CHAPTER 1

INTRODUCTION

In this chapter, the introduction to the system and the system overview with the overall block diagram of the system are described. Aim and objectives of the system are also explained in this section along with the scopes and limitation of the overall system.

1.1 Introduction

The automation includes electronic, electrical and mechanical components. Accuracy, repeatability and manufacturing process are a major driving forces in the development of automation products. This system is for automation of pick and places for the objects. Before automation to manufacturing, all these tasks were carried out manually or with the help of electromechanical devices. The manual execution of tasks was time-consuming and lowered productivity. There are very dangerous areas of machines which can cause serious human injury. Automation overcomes all these barriers in industrial processes. Manual or semi-automatic machines are making production less efficient, human risk is higher and the production cannot be imagined with an unstructured system. Programmable logic controller is the logic controller whose program can be written in very flexible way. All the hardwiring on the electrical panels of machines or processes has been reduced because of PLC.

This research is to help maximize the efficiency of any automated assembly system by reducing labor costs and increasing productivity. Pick and place automation systems are widely used in industrial processes as a section of complete assembly system. Most of pick and place systems are typically used to pick and place parts or components in fixtures for the assembly process, or these are used to pull components of the system and position them for packaging or sending them to the next stage in the assembly process.

In this research, the automatic pick and place conveyor transfer system is designed and implemented based on Mitsubishi FX3U 24MT PLC. Push buttons, N20 gear motor, photoelectric sensor, conveyor belts, proximity sensors, electromagnet, pully 40T, pully 10T, pneumatic cylinder, power supply, DC 12V motors and keypad HMI are required equipment of the system. The target of automatic pick and place conveyor transfer system is to automate the process of moving objects from one location to another within a manufacturing or packaging environment [1].

1.2 System Overview

This system consists of two conveyor belts placed at 90-degree angle to carry the metal and non-metal blocks. Both conveyors will begin operation when the start button is pressed. The blocks (metal or non-metal) are put on the first conveyor and as soon as the metallic block on the first conveyor is detected by proximity sensor 1, the sensor receives the input signal which causes both conveyors to stop. At this point, the block is picked with the help of an electromagnetic arm mounted on pneumatic cylinder with solenoid valves and the block is placed on the second conveyor to reach the destination. Once the metallic block is detected by proximity sensor 2 on the second conveyor, both conveyors will resume operation. There is also photoelectric sensor at the end of the first conveyor to detect the non-metallic blocks. The entire system can be monitored on HMI and the numbers of metal and non-metal blocks are expressed on HMI. Human machine interface can also be used to start and stop the system. The overall block diagram of the system is shown in Figure 1.1.

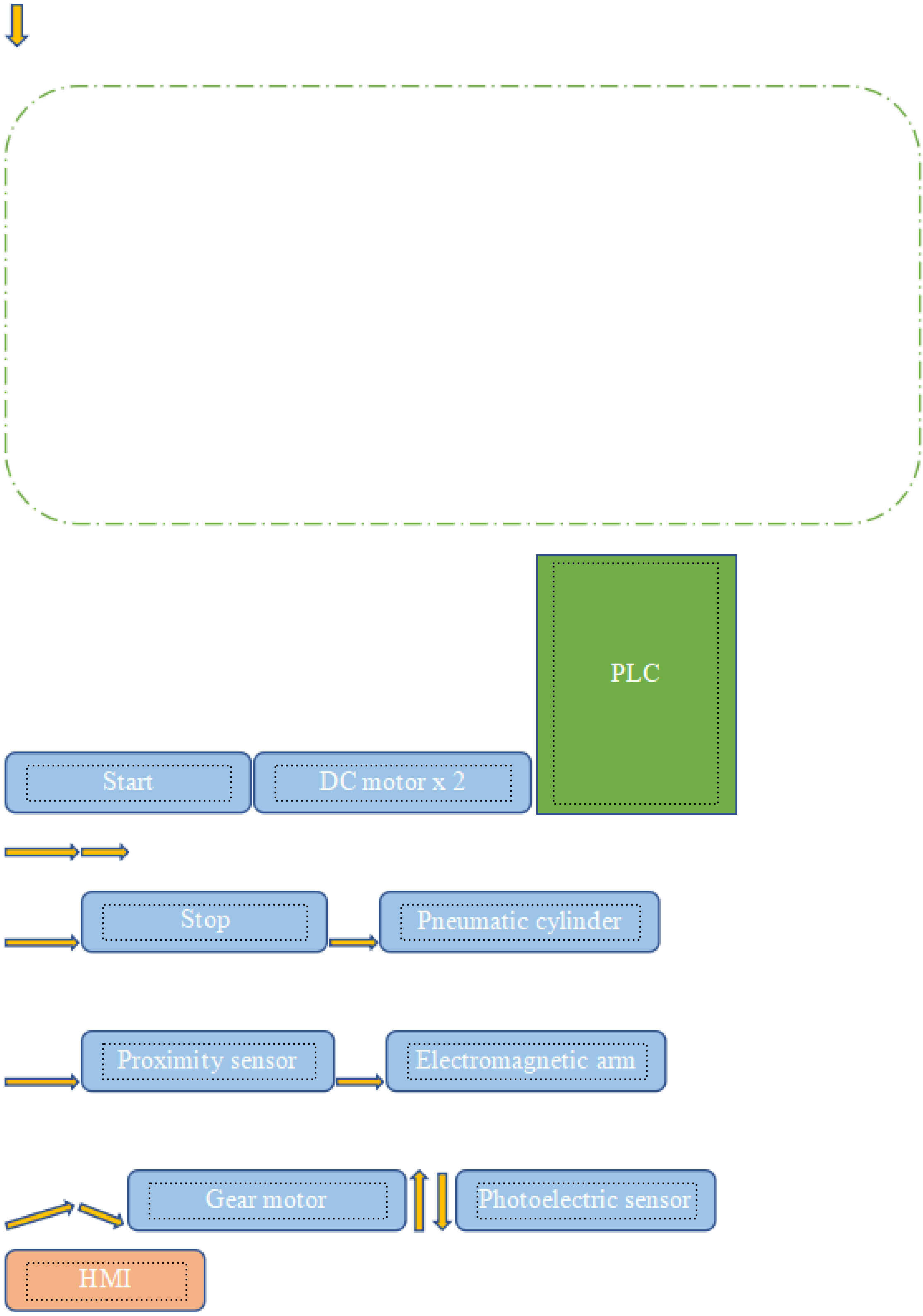

PLC
Start
DC motor x 2
Stop
Pneumatic cylinder
Proximity sensor
Electromagnetic arm
Gear motor
Photoelectric sensor
HMI

Figure 1.1 Overall Block Diagram of the System

1.3 Aim and Objectives

The aim of the study is to simulate the automatic pick and place conveyor transfer system using CX-Programmer software. This automation is designed to achieve several key objectives:

- To reduce the time required to move products from one stage of production to another
- To increase rate of process completion
- To implement pick and place conveyor transfer system using ladder logic
- To learn more about PLC and HMI
- To reduce labor costs by eliminating the need for manual labors
- To study about CX-Programmer programming language
- To study instructions sets, basic functions and special functions of CX-Programmer

1.4 Scopes and Limitation of Thesis

This system is primarily focused to develop desires and experiences. The research aims to investigate the design and implementation of pick and place conveyor transfer system. The main functions of this system are picking the metallic blocks on conveyor 1 by using electromagnetic arm, placing the metallic blocks on conveyor 2 with the help of pneumatic cylinder, counting the number of metallic and non-metallic blocks and showing the results on Human Machine Interface (HMI).

1.5 Outlines of Thesis

In this thesis, five chapters will be discussed. Firstly, in chapter one, introduction of the system and system overview are described. In chapter two, background theory of the thesis will be described with brief description of PLC, and introduction of OMRON software, CX-Programmer. Then, overall operation and system software tools will be described in chapter three. In chapter four, test and results of the overall system will be explained. Finally, discussion, conclusion and further extension of the system will be expressed in chapter five.

CHAPTER 2

BACKGROUND THEOREY

Introduction to programmable logic controller including brief description of PLC and introduction to CX-Programmer are described in this chapter.

2.1 Introduction to Programmable Logic Controller

A programmable logic controller (PLC) is a special form of microprocessor-based controller that uses programmable memory to store instructions and to implement functions such as logic, sequencing, timing, counting, and arithmetic in order to control machines and processes. PLCs are capable of storing instructions, such as sequencing, timing, counting, arithmetic, data manipulation, and communication, to control industrial machines and processes.

PLCs were initially designed to replace electromechanical relay systems in order to offer a simpler solution for modifying the operation of a control system. PLCs are user-friendly, microprocessor based specialized computers that carried out control functions of many types and level of complexity. Its purpose is to monitor crucial process parameters and adjust process operations accordingly. It can be programmed, controlled and operated by a person. PLCs have the great advantage that the same basic controller can be used with a wide range of control systems.

The majority of PLCs today are modular, allowing the user to add an assortment of functionality including discrete and analog inputs and outputs, PID control, position control, motor control, serial communication, and high-speed networking. Compared to older technologies such as relay banks, the PLC is far easier to troubleshoot and maintain, more reliable, more cost-effective, and more versatile. PLCs are similar to computers, but whereas computers are optimized for calculation and display tasks [1].

PLCs are optimized for control tasks and the industrial environment. PLCs are rugged and designed to withstand vibrations, temperature, humidity, and noise. PLCs have interfacing for inputs and outputs already inside the controller. PLCs are easily understood programming language.

PLCs were first developed in the automobile manufacturing industry to provide flexible, ruggedized and easily programmable controllers to replace hard-wired relays, timers and sequencers. Since then, they have been widely adopted as high-reliability automation controllers suitable for harsh environments. A PLC is an example of a "hard" real time system since output results must be produced in response to input conditions within a limited time, otherwise unintended operation will result. PLCs can range from small modular devices with tens of inputs and outputs to large modular devices with a count of thousands of inputs and outputs.

Nowadays, PLCs are being incorporated, through networks, into computer-integrated manufacturing (CIM) systems, combining their power and resources with numerical controls, robots, CAD/CAM systems, personal computers, management information systems, and hierarchical computer-based systems [1].

2.2 Hardware Components of PLC System

Typically, a PLC system has the basic functional components of processor unit, memory, power supply unit, input/output interface system, communications interface and the programming device. The processor unit or central processing unit (CPU) is the unit containing the microprocessor and this interprets the input signals and carries out the control actions, according to the program stored in its memory, communicating the decisions as action signals to the outputs. Figure 2.1 shows the PLC system.

