PLC AND SCADA BASED INDUSTRIAL AUTOMATION SYSTEM

A
DISSERTATION
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
CONTROL AND INSTRUMENTATION (2018-2020)

SUBMITTED BY:
BASANT TOMAR
2K18/C&I/06

UNDER THE SUPERVISION OF

PROF. NARENDRA KUMAR (II)

DEPARTMENT OF ELECTRICAL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

AUGUST, 2020

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering), Bawana Road,

Delhi-110042

CANDIDATE'S DECLARATION

I BASANT TOMAR, (2K18/C&I/06) student of M.Tech (Control & Instrumentation), hereby declare that the project Dissertation titled "**PLC AND SCADA BASED INDUSTRIAL AUTOMATION SYSTEM**" which is submitted by me to the Department of Electrical Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Basant Tomar

Place: Delhi

Date: 31/08/2020

(BASANT TOMAR)

ELECTRICAL ENGINEERING DEPARTMENT

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering), Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled "**PLC AND SCADA BASED INDUSTRIAL AUTOMATION SYSTEM**" which is submitted by BASANT TOMAR, whose Roll No. is 2K18/C&I/06, Electrical Engineering Department, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Narendra

Place: Delhi **PROF. NARENDRA KUMAR (II)**

Date: 31/08/2020 (**SUPERVISOR)**

ACKNOWLEDGEMENT

I would like to thank to all people who have helped and inspired me during my dissertation work throughout these years.

I sincerely acknowledge the earnestness and patronage of my guide Prof. Narendra Kumar, Department of Electrical Engineering, Delhi Technological University, New Delhi, for his valuable guidance, support and motivation throughout this project work. The valuable hours of discussion and suggestion that I had with him have undoubtedly helped in supplementing my thoughts in the right direction for attaining the desired objective.

I wish express my love and gratitude to my beloved parents, siblings and friends for their understanding and endless love. Above all, thanks to Almighty for blessing and guiding me throughout my life.

BASANT TOMAR

(2K18/C&I/06)

ABSTRACT

Recently, it can be seen that the implementation of PLC is being massively understood and also being used everywhere in digitally moving world. The applications of PLCs are being implemented or applied in every private or public sector industries. Generally, we know that the Programmable logic controllers are being used in the automated industrial process field is to control any mechanized movements of small-scale machine and heavy or large machines for creation of a proficient production along with accurate and precise processing of signals received.

Here we will discuss the need of automation in industries for better future prospectus of our nation with the implementation of programmable logic controller (PLC) and data acquisition system.

A simulated prototype in software for the usage in every process industry like beverage, food processing industries and even in big pharmaceuticals companies is main purpose here. For getting automation in filling of various compositions of materials into the various types of containers which is to be based upon material type of the container i.e. metallic one or non-metallic ones, or to be based upon different heights.

For example, if we are considering the basis of distinguish over material type i.e. metallic container and non-metallic container, we will be having three different material compositions to be filled into the metallic container and two different material compositions into non-metallic containers. After the implementation of above discussed process, both types of containers will go into the capping unit and after that labelling unit separately. The process will be terminated after the counting of both type containers.

The process as a whole is being divided into two sub processes i.e. monitoring and visualization of the entire process is being carried out in Supervisory Control and Data Acquisition System (SCADA) software from Wonderware INTOUCH and entire process will be controlled with the use of OMRON 24 I/O NX1P2-9024DT1 PLC and simulation in SYSMAC STUDIO software.

TABLE OF CONTENTS

LIST OF FIGURES

ACRONYMS

PLC	Programmable Logic Controller
SCADA	Supervisory Control and Data Acquisition
CPU	Central Processing Unit
VFD	Variable Frequency Drive
PMMC	Permanent Magnet Moving Coil
PC	Personal Computer
LAN	Local Area Network
IPR	Inductive Proximity
LED	Light Emitting Diode

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Industrial automation is the use of control systems, such as computers or robots, and information technologies for handling different processes and machineries in an industry to replace a human being. It is the second step beyond mechanization in the scope of industrialization. The main idea behind the term “Automation” implies the control of industrial processes with very little human intervention [1]. That’s why the companies in the industrial sector are required to adopt flexible, cost effective and efficient processes to be competitive. Previously, the conventional industrial processes were mainly controlled manually by man power which was more time consuming and also expensive. But with the introduction of automation in the industry, all these issues were resolved by enforcing systems like machinery, software [2] etc. that work incorporated to gain more flexibility, more quality products in lesser production time with a great cost-effective manner. Automation also minimizes the probability of frequent errors occurring in a process which in return conserve the energy and resources in batch processes [3].

Industry 4.0 [4], the current trend of automation, has been developed to answer all those challenges which were faced during manual operation like irregular and inconsistence inspections by offering various advantages like interoperability, virtualization, decentralization, real-time data acquisition etc [5]. Although many people in the industrial sector seem very interested by the concept of Industry 4.0, there is not only a single way to describe or define it. It can be seen as a way to combine complex machineries and devices with sensors and software connected together in a network, utilized to control, predict and organize better outcome for business processes.

Industrial automation [6] has always been a hot topic for finding new ways aiming for gaining more productivity in the processes along with reducing the complexities earlier relay logic control (hard wired control) was used in the industries but due to its major drawbacks like complex and frequent wiring, bulky control panels, time consuming logics and uneasy troubleshooting, it has been replaced by PLC [7]. PLC provide reliability, flexibility and ease in programming has removed all those problems that were being faced while using hard wired logic [8].

A programmable logic controller (PLC) is considered as a solid-state device (SSD), that is designed to accomplish work which was previously done by electromechanically relays [9]. A PLC is basically a controller which is based upon ON-OFF logic states i.e. ON-OFF controller. Hence firstly all the sequences of logic states are programmed then they are logically processed by the CPU present in the PLC. All the operations are done in order to control a process, operations of manufacturing equipment in manufacturing units and machinery. So due to these facts PLCs are considered in the computer family. They are the main heart of the commercial and industrial applications processes [10].

Initially the PLC was used to replace relay logic, but its ever-increasing range of function means that it is found in the many and more complex applications. As the structure of a PLC is based on the same principle as those employed in computer architecture, it is capable of performing not only relay switching tasks but also other applications such as counting, calculating comparing and the processing of analog signals [11-13].

PLC was invented in the 60/70's for the automotive manufacturing industry. Since then it has shaped into one of the most versatile and handy tool that is being used for industrial automation [14]. The working and computational knowledge of PLCs and other microprocessor-based control systems are very critical to technical personal who are staying current with technology in industry now a days [15,18].

PLC is the hub of many manufacturing processes. These microprocessor-based units are being used in every process whether it is as simple as boxing machines or bagging equipment to controlling and tacking sophisticated manufacturing processes. They are basically in all new manufacturing, processing and package equipment in one form or another. Because of their immensely used in industries [17], it is becoming more important to learn skills related to these devices.

1.2 AIM OF THE PROJECT

The prime aim or objective of this project is to monitored or visualized an automatically running beverage process industry with the help of a Supervisory Control and Data Acquisition (SCADA) system and the controlling and handling the entire process will be done through an OMRON PLC. In this whole project we are going to use a 24 I/O (NX1P2-9024DT1) PLC from OMRON which will be programmed or simulated in OMRON own Sysmac Studio Software for programming with simple ladder logic coding or programming. Supervisory control SCADA software that we are going to use throughout this project is Wonderware INTOUCH SCADA software for visualizing or monitoring purpose. The SCADA system is complied with its own central processing unit along with the software package that will be used to handle, control and monitor various remote terminal units which are being placed all around the field. SCADA allows data handling and logging, graphical or visual representations, monitoring & correcting alarm system, changing and defining setpoints of the process etc.

The whole project is being converted or sectioned into multiple sections. Initial section focusses primarily over introductionary part of the automation in the industries & automation importance in the private and public sector. The second section is devoted to the beverage process industry automated process with different stages of filling mainly. The later part of the section elaborates an entire design layout of the whole automated container handling process in PLC & SCADA software. The very important fourth part focuses over the process flowing through charts and their explanations. The ending section depicts every simulated result from PLC & SCADA software along with the conclusions steps by steps.

Steps involved in the entire project of the automated beverage industry [19] process:

STEP1: Sortation of containers by the Inductive PR sensor on the basis of differentiating metallic containers with the non-metallic containers.

STEP2: Filling of containers with different materials compositions. Metallic containers will be having three liquids composition with 33% each but the non-metallic containers will be having only two liquid compositions with 33% each.

STEP3: Capping unit will put the caps or lids over each and every type container.

STEP4: Same Labelling unit will apply the labels over containers with distinguished color labels for metallic and non-metallic.

STEP5: Counting process of metallic and non-metallic containers at the end in each and every case.

1.3 DEVELOPMENT OF PROJECT

A. Software used in this project

- SYSMAC Studio software from OMRON for PLC programming and simulation.
- Wonderware INTOUCH software for SCADA as a process visualization.

B. Hardware components used in this project

- Two type of containers i.e. Metallic one and non-metallic one.
- PLC, OMRON NX1P2-9024DT1 as a controller of this project.
- SMPS providing a constant DC power supply of 24V.
- Three visible tanks.
- Three Solenoid valves.
- Unit for capping purpose.
- Unit for Tagging purpose.
- Single Inductive Proximity sensor IPR IME08 from SICK.
- Five Photoelectric Retro-Reflective sensor GL6 FROM SICK.
- 1214S Motor running Drive from CURTIS.

- DC motor of approx. 1KW power.
- Conveyor belt system.

1.4 PERSPECTIVES OF THIS PROJECT

In order to accomplish the desired objectives and purpose of this project, the scope of study and research mainly focuses over the Simulated result, design and hardware development of the prototype for the automated sortation, filling and counting of metallic and non-metallic containers by using PLC as brain and SCADA for monitoring purpose.

CHAPTER 2

PLC AND SCADA

2.1 ABOUT PLC

PLCs were first came into picture in the last of 1960's. The primary reason for designing such a device was eliminating the large cost involved in replacing the complicated relay based machine control systems. Bedford Associates (Bedford, MA) proposed something called a Modular Digital Controller (MODICON) to a major US car manufacturer. Other companies at the time proposed computer based schemes, one of which was based upon the PDP-8. The MODICON 084 brought the world's first PLC into commercial production [9].

When production requirements changed so did the control system. This becomes very expensive when the change is frequent. Since relays are mechanical devices they also have a limited lifetime which required strict adhesion to maintenance schedules. Troubleshooting was also quite tedious when so many relays are involved. Now picture a machine control panel that included many, possibly hundreds or thousands, of individual relays. The size could be mind boggling. How about the complicated initial wiring of so many individual devices! These relays would be individually wired together in a manner that would yield the desired outcome. As can be seen, there were many problems with this relay based design [14].

These "new controllers" also had to be easily programmed by maintenance and plant engineers. The lifetime had to be long and programming changes easily performed. They also had to survive the harsh industrial environment. That's a lot to ask! The answers were to use a programming technique most people were already familiar with and replace mechanical parts with solid-state ones.

In the mid70's the dominant PLC technologies were sequencer state-machines and the bit-slice based CPU. The AMD 2901 and 2903 were quite popular in Modicon and A-B PLCs. Conventional microprocessors lacked the power to quickly solve PLC logic in all but the smallest PLCs. As conventional microprocessors evolved, larger and larger PLCs were being based upon them. However, even today some are still based upon the 2903. (Ref A-B's

PLC-3) Modicon has yet to build a faster PLC than their 984A/B/X which was based upon the 2901.6 Communications abilities began to appear in approximately 1973.

The first such system was Modicon's Modbus. The PLC could now talk to other PLCs and they could be far away from the actual machine they were controlling. They could also now be used to send and receive varying voltages to allow them to enter the analog world. Unfortunately, the lack of standardization coupled with continually changing technology has made PLC communications a nightmare of incompatible protocols and physical networks. Still, it was a great decade for the PLC.

The 80's saw an attempt to standardize communications with General Motor's manufacturing automation protocol (MAP). It was also a time for reducing the size of the PLC and making them software programmable through symbolic programming on personal computers instead of dedicated programming terminals or handheld programmers. Today the world's smallest PLC is about the size of a single control relay.

The 90's have seen a gradual reduction in the introduction of new protocols, and the modernization of the physical layers of some of the more popular protocols that survived the 1980's. The latest standard (IEC 1131-3) has tried to merge plc programming languages under one international standard. We now have PLCs that are programmable in function block diagrams, instruction lists, C and structured text all at the same time! PC's are also being used to replace PLCs in some applications. The original company who commissioned the MODICON 084 has actually switched to a PC based control system.

2.2 APPLICATIONS OF PLC

In today’s modern/advanced era of industries, where automated or self-regulated machines are being used in almost every private or public sector industries [24], PLC is a very good controller to automate the process and is in great demand for these industries.

- PLC is being used in at most & almost all large scale banal automated processes in the substance processing type industries [22,23].

- PLC is being used to elevate/optimize and excel the process for surpass growth and enrichment in productivity.
- Setup performance, precision and accuracy have been increasing undoubtedly and significantly over the past years by using programmable logic controllers (PLCs) instead of earlier hardwired or relay logics.
- Now a days, traffic system controlling is being managed automatically from an alone substation which is happening in many developing and developed countries also accomplished with PLCs.
- PLC is useful in making bending CNCs machines, suspensions designing and tuning, and rotary assembly parts system for car manufacturing industries.
- Hundreds of modern or day to day applications are only being possible by using PLCs in food processing companies like Nestle and beverage industry for sorting, filling, capping, labelling and end counting etc. [25-28] (this project is all about these only).
- It is being used in every robotics company.
- It is in great demand in warehouse management industries for smooth running process.
- Then after completition of wiring a relay logic panel, we need to revert sequence, it will be compulsory to rewire every part inside the panel but no such condition is mandatory in PLC. Similar process will be run for countless time without any issue.
- It is used in making huge gantry system, AMRs, ASRS systems.
- It is used in metro systems, shuttles and mono rails.
- It is being largely used in AGVs and RGVs.
- Most of the relay logic control which were being used in old manufacturing industry units over a large period of time, are being taken over today by PLC programming software which directly results in less hardware failure with ease in wiring [29-32].

2.3 PLC and RELAY (Hard-Wired) CONTROL LOGIC DIFFERENCES

PLC	Relay logic control
Safe, secure and reliable system	Unsafe and unreliable system than PLC.
Simulation and Controlling is implemented with different programming languages so no compulsion of doing re-wiring for each type of logic in any process.	No such provision is there so we have to implement frequent wiring system for each logic type in any process.
Simplified wiring	Complexity in wiring system
Limited time-consuming or time constraint over logics	Logic takes lot of time to implement.
Damage assessment is very convenient.	Damage assessment is very tedious job.
Very Easy and swift to make new changes.	Chaos making process when to make new changes.

2.4 WHAT'S INSIDE A PLC?

Programmable Logic Controller (PLC) is enclosed with mainly these below mentioned units.

1) Power supply – for running of PLC at 24V DC

2) Input unit – at which input perceiving components like sensors, push buttons etc. are connected.

3) Output port – at which the output devices like actuators, valves, coils, motors, VFDs are linked.

4) CPU (central processing unit) – the processor of PLC.

5) Program memory – PM stores, delete, deliver and process or transform the data that is being infused into PLC.

2.5 PLC BLOCK DIAGRAM REPRESENTATION

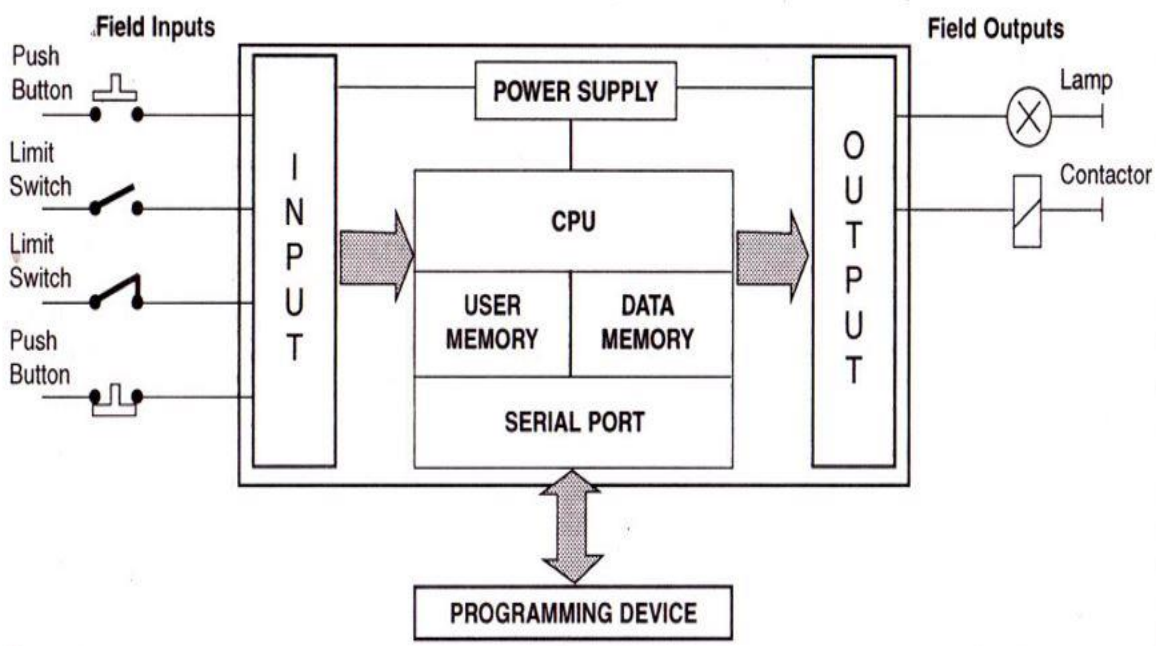


Fig 1: PLC Architecture

2.5.1 SWITCHED MODE POWER SUPPLY (SMPS)

PLC requires 24V DC power supply for its operation. But we do not get this DC voltage everywhere. So, for this purpose we need SMPS.

Two types of SMPS are available in the market. 3-phase SMPS convert given 3phase 440V into 24V DC and 1-phase SMPS convert give 1-phase 230v into 24V DC.

2.5.2 INPUT MODULE OR PORT

We have two types of input signals avail ports or modules are inbuilt or external cards are available for PLCs in the market:

- Digital inputs (0 or 1 signals)
- Analog inputs (4-20 ma or 0-10 V signals)

Digital inputs-They convert all available external binary signal from the process to the internally present digital signal level of programmable controller i.e. PLC.

Some Commonly used Digital Input devices are

- Push buttons (NO & NC type)
- Switches
- Limit Switches (for safety purpose)
- Proximity sensors (Inductive & Capacitive type)
- Photo electric sensors (reflective & diffusive type)

I. **Analog inputs-** Analog inputs cards are being used to convert any available continuous signal with the help of ADC (analog to digital converter) into discrete values for the PLC.

