NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Electrical and Electronic Engineering

LAMP CONTROL USING PLC

Graduation Project
EE- 400

Student: Saadet Burcu Baturalp (980278)

Supervisor: Mr. Özgür Cemal Özerdem

Nicosia-2002

ACKNOWLEDGEMENTS

First of all, I would like to say how grateful I am to my supervisor, Mr. Özgür Cemal Özerdem, friends, parents and elder brother. I could not have prepared this Graduation Project without the generous help of Mr. Cemal Kavalcıoğlu.

I would like to thank my supervisor Mr. Özgür Cemal Özerdem. Under his guidance, I successfully overcome many difficulties and learn a lot about Programmable Logic Controller(PLC).I asked him many questions in PLC, he explained my questions patiently.

I would like to express my gratitude to Vice-President Prof. Dr. Şenol Bektaş for him because he helped to me at each stage of my Undergraduate Education in Near East University.

I also wish to thank my advisor and supervisor Mr. Özgür Cemal Özerdem at my Undergraduate Education for his invaluable advices, for his help and for his patience also for his support.

Finally, I want to thank my family, especially my parents without their endless support, I could never have prepared this thesis without the encouragement and support of my parents and Cemal Şirin.

ABSTRACT

As PLC technology has advanced, so have programming languages and communications capabilities, along with many other important features. Today's PLCs offer faster scan times, space efficient high-density input/output systems, and special interfaces to allow non-traditional devices to be attached directly to the PLC. Not only can they communicate with other control systems, they can also perform reporting functions and diagnose their own failures, as well as the failure of a machine or process. Size is typically used to categorize today's PLC, and is often an indication of the features and types of applications it will accommodate.

In this project CPU 212 8 inputs, 6 outputs, 24 volt DC input 220 Volt AC output PLC is used, and an automation of a Lamp and an industrial motor is realized by PLC programs.

Real life experimental operation is performed and the Lamp and AC motor is physically operated.

TABLE OF CONTENTS

INTRODUCTION

A Programmable Logic Controller was defined by Capiel (1982) as:

" A digitally operating electronic system designed for use in an industrial environment, which uses a programmable memory for the internal storage of instructions for implementing specific functions such as logic, sequencing, timing, counting and arithmetic to control through analog or digital input/output modules, various types of machines or processes." Which explains the device perfectly.

In the late 1960's PLC's were first introduced. The first PLC can be traced back to 1968 when Bedford Associates, a company in Bedford, MA, developed a device called a Modular Digital Controller for General Motors (GM). The MODICON, as it was known, was developed to help GM eliminate traditional relay-based machine control systems.

The aim of this thesis is the control of the Lamp and AC Motor with SIMATIC S7-200 MICRO/WIN PLC.

The Thesis consists of the introduction, three chapters, and conclusion.

The Chapter-1 presents History of PLC, What is PLC?, Types of PLC's Hardware of PLC's, and chapter concluded with a brief information about Today's PLC's.

Chapter-2 presents the informations about Siemens SIMATIC S7-200 PLCs and application areas of this type of PLCs.

Chapter-3 presents a practical implementation about Lamp and AC motor Control with PLC program.

CHAPTER 1. PROGRAMMABLE LOGIC CONTROLLERS

1.1 HISTORY OF PLC

The first PLC can be traced back to 1968 when Bedford Associates, a company in Bedford, MA, developed a device called a Modular Digital Controller for General Motors (GM). The MODICON, as it was known, was developed to help GM eliminate traditional relay-based machine control systems. Because relays are mechanical devices, they have limited lifetimes. They are also cumbersome, especially in large applications where thousands of them may exist. With so many relays to work with, wiring and troubleshooting could be quite complicated. Since the MODICON was an electronic device, not a mechanical one, it was perfect for GM's requirements, as well as for many other manufacturers and users of control equipment. With less wiring, simpler troubleshooting, and easy programming, PLC technology caught on quickly. In the late 1960's PLC's were first introduced. The primary reason for designing such a device was eliminating the large cost involved in replacing the complicated relay based machine control systems. Bedford Associates (Bedford, MA) proposed something called a modular digital controller (MODICON) to a major US car manufacturer. Other companies at the time proposed computer based upon the PDP – 8. The MODICON 084 brought the world's first PLC into commercial production. When production requirements changed so did the control system. This becomes very expensive when the change is frequent. Since relays are mechanical devices they also have a limited lifetime that required strict adhesion to maintenance schedules. Troubleshooting was also quite tedious when so many relays are involved. Now picture a machine control panel that included many, possible hundreds or thousands, of individual relays. The size could be mind-boggling. How about the complicated initial wiring of so many individual devices! These relays would be individually wired together in a manner that would yield the desired outcome. These new controllers also had to be easily programmed by maintenance and plant engineers. The lifetime had to be long and programming changes easily performed. The also had to survive the harsh industrial environment. That's a lot to ask! The answers were to use a programming technique most people were already familiar with and replace mechanical parts with solid-state ones.

In the mid70's the dominant PLC techniques were sequencer state machines and the bit-slice based CPU. The AMD 2901 and 2903 were quite popular in MODICON and A-B PLC's. Conventional microprocessors lacked the power to quickly solve PLC logic in all but the smallest PLC's. As conventional microprocessors evolved, larger and larger PLC's were being based upon them. However, even today some are still based upon the 2903. MODICON has yet the build a faster PLC then their 984A/B/X, which was based upon the 2901.

Communications abilities began to appear in approximately 1973. The first such system was MODICON's MODBUS. The PLC could now talk to other PLC's and they could be far away from the actual machine they were controlling. They could also now be used to send and receive varying voltages to allow them to enter the analogue world. Unfortunately, the lack of standardization coupled with continually changing technology has made PLC communications a nightmare of incompatible protocols and physical networks.

The 80's saw an attempt to standardize communications with General Motor's manufacturing automation protocol. It was also a time for reducing the size of the PLC and making them software programmable through symbolic programming on personnel computers instead of dedicated programming terminals or handheld programmers.

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

1.2 WHAT IS A PLC ?

PLCs are often defined as miniature industrial computers that contain hardware and software that is used to perform control functions. A PLC consists of two basic sections: the central processing unit (CPU) and the input/output interface system. The CPU, which controls all PLC activity, can further be broken down into the processor and memory system. The input/output system is physically connected to field devices (e.g., switches, sensors, etc.) and provides the interface between the CPU and the information providers (inputs) and controllable devices (outputs). To operate, the CPU "reads" input data from connected field devices through the use of its input interfaces, and then "executes", or performs the control program that has been stored in its memory system. Programs are typically created in ladder logic, a language that closely resembles a relay-based wiring schematic, and are entered into the CPU's memory prior to operation. Finally, based on the program, the PLC "writes", or updates output devices via the output interfaces. This process, also known as scanning, continues in the same sequence without interruption, and changes only when a change is made to the control program

A programmable logic controller (PLC) is a device that was invented to replace the necessary sequential relay circuits for machine control. The PLC works by looking at its inputs and depending upon their state, turning on / off its outputs. The user enters a program, usually via software, that gives the desired results.

PLC's are used in many real world applications. If there is industry present, changes are good that there is a PLC present. If you are involved in machining, packaging material handling, and automated assembly or countless other industries you are probably already using them. If you are not, you are wasting money and time. Almost any application that needs some type of electrical control has a need for PLC.

For example, let's assume that when a switch turned on, we want turn on a solenoid for 5 seconds and then turn it off regularly of how long the switch is on. We can do this with a simple external timer. But what will happen if the process included 10 switches and solenoids? We would need 10 external timers. What will happen if the process also needed to count how many times the switches individually turned on? We would need a lot of external counters.

As you see, if the process becomes more complicated, then we have to use a device the simplify that. We use PLC for this process. We can program the PLC to count its inputs and turn the solenoids for the specified time.