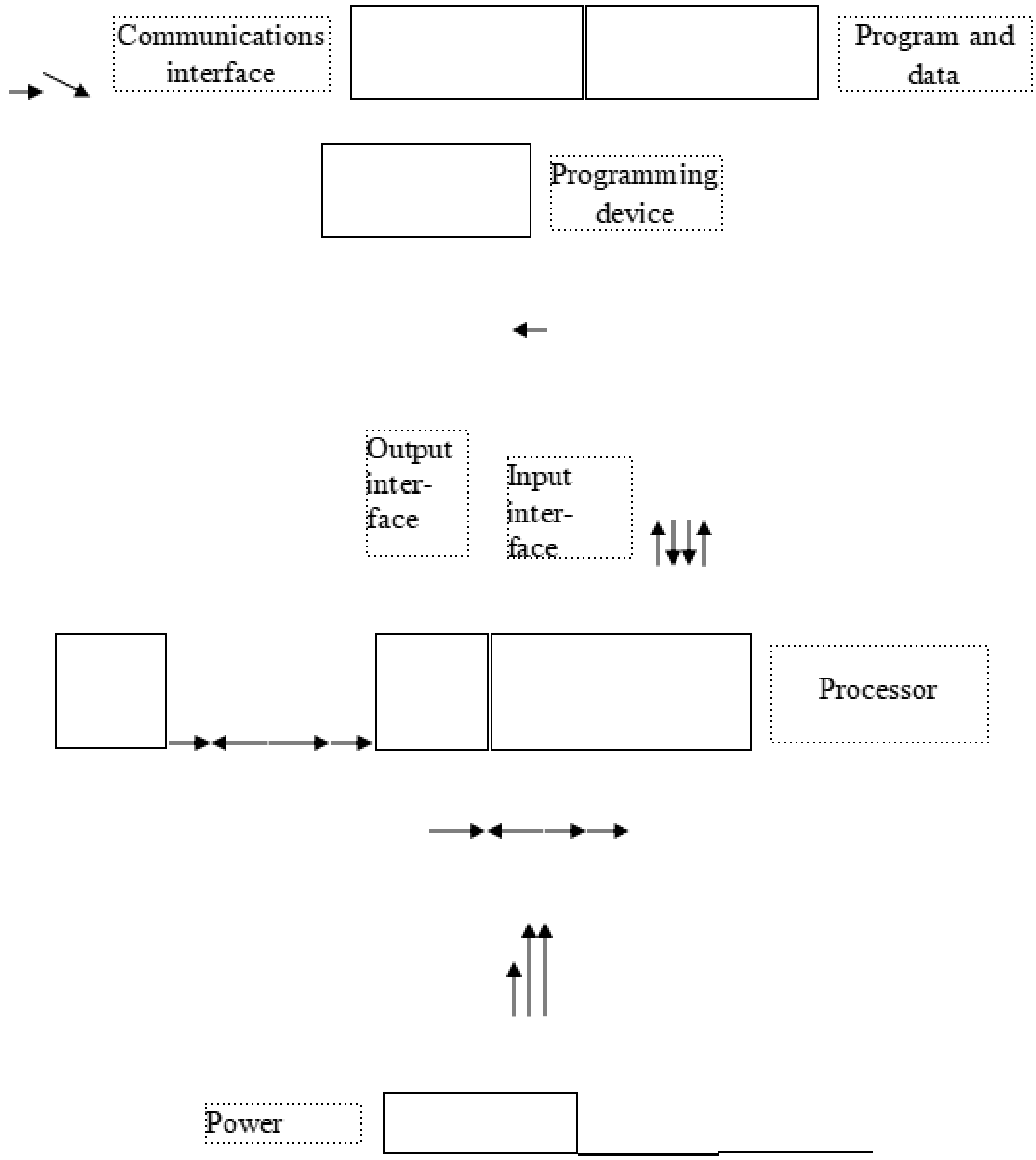


Figure 2.1 PLC System

The power supply unit is needed to convert the mains AC voltage to the low DC voltage (5V) necessary for the processor and the circuits in the input and output interface modules. The programming device is used to enter the required program into the memory of the processor. The program is developed in the device and then transferred to the memory unit of the PLC. The memory unit is a place where the program is stored that is used for the control actions to be exercised by the microprocessor and data stored from the input for processing and from the output for outputting [1].

2.2.1 Central Processing Unit (CPU)

CPU stands for central processing unit; it is also known as the brain of the computer. A CPU is a primary component of a computer that performs most of the processing and controls the operation of all components running inside a computer. The CPU is the core of the control system. The user program is stored in the internal storage of the CPU module after being downloaded through the programming software, and will be executed by the CPU. Meanwhile, it also executes the CPU self-test diagnostics: checks for proper operation of the CPU, for memory areas, and for the status of any expansion modules. The CPU consists of three components:

- Processor
- Memory system and
- System power supply

During its operation, the CPU completes three processes: (1) it reads, or accepts, the input data from the field devices via the input interfaces, (2) it executes, or performs, the control program stored in the memory system, and (3) it writes, or updates, the output devices via the output interfaces. This process of sequentially reading the inputs, executing the program in memory, and updating the outputs is known as scanning. Figure 2.2 illustrates a PLC scan. The system power supply provides all the voltages required for the proper operation of the various central processing unit sections [1].

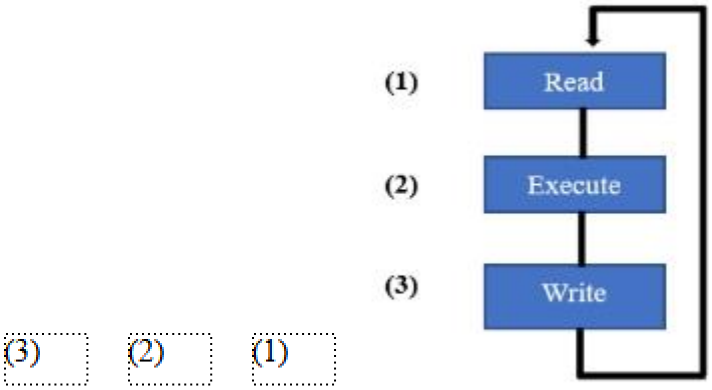


Figure 2.2 Illustration of a PLC Scan

2.2.2 Memory

There are several memory elements in a PLC system:

- System read-only-memory (ROM) to give permanent storage for the operating system and fixed data used by the CPU.
- Random-access memory (RAM) for the user's program.
- Random-access memory (RAM) for data. This is where the information is stored on the status of input and output devices and the values of timers and

counters and other internal devices. The data RAM is sometimes referred to as a data table or register table.

- Erasable and programmable read-only-memory (EPROM) for ROMs that can be programmed and then the program made permanent [1].

2.2.3 Power Supply

Today, learning all about PLC power supply, what it is, and how it works. A PLC without a power supply is much like a car without gas, or a laptop without a battery. It is the fuel for the PLC. How powerful a PLC is will greatly depend on how powerful the power supply.

Figure 2.3 Power Supply

A PLC power supply is the workhorse of the PLC system. It converts the line voltage, 120- or 240-volts AC, to a lower DC voltage, commonly 24 volts DC. The DC voltage is then sent into the rack to power the rest of the PLC components. Figure 2.3 shows the power supply of PLC. Most PLC controllers work either at 24VDC or 220VAC. Some PLC controllers have electrical supply as a separate module, while small and medium series already contain the supply module.

The different sizes are going reference current and be rated in amps or amperes. The common current ratings for PLC–s are anywhere from 2 to 10 amps for smaller systems and up to 50 amps for larger, more powerful controllers. This is an important rating for engineers and maintenance personnel to consider when designing a system or even modifying one. The current rating will directly affect how much work PLC system. While the main power supply is the primary source of power for the PLC, there is usually a battery backup as well [2].

2.2.4 Communications Interface

The communications interface is used to receive and transmit data on communication networks from or to other remote PLCs. It is concerned with such actions as device verification, data acquisition, synchronization between user applications and connection management. Figure 2.4 shows the basic communications model.

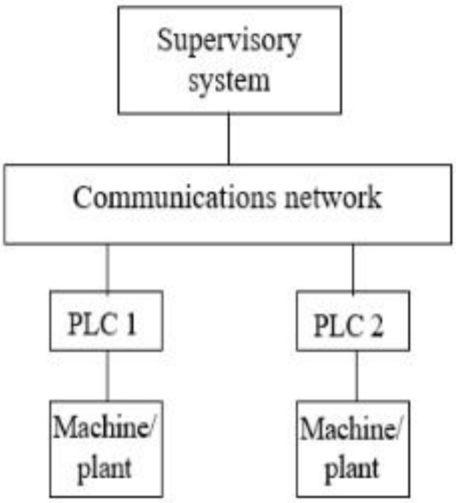


Figure 2.4 Basic Communications Model

2.2.5 Input/Output Interface System

The input/ output (I/O) system is physically connected to the field devices that are encountered in the machine or that are used in the control of a process. The main purpose of the interface is to condition the various signals received from or sent to external field devices. These field devices may be discrete or analog input/output devices, such as limit switches, pressure transducers, push buttons, motor starters, solenoids, etc.

The I/O interfaces provide the connection between the CPU and the information providers (inputs) and controllable devices (outputs). Incoming signals from sensors (e.g., push buttons, limit switches, analog sensors, selector switches, and thumbwheel switches) are wired to terminals on the input interfaces. Devices that will be controlled, like motor starters, solenoid valves, pilot lights, and position valves, are connected to the terminals of the output interfaces [2].

2.2.6 Programming Device

A programming device can be a handheld device, a desktop console, or a computer. Only when the program has been designed on the programming device and is ready it is transferred to the memory unit of the PLC.

- Handheld programming devices will normally contain enough memory to allow the unit to retain programs while being carried from one place to another.
- Desktop consoles are likely to have a visual display unit with a full keyboard and screen display.
- Personal computers are widely used for programming PLCs, the software allows users to create, edit, document, store and troubleshoot programs. The programming device is used to enter the required program into the memory of the processor. The program is developed in the programming device and then transferred to the memory unit of the PLC [3].

2.3 Components and Operation of PLC

Input relays, internal utility relays, counters, timers, output relays, and data storage are the associated components of PLC. Internal utility relays in PLCs are integral to optimizing control systems by simulating relay functions within the PLC itself, thus enhancing efficiency and reducing external hardware dependencies. Counters in PLCs are crucial for monitoring and controlling processes by simulating pulse counting operations and software-defined components designed to accurately tally pulses in industrial processes, offering versatile counting modes including up, down, and high-speed capabilities for precise control and monitoring.

Timers in PLCs are essential for controlling timing operations in industrial settings, offering a range of types such as on-delay, off-delay, and retentive options with timing increments from milliseconds to seconds. Output relays physically connect and control external devices by sending on/off signals based on programmed commands. This can be transistors, relays, or triacs depending upon the model chosen. PLCs use registers for temporary data storage, typically for performing calculations or manipulating data within the system. These registers can also retain stored information during power loss, maintaining data integrity. Today, programmable controllers are used for the control and operation of almost any process or production line [3].

2.4 PLC Languages

IEC 61131-3 is the standard for programmable controller languages. The following is a list of programming languages specified by this standard.

- Ladder diagram (LD)
- Sequential function charts (SFC)
- Functional block diagram (FBD)
- Structured text (ST)
- Instruction list (IL)

One of the primary benefits of the standard is that it allows multiple languages to be used within the same programmable controller. This allows engineers to choose the best language for each task, which improves efficiency. The ladder diagram language, known for its clear visual representation like relay logic, is widely used by PLC programmers for its simplicity in handling complex control logic, making operations smoother and maintenance faster in industries [4].

2.4.1 Ladder Logic

Ladder logic is a programming language that represents a program by a graphical diagram based on the circuit diagrams of relay logic hardware. It is primarily used to develop software for PLCs used in industrial control applications. Programs in this language resemble ladders, with two vertical rails and a series of horizontal rungs between them, and hence the name ladder.

Ladder logic is widely used to program PLCs, which require sequential control of a process or manufacturing operation. Ladder logic is useful for simple but critical control systems or for reworking old hardwired relay circuits. As PLCs became more sophisticated, it has also been used in very complex automation systems. Often the ladder logic program is used along with a human machine interface (HMI) program operating on a computer workstation [4].

2.4.2 Sequential Function Charts (SFC)

Sequential function chart (SFC) has been developed to accommodate the programming of more advanced systems. Sequential Function Chart (SFC) is a graphical language used for programming PLCs, defined by the IEC 61131-3 standard. It represents the sequence of operations with steps (actions) and transitions (conditions). Sequential functional chart allows for both parallel and sequential execution of tasks, making it suitable for complex control processes in automation. Its flowchart-like structure helps in easily visualizing and understanding the control logic [4].

2.4.3 Functional Block Diagram (FBD)

Functional block diagram is another graphical programming language. A function block diagram represents functions with blocks connected by lines. Inputs and outputs are linked to blocks, with outputs from one block serving as inputs to another. Connections, which carry data from left to right, must match in type at both ends. Multiple right connections, or divergences, allow one output to connect to several inputs. The main concept is the data-flow that starts from inputs and passes in block and generates the output [4].

2.4.4 Structured Text (ST)

Structured text (ST) is one of the five languages supported by the IEC 61131-3 standard, designed for programming PLCs. Structured text is a high-level language that is block structured and syntactically resembles Pascal is a structured literal programming language. Structured text allows the definition of variables and function calls using common elements, enabling the integration of different languages within the same program. Structured text supports complex statements and nested instructions, making it a versatile tool for industrial automation. It is very flexible and intuitive for writing control algorithms [4].

2.4.5 Instruction List

Instruction lists are also designed for PLCs. It is a low-level language and resembles assembly. The variables and function call are defined by the common elements so different languages can be used in the same program. Program control (control flow) is achieved by jump instructions and function calls (subroutines with optional parameters) [4].