Some Commonly used Analog inputs devices are

- Pressure transmitter (flow measurement)
- Flow transmitter (flow profile and quantity of fluid measurement)
- Level transmitter (with the help of DP & DT cells)
- Load cell (force, torque and strain measurement)

- Thermocouples (temperature measurement)
- Encoders (absolute & incremental type for pulse measurement)

2.5.3 CPU (Central Processing Unit)

CPU system uses microprocessor for the processing. CPU is known as the brain of the computer. CPU has taken over the place of various hard-wired devices such as timers, counters, sequencers and control relays. A processor is being appeared for only only one single time in a PLC and it can be either a one (single) bit operation or a word (16 bits) logic operation. PLCs with word processors are being used when we need to process text and numerical data, calculations of input signals, gauging and scaling of the inputs, controlling and recording of the available data in form of inputs and outputs any other devices present as well as the simple processing of signals in binary codes are being required to processed. The CPU accepts (reads) input data which need to be processed, from various sensing devices (which were mentioned above), executes the stored user program (programming is done any of the softwares like Sysmac Studio, TIA portal etc.), from memory and sends appropriate commands which is based on the execution of the program to control device or in this case, the output devices. A direct current (dc) source is required to produce the controlled power supply or level voltage used by the processor and the inputs and outputs modules. Here that power is supplied with the help of a SMPS (Switched Mode Power Supply).

The SMPS power supply can be

- A separately mounted power supply unit
- OR may be housed in the CPU unit

depending on manufacturer to manufacturer of PLC system. Most of the CPUs are infused with the backup batteries which keep the operating program in storage at the time of a power failure in a plant or NO power supply.

The processor memory module is a major part of the CPU housing. Memory is where the controlling plan or programming of any process is hold or stored in the controller. The information stored in the memory is related to the way on how the input from sensors and output data to the actuators, should be processed. The amount of memory required is directly proportional to the complexity and functionality of the program. Memory elements store individual pieces of information in the form of bits 0s and 1s (for binary digits), bytes and the words for bigger data.

The following functions are simultaneously performed by CPU for processing the available data which is known as SCAN CYCLE of the CPU.

- INPUT STATUS CHECK: It read the inputs.
- PROGRAM EXECUTION: It depends upon the reading executes.
- OUTPUT STATUS UPDATE: It return to values of the output.

The time that is required to complete the very only one cycle is called “CPU scan time”. The CPU scan time is totally dependent upon our complexity of our logic. If the scan time is more the PLC performance decreases. If the scan time is less the plc performance increases. Performance is inversely related to the scan time.

2.5.4 OUTPUT MODULES

Like the input modules, here also we are provided with two types of output modules of PLCs

- Digital outputs
- Analog outputs

I. Digital outputs: These convert the internal signal level of the programmable controller into the binary signal level required externally by the process.

Some Commonly used Digital outputs devices are

- Relays (electromagnet switch for ON/OFF purpose).
- Contractor (operate any system from remote area)
- Values (regulation of flow)

- LEDs (indicators)
- Solenoid value (closely regulate the amount of flow)
- Coilers (motors etc.)

II. **Analog outputs:** Analog outputs cards attached as an peripheral to those PLCs which do not support digital output, converts' digital values or the output from the PLC, back to the continuous signals with the help of a digital to analog converter (DAC).

Some commonly used Analog outputs devices are

- Flow control values
- Pressure control values
- Drive inputs
- Values/actuators
- Analog ports
- VFDs (variable frequency device)

2.5.5 PROGRAM MEMORY

Program memory is broadly classified into three types:

- System Memory
- Load Memory
- Work Memory

1. **System Memory:** In system memory there is a provision of all the details of the configuration of available hardware attached with the PLC.

2. **Load Memory:** Each CPU has its own available internal load memory. The size of this internal load memory is totally dependent on the type of CPU used. This internal load memory can be replaced or bypass by using advanced and sophisticated external memory cards. If there is no memory card inserted, the CPU uses the internal load memory but if a memory card is inserted, the CPU has to use this inserted memory card as load memory. The size of the usable external load memory can never, however, be greater than the internal load memory even if the inserted external SD card has more free space than the internal available memory.

3. **Work Memory:** Work memory is basically a type of non-retentive memory, which is an area for storing elements of the user program that are relevant for program execution. The user program is executed exclusively in this fast work memory and system memory.

2.6 SCADA SYSTEM

SCADA stands for Supervisory Control and Data Acquisition [33]. It requires a computer system for acquiring and analyzing the real scenario time data in industries like waste control, water management, telecommunication, refineries, transportation etc. [34,35].

2.7 SCADA SYSTEM CONTROL STRATEGIES

- **Key Priorities**

1. Safety must be the main concern in any private or public sector industries.
2. Reliability
3. Protection of the process components from damaging either in case of any physical damage or faults.
4. optimization in maximizing the production at lower production cost in a short time frame.

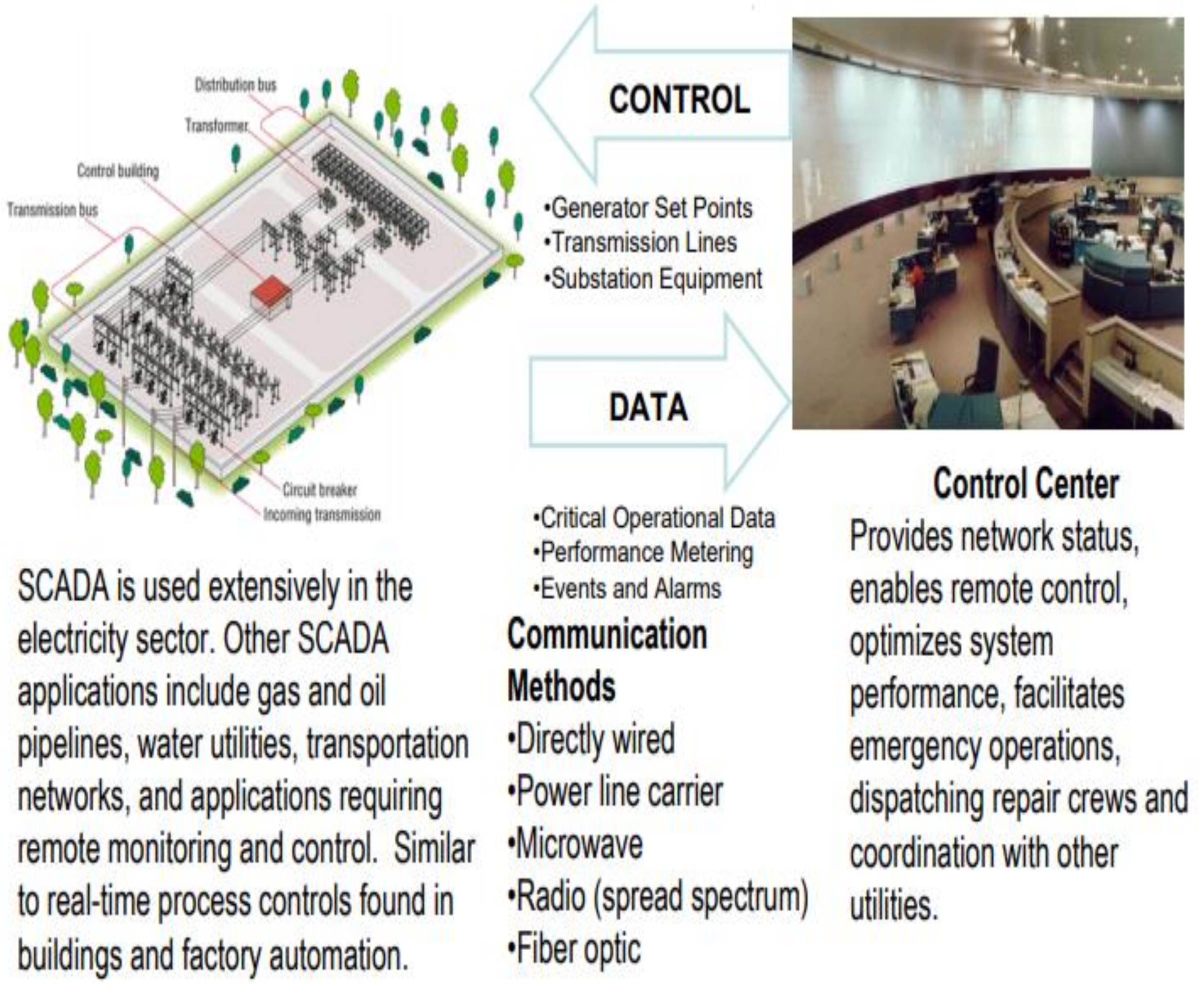


Fig 2: SCADA SYSTEM

- **Localization of substations in Power Plants**

a. Protection of the devices with the help of contactor, MCBs, protective relays.

b. Feedforward & Feedback control strategies with the help of sensing devices and actuating devices.

- **Center areas of control**

a. Control supervision with acquisition of data is the topmost priority.

b. Flaws observation and rectification with the help of observing any system operating boundaries properly.

c. proper coordination between the maintenance activities and in case of any emergency here should be a quick response function.

d. proper balancing between demand and the generation of goods dispatching.

2.8 FUNCTIONALITY OF SCADA SYSTEM

a. **Control Supervision**- Entire control process RTUs located in different sections of the process can be managed remotely with assisting from a common control room.

b. **Acquisition of data**- Field data is acquired by sensors/RTUs and then effective action is taken after processing the data.

c. **Database-** Database in real time is formed which is comprised of all real time data acquired.

d. **User Interface**- All functions like zoom, pan, clutch, declutch etc. are readily present here.

e. **Process Alarm-** Receiving rectifications of alarming stages in the process.

f. **Trending**- With the help of charts, figures, histograms, the decision-making ability & optimization in the process increase.

CHAPTER 3

PROGRAMMING OF PLC

3.1 GENERAL

The user is able to make communication with programmable logic controller (PLC) through various programming devices with the help of a programming language [32,37]. There are several different programming languages used by computer manufacturers and these are connected or able to convey the message to the system through various means of set of instructions with some fundamental controlling plan.

There are some most common types of programming languages available for programming PLC:

- Ladder diagram
- Structured text
- Function block diagram
- The sequential function chart

This project is based upon ladder logic language which is discussed below.

3.2 LADDER LANGUAGE

The ladder language is a symbolic instruction set that is used to create a programmable controller program. It is composed of six categories of instruction that include relay-type, timer/counter, data manipulation, arithmetic, data transfer, and program control. The ladder instruction symbols can be formatted to obtain control logic that is to entered into memory.

The main function of the ladder diagram program is to control outputs based on the input condition. This control is accomplished through the use of what is referred to as a ladder rung.

In general, a rung consists of a set of input conditions represented by relay contact type instruction and an output Instruction at end of the rung represented by the coil symbol. Throughout the section the contact instruction for a rung may be referred to as input conditions rung conditions, or control logic. Coils and contacts are the basic symbols of the ladder diagram instruction set. The contact symbol programmed in a given rung represents conditions to be evaluated in order to determine the control of the output all.

The format of the rung contacts is dependent on the desired control logic. Contacts may be placed in any configuration such as series parallel or series parallel that is required to control a given output for an output to be activated or energized at least one left-to –right path of contacts must be closed. A complete closed path is referred to as having logic continuity. When logic continuity exists in at least one path, it is said that the rung condition is TRUE. The rung condition is FALSE. If no path has continuity.

In the early year, the standard ladder instruction set was limited to performing only relay equivalent functions, using the basic relay-type contact and coil symbols similar to those illustrations.

A need for greater flexibility coupled with developments in technology, led to extended ladder diagram instructions that perform data manipulation, arithmetic and program flow control.

3.3 LADDER PROGRAMMING BIT LOGICS AND COMMON INSTRUCTIONS

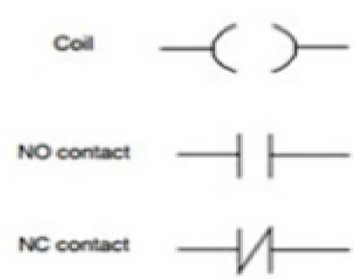


Fig 3: Bits Notation

- **Normally closed (NC) bit:** If the input side has signal state 0 (FALSE), NC contact will remain closed and current signal flows from one side to the other side of the contact at which the output is connected and the output turns HIGH. If the input side has signal state 1 (TRUE), NC contact will get open and current signal will not flow from one side to the other side of the contact at which the output is connected and the output turns LOW. **Placement of NC bit**

  This NC bit can be used at any position in the network logic.

- **Normally Open (NO) bit**: normally open (NO) provides just opposite action of the normally closed (NC) operation. If the input side has signal state 1 (TRUE), N0 contact will remain closed and current signal flows from one side to the other side of the contact at which the output is connected and the output turns HIGH. If the input side has signal state 0 (FALSE), NO contact will get open and current signal will not flow from one side to the other side of the contact at which the output is connected and the output turns LOW. **Placement of NO bit**

  This NO bit can be used at any position in the network logic.

A combining result of NO (normally open) & NC (normally closed) is used to formulate logic gates. This is being shown below in a tabular form.

Table 1: Representing AND logic

Instruction	**Combination of Input Variables**	**Output Result**
AND	Variable A: TRUE (1) Variable B: TRUE (1)	TRUE (1)
	All other cases.	FALSE (0)

Table 2: Representing OR logic

Instruction	**Combination of input variables**	**Output Result**
OR	Variable A: FALSE (0) Variable B: FALSE (0)	FALSE (0)
	All other cases.	TRUE (1)

- **Coil:**

It acts an output that is connected to the NC or NO contact. As soon as the signal is sent from NO or NC the HIGH and LOW of the coil is decided**.**

"Output coil" may be put in any network i.e. after the NC & NO and before and after any timers or counters in the network.

3.3.1 TIMING INSTRUCTIONS (Timers)

- **Timer ON-delay (TON)**: This instruction will turn the Q (output) TRUE (1) as soon as the Preset timing (PT) is carried out after the starting of the timer.

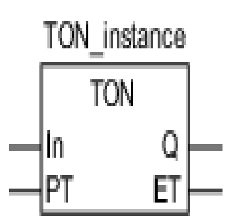
Pictorial representation


Fig 4: Pictorial representation of TON

Table 3: The Action Process of TON

Notation	Full Name	Action process
'In'	Timer input	TRUE (1): Timer gets start FALSE (0): Timer goes reset
'PT'	preset time (I/P)	Time to set to get the timer ON after this time.
'Q'	Output	TRUE (1): output gets HIGH FALSE (0): output gets LOW
'ET'	Accumulated time (O/P)	Total Accumulated time since beginning of timer

Time is in milliseconds by default.

FUNCTIONALITY OF TON

The TON instruction will turn the outputs TRUE (1) after when the Preset time (PT) is carried out after the starting of the timer.

The TON timer gets started after TON input IN turns HIGH. Accumulated time ET is increased as time passes by.

When ET is equal to preset timing PT, the timer TON Q output gets HIGH. ET won't increase after that.

TON timer gets reset as soon as IN turns LOW (0). ET gets LOW 0, and output Q turns to LOW/FALSE (0).

As soon as IN turns LOW/FALSE (0) just afterwards the starting of TON timer, the timer TON is reset and that too ahead of ET = PT.

- **OFF-Delay Timer (TOF):** This instruction will turn the Q (output) FALSE (0) as soon as the Preset timing (PT) is carried out after the starting of the timer

Pictorial representation

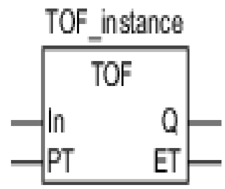


Fig 5: Pictorial representation of TOF

Table 4: The Action Process of TOF

Notation	**Full Name**	**Action Process**
‘PT’	Preset time (I/P)	Time to set to get the timer OFF after this time.
‘In’	Timer input	HIGH/TRUE (1): Timer gets reset LOW/FALSE (0): Timer goes start
‘Q’	Output of timer	TRUE (1): output gets HIGH FALSE (0): output gets LOW
‘ET’	accumulated time (O/P)	Total Accumulated time since beginning of timer

Time is in milliseconds by default.

FUNCTIONALITY OF TOF

The TOF instruction will turn the outputs FALSE (0) after when the Preset time (PT) is carried out after the starting of the timer.

The TOF timer gets started after TOF input IN turns HIGH. Accumulated time ET is increased as time passes by.

When ET is equal to preset time PT, timer TOF Q output gets LOW. ET won’t increase after that.

The timer gets set when IN turns FALSE (0). ET gets LOW 0, output Q turns to HIGH/TRUE (1).

As soon as IN turns LOW/FALSE (0) just after the starting of TOF timer, the timer TON is still set and that too before ET = PT.

- **Accumulation timer (AT):** This instruction sums up entire time period during which its signal input is HIGH/TRUE (1).

Pictorial representation

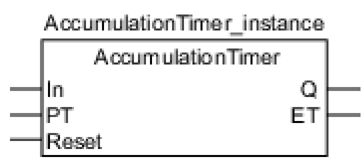


Fig 6: Pictorial representation of AT

Table 5: The Action Process of AT

Notation	Full Name	Action process
‘In’	input of TIMER	HIGH/TRUE (1): AT gets start LOW/FALSE (0): AT goes in stop mode
‘PT’	preset timing (I/P)	Maximum time or Time to set to get the timer OFF after this time
‘Reset’	Reset (I/P)	TRUE (1): Timer gets reset after this. FALSE (0): Timer will never get reset
‘Q’	output of TIMER	TRUE (1): ET is equal to PT. FALSE (0): ET is not equal to PT.
‘ET’	Total accumulated time (O/P)	Total Accumulated time since beginning of timer

Time is in milliseconds by default.

FUNCTIONALITY OF AT

First, reset = FALSE (0), the AT timer switch ON when IN turns TRUE (1). Total Accumulated time since beginning of timer will increase in steps.

The AT timer get switched OFF when IN turns FALSE (0). The accumulated value at ET O/P is freezed.

If IN turns TRUE (1) again, the AT timer resumes. Total Accumulated time since beginning of timer will increase in steps.

When ET = PT, AT timer output Q turns TRUE (1). The accumulated value at ET O/P is freezed now.

The AT timer gets reset when Reset input turns TRUE (1). ET becomes zero (0) and value of the Q output is LOW/FALSE (0).

3.3.2 COUNTING INSTRUCTION (COUNTERS)

They are being used to count the pulses.

- **CTU:** The CTU instruction will add up in the value of counter (CV) as soon as input side signal at input of CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

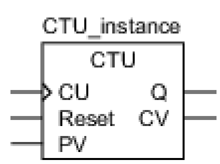
Pictorial representation


Fig 7: Pictorial representation of CTU

Table 6: The Action Process of CTU

Notation	Full name	Action process
'CU'	Input to CTU	Counter input to start the counter
'Reset'	Reset	TRUE (1): Reset the counter and establish CV = 0
'PV'	Preset value	preset value of no of pulses for counter
'Q'	Output of CTU	TRUE (1): output of CTU turns ON/HIGH FALSE (0): output of CTU turns OFF/LOW
'CV'	Counted or accumulated value	Accumulator value

FUNCTIONALITY OF CTU

The CTU instruction will add up in the value of counter (CV) as soon as input side signal at input of CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

As reset input signal Reset turns TRUE (1), accumulator or counted value CV = zero (0) as a result, counter CTU output value Q turns LOW/FALSE (0).

As CTU counter input CU bit turns TRUE (1), the accumulator number of CV is increased. If counted number of PV<=CV, Q turns TRUE/HIGH.

