

# **Series One™ Programmable Controllers**

---

Series One/One Plus  
User's Manual

**GE Fanuc Automation**

August 1988

GEK-90842C

# WARNINGS, CAUTIONS, AND NOTES AS USED IN THIS PUBLICATION

## WARNING

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

## CAUTION

Caution notices are used where equipment might be damaged if care is not taken.

## NOTE

Notes merely call attention to information that is especially significant to understanding and operating the equipment.

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware and software, nor to provide for every possible contingency in connection with installation, operation, and maintenance. Features may be described herein which are not present in all hardware and software systems. GE Fanuc Automation assumes no obligation of notice to holders of this document with respect to changes subsequently made.

GE Fanuc Automation makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. No warranties of merchantability or fitness for purpose shall apply.

---

---

GEK-90842

The purpose of this manual is to provide information for the user to install, program and implement the family of Series One™ family of Programmable Controllers (PCs) into a control system. The Series One family of PCs includes the Series One, Series One Model E, Series One Plus and Series One Plus 3.7K PCs, which are described in this manual. The Series One Junior PC is described in a separate User's Manual, GEK-90503.

The Series One PC provides the user with the capability of developing and programming a control system using the familiar ladder diagram logic approach. The Series One Plus includes the same ladder diagram function plus a group of data operation functions, which includes data moves, math functions, conversion, and fault diagnosis.

**Chapter 1, Introduction**, is an introduction to the Series One and Series One Plus PCs, with emphasis on features and capabilities. A summary of terms common to PCs is provided at the end of this chapter as an aid to first-time PC users.

**Chapter 2, Physical Equipment Configuration**, provides a detailed description of the hardware components of the PC. This chapter provides an understanding of the components of a Series One or Series One Plus PC system and how they are related to the overall system.

**Chapter 3, Installation**, provides the specifications and instructions required for installation of your Programmable Control system.

**Chapter 4, PC Operation**, describes the operation of the Series One and Series One Plus PCs, including features and functions of the programmer used for entering new programs, editing existing programs, monitoring the status of inputs or outputs, displaying timer or counter accumulated values, and displaying register contents. The last part of this chapter describes operation of peripheral devices which may be used with both Programmable Controllers. Peripherals include an audio cassette tape recorder for recording your program after it has been entered in order to have a permanent record of that program, a PROM Writer Unit, which allows a non-volatile means of program storage within the PC, and a Printer Interface Unit to allow documenting of your programs.

**Chapter 5, Programming**, provides the basic information required in order to develop, enter, and implement your programs. A description of each function is provided, including examples of using each function. Three sections are included: Programming, Basic Instructions, and Data Operation Instructions.

**Chapter 6, I/O Specifications and Wiring**, is a guide to the specifications of the input and output (I/O) modules and their physical connections to field devices.

**Chapter 7, Maintenance**, is a guide to basic maintenance of your system, should it be needed. Reliability of the Series One family of PCs is excellent and other than changing the Lithium back-up battery, when required, there should be little maintenance required of your PC. This chapter includes troubleshooting procedures and information on replacing components.

**Chapter 8, Applications**, provides several typical Applications using the capabilities of the Series One and Series One Plus PCs. This chapter should be especially helpful to first-time users of a programmable controller.

**Appendices A through D** contain a summary of Related Documentation, a Glossary of Programmable Controller Terms, a guide to the compatibility of the Series One family of Programmable Controllers, and a Description of other GE Fanuc Automation North America, Inc. Programmable Controllers. A comprehensive index is included as an aid to the location in the manual of particular items of interest.

---

*All references to Series One in this manual for hardware and programming apply to both the Series One and Series One Model E PCs, except where specifically noted in the text. Similarly, all references to Series One Plus apply to both the Series One Plus and Series One Plus 3.7K PCs.*

Should further information be required, contact your salesperson or GE Fanuc Automation North America, Inc., P. O. Box 8106, Charlottesville, Virginia 22906.

First Edition - May 1986  
Second Edition - March 1987  
Third Edition - September 1987

Henry A. Konat  
Senior Technical Writer

**NOTE**

The Series One/Series One Plus and associated modules have been tested and found to meet or exceed the requirements of FCC Rule, Part 15, Subpart J. The following note is required to be published by the FCC.

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. It has been tested and found to comply with the limits of a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.



GEK-90842

<b>CHAPTER 1.</b>	<b>SERIES ONE/SERIES ONE PLUS PC INTRODUCTION</b>	
	History of Programmable Controllers	1-1
	Advantages Over Other Control Devices	1-1
	Series One Programmable Controllers	1-1
	Series One Model E Programmable Controller	1-3
	Series One Plus 3.7K	1-3
	Programmable Controller Concepts	1-5
	Programming the Series One and Series One Plus PCs	1-5
	Function of the Central Processing Unit	1-6
	Memory Word Length	1-6
	Types of Memory for Program Storage	1-7
	Function of the Input/Output Circuitry	1-7
	Communicating With Other Devices	1-8
	Remote I/O for Series One and Series One Plus	1-8
	PC Terminology	1-8
<b>CHAPTER 2.</b>	<b>PHYSICAL EQUIPMENT CONFIGURATION</b>	
	Rack Description	2-1
	Module Location in Rack	2-1
	Heat Dissipation	2-2
	Adding Racks	2-2
	Programmer Tape Port	2-4
	Programmer Mount Assembly	2-4
	Hardware Requirements	2-7
	System Estimating	2-7
	Peripheral Devices Supporting Series One Family of PCs	2-9
	Data Communications Unit	2-10
	Printer Interface Unit	2-11
	PROM Writer Unit	2-12
<b>CHAPTER 3.</b>	<b>INSTALLATION</b>	
	Specifications for Installation	3-1
	Installation	3-2
	Power Supply Limitations for Racks	3-13
	Units of Load	3-13
	Safety Considerations	3-15
	Recommended Field Wiring Procedures	3-15
<b>CHAPTER 4.</b>	<b>PC OPERATION</b>	
	Introduction to PC Operation	4-1
	Programmer	4-1
	Programmer Features	4-2
	Mode Switch	4-2
	Address Data Display	4-2
	Status Display	4-2
	Logic Display	4-2
	Logic Keys	4-3
	Editing Keys	4-4

Shifted Functions	4-5
Data Operation Keys	4-6
Peripheral Jack	4-7
Program Checking and Error Codes	4-7
Operation Sequences	4-8
Monitor CPU Logic	4-10
Search CPU Logic	4-11
Alter One Logic Element	4-11
Delete One Logic Element	4-12
Clear All Memory	4-12
Insert One Logic Element	4-12
Monitor I/O Status	4-12
Monitor Timer or Counter Status	4-13
Display a Specific Address	4-13
Monitor Data Register Contents	4-14
Change Contents of a Data Register	4-14
Enter or Change a Password	4-14
Access to Programmer Functions with Password in Effect	4-14
Forcing References	4-15
Operation With Peripheral Devices	4-15
Tape Recorder Operation	4-15
Recording A Program	4-16
Verifying A Program	4-16
Loading A Program	4-18
Printer Interface Unit	4-18
Hardware Description	4-19
Hardware Features	4-20
80/132 Column Selection Switch	4-21
External Power Supply Connector	4-21
Power Supply Select Switch	4-21
Sequence of Operation	4-22
Power-up Sequence	4-22
User Program Transfer From PC To Printer Interface Unit	4-22
Selection of Printout Format and Type of PC	4-22
Start Printer Operation	4-22
Printing Of Error Messages During Ladder Diagram Listing	4-23
Printing Of Error Messages During Boolean Listing	4-23
Cross Reference Printout	4-23
Expanded Print Format	4-24
Printout Annotation Explanation	4-24
Sample Printout	4-25
PROM Writer Unit	4-29
Front Panel Features	4-30
Sequence of Operation	4-31
External Power Supply Connector	4-31
Power Supply Select Switch	4-31

GEK-90842

	Timer/Counter Setpoint Unit	4-32
	Timer/Counter Setpoint Unit Specifications	4-33
	Remote Mounting of Timer/Counter Setpoint Unit	4-33
	References for the Timer/Counter Setpoint Unit	4-33
	Example of Using Thumbwheel Inputs	4-34
<b>CHAPTER 5.</b>	<b>PROGRAMMING</b>	
Section 1.	Introduction to Programming	5-1
	General Information	5-1
	Planning a PC System	5-1
	How to Use This Chapter	5-1
	Programming Fundamentals	5-7
	Significance of References	5-7
	Significance of Input/Output References	5-8
	Internal Coils	5-13
	Use of the Special Function Coils	5-13
	Shift Register References	5-13
	Timer and Counter References	5-13
	Data Registers	5-13
	Flexibility in Using References	5-14
	Operating Principles	5-14
	Scanning	5-14
	Programmer Functions	5-15
	Basic Ladder Diagram Format	5-17
	Concept of Power Flow	5-17
	Unlimited References	5-18
Section 2.	Basic Instructions	5-19
	Basic Instructions	5-19
	How to Begin Programming	5-19
	Entering a Rung With Series Contacts	5-19
	Entering the Clear all Memory Sequence	5-19
	Entering a Rung with Parallel Contacts	5-20
	Entering a Simple Timer Rung	5-20
	Basic Relay Logic (Motor Starter)	5-20
	Motor Starter Logic Description	5-21
	Normally Closed Input	5-22
	Push-Down Stack	5-23
	Detailed Example of AND STR, OR STR	5-24
	Description of Operation	5-24
	Use of Retentive Coils as Latches	5-26
	Programming a Latched Relay	5-26
	Master Control Relay Functional Description	5-27
	Programming a Master Control Relay Function	5-28
	Multiple Master Control Relay Functions	5-28
	Disabling of Outputs	5-29
	Timer and Counter Functional Description	5-30
	Special Timer/Counter References	5-30
	Programming Timers	5-30
	Specifying Timer Preset Values	5-31
	Programming Counters	5-32

	Extending the Timer and Counter Range	5-34
	Sequencer Operation	5-36
	Referencing Sequencer Contacts with a Series One Plus PC	5-37
	Shift Register Functional Description	5-38
	Shift Register References	5-39
	Shift Register Operation	5-40
	Forcing I/O References	5-42
Section 3.	Data Operation Instructions for the Series One Plus Programmable Controller	5-43
	Data Operations	5-43
	Using References for Data Operations	5-44
	Entering a Constant Value	5-44
	Entering a Group Reference	5-44
	Data Register References	5-44
	Timer/Counter References	5-45
	Example of Specifying a Group Reference	5-45
	Programming the Data Operation Instructions	5-45
	Special Function Coils for Data Operations	5-46
	D•STR (F50)	5-47
	D•STR1 (F51)	5-48
	D•STR2 (F52)	5-48
	D•STR3 (F53)	5-49
	D•STR5 (F55)	5-49
	D•OUT (F60)	5-50
	D•OUT1 (F61)	5-50
	D•OUT2 (F62)	5-51
	D•OUT3 (F63)	5-51
	D•OUT5 (F65)	5-52
	CMPR (F70)	5-52
	Addition (+), BCD 4 Digit (F71)	5-53
	Subtraction (-), BCD 4 Digit (F72)	5-56
	Multiplication (X), BCD 4 Digit (F73)	5-58
	Division (:), BCD 4 Digit (F74)	5-59
	D•AND (DATA AND) (F75)	5-61
	D•OR (DATA OR) (F76)	5-63
	Shift Right (F80)	5-65
	Shift Left (F81)	5-65
	Decode (F82)	5-67
	Encode (F83)	5-68
	INV (F84)	5-68
	BIN (BCD to Binary) (F85)	5-69
	BCD (Binary to BCD) (F86)	5-70
	External Fault Diagnosis (F20)	5-71
CHAPTER 6.	I/O SPECIFICATIONS AND WIRING	
	I/O System Introduction	6-1
	Field Wiring to I/O Modules	6-2
	16 Circuit I/O Modules with Connectors	6-4
	I/O References for 16 Circuit Modules	6-4

GEK-90842

I/O Interface Cable Cross Reference List	6-5
I/O Module Catalog Numbers	6-6
I/O Module Specifications and Wiring	6-7
115 V ac Input IC610MDL125	6-7
230 V ac Input IC610MDL127	6-8
115 V ac Isolated Input IC610MDL126	6-9
24 V dc Sink Input (8 Circuits) IC610MDL101	6-10
24 V dc Sink Input (16 Circuits) IC610MDL106	6-11
24 V dc Sink Load Input (16 Circuits) with Removable Terminal Board IC610MDL107	6-12
24 V ac/dc Source Input IC610MDL111	6-13
24 V ac/dc Source Input (16 Circuits) with Removable Terminal Board IC610MDL112	6-14
115/230 V ac Output IC610MDL175	6-16
115/230 V ac Isolated Output IC610MDL176	6-17
24 V dc Sink Output (8 Circuits) IC610MDL151	6-18
24 V dc Sink Output (16 Circuits) IC610MDL156	6-19
24 V dc Sink Output (16 Circuits) with Removable Terminal Board IC610MDL157	6-20
24 V dc 2 Amp Sink Output IC610MDL153	6-22
24 V dc 2 Amp Sink/Source Output IC610MDL154	6-23
24 V dc Source Output IC610MDL155	6-24
24 V dc Source Output (16 Circuits) with Removable Terminal Board IC610MDL158	6-26
Relay Output (8 Circuits) IC610MDL180	6-27
Relay Output (16 Circuits) with Removable Terminal Board IC610MDL182	6-29
24 V dc Input/Output (4 In/4 Out) IC610MDL103	6-31
24 V dc Input/Relay Output (4 In/4 Out) IC610MDL104	6-33
Thumbwheel Interface IC610MDL105	6-36
High Speed Counter IC610MDL110	6-39
Module Location	6-41
Interface to Field Devices	6-42
Up/Down Counter Inputs	6-43
Encoder Interface 1	6-44
Binary Coded Decimal (BCD) Output	6-46
Counter Output	6-48
Interface to User Logic	6-49
Interface Function Definition	6-50
Output Logic	6-53
Manual Mode	6-53
Counter Mode	6-53
Filter Selection	6-55
Fast Response I/O IC610MDL115	6-56
Mode Selection	6-58
Output Logic - Manual Mode	6-58
Output Logic - Fast Response Mode	6-58
I/O Simulator IC610MDL124	6-61
Analog Input IC610MDL116	6-62

	Introduction	6-62
	Hardware Features	6-62
	Power Requirements	6-62
	General and Electrical Specifications	6-62
	I/O Reference Definitions	6-63
	Sample Ladder Logic	6-64
	Analog Input Module Features	6-66
	Selection Of Operating Mode	6-66
	Analog OutputIC610MDL166	6-67
	Introduction	6-67
	Hardware Features	6-67
	Power Requirements	6-67
	General and Electrical Specifications	6-67
	I/O Reference Definitions	6-68
	Sample Ladder Logic	6-69
	Analog Output Module Features	6-70
	UL Listed Products	6-71
	5-Slot Rack, UL ListedIC610CHS111	6-72
	115 V ac Input Module, UL ListedIC610MDL135	6-73
	Relay Output Module, UL ListedIC610MDL181	6-74
	115 V ac Output Module, UL ListedIC610MDL185	6-75
	115 V ac Input Module (16 Circuits) with Removable Terminal Board IC610MDL129	6-76
	115/230 V ac Output Module (8 Circuits) with Removable Terminal Board IC610MDL179	6-77
<b>CHAPTER 7.</b>	<b>MAINTENANCE</b>	
	Introduction to Maintenance Procedures	7-1
	Troubleshooting Aids	7-1
	Basic Troubleshooting Procedure	7-1
	General Troubleshooting Procedure	7-2
	Replacement of Components	7-3
	Replacing a Rack	7-3
	Replacing a CPU Module	7-3
	Replacing I/O Modules	7-4
	Replacing the Battery	7-4
	Adding Memory	7-5
	Spare Parts and Components	7-6
	Fuse List	7-7
<b>CHAPTER 8.</b>	<b>APPLICATIONS</b>	
	Application 1 - One-Shots	8-1
	Application 2 - Flip Flop	8-2
	Application 3 - Event/Time Drum	8-3
	Application 4 - Cascaded Counters	8-5
	Application 5 - Coil 374, Power-Up One-Shot	8-7
	Description of Operation	8-7
	Application 6 - Coil 375, 10Hz Clock	8-8
	Example 1: - Cumulative Timer	8-8
	Example 2: - Time of Day Clock	8-8
	Application 7 - Start/Stop Circuit	8-11

GEK-90842

	Application 8 - High Speed Counter Applications	8-12
	Programmable Cam Switch	8-13
	Cut to Length	8-16
	Measuring a Random Length	8-18
	Sample Calculation	8-18
	Application 9 - Typical Shift Registers	8-21
	Shift Register References	8-21
<b>APPENDIX A.</b>	<b>Related Documentation</b>	<b>A-1</b>
	Introduction	A-1
	GEK-90477 Series One/Three Data Communications Manual	A-1
	GEK-90507 Remote I/O User's Manual	A-1
	GEK-90825 Series Six PC I/O Link Local Module User's Manual	A-2
	GEK-90846 Portable Programmer User's Manual	A-2
	GEK-96662 Logicmaster 1 Programmer Documentor Manual	A-3
	GFK-0075 Logicmaster 1 Family Programming and Documentation Software User's Manual	A-4
<b>APPENDIX B.</b>	<b>Glossary of Terms</b>	<b>B-1</b>
<b>APPENDIX C.</b>	<b>Series One Family of Programmable Controllers Compatibility Guide</b>	<b>C-1</b>
<b>APPENDIX D.</b>	<b>Other Programmable Controllers</b>	<b>D-1</b>
	Series One™ Junior Programmable Controller	D-1
	Series Three™ Programmable Controller	D-2
	Available I/O Types	D-2
	Series Six™ Programmable Controller	D-3
	Programming	D-3
	Optional Items	D-4
	Series Six™ Plus Programmable Controller	D-4

GEK-90842

Figure	1-1	Series One/Series One Plus Programmable Controllers	1-2
	1-2	Basic PC Block Diagram	1-5
	2-1	Typical Rack	2-1
	2-2	Typical Hardware	2-2
	2-3	Typical (5 Slot) Rack	2-3
	2-4	Typical (10 Slot) Rack	2-3
	2-5	Expander Cable Installation	2-4
	2-6	Programmer Mount Assembly	2-5
	2-7	Example of Relay Control	2-9
	2-8	Data Communications Unit	2-10
	2-9	Printer Interface Unit	2-11
	2-10	Prom Writer Unit	2-12
	3-1	Rack Mounting Dimensions for Proper Heat Dissipation	3-2
	3-2	Recommended Rack Grounding	3-3
	3-3	10 Slot Rack, 19 Inch Mounting Dimensions	3-4
	3-4	I/O Expansion Cable Connection	3-4
	3-5	I/O Addressing Switches in Series One Plus 10 Slot Racks	3-5
	3-6	Examples of Rack Configuration Switch Setting Series One Plus	3-6
	3-7	Typical Installation Dimensions	3-7
	3-8	CPU Module	3-8
	3-9	Proper Module Insertion	3-9
	3-10	Proper Module Removal	3-9
	3-11	Typical Routing of I/O Wiring	3-11
	3-12	Pull Tab on Programmer Cable	3-12
	3-13	Installation of Programmer Cable	3-12
	3-14	Programmer Mount Assembly	3-13
	4-1	Programmer Features	4-1
	4-2	Series Logic (AND)	4-3
	4-3	Parallel Logic (OR)	4-3
	4-4	Volume Control Range Setting	4-17
	4-5	Printer Interface Unit	4-20
	4-6	Sample Ladder Diagram Printout	4-25
	4-7	Sample Boolean Printout	4-26
	4-8	Sample Outputs Used Table Printout	4-27
	4-9	Sample Outputs Used Table Printout (continued)	4-28
	4-10	PROM Writer Unit Features	4-29
	4-11	Timer/Counter Setpoint Unit	4-32
	5-1	I/O References Per Physical Placement for 5 and 10-Slot Racks	5-9
	5-2	CPU Scanning Sequence	5-15
	5-3	Programmer for Series One and Series One Plus	5-16
	5-4	Typical Ladder Diagram	5-17
	5-5	Sample Relay Logic (Motor Starter)	5-21
	5-6	Example of Normally Closed Inputs	5-22
	5-7	AND STR and OR STR Connections	5-23
	5-8	Push-Down Stack Storage Locations	5-24
	5-9	Push-Down Stack Logical Operations	5-24
	5-10	AND STR/OR STR Example Number 1	5-25

5-11	AND STR/OR STR Example Number 2	5-26
5-12	Example of Latching Logic	5-27
5-13	Example of Master Control Relay Logic	5-28
5-14	Multiple Master Control Relay Logic	5-29
5-15	Sample Coil Disabler Logic	5-30
5-16	Sample of Timer Logic	5-31
5-17	Example of Timer Logic	5-32
5-18	Example of Counter Logic	5-33
5-19	Example of Counters	5-34
5-20	Example of Extended Counters	5-35
5-21	Illustration of Sequencer Operation	5-36
5-22	Example of Sequencer	5-37
5-23	Typical Sequencer Logic	5-38
5-24	Typical Shift Register	5-39
5-25	Example of Shift Register	5-40
5-26	Example of Shift Register Logic	5-41
5-27	Data Operation Block Diagram	5-43
6-1	A. Typical I/O Module      B. High Density Module With Removable Connector	6-3
6-2	Typical I/O Terminal Configuration	6-3
6-3	I/O Interface Cable Wiring List	6-4
6-4	Example of 16 Circuit I/O References	6-5
6-5	Wiring for 115 V ac Inputs	6-7
6-6	Wiring for 230 V ac Inputs	6-8
6-7	Wiring for 115 V ac Isolated Inputs	6-9
6-8	Wiring for 24 V ac Sink Inputs	6-10
6-9	I/O Points VS Temperature	6-11
6-10	Wiring for 16 CIRCUIT, 25 V dc Sink Input Module	6-11
6-11	I/O Points VS Temperature	6-12
6-12	Wiring for 16 Circuit, 24 V dc Sink Load Inputs	6-12
6-13	Wiring for 24 V ac/dc Inputs	6-13
6-14	I/O Points vs Temperature Chart	6-14
6-15	Wiring for 16 Circuit 24 V ac/dc Source Load Inputs	6-15
6-16	I/O Points vs Temperature	6-16
6-17	Wiring for 115/230 V ac Outputs	6-16
6-18	I/O Points vs Temperature Chart	6-17
6-19	Wiring for 15 V ac Isolated Outputs	6-17
6-20	I/O Points vs Temperature Chart	6-18
6-21	Wiring for 24 V dc Sink Outputs	6-18
6-22	I/O Points vs Temperature	6-19
6-23	Wiring for 16 Circuit 24 V dc Sink Outputs	6-19
6-24	I/O Points vs Temperature Chart	6-20
6-25	Wiring for 16 Circuit 24 V dc Sink Outputs	6-21
6-26	I/O Points vs Temperature Chart	6-22
6-27	Wiring for 24 V dc 2 Amp Sink Outputs	6-22
6-28	I/O Points vs Temperature	6-23
6-29	Wiring of 24 V dc 2 Amp Sink/Source Output	6-23
6-30	I/O Points vs Temperature Chart	6-24
6-31	Wiring for 24 V dc Source Outputs	6-25
6-32	I/O Points vs Temperature	6-26

6-33	Wiring for 24 V dc Source Outputs	6-26
6-34	Wiring for 8 Circuit Relay Outputs	6-28
6-35	Wiring for 16 Circuit Relay Outputs	6-30
6-36	Wiring for 24 V dc Inputs/Outputs	6-32
6-37	Typical Wiring Diagram	6-34
6-38	Sample 24 V dc INPUT/RELAY OUTPUT Circuits	6-35
6-39	Wiring for Thumbwheel Interface	6-37
6-40	Thumbwheel Interface Cable Wire List	6-38
6-41	High Speed Counter Overview	6-40
6-42	HSC Location in 5-Slot Rack	6-41
6-43	HSC Connector Pin Definition	6-42
6-44	Signal Direction	6-43
6-45	UP/DOWN/RESET Input Circuit	6-43
6-46	Encoder with RESET/MARKER Option Resetting Counter Once per Revolution of Encoder Shaft	6-44
6-47	Encoder with RESET Option in Series with Home Limit Switch Such That Counter is Reset When Both Home Limit Switch and RESET/MARKER Pulse are Enabled	6-44
6-48	Encoder With Limit Switch Resetting Counter and Photoelectric Cell Inhibiting the Counter Operation	6-45
6-49	Sample BCD Output Circuit	6-46
6-50	BCD Output Wiring Diagram	6-47
6-51	User Load Wiring Diagram	6-48
6-52	I/O and Counter Function Reference Chart	6-49
6-53	Filter Selection with Dip Switch	6-55
6-54	Module Reference NUMBER/LOGIC Symbol Definition	6-57
6-55	Fast Response Module Logic Diagram	6-59
6-56	Wiring for Fast Response I/O Module	6-60
6-57	I/O Simulator Module Faceplate	6-61
6-58	Analog Input Module Faceplate	6-66
6-59	Analog Output Module Faceplate	6-70
6-60	UL Listed 5-Slot Rack	6-72
6-61	Wiring for UL Listed 115 V ac Input Module	6-73
6-62	Wiring for UL Listed Relay Output Module	6-74
6-63	Wiring for UL Listed 115V ac Output Module	6-75
6-64	I/O Points vs Temperature Chart	6-76
6-65	Wiring for 115 V ac Inputs with Removable Terminal Board	6-76
6-66	I/O Points vs Temperature Chart	6-77
6-67	Wiring for 115/230 V ac Output with Removable Terminal Board	6-77
7-1	Troubleshooting Indicators	7-2
7-2	Battery Location and Connection	7-4
7-3	Location of Extra Memory Socket	7-5
7-4	Accessory Kit for Series One/One Plus	7-7
8-1	Typical One-Shot Timing	8-1
8-2	Typical One-Shot Logic	8-1
8-3	Typical Flip-Flop Timing Diagram	8-2
8-4	Typical Flip-Flop Logic	8-2
8-5	Event/Time Drum Logic	8-3

8-6	Sample Cascaded Counter Logic	8-6
8-7	Typical Machine or Process Power-Up Inhibit Logic	8-7
8-8	Cumulative Timer	8-8
8-9	Sample 24 Hour Time Clock Logic	8-9
8-10	Time Clock Resetting	8-10
8-11	Fast Response Start/Stop Logic	8-11
8-12	Worst Case Timing Diagram	8-11
8-13	Application System	8-12
8-14	Programmable CAM Switch Logic Timing Diagram	8-13
8-15	Programmable CAM Switch	8-14
8-16	Output No. 1 Logic Diagram	8-16
8-17	Wiring Diagram	8-16
8-18	Cut to Length Program	8-17
8-19	Measuring a Random Length	8-18
8-20	Shift Register Example	8-21
8-21	Typical Shift Register Logic	8-22

GEK-90842

<b>Table</b>	1-1	Series One Family General Specifications	1-4
	1-2	Common PC Terminology	1-8
	2-1	Series One/Series One Plus Catalog Numbers	2-5
	3-1	Series One Installation Specifications	3-1
	3-2	CPU Option Settings	3-8
	3-3	Memory Size	3-8
	3-4	Units of Load Supplied by Rack	3-14
	3-5	Units of Load Used by Modules	3-14
	4-1	Error Code Definitions	4-7
	4-2	PC Operation Sequences	4-9
	4-3	Printer Interface Unit Specifications	4-19
	4-4	Printer Requirements	4-19
	4-5	Format and PC Selection	4-22
	4-6	Ladder Diagram Listing Error Messages and Definitions	4-23
	4-7	Boolean Listing Error Messages and Definitions	4-23
	4-8	Timer/Counter Setpoint Specifications	4-33
	5-1	Series One Model E/One Plus Basic Ladder Diagram Instructions	5-2
	5-2	Series One Execution Times	5-3
	5-3	Series One Plus Data Operation Instructions	5-4
	5-4	Series One/One Plus Instruction Groups	5-7
	5-5	Summary of References	5-8
	5-6	Summary of I/O References for 8 Circuit Modules	5-10
	5-7	Effect of Coil 376 (Output Disabler)	5-29
	5-8	Data Operation Function Numbers	5-46
	6-1	I/O Module Catalog Numbers	6-6
	6-2	Maximum Current vs Load Type for Relay Outputs	6-27
	6-3	Maximum Current vs Load Type for Relay Outputs	6-29
	6-4	Maximum Current vs Load Type for Relay Outputs	6-34
	6-5	Number of HSCs vs Discrete I/O Capacity	6-42
	6-6	Manual Mode Output Logic Truth Table	6-53
	6-7	Real Time Comparison Table for Preset Initially > Current Count	6-53
	6-8	Real Time Comparison Table for Preset Initially < Current Count	6-54
	6-9	Output State vs Real Time Comparison Status in Counter Mode	6-54
	6-10	Output Logic in Counter Mode	6-55
	6-11	Maximum Current vs. Load Type for Relay Outputs	6-57
	6-12	Operating Mode Selection	6-58
	6-13	Truth Table for Output in Fast Response Mode (S5 or S7 Have Been Enabled)	6-58
	6-14	Analog Input Module Specifications	6-63
	6-15	I/O Point Definition	6-63
	6-16	Analog Output Module Specifications	6-68
	6-17	I/O Point Definition	6-68
	7-1	I/O Module Fuse List	7-7
	C-1	I/O Module Capability/Compatibility Guide	C-3

---

**GEK-90842**

C-2	Summary of Programming References for Series One Family of Programmable Controllers	C-4
C-3	Programming Function Compatibility Guide	C-5
C-4	Examples of Valid System Configurations for I/O Points	C-6
D-1	Series One Junior Capabilities	D-1
D-2	Series Three PC Specifications	D-2
D-3	Series Six CPU Capacities	D-3
D-4	Series Six Plus Combined Memory Configurations	D-4



GEK-90842

## History of Programmable Controllers

The factory with a future is here today, with machine and process controls provided by modern electronic devices. Today's automated factory can provide improved system reliability, product quality, information flow, reduced costs, efficiency, and flexibility. One of the basic building blocks of such a factory is an electronic device called a Programmable Controller. This device was first introduced in 1970 and has been refined every 4-7 years as newer electronic components, such as microprocessors, are made available. Today's Programmable Controllers are designed using the latest in microprocessor designs and electronic circuitry which provide reliable operation in industrial applications where many hazards such as electrical noise, high temperature, unreliable AC power, and mechanical shock exist. Here is where the Programmable Controller is in its element; it was designed for the industrial environment from its conception.

## Advantages Over Other Control Devices

Programmable Controllers, or PCs or PLCs as they are frequently referred to, offer many advantages over other control devices such as relays, electrical timers and counters, and drum type mechanical controllers. These advantages should be considered beyond just price when selecting any control device:

- Improved reliability
- Smaller space required
- Easier to maintain
- Reusable
- Reprogrammable if requirements change
- More flexible-performs more functions

## Series One Programmable Controllers

A complete Series One PC contained in one 5-slot rack can have over 1700 words of user logic, up to 64 timer/counters, and up to 64 I/O points in a panel surface area of only 54 square inches. That is less space than four 4-pole relays might occupy. An available 10-slot rack can contain up to 120 I/O points. By adding additional racks (up to a total of three), the I/O can be expanded up to a total of 112 I/O points in a Series One PC or 168 points in a Series One Plus PC. Modules can be replaced individually without disturbing adjacent modules or their field wiring. The I/O can be intermixed in many configurations, limited only by the module types available (either 4, 8, or 16 circuits per module - see Chapter 6). Users need only purchase those modules required for their application; there is no fixed mix such as 12 in/8 out or 20 in/12 out.