2.5 Advantages and Disadvantages of PLC

The advantages of PLC are:

- The wiring of the system usually reduces by 80% compared to conventional relay control system.

- The power consumption is greatly reduced as PLC consumes much less power.
- The PLC self-diagnostic functions enable easy and fast troubleshooting of the system.
- In PLC system spare parts for relays and hardware times are greatly reduced as compared to conventional control panel.
- The machine cycle time is improved tremendously due to the speed of PLC operation is a matter of milliseconds. Thus, productivity increases.
- It costs much less compared to conventional system in situation when the number of I/O is very large and control functions are complex.
- The reliability of the PLC is higher than the mechanical relays and timers.
- An immediate printout of the PLC program can be done in minutes. Therefore, hardcopy of documentation can be easily maintained.
- Communications capability, PLC can communicate with other controllers or computer equipment.
- Modification of control sequence or application can easily be done by programming through the console or computer software without changing of I/O wiring, if no additional input or output devices are required.
- Faster response time, PLCs operate in real-time which means that an event taking place in the field will result in an operation or output taking place.
- An immediate printout of the PLC program can be done in minutes. Therefore, hardcopy of documentation can be easily maintained.
- Flexible in programming and one PLC can operate number of machines.

The disadvantages of PLC are:

- Complexity
- Difficulty with changes or replacements.
- When a problem occurs, hold-up time is indefinite, usually long.
- Unemployment increases due to machine replacing human.
- Huge initial investment.
- Dependence on programming
- Fixed circuit operation

2.6 Omron CX-Programmer Software

CX-Programmer is a software program created by Omron specifically designed for programming Omron programmable logic controllers (PLCs). CX-programmer allows users to write instructions and configure the PLC to control devices and machines based on specific conditions. When creating a new project using the CX programmer, it is essential to understand the basics of PLC programming, including input/output (10) addressing, timers, counters, and other essential instructions. CX-programmer supports various communication protocols, facilitating seamless integration with other devices and systems within an automation environment.

The software also includes comprehensive online monitoring tools, which allow developers to observe the PLC operation in real-time, making it easier to troubleshoot and optimize the system. Users can also make advanced programs by using data blocks of similar-looking data types, also known as Arrays. Meanwhile, different-looking data types are known as Structures. Member symbols of the new User Defined Type can simply be accessed from the program. Symbol creation is also faster since memory allocation and management is automatic. Users will be able to easily monitor every member symbol in watch window, just by using name [4].

2.7 Features of CX-Programmer Software

CX-One allows integrated management of support software for OMRON─s PLC components. Installation on only one personal computer allows a user to handle support software for OMRON products. Only one licensing key is required to install all support software.

It allows integrated management of one save location for files created by support software. Support software dedicated to CPU bus Units and special I/O units can be started on the I/O Table. The appropriate dedicated support software can be automatically started by specifying a registered Unit in the I/O Table (Unit configuration table attached to a PLC). In addition, setup information such as PLC model can be passed to the dedicated support software at start-up, allowing easier switching between Support Software.

The following functions are available by the introduction of the ID information file (CPS) for OMRON Components. Setting up of CPU bus units and special I/O units without manual configuration or address recognition typically involves using automated tools or software (Parameter and selection item names as well as available range of setup are automatically). CPU bus units and special I/O units setting on personal computer and data on actual PLC (CPU Unit) can be verified online, and unmatched item/readout data is displayed graphically. Unit configuration is displayed on the I/O Table based on unit model. Device type on the network can be checked for its Unit model, allowing exact verification of network configuration [4].

2.8 Installation of CX-Programmer Software

CX-programmer software is inserted into personal computer─s CD-ROM drive. Setup Language dialog box is chosen in next step. CX-Programmer splash screen is displayed, then CX-programmer installation is started.

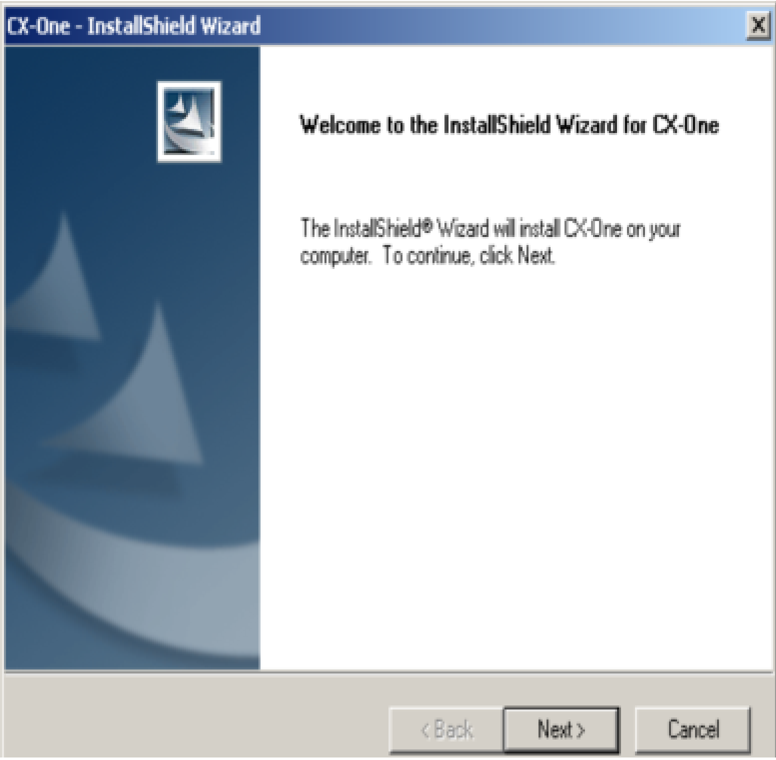


Figure 2.5 Installation of CX-Programmer

After installation is finished, the PLC program is written in CX-One programmer. The installation of CX-One programmer is done step-by step procedure. Installation of CX-One programmer is shown in Figure 2.5 [4].

CHAPTER 3

SYSTEM SOFTWARE TOOLS

In this chapter, contents and functions of CX-programmer main window and commonly used input/output and control instructions from this programming software are described. And, the operation summary of the overall system with the system flowchart is also described with the overall system flowchart in this chapter.

3.1 Software Menu of CX-Programmer

CX-programmer is included in CX-One FA (factory automation) Integrated Tool Package. After installation procedure and starting up of CX-Programmer, the initial screen will be displayed. When a new project is opened to create a program, type of PLC device and CPU must be chosen according to the actual PLC which is connected. Figure 3.1 shows the main window of CX-Programmer with respective labels and its each function is explained in Table 3.1.

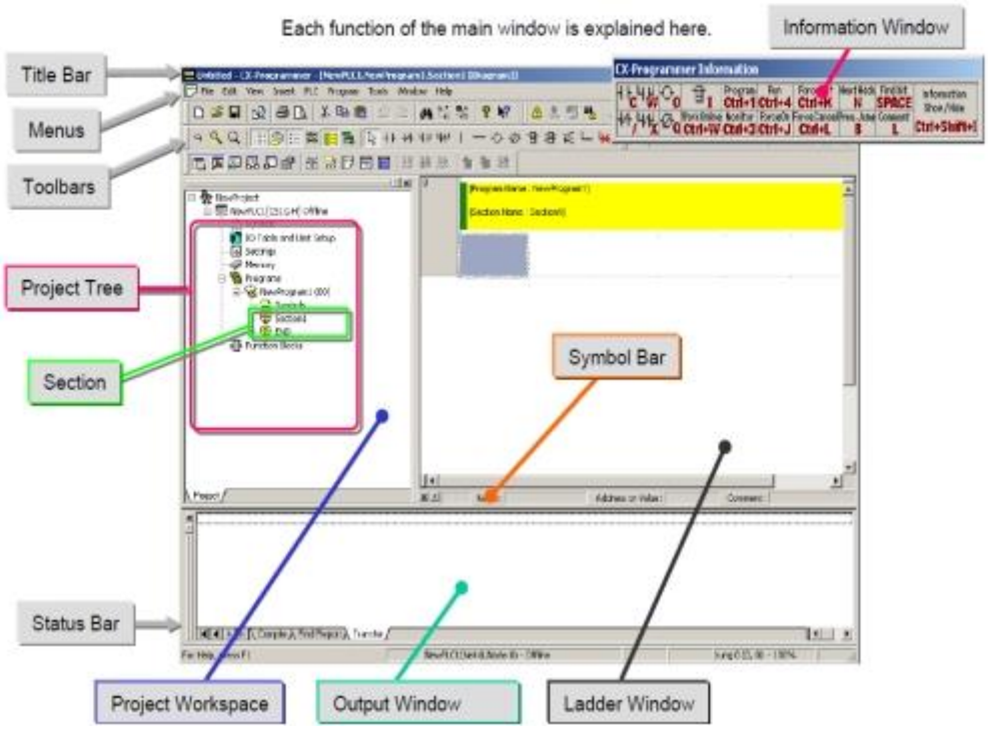


Figure 3.1 Main Window of CX-Programmer

Table 3.1 Contents/Function of Main Window [4]

Name	Contents/Function
Title Bar	Show the file name of save data created in CX-Programmer.
Menus	Enable to select menu items.
Toolbars	Enable to select functions by clicking icons. Select [View] -> [Toolbars], and toolbars can be selected to display. Dragging toolbars enables to change the display positions by the group.
Section	Enables to divide one program into a given number of blocks. Each can be created and displayed.
Project Workspace Project Tree	Controls programs and data. Enables to copy data by the element by executing drag and drop between different projects or within a project.
Ladder Window	A screen for creating and editing a ladder program.
Output Window	• Shows error information in compiling (error check). • Shows the results of searching for contacts/coils in the list form. • Shows error details when loading a project file.
Status Bar	Shows information such as a PLC name, online/offline, location of an active cell.
Information Window	Displays a small window to show the basic shortcut keys used in CX-programmer. Select [View] -> [Information Window] to show or hide the information window.
Symbol Bar	Displays the name, address or value, and comment of the symbol presently selected by the cursor.

3.2 Creating the Program

The required steps in creating a new program are described completely.

At first, a project can be created to display the New Project Screen as shown in Figure 3.2.

- ‘New’ icon from ‘Project Menu’ is selected and “New” is clicked:
- Click ‘New’

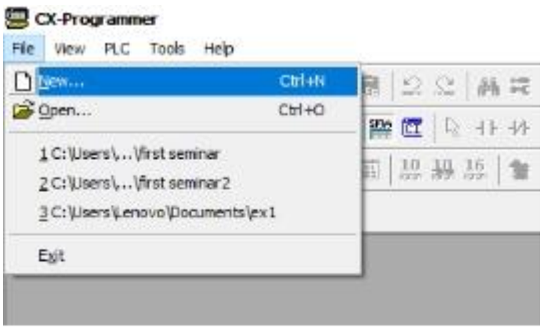


Figure 3.2 Creating a Project

The ‘Device type’ ‘Network Type’ ‘CPU type’ from the list boxes can be selected for the new project. In Figure 3.3 shows ‘Setting’ conditions, after clicking the button, “Setting” is selected.

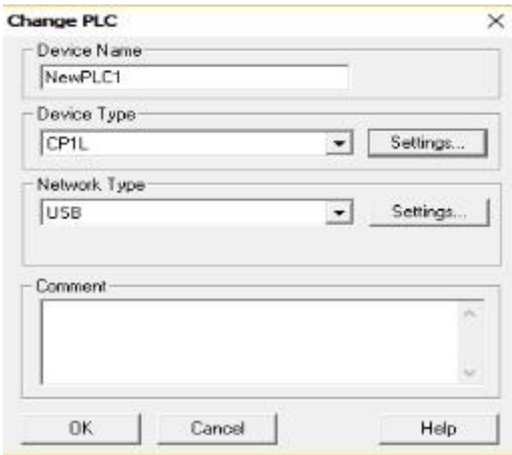


Figure 3.3 Selecting Setting

After creating a new project using CX-Programmer, the following window will be appeared as shown in Figure 3.4.

