ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

DEPARTMENT OF ELECTROMECHANICAL ENGINEERING

Final Year Project Paper for The Fulfillment of Bachelor's Degree in Electromechanical Engineering

Title – Design and simulation of automatic defected bottle inspection and rejection system using PLC with computer vision

PREPARED BY	ID No.
1. BEMNET MARKOS	ETS 0221/10
2. BEMNETU MOLLA	ETS 0222/10
3. MELAKU YEMANEBERHAN	ETS 0785/10
4. MULAT GETAYE	ETS 0882/10
5. ROZA MENGESHA	ETS 1016/10

Advisor: Mr. Mitku B.

Addis Ababa, Ethiopia

JUNE 20, 2022

DECLARATION

We hereby declare that this project belongs to us. We have done a project titled "**Design and simulation of automatic defected bottle inspection and rejection system using PLC with computer vision**". we were fully engaged to write this project paper.

We assert the statements made and conclusions are drawn as an outcome of our research work. We further certify that

I. The work has not been submitted to any other institution.
II. The work contained in the paper is original and has been done by the group members only under the general supervision of our supervisors.
III. We have followed the university guidelines in writing the project paper.
IV. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and given their details in the references.

Finally, we would like to tell you we invest a lot of energy to prepare this project paper and widen our knowledge from different perspectives, this can be seen in every detail of the paper.

Group members	Signature
1. Bemnet Markos	______________________
2. Bemnetu Molla	______________________
3. Melaku Yemaneberhan	______________________
4. Mulat Getaye	______________________
5. Roza Mengesha	______________________

Approved by

Advisor Mr. Mitiku B. ______________________

Date ______________________

ACKNOWLEDGMENT

First of all, we would like to thank the almighty God for helping us throughout our project completion. Second of all, we owe sincere and earnest thankfulness to our advisor Mr. Mitiku B. for the support guidance, and patience, he showed to us throughout the thesis writing and technical guidance. We would also like to show our gratitude to Mr. Yemane G/Meskel for his support in writing our project paper.

Finally, sincerely and heartily grateful to our families and friends who boosted us morally, we are sure this would have not been possible without their help.

ABSTRACT

This paper dwells upon the automated inspection and rejection system of a defected bottles. Image processing is becoming a vital part of production industries to define the efficiency and accuracy of the bottles become more important. To improve the quality of the line, this project deals with an Automatic vision inspection machine that contains vision inspection parts and a conveyor line implemented using PLC and proximity sensors, and a Mobile camera. Stream of real-time videos is acquired from the camera to MATLAB using a wireless network, there were methods and algorithms used in the visual inspection to analyze and detect the presence and properties of a bottle and then extract meaningful information using Graphical User Interface, Noise or unwanted objects were mostly removed using filtering, segmentation and blob analyses. Then the inspection result is given to the PLC as an input, here we analyze the simulation study of one programmable logic controller (PLC) based control system by building up a program using a ladder diagram. This paper also presents a comprehensive simulation of the performance of the sensors used for sensing the presence of bottles in the moving conveyer belt which are ready for inspection monitored by PLC. And also, we have developed motion analysis and stress analysis in SolidWorks. Also, for better visualization, we have developed logical animation in factory IO software. The design, simulation, and analysis of the whole system is done with the help of suitable software, such as MATLAB, Arduino, SolidWorks, and Factory IO. The experimental results of the system show that its accurate if the camera is positioned at a stable, fixed distance from the bottle, at a fixed speed of the motor which can impact the results of the high speed is adjusted to the motor, the defect detection degree has reached a high level with good accuracy in detection.

Keywords: *Bottle inspection, MATLAB, Simulation, Segmentation, Blob analysis*

TABLE OF CONTENTS PAGE

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AC	Alternating current
APIM,	Accessory protocol interface module
AT	Assistive technology
AVI	Audio visual informations
CAD	Computer aided design
CCD	Charge couple device
CMOS,	Complementery metal oxide
COM	Component object model
CV	Constant velocity
DBI	Database interface
DC	Direct current
DOF	Degree of freedom
FBD	Functinal block diagram
GB	Gigabite
GUI	Graphical user interface
ICI	Internate computer integration
IO	Input output
IPC	Internate personal computer
MKS	Meter kilogram second
MMF	Memory maped file
PC	Personal computer
PIR	Passive infrared
PVC,	Polyvinyl chloride
ROI	Resolution independat object
SF	Single frequency

CHAPTER ONE

INTRODUCTION

1.1 Introduction

For industrial purposes, there is a recognized hierarchy of damage control measures based on the principle that loss of production needs to be reduced to an acceptable level by engineering means. Automation is the use of control systems and information technology to reduce the need for human work in the production of goods and services. In the scope of industrialization, automation is a step beyond mechanization. Whereas mechanization provides the human operator with machinery to assist them with the muscular requirements of work, automation greatly decreases the need for human sensory and mental requirements. In the filling industry, a faulty bottle detection routine is an imperative part being done by several discrete mechanisms working in harmony. So, in the industry the whole system is done by many separate machines like faulty bottle detection system, filling machine, capping machine, labeling process, etc. Here the focal argument issue is about detecting the faulty bottle before entering the main plant.

Machine-based vision inspection systems provide a huge understanding of objects that are desirable from a single image or many arrays of images. In industrial applications, machine inspection systems enhance productivity as well as improve the quality of the manufactured products, in this chapter we are going to demonstrate or study historical background problems to be solved, the scope, and constraints of the project.

After the inspection is made for the rejection part it can be used PLC system which allows using implying delay, timers, and counters.

1.2 Background and justification

Captured input images are the main objective of this system. Application of the vision system in industries to automate the manufacturing process is considered Automated visual inspection (AVI) when attempts are made to inspect, control products, and recognize the defects by using only images of the products. Human inspectors are slower and their efficiency is affected by their state of illness, exhaustion, or other human shortcomings. In some applications, they need sometimes special environments which are dangerous and not conducive for human operation. On the other hand, especially in the manufacturing environment, it is necessary to improve quality control and productivity.

Inspection is carried out by machine vision system via image processing technique is an application in beverage and food industries, milk industries, medicine industries, and other chemical product industries. In this area, accurate filling, an inspection of cap closure, sorting, recycling of plastic bottles, recognition between glass bottles and pet bottles, inspection for over-fill or under-fill, verification of label quality, and detection of defected products are inspected automatically. The industrial vision system is not able to handle all tasks in every field of application. It means that each system due to its specific requirements and its limitations needs particular techniques that should be considered.

The basic components for all bottle visual inspections as for any other VIS are shown. The first step in a visual inspection system is image acquisition, which is concerned with capturing a good quality image through a camera. Camera resolution, its position, the color of the background, speed of the conveyor belt, and light(brightness) have an important effect on image quality. In this paper, we have shown the basic principle of bottle visual inspection by presenting different methods and techniques for automation of bottling and the most recent algorithms have been stated. In this paper, we have concentrated on developing a combined algorithm for four quality inspections in bottle caps: Detection of tamper in side view, Detection of tamper in top view, and Detection of without cap condition of the bottle on a conveyor and different colored cap of the bottle on a conveyor. So, by developing such an algorithm where the four most important parameters of automation of bottling are combined we get an efficient bottling system. Besides we have presented a detailed explanation of image processing of bottle cap detection and a new algorithm combining the four parameters through a list of figures. MATLAB, this software will provide various digital image processing techniques used for obtaining required information from an acquired image. Based on the extracted image the processor will make a decision i.e. acceptable or not acceptable. Thus, the NOT acceptable components will be removed from the production line using an ejection mechanism such as an actuator or blower.

1.3 Problem statement

One of the biggest challenges in a production line is getting a product that has high quality and zero defects. Consider in any bottle seals and caps are used to protect the product from leakage, spoilage, and tampering. Damaged seals or caps have a direct impact on company profitability due to waste, it makes customer rejection and potential liability. The integrity of these seals and caps is also a signal to the consumer about the product's safety and quality. Producers must find reliable and cost-effective means of detecting these defects and weeding them out before they get into the supply chain. Conventional sensor-based inspection methods have proven to be expensive and difficult to set up the system and change. Human inspectors are slower and their efficiency is affected by their states of tiredness, illness, or human shortcomings, In some cases, they don't have proper skills and in some toxic and hazardous cases, human inspectors are not sufficient. So, using a machine vision system for the inspection process overcomes this problem. In industrial inspection systems to reduce human efforts and improve the quality of inspection machine vision system is being developed and automate the inspection process.

1.4 Objectives

1.4.1 General Objective

The general objective of the project entails the **Design and simulation of an automatic defected bottle inspection and rejection system using PLC with computer vision**.

1.4.2 Specific Objective

- ✓ To design a webcam-based inspection system via MATLAB to inspect bottles that arrived at the position determined in the conveyor.
- ✓ To design a graphical user interface in MATLAB.
- ✓ To simulate the rejection system in PLC.
- ✓ To detect and analyze the properties of the bottle's cap, liquid level, deformation, and so on.
- ✓ To develop 2D or 3D part and assembly design of the mechanical components of the belt conveyor using SolidWorks.
- ✓ To simulate the inspection and rejection system in SolidWorks and factory io.
- ✓ To develop stress analysis of the conveyor in SolidWorks.
- ✓ To develop the controlling mechanism of the conveyor.

1.5 Significance of the Project

In the production process industry especially beverage production, bottles mostly experience defects that appear in their mouth, cracks in the bottle's upper and bottom parts, and mismatched liquid levels are most cases production company experience. These defects have increased in the industry invention and many types of research to ensure quality inspection machines to guarantee the product's quality, reliability, and continuity. Overall design and development of PLC-based Automatic faulty bottle rejection system are to find the best solution for the problems that are stated. It also increases the country's economic development.

1.6 Scope of the project

The scope of this project tends to the design and simulation of a fully automatic bottle inspection and rejection system. The design section includes electrical, mechanical design, and programming development. In electrical design, we included a control and rejection system. The mechanical design section includes the design of the belt conveyor.

Reason	❖ Why do we need a project? What problem are we trying to address? ▪ To implement an automated inspection and rejection system in the beverage industry.
Aim	❖ What do we want to achieve? ▪ To design and simulate an automatic bottle inspection and rejection system which can be used for industry quality mass production.
Important activities	❖ What activities will project team members undertake? ▪ Design every mechanical system of the machine on SolidWorks. ▪ Design motion study in SolidWorks. ▪ Develop programming code on Arduino and MATLAB. ❖ Who will do the activities? ▪ Group members. ❖ When will they do the activities?

	▪ Five days per week until the goal is implemented.
Deliverables	❖ What outputs will we have at the end of the project? ▪ A system that can inspect and reject the defective bottle.
Period	❖ How long will it take to accomplish the project? ▪ Two months
Evaluation plane	❖ How will we know that the project is successful? How will we measure this? ▪ We will measure the progress per week so, we can predict the project will going to be successful. ▪ Test on the prototype level.
Project lead	❖ Who will lead the project? ▪ All the group members by taking different tasks.
Team members	❖ Who will be involved in planning and implementing the project? ▪ The project owners and the company project manager.

Table 1. 1 Scope of the project

1.7 Limitations of the project

- The main limitation extract the object from the desired bottle if the unwanted object is introduced to the inspection area.
- Getting the appropriately selected materials based on the design for making the prototype was very difficult.
- Difficult to simulate with a small capacity of a computer processor.
- Needs a high-quality camera for better image acquisition.

1.8 Structure of the project

The remaining parts of this research project will be organized and modeled as follows:

- Chapter two deals with the review of the literature and relevant research associated with the image inspection and rejection systems, discuss and analyses algorithms their results, and their evolution of design and work done to reach this highly enhanced vision and rejection system.

- Chapter three defines the research methodology of image inspection machines using MATLAB and rejection systems in PLC and also, and we are going to discuss the hardware design and procedures used in programming software simulation also its hardware component programming.
- Chapter four contains the implementation of the system using procedures we discussed in chapter 3, Analysis of the result of each process of implementing wireless communication base webcam image processing with using PLC to inspect and run the conveyor with a stepper motor and Sensors that detect and position the bottle.
- Chapter five is the result and discussion of the system and the outcome of achievement or objectives. we reached during the methodology, design, and implementation of the project.
- Chapter six finally is the conclusion of the system and recommendations for future research and developing or advancing such a project.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The recent technological development of data and image acquisition systems has directed the development and invention of new machine vision solutions in a variety of manufacturing sectors. In this chapter, we are going to discuss and analyze the historical background of bottle inspection visual machining using algorithms to monitor bottles moving on the conveyor after sensors detected the presence of a bottle and the rejection system if a defect is present.

Image processing and vision machines are one of the most important sectors of manufacturing industries and it's among the technologies that are rapidly growing with their major and various applications to enhance and analyze the properties of the product to ensure the quality and safety of the product.

Using the automated technology in helping solve image inspection problems including bottle cracks, lips of the bottle, cracking in the mouth of the bottle, the level of the liquid is not standard, etc. these issues in the industry with its research done have discovered many quality inspection machines to guarantee quality, reliability, and continuity of the product, like sensors that take place quality inspection or even image processing.

In the case of bottle filling, most conventional machines consider only controlling the level of the liquid. Nowadays technologies there are many vision solutions in other sectors of the manufacturing industry like product selection, raw material washing and final product packaging, and other systems that pose vision machines. Present case solutions which are applied after filling of the liquid in the bottle takes place before it is sent to the final consumer, it's so important to inspect for defects in several ways including if the bottle is not damaged, the label is correctly in position, bottle caps, level of the liquid, and uniqueness or likelihood of the liquid in the process. Any of the issues mentioned above pose unique analysis algorithms and image processing that are different from each other. And these algorithms have to be discovered using gray and binary and color images (Campos, Ferreira, Martins, & Santos, 2010).

In a literature review, with the advancement in technology, a variety of inventions are made on quality inspection of the bottles manufactured. Ufuk Sanver, Erdem Yavuz, and Can Eyupoglu 2017 investigated mineral water bottles to detect defects and faulty before packaging using image processing with an algorithm of edge detection called Sobel method and also Haugh transform for counting several bottles, the system they designed has achieved the huge improvements in detecting bottle which is faulty from a couple of test samples.

(Li, Hang, Yu, Wei, & Xinyu, 2017), the study they did propose shows a method of defect detection of a mouth class using or based on the Open CV connectivity domain feature to detect defects of the bottle. (Zhang et al, 2018) Presented Bayesian Bottle Size Detection Classifier especially bottles used in pharmacy by exposing Open CV function to the processing and also identifying of the type of the bottle by using Naïve Bayesian classifier and also, they figured out the bottle characteristics.

The study of (Yang et.al, 2015) showed a method of detection of defects method in the mouth glass of the bottle by using image processing. First, image smoothing is taking place to eliminate any sort of noise in the image, then segmentation of the image by thresholding the level of gray image finally edges detection extraction takes place. When compared to other methods of solutions in finding edge detection in this study Canny operator was adopted as was a base in this study.

Study presented by Leila Yazdi, Anton Satria Prabuwono and Ehsan Golkar, 2011

Summarized in an automated visual inspection system, the algorithm proposed a new feature extraction technique to investigate these two parameters to detect liquid fill level defects and close bottle caps. In the study by Sile Ma, Bin Huang, Huajie Wang, and Junmei Guo, an inspection of a defective mouth and bottom base empty bottle were presented using straight lines to find points in the bottle mouth position and the position of the bottom of the bottle by random Hough transform is used to inspect the circle defect at the bottom of the bottle.

2.2 visual measurement

It is the case for bottle filling, where conventional systems are only concerned with level control. Vision solutions have nowadays been used in other sub-sectors of the manufacturing systems, such as the final product packaging, washing raw materials, selection of a product, etc. Here is the case of the present solution, which intends to be applied after the filling of beverage bottles that are going to place in the crates. Before these bottles are boxed are sent to the final customer, several items must be inspected, including If the bottle has the right shape and is filled well if the bottle is damaged if the bottle has a capsule and if the capsule is correct. The system is currently installed and integrated into the production line and in final adjustment testing (Campos et al., 2010). One of the computer vision system's primary requirements was that it had to adapt to clear line space. This requirement has affected the layout of the vision system and various algorithms ' specifications and developments. The direct result of this requirement is the small size of the light box and the resulting low light uniformity, which was ideally intended to be uniform and diffuse. The distance from the bottle to the camera, which also had to be reduced, is another consequence. This means that a lens with a small focal length is used to create a strong perspective effect in the image (Campos et al., 2010) In these years, the APIM has developed to meet pharmaceutical factory requirements. To avoid false detection, the idea of double-checking is carried out that two different cameras inspect a product twice. However, these changes are still based on the monocular camera algorithm, and because of the inherent shortcomings, there are two important problems listed below that cannot be easily solved at low cost either in computing time or hardware consumption:

1) Narrow Field Depth (NDOF): We usually use a small focal lens and a large F-number aperture to get enough DOF. However, due to the small size of CCDs or CMOS, industrial cameras usually use C - Mount lenses, which cannot be viewed in wide depth due to changes in focal length and F-number, such as large infusion bottles with a radius greater than 25 mm. In addition, the aperture diameter is relatively smaller than normal. To get a greater depth of view, we need a larger F-number, but a larger F-number requires higher light source power, which leads to growth in hardware costs and cooling problems. As a result, the field depth is never sufficiently large and small particles cannot be imaged for the following processing.

2) *Dynamic Background Interference (DBI):* It is mainly due to two types of light interference, the external and the internal interference. The external interference from outside the APIM refers to the light indoors, the sun's rays, and so on. Internal interference is caused by multiple reflections within the APIM between the spinning surface of the liquid, machine pillars, scratched sidewalls, even adjacent bottles, etc. Under the conditions of the laboratory, these interferences can be easily separated by sequential images (Fang et al., 2013).

CHAPTER THREE

METHODOLOGY

3.1 Methodology flow chart

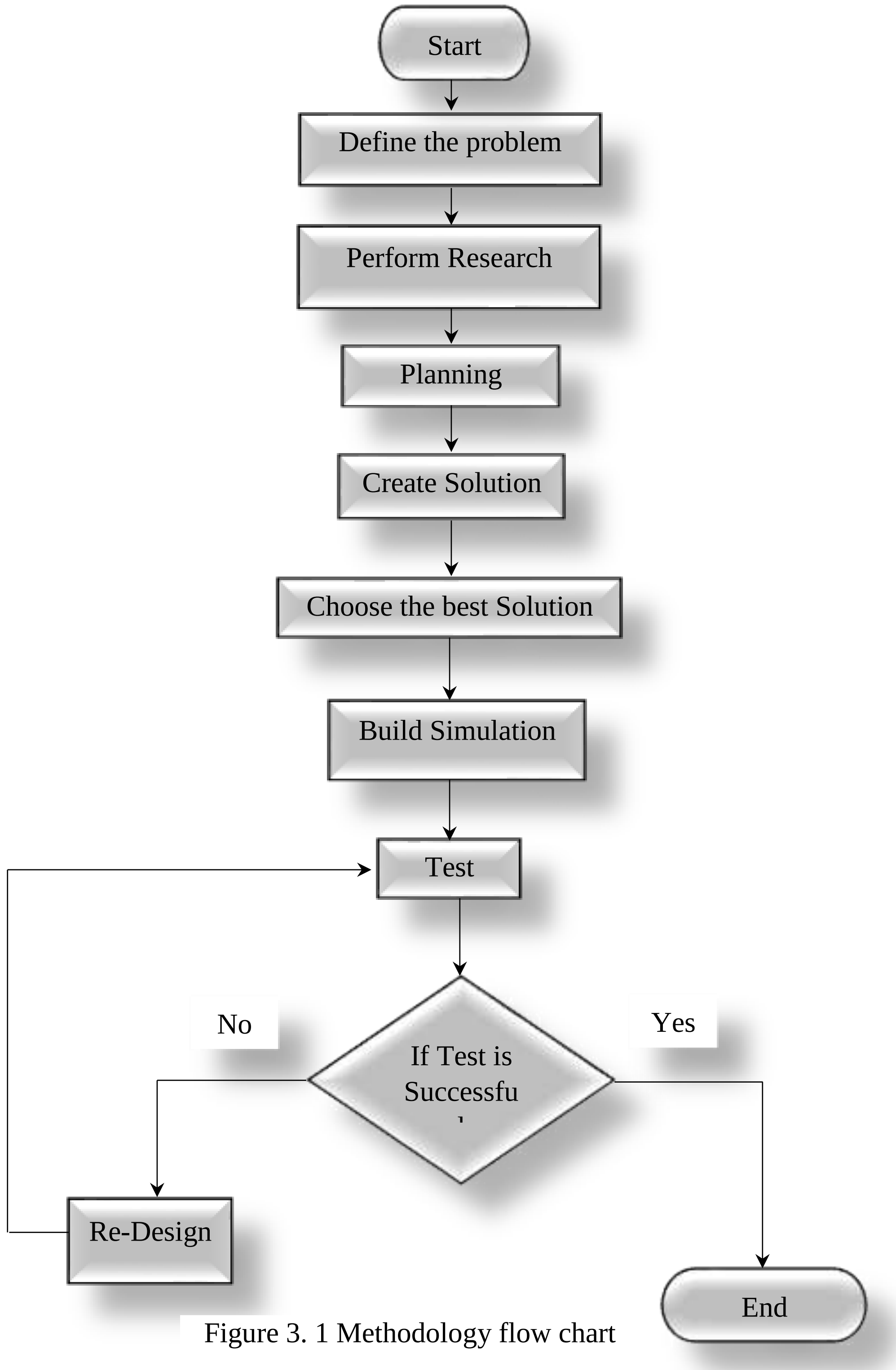


Figure 3. 1 Methodology flow chart

In this chapter, we are going to visualize methods used entirely in various activities that took place during the duration of the project. The quality inspection system is the main important factor in manufacturing systems, and it's what makes the industry meet its standard and specifications and also customer needs when the product is finally on the market. So, we are here to design a conveyor line using a variety of materials to meet our requirements, and the bottle inspection system is the next to inspect the bottle before its packaged for final processes.

