MOHD ZULHILMI BIN ZAKARIA
B010410105

PROGRAMMABLE LOGIC CONTROLLER (PLC) BASED INDUCTION MOTOR STARTER

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACULTY OF ELECTRICAL ENGINEERING

PROGRAMMABLE LOGIC CONTROLLER (PLC) BASED INDUCTION MOTOR STARTER

MOHD ZULHILMI BIN ZAKARIA
B010410105

23 APRIL 2008

"I hereby declare that I have read through this report and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Electrical Engineering (Industrial Power)"

Signature : ....

Supervisor's Name : Musse Mohamud Ahmed

Date : 07 May 2008

PROGRAMMABLE LOGIC CONTROLLER (PLC) BASED INDUCTION MOTOR STARTER

MOHD ZULHILMI BIN ZAKARIA

This Report Is submitted In Partial Fulfillment Of Requirement For The Degree of Bachelor In Electrical Engineering (Industrial Power)

Fakulti Kejuruteraan Elektrik
Universiti Teknikal Malaysia Melaka

APRIL 2008

" I hereby declare that this report is a result of my own work except for the experts that have been cited clearly in the reference."

Signature : ....

Name : MOHD ZULHILMI BIN ZAKARIA

Date : 7 MAY 2008

ABSTRACT

This project is about to design an Induction Motor Starter using Programmable Logic Controller (PLC). The main objective is to develop one device that can control motor and can be monitored in one monitor. PLC is a controller which uses digital computer and is used in automation for machines. The PLC is designed for multiple inputs and output arrangements so that it can monitor using one monitor. This project will design one induction motor starter and can be monitored from one place and also can be changed the setting of the induction motor. In other words, the setting or programming can be changed only from control room and no need to change at the connection. It will be used for a 3-phase induction motor using star-delta connection. This starter can recognize the faults and after time delay and the motor will not switch to delta connection, so that can avoid any damage. Using PLC is easy to make troubleshooting and also no need to use many hard wirings.

ABSTRAK

Projek ini adalah untuk mereka bentuk satu projek iaitu 'Induction Motor Starter using Programmable Logic Controller (PLC)'. Matlamat utama adalah untuk membuat satu peranti yang boleh mengawal dalam satu system kawalan PLC direka untuk mengawal mesin- mesin mengunakan computer. PLC direka dengan dengan beberapa input dan juga output supaya dapat memantau dalam satu kawalan. Projek ini direka untuk mengawal pemulaan operasi motor aruhan dari satu tempat. Dengan mengunakan PLC, mudah untuk menyelengara dan juga dapat mengurangkan mengunakn kabel yang banyak

ACKNOWLEDGEMENT

Firstly, I am Thankful to Allah who has given me a chance to complete this subject and also my formal report for Projek Sarjana Muda 2 (PSM2) in this semester.

Secondly I want to thank to the Universiti Teknikal Malaysia Melaka (UTeM) especially Faculty of Electrical Engineering (FKE) for giving me an opportunity to do my Projek Sarjana Muda 2(PSM 2) in fulfillment for Bachelor of Electrical Engineering.

I am also thankful to my supervisor, Dr Musse Mohamud Ahmed for the advices he has been give to me. Without his guidance, maybe my project can't be completed on time.

Lastly I want to thank to my family because they always support me. I am also thankful to my coordinator, all lectures and my friends who helped me in this report. I hope what I have learned here will help me in my studies and also in the future. I also hope that this report will be guidelines for all future students and will get good judgment from all sides.

MOHD ZULHILMI BIN ZAKARIA
Bachelor in Electrical Engineering (Industrial Power)
Faculty of Electrical Engineering
Universiti Teknikal Malaysia Melaka

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURE

LIST OF APPENDIX

CHAPTER 1

INTRODUCTION

1.1 Overview

A Programmable Logic Controller which is usually called a PLC or more commonly, simply a programmable controller is a solid-state, digital and industrial computer. The programmable logic controller (PLC) or programmable controller can be classified as a solid-state member of the computer family. A programmable controller is an industrial computer in which control devices such as limit switches, push buttons, sensors, or in other mean provide incoming control signal into the unit.

In this project, a PLC is used for start induction motor using star-delta connection. Most of the connection using hardwires and in this project, a PLC will be use to changed from hardwires connection.

1.2 Objectives

The objectives of this project are to design, develop and make interface the PLC with the hardware. The Programmable Logic Controller (PLC) is known for controlling devices and easy for their troubleshooting when used.

The first objective is to understand the operation of the PLC before the project can get started. The operation of each PLC is different depending on their model and brand. This is because the programming, the symbol and the characteristic is different from application to application.

The second objective of this project is to design the circuit and also to do the programming using PLC for the induction motor starter. The programming is the main part of this project because if the programming cannot run or has a problem, so the hardware also cannot run. After the circuit has been designed, the programming will be constructed based on the circuit diagram before interfacing to the hardware to simulate.

Then, the next objective is that the hardware of the design will be built exactly similar to the circuit diagram that has been designed. If the components and elements will be carefully chosen in this stage then the hardware will implemented with the circuit diagram, so that we will be able to interface the PLC with the hardware and make the simulation. Both of the simulation and the hardware results will be compared so that we will be able to recognize the actual error that occurs which affects the output results that cannot be detected during simulation.

The final objective of this project is to come up with an automated 3-phase induction motor using PLC programming and its simulation results.

1.3 Problem Statements

Most of 3 phase induction motors in industrial use hard wire for the connections of each motor. The reason of using hard wire is because of the low cost of installation comparing with using PLC. The weakness when using hard wire is very hard to make troubleshooting and do maintenance. When having some problems, it will take some time to find the fault. This is because the troubleshooting must start form the beginning that is from the source of the motor until at the ending of the connection at the motor. This will take long time to find the fault and to fix it. Since using hard wire connections have some problems, the PLC can make the troubleshooting and maintenance becomes easier for the 3-phase induction motors. So the problem and fault will be easy to find and take that a short time to fix it.

CHAPTER 2

LITERATURE REVIEW

2.1 Programmable Logic Controller

Programmable logic controllers, also called programmable controllers or *PLC*, are solid-state members of the computer family, using integrated circuits instead of electromechanical devices to implement control functions. They are capable of storing instructions, such as sequencing, timing, counting, arithmetic, data manipulation, and communication, to control industrial machines and processes. [1] In the late 1960's an American company named Bedford Associates released a computing device they called the *MODICON*. As an acronym, it meant **Mod**ular **Di**gital **Con**troller, and later became the name of a company division devoted to the design, manufacture, and sale of these special-purpose control computers. Other engineering firms developed their own versions of this device, and it eventually came to be known in non-proprietary terms as a *PLC*, or **P**rogrammable **L**ogic **C**ontroller. The purpose of a PLC was to directly replace electromechanical relays as logic elements, substituting instead a solid-state digital computer with a stored program, able to emulate the interconnection of many relays to perform certain logical tasks. [2], [3]. The main difference from other computers is that PLC is armored for severe condition (dust, moisture, heat, cold, etc) and has the facility for extensive input/output (I/O) arrangements. [4] A PLC has many input terminals, through which it interprets high and low logical states from sensors and switches. It also has many output terminals, through which it outputs high and low signals to power lights, solenoids, contactors, small motors, and other devices lending themselves to on/off control. In an effort to make PLC's easy to program, their programming language was designed to resemble ladder logic diagrams. Thus, an industrial electrician or electrical engineer accustomed to reading ladder logic schematics would feel comfortable programming a PLC to perform the same control functions. That why PLC's have been gaining popularity on the factory floor and will probably remain 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.

