DOCUMENT RESUME

ED 339 860 | CE 059 795

AUTHOR	Insolia, Gerard; Anderson, Kathleen
TITLE	Programmable Logic Controllers.
INSTITUTION	Northampton Community Coll., Bethlehem, PA.
SPONS AGENCY	Office of Vocational and Adult Education (ED), Washington, DC.
PUB DATE	91
CONTRACT	V100A00063-90
NOTE	222p.; For related documents, see CE 059 793-796.
PUB TYPE	Guides - Classroom Use - Teaching Guides (For Teacher) (052)
EDRS PRICE	MF01/PC09 Plus Postage.
DESCRIPTORS	College Programs; Community Colleges; Computer Assisted Manufacturing; *Computer Software; *Course Content; Industrial Education; *Industrial Training; Job Training; Learning Modules; *Programing; Resource Centers; Retraining; Technical Education; *Technological Advancement; Two Year Colleges; Units of Study
IDENTIFIERS	Northampton County Area Community College PA; *Programable Logic Control

ABSTRACT

This document contains a 40-hour course in programmable logic controllers (PLC), developed for a business-industry technology resource center for firms in eastern Pennsylvania by Northampton Community College. The 10 units of the course cover the following: (1) introduction to programmable logic controllers; (2) DOS primer; (3) prerequisite mathematical understanding; (4) basic principles of operation; (5) discrete input/output systems; (6) basics of PLC programming; (7) implementing and programming the PLC; (8) guidelines for installation, start-up, and maintenance; (9) PLC system selection guidelines; and (10) course review. Units include information sheets, transparency masters, and (in some cases) pretests and exercises. (KC)

ED339860

PROGRAMMABLE LOGIC CONTROLLERS

Developed by
Gerard Insolia
Kathleen Anderson

Center for Business & Industry
Northampton Community College

PROGRAMMABLE LOGIC CONTROLLERS
(40 hours program)

I. INTRODUCTION TO PROGRAMMABLE LOGIC CONTROLLERS
   A. Definition
   B. Historical Background
   C. Basic Principles of Operation
   D. PLCs vs Other Types of Controls
   E. Typical Areas of PLC Applications
   F. PLC Size & Application
   G. The Benefits of Using PLCs

II. DOS PRIMER

III. PREREQUISITE MATHEMATICAL UNDERSTANDING
   A. Number Systems & Binary Codes
   B. Logic Concepts
   C. Entering a Ladder Diagram

IV. BASIC PRINCIPLES OF OPERATION
   A. Processors, Power Supply System, & Programming Devices
      1. Processors & Processor Scan
      2. Subsystems, Error Checking, Diagnostics
      3. The System Power Supply
      4. Programming Devices
   B. The Memory System & I/O Interaction
      1. Memory Types
      2. Memory Structure & Capacity
      3. Memory Organization & I/O Interaction
      4. Memory Map Examples and I/O Addressing
      5. Memory Considerations

V. DISCRETE INPUT/OUTPUT SYSTEMS
      1. I/O Rack Enclosures and Table Mapping
      2. Remote I/O Systems
      3. Discrete Inputs
      4. Discrete Outputs
      5. Interpreting I/O Specifications

VI. BASICS OF PLC PROGRAMMING
   A. Basic PLC Instructions and Addressing
      1. Types of instructions
      2. Instruction addressing
      3. Branch instructions
   B. Programming Timers
   C. Programming Counters
   D. Program Flow Control Instructions
   E. Program Data Manipulation Instructions

VII. IMPLEMENTING & PROGRAMMING THE PLC
   A. Control Definition
   B. Control Strategy
   C. Implementation Guidelines
   D. Programming Organization
   E. PLC System Documentation

VIII. GUIDELINES FOR INSTALLATION, START-UP, AND MAINTENANCE
   A. System Layout
   B. Power Requirements and Safety Circuitry
   C. Noise, Heat, and Voltage Considerations
   D. I/O Installation, Wiring, and Precautions
   E. PLC Start-up, and Checking Procedures
   F. PLC System Maintenance
   G. Basic Troubleshooting

IX. PLC SYSTEM SELECTION GUIDELINES
   A. PLC Sizes and Scope of Applications
   B. Process Control System Definition

X. REVIEW OF COURSE MATERIAL

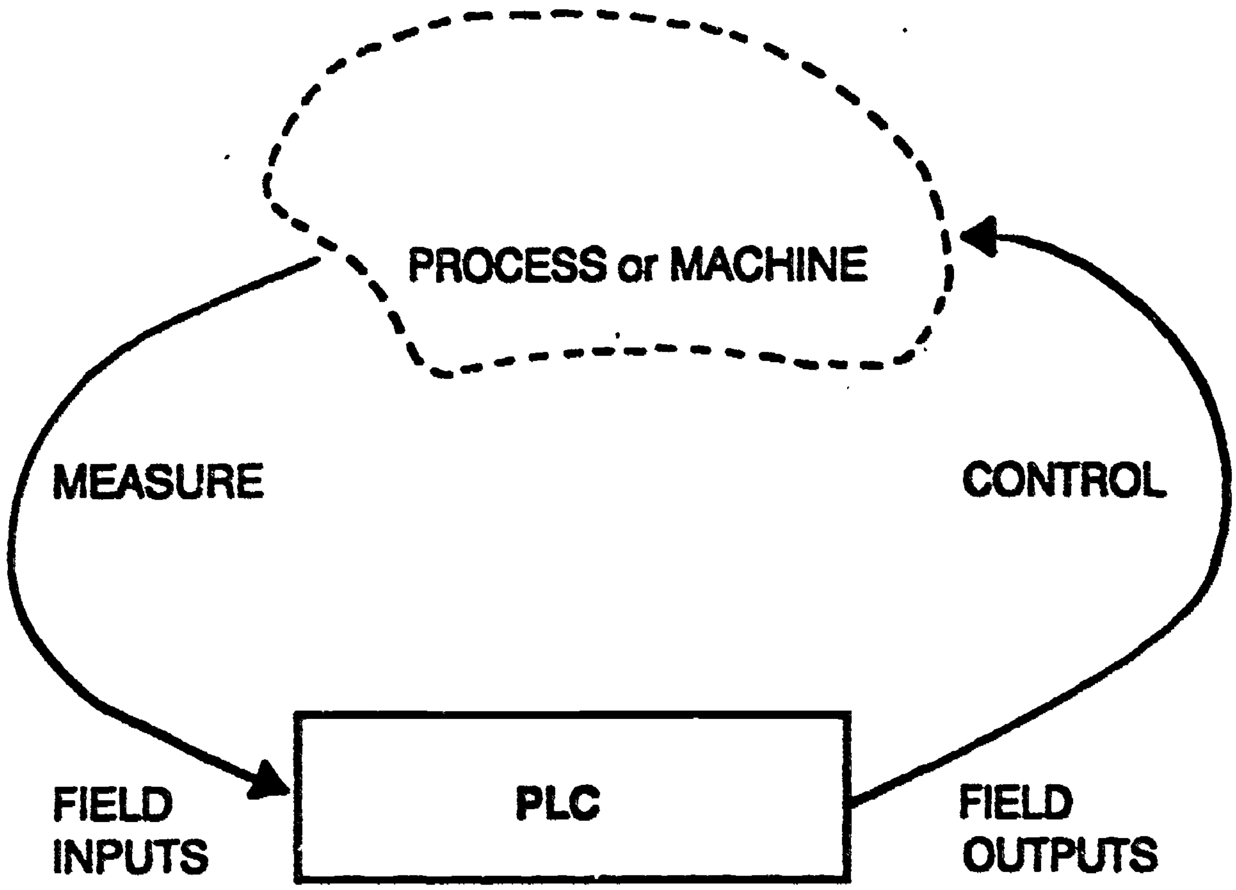
RECOMMENDED TEXT: Programmable Controllers: Theory & Implementation/Industrial Text Co.

CORPORATE SPONSORS
AUTOMATED MANUFACTURING

Air Products & Chemicals, Inc.
Allen-Bradley
Engineered Systems, Inc.
The Du Pont Company
Fastman, Inc.
Fischer & Porter Company
Follett Corporation
GE Fanuc Corporation
Integrated Facilities Corporation
Integrated Technologies Corporation
Jenkins Machine Inc.
Just Born
SI Handling Systems, Inc.
Stanley Vidmar
Texas Instruments
Westinghouse Electric Corporation

INTRODUCTION TO PROGRAMMABLE LOGIC CONTROLLERS

DEFINITION



<u>DEFINITION</u>:

Solid State computer capable of storing instructions to implement control functions to control industrial machines or processes.

TYPICAL INSTRUCTIONS:
- SEQUENCING
- TIMING
- COUNTING
- ARITHMETIC
- DATA MANIPULATION
- COMMUNICATION

HISTORICAL BACKGROUND

Introduction to Programmable Controllers

- Programmable (Logic) controllers: Factory-hardened computers.

- Usually referred to as a PLC (Programmable Logic Controller) to avoid confusion with a personal computer or PC. Sometimes abbreviated as P/C.

- Originally designed as relay replacing devices.

- Reusable.

- Programmed and maintained by plant engineers and technicians.

- Reduce machine downtime with easily replaceable I/O interfaces.

- Provide expandability for the future.

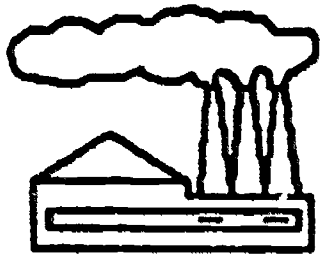
- Advanced by the late 1970's to handle complex mathematical decisions necessary in process control.

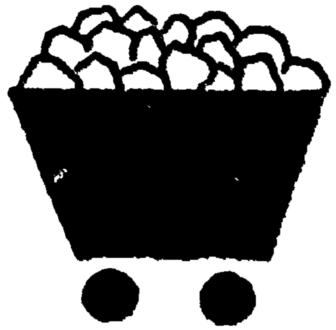
- Terminology is a mixture of electromechanical words and computer jargon.

TYPICAL AREAS OF PLC APPLICATIONS

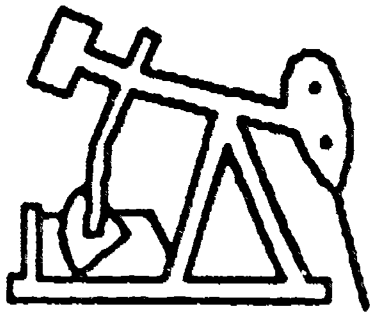
What Are PLCs Used For?

Machine Control
- Car Wash
- Bottling or Packaging
- Material Handling

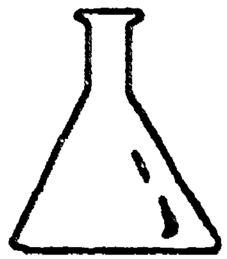


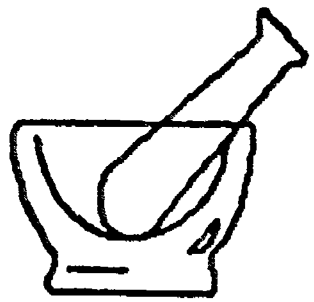


Data Acquisition
- Pipe Line Monitoring
- Hydro Electric Dams



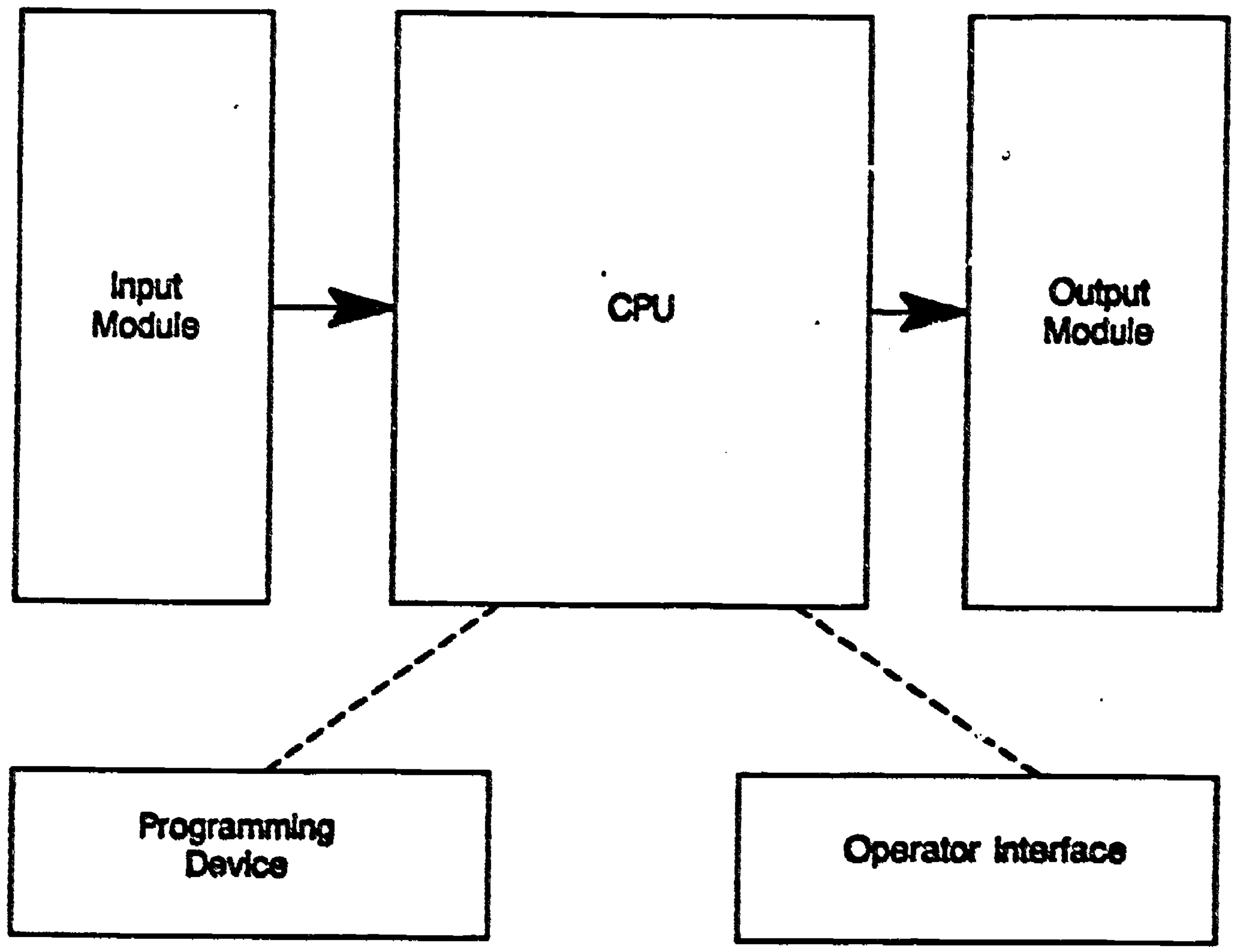
Process Control
- Food Mixing or Cooking
- Chemical Processing





BASIC PRINCIPLES
OF
OPERATION

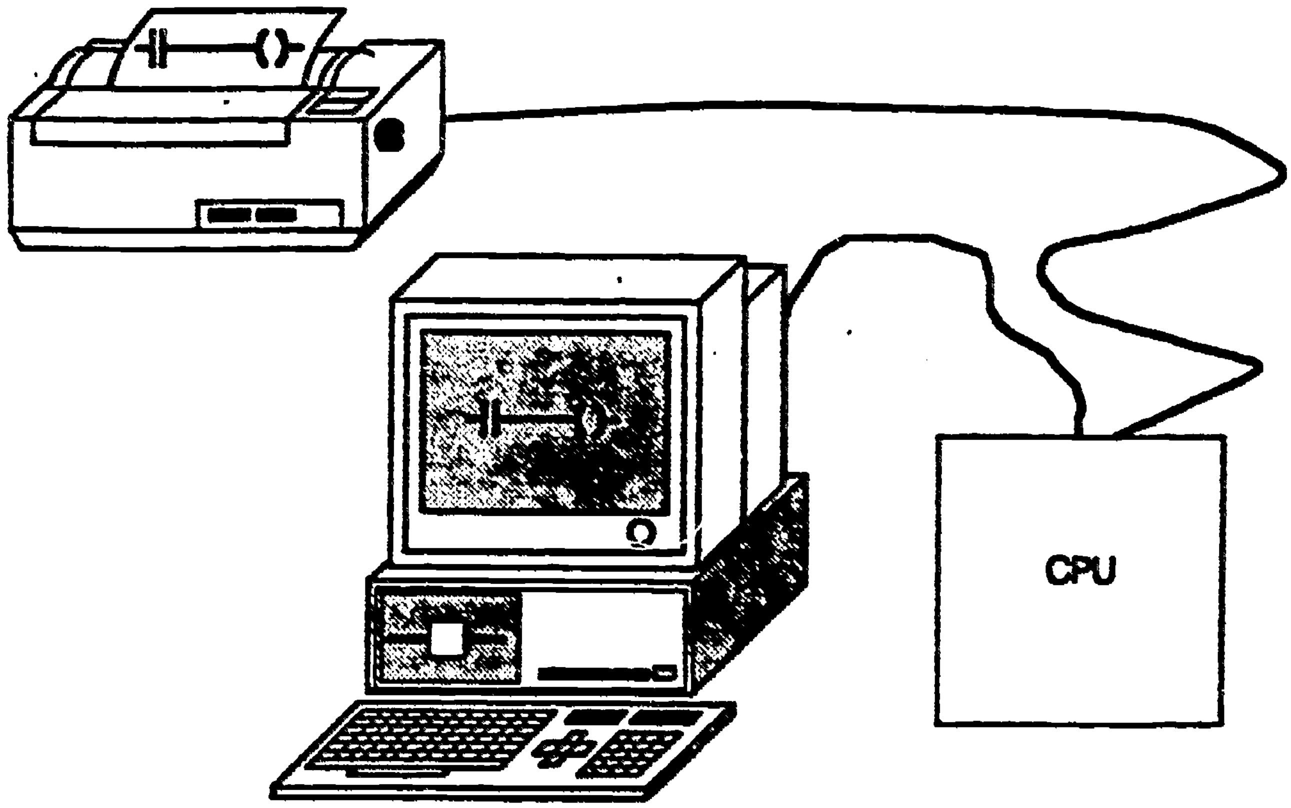
Typical PLC System Components



A typical PLC system is made up of the following components.

- **Input Module** — Brings in the field input signals.
- **Central Processing Unit (CPU)** — Makes decisions based on input signals.
- **Output Module** — Controls field devices.
- **Programming Device** — Device for humans to tell a PLC what to do and when to do it.
- **Operator Interface** — Devices that allow system operators to display process information and enter new parameters for control.

Terminal



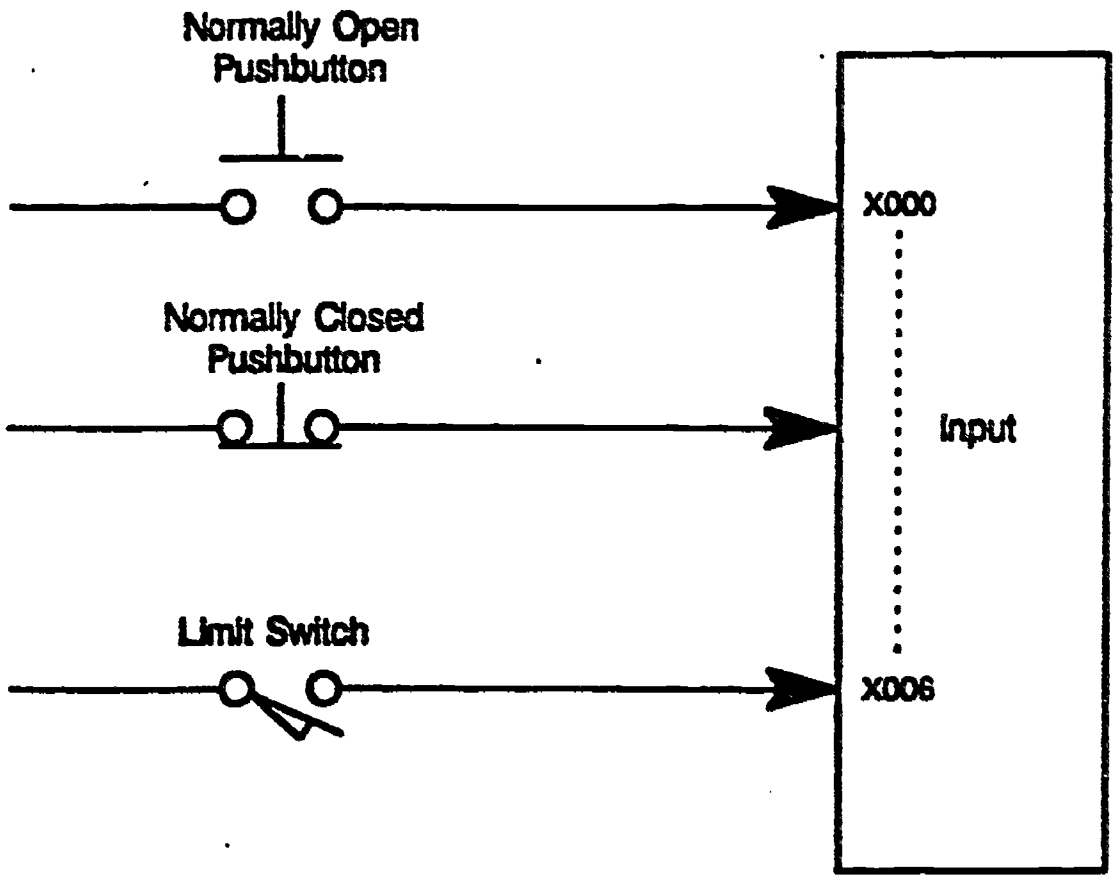
A **Terminal** is generally a piece of hardware that interfaces a human with a computer system.

A *Terminal* can be one of the following.

- CRT — for speed of interaction
- PRINTER — for documenting programs and application data

Hardcopy is the paper printout.

Discrete Inputs



Discrete Input — Field input that is either on or off.

Example: A push button, toggle switch, limit switch, float switch, flow switch, foot switch, pressure switch, proximity switch, etc.

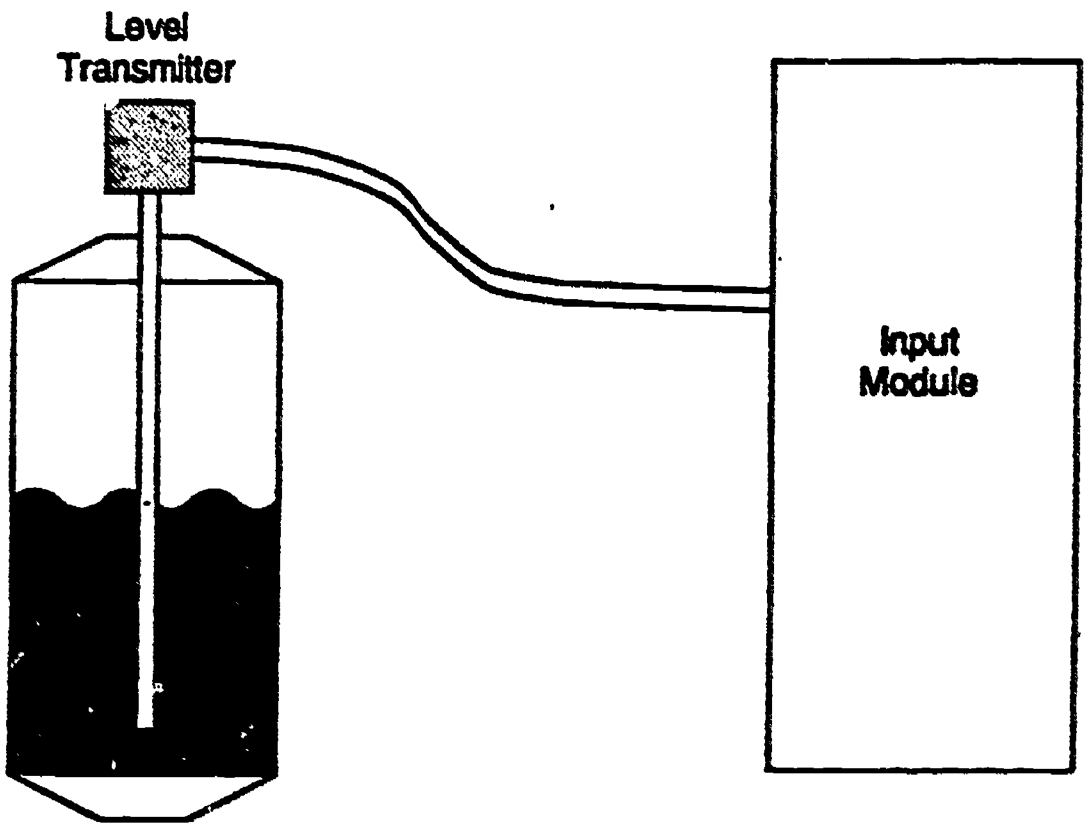
I/O Points — Terminal points where the user connects real world input and output field device wires to the PLC I/O modules.

discrete inputs are given a number that is set by automatic default or manual configuration.

Example: X006 is a discrete input, wired to the 7th point on the first input module.

Digital Input — Another term for a Discrete Signal.

Analog Input



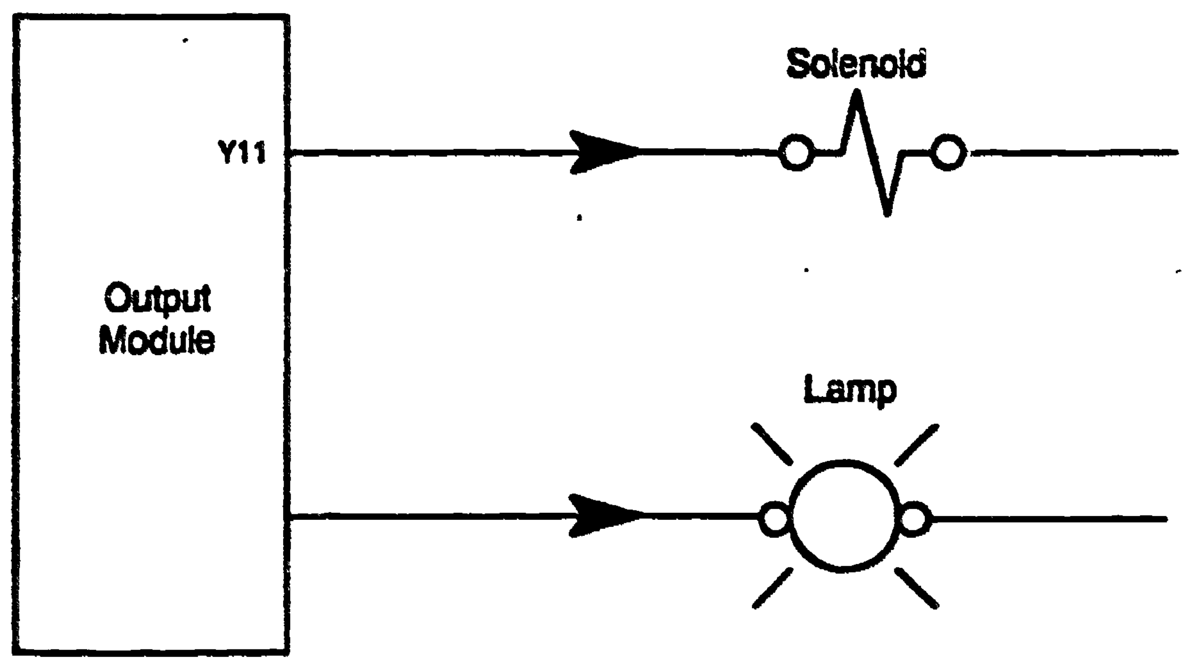
Analog Input — Signal where an electrical signal from a field sensor varies as the process condition changes.

Example: Level transmitter monitoring liquid level in a vessel.

This can be proportionate or disproportionate, depending on the primary element, transmitter, etc..

More on Analog later!

Discrete Outputs



Output signals can also be *Discrete* or *Analog*.

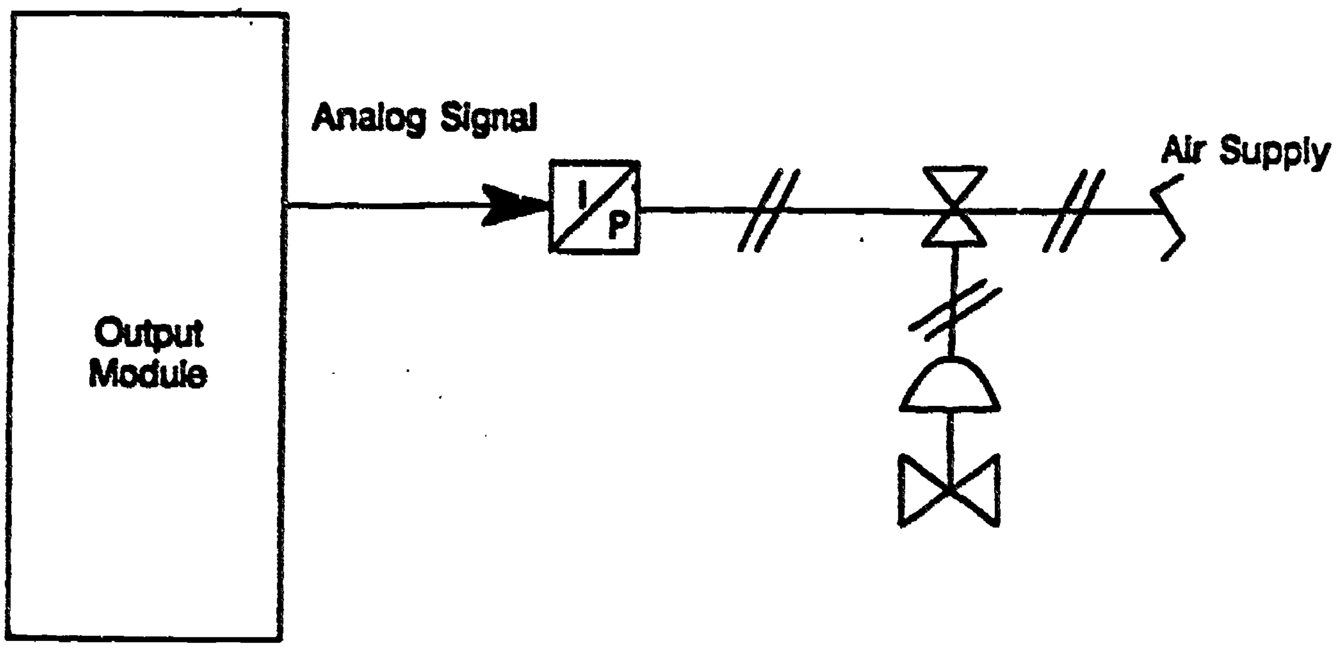
Discrete Output — Used to turn field actuators either on or off.

Example: Solenoids, contactor coils, lamps, etc.

discrete outputs are given a number that is set by automatic default or manual configuration.

Example: Y11 is a discrete output, wired to an output module.

Analog Outputs



Analog output — A signal that controls variable position field devices with a varying electrical signal.

Example: Current to pneumatic transducer for an air-operated flow-control valve.

This can be proportionate or disproportionate, depending on the primary element, transmitter, etc.

More on analog later!

Central Processing Unit (CPU)

Decision Making	User Memory
	1. Decisions to be made 2. Process information

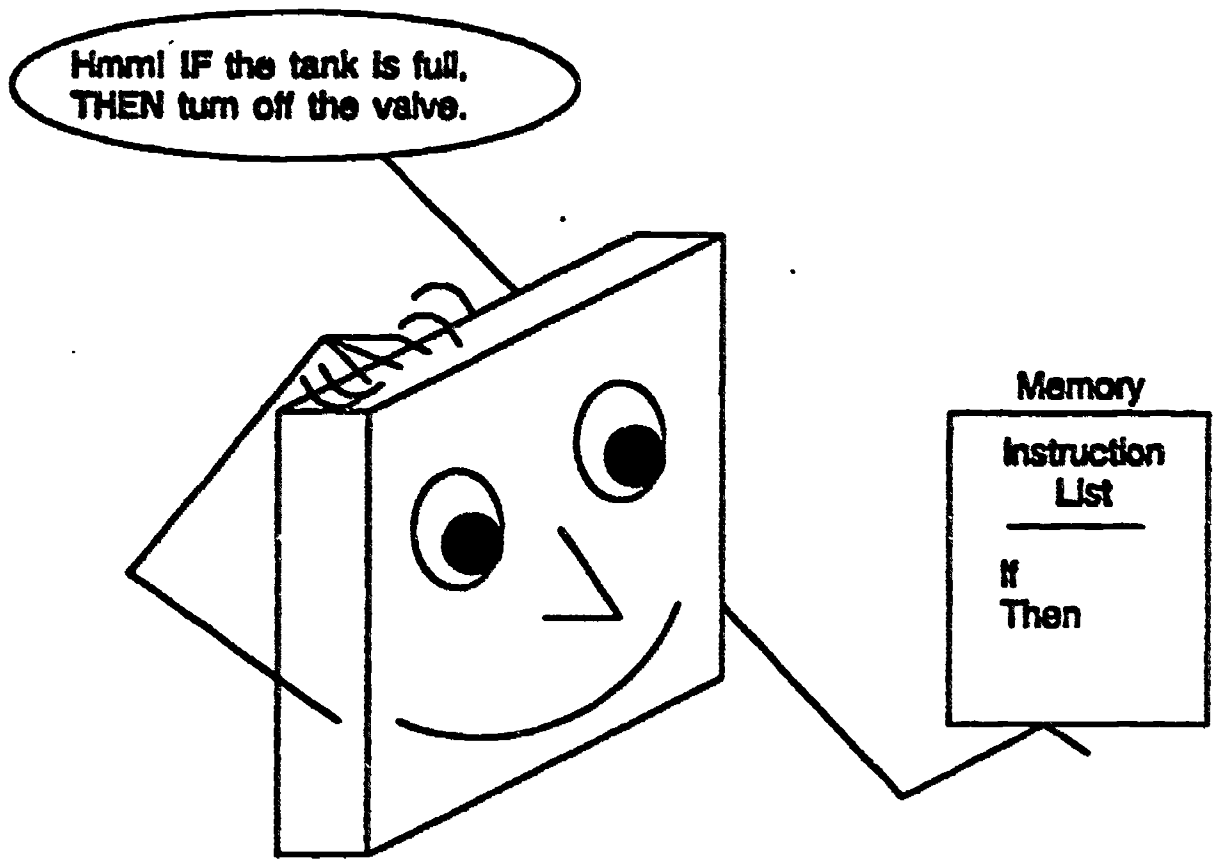
The Central Processing Unit has two major functional sections to a PLC user.

- *Decision Making Section* — Makes decisions based on the signals received from the inputs. These decisions are totally controlled by the user.
- *User Memory* -- Section that stores two types of user information:

  1) Instructions necessary to make the decisions

  2) Process or application dependent information

Decision Making



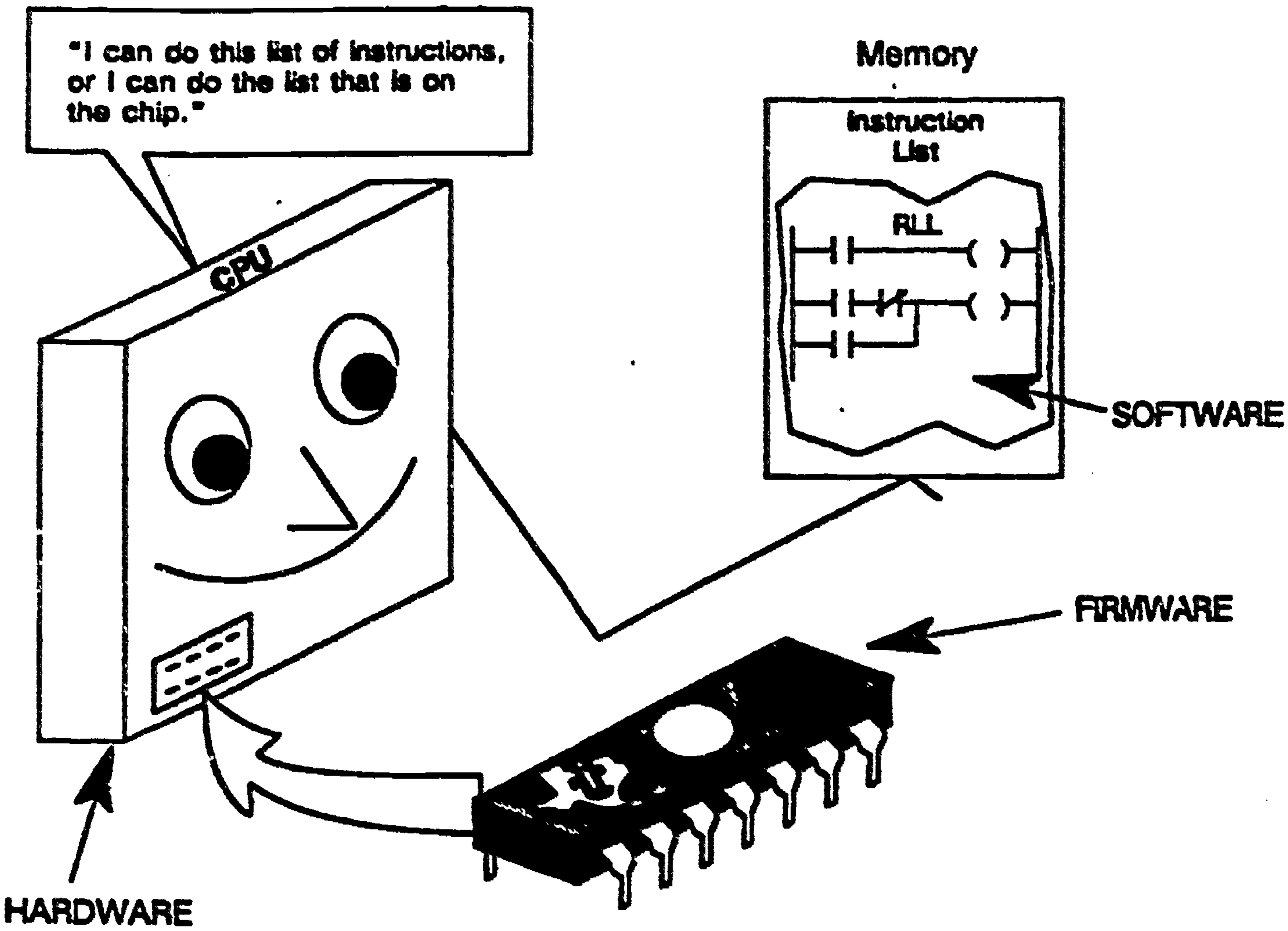
The CPU *makes decisions by following instructions* given by a human.

