

CERTIFICATION OF APPROVAL

PLC ANALOGUE CONTROLLER FOR A BATCH PLANT

by

Wan Saladiwiyana Bt Wan Husin

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:



Dr. Nordin Bin Saad
Project Supervisor

**UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK**

JUNE 2006

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



Wan Saladiwiyana Bt Wan Husin

ABSTRACT

This report addresses the approach of designing and implementing the logic control functions for a batch process. The aim is to develop the controller for a batch process using PLC. Batch process involves sequential processing that leads to the production of finite quantities of material by subjecting quantities of input materials to an ordered set of processing activities over a finite period of time, using one or more pieces of equipment. Batch process can be classified as a discontinuous process. Liquid soap making is chosen as the case study for the project. The CX-Programmer programming software was used to develop the ladder diagram for the PLC, implementing the batch control. The concept of controlling a control system involves a systematic approach, and the operation procedure includes the determination of the machine sequence, inputs and outputs addressing, and the PLC programming. Troubleshooting is conducted to ensure that the program run in accordance to the intended process sequences and give the desired response of the control valves, the final control elements connected to the PLC. This approach is found to be effective in designing a logic ladder diagram for a PLC to control a batch process, in this case a liquid soap making process.

ACKNOWLEDGEMENT

First and foremost, I would like to thank God for constantly brace and enlighten every step through the project.

I would like to express my heartfelt appreciation to my supervisor, Dr. Nordin Saad. He has done a good job in monitoring and guiding me in completing the project. He also had been very supportive during the entire project period.

I am also indebted towards technicians especially for their willingness to offer helps and guidance in their area of specialty. Their experience, knowledge and view on the project have given me a better outlook in design project.

Last but not least, I would like to take this opportunity to express my sincere gratitude to all colleagues, friends and any individuals who are involve directly and indirectly for helping and assisting for the first stage requirement.

TABLE OF CONTENTS

CERTIFICATION OF APPROVAL	i
CERTIFICATION OF ORIGINALITY	ii
ABSTRACT	iii
ACKNOWLEDGEMENT	iv
LIST OF FIGURES	vii
LIST OF TABLES	viii
CHAPTER 1: INTRODUCTION	1
1.1 Background of Study	1
1.2 Problem Statement	2
1.3 Project Objectives	2
CHAPTER 2: LITERATURE REVIEW	4
2.1 The Batch Process	4
2.1.1 Definition of Batch Process	4
2.1.2 Batch Process Control.	5
2.1.3 Batch Cycle Time	5
2.1.4 Application of Batch Process Industry	5
2.2 Programmable Logic Controller (PLC)	6
2.2.1 PLC History	6
2.2.2 PLC Components	7
2.2.3 PLC Operation	8
2.2.4 PLC Programming	9
2.2.5 PLC Languages	9
2.2.6 Ladder Languages	10
2.2.7 PLC Advantages	12
2.2.8 Timing Diagram	13
2.3 Relationship between PLC and Batch Process	14
CHAPTER 3: METHODOLOGY	15

3.1	Project Work	15
3.2	Studies on Liquid Soap Making Process	16
3.3	Analogue Module	18
3.4	A Systematic Approach to PLC Design	22
	3.4.1 The Machine Sequence of Operation	22
	3.4.2 Inputs and Outputs Addressing	23
	3.4.3 PLC Programming	23
3.5	Troubleshooting	23
3.6	Hardware Connection	23
CHAPTER 4: RESULTS AND DISCUSSION		24
4.1	Process Sequences	24
4.2	Inputs and Outputs Addressing	26
4.3	Ladder Diagram	27
CHAPTER 5: CONCLUSION AND RECOMMENDATION		33
REFERENCES		34
APPENDICES		35

LIST OF FIGURES

- Figure 2-1 A typical Unitary PLC
- Figure 2-2 A simple Ladder Logic Diagram
- Figure 2-3 Time-based timing diagram
- Figure 2-4 Value-based timing diagram
- Figure 3-1 Project Workflow
- Figure 3-2 Application procedures of analogue module
- Figure 3-3 Analogue module configuration
- Figure 3-4 Analogue inputs data and converted values specification
- Figure 3-5 Analogue settings and output data application
- Figure 4-1 Process Sequences Flowchart
- Figure 4-2 Process ladder diagram programming

LIST OF TABLES

Table 3-1	DM 6611 PC Setup Setting
Table 4-1	Digital inputs addressing
Table 4-2	Digital outputs addressing
Table 4-3	Analogue inputs addressing
Table 4-4	Analogue outputs addressing

CHAPTER 1

INTRODUCTION

1.1 Background of Study

A batch process produces products in finite quantities. In an ideal situation, this product is determined by a recipe that has a name and contains information on the ingredients or raw material to be used, the order of process steps and the process conditions, and finally the equipment to be used in the process.

In practice, batch process control application varies widely in complexity. The simplest batch control applications produce only one end product. A single product process follows the same procedure and process set points and requires the same raw material amounts for each batch. Some batch control applications require different sets of process steps for different products. A different sequence of activities is required to make different products. Typically, different products also required different raw material amounts and process set points.

The aim of this project is to design a control system of a liquid soap making batch process. The designed of the control system is carried out by using Programmable Logic Controller (PLC). The CX-Programmer is the programming software that was used to design the control system where the programming language of the software is the ladder diagram.

A PLC consists of a central processing unit (CPU) containing an application program and input and output interface modules, which are directly, connected to the field I/O devices. The program controls the PLC so that when an input signal from an input

device turns ON, the appropriate response is made. The task of commanding the plant devices is devoted to a modular control section and the program is based on the flowchart and timing diagram of the operations sequences.

1.2 Problem Statement

The project required research and literature review to develop understanding on batch process and PLC. In addition, the understanding of liquid soap making process control system is important to ensure that the program developed is consistent with the process requirement. Besides that, the design on the process flow chart and timing diagram are also important for a successful output of the project.

The PLC that was used for the project has digital and analogue modules which represent the signal used in the programming. The configuration of both modules was conducted to obtain the outcome of the project as desired. The understanding on how the program of the process should be developed is a critical task because without the knowledge the program might not be successfully achieved.

The implementation of the control valve as the output of the analogue controller in the control system is also critical. The signal sends to the control valve from the PLC analogue output point must be determined correctly for the control valve to response on the control system.

1.3 Project Objectives

The objectives of the project are:

- To enhance understanding and knowledge on batch process plant and PLC digital and analogue modules as well as its programming system.
- To design the control system and illustrate the liquid soap making batch process.

The followings are the several aspects to be focused in completing the project:

- Research, design and develop the process control system.
- Flow chart and timing diagram construction for the operation of the overall control system.
- Configuration of the PLC inputs and outputs of both digital and analogue modules and also the control valve connection.
- Development of ladder diagram for the overall process system using CX-Programmer programming tool.
- Hardware connection between the PLC and the control valve.
- Simulation and troubleshooting of the designed ladder diagram.

CHAPTER 2

LITERATURE REVIEW

2.1 Batch Process

2.1.1 Definition of Batch Process

A batch process is defined as a process that leads to the production of finite quantities of material by subjecting quantities of input materials to an ordered set of processing activities over a finite period of time using one or more pieces of equipment. The batch mode of production has become increasingly important in the process industry during the last decade. Batch plants can be made highly flexible, and thereby well suited for manufacturing of a large variety of product types. Batch processing is normally associated with the chemical process industries [5].

Batch processes are classified by the number of products they can make and the structure of the plant. A single product batch plant produces the same product in each batch. A multi-product batch plant produces different products using the same production cell. In a single-path batch process every batch goes through the same series of equipment units. In a network-structured batch plant the equipment units are flexibly interconnected.

A batch can be defined as the material that is being produced or that has been produced by a single execution of a batch process. It also can be known as an entity that represents the production of a material at any point in the process [1].

2.1.2 Batch Process Control

Batch control is well known, though complex, way of controlling a plant. Its complexity is the reason why batch control automation has stayed behind in relation to continuous control or even sequential discrete parts manufacturing. In effect batch processing contains the ideas of both techniques along with the elements of logistic manufacturing and handling [5].

Even though batch automation is harder to perform, more difficult to design, and more complex to operate, batch plants exist and are of an increasing interest. Despite the specific and almost obstinate characteristic of batch control, it is very challenging to use batch operated facilities.

Batch control refers to the type of control that is typically used with a batch process. A batch process control is one in which the functions of material transfer and/or material processing are cyclical with repeatable results. A process is one in which the measured and controlled variables are manipulated to provide a product at a specified continuous rate. A batch process is one that provides a homogeneous, discrete quantity of product at the end of a specified time.

2.1.3 Batch Cycle Time

The length of production time, from beginning a batch of a desired product in process equipment to completing that process and preparing for the next batch to begin, is referred to as one batch cycle of the process unit of operation. This batch cycle time is the most important productivity concern after a product's mass balance and quality have been established and becomes repeatable. It is the cycle time that ultimately determines the magnitudes of investment required to produce saleable in the marketplace.

2.1.4 Application of Batch Process in Industry

Batch processes are familiar to most people since we use them in everyday life. For instance, when we bake a cake, we follow a recipe that involves adding ingredients,

stirring the mixture, pouring it into baking pans, putting the pans into the oven for a specific time at a specific temperature, etc. Industrial batch processes are similar to the process of baking a cake but scaled up to produce a larger quantity of material. A variety of products are produced using batch processes. Food, beverages, pharmaceutical products, paint, fertilizer, and cement are a few of the categories of products produced using batch processes. Some products such as food, beverages, and pharmaceuticals require precise tracking of batch information for safety and regulatory purposes [6].

2.2 Programmable Logic Controller (PLC)

2.2.1 PLC History

Control engineering has evolved over time. In the past, humans were the main method for controlling a system [8]. More recently electricity has been used for control and early electrical control was based on relays (*refer Appendix A, Figure 1*). These relays allow power to be switched on and off without a mechanical switch. It is common to use relays to make simple logical control decisions. The development of low cost computer has brought the most recent revolution, the Programmable Logic Controller (PLC). The advent of the PLC began in the 1970s, and has become the most common choice for manufacturing controls.

Programmable Logic Controller is an electronic control unit used to regulate the operation of a given packaging machine. PLC controls the machine's operating parameters and allow programming or changing of those parameters to accommodate differing application requirements in microprocessor-based industrial control system. It communicates with other process control components through data links. It is used in process control for simple switching tasks, PID control, complex data manipulation, arithmetic operations, timing and process and machine control [2].