2.2 History of PLC.

In the 1960s and 1970s, electromechanical relays, timers, counters, and sequencers were the standard. Many control panels contained hundreds of these electromechanical devices and, in some cases, a mile or more of wire. The primary negative aspects of mechanical control were that reliability was low, in contras to the maintenance cost associated with keeping these panels operating, which were extremely high. The auto industry complained that the real cost to purchase and replace a single relay could run as $50. a second major factor was the time, expense, and labor required when a change in control needs a control panel modification. In fact, during the model year changeover, the auto industry discarded entire control panels and replaced them with new ones as the quickest and cheapest solution.

The programmable controller is a solid state electronic device designed in the early 1970s to replace electromechanical relays, mechanical timers, counters, and sequencers. The Hydramatic division of general Motor was the first to see the need for a device that would become what we know as the programmable logic controller (PLC). High-speed manufacturing, as in the auto industry, required reliable control device that were smaller, consumed less power, featured fast switching, and were quickly and easily changeable. These devices must also be able to withstand the harsh industrial environment.

2.3 Operation of PLC

PLC's are industrial computers, and as such their input and output signals are typically 120 volts AC, just like the electromechanical control relays they were designed to replace. Although some PLC's have the ability to input and output low-level DC voltage signals of the magnitude used in logic gate circuits, this is the exception and not the rule.[2]

Signal connection and programming standards vary somewhat between different models of PLC, but they are similar enough to allow a generic introduction to PLC programming here. The following illustration shows a simple PLC, as it might appear from a front view. Two screw terminals provide connection to 120 volts AC for powering the PLC's internal circuitry, labeled L1 and L2. Six screw terminals on the left-hand side provide connection to input devices, each terminal representing a different input channel with its own "X" label. The lower-left screw terminal is a "Common" connection, which is generally connected to L2 (neutral) of the 120 VAC power source.

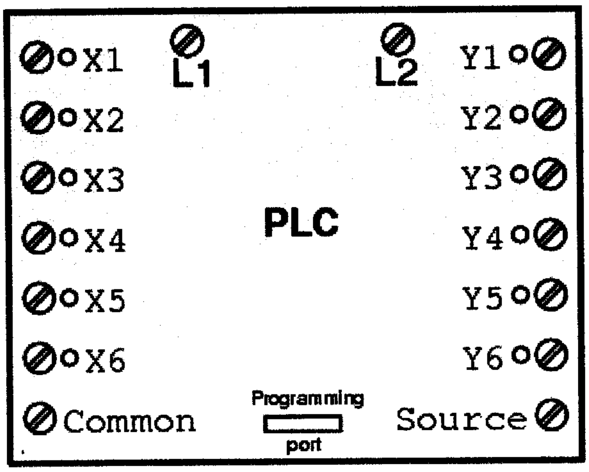

Figure 1: Simple PLC, front view

Inside the PLC housing, connected between each input terminal and the Common terminal, is an opto-isolator device (Light-Emitting Diode) that provides an electrically isolated high logic signal to the computer's circuitry (a photo-transistor interprets the LED's light) when there is 120 VAC power applied between the respective input terminal and the Common terminal. An indicating LED on the front panel of the PLC gives visual indication of an energized input [2]

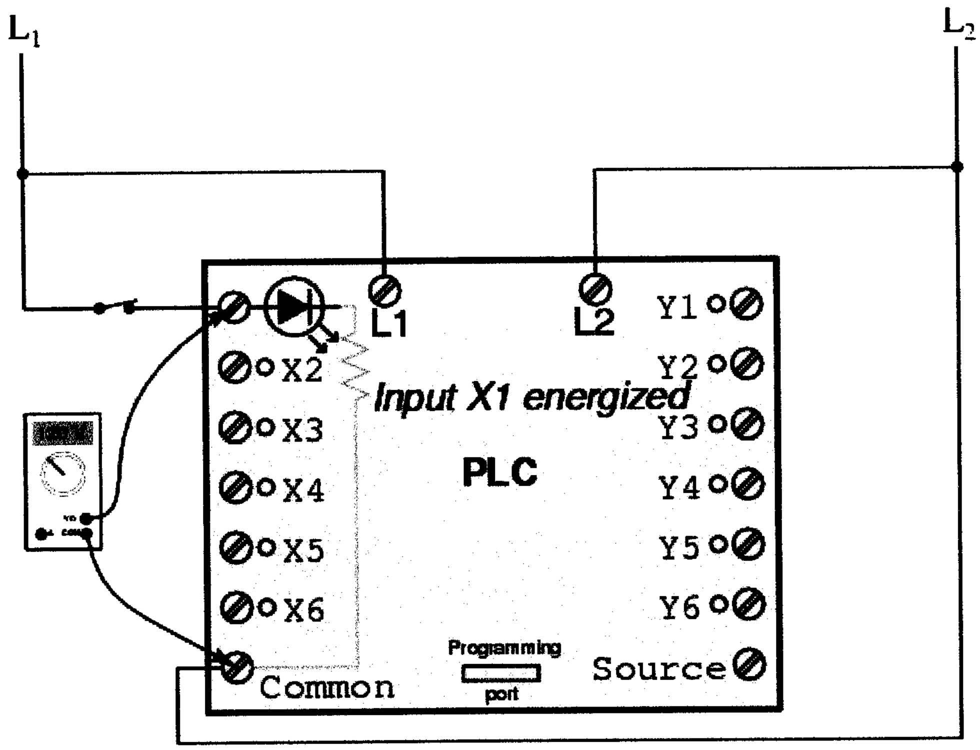

Figure 2: Light-Emitting Diode

Output signals are generated by the PLC's computer circuitry activating a switching device (transistor, TRIAC, or even an electromechanical relay), connecting the "Source" terminal to any of the "Y-" labeled output terminals. The "Source" terminal, correspondingly, is usually connected to the L1 side of the 120 VAC power source. As with each input, an indicating LED on the front panel of the PLC gives visual indication of an "energized" output [2]

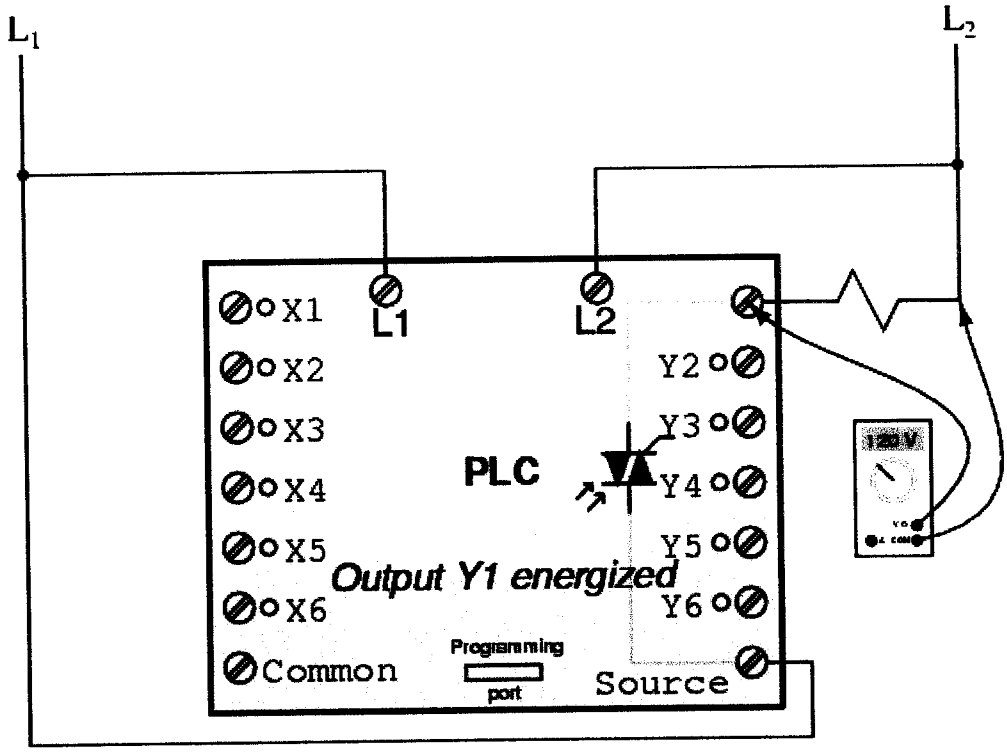

Figure 3: Output signal

In this way, the PLC is able to interface with real-world devices such as switches and solenoids.

