UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Faculty of Electrical and Electronic Engineering Technology

DEVELOPMENT OF MODULAR PRODUCTION SYSTEM (MPS) WITH SENSOR'S AND ACTUATOR'S PREDICTION FOR MAINTENANCE USING PROGRAMMABLE LOGIC CONTROLLER (PLC)

AHMAD TIRMIZI BIN AZMAN

Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours

2021

DEVELOPMENT OF MODULAR PRODUCTION SYSTEM (MPS) WITH SENSOR'S AND ACTUATOR'S PREDICTION FOR MAINTENANCE USING PROGRAMMABLE LOGIC CONTROLLER (PLC)

AHMAD TIRMIZI BIN AZMAN

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I declare that this project report entitled "Development of Modular Production System (MPS) with Sensor's and Actuator's Prediction for Maintenance using Programmable Logic Controller (PLC)" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature :

Student Name : Ahmad Tirmizi bin Azman

Date : 10/1/2022

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours.

Signature	:	shahrizal
Supervisor Name	:	Ts. Shahrizal bin Saat
Date	:	10/1/2022

DEDICATION

To my beloved mother and father, thank you for the courage word and support given to me in order to fulfil this report.

ABSTRACT

Nowadays, with the revolution IR 4.0, most industry have used the automatic machines. The purpose of this project is to study about prediction maintenance for sensor and actuator by using Programmable Logic Controller (PLC). Sensor and actuator most important for industry that use the automation system. Besides, one of the limitations is human ability to find out the damage of the sensor and actuator. These systems can notify the state of sensor and actuator so that the factory can detect of how much the machine works and it can prevent unpredictable system damage. The PLC is used as a medium in giving instruction to MPS where the PLC programmed to control each step of MPS. MPS is an automatic machine which it has a sensor and actuator for automation system in the industry. The PLC count and store the data of cycle sensor and actuator, then sent to HMI screen. Human Machine Interface (HMI) is a graphical interface that allows humans and machines interact. HMI screen display the state of lamp for condition of sensor and actuator. By The state of lamp depends on how the machine functioning. There are 3 state that can be implant in this project to make the sensor and actuator in the MPS have the features like smart sensor and actuator. If the sensor and actuator are in good condition, the lamp will show green colour. If the sensor and actuator is almost reaching the ecpectacy of the lifespan then it is in state of alert, the lamp will turn to yellow.Last but not least, when the sensor and actuator already reach the expectac lifespan then its the condition of the sensor and actuator needs to be checked.

ABSTRAK

Pada masa ini, dengan revolusi IR 4.0, kebanyakan industri telah menggunakan mesin automatik. Tujuan projek ini adalah untuk mengkaji mengenai penyelenggaraan ramalan untuk sensor dan penggerak dengan menggunakan Programmable Logic Controller (PLC). Sensor dan penggerak paling penting untuk industri yang menggunakan sistem automasi. Selain itu, salah satu batasannya adalah kemampuan manusia untuk mengetahui kerosakan sensor dan penggerak. Sistem ini dapat memberitahu keadaan sensor dan penggerak sehingga kilang dapat mengesan berapa banyak mesin berfungsi dan dapat mencegah kerosakan sistem yang tidak dapat diramalkan. PLC digunakan sebagai media dalam memberi arahan kepada MPS di mana PLC diprogramkan untuk mengawal setiap langkah MPS. MPS adalah mesin automatik yang mempunyai sensor dan penggerak untuk sistem automasi dalam industri. PLC mengira dan menyimpan data sensor kitaran dan penggerak, kemudian dikirim ke layar HMI. Human Machine Interface (HMI) adalah antara muka grafik yang membolehkan manusia dan mesin berinteraksi. Skrin HMI memaparkan keadaan lampu untuk keadaan sensor dan penggerak. Keadaan lampu bergantung pada jarak fungsi mesin. Terdapat 3 keadaan yang dapat diserapkan dalam kaian ini untuk menjadikan sensor and penggerak biasa dalam MPS mempunyai ciri-ciri seperti sensor dan penggerak pintar. Sekiranya sensor dan penggerak berada dalam keadaan baik, lampu akan menunjukkan warna hijau. Jika sensor dan penggerak hampir mencapai jangka hayat penyelenggaraan ramalan maka ia akan berada dalam keadaan berjaga-jaga, lampu akan bertukar kepada warna kuning. Akhir sekali, apabila sensor dan penggerak sudah mencapai jangka hayat penyelenggaraan ramalan maka sensor dan penggerak perli diperiksa, lampu akan bertukar kepada warna merah.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Ts. Shahrizal bin Saat for his precious guidance, words of wisdom and patient throughout this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support through lending me the equipment in the labotary which enables me to accomplish the project. Not forgetting my fellow colleague, housemates for the willingness of sharing his thoughts and ideas regarding the project.

My highest appreciation goes to my parents, and family members for their love and prayer during the period of my study.

Finally, I would like to thank all the staffs at the Industrial Control Laboratory, fellow colleagues and classmates, as well as other individuals who are not listed here for being co-operative and helpful.

TABLE OF CONTENTS

		PAGE
DECLARATION		
APPROVAL		
DEDICATIONS		
ABSTRACT		**i**
ABSTRAK		**ii**
ACKNOWLEDGEMENTS		**iii**
TABLE OF CONTENTS		**i**
LIST OF TABLES		**iii**
LIST OF FIGURES		**v**
LIST OF ABBREVIATIONS		**viii**
CHAPTER 1	**INTRODUCTION**	**9**
1.1	Background	9
1.2	Problem Statement	10
1.3	Project Objective	11
1.4	Scope of Project	11
CHAPTER 2	**LITERATURE REVIEW**	**12**
2.1	Introduction	12
2.2	Research by Journal	12
2.2.1	Design of Automatic Sleeve for Transfer Nut Clutch using Programmable Logic Controller.	12
2.2.2	Fault Identification and Protection of Induction Motor using PLC and SCADA	14
2.2.3	Automation of Packaging and Material Handling Using Programmable Logic Controller.	15
2.2.4	Designing Human Machine Interface for Vehicle's EFI Engine using Siemen's PLC and SCADA System	16
2.2.5	Controlling Process of a Bottling Plant using PLC and SCADA	18
2.2.6	Application of IoT Concept on Predictive Maintenance of Industrial Equipment	20
2.2.7	Intelligent and Predictive Maintenance in Manufacturing System	21
2.2.8	Initiating Predictive Maintenance for a Conveyor Motor in a Bottling Plant using Industry 4.0 Concepts	23
2.2.9	Smart Garden Monitoring System Using IoT	24
2.2.10	Smart Management of Technologies: Predictive Maintenance of Industrial Equipment using Wireless Sensor Networks	25

2.3 Table of Summary 26
2.4 Summary 28

CHAPTER 3 METHODOLOGY 29
3.1 Introduction 29
3.2 Project Planning 29
3.2.1 Flowchart of general flow of PSM 30
3.2.2 Gantt Chart 31
3.3 Project Development 32
3.4 Hardware Implementation 34
3.4.1 Modular Production System (MPS) 34
3.4.2 Programmable Logic Controller (PLC) 38
3.4.3 Human Machine Interface (HMI) 39
3.5 Software Implementation 40
3.5.1 CX-Programmer 40
3.5.2 NB-Designer 41
3.6 Conclusion 41

CHAPTER 4 RESULTS AND ANALYSIS 42
4.1 Introduction 42
4.2 Results 42
4.3 Analysis 65
4.3.1 Data analysis for grafcet of original program and improve program 65
4.3.2 Data analysis for the time taken and total to produce production box 71
4.3.3 Data analysis for total detection of actuator to produce production box for original program 73
4.3.4 Data analysis for total detection of actuator to produce production box for improve program 76
4.3.5 Data analysis for average expectacy lifespan of sensor for original program. 79
4.3.6 Data analysis for average expectacy lifespan of sensor for improve program. 80
4.3.7 Data analysis for average expectacy lifespan of actuator for original program. 81
4.3.8 Data analysis for average expectacy lifespan of actuator for improve program. 82
4.4 Summary 83

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS 84
5.1 Conclusion 84
5.2 Future Works 84

REFERENCES 85

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1: Summary of Literature Review		26
Table 3.1: Gantt Chart		31
Table 3.2: Input/Output for MAP-205 subsystem		37
Table 4.1: HMI info address 1		46
Table 4.2: HMI info address 2		47
Table 4.3: HMI info address 3		48
Table 4.4: HMI info address 4		49
Table 4.5: HMI info address 5		50
Table 4.6: HMI info address 6		51
Table 4.7: HMI info address 7		52
Table 4.8: HMI info address 8		53
Table 4.9: HMI info address 9		54
Table 4.10: HMI info address 10		55
Table 4.11: HMI info address 11		56
Table 4.12: HMI info address 12		57
Table 4.13: HMI info address 13		58
Table 4.14: HMI info address 14		59
Table 4.15: HMI info address 15		60
Table 4.16: HMI info address 16		61
Table 4.17: HMI info address 17		62
Table 4.18: HMI info address 18		63
Table 4.19: HMI info address 19		64
Table 4.20: The type of motion and symbol of sensor		65

Table 4.21: The type of motion and symbol of actuator 66

Table 4.22: The comparison of time taken to produce one production box 71

Table 4.23: Total of produce production box 71

Table 4.24: Calculation of production box 72

Table 4.25: Total detection of sensor for original program 74

Table 4.26: Total detection of actuator for original program 75

Table 4.27: Total detection of sensor for improve program 77

Table 4.28: Total detection of actuator for improve program 78

Table 4.29: The average expectacy lifespan cycles of sensor for original program. 79

Table 4.30: The average expectacy lifespan cycles of sensor for improve program 80

Table 4.31:The average expectacy lifespan cycles of actuator for original program 81

Table 4.32:The average expectacy lifespan cycles of actuator for improve program 82

LIST OF FIGURES

FIGURE	TITLE	PAGE

Figure 2.1: The Schematic of Press Machine 13

Figure 2.2: Flow coupling nut production process 261-04E06-00S 13

Figure 2.3**:** Block diagram of the protection system 14

Figure 2.4: Conceptual framework for automation using PLC 15

Figure 2.5: Design project flow for lid folding and sealing stations. 15

Figure 2.6: Sealing process station 16

Figure 2.7: Main control panel for EFI Engine 17

Figure 2.8: HMI display for Oxygen sensor 17

Figure 2.9: Displa al all sensors of EFI Engine 17

Figure 2.10: The PLC program using CX-Programmer 18

Figure 2.11: SCADA design process of a bottling plant 19

Figure 2.12**:** Simplified scheme of an IoT platform the uses an Arduino board 21

Figure 2.13: Intelligent and predictive maintenance system architecture 22

Figure **2.14:** Overall system of caption system for monitoring vibration of motor 23

Figure 2.15: Block diagram 24

Figure 2.16: Genralized block diagram of the wireless sensor 25

Figure 3.1: Flowchart of planning project 30

Figure 3.2: Flowchart operation system of MPS 32

Figure 3.3: Block diagram of MPS operation system 33

Figure **3.4:** Modular Production System **(MPS)** 34

Figure **3.5:** Body Feeder 35

Figure **3.6:** Vertical Revolving Handling Device 35

Figure **3.7:** Horizontal Rotolinear Handling Device 36

Figure **3.8:** Vacuum-held Handling Device 36

Figure **3.9:** Programmable Logic Controller Hardware CPM2A 38

Figure **3.10:** Human Machine Interface 39

Figure 3.11: CX-Programmer software 40

Figure 3.12: NB-Designer software 41

Figure 4.1: Ladder diagram of parallel sequence 45

Figure 4.2: HMI screen display the main menu page 46

Figure 4.3: Actuator selection for display work cycle of sensor 47

Figure 4.4: HMI Screen display work cycle of sensor a0 and a1 48

Figure 4.5: HMI Screen display work cycle of sensor b1 49

Figure 4.6: HMI Screen display work cycle of sensor c1 50

Figure 4.7: HMI Screen display work cycle of sensor e1 and e2 51

Figure 4.8: HMI Screen display work cycle of sensor e0 and e1 52

Figure 4.9: HMI Screen display work cycle of sensor g1 and g0 53

Figure 4.10: HMI Screen display work cycle of sensor h0 and h1 54

Figure 4.11: HMI Screen display work cycle of sensor i0 and i1 55

Figure 4.12: HMI Screen display work cycle of sensor j1 56

Figure 4.13: HMI Screen display work cycle of sensor j0 57

Figure 4.14: HMI Screen display work cycle of sensor k0 and k1 58

Figure 4.15: HMI Screen display work cycle of sensor v 59

Figure 4.16: HMI screen display the selection to view subsystem status 60

Figure 4.17: HMI screen display the status for actuator A+, B+ and C+ 61

Figure 4.18 HMI: screen display the status for actuator E+, E- and F+ 62

Figure 4.19: HMI screen display the status for actuator G+, H+ and I+ 63

Figure 4.20: HMI screen display the status for actuator J+, J-, K+ and V+ 64

Figure 4.21: Improvement grafcet of sequence for automation system 68

Figure 4.22: Original grafcet of sequence for automation system 70

Figure 4.23: The time chart monitor for one production box with original program 71

Figure 4.24: The time chart monitor for one production box with original program 71

Figure 4.25: The time chart monitor for full sequence with original program. 73

Figure 4.26: The time chart monitor with improve program 76

LIST OF ABBREVIATIONS

PLC – Prorammable Logic Controller
IoT – Internet of Thing
MPS – Modular Production System
HMI – Human Machine Interface

CHAPTER 1

INTRODUCTION

1.1 Background

Nowadays, fully automatic machine have been used in industrial manufacturing. Automatic systems are highly regarded in the industrial world because they ensure quality of manufactured goods, minimise processing time and lower human labour costs. The Programmable Logic Controller (PLC) is one of the most common controllers, especially for sequential systems. PLC is characterised as an advanced electronic device with programmable memory to store specific capability guidelines. Then, we need an interface to monitor and control PLC works connecting humans to computer technology called Human Machine Interface (HMI). HMI can be manually or using real-time computer visualisation as control and status visualisation. Modular Production System (MPS) is an automatic machine that used PLC controllers to control each station's sequence. PLC controllers are used to control each station's MPS sequence.

1.2 Problem Statement

The complexity and integration of industrial machinery and equipment has increased significantly as a result of the growing demand for technology in machining and manufacturing. Because of the human ability to detect sensor and actuator harm, maintenance prediction is needed to reduce the impact on quality, cost, and output. Then, as problems arise, maintenance fails to realise. Furthermore, due to the large number of machines present in industries, manually monitoring the condition of machines is time consuming. The technique for predicting an equipment's lifetime is to ensure the equipment is in good condition and does not malfunction during manufacturing.

From the previous program’s production rate for MAP-205, we can see that it is too slow since the assembly process is done one by one and it takes time to wait for the assembly process to be completed. The manufacturing speed will be increased when the parallel sequence technique is used in MAP-205.

Sensor and actuator errors and failures are difficult to predict, if the sensor’s and actuator’s failure or errors while production is running, then it will cause problems. With HMI, it is easy for the technician to monitor and prepare by displaying how frequent of cycle sensor and actuators is running.

1.3 Project Objective

The main objective of this project is:

i) To improved existing conventional sensor and actuator to become a smart sensor and actuator by manipulating a data process by Programmable Logic Controller (PLC).

ii) To improve the production rate of MAP-205 by implement parallel sequence programming technique.

iii) To reduce risk of emergency operation shutting down due to sensor and actuator failure by notification system of the life span of sensor and actuator displaying on Human Machine Interface (HMI).

1.4 Scope of Project

The scope of this project are as follows:

i) Develop prediction system for maintenance of machines in industries by using the Programmable Logic Controller (PLC).

ii) Having automated machine such as modular production system which is composed of smart sensor and actuator will operated by PLC and delivered the data to Human Machine Interface (HMI).

iii) Apply the prediction maintenance in the industry.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter will discuss the research work for the related project, which will include information about the previous project's studies and ideas, as well as the hypothesis that will be tested in this project. The methods used to execute this project will be discussed in details below.

2.2 Research by Journal

2.2.1 Design of Automatic Sleeve for Transfer Nut Clutch using Programmable Logic Controller.

Switching devices produced on press machine operators, replacing the risky and ineffective work of the die machine after the press, where it is done manually by a person, and these instruments can help increase the production capacity expected by the company. The automatic nut transfer coupling system uses a Programmable Logic Controller and is able to save energy by increasing production from 180,000 pieces to 470,000 pieces per month.[1]

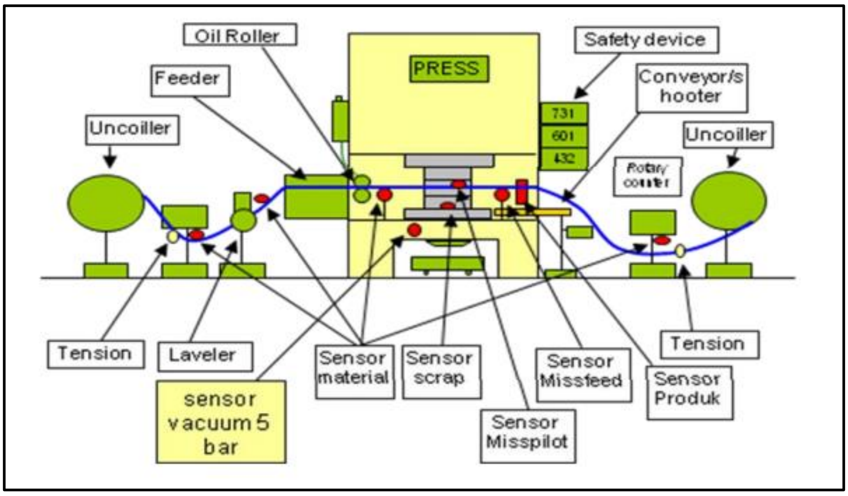


Figure 2.1: The Schematic of Press Machine

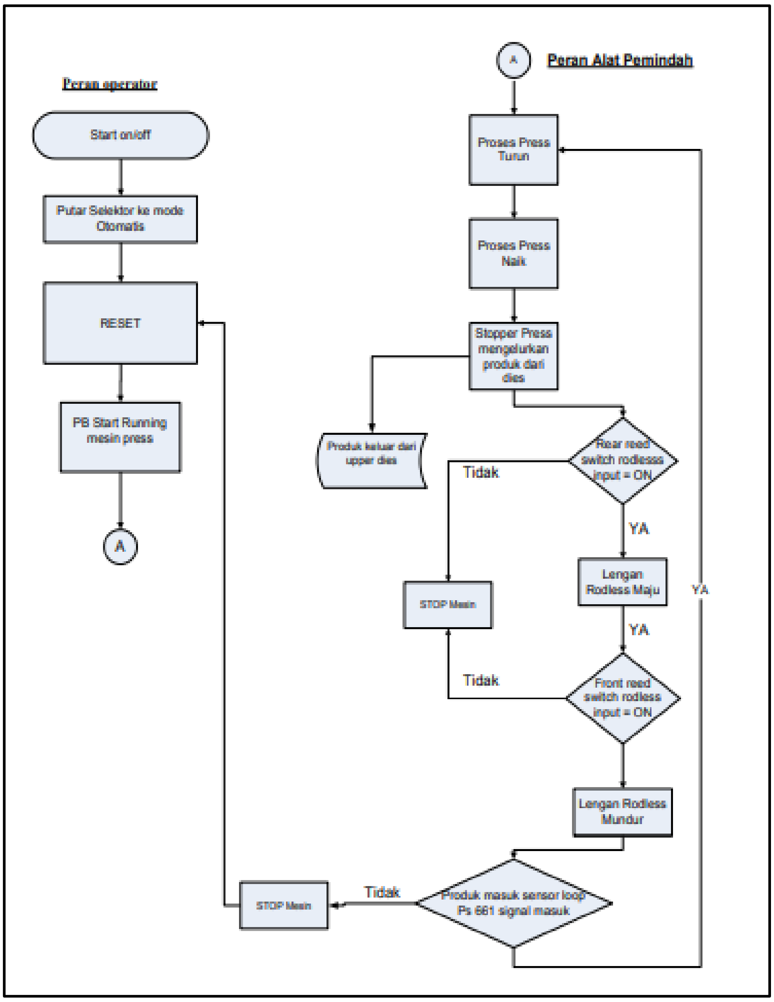


Figure 2.2: Flow coupling nut production process 261-04E06-00S

2.2.2 Fault Identification and Protection of Induction Motor using PLC and SCADA

This article uses a Programmable Logic Controller (PLC) and sensors to prevent induction motor failure by detecting induction motor characteristics such as current, voltage, temperature, speed, and vibration. During motor operation, all these features are continuously monitored with the help of SCADA. If an error occurs, one or more parameters will change, allowing us to take the necessary steps and avoid damaging the induction motor.[2]