2.2.2 PLC Components

A PLC consists of a central processing unit (CPU) containing an application program and input and output interface modules, which are directly, connected to the field I/O devices (*refer Appendix A, Figure 2*). The program controls the PLC so that when an input signal from an input device turns ON, the appropriate response is made. The response normally involves turning ON an output signal to some sort of output devices.

In an automated system, the PLC is commonly regarded as the heart of the control system. With a control application program (stored within PLC memory) in execution, the PLC constantly monitors the state of the system through the field input devices' feedback signal. It will then base on the program logic to determine the course of action to be carried out at the field output devices [8]. A typical unitary of PLC is shown below:

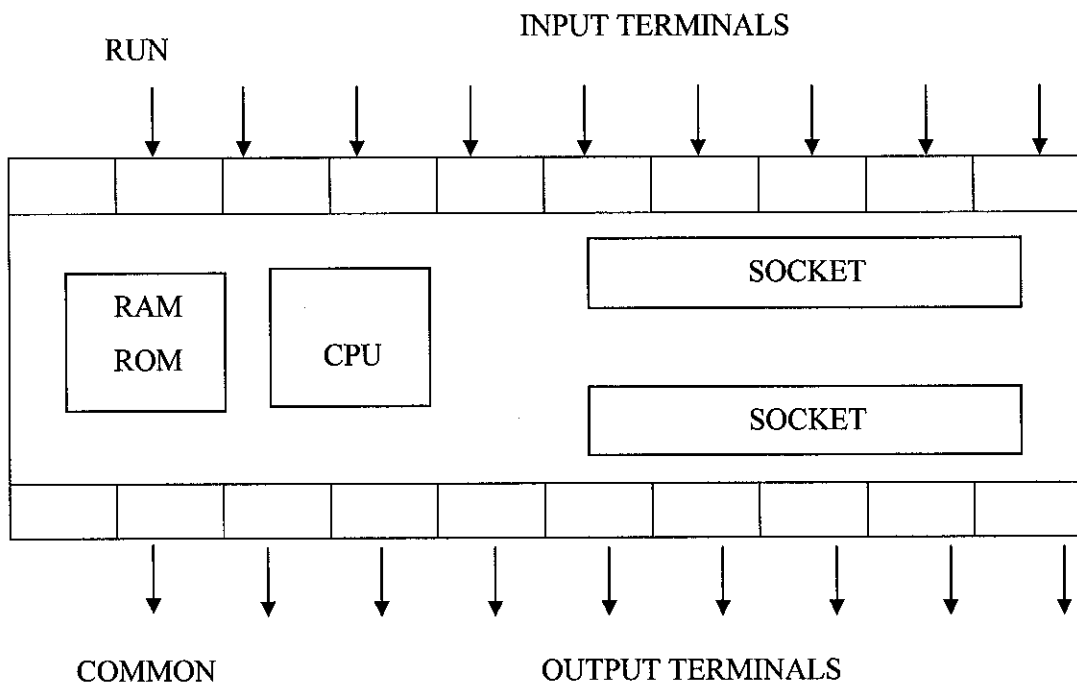


Figure 2-1: A typical unitary PLC

The CPU used in PLC system is a standard CPU present in many other microprocessor controlled systems [3]. The choice of the CPU depends on the process to be controlled. Generally 8 or 16 bit CPUs fulfill the requirements adequately.

Memory in a PLC system is divided into the program memory which is usually stored in EPROM/ROM, and the operating memory. The RAM memory is necessary for the operation of the program and the temporary storage of input and output data. Typical memory sizes of PLC systems are around 1kb for small PLCs, a few kb for medium sizes and greater than 10-20 kb for larger PLC depending on the requirements. Many PLC would support easy memory upgrades.

Input/Output units are the interfaces between the internal PLC systems and the external processes to be monitored and controlled. Since the PLC is a logic based device with a typical operating voltage of 5 volts and the external processes usually demand higher powers and currents, the I/O modules are optically or otherwise isolated. The typical I/O operating voltages are 5V - 240 V dc (or ac) and currents from 0.1A up to several amperes. The I/O modules are designed in this way to minimize or eliminate the need for any intermediate circuitry between the PLC and the process to be controlled. Small PLC units would have around 40 I/O connections with larger ones having more than 128 with both local or remote connections, and extensive upgrade capabilities.

2.2.3 PLC Operation

The PLC operates internally in a way very similar to computers. The inputs are continuously monitored and copied from the I/O module into RAM memory which is divided into the input and output sections. The CPU steps through the control program in another section of the memory and fetches the input variables from the input RAM. Depending on the program and the state of inputs, the output RAM is filled with the control variables which are then copied into the I/O module where they control the processes [8].

2.2.4 PLC Programming

Programming units are essential components of the PLC systems. Since they are used only in the development/testing stage of a PLC program, they are not permanently attached to the PLC. The program in a ladder diagram or other form can be designed and usually tested before downloading to the PLC. The Programming unit can be a dedicated device or a personal computer. It allows the graphical display of the program (ladder diagram). The unit, once connected to the PLC can download the program and allows for the real time monitoring of its operation to assist debugging. Once the program is found to operate as required the Programming Unit is disconnected from the PLC which continues the operation [8].

2.2.5 PLC Languages

The function of all programming languages is to allow the user to communicate with the programmable logic controller (PLC) via a programming device. They all convey to the system, by means of instructions, a basic control plan.

Ladder diagrams, function blocks, and the sequential function chart are the most common types of languages encountered in programmable controller system design. Ladder diagrams form the basic PLC languages, while function blocks and the sequential function charting are categorized as high-level languages. The basic programmable controller languages consist of a set of instructions that will perform the most common type of control functions like relay replacement, timing, counting, sequencing, and logic. However, the instruction set may be varied from one controller to another, because it depends on controller model, specification and requirements. It may be extended or enhanced to perform other basic operations [9].

The high-level languages have been brought about by a need to execute more powerful instructions that go beyond the simple timing, counting and ON/OFF control. High-level languages are used for analog control, data manipulation, reporting, and other functions that are not possible with the basic instruction sets. The

language used in a PC dictates the range of applications in which the controller can be applied. Depending on the size and capabilities of the controller, one or more languages may be used. Here are some typical combinations of the languages:

- a) Ladder diagrams only.
- b) Ladder diagrams and function blocks.
- c) Ladder and sequential function chart.
- d) Ladder, function blocks, sequential function chart.

2.2.6 Ladder Languages

Modern control systems still include relays, but these are rarely used for logic. A relay is a simple device that uses a magnetic field to control a switch. When a voltage is applied to the input coil, the resulting current creates a magnetic field. The magnetic field pulls a metal switch (or reed) towards it and the contacts touch, closing the switch. The contact that closes when the coil is energized is called normally open. The normally closed contacts touch when the input coil is not energized. Relays are normally drawn in schematic form using a circle to represent the input coil. The output contacts are shown with two parallel lines. Normally open contacts are shown as two lines, and will be open (non-conducting) when the input is not energized. Normally closed contacts are shown with two lines with a diagonal line through them. When the input coil is not energized the normally closed contacts will be closed (conducting) [6].

The ladder diagram language is a symbolic instruction set that is used to create a programmable controller program. Before the extension of the ladder language, the standard ladder instruction set was limited to performing only relay equivalent functions, using the basic relay-type contact and coil.

A necessary for greater flexibility, coupled with developments in technology, it is now extended to six sub-instruction sets and they are relay-type, timer/counter, data manipulation, arithmetic, data transfer, and program control. Desired control logic can be obtained by formatting the ladder instruction symbols and store into memory.

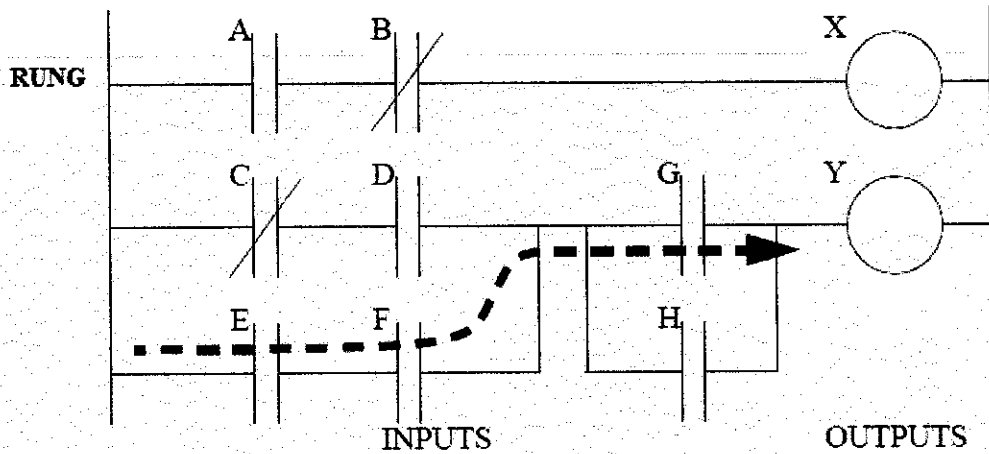


Figure 2-2: A simple Ladder Logic Diagram

The aim of ladder diagram program is to control outputs based on input conditions. Ladder rung is used for the control. Control rung, in general, consists of a set of input conditions represented by relay contact-type instructions and an output instruction at the end of the rung represented by the coil symbol. A basic structure of the ladder rung is shown in Figure 2-2. Coils and contacts are the basic symbols of the ladder diagram instruction set. The coil symbols represent all the outputs, where contacts represent conditions to be evaluated in order to determine the control of the output. Each coil and contact is referenced with an address number that is used to identify the operating conditions.

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

2.2.7 PLC Advantages

One of the main advantages of the PLC controller is that it is a programmable device, which makes it possible, unlike in the relay logic, to easily design and modify the control program or process without any changes in the wiring (no hardware modifications). To make the programming of the PLC systems easy and efficient, industry standards defining the programming approach and the programming languages used were adopted. This reduces the need for personnel training by making a set of languages standard for all PLC platforms on the market. Knowing the PLC programming languages and programming standards is thus one of the most important considerations for anyone involved in the PLC area [9].

PLC controllers are low cost, compact, versatile units based on the standard microprocessor architecture used in the control of machines or processes. They are designed for ease of programming and maintenance. The PLC systems replace the old relay logic control systems in automated manufacturing and are designed to provide easy and efficient replacements for the bulky relay logic controllers.