PLC controlled processes are very efficient because the CPU follows its instructions fast and accurately.

"*Computer Error*" almost always *comes from human or mechanical devices* that give the computer instructions to do the job incorrectly.

Remember, if told to do something wrong, the CPU does it wrong, fast, and accurately!!!

Programming Terms



Just like humans, these instructions must be in a language that the CPU understands.

Relay Ladder Logic (RLL) — The most common PLC language. This was derived from electromechanical schematic symbols.

Machine Stage Programming — Combines relay ladder logic elements with a sequential function chart of the machine to be controlled to develop a control program.

Program — One or more instructions that accomplish a task.

Software — One or more programs that control a process

Hardware — Any part of a PLC system that physically exists.

Firmware — Software that has been physically placed *into* hardware, such as an integrated circuit chip.

PROGRAMMING

HARDWARE

- PC's
- Handhelds

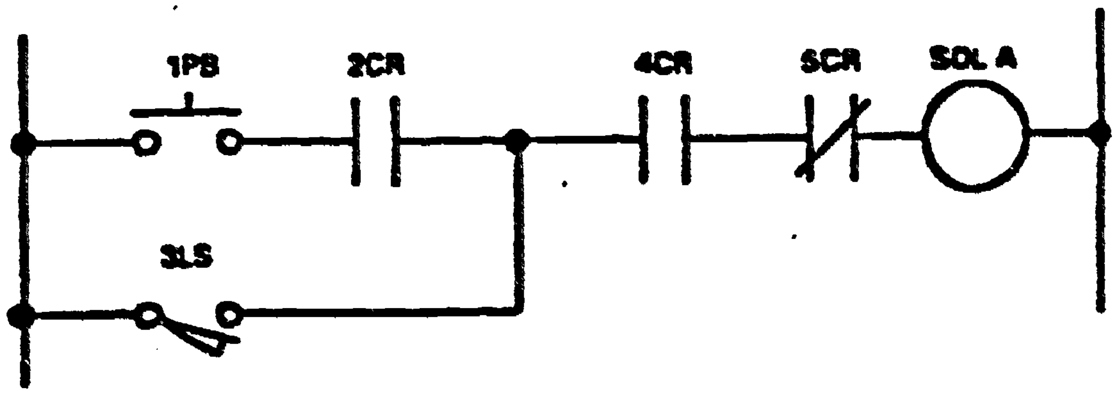
MODE

- On-Line
- Off-Line

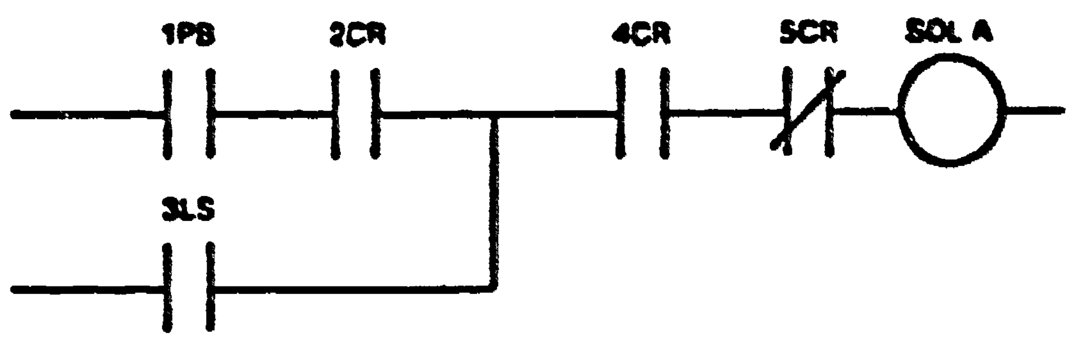
LANGUAGE

- Relay Symbology
- Boolean Logic
- Computer Language

RELAY LADDER DIAGRAM



FREE FORMAT EQUIVALENT PC DIAGRAM



BOOLEAN STATEMENT

$((1PB \bullet 2CR) + 3LS) \bullet 4CR \bullet \overline{5CR} = SOL\ A$

CODE OR MNEMONIC LANGUAGE

LOAD	1PB
AND	2CR
OR	3LS
AND	4CR
CAND	5CR
STORE	SOLA

Here is a comparison of programmable languages that are used with various programmable controllers. The most popular is still the relay ladder diagram because plant personnel are more familiar with it.

PLCs
vs
OTHER TYPES OF CONTROLS

PLCs vs OTHER TYPES OF CONTROL

RELAYS

- PLC offers superior Flexibility
  Reliability
  Capability (Data Collection,
  Expandability)
  Repeatability

- PLC requires less space

PLCs vs OTHER TYPES OF CONTROL

COMPUTERS AND PERSONAL COMPUTERS

A) Environment

Operating temperature	0° to 60°C (32° to 140°F)		
Storage temperature	−20° to 70°C (4° to 158°F)		
Humidity	5 to 95%, noncondensing		
Environmental air	No corrosive gases		
Vibration	MIL STD 810C 514.2		
Shock resistance	MIL STD 810C 516.2		
Voltage withstand	1500 VAC. 1 min. between primary, secondary, FG and RUN contact.		
Insulation resistance	20M ohms. 500 Vdc between primary, secondary, FG and RUN contact.		
Noise immunity	NEMA ICS3-304. Impulse 1000V 1 μs.		
Dimensions (mm)	108 (W) x 150 (H) x 110 (D)		
Voltage range	85 to 132 VAC/170 to 264 VAC		
Frequency range	47 to 63 Hz.		
Input current	1.3A		
Inrush current	20A maximum		
Input power	50W maximum		
Output voltage	5.1V ± 0.25V	20 to 28V	+12V -12V
Ripple	Less than 100mVp-p	More than 1Vp-p	Less than 100mV
Output current	0.1 to 3.7A	0 to 0.4A	25mA

PLCs vs OTHER TYPES OF CONTROL

COMPUTERS AND PERSONAL COMPUTERS

B) Software and Programming

- Self diagnosing interface circuits able to pinpoint malfunctions
- Conventional relay ladder symbols which are familiar to plant personnel
- PLC executes single program sequentially from first to last instruction
  - Computers capable of executing several programs simultaneously and in any order
  - PLCs becoming more intelligent:
    Subroutines
    Interrupt Routines
    Jump Instruction

PLCs vs OTHER TYPES OF CONTROL

COMPUTERS AND PERSONAL COMPUTERS

C) Maintenance

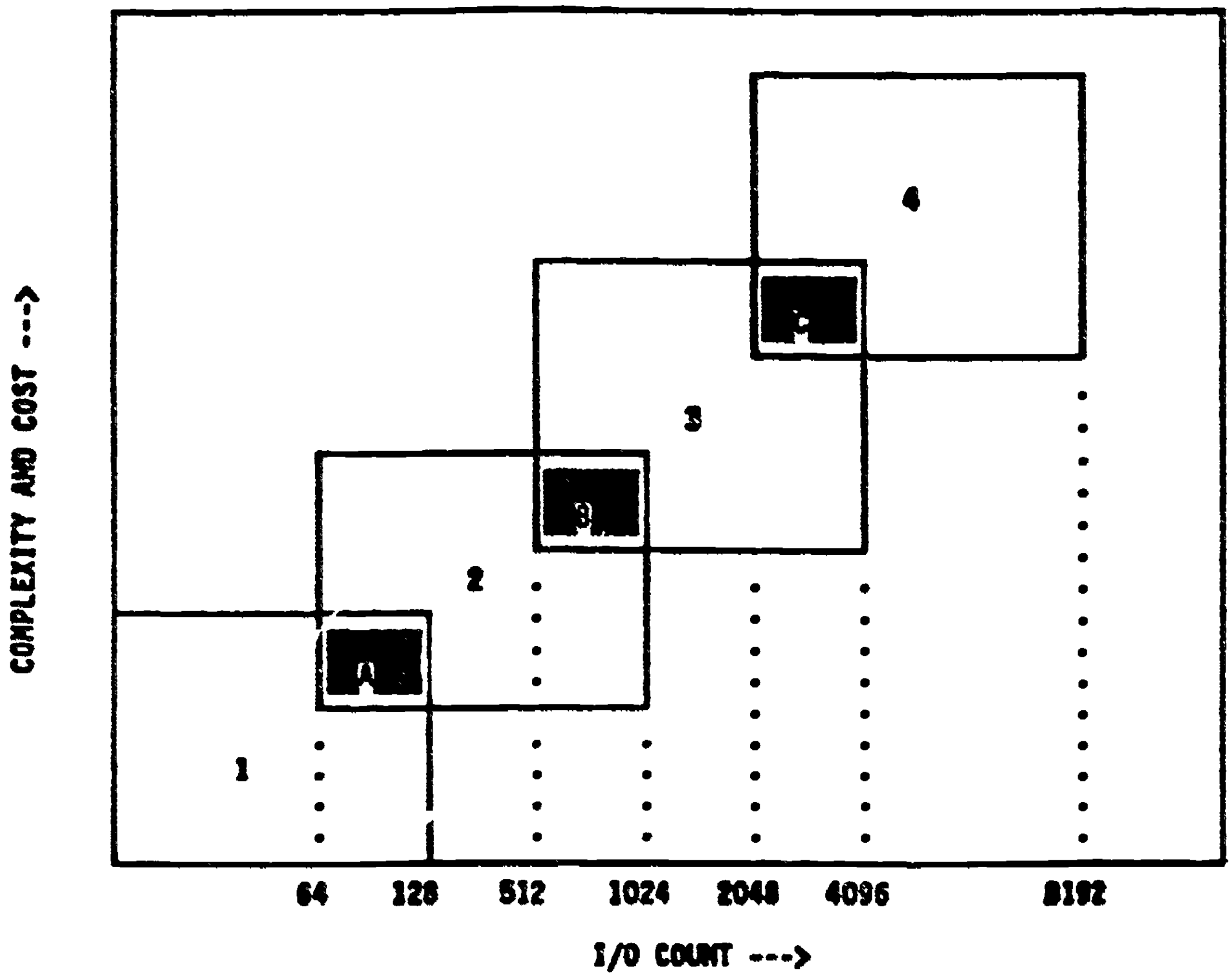
- Modular components which are easily removed and replaced
- Hardware interfaces for connecting field devices are part of the PLC
- Hardware and software designed for use by plant electricians and technicians

PLC SIZES
AND
SCOPE OF APPLICATIONS

PLC SIZE AND APPLICATION

	I/O	MEMORY	CAPABILITY
SMALL	≤128	256-2048 bytes usually CMOS RAM with battery backup	-Simple to Advanced -Discrete
MEDIUM	128 - 2048	32 K Bytes usually CMOS RAM with battery backup	-Simple to Advanced -Discrete and Analog -Data Communication RS-232 or 20 mA loop
LARGE	up to 8192	750 K Bytes	Individual processes to entire plants -Master/slave -Servo drive -Communications

PLC SIZE AND APPLICATION



THE BENEFITS OF USING PLCs

BENEFITS OF USING PLCs

FEATURES AND BENEFITS

INHERENT FEATURES	BENEFITS
Solid-State Components	•High Reliability
Programmable Memory	•Simplifies Changes •Flexible Control
Small Size	•Minimal Space Requirements
Microprocessor Based	•Communication Capability •Higher Level Of Performance •Higher Quality Products •Multi-functional Capability
Software Timers/Counters	•Eliminate Hardware •Easily Changed Presets
Software Control Relays	•Reduce Hardware/Wiring Cost •Reduce Space Requirements
Modular Architecture	•Installation Flexibility •Easily Installed •Hardware Purchase Minimized •Expandability
Variety Of I/O Interface	•Controls a Variety Of Devices •Eliminates Customized Control
Remote I/O Stations	•Eliminate Long Wire/Conduit Run
Diagnostic Indicators	•Reduce Trouble-shooting Time •Signal Proper Operation
Modular I/O Interface	•Neat Appearance of Control Panel •Easily Maintained •Easily Wired
Quick I/O Disconnects	•Service w/o Disturbing Wiring
All System Variables Stored In Memory Data	•Useful Management/Maintenance •Can Be Output In Report Form

BENEFITS OF USING PLCs

INSTALLATION

- Relatively Small Size
  - Locate conveniently
- Remote I/O with Coax or Twisted Pair Connection
  - Reduce material and labor
  - Allow pre-wiring before installation

BENEFITS OF USING PLCs

MAINTENANCE AND TROUBLE-SHOOTING

- Solid-State Components
  - Modular, plug-in systems
- Fault Detection Circuits and Diagnostic Indicators
  - Assess proper operation of components
- Programming Device
  - View programmed logic in operation

DOS PRIMER

REVIEW OF THE PC

- System Configuration
- Diskette Care

DOS DEFINED

- Operating Systems Defined
- MS-DOS & PC-DOS
- Different Versions of DOS

RUNNING DOS

- Loading a Program
- The DOS Startup Disk
- DATE & TIME Commands
- Logged Drive
- System Shutdown

ENTERING DOS COMMANDS

- Internal vs. External Commands
- DOS Functions

DISK FORMATTING

- File Storage on Disk
- FORMAT Command
- Types of Disks & Formats
- FORMAT /S

FILES & DIRECTORIES

- FileDirectories Defined
- DIR Command
- Directory T ree Structure
- MKDIR, CHDIR, RMDIR Commands
- PROMPT Command

FILE COMMANDS

- COPY
- Filenames & Wildcards
- Pathnames
- RENAME
- DELETE/ERASE

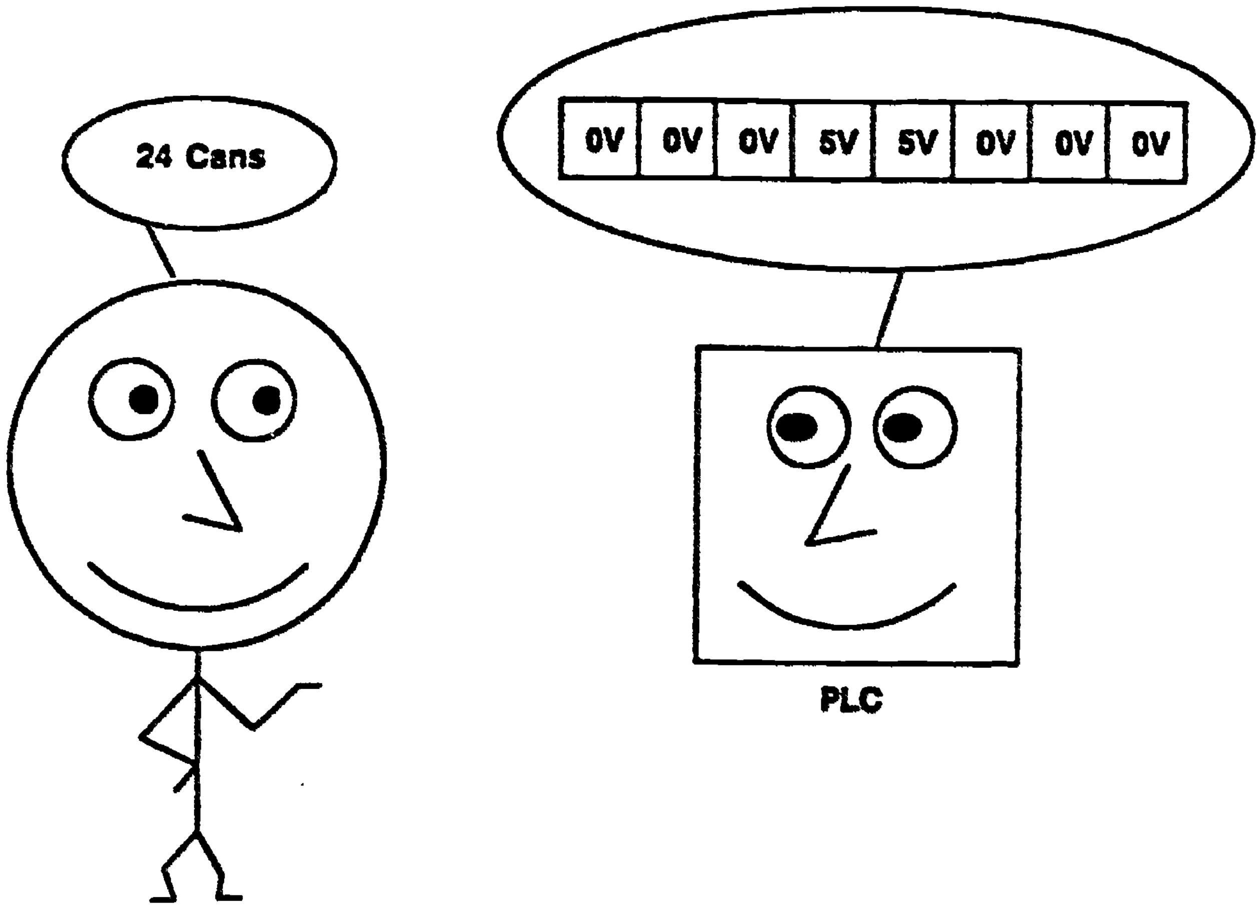
OTHER DOS COMMANDS

- CHKDSK
- DISKCOPY
- DISKCOMP
- BACKUP & RESTORE
- TYPE
- PRINT

PREREQUISITE MATHEMATICAL UNDERSTANDING

NUMBER SYSTEMS AND BINARY CODES

How Information Is Stored In Memory



Data in a memory can be instructions to a PLC or Process Information.

This data is stored electronically.

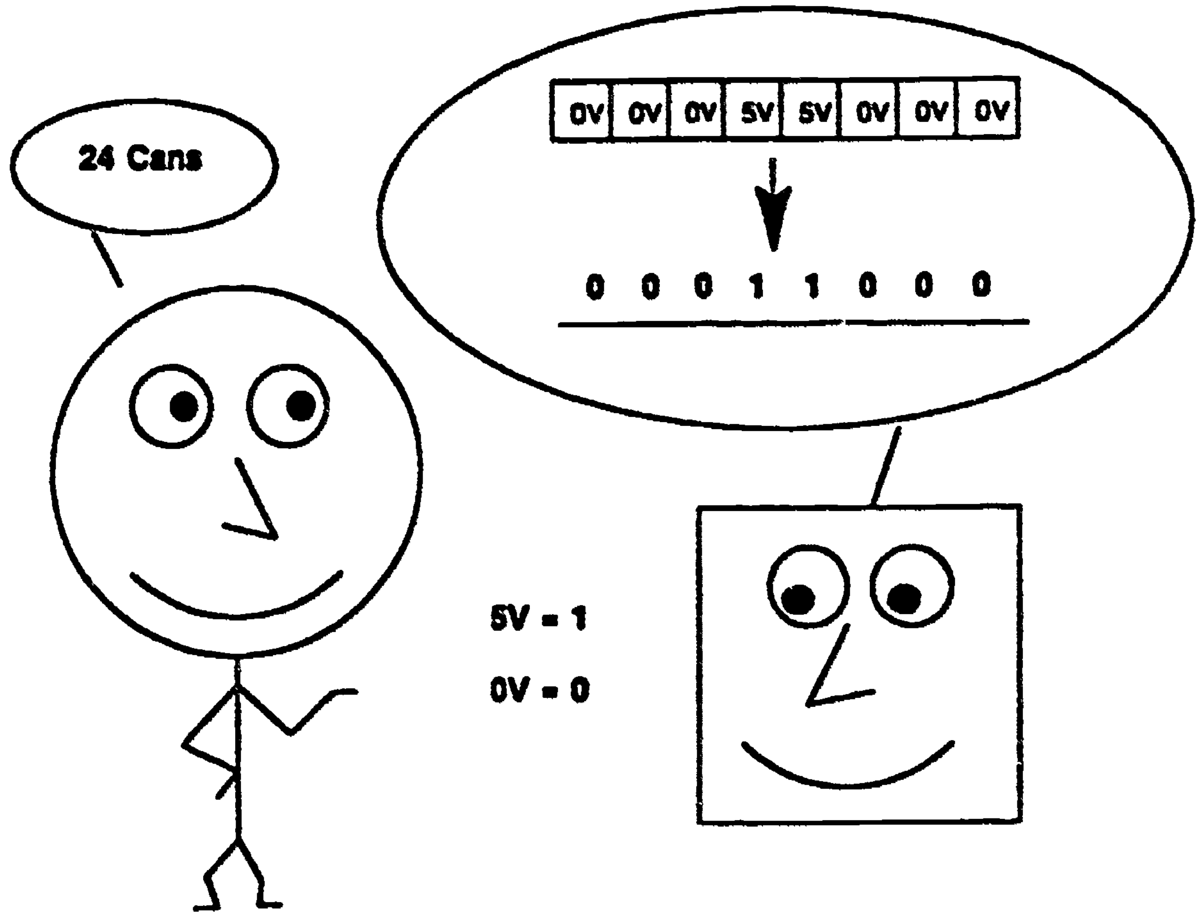
As an example of data storage, consider when a human must enter numbers into a PLC for control purposes.

Humans enter numeric values using the decimal number system.

The memory of the controller stores these numbers using electronic signals such as 0V and 5V.

Translation of these numbers from machine terms to human (decimal Integers) is an important concept.

Binary Numbers



Numeric data is stored in groups of electronic signals such as current, voltage, magnetism, etc.

Using a voltage as an example, there are only 2 electronic signals possible.

Human beings generally find it easier to refer to these as 1's and 0's, where:

5 v = 1

0 v = 0

Since there is no voltage recognized between 0 and 5 volts, each storage group has signals that are 1's or 0's, but never anything else.

The numeric pattern used by programmable controllers, base Two, is called a Binary Number.

NUMBERING SYSTEMS

1. BINARY

   - Usable valid values - 0,1

   - Bit - 1 place

   - Nibble - 4 places

   - Byte - 8 places

   - Word - 16 places or 32 places

BINARY NUMBER SYSTEM

128	64	32	16	8	4	2	1
0	0	0	1	1	0	0	0

Remember in Binary Numbers, there are only 2 valid digits used (0 and 1).

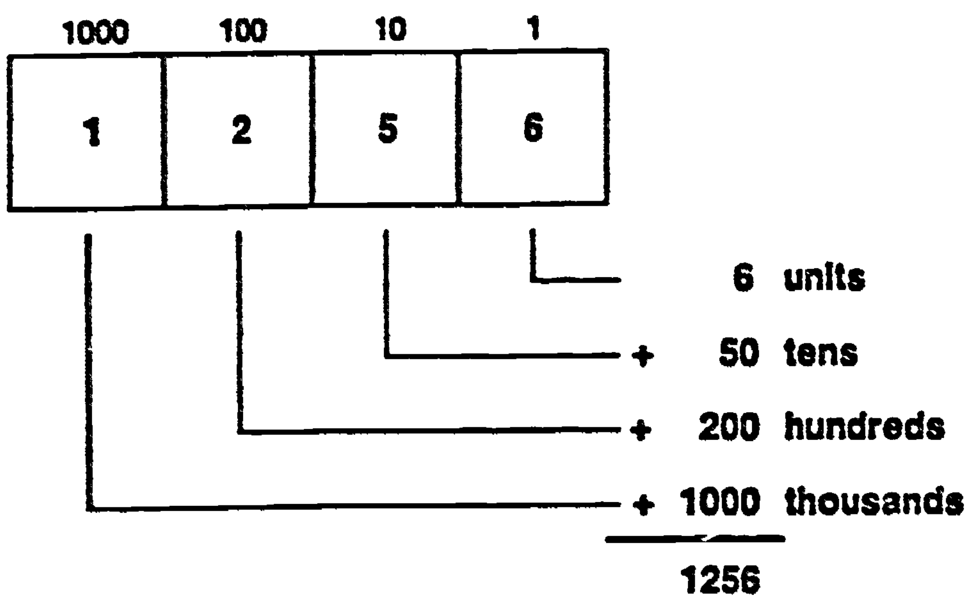
In the Binary Number System, we take a group of 1's and 0's and arrange them into columns.

Each column is weighted with a decimal value which is based on a power of 2.

The columns' decimal weight is generated by first assigning a value of "1" to the right hand column.

Then move from right to left and double the previous weighted value.

DECIMAL NUMBER SYSTEM



Human beings work with decimal numbers everyday, using 10 digits from 0 – 9.

The concept of "weighted columns" is not new.

The Decimal Number System also weights each column, but with a power of 10.

The sum of all the weighted columns is the total decimal number.

To Convert a Binary Number to Decimal (Positive Integer)

128	64	32	16	8	4	2	1
0	0	0	1	1	0	0	0

$$\begin{array}{r} 8 \\ +\ 16 \\ \hline 24 \end{array}$$

To better understand how a PLC can "remember" data, consider the steps on how to interpret a decimal number from a binary value.

Step # *Action*

1. *Search from right to left for "1's".*
2. *Write down the decimal weight of each column that contains a "1".*
3. *Add the column values together.*

The sum of all the weighted columns that contain a "1" is then the decimal number that the PLC has stored.

BINARY NUMBER EXERCISE

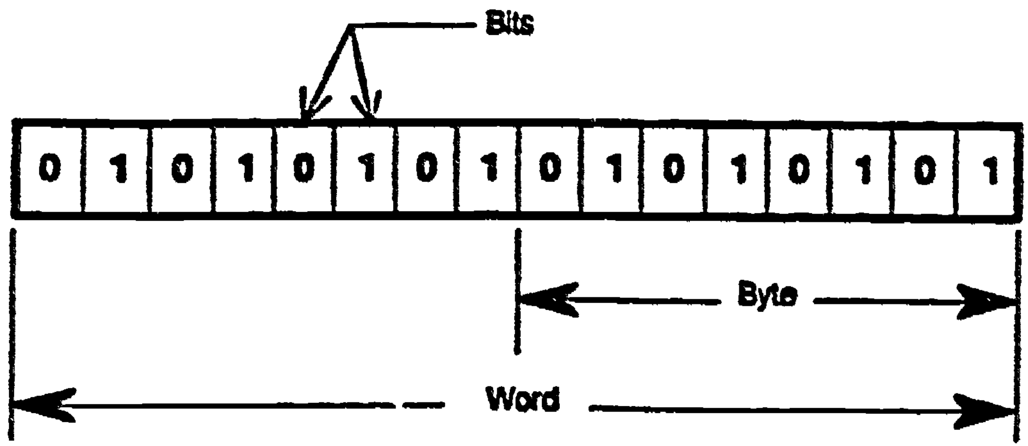
Convert the following binary numbers to decimal numbers.

	128	64	32	16	8	4	2	1		
1)	0	0	0	1	0	0	0	1	=	________
2)	0	1	1	1	0	1	1	1	=	________
3)	1	0	1	0	1	0	1	0	=	________

For Extra Credit:

0 1 0 0 0 0 0 1 0 0 0 1 = ________

BITS, BYTES, AND WORDS



BIT — a single Binary digIT

BYTE — 8 bits

WORD — 16 bits or 32 generally

*The number of bits in a word is actually dependent on the specific equipment.

BINARY NUMBERING SYSTEM

BINARY 16 BIT WORD MAX NUMBER 65536

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

BIT

NIBBLE

BYTE

16 BIT WORD

NUMBERING SYSTEMS Continued

2. OCTAL

   - Usable valid values 0,1,2,3,4,5,6,7

   - No 8 or 9 is used in this system

   - Some PLC's use this system for I/O Addressing

3. DECIMAL

   - Usable valid values 0,1,2,3,4,5,6,7,8,9

   - Most common numbering system

NUMBERING SYSTEMS Continued

4. HEXADECIMAL

- Usable valid values 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

- Used when programming computers

- EPROMS store memory with this numbering system

5. BCD

- Used on thumb wheels

- Some PLC's use BCD math on I/O system

BCD NUMBERING SYSTEM

BCD 16 BIT WORD   MAX NUMBER 9999

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4				3				2				1			

8 BIT WORD

16 BIT WORD

ASCII CONVERSION TABLE

Character	Decimal	Hexadecimal	Character	Decimal	Hexadecimal	Character	Decimal	Hexadecimal
NUL	0	00	0	48	30	[	91	5B
SOH	1	01	1	49	31	\	92	5C
STX	2	02	2	50	32	]	93	5D
ETX	3	03	3	51	33	^	94	5E
EOT	4	04	4	52	34	_	95	5F
ENQ	5	05	5	53	35	`	96	60
ACK	6	06	6	54	36	a	97	61
BEL	7	07	7	55	37	b	98	62
BS	8	08	8	56	38	c	99	63
HT	9	09	9	57	39	d	100	64
LF	10	0A	:	58	3A	e	101	65
VT	11	0B	;	59	3B	f	102	66
FF	12	0C	<	60	3C	g	103	67
CR	13	0D	=	61	3D	h	104	68
SO	14	0E	>	62	3E	i	105	69
SI	15	0F	?	63	3F	j	106	6A
DLE	16	10	@	64	40	k	107	6B
DC1	17	11	A	65	41	l	108	6C
DC2	18	12	B	66	42	m	109	6D
DC3	19	13	C	67	43	n	110	6E
DC4	20	14	D	68	44	o	111	6F
NAK	21	15	E	69	45	p	112	70
SYN	22	16	F	70	46	q	113	71
ETB	23	17	G	71	47	r	114	72
CAN	24	18	H	72	48	s	115	73
EM	25	19	I	73	49	t	116	74
SUB	26	1A	J	74	4A	u	117	75
ESC	27	1B	K	75	4B	v	118	76
FS	28	1C	L	76	4C	w	119	77
GS	29	1D	M	77	4D	x	120	78
RS	30	1E	N	78	4E	y	121	79
US	31	1F	O	79	4F		122	7A
SP	32	20	P	80	50	{	123	7B
!	33	21	Q	81	51	¦	124	7C
"	34	22	R	82	52	}	125	7D
#	35	23	S	83	53	~	126	7E
$	36	24	T	84	54	DEL	127	7F
%	37	25	U	85	55			
&	38	26	V	86	56			
'	39	27	W	87	57			
(	40	28	X	88	58			
)	41	29	Y	89	59			
*	42	2A	Z	90	5A			
+	43	2B						
,	44	2C						
-	45	2D						
.	46	2E						
/	47	2F						

CONVERSIONS

- Binary to Decimal
- Octal to Decimal
- Hexadecimal to Decimal
- ASCII Chart

WHY HEXADECIMAL?

Hexadecimal numbers are easier to use and handle than Binary Numbers, at least in terms of length. Computers/software understands Hex numbers.

DECIMAL	HEX DIGIT
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

CONVERTING HEXADECIMAL TO DECIMAL

Just as with decimal numbers, we build multiple-digit Hex numbers by adding more digits on the left. Suppose, for example, we add the number 1 to the largest single-digit decimal number 9. The result is a two digit number 10. What happens when we add 1 to the largest single-digit Hex number, F ? We get 10 again. The number 10 in Hex in decimal is 16; not 10. We know that 10h is 16, but how do we convert a larger Hex number, such as D3h, to a decimal number without counting up to D3h from 10h? Or, how do we convert the decimal number 173 to Hex?

What does the number 276 mean? In grade school, we learned that 276 means we have two hundreds, seven tens, and six ones, or:

2	x	100	=	200
7	x	10	=	70
6	x	1	=	6
276			=	276

We can use this same graphical method on Hexadecimal numbers. Consider the number D3h we mentioned earlier. D is the Hexadecimal digit 13, and there are 16 Hex digits versus 10 for decimal; so, D3h is thirteen sixteens and three ones, or:

D	===>	13	x	16	=	208
3	===>	3	x	1	=	3
D3h					=	211

For the decimal number 276, we multiplied the digits by 100, 10, and 1. For the Hex number D3, we multiplied the digits by 16 and 1. If we had four decimal digits, we would multiply by 1000, 100, 10, and 1. Which four numbers would we use with four Hex digits? For decimal, the numbers 1000, 100, 10, and 1 are all powers of 10:

$$10^3 = 1000$$

$$10^2 = 100$$

$$10^1 = 10$$

$$10^0 = 1$$

We can use exactly the same method for Hex digits, but with powers of 16, instead of 10. Our four numbers are:

$$16^3 = 4096$$

$$16^2 = 256$$

$$16^1 = 16$$

$$16^0 = 1$$

Let's convert some Hex numbers to decimals.

7C:

7	===> 7	x	16	=	112
C	===> 12	x	1	=	12
7Ch				=	124

3F9h:

3	===> 3	x	256	=	768
F	===> 15	x	16	=	240
9	===> 9	x	1	=	9
3F9h				=	1,017

AF1Ch:

A	===> 10	x	4,096	=	40,960
F	===> 15	x	256	=	3,840
1	===> 1	x	16	=	16
C	===> 12	x	1	=	12
AF1Ch				=	44,828

3B8D2h:

3	===> 3	x	65,536	=	196,608
B	===> 11	x	4,096	=	45,056
8	===> 8	x	256	=	2,048
D	===> 13	x	16	=	208
2	===> 2	x	1	=	2
3B8D2h				=	243,922

CONVERTING DECIMAL TO HEXADECIMAL

Like grade school math, we will start by recalling what we learned in division. We would divide 9 by 2 to get 4 with a remainder of 1. We will use the remainder to convert decimal numbers to Hex. Let's see what happens when we repeatedly divide a decimal number; in this case, 493, by 10:

$$\frac{493}{10} = 49 \quad \text{remainder} \quad 3$$

$$\frac{49}{10} = 4 \quad \text{remainder} \quad 9$$

$$\frac{4}{10} = 10 \quad \text{remainder} \quad 4$$

4 9 3

The digits of 493 appear as the remainder in reverse order -- that is, starting with right most digit C3.

We saw in the last section that all we needed for our Hex to decimal conversion was to replace powers of 10 with powers of 16.