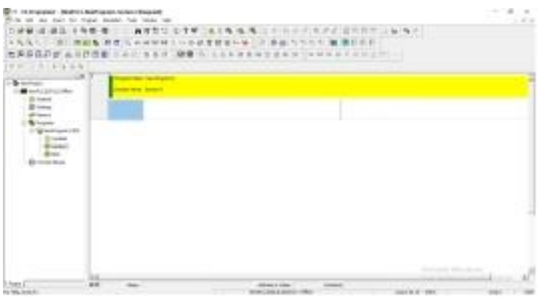


Figure 3.4 Creating a New Window

3.3 Offline Simulation

When the program is written, click the simulation on the menu bar and select the work online simulator to run the program. Performance of user program must be tested and monitored during the simulation. Finally, the simulation must be stopped and the project must be saved. In this method, the program output can be checked and monitored. The program can be easily corrected before connecting program to PLC hardware. Figure 3.5 shows the building a program and Figure 3.6 shows the simulation of the program.

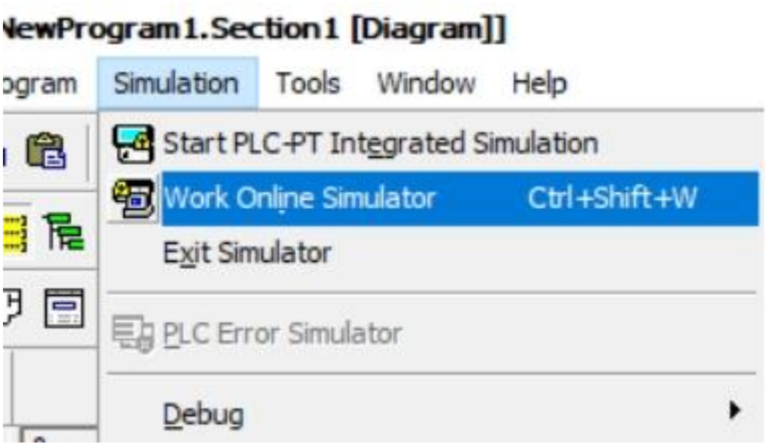


Figure 3.5 Building a Program

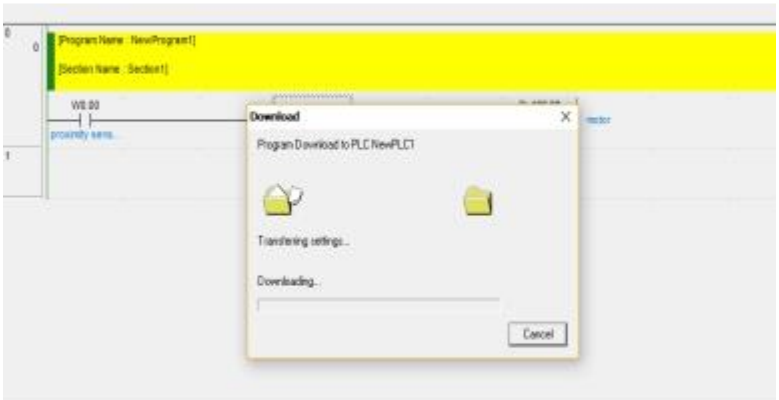


Figure 3.6 Simulation the Program

3.4 Ladder Instructions of CX-Programmer

There are various ladder instructions in this programming software with respective groups like input/output, control, timer and counter, data movement, comparison, increment/decrement instructions, and so on. Among them, instructions used in this research and commonly used instructions will be briefly described.

3.4.1 Timers (TIM)

Timers (TIM) operates a decrementing timer with units of 0.1s, normally used for time delay. It can be ON delay, OFF delay, etc. Timer instruction requires a timer number (N) and a set value (SV) ranging from 0000 to 9999 (0 to 999.9 seconds).

Figure 3.7 shows the timing diagram of timer (TIM) in different conditions. When the timer input is OFF, the timer specified by N is reset, i.e., the timer's PV is reset to the SV and its Completion Flag is turned OFF. When the timer input goes from OFF to ON, TIM starts decrementing the PV. The PV will continue timing down as long as the timer input remains ON and the timer's Completion Flag will be turned ON when the PV reaches 0. The status of the timer's PV and Completion Flag will be maintained after the timer times out.

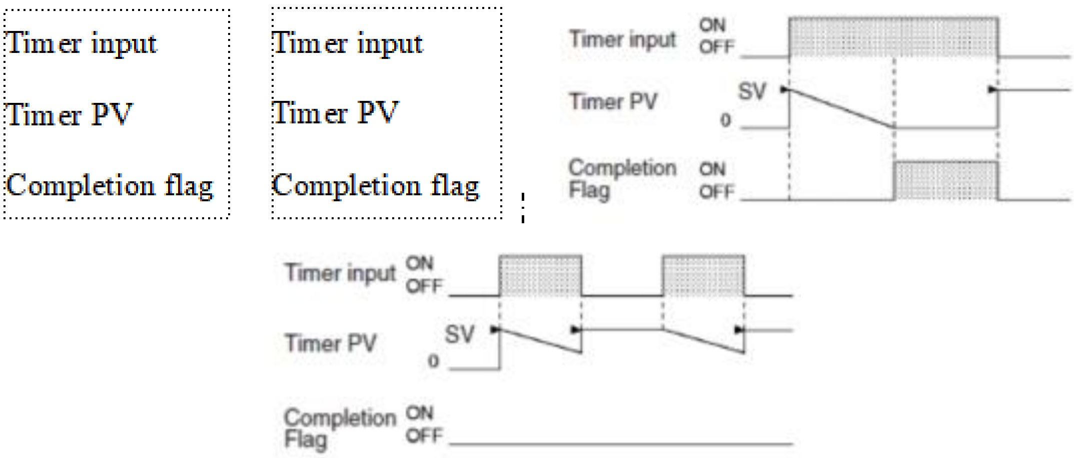


Figure 3.7 Timing Diagram of Timer (TIM)

3.4.1.1 ON delay circuit

ON delay circuit is a type of delay circuit that introduces a time delay between the application of an input signal and the activation of an output. Figure 3.8 shows the ladder and timing diagram of ON delay circuit.

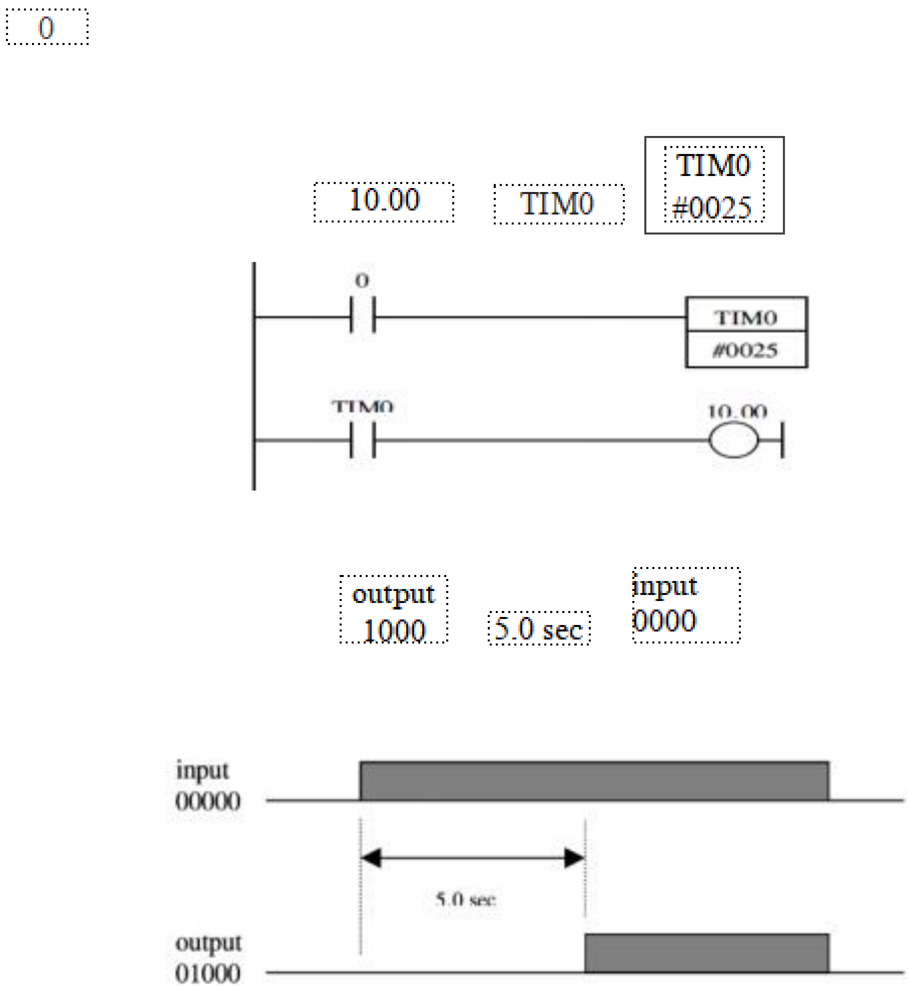


Figure 3.8 Ladder and Timing Diagram of ON Delay Circuit

3.4.1.2 ON and OFF delay circuit

The circuit is used to delay the ON/OFF timer of an input signal for a given time. Timing diagram of ON and OFF delay circuit is shown in Figure 3.9.

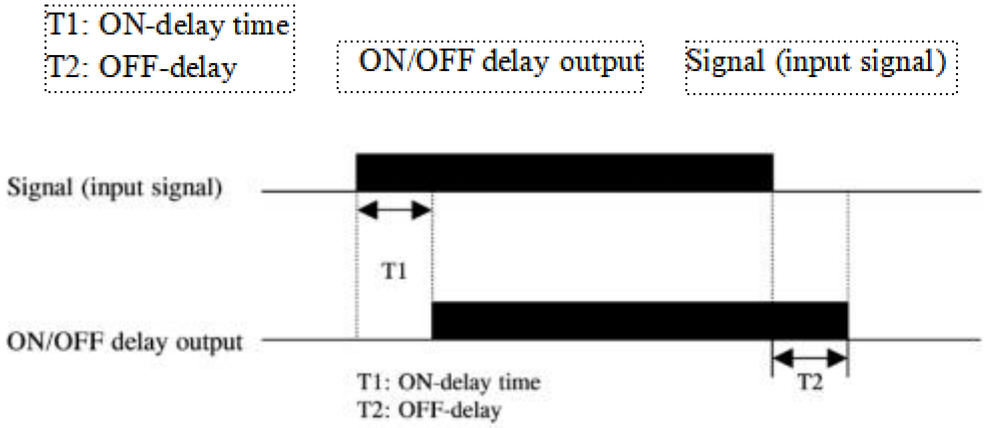


Figure 3.9 Timing Diagram of ON and OFF Delay Circuit

Both circuits use timers to manage delays, and components like relays or timer modules are used to adjust the delay duration. Ladder diagram of ON and OFF delay circuit is shown in Figure 3.10.

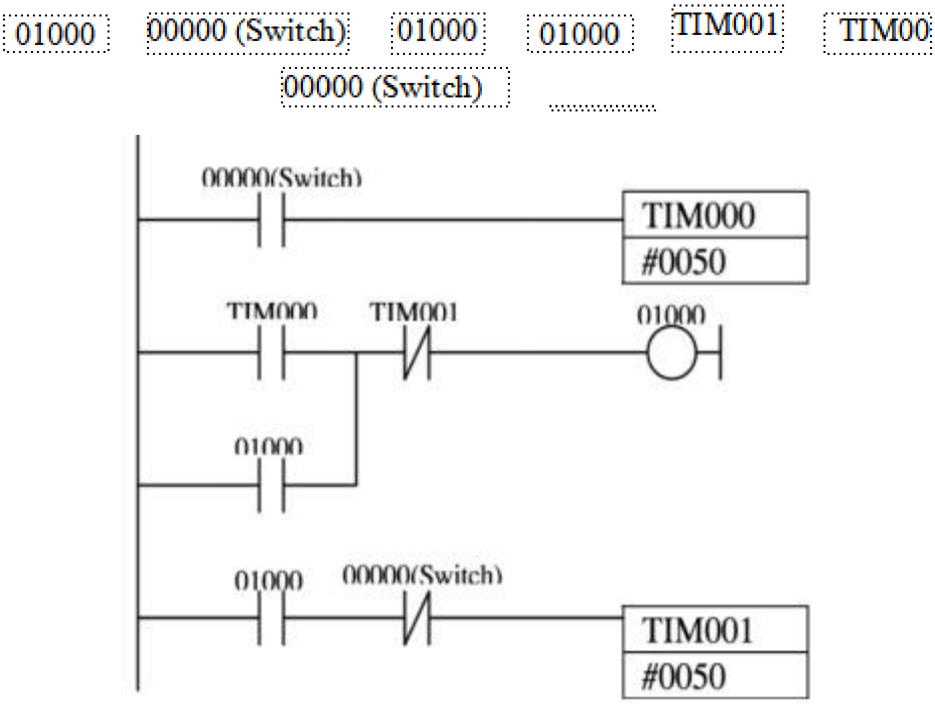


Figure 3.10 Ladder Diagram of ON and OFF Delay Circuit

3.4.1.3 OFF delay circuit

OFF delay circuit is a type of timing circuit where the output remains ON for a specific period after the input signal is removed. It is commonly used in automation systems where it is essential to maintain an output state even after the input signal disappears. Figure 3.11 shows the ladder diagram of OFF delay circuit.

