NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Electrical and Electronic Engineering

CONVEYOR SYSTEM CONTROL BY PLC

Graduation Project
EE – 400

Student: Necip Fazıl Battal (20011431)

Supervisor: Özgür Cemal Özerdem

Lefkoşa – 2006

ACKNOWLEDGEMENTS

First and Foremost I would like to thank almighty ALLAH for giving me the strength and sincereness during this Project.

Special thanks to my supervisor Ass. Prof. Dr. Özgür Cemal ÖZERDEM, for his valuable advice and outmost support in completing this project. He always helped me a lot and I felt remarkable progress during my project preparation. Also I would like to thank to him for giving his time during my study and my advisering.

I would like to thank to Şahin BAYRAK, Yusuf DEMİREŞİK and Levent Hamdi UYGUN for their help and spiritual support during my bachelor study.

I also would like to thank to all my friends and specially Kadir TOK, Fatih GÜNDE, Nurullah SEZİK, Hakan AYAN, Şenol KOÇ, Nader SAMEER and Bashar JALAL who supported and helped me all the time. I never forget my wonderful times that I spent in Cyprus and Near East University with my good friends.

Last but not least, I would like to express my greatest thanks to my family, especially my parents for being patientfull during my undergraduate degree study. I could never have completed my study without their encouragement and endless support.

Necip Fazıl BATTAL

ABSTRACT

Production process has become easier with PLC use in today's industry. PLC technologies offer a wide variety of features including, advanced programming techniques, special communication capabilities, flexible automation solutions and many other important features. Today's PLCs offer faster scan time; ergonomical structure with space efficient, power saving design and large number of inputs/outputs.

Modern PLC technology offers communication with other control systems; it also provides recognition of failures and diagnosis of errors and special communication protocols for PLC-field devices interface.

In this project, conveyor system automation is realized with 12VDC motors, signal lamps, start/stop buttons and 24VDC diffuse reflective type photoelectric sensors. Therefore, CPU 212 8 inputs, 6 outputs (24VDC input/output type) PLC is used depending on the types and the number of inputs/outputs.

Conveyor System is performed and the indicator lamps, DC motors and sensors are operated physically with industrial considerations using the Programmable Logic Controller (PLC). DC motors are driven and controlled by 24VDC type relays.

CONTENTS

INTRODUCTION

Industrial Revolution has played a vital role in human life and it changed human perspective towards the world. Human expectations, needs, ideas, relations, life style, etc. have changed. Easy, quick, economic, ergonomic, portable and smart things have become popular all over the world. Technology was commented as an answer to human expectations and a huge market for trade.

Production Management and Quality Control has become the most important concern to increase in productivity and reduce the unit cost of a product.

Flexible systems are being replaced with bulk, complex and expensive systems. Imagine an automobile production factory where hundreds of cars are being produced and many sequential processes are required in production period. Production failure is a fatal error and for every second, company is loosing money because workers are employed but they cannot perform their tasks and orders are not prepared on time. Usually, problems are not simple. It is not easy to find the problem in a very complicated system. For example, a problem occurred in the robot arm of a conveyor belt. This robot signal flows and electrical energy transfer is distributed/controlled by a control panel. The problem may be related with internal part of robot or it may be energy failure or any other reasons. Imagine problem is determined. Signal transfer fails and it does not appear in the robot arm. It is so hard where the cable is broken or which cable is broken.

A Programmable Logic Controller (PLC) unit is a very nice device for control applications. It offers a flexible structure which enables the user to enter his/her defined programs to run in PLC. It takes few minutes to change the system parameters or the system itself. The operator does not deal with cables and connections too much but deals with program and its behaviors (reaction) to the system.

Another advantage of PLC, it offers many features (timers, counters, interrupts, arithmetic& logic operations etc.). The Operator can construct very complex systems easily (compare to classical control system combined with electromagnetic switches).
For example, a workman can break the concrete with a hammer in half hour or with a drill hammer within few minutes.

PLC's are widely used in industrial places, factories, hotels, art galleries etc. where the physical values such as pressure, temperature, humidity, etc. are very important during production process or for the environment conditions. In an art gallery, humidity is a very important thing because priceless paintings can be exhibited to the visitors and thus humidity can destroy these paintings (human effect on humidity in the air) if the humidity is not measured and controlled in short time intervals. A PLC makes the control of these physical parameters by sensing them with special sensors.

The Project Consists of Three Chapters, Conclusion, Appendix and Reference.
Chapter-1 presents the description of PLC, the types of PLC and the use of PLCs
Chapter-2 presents PLC structure, PLC Communication Protocols, PLC Programming and Programming Examples.
Chapter-3 presents the hard-wired control systems with examples, industrial control devices, commercial sensors and their applications including the other electronic devices and units used in conveyor carrying system prototype.

CHAPTER ONE
PLCs and PLC TYPES

1.1 What is a PLC?

A programmable logic controller, also called a PLC or programmable controller, is a computer-type (microprocessor based) electronic device which consists of integrated circuits rather than electromechanical devices, performs control functions in an industrial environment. PLCs include a programmable memory unit allocated to perform special functions such as logical, arithmetical, timing, counting, and sequencing operations with internal and external communication through digital or analog input/output modules to control industrial machines or process. Conveyor systems, food processing machinery, textile machinery, pharmaceutical products machinery, auto assembly lines, PID control applications, physical qualities control (temperature, pressure, humidity, etc.) packaging and material handling are samples of PLC usage in today's world.

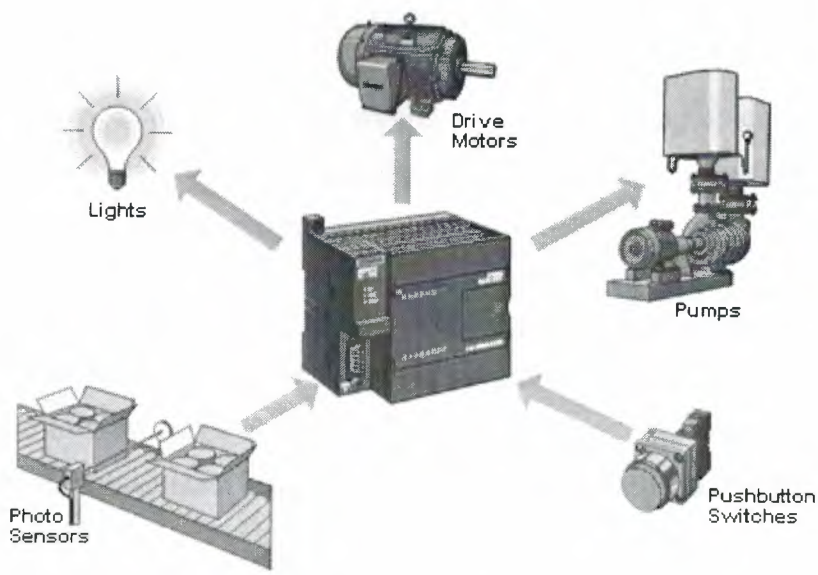


Figure 1.1 PLC application scheme

1.2 PLC History

In the late 1960's PLCs 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 schemes, one of which was based upon the PDP-8. The MODICON 084 brought the world's first PLC into commercial production.

Early machines were controlled by mechanical means using cams, gears, levers and other basic mechanical devices. Later, control system contained wired relay and switch control elements. These elements were wired as required to provide the control logic necessary for the particular type of machine operation. This was acceptable for a machine that never needed to be changed or improved. 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 which required strict adhesion to maintenance schedules. Troubleshooting was also quite tedious when so many relays are involved. Machine control panels are usually included many, possibly hundreds or thousands of individual relays.

The size could be out of concern but complicated initial wiring of so many individual devices! These relays would be individually wired together in a manner that would yield the desired outcome. Were there problems? Who knows?

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. They also had to survive the harsh industrial environment. The solution was 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 technologies were sequencer state-machines and the bit-slice based CPU. The AMD 2901 and 2903 were quite popular in Modicon and A-B PLCs. Conventional microprocessors lacked the power to quickly solve PLC logic in all but the smallest PLCs. As conventional microprocessors evolved, larger and larger PLCs were being based upon them.

However, even today some are still based upon the 2903.(ref Allen Bradley's PLC-3) Modicon has yet to build a faster PLC than 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 PLCs 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 analog world.

The 80's saw an attempt to standardize communications with General Motor's manufacturing automation protocol (MAP). It was also a time for reducing the size of the PLC and making them software programmable through symbolic programming on personal computers instead of dedicated programming terminals or handheld programmers. Today the world's smallest PLC is about the size of a single control relay!

The 90's have seen a gradual reduction in the introduction of new protocols, and the modernization of the physical layers of some of the more popular protocols that survived the 1980's. The latest standard (IEC 1131-3) has tried to merge PLC programming languages under one international standard. We now have PLCs that are programmable in function block diagrams, instruction lists, etc.

1.3 Why Use PLCs

The soft wiring advantage provided by programmable controllers is tremendous. In fact, it is one of the most important features of PLCs. Soft wiring makes changes in the control system easy and cheap. If you want a device in a PLC system to behave differently or to control a different process element, all you have to do is change the control program. In a traditional system, making this type of change would involve physically changing the wiring between the devices, a costly and time-consuming endeavor.

In addition to the programming flexibility we just mentioned, PLCs offer other advantages over traditional control systems.

These advantages include:

- Being solid state
- High reliability
- Simplicity of programming

- Small space requirements
- Very short response time
- Microprocessor based control
- Programmable memory
- Computing capabilities
- Communication availability (PLC-PC with RS232 or RS485 or USB protocols)
  Modern PLCs can be controlled remotely by TCP/IP protocol through internet via special softwares. It is also possible to access PLC data through internet
- Software timers/counters
  Reduces hardware/wiring cost and space requirements and decreases the frequency of connection errors i.e., electrical arc, disconnection between devices
- Modular I/O interface
- Human machine interface with optional visual control unit
- Easy maintenance, wiring and high MTBF(Mean Time Between Failures)
- Reduced costs
- Ability to withstand harsh environments
- Expandability

In a traditional system, all control devices are wired directly to each other

... In a PLC system, all control devices are wired to the PLC

Figure 1.2 PLC and hard-wired systems comparison

1.4 Types of PLC

PLCs can be classified into two groups according to their internal arrangement. Compact PLCs and Modular PLCs. Below are given the necessary explanation for both PLC types.

1.4.a Compact PLCs

Compact PLCs are manufactured as all the units of PLC take place in the same casing. They generally have less memory and accommodate a small number of inputs and outputs in fixed configurations with less prices. They are ideal in small industrial applications. There is also high capacity compact PLCs manufactured from different producers.

1.4.b Modular PLCs

In Modular PLCs, the units are separated (not on-board) and these units are combined and connected to each other to form a single PLC system. They can have different memory capacity, I/O numbers, power supply up to the necessary limits. Below are given a modular PLC and its features manufactured by IDEC Inc.

Figure 1.3 Idec FA3S Modular PLC (IDEC Inc.)

Idec FA3S Modular PLC Key Features:

- Four CPUs available
- 0.3-0.6 ms per basic instruction
- 1 to 8K program memory
- Supports up to 256 I/O
- Built in ASCII communications
- Interrupt and high speed I/O
- Fiber optic remote I/O
- Expansion RS232 communication ports

- Data link network with FA-NET RS485
- Data acquisition
- Special modules are available for data communication, fiber optic remote I/O, interrupt processing

1.5 PLC Configurations and Selection

There are many kinds of PLCs with different features and capacities from different manufacturers. Choosing correct type of PLC is vitally important for every programmer. Otherwise, it will waste the money unnecessarily and selected PLC model will not be used in desired applications. The following criteria should be considered for selecting a suitable and correct type of PLC:

1) Whether PLC will be used in fixed or various types of applications?
    a) Fixed applications – Compact PLC
    b) Expandable and various applications – Modular PLC

2) How many I/O modules range is suitable for the applications?
    a) Up to 32 I/O ports – Micro PLC
    b) Up to 128 I/O ports – Small PLC
    c) Up to 1024 I/O ports – Medium PLC
    d) Up to 4096 I/O ports – Large PLC

3) What kind of input/output and PLC feed powers you need in your control system? For instance, 24VDC Power Supply/ 24VDC in/ 24VDC out or 24VAC/24VAC/24VAC type or 240VAC/24VDC/ 24V Relay type etc.

4) Do your applications need PID functions, thermocouple output etc. in the PLC?

5) How much memory is needed for PLC program and user data?

6) Does your PLC control system need special communication configurations such as DeviceNET, Profibus, Control Net and Ethernet Option Modules?

7) Does your PLC system need Real Time Clock, high speed counters, mathematical functions(addition, subtraction, multiplication, division, square root, double precision, floating point, cosine functions), hardware input interrupts etc.

1.5.a Micro PLCs

Micro PLCs are designed and manufactured for the applications that need limited I/O units and minimum PLC configurations. They are widely used in conveyor systems, industrial machines control, etc. Some micro PLCs have high speed counters built in casing.

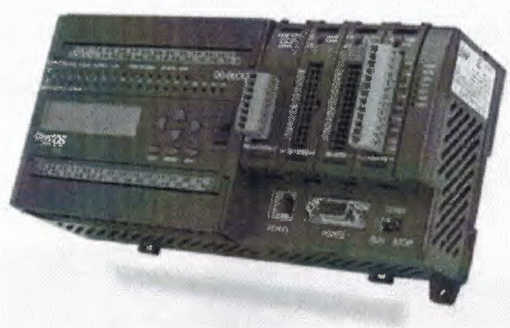


Figure 1.4 DL-06 Micro PLC (Direct Logic Inc.)

DL -06 Micro PLC Features:

- 20 inputs / 16 outputs
- 16-bit processor
- 14.8 Kbytes of total memory
- 229 instructions, including 8 PID loops
- Expandable up to 100 I/O
- PID capability, high-speed counting, floating point number handling
- Integrated high-speed inputs and pulse output
- ASCII in/out
- Thermocouple input
- Built-in real-time clock/calendar
- Two communication ports, including RS232/ 422/ 485 capability

- Supports networking for MODBUS RTU master/slave, RTD temperature input, a DeviceNET slave option module, an Ethernet option module, and a Profibus slave module

(***NOTE1:*** *Only one high-speed I/O feature may be in use at one time. You cannot use a high-speed input and the pulse output at the same time.*)

1.5.b Medium PLCs

Medium PLCs have a few hundred I/O units where the application or control process needs high capacities and special communication protocols. Medium PLCs are mainly used in traffic light controls, traffic guidance systems, sewage treatment plants, climate controlled warehousing, transportation systems, mining industry, chemical plants, material handling and packaging etc. Below are given Omron CJ1-Medium PLC key features.

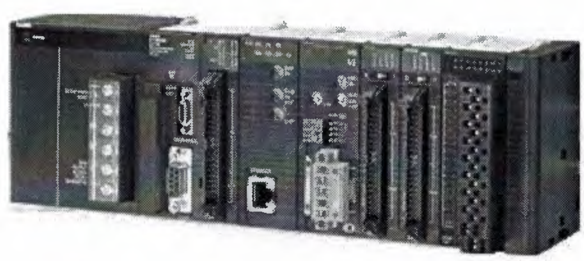


Figure 1.5 Omron CJ1 Medium PLC (Omron Electronics)

Omron CJ1- M Series PLCs Features:

- 120K words programming capacity
- up to 256K words data memory
- 20 – 40 nanosecond execution time
- up to 2560 local I/O, up to 3 racks
- CompoBus/S, RS-232C, 422/485, Ethernet, DeviceNet, ControllerLink, Profibus-DP, Protocol Macro.
- Rack-less design eliminates the need for a PLC rack, simplifying configuration and lowering system costs
- Flash Memory Cards store up to 64 MB for easy program transfer and data storage

1.5.c Large PLCs

Large PLCs are used in complicated, huge control systems which needs data manipulation, data acquisition, reporting, etc. Large PLCs have extremely high capacities, special communication protocols, PID modules, Math capabilities, extended data handling, fast speed processing etc. Large PLCs are typically used in Plant Engineering applications such as nuclear plants, chemical plants, etc.

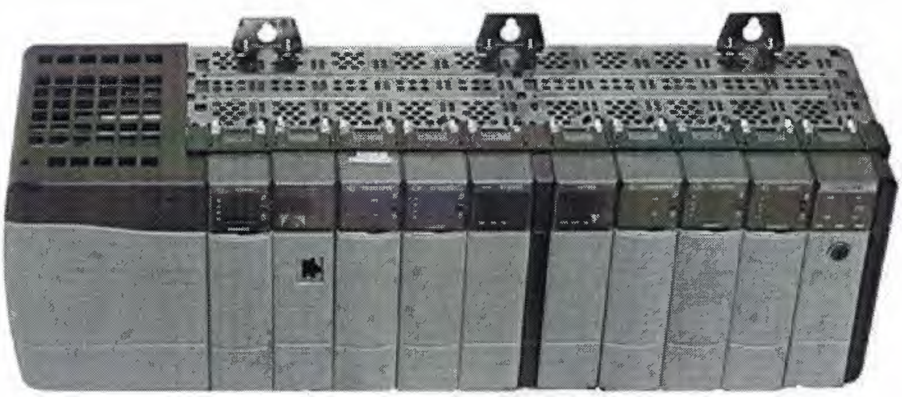


Figure 1.6 ControlLogix Series PLC

ControlLogix Series PLCs Features:

- Multiple processors
- 750K to 8M bytes with non-volatile options available
- Up to 128,000 digital or 4,000 analog I/O points with a wide range of digital, analog, and intelligent I/O modules
- Local and remote I/O
- Tag-based addressing

  It means you no longer have to worry about specific memory addresses. Logix automatically creates the memory structures required for status and diagnostic information for instructions and I/O modules
- EtherNet/IP, ControlNet, DeviceNet, Universal Remote I/O, DF1 / DH-485, DH+, RS-232 links, and other networks
- NetLinx open network

  It helps to save system design and programming time and needs less hardware because gateway controllers are not needed

- Integrated motion control

  It eliminates the need to create special programming to communicate, coordinate, and synchronize motion and sequential control among multiple axes of motion.)
- Drum timers and sequencers
- Timers/counters/shift registers
- Subroutines and interrupts
- Machine diagnostics
- Enhanced data handling (compare, data conversion, move register/file, matrix functions, block transfer, binary table, ASCII table, LIFO, FIFO)
- Math operations (Addition, subtraction, multiplication, division, square root, double precision, floating point, cosine functions)
- ViewAnyWare visualization products

CHAPTER TWO

PLC INTERNAL STRUCTURE
&
PROGRAMMING TECHNIQUES

2.1 How a PLC Operates?

A programmable controller operates by checking its inputs and running the program according to the states of inputs and results of the output continuously. Programmable controller performs this operation using basic sections:

- The central processing unit
- The input/output interface system
- Memory
- Power supply unit
- Communication Unit

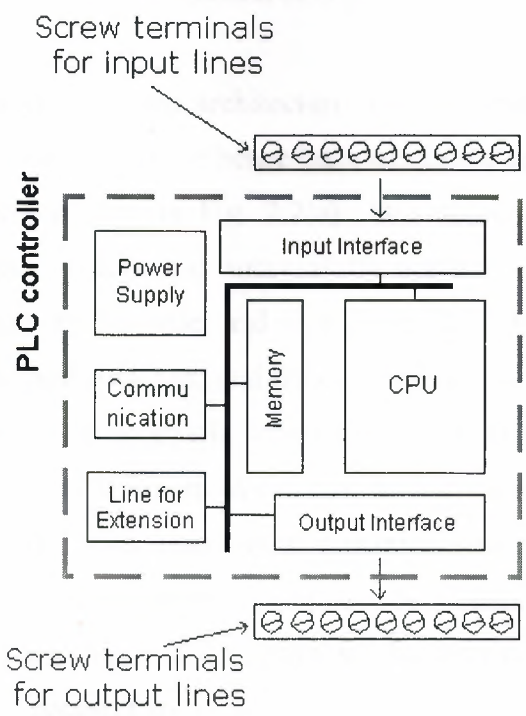


Figure 2.1 PLC Internal Structure Block Diagram

2.2 Central Processing Unit

Central Processing Unit (CPU) is the brain of a PLC controller. CPU itself can be either a microcontroller or a microprocessor. Early PLCs were 8-bit microcontrollers such as 8051, and now these are 16- and 32-bit microcontrollers.

There are two fundamental CPU architectures: The Harvard architecture and Von Neumann architecture.

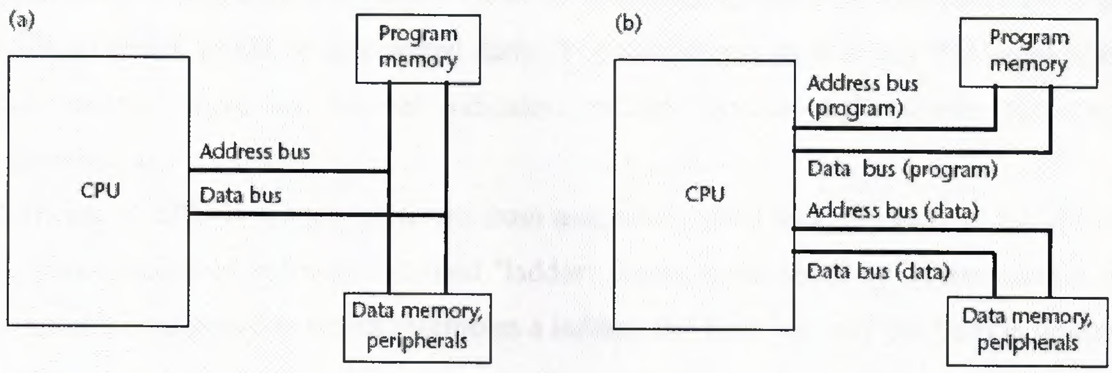


Figure 2.2 (a) The von Neumann structure; (b) the Harvard structure.

