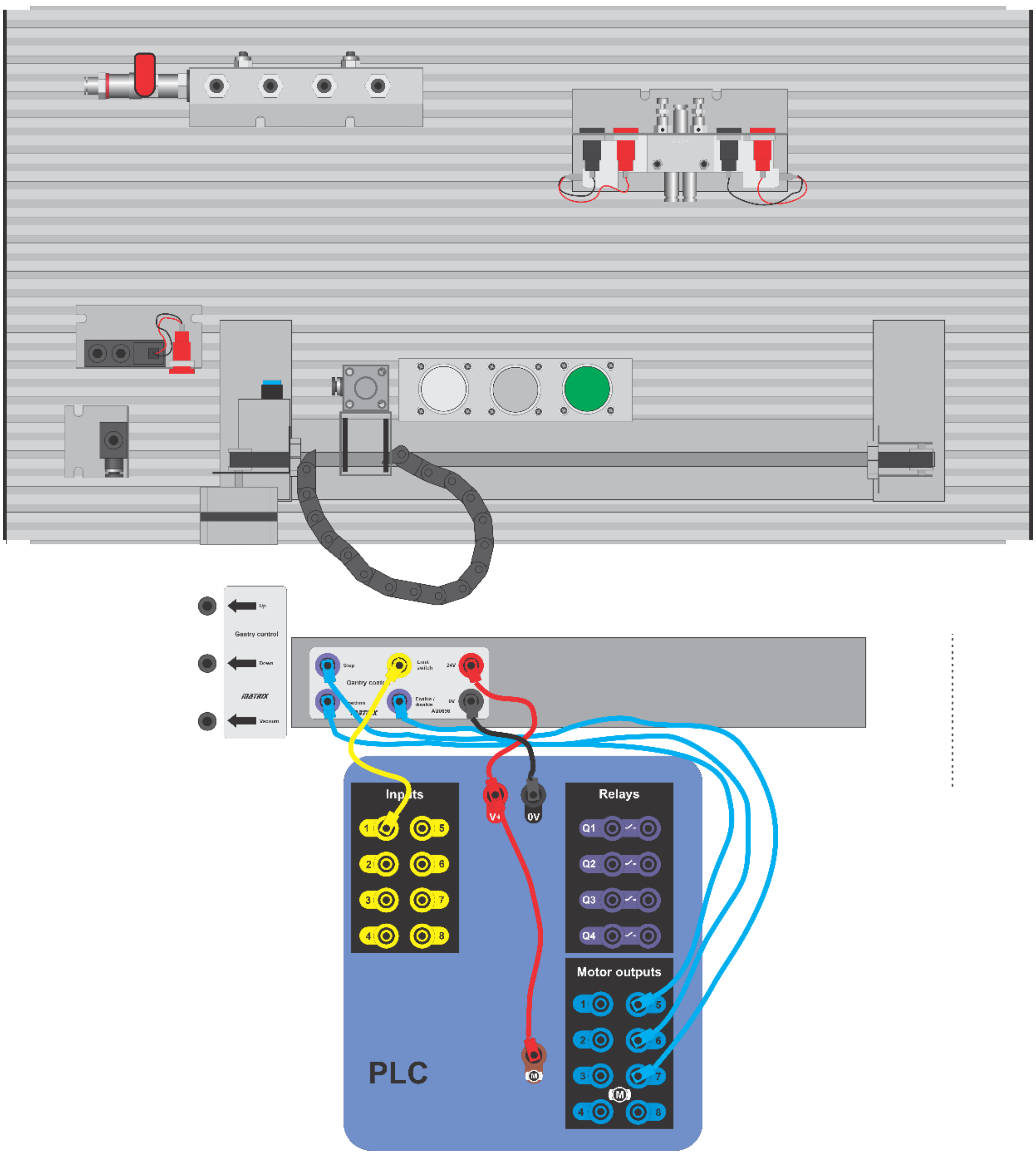


- Build the system you can see in the diagram above.
- The instructions for this task are simple: develop a program on your PLC or controller that can sort the counters into three types: steel, aluminium and plastic.
- The Hall sensor detects the presence of steel. The capacitance sensor detects the presence of a metal.
- The beam interrupters detect the presence of a counter.
- You know the speed of the conveyor and can control it. You can detect when a counter is under each sensor. You know how the rejector mechanisms work. Armed with this knowledge you can develop a program that sorts the counters into the appropriate parts.

## Worksheet 5 – Driving the Stepper Motor

There are many situations in which we need to know the position of a part of a machine in an automated system. The gantry makes use of a limit switch: when the switch is triggered then the mechanism knows that the actuating mechanism has reached a defined position. If movement from that point can be measured then we always know the position of the actuator.





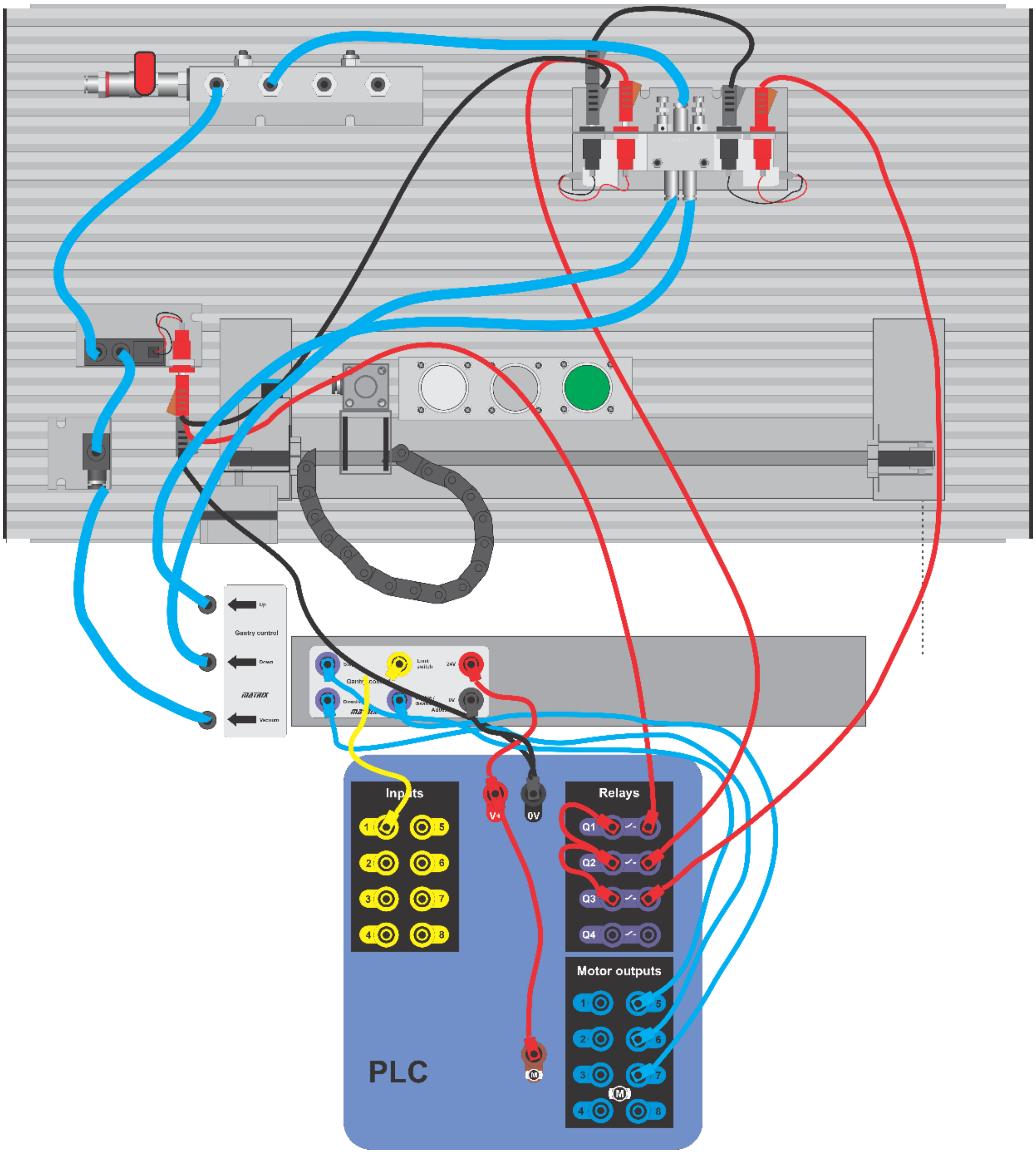
- The stepper motor on the gantry has three inputs: direction, step and enable/disable.
- DIRECTION should be 0V to move towards the home position, +24V to move away from home.
- DISABLE is normally 0V (or disconnected) to enable/power the motor, 24V to disable it. Disable can be left disconnected and the system will function. When DISABLE is high then the current is removed from the stepper motor. This makes it easy to move the gantry by hand. When the motor is enabled there are still holding currents in the windings when it is stopped. This is useful to stop the gantry moving due to pressure from the "pipe-work".
- STEP is normally low (0V). Pulse high and back low to step.
- The 'step size' for this motor is 1.8 degrees. It will require 8 pulses (micro steps) to make a single step.
- The number of teeth on the stepper cog is 20.
- The pitch of the teeth on the drive belt is 2mm. You can now calculate the distance the gantry mechanism moves for each step and micro step (pulse).
- Wire up the gantry mechanism to your PLC as shown above.
- Develop a program with your PLC that moves the stepper motor forwards and backwards.
- Refine this program so that on power up, or the first movement, the stepper motor moves left to the point where the gantry hits the limit switch and internally marks this as '0'.
- Lock the counter holder down on the platform. Create a table that shows the position of each counter storage area.

