1. **Introduction**

2. **System Configuration**

3. **Features and Functions**

1. INTRODUCTION

❑ What is a Control System ?

In general, a Control System is a collection of electronic devices and equipment which are in place to ensure the stability, accuracy and smooth transition of a process or a manufacturing activity. It takes any form and varies in scale of implementation, from a power plant to a semi-conductor machine. As a result of rapid advancement of technology, complicated control tasks accomplished with a highly automated control system, which may be in the form of Programmable Controller (PLC) & possibly a host computer, etc. Besides signal interfacing to the field devices (such as operator panel, motors, sensors, switches, solenoid valves and etc.), capabilities in network communication enable a big scale implementation and process co-ordination besides providing greater flexibility in realizing distributed control system. Every single component in a control system plays an important role regardless of size. For instance, as shown in Fig 1.1 the PLC would not know the happenings around it without any sensing devices. And if necessary, an area host computer has to be in place to co-ordinate the activities in a specific area at the shopfloor.

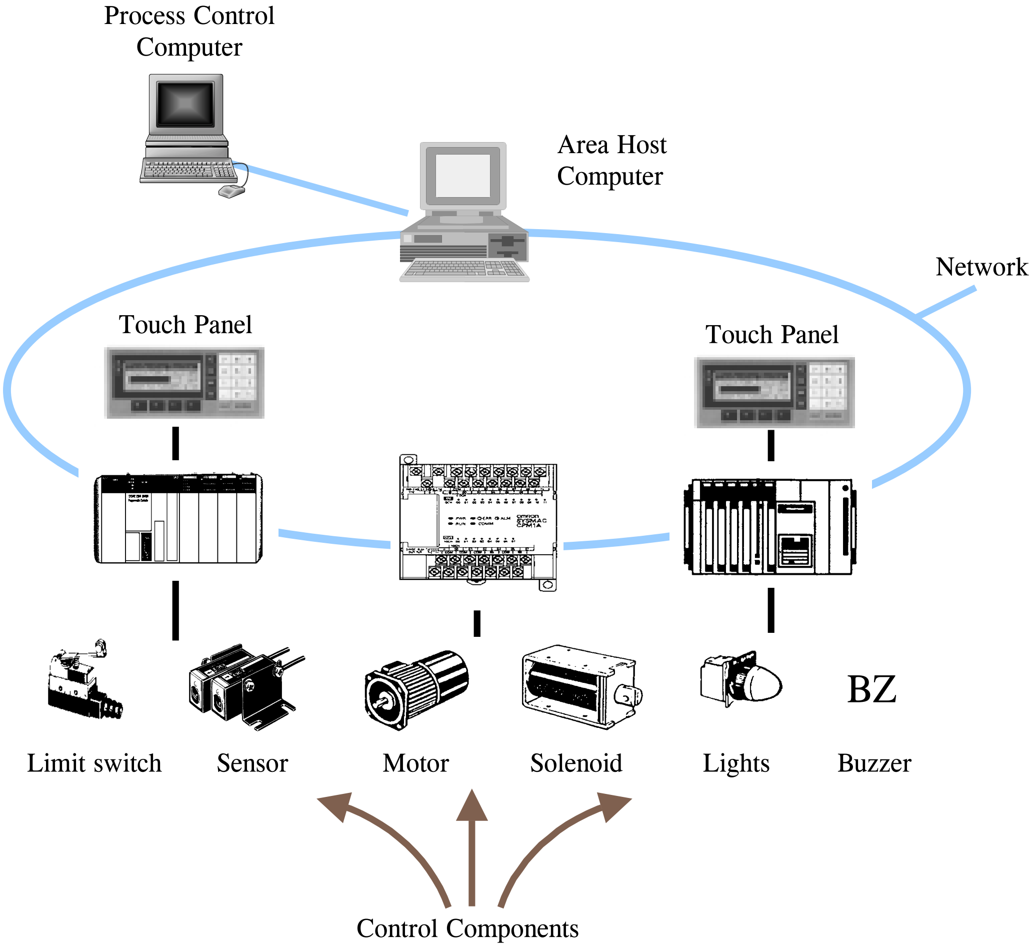
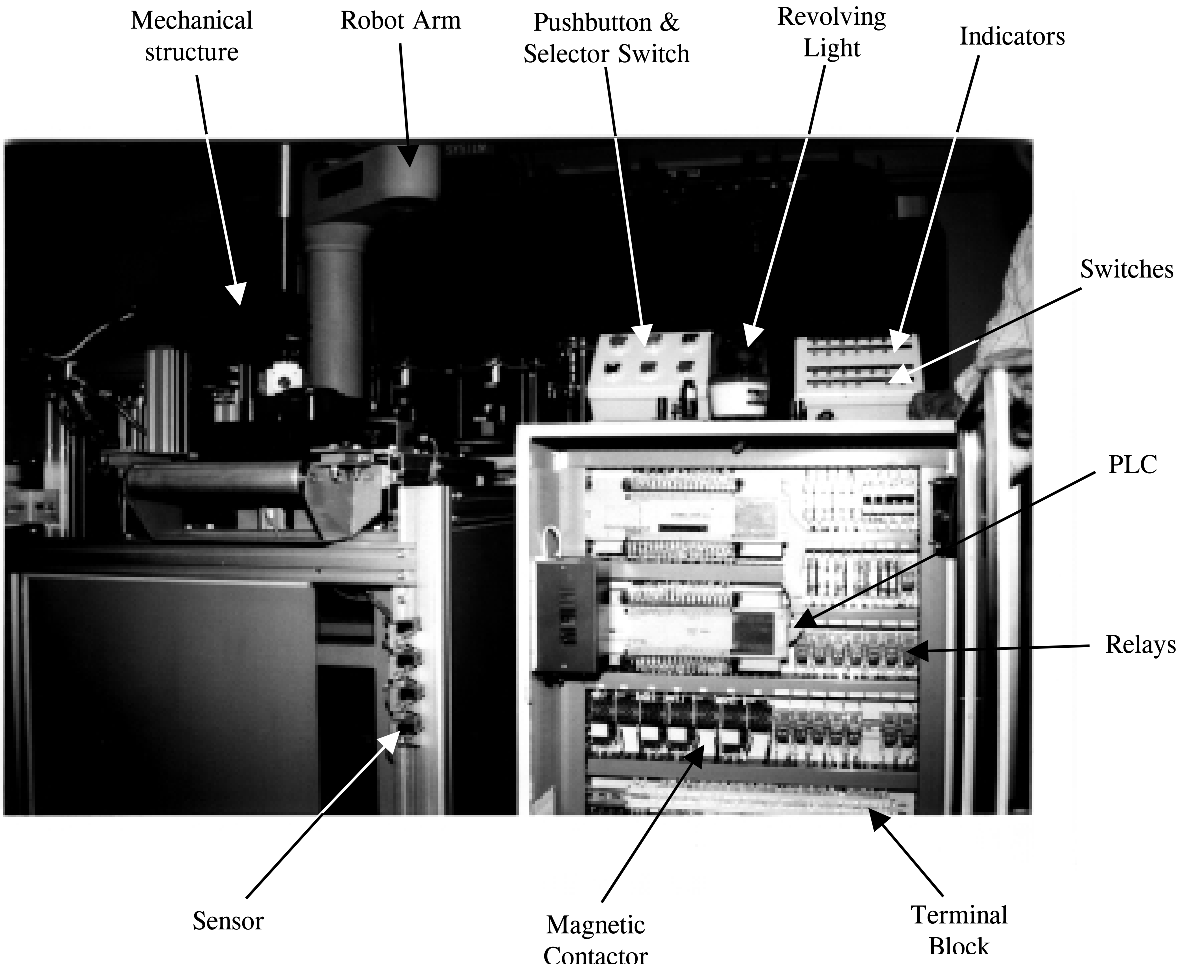


Fig. 1.1:

It could also be an application as small as a single PLC controlling a single or some output devices.

1. INTRODUCTION



❑ Typical Programmable Logic Controller-base Control System

Fig 1.2, is a typical application of a Gantry Robot Control Machine. It is used in a pick and place operation. The whole process sequence is controlled by a PLC. The various input devices such as selector switches, push buttons, togle switches, sensors are connected to the input of the PLC via the input terminal block. The output devices such as the revolving light, indicators, relays, contactors and solenoid valves are connected to the output terminals of the PLC. The whole process is controlled by a ladder program loaded into the PLC CPU memory. The program will execute a sequence automatically according to the pre-defined sequence of operations. Manual operation are also provided to allow operator to activate the machine manually by the switches, emergency push-button for the purpose of safety in case you need to stop the operation abruptly. In this application, the control system operates as a stand-alone operation

1. INTRODUCTION

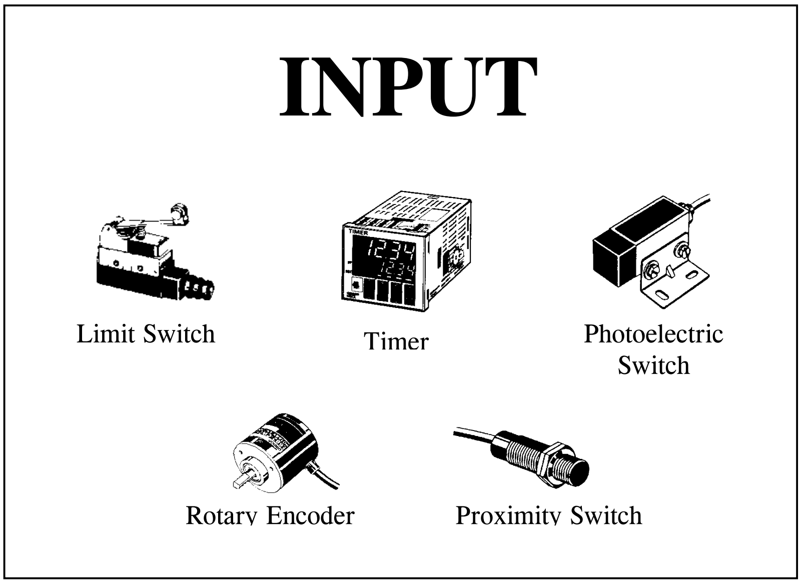
❑ The Role of the Programmable Controllers (PLC)

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

The PLC may be used to control a simple and repetitive task, or a few of them may be interconnected together with other host controllers or host computers through a sort of communication network, in order to integrate the control of a complex process.

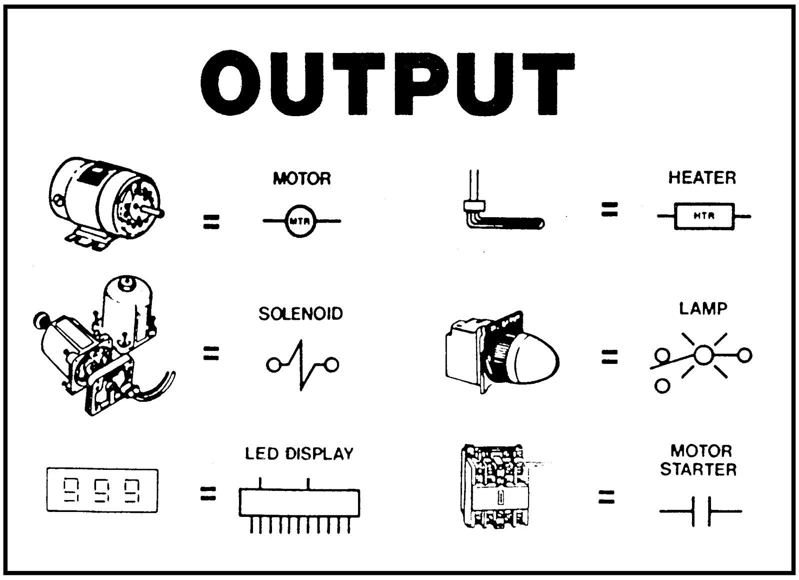
Input Devices
Intelligence of an automated system is greatly depending on the ability of a PLC to read in the signal from various types of automatic sensing and manual input field devices.

Push-buttons, keypad and toggle switches, which form the basic man-machine interface, are types of manual input device. On the other hand, for detection of workpiece, monitoring of moving mechanism, checking on pressure and or liquid level and many others, the PLC will have to tap the signal from the specific automatic sensing devices like proximity switch, limit switch, photoelectric sensor, level sensor and so on. Types of input signal to the PLC would be of ON/OFF logic or analogue. These input signals are interfaced to PLC through various types of PLC input module.



Output Devices
An automatic system is incomplete and the PLC system is virtually paralysed without means of interface to the field output devices. Some of the most commonly controlled devices are motors, solenoids, relays indicators, buzzers and etc. Through activation of motors and solenoids the PLC can control from a simple pick and place system to a much complex servo positioning system. These type of output devices are the mechanism of an automated system and so its direct effect on the system performance.

However, other output devices such as the pilot lamp, buzzers and alarms are merely meant for notifying purpose. Like input signal interfacing, signal from output devices are interfaced to the PLC through the wide range of PLC output module.



1. INTRODUCTION

❑ What is a Programmable Controller ?

A PLC consists of a Central Processing Unit (CPU) containing an application program and Input and Output Interface modules, which is directly connected to the field I/O devices. The program controls the PLC so that when an input signal from an input device turns ON, the appropriate response is made. The response normally involves turning ON an output signal to some sort of output devices.

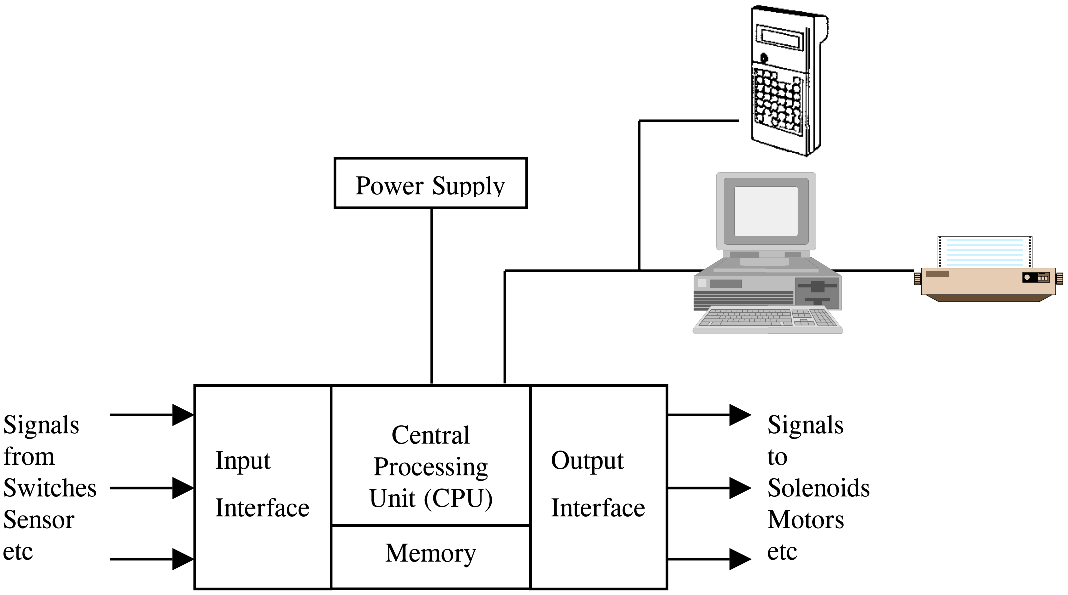

Power Supply
Signals from Switches Sensor etc
Input Interface
Central Processing Unit (CPU)
Memory
Output Interface
Signals to Solenoids Motors etc

Fig. 1.3: Block Diagram of PLC

Central Processing Unit
The Central Processing Unit (CPU) is a microprocessor that co-ordinates the activities of the PLC system. It executes the program, processes I/O signals & communicates with external devices.

Memory
There are various types of memory unit. It is the area that hold the operating system and user memory. The operating system is actually a system software that co-ordinates the PLC. Ladder program, Timer and Counter Values are stored in the user memory. Depending on user’s need, various types of memory are available for choice:

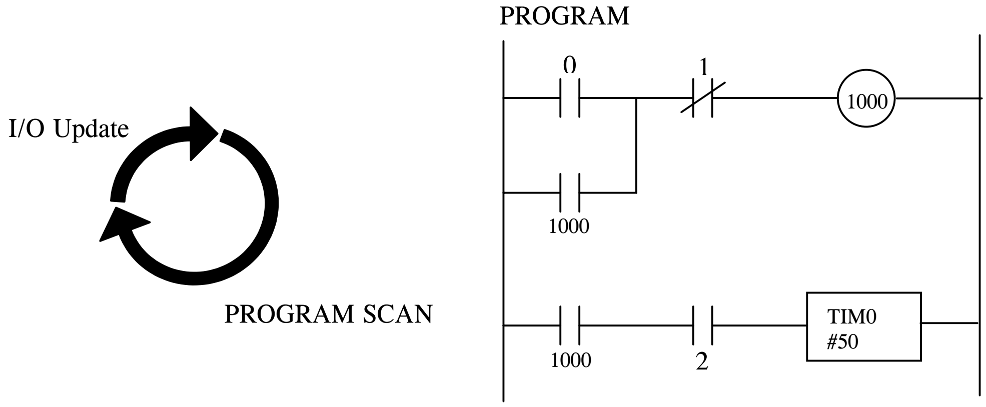
(a) Read –Only Memory (ROM)
ROM is a non-volatile memory that can be programmed only once. It is therefore unsuitable. It is least popular as compared with others memory type.

(b) Random Access Memory (RAM)
RAM is commonly used memory type for storing the user program and data. The data in the volatile RAM would normally be lost if the power source is removed. However, this problem is solved by backing up the RAM with a battery.

(c) Erasable Programmable Read Only Memory (EPROM)
EPROM holds data permanently just like ROM. It dose not require battery backup. However, its content can be erased by exposing it to ultraviolet light. A prom writer is required to reprogram the memory.

(d) Electrically Erasable Programmable Read Only Memory (EEPROM)
EEPROM combines the access flexibility of RAM and the non-volatility of EPROM in one. Its contents can be erased and reprogrammed electrically, however, to a limit number of times.

SCAN TIME
The process of reading the inputs, executing the program and updating the outputs is known as scan. The scan time is normally a continuous and sequential process of reading the status of inputs, evaluating the control logic and updating outputs. Scan time specification indicates how fast the controller can react to the field inputs and correctly solve the control logic.



Factors influencing Scan Time
The time required to make a single scan (scan time) varies from 0.1 ms to tens of ms depending on its CPU processing speed and the length of the user program. The user of remote I/O subsystems increases the scan time as a result of having to transmit the I/O updates to remote subsystem. Monitoring of the control program also adds overhead time to the scan because the controller's CPU has to send the status of coils and contacts to the CRT or other monitoring device.

1. INTRODUCTION

❑ Conventional Control Panel and Its Difficulties

In the beginning of the Industrial revolution, especially in the 1960 & 1970, automated machines were controlled by electromechanical relays. These relays were all hardwired together inside the control panel. In some cases, the control panel was so huge that it could cover the entire wall. Every connections in the relay logic must be connected. Wiring is not always perfect, it takes time to troubleshoot the system. This is a very time consuming affair. On top of that, the relays have limited contacts. If modification is required, the machine has to be stopped, space may not available and wiring has to be traced to accommodate changes. The control panel can only be used for that particular process. It cannot be changed immediately to a new system. It has to be redone. In terms of maintenance, an electrician must be well trained and skillful in troubleshooting the control system. In short, conventional relay control panel are very inflexible.

A typical example of the conventional control panel is shown in Fig. 1.4

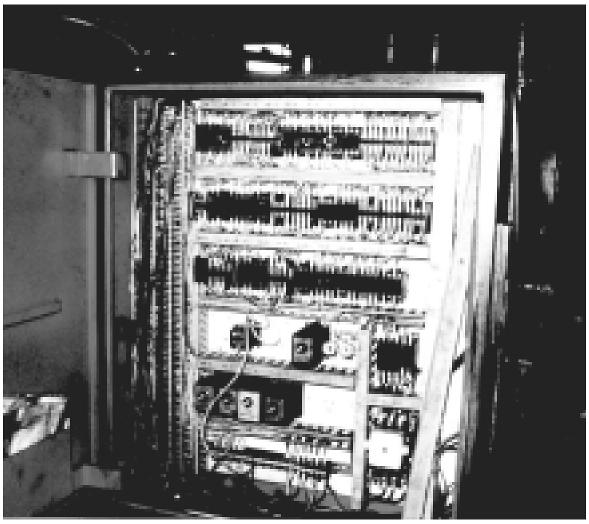


Fig. 1.4: Typical Conventional Control Panel

❑ Disadvantage of Conventional Control Panel

In this panel we can observe the following points

- There are too many wiring work in the panel.
- Modification can be quite difficult.
- Troubleshooting can be quite troublesome as you may require a skillful person.
- Power consumption can be quite high as the coil consumes power.
- Machine downtime is usually long when problems occur, as it takes a longer time to troubleshoot the control panel.
- Drawings are not updated over the years due to changes. It causes longer downtime in maintenance and modification.

1. INTRODUCTION

❑ Programmable Controller Control Panel and Their Advantages

With the arrival of programmable controllers, the control design and concept improve tremendously. There are many advantages in using the programmable controllers.

A typical example of the PLC control panel is shown in Fig 1.5.

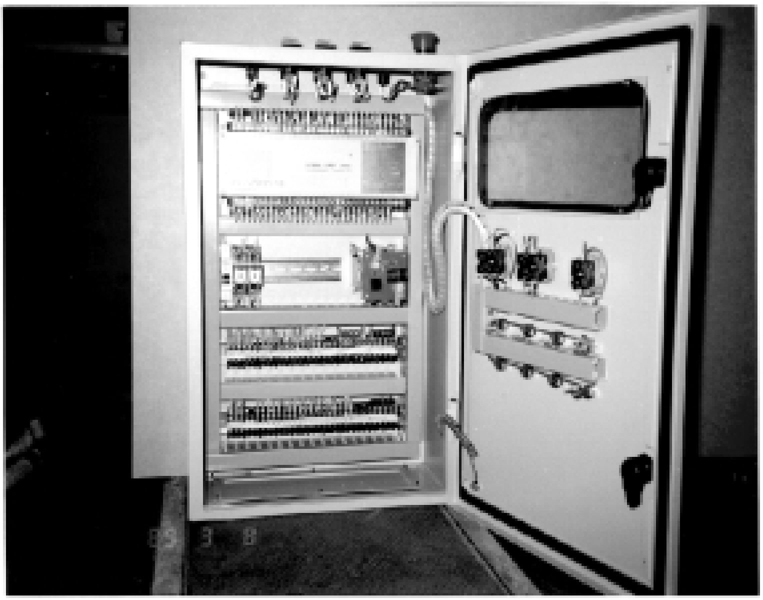


Fig. 1.5: Typical PLC Control Panel

❑ Advantages of PLC Control Panel

- Here are the major advantages that can be distinguishably realized.
- The wiring of the system usually reduces by 80% compared to conventional relay control system.
- The power consumption is greatly reduced as PLC consume much less power.
- The PLC self diagnostic functions enable easy and fast troubleshooting of the system.
- Modification of control sequence or application can easily be done by programming through the console or computer software without changing of I/O wiring, if no additional Input or Output devices are required.
- In PLC System spare parts for relays and hardware timers are greatly reduced as compared to conventional control panel.
- The machine cycle time is improved tremendously due to the speed of PLC operation is a matter of milliseconds. Thus, productivity increases.
- It cost much less compared to conventional system in situation when the number of I/Os is very large and control functions are complex.
- The reliability of the PLC is higher than the mechanical relays and timers.
- An immediate printout of the PLC program can be done in minutes. Therefore, hardcopy of documentation can be easily maintained.

❑ Conversion of Conventional Control Circuit to PLC

Example 1: Starting and Stopping of a 3-phase motor.

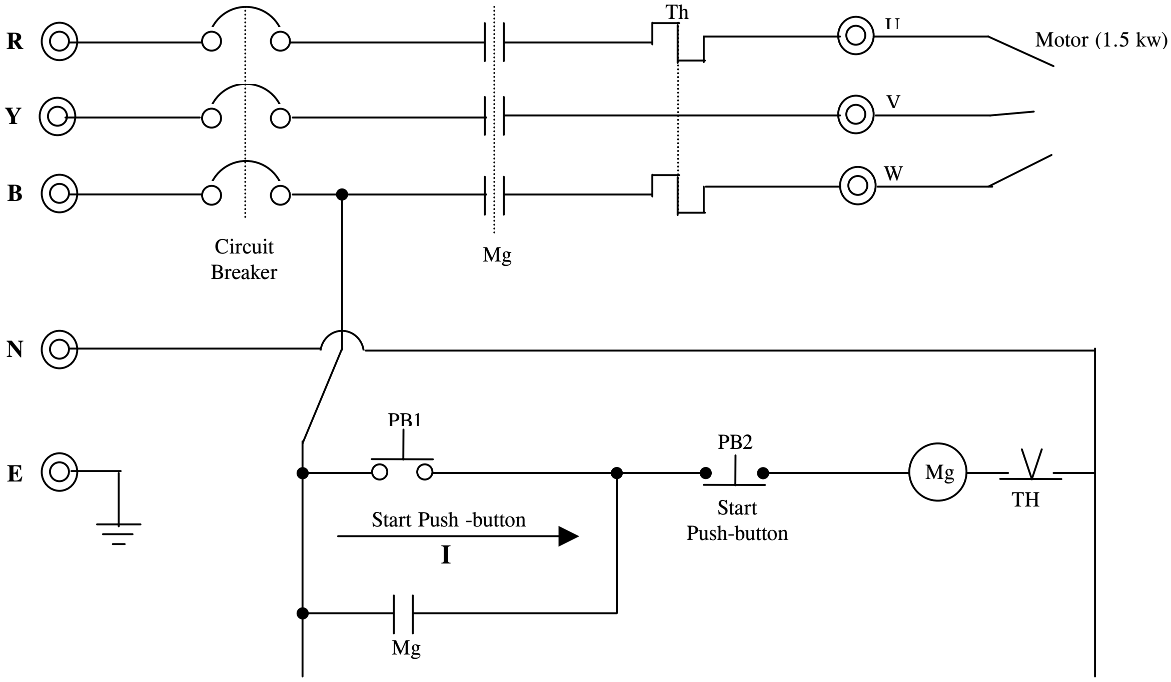


Fig. 1.5

When the push-button PB1 is pressed, current I will flow through the circuit and energize magnetic contact Mg which in turn closes the Mg contacts. The contact Mg parallel the push-button PB1 is for self-holding so that PB1 can be released. The other Mg contacts closes to switch on the 3-pbase motor.

To connect the above circuit in a PLC system to PLC wiring circuit, we need to identify the input and output devices. The input devices are start push-button (PB1) and stop push-button (PB2) and the output device in this case is only one magnetic contactor that controls the 3-phase motor. The wiring circuit is shown in Fig. 1.6

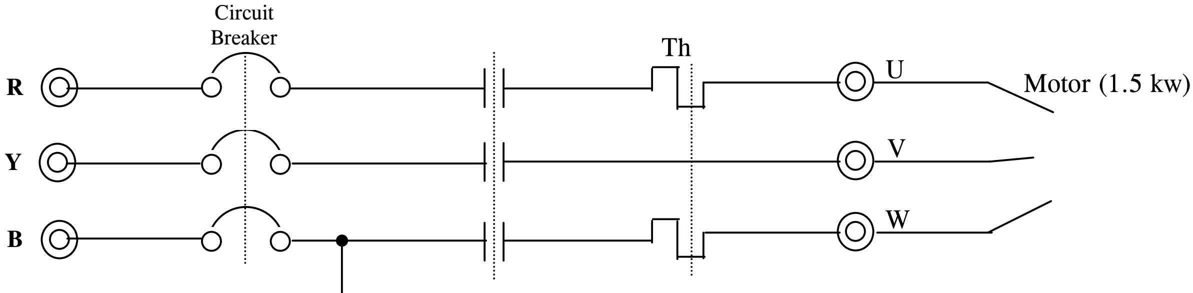
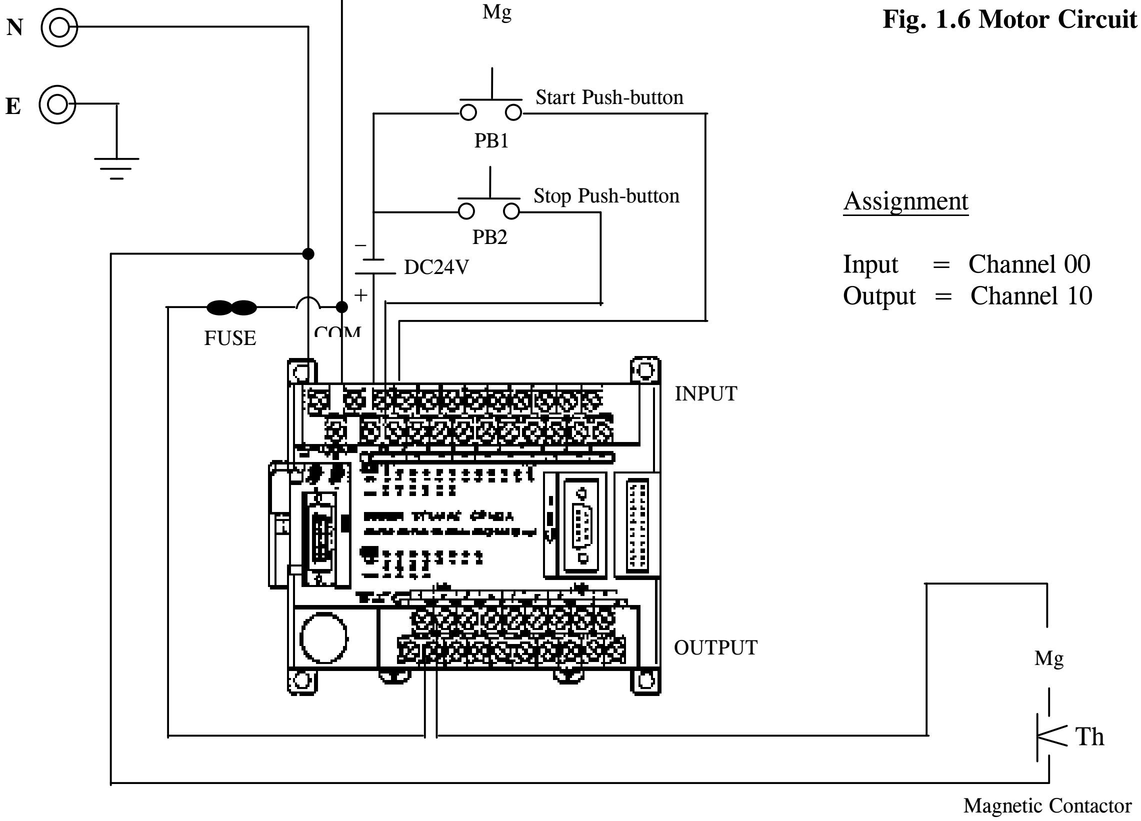


Fig. 1.6 Motor Circuit



Assignment

Input = Channel 00
Output = Channel 10

Fig. 1.7 Hard Wire Circuit for PLC Connection

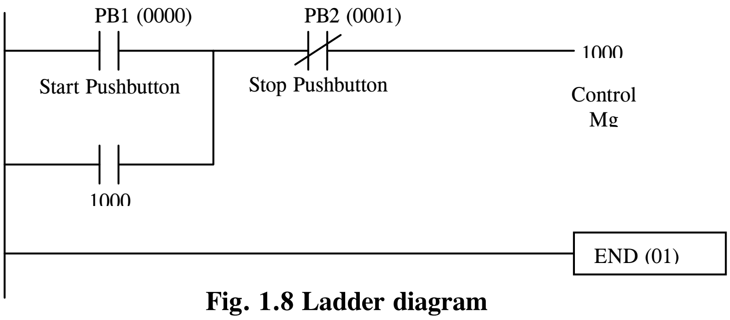


Fig. 1.8 Ladder diagram

Fig. 1.7 shows the wiring circuit of the I/O devices.
Fig. 1.8 is the ladder diagram for the conversion. It must be programmed into the PLC.

Example 2: Sorting Machine

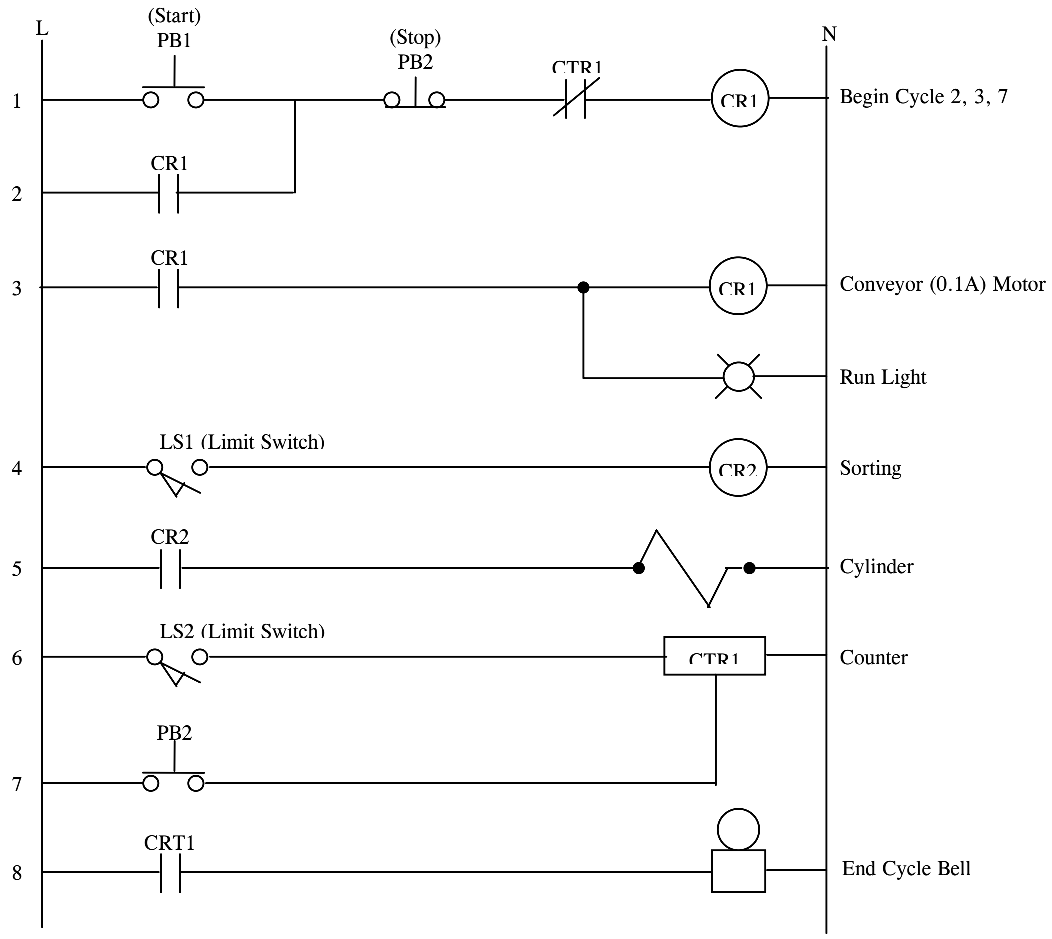


Fig. 1.9 Conventional Circuit for Sorting Machine

In this example, you have to determine again the number of input and output devices used for the control circuit. Assign the I/O for the PLC and then wire it according to the actual I/O devices. The wiring circuit is shown in Fig. 1.10. The input devices such as PB1 is assign to the input 0, LS1 as input 1, LS2 as input 2 and PB2 as input 3. The output devices such as conveyor motor is assigned to output 1000, cylinder solenoid as output 1001 and End Cycle Bell as output 1002.

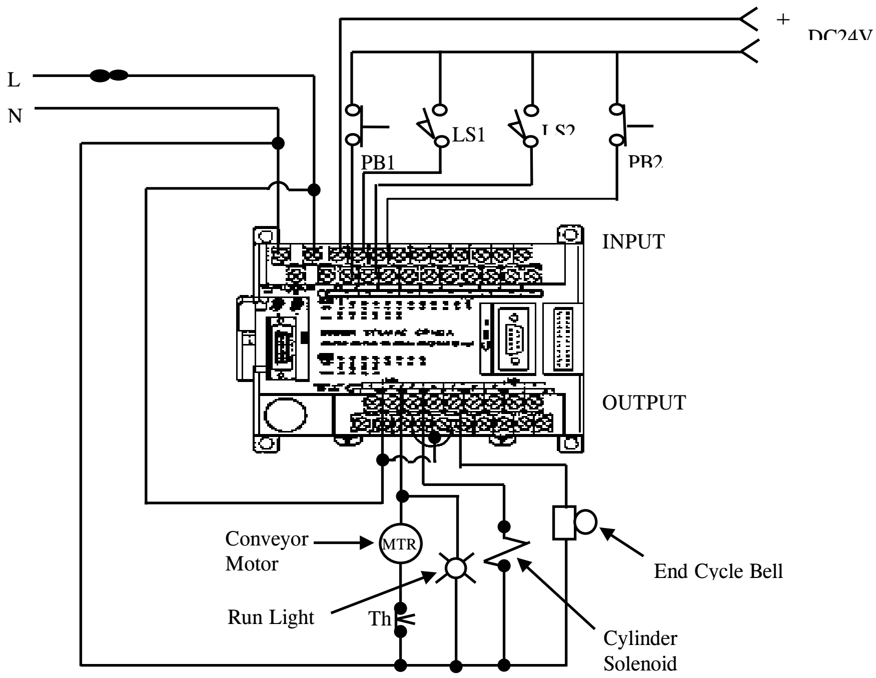


Fig. 1.10 Wiring Circuit of PLC for Sorting Machine

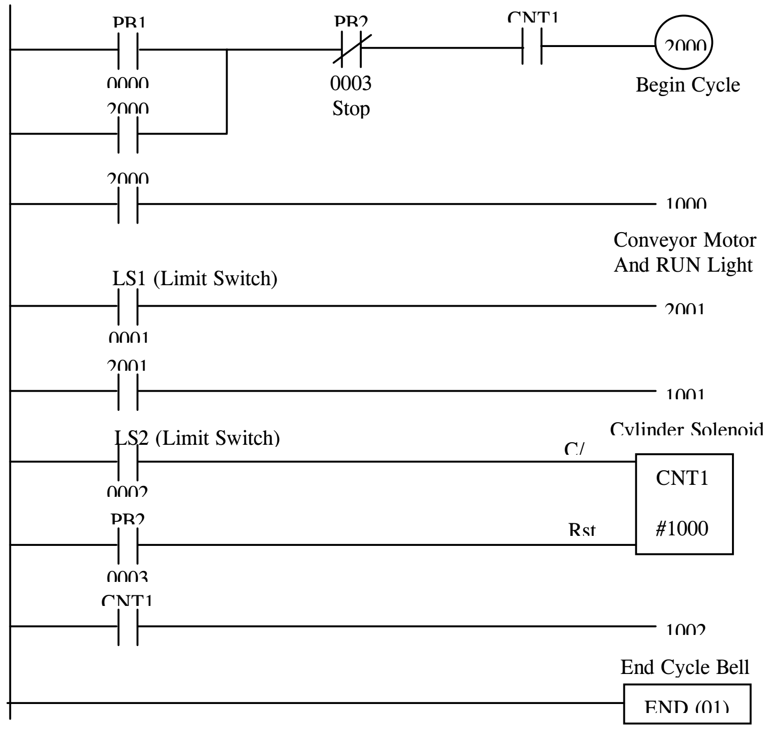


Fig. 1.11 Ladder Diagram of PLC

Fig. 1.10 shows the wiring circuit of the I/O devices.

Fig. 1.11 is the software ladder diagram to be loaded into the PLC in order for the PLC to run according to the sequence of operation.

❑ A Systematic Approach of Control System Design Using A Programming Logic Controller

The concepts of controlling a control system is a very simple and easy task. It involves a systematic approach by following the operation procedure.

1. **Determine The Machine Sequence of Operation**
   Firstly, you have to decide what equipment or system you want to control. The ultimate purpose of the programmable controller is to control an external system. This system to be controlled can be a machine equipment, or process and is often generically called the controlled system.

   The movement of the controlled system is constantly monitored by the input devices that give a specified condition and send a signal to the programmable controller. In response, the programmable controller outputs a signal to the external output devices which actually controls the movement of the controlled system as specified and thus achieves the extended control action. In simplicity, you need to determine the sequence of the operation by drawing out the flowchart.

2. **Assignment of Inputs and Outputs**
   Secondly, all external input and output devices to be connected to the programmable controllers must be determined. The input devices are the various switches, senses, etc. The output devices are the solenoids, electromagnetic valves, motor, inductors etc.

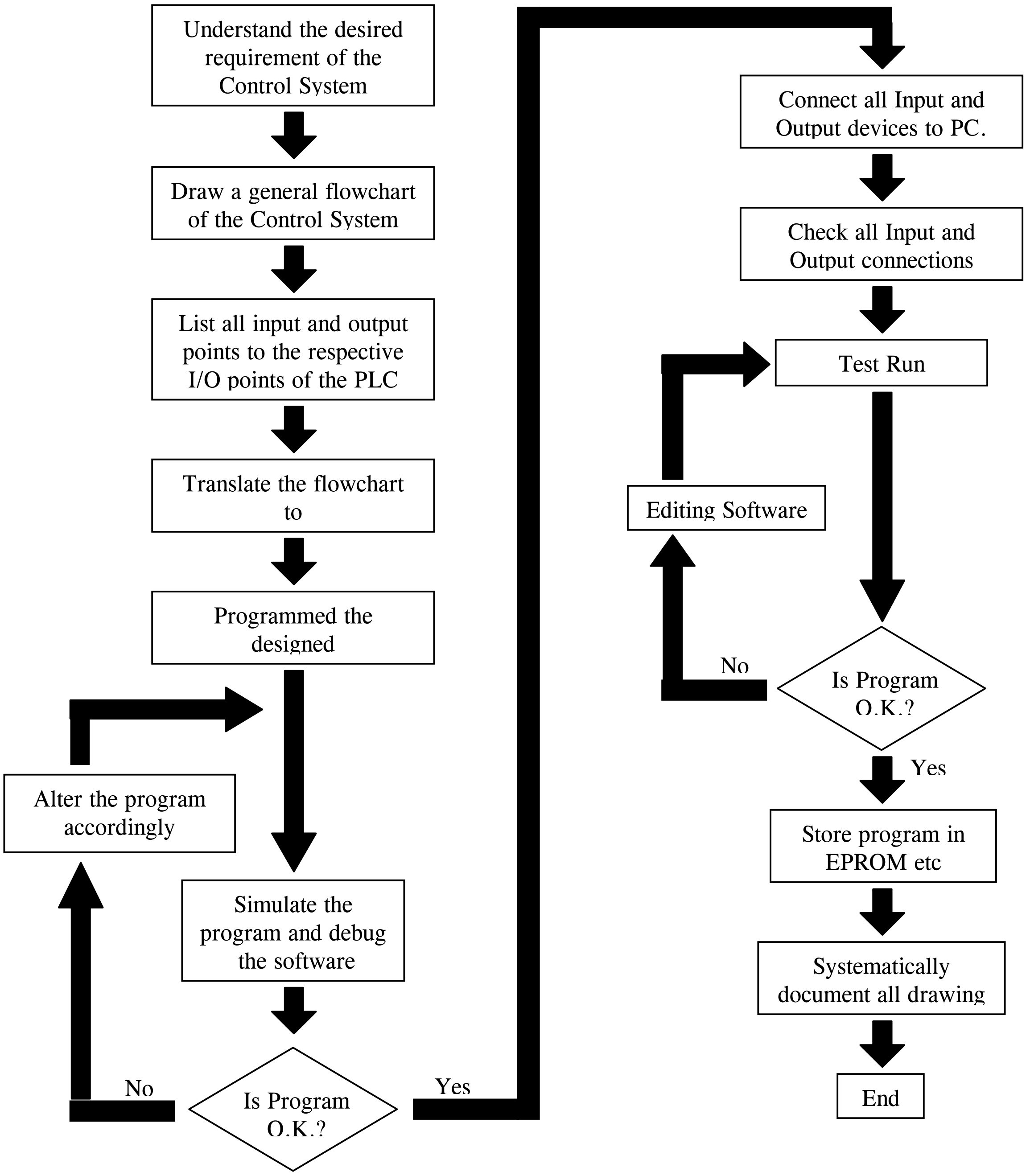
   After identifying all the various INPUT and OUTPUT devices, assigned the numbers corresponding to the INPUT and OUTPUT number of the particular programmable controller you will be using. The actual wiring will follow the numbers of the programmable controller. The assignment of INPUT and OUTPUT numbers must be carried out before writing the ladder diagram because the number dictate what is the precise meaning of the contacts in the ladder diagram.

3. **Writing of the Program**
   Next, write the ladder diagram program by following the control system sequence of operation as determined by step one.

4. **Programming into Memory**
   Now, you can apply power to the programmable controller. Depending on the type of programmable controller, you may have to do a I/O generation to prepare the system configuration. After that, you can enter your program in the memory either by programming console or by computer aided ladder software tool. After completion of the programming, you should check for any coding errors by means of diagnostic function, and if possible simulate the whole operation to see that it is alright.

5. **Running the System**
   Before the start push-button is pressed, thoroughly ensure that the input and Output wiring are correctly connected according to the I/O Assignment. Once confirmed, the actual operation of the PLC can now be started. You may need to debug along the way and fine tune the control system if necessary. Test run thoroughly until it is safe to operate by anyone.

A SYSTEMATIC APPROACH TO PROGRAMMABLE CONTROLLER DESIGN


Understand the desired requirement of the Control System
Draw a general flowchart of the Control System
List all input and output points to the respective I/O points of the PLC
Translate the flowchart to
Programmed the designed
Alter the program accordingly
Simulate the program and debug the software
No
Is Program O.K.?
Yes
Connect all Input and Output devices to PC.
Check all Input and Output connections
Test Run
Editing Software
No
Is Program O.K.?
Yes
Store program in EPROM etc
Systematically document all drawing
End

1. INTRODUCTION

❑ Programmable Controller Applications

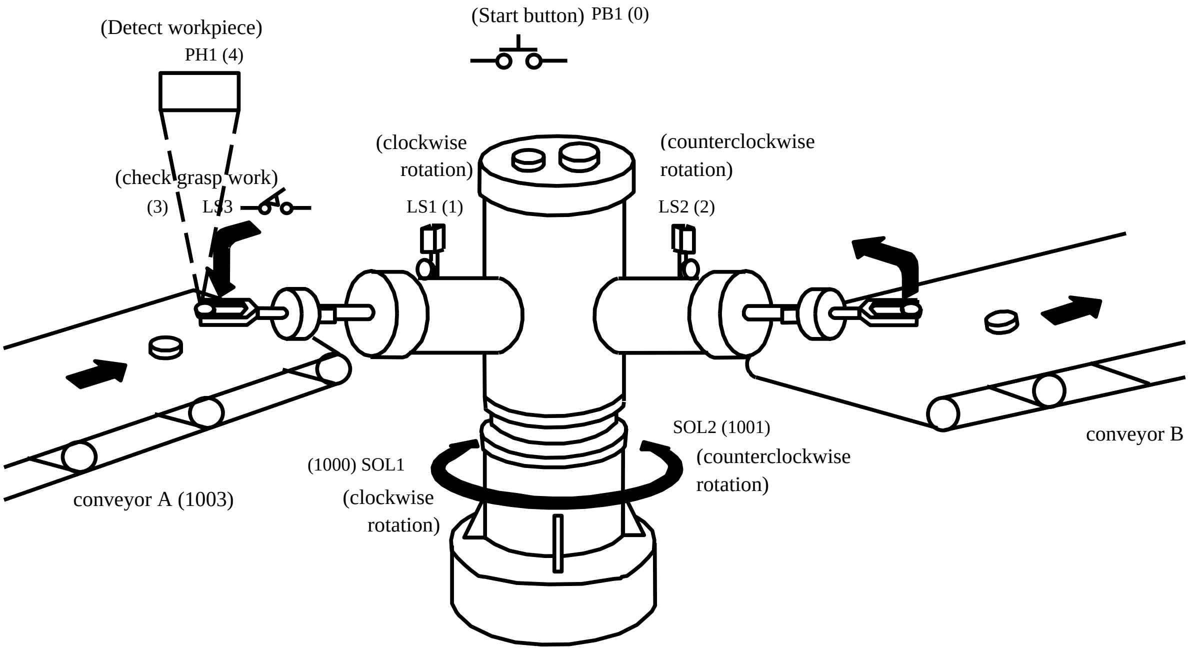
There are so many applications that you can find PLCs are use in the various industries. Here are the list of applications.

- Material Handling
- Conveyor System
- Packaging Machine
- Pick and Place Robot Control
- Pump Control
- Swimming Pool
- Water Treatment
- Chemical Processing Plant
- Paper and Pulp Industries
- Glass Manufacturing
- Precast Concrete Industries
- Cement Manufacturing
- Printing Industries
- Electro-plating Plants
- Food Processing
- Machine Tools
- Tobacco Industries
- Plastic Moulding machine
- Semi-conductor Manufacturing Machine
- Sugar Manufacturing Plant
- Palm Oil Manufacturing Plant
- Air Condition Control
- Power Station Plant
- Process Monitoring Control
- Electrical/Electronic Appliance Manufacturing
- Disk Drive Manufacturing
- Petrol Chemical Plant
- Traffic Light System
- Train Control Station system
- Plastic Manufacturing Industries
- Car Manufacturing Plant
- Iron and Steel Mill
- Diary Product Manufacturing Plant
- Building Automation
- Tyre Manufacturing
- Integrated Circuit Chip Manufacturing
- Sewage Treatment Plant
- Security Control System
- Lift Control System
- Generator Control System
- Amusement Park Control

1. INTRODUCTION

❑ Consider A Simple Project

Suppose you wish to control the following robot's movement.



This kind of robot is seen in many automated factories. As is apparent from the figure, this robot picks up a work being carried on conveyor A, and places it on conveyor B.

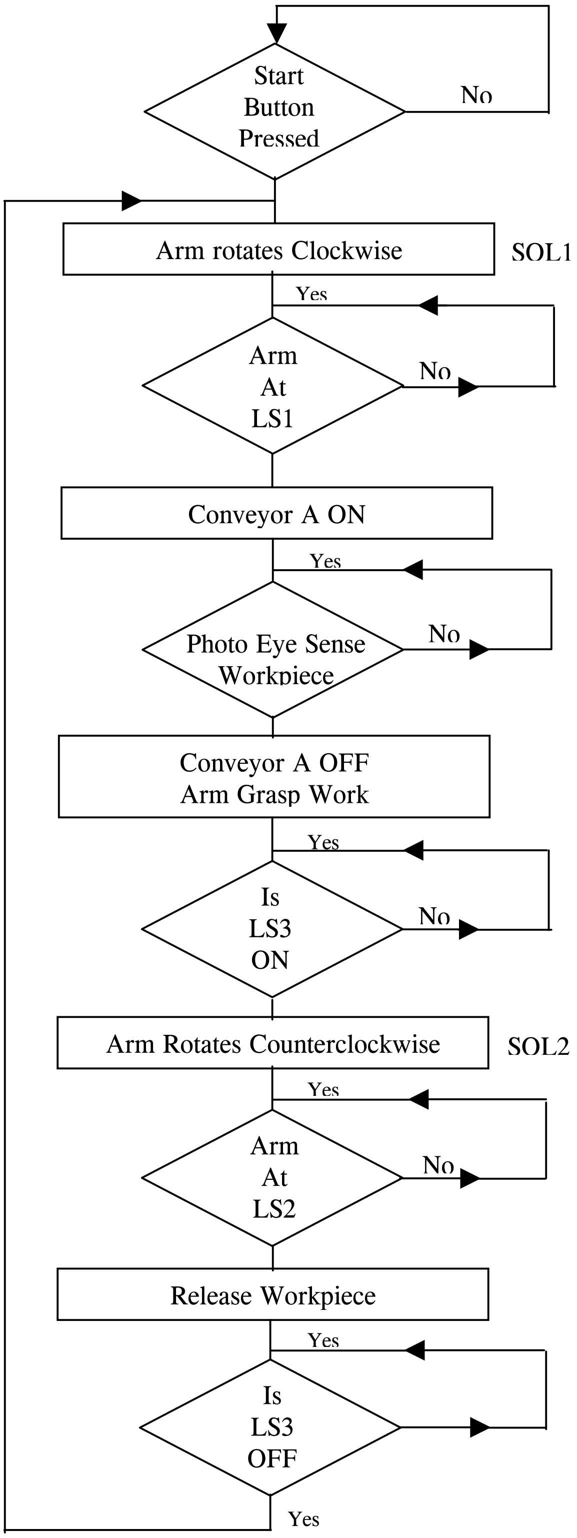
Although seemingly simple, executing this series of operations with electric devices and circuitry is harder than realized.

Looking at the above figure closely will disclose that the robot performs one operation at a time when a given condition is met. Let's analyze these operations and conditions.

1. When the start button is pressed, the robot rotates its arm clockwise.
2. When the robot arm has moved to the position of the work in conveyor A, the arm grasps the work.
3. When the arm has grasped the work, it rotates counterclockwise.
4. When the arm has rotated to the position of conveyor B, it releases the work.

1. INTRODUCTION

1. Determine the Sequence of Operation by Drawing the Flowchart.



STOP CONDITION:

At ant time,
STOP BUTTON is
pressed, the ARM stop
where it is.

Upon pressing start
Button, the arm continues

1. INTRODUCTION

2. Assignment of I/O

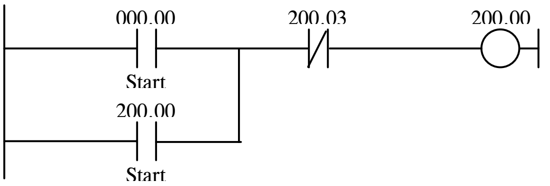
INPUT	DESCRIPTION	OUTPUT	DESCRIPTION
00000	start push button	1000	SOL 1
00001	LS 1	1001	SOL 2
00002	LS 2	1002	SOL 3
00003	LS 3	1003	Conveyor A
00004	PH 1		
00005	stop button		
00006	reset button		

3. Writing the Program

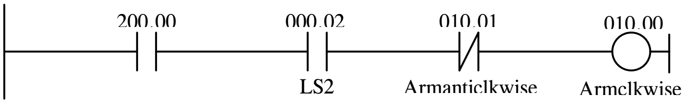
Ladder Diagram : Main 1 Network 1

Main 1 – Robot control
This program demonstrates that the robot picks up a work piece from conveyor A and places it on conveyor B.

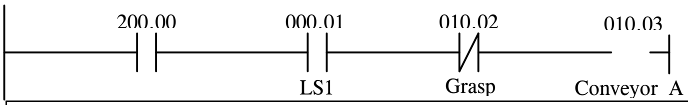
Network 1 – Start condition



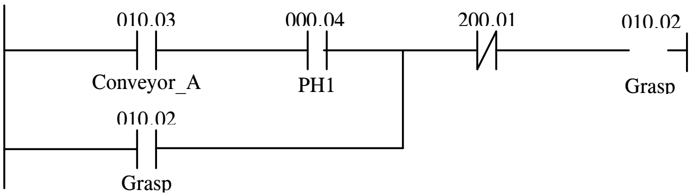
Network 2 – Arm clockwise



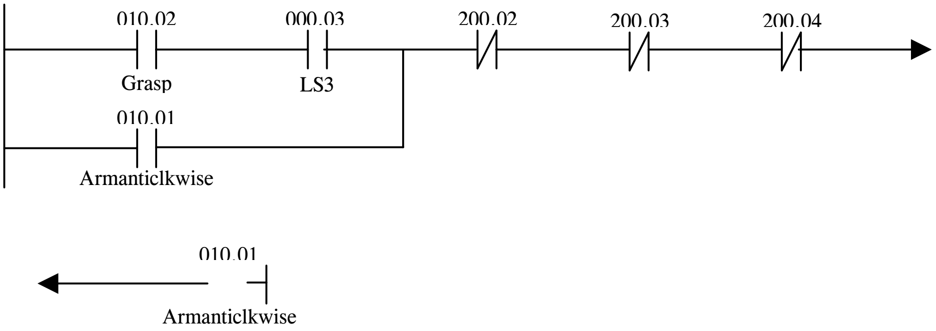
Network 3 – Conveyor A



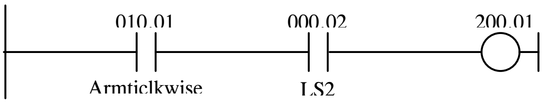
Network 4 – Grasp



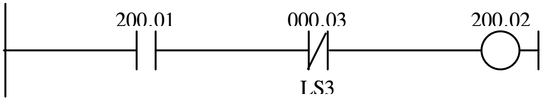
Network 5 – Arm anticlockwise



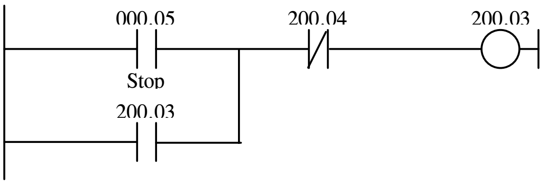
Network 6 – Release part



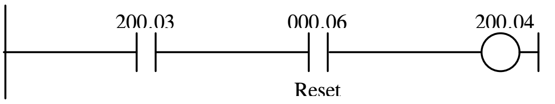
Network 7 – Release CYCLE



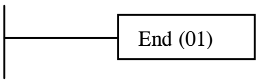
Network 8 – Stop condition



Network 9 – Reset condition
Reset after stop



Network 10 – End



- **Initial System Checks**

Check the following items after setting up and wiring the PLC. Be sure to check the wiring and connections before performing a test run.

Item	Points to Check
Power supply and I/O connections	Is the wiring correct? Are the terminals securely tightened? Are there any shorts between crimp connectors or wires?
Connecting cables	Are the cables all connected correctly and locked?

- **Test run Procedure**

1. Power Supply Application
   a) Check the PLC's power supply voltage and terminals connections.
   b) Check the I/O devices' power supply voltage and terminal connections.
   c) Turn on the power supply and check that the "POWER" indicator illuminates.
   d) Use a Peripheral Device or Syswin to set the CPM1A to PROGRAM mode.

2. I/O Wiring Checks
   a) With the CPM1A in PROGRAM mode, check the output wiring by turning on the output bits with the force set and force reset operations.
   b) Check the input wiring with the PLC's input indicators or a Peripheral Device's monitor operations.

3. Test Run
   a) Use a Peripheral Device to set the PLC to RUN or MONITOR mode and check that the "RUN" indicator lights up.
   b) Check the sequence of operation with the force set/reset operations, etc.

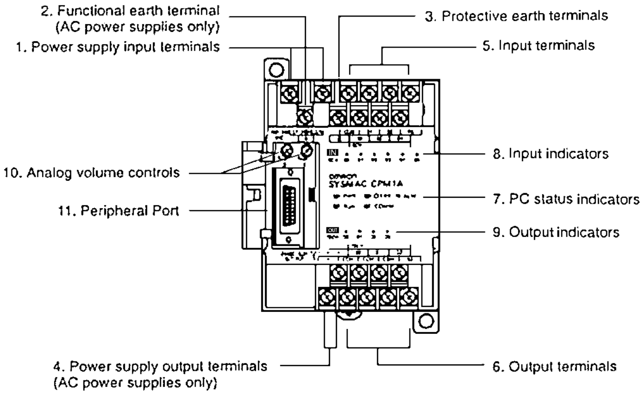
4. Debugging
   Correct any programming errors that are detected.

5. Saving the Program
   a) Use Syswin to save the program to disk.
   b) Print out a hard copy of the program with a printer.

2. SYSTEM CONFIGURATION

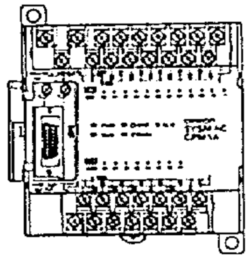
❑ CPU Components

CPM1A-10CDR-

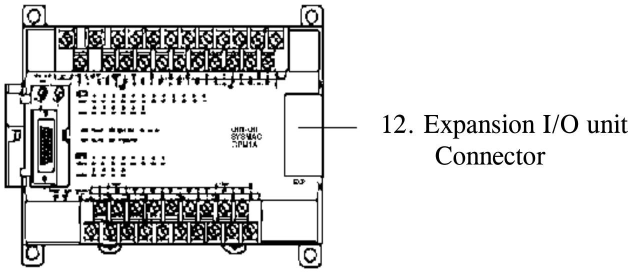


2. Functional earth terminal (AC power supplies only)
1. Power supply input terminals
3. Protective earth terminals
5. Input terminals
10. Analog volume controls
11. Peripheral Port
8. Input indicators
7. PC status indicators
9. Output indicators
4. Power supply output terminals (AC power supplies only)
6. Output terminals

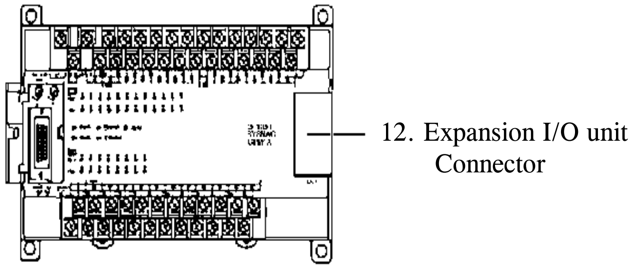
CPM1A-20CDR-



CPM1A-30CDR-



CPM1A-40CDR-



2. SYSTEM CONFIGURATION

- **PLC Status Indicators**

Indicator	Status	Meaning
PWR (green)	ON	Power is being supplied to the PLC.
	OFF	Power isn't being supplied to the PLC.
RUN (green)	ON	The PLC is operating in RUN or MONITPR mode.
	OFF	The PLC is in PROGRAM mode or a fatal error has occurred.
ERR/ALM (red)	ON	A fatal error has occurred. (PLC operation stops.)
	Flashing	A non-fatal error has occurred. (PLC operation continues.)
	OFF	Indicates normal operation.
COMM (orange)	ON	Data is being transferred via the Peripheral Port.
	OFF	Data isn't being transferred via the Peripheral Port.

- **Input Indicators**

These indicators are lit when the corresponding input terminal is ON.
When a fatal error occurs, the input indicators change as follows:

CPU error or I/O bus error:	The input indicators turn OFF
Memory error or system error:	The input indicators maintain their status before the error occurred, even if the input status is changed.

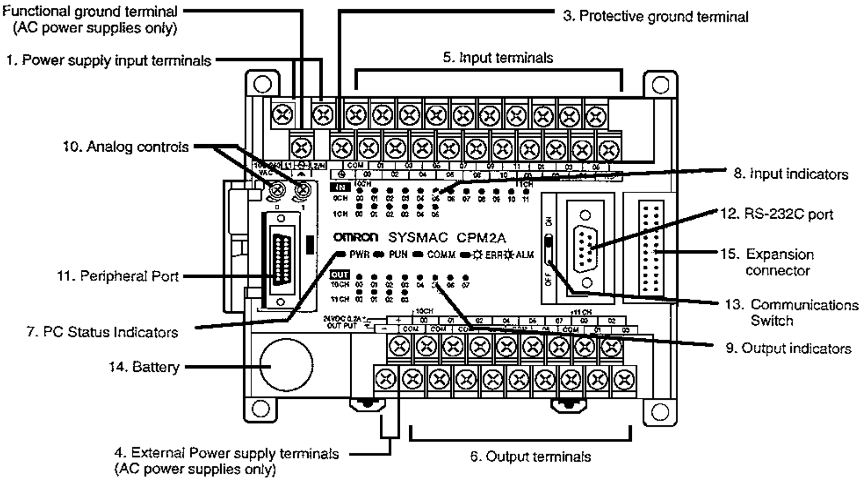
- **Output Indicators**

These indicators are lit when the output relays are turned on.

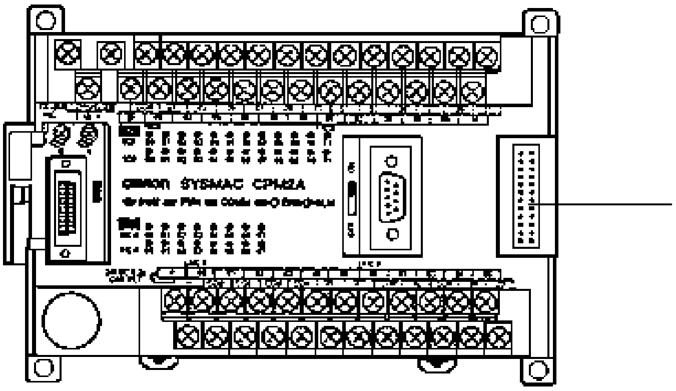
2. SYSTEM CONFIGURATION

❑ CPM2A CPU Components

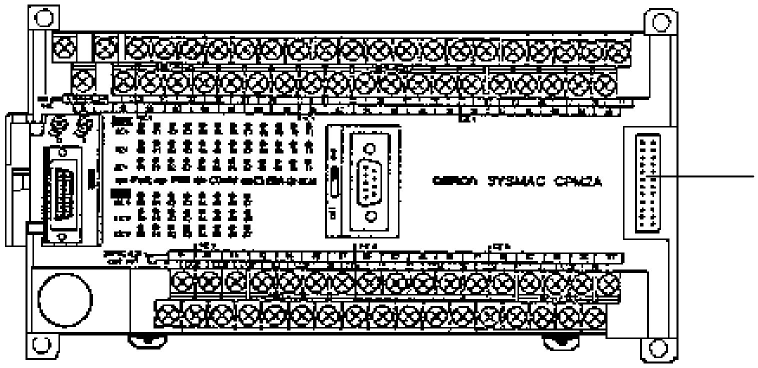
CPM2A-30CD[]-[]



CPM2A-40 CD[]-[]



CPM2A-60 CD[]-[]



- **PLC Status Indicators**

Indicator	Status	Meaning
PWR (green)	ON	Power is being supplied to the PLC.
	OFF	Power isn't being supplied to the PLC.
RUN (green)	ON	The PLC is operating in RUN or MONITOR mode.
	OFF	The PLC is in PROGRAM mode or a fatal error has occurred.
ERR/ALM (red)	ON	A fatal error has occurred. (PLC operation stops.)
	Flashing	A non-fatal error has occurred. (PLC operation continues).
COMM. (orange)	ON	Data is being transferred via the Peripheral Port or RS-232C port.
	OFF	Data isn't being transferred via the Peripheral Port or RS-232C port.

- **Input Indicators**

These indicators are lit when the corresponding input terminal is ON.
When a fatal error occurs, the input indicators change as follows:

CPU error or I/O bus error:	The input indicators turn OFF
Memory error or system error:	The input indicators will change with the status of the input signal, but inputs will not be updated in memory.

- **Output Indicators**

These indicators are lit when the output relays are turned on. The indicators are lit during I/O refreshing. When pulse outputs are being used, the indicator will remain lit continuously while the pulses are being output.

- **Analog Control**

Turn these controls to change the analog setting (0 to 200) in IR 250 and IR 251.

- **Peripheral Port**

Connects the PLC to a Programming Device (including Programming Consoles), host computer, or standard external device.

2. SYSTEM CONFIGURATION

- **RS-232C Port**

Connects the PLC to a programming Device (excluding Programming Consoles), host computers, Programmable Terminal, or standard external device.

- **Communication Setting**

This switch selects whether the Peripheral port and RS-232C port will use the communications setting in the PC Setup or the standard settings.

OFF: The Peripheral Port and RS-232C port operate according to the communication setting in the setup, except for a Programming Console connected to the Peripheral port.

ON: The Peripheral Port and RS-232C port operate according to the standard communications settings, except for a Programming Console connected to the Peripheral port.

- **Battery**

This battery backs up the memory in the CPU Unit and is connected when the Unit is shipped.

- **Expansion Connector**

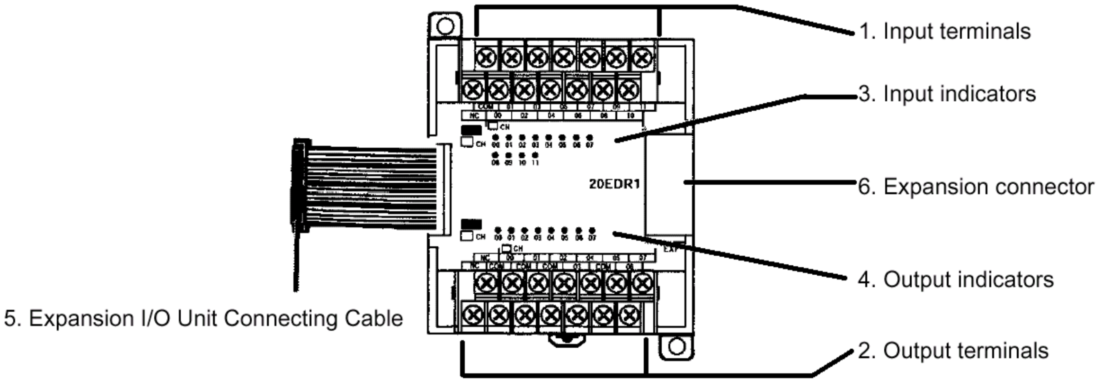
Connects the PLC's CPU unit to the Expansion Unit (Expansion I/O unit, Analog I/O unit, or the CompoBus/S I/O Link Unit.) Up to 3 Expansion Units can be connected to a CPU unit.

2. SYSTEM CONFIGURATION

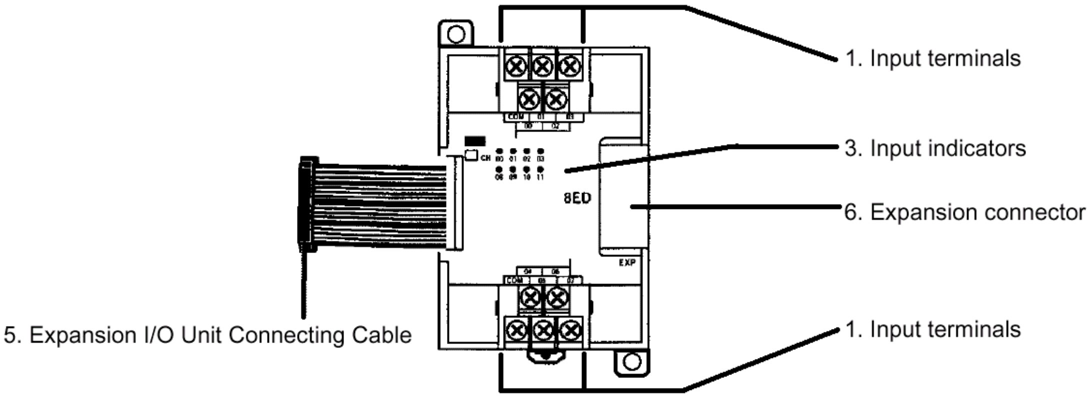
• Expansion I/O Unit Components

The expansion units are shared between CPM1A and CPM2A.
Expansion I/O with 20 I/O Terminals

- CPM1A-20EDR
- CPM1A-20EDT
- CPM1A-20EDT1

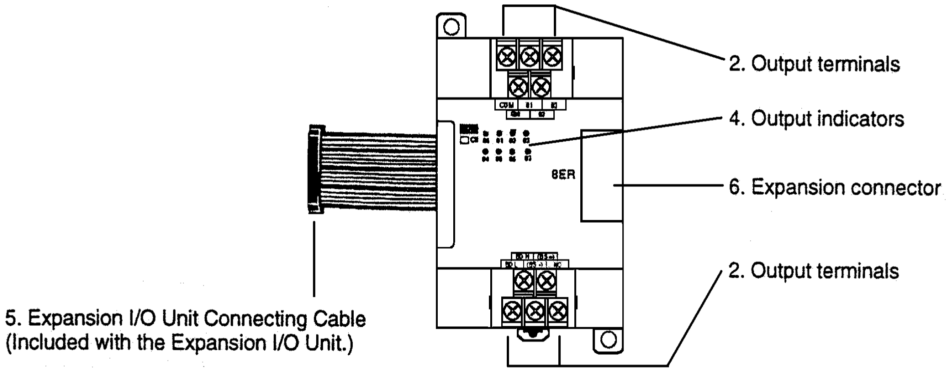


Expansion I/O with 8 Input Terminals
CPM1A-8ED



Expansion I/O with 8 Output Terminals

- CPM1A-8ER
- CPM1A-8ET
- CPM1A-8ET1



1. Input Terminals
   Connect to the input circuits.

2. Output Terminals
   Connect to the output circuits.

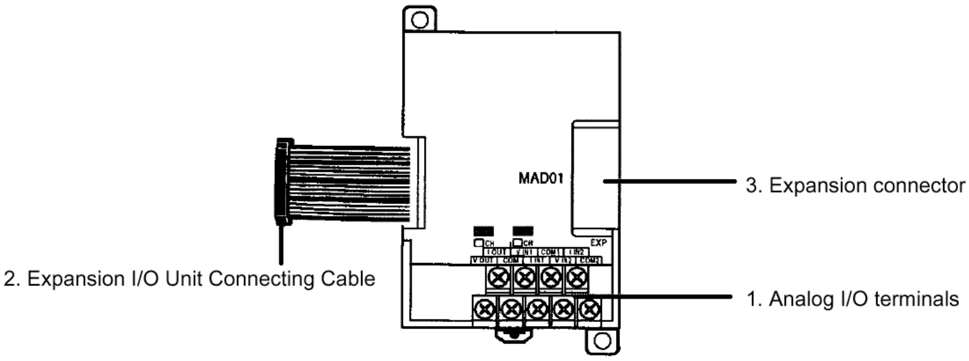
3. Input Indicators
   These indicators are lit when the corresponding input terminal is ON.

4. Output Indicators
   These indicators are lit when the corresponding output terminal is ON.

5. Expansion I/O Unit Connecting Cable
   Connects the Expansion I/O Unit to the Expansion Connector on the PLC's CPU Unit or another Expansion I/O Unit.

6. Expansion Connector
   Connects to another Expansion unit (Expansion I/O Unit, Analog I/O Unit, or CompoBus/S I/O Link Unit). Up to three Expansion Units can be connected to a CPU Unit.

Analog I/O Unit Components
CPM1A-MAD01

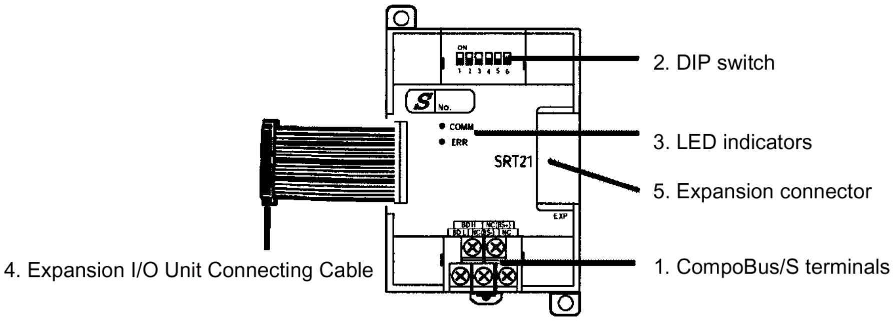


1. Analog Terminals
   Connects the unit to analog input devices and analog output devices.

2. Expansion I/O unit Connecting Cable
   Connects the Analog I/O unit to the Expansion Connector on the PLC's CPU Unit or another Expansion Unit.

3. Expansion Connector
   Connects to another Expansion Unit (Expansion I/O Unit, Analog Unit, or CompoBus/S I/O Link Unit). Up to 3 Expansion Unit can be connected to a CPU unit.

CompoBus/S Terminals
CPM1A-SRT21



2. SYSTEM CONFIGURATION

1. CompoBus/S Terminals

2. Dip Switch
   The DIP switch sets the Unit's node number of CompoBus/S remote I/O network and determines whether or not the outputs will be cleared in the event of a communications error.

3. LED Indicators
   These indicators show the status of the CompoBus/S communications.

4. Expansion I/O Unit Connecting cable
   Connects the CompoBus/s I/O Link Unit to the Expansion connector on the PLC's CPU Unit or another Expansion unit.

5. Expansion Connector
   Connects to another Expansion Unit (Expansion I/O Unit, Analog I/O Unit, or CompoBus/S I/O Link Unit). Up to three Expansion Units can be connected to a CPU Unit.

CPU and Expansion I/O Unit Configuration

The following table describes the CPM1A CPUs. All outputs are relay outputs.

The following table describes the six CPM1A CPUs. All outputs are relay outputs.

Number of I/O terminals	Inputs	Outputs	Power supply	Model number
10	6 points	4 points	AC	CPM1A-10CDR-A
			DC	CPM1A-10CDR-D
20	12 points	8 points	AC	CPM1A-20CDR-A
			DC	CPM1A-20CDR-D
30	18 points	12 points	AC	CPM1A-30CDR-A
			DC	CPM1A-30CDR-D
40	24 points	16 points	AC	CPM1A-40CDR-A
			DC	CPM1A-40CDR-D

2. SYSTEM CONFIGURATION

The following table describes the CPM2A CPUs.

Number of I/O terminals	Inputs	Outputs	Power supply	Model number
30	18 points	12 points	AC	CPM2A-30CDR-A
			DC	CPM2A-30CDR-D
			DC	CPM2A-30CDT-D
			DC	CPM2A-30CDT1-D
40	24 points	16 points	AC	CPM2A-40CDR-A
			DC	CPM2A-40CDR-D
			DC	CPM2A-40CDT-D
			DC	CPM2A-40CDT1-D
60	36 points	24 points	AC	CPM2A-60CDR-A
			DC	CPM2A-60CDR-D
			DC	CPM2A-60CDT-D
			DC	CPM2A-60CDT1-D

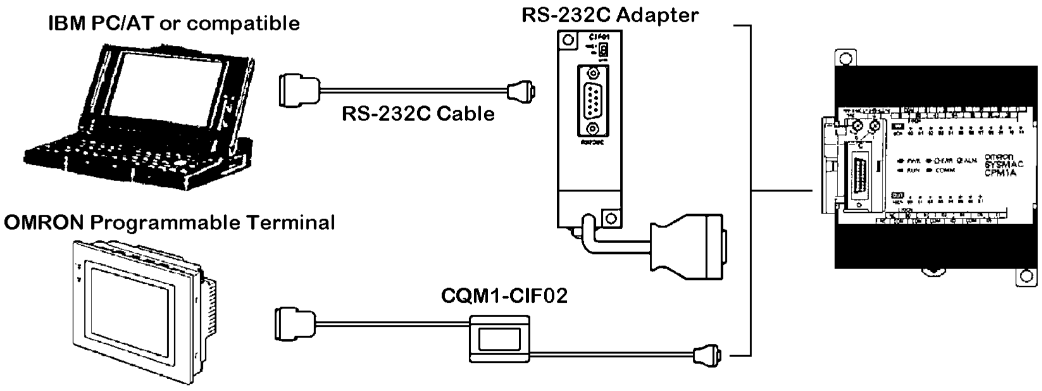
The following table describes the expansion unit shared by CPM1A and CPM2A.