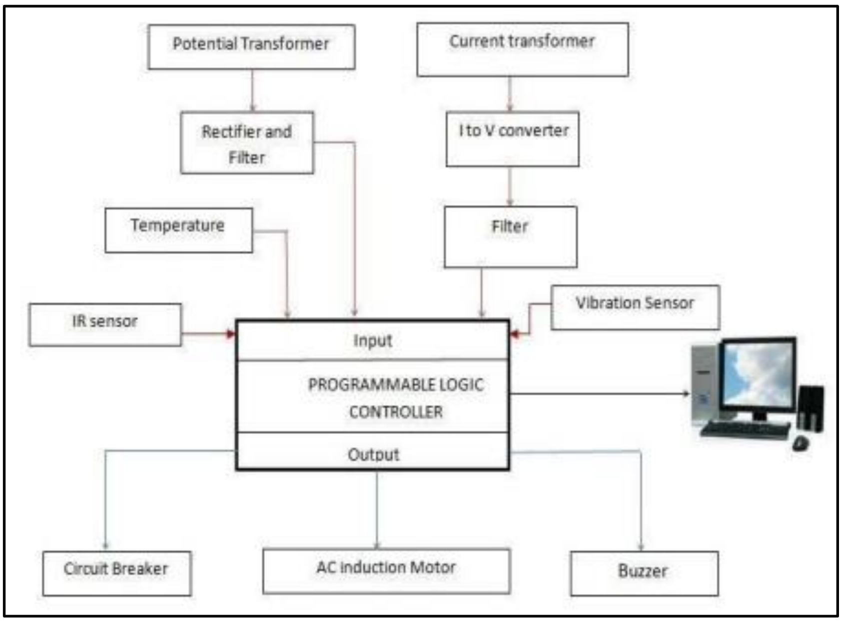


Figure 2.3: Block diagram of the protection system

2.2.3 Automation of Packaging and Material Handling Using Programmable Logic Controller.

This This article presents automated packaging and material handling using a programmable logic controller. The goal is to automate the process of loading items into the box, identify good and bad items based on weight, and close the box using packing tape. Besides that, this research aims to replace industrial manual systems and compare the time and manpower requirements for current and proposed automated systems. The system is mechanised using a Mitsubishi FX series programmable logic controller. The system receives input from proximity and load sensor. The output is provided by the motors, pneumatics, and solenoids.[3]

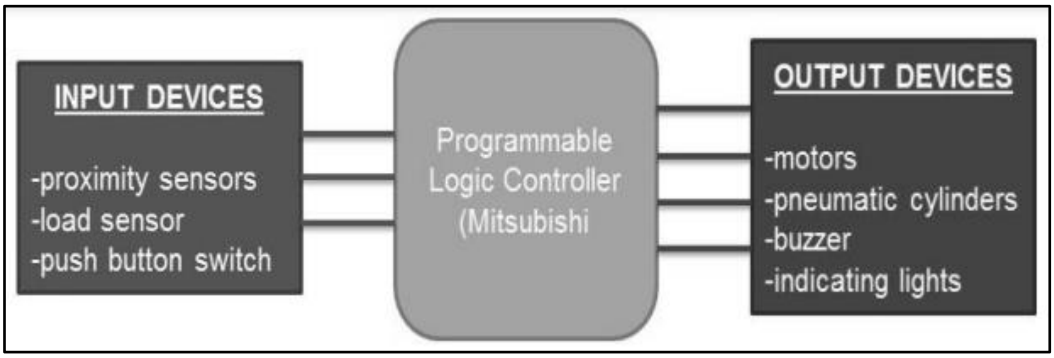


Figure 2.4: Conceptual framework for automation using PLC

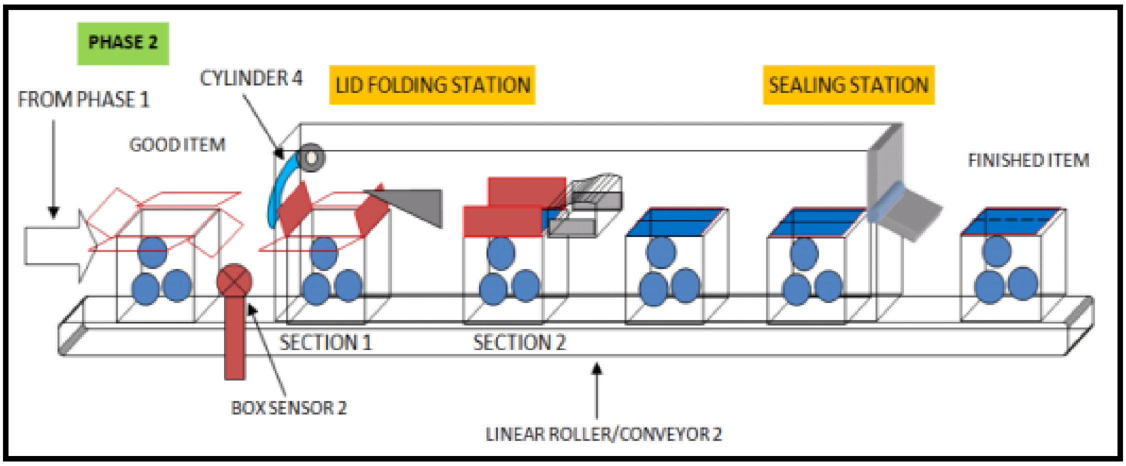


Figure 2.5: Design project flow for lid folding and sealing stations.

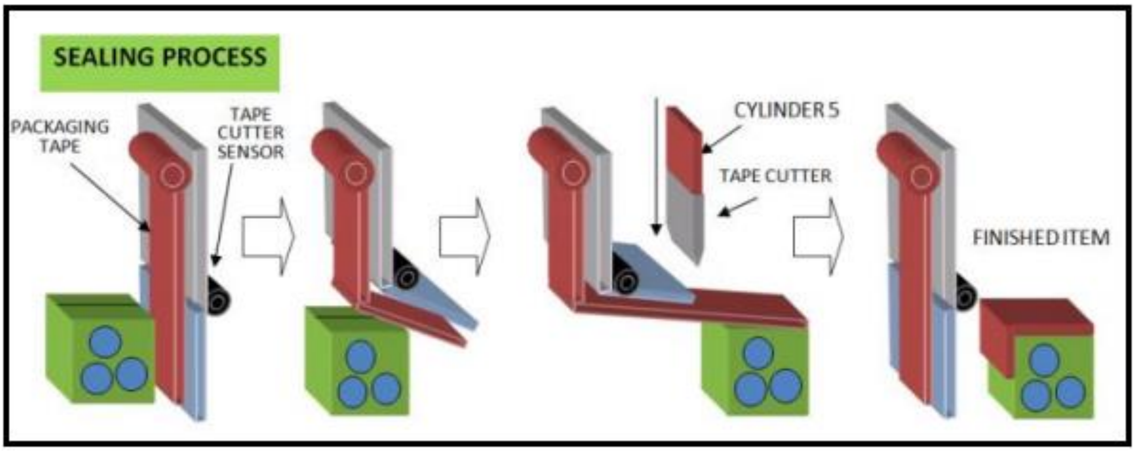


Figure 2.6: Sealing process station

Figure 2.5 shows the second phase of the design project. It is made up of two stations: the lid folding station and the sealing station. The detailed automated procedure for the packing station is depicted in Figure 2.6. To seal the boxes in a single linear direction, a 1-inch wide packing tape is utilised. A proximity sensor detects the length of the packaging tape that will be used to seal the boxes.

2.2.4 Designing Human Machine Interface for Vehicle's EFI Engine using Siemen's PLC and SCADA System

This article proposes a technique to monitor and regulate the Electronic Fuel Injection (EFI) diesel engine sensors. Siemens S7-300 modular Programmable Logic Controller (PLC) and WinCC Flexible Human Machine Interface (HMI) is used to constructing a testbed utilising the SCADA system. PLC observes digital and analogue input data from EFI sensors in the suggested architecture, generating desired regulated output signals based on the PLC's ladder logic programme. Contemporary human operator (driver) may monitor process variables on a single HMI screen and also control outputs with one touch.[4]

Figure 2.7: Main control panel for EFI Engine

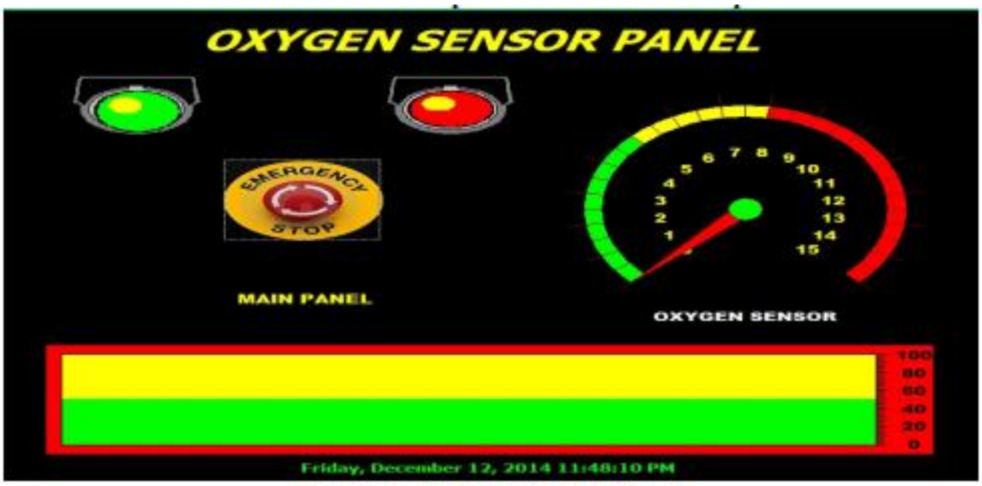


Figure 2.8: HMI display for Oxygen sensor

Figure 2.9: Displa al all sensors of EFI Engine

2.2.5 Controlling Process of a Bottling Plant using PLC and SCADA

This article presents the basic stages of bottling plant operation. The goal is to control filling and capping section simultaneously. At first, an empty bottle set is run using a conveyer to fill the section. After the operation, the filled bottles are transported to the capping section. After a successful capping operation, the sealed bottles end towards the exit and a new set of empty bottle arrives, so the process repeats. This document provides the procedure of filling and capping several bottles at one instant. This strategy made it more adaptable and timesaving. Using Programmable Logic Controllers (PLC), filling and capping activities are very user-friendly, cost-effective, and easy to control. PLC automation keeps the entire process under control. Using a display system, SCADA to monitors the whole process.[5]

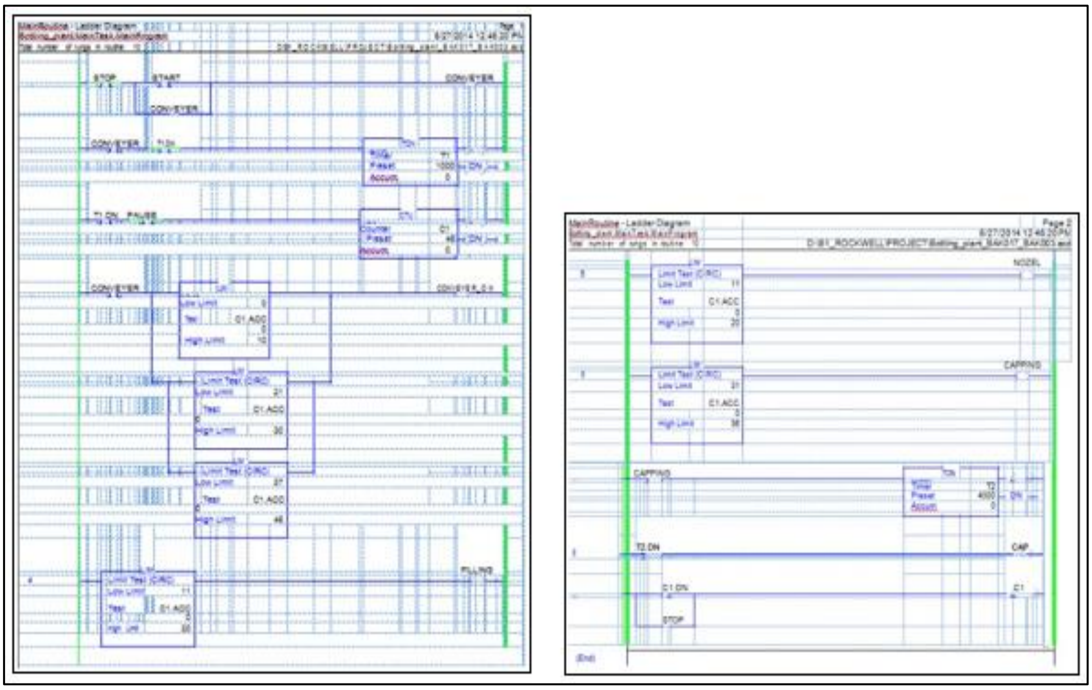
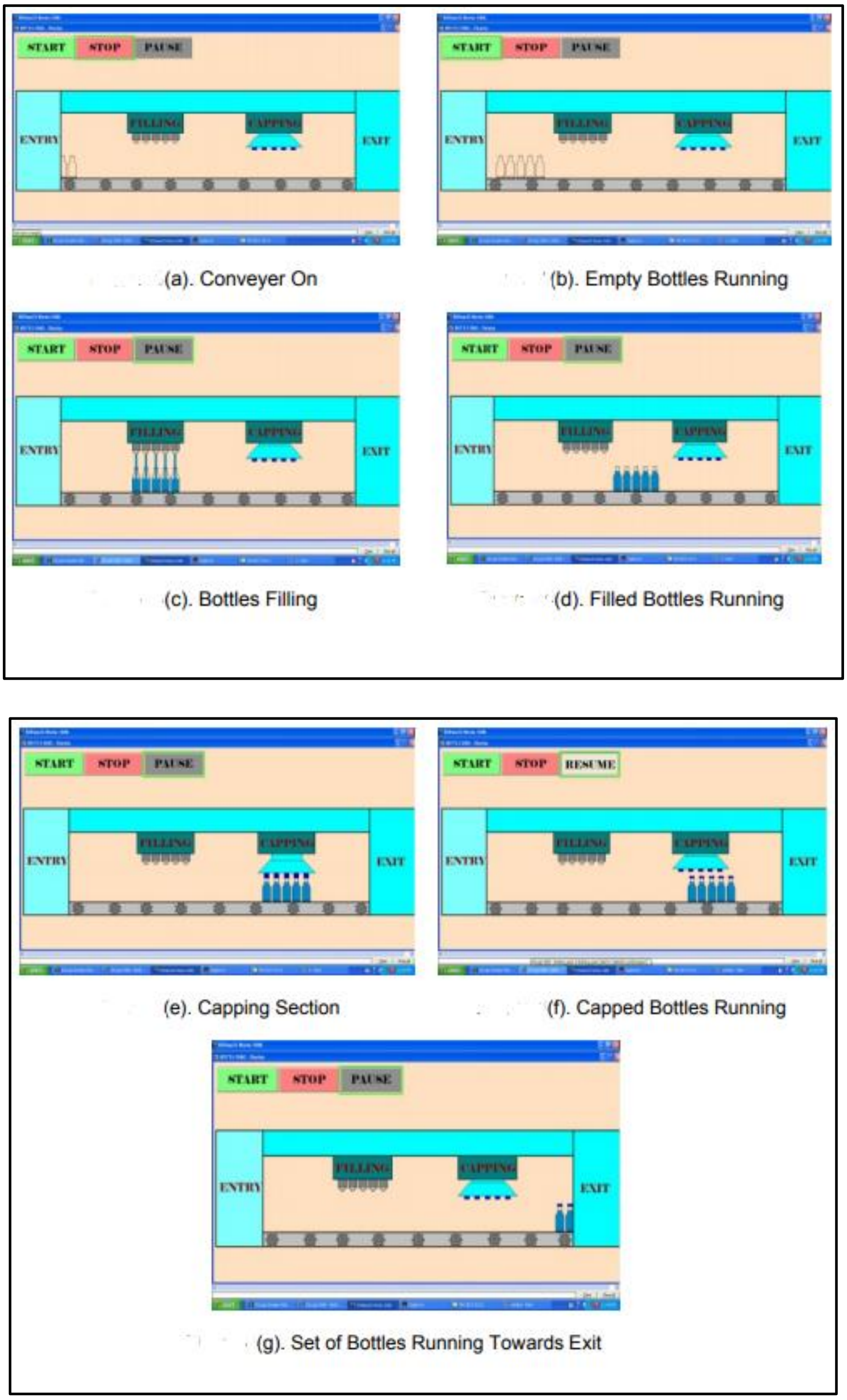


Figure 2.10: The PLC program using CX-Programmer



(a). Conveyer On

(b). Empty Bottles Running

(c). Bottles Filling

(d). Filled Bottles Running

(e). Capping Section

(f). Capped Bottles Running

(g). Set of Bottles Running Towards Exit

Figure 2.11: SCADA design process of a bottling plant

2.2.6 Application of IoT Concept on Predictive Maintenance of Industrial Equipment

This article proposes a new method for online monitoring and predictive maintenance. This strategy is based on software interface tools and requires procedures for converting various industrial equipment into network protocols. Furthermore, the project produces two linked method aspects in the production process. First is the process monitoring for regular quality control, and the other is condition monitoring to reduce unplanned downtimes. The hardware used for testing in this project is automatic polishing and sanding equipment for high-gloss devices. Automatic machines provide the possibility to acquire high-quality items in a short period with minimal human participation. The machines in this project have three sorts of movements: conveyor table rotation with a fixed wooden panel, brush head movement in a transversal position route into the wood panel, and continual brush rotating. Then the three motor drives will be controlled by a PLC using the RS-485 / Modbus RTU protocol. Link hardware and software together by using Arduino. The data will be sent to the IoT platform. As a result, this project aims to create a monitoring system that can track the state of motors, their speed, and the temperature surrounding the first motor. The speed, measured temperature, and operational condition will all be recorded and sent to an IoT platform for maintenance prediction. Aside from that, the proposed predictive maintenance system's major feature is sending notifications about equipment use or condition through email or other messaging services.[6]

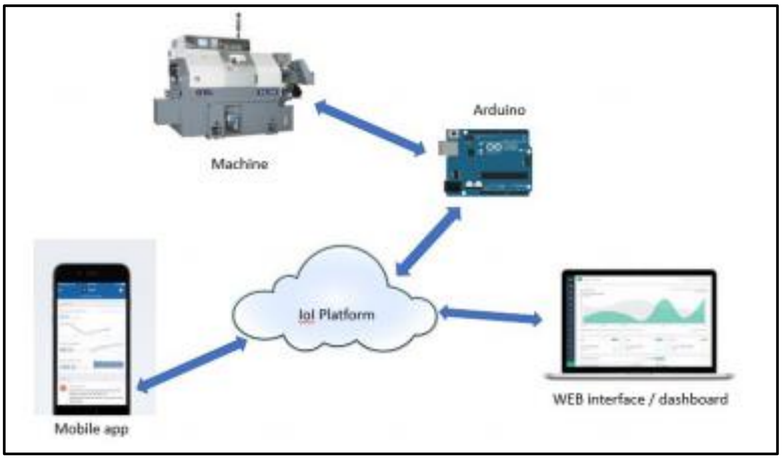


Figure 2.12**:** Simplified scheme of an IoT platform the uses an Arduino board

2.2.7 Intelligent and Predictive Maintenance in Manufacturing System

In this article, the author proposes the best method for managing component or equipment failures during manufacturing processes by predicting maintenance in the manufacturing system. This project focuses on designing a dynamic monitoring module to monitor the situation and activate alerts when a possible disruption are discovered or as expected. The dynamic monitoring module divides into two sections, visualisation and early detection failures. The visualisation component facilitates the comparison of key performance indicators (KPI) with predicted operating limitations.[7]