| | Column 1 | Column 2 | Column 3 |
|---|---|---|---|
| **mm** | | | |
| **Steps from home** | | | |

## Worksheet 6 – Understanding the Plunger

Pneumatic actuators are great tools for picking and placing objects in production lines. Whilst it takes time to get the part setup accurately at the design stage, they will work for millions of cycles in factories before any maintenance is needed.

Photograph shows a suction device in a production cell.

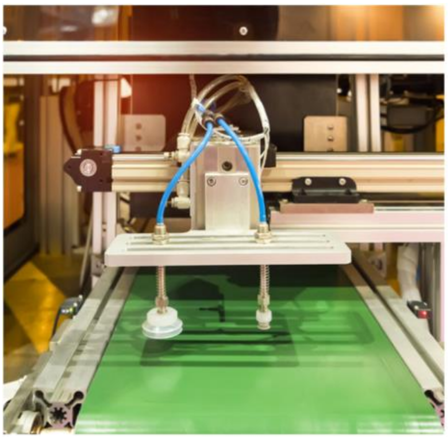


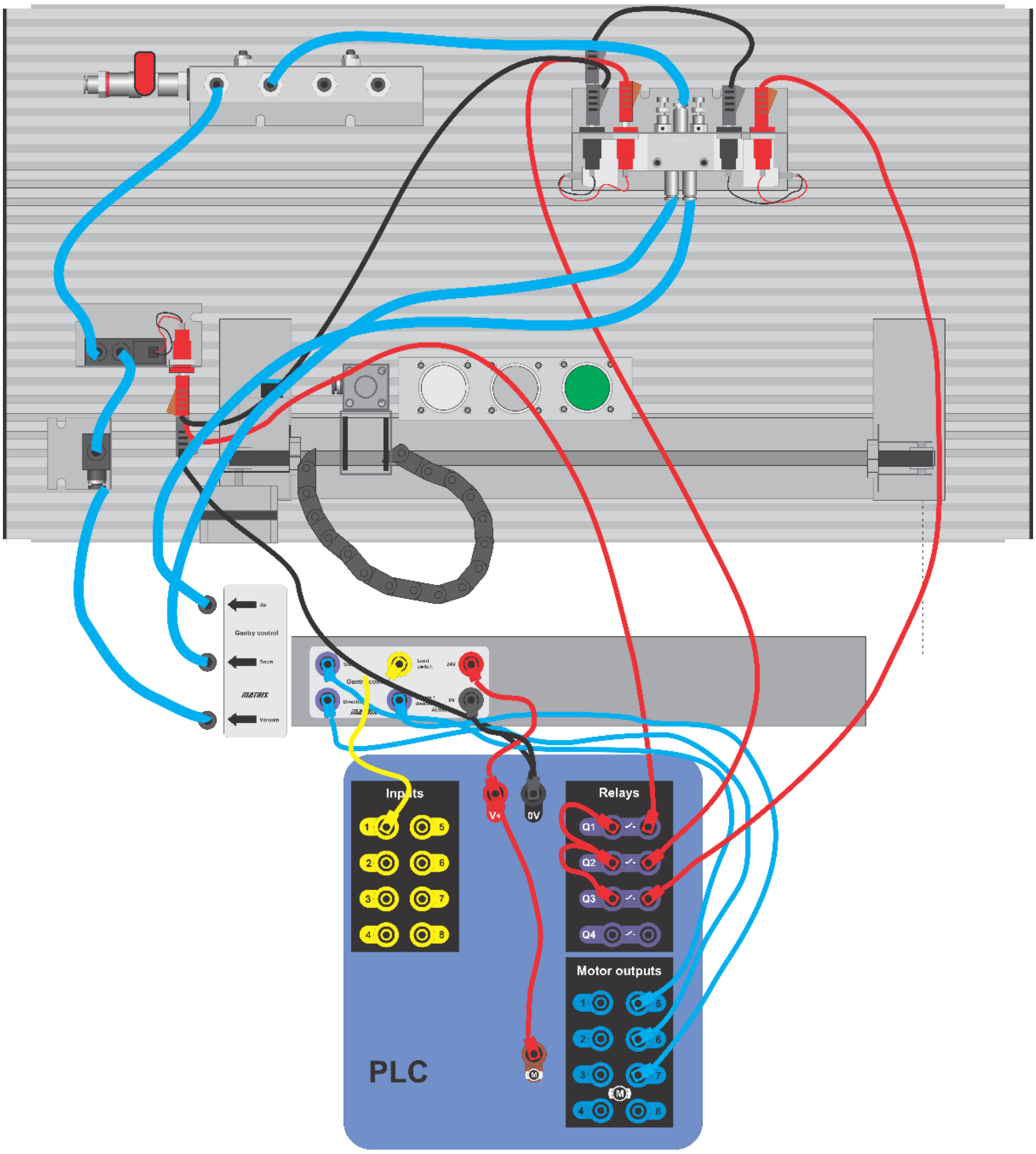
- Wire up the gantry mechanism to your PLC as shown above.
- The plunger is a simple push out, push in mechanism. The 5/2 valve is used to channel the flow of air pressure to first push the plunger piston out and then to push the piston in.
- In the circuit we have shown relays used to switch 24V to the 5/2 valve for the plunger control. If your PLC has spare transistor outputs then you can use those.
- Develop a simple program to make sure that the default position of the plunger is in and make sure you can push the plunger up and down.
- Vacuum for the suction cup on the end of the plunger is generated by a special vacuum unit. This connects to the plunder via a simple solenoid valve.
- Develop a program that allows you to turn the suction on and off.
- You now have control of the stepper motor, the plunger and the suction cup.