Number of I/O terminals	Inputs	Outputs	Power supply	Model number
20	12 points	8 points	DC	CPM1A-EDR
				CPM1A-EDT
				CPM1A-EDT1
8	8 points	0 points	DC	CPM1A-8ED
8	0 point	8 points	DC	CPM1A-8ER
				CPM1A-8ET
				CPM1A-8ET1
-	2 Analog	1 Analog	DC	CPM1A-MAD01
16	8 points	8 points	DC	CPM1A-SRT21

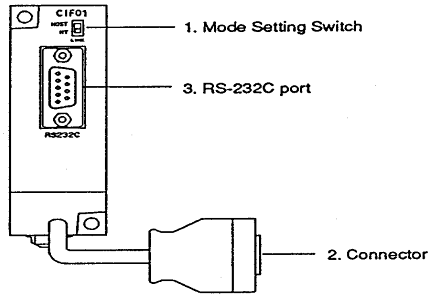
□ Host Link Communications (CPM1A)

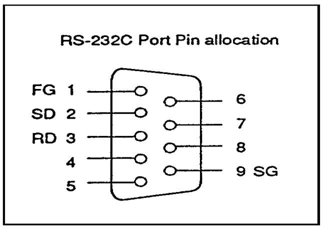
Host Link communications allows up to 32 OMRON PCs to be controlled from a single host computer. The computer to PC connections can be established using either RS-232C or RS-422 Adapters.

1:1 Communications to Computer
The following diagram shows t, the possible methods for a 1 to 1 connection between a CPM1A, OMRON Programmable Terminal (an operator interface device) and an IBM PC/AT or compatible computer.



Communications Adapter Components
<u>RS-232C Adapter</u>





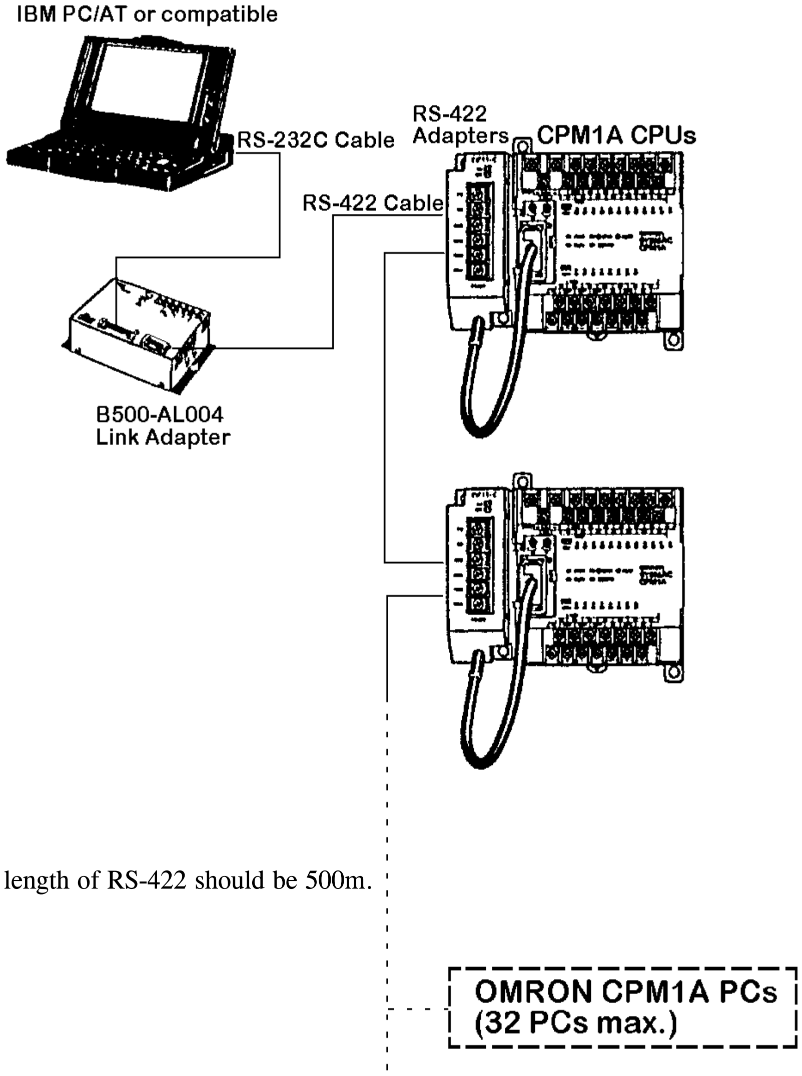
1. Mode Setting Switch
   Set this switch to “HOST” when using a Host Link system to connect to a personal computer. (Set this switch to “NT” when connecting to a programmable Terminal via a 1:1 NT link).

2. Connector
   Connector to the CPU’s Peripheral Port.

3. RS-232C Port
   Connects to the RS-232C cable from the other devices such as a personal computer, Peripheral Device, or Programmable Terminal.

2. SYSTEM CONFIGURATION

Multi-drop Communications to Computer
The following diagram shows how to connect up to 32 CPUs (CPM1A only) to an IBM PC/AT or compatible computer.



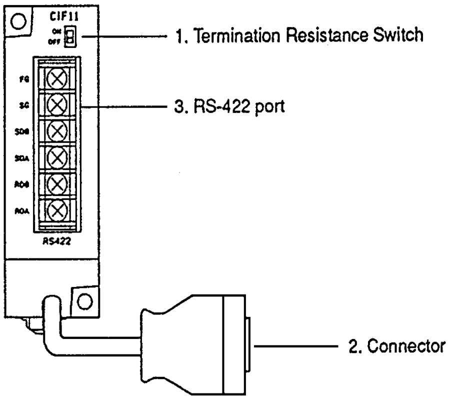
The maximum cable length of RS-422 should be 500m.

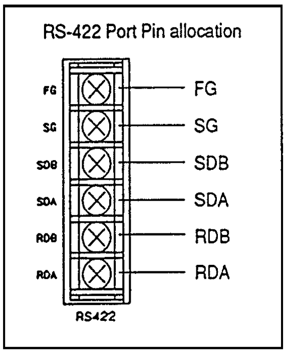
Adapters and Cables
The following table lists some of the Adapters and Cables used in Host Link communications.

Name	Usage	Model number
RS-232C Adapter	Converts to peripheral port-level communications	CPM1-CIF01
RS-422 Adapter		CPM1-CIF11
Connecting Cables	Used to connect IBM PC/AT or compatible computers. (Cable length:3.3 m)	CQM1-CIF02
Link Adapter	Converts between the RS-232C and RS-422	B500-AL004

	formats.	

RS-422 Adapter





1. Termination Resistance Switch
   Set the Link Adapter and RS-422 Adapter at each end of the main line to "ON" when using the termination resistance switch.

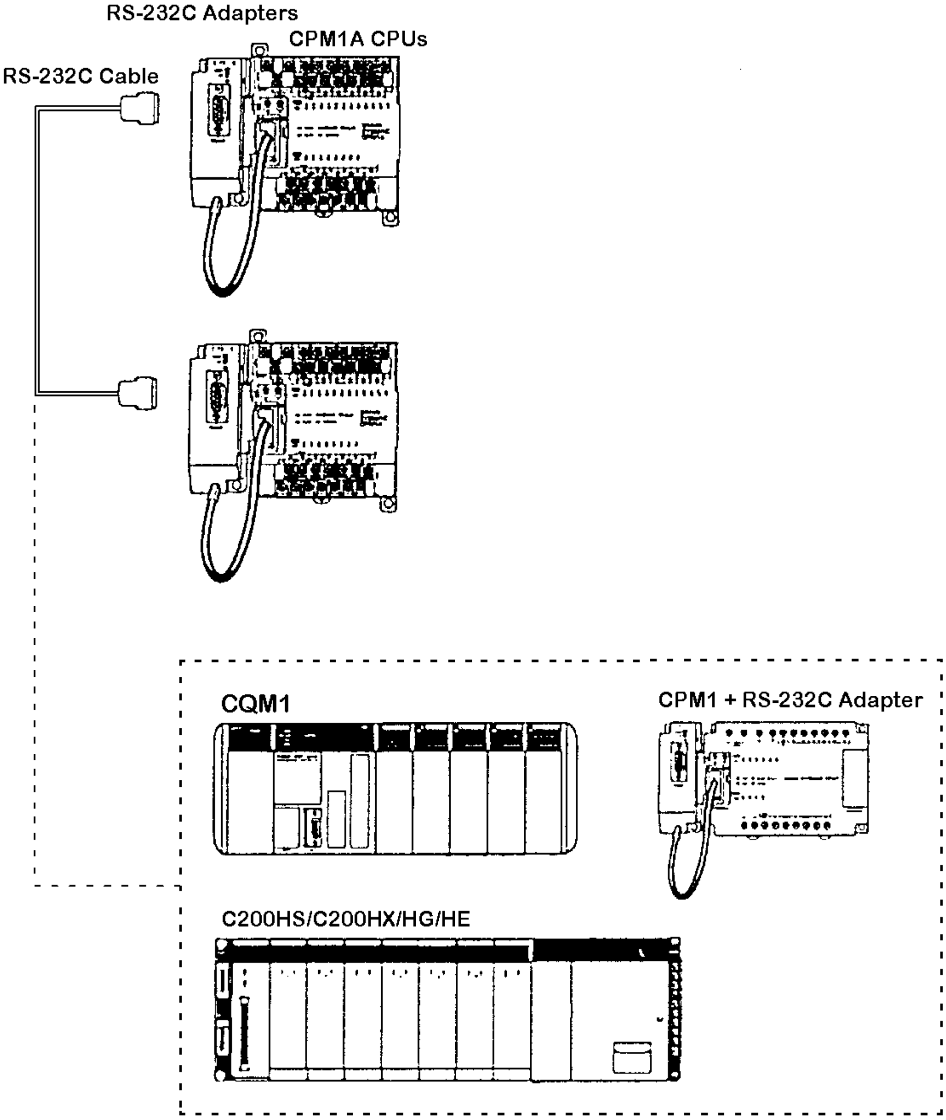
2. Connector
   Connects to the CPU's Peripheral Port.

3. RS-422 Port
   Connects to the Host Link network.

2. SYSTEM CONFIGURATION

• 1-to-1 PC Communication Links

A data link can be created with the LR data area in another CPM1A, CPM1, CQM1, SRM1 or C200HS PC or C200HX/HE/HG PC. An RS-232C Adapter is used to make the 1-to-1 connection.

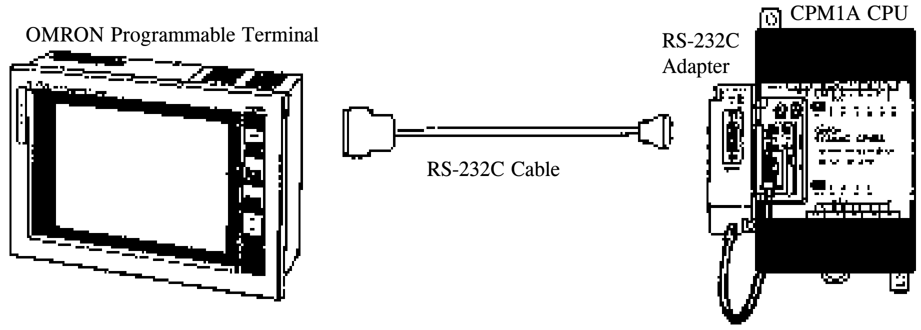


Name	Usage	Model number
RS-232C Adapter	Converts to peripheral port format.	CPM1-CIF01

2. SYSTEM CONFIGURATION

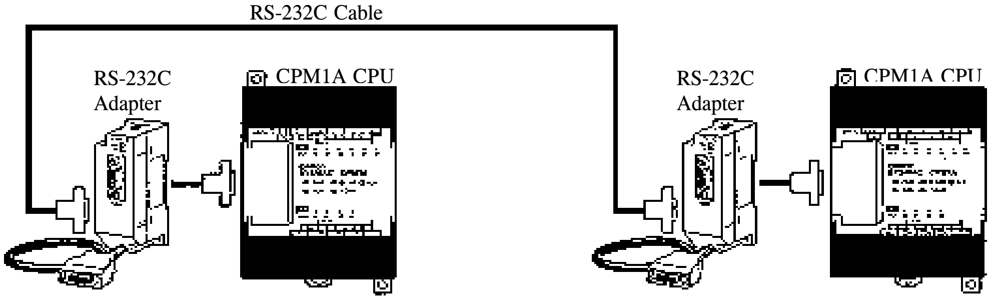
• NT Link Communication

NT Link is a high speed interface between the CPM1A and a Programmable Terminal. The RS232C adapter needs to be used for this connection, with the mode switch on the adapter set to “NT Link”.

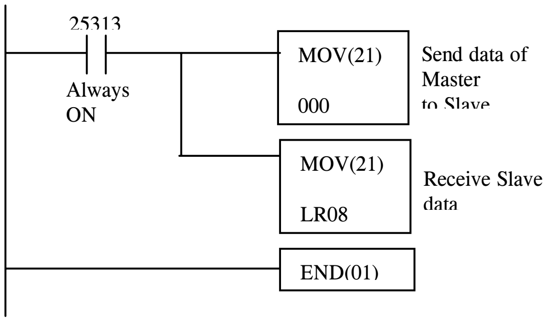


Name	Usage	Model number
RS-232C Adapter	Converts to peripheral port format.	CPM1-CIF01

• Programming Example of 1:1 PLC Link Between 2 CPM1A Units

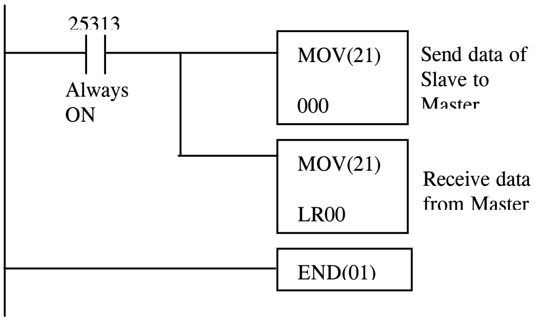


MASTER PROGRAM



Note: Set DM6650 = 3000 and
must set interface to NT

SLAVE PROGRAM



Note: Set DM6650 = 3000 and
must set interface to NT

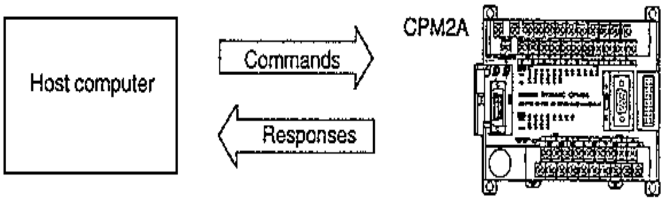
2. SYSTEM CONFIGURATION

❑ CPM2A Communication Functions

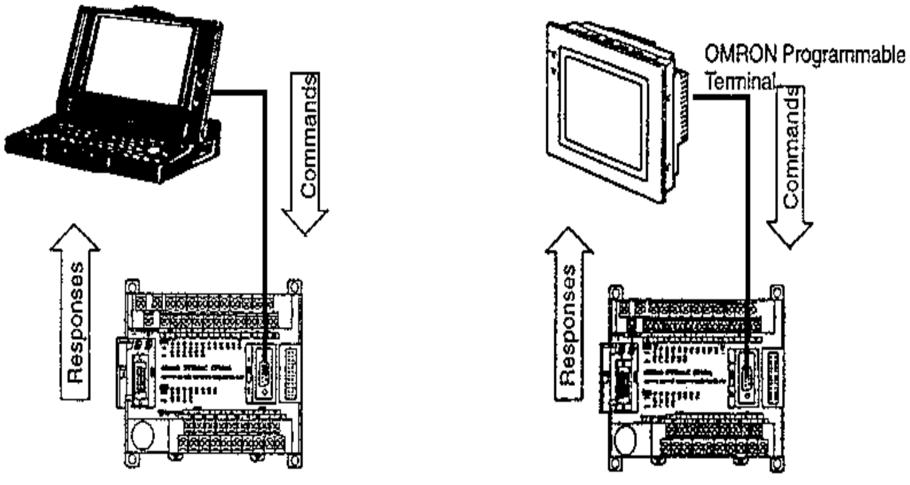
This section describes how to use CPM2A communications functions. The following types of communications are being describe: Host Link, No-Protocol, NT Link, or one-to-one link communications.

• Host Link Communications

Host Link communication area a conversational-type communication protocol, in which the PLC sends responses to commands issued from a host computer and can be used to read or write data in PLC's data areas and control some PLC operations. Host Link communications can be used through the peripheral port or the CPM2A's RS-232C port.



1-to-1 Communications



CPM2A RS-232C port connection (See note.)

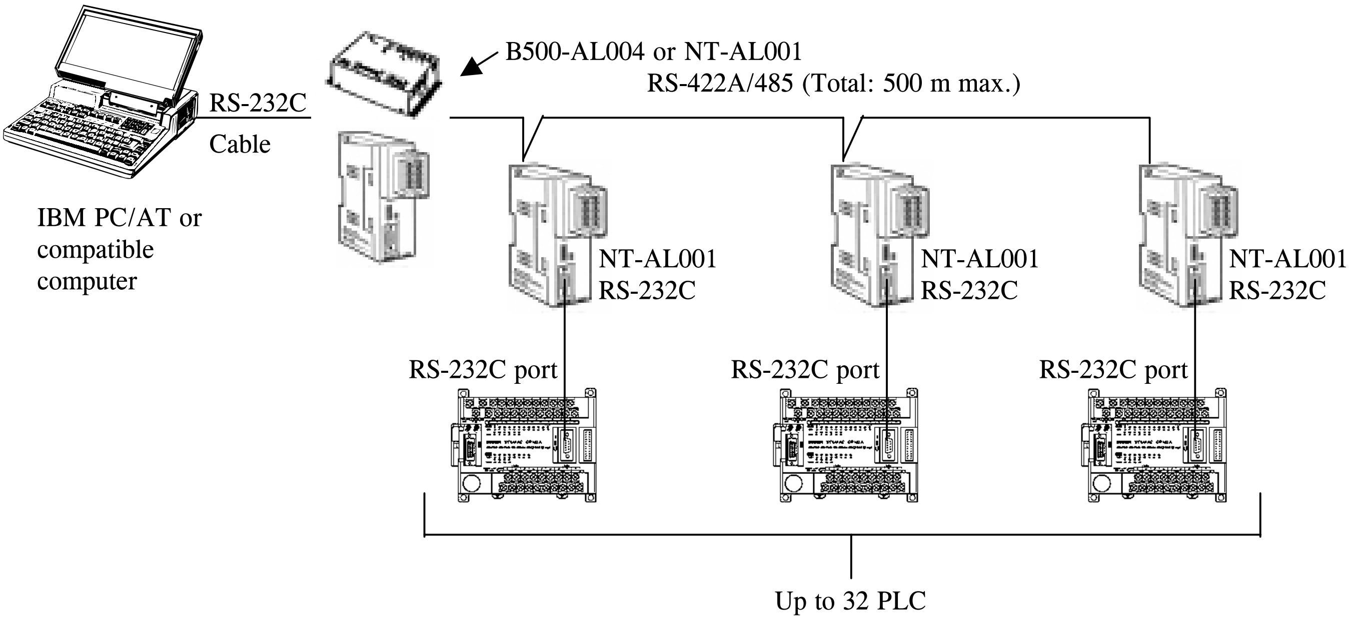
CPM2A RS-232C port connection (See note.)

Note When connecting to the peripheral port, an RS-232C Adapter or computer connection cable (CQM1-CIF01, CQM1-CIF02) is necessary.

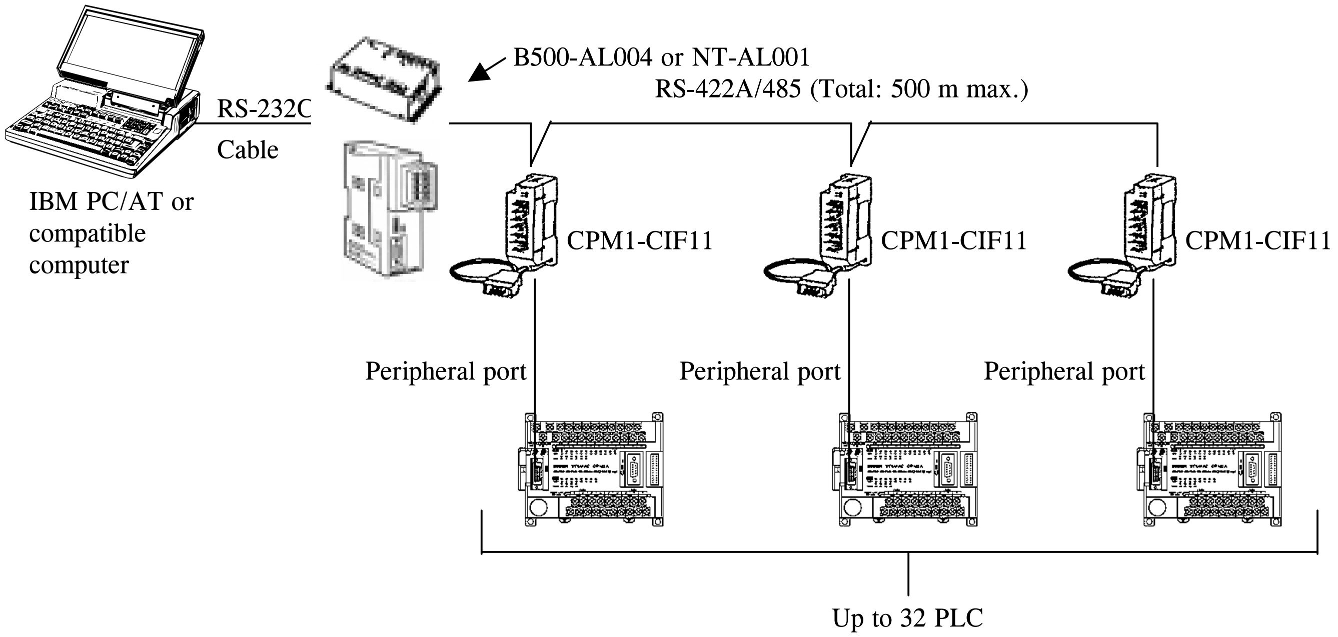
2. SYSTEM CONFIGURATION

Multi-drop Communications to Computer
Up to 32 Omron PLCs, including the OMRON PLCs, can be controlled from a host computer. The following diagram show separate RS-232C port and Peripheral Port configurations, but both port can be used simultaneously.

RS-232C Port Connection



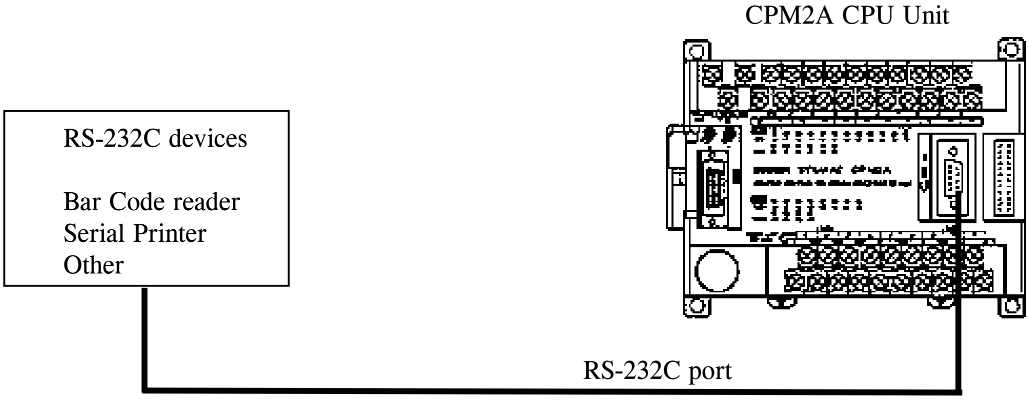
Peripheral Port Connection

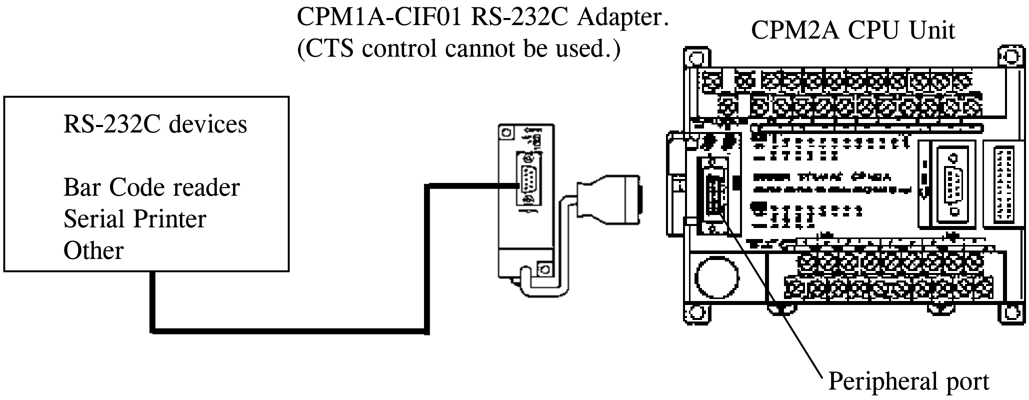


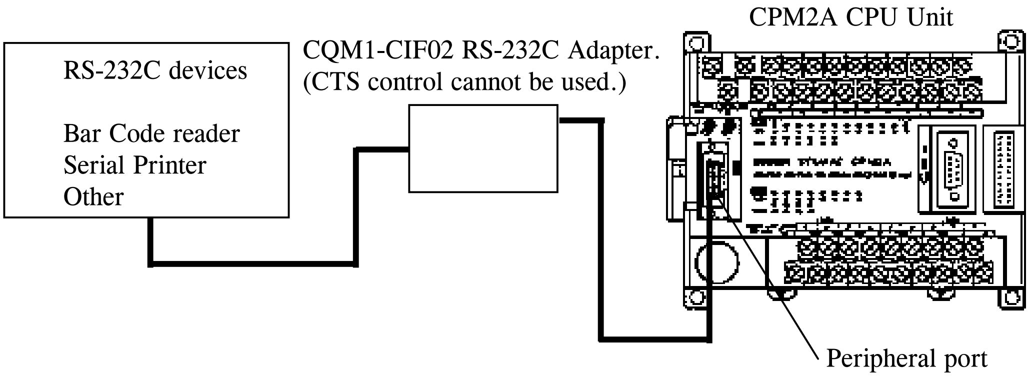
The NT-AL001 must be supplied externally with 5 VDC.

- **No-Protocol Communication (RS-232C Communication)**

When no-protocol communications are used, data can be exchanged with serial devices such as bar code readers and serial printers using TXD(48) and RXD(47). No-protocol communications can be used with either an RS-232C port or peripheral port.





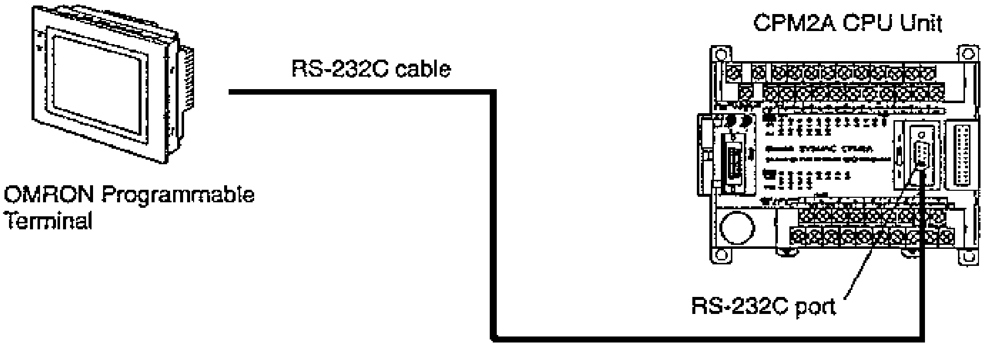


Note When using the No-protocol communication, it is necessary to setup the communication setting in the DM area. For detail, please refer to Appendix behind.

2. SYSTEM CONFIGURATION

• NT Link Communication

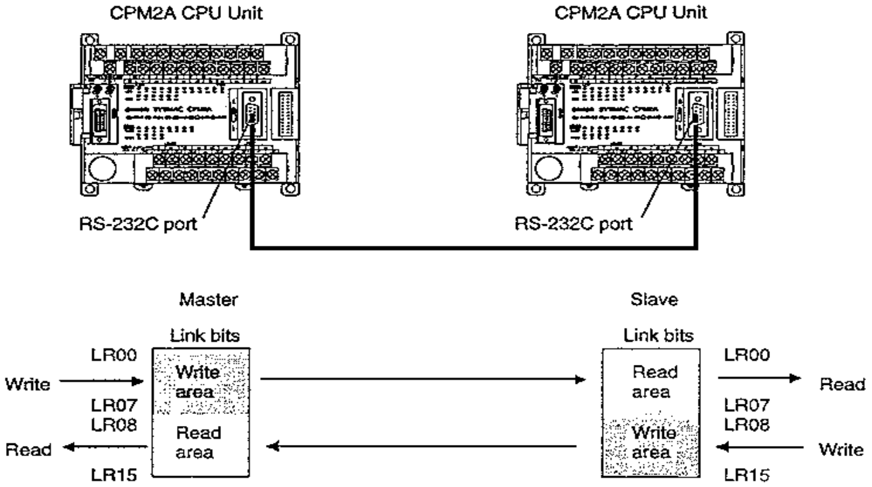
The NT Link allows a CPM2A PLC to be connected directly to an OMRON Programmable Terminal. There is no need for a communications program on the PLC. The NT Link can be used with a RS-232C port.



• One-to-one Link communications

A one-to-one data link of up to 256 bits (LR 00000 to LR 01515) can be created with the data area of another CPM2A, CQM1, CPM1A, SRM1(-V2), or a C200HX/HG/HE PLC, where one serves as the MAster, that other as Slave. There is no need for a communications program on the PLC.

The one-to-one link can be used with an RS-232C port.



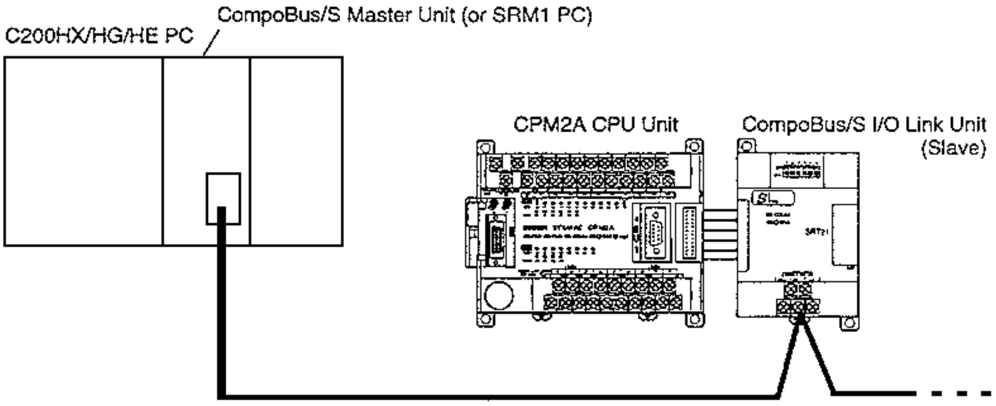
The link relay area on the CPM2A PLCs is only 16 words, LR00 to LR15. When performing a one-to-one link with a CPM2A PLC and CQM1, C200HS or C200HW/HG/HE use the corresponding 16 words, LR00 to LR15 on the CQM1, C200HS or C200HW/HG/HE. A one-to-one link with a CPM2A PLC cannot be performed using LR16 to LR63.

2. SYSTEM CONFIGURATION

• CompoBus /S I/O Link Connections

A CompoBus/S I/O Link can be used to create an I/O link (remote I/O) of 8 input points and 8 output points with a CompoBus/S Master or Unit SRM1 PLC. The connection is made through a CompoBus/S I/O Link Unit.

From the standpoint of the CPM2A CPU unit, the area allocated to the CompoBus/S I/O Link Unit can be treated just like the area allocated to an Expansion I/O Unit. The difference is that the bits are not actual I/O points, but I/O bits in the Master Unit.



Cables
Use special flat cable or VCTF cable to connect the nodes in the CompoBus/S I/O link. (Special flat cables and VCTF cables cannot be combined in the same system.)

Name	Model Number	Specifications
Flat Cable	XB1T-W10	4-conductor flat cable, 0.75 mm^2
VCTF Cable	---	2-conductor VCTF, 0.75 x 20

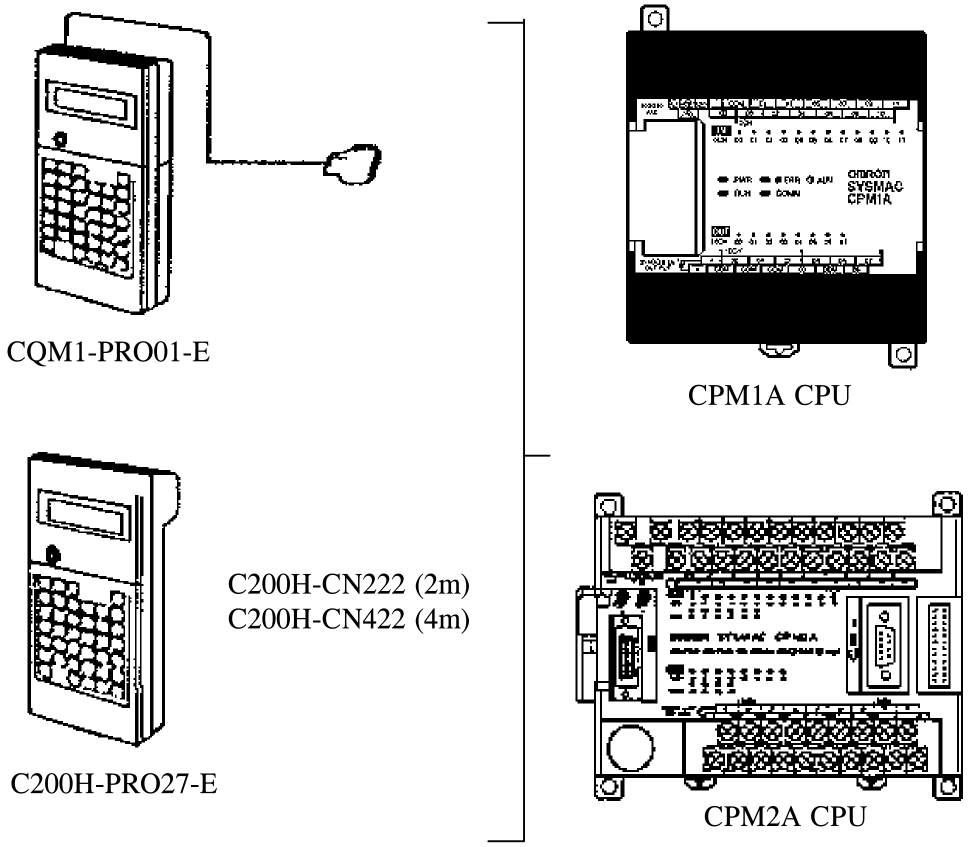
2. SYSTEM CONFIGURATION

• Peripheral Device Connections

CPM1A and CPM2A programming can be created or edited with a Programming Console or a personal computer running SYSWIN.

• Programming Console

A CQM1-PR001-E or C200H-PRO27-E Programming Console can be connected to the CPM1 and CPM2A, as shown in the following diagram.

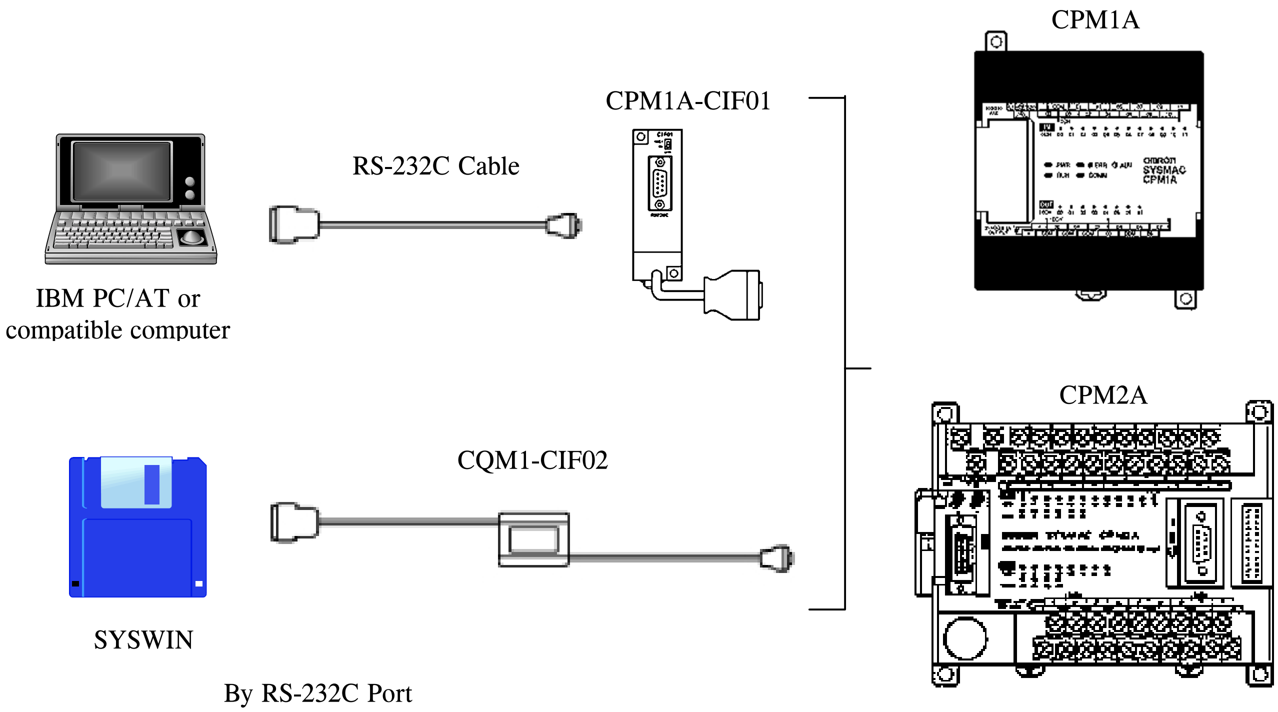


Name	Model number
CQM1-series Programming Console (The Connecting Cable is attached.)	CQM1-PRO01-E
C200H-series Programming Console (Cable C200H-CN222 needed for connection.)	C200H-PRO27-E

2. SYSTEM CONFIGURATION

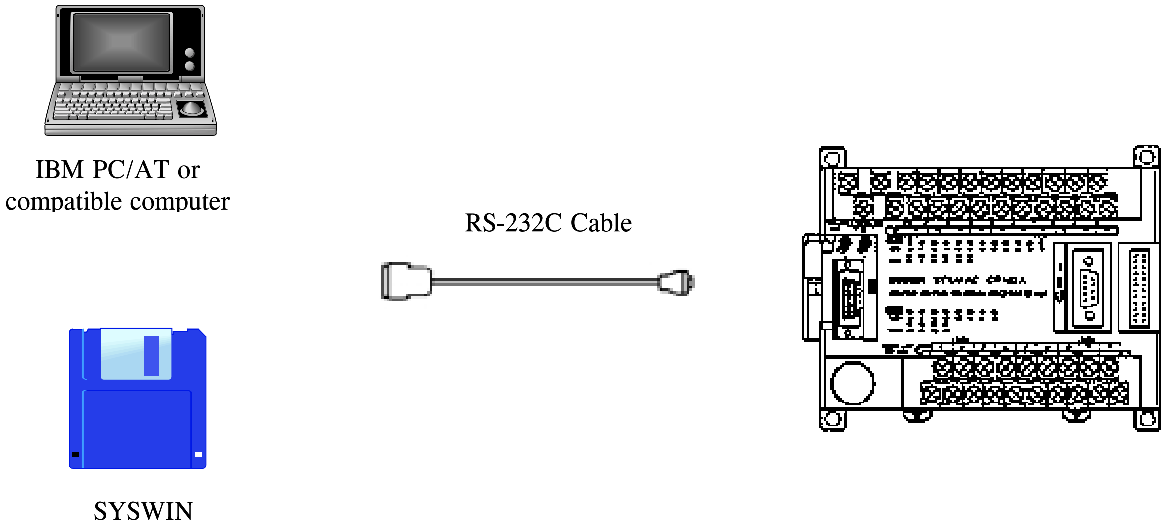
• SYSMAC Window-based Support Software

An IBM PC/AT compatible personal computer running SYSWIN can be connected to the CPM1A as shown in he following diagram.



By RS-232C Port

Connect to the CPU Units RS-232C port with an XW2Z-[]00S or XW2Z-[]00S-V Connecting Cable or Hostlink Cable Self-made.



Note The XW2Z- 00S cables have a D-Sub 25-pin connector and the XW2Z- 00S-V cables have a D-Sub 9-pin connector.

2. SYSTEM CONFIGURATION

Name	Usage	Model number
RS-232C Adapter	Converts to Peripheral Port format.	CPM1-CIF01
		CQM1-CIF01
Connecting Cable	Used to connect IBM PC/AT or compatible computers. (Length: 3.3 m)	CQM1-CIF02
RS232 Cable	From RS-232C port to computer. (CPM2A)	XW2Z- 00S XW2Z- 00S-V
SYSWIN (Windows Version)	For IBM PC/AT or compatible computers (3.5” disks, 2HD)	SYSWIN-E-V3.3

Communication Settings (For CPM2A)

Set the CPU Unit’s Communication Switch to ON when using the standard communications settings.

Set the switch to OFF to use the setting in the PLC Setup Area.

Setting	Peripheral port	RS-232C port	Setting
Mode	DM 6650 bits 12 to 15	DM 6645 bits 12 to 15	0: Host Link
Baud Rate	DM 6651 bits 00 to 07	DM 6646 bits 00 to 07	00: 1,200 bps 01: 2,400 bps 02: 4,800 bps 03: 9,600 bps 04: 19,200 bps 05: 38,400 bps
Frame format	DM 6651 bits 08 to 15	DM 6646 bits 08 to 15	00 (See note)

Note The standard settings are: Host Link, 9600 bps, 7 data bits, 1 start bit, 2 stop bits, and even parity.

2. SYSTEM CONFIGURATION

- **Training Kit Configuration**

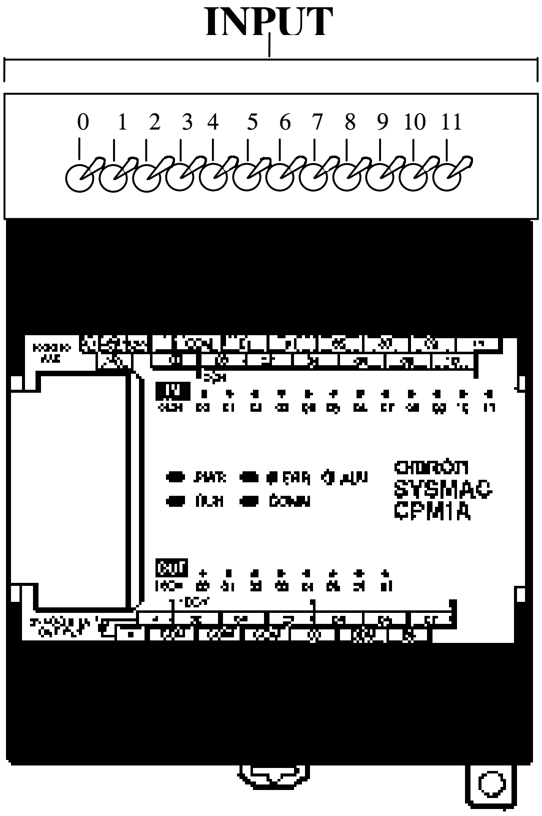


**Fig. 2.1:
The CPM1A
Training Kit**

In Fig. 2.1, we see the overall outlook of the CPM1A training kit. It consists of : 12 Input switches connected to the input terminals of the CPM1A CPU.

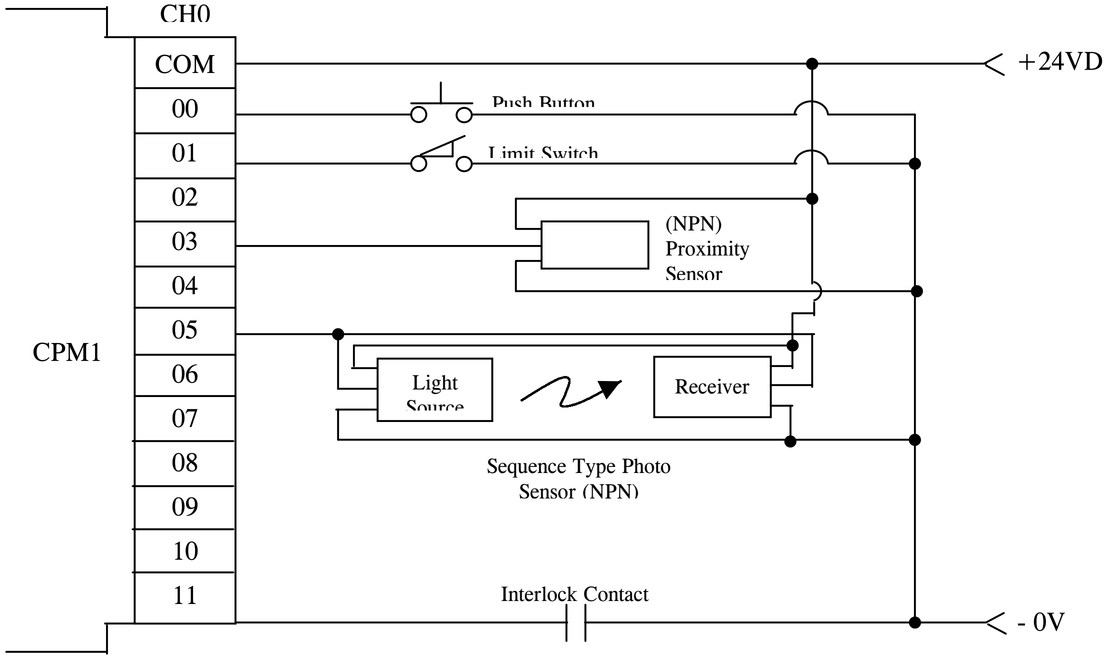
On the top surface of the CPM1A, we could find the peripheral port, 4 LED indicators and the expansion port. The peripheral port is meant for the programming console or the RS232C peripheral interface. The four LED indicators are RUN, POWER, ERROR and Coummnication.

The L and N terminals is connected to power source to supply power to the CPM1A.

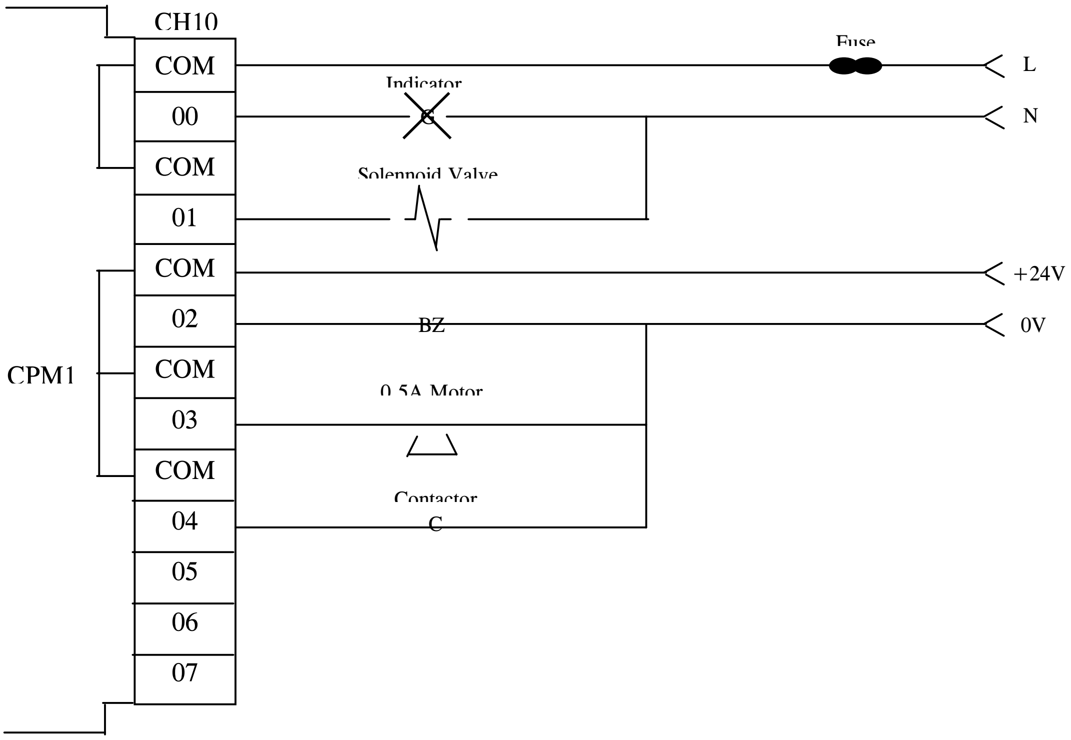
In the CPM1A, channel 0 is assigned as input and channel 10 is assigned as output. Each channel consists of 16 bits starting from bit 00 to bit 15. For more information about the other memory addresses, refer to Appendix C of this manual. You could find the internal Auxiliary, Data Memory, Link Relays, Timers/Counters & Holding Relays areas, etc. For complete understanding, you should read through this manual.

- **PLC Electrical Wiring Example**

INPUT WIRING (DC24V)



OUTPUT WIRING (Contact)



3. FEATURES & FUNCTIONS

❑ CPM1A General

- The CPM1A is a compact PLC with 10, 20, 30 or 40 I/O terminals built into the CPU.

- An Expansion I/O Unit can be connected to the 30 and 40 I/O–point CPU to add an extra I/O point.

- Flash memory provides memory backup without a battery.

- The CPM1A–10CDR– PCs can handle 2 interrupt inputs; the CPM1A–20CDR– , CPM1A–30CDR– and CPM1A–40CDR– PCs can handle 4 interrupt inputs. In addition to normal input interrupts, the CPM1A has a counter mode that counts high-speed input signals and triggers interrupts at fixed count multiples.

- Quick-response inputs can detect input signals with a pulse width as short as 0.2 ms regardless of their timing during the PLC cycle. Quick-response inputs and interrupt inputs use the same input terminals.

- CPM1A PLCs have a high-speed interval timer which can be set from 0.5 ms to 319,968 ms. The timer can be set to trigger a single interrupt (one-shot mode) or repeat scheduled interrupts (scheduled interrupt mode).

- CPM1A PLCs have a high-speed counter that can be used in incremental mode or up/down mode. The high-speed counter can be combined with input interrupts to perform target value control or zone comparison control that isn't affected by the PLC's cycle time.

- The CPM1A PLCs have 2 analog volume controls that can be used to make manual analog settings manually. These can be used to externally set timing or counter values etc.

- The CPM1A PLCs are compatible with the Host Link, which allows communications with personal computers or Programmable Terminals.

- An RS–232C Adapter is used for 1–to–1 communications and an RS–422 Adapter is used for 1–to–n communications.

- A data link can be created with the LR data area in another CPM1A, CPM1, CQM1, or C200HS PLC. An RS–232C Adapter is used to make the 1–to–1 connection.

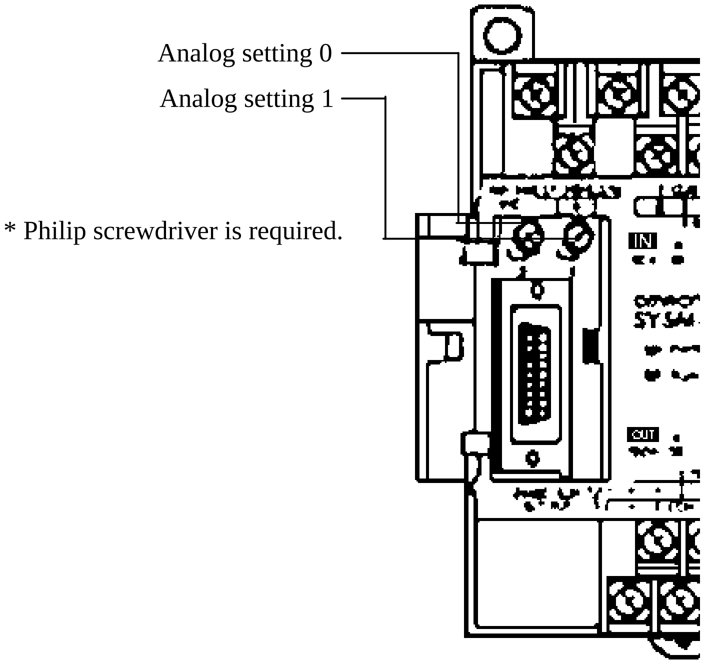
- High–speed Man–Machine Interface operations can be achieved by connecting the CPM1A to the Programmable Terminal through the NT Link Interface. An RS–232 Adapter is used for this connection.

- The CPM1A uses the same Programming Consoles and Syswin Programming Software as other Omron C-Series PLCs.

3. FEATURES & FUNCTIONS

• Analog Setting Function

CPM1A PLCs have 2 analog volume controls that can used to make analog timer and counter settings manually. When one of the volume controls is turned, the content of the corresponding Internal Relay word is set automatically between 0 and 200 (BCD).



The following table shows which SR bits are allocated to the I/O terminals on the CPM1A's CPU and Expansion I/O Unit.

Control	Corresponding IR word	Setting range (BCD)
Analog volume control 0	SR 250	0000 to 0200
Analog volume control 1	SR 251	

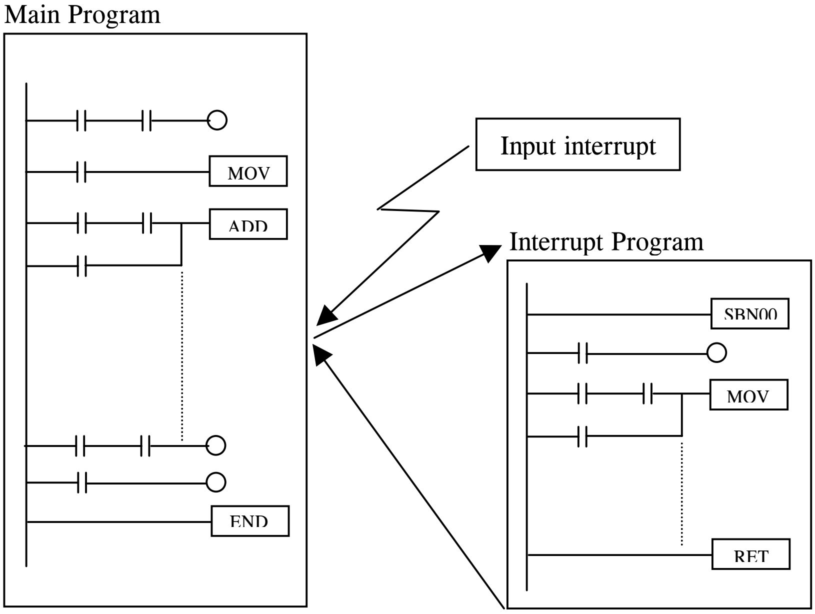
3. FEATURES & FUNCTIONS

• Inputs Interrupts

There are two input interrupts in the CPM1A 10–point I/O CPU and four in the 20–, 30– and 40–point I/O CPUs. Input interrupts are available in two modes: input interrupt mode and counter mode.

1. When an interrupt occurs in Input Interrupt Mode, the main program shuts down irrelevant of the cycle time, and the interrupt program is executed immediately.
2. In Counter Mode, external input signals are counted at high speed (up to 1 kHz) and an interrupt is generated each time the count reaches the set value. When an interrupt occurs, the main program shuts down and the interrupt program is executed. The set value can be set from 0 to 65,535.

The following diagram shows the program execution when an interrupt occurs.



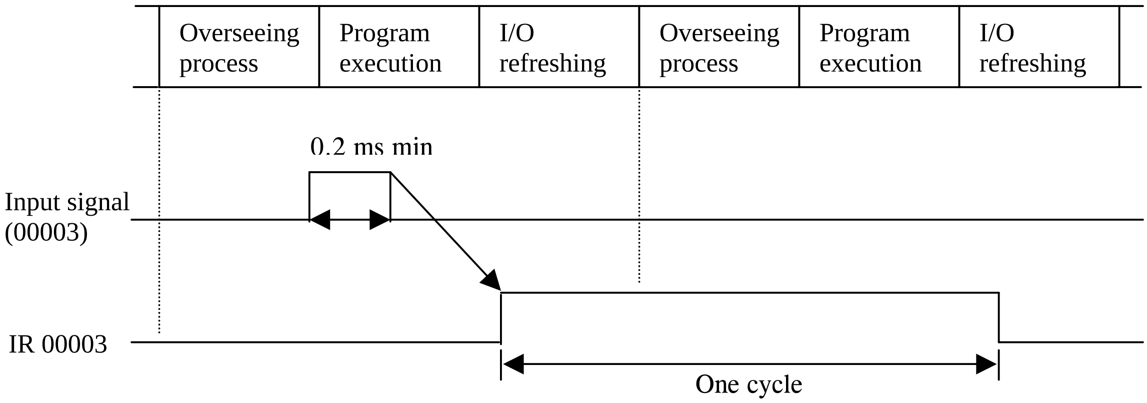
PC model	Input bits	Response time
CPM1A–10CDR–	IR 00003 to IR 00004	0.3 ms (1 kHz in Counter Mode)
CPM1A–20CDR– CPM1A–30CDR– CPM1A–40CDR–	IR 00003 to IR 00006	

3. FEATURES & FUNCTIONS

• Quick-response Inputs Functions

The CPM1A–10CDR–[] PLCs have 2 quick-response input terminals and the CPM1A–20CDR–[]/30CDR–[]/40CDR–[] PLCs have 4 quick-response input terminals.
(The same terminals are used for quick-response inputs and interrupt inputs.)

Quick-response inputs have an internal buffer, so input signals shorter than one cycle can be detected.



The following table shows the input bits use for Quick-response input function.

PC model	Input bits	Minimum input pulse width
CPM1A–10CDR–[]	IR 00003 to IR 00004	0.2 ms
CPM1A–20CDR–[] CPM1A–30CDR–[] CPM1A–40CDR–[]	IR 00003 to IR 00006	

• Interval Timer Function

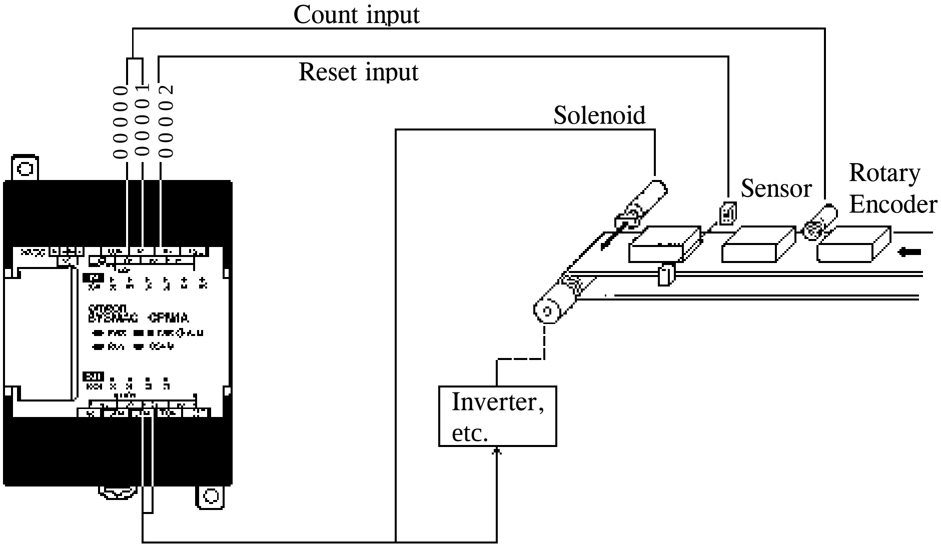
CPM1A are equipped with an interval timer which can be set from 0.5 ms to 319,968 ms in units of 0.1 ms. The timer can be set to trigger a single interrupt (one–shot mode) or repeat scheduled interrupts (scheduled interrupt mode).

Mode	Function
One–shot	Generates a single interrupt the first time that the timer times out.
Scheduled interrupt	Generates an interrupt each time that the timer times out.

3. FEATURES & FUNCTIONS

• High-speed Counter

CPM1A have a high–speed counter function that can be used in increment mode or up/down mode. Using this function together with the input interrupts enables target value control or zone comparison control irrelevant of the PLC's cycle time.



Mode		Incremental	Up/Down
Input no.	00000	Count input	A–phase input
	00001	- - -	B–phase input
	00002	Reset input	Z–phase input
Input method		single–phase input	Phase–difference, 4x inputs
Count frequency		5.0 kHz	2.5 kHz
Count range		0 to 65535	–32767 to 32767
Control method	Target value control	Up to 16 target values and interrupt subroutine numbers can be registered.	
	Zone comparison control	Up to 8 sets of upper and lower limit values, and interrupt subroutine numbers can be registered.	

Note In increment mode, this input (00001) can be used as an regular input.

3. FEATURES & FUNCTIONS

❑ CPM2A General

- The CMP2A is a compact PLC with 30, 40 or 60 I/O terminals built-in.

- Expansion Unit can be connected to the 30, 40 or 60 I/O-point CPU to add extra I/O points. Maximum of 3 expansion unit can be connected for a maximum I/O configuration of 120 points.

- CPM2A PLCs have built-in RS-232C port and Peripheral Port.

- CPM2A PLCs have a high speed counter that can be used in Incremental mode, Pulse + Direction input mode and Up/Down pulse input mode which allows a response frequency of up to 20 kHz. The high speed counter can be also be used in Differential phase input mode with a maximum response frequency of 5kHz.

- CMP2A PLCs have 4 points for interrupt inputs function.

- CMP2A PLCs have 2 analog volume controls that can be used to make analog setting manually, for instance, externally set control period or counter values etc.

- CPM2A PLCs have 2 pulse outputs.

- CPM2A-   CDT-D and CPM2A-   CDT1-D have Synchronized Pulse control.

- **Interrupt Functions**

The CPM2A provides the following kinds of interrupt processing.

Interrupt Inputs
Interrupt programs are executed when inputs to the CPU Unit's built-in input points (00003 to 00006) are turned from OFF to ON. Interrupt subroutine numbers 000 to 003 are allocated to input pints 00003 to 00006.

Interval Timer Interrupts
Interval timer interrupt programs are executed with the precision of 0.1ms. Interrupt subroutine numbers 000 to 049 are allocated by instructions.

Count-up interrupts Using Interrupts Using Interrupt Inputs (Counter Mode)
Input signals to the CPU Unit's built-in input points (0003 to 00006 are counted at high speed (2kHz), and the normal program is stopped and an interrupt program is executed. Interrupt subroutine numbers 000 to 003 are allocated to input points 00003 to 00006.

Count-check Interrupts Using the High-speed Counter
Pulse inputs to the CPU Unit's built-in input points (00003 to 00006) are counted at high speed (20 kHz/5 kHz), and an interrupt program is executed when the present values matches the target value or falls within a given range. Interrupt subroutine numbers 000 to 049 are allocated by instructions.

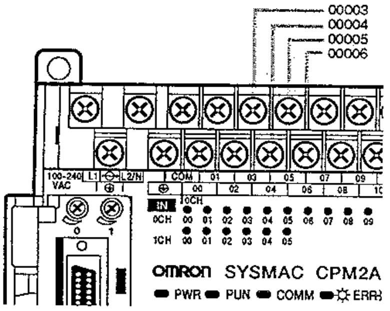
The order of priority for interrupts is as follows:

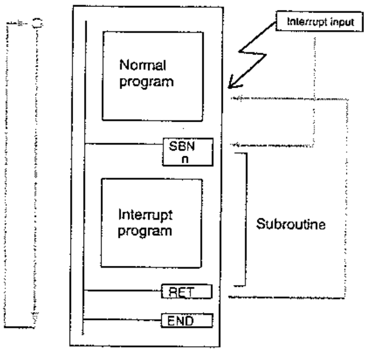
Interrupt inputs = Count-up interrupts > Interval timer interrupts = High-speed counter count check interrupts

If an interrupt with a higher priority is generated. During interrupt program execution, the interrupt that is currently being processed will be stopped and the new interrupt will be processed first. Then the original interrupt will resumed after the higher-priority interrupt processing has been completed.
If interrupts of the same priority are generated simultaneously, they will be processed in the following order:
Interrupt inputs → interrupt input 1 → Interrupt input 2 → Interrupt input 3 → (including count-up mode)
Interval timer interrupt → High-speed counter interrupt.

• Interrupt Inputs

By tuning the CPM2A CPU Unit’s built-in input points from OFF to ON, the normal program can be stopped and the interrupt program executed. The interrupt inputs are allocated to four points (00003 to 0006).

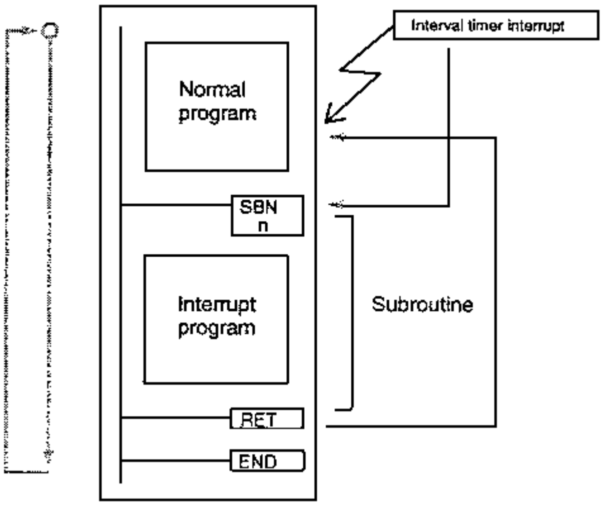




Input Number	Interrupt Number	Subroutine Number	Minimum input signal width	Interrupt response time
00003	0	000	0.05ms	0.3 ms (from when input turns ON until program execution)
00004	1	001		
00005	2	002		
00006	3	003		

• Interval Timer Interrupts

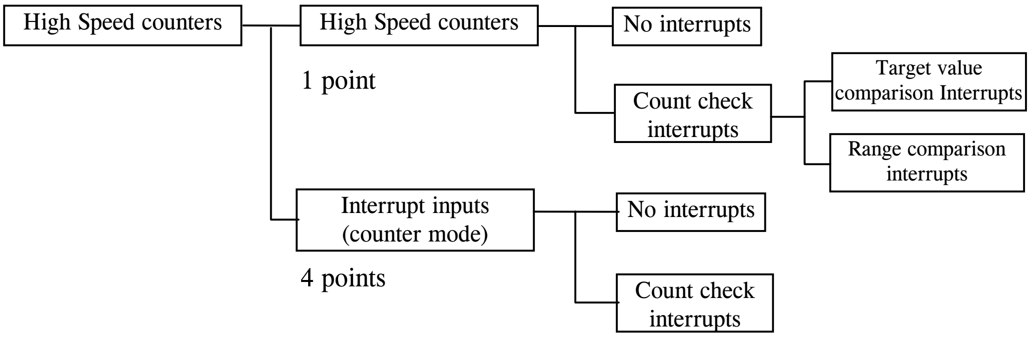
The CPM2A has one interval timer (precision: 0.1 ms) that can be set from 0.5 ms to 319,968 ms. There are two interrupt modes: the one-shot mode, in which a single interrupt is executed when the time is up, and the schedule-interrupt mode, in which interrupts are executed at regular intervals.

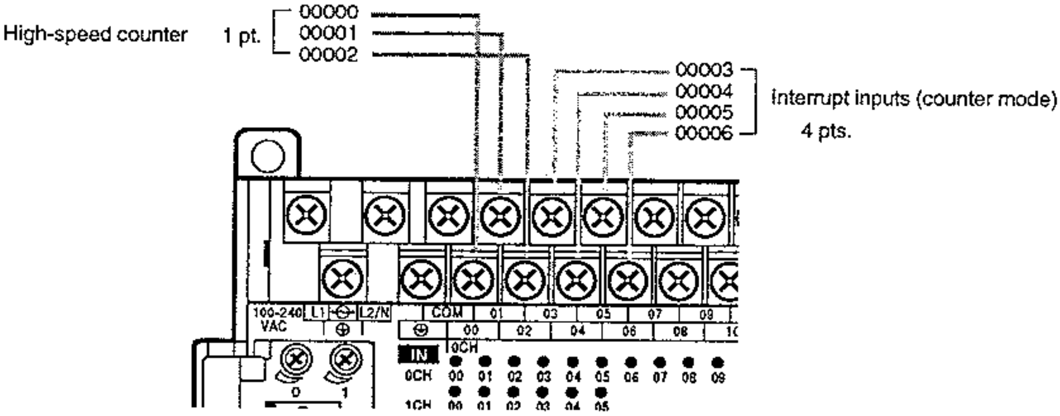


	One-shot mode	Schedule-Interrupt mode
Operation	Interrupt is executed once when time has elapsed.	Interrupt is executed at regular intervals.
Set time	0.5 to 316,968 ms (Unit:0.1 ms)	
Interrupt response time	0.3 ms (from when time has elapsed until execution of interrupt program)	

- **High Speed Counters**

The CPM2A CPU Unit has a total of five points for high-speed counters: one point for high-speed counter with a maximum response frequency of 20 kHz, and four points for interrupt inputs (counter mode).





The CPM2A provides both a built-in high-speed counter and built-in interrupt inputs.

• Interrupt Inputs (Counter Mode)

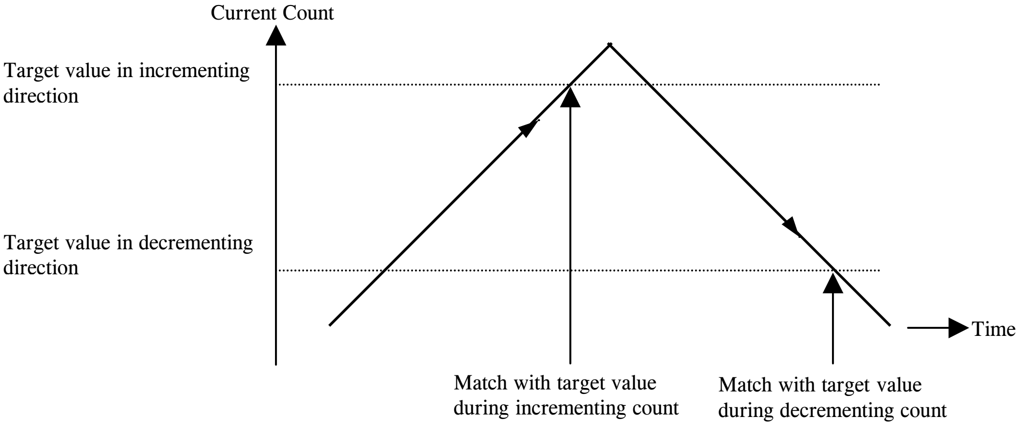
Interrupt inputs (counter mode) are counters based on inputs to the CPU Unit's built-in points 00003 to 00006. These counters have four points, and they can provide either an incrementing or decrementing count depending on the mode setting. Since this function utilizes interrupt inputs for counting, it is not possible to use the same inputs for other interrupt inputs.

Input Number (Note 1)	Response frequency	Input Mode (Count value)	Control Method
00003	2 kHz	Incrementing counter (0000 to FFFF) Decrementing counter (0000 to FFFF)	Count-up interrupts
00004			
00005			
00006			

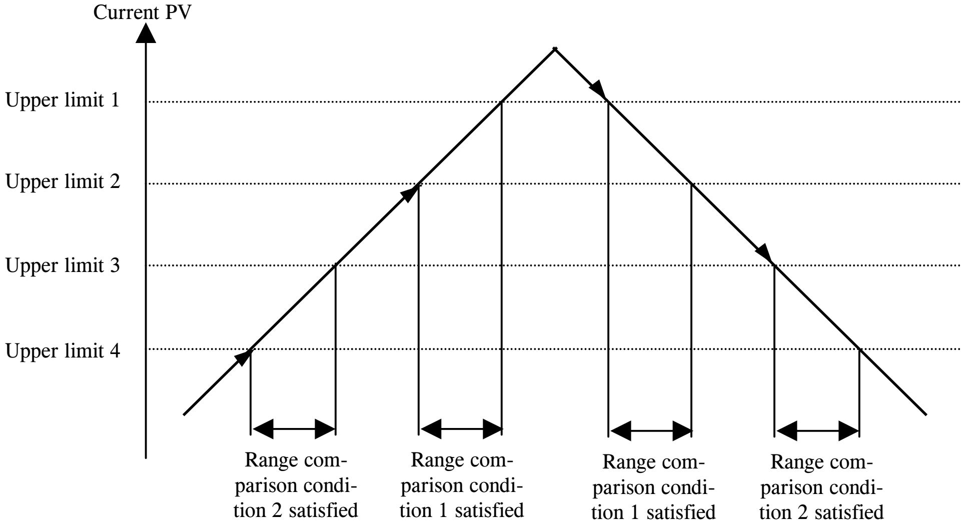
• Interrupt by High Speed Counter Inputs (Count-check Interrupts)

Target Value Comparison Interrupts

The current count is compared to each target value in the order that they are registered in the table. When the count is the same as the current target value, an interrupt subroutine is executed. Up to 16 target values and interrupt subroutines can be registered in the table in either the incrementing or decrementing direction.


Current Count
Target value in incrementing direction
Target value in decrementing direction
Time
Match with target value during incrementing count
Match with target value during decrementing count

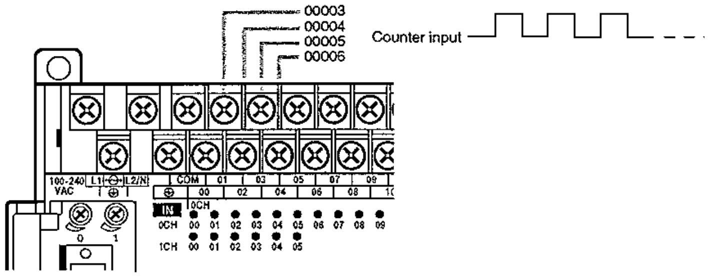
Range Comparison Interrupts
A range comparison table contains up to eight ranges which are each defined by a lower limit and an upper limit, as well as their corresponding subroutine numbers. The corresponding subroutine is called and executed when the current count (the counter PV) falls within a given range.



Count-up interrupts by Interrupt Inputs (Counter Mode)
An interrupt subroutine is executed each time the condition is satisfied that the counter PV equals the counter SV (in incrementing mode) or 0 (in decrementing mode).

Example of Input Interrupt In Counter Mode:

The four built-in interrupt inputs in the CPM2A's CPU unit be used in counter mode as inputs of up to 2 kHz. These inputs can be used as either incrementing counters of decrementing counters, triggering an interrupt (i.e., executing an interrupt subroutine) when the count matches the set value.



Input Number	Count	Interrupt number	Subroutine number	Response frequency
00003	0 to 65535 (0000 to FFFF)	0	000	2 kHz
00004		1	001	
00005		2	002	
00006		3	003	

3. FEATURES & FUNCTIONS

• High Speed Counter

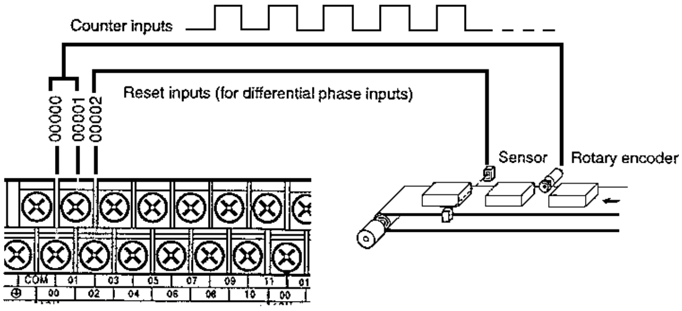
The built-in high-speed counter is a counter based on inputs to CPU Unit's built-in points 00000 to 00002. The high-speed counter itself has one-point, and it can provide either an incrementing/ decrementing or just an incrementing count depending on the mode setting.

Input No. (See Note).	Response frequency	Input mode (count value)	Control Method
00000 00001 00002	5 kHz	Differential phase input mode (-8388608 to 8388607)	Target value comparison interrupts Range comparison interrupts
	20 kHz	Pulse + direction input mode (-8388608 to 8388607) Up/ down pulse input mode (-8388608 to 8388607) Increment mode (0 to 16777215)	

Input points not used for counter inputs can be used as ordinary inputs.

Example of using High Speed Counter

The CPM2A's CPU Unit has one built-in channel for a high-speed counter that can count inputs at a maximum of 20 kHz. Using the conjunction with the interrupt function enables target value comparison control or range comparison control to be executed without deviating from the cycle time.



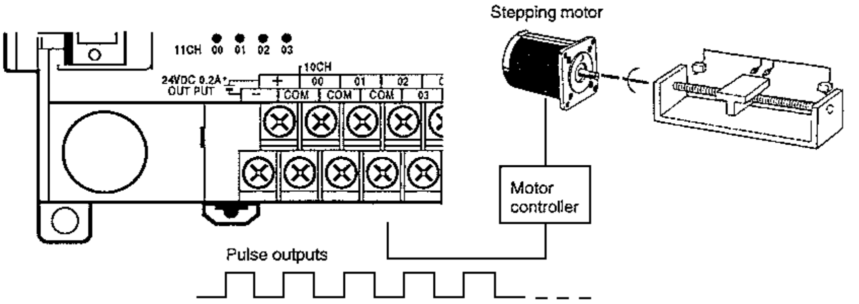
		Input Mode			
		Differential phase input mode	**Pulse + direction outputs**	**Up/ down pulse outputs**	**Increment mode**
Input number	**00000**	A-phase input	Count input	CW input	Count input
	00001	B-phase input	Direction input	CCW input	See note 1.
	00002	Z-phase input (Reset inputs)(See note 1.)			
Input method		Differential phase input (4x)	Phase inputs	Phase inputs	Phase inputs
Response frequency		5 kHz	20 kHz	20 kHz	20 kHz
Control value		-8388608 to 8388607			0 to 16777215
Counter PV storage destination (see note 2.)		Words SR 248 (rightmost digit) and SR249 (leftmost digit)			
Interrupts	**Target value comparison**	Up to 16 target values and interrupt subroutine number can be registered in either the incrementing or decrementing direction.			
	Range Comparison	Up to 8 ranges (with upper and lower limits) and subroutine numbers can be registered.			
Counter reset method		Z-phase signal + software reset: Counter is reset when IR 00002 turns ON while SR25200 is ON. Software reset: counter is reset when SR 25200 turns ON.			

Note: 1. Input points not used for counter inputs can be used as ordinary inputs.
2. When not used for counter PV storage destination, these words can be used as ordinary IR words.

- **Pulse Output Function**

The CPM2A has two pulse outputs. By means of a selection in the PLC setup, these outputs can be used as two single-pulse outputs without acceleration and deceleration, two variable duty ratio pulse outputs, or pulse outputs with trapezoidal acceleration/ deceleration, (one pulse + direction output and one up/down pulse output). The pulse output PV coordinate system can also be specified in the PLC setup as either relative or absolute.

