People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University M'Hamed BOUGARA- Boumerdes

Institute of Electrical and Electronic Engineering

Department of Power and Control

Final Year Project Report Presented in Partial Fulfilment of

The requirement for the degree of

Master

In **Electrical and electronic engineering**

Option: **Control engineering**

Title:

SIMULATION OF THE CONTROL OF RAW MATERIALS PRODUCTION LINE using FACTORY I/O & SIEMENS S7-1200

Presented by:

-Said Imad Eddine.

-Benkhelif Mohamed.

Supervisor:

Dr. Zitouni

Registration Number........... /2023

Abstract

This report presents an exploration and utilization of three software applications for industrial automation purposes. The initial software, TIA Portal, serves as a Programmable Logic Controller (PLC) programming tool, while the accompanying simulator, PLCSim, both developed by Siemens, facilitates the automation process of converting raw plastic and metal materials into refined products. The focus of this study lies not in the crafting of raw materials, but rather on the subsequent processes that occur after the materials have been refined. Within these processes, proximity sensors are employed to track the location of the materials, belt conveyors are utilized for material transportation, color sensors are employed for sorting the materials based on their respective colors, and robotic arms are utilized to mount the refined materials onto their designated containers. The third software application, Factory I/O, developed by Real Games, enables the creation of virtual factory environments that can be accessed from any location. In our project, we have constructed a tangible application of a real factory that incorporates a sorting and wrapping procedure dependent on the color of the final product. Additionally, we have simulated an equivalent system using Factory I/O. This approach allows students to experiment with their control mechanisms in a risk-free environment prior to their implementation in real-world systems. Moreover, this methodology aligns with the principles of the Industry 4.0 framework.

ACKNOWLEGEMENT

We would like to express our sincerest gratitude and warm appreciation to our project guide, Dr Zitouni for believing in our ability to work under his supervision.

Another special appreciation goes to all the teachers and workers of INELEC and everyone who contributes in our good formation at this institute.

Finely we are extremely grateful for our parents for their love, caring and sacrifices for educating and preparing us to the future.

SAID Imad eddine

BENKHELIF Mohamed

Boumerdes University

Institute of Electrical and Electronics Engineering

Control Engineering

Dedications

To my loving family and dear friends,

This achievement would not have been possible without your unwavering support, encouragement, and belief in me throughout my entire study career. You have been there for me in both the triumphs and the challenges, lifting me up when I needed it most and cheering me on every step of the way.

Your constant love, understanding, and motivation have been the driving force behind my success. Your words of wisdom, late-night discussions, and heartfelt conversations have shaped my perspective and guided me towards my goals. Your sacrifices and selflessness have shown me the true meaning of unconditional love and support.

I am forever grateful for the countless moments of celebration, the shoulder to lean on during moments of doubt, and the reassurance that I am never alone in this journey. Your belief in my abilities has fueled my determination, and your belief in my dreams has inspired me to reach for the stars.

To my family, your endless love, patience, and sacrifices have laid the foundation for my education. Your unwavering faith in me has given me the strength to persevere and overcome any obstacle. Thank you for being my pillars of strength and for always reminding me of the importance of education and continuous growth.

To my friends, you have been my companions, confidants, and partners in both studying and fun. Your laughter, camaraderie, and shared experiences have made this journey unforgettable. Thank you for your support, encouragement, and for always pushing me to be the best version of myself.

As I embark on the next chapter of my life, I carry with me the love, lessons, and memories we have shared. This accomplishment is as much yours as it is mine. Together, we have triumphed, and together, we will continue to celebrate future achievements.

With heartfelt gratitude and love,

SAID Imad Eddine

Dedications

To my beloved family and cherished friends,

Your unwavering support, unwavering belief, and endless encouragement have been the bedrock of my academic journey. In moments of triumph and moments of struggle, you have lifted me up and urged me forward.

Your ceaseless love, empathy, and motivation have propelled me to success. The wisdom you've shared, the late-night conversations we've had, and the heartfelt dialogues we've engaged in have shaped my perspective and steered me towards my aspirations. Through your selflessness and sacrifices, you have exemplified the true essence of unconditional affection and support.

I am eternally grateful for the numerous instances of celebration, the unwavering shoulder I could lean on in times of uncertainty, and the constant reminder that I am never alone on this expedition. Your unwavering faith in my abilities has fueled my determination, while your unwavering belief in my dreams has ignited my desire to reach for the highest of aspirations.

To my family, your boundless love, patience, and sacrifices have laid the groundwork for my education. Your unwavering faith in my abilities has provided me with the resilience and fortitude to conquer any obstacle. I extend my deepest appreciation for being my unwavering pillars of strength and consistently reminding me of the significance of education and perpetual growth.

To my friends, you have been my companions, confidants, and partners in both scholarly pursuits and moments of levity. The laughter we've shared, the camaraderie we've fostered, and the collective experiences we've created have etched indelible memories in my heart. Thank you for your unwavering support, encouragement, and the perpetual motivation to unleash my utmost potential.

With profound gratitude and boundless love.

BENKHELIF Mohamed.

List of Figures

Nomenclature

PLC	Programmable logic controller.
CPU	Central process unit.
TIA	Totally integrated Automation
PLCSim	Programmable Logic Controller Simulator
Factory I/O	Factory input/output

Table of Contents

GENERAL INTRODUCTION

This technical report aims to emphasize the imperative nature of studying Industry 4.0 through the implementation of a production line simulation utilizing TIA Portal, PLCsim, and Factory I/O. By delving into the intricacies of this simulation, our objective is to comprehensively understand the transformative potential of Industry 4.0 and the specific objectives it aims to accomplish. The primary goal of this study is to demonstrate the tangible advantages that result from the application of Industry 4.0 principles within a production line environment. Through the utilization of TIA Portal, a comprehensive engineering framework for automation, we can effectively design and control various elements of the production line. This encompasses a wide range of components, including sensors, actuators, programmable logic controllers (PLCs), and human-machine interfaces (HMIs). By employing PLCsim, a simulation software, we can accurately emulate the behaviour of a real PLC, thus enabling us to validate and optimize our control strategies. Furthermore, the integration of Factory I/O provides a virtual environment that allows for intuitive visualization and interaction with the simulated production line. Within our sorting factory, numerous automation systems based on wired logic technology are implemented, where the electrical control and monitoring of material sorting processes are overseen by a programmable logic controller (PLC) programmed using LADDER language. The overarching objective of this project is to establish a transportation and sorting system using the SIEMENS S7-1200 industrial programmable logic controller. This proposed solution aims to ensure a seamless and secure sorting and transportation process, transforming raw and unprocessed materials into fully crafted and ready-to-use products. To present a comprehensive overview, this report is divided into three distinct chapters. In the initial chapter, we will introduce fundamental information and provide a detailed description of the hardware employed in our project. The second chapter is devoted to an exploration of the various software utilized to create a virtual factory, replicating a genuine production environment. Finally, in the third chapter, we will guide readers through each step of the process, illustrating both the corresponding blocks within the TIA Portal and the specific control mechanisms they govern in Factory I/O. By undertaking this study and providing a thorough analysis of the simulation process and its implications, we aim to contribute valuable insights into the practical implementation of Industry 4.0 principles within a production line context.

Chapter I

Hardware part

I.1. Introduction:

This chapter is meant to give basic information on automation and the basic elements used on it. We will deal with functions and configurations of PLCs, with emphasis on the S7-200 PLC family and state its advantages, after that we will states the rest of hardware used in the factory.

I.2. Control system:

A Control System is a system, which provides the desired response by controlling the output. The following figure shows the simple block diagram of a control system.

Fig I.1 Control System.

Here, the control system is represented by a single block. Since the output is controlled by varying input, the control system got this name. We will vary this input with some mechanism. In the next section on open loop and closed loop control systems, we will study in detail about the blocks inside the control system and how to vary this input in order to get the desired response.

I.3. Manual and automatic control systems:

I.3.1. Manual Control systems:

A control system that needs a human operator is said to be a manual control system, in terms of engineering we can define it as an open loop system in a way that errors are recorded and corrected by hand in order to achieve desired output.

I.3.1.1 Advantages of manual control systems:

- Low cost (economical).
- Easy in both learning and maintenance points of view.
- Convenient when the output is difficult to be measured.

I.3.1.2 Disadvantages of manual control systems:

- Hard to maintain quality and accuracy.
- Unreliable and less efficient.
- Time consuming.
- Cannot sense environmental changes.

I.3.2. Automatic Control systems:

A control system that behaves the way we want using other components around it with minimum or without human intervention is said to be an automatic control system (self-operating system).

In terms of engineering we can define it as a closed loop system that is a combination of both hardware and software which is designed and programmed to operate automatically,hence have the ability to initiate, adjust , action, measure the variable in the process and stop the process in order to obtain the desired output.

It is generally built up using interconnection of subsystems and processes forming a configuration that uses the difference between the desired output (reference input) and a feedback of the actual output to control the system.

I.3.2.1 Advantages of automatic control systems:

- High accuracy and easy to maintain quality.
- High reliability and high accuracy.
- Safety as it replaces humans in tasks done in dangerous environments (fire, space, nuclear facilities, underwater etc…).
- Reduce operation time and work handling time significantly.

- Performing tasks that are beyond human capabilities of size, weight, speed, endurance, etc…
- Reduce workload especially in tasks that involve physical work.

I.3.2.2 Disadvantages of automatic control systems:

- Unemployment rate increases due to machines replacing humans and putting them out of their jobs.
- Installation and maintenance costs are expensive.
- Pollution rises as most automated machines exhaust gasses and chemical residue.

I.4 PLC (Programmable Logic Controller):

Programmable Logic Controller (PLC) is basically a special computer that is generally used in various industrial applications.PLCs simply process inputs and other variables then control outputs based on a set of stored instructions (program), these controlled outputs are used to automate a process or a machine. **[1]**

I.4.1 History of PLC:

PLC originated in the late 1960s in the automotive industry in the US and were designed to replace relay logic systems. Before, control logic for manufacturing was mainly composed of relays, cam timers, drum sequencers, and dedicated closed-loop controllers.

The PLC was developed with several advantages over earlier designs. It tolerated the industrial environment better than computers and was more reliable, compact and required less maintenance than relay systems. It was easily extensible with additional I/O modules, while relay systems required complicated hardware changes in case of reconfiguration. This allowed for easier iteration over manufacturing process design. Compared to a computer, PLC in a rack format can be more easily extended with additional I/O in the form of I/O cards. With a simple programming language focused on logic and switching operations it was more user-friendly than computers using general purpose programming languages. It is also permitted for its operation to be monitored**.[2]**

Early PLCs were programmed in ladder logic, which strongly resembled a schematic diagram of relay logic. This program notation was chosen to reduce training demands for the existing technicians. Other PLCs used a form of instruction list programming, based on a stack-based logic solver.

I.4.2 How does the PLC works?

Most PLCs consist of the following main components:

- **CPU:** is the unit containing the microprocessor or microcontroller.
- **Power supply uni**t: is needed to convert the main AC voltage to low DC voltage. Commonly it generates 24V DC.
- **Memory unit:** there are two fundamental types of memory used in some PLCs,program and data memory. Memory is organized into blocks of up to say n elements in an array called a file .The program file holds programs, such as ladder logic.
- **Input-Output sections:** are where the processor receives information from external devices (Input section) and communicates information to external devices (Output section). I/O can be digital or analog.
- **Programming device:** is used to upload or download programs from or to PLC’s memory.

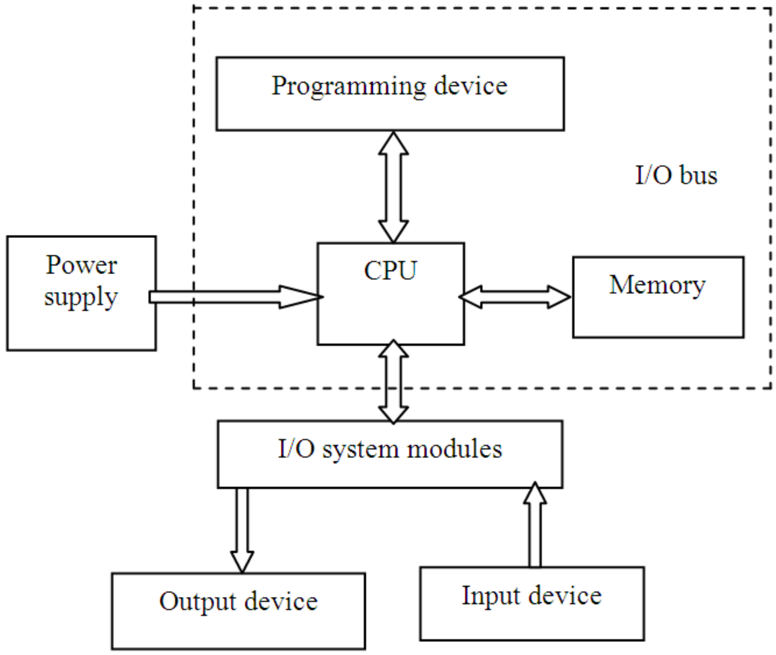


Fig I.2 The architecture of a PLC.

I.4.3 PLC Scan:

The PLC program is executed in a repetitive process called a scan. At the beginning of a scan PLC reads all input status.The application program uses these status informations and gets executed. Once the program is completed,the CPU performs internal diagnostics and related communication tasks. The scan cycle ends by updating the output, then repeats again.The cycle time depends on the size of the program,the number of I/O,speed of PLC and the amount of communication required. **[3]**

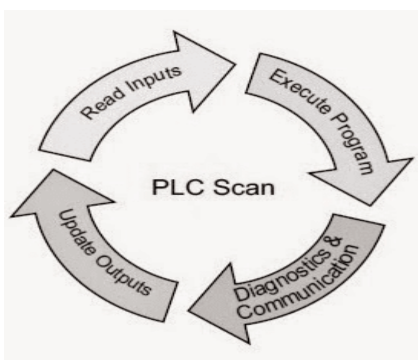


Fig I.3 PLC Scan process.

I.4.4 Types of PLCs:

I.4.4.1 Types of PLC based on hardware setup:

- **Compact PLC:**

Compact PLC also called BRICK PLC or FIXED I/O PLC, it comes with all in package meaning that it has an inbuilt power supply, processor(CPU), communication port to upload/download program and limited number of input and output integrated which their capabilities are decided by the manufacturer and not the user.

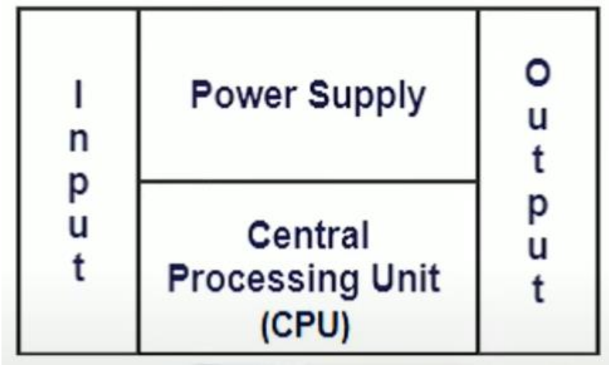


Fig I.4 Compact PLC block diagram.

- **Modular PLC:**

Modular PLC is a rack-based control system with individual power supply, processor, various I/O modules, and extension unit mounted on one or more racks.

It allows the user to configure and reconfigure their controller depending on the number I/O needed, and they are usually easier to use because each component is independent of each other.

Modular PLCs are easy to maintain and repair and has more memory and capabilities to store data compared to compact PLCs, hence it makes them attractive for high-growth companies planning to expand quickly into new markets. **[4]**

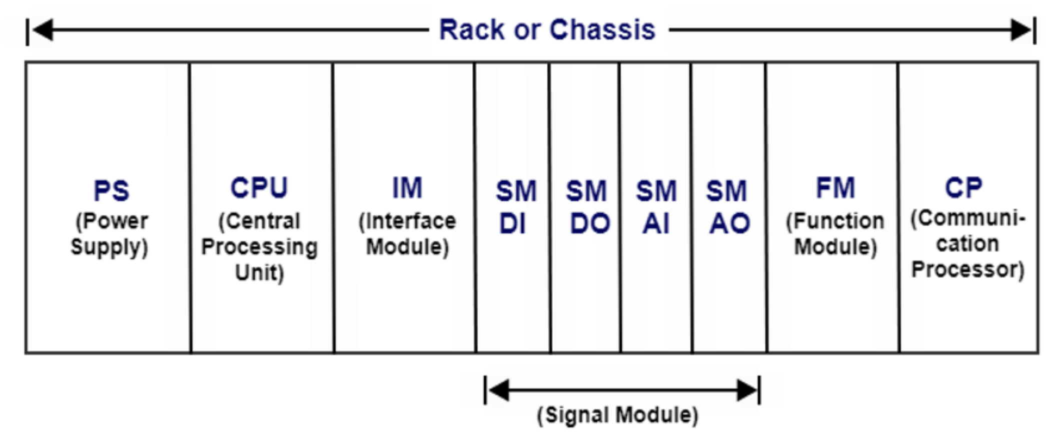


Fig I.5 Modular PLC block diagram.

I.4.3.2 Types of PLC based on size:

- **Micro PLC:**
  - Capable of providing simple level machine control.
  - Limited numbers of Input and output.
  - Can control up to 32 I/O devices.
  - Very compact in size.
  - Fix I/O, no I/O Expandability.
  - Processor, power supply, and inbuilt I/O in a single package.
  - Examples: Siemens S-200, Omron CPM1A, Micro Logix 100.

- **Small PLC:**
  - Capable of providing simple to advanced level machine control.
  - Can be compact or Modular in design.
  - More I/O handling than micro PLC.
  - Can control 32 to 128 I/O devices.
  - Have memory up to 2 Kbyte.
  - Max 1 or 2 modules can be expanded in small PLC.
  - Examples: Siemens Logo, Omron CJ1.

- **Medium PLC:**
  - Capable of providing complex advanced machine control.
  - Robust and modular design.
  - Additional communication modules can be added.
  - Can Control 128 to 2048 I/O devices.
  - Have memory up to 32 KByte which is more expandable.
  - Good I/O expandability.

- **Large PLC:**
  - Used in supervisory control and data acquisition system (SCADAS).
  - Used where complex process control functions are required.
  - Can control entire larger plants or distributed control systems.
  - Can control 2048 to 8192 I/O devices.
  - Most sophisticated in the PLC family.
  - The capacities are quite higher than medium PLCs.
  - More memory and multiple programming languages.
  - Support multiple communication modules.