## Worksheet 7 – Delivering Counters

XY gantries with suction devices are often used to move a production part from one operation to another. Now that you have understood the movement of the stepper motors, the plunger, and the suction cup you can put all this together to produce a system that picks up counters from the counter storage unit and places them on the Conveyor.



Photograph shows a suction cup and sensor on a production line.



- You are now ready to put the gantry program together.
- Develop a program that can deliver 9 counters - 3 from each storage area - to the ‘conveyor belt’. You will have to assume 9 counters are always present as there is no feedback from the plunger mechanism.
- Of course at this stage the gantry is not connected to the conveyor belt so you will simply release the counter at an appropriate point.

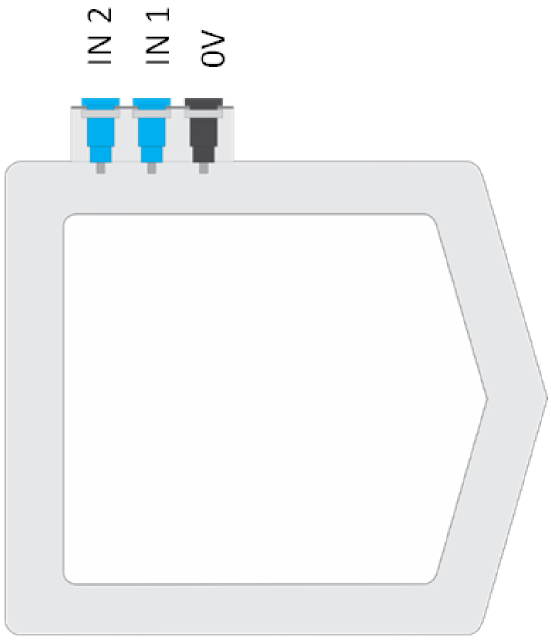
 

## Worksheet 8 – Robot Arm

In a robot work cell there will be several programmable robotic modules that need to work together. This can be achieved by sending messages from one module to another, or it can be achieved by a much more simple method of using spare input outputs as 'flags' to indicate certain actions can be taken.

Photograph shows a work cell with a robot arm loading a CNC lathe.

**Over To You**




- Find the robot arm curriculum pack on our web site.
- Go through worksheets 1, to 3 to make sure you can program the robot arm using the pendant programming tool and G code.
- The G code editor will allow you to detect an input and then start a routine. You can use this to allow the PLC to trigger an action on the robot arm.
- If you want to use the colour sorter then you will need to use a proper programming language and the robot arm Application Programming Interface.
- If you are planning a full Industry 4.0 implementation then you will need to use a programming language that has the ability to create internet based communications such as Flowcode App Developer.

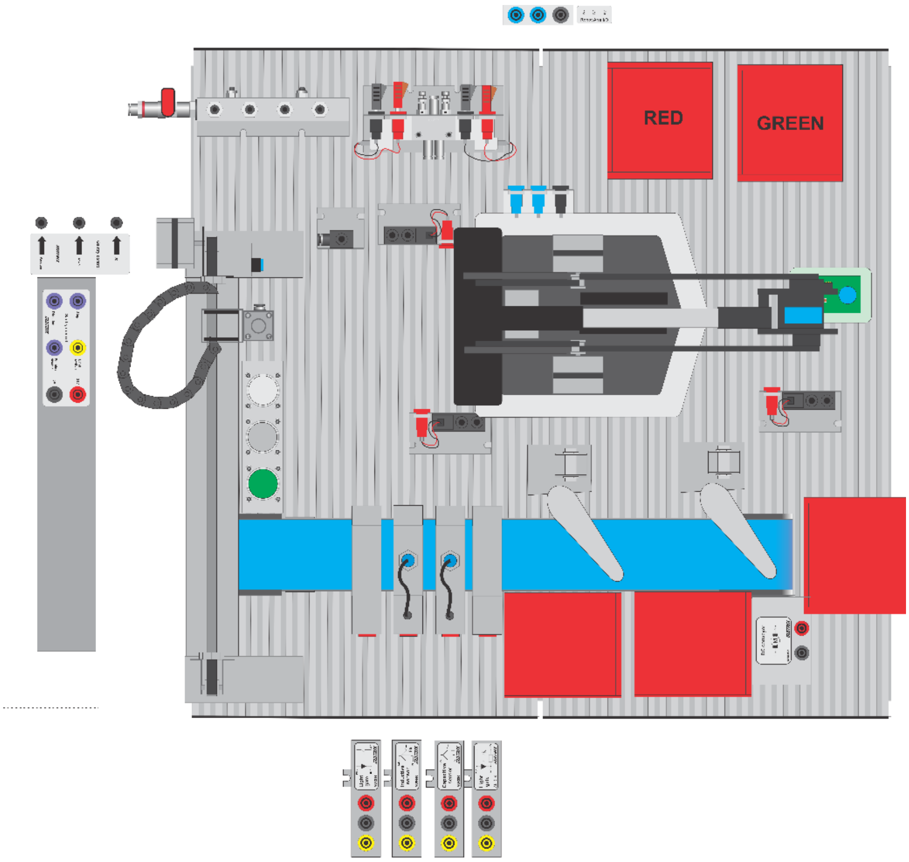
*NOTE: Remember that the Robot arm runs of 12v system and has its own power supply provided. Please don't run using AU0205 as this is 24v.*

## Worksheet 9 – Commissioning The Cell

The development of a production line will involve many engineers and many sets of technology. Now that the individual parts of the smart factory have been commissioned individually its time to bring all the parts together.

Photograph shows two robots in a  packaging work cell.