In order to utilize pulse outputs, it is necessary to use a CPU Unit with transistor outputs (either a CPM2A-[][][][]T-[] or CPM2A-[][][][]T1-[]).



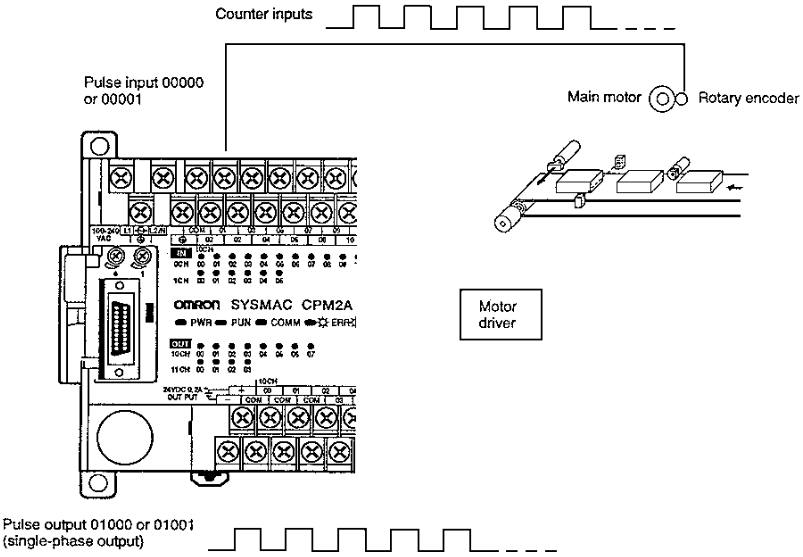
		Single-phase pulse outputs without accel/ decel	Variable duty ratio pulse outputs	Single-phase pulse outputs with trapezoidal acceleration and deceleration			
				Pulse + direction outputs		Up/ down pulse outputs	
Execution Instruction		PULS(65) and SPED(64)	PWM(-)	PULS(65) and ACC(-)			
Output number	**01000**	Pulse output 0 (See note 1)	Pulse output 0 (See note 1)	Pulse output 0	Pulse output	Pulse output 0	CW pulse output
	01001	Pulse output 1 (See note 1)	Pulse output 1 (See note 1)		Direction output		CCW pulse output
Output frequency range		0.01 to 10 kHz	0.1 to 999.9 Hz	0.01 to 10 kHz		0.01 to 10 kHz	
	Pitch	10 Hz	0.1 Hz	10 Hz		10 Hz	
Up.down frequency pitch		--	--	10 Hz (See note 2.)		10 Hz (See note2.)	
Start speed pitch		--	--	10 Hz		10 Hz	
Output mode		Continuous, independent	Continuous	Continuous		Continuous, independent	
	Numbers of pulses	1 to 16777215	--	± 1 to 16777215		± 1 to 16777215	
Duty ratio		50%	0 to 100%	50%		50%	
Control method	**Movement specification**	Yes	No	Yes		Yes	
	Accel/ Decele specification	No	No	Yes		Yes	
	Start Speed specification	No	No	Yes		Yes	
	Duty specification	No	Yes	No		No	

1. With single-phase pulse outputs, pulse outputs 0 and 1 can each be output independently.
2. Pulse outputs can be accelerated or decelerated in units of 10 Hz every 10 ms.

• Synchronized Pulse Control

By combining the CPM2A's high-speed counter function with the pulse output function, the output pulse frequency can be controlled as a specified multiple of the input pulse frequency.

A CPU Unit with transistor outputs (CPM2A-[][]CDT-D or CPM2A-[][]CDT1-S) is required in order to use synchronized pulse control.



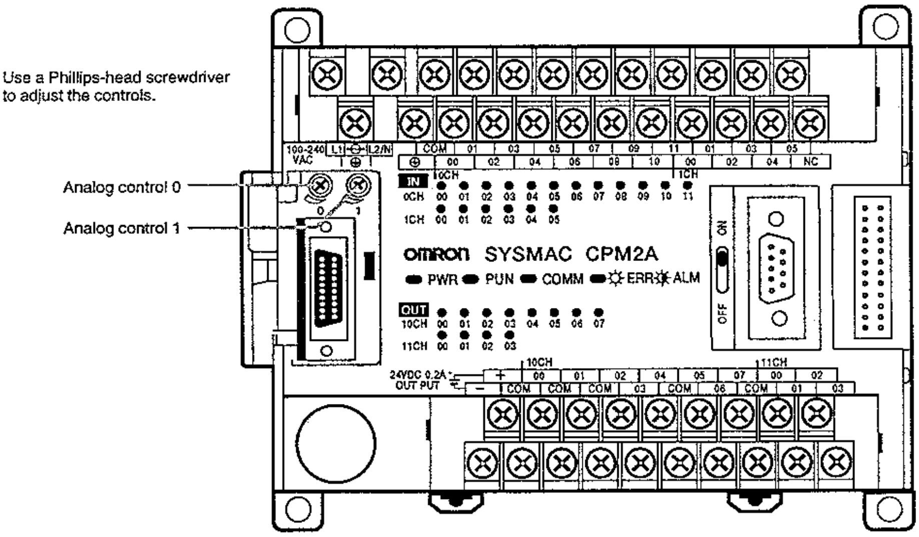
		Input Mode			
		Differential phase input mode	Pulse + direction outputs	Up/ down pulse outputs	Increment mode
Input number	**00000**	A-phase input	Count input	CW input	Count input
	00001	B-phase input	Direction	CCW input	See note 1.
Input method		Differential phase input (4z)	Single-phase input	Single-phase input	Single-phase input
Input frequency range		10 Hz to 500 Hz (accuracy ± 1 Hz) 20 Hz to 1 kHz (accuracy ± 1 Hz) 300 Hz to 20 kHz (accuracy ± 25 Hz)(See note 2.)			
Output number (See note 3.)	**01000**	Pulse output 0			
	01001	Pulse output 1			
Output method		Single-phase output			
Output frequency range		10 Hz to 10 kHz (accuracy 10 Hz)			
Frequency ratio		1 % to 1,000 % (Can be specified in units of 1 %)			
Synchronized control cycle		10 ms			

1. Can also be used as an ordinary input.
2. When 10 kHz or less, then the accuracy is ± 10 Hz
3. Either can be selected as the output number, using SYNC(-).

3. FEATURES & FUNCTIONS

• Analog Controls

The CPM2A has two analog controls that can be used for a wide range of timer and counter analog settings. As these controls are turned, values from 0 to 200 (BCD) are stored in the SR Area.

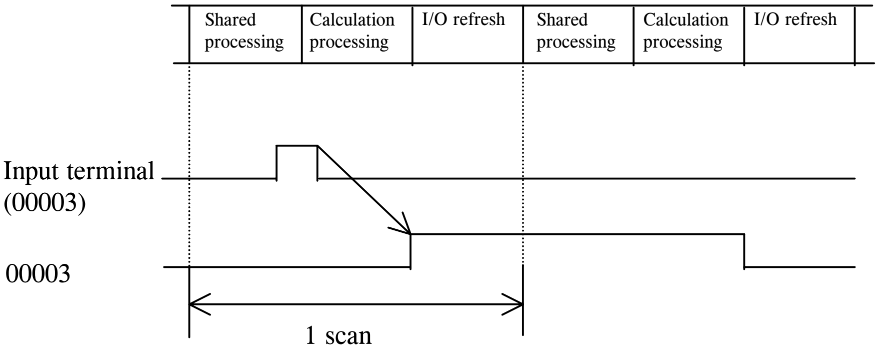


Control	Corresponding IR word	Setting range (BCD)
Analog volume control 0	SR 250	0000 to 0200
Analog volume control 1	SR 251	

The storage area is refreshed once with every CPM2A cycle.

• Quick-response Inputs

The CPM2A has four inputs used for quick response inputs (shared with interrupt inputs and 2-kHz high-speed counter inputs). With quick-response inputs, signals that are changed within a scan can be received by maintaining an internal buffer.



Input number (See note.)	Minimum input signal width
00003	50 µs
00004	
00005	
00006	

The following table shows the relationships between interval timer interrupts and the CPM2A's other functions.

	Interval time interrupts
Synchronized pulse control	Can be used simultaneously.
Interrupt inputs	See note 1.
Interval timer interrupts	Can be used simultaneously.
High speed counters	Can be used simultaneously.
Interrupt inputs (counter mode)	See note 2.
Pulse outputs	Can be used simultaneously.
Quick-response inputs	See note 3.
Input time constant	See note 4.
Clock	Can be used simultaneously.

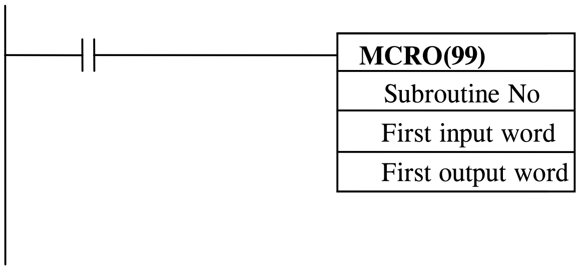
Note

1. Quick response input utilize the interrupt input function, so the same input number from the 00003 to 00006 cannot be assigned for both a quick response input and an interrupt input in the PLC setup.
2. A quick-response input and an interrupt in counter mode cannot be assigned the same input number in the PLC setup.
3. The input numbers allocated for quick quick-response inputs are 00003 to 00006. These inputs can be set and operated as quick-response inputs.
4. Input time constants are disable for all inputs that are set as quick-response inputs.

- **Marco Function**

The macro function allows a single subroutine (programming pattern) to be used by simply changing the I/O word. An number of similar program sections can be managed with just one subroutine, thereby greatly reducing the number of steps in the program and making the program easier to understand.

To use a macro, call a subroutine by means of the MACRO instruction, MCRC(99), as shown below, instead of SBS (91) (SUBROUTINE ENTRY).



4. UNIT SPECIFICATIONS AND COMPONENTS

❑ General Specifications of CPM1A CPUs Unit

Item		**10-point I/O**	**20-point I/O**	**30-point I/O**	**40-point I/O**
Supply voltage	AC type	100 to 240 VAC, 50/60 H			
	DC type	24 VDC			
Operating voltage range	AC type	85 to 264 VAC			
	DC type	20.4 to 26.4VDC			
Power consumption	AC type	30 VA max.		60 VA max.	
	DC type	6 W max.		20 W max.	
Inrush current		30 A max.		60 A max.	
External power supply (AC type only)	Power supply voltage	24 VDC			
	Power supply output capacity	200 mA		300 mA	
Insulation resistance		20 MΩ min. (at 500 VDC) between the external AC terminals and protective earth terminals.			
Dielectric strength		2,300 VAC 50/60 Hz for 1 min between the external AC and protective earth terminals, leakage current: 10 mA max.			
Noise immunity		1,500 Vp–p, pulse width: 0.1 to 1 µs, rise time: 1 ns (via noise simulation)			
Vibration resistance		10 to 57 Hz, 0.075–mm amplitude, 57 to 150 Hz, acceleration: 9.8 m/s^2 (1G) in X, Y and Z directions for 80 minutes each (i.e. swept for 8 minutes, 10 times)			
Shock resistance		147 m/s^2 (20G) three times each in X, Y and Z directions			
Ambient temperature		Operating: 0 ^{0} to 55 ^{0}C Storage: –20 ^{0} to 75 ^{0}C			
Ambient Humidity (operating)		10% to 90% (with no condensation)			
Ambient environment (operating)		With no corrosive gas			
Terminal screw size		M3			
Power supply holding time		AC type: 10 ms min; DC type: 2 ms min. (A power interruption occurs if power falls below 85% of the rated voltage for longer than the power interruption time.)			
CPU weight	AC type	400 g max.	500 g max.	600 g max.	700 g max.
	DC type	300 g max.	400 g max.	500 g max.	600 g max.

Note : The specifications of the Expansion I/O Unit are the same as for the CPU except that the power is supplied from the CPU and the weight is 300g.

4. UNIT SPECIFICATIONS AND COMPONENTS

❑ Characteristics

Item		10-point I/O	20-point I/O	30-point I/O	40-point I/O
Control method		Stored program method			
I/O control method		Combination of the cyclic scan and immediate refresh processing methods.			
Programming language		Ladder diagram			
Instruction length		1 step per instruction, 1 to 5 words per instruction			
Types of instructions		Basic instruction: 14 Special instructions: 77 types, 135 instructions			
Execution time		Basic instructions: 0.72 to 16.2 μs Special instructions: 12.375 μs (MOV instruction)			
Program capacity		2,048 words			
Maximum I/O points	CPU only	10 point (6 input/4 output)	20 point (12 input/8 input)	30 point (18 input/12 output)	40 point (24 input/16 output)
	With Expansion I/O unit	- - -	- - -	99 point (54 input/36 output)	100 point (60 input/40 output)
Input bits		00000 to 00915 (Words 0 to 9)			
Output bits		01000 to 01915 (Words 10 to 19)			
Work bits (IR area)		512 bits: IR 20000 to 23115 (Words IR 200 to IR 231)			
Special bits (SR area)		384 bits: SR 23200 to 25515 (Words SR 232 to IR 255)			
Temporary bits (TR area)		8 bits (TR0 to TR7)			
Holding bits (HR area)		320 bits: HR 0000 to HR 1915 (Words HR 00 to HR 19)			
Auxiliary bits (AR area)		256 bits: AR 0000 to AR 1515 (Words AR 00 to AR 15)			
Link bits (LR area)		256 bits: LR 0000 to LR 1515 (Words LR 00 to LR 15)			
Timers/Counters		128 timers/counters (TIM/CNT 000 to TIM/CNT 127) 100-–ms timers: TIM000 to TIM 127 10–ms timers : TIM 00 to TIM 127 Decrementing counters and reversible counters			
Data memory		Read/write: 1,024 words (DM 0000 to DM 1023) Read–only: 512 words (DM 6144 to DM 6655)			
Interrupt processing		2 points (Response time: 0.3 ms max.	4 points (Response time: 0.3 ms max.)		
Memory protection		HR, AR, Data Memory area contents and counter values maintained during power interruptions.			
Memory backup		Flash memory: User program, data memory (Read only) (Non-battery powered storage) Capacitor backup: Data Memory (Read/Write), holding bits, auxiliary memory bits, counter (20 days at ambient temperature of 25 °C).			
Self–diagnostic functions		CPU failure (watchdog timer), I/O bus error, and memory failure			
Program checks		No END instruction, programming errors (continuously checked during operation)			
High–speed counter		1 point: 5 kHz single–phase or 2.5 kHz two–phase (linear count method) Increment mode: 0 to 65,535 (16 bits) Up/Down mode: –32,767 to 32,767 (16 bits)			
Quick–response inputs		Together with the external interrupts inputs. (Min. pulse width: 0.2 ms)			
Input time constant		Can be set to 1 ms, 2 ms, 4 ms, 8 ms, 16 ms, 32 ms, 64 ms, or 128 ms.			
Analog volume settings		2 controls (0 to 200 BCD)			

Note: Bits not used as input bits can used as work bits.

4. UNIT SPECIFICATIONS AND COMPONENTS

❑ Structure of Memory Area

Data area		Words	Bits	Function
IR area [1]	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	These bits can be allocated to the external I/O terminals.
	Output area	IR 010 to IR 019 (10 words)	IR 01000 to IR 01915 (160 bits)	
	Work area	IR 200 to IR 231 (32 words)	IR 20000 to IR 23115 (512 bits)	Work bits can be freely used within the program.
SR area		SR 232 to SR 255 (24 words)	SR 23200 to SR 25515 (384 bits)	These bits serve specific functions such as flags and control bits.
TR area		- - -	TR 0 to TR 7 (8 bits)	These bits are used to temporarily store ON/OFF status at program branches.
HR area [2]		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	These bits store data and retain their ON/OFF status when power is turned off.
AR area [2]		AR 00 to HR 15 (16 words)	AR 0000 to HR 1515 (256 bits)	These bits serve specific functions such as flags and control bits.
LR area [1]		LR 00 to LR 15 (16 words)	LR 00000 to LR1515 (256 bits)	Used for a 1:1 data link with another PC.
Timer/Counter area [2]		TC 000 to TC 127 (timer/counter numbers) [3]		The same numbers are used for both timers and counters.
DM area	Read/write[2]	DM 0000 to DM 0999 DM 1022 to DM 1023 (1,002 words)	- - -	DM area data can be accessed in word units only. Word values are required when the power is turned off.
	Error log [4]	DM 1000 to DM 1021 (22 words)	- - -	Used to store the timer of occurrence and error code of errors that occur. These words can be used as ordinary read/write DM when the error log function isn't being used.
	Read–only [4]	DM 6144 to DM 6599 (456 words)	- - -	Cannot be overwritten from program.
	PC Setup [4]	DM 6600 to DM 6655 (56 words)	- - -	Used to store various parameters that control PC operation.

Note:

1. IR and LR bits that are not used for their allocated functions can be used as work bits.

2. The contents of the HR area, LR area, Counter area, and read/write DM area are backed up by a capacitor. At 25 °C, the capacitor will back up memory for 20 days.

3. When accessing a PV, TC numbers are used as word data; when accessing Completing Flags, they are used as bit data.

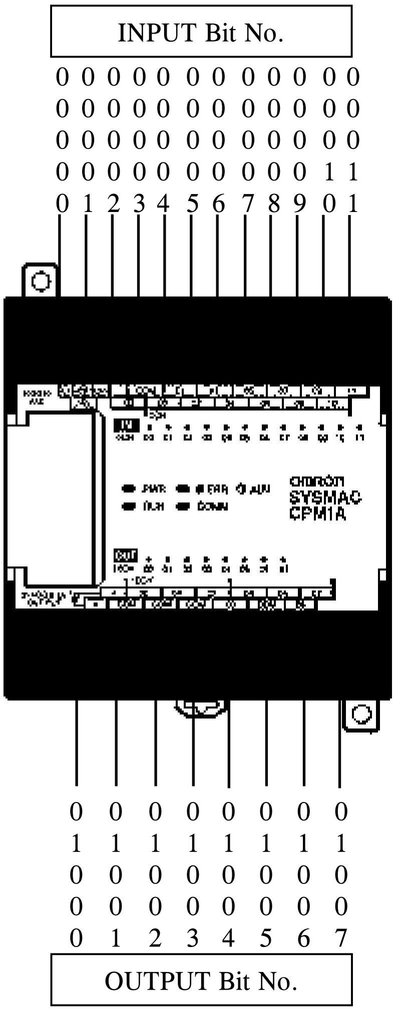
4. Data in DM6144 to DM6655 cannot be overwritten from the program, but they can be changed from a Peripheral Device.

4. UNIT SPECIFICATIONS AND COMPONENTS

❑ I/O Terminal – IR Bit Allocation

The following table shows which IR bits are allocated to the I/O terminals on the CPM1A's CPU and Expansion I/O Unit.

Number of I/O Terminals on the CPU	CPU Terminals		Expansion I/O Unit Terminals		Power Supply	Model Number
	Inputs	Outputs	Inputs	Outputs		
10	6 points: 00000 to 00005	4 points: 01000 to 01003	- - -	- - -	AC	CPM1A-10CDR-A
					DC	CPM1A-10CDR-D
20	12 points: 00000 to 00011	8 points: 01000 to 01007	- - -	- - -	AC	CPM1A-20CDR-A
					DC	CPM1A-20CDR-D
30	18 points: 000000 to 00011 00100 to 00105	12 points: 01000 to 01007 01100 to 01103	36 points: 00200 to 00211 00300 to 00311 00400 to 00411	24 points: 01200 to 01207 01300 to 01307 01400 to 01407	AC	CPM1A-30CDR-A
					DC	CPM1A-30CDR-D
40	24 points: 00000 to 00011 00100 to 00111	16 points: 01000 to 01007 01100 to 01107			AC	CPM1A-40CDR-A
					DC	CPM1A-40CDR-D



❑ General Specifications of CPM2A CPU Units

Item		30-point I/O	40-point I/O	60-point I/O
Supply voltage	AC type	100 to 240 VAC, 50/60 H		
	DC type	24 VDC		
Operating voltage range	AC type	85 to 264 VAC		
	DC type	20.4 to 26.4VDC		
Power consumption	AC type	60 VA max.		
	DC type	20 W max.		
Inrush current	AC type	60 A max.		
	DC type	20 A max.		
External power supply (AC type only)	Supply voltage	24 VDC		
	Output capacity	300 mA (When the external power supply provides an overcurrent or is short circuited, the external power supply voltage will drop but the PC will continue operation.)		
Insulation resistance		20 MΩ min. (at 500 VDC) between the external AC terminals and protective earth terminals.		
Dielectric strength		2,300 VAC 50/60 Hz for 1 min between the external AC and protective earth terminals, leakage current: 10 mA max.		
Noise immunity		1,500 Vp–p, pulse width: 0.1 to 1 µs, rise time: 1 ns (via noise simulation)		
Vibration resistance		10 to 57 Hz, 0.075–mm amplitude, 57 to 150 Hz, acceleration: 9.8 m/s^2 (1G) in X, Y and Z directions for 80 minutes each (i.e. swept for 8 minutes, 10 times)		
Shock resistance		147 m/s^2 (20G) three times each in X, Y and Z directions		
Ambient temperature		Operating: 0 ^{0} to 55 ^{0}C Storage: –20 ^{0} to 75 ^{0}C		
Humidity		10% to 90% (with no condensation)		
Environment		With no corrosive gas		
Terminal screw size		M3		
Power supply holding time		AC type: 10 ms min. DC type: 2 ms min. (A power interruption occurs if power falls below 85% of the rated voltage for longer than the power interruption time.)		
CPU weight	AC type	800 g max.	900 g max.	1,100 g max.
	DC type	700 g max.	800 g max.	1,000 g max.
Expansion I/O Unit weight		Units with 20 I/O points: 300g max. Units with 8 output points: 250g max. Units with 8 input points: 200g max. Analog I/O Units: 200g max. CompoBus/S I/O Link Units: 200g max.		

4. UNIT SPECIFICATIONS AND COMPONENTS

❑ Characteristics

<table>
<tr><th colspan="2">Item</th><th colspan="3">Specification</th></tr>
<tr><td colspan="2">Control method</td><td colspan="3">Storage program method</td></tr>
<tr><td colspan="2">I/O control method</td><td colspan="3">Cyclic scan with direct output (Immediate refreshing can be performed with IORF(97).)</td></tr>
<tr><td colspan="2">Programming language</td><td colspan="3">Ladder diagram</td></tr>
<tr><td colspan="2">Instruction length</td><td colspan="3">1 step per instruction, 1 to 5 words per instruction</td></tr>
<tr><td colspan="2">Types of instructions</td><td colspan="3">Basic instructions: 14
Special instructions: 106 instructions,185 variations</td></tr>
<tr><td colspan="2">Execution time</td><td colspan="3">Basic instructions: 0.9 µs(LD instruction)
Special instructions: 12.375 µs(MOV instruction)</td></tr>
<tr><td colspan="2">Program capacity</td><td colspan="3">4,096 words</td></tr>
<tr><td rowspan="2">Max. I/O points</td><td>CPU only</td><td>30 points</td><td>40 points</td><td>60 points</td></tr>
<tr><td>With Expansion I/O Unit</td><td>90 points max.</td><td>100 points max.</td><td>120 points max.</td></tr>
<tr><td colspan="2">Input bits</td><td colspan="3">00000 to 00915 (Words not used for input bits can be used for work bits.)</td></tr>
<tr><td colspan="2">Output bits</td><td colspan="3">01000 to 01915 (Words not used for input bits can be used for work bits.)</td></tr>
<tr><td colspan="2">Work bits (IR area)</td><td colspan="3">512 bits: IR 20000 to 23115 (Words IR 200 to IR231)</td></tr>
<tr><td colspan="2">Special bits (SR area)</td><td colspan="3">448 bits: SR 20000 to 25515 (Words SR 228 to SR255)</td></tr>
<tr><td colspan="2">Temporary bits (TR area)</td><td colspan="3">8 bits (TR0 to TR7)</td></tr>
<tr><td colspan="2">Holding bits (HR area)</td><td colspan="3">320 bits: HR 0000 to HR 1915 (Words HR 00 to HR 19)</td></tr>
<tr><td colspan="2">Auxiliary bits (AR area)</td><td colspan="3">256 bits: :LR 0000 to AR 2315 (Words RA 00 to AR 23)</td></tr>
<tr><td colspan="2">Link bits (LR area)</td><td colspan="3">256 bits: LR 0000 to LR 1515 (Words LR 00 to LR 15)</td></tr>
<tr><td colspan="2">Timers/Counters</td><td colspan="3">256 timers/counters (TIM/CNT 000 to TIM/CNT 255).
1-ms timers: TMHH(-)
10-ms timers: TIMH(15)
100-ms timers: TIM
1-s/10-s timers: TIML(-)
Decrementing counters: CNT
Reversible counters: CNTR(12)
Use timer numbers 004 to 007 when creating a timer using a high-speed timer instruction (TMHH(-) or TIMH(15) to perform interrupt processing.</td></tr>
<tr><td colspan="2">Data memory</td><td colspan="3">Read/Write: 2,048 words (DM 0000 to DM 2047)*
Read-only: 456 words (DM 6144 to DM 6599)
PC Setup: 56 words (DM 6600 to DM 6655)
*The Error Log is contained in DM 2000 to DM 2021.</td></tr>
<tr><td colspan="2">Interrupt processing</td><td colspan="3">External Interrupts: 4
Shared by the external inputs (counter mode) and the quick response inputs.</td></tr>
<tr><td colspan="2">Interval timer interrupts</td><td colspan="3">1 (Scheduled Interrupt Mode or Single Interrupt Mode)</td></tr>
<tr><td colspan="2">High-Speed Counter (Hardware counter)</td><td colspan="3">One high-speed counter: 20 kHz single-phase or 5 kHz two-phase (Linear count method)
Counter interrupt: 1 (set value comparison or set value range comparison)</td></tr>
</table>

Item	Specification
Interrupt Inputs (Counter mode)	Four inputs (share by the external interrupt inputs (counter mode) and the quick-response inputs.) Counters interrupts: 4 (shared by the external interrupt inputs and quick-response inputs.
Pulse Output	Two points with no acceleration/deceleration, 10 Hz to 10 kHz each, and no direction control. One point with waveform acceleration/deceleration, 10 Hz to 10 kHz, and direction control. Two points with variable duty-ration outputs.

4. UNIT SPECIFICATIONS AND COMPONENTS

❑ CPM2A Memory Area Structure

Data area		Words	Bits	Function
IR area [1]	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	These bits can be allocated to the external I/O terminals.
	Output area	IR 010 to IR 019 (10 words)	IR 01000 to IR 01915 (160 bits)	
	Work area	IR 020 to IR 049, IR 200 to IR 231 (58 words)	IR 02000 to IR 04915, IR 20000 to IR 23115 (928 bits)	Work bits can be freely used within the program.
SR area		SR 228 to SR 255 (28 words)	SR 22800 to SR 25515 (440 bits)	These bits serve specific functions such as flags and control bits.
TR area		- - -	TR 0 to TR 7 (8 bits)	These bits are used to temporarily store ON/OFF status at program branches.
HR area [2]		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	These bits store data and retain their ON/OFF status when power is turned off.
AR area [2]		AR 00 to HR 23 (23 words)	AR 0000 to HR 2315 (384 bits)	These bits serve specific functions such as flags and control bits.
LR area [1]		LR 00 to LR 15 (16 words)	LR 00000 to LR1515 (256 bits)	Used for a 1:1 data link with another PC.
Timer/Counter area [2]		TC 000 to TC 225 (timer/counter numbers) [3]		The same numbers are used for both timers and counters.
DM area	Read/write[2]	DM 0000 to DM 1999 DM 2022 to DM 2047 (2,026 words)	- - -	DM area data can be accessed in word units only. Word values are retained when the power is turned off.
	Error log [4]	DM 2000 to DM 2021 (22 words)	- - -	Used to store the time of occurrence and error code of errors that occur. These words can be used as ordinary read/write DM when the error log function isn't being used.
	Read–only [4,5]	DM 6144 to DM 6599 (456 words)	- - -	Cannot be overwritten from program.
	PC Setup [4,5]	DM 6600 to DM 6655 (56 words)	- - -	Used to store various parameters that control PC operation.

Note

1. IR and LR bits that area not used for their allocated function can be used as work bits.
2. The contents of the HR area, LR area, Counter area, and read/write DM are backed up by the CPU Unit's battery. If the battery is removed or fails, the content of these area will be lost and unstable.
3. When a TC numbers is used as a word operand, the timer or counter PV is accessed; when used as bit operand, its Completion Flag is accessed.
4. Data is DM 6144 to DM 6655 cannot be overwrite from the program, but they can be changed from a Programming Device.
5. The program and data in DM 6144 to DM 6655 are stored in flash memory.

❑ I/O Terminal – IR Bit Allocation

CPU Unit	I/O	CPU Unit Terminals
CPM2A-30CD -	18 inputs	00000 to 00011 (IR 00000 to IR 00011) and 00100 to 00105 (IR 00100 to IR 00105)
	12 outputs	01000 to 01007 (IR 01000 to IR 01007) and 01100 to 01103 (IR 01100 to IR 01103)
CPM2A-40CD -	24 inputs	00000 to 00011 (IR 00000 to IR 00011) and 00100 to 00111 (IR 00100 to IR 00111)
	16 outputs	01000 to 01007 (IR 01000 to IR 01007) and 01100 to 01107 (IR 01100 to IR 01107)
CPM2A-60CD -	36 inputs	00000 to 00011 (IR 00000 to IR 00011), 00100 to 00111 (IR 00100 to IR 00111) and 00200 to 00211 (IR 00200 to IR 00211)
	24 outputs	01000 to 01007 (IR 01000 to IR 01007), 01100 to 01107 (IR 01100 to IR 01207) and 01200 to 01207 (IR 01200 to IR 01207).

❑ Expansion Unit Allocation

Up to 3 Expansion Units can be connected to a CPU Unit. I/O bits are allocated to the Expansion Units beginning with the next available I/O words.

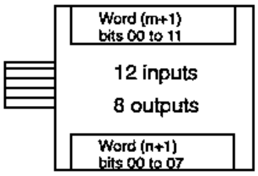
Type of Expansion Unit available.

No. of I/O terminals	Inputs	outputs	Power Supply	Model
20	12 points	8 points	DC	CPM1A-EDR
				CPM1A-EDT
				CPM1A-EDT1
8	8 points	0 point	DC	CPM1A-8ED
8	0 point	8 points	DC	CPM1A-8ER
				CPM1A-8ET
				CPM1A-8ET1
-	2 Analog	1 Analog	DC	CPM1A-MAD01
16	8 points	8 points	DC	CPM1A-SRT21

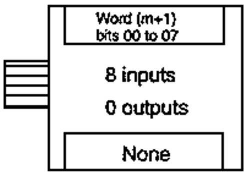
The expansion unit as stated above can also be used by CPM1A

❑ Example of I/O Allocation

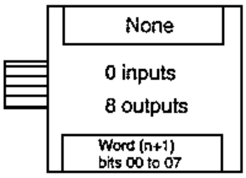
CPM1A-20ED□□□
Expansion I/O Unit



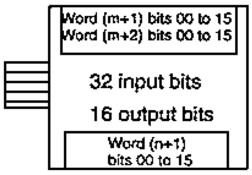
CPM1A-8ED
Expansion I/O Unit



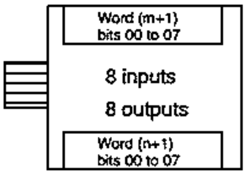
CPM1A-8ER/8ET/8ET1
Expansion I/O Unit



CPM1A-MAD01
Analog I/O Unit



CPM1A-SRT21
CompoBus/S I/O Link Unit



Example 1:

A CPM1A-20ED Expansion I/O unit (12 inputs and 8 outputs) is connected to a CPU unit with 30 I/O points.

Unit		Allocated input bits	Allocated output bits
1	CPU Unit (CPM2A-30CD -)	IR 00000 to IR 00011 and IR 00100 to IR 00105	IR 01000 to IR 01007 and IR 01100 to IR01103
2	Expansion I/O unit (CPM1A-20ED)	IR 00200 to IR 00211	IR 01200 to IR 01207

Example 2:

Three Expansion Unit are connected to a CPU Unit with 30 I/O points.

Unit		Allocated input bits	Allocated output bits
1	CPU Unit (CPM2A-30CD -)	IR 00000 to IR 00011 and IR 00100 to IR 00105	IR 01000 to IR 01007 and IR 01100 to IR01103
2	Expansion I/O unit (CPM1A-20ED)	IR 00200 to IR 00211	IR 01200 to IR 01207
3	Analog I/O unit (CPM1A-MAD01)	IR 00300 to IR 00315 and IR 00400 to IR 00415	IR 01300 to IR 01315
4	Expansion I/O Unit (CPM1A-ED)	IR 00500 to IR 00511	IR 04100 to IR 01415

❑ Example of I/O Allocation (continue)

Example 3:

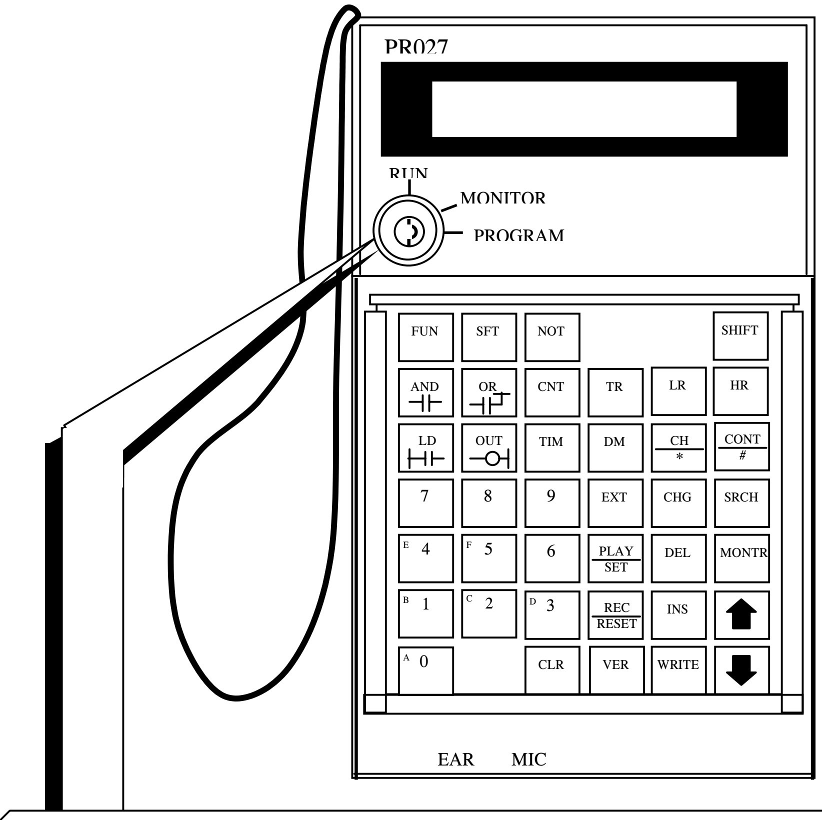
Three Expansion Unit are connected to a CPU Unit with 30 I/O points.

Unit		Allocated input bits	Allocated output bits
1	CPU Unit (CPM2A-30CD -)	IR 00000 to IR 00011 and IR 00100 to IR 00105	IR 01000 to IR 01007 and IR 01100 to IR01103
2	Expansion I/O unit (CPM1A-20ED)	IR 00200 to IR 00211	IR 01200 to IR 01207
3	Expansion I/O unit (CPM1A-MAD01)	IR 00300 to IR 00307	None*
4	Expansion I/O Unit (CPM1A-8ER/ET/ET1)	None*	IR 01300 to IR 01037

Note If an Expansion I/O Unit does not have inputs, no input word will be allocated. The same is true for outputs.

❑ Programming Console

Before we start to program the PLC, lets familiarize ourselves with the essential operations of the keyboard and initial settings required.



The PC may be set to either *PROGRAM*, *MONITOR* or *RUN* modes.

PROGRAM mode is used for preparing programs or for making modifications or corrections to existing programs.

MONITOR mode is used when changing the contents of memory areas while the PLC is actually in operation.

RUN mode is used to execute the program that has been entered into the PLC. No changes can be made to the internal data in the run mode.

4. UNIT SPECIFICATIONS AND COMPONENTS

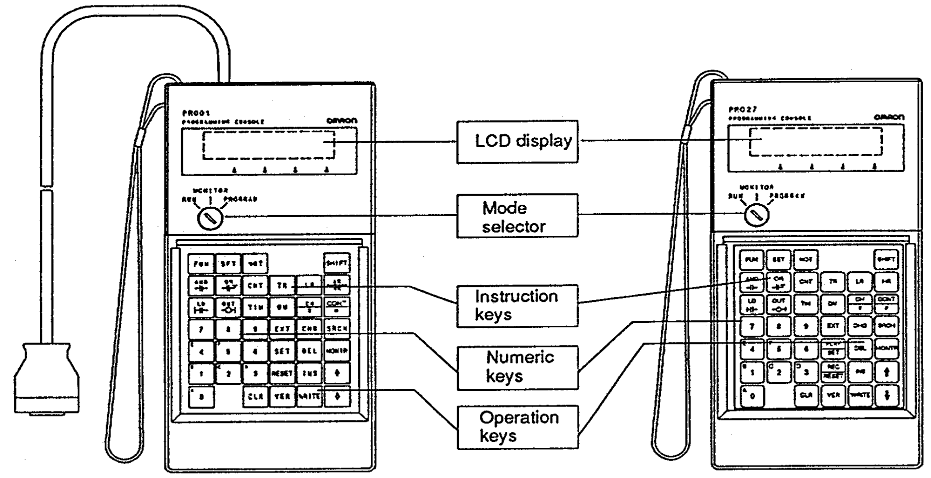
• A look at the Programming Console

In order for the PLC to operate, you must first put a program into the memory. The program is made by sequentially inputting commands using the Programming Console.

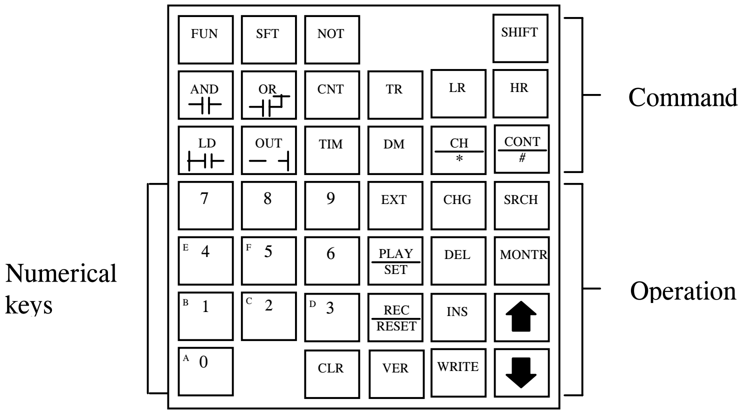
Please note that CQM1–PRO01E/C200H–PRO27E can be used with CPM1A, CPM1, CQM1, C200HS AND C200HX.

CQM1-PRO01-E
(A 2-m connecting cable is attached.)

C200H-PRO27-E
(Use a C200H-CN222 (2 m) or C200H-CN422 (4 m) Connecting Cable.)



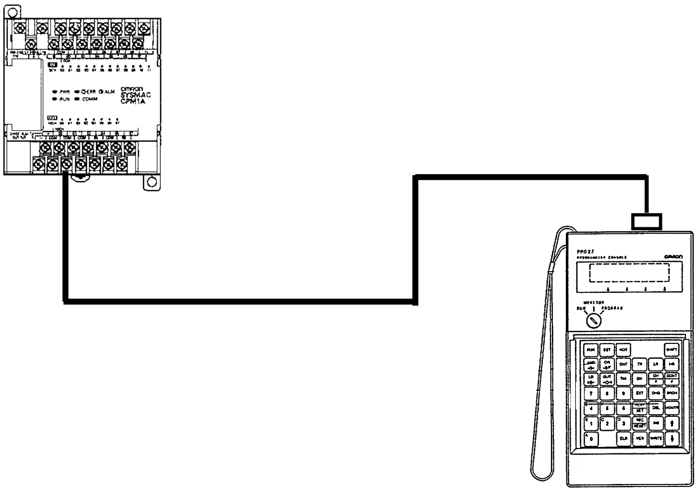
Keyboard



FUN	SFT	NOT			SHIFT
AND	OR	CNT	TR	LR	HR
LD	OUT	TIM	DM	CH *	CONT #
7	8	9	EXT	CHG	SRCH
E 4	F 5	6	PLAY SET	DEL	MONTR
B 1	C 2	D 3	REC RESET	INS	⬆
A 0		CLR	VER	WRITE	⬇

❑ Password Input

The PLC has a password control to prevent unauthorized access to its program. The PLC always prompts you for the password when power is first applied or after programming console is installed when the PLC is in the operation mode. To enter the password, press the CLR, MONTR, CLR keys.



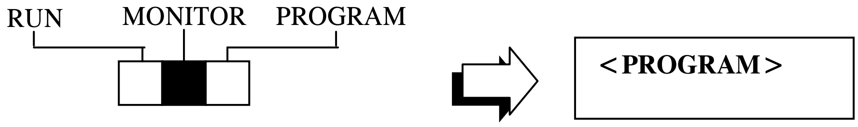
```
<PROGRAM>
PASSWORD!
```

CLR → MONTR → CLR → <PROGRAM>

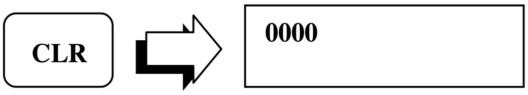
❑ Clearing All Program

Erasing a previous program in the memory of the CPU is referred to as an ALL CLEAR operation. The memory areas of the Holding Relay, Counter and Data Memory can be retained by pressing the appropriate key before pressing the MONTR key.

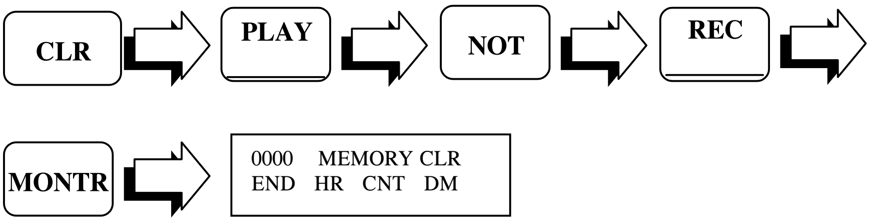
The ALL CLEAR operation can only be performed with the Mode Selection switch set to PROGRAM mode.



Pressed the CLR key until you see 0000 displayed on the programming console.



The PLC memory will be cleared after the following keystrokes:



The display should now look like

! PARTIAL CLEAR

Press the HR, CNT or DM keys before pressing MONTR key if you want to preserve any of these data areas.

5. WINDOW BASED PROGRAMMING SOFTWARE-SYSWIN

❑ Programming of CPM1A and CPM2A

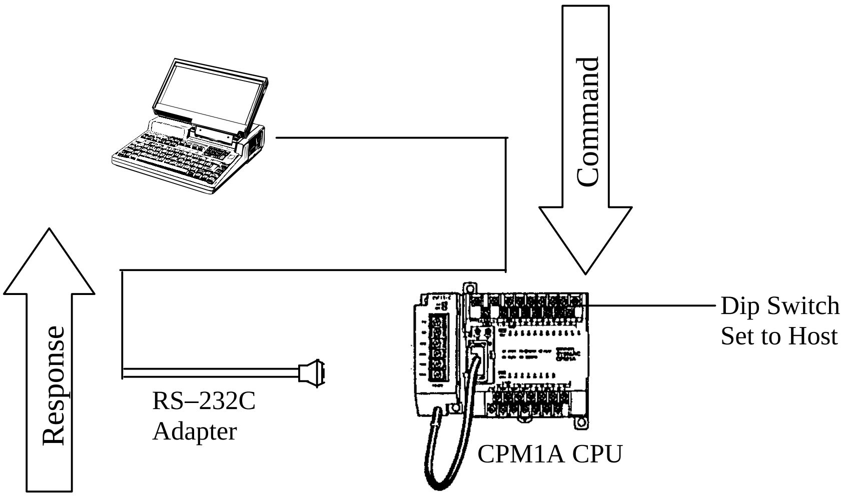
SYSWIN is a ladder programming software written to give users the flexibility and ease of a Window software. It is a tool that caters to Omron’s C-series PLCs, CV-series PLCs and all connectivity options.

Note: SYSWIN is provided without token or dongle only to program CPM1 and CPM1A. For C-series, and CV series PLCs, token or dongle is necessary.

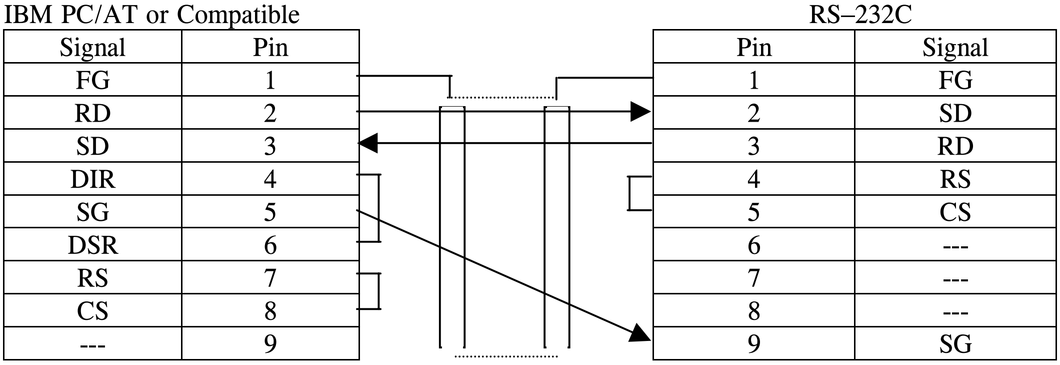
• Programming Using SYSWIN

Connection to the PLC

The CPM1A or CPM2A can be connected to the PC via an RS–232C cable. One end of the RS-232C is connected to the PC serial port (either 9 pin or 25 pin adapter), while the other end of the cable which is to be connected to the RS–232C adapter attached to the CPM1Aor CPM2A. The DIP switch of the adapter must be set to Host in order to link to the PC. The cable can also be connected to CPM2A through the built-in RS-232C port provided.



RS232C Connector Configuration



IBM PC/AT or Compatible

Signal	Pin
FG	1
RD	2
SD	3
DIR	4
SG	5
DSR	6
RS	7
CS	8
---	9

RS–232C

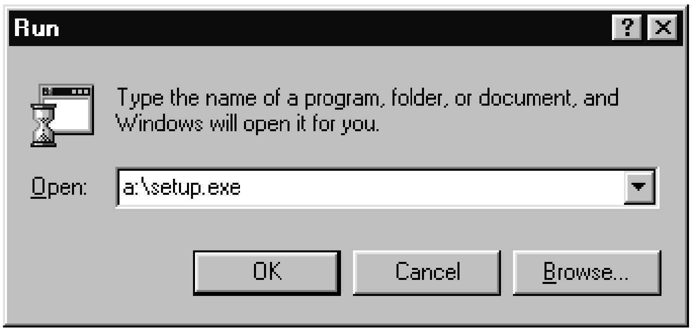
Pin	Signal
1	FG
2	SD
3	RD
4	RS
5	CS
6	---
7	---
8	---
9	SG

The wiring connection of CPM2A and CPM1A is the same.

5. WINDOW BASED PROGRAMMING SOFTWARE-SYSWIN

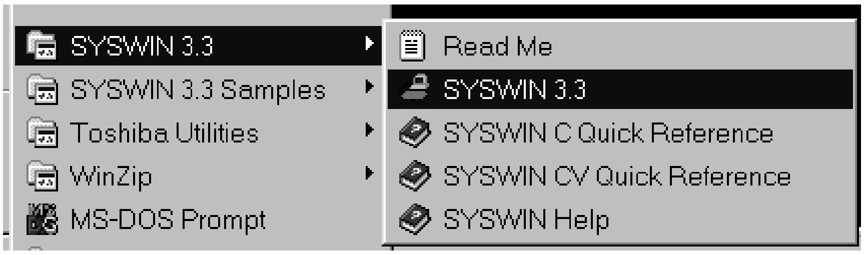
Installing SYSWIN Program

The CPM1A training kit comes with three SYSWIN installation disks. The SYSWIN program can be setup in Windows 3.1 or 3.11, Window 95,98 or Window NT 4.0. To install, select RUN from the START menu. The following dialog box will appear, enter a:\setup.exe in the command line, click OK to execute and follow the directions as they appear on the screen.


Run
?
Type the name of a program, folder, or document, and Windows will open it for you.
OK
Cancel
Browse...

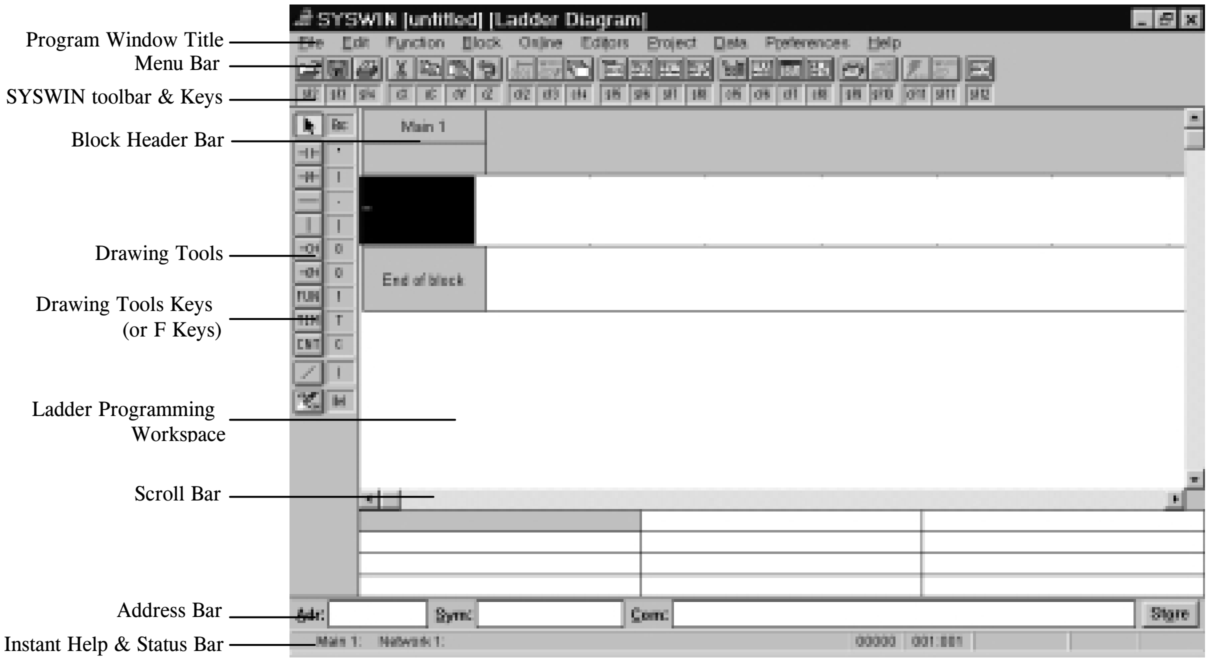
Open: a:\setup.exe

Once installation is completed, two Program Group icons will appear in the Program Manager. The SYSWIN samples contains sample programs whereas SYSWIN contains the necessary program to run SYSWIN. To execute SYSWIN, double click on the SYSWIN icon.



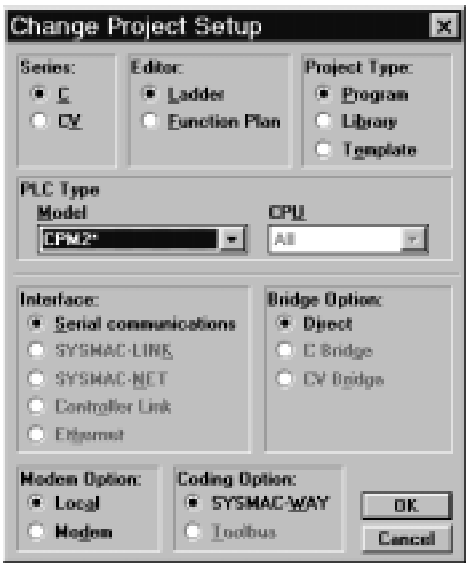
5. WINDOW BASED PROGRAMMING SOFTWARE-SYSWIN

SYSWIN Window


Program Window Title
Menu Bar
SYSWIN toolbar & Keys
Block Header Bar
Drawing Tools
Drawing Tools Keys (or F Keys)
Ladder Programming Workspace
Scroll Bar
Address Bar
Instant Help & Status Bar
SYSWIN [untitled] [Ladder Diagram]
File Edit Function Block Online Editors Project Data Preferences Help
Main 1
End of block
Adr: Sym: Com: Store
Main 1: Network 1:
00000 001:001

Start Up With SYSWIN

Select New Project from File menu. The New Project Setup dialog box will appear as shown below. Make all the selections as shown below. (The default values should be correct).



Change Project Setup

Series: ● C ○ CV

Editor: ● Ladder ○ Function Plan

Project Type: ● Program ○ Library ○ Template

PLC Type
Model: CPM2*
CPU: All

Interface: ● Serial communications ○ SYSMAC-LINK ○ SYSMAC-NET ○ Controller Link ○ Ethernet

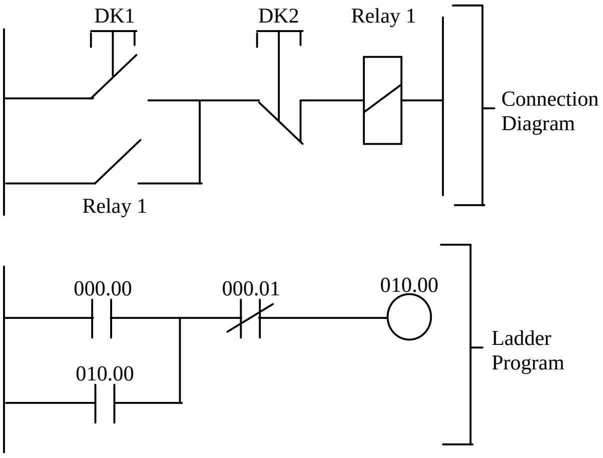
Bridge Option: ● Direct ○ C Bridge ○ CV Bridge

Modem Option: ● Local ○ Modem

Coding Option: ● SYSMAC-WAY ○ Toolbus

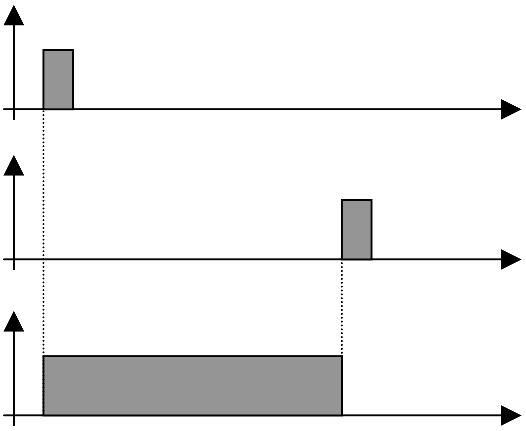
Programming with SYSWIN

Enter a sample program into the SYSWIN as shown.

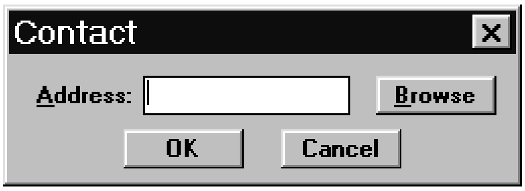


Instruction Codes

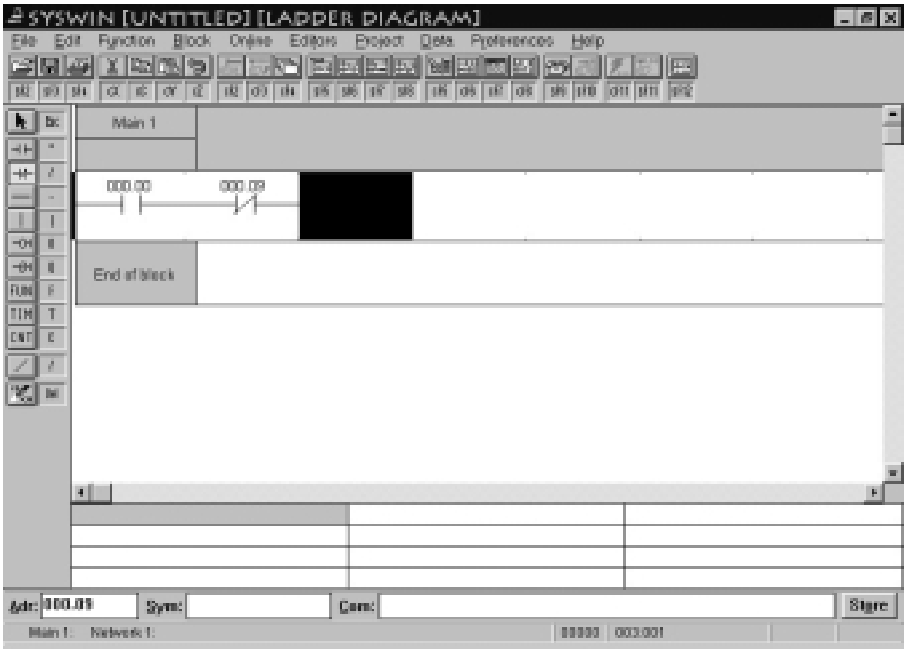
Address	Instruction	Data
00000	LD	000.00
00001	OR	010.00
00002	AND NOT	000.01
00003	OUT	010.00
00004	END	
00005		
00006		
00007		



Select the symbol of ladder program ([F2], normally open contact) from the drawing Toolbox on side of the SYSWIN window. Click on the symbol and move the symbol to anywhere in the window and click to release. The address of the contact must be specified and key into the address dialog box. In this example, 000.00 is entered.

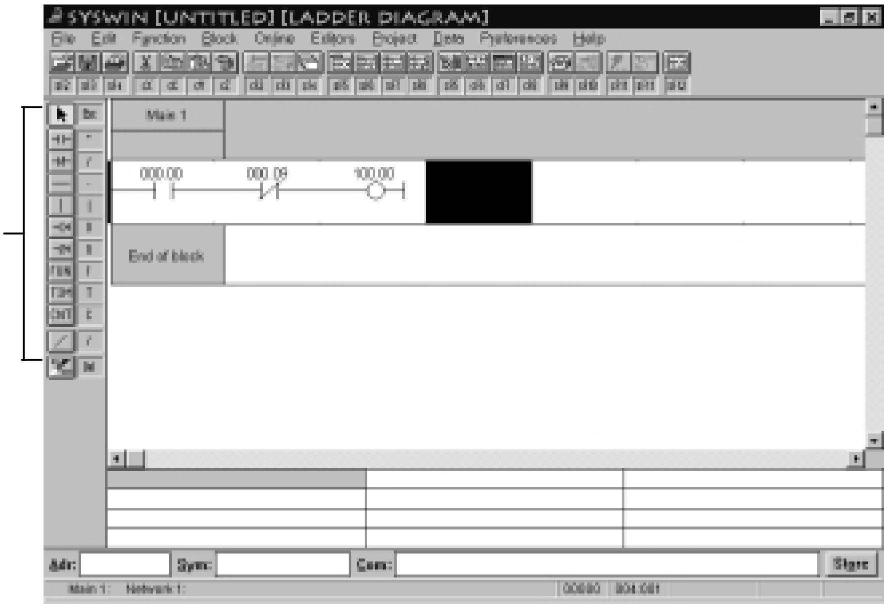


The address of the specific contact will be labeled on top of each component as shown.

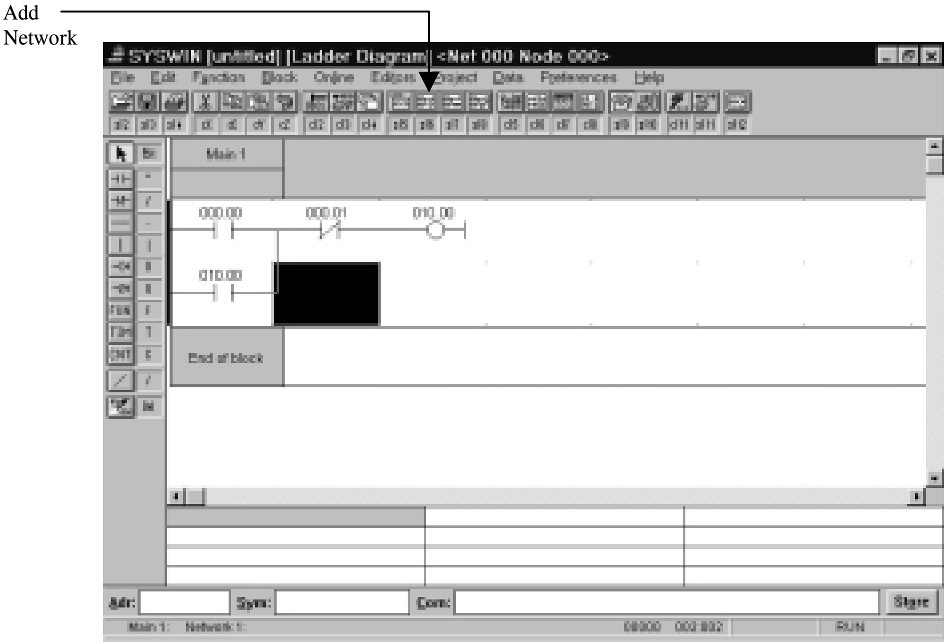

SYSWIN [UNTITLED] [LADDER DIAGRAM]
File Edit Function Block Online Editors Project Data Preferences Help
Main 1
000.00
000.09
End of block
Adr: 000.09
Sym:
Com:
Store
Main 1: Network 1:
00000 003:001

Each particular model of PLC has its own address for input and output, check the catalogue for each type of PLC. The address 000.00 represent address of the contact, the first three zero represent the channel number and the other 2 zero after the decimal represent the particular bit of the channel.

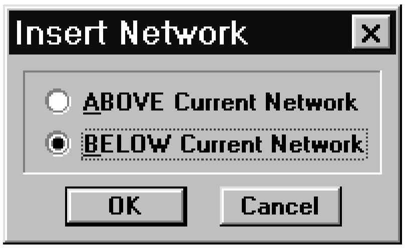
Continue with the rest of the sample program by clicking on the respective symbols from the Drawing Toolbox.


SYSWIN [UNTITLED] [LADDER DIAGRAM]
File Edit Function Block Online Editors Project Data Preferences Help
Main 1
000.00
000.09
100.00
End of block
Adr:
Sym:
Com:
Store
Main 1: Network 1:
00000 004:001

Drawing Tool Box


Add Network
SYSWIN [untitled] [Ladder Diagram] <Net 000 Node 000>
File Edit Function Block Online Editors Project Data Preferences Help
Main 1
000.00
000.01
010.00
010.00
End of block
Adr:
Sym:
Com:
Store
Main 1: Network 1:
RUN

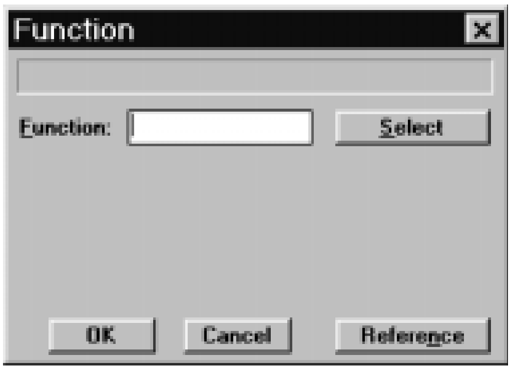
Note that each line of the ladder program is classified as a network. So in order to create another line, click the icon **Add Network (Shift+F6)** from the Toolbox to insert another network. The dialog box shown below appear when icon is clicked. Select below current network and click OK.


Insert Network

○ ABOVE Current Network

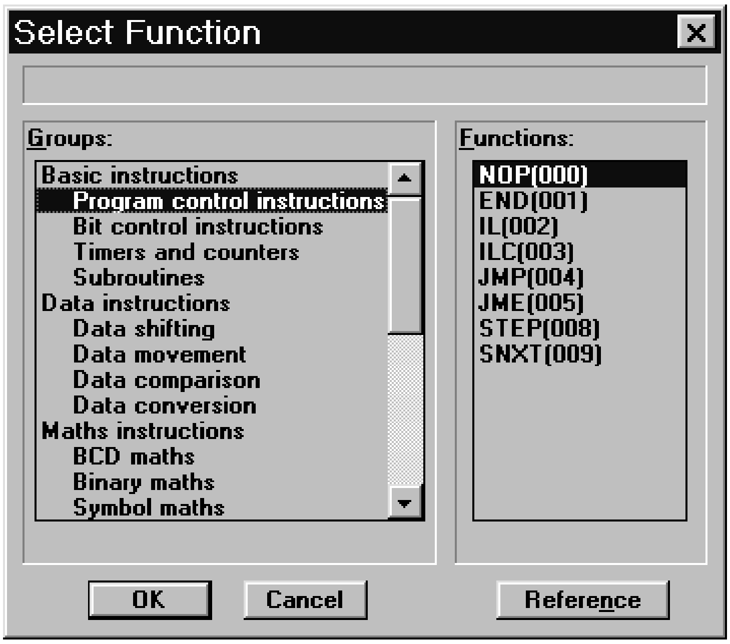
● BELOW Current Network

The same program will end with an end instruction line. In order to insert the end command, click on the FUN icon on side of the window, move it to the location to end the program, click to execute. A function dialog box appear, requiring you to enter the function number to execute.


Function

Function:

To browse for the types of functions available, lick on the select button. The select function box will be shown.

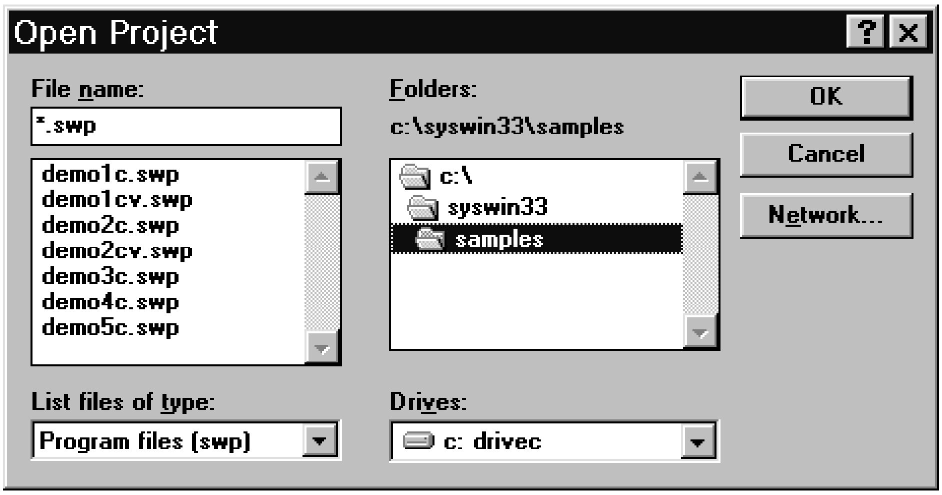


Groups:
- Basic instructions
  - Program control instructions
  - Bit control instructions
  - Timers and counters
  - Subroutines
- Data instructions
  - Data shifting
  - Data movement
  - Data comparison
  - Data conversion
- Maths instructions
  - BCD maths
  - Binary maths
  - Symbol maths

Functions:
- NOP(000)
- END(001)
- IL(002)
- ILC(003)
- JMP(004)
- JME(005)
- STEP(008)
- SNXT(009)

Saving the Project

After completing the ladder diagram, the project must be saved. Select Save Project in the file menu, a save project dialog box appear as shown below. Enter the file name e.g. test.swp in the file name box with a .swp extension. Click OK to start saving.



File name: *.swp

- demo1c.swp
- demo1cv.swp
- demo2c.swp
- demo2cv.swp
- demo3c.swp
- demo4c.swp
- demo5c.swp

Folders: c:\syswin33\samples

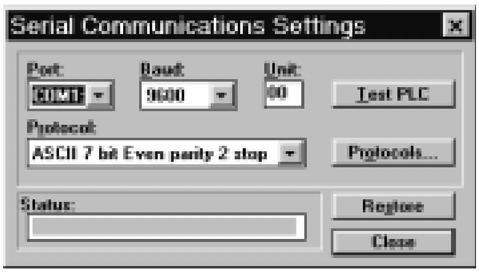
- c:\
- syswin33
- samples

List files of type: Program files (swp)

Drives: c: drivec

Transfer Program to PLC

Before down loading program to PLC, first check the connection of the PLC to your PC. Next, select Communication under the Project menu to setup the serial communication setting of the PLC to PC. Select the port of the PC where RS232C is connected, the baud rate for the communication, and the type of protocol to be used. Default setting of CPM1A is COM1, 9600 Baud, Unit 00, Protocol ASCII 7 bit Even Parity 2 stop. Click Test PLC to establish connection. Monitor the status bar to check whether connection is successful.



Serial Communications Settings

Port: COM1

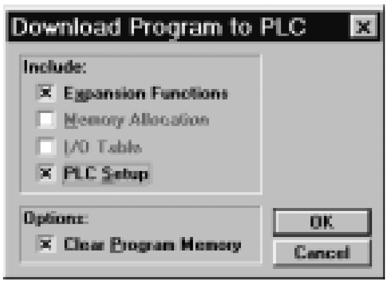
Baud: 9600

Unit: 00

Protocol: ASCII 7 bit Even parity 2 stop

Status:

Once connection established, click download under on–line menu to download the complete ladder diagram to PLC. Select whether to include the expansion function or memory allocation. It is usually prefer to clear program memory of PLC before downloading the program. Click OK upon confirmation.



Download Program to PLC

Include:

☒ Expansion Functions

☐ Memory Allocation

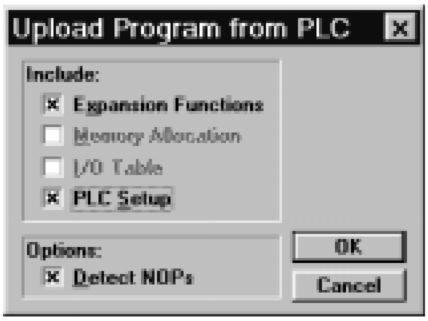
☐ I/O Table

☒ PLC Setup

Options:

☒ Clear Program Memory

Uploading of program from PLC to PC is also possible. Select detect NOPs (no operation) for the options as shown.



Upload Program from PLC

Include:

☒ Expansion Functions

☐ Memory Allocation

☐ I/O Table

☒ PLC Setup

Options:

☒ Detect NOPs

Finally when uploading or downloading completed, a download or upload successful dialog box will appear.

Note: All application examples in this manual can either be programmed using window based SYSWIN Software or the Programming Console.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Overview of Instructions

The CPM1A has a large selection of programming instructions that allows for easy programming of complicated control processes. The instructions explained in this section includes ladder diagram symbols and the mnemonic codes for the instructions. Application examples are also provided.

The Command Keys.

FUN	*Numerous special Application command called FUNctions may be input using this key.*
LD ├┤├	*LoaDs points into the program, and provides a means of branching points.*
AND ┤├	*AND allows points to be connected together, to form a series circuit*
OR ┤├┬	*OR allows points to be Connected together, to Form parallel circuit*
OUT ─ ┤	*The OUTput commands.*
TIM	*The TIMers are control using this command.*
CNT	*The COUNTers are controled using this command.*
NOT	*Using with LD, AND, and OR command keys to designate NC (normally closed) points.*
HR	*Designates Holding Relays.*
HR	*Designates Temporary Relays.*
SFT	*Displays the operations of the SHIFT register.*
SHIFT	*SHIFT is used to obtain the Alternate function of the Four keys with more than One used, labelled, Channel, And CONTact.*
A 0 ~ **9**	*Used to input decimal or hexadecimal numbers when programming*

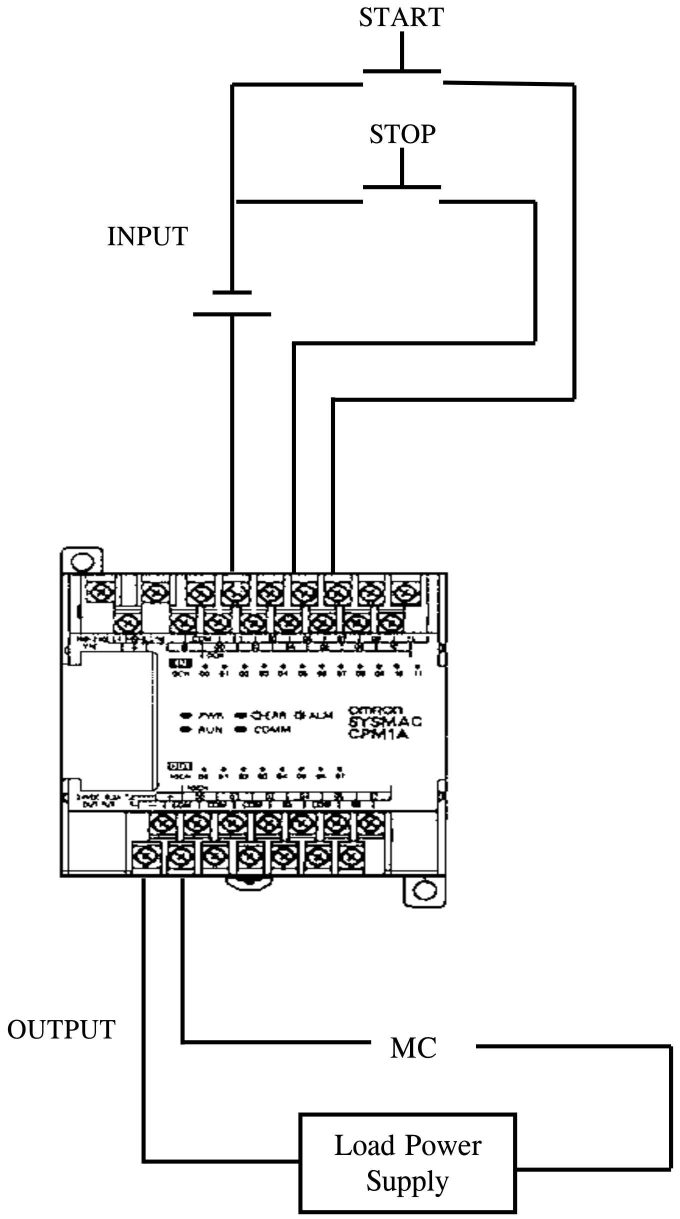
6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Application #1: Self Holding Circuit

Input	Devices
00000	Start Pb.
00001	Stop Pb.

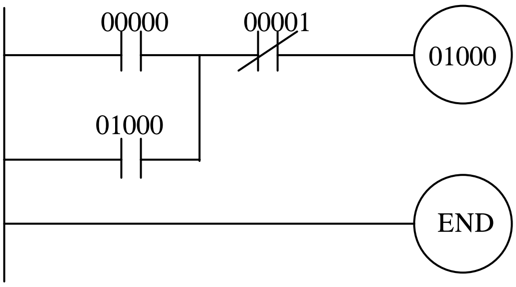
Output	Devices
01000	Motor

Circuit Diagram



6. APPLICATION EXAMPLES OF BASIC COMMANDS

Ladder Diagram



Mnemonic Codes

Address	Instruction	Data
0000	LR	00000
0001	OR	01000
0002	AND NOT	00001
0003	OUT	01000
0004	END(01)	

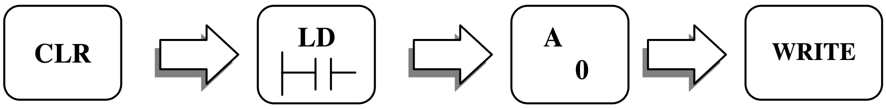
Note: This program enables the output 01000 to stay "ON" despite the status of the input 00000 which triggered it. This is often desirable in continuous output operations which begin with only an instantaneous ON input.

❑ Programming With Program Console

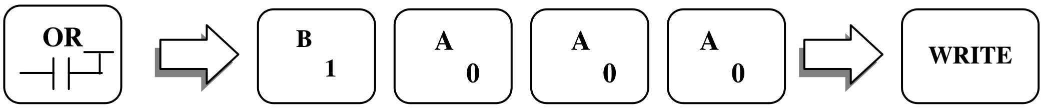
• PROGRAM ENTRY PROCEDURE

1. Set the MODE SELECTOR switch to PROGRAM mode. Press CLR, if necessary, until 0000 appears on the display.

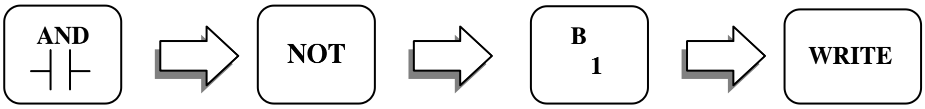
2.



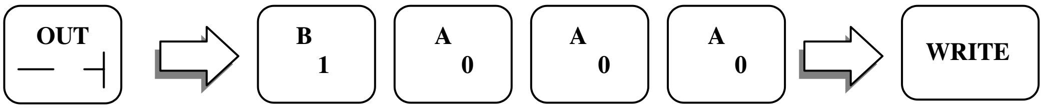
3.



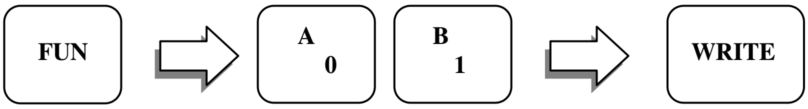
4.



5.

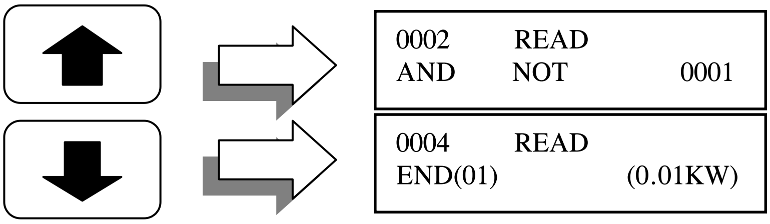


6.



Note: All instructions will only be stored in the PLC memory after the **WRITE** key is pressed and every program must have an END (FUN 01) instruction. All examples given in this manual will follow the same procedure of entry and henceforth the program entry procedure will not be mention again.

