DOCUMENT RESUME

ED 347 371 CE 061 616

AUTHOR Lipsky, Kevin
TITLE Programming Programmable Logic Controller. High-Technology Training Module.
INSTITUTION Wisconsin Indianhead Technical Coll., New Richmond.
SPONS AGENCY Office of Vocational and Adult Education (ED), Washington, DC.
PUB DATE May 90
CONTRACT V199A90151
NOTE 23p.; Developed as part of the High-Technology Training Model for Rural Based Business and Industry, Technical Colleges, and Local and State Educational Agencies.
PUB TYPE Guides - Classroom Use - Teaching Guides (For Teacher) (052)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Electromechanical Aids; *Electromechanical Technology; *Equipment Maintenance; *Learning Modules; *Programing; Technological Advancement; Two Year Colleges
IDENTIFIERS Packaging; *Programable Logic Control

ABSTRACT

This training module on programming programmable logic controllers (PLC) is part of the memory structure and programming unit used in a packaging systems equipment control course. In the course, students assemble, install, maintain, and repair industrial machinery used in industry. The module contains description, objectives, content outline, methodology, activities, five resources (equipment, supplies, and printed materials), and evaluation--written test and program creation. The content outline indicates the following units: functions of PLCs, programming documentation, and developing application programs. The following materials are included with this module: pretest for programming a PLC; posttest for part 1 of programming a PLC; instructor evaluation of program development; Programmable Controller (PC) ladder logic diagram; PC instruction summary; instructions on writing the user program; and other diagrams and overhead transparencies used as teaching aids. (NLA)

High-Technology Training Module

Module Title: PROGRAMMING PROGRAMMABLE LOGIC CONTROLLER

Unit: MEMORY STRUCTURE AND PROGRAMMING

Course: PACKAGING SYSTEMS EQUIPMENT CONTROL

Grade Level (s): 2 YEAR VOCATIONAL DIPLOMA

Developed by: KEVIN LIPSKY

Date: MAY 1990

School: WISCONSIN INDIANHEAD TECHNICAL COLLEGE

1019 SOUTH KNOWLES AVE., NEW RICHMOND, WI 54017

Developed as a part of the High-Technology Training Model for Rural Based Business and Industry, Technical Colleges and Local and State Educational Agencies under Grant No. V199A90151.

TITLE: PROGRAMMING PROGRAMMABLE LOGIC CONTROLLER

SCHOOL: WISCONSIN INDIANHEAD TECHNICAL COLLEGE - KEVIN LIPSKY

DESCRIPTION:

This module is designed to be a part of the course, Packaging Systems Equipment Control. This course is in the fourth semester of a two year vocational diploma program that offers students an opportunity to assemble, install, maintain, and repair industrial machinery used in industry. Reading prints and schematics, AC/DC circuits, digital logic and electromechanical componentry courses have been taken previously to aid in the understanding of machine control. This module is used for the development of programs and the importance of proper documentation for referencing.

OBJECTIVES:

This module is for students in a lab setting using trainers and machines to gain knowledge of machine operation using PLC's. Upon completion of this module, the student will be able to:

1. Identify functions of machine control and PLC's with 100% accuracy.

2. Perform conversion of relay ladder logic to programming language using common standard symbols with 100% accuracy.

3. Develop a PLC program for a desired outcome using verbal and written descriptions of control with the steps used to create the program outlined and evaluated for each of the steps with 100% accuracy.

4. Enter programs using programming devices common to most PLC's with 100% accuracy.

CONTENT OUTLINE

I. FUNCTIONS OF PLC'S

A. Flow of Control

B. Timing and Counting

C. Relay Logic

D. Arithmetic

E. Data Manipulation

F. Data Transfer

II. PROGRAMMING DOCUMENTATION

A. Programming Conversions

1. Ladder logic - basic to all PLC languages
2. Digital logic - codes and terms of memory maps
3. Sequence of control - logical flow
4. Input and output addressing
5. Line or rung descriptions

B. Programming Languages

1. Ladder diagrams
2. Boolean algebra
3. Function blocks
4. English statements

III. DEVELOPING APPLICATION PROGRAMS

A. Analysis of Machine Operation

1. Define the problem
2. Sketch the machine
3. List sequence of steps
4. Identify controls
5. List inputs and outputs
6. Develop PLC ladder diagram

B. Enter Into Memory Using Programming Device

C. Program Revisions or Modifications

METHODOLOGY

1. Obtain additional materials from PLC manufacturer with operations and self teach manuals.

2. Prepare overheads of machines as examples of control requirements and machine functions.

3. Lecture on the functions of PLC's.

4. Distribute handout of language examples and conversion of symbols from each language.

5. Classroom discussion using a sequence of steps used to create an application program.

6. Give an example of a control problem and supervise the creation of a program using small groups.

7. Evaluate level complexity and mastery of entering program into memory of PLC.

ACTIVITIES

1. The students will discuss machine functions and perform conversions from diagrams to PLC languages.

2. The creation of a program by each student given a control problem and steps used to analyze the control needs.

3. The documentation of related addresses from inputs and outputs using numbering systems from PLC's and worksheets found in operations manuals as a prerequisite to developing a PLC ladder diagram.

4. The development of the ladder diagram and the entering into the memory of the PLC with a demonstration of the accomplished results.