3.2 Concept generation and Customer specification

3.2.1 Customer specification

The main customer specifications in the design of this machine are the problem statements that are listed in section 1.3 but also there are some additional customer specifications.

- ✓ Precise control of the faulty bottle detection
- ✓ Reduce the error of manual inspection
- ✓ Compact design or small space
- ✓ Reduced cost
- ✓ Fully automatic faulty bottle detection

3.2.2 Concept Generation

To answer the above customer specifications, we compare two faulty bottle detection system designs and by comparing them select the best method which meets the customer specification. As the system consists of two unit's inspection units and a rejection unit, the comparison is done in each unit by taking two methods and selecting the best. The different parameters that are used to compare the two methods in each unit are precision, Design compactness, cost of the system, time usage, reduction of the wastage, and efficiency.

I) Inspection unit

Parameters	**Manual inspection**	**Computer vision-based inspection**
Precision	Less precise	More precision

Cost	Periodical payment for employees, more cost	Cost for first installation and maintenance, less cost
Time usage	More time usage because it uses human power	Less time because it is automatic
Applicability	Is not applicable in all environment need a favorable environment for the employee	Can be applied in all environment
Efficiency	Less efficient	More efficient

Table 3. 1 Inspection Methods

II) **Rejection uni**t

Parameters	**Pneumatic system**	**Using motor and gear**
Precision	high	low
Efficiency	high	low
Time usage	fast	slow

Table 3. 2 Rejection Methods

Therefore, for the inspection and rejection system, computer vision and pneumatic system are selected respectively. These selected approaches are discussed below in detail.

3.3 Inspection Unit

3.3.1 Bottle Detection

The bottles pass from one end to another end of a conveyor belt. When the bottle comes under the sensing range of PIR sensor, bottles are sensed by a sensor. Then the signal will be sent to the inspection to be ready for inspecting the coming bottle through the conveyor.

3.3.2 Image Acquisition

Acquiring images from source hardware plays a vital role in this project, Machine vision system we are going to demonstrate consists of a Mobile Camera which is a 16 Mega-pixel camera are used to acquire a stream of images to inspect but for real applications high industrial level camera will be used, Background light source is used to minimize the level of noise and increase contrast, focus, and accuracy of the image, Personal Computer and Software Platform called MATLAB which we discuss later. An Android Application called Droidcam is used here to transfer live video wirelessly from mobile straight to MATLAB.

3.3.3 Bottle Inspection

In the stage when the system is fully run-up, the Droidcam app transfers live video wirelessly from mobile straight to MATLAB a series of images are collected in microseconds by the Camera for image processing, after the image is pre-processed in the methods we discussed earlier, we make extraction of the image to obtain the relevant information and properties we needed including the status of the bottle, the label of the bottle, the level and cap information, then defect analysis began to process if there are any problems in the bottle, and each of that process is done within a short period. The flowchart shown below provides the steps involved in image processing.

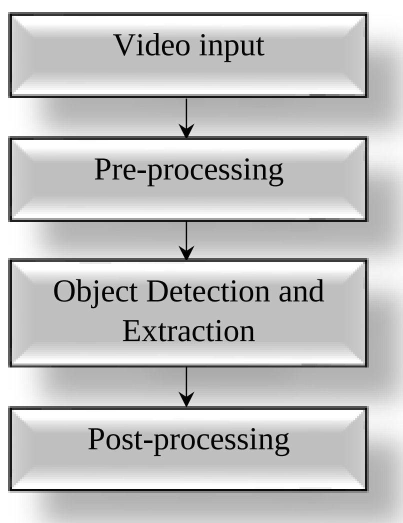


Figure 3. 2 Framework of defected and object detecting

3.3.4 Bottle Positioning

For bottle inspection to take place, it’s important to know that positioning a bottle in a camera's field of vision has an effective result, in here bottle to be inspected is in transmission to the destination at a certain speed, and the distance between the camera and bottle must be constant like 15cm, else the characteristics of the image captured at a certain time differ and causes a fault in the result. Therefore, to ensure that it is in the region of interest, it is necessary to calculate the position of the bottle bottom in each image accurately.

3.3.5 Video Processing

There are two possible ways of videos used as an input in MATLAB, Real-time lives streaming video from a camera source continuously monitoring specific areas and Stored video that is already recorded and easily obtained from storage. Image/Video preprocessing is a vital part of the analysis of the image especially when it comes to transforming into Gray-scale, binarizing image, enhancing and thresholding image segmentation also edge or corner detection. The image transformation includes converting the captured image into grayscale at a specific requirement of the subsequent processes of the image.

As in the figure below converting the image into grayscale and then into binary makes it easier to follow and analyze the specific regions of the image that had many layers in RGB mode. Due to the many factors affecting experimental conditions such as, various hardware and lighting factors, the images of the bottle collected in the image acquisition device may contain a lot of random noise, and there are more deficiencies in the recognition of the bottle. Pre-processing the image improves the region of interest and highlights the image.

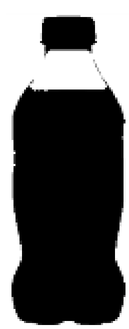


Figure 3. 3 Binarized bottle

These binarized effects of black and white, make it easy to follow any regional division by taking specified ROI (Region of Interest). Then segmentation of the image takes place to process the target area which poses its characteristics when compared to other non-targets and we extract the location we want to research if there is a defect to meet the requirements and goals we want to reach. In the image segmentation, there are many noises in the image that needs to be removed, thresholding is used to differentiate the image of the object and its background, and setting and having a threshold median value can help the system to know what and which is the image and background. The proper threshold can remove the white noise in the corner and sides of the target area of the image before extraction.

3.3.6 Graphical User Interface (GUI)

GUI is a friendly point-and-click interface that is used to display and monitor the results in the inspection process. In the figure below is the main prototype of the inspection.

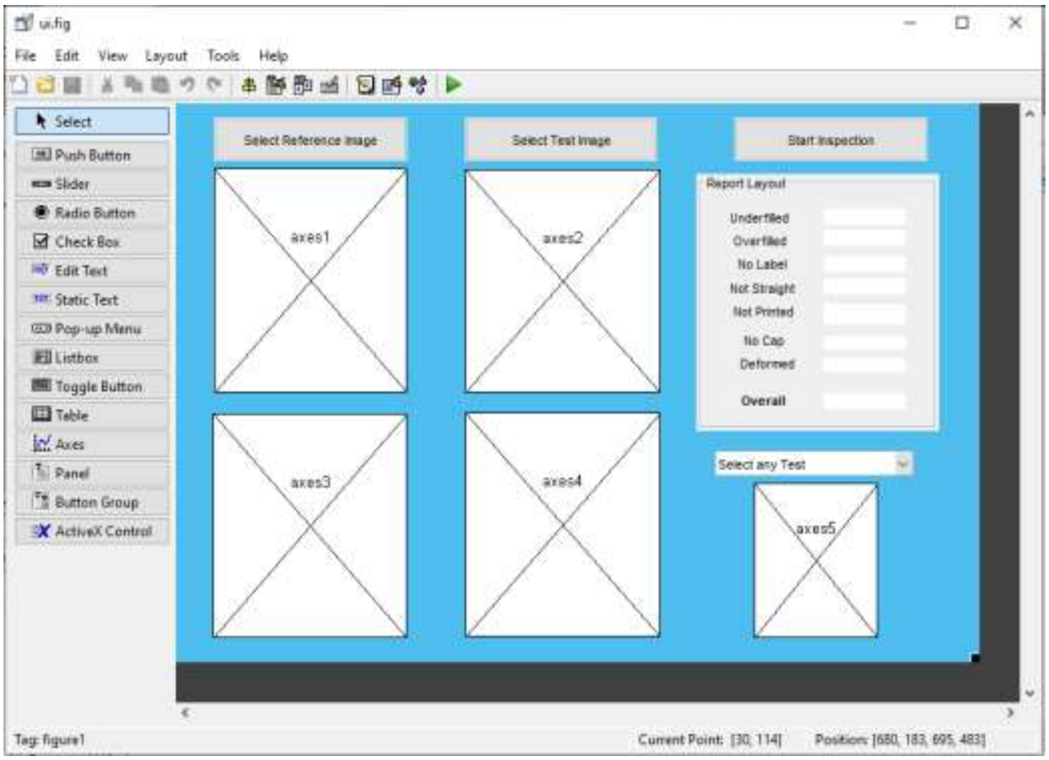


Figure 3. 4 Graphical user interface layout

Different axes are used in the figure above which show the Livestream of the visual field and the presence of the bottle in the area. An axis1 is used to monitor the main reference image, an axis2 is used to show the image that will be tested, and other axes are 3 and 4 which show the Segmentation process such as gray image and Binarizing image.

The figure 3.5 shown below is one of the important parts is where the report results are shown, to monitor the presence of a bottle and measuring the liquid level, defect detecting if there are no caps on it, the level of the liquid, the label of the bottle in that visual area.

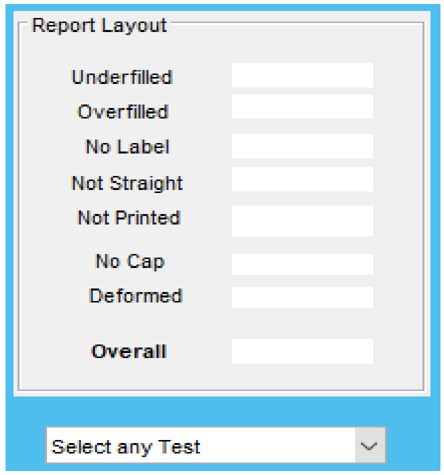


Figure 3. 5 Inspection report layout

3.4 Rejection Unit

When there is a coming defected bottle on the conveyor the signal will send from the Arduino output which is 5v this will energize the pneumatic pusher. After the pneumatic pusher is energized the defected bottle will be rejected from the conveyor. But if there is no any defect on the coming bottle the pneumatic pusher will not be energized as a result the bottle will delivered to the next line of production.

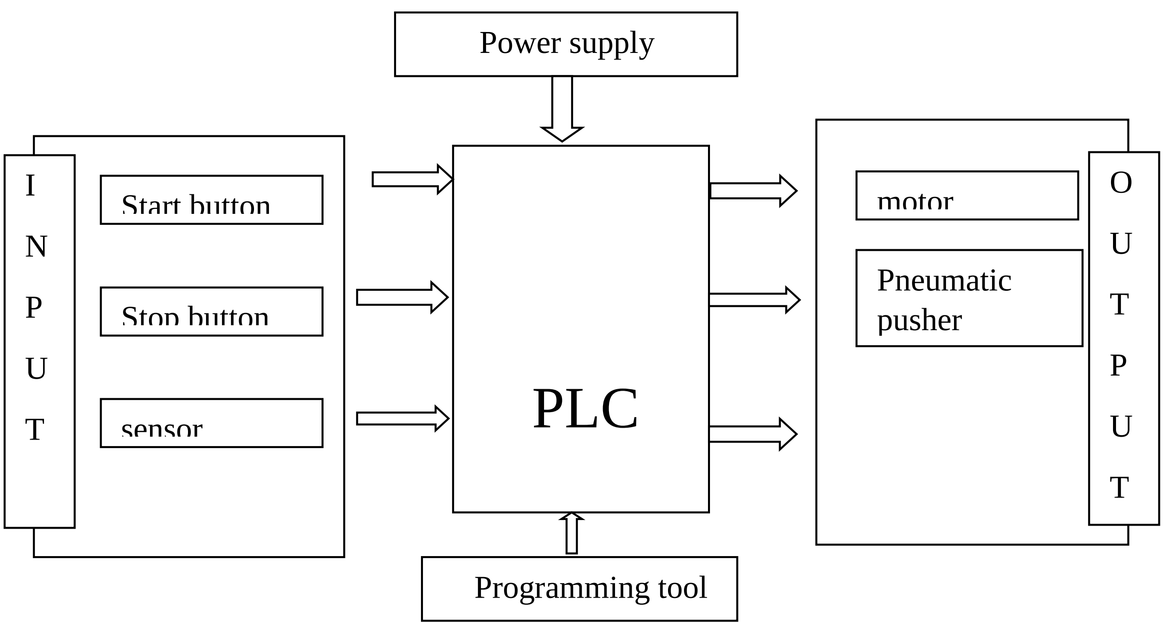


Figure 3. 6 Block diagram of the system

3.6 Block diagram of the project

The first 12V DC motor starts and allows the conveyor belt to move in different positions the bottles pass from one end to another end of the conveyor belt. When the bottle comes under the sensing range of the PIR sensor, bottles are sensed by the sensor. The camera then captures the image of the bottle continuously and sends them to a computer for processing, in real applications high-quality industrial cameras which high accuracy in image acquisition are used in fast-moving bottles when a motor stops at a period, image processing takes place in a series of steps, after image processing, the computer displays the result of the bottle image weather the bottle is original or defective. According to the image processing result rejection system act on the defective bottle. The inspection and rejection system are controlled by a microcontroller and motors are interfaced with the microcontroller through a driver circuit. PIR sensor is used to sense the presence of a bottle on the conveyor and send a signal to the microcontroller to control the motor's speed. The process is then repeated for every cycle.

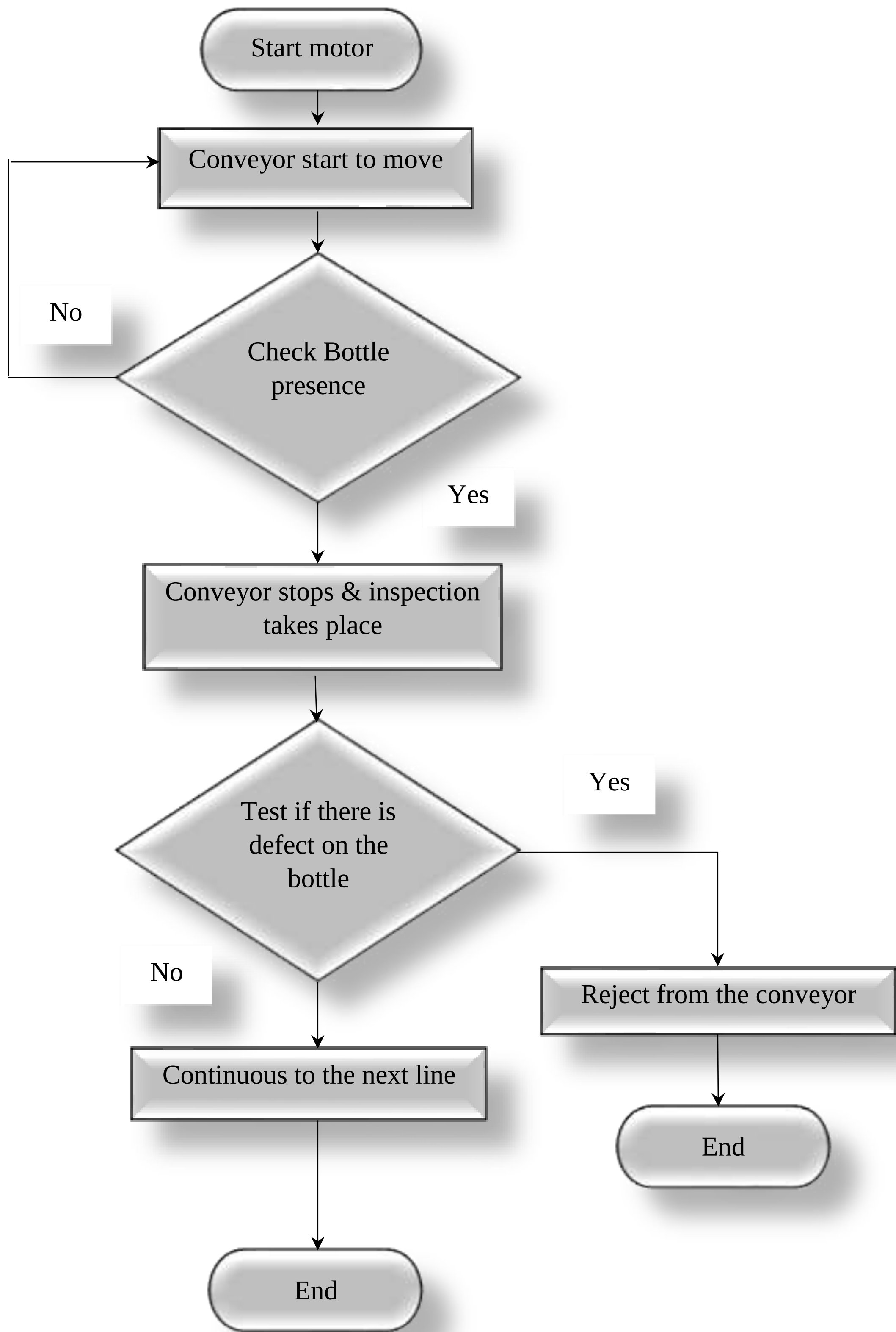


Figure 3. 7 Block diagram of the project

CHAPTER FOUR

MODELING AND ANALYSIS

4.1 Modeling

Working Principle of the whole System The system inspects different types of the defected bottle via image processing technique using MATLAB algorithm. The defected bottle of different types of defection can be effectively detected by image processing technique and rejection takes place from the rejection system. There are two units are implemented for this system.

I. Inspection unit

II. Rejection unit

These systems contain electrical, mechanical, and programming model. So, the modeling can be discus in main three modeling categories as follows.

4.1.1 Mechanical Modeling

Belt conveyor design

Conventional Design Aspects of Belt Conveyors and Mathematical Formula Analysis

Short belt conveying systems are typically designed for static conditions only considering additional loads during startup and breaking due to belt elongations only being very simplistic.

Resistance to Motion The power of the conveying system has to be designed to move and accelerate belt and load and has to overcome to following resistances to motion

- Main Resistance
- Secondary Resistance
- Slope Resistance
- Special Resistance

- The sum of these resistances to motion results in the peripheral force or effective tension that has to be transferred from the drive pulley to the belt during stationary conditions.

Belt Tensions

To calculate the maximum belt tension and hence the strength of the belt that is required, it is first necessary to calculate the effective tension. This is the force required to move the conveyor and the load it is conveying at a constant speed. Since the calculation of effective tension is based on a constant speed conveyor, the forces required to move the conveyor and material are only those to overcome frictional resistance and gravitational force.

Mass of Moving Parts

For the sake of simplicity, the conveyor is considered to be made up of interconnected unit length components all of the equal mass. The mass of each of these units is called the mass of the moving parts and is calculated by adding the total mass of the belting, the rotating mass of all the carrying and return idlers, and the rotating mass of all pulleys. This total is divided by the horizontal length of the conveyor to get the mean mass of all the components. At the outset, the belt idlers and pulleys have not been selected and hence no mass for these components can be determined.

Mass of the load per unit length (Q)

As is the case with the components the load that is conveyed is considered to be evenly distributed along the length of the conveyor. Given the peak capacity in tons per hour, the mass of the load per unit length is given by:

$$Q = 0.278\frac{T}{S}$$

The effective tension (Te) is made up of 4 components, these are:

The tension to move the empty belt (Tx)

The tension to move the load horizontally (Ty)

The tension to raise or lower the load (Tz)

The tension to overcome the resistance of accessories (Tu)

The effective tension is the sum of these four components

$$Te = Tx + Ty + Tz + Tu$$

$$Tx = 9.8G \times fx \times Lc$$

$$Tz = 9.8Q \times H$$

The tension required to overcome the resistance of skirt boards T_{us} :

$$T_{us} = \frac{9.8 * fs * Q * Ls}{S * b^2}$$

Tension to overcome the resistance of scrapers :

$$T_{uc} = (A * fc * \rho)$$

In the case of a belt, the additional tensional required to overcome the resistance of each plow is :

$$T_{up} = 1.5 * W$$

Moving trippers require additional pulleys in the system and therefore add tension. If the mass of the additional pulleys has been included in the mass of moving parts then no additional tension is added. However, if a separate calculation of the tension to overcome the resistance of the additional pulleys is required this can be determined for each additional pulley as follows:

$$T_{ut} = 0.01\ \frac{d0 * T1}{Dt}$$

Corrected length (Lc)

Conveyors require relatively more force to overcome frictional resistance than longer conveyors and therefore an adjustment is made to the length of the conveyor used in determining the effective tension. The adjusted length is always greater than the actual horizontal length.

$$\mathbf{LC = L + 70}$$

The length correction factor (C) is:

$$C = \frac{Lc}{L}$$

All conveyors require additional tension in the belt to enable the drive pulley to transmit the effective tension into the belt without slipping. This tension, termed the slack side tension T2, is induced by the take-up system. In the case of a simple horizontal conveyor the maximum belt tension T1 is the sum of the effective tension Te and the slack side tension T2:

$$T1 = Te + T2$$

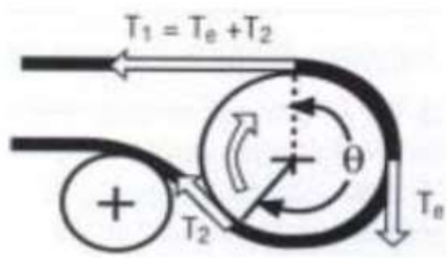


Figure 4. 1 Belt tension on a pully

T1 is the tight side tension and T2 is the slack side tension for a more complex conveyor profile that is inclined, additional tensions are induced due to the mass of the belt on the slope. This tension is termed the slope tension 'h and increases the total tension. Thus

$$T1 = Te + T2 + Th$$

The slack side tension is determined by consideration of two conditions that must be met in any conveyor. The first condition is that there must be sufficient tension on the slack side to prevent belt slip on the drive. The second condition is that there must be sufficient tension to prevent excessive sag between the carrying idlers.