This site gives you enough information to be able to write programs for more complicated then the simple than above. We will take a look at what is considered to be the 'top 20' PLC instructions. It can be safely estimated that with affirm understanding of these instructions, that just one of them can solve more than 80 % of the applications in existence.

1.3.GENERAL PHYSICAL BUILD MECHANISM

PLC's are separated into two according to their building mechanisms.

1.3.1 Compact PLC'S

Compact PLC's are manufactured such that all units forming the PLC are placed in a case. They are low price PLC with lower capacity. Small or medium size machine manufacturers usually prefer them. In some types compact enlargement module is present.

1.3.2 Modular PLC's

Combining separate modules together in a board forms them. They can have different memory capacity, I / O numbers, power supply up to the necessary limits.

Some examples: SIEMENS S5-115U, SIEMENS S7-200, MITSUBISHI PC40, TEXAS INSTRUMENT PLC'S, KLOCKNER-MOELLER PS316, OMRON C200H.

1.4 PROGRAMMABLE CONTROLLER

1.4.1 Overview

The need for low cost, versatile and easily commissioned controllers has resulted in the development of programmable-control systems standard units based on a hardware CPU and memory for the control of machines or processes. Originally designed as a replacement for the hard-wired relay and timer logic to be found in traditional control panels, PLC's provides ease and flexibility of control based on programming and executing simple logic instructions. PLC's have internal functions such as timers, counters and shift registers, making sophisticated control possible using even the smallest PLC.

A programmable control operates by examining the input signals from a process and carrying out logic instructions on these input signals, producing output signal to drive process equipment or machinery. Standard interfaces build into PLC's allow them to be directly connected to process actuators and transducers (pumps and valves) without the need for intermediate circuitry or relays.

Through using PLC's it became possible to modify a control system without having the disconnect or re-route a signal wire. It was necessary to change only the control program using a keypad or VDU terminal. Programmable controllers also require shorter installation and commissioning times than do hardwired systems. Although PLC's are similar to conventional computers in terms of hardware technology, they have specific features suited to industrial control:

- Rugged, noise immune equipment;
- Modular plug-in construction, allowing easy replacement\addition of units (input\output);
- Standard input\output connections and signal levels;
- Easily understood programming language (ladder diagram and function chart),
- Ease of programming and reprogramming in-plant.

These features make programmable controllers highly desirable in a wide variety of industrial-plant and process-control situations.

1.4.2 Background

The programmable controller was initially conceived by a group of engineers from General Motors in 1968, where an initial specification was provided: the controller must be:

1 Easily programmed and reprogrammed, preferably in-plant to alter its sequence of operations.

2 Easily maintained and repaired- preferably using plug-in modules.

a) More reliable in plant environment.

b) Smaller than it is relay equivalent.

3 Cost competitive, with solid-state and relay panels than in use.

This provoked a keen interest from engineers of all disciplines in how to PLC could be used for industrial control. With this came demands for additional PLC capabilities and facilities, which were rapidly implemented as the technology became available. The instruction sets quickly moved from simple logic instructions to include counters, timers and shift registers, than onto more advanced mathematical functions on the machines. Developments hardware were also occurring, with larger memory and greater numbers of input / output points featuring on new models. In 1976 became possible to control remote I / O racks, where large numbers of distant I / O points were monitored updated via a communications link, often several hundred meters from the main PLC. The Allan-Bradley Corporation in America introduced a microprocessor-based PLC in 1977. It was based on an 8080 microprocessor but used an extra processor to handle bit logic instruction at high speed.

The increased rate of application of programmable controllers within industry has encouraged manufacturers to develop whole families of microprocessor-based systems having various levels of performance. The range of available PLC's now extends from small self-contained units with 20 digital I / O points and 500 program steps, up to modular systems with add-on function modules:

- Analogue I/O;
- PID control (proportional, integral and derivative terms);
- Communications;
- Graphics display;
- Additional I/O;
- Additional memory.

This modular approach allows the expansion or upgrading of a control system with minimum cost and disturbance.

Programmable controllers are developing at a virtually the same pace as microcomputers, with particular emphasis on small controllers, positioning\numeric control and communication networks. The market for small controllers has grown rapidly since the early 1980's when a number of Japanese companies introduced very small, low cost units that were much cheaper than others available at that time. This brought programmable controllers within the budget of many potential users in the manufacturing and process industries, and this trend continues with PLC's offering ever-increasing performance at ever-decreasing cost.

The Mitsubishi F40 PLC is a typical example of a modern small PLC, providing 40 I / O points, 16 timers and counters, plus other functions. The controller uses a microprocessor and has 890 RAM locations fur user programs. The 24-input channels of the F40 operate at 24 V d.c. Whilst 16 outputs may be 24 V d.c. or 240 V a.c. to provide easy interfacing to industrial equipment.

1.4.3. Terminology-PC or PLC

There are several different terms used to describe programmable controllers, most referring to the functional operation of the machine in question:

- PC programmable controller
- PLC programmable logic controller
- PBS programmable binary system

By their nature these terms tend to describe controllers that normally work in a binary environment. Since all but the smallest programmable controllers can now be equipped to process analogue inputs and outputs these labels are not representative of their capabilities. For these reason the overall term programmable controller has been widely adopted to describe the family of freely programmable controllers. However, to avoid confusion with the personal computer PC, this text uses the abbreviation PLC for programmable (logic) controller.

1.4.4. PLC Hardware Design

Programmable controllers are purpose-built computers consisting of three functional areas:

- Processing:
- Memory:
- Input / output:

Input conditions to the PLC are sensed and than stored in the memory, where the PLC performs the programmed logic instructions on these input states. Output conditions are then generated to drive associated equipment. The action taken depends totally on the control program held in memory.

In smaller PLC these functions are performed by individual printed circuit cards within a single compact unit, whilst larger PLC's are constructed on a modular basis with function modules slotted in to the back plane connectors of the mounting rack.

This allows simple expansion of the system when necessary. In both these cases the individual circuit board are easily removed and replaced, facilitating rapid repair of the system should faults develop.

In addition a programming unit is necessary to download control programs to the PLC memory.

a) Input output / units

Most PLC'S operate internally at between 5 and 15 V d.c. (Common TTL and CMOS voltages), whilst process signals much greater, typically 24 V d.c. to 240 V a.c. at several amperes.

The I / O units form the interface between the microelectronics of the programmable controller and real world outside, and must therefore provide all, necessary signal conditioning and isolation functions. This often allows a PLC to be directly connected to process actuators and transducers (pumps and valves) without the need for intermediate circuitry and relays.

To provide this signal conversion programmable controllers are available with a choice of input / output units to suit different requirements.

For example:

Inputs:	5 V (TTL level) switched I/ P
	24 V switched I/ P
	110 V switched I/ P
	240 v switched I/ P
Outputs:	24 V 100 mA switched O/ P
	110 V 1mA
	240 V 1 A a.c. (triac)
	240 V 2 A a.c. (relay)

It is standard practice for all I/O channels to electrically isolated from the controlled process, using opto-isolator circuits on the I/O modules. An opto-isolator circuit consists of a light emitting diode and a phototransistor, forming an opto coupled pair that allows small signals to pass through, but will clamp any high voltage spikes or surges down to the same small level. This provides protection against switching transients and power-supply surges, normally up to 1500V.

In small self contained PLC's in which all I/ O points are physically located on the one casing, all inputs will be of one type (e.g. 24 V) and the same for outputs (e.g. all 240 V triac). This is because manufacturers supply on the standard function boards for economic reasons. Modular PLC's have greater flexibility of I/ O, however, since the user can select from several different types and combinations of input and output modules.