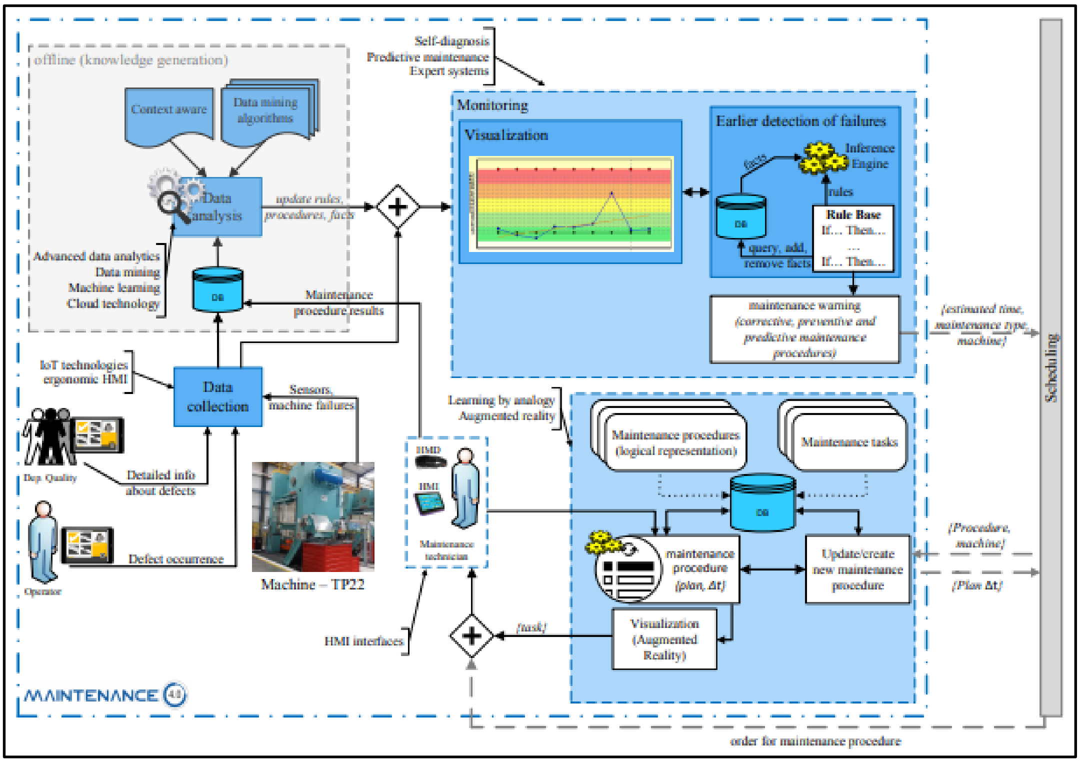


Figure 2.13: Intelligent and predictive maintenance system architecture

Figure 2.13 above shown the intelligent and predictive maintenance system architecture. The developed 4.0 system architecture preservation project integrates all the mentioned modules to create a functional system that enables intelligent maintenance and forecasting to be performed. The general system uses various technologies such as IoT technology and machine learning. In addition, early fault component identification controls engine facts and indications that trigger maintenance notices when problems are identified early. Data will be provided to a scheduling tool that will plan the intervention as suggested by current production conditions, identifying the availability of management resources for the implementation intervention. Next, smart decision support modules can build or modify new maintenance procedures when discrepancies found may not have a specified maintenance process. The maintenance professional will be provided with the advanced Human Machine Interface (HMI) when performance maintenance is necessary for the maintenance schedule.

2.2.8 Initiating Predictive Maintenance for a Conveyor Motor in a Bottling Plant using Industry 4.0 Concepts

This article proposes developing an experimental technique for integrating Industry 4.0 into a small bottling plant by identifying early problems or threats in conveyor motors and developing a predictive maintenance schedule. Vibration speed data is monitored by vibration sensors installed on the motor, and an efficient predictive maintenance plan is developed using sophisticated programming features of a Siemens S7-1200 PLC managing the bottling plant. SCADA and graphical user interface (GUI) connects with the PLC, displays maintenance schedules in real-time, and enables new flexible maintenance rules as needed. This article also proposes a decentralised monitoring system that monitors vibration speed condition on a cloud-based report accessible via the internet. It sends quick email alerts to the designated supervisor for each maintenance plan issued.[8]

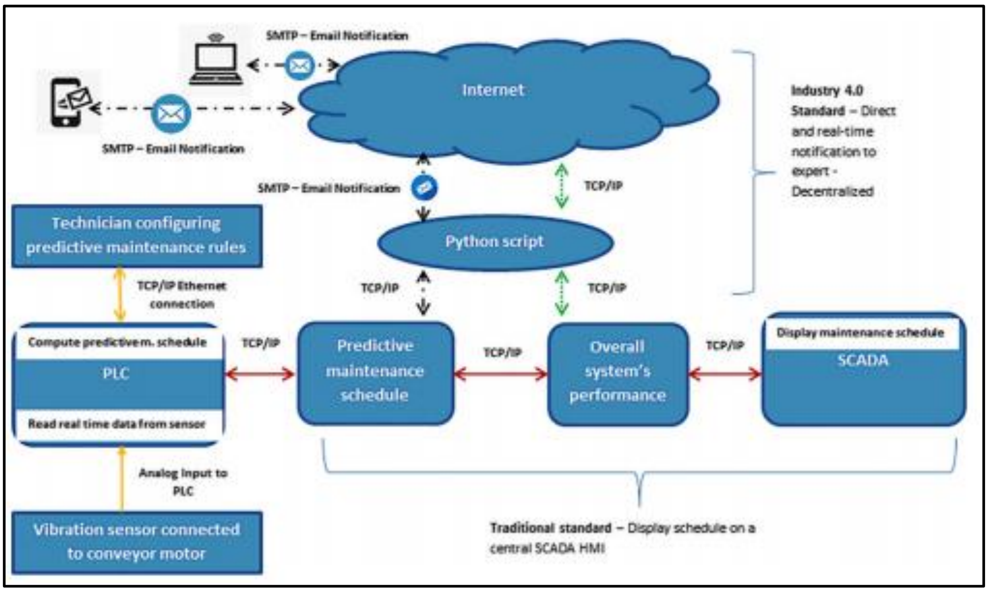


Figure 2.14: Overall system of caption system for monitoring vibration of motor

2.2.9 Smart Garden Monitoring System Using IoT

The smart garden comprises a NodeMCU hub to which various kinds of sensors like moisture sensors, humidity sensors, temperature sensors and ultrasonic sensors are connected. The ultrasonic sensor is connected to a water tank, displaying the water level in the tank. Other sensors are linked to their proper placements, and these sensors communicate data to the NodeMCU, which has built-in Wi-Fi. Firebase is an internet-based database in which the sensor's real-time measurements are updated every second. The android studio programme is used to create the app. The connection between the application and firebase will be established inside the programme. As a result, the parameters may be monitored from any location. The kind of soil determines how much water is needed in the garden. As a result, the sensor values are predefined inside the programme for automation reasons. A switch in the programme will automate the operation whenever the user needs to water the garden. This helps in the overall maintenance of the garden.[9]

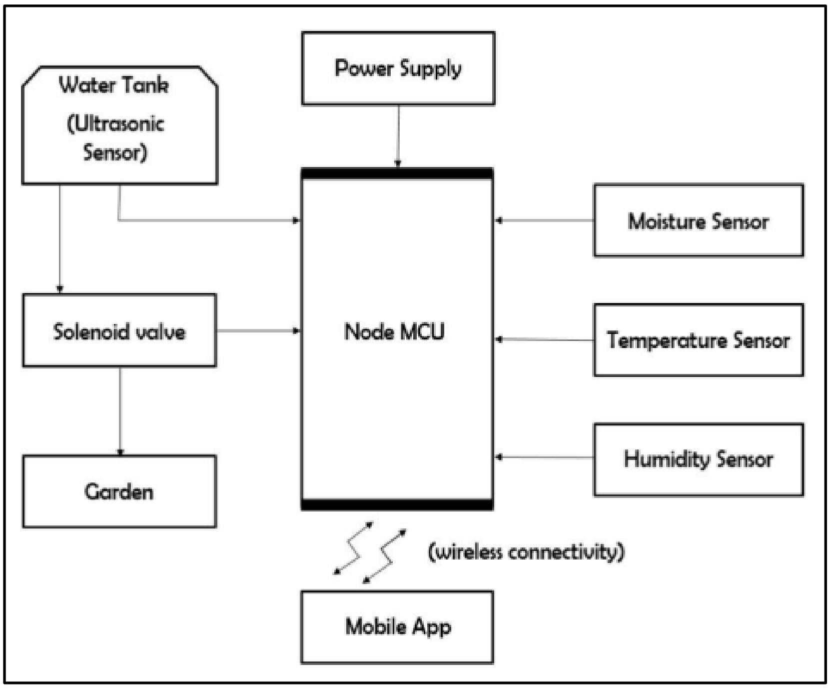


Figure 2.15: Block diagram

2.2.10 Smart Management of Technologies: Predictive Maintenance of Industrial Equipment using Wireless Sensor Networks

This article discusses industrial equipment servicing methods, focusing on predictionmaintenance. The techniques of maintaining industrial equipment are discussed in this study, with an emphasize on the predictive maintenance, known as real maintenance. The case for using wireless data gathering and processing devices is outlined. Depending on the sector of application, the concepts of establishing wireless sensor networks and the data transmission protocols used to gather statistical information on the status of the pieces of industrial equipment are examined. The study's goal is to prove that deploying wireless sensor networks as technical diagnostic tools is feasible from both an economic and technological perspective. The suggested notion of a predictive maintenance system is the outcome. The article backs up the concept of predictive repair optimization utilising wireless sensor networks. This strategy is predicated on lowering equipment maintenance expenses as much as possible.[10]

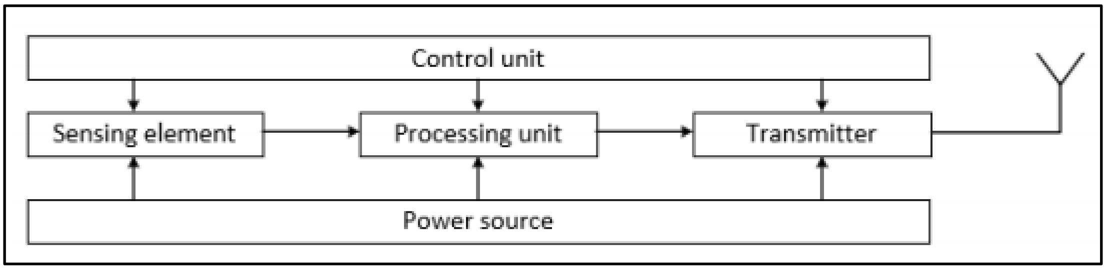


Figure 2.16: Genralized block diagram of the wireless sensor

2.3 Table of Summary

Table 2.1: Summary of Literature Review

Title	Objective	Description	Author	Year
Design of Automatic Sleeve for Transfer Nut Clutch using Programmable Logic Controller	To control the pneumatic system by sing Programmable Logic Controller.	These switching devices, which are installed on press machine operators, are intended to replace dangerous and ineffective tasks that would otherwise be performed manually by people.	Syahril Ardi, Agus Ponco , Adhari Faried Ardin	2012
Fault Identification and Protection of Induction Motor using PLC and SCADA	Monitoring the operation of the AC induction motor conditions by using the protection system.	It prevents unexpected motor failure and protects the entire industrial process from abruptly shutting down, which might harm anybody working nearby.	Sujith John Mathew, B.Hemalatha	2014
Automation of Packaging and Material Handling Using Programmable Logic Controller	To design an automated packaging and material handling system.	To turn the system into an automated process, a programmable logic controller is utilised. Sensors and switches are utilised as input components to signal the condition matching to the PLC project design's hardware flow diagram. The system was interfaced by the programmable logic controller, which also provided the ladder diagram for the design. Output components such as motors and pneumatic cylinders are utilised to indicate the system's main objective.	Joanna Marie M. Baroro, Melchizedek I. Alipio, Michael Lawrence T. Huang, Teodoro M. Ricamara, Angelo A. Beltran Jr.	2014
Designing Human Machine Interface for Vehicle's EFI Engine using Siemen's PLC and SCADA System	To monitor and control the sensors of Electronic Fuel Injection diesel engine.	The SCADA system is used to build a test bed utilising the Siemens S7-300 modular Programmable Logic Controller (PLC) and WinCC Flexible Human Machine Interface (HMI). In the proposed architecture, the PLC monitors digital and analogue input signals from EFI sensors	Umair Younas, Sajjad Durrani	2015

		before creating required regulated output signals using the PLC's ladder diagram software. On a single HMI screen, a human operator may monitor process changes and control outputs.		
Controlling Process of a Bottling Plant using PLC and SCADA	To control the filling and capping section simultaneously .	Controlling Industrial Machinery and Processes with a control system while minimising human intervention. When comparing a work performed by people with one performed by Automation, the physical component of the activity is replaced by a Machine, while the human mental skills are replaced by Automation.	Kunal Chakraborty , Indranil Roy , Palash De	2015
Application of IoT concept on predictive maintenance of industrial equipment	To monitor online and predict maintenance of industrial equipment.	The manufacturing is connected by two features. Which is process monitoring for constant quality assurance and condition monitoring in order to prevent unplanned downtimes.	Radu Constantin Parpala and Ro bert Iacob	2017
Intelligent and Predictive Maintenance in Manufacturing Systems.	To create a system architecture for an smart predictive maintenance system	Create an overall system architecture for an intelligent and innovative maintenance system and develop a monitoring module that permits monitoring of an asset's status and triggers alarms when possible disturbance is identified or forecasted.	Prof. Dr. Paulo Jorge Pinto Leitão, Prof. Dr. José Fernando Lopes Barbosa	2018
Initiating predictive maintenance for a conveyor motor in a bottling plant using industry 4.0 concepts	To detect damage or initial threats to the conveyor motor and create a predictable maintenance schedule.	To implant in various important equipment around the plant in order to collect data that is then analysed and interpreted With the development of Industry 4.0, more data and value will be added to system maintenance predictions, making this work more efficient.	Kahiomba Sonia Kiangala, Zenghui Wang	2018
Smart Garden Monitoring	To sustain the nature of the plants by	Android software is used to develop mobile applications that monitor garden conditions	T.Thamaraima nalan, S.P.Vivekk ,	2018

System Using IOT	constantly monitoring the conditions, resulting in a longer life for both plants and humans.	and automate watering. NodeMCU is used to connect several sensors that gather soil characteristics and send data to a firebase through Wi-Fi.	G.Satheeshkumar, P.Saravanan	
Smart management of technologies: predictive maintenance of industrial equipment using wireless sensor networks	Methods of servicing industrial equipment that focus predictive maintenance.	The predictive repair optimization methodology employing wireless sensor networks. This research is built on reducing the expenses of equipment maintenance. The suggested concept of a predictive maintenance system based on sensor networks enables real-time study of equipment status. To use smart technology management in order to ensure the stability of company operations.	Andrey I. Vlasov, Pavel V. Grigoriev, Aleksey I. Krivoshein, Vadim A. Shakhnov, Sergey S. Filin, Vladimir S. Migalin	2019

2.4 Summary

According to data collected from the previous system development, the majority of them use a Programmable Logic Controller as their controller. This is because that this controller is simple to programme and easy to understand the programming language.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This section will explain and examine the flow and a few techniques and strategies used during the final year project. It is more about the project's layout and thought to ensure that the project's development is made easy and follows appropriately. A portion of the methodology is used to demonstrate the progress made in completing this type of project.

3.2 Project Planning

The flowchart shown in Figure 3.1 as a project process flow. To achieve development, this project must be carried out flow by flow. The project will then be made easier by following the predefined procedure. The planning project assembles the flow from the beginning to the end of the project.

3.2.1 Flowchart of general flow of PSM

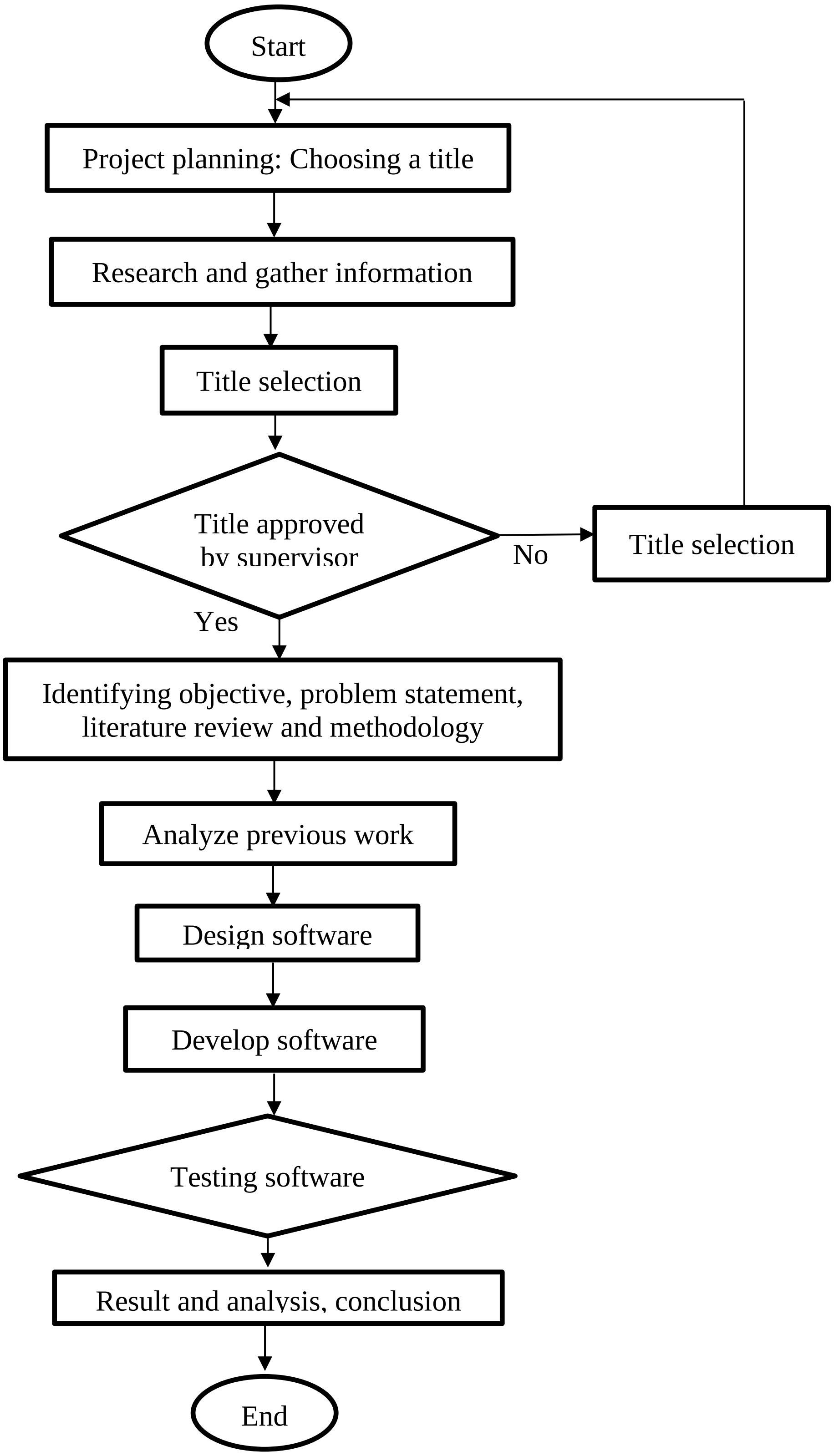


Figure 3.1: Flowchart of planning project

3.2.2 Gantt Chart

Table 3.1: Gantt Chart

PROJECT PLANNING

Listdown the main activity for the project proposal. State the time frame needed for each activity.