Minimum tension to prevent slip Tm

At the point of slipping the relationship between T1 and T2 is

$$\frac{T1}{T2} = e^{u\theta}$$

Since: $T1=T_e + T2$

$$T2 = \frac{1}{e^{u\theta} - 1}\mathrm{Te}$$

At the point of slipping the relationship between T1 and T2 is called the driving factor k. and the value of T2 that will just prevent slip is referred to as the minimum to prevent slip Tm and therefore:

$$Tm = k \times Te$$

Minimum tension to limit belt sag Ts

The tension required to limit sag is dependent on the combined mass of belt and load, the spacing of the carry idlers, and the amount of sag that is permissible.

Ts = 9.8Sf x (B + Q) x ld

The value of the slack side tension must ensure that both conditions are met and therefore T2mmust be the larger of Tm or Ts.

Slope tension Th

The slope tension is the product of the belt weight and the vertical lift and has its maximum value at the highest point of the conveyor.

$$Th = 9.8B \times H$$

Unit tension T

The maximum belt tension T1 has as its reference width the full width of the belt. Usually, this is converted to the tension per unit of belt width as this is the reference dimension for belt strengths.

$$\boldsymbol{T = \frac{T1}{W}}$$

Absorbed power

The amount of power required by the conveyor is by definition of power equal to the product of the force applied and the speed at which the conveyor belt travels. The force applied is the effective tension and hence the power required at the shaft of the drive pulley/s is:

P = Te x S

Step-by-step procedure of belt conveyor tension calculation and design

we take the following data to design a belt conveyor for our project:
Belt width ------1430 mm

Conveyor length (L)----1 m

Capacity of mass load ---- 3600 bottle per hour (1098 Kg / hr // 1.098 T/hr)

Belt speed (S)----- 0.25 m/s

Idler spacing -------1.2m

Idler roll diameter-----------120 mm

Drive--------------------- 210-degree wrap. Lagged drive pulley

Based on the above we design the following design

1. Determine the mass of the load per unit length

$$Q = 0.278\,\frac{T}{S}$$

$$Q = 0.278 \times 1.098 / 0.25$$

$$= \mathbf{1.221\ kg/m}$$

2. We look up the value of the mass of moving parts from given data for the idler roll diameter and the nature of the material conveyed in the application is considered a medium-duty. For belt width between 0 -700 mm, we take the mass of moving parts is **G= 25 kg/m**.

3. Calculate the corrected length and the length correction factor.

LC = L + 70

$$L_c = L + 70$$

$$= 1 + 70$$

$$= \mathbf{71\ m}$$

4. Tension to move the empty belt.

$$T_x = 9.8G \times f_x \times L_c$$

$$= 9.8 \times 25 \times 0.022 \times 71$$

$$= \mathbf{382.69N}$$

5. Tension to move the load horizontally.

$$T_y = 9.8Q \times f_y \times L_c$$
$$= 9.8 \times 1.221 \times 0.027 \times 71$$
$$= 22\mathbf{.938N}$$

6. Tension to lift the load.

$$T_z = 9{,}8Q \times H$$
$$= 9.8 \times 1.221 \times 20 \times 0$$
$$= \mathbf{0\ N}$$

7. Accessories are present and therefore the tension to overcome the resistance of accessories is:

$$T_u = \frac{9.8 * fs * Q * Ls}{S * b^2}$$

8. Effective tension

$$T_e = T_x + T_y + T_z + T_u$$
$$= 382.69 + 22.938 + 0 + 0$$
$$= 405.628\ \mathbf{N}$$

9. The absorbed power

$$P = T_e \times S$$
$$= 405.628 \times 0.25$$
$$= \mathbf{101.1W}$$

10. The slack side tension.

Slack side tension to prevent slip. The driving factor for 210-degree wrap and lagged pulley with a gravity take-up, from our data, is 0.38 (k). Slack side tension to limit sag to 2%. The sag factor for 2% sag is 6.3 and the estimated belt mass for a medium load and the range of 0 -700 mm belt width, according to the given data that we got, is 11.1kg/m.

$$T_m = k \times T_e$$
$$= 0.38 \times 405.628\ N$$
$$= \mathbf{154.13864\ N}$$

$$TS = 9.8Sf\,(B + Q) \times ld$$
$$= 9.8 \times 6.3 \times (11.1 + 1.221) \times 1.2$$
$$= \mathbf{912.8382\ N}$$

11. Slope tension using the estimated belt mass found in Table 8 for medium load and 900 mm belt width is:

$$Th = 9.8B \times H$$
$$= 9.8 \times 11{,}1 \times 0$$
$$= \mathbf{0\ N}$$

12. The maximum belt tension

$$T1 = Te + T2 + Th$$
$$= 405.628N + 154.13864N + 0N$$
$$= \mathbf{559.76664\ N}$$

The maximum belt tension is converted to the unit tension. Effective tension.

$$T = T1/W$$
$$= 559.76664\ N/130\ mm$$
$$= 4.325\ N/mm$$
$$= \mathbf{4.325\ kN/m}$$

The main frame of the conveyor

In constructing the frame of the conveyor, there are a lot of mechanical elements with different lengths like Bearings, Belts, Aluminum profile extrusion frames, motor brackets, different types of screws and nuts, teeth pulleys, and linear bearing shafts.

Design Calculations for pneumatic pusher

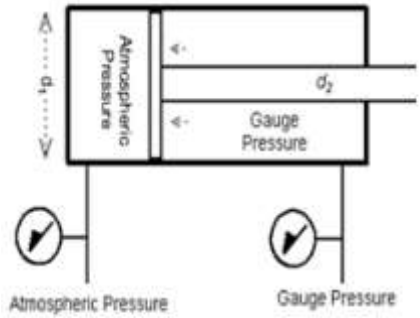


Figure 4. 2 Schematic diagram of pneumatic pusher

SPECIFICATION

- Bore Dia:15 mm
- Stroke Length:100 mm
- Piston Rod Diameter :10 mm
- Operating Pressure:0.5-10 Bar

CALCULATION

$$\text{Force output} = \pi\,(D^2/4 - d^2/4) * P$$

D-Cylinder bore (mm)

d-Piston rod diameter (mm)

P-Pressure (bar)

Operating pressure = 1 Bar

$$\text{Force output} = \pi * (15^2/4 - 10^2/4) * 0.1 \text{ Newton}$$

$$= 9.8 \text{ N}$$

4.1.2 Electrical Modeling

Design of three-phase induction motor

The conveyor motor will be controlled by a variable frequency drive (VFD).

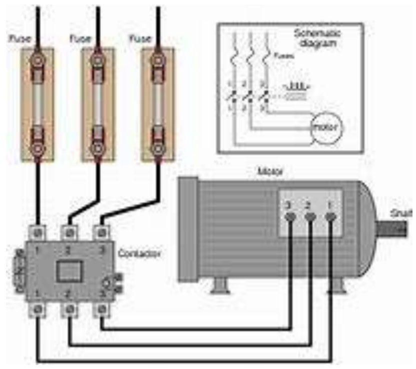


Figure 4. 3 Schematic diagrams of three-phase induction motor

Stator design Output equation of induction motor:

$$Q = C_o D^2 L * n_s$$

Where, D = diameter of stator (in meter)

L = length of stator (in meter)

Co = output coefficient,

$C_O = 11\ B_{av} a_c = \text{kw}\ 10^{-3}$

Also,

$$Q = \frac{H_p * 0.746}{\eta \cos \emptyset}$$

Where B_{av}= specific magnetic loading.

a_c = specific electric loading. Kw

kw= winding factor. Output in KW

Q = input power * e_{ff}

Input power to motor =3*Vph*Iph*cos*10^{-3}

The output power is required to be

$$Q = 1.1\ KW\ or\ 1.5\ hp$$
$$V_{ph} = Emf_{ph} = 4.44 * f * \emptyset * T_{ph} * KW$$
$$= 2.22f * \emptyset * Z_{ph} * KW$$

Frequency is given as 50 Hz

KW is given as 0.955

$$\emptyset = B_{av} * \tau * L$$

$$\tau = \frac{\pi * D}{p}$$

Let = 0.6 considering the advantage of selecting a higher value.

The specific electric loading ranges

$$10000\ {ac}/_{m} - 450000\ {ac}/_{m}$$

Designing to obtain a good power factor we choose

$$\frac{L}{\tau} = 1.3$$

From the relation of

$$\frac{pole\ pitch}{core\ length} = \frac{0.18}{pole\ pitch}$$

$$\left(\frac{\pi D}{p}\right)\Big/_{L} = \frac{0.18}{\pi D/_{p}}$$

$$D = 0.135 * p\sqrt{L}$$

Calculating the synchronous speed in rps

$$f = \frac{p * n_s}{2}$$

$$n_s = \frac{2f}{P} = \frac{2 * 50\ Hz}{4} = 25\ Hz$$

Calculating the output coefficient

$$C_o = 11B_{av} * z * KW * 10^{-3}$$

$$= 11 * 0.6 * 20000\, ac/m * 0.955 * 10^{-3}$$

$$= 126.06$$

From the output equation

$$Q = C_o * D^2 * L * n_s$$

$$1.1 = 126.06\, D^2 L * \frac{2 * 50}{P}$$

$$\frac{D^2 * L}{P} = 8.73 * 10^{-5}$$

Putting the first equation into this

$$\frac{0.02P^2L^2}{P} = 8.73 * 10^{-5}$$

$$\sqrt{L^2} = \sqrt{0.0019}$$

$$L = 0.033m$$

$$D = 0.135 * 4 * \sqrt{0.033}$$

$$= 0.098\, m$$

From this, calculate D^2 L.

From the ratio⁄, L is calculated. Where is the pole pitch

$L/_{\tau}$= 1, for good overall design.

The peripheral speed must be below 30 m/s

$$v = \pi * D * n_s$$
$$= \pi * 0.98m * 25rps$$
$$= 7.7\, m/s$$
$$which\ is\ less\ than\ 30m/s$$

Length of air gap, δ = 0.2 + √(DL) in mm.

Calculating the length of the air gap

$$\delta = 0.2 + \sqrt{DL}$$

$$= 0.2 + \sqrt{0.098 * 0.033m^2}$$

$$= 0.2 + 56.8mm$$

$$= 57mm$$

We considered choosing a partially closed slot due to some advantages,

Better mechanical strength.

Good power factor.

Low loss.

Number of stator slots

$$Emf_{ph} = 4.44 * f * \emptyset_m * T_s * k_{ws} * 10^{-3}$$

T_s= no of turns of the stator winding.

$$k_{ws} = \text{Stator winding factor}(k_{ws} \approx 0.955)$$

$$\emptyset_m = B_{av}\tau L$$

$$\tau = \frac{\pi D}{p}$$

We can calculate the no of turns (Ts), First, we need to find other parameters ...

Calculating several stator slots. Let k=1

$$S_s = \left(\frac{m}{k}\right) * p$$

$$\frac{3}{1} * 4 = 12$$

m = phase number

p = number of poles

he selected many slots that should be satisfying the consideration of stator slot pitch at the air gap surface, which should be between 1.5 to 2.5 cm.

stator slot pitch at the air gap surface = λ_s

$$\lambda_s = \pi * \frac{D}{S_s}$$

D = 0.098m = 9.8m

$$\lambda_s = \frac{\pi * 9.8cm}{12} = 2.5\ cm$$

the number slot is satisfying since the slot pitch at the air gap is 2.5cm

Calculating turns per phase

$$E_{ph} = 4.4f * \emptyset * T_{ph} * KW$$

$$T_{ph} = \frac{E_{ph}}{4.44 * f * \emptyset * KW}$$

Induced emf can be assumed to be equal to the applied voltage per phase flux per pole.

$$\emptyset = B_{av} * {}^{\pi * D * L}/_{p}$$

$$= 0.6 * \pi * \frac{0.098 * 0.033}{4}$$

$$= 1.5 * 10^{-3}\ \tau$$

Solving for the input power

$$\frac{m * I_{ph} * Z_{ph}}{\pi * D} = q$$

$$I_{ph} * Z_{ph} = \frac{q * \pi * D}{m} \qquad \dots eq(1)$$

Input power = $m * v_{ph} * I_{ph} * cos\emptyset * 10^{-3}$ KW

$$V_{ph} = 2.22 * f * \emptyset * Z_{ph} * KW \qquad \dots eq(2)$$

$$input\ power = m * 2.22 * f * \emptyset * Z_{ph} * KW * I_{ph} * cos\emptyset * 10^{-3} \qquad \dots eq(3)$$

Putting equation (1) into equation (3)

$$Input\ power = 2.22 * f * \emptyset * KW * q * \pi * D * cos\emptyset * 10^{-3}$$

$$= 2.22 * 50 * 1.5 * 10^{-3} * 0.955 * 1.3 * 20000 * \pi * 0.098 * 10^{-3}$$

$$Inpit\ power = 1.27KW$$

Stator conductor cross-section,

We have stator current per phase.

$$I_s = \frac{Q * 10^{-3}}{E_s} \qquad (Amp)$$

$$Q = E_s I_s 10^{-3} \qquad E_s = voltage$$

$$a_s = \frac{I_s}{\delta_s} \quad (mm^2)$$

$$\delta_s = 3\ to\ 5\ ({}^{Amp}/_{mm^2})$$

Where a_s= area of stator conductor

δ_s= current density.

Assuming the cross-section area of the conductor = 1 mm^2

From the relation of an area with the current per phase

$$a_s = {}^{I_s}/_{\delta_s}$$

The current density δ_s is usually given for the stator winding is 3 to 5 $amps/mm^2$

We have chosen the δ_s to be 4 $amps/mm^2$ considering the advantage and the disadvantage.

$$I_s = a_s * \delta_s$$

$$= 1mm^2 * 4\,amps/mm^2$$

$$I_s = 4amps$$

Then,

$$Input\ power = m * V_{ph} * I_{ph} * \cos\emptyset * 10^{-3}$$

$$V_{ph} = \frac{Input\ power}{m * V_{ph} * I_{ph} * \cos\emptyset * 10^{-3}}$$

$$= \frac{1.27 * 10^{-3}Kw}{3 * 4amp * 1.3}$$

$$V_{ph} = 81.4\ volt$$

$$V_{ph} = E_{ph}$$

Hence calculating turns per phase

$$T_{ph} = \frac{E_{ph}}{4.44 * f * \emptyset * KW}$$

$$= \frac{81.4}{4.44 * 50 * 1.5 * 10^{-3} * 0.955}$$

$$T_{ph} = 256\ turns$$

Number of conductors per phase

$$Z_{ph} = 2 * T_{ph}$$

$$= 2 * 256 = 512$$

Total number of stator conductors

$$z = 2 * m * T_{ph}$$

$$= 2 * 3 * 256$$

$$= 1536$$

Total no of conductors per slot,

$$Z_{ss} = \frac{Z_s}{S_s} = \frac{6T_s}{S_s}$$

$$Z_s = \frac{Z}{S_s} = \frac{6 * T_{ph}}{S_s}$$

$$= \frac{1536}{12} = 128$$

Area of stator slot

$$\text{Ass} = \frac{Area\ of\ copper\ per\ slot}{Space\ factor} = \frac{Zs * as}{Sf}$$

$$Sf = \frac{Total\ conductor\ area}{Actual\ slot\ area} = 0.25\ to\ 0.04$$

The space factor is assumed to be between 0.25 and 0.4

$$Area\ of\ eah\ slot = \frac{128 * 1mm^2}{0.4}$$

$$= 320mm^2$$

Length of mean turn

$$Lmts = 2L + 2.3\tau + 0.24$$

$$= 2 * (0.033) + 2.3 * (0.025) + 0.24$$

$$= 0.36m$$

Resistance of stator winding

$$R_s = \frac{\rho_c * L_{mt} * T_{ph}}{a_s}$$

Resistivity of copper ρ_c= 0.021 Ω m

$$R_s = \frac{0.021\Omega m * 0.36m * 256}{1mm^2}$$

$$= 1.94\Omega$$

Total copper losses in the stator winding

$$= m(I_s)^2 R_s$$

$$= 3 * (4amp)^2 * 1.94\Omega$$

$$= 23.28watt$$

Stator teeth design

$$minimum\ teeth\ area\ per\ pole = \frac{\emptyset_m}{1.7}$$

$$\emptyset_m = maximum\ flux\ per\ pol$$

Hence

$$\frac{\emptyset_m}{1.7} = \frac{S_s}{p} * L_i * W_{ts}$$

Here Li = length of the stator core

Then calculate W_{ts}

Stator teeth design

Let's assume the length of the stator core to be = 9cm

$$Wts = \frac{\emptyset_m * p}{1.7 * S_s * L_i} = \frac{1.5 * 10^{-3} * 4}{1.7 * 12 * 0.09}\ \frac{Wb}{m} = 3.3 * 10^{-3}\frac{Wb}{m}$$

Stator core design

Flux per pole is = $\emptyset_m$

Flux through stator core = $\frac{\emptyset_m}{2}$

$$Area\ of\ stator\ core = \frac{flux\ through\ core}{flux\ density\ in\ core} = \frac{\frac{\emptyset_m}{2}}{B_{cs}} = \frac{\emptyset_m}{2B_{cs}}$$

B_{cs} is flux density in stator core

$$1.2\ to\ 1.4\ {}^{Wb}/_{m^2}$$

$$Area\ of\ stator\ core = L_s * d_{cs}$$

Here d_{cs} = depth of stator core

$$d_{cs} = \frac{\emptyset_m}{2B_{cs}L_i}$$

$$L_i = length\ of\ stator\ core$$

Let B_{cs}(flux density in stator core) $= 1.4\frac{Wb}{m^2}$

$$d_{cs} = \frac{\emptyset_m}{2 * B_{cs} * L_i} = \frac{1.5 * 10^{-3}}{2 * 1.4 * 0.09}$$

$$d_{cs} = 6 * 10^{-3}m$$

A of stator core $= L_i * d_{cs}$

$$= 0.09m * 6 * 10^{-3}$$

$$= 54 * 10^{-5}mm^2$$

The outside diameter of the stator core is

$$D_o = D + 2d_{ss} + 2d_{cs}$$

depth stator core $= d_{cs}$

$$D_o = D + 2 * h_{ss} = 2d_{cs}$$

$$D_o = 2 * 6 * 10^{-3} = 12 * 10^{-3}m$$

Design of Rotor

A suitable combination of stator and rotor slots

$$S_s - S_r \neq 0, \pm P, \pm 2P, \pm 3P, 5P, \pm 1, \pm 2, (P \pm 1), (P \pm 2)$$

$$P\ is\ number\ of\ poles$$

- The rotor conductors are arranged such that there should be some angle between stator conductors to avoid cogging.
- To avoid crawling the rotor, slots do not exceed the number of stator slots by 15 to 30%. Factors affecting the length of the air gap
- Power factor: the larger the air gap, the MMF required to pass current through the air gap should be high

$$MMF = AT_g = flux(\emptyset) * Reluctance(\rho)$$

There fore

$$AT_g \propto L_g$$

If $AT_g \uparrow$ then $L_g \uparrow$ also magnetic current $\uparrow$, hence $\cos \emptyset \uparrow$

$$AT_g = amphere\ turns$$

$$L_g = length\ of\ air\ gap$$

$$Cos\ \emptyset = power\ factor$$

- Over load capacity: is the ratio of maximum power to the rated power. It is greater when the leakage reactance is less.
- Pulsation loss

$$\uparrow L_g \propto \frac{1}{pulsation\ loss} \downarrow$$

- Cooling If $L_g \uparrow$ then the cooling is better.
- Noise If $L_g \uparrow$ then the noise is less.
- Length of the air gap

$$L_g = 0.2 + 2\sqrt{D * L}\ (mm)$$

- The length of the air gap should be higher to avoid unbalanced magnetic pull, which is the radical force acting on the rotor due to the non-uniform air gap around the armature periphery.

4.1.3 Programming Development

An algorithm is a step-wise representation of a solution to a given problem. In Algorithm the problem is broken down into smaller pieces or steps.

Inspection Unit

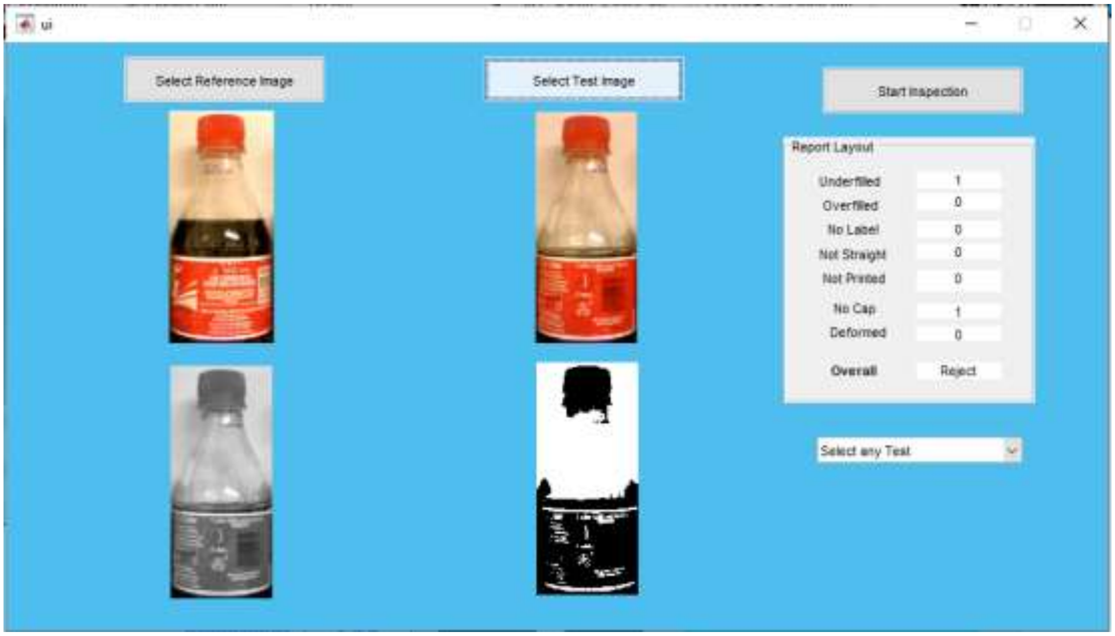
In inspection unit, it consists of a microcontroller, IR module, Web cam, a driver circuit, and conveyor. When bottles come under the sensing range IR module senses the presence of bottles on the conveyor and sends the signal to the microcontroller. The microcontroller actuates and sends a signal to the motor via the driver circuit and controls the motor speed finally conveyor will stop.