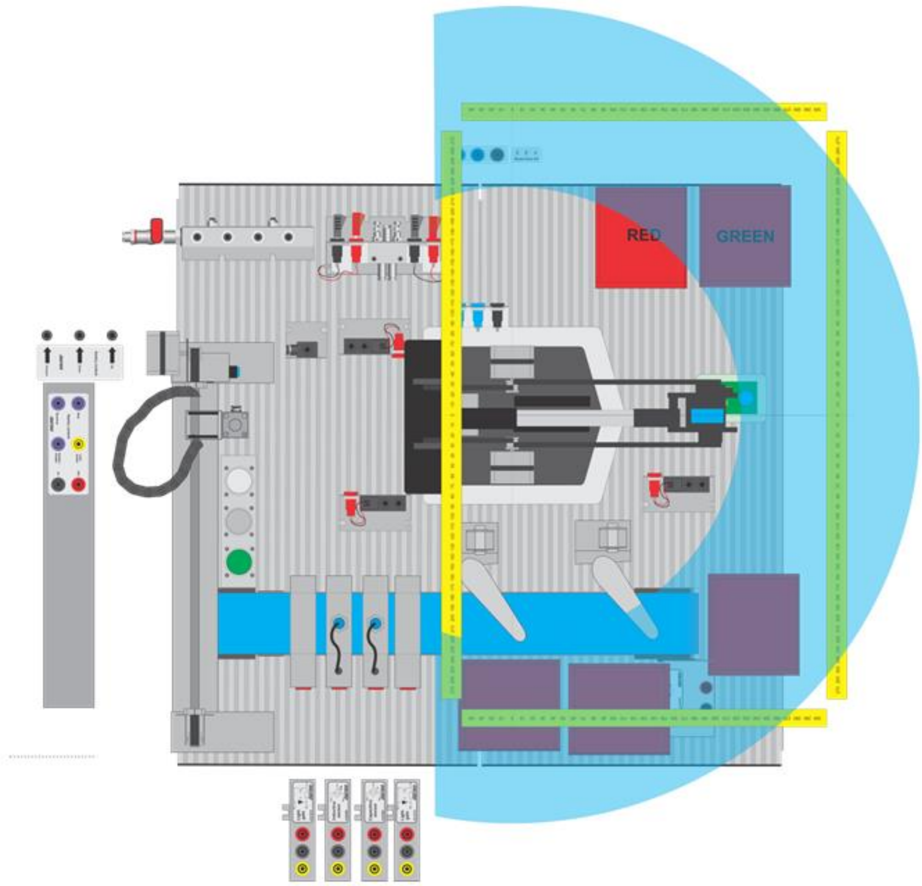

### Over To You

- Bolt the two platforms together.
- Rearrange the Gantry and the Conveyor onto the platforms as shown above.
- Position the robot arm onto  the centre of the two platforms. You can use the registration plate  to mark the position of the arm so that it can be removed from the system and easily put back in the right place.
- Gantry: fine tune the program you wrote earlier to deliver 9 counters in succession to the conveyor.

## Worksheets

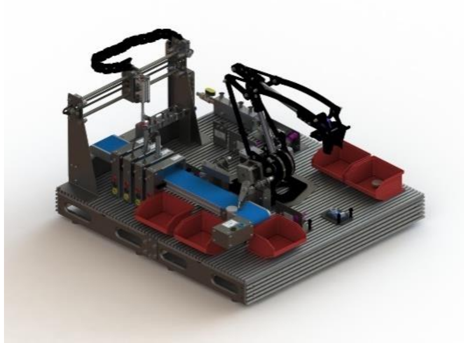
- Gantry and Conveyor: make sure that the Gantry and Conveyor programs work together to delivery and sort 9 counters of varying types. For now there is no communication between the conveyor and the gantry: just use appropriate timing. You can use an extra input and output on your controller to develop communications between the Gantry and the Conveyor.
- Robot arm: Alter the counter sorting program you wrote earlier to pick up a plastic counter from the end of the conveyor, test its colour with the colour sensor, and deposit it into the 'RED' or 'GREEN' bin on the workspace. You can see the workspace plan in the image below. You can do this with the G code editor program. If you want to use the colour sorter to differentiate between colours then you will need to use a more sophisticated programming language ( like Flowcode App Developer). To make your programming task easier you will need to develop a measuring system so that you know the locations of the conveyor, colour sensor and bins.

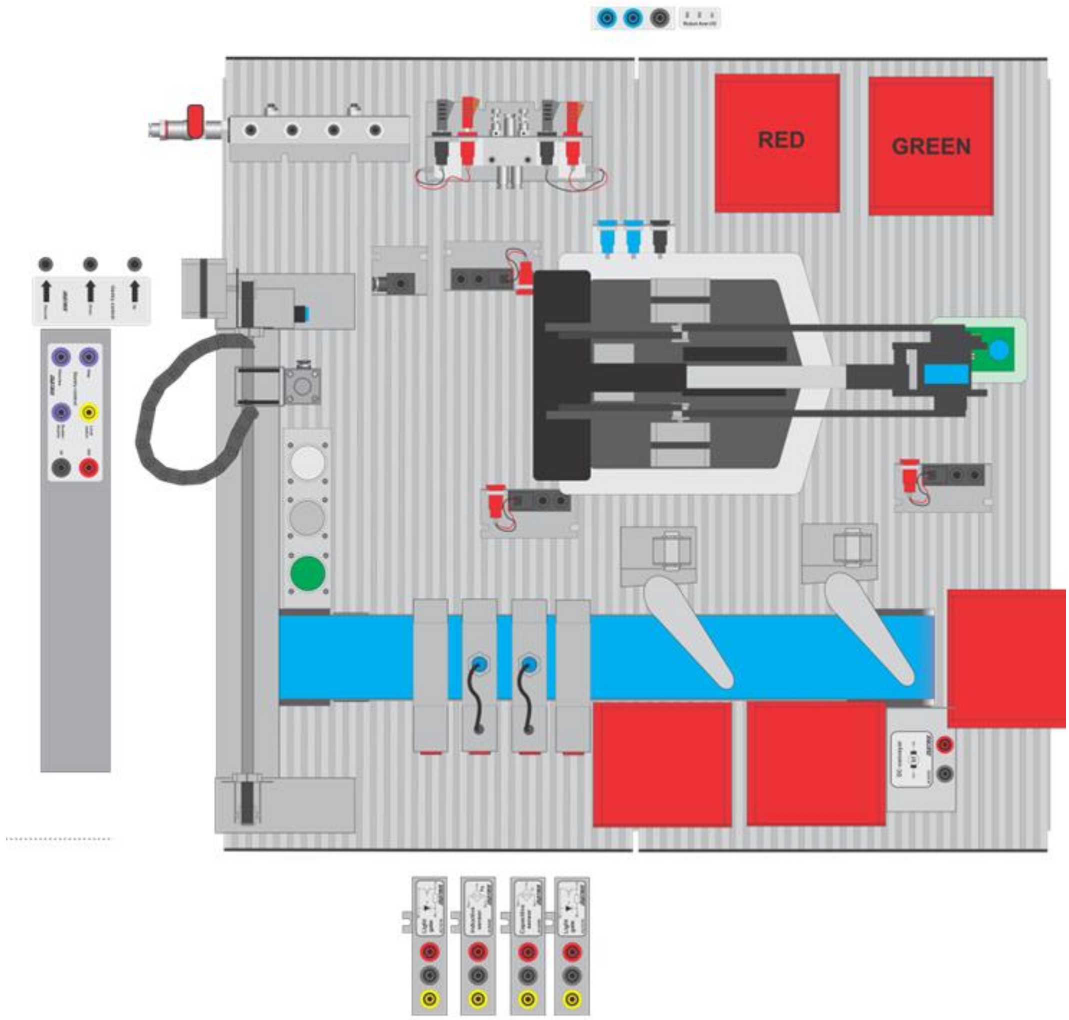


## Worksheets

## Worksheet 10 – Completing the Smart Factory

Many factories now make use of web based control systems. This offers lots of benefits including remote system monitoring, maintenance and even programming - allowing technicians to work on systems without being there. Because IP technology is so widespread the boundaries between building based and world based systems is becoming more indistinguishable.





- Next develop your Conveyor program so that when a plastic counter is detected it stops at an appropriate point so that the robot arm can pick up the counter and sort it. You will need to connect an output of your PLC to the IN1 or IN2 input on the robot arm.
- Develop a robot arm program to pick the counter, pass it over the colour sensor and sort it by red/green colour. You will need to initialise this manually as there is no communication between the arm and the other parts of the system.
- The robot arm is fitted with Wi-fi connectivity and has a full Application Programming Interface that allows it to communicate with other LAN/Wi-fi connected products. Develop communications between the arm and the other parts of the system so that the Smart factory works in a fully automatic mode to sort the workpieces.