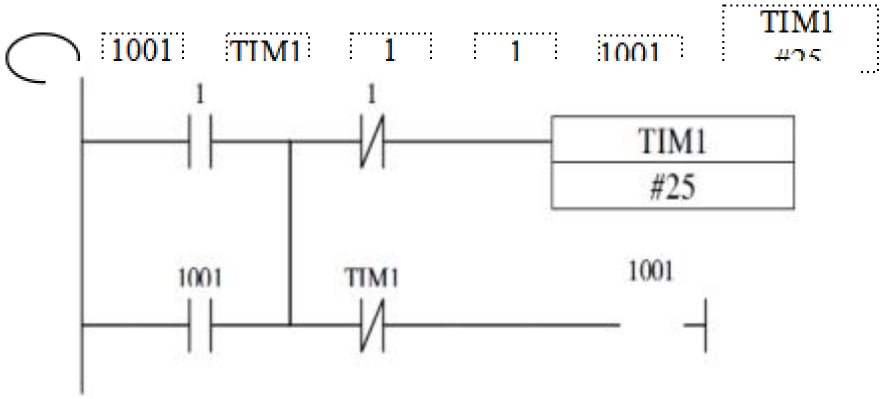


Figure 3.11

Ladder Diagram of OFF Delay Circuit

3.4.2 Counter (CNT)

Counter (CNT) is a preset decrement counter. It decrements one count every time an input signal goes from OFF to ON. The counter must be programmed with a count input, a reset value, a counter number and a set value (SV) can arrange from 0000 to 9999. Figure 3.12 shows the ladder diagram of counter.

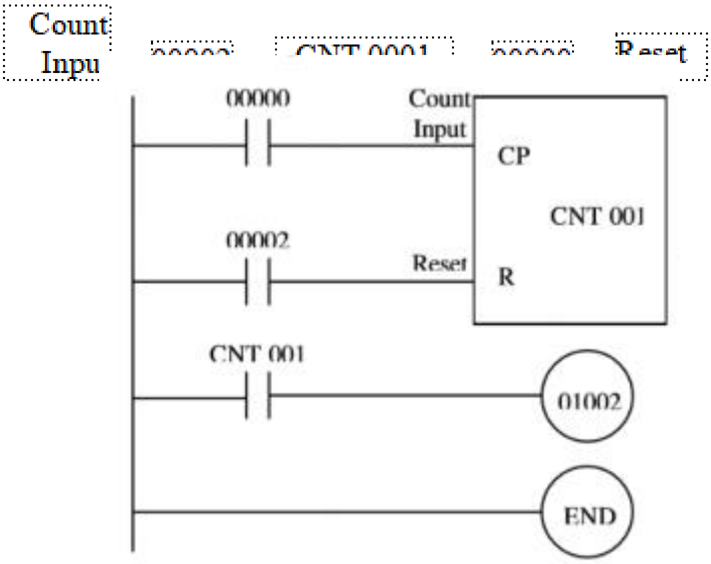


Figure 3.12 Ladder Diagram of Counter

The counter PV (preset value) is decremented by 1 every time that the count input goes from OFF to ON. The completion flag is turned ON when the PV reaches 0. Once the completion flag is turned ON, reset the counter by turning the reset input ON or by using the CNR (545) instruction. Otherwise, the counter cannot be restarted. The counter is reset and the count input is ignored when the reset input is ON. When a counter is reset, its PV is reset to the SV and the Completion Flag is turned OFF. Timing diagram of a counter is shown in Figure 3.13.

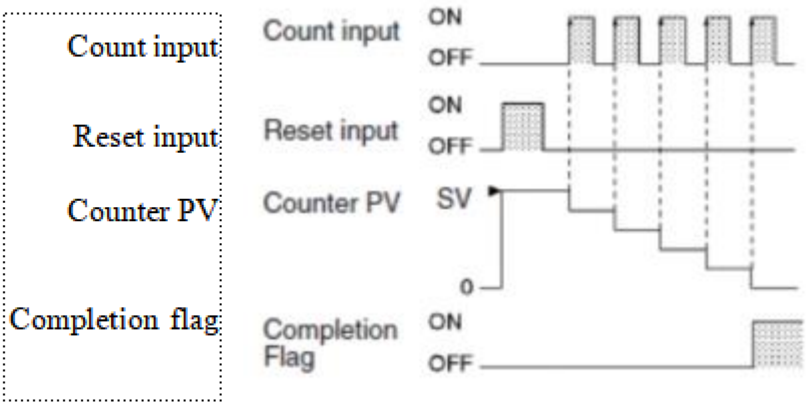
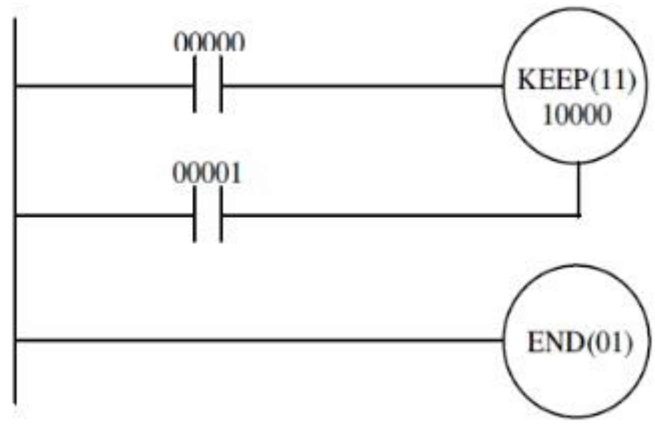


Figure 3.13 Timing Diagram of Counter

3.4.3 KEEP (Latching Relay)

KEEP is used as a latch. It maintains an ON or OFF state of a bit until one of its two inputs sets or resets it. If KEEP is used together with a HR relay, the state of the latched output is retained even during a power failure. It has two inputs; the top one is a SET input, and the bottom one is a RESET input. When the top execution condition is true, it sets the output bit to true, if the bottom is true the output is set to false. If neither is ON, the output retains its value from the last PLC scan cycle. Turning on both inputs will turn the output off. Figure 3.14 shows the ladder diagram of KEEP.

00000



00001

Figure 3.14 Ladder Diagram of KEEP

When S turns ON, the designated bit will go ON and stay ON until reset, regardless of whether S stays ON or goes OFF. When R turns ON, the designated bit will go OFF. The relationship between execution conditions and KEEP bit status is shown below on the right. Figure 3.15 shows the timing diagram of KEEP instruction.

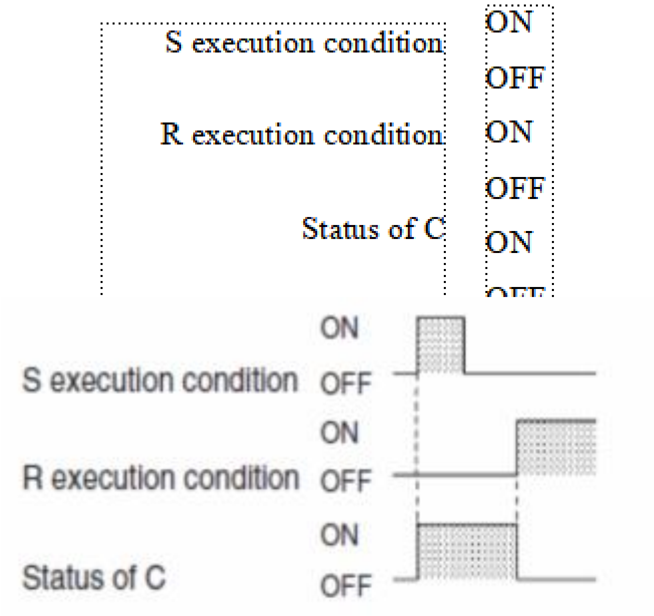


Figure 3.15 Timing Diagram of KEEP

3.4.4 SET and RSET

SET turns the operand bit ON when the execution condition is ON. After this, the specified contact will remain ON regardless of ON/OFF of the input condition. RSET turns the operand bit OFF when the execution condition is ON. After this, the specified contact will remain OFF regardless of ON/OFF of the input condition. Figure 3.16 shows the timing diagram of SET and RSET instructions.

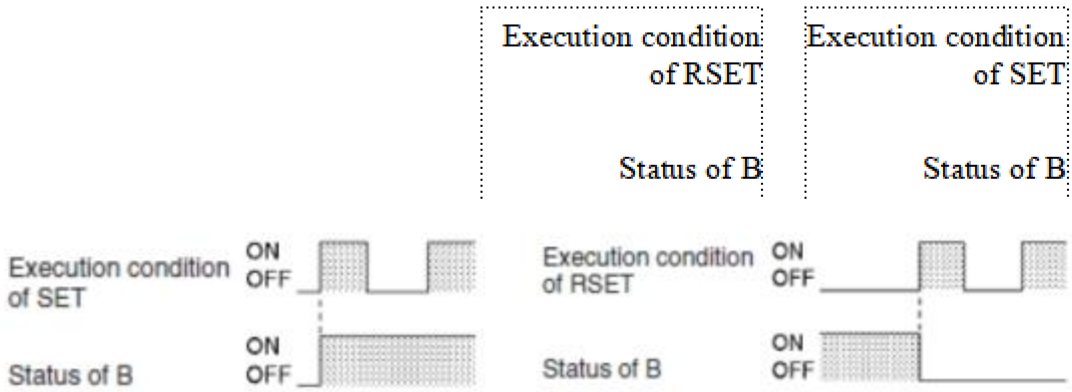


Figure 3.16 Timing Diagram of SET and RSET

3.4.5 DIFU and DIFD (Differentiation)

DIFU and DIFD turn an output ON for one scan only. DIFU turns the designated bit ON for one cycle when the execution condition goes from OFF to ON (rising edge). DIFD turns the designated bit ON for one cycle when the execution condition goes from ON to OFF (falling edge). Figure 3.17 shows the timing diagram of DIFU and DIFD instructions.