1,069:

$\frac{1,069}{16} = 66$ remainder 13

$\frac{66}{16} = 4$ remainder 2

$\frac{4}{16} = 0$ remainder 4

4 2 D h

57,109:

$\frac{57,109}{16} = 3,569$ remainder 5

$\frac{3,569}{16} = 223$ remainder 1

$\frac{223}{16} = 1$ remainder 15

$\frac{13}{16} = 1$ remainder 13

D F 1 5 h

CONVERT HEXADECIMAL TO DECIMAL

8C =

3D8 =

CD1A =

3B102 =

CONVERT DECIMAL TO HEXADECIMAL

504 =

1,362 =

47,392 =

0100	11C1	0001	1100
4	D	1	A

WORD

LOGIC CONCEPTS

BOOLEAN ALGEBRA

- 1849, George Boole, England
  - Tool to aid in the logic of reasoning(philosophy)
  - Simple way of writing complicated combinations of " logic statements " that can be true or false
- Two-Valve Concept
  - True = 1
  - False = ϕ
- All digital systems based on this true/false or two-valued concept

BOOLEAN ALGEBRA

RULES

$A + B = B + A$ — *COMMUTATIVE LAWS*

$AB = BA$

$A + (B + C) = (A + B) + C$ — *ASSOCIATIVE LAWS*

$A(BC) = (AB)C$

$A(B + C) = AB + AC$ — *DISTRIBUTIVE LAWS*

$A + BC = (A + B)(A + C)$

$A(A + B) = A + AB = A$ — *LAW OF ABSORPTION*

$\overline{(A + B)} = \overline{A}\ \overline{B}$ — *DE MORGAN'S LAWS*

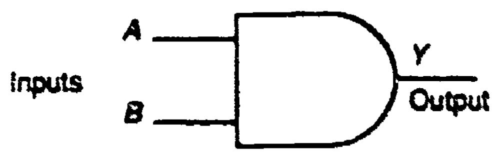
$\overline{(AB)} = \overline{A} + \overline{B}$

$\overline{\overline{A}} = A\ ,\ \overline{1} = 0\ ,\ \overline{0} = 1$

$A + \overline{A}B = A + B$

$AB + AC + B\overline{C} = AC + B\overline{C}$

AND

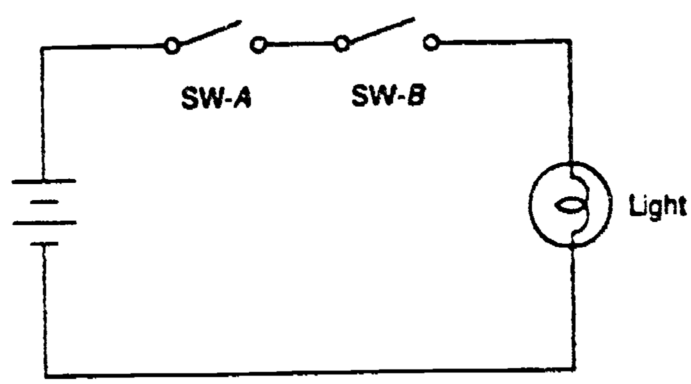


Two-input
AND gate
symbol

Inputs		Output
A	*B*	*Y*
0	0	0
0	1	0
1	0	0
1	1	1

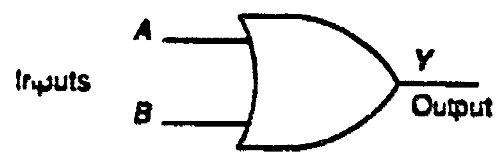
AND truth table

Circuit



Boolean Equation: AB = Y

OR

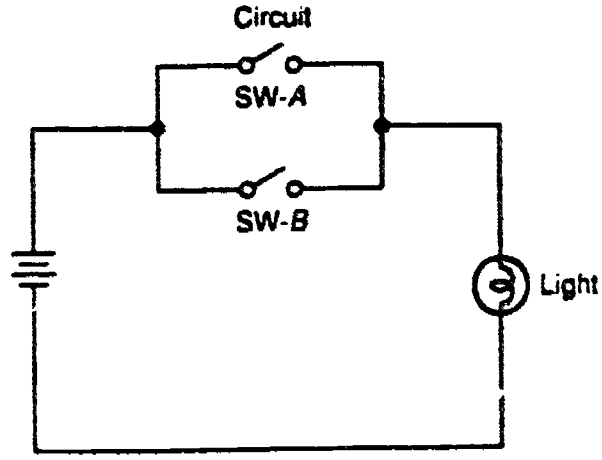


Two-input
OR gate
symbol

Inputs		Output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

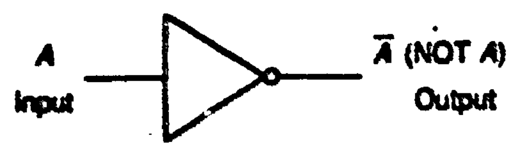
OR truth table

Circuit



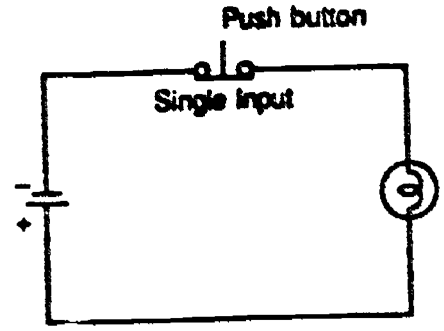
Boolean Equation: $A + B = Y$

NOT



A	NOT A
0	1
1	0

NOT truth table



Boolean Equation: A = Y

BASIC RELAY INSTRUCTIONS

CONTACTS - Represent conditions to be evaluated

Examine ON —| |—

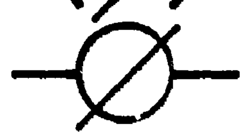
Examine OFF —|/|—

COILS

- Represent outputs
- Controlled by evaluation of conditions

OUTPUT —( )—

Not OUTPUT —(/)—

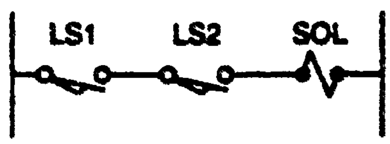


Latch —(L)—

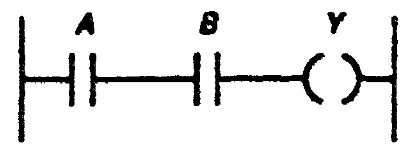
Unlatch —(U)—

LOGIC CONCEPTS ILLUSTRATIONS

Relay ladder diagram

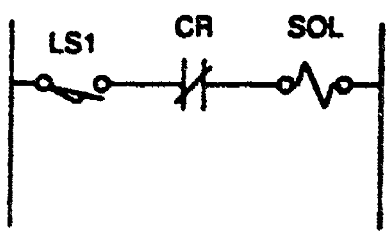


Logic ladder diagram program

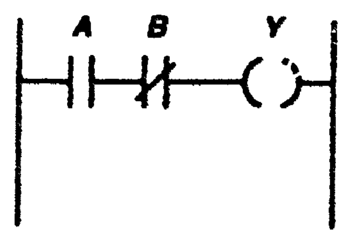


Boolean equation: $AB = Y$

Relay ladder diagram

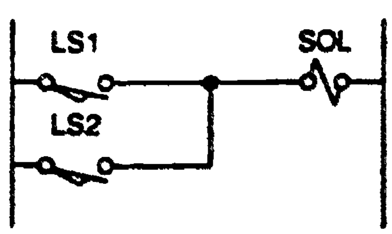


Logic ladder diagram program

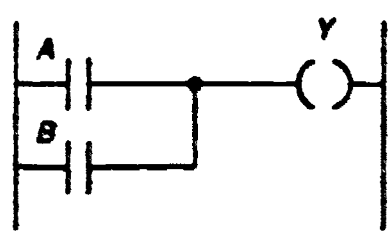


Boolean equation: $A\bar{B} = Y$

Relay ladder diagram



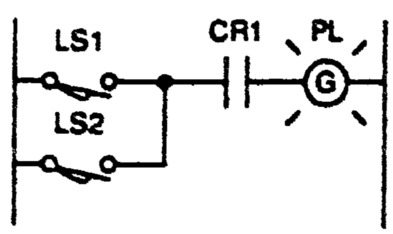
Logic ladder diagram program



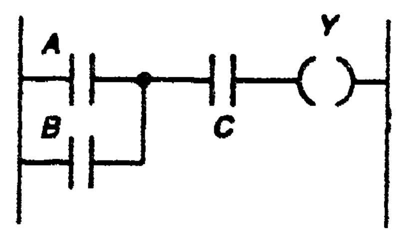
Boolean equation: $A + B = Y$

LOGIC CONCEPTS ILLUSTRATONS
Continued

Relay ladder diagram

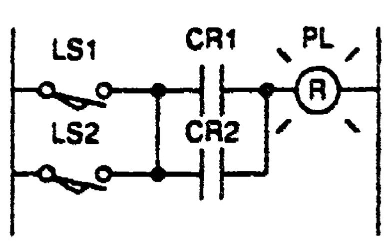


Logic ladder diagram program

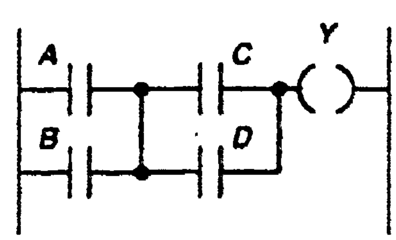


Boolean equation: $(A + B)\ C = Y$

Relay ladder diagram



Logic ladder diagram program



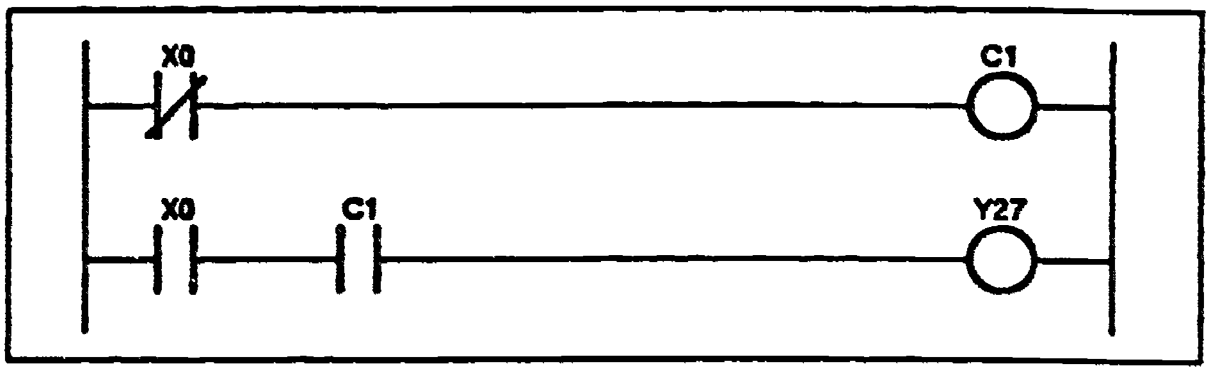
Boolean equation: $(A + B)\ (C + D) = Y$

EXERCISE

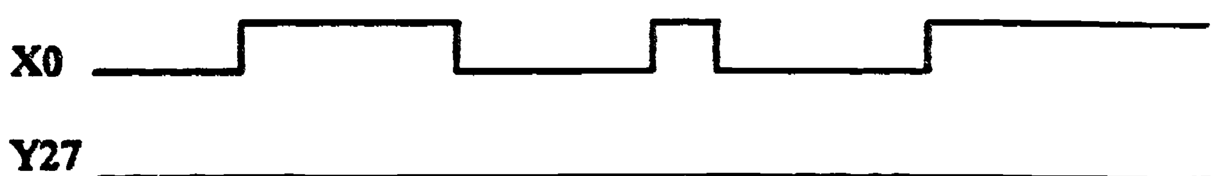
Effect of RLL Sequence In PLCs

This logic scanning sequence can work to your advantage or disadvantage.

Knowing all this, consider the following groups of RLL rungs that have been entered into a PLC. They are identical, except for the order that they are placed into the PLC.

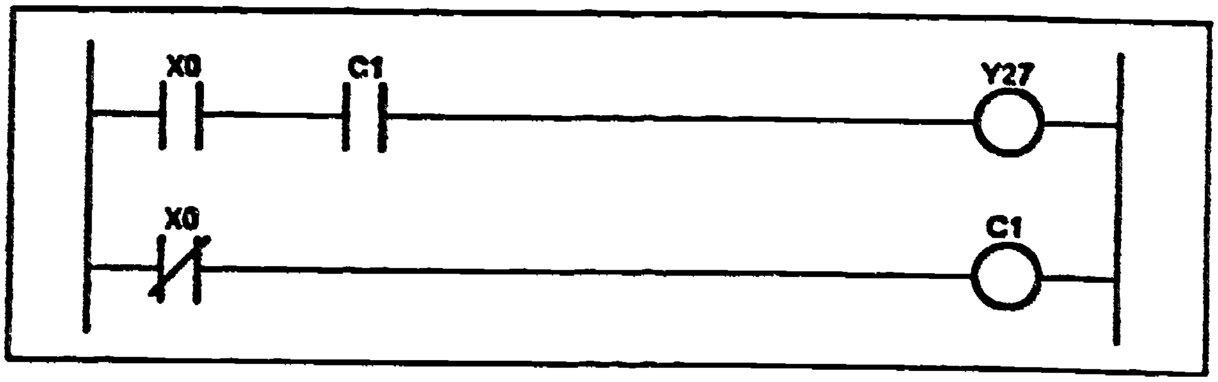


If input X0 opens and closes several times, what would be the action of output Y27?

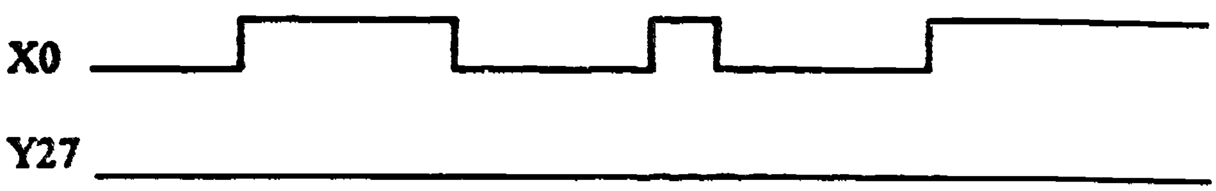


EXERCISE (continued)

Now try this one, the same rungs, but in a different order.



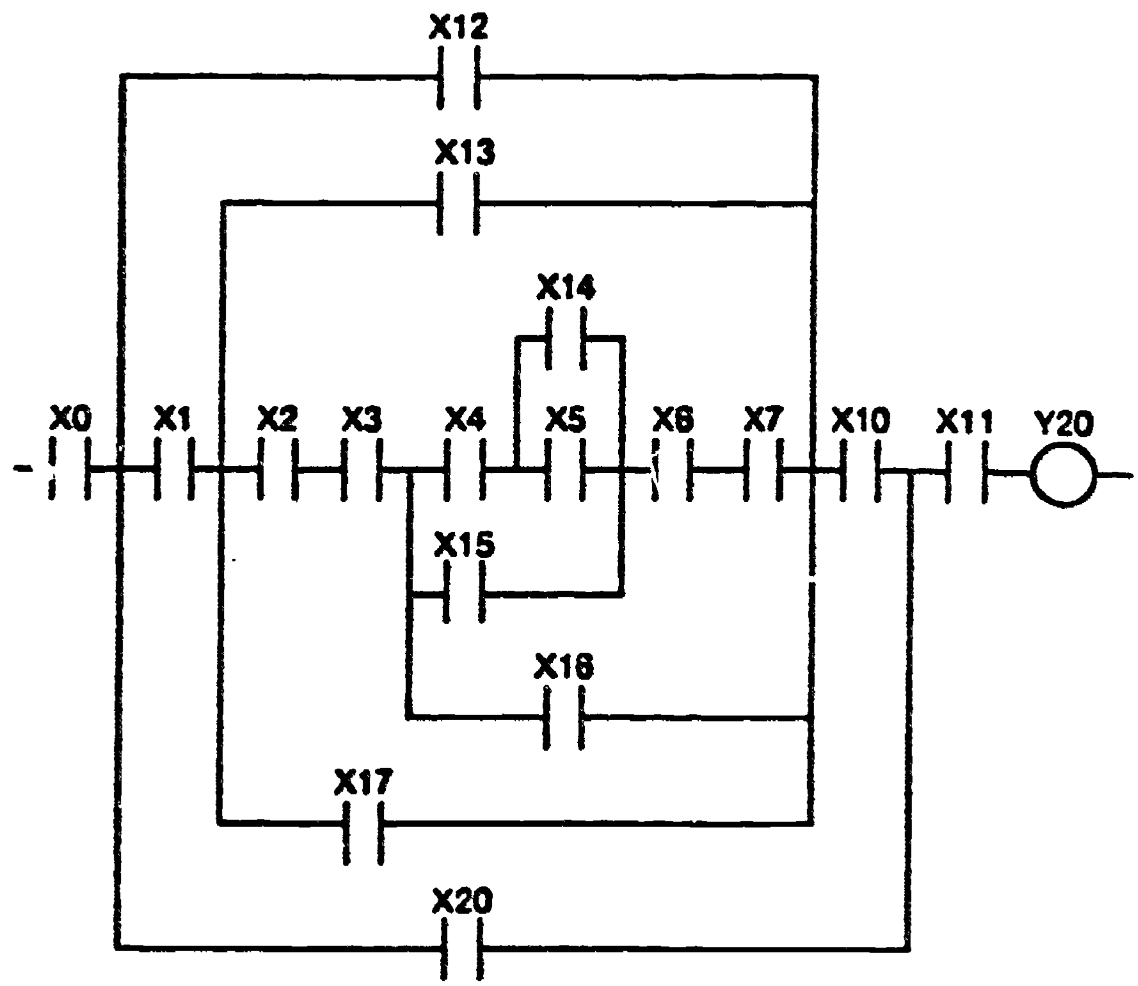
If input X0 opens and closes several times, what would be the action of output Y27?



EXERCISE

Extra Credit

Try the following complex interlock circuit for extra credit.



Extra Credit Exercise: For Wizards Only

If you are the first lab team to successfully write the code, yell out loud, "We are the best lab group, and the rest of you are losers!"

ENTERING A LADDER DIAGRAM

ENTERING A LADDER DIAGRAM

Entering Contacts

To enter/edit a contact, position the cursor in the input columns. To make the entry, use the contact function keys for the element type. Key in the element identifier with the numeric keys. If you decide not to complete a change, you can press ☐ ABORT-F1 to restore the original ladder element.

For relational contacts (less-than, greater-than, equal-to, and not-equal-to), use the arrow keys to position the cursor in the fields for making parameter entries.

After completing the entry, press ☐ RETURN to write the contact to the network.

Entering Coils

To enter/edit a coil, press ☐ -( )- -F6. Select the coil type with the function keys, or key in the mnemonic for the coil type and press ☐ RETURN. After the coil is displayed in the output column, key in the parameter(s). For coils specifying a range of elements (SET, RST, etc.), use the arrow keys to position the cursor. If you decide not to complete a change, press ☐ ABORT-F1 to restore the original ladder element.

After completing the entry, press ☐ RETURN to write the coil to the network.

Entering Boxes

To enter/edit a box, press ☐ BOX-F7 (or the B hard key). If you do not know the mnemonic for the box type, display the HELP screen for a complete listing. Key in the mnemonic for the box type and press ☐ RETURN.. After the box is displayed in the output column, key in the parameters. For boxes, such as timers, with parameters inside the box, use the arrow keys to move from field to field inside the box. If you decide not to complete a change, you can press ☐ ABORT-F1 to restore the original ladder element.

The display format of some box instructions changes when you are in edit mode. The ISG, SG, LBL, SBR, INT, and DLBL boxes are displayed on the left of the screen until you press the EDIT key. They are then displayed as single input, single parameter, output boxes on the right side of the screen.

After completing the entry, press ☐ RETURN to write the box to the network.

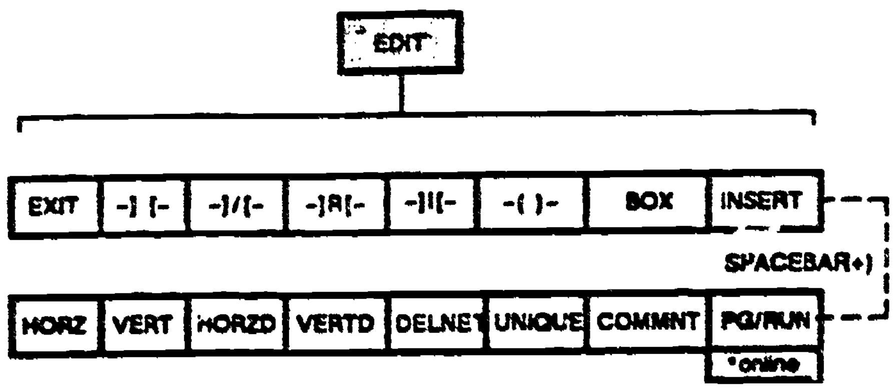
Correcting Entries

If you make an incorrect entry at a prompt line, key in characters until the field is overflowed. When the overflow occurs, the field resets.

ENTERING A LADDER DIAGRAM

Invoking Edit Functions

To enter/edit a program, first press ☐ EDIT–F2 from the Ladder Display. Figure 6-1 illustrates the function key line available after you select edit mode. Use the guidelines and procedures given in this chapter to enter a program for the first time or to make changes to an existing program.



Entering/Editing Elements

The term *element* is used in TISOFT programming to designate a component used in your program. For example, a coil, a timer, and a counter are all *elements*. Each element requires an *identifier*, which is the number that distinguishes one element of a particular type from another, such as X1, X2, etc.

Making TISOFT Entries

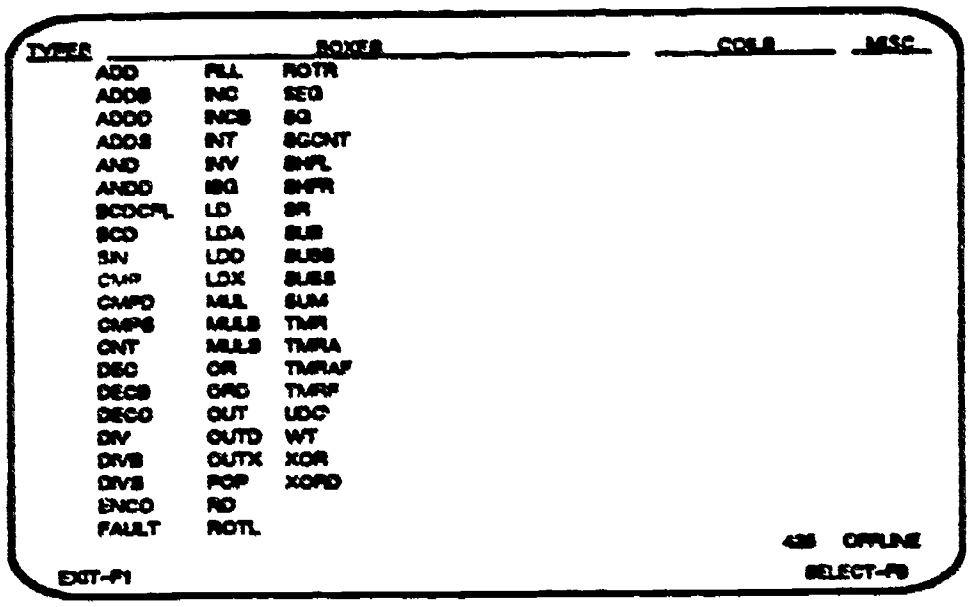
To accommodate your programming preferences, TISOFT provides different methods of making entries. You can use the function keys and step through from element type (as contact) to type of contact (as I), then key in the identifier.

Alternately, you can position your cursor at the point you want to add an element and key in an identifier (such as X2). The contact is then automatically drawn with X2 as the identifier.

ENTERING A LADDER DIAGRAM

Using the HELP Screen

You can access a display listing the valid entries for a field at any point with a prompt requesting an entry. For example, at the prompt for entering boxes, the HELP display gives a listing of valid box types. Figure 6-2 shows the display accessible after selecting BOX when entering a program. Invoke the HELP display by pressing *SHIFT ?*.



Use the arrow keys to position the cursor on the box type you want displayed on the prompt line, and press SELECT-F8. The prompt line reappears with your box type displayed.

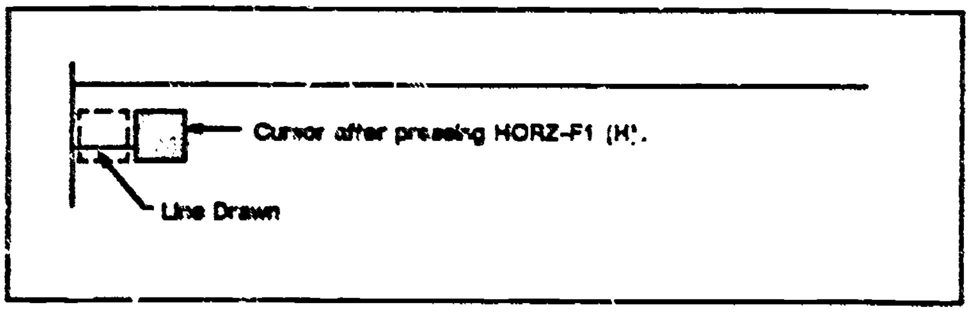
ENTERING A LADDER DIAGRAM

Drawing Lines for Complex Networks

Use HORZ-F1 and VERT-F2 to add the connecting lines when you are entering a complex network. You cannot draw a horizontal line in the last column of a row. With the vertical line key, you cannot draw a line from the first column, on the last row, or on the top of a box.

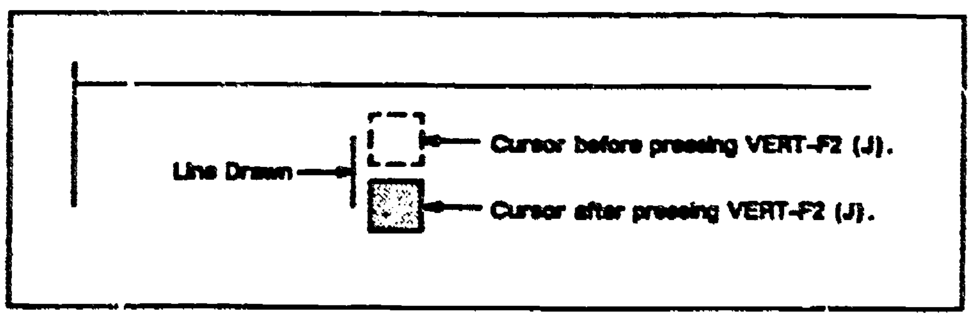
Drawing Horizontal Lines

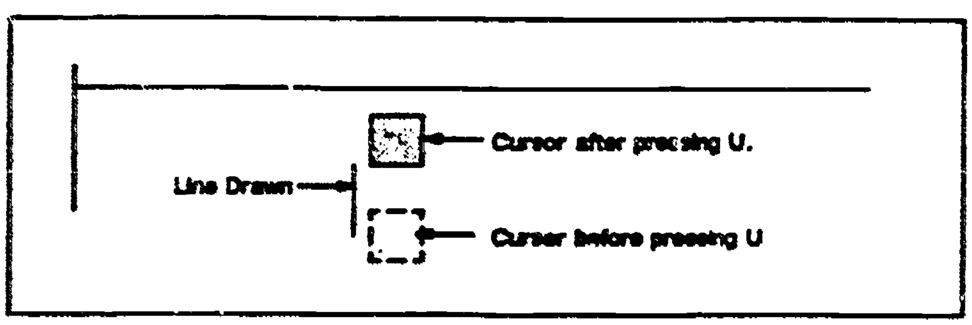
Use HORZ-F1 (or H hard key) for drawing horizontal lines. The drawing below shows positioning of the cursor before and after the line is drawn.



Drawing Vertical Lines

You have two options for drawing vertical lines. You can use VERT-F2 or the U hard key. The illustrations below show the line drawn with each of the keys.





ENTERING A LADDER DIAGRAM

Editing Programs

Accessing edit functions to make changes to your program is as simple as first entering the program. You only add the step of finding the point in the program you wish to edit.

First, position the cursor at the point where you want changes. Use the FIND operation to locate the ladder address or element for you.

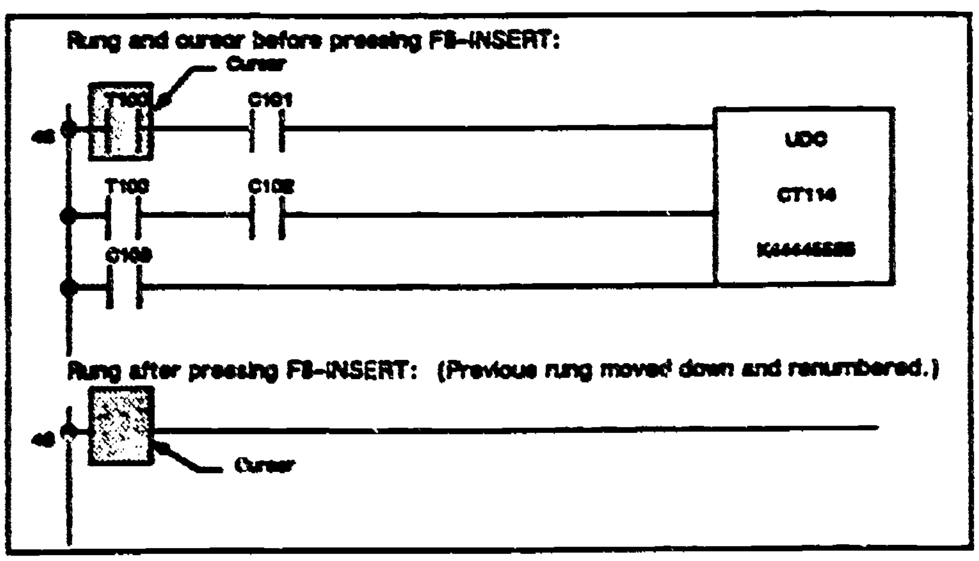
After locating the element(s) you wish to change, access edit mode by pressing ❑ EDIT–F2 from the ladder display menu.

Saving Edits

Since off–line and on-line edits are performed in the programming device RAM, it is advisable to save changes often to your program disk (off-line) or the controller (on-line). To do this, press ❑ WRITDK-F8 (off-line) or ❑ WRITPC-F8 (on-line) from the ladder display menu.

Inserting Rungs

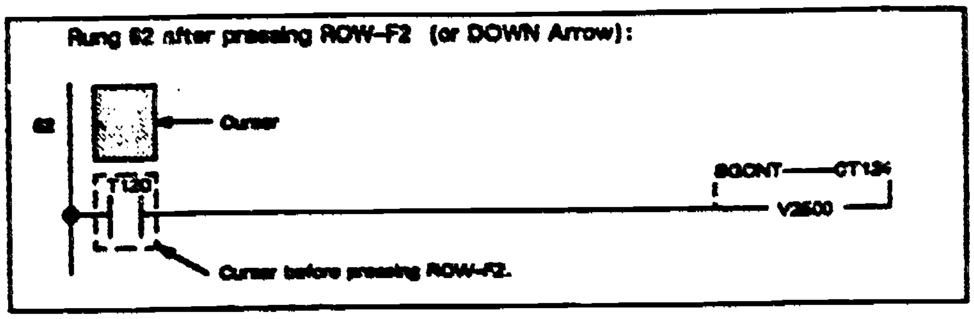
To insert a blank rung in a display, use INSERT-F8 available in the edit mode function keys. The drawing below illustrates the new rung displayed after pressing INSERT.

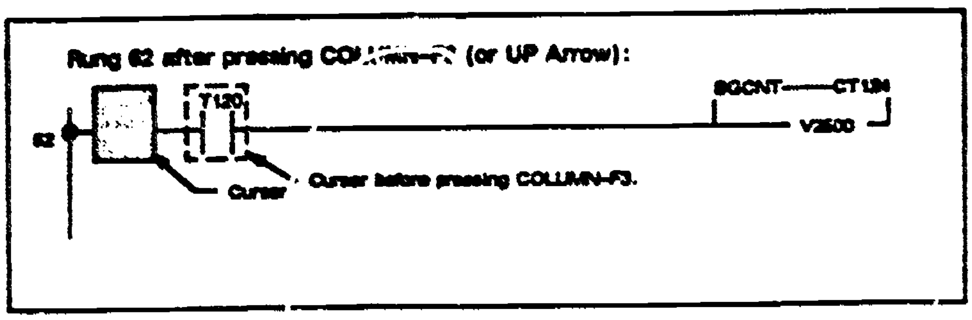


Inserting Elements

To insert an element in an existing network, use the insert hard key. After you press the insert hard key, you have available RO V-F2 and COLUMN-F3 to insert space for elements. The illustrations below show an example of how the inserts are made.

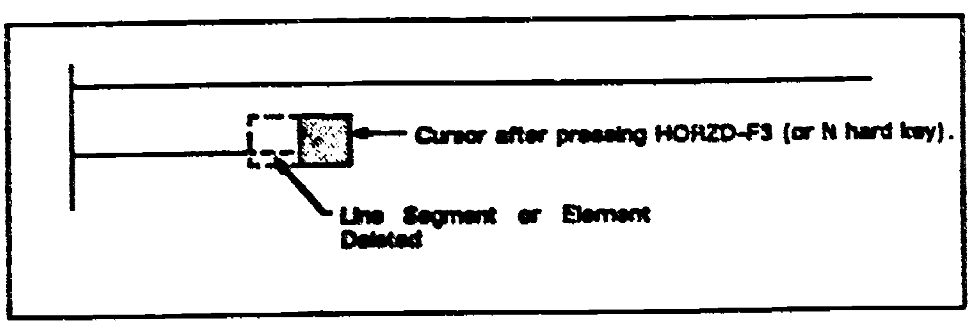
ENTERING A LADDER DIAGRAM

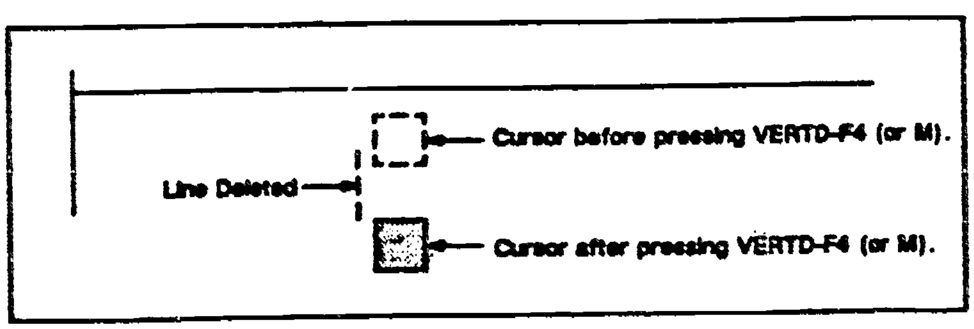




Deleting Lines and Elements

To delete lines and elements from your ladder program, use the HORZD-F3 and VERTD-F4 keys on the edit function key line. The figures below illustrate horizontal and vertical line deletes. If you are deleting an element, place the cursor on the element and press ☐ HORZD-F3. To delete an entire network, position the cursor on the network, and press ☐ DELNET-F5.





BASIC PRINCIPLES OF OPERATION

PROCESSORS, POWER SUPPLY SYSTEM, AND PROGRAMMING DEVICES

Processor

CPU

The "Brains" of the system.

Consists of:

Microprocessor
User memory
Power supply
Communication Interface
I/O Interface

Central Processing Unit (CPU)

Decision Making	User Memory
	1. Decisions to be made 2. Process information

The Central Processing Unit has two major functional sections to a PLC user.

- *Decision Making Section* — Makes decisions based on the signals received from the inputs. These decisions are totally controlled by the user.
- *User Memory* — Section that stores two types of user information:

  1) Instructions necessary to make the decisions

  2) Process or application dependent information

BASIC PRINCIPLES OF OPERATION

Subsystems, Error Checking, Diagnostics

- **Subsystem Communication**
  - Data transfer at the end of each program scan
  - Distance between CPU and subsystem may reach 15,000 feet
  - Media may be twisted pair, coax or fiber
  - Normally incorporates error checking between processor and subsystem

- **Error Checking**
  - Monitor status of memory, communications and processor operation
  - Common methods: Parity (VRC)
Checksum (CRC, LRC)

- **CPU Diagnostics**
  - Processor performs error checks and signaling during its operation
  - Typical diagnostics are power supply OK, processor OK, memory OK, battery OK

BASIC PRINCIPLES OF OPERATION

The System Power Supply

- **Input Voltage**
  - Usually AC (120 or 220), could be DC (24 V)
  - Tolerates 10-15% variation in line conditions
  - Use constant voltage transformer for unstable line voltage
  - Use isolation transformers in high EMI areas

- **Loading Considerations**
  - Each power supply can provide a maximum current at a given voltage
  - Undercurrent situations result in unpredictable operation of the I/O system (module current requirements)
  - Overload conditions are usually a function of the combination of outputs that are ON at a given time

- **Provides power to the:** CPU
  Base
  Input Modules
  Output Modules
  Handheld Programmer

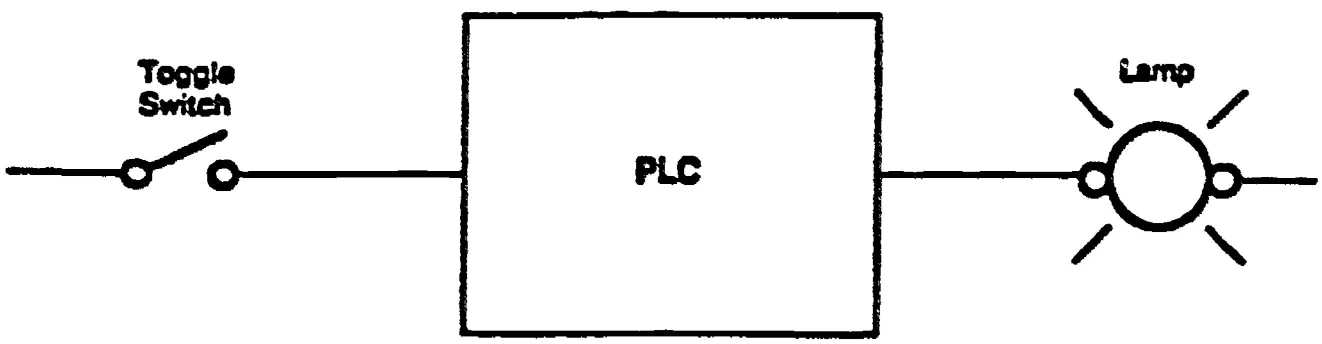
- **Care must be taken to size the power supply to handle the required modules**

TYPICAL COMPONENT POWER REQUIREMENTS

	Power Supplied (mA)
CPUs	
425 CPU with AC Power Supply[1] (includes 3.5K words of program RAM)	3700
435 CPU with AC Power Supply[1] (Requires memory cartridge)	3700
Expansion Units	
Expansion unit with AC power supply[1]	4000
	Power Required (mA)
Bases	
4 Slot Base with expansion	80
6 Slot Base with expansion	80
8 Slot Base with expansion	80
4 Slot Base without expansion	80
6 Slot Base without expansion	80
8 Slot Base without expansion	80
DC Input Modules	
24/48 VDC input module, 8 points (-/+ logic)	100
12/24 VDC input module, 16 points (- logic)	150
12/24 VDC/AC input module, 16 points	150
AC Input Modules	
110/220 VAC input module, 8 points	100
110 VAC input module, 16 points	150
DC Output Modules	
12/24 VDC output module, 8 points, 2A (- logic)	150
12/24 VDC output module, 16 points, 0.5A (- logic)	200
12/24 VDC output module, 16 points, 0.5A (+ logic)	350
AC Output Modules	
110/220 VAC output, 8 points, 2A	250
110/220 VAC output, 16 points, 0.5A	450
Relay Output Modules	
Relay output module, 8 points, 2A	550
Relay output module, 16 points, 1A	1000
Programming	
Machine Interface Unit (includes PROM burner) (963 mA to write EPROM)	320

1. These devices can also provide power to external devices (24 VDC @ 400 mA).

Simple Discrete Control

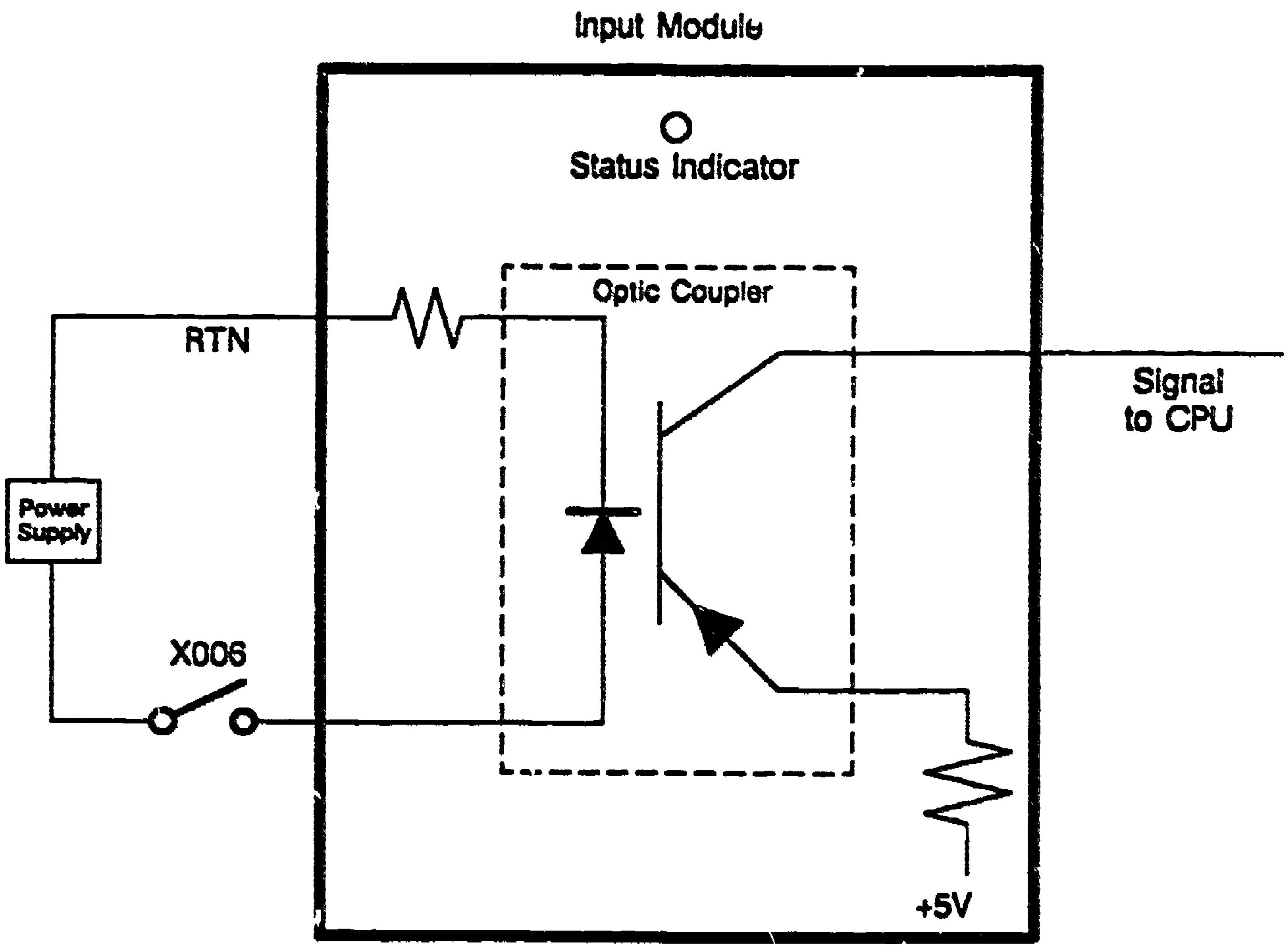


To understand the discrete control of a programmable controller, consider an example where a toggle switch controls a lamp, ON and OFF.