	2021																																					
Project Activity	**MAC**			**APR**				**MAY**				**JUN**					**JULY**		**J**	**A**	**OCT**				**NOV**				**DEC**				**JAN**					**J**
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**X**	**X**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
Literature review	**BP Briefing**							**Mid Term Break**								**Study Week**	**Final Semester Examination**		**Semester**									**Mid Term Break**								**Study Week**	**Final Semester Examination**	
Project Planning																																						
Proposal Preparation																																						
Study of Modular Production System																																						
Study of Programmable Logic Controller																																						
Study of Human Machine Interface																																						
Construct process simulation																																						
Simulation the PLC Ladder Diagram																																						
Simulation NB Designer																																						
Perform Analysis																																						
Report preparation																																						

3.3 Project Development

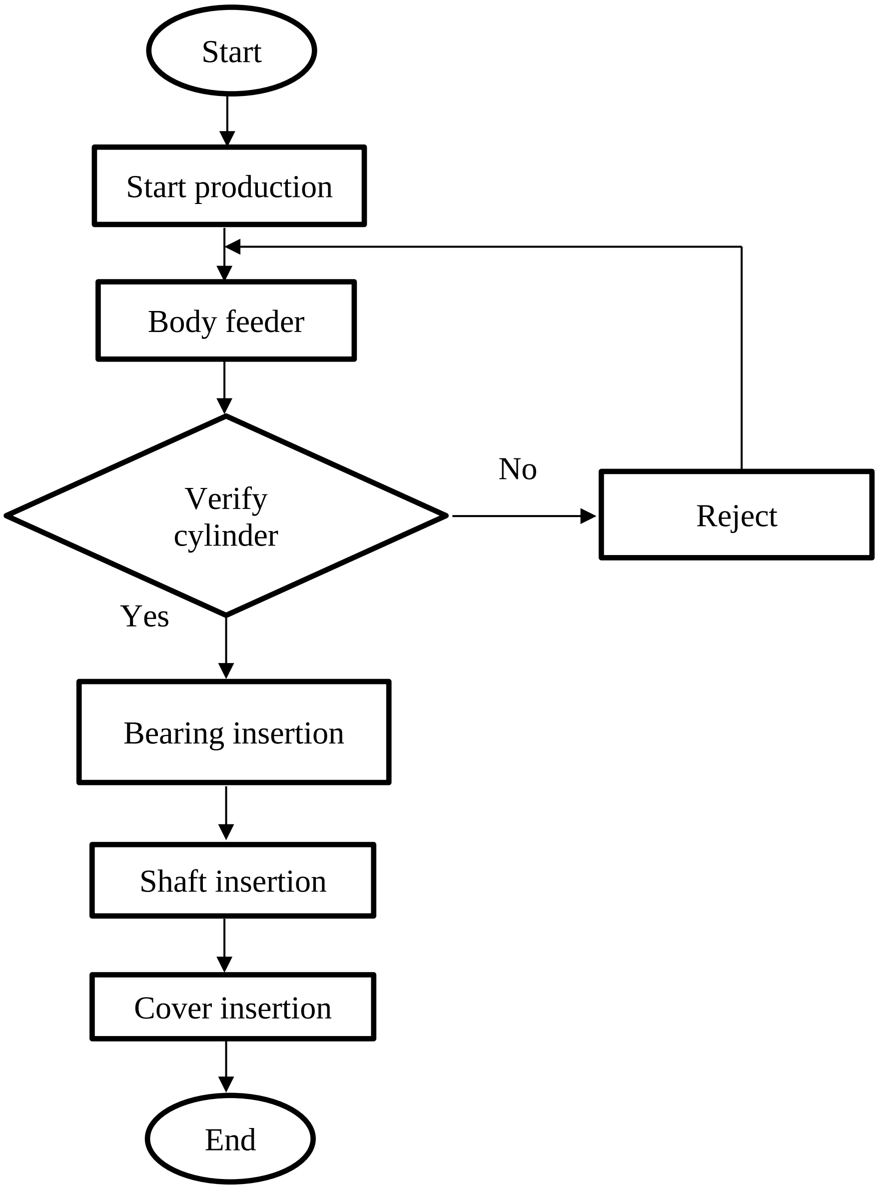


Figure 3.2: Flowchart operation system of MPS

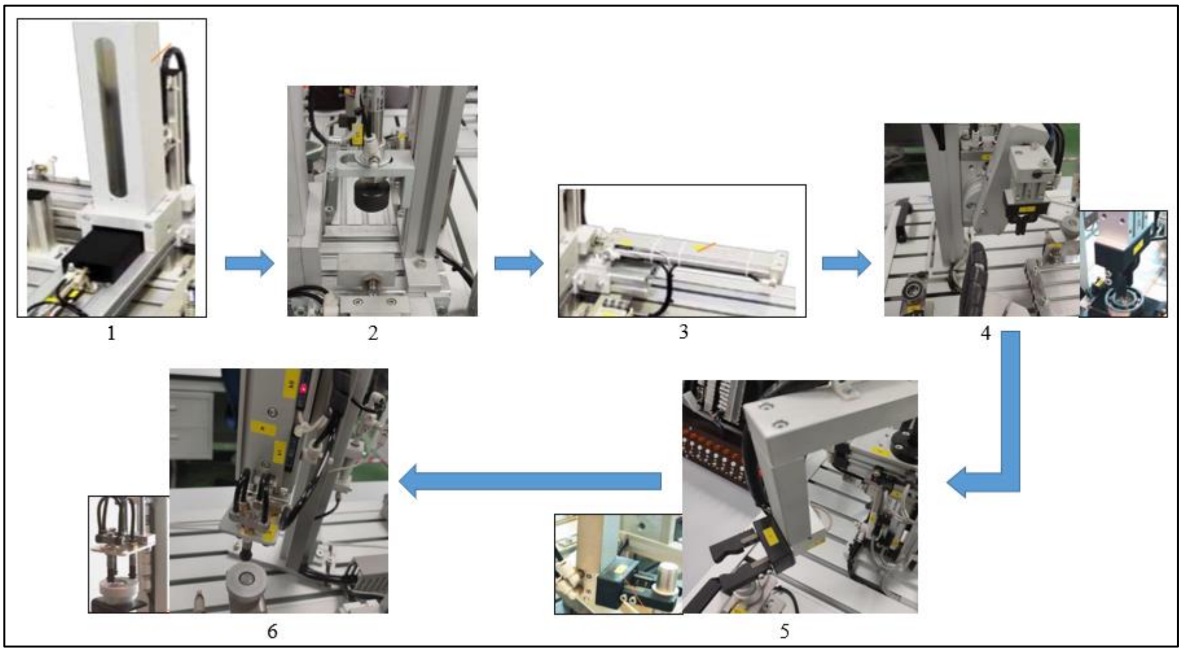


Figure 3.3: Block diagram of MPS operation system

Figure 3.3 above show the step of MPS operation system. The steps been describe below:

Step 1: Body feeder

Step 2: Verify cylinder

Step 3: Body transfer

Step 4: Rotary handling bearing

Step 5: Shaft insertion

Step 6: Cover insertion

3.4 Hardware Implementation

3.4.1 Modular Production System (MPS)

Figure 3.4: Modular Production System (MPS)

An MPS is a learning system that can be used to predict the complexity of industrial production equipment. This project uses MPS as a plant for automatic system that includes a sensor and actuator. This project will use the MAP-205. There are a number of subsystems in MAP-205, which include body feeder, rotary machine for bearing insertion, hand handling for shaft insertion, and vacuum for cover insertion. The MPS used the PLC to control the sensor and actuator for operating and sequencing, which can be applied to the design of more complex systems. In the other case, if the sensor and actuator in MPS fail to operate within the sequence time, the PLC will detect the failures.

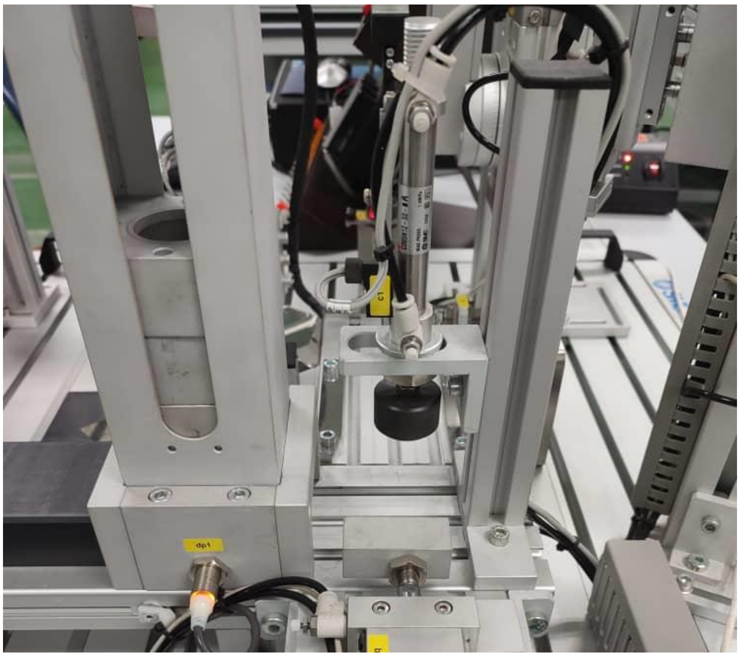


Figure 3.5: Body Feeder

Figure 3.6: Vertical Revolving Handling Device

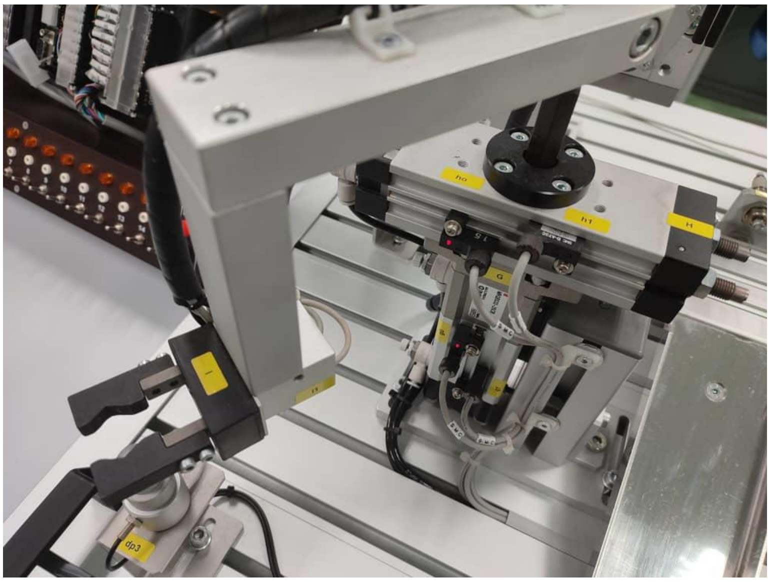


Figure 3.7: Horizontal Rotolinear Handling Device

Figure 3.8: Vacuum-held Handling Device

Table 3.2: Input/Output for MAP-205 subsystem

<table>
<tr><td rowspan="2">MAP-205</td><td>Subsystem</td><td>Input/Output</td></tr>
<tr><td>• Body Feeder
• Vertical Revolving Handling
• Horizontal Rotolinear Handling
• Vacuum-held Handling</td><td>Digital 24/15</td></tr>
</table>

3.4.2 Programmable Logic Controller (PLC)

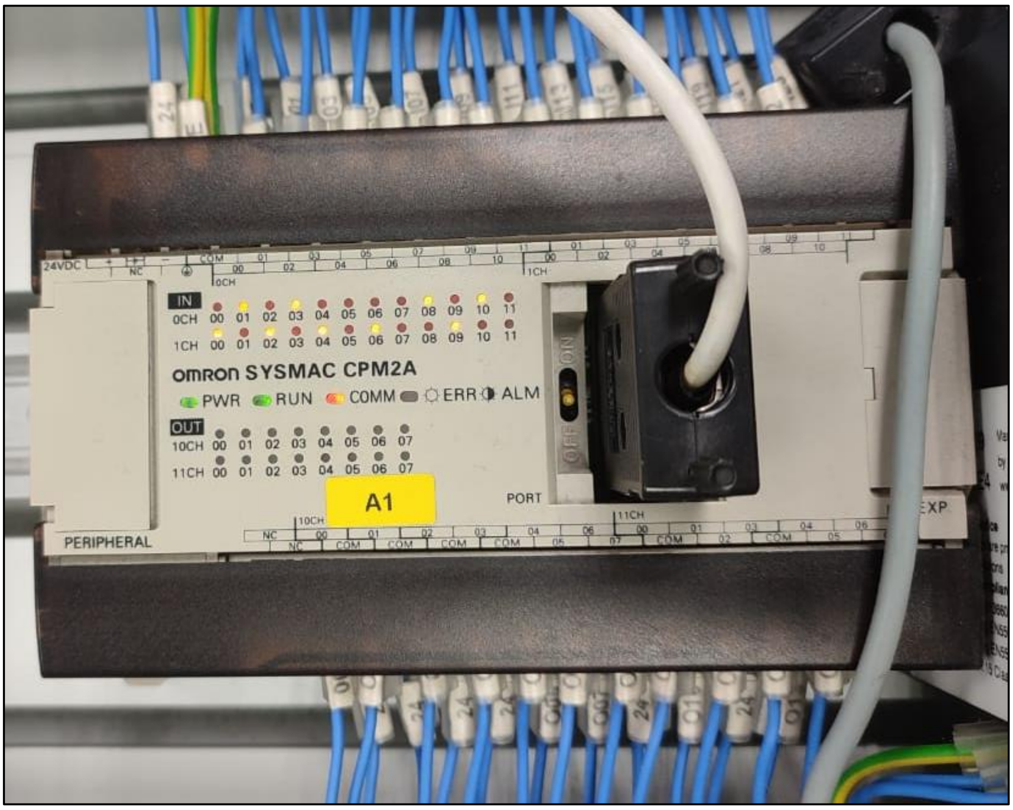


Figure 3.9: Programmable Logic Controller Hardware CPM2A

PLCs are fundamentally computers, designed instruction for control processes and machines. PLC internal instructions control the machine with programmable memory. The type of PLC used in this project is PLC Omron SYSMAC CPM2A. Besides that, this project PLC was also used as a main component for the control of MPS sequences and for sensor and actuator failure control. In this project, a PLC has been built as a controller to detect failures, based on the ladder diagram. Furthermore, functions based on the ladder diagram were moved and implemented in order to ensure the programme would perform properly. The PLC detects sensor and actuator failure and counts cycles.

3.4.3 Human Machine Interface (HMI)

Figure 3.10: Human Machine Interface

HMI is a programme that displays information on the process state, and then receives and applies operator control instructions. For this project, HMI Omron NB5W-TW00B was used. In order to monitor and display the conditions of smart sensors and actuators, HMI have been used. Operator actions are being influenced by the HMI architecture. The HMI will show the sensor and actuator lifespan, as well as the signal lamp indicator on the screen. The sensor and actuator have different lifespans of cycles. Lifespan is based on the components. After that, set the target based on lifespan. For this project, if the cycles of lifespan go over the target, the screen will display a red lamp. If it reaches the target, the HMI will light up yellow on the screen. HMI will display green lamp if the cycle is stable and did not exceed the target. Other than that, the HMI also display dates and times as well as saving all data for references.

3.5 Software Implementation

3.5.1 CX-Programmer

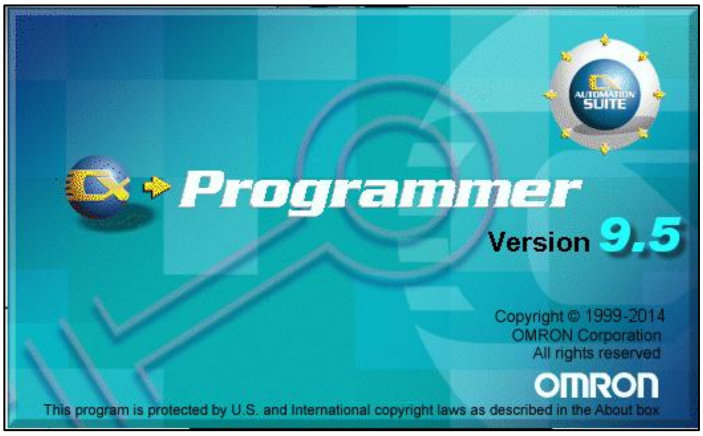


Figure 3.11: CX-Programmer software

CX-Programmer is a PLC programming tool that employs the ladder diagram methods to build programmes for OMRON CS/CJ/CP-series PLCs, CV-series PLCs, and C-series PLCs. The ladder diagram utilizes graphic symbol like relay schematic circuit diagrams. The input and output are entirely identified by their addresses, with the symbol used depending on the PLC manufacturer. This software, on the other hand, may perform PLC configuration and operation functions such as debugging programmes, displaying address and values, organizing and monitoring, and remote programming over the network.

3.5.2 NB-Designer

Figure 3.12: NB-Designer software

The NB–Designer HMI software is used to create or edit screens that appear on the HMI series screen. Aside from that, the NB–Designer has all of the features and functionality needed to quickly create an automatic operator screen. The design that we construct will be display on HMI screen.

3.6 Conclusion

In conclusion, this chapter discusses the methods and process to be used in develop predictive maintenance for industrial monitoring and control system in industrial machine. The project planning is very important in developing a system that is well planned and organized in order to complete it in a given time.