To view the program steps, use either the UP ARROW pr DOWN ARROW keys.



```
0002      READ
AND       NOT         0001
```

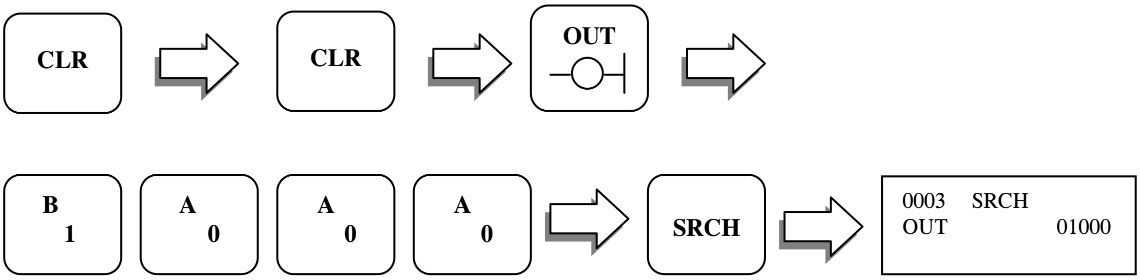
```
0004      READ
END(01)           (0.01KW)
```

- **RUNNING THE PROGRAM**

To execute the program that has stored in the PLC, set the Mode Selector switch to RUN or MONITOR mode.

- **RECALLING A PROGRAM**

To search for and confirm a particular command in program, perform the following keystrokes on the console.



```
0003    SRCH
OUT             01000
```

This confirms the OUT 01000 instruction has been written into the PLC memory address 00003.

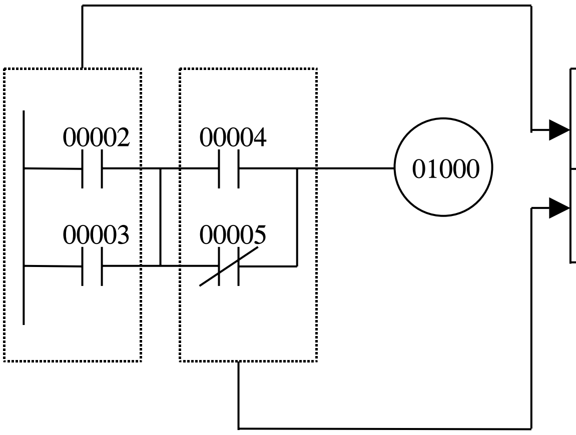
Note: If the above operations are performed during MONITOR or RUN mode, the state of each address can be monitored.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ AND LD and OR LD instructions

AND LD connects two blocks in series

Ladder diagram

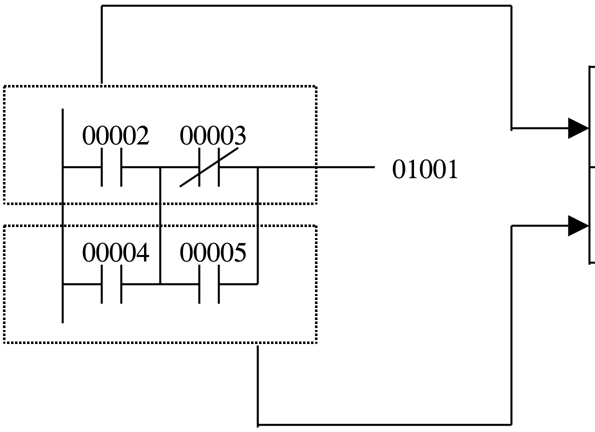


Mnemonic codes

Address	Instruction	Data
00000	LD	00002
00001	OR	00003
00002	LD	00004
00003	OR-NOT	00005
00004	AND-LD	-----
00005	OUT	01000

OR LD connects two blocks in parallel

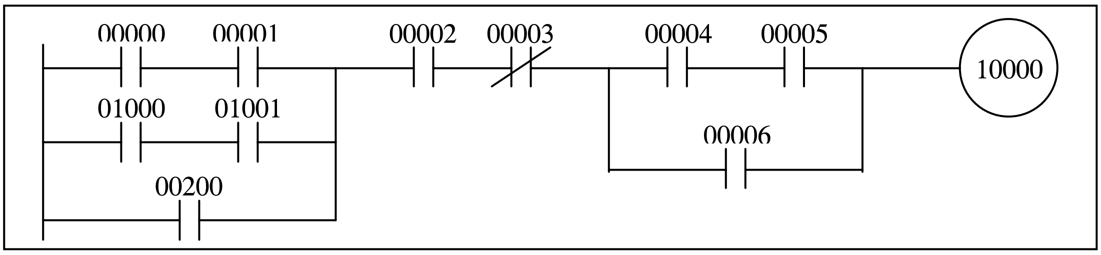
Ladder diagram



Mnemonic codes

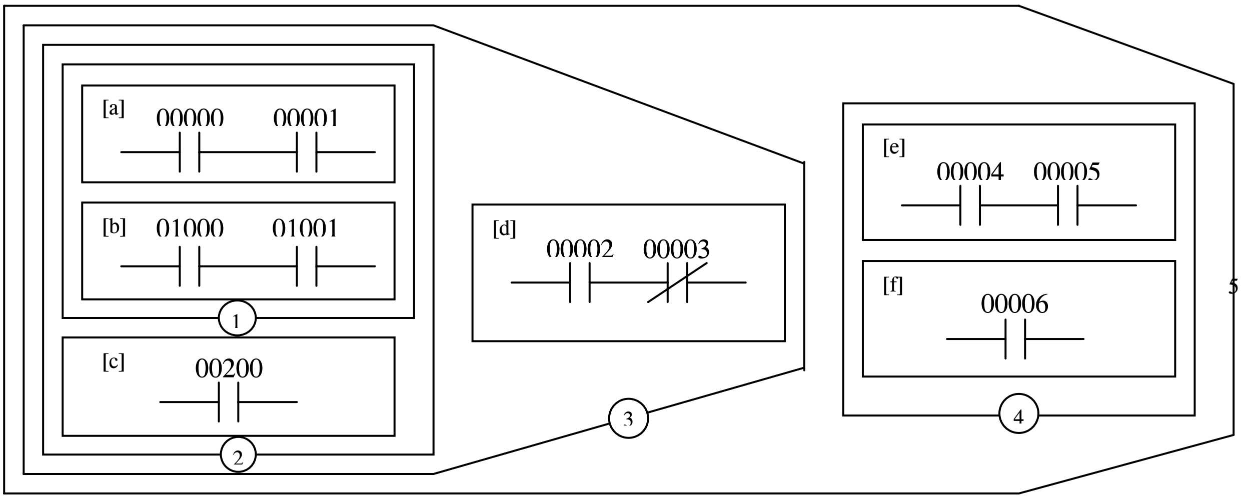
Address	Instruction	Data
00000	LD	00002
00001	AND-NOT	00003
00002	LD	00004
00003	AND	00005
00004	OR-LD	-----
00005	OUT	01001

Program Organisation

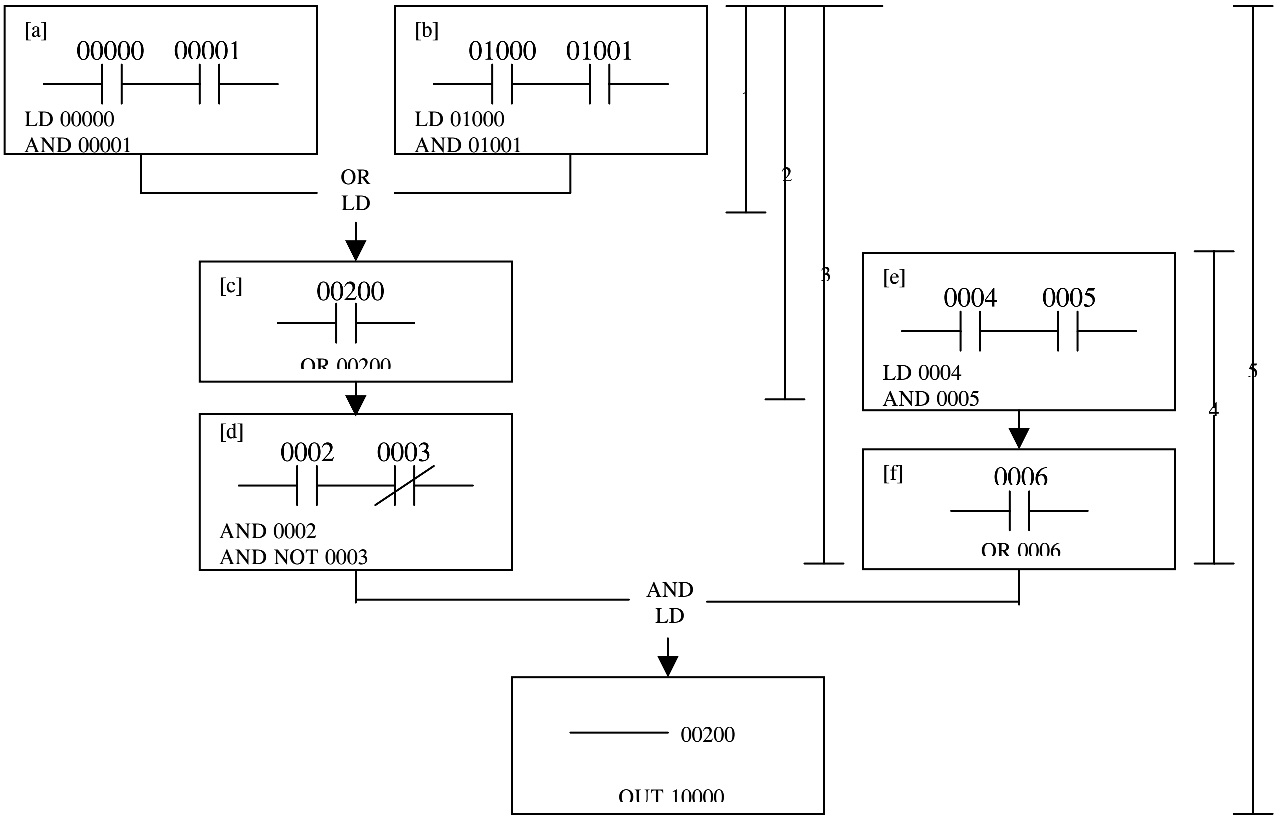


To organize the code for the circuit shown above

(1) Divide the circuit into smaller blocks [a] to [f]



(2) Program each block from top to bottom, then from left to right



```
LD 00000
AND 00001
```

```
LD 01000
AND 01001
```

```
OR 00200
```

```
AND 0002
AND NOT 0003
```

```
LD 0004
AND 0005
```

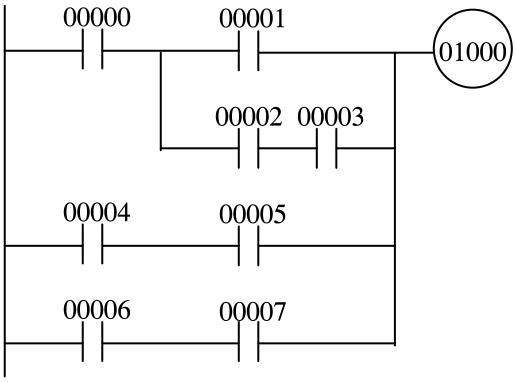
```
OR 0006
```

```
OUT 10000
```

Review Questions

Convert the ladder diagram given below to its equivalent mnemonic codes.

Ladder diagram



Mnemonic codes

Address	Instruction	Data

6. APPLICATION EXAMPLES OF BASIC COMMANDS

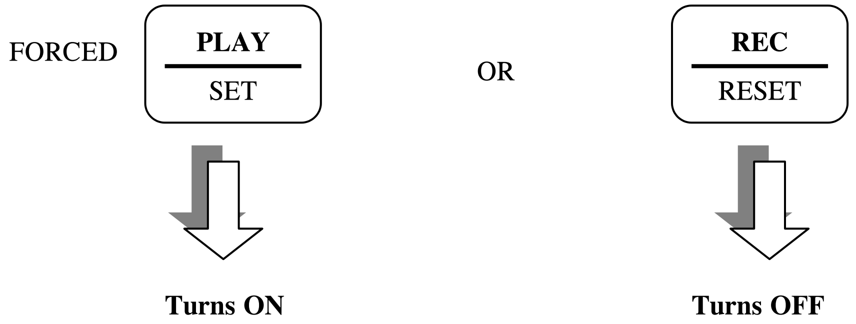
❑ To Check For Proper Connection Of External Device To PLC

a) INPUT devices to PLC
Simply TURN-ON input manually. Input indicator will show functionality of device.

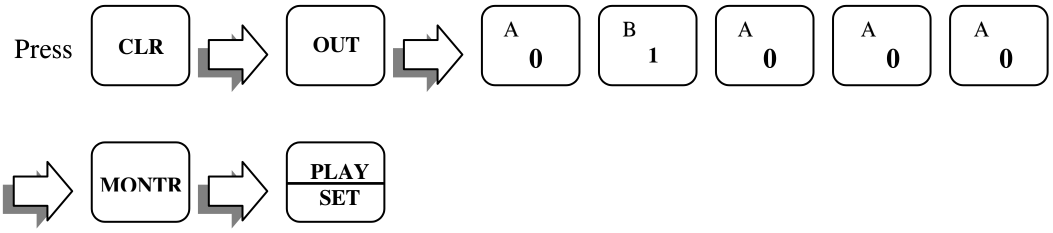
b) OUTPUT devices to PLC
Use FORCED SET/RESET command.

❑ Forced Set/Reset

Forcing outputs or other parts of the program ON or OFF irrespective to the Program is referred to as FORCED SET/RESET.



Using a Programming Console, Set the Mode Selector switch to MONITOR



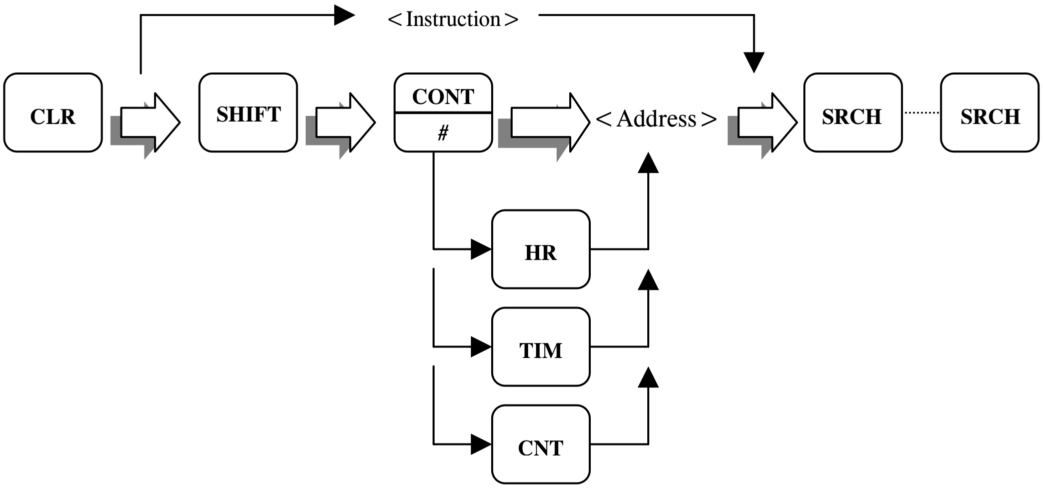
Output 01000 LED should be lighted.

- The above can also be done in the PROGRAM mode.

❑ Program Search

The program can be searched for occurrences of any designated instruction or data area bit address used in instruction. Searches can be performed from any currently displayed address or from a cleared display.

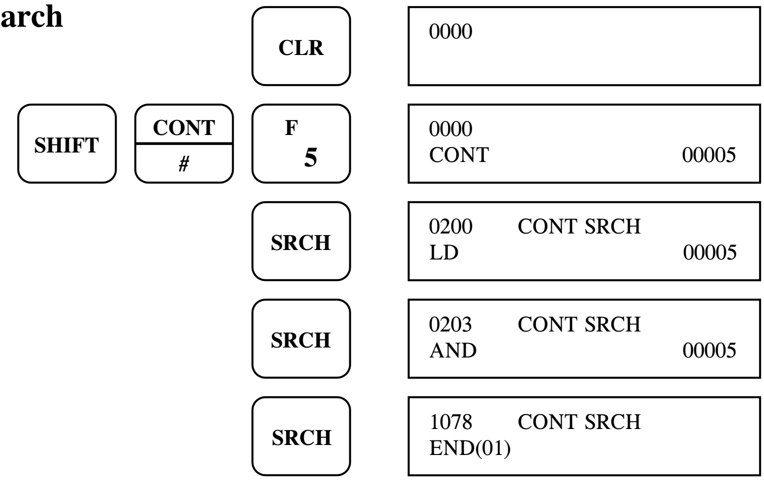
To designate a bit address or an instruction,



Once an occurrence of an instruction or bit address has been found, any additional occurrences of the same instruction or bit can be found by pressing SRCH again. SRCH'G will be displayed while a search is in progress. When the first word of a multiword instruction is displayed for a search operation, the other words of the instruction can be displayed by pressing the DOWN key before continuing the search.

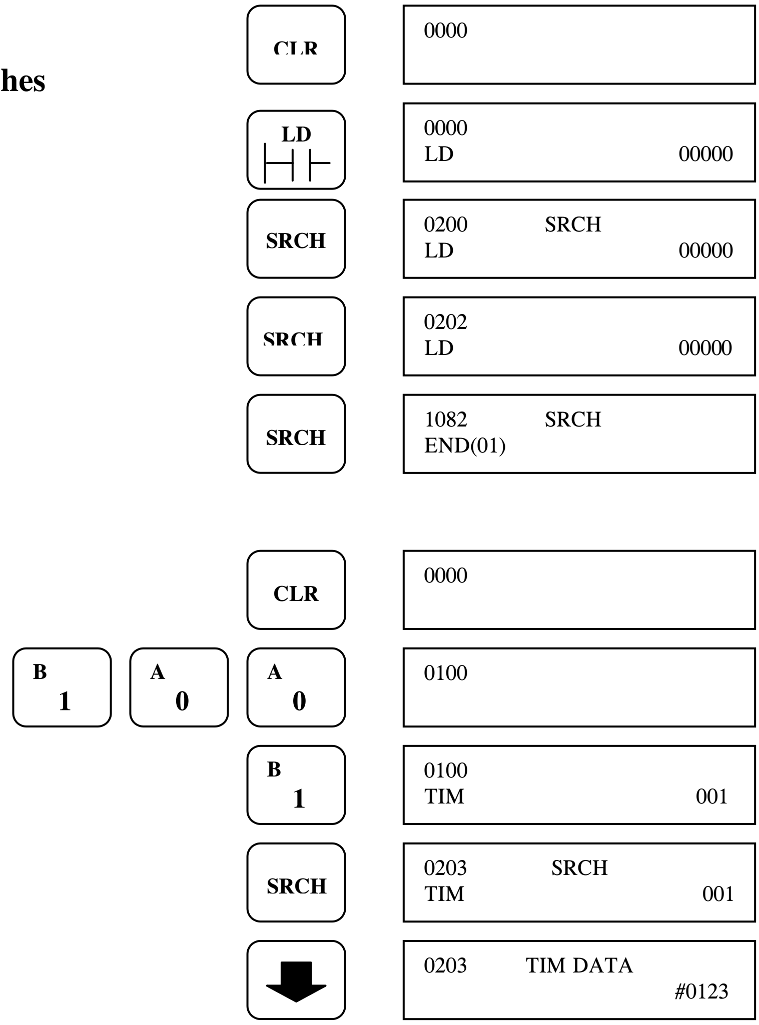
If Program Memory is read in RUN or MONITOR mode, the ON/OFF status of any displayed will also be shown.

- **Example: Bit Search**



```
CLR                  0000

SHIFT  CONT/#  F 5   0000
                     CONT                 00005

SRCH                 0200     CONT SRCH
                     LD                   00005

SRCH                 0203     CONT SRCH
                     AND                  00005

SRCH                 1078     CONT SRCH
                     END(01)
```

- **Example:**
  Instruction Searches



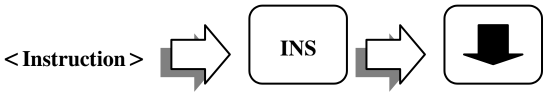
```
CLR                  0000

LD                   0000
                     LD                   00000

SRCH                 0200        SRCH
                     LD                   00000

SRCH                 0202
                     LD                   00000

SRCH                 1082        SRCH
                     END(01)
```

```
CLR                  0000

B 1   A 0   A 0      0100

B 1                  0100
                     TIM                    001

SRCH                 0203         SRCH
                     TIM                    001

↓                    0203      TIM DATA
                                          #0123
```

❑ Inserting And Deleting Instruction

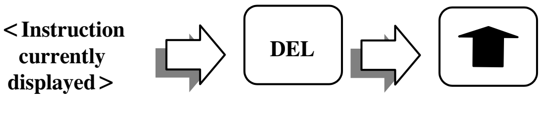
In PROGRAM mode, any instruction that is currently displayed can be deleted or another instruction can be inserted before it. These are not possible in RUN or MONITOR modes.

To insert an instruction, display the instruction before which you want the new instruction to be placed, input the instruction in the same way as when inputting a program initially, and then press INS and the DOWN key.



To delete an instruction, display the instruction to be deleted and press DEL and the UP key.

CAUTION: Be careful not to delete instructions inadvertently. Once deleted, there is no way to recover without reinputting the instruction again.

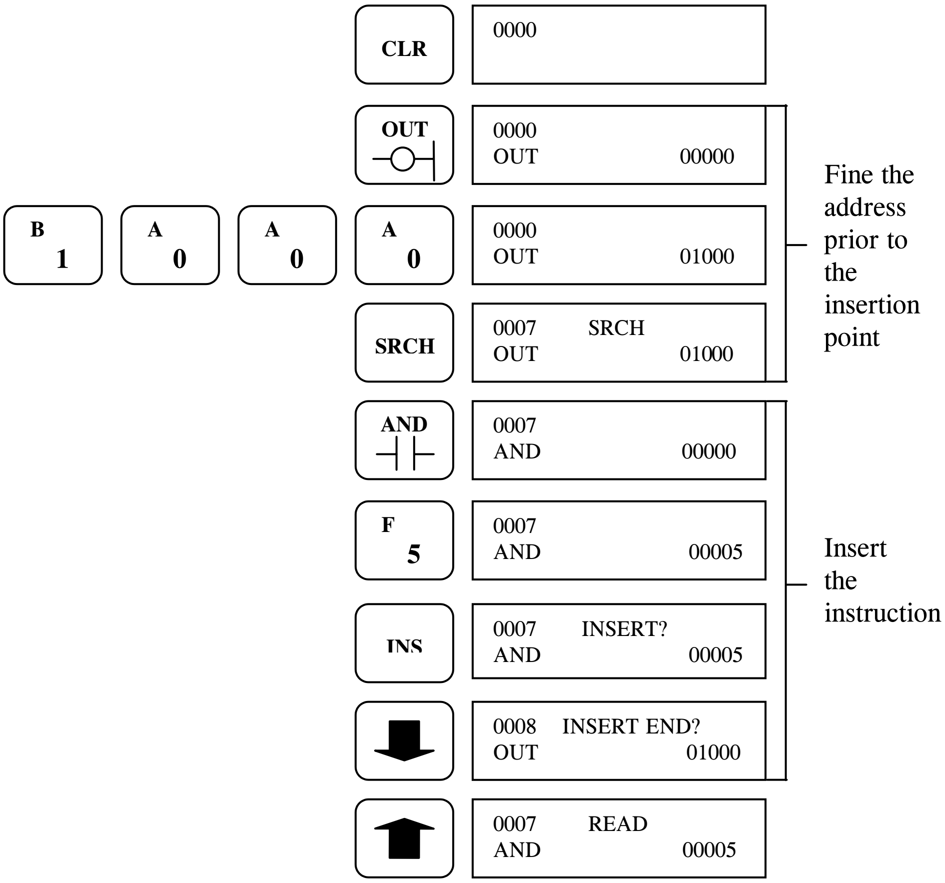


When an instruction(s) is inserted or deleted, all address in Program Memory following the operation are adjusted automaticaly so that there are no blank address and no unaddressed instructions.

- **Example :** The following mnemonic code shows the changes that are achieved in a program through insertion and deletion.

Before Insertion

Addr	Instruction	Data
0000	LD	00000
0001	AND	00001
0002	LD	01000
0003	AND NOT	00002
0004	OR LD	
0005	AND	00003
0006	AND NOT	00004
0007	OUT	01000
0008	END(01)	



After Insertion

Addr	Instruction	Data
0000	LD	00000
0001	AND	00001
0002	LD	01000
0003	AND NOT	00002
0004	OR LD	
0005	AND	00003
0006	AND NOT	00004
0007	AND	00005
0008	OUT	01000
0009	END(01)	

- **Example : (con't)**

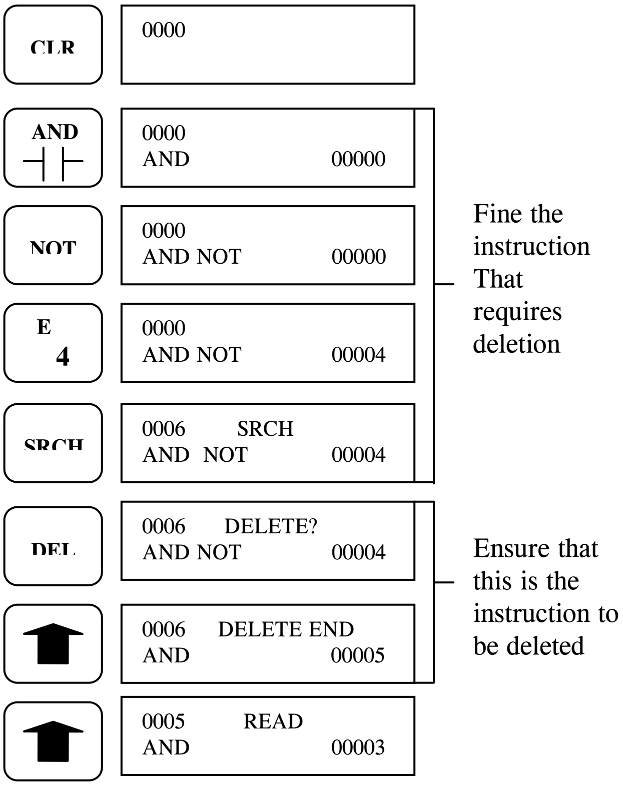
Before Deletion

Addr	Instruction	Data
0000	LD	00000
0001	AND	00001
0002	LD	01000
0003	AND NOT	00002
0004	OR LD	
0005	AND	00003
0006	AND NOT	00004
0007	AND	00005
0008	OUT	01000
0009	END(01)	

After Deletion

Addr	Instruction	Data
0000	LD	00000
0001	AND	00001
0002	LD	01000
0003	AND NOT	00002
0004	OR LD	
0005	AND	00003
0006	AND	00005
0007	OUT	01000
0008	END(01)	

Procedure for Deletion



CLR	0000
AND ┤├	0000 AND 00000
NOT	0000 AND NOT 00000
E 4	0000 AND NOT 00004
SRCH	0006 SRCH AND NOT 00004
DEL	0006 DELETE? AND NOT 00004
↑	0006 DELETE END AND 00005
↑	0005 READ AND 00003

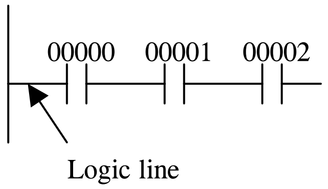
Fine the instruction That requires deletion

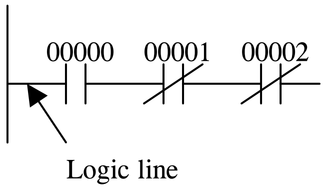
Ensure that this is the instruction to be deleted

6. APPLICATION EXAMPLES OF BASIC COMMANDS

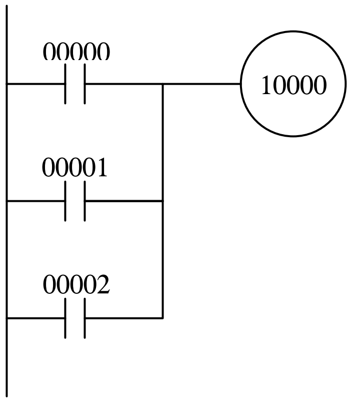
❑ Number Of Contacts

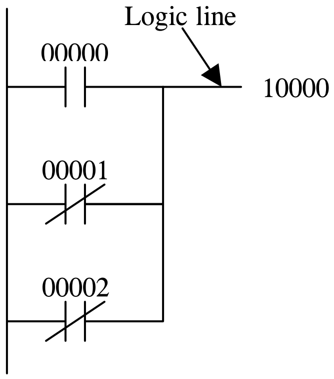
The number of contacts used in a rung is limited only by memory size. As many contacts as required can be connected by means of the AND and the AND NOT keys.





As many contacts as required can be connected by means of the OR and the OR NOT keys. The number of contacts is also not limited for use on a logic line.

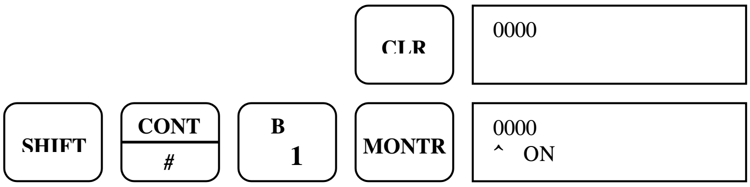




❑ Bit Monitor

Monitor the status of a particular bit. It is possible in any PLC operation mode.

Example :



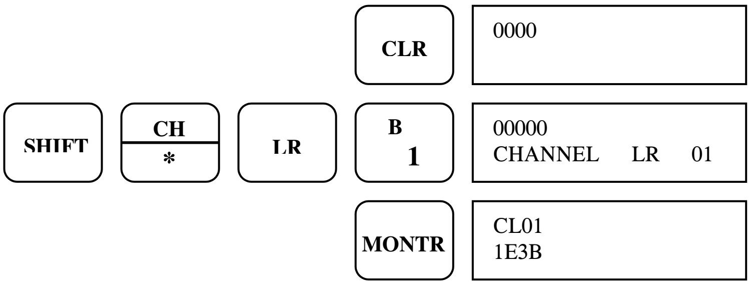
Note : a) Press the ⬆ or ⬇ key to display the status of the previious or next bit

b) If the PLC is in PROGRAM or MONITOR mode, the bit's status can be changed using the FORCE SET / RESET operation.

❑ Word Monitor

Monitor the content of a particular word. It is possible in any operation mode.

Example :



Note : a) Press the ⬆ or ⬇ key to display the status of the previious or next word.

b) If the PLC is in PROGRAM or MONITOR mode, the word's content can be changed using the HEX/DEC Data Modification operation. (See page 72)

❑ Multiply Address Monitoring

The status / contents of up to 6 bits and words can be monitored simultaneously, although only 3 can be shown on the display at any one time.

Example :

Step 1				CLR	0000
Step 2		TIM	B 1	MONTR	T001 0100
Srep 3	SHIFT	CONT #	B 1	MONTR	00001 T001 ^ OFF 0100
Step 4	DM	B 1	A 0	MONTR	D0010 00001 T001 1 2 3 4 ^ OFF 0100

a) If 4 or more bits and words are being monitored, the bits and words that do not appear on the display can be displayed by pressing the MONTR key. If the MONTR key is pressed alone, the display will shift to the right.

If more than 6 bits and words are monitored, monitoring of the first bit or word input will be cancelled.

c) Press the CLR key to stop monitoring the leftmost bit or word and clear it from the display.

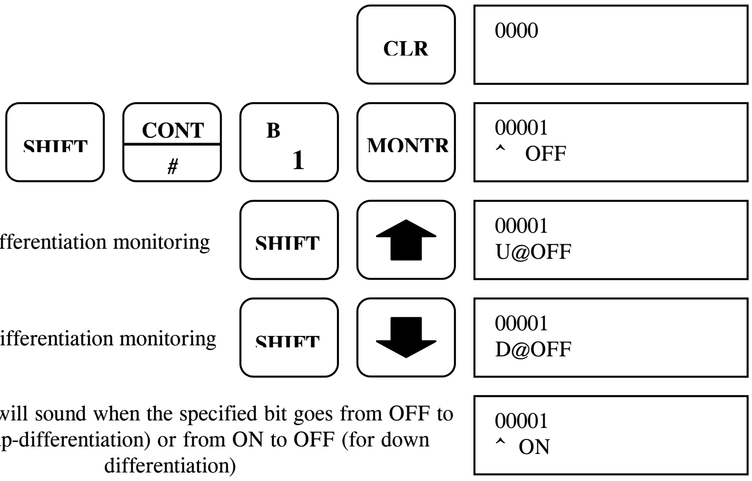
d) Press the SHIFT + CLR key to end monitoring altogether.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Differentiation Monitor

Monitor the up or down differentiation status of a particular bit. It is possible in any PLC operation mode. (Differential mode is where a contact is turned On/Off for one scan only).

Example :



Up-differentiation monitoring

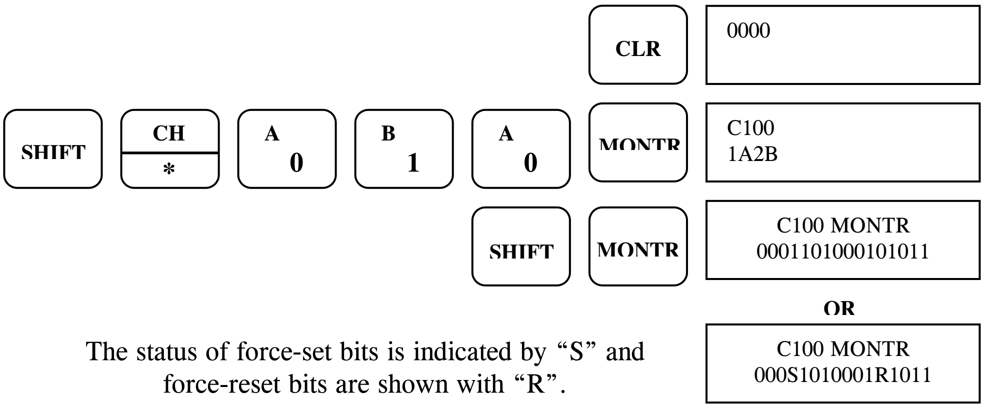
Down-differentiation monitoring

The buzzer will sound when the specified bit goes from OFF to ON (for up-differentiation) or from ON to OFF (for down differentiation)

❑ Binary Monitor

Monitor the ON/OFF status of any word's 16 bits. It is possible in any PLC operating mode.

Example :



The status of force-set bits is indicated by "S" and force-reset bits are shown with "R".

Note : a) The status of displayed bit can be changed at this point by using Binary Data Modification.

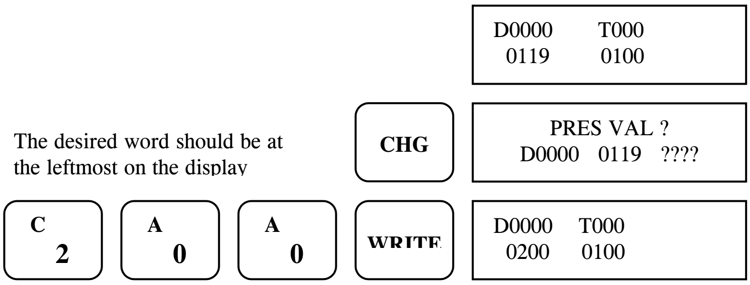
b) Press ↑ or ↓ key to display the status of the previous or next word's bits.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Hex/BCD Data Modification

This operation is used to change the BCD or Hexadecimal value of a word being monitored. It is possible in the MONITOR or PROGRAM mode only.

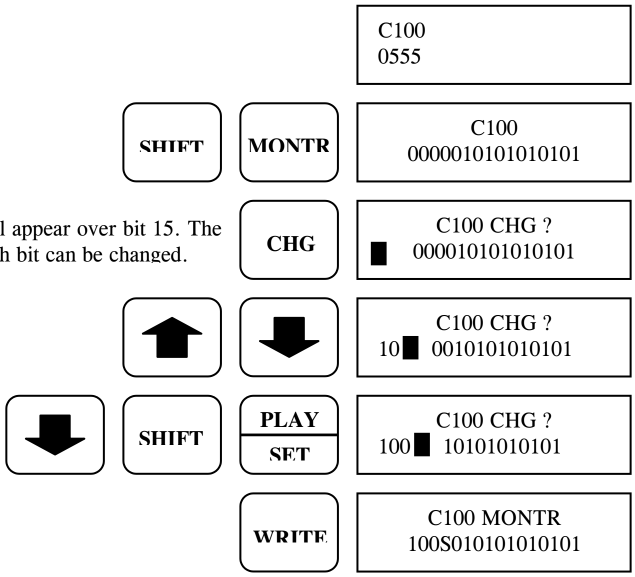
Example :


The desired word should be at the leftmost on the display

❑ Binary Data Modification

This operation is used to change the status of a word's bits when the word is being monitored in 16-bit format, using the Binary monitor. It is possible in MONITOR or PROGRAM mode only.

Example :


A flashing cursor will appear over bit 15. The cursor indicates which bit can be changed.

Note : a) Use the ⬆ and ⬇ keys to move the cursor to the left and right.

b) Use the B 1 and A 0 keys to change a bit's status to On or Off. The cursor will move one bit to the right after one of the keys is pressed.

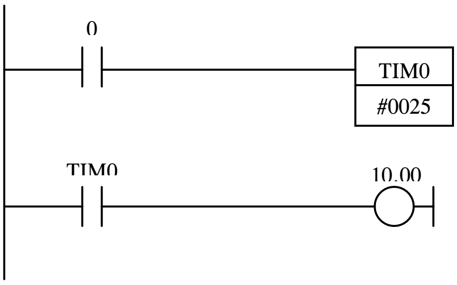
c) Use the SHIFT + PLAY/SET and SHIFT + REC/RESET keys to force-set or force-reset a bit's status. The NOT key will clear force-set or force-reset status.

❑ Timers

Timers are nomally used for time delay. It can be ON delay, OFF delay etc.

❑ Application #2: On Delay Circuit

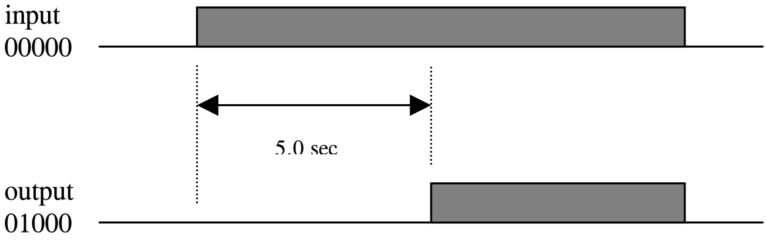
Ladder diagram



Mnemonic codes

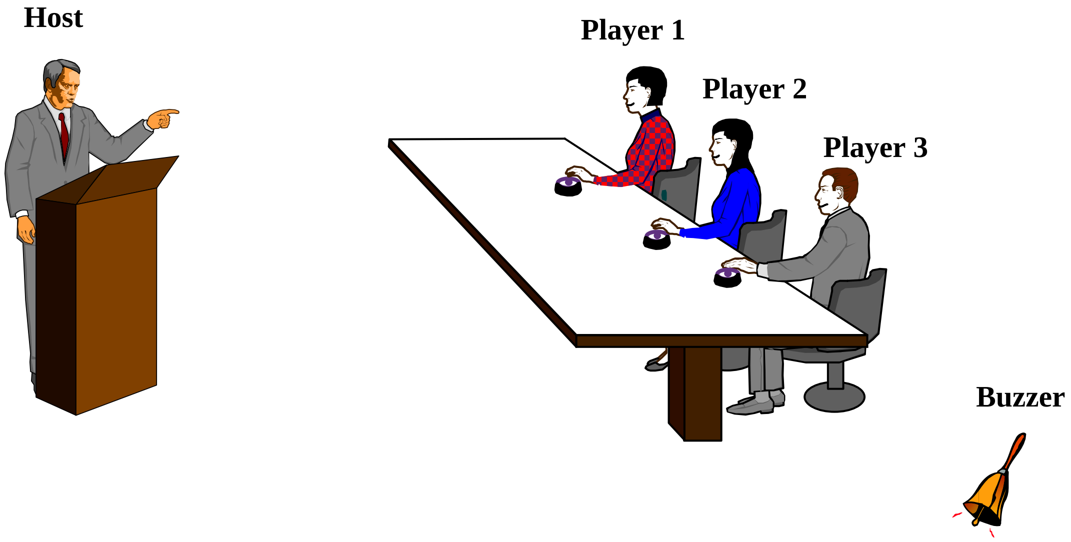
Address	Instruction	Data
00000	LD	00000
00001	TIM	000
		#0025
00002	LD	TIM 000
00003	OUT	01000
00004	END(01)	

Timing diagram



TIM is a decrementing ON-delay timer instruction which requires a timer number and a set value (SV) ranging from 0000 to 9999 (0 to 999.9 seconds).

❑ Example : Priority Determination Design



The game buzzer control requirement:

1. After the Host has finished with question.

2. The 3 players will press the switch in front of them to fight to be first to answer the question.

3. The buzzer will sound for 10 sec after any one of the players has touched the switch.

4. The light indicator in front of each player will light-up and only reset by the Host switch.

- **I/O Assignment**

Input	Device
00000	PB1
00001	PB2
00002	PB3
00003	RST (reset)

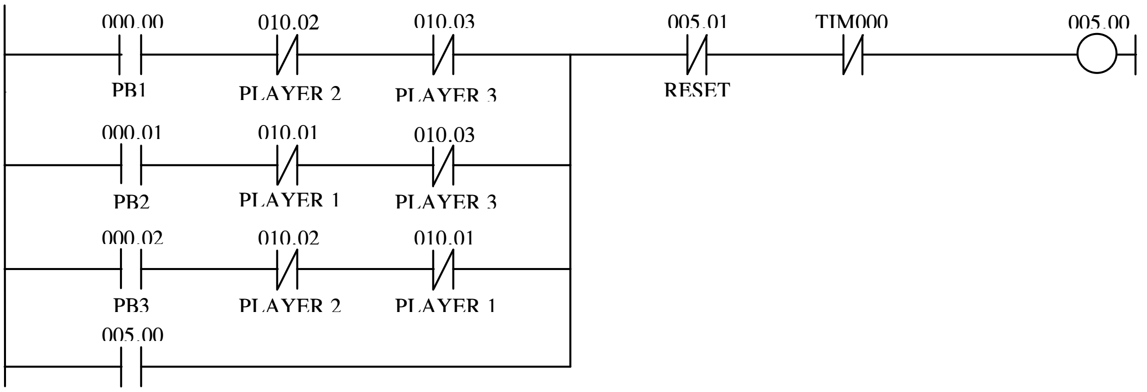
Output	Device
01000	Buzzer
01001	Player 1 light
01002	Player 2 light
01003	Player 3 light

By Using SYSWIN Programming Software, the design of the buzzer control ladder diagram is shown as follow.

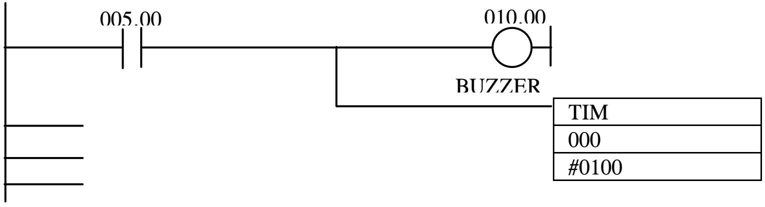
Ladder Diagram : Main 1 Network 1

Main 1 – Who press first
(Priority Determination)
This prigram to determine which player press the switch first, after the host have finish asking a question.

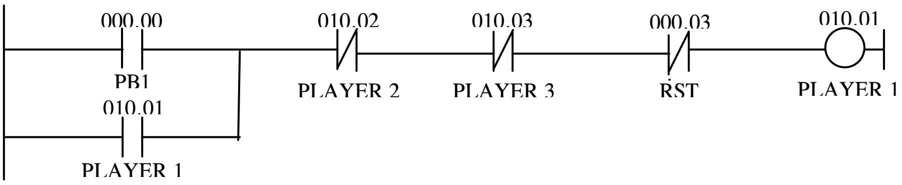
Network 1 – Interlocked
Interlocked network for the 3 player playing the game.



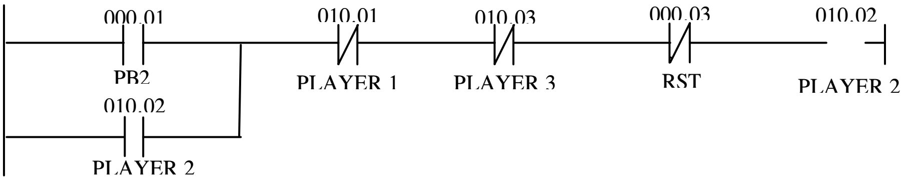
Network 2 – Buzzer
ON Buzzer when any switch is pressed and timer will cut the buzzer after specified time



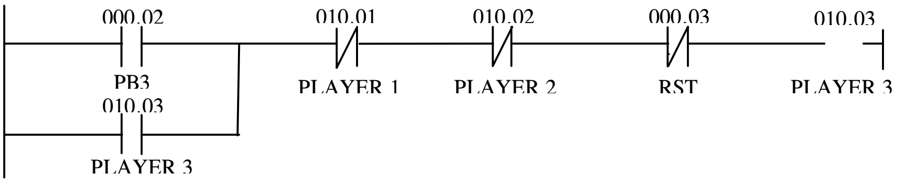
Network 3 – Player 1
Player 1 Network



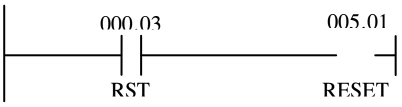
Network 4 – player 2
Player 2 Network



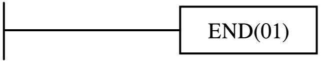
Network 5 – Player 3
Player 3 Network



Network 6 – Reset
Reset for the Game



Network 7



6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ On and Off Delay Circuit

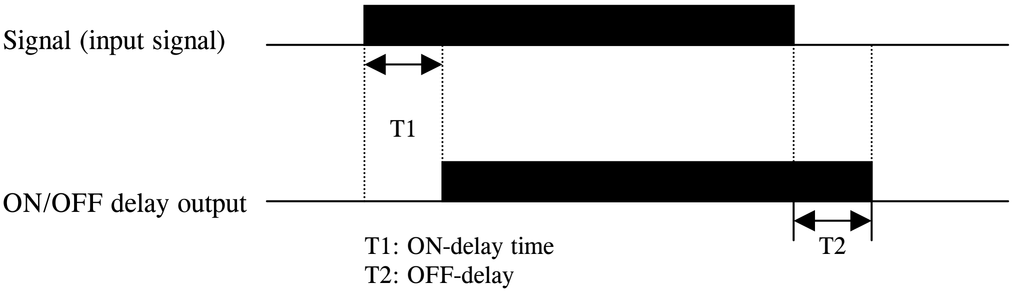
This circuit is used to delay the ON/OFF time of an input signal for a given time.

I/O Assignment

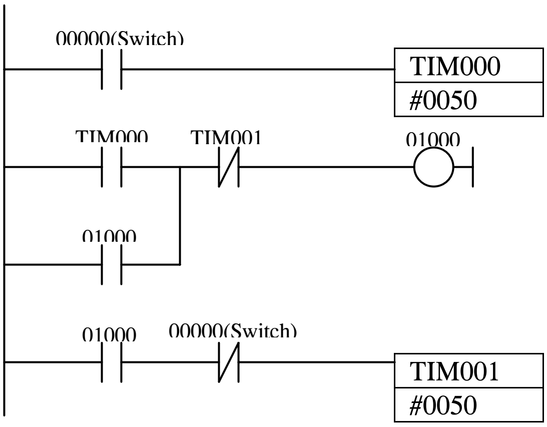
Input	Devices
00000	Switch

Output	Devices
01000	Motor

Timing diagram



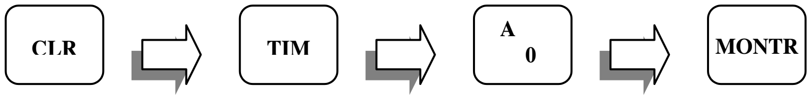
Ladder diagram



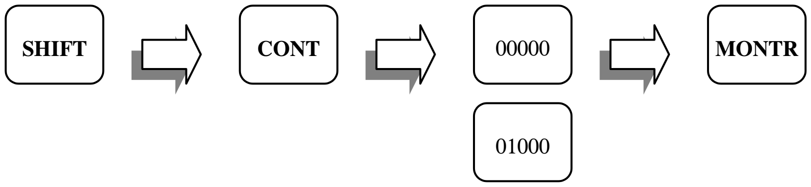
Address	Instruction	Data
00000	LD	00000
00001	TIM	000
		#0050
00002	LD	TIM000
00003	OR	01000
00004	AND-NOT	TIM0001
00005	OUT	01000
00006	LD	01000
00007	AND-NOT	00000
00008	TIM	001
		#0050
00009	END(01)	

- **Note :**

The timer's present value can be monitored from the programming console by pressing:

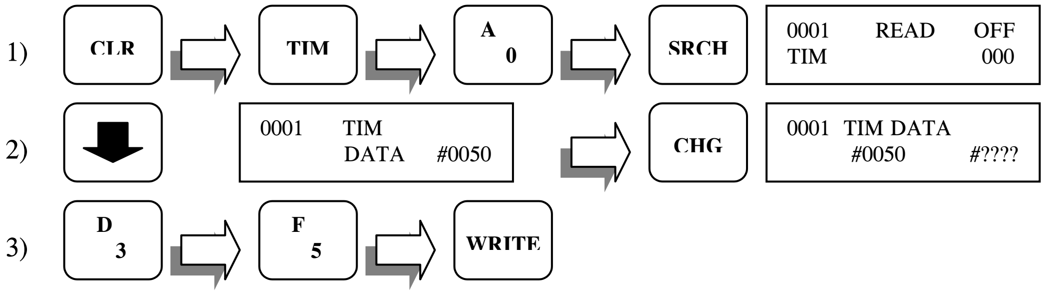


Input and output status can also be monitored by pressing:

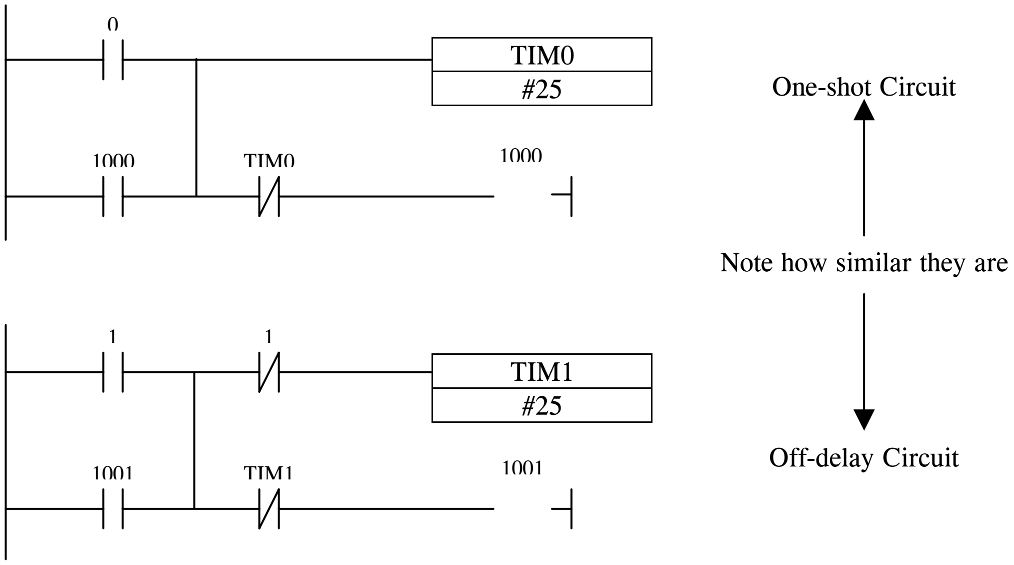


- **To change the timer value**

Set the mode selector switch to MONITOR



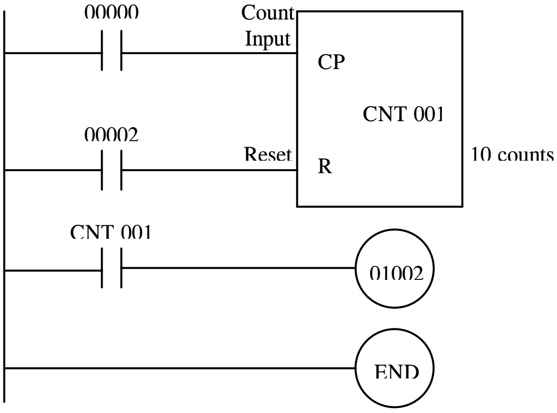
- **Off delay circuit**



❑ Counter

CNT (counter) is a preset decremental counter. That is, itdecrements one count everytime an input signal goes from PFF to ON. The counter must be programmed with a count input, a reset input, a counter number and a set valie (SV) can range from 0000 to 9999.

Ladder diagram

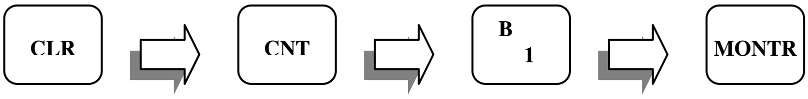


Mnemonic codes

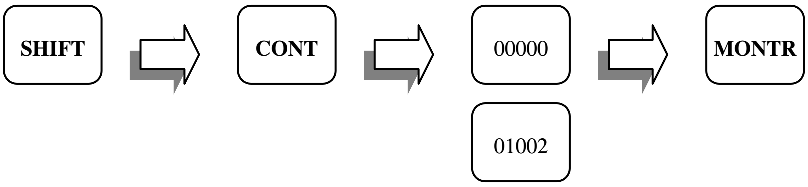
Address	Instruction	Data
0000	LD	00000
0001	LD	00002
0002	CNT	001
		#0010
0003	LD	CNT 001
0004	OUT	01002
0005	END(01)	

- **Note :**

The counter number must not be duplicated with the timer number since both share the same data area within the PLC memory. The counter;s present value can be monitored from the programming console by pressing:

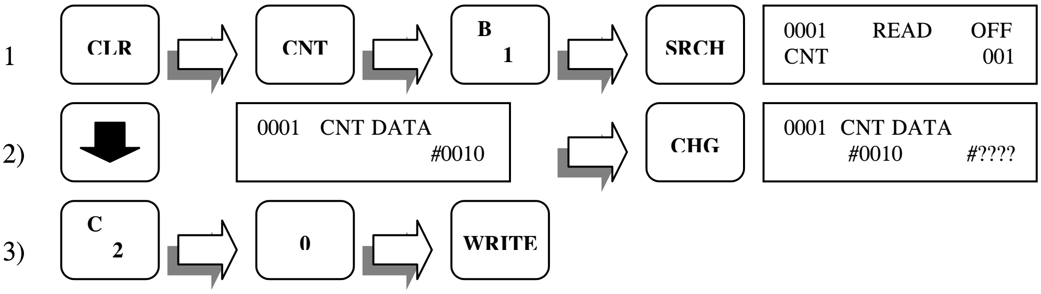


Input and output status can also be monitored by pressing:



- **To change the counter value**

Set the mode selector switch to MONITOR

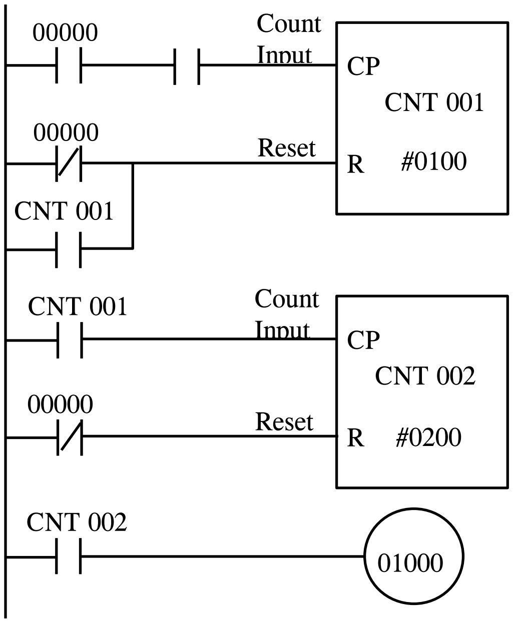


*Timer/Counter value can also be changed via SYSWIN Software

Use of CNT and TIM instructions

Example #1: Count to 20,000 counts

Ladder diagram

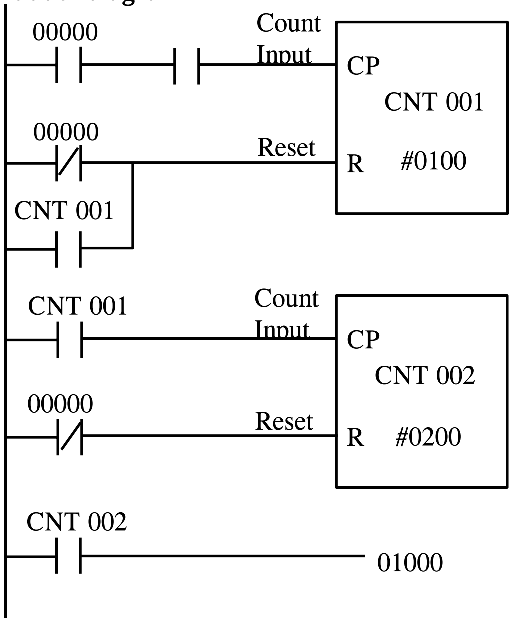


Mnemonic codes

Address	Instruction	Data
00000	LD	00000
00001	AND	00001
00002	LD-NOT	TIM 000
00003	OR	CNT 001
00004	CNT	001
		#0100
00005	LD	CNT001
00006	LD-NOT	00000
00007	CNT	002
		#0200
00008	LD	CNT002
00009	OUT	01000
00010	END(01)	

Example #2: Prolong time range to 1,000 hours

Ladder diagram



Mnemonic codes

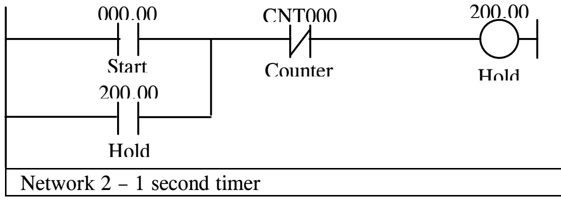
Address	Instruction	Data
00000	LD	00000
00001	AND-NOT	TIM 001
00002	TIM	001
		#6000
00003	LD	TIM001
00004	LD	00001
00005	CNT	002
		#6000
00006	LD	CNT 002
00007	OUT	01000

• Example : #3

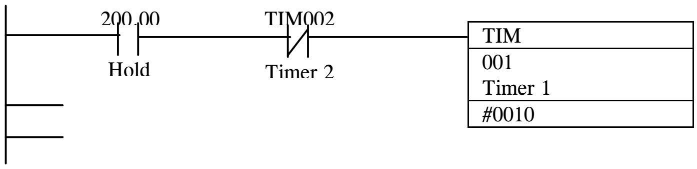
Ladder diagram : Main 1 Network 1

Main 1 – Flicker Output
This program makes output IR 01000 flicker ON/OFF (one second ON, one second OFF) ten times after input IR 00000 is turned ON.

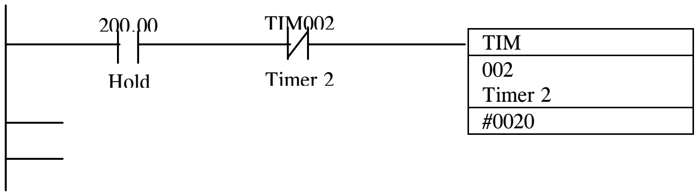
Network 1 – Self holding



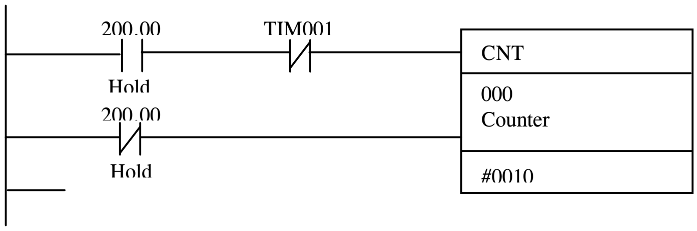
Network 2 – 1 second timer



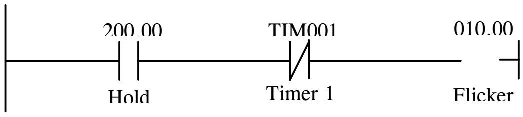
Network 3 – 2 second timer



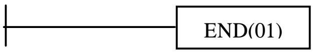
Network 4 – Counter
Preset Counter at 10



Network 5 – Flicker Output
Flicker for 10 times



Network 6 - End



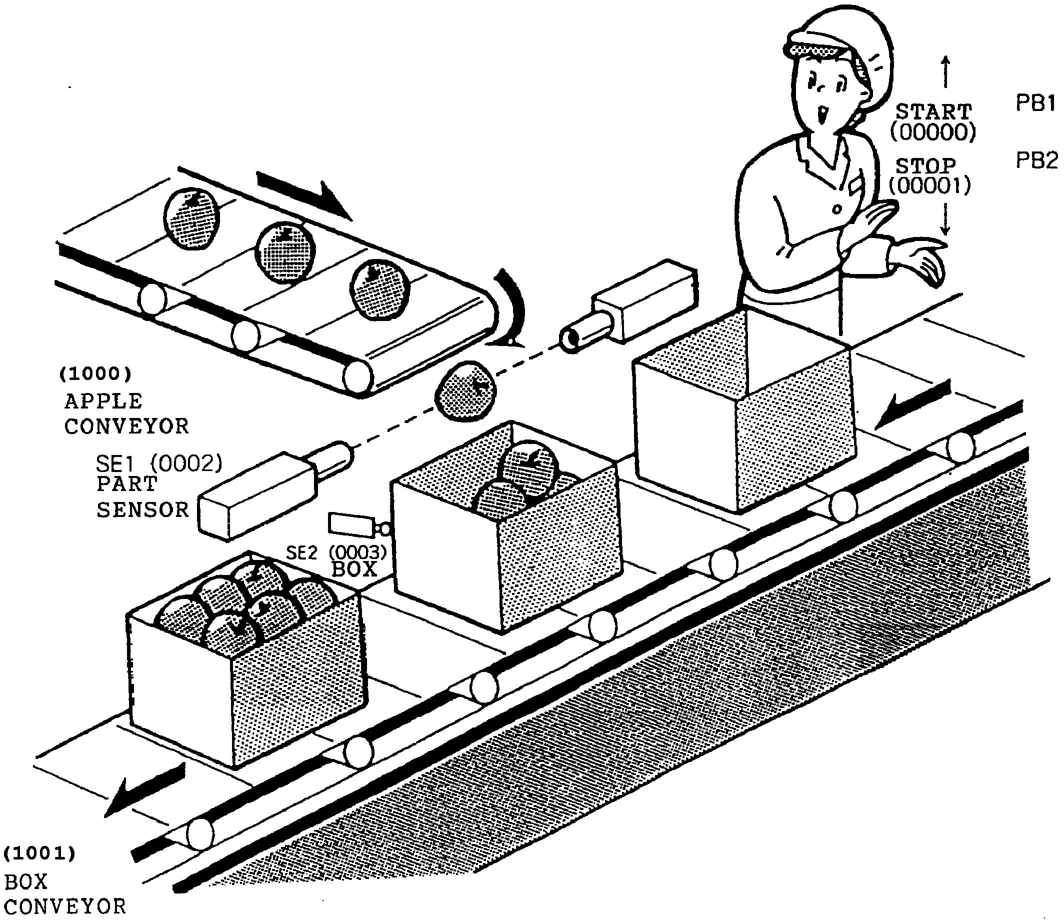
6. APPLICATION EXAMPLES OF BASIC COMMANDS

The mnemonic list for the example program is shown in the 000 following table.

Address	Instruction	Data	Remarks
00000	LR	00000	(1) Self-holding bit
00001	OR	20000	
00002	AND NOT CNT	000	
00003	OUT	20000	
00004	LD	20000	(2) 1-second timer
00005	AND NOT TIM	002	
00006	TIM	001	
	#	0010	
00007	LD	20000	(3) 2-second timer
00008	AND NOT TIM	002	
00009	TIM	002	
	#	0020	
00010	LD	20000	(4) 10-count counter
00011	AND NOT	TIM 001	
00012	LD NOT	20000	
00013	CNT	000	
	#	0010	
00014	LD	20000	(5) Flicker output (10 counts)
00015	AND NOT TIM	001	
00016	OUT	01000	
00017	END(01)	---	(6) END (001) instruction

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Example : Packaging Line Control

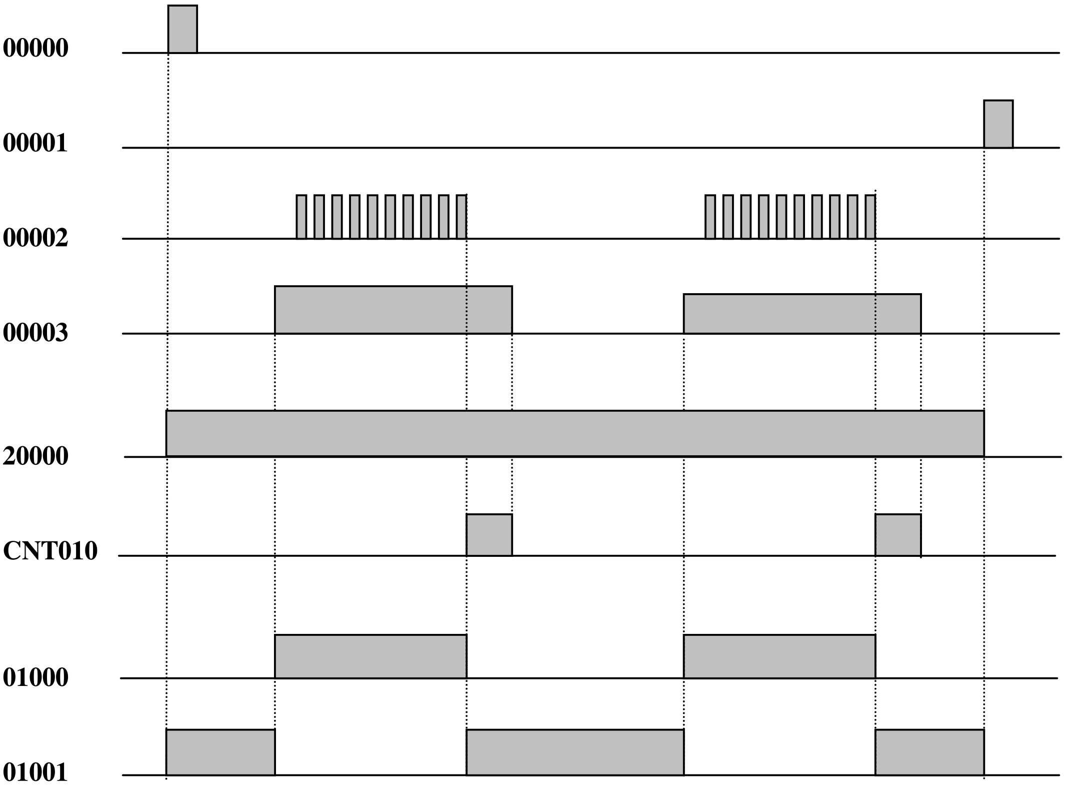


❑ Operation

When PB1 (START Push Botton) is pressed, the box conveyor moves. Upon detection of box present, the box conveyor stops and the Apple conveyor starts. Part sensor will count for 10 apples. Apple conveyor stops and box conveyor starts again. Counter will be reset and operation repeats until PB2 (STOP Push Button) is pressed.

Input	Devices
00000	START Push Button (PB1)
00001	STOP Push Button (PB2)
00002	Part Present (SE1)
00003	Box Present (SE2)

Output	Devices
01000	Apple Conveyor
01001	Box Conveyor

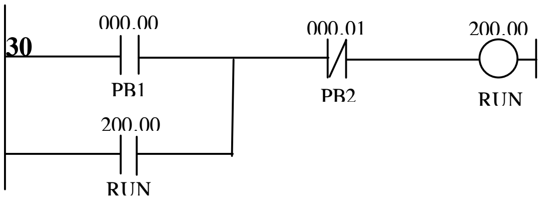


6. APPLICATION EXAMPLES OF BASIC COMMANDS

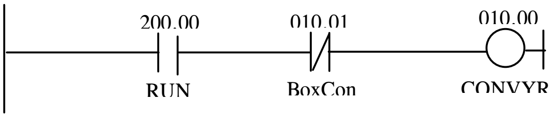
Ladder diagram : Main 1 Network 1

Main 1 – Packaging
Packaging line control for Apples

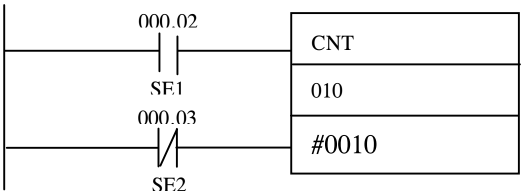
Network 1 – Start condition



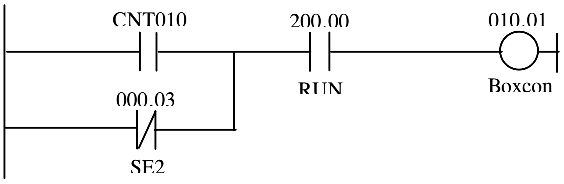
Network 2 – Apple conveyor



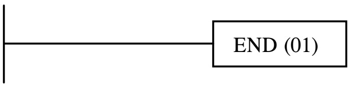
Network 3 – Counter
Counter preset at 10



Network 4 – Box conveyor



Network 5 - END

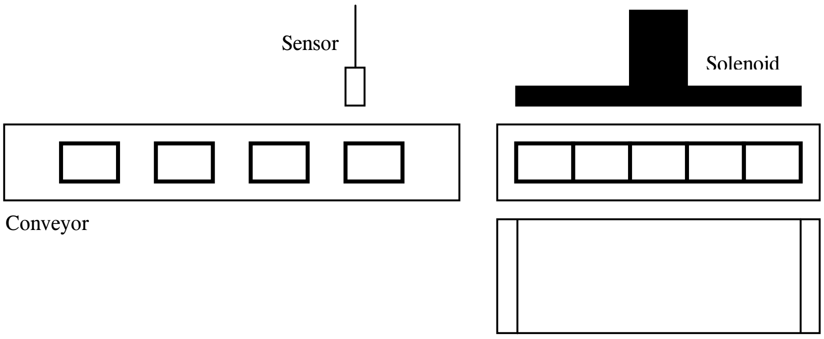


Address	Instruction	Data
0000	LD	00000
0001	OR	20000
0002	AND NOT	00001
0003	OUT	20000
0004	LD	20000
0005	AND NOT	01001
0006	OUT	01000
0007	LD	00002

Address	instruction	Data
0008	LD NOT	00003
0009	CNT	010
		#0010
0010	LD CNT	010
0011	OR NOT	00003
0012	AND	20000
0013	OUT	01001
0014	END (01)	

❑ Application #3 : Control Circuit For Packaging Machine

The control circuit is used to detect and count the number of products being carried on an assembly line. When it counts five products, the circuit energizes a solenoid. The solenoid is energized for a period of two seconds and is then shunt off, causing it to retract.

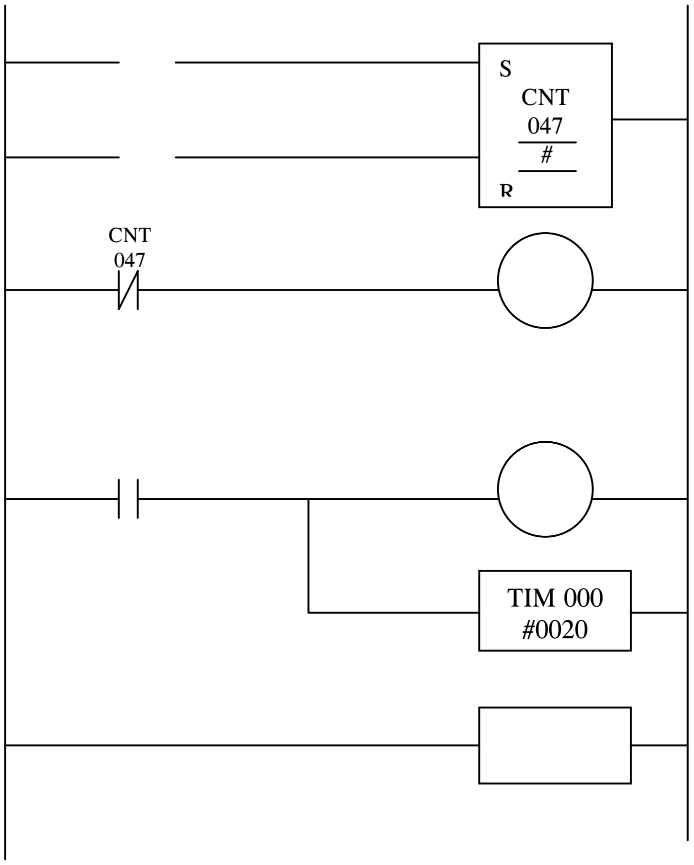


Input	Device
00000	Sensor

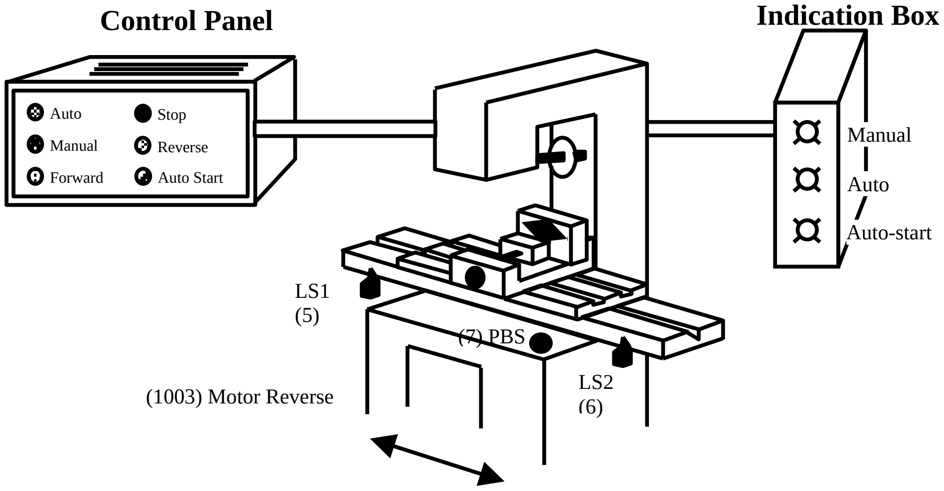
Output	Device
01000	Conveyor
01001	Solenoid

Others	
CNT 047	Product counter
TIM 000	Solenoid energizer timer

Question: Fill in the gaps below to make this circuit operate.



❑ Application #4 : Drilling Control Operation



- **I/O Assignments**

Input	Device
00000	Auto Switch
00001	Manual
00002	Forward Switch (SW1)
00003	Stop Switch (SW2)
00004	Reverse Switch (SW3)
00005	Limit Switch (LS1)
00006	Limit Switch (LS2)
00007	Auto Start Button (PBS)

Output	Device
1000	Auto Indicator
1001	Manual Indicator
1002	Motor Forward
1003	Motor Reverse
1004	Auto Start Indicator

- **Procedure**

1. **Manual Operation**
   1.1 When SW1 is ON, Motor moves forward. It can be stopped by SW2. When the drill touches LS2, the Motor is cut-off.
   1.2 When SW3 is ON, Motor moves in reverse. It can be stopped by SW2. When he drill touches LS1, the Motor is cut-off.

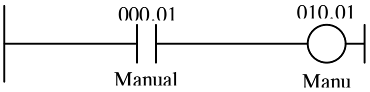
2. **Auto-cycle**
   2.1 When PB and LS1 is ON, the Motor moves forward until LS2 is activated. The Timer then starts timing down. The Motor reverses when the timer reaches 2 seconds. When it returns to LS1 position, the cycle is repeated.

Answer

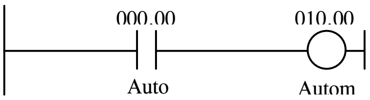
Ladder diagram : Main 1 Network 1

Main 1 – Drilling Control
Drilling control operation

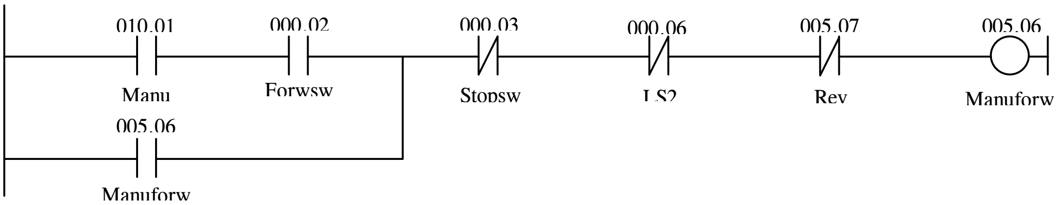
Network 1 – Manual indication



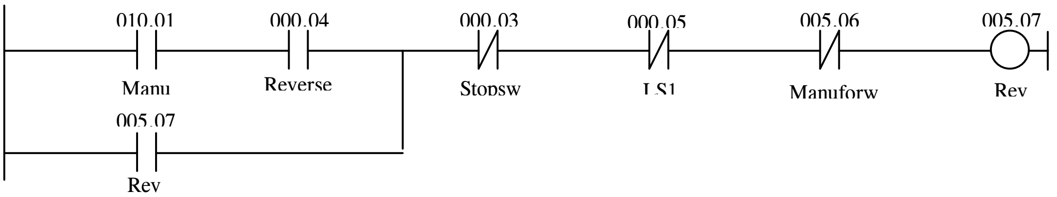
Network 2 – Auto indication



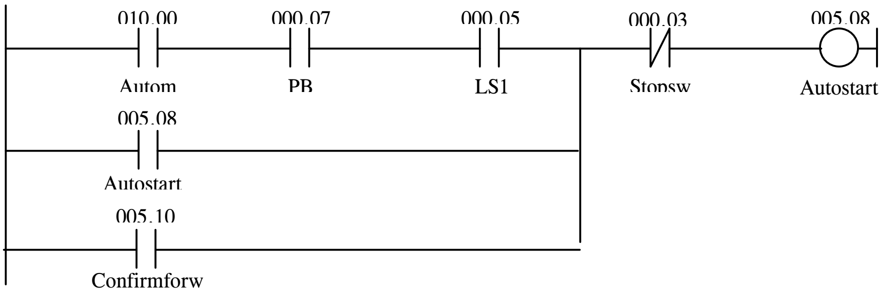
Network 3 – Manual forward



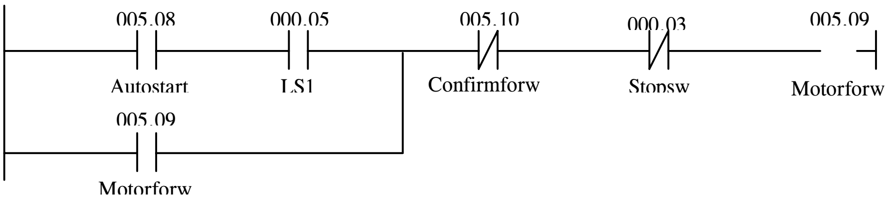
Network 4 – Manual reverse



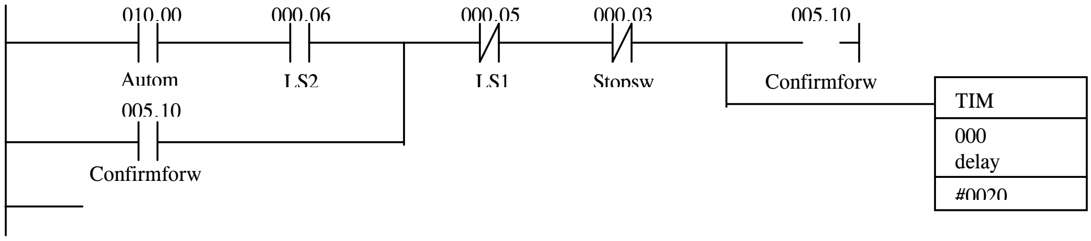
Network 5 – Auto start



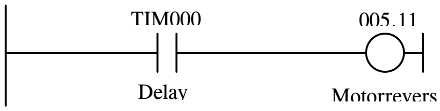
Network 6 – Motor forward



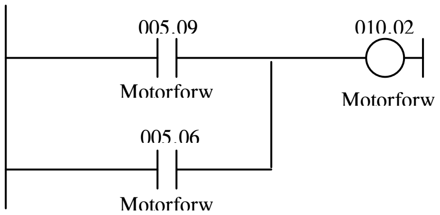
Network 7 – Confirm forward



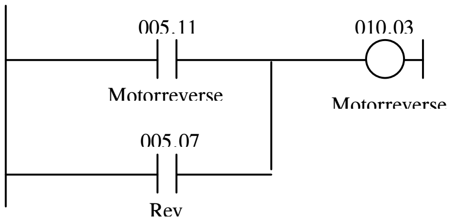
Network 8 – Motor reverse



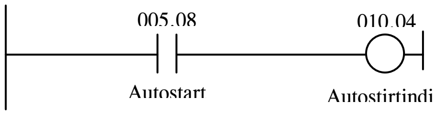
Network 9 – Motor forward



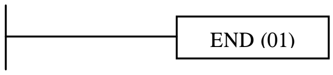
Network 10 – Motor reverse



Network 11 – Autostart
Autostart indicator



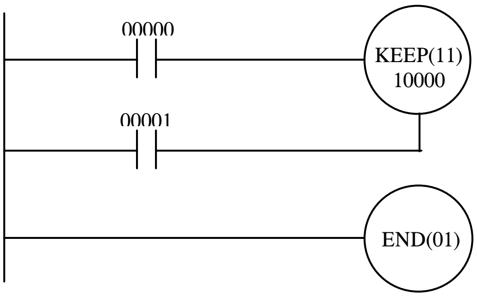
Network 12 – End



❑ KEEP(11) – Latching relay

KEEP is used as a latch. It maintains an ON or OFF state of a bit until one of its two inputs sets or resets it. If the KEEP function is used together with a HR relay, the state of the latched output is retained even during a power failure.