PLCs have been gaining popularity on the factory floor and will probably remain as a predominant for some time to come. Most of this is because of the advantages they offer.

- Cost effective for controlling complex systems.
- Flexible and can be reapplied to control other systems quickly and easily.
- Computational abilities allow more sophisticated control.
- Trouble shooting aids make programming easier and reduce downtime.
- Reliable components make these likely to operate for years before failure.

Although the PLC systems have many advantages, there are also disadvantages. These include fault finding, as PLC systems are often much more complex than the hard-wired relay systems. Secondly, the failure of the PLC may completely stop the controlled process, whereas a fault in a conventional control system would only disrupt the process. And thirdly, external electrical interference may disrupt the PLC memory.

PLC is a specialized device used to provide high-speed, low-level control of a process. It is programmed using ladder diagram (LD) or some form of structured language, so that engineers can program it. PLC hardware typically costs a lot of money, and often has very good redundancy and fail-over capabilities.

2.2.8 Timing Diagram

Timing diagrams are used to show changes and their relationship to clock times. It provides a visual representation of objects changing state and interacting over time. Then the Boolean Expressions for each output of the process were derived from the timing diagrams. It provides a visual representation of objects changing state and interacting over time. Timing diagrams can be used for defining hardware-driven or embedded software components [7].

Timing diagrams can be drawn from a value or time based point of view.

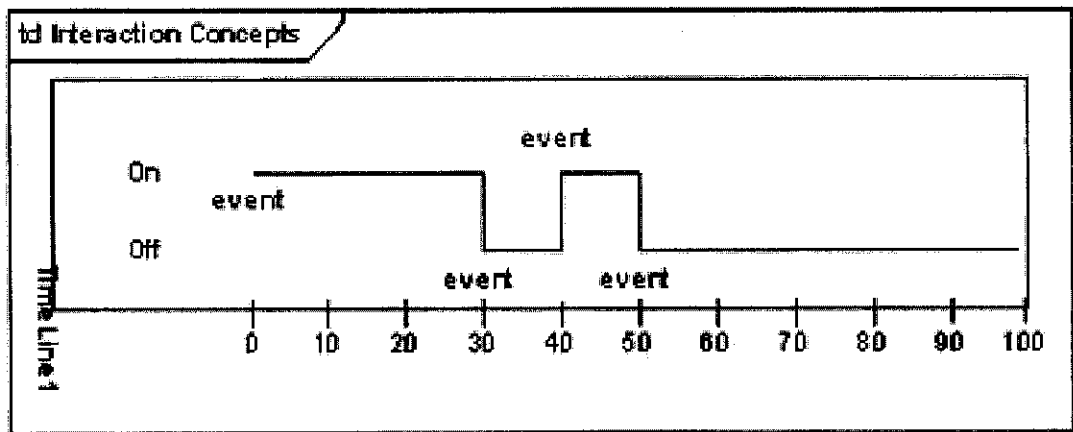


Figure 2-3: Time-based timing diagram

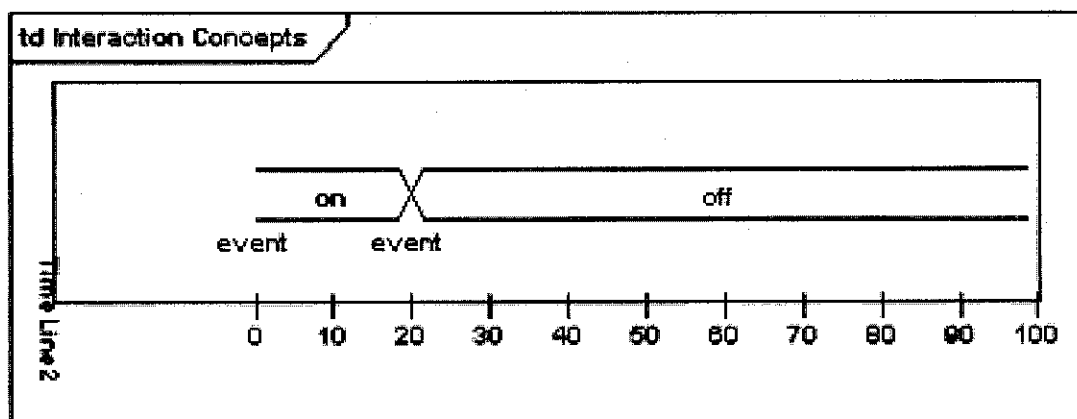


Figure 2-4: Value-based timing diagram

2.3 Relationship between PLC and Batch Process

One characteristic of batch process is its extensive use of analog data. Analog values can vary continuously within a specified range. The analog data may be representative of temperature, pressure, rate of flow, weight, thickness, viscosity, humidity, or any other characteristic of importance to the process. Both batch and continuous processes require continuous monitoring at numerous points throughout the process. In addition, a corrective action is often required to insure that the process stays within specifications. This type of control that involves measuring a value, comparing the measured value to a desired value or set point, and correcting for the error is called closed-loop control.

A variety of approaches can be used for process control depending upon the complexity of the process being controlled. A small batch process often lends itself well to control by one PLC or a few networked PLCs. A representation of the process showing its current status and a history of data recorded at various points in the process is often provided to the PLCs.

CHAPTER 3

METHODOLOGY/ PROJECT WORK

3.1 Project Work

Studies and reviews were conducted to increase the understanding on batch process and Programmable Logic Controller (PLC). These are important to help in the program development.

In PLC programming, the development of the flowcharts and timing diagrams are the first step to do before writing the ladder diagram program. The Boolean expressions were derived from the timing diagram and these will be used in the designing the ladder diagram. It is important to undergo a troubleshooting process before any program can be downloaded to the PLC.

The hardware connection between the PLC, computer station, and the inputs and outputs and the CPU unit was conducted and this is followed by the troubleshooting process to eliminate any errors in the logic sequences.

Any errors in the output that were not displayed during the execution, indicates that the programming encountered some problems. Thus, the program needs to be modified to obtain the desired results. Below is the flow chart describing the project workflow:

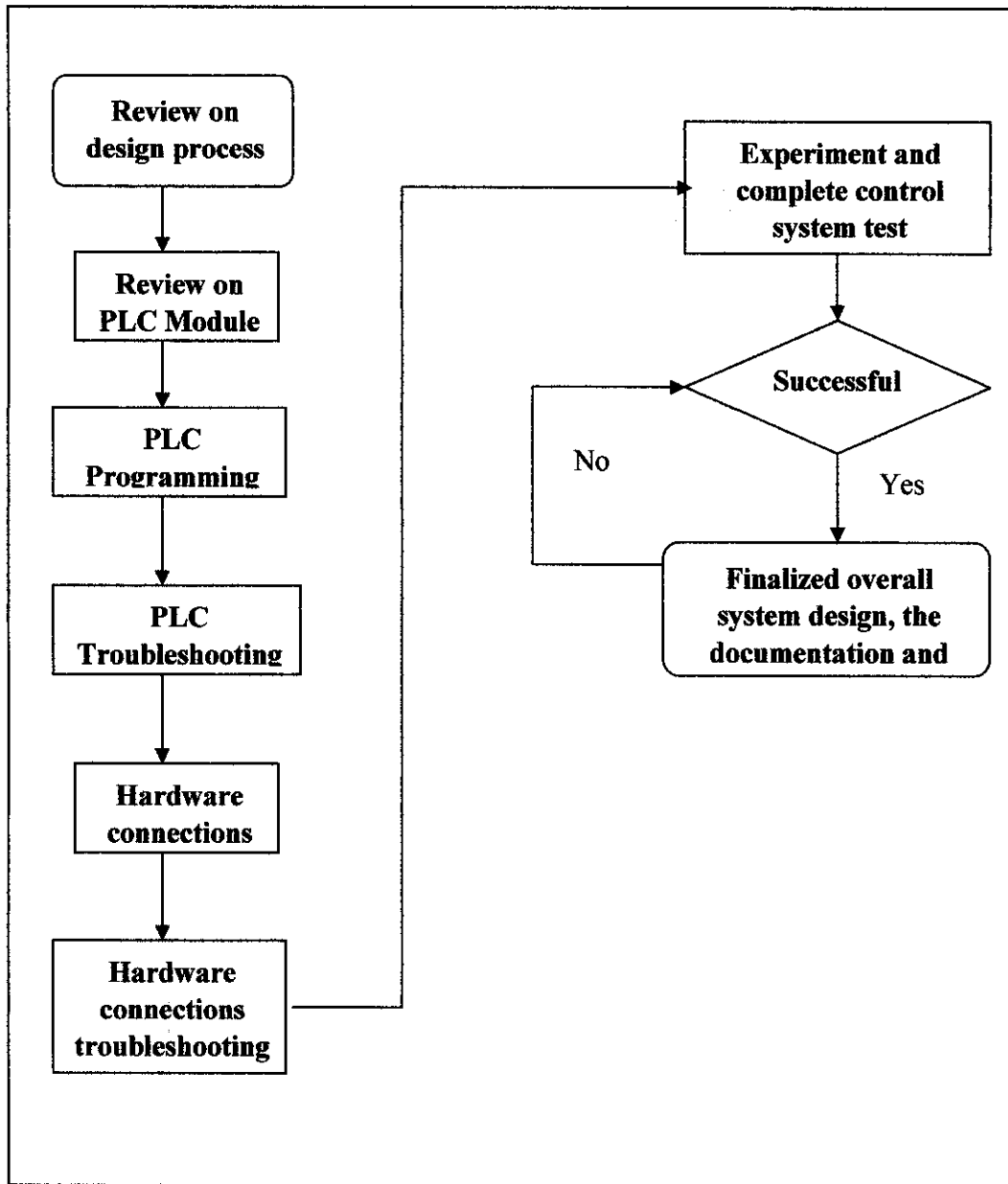


Figure 3-1: Project Workflow

3.2 Studies on Liquid Soap Making Process

Liquid soap making process applies the batch process control [10]. Recipe is a set of information that uniquely defines the production requirements for a specific product.

The making process involves a sequential processing that leads to the production of finite quantities of liquid soap by subjecting quantities of the raw materials to an ordered set of processing activities such as weighing, heating and mixing over a finite period of time which is set using the timer with one or more pieces of equipment.