## Worksheet 11 – Defects & Reset Sequence

No system runs perfectly all the time. Sensors can fail, parts can jam, and unexpected issues can occur. Smart Factory 2 includes a fault simulation feature to help you practise troubleshooting. In this worksheet, you'll insert defects such as a missing part, blocked sensor, or actuator fault, then update the PLC program and HMI to detect, respond, and reset correctly.


**Over To You**

1. Load and start the Smart Factory program on the HMI. Make sure the conveyor, gantry, and robot arm are all in their home positions before starting the sequence.
2. Run a normal production cycle until a defective counter is detected by the vision or sensor system. The HMI will show a Defect Detected message and highlight the affected station.
3. When the fault occurs, the system will pause automatically. The HMI will present options such as:
   - *View defect details* (shows sensor readings, counter type, or photo if a vision module is connected).
   - *Acknowledge defect* (confirms you've reviewed the issue).
   - *Discard item* (removes the counter from process).
   - *Approve and re-sort* (returns the counter to the sorting sequence).
4. Use the on-screen Acknowledge / Clear Defect button to confirm that the issue has been dealt with. The HMI indicator for "Defect Active" should turn off.
5. Once all active defects are cleared, press Reset Sequence on the HMI. This command will:
   - Move the robot arm and gantry to their home positions.
   - Reset the conveyor and reject mechanisms.
   - Clear production counters and restore the system to its initial state.
6. Press Start Cycle again to resume automatic operation.

## So What?

Being able to manage defects and perform a full system reset from the HMI is a key feature of modern automated control systems. In real industrial environments, this function prevents unplanned downtime by allowing operators to handle process faults through software logic rather than manual intervention. The PLC records the defect condition, halts motion safely, and waits for operator acknowledgment before resuming.

When the Reset Sequence is triggered, the PLC reinitialises all outputs, clears internal memory flags, and re-homes key actuators such as the conveyor, gantry, and reject paddles. This ensures that no cylinders remain extended, valves left open, or counters double-counted between runs. The process demonstrates how HMIs, PLC logic, and interlocks work together to achieve a controlled recovery routine, essential for maintaining system integrity, traceability, and safety in automated production lines.

### Key Takeaways

- The Smart Factory's defect handling system shows how automated lines detect, isolate, and recover from process faults using PLC and HMI control.
- Managing defects through the HMI gives operators real-time visibility of system status, sensor data, and part quality for informed decision-making.
- Communication between the HMI and PLC ensures all actuators move to a safe state before restarting, preventing damage or incorrect sorting.
- The Reset Sequence command reinitialises all main components including the conveyor, gantry, robot arm, and pneumatic valves to a known start position.
- This process reinforces key industrial control concepts such as safe shutdown, fault recovery, data logging, and process traceability.
- Understanding and applying these methods prepares students to maintain and troubleshoot automated systems that meet professional safety and reliability standards.

### Student Handbook

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheet 12 – Vision System

Machine vision helps factories inspect, guide, and sort items by detecting colours, shapes, or text that standard sensors can't. Smart Factory 2 uses a Raspberry Pi camera for this. In this worksheet, you'll calibrate the camera, detect counter attributes like colour or shape, and send that data to the PLC to affect sorting and rejection.

**Over To You**

1. Power on the Smart Factory and ensure the conveyor, gantry, and camera module are connected.
2. On the HMI, open the **Vision System** screen. Observe the live camera feed or simulated view of the inspection area.
3. Place different counters on the conveyor and allow them to pass under the camera. The HMI will display inspection results such as **Pass**, **Fail**, or **Unidentified**.
4. Use the HMI sliders or input fields to adjust settings such as **Brightness**, **Threshold**, or **Colour Sensitivity**. Note how these changes affect the system's ability to detect differences between materials or spot surface defects.
5. Deliberately place a scratched, discoloured, or incorrectly oriented counter on the conveyor. Confirm that the system flags it as a defect and logs it to the defect list.
6. When finished, reset all settings to default and clear the inspection log to prepare for the next run.

**So What?**

The vision system simulates a common quality control stage used in manufacturing, where cameras and image processing software detect faults faster and more accurately than manual inspection. Adjusting thresholds and lighting parameters demonstrates how vision algorithms depend on contrast, colour balance, and tolerance settings to classify parts correctly. In the Smart Factory, these parameters are linked to the PLC through the HMI, showing how real-world inspection data can drive automation decisions such as rejecting defective parts or triggering maintenance actions.

Understanding how the system analyses images and communicates with the PLC gives students insight into how modern factories use digital vision to maintain consistent product quality.

**Key Takeaways**

- The vision module automates product inspection using image analysis to detect surface and material defects.
- HMI controls allow fine-tuning of camera parameters, demonstrating how sensitivity and lighting influence detection accuracy.
- Inspection results are communicated to the PLC, which can automatically trigger reject mechanisms or log faults.
- The exercise introduces key principles of industrial vision, including thresholding, calibration, and real-time decision-making.
- Students learn how optical inspection integrates with PLC-based automation for quality assurance and production efficiency.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheet 13 – RFID

RFID technology lets factories track parts, bins, or pallets wirelessly using tags and a reader. Smart Factory 2 includes an RFID reader to improve counter tracking. In this worksheet, you'll set up the reader, read tag data via the PLC or Pi, and use it in the control logic to log or customise item processing.


### Over To You

1. Power on the Smart Factory and open the **RFID** screen on the HMI. Ensure the RFID reader is connected to the PLC or Raspberry Pi via the communication port.
2. Place an RFID-tagged counter or tag card near the reader and observe the **Tag Detected** indicator on the HMI. The tag ID number should appear automatically in the data field.
3. Move the tag slightly away and reintroduce it to learn how detection range affects reliability. Adjust any sensitivity or scan interval settings on the HMI if available.
4. Use the **Log Tag** button to record the tag data to the system log. You should see the tag ID, time stamp, and part type stored in the data table.
5. Assign a process action to a specific tag (for example, direct Tag 001 to Bin 1 or apply a special inspection). Modify the control logic or recipe selection using the HMI dropdown.
6. Run the conveyor and confirm that when the tagged part is read, the system automatically performs the correct programmed action.
7. Finally, clear the tag log and restore default settings for the next user.

### So What?

RFID technology allows automatic identification without direct line of sight, unlike barcodes or optical sensors. In industrial settings, RFID enables flexible production management, where each part carries its own digital identity. The PLC or edge processor reads the tag data, interprets it through control logic, and makes decisions such as routing parts, applying quality checks, or logging production data. By experimenting with tag parameters and observing the live data exchange through the HMI, students gain insight into how modern factories use RFID for real-time traceability, inventory control, and adaptive manufacturing.

**Key Takeaways**

- RFID enables wireless identification of parts or materials for tracking and custom process control.
- The HMI allows users to view, log, and assign actions based on unique tag IDs.
- Integration with the PLC or Raspberry Pi shows how tag data influences automation logic.
- RFID improves traceability and supports smart manufacturing by linking physical items to digital data.
- Understanding this system prepares students to design and troubleshoot industrial tracking systems that rely on data-driven control.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheets

## Worksheet 14 – Network & Communications

Modern factories use Ethernet networks to link PLCs, HMIs, robots, and sensors. In Smart Factory 2, the Siemens PLC connects with the Raspberry Pi for data exchange. This worksheet covers setting up that network link using Modbus TCP or similar and enabling remote monitoring or control.