In all cases the input/output units are designed with the aim of simplifying the connections of process transducers and actuators to the programmable controller. For these purpose all PLC'S are equipped with standard screw terminals or plugs on every I\O point, allowing the rapid and simple removal and replacement of a faulty I/ O card. Every input\output point has a unique address or channel number, which is using during program development to specify to monitoring of an input or the activating of a particular output within the program. Indication of the status of input\output channels is provided by light-emitting diode (LED's) on the PLC or I/ O unit, making it simple to check the operation of process inputs and outputs from the PLC itself.

b) Central Processing Unit (CPU)

The CPU controls and supervises all operations within the PLC, carrying out programmed instructions stored in memory. An internal communications highway or bus system carries information to and from the CP, memory and I/ O units, under control of the CPU. The CPU is supplied with a clock frequency by an external quartz crystal or RC oscillator, typically between 1 and 8 megahertz depending on the microprocessor used and the area of application.

The clock determines the operating speed of the PLC and provides timing\synchronization for all elements in the system. Virtually all modern programmable controllers are microprocessor based using a micro as a system CPU. Some larger PLC's also employ additional microprocessor to control complex, time-consuming functions such as mathematical processing, three terms PID control.

c) Memory

- For program storage all modern programmable controllers use semiconductor memory devices such as RAM read\write memory, or a programmable read-only memory of the EPROM or EEPROM families.

In the virtually all cases RAM is used for initial program development and testing, as it follows changes to be easily made in program. The current trend is to be providing CMOS RAM because of it is very low power consumption, to provide battery back up to this RAM in order to maintain the contents when the power is removed from the PLC system. This battery has a lifespan of at least one year before replacement is necessary, or alternatively a rechargeable type may be supplied with the system being recharge whenever the main PLC power supply is on.

This feature makes programs stored in RAM virtually permanent. Many users operate their PLC systems on this basis alone, since it permits future program alterations if and when necessary.

After a program is fully developed and tested it may be loaded (blown) into a PROM or EPROM memory chip, which are normally cheaper than RAM devices. PROM programming is usually carried out with a special purpose programming unit, although many programmable controllers now have this facility built-in, allowing programs in the PLC RAM to be down loaded into a PROM IC placed in a socket provided on the PLC itself.

(a) In addition to program storage, a programmable controller may require memory for other functions:

1- Temporary buffer store for input\output channels status- I/ O RAM

2- Temporary storage for status of internal function (timers, counters, marker relays)

Since these consist of changing data they require RAM read\write memory, which may be battery-backed in sections.

d) Memory size

Smaller programmable controllers normally have a fixed memory size, due in part to the physical dimensions of the unit. This varies in capacity between 300 and 1000 instructions depending on the manufacturer. This capacity may not appear large enough to be very useful, but it has been estimated that 90 % of all binary control tasks can be solved using less than 1000 instructions, so there is sufficient space to meet most users needs.

Larger PLC's utilize memory modules of between 1K and 64K in size, allowing the system to be expanded by fitting addition RAM or PROM memory cards to the PLC rack.

As integrated circuit memory costs continue to fall, the PLC manufacturers are providing larger program memories on all products.

e) Logic instruction set

The most common technique for programming small PLC's is to draw s ladder diagram of the logic to be used, and then convert this in to mnemonic instructions, which will be keyed in to programming panel attached to the programmable controller. These instructions are similar in appearance to assembly-type codes, but refer to physical inputs, outputs and functions within the PLC itself.

The instruction set consists of logic instructions (mnemonics) that represent the actions that may be performed within a given programmable controller. Instructions sets vary between PLC's from different manufacturers, but are similar in terms of the control actions performed.

Because the PLC logic instruction set tends to be small, it can be quickly mastered and used by control technicians and engineers.

Each program instructions are made up of two parts: a mnemonic operation component or opcode, and an address or operand component that identifies particular elements

(E.g. outputs) within the PLC.

For example;

Opcode	Operand
OUT	Y430
Device symbol	Identifier

Here the instruction refers to output (Y) number 430

f) Input\output numbering

These instructions are used the program logic control circuits that have been designed in ladder diagram form, by assigning all physical inputs and outputs with an operand suitable to the PLC being used. The numbering system used differs between manufacturers, but certain common terms exist. For example, Texas instrument and Mitsubishi use the symbol X to represent inputs, and Y to label outputs.

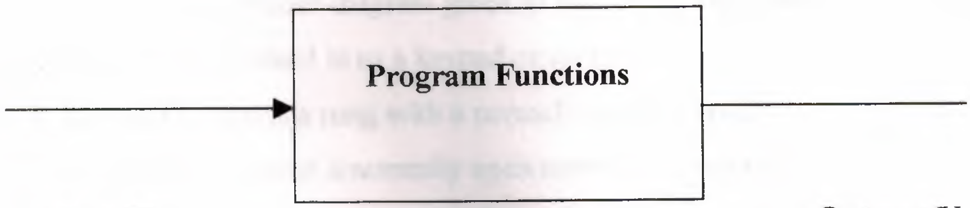


Figure 1.1 Programmable Controller

A range of addresses will be allocated to particular elements: for example

Mitsubishi F40 PLC: 24 inputs: X400 – 407, 410 – 413

X500 – 507, 510 – 513

16 Outputs: Y430 – 437

Y530 – 537

Inspections of these numbers ranges will reveal that there is no overlap of addresses between functions; that is, 400 must be an input, 533 must be an output. Thus for these programmable controllers the symbol X or Y is redundant, being used purely for the benefit of the user, who is unlikely to remember what element 533 represents. However, for many PLC's both parts of the address are essential, since the I\O number ranges are identical. For example the Klockner-Moeller range of controllers:

Sucos PS 21 PLC: 8 inputs I0 to 7, etc. 8 Outputs Q0 to 7, etc

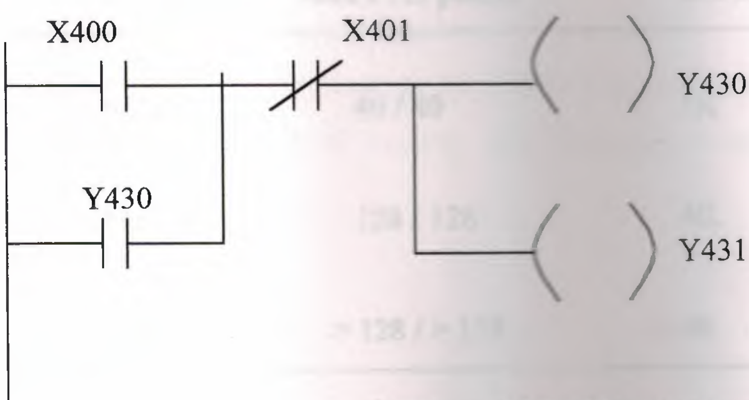


Figure 1.2 Ladder Diagram

To program the ladder diagram given in figure 1.2, the following code would be written, and then programmed in to a keypad or terminal.

1. LD X400 starts a rung with a normally open contact
2. OR Y430 connect a normally open contact in parallel
3. ANI X401 connect a normally closed contact in series
4. OUT Y430 drive an output channel
5. OUT Y431 drive another channel
6. END end of program-return to start

Notice the contact Y430 that forms a latch across X400. The Y contact is not a physical element, but is simulated within the programmable controller and will operate in unison with the output point Y430. The programmer may create as many contacts associated with an output as necessary.

1.5. Types Of PLC

The increasing demand from industry for programmable controllers that can be applied to different forms and sizes of control tasks has resulted in most manufacturers producing a range of PLC's with various levels of performance and facilities.

Typical rough definitions of PLC size are given in terms of program memory size and the maximum number of input\output points the system can support. Table 7.1 gives an example of these categories.

Table1.1 Categories of PLC

PC size	Max I \ O points	Use memory size
Small	40 / 40	1K
Medium	128 / 128	4K
Large	> 128 / > 128	> 4K

However, to evaluate properly any programmable controller we must consider many additional features such as its processor, cycle time language facilities, functions, and expansion capabilities.

A brief outline of the characteristics of small, medium of large programmable controller is given below, together with typical applications.

1.5.1. Small PLC's

In general, small and 'mini' PLC's are designed as robust, compact units, which can be mounted on or beside the equipment to be controlled. They are mainly used the replaced hard-wired logic relays, timers, counters. That control individual items of plant or machinery, but can also be used to coordinate several machines working in conjunction with each other.