5. Revisions for the program or the effect of variables demonstrated for each example of control needs.

RESOURCES

Programmable Controllers Principles and Applications, John Weber, 1988, Merrill Publishers.

Electrical Controls for Machines, Rexnord, 1988, Delmar Publishers.

Programmable Logic Controllers On Packaging Machines, C. Glen Davis, 1988, Packaging Machinery Manufacturers Institute.

PLC with programmer and input/output simulator

Graph paper

EVALUATION

The module will have two evaluations. The first will be a written test to cover the functions of PLC and symbols used in diagrams and the documentation requirements of programming. The second evaluation will be on the entering and accomplished results of creating a program using a rating scale for each step performed by each student.

PRE-TEST FOR PROGRAMMING A PC

Answer the following questions with the best answer.

1. A computer program is a set of:

   ___ a. instructions
   ___ b. electronic components
   ___ c. wiring diagrams

2. The operating system program is normally:

   ___ a. installed by the manufacturer
   ___ b. designed for the tasks of the specific machine
   ___ c. either of the above

3. A PC ladder logic diagram is frequently used to program:

   ___ a. operating system programs
   ___ b. applications program
   ___ c. operation system or applications programs

4. The first step in development of a ladder logic program is a:

   ___ a. machine sketch
   ___ b. complete analysis
   ___ c. list of inputs and outputs

5. The computer scans a ladder logic diagram program:

   ___ a. randomly
   ___ b. from the bottom to the top
   ___ c. from the top to the bottom

6. Electronic symbols are used on:

   ___ a. relay ladder diagrams
   ___ b. PC ladder logic diagrams
   ___ c. both relay and PC ladder logic diagrams

7. Input and output numbers are used on:

   ___ a. relay ladder diagrams
   ___ b. PC ladder logic diagrams
   ___ c. both relay and PC ladder logic diagrams

8. Ladder logic diagram list the operational steps in:

   ___ a. sequence
   ___ b. random order
   ___ c. sequence and random order

9. Changes can be made in the PC ladder logic diagram program by:

___ a. the machine operator
___ b. the machine set-up mechanic
___ c. only authorized programmers

10. Small changes in the ladder diagram:

___ a. can cause major problems in machine operation
___ b. can cause only minor problems in machine operation
___ c. can cause major or minor problems in machine operation

11. ____________________ and ____________________ are the basic symbols of the ladder diagram instruction set.

12. Name the six most common function types available in PCs.

13. In ladder diagram programs, all outputs are represented by:

___ a. contact symbols
___ b. coil symbols
___ c. a or b
___ d. none of the above

14. List four programming languages used in PCs.

POST-TEST FOR PART 1 OF PROGRAMMING A PLC

Answer the following questions with the best answer.

1. The program tells the computer what to _______________ and _______________ to do it.

2. The instructions in a computer program must be _______________, _______________, and in the proper _______________.

3. The computer is usually (forgiving, unforgiving).

4. The first step in the development of a Ladder Diagram is a complete _______________.

5. The steps in an analysis for a computer program include:

   a. Define _______________
   b. Make a _______________
   c. List the _______________ of _______________
   d. Identify _______________
   e. List _______________

6. The definition of the problem should include a description of the _______________ that is to be _______________.

7. A photo-electric sensor is shown on a relay ladder diagram as a(n) (symbol, input or output number).

8. A motor starter coil is shown on a PC ladder logic diagram as a (symbol, input or output number).

9. The main ON/OFF power switches are usually operated by the PLC, hard wired components).

10. The ladder diagram shows the st ps in a machine operation in the _______________ in which they will occur.

11. The PC ladder logic diagram is scanned from the _______________ to the _______________.

12. The components on a PC ladder logic diagram are identified by numbers which identify them as _______________ or _______________.