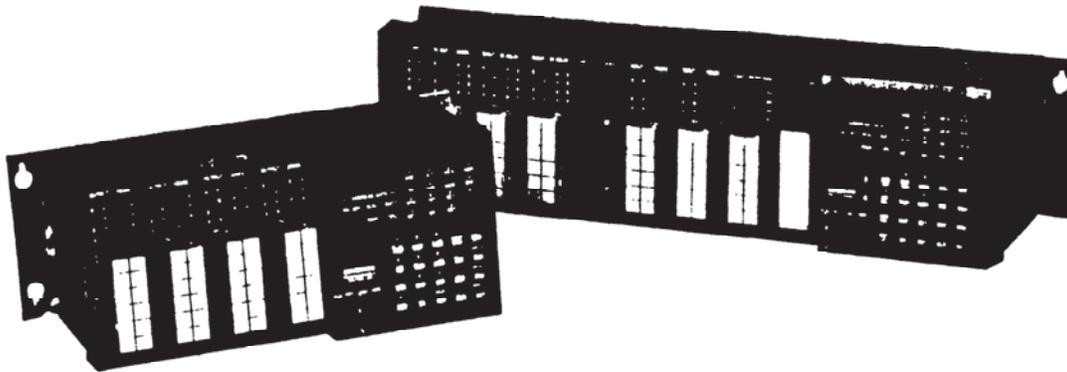
The hand-held programmer can be fixed onto the CPU for quick reference, removed and carried in a pocket for future use, connected to the CPU via a 5 foot (1.5m) cable for flexible temporary use or mounted on the outside of a panel or console. It is a simple pushbutton and display unit designed for easy transportation and rugged use. Programming can also be done with the Portable Programmer or with a Workmaster Industrial computer using Logicmaster 1 application software for Series One or Logicmaster 1F application software for Series One Model E, Series One Plus and Series One Plus 3.7K.

The Series One and Series One Plus PCs are designed to be very cost effective relay or other sequential control replacements. Despite the low cost, there is a lot of capability within the Series One and Series One Plus PCs.

The advantages and features offered by the Series One and Series One Plus PCs include:

- Small size - (64 I/O points in a 5-slot rack)      54 in<sup>2</sup> panel space, 5 1/2 in. maximum depth, 250 in<sup>3</sup> total volume
  - Large Memory Capacity -      Up to 1724 words of CMOS or PROM
  - Timers/Counters -      Up to 64 (4 digit, 9999 maximum preset)
  - Programming Language-      Boolean Based Relay Ladder Data Operations  
(Series One Plus Only)
  - Maximum I/O -      112 (Series One) - 168 (Series One Plus)
  - Latching relays      28 total
  - 16-bit Data Registers      64 (Series One Plus), 124 with TC references
  - Flexible I/O -      Any mix in groups of 4, 8 or 16
  - Replaceable I/O Modules
  - Portable Programmer-      Weight only 7 ozs.
  - Personal Password Security      Series One Plus and Series One Model E  
(Permits only Authorized Access To User's Programs)
- Internal and external indication of low battery
  - Reliable operation without fan from 0° to 60° C (convection cooling)
  - Retentive counters
  - Shift Registers (128) Built-in
  - Sequencers Standard - Up to 64, each with up to 1000 steps
  - Meets or exceeds NEMA ICS3-304 for noise rejection
  - Self Diagnostics
  - Internal Power for 24 V dc Inputs
  - Inexpensive
  - High Speed Counter
  - Remote I/O - Saves installation and wiring cost

a40793



**Figure 1-1. Series One/Series One Plus Programmable Controllers**

---

---

GEK-90842

### Series One Model E Programmable Controller

The Series One Model E PC is an enhanced version of the Series One PC that offers more capability while retaining all of the features and functions of the Series One PC. *The Model E requires an IC610CPU104 CPU module.* The additional features of the Series One Model E are:

- **Faster Scan Rate.** Typical scan times are:
  - 8 mSec for a 0.5K word program
  - 12 mSec for a 1.0K program
  - 15 mSec for a 1.7K word program
- **Password Protection.** As with the Series One Plus PC, this is a valuable feature in that it permits only authorized access to user programs stored in the PC memory. A unique 4-digit password is entered by the user and thereafter access to all functions (except monitor functions) is gained through a LOG ON sequence entered by the operator. After completion of required operations, a LOG OUT sequence returns the PC to password protection.
- **Faster Data Communications.** The Series One Model E PC requires the IC610CCM105 Data Communications Unit to communicate with external devices. Communications speed is noticeably faster when communicating with external devices, such as a host computer, than with a Series One PC using either the older version IC610CCM100 or the IC610CCM105 Data Communications Unit. Program upload or download time, and access time to I/O, and timer/counter data is reduced during communications sessions.

### Series One Plus 3.7K

The Series One Plus 3.7K PC is an enhancement of the Series One Plus PC which offers greater memory capability, while retaining all other features and functions of the Series One Plus PC. The Series One Plus 3.7K PC uses an IC610CPU106 CPU module that provides 3700 words of CMOS RAM or optional PROM memory (compared to 1700 words of memory available with a Series One Plus with an IC610CPU105 CPU module). A typical scan time for a 3.7K word program is 36 milliseconds.

This additional memory capability allows the Series One Plus 3.7K PC to more fully take advantage of programming with the data operations since those ladder logic programs that require numerous data operations also require more memory than do programs using the basic functions. The additional memory in a Series One Plus 3.7K PC provides the user with a great deal of capability in a small package.

An additional feature of the Series One Plus 3.7K PC is that timers can be programmed to be .01 second timers instead of .1 second. This is done by setting output 770 (SET 770) to the ON condition. The timer preset range with a .01 second duration selected is 0.01 to 99.99 seconds. When selecting timers to be .01 in this manner, care must be exercised to prevent your program from resetting coil 770, since this would cause all timers programmed to then be .1 second timers. Timer durations cannot be mixed, they can all only be .1 second or .01 second in the same program. Any Timer/Counter accumulate registers not being used for a Timer/Counter can be accessed and used as data registers as with the Series One Plus PC. Table 1-1 provides a summary of general specifications for both PCs.

Table 1-1. Series One Family General Specifications

<b>Operating Temperature</b>	0° to 60°C (32° to 140°F)
<b>Storage Temperature</b>	-10° to 70°C (14° to +158°F)
<b>Humidity (non-condensing)</b>	5 to 95%
<b>AC Power Required:</b>	
<b>IC610CHS101/110/130 Rack</b>	
<b>Voltage</b>	115V/230 V ac 15%
<b>Frequency</b>	47-63 Hz
<b>Maximum Load</b>	30 VA (CHS101/110) 36.7 VA (CHS130)
<b>Output Current, Maximum Individual</b>	1.4 A at 5 V dc, (0.4 A, CHS101) 0.8 A at 9 V dc, CHS110 (1.7 A, CHS130), (0.6 A, CHS101) 0.5 A at 24 V dc, (0.2 A, CHS101)
<b>DC Power Required</b>	
<b>IC610CHS114/134 Rack</b>	
<b>Voltage</b>	20.5 - 30 V dc (100% of capacity used) 18 - 30 V dc (90% of capacity used)
<b>Ripple</b>	10% of Input Voltage
<b>Output Current</b>	1.4 A at 5 V dc
<b>Maximum Individual</b>	0.8 A at 9 V dc, CHS114 (1.7 A, CHS134) 0.4 A at 24 V dc, CHS114 (0.5 A, CHS134)
<b>Maximum, Total (All Voltages)</b>	2.2 A, CHS114 (2.3 A, CHS134)
<b>Typical Battery life * (loaded)</b>	2-5 years
<b>shelf life * (no load)</b>	8-10 years
<b>*Depends upon operating temperature</b>	
<b>Memory Size and Type (16-bit words)</b>	700 words (CMOS) or 1724 words (CMOS or EPROM) 3700 words (Series One Plus 3.7K (CMOS or EPROM)
<b>Typical Scan Time per K of Memory</b>	20 mSec (0.5K)
<b>(Only memory programmed is scanned)</b>	Series One 40 mSec (1.0K)
<b>Overhead time must be added to the logic solution for total real scan, time overhead is typically 4 to 5 mSec.</b>	65 msec (1.7K) 8 mSec (0.5K)
	Series One Plus 12 mSec (1.0K)
	and Model E 15 mSec (1.7K)
	One Plus 3.7K 36 mSec (3.7K)
<b>Maximum I/O</b>	112 (Series One) 168 (Series One Plus)
<b>Internal Coils</b>	144
<b>Special Function Coils</b>	4
<b>Retentive Coils (Latches)</b>	28
<b>Timer/Counters</b>	64 (4-digit)
<b>Shift Register Stages</b>	128
<b>Sequencers</b>	64 at 1000 steps each
<b>Data Registers (Series One Plus and Series One Plus 3.7K)</b>	64 (16-bit), up to 124 with unused T/C references

GEK-90842

## Programmable Controller Concepts

When using a new product for the first time, there are always new concepts and terms to become familiar with. Although PC's are relatively easy to install, program, and apply, there are some simple principles to follow. Figure 1-2 illustrates a general block diagram of a Programmable Controller. Specific hardware components to illustrate this diagram will be described in Chapter 2.

pc-s1-83-0001

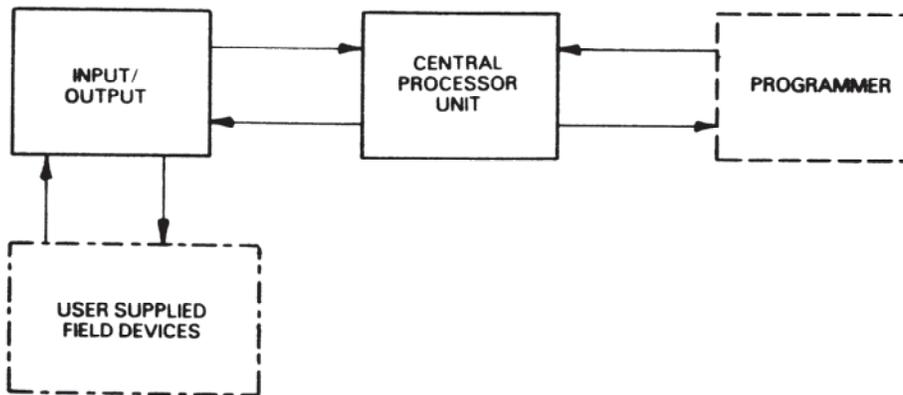


Figure 1-2. Basic PC Block Diagram

## Programming the Series One and Series One Plus PCs

The programming devices are used to enter the specific logic the user desires the PC to follow. This logic, to be described in detail in Chapter 5 is what makes the user's PC a unique unit, different from all others unless the identical logic is entered into another unit. Recording and reloading logic from one PC to another or to itself is also a standard feature with the programmers.

### Hand-Held Programmer

The programmer can display any previously entered logic, allow the user to edit it (make changes, add or delete portions of the logic), or display the current value of any internal timer or counter. It is a very valuable and powerful piece of peripheral equipment, for entry of logic, control system checkout, and troubleshooting. It can be permanently connected to the CPU or removed without disturbing the operation of the CPU. If removed, one programmer can service several CPUs. The exact quantity of CPUs is dependent upon the expected rate of usage but typically is between 5 and 20 CPUs.

### Portable Programmer

The portable programmer uses a liquid crystal display (LCD) screen to allow ladder logic programming for both the Series One and Series One Plus PCs. New programs can be created, previously entered ladder logic displayed and existing logic edited. An on-line monitor function is available when the Portable Programmer is used with the Series One Plus. Programs can be transferred to the PCs or stored

on cassette tape. The program can be printed using a standard parallel or Centronics compatible printer through the printer port located on the rear of the Portable Programmer. For detailed information refer to GEK-90846, which is the Portable Programmer User's Manual.

### **Logicmaster 1 Family Application Software**

Another option available for programming the Series One Family of PCs is the Workmaster industrial computer with Logicmaster 1 Family application software. This software allows you to write, edit, display, and print programs in ladder diagram format. You can view up to 7 lines of your ladder diagram program on one screen. Programs can be annotated by assigning names and nicknames to program elements, assigning labels to coils and adding explanations of program rungs or segments. Many programs can be stored on a single 3 inch diskette and each program can be assigned a unique name. For detailed information on using the Workmaster industrial computer with Logicmaster 1 application software refer to GFK-0075, which is the Logicmaster 1 Family Programming and Documentation Software User's Manual.

### **Series One Plus Program Protection**

The Series One Plus and Series One Model E PCs allow the user to enter a unique password with the hand-held programmer or the LCD Portable Programmer, which prevents unauthorized users or inadvertent program access. When a password is entered, all programmer functions are disabled except the I/O, T/C accumulated value and register monitor functions. In order to have access to all programmer functions when a password has been entered, a log on sequence must be entered.

### **Function of the Central Processing Unit**

The next element of the PC is the Central Processor Unit (CPU). The CPU is the "brain" behind all logical decision making. It reads in the status of the control system, makes decisions based upon the logic it has been provided, and then provides decisions to the actuating portion of the control system. The CPU also performs self checking of its internal operation to ensure reliable operation. If an error is detected, it will shut itself down. The logic entered by the programmer is actually stored in the CPU along with storage for the operation of timers and counters.

### **Memory Word Length**

The memory provided for this storage function is normally measured in K words, where K is an abbreviation for kilo or 1024. Typically, one word is required storage for each function such as a relay contact, timer preset or timer storage. These words can be of various lengths such as 16 bits, 8 bits, or even 4 bits, wherein a bit is the most elementary measurement and can have only two states (on or off). The word length is much like a ruler used to measure wire, sheet steel, or fabric. It can be a yard long (16 bits) or a foot (8 bits) or an inch (4 bits). Numerical values are for illustrative purposes only and do not represent exact ratios. Thus when quoting memory in K words (1K, 2K, 4K, etc.) always check to verify the word length. The Series One and Series One Plus PCs use the most common measurement, 16 bits per word.

---

---

GEK-90842

### Types of Memory for Program Storage

There are several types of memory used in PCs to store both logic and data. The two used in the Series One and Series One Plus PCs are CMOS and PROM. CMOS or CMOS RAM, which is an acronym for Complimentary Metal-oxide Semiconductor, Random Access Memory, provides a fast, low cost, low power memory that can be both examined (read) and also changed (written) easily. However, it is volatile, which means that it can lose its content if power is removed. To avoid reloading memory (and losing counts and system status) every time power is turned off, the CMOS memory is usually provided (as it is in Series One and Series One Plus) with a back-up battery to maintain its content (not system operation) when power fails. Due to the low power drain of CMOS technology, a single new lithium battery can maintain memory without application of power for up to 2 to 5 years. The battery is not used when the power is applied and the system is operating normally. Its storage or shelf life is many years, typically 8 to 20 years.

The second memory is PROM(Programmable Read only Memory) that again is fast, relatively low cost, and retentive upon loss of power. However, this memory cannot be easily changed. It can be examined (read) at anytime, but to change (write) it requires some special action on the part of the user. In this system, the PROM must be cleared of all previous contents (new PROM, or erased with an intense ultraviolet light) and then placed into a special loader. A previous logic program developed in CMOS is then written into the PROM. Finally, the PROM is removed from the loader and placed into the CPU.

### Function of the Input/Output Circuitry

The final element of the PC is the Input/output section. Electrical noise such as spikes on the power lines, inductive "kick-back" from loads, or interference picked up from field wiring is very prevalent in industrial applications. Since the CPU operates at relatively low voltage levels (typically 5 volts), this noise would have serious impact on its operation if allowed to reach the internal circuits of the CPU.

The I/O section, both inputs and outputs, protects the CPU from electrical noise entering via the I/O modules or wiring. The I/O section is where status signals are filtered to remove noise, voltage levels are validated, and where decisions made by the CPU are put into operation. Inputs provide their status to a storage area within the CPU and outputs are driven from similar stored status in the CPU.

In general, the I/O section is modular in design and can accommodate a variety of signals. A complete discussion of the types and capacities available for both the Series One and Series One Plus PCs is provided in Chapter 6. The specific type of module (e.g. 115 V ac or 24 V dc) is usually determined by the field device the user selects. Decisions such as number of 115 V ac solenoids, 24 V dc solenoids, motor starters, limit switches (their voltages), control panel lamps (what voltage), pushbuttons, and external relays have a major impact on the configuration of any PC. These parameters should be established as early as possible in the overall design of the control system. Of course, being a flexible device, the PC configuration either on paper or in hardware, can be changed if requirements change. Typically, the user provides the field devices, wires them to the I/O section, and provides the power source to operate them.

### UL Listed Products

Several Series One family products are available that have been tested and approved by the Underwriters Laboratory (UL). These UL listed products should be used in installations where UL listed products are required. The UL listed products include the Series One CPU (IC610CPU101, revision C), 5-slot

rack (IC610CHS111), 115 V ac Input module (IC610MDL135), Relay Output module (IC610MDL181), and a 115 V ac Output module (IC610MDL185). Specifications and wiring information for the UL listed modules can be found in chapter 6 of this manual.

*When installing a system requiring UL approval, do not mix non UL listed products with the UL listed products.*

### Communicating With Other Devices

An available Data Communications Unit (DCU) allows the Series One and Series One Plus to communicate with external devices. The Series One PC uses the IC610CCM100 DCU, while the Series One Model E, Series One Plus and Series One Plus 3.7K PCs use the IC610CCM105 DCU. These devices can be other programmable controllers, computers, or other smart devices. User programs and I/O information in a Series One or Series One Plus PC can be uploaded and downloaded to or from any master device that supports the Series Six CCM2 protocol as defined in the Series Six Data Communications Manual, GEK-25364. For detailed information on how to use a DCU in a Series One or Series One Plus PC system, refer to the Series One Data Communications Manual, GEK-90477.

### Remote I/O for Series One and Series One Plus

Another option available for a Series One or Series One Plus PC system is Remote I/O. By using Remote I/O, I/O modules can be located in a rack convenient to the input sensors or the output devices being controlled by the PC at a distance of up to .6 miles (1 km) from the PC. This is accomplished by installing a Link Local module in the CPU rack, a Link Remote module in the distant I/O rack and connecting them through a single twisted-pair cable. For detailed information on using Remote I/O in a Series One or Series One Plus PC system, refer to the Series One Remote I/O Manual, GEK-90507.

### PC Terminology

To summarize the preceding discussion of Programmable Controller concepts, table 1-2 provides a definition of terms discussed above that you should be familiar with, relating to PCs. A more complete list of terms is provided in a glossary at the end of this manual.

**Table 1-2. Common PC Terminology**

TERM	DEFINITION
PC	Programmable Controller or Programmable Logic Controller. An industrial control device using microprocessor technology to perform logic decision making with relay ladder diagram based programming.
Programmer	A device for entry, examination and alteration of the PC's memory including logic and storage areas.
Logic	A fixed set of responses (outputs) to various external conditions (inputs). All possible situations for both synchronous and non-synchronous activity must be specified by the user. Also referred to as the program.
CPU	Central Processor Unit - the physical unit in which the PC's intelligence resides. Decision making is performed here.
Memory	A physical place to store information such as programs and/or data.

GEK-90842

Table 1-2. Common PC Terminology - Continued

TERM	DEFINITION
<b>K</b>	An abbreviation for kilo or exactly 1024 in the world of computers. Usually related to 1024 words of memory.
<b>Word</b>	A measurement of memory usually 16, 8, or 4 bits long.
<b>CMOS</b>	A read/write memory that requires a battery to retain content upon loss of power.
<b>PROM</b>	A read only memory that requires a special method of loading, but is inherently retentive upon power loss.
<b>I/O</b>	Input/Output - that portion of the PC to which field devices are connected. Isolates the CPU from electrical noise.
<b>Noise</b>	Undesirable electrical disturbances to normal signals generally of high frequency content.
<b>Inputs</b>	A signal, typically ON or OFF, that provides information to the PC.
<b>Outputs</b>	A signal typically ON or OFF, that originates from the PC with user supplied power that controls external devices based upon commands from the CPU.
<b>Modules</b>	A replaceable electronic subassembly usually plugged in and secured in place but easily removable in case of fault or system redesign.
<b>Field Devices</b>	User supplied devices typically providing information to the PC (Inputs: pushbutton, limit switches, relay contacts etc.) or performing PC tasks (Outputs: motor starters, solenoids, indicator lights, etc.).

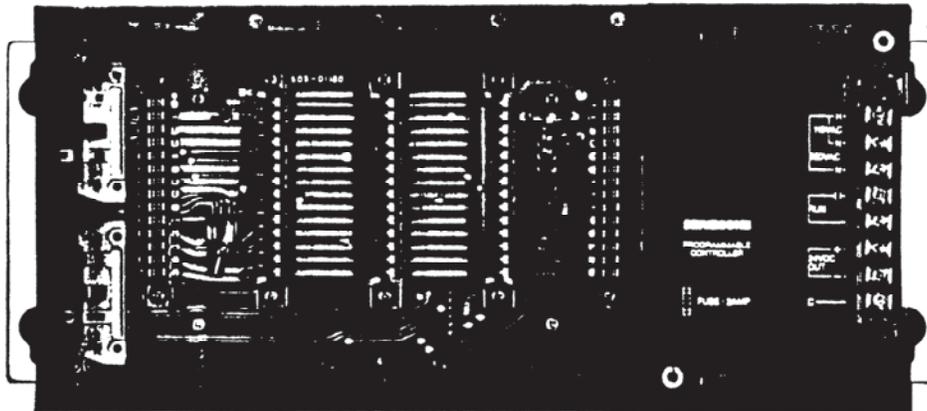


GEK-90842

**Rack Description**

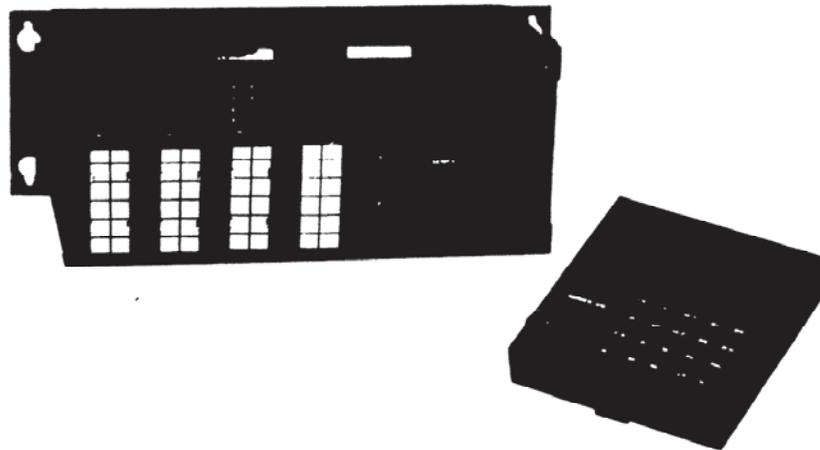
The Series One and Series One Plus PCs are provided as a family of racks into which modules can be inserted. Each rack contains a power supply to the right and space for up to either 5 or 10 modules (see figure 2-1). Racks are available in 6 versions, 5 or 10-slot that accept 115/230 V ac input power, 5 or 10-slot that accept 24 V dc, a UL listed 5-slot rack, IC610CHS111 which accepts only a 115 V ac power source, and a low-cost 5-slot rack (IC610CHS101) that does not include an expansion part, 24 V dc terminals for external use, or a RUN relay. Each supplies internal power to the modules inserted into the rack. Mounting is provided by the brackets each with two keyholes at the rear of the rack. All racks are similar. The differences being the function of the rack as determined by the placement of modules by the user, the number of modules which may be inserted into a rack, and the input power required. Figure 2-2 illustrates typical modules for the Series One Family of PCs, illustrating in hardware, the block diagram concept of figure 1-1.

a40535

**Figure 2-1. Typical Rack****Module Location in Rack**

The CPU module is always placed adjacent to the power supply in the first rack; it contains the microprocessor and required memory storage. There is a connector on the CPU to which the programmer is attached when mounted on the rack. As an option, between the CPU and programmer a 5 foot (1.5m) cable can be installed for more flexible operation. The remaining slots can contain I/O modules in any mix of inputs versus outputs or voltage levels desired by the user for his particular application. All modules as well as the programmer are secured to the rack by two snap-locks which can be released by squeezing the module top and bottom toward the center (see Chapter 3 for additional installation details).

a40019



**Figure 2-2. Typical Hardware**

### Heat Dissipation

The rack is designed to dissipate internal heat through convection cooling only and does not require a fan for forced air cooling. However, to ensure efficient operation, free air flow should not be inhibited at the top and bottom of the unit. A minimum of 3 inches (75mm) is recommended at the top and 4 inches (100mm) at the bottom with 6 inches (150mm) between racks. Both sides should be free of obstacles to allow easy removal of the unit, approximately 3 inches (75 mm) from each side excluding the mounting brackets is recommended. Furthermore, the unit should be mounted horizontally as shown in figure 2-3 and not inverted nor rotated 90°. If not oriented as shown, derating of the maximum ambient temperature specification would need to be considered. Placements of other sources of large volumes of heat near the units should also be avoided, especially directly below the rack. For reliable operation, the air entering the bottom of the rack should not be at a higher temperature than 60°C (140°. Wiring to the I/O modules and the power supplies should be placed so as to avoid blocking the air flow, yet provide a suitable service loop to allow easy removal of modules with wiring attached. Wires should be tied to maintain their order in the event they must be disconnected during module replacement.

### Adding Racks

If more I/O points are required than one rack can contain, additional racks can be installed similar to the first unit previously discussed (IC610CHS101 cannot be used as an expansion rack). At the left of each rack are two connectors used to connect to additional I/O. An 18 inch (460mm) cable is available to connect the rack containing the CPU module to the first I/O expansion rack. The ends of this cable are marked "CPU" and "EXP" (Expander). The CPU end is plugged into the bottom connector at the first rack and the EXP end similarly connected to the second rack using the top connector (see figure 2-5). If a third rack is used, another cable links the second rack to the third rack, the CPU end is inserted into the bottom connector on the second rack and the EXP end into the top connector on the last rack. Within these added racks, I/O modules can be inserted in any order desired, up to five modules in a 5-slot rack and up to 10 modules in a 10-slot rack. No additional CPU modules can be installed, nor are required.

a40018

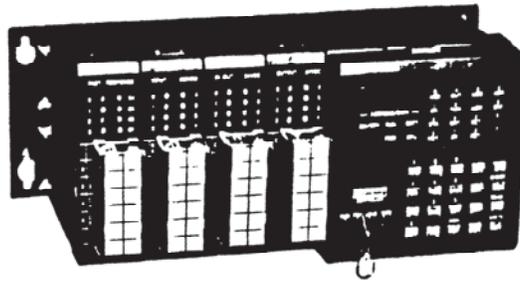


Figure 2-3. Typical (5 Slot) Rack

### 10-Slot Rack

The 10-slot rack provides in a single housing, as shown in figure 2-4, the same number of I/O module slots as two 5-slot racks. An added advantage of the 10-slot rack is easier and less expensive installation since fewer racks have to be mounted and wired. A system can be configured in many ways to contain various quantities of I/O. Refer to Appendix B, which contains examples of valid I/O rack configurations.

### Rack Mount Brackets

Rack mount brackets (IC610CHS191) are available to adapt the 10-slot racks for mounting in 19 inch racks. With the brackets assembled on either version of a 10-slot rack, the rack can be mounted on standard mounting rails in 19 inch cabinets and consoles.

a40792

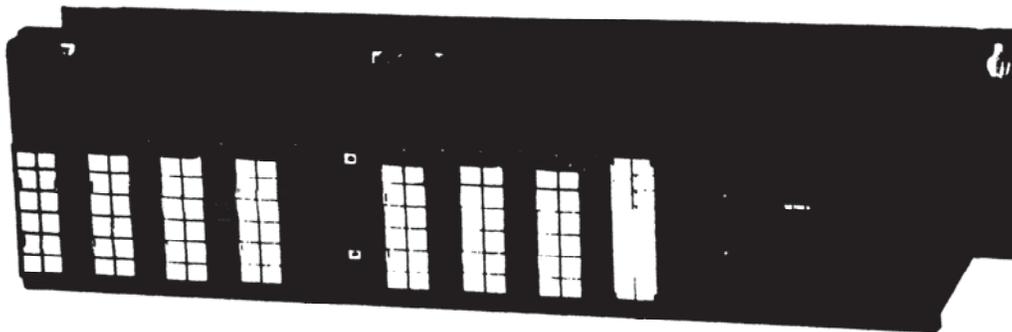
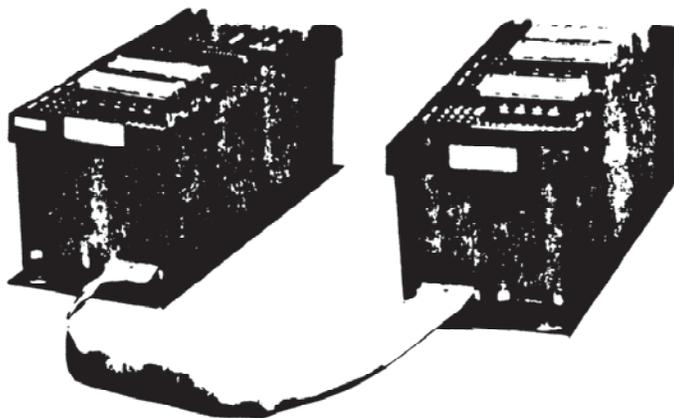


Figure 2-4. Typical (10 Slot) Rack

83-pc-35mm-4-1-31a



**Figure 2-5. Expander Cable Installation**

## **Programmer Tape Port**

The hand-held programmer, when installed on the CPU module, provides an auxiliary tape port for memory transfers to peripheral devices. Tape recordings of user programs can be made on virtually any audio cassette recorder and once made can be used to initialize any CPU to that program. Thus programs can be made once and transported to other CPUs without manually being re-entered. Of course, once entered they can be edited if additional tailoring is required. In the unlikely event that a CPU fails, a replacement can be installed and quickly reloaded to perform specifically the task its predecessor was accomplishing, if a tape record was made. The tape recorder functions are discussed in detail in Chapter 4, PC Operation.

## **Programmer Mount Assembly**

A Programmer Mount Assembly (IC610PRG190) is available that can be used to mount and protect the hand-held programmer on the outside of a panel or console. A hand-held programmer, when mounted externally, can be used as an operator interface unit to change timer or counter presets, monitor timer or counter current values, monitor 16 consecutive I/O points, monitor the entire contents of the user program, and, with a Series One Plus PC, monitor register contents. The programmer mount assembly includes a mounting bezel, a clear plastic cover, and a cable fastener. In addition to the mounting assembly, a shielded, round CPU/Programmer cable (Catalog No. IC610CBL102), designed specifically for mounting the hand-held programmer away from the CPU rack must be ordered separately for use when installing the hand-held programmer in this manner.