### Over To You

- Power on the Smart Factory and connect the Siemens PLC, HMI, and Raspberry Pi to the same Ethernet switch or hub.
- On the HMI, open the **Network Setup** screen and confirm that the PLC and Pi IP addresses are displayed. If not, use the on-screen fields to assign suitable static IPs (for example, 192.168.1.10 for the PLC, 192.168.1.20 for the Pi).
- Open the **Communications Test** page on the HMI and select **Ping Device** to check that both units are reachable across the network.
- Enable **Modbus TCP** (or Snap7 if available) on the Raspberry Pi application. Enter the PLC's IP address and port number, then select the data register or tag to read.
- Monitor the live data feed on the HMI. You should see values such as counter totals, conveyor speed, or defect status updating in real time from the Pi.
- Use the HMI controls to send a command back to the Pi, for example, toggling a simulated process variable or requesting data logging to start.
- Disconnect and reconnect one device to observe how the system detects a loss of communication and resumes automatically when the link is restored.

### So What?

Establishing reliable communication between controllers, sensors, and external processors is essential in Industry 4.0 systems. Using Ethernet-based protocols such as Modbus TCP, Snap7, or MQTT allows the Smart Factory to exchange data instantly between the PLC, Raspberry Pi, and HMI.

## Worksheets

This architecture mirrors real industrial networks, where local PLCs handle control while edge devices manage analytics, databases, or cloud connectivity. Understanding how to configure IP addresses, test connectivity, and manage data flow gives students practical experience in designing connected automation systems that support IIoT integration, remote diagnostics, and smart manufacturing workflows.

**Key Takeaways**

- Industrial communication networks allow different controllers and devices to share data for coordinated operation.
- Modbus TCP or Snap7 provide standard, reliable methods for data exchange between PLCs, HMIs, and edge processors.
- IP configuration, connectivity testing, and error recovery are core skills for modern automation engineers.
- Networked communication enables remote monitoring, data logging, and integration with analytics or cloud systems.
- Understanding these principles helps students build scalable, connected systems that reflect real Industry 4.0 architecture.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheet 15 – Data Logging

Data logging collects and stores production data like counts, times, and faults for analysis and traceability. In Smart Factory 2, the Raspberry Pi acts as the data hub. In this worksheet, you'll set up logging on the Pi, record key data from the PLC, and save it in a usable format such as CSV or a database.

### Over To You

1. Power on the Smart Factory and ensure the PLC, HMI, and Raspberry Pi are connected via Ethernet.
2. On the HMI, open the Data Logging screen and check that the Pi connection is active.
3. On the Raspberry Pi, start the Logging Service or open the logging application.
4. Select which data points you want to log, such as:
   - Total counters processed
   - Conveyor motor status
   - Defect flags or reject counts
   - Energy or cycle time (if available)
5. Choose a storage format, either a CSV file for manual review or an SQL database for long-term storage.
6. Run the Smart Factory for several minutes and monitor the live data being written. You should see new entries appearing in the HMI display or database viewer.
7. Stop the process, open the stored data file, and verify that all logged values match the system's real-time behaviour.
8. Clear or archive the log before starting a new session.

### So What?

Data logging forms the foundation of process monitoring, performance optimisation, and predictive maintenance. By automatically recording system data, engineers can identify trends such as increasing cycle times, rising defect rates, or inconsistent sensor readings.

## Worksheets

The Raspberry Pi acts as an edge data server, communicating with the PLC to collect structured information and store it securely. This mirrors modern factory setups where local databases or cloud systems track every operation for quality assurance and traceability.

Learning how to define, collect, and manage logged data helps students understand how industrial systems use information for decision-making, continuous improvement, and compliance with production standards.

**Key Takeaways**

- Data logging captures key operational information such as counts, faults, and timing data for later analysis.
- The Raspberry Pi functions as an edge data hub, storing information from the PLC in CSV or database format.
- Logged data supports troubleshooting, predictive maintenance, and process optimisation.
- Understanding how to configure and interpret data logs helps students link automation hardware with data analytics tools.
- This exercise demonstrates how modern factories use continuous data capture to improve efficiency, quality, and traceability.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheet 16 – Analytics

Raw data is valuable, but analysing it reveals trends and insights. In this worksheet, you'll use the data from Worksheet 15 to calculate metrics like throughput and quality, create charts, and spot patterns such as faults increasing with speed, using tools on the Raspberry Pi or a PC.

**Over To You**

1. Open the Data Analytics screen on the HMI or launch the analytics tool on the Raspberry Pi.
2. Load the most recent data log (CSV or SQL database) created in Worksheet 15.
3. View the raw dataset and identify key variables such as cycle time, total output, defect count, and conveyor speed.
4. Use the built-in plotting tool or spreadsheet software to generate charts showing:
   - Throughput (items per minute) over time
   - Defect rate versus conveyor speed
   - Downtime duration or frequency of stops
5. Adjust parameters such as time range or sample size to explore different trends.
6. Interpret your graphs: look for correlations, e.g. higher defect rates at increased speeds or improved consistency after a reset.
7. Create a simple summary report showing calculated metrics such as:
   - OEE (Overall Equipment Effectiveness)
   - FPY (First Pass Yield)
   - Average cycle time
8. Save your report and discuss how these insights could guide improvements in a real manufacturing environment.

**So What?**

Analytics transforms raw data into actionable knowledge. In industrial environments, engineers rely on data interpretation to optimise performance, reduce waste, and predict failures. By analysing logged Smart Factory data, you can assess how effectively the system runs and pinpoint causes of inefficiency. Metrics like OEE and FPY provide a quantitative measure of productivity and quality, while trend graphs reveal underlying process behaviour.

Performing this analysis mirrors real-world manufacturing practices where continuous improvement is driven by evidence-based decision-making supported by live production data.

**Key Takeaways**

- Analysing logged data helps reveal performance trends, process inefficiencies, and quality issues.
- Metrics such as throughput, OEE, and FPY provide measurable indicators of system productivity.
- Visualising data with charts allows rapid identification of patterns like rising defect rates or irregular cycle times.
- Analytics links operational data with engineering decisions, forming the basis of continuous improvement.
- Understanding how to interpret production data prepares students to apply real industrial performance monitoring techniques.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheets

## Worksheet 17 – IO Link

IO-Link is a smart interface that provides detailed sensor data and remote configuration. Smart Factory 2 includes an IO-Link master and devices. In this worksheet, you'll read advanced data (like distance or device ID) from an IO-Link sensor and adjust its settings to see how IO-Link simplifies setup, maintenance, and diagnostics.

**Over To You**

1. Power on the Smart Factory and open the IO-Link screen on the HMI.
2. Confirm that the IO-Link master and connected sensors (for example, inductive or capacitive) are detected. The HMI should display each device's Port Number and Device ID.
3. Select one of the connected IO-Link sensors from the list. Review its live data values such as distance, signal quality, or switching threshold.
4. Use the HMI controls to adjust the sensor's configuration — for example, changing the switching point or response time. Observe the effect in real time on the HMI data display.
5. Intentionally disconnect or cover the sensor to simulate a fault. Watch how the IO-Link system generates diagnostic information such as Device Error or Communication Lost, displayed automatically on the HMI.
6. Reconnect the device and confirm that the system resumes normal operation without manual reconfiguration.
7. Record your observations, noting how IO-Link simplifies wiring, identification, and maintenance compared to traditional sensors.