In the conventional von Neumann architecture, program and data memory share the same address and data buses, and are hence both within the same memory map. This is illustrated in very simple form in Fig. 2.2(a). This approach is simple, robust and practical, and has been widely and successfully applied. If data memory is being accessed, program memory lies idle, and vice versa. It is, however, possible to have more than one address and data bus, and hence to place data and program memory in different memory maps. This approach, sometimes called a Harvard structure, is shown in simple form in Fig 2.2(b). Instructions can now be fetched independently from, and if necessary simultaneously with instruction execution, thereby eliminating the von Neumann bottleneck. The two data buses can now be of different sizes, as can the two address buses. This allows each to be optimized for its own use, and has important implications in certain processor structures.

PLC is basically a microcontroller system (in von Neumann structure) with peripherals that can be digital inputs, digital outputs or relays as in our case. However, this is not an "ordinary" microcontroller system. Large teams have worked on it, and a checkup of its function has been performed in real world under all possible circumstances.

CPU takes care of communication, interconnectedness among other parts of PLC controller, program execution, memory operation, overseeing input and setting up of an output. PLC controllers have complex routines for memory checkup in order to ensure that PLC memory was not damaged (memory checkup is done for safety reasons). In general, CPU unit makes a great number of check-ups of the PLC controller itself so eventual errors would be discovered early. You can simply look at any PLC controller and see that there are several indicators in the form of light diodes for error signalization.

Software of CPU is entirely different from assemblers used thus far, such as BASIC or C. This specialized software is called "ladder" (name came about by an association of program's configuration which resembles a ladder, and from the way program is written out).

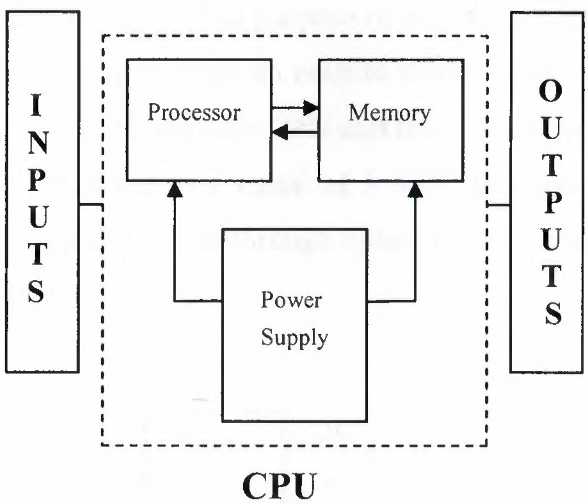


Figure 2.3 CPU Structure Block Diagram

2.2.1 The Input Interface System

Intelligence of an automated system depends largely on the ability of a PLC controller to read signals from different types of sensors and input devices. Keys, keyboards and by functional switches are a basis for man versus machine relationship. On the other hand, in order to detect a working piece, view a mechanism in motion, check pressure or fluid level you need specific automatic devices such as proximity sensors, marginal switches, photoelectric sensors, level sensors, etc. Thus, input signals can be logical (on/off) or analogue. Smaller PLC controllers usually have only digital input lines while larger also accept analogue inputs through special units attached to PLC controller. One of the most frequent analogue signals are a current signal of 4 to 20 mA and millivolt voltage signal generated by various sensors. Sensors are usually used as inputs for PLCs. You can obtain sensors for different purposes. They can sense presence of some parts, measure temperature, pressure, or some other physical dimension, etc. (ex. Inductive sensors can register metal objects). Other devices also can serve as inputs to PLC controller. Intelligent devices such as robots, video systems, etc. often are capable of sending signals to PLC controller input modules (robot, for instance, can send a signal to PLC controller input as information when it has finished moving an object from one place to the other.)

2.2.1.a The Input Interface Configuration

The input interface is placed between input lines and a CPU unit. This unit is also called as "input adjustable interface". The purpose of adjustable interface is to protect a CPU from disproportionate signals from an outside world. Input adjustment module turns a level of real logic to a level that suits CPU unit (ex. input from a sensor which works on 24 VDC must be converted to a signal of 5 VDC in order for a CPU to be able to process it). This is typically done through opto-isolation, and this function is given by the figure shown below.

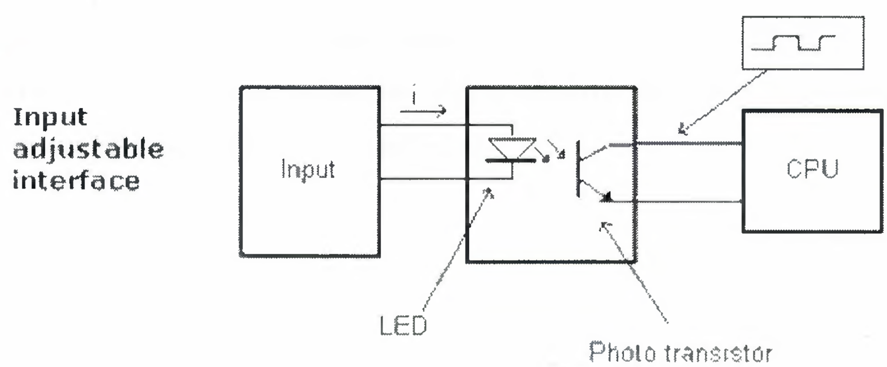


Figure 2.4 The input and the CPU interface

Opto-isolation means that there is no electrical connection between external world and CPU unit. They are "optically" separated, or in other words, signal is transmitted through light. The way this works is simple. External device brings a signal which turns LED on, whose light in turn incites photo transistor which in turn starts conducting, and a CPU sees this as logic zero (supply between collector and transmitter falls under 1V). When input signal stops LED diode turns off, transistor stops conducting, collector voltage increases, and CPU receives logic 1 as information.

2.2.2 The Output Interface System

Automated system is incomplete if it is not connected with some output devices. Some of the most frequently used devices are motors, solenoids, relays, indicators, sound signalization and similar. By starting a motor, or a relay, PLC can manage or control a simple system such as system for sorting products all the way up to complex systems such as service system for positioning head of CNC machine. Output can be of analogue or digital type. Digital output signal works as a switch; it connects and disconnects line. Analogue output is used to generate the analogue signal (ex. motor whose speed is controlled by a voltage that corresponds to a desired speed).

2.2.2.a The Output Interface Configuration

Output interface is similar to input interface. CPU brings a signal to LED diode and turns it on. Light incites a photo transistor which begins to conduct electricity, and thus the voltage between collector and emitter falls to 0.7V , and a device attached to this output sees this as a logic zero. Inversely it means that a signal at the output exists and is interpreted as logic one. Photo transistor is not directly connected to a PLC controller output. Between photo transistor and an output usually there is a relay or a stronger transistor capable of interrupting stronger signals.

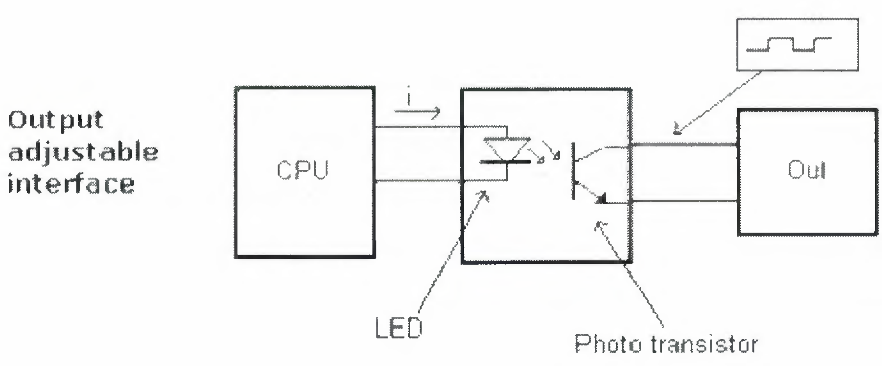


Figure 2.5 The output and the CPU interface

2.2.3 Memory

System memory (today mostly implemented in FLASH technology) is used by a PLC for a process control system. Aside from this operating system it also contains a user program translated from a ladder diagram to a binary form. FLASH memory contents can be changed only in case where user program is being changed. PLC controllers were used earlier EPROM memory instead of FLASH memory which had to be erased with UV light (chip should be exposed to UV light about 10 minutes) and programmed on programmers. With the use of FLASH technology this process was greatly shortened. Reprogramming a program memory is done through a serial cable in a program for application development. User memory is divided into blocks having special functions. Some parts of a memory are used for storing input and output status. The real status of an input is stored either as "1" or as "0" in a specific memory bit. Each input or output has one corresponding bit in memory. Other parts of memory are used to store variable contents for variables used in user program. For example, timer value, or counter value would be stored in this part of the memory.

2.2.4 Power Supply

Electrical supply is used in bringing electrical energy to central processing unit. Most PLC controllers work either at 24 VDC or 220 VAC. On some PLC controllers you'll find electrical supply as a separate module. Those are usually bigger PLC controllers, while small and medium series already contain the supply module. User has to determine how much current to take from I/O module to ensure that electrical supply provides appropriate amount of current. Different types of modules use different amounts of electrical current. This electrical supply is usually not used to start external inputs or outputs. User has to provide separate supplies in starting PLC controller inputs or outputs because then you can ensure so called "pure" supply for the PLC controller. Pure supply for the PLC means that industrial environment can not affect it damagingly. Some of the smaller PLC controllers supply their inputs with voltage from a small supply source already incorporated into a PLC.

The desired property of a good power supply is that, being able to regulate the voltage within predefined limits and supplying required current amount for the units. So that, the CPU and the other electronic devices will not be affected due to voltage changes and the required power will be provided.

2.3 Communication in PLCs

PLCs have two types of communications. (1) internal communication (2) external communication.

2.3.1 Internal Communication

Since PLCs are microprocessor based systems, they need external integrated circuits to control and communicate between CPU, RAM, memory and other peripherals. These communications are provided by three basic buses: Data Bus, Control Bus and Address Bus.

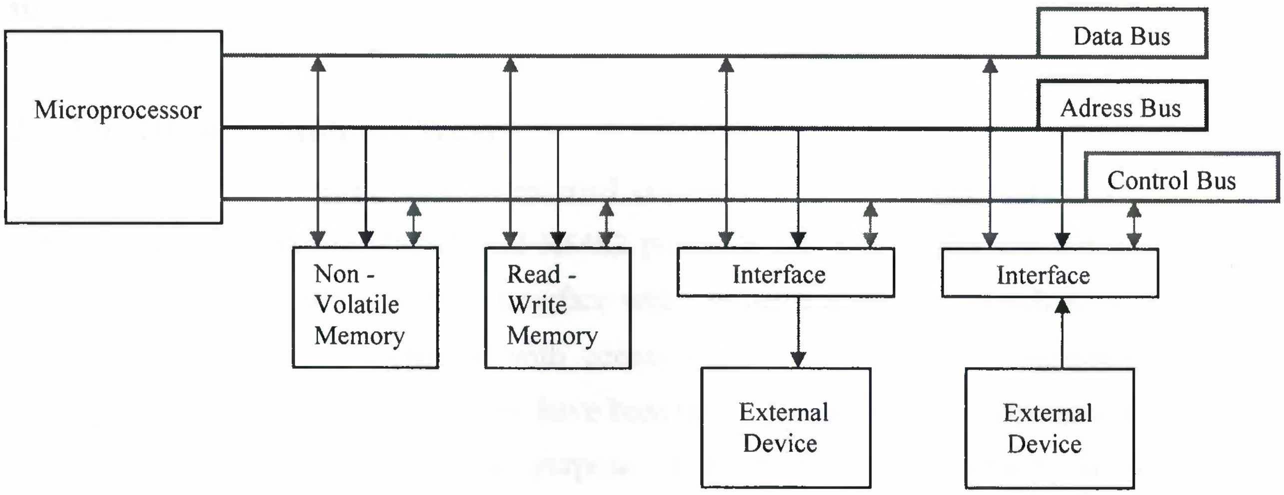


Figure 2.6 CPU and other peripherals communication

2.3.1.a Data Bus

The data bus is used for transferring data between microprocessor and external devices such as memory. Most embedded microprocessors have either an 8-bit or 16-bit data bus. The data bus is a bi-directional and data can flow either from microprocessor to devices or devices to microprocessor. Devices which can be ready by the processor must have tri-state logic connecting to the data bus to avoid contention.

2.3.1.b Address Bus

The address bus is used to specify the device, and the location within the device, that is being accessed. Most embedded microprocessors have either an 16-bit, 24-bit or 32-bit address bus. The address bus is unidirectional and is controlled by the microprocessor. All the devices must monitor the address bus, and only respond when the address assigned to them appears on the bus.

2.3.1.c Control Bus

A typical control bus might have the following lines:

RESET: (active-low) puts the processor and devices in a well-defined initial state on power-up

R/W: read not write – defines the direction of data flow

DAV: (active low) data valid – data on the data-bus is valid

DTA: (active low) data transfer acknowledge – data transfer is complete

IRQ: (active low) interrupt request

2.3.2 External Communication

PLC programmers or users need an external communication to access to PLC via personal computers (PCs). RS232, and RS482 protocols are used in this manner. In some complicated applications, PLC interface with sensors, transducers, switches, etc. is very boring, not easy to handle with connections. Today's software engineers, programmers, communication engineers have been developing new software programs, communication protocols for this purpose. For example, DeviceNET is a communication protocol for PLC-industrial devices interface with a smart solution in a flexible structure. Below are given the most important communication protocols used in PLCs.

2.3.2.a The RS-232 Protocol

The RS-232-C is designed to carry out point-to-point communications between two devices, a Data Terminal Equipment (DTE) and a Data Communication Equipment (DCE). The DTE is normally a computer and the DCE, a peripheral, which can be any kind of digital instrument.

The standard defines electrical, mechanical, and functional characteristics. The *electrical* characteristics include parameters such as voltage levels and cable impedance. The *mechanical* section describes the number of pins of the connectors and the function assigned to each pin. Although the connector itself is not specified, in practice DB-25 or DB-9 connectors are almost always used. The *functional* description defines the functions of the different electrical signals to be used.

Two important limitations of RS-232-C are the low transfer rates (up to 20 kilobits per second) and the maximum length of the wires (around 16 meters). To overcome these limitations, the new standards, RS449 (mechanical) and RS423 (electrical), have been defined. These new standards are upward-compatible with RS-232-C and can operate at data rates up to 10 megabits per second and at distances of up to 1200 meters. However, changing to a new standard is a long and costly process, so the penetration of this new standard in the market is, as yet, very limited.

Despite its limitations, RS-232-C communications are widely used, specially in low-cost installations. However, if more sophisticated communications are required, more specialized buses, such as the GPIB or field buses must be used.

2.3.2.b Fieldbuses

Fieldbuses are designed to communicate digital devices and instrumentation systems in industrial environments. Sensors, actuators, frequency converters (to control electrical motors), position controllers, PLC's and industrial computers are examples of the wide variety of devices which can be found in an industrial plant and which must communicate with each other. A fieldbus is an interconnection system which allows the communication of all these types of devices using a defined communication protocol. To connect a device to a determined fieldbus, the device must be specifically developed for that particular fieldbus.

Fieldbuses provide important advantages in relation to traditional industrial communications. Analysis of actual fieldbus projects show significant savings when they are used instead of conventional cabling arrangements and this is true even for small installations. Costs are also reduced in plant design, installations, and operations. During the project design stage, fully integrated user-friendly project design tools allow field devices to be integrated quickly and easily. As a result of this centralized engineering, the number of potential sources of error is substantially reduced. In the installation and cabling of the field devices the cost advantage is particularly clear. Instead of thick bundles of cables, just one two-wire bus cable is laid. In addition to saving time and material, a further benefit is that the likelihood of wiring errors is drastically reduced.

In operations, the benefits of fieldbus solutions are shown primarily in troubleshooting and maintenance. Errors can be more easily detected, analyzed and localized, and the stocking of spares is simplified. When subsequent plant upgrades arise, the field bus is the better choice, too. Instead of complex recabling, or utilization of expensive cable reserves, the existing bus is simply extended to the new field devices.

During 1990s a significant number of field bus definitions were achieved. However, each bus definition presents particular characteristics and, consequently, some buses are more suitable for determined installations than others. For example, the AS-I bus (1993) is oriented to low-level communications, connecting the sensors and actuators of a plant with its automation system. Other buses, such as PROFIBUS (1994/1995), are devoted to managing higher level communications, which allows communication among PLC's, drive units, I/O modules and industrial computers. A peculiar type of field bus is the CAN bus (1995), which was specifically developed for the automotive industry. The CAN bus is used to communicate and control the great number of digital devices which can be found in a vehicle: ABS breaks, airbags, electric windows, etc. Other important examples of field buses in the market are INTERBUS-S (1984), DeviceNet (1994) and the Foundation Fieldbus H1 (1995).

Major international efforts for the standardization of Fieldbus can be subdivided into two categories: the first one is based on the use of the Data Link Layer (DLL) of a centralized control protocol such as FIP Fieldbus, and the second one is based on the distributed control protocol.

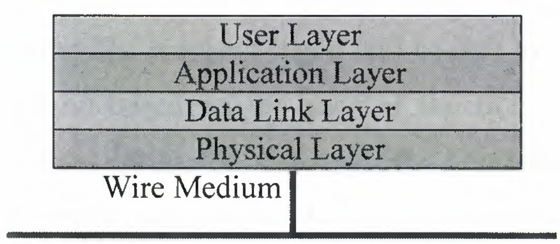


Figure 2.7 Common Fieldbus Protocol Layers

2.4 PLC Operation Principle

A PLC works by continually ***scanning*** a program. We can think of this scan cycle as consisting of three important steps. There are typically more than three steps but we can focus on the important parts simply. Typically the others are checking the system and updating the current internal counter and timer values.

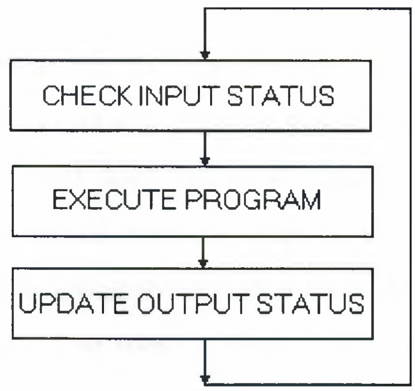


Figure 2.8 Illustration of a scan process

Step 1) **CHECK INPUT STATUS** -First the PLC reads (checks) each input device such as sensors, transducers, limit switches, push buttons, etc. to determine if it is on or off. In other words, is the sensor connected to the first input on? How about the second input? How about the third... It records the state of each input into its memory location to be used in the following step.

Step 2) **EXECUTE PROGRAM** -Next the CPU executes the program, starting with the first instruction and proceeding to the end instruction. Maybe your program said that if the first input was on then it should turn on the first output. Since it already knows which inputs are on/off from the previous step it will be able to decide whether the first output should be turned on based on the state of the first input. It will store the execution results for use later during the next step. The immediate I/O instructions give you immediate access to inputs and outputs during the execution of either the program or an interrupt routine. If you use interrupts in your program, the interrupt routines that are associated with the interrupt events are stored as part of the program. The interrupt routines are not executed as part of the normal scan cycle, but are executed when the interrupt event occurs (which may be at any point in the scan cycle).

Step 3) **UPDATE OUTPUT STATUS** -Finally the PLC updates the status of the outputs. It updates the outputs based on which inputs were on during the first step and the results of executing your program during the second step. Based on the example in step 2 it would now turn on the first output because the first input was on and your program said to turn on the first output when this condition is true. After the third step the PLC goes back to step one and repeats the steps continuously.

The time that a PLC spends during the scan process is called *"**scan time**"*. In a scan time period, PLC checks each input and store the state of these inputs, updates its inputs and results output as activation or deactivation of related output. Scan time depends on two parameters: (1) the amount of memory taken by the control program and (2) the type of instructions used in the program (which affects the time needed to execute the instructions). The time required to make a single scan can vary from a few hundred nanoseconds to 50 milliseconds.

2.5 PLC Programming Languages

There are a lot of PLC manufacturer companies. For this reason, PLCs are programmed in different languages. Ladder, Boolean and transition function chart are main languages. Transition function chart originated in France. It is known as ***"Grafcet"***. This language is not common, so that it will not be explained. Ladder program was developed for technicians who are already familiar to classical electrical control systems. Some PLC manufacturers offer Boolean language to program their PLCs. Boolean language is similar to assembly programs. Instructions are the abbreviations of the functions. Modern PLCs can be programmed in high- level languages such as C, C++, etc. Below are given the necessary explanation for each programming language.

2.5.a Ladder Language

PLCs were manufactured as an alternative to the classical electrical control systems. Since the users of PLCs are mostly electricians, technicians and engineers, the language of PLC should be familiar to them. The ladder language is the most suitable PLC language for these people because the ladder language is very similar to relay based control systems. Ladder language is composed of symbols with addresses, destinations and other parameters according to the instruction.

Below are given a hard-wired control circuit and its equivalent PLC ladder representation.