When accumulator value is equal to number of value at preset or even higher than that, the CV value will not increase or decrease even if counter input turns ON/HIGH.

If reset value is turned true then both CU and CV will be ignored.

- **UP-DOWN Counter (CTUD):** The CTUD instruction will decrease or add up in the value of counter (CV) as soon as input side signal at input of CD & CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

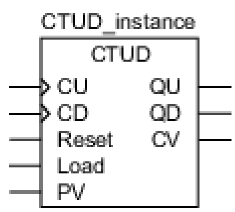
Pictorial representation


Fig 8: Pictorial representation of CTUD

Table 7: The Action Process of CTUD

NOTATION	**FULL NAME**	**ACTION PROCESS**
‘CD’	Input to CTU	Counter input to start the counter
‘CU’	Input to CTU	Counter input to start the counter
‘Load’	Load signal	Set CV=PV
‘Reset’	Reset	TRUE (1): Reset the counter and establish CV = 0
‘PV’	Preset value	Preset (ending and starting) value of no of pulses for counter for up and down counter respectively
‘QD’	Output of CTD	TRUE (1): output of CTD turns ON/HIGH FALSE (0): output of CTD turns OFF/LOW
‘QU’	Output of CTU	TRUE (1): output of CTU turns ON/HIGH FALSE (0): output of CTU turns OFF/LOW
‘CV’	Counted or accumulated value	Accumulator value

FUNCTIONALITY OF CTUD

The CTUD instruction will decrease or add up in the value of counter (CV) as soon as input side signal at input of CD & CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

It performs all the functions of both the counters.

- **CTD (down counter):** The CTU instruction will decrease in the value of counter (CV) as soon as input side signal at input of CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

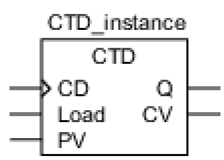
Pictorial representation
 CTD

Fig 9: Pictorial representation of CTD

Table 8: The Action Process of CTD

Notation	**Full name**	**Action process**
'CD'	Input to CTD	Counter input to start the counter
'PV'	Preset value	preset value of no of pulses for counter
'Load'	Load signal	Set CV=PV
'Q'	Output of CTD	TRUE (1): output of CTD turns ON/HIGH FALSE (0): output of CTD turns OFF/LOW
'CV'	Counted or accumulated value	Accumulator value

FUNCTIONALITY OF CTD

The CTD instruction will decrease in the value of counter (CV) as soon as input side signal at input of CU is reached. The value at present (PV) and the value of counter (CV) have to be integers value.

As load input signal set turns TRUE (1), accumulator or counted value CV = zero (0) as a result, counter CTU output value Q turns LOW/FALSE (0).

As CTD counter input CD bit turns TRUE (1), the accumulator number of CV is increased. If counted number of PV<=CV, Q turns TRUE/HIGH.

When accumulator value is equal to number of value at preset or even higher than that, the CV value will not increase or decrease even if counter input turns ON/HIGH.

If load value is turned true then both CU and CV will be ignored.

3.3.3 OUTPUT SEQUENCE INSTRUCTION

SR: Set-Reset (SR) instruction keeps BOOL variable value. Set input will be given priority even if both of the Reset input and Set input get TRUE (1). It keeps the Set priority.

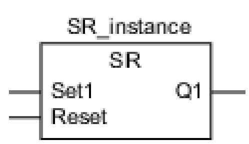
Graphic expression


Fig 10: Graphical expression of SR

Table 9: The Action Process of SR

Bits	ACTION	I/O
"Set1"	input is set	Input signal
"Reset"	Input is reset	Input signal
"Q1"	Output is kept	Output signal

FUNCTIONALITY OF SR

Instruction SR defines output which gives priority only to Set. The below shown table explains the relationship between Set, Reset and Output.

Set1	Reset	Q1
HIGH	HIGH	HIGH
HIGH	LOW	HIGH
LOW	HIGH	LOW
LOW	LOW	No effect

CHAPTER 4

COMMUNICATION

4.1 GENERAL

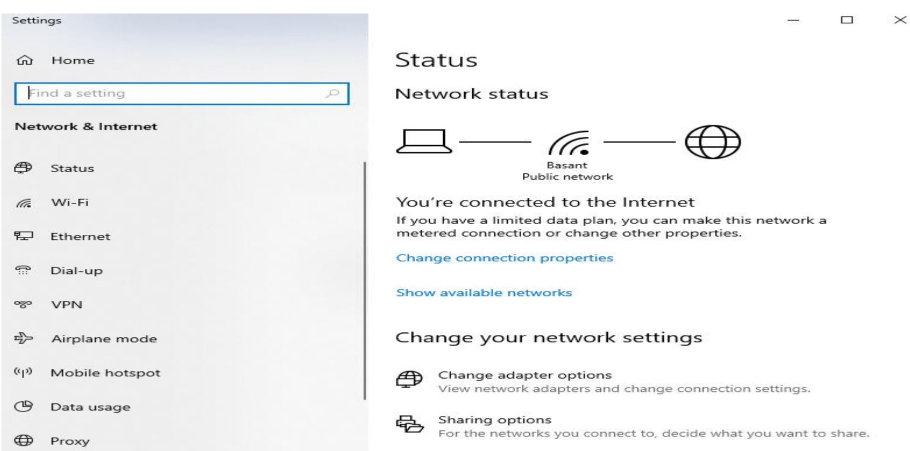
The communication between PLC & PC is required because we need to configure the PLC and other I/O devices and then programmed [39]. For all these purposes a PC must communicate with PLC and other hardware units for configuration.

Here we will discuss communication on the basis of Ethernet/IP protocol where connections can be established through:

- LAN connectivity: here we use Ethernet cable which is directly connected between PLC and PC.
- Wireless connectivity: here PLC is connected to a WIFI device like WLAN-1101 from PHOENIX CONTACT, and then the communication is setup between PC and WLAN.

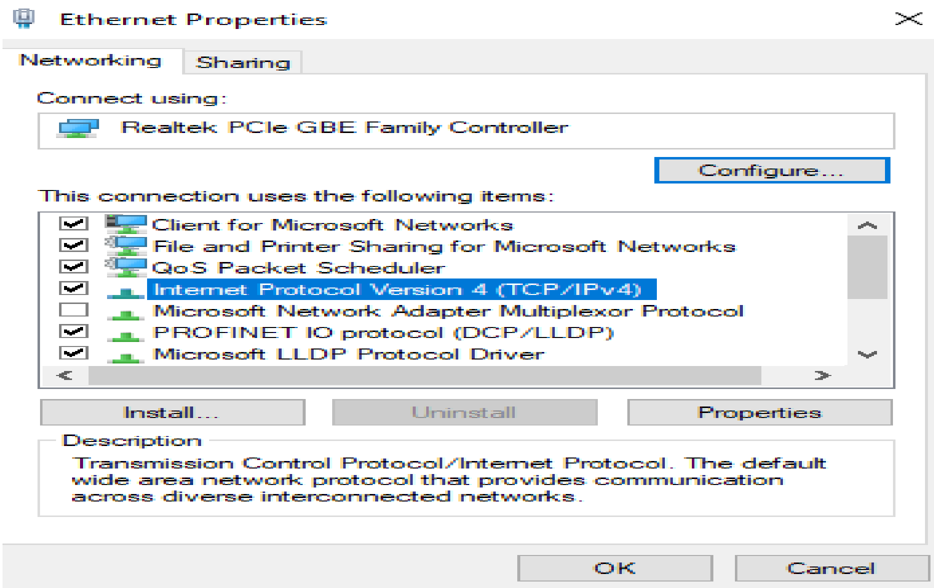
Before connecting any device with PC, we must apply some settings. PC's IP address must be in the same sequence as that of the other peripheral devices. For this here are some implementations:

- Go to network and internet settings in PC.



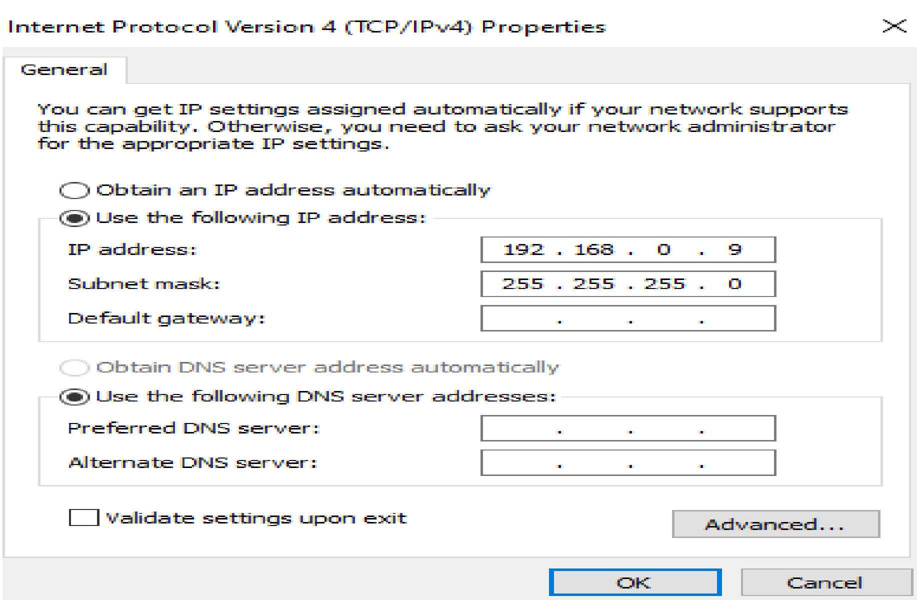
STEP1

- Click on ethernet option. If the ethernet cable properly connected, then (x) mark will not be there.
- Click on ethernet. This window will appear.



STEP2

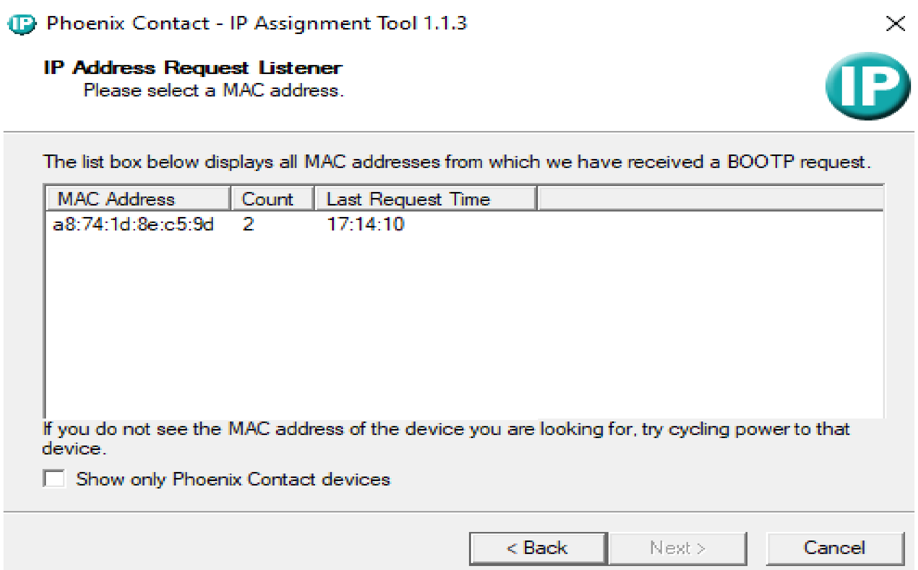
- Click on properties and make desired changes. The **IP** address must be in the series of **WIFI module** and **PLC** i.e. **192.168.0.9**



STEP3

Settings of WIFI module varies from model to model. Here is a procedure of setting IP address of PHOENIX CONTACT WIFI WLAN-1101.

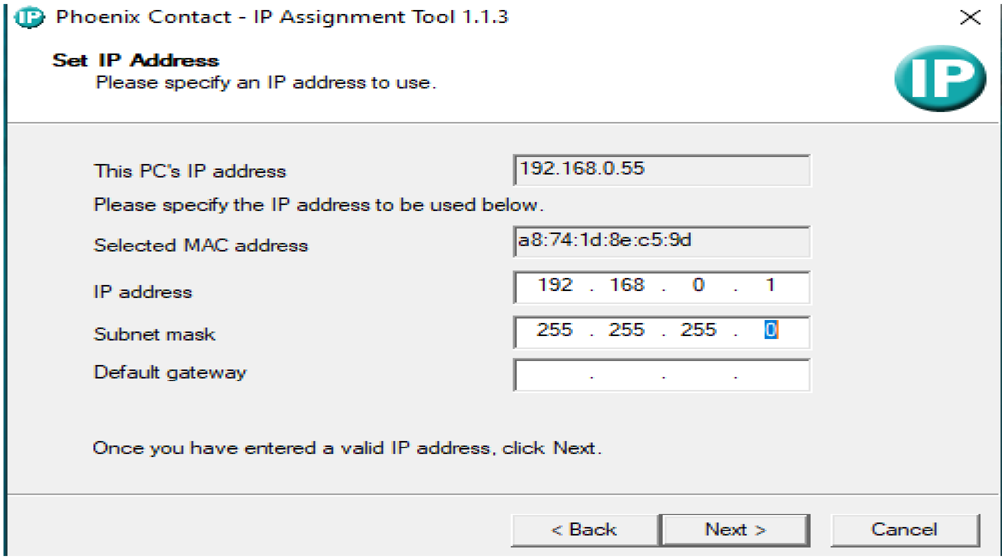
- Provide 24v DC. Connect the module to the computer via ethernet cable. The green led will glow only.
- Open the **IP ASSIGNMENT TOOL** in the computer.
- Click on next. The following window will appear. It will automatically detect the **mac address** of the connected Wi-Fi module.



STEP4

Note: If the mac address is not detected automatically then reset the Wi-Fi module by giving +24v dc supply to the reset slot in the Wi-Fi module and try again.

- Click on the mac address. Press next. The following window will appear.
- Give the **IP ADDRESS** and **SUBNET MASK**. Click on next. Leave the gateway as it is.
- If next option is not visible. Click anywhere in the subnet mask area or the gateway area.



STEP5

- IP ADDRESS process is finished.

After making changes we can connect PLC and PC with a secured connection. Now we can go to Sysmac Studio programming software and write a programming code for the discussed process.

CHAPTER 5

SYSTEM DESIGN

5.1 AUTOMATION IN BEVERAGE INDUSTRIES

The prime aim or objective of this project is to monitored or visualized an automatically running beverage process industry with the help of a Supervisory Control and Data Acquisition (SCADA) system and the controlling and handling the entire process will be done through an OMRON PLC. In this whole project we are going to use a 24 I/O (NX1P2-9024DT1) PLC from OMRON which will be programmed or simulated in OMRON own Sysmac Studio Software for programming with simple ladder logic coding or programming. Supervisory control SCADA software that we are going to use throughout this project is Wonderware INTOUCH SCADA software for visualizing or monitoring purpose. The SCADA system is complined with its own central processing unit along with the software package that will be used to handle, control and monitor various remote terminal units which are being placed all around the field. SCADA allows data handling and logging, graphical or visual representations, monitoring & correcting alarm system, changing and defining setpoints of the process etc.

5.2 SEQUENCE OF THE BEVERAGE INDUSTRY PROCESS

Almost in every beverage industry, filling of different liquids process is the most commonly practiced [38,40]. All industries concerned with beverages are designed in order to keep drinks of different material compositions into different container types. The whole container filling and processing is partitioned into following different stages:

1) **Sortation stage:** The containers that are moving over the conveyor belt will be sorted as metallic container & non-metallic container. The sorting is carried out with the help of (IPR) Inductive proximity IME08 sensor. This sensor will be able to sense the metallic things only and ignore all other material type like non-metallic container.

2) **Liquid filling stage:** Filling of containers with different materials compositions. Metallic containers will be having three liquids composition with 33% each but the non-metallic containers will be having only two liquid compositions with 33% each. Photo-electric retro reflective sensor will be used in this stage along with valves.

3) **Capping work:** as soon as filling of containers is completed, capping work unit will come into picture to apply caps and tightens them over filled one container. Here also Photo-electric retro reflective sensor will be used.

4) **Labelling work:** capping of the containers is followed by the labelling work unit. For differentiating containers, we will be using labels of different colors for particular container type. Here also Photo-electric retro reflective sensor will be used.

5) **Counting**: Counting process of metallic and non-metallic containers at the end in each and every case.

5.3 BLOCK REPRESENTATION OF PROCESS

- The communication between SCADA & PLC is via PC.
- The PLC needs regulated power supply of 24V DC to perform its functions. SMPS (switched mode power supply) provide this required voltage to PLC.

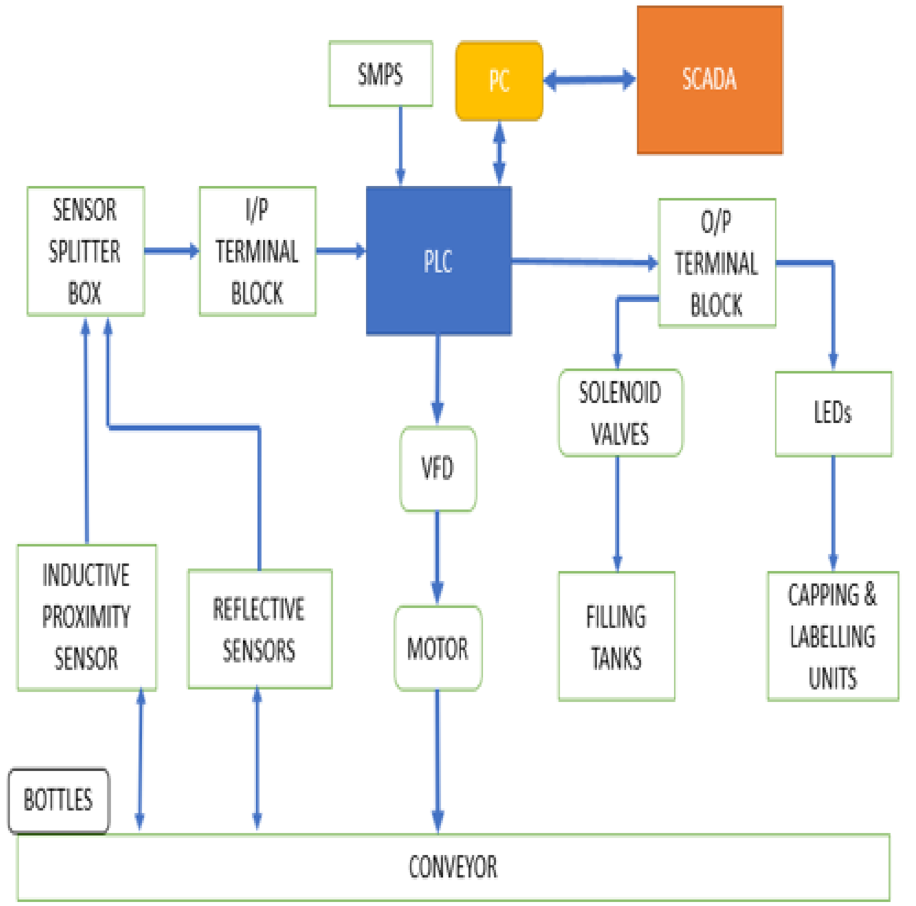


Fig 11: Block Diagram of Process

- Programmable logic controller is the center of the entire process. Input data or signals from input devices like sensors is feed into PLC, input sensor's information is processed, logical operations will be performed over inputs and a desired response is provided for the output connected devices at the output terminal block such as solenoid valves to filling tanks, LEDs to cap & label units, VFD to motor to conveyor belt.
- Conveyor system motion is governed by a DC motor, which get driven by CURTIS drive in order to provide precise and accuracy in motion.
- Both type of Containers will be sensed by inductive proximity sensor IPR and photoelectric retro-reflective sensors present over the conveyor. Both type of sensors are linked with a box called sensor splitter, which indicates the status of all sensors. Then infeed signal reaches the PLC input ports or terminals. Output

devices are linked with PLC via output ports or terminals. Different type of liquid is poured into both type of containers through valves connected to the filling tanks. Labelling & capping units are also controlled with reflective sensors signal conditioning to the PLC.