Small programmable controllers can normally have their total I/ O expanded by adding one or two I/ O modules, but if any further developments are required this will often mean replacement of the complete unit. This end of the market is very much concerned with non-specialist and users, therefore ease of programming and a 'familiar' circuit format are desirable. Competition between manufacturers is extremely fierce in this field, as they vie to obtain a maximum share in this partially developed sector of the market.

A single processor is normally used, and programming facilities are kept a fairly basic level, including conventional sequencing controls and simple standard functions: e.g. timers and counters. Programming of small PLC's is by way of logic instruction list (mnemonics) or relay ladder diagrams.

Program storage is given by EPROM or battery-backed RAM. There is now a trend towards EEPROM memory with on-board programming facilities on several controllers.

Table 1.2 Features of a typical small PLC – Mitsubishi F20

Electrical: 240 V a.c. supply;

24 V d.c. On-board for input requirements;

12 inputs, 8 output points;

LED indicators on all I/ O points;

All I\O Opto-isolated

Choice of output: Relay (240 V 2 a rated)

Triac (240 V 1 A rated)

Transistor (24 V d.c. 1 A)

320 – step memory (CMOS battery-backed RAM)

Programming: Ladder logic or instruction set using hand-held or graphic LCD programmer, with editor, test and monitor facilities;

Facilities:

- 8 counters, range 1-99 (can be cascaded)
- 8 timers, range 0.1-99s (can be cascaded)
- 64 markers\auxiliary relays; can be used individually or in blocks
- Forming shift registers;
- Special function relays;
- Jump capability.

1.5.2 Medium-sized PLC's

In this range modular construction predominates with plug-in modules based around the Euro card 19-inch rack format or another rack mounting system. This construction allows the simple upgrading or expansion of the system. This construction allows the simple upgrading or expansion of the system by fitting additional I/ O cards in to the cards into the rack, since most rack, systems have space for several extra function cards. Boards are usually 'rugged zed' to allow reliable operation over a range of environments.

In general this type of PLC is applied to logic control tasks that can not be met by small controllers due to insufficient I\O provision, or because the control task is likely to be extended in the future. This might require the replacement of a small PLC, where as a modular system can be expanded to a much greater extent, allowing for growth. A medium-sized PLC may therefore be financially more attractive in the long term.

Communications of a single and multi-bit processor are likely within the CPU. For programming, standard instructions or ladder and logic diagrams are available. Programming is normally carried out via a small keypad or a VDU terminal. If different sizes of PLC are purchased from a single manufacturer, it is likely that programs and programming panels will be compatible between the machines.

1.5.3 Large PLC

Where control of very large numbers of input and output points is necessary and complex control functions are required, a large programmable controller is the obvious choice. Large PLC's are designed for use in large plants or on large machines requiring continuous control. They are also employed as supervisory controllers to monitor and control several other PLC's or intelligent machines. e.g. CNC tools.

Modular construction in Euro card format is standard, with a wide range of function cards available including analogue input output modules. There is a move towards 16-bit processor, and also multi-processor usage in order to efficiently handle a large range of differing control tasks.

For example;

- 16-bit processor as main processor for digital arithmetic and text handling.
- Single-bit processor as co-or parallel processor for fast counting, storage etc.
- Peripheral processor for handling additional tasks which are time-dependent or time-critical, such as:

- Closed-loop (PID) control
- Position controls
- Floating-point numerical calculations
- Diagnostic and monitoring
- Communications for decentralized
- Remote input\output racks.

This multi-processor solution optimizes the performance of the overall system as regards versatility and processing speed, allowing to PLC to handle very large programs of 100 K instructions or more. Memory cards can now provide several megabytes of CMOS RAM or EPROM storage.

1.5.4. Remote input\output

When large numbers of input / output points are located a considerable distance away from the programmable controller, it is uneconomic to run connecting cables to every point. A solution to this problem is to site a remote I/ O unit near to the desired I/ O points. This acts as a concentrator to monitor all inputs and transmit their status over a single serial communications link to the programmable controller. Once output signals have been produced by the PLC they are feedback along the communications cable to the remote I/ O unit, which converts the serial data into the individual output signals to drive the process.

1.5.5 Programming large PLC's

Virtually any function can be programmed, using the familiar ladder symbols via a graphics terminal or personal computer. Parameters are passed to relevant modules either by incorporating constants in to the ladder, or via on screen menus for that module.

There may in addition be computer-oriented languages, which allow programming of function modules and subroutines.

There is progress towards standardization of programming languages; with programs becoming easier to over-view through improvement of text handling, hand improved documentation facilities. This is assisted by the application of personal computers as workstations.

1.5.6. Developments

Present trends include the integration of process data from a PLC into management databases, etc. This allows immediate presentation of information to those involved in scheduling, production and planning.

The need to pass process information between PC's and PLC sand other devices within a automated plants has resulted in the provision of a communications capability on all but the smallest controller. The development of local area networks (LAN) and in particular the recent MAP specification by General Motors (manufacturing automation protocol) provides the communication link to integrate all levels of control systems.

1.6.Today's PLC

As PLC technology has advanced, so have programming languages and communications capabilities, along with many other important features. Today's PLCs offer faster scan times, space efficient high-density input/output systems, and special interfaces to allow non-traditional devices to be attached directly to the PLC. Not only can they communicate with other control systems, they can also perform reporting functions and diagnose their own failures, as well as the failure of a machine or process. Size is typically used to categorize today's PLC, and is often an indication of the features and types of applications it will accommodate. Small, non-modular PLCs (also known as fixed I/O PLCs) generally have less memory and accommodate a small number of inputs and outputs in fixed configurations. Modular PLCs have bases or racks that allow installation of multiple I/O modules, and will accommodate more complex applications. When you consider all of the advances PLCs have made and all the benefits they offer, it's easy to see how they've become a standard in the industry, and why they will most likely continue their success in the future.

CHAPTER 2. AN OVERVIEW OF SIEMENS S7-200 MICRO-CONTROLLER

2.1. Overview of an S7-200

STEP 7-Micro/WIN supports the S7-200 CPUs by giving you the features to set up and manage your application project. A project consists of the program you enter with STEP 7-Micro/WIN, along with the documentation you write for the program and the configuration you set up for the CPU.

You have the option of selecting either Ladder or Statement List as your programming language. With the S7-200 CPUs, you have a basic program structure that gives you flexibility in setting up any subroutines or interrupts that you program.

2.2. Introduction to the Simatic S7-200 Micro PLC

The Simatic S7-200 series is a line of micro-programmable logic controllers (Micro PLCs) that can control a variety of automation applications. Figure 2.1 shows an S7-200 Micro PLC. The compact design, expandability, low cost, and powerful instruction set of the S7-200 Micro PLC make a perfect solution for controlling small applications. In addition, the wide variety of CPU sizes and voltages provides you with the flexibility you need to solve your automation problems.

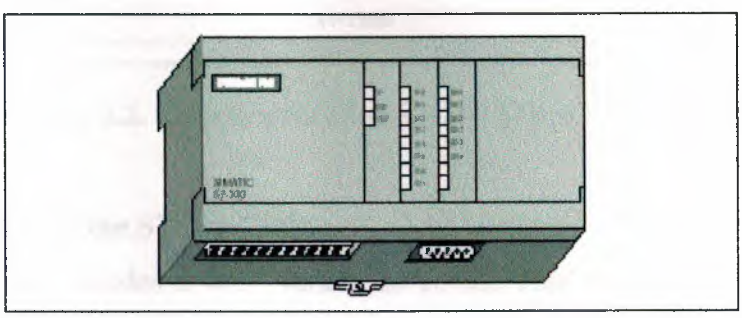


Figure 2.1. S7-200 Micro PLC

2.3. Comparing the Features of the S7-200 Micro PLCs

2.3.1 Equipment Requirements

Figure 2.2 shows the basic S7-200 Micro PLC system, which includes an S7-200 CPU module, a personal computer, STEP 7-Micro/WIN programming software, and a communications cable.