I.5 Programming languages of a PLC:

Programs used with PLCs can be written in different formats. To enable engineers with little programming knowledge to write programs for PLCs, the contact language has been designed. Most PLC manufacturers have adopted this method of writing programs.

However, since everyone has tended to develop their own versions, an international standard has been established for the contact language and, consequently, for all programming methods used with the PLCs **.**

The IEC 1131-3 standard defines five languages that can be used for programming PLCs. These five languages are:

1- Ladder Diagram (LD) :

It is the most popular PLC language. It is a graphical PLC programming language which expresses logic operations with symbolic notation using ladder diagrams, Which is commonly referred to as ladder logic.

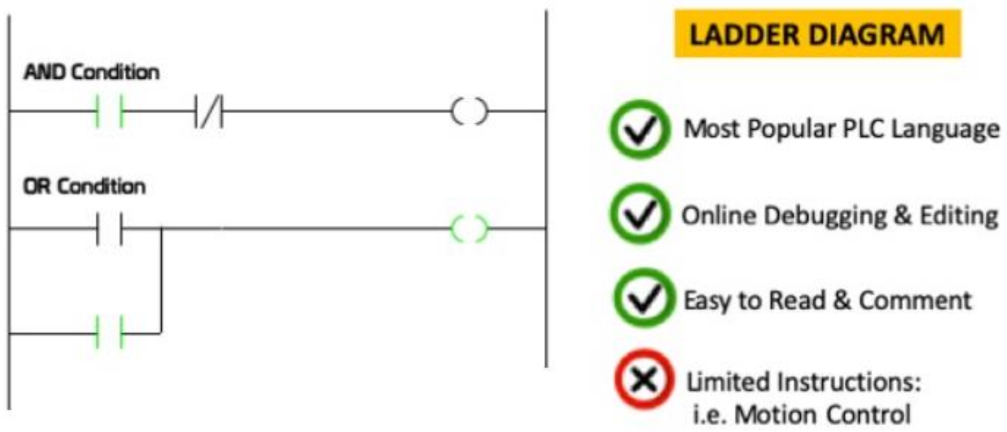


Fig I.6 Modular PLC block diagram

2- Function block diagram :

The second most popular PLC programming language is called Function Block Diagram (FBD). In the Function Block Diagram, program blocks are connected together to create a program. A lot of the same commands used in Ladder Logic are used in FBD, but it is often easier to read and conceptualize. An added benefit to using Function Block Diagram is that you can consolidate many lines of code into a single block.

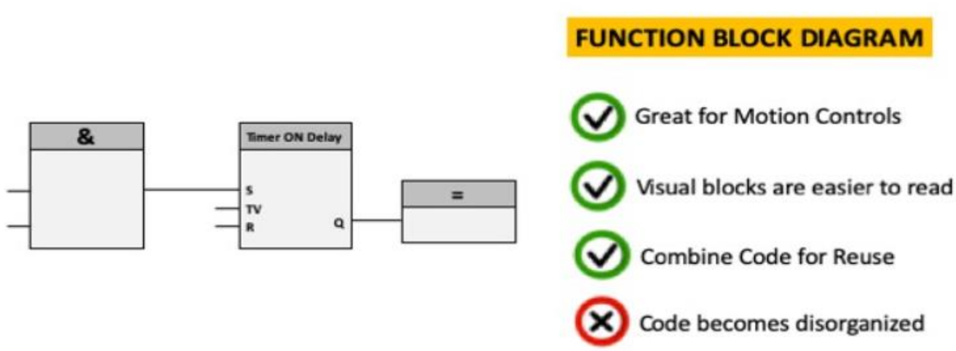


Fig I.7 Modular PLC block diagram

3- Structured Text (ST):

Third, on the list of PLC programming languages is Structure Text (ST). Structured Text looks very similar to BASIC or C programming. It's best used for control systems that require mathematics or complex tasks. Trigonometry, calculus, and data analysis can be implemented far easier in this language than in Ladder Diagrams. Because ST isn't a graphical language, it also runs faster and requires less space.

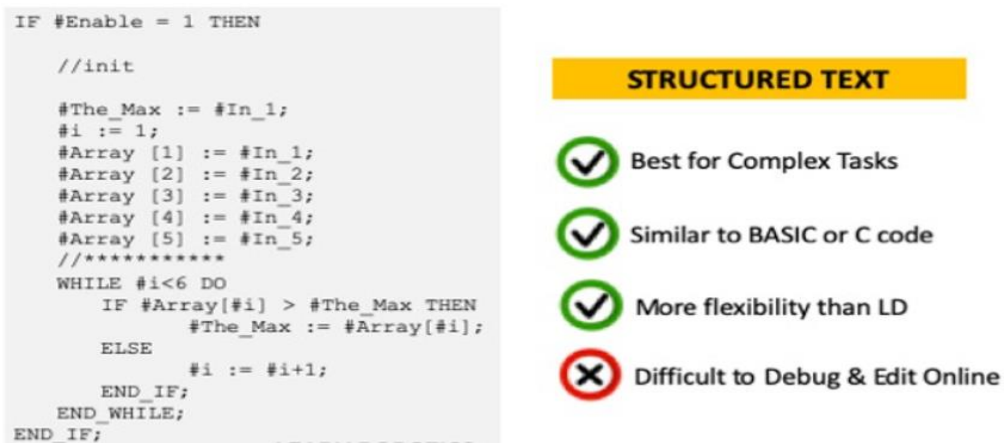


Fig I.8 Structured text.

4- Instruction List (IL):

Next is the Instruction List (IL). This programming language consists of many lines of code, with a single instruction per line. It's read top to bottom and left to right. The Instruction List is very straight-forward to read because each line is executed sequentially. Once you learn the mnemonic codes (Load = LD, Start = ST, etc.), Instruction List is a great language because we can create compact and time-critical code as required by our application.

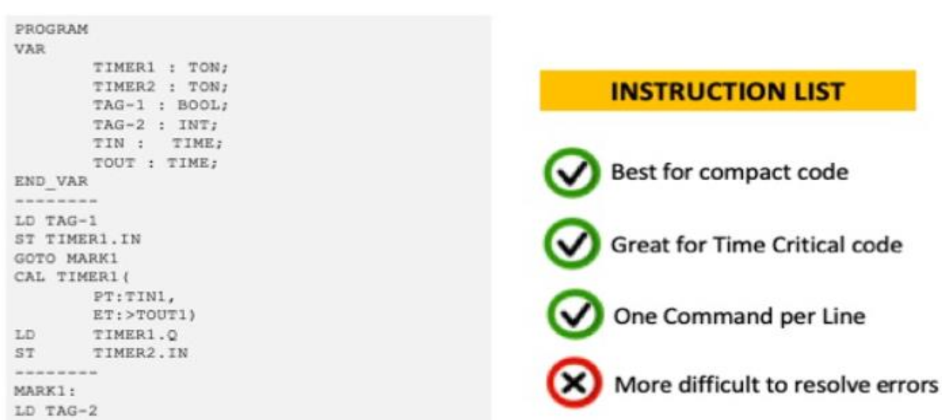


Fig I.9 Instruction List.

5- Sequential function chart (SFC):

Lastly is Sequential Function Chart (SFC). The concept of SFC is simple: an action box is active until the transition step below it activates. The transition step contains all of the conditions that must be met in order for the next box to activate. if the project has repeatable steps that can be broken into smaller tasks, then SFC is the easiest of the languages to implement.

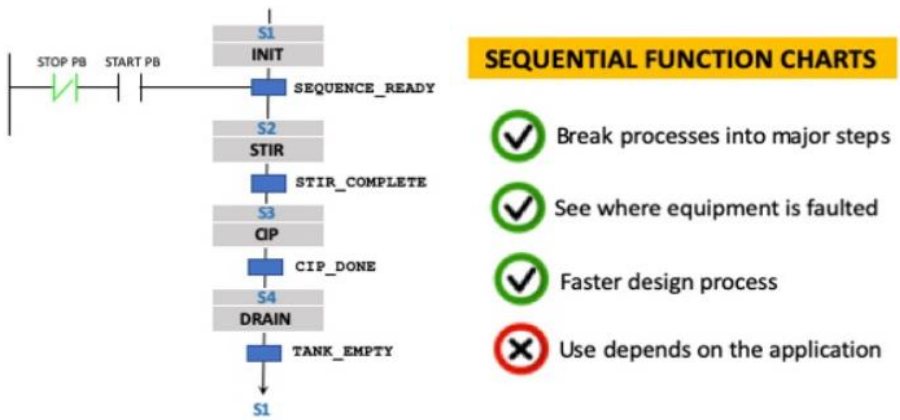


Fig I.10 Sequential function.

I.6 Siemens PLCs:

Siemens makes several PLC product lines in the SIMATIC S7 family. They are: S7-200, S7-300, S7-400, S7-1200, S7-1500. Siemens Step 7 is a powerful integrated software solution for automation, and includes the programming environment for Siemens programmable logic controllers (PLCs). This software provides unique and powerful programming tools with multiple benefits including: creation of reusable logic, a structured program architecture, and single project integration of multiple automation devices.

I.6.1 Siemens Modular PLCs:

Siemens SIMATIC PLCs are the basis upon which our Totally Integrated Automation (TIA) concept is based. The SIMATIC PLCs are available as conventional modular controllers.

Fig I.11 S7-200

Fig I.12 S7-1200

Fig I.13 S7-300

Fig I.14 S7-1500

I.6.2 Siemens S7-1200 PLC:

I.6.2.1 Presentation of S7-1200:

SIMATIC S7-1200 controllers are the ideal choice when it comes to flexibly and efficiently performing automation tasks in the lower to medium performance range. They feature a comprehensive range of technological functions and integrated communication as well as especially compact and space-saving design.

The CPU combines the following elements and more in a compact housing to create a powerful controller:

- A microprocessor.
- An integrated power supply.
- Input and output circuits.
- Built-in PROFINET (can communicate with HMI panels or another CPU).
- High-speed motion control I/O.

After you download your program, the CPU contains the logic required to monitor and control the devices in your application. The CPU monitors the inputs and changes the outputs according to the logic of your user program, which can include Boolean logic, counting, timing, complex math operations, motion control, and communications with other intelligent devices.

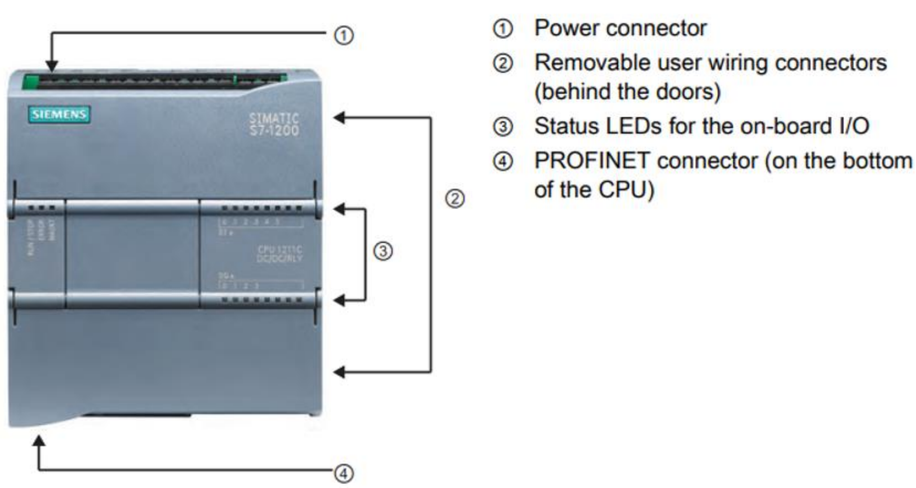


Fig I.15 S7-1200 diagram.

I.6.2.2 Siemens S7-1200 Modules:

1. Communication modules (up to 3).
2. CPU Slot 1.
3. CPU E-net Port.
4. Signal board,communication or battery board.
5. Up to 8 analog/digital expansion modules.

Fig I.16 S7-1200 modules.

I.6.2.3 Siemens S7-1200 Network:

The way we connect our CPU to other devices is via Ethernet port, we can connect it to a switch that has a HMI, a PC that we will use to program the CPU.

I.6.2.3 Siemens S7-1200 Wiring:

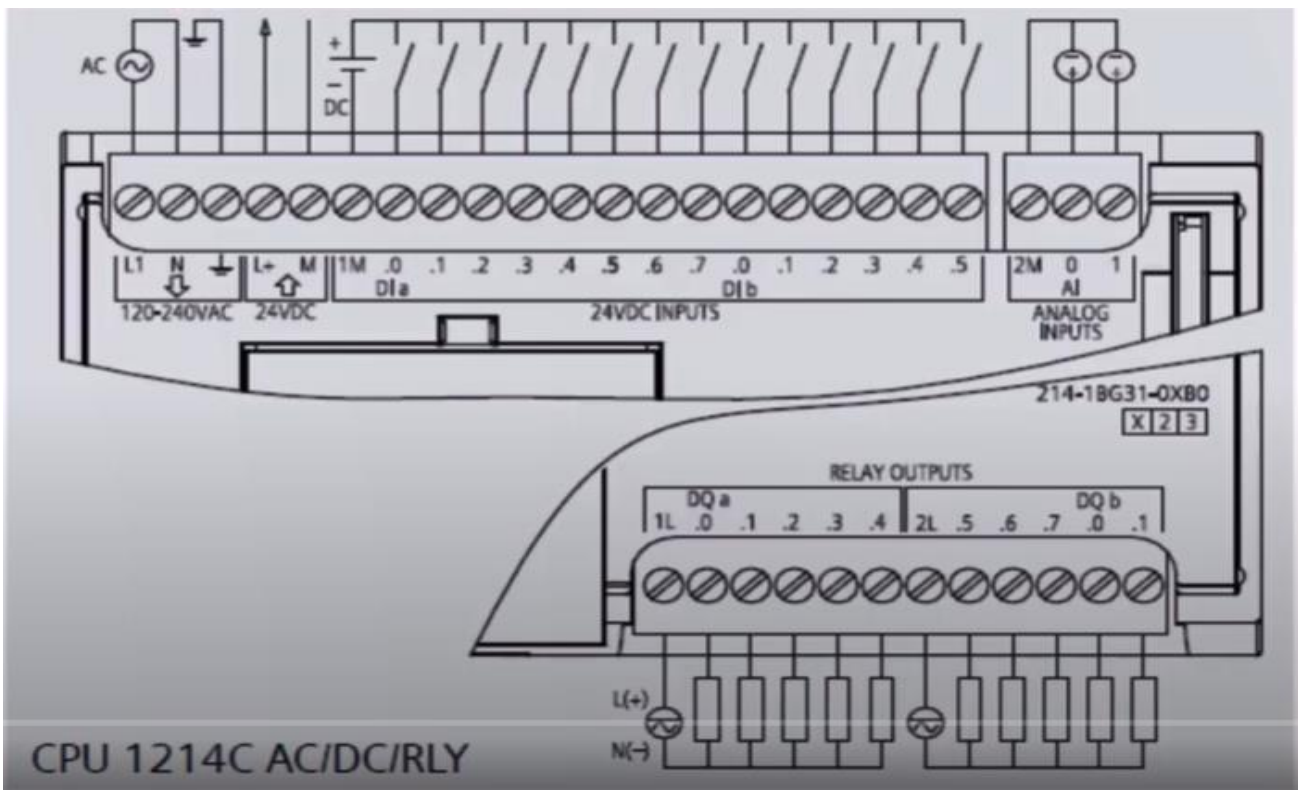


Fig I.17 S7-1200 CPU 1214C AC/DC/RLY Wiring diagram.

- **The compact CPU 1214C has:**
  - 3 device versions with different power supply and control voltages (AC/DC/RLY, DC/DC/DC ,DC/DC RLY).
  - Integrated power supply either as wide-range AC or DC power supply (85 ... 264 V AC or 24 V DC).
  - Integrated 24 V encoder/load current supply for direct connection of sensors and encoders. With a 400 mA output current, it can also be used as a load power supply.
  - 14 integrated digital inputs 24 V DC (current sinking/sourcing input (IEC type 1 current sinking)).
  - 10 integrated digital outputs, either 24 V DC or relay.
  - 2 integrated analog inputs 0 ... 10 V.
  - 2 pulse outputs (PTO) with a frequency of up to 100 kHz.
  - Pulse-width modulated outputs (PWM) with a frequency of up to 100 kHz.
  - Integrated Ethernet interface (TCP/IP native, ISO-on-TCP).
  - 6 fast counters (3 with max. 100 kHz; 3 with max. 30 kHz), with parameterizable enable and reset inputs, can be used simultaneously as up and down counters with 2 separate inputs or for connecting incremental encoders.
  - Expansion by additional communication interfaces, e.g. RS485 or RS232.
  - Expansion by analog or digital signals directly on the CPU via signal board (with retention of CPU mounting dimensions).
  - Expansion by a wide range of analog and digital input and output signals via signal modules.
  - Optional memory expansion (SIMATIC Memory Card).
  - PID controller with auto-tuning functionality.
  - Integral real-time clock.
  - Interrupt inputs for extremely fast response to rising or falling edges of process signals.

I.6.2.4 Siemens S7-1200 Functions:

- **Comprehensive instruction set:**

  - A wide range of operations facilitates programming.

  - Basic operations such as binary logic operations, result allocation, save, count, create times, load, transfer, compare, shift, rotate, create complement, call subprogram (with local variables).

  - Integral communication commands (e.g. USS protocol, Modbus RTU, S7 communication "T-Send/T-Receive" or Freeport) -
  User-friendly functions such as pulse-width modulation, pulse sequence function, arithmetic functions, floating point arithmetic, PID closed-loop control, jump functions, loop functions and code conversions.

  - Mathematical functions, e.g. SIN, COS, TAN, LN, EXP.

- **Counting:**

  User-friendly counting functions in conjunction with the integrated counters and special commands for high-speed counters open up new application areas for the user.

- **Interrupt processing:**

  - Edge-triggered interrupts (activated by rising or falling edges of process signals on interrupt inputs) support a rapid response to process events.

  - Time-triggered interrupts.

  - Counter interrupts can be triggered when a setpoint is reached or when the direction of counting changes.

  - Communication interrupts allow the rapid and easy exchange of information with peripheral devices such as printers or bar code readers.

- **Password protection.**

- **Test and diagnostics functions:**

  Easy-to-use functions support testing and diagnostics, e.g. online/offline diagnostics.

- **"Forcing" of inputs and outputs during testing and diagnostics**

  Inputs and outputs can be set independently of cycle and thus permanently, for example, to test the user program.