The actual *logic* of the control system is established inside the PLC by means of a computer program. This program dictates which output gets energized under which input conditions. Although the program itself appears to be a ladder logic diagram, with switch and relay symbols, there are no actual switch contacts or relay coils operating inside the PLC to create the logical relationships between input and output. These are *imaginary* contacts and coils, if you will. The program is entered and viewed via a personal computer connected to the PLC's programming port.

Consider the following circuit and PLC program below:

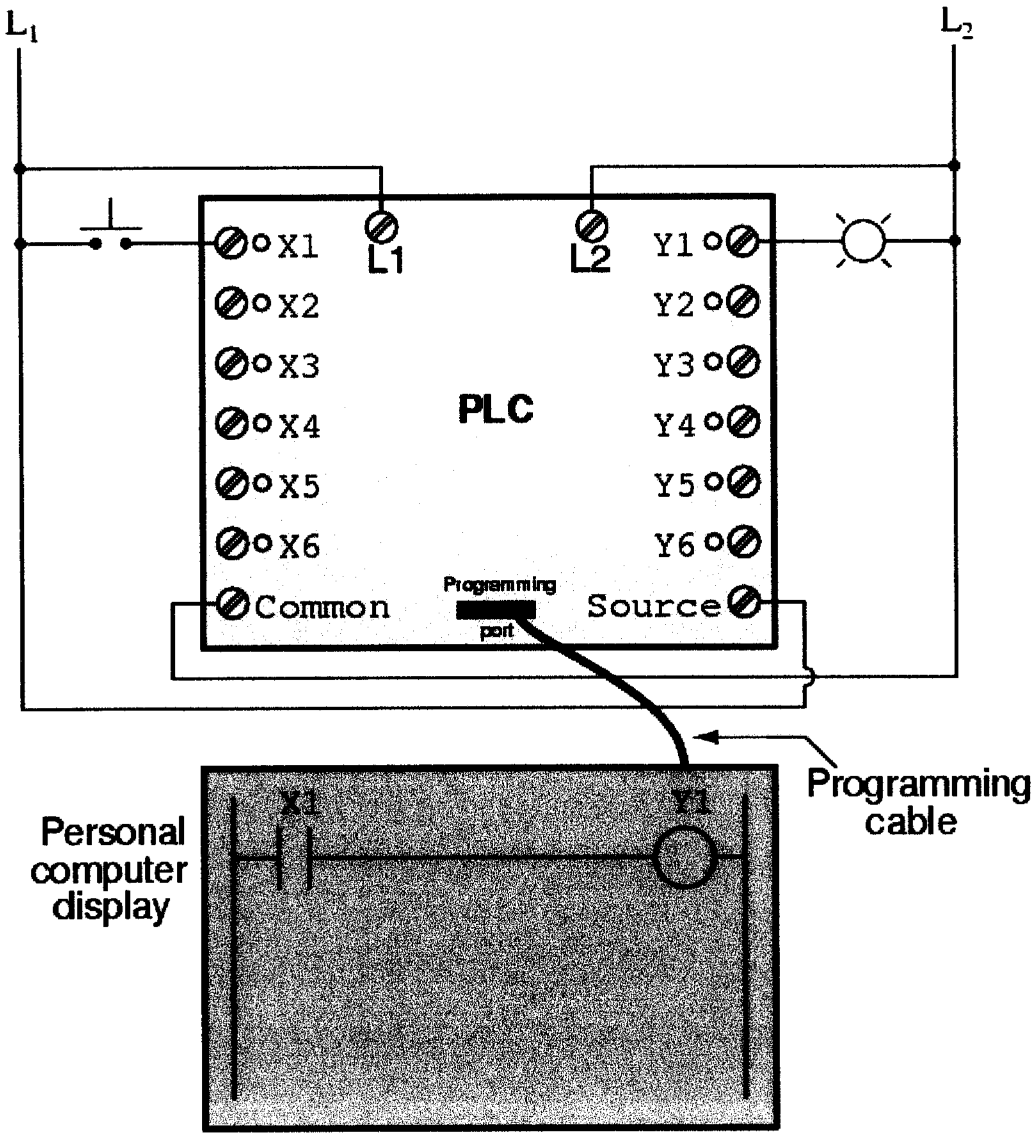

Figure 4: The actual *logic* of the control system

When the pushbutton switch is inactivated (impressed), no power is sent to the X1 input of the PLC. Following the program, which shows a normally-open X1 contact in series with a Y1 coil, no power will be sent to the Y1 coil. Thus, the PLC's Y1 output remains de-energized, and the indicator lamp connected to it remains dark. [2]

If the pushbutton switch is pressed, however, power will be sent to the PLC's X1 input. Any and all X1 contacts appearing in the program will assume the actuated (non-normal) state, as though they were relay contacts actuated by the energizing of a relay coil named "X1". In this case, energizing the X1 input will cause the normally-open X1 contact will close, sending power to the Y1 coil. When the Y1 coil of the program energizes, the real Y1 output will become energized, lighting up the lamp connected to it [2]