5.4 PROCESS MODELLING IN SCADA

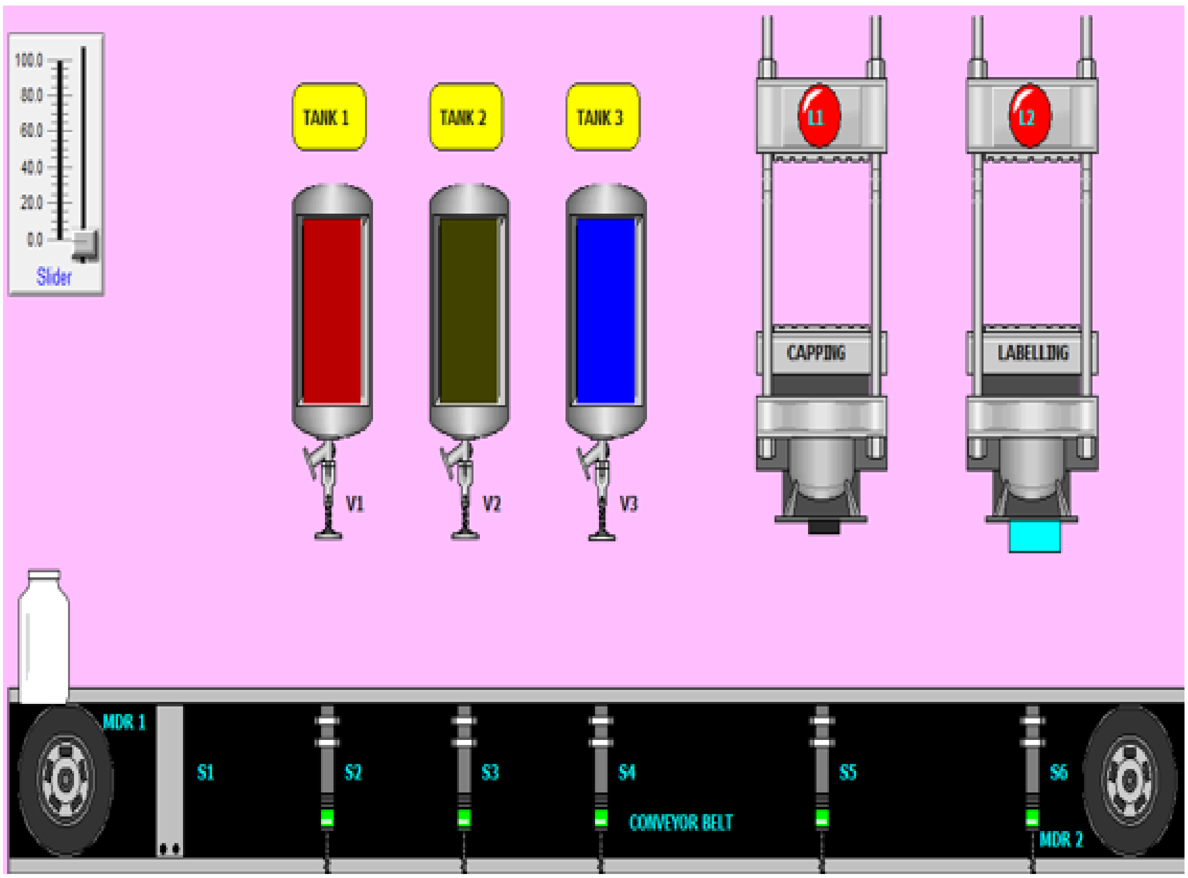


Fig 12: Model

INPUT COMPONENTS

IPR S1- One Inductive proximity sensor.

S2 to S6- Five Photo-electric retro-reflective sensors at various levels.

OUPUT COMPONENTS

Valve V1, valve V2, valve V3

Conveyor belt motor

LED L1 for capping process unit

LED L2 for labelling process unit.

FURTHER COMPONENTS

One unit of PLC.

Conveyor system

Two motor dependent or driven rollers

Three Tanks for different liquid types.

Unit for capping

Unit for labelling

5.5 COMPONENTS INFORMATION

5.5.1 PLC (Programmable Logic Controller (OMRON NX1P2-9024DT1)

SR.NO.	NOTATION	MEANING
1	**'NX1'**	Name of the series
2	**'P 2'**	Model with DC 24v power supply with in-built I/O
3	**'9 0'**	one-axis position control but two axes may be correlated
4	**'24'**	14 I/P + 10 O/P = 24
5	**"D"**	DC 24V inputs only
6	**T 1**	Outputs are PNP type

Fig 13: OMRON PLC

DESCRIPTION

- ✓ Motion and sequence controlling are integrated.
- ✓ Network control via ether-CAT is possible
- ✓ Based on Ether-CAT, 8 axes can be controlled
- ✓ Ether-CAT base subsystem for safety.
- ✓ operation can be without battery too.
- ✓ 24 in-built I/O
- ✓ eight (8) NX input output Units are connectable
- ✓ 16 remote type NX input outputs are connectable through coupler.
- ✓ IEC 61131-3 programming standard.

5.5.2 PROGRAMMING OF PLC IN SOFTWARE

For this purpose, we are using SYSMAC STUDIO from OMRON.

PLC programming software Sysmac Studio is the software that was developed for providing the integrated environment for programming, setting the drive parameters, troubleshooting and debugging of automated machine controllers which includes the NX&NJ units of CPU, slaves of Ether-CAT Slave with integration of HMIs.

FEATURES

• Compliant totally with JIS B3503 Japanese standard & IEC 61131-3 open standard.

• It Supports Structured programming, Ladder logic programming, functional block & Text programming with lots of instruction sets.

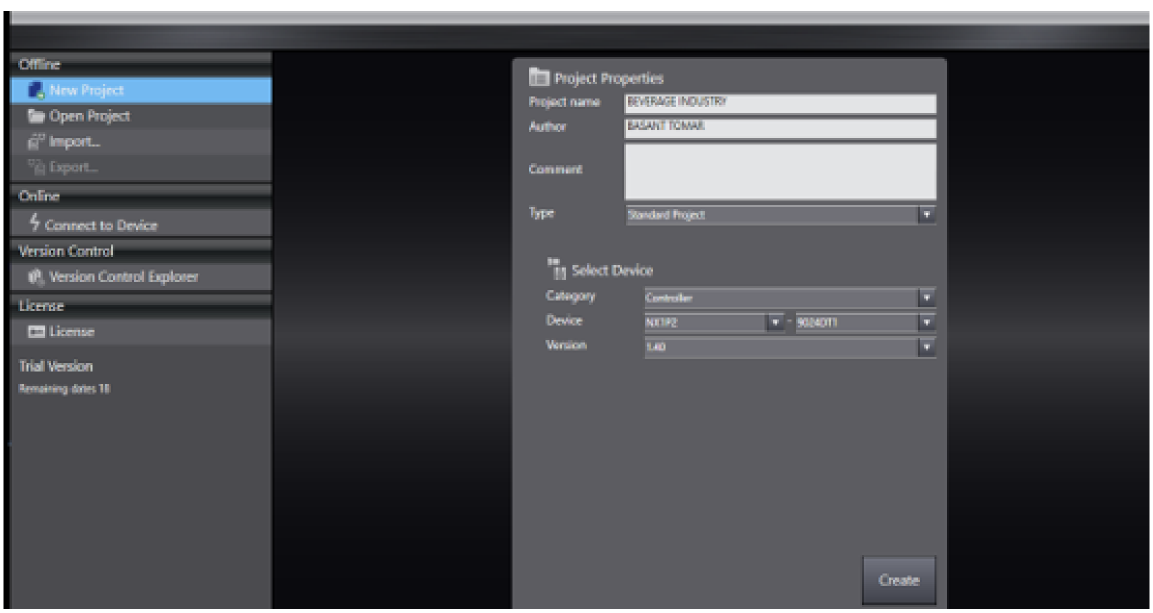
• CAM editor is used to ease the programming with entangled & complex profiles in motion.

• Single tool for simulation of motion & sequence in both 3D & 2D environment.

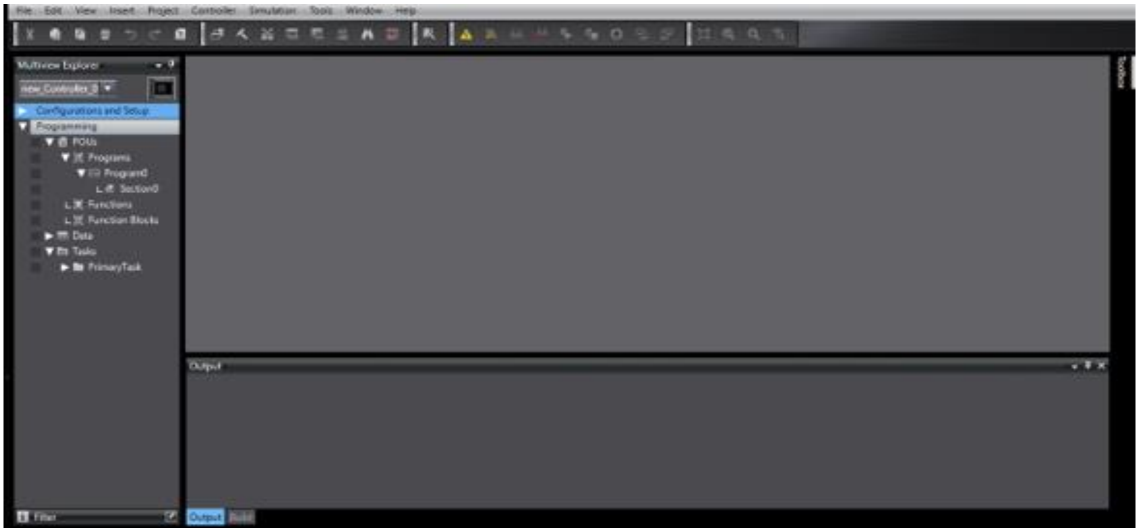
• Security function of advanced level, supporting security optimized password of 32 digits.

SYSMAC studio interface

STEP 1



STEP 2



STEP 3

5.5.3 WONDERWARE INTOUCH SCADA SOFTWARE

An application window is a container for one or more graphics that model your production processes. InTouch windows are visual panels that contain the user interface of your InTouch application.

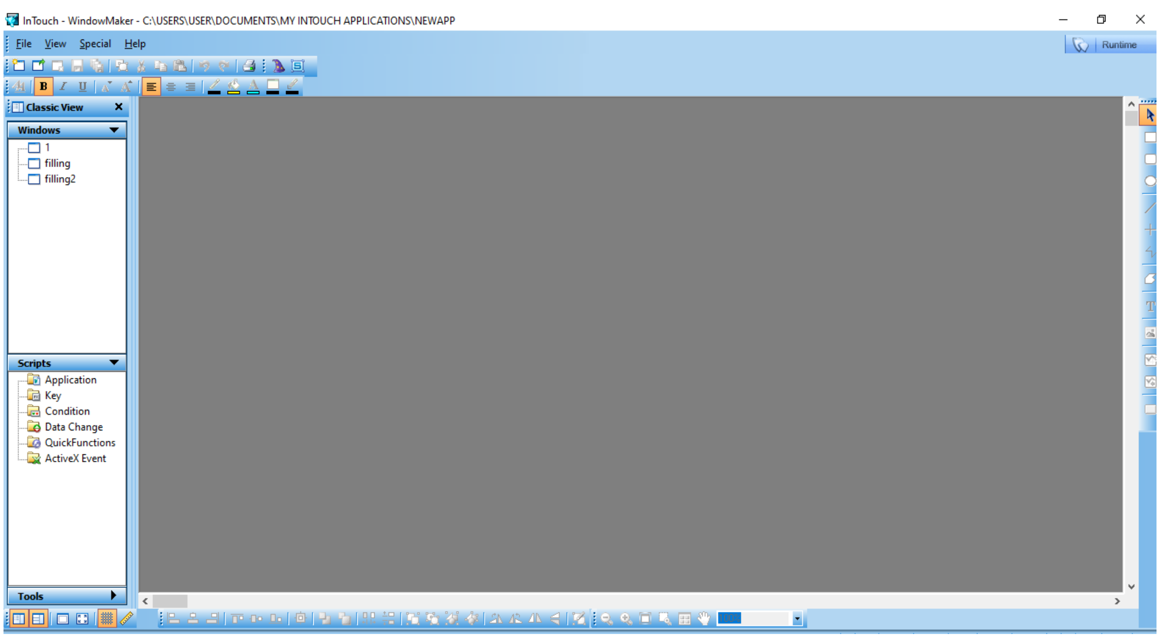


Fig. 14: Interfacing Screen

- **Key Points**

  There are three types of InTouch windows:

  - A replacement window automatically closes any windows it intersects with when it appears on the screen.
  - A popup window stays on top of all other windows. Popup windows usually require a response from the user to be closed.
  - An overlay window appears on top of currently open windows. Overlay windows are useful for creating applications consisting of multiple panels with conditional or optional variations.
  - You can define the following characteristics of a window when you create it:
  - Horizontal and vertical size in pixels
  - Type of frame
  - Background color
  - Presence of standard controls for windows

- **Visualization**

  Visualization topics describe how to create a graphical operator interface for an InTouch application. These topics cover:

  - Navigating in the Window Maker development environment
  - Creating and managing application windows
  - Creating and editing simple or complex graphic objects
  - Adding run-time functionality to graphic objects using animation links
  - Embedding wizards and ActiveX controls

- **Applications at Run Time**

Window Viewer is the run-time environment for your InTouch applications. Based upon your application's requirements, you can configure properties that determine the visual appearance and operational characteristics of Window Viewer.

You can run stand-alone and managed InTouch applications with Window Viewer. The following figure shows that a managed application is deployed from the ArchestrA IDE to the computer running the InTouch HMI. The first time you deploy an InTouch View App object, the associated InTouch application files are copied to the node of the platform that hosts the object

5.5.4 INDUCTIVE PROXIMITY IPR SENSOR IME08

Inductive proximity IPR sensor is a sensing device which works on the fundamentals of electromagnetic induction for detecting or measuring objects. A present inductor(L) will develop a magnetism phenomenon called field as soon as current flows into that. This similar phenomenon is used in detection of metallic materials that mesh with some magnetic field. The substance which are non-metallic like liquids, plastic, wood & dirt, do not mesh along the magnetic field. An IPR sensor is also operated in both wet and dirty surroundings.

Fig. 15: SICK IME08

CHARACTERICTICS	VALUE
Power Supply	DC 24V
Switching frequency	4 KHz
Output Switching mode	PNP type
Max Current	<10 mA
Optimum range of Sensing	<=3 mm
Sensing head diameter	8 mm
Wiring system	3-wire based system, DC
rating	IP67
Range of Hysteresis	(4.5-15.5) %
SC protection	Available
Reverse polarity protection	Available
Operational temperature	-20 to +80

5.5.5 PHOTOELECTRIC RETRO-REFLECTIVE SENSOR - GL6-P4212

It is a reflective type sensor. Continuously the signals from the head unit are emitted and they get reflected back from the provided reflected surface. As soon as any object comes in the path of light between head and tape, it will LOW its output and detection are done.

Fig 16: GL6 sensor and GL6 Reflector

FEATURE	DATA
Supply voltage	24V DC
Ripples	10%
Current consumption	30 mA
Switching output	NPN
Switching mode	Light/dark
Switching frequency	1000 Hz
Response time	625 micro sec
Connection type	Male connector m8- 4 pin
Sensor/ detection principle	Photoelectric retro-reflective sensor, Dual lens
Dimensions (W x H x D)	12 mm x 31.5 mm x 21 mm
Sensing range max.	≤ 6 m
Sensing range	≤ 5 m
Type of light	Visible red light
Light source	Pin Point LED
Light spot size (distance)	Ø 8 mm (350 mm)
Wave length	650 nm

5.5.6 DC DRIVE (CURTIS 1212S) & ROTOMAG 1KW DC MOTOR

DC motor drive, Curtis-1212S speed controller of motors is operated at a power supply of 24V DC. It provides accurate, precise and refine control for PMMC (permanent magnet) travelling motors which are battery driven shuttles.

The Linearity in cutback of current will ensure finished controlling, with no accidental power loss during the case of overtemperature or undervoltage. This controller drive is being used in operating Rotomag HK-series 1KW DC motor. 1212S-CURTIS controllers are totally reprogrammable with the help of a software, Curtis 1214 OEM programming software.

Fig 17: Rotomag motor and Curtis Drive

5.5.7 SOLENOID VALVES

Here we are using 3 solenoid valves. A solenoid valve is basically an electrically operated valve. The valve constitutes a hollow solenoid (an electric hollow coil) with a moving ferro or ferrous magnetic core (called as plunger) at the center. When at resting position, the plunger shut down the small opening or orifice. But when current is passed through this the magnetic coil get energized and as a result the plunger moves up and orifice get open.