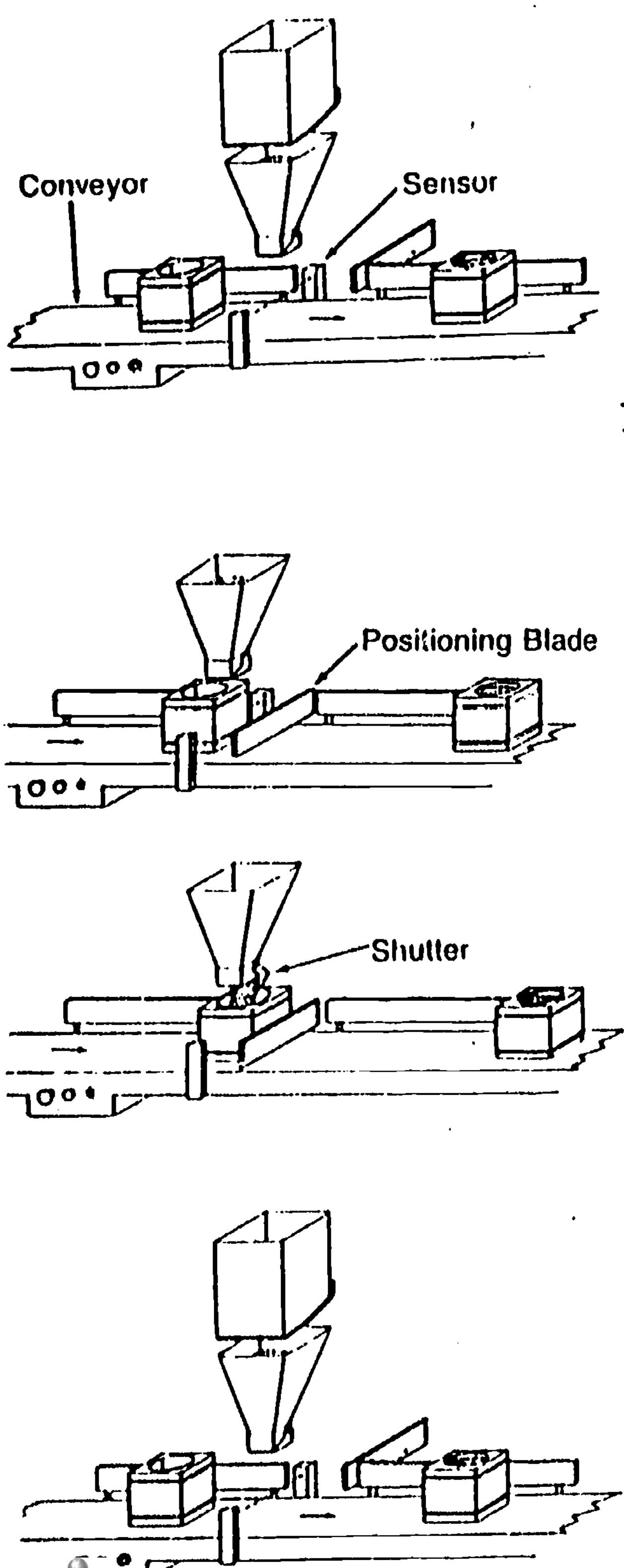
13. The information on the (rungs, side rails) of the ladder specify the conditions that must be met before an operation can take place.

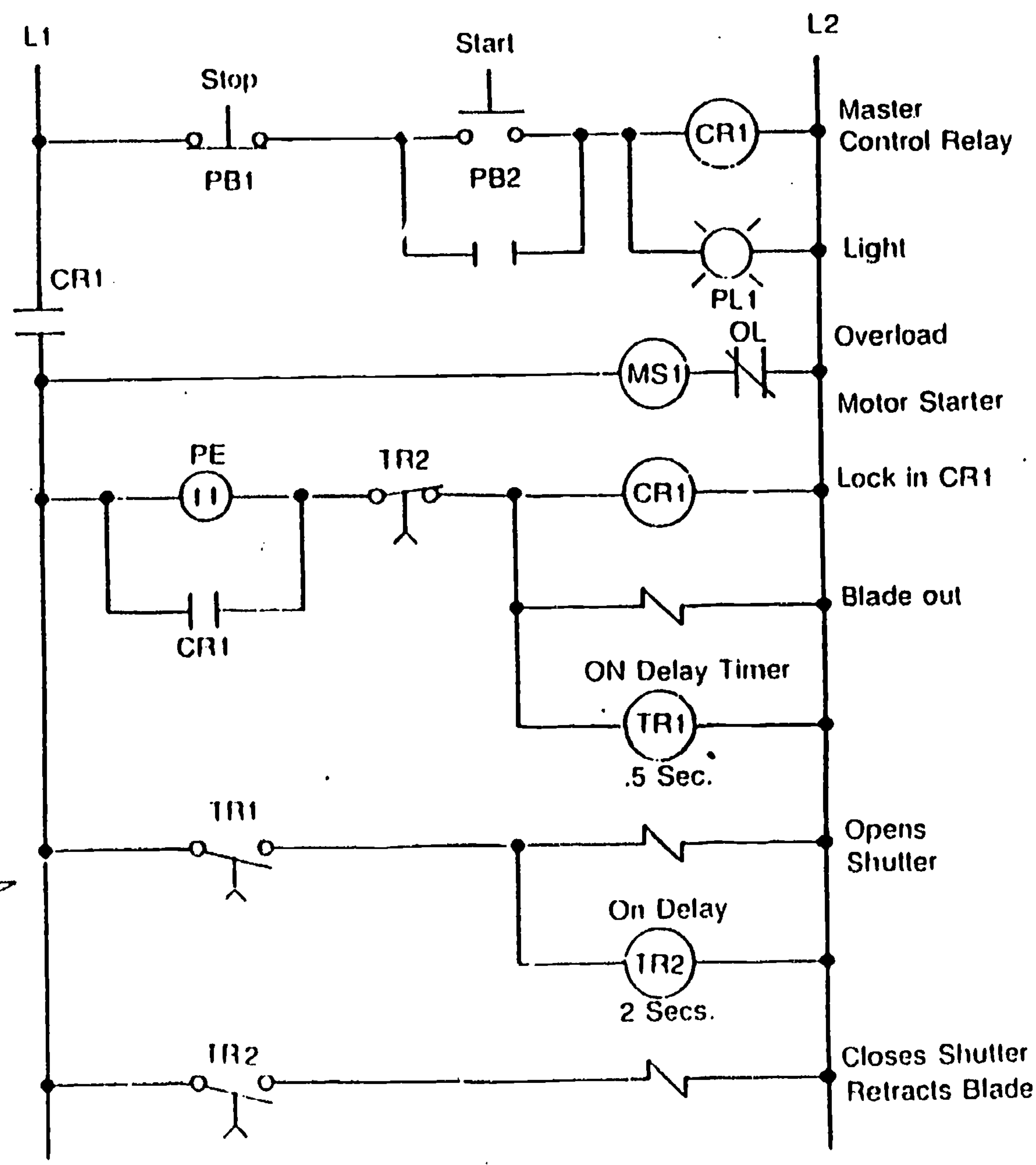
14. The electrical power source is shown by the (rungs, side rails) of the ladder diagram.