- When the toggle switch is closed, the lamp is ON.
- When the toggle switch is open, the lamp is OFF.

(Granted, a circuit this simple would not require a programmable controller, but this is for instructional purposes!)

Input Hardware



Consider when the toggle switch is open, an electrical circuit is open.

Based on hardware and software configuration, the signal wire attached to the PLC is designated "X" and a number.

Example: X006

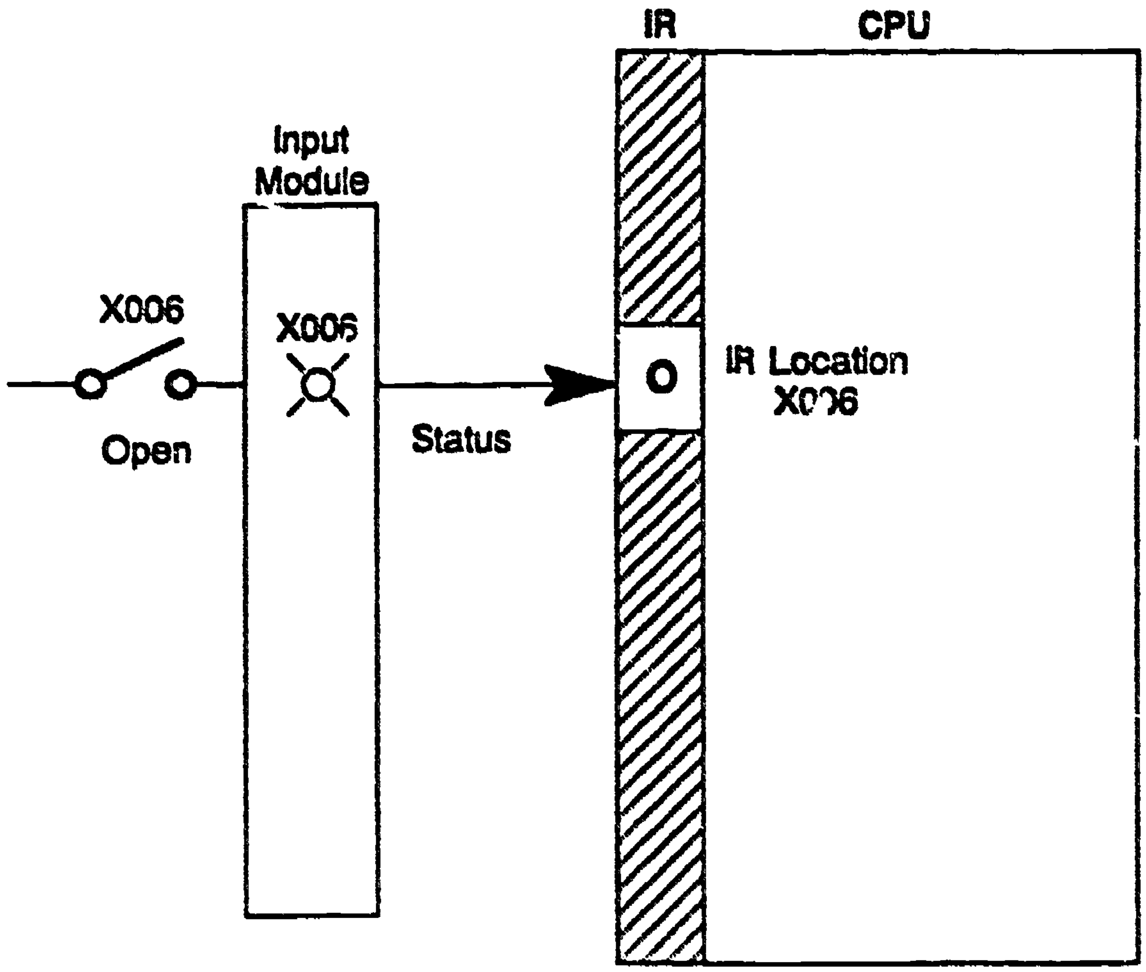
A sensor in the Input Module detects the open circuit and performs 2 actions:

1 Turns off a status lamp indicator on the module.

2 Changes the signal voltage levels from industrial strength (ex: 24 VDC, 230 VAC, etc.) to safe DC levels that the CPU can tolerate (ex: +5 VDC, etc.).

An Optic Coupler is used to isolate the CPU from hostile field voltages on the input wires.

The Signal Moves to CPU User Memory



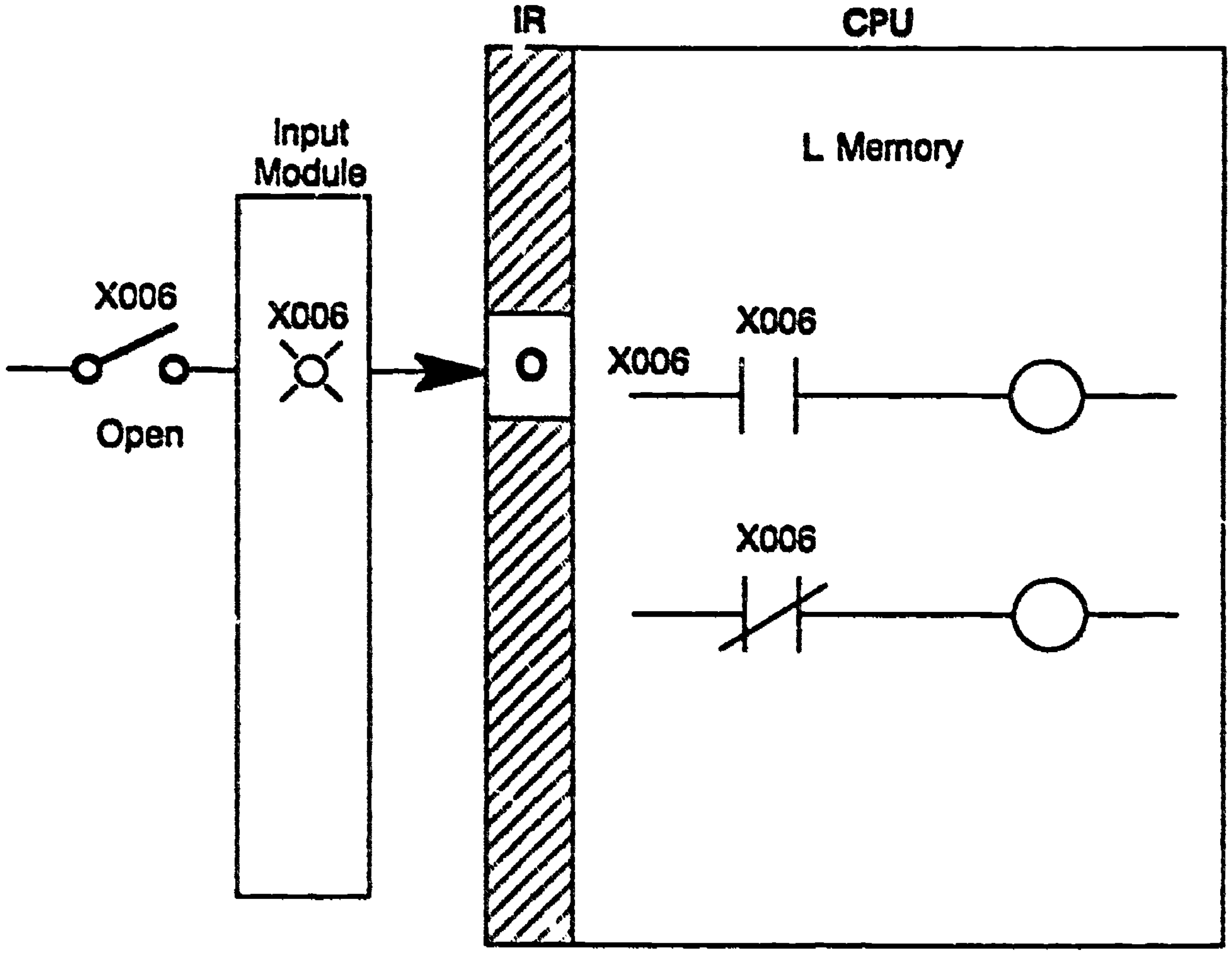
Since the input module has detected that the field device is open, the status of the device moves to a special memory location in the CPU that is dedicated to that input point.

No other input point can interfere with this one-bit memory location.

This memory is called **Image Register (IR)** in T.I. Programmable Controllers.

There is a one-bit memory location for every discrete input used in a TI PLC.

The Story Continues



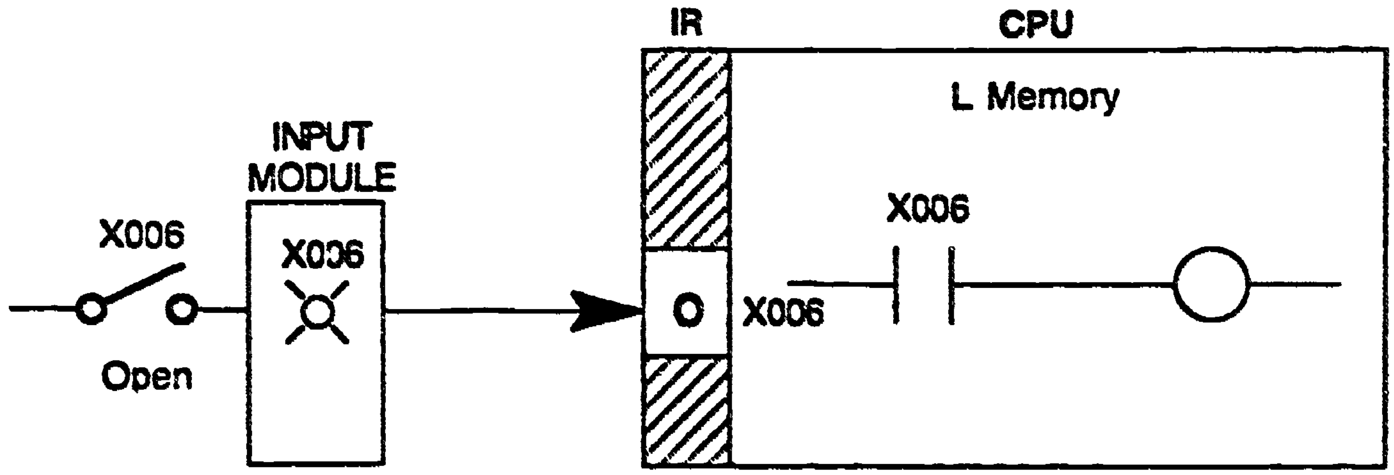
RLL (Relay Ladder Logic) is the language most often used in a PLC.

This language evolved from relay replacement applications for programmable controllers and was designed to work as much like the electromechanical devices as possible.

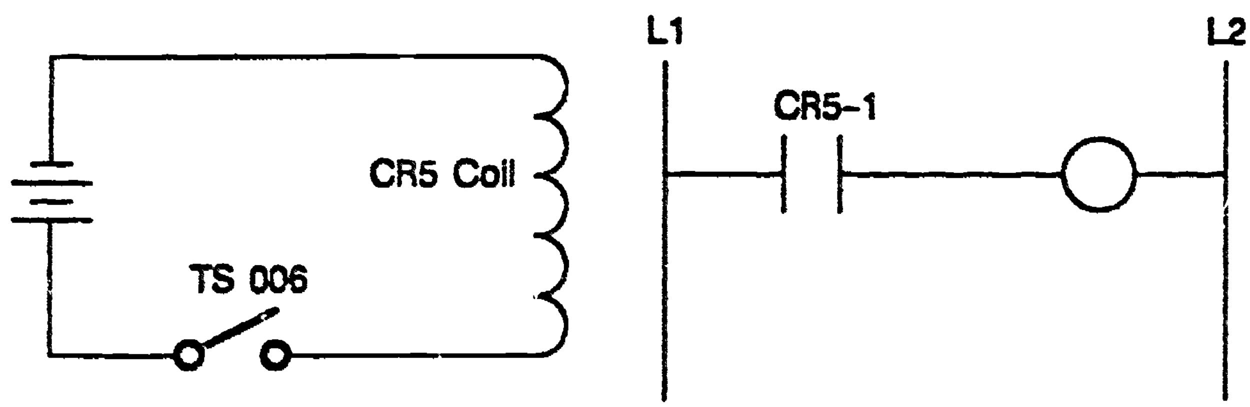
Symbols of N.O. (Normally Open) or N.C. (Normally Closed), used in RLL, are very close to standard detached IEEE electrical schematic symbols.

RLL resides in a special user memory called L Memory.

How Simple RLL Inputs Work



Electromechanical Analogy: IR Bit = Relay Coil



If the PLC user has an electromechanical background, a strong analogy can be made.

For example the N.O. contact can be designated in software as an "X006." This contact can be thought to open or close as in electromechanical relays. To open or close, a relay contact must have a coil.

It's easiest to think of the X006 IR bit as the coil of the relay.

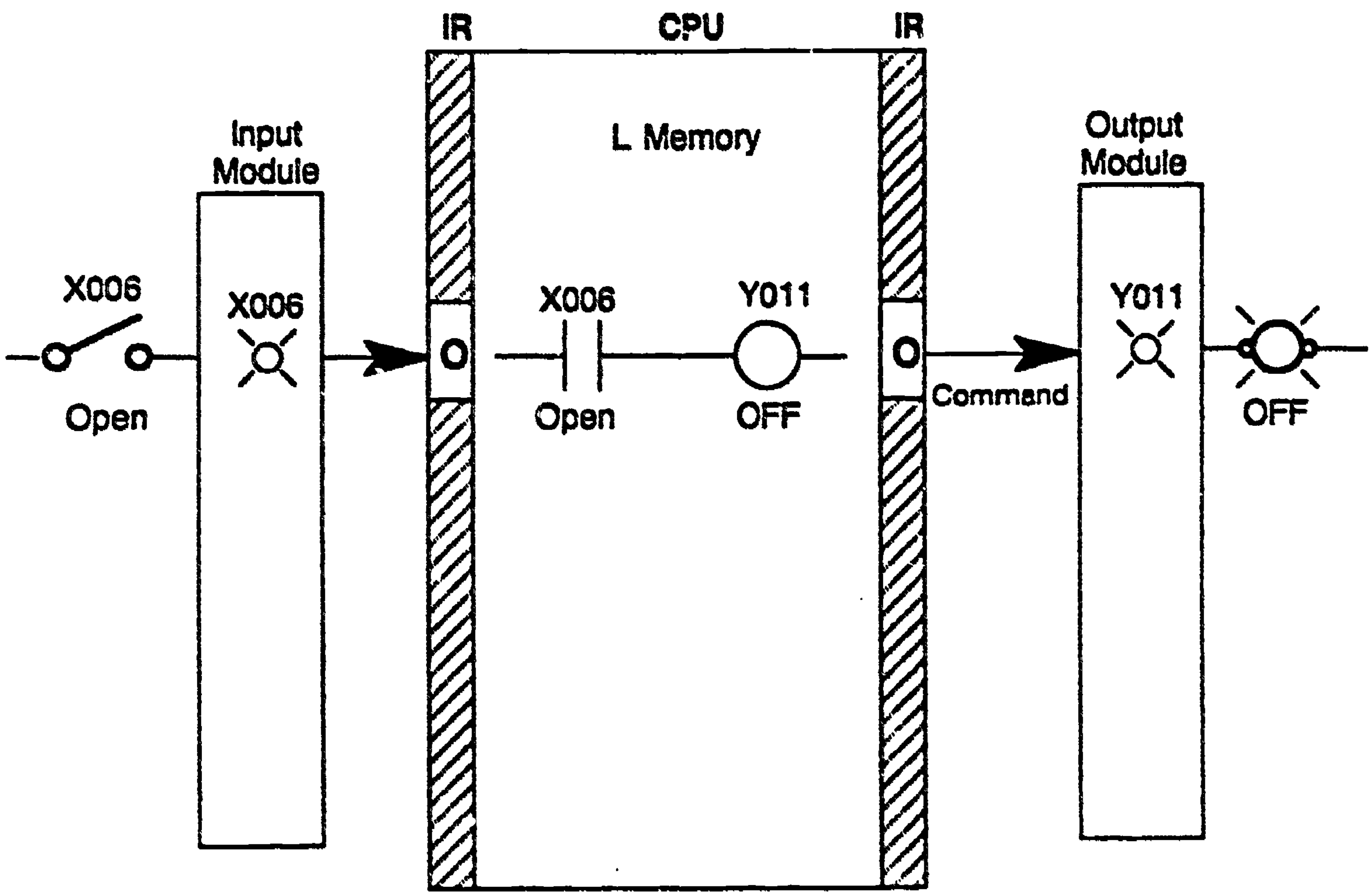
In this example, since:

The toggle switch is OPEN,

There is a "0" in the X006 IR bit

So the N.O. software contact can be said to be OPEN.

How Simple RLL Inputs Work (continued)



As with relays, if the control contact is open, the coil in software (Y011) is OFF or DOWN.

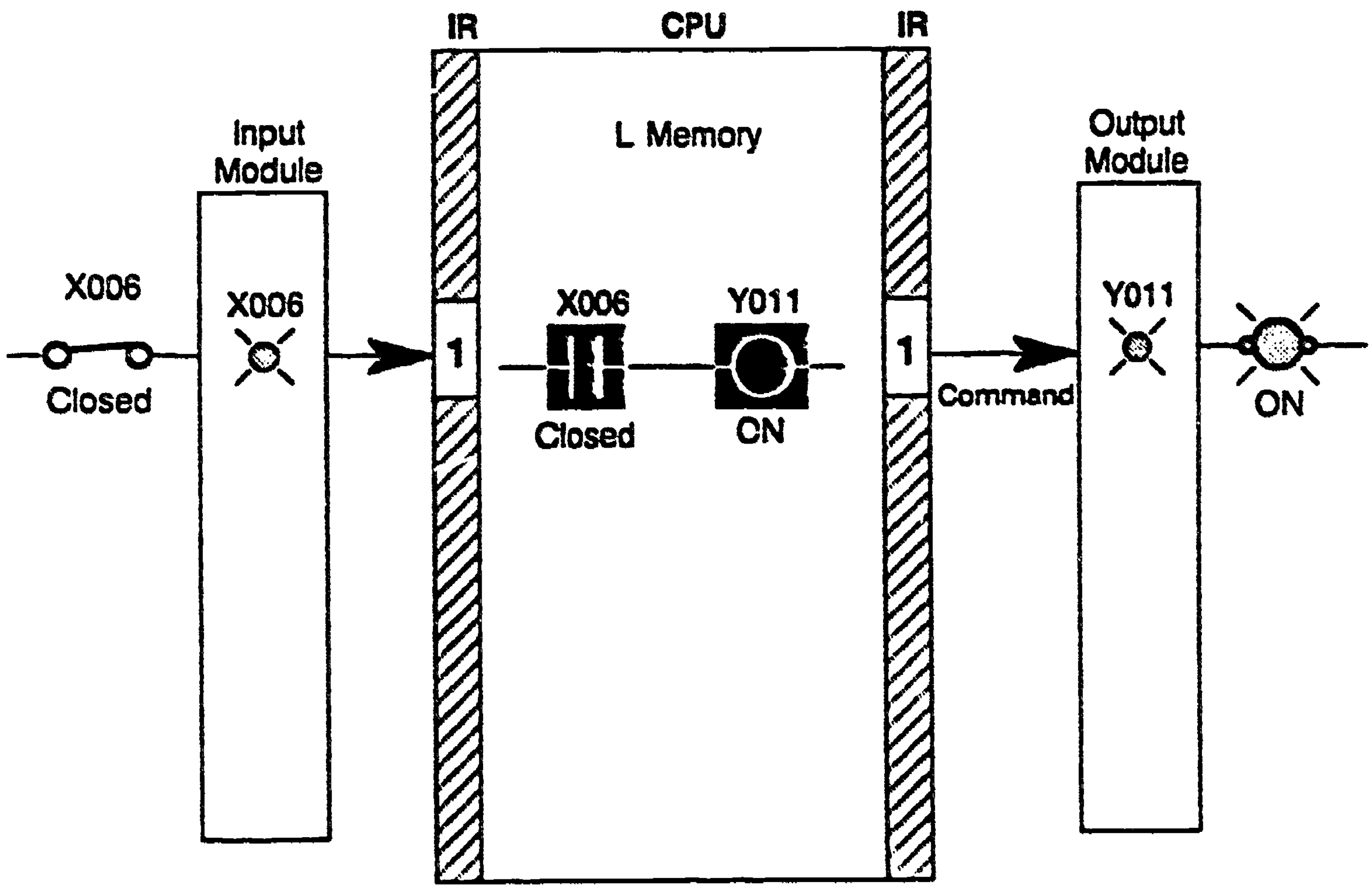
This is reflected again in an IR bit that is unique to the output of the PLC, Y011.

Once the output IR contains a bit, "0" or "1", this is a "command" to the output module hardware to switch through or not switch through a field potential to a load.

In this example, a "0" to Y011 commands the lamp OFF.

As with the input module, the output module isolates the hostile field voltage from the milder CPU logic voltage.

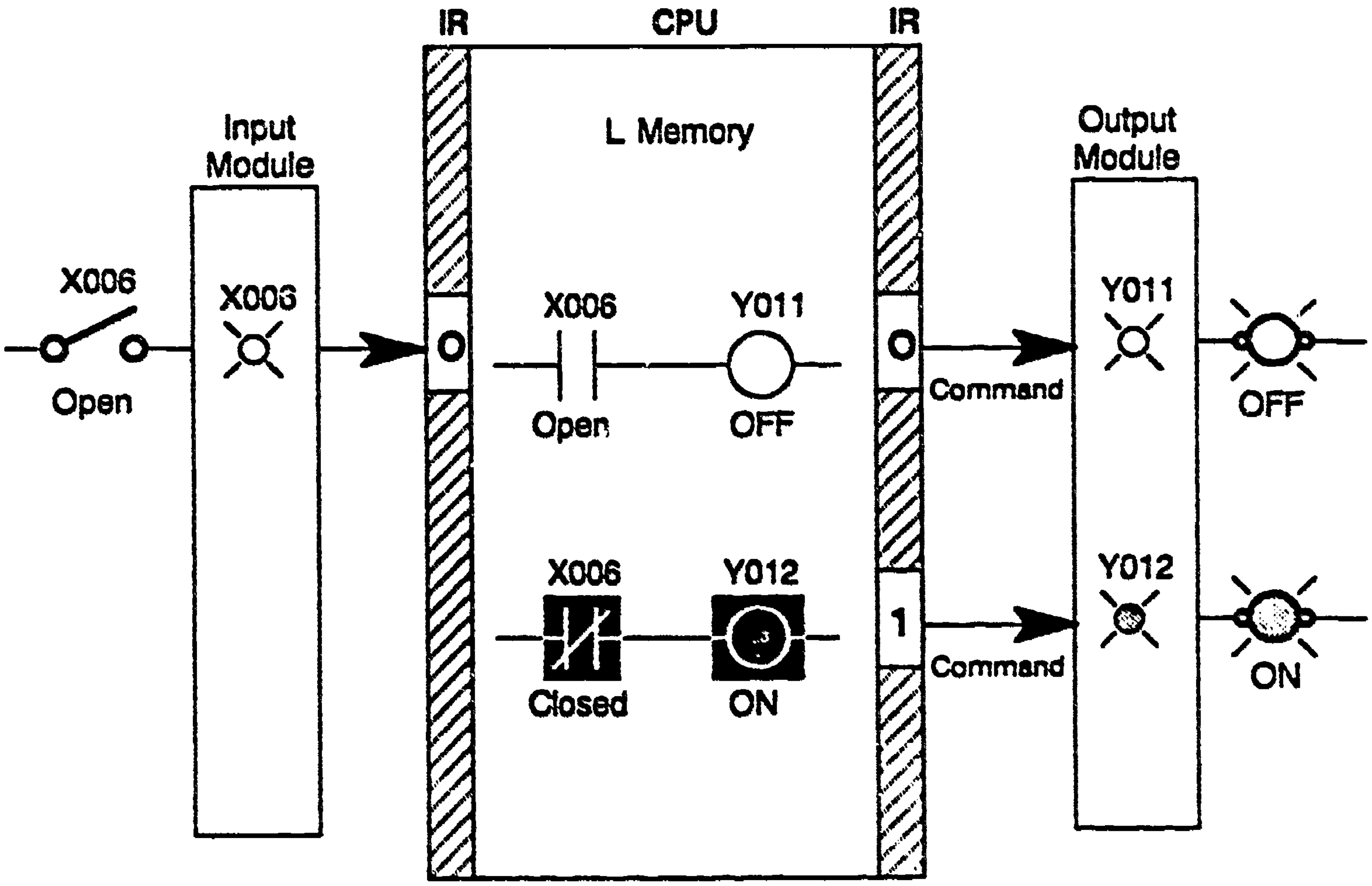
Try It Again



Consider the same signal flow with the toggle switch in the *closed* position.

NOTE:	Toggle Switch	= closed
	Input Indicator	= on
	X006 IR Bit	= 1
	X006 RLL N.O. Contact	= closed
	Y011 RLL Coil	= on
	Y011 IR Bit	= 1
	Output Indicator	= on
	Lamp	= on

A NOT in the Tale



Consider a scenario again with the toggle switch OPEN, but this time with some added software.

Another "rung" of ladder logic has been added. This addition uses the RLL version of an N.C. (Normally Closed) contact. The state of this N.C. contact, since it is still referred to as X006, is still dependent on the IR bit:

In this example, since:

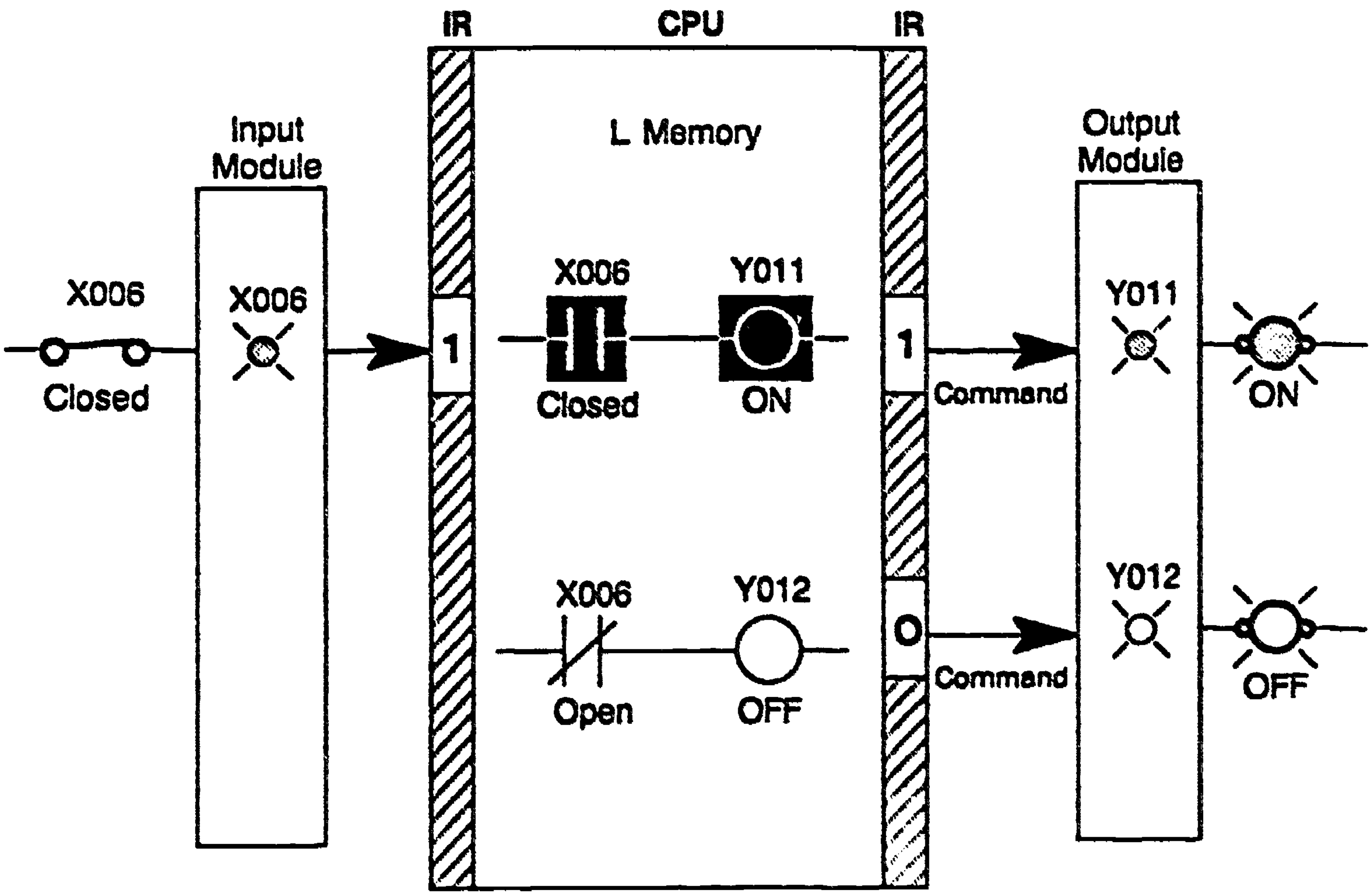
The toggle switch is OPEN, there is a "0" in the X006 IR bit

So the N.O. software contact is said to be OPEN, and Y011 = 0

So the N.C. software contact is said to be CLOSED, and Y012 = 1

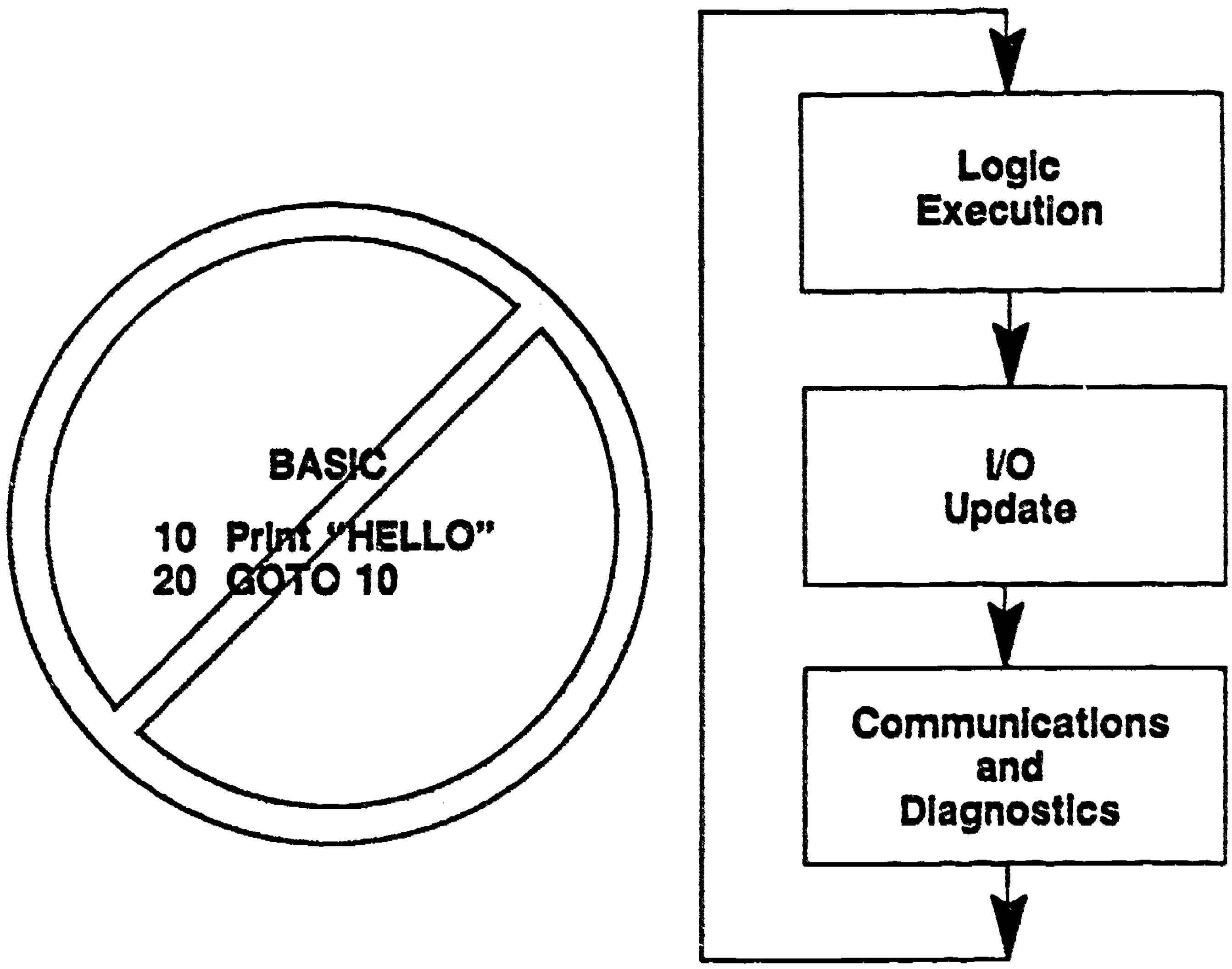
N.C. software contacts in RLL are often referred to as "NOTted" contacts, which means they will be in the *opposite* state of the image register.

Try It Again



Consider the same signal flow with the toggle switch in the CLOSED position.

Time Becomes a Dimension



Programmable Controllers often control expensive and sometimes dangerous applications.

Although an RLL program can be written incorrectly and cause damage to a process, PLC manufacturers engineer in a time management system that prevents software from "Getting Stuck In A Loop".

This system also detects almost any type of CPU hardware problem and causes the CPU to shut down to a SAFE STATE. (Discrete Outputs = OFF, Analog Outputs = 0.)

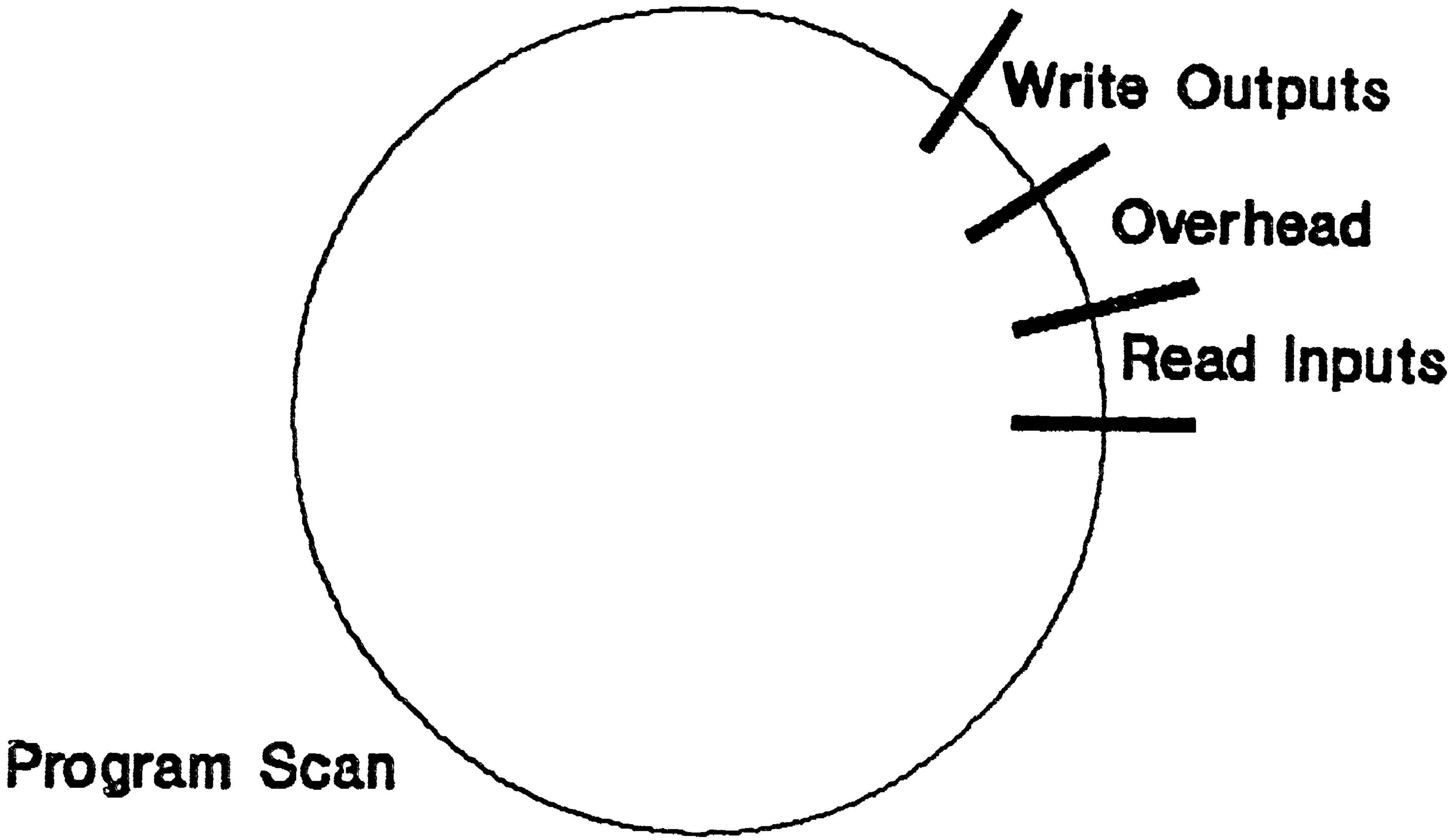
This time management system is called PLC Scan Time.