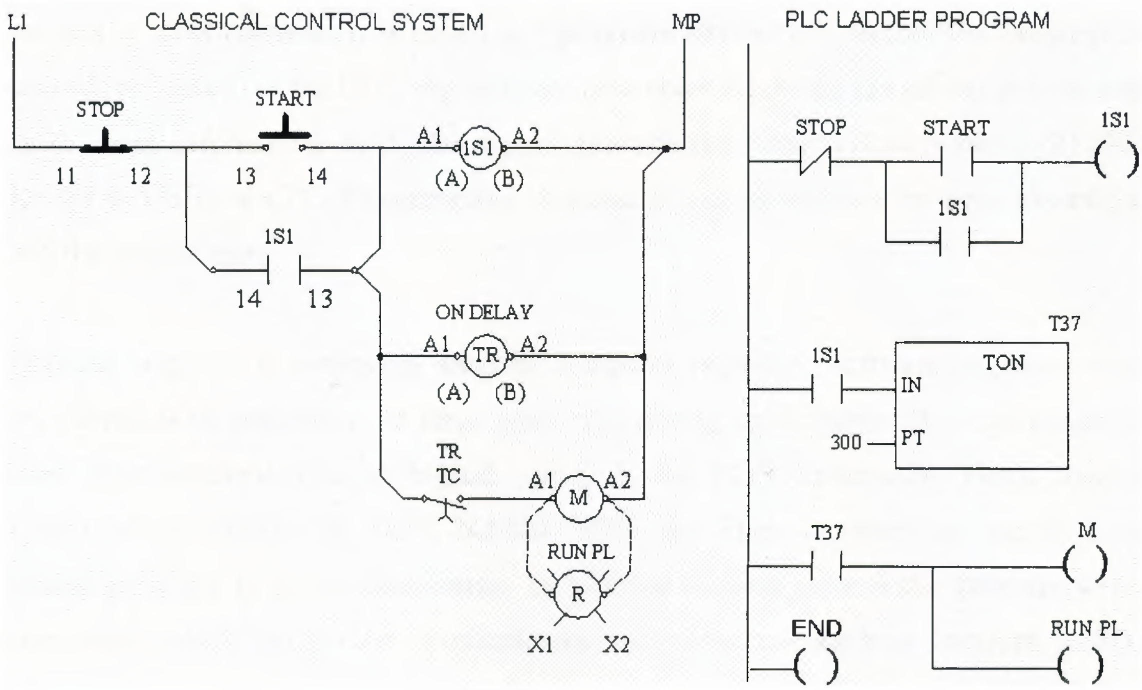


Figure 2.9 Classical Control Circuit and Its PLC Ladder Program

In the above given circuit, it is clearly shown that the motor will start after 30 seconds if the start button is pressed. Actually, in the figure it is not mentioned when the motor will start exactly. The electrician or technician has to select correct type of timer relay for desired time interval and adjust it to 30 second position. As you see, when the operator press the start button and release it, the contactor will be de-energized and timer will stop operating. To prevent this event, start contact is connected to N/O contact of contactor in parallel. This is called as ***"locking process"*** in electrical control systems. This will be explained in details later.

In the same figure, its equivalent PLC representation is shown. You can see how PLCs are flexible compare to classical control systems. Let us change the duration for the motor start and make it 30 minutes. Is it possible to use same timer for this time interval? Certainly it is not. Timers are made in fixed time intervals but there are also timers can be programmed daily, weekly or monthly such timers are called as ***" switch computers "***. These switch computers are very expensive, a few hundred dollars.

2.5.b Boolean Language

Some PLC manufacturers support Boolean language in their PLC softwares. PLC software program converts the program into a suitable form that PLC CPU understands, process it. This conversion is called as **"program assembly"**. Before the program is sent (downloaded) to the PLC, the software assembles it into the hexadecimal form and sends these information as digital signals through the communication ports (RS232, RS485 or USB) of a PC. Programming language is just an interface between assembler and the programmer.

Boolean language is commonly used by computer engineers, software engineers who are related with automation in large plant engineering applications. Boolean language uses short abbreviations of logical operands for PLC commands. These logical commands are AND, OR, NOT, NAND, NOR, etc. Their functions are the same as logical gates use in digital electronics. In addition to these commands, there are some commands which are similar to microprocessor commands such as interrupt, LIFO, FIFO, and move, rotate, shift commands.

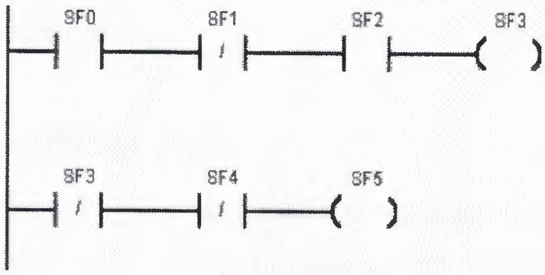
Below are given some Boolean commands for different PLCs from different manufacturers.

Table 2.1 Boolean language commands from different PLC manufacturers

Logical Expression	S5-90U	S7-200	OMRON	HITACHI	MITSUBISHI
AND	A	A	AND	AND	AND
OR	O	O	OR	OR	OR
NOT	N	NOT	NOT	NOT	I
NAND	AN	AN	AND NOT	AN I	AN I
NOR	ON	ON	OR NOT	OR I	OR I
AND BLOCK	A(	A LD	AN LD	AN B	AN B
OR BLOCK	O(	O LD	OR LD	OR B	OR B
LOAD	-	LD	LD	LD	LD
INPUT	I	-	-	-	-
LOAD COMPLEMENT	-	LD N	LD NOT	LD I	LD I
OUTPUT	Q	Q	OUT	OUT	OUT
DESIGNATE TO OUTPUT	=	=	OUT	OUT	OUT
END OF PROGRAM	BE	MEND	END	END	END

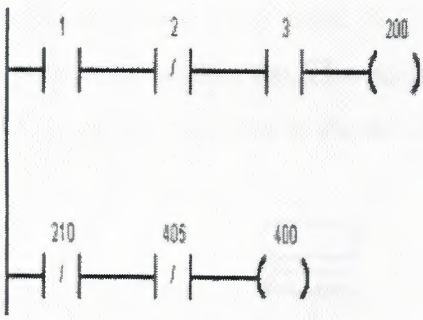
Ladder Program | **Boolean Program**

FESTO



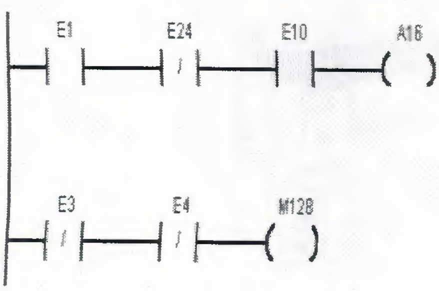

```
LD FLAG 0
AND NOT FLAG 1
AND FLAG 2
= FLAG 3
LD NOT FLAG 3
AND NOT FLAG 4
=FLAG 5
LD PROG 0
```

IDEC



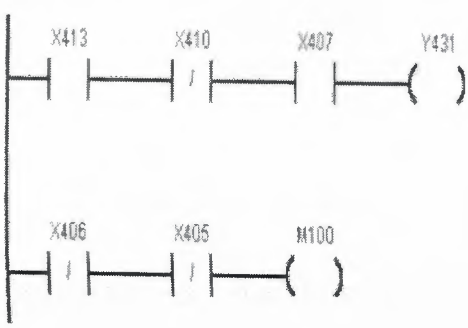

```
LOD 1
AND NOT 2
AND 3
OUT 200
LOD NOT 210
AND NOT 405
OUT 400
END
```

AEG




```
U     E1
UN    E24
U     E10
=     A16
UN    E3
UN    E4
=     M128
PE
```

MITSUBISHI




```
LD    X413
ANI   X410
AND   X407
OUT   Y431
LDI   X406
ANI   X405
OUT   M100
```

Figure 2.10 Ladder and Boolean language programs from various PLC manufacturers

2.6 Introduction to PLC Programming

In this section, Siemens S7-200 PLCs will be introduced and the ladder language will be used for PLC programming.

2.6.1 Siemens S7-200 PLCs

The S7-200 Micro PLC is the smallest, compact type programmable controller in SIMATIC S7 family. The central processing unit is inside the PLC unit. Input/Output units are placed on the main unit and these units are not separable from the PLC. Inputs take the analog/digital information signals from field devices, such as switches, sensors, transducers, etc. and introduce them to the PLC. Outputs actuate (control) other devices, such as motors, pumps, solenoids, etc. The programming port is the connection between PLC and PC. PLC programs are sent to the device via this communication port.

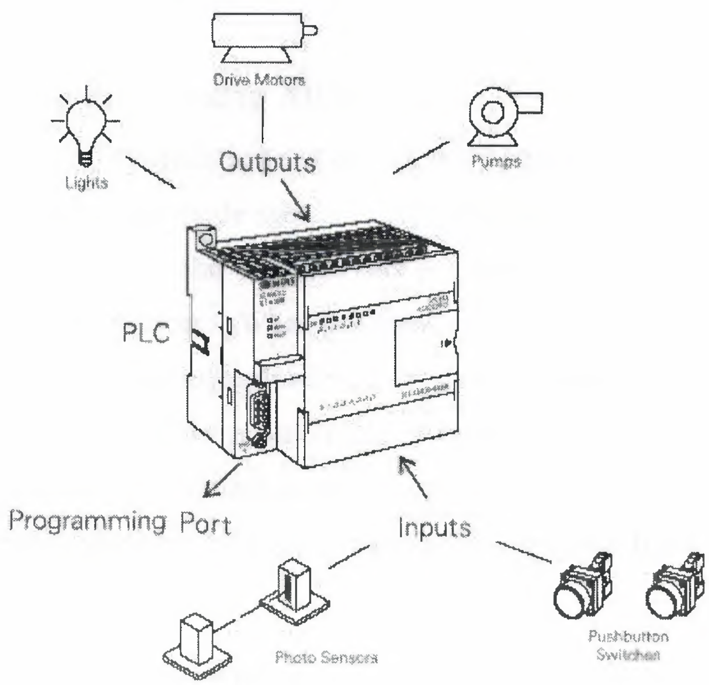


Figure 2.11 Siemens S7-200 Micro PLC

The S7-200 family members are classified according to their CPU types. There are four S7-200 CPU types: S7-221, S7-222, S7-224, S7-226, and S7-226XM and three power supply configurations for each type. The model description indicates the type of CPU, the power supply, the type of input, and the type of output (Figure 2.12).

Table 2.2 The S7-200 CPU Types and Descriptions

Model Description	Power Supply	Input Types	Output Types
221 DC/DC/DC 221 AC/DC/Relay	20.4-28.8 VDC 85-264 VAC 47-63 Hz	6 DC Inputs 6 DC Inputs	4 DC Outputs 4 Relay Outputs
222 DC/DC/DC 222 AC/DC/Relay	20.4-28.8 VDC 85-264 VAC 47-63 Hz	8 DC Inputs 8 DC Inputs	6 DC Outputs 6 Relay Outputs
224 DC/DC/DC 224 AC/DC/Relay	20.4-28.8 VDC 85-264 VAC 47-63 Hz	14 DC Inputs 14 DC Inputs	10 DC Outputs 10 Relay Outputs

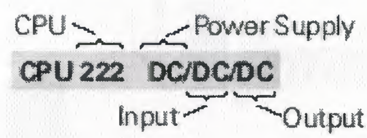


Figure 2.12 S7-200 CPU Description

2.6.1.a Mode Switch and Analog Adjustment

The mode switch is a toggle switch that can be adjusted to three positions. RUN, STOP and TERM positions. When the mode switch is in RUN position, the CPU starts to run the program loaded to the PLC and executes this program. According to the states of inputs, the outputs can vary in time. When the mode switch is in the STOP position the CPU is stopped. When downloading/uploading a program into the PLC, the PLC should be in this position. In the TERM position, the programming device can select the operating mode. The analog adjustment is used to increase or decrease values stored in special memory. These values can be used to update the value of a timer or counter, or can be used to set limits.

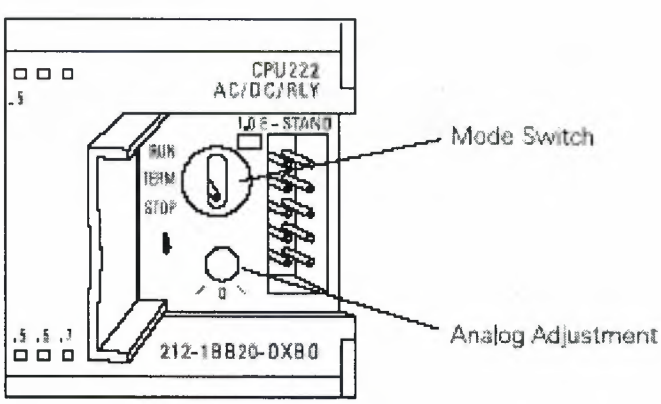


Figure 2.13 Mode Switch and Analog Adjustment

2.6.1.b Inputs

Input devices can be pushbuttons, switches, limit switches, proximity switches, photoelectric sensors and other sensing devices. These devices are connected to the terminal strip under the bottom cover of the PLC (See Figure 2.14).

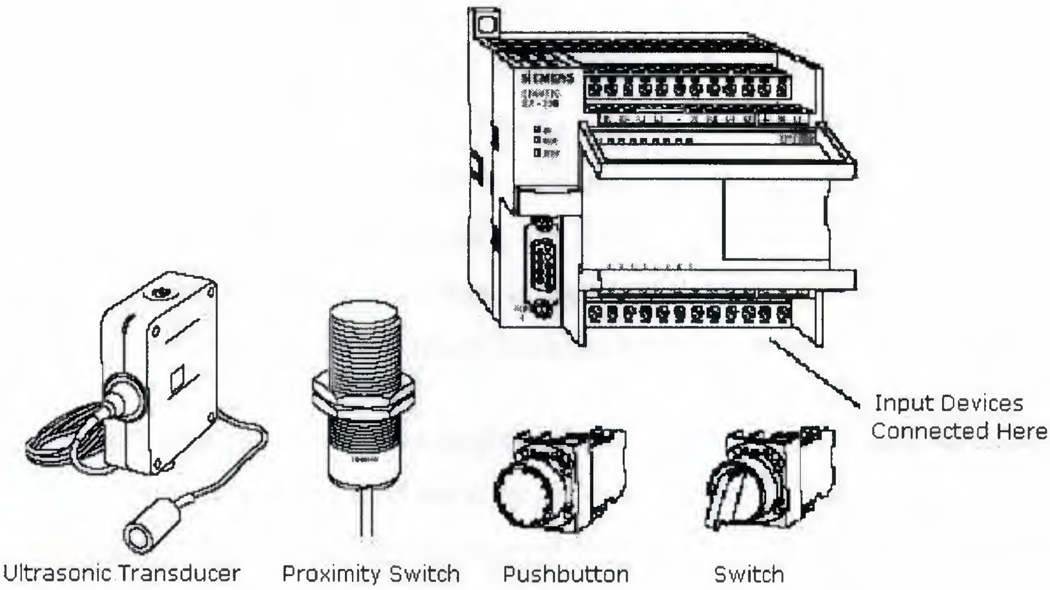


Figure 2.14 Siemens S7-200 Inputs

2.6.1.c Outputs

Typical output devices are signal lamps, contactors, relays, solenoids, output transducers, etc. These devices are connected to the terminal strip under the top cover of the PLC. It is not necessary to connect them to the PLC when testing a program to see the effects of the program. The LED is lit for related output if that output is active.

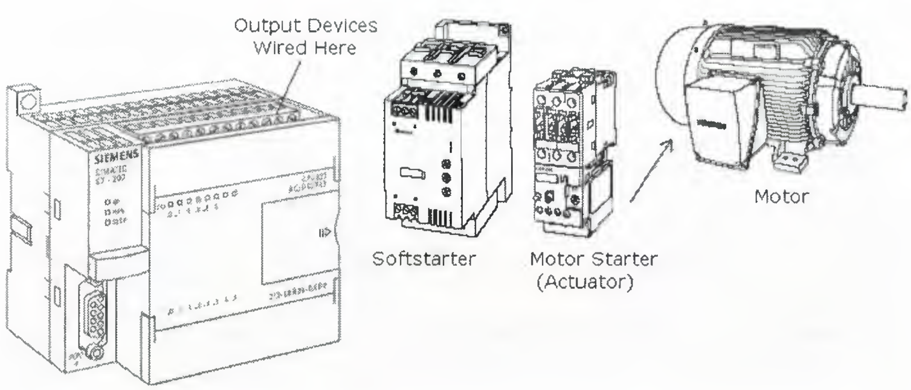


Figure 2.15 Siemens S7-200 Outputs

2.6.2 Timers

Timers are devices that count increments of time. You can use timers to implement time-based counting functions. Timers are used in traffic lamps, conveyors, chemical process applications where the time is essential for the applications.

The S7-200 provides two different timer instructions: the On-Delay Timer (TON), and the Retentive On-Delay Timer (TONR). Both TON and TONR timers time up while the enabling input is on. The timers do not time up while the enabling input is off, but when the enabling input is off, a TON timer is reset automatically and a TONR timer is not reset and holds its last value. Therefore, the TON timer is best used when you are timing a single interval. The TONR timer is appropriate when you need to accumulate a number of timed intervals. S7-200 timers have the following characteristics:

- Timers are controlled with a single enabling input, and have a current value that maintains the elapsed time since the timer was enabled. The timers also have a preset time value (PT) that is compared to the current value each time the current value is updated and when the timer instruction is executed.

- A timer bit is set or reset based upon the result of the comparison of current value to the preset time value. When the current value is greater than or equal to the preset time value, the timer bit (T-bit), is turned on.

- When you reset a timer, its current value is set to zero and its T-bit is turned off. You can reset any timer by using the Reset instruction, but using a Reset instruction is the only method for resetting a TONR timer. Writing a zero to a timer's current value does not reset its timer bit. In the same way, writing a zero to the timer's T-bit does not reset its current value.

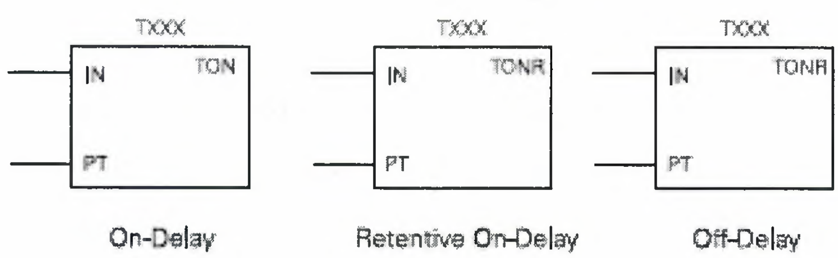


Figure 2.16 Types of Timers

2.6.2.a On-Delay Timer (TON)

On-Delay Timer (TON) starts to time up when its input is activated. It continues timing up if its input is still on. If the input is de-activated at any time during the operation, the timer is reset and it will stop timing up. When the preset time value is reached the related contact of the timer is activated.

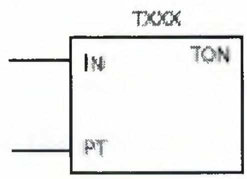
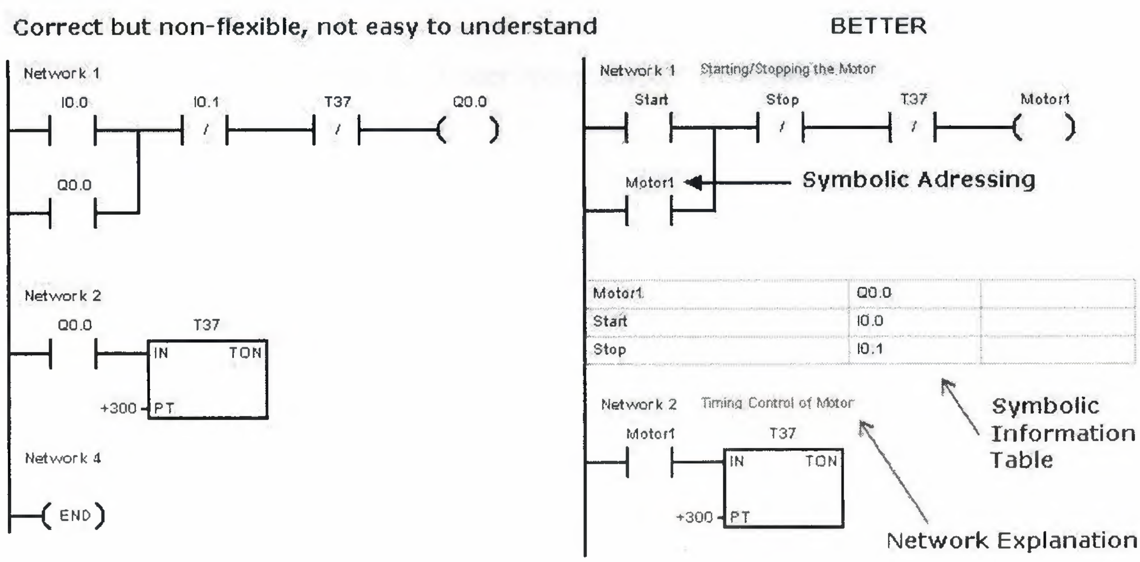


Figure 2.17 On-Delay Timer

Example 1)

A motor is controlled by PLC and the control system performs the following function: Motor1 operates for 30 seconds and stops after start button is pressed. The operator can stop the motor by pressing the stop button.

SOLUTION:



When start button is pressed, "Start" contact becomes on (logic 1). "Stop" and "T37" normally closed contacts are on if the stop button is not pressed and T37 timer is not active. Thus, "Motor1" output coil becomes active (logic 1). When start button is released, "Start" contact becomes off and "Motor1" output coil becomes off respectively. If we put "Motor1" contact parallel to "Start" contact, the motor will not stop after we release start button. This process is called as ***"locking process"***. T37 timer starts to time up because its input (Motor1 contact) is enabled. When preset time is reached, its Normally Closed (N/C) contact opens the contact and motor stops.