CHAPTER 4

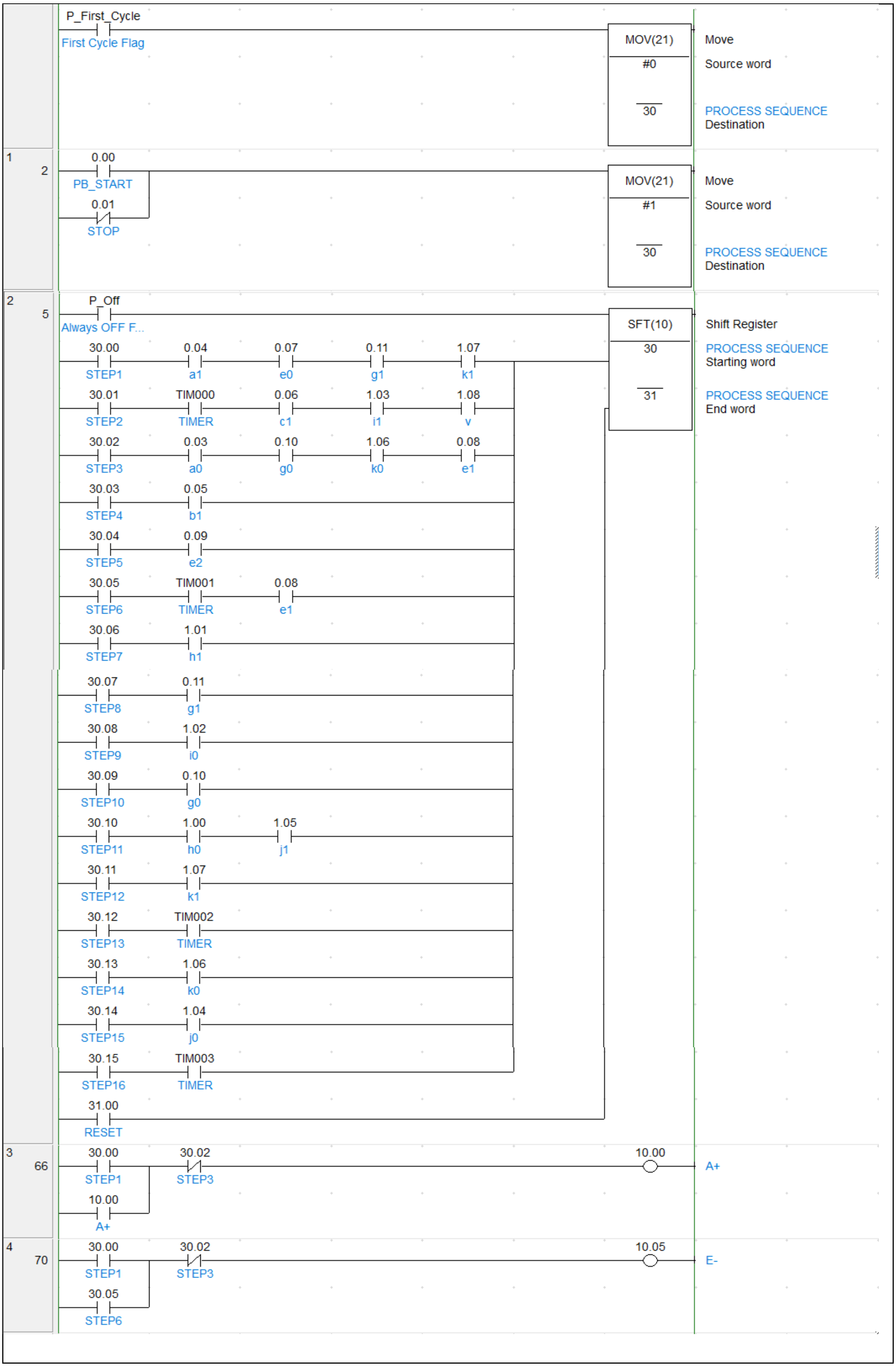
RESULTS AND ANALYSIS

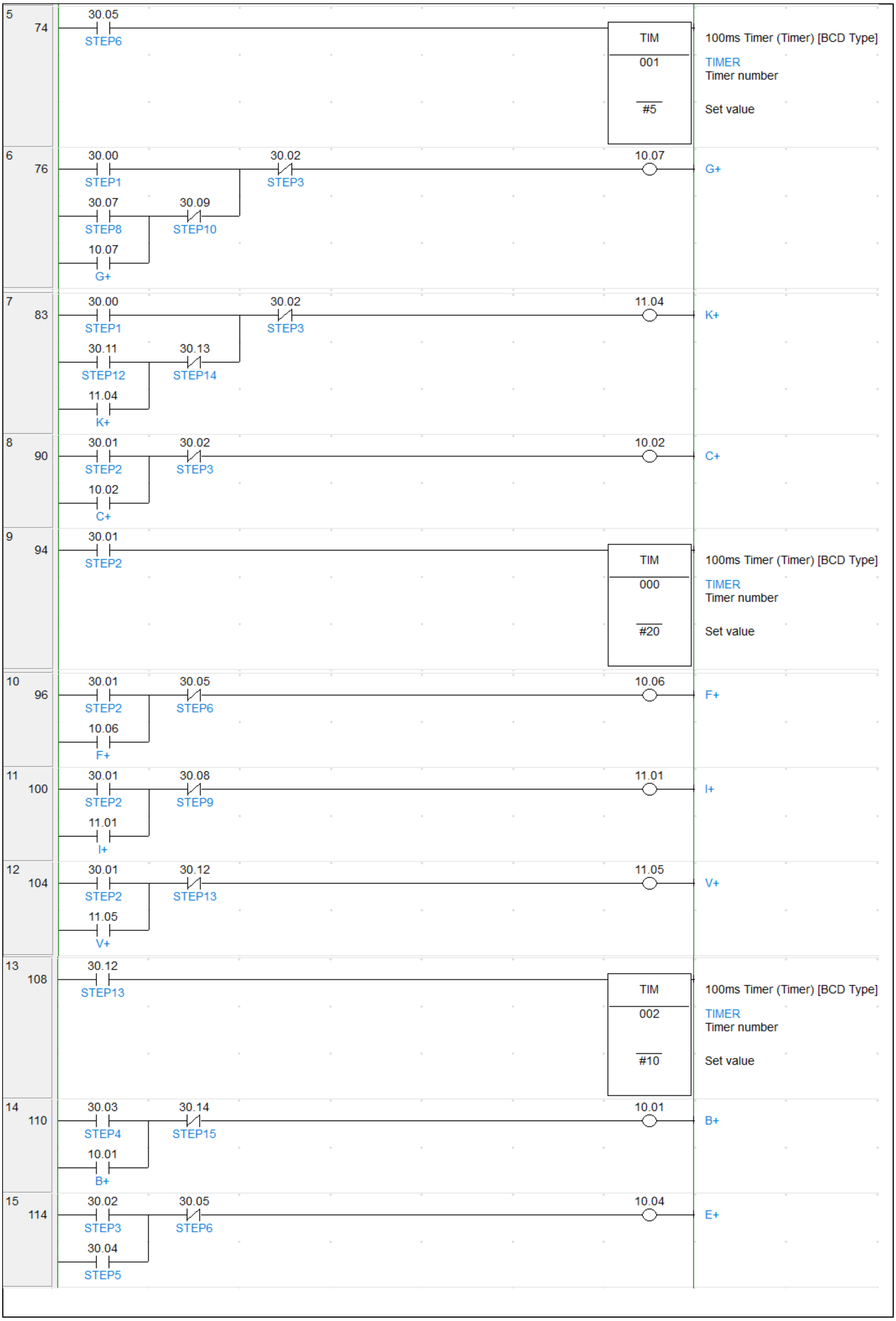
4.1 Introduction

This chapter presents the results and analysis project that has been conducted. The implementation of the software and hardware are presented in detail to verify its functionality in achieving the desired objective. This system should work successfully and any problems that arise will be discussed.

4.2 Results

After the ladder diagram instruction are transfer to the Programmable Logic Circuit and then connect to the Human Machine Interface, the result can be observed. All he software and hardware part need to run according to the purposed systems.


P_First_Cycle
First Cycle Flag
MOV(21)
#0
30
Move
Source word
PROCESS SEQUENCE
Destination
1
2
0.00
PB_START
0.01
STOP
MOV(21)
#1
30
Move
Source word
PROCESS SEQUENCE
Destination
2
5
P_Off
Always OFF F...
SFT(10)
30
31
Shift Register
PROCESS SEQUENCE
Starting word
PROCESS SEQUENCE
End word
30.00 STEP1
0.04 a1
0.07 e0
0.11 g1
1.07 k1
30.01 STEP2
TIM000 TIMER
0.06 c1
1.03 i1
1.08 v
30.02 STEP3
0.03 a0
0.10 g0
1.06 k0
0.08 e1
30.03 STEP4
0.05 b1
30.04 STEP5
0.09 e2
30.05 STEP6
TIM001 TIMER
0.08 e1
30.06 STEP7
1.01 h1
30.07 STEP8
0.11 g1
30.08 STEP9
1.02 i0
30.09 STEP10
0.10 g0
30.10 STEP11
1.00 h0
1.05 j1
30.11 STEP12
1.07 k1
30.12 STEP13
TIM002 TIMER
30.13 STEP14
1.06 k0
30.14 STEP15
1.04 j0
30.15 STEP16
TIM003 TIMER
31.00 RESET
3
66
30.00 STEP1
30.02 STEP3
10.00
A+
10.00 A+
4
70
30.00 STEP1
30.02 STEP3
10.05
E-
30.05 STEP6


5
74
30.05
STEP6
TIM
001
#5
100ms Timer (Timer) [BCD Type]
TIMER
Timer number
Set value
6
76
30.00
STEP1
30.02
STEP3
10.07
G+
30.07
STEP8
30.09
STEP10
10.07
G+
7
83
30.00
STEP1
30.02
STEP3
11.04
K+
30.11
STEP12
30.13
STEP14
11.04
K+
8
90
30.01
STEP2
30.02
STEP3
10.02
C+
10.02
C+
9
94
30.01
STEP2
TIM
000
#20
100ms Timer (Timer) [BCD Type]
TIMER
Timer number
Set value
10
96
30.01
STEP2
30.05
STEP6
10.06
F+
10.06
F+
11
100
30.01
STEP2
30.08
STEP9
11.01
I+
11.01
I+
12
104
30.01
STEP2
30.12
STEP13
11.05
V+
11.05
V+
13
108
30.12
STEP13
TIM
002
#10
100ms Timer (Timer) [BCD Type]
TIMER
Timer number
Set value
14
110
30.03
STEP4
30.14
STEP15
10.01
B+
10.01
B+
15
114
30.02
STEP3
30.05
STEP6
10.04
E+
30.04
STEP5

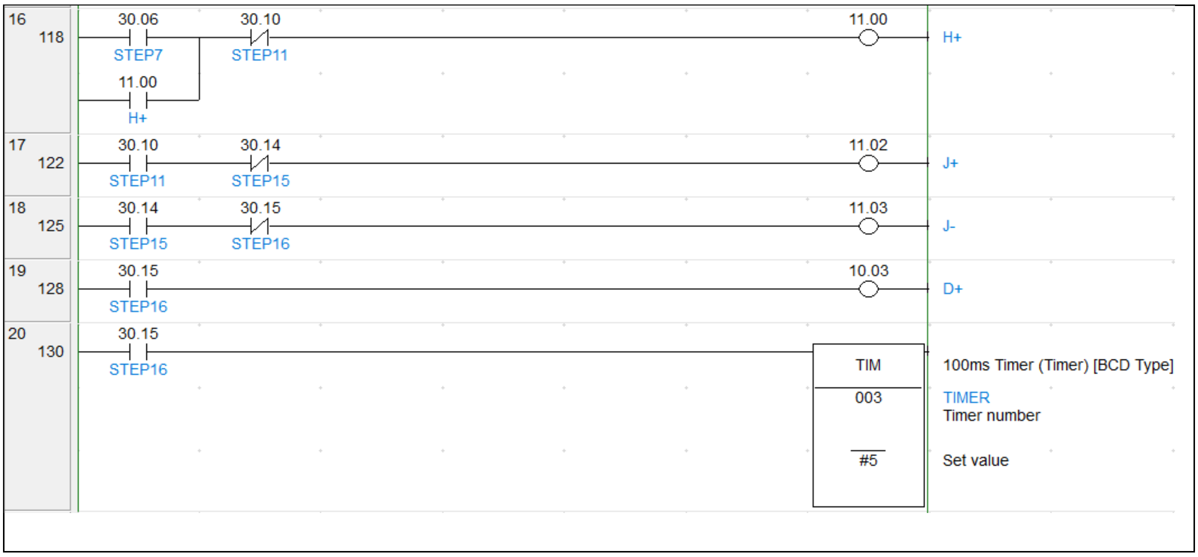


Figure 4.1: Ladder diagram of parallel sequence

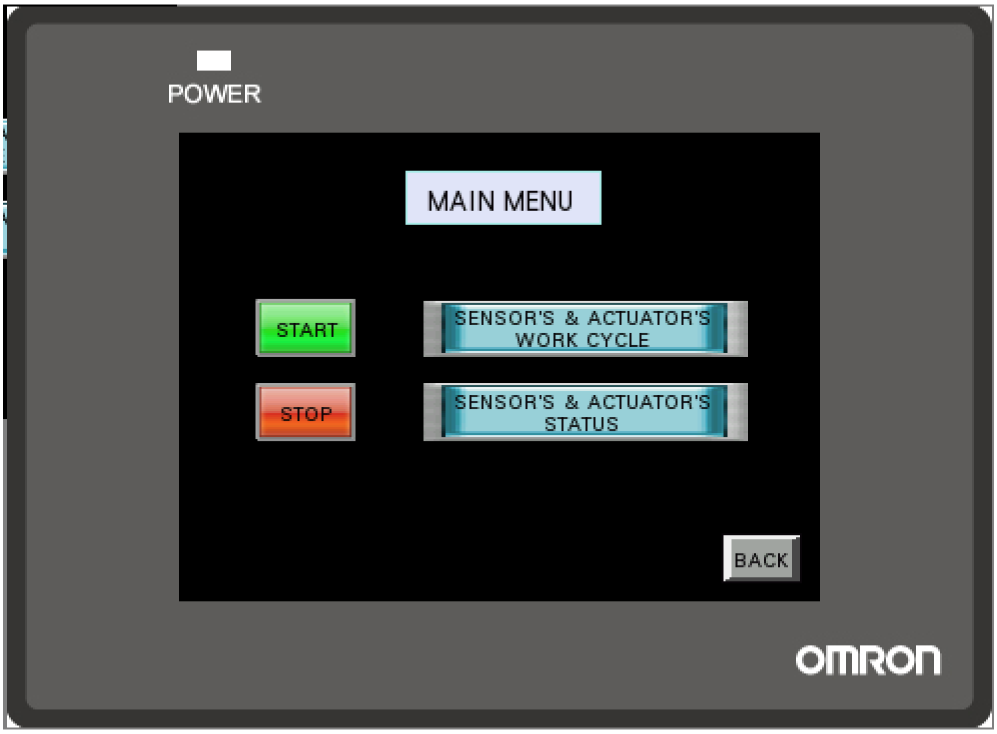


Figure 4.2: HMI screen display the main menu page

Table 4.1: HMI info address 1

Name	Type of Component	Type of address	Area/Variable	Address
Start	Bit button	Write address	CIO_IR_bit	0.00
Stop	Bit button	Write address	CIO_IR_bit	0.01
Sensor’s & Actuator’s work cycle	Function key	-	Change screen	-
Sensor’s & Actuator’s Status	Function key	-	Change screen	-
Back	Function key	-	Change screen	-

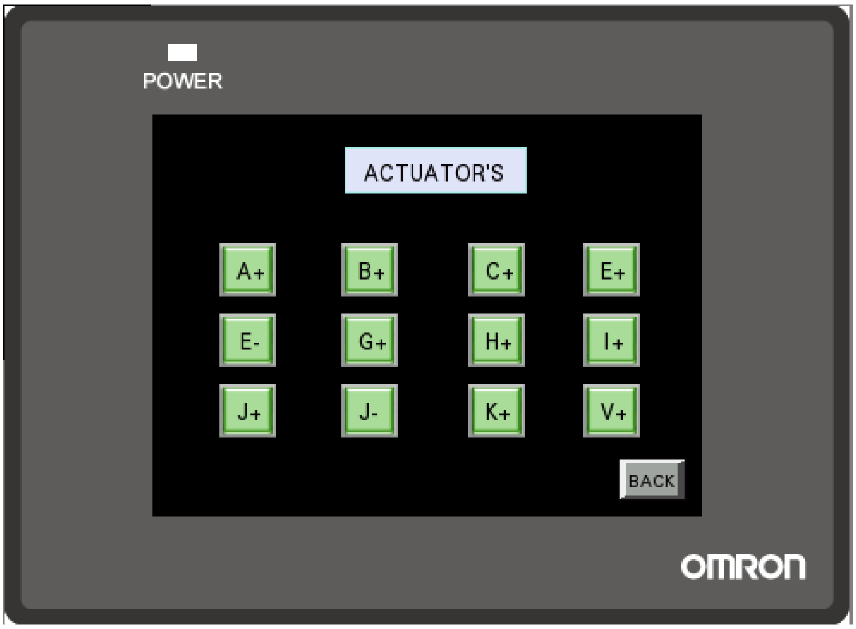


Figure 4.3: Actuator selection for display work cycle of sensor

Table 4.2: HMI info address 2

Name	Type of Component	Type of address	Area/Variable	Address
A+	Function key	-	Change screen	-
B+	Function key	-	Change screen	-
C+	Function key	-	Change screen	-
E+	Function key	-	Change screen	-
E-	Function key	-	Change screen	-
F+	Function key	-	Change screen	-
G+	Function key	-	Change screen	-
H+	Function key	-	Change screen	-
I+	Function key	-	Change screen	-
J+	Function key	-	Change screen	-
J-	Function key	-	Change screen	-
K+	Function key	-	Change screen	-
V+	Function key	-	Change screen	-
Back	Function key	-	Change screen	-

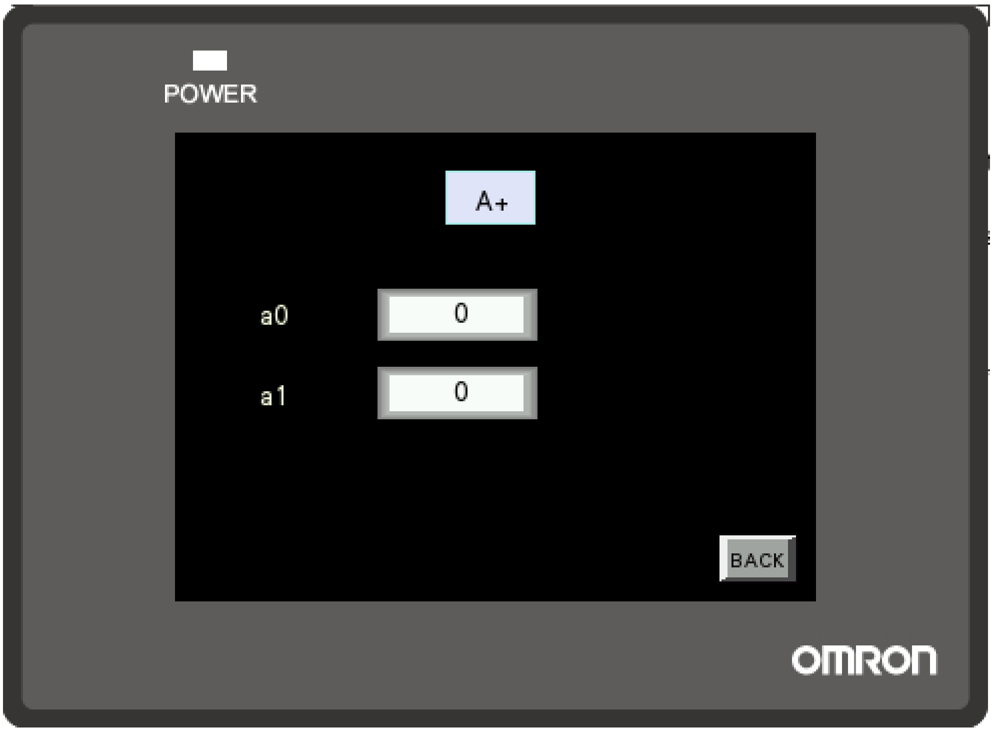


Figure 4.4: HMI Screen display work cycle of sensor a0 and a1

Table 4.3: HMI info address 3

Name	Type of Component	Type of address	Area/Variable	Address
a0	Number display	Read address	D_bit	1
a1	Number display	Read address	D_bit	2
Back	Function key	-	Change screen	-

Figure 4.5: HMI Screen display work cycle of sensor b1

Table 4.4: HMI info address 4

Name	Type of Component	Type of address	Area/Variable	Address
b1	Number display	Read address	D_bit	3
Back	Function key	-	Change screen	-

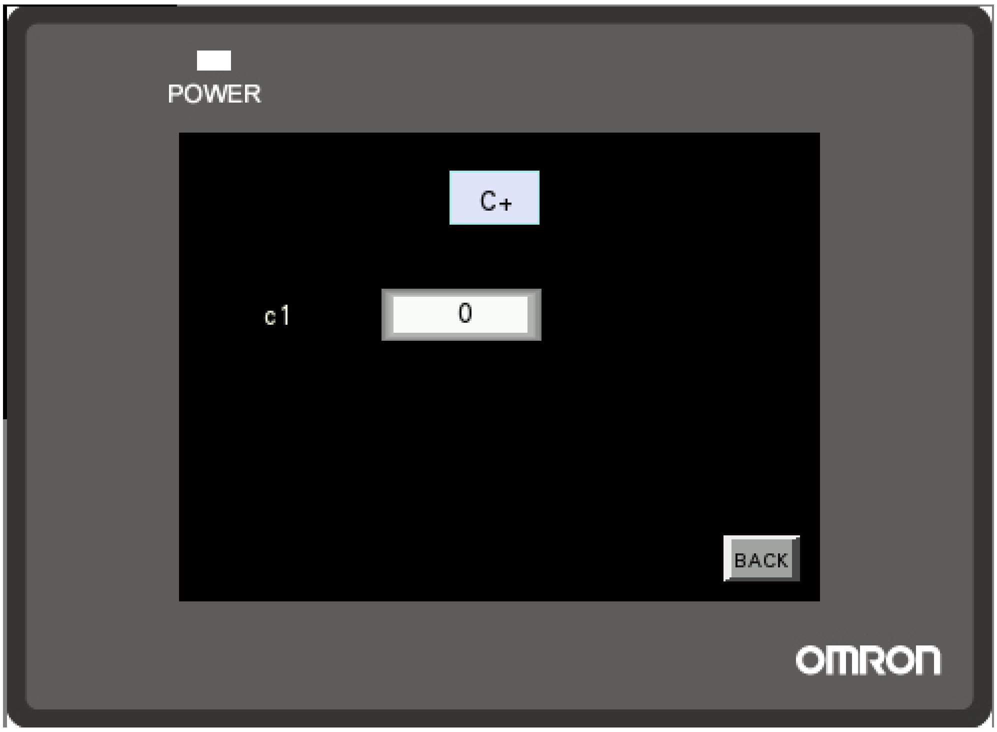


Figure 4.6: HMI Screen display work cycle of sensor c1

Table 4.5: HMI info address 5

Name	Type of Component	Type of address	Area/Variable	Address
c1	Number display	Read address	D_bit	4
Back	Function key	-	Change screen	-

Figure 4.7: HMI Screen display work cycle of sensor e1 and e2

Table 4.6: HMI info address 6

Name	Type of Component	Type of address	Area/Variable	Address
e1	Number display	Read address	D_bit	6
e2	Number display	Read address	D_bit	8
Back	Function key	-	Change screen	-