It is comprised of 3 major segments of time:

- **Logic Execution**
- **I/O Update**
- **Communications and Diagnostics**

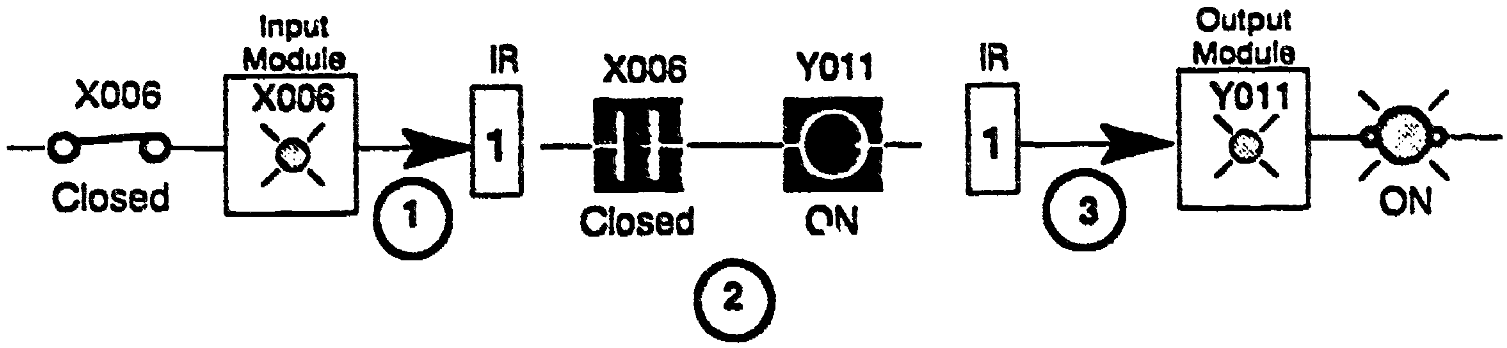
This cycle repeats.

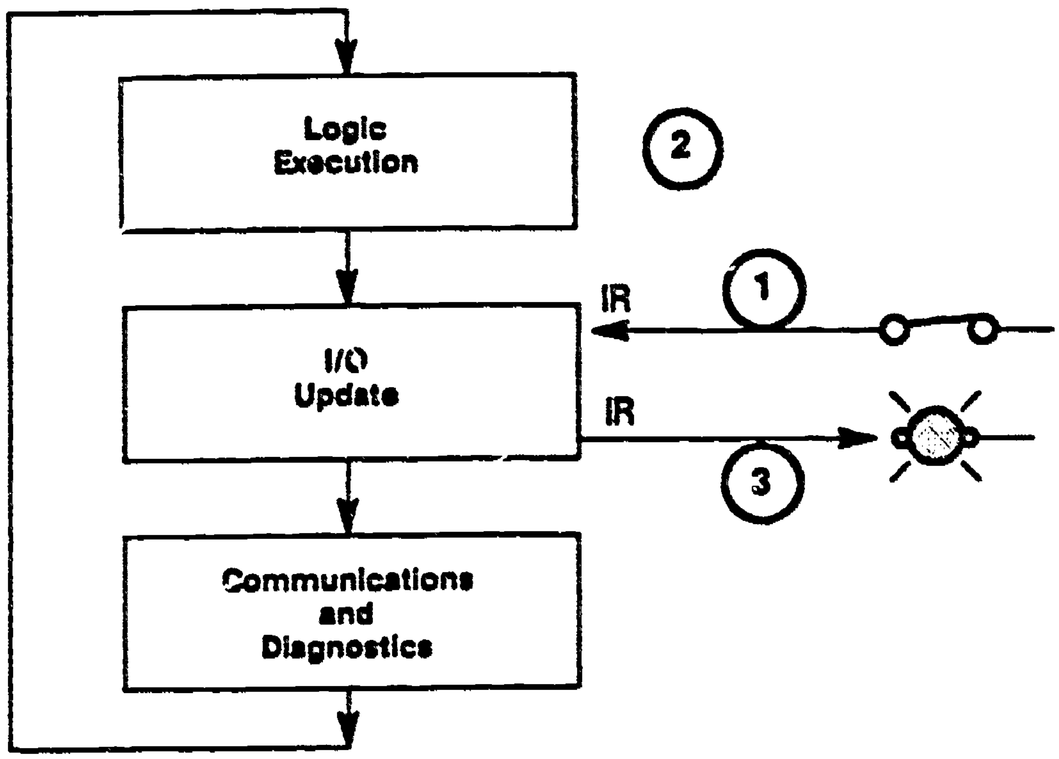
PLC Scan Operation



10 to 700 scans each second

Scan Time vs. Data Flow





To properly use programmable controllers, scan time must be understood and applied to discrete data flow through a programmable controller:

1 Status of the field device is brought into the input section of image register.

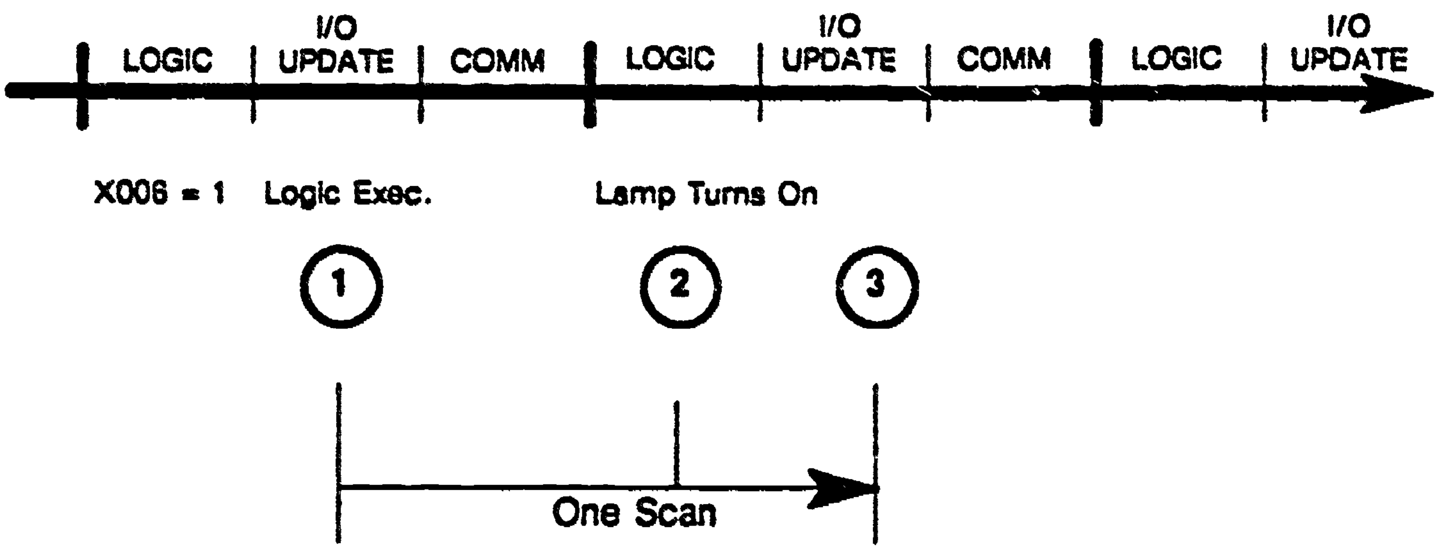
2 RLL is executed — software contacts are tested for OPEN or CLOSED.
— software outputs respond to logic; 1 or 0 to output IR.

3 Output section of image register "commands" the output module to turn on or turn off the associated field device.

Scan time is measured in milliseconds.

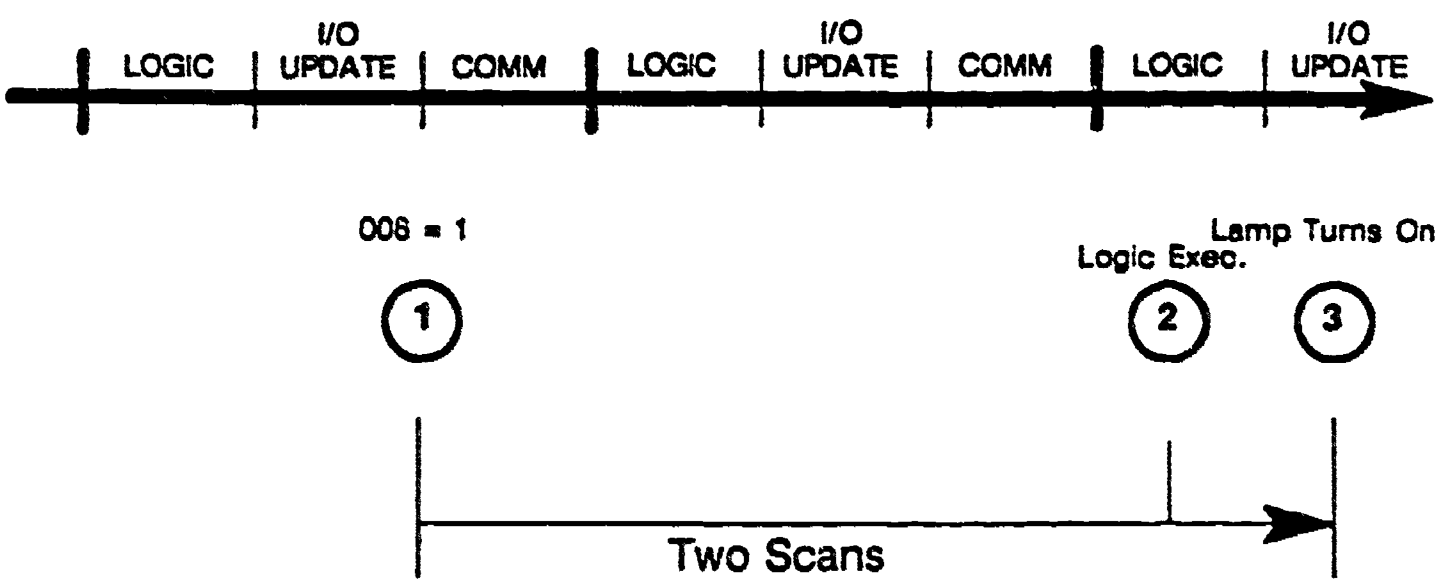
Scan time varies with different models of programmable controllers in different applications.

PLC Scan Time



Best Case

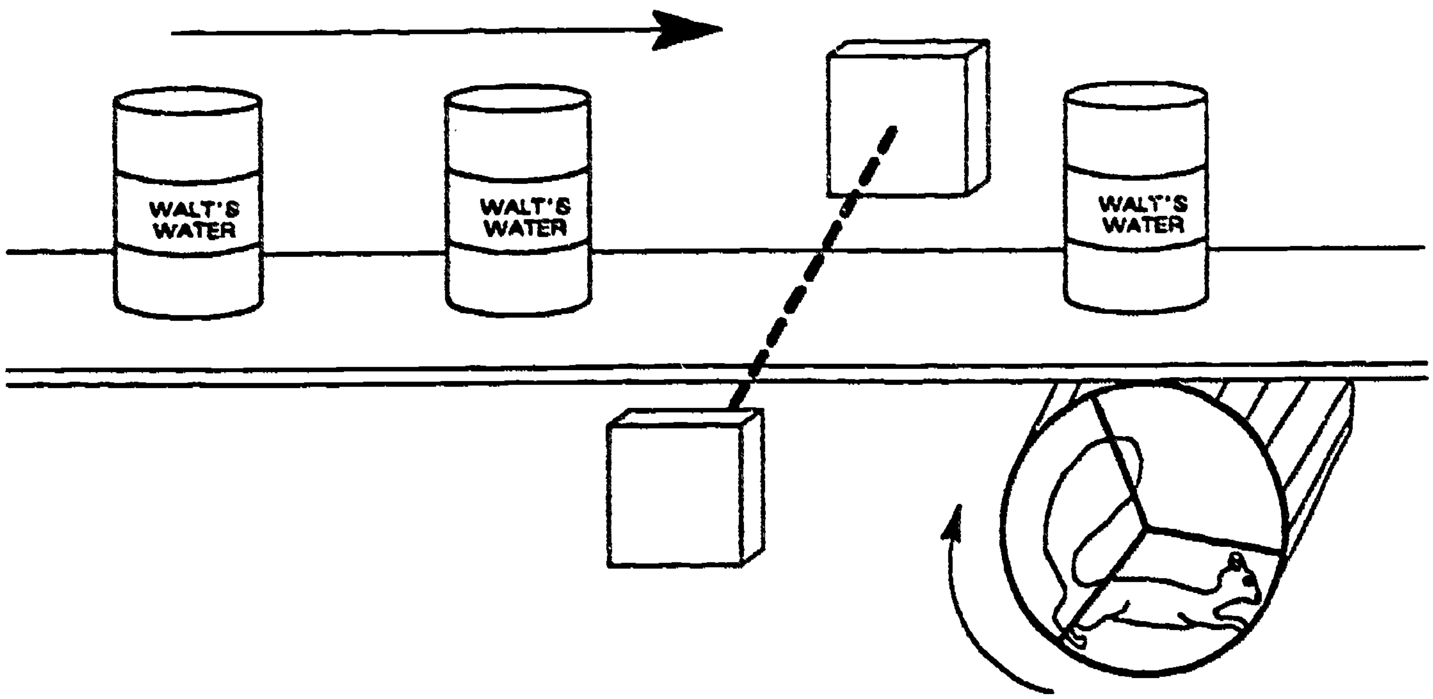
Turn-on Time = One Scan



Worst Case

Turn-on Time = Approximately Two Scans

Why Concern for PLC Scan Time?



In most cases, the common PLC scan is fast enough because the typical range of scan times is from 3 – 5 milliseconds (ms). (Ladder logic can be executed as fast as 0.49 ms/K words of RLL.)

In some applications, the concern about PLC scan time is whether or not the PLC is fast enough for the application or process involved.

Consider an extreme example where scan time may be too slow.

If cans were zipping along a conveyor line at a very fast pace of 100 cans per second, then the PLC scan time required to count a can would have to be less than 10 ms (i.e., 100 cans per sec. = 10 ms/can).

Any scan time longer than 10 ms would cause the PLC to miss counting some of the cans.

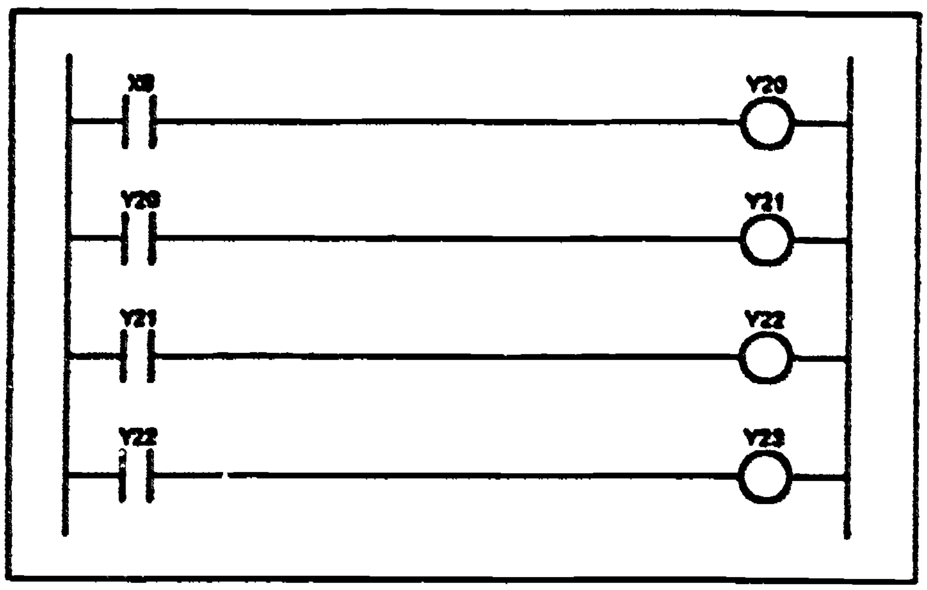
EXERCISE

Relay Circuits

Relay schematics are drawn to show electrical control, but the order in which the relay ladder rungs are put down on paper does not change the control logic.

The order of control logic is significant in a PLC.

Consider the following relay rungs as they are documented for an electromechnical relay control circuit (not a PLC).



Relay Control Circuit #1

1. What would happen to the coils if the normally open contact, X0, is closed?

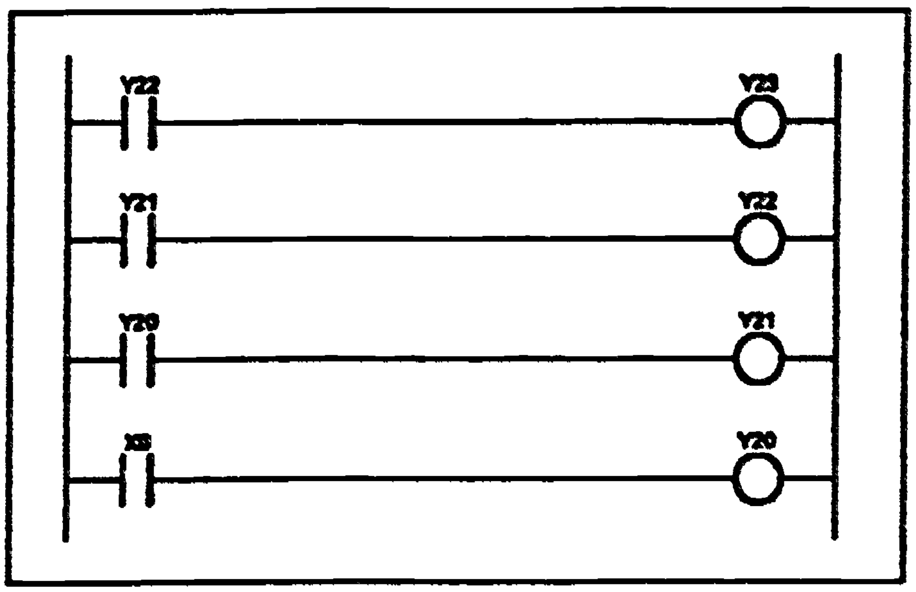
____

2. What would the timing be on the coil energizing?

____

EXERCISE

Now consider the following relay rungs. They are still documented for an electromechanical relay control circuit (not a PLC).



Relay Control Circuit #2

1. What would happen to the coils if the normally open contact, X0, is closed?

____

2. What would the timing be on the coil energizing?

____

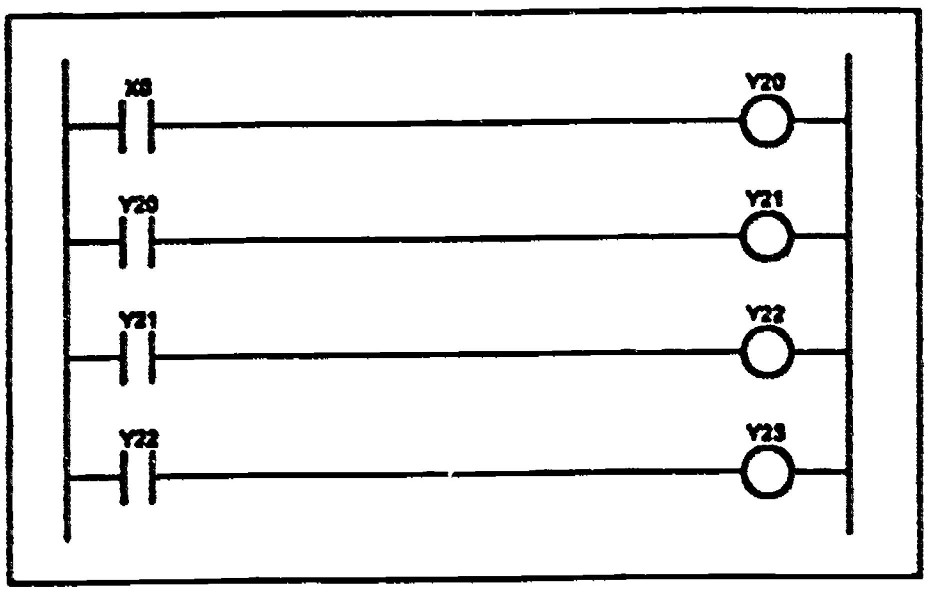
EXERCISE

RLL In a PLC

In a PLC, relay ladder logic schematics are drawn to show logical control. Now the order in which the relay ladder rungs are put down on paper can change the control logic.

The order of control logic is significant in a PLC.

Consider the following relay rungs as they are documented and entered into a PLC.



Relay Ladder Logic #1

1. What would happen to the coils if the normally open contact, X0, is closed?

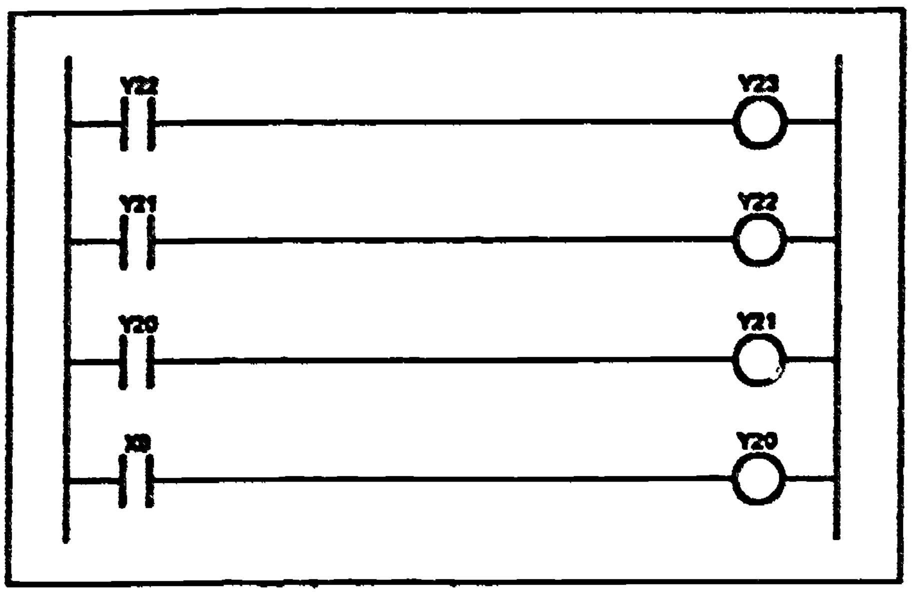
   ____

2. What would the timing be on the coil energizing?

   ____

EXERCISE

Now consider the following rungs. They are still documented and entered into a PLC.



Relay Ladder Logic #2

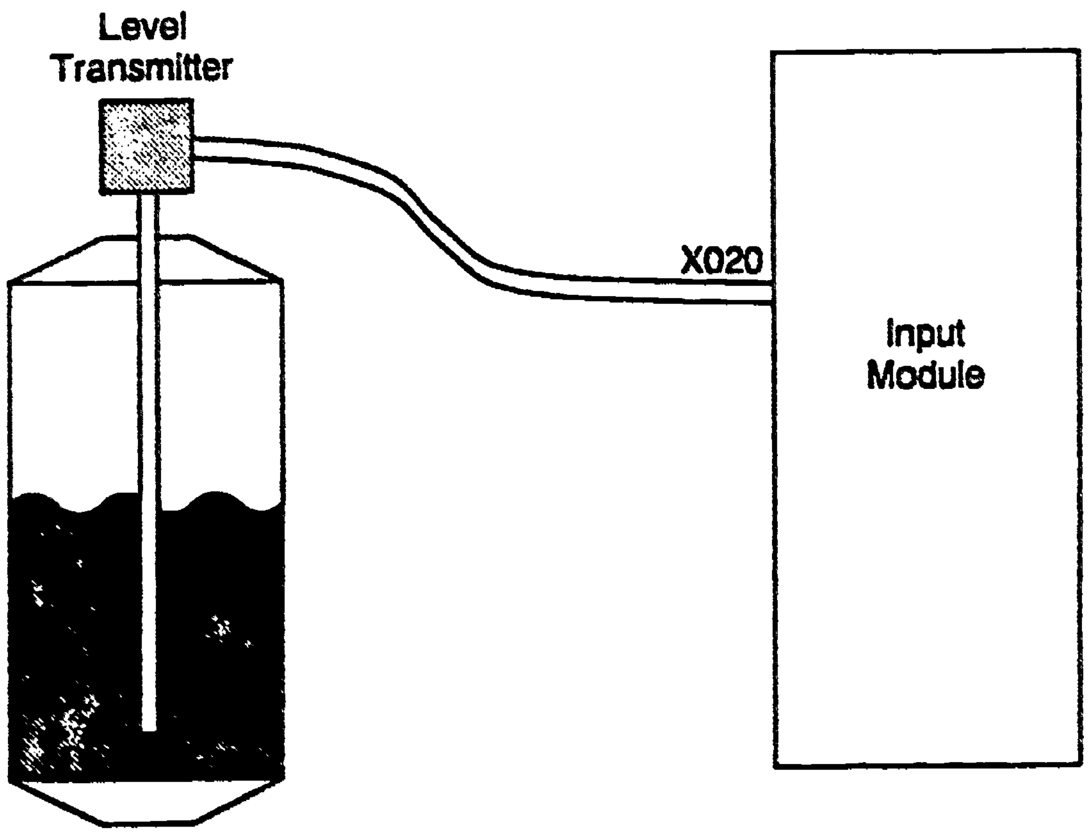
1. What would happen to the coils if the normally open contact, X0, is closed?

   ___

2. What would the timing be on the coil energizing?

   ___

Simple Analog Control



The flow of analog control is very much like the flow of discrete control in a PLC.

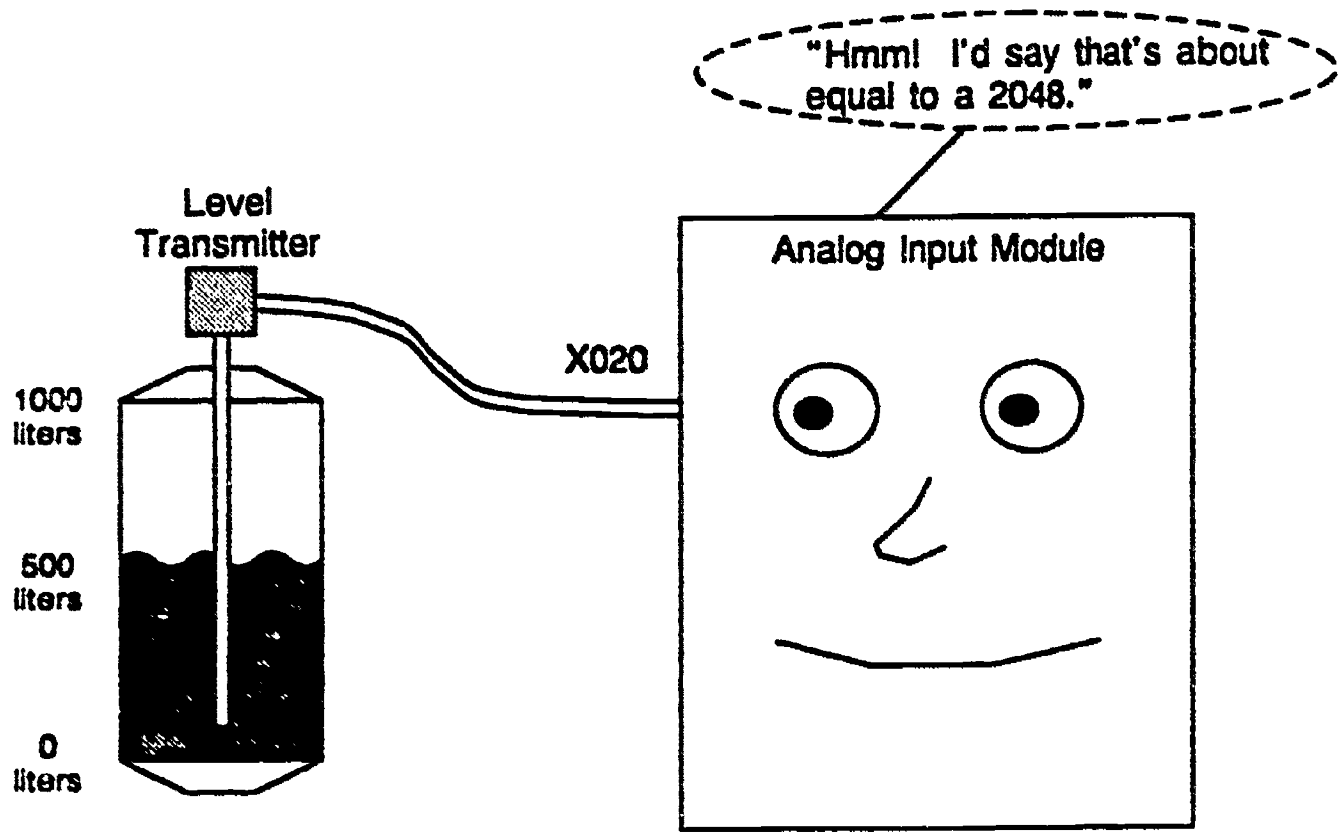
Instead of servicing one bit at a time, as in discrete, analog signals are handled in groups of 12 bits.

Generally, the signal begins in the process at the primary element tied to some type of electrical transmitter.

A varying electrical signal is transmitted to an Analog Input module.

Based on hardware and software configuration, the signal wire attached to the PLC is designated as the first address of a contiguous block of 12 discrete inputs.

Analog Signal Is Digitized

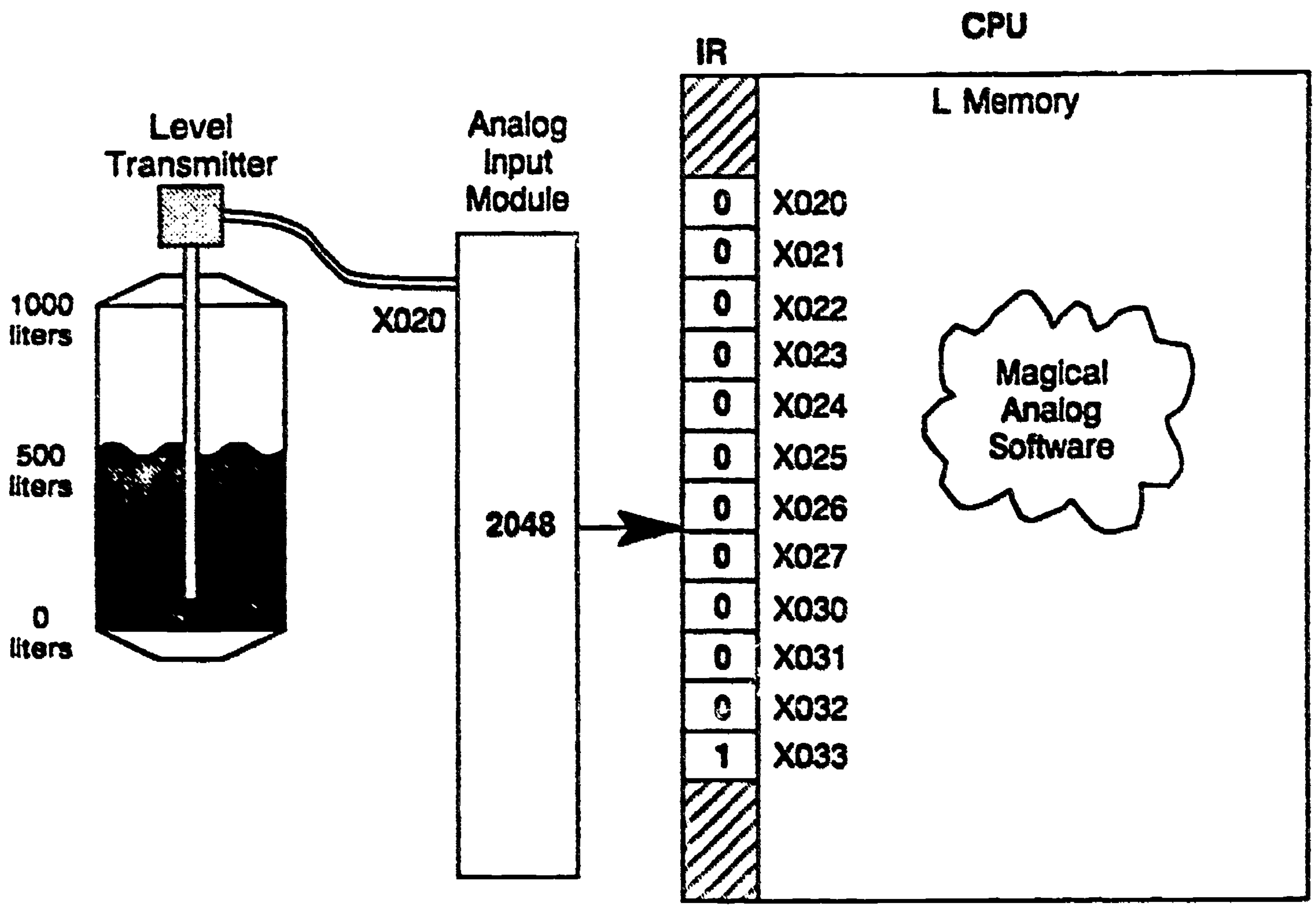


The Input Module converts this varying electrical signal to an integer value.

This is an **Analog-to-Digital (A/D)** conversion because a PLC must store an integer as a binary number.

This integer value is retained in the Analog Input Module until the CPU of the PLC does the input portion of an I/O update.

User Software Again Takes Over



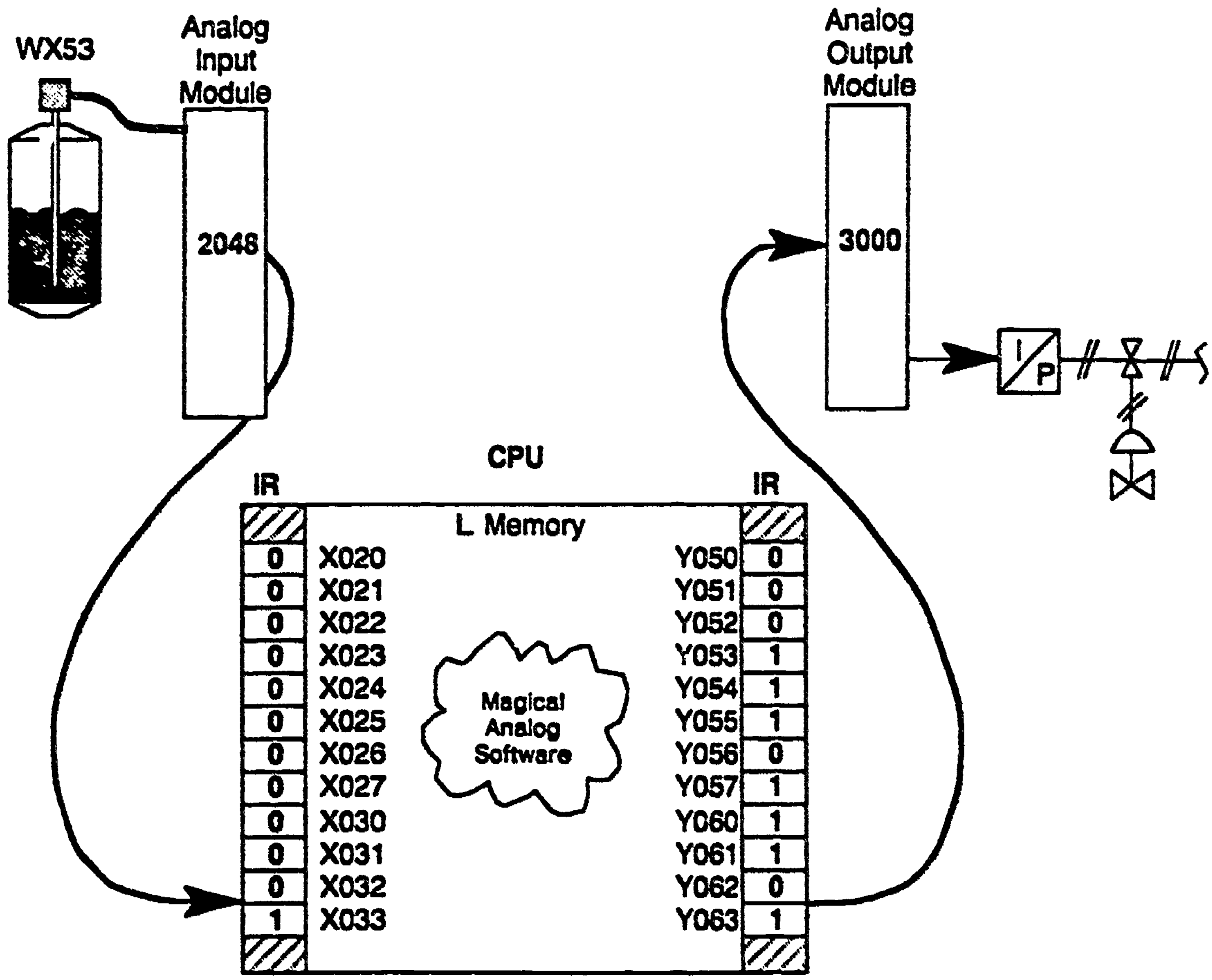
The integer representation of the analog signal is stored in a group of 12 contiguous discrete image register bits.

The CPU and I/O system hardware controls getting the analog data to 12 input bits.