2.6.2.b Retentive On-Delay (TONR)

The Retentive On-Delay Timer (TONR) operation principle is similar to TON type timer. It starts to time up when its input is activated. The difference is that it does not reset when its input is de-activated. The only way to stop the timer is to use a reset instruction. Preset time value is selected according to the table (Table 2.3). For instance, for 30 seconds time duration, T37 can be written with preset value =300.

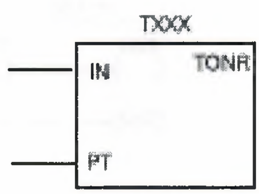


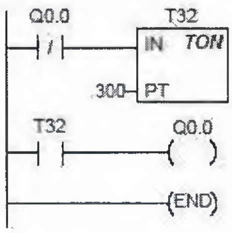
Figure 2.18 Retentive On-Delay (TONR) Timer

Table 2.3 Timer Numbers and Resolutions

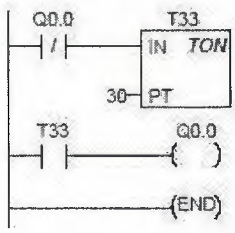
Timer	Resolution	Maximum Value	CPU 212	CPU 214	CPU 215/216
TON	1 ms	32.767 seconds (s)	T32	T32, T96	T32, T96
	10 ms	327.67 s	T33 to T36	T33 to T36, T97 to T100	T33 to T36, T97 to T100
	100 ms	3276.7 s	T37 to T63	T37 to T63, T101 to T127	T37 to T63, T101 to T255
TONR	1 ms	32.767 s	T0	T0, T64	T0, T64
	10 ms	327.67 s	T1 to T4	T1 to T4, T65 to T68	T1 to T4, T65 to T68
	100 ms	3276.7 s	T5 to T31	T5 to T31, T69 to T95	T5 to T31, T69 to T95

Required Time= Resolution * Preset Time (PT)

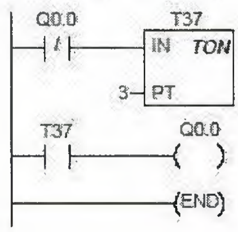
Using a 1-ms Timer



Using a 10-ms Timer



Using a 100-ms Timer



2.6.3 Counters

Counters are used in the same manner as electronic counters. Electronic counters count up or down when the clock pulse is sent each time. Counting mechanism is a function of central processing unit.

There are two types of counters. Normal counters and high speed counters. Normal counters are the up counter (CTU), the down counter (CTD) and the up/down counter (CTUD). The S7-200 counters have the following characteristics:

- The Up Counter (CTU) counts up from the current value of that counter each time the count-up input makes the transition from off to on. The counter is reset when the reset input turns on, or when the Reset instruction is executed. The counter stops upon reaching the maximum value (32,767).

- The Up/Down Counter (CTUD) counts up each time the count-up input makes the transition from off to on, and counts down each time the count-down input makes the transition from off to on. The counter is reset when the reset input turns on, or when the Reset instruction is executed. Upon reaching maximum value (32,767), the next rising edge at the count-up input causes the current count to wrap around to the minimum value (-32,768). Likewise on reaching the minimum value (-32,768), the next rising edge at the count-down input causes the current count to wrap around to the maximum value (32,767). When you reset a counter using the Reset instruction, both the counter bit and the counter current value are reset.

- The Up and Up/Down counters have a current value that maintains the current count. They also have a preset value (PV) that is compared to the current value whenever the counter instruction is executed. When the current value is greater than or equal to the preset value, the counter bit (C-bit) turns on. Otherwise, the C-bit turns off.

2.6.3.a Up Counter

The up counter counts up from a current value to a preset value (PV). Input of the counter (CU) checks each transition from off to on that comes from input contacts. Each transition from logic 0 to logic 1 increases one to the previous value and when the current value is equal to or greater than the preset value, the related contact of the counter is activated (logic 1).

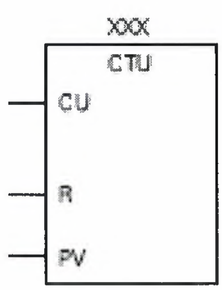


Figure 2.19 The Up Counter

Example 2)

In a packaging machine, retroreflective photoelectric sensor checks the number of bottles. The worker starts the conveyor by pressing the start button and stops it with stop button. Each package contains 6 bottles and when six bottles pass through the sensor to fall onto the adjacent belt for bottling, the belt pauses for 5 seconds and resumes carrying the belt contents. This process continues until the conveyor is stopped.

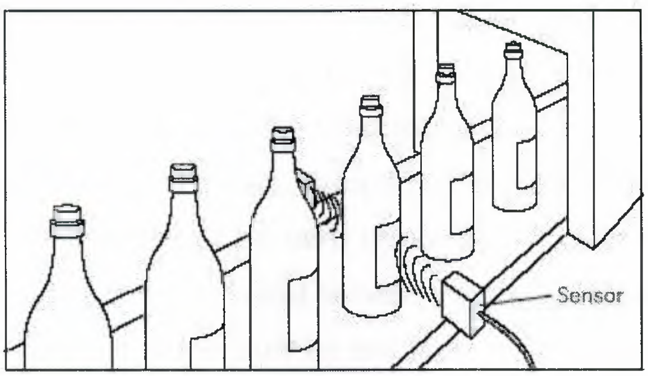
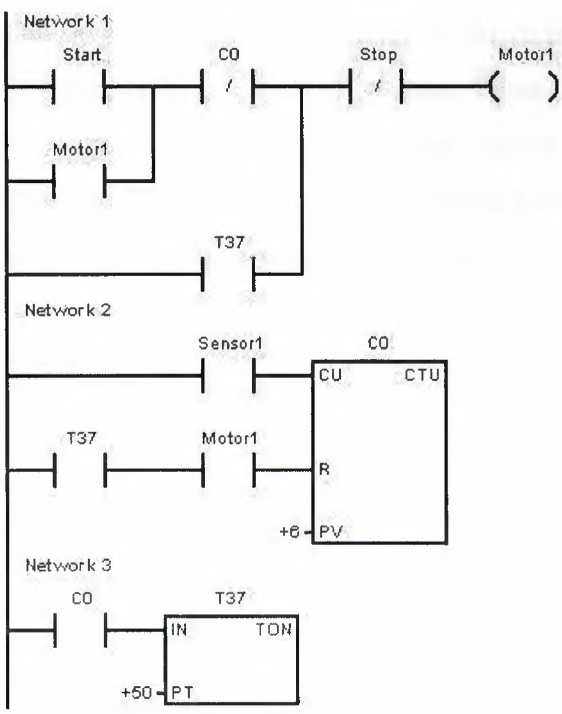


Figure 2.20 Bottling Machine Example

SOLUTION:



SYMBOL TABLE: ***View / Symbol Table*** in MicroWIN 3.2 Siemens PLC Software Prog.

Name	Address	Comment
Start	I0.0	Start Button is connected to I0.0
Stop	I0.1	Stop Button is connected to I0.1

Sensor1	I0.2	Photoelectric Sensor is connected to I0.2
Motor1	Q0.0	Motor is actuated by using a contactor or a relay
		whose energitizing A1-A2 contact is connected to Q0.0 and N terminals

When start button is pressed and released "Start" contact becomes on and it locks the output coil with its parallel contact(Motor1). When sensor senses the bottle "Sensor1" contact is on (logic 1) and adds one to the counter. When six bottles pass through the sensing area of sensor, the counter current value reaches to 6 and N/C contact of C0 in the first rung changes to off state. Thus, when 6 bottles passed, the motor stops for a while. In Network3, N/O contact of C0 starts the timer T37. After 5 seconds from the motor stop time, the T37 timer is activated because 5 seconds elapsed. The N/O contact of T37 timer in the first network is now on and it starts the motor again. When stop button is pressed the output contact parallel with "Start" contact becomes off and it breaks out the locking process.

2.6.3.b Up/Down Counter(CTUD)

The up/down counter counts up or down from the preset value each time either CD or CU transitions from off to on state. When the current value is equal to the preset value, the output QU turns on. When the current value (CV) is equal to zero, the output QD turns on. The counter loads the current value (CV) with the preset value (PV) when the load input (LD) is enabled. Similarly, the counter resets and loads the current value (CV) with zero when the reset (R) is enabled. The counter stops counting when it reaches preset or zero.

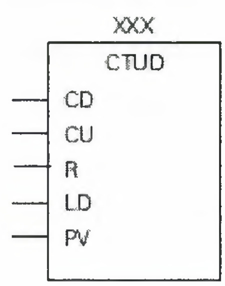


Figure 2.21 Up-Down Counter

2.6.4 Compare Instructions

Compare commands are used to compare two parameters either two bytes, integer numbers or real numbers. Compare instructions are very useful specially when sequential actions take place in a control system. For example, a control system performs the following function the motor starts for 30 seconds and stops. Then, mixer starts operation for 50 seconds and stops. After that, heater operates for 20 seconds and stops. It is not a good programming method to use a separate timer for each device because timers are limited and too many timers make the program bulky and complicated.

2.6.4.a Compare Byte

The Compare Byte instruction is used to compare two values: n1 to n2. A comparison of n1 = n2, n1 >= n2, or n1 <= n2 can be made. The contact is on when the comparison is true.

Operands: n1, n2: VB, IB, QB, MB, SMB, AC, Constant.

2.6.4.b Compare Word Integer

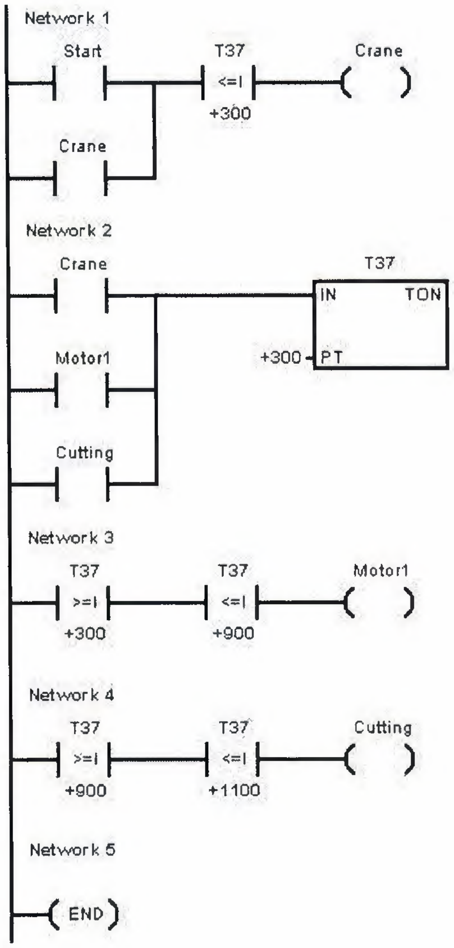
The Compare Word Integer instruction is used to compare two values: n1 to n2. A comparison of n1 = n2, n1 >= n2, or n1 <= n2 can be made. The contact is on when the comparison is true."

Operands: n1, n2: VW, T, C, IW, QW, MW, SMW, AC
AIW, Constant, *VD, *AC, SW

Example 3)

A control system performs the following process. When start button is pressed, the crane starts operation to carry the blocks of planks from floor onto the belt and stops. It takes 30 seconds for this operation. Then, Motor1 starts to carry the belt contents and it also stops 60 seconds later. Cutting tool starts to cut planks and it stops after 20 seconds and all machines will be off at the end of 110 seconds.

SOLUTION



The control system is started with start button. When start button is pressed, "Start" contact becomes on and "T37 <= I 300 " contact becomes on because the contact satisfies the condition. This comparison contact is on when the current value of the timer T37 is less than or equal to 300. Now, crane is on and the T37 timer starts timing up. After 30 seconds, crane stops due to "T37 <= I 300 " contact in Network1, which will be off depending on the contact condition(<= 30 seconds). The T37 timer continues timing up even its input contact "Crane" contact is off but "Motor1" contact does not let the timer to stop, being activated (on) before the crane is off.

In Network3, "T37 >= I 300 " and "T37 <= I 900 " contacts activate "Motor1" output coil together. If "T37 >= I 300 " contact were placed alone the motor would not stop and 60 seconds operation restriction would not be carried out.

In Network4, cutting tool operation takes place after 90 seconds from crane start time. We declare this by T37 >= I 900 ” contact and “T37 <= I 1100 ” contact which will stop cutting tool.

CHAPTER THREE
INDUSTRIAL APPLICATIONS
&
CONVEYOR CONTROL SYSTEMS

3.1 Introduction to Control Systems

Electrical control systems are designed using basic devices or equipments such as relays, contactors, thermistor relays, motor protection relays, motor starters, fuses, timers, counters, signal lamps, start/stop buttons, various sensors etc.

3.2 Control System Devices

In this section, control devices will be explained. There are so many control devices. So that basic devices or equipments will be introduced.

3.2.1 Electrically Controlled Switches

3.2.1.a Contactors

Contactors are useful in commercial and industrial applications, particularly for controlling large lighting loads and motors. One of their hallmarks is reliability. However, like any other device, they are not infallible. Usually, the reason for contactor failure is misapplication. When someone uses a lighting contactor in a motor application, which is a misapplication. The same is true when someone uses a "normal operation" motor contactor for motor jogging duty. Contactors have specific designs for specific purposes.

Contacts will overheat if they transmit too much current, if they do not close quickly and firmly, or if they open too frequently. Any of these situations will cause significant deterioration of the contact surface and the shape of that surface. Coils can overheat if operating voltages are too low or too high; if the contacts fail to open or close because of dirt or misalignment; or if they have suffered physical damage.

The following figure shows the interior of a basic contactor. There are two circuits involved in the operation of a contactor: the control circuit and the power circuit.

The control circuit is connected to the coil of an electromagnet, and the power circuit is connected to the stationary contacts.

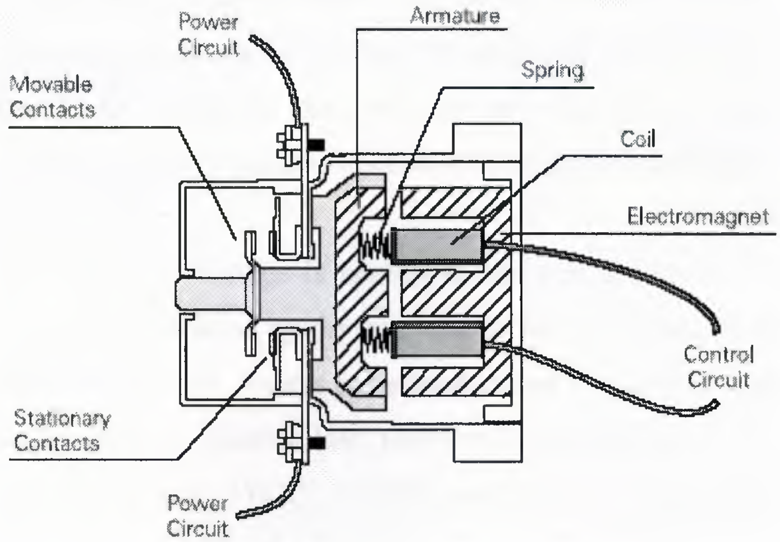


Figure 3.1 Contactor Internal Structure

When power is supplied to the coil from the control circuit, a magnetic field is produced magnetizing the electromagnet. The magnetic field attracts the armature to the magnet, which in turn closes the contacts. With the contacts closed, current flows through the power circuit from the line to the load. When the electromagnet's coil is de-energized, the magnetic field collapses and the movable contacts open under spring pressure. Current no longer flows through the power circuit.

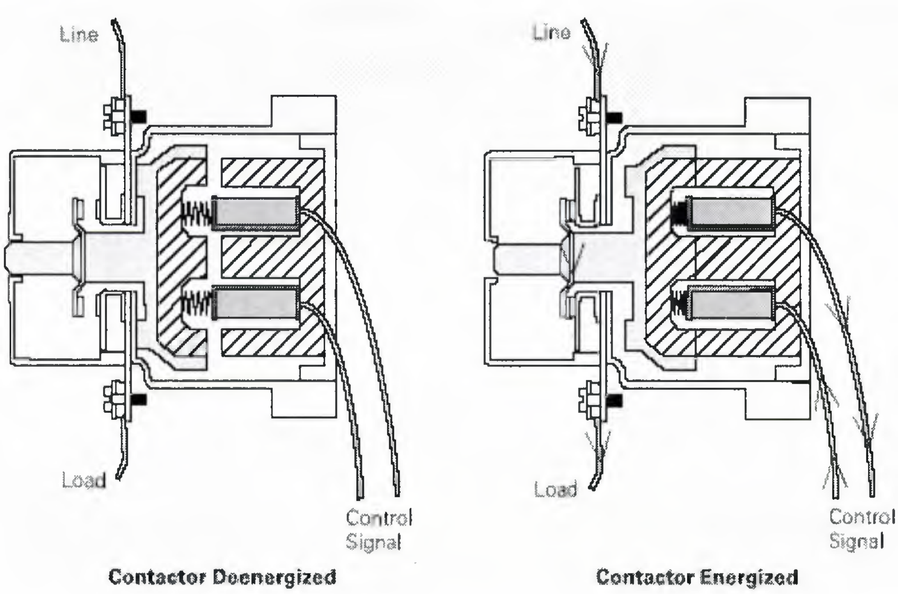


Figure 3.2 Contactor Operation Principle

3.2.1.b Relays

Relays are similar to contactors as operation principle. Relays operate utilizing electromagnetic principles. A simple electromagnet can be fashioned by winding a wire around a soft iron core. When a DC voltage is applied to the wire, the iron becomes magnetic. When the DC voltage is removed from the wire, the iron returns to its non-magnetic state. This principle is used to operate electromagnetic switches.

A relay operates when the voltage is applied to the coil terminals. When a relay is energized, the palette is attracted by the core due to electromagnetism and closes N/O contact and opens N/C contact. Relays are manufactured in different sizes and different operating voltages. Relays control low power electrical devices. Typical nominal voltage ratings (of coil) are 5VDC, 12VDC and 24VDC. Relays typically have a Normally Closed (NC) contact and a Normally Open (NO) contact. A relay consists of three parts:

- Electromagnetic parts (core and coil)
- Palette
- Contacts

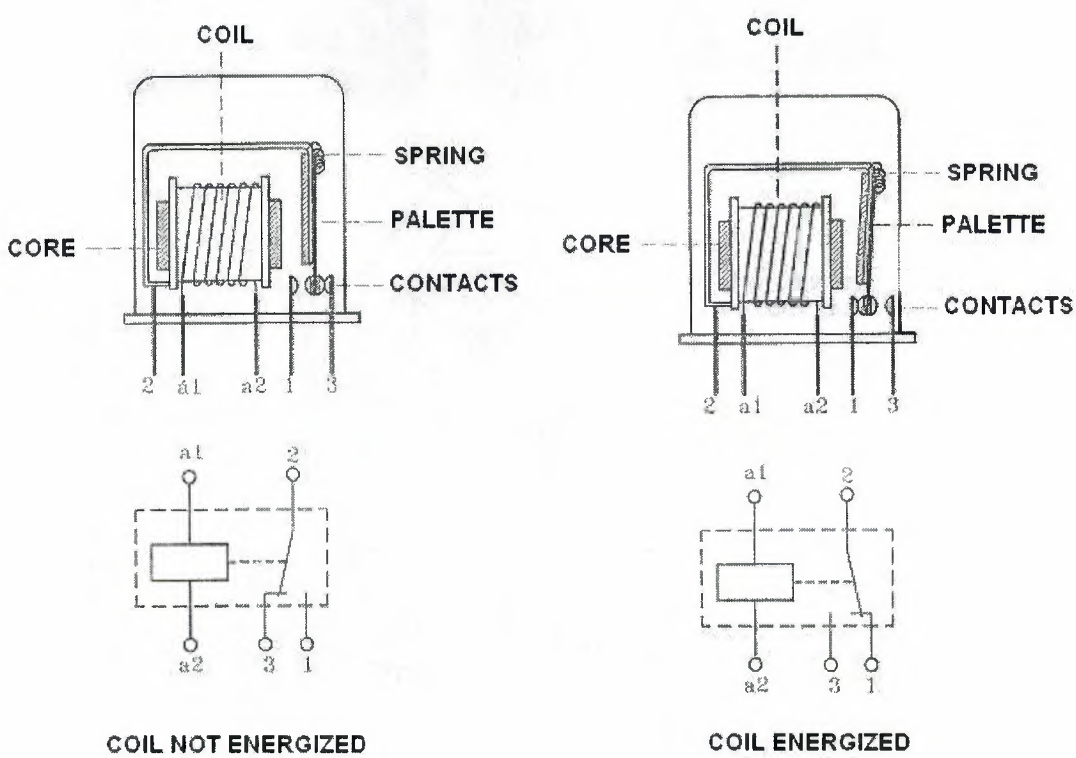


Figure 3.3 Relay Internal Structure and Operation Principle

3.2.2 Fuses

Fuses are protective devices which operate when the current exceeds specified limits at a saturated duration. Normally, current can be undulating in short time intervals. When the current exceeds the limits for enough time, fuse stops current flow, removing the power from the load.

Fuses are manufactured in various types but miniature circuit breakers (MCB) are so popular because of being long life and practical. NH fuses (knife fuses) are being replaced with compact switches which can control high currents such as 320A, 630A easily.

3.2.2.a Miniature Circuit Breakers(MCB)

Miniature circuit breakers (MCB) have a variety of application places such as houses, buildings, factories, etc. They are manufactured at rated current values of 1-2 -3 -5 -6 - 10 -16 -20 -25 -32 -40 -50 -63 A. They are manufactured as 1 pole, 3 poles (3ø line) and 4 poles (for 3ø line and a neutral line).