In order to use a personal computer (PC), you must have one of the following sets of equipment:

- A PC/PPI cable
- A communications processor (CP) card and multipoint interface (MPI) cable
- A multipoint interface (MPI) card. A communications cable is provided with the MPI card.

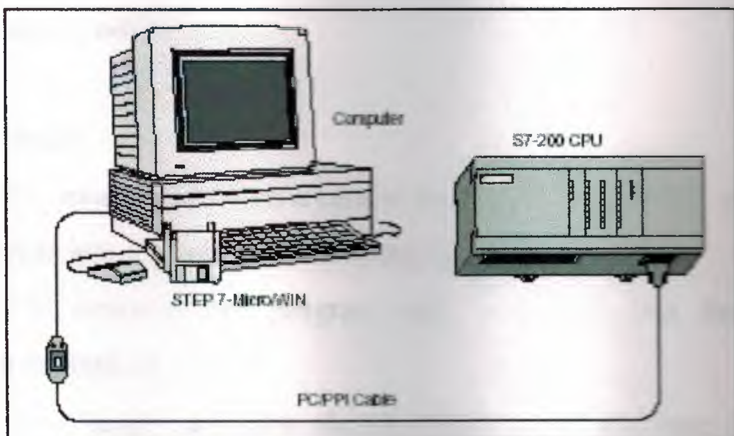


Figure 2.2. Components of an S7-200 Micro PLC System

2.3.2 Capabilities of the S7-200 CPUs

The S7-200 family indudes a wide variety of CPUs. This variety provides a range of features to aid in designing a cost-effective automation solution. Table 2.1 provides a summary of the major features of each S7-200 CPU.

Table 2.1. Summary of the S7-200 CPUs

Feature	CPU 212	CPU 214	CPU 215	CPU 216
Physical Size of Unit	160 mm x 80 mm x 62 mm	197 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm
Memory				
Program (EEPROM)	512 words	2 Kwords	4 Kwords	4 Kwords
User data	512 words	2 Kwords	2.5 Kwords	2.5 Kwords
Internal memory bits	128	256	256	256
Memory cartridge	None	Yes (EEPROM)	Yes (EEPROM)	Yes (EEPROM)
Optional battery cartridge	None	200 days typical	200 days typical	200 days typical
Backup(super capacitor)	50 hours typical	190 hours typical	190 hours typical	190 hours typical
Inputs/Outputs (I/O)				
Local I/O	8 DI / 6 DQ	14 DI / 10 DQ	14 DI / 10 DQ	24 DI / 16 DQ
Expansion modules (max.)	2 modules	7 modules	7 modules	7 modules
Process-image I/O register	64 DI / 64 DQ	64 DI / 64 DQ	64 DI / 64 DQ	64 DI / 64 DQ
Analog I/O (expansion)	16 AI / 16 AQ	16 AI / 16 AQ	16 AI / 16 AQ	16 AI / 16 AQ
Selectable input filters	No	Yes	Yes	Yes

2.4. Major Components of the S7-200 Micro PLC

An S7-200 Micro PLC consists of an S7-200 CPU module alone or with a variety of optional expansion modules

2.4.1. CPU Module

The S7-200 CPU module combines a central processing unit (CPU), power supply, and discrete I/O points into a compact, stand-alone device.

- The CPU executes the program and stores the data for controlling the automation task or process.
- The power supply provides electrical power for the base unit and for any expansion module that is connected.
- The inputs and outputs are the system control points: the inputs monitor the signals from the field devices (such as sensors and switches), and the outputs control pumps, motors, or other devices in your process.
- The communications port allows you to connect the CPU to a programming device or to other devices. Some S7-200 CPUs have two communications ports.
- Status Lights provide visual information about the CPU mode (RUN or STOP), the current state of the local I/O, and whether a system fault has been detected.

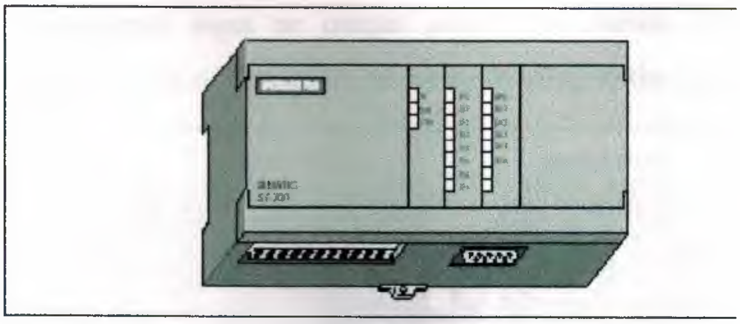


Figure 2.3 S7-212 CPU Module

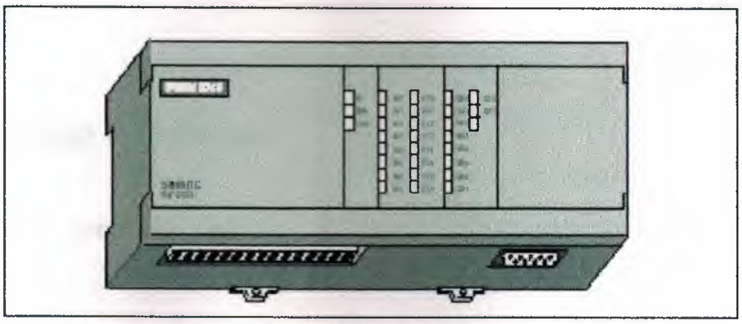


Figure 2.4 S7-214 CPU Module

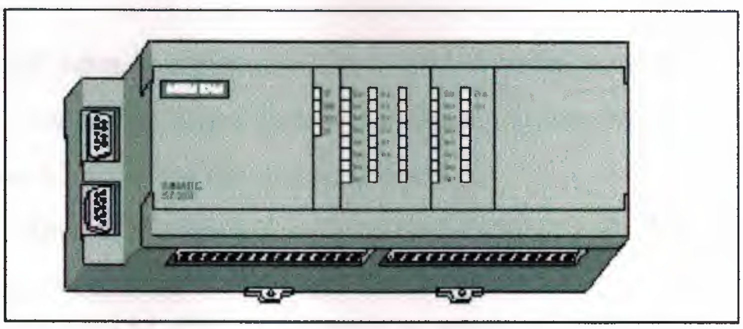


Figure 2.5 S7-215 and S7-216 CPU Module

2.4.2. Expansion Modules

The 7-200 CPU module provides a certain number of local I/O. Adding an expansion module provides additional input or output points. As shown in Figure 2.4, the expansion module comes with a bus connector for connecting to the base unit.

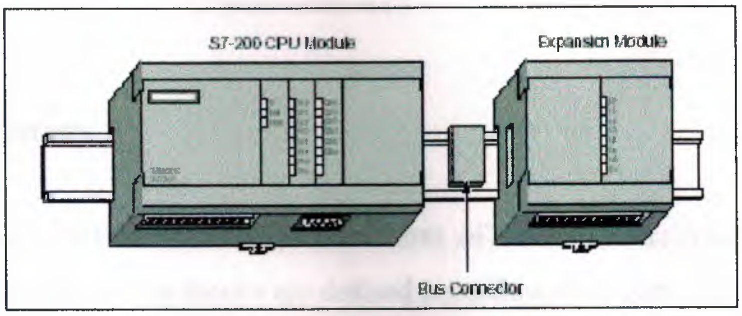


Figure 2.4 CPU Module with an Expansion Module

2.5. Programming Languages

2.5.1. Ladder Programs

In Ladder programs, the basic elements of logic are represented with contacts, coils, and boxes. A set of interconnected elements that make a complete circuit is called a network.

A hard-wired input is represented by a symbol called a contact. A normally open contact enables power flow when closed. A contact can also be normally closed. In this case, power flow occurs when the contact is opened.