The raw materials used in making the liquid soap are canola oil, palm oil, raw water, borax, kasturi oil and potassium (*refer Appendix B, Figure 1*). Below are the process sequences of the liquid soap making process:

- | Step | Action |
|------|---|
| 1 | Press push button PB001. |
| 2 | Start the heater 1 (H001) and heat up the distilled water for one hour duration of time with control of 75°C of temperature. Delay for 1 hour duration. |
| 3 | Start pump 1 (P001) and open control valve 1 (CV001) to 50% opening to circulate the heated distilled water from tank 1 (T001) to the outside layer of heating tank 6 (T006). |
| 4 | Open solenoid valve 2(SV002) to let the potassium solution of tank 2 (T002) flow to the weighing tank 5(T005). Wait . The inlet is measured by the load cell (LC001, LC002 and LC003) until it reach the set point. |
| 5 | Close SV002. Start agitator 1 (A001) to agitate the potassium solution. |
| 6 | Open solenoid valve 3 (SV003) with timer TIM001 to let the palm oil flow from T003 to T006. Close SV003 when TIM001 is off. |
| 7 | Open solenoid valve 4(SV004) with timer TIM002 to let the coconut oil flow from T004 to T006. Close SV004 when TIM002 is off. |
| 8 | Start agitator 2 (A002) to agitate the mix material. Delay the mixing process for 20 minutes. |
| 9 | Stop A001 and Open solenoid valve 5 (SV005). |
| 10 | Start pump 2 (P002) to pump out the potassium solution from T005 to T006. Wait . Low level sensor 1 (LS001) will sense the low level of the material in T005 to start on the timer TIM003. |
| 11 | Close SV005 and stop P002 when TIM003 is off. Delay for 1 hour. |
| 12 | Open solenoid valve 1 (SV001) to let the ratio of distilled water flow from T001 to T005. Wait . The inlet is measured by the load cell (LC001, LC002 |

and LC003) until it reach the set point.

- 13 **Close SV001. Open SV005 and start P002 to pump out the distilled water from T005 to T006. Wait.** LS001 will sense the low level of the material in T005 to start on the timer TIM004.
- 14 **Close SV005 when TIM004 is off. Delay** the mixing process for 15 minutes.
- 15 **Start Dosing Pump (DP001) and purge out the Borax from T007. Wait.** The pH sensor (pH001) will sense the mixture with the range of 8 to 9 pH level of neutralization.
- 16 **Stop DP001 and start dosing pump 2 (DP002) with timer TIM005 to purge out the extract of Kasturi as a fragrance. Stop DP002 when TIM005 off.**
- 17 **Delay** the mixing process for 5 minutes. **Stop A002.**
- 18 **Open control valve (CV006) to 30% opening to drain out the complete mixture from T006 to product tank (T009). Wait.** LS002 will sense the low level of the material in T006 to start on the timer TIM006.
- 19 **Fully Closed SV006.**

3.3 Analogue Module

This project focused on an approach to the implementation of analogue module of PLC for a batch process. Firstly, the input and output ranges and the number of inputs and outputs required for the process were determined. Then, both analogue inputs and outputs were wired based on the PC setup setting. After the wiring process had been completed, the process flow was programmed into ladder diagram. The understanding on analogue modules is important and was enhanced by following the application procedure below:

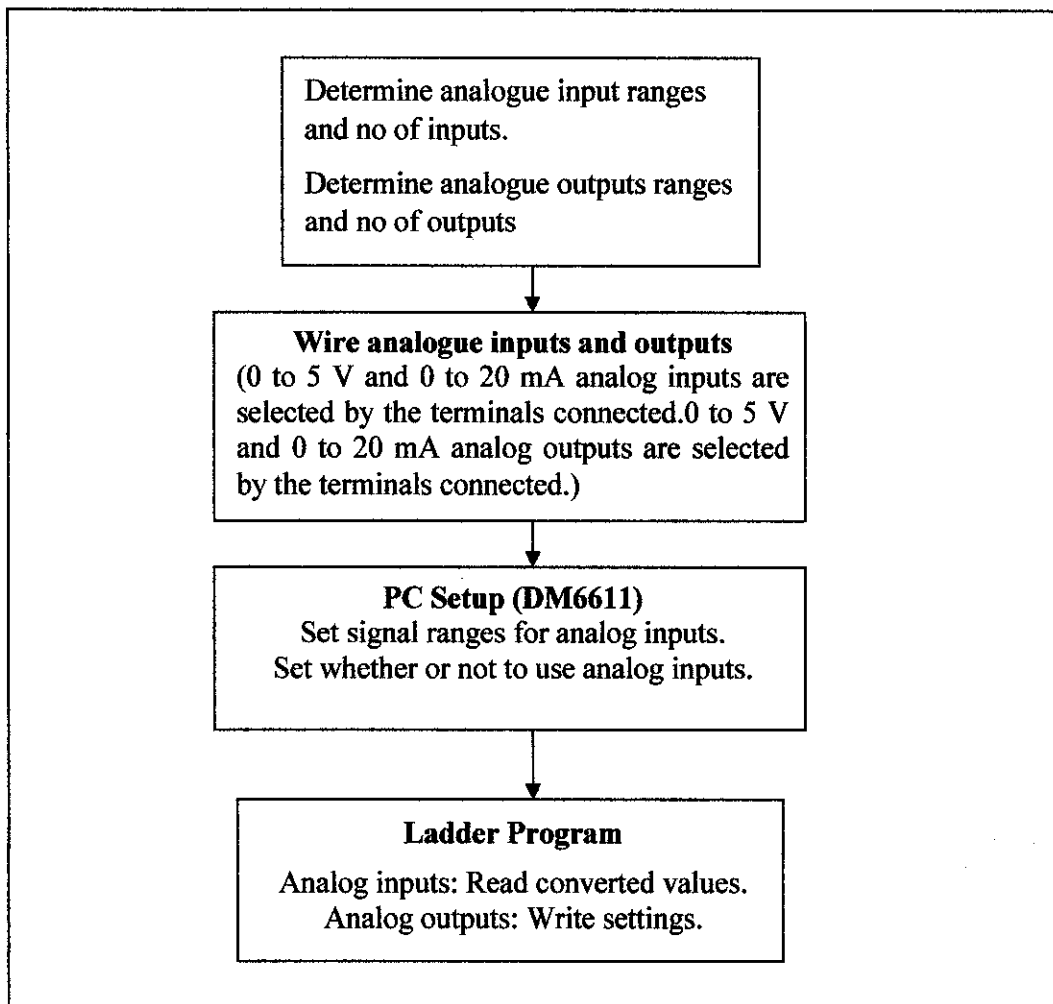


Figure 3-2: Application procedures of analogue module

The Analog I/O Board is an Inner Board featuring four analog inputs and two analog outputs as shown in Figure 3-4. The signal ranges that can be used for each of the four analog input points are -10 to $+10$ V, 0 to 5 V, and 0 to 20 mA. A separate range is set for each point.

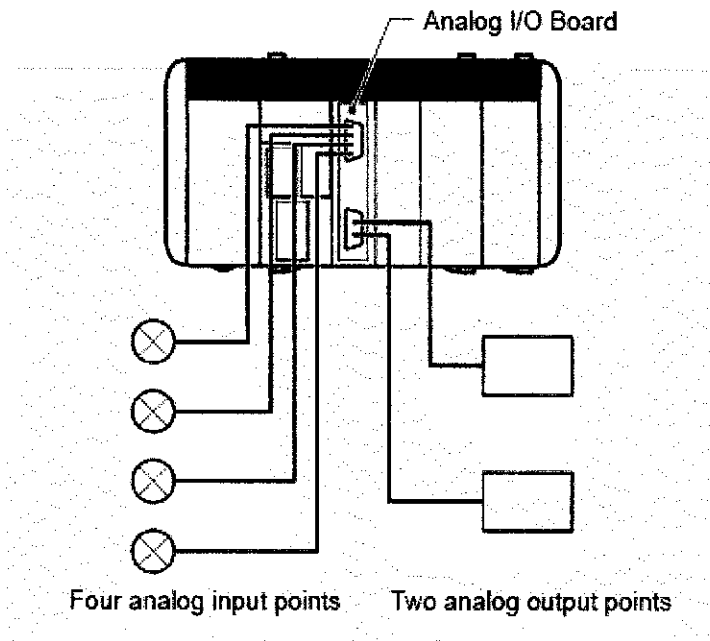


Figure 3-3: Analogue module configuration

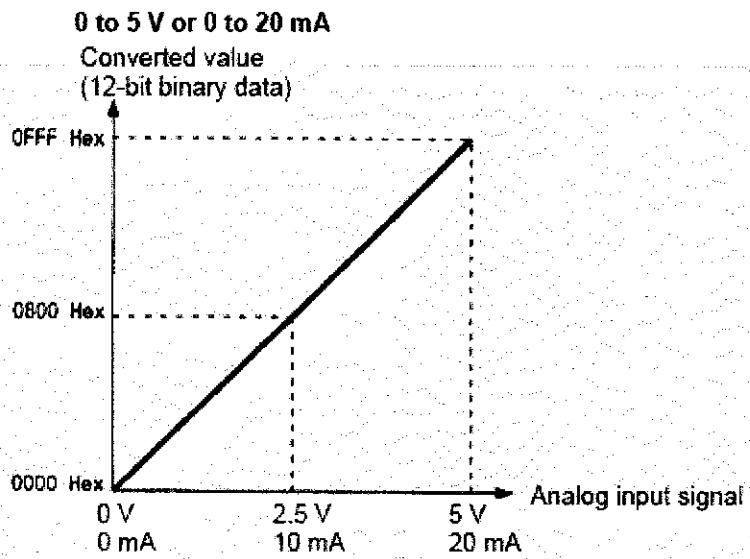


Figure 3-4: Analogue inputs data and converted values specification

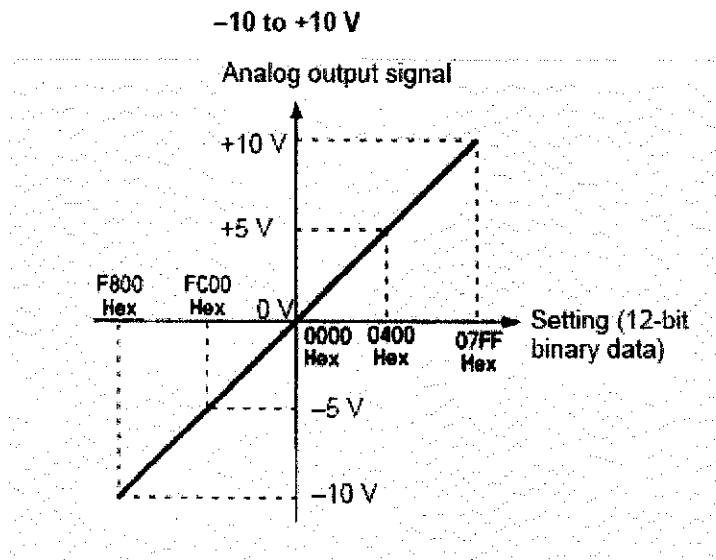


Figure 3-5: Analogue settings and output data specification

The settings in DM 6611 determine the signal ranges. The signal ranges that can be used for each of the two analog output points are -10 to $+10$ V and 0 to 20 mA. A separate signal range can be selected for each point. Signal of 0 to 20 mA was selected as the input signal range while 1 to 5 V for the output signals range.