Now the various types of RLL, or comparable software, take the analog value and perform all the necessary tasks upon this data:

Absolute Alarm Checking
Deviation Alarm Checking
etc.

Analog Output Control



If controlling an analog output is the goal of a user software package, then the software is written to deposit an integer value in 12 discrete output locations.

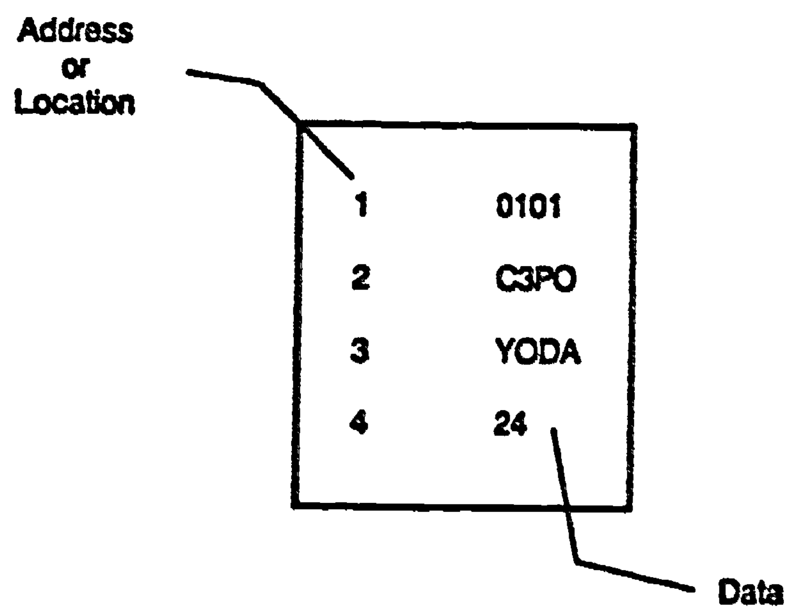
CPU and I/O system hardware take over and use the integer value to "command" an **Analog Output Module** to control an analog field device.

The Analog Output Module changes the integer in a **Digital-to-Analog (D/A)** conversion and passes this analog electrical signal to an appropriate field device.

Example: I/P Transducer

THE MEMORY SYSTEM AND I/O INTERACTION

Memory



Electronic memories — Places in an electronic device that store information for possible use at a future time.

Memory consists of two major parts:

- **Address or Location** — A place in the memory to store information.

  An address is a term that actually means the storage number of the location where the information is kept, but the two words are used interchangeably.

- **Data** — Information stored at any address. In a PLC this could be process dependent information or RLL program instructions.

MEMORY TYPES

1. RAM - RANDOM ACCESS MEMORY

   - Used for Volatile Memory

   - Runs Application Software

   - Sometimes Battery Backed to Store:

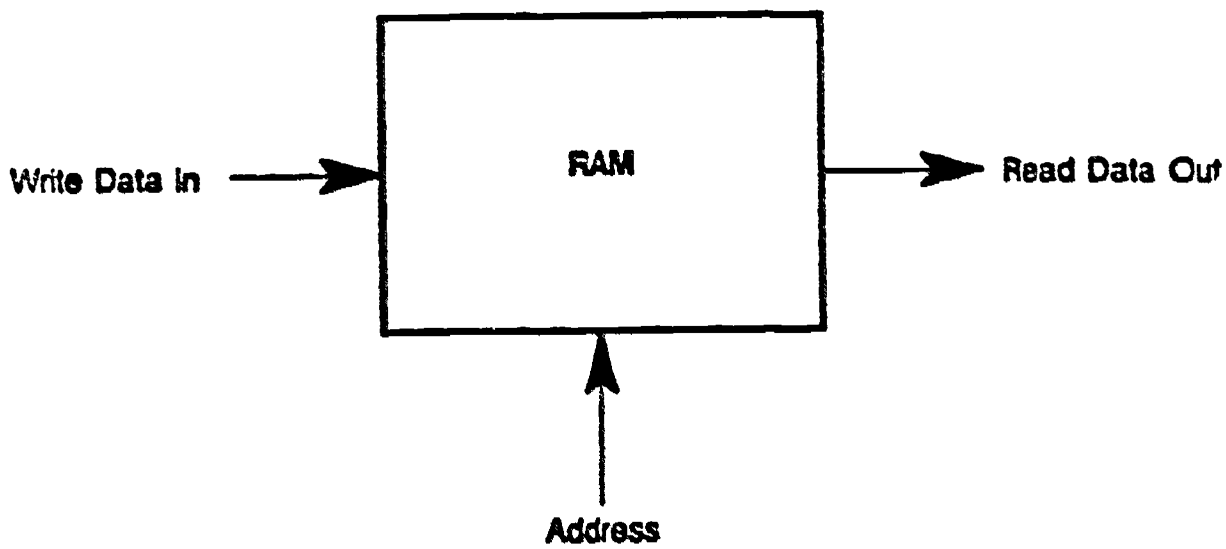
     - Clock
     - Date
     - System Information
       - Drive Types
       - Configurations

   - DRAM - Dynamic RAM

   - SRAM - Static RAM

   - VRAM - Video RAM

Read/Write Memory



Random Access Memory (RAM) — Literally means a memory where data can be accessed at any address without having to read a number of sequential addresses.

Random Access Memory (RAM) — Today means a memory where data can be read from and written to storage locations.

Read/Write Memory — Today means the same as RAM.

- *Advantage* — RAM allows the user to customize the decision making of the PLC quickly to adjust to the changing needs of the application.
- *Disadvantage* — RAM is volatile memory, which means a complete loss of power causes a loss of the program that may be stored in the RAM.
- *Battery Back-up* — Is required to avoid losing program and process data in the event of power loss.

MEMORY TYPES Continued

2. EPROM - ERASABLE PROGRAMMABLE READ ONLY MEMORY

   - Used for Non-Volatile Memory

   - Must be burned in with a Prom Burner

   - Stores Basic Input/Output System (BIOS)

   - Used to Store Permanent Applications Software on diskless PC's or SBC - Single Board Computers

   - Can only be erased with U.V. light

3. EE PROM - ELECTRICAL ERASABLE PROGRAMMABLE READ ONLY MEMORY

   - Same as EPROM except chip can be programmed and erased on board with software commands from the controller

Memory Size

1 K Memory	1 K Memory	1 K Memory
1 Bit	1 Byte	1 Word
2 Bits	2 Bytes	2 Words
3 Bits	3 Bytes	3 Words
⋮	⋮	⋮
1024 Bits	1024 Bytes	1024 Words

The size of memory is measured in *K* of memory. *K* is short for "*Kilo*."

When talking about computer memories:

1K *does not* mean 1000,

1K means 1024

A 1K memory refers to 1024 locations in memory.

These can be bits, bytes or words, depending on the memory type.

So, a 4K memory could be 4096 bits, or 4096 bytes, or 4096 words, etc.

TI MEMORY MAP

Operands	Quantity	Octal Numbering	V Memory Octal Word Address
Timer - Tn n	128	T0-T177	V00000-V00177
Counter CNT: PRE:	128	CT0-CT177	V01000-V01177
User Data (V Memory) V01400	3072	V01400-V07377	V01400-V07377
System Parameters V07400	256	V07400-V07777	V07400-V07777
Remote Inputs/Outputs GX GX	512	GX0-GX777	V40000-V40037
Inputs X	320	X0-X477	V40400-V40423
Outputs Y	320	Y0-Y477	V40500-V40523
Control Relays C	480	CR0-CR737	V40600-V40635
Stages 001	384	S0-S577	V41000-V41027
Timer Relays T:	128	T0-T177	V41100-V41107
Counter Relays CT:	128	CT0-CT177	V41140-V41147
Special Relays SP	96 192	SP0-SP137 SP320-SP617	V41200-V41205 V41215-V41230

Selecting Equipment Pretest

1. What do I need to know about controlling 4 toggle switches and 4 pushbuttons to select the correct module?

   a. Input impedance
   b. ON-state voltage drop
   c. OFF-state current leakage
   d. Input voltage
   e. I have no idea!

2. Why might the leakage current specification be an important consideration when selecting an output module?

   a. Certain process operations cannot tolerate any leakage
   b. Neon status lamps might stay half-lit on the Off state
   c. When isolation of wiring to field devices is required
   d. I have no idea!

3. What might happen if temperatures in a PLC enclosure regularly exceed the recommended maximum?

   a. One or more modules might fail prematurely
   b. CPU shuts down noncritical outputs to keep modules within specified operating ranges
   c. MIU displays an ASCII warning message to the operator
   d. I have no idea!

Pretest (continued)

4. Which module is required to drive a 4-digit BCD display?

   a. Data Communications Module
   b. A 16-point discrete output module
   c. ASCII/BASIC module
   d. Interrupt Input module
   e. I have not idea!

5. Which of the following devices provides a way to store programs safely?

   a. EPROM cartridge
   b. CMOS RAM cartridge
   c. Data Communications Module
   d. Cryonic Storage Refrigerator Unit
   e. I have no idea!

6. What electrical signal is provided by an analog output module to control the flow of liquid?

   a. 10 - 50 V
   b. 0 - 5 mA
   c. 4 - 20 mA
   d. I have no idea!

DISCRETE INPUT/OUTPUT SYSTEMS

DISCRETE INPUT/OUTPUT SYSTEMS

- I/O Rack Enclosures and Table Mapping
- Remote I/O Systems
- Discrete Inputs
- Discrete Outputs
- Interpreting I/O Specifications

DISCRETE I/O SYSTEMS

- **I/O RACK ENCLOSURES**

  Master Rack

  - Contains CPU and possibly I/O modules
  - CPU for entire system

  Local Rack

  - Contains local I/O processor and I/O modules
  - Same location as Master Rack
  - Local I/O processor (if not Master) receives and sends data to/from CPU
  - I/O addresses mapped to the I/O table

  Remote Rack

  - Located away from the CPU
  - Remote I/O processor and I/O modules
  - I/O addresses mapped to the I/O table

DISCRETE I/O SYSTEM

- I/O TABLE MAPPING

  - Racks emphasize physical location and type of processor

  - Table mapping emphasizes addresses used to reference all I/O modules

  - Factors which determine address:
    Type of Module
    Rack Number
    Slot Number
    Terminal Number

DISCRETE I/O SYSTEMS

REMOTE I/O SYSTEMS

- CHARACTERISTICS

  - Rack-Type enclosure located remotely from CPU
  - Power supply, communications, and I/O capability
  - High speed (up to 1M band) communication to CPU
  - Discrete, analog, and special function I/O

- Topology

  - Daisy Chain
  - Star
  - Multidrop (bus)

- Justification

  - Save on wiring materials and labor costs
  - Start up subsystems independently
  - Perform maintenance on individual subsystems
  - Simplify trouble-shooting

DISCRETE I/O SYSTEMS

Discrete Inputs

- **Characteristics**
  - ON/OFF, Open/Closed, Switch closure equivalent

- **Operation**
  - Input power from back plane of rack or base
  - Signal from input field devices

- **Examples**
  - Pushbuttons
  - Limit switches
  - Selector switches
  - Proximity switches
  - Level switches
  - Photoelectric eyes
  - Relay contacts
  - Circuit breakers
  - Motor starter contacts
  - Thumbwheel switches

- **Power**
  - 24, 48, 120, 230 Volts, AC or DC, TTL, Isolated, Non-voltage, 5-50 Volts DC (sink/source)

DISCRETE I/O SYSTEMS

Discrete Outputs

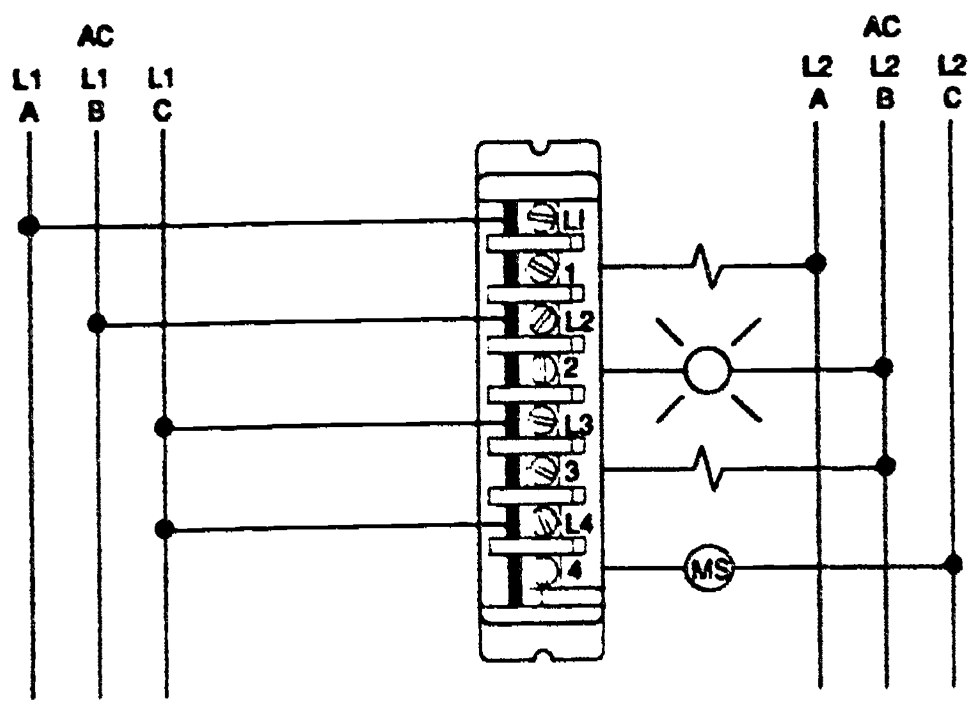
- **AC Outputs**
  - Logic section
  - Power section and switching circuit
  - Triac or SCR to switch the power

- **DC Outputs (sink/source)**
  - Similar to AC Outputs
  - Power circuit uses power transistor to switch load
  - Sink: current form the load flows into the module
  - Source: current flows from module, switching the positive voltage to the load

DISCRETE I/O SYSTEMS

Discrete Outputs

- **Isolated Outputs (AC and DC)**
  - Each output has its own return line (common) isolated from the other outputs
  - Allows control of outputs powered by different sources



DISCRETE I/O SYSTEMS

Discrete Outputs

- **TTL Outputs**
  - Seven-segment LED displays
  - Integrated circuits
  - Various 5 VDC devices
  - Generally require external 5 VDC power supply

- **Register or BCD Outputs**
  - Multi-bit interface
  - Provides parallel communication between the processor and an output device
  - 5 to 30 VDC and low current requirements
  - May have multiplexing capabilities (for passive devices)

- **Contact Outputs**
  - Allow output devices to be switched by a N.O. or N.C. relay contact

DISCRETE I/O SYSTEMS

Interpreting I/O Specifications

- **Specifications**
  - Define module capabilities and limitations
  - Define field equipment which module can operate

- **Electrical**
  - Input Voltage Rating or Range

    e.g. 120 VAC $\pm$ 10% => 108 to 132 VAC

  - Input Current Rating

    May appear indirectly as minimum power requirement

  - Input Threshold Voltage

    Volatge at which input is absolutely ON

  - Input Delay

    Duration of input signal in ON state in order to be recognized (9-25 msec for AC/DC inputs)

  - Output Voltage Rating or Range

    Typically $\pm$ 10% to 15%

  - Output Current Rating or ON-State Continuous Current Rating

    Maximum current that single circuit can safely carry under load (at specified temperature)

DISCRETE I/O SYSTEMS

Interpreting I/O Specifications

- **Electrical (continued)**
  - Output Power Rating
    Maximum total power that output module can dissipate with all circuits energized

    Verify rating for individual output when all other outputs are energized with manufacture

  - Current Requirements
    Current demand that I/O module places on system power supply

  - Surge Current (max)
    Defines maximum current and duration (e.g. 20 Amps at 0.1 sec)

  - Off State Leakage Current
    Maximum value of current that flows through triac/transistor in OFF state

  - Output ON-Delay
    Response time for OFF to ON transition after command to turn ON is received

DISCRETE I/O SYSTEMS

Interpreting I/O Specifications

- **Electrical (continued)**
  - Output OFF-Delay
    Response time for ON to OFF transition after OFF command is received from logic

  - Electrical Isolation
    Isolation in volts between I/O circuit and module logic circuitry. Protects logic circuits, but power circuitry of the module may be damaged

- **Mechanical**
  - Points Per Module
    Number of input or output circuits on a single module

  - Wire Size
    Number of conductors and largest wire gauge that I/O terminations will accept

DISCRETE I/O SYSTEMS

Interpreting I/O Specifications

- **Environmental**
  - Ambient Temperature
    Maximum temperature of air surrounding I/O system for best operating conditions. Should not be exceeded

  - Humidity Rating
    Humidity may cause circuit failure if moisture is allowed to condense on PC boards

TYPICAL DISCRETE INPUT MODULE SPECIFICATIONS

Specifications	U-05N 12-24 VDC input (Negative logic)	U-50N 24-48 VDC input (Isolated)	U-55N 12-24 VAC/VDC input
Circuits per module	16	8	16
Commons per module	2	8	2
Normal input voltage range	10.2-26.4 VDC	20-52.8 VDC	10.2-26.4 VAC/VDC
AC frequency	N/A	N/A	47-63 Hz
Maximum input voltage	26.4 VDC	52.8 VDC	37.5 VAC/VDC
Input current	3.8 mA (12 V) 8.3 mA (24 V)	5 mA (24 V) 10 mA (48V)	3.8 mA (12 V) 8.3 mA (24 V)
Input switching characteristics			
ON level	9.5 V	>15 V	9.5 V
OFF level	4.0 V	<7 V	3.0 V
OFF to ON response	1-7 ms	3-10 ms	5-40 ms
ON to OFF response	2-12 ms	3-12 ms	10-50 ms
Status indicator	Logic side	Logic side	Logic side
Maximum OFF current	1.5 mA	1.5 mA	1.5 mA
Minimum ON current	3.5 mA	3.5 mA	4.0 mA
Internal power consumption (max.) @ +5 V	150 mA	100 mA	150 mA
Weight	250g	250g	250g
Removable connector	Yes	Yes	Yes
Temperature derating	No	No	No

Specifications	U-20N 115/230 VAC input (Negative/positive logic)	U-25N 115 VAC input
Circuits per module	8	16
Commons per module	2	2
Normal input voltage range	80-265 VAC	80-132 VAC
AC frequency	47-63 Hz	47-63 Hz
Maximum input voltage	265 VAC	132 VAC
Input current	8.5 mA (100 V, 60 Hz) 20 mA (230 V, 60 Hz)	12.5 mA (100 V, 60 Hz) 14.5 mA (115 V, 60 Hz)
Input switching characteristics		
ON level	>70 V	>70 V
OFF level	<30 V	<30 V
OFF to ON response	5-30 ms	5-30 ms
ON to OFF response	10-50 ms	10-50 ms
Status indicator	Logic side	Logic side
Maximum OFF current	2 mA	2 mA
Minimum ON current	5 mA	7 mA
Internal current consumption (max.) @ +5 V	100 mA	150 mA
Weight	240g	270g
Removable connector	Yes	Yes
Temperature derating	No	No

TYPICAL DISCRETE OUTPUT MODULE SPECIFICATIONS

Specifications	U-01T Relay Output	U-05T Relay Output	U-12T 12/24 VDC Output (Negative logic, Sinking)	U-55T 12/24 VDC Output (Positive logic, Sourcing)
Output type	N. O. relay	N. O. relay	NMOS FET Open drain	NPN
Circuits per module	8	16	8	16
Commons per module	2	2	2	2
Operating voltage	5-250 VAC 5-30 VDC	5-250 VAC 5-30 VDC	10.2-26.4 VDC	10.2-26.4 VDC
Peak voltage	265 VAC	265 VAC	40 V	40 VDC
Maximum current (resistive)	2 A 5 A/common	1 A 5 A/common	2 A 5 A/common	0.5 A 3 A/common
Maximum leakage current	0.1 mA @ 265 VAC	0.1 mA @ 265 VAC	0.1 mA @ 40 VDC	0.1 mA @ 40 VDC
ON voltage drop	N/A	N/A	0.5 V @ 2 A	1.5 V @ 0.5 A
Smallest recommended load	5 mA @ 5 VDC	5 mA @ 5 VDC	0.2 mA @ 5 VDC	0.2 mA @ 5 VDC
Maximum inrush	2 A	4 A	8 A for 100 ms 12 A for 10 ms	1 A for 100 ms 2 A for 10 ms
OFF to ON response	12 ms	10 ms	1 ms	1 ms
ON to OFF response	12 ms	10 ms	1 ms	1 ms
Status indicator	Logic side	Logic side	Logic side	Logic side
Internal fuses	2 @ 8 A	2 @ 8 A	2 @ 7 A	2 @ 5 A
Replaceable fuses	No	No	No	No
Internal current consumption (max.) @ +5 V	550 mA	1000 mA	150 mA	350 mA
Weight	260g	310g	240g	280g
Removable connector	Yes	Yes	Yes	Yes
Temperature derating	No	No	No	Yes
External power source required	No	No	36 mA max @ 26.4 VDC	No

Specifications	U-15T 24 VDC Output (Negative logic, Sinking)	U-20T 115/230 VAC Output	U-25T 115/230 VAC Output
Output type	NPN Open collector	SSR	SSR
Circuits per module	16	8	16
Commons per module	2	2	2
Operating voltage	4.5-26.4 VDC	15-265 VAC	15-265 VAC
Peak voltage	40 V	265 V	265 VAC
Maximum current (resistive)	0.5 A 3 A/common	2 A 5 A/common	0.5 A 3 A/common
Maximum leakage current	0.1 mA @ 40 VDC	5 mA @ 265 VAC	4 mA @ 265 VAC
ON voltage drop	0.5 V @ 0.5 A	1.5 V @ 2 A	1.5 V @ 0.5 A
Smallest recommended load	0.2 mA	10 mA @ 15 V	10 mA @ 15 V
Maximum inrush	1 A for 100 ms 2 A for 10 ms	10 A for 100 ms 30 A for 10 ms	10 A for 100 ms 15 A for 10 ms
OFF to ON response	0.5 ms	1 ms	1 ms
ON to OFF response	0.5 ms	1 ms + 1/2 cycle	1 ms + 1/2 cycle
Status indicator	Logic side	Logic side	Logic side
Internal fuses	2 @ 5 A	2 @ 8 A	2 @ 5 A
Replaceable fuses	No	No	No
Internal current consumption (max.) @ +5 V	200 mA	250 mA	450 mA
Weight	270g	330g	350g
Removable connector	Yes	Yes	Yes
Temperature derating	No	Yes	Yes
External power source required	125 mA @ 24 VDC	No	No

Planning and Documenting Installation
Pretest

1. Under what circumstances might humidity be a concern in a PLC installation?

   a. When there is condensation
   b. When the humidity level goes below 20%
   c. When the humidity level is between 75% and 85%
   d. When the humidity level cannot be measured
   e. I have no idea!

2. In an octal numbering system, the number that follows 77 is:

   a. 78
   b. 80
   c. 100
   d. 101
   e. I have no idea!

3. If you are using a TI435 CPU, what component must be installed to run an application?

   a. Memory cartridge
   b. TISOFT
   c. Machine Interface Unit
   d. Floppy disk drive
   e. I have no idea!

Pretest (continued)

4. What is the maximum distance that a remote I/O base can be located from the local CPU base in a Series 405 system?

    a. 100 meters
    b. 1000 meters
    c. 1000 feet
    d. 10,000 feet
    e. I have no idea!

5. How does a typical Machine Interface Unit communicate with the CPU?

    a. Through a 15-pin cable
    b. Through a standard 9-pin RS-422 cable
    c. Through a noise-free fiber-optic link
    d. I have no idea!

6. What is the minimum memory required on your personal computer to run typical PLC programming software?

    a. A minimum of 5.12 kilobytes of RAM
    b. A minimum of 5 Megabytes of RAM
    c. A minimum of 512 kilobytes of RAM
    d. A minimum of 512 kilobytes of ROM
    e. I have no idea!

Pretest (continued)

7. What is AUTOEXEC.BAT?

    a. A special batch file in a computer operating system
    b. A data file in which TISOFT stores user RLL programs
    c. A baseball device for a car company vice president
    d. I have no idea!

8. What is the recommended method of rebooting a personal computer with minimum stress on electronic components?

    a. Turn power OFF, then ON again
    b. Unplug the system unit, then plug it in again
    c. Press the keyboard keys: <Ctrl> <Alt> <Del>
    d. Punt it, don' t place kick it
    e. I have no idea!

9. What module would you use to communicate with a pulse encoder?

    a. Interrupt Input module
    b. High-Speed Counter module
    c. Analog Input module
    d. A heart rate monitor
    e. I have no idea!

Pretest (continued)

10. What hardware components are required to configure a typical remote I/O system?

    a. Data Communications module and a host computer
    b. Data Communications module and a CPU module in each remote base
    c. A module in the CPU base and a remote interface module in each remote base
    d. I have no idea!

11. How can I/O numbers be assigned?

    a. In a numerical count, starting nearest the CPU with a count of 1 and counting up
    b. In byte boundaries, automatically or manually
    c. Automatically, but not manually
    d. Manually, but not automatically
    e. This is a trick question -- I/O numbers are fixed by I/O slot
    f. Varies by manufacturer

12. Which module is required to drive a 4-digit BCD display?

    a. Data Communications module
    b. A 16-point discrete output module
    c. An analog output module
    d. Link Interface Controller module
    e. I have no idea!

Pretest (continued)

13. How much vertical space should typically be allowed between I/O bases in an enclosure installation?

    a. 2 inches
    b. 4 inches
    c. 6 inches
    d. No space required
    e. I have no idea!

14. Which DOS command creates a subdirectory called Fred?

    a. DIR/C FRED
    b. CD FRED
    c. MD FRED
    d. CR FRED
    e. I have no idea!

15. Which DOS command would copy files from a C: drive to A: drive?

    a. COPY A:*.*C:
    b. COPY C:*.*A:
    c. EDLIN A:*.*C:
    d. CD A:*.*C:
    e. CD C:*.*A:
    f. I have no idea!

Pretest (continued)

16. To find the names of the directories on a disk with DOS, type:

    a. TREE
    b. FORMAT
    c. DISP*
    d. DSP*
    e. DISP DIR
    f. I have no idea!

17. If you suspect hot spots on your PLC enclosure, where should temperature measurements be taken to see if it is within specifications?

    a. Measure 2" (5 cm) above the modules
    b. Measure 6" (15 cm) above the modules
    c. Measure between the modules
    d. Place a can of soup on the modules for 7 minutes and test by drinking
    e. I have no idea!

18. Which two items below should be powered down before removing an I/O module?

    a. I/O base and field power for all points on module
    b. CPU memory cartridge and field power for all points on module
    c. 100% of PLC system and field power for all points on module
    d. I have no idea!

BASICS OF PLC PROGRAMMING

BASICS OF PLC PROGRAMMING

- PLC Instructions
- Programming Timers
- Programming Counters
- Program Flow Control Instructions
- Entering Ladder Diagrams
- PLC System Documentation

BASIC PLC INSTRUCTIONS

PLC INSTRUCTIONS

BASIC RELAY

TIMER AND COUNTER

PROGRAM CONTROL

DATA MANIPULATION AND TRANSFER

SPECIAL FUNCTION

BASIC RELAY INSTRUCTIONS

CONTACTS - Represent conditions to be evaluated

Examine ON —| |—

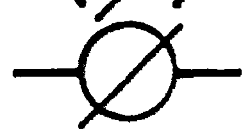
Examine OFF —|/|—

COILS

- Represent outputs
- Controlled by evaluation of conditions

OUTPUT —( )—

Not OUTPUT —(/)—

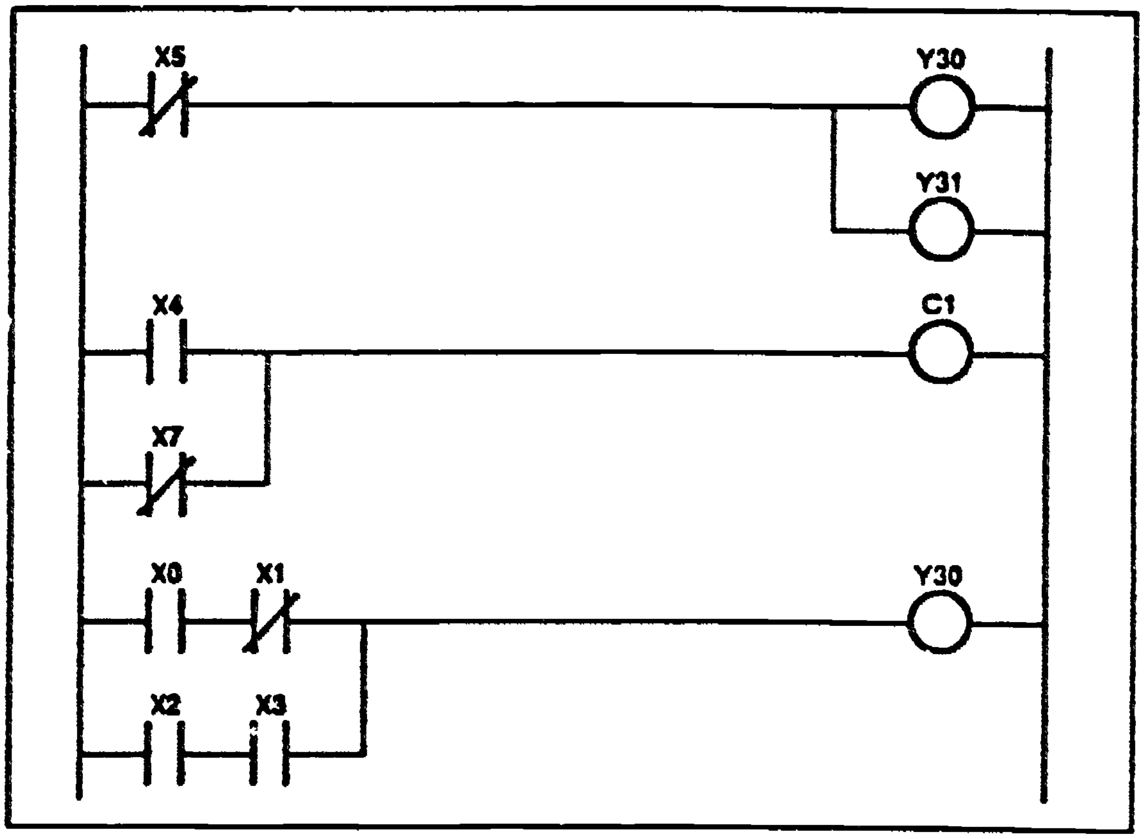


Latch —(L)—

Unlatch —(U)—

EXAMPLE: INVALID REFERENCE

Avoid using the same output coil reference twice in an RLL program stage. Some manufacturers will not allow it.



RLL Example, Invalid Reference

If the CPU could run this logic, on the RLL rung controlling the last reference to output Y30 would control the factory actuators.

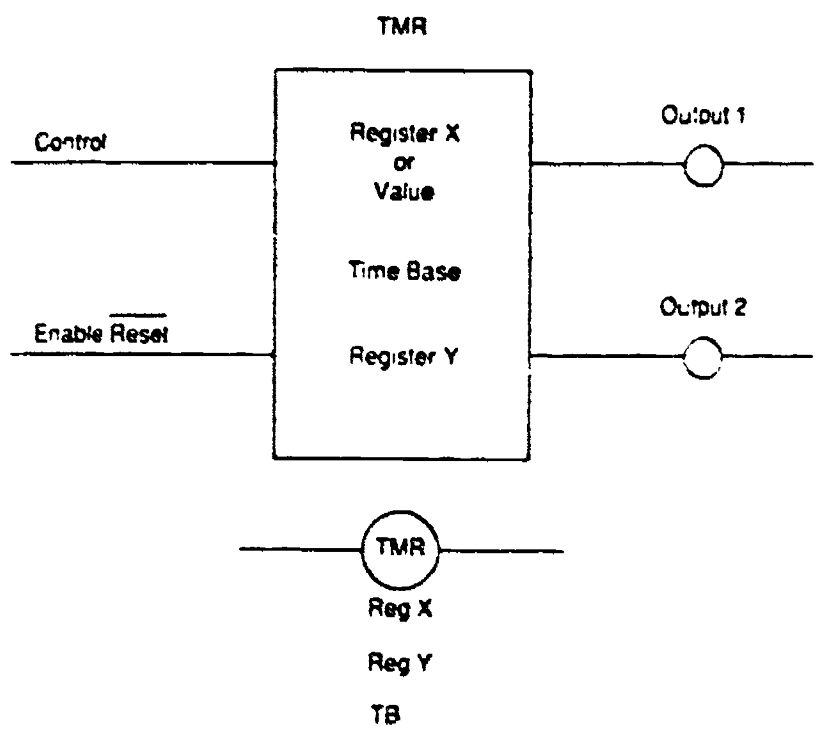
PROGRAMMING TIMERS

TIMER AND COUNTER INSTRUCTIONS

- TIMERS
  - Timer ON Delay Energize
  - Timer ON Delay De-Energize
  - Timer OFF Delay Energize
  - Timer OFF Delay De-Energize
  - Retentive Timer
- COUNTERS
  - Up-counter
  - Down-counter
  - Counter Reset

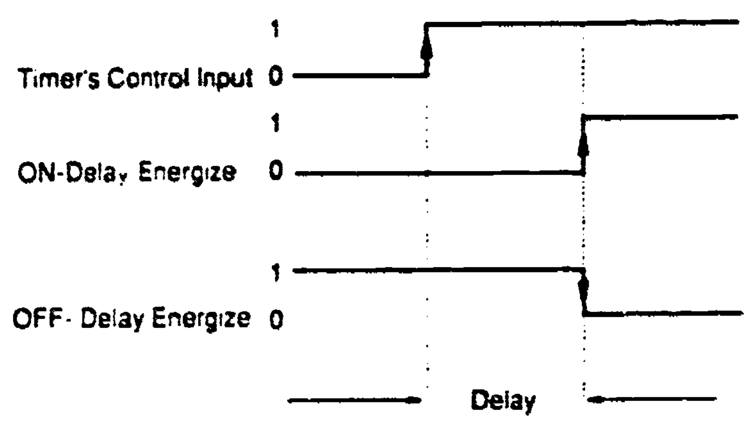
TIMERS

- Timer ON Delay Energize
- Timer ON Delay De-energize
- Timer OFF Delay Energize
- Timer OFF Delay De-energize
- Retentive Timer