A hard-wired output is represented by a symbol called a coil. When a coil has power flow, the output is turned on.

A box is a symbol for a complex operation performed within the CPU. The box simplifies programming of the operation. For example, boxes represent timers, counters, and math operations.

2.5.2. STL Programs

STL program elements are represented by a set of instructions for performing the desired functions. Instead of using the graphic display as shown by ladder programs, the STL program is shown in text format.

2.6. CPU Memory

The user memory in the S7-200 CPUs consists of three blocks: program, data, and configurable parameters. The blocks are defined according to usage:

- Program memory stores the user program.
- Data memory includes a temporary area for the program and storage of data. The temporary storage, calculations, and constants reside in data memory. Additionally, data for timers, counters, high-speed counters, and analog inputs and outputs are stored in data memory.
- Configurable Parameter memory stores either the default or the modified parameters of the program setup. The configurable parameters include items such as protection level, password, station address, and retentive range information.

2.7. Simatic S7-200 Application Areas

The SIMATIC S7-200 series is a line of micro-programmable logic controllers (Micro PLCs) that can control a variety of automation applications. Compact design, low cost, and a powerful instruction set make the S7-200 controllers a perfect solution for controlling small applications. The wide variety of CPU sizes and voltages, and the windows-based programming tool, give you the flexibility you need to solve your automation problems.

2.7.1. The S7-200 is characterized by the following properties

- Easy entry
- Uncomplicated operation
- Peerless real-time characteristics
- Powerful communications capabilities

The S7-200 achieves its full performance potential in distributed automation solutions thanks especially to the integrated ProFi Bus-DP connection. The application area of the SIMATIC S7-200 extends from replacing simple relays and contactors right up to more complex automation tasks.

The S7-200 also covers areas where previously special electronics have been developed for cost reasons. Application areas include:

- Baling processes
- Plaster & Cement mixers
- Suction Plants
- Centralized lubricating systems/flange lubricating systems
- Woodworking machinery
- Gate controls
- Hydraulic lifts
- Conveyor systems
- Food & Drink Industry
- Laboratories
- Modem applications via dial-up, leased-line, or radio remote monitoring (SCADA)
- Electrical Installations

2.7.2. Mechanical features include

- Rugged, compact plastic housing using SIMATIC's prize-winning design
- Easily accessible wiring and operator control and display elements protected by front covers
- Installs on standard horizontal or vertical DIN rail or direct cabinet mounting with built-in mounting
- Terminal block as permanent wiring assembly (optional)

2.7.3. Design features

- International standards; Meets the requirements through compliance with VDE, UL, CSA and FM standards.
- The quality management system used during manufacturing has ISO 9001 certification; and Data back up; the user program and the most important parameter settings are stored in the internal EEPROM. A heavy-duty capacitor provides additional back up for all data over longer periods (typically up to 50 or 190 hours). An optional battery module ensures that the data remain stored for 200 days (typically) after power failure.

2.7.4. Benefits of the S7-200

Complete Automation Solution

The SIMATIC S7-200 Micro PLC is a full-featured programmable logic control system offering stand-alone CPUs, micro-modular expansion capability, and operator interface solutions. Almost any application that requires automation, from basic discrete or analog control, to intelligent networked solutions, can benefit by using the powerful S7-200 family of products.

Value for OEMs

Wherever central controllers or expensive custom electronic control systems are used, the SIMATIC S7-200 offers a significantly more economical alternative. Our off-the-shelf, compact solution, is packed with features, and is accepted around the world as a Micro PLC standard.

Real-time Speed & Versatility

The SIMATIC S7-200 offers real-time control with Boolean processing speeds of 0.37μs per instruction. This fast execution speed, combined with our 20Khz high-speed counters, interrupts, and 20KHz pulse outputs, provide quick responses in demanding real-time applications. The S7-200 has over 200 instructions, including math, PID, For/Next loops, subroutines, sequence control, and more!

Integrated Communications

All S7-200 CPUs offer at least one RS485 communication port with speeds up to 187.5Kbaud. This not only provides fast access for programming and maintenance, but also allows you to build master/slave networks with up to 31 stations.

Using our Freeport capability can also connect non-S7-200 devices, such as bar code readers, intelligent machines, etc.. With Freeport, you can easily adapt the S7-200 CPU to virtually any serial ASCII protocol.

2.8 ADVANTAGES

2.8.1. Accuracy

In relay control systems logical knowledge's carries in electro-mechanical contactors, they can lose data because of mechanical errors. But PLC's are microprocessor-based system so logical data are carried inside the processor, so that PLC's are more accurate than relay type of controllers.

2.8.2. Data Areas

Data memory contains variable memory, and register, and output image register, internal memory bits, and special memory bits. This memory is accessed by a byte bit convention. For example to access bit 3 of variable memory byte 25 you would the address V25.3.

The following table shows the identifiers and ranges for each of the data area memory types:

Area Identifier	Data Area	CPU 212	CPU 214
I	Input	I0.0 to I7.7	I0.0 to I7.7
Q	Output	Q0.0 to Q7.7	Q0.0 to Q7.7
M	Internal memory	M0.0 to M15.7	M0.0 to M31.7
SM	Special Memory	SM0.0 to SM45.7	SM0.0 to SM85.7
V	Variable Memory	V0.0 to V1023.7	V0.0 to V4095.7

2.8.3. Logic Control of Industrial Automation

Everyday examples of these systems are machines like dishwashers, clothes washers and dryers, and elevators. In these systems, the output tend to be 220 V AC power signals to motors, solenoids, and indicator lights, and the inputs are DC or AC signals from user interface switches, motion limit switches, binary liquid level sensor, etc. Another major function in these types of controllers is timing.

2.8.4. Data Object

The S7-200 has six kinds of devices with associated data: timers, counters, analogue inputs, analogue outputs, accumulators and high-speed counters. Each device has associated data. For example, the S7-200 has counters devices. Counters have a data value that maintains the current count value. There is an also a bit value, which is set when the current value is greater than or equal to the present value. Since there are multiple devices are numbered from 0 to n. The corresponding data objects and object bits are also numbered.

The following table shows the identifiers and ranges for each of the data object memory types:

Area Identifier	Data Area	CPU 212	CPU 214
T	Timers	T0 to T63	T0 to T127
C	Counters	C0 to C63	C0 to C127
AI	Analogue Input	AIW0 to AIW0	AIW0 to AIW30
AQ	Analogue Output	AQW0 to AQW30	AQW0 to AQW30
AC	Accumulator	AC0 to AC3	AC0 to AC3
HC	High-speed Counter	HC0	HC0 to HC2

2.8.5. Flexibility

When the control needs a change, relay type of controllers modification are hard, in PLC, this chance can be made by PLC programmer equipment.

2.8.6. Communication

PLC's are computer-based systems. That's why, they can transfer their data to another PC, or they can take external inputs from another PC. With this specification we can control the system with our PC. With relays controlled system it's not possible.

CHAPTER 3. PRACTICAL IMPLEMENTATION WITH PLC

3.1 OVERVIEW

The main aim of this chapter is to control the Lamp and also control of the Industrial AC motor with same PLC program. In this PLC program we will use these inputs and outputs and the steps of program will be given.

Table 3.1 Inputs, Outputs, Timers, and Counters that used in the Program

INPUTS	OUTPUTS	TIMERS	COUNTERS
I0.0: Starting the System	**Q0.3:**Operation of the Lamp	**Timer On Delay:** T 37 for 10 seconds ON and T38 for 20 seconds OFF	**Count Up Counter:** **C0 for repeating the operation 5 times**
I0.1: Reset button of the Counter		**Timer On Delay CPU 212** T32 1ms T33-T36 10ms T37-T63 100 ms	**Count Up Counter CPU 212** C0-C63

3.1.1 Explanation of the Networks that will be used in the operation

Network-1: When the I0.0 input will be activated at same time T38 and C0 will be closed because of this System will start and Q0.3 will be activated.