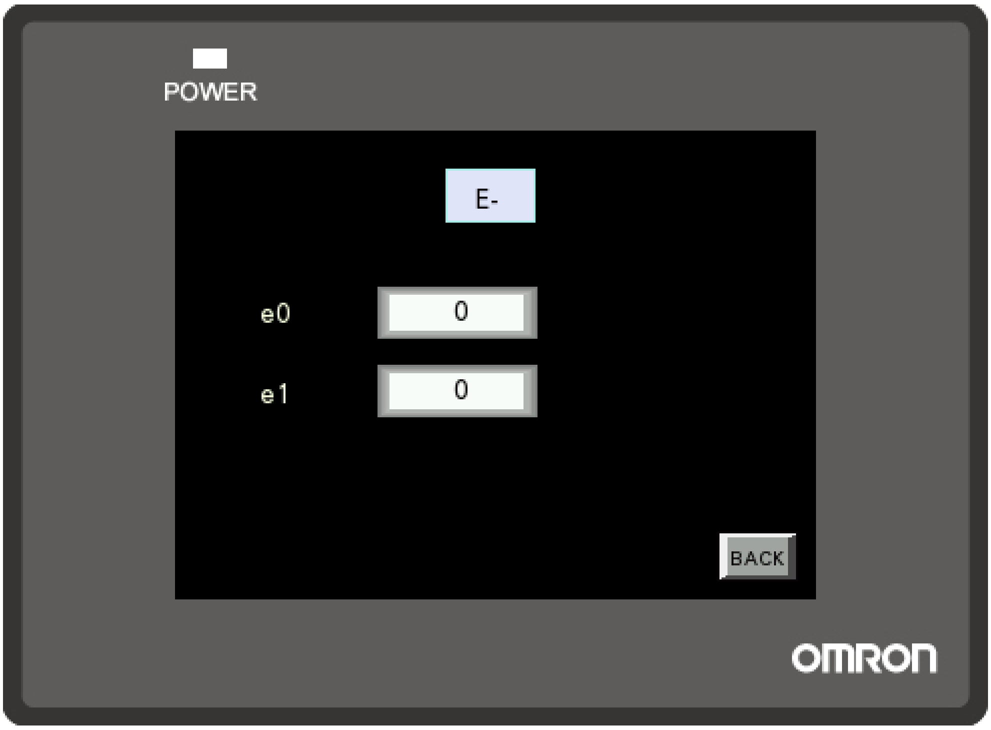


Figure 4.8: HMI Screen display work cycle of sensor e0 and e1

Table 4.7: HMI info address 7

Name	Type of Component	Type of address	Area/Variable	Address
e0	Number display	Read address	D_bit	5
e1	Number display	Read address	D_bit	6
Back	Function key	-	Change screen	-

Figure 4.9: HMI Screen display work cycle of sensor g1 and g0

Table 4.8: HMI info address 8

Name	Type of Component	Type of address	Area/Variable	Address
g0	Number display	Read address	D_bit	9
g1	Number display	Read address	D_bit	10
Back	Function key	-	Change screen	-

Figure 4.10: HMI Screen display work cycle of sensor h0 and h1

Table 4.9: HMI info address 9

Name	Type of Component	Type of address	Area/Variable	Address
h0	Number display	Read address	D_bit	11
h1	Number display	Read address	D_bit	12
Back	Function key	-	Change screen	-

Figure 4.11: HMI Screen display work cycle of sensor i0 and i1

Table 4.10: HMI info address 10

Name	Type of Component	Type of address	Area/Variable	Address
i0	Number display	Read address	D_bit	13
i1	Number display	Read address	D_bit	14
Back	Function key	-	Change screen	-

Figure 4.12: HMI Screen display work cycle of sensor j1

Table 4.11: HMI info address 11

Name	Type of Component	Type of address	Area/Variable	Address
j1	Number display	Read address	D_bit	16
Back	Function key	-	Change screen	-

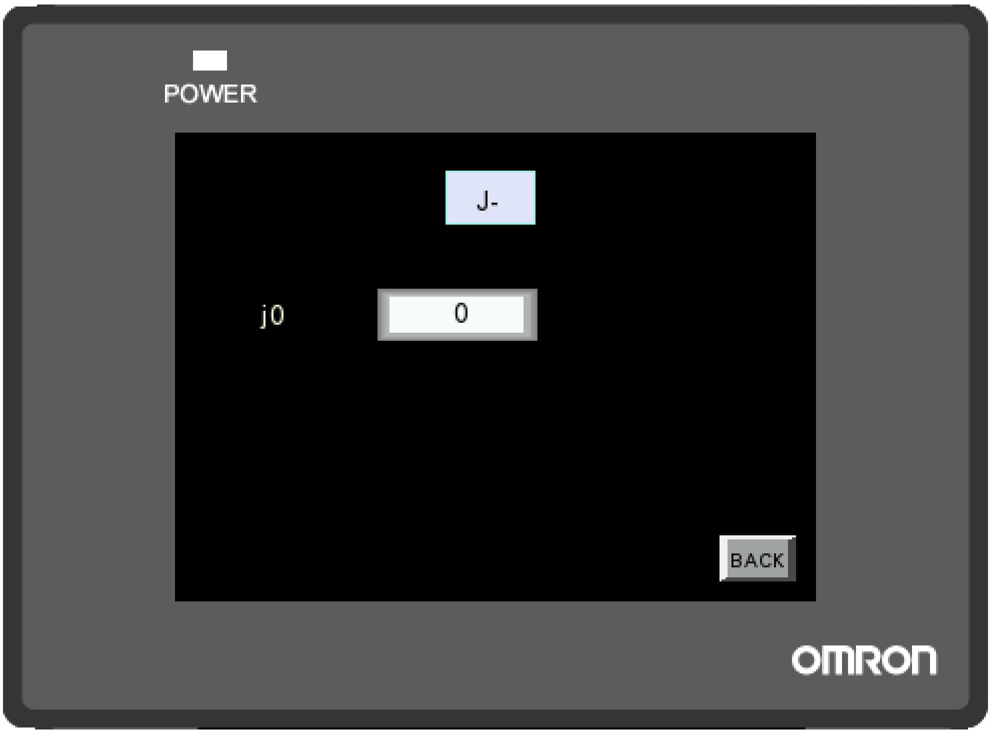


Figure 4.13: HMI Screen display work cycle of sensor j0

Table 4.12: HMI info address 12

Name	Type of Component	Type of address	Area/Variable	Address
j0	Number display	Read address	D_bit	15
Back	Function key	-	Change screen	-

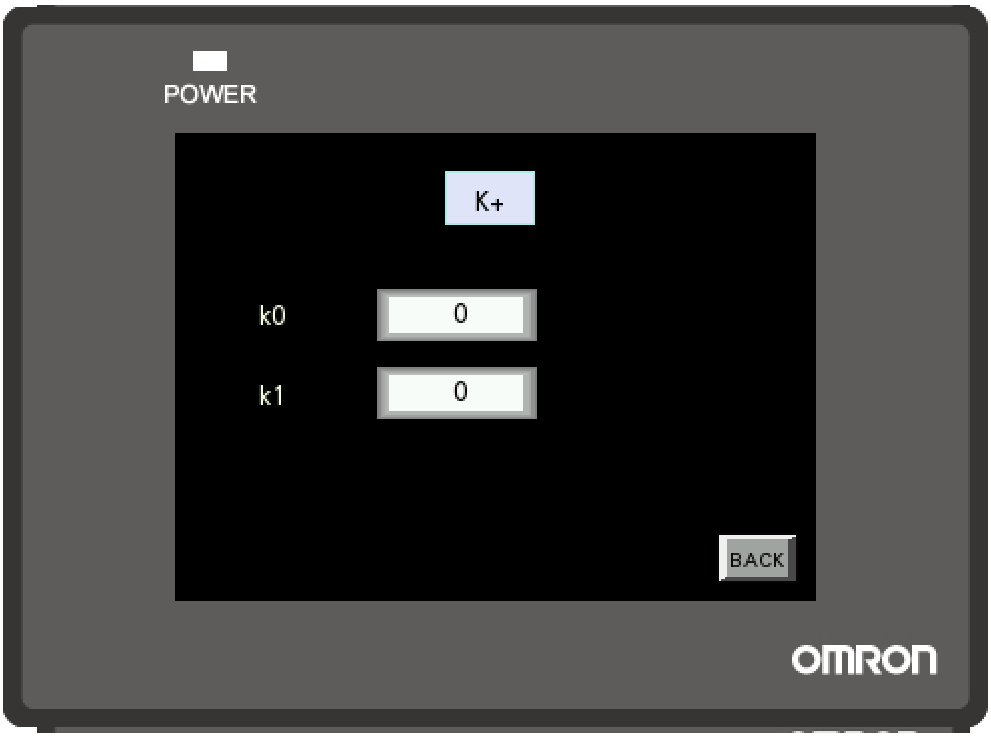


Figure 4.14: HMI Screen display work cycle of sensor k0 and k1

Table 4.13: HMI info address 13

Name	Type of Component	Type of address	Area/Variable	Address
k0	Number display	Read address	D_bit	17
k1	Number display	Read address	D_bit	18
Back	Function key	-	Change screen	-

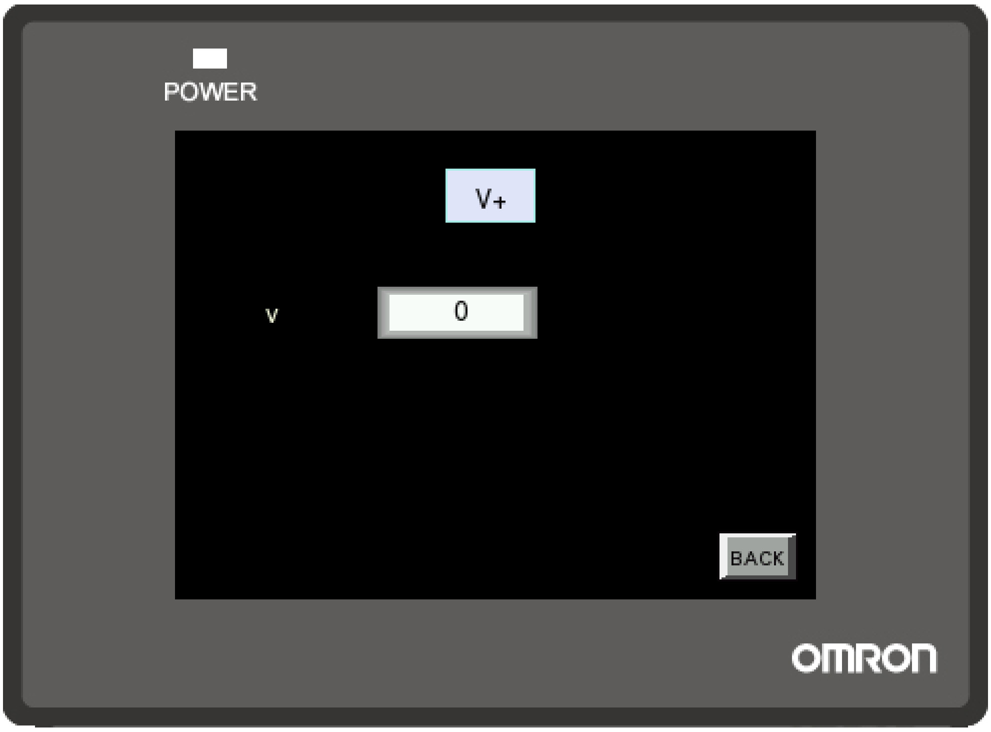


Figure 4.15: HMI Screen display work cycle of sensor v

Table 4.14: HMI info address 14

Name	Type of Component	Type of address	Area/Variable	Address
v	Number display	Read address	D_bit	19
Back	Function key	-	Change screen	-

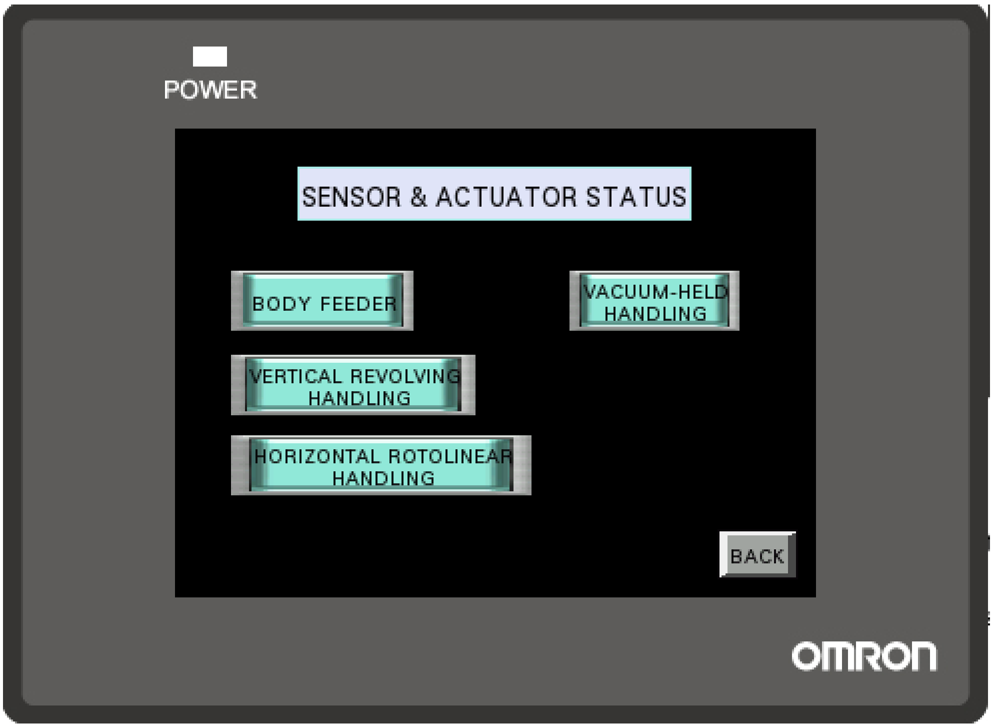


Figure 4.16: HMI screen display the selection to view subsystem status

Table 4.15: HMI info address 15

Name	Type of Component	Type of address	Area/Variable	Address
Body Feeder	Function key	-	Change screen	-
Vertical Revolving Handling	Function key	-	Change screen	-
Horizontal Rotolinear Handling	Function key	-	Change screen	-
Vacuum-held Handling	Function key	-	Change screen	-
Back	Function key	-	Change screen	-

Figure 4.17: HMI screen display the status for actuator A+, B+ and C+

Table 4.16: HMI info address 16

Name	Type of Component	Type of address	Area/Variable	Address
A+	Bit lamp	Read address	CIO_IR_bit	10.00
B+	Bit lamp	Read address	CIO_IR_bit	10.01
C+	Bit lamp	Read address	CIO_IR_bit	10.02
Back	Function key	-	Change screen	-

Figure 4.18: HMI screen display the status for actuator E+, E- and F+

Table 4.17: HMI info address 17

Name	Type of Component	Type of address	Area/Variable	Address
E+	Bit lamp	Read address	CIO_IR_bit	10.04
E-	Bit lamp	Read address	CIO_IR_bit	10.05
F+	Bit lamp	Read address	CIO_IR_bit	10.06
Back	Function key	-	Change screen	-

Figure 4.19: HMI screen display the status for actuator G+, H+ and I+

Table 4.18: HMI info address 18

Name	Type of Component	Type of address	Area/Variable	Address
G+	Bit lamp	Read address	CIO_IR_bit	10.07
H+	Bit lamp	Read address	CIO_IR_bit	11.00
I+	Bit lamp	Read address	CIO_IR_bit	11.01
Back	Function key	-	Change screen	-

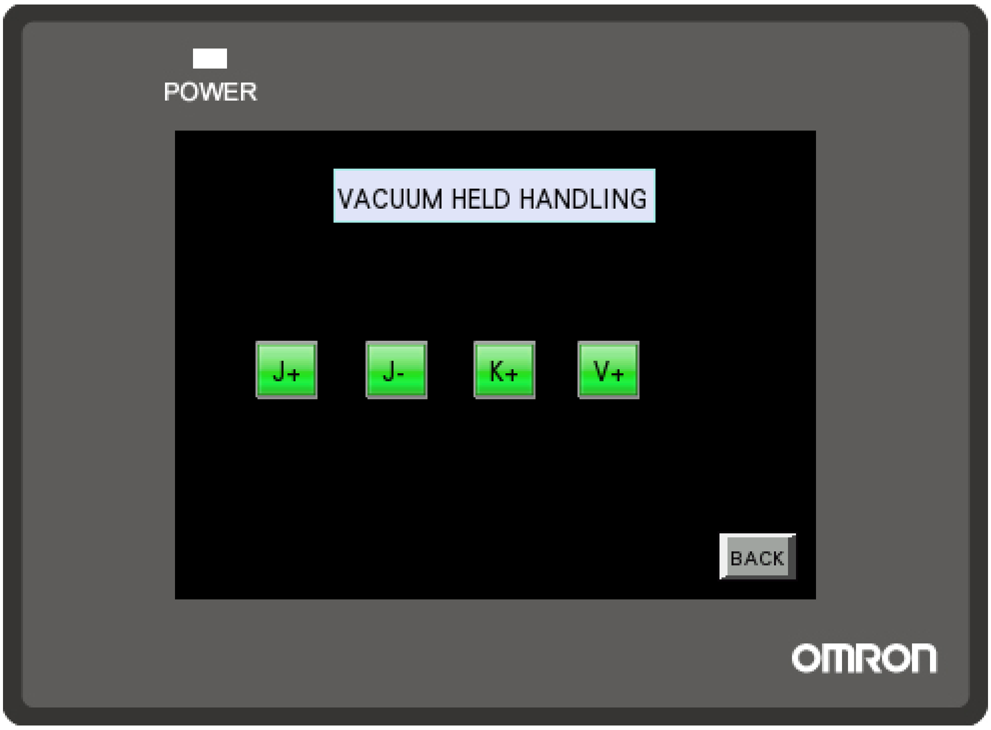


Figure 4.20: HMI screen display the status for actuator J+, J-, K+ and V+

Table 4.19: HMI info address 19

Name	Type of Component	Type of address	Area/Variable	Address
J+	Bit lamp	Read address	CIO_IR_bit	11.02
J-	Bit lamp	Read address	CIO_IR_bit	11.03
K+	Bit lamp	Read address	CIO_IR_bit	11.04
V+	Bit lamp	Read address	CIO_IR_bit	11.05
Back	Function key	-	Change screen	-

Figure 4.1 show the sequence of ladder diagram for the full process of MAP-205. For figure 4.2 show the Main Menu on the HMI screen display, there have start button to run the MAP205, stop button to stop the process, and Sensor's & Actuator's button to display part of actuator's and sensor's of Modular Production Sytem at the next page. Figure 4.4 to figure 4.15 show the number display for the sensor work cycle. Lastly, figure 4.16 and figure 4.20 show the subsystem selection and the status of actuator.