In this project 0 to 20 mA of input signal was used for the input signal range. The setting of DM 6611 was set in the memory window based on Table 3-1 that shows the way to assign the bit of DM 6611 [4].

Table 3-1: DM 6611 PC Setup Setting

Word	Bits	Functions	
DM 6611	00 to 07	00, 01: Analog input 1 input signal range	00: -10 to $+10$ V 01: 0 to 10 V
		02, 03: Analog input 2 input signal range	10: 0 to 5 V/ 0 to 20 mA
		04, 05: Analog input 3	11: Not used.
			(0 to 20 mA are

		input signal range 06, 07: Analog input 4 input signal range	distinguished by the connected terminal.)
	08	Analog input 1 usage selection	Specifies use or non-use of A/D conversion for each port. 0: Use input (conversion) 1: Do not use input (no conversion)
	09	Analog input 2 usage selection	
	10	Analog input 3 usage selection	
	11	Analog input 4 usage selection	
	12 to 15	Not used. (Fixed at 0.)	

3.4 A Systematic Approach to Programmable Controller Design

The concept of controlling a control system involves a systematic approach, and the operation procedure includes the determination of the machine sequence, inputs and outputs addressing, and the PLC programming.

3.4.1 *The machine sequence of operation*

Firstly, the process system or the recipe to be controlled was studied. Recipe is a set of information that uniquely defines the production requirements for a specific product. The ultimate purpose of the programmable logic controller is to control an external system. The system to be controlled can be machine equipment, a process and is often generically called the control system. The movement of the control system is constantly monitored by the input devices that give a specified condition and send a signal to the external output devices which actually controls the movement of the control system as specified and thus achieved the extended control action. A flowchart of the recipe was developed to simplify the control sequences. From the flowchart, a timing diagram was developed and the Boolean expressions were derived out from it. The expressions were used to develop the ladder diagram.

3.4.2 Inputs and Outputs Addressing

All external input and output devices were identified and assigned to the different address that corresponds to the inputs and outputs. The inputs and outputs used in the project were limited to the amount that can be provided by the PLC which is 4 analogue inputs, 2 analogue outputs and 16 digital inputs and outputs, respectively,. The configuration on the wiring between the PLC and the devices was carried out before developing the ladder diagram, since the address dictates the precise meaning of the contacts in the ladder diagram and need to be specified.

3.4.3 PLC Programming

The process was programmed into ladder diagram by using the CX-Programmer tool. The completed program was transferred into the memory of the programmable controller and the simulations were done on the programming to ensure the process designed work properly and validate the system behavior. Then, the program is ready to be implemented.

3.5 Troubleshooting

Troubleshooting is required to ensure that the program can run in accordance to the intended process sequences and give the desired response of the control valves which were connected to the PLC. The process is also to correct any error occurred during the accomplishment of the program.

3.6 Hardware connection

Once the program has been developed and tested, the implementation of the system can be realized. The connection between the inputs and the control valves was justified and set up for the program to be implemented. The program then was run to observe the response of the control valve on the control system designed.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Process Sequences

Liquid soap making process applies the batch process control. The push button used as the start-up of the sequences. The heater is used to heat the distilled water for an hour before further process can take place. The heating process is important for the process to be efficient. Pumps are used to pump out materials from one tank to another tank while agitators are used to mix the materials constantly. The load cells that are attached to Tank 5 are to measure the weight of certain materials that are fed into the tank. The desired level of materials or mixture is indicated by the level sensors. Material borax is used to change the pH of the mixture (liquid soap) while small amount of fragrance is adequate to bring up the smell. Thus, dosing pumps are used in the process to control the amount of the borax and fragrance (kasturi oil) by purging out the materials in a small amount. The process sequences of the liquid soap making were simplified into flowchart for an easy understanding as shown in Figure 4-1.

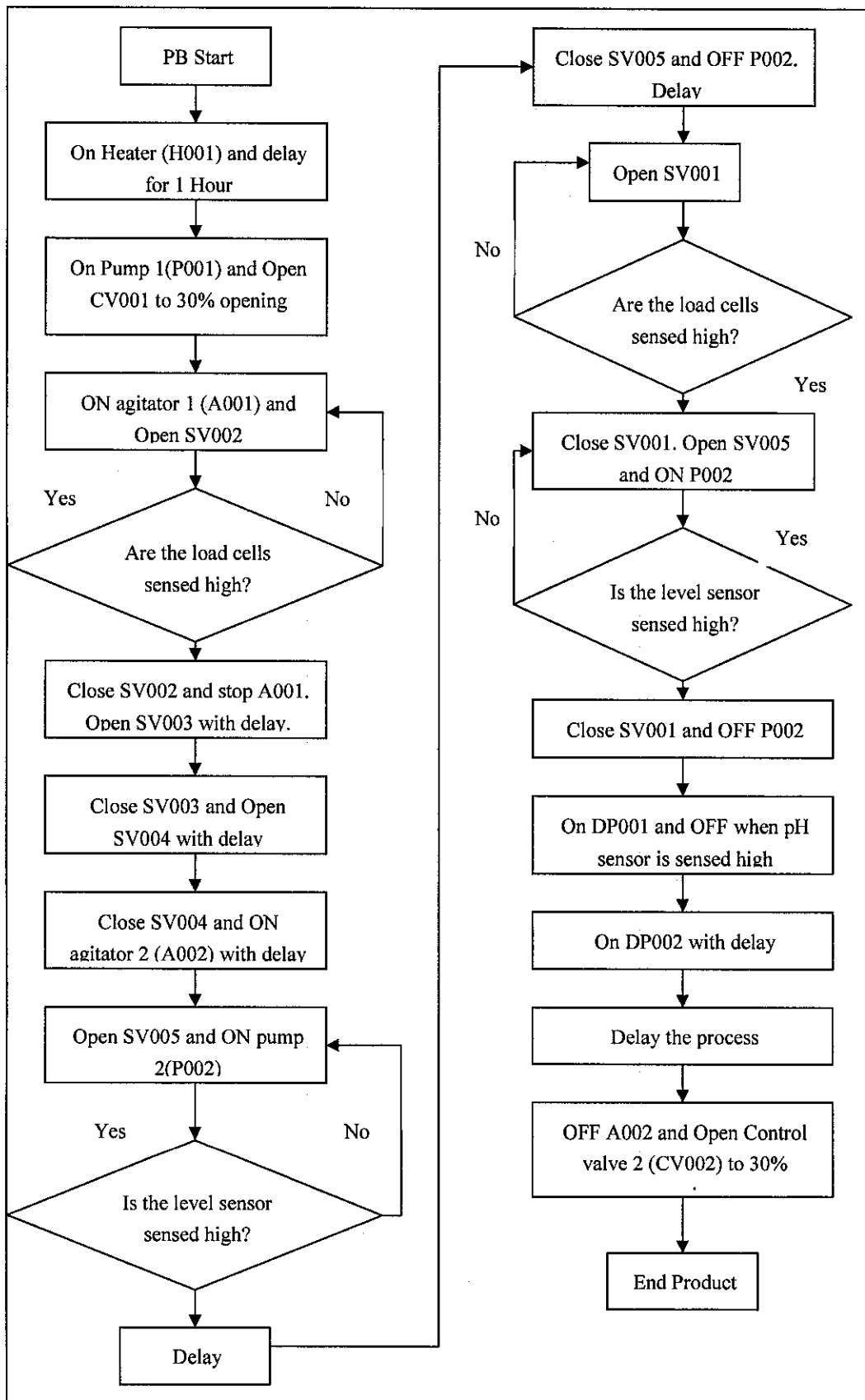


Figure 4-1: Process Sequences Flowchart

4.2 Inputs and Outputs Addressing

There are 4 digital inputs, 12 digital outputs, 2 analogue inputs and 2 analogue outputs that were used in the programming design which were based on the process sequences. Basically, the control valve working standard is between 1V to 5V. 1V corresponds to 0 % opening (fully closed) while 5V indicates that the valve is 100% opening (fully opened). Normally, with digital control, the valve is limited to fully open and closed. However with analogue control, the valve can be regulated in the range of 0% to 100% opening. As in the project, two control valves are used and regulated to 30% opening for the first control valve and 50% opening for the second one. Tables below show the addressing of both digital and analogue input and output signals:

Table 4-1: Digital inputs addressing

Inputs	Address
Push Button	000
Load Cell	001
Level Sensor 1	003
Level Sensor 2	005

Table 4-2: Digital outputs addressing

Outputs	Address
Heater	100
Pump 1	102
Pump 2	113
Valve 1	103
Valve 2	104
Valve 3	106

Valve 4	108
Valve 5	112
Agitator 1	105
Agitator 2	110
Dosing Pump 1	101
Dosing Pump 2	107