The Programmer Mount Assembly bezel installs on the outside of a panel or console with only four screws. The hand-held programmer snaps into the bezel using its two snap-locks. A clear plastic cover then fits over the bezel and programmer, thereby protecting it from its industrial environment. The Programmer Mount Assembly can also be used as a table top stand for the hand-held programmer by

GEK-90842

mounting four rubber feet, which are included with the assembly, on the reverse side of the bezel using the panel mounting holes.

a40148

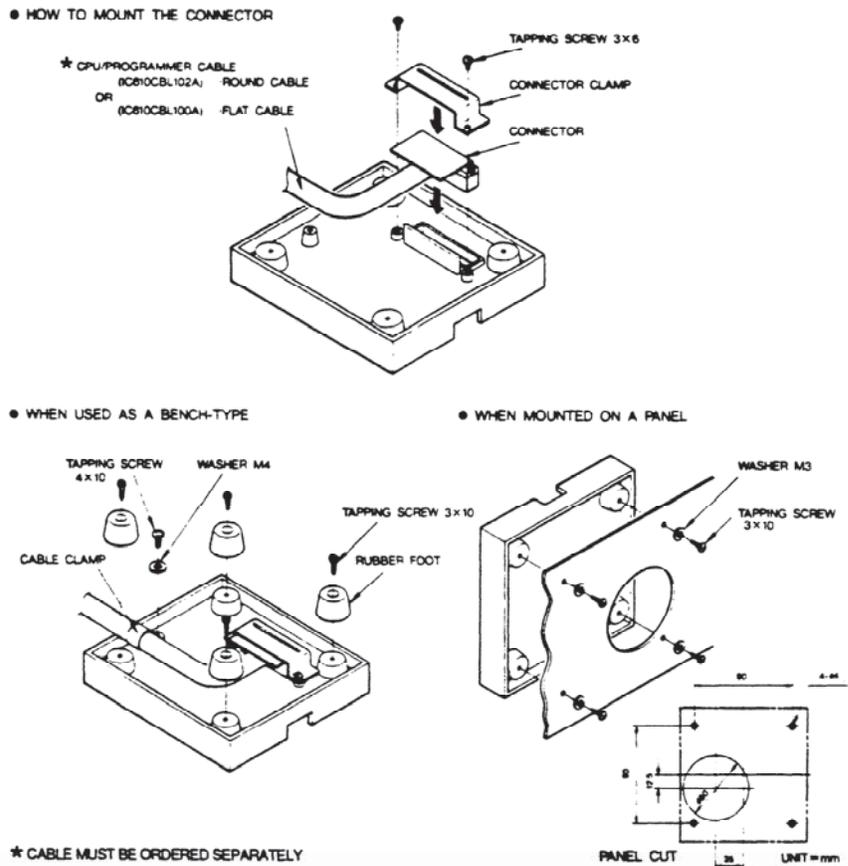


Figure 2-6. Programmer Mount Assembly

Table 2-1 lists the Series One/Series One Plus PC catalog numbers and nomenclature for the various modules, cables, peripherals, and accessories. For detailed information on I/O module specifications and wiring diagrams, see Chapter 6.

Table 2-1. Series One/Series One Plus Catalog Numbers

DESCRIPTION	CATALOG NUMBER
CPU, Series One (700 words of CMOS Memory, standard) Version C is UL listed	IC610CPU101
CPU, Series One Model E (Enhanced Version of CPU101)	IC610CPU104
CPU, Series One Plus (700 words of CMOS Memory, std)	IC610CPU105
CPU, Series One Plus 3.7K (3700 words of CMOS Memory)	IC610CPU106
I/O Expander Cable	IC610CBL101
Programmer with Keylock	IC610PRG100
Programmer w/Keylock (Required for Series One Plus)	IC610PRG105
CPU to Programmer Cable, 5' (1.5m)	IC610CBL100

Table 2-1. Series One/Series One Plus Catalog Numbers - Continued

DESCRIPTION	CATALOG NUMBER
Rack, 115/230 V ac Power Source, 5-slot	IC610CHS110
Rack, 115/230 V ac Power Source, 5-slot (no expansion, 24 V dc Terminals, or Run Relay)	IC610CHS101
Rack 115 V ac Power Source, 5-slot, UL listed	IC610CHS111
Rack w/24 V dc Power Source, 5-slot	IC610CHS114
Rack, 115/230 V ac Power Source, 10-slot	IC610CHS130
Rack, w/24 V dc Power Source, 10-slot	IC610CHS134
Filler Module	IC610MDL100
24 V dc Sink Input, 8 Circuits	IC610MDL101
24 V dc Input/Output, 4 Inputs/4 Outputs	IC610MDL103
24 V dc Sink Input/Relay Output, 4 Inputs/4 Outputs	IC610MDL104
Thumbwheel Interface	IC610MDL105
24 V dc Sink Input, 16 Circuits w/LEDs	IC610MDL106
I/O Interface Cable 10' (3m)	IC610CBL105
24 V dc Sink Load Input, 16 Circuits	IC610MDL107
24 V ac/dc Source Input, 8 Circuits	IC610MDL111
24 V ac/dc Source Input, 16 Circuits	IC610MDL112
115 V ac Input, 8 Circuits	IC610MDL125
115 V ac Isolated Input, 4 Circuits	IC610MDL126
230 V ac Input, 8 Circuits	IC610MDL127
115 V ac Input, 6 Circuits (UL listed)	IC610MDL135
24 V dc Sink Output, 8 Circuits	IC610MDL151
24 V dc 2 Amp Sink Output, 4 Circuits	IC610MDL153
24 V dc 2 Amp Sink/Source Output, 4 Circuits	IC610MDL154
24 V dc Source Output, 8 Circuits	IC610MDL155
24 V dc Sink Output, 16 Circuits w/LEDs	IC610MDL156
24 V dc Sink Output, 16 Circuits	IC610MDL157
24 V dc Source Output, 16 Circuits	IC610MDL158
115/230 V ac Output, 8 Circuits	IC610MDL175
115/230 V ac Isolated Output, 4 Circuits	IC610MDL176
Relay Output, 8 Circuits	IC610MDL180
Relay Output, 5 Circuits (UL listed)	IC610MDL185
Relay Output, 16 Circuits	IC610MDL182
115 V ac Output, 6 Circuits (UL listed)	IC610MDL181
High Speed Counter	IC610MDL110
I/O Interface Cable (High Speed Counter)	IC610CBL107
Fast Response I/O	IC610MDL115
I/O Simulator, 8 Inputs	IC610MDL124
Printer Interface Unit	IC610PER151
PROM Writer Unit	IC610PER154
Accessory Kit	IC610ACC120
Lithium Battery	IC610ACC150
PROM Memory (4 Chips), Series One, Series One Junior	IC610ACC151
CMOS Memory (4 Chips), Series One	IC610ACC152
PROM Memory (4 Chips), Series One Plus	IC610ACC155
CMOS Memory (4 Chips), Series One Plus	IC610ACC156
Rack Mount Brackets	IC610CHS191
Programmer Mount Assembly	IC610PRG190

GEK-90842

Table 2-1. Series One/Series One Plus Catalog Numbers - Continued

DESCRIPTION	CATALOG NUMBER
Remote CPU/Programmer Cable	IC610CBL102
Data Communications Unit	IC610CCM105
I/O Link Local	IC610CCM110
I/O Link Remote	IC610CCM111

## Hardware Requirements

The Series One and Series One Plus PCs are an excellent relay and timer/counter replacer or substitute for other sequential type control devices, such as drum or stepping switch based systems. However, one question always arises - how much hardware do I need to buy? The answer varies extensively based upon different applications and their attendant complexities. Areas of concern include amount of memory, mix of inputs versus outputs, voltages of I/O, and physical size of the Series One or Series One Plus PC system. The following steps are guides to estimate the requirements of the Series One or Series One Plus PC system. With a little experience, estimating required components will become second nature. If you require assistance, please contact your local GE Fanuc Automation distributor who handles the Series One Family of PCs.

### NOTE

CPU Module IC610CPU101A is different from 101B and 101C in that 101A has a 3 digit preset for Timers and Counters, while the updated modules, 101B and 101C have a 4 digit preset. The 101A method of monitoring the accumulated value of Timers and Counters is also different. With the 101A module, each individual Timer or Counter must be accessed by using the sequence SHF, 6XX, MON for each Timer/Counter to be monitored. With the 101B or the 101C CPU module, the keys NXT or PRV will move the monitor display to the next or previous Timer or Counter.

## System Estimating

The Series One and Series One Plus PCs are provided with 700 words of CMOS memory as a standard feature. Expansion to 1724 words is possible by adding a CMOS memory chip. If the unit is to be PROM based, the program can be up to 1724 words. The Series One Plus 3.7K PC provides the user with 3700 words of CMOS memory as a standard feature. For average complexity relay replacement, 700 words should be adequate for up to 64 I/O, and the 1724 should be adequate for up to 168 I/O. If the logic is considered more complex than that used as examples in this manual, a representative sample (10 to 15%) of the logic should be programmed. From the amount of memory (on paper) the sample requires, the total memory requirements can be estimated. If in doubt, obtain the optional memory with your unit to ensure simple system design.

The key to many of the answers (cost, physical size, memory requirements, etc.) is the I/O structure. If a design exists such as shown in figure 2-7, assume that all the relays and timers are enclosed within a box; these are the elements to be replaced. This figure is for illustrative purposes only; no indication is given that it performs any real functions. Passing through this box are wires from switches, auxiliary contacts, overload relays, etc.; these are inputs to the control system. There are wires connecting to loads or actuating devices such as solenoid valves, motor starters, indicator lights, etc.; these are outputs

from the control system. The power lines (e.g., 115 V ac and 24 V dc) are not considered I/O, but are important to estimating the number of I/O modules required.

Use the following steps to estimate rack and module requirements; figure 2-7 is used as an example:

1. Add total number of inputs and outputs separating them by type and voltage. (e.g. (4) 115 V ac and (5) 24 V dc inputs plus (3) 115 V ac and (1) 24 V dc output).
2. Divide each separate category by 8 (assume 8 circuit modules), (e.g. (1) 115 V ac and (1) 24 V dc input modules and (1) 115 V ac and (1) 24 V dc output modules). See Chapter 6 for I/O that uses 4 or 16 circuits per module.
3. Add total I/O modules (e.g. 4 I/O modules). One 5-slot rack is required for 1-4 I/O modules, two 5-slot racks for 5-9, and three 5-slot racks for 10-14.
4. If 10-slot racks are used one 10-slot rack is required for 1-9 I/O modules, two 10-slot racks or a 10-slot and a 5-slot rack will contain 1-14 I/O modules.
5. Each additional rack also requires an I/O expander cable. Empty slots should be covered by blank filler plates.
6. Estimate memory requirements as follows:

Total I/O Points	Estimated Memory
1-64	Basic Unit with 700 words
65-168	Add 1K Memory Chip

7. The following optional hardware is available and should be considered when configuring a system:

Hand-Held Programmer	PROM Writer Unit
Portable Programmer	Printer Interface Unit
Extender Cable	Thumbwheel Interface Unit
Programmer Mount Assembly	Timer/Counter Setpoint Unit (Series One Plus)

If the system has not yet been designed, the same basic technique can be used. Inputs are signals the PC will require to perform its assigned functions. Any device or person regardless of intelligence can not respond to events it does not know occurred. Plan to provide the PC all the information you would require to perform the same function. Make a list of those inputs, including source and voltage level (if currently defined). Outputs, on the other hand, are devices the PC will use to perform its functions. Again, even if the PC knows it must do something, if it is not given control over these actuating devices, it can not maintain proper control. Make a list of these devices including their voltage levels and current/power requirements. Use the above steps with the list of I/O devices. In any case, when

GEK-90842

estimating system requirements, review the following standard features as discussed in later chapters to define their impact on your control needs:

- Up to 64 Timers and Counters (4 digits each)
- 128 Stage Shift Register
- Up to 64 Sequencers Each With Up to 1000 Steps
- 28 Latched Relays
- Data Operations (Series One Plus and Series One 3.7K)
- 64 Sixteen Bit Data Registers (Series One Plus and Series One 3.7K)
  - Up to 124 Data Registers are possible when unused T/C references are used as Data Registers.

a42153

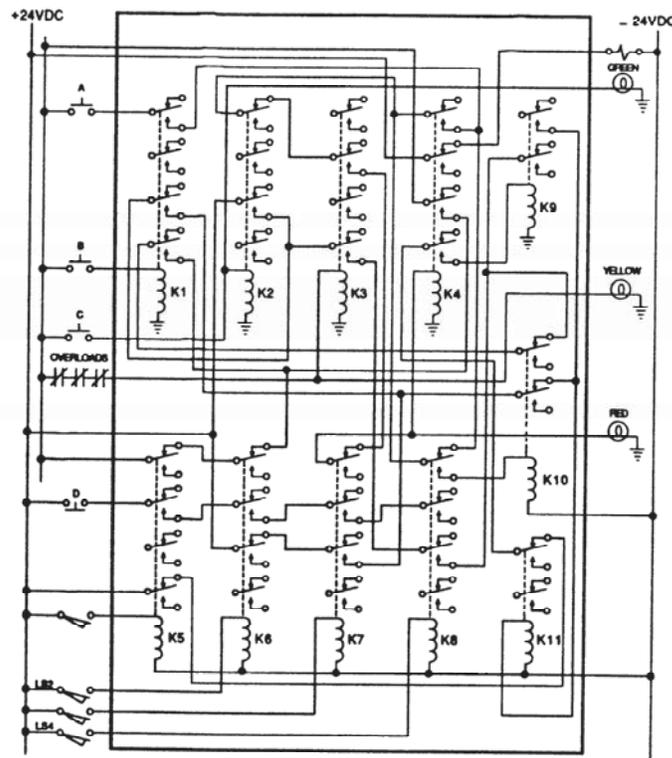


Figure 2-7. Example of Relay Control

### Peripheral Devices Supporting Series One Family of PCs

Several peripheral units are available to support a Series One or Series One Plus Programmable Controller. A basic description of these units is provided in the following paragraphs. For a more detailed description of the use and operation of these units, refer to Chapter 4, Operation, in this manual.

### Data Communications Unit

The Data Communications Unit (DCU), IC610CCM100/105, provides the ability for external devices to communicate with the Series One, Series One Model E, Series One Plus or Series One Plus 3.7K PC. Series One Model E, Series One Plus and Series One Plus 3.7K PCs require the CCM105 DCU. These devices function as a host to the Series One or One Plus PC and can be other programmable controllers, computers, or other smart devices. User programs and I/O information in the Series One or One Plus PC can be uploaded and downloaded to or from any master device that supports the Series Six CCM2 (Communications Control Module, Version 2) master/slave protocol as defined in GEK-25364, which is the Series Six Data Communications Manual. The Series One or Series One Plus can only function as a slave device during a communications session.

a40537

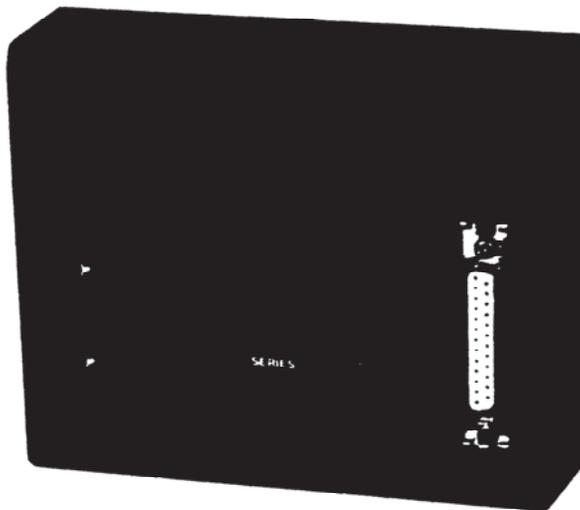


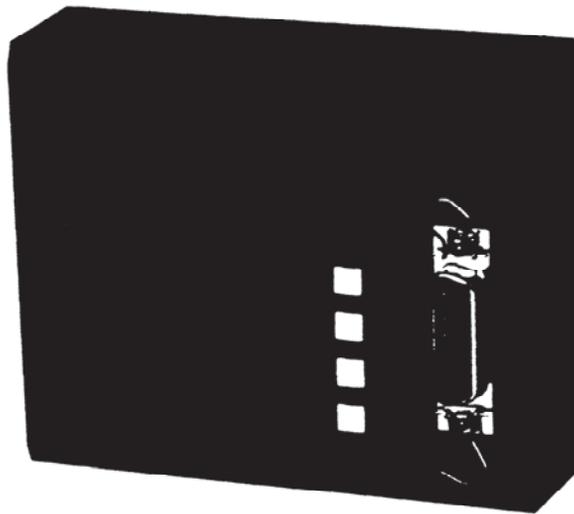
Figure 2-8. Data Communications Unit

GEK-90842

**Printer Interface Unit**

The Printer Interface Unit, IC610PER151, is a compact, easy to use device that attaches to the Series One, One E, One Plus or One Plus 3.7K PC in the same manner as the programmer. This peripheral interfaces to many readily available personal computer printers and provides a means of obtaining a hard-copy printout of the user program in either boolean or ladder diagram format. Version B works with Series One, One E and One Plus PCs. A higher revision will be available that will also work with the Plus 3.7K PC.

a40538

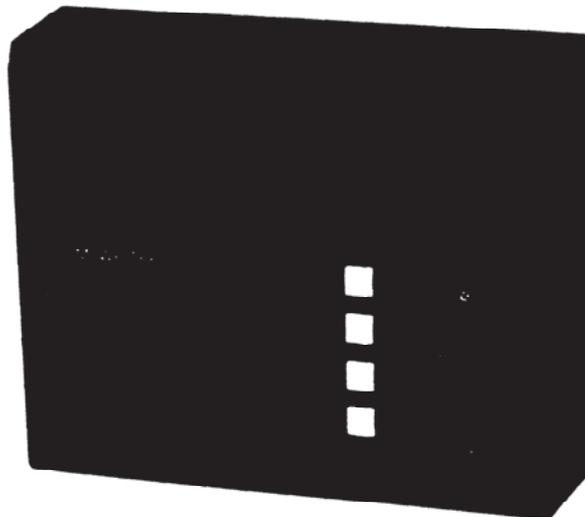


**Figure 2-9. Printer Interface Unit**

### PROM Writer Unit

Also available is a PROM Writer unit, IC610PER154, which allows the user program in CMOS memory to be transferred to PROM memory, thereby providing a convenient method of non-volatile (permanent) storage for those programs. The PROM Writer unit is a compact, easy to use unit which attaches directly to the PC in the same manner as the programmer. In addition to providing a means of non-volatile storage, an added advantage of PROM memory is that several PROMs can be programmed, each containing a different program, for use as required. Version B of this unit replaces version A and can be used with a Series One Plus PC as well as Series One and Series One Junior PCs. Version B will not work with the Series One Plus 3.7K PC. A revised version will be available that can be used for all Series One Family PCs.

a40539



**Figure 2-10. Prom Writer Unit**

<b>PROGRAMMABLE CONTROLLER</b>	<b>PROM</b>
Series One/One E	2732A-2
Series One Plus/3.7K	27256-25

GEK-90842

### Specifications for Installation

The Series One and Series One Plus PC can be easily installed in any NEMA panel or similar metal surface. Figure 3-1 provides details on the mounting of up to three racks except for input voltage requirements to form a single PC system. Since a completed 5-slot rack weighs less than five pounds (10-slot rack, less than 7 pounds), it can be easily installed by one technician. Table 3-1 provides specifications to be met during installation.

**Table 3-1. Series One Installation Specifications**

<b>Rack Size</b>	
<b>5-slot</b>	11.4" x 4.7" x 5.5" (290 x 120 x 140mm)
<b>10-Slot</b>	18.3" x 4.7" x 5.5" (465 x 120 x 140mm)
<b>Completed Rack Weight</b>	
<b>(less wiring)</b>	4.5 lbs (2.0 Kg), 5-Slot 6.5 lbs (2.9 Kg), 10-slot
<b>Ambient Temperature</b>	0° to 60°C (32° to 140°F)
<b>Storage Temperature</b>	-10° to 70°C (14° to 158°F)
<b>Humidity</b>	5-95% (Non-Condensing)
<b>AC Power Required:</b>	
<b>IC610CHS101/110/130 Rack</b>	
<b>Voltage</b>	115V/230 V ac 15%
<b>Frequency</b>	47-63 Hz
<b>Maximum Load</b>	30 VA, CHS101/110 (70 VA, CHS130)
<b>Output Current</b>	1.4 A at 5 V dc (0.4 A CHS101)
<b>Maximum Individual</b>	0.8 A at 9 V dc, CHS110 (0.6 A, CHS101; 1.7 A, CHS130) 0.5 A at 24 V dc, (0.2 A, CHS101)
<b>DC Power Required</b>	
<b>IC160CHS114/134</b>	
<b>Voltage</b>	20.5 - 30 V dc (100% of capacity used) 18 - 30 V dc (90% of capacity used)
<b>Ripple</b>	10% of Input Voltage
<b>Output Current</b>	1.4 A at 5 V dc
<b>Maximum Individual</b>	0.8 A at 9 V dc, CHS114 (1.7 A, CHS134) 0.4 A at 24 V dc, CHS114 (0.5 A, CHS134)
<b>Maximum, Total</b>	2.2 A, CHS114 (2.9 A, CHS134)
<b>(All voltages)</b>	
<b>Run relay</b>	250 V, 4 amp, Resistive Load (Not Present on CHS101)
<b>Vibration</b>	Meets JIS C 0911 IIB Class 3 Tested to MIL STD 810C Method 514.2
<b>Shock</b>	Meets JIS C 0912
<b>Noise Immunity</b>	Meets NEMA ICS3-304

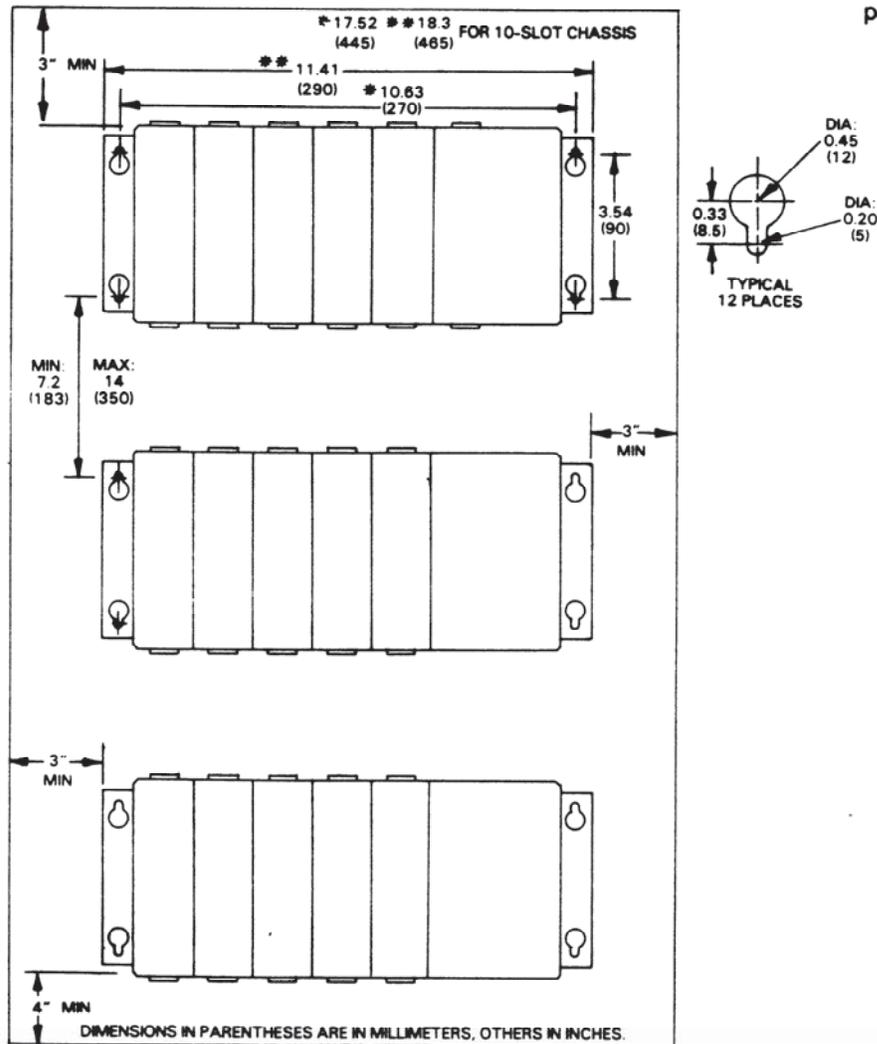


Figure 3-1. Rack Mounting Dimensions for Proper Heat Dissipation

## Installation

Unpack each unit carefully and retain any instructions shipped with the units. Two spare fuses are attached to the top of each rack; they should be removed and retained for future use. The racks can be installed either with no modules or with modules installed. The following steps will assist in organizing and simplifying the installation of a Series One or Series One Plus PC System.

### Racks

1. Using the rack as a template, mark where mounting holes are to be drilled.
2. Drill the four mounting holes (1/4" (6mm) if using pass through bolts, or 3/16" (5mm) if using tapped holes).
3. Insert top 2 bolts (3/16" X 1-1/2" or 5mm X 40mm), put unit in place, and loosely secure with washers, lock washers and nuts.

OR

GEK-90842

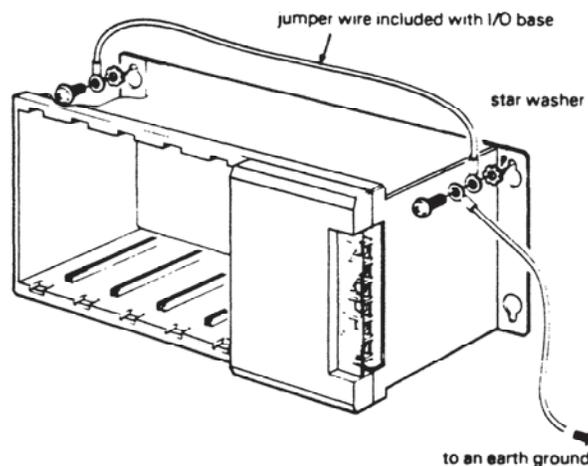
4. Tap holes and insert top two bolts. Place unit onto mounting bolts and loosely secure.

**NOTE**

When inserting the 2 top bolts, attach the green jumper as shown in figure 3-2. This jumper wire is packed with each rack and provides a method of grounding the rack when an earth ground (such as the shield or hard wire in the power cord) is also connected to a mounting bolt.

5. The power supply ground connection is made to one of the mounting bolts (step 3 or 4 above). A jumper wire is included with each rack to interconnect the mounting brackets. The "C" terminals on the power supplies are interconnected when more than one rack is used. Otherwise, the "C" terminal is not connected to anything in a single rack system.

a40147

**Figure 3-2. Recommended Rack Grounding**

6. Complete the installation of the bottom two bolts and tighten all mounting hardware. Power supplies are shipped installed in each base.
7. If additional racks are to be used, repeat steps 1-4 above. If only one base unit is used, go to step 15.

**NOTE**

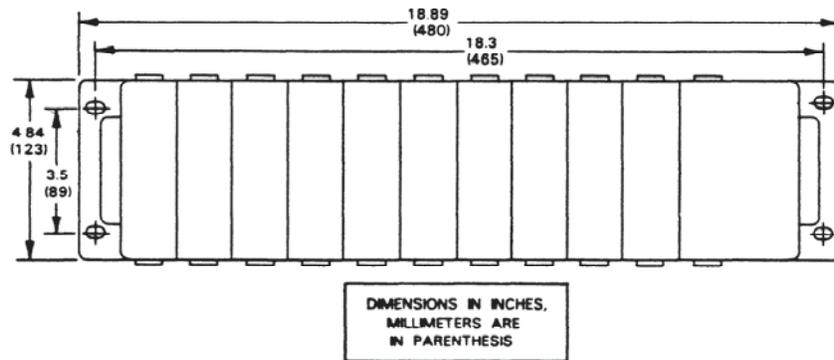
When drilling or tapping holes ensure that metal chips do not enter unit already installed. Clear work area before installing base units.

8. Obtain flat ribbon cable used to interconnect racks. Locate end marked "To CPU", remove dust cover from bottom connector of first (CPU) rack, and insert cable connector (see figure 3-4) fully into receptacle until locking tabs capture connector.
9. Fold cable as shown on figure 3-7. Remove dust cover from top connector of second rack and insert opposite end of cable marked "To Expander."
10. Secure cable in place with wire wraps or cable ties.

GEK-90842

11. If a third rack is used, repeat steps 8-10 with CPU end of cable in bottom connector on second rack and Expander end in top connector of third and last rack.
12. If a 10-slot rack (IC610CHS130 or IC610CHS134) is to be included in a system, it can be mounted on standard mounting rails in 19 inch cabinets and consoles by attaching the rack mount brackets. Two adapter brackets and hardware required for assembly are included in the bracket package, IC610CHS191.

a40214



**Figure 3-3. 10 Slot Rack, 19 Inch Mounting Dimensions**

a40280



**Figure 3-4. I/O Expansion Cable Connection**

13. Unused rack connectors such as at the top of the CPU unit and the bottom of the last rack should retain their dust covers.

GEK-90842

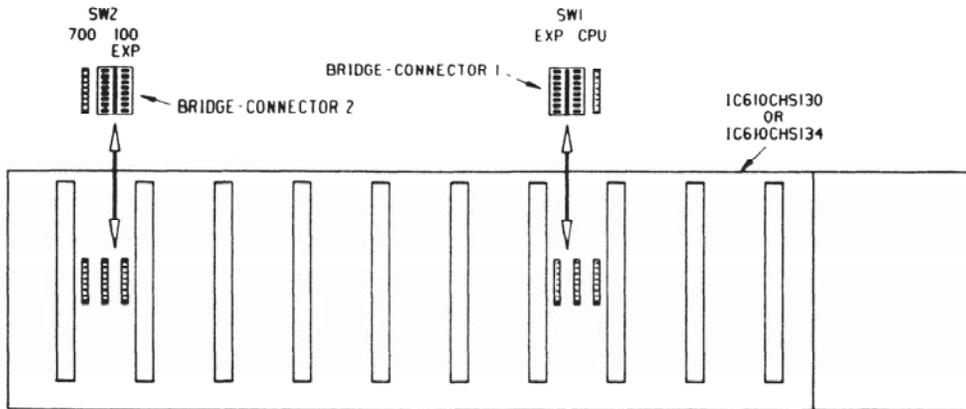
14. Inside each 5-slot rack at the rear of the leftmost module slot is a two position switch. On the CPU unit, this switch must be positioned towards the left. On the first expander it must be towards the right. If a third rack is used, the switch must be towards the left again. An exception to the described switch settings is when a 5-slot rack is used as an expansion rack in a Series One Plus system, where the CPU rack is a 10-slot rack. In this case, the switch must be positioned to the left. This switch is not included in the IC610CHS101 rack, since it cannot be used as an expansion rack.