4.3 Analysis

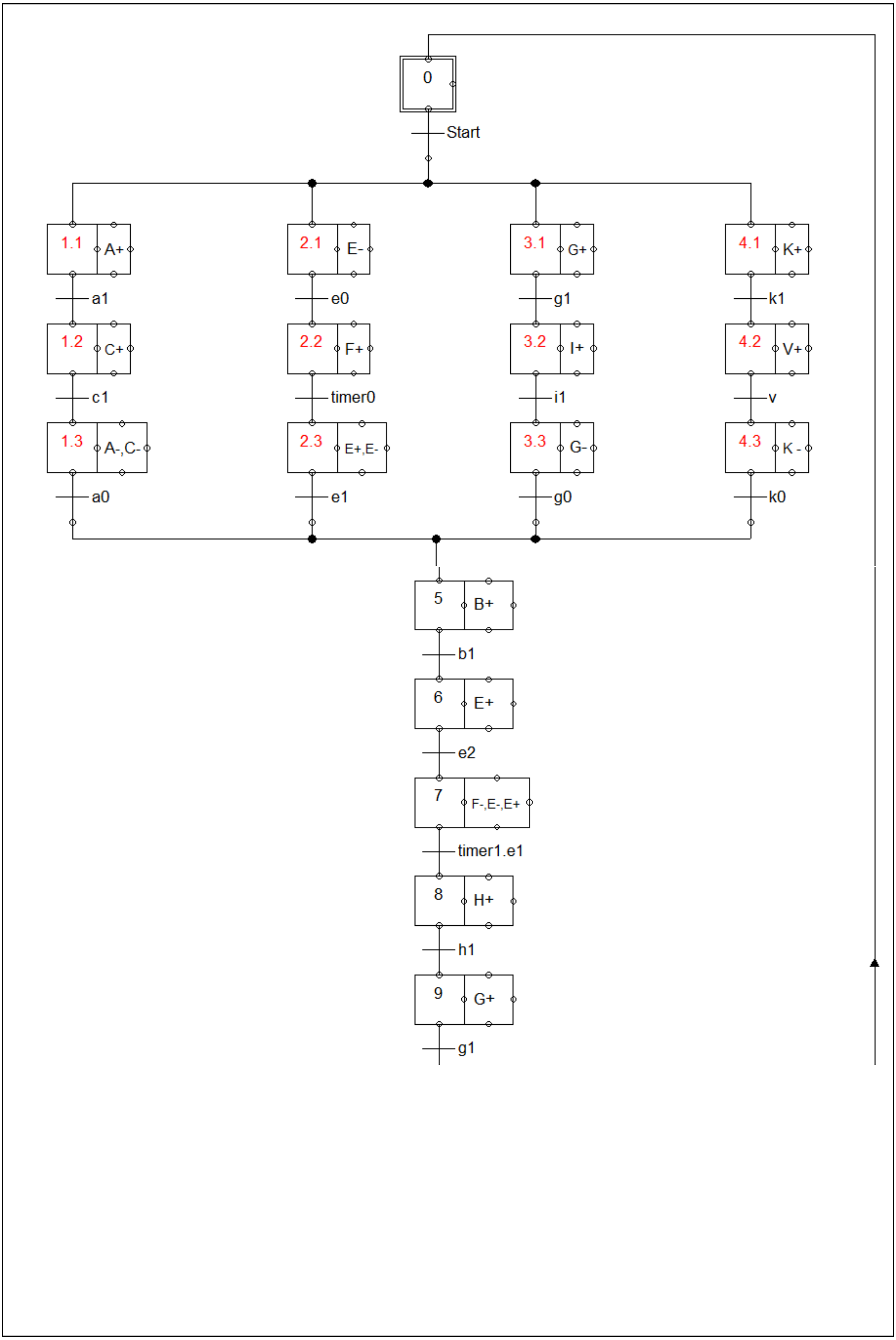
4.3.1 Data analysis for grafcet of original program and improve program

Table 4.20: The type of motion and symbol of sensor

SENSOR		
Type of motion	Symbol	PLC I/O Address
Body feeder backward	a0	0.03
Body feeder forward	a1	0.04
Body transfer cylinder forward	b1	0.05
Position verification cylinder downward	c1	0.06
Rotary actuator backward	e0	0.07
Rotary actuator in the middle position	e1	0.08
Rotary actuator forward	e2	0.09
Shaft insertion manipulator upward	g0	0.10
Shaft insertion manipulator downward	g1	0.11
Shaft insertion manipulator backward	h0	1.00
Shaft insertion manipulator forward	h1	1.01
gripper opened	i0	1.02
gripper closure	i1	1.03
Cover insertion manipulator backward	j0	1.04
Cover insertion manipulator forward	j1	1.05
Cover insertion manipulator upward	k0	1.06
Cover insertion manipulator downward	k1	1.07
Vacuum	v	1.08

Table 4.21: The type of motion and symbol of actuator

ACTUATOR		
Type of motion	Symbol	PLC I/O Address
Feeder cylinder forwards	A+	10.00
Body transfer cylinder forwards	B+	10.01
Verification cylinder downward	C+	10.02
Rejection cylinder forward	D+	10.03
Rotary actuator forward	E+	10.04
Rotary actuator backward	E-	10.05
Opening of the bearing gripper	F+	10.06
Shaft insertion manipulator downwards	G+	10.07
Shaft insertion manipulator forwards	H+	11.00
Shaft gripper closure	I+	11.01
Cover insertion manipulator forward	J+	11.02
Cover insertion manipulator backward	J-	11.03
Cover insertion manipulator downwards	K+	11.04
Vacuum	V+	11.05



0
Start
1.1
A+
a1
1.2
C+
c1
1.3
A-,C-
a0
2.1
E-
e0
2.2
F+
timer0
2.3
E+,E-
e1
3.1
G+
g1
3.2
I+
i1
3.3
G-
g0
4.1
K+
k1
4.2
V+
v
4.3
K -
k0
5
B+
b1
6
E+
e2
7
F-,E-,E+
timer1.e1
8
H+
h1
9
G+
g1

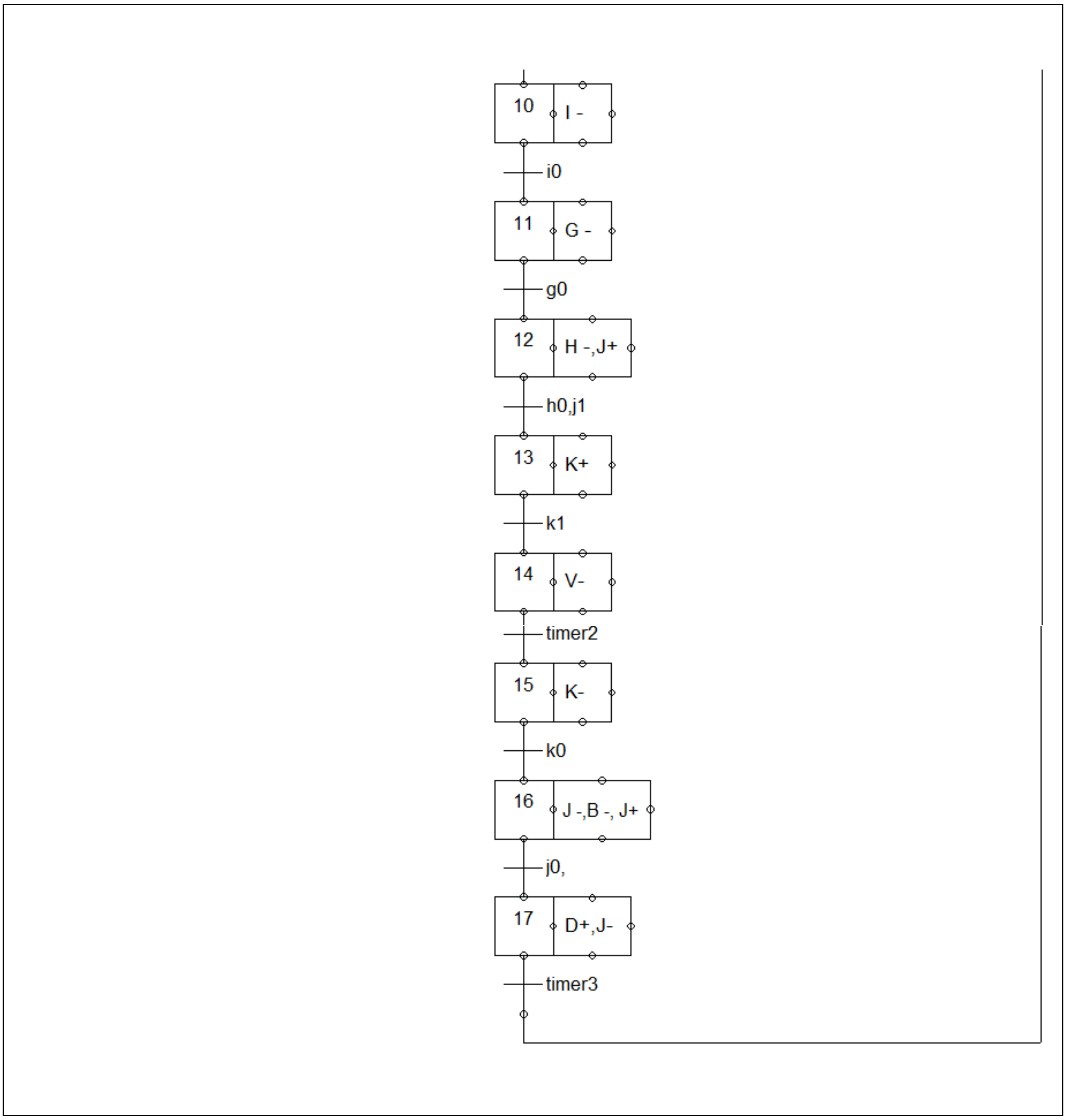
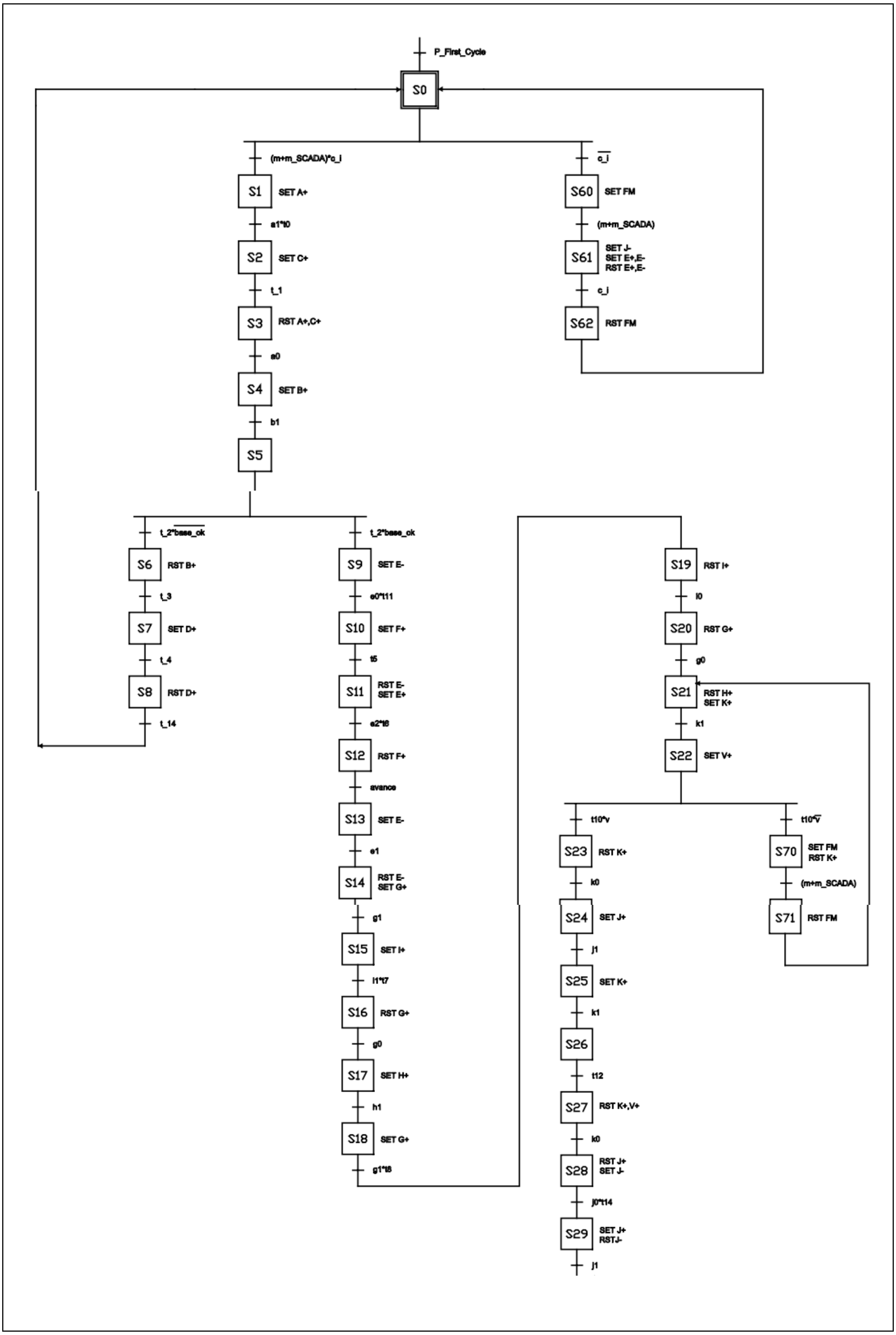


Figure 4.21: Improvement grafcet of sequence for automation system


P_First_Cycle
S0
(m+m_SCADA)*c_i
S1
SET A+
a1*t0
S2
SET C+
t_1
S3
RST A+,C+
a0
S4
SET B+
b1
S5
c_i
S60
SET FM
(m+m_SCADA)
S61
SET J-
SET E+,E-
RST E+,E-
c_i
S62
RST FM
t_2*base_ok
S6
RST B+
t_3
S7
SET D+
t_4
S8
RST D+
t_14
t_2*base_ok
S9
SET E-
e0*t11
S10
SET F+
t5
S11
RST E-
SET E+
e2*t6
S12
RST F+
avance
S13
SET E-
e1
S14
RST E-
SET G+
g1
S15
SET I+
i1*t7
S16
RST G+
g0
S17
SET H+
h1
S18
SET G+
g1*t8
S19
RST I+
i0
S20
RST G+
g0
S21
RST H+
SET K+
k1
S22
SET V+
t10*v
S23
RST K+
k0
S24
SET J+
j1
S25
SET K+
k1
S26
t12
S27
RST K+,V+
k0
S28
RST J+
SET J-
j0*t14
S29
SET J+
RSTJ-
j1
t10*v
S70
SET FM
RST K+
(m+m_SCADA)
S71
RST FM

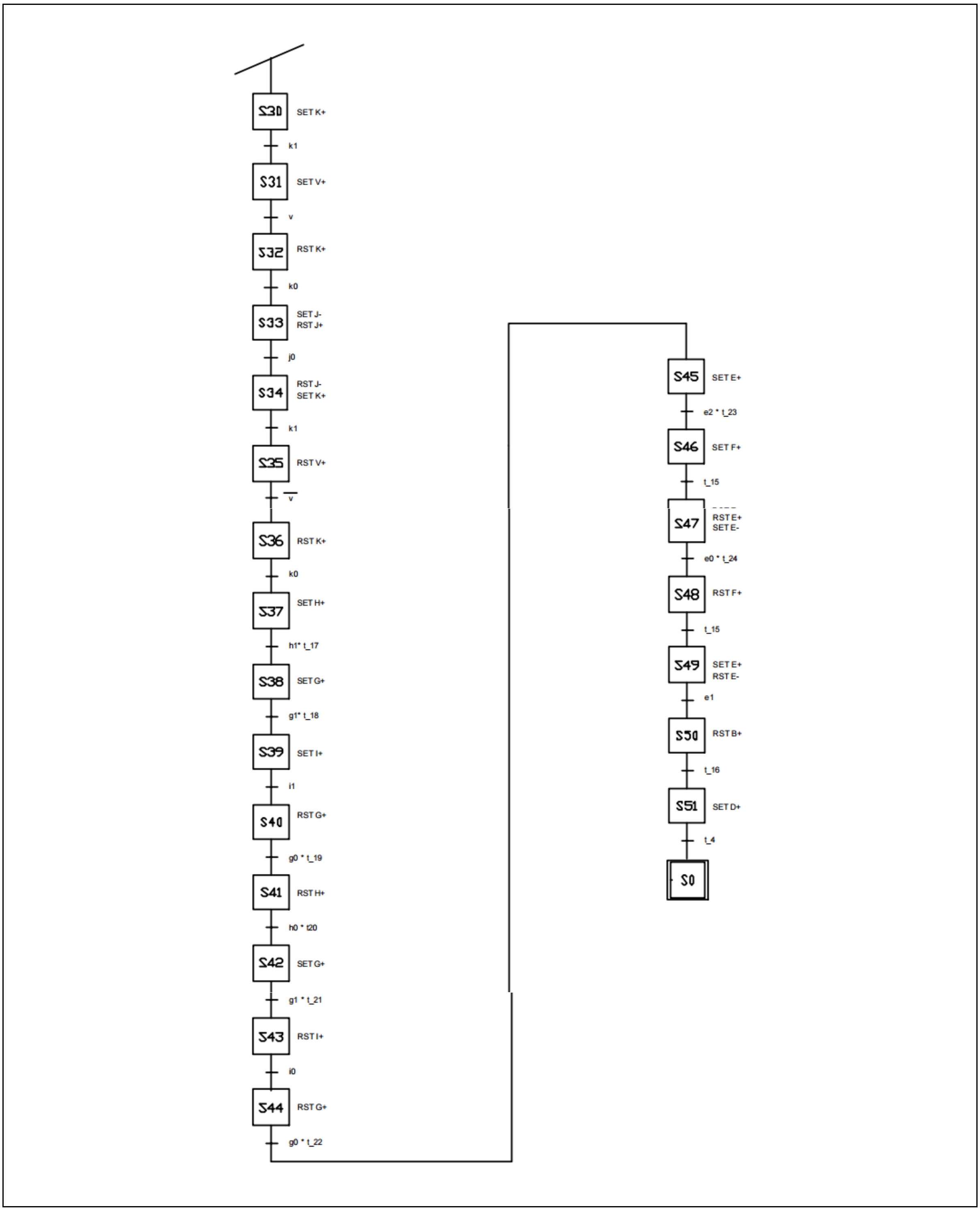


Figure 4.22: Original grafcet of sequence for automation system

4.3.2 Data analysis for the time taken and total to produce production box

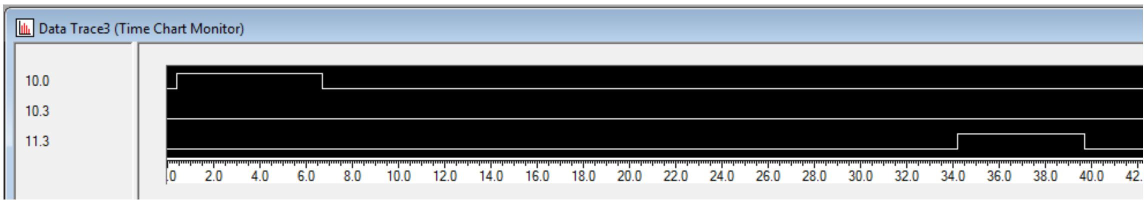


Figure 4.23: The time chart monitor for one production box with original program

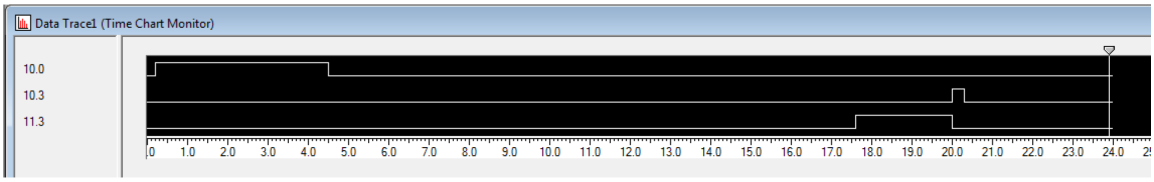


Figure 4.24: The time chart monitor for one production box with original program

Table 4.22: The comparison of time taken to produce one production box

	Original Program Full Process	Original Program Assembly process	Improve Program Assembly process
Time Taken for One Box Production	72.4 second	39.8 second	20 second

Table 4.23: Total of produce production box

	Original Program	Improve Program
Day (8 hours)	814 box	1620 box
Month (30 days)	24420 box	48600 box
Year (365 days)	297110 box	591300 box

Table 4.24: Calculation of production box

Original program	Improve program
Time taken for one day $= 60 \text{ second } \times 60 \text{ minute } \times 9 \text{ hours} = 32400 \text{ second}$	
Time taken for one box $= \frac{32400}{39.8}$ $= 814 \text{ box}$	Time taken for one box $= \frac{32400}{20}$ $= 1620 \text{ box}$
Time taken for produce box (month) $= 814 \text{ box} \times 30 \text{ day}$ $= 24420 \text{ box/month}$	Time taken for produce box (month) $= 1620 \text{ box} \times 30 \text{ day}$ $= 48600 \text{ box/month}$
Time taken for produce box (year) $= 814 \text{ box} \times 365 \text{ day}$ $= 297110 \text{ box/year}$	Time taken for produce box (year) $= 1620 \text{ box} \times 365 \text{ day}$ $= 591300 \text{ box/year}$

Table 4.22 above show the comparison between original program and improve program. For table 4.23 above shown the data analysis for time taken of produce one box production and how much the production box produce for day, month and years .