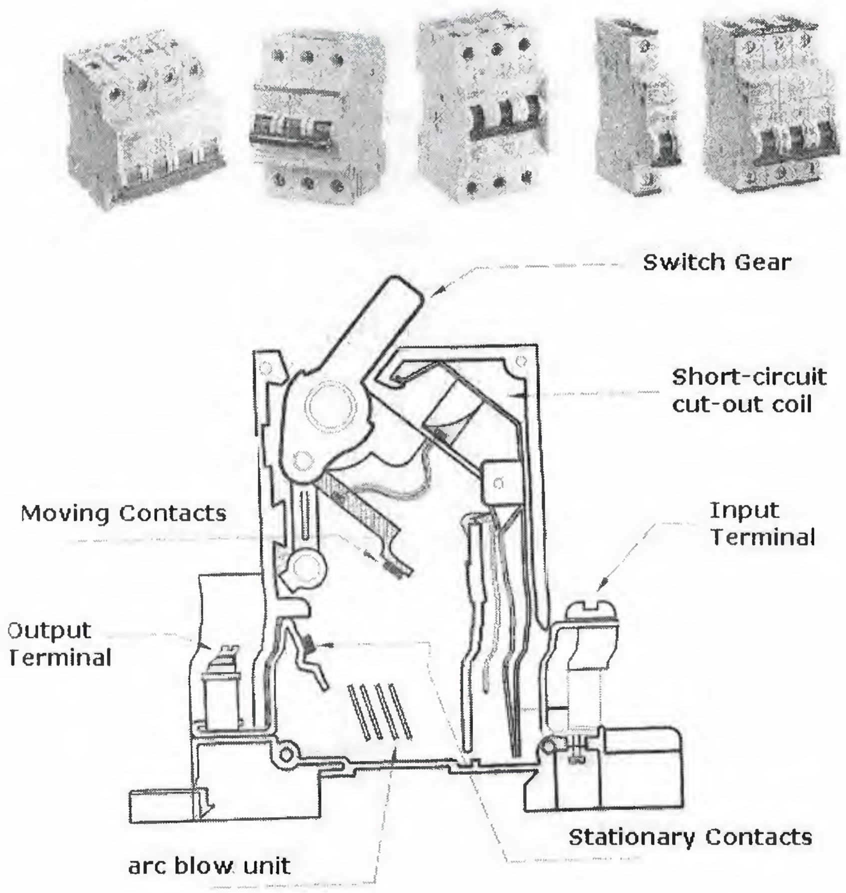


Figure 3.4 Various MCBs and MCB Internal Structure

3.2.3 Push Buttons

Push buttons are used to start/stop other control devices (contactor, relay, solenoid, valve, etc.) or send electrical signal to PLC or any other electrical control units. There are two types of pushbutton, the momentary and maintained. The momentary pushbutton switch is activated when the button is pressed, and deactivated when the button is released. The deactivation is done using an internal spring which separates two terminals electrically. The maintained pushbutton activates when pressed, but remains activated when it is released. Then to deactivate the button, it must be pressed again. There are three kinds of momentary push buttons:

1- Start button
2- Stop button
3- Two-way button (jog)

3.2.3.a Start Button

Start button is used to start the electrical control sytem or control circuit. It is a normally open (NO) contact. It lets the power flow when it is pressed and it does not allow when it is released.

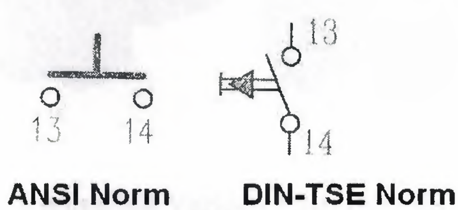


Figure 3.5 Start Button Symbol

3.2.3.b Stop Button

Stop button is used to stop the control circuit or any device (motor, timer, etc.). It is normally closed (NC) contact and internal spring holds the contact terminals closed. When stop button is pressed, it stops power flow until it is released.

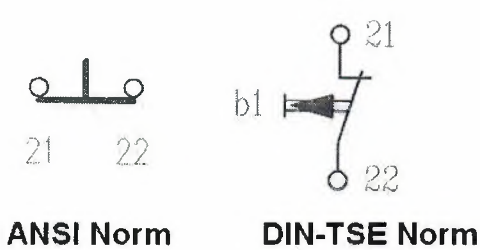


Figure 3.6 Stop Button Symbol

3.2.3.c Two –Way Button

Two-way buttons are used in the applications that when a device is started another device is needed to stop. It has a normally open (NO) contact and a normally closed (NC) contact. The two contacts operate inversely. When one of them is on, another off and vice versa.

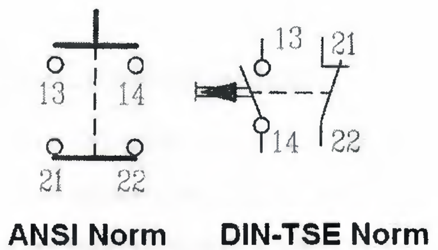


Figure 3.7 Two-Way Button Symbol

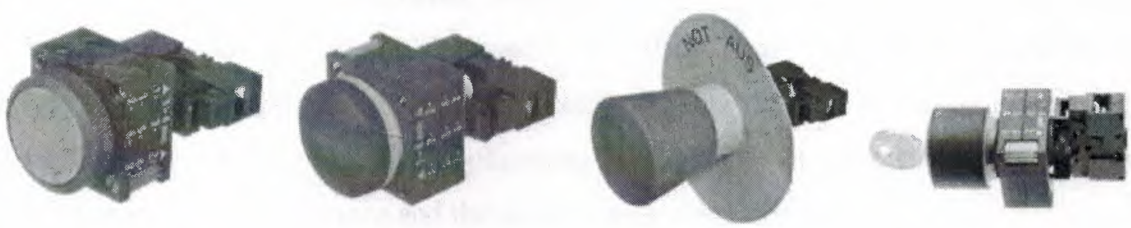


Figure 3.8 Various Push Buttons

3.2.4 Signal Lamps

Signal lamps are used to indicate whether a device is operating or not or failed. They can be used as warning equipment between the machine and the operator in the control panel. Signal lamps are manufactured in different colors; red, yellow, orange, green. The bulb is a neon lamp that can be Swan or Edison type with 12-380V operating range.

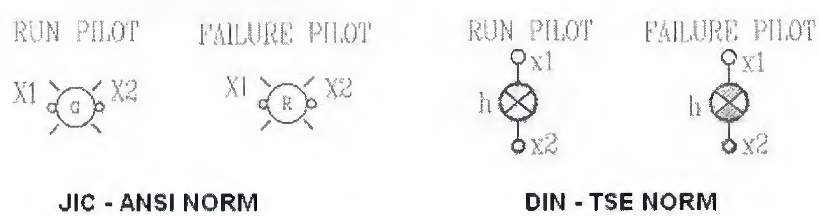


Figure 3.9 Signal Lamp Symbols

3.3 Sensors

A sensor is a device for detecting and signaling a changing condition. A changing condition can be simply the presence or absence of an object or material (discrete sensing). It can also be a measurable quantity like a change in distance, size or color (analog sensing). This information, or the sensor's output, is the basis for the monitoring and control of a manufacturing process.

3.3.1 Sensor Characteristics/Specifications

When specifying sensors, it is important to understand the common terms associated with the technology. While the exact terms differ from manufacturer to manufacturer, the concepts are globally understood within the industry.

3.3.1.a Sensing Distance

When applying a sensor to an application nominal sensing distance and effective sensing distance must be evaluated. Nominal sensing distance is the rated operating distance for which a sensor is designed. This rating is achieved using standardized criteria under average conditions. The effective sensing distance is the actual sensing distance achieved in an installed application. This distance is somewhere between the ideal nominal sensing distance and the worst case sensing distance.

3.3.1.b Hysteresis

Hysteresis or differential travel is the difference between operate (switch on) and release (switch off) points when the target is moving away from the sensor face. It is expressed as a percentage of the sensing distance. Without sufficient hysteresis a proximity sensor will continuously switch on and off, or "chatter," when there is excessive vibration applied to the target or sensor. It can also be made adjustable through added circuitry.

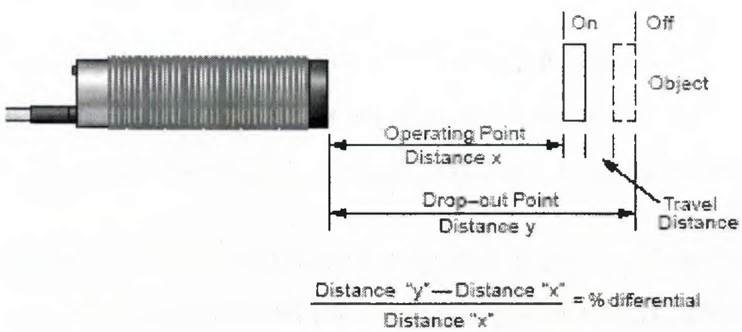


Figure 3.10 Hysteresis and Parameters

3.3.1.c Repeatability

Repeatability is the ability of a sensor to detect the same object at the same distance at all times. Expressed as a percentage of the nominal sensing distance, it is based on a constant ambient temperature and supply voltage.

3.3.1.d Switching Frequency

Switching Frequency is the number of switching operations per second achievable under standardized conditions. In more general terms, it is the relative speed of the sensor.

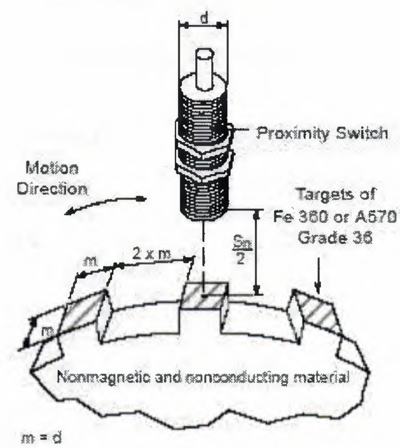


Figure 3.11 Standardized Switching Frequency Setup

3.3.1.e Response Time

The response time of a sensor is the amount of time that elapses between the detection of a target and the change of state of the output device (ON to OFF or OFF to ON). It is also the amount of time it takes for the output device to change state once the target is no longer detected by the sensor. The response time required for a particular application is a function of target size and the velocity at which it passes the sensor.

3.3.2 Sensor Power Ratings

Four voltages are typically available to power industrial sensors: 12VDC - 24VDC - 120VAC - 240VAC. Industrial sensors are typically designed to operate within one of four voltage ranges: 10-30VDC, 20-130VAC, 90-250VAC, and 20-250V AC/DC.

AC sensors and switches can receive power directly from the power line or a filtered source helping to eliminate the need for a separate power supply. Most DC sensors require a separate supply that isolates the DC portion of the signal from the AC line.

Typical current ratings for each sensor type:

- Photoelectric 35mA
- Ultrasonic 70mA
- Inductive 15mA
- Capacitive 15mA

3.3.3 Sensor Output Configuration

Output configurations fall into two categories, electromechanical and solid-state. Solid-state outputs should be considered for applications that require frequent switching or switching of low voltages at low currents. Only transistor output will be explained.

Solid-State or Electronic Outputs

- Transistor
- Field Effect Transistor (FET)
- Triac
- Analog
- Network or Bus

3.3.3.a Transistor Output

There are two kinds of transistor outputs: NPN and PNP types of outputs. For an NPN transistor output, the load must be connected between the sensor output and the positive (+) power connection. This is also known as a ***'sinking'*** output. A PNP transistor output is considered a ***'sourcing'*** output. The load must be connected between the sensor output and the negative (-) power connection.

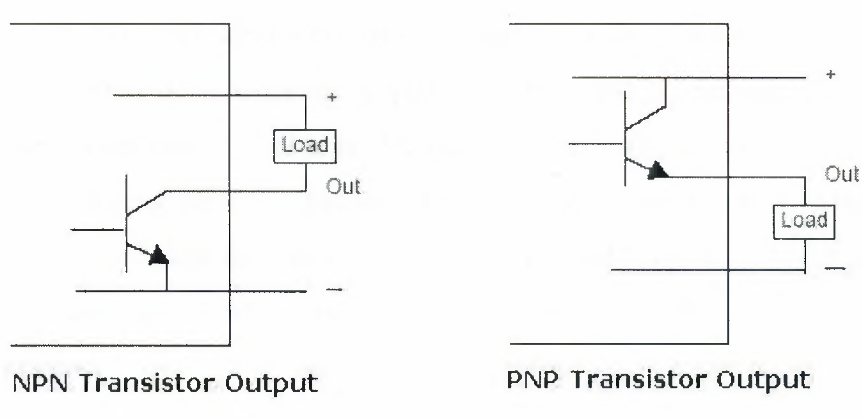


Figure 3.12 NPN and PNP Transistor Output Connections

3.3.4 Inductive Proximity Switches

Inductive proximity sensors are used to sense metal objects. Inductive proximity sensors are operated using an Eddy Current Killed Oscillator (ECKO) principle. This type of sensor consists of four elements: coil, oscillator, trigger circuit, and an output. The oscillator is an inductive capacitive tuned circuit that creates a radio frequency. The electromagnetic field produced by the oscillator is emitted from the coil away from the face of the sensor. The circuit has just enough feedback from the field to keep the oscillator going.

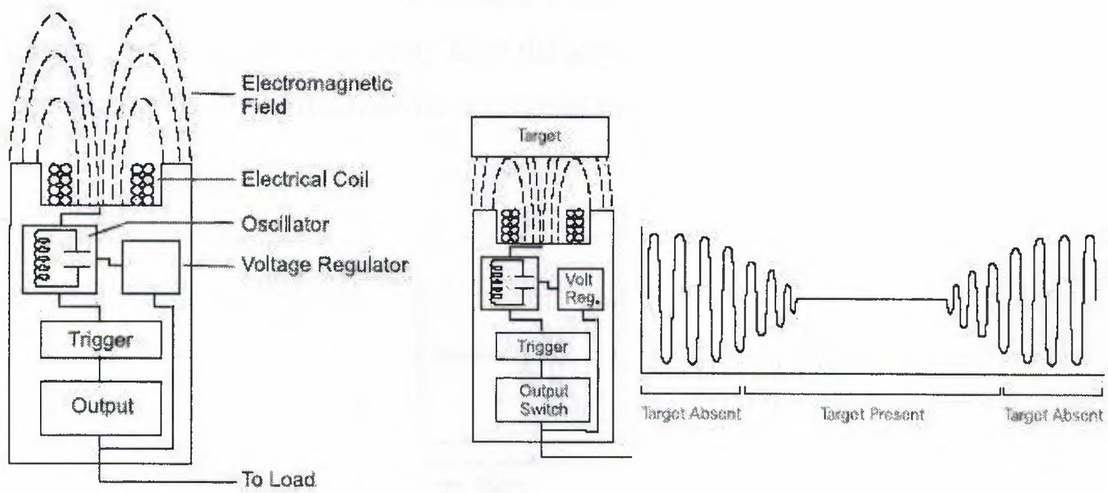


Figure 3.13 Inductive Proximity Switch and Operation Principle

When a metal target enters the field, eddy currents circulate within the target. This causes a load on the sensor, decreasing the amplitude of the electromagnetic field. As the target approaches the sensor the eddy currents increase, increasing the load on the oscillator and further decreasing the amplitude of the field. The trigger circuit monitors the oscillator's amplitude and at a predetermined level switches the output state of the sensor from its normal condition (on or off). As the target moves away from the sensor, the oscillator's amplitude increases. At a predetermined level the trigger switches the output state of the sensor back to its normal condition (on or off).

3.3.5 Capacitive Proximity Switches

Capacitive proximity sensors are similar to inductive proximity sensors. The main difference between the two types is that capacitive proximity sensors produce an electrostatic field instead of an electromagnetic field. Capacitive proximity switches will sense metal as well as nonmetallic materials such as paper, glass, liquids, and cloth.

The sensing surface of a capacitive sensor is formed by two concentrically shaped metal electrodes of an unwound capacitor. When an object nears the sensing surface it enters the electrostatic field of the electrodes and changes the capacitance in an oscillator circuit. As a result, the oscillator begins oscillating. The trigger circuit reads the oscillator's amplitude and when it reaches a specific level the output state of the sensor changes. As the target moves away from the sensor the oscillator's amplitude decreases, switching the sensor output back to its original state.

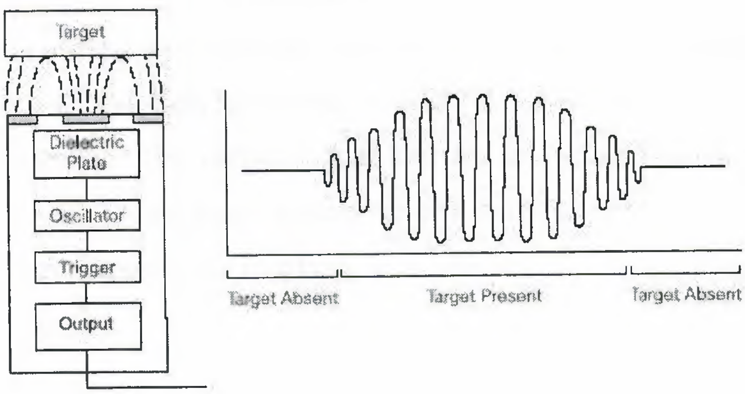


Figure 3.14 Capacitive Proximity Switch Operation Principle

Standard targets are specified for each capacitive sensor. The standard target is usually defined as metal and/or water. Capacitive sensors depend on the dielectric constant of the target. The larger the dielectric number of a material the easier it is to detect.

The advantages of capacitive proximity sensors include:

1. Detects metal and nonmetal, liquids and solids
2. Can "see through" certain materials (product boxes)
3. Solid-state, long life
4. Many mounting configurations

The disadvantages of capacitive proximity sensors include:

1. Short (1 inch or less) sensing distance varies widely according to material being sensed
2. Very sensitive to environmental factors—humidity in coastal/water climates can affect sensing output
3. Not at all selective for its target

3.3.6 Photoelectric Sensors

All photoelectric sensors operate by sensing a change in the amount of light received by a photodetector. The change in light allows the sensor to detect the presence or absence of the object, its size, shape, reflectivity, opacity, translucence, or color. Photoelectric sensors provide accurate detection of objects without physical contact.

A light source sends light toward the object. A light receiver, pointed toward the same object, detects the presence or absence of direct or reflected light originating from the source. Detection of the light generates an output signal for use by an actuator, controller, or computer. The output signal can be analog or digital. Some sensors modify the output with timing logic, scaling, or offset adjustments.

A photoelectric sensor consists of five basic components:

- Light source
- Lightdetector
- Lenses
- Logic circuit
- Output

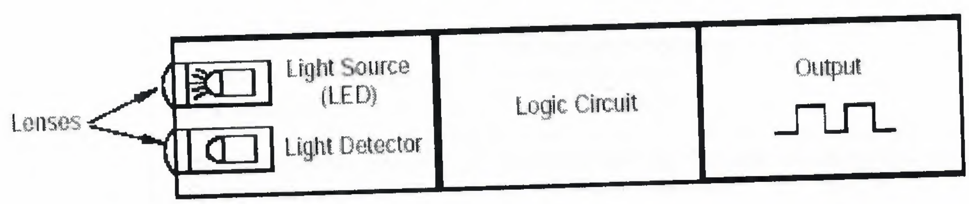


Figure 3.15 Photoelectric Sensor Components

Table 3.1 Photoelectric Sensors Comparison

Sensing Mode	Applications	Advantages	Cautions
Transmitted Beam	General purpose sensing Parts counting	• High margin for contaminated environments • Longest sensing distances • Not affected by second surface reflections • Probably most reliable when you have highly reflective objects	• More expensive because separate light source and receiver required, more costly wiring • Alignment important • Avoid detecting objects of clear material
Retroreflective	General purpose sensing	• Moderate sensing distances • Less expensive than transmitted beam because simpler wiring • Easy alignment	• Shorter sensing distance than transmitted beam • Less margin than transmitted beam • May detect reflections from shiny objects (use polarized instead)
Polarized Retroreflective	General purpose sensing of shiny objects	• Ignores first surface reflections • Uses visible red beam for ease of alignment	• Shorter sensing distance than standard retroreflective • May see second surface reflections
Standard Diffuse	Applications where both sides of the object cannot be accessed	• Access to both sides of the object not required • No reflector needed • Ease of alignment	• Can be difficult to apply if the background behind the object is sufficiently reflective and close to the object
Sharp Cutoff Diffuse	Short-range detection of objects with the need to ignore close distance backgrounds	• Access to both sides of the object not required • Provides protection against sensing of close backgrounds • Detects objects regardless of color within specified distance	• Only useful for very short distance sensing
Background Suppression Diffuse	General purpose sensing Areas where you need to ignore backgrounds that are close to the object	• Access to both sides of the target not required • Ignores backgrounds beyond rated sensing distance regardless of reflectivity • Detect objects regardless of color at specified distance	• More expensive than other types of diffuse sensors • Limited maximum sensing distance
Fixed Focus Diffuse	Detection of small objects Detects objects at a specific distance from sensor Detection of color marks	• Accurate detection of small objects in a specific location	• Very short distance sensing • Not suitable for general purpose sensing • Object must be accurately positioned
Wide Angle Diffuse	Detection of objects not accurately positioned Detection of very fine threads over a broad area	• Good at ignoring background reflections • Detecting objects that are not accurately positioned • No reflector needed	• Short distance sensing
Fiber Optics	Allows photoelectric sensing in areas where a sensor cannot be mounted because of size or environment considerations	• Glass fiber optic cables available for high ambient temperature applications • Shock and vibration resistant • Plastic fiber optic cables can be used in areas where continuous movement is required • Insert in limited space • Noise immunity • Corrosive areas placement	• More expensive than lensed sensors • Short-range sensing

Figure 3.16 Photoelectric Sensors

3.4 Brushed DC Motor

Brushed DC motors are widely used in applications ranging from toys to push-button adjustable car seats. Brushed DC (BDC) motors are inexpensive, easy to drive, and are readily available in all sizes and shapes.