When the conveyor stops the web, the cam captures the image of the bottle in the required size and sends it to the computer for processing. Image processing is carried out in a computer and the computer displays the result of bottles. There are eight different cases are inspected in the inspection unit

1. Standard bottle
2. Bottle cap missing defect
3. Overfilled bottle defect
4. Under filled bottle defect
5. Straight label defect
6. Missing Label defect
7. Deformed Bottle defect
8. Label Not Printed defect

Visual measurement

In this section, different algorithms for defect detection are used on a streaming video, with a resolution of 640x480 and a series of frames or pictures taken within microseconds, we can construct a series of algorithms to detect and track the presence of a bottle in the region, extraction and analysis are made to decide the properties of the bottle in the Region of interest (ROI), the figure below shows the prototype of the developed system's graphical user interface which is figured out results of the photos taken by the camera during the inspection.

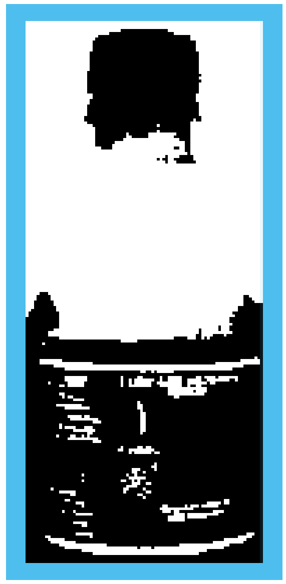


For performance evaluation, a combination of foreground detection and blob analysis will be analyzed. These methods are designed to overcome specific problems related to the different methodologies. The systems are tested to detect the bottles passing by.

Then, experiments are performed on these three data sets for evaluating the parameter sensitivity and performances of the ROI localization and defect detection. Experiments are conducted on a computer equipped with an Intel(R) Core (TM) i5-2848M (2.3GHz) and 4 GB of memory, it's highly recommended to use other updated alternatives to achieve flexibility and get accurate results. The program is implemented in MATLAB.

Image filtering and segmentation

After image is captured in milli seconds or lesser, an image frame is clustered into three-layer RGBs, then filtering takes place using ***imopen*** function to perform binary image with structuring element ***strel*** or SE, this element must be a single structuring element for example in this occasion we used '*diamond*' which creates flat, diamond structured element to oppose as an array of objects. The morphological open operation is an erosion followed by a dilation, using the same structuring element for both operations. The figure below shows after this simultaneous operation to convert the snapped image to binary. Which shows a lot of noises in the neck of the bottle being formed in the snapping process which later to is removed.



Blob analysis

Blob analysis is a procedure for extracting pixels of connectivity or pixels in a binary image region. The pixels are divided into black and white by calculating the object's position and size. Extraction also processes the position of the object moving. The Blob Analysis object computes statistics for connected regions in a binary image. Using the binarized image in the above step we have used blob analysis to compute and return statistics of the binary image above to extract the properties by choosing some property parameters specified in the code.

Rejection unit

A pneumatic system that is interfaced with PLC is used for the rejection system. If the defected bottles come on the conveyor the microcontroller sends a signal to PLC, the PLC actuates the rejection system and hit the defected bottle, and kicks it out from the conveyor.

The main component of the PLC is shown in the figure below. The whole PLC system is built up with two proximity switches to count bottles passing by, one inspector to inspect each bottle and compare it with a preloaded image, and one pneumatic to reject away the defected bottle.

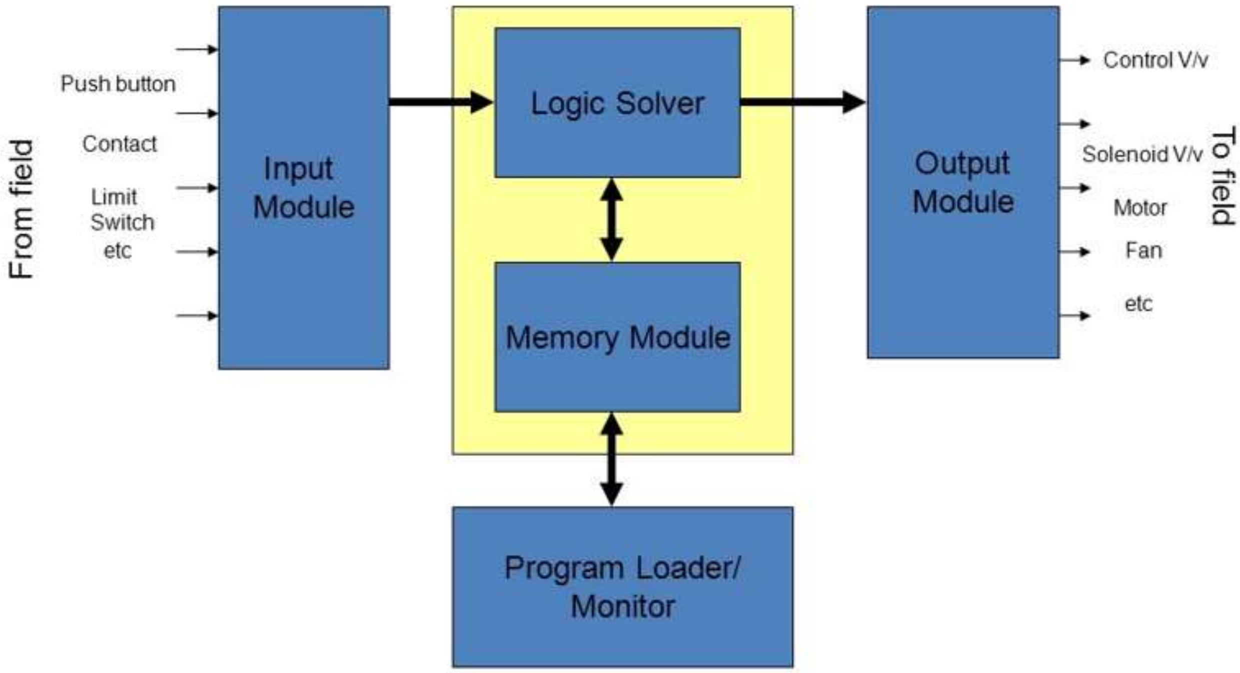


Figure 4. 4 Components of PLC

The bottles are kept in conveyor belts. There is a proximity switch. One proximity is calibrated such that it detects all the bottles passing on the conveyor. The scanner is used such that it detects defective bottles. Inspector inspects the bottle and compares it with the default image of the bottle. If there is an image mismatch the Bit Shift Register bit will be set to 1 so after some steps the rejecter will be actuated. Then rejecter will be switched ON and throw the bottle on the second conveyer. If there is no mismatch, then the rejecter will not switch ON.

Ladder Programming

There are five different approaches to operating the PLC for any system properly, i.e.,

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

Though, there are many languages for operating PLC, the program for this project has been written using ladder programming because it is quite simple to learn and less complex than other languages. This programming language is specifically designed for processing Boolean signals (1 ≡ TRUE or 0 ≡ FALSE).

The program is to be written in PLC SIEMENS (software). Basic symbols for ladder programming that are used in this program are given:

Designing Program and Simulation Part Program Part

This project comprises both hardware and software parts. A program is developed using PLC SIEMENS software (latest generation programming software). This program has developed using the above basic symbols such as normally open (NO), normally closed (NC), and output. While building up the required ladder logic we have used different addressing modes of Siemens PLC as mentioned below.

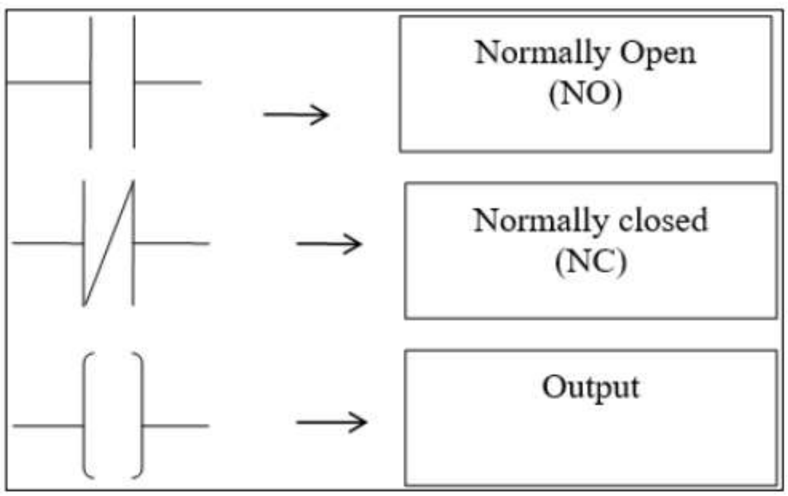


Figure 4. 5 Basic symbols of ladder programming

List of Inputs and Outputs

I10	= Start	(Input)
I1	= Stop	(Input)
I2	= Bottle Proximity	(Input)
I3	= Defected bottle detector	(Input)
Q1	= Master coil / Run	(Output)
Q2	= Conveyor motor	(Output)

Q3	= Rejecter	(Output)
Q4	= Green light	(Output)
Q4	= Red light	(Output)
SF001	= Bit shift left instruction	(Logical)
S1.4	= Bit to energize rejecter	(Bit)
T002	= control on delay of rejecter	(Timer)
T003	= stop the conveyor for scanning	(Timer)

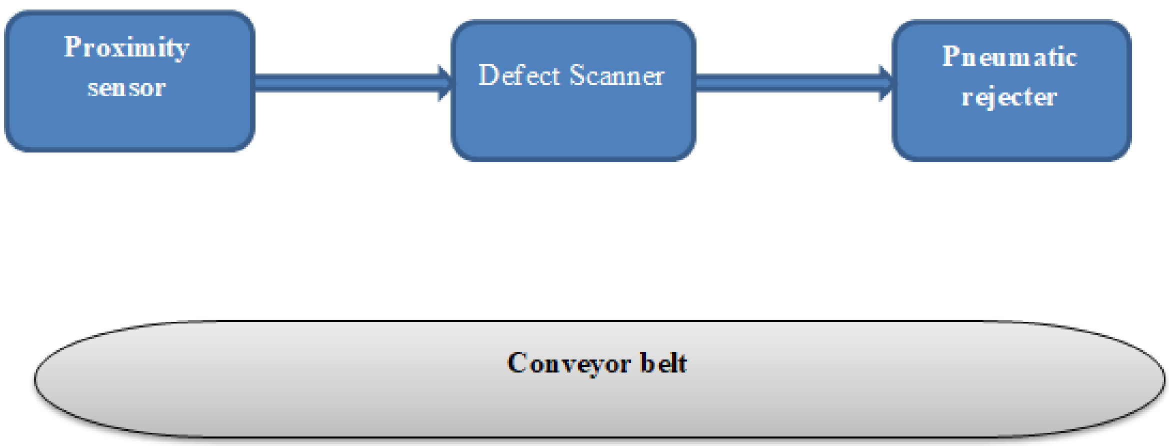


Figure 4. 6 flow chart of the PLC system

Program Description

Step 1: - After pressing the start push button [I10] the master coil is energized.

Step 2: - after the master coil is energized bottles will be moving on the conveyor belt.

Step 3: - when bottles are passing through the conveyor were detected by the Proximity switch [I2].

Step 4: - The Scanner will be energized and detect the bottle whether faulty or normal. In this step, the conveyor will be stopped for 2 sec to take picture of the bottle.

Step 5: - When the scanner detects a faulty bottle the shift register will become triggered and set an address for that specific bottle and also red light will be on Q5.

Step 6: -when faulty bottles are detected by the scanner it gives back input to our PLC with input I3, it sets bit and is shifted left every time a bottle is detected by bottle proximity with address I2.

Step 7: - From the inspector to a blower, the distance is 4 steps. Hence bit S1.4 of the S1 register is used to operate Rejecter.

Step 8: - When the B1.4 bit is set that is when the empty bottle is detected by input I3, after 4 steps, the rejecter will be activated for 1 sec activated and the faulty bottle is removed.

otherwise, if there is no defective bottle the process will continue normally so the red light will be on.

Step 9: - When we press the stop push button [I1] the total process will be rested.

4.2 Material Selection

Engineering materials are materials whose properties are technologically useful. Selecting the right materials for a specific design is probably the single most important decision facing designers. Materials are chosen for a variety of reasons such as cost-effectiveness, availability, workability, and affordability. We divide our design into two parts electrical design and mechanical design parts as we designed both the mechanical (framework) of the machine and electrical (operating) parts. We include both differentiated parts components, their definition, and specifications.

4.2.1 Hardware Components of Inspection unit

I) Mechanical Components

A) **Components of Belt Conveyor**

Belt conveyors are typically stationary conveying systems and belong to the group of continuous material handling systems. The main application is the transport of bulk material over long distances. Belt conveyors are highly reliable, productive, and safe systems covering a broad range of applications which makes these machines indispensable for decades and are crucial elements in industrial facilities of all kinds. Especially in the mining industry, in surface and underground mining as well as material handling, belt conveyors are used to moving massive volumes of minerals and overburdened material.

A belt conveyor consists of two or more pulleys with a continuous belt rotating about them. One or more pulleys are powered by an electric motor moving the belt and the material on it, called a drive pulley. The belt is supported and narrowed down to form a concave shape by idlers, which are distributed along the conveying route. Typically, there are more idlers required on the carrying side as the load of the belt and the load of the material has to be supported, whereas, on the return side, only the load of the belt is present. At the loading chute, sometimes damped idlers are used to take the load of the material dropping on the belt. The belt has to be kept under a certain tension to ensure the proper functioning of the conveying system. There are more possibilities to keep the belt under tension. As the belt can be tensioned by applying force via a pulley. Scrappers are used to keeping the belt clean, and to avoid wear on the belt, idlers, and pulleys.

1. Belt

The conveyor belt is the most important part of a belt conveyor system. The belt has not only got to transport the load but also to absorb stresses at start-up and breaking, absorb impact energy at the loading point, withstand temperature and chemical effects and meet safety standards to ensure the proper functioning of the conveyor system. A belt typically consists of two parts: the carcass and the cover.

The carcass makes up the inside of the belt and can be made from various materials to meet the above-mentioned requirements. For moderate strength belts, the carcass can be made from textile piles whereas higher strength belt car cases are usually made from steel waves or steel cords.

The cover, which is usually made from different quantities of rubber or PVC, consists of a pulley side cover and a load side cover protecting the car case and acts as a shock absorber dampening impact loads.

2. Idler

Idlers are used on the carrying and the return side, to support the load of the belt and the material transported, to guide the belt, form a trough, and absorb impact loads. Typically, the carrying side requires a closer distance and therefore more idlers as the weight of the belt and the material on it has to be supported. On the return side only, the weight of the belt has to be supported. Increasing idler spacing reduces the load that can be supported and increases the belt slag between two idlers.

3. Pulley

Depending on the drive system and the tensioning system several pulleys are necessary to drive and guide the belt. The pulley diameter depends on belt construction, the duty, and the method of splicing.

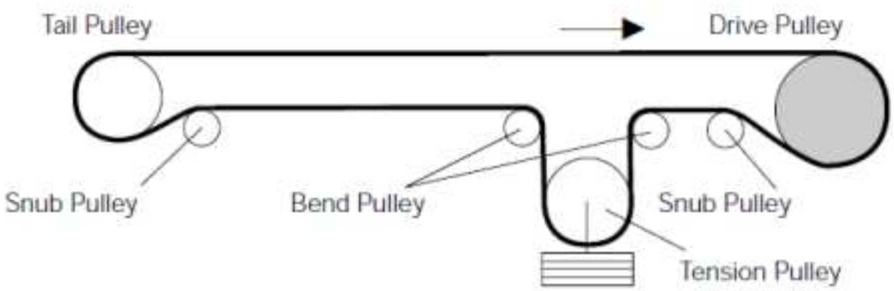


Figure 4. 7 Pulley

4. Tensioning System

It is important to keep the belt under a certain tension to ensure the proper functioning of the belt conveyor system. Tensioning systems may be fixed, use weight attached to a guided pulley, or actively regulate the tension in the belt.

5. Belt Cleaner

The belt of a belt conveying system has to be cleaned continuously. During operation, depending on the characteristics of the material transported, residual load material sticks to the carrying side of the belt.

6. Loading Station

At the loading station, the material which is about to be transported by the conveying a system is transferred onto the conveyor belt. This is usually done through a loading chute. The aim is to take as much energy from the material before it drops onto the belt through impact deflection plates or grizzly bears.

II) Electrical Components

PLC (programmable logic controller)

A Programmable Logic Controller, PLC is a digital computer used for automation is a digital electronic device that uses a programmable memory to store instructions and to implement specific functions such as logic, sequence, timing, counting, and arithmetic operations to control machines and processes. In other words, PLC is a solid-state/computerized industrial controller that performs discrete or sequential logic in a factory environment. A sequence of instructions can be programmed by the user into the PLC and it can demonstrate the use of these operations in a situation. It is an interface between the program and the inputs. It is programmable software.

A PLC is an example of a real-time application and is therefore used to control various devices. The PLC works depending on the inputs given and their state, turning on/off its outputs. The user enters a program, usually through software that gives the results. PLC is used in many "real world" applications. For all application that needs some type of electrical signal, PLC works based on inputs given by the user.

Selection criteria for plc

While selecting a PLC, various factors should be considered as per the application. The important factors that affect this criterion are:

- a number of Inputs and Outputs.
- Scan Time
- Communication Protocol
- Memory Size
- Software

Different PLCs have different configurations of input and output modules. Normally, compact type PLC has fixed inputs and outputs. Expansion modules are available to increase the I/O but they are compactible only with the high-end models. Also, if we want to save the programs on a card then flash clips/memory modules are available in the market in which we can save the program. The next important thing is the communication protocol and software.

In this paper, the PLC used is a LOGO! Soft Comfort V8: Only one station with its 10-logic input and output ports is used. According to the requirement, a greater number of stations can be added thus increasing the potential of this system to be used in very large industries.

Figure 4. 8 Programmable logic controller

Motor

Figure 4. 9 AC motor

Proximity Switch

A proximity sensor often emits an electromagnetic field or a beam of electromagnetic radiation (infrared, for instance), and looks for changes in the field or return signal. The object being sensed is often referred to as the proximity sensor's target. Different proximity sensor targets demand different sensors. A proximity switch uses this proximity sensor to trigger ON or OFF certain circuits required for the processing of the system. We use an IR sensor for the detection of the coming bottle to the camera.

Figure 4. 10 PIR Sensor

Industrial Camera

Figure 4. 11 Vision camera

Industrial Pc

The main difference which separates an industrial computer (also called IPC) from a normal PC is its design and construction. IPCs are rugged and are more useful in an industrial environment, which protects them from dust, water, debris, fire, shock, or vibration. Inch Android Windows Touch Screen Industrial Panel PC is used for running the software's MATLAB and PLC for the system.

Figure 4. 12 Industrial level PC

Arduino Uno

Arduino UNO is a low-cost, flexible, and easy-to-use programmable open-source microcontroller board that can be integrated into a variety of electronic projects. This board can be interfaced with other Arduino boards, Arduino shields, and Raspberry Pi boards and can control relays, LEDs, servos, and motors as an output.

Arduino UNO is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button.

Figure 4. 13 Arduino

4.2.2 Hardware component of rejection unit

Pneumatic Cylinder

Pneumatic cylinders also called air cylinders are the final component of the pneumatic or compressed air control mechanical device. Air or pneumatic cylinders are devices that transform compressed air power into mechanical energy.

The mechanical energy produces linear or rotary motion. The pneumatic air cylinder functions as the actuator in the pneumatic system. So, it is called a pneumatic linear actuator. Selecting the right pneumatic cylinder can ensure the long-term success of an application and improves the proper overall performance of the machine.

Figure 4. 14 Pneumatic pusher

Principle

The pneumatic cylinder converts the pressure energy of a compressed air medium into mechanical energy in the form of linear or rotary motion.

Single Acting Cylinder

Only one side of the piston is supplied with certain working pressure. Force acts in one direction to control the movement, and returns to a normal state by an external force such as a spring inside.

Types of Single Acting Cylinders

Based on the operation the Single-acting cylinders are classified as

1. Push type and
2. Pull type.

Push type Single-acting Cylinder

Compressed air enters to push the piston out of the cylinder. The spring automatically retracts the piston to its home position when the pressure is removed.

Pull type Single-acting Cylinder

Compressed air enters to pull the piston inside of the cylinder. When the compressed air passed through the port, the piston in the cylinder starts retracting. The pressure port is located at the cylinder end.

In this project we used pusher type pneumatic single acting cylinder to push the defected bottle out of the conveyor.it is actuated by PLC.

4.2.3 System software description

MATLAB

Matrix Laboratory MATLAB is a Multi-programming language, Multi-operating system compatible platform started in the late 1970s, MATLAB is a numerical computing platform that allows high order matrix operation, functional plotting of data and implementing algorithms, and creation of graphical user interfaces. We use MATLAB for system software development. The image acquisition toolbox is used to capture the images from the webcam vision camera interfaced with the PC. The image processing toolbox is used to develop the algorithm to inspect the bottle defect.