15. The PLC can display the PC ladder logic diagram on the CRT _______________ or as a _______________.

5 Points — List five functions of a PLC with an example of each function and how it might be used.

10 Points — Given the following relay ladder diagram of a filling operation and the PLC equivalent program, identify each of the rungs in the PLC program for its relation to the control of the machine.





Relay Ladder Logic Diagram

INSTRUCTOR EVALUATION OF PROGRAM DEVELOPMENT

SECTION I RATING SCALE USED FOR EACH 1 - 5

1. Adaptation to the requirements and limitations of the system used. ______

2. Problem defined with statement of machine operation and any special requirements. ______

3. Sketch of the machine that shows relationship of the systems. ______

4. Listing of the sequence of steps to be performed in order and listing of any variations. ______

5. Inputs and outputs identified for what they do for machine control. ______

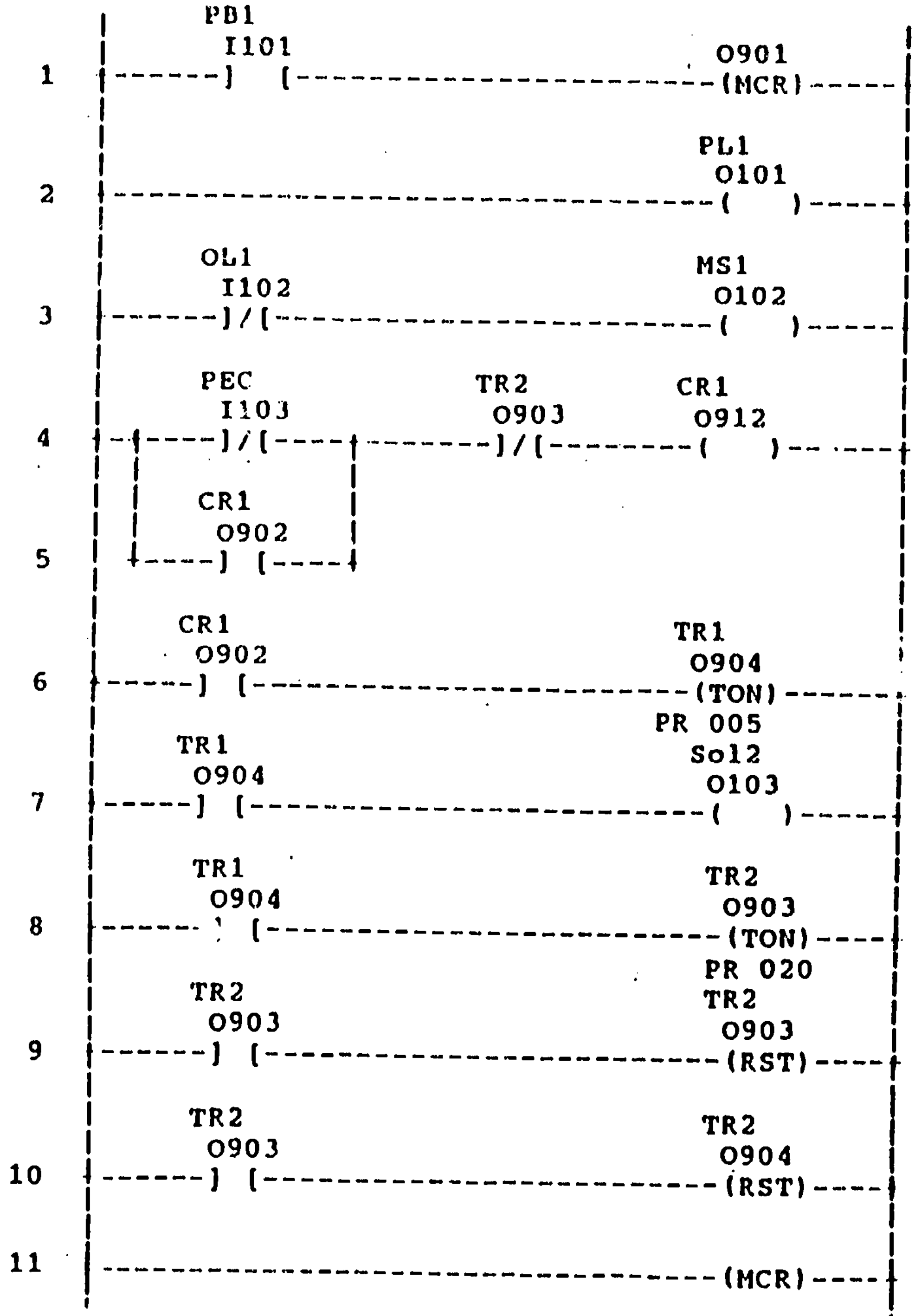
6. Addresses assigned to each input and output. ______