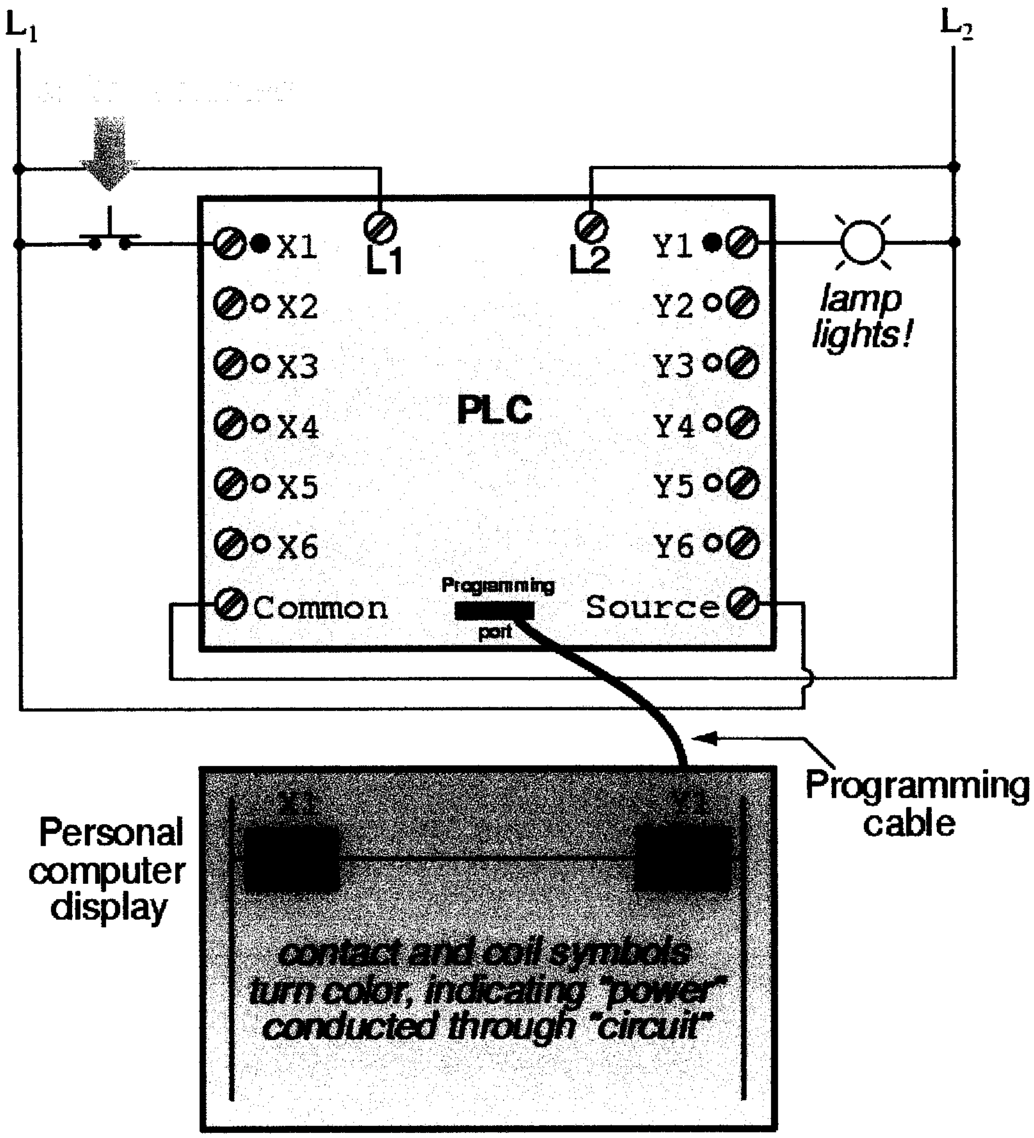

Figure 5: Personal computer's display

It must be understood that the X1 contact, Y1 coil, connecting wires, and power appearing in the personal computer's display are all *virtual*. They do not exist as real electrical components. They exist as commands in a computer program (a piece of software only) that just happens to resemble a real relay schematic diagram.

Equally important to understand is that the personal computer used to display and edit the PLC's program is not necessary for the PLC's continued operation. Once a program has been loaded to the PLC from the personal computer, the personal computer may be unplugged from the PLC, and the PLC will continue to follow the programmed commands. The personal computer display in these illustrations to understand the relationship between real-life conditions (switch closure and lamp status) and the program's status (power through virtual contacts and virtual coils). [2]

The true power and versatility of a PLC is revealed when want to alter the behavior of a control system. Since the PLC is a programmable device, we can alter its behavior by changing the commands we give it, without having to reconfigure the electrical components connected to it.

Furthermore, since each output in the PLC is nothing more than a bit in its memory as well, we can assign contacts in a PLC program actuated by an output (Y) status. Take for instance this system, a motor start-stop control circuit [2]

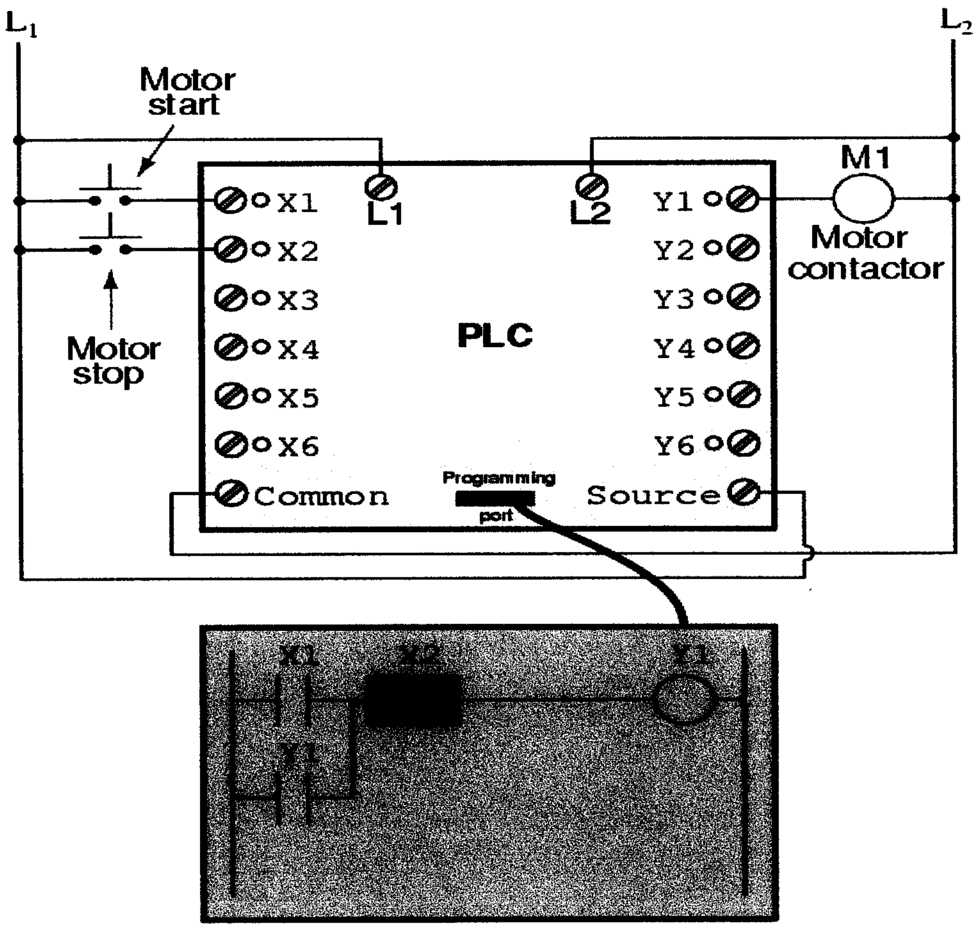

Figure 6: Motor start-stop control circuit

The pushbutton switch connected to input X1 serves as the "Start" switch, while the switch connected to input X2 serves as the "Stop." Another contact in the program, named Y1, uses the output coil status as a seal-in contact, directly, so that the motor contactor will continue to be energized after the "Start" pushbutton switch is released. You can see the normally-closed contact X2 appear in a colored block, showing that it is in a closed (electrically conducting) state. [2]

If we were to press the "Start" button, input X1 would energize, thus closing the X1 contact in the program, sending power to the Y1 coil, energizing the Y1 output and applying 120 volt AC power to the real motor contactor coil. The parallel Y1 contact will also close, thus latching the circuit in an energized state [2]