Network-2: Timer T38 is still OFF, I0.0 input that activated in Network-1 will active the Timer T37 for 10 seconds (we used T37 type of Timer so T37: 100ms Preset Value 100: 100*100ms= 10 seconds). When T37 reached to the Preset Value it will active the Lamp will blow or AC motor will start to work.

Network-3: T37 activates T38 when it after 20 seconds (we used T38 type of Timer so T38: 100ms Preset Value 200: 200*100ms= 20 seconds) later When T38 reached to the Preset Value it will stop the Lamp or AC Motor.

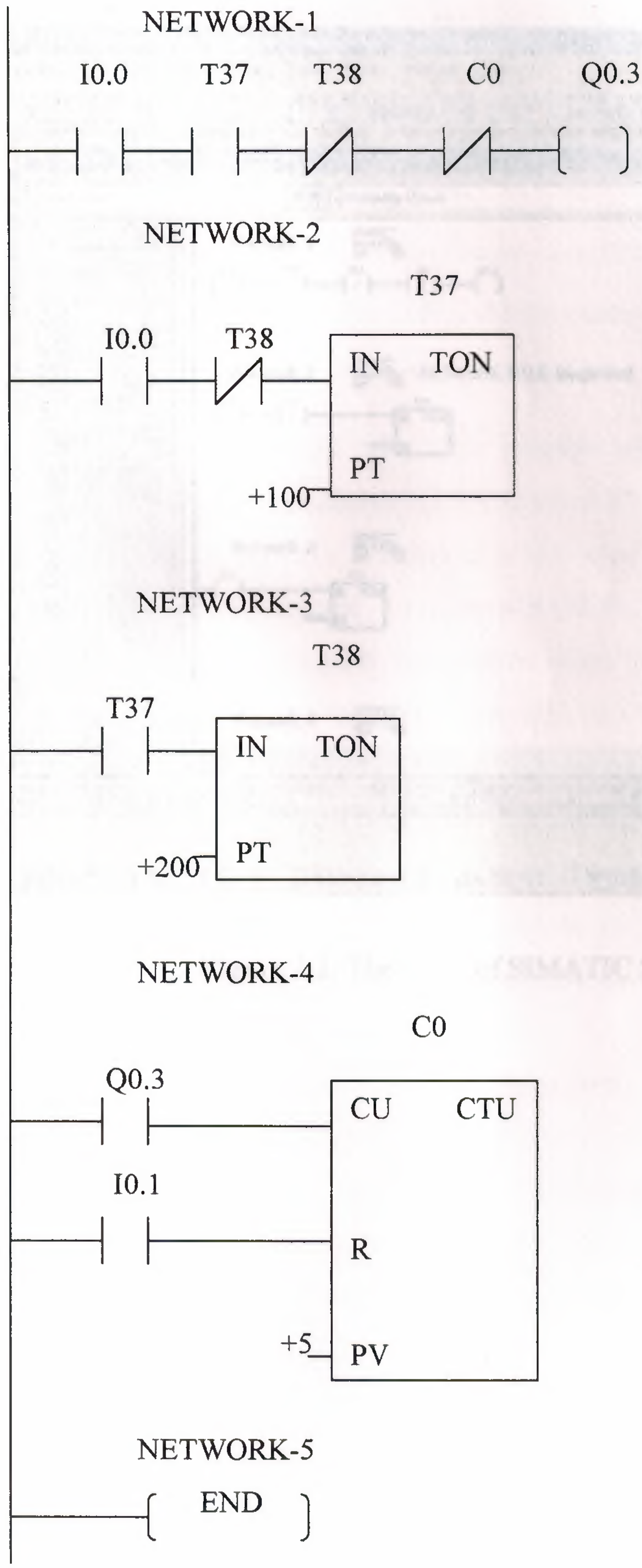
Network-4: In this Network Count Up Counter will enter to the system that we will use this counter as a C0; this counter will be activated by Q0.3 output that it will count 5 times means operation will be repeated 5 times.

Network-5: In the last network we must use END instruction this instruction will stop the program.

3.1.2. Statement List of the PLC Program

```
NETWORK 1
LD      I0.0
A       T37
AN      T38
AN      C0
=       Q0.3

NETWORK 2
LD      I0.0
AN      T38
TON     T37, +30

NETWORK 3
LD      T37
TON     T38, +100

NETWORK 4
LD      Q0.3
LD      I0.1
CTU     C0, +5

NETWORK 5
MEND
```

3.1.3. Ladder Diagram of the PLC Program



3.2. The View of SIMATIC S7-200 PLC Program

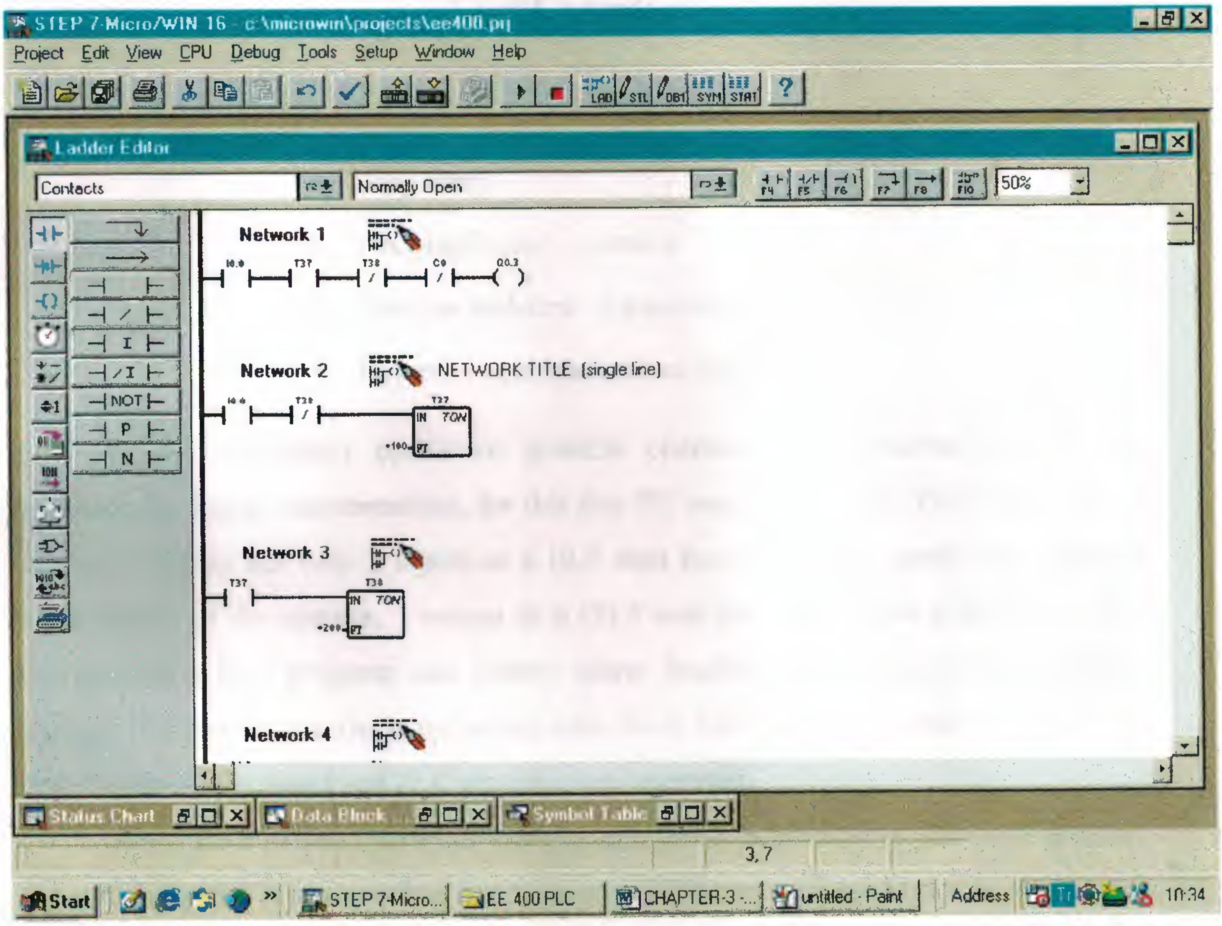


Figure 3.1. The View of SIMATIC S7-200 PLC Program

CONCLUSION

The S7-200 is summarized by the following properties:

- Easy entry
- Uncomplicated operation
- Peerless real-time characteristics
- Powerful communications capabilities

In the Laboratory conditions possible operations were carried out. In this operation the Lamp was controlled, for this first PC was connected to PLC. PLC has a 8 inputs, 6 outputs but only 2 inputs as a I0.0 start button and I0.1 used were used for Reset button of the counter, 1 output as a Q0.3 was the lamp output according to this informations a PLC program was written when loaded the program to PLC at same time the PLC outputs to the lamp circuit with about 220V was connected then when the start button was pressed and PLC program was operated.