7. Ladder diagram with PLC conversion complete showing the operation of the control system. ______

8. Completed program entered into the PLC memory. ______

9. Program demonstrated matches the analysis. ______

10. Revisions and variables explained for changes to the control system. ______



PC Ladder Logic Diagram

There are four types of languages normally encountered in programmable controllers:

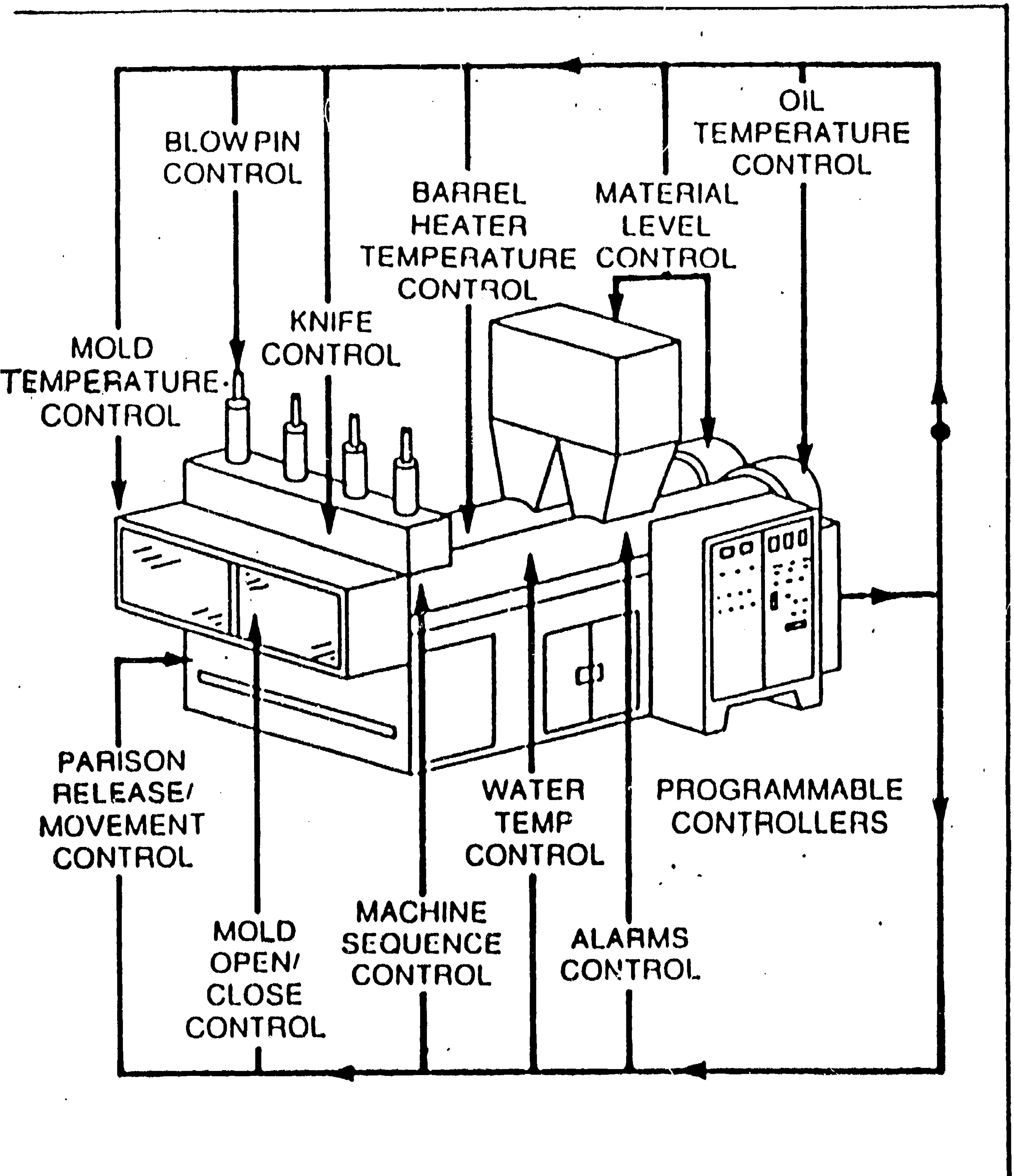
- Ladder Diagrams
- Functional Blocks
- Boolean Mnemonics
- English Statements