I.6.2.4 Operating modes of the CPU:

The CPU has the following operating modes:

“STARTUP” mode, the CPU starts a startup procedure

“STOP” mode, the CPU does not execute the program, and a project can be loaded.

“RUN” mode, the program is executed cyclically, projects cannot be loaded into the CPU in RUN mode (note for all PLCs).

I.7 Hardware of the factory:

I.7.1 Sensors:

I.7.1.1 Definition:

A sensor is a device that detects and responds to changes or stimuli in the physical or environmental conditions. It converts the measured parameter into an electrical or digital signal that can be analyzed, processed, or used for control purposes. Sensors can measure various physical quantities such as temperature, pressure, light, sound, and more. They can be based on different principles and provide analog or digital output signals. Sensors are used in industries such as automation, automotive, healthcare, and consumer electronics to collect data, enable control, and integrate physical systems with digital technologies. **[5]**

I.7.1.2 Types Of sensors:

There are a variety of sensors that are commonly used in automation according to their use such as Proximity sensors, Color sensors, Force and Load sensors. Temperature sensors, Pressure sensors, Flow sensors, Position sensors, Motion sensors etc….

These sensors play crucial roles in monitoring and controlling various aspects of automated systems, enabling efficient and precise operation.

In this section we will define only the first three sensors that we have used in our project.

- **Proximity sensors :**

Proximity sensors are electronic devices that detect the presence or absence of objects within a specified range without making physical contact. They are commonly used in industrial automation, robotics, security systems, and various other applications where reliable object detection is required.

They operate based on different principles such as electromagnetic fields, capacitance changes, light beams, or ultrasonic waves.

- Inductive Proximity Sensors: Detect metallic objects by measuring changes in electromagnetic fields.

- Capacitive Proximity Sensors: Detect objects by changes in capacitance when an object enters the sensor's electric field.

- Photoelectric Proximity Sensors: Use light beams to detect objects by measuring changes in light intensity.

- Ultrasonic Proximity Sensors: Emit and receive sound waves to determine object presence or distance.

Proximity sensors have a specified sensing range, ranging from millimeters to meters. They provide digital output signals indicating the presence or absence of an object. Sensors can be mounted in different ways and come in various physical forms.

- **Color Sensors :**

Color sensors are electronic devices used to detect and analyze colors. They are designed to measure the intensity of light across different wavelengths within the visible spectrum and provide information about the color of an object or surface. Color sensors utilize various technologies, such as light-sensitive diodes, photodiodes, or color filters, to differentiate between different colors and provide output signals that can be interpreted by a system or user. These sensors find applications in color detection, sorting, quality control, color measurement, and automation processes in industries such as manufacturing, printing, textiles, food processing, and more. They play a crucial role in ensuring accurate color recognition, consistency, and control in a wide range of products and systems.

- **Force and Load Sensors:**

Force and load sensors, also known as force transducers or load cells, are electronic devices that measure and detect the force or load applied to an object. These sensors convert mechanical force into an electrical signal that can be measured and analyzed. They are widely used in industries and applications where precise measurement of forces and loads is necessary.

Force and load sensors typically consist of a sensing element, such as strain gauges, that deforms under the applied force. This deformation generates a change in electrical resistance, voltage, or current, which is then converted into a measurable output signal. The magnitude of the force or load can be determined by analyzing the output signal.

These sensors are utilized in various fields, including industrial automation, robotics, automotive, aerospace, materials testing, and weighing applications. They enable accurate force measurement, load monitoring, tension control, and feedback in mechanical systems. Force and load sensors play a crucial role in ensuring safety, quality, and efficiency in processes where forces and loads need to be precisely monitored and controlled.

I.7.2 Factory convoys:

Factory convoys refers to the organized movement of materials, components, or products within a factory or industrial facility. It involves the transportation of these items from one area or station to another using a group of vehicles or equipment. **[6]**

In a factory setting, the use of convoys aims to improve the efficiency of material handling and streamline the flow of goods. It helps ensure that materials are delivered to the right location at the right time, minimizing delays and optimizing production processes.

Factory convoys often involve the use of carts, forklifts, automated guided vehicles (AGVs), or conveyor systems to transport the materials. These convoys may follow predetermined routes or schedules, enabling systematic movement between different production areas, assembly lines, storage areas, or shipping/receiving zones.

By implementing factory convoys, companies can enhance operational efficiency, reduce manual handling, minimize bottlenecks, and maintain a smooth workflow. It allows for better coordination and organization of material movement, ultimately contributing to improved productivity and overall manufacturing effectiveness.

A wheel sorting machine, also known as a rotary sorting machine or wheel diverter, is a mechanical device used to sort different materials or items into separate paths or destinations based on specific criteria. It is commonly employed in industries such as logistics, packaging, manufacturing, and material handling.

I.7.3 The wheel sorting machine:

The wheel sorting machine consists of a rotating wheel or disc with multiple pockets or compartments. Each pocket is designed to hold and guide an individual item or material. As the wheel rotates, the items are transported along the wheel's periphery, and at specific points, they are diverted into different paths or chutes based on predetermined sorting criteria. **[7]**

The sorting criteria can vary depending on the application. It could be based on attributes such as size, shape, weight, color, barcode, or other identifying features of the materials being sorted. The wheel sorting machine is equipped with sensors or detection systems to identify these attributes and determine the appropriate path for each item.

The diverted paths lead the sorted items to designated collection bins, conveyor belts, or other handling systems for further processing, packaging, or distribution. The speed and accuracy of the wheel sorting machine contribute to efficient material flow and sorting operations in industrial settings.

Overall, the wheel sorting machine automates the process of segregating different materials into distinct paths, enabling efficient sorting, distribution, and downstream processing based on specific criteria or characteristics of the items being sorted.

I.8 Conclusion:

In this chapter we have explained briefly the essential definitions of automations, types of PLC and the differents aspects of Siemens PLC and the hardware that we have used in this project, which will allows the reader to understand the concept of work that will be shown in the next chapter.

Chapter II

Software part

II.1. Introduction:

The simulation tools are a critical way to test your project before going to the field, so many manufacturers started to focus on this side, now you can educate your project minimizing the loss of money for experiments. Siemens and factory I/O made a great combination for such purpose.

II.2. TIA Portal (Totally Integrated Automation):

II.2.1 Introduction:

In a world that requires optimization Siemens came up with all in one software called TIA Portal.

The new software engineering frame enables users to develop and commission automation systems rapidly and intimately, which eliminates the traditional time-consuming and expensive integration of separate software packages. Designed for high effectiveness and user-friendliness, the TIA Portal is suitable for both first-time users and experienced users.

The TIA Portal provides a single interface for all Siemens automation products. This makes it easy to work with different types of devices, including programmable logic controllers (PLCs), human-machine interfaces (HMIs), drives, and motion control systems.

II.2.2 Description:

Siemens TIA Portal is a software program that provides users with the tools to configure and program Siemens controllers. The software offers a wide range of features, including programming support for Ladder Logic, ST (Structured Text), and FBD (Function Block Diagram). It also supports various communication protocols such as Profibus and Ethernet. **[8]**

II.2.2 Advantages:

The advantages of TIA Portal are:

- Intuitive and fast programming: with newly developed programming editors LD, FBD, ST, IL and SFC.
- Increased efficiency thanks to STEP 7's linguistic innovations: uniform symbolic programming, addition of blocks during operation and much more.
- Increased performance thanks to integrated functions: simulation with PLCSIM,

remote maintenance with TeleService and consistent system diagnostics.

- Flexible technology: scalable and efficient motion control functionality for the S7-1200 and S7-1500 PLCs.
- Increased security with Security Integrated: know-how protection, copy protection, access protection and forgery protection.
- Common configuration environment with HMI panels and drives in the TIA Portal engineering environment.

II.2.3 SIMATIC STEP 7:

The STEP 7 Professional software (TIA Portal V15.1) is the programming tool for PLCs SIMATIC S7-1200 and SIMATIC S7-1500.

With STEP 7 Professional V15.1, the following functions can be used to automate a system:

- Hardware configuration and parameterization.
- Communication settings.
- Programming.
- Testing, commissioning and troubleshooting with operating and diagnostic functions.
- Documentation.
- Generation of visualization screens for SIMATIC Basic Panels with integrated WinCC.
- It is also possible to generate display screens for PCs and other Panels using other WinCC software packages.

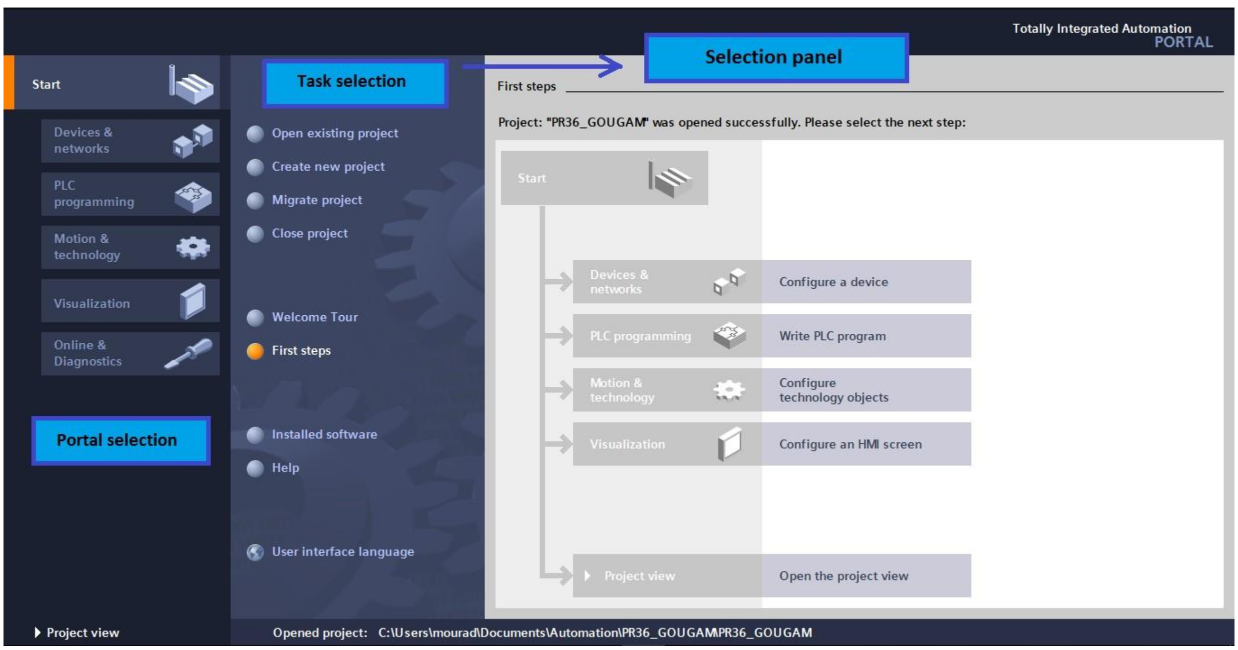
FigII.1 TIA Portal V15.1

II.2.4 Portal view and Project view:

TIA has **two type of View**. Task-oriented set of portals that are organized on the functionality of the tools (Portal view), or a project-oriented view of the elements within the project (Project view).

- **Portal View:**

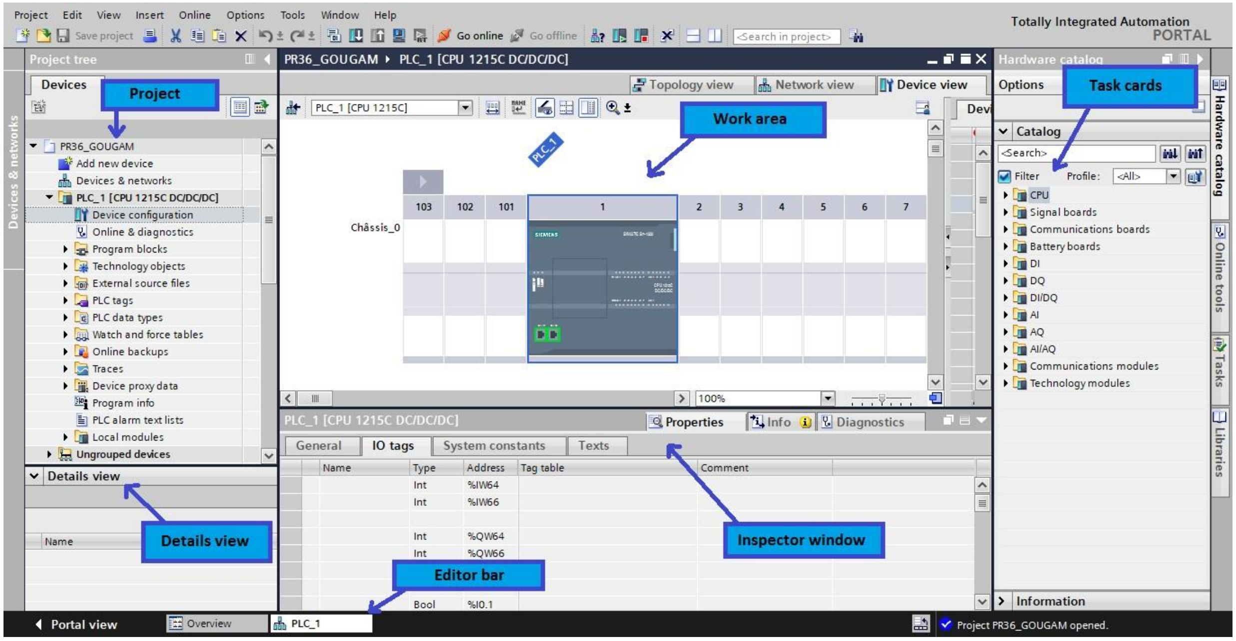
The portal view is a high-level overview of your entire project, including all the devices, networks, and applications that are part of it. This is a great way to get an overview of your project and see how everything fits together.



FigII.2 Portal View of TIA Portal.

- **Project View:**

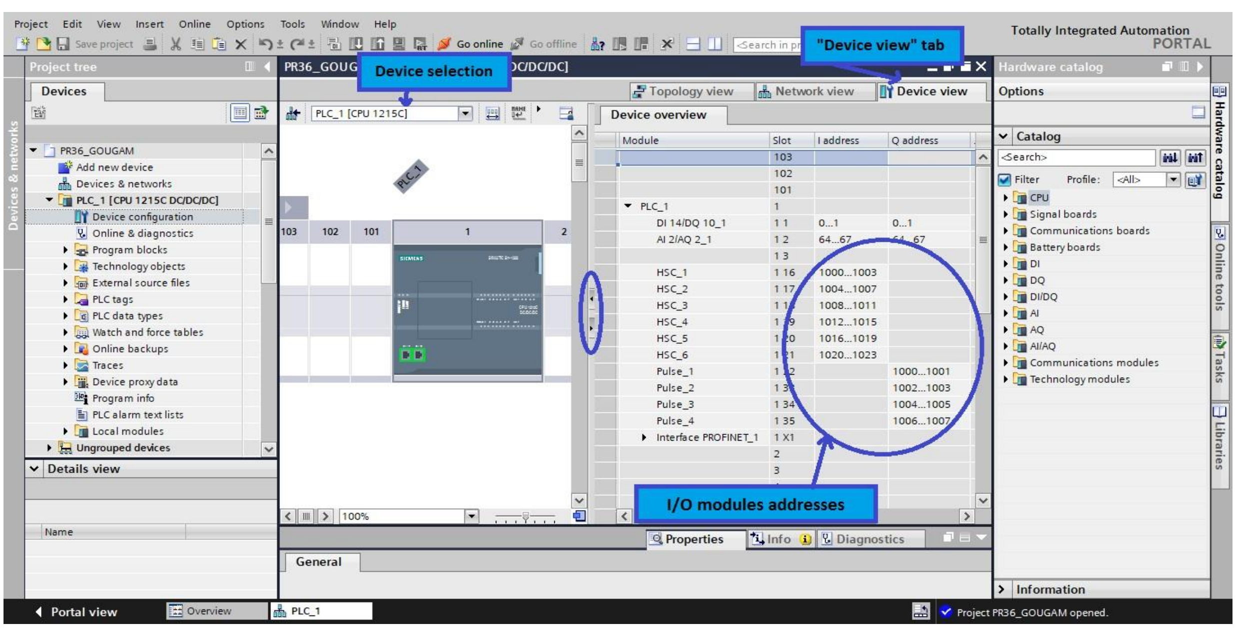
The project view is a more detailed look at each individual component in your project. This is where you will configure the settings for each device or application. You can also test individual components in this view to make sure they are working properly before putting them into production.



FigII.3 Project View of TIA Portal.

II.2.5 I/O addressing:

To know the addressing of the inputs and outputs present in the hardware configuration, go to “Devices & networks” in the project browser. In the work area, make sure you are in the “Device view” tab and select the desired device.



FigII.4 I/O addressing.

Select the CPU and use the two small arrows (see Figure) to open the “Device overview” tab. The addresses of the inputs and outputs appear. It can be changed by entering a new value in the corresponding box.

II.3. PLC SIM:

PLCSIM is a Siemens software that allows users to test programs on a virtual PLC. This is beneficial because it allows for debugging and troubleshooting without having to use physical hardware. Additionally, PLCSIM can be used to create simulations of process plants or machines. **[9]**

The focus of S7-PLCSIM is to support debugging and validating a single PLC program without requiring actual hardware. S7-PLCSIM allows you to use all STEP 7 debugging tools, including the watch table, program status, online & diagnostics functions, and other tools.

S7-PLCSIM also provides tools that are unique to S7-PLCSIM, including a SIM table and sequence editor.

S7-PLCSIM V15.1 operates in conjunction with STEP 7 in the TIA Portal V15.1. You can:

- Configure your PLC and any associated modules in STEP 7.
- Program your application logic.
- Download the hardware configuration and program to S7-PLCSIM.

FigII.5 PLCSIM

II.4. Factory I/O:

Factory I/O is a real-time 3D simulation software for factories, machines and processes. It allows users to create virtual prototypes of their products and systems, test them in a realistic environment and identify potential problems early on.

Factory I/O can be used for training purposes as well – employees can learn how to operate machinery without the risk of injury, and new procedures can be tested before being implemented in the real world. **[10]**

The software is also valuable for marketing purposes – potential customers can get an idea of how a product will work in their own factory, and see if it meets their needs.

The most common scenario is to use Factory I/O as a PLC training platform since PLC are the most common controllers found in industrial applications. However, it can also be used with microcontrollers, SoftPLC, Modbus, among many other technologies.