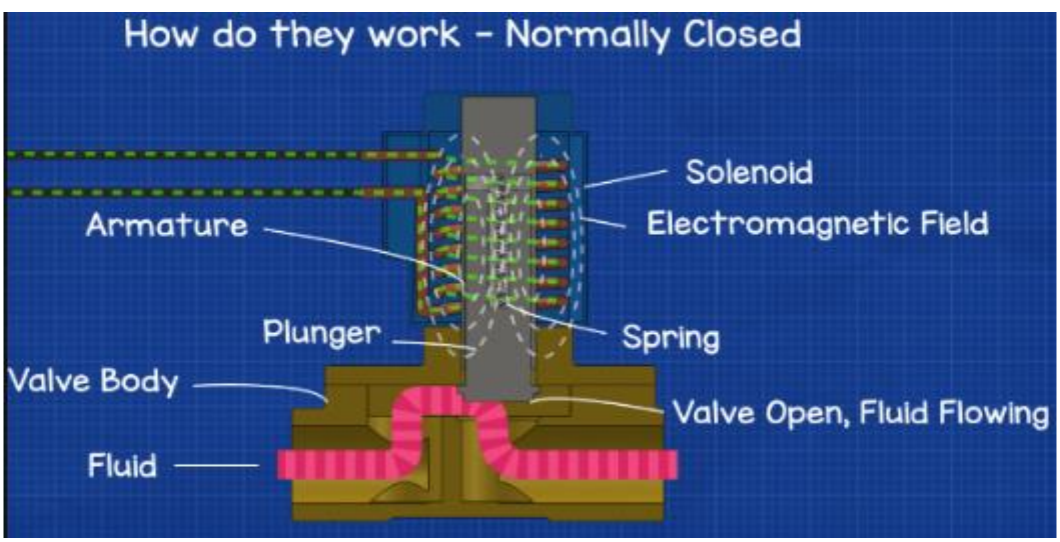


Fig 18: Solenoid Valve

CHAPTER 6

FLOW DIAGRAMS

6.1 PROCESS DESCRIPTION WITH FLOW DIAGRAMS

The whole system process is described with the help of flow charts discussed below

- In this flow chart functioning of the conveyor belt motor is depicted. Accuracy in the motion of the conveyor belt is one of the most keen factor of this process. The conveyor belt motion is governed or we can say totally influenced by input signals sent by the sensors (IR sensors and reflective sensors) and output signal send by the VFD (Variable Frequency Drive, CURTIS 1214S). Inductive Proximity Sensor (IPR S1) does not directly influence the stopping and motion of the conveyor belt but whenever any of the five Photoelectric Retro Reflective type sensors (S2, S3, S4, S5, S6) sense any of the container metallic or non-metallic then the conveyor belt motor stops its motion and resumed after a fixed time interval when any of the processes (container filling, capping of container and labelling of the container) is finally done.

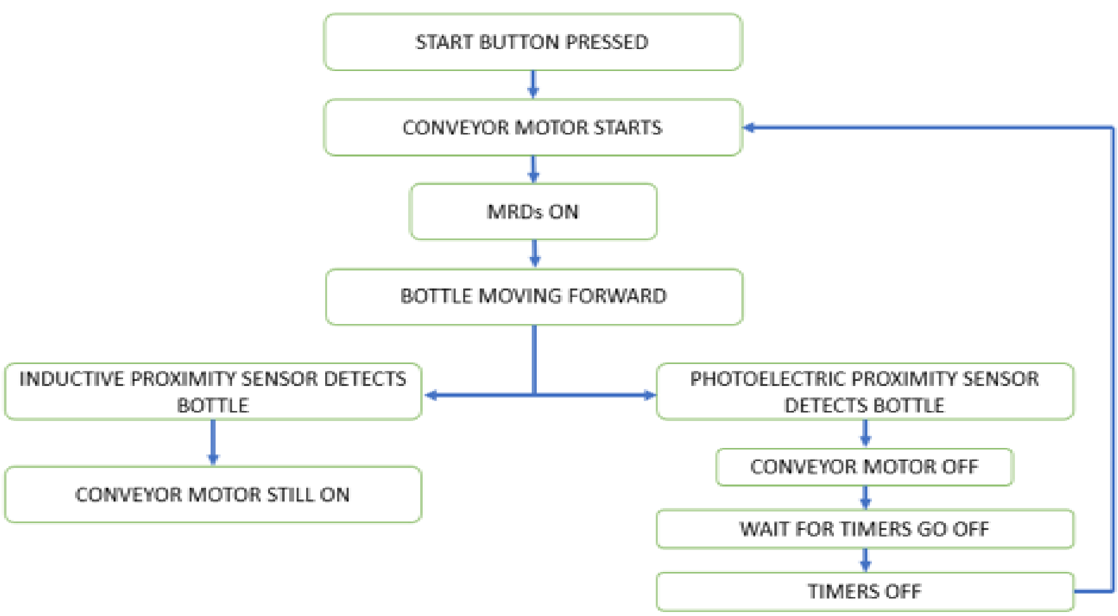


Fig 19: Flow diagram 1

- In this second flow chart, the whole process of sorting of containers and filling of each type of containers is depicted. When any type of container is moving on the conveyor belt, then Inductive proximity sensor (IPR S1) will decide the type of the container, whether it is metallic or non-metallic. If the presence of metallic container is there, then the Inductive PR Sensor S1 turns green, else there will be no change in PR sensor in case of non-metallic container. After that, both type of containers moves further and in later stage, they are detected by the Photoelectric retro- reflective type sensor (S2), which results into the opening of solenoid valve (V1) and as a result of this the tank1 starts filling the liquid into the container and this task lasts till the 33% or approximately one-third of the container gets filled. Then timer of the tank1 goes OFF and the motion of the conveyor belt starts again and continues till the next upcoming Photoelectric retro-reflective type sensor (S3) detects the container, which results into the opening of solenoid valve (V2) and as a result of this the tank2 starts filling the liquid into the container and this task lasts till the 66% or approximately two-third of the container gets filled. Then timer of the tank2 goes OFF and motion of the conveyor belt starts again and continues till the next upcoming Photoelectric retro-reflective type sensor (S4) detects the container, which results into the opening of solenoid valve (V3) and as a result of this the tank3 starts filling the liquid into the container and this task lasts till the 99% or approximately full container gets filled. But if the container that was not detected by the IPR S1 then as a action of that the solenoid valve (V3) will not open and as a result the tank 3 will not fill the non-metallic container in this scenario. As a consequence of this, the non-metallic container will be filled by 66% or two-third of the liquid in the entire process.

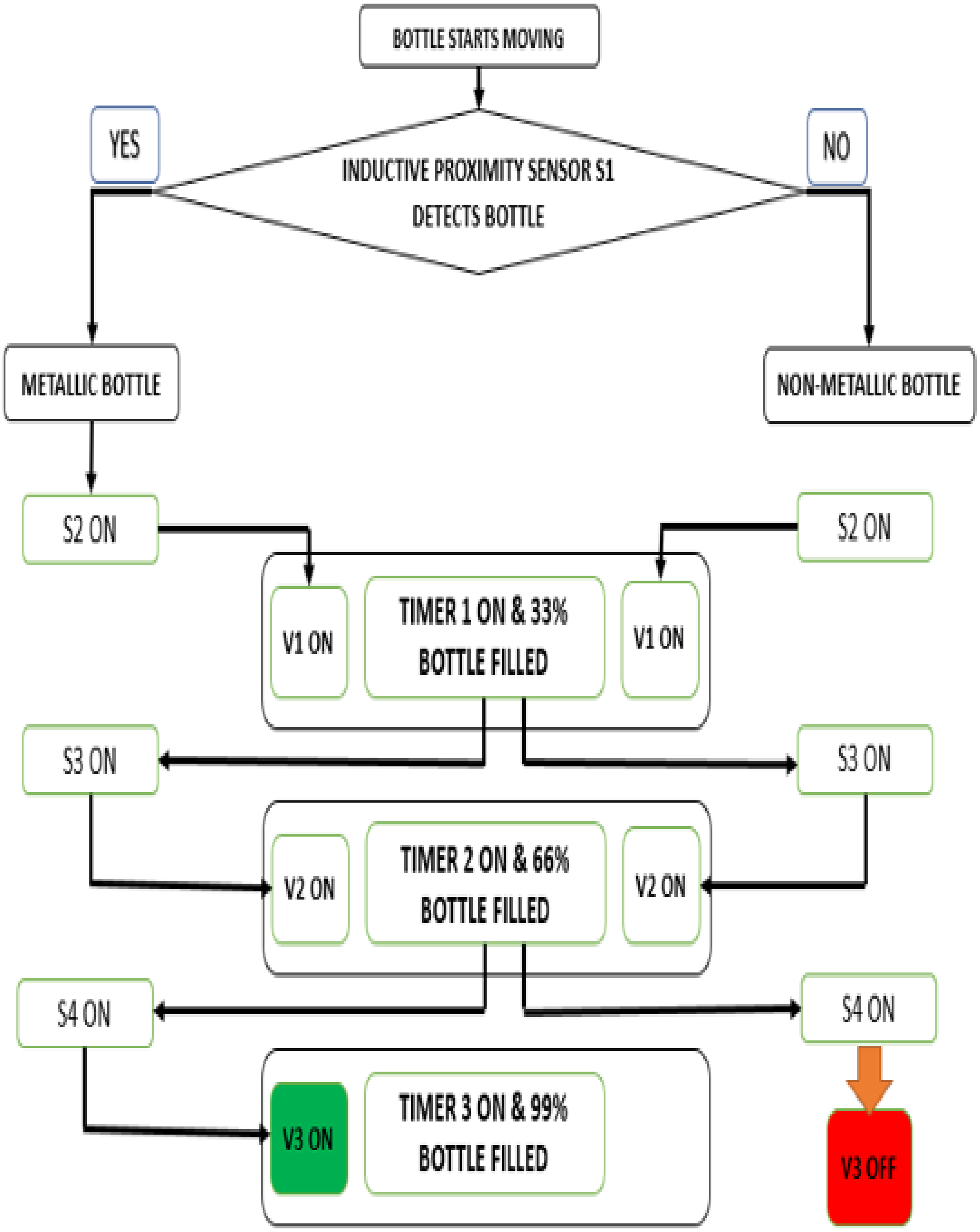


Fig 20: Flow diagram 2

- This flow chart shows the process of capping of the containers along with the process of labelling of the containers. When Photoelectric retro-reflective sensor (S5) detects any type of the containers then the LED L1 of the unit of capping turns green and capping initiated to put the cap over the containers. During this entire process the conveyor belt motion is stopped. And after the competition of the process the motor resumes again. Now when Photoelectric retro-reflective sensor (S6) detects any type of the containers then the LED L2 of the unit of labelling turns green and labelling initiated to put the labels over the containers. During this entire process the conveyor belt motion is stopped. And after the competition of the process the motor resumes again.

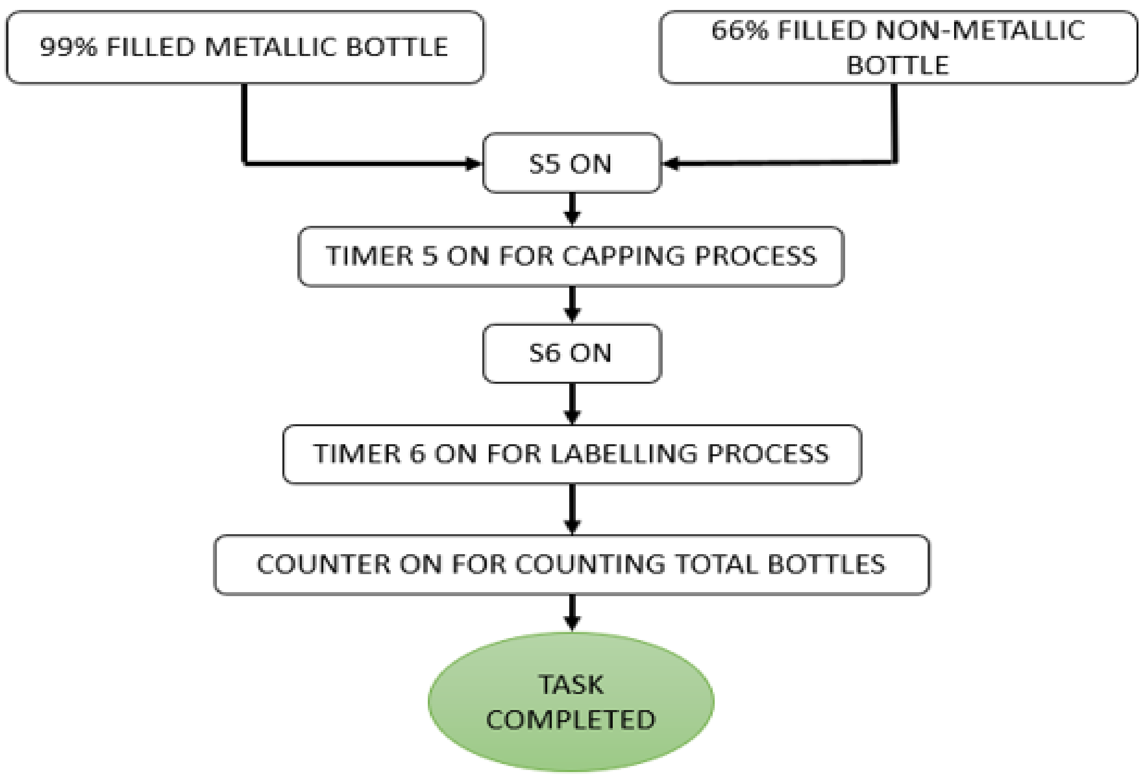


Fig 21: Flow diagram 3

CHAPTER 7

INFERENCE & RESULTS

7.1 EXPLANATION

Here, in this result section we are going to observe every result from the two used software of SCADA & PLC i.e. INTOUCH SCADA software by Wonderware & PLC programming simulation observations done in OMRON software Sysmac Studio. The whole process of visualization or monitoring can be observed in the INTOUCH SCADA software by Wonderware screen one by one, in the form of steps and PLC programming simulation of various steps is shown below of each SCADA software screen.

Start the process by pressing START button. Due to this action, the Conveyor motor start running and as a result MDRs move the conveyor belt.

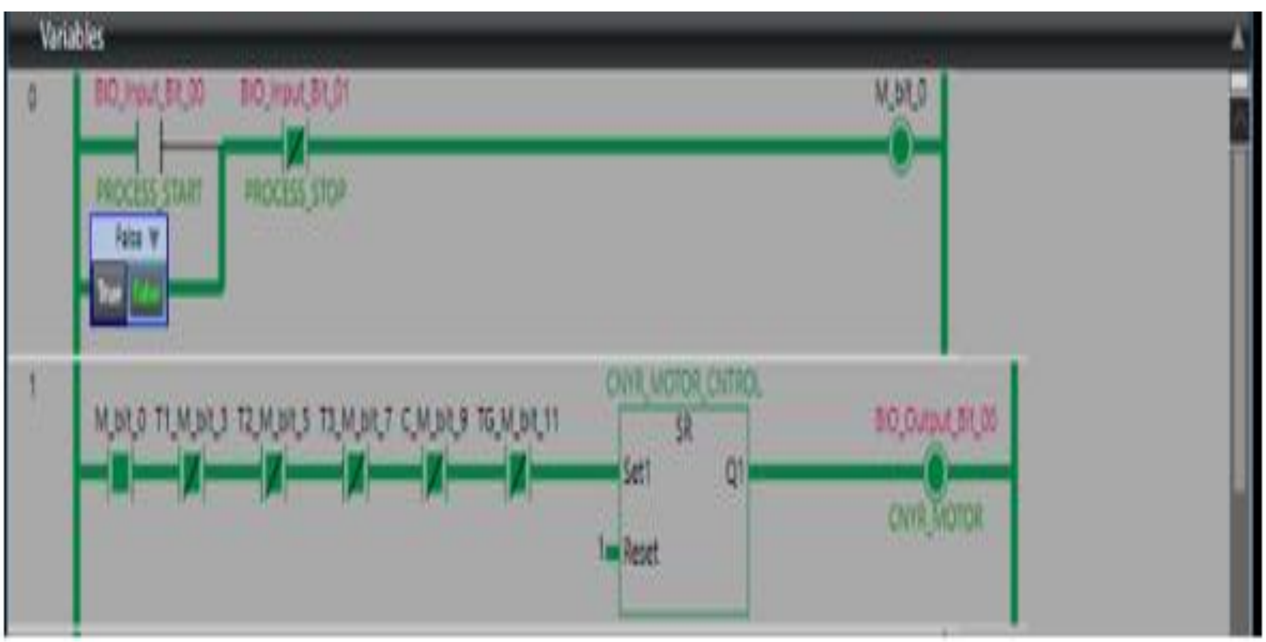


Fig 22: Initiation of the Process

7.2 METALLIC CONTAINER MOVING CASE

1. Inductive Proximity (IPR) sensor (S1) is able to spot metallic container which is moving on the conveyor.

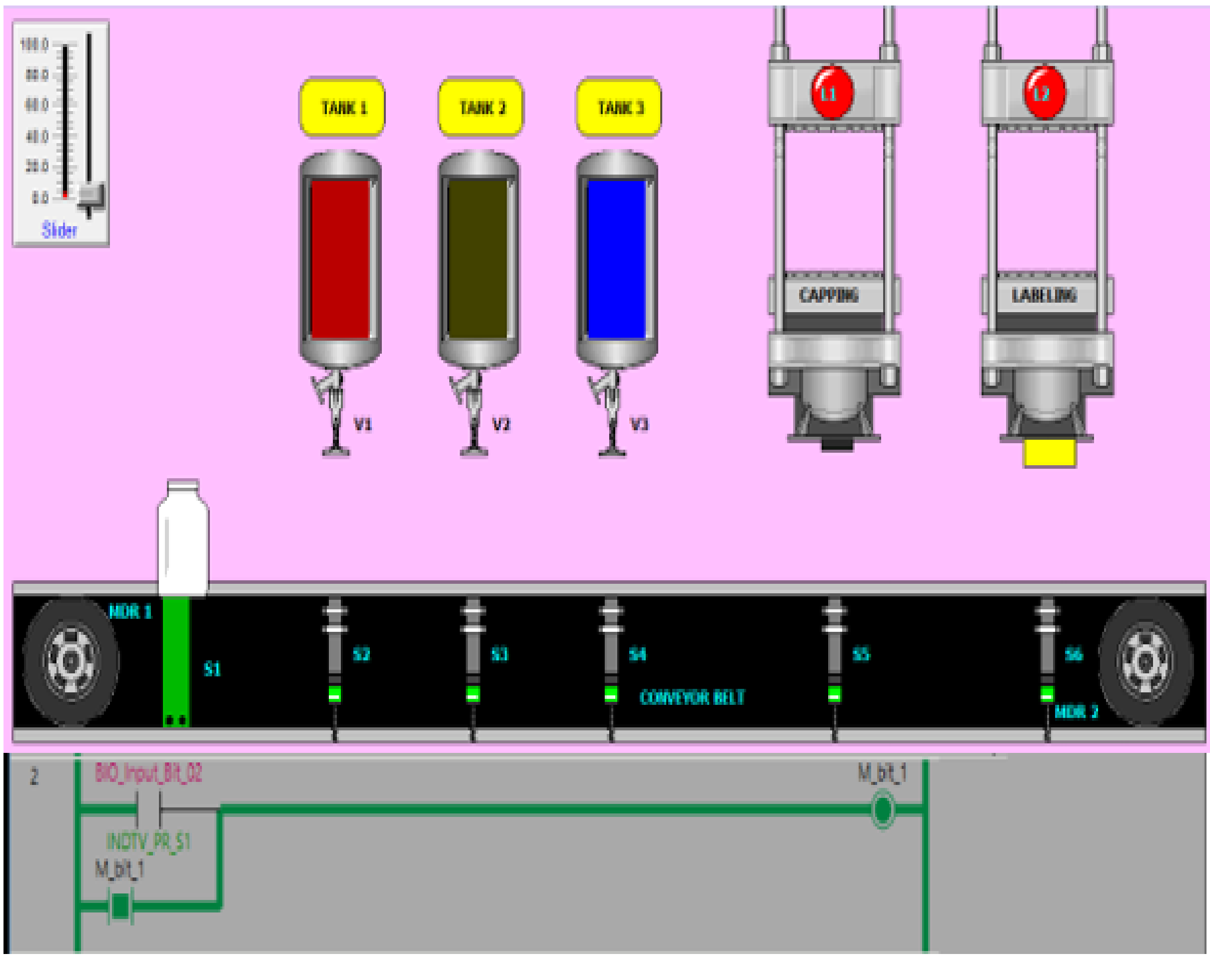


Fig 23: Sorting with Code

2. Now the (S2) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops and the V1 valve opens and container got filled one-third. The moving of conveyor belt continues after some predefined time i.e. time required to fill the container one-third.