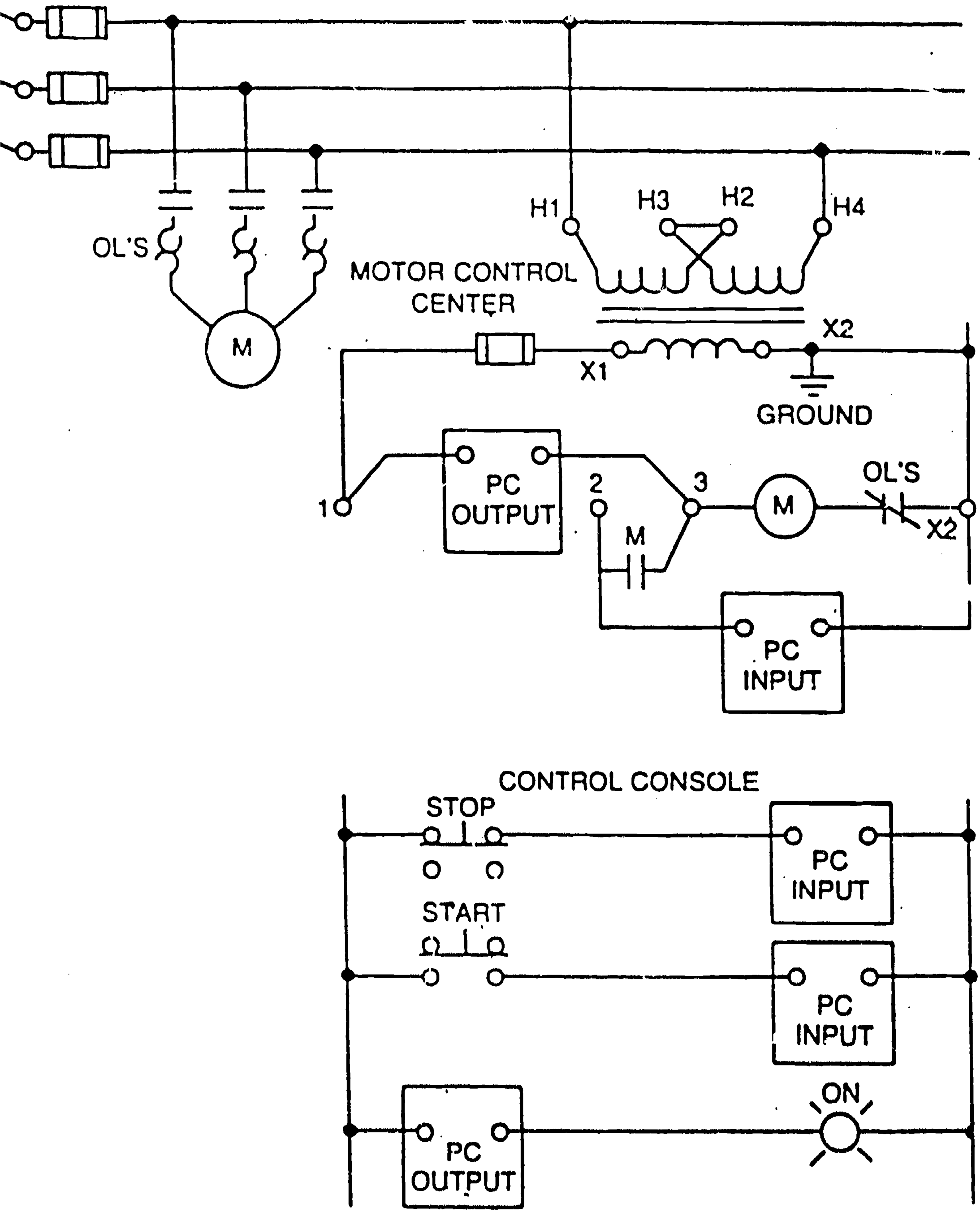
7-2 PC INSTRUCTION SUMMARY

This section contains an overview of the various programmable controller instructions. These instructions form a set of tools that include all the machine functions to perform the following six operations:

- Relay Logic
- Arithmetic
- Data Transfers
- Timing and Counting
- Data Manipulation
- Flow of Control

Operation Type	Basic Level Language	High Level Language
Relay Logic	—| |—	
	—|/|—	
	—( )—	
	—(/)—	
	—(L)—	
	—(U)—	
	—|↑|—	
	—|↓|—	
Timer and Counter	—(TON)—	TMR On
	—(TOF)—	TMR Off
	—(RTO)—	Count Up
	—(RTR)—	Count Down
	—(CTU)—	Count Up/Down
	—(CTD)—	
	—(CTR)—	
Arithmetic	—|+|—	Add
		Sub
	—|-|—	Mul
		Div
		Double Prec Add
	—| |—	Double Prec Sub
		Double Prec Mul
	—|÷|—	Double Prec Div
		Square Root
		Floating Point
		Trigonometric
Data Manipulation	—|CMP =|—	CMP =
		CMP >
	—|CMP >|—	CMP <
		CMP ≥
	—|CMP <|—	CMP ≤
		LIMIT
		Logic Matrix
		Convert BCD → BIN
		Convert BIN → BCD
		Absolute
		Invert
		Complement
		Set Constant
		Shift
		Rotate
		Examine Bit
		Increment Register
Data Transfer	—| GET |—	Move Register
		Move Point
	—(PUT)—	Move With Mask
		Move Block
		Table To Register
		Register To Table
		Block Transfer In
		Block Transfer Out
		ASCII Transfer
		FIFO Stack Transfer
Program Flow Control	—(MCR)—	
	—(ZCL)—	
	—(SKD)—	
	—(SKR)—	
	—(JMP)—	
	—|LBL|—	
	—(JSB)—	
	—(RET)—	


BLOWPIN CONTROL
OIL TEMPERATURE CONTROL
BARREL HEATER TEMPERATURE CONTROL
MATERIAL LEVEL CONTROL
KNIFE CONTROL
MOLD TEMPERATURE CONTROL
PARISON RELEASE/ MOVEMENT CONTROL
WATER TEMP CONTROL
PROGRAMMABLE CONTROLLERS
MACHINE SEQUENCE CONTROL
ALARMS CONTROL
MOLD OPEN/ CLOSE CONTROL