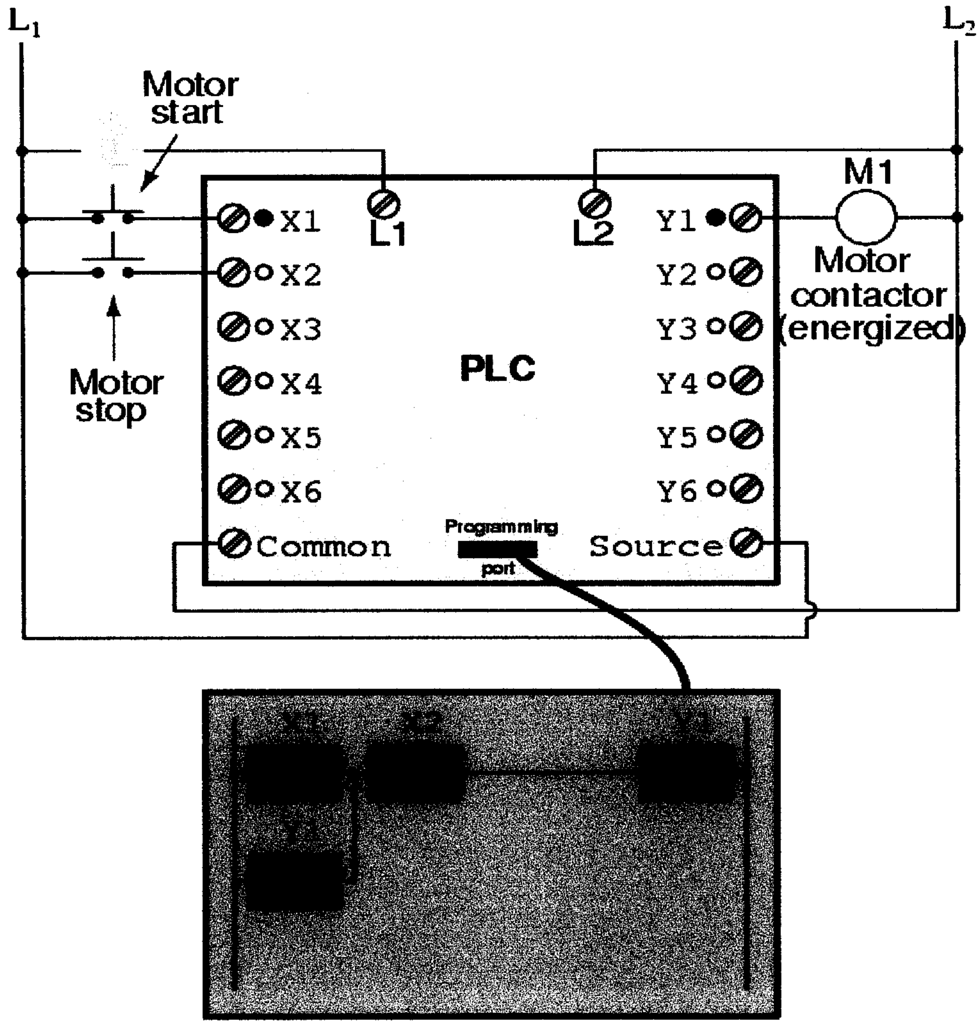

Figure 7: Motor start-stop control circuit (when press start button)

When release the "Start" pushbutton, the normally-open X1 contact will return to its open state, but the motor will continue to run because the Y1 seal-in contact continues to provide continuity to power coil Y1, thus keeping the Y1 output energized [2]

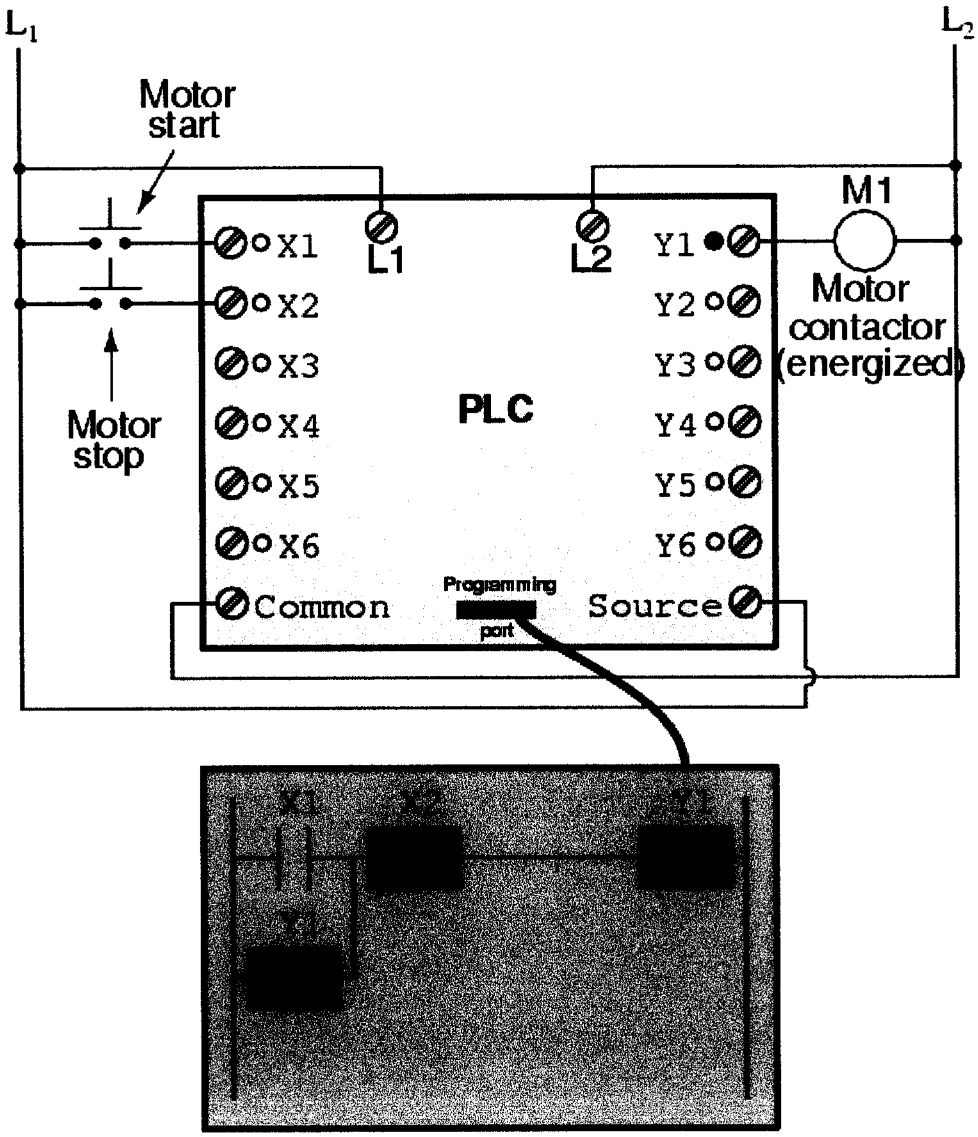

Figure 8: Motor start-stop control circuit (after press start button)

To stop the motor, we must momentarily press the "Stop" pushbutton, which will energize the X2 input and open the normally-closed contact, breaking continuity to the Y1 coil. [2]