**So What?**

IO-Link represents a major step forward in sensor communication. It converts simple binary or analogue connections into a bidirectional digital link, allowing both real-time process values and diagnostic data to flow between devices and the PLC. This eliminates manual sensor calibration and reduces downtime caused by unknown faults.

## Worksheets

In Smart Factory 2, the IO-Link master acts as a gateway, giving the PLC full access to sensor parameters, health data, and configuration options. Students can see how this technology supports predictive maintenance, remote setup, and plug-and-play replacement of sensors — all key aspects of Industry 4.0 and smart manufacturing.

**Key Takeaways**

- IO-Link enables two-way digital communication between sensors and the PLC using standard cabling.
- Live process data and diagnostic information can be viewed and adjusted directly from the HMI.
- Device configuration and parameter changes can be made remotely without rewiring.
- IO-Link improves system uptime through automatic device detection and fault diagnostics.
- Understanding IO-Link prepares students to design and maintain future-ready automation systems that support smart monitoring and rapid maintenance.

**Student Handbook**

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

## Worksheet 18 – Predictive Maintenance

Predictive maintenance uses system data to plan service before failures occur. With Smart Factory 2's sensors and data logging, you'll monitor indicators like cycle count or signal quality to predict faults. This worksheet simulates wear or failure to show how the system can warn of maintenance needs in advance.

### Over To You

1. Power on the Smart Factory and open the Maintenance Dashboard screen on the HMI.
2. Observe live readings such as cycle count, motor current, sensor signal quality, or plunger travel time. These are the parameters typically used to monitor component health.
3. Start a continuous operation cycle and allow the system to run for several minutes. Watch how the cycle counter and operation hours increase.
4. Gradually simulate a developing fault, for example:

    - Partially block a sensor to reduce its signal quality.
    - Increase conveyor load slightly to simulate motor strain.
    - Delay pneumatic response by restricting airflow.

5. Note how the HMI begins to display Warning or Service Due alerts once parameters drift outside their expected range.
6. Record the data log and identify which trend triggered the early warning.
7. Use the Acknowledge Maintenance or Reset Counter button on the HMI to confirm the fault has been reviewed and the system serviced.
8. Resume operation and verify that the warning clears and the system returns to normal.

## So What?

Predictive maintenance combines sensor feedback, data logging, and analytics to improve reliability and reduce unplanned downtime. Rather than waiting for faults to occur, the system monitors condition-based indicators such as temperature, vibration, current draw, or signal stability. When these values deviate from normal ranges, the PLC or edge processor triggers early maintenance alerts.

In Smart Factory 2, this demonstrates how live data from sensors and actuators can be used to anticipate issues, allowing maintenance teams to schedule service efficiently. Students gain insight into how Industry 4.0 systems use predictive algorithms and trend data to extend component life, improve availability, and cut operational costs.

### Key Takeaways

- Predictive maintenance uses real-time data to identify signs of wear or failure before breakdowns occur.
- Monitoring parameters such as cycle count, signal strength, or actuator speed helps detect abnormal trends.
- Automated alerts on the HMI support proactive maintenance scheduling and reduce downtime.
- Logged performance data can be analysed to refine maintenance intervals and optimise system reliability.
- This exercise demonstrates how data-driven diagnostics form the foundation of modern, connected maintenance strategies.

### Student Handbook

Complete The Questions & Answers in the Student Handbook relating to this worksheet.

# STUDENT HANDBOOK

**Questions:**

1. What is the purpose of a closed-loop flow control system?

_______________________________________________

_______________________________________________

2. What must you check before starting the system?

_______________________________________________

_______________________________________________

3. What happens when you restrict the flow using the hand valve?

_______________________________________________

_______________________________________________

4. What role does the PLC play in the system?

_______________________________________________

_______________________________________________

5. Why should you avoid making rapid changes to the valve?

_______________________________________________

_______________________________________________

## Worksheet 2 – Reject Mechanisms

**Questions:**

1. What is the main purpose of an Emergency Stop (E-Stop)?

2. How do you reset the E-Stop after it has been pressed?

3. What makes an E-Stop more reliable than a normal stop button?

4. When should an E-Stop be used?

5. Why is it important to regularly inspect the E-Stop?

**Questions:**

1. What is the purpose of the Status LED in the system?

____________________________________________________________

____________________________________________________________

2. What does a green Status LED typically indicate?

____________________________________________________________

____________________________________________________________

3. What colour does the LED turn when a fault is detected, but the system may still function?

____________________________________________________________

____________________________________________________________

4. How can you trigger a safety fault to observe the LED behaviour?

____________________________________________________________

____________________________________________________________

5. Why is it important to monitor the Status LED regularly?

____________________________________________________________

____________________________________________________________

**Questions:**

1. What is a PLC and what is it designed for?

______________________________________________________________________

______________________________________________________________________

2. What do the status LEDs on the PLC indicate?

______________________________________________________________________

______________________________________________________________________

3. What does it mean if an input LED expected to be on is not lit?

______________________________________________________________________

______________________________________________________________________

4. What should you observe on the output LEDs when the system starts?

______________________________________________________________________

______________________________________________________________________

5. How can you tell the flow sensor is working by looking at the PLC?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. What is the purpose of a Human-Machine Interface (HMI)?

_______________________________________________________________

_______________________________________________________________

2. Where can you view system faults and alarms on the HMI?

_______________________________________________________________

_______________________________________________________________

3. What happens when you adjust the High or Low Temperature Cutout settings beyond the current temperature?

_______________________________________________________________

_______________________________________________________________

4. How does the HMI assist with troubleshooting and maintenance?

_______________________________________________________________

_______________________________________________________________

5. What real-time information can be seen on the main page of the HMI during system operation?

_______________________________________________________________

_______________________________________________________________

**Questions:**

1. What is the function of a pump in an industrial system?

___

___

2. How can a pump be controlled to vary its speed?

___

___

3. What might it indicate if a pump is running at full speed but still not meeting the setpoint?

___

___

4. Why must pumps not be run dry, and what protects against this?

___

___

5. What are some signs that maintenance engineers should look for to detect pump issues early?

___

___

**Questions:**

1. What is the main function of a valve in an industrial system?

______________________________________________________________________

______________________________________________________________________

2. How is the valve position controlled in this system?

______________________________________________________________________

______________________________________________________________________

3. What is meant by the term "deadband" in valve control?

______________________________________________________________________

______________________________________________________________________

4. What should you do to check if a valve is active during operation?

______________________________________________________________________

______________________________________________________________________

5. Why is it important to observe valve behaviour during an Emergency Stop?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. What is the main purpose of a float switch in an industrial system?

___

___

2. What is the difference between a normally open (NO) and a normally closed (NC) float switch?

___

___

3. Why is the low-level float switch normally open (NO)?

___

___

4. Why is the high-level float switch normally closed (NC)?

___

___

5. What are some common float switch faults, and how can they be identified?

___

___

**Questions:**

1. What is a proximity switch and what does it detect?

______________________________________________________________________

______________________________________________________________________

2. What are the three main types of proximity switches mentioned?

______________________________________________________________________

______________________________________________________________________

3. What problems can a faulty or misaligned proximity switch cause?

______________________________________________________________________

______________________________________________________________________

4. What should you check if a proximity switch fails to trigger correctly?

______________________________________________________________________

______________________________________________________________________

5. Why is proper mounting and alignment important for proximity switches?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. What does a flow sensor measure and why is it important in automated systems?