The construction of a simple BDC motor is shown in Figure 3.17. All BDC motors are made of the same basic components: a stator, rotor, brushes and a commutator. The following paragraphs will explain each component in greater detail.

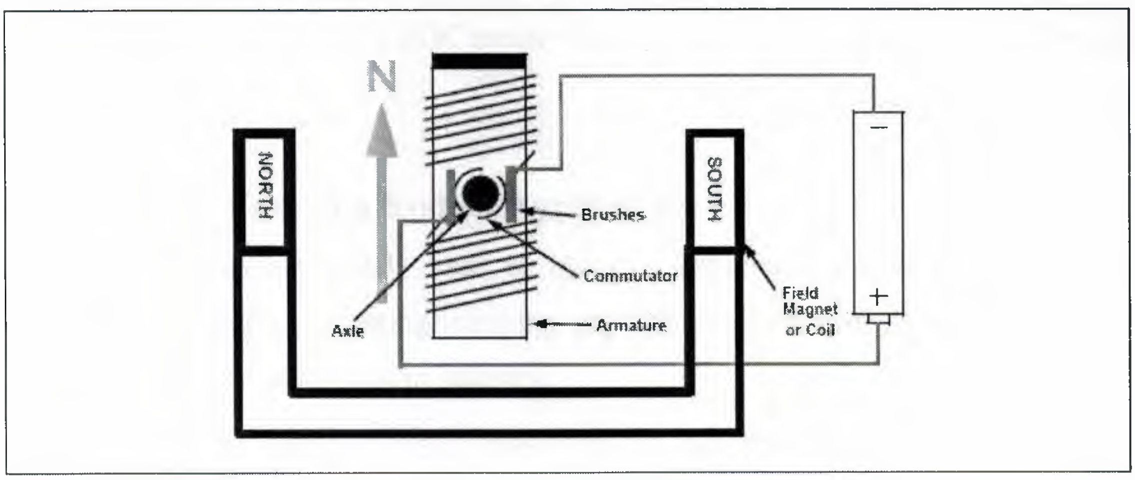


Figure 3.17 Simple Two-Pole Brushed DC Motor

3.4.1 Stator

The stator generates a stationary magnetic field that surrounds the rotor. This field is generated by either permanent magnets or electromagnetic windings. The different types of BDC motors are distinguished by the construction of the stator or the way the electromagnetic windings are connected to the power source.

3.4.2 Rotor

The rotor, also called the armature, is made up of one or more windings. When these windings are energized they produce a magnetic field. The magnetic poles of this rotor field will be attracted to the opposite poles generated by the stator, causing the rotor to turn. As the motor turns, the windings are constantly being energized in a different sequence so that the magnetic poles generated by the rotor do not overrun the poles generated in the stator. This switching of the field in the rotor windings is called ***"commutation"***.

3.4.3 Brushes and Commutator

Unlike other electric motor types (i.e., brushless DC, AC induction), BDC motors do not require a controller to switch current in the motor windings. Instead, the commutation of the windings of a BDC motor is done mechanically. A segmented copper sleeve, called a commutator, resides on the axle of a BDC motor. As the motor turns, carbon brushes slide over the commutator, coming in contact with different segments of the commutator. The segments are attached to different rotor windings; therefore, a dynamic magnetic field is generated inside the motor when a voltage is applied across the brushes of the motor. It is important to note that the brushes and commutator are the parts of a BDC motor that are most prone to wear because they are sliding past each other.

3.5 Conveyor Carrying System Application

Conveyor systems are widely used in industry, especially in factories as product processing, packaging, counting, carrying applications. Conveyor systems are vitally important for mass-production in industry.

A conveyor has three parts: Belt, rotating roller and rotating part (motor fastened roller). Belt is stretched and placed on the top of rotating roller and rotating part. Rotating roller is a tubular shape polyamide or plastic solid which is penetrated into a cylindrical metal body from both tips. Thus, roller slides with belt as motor turns over to move the belt.

3.5.a Conveyor Operation and Control

Starting/Stopping the Conveyor

Operator or user can control the conveyor by using the control panel. Control panel has the following equipments mounted on the surface: Start/Stop and Replace buttons; red, yellow and green indicator lamps. Red signal lamp is for warning, yellow and green lamps are for Motor1 and Motor2 operation indication respectively.

When "Start Button" is pressed, "Warning Lamp" starts blinking for ten seconds and it stops. It alerts that "Belt1" is going to start soon. After ten seconds, "Motor1" starts operation and "Belt1" starts moving.

When "Stop Button" is pressed, all motors and indicator lamps go off immediately.

Normal Operation

"Motor1" starts to rotate the "Belt1" and the product moves on the belt. When the product passes front of "Sensor1" sensing face, it is sensed by the sensor. After 1 second, "Motor1" stops and starts again a second later.

After the product is sensed by the "Sensor1", "Motor2" starts after 3 seconds. So that, the product will not be shaken on the second belt.

After the product is sensed by the "Sensor2", "Motor2" stops 1 second later. "Sensor2" also performs counting function and when 6 products fall into the package, the two belts stop immediately to avoid overloading the package. "Warning Lamp" is lit and it indicates that the package is full.

Operator changes the package and pressing the "Replace Button", conveyor returns back to beginning operation.

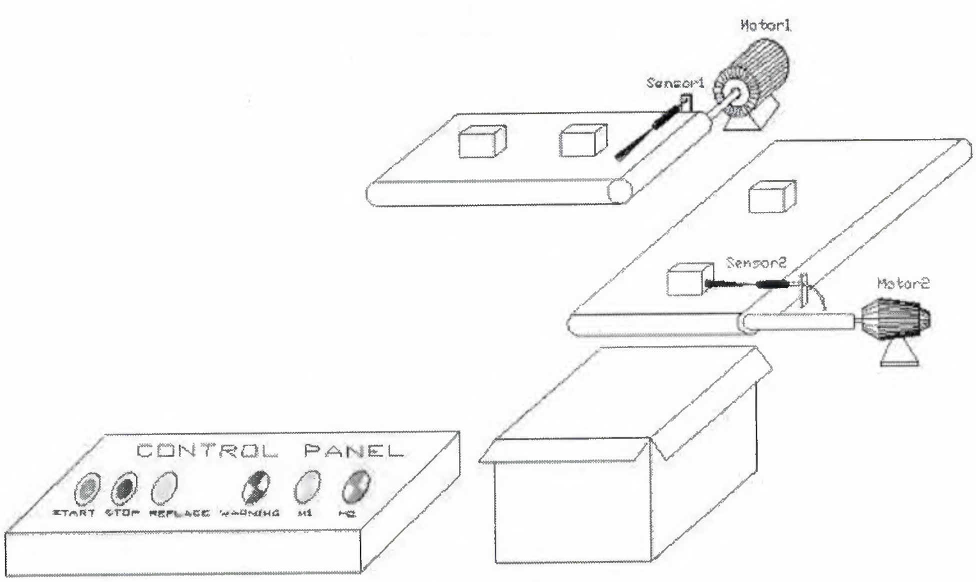
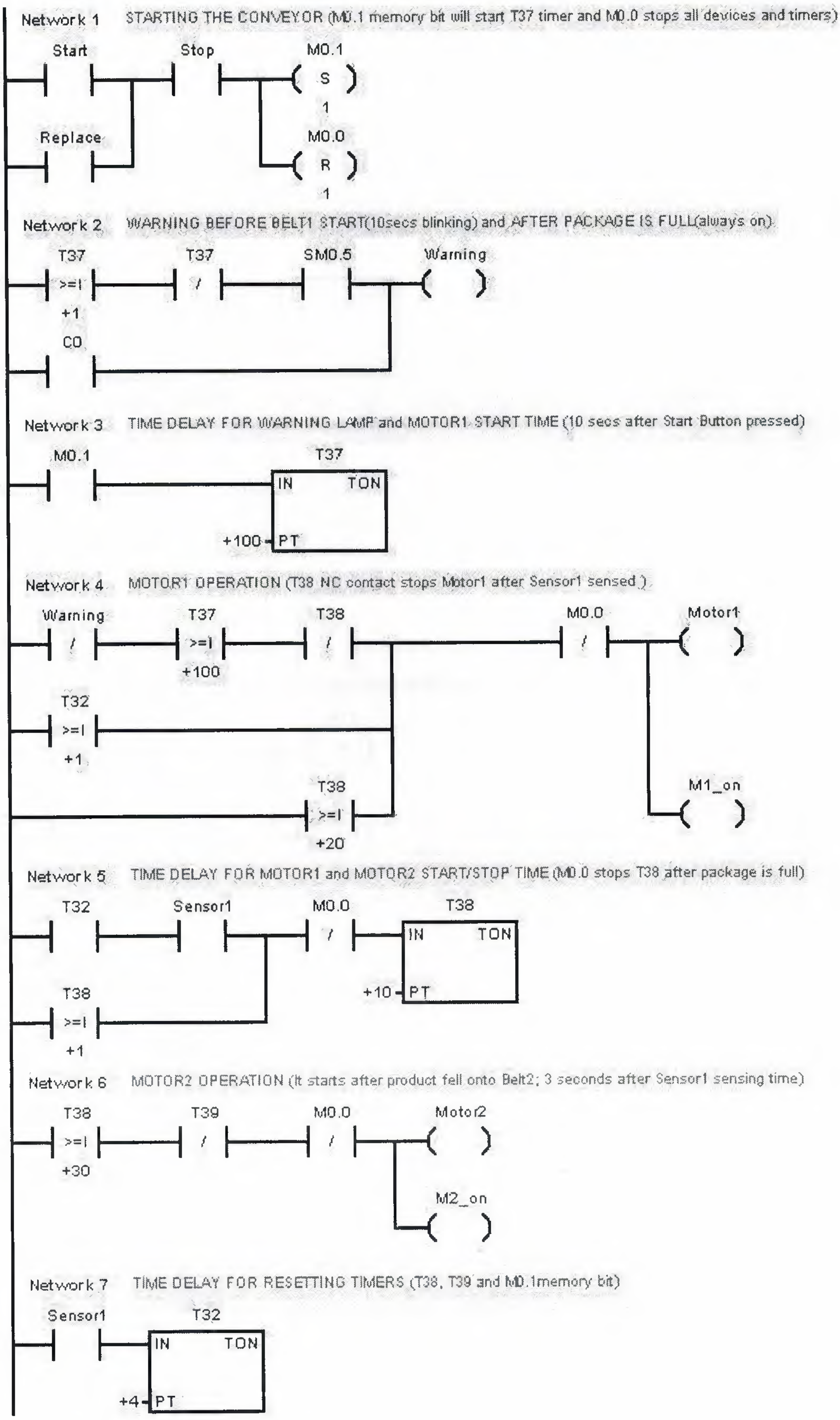
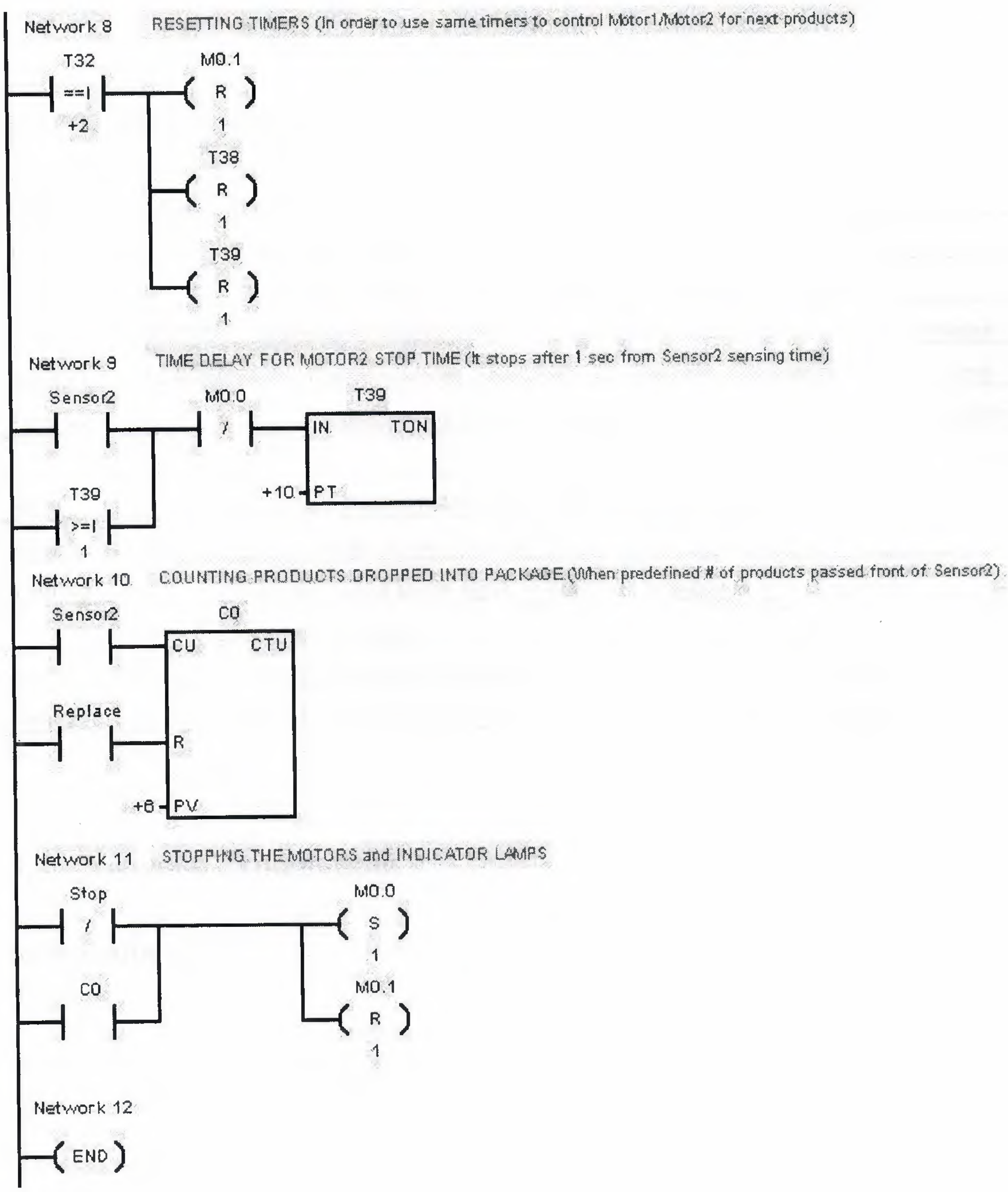


Figure 3.18 Conveyor Control System View

3.5.b Ladder Logic Program





NOTE2: *Programmers can appoint labels such as "Start", "Stop", "Motor1", etc. instead of input/output bits "I0.0", "Q0.0" in MicroWIN PLC Program, clicking on the pull-down menu "View / Symbol Table" and filling the "Symbol Table". Above given program is wrong unless "Symbol Table" is filled as given below. This helps the user to change the address parameters easily and focus on the program better.*

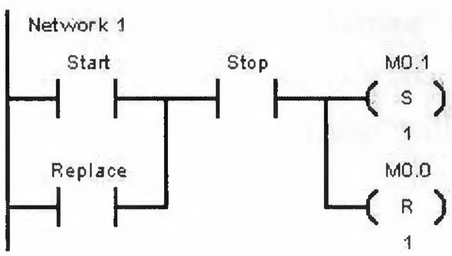
Table 3.2 Symbol Table for the PLC Program

SYMBOL TABLE*: View / Symbol Table* in MicroWIN 3.2 Siemens PLC Software Program

Name	**Address**	**Comment**
Start	I0.0	"Start " label responds to Start Button input to start the conveyor
Stop	I0.1	"Stop" label represents the Stop Button input to stop the conveyor
Replace	I0.2	"Replace" label checks the contact states of Replace Button to start the conveyor
Sensor1	I0.3	"Sensor1" determines position of object on the belts and arranges start/stop time
Sensor2	I0.4	"Sensor2" determines position of object and performs counting function of objects
M1_on	Q0.0	"M1_on " activates/de-activates Yellow Signal Lamp for Motor1 operation indication
M2_on	Q0.1	"M2_on" sends signals to Green Lamp whether Motor2 is on or off
Warning	Q0.2	"Warning" output sends signal to Warning Lamp before Motor1 operation
		and when package is full
Motor1	Q0.3	"Motor1" output sends 24VDC voltage to related relay to start Motor1
Motor2	Q0.4	"Motor2" output sends 24VDC voltage to related relay to start Motor2

3.5.c Network Explanations

NETWORK 1



Conveyor operation control starts with Network1 and when "Start Button" is pressed, "Start" labeled normally open (N/O) contact becomes on, "M0.1" memory bit is set and "M0.1" enables the T37 TON type timer. T37 timer starts timing up and it provides 10 seconds delay for "Warning Lamp" operation. After 10 seconds, "Warning Lamp" will be off and "Motor1" will start to rotate "Belt1". When predefined number of products or objects dropped into package, conveyor system will stop all the motors and indicator lamps. Conveyor will start again after "Replace Button" is pressed, so that "Replace" N/O contact is parallel with "Start" contact in this manner.

"M0.0" memory bit will be reset in order to unlock the "Motor1" and "Motor2" operation. Otherwise, they will be off even we pressed "Replace Button". Here, "Stop" N/O contact will not allow conveyor start if "Stop Button" is pressed at the same time.

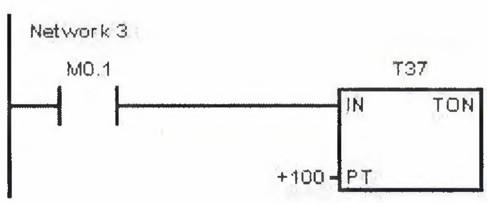
NETWORK 2

In Network 2, warning process is performed. After "Start Button" is pressed "Warning Lamp" blinks for 10 seconds and stops. This warning alerts the people that belt will start soon.

After "Start Button" is pressed, T37 timer starts timing up and "T37 >=I +1" contact is on and "Warning" output coil becomes on.

"SM0.5" status bit provides a clock pulse that is on for 0.5 seconds and then off for 0.5 seconds for a cycle time of 1 second. It makes "Warning Lamp" on and off continuously with 0.5 second time intervals. After 10 seconds; T37 reaches to 100, T37 N/C contact is off and "Warning" coil is off; forcing the "Warning Lamp" to off-state. When the package is full, C0 counter bit will be on and C0 counter's N/O contact will be on. Thus, "Warning Lamp" will be on if the package is full (C0 current value equals to PV).

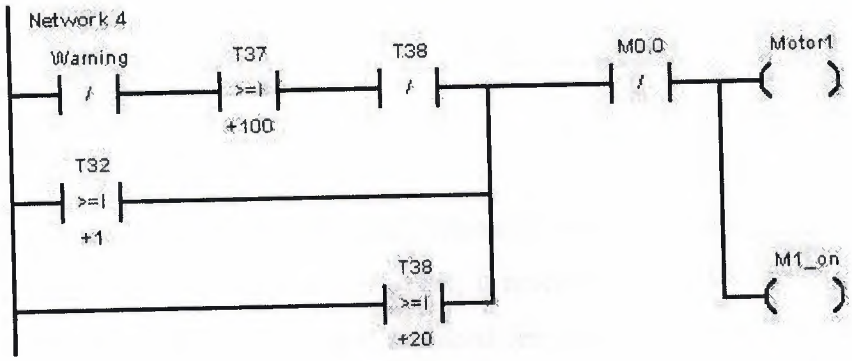
NETWORK 3



When "Start Button" is pressed "M0.1" will be set and T37 timer will start timing up. T37 is a 100 ms type timer and Preset Time (PT) value is selected as "100" for 10 seconds delay. After 10 seconds, T37 timer will be on.

Its N/C contact will be off to stop "Warning Lamp" (See Network2).

NETWORK 4

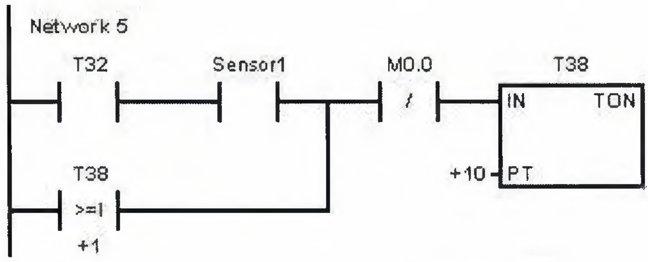


In Network 4, "Motor1" control is performed. After 10 seconds from "Start Button" pressing time and "Warning Lamp" is in off-state, "Motor1" will operate. This task is declared as N/C contact of "Warning" output series with "T37 >=I +100" contact. T38 N/C contact will be off after 1 second from "Sensor1" sensing time (See Network5). After "Motor1" stopped due to T38 N/C contact, it starts again by means of "T38 >=I +200" contact 1 second later.

T32 is a 1ms type timer which arranges reset time of T37, T38 and T39 timers. After first product passed from both sensor faces, T38 and T39 timers continues timing. These timers have to be reset to be used for second, third and other products. Otherwise, timers never equal to preset time or integer value in comparison contacts, i.e. T38 N/C contact, "T38 >=I +20" contact and start/stop timing of motors will not work properly. "T32 >=I +1" contact prevents "Motor1" and "Motor2" stop when the timers are reset due to "T32 ==I +2" contact in Network 8.