CONTROL STATION WIRING
WITH PROGRAMMABLE

The assignment of input and output addresses is one of the most important procedures that takes place during the conversion of your ladder logic diagram to PLC program diagram. The assignment of real inputs and outputs, as well as internals can be documented as shown

	I/O ADDRESS			
MODULE TYPE	RACK	GROUP	TERMINAL	DESCRIPTION
INPUT	0	0	0	LS1- POSITION
	0	0	1	LS2- DETECT
	O	0	2	SET SWITCH-SELECT 1
	0	0	3	PB 1-START
OUTPUT	0	0	4	PL 1
	0	0	5	PL 2
	0	0	6	MOTOR M1
	0	0	7	SOL 1
	0	1	0	SOL 2
	0	1	1	PL 3

Once addresses have been assigned you are ready to enter your program into the PLC. To enter your program you will have to use a certain language that your particular programmable controller understands. There are four types of languages currently being used in programmble controllers:

1. Ladder Diagrams
2. Functional Blocks
3. Boolean Mnemonics
4. English Statements

Of the four listed above Ladder Diagrams and Boolean Mnemonics are basic PLC languages. Functional Blocks and English Statements are high level languages. The basic PLC languages consist of a set of instructions that will perform the most basic type of control functions such as, relay replacement, timing, counting, sequencing and logic. The high level languages have been created by the need to execute more powerful instructions that go beyond the simple timing, counting, and on/off control.

To create a good, workable and safe program it is necessary to define what you want your program to do. This is accomplished by careful planning. Some steps in planning a program are as follows:

1. Define the process to be controlled.
2. Make a sketch of the process operation.
3. Create a written step sequence listing for the process.
4. Add sensors on the sketch as needed to carry out the control sequence.
5. Add manual controls as needed for process setup or operational checking.
6. Consider the safety of the operating personal and make additions and adjustments as needed.
7. Add master stop switches as required for safe shutdown.
8. Create the ladder logic diagram that will be used as the basis for the PLC program.
9. Consider the "what if's" where the process may go astray.

When your planning is completed you will be ready to convert your ladder logic diagram into a PLC program.

Section 12
WRITING THE USER PROGRAM

12.0 GENERAL

A thorough understanding of the programming instructions and Processor operation as described in Section 1.4, Hardware/Program Interface is essential for writing the User Program. Although approaches to and methods of writing programs that control machine operation vary, there are some guidelines that should be followed.

12.1 DEVELOPING THE PROGRAM

The first step in developing the User Program is to establish an operating sequence for input and output devices. The sequence must be evaluated to determine what the devices must do, what the conditions must be and the order in which they must operate.

After evaluating the operating sequence, the action of the different devices should be described in proper sequence with proper conditions for energizing each output device. This description is then used to develop the ladder diagram program. If a process diagram exists, it can be used as an aid in developing a ladder diagram program.

12.2 SAMPLE PROGRAM

The way a ladder diagram program is developed is best described by a simple example.

The application is one of separating good parts from bad parts. Figure 12-1 shows a part moving along a conveyor belt. Each part will trip a series of limit switches and will be sorted according to its height. The desired height is 1.0" ±0.1".

If a part trips 2LS but not 3LS, the part is greater than or equal to 0.9" and less than or equal to 1.1". Because it is a good part, a storage bit (3SB) is latched ON. When the part trips 4LS, SOL1 is energized which moves the swingarm actuator, directing the part onto the good part conveyor.

If the part trips both or neither 2LS and 3LS, the part is too large or too small. When either condition occurs, a storage bit (4SB) is latched