FigII.6 Factory I/O.

II.5.Software Design Approach:

The software design for simulating the factory involves the following steps:

II.5.1 Designing Control Logic in TIA Portal:

The initial step is to design the control logic using the TIA Portal software. This entails defining the sequence of operations and configuring the necessary input and output devices. TIA Portal offers a visual programming environment that allows users to create a comprehensive control strategy for the automation system. By utilizing the features provided by TIA Portal, such as configuring material tracking, belt conveyors, color sensors, and robotic arms, the control logic is developed to orchestrate the desired actions. **[11]**

II.5.2 Validating Control Logic with PLCSim:

Once the control logic is created in TIA Portal, it is imported into PLCSim for validation and testing purposes. PLCSim acts as a simulator, replicating the behavior of the actual Programmable Logic Controllers (PLCs). This integration allows users to execute the control logic virtually and observe the system's response. Through this validation process, potential issues or errors in the control logic can be identified and rectified before proceeding to the simulation phase.

II.5.3 Creating the Virtual Factory Environment in Factory I/O:

To replicate the real factory environment, Factory I/O is utilized to create a virtual factory setting. This involves modeling the machinery, sensors, and actuators that exist in the physical factory within a 3D environment. Components are placed and configured within Factory I/O to mimic their properties, behavior, and interactions. By leveraging the user-friendly interface of Factory I/O, a virtual automation system can be set up to resemble the actual factory. **[12]**

II.5.4 Integrating TIA Portal, PLCSim, and Factory I/O:

The integration of TIA Portal, PLCSim, and Factory I/O is crucial for achieving a cohesive simulation. TIA Portal is connected to PLCSim to enable the execution of the control logic created within TIA Portal in the virtual environment provided by PLCSim. This allows for the synchronization of the control logic with the simulated PLC behavior. Additionally, Factory I/O interfaces with PLCSim to establish a connection between the virtual factory environment and the simulated control signals. This integration ensures that the virtual components within Factory I/O respond accordingly to the control logic executed through PLCSim.

II.5.5 Simulating the Factory Operations:

Once the integration is established, the simulation is initiated by running the control logic in TIA Portal. This triggers the corresponding actions within PLCSim and the virtual factory environment in Factory I/O. Users can observe and analyze the simulated operations, including the movement and handling of materials, as they unfold within the virtual environment. By executing the control logic and examining the simulated results, users can gain insights into the system's behavior, identify potential improvements, and validate the effectiveness of the control mechanisms.

II.6.Conclusion:

By following this software design approach, the integrated use of PLCSim, TIA Portal, and Factory I/O allows for the development of a comprehensive simulation that replicates factory operations in a virtual environment.

Chapter III

The Simulation of the Factory

III.1. Introduction:

After getting to know the softwares we are going to use to simulate the factory, we need to put those apps together in action. This chapter is for that reason.

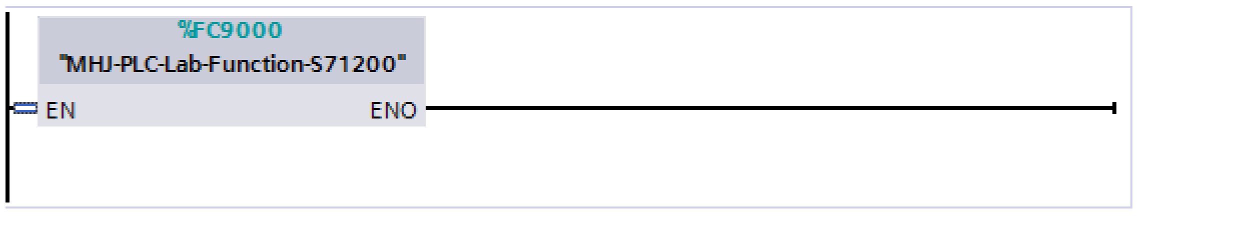
III.2. Factory I/O and Tia portal Programming:

In this part, I will guide through each step in the process showing both the block in the TIA Portal and what that block is controlling in Factory I/O .Starting with a picture of the whole factory:

FigIII.1 Full view of the factory.

- **Network 1:**

The first network contains only one block "MHJ-PLC-Lab-Function-S71200" this block is needed to connect with Factory I/O:



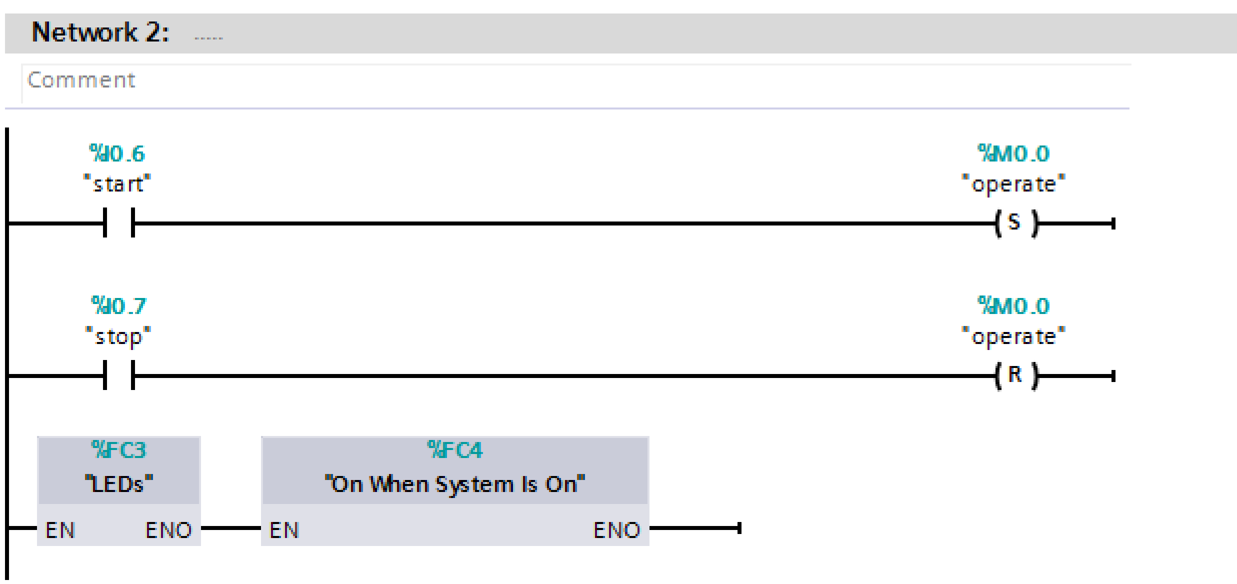
FigIII.2 Network 1

- **Network 2:**

The second network has the start and stop buttons the start button will SET the "operate" state register, and the stop button will RESET it, this "operate" register is need in later steps of the process.

It also has a block "LEDs" this turns ON the start button LED when the factory is operating as seen in **FigIII.4**, and turn ON the stop button LED when factory in not operating as seen in **FigIII.5**.

The last block "On When System Is On" turns ON some actuators when is system is operating and turns them OFF when it is not these elements are the conveyers highlighted in **FigIII.7** and **FigIII.8.**



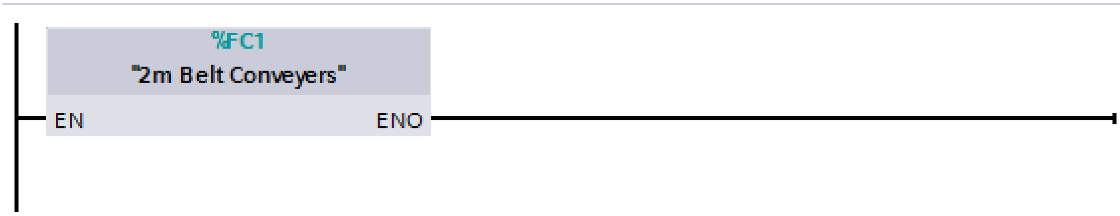
FigIII.3 Network 2

FigIII.4 The state of LEDs when operating

FigIII.5 The state of LEDs when not operating

- **Network 3:**

The third network contains the "2 meters belt conveyers" block these belt conveyers are highlighted in Fig 6 and Fig 7, this block will SET the 2 meters belt conveyers first when the operation starts, to move the raw material inside the Machining Center. Then RESETs the belt conveyers when Machining Center is busy, then SET them again when Machining Center is Not busy.



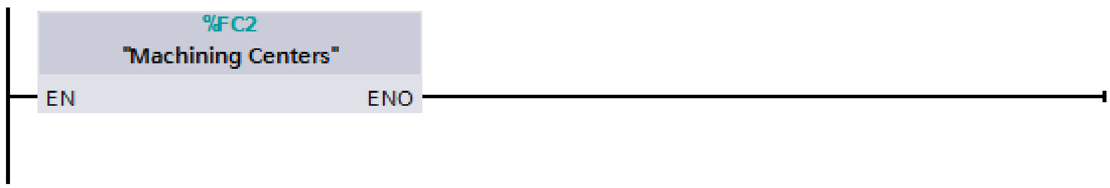
FigIII.6 Network 3

FigIII.7 Right 2-meter belt conveyer.

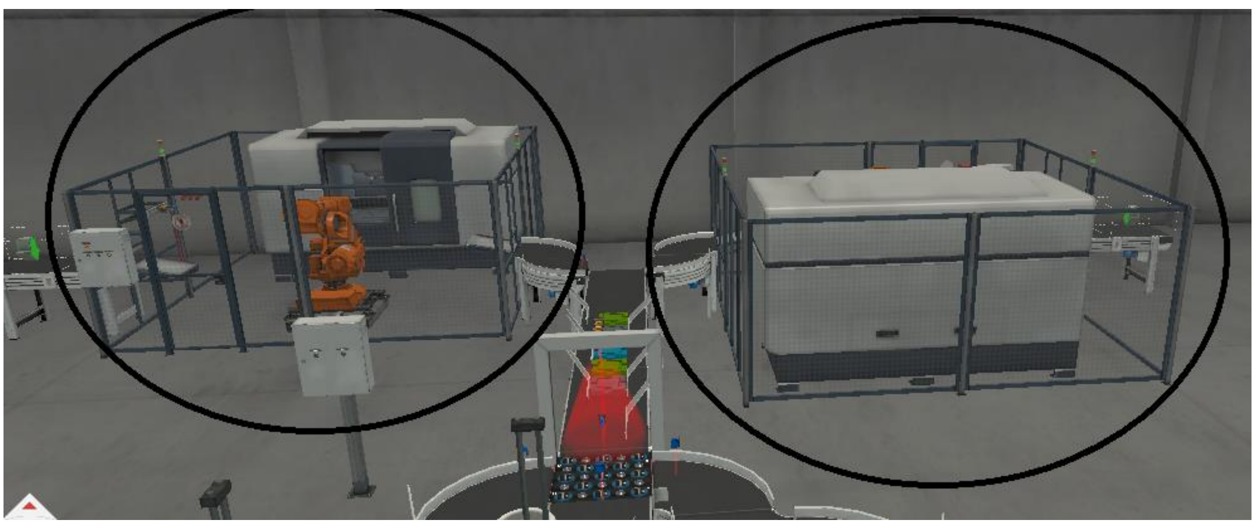
FigIII.8 Left 2-meter belt conveyer.

- **Network 4:**

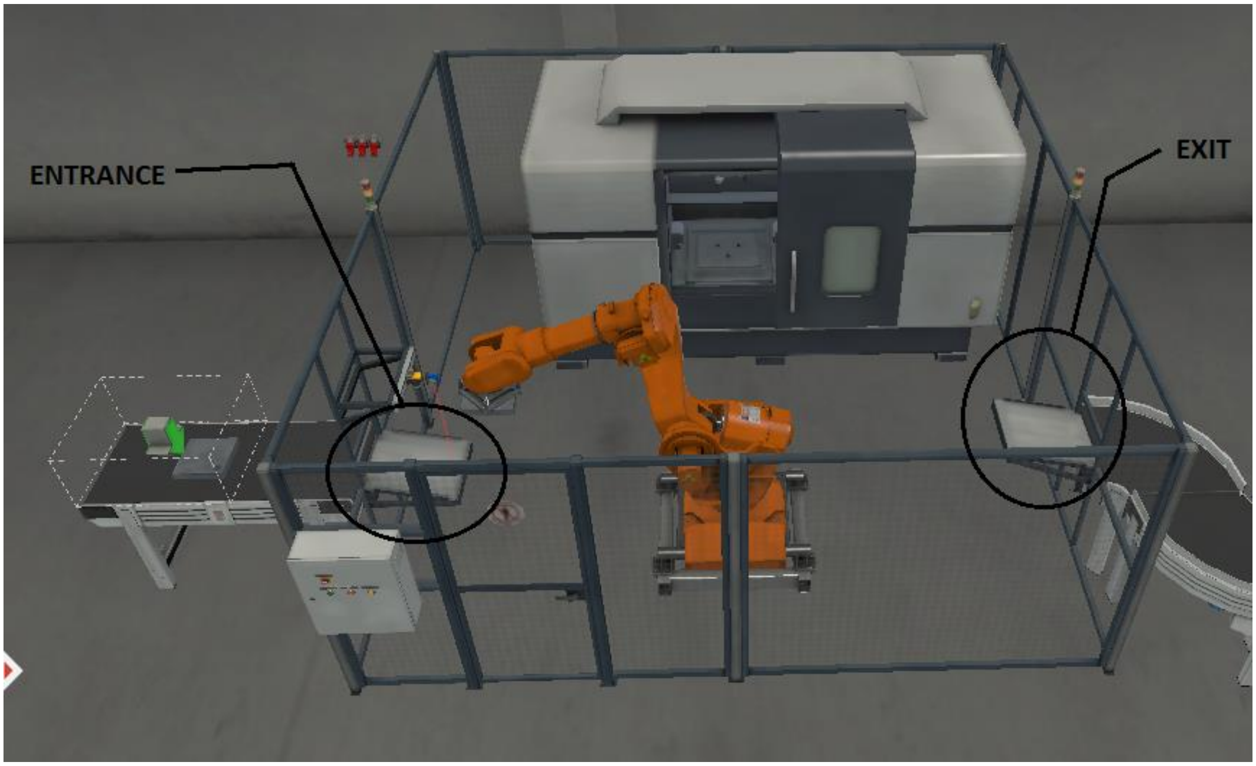
The fourth network contains the “Machining Centers” block as shown in **FigIII.10**, this machine start crafting the raw material when its proximity sensor detects a raw material at the entrance, during that crafting a busy signal is SET indicating that the machine is working. When the crafting is done the arm inside the Machining Center puts the crafted material on the exit.



FigIII.9 Network 4



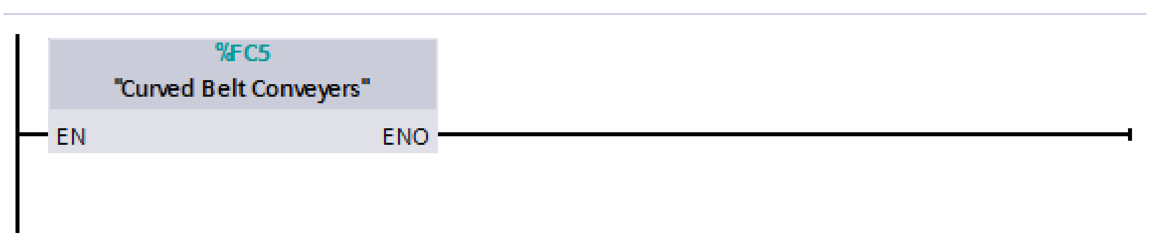
FigIII.10 Machining Center 1 & 2



FigIII.11 Machining Center entrance and exit

- **Network 5:**

The fifth network contains the "Curved Belt Conveyers" block, these Belt Conveyers will be SET when the operation starts and one of these two will RESET when there is a collision between two crafted martials, and will be SET again when the material on the opposite side passes.

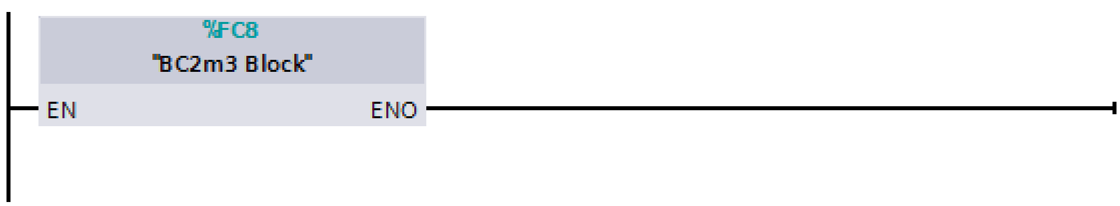


FigIII.12 Network 5

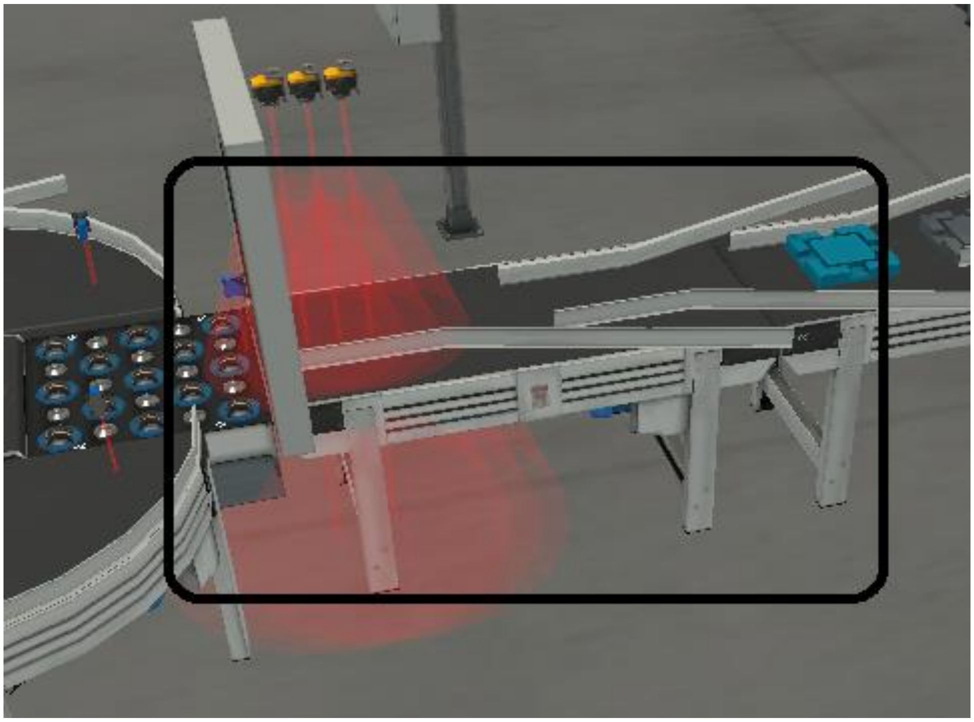
FigIII.13 The curved belt conveyers

- **Network 6:**

The sixth network contains the “2 meters Belt Conveyer” highlighted in **FigIII.15**, this belt conveyer will be SET when operation starts, and RESET when the crafted material enters the Pop Up Wheel Sorter that we will discuss next.