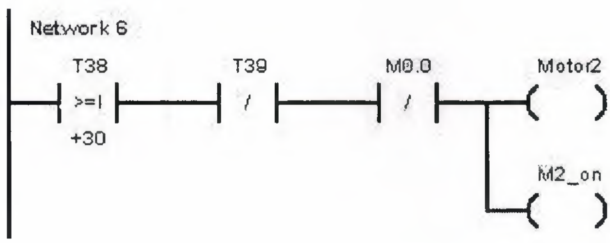
M0.0 memory bit will be on when 6 objects dropped into package and its N/C contact will not allow "Motor1" and "Motor2" to operate when the package is full. Also, T38 and T39 timers will not perform timing depending on "Sensor1" and "Sensor2" output states; maybe other objects are in front sensors.

NETWORK 5



In this network, "Motor1" and "Motor2" start/stop timing is provided using T38 timer. When "Sensor1" senses the object, it resets T38 and T39 timers first (T32 N/O contact and "Sensor1" N/O contact is placed for this aim). Then, T38 timer starts operation to stop "Motor1" after 1 second from "Sensor1" sensing time and restart again after 1 second. In the third second, "Motor2" starts to rotate "Belt2" due to "T38 >=I +30" contact in Network6. "T38 >=I +1" contact is necessary for locking process and it does not let timer to stop when the object passed front of sensor.

NETWORK 6

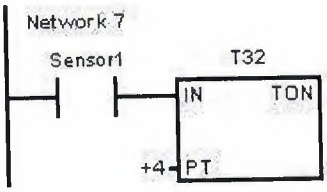


In Network 6, "Motor2" control is performed similar to "Motor1" operation principle. When the object passes front of "Sensor1", T38 timer starts timing up and "Motor2" starts to rotate "Belt2" 3 seconds later due to "T38 >=I +30" contact. "Motor2" stops after the object passed front of "Sensor2" a second later (with T39 N/C contact). It starts again if an object is sensed by "Sensor1".

"M2_on" output coil represents the output to indicate that "Motor2" is on. This output supplies a 24VDC source to turn on green signal lamp when "Motor2" is operating.

When 6 products dropped into package, M0.0 memory bit is on and its N/C contact does not let power flow to the related relay's coil terminals (A1-A2) to stop "Motor2".

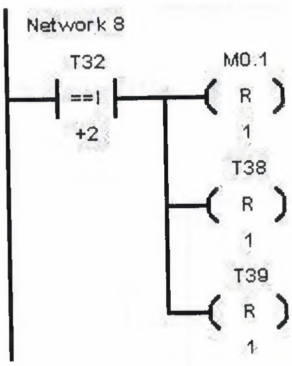
NETWORK 7



In this network, reset timing is provided using a 1ms timer. As it is mentioned in Network4, T32 is a TON type timer and it arranges reset time of T38, T39 timers to be used for the control of motors for the following products or objects.

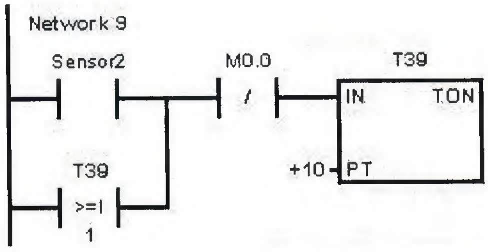
The main reason for selecting a TON type timer is that when product passed front of "Sensor1", the timer must be reset for next object process. Otherwise, extra programming techniques would be used unnecessarily.

NETWORK 8



In Network 8, T38, T39 and M0.1 bits are reset. T38 and T39 timers are reset for the use of same timers to control both motors for next coming objects or products.
As you see in Network 3, T37 timer is enabled with M0.1 memory bit and it is already set in the first network. If we reset T37 timer instead of M0.1 bit, T37 will restart timing again just after reset occurred. So that, we needed to reset M0.1 memory bit to stop T37 timer.

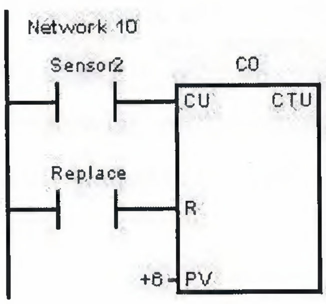
NETWORK 9



In Network 9, "Motor2" stop timing is provided. T39 timer is used in this manner. When "Sensor2" sensed the object, T39 timer starts timing up and when preset value is reached (after 10 seconds from object sensing time); N/C contact of T39 in Network 6 becomes off. Thus, Motor2 stops due to the activation of timer.

Sensors naturally provide momentary output, timers need locking process.
"T39 >=I +1" contact is used for this reason. When the package is full, T39 timer is disabled to ensure correct operation of conveyor in the beginning operation.

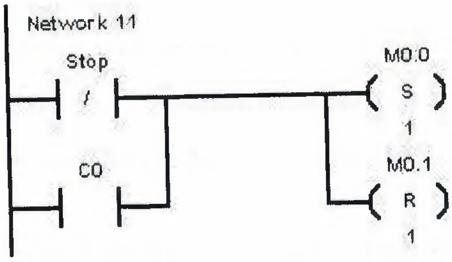
NETWORK 10



In this network, "C0" up counter counts the number of objects dropping into the package. When "Sensor2" is on for each time, counter's current value increases one and when the number of objects in the package is equal to 6 (depending on preset value), "C0" counter is active. "C0" normally closed contact hinders the power flow to the motors (See Network4, 6), "Motor1" and "Motor2" is stopped.

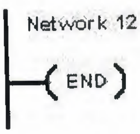
When "Replace Button" is pressed "C0" counter is reset and it start counting from the beginning value.

NETWORK 11



In Network 11, conveyor stop operation performed. When stop button is pressed, M0.0 is set to stop all motors and indicator lamps due to N/C contact of M0.0 bit. Also, it resets M0.1 to stop T37 timer. These functions are also valid when package is full (when C0 is on).

NETWORK 12



This network terminates the main program using the unconditional end instruction. It is necessary to use end instruction to declare to the assembler that the program finished. Otherwise, programming software will not compile the program and it will not be downloaded to the PLC.

CONCLUSION

This project aims to present a practical automation system using the PLC as a control unit. PLC is used for controlling and automating the industrial machines, industrial process in production stages, etc.

In this project, automation task was the control of motors to move the conveyor belts and photoelectric sensors arrange the start/stop time of motors and performing the counting function of the products in a packaging system.

First of all, equipments of control system were determined and their behaviors to the system were taken into account. Correct types of equipments were selected according to this evaluation.

PLC program was written considering all possible cases and clashes in great detail. Wrong programming structures were abstained. Easy, simple programming techniques were used to avoid clashes, unexpected behaviors. Symbolic addressing was used in programming to concentrate on the program deeply and sharply. Compare instructions were used to minimize the number of timers, counters as much as possible. Thus, the number of networks was decreased surprisingly and the program memory was used efficiently.

Consequently, a flexible PLC based control system was designed and the advantages, facilities and the convenience of Programmable Logic Controllers were observed and revealed. In any industrial or real life application process, PLCs take great mission to control all the system.

As a conclusion, PLCs are cheap according to their performance and functionality and offer flexible automation solutions with less wiring, easy modification in the system and reliability because of being microprocessor based device.

REFERENCES

[1] Özgür Cemal Özerdem, ''Programmable Logic Controller and Programming'',

Near East University Press, 2002.

[2] Information about Micro PLCs

'' *http://www. automationdirect.com* ''

[3] Information about Large PLCs

''*http://www.ab.com/en/epub/catalogs/*''

[4] Information about IDEC PLCs

''*http://www.idec.com/usen/products/Catalogs/PLCs/PLCsCategory.html*''

[5] Information about Relays and Control Devices

Lütfü Hayta, '' Elektrik Kumanda Devreleri ve Deneyleri ''

[6] Information about Siemens S-7 200 Micro PLC.

''*http://www.sea.siemens.com*''

SIMATIC S7-200 PLC System Manual published in 2001

[7] Central Processing Unit (CPU) Structure of PLC

Tim Wilmshurst, ''the Design of Small-Scale Embedded Systems''

[8] Information about Brushed DC Motors

Brushed DC Motor Fundamentals Application Note (AN905) Microchip Inc.

[9] Sensors and Applications

Fundamentals of Sensing Training Manual Rockwell Automation/Allen Bradley

Photoelectrics
Diffuse-reflective
Type PD32CND50

CARLO GAVAZZI

- Miniature sensor range
- Range: 500 mm
- Sensitivity adjustment by Teach-In programming
- Modulated, red light 660 nm
- Supply voltage: 10 to 30 VDC
- Output: 100 mA, NPN or PNP preset
- Make and break switching function programmable
- LED for output indication, signal stability and power ON
- Protection: reverse polarity, short circuit and transients
- Cable and plug versions
- Compact housing
- Excellent EMC performance

Product Description

The PD32CND50 sensor family comes in a compact 12 x 32 x 20 mm reinforced PMMA/ABS-housing.
The sensors are useful in applications where high-accuracy detection as well as small size is required.

The Teach-In function for adjustment of the sensitivity makes the sensors highly flexible. The output type is preset (NPN or PNP), and the output switching function is programmable (NO or NC).

Ordering Key **PD32CND50PPM5T**

- Type
- Housing style
- Housing size
- Housing material
- Housing length
- Detection principle
- Sensing distance
- Output type
- Output configuration
- Connection type
- Teach-In

Type Selection

Housing W x H x D	Range S_n	Ordering no. NPN & PNP cable Make & break switching	Ordering no. NPN & PNP plug Make & break switching
12 x 32 x 20 mm	500 mm	**PD 32 CND 50 NPT** **PD 32 CND 50 PPT**	**PD 32 CND 50 NPM5T** **PD 32 CND 50 PPM5T**

Specifications

Rated operating distance (S_n)	Up to 500 mm, reference target Kodak test card R 27, white, 90% reflectivity, 100 x 100 mm
Blind zone	None
Sensitivity	Adjustable by Teach-In (push button or wire)
Temperature drift	≤ 1%/°C
Hysteresis (H) (differential travel)	≤ 10%
Rated operational volt. (U_B)	10 to 30 VDC (ripple included)
Ripple (U_{rpp})	≤ 10%
Output current Continuous (I_e) Short-time (I)	 ≤ 100 mA ≤ 100 mA (max. load capacity 100 nF)
No load supply current (I_o)	≤ 25 mA @ 24 VDC
Minimum operational current (I_m)	0.5 mA
OFF-state current (I_r)	≤ 100 µA
Voltage drop (U_d)	≤ 2.4 VDC @ 100 mA
Protection	Short-circuit, reverse polarity and transients

Light source **Light type** **Sensing angle** **Ambient light** **Light spot**	GaAlAs, LED, 660 nm red, modulated ± 2° 5,000 lux 12 x 12 mm @ 160 mm
Operating frequency	1000 Hz
Response time OFF-ON (t_{ON}) ON-OFF (t_{OFF})	 ≤ 0.5 ms ≤ 0.5 ms
Power ON delay (t_v)	≤ 300 ms
Output function NPN and PNP NO/NC switching function	 Preset Set up by button
Indication Output ON Signal stability ON and power ON	 LED, yellow LED, green
Environment Installation category Pollution degree Degree of protection	 II (IEC 60664/60664A; 60947-1) 3 (IEC 60664/60664A; 60947-1) IP 67 (IEC 60529; 60947-1)

CARLO GAVAZZI

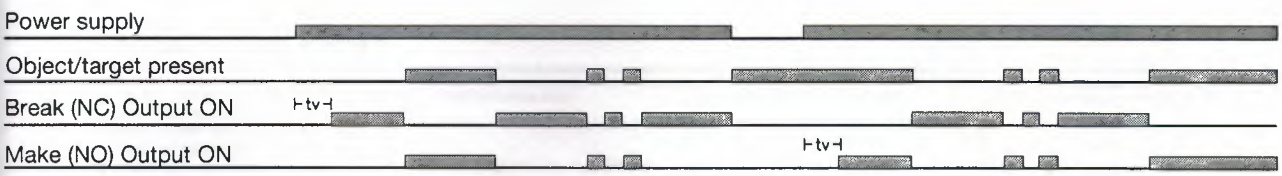
Specifications (cont.)

Ambient temperature	
Operating	-20° to +60°C (-4° to +140°F)
Storage	-20° to +80°C (-4° to +176°F)
Vibration	10 to 55 Hz, 0.5 mm/7.5 g (IEC 60068-2-6)
Shock	30 g / 11 ms, 3 pos, 3 neg per axis (IEC 60068-2-6, 60068-2-32)
Rated insulation voltage	500 VAC (rms)
Housing material	
Body	ABS, black
Front material	PMMA, red

Connection	
Cable	PUR, black, 2 m 4 x 0.14 mm², Ø = 3.6 mm
Plug	M8, 4-pin
Weight	With cable: 40 g With plug: 10 g
CE-marking	Yes
Approval	cUL

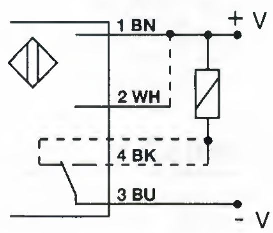
Operation Diagram

tv = Power ON delay

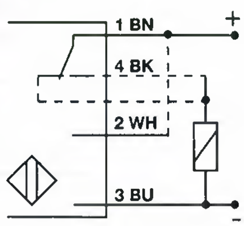


Wiring Diagrams

NPN



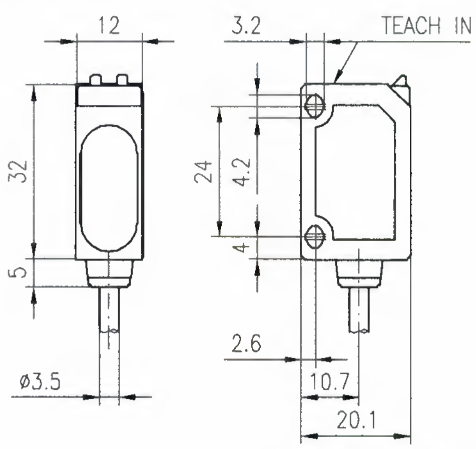
PNP



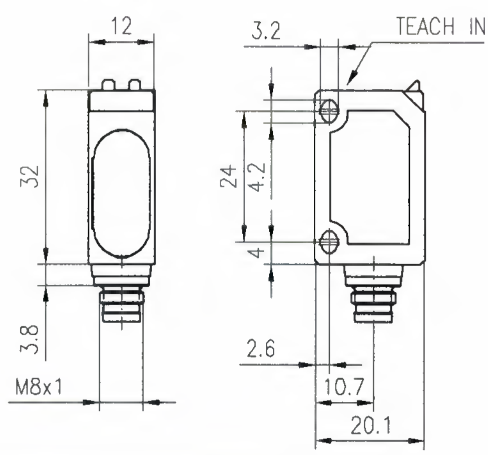
Teach input is active when 2 WH wire is connected to V+ (1 BN).

Dimensions

Cable version

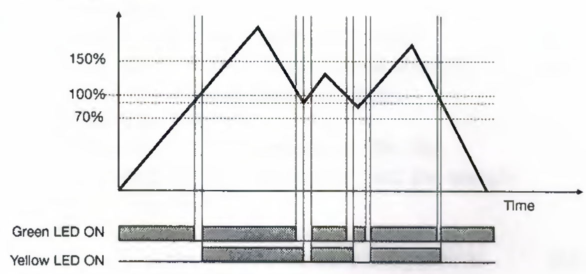


Plug version



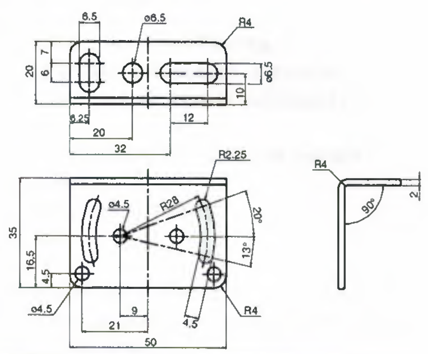
CARLO GAVAZZI

Signal Stability Indication



Accessories

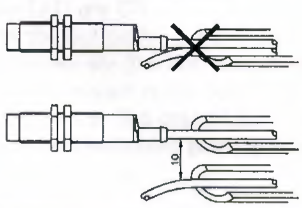
Mounting bracket APD32-MB1



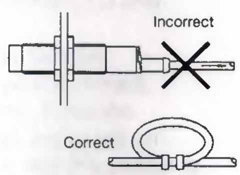
For further information refer to "Accessories"

Installation Hints

To avoid interference from inductive voltage/current peaks, separate the prox. switch power cables from any other power cables, e.g. motor, contactor or solenoid cables

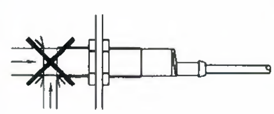


Relief of cable strain



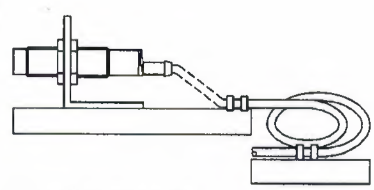
The cable should not be pulled

Protection of the sensing face



A proximity switch should not serve as mechanical stop

Switch mounted on mobile carrier



Any repetitive flexing of the cable should be avoided

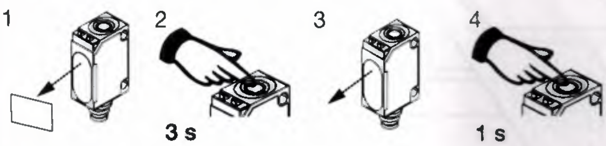
Delivery Contents

- Photoelectric switch: PD 32 CND 50 ...
- Installation instruction
- **Packaging:** Cardboard box

Adjustment

Sensitivity adjustment, with static object

1. Line up the sensor with the object. Yellow LED and green LED are ON.
2. Press the button for 3 s until both LED's flash simultaneously (the first switching point is stored).
3. Place the object outside the detection area.
4. Press the button for 1 s.
   a) The green LED flashes and stays ON: the second switching point is stored, and the sensor is ready to operate.
   b) Both LED's flash simultaneously: the sensor cannot detect the object, no switching points are stored.



Sensitivity adjustment, with only one object

1. Line up the sensor with the object. Yellow LED and green LED are ON.
2. Press the button for 3 s until both LED's flash simultaneously (the first switching point is stored).
3. Leave the object in the detection area, press the button for 1 s. The green LED flashes and stays on: the second switching point is stored, and the sensor is ready to operate.

Sensitivity adjustment, with a running process

1. Line up the sensor with the object. Green LED is ON. At this stage the status of the yellow LED can be ignored.
2. The running process must be the only "object" within the detection area. Press the button for 3 s until both LED's flash simultaneously.
   3 s
3. Press the button for at least the duration of one process cycle.
   1 cycle
   a) The green LED flashes and stays ON: both switching points have been stored, and the sensor is ready to operate.
   b) Both LED's flash simultaneously: the sensor cannot detect the object, no switching points are stored.

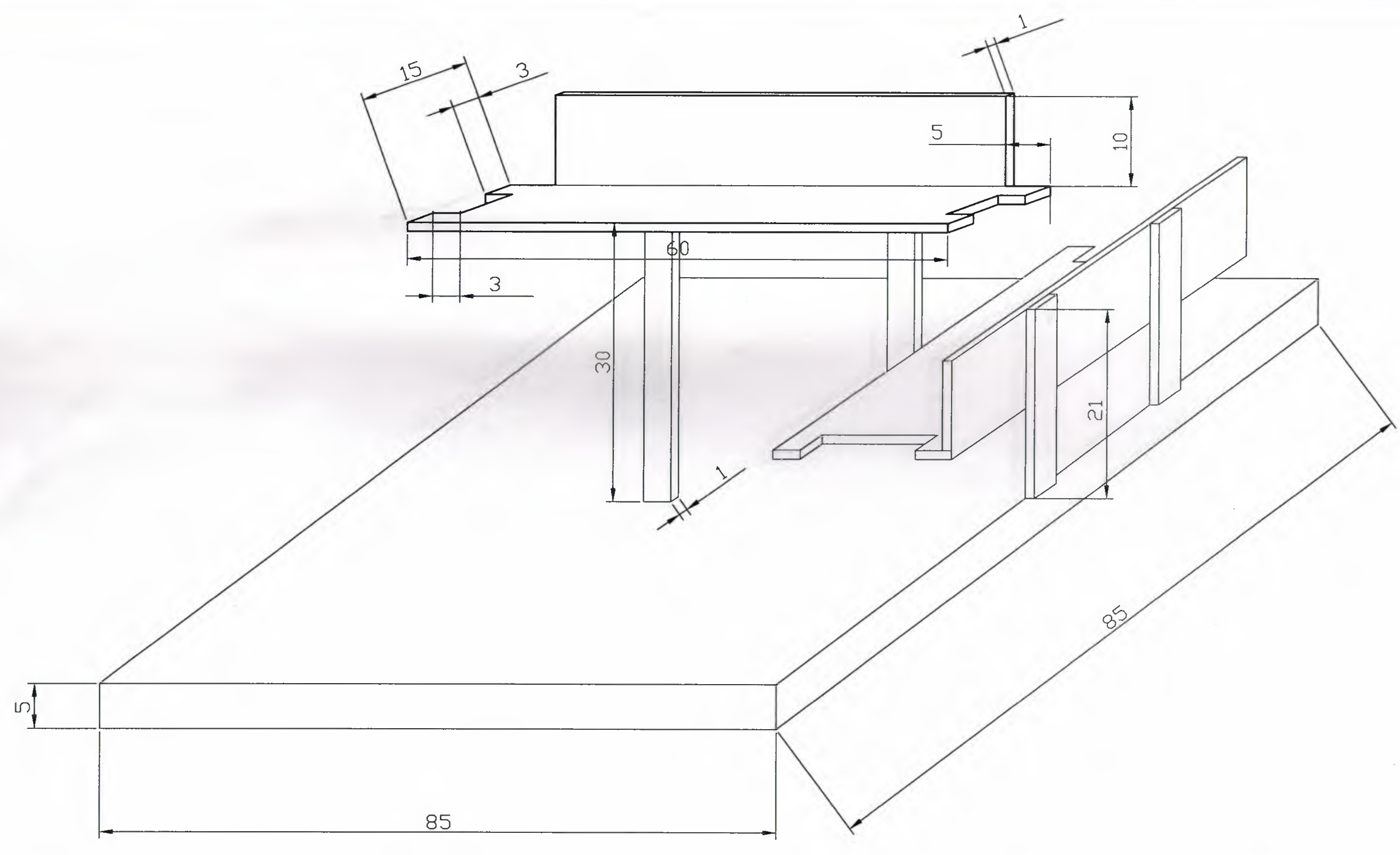
Programming of make and break switching function

1. Press the button for 13 s. **13 s**
   Both LED's flash alternately.
2. Release the button: the green LED flashes.
3. While the green LED flashes, the output is inverted each time the button is pressed. This is indicated by the yellow LED.
   When the button is not pressed for 10 s, the current output function is stored.
   The sensor is now ready for operation.