Table 4-3: Analog input addressing

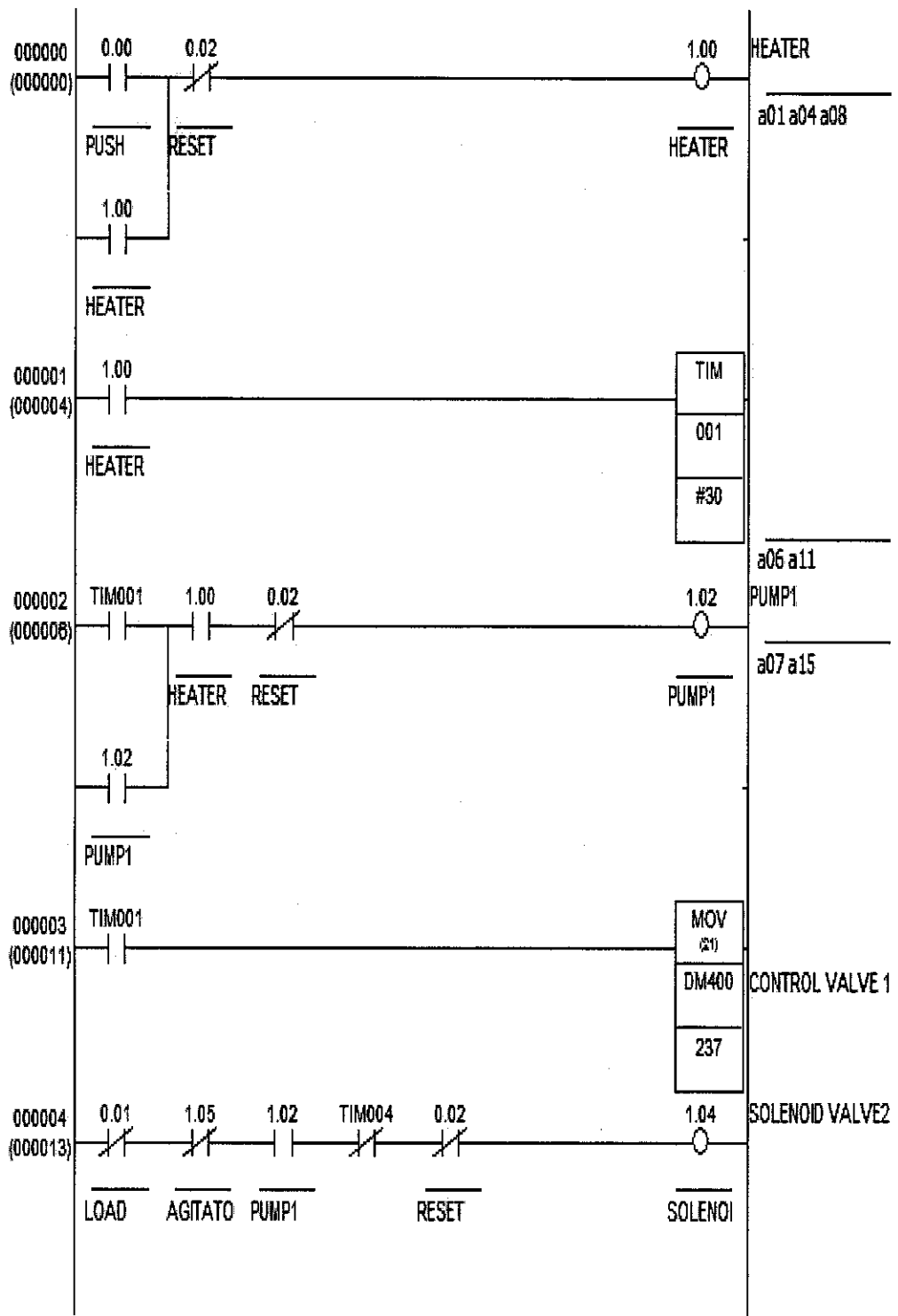
Inputs	Address
DM400	IR 232
DM401	IR 233

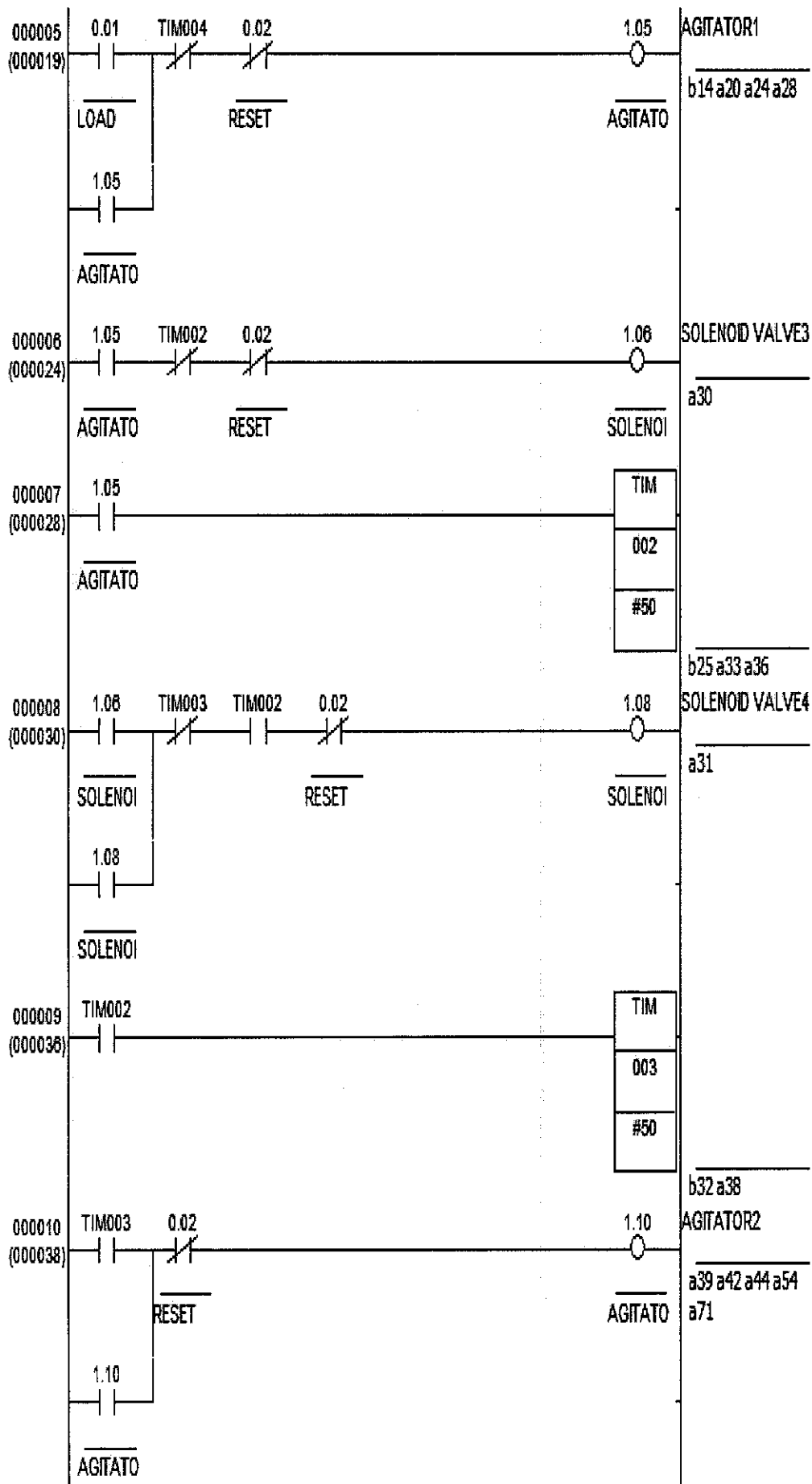
Table 4-4: Analog output addressing

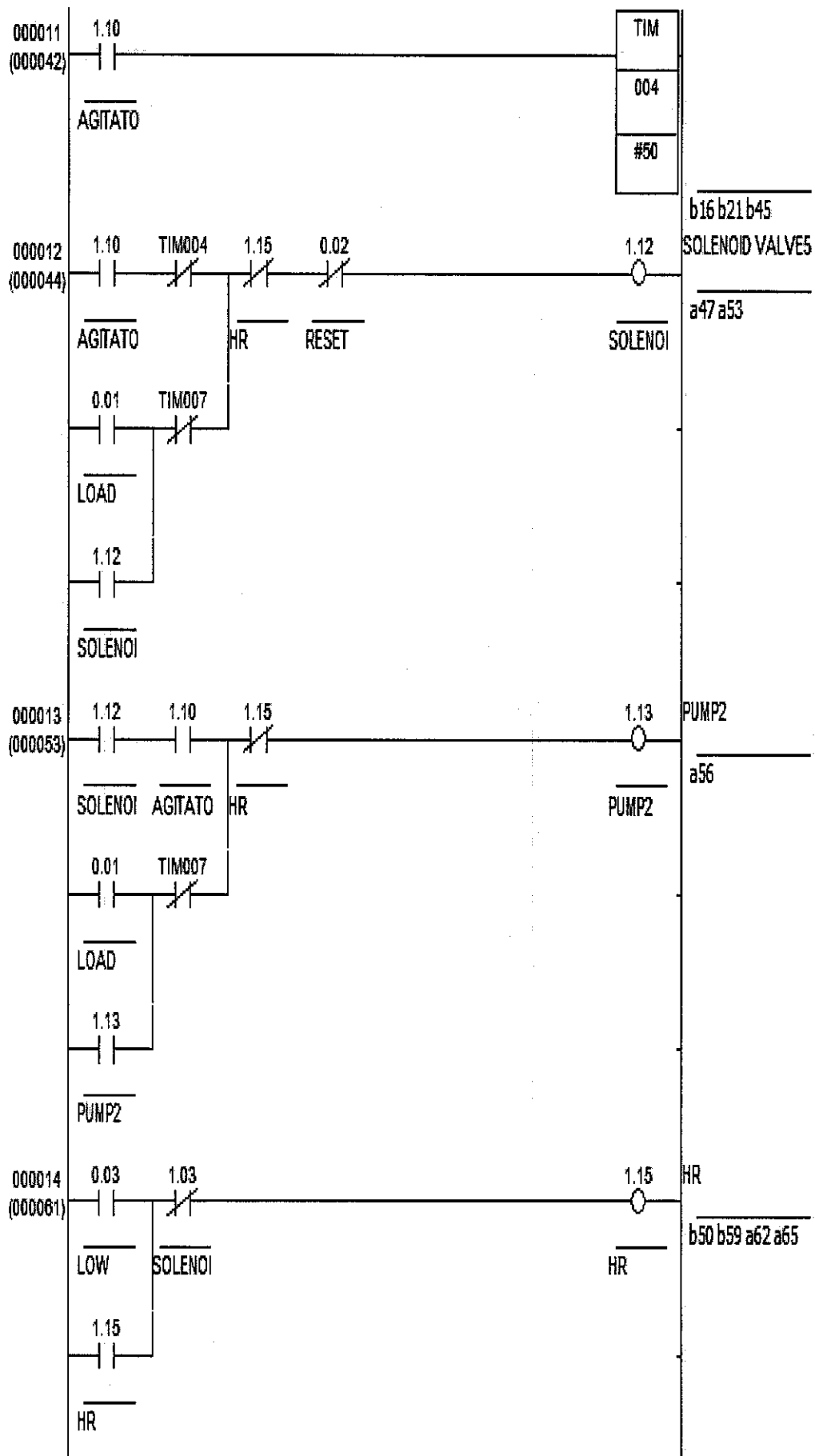
Outputs	Address
Control Valve 1	IR 236
Control Valve 2	IR 237