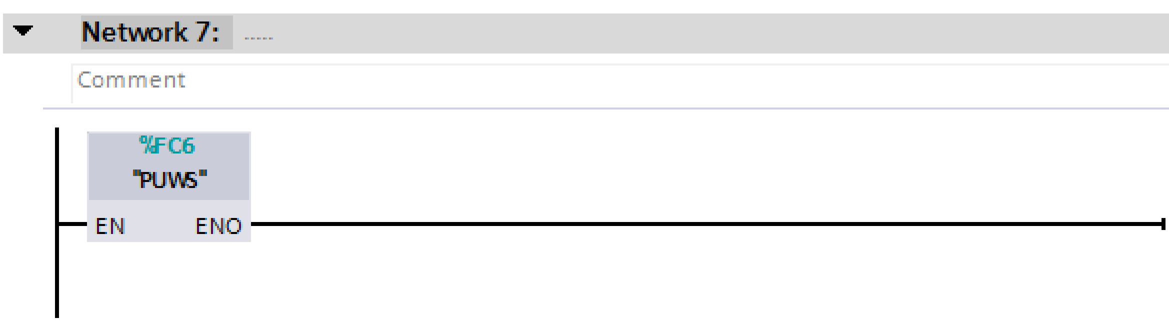
FigIII.14 Network 6



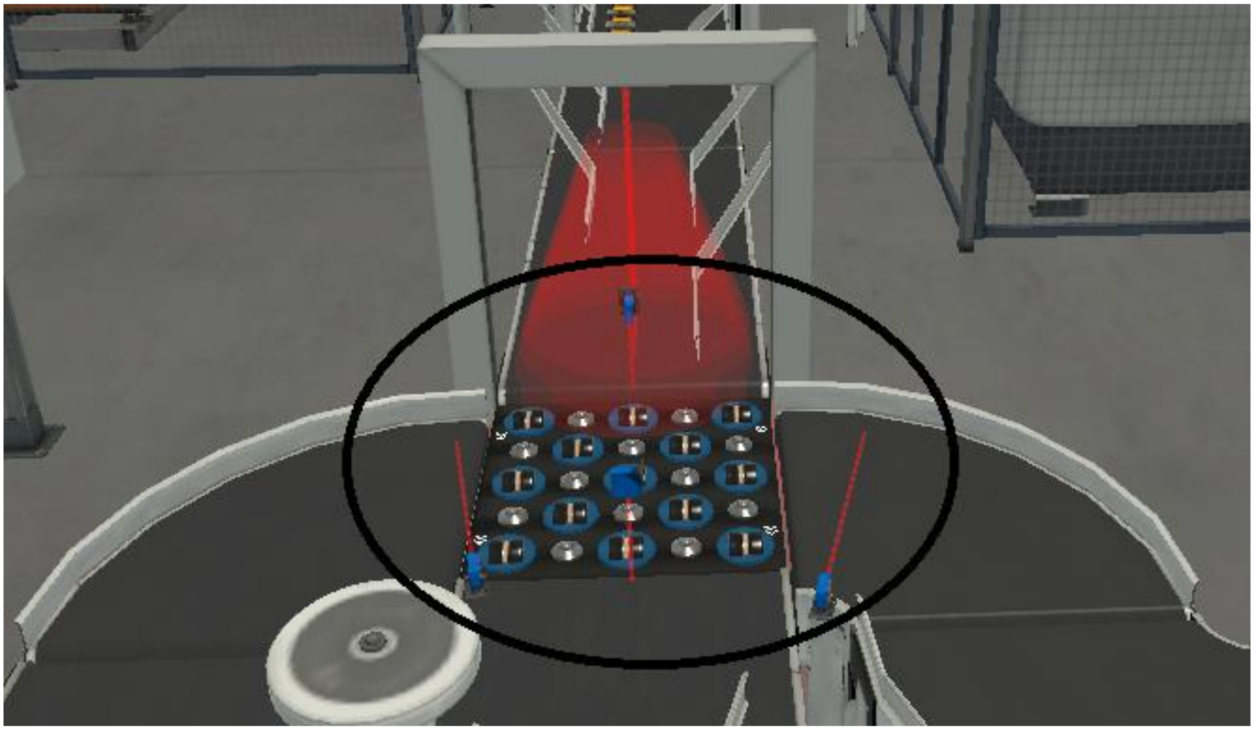
FigIII.15 The third 2 meters belt conveyer

- **Network 7:**

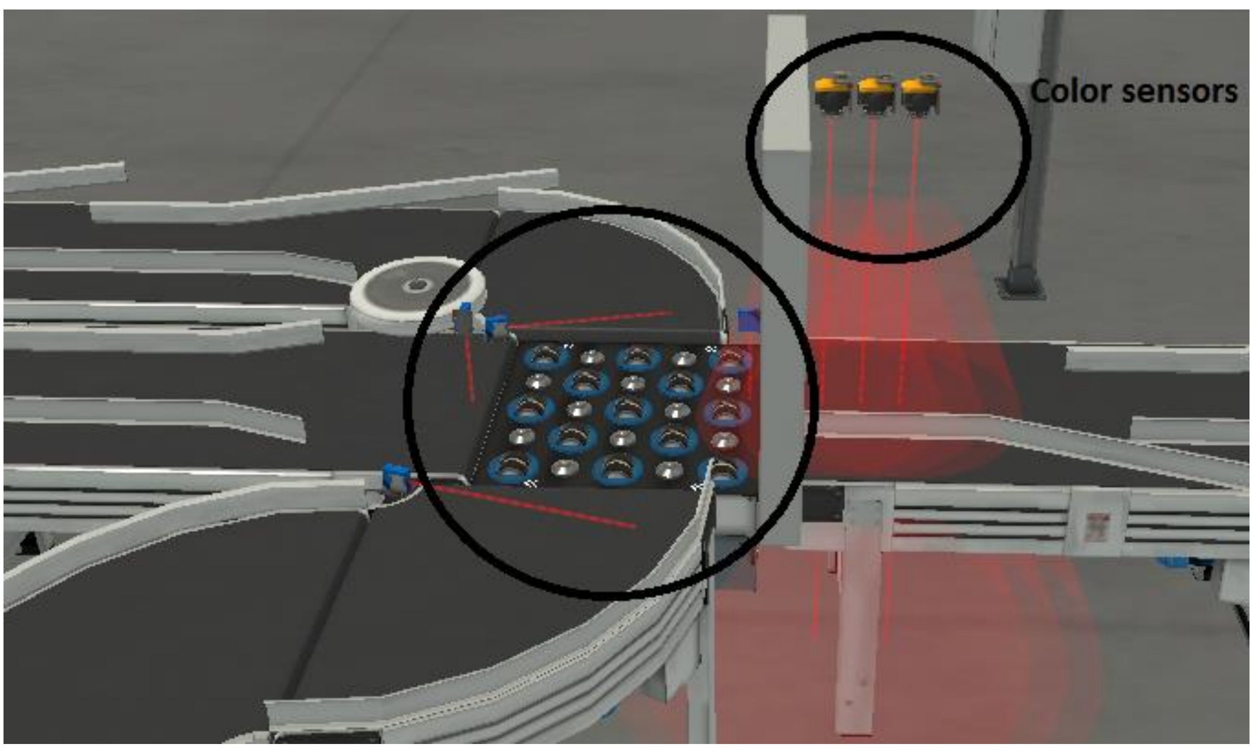
The seventh network contains the “Push Up wheel Sorter” block as shown in **FigIII.17**, This wheel sorter will sort the crafter material according to their color using the color sensors highlighted in Fig 17, the blue ones will go to the middle belt conveyer, the grey ones will go right, and the green ones will go left.



FigIII.16 Network 7



FigIII.17 The push up wheel sorter



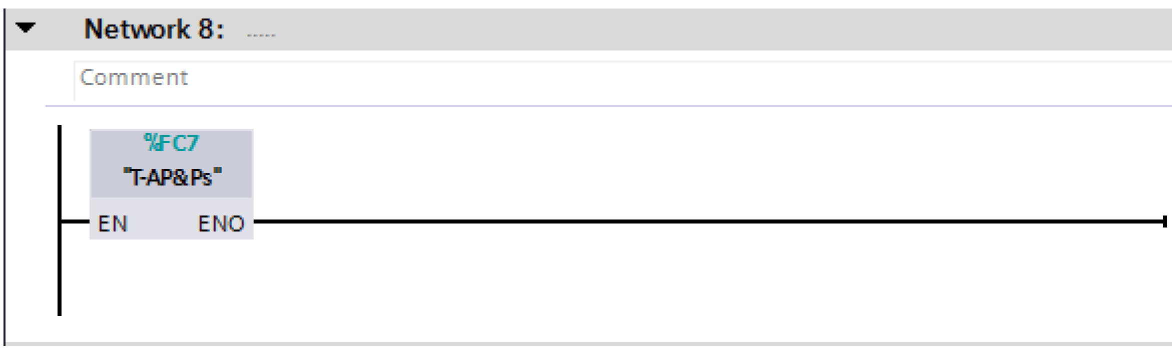
FigIII.18 The color sensors

- **Network 8:**

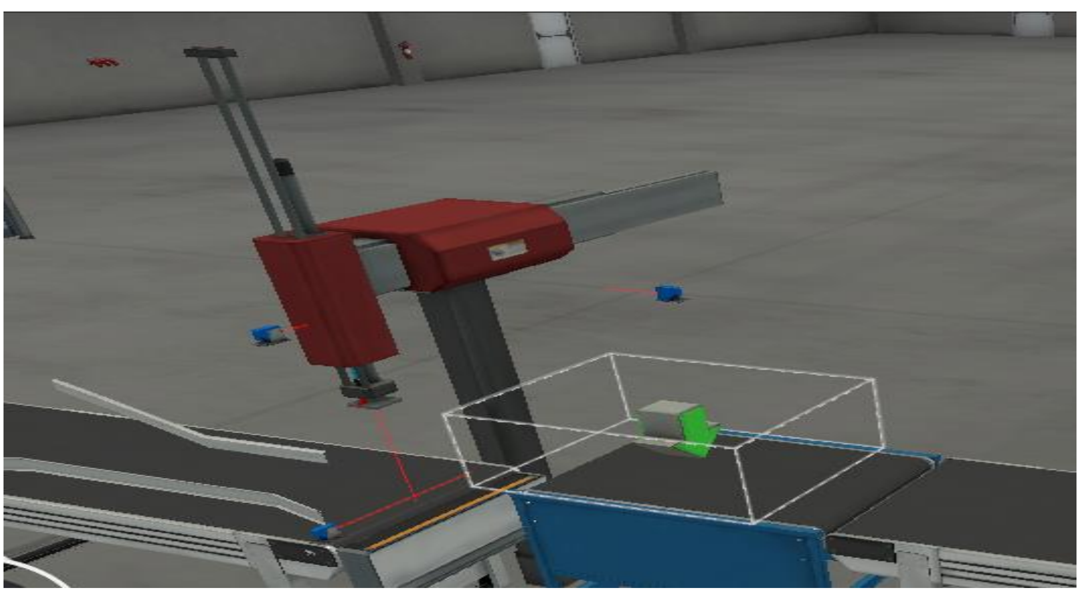
The eighth and last network contains the “Two Axis Pick And Place” block, the this block controls how the arm with 8 degrees of freedom operate as seen in **FigIII.20**. This block is the most complicated block in the program, because each arm needs 4 external proximity sensors as shown in **FigIII.21**.

PS 1 is the responsible for counting the number of crafted material, Whenever the crafted material interrupts the proximity sensor signal, 1 is added to the number of crafted material that needs to be transported to their final destination the stackable boxes.

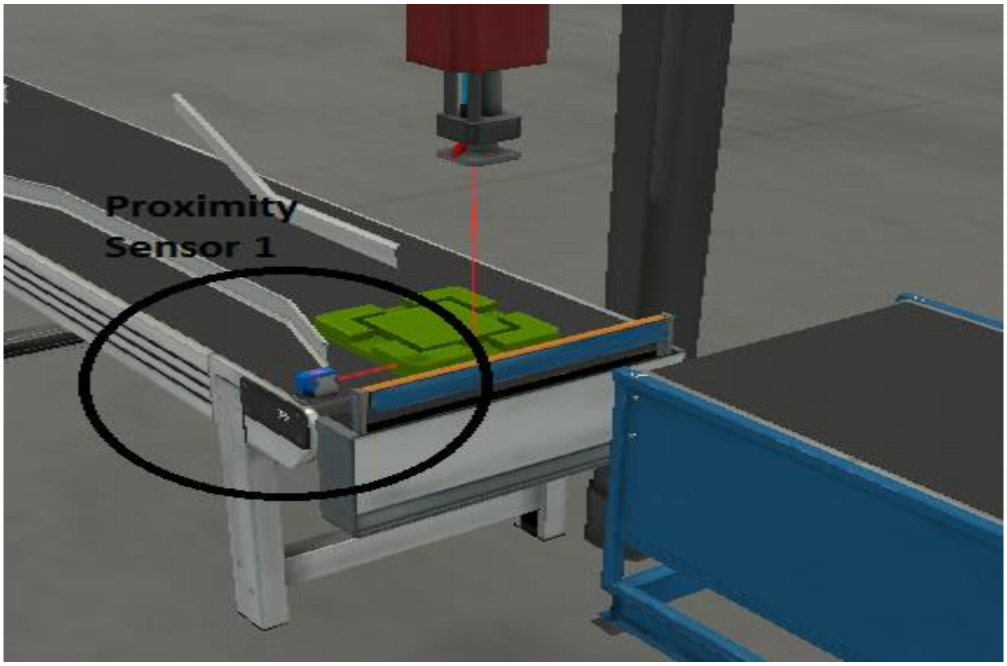
This stackable box will hold 5 pieces and then sent away to be shipped and leave the place for another box to be placed as shown **FigIII.22**.



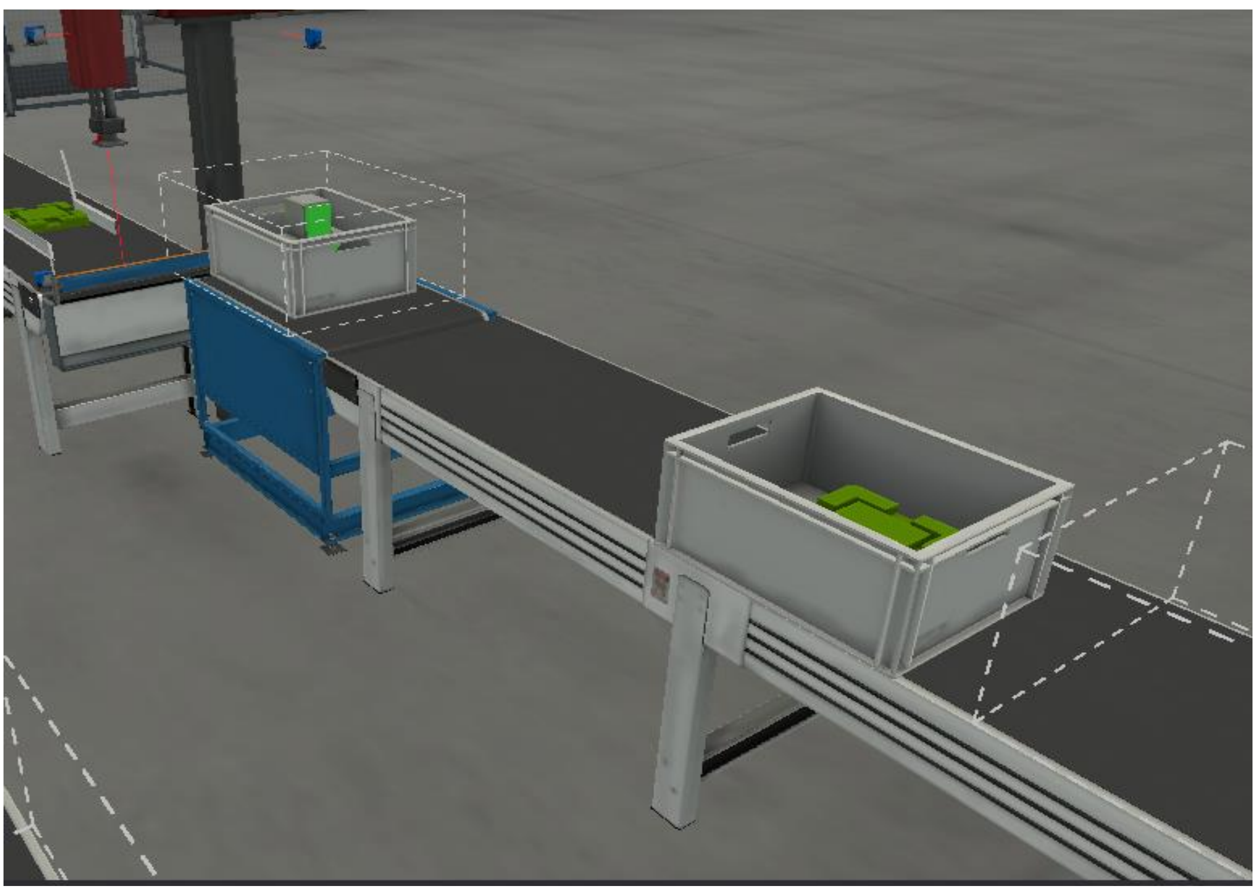
FigIII.19 Network 8



FigIII.20 The two Axis Pick and Place Arm.



FigIII.21 The counting proximity sensor.



FigIII.22 Stackable boxes.