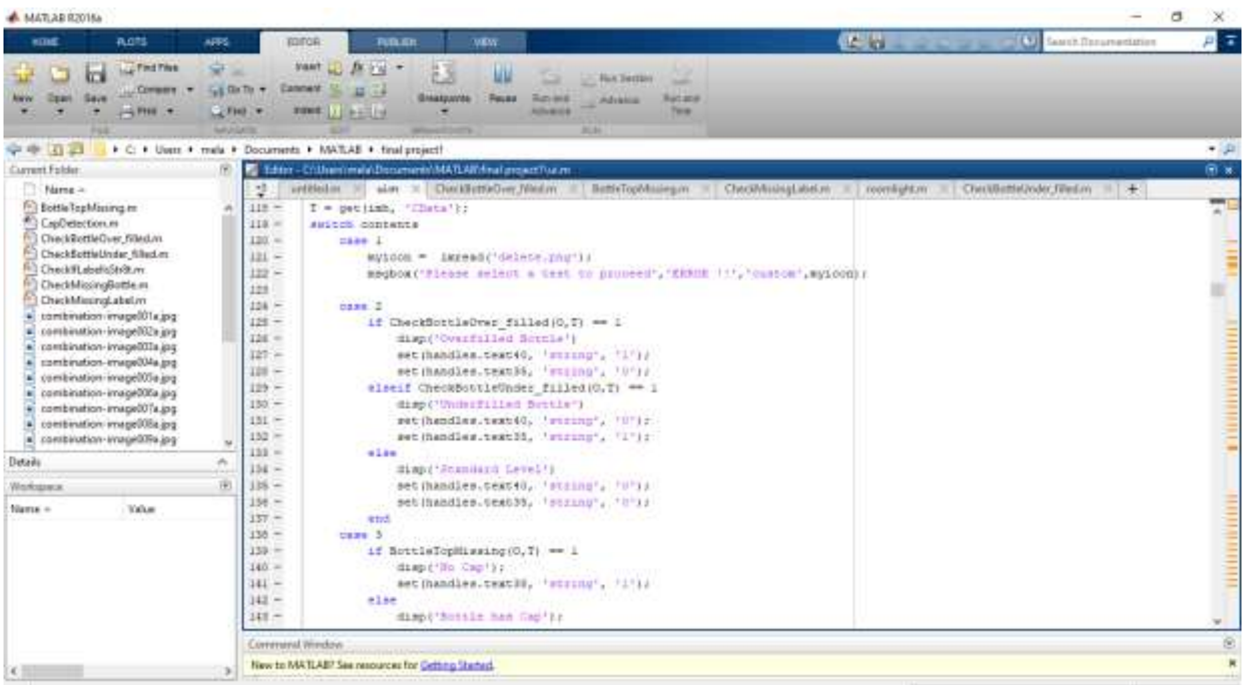


Figure 4. 15 MATLAB

PLC SIEMENS software

We STEP 7 (TIA Portal) to code and run PLC. SIMATIC STEP 7 offers maximum user-friendliness and helps to perform your engineering tasks intuitively and efficiently.

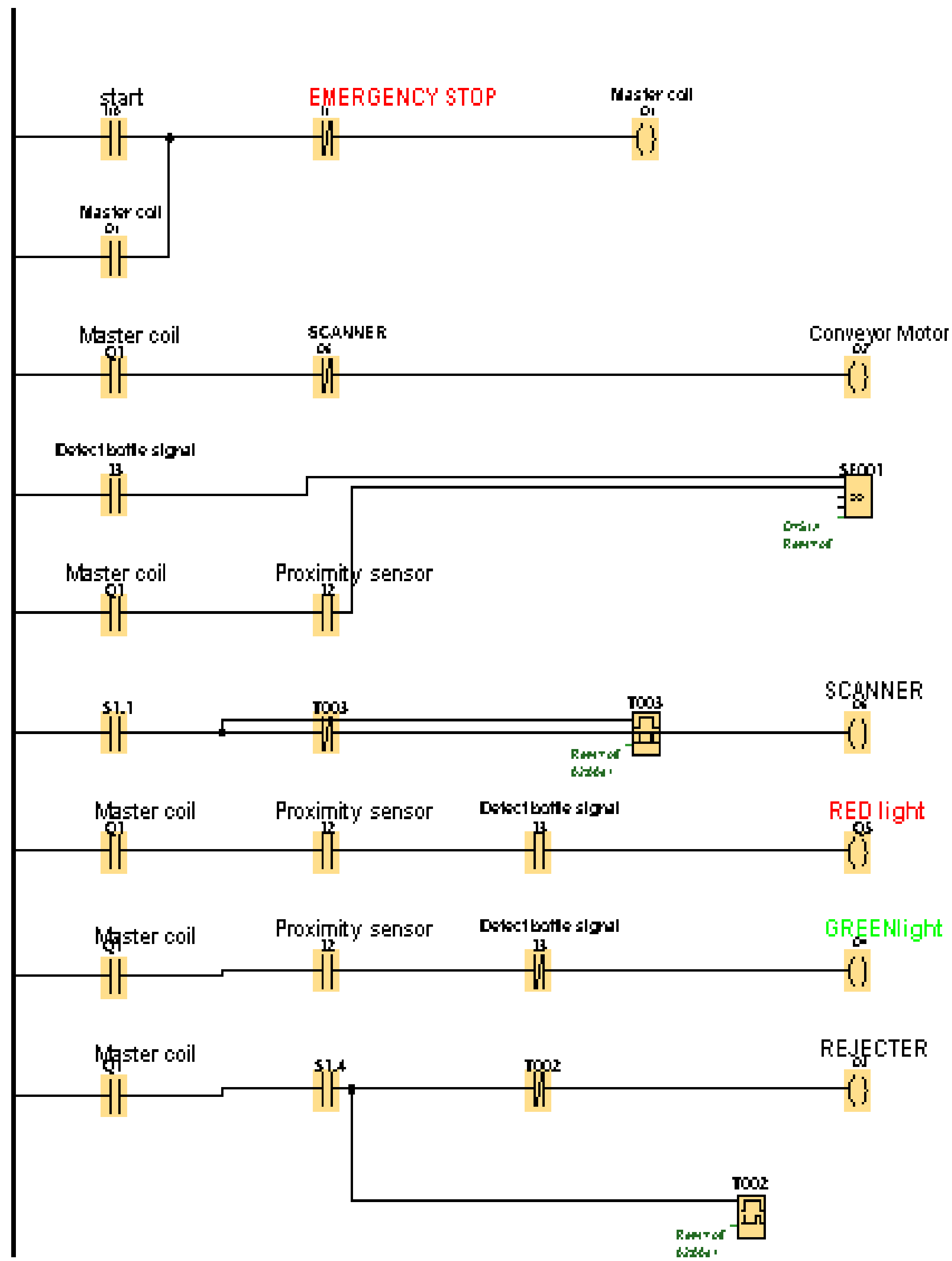


Figure 4. 16 PLC ladder programming

Solid Works

Dassault Systems SOLIDWORKS Corp. develops and markets 3D CAD design software, analysis software, and product data management software. In this project, it is used to design the belt conveyor, motor and pneumatic cylinder.

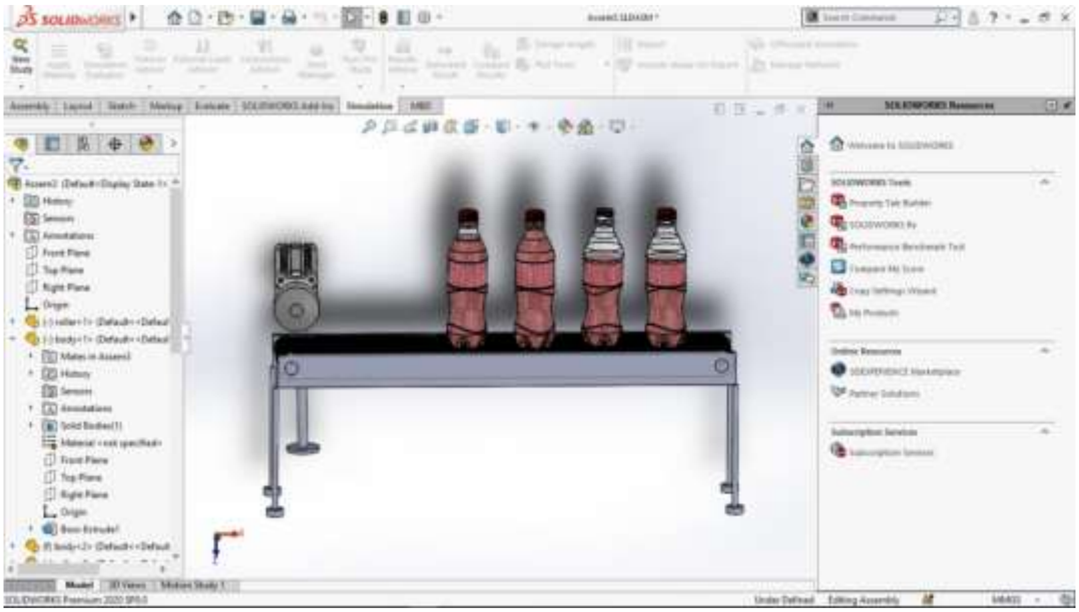


Figure 4. 17 SolidWorks

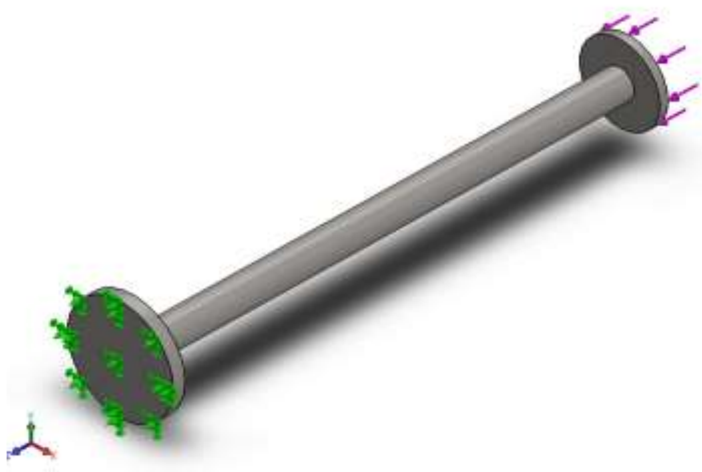
Factory IO

Factory I/O is a 3D factory simulation for learning automation technologies. Designed to be easy to use, it allows to quickly build a virtual factory using a selection of common industrial parts. We used this software to see the 3D simulation of the system and to see the validity of the designed system.

Figure 4. 18 Factory IO

4.3 Stress Analysis

4.3.1 Stress Analysis for leg



Model name: leg 2
Current Configuration: Default

Solid Bodies		
Document Name and Reference	Treated As	Volumetric Properties
Revolve1 	Solid Body	Mass:0.394146 kg Volume:5.11877e-05 m^3 Density:7,700 kg/m^3 Weight:3.86263 N

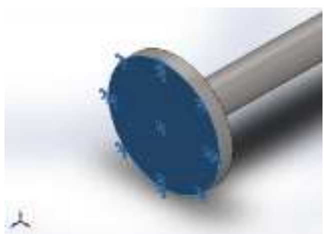
Table 4. 1 Model Information for leg

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m^2

Table 4. 2 Units of leg (support)

Model Reference	Properties	Components
	Name: **Cast Stainless Steel** Model type: **Linear Elastic Isotropic** Default failure criterion: **Unknown** Elastic modulus: **1.9e+11 N/m^2** Poisson's ratio: **0.26** Mass density: **7,700 kg/m^3** Shear modulus: **7.9e+10 N/m^2** Thermal expansion coefficient: **1.5e-05 /Kelvin**	**SolidBody 1(Revolve1)(leg 2)**
Curve Data:N/A		

Table 4. 3 Material properties for leg

Fixture name	Fixture Image	Fixture Details
Fixed-1		**Entities:** **1 face(s)** **Type:** **Fixed Geometry**

Resultant Forces

Components	X	Y	Z	Resultant
Reaction force(N)	3.33123e-05	3.80635e-05	-10.0003	10.0003
Reaction Moment(N.m)	0	0	0	0

Table 4. 4 Fixture for leg

Load name	Load Image	Load Details
Force-1		**Entities:** **1 face(s)** **Type:** **Apply normal force** **Value:** **10 N**

Table 4. 5 Load for leg

Resultant Forces

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	3.33123e-05	3.80635e-05	-10.0003	10.0003

Reaction Moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	0	0	0	0

Free body forces

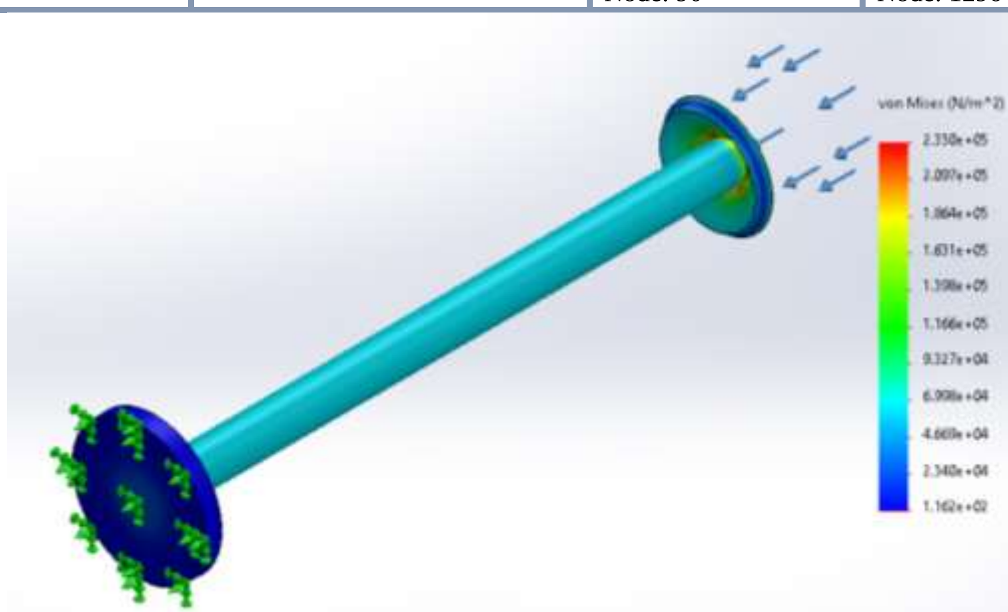
Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	-0.00105131	-0.00133876	-0.000344181	0.00173666

Free body moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	0	0	0	1e-33

Table 4. 6 Resultant Forces for leg

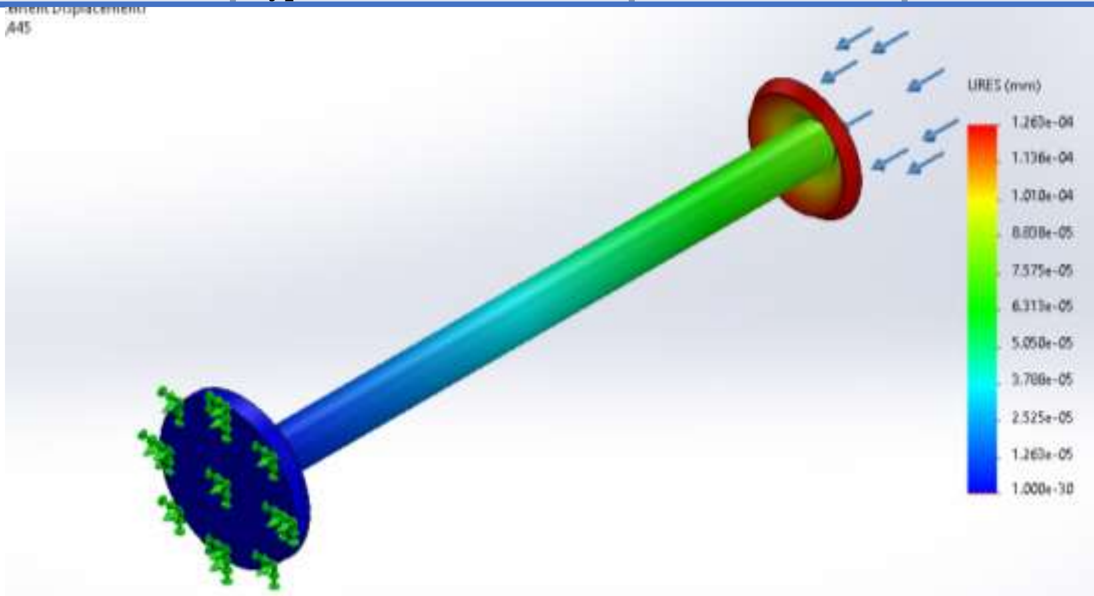
Name	Type	Min	Max
Stress1	VON: von Mises Stress	1.162e+02N/m^2 Node: 90	2.330e+05N/m^2 Node: 1296



leg 2-Static 1-Stress-Stress1

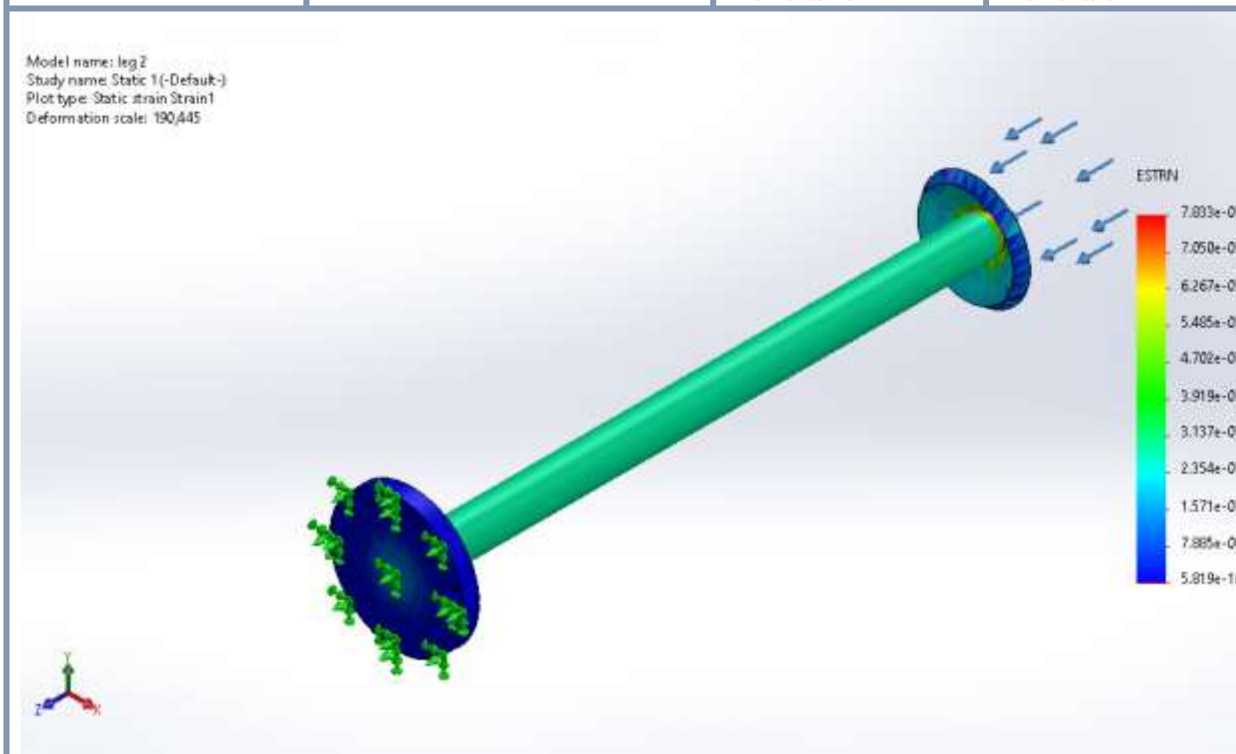
Name	Type	Min	Max
Displacement1	URES: Resultant Displacement	0.000e+00mm Node: 18	1.263e-04mm Node: 691

Name	Type	Min	Max



leg 2-Static 1-Displacement-Displacement1

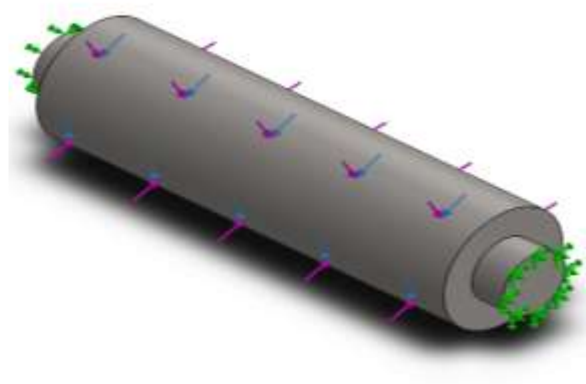
Name	Type	Min	Max
Strain1	ESTRN: Equivalent Strain	5.819e-10 Element: 1914	7.833e-07 Element: 6172



leg 2-Static 1-Strain-Strain1

Table 4. 7 Study result for leg

4.3.2 Stress Analysis for Roller



Model name: roller
Current Configuration: Default

Solid Bodies		
Document Name and Reference	Treated As	Volumetric Properties
Revolve1 	Solid Body	Mass:1.18363 kg Volume:0.000153718 m^3 Density:7,700 kg/m^3 Weight:11.5996 N

Table 4. 8 Model Information for roller

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m^2

Table 4. 9 Units of roller

Model Reference	Properties		Components
	Name:	**Cast Stainless Steel**	**SolidBody 1(Revolve1)(roller)**
	Model type:	**Linear Elastic Isotropic**	
	Default failure criterion:	**Unknown**	
	Elastic modulus:	**1.9e+11 N/m^2**	
	Poisson's ratio:	**0.26**	
	Mass density:	**7,700 kg/m^3**	
	Shear modulus:	**7.9e+10 N/m^2**	
	Thermal expansion coefficient:	**1.5e-05 /Kelvin**	
Curve Data:N/A			

Table 4. 10 Material properties for roller

Fixture name	Fixture Image	Fixture Details
Fixed-1		**Entities:** **2 face(s)** **Type:** **Fixed Geometry**

Resultant Forces

Components	X	Y	Z	Resultant
Reaction force(N)	0.00197972	-0.000711441	0.00116587	0.00240514
Reaction Moment(N.m)	0	0	0	0