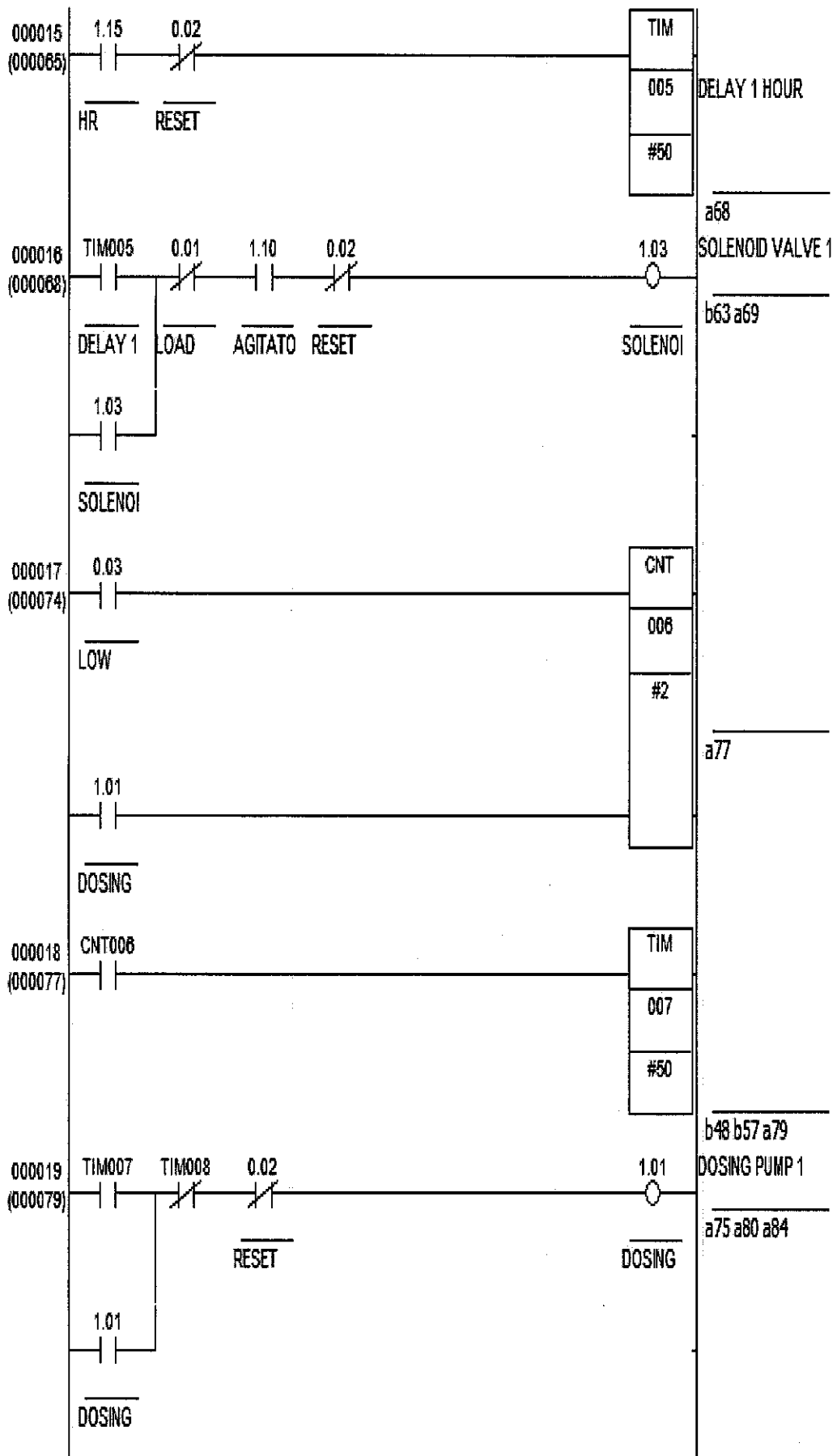
4.3 Ladder Diagram

The instructions that are normally applied for the digital control are Contact, Closed Contact, Timer, and Counter. For the analogue controller, the Move instruction was used to regulate the opening of the control valve. The opening of the valve was determined through the input value that was set in the DM memory of the PLC that is DM 400 and DM 401 for control valve 1 and 2 respectively. The developed ladder diagram can be seen in Figure 4-2:









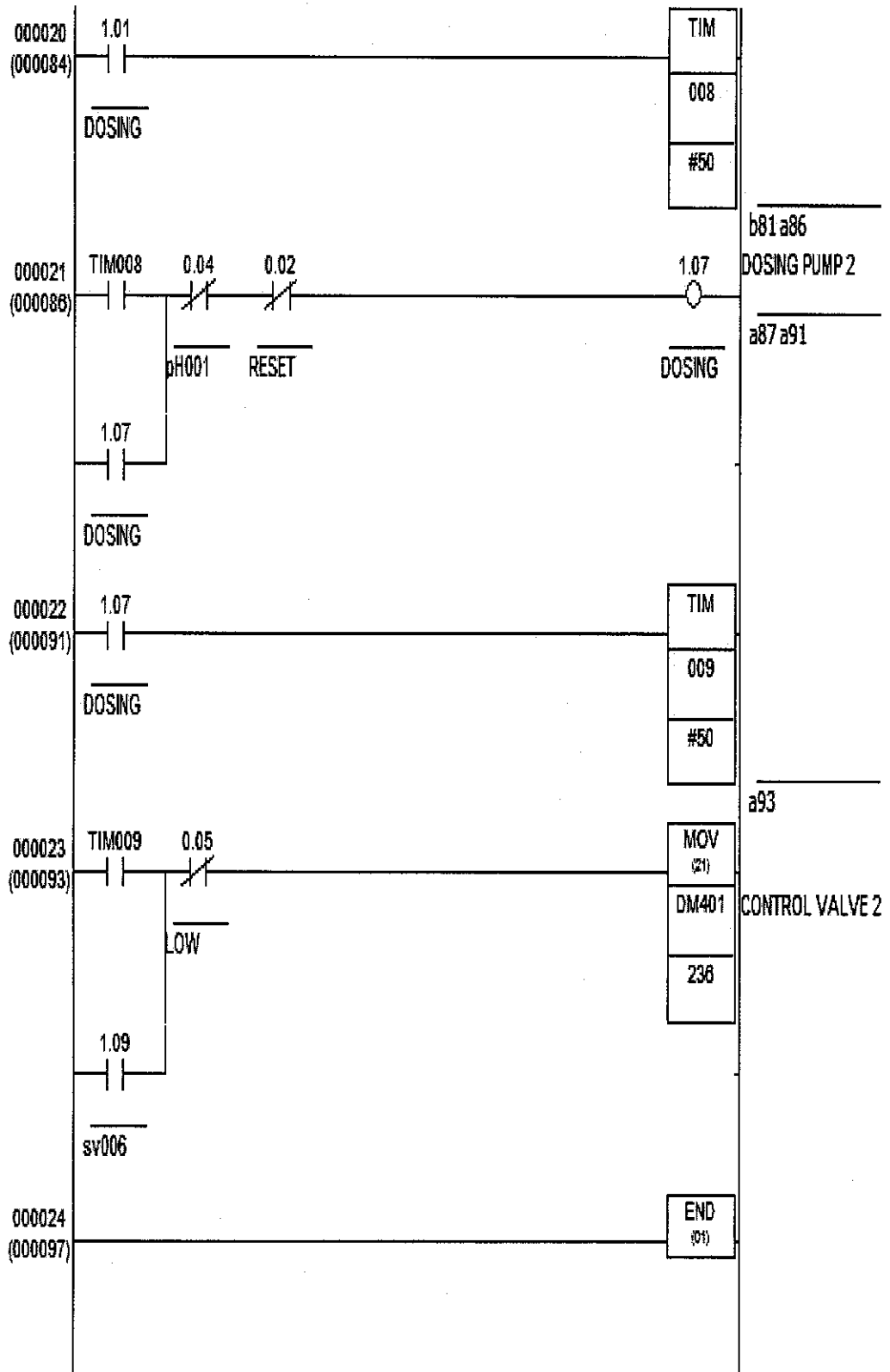


Figure 4-2: Process ladder diagram programming

CHAPTER 5

CONCLUSION AND RECOMMENDATION

Literature reviews and studies have been conducted to increase the understanding of batch process control and Programmable Logic Controller (PLC). Batch process involves sequences process that leads to the production of finite quantities of material by subjecting quantities of input materials to an ordered set of processing activities over a finite period of time using one or more pieces of equipment. A PLC consists of a central processing unit (CPU) containing an application program and input and output interface modules, which are directly, connected to the field I/O devices. The software used was the CX-Programmer which is installed into the computer and linked to the PLC. The program developed was transferred into PLC and tested to observe that the outputs obtained are like the one that are desired.

The CX-Programmer programming software was used to develop the ladder diagram for the PLC, implementing the batch control. The concept of controlling a control system involves a systematic approach, and the operation procedure includes the determination of the machine sequence, inputs and outputs addressing, and the PLC programming. Troubleshooting is conducted to ensure that the program run in accordance to the intended process sequences and give the desired response of the control valves, the final control elements connected to the PLC. This approach is found to be effective in designing a logic ladder diagram for a PLC to control a batch process, in this case a liquid soap making process

The project is recommended to be implemented by using another family of PLC such as Allen- Bradley and Siemens to expose to a different tool and skills of ladder diagram programming.