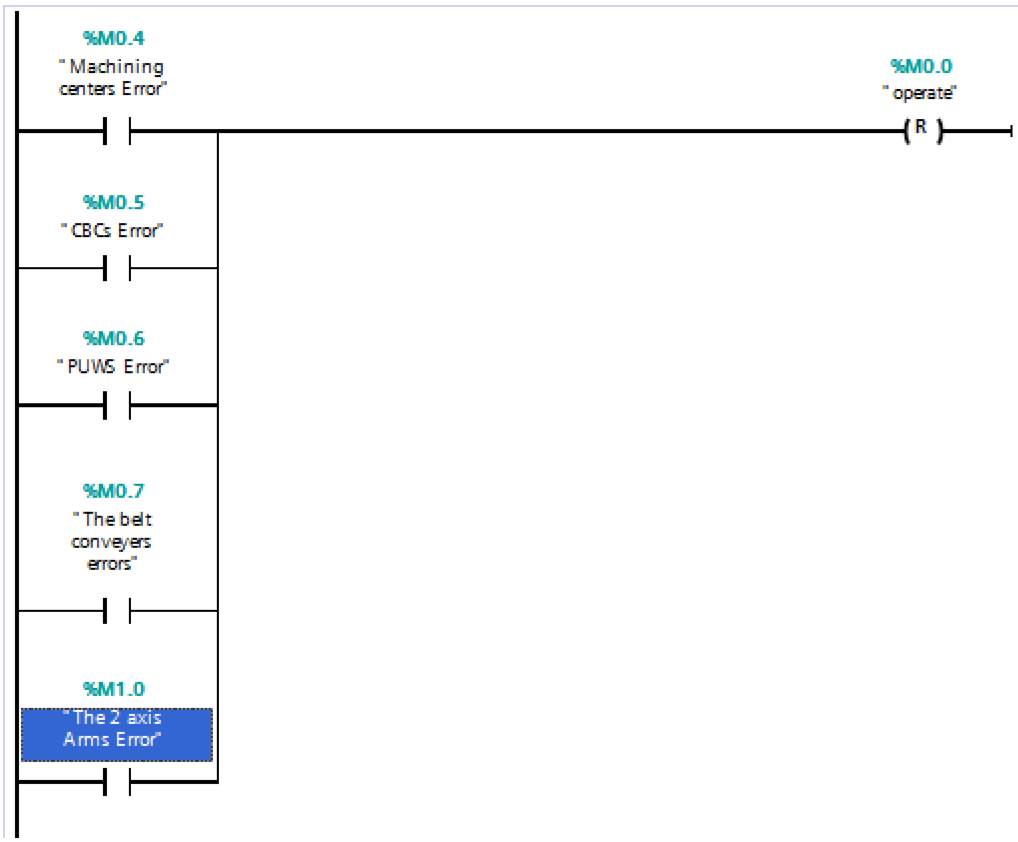
III.3. Alarms and safety:

The last network of the program network 9 contains the alarms of the system, these alarms are:

- The Machining centers Error.
- The curved belt conveyers Error.
- The Push up Wheel Sorter Error.
- The Straight Belt conveyers Errors.
- The Two Axis Pick and Place Error.

Any one of these Errors will stop the Operation.

- **Network 9:**

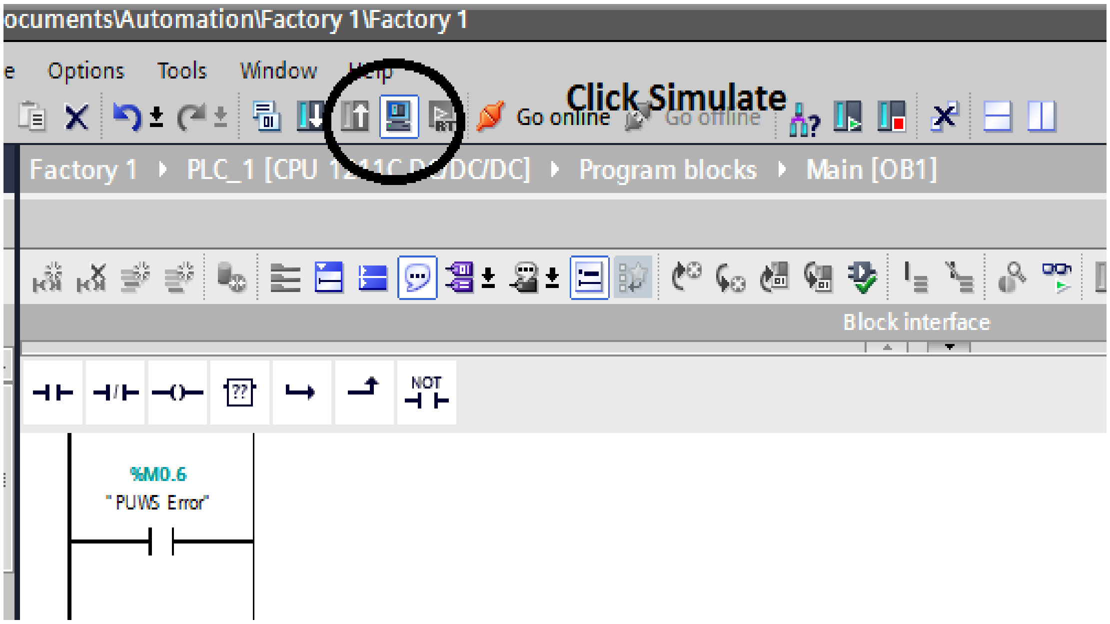


FigIII.23 Network 9.

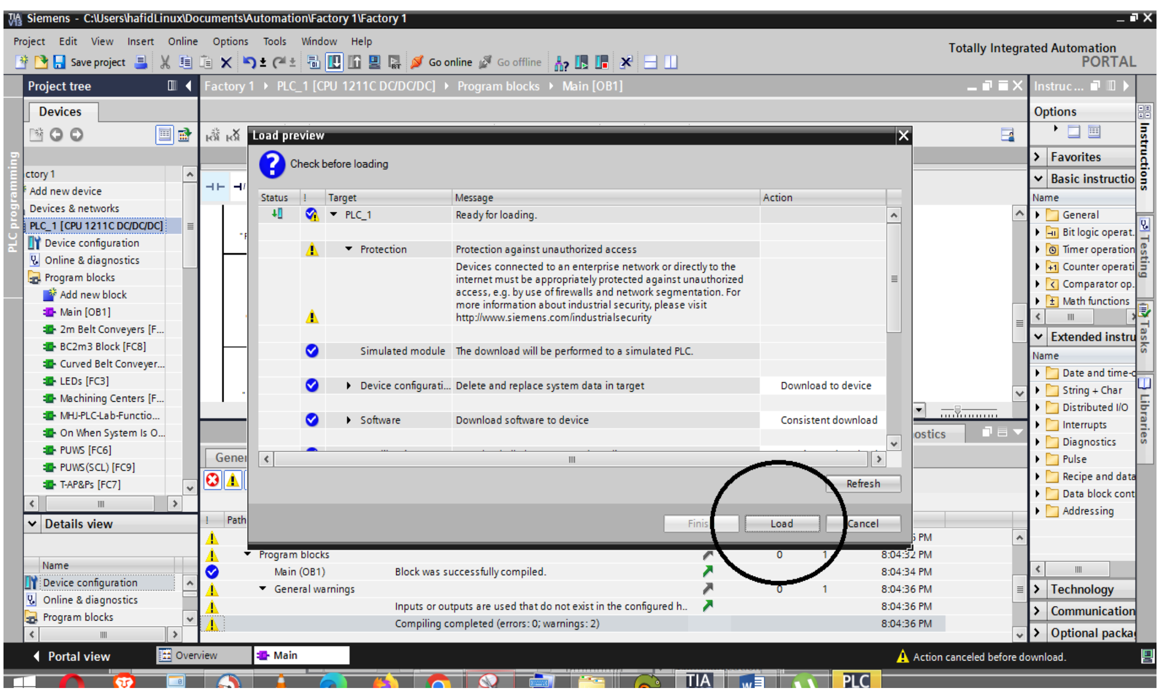
III.4. Connecting factory I/O to the program:

In order to connect the program to factory I/O:

1. Simulate the Program from TIA Portal Using PLCSIM:



FigIII.24 Step 1 of connecting Factory I/O to TIA.



FigIII.25 Step 2 of connecting Factory I/O to TIA.

2. In Factory I/O Connect to PLCSIM:

FigIII.26 Step 3 of connecting Factory I/O to TIA.

.

GENERAL CONCLUSION

GENERAL CONCLUSION

In this modest work, we were able to create a virtual factory and getting to control different industrial hardware and movement parts, which allowed us to identify a solution for different problems, especially in acomplex project such as the organization of the program and taking into consideration every phenomena that can face us, thus enriching our knowledge of industrial automation and to get an experience acquired in this field.

The aim of this work was to automate a virtual raw material factory using a programmable industrial controller PLC. Perform simulations to see the terminal control response and potential alarmsbefore starting, taking into account the faults that may damage our system.

The programming software TIA portal and the software factory allowed us to test a real program on a simulated industrial parts and check the errors that may occur during the fabrication process, Also giving us a financial idea about the project we want to start.

The professional background has allowed us to better understand the project and to familiarize ourselves with the responsibilities of field engineers in this area specially the module EE452.

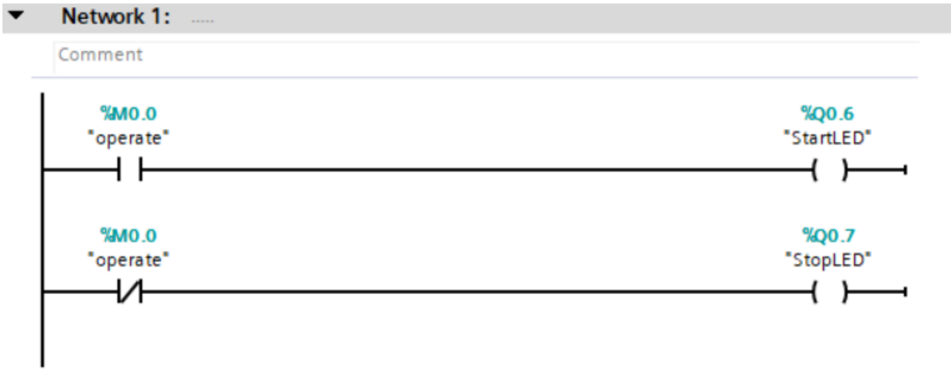
We hope that this humble work will be the starting point of our professional life and that it will be useful for future promotions, God willing.

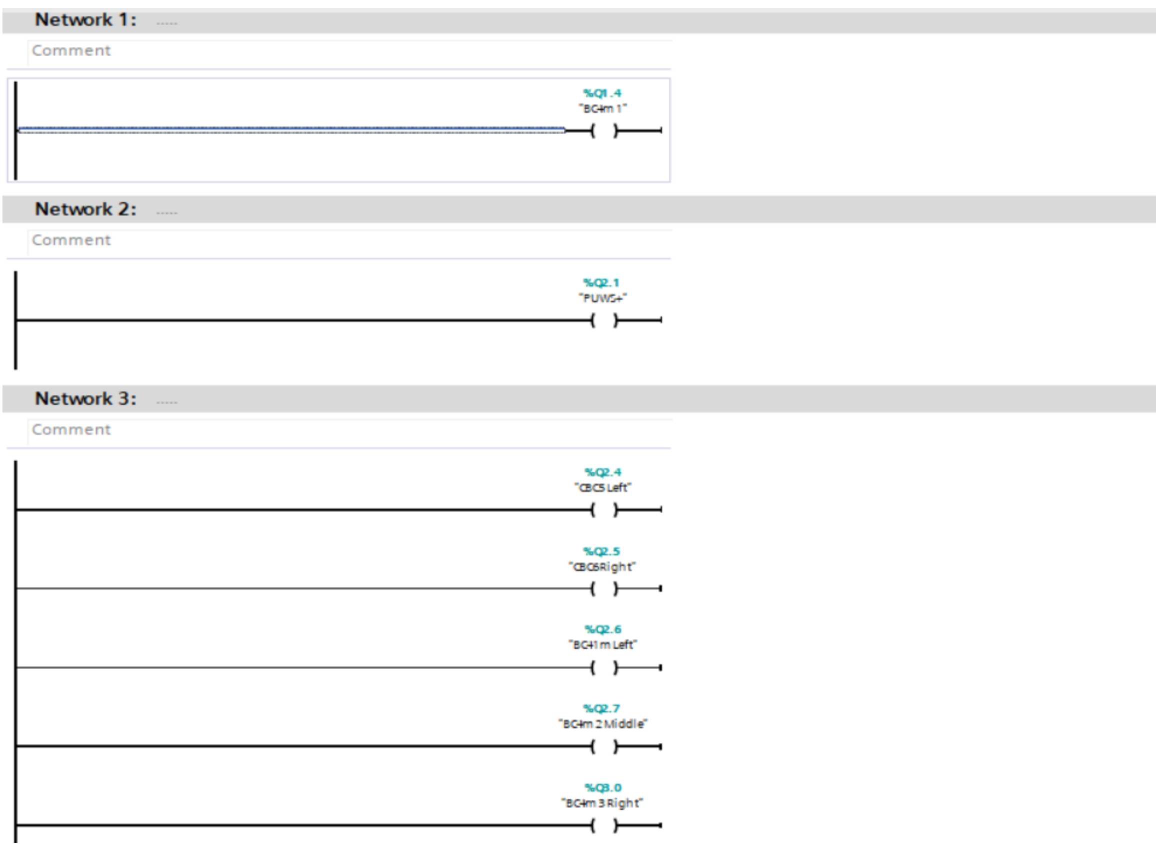
APPENDIX

APPENDIX

The PLC program of the simulation:

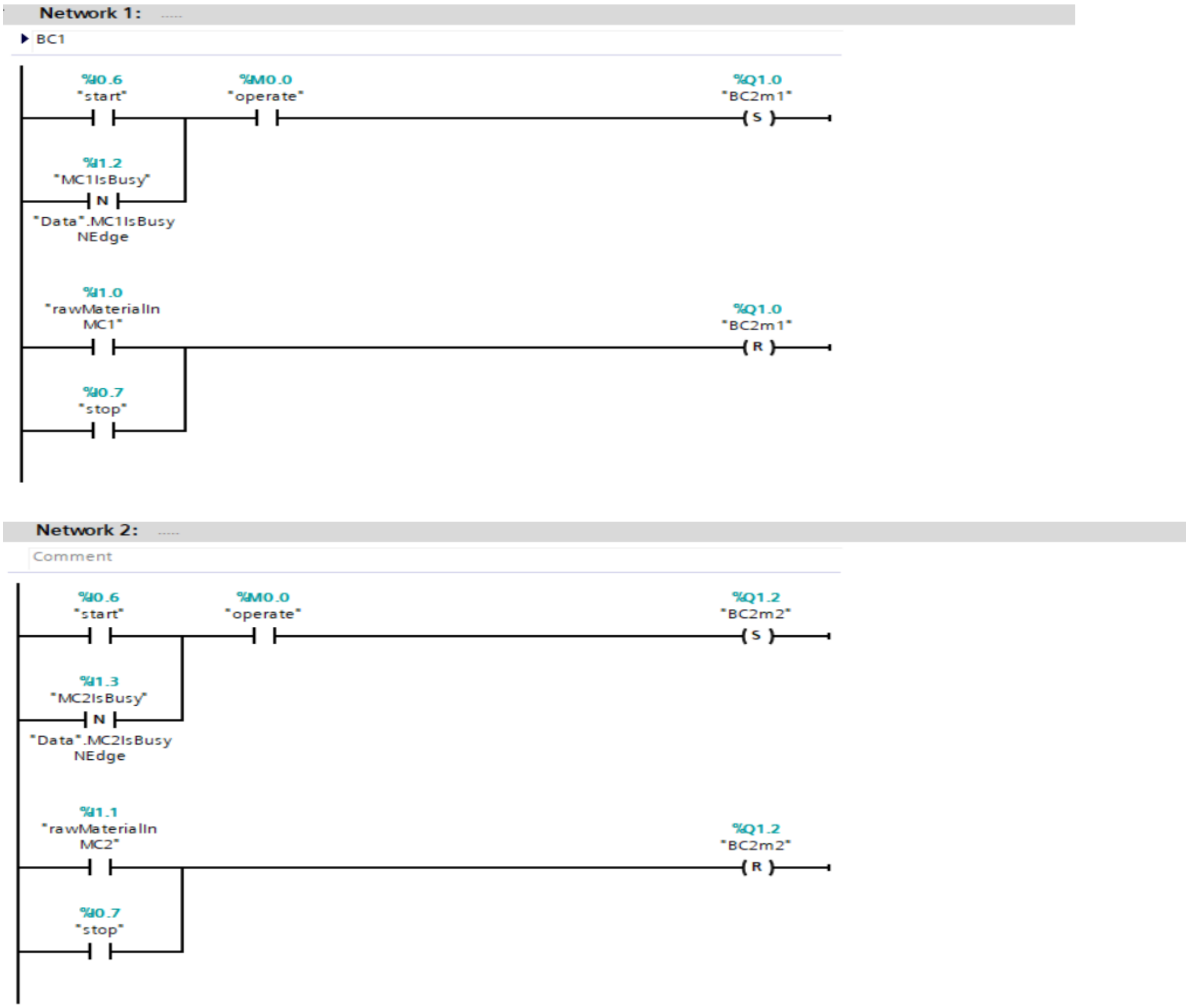
- **Network 2:**



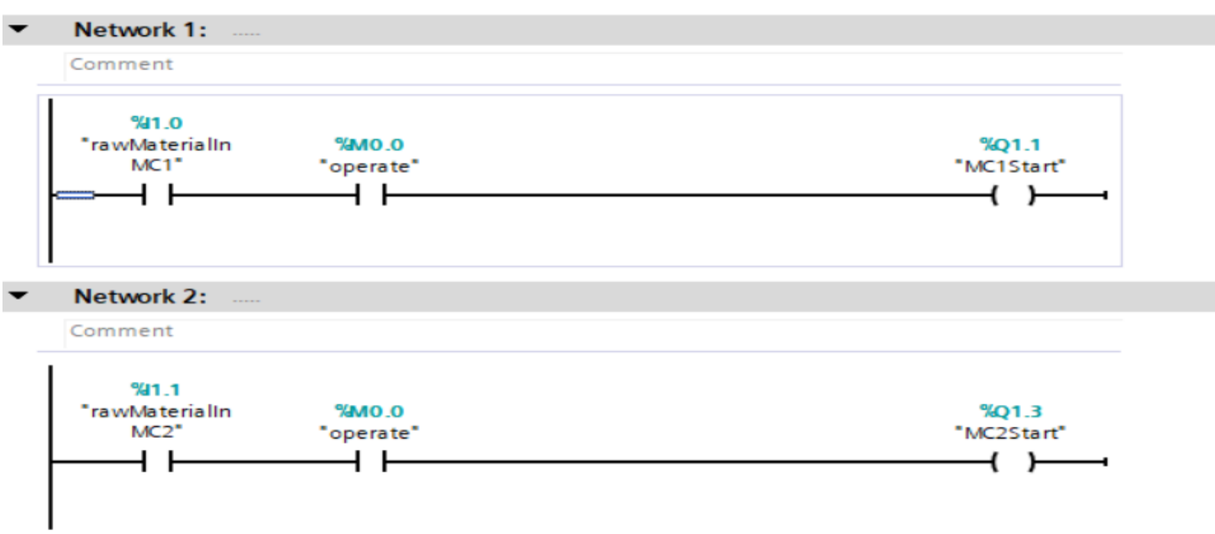


APPENDIX

- **Network 3:**

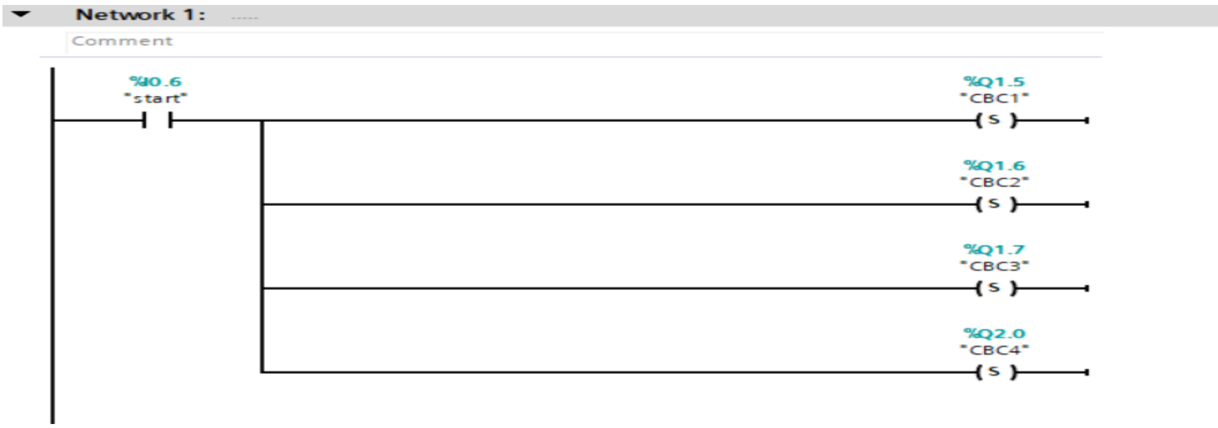


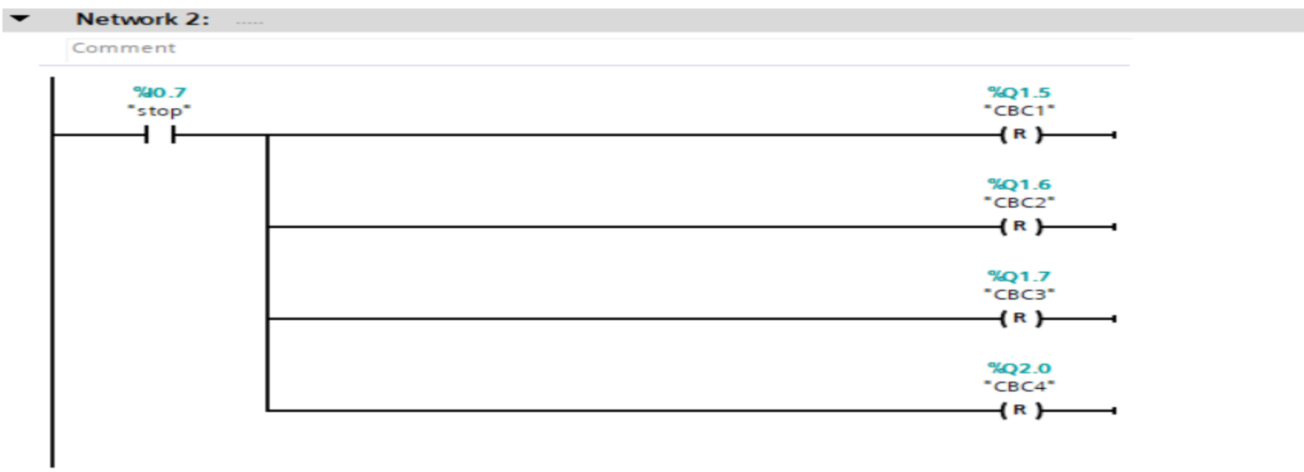
- **Network 4:**

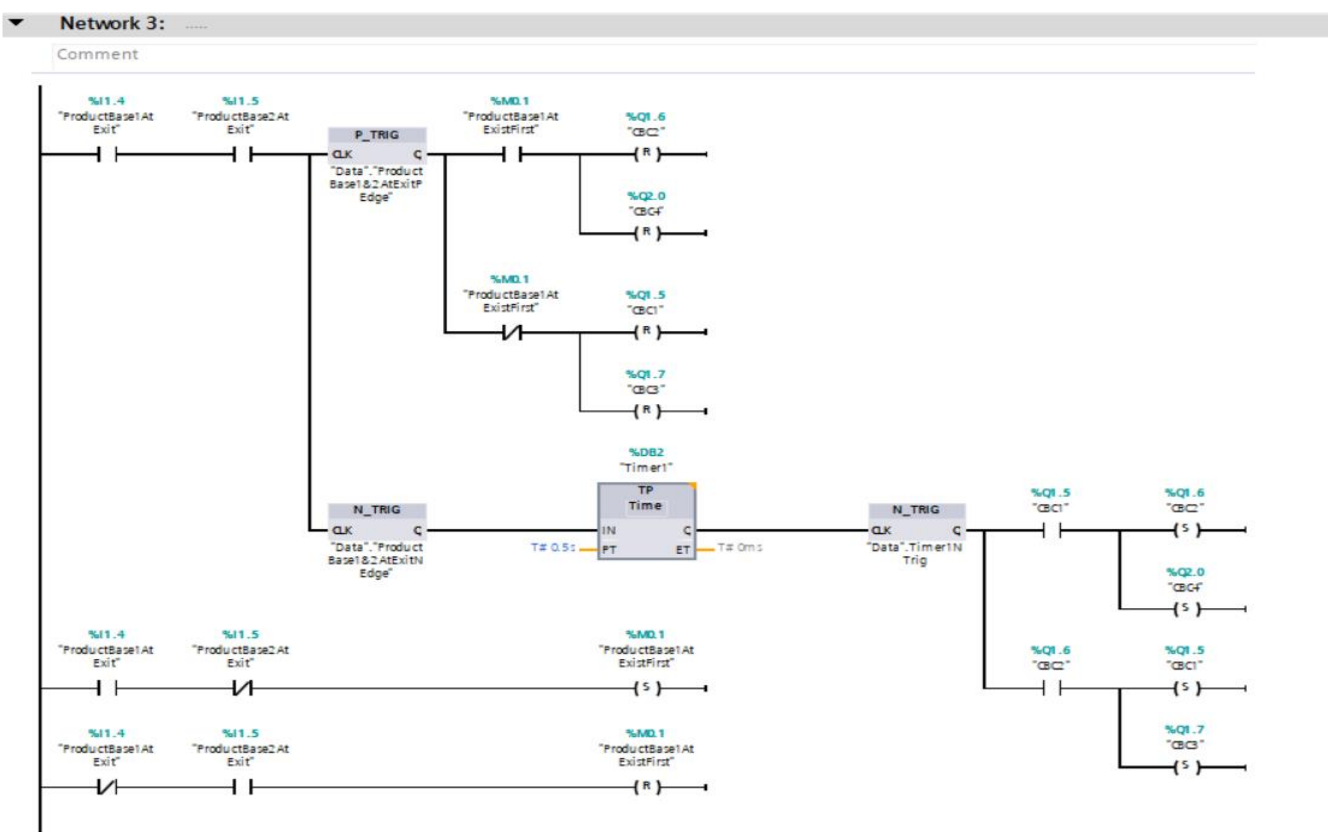


APPENDIX

- **Network 5:**

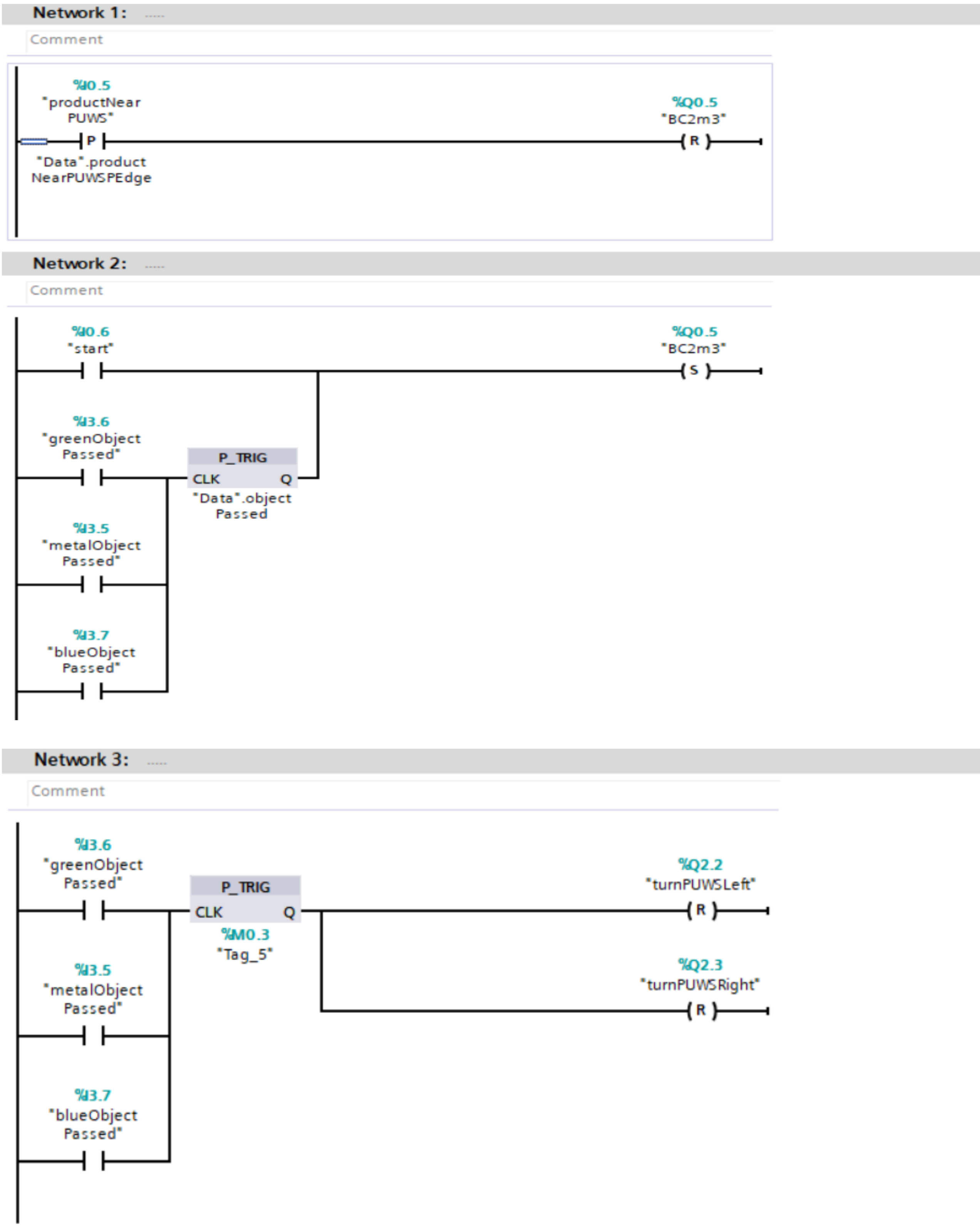






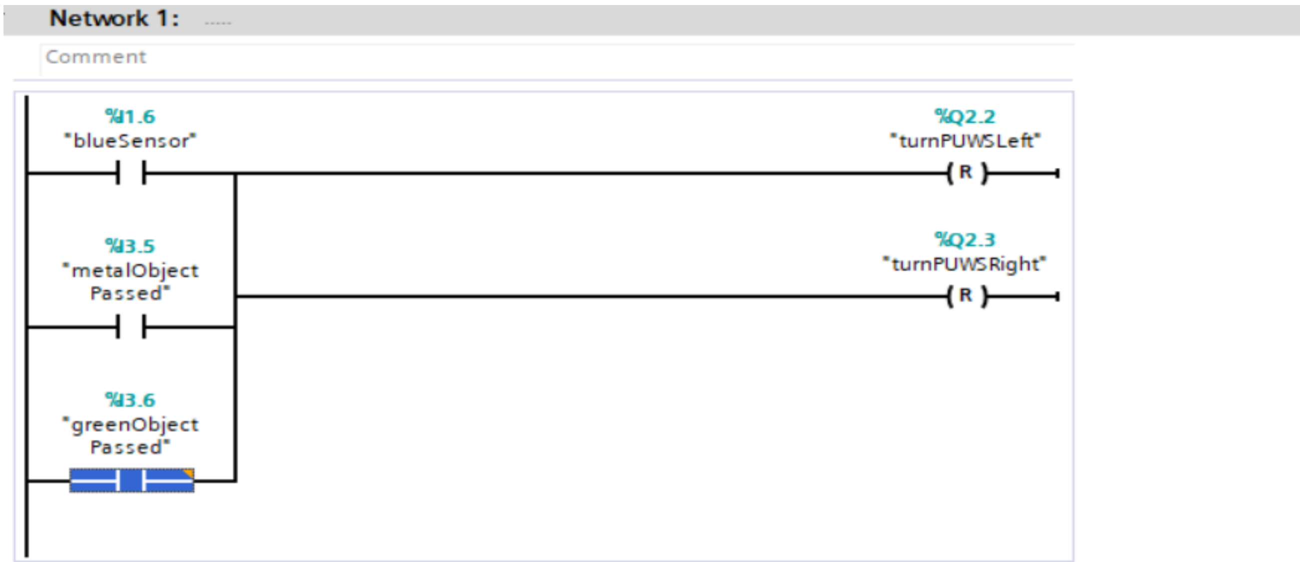
APPENDIX

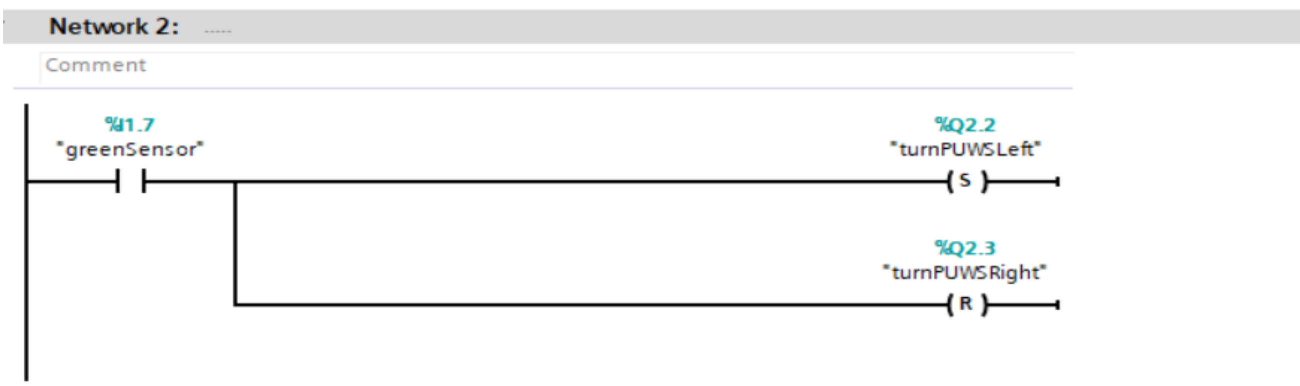
- **Network 6:**

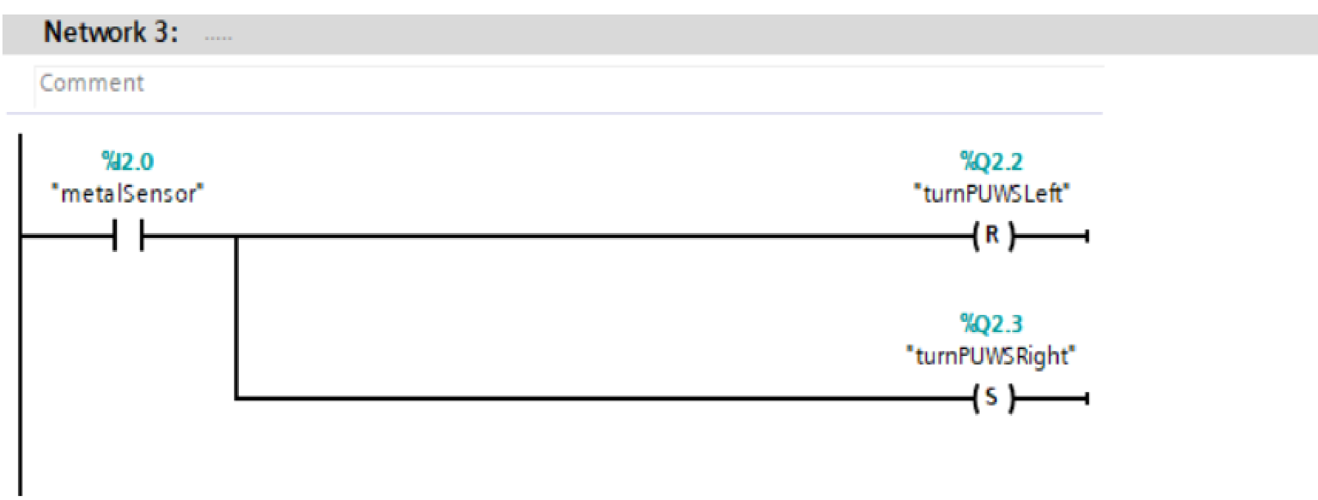


APPENDIX

- **Network 7:**

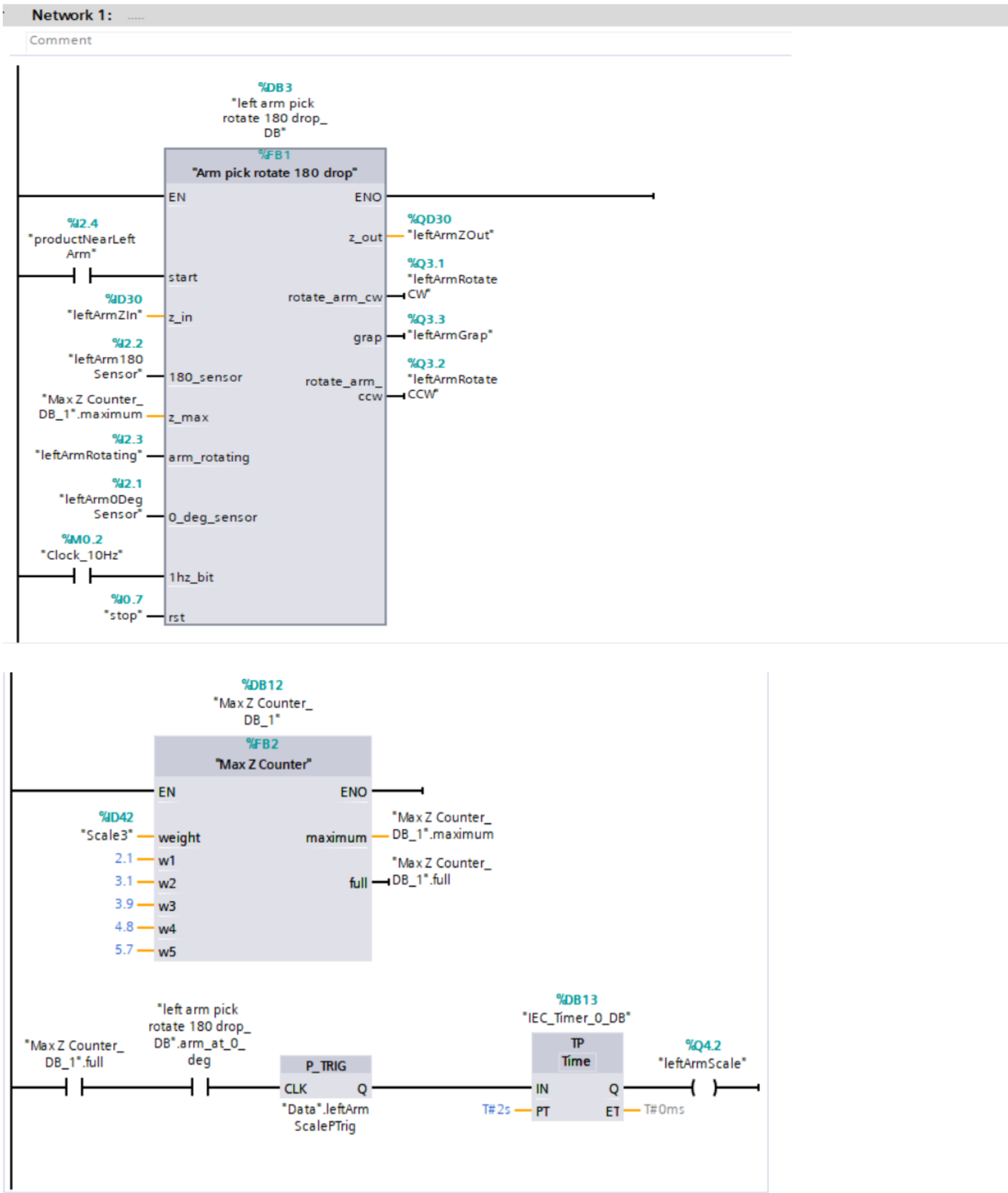




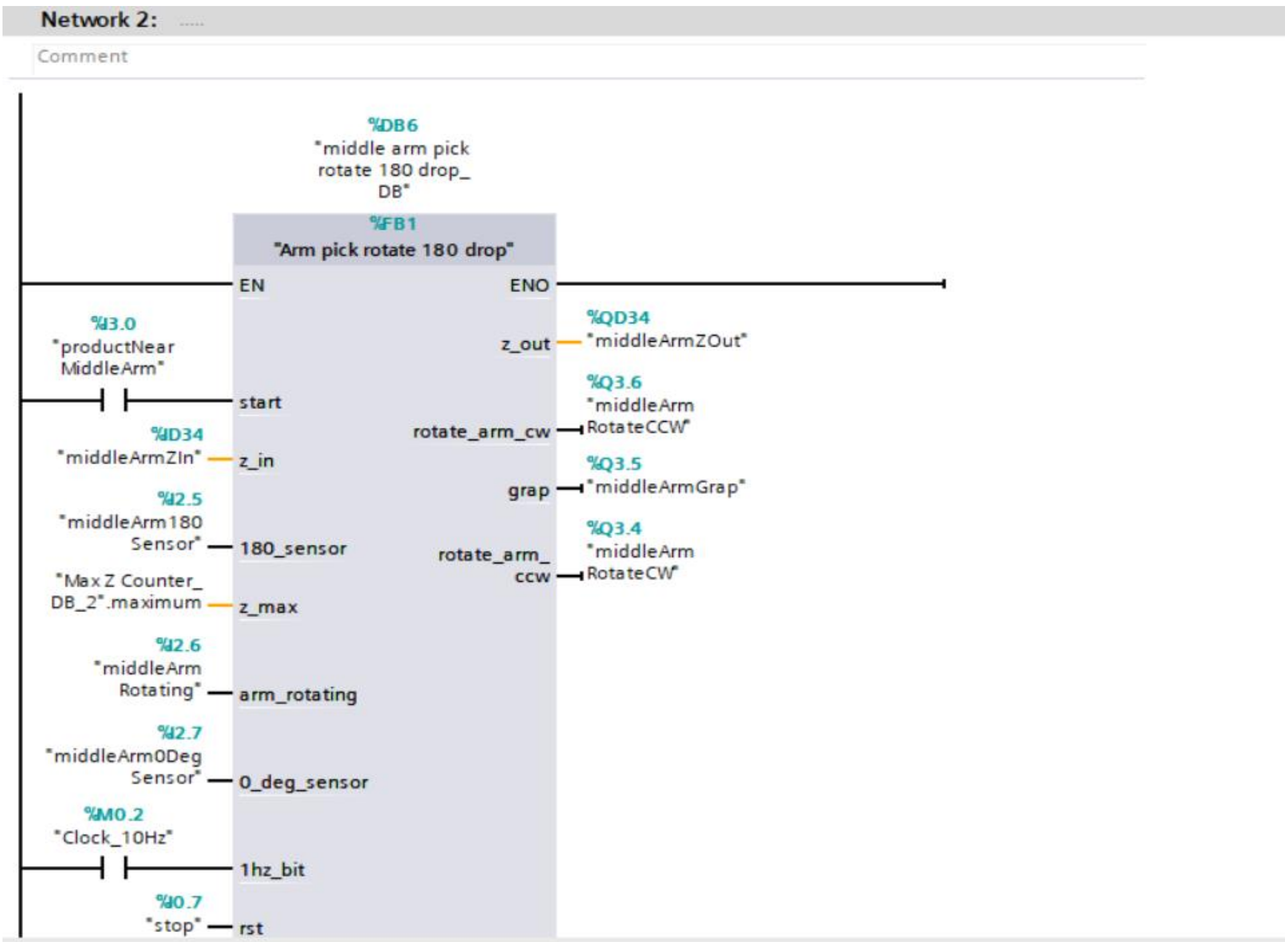


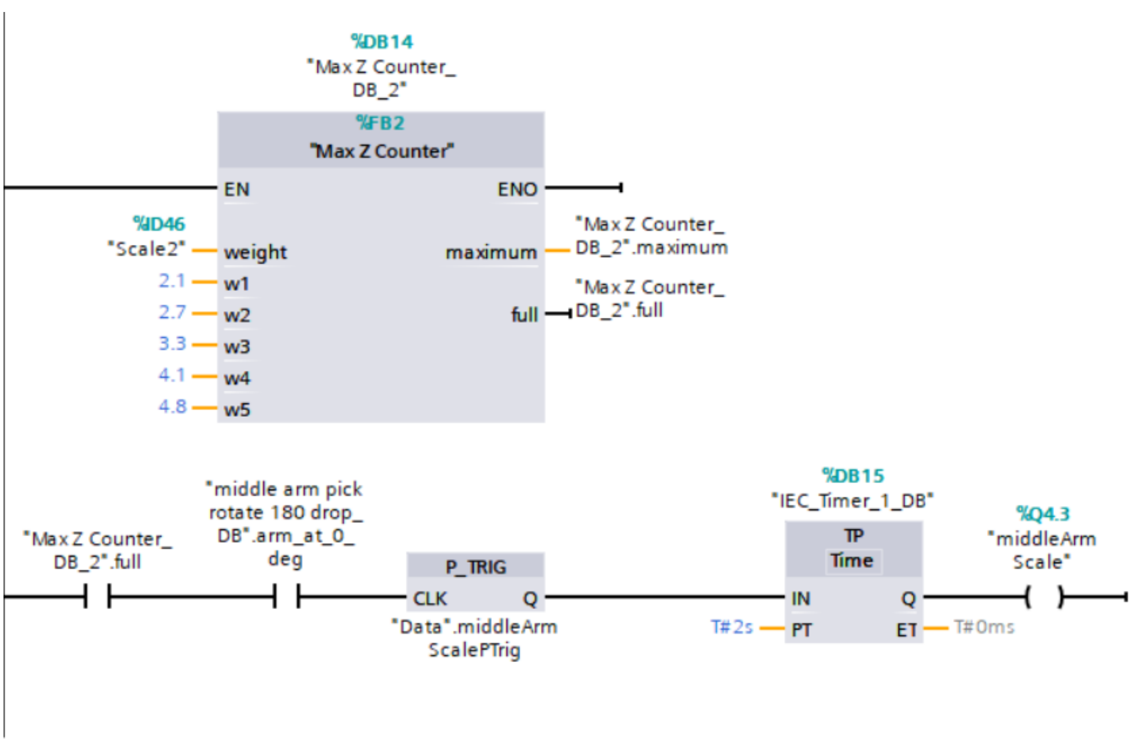
APPENDIX

- **Network 8:**

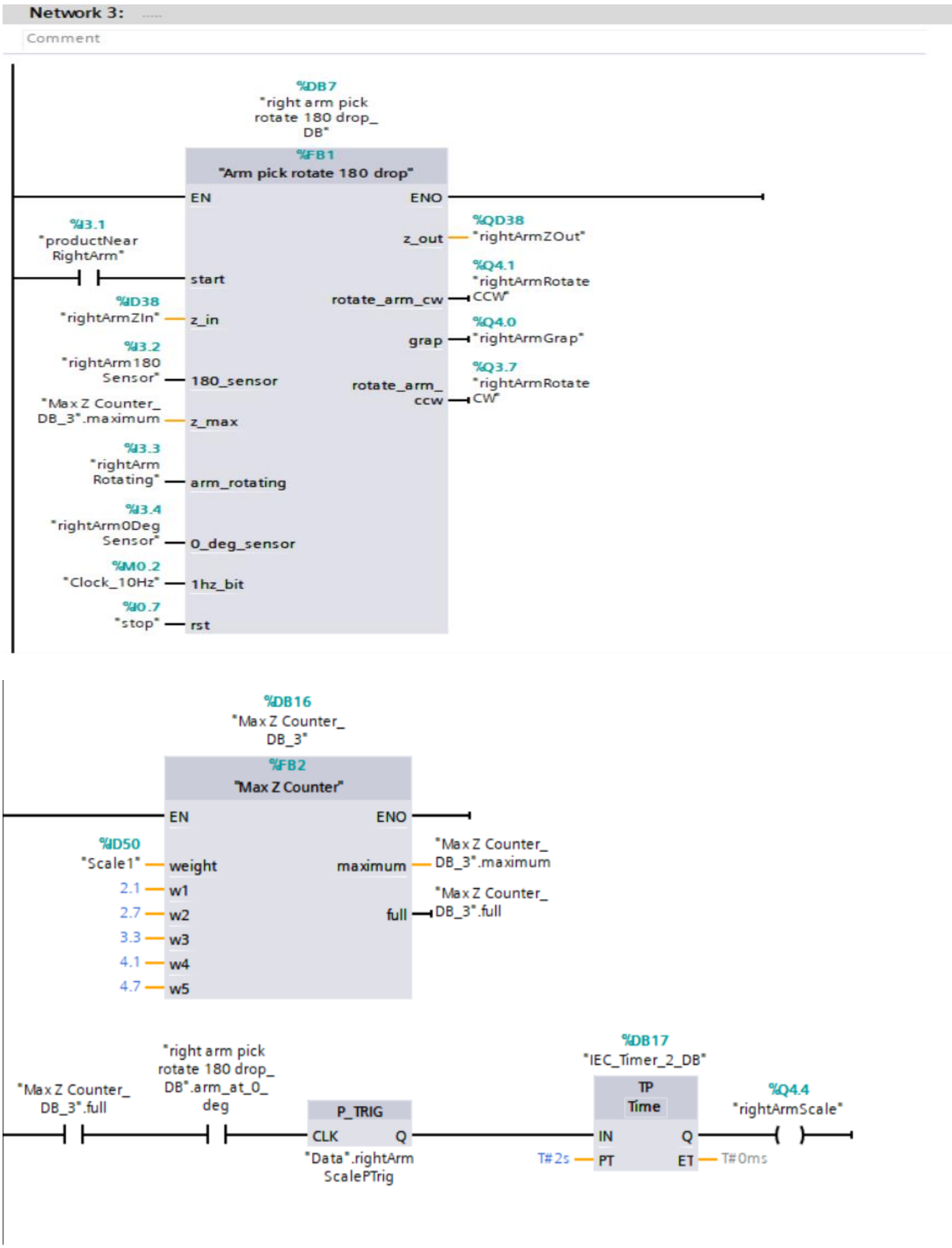


APPENDIX


Network 2:
Comment
%DB6
"middle arm pick rotate 180 drop_DB"
%FB1
"Arm pick rotate 180 drop"
EN
ENO
%I3.0
"productNear MiddleArm"
start
%ID34
"middleArmZIn"
z_in
%I2.5
"middleArm180 Sensor"
180_sensor
"Max Z Counter_DB_2".maximum
z_max
%I2.6
"middleArm Rotating"
arm_rotating
%I2.7
"middleArm0Deg Sensor"
0_deg_sensor
%M0.2
"Clock_10Hz"
1hz_bit
%I0.7
"stop"
rst
z_out
%QD34
"middleArmZOut"
rotate_arm_cw
%Q3.6
"middleArm RotateCCW"
grap
%Q3.5
"middleArmGrap"
rotate_arm_ccw
%Q3.4
"middleArm RotateCW"


%DB14
"Max Z Counter_DB_2"
%FB2
"Max Z Counter"
EN
ENO
%ID46
"Scale2"
weight
2.1
w1
2.7
w2
3.3
w3
4.1
w4
4.8
w5
maximum
"Max Z Counter_DB_2".maximum
full
"Max Z Counter_DB_2".full
"Max Z Counter_DB_2".full
"middle arm pick rotate 180 drop_DB".arm_at_0_deg
P_TRIG
CLK
Q
"Data".middleArm ScalePTrig
%DB15
"IEC_Timer_1_DB"
TP
Time
IN
Q
T#2s
PT
ET
T#0ms
%Q4.3
"middleArm Scale"

APPENDIX



Network 3:
Comment
%DB7
"right arm pick rotate 180 drop_DB"
%FB1
"Arm pick rotate 180 drop"
EN
ENO
%I3.1
"productNear RightArm"
start
%ID38
"rightArmZIn"
z_in
%I3.2
"rightArm180 Sensor"
180_sensor
"Max Z Counter_DB_3".maximum
z_max
%I3.3
"rightArm Rotating"
arm_rotating
%I3.4
"rightArm0Deg Sensor"
0_deg_sensor
%M0.2
"Clock_10Hz"
1hz_bit
%I0.7
"stop"
rst
z_out