Input

DIFU

DIFD

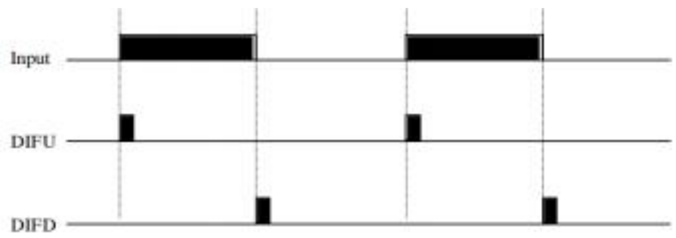
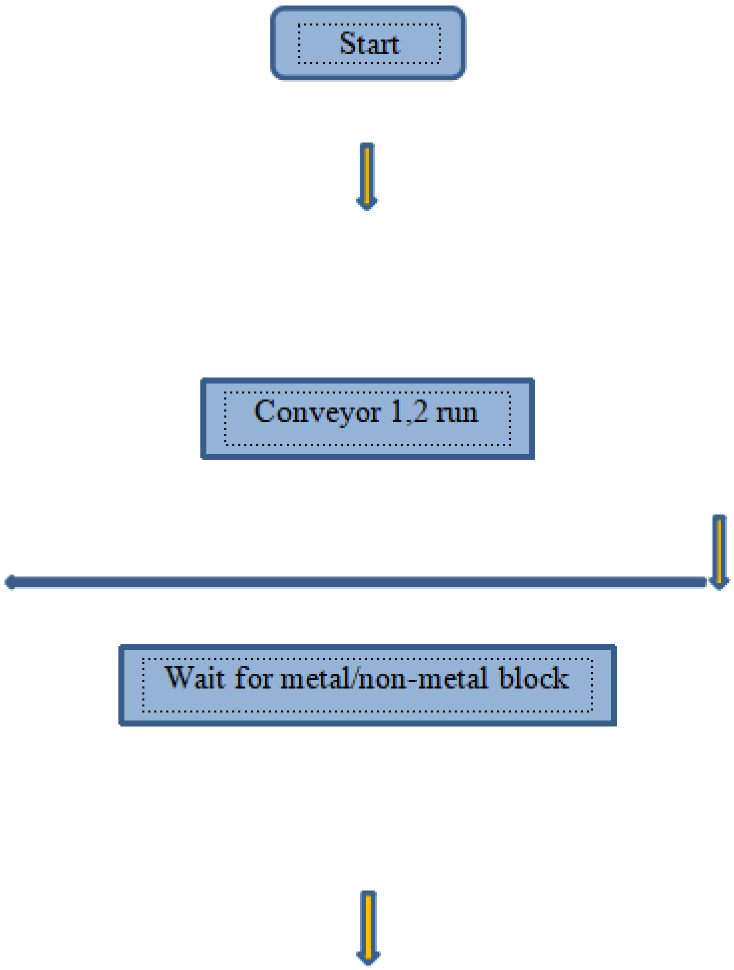
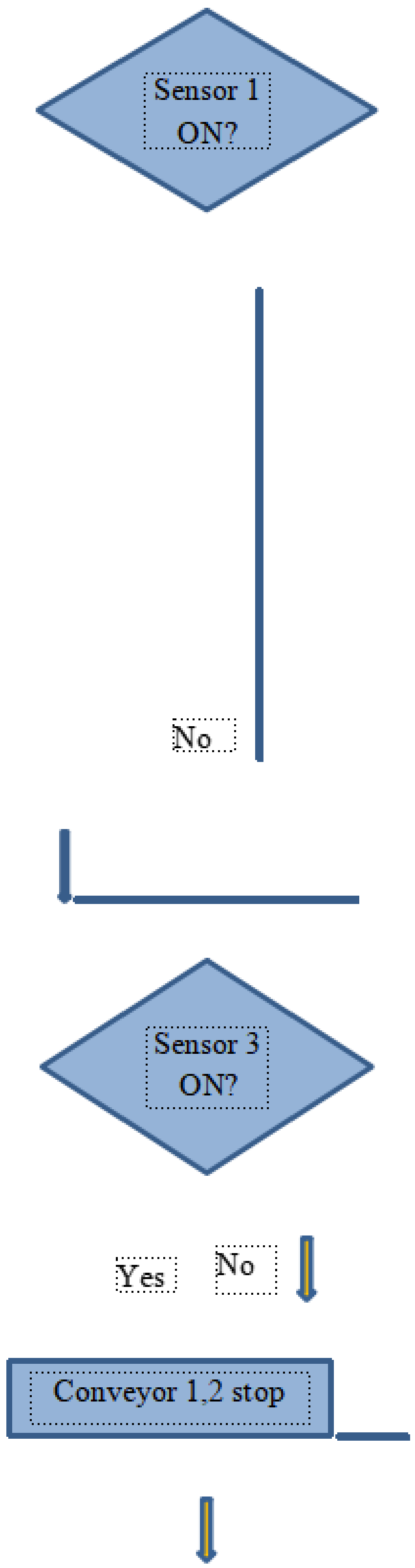


Figure 3.17 Timing Diagram of DIFU and DIFD

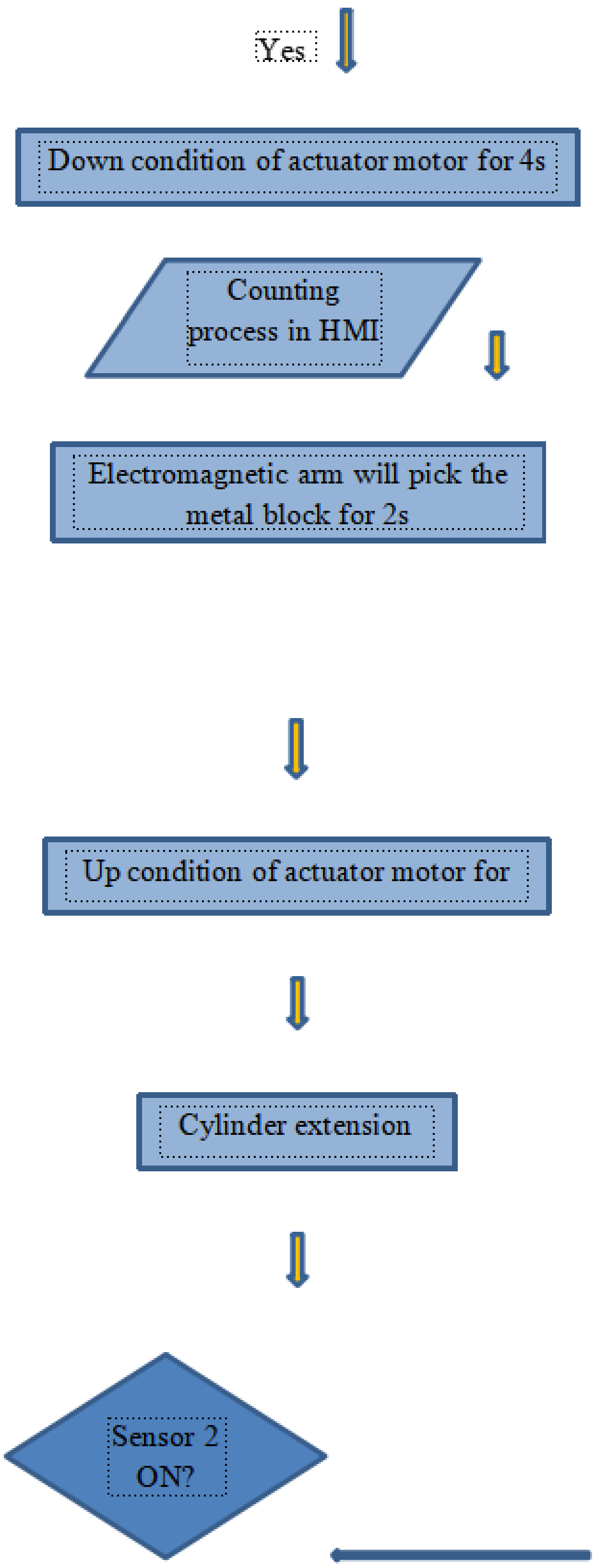
3.5 Flowchart for Automatic Pick and Place Conveyor Transfer System

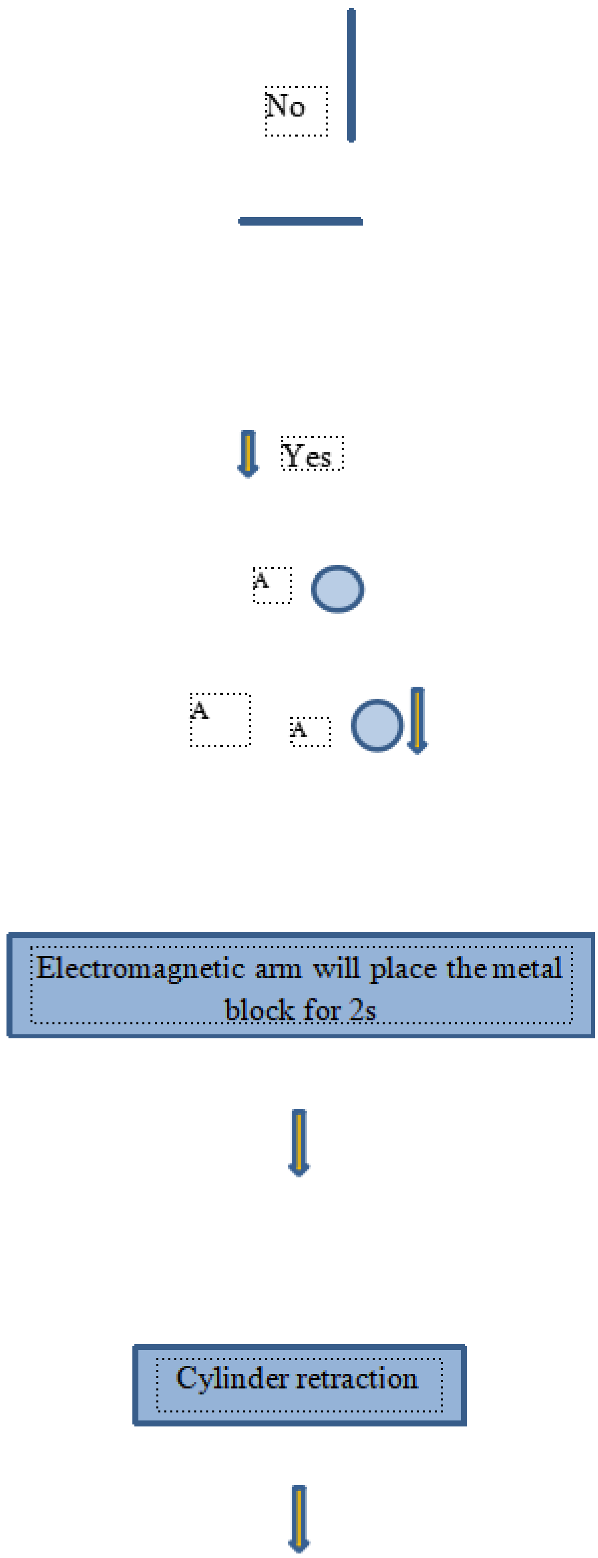
The following Figure 3.16 shows the flowchart of the automatic pick and place conveyor transfer system.





Sensor 1
ON?
No
Sensor 3
ON?
Yes
No
Conveyor 1,2 stop


Yes
Down condition of actuator motor for 4s
Counting process in HMI
Electromagnetic arm will pick the metal block for 2s
Up condition of actuator motor for
Cylinder extension
Sensor 2 ON?


No
Yes
A
A
A
Electromagnetic arm will place the metal block for 2s
Cylinder retraction

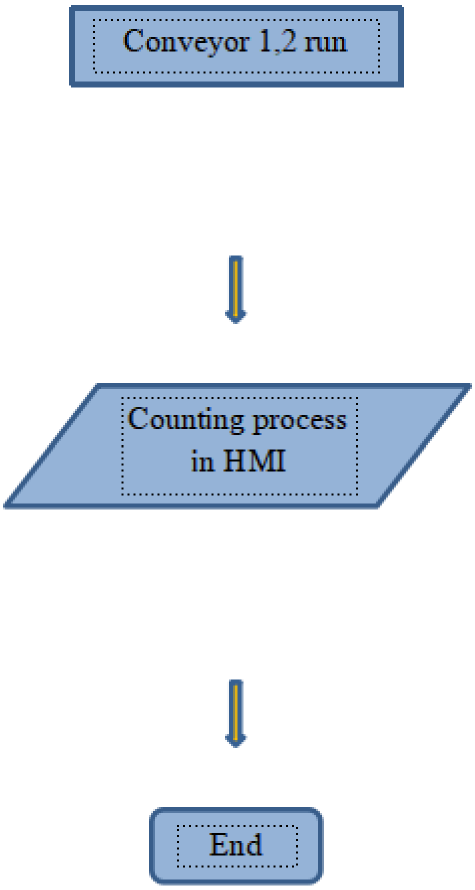


Figure 3.16 Flowchart of the Automatic Pick and Place Coneyor Transfer System

3.6 Operation Summary of Overall System

In this pick and place conveyor transfer system, there are two conveyors, two DC motors, gear motor, two inductive proximity sensors, photoelectric sensor, pneumatic cylinder, electromagnet, keypad HMI, metals and non-metals blocks. When power is applied to this system, the process to carry on is in the ready stage.