Table 4. 11 Fixture for roller

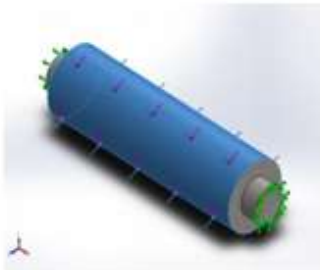
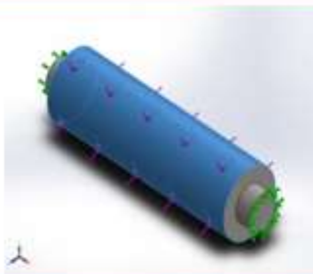
Load name	Load Image	Load Details
Force-1		Entities: 1 face(s) Type: Apply normal force Value: 15 N
Torque-1		Entities: 1 face(s) Type: Apply torque Value: 5 N.m

Table 4. 12 Loads for roller

Reaction forces

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	0.00197972	-0.000711441	0.00116587	0.00240514

Reaction Moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	0	0	0	0

Free body forces

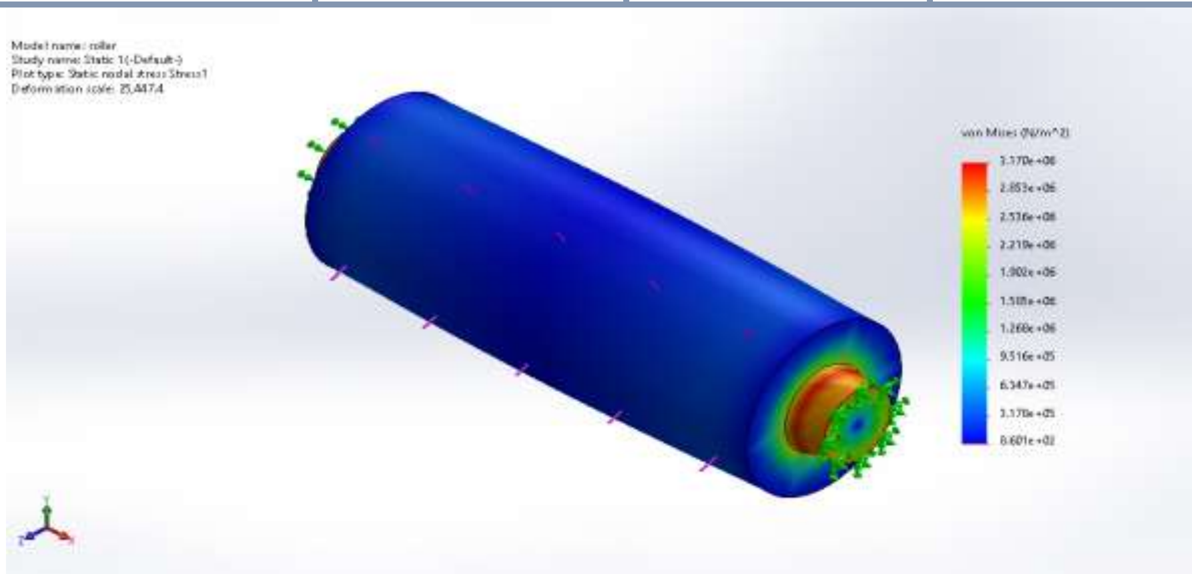
Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	-0.00197499	-0.00708239	5.03706e-06	0.00735261

Free body moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N.m	0	0	0	1e-33

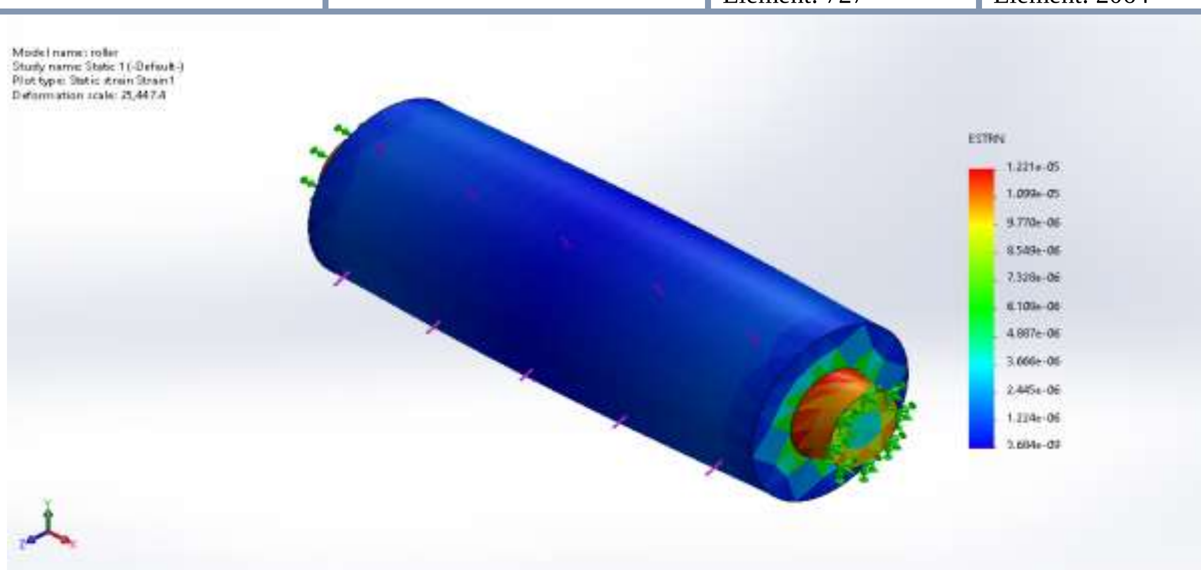
Table 4. 13 Resultant forces for roller

Name	Type	Min	Max
Stress1	VON: von Mises Stress	8.601e+02N/m^2 Node: 5204	3.170e+06N/m^2 Node: 7002



roller-Static 1-Stress-Stress1

Name	Type	Min	Max
Strain1	ESTRN: Equivalent Strain	3.684e-09 Element: 727	1.221e-05 Element: 2064



roller-Static 1-Strain-Strain1

Table 4. 14 Study result for roller

CHAPTER FIVE

RESULT AND DISCUSSION

To verify the validity and claims discussed and presented in the previous chapter, here we are considering the implementation, testing, the whole system components and connecting all phases and implement them with each other using separate physical or hardware components and designing it with its vision system as we mentioned earlier and we will be discussing the results obtained from the system.

First, the PIR Sensor detects the coming bottle once the bottle is detected it will send a signal to the inspector which is the MATLAB toolbox. This signal will be sent through the serial port communication mechanism of MATLAB and Arduino. This can be illustrated in figure 5.1 below.

```
arduino = serial('COM17', 'BaudRate', 9600);
    fopen(arduino);
```

Figure 5. 1 Serial communication between MATLAB and Arduino

Then MATLAB will start processing the bottle image. But In this project, the image is taken from the Directory. The first GUI popup then gives two select buttons to insert the reference image and the image to be inspected. This can be illustrated in figure 5.2 below.

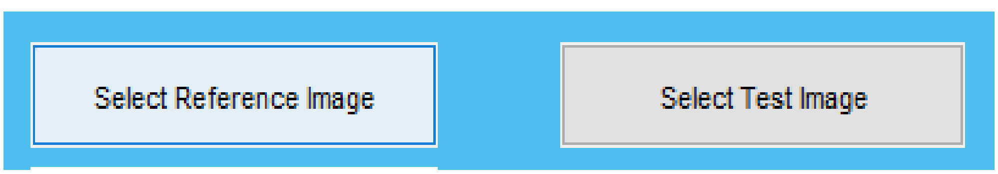


Figure 5. 2 Buttons to insert the reference image and image to be inspected

Now the next step will be inserting the two images from a directory as illustrated in the figure below.

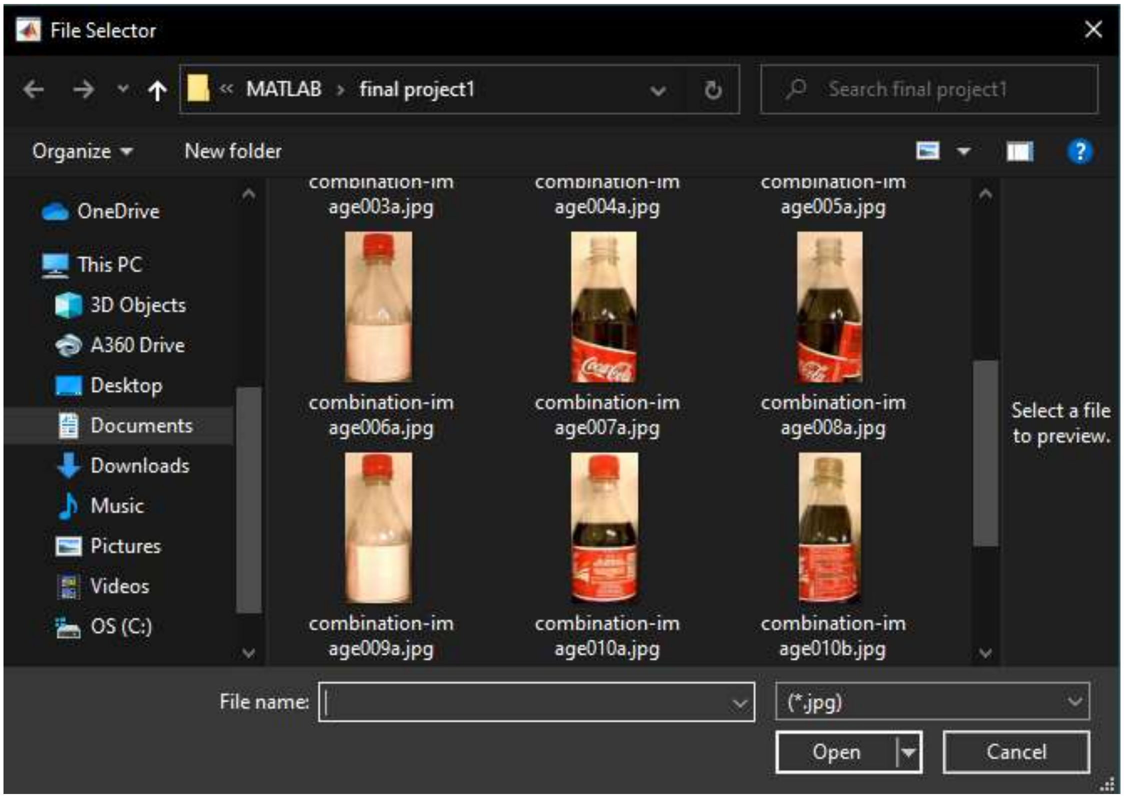


Figure 5. 3 Inserting images

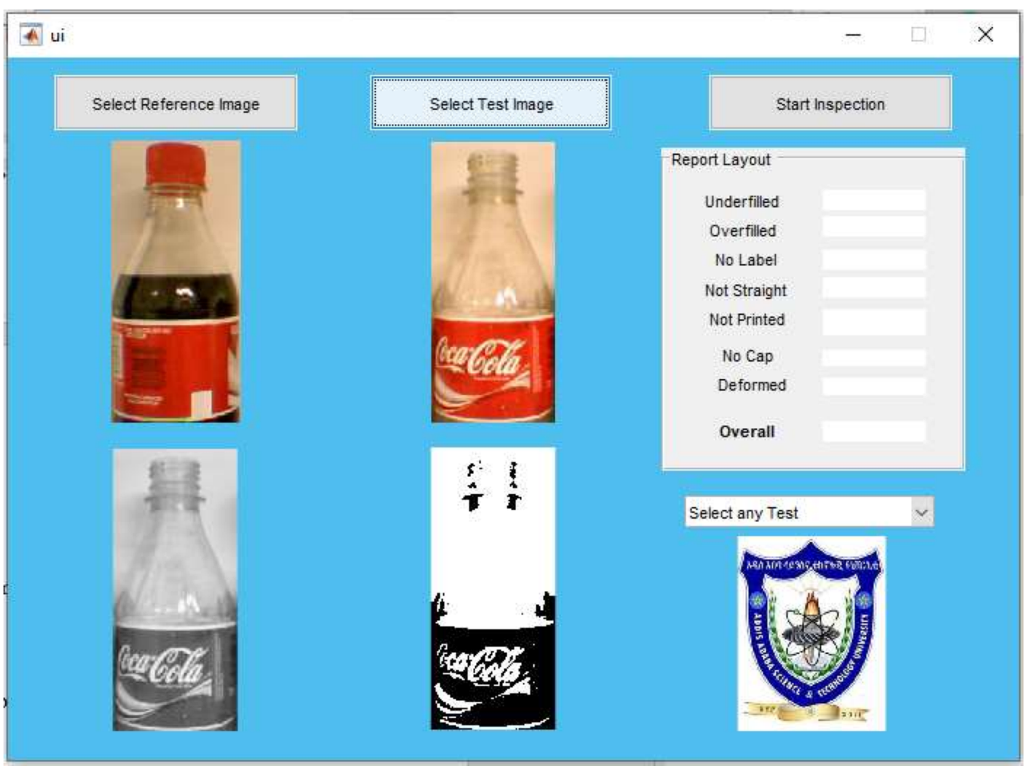


Figure 5. 4 Image selected

Figure 5.4 illustrates the image is ready for inspection. The next step is taking place the inspection using the "start inspection" button.

Figure 5. 5 Start button

After the "start inspection" button is pressed the inspection will be held and then printed out in the report layout panel this can be illustrated in figure 5.5 below.

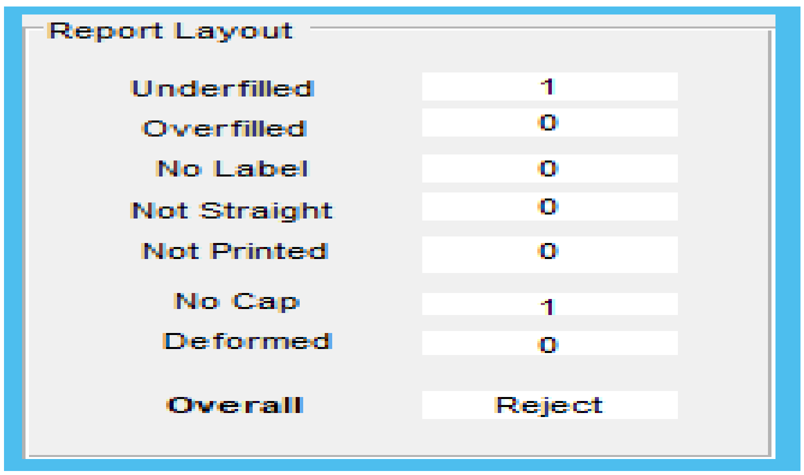


Figure 5. 6 Displayed report

Different defect cases will be discussed below in detail.

Bottle Underfilled and no cap test

Figure 5. 7 Underfilled and no cap bottle

as the image is shown in figure 5.7 the bottle is underfilled relative to the standard (reference bottle) and at the same time it has no bottle cap on it. This will be reported in the report panel using binary numbers ‘0’ and ‘1’. If a defect is present ‘1’ will be printed and the bottle will be rejected else ‘0’ will be printed and the bottle will pass to the next production line. This can be illustrated in the figure below.

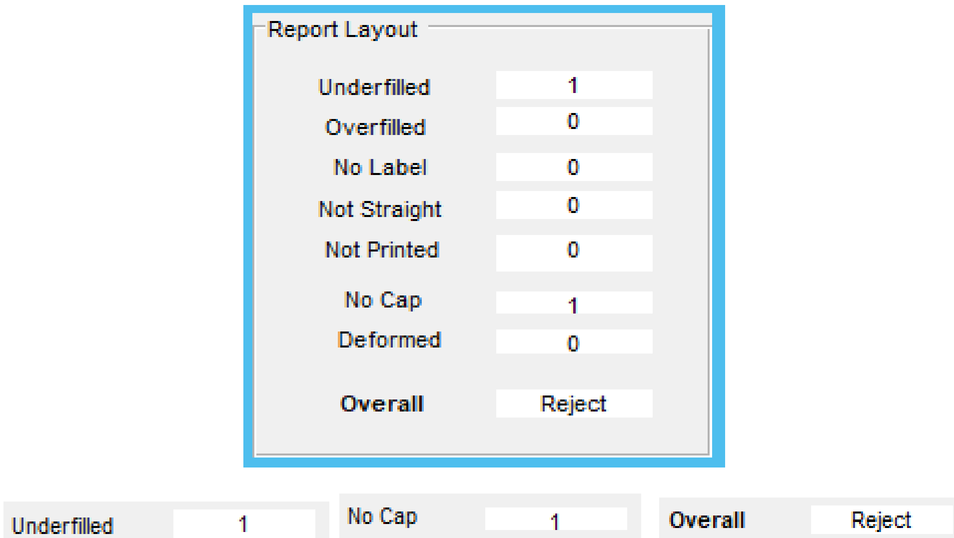


Figure 5. 8 Underfilled and no cap defect

Bottle overfilled and label not printed test

Figure 5. 9 Overfilled and label not printed

as the image is shown in figure 5.9, the bottle is overfilled relative to the standard (reference bottle), and at the same time, no label is printed on the bottle. This will be reported in the report panel using binary numbers ‘0’ and ‘1’. If a defect is present ‘1’ will be printed and the bottle will be rejected else ‘0’ will be printed and the bottle will pass to the next production line. This can be illustrated in the figure below.

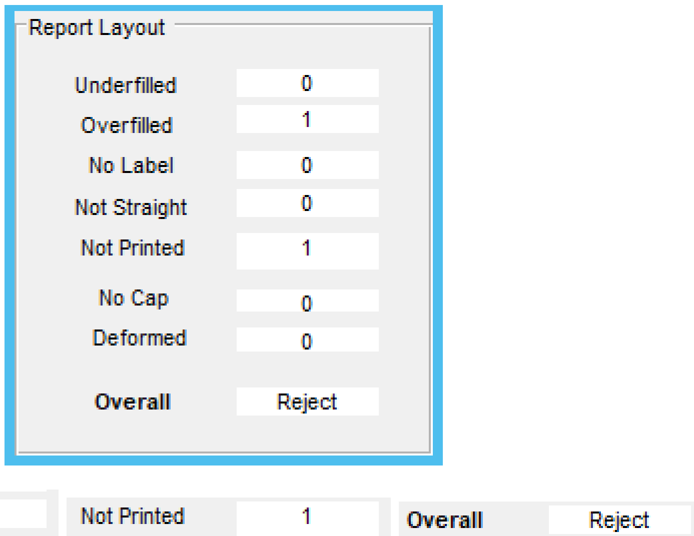


Figure 5. 10 Overfilled and label not printed defect

Standard bottle test

Figure 5. 11 Standard Bottle

as the image is shown in figure 5.11, the bottle has no defect relative to the standard (reference bottle). This will be reported in the report panel using binary numbers ‘0’ and ‘1’. If a defect is present ‘1’ will be printed and the bottle will be rejected else ‘0’ will be printed and the bottle will pass to the next production line. This can be illustrated in the figure below.

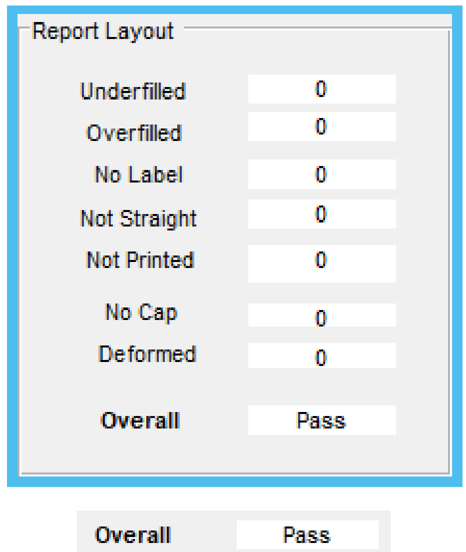


Figure 5. 12 Standard bottle

The same is for all kinds of defects.

After this, if the defect is detected the signal will be sent back to the Arduino board, and one of the Arduino pins will get high (5v). this 5v will be given to the PLC as an input to energize the pneumatic pusher. After some instant time, the defective bottle will reach the pusher, then the pusher will kick it out from the conveyor.

Details of the PLC are simulation results are explained below

Step 1 Press the "START" Push Button. The master coil becomes activated

Step 2 Then the motor will be activated

Step 3 The scanner and the proximity sensor give their input to the shift register. If both inputs are high that particular bit will be set.

Step 4 If the bottle is detected by the proximity sensor after 1 step the conveyor will be stopped for a 2-sec duration for scanning.

Step 5 If both the scanner and the proximity sensors are high red light will be shown which indicates a defective bottle otherwise we see Green light at every step.

Step 6 If the scanner detects the bottle is faulty after 4 steps the rejecter will be energized for a 2-sec duration

First Rung

The figure shown below illustrates in the first rung, there are two input push-button controls I10 and I1, one start button and one-stop button. If the start button is pressed, the master coil (Q1) is ON and even if the start button is OFF, the position of the master coil continues to be ON. To stop the conveyor system, a stop button is used so the output relay will disconnect the power of the memory bit. The above technique is called latching and unlatching. In case of emergency, we use the emergency push button to terminate the whole process.

Figure 5. 13 Latching

Second Rung

The figure shown below illustrates in the second rung there are two inputs, the master coil (Q1) and inspector input (Q6). So the motor will be energized when the two cases are satisfied. Which is the master coil Q1 is 1 and scanner input q6 is 0, otherwise conveyor will be stopped.

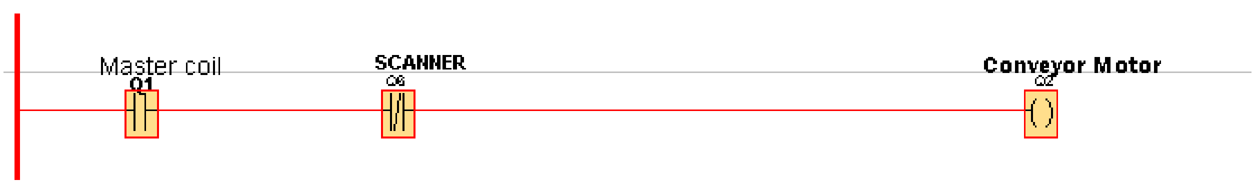


Figure 5. 14 Conveyor ON

Third Rung

In the third rung, there are two inputs, one latched memory bit (Q1) and one sensor input (I2). When sensor (I2) is 1 and the master coil is 1 it triggers the shift register SF001 as we see I3 is open and it becomes closed when the scanner is set on. So when both inputs I3 and I2 are set to 1 that specific memory address of the register is set to 1.