**CAUTION**

**All switches must be set properly. The Series One or Series One Plus PC may not function correctly if any switch is not in the correct position.**

15. The 10-slot racks, IC610CHS130/134 have two bridge connectors on the back plane which must be configured. Bridge connector SW1, located between slots 3 and 4, has 2 positions EXP and CPU. The jumper must be positioned on the corresponding pins to specify whether the rack is a CPU rack or an Expansion rack. Bridge connector SW2, located between slots 9 and 10, selects the address to be assigned to slot 10 and the expansion rack slots. The selections are either 100 EXP or 700. Figure 3-5 shows the location of SW1 and SW2.

a40794



**Figure 3-5. I/O Addressing Switches in Series One Plus 10 Slot Racks**

16. Example of SW1 and SW2 settings are shown below in figure 3-6.

a40795

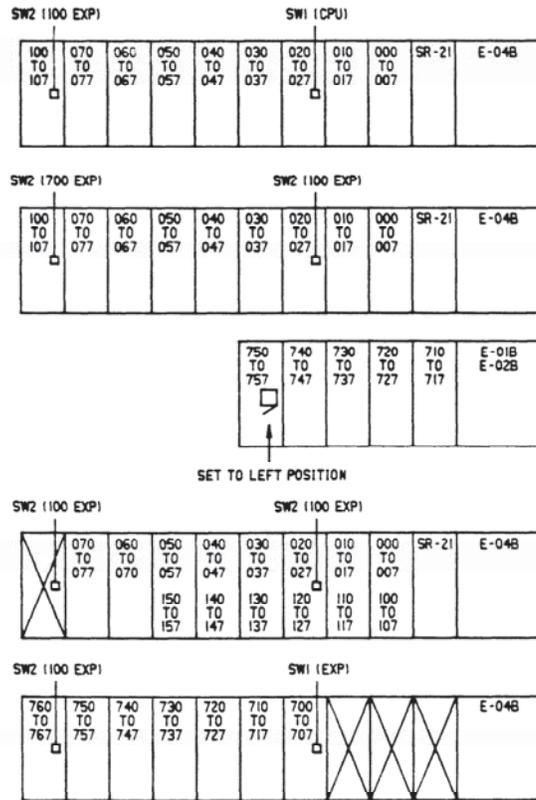


Figure 3-6. Examples of Rack Configuration Switch Setting Series One Plus

pc-s1-83-0027

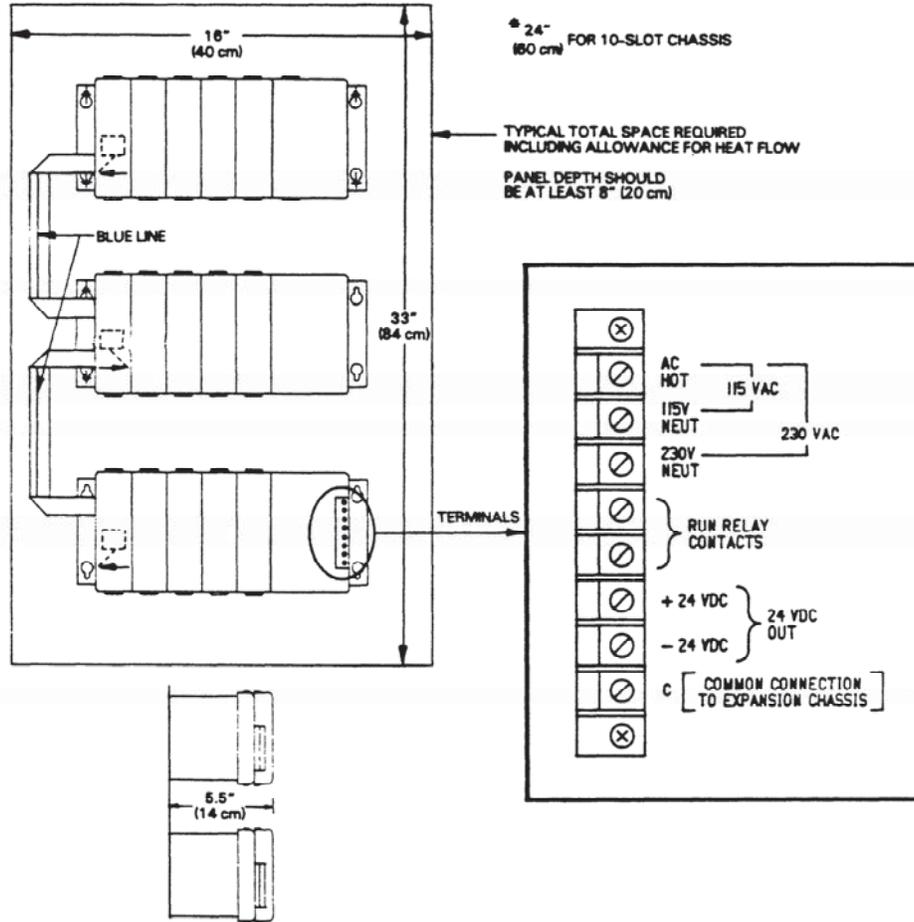


Figure 3-7. Typical Installation Dimensions

NOTE

Ground Connection should be made to mounting bracket, not to the terminal strip.

CPU

17. Locate the CPU module (see figure 3-8) and set switches and shorting bridge connectors per table 3-2. If additional CMOS or PROM memory is to be installed, now is the time to do it. For detailed instructions, see Chapter 7.
18. The CPU module must be installed adjacent to the power supply in the first or topmost rack.

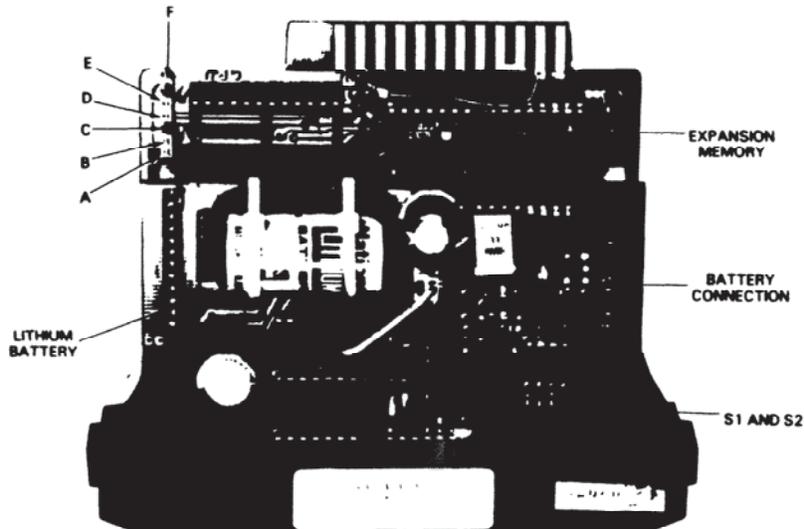


Figure 3-8. CPU Module

Table 3-2. CPU Option Settings

SWITCH 1 - POWER UP FUNCTIONS		SWITCH 2 - MEMORY TYPE	
OFF	Clear Coils	OFF	PROM
ON	Retain Coils	ON	CMOS

NOTE

ON is Towards the Faceplate. Factory setting is: Clear Coils and Select CMOS memory. Retain Coils affects 340-373 only. Counters and Shift Registers are always retentive.

Table 3-3. Memory Size

MEMORY SIZE	CONNECT JUMPERS BETWEEN PINS
700 Words CMOS	$\overline{A} \ \overline{B}$ and $\overline{D} \ \overline{E}$
1724 Words CMOS	$\overline{B} \ \overline{C}$ and $\overline{E} \ \overline{F}$
1724 Words PROM	$A \ B$ and $D \ E$

CAUTION

Remove jumper  $\overline{E} \ \overline{F}$  before installing PROM. Battery will discharge in a very short time if  $\overline{E} \ \overline{F}$  is not disconnected.

NOTE

Pin A is towards the faceplate. Factory setting is: 700 words CMOS, 1724 words PROM (same Setting)

GEK-90842

a40282



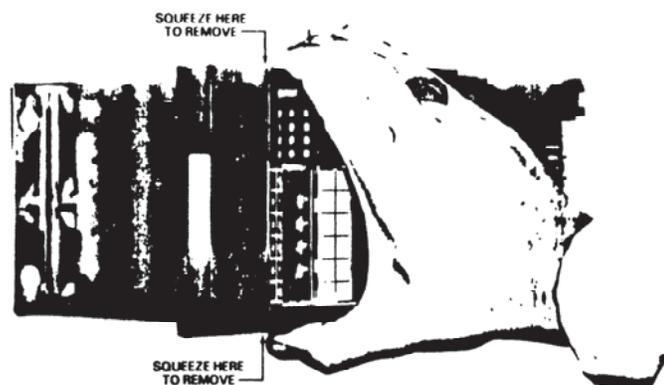
**Figure 3-9. Proper Module Insertion**

19. Tilt the module approximately  $10^\circ$  as shown in figure 3-9. Insert bottom of the large printed circuit board into the bottom card slot.
20. When the bottom slot is engaged, rotate the module to engage top slot. Slide module into base unit until it is firmly seated and snap locks engage.

### I/O

21. Other modules are installed into racks at slot locations as determined by installation plans. Follow steps 19 and 20 above for each I/O module.
22. To remove a module, squeeze the snap locks top and bottom (see figure 3-10) towards the center and pull the module straight out. Squeeze force should be approximately 10 pounds (5 Kg) and pull force 8-12 pounds (4-5 Kg).

a40283



**Figure 3-10. Proper Module Removal**

**Power Connections**

23. AC or DC power connections are made to the terminal strip on the far right of each unit. See table 3-1 for power requirements. The minimum recommended wire size for power connections is AWG No. 18 (1mm).
24. Strip  $0.4 \pm 0.1$  inches ( $10 \pm 2$ mm) of insulation from each wire (hot, neutral, and ground or + and -) or place a No. 6 insulated lug (ringed or forked) onto each wire.
25. Remove plastic cover over terminal strip and retain. Connect the hot wire to the top terminal. The neutral is connected to either the next (second) terminal for 115 V ac operation or the third terminal for 230 V ac operation as marked on the base unit. For 24 V dc racks, connect the + and - terminals to the DC power source.

**WARNING**

**Do not use the top screw for power connections; it secures the terminal block. Verify connections before applying power.**

26. Connect an earth ground wire to the rack mounting bracket as shown in figure 3-2. (Typically this ground wire is the green wire from the ac power source.)

**WARNING**

**Ensure that all exposed wiring is either under the screw-down plate of the terminals or insulated by shrink tubing or sleeves.**

27. The terminals labeled +24 V dc and -24 V dc provide 24 V dc @ 100 mA for connection to an external sensor (not on IC610CHS101).
28. The two remaining terminals are used with the RUN indicator. Use of this standard feature on all racks is optional. It can be used to drive an external indication of the functional state of this rack. The Run relay is closed when the CPU is scanning (not on IC610CHS101).
29. If the Run indication is desired, it can be wired separately to an external indicator (light, bell, whistle, etc.) or in series with other racks. Follow steps 24-25 above for guidance on connecting these wires. Then replace the plastic cover.

**I/O Field Wiring**

30. Recommended wire size for connection to the I/O modules is stranded AWG No. 12 (0.65-2mm) wire. Two wires per terminal are possible with AWG No. 14 (0.65-1.6mm) wire.
31. Strip  $0.3 \pm 0.05$  inches ( $8 \pm 1.5$ mm) from each wire to be connected to the I/O modules or install a No. 6 insulated lug (ringed or forked). Bare wire connections are recommended for multiple wire connections to one terminal.
32. Carefully remove plastic covers over I/O terminal connections by lifting top or bottom leg and sliding it to right or left.
33. Starting with the lower terminals, connect the field wires to all I/O terminals. Power connections such as those to commons (C) should be made last. No connections are required to unused circuits; however, screws on unused terminals should be tightened.

GEK-90842

34. For 16 point I/O modules that connect to I/O through a connector on the faceplate, secure the connector on the I/O Interface cable to the connector on the module.

**WARNING**

**Ensure that wires to the top set of terminals do not extend beyond the screw-down plate. Any wire exposed before the terminal plate must be covered by shrink tubing or sleeves.**

35. Wires should be laced together to leave a service loop adequate for removal of I/O modules without disconnecting wires. See figure 3-11. After testing of I/O wiring, replace plastic covers.

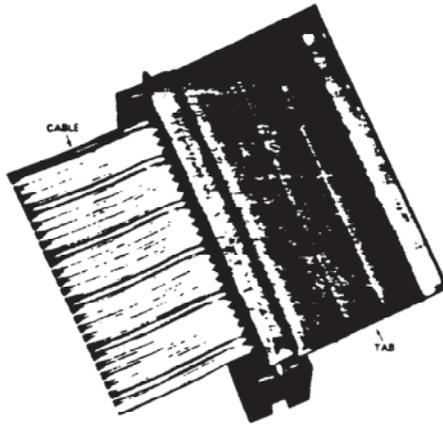
a40017



**Figure 3-11. Typical Routing of I/O Wiring**

### **Programmer**

36. The hand-held programmer (catalog no. IC610PRG100 or IC610PRG105) can be placed over the CPU and power supply for permanent or temporary mounting. Its connector fits into the receptacle on the CPU and it snap locks onto the power supply.
37. If an extender cable is used with the programmer, it should be installed next or when required. One end of the cable has a push tab (see figure 3-12). This end is connected to the programmer (see figure 3-13). The other end is connected to the 26-pin connector on the CPU. Both ends are keyed for proper installation. The red edge of the ribbon cable is installed up at both ends.



**Figure 3-12. Pull Tab on Programmer Cable**

**NOTE**

For proper noise immunity, it is recommended that the extender cable be used on a temporary basis and not permanently installed with the programmer.

38. To install the programmer directly onto the CPU, align the programmer on the outside dimensions of the power supply and gently push down to engage snap locks.

**CAUTION**

To ensure proper CPU operation, it is recommended that the programmer not be connected nor disconnected with ac power applied.

83-pc-35mm-4-3-31a



**Figure 3-13. Installation of Programmer Cable**

GEK-90842

39. The Programmer Mount Assembly (catalog no. IC610PRG190) can be used when installing the hand-held programmer on the outside of a panel or console, or can be used as a table top stand for the programmer. Figure 3-14 shows how to install the mounting assembly. *The CPU/Programmer cable must be ordered separately.*

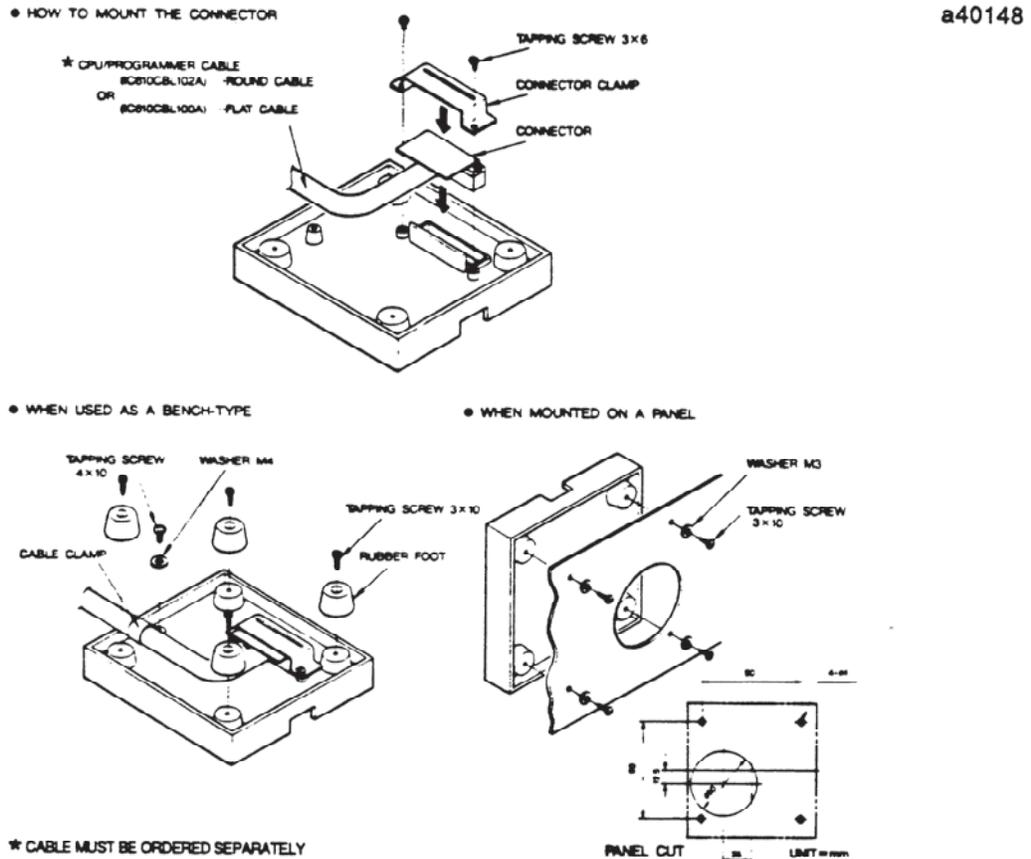


Figure 3-14. Programmer Mount Assembly

### Power Supply Limitations for Racks

If the power supply in either a CPU or expansion rack should become overloaded, unpredictable system operation may occur. To ensure that this does not happen, the total current capabilities of the modules placed in the rack must not exceed the current carrying capabilities of the rack power supply.

### Units of Load

The power used by each module is expressed in (units of load), where 1 unit equals 10 mA. Calculations are based on the worst case condition with all inputs and outputs on. Table 3-4 list the units of load supplied by each rack, and table 3-5 is a list of units of load used by each module. When configuring a rack, note the units of load supplied, then add the total units of load used by the modules you have selected. The total units of load for the modules must not exceed the total units of load supplied by the rack. If they do, the system should be redesigned.

Table 3-4. Units of Load Supplied by Rack

CATALOG NUMBER	RACK DESCRIPTION	POWER SUPPLIED IN UNITS OF LOAD			
		+5 V	+9 V	+24 V	+24 V External
IC610CHS101	5-slot, 115/230 V ac	40	60	20	-
IC610CHS110	5-slot, 115/230 V ac	140	80	40*	10
IC610CHS111	5-slot, 115 V ac, UL	140	80	50	10
IC610CHS114	5-slot w/24 V dc P/S	140	80	40	-
IC610CHS130	10-slot, 115/230 V ac	140	170	50*	10
IC610CHS134	10-slot w/24 V dc P/S	140	170	50	-

\*If an external sensor is connected to the 24 V + and - terminals on the power supply, the current used by the sensor (up to the maximum of 100 mA), should be deducted from the available listed units of load.

Table 3-5. Units of Load Used by Modules

CATALOG NUMBER	MODULE DESCRIPTION	POWER USED IN UNITS OF LOAD		
		+5 V	+9 V	+24 V
IC610CPU101	CPU	25	-	-
IC610CPU104	CPU25	-	-	-
IC610CPU105	CPU	25	-	-
IC610CPU106	CPU	25	-	-
IC610PRG100	Programmer	6	5	-
IC610PRG105	Programmer	6	5	-
IC610MDL101	24 V dc Sink Input (8)	-	1	10
IC610MDL103	24 V dc In/Out (4/4)	-	2	7
IC610MDL104	24 V dc In/Relay Out (4/4)	-	20	6
IC610MDL105	Thumbwheel Interface	-	1	9
IC610MDL106	24 V dc Sink In w/LEDs (16)	-	3	24
IC610MDL107	24 V dc Sink Load In (16)	-	3	23
IC610MDL110	High Speed Counter	-	7	-
IC610MDL111	24 V dc ac/dc Input (8)	-	1	-
IC610MDL112	24 V ac/dc Source In (16)	-	13	-
IC610MDL115	Fast Response I/O (4/2)	-	8	6
IC610MDL124	I/O Simulator (8) Inputs	-	1	11
IC610MDL125	115 V ac Input (8)	-	1	1
IC610MDL126	115 V ac Isolated Input (4)	-	1	-
IC610MDL127	230 V ac Input (8)	-	1	-
IC610MDL135	UL, 115 V ac Input (6)	-	1	-
IC610MDL151	24 V dc Sink Output (8)	-	2	3
IC610MDL153	24 V dc 2A Sink Out (4)	-	1	1
IC610MDL154	24 V dc Sink/Source Out (4)	-	1	10
IC610MDL155	24 V dc Source Output (8)	-	3	-
IC610MDL156	24 V dc Sink Out w/LEDs (16)	-	4	0
IC610MDL157	24 V dc Sink Out w/LEDs (16)	-	4	10

GEK-90842

Table 3-5. Units of Load Used by Modules - Continued

CATALOG NUMBER	MODULE DESCRIPTION	POWER USED IN UNITS OF LOAD		
		+5 V	+9 V	+24 V
IC610MDL158	24 V dc Source Out w/LEDs (16)	-	20	-
IC610MDL175	115/230 V ac Output (8)	-	16	-
IC610MDL176	115/230 V ac Isolated OUT (4)	-	16	-
IC610MDL180	Relay Output (8)	-	34	-
IC610MDL181	UL, Relay Output (5)	-	23	-
IC610MDL182	Relay Output (16)	-	48	-
IC610MDL185	UL, 115 V ac Output (6)	-	12	-
IC610CCM105	Data Communications Unit	30	-	-
IC610CCM110	I/O Link Local	60	-	-
IC610CCM111	I/O Link Remote	60	-	-
IC610PER151	Printer Interface Unit	26	-	-
IC610PER154	PROM Writer Unit	50	-	-
IC609CTU100	Timer/Counter Setpoint Unit	4	1	-

1 unit of load = 10 mA. Calculations are based on the worst case, that is, all inputs and outputs on.

## Safety Considerations

When planning the layout of a system, safety should be a prime consideration. System planning should include procedures and methods to ensure the physical safety of personnel, the Series One or One Plus system and the equipment or process being controlled. Those personnel who are involved in the planning and installation of a system should be familiar with all local and national electrical codes as well as installation instructions in this manual.

All practices should be followed that are specified by the IEEE (Institute of Electrical and Electronic Engineers) Standard 510 which includes tray and conduit spacing and wiring procedures. A copy of this standard can be obtained by writing:

*Institute of Electrical and Electronic Engineers  
345 East 47th Street  
New York, NY 10017*

## Recommended Field Wiring Procedures

The following procedures are recommended when running field wiring:

- Low-level signal wires should be separated from other field wiring.
- AC power wiring should be separated from DC field wiring.
- Wiring should not be routed near devices causing electrical interference.
- If severe noise problems are present, additional power supply filtering or an isolation transformer may be required. Contact your GE Fanuc Automation sales representative if assistance is required.
- Proper grounding should be provided to minimize hazards to personnel.
- Label all I/O wires. Circuit numbers or other identification can also be marked on the cover over the wire terminals on each I/O module.
- I/O wires should be no larger than No. 12 AWG.



### Introduction to PC Operation

After completing the installation procedures described in the previous chapter, your Series One or Series One Plus PC is ready for operation. This chapter describes the operation sequences for those PCs. A description of the features and functions of the programmer are the basis for the information in this chapter. The last portion of this chapter describes the operation of peripheral devices that can be used with the Series One or Series One Plus. The information in this chapter will allow you to become familiar with the programmer keys and the keystroke sequences required for the various programmer operations.

### Programmer

The hand-held programmer (figure 4-1), when connected to a PC by one of the three methods described in the previous chapter, can be used for entering a new program, examining a previously entered program, editing (changing) a previously entered program if required, monitoring the status of input or output points, displaying the contents of registers, and displaying timer or counter accumulated values. In addition, with the Series One Plus, a password can be entered with the programmer to protect the contents of your program from unauthorized or inadvertent entry or changing. Its features and functions are described in the following paragraphs. For detailed descriptions of the programming functions, refer to programming, Chapter 5.

a41927

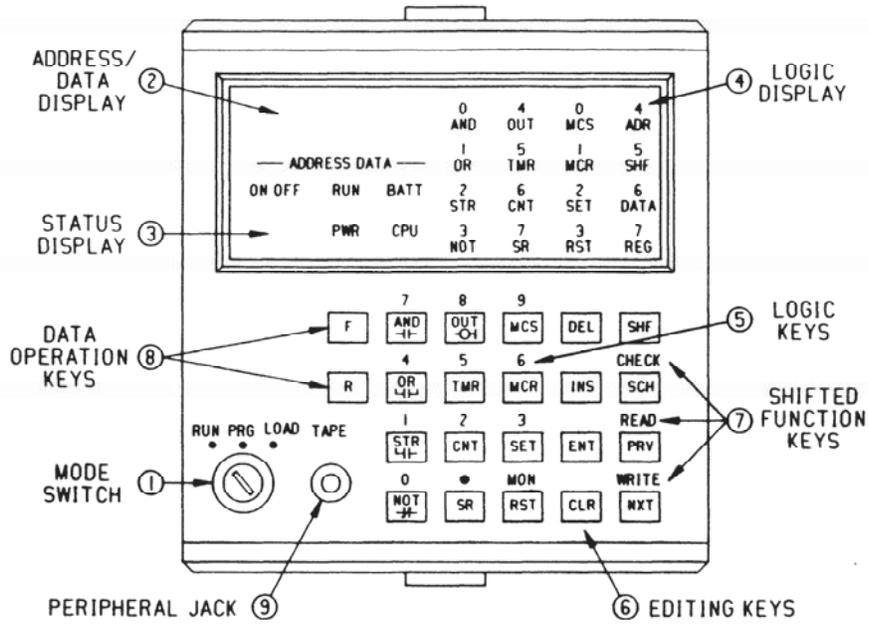


Figure 4-1. Programmer Features

## Programmer Features

The programmer features shown in the illustration on the previous page are described in this section. The descriptions of the features are keyed to the numbers (1 to 8) next to each feature in the illustration.

### 1. Mode Switch

This is a three-position keyswitch used for selecting the operating mode of the PC. This switch can be repositioned at any time as necessary without disrupting AC power. The left position (RUN) allows program execution with outputs enabled. The CPU scans its stored logic and allows timer/counter and relay contacts to be displayed. However, in the Run mode, changes to the logic are not allowed. In the center (PRG or Program) position, new programs can be entered and previously entered logic can be altered; however, no solving of the logic is performed. The right (LOAD) position connects the programmer to an external device such as a tape recorder through the adjacent tape port. Logic is not solved while in the LOAD position.

**WARNING**

**If a version A or B (IC610CPU101A OR IC610CPU101B) CPU IS installed, the Series One programmable controller will always power up in the run (operating) mode unless a programmer is connected and the programmer is in the program or load mode.**

### 2. Address Data Display

This is a four digit display used to indicate either the address, in decimal format, of where the display is in the user program stored in the PC's memory or reference data used as a part of the logic. To indicate that addresses are being displayed, periods appear near the bottom and to the right of each digit (for example, 0.1.2.3.). The LED to the top right behind ADR will also be lit.

### 3. Status Display

These five LEDs are energized to indicate the following functions or status of the Series One:

<b>ON/OFF</b>	When in the Run mode, this LED indicates the status of discrete references (I/O, internal coils, and shift register stages). It is ON when a reference is energized and OFF when de-energized.
<b>RUN</b>	ON when in the RUN mode and CPU is solving logic.
<b>BATT</b>	ON when the voltage of the internal lithium battery, which is used for maintaining the program stored in CMOS memory during no-power conditions, is at a low level and should be replaced within 10 days - see Chapter 7. This LED is OFF when battery voltage is OK, or battery is disconnected.
<b>PWR</b>	ON when the internal power supply is producing DC power. If OFF the rack power supply should be checked - see Chapter 7.
<b>CPU</b>	ON when internal error checking has detected a fault in internal hardware - see Chapter 7.

The above four LEDs (RUN, BATT, PWR and CPU) duplicate the operation of the indicators on the CPU when the programmer is installed.

### 4. Logic Display

These sixteen LEDs are used to indicate the type of logic entered into memory. While being programmed, they reflect the logic selected by the user prior to actual entry into CPU memory. For definition of the first 12 LEDs functions, see key definitions under 5 below. The 4 LED's to the right have special functions as follows:

<b>ADR</b>	ON when display is indicating an address value. The address is displayed in decimal notation, beginning at 0000 and ending at the last address in the user program or a maximum of 1723 (maximum number of 16-bit words in memory is 1724).
------------	---

GEK-90842

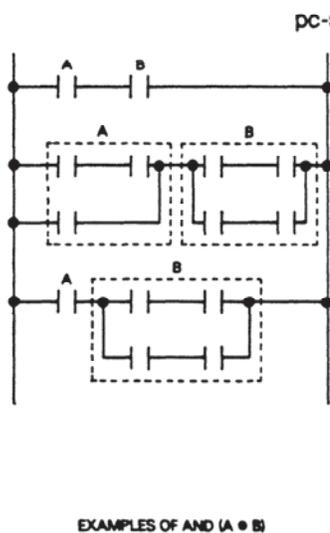
- SHF** ON when operator has selected the Shift key and is an indication that future key selections will be based upon the upper key labels. Shift function will remain in effect until either the Enter or Clear key is selected.
- DATA** (Series One Plus only) On when monitoring the contents of a register and the Address/Data display contains the data (4-digit BCD value) in that register. The contents of 2 consecutive registers will be displayed.
- REG** (Series One Plus only) On when monitoring the contents of a register and the address of the selected register is displayed. The display can be alternated between REG and DATA by depressing MON.

These sixteen LEDs can also indicate the status of 16 consecutive I/O states when used with the monitor function discussed later in this chapter. The numerals above the alphabetical characters are used during this function.

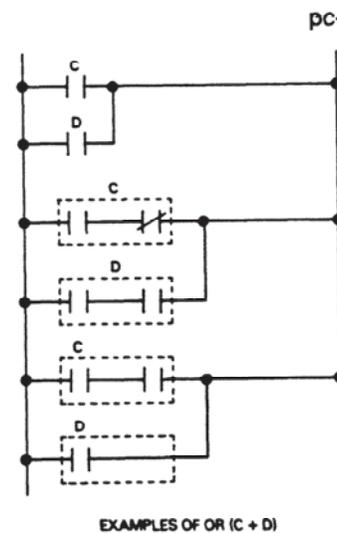
**5. Logic Keys**

These twelve keys are used to select the required function and enter logic when in the Program mode. The upper labels are used to enter numerical values when preceded by the Shift key. Numerical values, when entered, are viewed in the Address Data display. The decimal point is used only for the entry of timer preset values in tenths of a second. The Monitor function will be discussed below as part of the Upper Case keys. The function of the lower or normal case labels on these keys is described in the following key discussion.

- AND** Places logic such as two contacts or two groups of contacts in series (see figure 4-2). Power flow must be possible through both elements before it will be passed to the next element.
- OR** Places logic such as two contacts or two groups of contacts in parallel (see figure 4-3). Power flow can be possible through either (or both) elements before it will be passed to the next element.



**Figure 4-2. Series Logic (AND)**



**Figure 4-3. Parallel Logic (OR)**

- STR** Begins (Starts) a new group of logic and stores current logic results into a Last In, First Out (LIFO) pushdown stack.
- NOT** Inverts the state of the referenced status to create a normally closed contact. NOT AND results in a normally closed series contact. NOT OR results in a normally closed parallel contact.
- OUT** Specifies the end of a string of logic by entering a coil. With a valid reference, such as OUT 19, this coil will reflect the output of the rung of relay logic by turning ON or OFF as dictated by the power flow. Outputs can be paralleled by entering additional OUT functions at the end of a rung of logic.