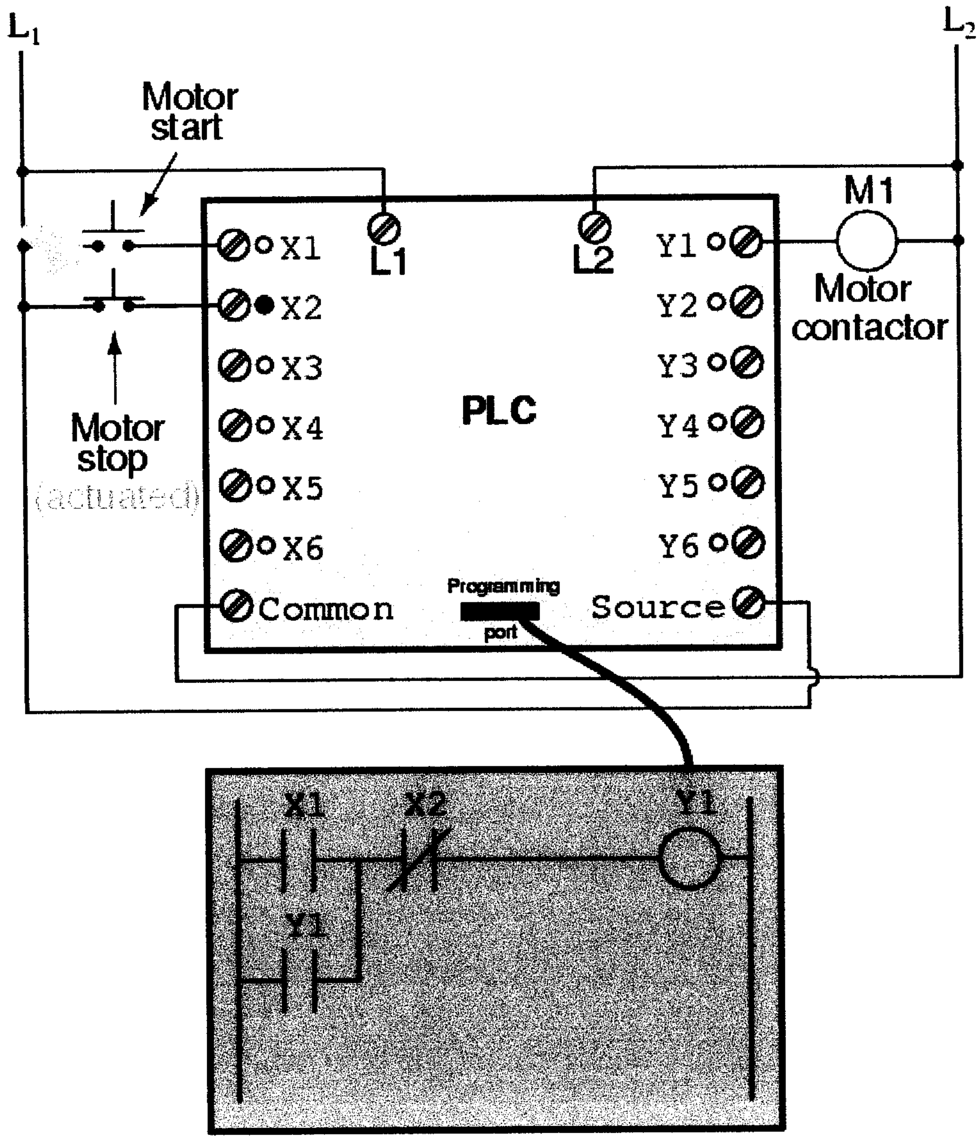

Figure 9: Motor start-stop control circuit (after press stop button)

When the "Stop" pushbutton is released, input X2 will de-energize, returning contact X2 to its normal, closed state. The motor, however, will not start again until the "Start" pushbutton is actuated, because the seal-in of Y1 has been lost. [2]

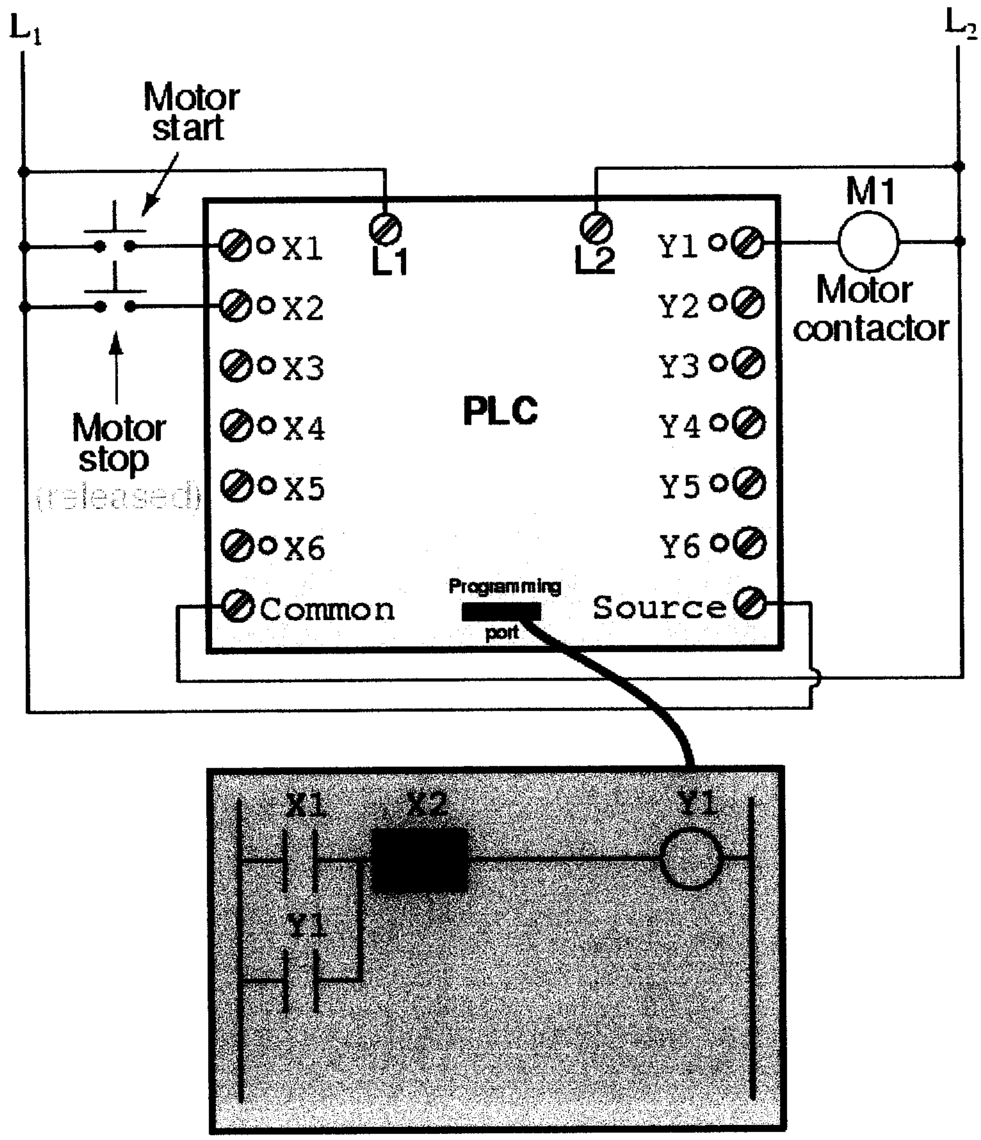

Figure 10: Motor start-stop control circuit (back to normal)

An important point to make here is that *fail-safe* design is just as important in PLC-controlled systems as it is in electromechanical relay-controlled systems. One should always consider the effects of failed (open) wiring on the device or devices being controlled. In this motor control circuit example, we have a problem: if the input wiring for X2 (the "Stop" switch) were to fail open, there would be no way to stop the motor. [2]

The solution to this problem is a reversal of logic between the X2 "contact" inside the PLC program and the actual "Stop" pushbutton switch:

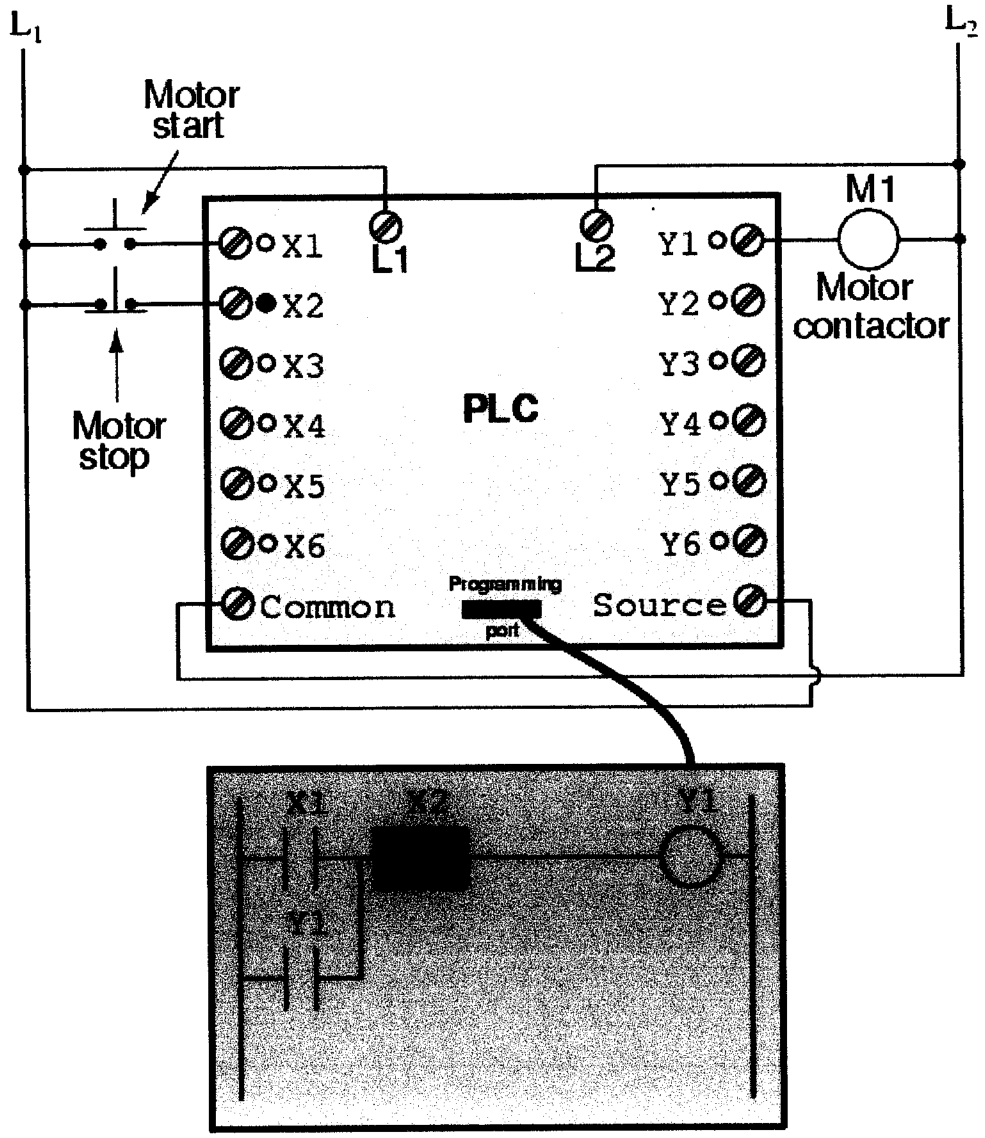

Figure 11: Stop button fail to open

When the normally-closed "Stop" pushbutton switch is inactivated (not pressed), the PLC's X2 input will be energized, thus closing the X2 contact inside the program. This allows the motor to be started when input X1 is energized, and allows it to continue to run when the "Start" pushbutton is no longer pressed. When the "Stop" pushbutton is actuated, input X2 will de-energize, thus opening the X2 contact inside the PLC program and shutting off the motor. So, we see there is no operational difference between this new design and the previous design. [2]

However, if the input wiring on input X2 were to fail open, X2 input would de-energize in the same manner as when the "Stop" pushbutton is pressed. The result, then, for a wiring failure on the X2 input is that the motor will immediately shut off. This is a safer design than the one previously shown, where a "Stop" switch wiring failure would have resulted in an *inability* to turn off the motor.

In addition to input (X) and output (Y) program elements, PLC's provide internal coils and contacts with no intrinsic connection to the outside world. These are used much the same as "control relays" (CR1, CR2, etc.) are used in standard relay circuits: to provide logic signal inversion when necessary. [2]

2.4 Normal star-delta starters

To enable the motor to start, the motor windings are configured in a star formation to the supply voltage. The voltage applied to the individual motor windings therefore reduced by a factor of $1\sqrt{3} = 0.58$. This connection amounts is approximately 30% of the delta values. The starting current is reduced to one third of the direct starting current. Due to the reduced starting torque, the star-delta-connection is suitable for drives with a high inertia mass but a resistance torque which is low or only increases with increased speed. It is preferably used for applications where the drive is only put under a load after run-up. For example is for presses, centrifuges, pumps, ventilators, etc.

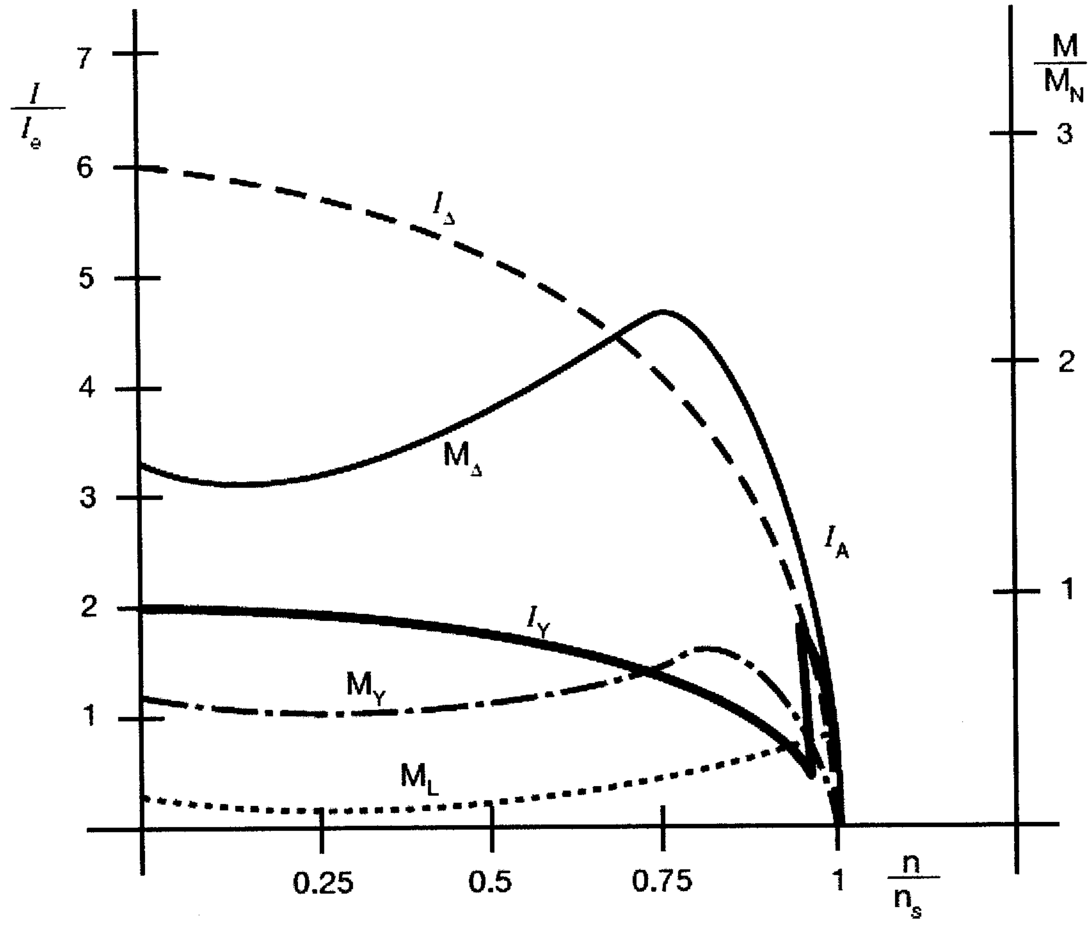

Figure 12: Typical current and torque curve for star-delta starters

I - Motor current

MΔ - Torque for delta connection

ns - Synchronous speed

I_Y - Current in star connection

I_A - Current curve for star-delta start

Ie - Rated operating current of motor

ME - Rated operating torque of motor

M_L - Load torque

IΔ - Current in delta connection

CHAPTER 3

METHODOLOGY

3.1 Research about the project through resources like internet, books and journal.

A study will be conducted to achieve a better understanding in PLC design using books, articles and even websites as a reference. Both programming and design circuit will be researched during this stage

3.2 Design a circuit diagram for the 3-phase induction motor starter.

The design of the circuit diagram includes type of component and ratings.

3.3 Design PLC program using PLC software

The design of PLC programming is base on circuit diagram controller will using the own software of PLC. The software of Mitsubishi FX Series PLC is MELSOFT Application-GX Developer. During this step, the ladder diagram will be construct base on circuit diagram.