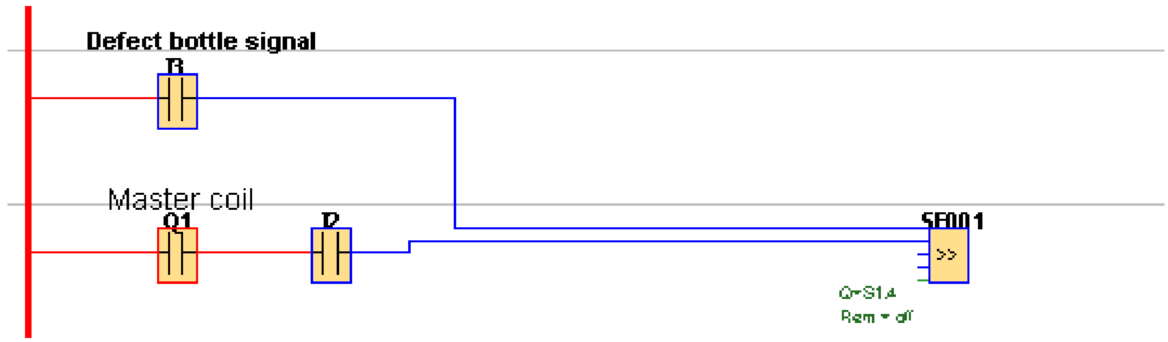


Figure 5. 15 Setting bits

Fourth rung

The figure shown below illustrates in the fourth rung, which uses. the S1.1 bit of the shift register to command the scanner of the bottle in the front. When S1.1 is 1 timer1(TOO3) is set and SCANNER will set to 1. For a delay of 2 sec.

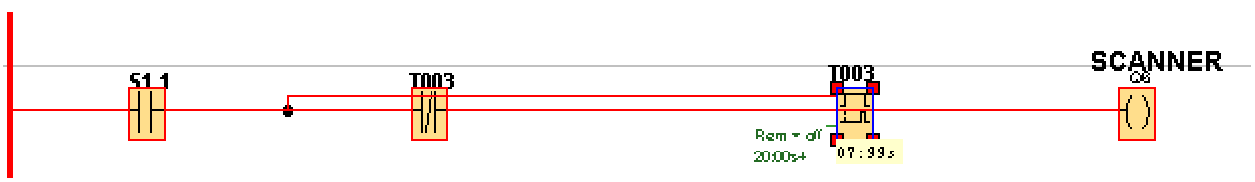


Figure 5. 16 Inspector ON

Fifth Rung

On the fourth rung, there are to indicate lights red light (Q5) and a green light (Q4). When both proximity sensor input I2 and defect bottle signal I3 are set to 1 then the red light (Q5) will be ON. Otherwise, green light Q4 will be on which indicates the normal process is running.

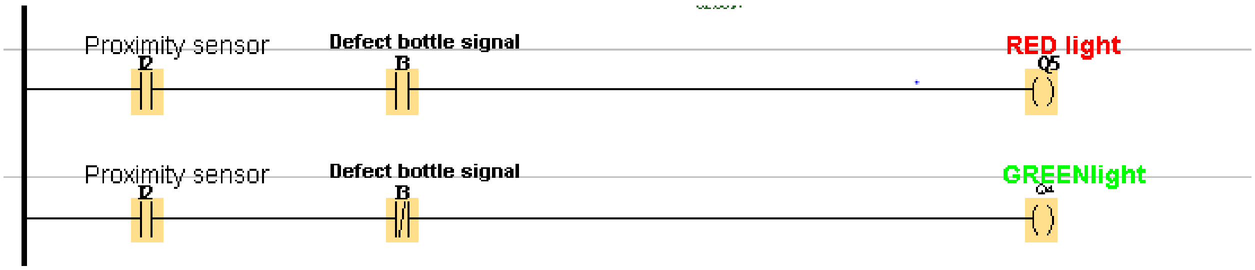


Figure 5. 17 Indicating lights

Sixth rung

On the fourth rung, rejecter will be controlled. The rejecter mechanism will be energized when the fourth bit of the shift register S1.4 is set to 1 and also timer2 (T002) is set. The rejecter will be ON for a delay of 2 sec which is controlled by the timer.

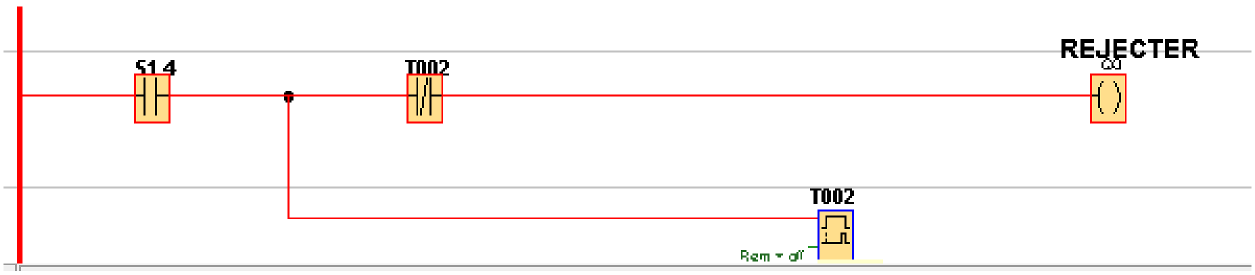


Figure 5. 18 Rejector ON

Factory IO result

let the green box is the bottle that has no defect on it and the blue box is the bottle with defects. So the defective bottle will be rejected by another production line. this can be illustrated in figure 5.19 below.

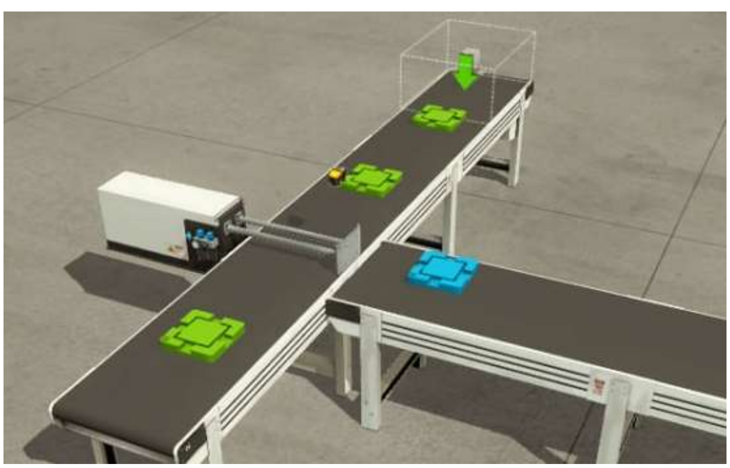


Figure 5. 19 Factory IO simulation result

CHAPTER SIX

CONCLUSION AND RECOMMENDATION

6.1 Conclusion

The project was undertaken to explore the practicality of automated bottle inspection and rejection by using a computer vision and PLC system and it has been designed to inspect the defected bottle and reject if there is a defect on the bottle along the production line.

This recapitulation confers the fundamental design of a computer vision and PLC-based faulty bottle identification and rejection system respectively that could be effectively used for industrial purposes. A simple assessment logic scheme was used to process the outputs of the above system to indicate whether the bottle is perfect or not. Automated vision and PLC-based inspector and rejector system is an effective system in inspecting and rejecting the defected bottle if the components are properly designed and integrated with the system.

The above work presents an automated inspection of bottle caps, level, label, and bottle deformation using a machine vision system, which can inspect bottles and detect defects. This is achieved by using the image processing technique. The present work provides a lot of applications in the field of low-cost automation by using computer vision systems, especially in the inspection process where there is a need to improve the quality of the inspection. The programming of this system developed is flexible, quick, and easy to understand. The same as for the rejection system, the logic of the system is very accurate, quick, and cost-effective. This will increase the total quality control and eliminate human inspector and rejector efforts.

The main component of the system is designed in MATLAB, Arduino, Proteus8, SolidWorks, and factory IO. And, the material is selected from a different perspective, three main consideration is taken to select the appropriate materials, such as Mechanical properties, electrical properties, and cost. It also calculated stress analysis of components, checked their safety factor, and found that all components of the conveyor are safe and will work efficiently under the given load requirements. The same as for the electrical part it is tasted on Proteus8.

In conclusion, therefore, the project successfully explored analytically the practicality of the automated defected bottle inspection and rejection system.

6.2 Major Finding

In starting every study or a project, it should have objectives that are to be reached and goals being met after all has been done. This project used some objectives for the establishment until implementation to reach those objectives.

1. Design a webcam-based inspection system via MATLAB to inspect bottles arrived at the position determined in the conveyor The goal of this objective was to design a wireless camera attached to the MATLAB, that can detect live and stream frames from a visual field and its almost reached, we have used Droidcam app to send a stream of images to code and then image enhancement process starts to eliminate the unwanted objects in the image using a variety of algorithms from filtering, noise removal, using blob analysis. To get the best results; one important thing to remember is the distance between the camera and the bottle which must be constant for this study we took 20cm. we have avoided too much noise in the field by taking white background that makes it easy to separate the bottle from the background or other noise in the area.

2. Implement a conveyor line that works with a motor with PIR sensors to detect visibility and the position of the bottle. In this part we have implemented a whole system of a conveyor that contains a PLC Module, mainframe, conveyor belts, sensors, gear type DC motor, power supply, cameras, and other mechanical elements, which the results we have got were accurate, we have set the speed of the motor by using relay and Arduino to control the currents in the motor, in sensor part, we have used PIR motion sensor to detect the presence of the bottle in the field, if it detects, the sensor sends a signal to the motor to stop for image processing purposes, after two seconds the bottle moves again until another sensor detects to stop the DC motor. Along this process, the rejection system is also implemented at some point of the conveyor after the inspection area. This system is implemented using a pneumatic rejector.

6.3 Recommendation

Although the goal of the project was mostly covered, further research can be aimed at developing and enhancing some of the shortcomings of the current simulations and extending the ideas explored here to address several other relevant projects in emerging modern vision inspection systems and replacing conventional PLCs with Controlling since they are more features and flexibilities.

Finally, its recommended to use a variety of different algorithms for feature extraction to get better results, also object prediction multiple Gaussian mixture models to remove additional unwanted moving objects in the video that are not part of the foreground bottles which is moving in a constant velocity.

REFERENCE

[1] A. Roy, S. Sinha, S. Nandi, R. Chakraborty, K. Chatterjee, Z. Haldar and Apurba Ghosh, "*PLC based home security system by simulation method*," MDCCT, March, 2010.

[2] Domingo Mery and Olaya Medina, "*Automated visual inspection of glass bottles using adapted median filtering*", IEEE International Journal, 2010.

[3] John W. webb, Ronald A. Reis, "*Programmable Logic Controller , Principles & Applications*," 5th Edition, PHI, 2003.

[4] Li, F., Hang, Z., Yu, G., Wei, G., & Xinyu, C. (2017). *The Method for Glass Bottle Defects Detecting Based on machine vision*, 2–5. https://doi.org/10.1109/CCDC.2017.7978567

[5] Mario Campos, Manuel Ferreira, Teresa Martins, "*Inspection of bottle crates in the Beer Industry through Computer Vision*", IEEE International Journal, 2010.

[6] Neal Babcock, "*Beginner's Guide to PLC Programming*," modern media, 2008.

[7] S. Ghosh, S. Bairagya, C. Roy, S. Dey, S. Goswami and Apurba Ghosh, "*Bottle filling system using PLC*," In Proc. Nat. Conf. ET&TE 2008.

[8] Yang, Z. (2015). *Vial Bottle Mouth Defect Detection Based on Machine Vision. 2015 IEEE International Conference on Information and Automation*, (August), 2638– 2642. https://doi.org/10.1109/ICInfA.2015.7279730

Appendices

Appendix I

Underfilled bottle test

```
function result = CheckBottleUnder_filled(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[25,135,79.98,10.98]);
gray_cropped_T = rgb2gray(cropped_T);%change the cropped image in to gray scale image
BW = imbinarize(gray_cropped_T, double(150/256));
BW = imfill(BW,'holes');
nBlack = sum(BW(:) == 0);
nWhite = numel(BW(:));
percentage = nBlack/nWhite;
if percentage < 0.25
    result = 1;
else
    result = 0;
end
end
```

overfilled bottle test

```
function result = CheckBottleOver_filled(O,T)
im_O = O;
im_T = T;

% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[55,84.51,25.98,80.98]);
gray_cropped_T = rgb2gray(cropped_T);%change the cropped image in to gray scale image
```

```
BW = imbinarize(gray_cropped_T, double(100/256));
BW = imfill(BW,'holes');
nBlack = sum(BW(:) == 0);
nWhite = numel(BW(:));
percentage = nBlack/nWhite;

if percentage > 0.4
   result  = 1;
else
   result = 0;
end
end
```

Missing Label test

```
function result = CheckMissingLabel(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[13,192.51,109.98,67.98]);
gray_cropped_T = rgb2gray(cropped_T);%change the cropped image in to gray scale image
BW = imbinarize(gray_cropped_T, double(45/256));
nBlack = sum(BW(:) == 0);
nWhite = numel(BW(:));
percentage = nBlack/nWhite;
if percentage > 0.50;
   result = 1;
else
   result = 0;
end
end
```

Label Not Printed test

```
function result = LabelNotPrinted(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[20,192.51,109.98,67.98]);
gray_cropped_T = rgb2gray(cropped_T);%change the cropped image in to gray scale image
BW = imbinarize(gray_cropped_T, double(145/256));
nBlack = sum(BW(:) == 0);
nWhite = numel(BW(:));
percentage = nBlack/nWhite;
if percentage < 0.50;
   result = 1;
else
   result = 0;
end
end
```

Bottle cap missing test

```
function result = BottleTopMissing(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[25,5,78.01,54.01]);
YCBCR = rgb2ycbcr(cropped_T);
red_channel = YCBCR(:,:,3);
BW = imbinarize(red_channel, double(170/256));
nBlack = sum(BW(:) == 0);
nWhite = numel(BW(:));
```

```
percentage = (nBlack/nWhite);
if percentage > 0.8
    result = 1;
else
    result = 0;
end
end
```

straight label test

```
function result = CheckIfLabelIsStr8t(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image , height of the image]
cropped_T = imcrop(im_T,[5,154.51,106.98,100.98]);
red_channel = cropped_T(:, :, 1);
imageR = imadjust(red_channel);
BW1 = edge(imageR,'sobel','horizontal');
BW = bwareaopen(BW1,45);
cutskelimg = bwmorph(BW,'skel');
mn =bwmorph(cutskelimg,'endpoints');
[row, column] = find(mn);
x = length(column);
Angle1 = atan((row(x)- row(1))/(column(x)-column(1)));
if Angle1 > 0.08 && (column(x)-column(1)) < 120
    result = 1;
else
    result = 0;
end
end
```

Deformed Bottle test

```
function result = DeformedBottle(O,T)
im_O = O;
im_T = T;
% crope the image [min value of x to be cropped , min value of y to be cropped, width of the image
, height of the image]
cropped_T = imcrop(im_T,[4,163.51,131.98,31.98]);
red_channel = cropped_T(:, :, 1);
imageR = imadjust(red_channel);
BW1 = edge(imageR,'sobel','horizontal');
BW = bwareaopen(BW1,60);
cutskelimg = bwmorph(BW,'skel');
mn =bwmorph(cutskelimg,'endpoints');
[row, column] = find(mn);
x = length(column);
if (column(x)-column(1)) < 110
    result = 1;
else
    result = 0;
end
end
```

Graphical user interface main code

```
function Reference_image_Callback(hObject, eventdata, handles)
% hObject    handle to Reference_image (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[filename pathname] = uigetfile({'*.jpg';'*.bmp'},'File Selector');
 handles.myImage = strcat(pathname, filename);
 axes(handles.axes1);
```

```
 imshow(handles.myImage)
 guidata(hObject,handles);

function test_image_Callback(hObject, eventdata, handles)
% hObject    handle to test_image (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[filename pathname] = uigetfile({'*.jpg';'*.bmp'},'File Selector');
 handles.myImage = strcat(pathname, filename);
axes(handles.axes2);
 imshow(handles.myImage)
 imh = findobj(handles.axes1, 'type', 'image');
imh = findobj(handles.axes2, 'type', 'image');
T = get(imh, 'CData');
gray_T = rgb2gray(T);
axes(handles.axes3);
imshow(gray_T);
bw_T = im2bw(T, 0.6);
axes(handles.axes4);
imshow(bw_T);
 guidata(hObject,handles);

function popupmenu1_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
contents = get(hObject,'Value');
imh = findobj(handles.axes1, 'type', 'image');
O = get(imh, 'CData');
imh = findobj(handles.axes2, 'type', 'image');
T = get(imh, 'CData');
```

```
switch contents
   case 1
      myicon =  imread('delete.png');
      msgbox('Please select a test to proceed','ERROR !!','custom',myicon);

   case 2
      if CheckBottleOver_filled(O,T) == 1
         disp('Overfilled Bottle')
         set(handles.text40, 'string', '1');
         set(handles.text35, 'string', '0');
      elseif CheckBottleUnder_filled(O,T) == 1
         disp('Underfilled Bottle')
set(handles.text40, 'string', '0');
         set(handles.text35, 'string', '1');
      else
         disp('Standard Level')
         set(handles.text40, 'string', '0');
         set(handles.text35, 'string', '0');
      end
   case 3
      if BottleTopMissing(O,T) == 1
         disp('No Cap');
         set(handles.text38, 'string', '1');
      else
         disp('Bottle has Cap');
         set(handles.text38, 'string', '0');
      end
   case 4
      if CheckMissingLabel(O,T) == 0
         set(handles.text41, 'string', '0');
         if LabelNotPrinted(O,T) == 0 && CheckIfLabelIsStr8t(O,T) == 0
```

```
            disp('label has no defect');
            set(handles.text40, 'string', '0');
            set(handles.text36, 'string', '0');
        elseif CheckIfLabelIsStr8t(O,T) == 1
            disp('label is not straight');
            set(handles.text36, 'string', '1');
        else
            disp('no label printed on the bottle');
            set(handles.text36, 'string', '0');
            set(handles.text44, 'string', '1');
        end
    else
        disp('No label in the bottle')
        set(handles.text41, 'string', '1');
    end
case 5
    if DeformedBottle(O,T) == 1
        disp('Deformed Bottle');
        set(handles.text43, 'string', '1');
    else
        disp('No Deformation');
        set(handles.text43, 'string', '0');
    end
  case 6
    if CheckMissingBottle(O,T) == 0 && BottleTopMissing(O,T) == 0 && CheckBottleUnder_filled(O,T) == 0 && CheckMissingLabel(O,T) == 0 && LabelNotPrinted(O,T) == 0 && CheckBottleOver_filled(O,T) == 0 && CheckIfLabelIsStr8t(O,T) == 0 && DeformedBottle(O,T) == 0
    disp('Bottle is Standard!')
    set(handles.text35, 'string', '0');
    set(handles.text40, 'string', '0');
```

```
        set(handles.text35, 'string', '0');
        set(handles.text36, 'string', '0');
        set(handles.text44, 'string', '0');
        set(handles.text41, 'string', '0');
        set(handles.text43, 'string', '0');
        set(handles.text38, 'string', '0');
        set(handles.text39, 'string', 'Pass');
        myicon =  imread('tick.png');
        msgbox('Has no Defect','PASS !!','custom',myicon);
        end
if CheckMissingBottle(O,T) == 1 || BottleTopMissing(O,T) == 1 || CheckBottleUnder_filled(O,T)
== 1 || CheckMissingLabel(O,T) == 1 || LabelNotPrinted(O,T) == 1 ||
CheckBottleOver_filled(O,T) == 1 || CheckIfLabelIsStr8t(O,T) == 1 || DeformedBottle(O,T) == 1
        disp('Bottle has Defect!');
        set(handles.text39, 'string', 'Reject');
        myicon =  imread('delete.png');
        msgbox('Has Defect','REJECT !!','custom',myicon);
        end
    otherwise
end
function Start_inspection_Callback(hObject, eventdata, handles)
% hObject    handle to Start_inspection (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
arduino = serial('COM17', 'BaudRate', 9600);
    fopen(arduino);
imh = findobj(handles.axes1, 'type', 'image');
O = get(imh, 'CData');
imh = findobj(handles.axes2, 'type', 'image');
T = get(imh, 'CData');
gray_T = rgb2gray(T);
```