Ladder diagram

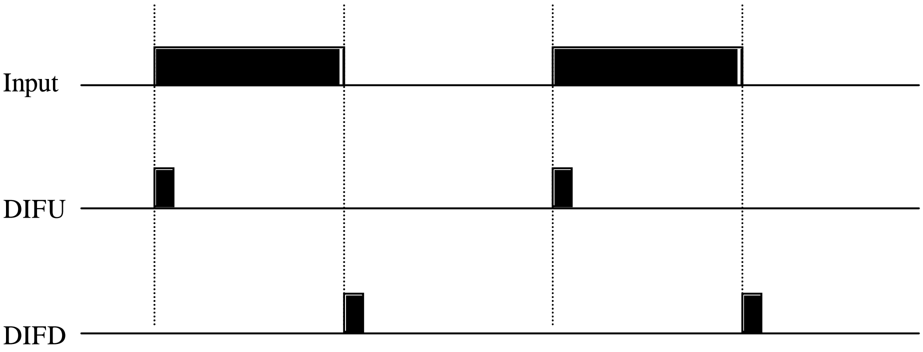


Mnemonic codes

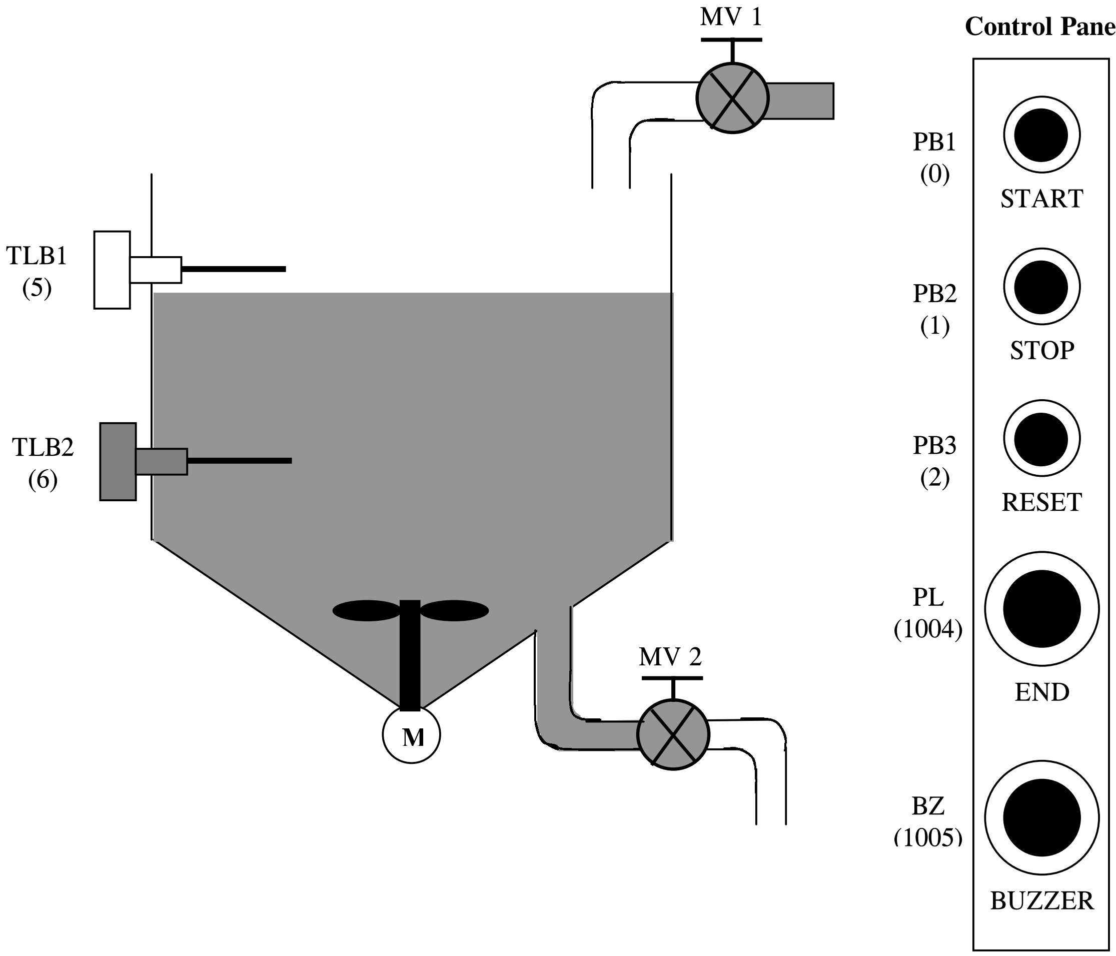
Address	Instruction	Data
0000	LD	00000
0001	LD	00001
0002	KEEP(11)	10000
0003	END(01)	

❑ DIFU(13) and DIFD(14) – Differentiation

DIFU and DIFD turns an output ON for one scan only.
DIFU turns its output ON when it detects an OFF -> ON transistion in its input signal.
DIFD turns its output ON when it detects an ON -> OFF transistion in its input signal.



❑ Application #5: Filling/Draining Control Operation



- **I/O Assignments**

Input	Device
00000	Start Button (PB1)
00001	Stop Button (PB2)
00002	Reset Button (PB3)
00005	Upper Level Switch (TLB1)
00006	Lower Level Switch (TLB2)

Output	Device
01000	Water Supply Valve (MV1)
01001	Drain Valve (MV2)
01002	Stirring Motor (M)
01004	End Indicator
01005	Buzzer

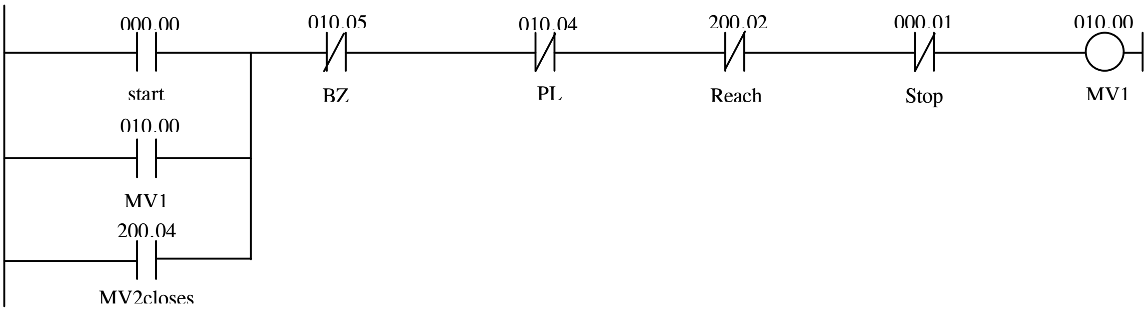
- **Procedure**

1. As the PB1 is pressed, MV1 opens and the water begins to fill the tank. At the same time, the stirring motor M starts operations.
2. When the water level passes TLB2 and reaches TLB1, the MV1 closes and the stirring motor stops.
3. Next, MV2 opens and starts draining the water. When the water levels drops below TLB2, MV2 closes.
4. When the cycle of operation has repeated four times, the operation END indicator illuminates, and the filling and draining operation will not restart even if PB1 is pressed.

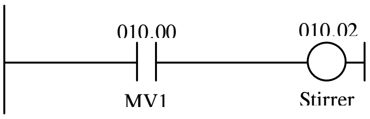
Ladder diagram : Main 1 Network 1

Main 1 – Fill / Draining
Filling / Draining control operation

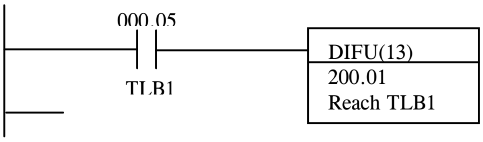
Network 1 – Start condition



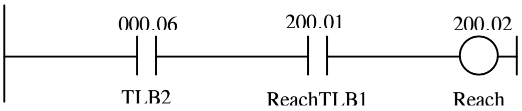
Network 2 – Stirring



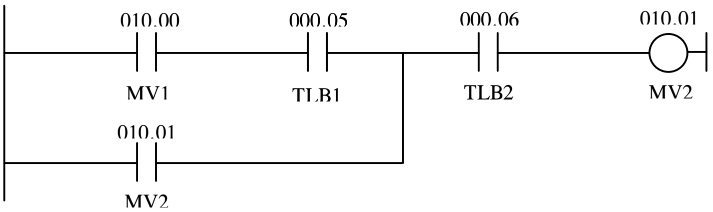
Network 3 – Reach TLB 1



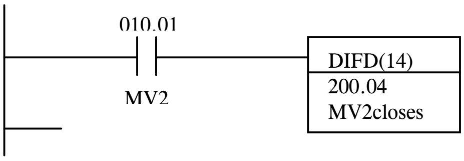
Network 4 – Water reach TLB



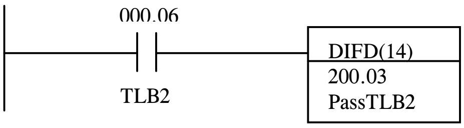
Network 5 – Drain



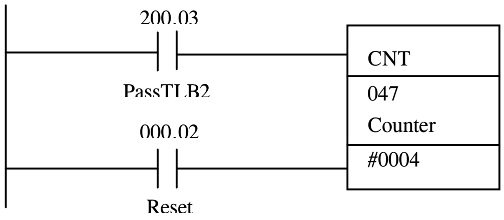
Network 6 – MV2 closes



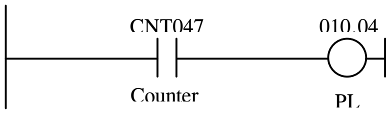
Network 7 – Pass TLB2



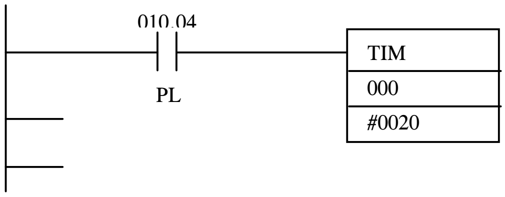
Network 8 – Counter
Counter preset at 4



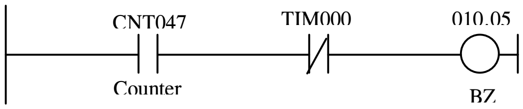
Network 9 – End indication



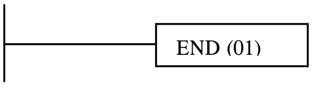
Network 10 – Delay
Delay for 2 seconds



Network 11 – Buzzer



Network 12 – End

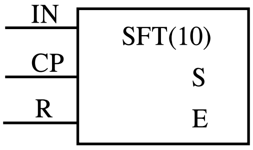


6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Shift Register – SFT(10)

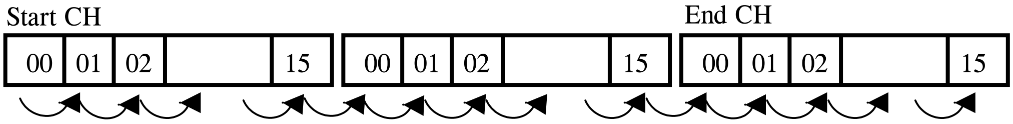
Shift Register (SFT) shifts a 16-bit data in specified channel by 1 bit. Although this instruction shifts data within channels, both a start channel and an end channel must be specified as the data.

Ladder diagram



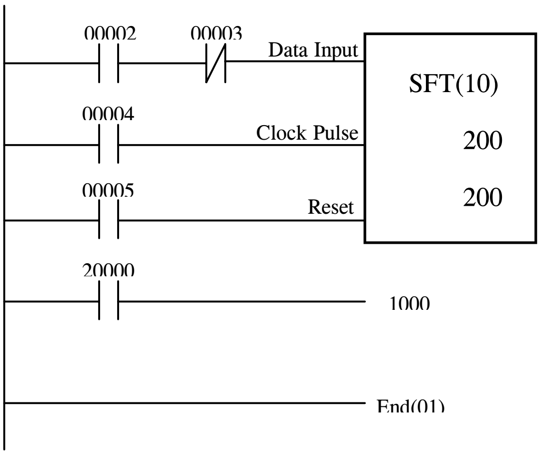
Operand Data Areas

I/O, Internal auxiliary Relay
Holding Relay



Example circuit

Ladder Diagram



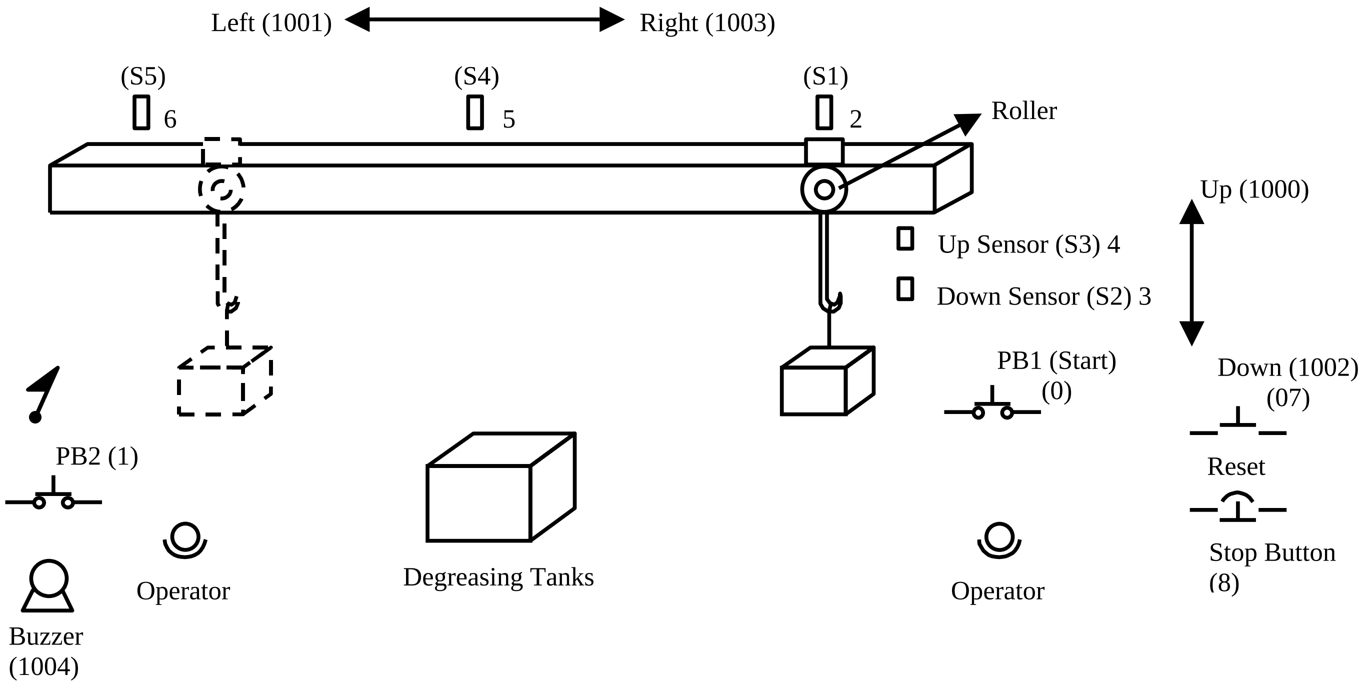
Mnemonic code

Address	Instruction	Data
0000	LD	00002
0001	AND NOT	00003
0002	LD	00004
0003	LD	00005
0004	SFT(10)	
		200
		200
0005	LD	20000
0006	OUT	1000
0003	END(01)	

- **Note :** When a reset input is applied to the Shift Register, all 16 bits are reset together. If the Holding Relay area is used, the data are retained during power failure.

❑ Application #6: Overhead Crane Control of Degreasing Operation

In this application, the part needs to be degreased in the degreasing tank before being passed to the next section.



When the PB1 is pressed, the roller will coil up the hook until the up-sensor (S3). The hook will then transverse left (via 01001) until it reaches the S4 position.

It will then stop and lower the product into the degreasing tank. When it reaches down to S2, the product will stay in the degreasing tank for 20 seconds. After the time is up, the product is lifted up and transverse left until S5 position and stop. It continue to come down. Until the down position, where the Buzzer will sound. The operator will collect the product and press the PB2 to return the crane back to the home position.

At anytime, the Stop Push Button can stop the crane from moving. Upon release, it will continue from where it stops. The Reset Push Button is used when you want to start over again from the beginning.

- **I/O Assignment**

Input	Device
00000	PB1 (Start button)
00001	PB2 (Return button)
00002	S1 (Sensor 1)
00003	S2 (Down sensor)
00004	S3 (Up sensor)
00005	S4 (Degrease sensor)
00006	S5 (End sensor)
00007	RST (Reset button)
00008	Stop (Stop button latch)

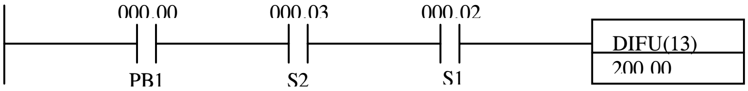
Output	Device
01000	Up motor
01001	Left motor
01002	Down motor
01003	Right motor
01004	Buzzer

6. APPLICATION EXAMPLES OF BASIC COMMANDS

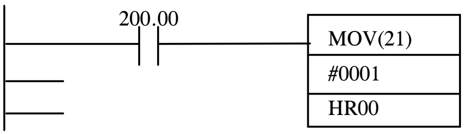
Ladder diagram : Main 1 Network 1

Main 1 –
Program for degreasing tank

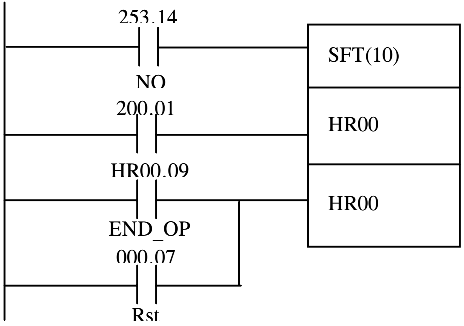
Network 1 – Start
Start operation



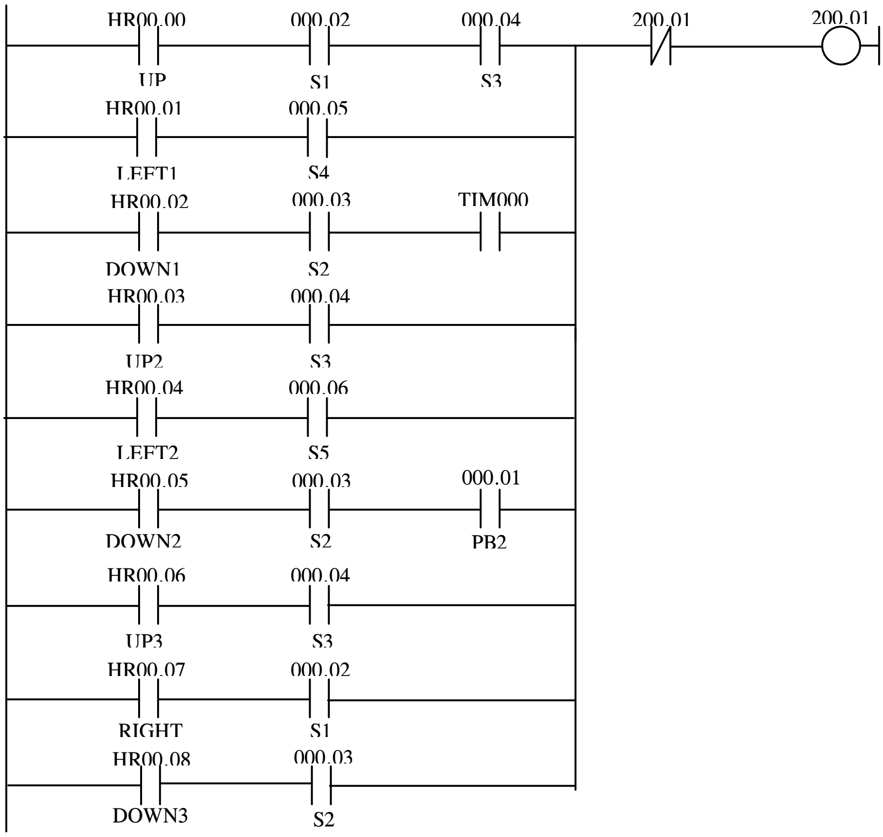
Network 2 – Set 1st bit on
Set 1st bit of HR 00 on



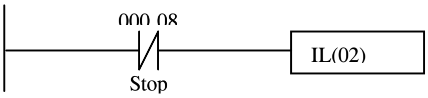
Network 3 – Shift Register



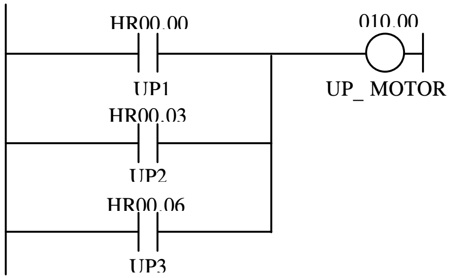
Network 4 – Shift operation
The shift register clock input to control the sequence of operation



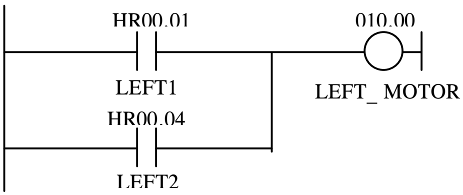
Network 5 – Interlock



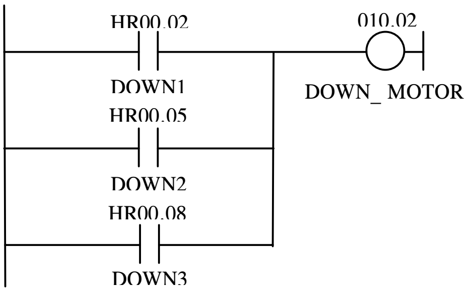
Network 6 – Up



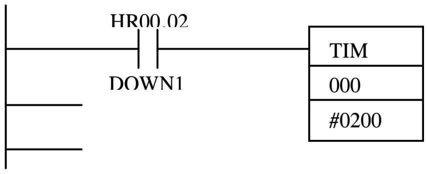
Network 7 – Left



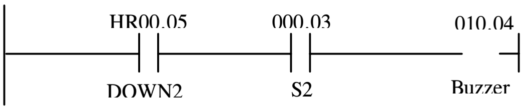
Network 8 – Down



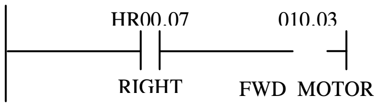
Network 9 – Time delay
Time delay for 20 secs



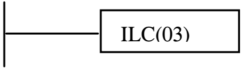
Network 10 – Buzzer



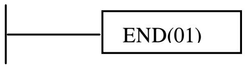
Network 11 – Right



Network 12 – Close interlock

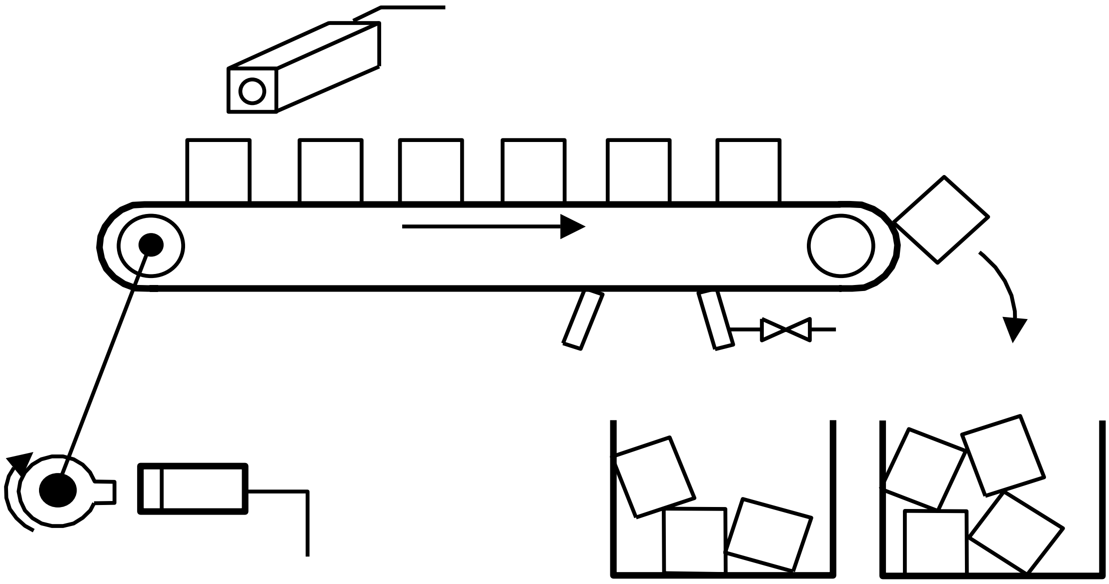


Network 13 – End



❑ Application #7: Parts Sorting

In this application, effective products are detected and rejected from those being carried on the conveyor.

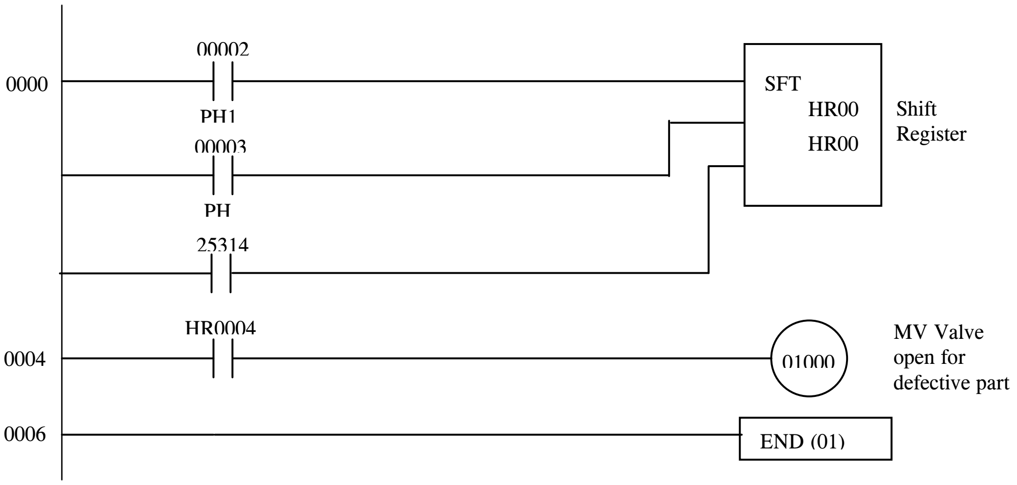


Photoelectric sensor (PH1) serves as the data input to the shift register. The signal output from this sensor turns ON when a defective product has been detected; otherwise it remains OFF.

Photoelectric sensor (PH2) is used as a clock generator that serves as the clock input to the shift register. It generates one pulse each time the product, spaced at a fixed interval from each other, has traveled a predetermined distance.

From the moment a defective product is detected by PH1, it is traced by the shift register until the product arrives at the predetermined position on the conveyor where it is ejected by the magnetic valve MV.

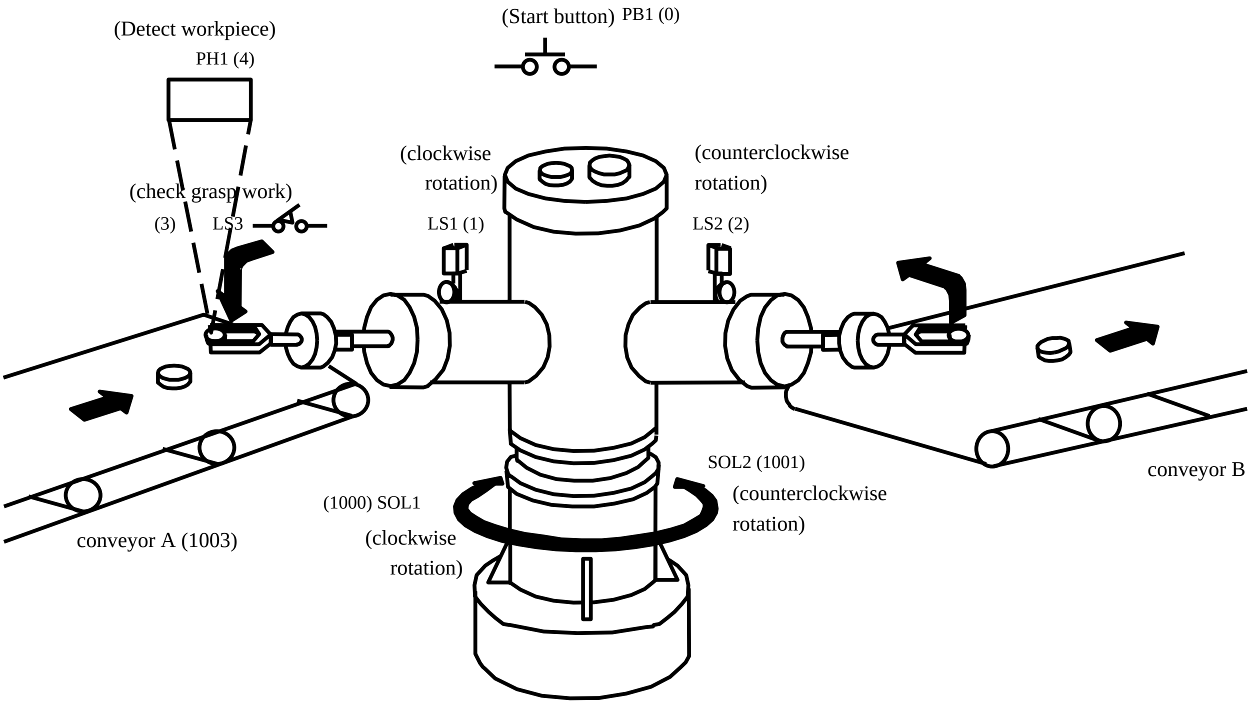
Ladder diagram



Mnemonic codes

Address	Instruction	Data
0000	LD	00002
0001	LD	00003
0002	LD	25314
0003	SFT	HR00
		HR00
0004	LD	HR0004
0005	OUT	01000
0006	END(01)	

❑ Application #8: Robot Movement Control



This kind of robot is seen in many automated factories. As is apparent from the figure, this robot picks up a work being carried on conveyor A and places it on conveyor B.

The operations and conditions are as follows:

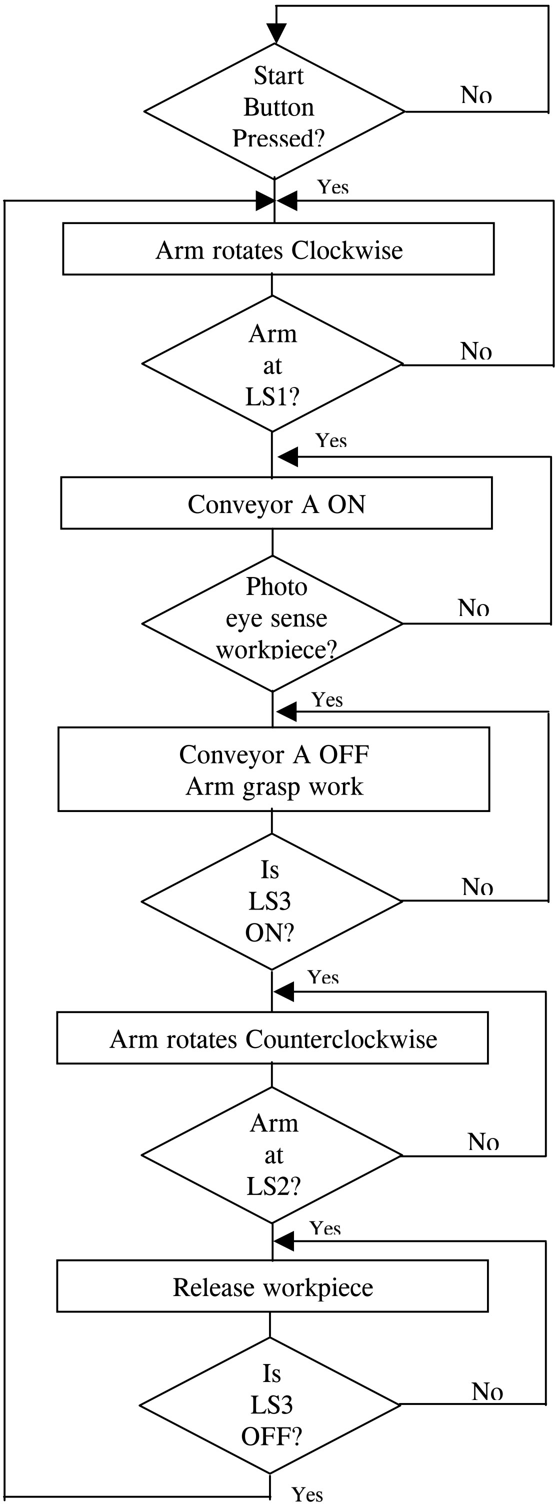
1. When the start button is pressed, the robot rotates its arm clockwise.
2. When the robot arm has moved to the position of the work on the conveyor A, arm grasps the work.
3. When the arm has grasped the work, it rotates counterclockwise.
4. When the arm has rotated to the position of conveyor B, it releases the work.

- **I/O Assignment**

Input	Devices
0000	PB1 (start button)
0001	LS1 (clockwise rotation)
0002	LS2 (counterclockwise rotation)
0003	LS3 (checking grasped work)
0004	PH1 (detect workpiece)
0005	Stop button
0006	Reset button

Output	Devices
1000	Sol 1 (clockwise rotation)
1001	Sol 2 (counterclockwise rotation)
1002	Sol 3 (grasping work)
1003	Conveyor A

- **Procedure**

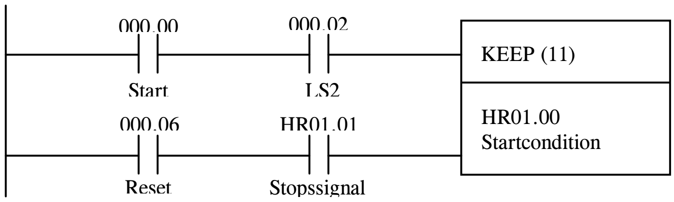


6. APPLICATION EXAMPLES OF BASIC COMMANDS

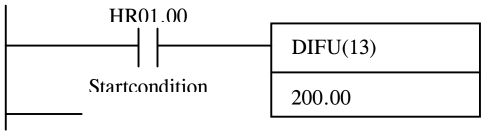
Ladder diagram : Main 1 Network 1

Main 1 – Robot control
This program demonstrates that the Robot picks up a work piece from conveyor A and places it on conveyor B.

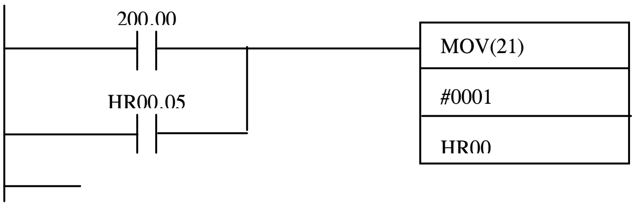
Network 1 – Start condition



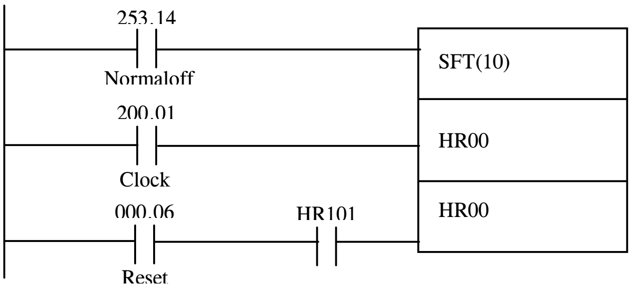
Network 2 – Start



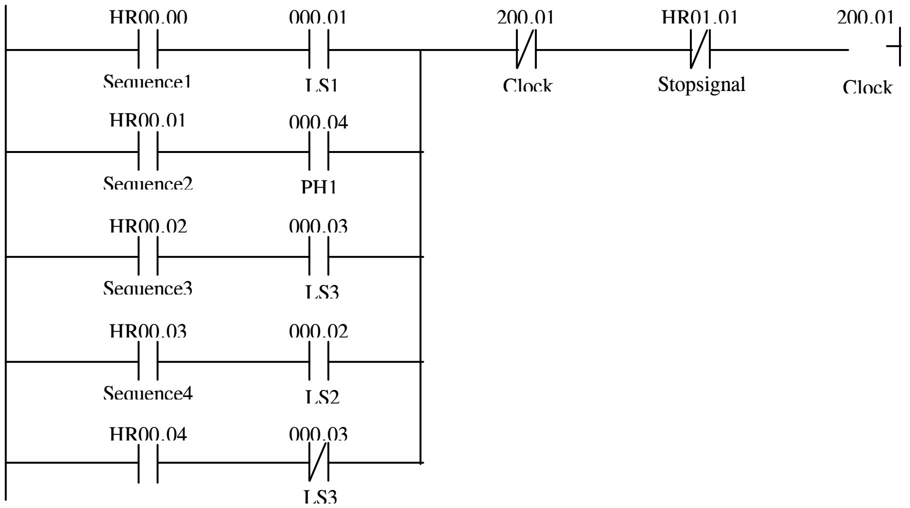
Network 3 – Set 1st bit on
Set 1st bit of HR 00 on



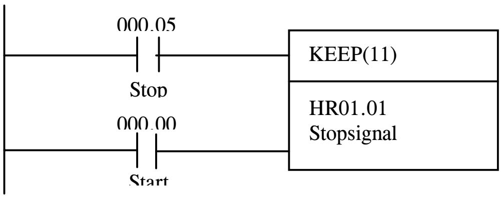
Network 4 - Shift register



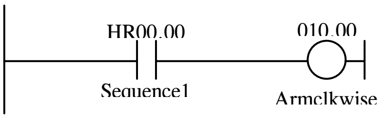
Network 5 – Clock input
Clock input to shift register to control sequence of operation



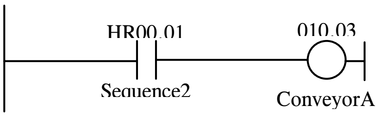
Network 6 – Stop condition



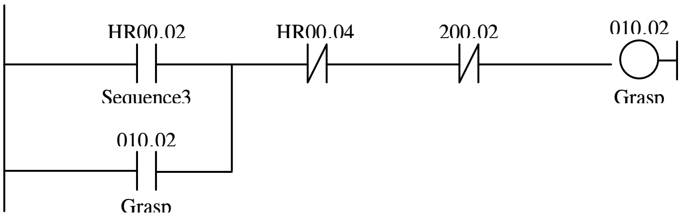
Network 7 – Arm clockwise



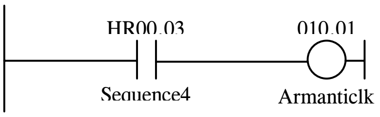
Network 8 – Conveyor A



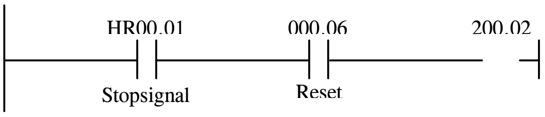
Network 9 – Grasp



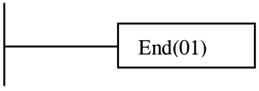
Network 10 – Arm anticlockwise



Network 11 – Reset
Reset after stop

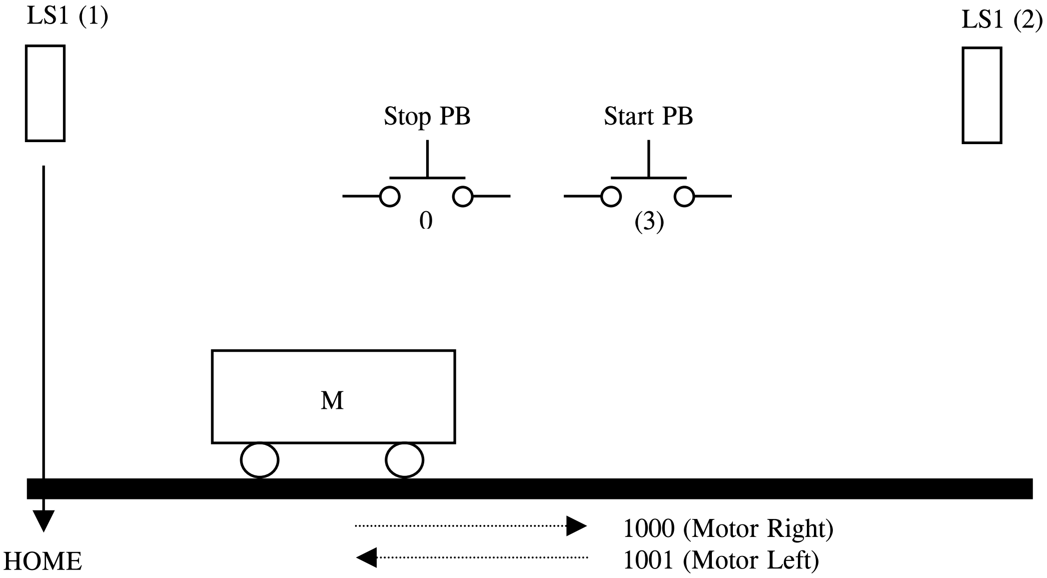


Network 12 – End



❑ Application #9: A Simple Sequence Control Concept

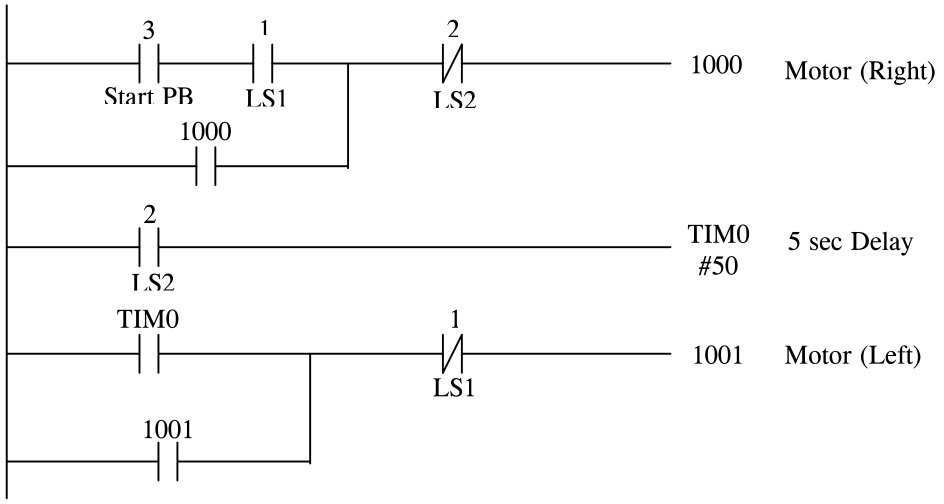
When the start button is pushed, the motor (M) will move from left to right. When LS2 is ON, the Motor stops, delay for 5 seconds and then moves back to Home. When LS1 (Home) is ON, Motor cuts off, signifying that the sequence is completed.



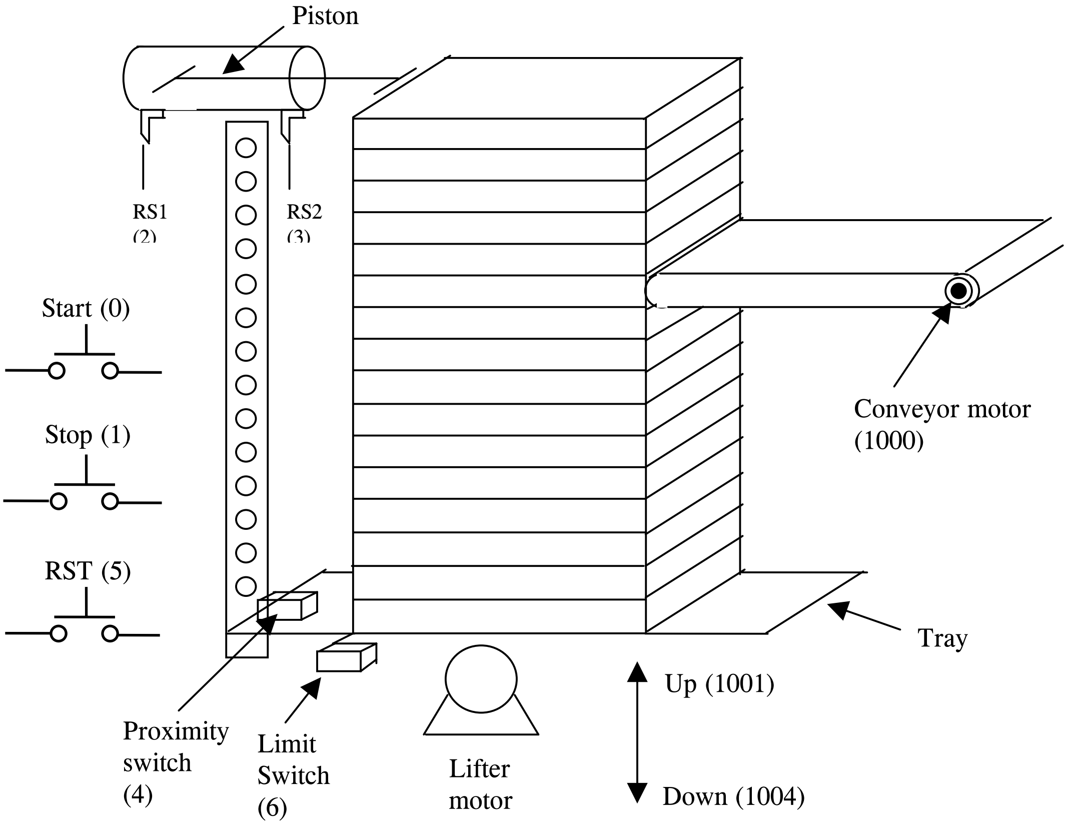
- **I/O Assignment**

Input	Devices
00000	Stop PB
00001	LS1 (HOME)
00002	LS2
00003	Start PB

Output	Devices
01000	Motor (right)
01001	Motor (Left)



❑ Example : PBC Packing Conveyor



In this application, a lifter motor is used to lift a tray of PCB boards up, before being pushed by the piston onto the conveyor for packing.

When the start button is pressed, the conveyor motor and the lifter motor will turn on. The proximity switch will temporarily stop the lifter motor for the piston to push the PCB onto the conveyor belt.

After the PCB is pushed, the piston will retract and the lifter motor starts again. The whole procedure will repeat itself thereafter.

When all the PCBs have been pushed onto the conveyor belt, the lifter motor will move down until the limit switch (LSW) is being activated. Thereafter, the whole procedure can only be started by the start switch.

- **I/O Assignment**

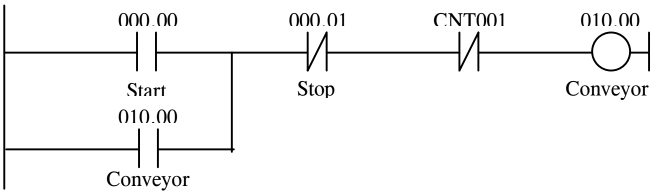
Input	Device
00000	Start Push Button
00001	Stop Push Button
00002	RS1 (Left Limit)
00003	RS2 (Right Limit)
00004	Proximity Switch
00005	Reset
00006	Limit Switch

Output	Device
01000	Conveyor motor
01001	Up Lifter motor
01002	Right piston
01003	Left piston
01004	Down lifter motor

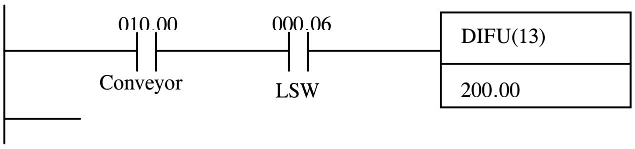
Ladder diagram : Main 1 Network 1

Main1 –
PCB packing conveyor program utilising (Conventional method)

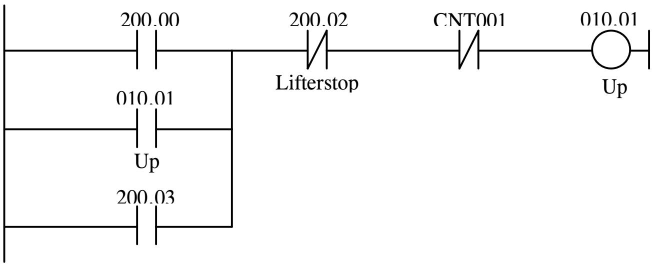
Network 1 –
Start / Stop control



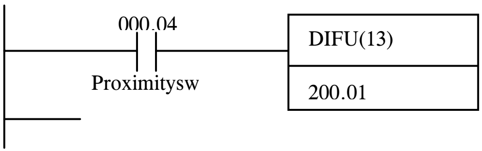
Network 2 – Left limit
Left limit reached for piston



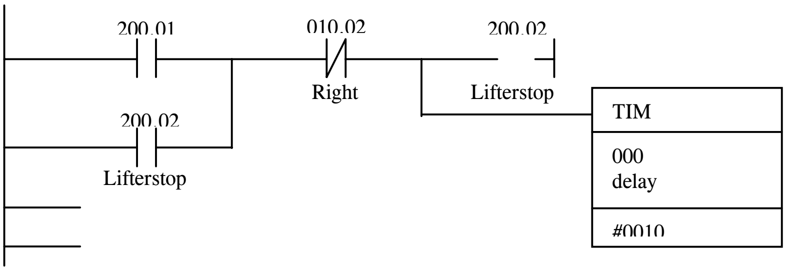
Network 3 – Lifter up
Lifter motor up



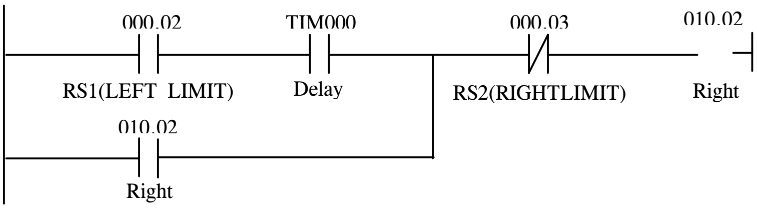
Network 4 – Index switch
Proximity switch detect the next index of PCB



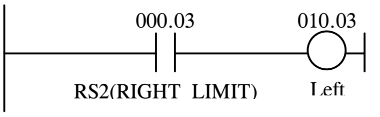
Network 5 – Lifter up
Lift the PCB up one after another



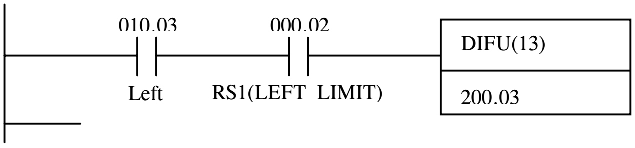
Network 6 – Pistonright



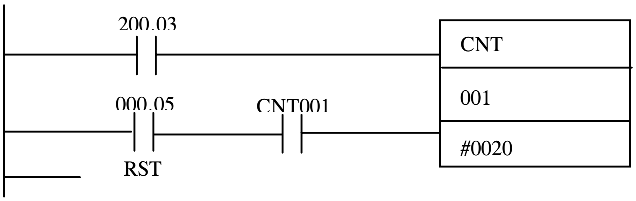
Network 7 – Pistonleft
Move piston after right limit is reached



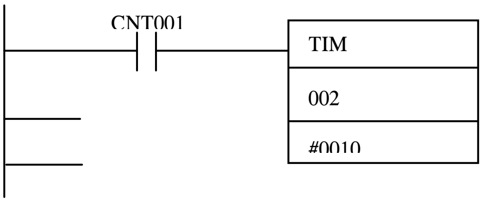
Network 8 – Pistonleft



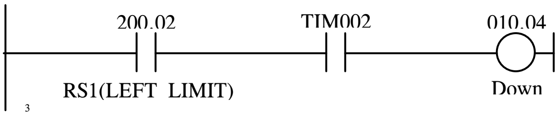
Network 9 – Counter
Counter preset at 20



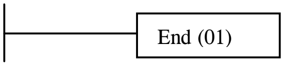
Network 10 – Delay of 1 sec



Network 11 – Liftdown



Network 12 – End

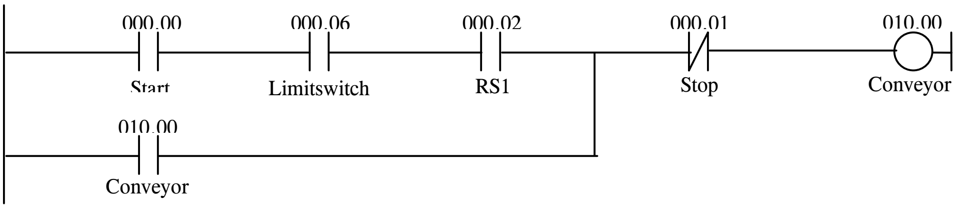


6. APPLICATION EXAMPLES OF BASIC COMMANDS

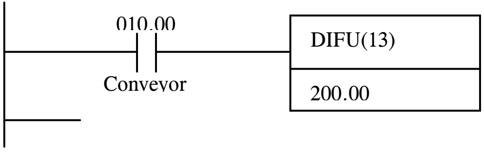
Ladder Diagram : Main 1 Network 1

Main 1 – SFT (10) method
PCB packing conveyor program utilising (shift register method)

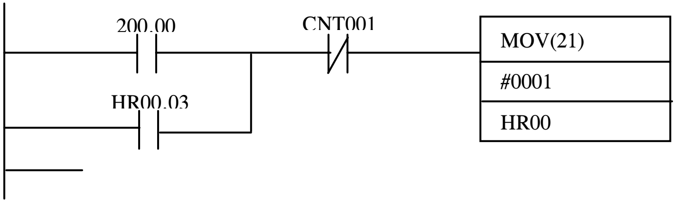
Network 1 – Conveyor



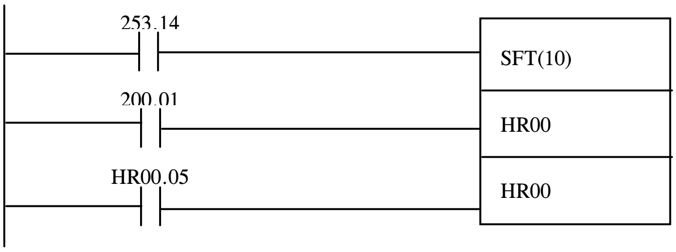
Network 2 – Set bit



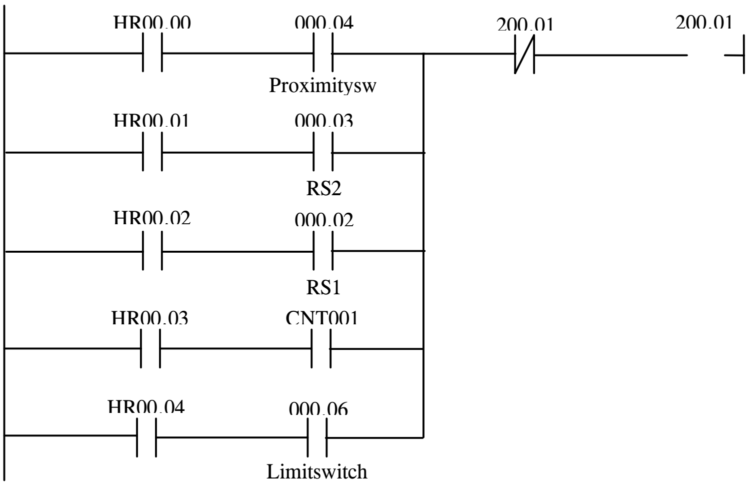
Network 3 – Set 1st bit
Set first bit of HR 00 on



Network 4 – Shift register

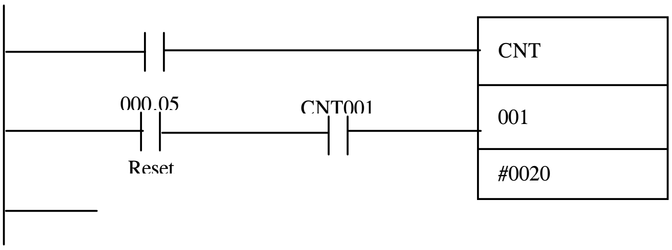


Network 5 – Clock
Clock input to shift register to control sequence of operations

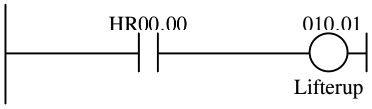


Network 6 – Counter

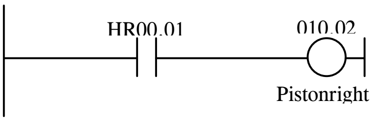
Counter preset at 20



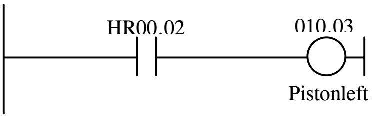
Network 7 – Lifter up



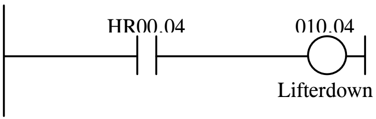
Network 8 – Piston right



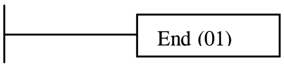
Network 9 – Piston left



Network 10 –Lifter down

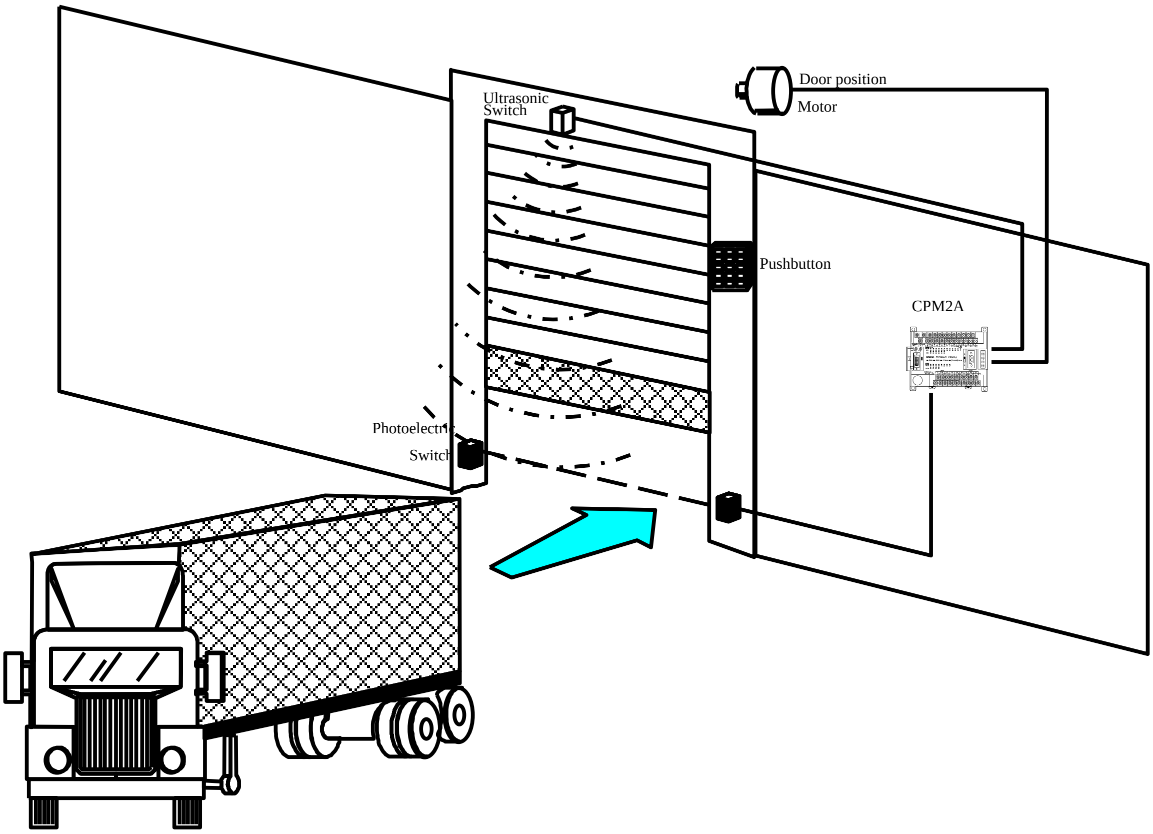


Network 11 – End



❑ Application #10: Automatic Control Of Warehouse Door

The input ultrasonic switch is employed to detect the presence of an approaching vehicle. A separate photosensor detects the passing of a vehicle via the interruption of the light beam. In response to these signals, the control circuit controls the outputs that drive the motor of the door for opening and closing.



- **I/O Assignment**

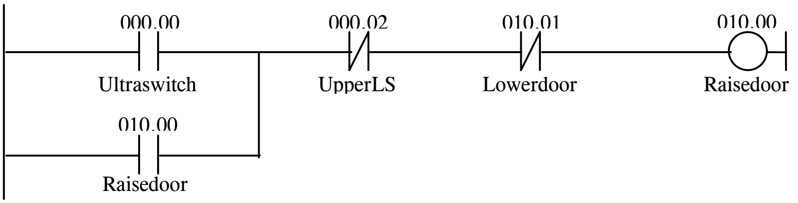
Input	Devices
00000	Ultrasonic switch
00001	Photoelectric switch
00002	Door upper limit switch
00003	Door lower limit switch

Output	Devices
01000	Motor to raise door
01001	Motor to lower door

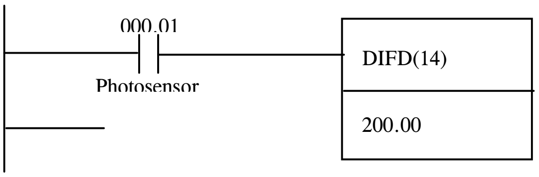
Ladder diagram : Main 1 Network 1

Main 1 – Autodoor
This program shows the automatic control of warehouse door

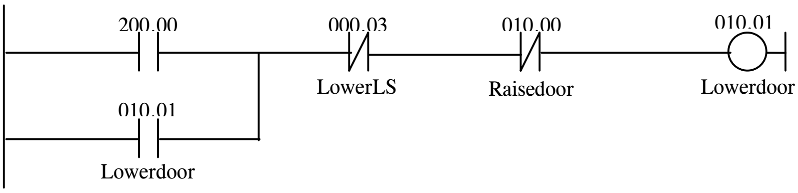
Network 1 – Raise door



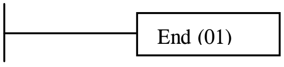
Network 2 – Photosensor
Sense until differentiation down



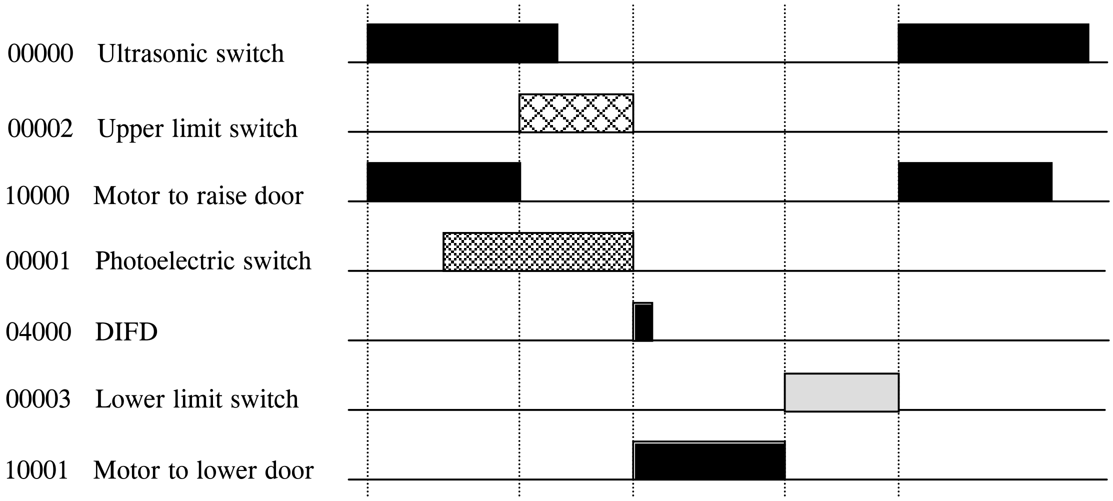
Network 3 – Lower door



Network 4 – End

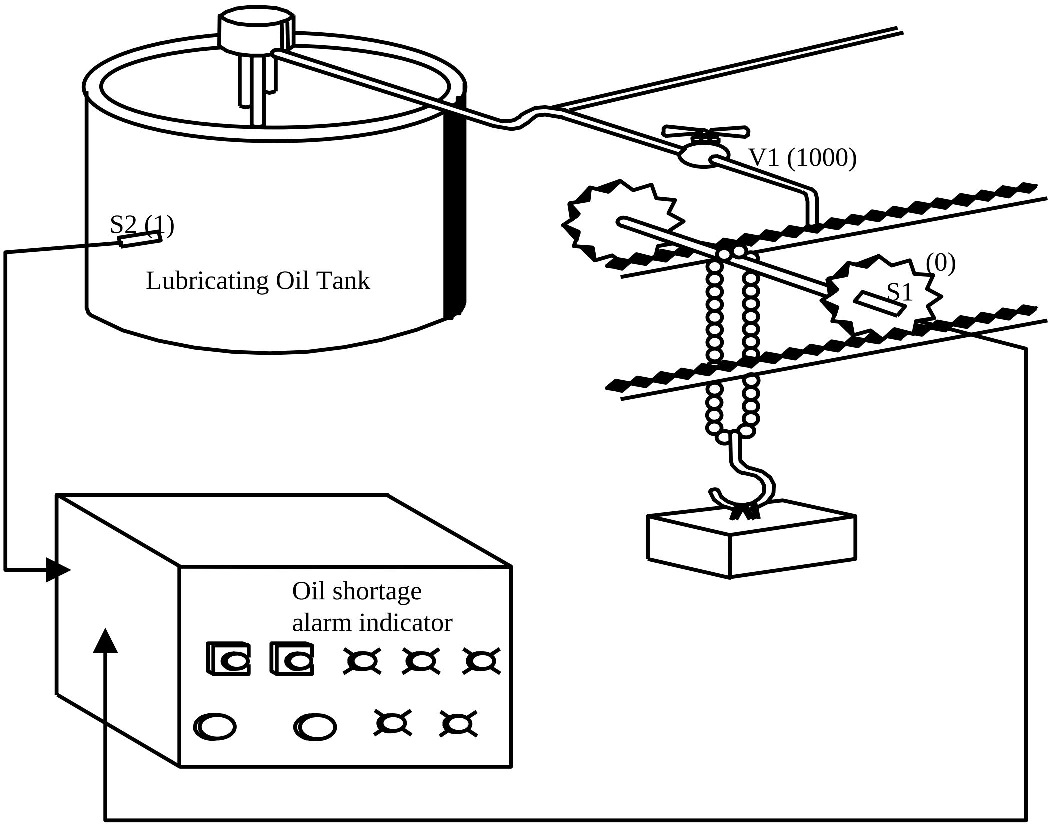


Timing diagram



Application #11: Automatic Lubrication of Gear

When the gear is moved towards S1, the sensor S1 will detect the gear and signal the electromagnetic valve for oil supply on the gear. The valve (V1) will open for a short period of time, supplying a predetermined quantity of oil. When sensor S2 sense that the lubricating tank oil level is low, the oil shortage alarm indicator will be ON.



- **I/O Assignment**

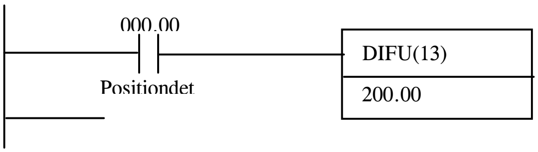
Input	Devices
00000	Position detection (S1)
00001	Lower limit of oil (S2)

Output	Devices
01000	Electromagnetic valve for oil supply (V1)
01001	Oil shortage alarm indicator

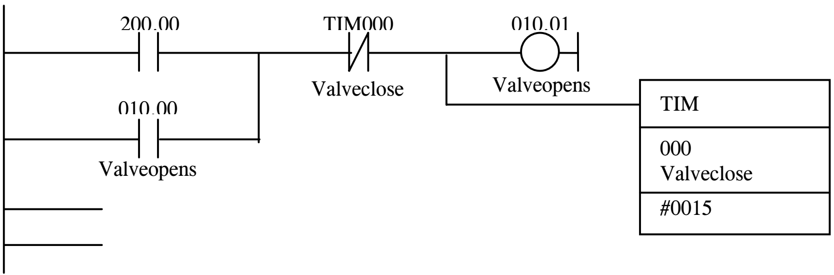
Ladder Diagram : Main 1 Network 1

Main 1 – Auto lubricate
Auto lubricate of gear

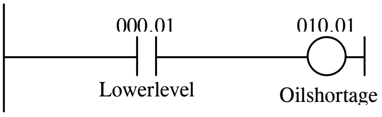
Network 1 – Start



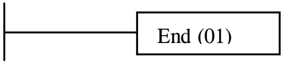
Network 2 – Open valve
Open valve and delay 1.5 sec



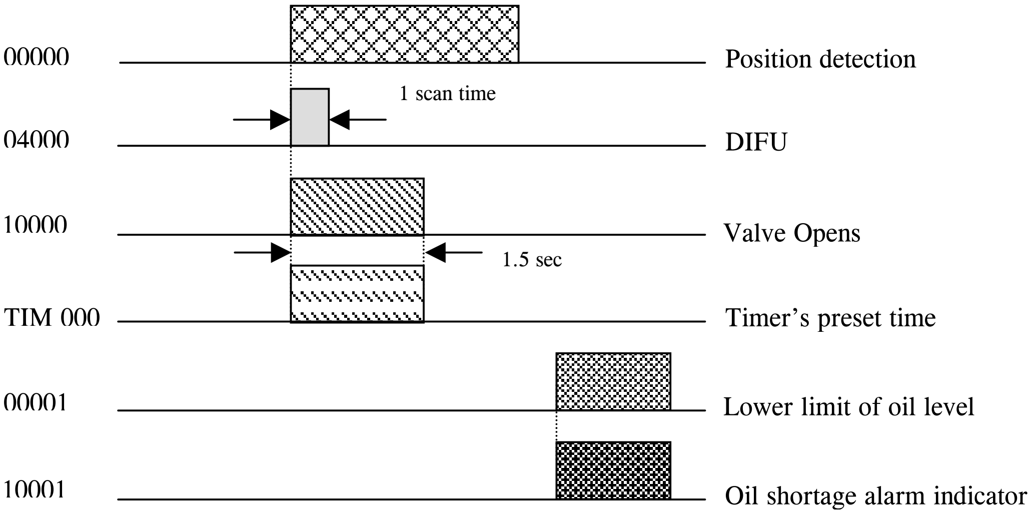
Network 3 – Oil shortage



Network 4 – End



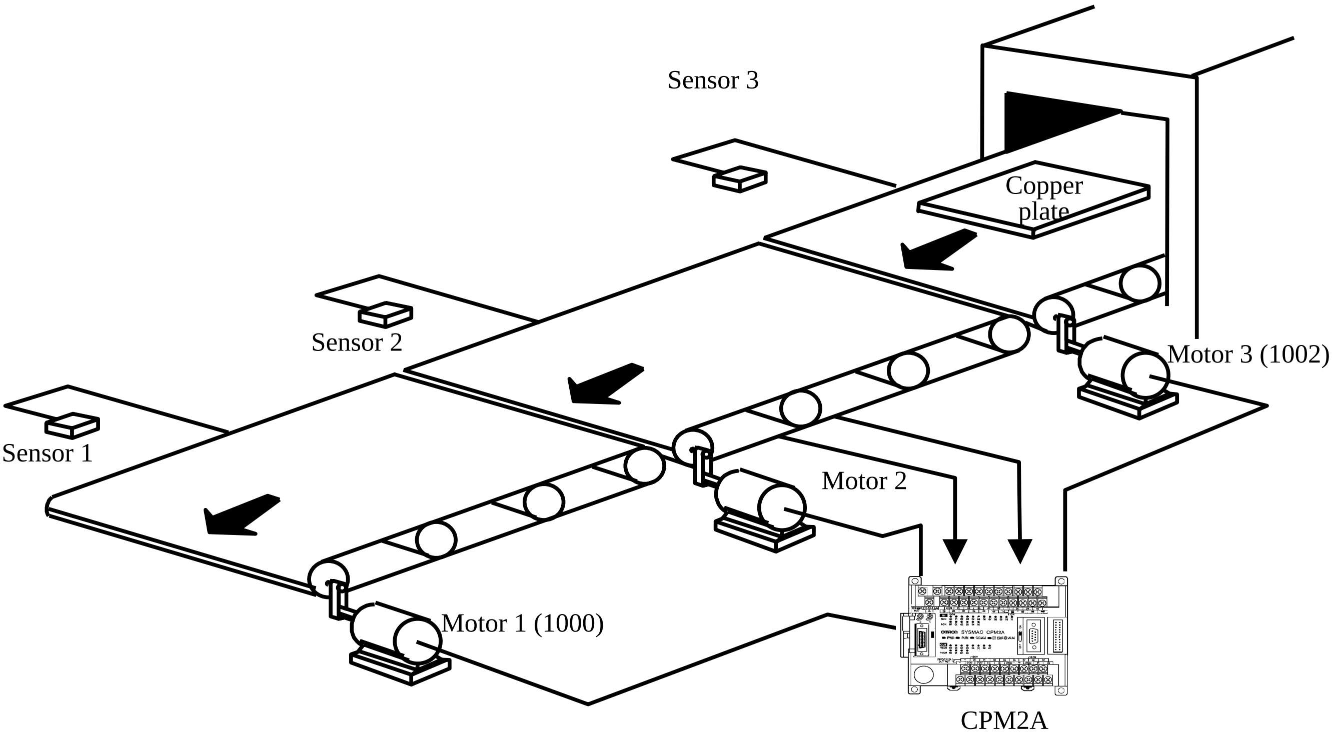
Timing diagram



6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Application #12: Conveyor Belt Motor Control

In this application, the PLC is used to start and stop the motors of a segmented conveyor belt. This allows only belt sections carrying an object (i.e. mental plate) to move. The position of a mental plate is detected by a proximity switch located next to each belt segment. As long as the plate is within the detecting range of the switch, the motor will work. If the plate moves beyond the range, a timer is activated and when this set time has lapsed, the motor of that belt stops.



- **I/O Assignment**

Input	Devices
00000	Sensor 1
00001	Sensor 2
00002	Sensor 3

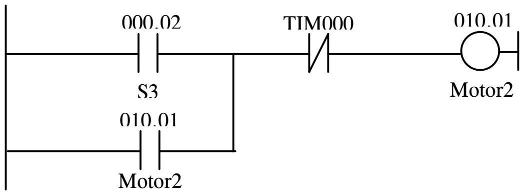
Output	Devices
01000	Motor 1
01001	Motor 2
01002	Motor 3

6. APPLICATION EXAMPLES OF BASIC COMMANDS

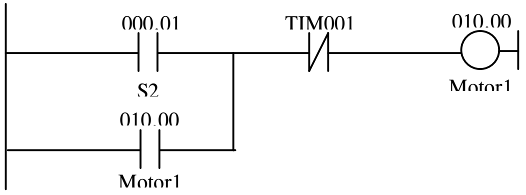
Ladder diagram : Main 1 Network 1

Main 1 – Conveyorcontrol
Conveyor belt control application

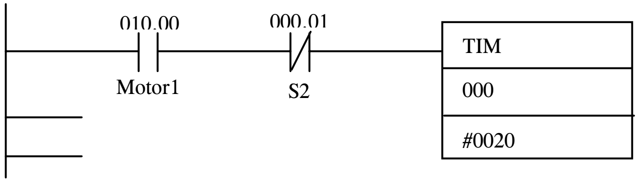
Network 1 – Motor2



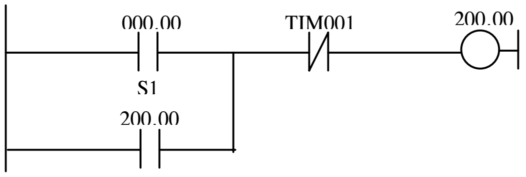
Network 2 – Motor1



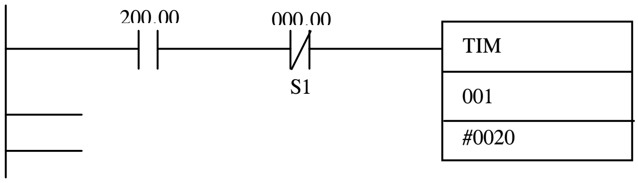
Network 3 – Delay for 2sec



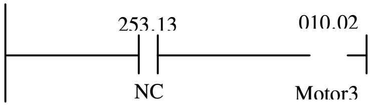
Network 4 – Sensor1



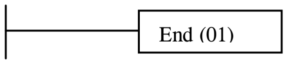
Network 5 – Delay for 2sec



Network 6 – Motor3



Network 7 – End



- **Operation :**

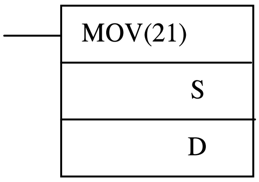
a) Motor 2 turns ON when Sensor 3 detects the product
b) Motor 2 is ON until Motor 1 is turned ON and product is out of detection range of Sensor 2
c) Motor 1 turns ON when Sensor 2 detects the product
d) Motor 1 is ON until product is out of detection range of Sensor 1

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Move – MOV(21)

MOV transfer source data (either the data in a specified channel or a four digit hexadecimal constant) to a destination channel. Therefore, MOV requires two data parameters to be specified: the source channel or constant and the destination channel.