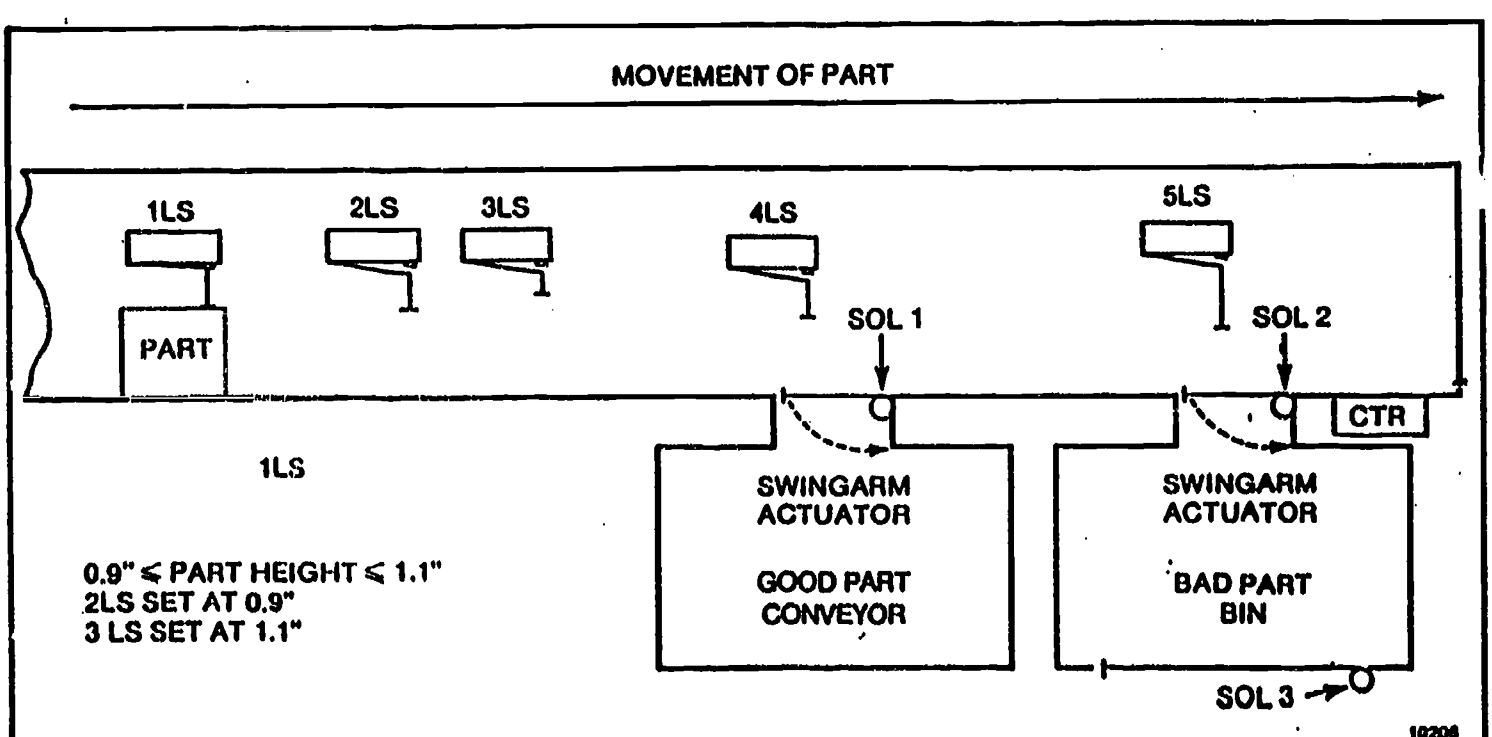


FIGURE 12-1 — Conveyor Belt Example

ON. Although the part will trip 4LS, it will continue along and trip 5LS, which energizes SOL2. The swingarm actuator will direct the part into the bad part bin. Each time a part enters the bad part bin, a counter is incremented. When the bin is full (count complete), SOL3 is energized which opens the bottom of the bin long enough to empty it. The counter will then reset automatically. Each time a new part enters the conveyor belt, 1LS is tripped which unlatches the storage bits and begins a new cycle.

The conveyor motor can be started or stopped with pushbutton START or STOP switches. Motor starter, MS1, controls the conveyor motor. A watchdog timer is used to monitor the flow of parts. If parts should become jammed causing a delay between 1LS and 4LS, the timer will time-out and turn OFF the conveyor motor. Another watchdog timer detects if a part becomes jammed beneath 4LS or 5LS. A conveyor RUN indicator and a parts JAM indicator allow remote observation of the conveyor operation. Additional documentation (not shown) would include a Power Distribution schematic showing a hardwired master control relay and emergency stop switches.

The logic can be written as a PC ladder diagram program (Figure 12-2). Data Table addresses are assigned to the hardwired devices. (Table 12-1) The ladder diagram should be developed by analyzing the logic required to operate the machine. A rung by rung description of the logic follows.

Rung 1. This rung provides 3-wire control of the conveyor motor with jam detection for automatic shut down.

Rung 2. The auxiliary contact of the motor starter is monitored to provide a conveyor RUN indicator.

Rungs 3, 4, 5, 6. The part trips the first limit switch and unlatches storage bits 1-4 to begin a new cycle.

Rung 7. The first limit switch enables a Retentive Timer which is latched by the timer Enable bit. A jam condition is detected if the timer times out.

Rung 8. Limit switch 4 (or the START pushbutton) resets the timer. If reset prior to 5 seconds, no jam has occurred between 1LS and 4LS. A jam beneath 4LS or to the right of it is not detected by this rung.

Rung 9. A part passing 2LS latches SB1 if the height $\geq$ 0.9 inch. SB1 remains unlatched if the height $<$ 0.9 inch.

Rung 10. A part passing 3LS latches SB2 if the height $>$ 1.1 inch. SB2 remains unlatched if the height $<$ 1.1 inch.

Rung 11. A part within tolerance latches SB3.

TABLE 12-1 — Data Table Addresses for Hardwired Devices

INPUT DEVICE	ADDRESS
STOP Pushbutton	112/00
START Pushbutton	112/01
Motor Starter Auxiliary	112/02
Limit Switch (1LS)	112/03
Limit Switch (2LS)	112/04
Limit Switch (3LS)	112/05
Limit Switch (4LS)	112/06
Limit Switch (5LS)	112/07
OUTPUT DEVICE	
Motor Starter (MS1)	114/00
Conveyor RUN Indicator	114/01
Good Part Solenoid (SOL1)	114/02
Bad Part Solenoid (SOL2)	114/03
Bin Dump Solenoid (SOL3)	114/04
JAM Detect Indicator	114/05
INTERNAL FUNCTIONS	
Storage Bit 1 (SB1)	012/01
Storage Bit 2 (SB2)	012/02
Storage Bit 3 (SB3)	012/03
Storage Bit 4 (SB4)	012/04
Storage Bit 5 (SB5)	012/05
Retentive Timer, Watchdog	050
Timer, Bin Dump	051
Timer, Watchdog	052
Counter	060