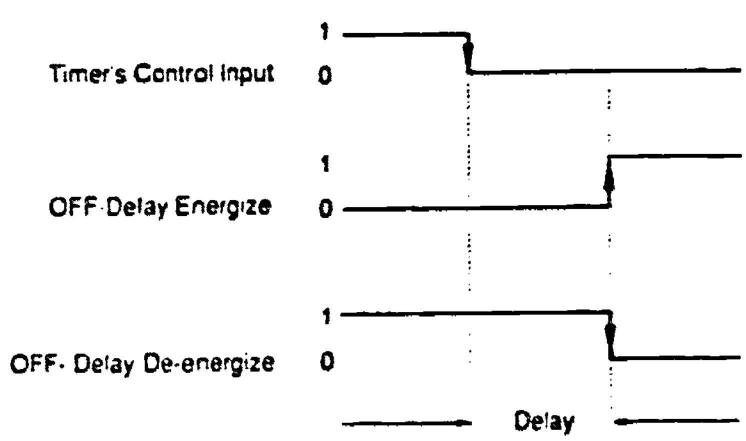


TIMER TIMING DIAGRAMS

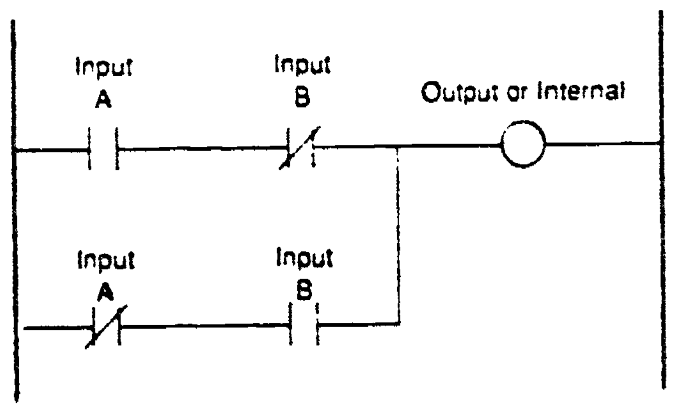
ON DELAY



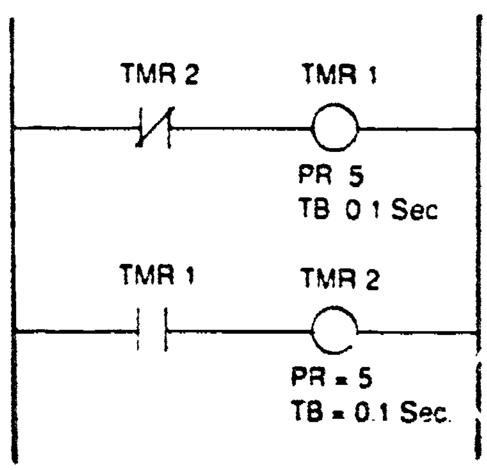
OFF DELAY



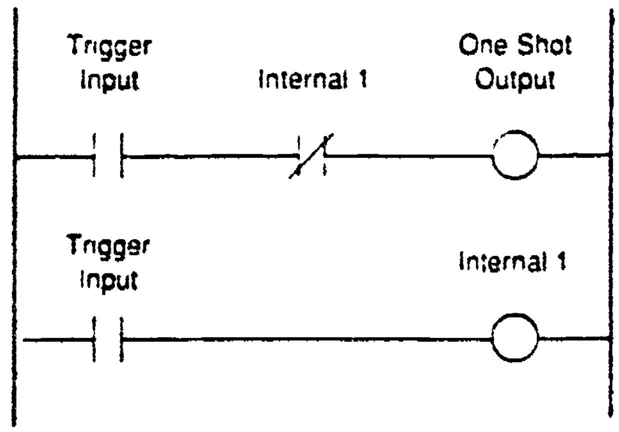
Exclusive OR



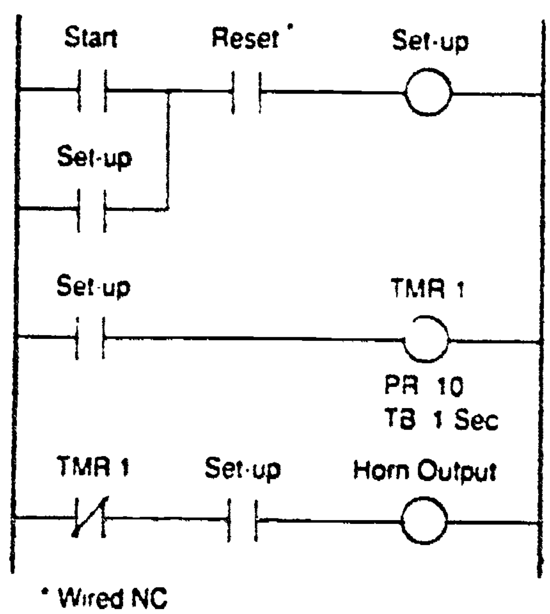
Oscillator Circuit



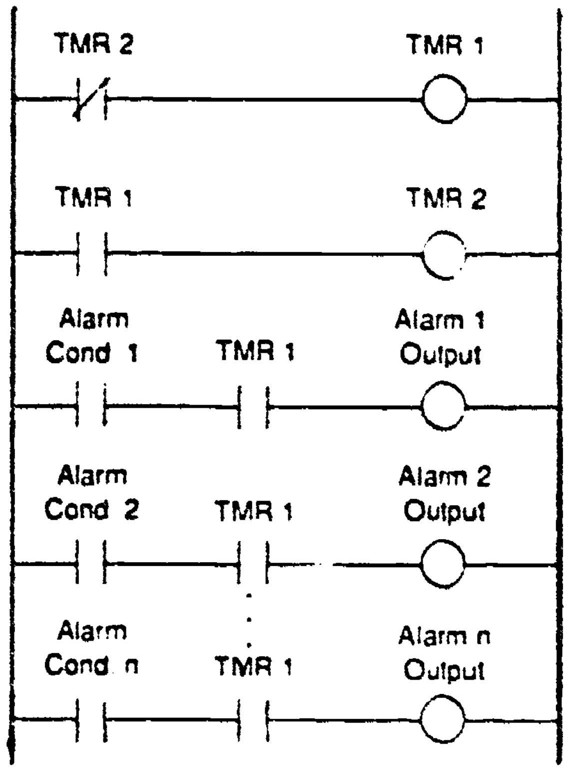
The One-Shot Signal



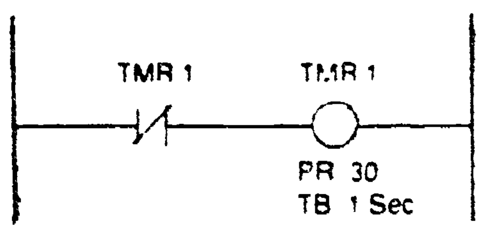
System Start-UP Horn



Annunciator Flasher Circuit

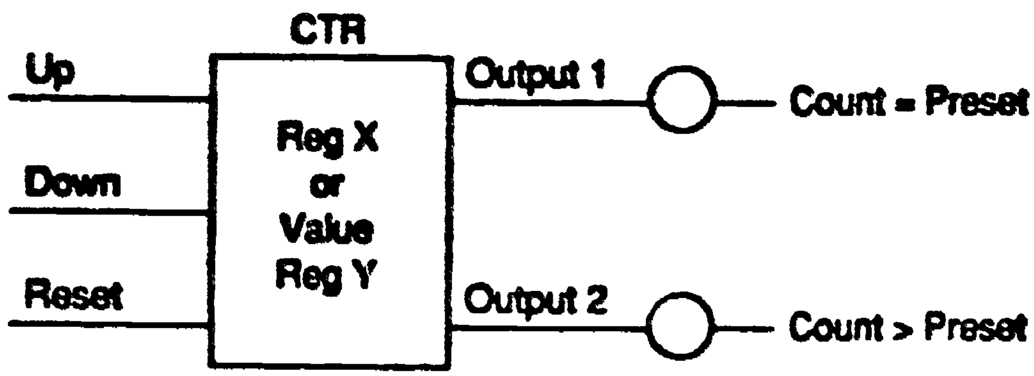


Self-Resetting Timer

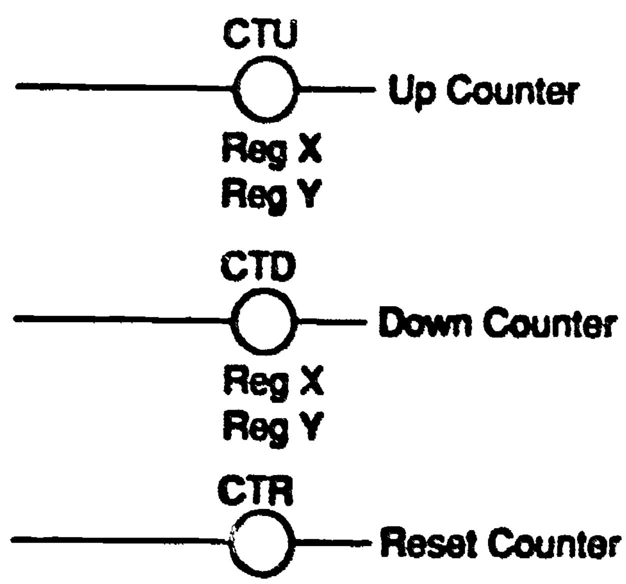


PROGRAMMING COUNTERS

COUNTERS



Block format



Basic ladder format

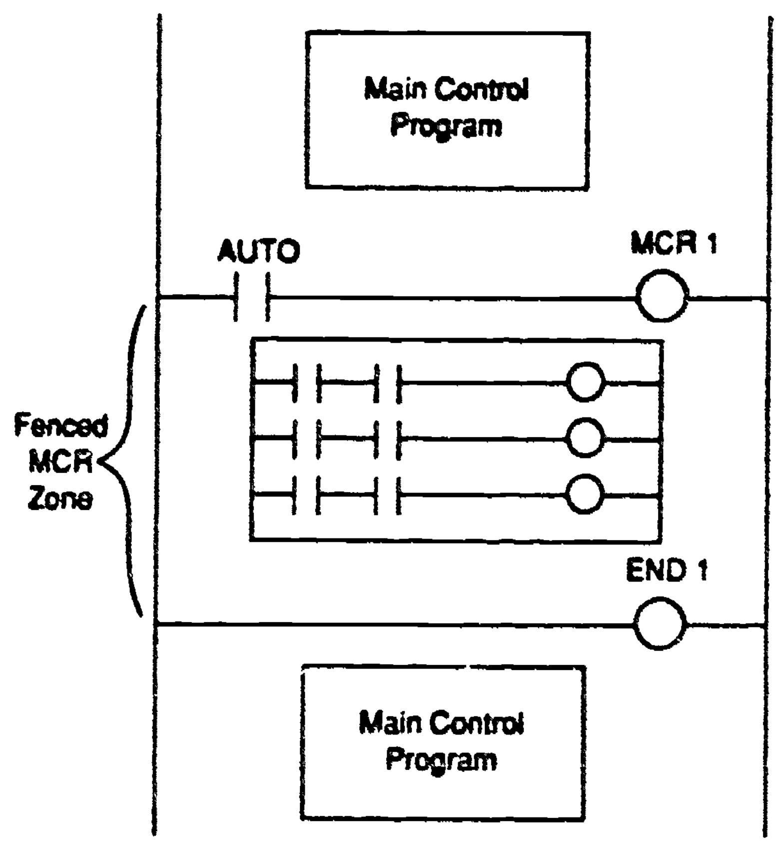
TIMER AND COUNTER SUMMARY

- TIMER (TMR)
- FAST TIMER (TMRF)
- ACCUMULATING TIMER (TMRA)
- FAST ACCUMULATING TIMER (TMRAF)
- COUNTER (CNT)
- STAGE COUNTER (SGCNT)
- UP/DOWN COUNTER (UDC)

PROGRAM FLOW CONTROL INSTRUCTIONS

PROGRAM CONTROL

- MASTER CONTROL RELAY (MCR)
  - (De)Activate execution of group or zone of ladder rungs
  - Used in conjunction with an END rung
  - If MCR is turned OFF, all non-retentive(non-latched) outputs within zone will be de-energized



PROGRAM CONTROL

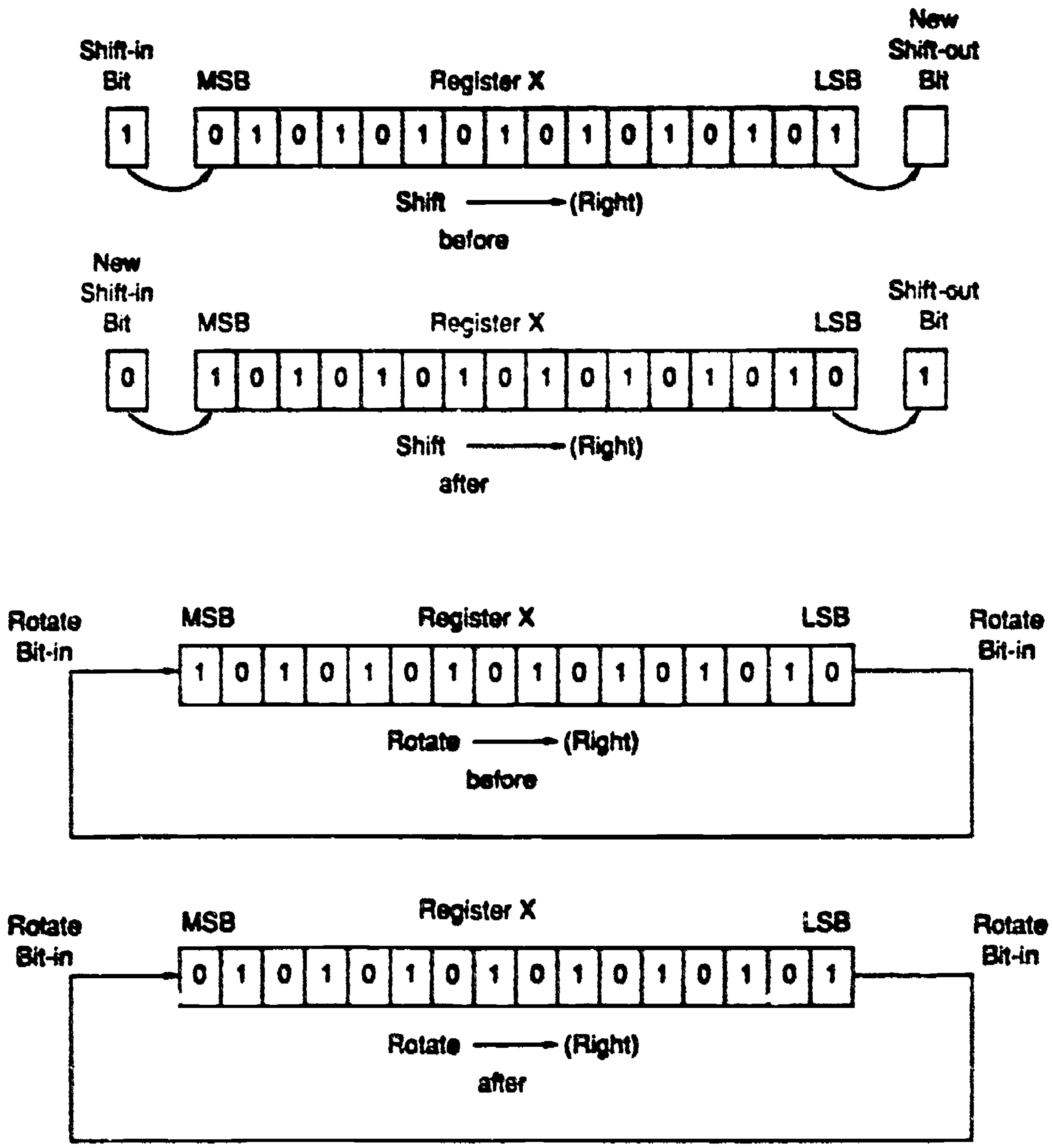
- ZONE CONTROL LAST STATE (ZCL)
  - Similar to MCR
  - If ZCL is turned OFF, all outputs within the zone will be held in this last state

DATA MANIPULATION AND TRANSFER

- DATA CONVERSIONS
  - BCD to Binary
  - Binary to BCD

DATA MANIPULATION AND TRANSFER

- LOGICAL SHIFTS AND ROTATES



IMPLEMENTING AND PROGRAMMING THE PLC

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

- CONTROL DEFINITION
- CONTROL STRATEGY
- IMPLEMENTATION GUIDELINES
- PROGRAM ORGANIZATION AND IMPLEMENTATION
  - Flowcharting
  - Configuring
  - I/O Assignments
  - Register Address Assignment
  - Portions to Leave Hardwired

CONTROL DEFINITION

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

CONTROL DEFINITION

- List requirements for the task
  - Operations
  - Users
  - Data reporting
- List Constraints
  - Time of execution
  - Types of field devices
  - Space and geography
- Review Current Procedure (if applicable)
  - Take advantage of PLC capabilities

CONTROL STRATEGY

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

CONTROL STRATEGY

- Determine Algorithm
  - Sequence of steps which produce the desired result
  - May need to add to control definition
  - Do not program (focus on design)
- Consider Alternative Approaches
  - Benefits: shortens programming time
    reduces debugging time
    accelerates start-up

IMPLEMENTATION GUIDELINES

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

IMPLEMENTATION GUIDELINES

- Flowchart the process operation
- Implement flowchart
  (logic diagrams or relay logic symbology)
- Assign I/O addresses (real and internal)
- Translate flowchart to PLC coding

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

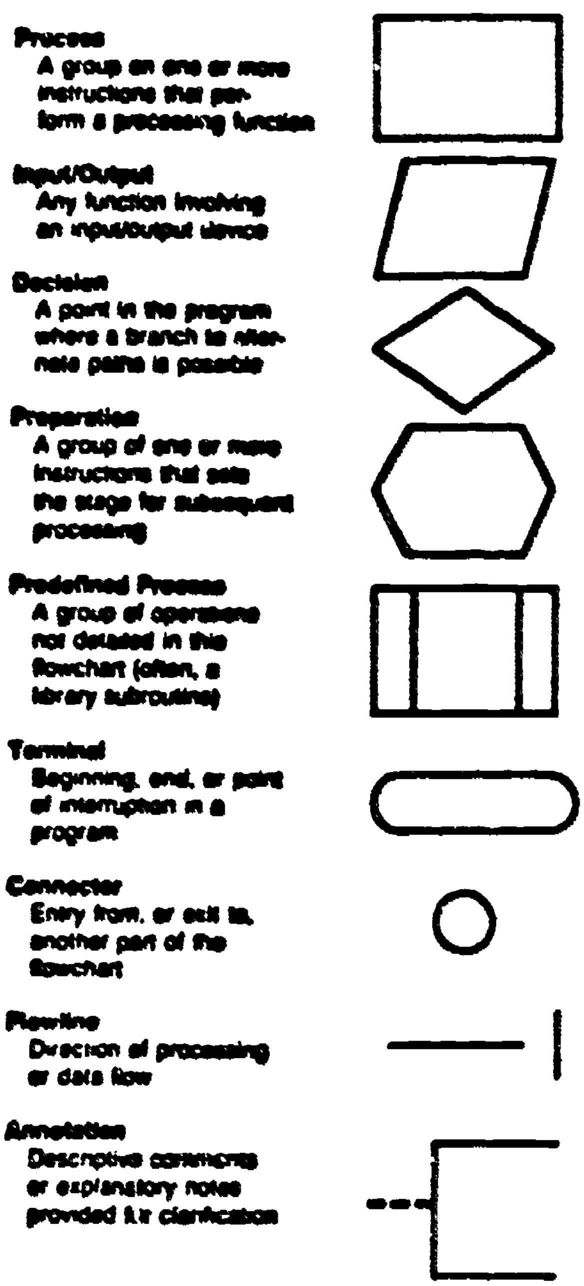
PROGRAMMING ORGANIZATION & INPLEMENTATION

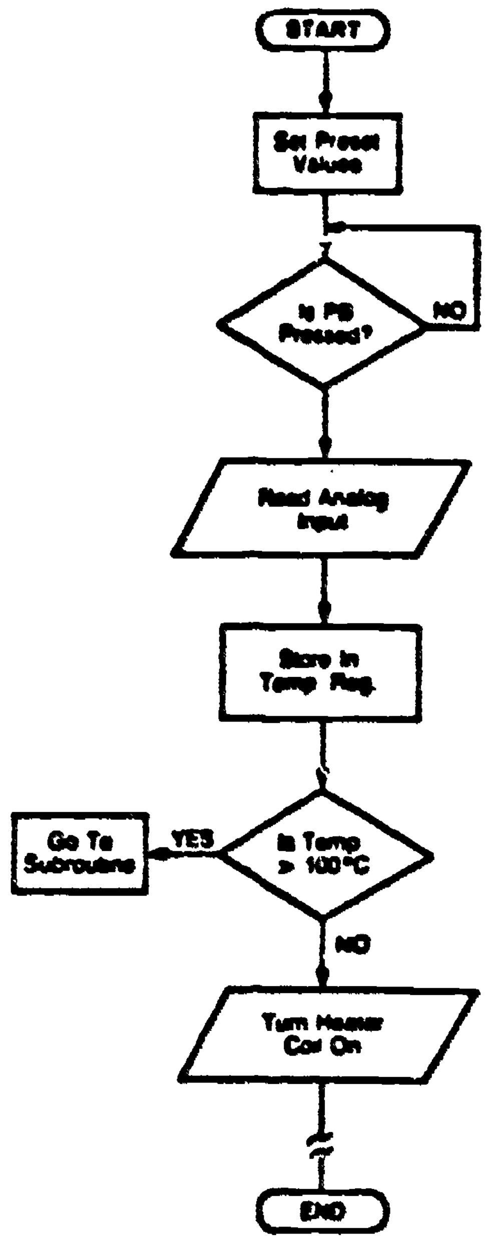
- Flowcharting
  - Pictorial representation for recording, analyzing, and communication information
  - Clearly depicts sequences and relationships
  - Broad or focused

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION & INPLEMENTATION

- Flowchart Symbols





PROGRAMMING ORGANIZATION

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION

- Configuring the PLC system
  - Required modules
    - Present
    - Future
  - Organize modules within racks
    - Grouping like signals
  - Location of racks
  - Power supply

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION

- Real and Internal I/O Assignment
  - Indicate connections between PLC I/O and field devices
  - Indicate assignment of internal address (timers, counters, MCRs, coils and contacts)
  - Recommended to assign all inputs or all outputs at the same time

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION

Register Address Assignment

- Work from list of all available registers
- Enter contents, description, and function as assigned

TI MEMORY MAP

Operands	Quantity	Octal Numbering	V Memory Octal Word Address
Timer – Tn n	128	T0-T177	V00000-V00177
Counter CNT: PRE:	128	CT0-CT177	V01000-V01177
User Data (V Memory) V01400	3072	V01400-V07377	V01400-V07377
System Parameters V07400	256	V07400-V07777	V07400-V07777
Remote Inputs/Outputs GX GX	512	GX0-GX777	V40000-V40037
Inputs X	320	X0-X477	V40400-V40423
Outputs Y	320	Y0-Y477	V40500-V40523
Control Relays C	480	CR0-CR737	V40600-V40635
Stages 001	384	S0-S577	V41000-V41027
Timer Relays T:	128	T0-T177	V41100-V41107
Counter Relays CT:	128	CT0-CT177	V41140-V41147
Special Relays SP	96 192	SP0-SP137 SP320-SP617	V41200-V41205 V41215-V41230

Glossary of programmable controller terms

This glossary was abstracted from "Glossary of Programmable Controller Terms," compiled by Allen-Bradley Company.

.....

A

ac input module: I/O Rack module which converts various ac signals originating in user switches to the appropriate logic level for use within the processor.

ac output module: I/O Rack module which converts the logic levels of the Processor to a usable output signal to control a user's ac load.

address: A location in the Processor's memory; usually used in reference to the Data Table.

analog input module: An I/O Rack module which converts an analog signal from a user device to a digital signal which may be processed by the Processor.

analog output module: An I/O Rack module which converts a digital signal from the Processor into an analog output signal for use by a user device.

arithmetic capability: The ability to do addition, subtraction, and in some cases multiplication and division, with the PC Processor.

ASCII: An eight-level (7 bits + a parity bit) code forms the American Standard Code for Information Interchange.

asynchronous: Not related through repeating time patterns.

asynchronous shift register: A shift register which does not require a clock. Register segments are loaded and shifted only at data entry.

B

baud: 1) A unit of data transmission speed equal to the number of code elements (bits) per second. 2) A unit of signaling speed equal to the number of discrete conditions or signal events per second.

BCD (Binary Coded Decimal): The 4-bit binary notation in which individual decimal digits (0 thru 9) are represented by 4-bit binary numerals; e.g., the number 23 is represented by 0010 0011 in the BCD notation.

binary: A numbering system using only the digits 0 and 1. Also called "base 2."

binary word: A related grouping of ones and zeroes having meaning assigned by position, or numerical value in the binary system of numbers.

bit: 1) An acronym for Binary digiT; the smallest unit of information in the binary numbering system. Represented by the digits 0 and 1. 2) The smallest division of a PC word.

bit rate: The rate at which binary digits, or pulses representing them, pass a given point in a communication line.

Boolean algebra: Shorthand notation for expressing logic functions.

bus: An electrical channel along which data can be sent or received.

byte: A sequence of binary digits usually operated upon as a unit. (The exact number depends on the system, but often 8 bits.)

C

card reader: A device for reading information from punched cards.

cassette recorder: A magnetic tape recording and playback device for entering or storing programs.

central processing unit (CPU): Another term for Processor. It includes the circuits controlling the interpretation and execution of the user-inserted program instructions stored in the PC memory.

character: One symbol of a set of elementary symbols, such as a letter of the alphabet or a decimal numeral. Characters may be expressed in many binary codes. For example, an ASCII character is a group of 7 bits.

chip: A tiny piece of semiconductor material on which microscopic electronic components are photo-etched to form one or more circuits. After connection leads and a case are added to the chip, it is called an Integrated Circuit.

circuit card: A printed circuit board containing electronic components.

clear: To return a memory to a non-programmed state, usually represented as "0" or OFF (empty).

clock: A pulse generator which synchronizes the timing of various logic circuits and memory in the Processor.

clock rate: The speed (frequency) at which the Processor operates, as determined by the rate at which words or bits are transferred through internal logic sequences.

CMOS: Complementary metal oxide semiconductor circuitry. An integrated circuit family which has low power consumption.

coding: The preparation of a set of instructions or symbols which, when used by a programmable controller, have a special external meaning.

computer interface: A device designed for data communication between a central computer and another unit such as a PC Processor.

contact symbology diagram: Commonly referred to as a ladder diagram, it expresses the user-programmed logic of the controller in relay-equivalent symbology.

core memory: A device used to store information in ferrite cores. Each may be magnetized in either polarity, which are represented by a logical "1" or "0." This memory is non-volatile; the contents of the memory are retained while power is off.

counter: In relay-panel hardware, an electro-mechanical device which can be wired and preset to control other devices according to the total cycles of one ON and OFF function. In PC, a counter is internal to the Processor; i.e., it is controlled by a user-programmed instruction. A counter instruction has greater capability than any hardware counter. Therefore, PC applications do not require hardware counters.

CRT terminal: A terminal containing a cathode ray tube to display programs as ladder diagrams which use instruction symbols similar to relay characters. A CRT terminal can also display data lists and application reports.

cursor: A visual movable pointer used on a CRT by the PC programmer to indicate where an instruction is to be added to the PC program. The cursor is also used during editing functions.

D

data link: Equipment, especially transmission cables and interface modules, which permits the transmission of information.

debugging: Process of detecting, locating, and correcting mistakes in hardware (system wiring) or software (program).

diagnostic program: A test program to help isolate hardware malfunctions in the programmable controller and application equipment.

digital: The representation of numerical quantities by means of discrete numbers. It is possible to express in binary digital form all information stored, transferred, or processed by dual-state conditions; e.g., on/off, open/closed, octal, and BCD values.

documentation: An orderly collection of recorded hardware and software data such as tables, listing, diagrams, etc., to provide reference information for PC application operation and maintenance.

downtime: The time when a system is not available for production due to required maintenance.

duplex: Two-way data transmission in two directions. Full-duplex describes two data paths which allow simultaneous data transmission in both directions. Half-duplex describes one data path which allows data transmission in either of two directions, but only one direction at a time.

E

edit: To deliberately modify the user program in the PC memory.

execution: The performance of a specific operation such as would be accomplished through processing one instruction, a series of instructions, or a complete program.

execution time: The total time required for the execution of one specific operation.

F

feedback: The signal or data fed back to the PC from a controlled machine or process to denote its response to the command signal.

filter: Electrical device used to suppress undesirable electrical noise.

firmware: A series of instructions in ROM (read-only memory). These instructions are for internal Processor functions only, and are transparent to the user.

flow chart: A graphical representation for the definition, analysis, or solution of a problem. Symbols are used to represent a process or sequence of decisions and events.

full duplex: A mode of data transmission that is the equivalent of two paths—one in each direction simultaneously.

H

half duplex: A mode of data transmission capable of communicating in one of two directions, but in only one direction at a time.

hard copy: Any form of printed document such as ladder diagram program listing, paper tape, or punched cards.

hardware: The mechanical, electrical and electronic devices which compose a programmable controller and its application.

I

input devices: Devices such as limit switches, pressure switches, push buttons, etc., that supply data to a programmable controller. These discrete inputs are two types: those with common return, and those with individual returns (referred to as isolated inputs). Other inputs include analog devices and digital encoders.

instruction: A command or order that will cause a PC to perform one certain prescribed operation.

interfacing: Interconnecting a PC with its application devices, and data terminals through various modules and cables. Interface modules convert PC logic levels into external signal levels, and vice versa.

I/O: Abbreviation for INPUT/OUTPUT.

I/O electrical isolation: Separation of the field wiring circuits from the logic level circuits of the PC, typically done with optical isolation.

I/O module: The printed circuit board that is the termination for field wiring of I/O devices.

I/O rack: A chassis which contains I/O Modules.

I/O scan: The time required for the PC Processor to monitor all inputs and control all outputs. The I/O Scan repeats continuously.

isolated I/O module: A module which has each input or output electrically isolated from every other input or output on that module. That is to say, each input or output has a separate return wire.

L

ladder diagram: An industry standard for representing control logic relay systems.

language: A set of symbols and rules for representing and communicating information (data) among people, or between people and machines.

large scale integration (LSI): Any integrated circuit which has more than 100 equivalent gates manufactured simultaneously on a single slice of semiconductor material.

latching relay: A relay with 2 separate coils, one of which must be energized to change the state of the relay; it will remain in either state without power.

line: In communications, describes cables, telephone lines, etc., over which data is transmitted to and received from the terminal.

line driver: An integrated circuit specifically designed to transmit digital information over long lines—that is, extended distances.

line printer: A high-speed printing device that prints an entire line at one time.

location: A storage position in memory.

logic: A means of solving complex problems through the repeated use of simple functions which define basic concepts. Three basic logic functions are AND, OR, and NOT.

logic diagram: A drawing which represents the logic functions AND, OR, NOT, etc.

logic level: The voltage magnitude associated with signal pulses representing ones and zeroes ("1" and "0") in binary computation.

M

matrix: A two-dimensional array of circuit elements, such as wires, diodes, etc., which can transform a digital code from one type to another.

memory: A grouping of circuit elements which has data storage and retrieval capability.

memory module: A Processor module consisting of memory storage and capable of storing a finite number of words (e.g., 4096 words in a 4K memory module). Storage capacity is usually rounded off and abbreviated with K representing each 1024 words.

micro electronics: Refers to circuits built from miniaturized components and includes integrated circuits.

microprocessor: An electronic computer processor section implemented in relatively few IC chips (typically LSI) which contain: arithmetic, logic, register, control, and memory functions.

microsecond (μs): One millionth of a second: 1 x 10^{-6} or 0.000001 second.

millisecond (ms): One thousandth of a second: 10^{-3} or 0.001 second.

modem: Acronym for data set (MOdulator/DEModulator).

module: An interchangeable "plug-in" item containing electronic components which may be combined with other interchangeable items to form a complete unit.

MOS: Stands for metal-oxide semiconductor. It is a method of manufacturing low current drain transistors and integrated circuits.

multiplexing: The time-shared scanning of a number of data lines into a single channel. Only one data line is enabled at any instant.

N

noise: Extraneous signals; any disturbance which causes interference with the desired signal or operation.

non-volatile memory: A memory that does not lose its information while its power supply is turned off.

O

off-line: Describes equipment or devices which are not connected to the communications line.

on-line: Describes equipment or devices which are connected to the communications line.

on-line operation: Operations where the programmable controller is directly controlling the machine or process.

output: Information transferred from PC through output modules to control output devices.

output devices: Devices such as solenoids, motor starters, etc., that receive data from the programmable controller.

P

parallel operation: Type of information transfer whereby all digits of a word are handled simultaneously.

parallel output: Simultaneous availability of two or more bits, channels, or digits.

parity: A method of verifying the accuracy of recorded data.

parity bit: An additional bit added to a memory word to make the sum of the number of "1's" in a word always "even parity" or "odd parity" (always even or odd).

parity check: A check that tests whether the number of "1's" in an array of binary digits is odd or even.

PC: Abbreviation for Programmable Controller.

peripheral equipment: Units which may communicate with the programmable controller, but are not part of the programmable controller; e.g., Teletype, cassette recorder, CRT terminal, tape reader, etc.

printed circuit: A board on which a predetermined pattern of printed connections has been formed.

processor: A unit in the programmable controller which scans all the inputs and outputs in a predetermined order. The Processor monitors the status of the inputs and outputs in response to the user programmed instructions in memory, and it energizes or de-energizes outputs as a result of the logical comparisons made through these instructions.

program: A sequence of instructions to be executed by the PC Processor to control a machine or process.

program panel: A device for inserting, monitoring, and editing a program in a PC.

program scan: The time required for the PC Processor to execute all instructions in the program once. The program scan repeats continuously. The program monitors inputs and controls outputs through the Input and Output Image Tables.

programmable controller: A solid state control system which has a user programmable memory for storage of instructions to implement specific functions such as: I/O control logic, timing, counting, arithmetic, and data manipulation. A PC consists of central processor, input/output interface, memory, and programming device which typically uses relay-equivalent symbols. PC is purposely designed as an industrial control system which can perform functions equivalent to a relay panel or a wired solid state logic control system.

PROM: Abbreviation for Programmable Read-Only Memory. A digital storage device which can be written into only once but continually read.

protocol: A defined means of establishing criteria for receiving and transmitting data through communication channels.

R

RAM: A Random Access Memory is an addressable LSI device used to store information in microscopic flip-flops or capacitors. Each may be set to an ON or OFF state, representing logical "1" or "0." This type of memory is volatile. That is to say, memory is lost while power is off, unless battery back-up is used.

read: To sense the presence of information in some type of storage, which includes RAM memory, magnetic tape, punched tape, etc.

resolution: A measure of the smallest possible increment of change in the variable output of a device.

ROM: A Read-Only Memory is a digital storage device specified for a single function. Data is loaded permanently into the ROM when it is manufactured. This data is available whenever the ROM address lines are scanned.

rung: A grouping of PC instructions which controls one output. This is represented as one section of a logic ladder diagram.

S

scan time: The time necessary to completely execute the entire PC program one time.

self-diagnostic: The hardware and firmware within a controller which allows it to continuously monitor its own status and indicate any fault which might occur within it.

serial operation: Type of information transfer within a programmable controller whereby the bits are handled sequentially rather than simultaneously, as they are in parallel operation. Serial operation is slower than parallel operation for equivalent clock rate. However, only one channel is required for Serial operation.

shift register: A program, entered by the user into the memory of a programmable controller, in which the information data (usually single bits) are shifted one or more positions on a continual basis. There are two types of shift registers: asynchronous and synchronous.

significant digit: A digit that contributes to the precision of a number. The number of significant digits is counted beginning with the digit contributing the most value, called the Most Significant Digit, and ending with the one contributing the least value, called the Least Significant Digit.

software: The user program which controls the operation of a programmable controller.

solid state devices (semiconductors): Electronic components that control electron flow through solid materials such as crystals; e.g., transistors, diodes, integrated circuits.

special purpose logic: Proprietary features of a programmable controller which allow it to perform logic not normally found in relay ladder logic.

start-up: The time between equipment installation and the full operation of the system.

state: The logic "0" or "1" condition in PC memory or at a circuit's input or output.

storage: Synonymous with Memory.

strip printer: A peripheral device used with a PC to provide a hardcopy of process numbers, status and functions.

surge: A transient variation in the current and/or potential at a point in the circuit.

synchronous shift register: Shift register which uses a clock for timing of a system operation and where only one state change per clock pulse occurs.

system: A collection of units combined to work as a larger integrated unit having the capabilities of all the separate units.

T

tape reader: A unit which is capable of sensing data from punched tape.

Teletype: A peripheral electromechanical device for inserting or recording a program into or from a PC memory in either a punched paper tape or printed ladder diagram format.

termination: 1) The load connected to the output end of a transmission line. 2) The provisions for ending a transmission line and connecting to a bus bar or other terminating device.

thumbwheel switch: A rotating numeric switch used to input numeric information to a controller.

timer: In relay-panel hardware, an electromechanical device which can be wired and preset to control the operating interval of other devices. In PC, a timer is internal to the Processor, which is to say it is controlled by a user-programmed instruction. A timer instruction has greater capability than any hardware timer. Therefore, PC applications do not require hardware timers.

transducer: A device used to convert physical parameters, such as temperature, pressure, weight, into electrical signals.

translator package: A computer program which allows a user program (in binary) to be converted into a usable form for computer manipulation.

truth table: A matrix which describes a logic function by listing all possible combinations of inputs, and by indicating the outputs for each combination.

TTL: Abbreviation for Transistor/Transistor Logic. A family of integrated circuit logic. (Usually 5 volts is high or "1" and 0 volts is low or "0"; 5V = 1, 0V = 0.)

TTY: An abbreviation of Teletype.

U

UV erasable PROM: An ultraviolet erasable PROM is a programmable read-only memory which can be cleared (set to "0") by exposure to intense ultraviolet light. After being cleared, it may be reprogrammed.

V

volatile memory: A memory that loses its information if the power is removed from it.

W

weighted value: The numerical value assigned to any single bit as a function of its position in the code word.

word: A grouping or a number of bits in a sequence that is treated as a unit and is stored in one memory location.

word length: The number of bits in a word; in PC literature these are generally only data bits. One PC word = 16 data bits.

write: The process of loading information into memory.

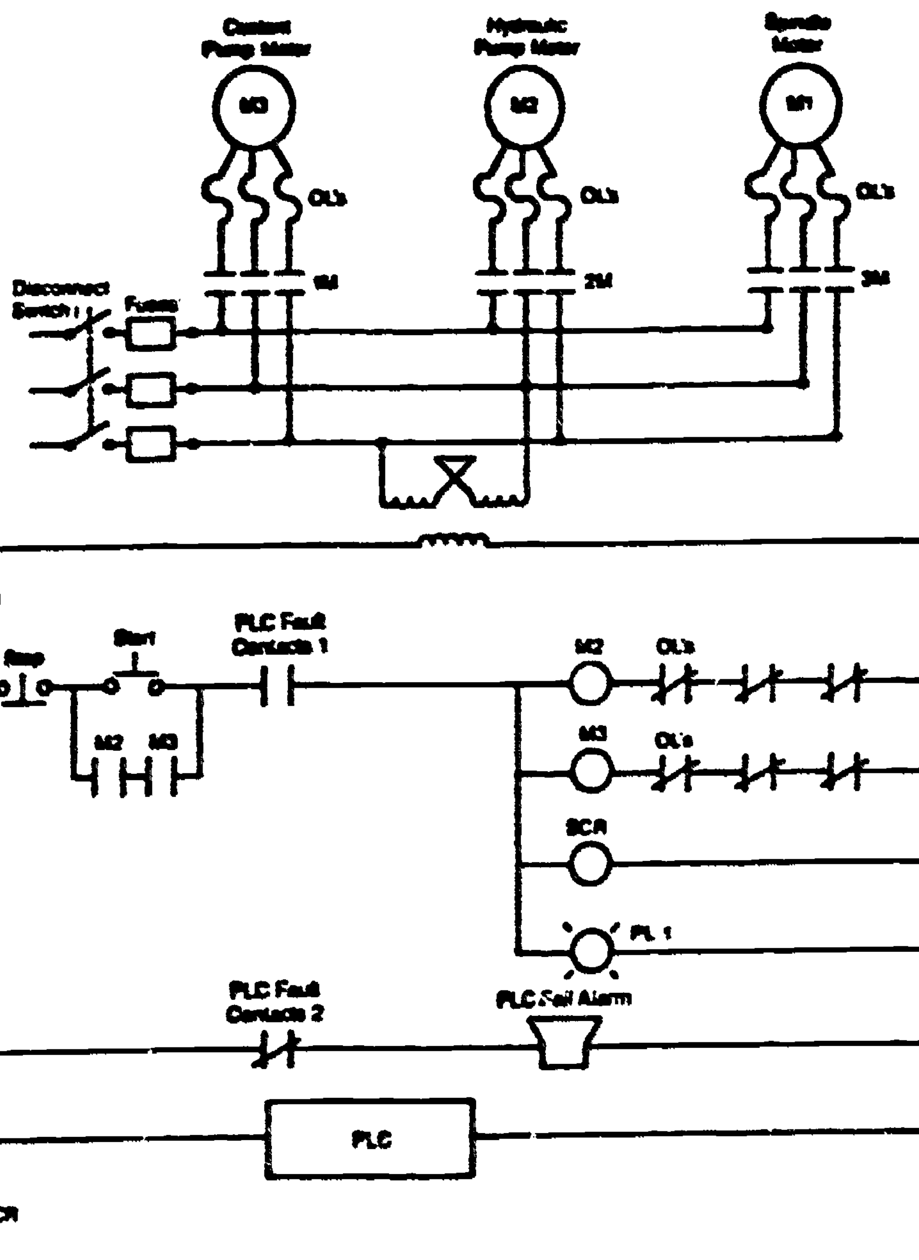
IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION

- Portions to leave hardwired
  - Hardwired elements remain part of the control logic
- Should include:
  - Elements not frequently switched off after the start
    - Compressors
    - Hydraulic pumps
    - Other motors
  - Elements required for safety purposes
    - Emergency stops
    - Master start pushbuttons
    - Alarms

IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION



IMPLEMENTING AND PROGRAMMING THE PLC SYSTEM

PROGRAMMING ORGANIZATION

- Special Cases
  - Programming NO wired devices as NC (and vice versa)
  - MCR

PLC SYSTEM DOCUMENTATION

PLC SYSTEM DOCUMENTATION

- SYSTEM CONFIGURATION
- I/O WIRING CONNECTION DIAGRAM
- INPUT/OUTPUT ADDRESS ASSIGNMENT
- INTERNAL STORAGE ADDRESS ASSIGNMENTS
- STORAGE REGISTER ASSIGNMENTS
- CONTROL PROGRAM PRINTOUT

PLC SYSTEM DOCUMENTATION

RECORDED INFORMATION CONCERNING:

1) Operation of machine or process
2) Hardware components of control system
3) Software components of control system

RECORDS USED DURING:

- System Design
- Installation
- Start-Up
- Debugging
- Maintenance

RECORDS PROVIDE:

1) Communication tool
2) Ability to diagnose and modify later
3) Training material for operators and maintenance
4) Ability to reproduce the system

PLC SYSTEM DOCUMENTATION

SYSTEM ABSTRACT

- Clear statement of control problem or task

- Description of the design strategy or philosophy
    - Define function of major hardware and software components and why selected

- Statement of objectives
    - Will allow the user to measure the success of the control implementation

Example:

A single CPU, located in an enclosure adjacent to the machine, will control a pneumatic twelve position index table and the functional equipment at each of the twelve stations. Local DC input and output modules will be used to interface the field devices with the CPU.