3.4 Make simulation for the PLC programming

The design of PLC programming will be simulated to make sure that the program that is constructing can operate

3.5 Develop induction motor starter hardware

The design for the circuit diagram will be developed to hardware same like circuit diagram so the hardware can be implant with PLC. Hardware can be divided by two part, controller part and operation part

3.6 Apply the PLC programming with circuit

By completing the project, the PLC will be interface with hardware part to make sure the operation part of hardware can operate same like the objective of this project.

CHAPTER 4

ANALYSIS

4.1 Advantages and disadvantages of PLC

- Flexibility: One single Programmable Logic Controller can easily run many machines.
- Correcting Errors: In old days, with wired relay-type panels, any program alterations required time for rewiring of panels and devices. With PLC control any change in circuit design or sequence is as simple as retyping the logic. Correcting errors in PLC is extremely short and cost effective.
- Space Efficient: Today's Programmable Logic Control memory is getting bigger and bigger this means that we can generate more and more contacts, coils, timers, sequencers, counters and so on. We can have thousands of contact timers and counters in a single PLC. Imagine what it would be like to have so many things in one panel.
- Low Cost: Prices of Programmable Logic Controllers vary from few hundreds to few thousands. This is nothing compared to the prices of the contact and coils and timers that you would pay to match the same things. Add to that the installation cost, the shipping cost and so on.
- Testing: A Programmable Logic Control program can be tested and evaluated in a lab. The program can be tested, validated and corrected saving very valuable time.
- Visual observation: When running a PLC program a visual operation can be seen on the screen. Hence troubleshooting a circuit is really quick, easy and simple.

Another advantage of the controller is reduced downtime. Consider the time difference between troubleshooting a panel that has a safety PLC installed with minimal wiring and diagnostic capabilities versus a panel that utilizes only traditional relays, numerous metres of wiring and little or no diagnostics. Potential downtime can be prevented as the simplified wiring avoids many of the installation trip-ups that can occur and allows maintenance a fighting chance of trouble-shooting problems. As well, the controller program can be security-protected to restrict access to change the program code. Overall, these can have a direct impact on a plant's overall equipment effectiveness.

4.2 Relay Output - Advantages

Most OEM's use PLCs with relay outputs because they are less expensive and offer greater current capabilities, usually around 2 amps / output. The vast majority of PLCs offering triac outputs offer ratings of only about 500 mA / output. As a result of greater current handling capability, relay outputs offer the advantage of controlling a larger solenoid than do PLCs with triac outputs.

Below is a table of several PLC manufacturers and their triac ratings. The information comes from current data sheets. Due to limited space for this article, we could not mention all manufacturers.

Manufacturer	***Model / Part Number***	***Triac Output Rating***
Allen Bradley	Micrologix: 1761-L32AAA	500 mA / Output
Automation Direct	DL205: D2-12TA	300 mA / Output
Mitsubishi	FX: FX-16EYS-ES-UL	300 mA / Output
Omron	CQM1: CQM-OA-221	400 mA / Output

Table 1: Several PLC manufacturers and their triac ratings

4.3 Relay Output - Disadvantages

Most OEMs rarely consider the ramifications of using relay outputs, mainly due to cost. However, there are several disadvantages. A PLC's relay, just as panel mount relays, have armatures, therefore they will ware out in a shorter period of time than a triac output, which has no moving parts. In order for a relay to operate, a certain amount of current must run through it, for the armature to move from an "off" position to an "on" position, most relays require between 8 - 10 milliseconds to turn on. Should your application require a fast response time, relays may not be adequate. A relay is rated to operate for approximately 1,000,000 mechanical cycles.

Example: To operate an output once a minute, 24 hours a day, 365 days a year would require over 500,000 cycles, giving your PLC a life expectancy of approximately two years of continuous operation.

Keeping in mind that the above example is an extreme situation, however, the life rating of a relay is something that should be taken very seriously. Product reliability is essential to the continued success of an OEM's product. In an operation where the equipment is running continuously, down time is the
greatest cost when a piece of equipment fails, and probably the most dangerous. The potential for a relay failure is far greater than a triac for the reasons mentioned previously. A line down situation could potentially affect the entire company's ability to ship product.

The other cost, is service to a piece of equipment. The greatest costs in servicing a down piece of equipment are the labor and travel involved. In many situations, the OEM, who has supplied the equipment to their customer, is required to maintain the equipment through service contracts. Regardless of who is responsible for the costs, the end result is that someone will have to pay. By paying more up front, in the form of a more reliable output configuration, will constitute less expense down the road. When equipment goes down, it doesn't always happen when you want a long weekend or holiday. The costs, then to service the equipment have no boundaries.

4.4 PLC in industrial

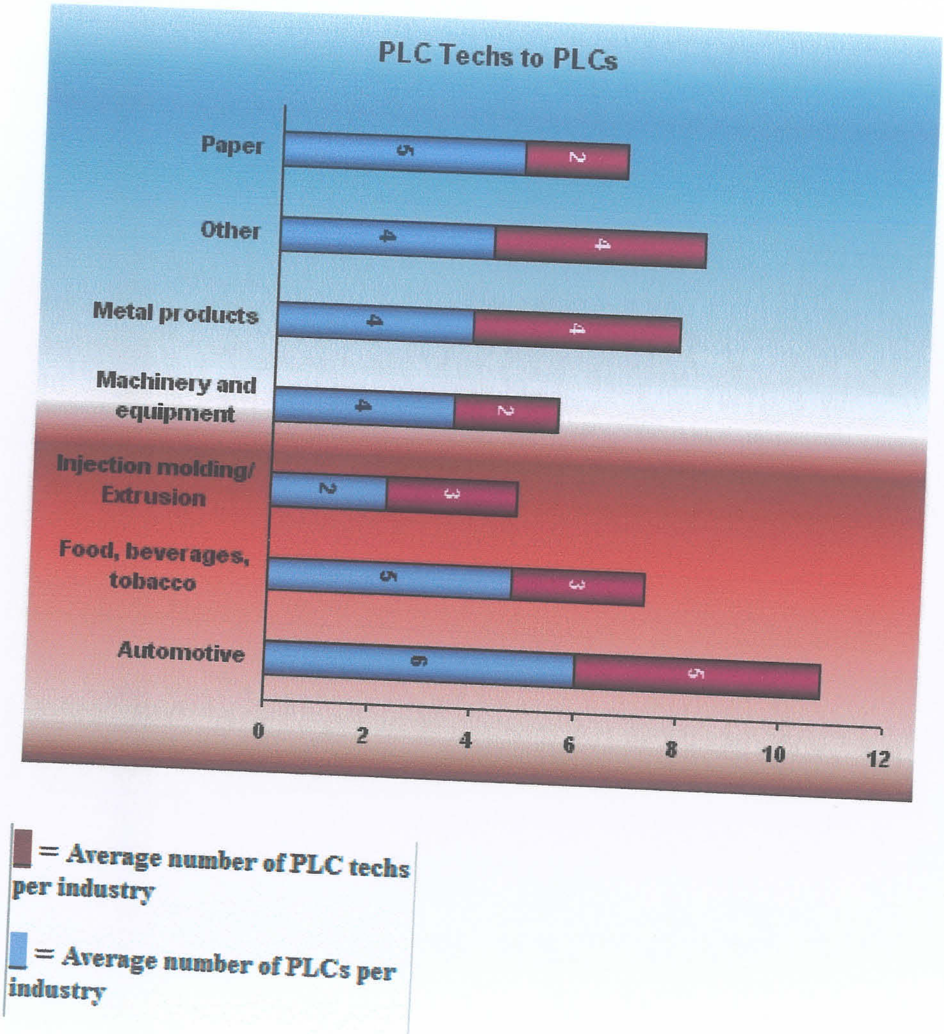

Figure 13: PLC in industrial

In figure above shows that in automotive industrial have use more PLC than another sector. It is because in automotive sector use more robotic machine than using manual because to improve the quality and quantity. Using PLC as a controller of the