%QD38
"rightArmZOut"
rotate_arm_cw
%Q4.1
"rightArmRotate CCW"
grap
%Q4.0
"rightArmGrap"
rotate_arm_ccw
%Q3.7
"rightArmRotate CW"
%DB16
"Max Z Counter_DB_3"
%FB2
"Max Z Counter"
EN
ENO
%ID50
"Scale1"
weight
2.1
w1
2.7
w2
3.3
w3
4.1
w4
4.7
w5
maximum
"Max Z Counter_DB_3".maximum
full
"Max Z Counter_DB_3".full
"Max Z Counter_DB_3".full
"right arm pick rotate 180 drop_DB".arm_at_0_deg
P_TRIG
CLK
Q
"Data".rightArm ScalePTrig
%DB17
"IEC_Timer_2_DB"
TP
Time
IN
Q
T#2s
PT
ET
T#0ms
%Q4.4
"rightArmScale"

APPENDIX

Network 4:
Comment
%DB4
"timer2"
TON
Time
"timer3".Q
IN
Q
T#50ms
PT
ET
T#0ms
%DB5
"timer3"
TON
Time
IN
Q
T#50ms
PT
ET
T#0ms
%M0.2
"Clock_10Hz"
Network 5:

REFERENCES

REFERENCES

[1] SIEMENS, « S7-1200_System_Manual », reference number of the document A5E02486682-AG 03/2014.

[2] PLC Basic Course with SIMATIC S7 by Jürgen Kaftan, first Edition (2011).

[3] SIMATIC S7-1200 manual.

[4] SIEMENS SIMATIC, programming with step7 manual.

[5] Sadiku, M. N. O., & Adediji, O. A. (2019). Introduction to Sensors. In Modern Sensors, Transducers, and Sensor Networks (pp. 3-15). John Wiley & Sons.

[6] O'Grady, P., & Guinee, V. (2018). Conveyors and Automation. In Introduction to Design for Civil Engineers (pp. 329-334). CRC Press.

[7] Öz, H., & Abdulkadir, G. (2018). A New Approach for Conveyor Belt Control in Industrial Applications. International Journal of Advanced Engineering Research and Science, 5(3), 142-150.

[8] Siemens system manual, "SIMATIC S7-1200 Programmable controller."

[9] https://cache.industry.siemens.com/dl/files/940/109772940/att_1002391/v1/s71200_system_manual_ en-US_en-US.pdf.

[10] Siemens, "Siemens en Algérie." Available: https://new.siemens.com/dz/fr/entreprise/about.html.

[11] SIEMENS PLC training document source : http://waterheatertimer.org/Woods-timers-and-manuals.html#Omron.

[12] TAFE Queensland. (n.d.). Programmable Logic Controller (PLC) Basics. Retrieved from https://tafeqld.edu.au/courses/16656/programmable-logic-controller-plc-basics