______________________________________________________________________

______________________________________________________________________

2. How does a turbine flow sensor work?

______________________________________________________________________

______________________________________________________________________

3. What role does the PLC play in reading the flow sensor data?

______________________________________________________________________

______________________________________________________________________

4. What might unstable or incorrect flow readings indicate?

______________________________________________________________________

______________________________________________________________________

5. Why is regular calibration of flow sensors important?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. What is the main purpose of a temperature sensor in an automated system?

_______________________________________________________________________

_______________________________________________________________________

2. How does an RTD temperature sensor work?

_______________________________________________________________________

_______________________________________________________________________

3. What happens if the temperature exceeds the high cutout limit set on the HMI?

_______________________________________________________________________

_______________________________________________________________________

4. What could cause a temperature sensor to give incorrect or no readings?

_______________________________________________________________________

_______________________________________________________________________

5. Why is calibration important for temperature sensors?

_______________________________________________________________________

_______________________________________________________________________

**Questions:**

1. What type of signal does a digital sensor send to a PLC?

___________________________________________________________

___________________________________________________________

2. How can you monitor the status of a digital input?

___________________________________________________________

___________________________________________________________

3. What is the difference between a Normally Open (NO) and Normally Closed (NC) sensor?

___________________________________________________________

___________________________________________________________

4. Why are Emergency Stop buttons typically wired as Normally Closed (NC)?

___________________________________________________________

___________________________________________________________

5. Can you name the seven digital inputs used in the training system?

___________________________________________________________

___________________________________________________________

**Questions:**

1. What is the only analogue sensor used on this training rig, and what does it measure?

_______________________________________________________________

_______________________________________________________________

2. What voltage signal does the temperature sensor send to the PLC, and what temperature range does it represent?

_______________________________________________________________

_______________________________________________________________

3. What does the PLC do with the analogue voltage signal it receives from the sensor?

_______________________________________________________________

_______________________________________________________________

4. What happens if the sensor range changes from 0–100°C to 0–300°C but the PLC scaling is not updated?

_______________________________________________________________

_______________________________________________________________

5. Why might an offset be used, and how does it affect the final temperature value?

_______________________________________________________________

_______________________________________________________________

**Questions:**

1. Why is fault finding important in industrial maintenance?

___

___

2. What are the three main types of faults in industrial systems?

___

___

3. How does the HMI help with diagnosing faults?

___

___

4. What is the purpose of checking the alarm raise time on the HMI?

___

___

5. How can practising with simulated faults improve troubleshooting skills?

___

___

**Questions:**

1. Why is fault finding important in industrial maintenance?

______________________________________________________________________

______________________________________________________________________

2. What are the three main types of faults in industrial systems?

______________________________________________________________________

______________________________________________________________________

3. How does the HMI help with diagnosing faults?

______________________________________________________________________

______________________________________________________________________

4. What is the purpose of checking the alarm raise time on the HMI?

______________________________________________________________________

______________________________________________________________________

5. How can practising with simulated faults improve troubleshooting skills?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. Why is fault finding important in industrial maintenance?

______________________________________________________________________

______________________________________________________________________

2. What are the three main types of faults in industrial systems?

______________________________________________________________________

______________________________________________________________________

3. How does the HMI help with diagnosing faults?

______________________________________________________________________

______________________________________________________________________

4. What is the purpose of checking the alarm raise time on the HMI?

______________________________________________________________________

______________________________________________________________________

5. How can practising with simulated faults improve troubleshooting skills?

______________________________________________________________________

______________________________________________________________________

**Questions:**

1. Why is fault finding important in industrial maintenance?

____________________________________________________________

____________________________________________________________

2. What are the three main types of faults in industrial systems?

____________________________________________________________

____________________________________________________________

3. How does the HMI help with diagnosing faults?

____________________________________________________________

____________________________________________________________

4. What is the purpose of checking the alarm raise time on the HMI?

____________________________________________________________

____________________________________________________________

5. How can practising with simulated faults improve troubleshooting skills?

____________________________________________________________

____________________________________________________________

**Questions:**

1. Why is fault finding important in industrial maintenance?

___

___

2. What are the three main types of faults in industrial systems?

___

___

3. How does the HMI help with diagnosing faults?

___

___

4. What is the purpose of checking the alarm raise time on the HMI?

___

___

5. How can practising with simulated faults improve troubleshooting skills?

___

___

## Safety Note

## SAFETY NOTE

As you step into the world of industrial maintenance, **safety must always come first**. Every machine, tool, and system you work with has risks, but following the right procedures keeps you and your team safe.

There are official safety standards and regulations, like the **Machinery Directive 2006/42/EC**, that exist to ensure all maintenance work is carried out correctly and safely. In the **EU** there is the **EN 60204-1** (electrical safety for machinery) and in the **US**, **OSHA 1910** (general industry safety) and **NFPA 70E** (electrical safety for workers). These standards help maintenance engineers work safely, reduce risks, and comply with legal requirements. Once you get into Industry, get a copy of these and learn them. They may save your life. The best engineers aren't just good at fixing machines; they're smart enough to work safely.

**1. Lock It Out, Stay Safe**

Before you touch any machine, make sure it's isolated from power. Use Lockout/Tagout (LOTO) procedures to lock off energy sources (electricity, hydraulics, pneumatics). Machines can start unexpectedly, so always check before working.

**2. Read the Manual, Know the Rules**

Every machine has a manufacturer's manual it's there for a reason. Read it. Follow it. Never assume you know how something works without checking first.

**3. Wear the Right Gear**

Your Personal Protective Equipment (PPE) isn't optional. Safety glasses, gloves, boots, and hearing protection help prevent injuries that could end your career before it starts.

**4. Only Work on What You Understand**

If you're unsure about something, ask. Rushing in without the right knowledge can be dangerous. Learning the correct procedures now will make you a better and safer engineer in the long run.

## Safety Note

### 5. Keep Your Workspace Clear

Slips, trips, and falls are some of the most common accidents in maintenance. Keep tools, cables, and waste tidy to prevent hazards.

### 6. Electrical Safety is No Joke

Always turn off and test before working on electrical systems. Use insulated tools and check grounding a bad connection can be deadly.

### 7. Moving Parts Can Kill

Machines have stored energy, even when turned off. Always block, chock, or brace moving parts before working on them. Never assume something is safe just because it's not running.

### 8. Watch Out for Pressure

Hydraulics and pneumatics store energy in pipes, hoses, and tanks.

- Always depressurize systems before working on them.
- Inspect hoses and fittings for leaks never use your hands to check, use cardboard instead.
- A burst hose can cause serious injuries look for signs of wear before starting work.

### 9. Be Ready for Emergencies

Know where emergency stops, fire extinguishers, and first aid stations are. If something goes wrong, you need to act fast and correctly.

### 10. Speak Up About Faults

If you find something wrong don't ignore it. Report faults immediately, and don't use unsafe equipment. A quick fix now could prevent a serious accident later.

## Version Control

## VERSION CONTROL

| | |
|---|---|
| 30 10 25 | First Revision Created |