As soon as the push button (start) is pressed, the system will start. The two conveyors will run and the blocks are put over the first conveyor. When the inductive proximity sensor 1 detects the metallic box, the two conveyors will stop and the pneumatic arm will work. And then the electromagnetic arm will pick the metallic object and transferring this object to the second conveyor.

When the inductive proximity sensor2 detect the metallic block, the electromagnetic arm will takes off the metallic block and the two conveyors will resume operation. And then there is a photoelectric sensor at the end of the first conveyor to detect the non-metallic blocks. The entire system can be monitored on HMI and the numbers of metals and non-metal objects are expressed on HMI. Human machine interface can also be used to start and stop the system.

CHAPTER 4

TEST AND RESULTS

This chapter includes implementation of the program and test and results of overall operation of the program.

4.1 Implementation of the Program

To implement the program, the principle and operation of the contacts and the functions of basic instructions are needed to know. Fore mostly, basic ladder logic of normally open contact, normally close contact, set coil, reset coil, null, raising trigger, falling trigger, on delay timer, off delay timer, up counter, down counter and latching circuit must be learned and understood. In creating the actual program, input and output addresses are needed to be assigned. Table 4.1 shows the input and output address assignment of the entire automatic pick and place conveyor transfer program.

Table 4.1 Input and Output Address Assignment

Input	Address	Output	Address
Push button(start)	0.00	Conveyor motor 1	100.07
Push button(stop)	0.01	Conveyor motor 2	100.00
Inductive proximity sensor 1	0.02	Actuator motor down	100.05
Inductive proximity sensor 2	0.03	Actuator motor up	100.04
Photoelectric sensor	0.04	Electromagnetic	101.00
–	–	Pneumatic cylinder	100.06

4.1.1 Conveyors Operation

When the start push button (I 0.00) is pressed, the internal relay (w 0.00) is energized and the conveyor motors Q 100.07 and Q 100.00 start running. If the button

(I 0.01) is pressed, internal relay (w 0.00) de-energizes, cutting off the outputs, stopping both conveyors. And internal relay (w 1.00, w 0.13) is used to start and stop the system by monitoring on HMI as shown in Figure 4.1.

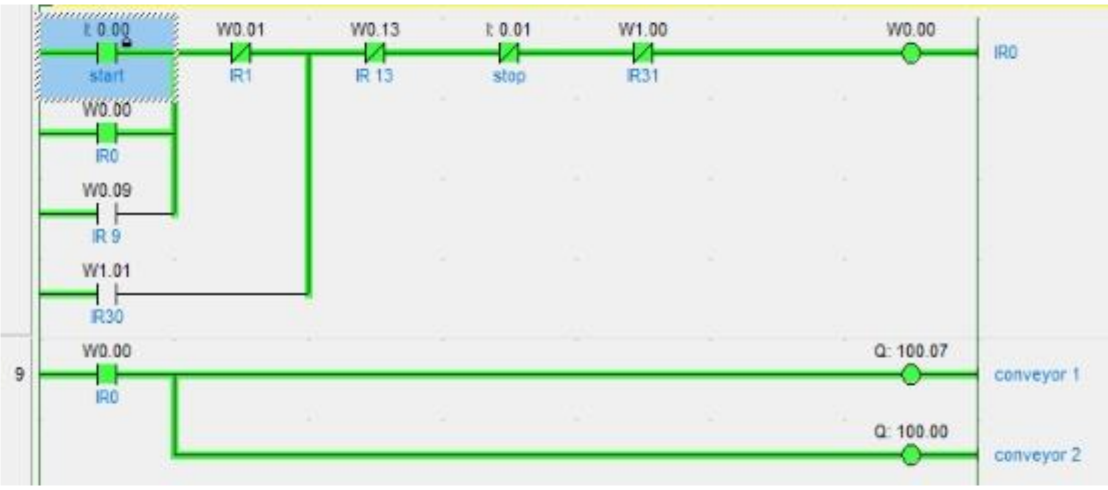


Figure 4.1 Simulation Result of Conveyors Operation

When the proximity sensor 1 (I 0.02) detects a metallic object, it activates IR 1 (w0.01). This suggests that IR 1 is responsible for handling logic related to the detection of objects on the conveyor 1. IR 1 is connected to the proximity sensor 1 input. When the sensor is activated, IR 1 is energized. Both conveyors will stop when IR 1 is activated. This is shown in Figure 4.2.

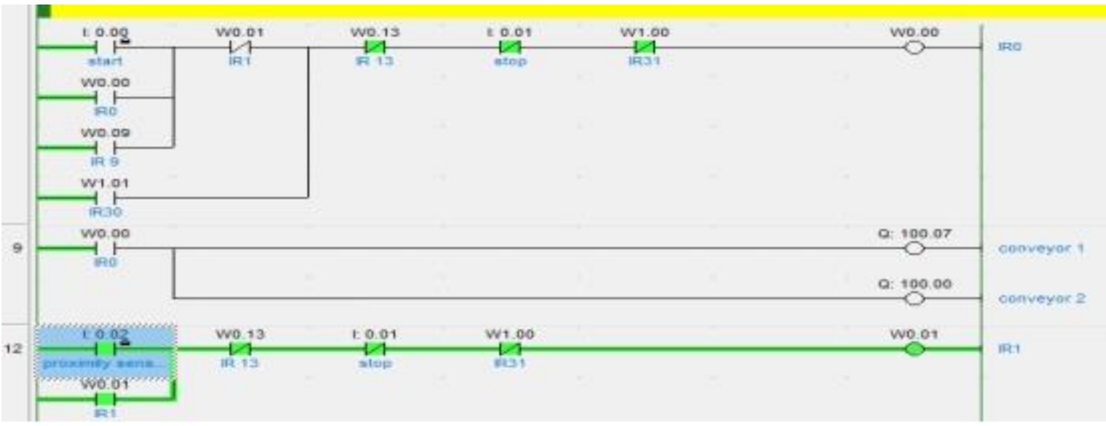


Figure 4.2 Testing Result of Conveyor System for Stop Condition

4.1.2 Down Condition of Actuator Motor

When proximity sensor 1 (I 0.02) is triggered, it closes its contact, which energizes IR1 (w 0.01). If IR 1 (w 0.01) and IR 11 (w 0.11) on the rungs are satisfied, the output (Q 100.04) is activated, driving the actuator motor down. The timer T 0000 ensures that the actuator motor down operates for a specific time (4s). This timer could be controlling how the actuator motor stays engaged or how quickly it needs to respond. This is shown in Figure 4.3.

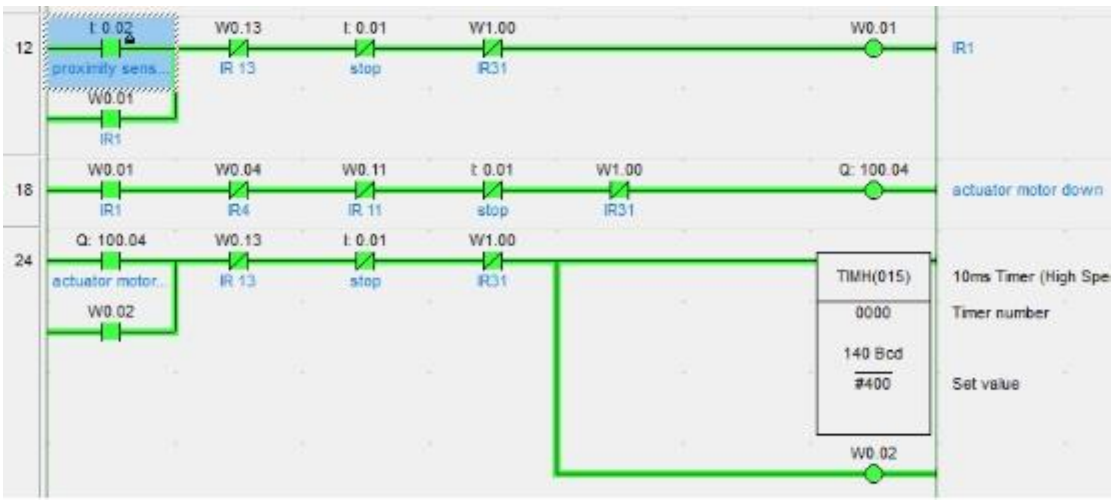


Figure 4.3 Testing Result of Actuator Motor for Down Condition

4.1.3 Operation of Electromagnetic Arm

When the timer T 0000 contact closes, provided its conditions are met. If all relay contacts (w 0.11, w 0.13, w 1.00) are closed and the stop button (I 0.01) is not pressed, the output coil (Q 101.0) will activate, turning on the electromagnet. The timer T 0001 ensures that the electromagnet operates for a specific time (2s). This is shown in Figure 4.4.

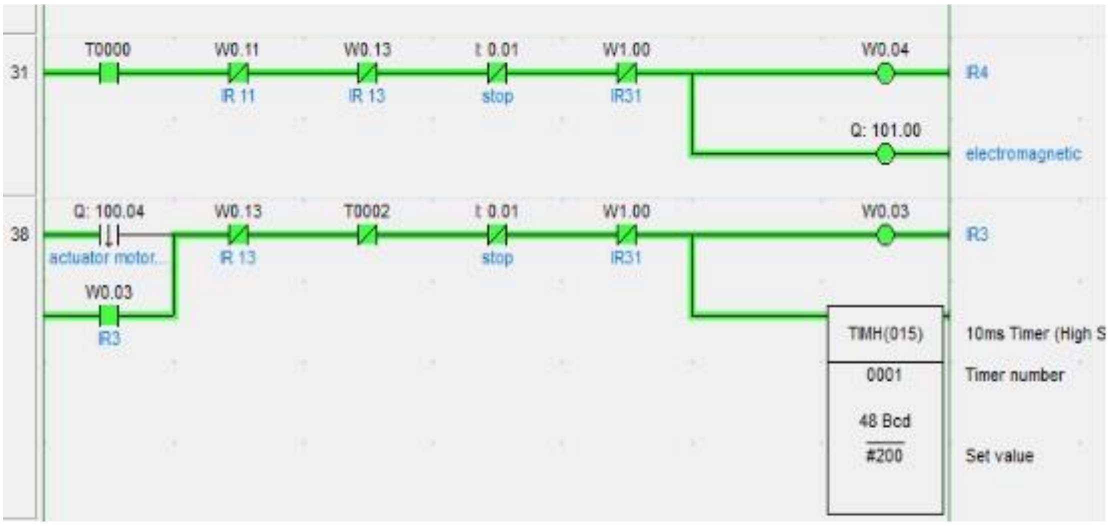


Figure 4.4 Simulation Result of Electromagnetic Arm

4.1.4 Up Condition of Actuator Motor

When the timer T 0001 contact closes, provided its conditions are met. If all relay contacts (w 0.13, I 0.01, w 1.00) are closed, and the stop button (I 0.01) is not presses, the output coil (Q 100.05) will be activated, turning on the actuator motor to move up. When the actuator motor up contact (Q 100.05) and all relay contacts (w 0.13, w0.09, I 0.01, w 1.0000 are closed, the timer T 0002 will activate. The timer T 0002 ensures that the actuator motor up operates for a specific time (4s). This is shown in Figure 4.5.