<b>TMR</b>	Specifies a timer function to end a rung of logic. The timer requires a unique 6XX reference for storage. Two types can be programmed (seconds and tenths of seconds) depending upon the numerical preset (with or without decimal point) entered after the 6XX reference. For example, 0.9 or 9.0 seconds can be programmed, depending on the decimal point.
<b>CNT</b>	Specifies a counter function to end two logic rungs. The two logic rungs required by this function are count and reset. Count, being the top rung of conditional logic. When all programmed conditions permit power flow, the counter will begin counting. The bottom rung is the reset rung and is programmed with the conditional logic required for resetting the counter. The counter also requires a unique 6XX reference for storage. Transition detection is a standard built-in function.
<b>SR</b>	Specifies a Shift Register function to end three logic rungs (input, clock, and reset). The reference used with SR (for example, SR 410) is the first stage of this shift register. The last stage must be entered immediately after the SR and its reference. A shift register can contain a total of 128 steps or several shift registers can be programmed, each with a different number of steps, as long as the total of 128 steps is not exceeded.
<b>MCS</b>	The MCS (Master Control Start) key specifies the beginning of a Master Control relay function. This provides an efficient method of programming for controlling large quantities of coils if a specific permissive condition is not satisfied.
<b>MCR</b>	The MCR (Master Control Reset) key specifies the end of a Master Control relay function. A similar number of MCRs must be entered to terminate, one at a time, the Master Control Start functions entered. One MCR terminates only one previous MCS function.
<b>SET</b>	Used with latches, shift registers, and coils. It specifies where latches are to be turned ON (set), shift register stages set to the ON state, or coils to be turned ON and not affected by internal reference 376 (disable all outputs).
<b>RST</b>	Performs functions similar to the SET key for latches, shift registers, and coils, except it specifies when these references will be turned OFF (reset).

## 6. Editing Keys

These eight keys select the action required to modify either the stored logic within the CPU or the logic address being displayed. Those keys that can modify previously entered logic (that is, Delete and Insert) require a confirming key operation to ensure that the operation is to be executed. Thus, if they are accidentally depressed, the error can be corrected by depressing the CLR (Clear) key without actually affecting any entered logic. A description of each editing key is as follows:

<b>DEL</b>	When logic is being displayed, this key (Delete) when included as the first keystroke of a two key sequence, will cause that single function to be removed (deleted) from the CPU memory. To be effective, after depressing Delete, the PRV (Previous) key must be depressed, which executes the delete operation.
<b>INS</b>	This key (Insert) allows logic functions to be inserted between existing logic functions. The function or its address that is after the location at which the new function is to be placed, is displayed. Then the new logic is built followed by the Insert key (not the Enter key) and the confirming NXT (Next) key. The new logic that is inserted by this key sequence will be placed in memory immediately before the displayed function or address.
<b>ENT</b>	The ENT (Enter) key is used to complete the entry of logic when initially building the CPU program or to replace an entire single word function. Logic functions are entered typically at the end of the existing logic.

GEK-90842

- CLR** The CLR (Clear) key, when selected, clears the programmer of previously entered commands. If error codes are displayed, this key will acknowledge the error and return the programmer to its normal (cleared) function. When monitoring a program, depressing Clear will cause the display to indicate the memory address in lieu of logic functions. When the key sequence CLR, SHF, 3, 4, 8, DEL, NXT is entered, the entire contents of memory will be cleared.

#### NOTE

This key sequence should be used with caution, be sure that you want to clear the entire program. Remember, if single functions or groups of functions are to be cleared, the DEL key, followed by PRV should be selected for each function to be cleared.

- SHF** This key (Shift) locks all other keys to their shifted (upper label) functions and causes the SHF LED to be lit in the display area. The shift operation is ended by selecting the keys Clear or Enter. The SHF key must precede a numerical key or a group of numerals before selecting the numerical keys. For example to enter an open contact with input reference 12 as the first contact in a rung of logic, the key sequence would be STR, SHF, 1, 2, ENT.
- SCH** This key (Search) allows the entire program to be searched for specific logic functions. The logic function is defined by the logic keys (without Enter), then Search is selected. The memory will be searched from the current location until either a match is found or all memory is searched. Successive depressions of this key will cause repeated searches with memory wrap-around. An unsuccessful search results in error code E99 being displayed.
- PRV** When displaying logic or monitoring I/O state, selecting the PRV (Previous)B key causes the previous logic function or I/O status to be displayed. Additional depressions of this key cause the display to decrement until memory address zero is reached.
- NXT** When displaying logic or monitoring I/O state, selecting the NXT (Next) key causes the next logic function or I/O status to be displayed. Additional depressions of this key cause the display to increment until the end of memory is reached.

## 7. Shifted Functions

The Shift key, when selected, causes most keys to change their function to those marked on the face of the programmer directly above the corresponding keys. Normal unshifted functions are as placed on the keys themselves. Most of the shifted functions are used when entering numerical values (digits 0-9 plus the decimal point). When entering a program, the unshifted functions must be selected first (AND, OR, NOT, OUT, TMR, etc.), then the SHF is selected to enter the numerical portion of the program. The use of the other four shifted functions are as follows:

- MON** When in the Run mode, this function allows the user to monitor the status of I/O references in two successive groups of 8. The specified I/O reference is used to select the first 8 references and the next group in numerical sequence is also shown to provide a total of 16 real time statuses. The specified reference should be the first reference in a group of 8, for example, specify 10 to monitor the I/O group 10 to 17. If a reference other than the first one in a group is specified, monitoring of the I/O will default to the first reference in the group. After the reference (for example 043) is entered and displayed (for example 040) as data, the status of the 8 references in that group (for example 040-047) are displayed by the first 8 LED's (AND, OR, STR, etc.) in the logic display.

The next 8 references (e.g. 050) are also displayed and their status is indicated by the last 8 LED's (MCS, MCR, SET, etc.). These 16 LED's will be ON or OFF as the I/O assigned to these references is energized or de-energized. As the I/O changes, the state of the LEDs changes.

To display another group of I/O references, the PRV or NXT key can be selected to decrement (40, 30, 20, etc.) or increment (50, 60, 70, etc.) the group number and its associated references. References up to 577 can be displayed; there is automatic wrap-around from 570 to 000 (NXT) or 000 to 570 (PRV). The current values of timers and counters can also be monitored by entering TMR or CNT, SHF, 6XX (timer or counter reference) followed by MON (the Monitor Key).

- WRITE** This function operates with an audio tape cassette recorder or certain other peripherals to effect a transfer of the CPU's logic to the peripheral. Connect the device to the tape port with the audio cable supplied with each programmer or peripheral. Set the mode switch to LOAD and turn the device ON. To begin the transfer, select the WRITE key. This starts the writing of the CPU logic onto the cassette or other peripheral. For more detailed information on transferring data to peripheral devices, refer to the discussion of each peripheral.
- READ** This function operates with the audio tape cassette recorder or other peripheral to load a CPU's memory from the cassette or other storage device. Connect the device to the tape port with the audio cable supplied with each programmer or peripheral. Set the mode switch to LOAD and select the READ key. To begin the CPU loading, turn the device ON. This starts the reading of the tape and the loading of the stored program into the CPU memory.
- CHECK** This function operates with the audio tape recorder or other peripheral to verify proper program transfer. After a transfer is made, it should be read back similar to the READ above; however, the CHECK key must be selected instead of READ. The Check operation does not alter either the CPU logic nor the data in the peripheral. A data compare is made between the two sources of information to ensure that there has been no error in the recording process.

## 8. Data Operation Keys

These two keys are used during entering the data operations when programming the Series One Plus PC. These keys have no effect when this programmer (Catalog no. IC610PRG105) is used to program a Series One PC. The function of each of these keys is described below.

- F** This key must be depressed before a 2-digit value corresponding to the data operation or Function to be performed (for example, F50, F62, F74, F82, etc.). A complete list of the data operations and their Function number can be found in Chapter 5, Programming. The shift key does not need to be depressed immediately before the function numbers as must be done when selecting other numerical values when programming a Series One or Series One Plus PC.

- R** This key serves a dual function, in that it is used to select either a data register or a Timer/Counter accumulated value register, or a group reference number when programming data operations. A register is operated on as a word (16 bits) in the Series One Plus PC. When a 3-digit data register reference (400 to 577) is specified, two consecutive 8-bit registers are actually selected to write data to or read data from. For example, if R followed by the reference 410 is entered, registers 410 and 411 are selected for that data operation. If R 6 1 0 were entered, the accumulated value register for the timer or counter assigned to that reference would be specified in a data operation. Unused T/C references, 600 - 673 can be used as data registers.

A register can be specified to contain a Timer or Counter preset value. In this case the register reference (R XXX) is specified immediately following the Timer or Counter reference when programming a Timer or Counter.

A group reference is a 1, or 2 digit reference corresponding to the first 1 or 2 digits of a group of 8 consecutive discrete references. These references can be real I/O points, internal coils, or shift registers. For example, if the R key followed by 37 is entered, the reference is to discrete points 370 through 377. If R 5 is entered, the reference is to I/O points 050 through 067 (050-057 and 060-067).

GEK-90842

### 9. Peripheral Jack

This is a serial port which allows connection of an audio cassette tape recorder to record (save) the program stored in user memory, verify the contents of a tape against the contents of user memory, and load a program stored on tape into user memory. It is recommended that programs entered into user memory be saved on tape, once they have been verified as being correct. Note that when recording a Series One Plus program on tape, the contents of the data registers are not saved.

## Program Checking and Error Codes

When entering ladder logic programs with the programmer, the CPU automatically performs many checks on the data and operations selected by the programmer. Functions entered are checked for proper key sequence, proper range of references entered, etc. Errors detected during these checks are indicated in the data display by the letter E followed by a two digit code (01-99). The CPU also performs a partial program error check any time that the PC is switched to the RUN mode. After entering a program, a manual error check should be performed, which checks the entire program.

Table 4-1 summarizes the meaning of each error code, its cause, and possible methods of clearing the error. The use of the programmer to enter logic is documented in Chapter 5 as part of programming. However, there are many other valuable functions it provides which are shown in the following table:

**Table 4-1. Error Code Definitions**

Code	Applicable Mode			Significance	Cause	Corrective Action
	Run	Prog	Load			
E1	X	X	X	Incorrect Operation	Operator attempted to perform illegal operation such as changing program in RUN mode.	Examine operation. Depress CLR. Reinitiate proper function.
E2	X			Fault in Program Structure. Series One.	CPU has detected error in program when placed into RUN mode. Example: Input module reference used as coil	Go to Program mode. Depress CLR. Address of faulty logic will be shown. Depress NXT to display content.
E2	X			Fault in Program Structure. Series One Plus.	CPU has detected error in program when placed into RUN mode. Example: Input module reference used as coil.	Go to Program mode. Depress CLR-SCH-CLR. Address of faulty logic will be shown. Depress NXT to display content.
E3	X			Stack Capacity Exceeded	More than eight status levels attempted to be stored in pushdown stack.	Go to Program mode. Depress CLR. Programmer will display location of first 9th STR error. Examine logic and reprogram as necessary.
E5	X			Duplicate Coil Reference	Coil (output, internal, timer, or counter) used as an OUT more than once.	Go to Program mode. Depress CLR. Programmer will display location of second coil of pair using same reference. Enter another coil reference.

Table 4-1. Error Code Definitions - Continued

Code	Applicable Mode			Significance	Cause	Corrective Action
	Run	Prog	Load			
E6	X			Incomplete Master Control	More MCR references than MCS in program.	Go to Program mode. Depress CLR. Programmer will display first unmatched MCR. Correct program by deleting MCR or adding MCS.
E7	X			Incomplete Counter or Shift Register.	All control lines not provided to one or more Counters and/or Shift Registers.	Go to Program mode. Depress CLR. Programmer will display errant function. Add required reset, clock or clear lines.
E8		X		Incorrect Operation.	Operator attempted to write instruction on second word of a 2-word instruction.	Depress CLR.
E9	X			Incomplete Logic.	Relay ladder line not connected to coil; relay contact(s) left incomplete or hanging.	Go to Program Mode. Depress CLR. Programmer will display first unfinished logic element. Add logic to tie this element into stored logic, or delete element(s) to remove incomplete logic.
E11		X		Memory Full.	Operator attempting to add logic to CPU already at limit.	Depress CLR. Restructure program so that logic limits will not be exceeded.
E21	X	X		Parity Failure.	CPU has detected a fault in the parity structure of its internal memory.	Go to Load Mode. Depress CLR. Reload memory from previously recorded tape or clear entire memory and reload manually. If BATT light not ON and fault cannot be cleared, replace CPU module.
E25			X	Faulty Comparison.	External device such as tape cassette has content that does not agree with CPU memory.	Depress CLR. Verify correct program number or tape. If correct, either re-record tape or reload CPU.
E28			X	Weak Record Signal.	Playback Signal level, such as from tape recorder, is below acceptable level.	Adjust volume level on tape recorder or other peripheral device. If ON steady for extended period of time, restart function to obtain reliable operation.
E99	X	X		Unsuccessful Search.	Search function has reviewed all memory and has not located required.	Depress CLR. To cause an additional search, re-enter function and restart.

## Operation Sequences

An understanding of the basic PC operation sequences is necessary in order to effectively and efficiently enter ladder diagram programs. You should be familiar with the use of each key, alone and in sequence

GEK-90842

with other keys. The programmer is an excellent tool for program entering, editing and monitoring. Table 4-2 lists the various operations, the keystrokes required to enter those operations, and the mode or modes in which the operation can be performed. Each of the modes is indicated by a letter; R (RUN), P (PROGRAM), or L (LOAD).

Table 4-2. PC Operation Sequences

Operation	Keystrokes or Operation Sequence	MODE		
		R	P	L
Insert or Remove Key		X		X
Clear all user memory	CLR SHF 3 4 8 DEL NXT		X	
Display present address	CLR	X	X	
Display present function	NXT	X	X	
Next function	NXT(After previous keystroke)	X	X	
Previous function	PRV	X	X	
Go to address 0000 in program memory	SHF NXT	X	X	
Go to specific address	SHF (address) NXT	X	X	
Go to next address	After previous entry NXT	X	X	
Clear displayed function	CLR	X	X	
Search for a specific function	(Function) SHF (Ref. No.) SCH NXT	X	X	
Search for a specific reference number	SHF (Ref. No.) SCH NXT	X	X	
Insert function before the displayed function (or address)	(Function) SHF (Ref. No.) INS NXT		X	
Cancel insert	CLRSHF INS		X	
Delete a function	(address) DEL PRV		X	
Check program for errors. If no error detected, the next available address is displayed.	CLR SCH	X	X	
Turn Timer/Counter ON	TMR or CNT SHF 6XX SET	X		
Turn Timer/Counter OFF	TMR or CNT SHF 6XX RST			
Change preset value of a Timer or Counter	SHF (preset value) ENT	X		
Monitor ON/OFF status of a group of 16 consecutive references (I/O, internal coils, Shift Register coils)	SHF (Beginning Ref. No.) MON	X		
Monitor Timer or Counter accumulated value	SHF (T/C No.) MON To monitor the previous or next T/C enter, PRV or NXT	X		
Monitor ON/OFF state of any coil or contact	The ON OFF LED turns ON or OFF to indicate the state of the displayed reference.	X		
Force a reference ON (will be overridden by user logic)	SET SHF (Ref. No.) ENT Reference can be Input, Output, Shift Register, Timer or Counter	X		

Table 4-2. PC Operation Sequences - Continued

Operation	Keystrokes or Operation Sequence	MODE		
		R	P	L
Force a reference OFF (will be overridden by user logic)	RST SHF (Ref. No.) ENT	X		
Enter a function into program memory	(Function) SHF (Ref. No.) ENT		X	
Edit a program	Same as above. Select address and enter change		X	
Transfer data to tape	(Optional program ID) SHF WRITE			X
Load program memory from tape	(Optional program ID) SHF READ			X
Verify data on tape or in PROM writer RAM against program memory	(Optional program ID) SHF CHECK			X
<b>SERIES ONE PLUS ONLY</b>				
Monitor contents of Data Registers (1)	R (Register Ref) SHF MON	X		
Monitor next register	NXT	X		
Monitor previous register	NXT	X		
Enter or change the contents of displayed register. A 4 digit BCD value can be entered (16-bits).	SHF XXXX ENT When the data contents are displayed, this sequence will overwrite the current contents of the register	X	X	
Enter or change a password (2) (3)	CLR SHF 9876 DEL SHF XXXX ENT NXT	X	X	
Log in sequence (3)	CLR SHF 5678 DEL SHF (Password) ENT NXT	X	X	
Log out sequence (3)	CLR SHF 1234 DEL NXT	X	X	

1. When the MON key is selected, the ADDRESS DATA display shows the address of the specified register. To display the data contents of the register, depress the MON key again.
2. 9876 is a fixed sequence. The password (XXXX) can be any numerical value between 0001 and 9999. The value 0000 is equivalent to no password. All programmer functions are operable when the password is 0000.
3. Password information also applies to the Series One Model E PC.

The above table provides a convenient reference to the programmer keystrokes required for the various PC operations. A more detailed description of each operation is provided on the following pages or in Chapter 5, Programming.

**Monitor CPU Logic** - This sequence of operation provides the steps required in order to observe the contents of user memory. You can either step forward or backward in user memory. With the Programmer installed and the mode switch in the Run or PRG (Program) position, observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Depress the NXT key, the logic content of memory location zero will be shown (beginning of memory or scan).

---

---

GEK-90842

3. Successive depressions of the NXT key will cause the programmer to step through the stored program in the order that the program is scanned, from address 0.0.0.0. to the last address in the program. At any time, the CLR key can be depressed to display the address of the logic then being viewed.
4. An additional depression of the NXT key will restore the display to the logic content.
5. At any time, depressing PRV will cause the logic of the previous (closer to zero) memory location to be displayed.
6. Successive depressions of PRV will cause the programmer to step backwards through the stored program in reverse of the order that the program is scanned. At location 0.0.0.0., PRV has no effect.
7. If by successive depressions of NXT, the end of the actual program is passed, the display will show End. With each successive selection (with the NXT or PRV keys) of an unprogrammed or empty memory word, the address of the new location will flash in the display for 1/4 to 1/3 of a second prior to going to End.

**Search CPU Logic** - In the event that you wish to quickly find a particular logic element (programming step) in user memory, this sequence of operation allows you to do so. After installing the programmer, with the mode switch in the RUN or PRG position, observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display, decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter the logic function whose location is to be searched for. For example, select AND, SHF, 1, 0, 5 (AND 105). Do not select ENT key.
3. Select the SCH key to begin the search. If the logic function is not found, error code E99 will be displayed.
4. If a match is found, the display will indicate the first memory address containing the desired function. Depressing NXT will cause the logic at that address to be displayed.
5. Successive depression of SCH, while the memory address is being displayed, will cause additional searches to occur from the current location.
6. As long as one match is found, the search will not stop at the end of memory, but will continue with memory address zero until a match (possibly the same location) is again detected.
7. To find the first empty memory word, depress CLR to obtain an address location. Then select SCH (search for zero content) to begin the search for the first available location.

**Alter One Logic Element** - This sequence of operation allows you to change the contents of a particular location in user memory. After installing the programmer, place the mode switch in the PRG position. Observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Move the display to the element to be altered using the NXT or PRV key or Search function.
3. Enter the new logic element, for example: OR, SHF, 2, and 5 (OR 25).
4. Select the Enter (ENT) key to cause the change to occur. The new logic element you have entered will take the place of the previous logic at that memory location. The next address will be automatically displayed.
5. If the new element is not to be entered or an error has been made in its construction, depressing the Clear (CLR) key will cancel the new logic and return the display to the address of the examined element.

**Delete One Logic Element** - This sequence of operation allows you to remove one logic element from the program in user memory. After installing the programmer, place the mode switch in the PRG position. Observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Move the display to the element to be deleted using the NXT or PRV key or the Search function.
3. Select the DEL key. Notice that the Address/Data display has a small "d" in the left digit of the display.
4. To confirm that the Delete operation is to be executed, select the PRV key. The next address will be displayed after the Delete is performed. The remaining user logic will automatically move back one address location (nearer to 0.0.0.0.) to fill the empty memory.
5. To cancel the Delete operation, select the CLR key (before selecting DEL). The display will revert to the element being considered for deletion.

**Clear All Memory** - This programming sequence should only be used when the entire contents of logic memory are to be cleared. After installing the programmer, place the mode switch in the PRG position. Observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Select the key sequence CLR, SHF, 3, 4, 8.
3. Select the DEL key; the display will change to.
4. To execute the clear function and cause all logic elements to be removed from user memory, depress the NXT key.
5. To cancel the clear function, the CLR key can be depressed before depressing NXT.

**Insert One Logic Element** - This sequence of operation allows you to insert one logic element between two existing program steps in the use program. After installing the programmer, place the mode switch in the Program position. Observe or perform the following actions:

1. The programmer will display address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Move the display to the element that will be after the location at which the new element is to be entered. This can be done by using the NXT or PRV key or the Search function.
3. Enter the new logic element, for example; AND, SHF, 1, 0, and 4 (AND 104).
4. Select the Insert (INS) key to cause the change to occur. Notice that the address display has a small "i" in the left digit of the display.
5. A confirming keystroke is required to ensure that the insert action by the operator is valid. To confirm the insert, select the NXT key. After the insert is performed, the display will show the address of the next logic element.

**Monitor I/O Status** - This operation sequence allows you to monitor the status (ON or OFF) of references. A total of 16 I/O references can be monitored at any one time, beginning with the lowest address in the group with the reference selected by the operator. Each reference is within a group of 8 references. The I/O status of the group containing the selected reference, plus the next higher group of 8, is displayed. After installing the programmer, place the mode switch in the RUN position. Observe or perform the following actions:

1. The programmer will display the address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter any reference to be monitored. For example, to monitor the real time status of references 020-027, or any one reference within that group, the following keys can be selected: SHF, 2 and 0.

GEK-90842

3. Select the Monitor function (MON). Note that the shift is still in effect (selected in previous step) causing the RST key to select the monitor function.
4. The display will revert to ]-] followed by the lowest reference (e.g. 020) in that group. The I/O references can be for a 4, 8, or 16 point module. If the module is a 4-point module, the status of the first 4 I/O points are real world outputs and their operating status will be displayed. The 4 higher references cannot be used as real world outputs, but can be assigned to internal coils. The 4 monitor LEDs normally assigned to the 4 higher references in this group will always be OFF.  
The status of an 8 point module (for example, 030 to 037) is indicated by the first 8 LED's (AND, OR, STR, .... SR) and the next module's status, if an 8 point (for example, 040-047) will be indicated by the last 8 LEDs (MCS, MCR, SHF....7). In the case of a 16-point module, the status of all 16 points will be consecutively displayed.
5. Depressing the NXT or PRV keys will cause the address display to increment or decrement to the next or previous group of 8 statuses or module, if in an expansion rack. All discrete references can be examined. The display returns to zero after the highest reference has been examined.

**Monitor Timer or Counter Status** - This operation sequence allows an operator to monitor the current accumulated value of a timer or counter. After installing the programmer, place the mode switch in the RUN position. Observe or perform the following actions:

1. The programmer will display the address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter the timer or counter reference to be monitored. For example, to monitor the operating status of timer 601, enter SHF, 6, 0, 1. If a counter had been assigned reference 601, you would then be monitoring the status of counter 601. The valid references for timers or counters are 600 to 677. References 674, 675, 676, and 677 are reserved for use with the Thumbwheel Interface unit (Series One or One Plus) or the Timer/Counter Setpoint Unit (only for Series One Plus). Remember, each valid reference can be assigned to only 1 timer or 1 counter.
3. Select the Monitor function (MON). Note that the shift selected in the previous step is still in effect causing the RST key to select the monitor function. If the selected reference is not used in the program, error code E01 will be displayed.
4. The display will contain the current content (accumulated value) of the selected timer or counter. Timer values will be displayed in tenths from 000.1 up to 999.9 and counter values will be 0001 to 9999. The Logic Display will also show the two least significant digits of the timer or counter reference (for example, 01 for timer or counter 601).
5. The NXT and PRV keys move the display to adjacent timer or counters. Any timer or counter not actually used in the logic cannot be displayed.

**Display a Specific Address** - This operation sequence allows you to select and display a specific memory address (location) and the logic content of that address. After installing the programmer, place the mode switch in the RUN or PRG position. Observe or perform the following actions:

1. The programmer will display the address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter the memory address to be observed. For example, if the logic contained in address 36 is to be observed, SHF, 3 and 6 would be selected.
3. The NXT key is then selected and the display will contain the selected address. To display the content of this memory address, the NXT key is depressed a second time.
4. At any time, selecting SHF then NXT will cause the display to go to address 0.0.0.0.

**Monitor Data Register Contents** (*Series One Plus only*) - This operation sequence allows an operator to monitor the current contents of any data register. After installing the programmer, place the mode switch in the RUN position and perform the following actions:

1. The programmer will display the address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter the reference for the data register to be monitored. Valid data register references are 400 through 577 and 600 through 673. For example, to monitor the contents of data register 452, enter the key sequence R 4 5 2 MON.
3. The display will contain the address of the selected data register.
4. To display the contents of the selected data register, depress the MON key a second time. The display will now contain the current contents (4-digit BCD value) of the selected register.
5. The NXT and PRV keys move the display to adjacent data registers.

**Change Contents of a Data Register** (*Series One Plus only*) - This operation sequence allows an operator to change the current contents of any data register. After installing the programmer, place the mode switch in the RUN position and perform the following actions:

1. The programmer will display the address zero indicated by four zero digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter the reference for the data register to be monitored, following the steps in the previous operation.
3. After depressing the MON key a second time and the current contents of the data register are displayed, the contents may be changed if desired.
4. To change the current contents of the displayed data register, enter the sequence SHF XXXX ENT, where XXXX is a 4-digit BCD value having a valid range from 0000 to 9999. The new value entered will write over (replace) the value originally displayed.

**Enter or Change a Password** (*Series One Model E and Series One Plus*) - This operation sequence allows an operator to enter a password or change a previously entered password. When a password is entered, all programmer functions are disabled, except for the monitor functions. This prevents unauthorized access to the program in a Series One Plus PC. After installing the programmer, place the mode switch in the RUN or PRG position and perform the following actions:

1. To enter a new password or change a previously entered password, use the following key sequence:  
CLR SHF 9876 DEL SHF XXXX ENT NXT
2. 9876 is a required sequence. Your password (XXXX) can be any numerical value from 0001 through 9999. The value 0000 is equivalent to no password, all programmer functions are operable when the password is 0000. If you do not have a requirement for password security, use 0000.

To have access to all programmer functions when a password is in effect, a LOG IN sequence must be entered by the operator. After performing any necessary operations with the programmer, the operator must then enter a LOG OUT sequence to prevent any unauthorized access to the program. The LOG IN and LOG OUT sequences are shown below.

**Access to Programmer Functions with Password in Effect** (*Series One Model E and Series One Plus*)  
After installing the programmer on a Series One Plus PC having a password in effect, place the mode switch in the RUN or PRG position and perform the following actions:

1. To LOG IN, enter the key sequence:  
CLR SHF 5678 DEL SHF XXXX ENT NXT
2. The numerical value, 5678 is a required entry. XXXX is the password that had been previously selected by the operator.

GEK-90842

3. The operator can now monitor or change the program as required. When not using the programmer, the operator should then LOG OUT to return the programmer to password security.
4. To LOG OUT, enter the following key sequence.

CLR SHF 1234 DEL NXT

5. Verify that the programmer functions have been disabled.

#### NOTE

If the user should for any reason forget the password entered into the PC, use of most programmer functions would not be possible. If this should happen, contact the GE Fanuc Automation service center in Charlottesville, Virginia for assistance.

**Forcing References** - The operation sequences described in this paragraph allow you to force an Input, Output, Shift Register, Timer, or Counter reference either on or off. Forcing the state of I/O references provides a convenient method of testing the operation of field devices and debugging the logic in the user program. If an input or output is forced on or off, it will remain in the forced state for 1 scan of the CPU. The programmer must be in the RUN mode to force any reference. After installing the programmer, place the mode switch in the RUN position. Use the following steps to force references.

1. The programmer will display address zero indicated by four digits in the display, with decimal points to the right of each digit (0.0.0.0.), and the ADR LED lit.
2. Enter one of the following sequences to force a reference either on or off as required:
  - To force a specific reference ON, enter the sequence: SET, SHF, XXX (Reference), ENT
  - To force a specific reference OFF, enter the sequence: RST, SHF, XXX (Reference), ENT
3. Repeat the above operation for other references to be forced.

**WARNING**

When forcing input points with the set or rst sequence, the actual state of the input may be overridden. If the forced set or rst occurs in the user logic program before the input is checked in the same i/o scan, the set or rst state will take precedence and could cause an output to be turned on or off at the wrong time.

## Operation With Peripheral Devices

Several peripheral devices are available for use with a Series One or Series One Plus PC. The Programmer is required for operation with the tape recorder, when recording user programs. The tape recorder connects to the Programmer through the Programmer's tape port. The tape port is located to the right of the mode switch and is labeled TAPE. The rest of this chapter describes the operation of these peripherals, which are listed below:

Audio Tape Recorder	Various models
Printer Interface Unit	Catalog Number IC610PER151
PROM Writer Unit	Catalog Number IC610PER154

### Tape Recorder Operation

Most audio tape recorders with auto-level control can be used with a Series One or Series One Plus PC. It is recommended that the recorder also be equipped with a counter to allow multiple programs to be recorded on tape. Units such as General Electric model 3-5148 have been tested and found fully compatible with the Series One family of PCs. This recorder is packaged with an interface module (not

required for the Series One or Series One Plus PCs) and can be ordered as Catalog Number 3-5156. The tape recorder allows three functions to be performed: (1) record a program onto tape, (2) load a CPU or peripheral from tape, and (3) verify the content of a tape. The operation of each of these functions is described below in a step-by-step manner. All logic memory is recorded on tape.

*Tapes created on Series One, Series One Model E or Series One Plus PCs can be loaded into any Series One, Series One Model E, Series One Plus or Series One Plus 3.7K PC. Tapes created on the Series One Plus 3.7K PC can only be loaded into another Series One Plus 3.7K PC.*

A 2.5 foot (0.75 meter) audio cable (Catalog number IC610CBL151), which is gray with a red tracer, is supplied with the programmer. This cable is used only with a tape recorder and connects it to the tape port on the programmer.

### Recording A Program

1. Install the Programmer and apply AC power to the PC.
2. Turn the mode switch on the Programmer to the LOAD position.
3. Apply AC power to the tape recorder. Verify presence of the write protect tab on a cassette and insert the cassette. If the write protect tab is not in place, data entered on the tape may be inadvertently erased.
4. Adjust the tone control to its highest position.
5. Connect the Programmer (TAPE port) to the tape recorder (MIC input) with the 2.5 ft. (0.75 m) audio cable (gray with red tracer).
6. Rewind tape to the beginning or to the desired record position if multiple programs are to be placed on one tape. Programs require approximately 1.5 minutes (700 words), 4 minutes (1724 words) or 8 minutes (3.7K words), as applicable, of tape per program.
7. For identification of a program, if desired, enter a four digit number (0000-9999) on the Programmer. When tape is accessed later to load the CPU, this number can be used to identify the correct program prior to altering CPU data. If a program number is not as expected, the operator can terminate the load operation and get the correct tape without loss of existing program nor delay incurred by loading a wrong program. However, this identification number is optional.
8. Begin the tape recorder operation by depressing the RECORD button (and PLAY if required by the tape recorder).
9. Depress the WRITE key on the Programmer. The record operation will now begin.
10. If after following all instructions, reliable operation cannot be obtained, try operating the recorder with batteries.
11. When the record is complete, the Programmer will display End in the Address/Data display and the ON/OFF LED will be off. Stop the recorder and note the counter position so that the amount of tape used for that program can be determined.
12. Depress the CLR (Clear) key on the Programmer to end the record operation.
13. It is recommended that the tape be rewound to where the record began and that the Verify operation described below be performed to ensure data integrity.