Alternative of this operation was thought that it is possible to use same program like AC industrial Motor control but important difference we could not work the AC motor with a 220V because of that the present motor was operating with 115V AC from transformer and the same application was carried out. Both of them were controlled and successful results were obtained.

These operations can be improved by using the program that is comprehensively in Building Automation and Industrial Automations.

Over the world most of the new technologies are using by that investigate applications with PLC, for example the advertisement panels, Building Automations, and Industrial Automations and this project is applicable to all.

REFERENCES

[1] SIMATIC S7-200 Programmable Controller, System Manual

[2] Özgür Cemal Özerdem, Programmable Logic Controller and Programming, Near East University Press, Lefkoşa 2002.

[3] http://www.sea.siemens.com/controls/product/s7200/CNs7200.htm

APPENDIX-A

A.1 CPU 212 AC Power Supply, DC Inputs, Relay Outputs

Model Number 6ES7 212-1BA01-0XB0

General Features	
Physical Size (L x W x D)	160 x 80 x 62 mm (6.3 x 3.15 x 2.4 in.)
Weight	0.4 kg (0.9 lbs.)
Power Dissipation	6 W
User Program Size / Storage	512 Words EEPROM
User Data Size / Storage Data Retention	512 Words RAM 50 hr typical (8 hr minimum at 40° C)
Local I/O[1]	8 Inputs 6 Outputs
Maximum Number of Expansion Modules	2
Digital I/O Supported	64 Inputs 64 Outputs
Analog I/O Supported	16 Inputs 16 Outputs
Boolean Execution Speed	1.2 us/Instruction
Internal Memory Bits	128
Timers	64 Timers
Counters	64 Counters
High-Speed Counters	1 Software (2 KHz max.)
Analog Adjustments	1
Standards Compliance	UL 508 CSA C22.2 142 FM Class I, Division 2 VDE 0160 compliant CE compliant

Output Points	
Output Type	Relay, dry contact
Voltage Range	5 to 30 VDC / 250 VAC
Maximum Load Current	2 A / Point
Overcurrent Surge	7 A with contacts closed
Isolation Resistance	100 MΩ minimum (new)
Switching Delay	10 ms maximum
Lifetime	10,000,000 Mechanical 100,000 with Rated Load
Contact Resistance	200 mΩ maximum (new)
Isolation Coil to Contact Contact to Contact	 1500 VAC, 1 minute 1000 VAC, 1 minute
Short Circuit Protection	None

Input Points	
Input Type (IEC 1131-2)	Type 1 Sinking
ON State Range	15-30 VDC, 4 mA minimum 35 VDC, 500 ms surge
ON State Nominal	24 VDC, 7 mA
OFF State Maximum	5 VDC, 1 mA
Response Time I0.0 to I0.7	 0.3 ms maximum
Optical Isolation	500 VAC, 1 minute

Power Supply	
Voltage Frequency Range	85 to 264 VAC at 47 to 63 Hz
Input Current	4 VA typical, CPU only 50 VA maximum load
Hold Up Time	20 ms minimum from 110 VAC
Inrush Current	20 A peak at 264 VAC
Fusing (non-replaceable)	2 A, 250 V, Slow Blow
5 VDC Current	260 mA for CPU 340 mA for expansion I/O
Isolated	Yes. Transformer, 1500 VAC, 1 minute

DC Sensor Supply	
Voltage Range	20.4 to 28.8 VDC
Ripple/Noise (<10Mhz)	1 V peak-to-peak maximum
24 VDC Available Current Short Circuit Current Limit	180 mA < 600 mA
Isolated	No

1 The CPU reserves 8 input and 8 output image register points for local I/O.

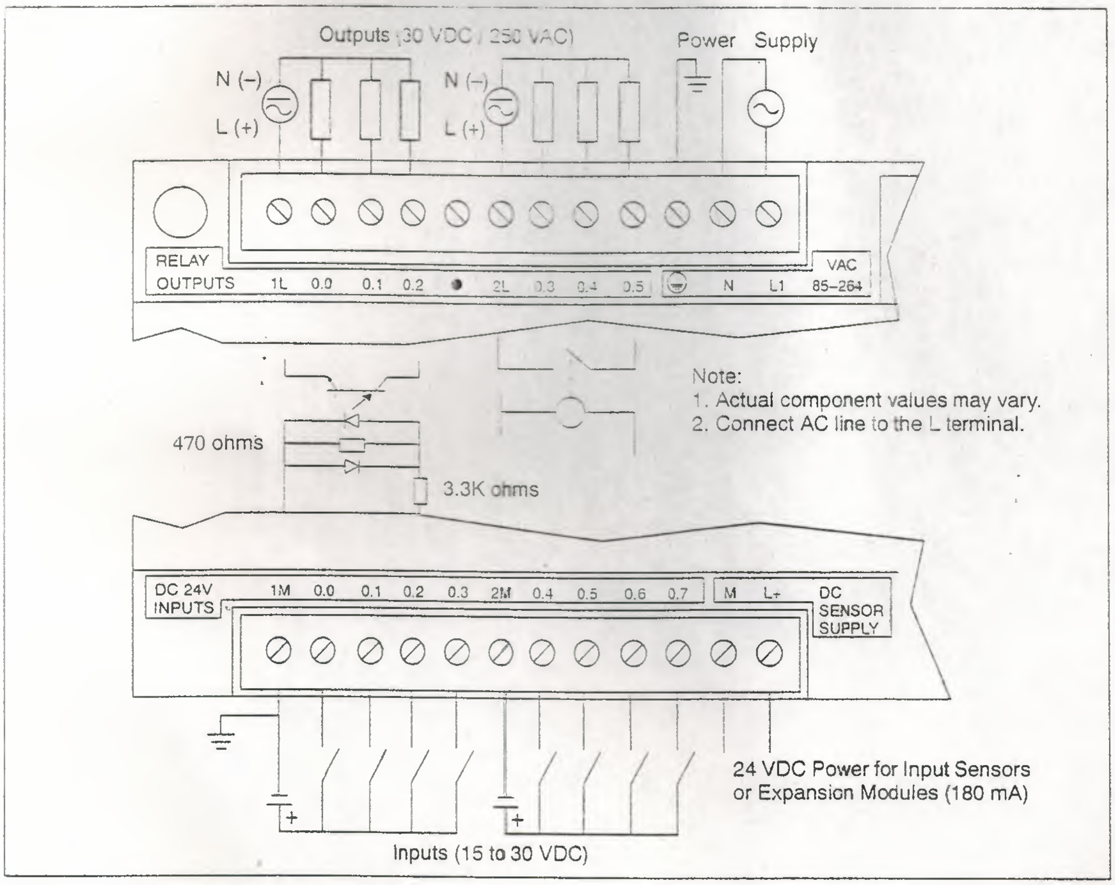


Figure A.1 Connector Terminal Identification for CPU 212 AC/DC Relay