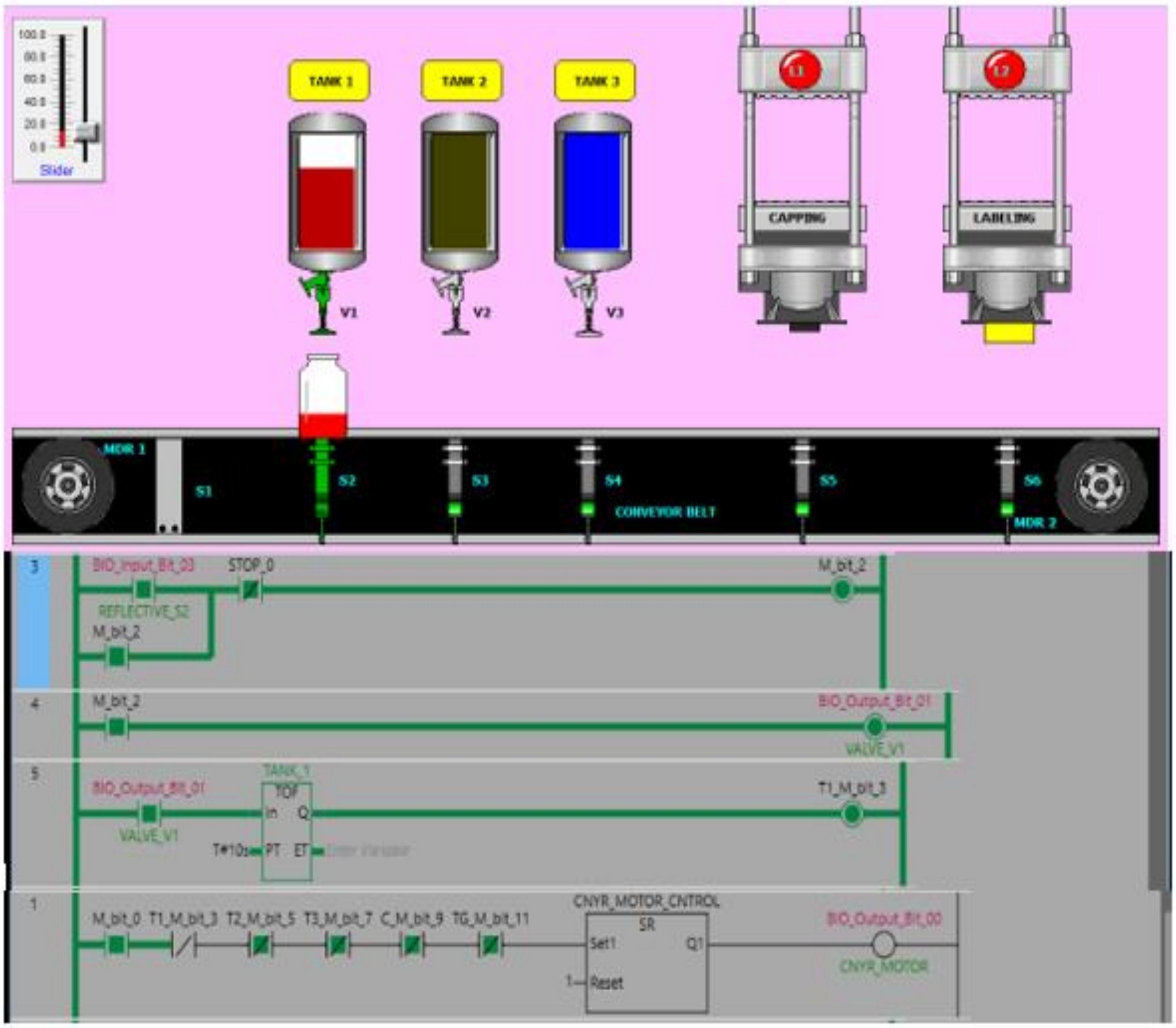


Fig 24: Tank1 Level with Code

3. Now the (S3) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops and the V2 valve opens and container got filled two-third. The moving of conveyor belt continues after some predefined time i.e. time required to fill the container one-third again.

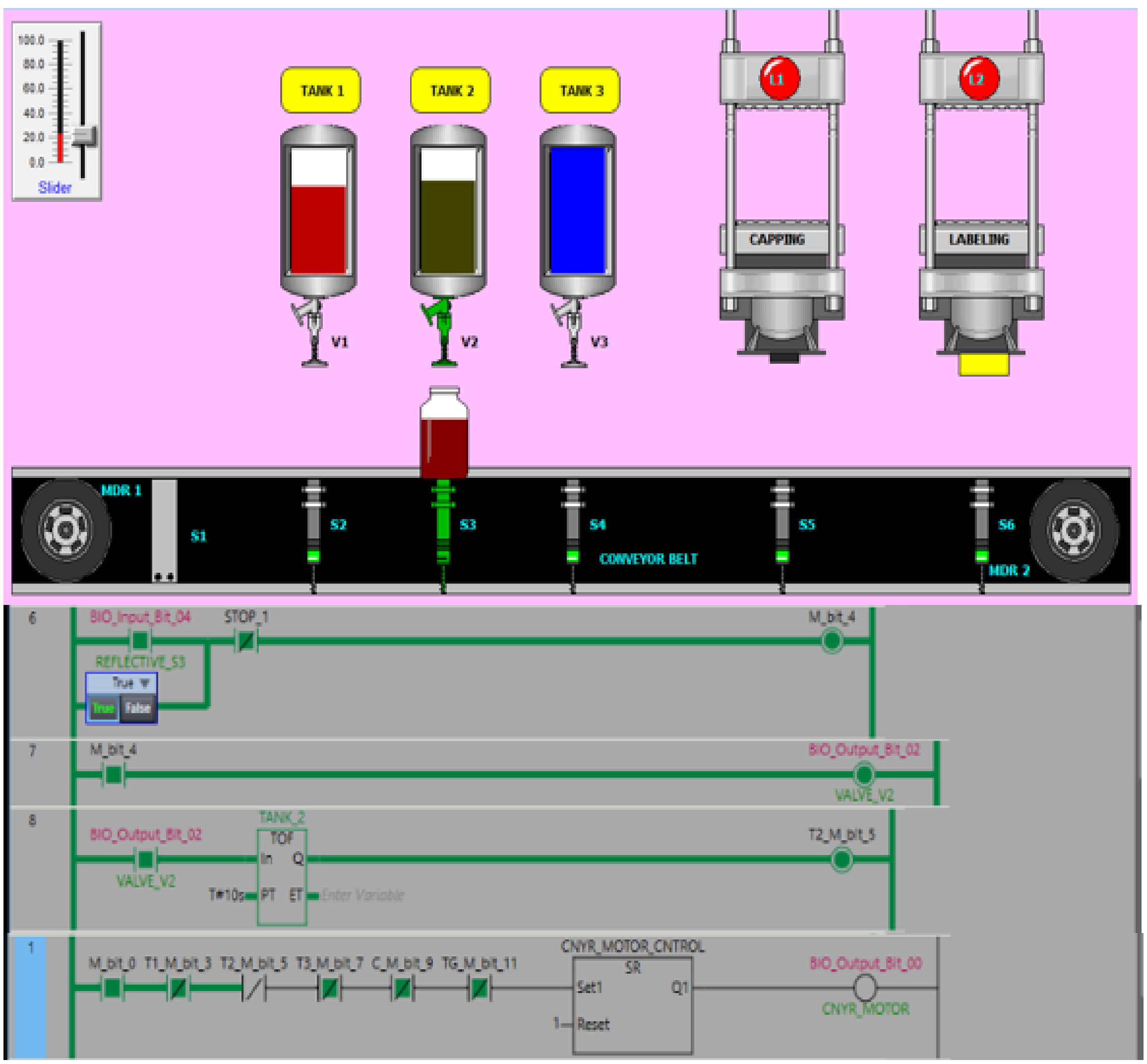


Fig 25: Tank 2 Level with Code

4. Now the (S4) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops and the V3 valve opens and container got filled almost full. The moving of conveyor belt continues after some predefined time i.e. time required to fill the container one-third again.

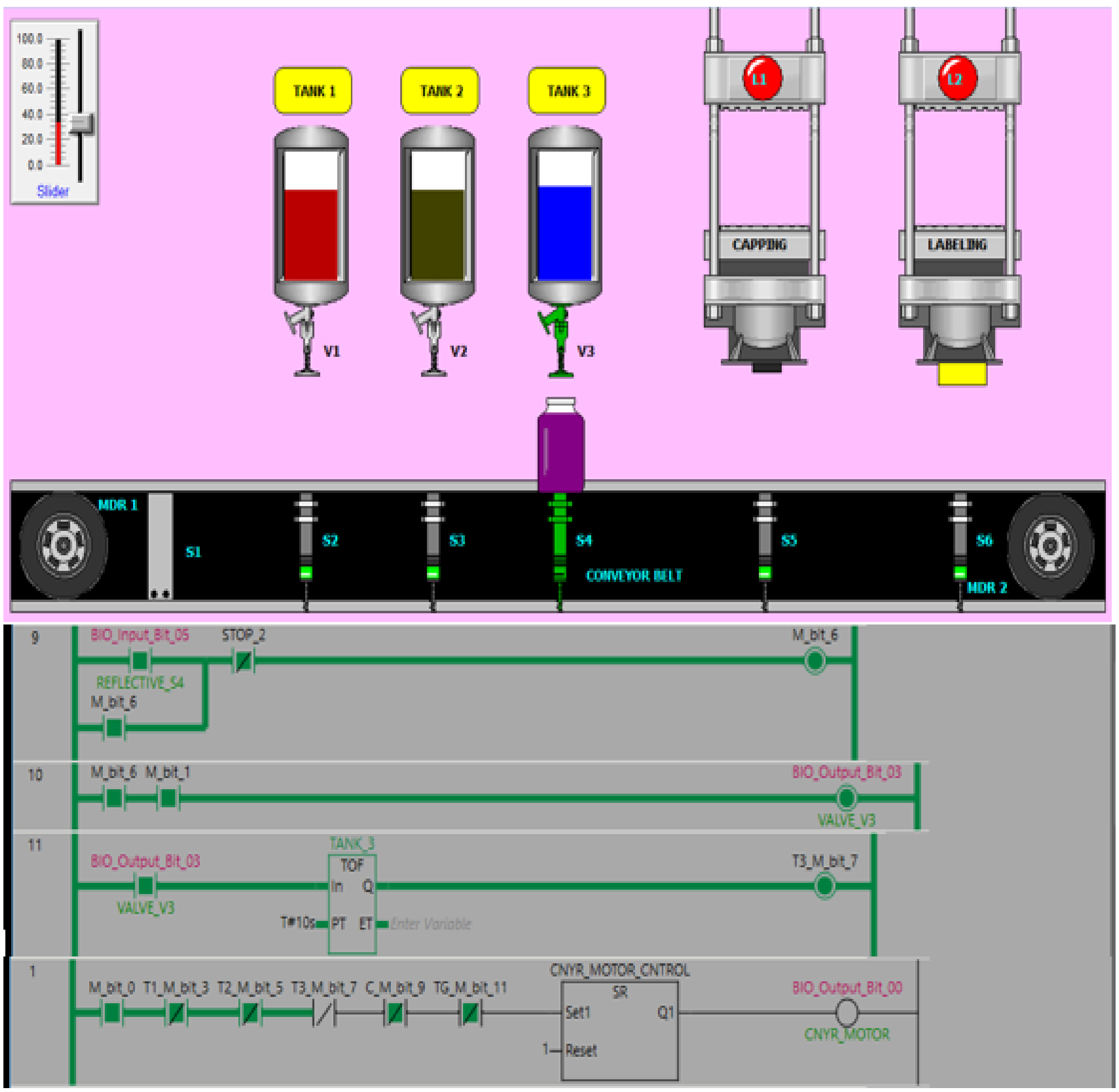


Fig 26: Tank 3 Level with Code

5. Now the (S5) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops, L1 LED turns into green from red. The container got capped by capping unit. The moving of conveyor belt continues after some predefined time i.e. time required to cap the container.

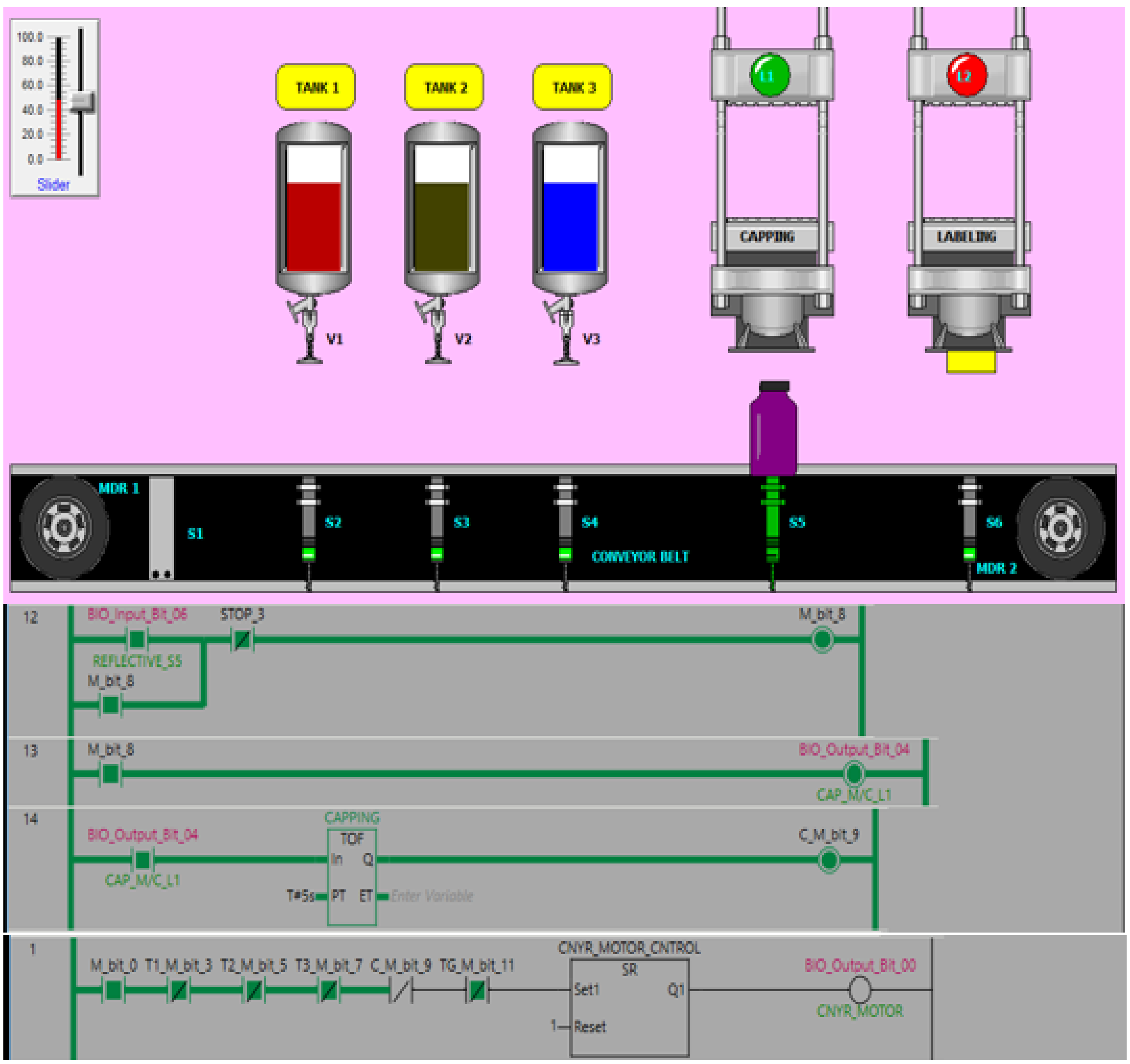


Fig 27: Capping with Code

6. Now the (S6) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops, L2 LED turns into green from red. The container got yellow label by labelling unit. The moving of conveyor belt continues after some predefined time i.e. time required to label the container.

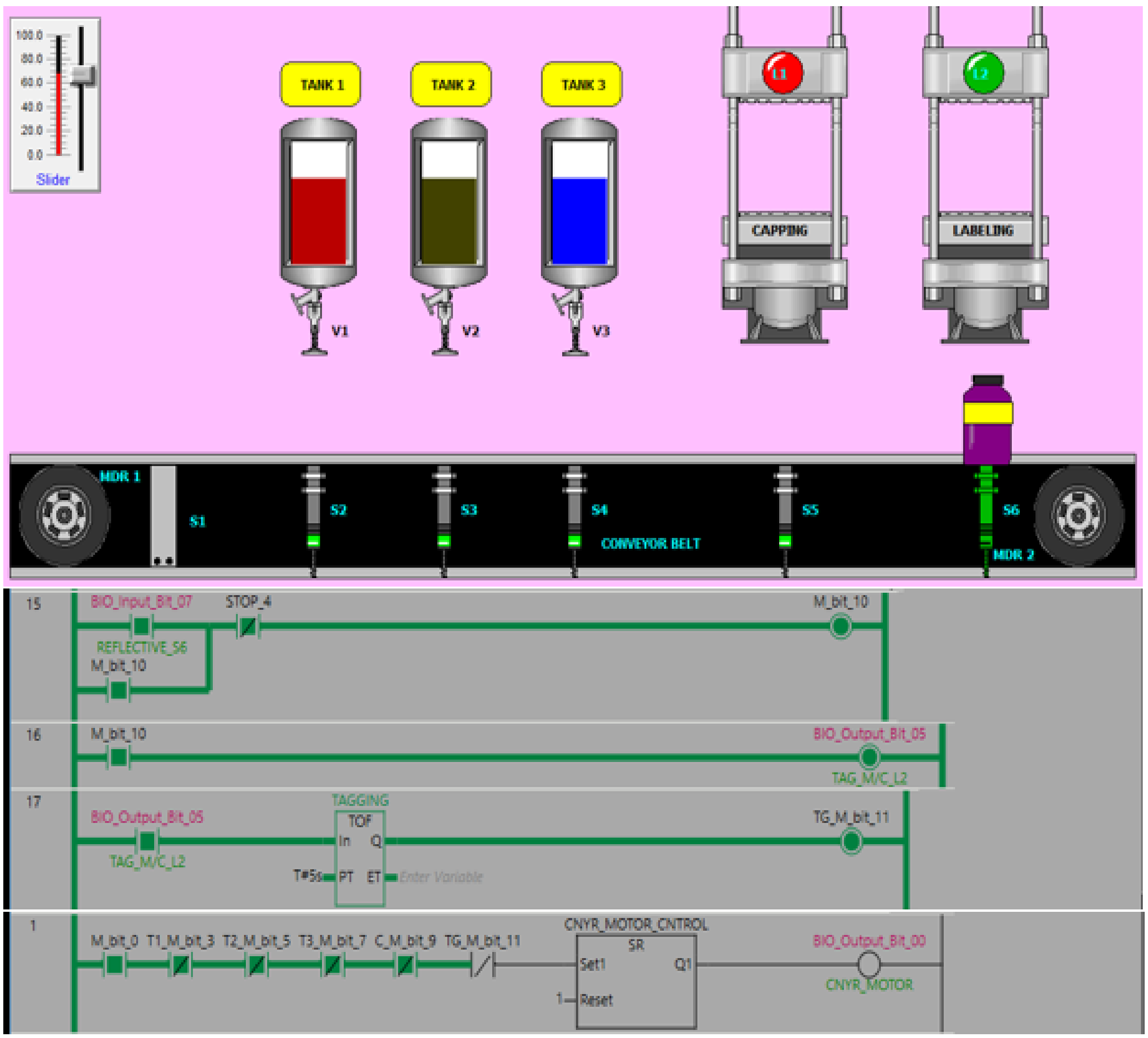


Fig 28: Labelling with Code

7.3 NON- METALLIC CONTAINER MOVING CASE

1) Inductive Proximity (IPR) sensor (S1) is **not** able to spot non-metallic container which is moving on the conveyor.