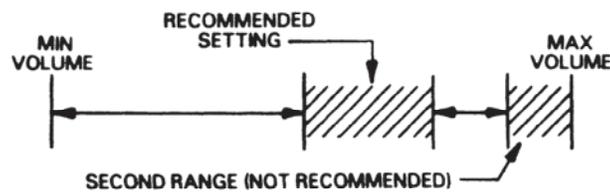
### Verifying A Program

1. Install the Programmer and apply AC power to the PC.
2. Turn the mode switch on the Programmer to the LOAD position.
3. Apply AC power to the tape recorder and insert the cassette containing the program to be verified.

GEK-90842

4. Adjust the tape recorder's volume control to the maximum setting. Adjust the tone control to its highest setting.
5. Connect the Programmer (TAPE port) to the tape recorder (EAR input) with the audio cable (gray with red tracer).
6. Rewind the tape to the beginning of the previously recorded program. The tape can also be on the blank area prior to the program, but not on another program. Enter the program identification number (if previously recorded).
7. Depress the CHECK key on the Programmer to select the verify operation.
8. Start the tape recorder by depressing the PLAY button. The verify operation now begins.
9. Any errors detected during the verify operation are indicated by an error code being displayed on the Programmer's Address/Data display. Error code E21 indicates that the tape has an internal parity error. E25 indicates a mismatch between the content of the tape and the CPU logic memory. E28 indicates that the play level is wrong and the verify operation should be stopped, volume adjusted, and the operation restarted (from step 6 above).
10. Setting of the volume control is critical for proper operation. The error code displayed for an incorrect (low) volume control setting is E28. Figure 4-4 illustrates the area of the volume control available for a proper setting.

pc-s1-83-0065



**Figure 4-4. Volume Control Range Setting**

With some recorders, there are two ranges where the signal level appears to be acceptable, one near the middle and one near maximum volume. The setting near maximum volume should not be used. The CPU will indicate that it is acceptable; however, an unacceptable amount of clipping distortion may occur in this area with some recorders.

11. Find the correct position for the mid-range of the acceptable volume control during the beginning or header portion of the tape. Some experimenting may be necessary. The duration of time for the header is about 12 seconds. It is important to choose the lower of the two ranges for the volume control setting if they both exist (see 10 above). For the recommended GE Fanuc Automation recorder this is at about 80% of the full maximum setting. Mark the proper setting with paint or some other method of identification.
12. If the volume control has been correctly set before the end of the header, the programmer display will be blank, and the lower right number 7 LED will turn on dimly. Before the end of the header the LED will turn off. A few seconds later, F will be displayed, indicating that the program has been found.
13. During the adjusting process in 11 above, the data from the tape may not be valid if the adjustment is not performed quickly enough. The tape loading should be repeated with the correct volume control setting. To clear the CPU to restart the loading, it is necessary to either power-down the CPU or remove the Programmer from the CPU, then reattach and depress the Clear key.

14. When the verify is complete without error, the **Programmer** will display End in the Address/Data display.
15. Stop the recorder and depress the CLR (Clear) key on the Programmer to end the **verify** operation.
16. The **verify** operation will require approximately the same time as the record operation.

### Loading A Program

1. Install the Programmer and apply AC power to the PC.
2. Turn the mode switch on the Programmer to the LOAD position.
3. Apply AC power to the tape recorder and insert the cassette containing the required program.
4. Adjust the volume control to the setting determined during the verify operation. Adjust the tone control to its highest setting.
5. Connect the Programmer (TAPE port) to the tape recorder (EAR input) with the audio cable (gray with red tracer).
6. Rewind the tape to the beginning of a previously recorded program. Tape can also be on the blank area prior to the program, but not on another program. Enter the program identification number (if previously recorded).
7. Select the READ key on the Programmer to establish the load operation.
8. Start the tape recorder by depressing the PLAY key. The load operation now **begins**.

**If the CPU** detects a program number different than the one **entered in** step 6, the programmer will beep and the display will show PA .

9. If the wrong program has been selected, the load operation can be aborted by stopping the recorder and powering-down the CPU or remove the programmer **from** the CPU, then reattach and depress the CLR key.
10. Any errors detected during the load operation are indicated by an error code being displayed on the Programmer's Address/Data display. Error code E21 indicates the tape has an internal parity error. A steady E28 indicates the play level is wrong and the load should be stopped, volume adjusted, and the operation restarted (step 6 above).
11. When the load is complete with no errors, the Programmer will display End in the Address/Data display and the ON/OFF LED will be off. Stop the recorder and depress the CLR (Clear) key on the Programmer to end the load operation.
12. The load operation will require approximately the same time as the record operation.

### Printer Interface Unit

The Printer Interface Unit (catalog number **IC610PER151**) provides an interface between a Series One or Series One Plus PC and a printer for the purpose of providing a convenient means of obtaining a hard-copy printout of the program residing in the PC's user memory. It can also be used with a Series One Junior PC. The format of the printout is switch selectable and can be either Boolean (mnemonic) or ladder diagram format. Many readily available, inexpensive printers can be used with the Printer Interface Unit.

A 6' (**2m**) Printer Interface cable and an external power supply cable are included with the Printer Interface Unit. Printer Interface Unit Specifications are listed below in table 4-3

**Table 4-3. Printer Interface Unit Specifications**

<p>Operating Temperature Storage Temperature Humidity (<b>non-condensing</b>) Required Operating Power</p> <p><b>Dimensions</b> <b>Weight</b> <b>Printer</b> Interface <b>Printing</b> Capacity (maximum)</p>	<p>0° to 60° C (32° to 140° F) -10° to 70° C (14° to 158° F) 5 to 95% +5 V dc + 5%, 300 mA (minimum) (Supplied either <b>internally</b> from the CPU or from an <b>external</b> power supply) 5.7" x 4.7" x 1.5" (145 x 120 x 38mm) 11.68 oz (330 g) <b>Centronics</b> (Parallel) <b>Ladder Diagram Listing</b> 13 contacts and 1 coil per line 16 lines per page <b>Boolean Listing</b> 200 steps per page (4 lines with 50 steps in each line). 700 steps maximum</p>
---	---

Table 4-4 lists the requirements that a printer must meet for use with the Printer Interface Unit.

**Table 4-4. Printer Requirements**

<ul style="list-style-type: none"> <li>• Must have a Centronics interface</li> <li>• Able to <b>generate ASCII character</b> code 7CH (Hexadecimal) as a '/' (slash).</li> <li>• Must respond to control codes: OE H (SO) - Expanded print ON OF H (SI) - Compressed print ON 12 H (DC2) - Compressed print OFF 14 H (DC4) - Expanded print OFF</li> <li>• Must be capable of printing 132 columns; however, if an 80 column printer can respond to the SI (OF H) control code to allow 132 or more characters per line, it can be used. This type of printer, when used with the Printer Interface Unit, will print compressed characters. Choice of normal (132 column) or compressed (80 column) print is switch selectable.</li> <li>• The following printers have been tested for operation and can be used with the Printer Interface Unit. <ul style="list-style-type: none"> <li>General Electric Personal Computer Printer, model 3-8100</li> <li>Epson model RP-100</li> <li>Hewlett Packard <b>Thinkjet</b>, model HP 2225C</li> <li>IBM Personal Computer Graphics Printer</li> <li>Seiko <b>model</b> GP-500</li> </ul> </li> </ul>
--

**Hardware Description**

The Printer Interface Unit is a compact unit that attaches to the Series One or Series One Plus PC. Electrical and physical connections to the PC are made through a 26-pin connector located on the back of the Printer Interface Unit. The Printer Interface Unit is attached to the PC by placing its 26-pin connector directly over the mating connector on the PC and gently pushing down on the unit until it is securely in place.

Connection from the Printer Interface Unit to the selected printer is made through the Printer Interface Cable, IC610CBL152 to a 24-pin connector on the front of the unit. Power to the unit can be supplied directly through the rear panel connector from the CPU power supply or from an external power source

capable of supplying **+5 V dc @ 300 mA**. The power source, either internal (**INT**) or external (**EXT**) is selected by a **2-position** switch located on the rear panel. Connection to an **external** power source is made through a **3-wire** cable supplied with the Printer Interface Unit.

The sequence for operation of the Printer Interface Unit is initiated by depressing pushbutton switches on the front panel. Two indicator lights on the **front** panel provide a visual status of the Printer Interface Unit operation. There are also 2 indicators that provide operating status of the PC. Figure 4-5 is an illustration of the Printer Interface Unit showing the features mentioned above.

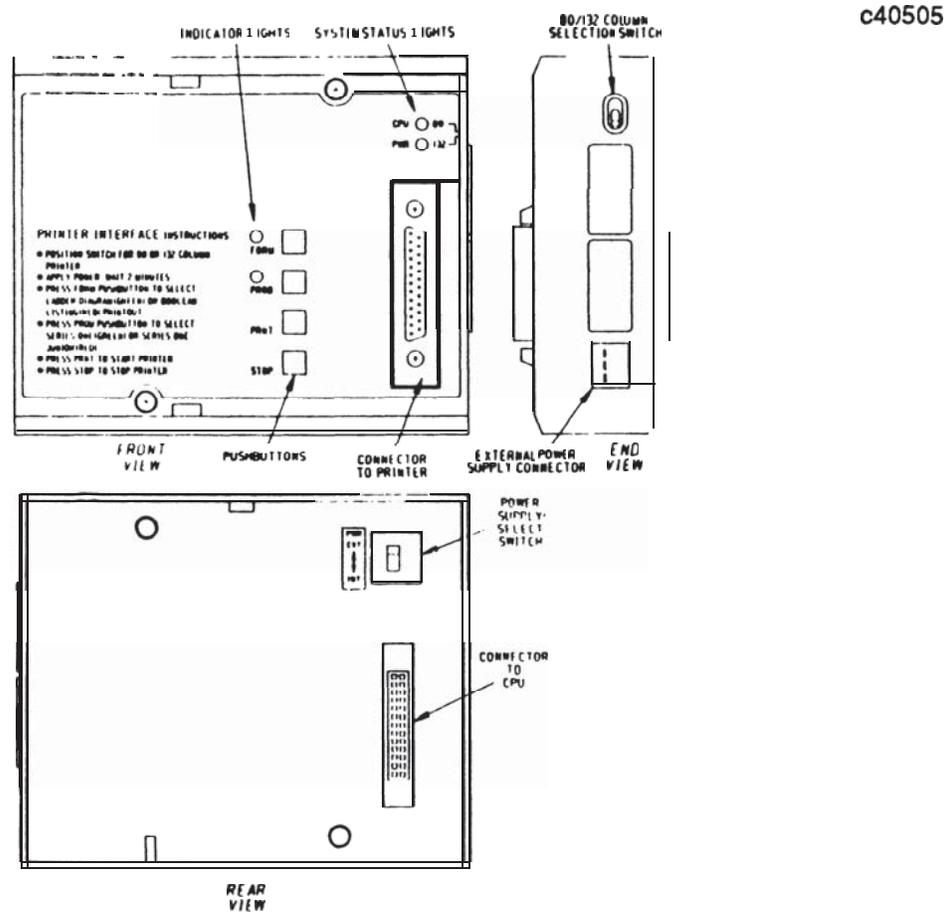


Figure 4-5. Printer Interface Unit

### Hardware Features

The Printer Interface Unit front panel has two **LEDs** located in the upper right corner used for visual indication of system status. The purpose of the indicators is described below.

CPU	The red CPU LED is an indication of the operating status of the CPU in the PC. This is identical to the CPU LED on the PC.
ON	CPU failure has been detected.
OFF	CPU operation is normal.
PWR	The green PWR LED is an indication of the status of dc power being supplied to the Printer Interface Unit.
ON	If power is being supplied by the Series One or Series One Plus internal power supply, this indicates that +5 V dc is being properly produced by the supply. If the Printer Interface Unit is being powered from an external source, the +5 V dc being supplied is within the specified tolerance.

GEL90842

OFF 5 V dc not being supplied or not in tolerance.

The **24-pin** connector located on the lower right of the front panel, provides a connection from the Printer Interface Unit to the selected printer through the Printer **Interface** cable, **IC610CBL152**.

Immediately to the left of the connector are **4** pushbuttons used to initiate operation of the Printer Interface Unit to get a hard copy printout. There are also **2 LEDs** used as status indicators during the set-up procedure.

**FORM** This pushbutton is used to select the format of the printout, either a ladder diagram or Boolean listing of the user program. The pushbutton is an alternate **function** switch. Each time it is depressed, **the** selected **printout format** will change.

As the pushbutton is depressed, the color of the LED will **alternate** between green and **red**. The LED, when **green**, indicates a ladder diagram printout; when red, indicates a Boolean listing printout.

**PROD** This pushbutton is also an alternate function switch. Each time it is **depressed**, either a Series One Junior PC or the Series One/Series One Plus PC is alternately selected as the program listing **source**. The **selection** must agree with the PC to which the **Printer** Interface Unit is attached.

As this pushbutton is **depressed**, the color of the LED **will alternate** between **green** and **red**. The LED, when red, indicates selection of Series One Junior, when green, indicates selection of Series **One/Series One Plus**.

**PRNT** When depressed, this pushbutton causes the printer to begin **printing** the user program in the selected **formats**.

**STOP** When depressed, this pushbutton causes the **printing** operation to stop.

### 80/132 Column Selection Switch

On the right side-panel is a **2-position** toggle switch used for selection of either 80 column printing format (compressed print) or 132 column printing format (normal print). The 80 column compressed print format is typically used with printers designed for use with personal computers. The selected printer column format corresponding with the switch position is printed on the right edge of the front panel, either 80 (towards the top of the unit) or 132 (towards the bottom of the unit). If desired, the 80 column compressed print format can be selected for use with a 132 column printer. The unused space to the right, beyond the 80th column, could be used for adding comments.

### External Power Supply Connector

A connector located on the lower right side of the unit provides the connections to an external power supply. A mating **3-pin** connector with attached wires which are **3** feet (1m) in length, is provided with the Printer Interface Unit for connection to the external supply. The color code for the external power supply cable and specifications for the power supply are as follows:

<b>White</b>	<b>+5 V dc, <math>\pm 5\%</math> (rated at 300 mA minimum)</b>
<b>Black</b>	Power supply logic ground
<b>Green</b>	Common system ground

### Power Supply Select Switch

**This is** a two-position switch located on the **bottom** of the unit, **directly** above **the 26-pin** connector. This switch is used for selection of either internal or external dc power for the Printer Interface **Unit**. The top switch position is labeled **EXT** (External) and the bottom position is labeled **INT** (Internal). If the Printer Interface Unit is to be powered by an external **+5 V** dc power **supply**, **the** Power Supply Select switch must be set to **EXT**.

## Sequence of Operation

As a convenience to the user, instructions for using the Printer Interface Unit are printed on the **front** panel of the unit. The sequence of operation is described below. If a Series One Plus program is to be printed and a password is in effect, the password must be changed to 0000 before mounting the Printer Interface Unit.

## Power-up Sequence

The following power-up sequence should be followed after the Printer Interface Unit has been properly configured.

1. Power to be supplied by the internal PC supply or an external power supply.
2. Place **EXT/INT** switch in the INT or **EXT** position as required.
3. Turn off power to the PC.
4. Mount Printer Interface Unit onto the PC.
5. Attach Printer Interface cable **from** connector on front panel of unit to printer.
6. Turn on power to the PC.
7. Turn on power to the external power supply, if used.

## User Program Transfer From PC To Printer Interface Unit

Immediately after the power-up sequence has been completed, the user program stored in the PC's user memory will automatically begin transferring to a buffer memory in the Printer Interface Unit. This program transfer will take about 1.5 minutes. When the program has been successfully transferred, the FORM and PROD LED indicators will turn on green. If the program transfer is not successful, the LED indicators will either flicker on and off red or neither LED will illuminate. If after 2 minutes, neither LED turns on, repeat the power-up procedure **from** the beginning of the sequence.

## Selection of Printout Format and Type of PC

Select the printout format, either ladder diagram or Boolean, and the PC model, either Series One Junior or Series One/Series One Plus by depressing the FORM and PROD switches as shown in the following table.

**Table 4-5. Format and PC Selection**

FORM		PROD	
PRINTOUT TYPE	LED ON	PC	LED ON
Ladder	<b>Green</b>	series one/be Plus	<b>Green</b>
Ladder	<b>Green</b>	Series <b>One</b> Junior	<b>Red</b>
<b>Boolean</b>	Red	series one/one Plus	<b>Green</b>
<b>Boolean</b>	Red	Series <b>One</b> Junior	<b>Red</b>

## Start Printer Operation

Depress PRNT pushbutton. The ladder diagram or Boolean program listing will begin to print and continue printing until the complete program has been listed or has been stopped by the operator.

If at any time, the program listing is to be stopped, depress the STOP switch. When this is done during a ladder diagram printout, the printout will stop. When the STOP switch is depressed during a Boolean

GEK-90842

listing printout, the **Boolean** listing printout will stop, the printer will formfeed, and the complete OUTPUTS USED TABLE will be **printed**.

**Printing Of Error Messages During Ladder Diagram Listing**

If any errors in the **printing** process are detected by the Printer Interface Unit during printing of a ladder diagram listing, an error message will be printed and the printing may stop, depending on the type of error. Table 4-6 lists the error messages and their definitions.

**Table 4-6. Ladder Diagram Listing Error Messages and Definitions**

ERROR MESSAGE	DEFINITION
ROW OVER COLUMN OVER STACK ERROR (SR) STACK ERROR (CNT) STACK OVER STACK OVER (MCS) MC ERROR PROGRAM ERROR	One rung of <b>logic</b> exceeds 16 lines. More than 13 <b>circuit</b> elements in line Clock or Reset line <b>not</b> proved in Shift Register logic. Reset line not <b>programmed in</b> Counter logic. Pushdown <b>stack</b> using AND STR and OR STR functions exceeds 8 levels. <b>Levels</b> of MCS control exceed 8. MCR functions exceed MCS <b>functions</b> . Any error not listed in this table.

NOTE

When the **ROW OVER** or **COLUMN OVER** error messages are printed, the printout of the ladder diagram will continue. When any other error messages are printed, a **PRINT STOP** message will be printed, paper will **feed** and printing will stop.

**Printing Of Error Messages During Boolean Listing**

If any errors in the printing process are detected by the Printer Interface Unit during printing of a Boolean program listing, an error message will be printed and the printing may stop, depending on the type of error. Table 4-7 is the Boolean listing **error** messages and their definitions.

**Table 4-7. Boolean Listing Error Messages and Definitions**

ERROR MESSAGE		DEFINITION
INSTRUCTION	OPERAND	
***ERROR***	???	Not a valid <b>instruction</b> <b>Incorrect</b> operand

**Cross Reference Printout**

When the ladder diagram or Boolean printout of the user program has been **completed**, the printer will **formfeed**, then begin to print a cross reference printout of all outputs. The heading of this printout is, **OUTPUTS USED TABLE**. The **outputs referenced** in the user program (real world **outputs**, internal relays, **shift** registers and timers/counters) will have an annotation **mark** to the right of the reference number.

The outputs used table will continue printing until **all** output references have been printed. This printout cannot be stopped, as can the ladder diagram and Boolean listing printouts.

## Expanded Print Format

When a more complex ladder diagram, using MCS and MCR functions, is to be printed out, an **expanded** print format is used. The starting point of each MCS function is denoted by a letter, the first is A, then B, etc. The letter is carried through to the end of each page and the beginning of the next page, providing a ready reference to the continuation of the ladder logic within the bounds of each **MCS/MCR** control. As multiple MCS functions and the logic under their control are printed, each succeeding group of logic is shifted to the right. When a group of logic under **MCS/MCR** control is ended by an MCR function, **(MCR)** is printed in the last column to the right and the letter corresponding to that **MCS/MCR** logic to its right.

## Printout Annotation Explanation

Several items appearing on the printouts in figures 4-6 and 4-7 **are** explained for clarification. The ladder diagram printout in figure 4-6 has a circled number (1-5) next to each annotation, which corresponds to the number preceding the explanation of each annotation. The circled number 6 appears before an annotation on the Boolean printout in **figure** 4-7. The circled numbers (1-6) are for discussion purposes only and do not normally appear on a printout.

1. The type of printout on each page appears on this line, either LADDER **DIAGRAM** PRINTOUT, BOOLEAN PRINTOUT, or CROSS REFERENCE PRINTOUT.
2. This annotation, **V X.X**, is the version of the system operating software contained in PROM memory in the Printer Interface Unit.
3. The model of PC selected by the user as the program listing source will be on this line. The annotation will be either SERIES ONE **JR/SR-10** for a Series One Junior PC or Series One, One **Plus/SR-20**, SR-21 for a Series One or Series One Plus PC.
4. The page number of the ladder diagram printout or Boolean printout will appear here as a **4-digit** decimal number, starting with PAGE **0001**.
5. Refers to ladder diagram printout only. This **4-digit** decimal number is the user program memory address at the start of each rung of logic. The **first** element in the rung is stored **at** that address. In the example in figure 4-6, the memory address of the start of the first rung is 0000. The first element in that rung is a normally open contact referenced as **001** (reference number is printed directly above the contact). The memory address at the start of the second rung is 0013, the first element in this rung is a normally open contact referenced as 040.
6. Refers to Boolean printout only (figure 4-7). The **#** sign immediately following a numerical value in the Boolean printout listing, **indicates that the** value is a reference assigned to an element at the end of a rung.
7. A symbol preceding a numerical value in a Boolean printout, indicates that the value is a constant.

GEK-90842

Sample Printout

A sample of each of the previously described printouts is shown in the following group of figures. For this group of printouts, a program was entered into a Series One Plus PC.

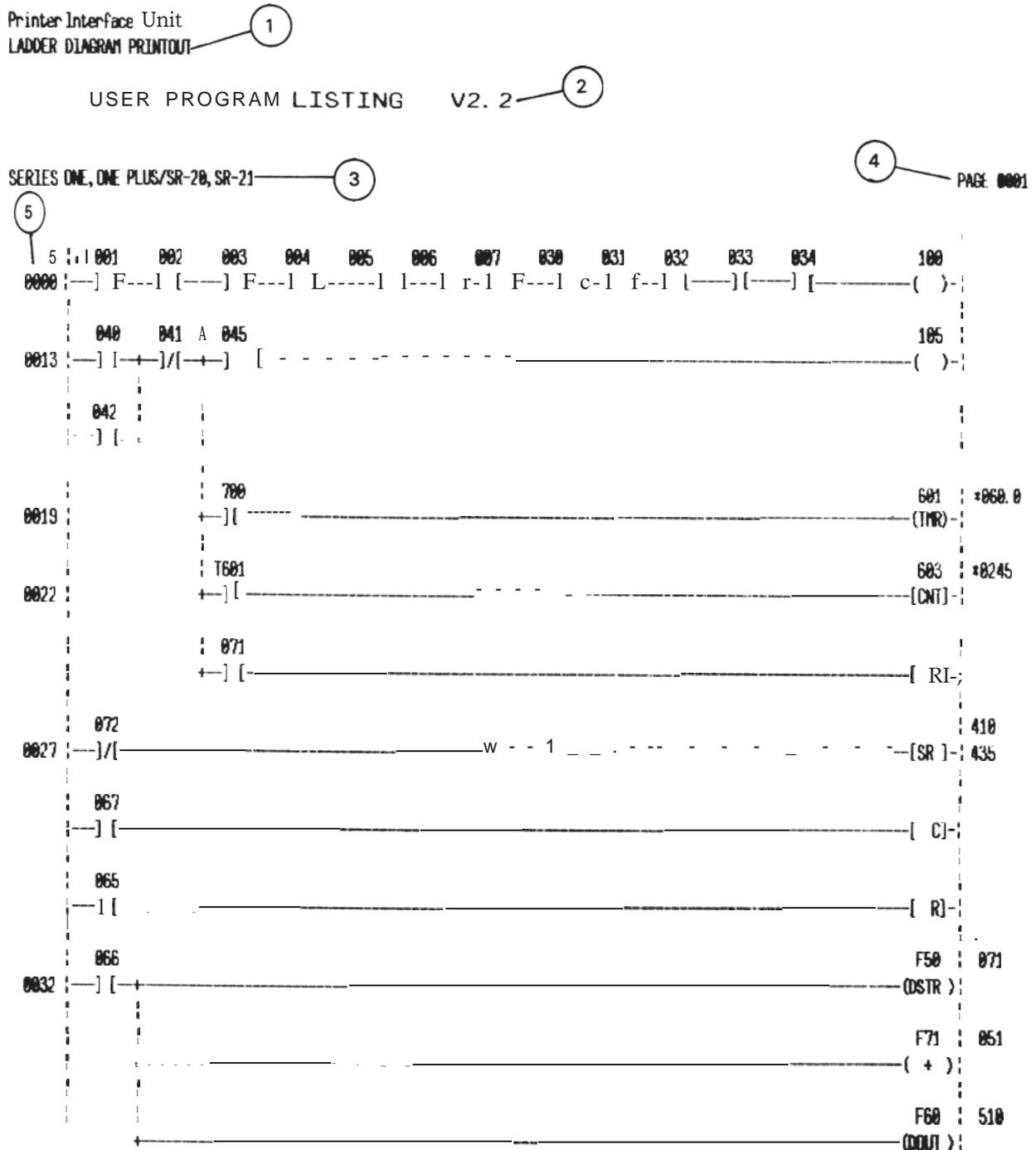


Figure 4-6. Sample Ladder Diagram Printout

Printer Interface hit  
 BOOLEAN PRINTOUT

USER PROGRAM LISTING

V2. 2

SERIES ONE, ONE PLUS/SR-20, SR-21

PAGE 0001

0000: STR	001	0050: -	0100: -	0150: -
0001: AND	002	0051: -	0101: -	0151: -
0002: AND	003	0052: -	0102: -	0152: -
0003: AND	004	0053: -	0103: -	0153: -
0004: AND	005	0054: -	0104: -	0154: -
0005: AND	006	0055: -	0105: -	0155: -
0006: AND	007	0056: -	0106: -	0156: -
0007: AND	030	0057: -	0107: -	0157: -
0008: AND	031	0058: -	0108: -	0158: -
0009: AND	032	0059: -	0109: -	0159: -
0010: AND	033	0060: -	0110: -	0160: -
0011: AND	034	0061: -	0111: -	0161: -
0012: OUT	1000	0062: -	0112: -	0162: -
0013: SIR	040	0063: -	0113: -	0163: -
0014: OR	042	0064: -	0114: -	0164: -
0015: AND NOT	041	0065: -	em: -	0165: -
0016: MCS		0066: -	0116: -	0166: -
0017: STR	045	0067: -	0117: -	0167: -
0018: OUT	1050	0068: -	0118: -	0168: -
0019: STR	700	0069: -	0119: -	0169: -
0020: THR	6010	0070: -	0120: -	0170: -
0021:	060 0	0071: -	8121: -	0171: -
0022: SIR THR	601	0072: -	0122: -	0172: -
0023: STR	071	0073: -	0123: -	0173: -
0024: CNT	6030	0074: -	0124: -	0174: -
0025:	0245	0075: -	0125: -	0175: -
0026: MCR		0076: -	0126: -	0176: -
0027: STR NOT	072	0077: -	0127: -	0177: -
0028: STR	067	0078: -	0128: -	0178: -
0029: STR	065	0079: -	0129: -	0179: -
0030: SR	4100	0080: -	0130: -	0180: -
0031:	4350	0081: -	0131: -	0181: -
0032: STR	066	0082: -	0132: -	0182: -
0033: DSIR	(F50)	0083: -	0133: -	0183: -
0034:	071	0084: -	0134: -	0184: -
0035: +	(071)	0085: -	0135: -	0185: -
0036:	051	0086: -	0136: -	0186: -
0037: DOUT	(F00)	0087: -	0137: -	0187: -
0038:	5100	0088: -	0138: -	0188: -
0039: -		0089: -	0139: -	0189: -
0040: -		0090: -	0140: -	0190: -
0041: -		0091: -	0141: -	0191: -
0042: -		0092: -	0142: -	0192: -
0043: -		0093: -	0143: -	0193: -
0044: -		0094: -	0144: -	0194: -
0045: -		0095: -	0145: -	0195: -
0046: -		0096: -	0146: -	0196: -
0047: -		0097: -	0147: -	0197: -
0048: -		0098: -	0148: -	0198: -
0049: -		0099: -	0149: -	0199: -

Figure 4-7. Sample Boolean Printout

GEK-90842

### PROM Writer Unit

The PROM Writer unit (catalog number **IC610PER154**, version B replaces version A and can be used with Series One Plus as well as Series One and Series One Junior) is a compact, easy to use device that connects directly to and mounts on a Series One or Series One Plus PC. When mounted on a PC, the PROM Writer unit is used to write the contents of user memory to a **2732A-2** PROM (Series One), or a **27256-25** for a Series One Plus, thereby providing a non-volatile means of user program storage. After being written to, the appropriate PROM can be installed in any Series One or Series One Plus PC as required. Programs stored in PROM memory will not be lost during no-power conditions.

An additional feature of PROM memory is that **different** programs can be stored on individual PROMS for use as required by various applications. Another function of the PROM Writer unit is to transfer the user memory contained in a PROM to the CMOS memory in a Series One or Series One Plus.

On a Series One or Series One Plus PC, the PROM Writer unit physically mounts on the right of the front panel, in the same manner as the programmer. A connector on the lower left rear of the PROM Writer unit attaches to the connector on the **front** panel of the Series One or Series One Plus PC. The source of power for the PROM Writer unit is switch selectable and can be **from** the internal supply of the PC or **from** an external source of 5 V dc. Figure 4-10 is an illustration of the PROM Writer showing the location of its features, which are described in the text following the illustration.

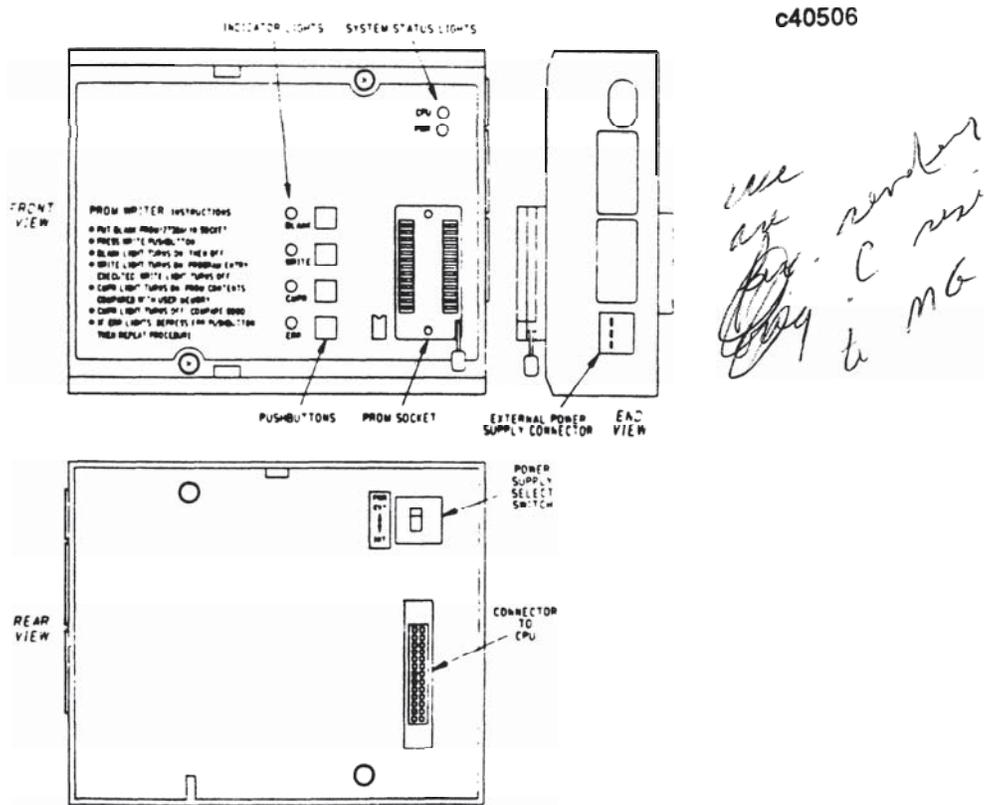


Figure 4-10. PROM Writer Unit Features

## Front Panel Features

The front panel has two **LEDs** in the upper right corner which are visual indicators of system status. The purpose of the each indicator is described below.