Ladder Symbols

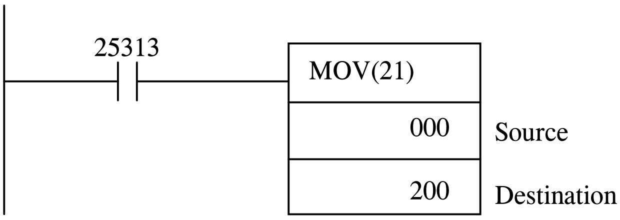


Operand Data Areas

S: Source channel
IR, SR, AR, DM, HR, TC, LR, #

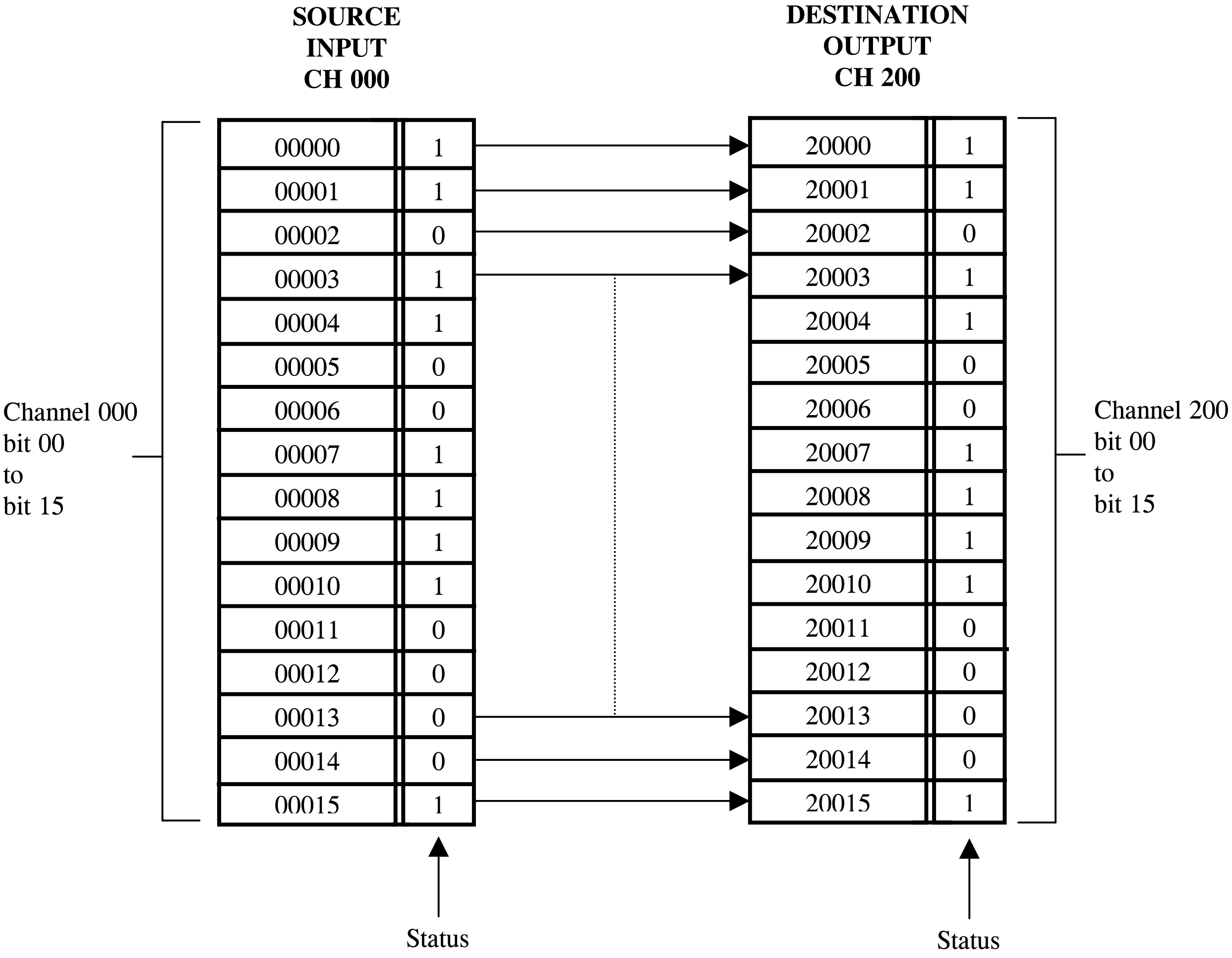
D: Destination channel
IR, AR, DM, HR, LR

Example circuit
Ladder diagram



Address	Instruction	Data
0000	LD	25313
0001	MOV(21)	
		000
		200
0003	END(01)	

The following diagram illustrates the MOV operation:



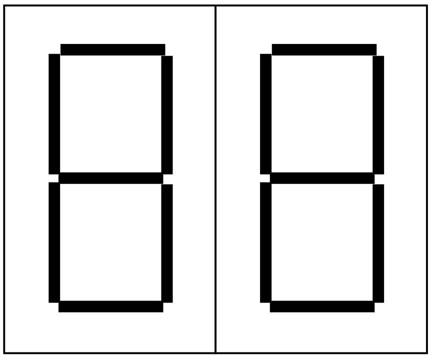
SOURCE INPUT CH 000		DESTINATION OUTPUT CH 200	
00000	1	20000	1
00001	1	20001	1
00002	0	20002	0
00003	1	20003	1
00004	1	20004	1
00005	0	20005	0
00006	0	20006	0
00007	1	20007	1
00008	1	20008	1
00009	1	20009	1
00010	1	20010	1
00011	0	20011	0
00012	0	20012	0
00013	0	20013	0
00014	0	20014	0
00015	1	20015	1

In this case, data in Input Channel 000 is moved to Output Channel 200.

❑ Application #13: Display Error Code Of Machine To Aid In Tracing The Source Of The Problem

1. Activation of error input signal 00001 to 00004 will sound an alarm and at the same time display the error code.
2. Input 00005 serves to reset the error code displayed upon machine recovery.

Error code



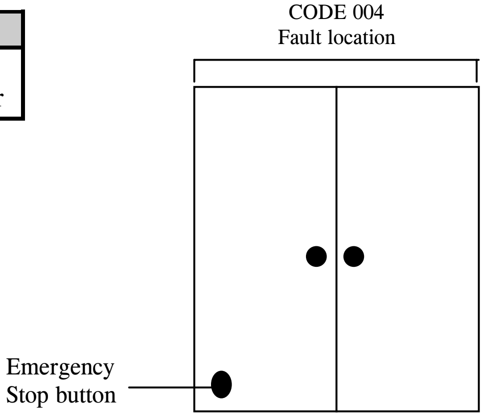
Code	Fault location
001	Feeding section problem Check Sensor No: S01 Check Limit Switch No: L03
002	Labeling Unit Fault Check Contactor No: C01
003	Conveyor Jam Check Label Sensor No: S05
004	Emergency stop Check Emergency Stop button

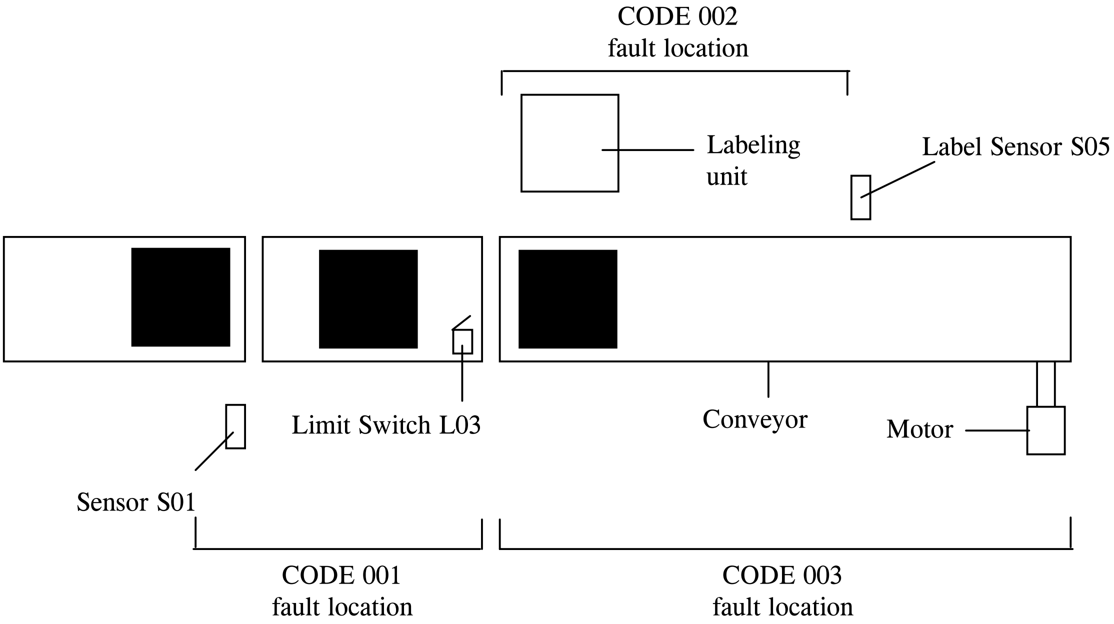
Diagram

I/O Assignment

Input	Devices
00001	PB1
00002	PB2
00003	PB3
00004	PB4

Output	Devices
1000	Alarm Annunicator



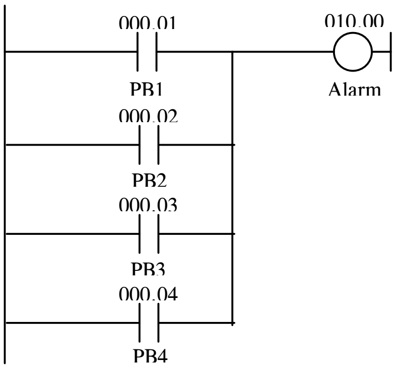


6. APPLICATION EXAMPLES OF BASIC COMMANDS

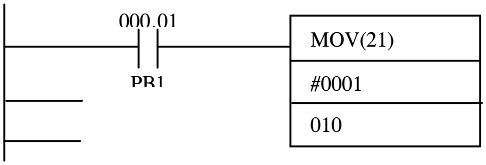
Ladder Diagram : Main 1 Network 1

Main 1 – Error Display
Display error code of machine to aid tracing source of problem

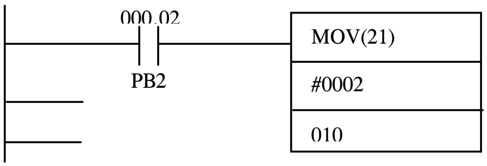
Network 1 - Alarm



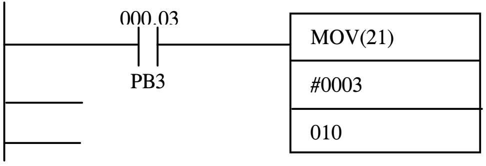
Network 2 – Code 1
Feeding Problem



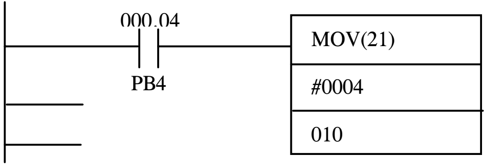
Network 3 – Code 2
No Label



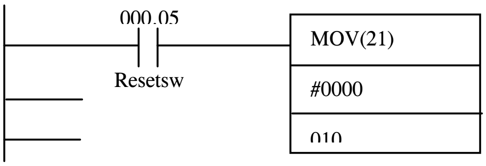
Network 4 – Code 3
Conveyor Jam



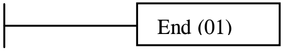
Network 5 – Code 4
Emergency Stop



Network 6 – Code 5
Reset Code Display



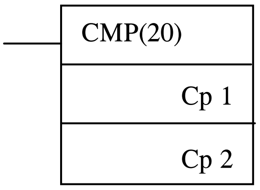
Network 7 – End



❑ Compare – CMP(20)

Compare (CMP) is used to compare the data in a specific channel, with the data in another channel, or a four-digit, hexadecimal constant. Therefore, two data must be specified immediately after the CMP(20) instruction.

Ladder Symbols

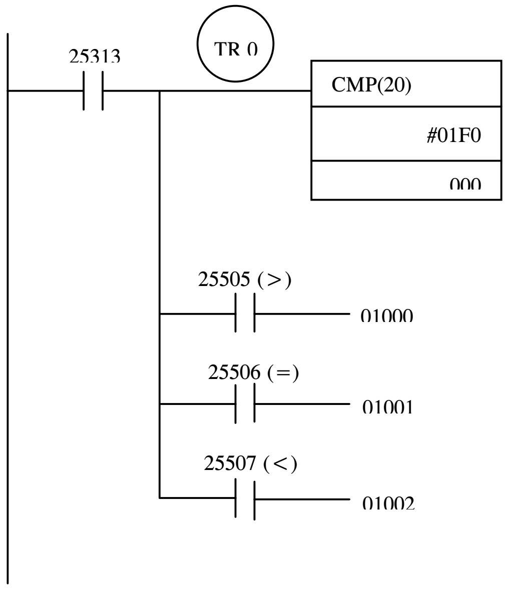


Operand Data Areas

Cp : First compare word
IR, SR, AR, DM, HR, TC, LR, #

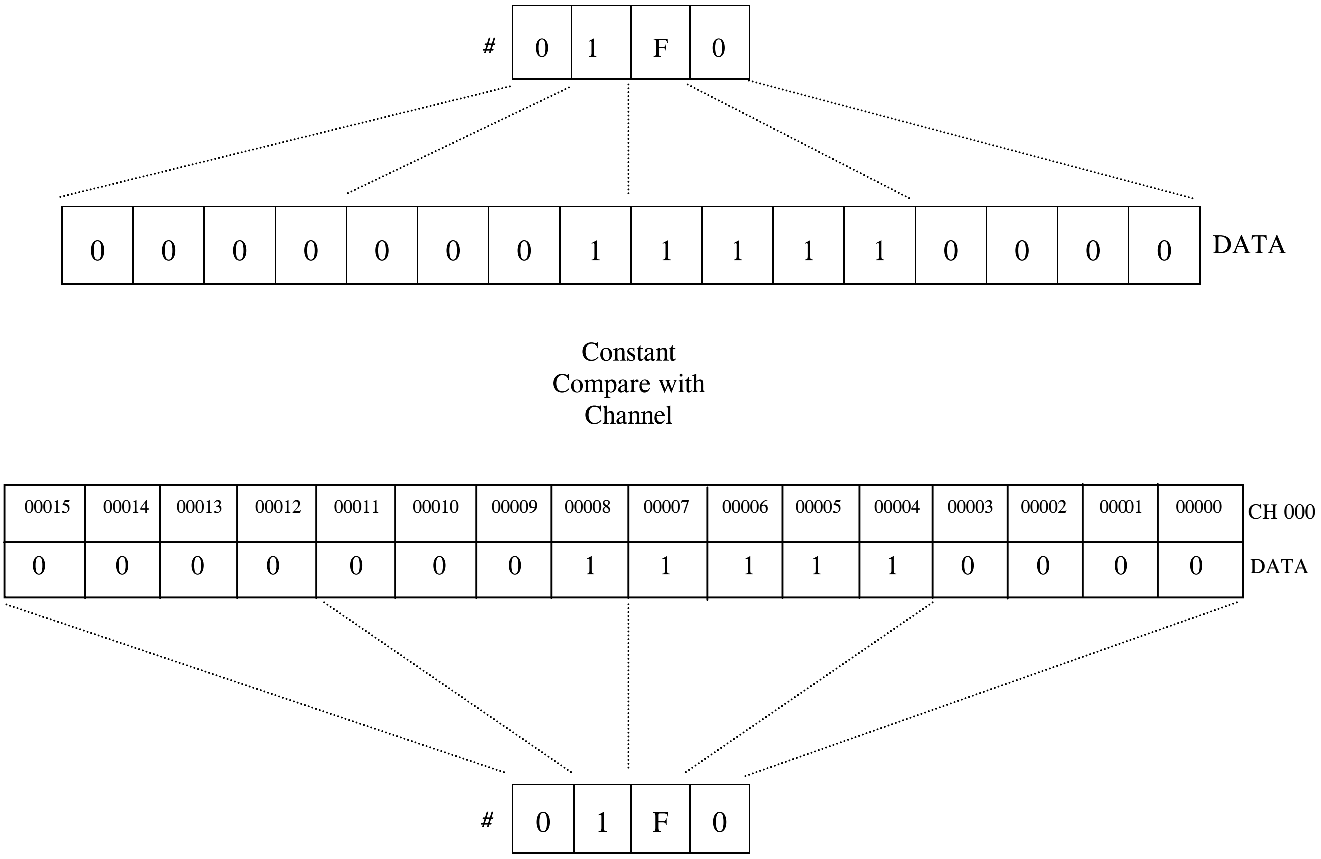
Cp2 : Second compare word
IR, SR, AR, DM, HR, TC, LR, #

Example circuit
Ladder diagram



Address	Instruction	Data
0000	LD	25313
0001	OUT	TR 0
0002	CMP(20)	-
		#01F0
		000
0003	AND	25505
0004	OUT	01000
0005	LD	TR 0
0006	AND	25506
0007	OUT	01001
0008	LD	TR 0
0009	AND	25507
0010	OUT	01002

The following diagram illustrates the CMP operation



#				
#	0	1	F	0

0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	DATA

Constant
Compare with
Channel

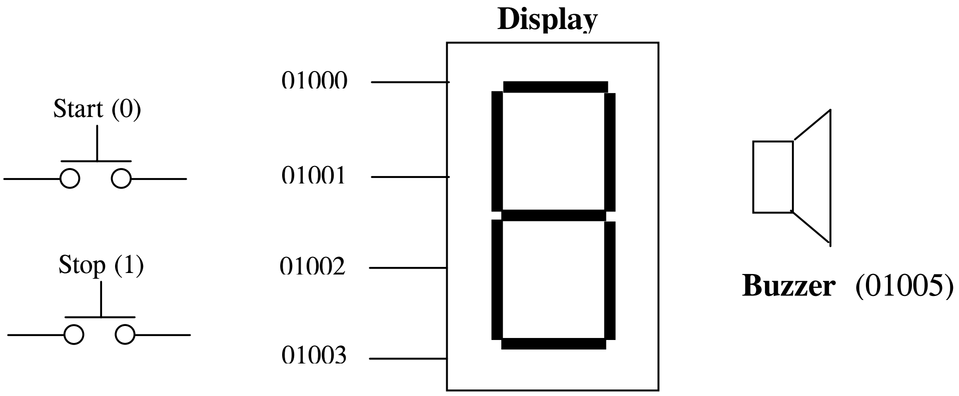
00015	00014	00013	00012	00011	00010	00009	00008	00007	00006	00005	00004	00003	00002	00001	00000	CH 000
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	DATA

#				
#	0	1	F	0

If the constant (#01F0) is ***equal*** to Channel 000 data, special relay 25506 turns ON. However, special relay 25505 will turn ON if the constant is ***greater*** than Channel 000 data and special relay 25507 will turn ON if the constant is ***less*** than Channel 000 data. At any one time, only one result is true, either relay 25505(>) or 25506(=) or 25507(<) is ON.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Example : A Time-out Warning



In this application, a 7-segment display & a buzzer is connected to the PLC. When the start button is pressed, the display will display a 9 and therefore will decrement until 0 before the buzzer is being sound. These type of circuit can be used as a warning circuit.

- **I/O Assignment**

Input	Device
00000	Start PB
00001	Stop PB

Output	Device
01000 to 01003	7-segment display
01005	Buzzer

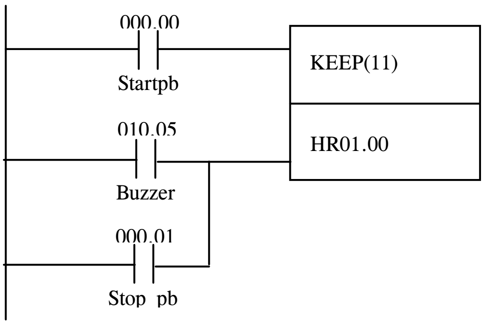
.

6. APPLICATION EXAMPLES OF BASIC COMMANDS

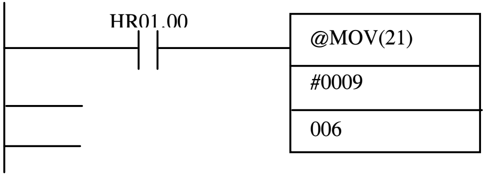
Ladder Diagram : Main 1 Network 1

Main 1 –
Program to display 9 before decrement it to 0

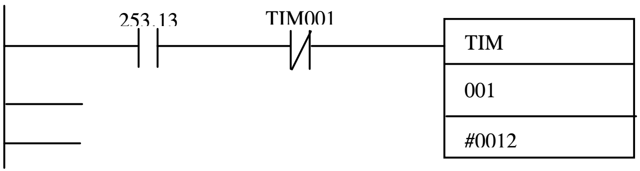
Network 1 –
Using the holding relay to decrement after every pulse



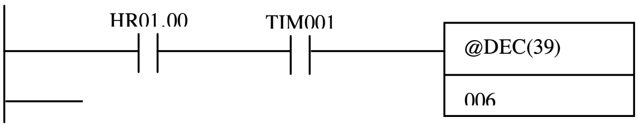
Network 2 – Move 9
Move 9 to channel 6



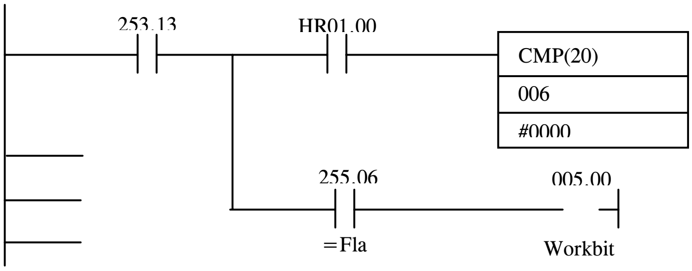
Network 3 – Timer
To produce an ON/OFF pulse



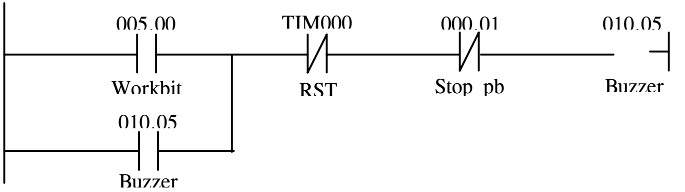
Network 4 – Decrement
Decrement every execution



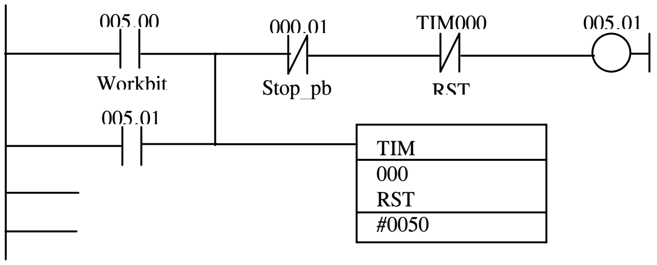
Network 5 –
Compare the o/p with 0, if equal (25506 on) output 00500 turn on



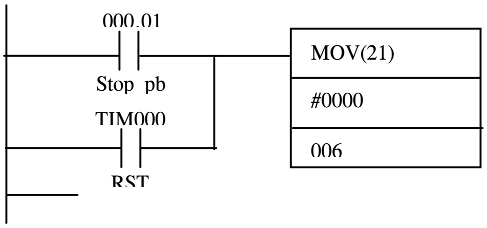
Network 6 - Buzzer



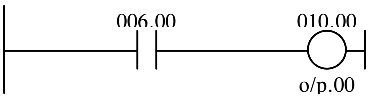
Network 7 –
Timer to cut the sound of buzzer



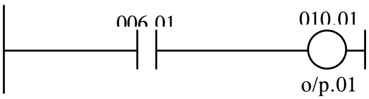
Network 8 – Stop
Reset channel 10



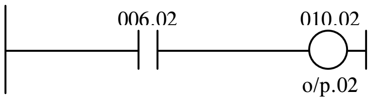
Network 9 – Segment1display



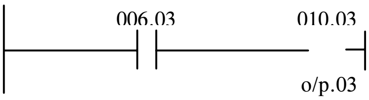
Network 10 – Segment2display



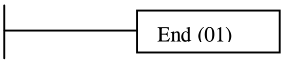
Network 11 – Segment3display



Network 12 – Segment4display



Network 13 – End



Notes : Instruction with @Sign will execute on the rising edge only.
It means that the instruction will execute for one scan only.

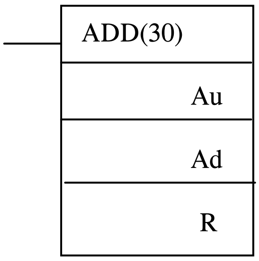
< < < Program to Display 9 before Decrement it to 0 > > >

Address	Instruction	Data	Comment	Comment Statement
00000	LD	00000		
00001	LD	01005	Start Pb	
		OR 1		
00002	KEEP(11)	HR 0100		Holding Relay
00003	LD	HR 0100		
00004	@MOV(21)			Move 9 to
		# 0009		
		6		
00005	LD	25313		
00006	AND NOT TIM	001		
00007	TIM	1		
		#0012		
00008	LD	HR 0100		
00009	AND TIM	1	Clk Pulse	
00010	@DEC(39)			Decrement o/p every clk
		6		
00011	LD	25313		
00012	OUT TR	0		
00013	AND	HR 0100		
00014	CMP(20)			Compare the o/p with
		6		
		# 0000		
00015	LD TR	0		
00016	AND	25506		Condition meet ON bit On
00017	OUT	00500		BUZZER
00018	LD	00500		
00019	OR	01005		
00020	AND NOT TIM	0		
00021	AND NOT	1		
00022	OUT	01005		
00023	LD	00500		
00024	OR	00501		
00025	OUT TR	0		
00026	AND NOT	1		
00027	AND NOT	TIM 000		
00028	OUT	00501		
00029	LD TR	0		Timer to cut the buzzer
	TIM	000		
00030		# 0050		
00031	LD	00001	STOP PB	RESET CH
00032	OR	TIM 000		
00033	MOV(21)			
		# 0000		
		6		
00034	LD	600		1
00035	OUT	1000		
00036	LD	601		2
00037	OUT	1001		
00038	LD	602		4
00039	OUT	1002		
00040	LD	603		8
00041	OUT	1003		
00042	END(01)			

❑ Add – ADD(30)

ADD totals the data in two different channels, or one channel and a constant and then outputs the sum to a third channel.
Therefore, three data parameters must be specified : an augend, an addend and a result channel.

Ladder Symbols



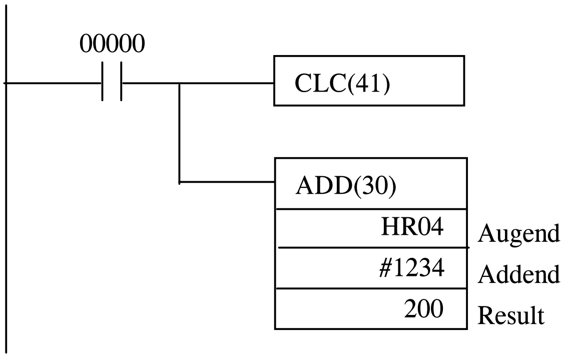
Operand Data Areas

Au: Augend channel
IR, SR, AR, DM, HR, TC, LR, #

Ad: Addend channel
IR, SR, AR, DM, HR, TC, LR, #

R: Result channel
IR, AR, DM, HR, LR

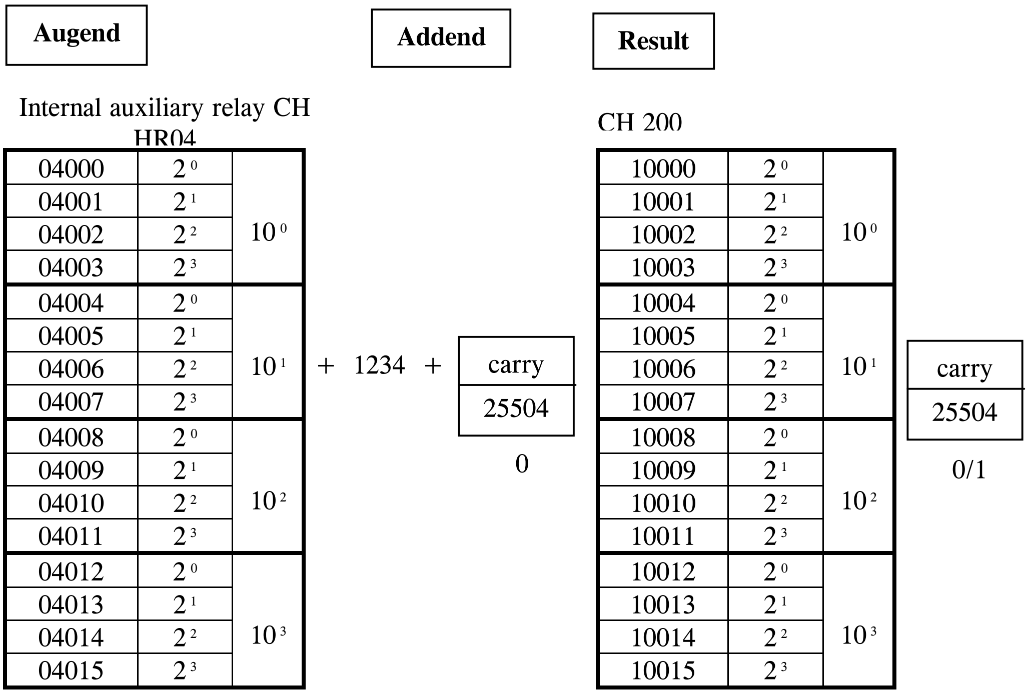
Example circuit
Ladder diagram



Address	Instruction	Data
0000	LD	00000
0001	CLC(41)	
0002	ADD(30)	
		HR04
		#1234
		200
0004	END(01)	

In the program, when input 00000 is turned ON, the data in internal relay **HR040** is added to the constant **1234**. The result of the addition is output to **CH 200**. If a carry is generated as a result of the addition, the carry flag (special relay 25504) is turned ON. If the result of the addition is 0000, special relay 25506 (the "=" flag) is turned ON.

The following diagram illustrates the ADD operation.



Augend **Addend** **Result**

Internal auxiliary relay CH HR04

04000	2^0	10^0
04001	2^1	
04002	2^2	
04003	2^3	
04004	2^0	10^1
04005	2^1	
04006	2^2	
04007	2^3	
04008	2^0	10^2
04009	2^1	
04010	2^2	
04011	2^3	
04012	2^0	10^3
04013	2^1	
04014	2^2	
04015	2^3	

+ 1234 +

carry
25504

0

CH 200

10000	2^0	10^0
10001	2^1	
10002	2^2	
10003	2^3	
10004	2^0	10^1
10005	2^1	
10006	2^2	
10007	2^3	
10008	2^0	10^2
10009	2^1	
10010	2^2	
10011	2^3	
10012	2^0	10^3
10013	2^1	
10014	2^2	
10015	2^3	

carry
25504

0/1

In the above example, before executing ADD, the Carry Flag/CY (special relay 25504) is turned OFF by the Clear Carry (CLC). The addition and subtraction instructions include CY in the calculation as well as in the result. Be sure to clear CY if its previous status is not required in the calculation, and to use the result placed in CY, if required, before it is changed by execution of any other instruction.

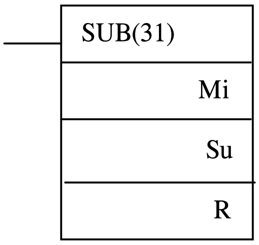
The augend and addend must be in **BCD**, if not special relay 25503 (Error Flag) is turned ON and ADD is not executed.

ADD is executed each time the CPU scans the program. To execute it only once.

❑ Subtract – SUB(31)

SUB finds the difference between the data in one channel and the data in another channel or a constant, and then outputs the result to a third channel.
Therefore, three data must be specified : an minuend, an subtrahend and a result channel.

Ladder Symbols



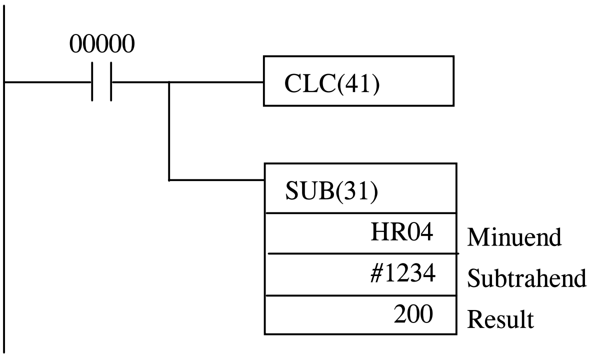
Operand Data Areas

Mi: Minuend channel
IR, SR, AR, DM, HR, TC, LR, #

Su: Subtrahend channel
IR, SR, AR, DM, HR, TC, LR, #

R: Result channel
IR, AR, DM, HR, LR

Example circuit
Ladder diagram



Address	Instruction	Data
0000	LD	00000
0001	CLC(41)	
0002	SUB(31)	
		HR04
		#1234
		200
0004	END(01)	

In the above example, before executing SUB, the Carry Flag (special relay 25504) is turned OFF by the Clear Carry (CLC). The addition and subtraction instructions include CY in the calculation as well as in the result. Be sure to clear CY if its previous status is not required in the calculation, and to use the result placed in CY, if required, before it is changed by execution of any other instruction.

The minuend and subtrahend must be in **BCD**, if not special relay 25503 (Error Flag) is turned ON and ADD is not executed.

SUB is executed each time the CPU scans the program. To execute it only once.

❑ Application #14: Measuring The Life Of A Cutting Knife

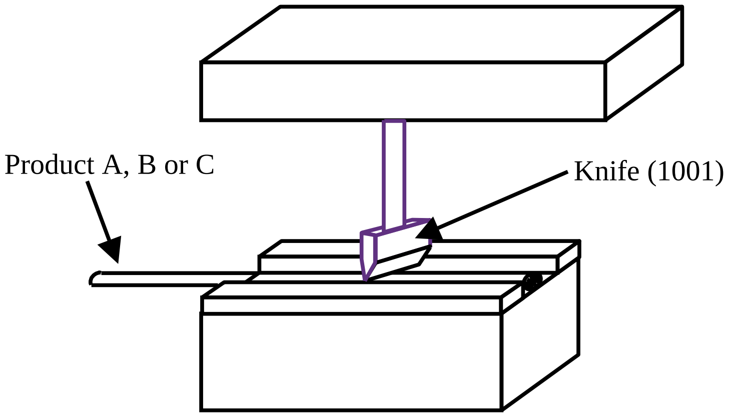
A knife is used to cut 3 products A, B and C and has to be changed after cutting 1000 pieces of A or 500 pieces of B or 100 pieces of C. but the products come at random. A buzzer is sound when the life of the knife is up.

3 sensors are assign to differentiate the 3 products. Another sensor is used to signal cutting completion. A pushbutton to start the process.

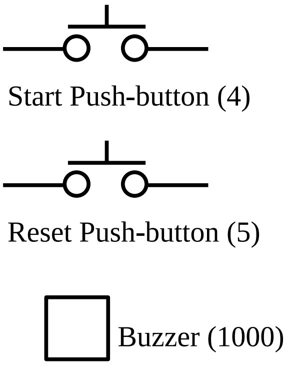
- **I/O Assignment**

Input	Device
00000	Cutter Sensor
00001	Product A
00002	Product B
00003	Product C
00004	Start pushbutton
00005	Reset

Output	Device
01000	Buzzer
01001	Cutter (Knife)



Cutting Machine

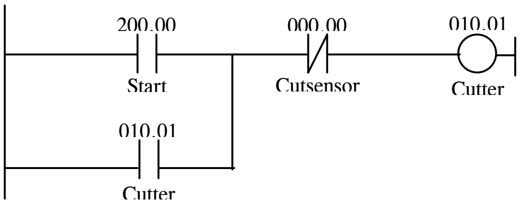


6. APPLICATION EXAMPLES OF BASIC COMMANDS

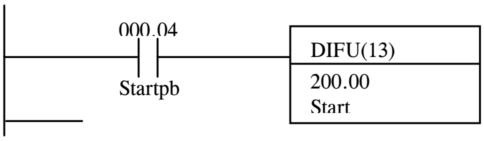
Ladder Diagram : Main 1 Network 1

Main 1 – Knife lift
Application : Measuring the life of a cutting life

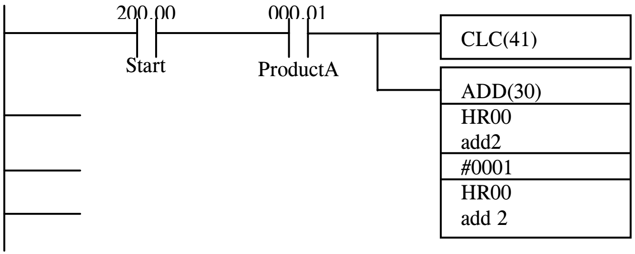
Network 1 – Cutter
Cutter activation



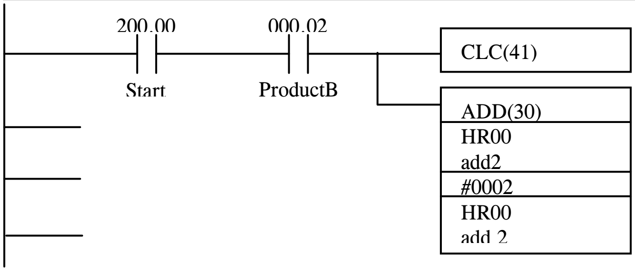
Network 2 – Cutter
Cutter start pushbutton



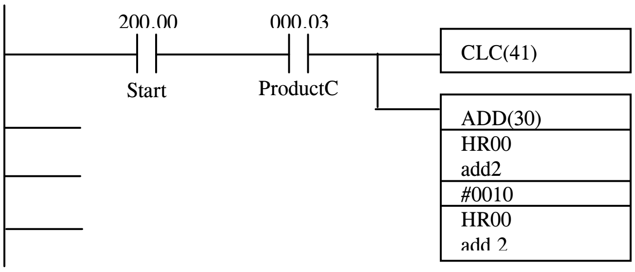
Network 3 – Add 1



Network 4 – Add 2



Network 5 – Add 10

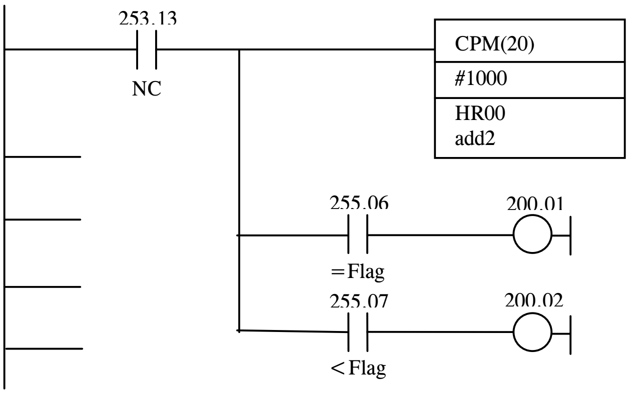


Mnemonic codes

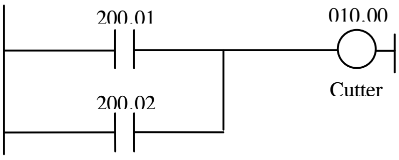
Address	Instruction	Data
0000	LD	20000
0001	OR	01001
0002	AND NOT	00000
0003	OUT	01001
0004	LD	00004
0005	DIFU(13)	20000
0006	LD	20000
0007	AND	00001
0008	CLC(41)	
0009	ADD(30)	
		HR00
		#0001
		HR00
0010	LD	20000
0011	AND	00002
0012	CLC(41)	
0013	ADD(30)	
		HR00
		#0002
		HR00
0014	LD	20000
0015	AND	00003
0016	CLC(41)	
0017	ADD(30)	
		HR00
		#0010
		HR00

6. APPLICATION EXAMPLES OF BASIC COMMANDS

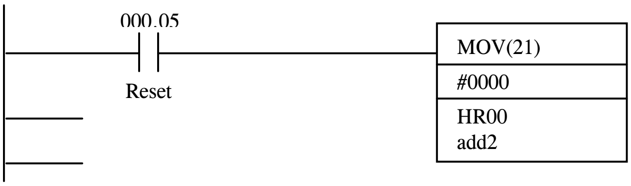
Network 6 – Compare



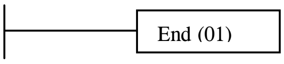
Network 7 – Buzzer



Network 8 – Reset



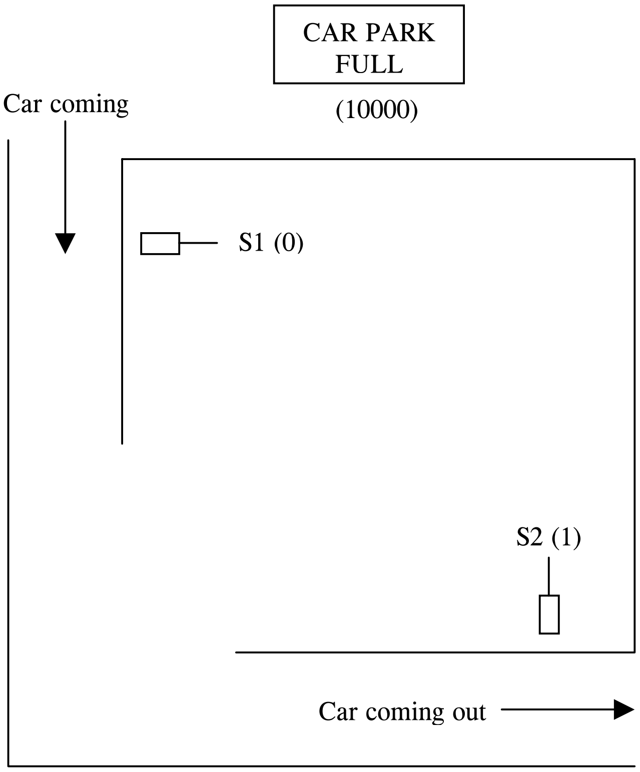
Network 9 – End



Address	Instruction	Data
0015	LD	25313
0016	OUT	TR 0
0017	CMP(20)	-
		#1000
		HR00
0018	AND	25506
019	OUT	20001
0020	LD	TR 0
0021	AND	25507
0022	OUT	20002
0023	LD	20001
0024	OR	20002
0025	OUT	01000
0026	LD	00005
0027	MOV(21)	-
		# 0000
		HR00
0028	END(01)	

❑ Application #15: Car Park Control

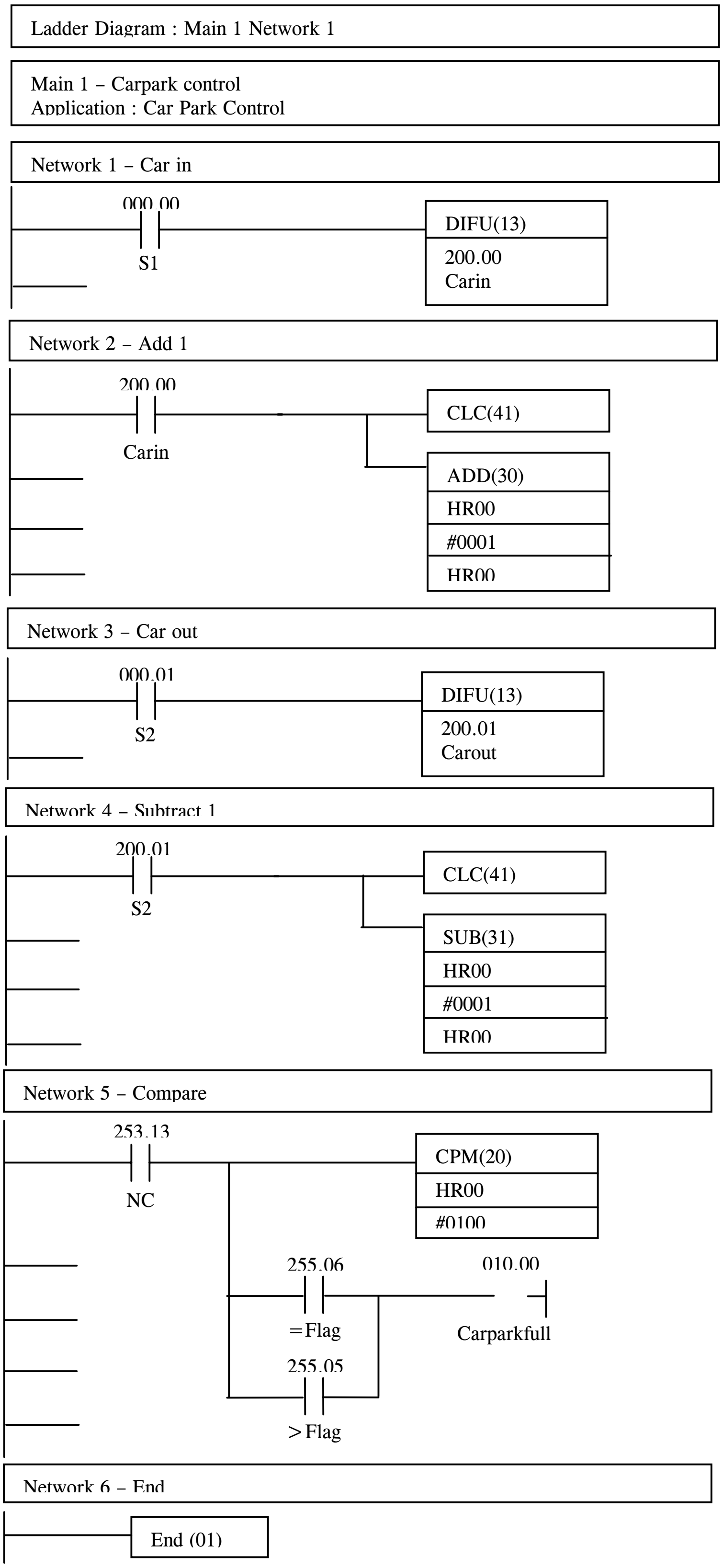
This is a simple car park control system that allows only a maximum of 100 cars parking space. Everytime a car comes in, the PLC will automatically add one through sensor S1. Any car that goes out will automatically be subtracted by one through sensor S2. When 100 cars are registered, the car park full sign will be lighted to inform oncoming vehicles not to enter.



- **I/O Assignment**

Input	Device
00000	Sensor S1
00001	Sensor S2

Output	Device
01000	Car Park Full Sign

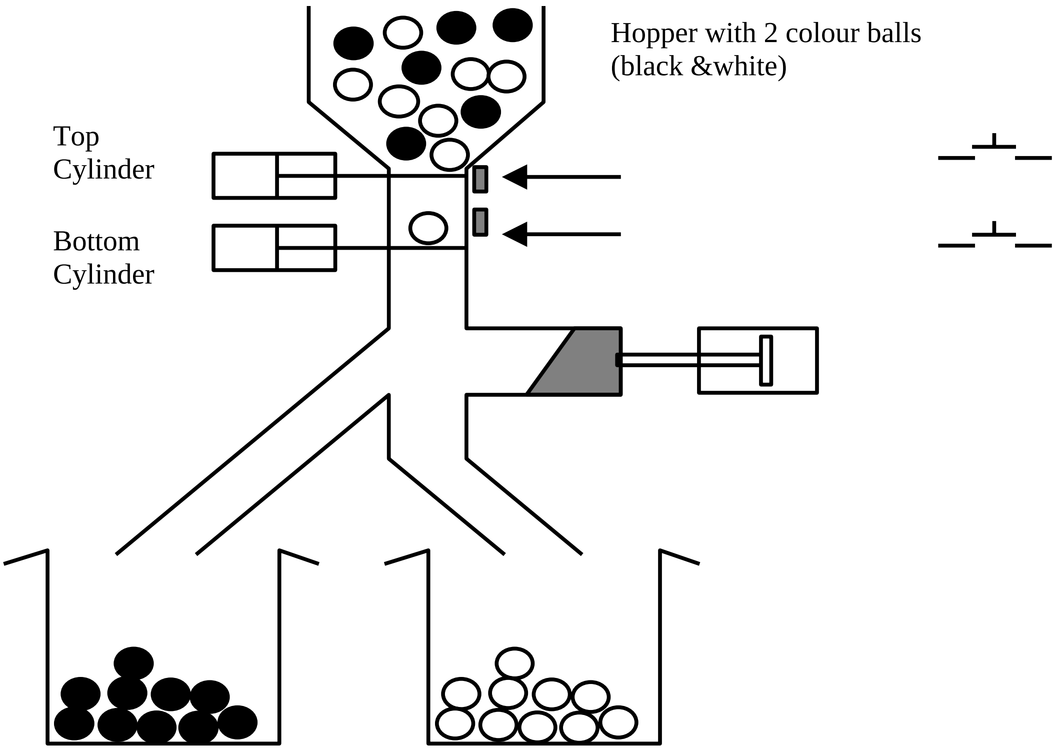


Mnemonic codes

Address	Instruction	Data
0000	LD	00000
0001	DIFU(13)	20000
0002	LD	20000
0003	CLC(41)	
0004	ADD(30)	
		HR00
		#0001
		HR00
0005	LD	00001
0006	DIFU(13)	20001
0007	LD	20001
0008	CLC(41)	
0009	SUB(31)	
		HR00
		#0001
		HR00
0010	LD	25313
0011	OUT	TR 0
0012	CMP(20)	
		HR00
		#0100
0013	AND	25506
0014	LD	TR 0
0015	AND	25505
0016	OR LD	
0017	OUT	01000
0018	END(01)	

6. APPLICATION EXAMPLES OF BASIC COMMANDS

❑ Example: Ball Sorter Mechanism



In this application, the system is to sort out the black & white balls into 2 different container.

The start button will start the operation, Ball sensor (S1) will sense the presence of the ball in the hopper. The top solenoid will release the ball for the colour sensor (S2) to differentiate the colour before being release into the container.

- **I/O Assignment**

Input	Device
00000	Start PB
00001	Ball sensor (S1)
00002	Colour sensor (S2)
00003	Stop PB

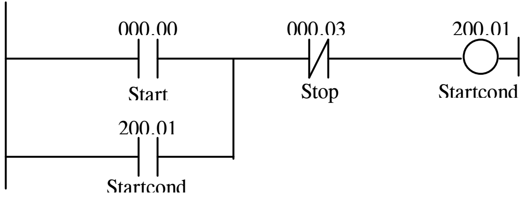
Output	Device
01000	Top cylinder
01001	Bottom cylinder
01002	Pusher

6. APPLICATION EXAMPLES OF BASIC COMMANDS

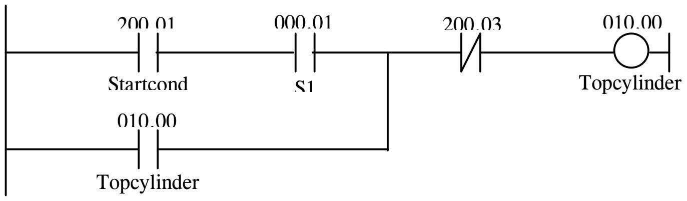
Ladder Diagram : Main 1 Network 1

Main 1 – Ball Sorting (Conventional Method)

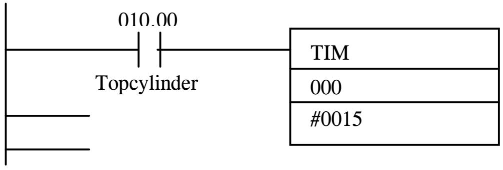
Network 1 – Start cond



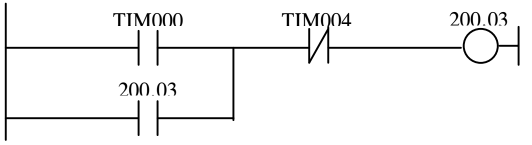
Network 2 – Top cylinder



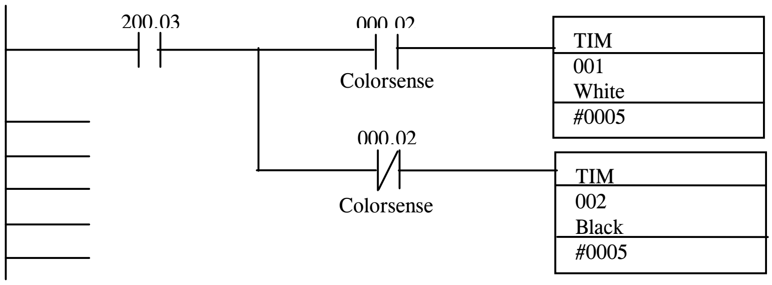
Network 3 – Timer



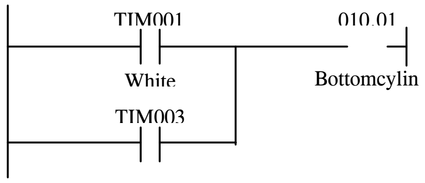
Network 4 – Delay



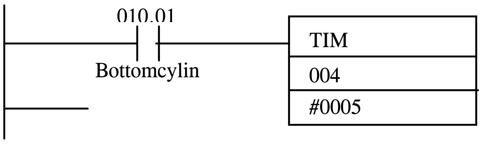
Network 5 – White&black



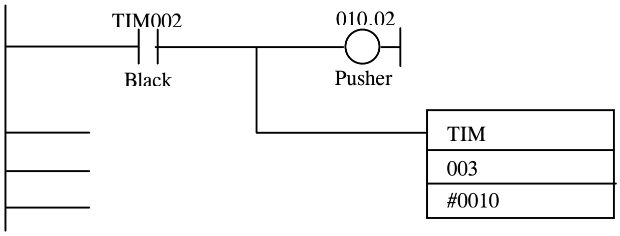
Network 6 - Bottomcylinder



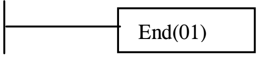
Network 7 – Delay 0.5sec



Network 8 - Pusher



Network 9 - End

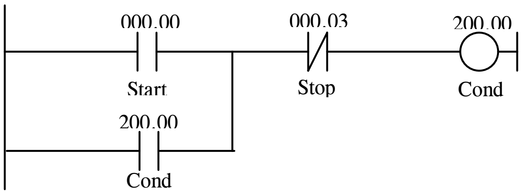


6. APPLICATION EXAMPLES OF BASIC COMMANDS

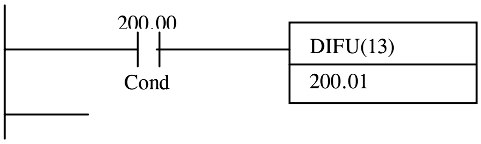
Ladder Diagram : Main 1 Network 1

Main 1 – Balls sorting
This program separates white balls from black balls using the shift register method

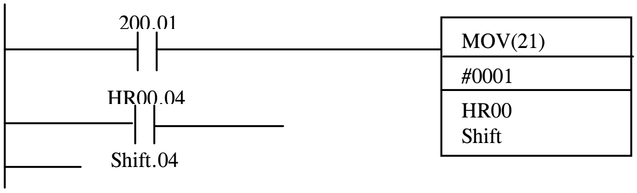
Network 1 – Start



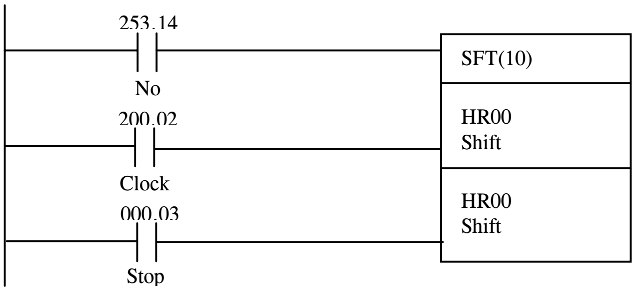
Network 2 –
Differentiation up



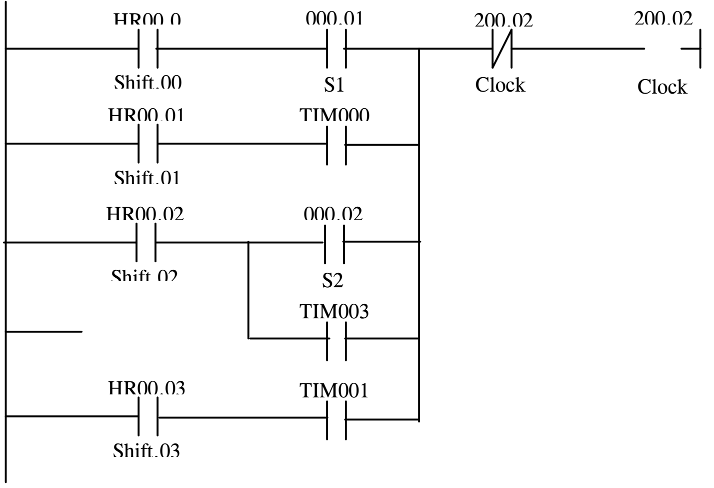
Network 3 – Set bit 1 on
First bit of HR 00 is set



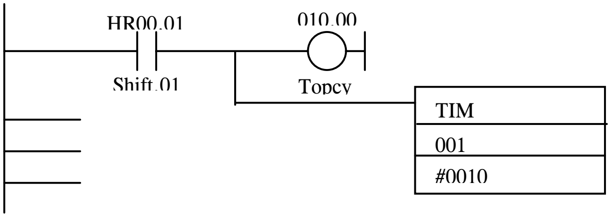
Network 4 – Shift register



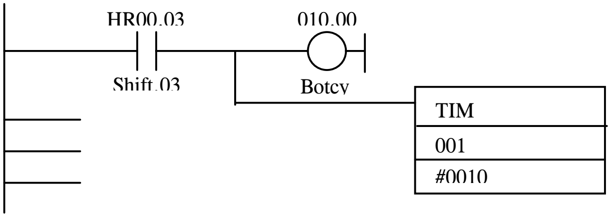
Network 5 – Process
Process sequencing clock input to shift register



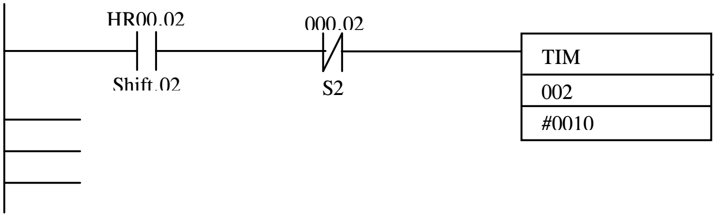
Network 6 – Set bit 1 on



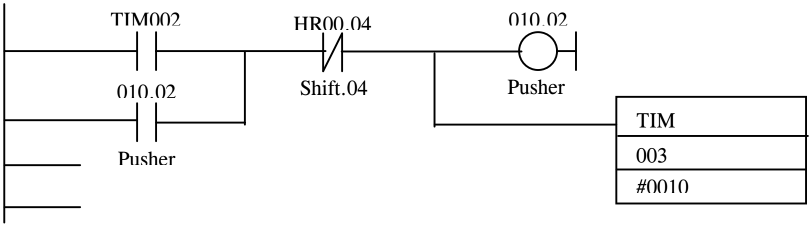
Network 7 – Set bit 1 on



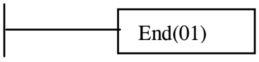
Network 8 – Delay 1sec



Network 9 – Pusher



Network 10 - End

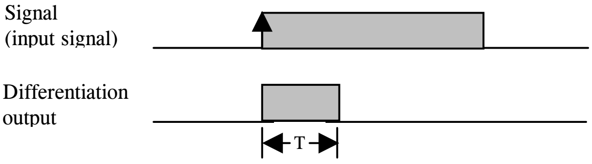


7. USEFUL CIRCUITRY

1 – Cycle Differentiation Circuit

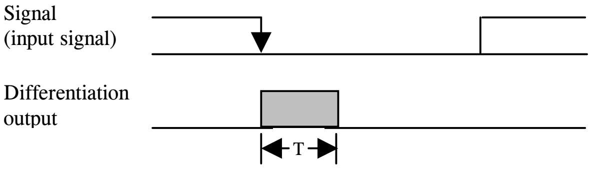
Description

Differentiation up circuit operates for only one cycle time when an input signal turns on (i.e., at the leading edge of the input signal).



T: 1 cycle execution time

Differentiation down circuit operates for only one cycle time when an input signal turns off (i.e., at the falling edge of the input signal).

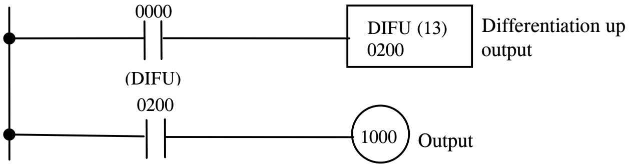


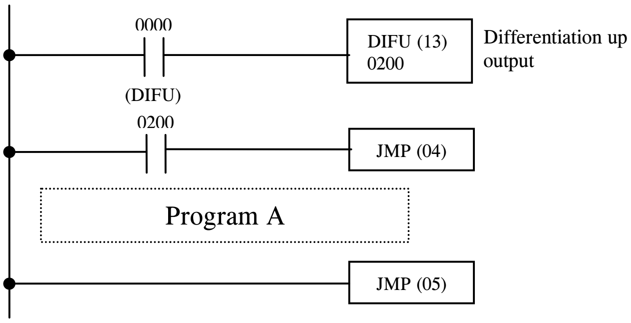
Application example
Using the 1-cycle differentiation up circuit when an arithmetic operation is to be executed only once at the leading edge of an input signal (i.e., when the input signal turns on), when a given program (A, in the example below) is to be executed only once at the leading edge of the input signal, etc. To execute the arithmetic operation, the given program, etc., only once at the trailing edge of an input signal (i.e., when the input signal turns off), use the 1-cycle differentiation down circuit.

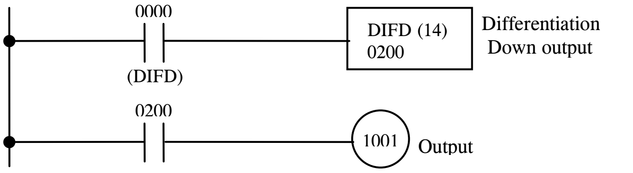
I/O Assignment

Signal (input signal)	0000
Differentiation output	0200
Output relay	1000
Output relay	1001

Circuit Example 1







This circuit causes program A to be executed only once at the leading edge of input signal 0000.

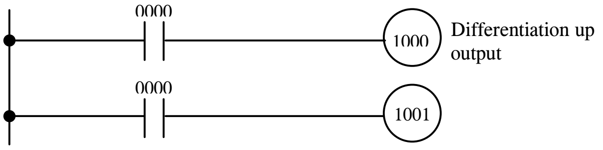
Write the JMP and JME instructions before and after program A, respectively.

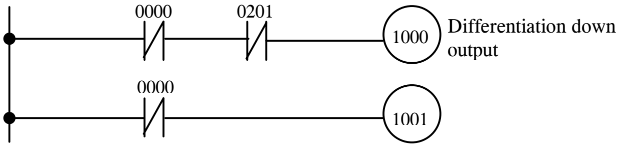
Circuit Example 2

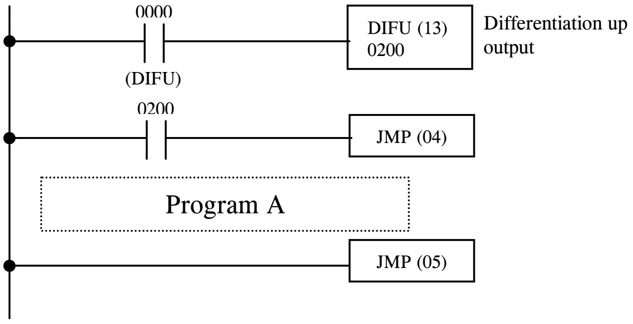
This circuit is an application example of the differentiation circuit using an auxiliary relay.

Signal (input signal)	0000
Differentiation output	1000
Auxiliary relay	1001

7. USEFUL CIRCUITRY







This circuit causes program A to be executed only once at the leading edge of input signal 0000.

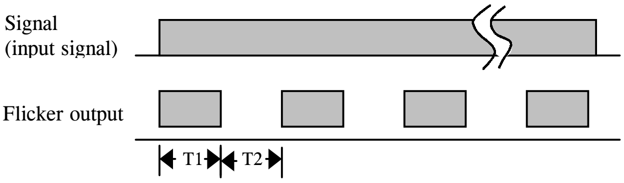
Write the JMP and JME instructions before and after program A, respectively.

7. USEFUL CIRCUITRY

Flicker Circuit

Description

This circuit repeat outputs at specified ON/OFF intervals when a signal (input signal) is applied.



T1, T2: Specified ON and OFF times

Application examples
Use this circuit for the count input of a long-timer (using a CNT instruction), the flickering failure indication of an annunciator, the timing generation of a relay circuit, etc.

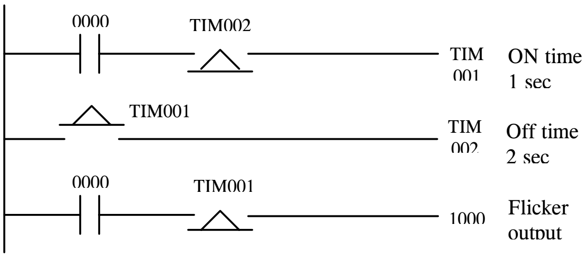
I/O Assignment

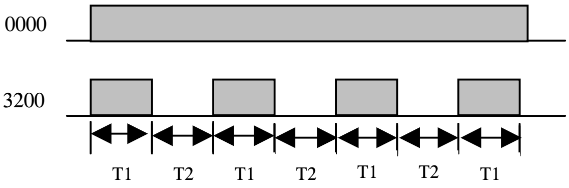
Signal (Input signal)	0000
Flicker output	relay 1000
1-sec clock pulse	Auxiliary relay 25502

Circuit Examples

The circuit examples below are applicable to many PLCs.

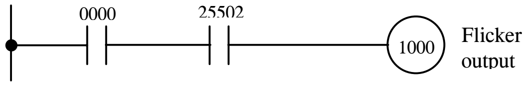
(1) Timers are used.

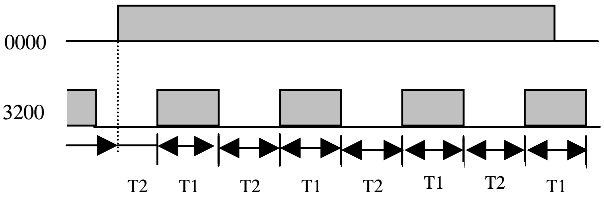




T1: 1 sec
T2: 2sec

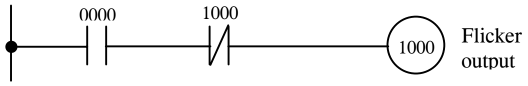
(2) 1-sec clock pulse (Auxiliary relay 25502) is used

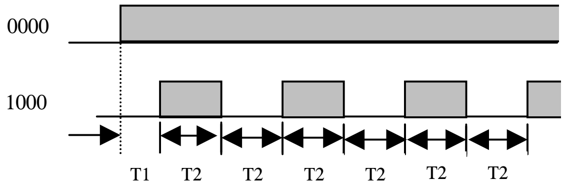




T1: 0.5 sec
T2: This time changes depending on the program position.

(3) 1-cycle execution time is used





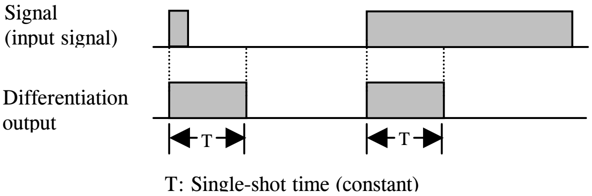
T1: This time changes depending on the program position.
T2: 1-cycle execution time

7. USEFUL CIRCUITRY

Single-shot Circuit

Description

This circuit is used to keep the ON time of a single (input signal) constant.

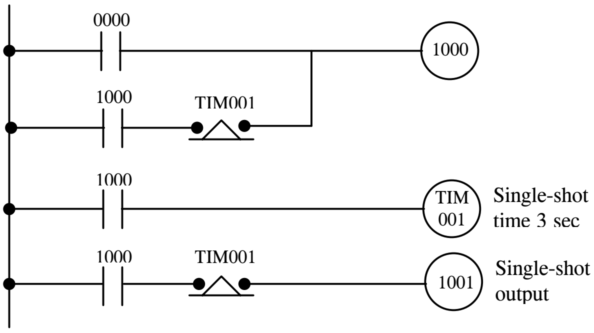


I/O Assignment

Signal (Input signal)	0000
Single-shot output	relay 1001
Output relay 1000	

Circuit Example

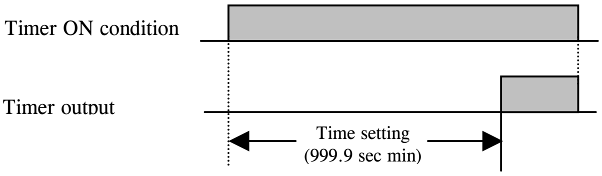
This circuit is applicable to many PLCs.



Long-Time Timer Circuit

Description

The timers can be set to a maximum of 999.9 sec. Use this circuit if the time setting exceeding this maximum is required.



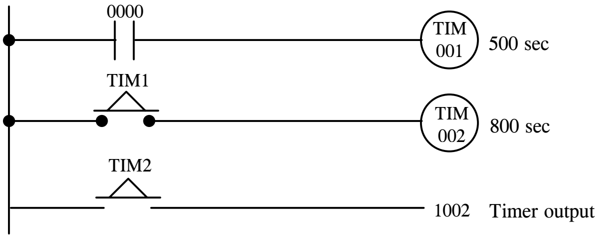
I/O Assignment

Timer ON condition 0000
Timer output: Output relay 1002

Circuit Example

The circuit examples below are applicable to many PLCs.

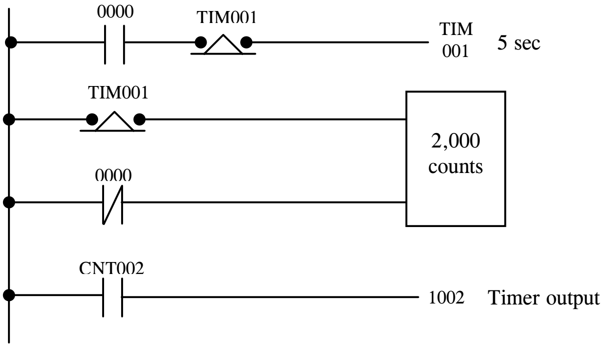
(1) A combination of two timers



Without two timers, a 1,300-sec (500 + 800sec) timer circuit is configured.

A timer circuit of 1999.8 sec (999.9 + 999.9sec) maximum can be configured.

(2) A combination of a timer and a counter

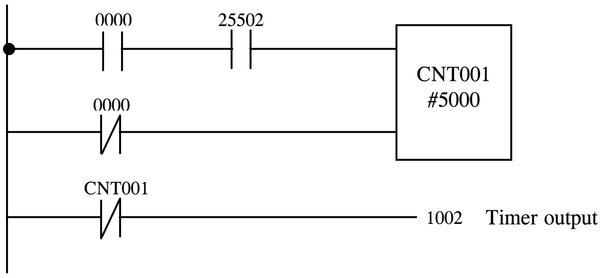


With this combination, a 10,000-sec (5sec x 2,000 counts) timer circuit is configured.

Timer TIM001 is used to generate 5-sec pulses which are then counted by counter CNT002 to 2,000 counts.

When the CNT instruction is used, the present value of the timer is retained during a power failure.

(3) A combination of a 1-sec clock (25502) and a counter (memory retentive type timer)



With this combination, a 5,000-sec timer circuit is configured.

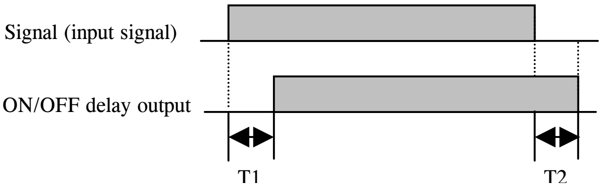
Special auxiliary relay 25502 generates a 1-sec clock.

When the CNT instruction is used, the present value of the timer is retained during a power failure.

ON/OFF-Delay Circuit

Description

This circuit is used to delay the ON/OFF time of a signal (input signal) for a given time.



T1 : ON-delay time
T2 : OFF-delay time

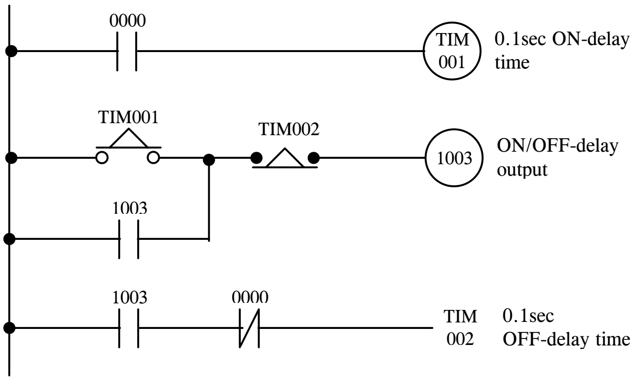
Application example
When a data input (BCD) and a data read input are received simultaneously from external devices, it is necessary to turn on the data read input after data has been accepted. With consideration given to the non-uniformity of response time among the input cards, use an ON-delay circuit for the data read input.

I/O Assignment

Signal (input signal) 0000
ON/OFF-delay output : Output relay 1003

Circuit Example

The circuit examples below are applicable to CPM1

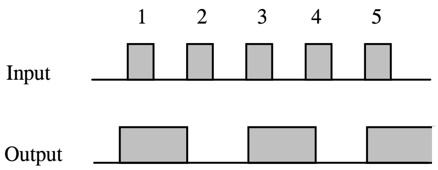


7. USEFUL CIRCUITRY

Push-On/Push-Off circuit (Binary Circuit)

Description

This circuit repeat outputs at specified ON/OFF intervals when a signal (input signal) is applied.

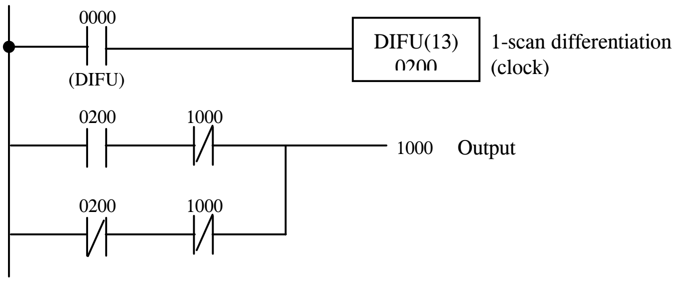


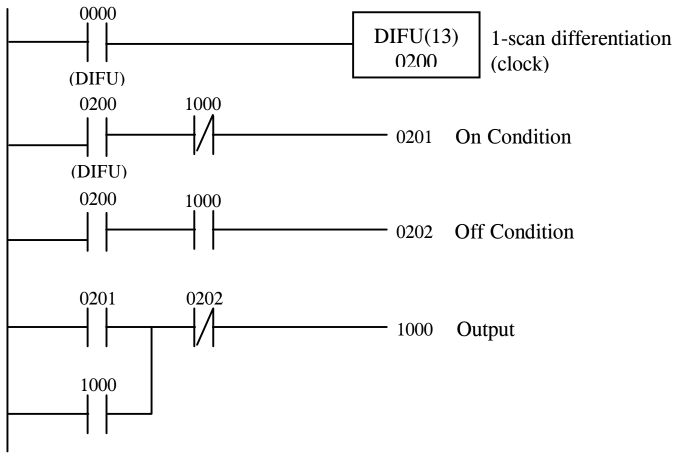
I/O Assignment

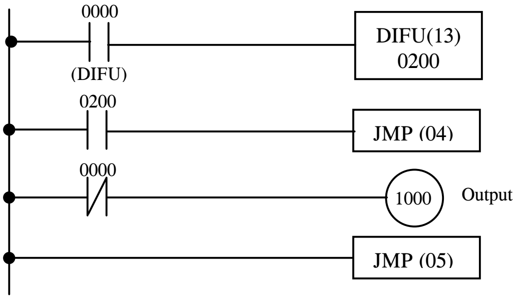
Input	0000
Output	1000
Work bit	0200
	0201
	0202

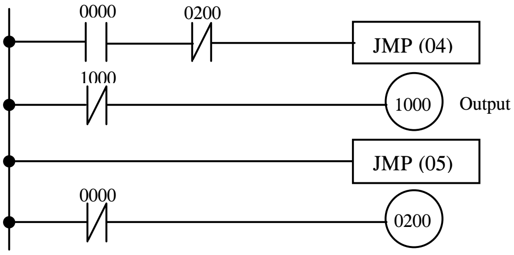
Circuit Examples

When data retention during power failure is unnecessary

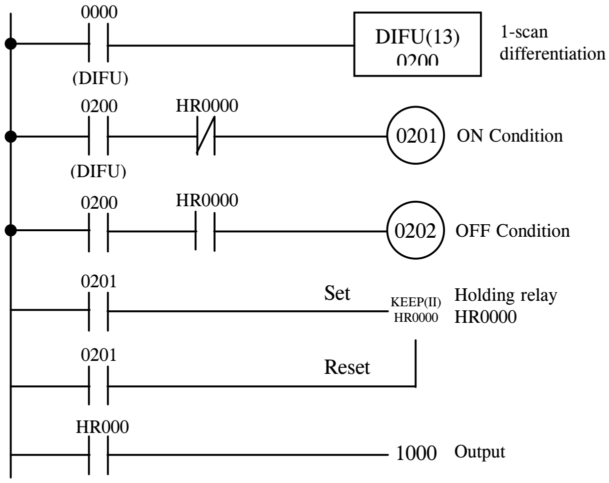


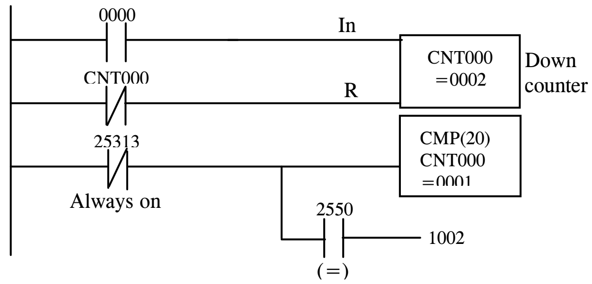






When data retention during power failure is unnecessary



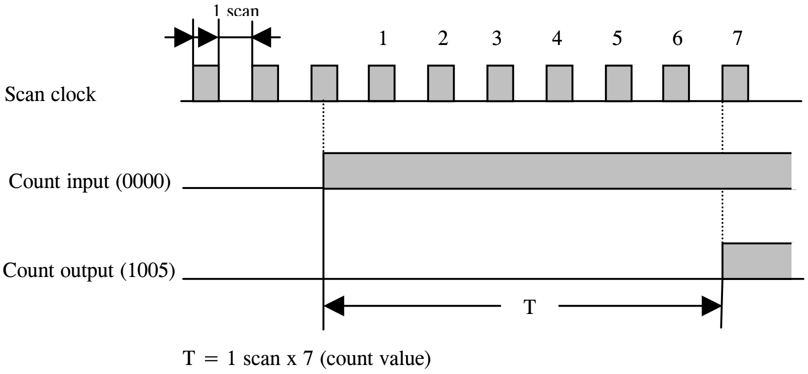


Auxiliary relay 25313 is normally ON. Auxiliary relay 25506 turns ON if the result when the Compare (CMP) instruction is executed is equal.