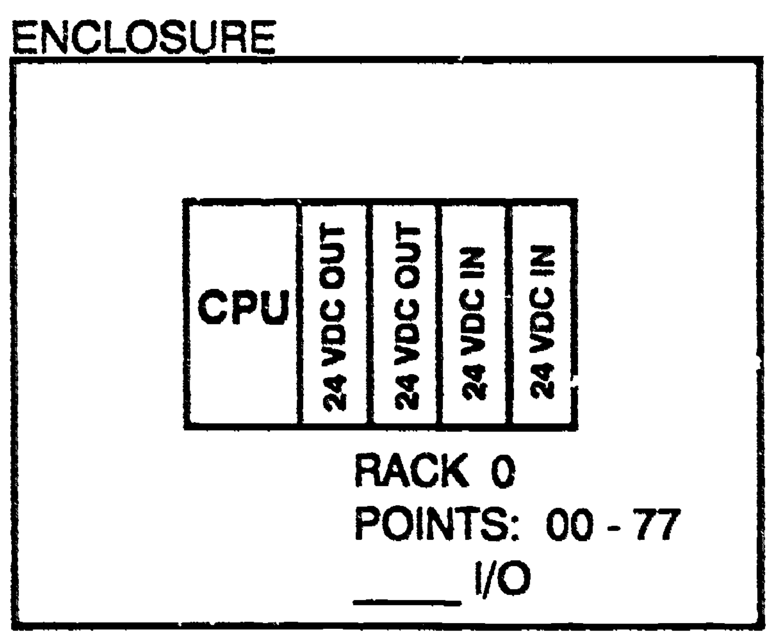
PLC SYSTEM DOCUMENTATION

SYSTEM CONFIGURATION

Diagram which shows:

- Physical location of subsystems

- Designation of I/O rack address assignments

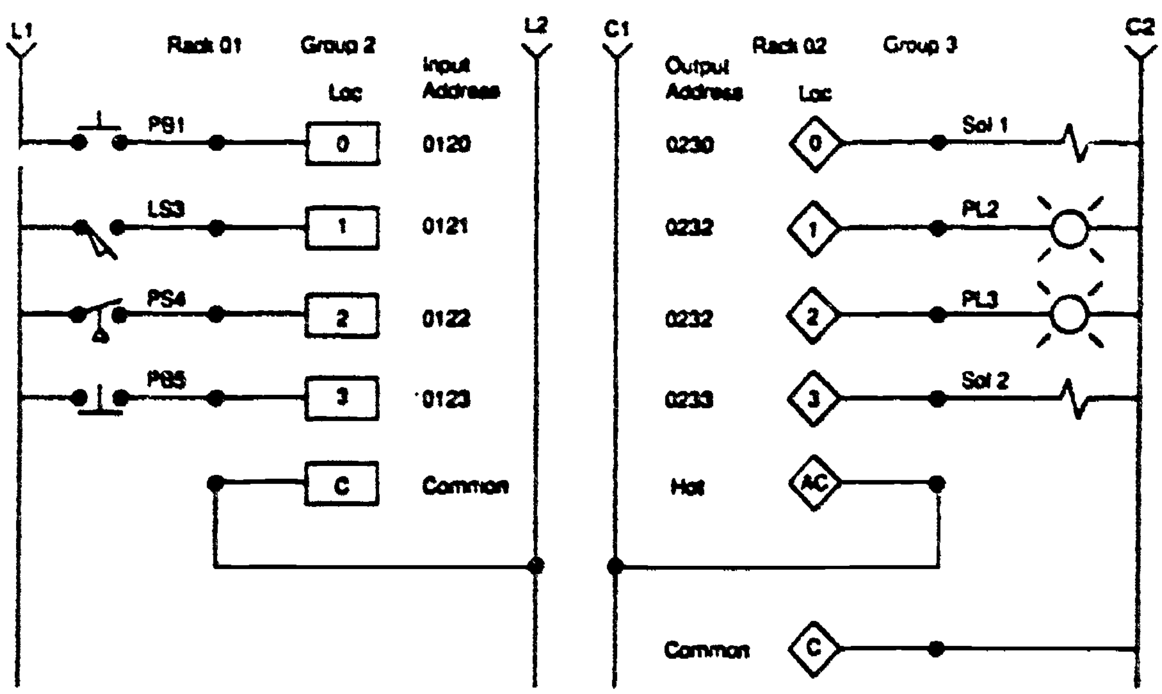


PLC SYSTEM DOCUMENTATION

I/O WIRING CONNECTION DIAGRAM

- Shows connections of field devices to PLC
  - May include power supplies and subsystems
  - Shows rack, group and module locations

- Show TB numbers if field device not directly wired to I/O module



PLC SYSTEM DOCUMENTATION

INPUT/OUTPUT ADDRESS ASSIGNMENT

- Table organized by I/O address
- Shows I/O type, device type, and device function
- Show both used and unused addresses

Address	I/O Type	Device	Function
0120	115 VAC In	PB	Start Push Button PB 1
0121	115 VAC In	LS	Up Limit #2
0122	115 VAC In	PS	Hydraulic Pressure OK
0123	115 VAC In	PB (NC)	Reset PB 2
•	•	•	•
•	•	•	•
•	•	•	•
0230	24 VAC Out	Sol	Retract #1
0231	24 VAC Out	PL	#2 in position
0232	24 VAC Out	PL	Running
0233	24 VAC Out	Sol	Fast Up #3

PLC SYSTEM DOCUMENTATION

INTERNAL STORAGE ADDRESS ASSIGNMENTS

- Internals not associated directly with field devices
  - Programming timers, counters
  - Replacement of control relays
- Tendency is to use them freely without accounting for usage
  - Good documentation will simplify modifications
  - Show both used and unused assignments

Internal	Type	Description
1000	Coil	Used to Latch Position
1001	Coil	Set-up Instantaneous Timer Contact
1002	Compare	Used for CMP Equal
1003	Add	Addition Positive
•	•	•
•	•	•
•	•	•
T100	Timer	Time on delay—Motor 1
C400	Counter	Count pieces on Conv. #1
•	•	•
•	•	•
•	•	•

PLC SYSTEM DOCUMENTATION

STORAGE REGISTER ASSIGNMENTS

- Consist of user storage and I/O registers
- Registers store information from address assignment plus comparison values and presets
- Table should show if register is used or not

Register	Contents	Description
3036	Temperature In	I/O Register with Analog Module
3040	Temperature In	I/O Register with Analog Module
4000	1200	20 sec preset of TDR3
4001	2000	Count preset for CMP =
4002	5000	Count preset for CMP >
•	•	
•	•	
•	•	
4100	0	Not Used
to	•	•
4200	•	•
	0	•

PLC SYSTEM DOCUMENTATION

CONTROL PROGRAM PRINTOUT

- Hard copy of control logic program in CPU memory
- Not adequate by itself
- Some manufacturers software provide additional documentation capabilities

INSTALLATION PRACTICES AND TROUBLESHOOTING

INSTALLATION PRACTICES

- PLC Enclosures
- Electrical Noise
- Leaky Inputs and Outputs
- Grounding
- Voltage Variations and Surges

PLC INSTALLATION PRACTICES

PLC ENCLOSURES

- NEMA 12 for most solid-state control devices
  - general opurpose areas
  - dust-tight
- Metal minimizes EMR generated by surrounding equipment
- Heat Dissipation Factors
  - generated by power supply. local I/O, CPU
  - proper spacing of components usually sufficient
  - fan or blower for high ambient temperature

PLC INSTALLATION PRACTICES

ELECTRICAL NOISE

- Enters through input, output or power supply lines
  - coupling of an electrostatic field
  - electromagnetic induction (EMI)

- Results in temporary occurrences of operating errors
  - may result in hazardous machine operation

Prevention

- Locate controller away from noise generating devices
  - large AC motors
  - high frequency welders
  - relays, solenoids, motor starters (especially when operated by hard contacts)

- Careful wire routing
  - separate input power routing from I/O wiring
  - never run signal and power in the same conduit
  - segregate I/O wiring by signal type

PLC INSTALLATION PRACTICES

LEAKY INPUTS AND OUTPUTS

- Solid State devices will have a small leakage current even when in the OFF state
  - Inputs: proximity switches
    SCR
    triac
    transistor
  - Outputs: high impedance output load devices

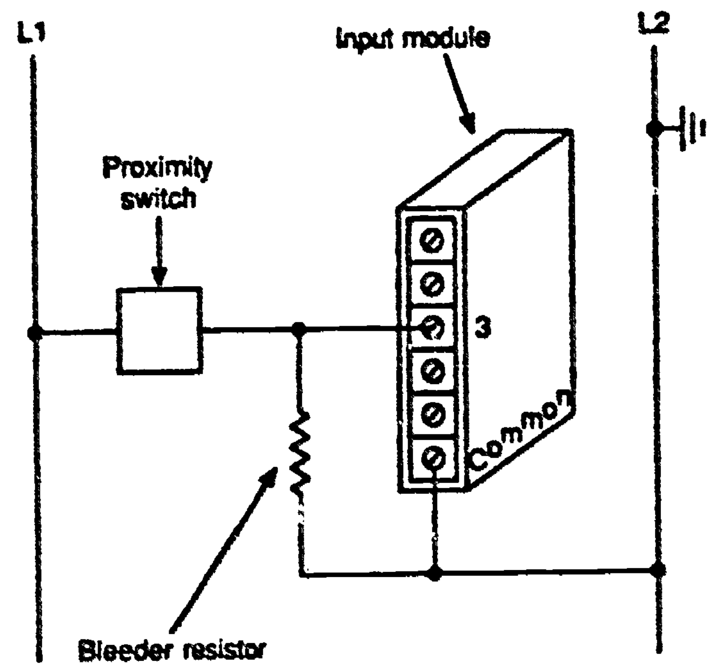
  Solution: Use a bleeder resistor connected in parallel with the device

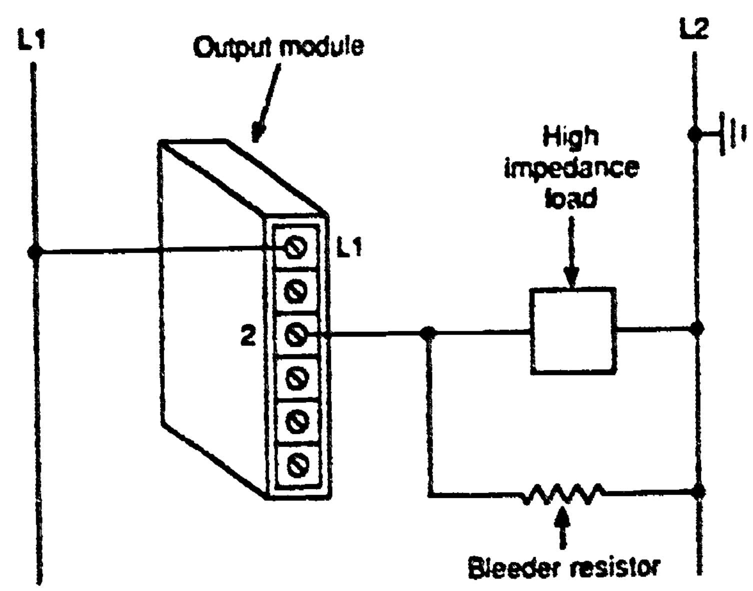
- Results in temporary occurrences of operating errors
  - may result in hazardous machine operation

Prevention

- Locate controller away from noise generating devices
  - large AC motors
  - high frequency welders
  - relays, solenoids, motor starters (especially when operated by hard contacts)

- Careful wire routing
  - separate input power routing from I/O wiring
  - never run signal and power in the same conduit
  - segregate I/O wiring by signal type


L1
Input module
L2
Proximity switch
3
Common
Bleeder resistor


L1
Output module
L2
High impedance load
L1
2
Bleeder resistor

PLC INSTALLATION PRACTICES

GROUNDING

- Safety: National Electrical Code for conductors, color codes, connections

  - path must be permanent (no solder)

  - path must safely conduct ground fault current

  - must ground: controller and its enclosure
    all controlled devices

- Noise Reduction
  - everything inside controller should be individually grounded to central point on enclosure frame

  - power and ground should be separated at point of entry to enclosure

  - all ground connections to enclosure should be made with star washers

  - paint should be scraped off

  - No. 12 AWG stranded for PLC (minimum)
    No. 8 AWG stranded for enclosure backplane

PLC INSTALLATION PRACTICES

VOLTAGE VARIATIONS AND SURGES

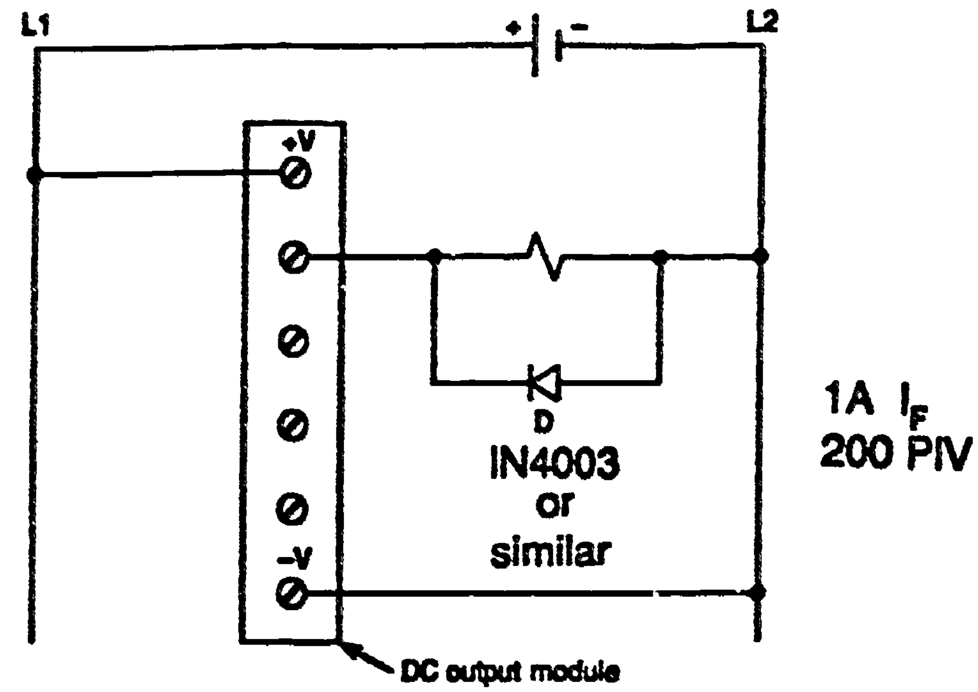
- Build-in Protection
  - power supply built to withstand fluctuations

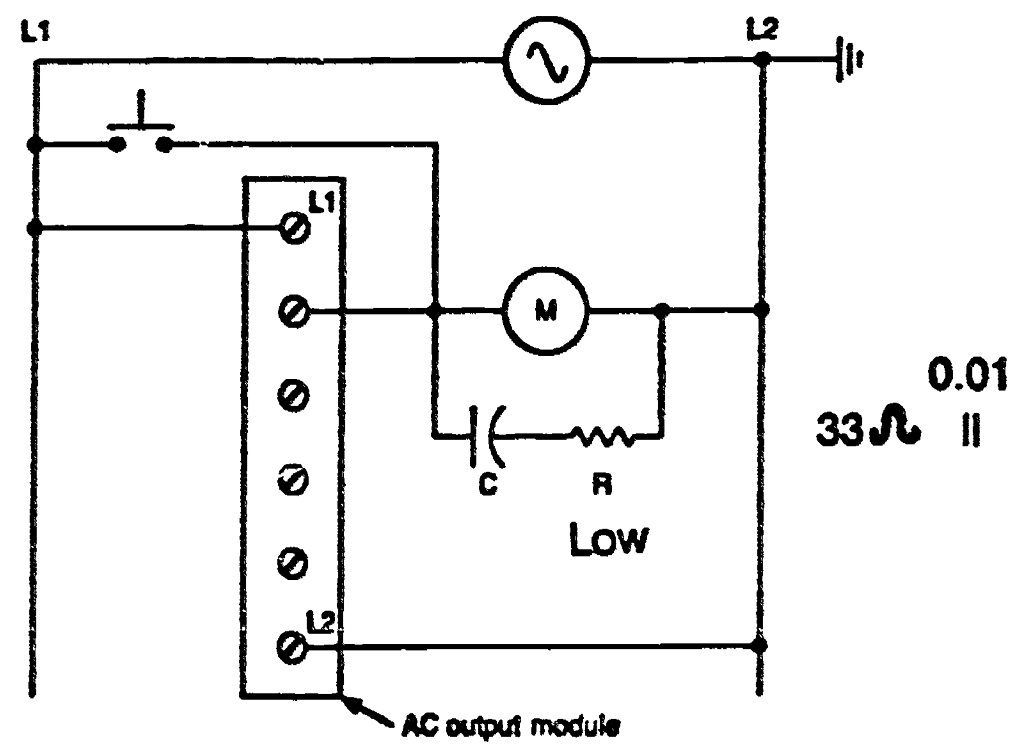
- Extra Protection
  - constant voltage transformer for excessive line voltage variations

  - suppression networks

    limit voltage spikes produced when current in an inductive load is interrupted or turned off

    limit rate of change of current through the inductor


L1
L2
+
-
+V
-V
D
IN4003
or
similar
1A I_F
200 PIV
DC output module


L1
L2
L1
M
C
R
Low
33Ω
0.01
L2
AC output module

TROUBLESHOOTING

- Procesor Module
- Input Malfunctions
- Output Malfunctions

TROUBLESHOOTING

Identify the problem and its source. Source can generally be narrowed to:

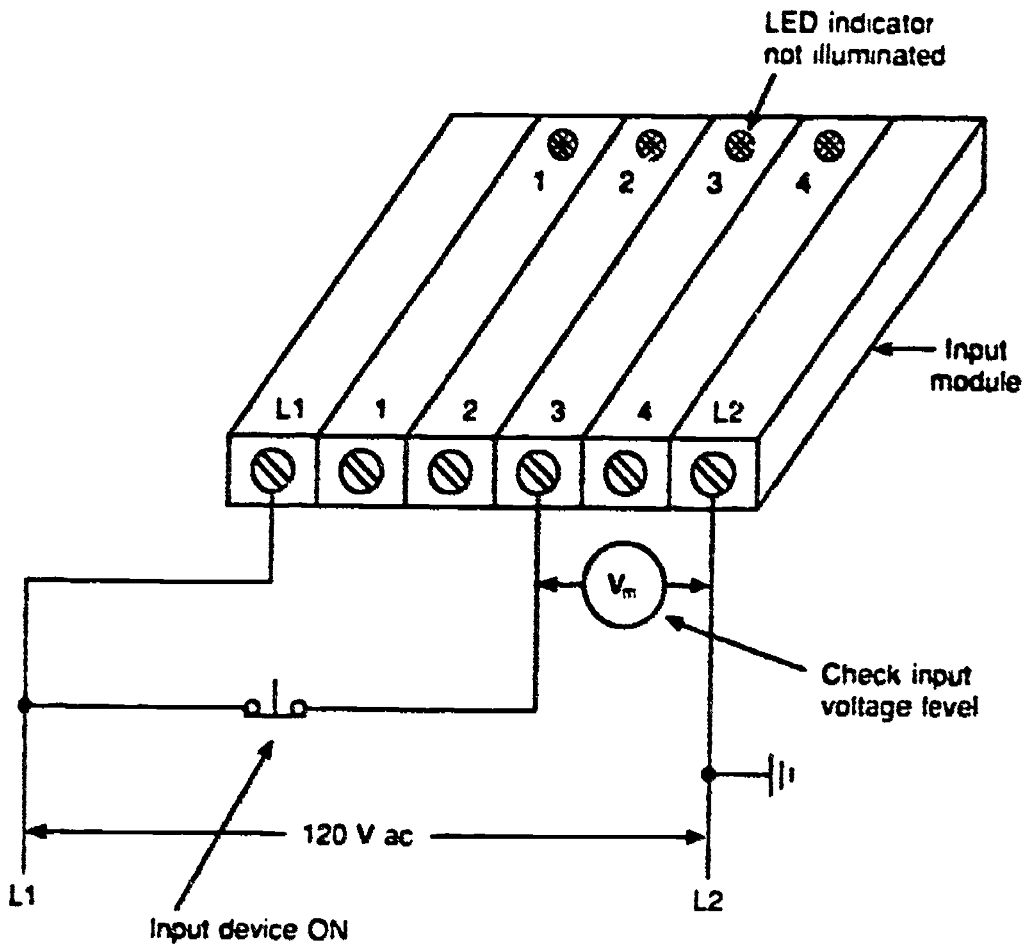
- Processor module
- I/O hardware
- Wiring
- Machine inputs or outputs

Tools:

- Status Indicators
  - on I/O module
  - programming device monitor
- Voltmeter
- Self-detection
- Watchdog timer

Input device troubleshooting guide				
Input device condition	Input module status indicator	Operator terminal status indicator		Possible problem source(s)
Closed – ON	ON	TRUE Dark	FALSE Normal	None, correct status indication.
Open – OFF	OFF	FALSE Normal	TRUE Dark	None, correct status indication.
Closed – ON	ON	FALSE Normal	TRUE Dark	1. I/O module. 2. Processor/operator terminal communication.
Closed – ON	OFF	FALSE Normal	TRUE Dark	1. Wiring/power to I/O module. 2. I/O module.
Open – OFF	OFF	TRUE Dark	FALSE Normal	1. Programming error. 2. Processor/operator terminal communication.
Open – OFF	ON	TRUE Dark	FALSE Normal	1. Short circuit in input device or wiring. 2. Input module.

Output device troubleshooting guide			
Output device condition	Output module status indicator	Operator terminal status indicator	Possible problem source
Energized – ON	ON	TRUE Dark	None, correct status indication.
De-energized – OFF	OFF	FALSE Normal	None, correct status indication.
De-energized – OFF	ON	TRUE Dark	1. Wiring to output device. 2. Output device.
De-energized – OFF	OFF	TRUE Dark	1. Blown fuse — output module. 2. Output module malfunction.


LED indicator
not illuminated
1
2
3
4
Input
module
L1
1
2
3
4
L2
V_m
Check input
voltage level
120 V ac
L1
L2
Input device ON

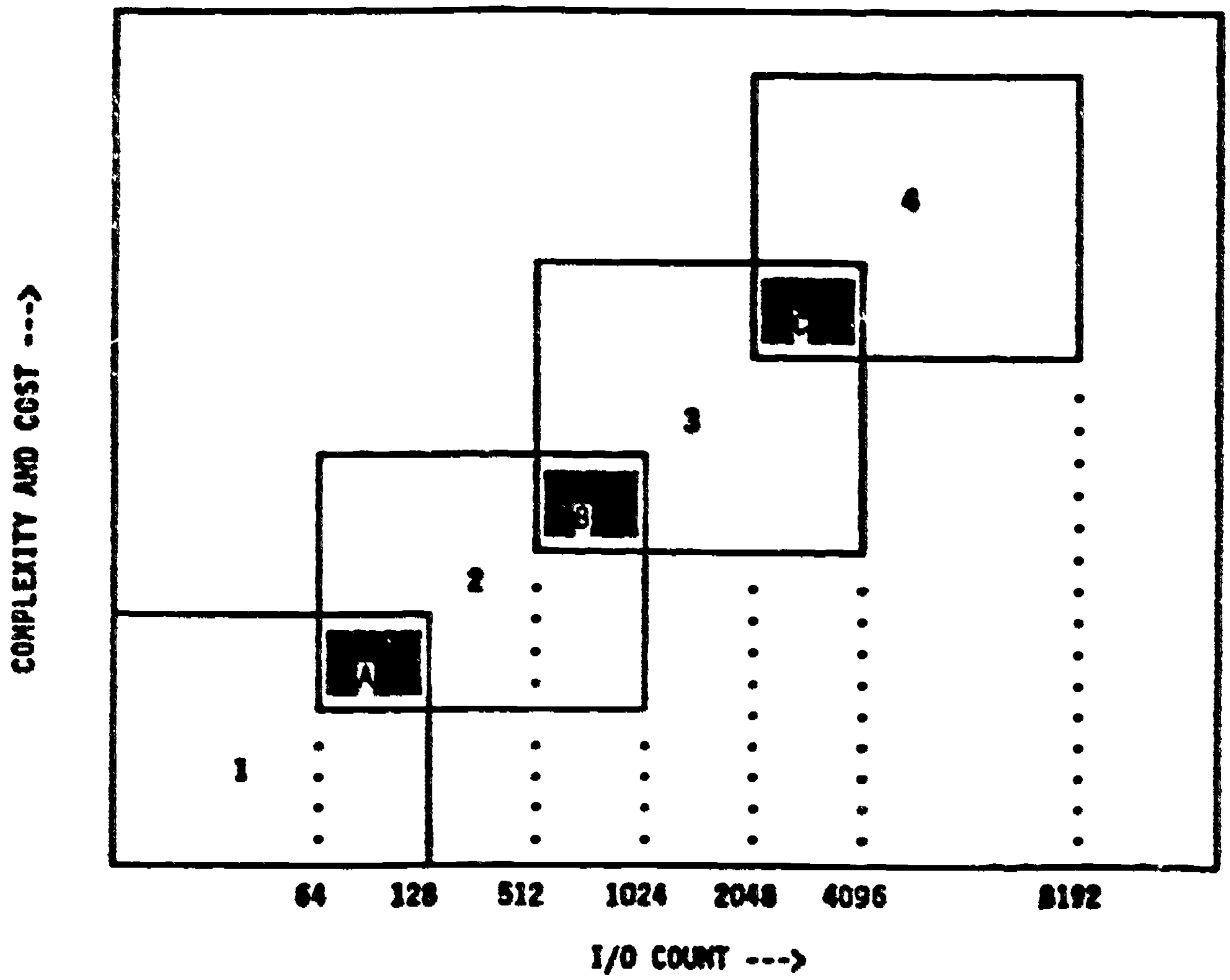
PLC SYSTEM SELECTION GUIDELINES

PLC SIZES
AND
SCOPE OF APPLICATIONS

PLC SIZE AND APPLICATION

	I/O	MEMORY	CAPABILITY
SMALL	≤128	256-2048 bytes usually CMOS RAM with battery backup	-Simple to Advanced -Discrete
MEDIUM	128 - 2048	32 K Bytes usually CMOS RAM with battery backup	-Simple to Advanced -Discrete and Analog -Data Communication RS-232 or 20 mA loop
LARGE	up to 8192	750 K Bytes	Individual processes to entire plants -Master/slave -Servo drive -Communications

PLC SIZE AND APPLICATION



PROCESS CONTROL SYSTEM DEFINITION

PROCESS CONTROL SYSTEM DEFINITION

ITEMS TO EVALUATE AND DEFINE:

- Input / Output
- Type of Control
- Memory
- Software
- Peripherals
- Physical Area and Environment

PROCESS CONTROL SYSTEM DEFINITION

INPUT/OUTPUT CONSIDERATIONS

- Count devices to be monitored or controlled
    - analog and/or digital devices
    - future expansion
    - spare points (typically 10% to 20%)
- Look for desirable characteristics
    - debounce circuitry
    - surge protection
    - isolation
    - fused circuits and blown fuse indicators
    - special function and remote I/O

PROCESS CONTROL SYSTEM DEFINITION

CONTROL SYSTEM ORGANIZATION

What type of control should you use?

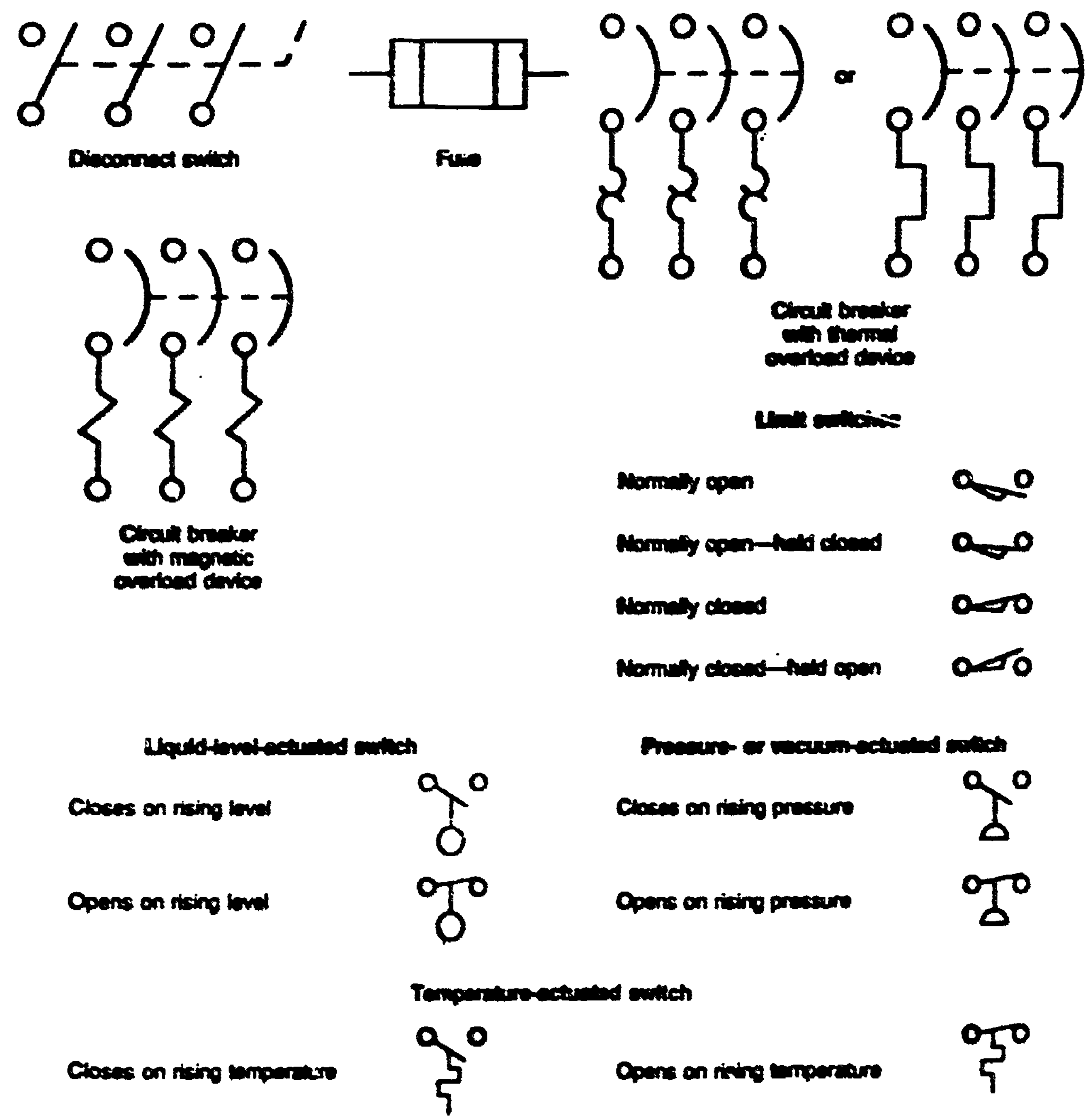
- Number of machines or processes?
- Require communication to other controllers or computers?

Memory Considerations
- type (volatile, non-volatile)
- amount

Software Considerations
- programming languages
- instruction set

Physical and Environmental
- ambient conditions

Typical graphic symbols for electrical diagrams



Contacts

Normally open 

Normally closed 

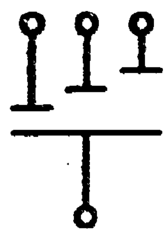
Normally open with time delay closing (TC)

Normally open with time delay opening (TO)

Normally closed with time delay opening (TO)

Normally closed with time delay closing (TC)

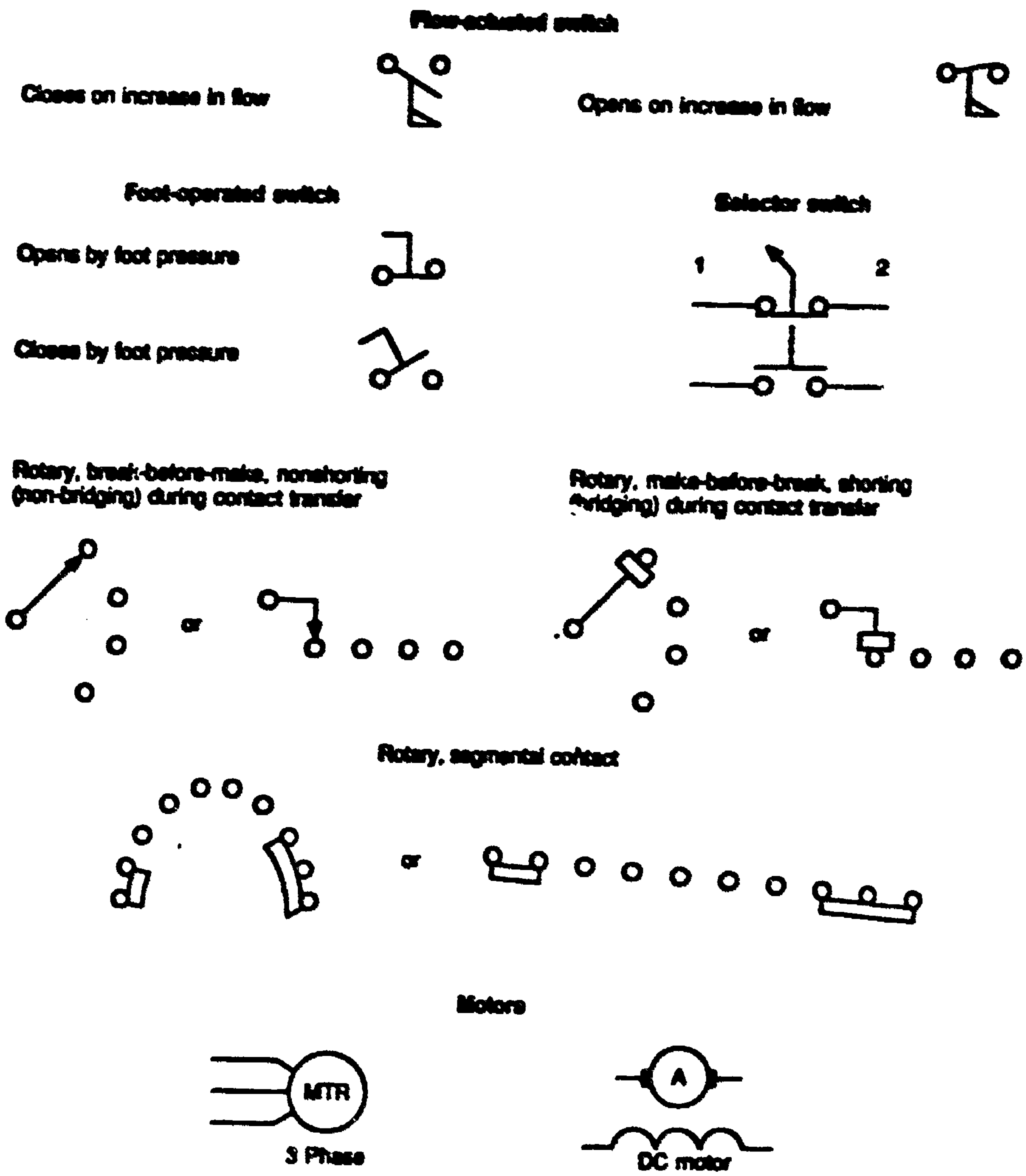
Time sequential closing



Static relay (proximity switch)

Normally ON 

Normally OFF 



Flow-actuated switch
Closes on increase in flow
Opens on increase in flow
Foot-operated switch
Opens by foot pressure
Closes by foot pressure
Selector switch
1
2
Rotary, break-before-make, nonshorting (non-bridging) during contact transfer
or
Rotary, make-before-break, shorting (bridging) during contact transfer
or
Rotary, segmental contact
or
Motors
MTR
3 Phase
A
DC motor

PROCESS CONTROL SYSTEM DEFINITION

CHECKLISTS:

I/O System:	I/O Count Digital I/O Analog I/O Remote I/O Special I/O Physical
CPU:	Processor Memory Power Supply Environment
Software:	Language Software Coils Math Data Handling
Programmer:	CRT or Computer Manual Storage Devices
Diagnostics:	Power Supply Memory Processor Communication Fault Indications