PWR	The <b>green</b> PWR LED monitors the status of dc power being supplied to the PROM <b>Writer</b> unit.
ON	If power is being supplied by the CPU rack, this indicates that <b>+5 V</b> dc is being produced by the supply. If the PROM Writer unit is being powered <b>from</b> an external source, the <b>+5 V</b> dc being supplied is within the specified tolerance.
OFF	5 V dc not being supplied or not in tolerance.
CPU	The red CPU LED monitors the operating status of the CPU module. Identical to the CPU LED on the Series One or Series One Plus CPU module.
ON	<b>CPU failure has been detected.</b>
OFF	CPU operation is normal.

The socket on the lower right of the panel is used to contain the PROM being written to. The socket is a zero insertion force socket. To insert a PROM into the socket, push the locking handle up, insert the PROM, then lock the PROM in place by moving the handle down to the horizontal position. The PROM should be placed in the socket with the notch towards the end of the socket closest to the PWR LED as indicated by the figure on the panel next to the socket. Even though the spacing of the slots in the socket allows easy insertion of a PROM, care should be taken to ensure that leads on the PROM are not damaged.

Immediately to the left of the socket are 4 pushbuttons and their associated LED indicators. These pushbuttons are used to initiate operation of the PROM Writer unit and the **LEDs** are indicators for each **part** of the operation.

**BLANK** When **depressed**, initiates checking of the PROM inserted in the socket for verification **that the** PROM does not have any **information** written into it. To initiate a blank check, the pushbutton is **depressed**. **The** light will turn on, then off, indicating a successful blank check. If the light remains on, and the ERR light turns on, the PROM **needs to be erased**.

**The BLANK light will also turn on**, then off during the normal operation of writing to a PROM as indicated by the instructions printed on the lower left of the **unit**.

**WRITE** When depressed, the WRITE pushbutton initiates the sequence of events that causes the user program in the Series One or Series One Plus CMOS memory to be **written** to the PROM. When the WRITE pushbutton is **depressed**, **the BLANK light will turn on**, then off. Next, the WRITE **light turns on**, program entry is executed, then **the WRITE light turns off**. A blank check and **verify are** performed **automatically** when the WRITE pushbutton is **depressed**.

**CMPR** During the sequence for writing to a PROM, this **light will turn on** while the contents of **the** PROM are being compared to the contents of user memory in RAM. The CMPR light **turns off when** the compare is completed **and is good**. **If the compare is not good, the CMPR light will remain on and the ERR light will turn on**. In addition, the contents of a PROM inserted in the PROM Writer unit socket can be **compared to the** contents of user **memory, whenever** the CMPR pushbutton is depressed.

**ERR** This light is a visual indication that the PROM writing operation has not been **successful**. If the **light turns on** during any portion of the **operation**, an error has **occurred**. If this does happen, depress **the** ERR pushbutton and **repeat the procedure**.

As a convenience to the user, the PROM Writer unit instructions for writing the contents of user memory to a PROM **are** printed on the lower **left** of the **unit**.

GEK-90842

### Sequence of Operation

The sequence of operation for writing the contents of user RAM memory to PROM memory is as follows:

WRITE OPERATION SEQUENCE	INDICATION
Depress WRITE pushbutton	WRITE light turn on
Blank check performed	WRITE light <b>turns</b> off
	BLANK light turns on
Write to PROM	BLANK light turns off
	<b>WRITE light turns on</b>
Verify contents of PROM with contents of RAM memory	WRITE light turns off
	CMPR light <b>turns</b> on
Write sequence successful	CMPR light turns off

The sequence of operation for transferring the contents of PROM memory to CMOS memory is as follows (the CMOS memory should be cleared first):

PROM TO RAM OPERATION SEQUENCE	INDICATION
Depress WRITE and <b>CMPR</b> pushbutton at the same time. Contents of PROM <b>will</b> be <b>transferred</b> to RAM memory in the PC.	WRITE light and <b>CMPR light will turn on.</b>
Contents of PROM <b>and RAM</b> memory are compared.	WRITE light turns off.
<b>Compare</b> good. Sequence of operation complete.	<b>CMPR light turns off *</b>

\*If an error is detected during the compare operation, the CMPR light will remain on and the ERR light will turn on.. The error can be cleared by depressing the ERR pushbutton. **When** this is done, the ERR and CMPR lights will turn off. If an **error** is **indicated**, repeat the operation.

### External Power Supply Connector

A connector located on the right side of the PROM Writer **unit** provides the connections to an external power supply. A mating **3-pin** connector with attached wires 3 feet (1m) in length, is provided with the PROM Writer unit for connection to the external supply. The color code for the external power supply cable and specifications for the power supply are as follows:

<b>White</b>	+5 V dc, <u>+5%</u> (rated at 0.5 amps)
<b>Black</b>	Power supply logic ground
<b>Green</b>	common system ground

### Power Supply Select Switch

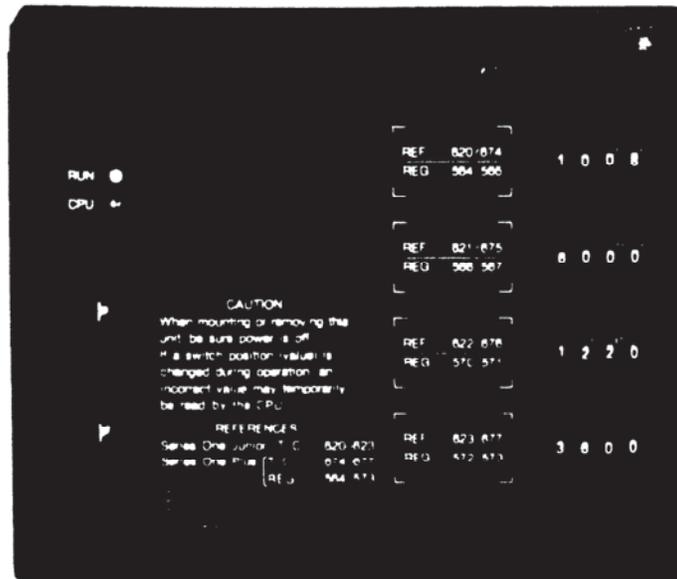
**This is a** two-position switch located on the bottom of the PROM Writer unit, directly **above the 26-pin** connector. The switch is used for selection of either internal or external dc power for the PROM Writer unit. The top switch position is labeled EXT (External) and the bottom position is labeled INT (Internal). When used with a Series One or Series One Plus PC mounted in a high-capacity rack, power can be supplied internally and the Power Supply Select Switch **is** set to INT.

If erratic **operation** is observed when attempting a data transfer using the PROM Writer, it is recommended ~~that~~ an external source of **5 V** dc be used.

## Timer/Counter Setpoint Unit

The Timer/Counter **Setpoint** Unit (IC609TCU100) provides an alternative method for entering a timer or counter preset value in a Series One Plus program. This unit can also be used with a Series One Junior. A Timer/Counter **Setpoint** Unit mounts on the Series One Plus rack that contains the CPU and plugs into the same connector as does the hand-held programmer. The Timer/Counter **Setpoint** Unit can also be connected remotely by using the 5 foot (1 .5m) remote programmer cable, IC610CBL102. With the unit mounted on the PC, the hand-held programmer can then be mounted on top of the Timer/Counter **Setpoint** Unit, thereby providing a convenient way to monitor the operation of timers or counters. The physical size of the unit is the same as the Data Communications Unit, Printer Interface Unit and the PROM Writer Unit.

a41 673



**Figure 4-11.** Timer/Counter **Setpoint** Unit

There are four **4-digit thumbwheel** switches on the unit, which provides a convenient way to enter a **4-digit BCD** value into each of 4 specific internal locations in the Series One Plus PC for use as Timer/Counter preset values. These values are simultaneously entered into data registers (two **8-bit** registers for each BCD digit) for use as preset values or for other data operations. When the BCD preset values have been entered into the PC, they are retained in the PC's memory as **presets**, even though power is removed **from** the PC and the unit is removed.

GEK-90842

**Timer/Counter Setpoint Unit Specifications**

The following table contains general specifications for the Timer/Counter **Setpoint** Unit.

**Table 4-8. Timer/Counter Setpoint Specifications**

<b>Number of Circuits</b>	4 (4 BCD digits <b>per circuit</b> )
<b>Timer/Counter References</b>	674, 675, 676, 677
<b>Register References</b>	564, 565 (T/C 674), 566, 567 (T/C 675) 570, 571 (T/C 676), 572, 573 (T/C 677)
<b>Preset values</b>	Timer: 0.1 to 999.9 seconds .01 to 99.99 seconds ( <b>Series One Plus 3.7 only</b> )
<b>Ambient Temperature</b>	<b>Counter:</b> 1 to 9999 events
<b>Storage Temperature</b>	0° to 50° (32° to 122°F)
<b>Humidity (non-condensing)</b>	-20° to 85°C (-4° to 185°F)
<b>Internal Power Consumption</b>	5% to 95%
<b>Units of Load</b>	5 V dc, 40 mA (maximum) 9 V dc, 5 mA (maximum)
<b>Environment Considerations</b>	4 units @ 5 V dc 1 unit @ 9 V dc No <b>corrosive</b> gases

**Remote Mounting of Timer/Counter Setpoint Unit**

A Unit Mounting Bracket, **IC610ACC190**, is available which allows mounting of the Timer/Counter **Setpoint** Unit on the outside of a panel or console. The Unit Mounting Bracket consists of a mounting bracket, connector clamp and a cable clamp. The Timer/Counter **Setpoint** Unit mounts on the bracket, secured by two captive screws on the unit. The unit connects to a Series One Plus PC through the round 5 foot (1 .5m) remote programmer cable, **IC610CBL102**.

**References for the Timer/Counter Setpoint Unit**

The Timer/Counter **references** for the memory locations into which the BCD values are entered in the Series One Plus are **674, 675, 676** and **677**. Each of the references refer directly to a memory location in the CPU that accepts one **4-digit** BCD value as it is entered with each **4-position** thumbwheel switch. Each thumbwheel position represents one BCD digit, with the least significant digit being the position to the right.

Each of the register references refers to an **8-bit** data register. Two consecutive registers are **required** for each **4-digit** BCD number.

Since all four BCD values are read into the PC each scan, discretion must be exercised when changing any values when the PC is running, since undesired intermediate values could be read by the CPU and used during one or several scans. It is recommended that the following CAUTION be followed.

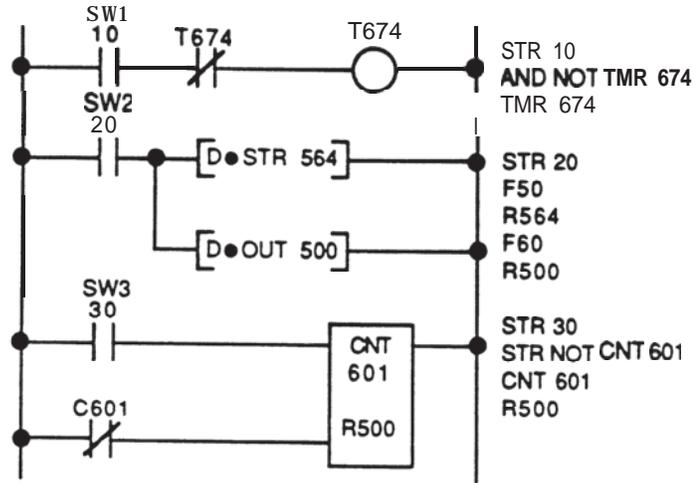
**CAUTION** |

**When mounting or removing the Timer/Counter Setpoint Unit, be sure that power is turned off. If a switch position (value) is changed during operation, an incorrect value may temporarily be read into the CPU.**

**Example of Using Thumbwheel Inputs**

The following example of a ladder **diagram rung** shows how the Timer/Counter **Setpoint** Unit is used to enter a preset value into a Timer or counter, or to enter data into data **registers**.

a421 86



**TIMER 674 USES THUMBWHEEL DATA FOR ITS PRESET VALUE.**

**WHEN SWITCH 2 IS CLOSED, THE THUMBWHEEL IS READ IN AND STORED IN A SAVE REGISTER, WHICH IS R500 IN THIS EXAMPLE.**

**THE VALUE IN THE SAVE REGISTER (500) IS USED AS THE PRESET VALUE FOR COUNTER 601.**

STR 10  
AND NOT TMR 674  
TMR 674

STR 20  
F50  
R564  
F60  
R500

STR 30  
STR NOT CNT 601  
CNT 601  
R500

## SECTION 1 Introduction to Programming

---

### General Information

The Series **One™** and **Series One™ Plus** are easy to use small PCs with many advanced features. One of the advantages of these PCs is the ability to be programmed or tailored specifically to the needs of an application. Furthermore, if requirements change, the PC can be reprogrammed to fit the new application. This chapter provides the information a user needs to develop or modify the logic within the Series One or Series One Plus. It will discuss the basics of CPU operation necessary to develop a proper logic program, the theory behind each function, examples of how that function could be used, and step-by-step **entry** of sample programs.

### Planning a PC System

When planning a **PC** system, the first step should be to **define** the system by writing a description of the **functional** requirements for that system. A description would usually consist of block diagrams and written descriptions of the various parts of the system. Input **devices** should be defined along with the process or machines to be **controlled**. The next step would be to develop the program required to control the system.

On a sheet of paper, **layout each rung of your ladder diagram and assign references to** inputs and outputs. The program can also be **written in mnemonic form (Boolean), which is how** it will be keyed into the Series One or Series One Plus when using the hand-held programmer. As an option, programs can be entered in ladder diagram Format using the Portable Programmer. This applies to both the Series One and Series One Plus. You can also enter programs with the **Workmaster®** industrial computer using **Logicmaster™1** application software. For **information** on using the **Workmaster** industrial computer for **programming**, refer to the **Logicmaster 1 Programming** manual, **GEK-96632** for Series One. For Series One Model E, Series One Plus, and Series One Plus **3.7K**, refer to GFK-0075 which is the **Logicmaster 1 Family Programming Manual**.

### How to Use This Chapter

This chapter is **divided into 3 sections**. Section 1 is a description of the requirements for programming. Included, are lists of all of the **programming** functions and a table of programming references assigned to each slot for 8 point modules. The purpose and types of references are explained, including "real world" use and internal use. A basic explanation of the scanning operation of the Series One and Series One Plus PCs is given. The scanning process is the basis for operation of all PCs and the user should have a good working knowledge of this operation.

The next 2 sections are devoted to programming. Section 2 describes and gives examples of the basic ladder diagram functions. Section 3 provides the user with a **description** of the data operation functions for use with a Series One Plus PC. The data operations greatly **extend the** number and complexity of applications **possible with the Series One** family.

Table 5.1 lists the basic ladder diagram functions for the Series One and Series One Plus. A more detailed explanation of each instruction can be found later in this chapter. Execution times in Table 5.1 are for Series One Model E, Series One Plus, and Plus 3.7K only. Execution times for Series One are listed in Table 5.2. Note that in the table, I/O = I/O points, IR = Internal Relay, SR = Shift Register.

**Table 5-1. Series One Model E/One Plus Basic Ladder Diagram Instructions**

Instruction	Definition	Memory Words	Execution Time (microseconds) For Series One Plus	
			Inactive	Active
STR	START. Start rung with a N.O. (Normally Open) contact.	1	6.6 50.9	6.6 I/O,IR,SR 50.9 Sequencer
STR TMR	START TIMER. Start rung with a N.O. contact referencing a timer.	1	10.3	10.3
STR CNT	START COUNTER. Start rung with N.O. contact referencing a counter.	1	10.3	10.3
STR NOT	START NOT. Start rung with a N.C. (Normally Closed) contact.	1	9.1 61.5	9.1 I/O,IR,SR 61.5 Sequencer
STR NOT TMR	START NOT TIMER. Start rung with a N.C. timer contact.	1	12.8	12.8
STR NOT CNT	START NOT COUNTER. Start rung with a N.C. counter contact.	1	12.8	12.8
AND	Add a N.O. contact in series with the previous contact.	1	5.3 6.2	5.3 I/O,IR,SR 59.1 Sequencer
AND TMR	AND TIMER. Add a N.O. timer contact in series with the previous contact.	1	5.3	5.3
AND CNT	AND COUNTER. Add a N.O. counter contact in series with the previous contact.	1	5.3	5.3
AND NOT	Add a N.C. contact in series with the previous contact.	1	8.4	8.4 I/O,IR,SR 60.3 Sequencer
AND NOT TMR	AND NOT TIMER. Add a N.C. timer contact in series with the previous contact.	1	8.4	8.4
AND NOT CNT	AND NOT COUNTER. Add a N.C. counter contact in series with the previous contact.	1	8.4	8.4
OR	Add a N.O. contact in parallel with the previous contact.	1	6.6 6.2	6.6 I/O,IR,SR 60.3 Sequencer
OR TMR	OR TIMER. Add a N.O. timer contact in parallel with the previous contact.	1	6.6	6.6
OR CNT	OR COUNTER. Add a N.O. counter contact in parallel with the previous contact.	1	6.6	6.6
OR NOT	Add a N.C. contact in parallel with the previous contact.	1	9.1 6.2	9.1 I/O,IR,SR 62.5 Sequencer
OR NOT TMR	OR NOT TIMER. Add a N.C. timer contact in parallel with the previous.	1	9.1	9.1

GEK-90842

**Table 5-1. Series One Model E/One Plus Basic Ladder Diagram Instructions - Continued**

Instruction	Definition	Memory Words	Execution Time (microseconds) For Series One Plus	
			Inactive	Active
OR NOT CNT	OR NOT COUNTER. Add a N.C. counter contact in parallel with the previous <b>contact</b> .	1	9.1	9.1
AND STR	AND STORE. Connects a logic group in series with the logic group preceding it.	1	38	
OR STR	OR STORE. Connects a logic group in parallel with the logic group <b>preceding</b> it.	1	38	
MCS	MASTER CONTROL START. Begin control of a block of logic with a master control relay.	1	5.0	
MCR	MASTER CONTROL RESET. Ends control of a block of logic with a master control relay.	1	3.0	
OUT	Defines a coil for a rung of logic, either an external output or an <b>internal coil</b> .	1	75	
SET	<b>Defines</b> a coil as being <b>latched</b> , the coil will remain on until turned off by RST.	1	17.5	
SET OUT	IF ON, the coil will remain ON even under control of the Output Disabler Coil (376).	1	10.0	
RST	RESET. Causes a coil that had been latched (SET) to be <b>unlatched</b> (Reset).	1	93	
SET OUT RST	SET OUT RESET. Defines a coil as being a one-shot (ON for one <b>scan</b> ).(One Plus Only)	1	19.3	
TMR	<b>TIMER</b> . Programs a Timer operation. Times up <b>from</b> zero to a preset value. Elapsed time stored in an accumulate register. Preset time range is 0.1 seconds to 999.9 seconds.	2	27.1	700.0 I/O,IR,SR 458.8 Register
CNT	COUNTER. Programs an up counter. Counts up from zero to a preset value which can be 1 to 9999. The <b>current</b> count is stored in an accumulate register. Counters are retentive. <b>Requires</b> 2 rungs of logic; <b>first</b> rung enables the counter, the second rung resets the counter.	2	27.1	706.8 I/O,IR,SR 465.6 Register
SR	<b>SHIFT REGISTER</b> . Programs a shift register which can be 128 steps in length or a number of shift registers of varying lengths (128 <b>bits total</b> ). <b>Retentive</b> . <b>Requires</b> 3 rings of logic; (1) <b>data</b> , (2) <b>clock</b> , (3) <b>reset</b> . can be <b>programmed to shift forward or backward</b> .	2	53.1	64.1 <b>+16.6</b> For each bit shifted

Note: I/O = I/O Points, IR = Internal Relay, SR = Shift Register

**Table 5.2** is a list of execution times for the Series One functions. Execution times are given in microseconds.

Table 5-2. Series One Execution Times

Instruction	I/O Point	Int Relay	Shift Register	Timer	Sequencer	
					Counter	Inactive
AND	33.9		32.7	37.5	36.0	84.0
AND NOT	36.5		35.1	39.9	40.8	88.8
OR	36.3		34.5	43.5	36	84.0
OR NOT	38.7		36.9	45.6	40.8	88.8
STR	40.8		39.6	48.0	89.4	89.4
STR NOT	42.0		43.2	50.4	97.2	97.2
SET	38.1		42.6			
RST	38.1		42.6			

Instruction	Inactive	Active
MCS	35.2	35.2
MCR	37.8	37.8
SET OUT	49.8	49.8
OUT	43.8	43.8
TMR	69.6	115.8
CNT	80.4	103.5
SR	55.2	66.9 + 12.9/N (1)
ANDSTR	34.2	34.2
OR STR	37.8	37.8

(1) N = number of bits shifted.

Table 5.3 is a list and description of the data operations that can be programmed, with a Series One Plus, in addition to the basic ladder diagram functions listed in Table 5.1. A more detailed explanation of each instruction is provided later in this chapter.

Table 5-3. Series One Plus Data Operation Instructions

Instruction	Definition	Memory Words	Execution Time (microseconds)	
			Inactive	Active
D•STR (F50)	DATA STORE. Loads accumulator with a binary or BCD value which can be a 4 digit constant or the contents of a specified 2 byte reference.	2	6.3	321.9 I/O,IR,SR 80.7 Register 14.3 Constant
D•STR1 (F51)	DATA STORE 1. Loads the lower byte (least significant) of the accumulator with the contents of a specified 1 byte reference.	2	6.3	140.9 I/O,IR,SR 63.8 Register
D•STR2 (F52)	DATA STORE 2. Loads the lower 4 bits of the accumulator with the upper 4 bits of a specified 1 byte reference.	2	6.3	172.2 I/O,IR,SR 95.0 Register
D•STR3 (F53)	DATA STORE 3. Loads the lower 4 bits of the accumulator with the lower 4 bits of a specified 1 byte reference.	2	6.3	173.8 I/O,IR,SR 96.6 Register

GEK-90842

**Table 5-3. Series One Plus Data Operation Instructions - Continued**

Instruction	Definition	Memory Words	Execution Time (microseconds)	
			Inactive	Active
D•STR5 (F55)	DATA STORE 5. Loads accumulator with a binary or BCD value <b>from</b> the 2 byte contents of a 16 circuit (high density) Input Module.	2	6.3	325.0 16 Point Input Module
D•OUT (F60)	DATA OUT. The contents of the accumulator (2 bytes) <b>are written</b> to a specified 2 byte <b>reference</b> (Output, Internal Relay, Shift Register Coils, Data Register or T/C Accumulate Register).	2	6.3	329.4 I/O,IR,SR 52.6 Register
D•OUT1 (F61)	DATA OUT 1. The contents of the lower byte of the accumulator are written to a <b>specified</b> 1 byte reference (Output, Internal Relay, Shift Register or Data Register).	2	6.3	160.1 I/O,IR,SR 39.1 Register
D•OUT2 (F62)	DATA OUT 2. The contents of <b>the</b> lower 4 bits of the accumulator are written to the upper 4 bits of a specified 1 byte reference (Output, Internal Relay, Shift Register or Data Register).	2	6.3	116.0 I/O,IR,SR 39.8 Register
D•OUT3 (F63)	DATA OUT 3. <b>The</b> contents of the lower 4 bits of the accumulator are written to <b>the</b> lower 4 bits of a specified 1 byte reference (Output, Internal Relay, Shift Register or Data Register).	2	6.3	108.1 I/O, IR, SR 55.0 Register
D•OUT5 (F65)	DATA OUT 5. The contents of the accumulator (2 bytes) are <b>written</b> to a specified 2 byte <b>reference</b> . Must be a 16 circuit ( <b>high density</b> ) Output module.	2	6.3	365.3 16 Point Output Module
= < (F70)	COMPARE. <b>The</b> contents of the accumulator are compared to the contents of a specified 2 byte <b>reference</b> (I, IR, SR, or DR) or a 4 <b>digit</b> BCD constant. <b>Internal</b> coils turn on to reflect the result of the comparison. <b>Acc &gt; DataOn</b> 772 <b>Acc = DataOn</b> 777 <b>Acc &lt; Data011</b> 774	2	6.3	354.0 I/O,IR,SR 112.8 Register 57.0 constant
+ (F71)	ADDITION (BCD). The contents of the accumulator are <b>added to a 4 digit BCD constant or the contents of a specified</b> 2 byte reference which must be a valid BCD number (Input, Internal Relay, Shift Register or Data Registers).	2	6.3	698.0 I/O,IR,SR 456.8 Register 262.0 Constant
- (F72)	SUBTRACTION (BCD). <b>The</b> contents of a specified 2 byte <b>reference</b> (Input, Internal Relays, Shift Register, or Data Registers) or a 4 digit BCD constant <b>are subtracted from the contents</b> of the accumulator.	2	6.3	557.0 I/O,IR,SR 3 15.8 Register 275.0 Constant
X (F73)	MULTIPLICATION (BCD). The contents of the accumulator <b>are</b> multiplied by <b>the</b> contents of a specified 2 byte reference (Input, Internal Relays, Shift Register, or Data Registers) or a 4 digit BCD <b>constant</b> . <b>The lower</b> 4 digits of the result remain in the accumulator, the upper 4 digits <b>are stored</b> in registers 576 and 577 which is the <b>auxiliary</b> accumulator.	2	6.3	497 to 2851 (I/O,IR,SR) 290 to 2644 (Register) 223 to 2576 (Constant)

Table 5-3. Series One Plus Data Operation Instructions - Continued

Instruction	Definition	Memory Words	Execution Time (microseconds)	
			Inactive	Active
÷ (F74)	DIVISION (BCD). The contents of the accumulator are divided by the contents of a specified 2 byte reference (I, IR, SR, or DR) or a 4 digit BCD constant. The first 4 digits of the answer are stored in the accumulator and the remainder is stored in the remainder is stored in the auxiliary accumulator (registers 576 and 577).	2	63	1218 to 2851 (I/O, IR, SR) 742 to 2645 (Register) 720 to 2577 (Constant)
D•AND (F75)	DATA AND (Logic product). The contents of the accumulator are logically AND-ed with each corresponding bit of a specified 2 byte reference (I, IR, SR, or DR) or a 4 digit BCD constant. The result is stored in the accumulator.	2	63	345.0 I/O,IR,SR 103.7 Register 55.6 Constant
D•OR (F76)	DATA OR (Logic Sum). The contents of the accumulator are logically OR-ed with each corresponding bit of a specified 2 byte reference (I, IR, SR, or DR) or a 4 digit BCD constant. The result is stored in the accumulator.	2	63	345.0 I/O,IR,SR 103.7 Register 55.6 Constant
SHIFT RIGHT (F80)	FUNCTION 80, SHIFT RIGHT N BITS. Contents of the accumulator are shifted to the right by the specified number of bits (1-15). Positions from which bits were shifted are filled with zeros.	2	63	216 + 13.4/N of bits shifted
SHIFT LEFT (F81)	FUNCTION 81, SHIFT LEFT N BITS. Contents of the accumulator are shifted to the left by the specified number of bits (1-15). Positions from which bits were shifted are filled with zeros.	2	63	220 + 13.4/N of bits shifted
DECODE (F82)	FUNCTION 82, DECODE. Decodes the lower 4 bits of the accumulator to a decimal number from 0 to 15 and places a 1 in the bit position in the accumulator which corresponds to that decimal number.	1	63	56.3
ENCODE (F83)	FUNCTION 83, ENCODE. Encodes a bit in the accumulator to a binary code representing the position number (0 - 15) and places the binary code in the lower four bits of the accumulator.	1	63	282.0
INV (F84)	INVERT. The 16 bits of the accumulator are inverted; i.e., 1 to 0 and 0 to 1. Example: 1101 1001 0110 0111 inverted to 0010 0110 1001 1000	1	63	30.3
BIN (F85)	BINARY. Converts a BCD value in the accumulator to a binary code.	1	63	412.2
BCD (F86)	BINARY CODED DECIMAL. Converts a binary code in the accumulator to a BCD value.	1	63	746.0
EXTERNAL FAULT DIAGNOSIS (F20)	FUNCTION 20, EXTERNAL FAULT DIAGNOSIS. Allows external &vices to be monitored for unwanted conditions. If a fault occurs a preassigned 4 digit BCD number is displayed on the programmer. Multiple fault conditions can be monitored.	1	63	355.3 I/O,IR,SR 114.0 Register 722 Constant

## Programming Fundamentals

This section provides a guide for entering and using each of the instructions. The instructions are presented in groups that correspond to their functionality. Table 5.4 lists the instructions grouped in this manner.

**Table 5-4. Series One/One Plus Instruction Groups**

BASIC INSTRUCTIONS	
	STR, STR TMR, STRCNT,STR NOT, STR NOT TMR, STR NOT CNT AND, AND TMR, AND CNT, AND NOT, AND NOT TMR, ANDNOTCNT OR, ORTMR, ORCNT, ORNOT, ORNOTTMR, ORNOTCNT ANDSTR, ORSTR MCS, MCR, OUT, SET, SET OUT, RST, SET OUT RST (ONE PLUS ONLY) TMR, CNT, SR
DATA OPERATIONS	
SERIES ONE PLUS ONLY	
MOVE	D•STR (F50), D•STR1 (F51), D•STR2 (F52) D•STR3 (F53), D•STR5 (F55) D•OUT (F60), D•OUT1 (F61), D•OUT2 (F62) D•OUT3 (F63), D•OUT5 (F65)
ARITHMETIC	>=< (F70), + (F71), - (F72), x (F73), ÷ (F74)
LOGICAL	D•AND (F75), D•OR (F76), INV (F84)
CONVERT	BIN (F85), BCD (F86), Shift Right (F80) Shift Left, (F81) Decode (F82), Encode (F83)
SPECIAL	External Fault Diagnosis (F20)

## Significance of References

Whenever programs are entered into any PC, they must be accompanied by reference numbers. These references help to tell the CPU which function is specified. For example, which pushbutton controls the starting of which motor? Which timer are you referring to? Reference numbers are a vital part of programming, and in the Series One and Series One Plus are octal based; that is, they start at 0 and go up to 7, then jump to 10; at 77 the next value is 100. They look like traditional decimal numbers, except the digits 8 and 9 do not exist. Table 5.5 **summarizes** the various reference values and their **significance**.