Scanning Counter

Description

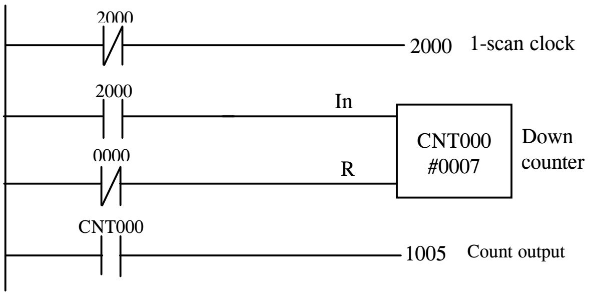
This circuit is used to count scan clocks to obtain timing at a very precise pulse duration.



I/O Assignment

Count input	0000
Count output	1005
Work bit	2000

Circuit Example



7. USEFUL CIRCUITRY

Concept of IL Instruction and JMP instruction

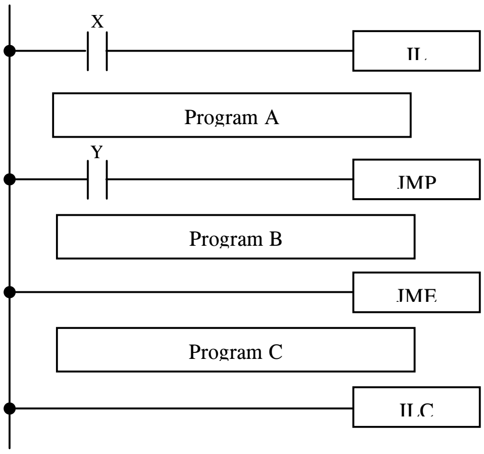
Description

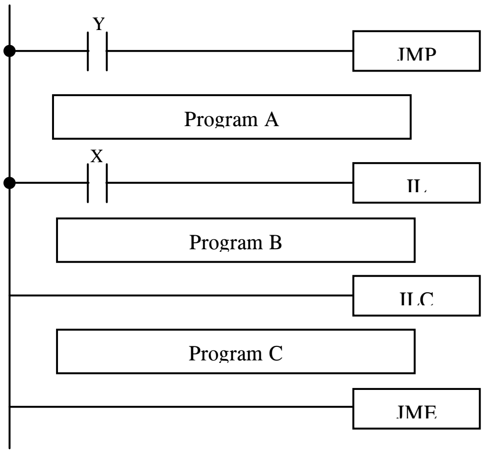
The IL instruction must always be used in conjunction with an ILC instruction, e.g., IL-ILC or IL-IL-ILC. When the IL condition is logical 1, the programs between the IL and IC instructions are executed according to the ladder diagram. When the IL condition is logical 0, all the output relays, internal auxiliary relays and timers in the programs between the IL and ILC instructions are turned OFF, and any counters, shift registers, holding relays and data memory relays in the same programs are held in their present status.

The JMP instruction must always be used in conjunction with a JMP instruction, e.g., JMP-JMP or JMP-JMP-JME. When the JMP condition is logical 1, the programs between the JMP and JME instructions are executed according to the ladder diagram. When the JMP condition is logical0, all the output relays, internal auxiliary relays, timers, counters, shift register, holding relays and data memory relays in the same programs are held in their present status.

Combination of IL and JMP instructions with combinations such as IL-JMP-ILC-JME and JMP-IL-JME-ILC, the CPU cannot execute programs properly and must therefore be avoided. However, with combinations such as IL-JMP-JME-ILC and JMP-IL-ILC-JME, the CPU performs program execution without problem.

Circuit Example





Legend

X: IL condition
Y: JMP condition
A: The programs are executed according to the ladder diagram.
B: Output relays, internal auxiliary relays and timers are turned off; counters, shift registers, holding relays and data memory relays are held in their present status.
C: Output relays, internals auxiliary relays, timers, relays are held in their present status.

Concicion		IL-JMP-JME-ILC			JMP-IL-ILC-JMP		
		Program A	Program B	Program C	Program A	Program B	Program C
X	“1”	A	A	A	A	A	A
Y	“1”						
X	1▶1	A	C	A	C	C	C
Y	1▶0						
X	1▶0	B	B	B	A	B	A
Y	1▶1						
X	1▶0	B	B	B	C	C	C
Y	0▶0						
X	0▶0	B	B	B	C	B	C
Y	1▶0						
X	0	B	B	B	C	C	C
Y	0						
X	0▶1	A	C	A	C	C	C
Y	0▶0						
X	0▶0	B	B	B	A	B	A
Y	0▶1						
X	1▶1	A	A	A	A	A	A
Y	0▶1						
X	0▶1	A	A	A	A	A	A
Y	0▶1						

7. USEFUL CIRCUITRY

First-In Input Priority Circuit

Description

When there are plural inputs, this circuit is used to accept only the first input and ignore all subsequent inputs. This circuit accepts only the first input after it has been cleared by a reset input.

I/O Assignment

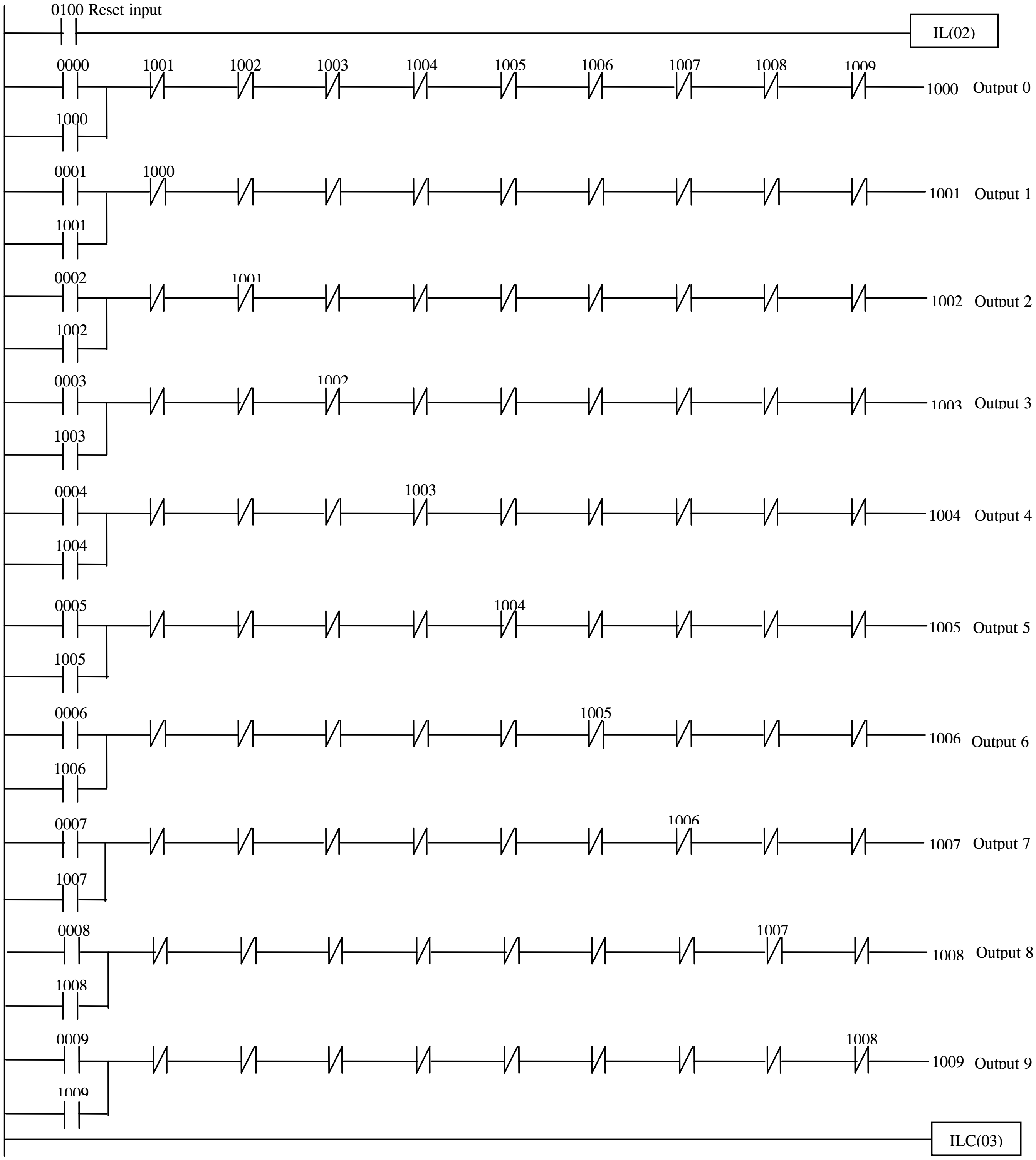
Input	Device
0000	Input 0
0001	Input 1
0002	Input 2
0003	Input 3
0004	Input 4
0005	Input 5
0006	Input 6
0007	Input 7
0008	Input 8
0009	Input 9
0010	Reset Input

Output	Device
1000	Output 0
1001	Output 1
1002	Output 2
1003	Output 3
1004	Output 4
1005	Output 5
1006	Output 6
1007	Output 7
1008	Output 8
1009	Output 9

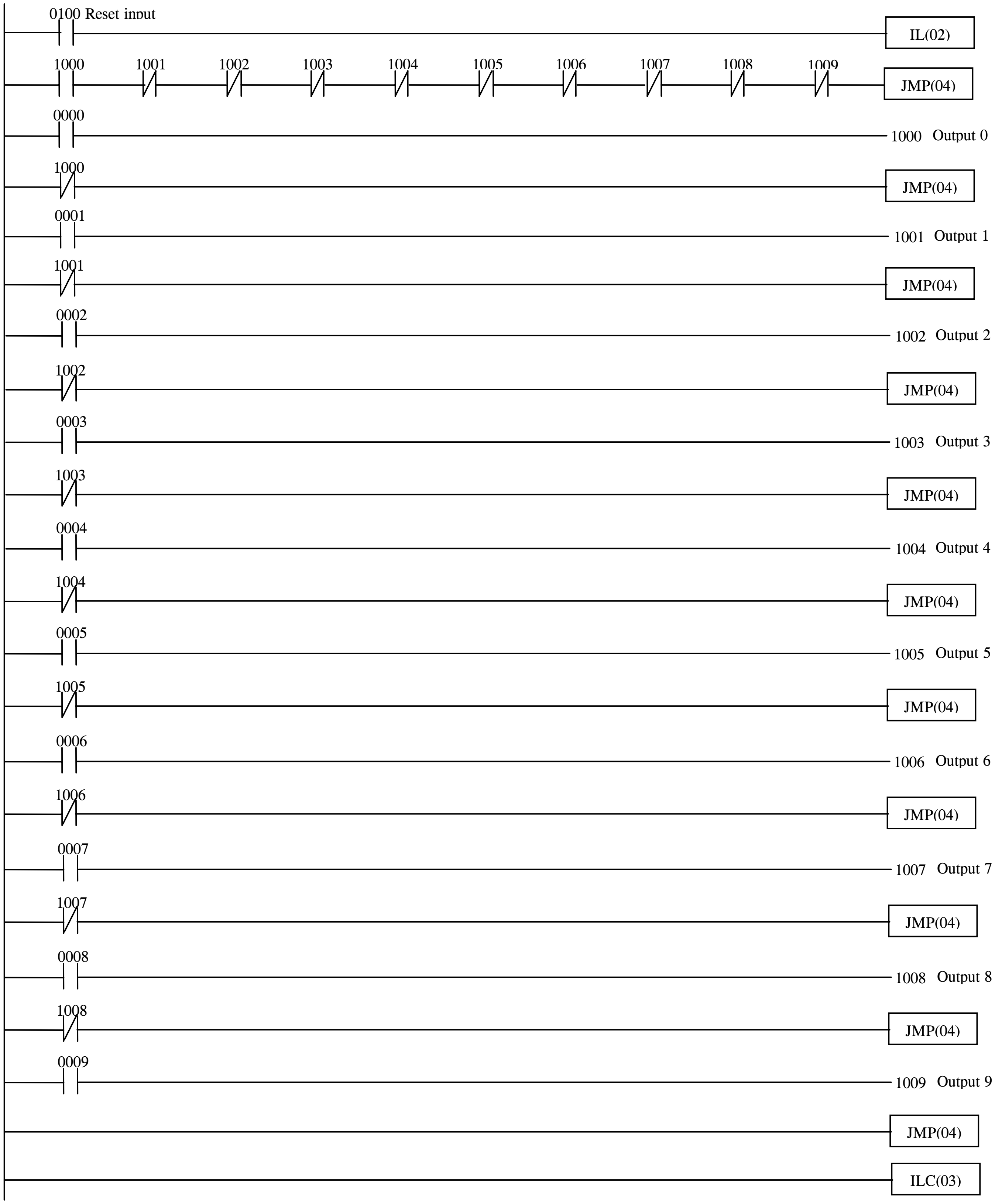
7. USEFUL CIRCUITRY

Circuit Example

Example 1
This circuit is applicable to CPM1A



Example 2


0100 Reset input
IL(02)
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
JMP(04)
0000
1000 Output 0
1000
JMP(04)
0001
1001 Output 1
1001
JMP(04)
0002
1002 Output 2
1002
JMP(04)
0003
1003 Output 3
1003
JMP(04)
0004
1004 Output 4
1004
JMP(04)
0005
1005 Output 5
1005
JMP(04)
0006
1006 Output 6
1006
JMP(04)
0007
1007 Output 7
1007
JMP(04)
0008
1008 Output 8
1008
JMP(04)
0009
1009 Output 9
JMP(04)
ILC(03)

Last-In Input Priority Circuit

Description

When there are plural inputs, this circuit is used to accept only the first input and clear all the preceding inputs. This is cleared by a reset input.

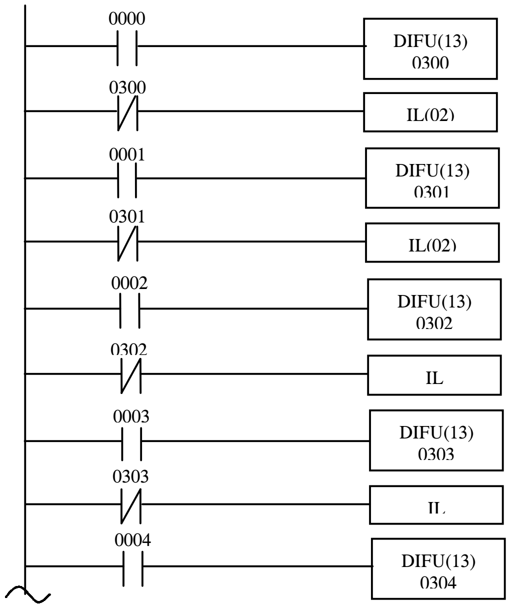
I/O Assignment

Input	Device
0000	Input 0
0001	Input 1
0002	Input 2
0003	Input 3
0004	Input 4
0005	Input 5
0006	Input 6
0007	Input 7
0008	Input 8
0009	Input 9
0010	Input 10
0011	Input 11
0012	Input 12
0013	Input 13
0014	Input 14
0015	Input 15

Output	Device
1000	Output 0
1001	Output 1
1002	Output 2
1003	Output 3
1004	Output 4
1005	Output 5
1006	Output 6
1007	Output 7
1008	Output 8
1009	Output 9
1010	Output 10
1011	Output 11
1012	Output 12
1013	Output 13
1014	Output 14
1015	Output 15

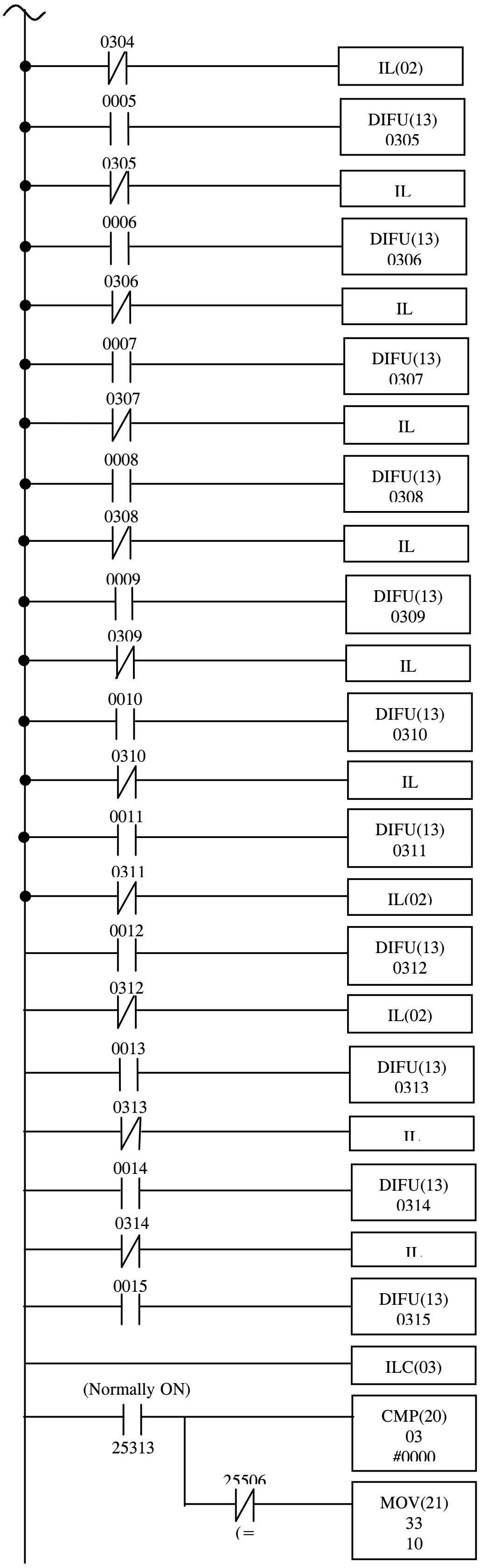
Others	
0300 to 0315	Auxillary relay

Circuit Example



This circuit outputs the differentiation of inputs 0 to 15 (i.e.,0000 to 0015).

For the differentiation outputs, auxiliary relays 0300 to 0315 are used.



When inputs 0 to 15 (0000 to 0015) turn ON within the same cycle, the input with the lowest program address number takes precedence over the other inputs.

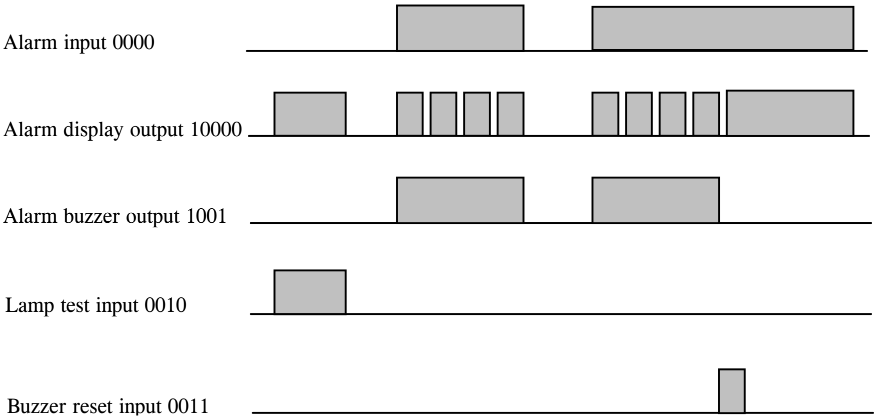
If the differentiation outputs are 0000, auxiliary relay 25506 turns ON.

If the differentiation outputs are not 0000, the data in auxiliary relay channel 03 are transferred to output relay channel 10. Therefore, the data in auxiliary relays 0300 to 0315 are transferred to 1000 to 1015 and the latter group of auxiliary relays are the outputs.

Non-Lock-In Annunicator

Description

This circuit is used to generate alarm display outputs if a failure occurs.



One-point Non-lock-in Annunicator

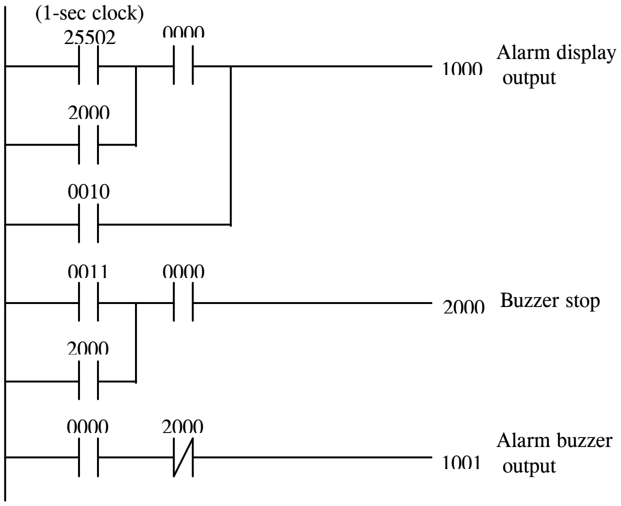
(1) I/O assignment

Input	Device
0000	Alarm input
0010	Lamp test input
0011	Buzzer reset input

Output	Device
1000	Alarm display output
1001	Alarm buzzer output

Others	
25502	1-sec clock
2000	Work bit

(2) Circuit example
This circuit is applicable to CPM1A



7. USEFUL CIRCUITRY

16-point Non-lock-in Annunicator

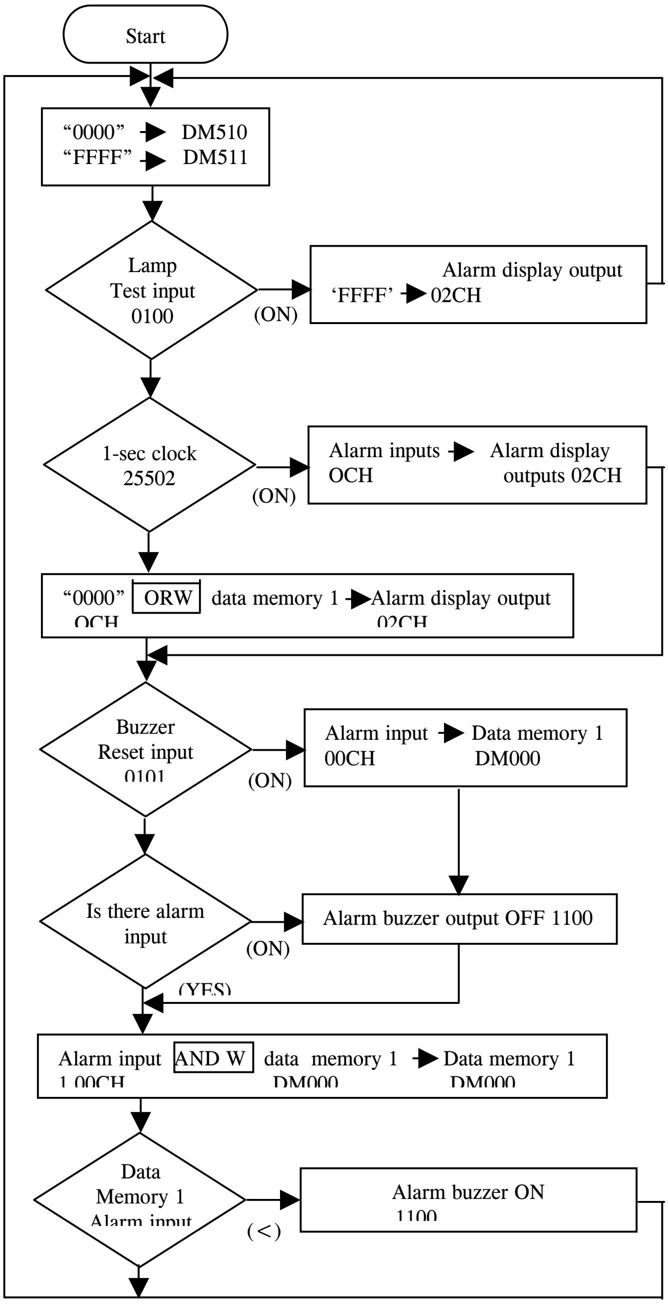
(1) I/O assignment

Input	Device
0000 to 0015 (16 points)	Alarm input
0100	Lamp test input
0101	Buzzer reset input

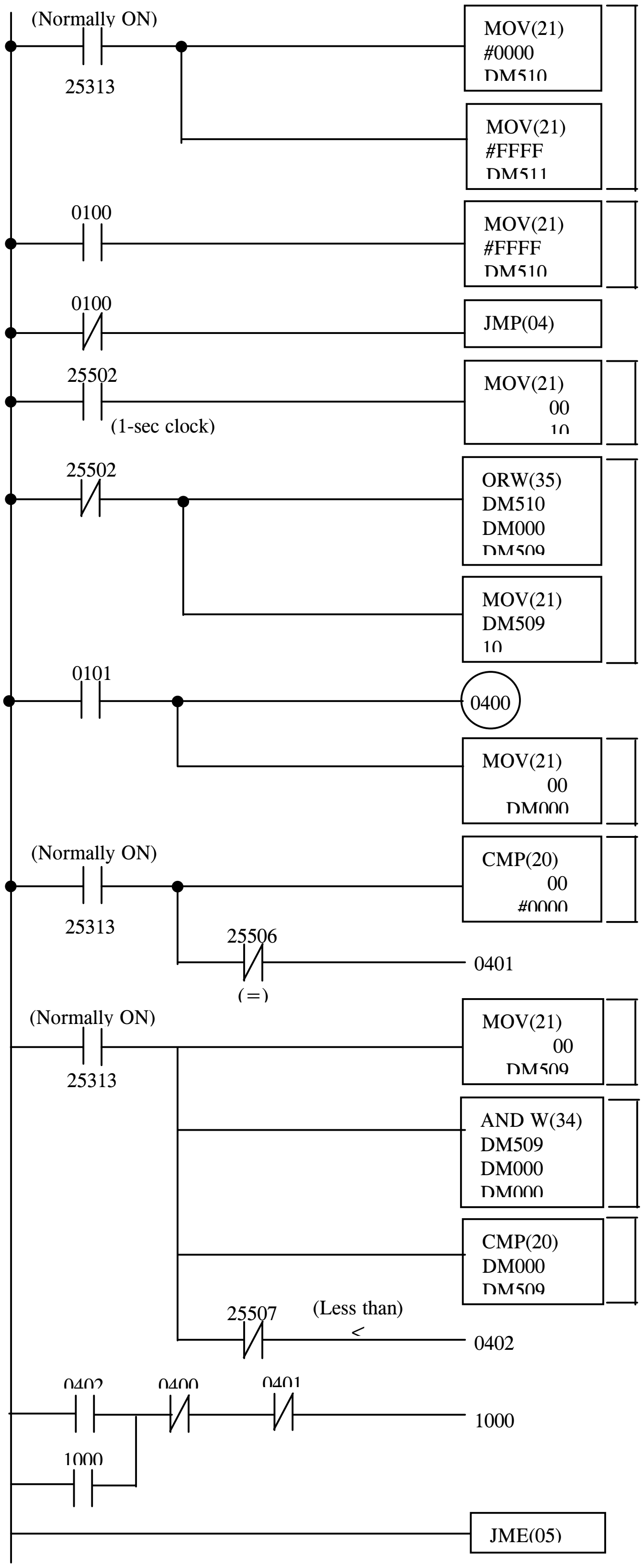
Output	Device
1000 to 1015 (16 points)	Alarm display output
1100	Alarm buzzer output

Others	
25502	1-sec clock
25313	Normally ON relay
25506	Equal (=)
25507	Less than (<)
DM000	Data memory 1
DM509	Work area
	Constant
DM510 "0000"	Data 1
DM511 "FFFF"	Data 2
0400 to 0402	Auxiliary relays

(2) Programming concept (Flowchart)


Start
"0000" → DM510
"FFFF" → DM511
Lamp Test input 0100
(ON)
Alarm display output 'FFFF' → 02CH
1-sec clock 25502
(ON)
Alarm inputs OCH → Alarm display outputs 02CH
"0000" OCH ORW data memory 1 → Alarm display output 02CH
Buzzer Reset input 0101
(ON)
Alarm input 00CH → Data memory 1 DM000
Is there alarm input
(ON)
Alarm buzzer output OFF 1100
(YES)
Alarm input 1 00CH AND W data memory 1 DM000 → Data memory 1 DM000
Data Memory 1 Alarm input
(<)
Alarm buzzer ON 1100

(2) Circuit Example



Constant “0000” and “FFFF” are store in DM510 and DM511, respectively.

Data “FFFF” is transferred to the display output (02CH).

When the 1-sec clock (auxiliary relay 25502) is ON, the contents of alarm input (00CH) are transferred to the display output (02CH).

When the 1-sec clock (auxiliary relay 25502) is OFF, data “0000” is Ored with the contents of DM000 and the result of the OR operation is transferred to the alarm display output (02CH).

Alarm buzzer output

When the buzzer reset input (0101) is ON, the contents of the alarm input (00CH) are stored in DM000.

Check if all the contents of the alarm input (00CH) are “0000”. If so, auxiliary relay 25506 is turned ON.

Alarm buzzer output

The contents of alarm input (00CH) are stored in data memory DM509.

When the buzzer reset input (0101) is ON, the old contents of alarm input which were stored in DM000 ANDed with the current contents of alarm input (00CH) and the result is stored in DM000. If the status of the alarm input is changed, auxiliary relay 25507 turns ON.

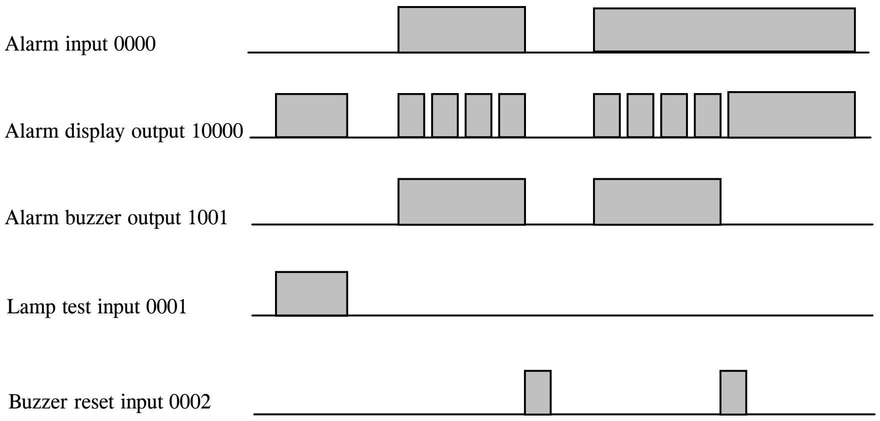
Alarm buzzer output

Alarm buzzer output.

Lock-In Annunicator

Description

This circuit is used to generate alarm display outputs.



One-point lock-in Annunicator

(1) I/O assignment

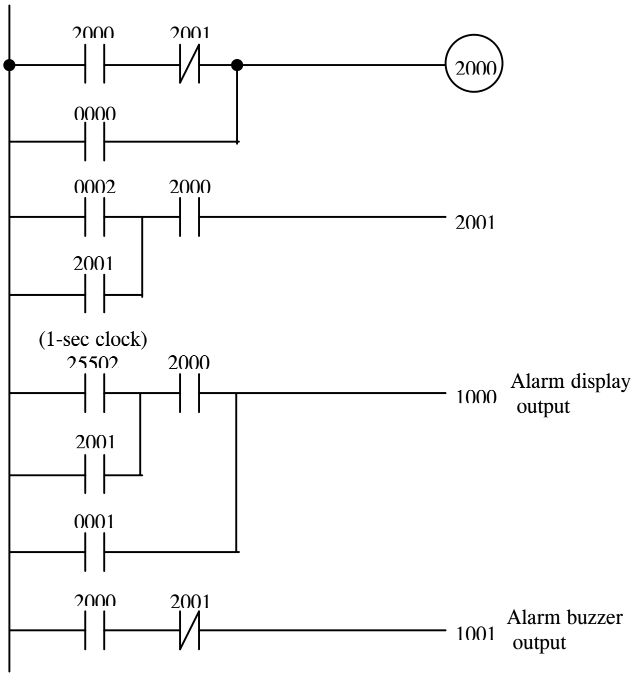
Input	Device
0000	Alarm input
0100	Lamp test input
0110	Buzzer reset input

Output	Device
1000	Alarm display output
1001	Alarm buzzer output

Others	
25502	1-sec clock
2000	Work bits
2001	

(2) Circuit example

This circuit is applicable to CPM1A



7. USEFUL CIRCUITRY

16-point lock-in Annunicator

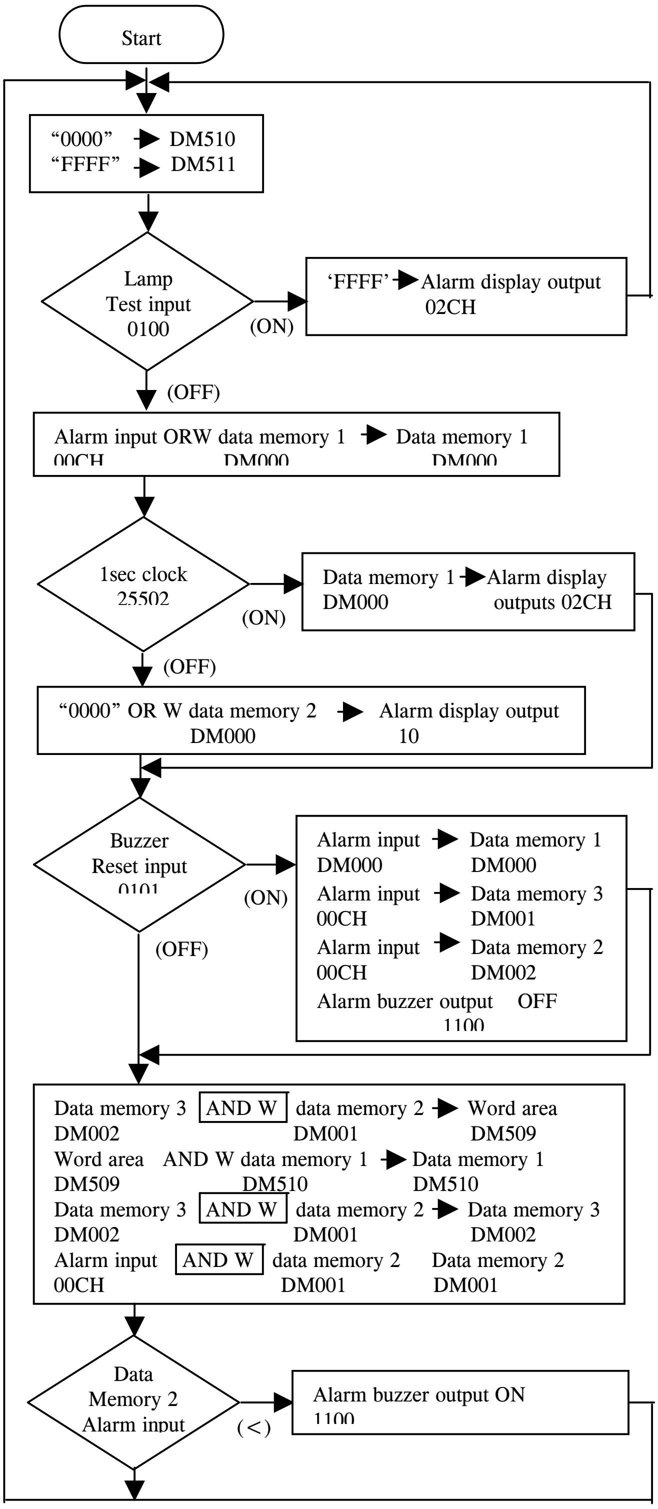
(1) I/O assignment

Input	Device
0000 to 0015 (16 points)	Alarm input
0100	Lamp test input
0101	Buzzer reset input

Output	Device
1000 to 1015 (16 points)	Alarm display output
1100	Alarm buzzer output

Others	
25502	1-sec clock
25313	Normally ON relay
25506	Equal (=)
25507	Less than (<)
DM000	Data memory 1
DM001	Data memory 2
DM002	Data memory 3
DM509	Work area
DM510 “0000”	Data 1
DM511 “FFFF”	Data 2
0200 to 0201	Auxiliary relays

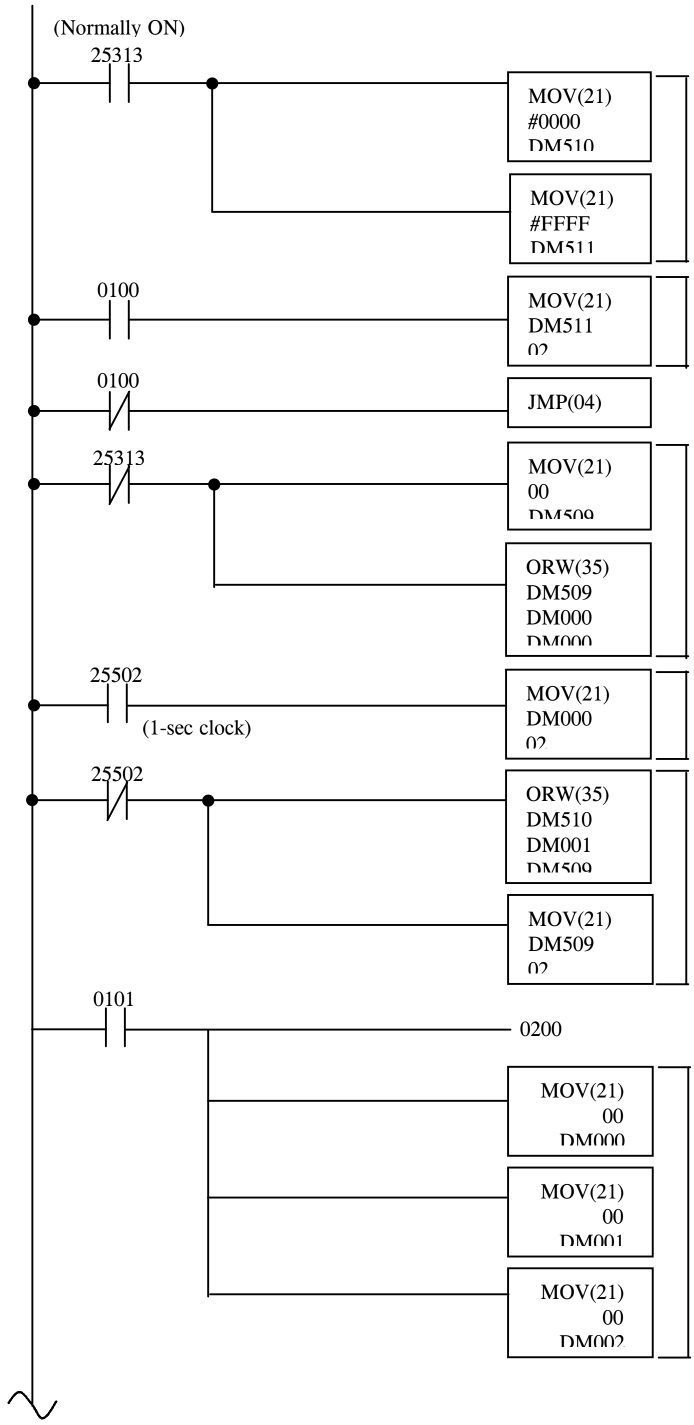
(2) Programming concept (Flowchart)



Start
“0000” → DM510
“FFFF” → DM511
Lamp Test input 0100
(ON)
‘FFFF’ → Alarm display output 02CH
(OFF)
Alarm input ORW data memory 1 → Data memory 1
00CH DM000 DM000
1sec clock 25502
(ON)
Data memory 1 → Alarm display
DM000 outputs 02CH
(OFF)
“0000” OR W data memory 2 → Alarm display output
DM000 10
Buzzer Reset input 0101
(ON)
Alarm input → Data memory 1
DM000 DM000
Alarm input → Data memory 3
00CH DM001
Alarm input → Data memory 2
00CH DM002
Alarm buzzer output OFF
1100
(OFF)
Data memory 3 AND W data memory 2 → Word area
DM002 DM001 DM509
Word area AND W data memory 1 → Data memory 1
DM509 DM510 DM510
Data memory 3 AND W data memory 2 → Data memory 3
DM002 DM001 DM002
Alarm input AND W data memory 2 Data memory 2
00CH DM001 DM001
Data Memory 2 Alarm input
(<)
Alarm buzzer output ON
1100

7. USEFUL CIRCUITRY

(1) Circuit Example



Constant "0000" and "FFFF" are store in DM510 and DM511, respectively.

Data "FFFF" is transferred to the alarm display output (02CH).

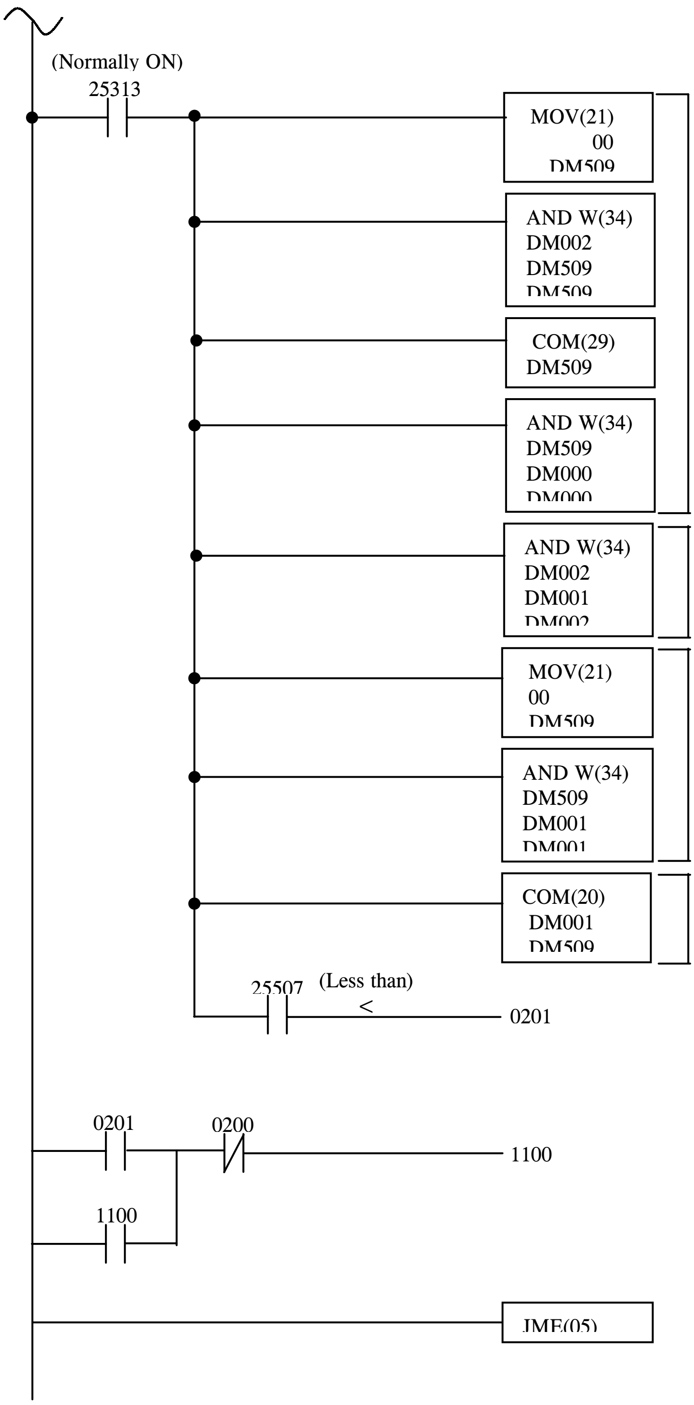
The contents of the alarm input (00CH) are stored in the data memory 1 (DM000).

When the 1-sec clock (25502) is ON, the contents of the data memory 1 (DM0000) are transferred to the alarm display output (02CH).

When the 1-sec clock (25502) is OFF, data "0000" is Ored with the contents of the memory 2 (DM001) and the result of the OR operation is transferred to the alarm display output (02CH).

Alarm buzzer output OFF.

When the buzzer reset input (0101) is ON, the contents of the alarm input (00CH) are stored in the data memories 1 to 3 (DM000 to DM 002).



The contents of the data memory 1 (DM000) are rewritten.

The contents of the data memory 3 (DM002) are rewritten.

The contents of the data memory 2 (DM001) are rewritten.

Checks the status of the alarm input (00CH) for any change. If any change exists, auxiliary relay 25507 is turned ON.

Alarm buzzer output ON.

Alarm buzzer output.

8. INSTALLATION AND WIRING CONSIDERATIONS

❑ PLC Power Interruptions

Supply Voltage Drop

When the supply voltage falls below 85% of the rated value, the PLC stops and the output goes OFF.

Momentary Power Failure Detection

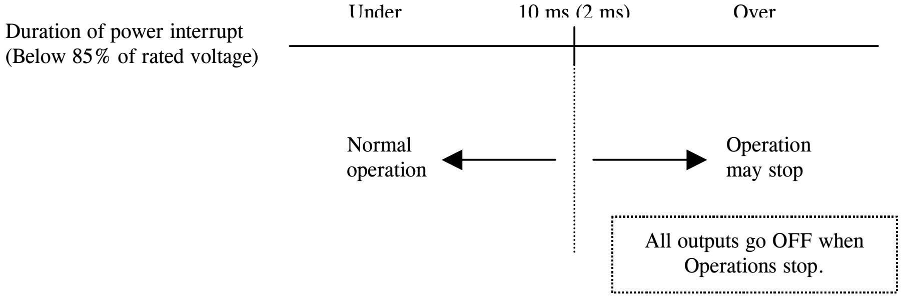
A momentary power failure lasting less than 10 ms with an AC power supply and 2 ms with a DC power supply is not detected and the CPU continues to operate.

A momentary power failure lasting longer than 10 ms with an AC power supply and 2 ms with a DC power supply may or may not be detected in an uncertain area.

When a momentary power failure is detected, the CPU stops operating and the output goes OFF.

Automatic Restart

When the supply voltage recovers to a value higher than 85% of the rated value, operations resumes automatically.


Under
10 ms (2 ms)
Over
Duration of power interrupt
(Below 85% of rated voltage)
Normal operation
Operation may stop
All outputs go OFF when Operations stop.

Note The PLC may repeat stop/start operations if the supply voltage of less than 85% of the rated value gradually goes up or down.
If this affects the equipment, etc., provide a protection circuit which shuts off the output if the supply voltage is not above the rated value.

Time Up to Start of Operation

The time from when the power supply is turned on to when the operation starts varies depending on the operation conditions such as power supply voltage, configuration, ambient temperature, etc. The minimum time is approximately 300 ms.

8. INSTALLATION AND WIRING CONSIDERATIONS

❑ Installation Site Considerations

The PLC is resistant to harsh conditions and highly reliable, but installing the PLC in a favourable site will maximize its reliability and operating lifetime.

• Installation site conditions

Avoid installing the PLC in a site with any of the following conditions.

- A site in direct sunlight
- A site where the ambient temperature exceeds the 0°C to 55°C range
- A site where the relative humidity exceeds the 10% to 90% RH range
- A site where condensation occurs due to sudden temperature changes
- A site with corrosive gases, flammable gasses or salt
- A site with water, oil, or chemical sprays
- A site subjected to direct vibration or shock

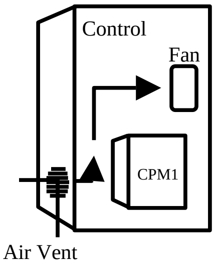
Be sure that the conditions at the installation site conform to the PLC's general specifications.

❑ Panel/Cabinet Installation

Consider PLC operation, maintenance, and surrounding conditions when installing the PLC in a panel or cabinet.

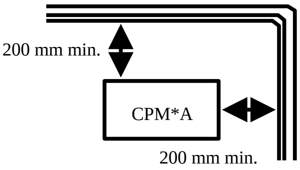
Overheating
The operation temperature range for the PLC is 0°C to 55°C. Be sure that there is adequate ventilation for cooling.

- Allow enough space for air circulation.
- Do not install the PLC above equipment that generates a large amount of heat, such as heaters, transformers, or large resistors.
- Install a cooling fan or system when the ambient temperature exceeds 55°C.



Electrical Noise
Power lines and high-voltage equipment can cause electrical noise in the PLC.

- Do not install the PLC in a panel or cabinet with high-voltage equipment.
- Allow at least 200 mm between the PLC and nearby power lines.



CPM*A-CPM1A and CPM2A.

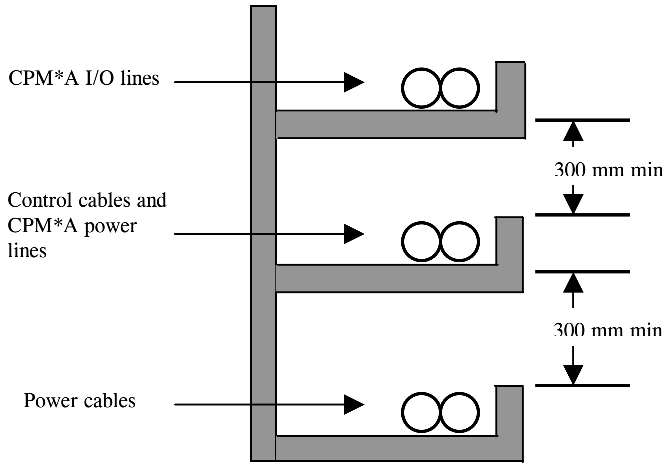
8. INSTALLATION AND WIRING CONSIDERATIONS

❑ General Precautions for Wiring

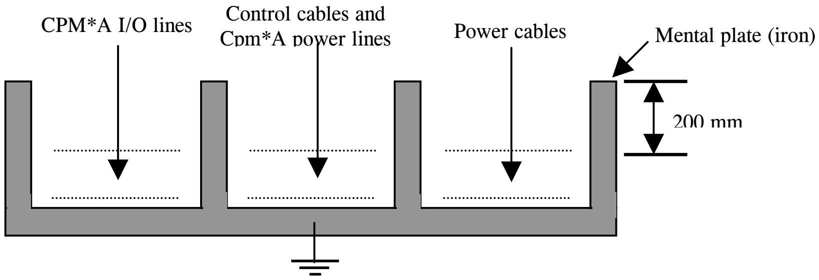
I/O Lines Noise

Do not run CPM1A I/O lines in the same duct or conduit as power lines.

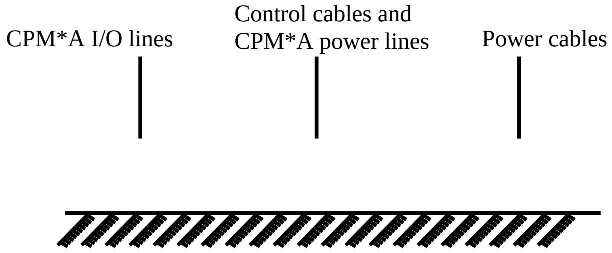
Handing Ducts
Leave at least 300 mm between the power cables and the I/O or control wiring, as shown in the following diagram.



Floor Ducts
Leave at least 200 mm between the wiring and the top of the duct, as shown in the following diagram.



Conduit
Separate the CPM*A I/O lines, power and control lines, and power cables, as shown in the following diagram.

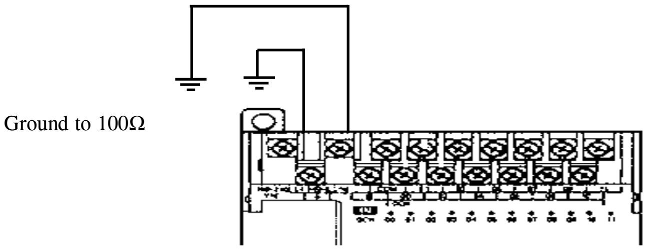


CPM*A-CPM1A and CPM2A.

8. INSTALLATION AND WIRING CONSIDERATIONS

Grounding

Be sure to ground the functional earth and protective earth terminals together to less than 100Ω in order to protect against electric shock and incorrect operation from electrical noise. Be sure to sue a wire of at least 2 mm^2 for grounding.

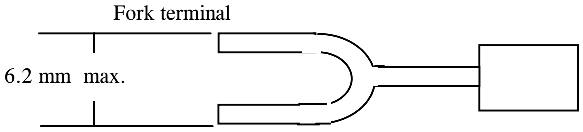


Crimp Connectors

Always use crimp connectors for the CPM*A's power lines and I/O lines or else use a single-wire line (instead of a stranded wire). Stray wire strands could accidentally short out.

Use M3 terminal screws and tighten the screws securely (0.5 to 0.6 N-m).

Recommended Terminal: Use the terminal shown below.



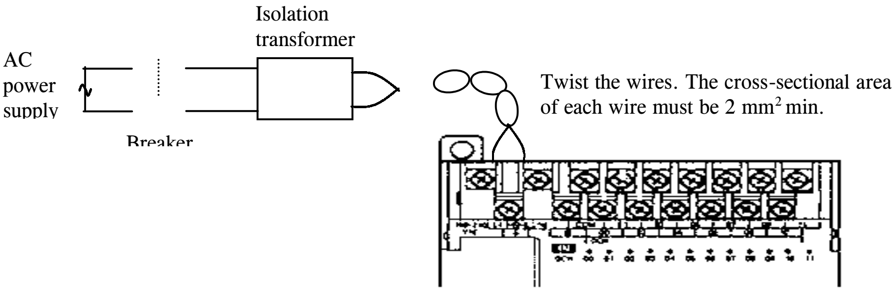
❑ Power Supply Wiring

100 to 240 VAC Power Supply

Wire a separate circuit for the CPM*A's power supply circuit so that there isn't a voltage drop from the inrush current that flows when other equipment is turned on.

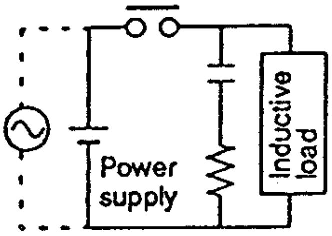
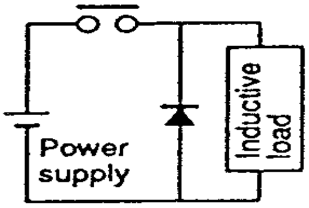
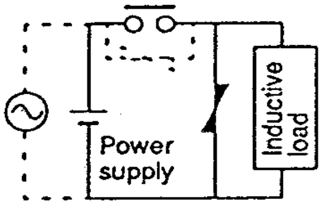
When several CPM*A PLCs are being used, it is recommended to wire the PCs on separate circuits to prevent a voltage drop from the inrush current or incorrect operation of the circuit breaker.

Use twisted power supply wires to prevent noise from the power supply lines. Adding a 1:1 isolating transformer reduces electrical noise even further.


AC power supply
Breaker
Isolation transformer
Twist the wires. The cross-sectional area of each wire must be 2 mm² min.

❑ Countermeasure for Inductive Load

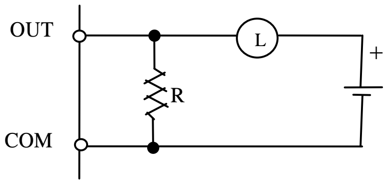
When switching an inductive load, connect an surge protector, diodes. etc. parallel with the load or contact as shown below.

Circuit	Current		Characteristic	Required element
	AC	DC		
CR method 	Yes	Yes	If the load is a relay or solenoid, there is a time lag between the moment the circuit is opened and the moment the load is reset. If the supply voltage is 24 or 48V, insert the surge protector in parallel with the load. If the supply voltage is 10 to 200V, insert the surge protector between the contacts.	The capacitance of the capacitor must be 1 to 0.5µF per contact current of A and resistance of the resistor must be 0.5 to 1Ω per contact voltage of V. These values, however, vary with the load and the characteristics of the relay. Decide these values from experiments, and take into consideration that the capacitance suppresses spark discharge when the contacts are separated and the resistance limits the current that flow into the load when the circuit is close again. The dielectric strength of the capacity must be 200 to 300V. If the circuit is an AC circuit, use a capacitor with no polarity.
Diode method 	No	Yes	The diode connected in parallel with the load changes energy accumulated by the coil into a current, which then flows into the coil so that the current will be converted into joule heat by the resistance of the inductive load. This time lag, between the moment the circuit is opened and the moment the load is reset, caused by this method is longer than that caused by the CR method.	The reversed dielectric strength value of the diode must be at least 10 times as large as the circuit voltage value. The forward current of the diode must be the same as or larger than the load current. The reversed dielectric strength value of the diode may be two to three times larger than the supply voltage if the surge protector is applied to electronic circuit with low circuit voltages.
Varistor method 	Yes	Yes	The varistor method prevents the imposition of high voltage between the contacts by using the constant voltage characteristic of the varistor. There is time lag between the moment the load is reset. If the supply voltage is 24 or 48V, insert the varistor in parallel with the load. If the supply voltage is 100 to 200V, insert the varistor between the contacts.	

8. INSTALLATION AND WIRING CONSIDERATIONS

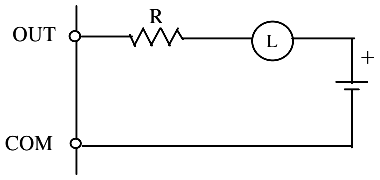
When switching a load with a high inrush current such as an incandescent lamp suppress the inrush current as shown below.

Countermeasure 1



Providing a dark current of approx. one-third of the rated value through an incandescent lamp

Countermeasure 2

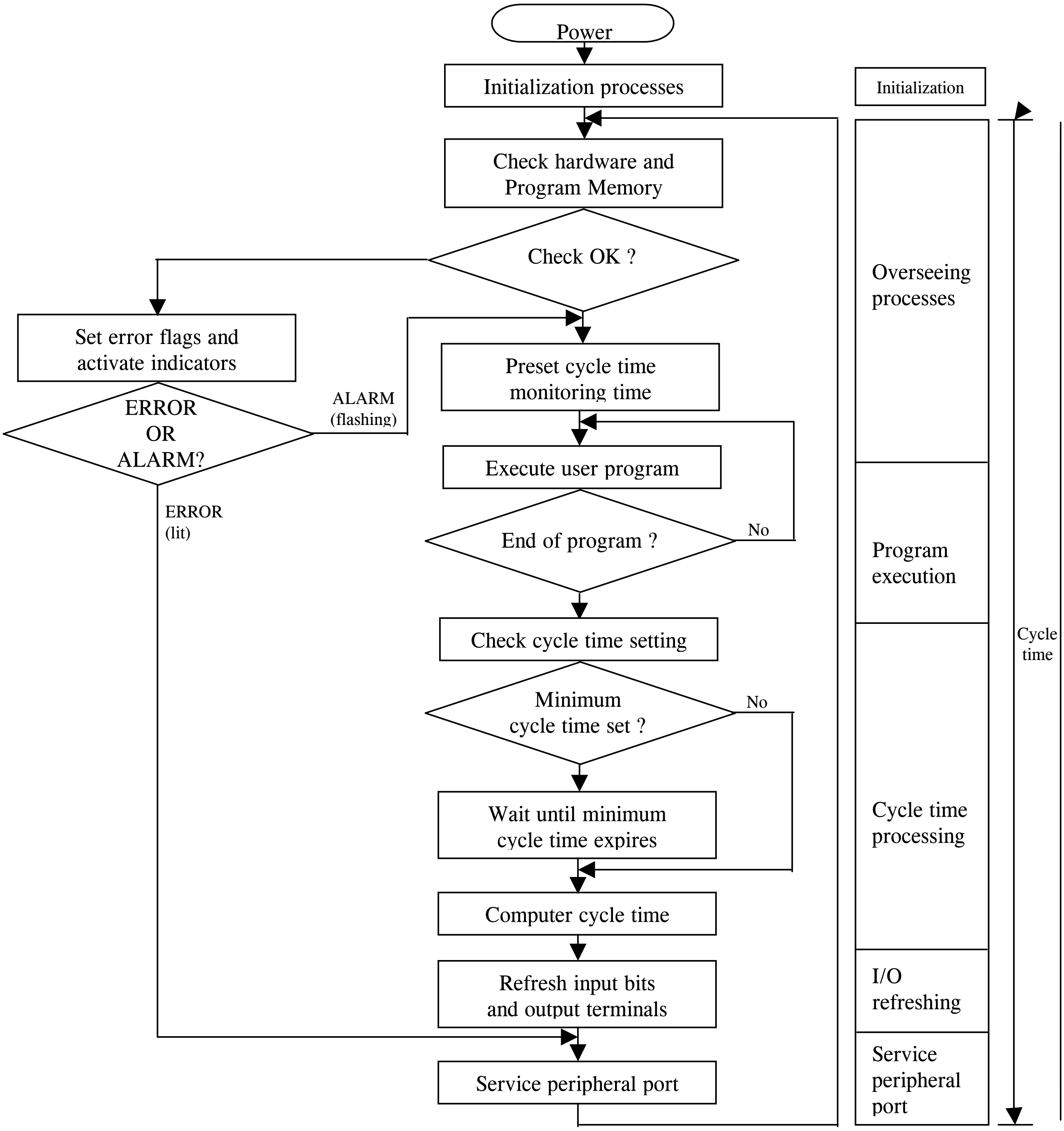


Providing a limiting resistor

9. MAINTENANCE AND TROUBLESHOOTING

❑ The PLC Cycle

The overall flow of PLC operation is as shown in the following flowchart. The PLC is initialized internally when the power is turned on. If no errors are detected, the overseeing processes, program execution, I/O refreshing, and peripheral Device servicing are executed consecutively (cyclically). The average cycle time can be monitor from a Peripheral Device.



Note Initialization process include cleaning the IR, SR, and AR areas, pre- setting system timers, and checking I/O Units.

9. MAINTENANCE AND TROUBLESHOOTING

❑ Self-diagnosis Functions

The PLC is equipped with a variety of self-diagnosis functions to help identify and correct errors that might occur an reduce down time.

PLC errors are divided into 2 categories based on the severity of the errors. Fatal errors are more serious errors which stop PLC operation. Non-fatal errors are less serious and don't stop PLC operation.

❑ Non-fatal Errors

PLC operation and program execution will continue after one or more of these errors have occurred. Although PLC operation will continue, the cause of the error should be corrected and the error cleared as soon as possible.

When one of these errors occurs, the POWER and RUN indicators will remain lit and the ERR/ALM indicator will flash.

The following messages may appear on the programming console display.

- SYS FAIL FAL**
- SCAN TIME OVER

You need to refer to the appropriate PLC manuals to check the flags to determine the cause of the problem and correct the error.

❑ Fatal Errors

PLC operation and program execution will stop and all outputs from the PLC will be turned OFF when any of these errors have occurred. CPM1A operation can't be restarted until the PLC is turned off and then on again or a Peripheral Device is used to switch the PLC to PROGRAM mode and clear the fatal error.

All CPU indicators will be OFF for the power interruption error. For all other fatal operating errors, the POWER and ERR/ALM indicators will be lit. The RUN indicator will be OFF.

The following messages may appear on the programming console display

- MEMORY ERR
- NO END INST
- I/O BUS ERR
- I/O UNIT OVER
- SYS FAIL FALS**

You need to check the error flag of the appropriate PLC manual to determine the cause of the problem and correct the error.

❑ User-defined Errors

There are three instructions that the user can use to define his own errors or messages. FAL(06) causes a non-fatal error, FAL(07) causes a fatal error, and MSG(46) sends a message to the Programming Console or host computer connected to the PC.

FAILURE ALARM-FAL(06)

FAL(06) is an instruction that causes a non-fatal error. The following will occur when an FAL(06) instruction is executed:

1,2,3... 1. The ERR/ALM indicator on the CPU will flash. PC operation will continue.
2. The instruction's 2-digit BCD FAL number (01 to 99) will be written to SR 25300 to SR 25307.

The FAL numbers can be set arbitrarily to indicate particular conditions. The same number cannot be used as both an FAL number and an FALS number.

To clear an FAL error, correct the cause of the error and then execute FAL 00 or clear the error using the Programming Console.

SEVERE FAILURE ALARM-FALS(07)

FALS(07) is an instruction that causes a fatal error. The following will occur when an FALS(07) instruction is executed:

1,2,3... 1. Program execution will be stopped and outputs will be turned OFF.
2. The ERR/ALM indicator on the CPU will be lit.
3. The instruction's 2-digit BCD FALS number (01 to 99) will be written to SR 25300 to SR 25307.
4. The FALS number and time of occurrence will be recorded in the PC's error log area if a Memory Cassette with a clock (RTC) is used.

The FALS numbers can be set arbitrarily to indicate particular conditions. The same number cannot be used as both an FAL number and an FALS number.

To clear a FALS error, switch the PC to PROGRAM Mode, correct the cause of the error, and then clear the error using the Programming Console.

MESSAGE-MSG(46)

MSG(46) is used to display a message on the Programming Console. The message, which can be up to 16 characters long, is displayed when the instruction's execution condition is ON.

❑ Programming Errors

These errors in program syntax will be detected when the program is checked using the Program Check operation.

Three levels of program checking are available. The desired level must be designated to indicate the type of errors that are to be detected. The following table provides the error types, displays, and explanations of all syntax errors. Check level 0 checks for type A, B, and C errors; check level 1, for type A and B errors; and check level 2, for type A errors only.

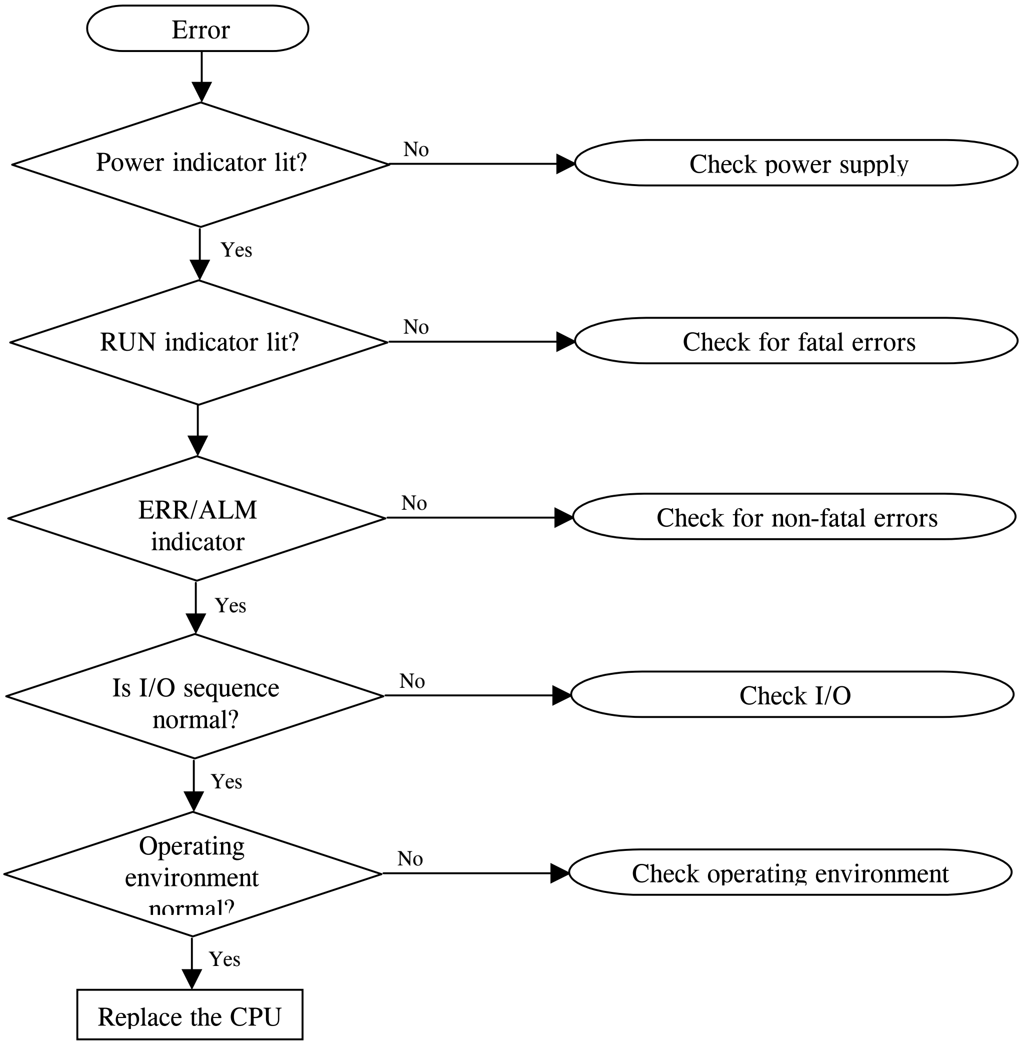
Type	Message	Meaning and appropriate response
A	?????	The program has been damaged, creating a non-existent function code. Re-enter the program.
	CIRCUIT ERR	The number of logic blocks and logic block instructions does not agree, i.e., either LD or LD NOT has been used to start a logic block whose execution condition has not been used by another instruction, or a logic block instruction has been used that does not have the required number of logic blocks. Check your program.
	OPERAND ERR	A constant entered for the instruction is not within defined values. Change the constant so that it lies within the proper range.
	NO END INSTR	There is no END(001) in the program. Write END(001) at the final address in the program.
	LOCN ERR	An instruction is in the wrong place in the program. Check instruction requirements and correct the program.
	JME UNDEFD	A JME(004) instruction is missing for a JMP(005) instruction. Correct the jump number or insert the proper JME(004) instruction.
	DUPL	The same jump number or subroutine number has been used twice. Correct the program so that the same number is only used once for each.
	SBN UNDEFD	The SBS(091) instruction has been programmed for a subroutine number that does not exist. Correct the subroutine number or program the required subroutine.
	STEP ERR	STEP(008) with a section number and STEP(008) without a section number have been used incorrectly. Check STEP(008) programming requirements and correct the program.

Type	Message	Meaning and appropriate response
B	IL-ILC-ERR	IL(002) and ILC(003) are not used in pairs. Correct the program so that each IL(002) has a unique ILC(003). Although this error message will appear if more than one IL(002) is used with the same ILC(003), the program will be executed as written. Make sure your program is written as desired before proceeding.
	JMP=JME ERR	JMP(004) and JME(005) are not used in pairs. Make sure your program is written as desired before proceeding.
	SEN-RET ERR	If the displayed address is that of SBN(092), two different subroutines have been defined with the same subroutine number. Change one of the subroutine numbers or delete one of the subroutines. If the displayed address is that of RET(093), RET(093) has not been used properly. Check requirements for RET (093) and correct the program.
C	COIL DUPL	The same bit is being controlled (i.e., turned ON and/or OFF) by more than one instruction (e.g., OUT, OUT NOT, DIFU(13), DIFD(14), KEEP(11), SFT(10)). Although this is allowed for certain instructions, check instruction requirements to confirm that the program is correct or rewrite the program so that each bit is controlled by only one instruction.
	JMP UNDEFD	JME(005) has been use with no JMP(004) with the same jump number. Add a JMP(004) with the same number or delete the JME(005) that is not being used.
	SBS UNDEFD	A subroutine exists that is not called by SBS(091). Program a subroutine call in the proper place, or delete the subroutine if it is not required.

❑ Troubleshooting Flowcharts

Use the following flowcharts to troubleshoot errors that occur during operation.

Main Check



9. MAINTENANCE AND TROUBLESHOOTING

❑ Preventive Maintenance

Although PLC is designed for reliability. It can fail eventually. Probably the biggest deterrent to system breakdown is a proper preventive maintenance program for the PLC and the control system.

Periodically, inspect the tightness of the I/O terminals screws. They can become loose over a period of time.

Ensure that components are free of dust. Proper cooling of the PLC is impossible if layer of dust are deposited.

Corrosion may take place in some environment. Check periodically for corrosion of connecting terminals. Printed circuit board and connector may become corroded internally.

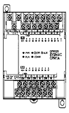
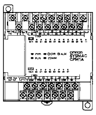
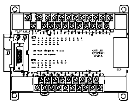
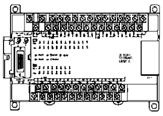
Maintain a certain amount of commonly used spare parts such as the input and output modules. It can be very costly if prolonged downtime occurs without parts.

Keep proper documentation of operating program and wiring circuit of the control system. You may need them in case of emergency.

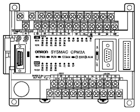
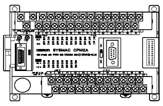
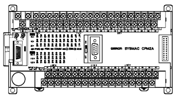
10. Appendix A- Standard Models

Standard Models

CPM1A CPUs

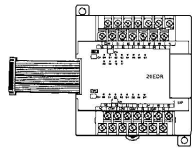
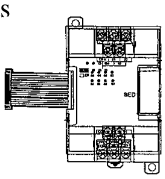
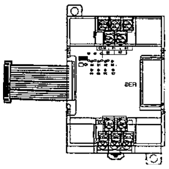
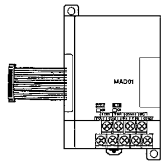
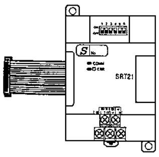
Description	Input points	Output points	Power Supply	Model Number
10 I/O points 	6 points	4 point Relay Output	100 to240 VAC, 50/60 Hz	CPM1A-10CDR-A
			24 VDC	CPM1A-10CDR-D
		Transistor NPN	24 VDC	CPM1A-l0CDT-D
		Transistor PNP	24 VDC	CPM1A-10CDT1-D
20 I/O points 	12 points	8 points	100 to 240 VAC, 50/60 Hz	CPMlA-20CDR-A
			24 VDC	CPM1A-20CDR-D
		Transistor NPN	24 VDC	CPM1A-20CDT-D
		Transistor PNP	24 VDC	CPMlA-20CDT1-D
30 I/O points 	18 points	12 points	100 to 240 VAC, 50/60 Hz	CPM1A-30CDR-A
			24 VDC	CPM1A-30CDR-D
		Transistor NPN	24 VDC	CPM1A-30CDT-D
		Transistor PNP	24 VDC	CPM1A-30CDT1-D
40 I/O points 	24 points	16 points	100 to 240 VAC, 50/60 Hz	CPM1A-40CDR-A
			24 VDC	CPM1A-40CDR-D
		Transistor NPN	24 VDC	CPM1A-40CDT-D
		Transistor PNP	24 VDC	CPM1A-40CDT1-D

CPM2A CPUs

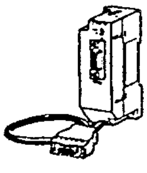
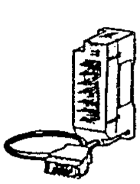
Description	Input points	Output points	Power Supply	Model Number
30 I/O points 	18 points	12 points	100 to 240 VAC, 50/60 Hz	CPM2A-30CDR-A
			24 VDC	CPM2A-30CDR-D
		Transistor NPN	24 VDC	CPM2A-30CDT-D
		Transistor PNP	24 VDC	CPM2A-30CDT1-D
40 I/O points 	24 points	16 points	100 to 240 VAC, 50/60 Hz	CPM2A-40CDR-A
			24 VDC	CPM2A-40CDR-D
		Transistor NPN	24 VDC	CPM2A-40CDT-D
		Transistor PNP	24 VDC	CPM2A-40CDT1-D
60 I/O points 	36 points	24 points	100 to 240 VAC, 50/60 Hz	CPM2A-60CDR-A
			24 VDC	CPM2A-60CDR-D
		Transistor NPN	24 VDC	CPM2A-60CDT-D
		Transistor PNP	24 VDC	CPM2A-60CDT1-D

10. Appendix A- Standard Models

Expansion I/O Unit

Description	Input points	Output points	Model Number
20 I/O points 	12 points	8 points Relay Contact	CPMlA-20EDR
		Transistor NPN	CPM1A-20EDT
		Transistor PNP	CPM1A-20EDT1
8 Input points 	8 points	0 points	CPM 1 A-8ED
8 Output points 	0 points	8 points Relay Contact	CPM1A-8ER
		Transistor NPN	CPM1A-8ET
		Transistor PNP	CPM1A-8ET1
Analog 	2 Analog	1 Analog	CPM1A-MAD01
16 points 	8 points	8 points	CPM1A-SRT2l

Communications Adapters

Description	Output points	Model Number
RS-232C Adapter 	Converts data communications between the peripheral port and RS-232C devices.	CPM1-CIF01
RS-422 Adapter 	Converts data communications between the peripheral port and RSA22 devices.	CPM1-CIF1 1

10. Appendix A- Standard Models

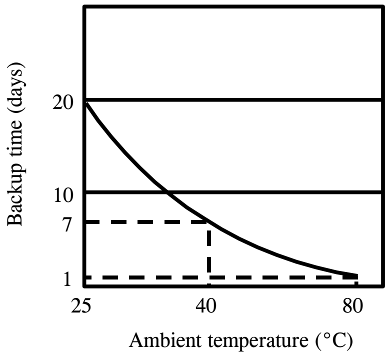
Peripheral Devices

Name	Model Number	Specifications
Programming Console	CQM1-PR001-E	2-rn Connecting Cable attached
	C200H-PR027-E	Hand-held, w/backlight; requires the C200H-CN222 or C200H-CN422 Connecting cable, see below.
SYSWIN	SYSWIN-E-V3.3	3.5” HD for IBM PC/AT compatible running Windows
Connecting Cable	CQM1-CIF02	Connects IBM PC/AT or compatible computers to Peripheral Port
	C200H-CN222	Connects C200H Programming Console to Peripheral Port (2m)
	C200H-CN422	Connects C200H Programming Console to Peripheral Port (4m)

10. Appendix B - Specifications

Backup Time vs. Temperature

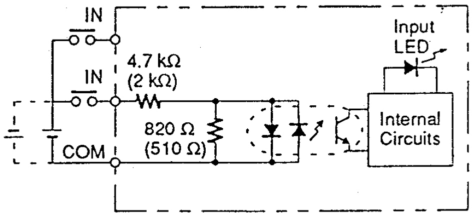
A capacitor in the CPU is used to backup the program, counter values and the contents of the DM area, HR area, and AR area. The capacitor backup time depends on the ambient temperature, as shown in the following graph.