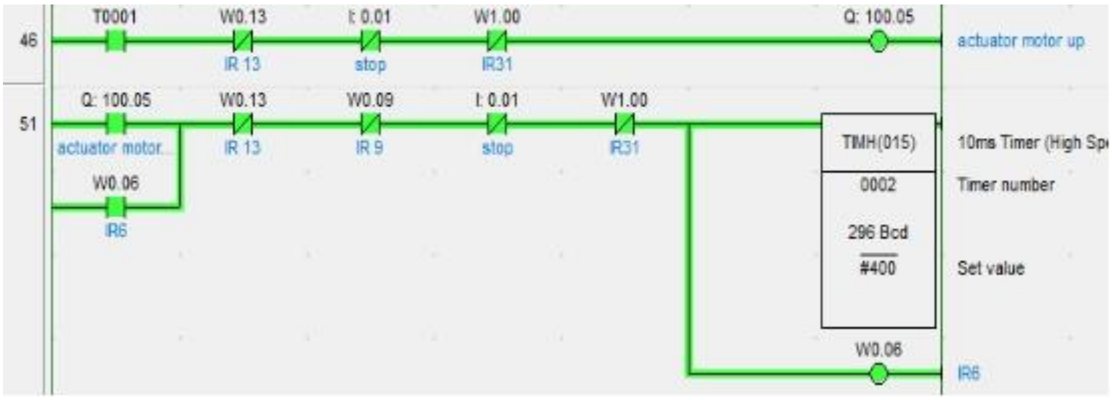


Figure 4.5 Testing Result of Actuator Motor

4.1.5 Operation of Pneumatic Cylinder

The logic starts by checking the condition of T 0002, w 0.09, I 0.01 and w 1.00 in series. If all these conditions are met (T 0002, w 0.09, I 0.01 and w 1.00), it sets output (Q 100.06) to activate the pneumatic cylinder. When the pneumatic cylinder contact (Q 100.06) and all relay contacts (w 0.13, I 0.01 and w 1.000) are closed, the timer T 0003 will activate. The timer T 0003 ensures that the pneumatic cylinder operates for a specific time (8s). This is shown in Figure 4.6.

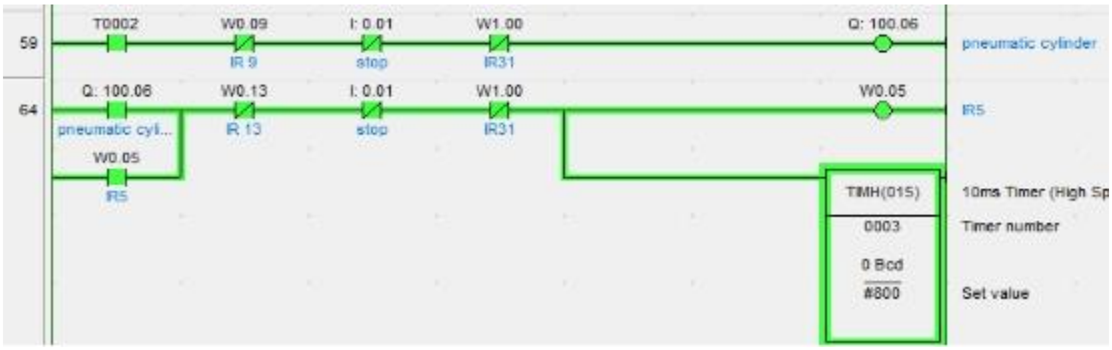


Figure 4.6 Simulation Result of Pneumatic Cylinder

4.1.6 Operation of Placing the Metal Block

When proximity sensor 2 (I 0.03) gets the signal, the internal relay (w 1.00) is activated and the electromagnetic arm will place the metal block on the conveyor 2. This is shown in Figure 4.7.

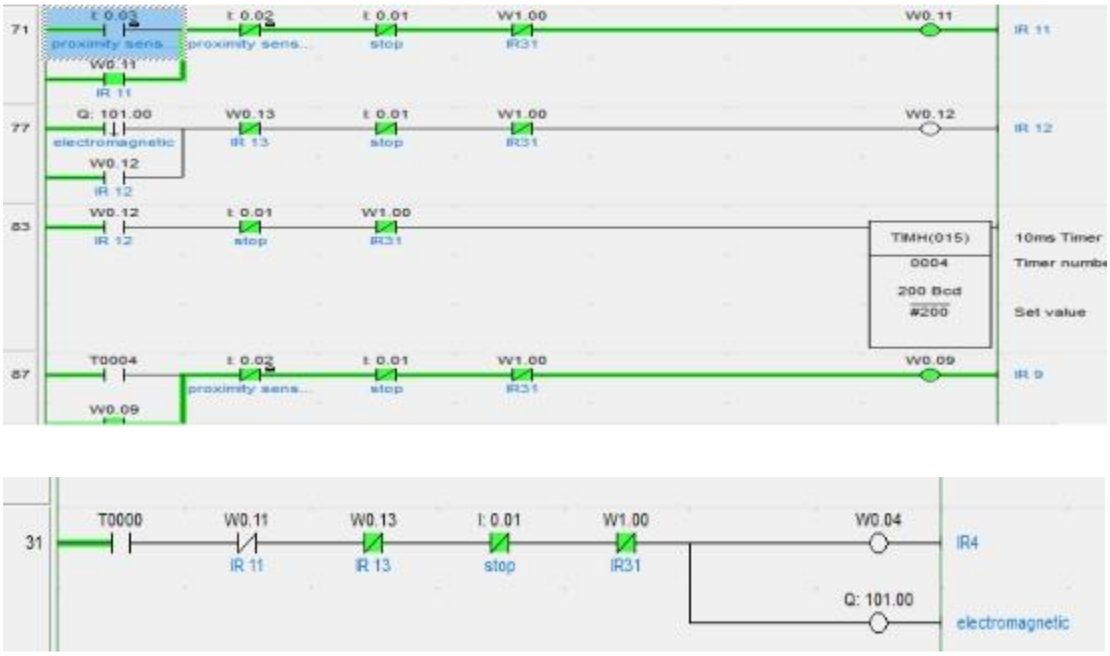


Figure 4.7 Simulation Result of Placing the Metal Block

4.1.7 Looping of the System

The internal relay (IR 12) is activated and cylinder is not running after 2s as shown in Figure 4.8. And then the internal relay (IR 9) is activated and then conveyor motor 1 and 2 will run again. When internal relay (IR 9) gets the signal and all process are completed as shown in Figure 4.9 and Figure 4.10. When the object is put on the conveyor motor 1 (Q 100.07), the system will restart from initial state and this system will operate in this way.

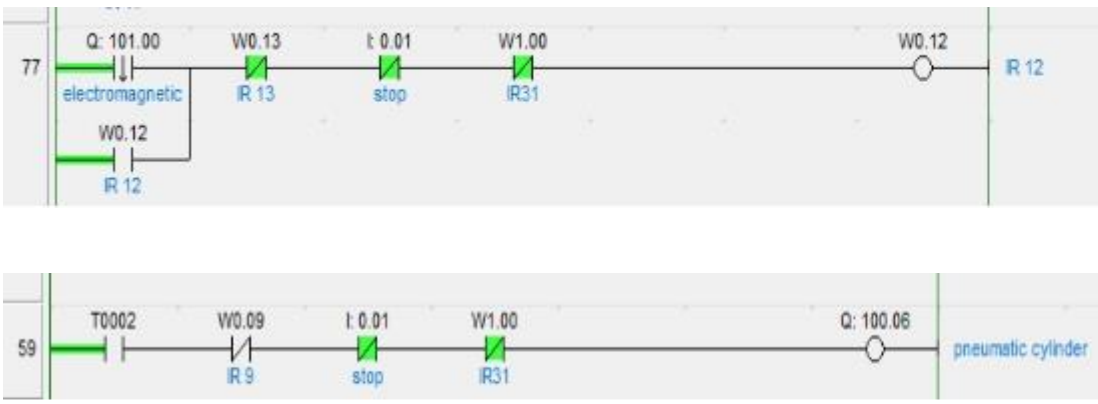


Figure 4.8 Testing Result of Cylinder Retraction

Figure 4.9 Testing Result of Looping State for System

Figure 4.10 Testing Result of Initial State for System

4.1.8 Operation of the Counter

When the proximity sensor 1 (I 0.02) at conveyor motor 1 (Q 100.07) gets the signal, counter 0 (C 0000) will show the amount of metal blocks. This is shown in Figure 4.11.

Figure 4.11 Testing Result of Metal Detection State

When the photoelectric sensor (I 0.04) at conveyor motor 1 (Q 100.07) gets the signal, counter 1 (C 0001) will show the amount of non-metal blocks. This is shown in Figure 4.12.

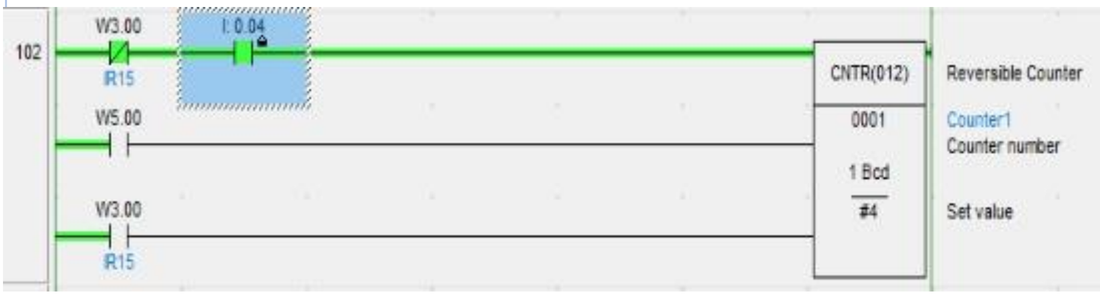


Figure 4.12 Testing Result of Non-metal Detection State

4.2 Algorithm

The program algorithm for the ladder diagram is shown in Table 4.2.

Table 4.2 Algorithm of Ladder Diagram

Step 1	Press the ‘’start’’ button
Step 2	Then the “conveyor 1 and 2” starts running.
Step 3	If the sensor 1 detects the metal block which is in position with the proximity sensor, then the conveyor 1 and 2 will stop.
Step 4	After the conveyor 1 and 2 stop, “actuator motor down” is working about 4s.

Step 5	The electromagnetic arm will work about 2s to pick the metal and then actuator motor up will work to lift the metal about 4s. Now, this is the ready stage to place the metal on the conveyor 2.
Step 6	The pneumatic cylinder will work to place metal about 8s.
Step 7	When the proximity sensor 2 get the signal, the electromagnetic arm does not work. After 2s the cylinder return the normal stage and the conveyor 1and 2 will run.
Step 8	When put another block on the conveyor and the process repeats itself from the initial state.
Step 9	When the proximity sensor 1 detect the metal block, the counter will count the number of the metals and show the number of metals on the HMI screen.

Continued;

Step 10	The proximity sensor 3 at the end of the conveyor 1 will get the signal when non-metal reaches in front of the sensor.
Step 11	The counter for non-metal will count the number of non-metal and show the number of non-metals on the HMI screen.
Step 12	When counters will reach preset value, both counters will reset and the counters will reach the initial stage.
Step 13	End the process.

CHAPTER 5

DISCUSSION, CONCLUSION AND FURTHER EXTENSION

This chapter describes the discussion and conclusion of the thesis. Finally, the further extension of the automatic pick and place conveyor transfer system is described.

5.1 Discussion

The automatic pick and place conveyor transfer system is a mechanized process that efficiently moves items from one location to another using conveyor belts and robotic or pneumatic arms. These systems are commonly used in industries such as electronics, automotive, food and beverage, and pharmaceuticals. Automatic pick and place systems represent a critical element in the advancement of industrial automation, driving productivity and enabling innovations across various sectors. As PLC is used in this system, this is more suitable than another controller system.

5.2 Conclusion

The automatic pick and place conveyor transfer system is designed and implemented by using PLC has been successfully designed and simulated by applying all the concept of control system in this research. It is important to work with a reputable manufacturer and installer to ensure that the system is designed and installed correctly. The purpose of implementing the pick and place conveyor transfer system is to reduce the time required to move products from one stage of production to another. In this research, this system is very useful in wide varieties of industries with the help of PLC and HMI.

In conclusion, the automatic pick and place conveyor transfer system has become essential tool for modern industries. In programming design, understandings of the desired control system and how to use the Ladder Diagram to translate the machine sequence of operation are the most important parts, because it has direct effect on the system performance. Omron programming software, CX-programmer is used in this research to implement the control program for the overall system simulation.

5.3 Further Extension

Continued advancements in robotics, sensor technology, and artificial intelligence will further enhance the capabilities of this system. Increased integration of vision systems, collaborative robots, and advanced control algorithms, leading to even greater flexibility, precision, and efficiency can be expected. By integrating these extensions, an automatic pick-and-place conveyor transfer system can become more adaptable, efficient, and capable of meeting the growing demands of modern manufacturing and logistics.