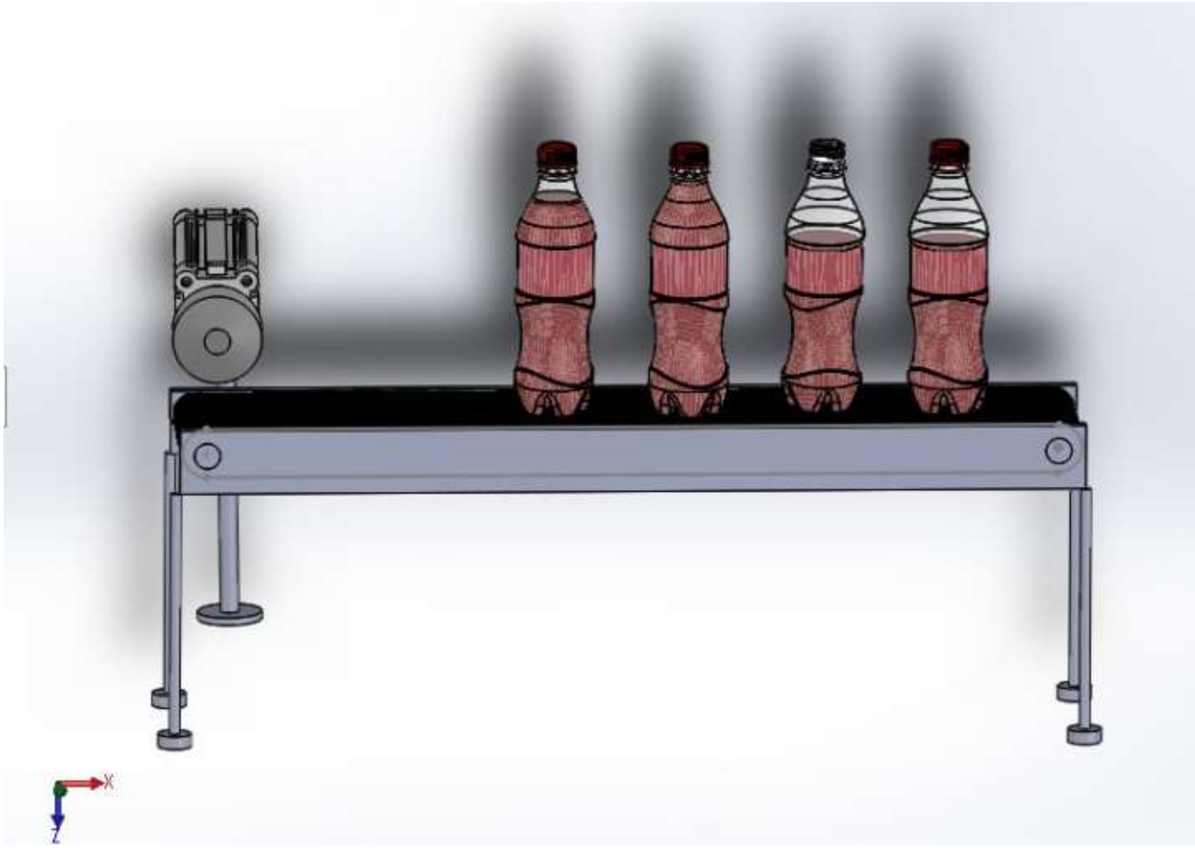
```
axes(handles.axes3);
imshow(gray_T);
bw_T = im2bw(T, 0.6);
axes(handles.axes4);
imshow(bw_T);
   MissingBottleCap = 0;
   Underfilled = 0;
   overfilled = 0;
   labelmissing = 0;
   labelnotPrinted = 0;
   labelnotstraight = 0;
   bottleDeformed = 0;
   count_defected_bottle = 0;
   count_passed_bottle = 0;
   count_MissingBottle = 0;
   count_MissingBottleCap = 0;
   count_Underfilled = 0;
   count_overfilled = 0;
   count_labelmissing = 0;
   count_labelnotPrinted = 0;
count_labelnotstraight = 0;
   count_bottleDeformed = 0;
     isMissing = CheckMissingBottle(O,T);
   if isMissing == 1
     disp('No bottle in the frame');
     count_MissingBottle = count_MissingBottle + 1;
   else
      MissingBottleCap = BottleTopMissing(O,T);
      if MissingBottleCap == 1
         disp('Bottle cap is Missing');
         count_MissingBottleCap = count_MissingBottleCap + 1;
```

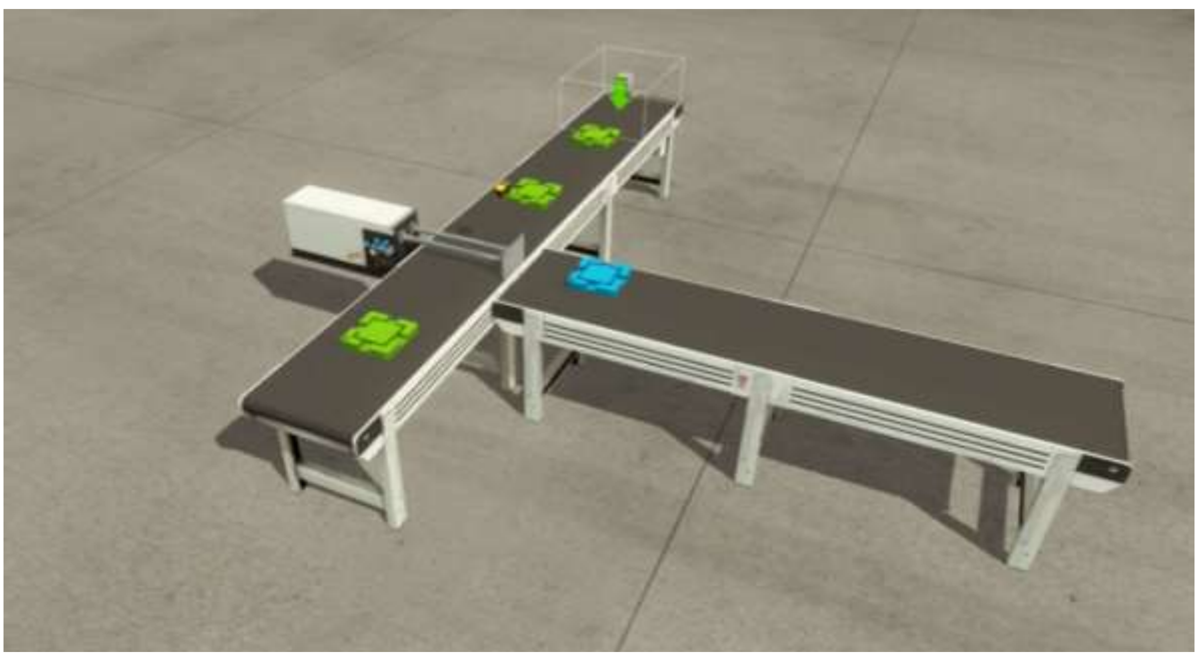
```
    set(handles.text38, 'string', '1');
  else
    set(handles.text38, 'string', '0');
  end
  Underfilled = CheckBottleUnder_filled(O,T);
  if Underfilled == 1
    disp('Bottle is Underfilled');
    count_Underfilled = count_Underfilled + 1;
    set(handles.text35, 'string', '1');
    set(handles.text36, 'string', '0');
    set(handles.text43, 'string', '0');
    set(handles.text44, 'string', '0');
  else
    set(handles.text35, 'string', '0');
  end

  overfilled = CheckBottleOver_filled(O,T);
  if overfilled == 1
    disp('Bottle is Overfilled');
    count_overfilled = count_overfilled + 1;
    set(handles.text40, 'string', '1');
  else
    set(handles.text40, 'string', '0');
end
  labelmissing = CheckMissingLabel(O,T);
  if labelmissing == 1
    disp('Label is Missing');
    count_labelmissing = count_labelmissing + 1;
    set(handles.text41, 'string', '1');
     set(handles.text36, 'string', '0');
    set(handles.text43, 'string', '0');
```

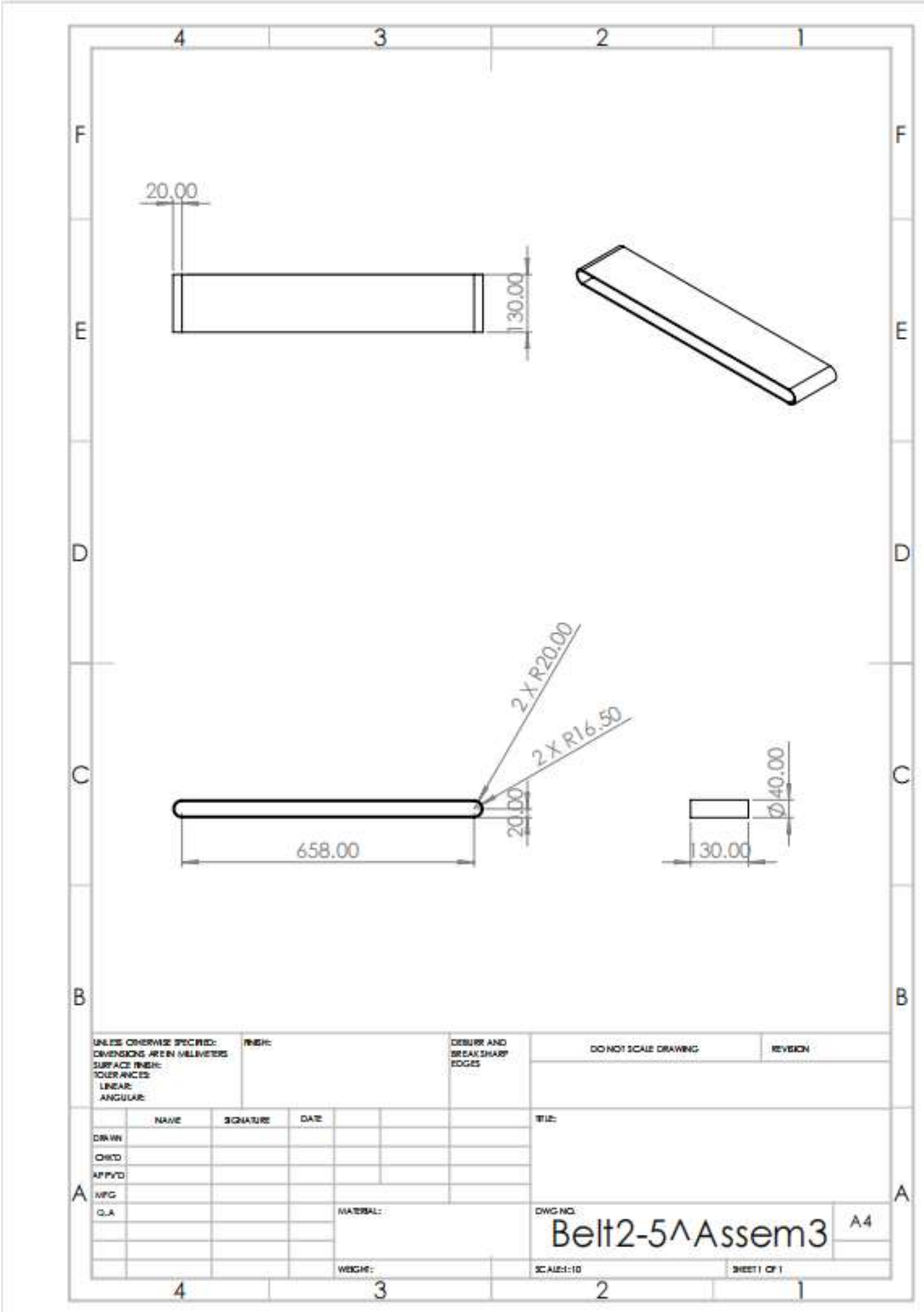
```
    set(handles.text44, 'string', '0');
else
    set(handles.text41, 'string', '0');
    labelnotPrinted = LabelNotPrinted(O,T);
    if labelnotPrinted == 1
        disp('Label not printed')
        count_labelnotPrinted = count_labelnotPrinted + 1;
        set(handles.text44, 'string', '1');
        set(handles.text43, 'string', '0');
        set(handles.text36, 'string', '0');
    elseif labelnotPrinted == 0 && Underfilled == 0
        labelnotstraight = CheckIfLabelIsStr8t(O,T);
        if labelnotstraight == 1
          disp('Label is not Straight')
          count_labelnotstraight = count_labelnotstraight + 1;
          set(handles.text36, 'string', '1');
          set(handles.text43, 'string', '0');
        else
            set(handles.text36, 'string', '0');
            bottleDeformed = DeformedBottle(O,T);
            if bottleDeformed == 1
                disp('Bottle is Deformed');
                count_bottleDeformed = count_bottleDeformed+ 1;
                set(handles.text43, 'string', '1');
            else
                set(handles.text43, 'string', '0');
            end
        end
        set(handles.text44, 'string', '0');
    end
end
```

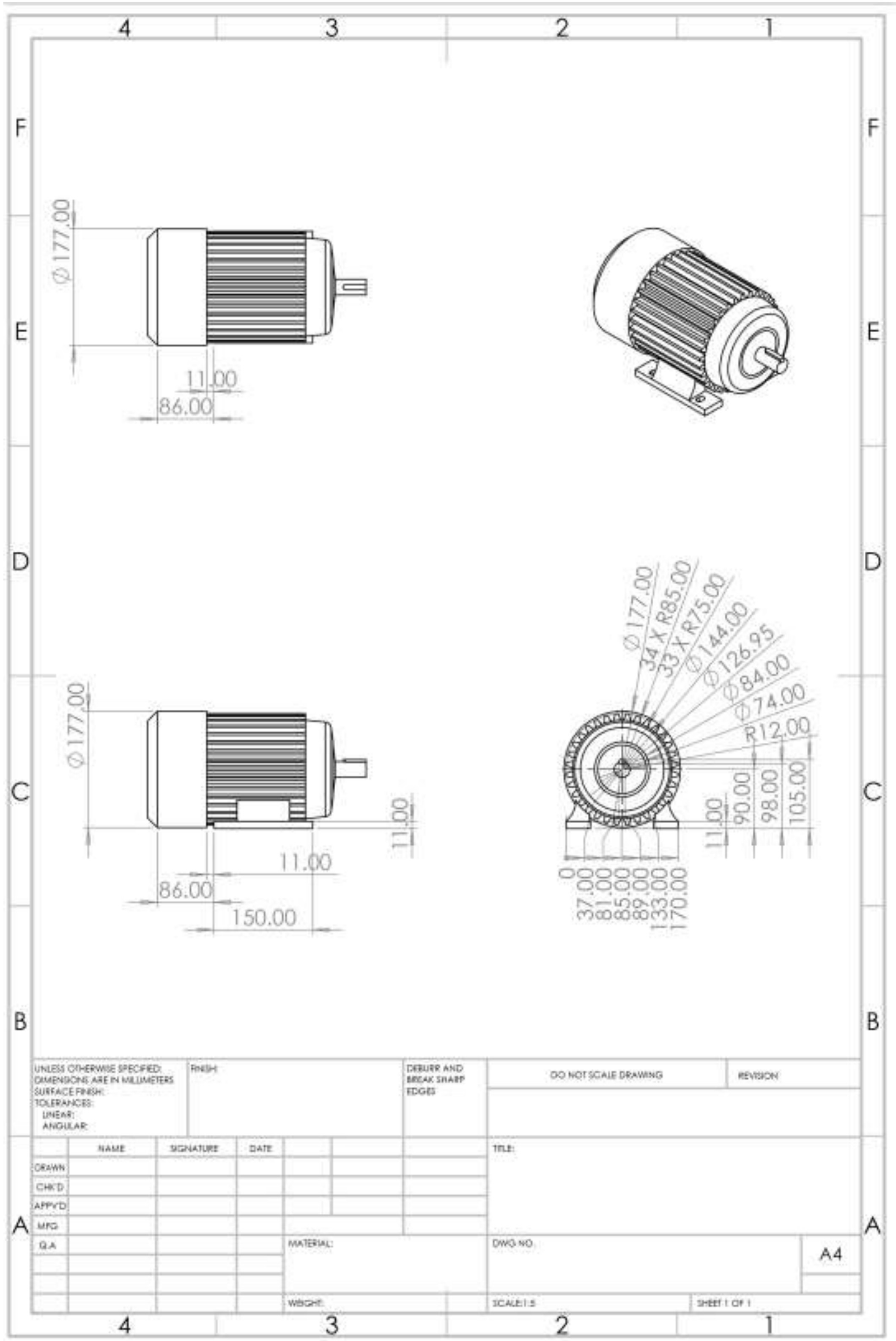
```
end
if CheckMissingBottle(O,T) == 0 && BottleTopMissing(O,T) == 0 && CheckBottleUnder_filled(O,T) == 0 && CheckMissingLabel(O,T) == 0 && LabelNotPrinted(O,T) == 0 && CheckBottleOver_filled(O,T) == 0 && CheckIfLabelIsStr8t(O,T) == 0 && DeformedBottle(O,T) == 0
    disp('Bottle is Standard!')
    count_passed_bottle = count_passed_bottle + 1;
    set(handles.text39, 'string', 'Pass');
    fprintf(arduino, 0);
end
if CheckMissingBottle(O,T) == 1 || BottleTopMissing(O,T) == 1 || CheckBottleUnder_filled(O,T) == 1 || CheckMissingLabel(O,T) == 1 || LabelNotPrinted(O,T) == 1 || CheckBottleOver_filled(O,T) == 1 || CheckIfLabelIsStr8t(O,T) == 1 || DeformedBottle(O,T) == 1
    disp('Bottle has Defect!')
    count_defected_bottle = count_defected_bottle + 1;
    set(handles.text39, 'string', 'Reject');
    fprintf(arduino, 1);
end
fclose(arduino);
```

Appendix II

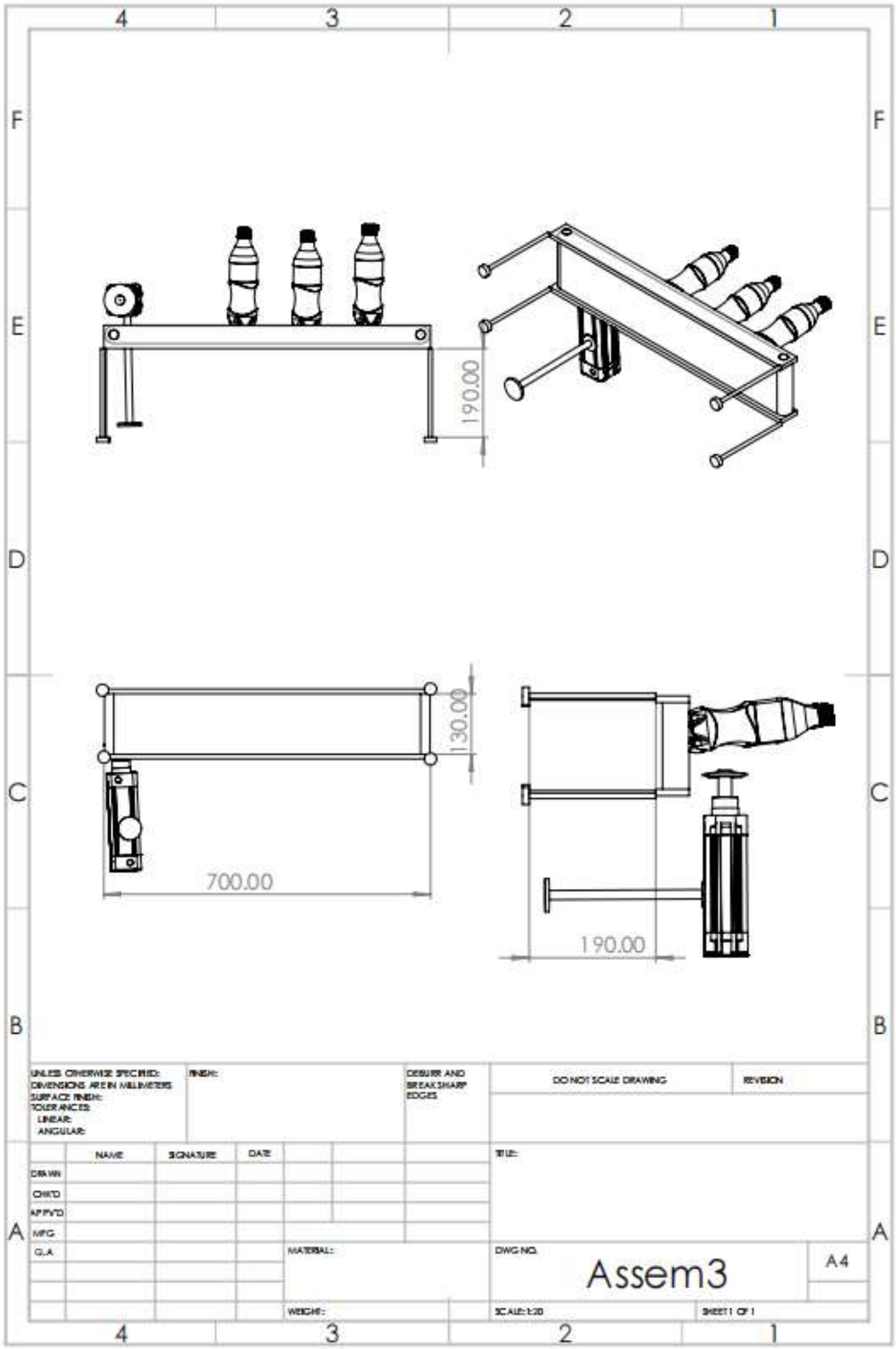





20.00
130.00
2 X R20.00
2 X R16.50
20.00
658.00
Ø40.00
130.00
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:
FINISH:
DEBURR AND BREAK SHARP EDGES
DO NOT SCALE DRAWING
REVISION
NAME
SIGNATURE
DATE
DRAWN
CHK'D
APPV'D
MFG
Q.A
MATERIAL:
WEIGHT:
TITLE:
DWG NO.
Belt2-5^Assem3
A4
SCALE:1:10
SHEET 1 OF 1



Ø177.00
11.00
86.00
150.00
Ø177.00
34 X R85.00
33 X R75.00
Ø144.00
Ø126.95
Ø84.00
Ø74.00
R12.00
90.00
98.00
105.00
0
37.00
81.00
85.00
89.00
133.00
170.00
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:
FINISH:
DEBURR AND BREAK SHARP EDGES
DO NOT SCALE DRAWING
REVISION
NAME
SIGNATURE
DATE
DRAWN
CHK'D
APPV'D
MFG
Q.A
TITLE:
MATERIAL:
DWG NO.
A4
WEIGHT:
SCALE:1:5
SHEET 1 OF 1


190.00
130.00
700.00
190.00
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN MILLIMETERS
SURFACE FINISH:
TOLERANCES:
LINEAR:
ANGULAR:
FINISH:
DEBURR AND BREAK SHARP EDGES
DO NOT SCALE DRAWING
REVISION
NAME
SIGNATURE
DATE
DRAWN
CHK'D
APPV'D
MFG
Q.A
TITLE:
MATERIAL:
DWG NO.
Assem3
A4
WEIGHT:
SCALE:1:20
SHEET 1 OF 1