4.3.3 Data analysis for total detection of actuator to produce production box for original program

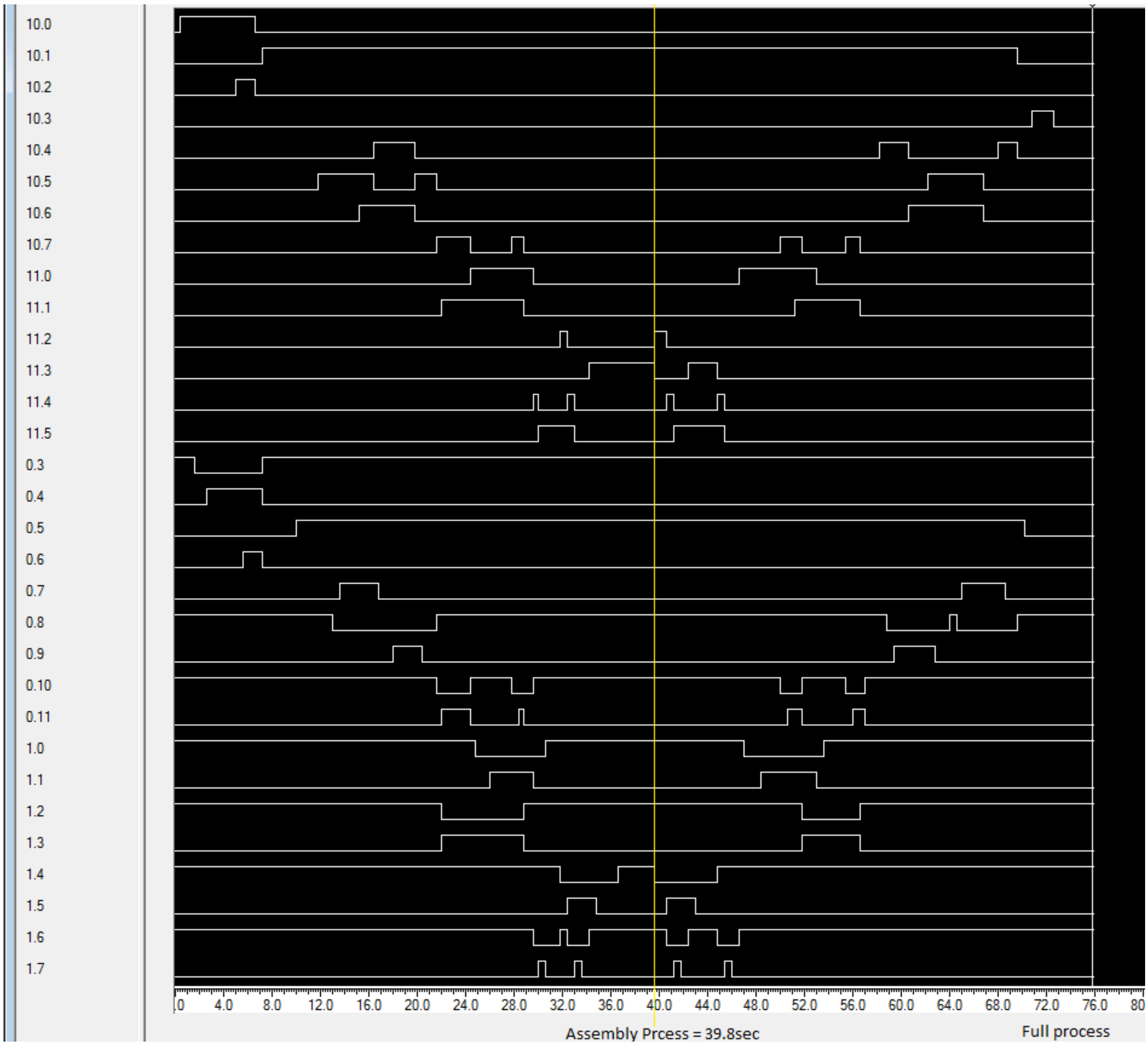


Figure 4.25: The time chart monitor for full sequence with original program.

$$\text{Time taken for one day} = 60 \text{ second } \times 60 \text{ minute } \times 9 \text{ hours} = 32400 \text{ second}$$

$$\text{Time taken for one box} = \frac{32400}{39.8} = 814 \text{ box}$$

$$\text{Time taken for produce box (month)} = 814 \text{ box} \times 30 \text{ day} = 24420 \text{ box/month}$$

$$\text{Time taken for produce box (year)} = 814 \text{ box} \times 365 \text{ day} = 297110 \text{ box/year}$$

Table 4.25: Total detection of sensor for original program

Original Program				
Sensor	Total Detection of One Production	Calculation		
		Day (8hours)	Month (30 days)	Year (365 days)
a0	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
a1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
b1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
c1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
e0	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
e1	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
e2	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
g0	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
g1	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
h0	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
h1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
i0	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
i1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
j0	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
j1	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
k0	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
k1	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
v	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle

Table 4.26: Total detection of actuator for original program

Original Program				
Actuator	Total Detection of One Production	Calculation		
		Day (8hours)	Month (30 days)	Year (365 days)
A+	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
B+	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
C+	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
D+	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
E+	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
E-	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
F+	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
G+	4 cycle	4 cycle × 814 = 3256 cycle	4 cycle × 24420 = 97680 cycle	4 cycle × 297110 = 1188440 cycle
H+	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
I+	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle
J+	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
J-	1 cycle	1 cycle × 814 = 814 cycle	1 cycle × 24420 = 24420 cycle	1 cycle × 297110 = 297110 cycle
K+	4 cycle	4 cycle × 814 = 3256 cycle	4 cycle × 24420 = 97680 cycle	4 cycle × 297110 = 1188440 cycle
V+	2 cycle	2 cycle × 814 = 1628 cycle	2 cycle × 24420 = 48840 cycle	2 cycle × 297110 = 594220 cycle

Figure 4.25 above show the time chart monitor for full sequence of original program. The time taken to produce one production box for original program is about 72.4 second and 39.8 second to finish the assembly process. Table 4.25 show total detection the cycle

of sensor for original program and table 4.26 total detection the cycle of actuator for original program.

4.3.4 Data analysis for total detection of actuator to produce production box for improve program

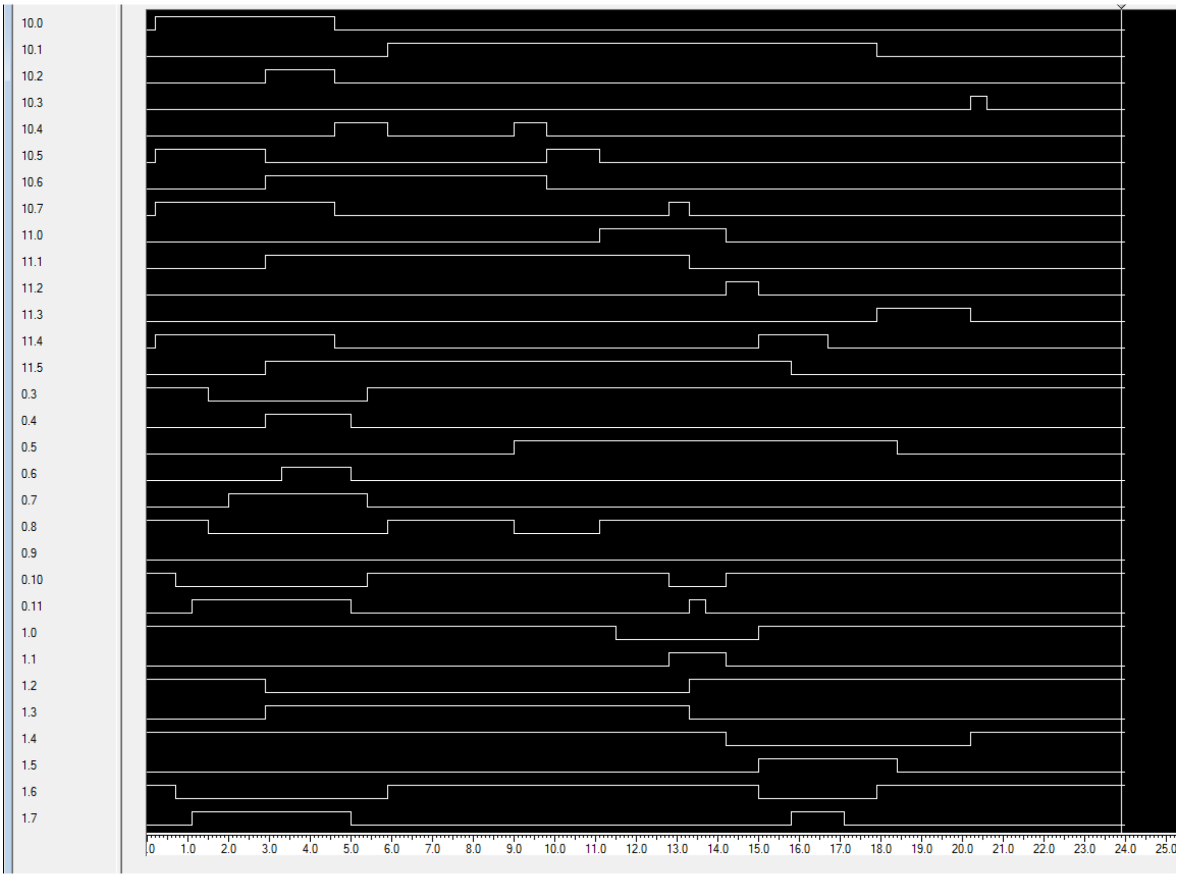


Figure 4.26: The time chart monitor with improve program

$$\text{Time taken for one day} = 60 \text{ second } \times 60 \text{ minute } \times 9 \text{ hours} = 32400 \text{ second}$$

$$\text{Time taken for one box} = \frac{32400}{20} = 16200 \text{ box}$$

$$\text{Time taken for produce box (month)} = \ 16200 \text{ box} \times 30 \text{ day} = 48600 \text{ box/month}$$

$$\text{Time taken for produce box (year)} = \ 16200 \text{ box} \times 365 \text{ day} = 591300 \text{ box/year}$$

Table 4.27: Total detection of sensor for improve program

Improve Program				
Sensor	Total Detection of One Production	Calculation		
		Day (8hours)	Month (30 days)	Year (365 days)
a0	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
a1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
b1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
c1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
e0	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
e1	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
e2	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
g0	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
g1	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
h0	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
h1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
i0	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
i1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
j0	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
j1	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
k0	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
k1	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
v	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle

Table 4.28: Total detection of actuator for improve program

Improve Program				
Actuator	Total Detection of One Production	Calculation		
		Day (8hours)	Month (30 days)	Year (365 days)
A+	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
B+	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
C+	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
D+	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
E+	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
E-	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
F+	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
G+	4 cycle	4 cycle × 1620 = 6480 cycle	4 cycle × 48600 = 194400 cycle	4 cycle × 591300 = 2365200 cycle
H+	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
I+	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle
J+	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
J-	1 cycle	1 cycle × 1620 = 1620 cycle	1 cycle × 48600 = 48600cycle	1 cycle × 591300 = 591300 cycle
K+	4 cycle	4 cycle × 1620 = 6480 cycle	4 cycle × 48600 = 194400 cycle	4 cycle × 591300 = 2365200 cycle
V+	2 cycle	2 cycle × 1620 = 3240 cycle	2 cycle × 48600 = 97200 cycle	2 cycle × 591300 = 1182600 cycle

Figure 4.26 above show the time chart monitor for sequence of improve program. The time taken to produce one production box take 20 second to finish the assembly process. Table 4.27 show total detection the cycle of sensor for improve program and table 4.28 total detection the cycle of actuator for improve program.

4.3.5 Data analysis for average expectacy lifespan of sensor for original program.

$\text{Average life span} = 100000$

Table 4.29: The average expectacy lifespan cycles of sensor for original program.

Original Program		
Sensor	Total for One day	Max working (days)
a0	814	$\frac{100000}{814} = 122.85 \text{ days}$
a1	814	$\frac{100000}{814} = 122.85 \text{ days}$
b1	814	$\frac{100000}{814} = 122.85 \text{ days}$
c1	814	$\frac{100000}{814} = 122.85 \text{ days}$
e0	814	$\frac{100000}{814} = 122.85 \text{ days}$
e1	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
e2	814	$\frac{100000}{814} = 122.85 \text{ days}$
g0	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
g1	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
h0	814	$\frac{100000}{814} = 122.85 \text{ days}$
h1	814	$\frac{100000}{814} = 122.85 \text{ days}$
i0	814	$\frac{100000}{814} = 122.85 \text{ days}$
i1	814	$\frac{100000}{814} = 122.85 \text{ days}$
j0	814	$\frac{100000}{814} = 122.85 \text{ days}$
j1	814	$\frac{100000}{814} = 122.85 \text{ days}$
k0	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
k1	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
v	1628	$\frac{100000}{1628} = 61.42 \text{ days}$

4.3.6 Data analysis for average expectacy lifespan of sensor for improve program.

$$\text{Average life span} = 100000$$

Table 4.30: The average expectacy lifespan cycles of sensor for improve program

Original Program		
Sensor	Total for One day	Max working (days)
a0	1620	$\frac{100000}{1620} = 61.73\text{ days}$
a1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
b1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
c1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
e0	1620	$\frac{100000}{1620} = 61.73\text{ days}$
e1	3240	$\frac{100000}{3240} = 30.86\text{ days}$
e2	1620	$\frac{100000}{1620} = 61.73\text{ days}$
g0	3240	$\frac{100000}{3240} = 30.86\text{ days}$
g1	3240	$\frac{100000}{3240} = 30.86\text{ days}$
h0	1620	$\frac{100000}{1620} = 61.73\text{ days}$
h1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
i0	1620	$\frac{100000}{1620} = 61.73\text{ days}$
i1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
j0	1620	$\frac{100000}{1620} = 61.73\text{ days}$
j1	1620	$\frac{100000}{1620} = 61.73\text{ days}$
k0	3240	$\frac{100000}{3240} = 30.86\text{ days}$
k1	3240	$\frac{100000}{3240} = 30.86\text{ days}$
v	3240	$\frac{100000}{3240} = 30.86\text{ days}$

4.3.7 Data analysis for average expectacy lifespan of actuator for original program.

$\text{Average life span} = 100000$

Table 4.31: The average expectacy lifespan cycles of actuator for original program.

Original Program		
Actuator	Total for One day	Max working (days)
A+	814	$\frac{100000}{814} = 122.85 \text{ days}$
B+	814	$\frac{100000}{814} = 122.85 \text{ days}$
C+	814	$\frac{100000}{814} = 122.85 \text{ days}$
D+	814	$\frac{100000}{814} = 122.85 \text{ days}$
E+	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
E-	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
F+	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
G+	3256	$\frac{100000}{3256} = 30.71 \text{ days}$
H+	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
I+	1628	$\frac{100000}{1628} = 61.42 \text{ days}$
J+	814	$\frac{100000}{814} = 122.85 \text{ days}$
J-	814	$\frac{100000}{814} = 122.85 \text{ days}$
K+	3256	$\frac{100000}{3256} = 30.71 \text{ days}$
V+	1628	$\frac{100000}{1628} = 61.42 \text{ days}$

4.3.8 Data analysis for average expectacy lifespan of actuator for improve program.

$\text{Average life span} = 100000$

Table 4.32: The average expectacy lifespan cycles of actuator for improve program.

Improve Program		
Actuator	Total for One day	Max working (days)
A+	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
B+	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
C+	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
D+	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
E+	3240	$\frac{100000}{3240} = 30.86\ \text{days}$
E-	3240	$\frac{100000}{3240} = 30.86\ \text{days}$
F+	3240	$\frac{100000}{3240} = 30.86\ \text{days}$
G+	6480	$\frac{100000}{6480} = 15.43\ \text{days}$
H+	3240	$\frac{100000}{3240} = 30.86\ \text{days}$
I+	3240	$\frac{100000}{3240} = 30.86\ \text{days}$
J+	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
J-	1620	$\frac{100000}{1620} = 61.73\ \text{days}$
K+	6480	$\frac{100000}{6480} = 15.43\ \text{days}$
V+	3240	$\frac{100000}{3240} = 30.86\ \text{days}$

Table 4.5 and table 4.6 above show the data of average expectacy lifespan cycles of actuator for original program and improvement program. The data show the total for maximum working years of actuatorif average lifespan of actuator is one million cycle.

4.4 Summary

To summarize in this chapter, production rate was successfully increased by using parallel sequence, in monitoring system also include the number display for sensor work cycle and the actuator work status. In addition, by using the existing data of sensor and actuator, calculation have been done in analysis to figure how many cycle of sensor and actuator work in a day, month and year. Sensor and actuator average cycle life varies by mode are designed fo up to 100000 full stroke. To find the earlier value for predictive maintenance is calculate the value of average life span devide by the value of work cycle, so from the value we get is the value for the sensor and actuator needed maintenance. From the analysis, unplanned breakdown can be avoided due to the sensor and actuator with the system monitoring.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, the system was able to control the parallel sequence of modular production system (MPS) and also be able to monitor sensor’s and actuator’s in modular production system by using Human Machine Interface (HMI). The benefit of using the improve program is because the gap difference of the time taken that produce production box from original program. For improve program only take just 20 second to finish one production box while the original program take 39.8 second. As we can see here that the less time taken to produce production box, the more output that can be produce.

This project also be able to see the sensor and actuator work cycle and status in HMI. The benefit is that this system can facilitate the work of the operator to check the condition of the sensor and actuator by monitoring the status and work cycle and at the same time can reduce risk of emergenc sguttiong down due to the sensor and actuator failure when the production is running.

5.2 Future Works

For future work that needs to be done:

i) To implement the system to monitoring the sensor’s and actuator’s work cycle and status by using the application with IoT.

REFERENCES

[1] S. Ardi, A. Ponco, and A. F. Ardin, “Design of Automatic Sleeve for Transfer Nut Clutch using Programmable Logic Controller,” no. 8, pp. 1–8.

[2] S. J. Mathew and B. Hemalatha, “Fault Identification and Protection of Induction Motor using PLC and SCADA,” pp. 8748–8754, 2014.

[3] M. Alipio *et al.*, “Automation of Packaging and Material Handling Using Programmable Logic Controller,” *Int. J. Sci. Eng. Technol.*, vol. 1, no. 3, pp. 767–770, 2014, [Online]. Available: https://www.researchgate.net/publication/270124023.

[4] U. Younas, S. Durrani, and Y. Mehmood, “Designing Human Machine Interface for Vehicle’s EFI Engine Using Siemen’s PLC and SCADA System,” *Proc. - 2015 13th Int. Conf. Front. Inf. Technol. FIT 2015*, pp. 205–210, 2016, doi: 10.1109/FIT.2015.44.

[5] K. Chakraborty, I. Roy, and P. De, “Controlling Process of a Bottling Plant using PLC and SCADA,” *Indones. J. Electr. Eng. Informatics*, vol. 3, no. 1, pp. 39–44, 2015, [Online]. Available: http://section.iaesonline.com/index.php/IJEEI/article/viewFile/125/pdf.

[6] R. C. Parpala and R. Iacob, “Application of IoT concept on predictive maintenance of industrial equipment,” *MATEC Web Conf.*, vol. 121, pp. 1–8, 2017, doi: 10.1051/matecconf/201712102008.

[7] A. M. L. Cachada, “Intelligent and predictive maintenance in manufacturing systems,” 2018, [Online]. Available: https://bibliotecadigital.ipb.pt/handle/10198/18301.

[8] K. S. Kiangala and Z. Wang, “Initiating predictive maintenance for a conveyor

motor in a bottling plant using industry 4.0 concepts," *Int. J. Adv. Manuf. Technol.*, vol. 97, no. 9–12, pp. 3251–3271, 2018, doi: 10.1007/s00170-018-2093-8.

[9] T. Thamaraimanalan, S. P. Vivekk, G. Satheeshkumar, and P. Saravanan, "Smart Garden Monitoring System Using IOT," *Asian J. Appl. Sci. Technol. (Open Access Q. Int. J.*, vol. 2, no. 2, pp. 186–192, 2018, [Online]. Available: www.ajast.net.

[10] A. I. Vlasov, P. V. Grigoriev, A. I. Krivoshein, V. A. Shakhnov, S. S. Filin, and V. S. Migalin, "Smart management of technologies: Predictive maintenance of industrial equipment using wireless sensor networks*," *Entrep. Sustain. Issues*, vol. 6, no. 2, pp. 489–502, 2018, doi: 10.9770/jesi.2018.6.2(2).