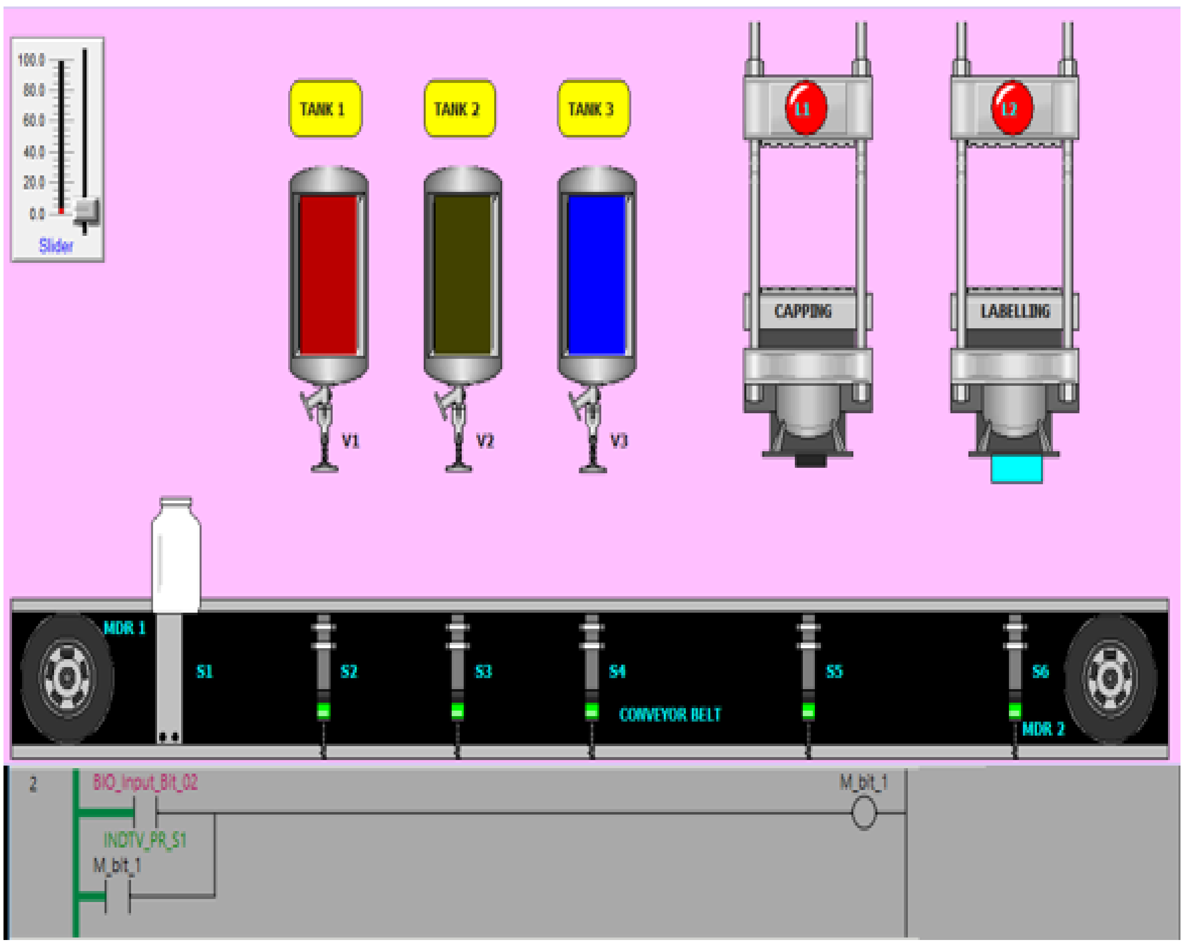


Fig 29: Non-detection with Code

2) Now the (S2) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops and the V1 valve opens and container got filled one-third. The moving of conveyor belt continues after some predefined time i.e. time required to fill the container one-third.

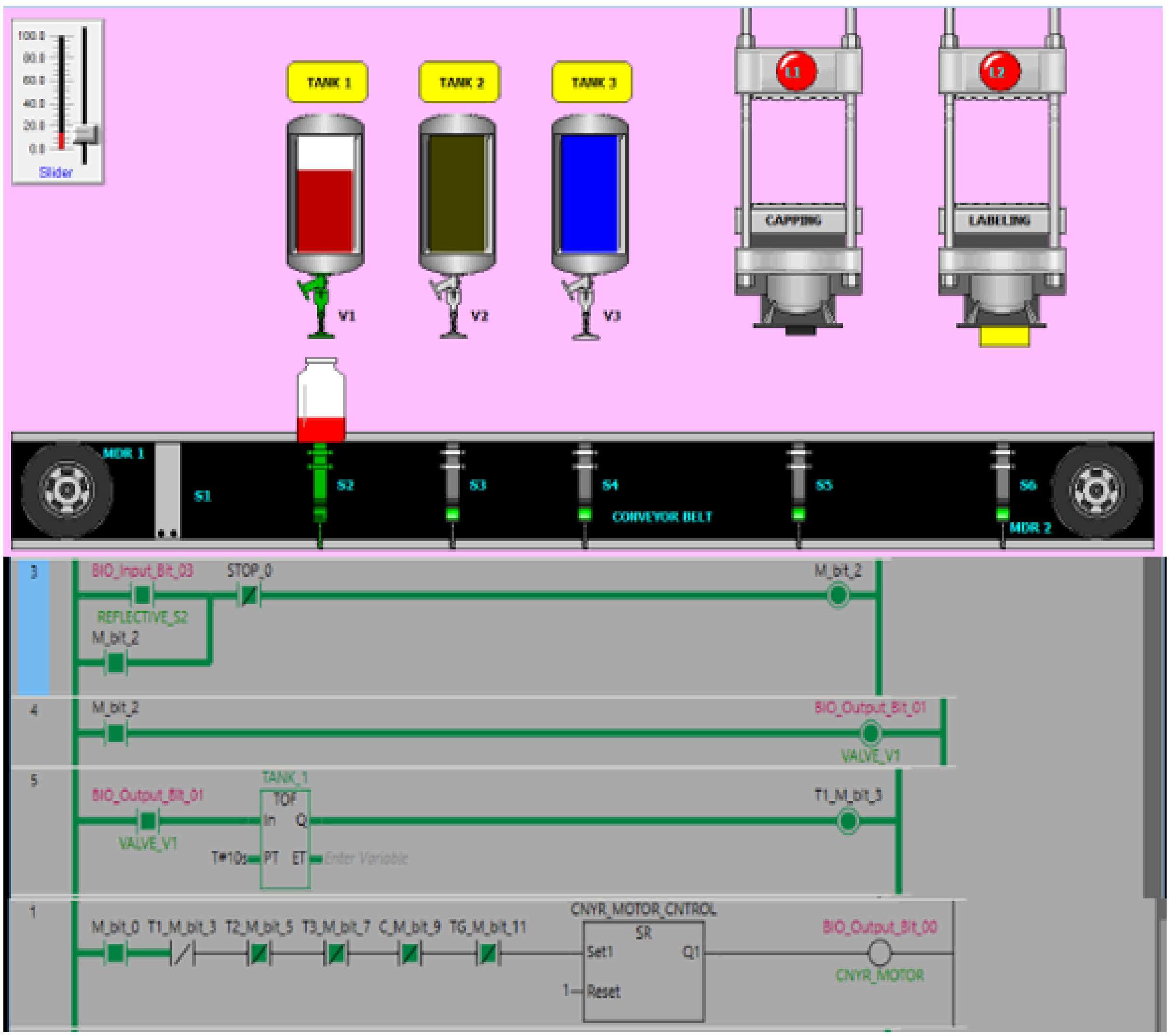


Fig 30: Tank 1 level with code Case B

3) Now the (S3) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops and the V2 valve opens and container got filled two-third. The moving of conveyor belt continues after some predefined time i.e. time required to fill the container one-third again.

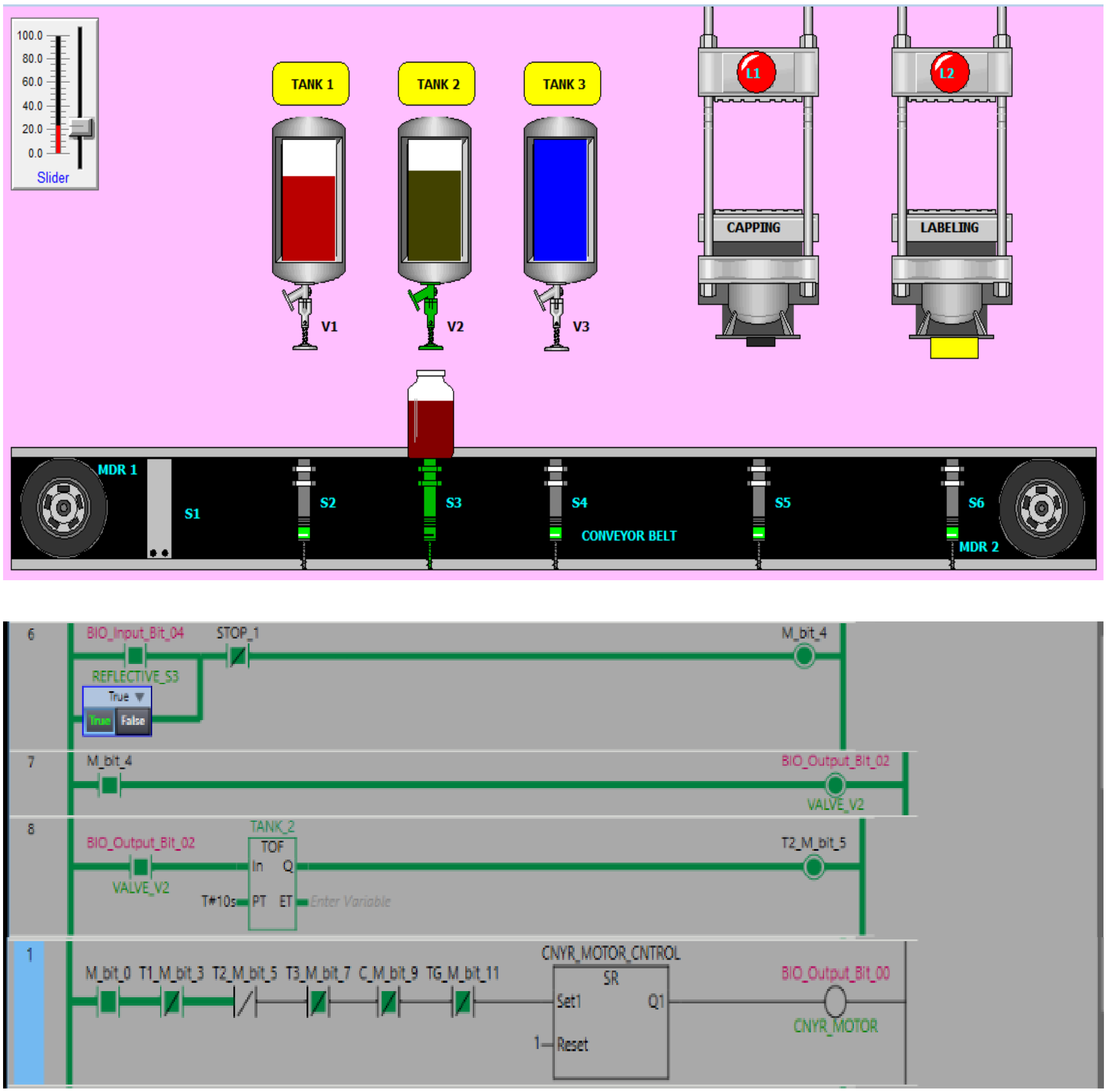


Fig 31: Tank 2 level with code Case B

4) Now the (S4) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops but the V3 valve will not open and container is still filled two-third. The moving of conveyor belt continues with no stopping.

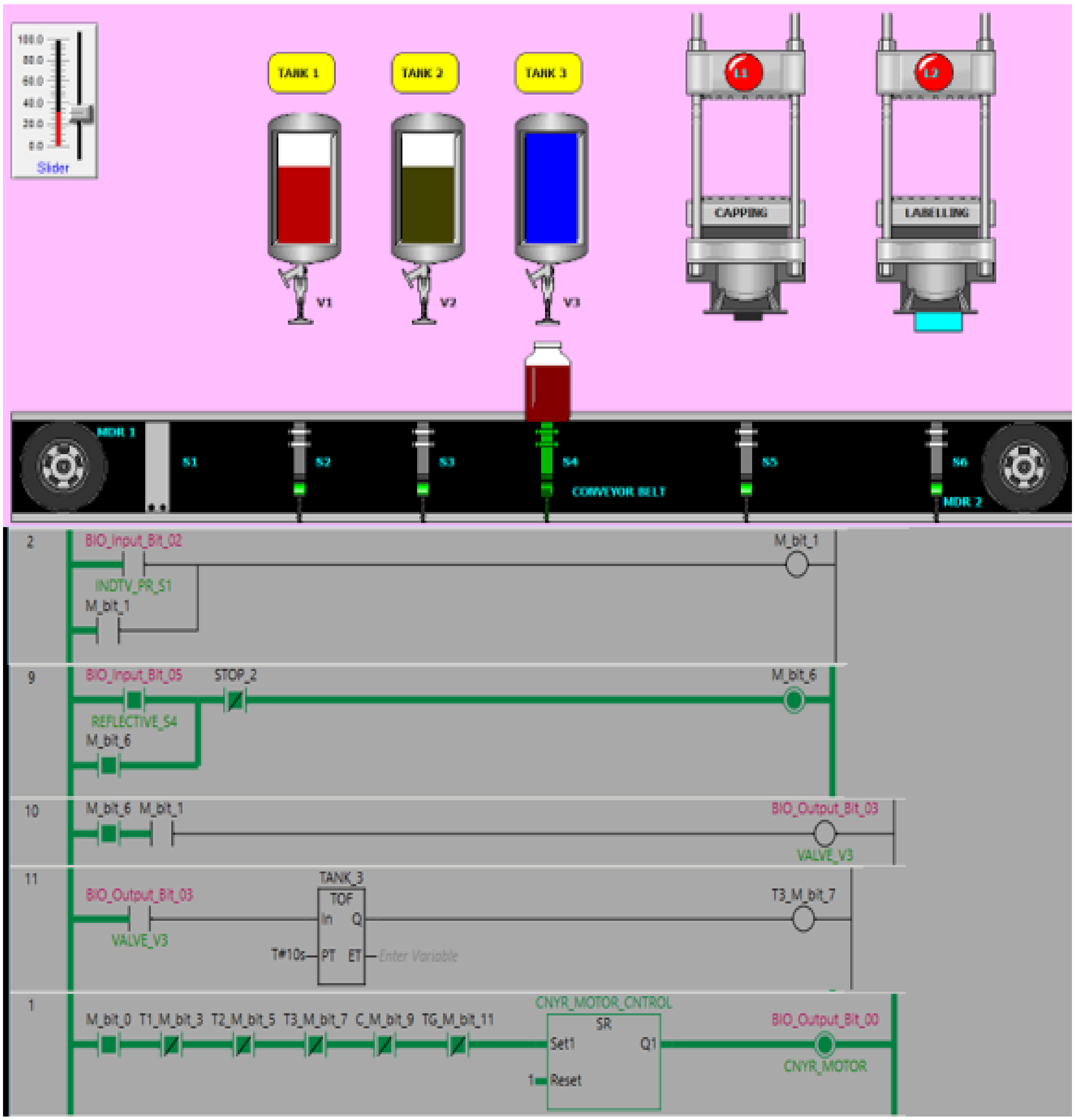


Fig 32: Tank 3 level with code Case B

5) Now the (S5) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops, L1 LED turns into green from red. The container got capped by capping unit. The moving of conveyor belt continues after some predefined time i.e. time required to cap the container.