Default setting

1. No object in the detection area: Press the button for 3 s, until both LED's flash simultaneously. **3 s**
2. No object in the detection area:
   Press the button for 1 s. **1 s**
   The sensor is set to maximum sensitivity.

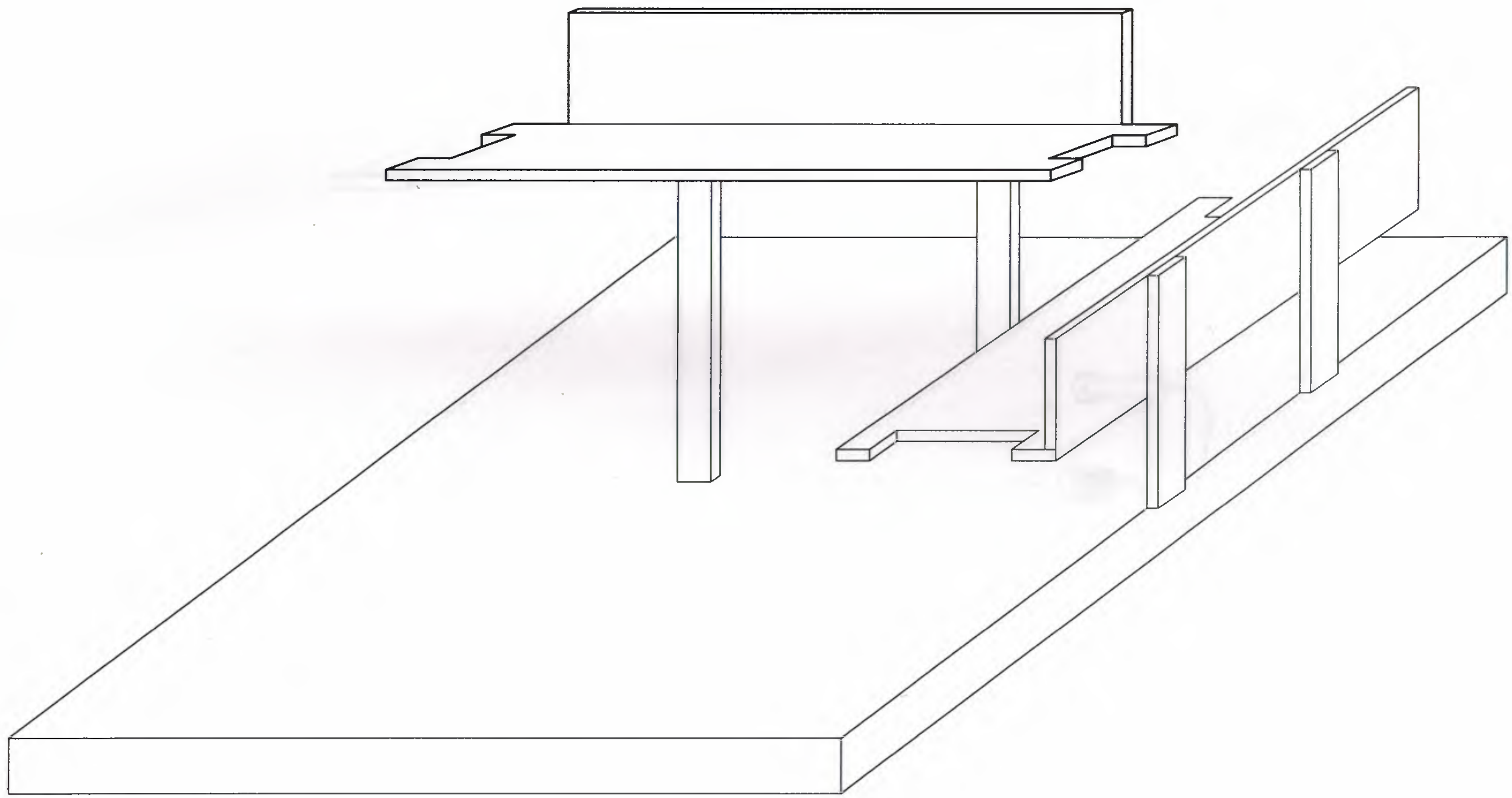
NB! The Teach Input (2 WH) will work similarly to the push button, active High.



CONVEYOR CARRYING SYSTEM DIMENSIONS (Dimensions in cm)

NCP12

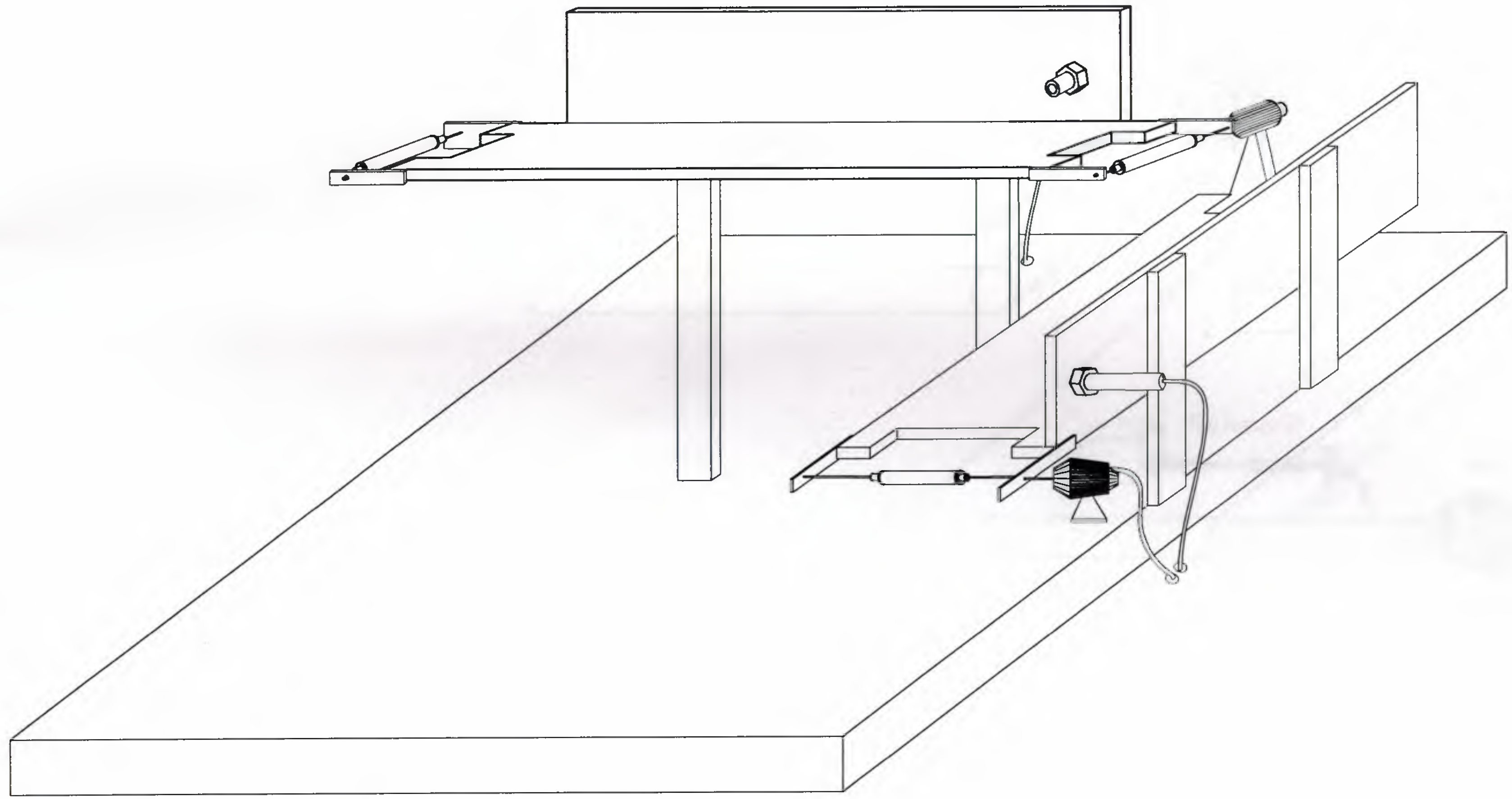
Name-Surname : NECIP FAZIL BATTAL	PROJECT: CONVEYOR CARRYING SYSTEM CONTROL BY PLC	PAGE NO:	1
Student No : 20011431 NEAR EAST UNIV.		DRAWING CODE:	01
Supervisor : ÖZGÜR CEMAL ÖZERDEM	DEFINITION: CONVEYOR CARRYING MECHANISM DIMENSIONS	DATE:	02 MAY 2006



CONVEYOR CARRYING SYSTEM

NCP12

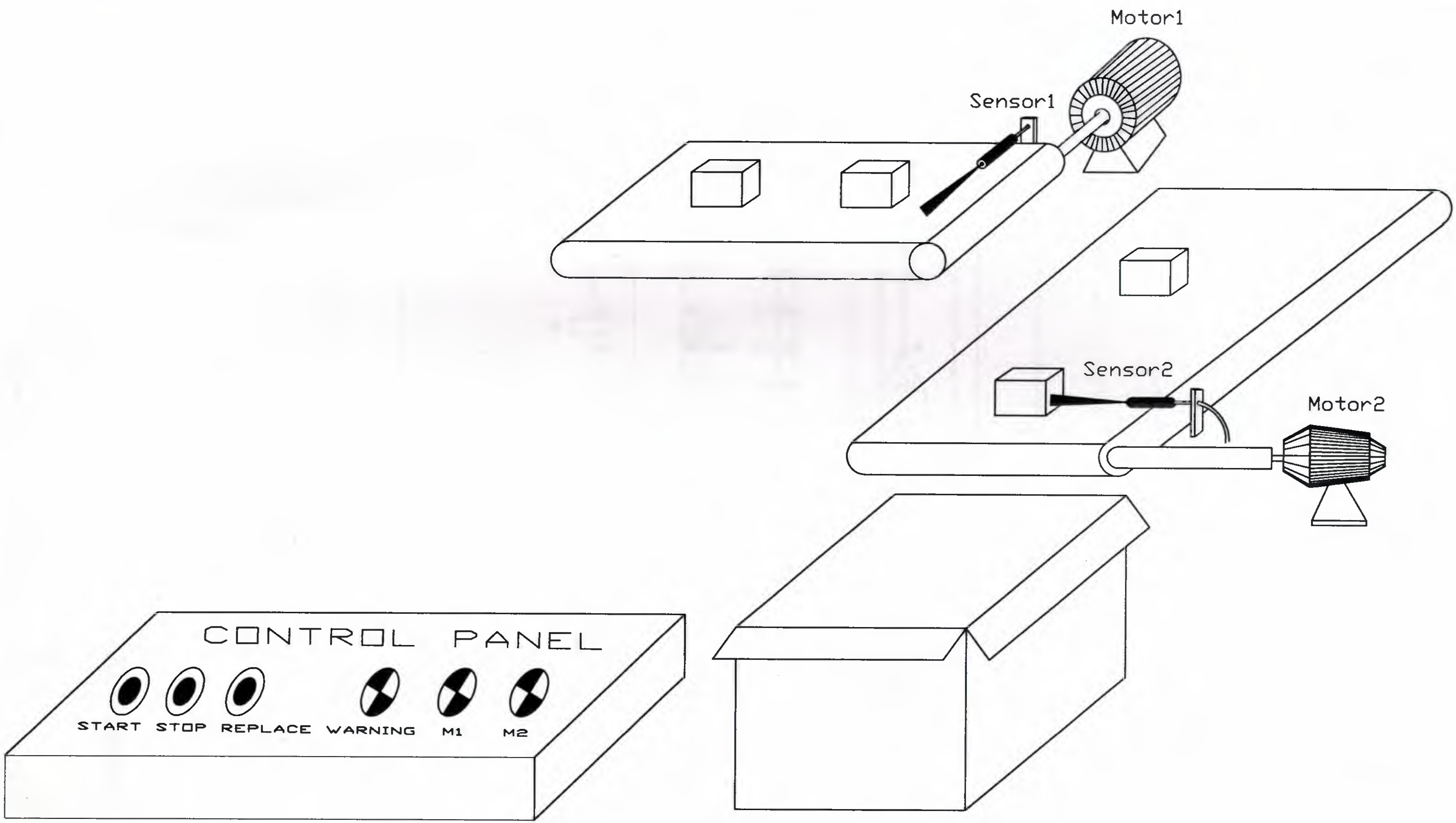
Name-Surname : NECIP FAZIL BATTAL	PROJECT: CONVEYOR CARRYING SYSTEM CONTROL BY PLC	PAGE NO:	1
Student No : 20011431 NEAR EAST UNIV.		DRAWING CODE:	02
Supervisor : ÖZGÜR CEMAL ÖZERDEM	DEFINITION: CONVEYOR CARRYING MECHANISM (2mm iron sheet used)	DATE:	02 MAY 2006



CONVEYOR CARRYING SYSTEM

NCP12

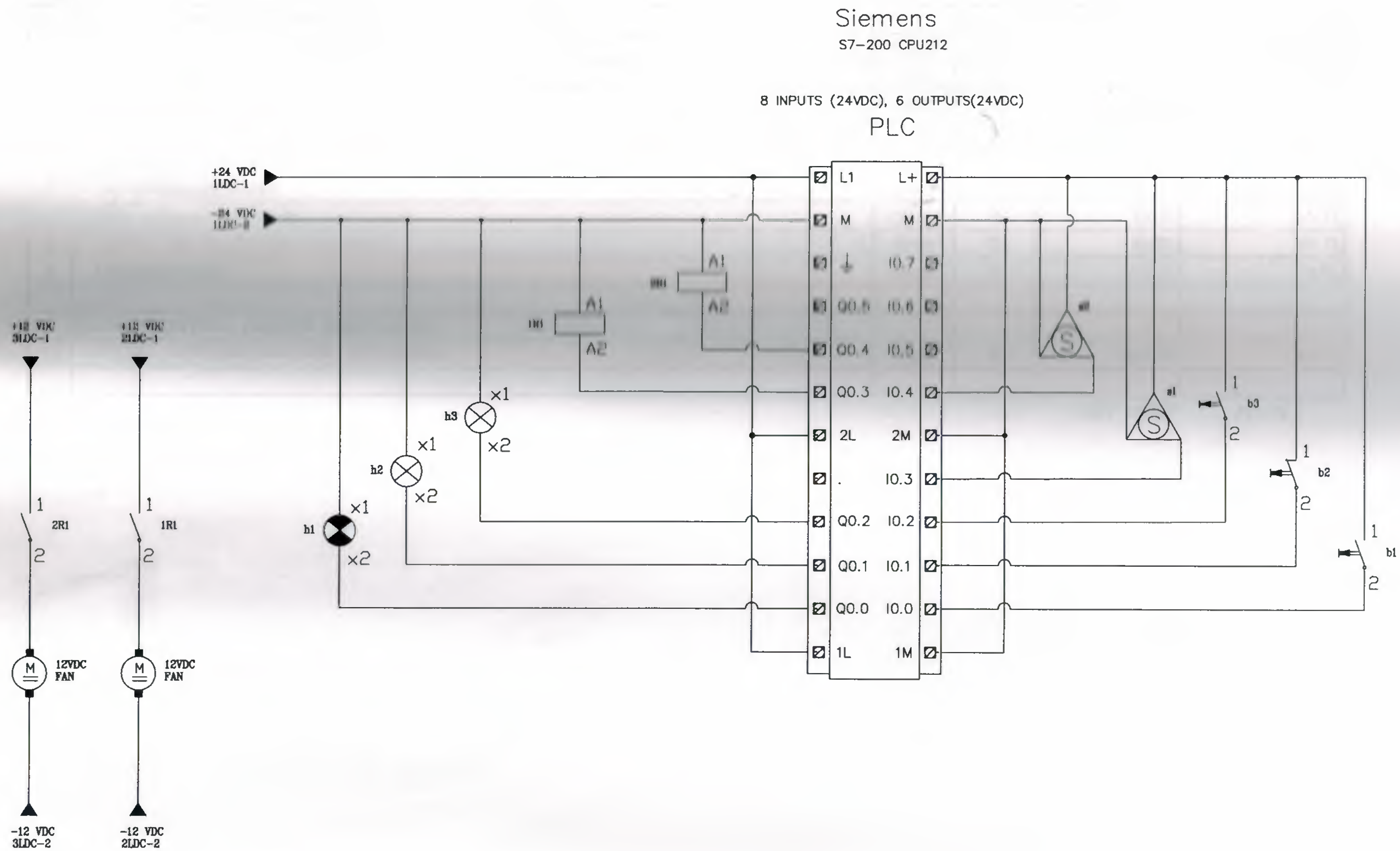
Name-Surname :	NECIP FAZIL BATTAL	PROJECT:	CONVEYOR CARRYING SYSTEM CONTROL BY PLC	PAGE NO:	1
Student No :	20011431 NEAR EAST UNIV.			DRAWING CODE:	03
Supervisor :	ÖZGÜR CEMAL ÖZERDEM	DEFINITION:	CONVEYOR MECHANICAL and ELECTRICAL PARTS	DATE:	08 MAY 2006



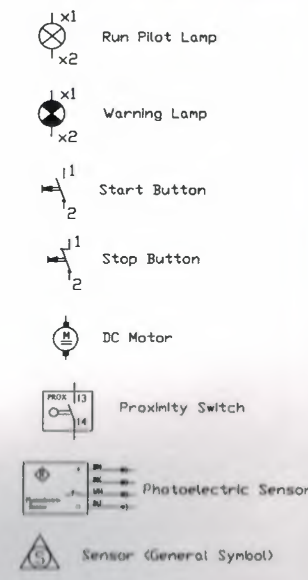
CONVEYOR CARRYING SYSTEM VIEW

NCP12

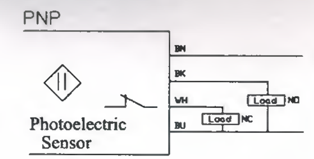
Name-Surname : NECIP FAZIL BATTAL	PROJECT: CONVEYOR CARRYING SYSTEM CONTROL BY PLC	PAGE NO:	1
Student No : 20011431 NEAR EAST UNIV.		DRAWING CODE:	04
Supervisor : ÖZGÜR CEMAL ÖZERDEM	DEFINITION: CONVEYOR CARRYING MECHANISM and CONTROL PANEL	DATE:	05 MAY 2006



ELECTRICAL SYMBOLS



SENSOR CONNECTION DIAGRAM



INPUTS/OUTPUTS TABLE

INPUTS	
Device	Adress
Start Button	I0.0
Stop Button	I0.1
Replace Button	I0.2
Sensor1	I0.3
Sensor2	I0.4

OUTPUTS	
Device	Adress
Red Pilot Lamp	Q0.0
Yellow Pilot Lamp	Q0.1
Green Pilot Lamp	Q0.2
Motor1	Q0.3
Motor2	Q0.4

CIRCUIT SCHEMATICS OF CONVEYOR CONTROL SYS.

NCP12

Name-Surname :	NECIP FAZIL BATTAL	PROJECT:	CONVEYOR CARRYING SYSTEM CONTROL BY PLC	PAGE NO:	1
Student No :	20011431 NEAR EAST UNIV.			DRAWING CODE:	05
Supervisor :	ÖZGÜR CEMAL ÖZERDEM	DEFINITION:	CIRCUIT SCHEMATICS and ELECTRICAL CONNECTIONS	DATE:	14 MAY 2006

COST CALCULATION OF CONVEYOR CARRYING SYSTEM 28.05.2006

No	Definition of Work / Production / Equipment	Unit	Quant.	Unit Price(YTL)	Total Cost (YTL)
	CONVEYOR BELT EQUIPMENTS				
1	Conveyor Carrying System (2mm Iron Sheet)	Global	1	70 TL	70 TL
2	8mm Screws for Conveyor Belt	Each	7	1,5 TL	11 TL
3	4mm Screws for Conveyor Belt	Each	10	0,6 TL	6 TL
4	Sky-e M18 Diffuse Photoelectric Sensor PNP out N/O - N/C Contact Sn=150mm 10-30VDC	Each	2	70 TL	140 TL
5	Start/ Stop Button Switch	Each	3	5 TL	15 TL
6	Signal Lamp 220V-24V	Each	3	4 TL	12 TL
7	Control Panel Metal Profil	Global	1	15 TL	15 TL
8	12VDC - 3A Car Radiator Fan Motor	Each	2	40 TL	80 TL
9	LIY(St)CY Cable	Meter	4	1,5 TL	6 TL
	MOTOR CONTROL CIRCUIT EQUIPMENTS				
10	24VDC - 10A Type Relay N/O - N/C Contact	Each	2	8,0 TL	16 TL

NET COST ***371 TL***