If the contents of the CPU's program area are lost, the program stored in flash memory will be read to the CPU's program area when the CPM1A is started. Although the program will be restored, the counter values and the contents of the DM area, Hr area, and AR area will be lost.

I/O Specifications

CPU Input Specifications of CPM1A

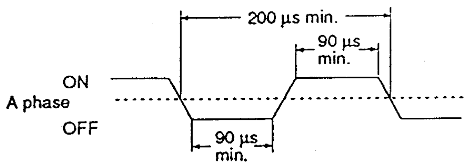
Item	Specification
Input voltage	
Input Irnpedance	IN00000 to IN00002: 2 kΩ ; other input: 4.7 kΩ
Input current	IN00000 to IN00002: 12 mA typical; other inputs: 5 mA typical
ON voltage	
OFF voltage	5.0 VDC max.
ON delay	
OFF delay	8 ms max. (see note).
Circuit configuration	 IN IN 4.7 kΩ (2 kΩ) 820 Ω (510 Ω) COM Input LED Internal Circuits Note: Figures in parentheses are for IN00000 to IN00002.

Note The input time constant for ON/OFF delays can be set to 1,2,4,8,16,32,64,128 ms. When IN00000 through IN00002 are used as high-speed counter puts, the delays are as shown in the following table.

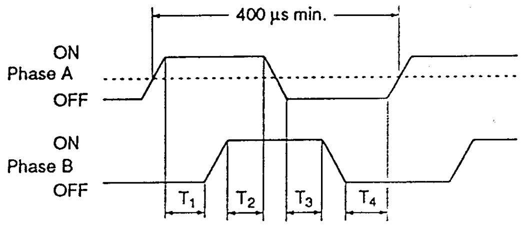
Item	Increment mode	Differential phase mode
IN00000 (A-phase)	5 KHz	2.5 KHz
IN0000l (B-phase) Normal input	Normal Input	
IN00002 (Z-phase)	ON: 100 µs max. ; OFF delay: 500 µs max.	

The minimum delay is as follows.

IN00000 (A Phase), IN00001 (B Phase)
In the increment mode (5 KHz max.)

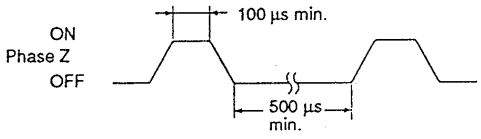


In the differential phase mode (2.5 KHz max.)



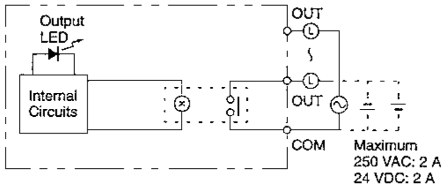
T_1 T_2 T_3 T_4 : 90 µs min.

IN00002 (Z Phase)



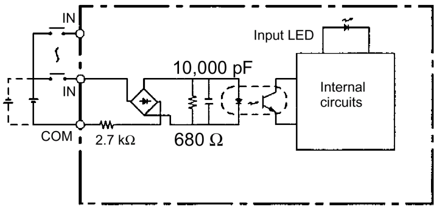
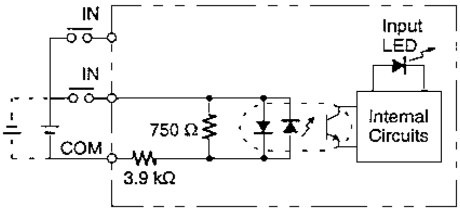
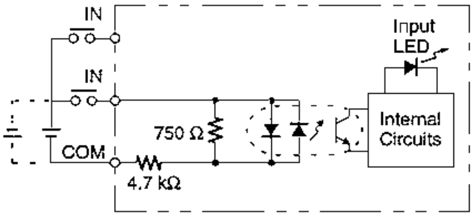
When 1N00003 through 1N00006 are used as interrupt inputs, the delay is 0.3 ms max. The delay is measured from the time that the input goes ON until the interrupt subroutine is executed.

CPU Unit Output Specifications

Item	Specification
Output type	All Outputs are relay outputs
Max. switching capacity	2 A, 250 VAC (cos ϕ = 1) 2 A, 24 VDC (4 A/common)
Min. Switching capacity	10 mA, 5 VDC
Service life of relay	Electrical: 300,000 operations (resistive load) 100,000 operations (inductive load) Mechanical: 20,000,000 operations
ON delay	15 rns max. (see note)
OFF delay	15 ms max. (see note).
Circuit configuration	 Output LED Internal Circuits OUT OUT COM Maximum 250 VAC: 2 A 24 VDC: 2 A

10. Appendix B - Specifications

CPU Unit Input Specification for CPM2A

Item	Input	Specification
Input voltage	All	24 VDC $^{+10\%}/_{-15\%}$
Input impedance	IN00000 to IN00001	1.8 kΩ
	IN00002 to IN00006	3.9 kΩ
	IN00007 to up	4.7 kΩ
Input current	IN00000 to 1N00001	9 mA typical
	1N00002 to 1N00006	6 mA typical
	1N00007 to up	5 mA typical
ON voltage/current	IN00000 to IN00001	17 VDC mm., 5 mA
	IN00002 to up	14.4 VDC min., 3 mA
OFF voltage/current	All	5.0 VDC max., 1 mA
ON delay	All	1 to 80 ms max. Default: 10 ms (See note.)
OFF delay	All	1 to 80 ms max. Default: 10 ms (See note.)
Circuit configuration	IN00000 to IN0000l	 IN IN COM Input LED 10,000 pF Internal circuits 2.7 kΩ 680 Ω
	IN00002 to IN00006	 IN IN COM Input LED 750 Ω 3.9 kΩ Internal Circuits
	IN00007 to up	 IN IN COM Input LED 750 Ω 4.7 kΩ Internal Circuits

Note The input time constant can be set to 1,2,3,5,10,20,40 ot 80 ms in the PLC Setup.

High-speed Counter Inputs

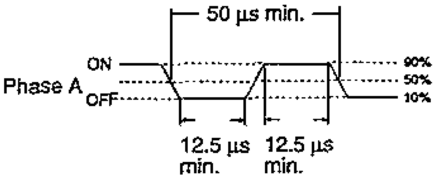
Inputs IN00000 through IN00002 can be used as high-speed counter inputs, as shown in the following table. The maximum count frequency is 5 kHz in differential phase mode and 20 kHz in the other modes.

10. Appendix B - Specifications

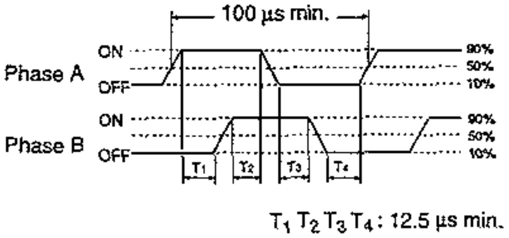
Input	Function			
	Differential phase mode	Pulse + direction input mode	Up/down Input mode	Increment mode input mode
IN00000	A-phase pulse input	Pulse input	Increment pulse input	Increment pulse input
IN00001	B-phase pulse input	Direction input	Decrement pulse input	Normal input
IN00002	Z-phase pulse input or hardware reset input (1N00002 can be used as a normal input when it is not used as a high-speed counter input.)			

The minimum pulse widths for the inputs IN00000 (A-phase input) and IN00001 (B-phase input) are as follows:

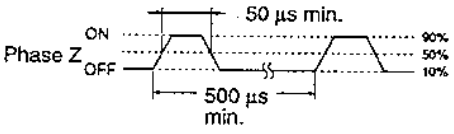
Pulse-Direction Input mode
Up/Down Input mode



Differential Phase mode,
Up/Down Input mode



The minimum pulse width for input IN00002 (Z-phase input) is as follows:

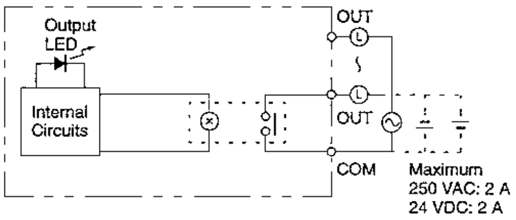


Interrupt Inputs

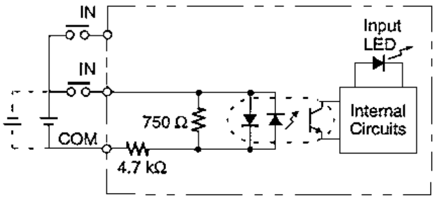
Inputs 1N00003 through IN00006 can be used as interrupt inputs (interrupt input mode or counter mode) and quick-response inputs. The minimum pulse width for these inputs is 0.05 ms.

10. Appendix B - Specifications

CPM2A CPU unit and Expansion I/O Unit Input Specifications (Relay Output)

Item	Specification
Max. switching-capacity	2 A, 250 VAC (cos ϕ = 1) 2 A, 24 VDC
Min. Switching capacity 10 mA	5 VDC
Service life of relay	Electrical: 150,000 operations (30-VDC resistive load) 100,000 operations (240-VAC inductive load, cos ϕ = 4) Mechanical: 20,000,000 operations
ON delay	15 ms max.
OFF delay	15 ms max.
Circuit configuration	 Output LED Internal Circuits OUT OUT COM Maximum 250 VAC: 2 A 24 VDC: 2 A

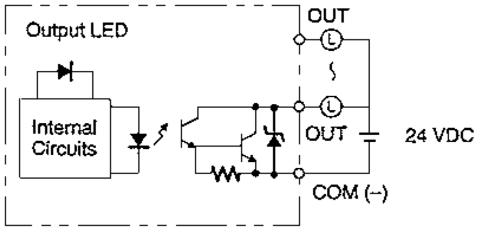
Expansion I/O Unit Input Specifications

Item	Specification
Input voltage	24 VDC
Input impedance	4.7 kΩ
Input current	5 mA typical
ON voltage	14.4 VDC min.
OFF voltage	5.0 VDC max.
ON delay	1 to 80 ms max. Default: 10 ms (See note.)
OFF delay	1 to 80 ms max. Default: 10 ms (See note.)
Circuit configuration	 IN IN COM 750 Ω 4.7 kΩ Input LED Internal Circuits

Note The input time constant can be set to 1,2,3,4,5,10,20, 40 or 80 ms in the PLC Setup.

10. Appendix B - Specifications

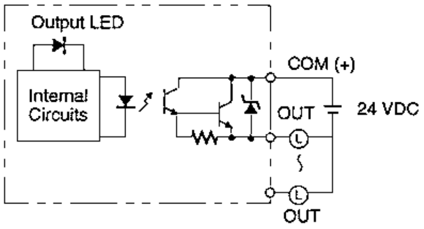
Transistor Output (Sinking)

Item	Specification				
	30 CDT-D	40 CDT-D	60 CDT-D	20 EDT	8ET
Max. Switching Capacity	4.5 to 30 VDC, 0.3 A/Output (See Note)				
	0.9A/Common 2.7 A/Unit	0.9A/Common 3.6 A/Unit	0.9A/Common 5.4A /Unit	0.9A/Common 1.8 A/Unit	0.9A/Common 1.8 A/Unit
Leakage Current	0.1 mA max.				
Residual Voltage	1.5 V max.				
On Delay	OUT01000 and OUT01001 : 20 µs max. OUT01002 and up: 0.1 ms max.				
OFF Delay	OUT01000 and OUT01001 : 40 µs max. (4.5 to 26.5 V, 10 to 100 mA) 0.1 ms max. (4.5 to 30 V, 10 to 300 mA) OUT01002 and up: 1 ms max.				
Fuse	1 fuse/output (cannot be replaced by the user)				
Circuit configuration	 OUT Output LED Internal Circuits OUT 24 VDC COM (–)				

Note When using OUT01000 or OUT01001 as a pulse output, connect a dummy resistor as required to bring the load current between 0.01 and 0.1A. If the load current is below 0.1A, the ON-to-OFF response time will be shorter and high-speed pulses (source-type transistor outputs) will not be output. On the other hand, if the load current is above 0.1A, the transistor will generate more heat and components may be damaged.

10. Appendix B - Specifications

Transistor Output (Sourcing)

Item	Specification				
	30 CDT1-D	40 CDT1-D	60 CDT1-D	20 EDT1	8ET1
Max. Switching Capacity	4.5 to 30 VDC, 0.3 A/Output (See Note)				
	0.9A/Common 2.7 A/Unit	0.9A/Common 3.6 A/Unit	0.9A/Common 5.4A /Unit	0.9A/Common 1.8 A/Unit	0.9A/Common 1.8 A/Unit
Leakage Current	0.1 mA max.				
Residual Voltage	1.5 V max.				
On Delay	OUT01000 and OUT01001 : 20 µs max. OUT01002 and up: 0.1 ms max.				
OFF Delay	OUT01000 and OUT01001 : 40 µs max. (4.5 to 26.5 V, 10 to 100 mA) 0.1 ms max. (4.5 to 30 V, 10 to 300 mA) OUT01002 and up: 1 ms max.				
Fuse	1 fuse/output (cannot be replaced by the user)				
Circuit configuration					

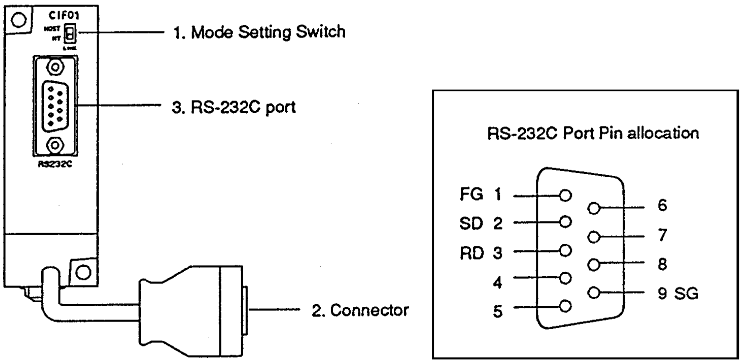
Note When using OUT01000 or OUT01001 as a pulse output, connect a dummy resistor as required to bring the load current between 0.01 and 0.1A. If the load current is below 0.1A, the ON-to-OFF response time will be shorter and high-speed pulses (source-type transistor outputs) will not be output. On the other hand, if the load current is above 0.1A, the transistor will generate more heat and components may be damaged.

10. Appendix B - Specifications

RS-232C Adapter Specifications

Item	Specification
Function	Converts between the CMOS format (PC CPU side) and the RS 232C format (personal computer side).
Insulation	The RS 232C (personal computer side) is isolated by a DC/DC converter and photocoupler.
Power supply	Power is supplied from the PC CPU
Power consumption	0.3 A max.
Baud rate	38,400 bps max.
Vibration resistance	10 to 57 Hz: 0.075-mm amplitude 57 to 150 Hz: 9.8 m/s^2 (1G) acceleration in X,Y, and Z directions for 80 minutes each (Time coefficient; 8 minutes x coefficient factor 10 =total time 80 minutes)
Shock resistance	147 m/s^2 (15G) three times each in X,Y, and Z directions
Ambient temperature	Operating: 0°C to 55°C Storage: -20°C to 75 °C
Humidity	10% to 90% (with no condensation)
Atmosphere	Must be free from corrosive gas
Weight	200 g max.

RS-232 Adapter



1,2,3....

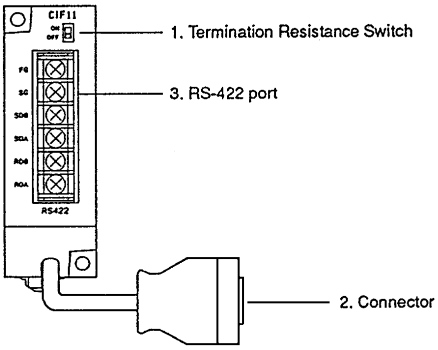
1. Mode Setting Switch
   Set this switch to "HOST" when using a Host Link System to connect to a personal computer. Set this switch to "NT" when connecting to Programmable Terminal or PC for 1:1 link.
2. Connector
   Connects to the CPU's Peripheral Port.
3. RS-232C Port
   Connects to the RS-232C cable from the other device such as a personal computer, Peripheral Device, or Programmable Terminal.

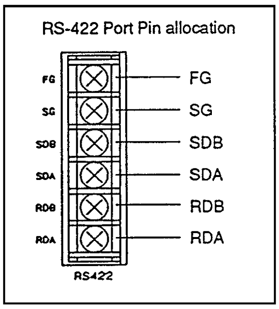
10. Appendix B - Specifications

RS-422C Adapter Specifications

Item	Specification
Function	Converts between the CMOS format (PC CPU side) and the RS 422C format (Peripheral Device side).
Insulation	The RS 232C (personal computer side) is insulated by a DC/DC converter and photocoupler.
Power supply	Power is supplied from the CPU
Power consumption	0.3 A max.
Baud rate	38,400 bps max.
Vibration resistance	10 to 57 Hz: 0.075-mm amplitude 57 to 150 Hz: 9.8 m/s^2 (1G) acceleration in X,Y, and Z directions for 80 minutes each (Time coefficient; 8 minutes x coefficient factor 10 =total time 80 minutes)
Shock resistance	147 m/s^2 (15G) three times each in X,Y, and Z directions
Ambient temperature	Operating: 0°C to 55°C Storage: -20°C to 75 °C
Humidity	10% to 90% (with no condensation)
Atmosphere	Must be free from corrosive gas
Weight	200 g max.

RS-422 Adapter





1,2,3....

1. Termination Resistance Switch
   Set the termination resistance switch to "ON" (upper side) for the link Adapters on both ends of the Host Link system and for the RS-422 Adapter.
2. Connector
   Connects to the CPU's Peripheral Port.

3. RS-422 Port
   Connects to the Host Link network.

10. Appendix C- Programming Instructions

Instruction available in CPM1A and CPM2A

• Sequence Instructions

Sequence Input Instructions

Instruction	Mnemonic	Code	Function
LOAD	LD	0	Connects an NO condition to the left bus bar.
LOAD NOT	LD NOT	0	Connects an NC condition to the left bus bar.
AND	AND	0	Connects an NO condition in series with the previous condition
AND NOT	AND NOT	0	Connects an NC condition in series with the previous condition
OR	OR	0	Connects an NO condition in parallel with the previous condition.
OR NOT	OR NOT	0	Connects an NC condition in parallel with the previous condition.
AND LOAD	AND LD	0	Connects two instruction blocks in series.
OR LOAD	OR LD	0	Connects two instruction blocks in parallel.

Note: 0: Instruction keys allocated to the Programming Console.

Sequence Output Instructions

Instruction	Mnemonic	Code	Function
OUTPUT	OUT	0	Outputs the result of logic to a bit.
OUT NOT	OUT NOT	0	Reverses and outputs the result of logic to a bit.
SET	SET	0	Force sets (ON) a bit.
RESET	RESET	0	Force resets (OFF) a bit.
KEEP	KEEP	11	Maintains the status of the designated bit.
DIFFERENTIATE UP	DIFU	13	Turns ON a bit for one cycle when the execution condition goes from OFF to ON.
DIFFERENTIATE DOWN	DIFD	14	Turns ON a bit for one cycle when the execution condition goes from ON to OFF.

Note: 0:Instruction keys allocated to the Programming Console.

Sequence Control Instructions

Instruction	Mnemonic	Code	Function
NO OPERATION	NOP	00	---
END	END	01	Required at the end of the program.
INTERLOCK	IL	02	It the execution condition for IL(02) is OFF, all outputs are turned OFF and all timer PVs reset between IL(02) and the next ILC(03).
INTERLOCK CLEAR	ILC	03	ILC(03) indicates the end of an interlock (beginning at IL(02)).
JUMP	JMP	04	If the execution condition for JMP(04) is ON, all instructions between JMP(04) and JME(05) are treated as NOP(OO).
JUMP END	JME	05	JME(05) indicates the end of a jump (beginning at JMP(04)).

10. Appendix C- Programming Instructions

Timer/Counter Instructions

Instruction	Mnemonic	Code	Function
TIMER	TIM	0	An ON-delay (decrementing) timer.
COUNTER	CNT	0	A decrementing counter.
REVERSIBLE COUNTER	CNTR	12	Increases or decreases PV by one.
HIGH-SPEED TIMER	TIMH	15	A high-speed, ON-delay (decrementing) timer.

Note: 0: Instruction keys allocated to the Programming Console.

Data Comparison Instructions

Instruction	Mnemonic	Code	Function
COMPARE	CMP	20	Compares two four-digit hexadecimal values.
DOUBLE COMPARE	CMPL	60	Compares two eight-digit hexadecimal values.
BLOCK COMPARE	(@)BCMP	68	Judges whether the value of a word is within 16 ranges (defined by lower and upper limits).
TABLE COMPARE	(@)TCMP	85	Compares the value of a word to 16 consecutive words.

Data Movement Instructions

Instruction	Mnemonic	Code	Function
MOVE	(@)MOV	21	Copies a constant or the content of a word to a word.
MOVE NOT	(@)MVN	22	Copies the complement of a constant or the content of a word to a word.
BLOCK TRANSFER	(@)XFER	70	Copies the content of a block of up to 1,000 consecutive words to a block of consecutive words.
BLOCK SET	(@)BSET	71	Copies the content of a word to a block of consecutive words.
DATA EXCHAGE	(@)XCHG	73	Exchanges the content of two words.
SINGLE WORD DISTRIBUTE	(@)DIST	80	Copies the content of a word to a word (whose address is determined by adding an offset to a word address).
DATA COLLECT	(@)COLL	81	Copies the content of a word (whose address is determined by adding an offset to a word address) to a word.
MOVE BIT	(@)MOVB	82	Copies the specified bit from one word to the specified bit of a word.
MOVE DIGIT	(@)MOVD	83	Copies the specified digits (4-bit units) from a word to the specified digits of a word.

10. Appendix C- Programming Instructions

Shift Instructions

Instruction	Mnemonic	Code	Function
SHIFT REGISTER	SFT	0/10	Copies the specified bit (0 or 1) into the rightmost bit of a shift register and shifts the other bits one bit to the left.
WORD SHIFT	(@)WSFT	16	Creates a multiple-word shift register that shifts data to the left in one-word units.
ASYNCHRONOUS SHIFT REGISTER	(@)ASFT	17	Creates a shift register that exchanges the contents of adjacent words when one is zero and the other is not.
ARITHMETIC SHIFT LEFT	(@)ASL	25	Shifts a 0 into bit 00 of the specified word and shifts the other bits one bit to the left.
ARITHMETIC SHIFT RIGHT	(@)ASR	26	Shifts a 0 into bit 15 of the specified word and shifts the other bits one bit to the right.
ROTATE LEFT	(@)ROL	27	Moves the content of CY into bit 00 of the specified word, shifts the other bits one bit to the left, and moves bit 15 to CY.
ROTATE RIGHT	(@)ROR	28	Moves the content of CY into bit 15 of the specified word, shifts the other bits one bit to the left, and moves bit 00 to CY.
ONE DIGIT SHIFT LEFT	(@)SLD	74	Shifts a 0 into the rightmost digit (4-bit unit) of the shift register and shifts the other digits one digit to the left.
ONE DIGIT SHIFT RIGHT	(@)SRD	75	Shifts a 0 into the rightmost digit (4-bit unit) of the shift register and shifts the other digits one digit to the right.
REVERSIBLE SHIFT REGISTER	(@)SFTR	84	Creates a single or multiple-word shift register that can shift data to the left or right.

Note: 0: Instruction keys allocated to the Programming Console.

Increment/Decrement Instructions

Instruction	Mnemonic	Code	Function
INCREMENT	(@)INC	38	Increments the BCD content of the specified word by 1.
DECREMENT	(@)DEC	39	Decrements the BCD content of the specified word by 1.

BCD/Binary Calculation Instructions

Instruction	Mnemonic	Code	Function
BCD ADD	(@)ADD	30	Adds the content of a word (or a constant).
BCD SUBTRACT	(@)SUB	31	Subtracts the contents of a word (or constant) and CY from the content of a word (or constant).
BDC MULTIPLY	(@)MUL	32	Multiplies the content of two words (or contents).
BCD DIVIDE	(@)DIV	33	Divides the contents of a word (or constant) by the content of a word (or constant).
BINARY ADD	(@)ADB	50	Adds the contents of two words (or constants) and CY.
BINARY SUBTRACT	(@)SBB	51	Subtracts the content of a word (or constant) an CY from the content of the word (or constant).
BINARY MULTIPLY	(©)MLB	52	Multiplies the contents of two words (or constants).
BINARY DIVIDE	(@)DVB	53	Divides the content of a word (or constant) by the content of a word and obtains the result and remainder.
DOUBLE BCD ADD	(@)ADDL	54	Add the 8-digit BCD contents of two pairs of words (or constants) and CY.

(Con't)

Instruction	Mnemonic	Code	Function
DOUBLE BCD SUBTRACT	(@)SUBL	55	Subtracts the 8-digit BCD contents of a pair of words (or constants) and CY from the 80digit BCD contents of a pair of words (or constants).
DOUBLE BCD MULITPLY	(@)MULL	56	Multiplies the 8-digit BCD contents of two pairs of words (or constants).
DOUBLE BCD DIVIDE	(@)DIVL	57	Divides the 8-digit BCD contents of a pair of words (or constants) by the 8 –digits BCD contents of a pair of words (or constants)

Data Conversion Instructions

Instruction	Mnemonic	Code	Function
BCD TO BINARY	(@)BIN	23	Converts 4-digit BCD data to 4-digit binary data.
BINARY TO BCD	(@)BCD	24	Converts 4-digit binary data to 4 digit BCD data.
4 to 16 DECODER	(@)MLPX	76	Takes the hexadecimal value of the specified digit(s) in a word and turn ON the corresponding bit in a word(s).
16 to 4 DECODER	(@)DPMX	77	Identifies the highest ON bit in the specified word(s) and moves the hexadecimal value(s) corresponding to its location to the specified digit(s) in a word.
ASCII CODE CONVERT	(@)ASC	86	Converts the designated digit(s) of a word into the equivalent 8-bit ASCII code.

Logic Instructions

Instruction	Mnemonic	Code	Function
COMPLEMENT	(@)COM	29	Turns OFF all ON bits and turns ON all OFF bits in the specified word
LOGICAL AND	(@)ANDW	34	Logically ANDs the corresponding bits of two word (or constants)
LOGICAL OR	(@)ORW	35	Logically ORs the corresponding bits of two word (or constants)
EXCLUSIVE OR	(@)XORW	36	Exclusively ORs the corresponding bits of two words (or constants)
EXCLUSIVE NOR	(@)XNRW	37	Exclusively NORs the corresponding bits of two words (or constants).

Special Calculation Instructions

Instruction	Mnemonic	Code	Function
BIT COUNTER	(@)BCNT	67	Counts the total number of bits that are ON in the specified block

10. Appendix C- Programming Instructions

Subroutine Instructions

Instruction	Mnemonic	Code	Function
SUBROUTINE ENTER	(@)SBS	91	Executes a subroutine in the main program.
SUBROUTINE ENTRY	SBN	92	Marks the beginning of a subroutine program.
SUBROUTINE RETURN	RET	93	Marks the end of a subroutine program.
MACRO	MACRO	99	Calls and executes the specified subroutine, substituting the specified input and output words for the input and output words in the subroutine.

Interrupt Control Instructions

Instruction	Mnemonic	Code	Function
INTERVAL TIMER	(@)STIM	69	Controls interval timers used to perform scheduled interrupts.
INTERRUPT CONTROL	(@)INT	89	Performs interrupts control, such as masking and unmasking the interrupt bits for I/O interrupts.

Step Instructions

Instruction	Mnemonic	Code	Function
STEP DEFINE	STEP	08	Defines the start of a new step and resets the previous step when used with a control bit. Defines the end of step execution when used without a control bit.
STEP START	SNXT	09	Starts the execution of the step when used with a control bit.

Peripheral Device Control Instructions

Instruction	Mnemonic	Code	Function
BCD TO BINARY	(@)BIN	23	Converts 4-digit BCD data to 4-digit binary data.
BINARY TO BCD	(@)BCD	24	Converts 4-digit binary data to 4-digit BCD data.
4 to 16 DECODER	(@)MLPX	76	Takes the hexadecimal value of the specified digit(s) in a word and turn ON the corresponding bit in a word(s).
16 to 4 DECODER	(@)DPMX	77	Identifies the highest ON bit in the specified word(s) and moves the hexadecimal value(s) corresponding to its location to the specified digit(s) in a word.
ASCII CODE CONVERT	(@)ASC	86	Converts the designated digit(s) of a word into the equivalent 8-bit ASCII code.

I/O Units Instructions

Instruction	Mnemonic	Code	Function
7-SEGMENT DECODER	(@)SDEC	78	Converts the designated digit(s)of a word into an 8-bit, 7-segment display code.
I/O REFRESH	(@)IORF	97	Refreshes the specified I/O word.

Display Instructions

Instruction	Mnemonic	Code	Function
MEASSAGE	(@)MSG	46	Reads up to 8 words of ASCII code (16 characters) from memory and displays the message on the Programming Console or other Peripheral Device.

High Speed Counter Control Instructions

Instruction	Mnemonic	Code	Function
MODE CONTROL	(@)INI	61	Starts and stops counter operation, compares and changes counter PVs, and stops pulse output.
PV READ	(@)PRV	62	Reads counter PVs and status data.
COMPARE TABLE LOAD	(@)CTBL	63	Compares counter PVs and generates a direct table or starts operation.

Damage Diagnosis Instructions

Instruction	Mnemonic	Code	Function
FAILURE ALARM	(@)FAL	06	Generates a non-fatal error when executed. The Error/Alarm indicator flashes and the CPU continues operating.
SEVERE FAILURE ALARM	FAL	07	Generates a fatal error when executed. The Error/Alarm indicator lights and the CPU stops operating.

Special System Instructions

Instruction	Mnemonic	Code	Function
SET CARRY	(@)STC	40	Sets Carry Flag 25504 to 1.
CLEAR CARRY	(@)CLC	41	Sets Carry Flag 25504 to 0.

Instruction available in CPM2A only

Data Conversion

Instruction	Mnemonic	Code	Function
DOUBLE BCD TO DOUBLE BINARY	(@)BINL	58	Converts BCD value in two consecutive source words into binary and outputs converted data to two consecutive result words
DOUBLE BINARY TO DOUBLE BCD	(@)BCDL	59	Converts binary values in two consecutive source words into BCD and outputs converted data two consecutive result words.

.

Expansion Instruction

Serial Communication Instruction

Instruction	Mnemonic	Code	Function
RECEIVE	(@)RXD	47	Receives data via a communication port.
TRANSMIT	(@)TXD	48	Sends data via a communication port.

Scaling Conversion Instruction

Instruction	Mnemonic	Code	Function
SCALE	(@)SCL	66	Performs a scaling conversion on the calculated value.
SIGNED BINARY TO BCD	(@)SCL2	--	Linearly converts a 4-digit signed hexadecimal value to a 4 digit BCD value.
BCD TO SIGNED BINARY	(@)SCL3	--	Linearly converts a 4-digit BCD value to a 4 digit signed hexadecimal value
ASCII-TO-HEXADECIMAL	(@)HEX	--	Converts ACSII data to hexadecimal data.
2'S COMPLEMENT	(@)NEG	--	Converts the four digit hexadecimal content of the source word to its 2's complement and outputs the result to R.

Data Search/Finds Instruction

Instruction	Mnemonic	Code	Function
DATA SEARCH	(@)SRCH	--	Searches the specified range of memory for the specified data. Outputs the word address(es) of words in the range that contain the data.
FIND MAXIMUM	(@)MAX	--	Finds the maximum value in the specified data area and outputs that value to another word.
FIND MINIMUM	(@)MIN	--	Finds the minimum value in the specified data area and outputs that value to another word.

Calculation Instruction

Instruction	Mnemonic	Code	Function
SUM CALCULATE	(@)SUM	--	Computes the sum if the contents of the words in the specified range of memory.
FCS CALCUTLATE	(@)FCS	--	Check for errors in data transmitted by a Host Link command.
AVERAGE CALCULATE	AVG	--	Adds the specified number if hexadecimal words and computes the mean value. Rounds off to 4 digits past decimal point.

Instruction available in CPM2A only

Expansion Instruction (continue)

Data Compare Instruction

Instruction	Mnemonic	Code	Function
AREA RANGE COMPARE	ZCP	--	Compare a word to a range defined by lower and upper limits and outputs the result to the GR,EQ and LE flags.
DOUBLE AREA RANGE COMPARE	ZCPL	--	Compares an 8-digit value to a range defined by lower and upper limits and outputs the result to the GR, EQ and LE flags.

Control Instruction

Instruction	Mnemonic	Code	Function
PULSE WIDTH VARIABLE DUTY RATIO	(@)PWM	--	Outputs pulses with the specified duty ratio (0% to 99%)
PID CONTROL	PID	--	Performs PID control based on the specified parameters.
ACCELERATION CONTROL	(@)ACC	--	Together with PULS(-), ACC(-) controls the acceleration and/or deceleration of pulses output.

Special Instruction on Timing Control

Instruction	Mnemonic	Code	Function
VERY HIGH SPEED TIMER	TMHH	--	A high speed, decrementing ON-delay timer that times in 1-ms units
LONG TIMER	TIML	--	A decrementing ON-Delay timer with SV of up to 99,990.
START SYNHRONIZED CONTROL	(@)SYNC	--	Multiples an input pulse frequency by fixed scaling factor and outputs pulses from the specified output at the resulting frequency.

10. Appendix D- Memory Areas

• CPM1A Memory Area Functions

Memory Area Structure

The following memory areas can be used with the CPM1A.

Data area		Words	Bits	Function
IR area	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	These bits can be allocated to the external I/O terminals
	Output area	IR010 to IR0l9 (10 words)	IR 0l000toIR 0l9l5 (160 bits)	
	Work area	IR 200 to IR 231 (32 words)	IR 20000 to IR 23115 (512 bits)	Work bits can be freely used within the program
SR area		SR 232 to SR 255 (24 words)	SR 23200 to SR 25515 (384 bits)	These bits serve specific functions such as flags and control bits.
TR area		--	TR 0 to TR 7 (8 bits)	Thes bits are used to temporarily store ON/OFF status at program branches
HR area2		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	These bits store data and retain their ON/OFF status when power is turned off.
AR area		AR00 to AR 15 (16 words)	AR 0000 to AR 1515 (256 bits)	These bits serve specific functions such as flags an control bits.
LR area1		LR 00 to LR 15 (16 words)	LR 00000 to LR 1515 (256 bits)	Used for a 1:1 data link with another PLC.
Timer /Counter Area		TC 000 to TC 127 (timer/counter numbers)3		The same numbers are used for both timers and counters.
DM area	Read/write 2	DM 0000 to DM0999 (1,002 words)	---	DM area data can be accessed in words units only. Word values are retained when the power is turned off.
	Error Log 4	DM1000 to DM 1021 (22 words)	---	Used to store the time of occurrence and error code of errors that occurred. These words can be used as ordinary read/write DM when the error log function isn't being used.
	Read-only 4	DM6144 to DM 6599	---	Cannot be overwritten from program
	PC Setup4	DM 6600 to DM 6655 (56 words)	---	Used to store various parameters that control PLC operation

Note: 1. IR and LR bits that are not used for their allocated functions can be used as work bits.

2. The contents of the HR area, LR area, Counter area, and read/write DM area are backed up by a capacitor. At 25 *C, the capacitor will back up memory for 20 days.
3. When accessing a PV, TC numbers are used as word data; when accessing Completing Flags, they are used as bit data.
4. Data in DM6144 to DM6655 cannot be overwritten from the program, but they can be changed from a Peripheral Device.

10. Appendix D- Memory Areas

SR Area

These bits mainly serve as flags related to CPM2A operation or contain present and set values for various functions. The functions of the SR area are explained in the following

Word(s)	Bit(s)	Function
SR 232 to SR 235	00 to 15	Macro Function Input Area Contains the input operands for MCRO(99). (Can be used as work bits when MCRO(99) is not used.)
SR 236 to SR 239	00 to 15	Macro Function Output Area Contains the output operands for MCRO(99). (Can be used as work bits when MCRO(99) is not used.)
SR240	00 to 15	Input Interrupt 0 Counter Mode SV SV when input interrupt 0 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 0 is not used in counter mode.)
SR241	00 to 15	Input Interrupt 1 Counter Mode SV SV when input interrupt 1 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 1 is not used in counter mode.)
SR242	00 to 15	Input Interrupt 2 Counter Mode SV SV when input interrupt 2 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 2 is not used in counter mode.)
SR 243	00 to 15	Input Interrupt 3 counter Mode SV SV when input interrupt 3 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 3 is not used in counter mode.)
SR 244	00 to 15	Input Interrupt 0 Counter Mode PV Counter PV- 1 when input interrupt 0 is used in counter mode (4 digits hexadecimal).
SR 245	00 to 15	Input Interrupt 1 Counter Mode PV Counter PV-1 when input interrupt 1 is used in counter mode (4 digits hexadecimal).
SR 246	00 to 15	Input Interrupt 2 Counter Mode PV Counter PV-1 when input interrupt 2 is used in counter mode (4 digits hexadecimal).
SR 247	00 to 15	Input Interrupt 3 Counter Mode PV Counter PV-1 when input interrupt 3 is used in counter mode (4 digits hexadecimal).
SR 248, SR 249	00 to 15	High-speed Counter PV Area (Can be used as work bits when the high-speed counter is not used.)
SR 250	00 to 15	Analog Volume Setting 0 Used to store the 4-digit BCD set value (0000 to 0200) from analog volume control 0.
SR 251	00 to 15	Analog Volume Setting 1 Used to store the 4-digit BCD set value (0000 to 0200) from analog volume control 1.
SR252	00	High Speed Counter Reset Bit
	01 to 07	Not Used.
	08	Peripheral Port Turn On to reset peripheral port.(Not valid when peripheral device is connected.) Automatically turns OFF when reset is complete.
	09	Not Used.
	10	PLC Setup Reset Bit Turn ON to initialize PC Setup (DM 6600 through DM 6655). Automatically turns OFF again when reset is complete. Only effective if the PC is in PROGRAM mode.
	11	Forced Status Hold Bit OFF : The forced status of bit that are forced set/reset are cleared when switching between PROGRAM mode and MONITOR mode. ON : The status of bits that are forced set/reset are maintained when switching between PROGRAM mode and MONITOR mode.

10. Appendix D- Memory Areas

SR area

Word(s)	Bit(s)	Function
SR 252	12	I/O Hold Bit OFF : IR and LR bits are reset when starting or stopping operation. ON : IR and LR bit status is maintained when starting or stopping operation
	13	Not used.
	14	Error Log Reset Bit Turn ON to clear error log. Automatically turns OFF again when operation is complete.
	15	Not Used
SR 253	00 to 07	FAL Error Code The error code (a 2-digit number) is stored here when an error occurs. The FAL number is stored here when FAL(06) or FALS(07) is executed. This word is reset (to 00) by executing a FAL 00 instruction or by clearing the error from a Peripheral Device.
	08	No Used
	09	Cycle Time Overrun Flag Turns ON when a cycle time overrun occurs (i.e., when the cycle time exceeds 100 ms).
	10 to 12	Not used.
	13	Always ON Flag
	14	Always OFF Flag
	15	First Cycle Flag Turns ON for 1 cycle at the start of operation.
SR 254	00	1-minute clock pulse (30 seconds ON; 30 seconds OFF)
	01	0.02-second clock pulse (0.01 second ON; 0.01 second OFF)
	02	Negative (N) Flag
	03 to 05	Not used.
	06	Differential Monitor Complete Flag
	07	STEP (08) Execution Flag Turns ON for 1 cycle only at the start of process based on STEP(08).
	08 to 15	Not used.
SR 255	00	0.1-second clock pulse (0.05 second ON; 0.05 second OFF)
	01	0.2-second clock pulse (0.1 second ON; 0.1 second OFF)
	02	1.0-second clock pulse (0.5 second ON; 0.5 second OFF)
	03	Instruction Execution Error (ER) Flag Turns ON when an error occurs during execution of an instruction.
	04	Carry (CY) Flag Turns ON when there is a carry in the results of an instruction execution.
	05	Greater Than (GR) Flag Turns ON when the result of a comparison operation is "greater."
	06	Equals (EQ) Flag Turns ON when the result of a comparison operation is "equal," or when the result of an instruction execution is 0.
	07	Less Than (LE) Flag Turns ON when the result of a c9mparison operation is "less."
	08 to 15	Not used.

AR Area

These bits mainly serve as flags related to CPM1A operation. These bits retain their status even after the CPM1A power supply has been turned off or when operation begins or stops.

Word(s)	Bit(s)	Function
AR00, AR01	00 to 15	Not used.
AR02	00	I/O Units Status Flag (First Unit)
	01	I/O Units Status Flag (Second Unit)
	02	110 Units Status Flag (Third Unit)
	03 to 07	Not used.
	12 to 15	Number of I/O Units Connected
AR03 to AR 07	00to 15	Not Used.
AR08	00 to 07	Not Used
	08 to 11	Peripheral Device Error Code
	12	Peripheral Device Error Flag
	13	Peripheral Device Transmission Enabled Flag
	14 to 15	Not Used.
AR09	00 to 15	Not Used.
AR10	00 to 15	Power-off Counter (4 digits BCD) This is the count of the number of times that the power has been turned off. To clear the count, write "0000" from a peripheral device.
AR11	00 to 07	High-speed Counter Range Comparison Flags 00 ON: Counter PY is within comparison range 1 0l ON: Counter PV is within comparison range 2 02 ON: Counter PV is within comparison range 3 03 ON: Counter PV is within comparison range 4 04 ON: Counter PV is within comparison range 5 05 ON: Counter PY is within comparison range 6 06 ON: Counter PV is within comparison range 7 07 ON: Counter PV is within comparison range 8
	08 to 14	Not used.
	15	Pulse Output Status ON: Stopped. OFF: Pulse being output.
AR12	00 to 15	Not Used.
AR13	00	Power-up PC Setup Error Flag Turns ON when there is an error in DM 6600 to DM 6614 (the part of the PC Setup area that is read at power-up).
	01	Start-up PC Setup Error Flag Turns ON when there is an error in DM 6615 to DM 6644 (the part of the PC Setup area that is read at the beginning of operation).
	02	RUN PC Setup Error Flag Turns ON when there is an error in DM 6645 to DM 6655 (the part of the PC Setup area that is always read).
	03,04	Not Used.
	05	Long Cycle Time Flag Turns ON if the actual cycle time is longer than the cycle time set in DM 6619.
	06,07	Not Used.

10. Appendix D- Memory Areas

Word(s)	Bit(s)	Function
AR13	08	Memory Area Specification Error Flag Turns ON when a non-existent data area address is specified in the program.
	09	Flash Memory Error Flag Turns ON when there is an error in flash memory.
	10	Read-only DM Error Flag Turns ON when a checksum error occurs in the read-only DM (DM 6144 to DM 6599) and that area is initialized.
	11	PC Setup Error Flag Turns ON when a checksum error occurs in the PC Setup area.
	12	Program Error Flag Turns ON when a checksum error occurs in the program memory (UM) area, or when an improper instruction is executed.
	13	Expansion Instruction Error Flag Turns ON when a checksum error occurs in the expansion instruction data.
	14,15	Not Used.
AR14	00 to 15	Maximum Cycle mile (4 digits BCD) The longest cycle time since the beginning of operation is stored. It is cleared at the beginning, and not at the end, of operation. The units can be any of the following, depending on the setting of in DM 6618. Default: 0.1 ms; ~ ms" setting: 0.1 ms; "100 ms" setting: 1 ins "1 5" setting: 10 ms
AR15	00 to 15	Current Cycle Time (4 digits BCD) The most recent cycle time during operation is stored. The Current Cycle Time is not cleared when operation stops. The units can be any of the following, depending on the setting of in DM 6618. Default: 0.1 ms; "10 ms" setting: 0.1 ms; "looms" setting: 1 ms; "1 5"

• CPM2A Memory Area Functions

10. Appendix D- Memory Areas

Memory Area Structure

The following memory area can be used with the CPM2A.

Data area		Words	Bits	Function
IR area	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	These bits can be allocated to the external I/O terminals
	Output area	IR010 to IR019 (10 words)	IR 01000toIR 01915 (160 bits)	
	Work area	IR 020 to IR 049 IR200 to IR 231 (58 words)	IR 02000 to IR 04915 IR 20000 to IR 23115 (928 bits)	Work bits can be freely used within the program
SR area		SR 228 to SR 255 (28 words)	SR 22800 to SR 25515 (440 bits)	These bits serve specific functions such as flags and control bits.
TR area		--	TR 0 to TR 7 (8 bits)	These bits are used to temporarily store ON/OFF status at program branches
HR area2		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	These bits store data and retain their ON/OFF status when power is turned off.
AR area		AR00 to AR 23 (23 words)	AR 0000 to AR 2315 (384 bits)	These bits serve specific functions such as flags an control bits.
LR area1		LR 00 to LR 15 (16 words)	LR 00000 to LR 1515 (256 bits)	Used for a 1:1 data link with another PLC.
Timer /Counter Area		TC 000 to TC 225 (timer/counter numbers)3		The same numbers are used for both timers and counters.
DM area	Read/write 2	DM 0000 to DM1999 DM2022 to DM 2047 (2,026 words)	---	DM area data can be accessed in words units only. Word values are retained when the power is turned off.
	Error Log 4	DM2000 to DM 2021 (22 words)	---	Used to store the time of occurrence and error code of errors that occurred. These words can be used as ordinary read/write DM when the error log function isn't being used.
	Read-only 4	DM6144 to DM 6599 (456 words)	---	Cannot be overwritten from program
	PC Setup4	DM 6600 to DM 6655 (56 words)	---	Used to store various parameters that control PLC operation

Note:
1. IR and LR bits that are not used for their allocated function can be used as work bits.
2. The contents of the HR area, LR area, Counter area, and read/write DM are backed up by the CPU Unit's battery. IF the battery is removed or fails, the content of these area will be lost and unstable.
3. When a TC numbers is used as a word operand, the timer or counter PV is accessed; when used as a bit operand, its Completion Flag is accessed.
4. Data is DM 6144 to DM *6655* cannot be overwritten from the program, but they can be changed from a Programming Device.
5. The program and data in DM 6144 to DM *6655* are stored in flash memory.

10. Appendix D- Memory Areas

SR Area

These bits mainly serve as flags related to CPM2A operation or contain present and set values for various functions. The functions of the SR area are explained in the following

Word(s)	Bit(s)	Function	
SR 228	00 to 15	Pulse output PV 0 96777215 (16777215) to 16777215	Low 4 digits
SR 229	00 to 15	When negative value, Left most bit = 1. This CH cannot be sued as aux relay even when Pulse output is not used	High 4 digits
SR 230	00 to 15	Pulse output PV 1 96777215 (16777215) to 16777215	Low 4 digits
SR 231	00 to 15	When negative value, Left most bit = 1. This CH cannot be sued as aux relay even when Pulse output is not used	High 4 digits
SR 232 to SR 235	00 to 15	Macro Function Input Area Contains the input operands for MCRO(99). (Can be used as work bits when MCRO(99) is not used.)	
SR 236 to SR 239	00 to 15	Macro Function Output Area Contains the output operands for MCRO(99). (Can be used as work bits when MCRO(99) is not used.)	
SR240	00 to 15	Input Interrupt 0 Counter Mode SV SV when input interrupt 0 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 0 is not used in counter mode.)	
SR241	00 to 15	Input Interrupt 1 Counter Mode SV SV when input interrupt 1 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 1 is not used in counter mode.)	
SR242	00 to 15	Input Interrupt Z Counter Mode SV SV when input interrupt 2 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 2 is not used in counter mode.)	
SR 243	00 to 15	Input Interrupt counter Mode SV SV when input interrupt 3 is used in counter mode (4 digits hexadecimal). (Can be used as work bits when input interrupt 3 is not used in counter mode.)	
SR 244	00 to 15	Input Interrupt 0 Counter Mode PV Counter PV when input interrupt 0 is used in counter mode (4 digits hexadecimal).	
SR 245	00 to 15	Input Interrupt 1 Counter Mode PV Counter PV when input interrupt 1 is used in counter mode (4 digits hexadecimal).	
SR 246	00 to 15	Input Interrupt 2 Counter Mode PV Counter PV when input interrupt 2 is used in counter mode (4 digits hexadecimal).	
SR 247	00 to 15	Input Interrupt 3 Counter Mode PV Counter PV when input interrupt 3 is used in counter mode (4 digits hexadecimal).	
SR 248, SR 249	00 to 15	High-speed Counter PV Area Differential Pulse Input mode: F8388608(-8388608) to 08388607 Pulse + Direction input mode: F8388608(-8388608) to 08388607 Reversible Pulse Input mode: F8388608(-8388608) to 08388607 Increment mode: 00000000 to 16777215 Sync mode:00000000 to 00020000(Hz) When negative value, Left most byte is F	
SR 250	00 to 15	Analog Volume Setting 0 Used to store the 4-digit BCD set value (0000 to 0200) from analog volume control 0.	
SR 251	00 to 15	Analog Volume Setting 1 Used to store the 4-digit BCD set value (0000 to 0200) from analog volume control 1.	

10. Appendix D- Memory Areas

Word(s)	Bit(s)	Function
SR 252	00	High-Speed Counter Reset Bit
	01 to 03	Not Used
	04	Pulse Output 0 PV Reset Bit Turn ON to clear the PV of pulse output 0.
	05	Pulse Output 0 PV Reset Bit Turn ON to clear the PV of pulse output 1.
	06,07	Not Used.
	08	Peripheral Port Reset Bit Turn ON reset the Peripheral Port. Automatically turns OFF when reset is complete.
	09	RS-232C Port Reset Bit Turn ON reset the RS-232C Port. Automatically turns OFF when reset is complete.
	10	PC Setup Reset Bit Turn ON to intialize PC setup (DM 6600 through DM6655). Automatically turns OFF again when reset is complete. Only effective if the PC is in PROGRAM mode.
	11	Forced Status Hold Bit OFF : The force status of bits that are forced set/reset is cleared when switching between PROGRAM mode and MONITOR mode. ON : The status of the bits that are forced set/reset are maintained when switching between PROGRAM mode and MONITOR mode.
	12	I/O Hold Bit OFF : IR and LR bits are reset when starting or stopping operation. ON : IR and LR bit status is maintained when starting or stopping operation.
	13	Not Used.
	14	Error Log Reset Bit Turn ON to clear error log. Automatically turns OFF again when operation is complete.
	15	Not Used.
SR 253	00 to 07	FAL Error Code The error code is stored here when an error occurs. The FAL number is stored here when FAL (06) or FAL (07) is executed. This word is reset (to 000) by executing a FAL 00 instruction or by clearing the error from a Programming Device.
	08	Battery Error Flag Turns ON when the CPU Unit backup battery's voltage is too low.
	09	Cycle Time Overrun Flag Turns ON when a cycle time overrun occurs
	10,11	Not Used.
	12	Changing RS-232C setup Flag Turns ON when the RS-232C port's setting are being changed.
	13	Always ON Flag.
	14	Always OFF Flag
	15	First Cycle Flag Turn s ON for 1 cycle at the start of operation.
SR254	00	1-minute clock pulse (30 seconds ON; 30 seconds OFF)
	01	0.02-second clock pulse (0.01 second ON; 0.01 second OFF)
	02	Negative (N) Flag
	03 to 05	Not used.
	06	Differential Monitor Complete Flag
	07	STEP (08) Execution Flag Turns ON for 1 cycle only at the start of process based on STEP(08).
	08 to 15	Not used.

10. Appendix D- Memory Areas

Word(s)	Bit(s)	Function
SR 255	00	0.1-second clock pulse (0.05 second ON; 0.05 second OFF)
	01	0.2-second clock pulse (0.1 second ON; 0.1 second OFF)
	02	1.0-second clock pulse (0.5 second ON; 0.5 second OFF)
	03	Instruction Execution Error (ER) Flag Turns ON when an error occurs during execution of an instruction.
	04	Carry (CY) Flag Turns ON when there is a carry in the results of an instruction execution.
	05	Greater Than (GR) Flag Turns ON when the result of a comparison operation is "greater."
	06	Equals (EQ) Flag Turns ON when the result of a comparison operation is "equal," or when the result of an instruction execution is 0.
	07	Less Than (LE) Flag Turns ON when the result of a c9mparison operation is "less."
	08 to 15	Not used.

10. Appendix D- Memory Areas

10. Appendix D- Memory Areas

AR Area

These bits mainly serve as flags related to CPM2A operation. These bits retain their status even after the CPM2A power supply has been turned off or when operation begins or stops.

Word(s)	Bit(s)	Function
AR00, AR01	00 to 15	Not used.
AR02	00 to 07	Not Used.
	08 to 11	Number of Expansion Units Connected
	12 to 15	Not Used.
AR03 to AR 07	00 to 15	Not Used.
AR08	00 to 03	RS-232 Port Error Code 0: Normal completion 1: Parity Error 2: Frame error 3: Overrun Error
	04	RS-232C Communication Error Flag Turns ON when an RS-232C communication error occurs.
	05	RS-232C Transmit Ready Flag Valid only when host link, Turns ON when the PLC is ready to transmit data.
	06	RS-232C Reception Completed Flag Valid only when RS-232C communications are used. (No-protocol only.)
	07	RS-232C Reception Overflow Flag Valid only when RS-232C communications are used. (No-protocol only.)
	08 to 11	Peripheral Port Error Code 0: Normal completion 1: Parity Error 2: Frame error 3: Overrun Error
	12	Peripheral Port Communications Error Flag Turns ON when an Peripheral communication error occurs.
	13	Peripheral Port Transmit Ready Flag Turns ON when the PLC is ready to transmit data. (Host Link only)
	14	Peripheral Reception Completed Flag Turns ON when the PLC has completed reading data. (No-Protocol only)
	15	Peripheral Port Reception Overflow Flag Turns ON when an overflow has occurred. (No-Protocol only)
AR09	00 to 15	RS-232C Port Reception Counter (4 digits BCD) Valid only when no-protocol communication are used.
AR10	00 to 15	Peripheral Device Reception Counter Valid only when no-protocol communication are used.

10. Appendix D- Memory Areas

AR Area

Word(s)	Bit(s)	Function
AR11	00 to 07	High-speed Counter Range Comparison Flags 00 ON: Counter PY is within comparison range 1 0l ON: Counter PV is within comparison range 2 02 ON: Counter PV is within comparison range 3 03 ON: Counter PV is within comparison range 4 04 ON: Counter PV is within comparison range 5 05 ON: Counter PY is within comparison range 6 06 ON: Counter PV is within comparison range 7 07 ON: Counter PV is within comparison range 8
	08	High Speed Counter Comparison ON: Operating OFF: Stopped
	09	High-Speed Counter PV overflow/underflow ON: Overflow/Underflow occurred. OFF: Normal operation
	10	Not Used.
	11	Pulse Output 0 Accelerating /Decelerating Flag ON: Accelerate/Decelerate OFF: Constant rate
	12	Pulse Output 0 Overflow/underflow Flag ON: Overflow/Underflow OFF: Normal
	13	Pulse Output 0 Pulse Quantity Set Flag ON: Pulse quantity has been set. OFF: Pulse quantity has not been set.
	14	Pulse Output 0 Output Completed Flag ON: Completed OFF: Not Completed
	15	Pulse Output 0 Output Status ON: Pulses being output. OFF: Stopped.
AR12	00 to 11	Not Used.
	12	Pulse Output 1 Overflow/underflow Flag ON: Overflow/Underflow OFF: Normal
	13	Pulse Output 1 Pulse Quantity Set Flag ON: Pulse quantity has been set. OFF: Pulse quantity has not been set.
	14	Pulse Output 1 Output Completed Flag ON: Completed OFF: Not Completed
	15	Pulse Output 1 Output Status ON: Pulses being output. OFF: Stopped.

AR Area

Word(s)	Bit(s)	Function
AR13	00	Power-up PC Setup Error Flag Turns ON when there is an error in DM 6600 to DM 6614 (the part of the PC Setup area that is read at power-up).
	01	Start-up PC Setup Error Flag Turns ON when there is an error in DM 6615 to DM 6644 (the part of the PC Setup area that is read at the beginning of operation).
	02	RUN PC Setup Error Flag Turns ON when there is an error in DM 6645 to DM 6655 (the part of the PC Setup area that is always read).
	03,04	Not Used.
	05	Long Cycle Time Flag Turns ON if the actual cycle time is longer than the cycle time set in DM 6619.
	06,07	Not Used.
	08	Memory Area Specification Error Flag Turns ON when a non-existent data area address is specified in the program.
	09	Flash Memory Error Flag Turns ON when there is an error in flash memory.
	10	Read-only DM Error Flag Turns ON when a checksum error occurs in the read-only DM (DM 6144 to DM 6599) and that area is initialized.
	11	PC Setup Error Flag Turns ON when a checksum error occurs in the PC Setup area.
	12	Program Error Flag Turns ON when a checksum error occurs in the program memory (UM) area, or when an improper instruction is executed.
	13	Expansion Instruction Error Flag Turns ON when a checksum error occurs in the expansion instruction data.
	14	Data Save Error Flag Urns ON if the data could not be retained with the backup battery.
	15	Not Used.
AR14	00 to 15	Maximum Cycle Time (4 digits BCD) The longest cycle time since the beginning of operation is stored. It is not cleared when operation stops, buts iti is cleared when the operation starts again.
AR15	00 to 15	Current Cycle Time (4 digits BCD) The most recent cycle time during operation is stored. The current Cycle Time is not cleared when operation stops.
AR16	00 to 15	Not Used.
AR17	00 to 07	Minute (00 to 59, BCD)
	08 to 15	Hour (00 to 59, BCD)
AR18	00 to 07	Second (00 to 59, BCD
	08 to 15	Minute (00 to 59, BCD)
AR19	00 to 07	Hour (00 to 23, BCD)
	08 to 15	Day of the Month (01 to31, BCD)

10. Appendix D- Memory Areas

AR Area

Word(s)	Bit(s)	Function
AR20	00 to 07	Month (01 to 12, BCD)
	08 to 15	Year (00 to 99,BCD)
AR21	00 to 07	Day of the Week 00: Sunday 01: Monday 02: Tuesday 03: Wednesday 04: Thursday 05: Friday 06: Saturday
	08 to 12	Not Used.
	13	30-second Compensation Bit
	14	Clock Stop Bit Turn this bit ON to stop the clock. The time/data can be overwritten while this bit is ON.
	15	Clock Set Bit To change the time/date, turn ON AR 2114, write the new time/date, and then turn this bit ON to enable a new time/date setting.
AR22	00 to 15	Not Used.
AR23	00 to 15	Power off Counter (4 digit BCD) This is the count of the number of times that the power has been turned off. To clear the count , write “0000” from a Programming Device.

10. Appendix E- PC Setup

CPM1A PC Setup Settings

The PLC Setup is broadly divided into four categories: 1) Settings related to basic PLC operation and I/O processes, 2) Settings related to the cycle time, 3) Setting related to interrupts, and 4) Settings related to communications. This section will explain the settings according to these classifications.

The following table shows the settings for CPMlA PLCs in order.

Word(s)	Bit(s)	Function
Startup Processing (DM 6600 to DM 6614) The following settings are effective after transfer to the PLC only after the PLC is restarted.		
DM 6600	00 to 07	Startup mode (effective when bits 08 to 15 are set to 02). 00: PROGRAM; 01: MONITOR 02: RUN
	08 to 15	Startup mode designation 00: Programming Console switch 01: Continue operating mode last used before power was turned off 02: Setting in 00 to 07
DM 6601	00 to 07	Not Used. (Set 0 00.)
	08 to 11	IOM Hold Bit (SR 25212) Status 0: Reset; 1: Maintain
	12 to 15	Forced Status Hold Bit (SR 25211) Status 0: Reset; 1: Maintain
DM 6602	00 to 03	Program memory write-protection 0: Program memory unprotected 1: Program memory write-protected (except DM 6602 itself)
	04 to 07	Programming Console display language 0: English; 1: Japanese
	08 to 11	Not Used.
DM 6603	00 to 15	Not Used.
DM 6604	00 to 07	00: A memory error will not be generated if data could not be retained by the battery. 01: A memory error will be generated if data could not be retained by the battery.
	08 to 15	Not Used.
DM 6605 to DM6614	00 to 15	Not Used.
Cycle Time Settings (DM 6615 to DM 6619) The following settings are effective after transfer to the PC the next time operation is started.		
DM 6615, DM6616	00 to 15	Not Used.
DM 6617	00 to 07	Servicing time for peripheral port (effective when bits 08 to 15 are set to 01) 00 to 99 (BCD): Percentage of cycle time used to service peripheral.
	08 to 15	Peripheral port servicing setting enable 00:5% of the cycle time 01: Use time in 00 to 07
DM 6618	00 to 07	Cycle monitor time (effective when bits 08 to 15 are set to 01, 02, or 03) 00 to 99 (BCD): Setting (see 08 to 15)
	08 to 15	Cycle monitor enable (Setting in 00 to 07 x unit; 99 5 max.) 00:120 ms (setting in bits 00 to 07 disabled) 01: Setting unit: 10 ms 02: Setting unit: 100 ms 03: Setting unit: 1 s

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
DM6619	00 to 15	Cycle Time 0000: Variable (no minimum) 0001 to 9999 (BCD): Minimum time in ms.
Interrupt Processing (DM 6620 to DM 6639) The following settings are effective after transfer to the PLC the next time operation is started.		
DM 6620	00 to 03	Input constant for IR 00000 to IR 00002 0:0.8 ms; 1:1 ms; 2: 2 ms; 3: 4 ms; 4: 8 ms; 5:16 ms; 6: 32 ms; 7: 64 ms; 8:128 ms
	04 to 07	Input constant for IR 00003 and IR 00004 (Setting same as bits 00 to 03)
	08 to 11	Input constant for IR 00005 and IR 00006 (Setting same as bits 00 to 03)
	12 to 15	Input constant for IR 00007 and IR 00011 (Setting same as bits 00 to 03)
DM 6621	00 to 07	Input constant for IR 001 0: 8 ms; 1:1 ms; 2: 2 ms; 3: 4 ms; 4: 8 ms; 5:16 ms; 6: 32 ms; 7: 64 ms; 8:128 ms
	08 to 15	Input constant for IR 002 (Setting same as for IR 001).
DM 6622	00 to 07	Input constant for IR 003 (Setting same as for IR 001).
	08 to 15	Input constant for IR 004 (Setting same as for IR 001).
DM 6623	00 to 07	Input constant for IR 005 (Setting same as for IR 001).
	08 to 15	Input constant for IR 006 (Setting same as for IR 001).
DM 6624	00 to 07	Input constant for IR 007 (Setting same as for IR 001).
		Input constant for IR 008 (Setting same as for IR 001).
DM 6625	00 to 07	Input constant for IR 008 (Setting same as for IR 001).
	08 to 15	Input constant for IR 009 (Setting same as for IR 001).
DM6626 to DM6627	00 to 15	Not Used.
DM6628	00 to 03	Interrupt enable for IR 00000 (0: Normal input; 1: Interrupt input; 2: Quick-response)
	04 to 07	Interrupt enable for IR 00001(0: Normal input; 1: Interrupt input; 2: Quick-response)
	08 to 11	Interrupt enable for IR 00002 (0: Normal input; 1: Interrupt input; 2: Quick-response)
	12 to 15	Interrupt enable for IR 00003 (0: Normal input; 1: Interrupt input; 2: Quick-response)
High-speed Counter Settings (DM 6640 to DM 6644) The following settings are effective after transfer to the PC the next time operation is started.		
DM6640 to DM6641	00 to 15	Not Used.
DM6642	00 to 03	High-speed counter mode 0: Up/down counter mode; 4: Incrementing counter mode
	04 to 07	High-speed counter reset mode 0: Z phase and software reset; 1: Software reset only
	08 to 15	High-speed counter enable 00: Don't use high-speed counter; 01: Use high-speed counter with settings in 00 to 07
DM6643, DM6644	00 to 15	Not Used.

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
Peripheral Port Settings The following settings are effective after transfer to the PLC.		
DM 6645 to DM6649	00 to 15	Not Used.
DM6650	00 to 07	Port settings 00: Standard (1 start bit, 7-bit data, even parity, 2 stop bits, 9, 6000 bps) 01: Settings in DM 6651 (Other settings will cause a non-fatal error and AR 1302 will turn ON.)
	08 to 11	Link area for one-to-one PC link via peripheral port: 0: LR00toLR 15
	12 to 15	Communications mode 0: Host link; 2: One-to-one PC link (slave) 3: One-to-one PC link (master); 4: NT link (Other settings will cause a non-fatal error and AR 1302 will turn ON.)
DM 6651	00 to 07	Baud rate 00:1.2K, 01: 2.4K, 02: 4.8K, 03: 9.6K, 04:19.2K
	08 to 15	Frame format Start Data Stop Parity 00: 1 bit 7 bits 1 bit Even 01: 1 bit 7 bits 1 bit Odd 02: 1 bit 7 bits 1 bit None 03: 1 bit 7 bits 2 bit Even 04: 1 bit 7 bits 2 bit Odd 05: 1 bit 7 bits 2 bit None 06: 1 bit 8 bits 1 bit Even 07: 1 bit 8 bits 1 bit Odd 08: 1 bit 8 bits 1 bit None 09: 1 bit 8 bits 2 bit Even 10: 1 bit 8 bits 2 bit Odd 11: 1 bit 8 bits 2bit None (Other settings will cause a non-fatal error and AR 1302 will turn ON.)
DM 6652	00 to 15	Transmission delay (Host Link) 0000 to 9999: in ms. (Other settings will cause a non-fatal error and AR 1302 will turn ON.)
DM 6653	00 to 07	Node number (Host Link) 00to31 (BCD) (Other settings will cause a non-fatal error and AR 1302 will turn ON.)
	08 to 15	Not Used.
DM 6654	00 to 15	Not Used.
Error Log Settings (DM 6655) The following settings are effective after transfer to the PLC.		
DM 6655	00 to 03	Style 0: Shift after 7 records have been stored 1: Store only first 7 records (no shifting) 2 to F: Do not store records
	04 to 07	Not Used.
	08 to 11	Cycle Time monitor enable. 0: Detect long cycles as non-fatal errors 1: Do not detect long cycles
	12 to 15	Not Used.

10. Appendix E- PC Setup

CPM2A PC Setup Settings

The PLC Setup is broadly divided into four categories: 1) Settings related to basic PLC operation and I/O processes, 2) Settings related to the cycle time, 3) Setting related to interrupts, and 4) Settings related to communications. This section explain the settings according to these classifications.

The following table shows the settings for CPM2A PLCs in order.

Word(s)	Bit(s)	Function
Startup Processing (DM 6600 to DM 6614) The following settings are effective after transfer to the PLC only after the PLC is restarted.		
DM 6600	00 to 07	Startup mode (effective when bits 08 to 15 are set to 02). 00: PROGRAM; 01: MONITOR 02: RUN
	08 to 15	Startup mode designation 00: Programming Console switch 01: Continue operating mode last used before power was turned off 02: Setting in 00 to 07
DM 6601	00 to 07	Not Used. (Set 0 00.)
	08 to 11	IOM Hold Bit (SR 25212) Status at startup 0: Reset; 1: Maintain
	12 to 15	Forced Status Hold Bit (SR 25211) Status at startup 0: Reset; 1: Maintain
DM 6602	00 to 03	Program memory write-protection 0: Program memory unprotected 1: Program memory write-protected (except DM 6602 itself)
	04 to 07	Programming Console display language 0: English; 1: Japanese
	08 to 11	Expansion Instruction function code assignments.
	12 to 15	Not Used.
DM 6603	0 to 15	Not used.
DM 6604	00 to 07	00: Memory error will not be generated if data could not be retained by the battery 01: Memory error will not be generated if data could not be retained by the battery.
	08 to 15	Not Used.
DM 6605 to DM6614	00 to 15	Not Used.

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
Cycle Time Settings (DM 6615 to DM 6619) The following settings are effective after transfer to the PC the next time operation is started.		
DM 6615	00 to 15	Not Used.
DM 6616	00 to 07	Servicing time for RS-232C port (effective when bits 08 to 15 are set to 01) 00 to 99 (BCD): Percentage of cycle time used to service RS-232C port.
	08 to 15	RS-232C port servicing setting enable 00:5% of the cycle time 01: Use time in 00 to 07
DM 6617	00 to 07	Servicing time for peripheral port (effective when bits 08 to 15 are set to 01) 00 to 99 (BCD): Percentage of cycle time used to service peripheral.
	08 to 15	Peripheral port servicing setting enable 00:5% of the cycle time 01: Use time in 00 to 07
DM 6618	00 to 07	Cycle monitor time (effective when bits 08 to 15 are set to 01, 02, or 03) 00 to 99 (BCD): Setting (see 08 to 15)
	08 to 15	Cycle monitor enable (Setting in 00 to 07 x unit; 99 5 max.) 00:120 ms (setting in bits 00 to 07 disabled) 01: Setting unit: 10 ms 02: Setting unit: 100 ms 03: Setting unit: 1 s
DM6619	00 to 15	Cycle Time 0000: Variable (no minimum) 0001 to 9999 (BCD): Minimum time in ms.
Interrupt Processing (DM 6620 to DM 6639) The following settings are effective after transfer to the PLC the next time operation is started.		
DM 6620	00 to 03	Input time constant for IR 00000 to IR 00002 0:10 ms; 1:1 ms; 2: 2 ms; 3: 3 ms; 4: 5 ms; 5:10 ms; 6: 20 ms; 7: 40 ms; 8: 80 ms
	04 to 07	Input time constant for IR 00003 and IR 00004 (Setting same as bits 00 to 03)
	08 to 11	Input time constant for IR 00005 and IR 00006 (Setting same as bits 00 to 03)
	12 to 15	Input time constant for IR 00007 and IR 00011 (Setting same as bits 00 to 03)
DM 6621	00 to 07	Input time constant for IR 001 00: 10 ms; 01:1 ms; 02: 2 ms; 03: 3 ms; 04: 5 ms; 05: 10 ms; 06: 20 ms; 07: 40 ms; 08: 80 ms
	08 to 15	Input constant for IR 002 (Setting same as for IR 001).
DM 6622	00 to 07	Input constant for IR 003 (Setting same as for IR 001).
	08 to 15	Input constant for IR 004 (Setting same as for IR 001).
DM 6623	00 to 07	Input constant for IR 005 (Setting same as for IR 001).
	08 to 15	Input constant for IR 006 (Setting same as for IR 001).
DM 6624	00 to 07	Input constant for IR 007 (Setting same as for IR 001).
	08 to 15	Input constant for IR 008 (Setting same as for IR 001).
DM 6625	00 to 07	Input constant for IR 008 (Setting same as for IR 001).
	08 to 15	Input constant for IR 009 (Setting same as for IR 001).
DM6626 to DM6627	00 to 15	Not Used.
DM6628	00 to 03	Interrupt enable for IR 00003 (0: Normal input; 1: Interrupt input; 2: Quick-response)
	04 to 07	Interrupt enable for IR 00004 (0: Normal input; 1: Interrupt input; 2: Quick-response)
	08 to 11	Interrupt enable for IR 00005 (0: Normal input; 1: Interrupt input; 2: Quick-response)
	12 to 15	Interrupt enable for IR 00006 (0: Normal input; 1: Interrupt input; 2: Quick-response)

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
DM 6629	00 to 03	PV coordinate system for pulse output 0 0: Relative coordinates; 1: Absolute coordinates
	04 to 07	PV coordinated system for pulse output 1 0: Relative coordinates; 1: Absolute coordinates
	08 to 15	Not Used.
DM 6630 to DM 6641	00 to 15	Not Used.
High-speed Counter Settings (DM 6640 to DM 6644) The following settings are effective after transfer to the PC the next time operation is started.		
DM6640 to DM6641	00 to 15	Not Used.
DM6642	00 to 03	High-speed counter mode 0: Differential phase mode (5kHz) 1: Pulse + direction input mode (20kHz) 2: Up/down input mode (20kHz) 4: Increment mode (20kHz)
	04 to 07	High-speed counter reset mode 0: Z phase and software reset; 1: Software reset only
DM 6642	08 to 15	High-speed counter/Synchronized pulse control for IR 00000 to 1R00002 00: Don't use either function. 01: Use as high speed counters. 02: Use for Synchronized pulse control (10 Hz to 500 Hz). 03: Use for Synchronized pulse control (20 Hz to 1 kHz). 04: Use for Synchronized pulse control (300 Hz to 20 kHz).]
DM 6643, DM6644	00 to 15	Not Used.
RS-232C Port Communications Settings The following settings are effective after transfer to the PC. If the CPU Unit's Communication Switch is ON, communications through the RS-232C port are governed by the default settings (all 0) regardless of the setting in DM 6645 through DM 6649.		
DM 6645	00 to 03	Port settings 00: Standard (1 start bit, 7-bit data, even parity, 2 stop bits, 9, 6000 bps) 01: Settings in DM 6646 (Any other settings will cause a non-fatal error and AR1302 will turn ON.)
	04 to 07	CTS control setting 0: Disable CTS control; 1: Enable CTS control. (Any other settings will cause a non-fatal error and AR1302 will turn ON.)
	08 to 11	Link Words for one-to-one link: 0: LR 00 to LR15 (Other setting is ineffective.)
	12 to 15	Communications mode 0: Host link; 1 No-protocol; 2: 1:1 PC link (Slave); 3: 1:1 PC link (Master); 4:NT link (Any other setting specifies host link mode, causes a non-fatal error, and turns ON AR1302.)

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
DM6646	00 to 07	Baud rate 00:1.2K, 01: 2.4K, 02: 4.8K, 03: 9.6K, 04:19.2K Os to 15 Frame format
	08 to 15	Frame Format Start Data Stop Parity 00: 1 bit 7 bits 1 bit Even 01: 1 bit 7 bits 1 bit Odd 02: 1 bit 7 bits 1 bit None 03: 1 bit 7 bits 2 bit Even 04: 1 bit 7 bits 2 bit Odd 05: 1 bit 7 bits 2 bit None 06: 1 bit 8 bits 1 bit Even 07: 1 bit 8 bits 1 bit Odd 08: 1 bit 8 bits 1 bit None 09: 1 bit 8 bits 2 bit Even 10: 1 bit 8 bits 2 bit Odd 11: 1 bit 8 bits 2bit None (Any other settings specifies standard settings (1 start bit, 7 data bits; even parity, 2 stop bits, 9600 bps), causes a non-fatal error, and turn ON AR1302)
DM 6647	00 to 15	Transmission (0000 to 9999 BCD set a delay of 0 to 99,990 ms) (Any other setting specifies a delay of 0 ms, causes a non-fatal error, turns ON AR1302.)
DM6648	00 to 07	Node Number (Host Link) 00 to 31 (BCD) (Any other setting specifies a node number of 00, causes a non-fatal error, and turns ON AR1302.)
	08 to 11	Start code selection for no-protocol communications. 0: Disable start code; 1 Enables start code in DM 6649 (Any other setting disables the start code, cause a non-fatal error, and turn ON AR1302).
	12 to 15	End code selection for no-protocol communications. 0: Disable end code; 1 :Enables end code in DM 6649; 2:Sets end code of CR,LF (Any other setting disables the end code, cause a non-fatal error, and turn ON AR1302).
DM6649	00 to 07	Start Code (00 to FF) (This setting is valid only when bits 8 to 11 of DM 6648 are set to 1.)
	08 to 15	When bits 12 to 15 of DM 6648 set to 0: Set the number of bytes to receive. (00:256 bytes; 01 to FF: 1 to 255 bytes) When bits 12 to 15 of DM 6648 set to 1: Sets the end code. (00 to FF).

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
Peripheral Port Communications Settings The following settings are effective alter transfer to the PC. If the CPU Unit's Communication Switch is ON, communications through the Peripheral port are governed by the default settings (all 0) regardless of the setting in DM 6650 through DM 6654. The communication Switch settings has no effect on the communications with the Programming Console connected to the peripheral port or Support software set for peripheral bus communications. The CPU unit will auto-detect either Programming Device and automatically establish communications.		
DM6650	00 to 03	Port settings 00: Standard (1 start bit, 7-bit data, even parity, 2 stop bits, 9, 6000 bps) 01: Settings in DM 6651 (Any other setting specifies standard settings, causes a non-fatal error, and turns ON AR1302.)
	04 to 11	Not Used.
	12 to 15	Communications mode 0: Host link; 2: No-protocol (Any other setting specifies host link, causes a non-fatal error, and turns ON AR 1302.)
DM 6651	00 to 07	Baud rate 00:1.2K, 01: 2.4K, 02: 4.8K, 03: 9.6K, 04:19.2K
	08 to 15	Frame format Start Data Stop Parity 00: 1 bit 7 bits 1 bit Even 01: 1 bit 7 bits 1 bit Odd 02: 1 bit 7 bits 1 bit None 03: 1 bit 7 bits 2 bit Even 04: 1 bit 7 bits 2 bit Odd 05: 1 bit 7 bits 2 bit None 06: 1 bit 8 bits 1 bit Even 07: 1 bit 8 bits 1 bit Odd 08: 1 bit 8 bits 1 bit None 09: 1 bit 8 bits 2 bit Even 10: 1 bit 8 bits 2 bit Odd 11: 1 bit 8 bits 2 bit None Any other settings specifies standard settings (1 start bit, 7 data bits; even parity, 2 stop bits, 9600 bps), causes a non-fatal error, and turn ON AR1302)
DM 6652	00 to 15	Transmission delay (Host Link) 0000 to 9999: in ms. (Any other setting specifies a delay of 0 ms, causes a non-fatal error, turns ON AR1302.)
DM 6653	00 to 07	Node number (Host Link) 00to31 (BCD) Any other setting specifies a node number of 00, causes a non-fatal error, and turns ON AR1302.)
	08 to 11	Start code selection for no-protocol communications. 0: Disable start code; 1 Enables start code in DM 6649 (Any other setting disables the start code, cause a non-fatal error, and turn ON AR1302).
	12 to 15	End code selection for no-protocol communications. 0: Disable end code; 1 :Enables end code in DM 6649; 2:Sets end code of CR,LF (Any other setting disables the end code, cause a non-fatal error, and turn ON AR1302).

10. Appendix E- PC Setup

Word(s)	Bit(s)	Function
DM 6654	00 to 07	Start Code (00 to FF) (This setting is valid only when bits 8 to 11 of DM 6648 are set to 1.)
	08 to 15	When bits 12 to 15 of DM 6648 set to 0: Set the number of bytes to receive. (00:256 bytes; 01 to FF: 1 to 255 bytes) When bits 12 to 15 of DM 6648 set to 1: Sets the end code. (00 to FF).
Error Log Settings (DM 6655) The following settings are effective after transfer to the PLC.		
DM 6655	00 to 03	Style 0: Shift after 7 records have been stored 1: Store only first 7 records (no shifting) 2 to F: Do not store records
	04 to 07	Not Used.
	08 to 11	Cycle time monitor enable 0: Generate a non-fatal error for a cycle time that is too long. 1: Do not generate a non-fatal error.
	12 to 15	Low battery error enable 0: Generate a non-fatal error for low battery voltage. 1: Do not generate a non-fatal error.