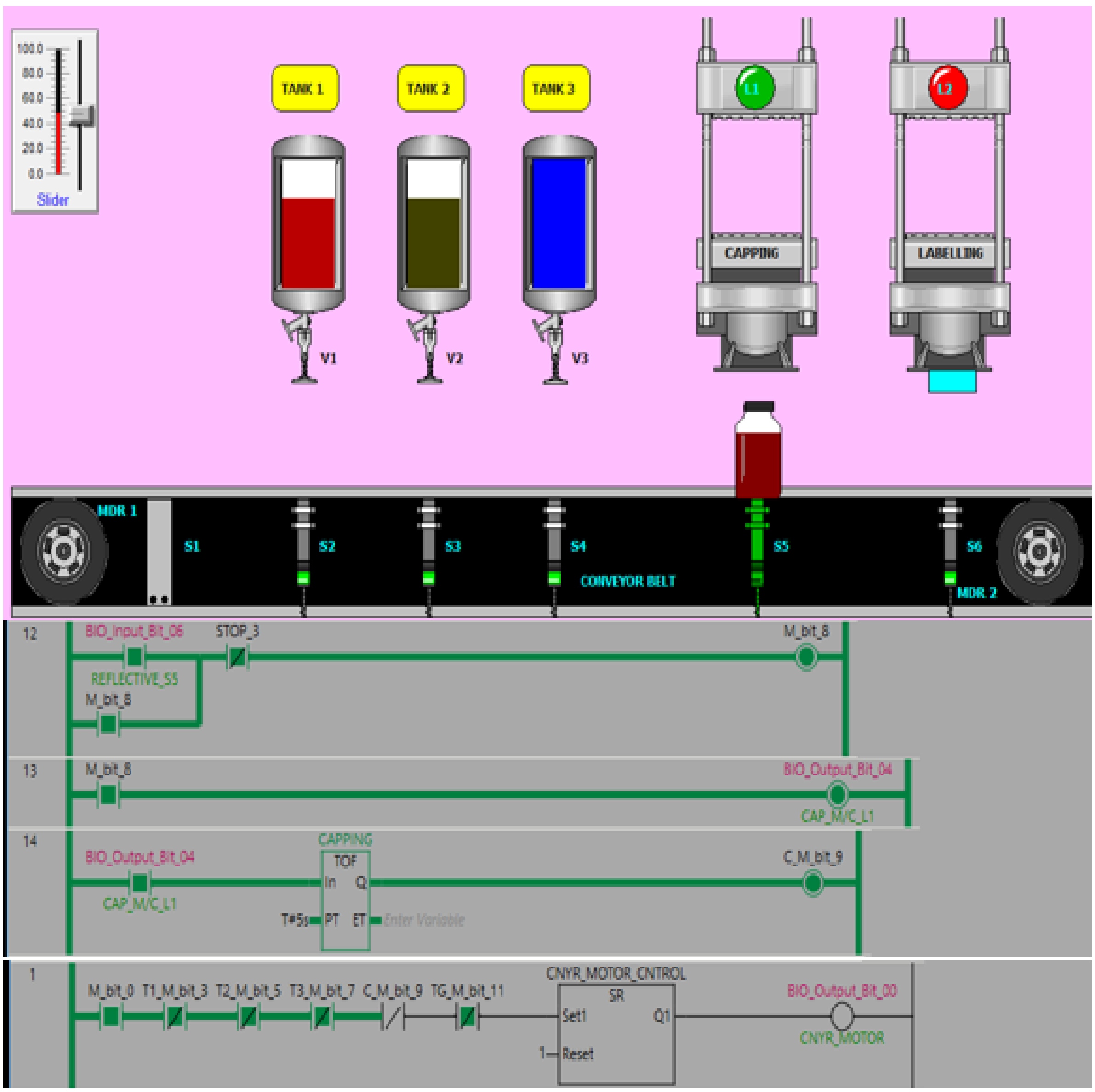


Fig 33: Capping with code Case B

6) Now the (S6) Photo-electric retro-reflective sensor is detecting moving container and due to which, conveyor motor stops, L2 LED turns into green from red. The container got blue label by labelling unit. The moving of conveyor belt continues after some predefined time i.e. time required to label the container.

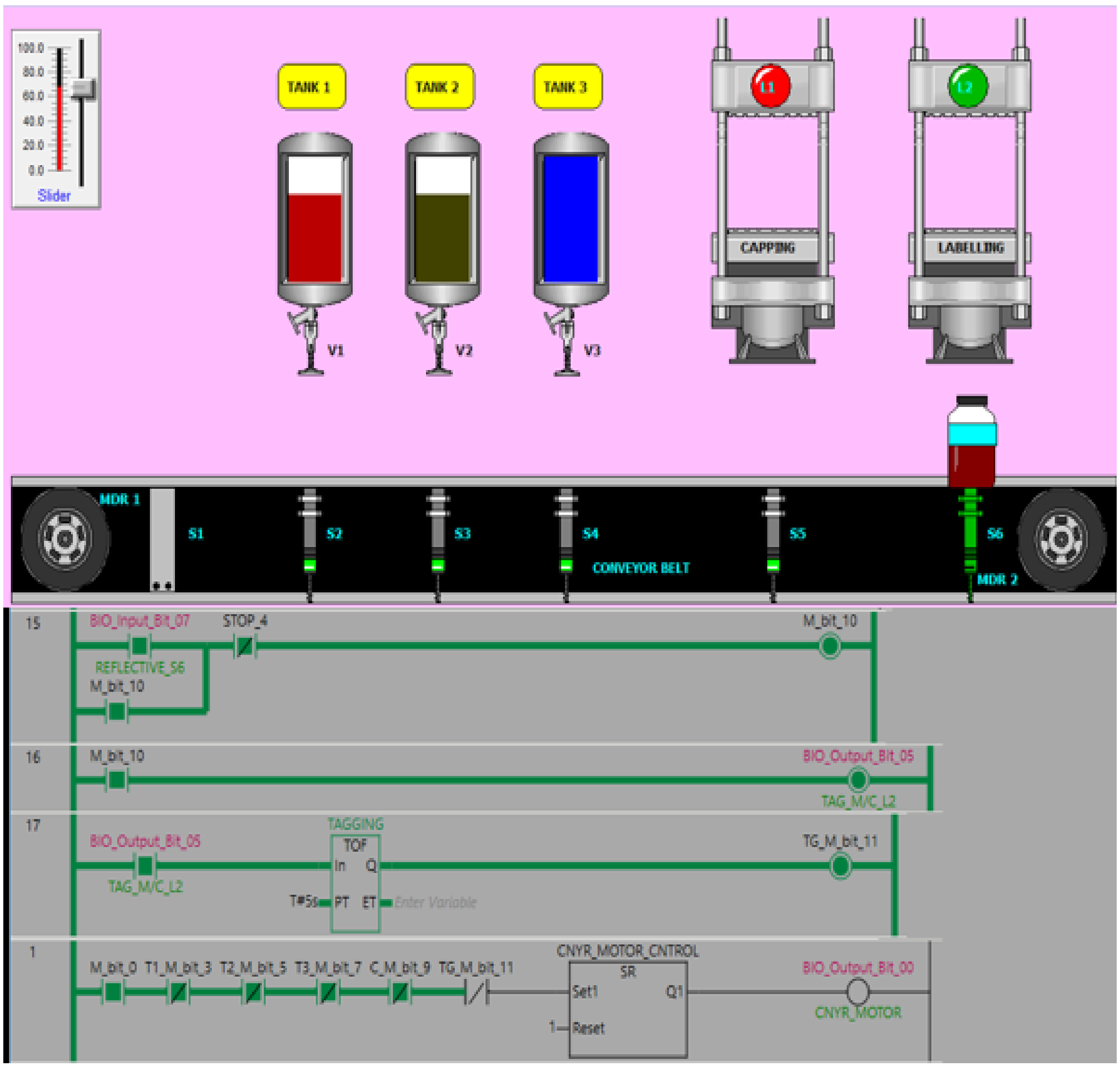


Fig 34: Labelling with code Case B

7.4 COUNTING OF CONTAINERS

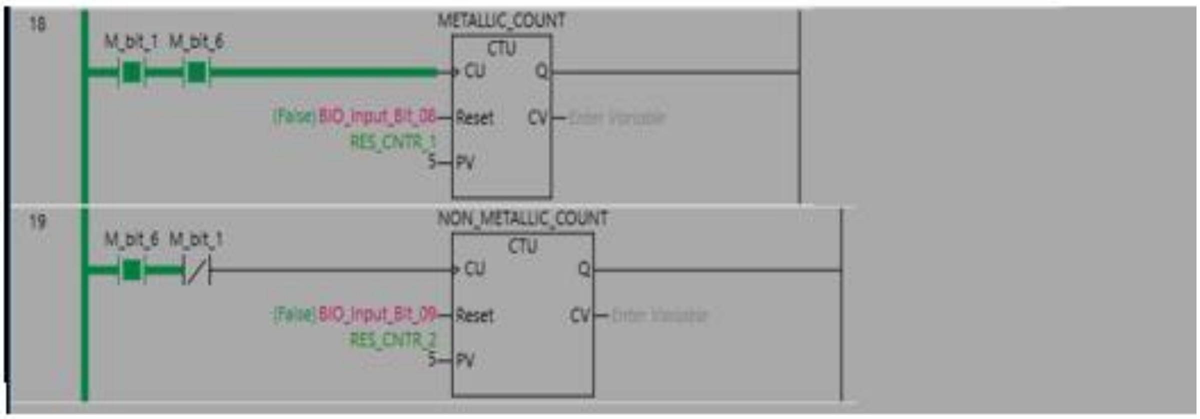


Fig 35: Counting Code

CHAPTER 8

CONLUSION AND FUTURE SCOPE

Monitoring and Controlling of the Bottle filling process using PLC and SCADA simulations has been demonstrated and carried out successfully. The simulation using PLC and SCADA is quite useful for other automated industrial operations and for Warehouse Management Systems where storage can be facilitated using simulation techniques. The use of highly accurate proximity sensors along with Programmable Logic Controller make the system very efficient over the earlier hard-wired systems.

The future scope of this project would include carrying it out on a real setup and understanding the whole process and economic impact. The work can also be employed with various sizes, shapes, and weights of the bottles and densities of the liquid. A level sensor can be used to detect the level of liquid filled. The process can be made more efficient using flow sensor which detects the liquid flow. Other applications and protocols like EtherCAT could be configured to allow monitoring of plant information from a decentralized server or using internet services. The process can be implemented on other controllers in order to compare the cost effectiveness. A numerical analysis of the space management, total cycle time and system throughput can be carried out which will reveal the advantage, cost savings in comparison to manual operations.

REFERENCES

[1] A global nonprofit partnership of industry, government and academia. The Industrial Internet Consortium; 2014, http://www.iiconsortium.org(retrieved 15.04.2016).

[2] Neves Valadão, Y.D., et al. (2018), "*Industrial wireless automation: overview and evolution of WIA-PA",* IFAC Papers OnLine 51(10), 175–180

[3] Shinde, K.S., Bhagat, P.H. (2017), *"Industrial process monitoring using IoT*", International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017)

[4] Kahiomba Sonia Kiangala and Zenghui Wang (2019), "*An Industry 4.0 approach to develop auto parameter configuration of a bottling process in a small to medium scale industry using PLC and SCADA*", Procedia Manufacturing Volume 35, 2019, Pages 725-730

[5] Kagermann H, Helbig J. Recommendations for implementing the strategic initiative INDUSTRIE 4.0 (2013), http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_websit

/Acatech/root/de/Material_fuer_Sonderseiten/Industrie_4.0/Final_report__Industrie_4.0_accessible .pdf (retrieved 15.04.2016).

[6] Marco Colla (2009), "*Design and Implementation of industrial Automation Control System: A Survey*", Industrial Informatics, 7th IEEE International Conference on Digital Object Identifier; IEEE; 2009. Pp. 570-575.

[7] Alphonsus, E.R., Abdullah, M.O. (2016), "*A review on the applications of programmable logic controllers (PLCs)*", Renew. Sustain. Energy Rev. 60, 1185–1205

[8] Saeed AUA, Al Mamun M, Zadidul Karim AHM (2012), "*Industrial application of PLCs in Bangladesh*", Int J Sci Eng Res (IJSER) 3(6). ISSN: 2229-5518

[9] Frank Petruzella, "Programmable Logic Controllers", Fifth Edition, Mc Grawhill Education, (2017).

[10] Gerardo Gonzalez-Filgueria, "*System of Automation of an Industrial plant of Elaborating and Packing of Industrial Detergent"*, University of A Coruna; Ferrol; Spain.

[11] L.Y. Anton, S. Prabuwono and E. Golkar (2011), "*Feature Extraction Algorithm for Fill Level and Cap Inspection in Bottling Machine*", International Conference on Pattern Analysis and Intelligent Robotics, 2011, PP.47-52.

[12] ZHANG Tianxia, DONG Feng, YUAN Hao, "*Application of PLC for Arranging Bottle in Beer Filling Production Line*", Tianjin Key Laboratory of Process Measurement and Control; School of Electrical Engineering and Automation; Tianjin University; Tianjin; China.

[13] Jaykumar Patel, Prof A1peshkumar Patel, Mr. Raviprakash Singh (2014), "*Development of PLC Based Process Loop Control for Bottle Washer Machine*", 2nd International conference on Innovations in Automation and Mechatronics Engineering; ICIAME 2014.

[14] Kevin Collins, "PLC Programming for Industrial Automation".

[15] Katre S, Hora J (2016), "*Microcontroller based automated solution filling module*", Int Res J Eng Technol (IRJET) 03(05)

[16] Mashilkar B, Khaire P, Dalvi G (2015), "*Automated bottle filling system*", IRJET 02(07). ISSN: 2395-0072

[17] Y. Neves Valadão, G. Künzel, and I. a. P. Müller, C. (2018), "*Industrial Wireless Automation: Overview and Evolution of WIA-PA,*" in International Federation of Automatic Control, 2018.

[18] Somavanshi AP, Austkar SB, More SA (2013), "*Automatic bottle filling using microcontroller volume correction*", Int J Eng Res Technol (IJERT) 2(3)

[19] A. Tiwary, A.K. Patro and J.B. Padhy (2015), "*Beverage Preparation and Bottling Unit using PLC & SCADA System as a Proposed Automation for a Small Scale Industries*", International Journal of New Technologies in Science and Engineering Vol. 2, Issue. 5, 2015, PP.81 – 86.

[20] Nadgauda, N., & Muthukumaraswamy, S. A. (2019), "*Design and Development of Industrial Automated System using PLC-SCADA*", 2019 IEEE 10th GCC Conference & Exhibition (GCC).

[21] J. Patel (2015), "*PLC based automatic bottle filling*", International Journal of Engineering

Research and General Science Volume 3, Issue 3, 2015, PP 136 – 144

[22] Cosmina IIIes, Gabriel Nicolae Popa and Ioan Filip (2013), "*Water Level Control System Using PLC and Wireless Sensors*", IEEE 9th International Conference on Computational Cybernetics; July 2013; Tihany; Hungary.

[23] Kunal Uttekar, Rahul Gosavi, Sarthak Lad, Jagruti Kamat (2011), "*Implementation of Sensor Network for Automation of Bottle Capping Assembly Station Using IEEE 802.15.4*", 2nd International Conference on Instrumentation Control and Automation; 15-17 November 2011; Bandung; Indonesia.

[24] E. R. Alphonsus and M. O. Abdullah (2016), "*A review on the applications of programmable logic controllers (PLCs)*," Renewable and Sustainable Energy Reviews, vol. 60, pp. 1185- 1205, 2016, doi: 10.1016/j.rser.2016.01.025

[25] Htet Htet Aung, Thae Thae Ei Aung (2019), "*Simulation and Implementation of PLC based for Nonstop Filling Process using PLCSIM and HMI*", International Journal of Creative and Innovative Research in All Studies, Volume 2, Issue 3

[26] Zar Kyi Win, Tin Tin New (2019), "*PLC Based Automatic Bottle Filling and Capping System*", International Journal of Trend in Scientific Research and Development (IJTSRD), Volume 3 Issue 6, October 2019

[27] N. Nadgauda, Senthil Arumugam Muthukumaraswamy, S. U. Prabha (2020), "*Smart Automated Processes for Bottle-Filling Industry Using PLC-SCADA System*", Chapter in Intelligent Manufacturing and Energy Sustainability, pp.693-702

[28] S. V. Viraktamath, A. S. Umarfarooq, Vinayakgouda Yallappagoudar and Abhilash P. Hasankar (2020), "*Implementation of Automated Bottle Filling System Using PLC*", Chapter in Inventive Communication and Computational Technologies, pp.33-41

[29] Mini Sreejeth and Shilpa Chouhan (2016), "*PLC based Automated Liquid Mixing and Bottle Filling System*", 1st IEEE International Conference on Power Electronics. Intelligent Control and Energy Systems (ICPEICES-2016)

[30] G. Pradhan et al (2018), "*Automation of tank level by using PLC and SCADA*", /HMI, Int. J.

Adv. Res. Ideas Innov. Tech. 4 1935.

[31] S. L. Kulkami and M. Elango (2016), “*Development of PLC based controller for bottle filling Machine*”, International Journal of Innovations In Engineering Research And Technology [IJIERT], Volume 3, Issue 4, 2016, PP.1-10.

[32] Savita and Lokeshwar (2014), “*Implementation and Performance Analysis of Bottle Filling Plant Using Ladder Language*”, International Journal of Science and Research (IJSR), Volume 3 Issue 7, 2014, PP. 1803 – 1806.

[33] Boyer SA (2009), “*SCADA—Supervisory control and data acquisition*”, 4th edn. International Society of Automation, USA

[34] Chakraborty K, Roy I, De P (2015), “*Controlling process of a bottling plant using PLC and SCADA*”, Indonesian J Electr Eng Inf (IJEEI) 3(1):39–44. ISSN: 2089-3272

[35] Dakre A, Sayed JG, Thorat EA, Chaudhary AMA (2015), “*Implementation of bottle filling and capping using PLC and SCADA*”, IRJET 02(9). ISSN: 2395-0072

[36] K. Chakraborty, M.G. Choudhury, S. Dasc and S. Paul (2020), “*Development of PLC-SCADA based control strategy for water storage in a tank for a semi-automated plant”*, Published by IOP Publishing for Sissa Medialab, Journal of Instrumentation, Volume 15, April 2020

[37] J. Dhiman and E.R. Dileep Kumar (2014), “*Hybrid method for automatically filling of the chemical liquid into bottles using PLC & SCADA*”, International Journal of Engineering Research and General Science, Volume 2, Issue 6, ISSN 2091-2730, 2014, PP.1000 –1007.

[38] A.I. Abashar, M.A. Mohammedeltoum and O.D. Abaker (2017), “*Automated and Monitored Liquid Filling System using PLC Technology*”, International Conference on Communication, Control Computing and Electronic Engineering (ICCCCEE), Khartoum, Sudan, 2017, PP. 1-5.

[39] T. Al-Hawari, F. Aqlan, M. Al-Buhaisi, and Z. Al-Faqeer (2010), "*Simulation-Based Analysis and Productivity Improvement of a Fully Automatic Bottle-Filling Production System: A Practical Case Study*", vol. 4, ed, 2010, pp. 195-199.

[40] Birmole, P., et al. (2018), “*Designing and implementation of chemical mixing and filling bottles using PLC*”, In: Proceedings of the 2nd International Conference on Inventive

Communication and Computational Technologies (ICICCT 2018), IEEE (2018)

[41] P. Birmole, M. Kamble, S. Naik, A. Sadamate, and H. V. Korgaonkar (2018), "*Designing and Implementation of Chemical Mixing and Filling Bottles Using PLC*", presented at the Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018), 2018.

[42] Kurniawan, D., Sulaiman, R. (2008), "*Design and implementation of visual inspection system in automatic bottling system based on PLC*", pp. 760–764

[43] Ahuja H, Singh A, Tandon S, Shrivastav S, Patil S (2014), "*Automatic fillingmanagement system for Industries*", IJETAE 4(Special Issue 1)

[44] J. Pannu, R. Kulkarni, and M. S. B. Ranjana (2016), " *On the automated multiple liquid bottle filling system*," in 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), 2016, vol. 60: IEEE, in Renewable and Sustainable Energy Reviews, pp. 1185-1205

[45] Kalaiselvi T, Praveena R (2012), "*PLC based bottle filling and capping using user defined volume selection*", ISSN: 2250-2459

[46] T. Kalaiselvi, R. Praveena, Aakansha .R, Dhanya (2012), ''*PLC Based Automatic Water Bottle Filling Capping System with User Defined Volume Selection''*, IJETAE, vol. 2, Issue8, (August 2012).

[47] Laurentiu Schiop, Marian Gaiceanu, "*Mathematical Modelling of Color Mixing Process and PLC Control Implementation by Using Human Machine Interface*", Dunarea de Jos University of Galati/Department of EIectrical Engineering; Galati; Romania.

[48] http://www.google.com/

[49] http://www.youtube.com/

[50] OMRON SYSMAC STUDIO PLC Software