REFERENCES

1. A.E. Nisenfield. *Practical Guides for Measurement and Control*, Instrument Society of America, 1996
2. Fred Swainston. *A Systems Approach to Programmable Controllers*. Delmar Publishers Inc, 1992.
3. Jon Stenerson. *Fundamental of PLC, Sensors, and Communication*. Pearson, Prentice Hall, 3rd Edition, 2000
4. *SYSMAC CQM1H Series Programming Manual*, Omron, Revised May 2000
5. A disturbance management approach to improving the performance of batch process operations
<http://www.repository.tudelft.nl/file/80684/007475>
6. Batch and Continuous Process
http://www.sea.siemens.com/step/pdfs/ep_5.pdf#search='batch%20and%20continuous%20process'
7. Francois Coetzee. Xpdian. *Timing Diagram*
<http://www.xpdian.com/TheTimingDiagram.html>
8. PROGRAMMABLE LOGIC CONTROLLERS (PDF)
http://claymore.engineer.gvsu.edu/~jackh/books/plcs/chapters/plc_intro.pdf
9. PLC Structure
<http://kernow.curtin.edu.au/www/plc/2STRUCT.HTM#PLCCOM>
10. Understanding Liquid Soap Making
<http://essencesupply.com/liquid/page3.html>

APPENDICES

APPENDIX A

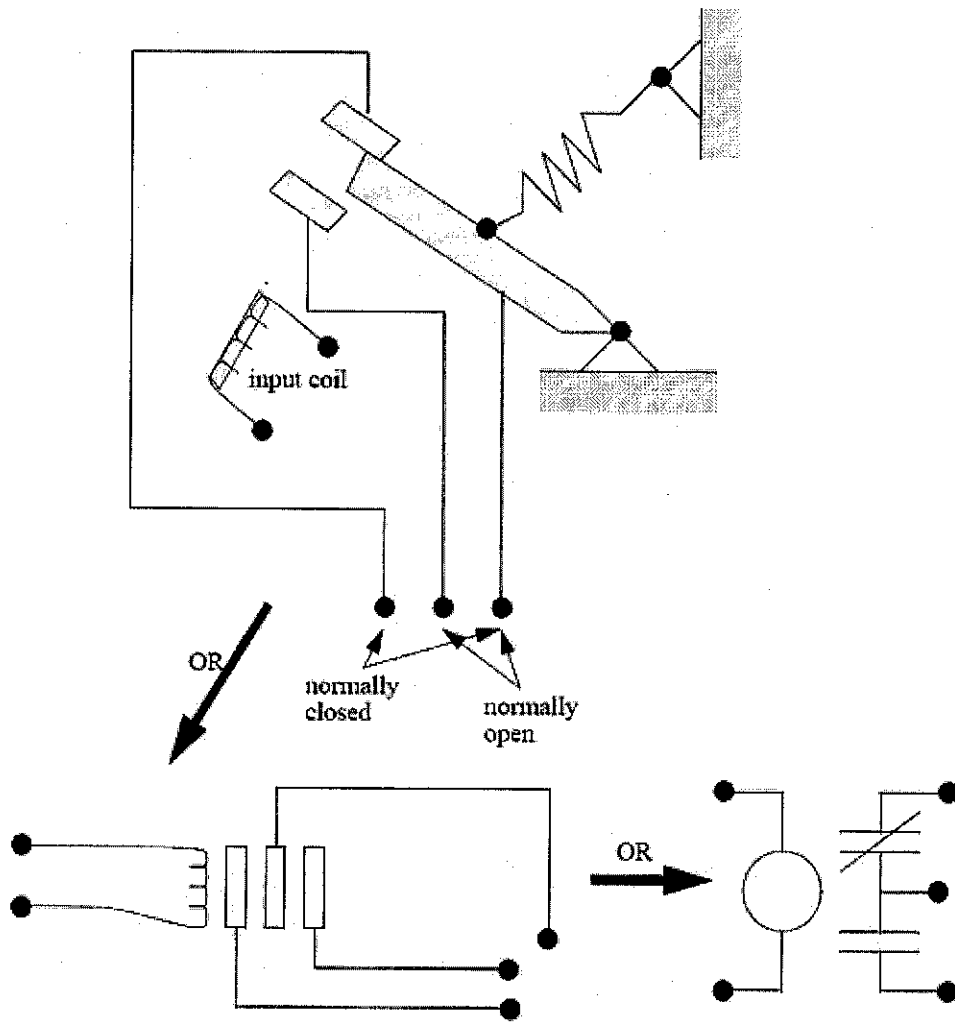


Figure 1: Relay Logic

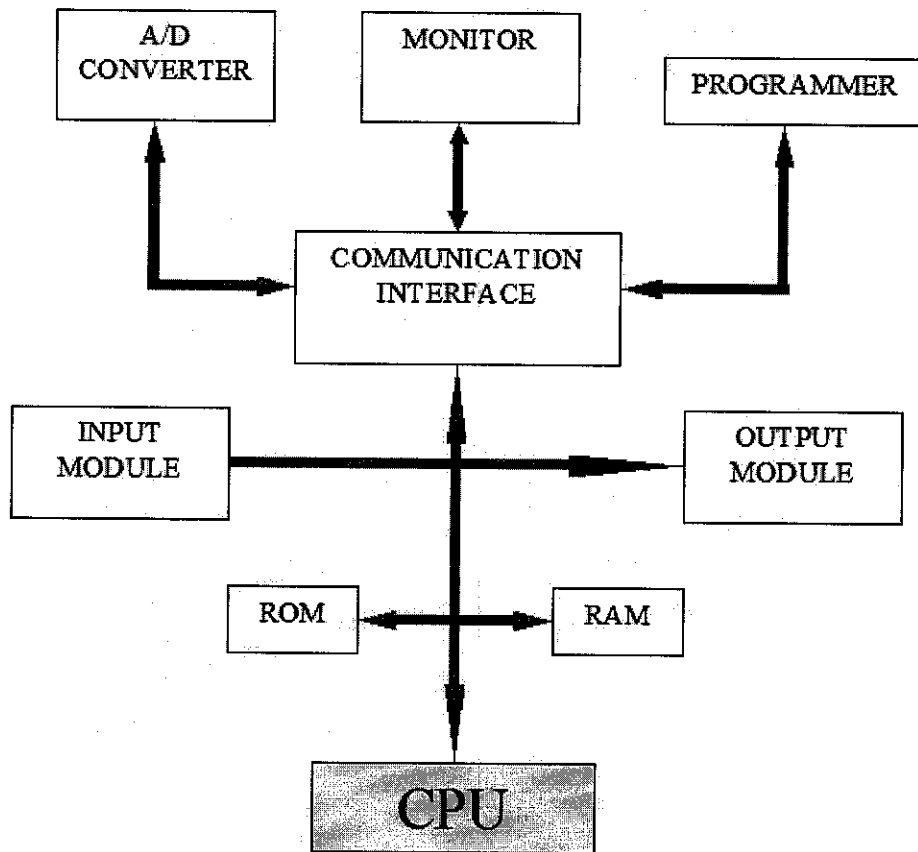
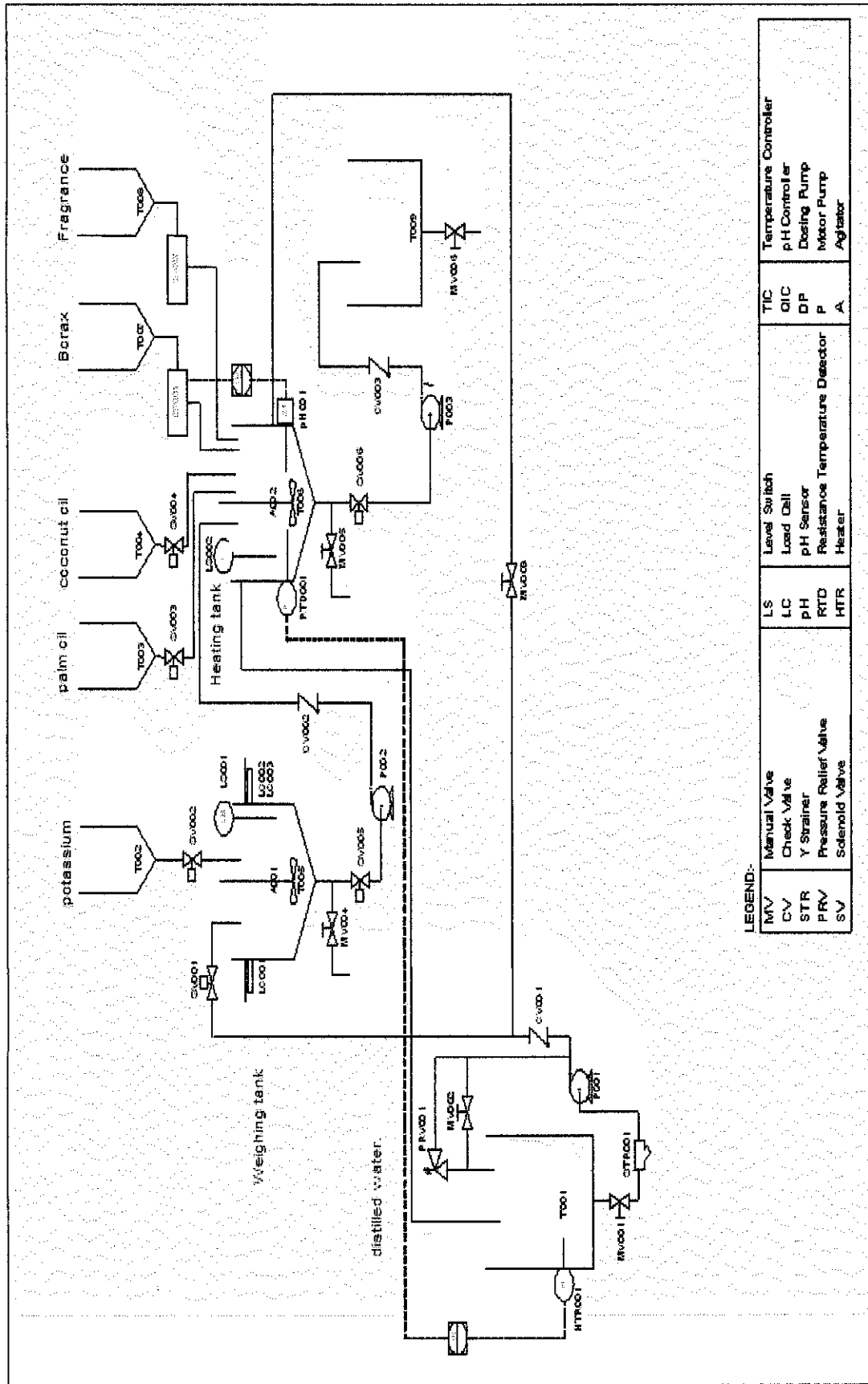


Figure 2: PLC Structure

APPENDIX B



LEGEND:-

MV	Manual Valve	LS	Level Switch	TIC	Temperature Controller
CV	Check Valve	LC	Level Cell	QIC	pH Controller
STR	Y Strainer	pH	pH Sensor	DP	Dosing Pump
PRV	Pressure Relief Valve	RTD	Resistance Temperature Detector	P	Motor Pump
SV	Solenoid Valve	HTR	Heater	A	Agitator

Figure 1: P&ID of the liquid soap making process