Table 5-5. Summary of References

Octal Values	References	Decimal Quantity
000-157	I/O points (Series One)	112 total
000-157	I/O points (Series One Plus)	168 total
700-767	(Series One Plus only)	
770-777	Special Function Coils (Series <b>One</b> Plus only)	8
160-337	<b>Internal</b> Coils (Non-retentive)	112
340-373	Retentive Coils ( <b>Internal</b> )	28 (1)
374-377	Special Functions	4 (1)
400-577	Shift Registers	128 (1)
600-677 (4)	Timers and/or Counters	64 (2) (1)(2)
400-577	Data Registers, 16-bit (Series One Plus only)	64 (3), 124 Total if T/C references used as data registers

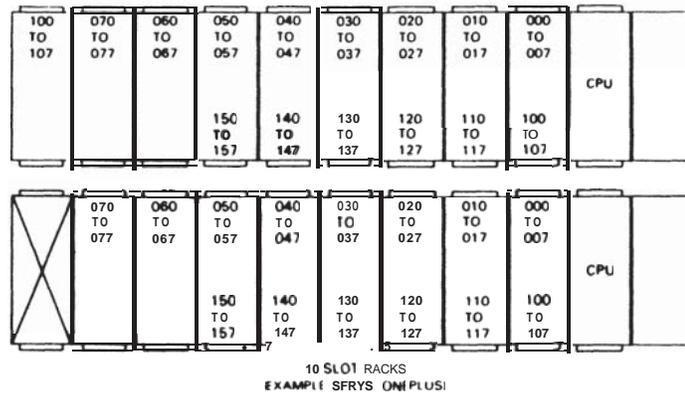
1. Retentive upon power failure.
2. Total maximum number of Timers and/or Counters.
3. **Shift register and data register** references are identical however, shift registers operate on bits, while data registers (located in a different area of memory) operate on bytes.
4. T/C references 600-673 can be used as data registers, if they are not referenced as timers or counters.

## Significance of Input/Output References

I/O modules can be placed in any mix desired; however, once installed the exact reference used in programming is established by their physical placement. In other words, references are assigned to each location by the operating system in the CPU, and therefore **are fixed** for each location. As shown in the example in Figure 5.1, the modules are numbered **from** zero (adjacent to CPU) towards the **left** through all racks until the last module is reached at the left of the last rack. The exact reference is obtained by appending the circuit number (**0-7**), to the module number. Circuit numbers depend upon the terminal to which the field device is wired as discussed in Chapter 6. Table 5.6 summarizes available I/O references for 8 circuit modules for your convenience. This table can be reproduced locally to form a part of your system documentation. Notice that whether reference 005 is an input or an output depends upon which module is inserted into the I/O slot adjacent to the CPU. Refer to the Series One family compatibility guide in Appendix C for additional valid I/O references per physical location of modules.

GEK-90842

a40797



pc-s1-83-0068

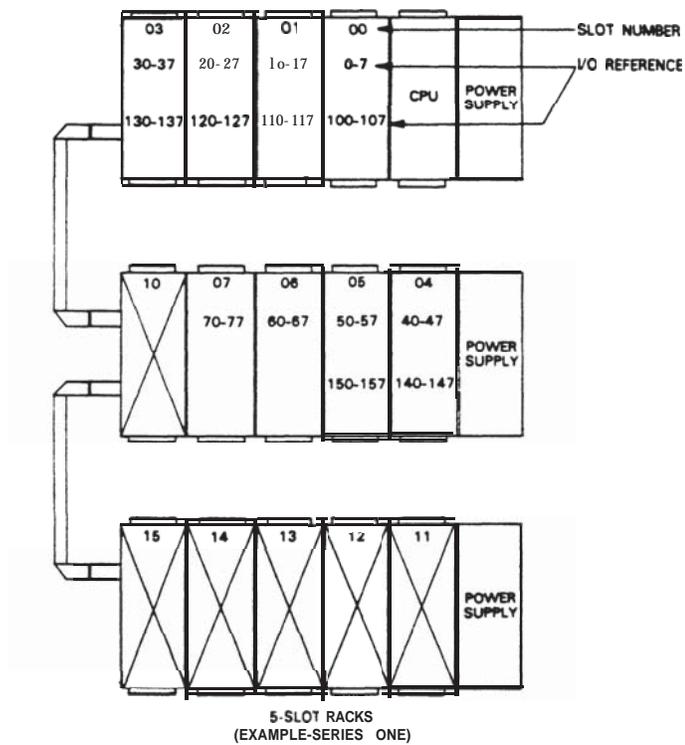


Figure 5-1. I/O References Per Physical Placement for 5 and 10-Slot Racks

NOTE

The addressing of slot 10 in a **10-slot** rack is determined by configuration of the two bridge connectors on the rack backplane.

Table 5-6. Summary of I/O References for 8 Circuit Modules

Module No.	Circuit No.	Reference	In/Out	Device Name	Wire No.
00	0	000			
	1	001			
	2	002			
	3	003			
	4	004			
	5	005			
	6	006			
	7	007			
01	0	010			
	1	011			
	2	0i2			
	3	013			
	4	014			
	5	015			
	6	016			
	7	017			
02	0	020			
	1	021			
	2	022			
	3	023			
	4	024			
	5	025			
	6	026			
	7	027			
03	0	030			
	1	031			
	2	032			
	3	033			
	4	034			
	5	035			
	6	036			
	7	037			
04	0	040			
	1	041			
	2	042			
	3	043			
	4	044			
	5	045			
	6	046			
	7	047			
05	0	050			
	1	051			
	2	052			
	3	053			
	4	054			
	5	055			
	6	056			
	7	057			
06	0	060			
	1	061			
	2	062			
	3	063			
	4	064			
	5	065			
	6	066			
	7	067			

GEK-90842

Table 5-6. Summary of I/O References for 8 Circuit Modules - Continued

Module No.	Circuit No.	Reference	In/Out	Device Name	Wire No.
07	0	070			
	1	071			
	2	072			
	3	073			
	4	074			
	5	075			
	6	076			
	7	077			
10	0	100			
	1	101			
	2	102			
	3	103			
	4	104			
	5	105			
	6	106			
	7	107			
11	0	110			
	1	111			
	2	112			
	3	113			
	4	114			
	5	115			
	6	116			
	7	117			
12	0	120			
	1	121			
	2	122			
	3	123			
	4	124			
	5	125			
	6	126			
	7	127			
13	0	130			
	1	131			
	2	132			
	3	133			
	4	134			
	5	135			
	6	136			
14	0	140			
	1	141			
	2	142			
	3	143			
	4	144			
	5	145			
	6	146			
	7	147			
15	0	150			
	1	151			
	2	152			
	3	153			
	4	154			
	5	155			
	6	156			
	7	157			

Table 5-6. Summary of I/O References for 8 Circuit Modules - Continued

Module No.	Circuit No.	Reference	In/Out	Device Name	Wire No.
70	0	700			
	1	701			
	2	702			
	3	703			
	4	<b>704</b>			
	5	705			
	6	706			
	7	707			
71	0	710			
	1	711			
	2	712			
	3	713			
	4	714			
	5	715			
	6	716			
	7	717			
72	0	720			
	1	721			
	2	722			
	3	723			
	4	724			
	5	725			
	6	726			
	7	727			
73	0	730			
	1	731			
	2	732			
	3	733			
	4	734			
	5	735			
	6	736			
	7	737			
74	0	740			
	1	741			
	2	742			
	3	743			
	4	744			
	5	745			
	6	746			
	7	747			
75	0	750			
	1	751			
	2	752			
	3	753			
	4	754			
	5	755			
	6	756			
	7	757			
76	0	<b>760</b>			
	1	761			
	2	762			
	3	763			
	4	764			
	5	765			
	6	766			
	7	767			

## Internal Coils

The internal coils are control relays that can be used to control logic in the Series One or Series One Plus user program. However, their status cannot be provided directly to the I/O section. Retentive relays are specialized internal coils that have their ON or OFF status retained during time periods that the PC is not operating such as during loss of AC power. They use dual operations (set and reset) similar to hard-wired latching relays. A switch (**S1**) on the CPU module must be set to the ON position in order for the internal coils, 340 to 377, to be retentive upon loss of power. If the switch is not set properly, these coils will function as non-retentive coils.

## Use of the Special Function Coils

Internal coils 374 through 377 are special purpose coils, in that they always perform specific internal functions and also provide useful internal system status. Coil 374 is a **first** scan reset or power-up indicator. Coil 375 provides a convenient way to program a 0.1 second clock pulse. **Coil 376 can be used to inhibit all hardware outputs, that is, outputs that are** connected to user devices (programmed with an OUT **XXX** instruction). Finally, coil 377 is an indicator of the operating status of the back-up battery for the CMOS memory devices. This coil could be used as a contact to **turn** on an output connected to a light, bell, etc. as a visual warning of a low battery.

## Shift Register References

A group of references (**400 to 577**) are used by the shift register. A total of 128 stages are possible and they can be used as one large shift register or a group of smaller shift registers (for example, 3 @ 30 stages and 2 @ 19 stages) as long as the total number of stages does not exceed 128. Additional details on all these functions will be provided later in this chapter.

## Timer and Counter References

Another group of **fixed** references (**600 to 677**) are used with timers and counters. Any mix of timers or counters totaling 64 can be used in a Series One or Series One Plus PC. Counters provide an additional function, in that they can also control a sequencer with up to 1000 steps. Counters are retentive upon power failure. References 600 to 673 can be used as **16-bit** data registers if they are not used as T/C references. T/C references 674-677 are reserved for Thumbwheel Interfaces or Timer/Counter Set Point use only.

## Data Registers

Data registers are available **only with** a Series One Plus PC. This is a group of sixty four **16-bit** registers which can be used for storage of numerical data. Data can be written to or read from the data registers for use by the data operation instructions. Additionally data stored in the **registers** can be used as Timer/Counter presets. The references to be assigned to these registers are 400 to 577. Note that these references are identical to the shift register references. However, shift registers operate on individual bits (**400, 401, 402, etc.**), while the data registers operate on bytes and are located in a totally **different area** of memory. A byte in the Series One Plus PC is a group of 8 consecutive bits and each register is 2 bytes in length. By using available T/C references, the total number of data registers references can be up **to 124**.

## Flexibility in Using References

Although references are assigned to specific functions as shown in Table 5.5, there is some flexibility in their use. I/O references not used by the hardware I/O section (for example, no module inserted into that slot or circuit not wired up) can be used as internal coils. However, if expansion is planned for the future, sufficient references to support that expansion should be reserved and not used in the current programming. Retentive coils can also be used as internal coils if their retentive function is not **required**. However, typically the unique value of retentive relays does not justify their sacrifice for a few internal coils. The shift register references can be used as a source for additional retentive or internal coils if they are not used for shift registers or sequencers.

## Operating Principles

Before discussing the details of programming and the use of references, some details on the internal operation of the CPU can be useful. The majority of Series One and Series One Plus applications can be solved and the programs developed without consideration of the internal structure of the CPU. However, some applications can be more efficiently solved if knowledge of the CPU operation is applied while the program is being developed. The basic operation of virtually all PCs is referred to as a scanning function. There are many hundreds of decisions to be performed in any program, and the CPU cannot do all of them simultaneously. Similar to any electronic processor, it performs its operations one at a time. However, the speed of performance internally, makes the external results appear to have all operations accomplished at once.

### Scanning

The term scan is a method of describing how the CPU performs its assigned tasks (see Figure 5.2). It begins at the **first** function, such as a relay contact, entered into the beginning of memory (address 0000). It proceeds sequentially through all memory addresses performing all functions entered by the user (for example, relay contacts, timers, latching relays, counters, sequencers, shift registers, etc.) until it reaches either the end of memory (address 1723) or the end of the program as entered by the user.

During the scan the logic sets or resets coils, according to the instructions entered into the logic program by the user. The status of these coils is immediately available to the next logic function. At the end of the logic scan, the CPU gets input data **from** the input modules and provides new data to output modules.

Next, the programmer, if connected, is serviced by making logic changes and/or updating its display. After servicing the programmer, the CPU performs a check of its internal hardware and resets the watchdog timer. The watchdog timer is a hardware timer set at 180 **msec** (milliseconds) to ensure that **memory or internal** circuit faults do not cause the CPU to enter **an** endless loop because of hardware failure. If a scan is not completed at least once every 180 **msec** (typical maximum scan is 60 msec), **the** hardware will shut the CPU down turning outputs OFF. Finally, any forced I/O **will** be entered. With successful completion of the internal checks, the CPU goes back to the start of the scan and continues its scanning sequence with the logic function entered at address **0000**. This repetitive scanning operation is performed from the time power is applied to the CPU until it is removed.

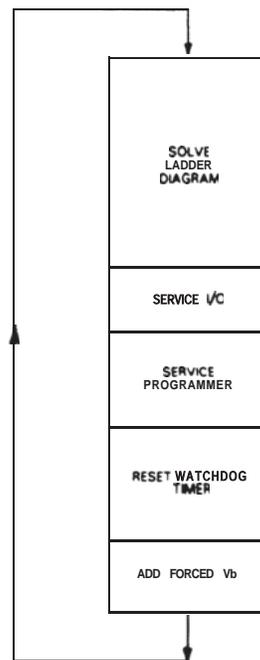
Total scan time includes user logic solution time plus overhead. The overhead time includes monitoring functions, I/O updates and housekeeping tasks. The typical scan times listed in the specifications in Chapter 1 assume that only basic instructions have been programmed (AND, OR, STR, NOT, etc.), they do not include overhead. The **overhead** time is about the same (4 to 5 **mSec**) for Series One and Series One Plus. This time must be added to the logic solution time to obtain the total real scan time.

GEK-90842

In order to achieve the maximum scan time, the monitor function should be disabled for both Series One and Series One Plus programs, since the monitor function adds about 2 **mSec** of overhead time when it is active. Also the overhead of a Series One Plus is increased slightly when communicating through the DCM, since both the Series One Plus CPU and DCM have been designed to provide a much higher throughput than the Series One **CPU/DCM** combination. Additionally, timer and counter instructions in the Series One Plus are slower than in the Series One because of the ability to use registers for preset values.

The scanning operation is very basic to PC operation and should not be overlooked. It provides a very useful verification of the CPU's reliability. For details on troubleshooting, refer to Chapter 7. The scanning operation also provides a fixed and definable sequence of logic decisions. Functions are solved in the order programmed. The results of one function (for example, coil, timer, shift register, etc.) are immediately available internally to the next logic element. In larger complex logic programs, internal races can be eliminated. The user does not need to be concerned with the time delay relays required to **pull in** or activate nor their variations (for example, a 4 pole vs 8 pole relay).

pc-s1-83-0004



**Figure 5-2. CPU Scanning Sequence**

## Programmer Functions

The basic programming **language** of the Series One and Series One Plus PCs is a simple relay ladder representation based upon standard Boolean functions (AND, OR, and NOT). Throughout the discussion of programming, examples will be used to illustrate the function described. The illustration of the Programmer (Figure 5.3) will be used as a reference for these examples and specific key sequences will

be provided so that the user can follow and demonstrate the example. A short-hand notation will also be provided that is recommended as a simple method to document your program. A complete discussion of the programmer is provided in Chapter 4. A brief overview of important keys is presented below to aid the user's understanding of the examples.

The keyboard has both upper and lower case functions similar to a typewriter. The numerical keys for most programming steps are accessed by first selecting the SHF (Shift) key to the upper right. Once depressed, the SHF LED is lit (display center right) and then the appropriate numerical digit(s) can be selected. The shift function is latched and will be released only by selection of the ENT (Enter) or CLR (Clear) key. The SHF key does not have to be held down. The display window in the upper left reflects either address or data information (not both) as the operation progresses. Typically, as keys are selected, their respective LEDs will light to display the function selected. Other important keys are as follows:

KEY	FUNCTION
AND	AND referenced status with previously entered logic (Series)
OR	OR referenced status with previously entered logic (Parallel)
STR	START new rung of a ladder diagram
NOT	NOT or invert (i.e. make normally closed) contact referenced
OUT	OUTPUT logic status to a coil, either output or internal
TMR	TIMER is selected as desired function
CNT	COUNTER is selected as desired function
SET	SET latching relay or shift register status
MCS	Master Control Start - begin control of master control relay
MCR	Master Control Reset - end control of master control relay
SR	Shift Register - establish range of shift register
RST	RESET latching relay or shift register status
F	FUNCTION - specifies a data operation (Series One Plus)
R	REGISTER - specifies a data register or group entry (Series One Plus)

a41 939

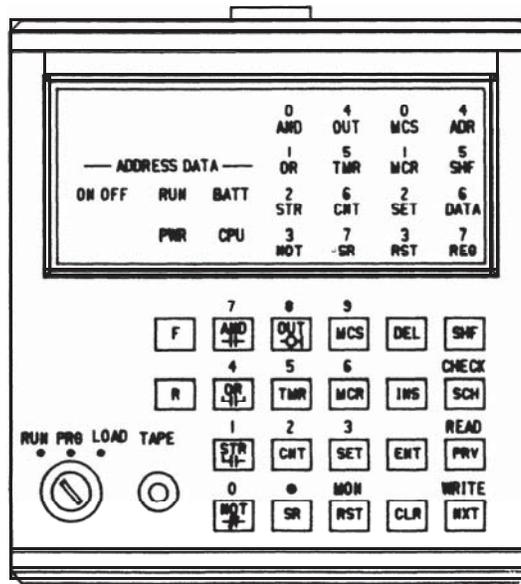


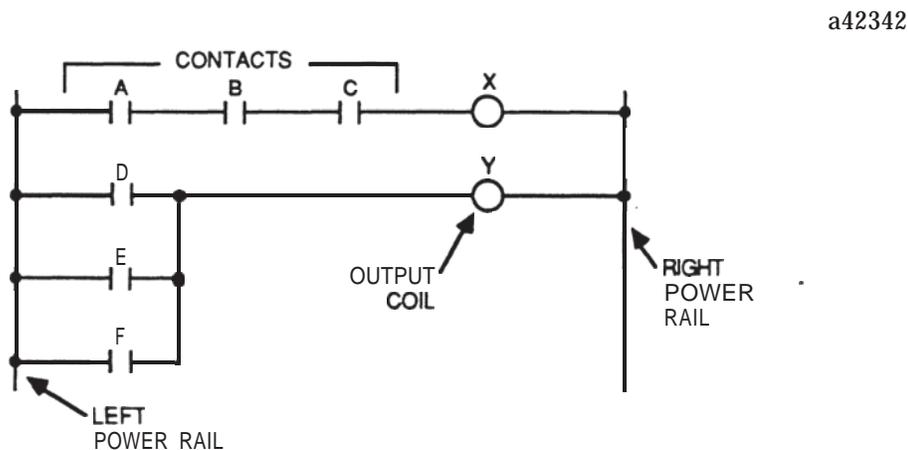
Figure 5-3. Programmer for Series One and Series One Plus

GEK-90842

### Basic Ladder Diagram Format

Figure 5.4 illustrates a typical ladder diagram. Between two vertical power rails to the extreme left and right, contacts are placed in horizontal strings or lines. Adjacent lines can be connected between contacts by a vertical line to allow logic to be solved in parallel. The horizontal strings (series) of contacts **are** equivalent to **ANDs**. For example line 1 can be described as: A and B and C must occur before coil X is energized; any one reference can prevent the coil from energizing. Similarly vertical contacts are in parallel and programmed as **ORs**. Thus, line 2 is described as: D or E or F will energize coil Y. Any one reference can by itself energize coil Y.

In the Series One and Series One Plus PCs, there is no internal limit on how many contacts can be **placed in series**, nor how many in parallel. However, as a practical limit for simple programming **and** system documentation, it is recommended that a horizontal string be limited to nine contacts and one coil; and a vertical array to seven parallel lines.



**Figure 5-4. Typical Ladder Diagram**

### Concept of Power Flow

One key feature of PCs is power flow. This is a conceptual flow of power used to visualize the operation of coils, timers, counters etc. Referring again to Figure 5.4, the left hand power rail can be envisioned as “hot” (connected to 115 V ac or +24 V dc) and the right as its associated “neutral” (connected to 115 V ac or -24 V dc). **The coils (e.g. X and Y) will be energized if there is a path for power flow from the left leg to the coil placing the full potential across the coil.** If there is no power (electron) flow, the coil will be de-energized (OFF).

Power **always flows from** the left towards the right and will pass through normally open contacts if their references **are** energized (ON) or **normally** closed with de-energized (OFF) references. Power flow is also allowed vertically either up or down between adjacent lines where **parallel connections are programmed**. However, power can NOT flow from right to left at any time through contacts or horizontal shunts. **This feature simplifies programming** and prevents undesired sneak **paths**. However, if hard wired **relays are replaced** by a Series One or Series One Plus PC, it is possible some adjustments may be necessary to the logic to either **simplify the programming** or **to add sneak paths** that may be a **basis for the relays functioning**.

## Unlimited References

Another difference between PC **programming** and hardwired relays is that any reference such as an input or coil can be used on relay contacts as often as necessary. Since references are merely a unique series of bits in a word of memory, they can be programmed wherever and whenever necessary. Now with PCs, there is no need to count relay contacts and try to limit individual references to four or less contacts; no need to plan to use form C relay configurations to squeeze two contacts **from** one pole. Any reference can be envisioned as a relay controlling many poles providing both normally open and normally closed contacts **that** operate without any time delay from pole 1 (closest to the coil) to pole 1000 (furthest away from coil).

---

 GEK-90842
 

---

## SECTION 2

# Basic Instructions

---

### Basic Instructions

This group can be considered as the basic building blocks for a relay ladder diagram. The instructions reference discrete bits that are to be part of an operation, that is, a conditional contact or the end of a rung, which could be an output or an internal relay.

### How to Begin Programming

The easiest way to begin feeling comfortable with **programming** a Series One or Series One Plus PC, is to enter one rung of logic with contacts in series that control a single coil. Physically connect input devices (pushbutton switches, limit switches, etc) and an output device (such as a lamp) to the corresponding terminals on an input and output module in the PC, turn the inputs on and off, and observe the result on the output device.

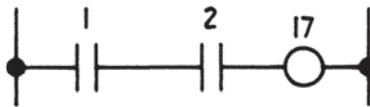
### Entering a Rung With Series Contacts

In the following program sequence, a rung of ladder logic is shown, then the keystroke sequence required for entering the logic. Notice that when entering a numerical sequence, the SHF key must be depressed before the number or number sequence. In the example, each keystroke is separated by a comma. Numbers with more than 1 digit are grouped together for ease of interpretation.

Turn the mode keyswitch to the PROG (**programming**) position in order to enter the logic.

Enter **two** normally open contacts (1 and 2) in series controlling the state of an output coil (17). Both contacts must be closed to tum on the output.

a40347



When input devices **wired** to inputs 1 and 2 are closed, output 17 will tum on. The operating state of the inputs and outputs, either ON or OFF, can be monitored by observing the **LEDs** on the Programmer's logic display that **correspond** to the input or output.

### Entering the Clear all Memory Sequence

After observing the operation of the series rung, the rung can be deleted by entering the clear all memory sequence, or the rung can remain in memory, with the next rung starting **at** the next memory address. The next memory address, assuming that **the** above rung was entered at address 0000, is **0003**.

If you wish to clear tie contents of memory and start the next rung at address 0000, enter the following key sequence.

**CLR, SHF, 348, DEL, NXT**

When the clear all memory sequence has been successfully entered, the rung of logic that had been entered will be cleared from memory. (Remember, whenever the clear all memory sequence is entered, **the entire contents of memory will be cleared**).

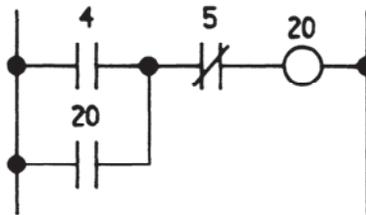
#### NOTE

Be sure that you want to clear the complete contents of memory when using this sequence, there is no recovery.

### Entering a Rung with Parallel Contacts

Continue the simple program entry by entering a normally open contact (4) in parallel with a second **normally open contact (20) that references the output coil (20)**. In series with these, add a normally closed contact (5), then the output coil.

a40347

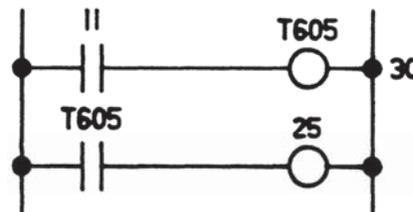


If an input device, such as a limit switch is wired to input 4, and the limit switch is closed, the output would turn on. The normally open contact referencing the output would close and become a seal contact, which would maintain the output in the on state, even if the limit switch wired to input 4 were then opened. The output could be turned off, in this example, by momentarily closing a switch connected to the closed contact, reference 5. Since this is a normally closed contact, it would open when the switch is depressed, and the output would turn off. This logic is commonly used as a motor starter.

### Entering a Simple Timer Rung

Next, enter two rungs of logic, the first ending with a timer, having a preset value of 30 **seconds**. The second **rung** has an output controlled by the state of the timer coil. When input 11 is closed, the timer begins **timing** up from 0 towards the preset value of 30 seconds. When the accumulated value of the preset reaches 30, the timer coil will turn on, contact **T605** will close, and output 25 will turn on.

a40347

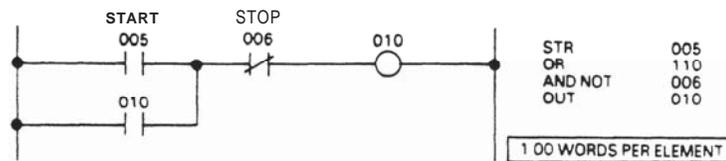


GEK-90842

**Basic Relay Logic (Motor Starter)**

With this background on the basics of programming a Series One and Series One Plus PC, Figure 5.5 provides a simple example to illustrate relay ladder programming. The references used were selected for illustrative purposes and can be adjusted as necessary to meet your application needs. This is a simple motor starter and seal circuit. An input module is assumed to be installed next to the CPU (005 = start pushbutton input, and 006 = stop pushbutton). Adjacent to the input module is assumed to be an output module (010 = output to motor starter). The program does not care whether these input and output modules are 115 V ac or 24 V dc. The programmer keys to be depressed to enter this logic are also shown in the figure. The start key begins a new rung of logic; this example and most that follow assume an empty memory so the example begins at address 0000.

pc-s1-83-0006



KEY	ON LED'S	DISPLAY	KEY	ON LED'S	DISPLAY
STR	STR SHF		AND	AND	
SHF	STR		NOT	AND. NOT	
5	STR. SHF	5	SHF	AND. NOT. SHF	
ENT	ADR	0 0 0 1	6	AND. NOT. SHF	6
OR	OR		ENT	ADR	0 0 0 3
SHF	OR SHF		OUT	OUT	
1	OR. SHF	1	SHF SHF	OUT. SHF	
0	OR. SHF	10	1	OUT. SHF	1
ENT	ADR	0 0 0 2	0	OUT. SHF	10
			ENT	ADR	0 0 0 4

**Figure 5-5. Sample Relay Logic (Motor Starter)**

**Motor Starter Logic Description**

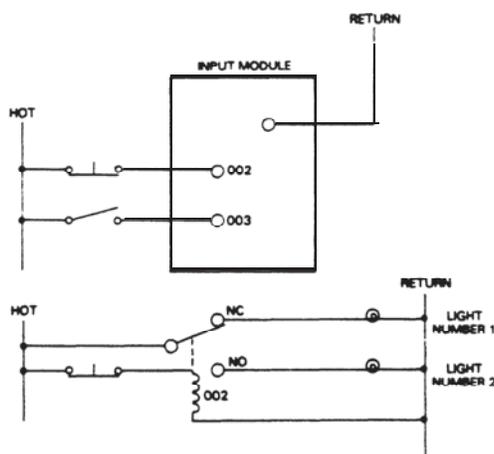
The first reference is entered (STR, SHF, 5, ENT) and a normally open contact is established since the NOT key was not depressed. Referring to Figure 5.5, the next function moving from the left (contact 005) to the right along the top line of the logic rung, is a parallel contact referenced as 010. Parallel logic is represented by the OR function, which is selected followed by the desired reference (10) and the Enter key to load the memory. Again moving to the right of the top line, the next contact is in series with the logic completed so far. Thus the ANTI key is selected (for series logic). Since this contact is normally closed, the NOT key is also selected followed by the reference 6 and the Enter key.

The final element of this logic is the coil or output of the rung. Selecting the Output key does not ensure a “real world” output will be generated; it also must refer to an I/O reference and have an output module inserted in the proper slot. To establish the rung output coil, the output key is selected followed by the Shift key, the reference (10) and the Enter key. Outputs can be paralleled by entering additional OUT functions immediately after the initial one.

### Normally Closed Input

After completing the logic of Figure 5.5, several features of programming should be noticed. First, all elements of the program are entered directly into CMOS memory as the Enter (**ENT**) key is depressed. Thus, if power should be interrupted while a program is being entered or altered, it will be totally saved except possibly for the one element being worked on. Second, there is feedback on the programmer **for all entries**. Something happens (LED ON, display changes, error codes appear, etc) for each key selected. Third, the logic is set up for inputs that are **wired** normally open. If an input is wired normally closed, an adjustment must be made to the symbol (NO or NC) used for that input. Referring to Figure 5.6, inputs sense voltage at their field terminals. They can not detect if a voltage is from a normally open limit switch that is depressed or from a **normally** closed pushbutton that is not depressed. Cover up the two field devices wired to inputs 002 or 003. If you were the input module, and both applied voltage to the field terminals, could you tell which was wired normally closed and which normally open? Because of this, the programmer or system designer must compensate.

pc-s1-83-0007



**Figure 5-6. Example of Normally Closed Inputs**

In a previous discussion, it was stated that inputs are like relays with many poles and you are selecting which type of pole is to be used when you program. In the lower half of Figure 5.6, coil 002 is shown in the de-energized position; however it is wired to a normally closed pushbutton. When you want power to flow if the pushbutton is **NOT** depressed, which contact type do you choose, NO or NC? The answer is normally open since coil 002 is always energized except when the pushbutton is **depressed**. **Therefore, in programming** with the Series One or Series One Plus PCs, a **normally** open contact is appropriate. If the stop button of Figure 5.6 were **wired normally closed**, its internal logic should be normally open to compensate] However, this concept does not normally cause problems for the programmer. **Since PCs are reprogrammable**, if an error is made and discovered when the system is exercised, it is easy to correct



**Detailed Example of AND STR, OR STR**

The push-down stack can be used for several groups of contacts, either in series or in parallel, and can be reused many times in the user's logic. As mentioned previously, the stack has eight levels or storage locations where an ON/OFF (1/0) condition is stored.

PUSH-DOWN STACK	
ON (1)	Location 1
ON (1)	Location 2
OFF (0)	Location 3
ON (1)	Location 4
OFF (0)	Location 5
OFF (0)	Location 6
ON (1)	Location 7
OFF (0)	Location 8

**Figure 5-8. Push-Down Stack Storage Locations**

**Description of Operation**

There are four basic rules that govern the operation of the push-down stack and they are explained in the following examples.

**Rule Number 1:** A STR X instruction pushes the contents of each position in the stack down one location. The contents of location number 1 are pushed down to location number 2, the contents of location number 2 are pushed down to location number 3, and so forth. The logical status of reference X is then moved into location number 1 of the stack.

**Rule Number 2:** The AND Y/OR Y instruction logically ANDs/ORs the status of reference Y (ON/OFF) with the ON/OFF status of location number 1 in the stack and writes the result back into location 1.

AND (SERIES)			OR (PARALLEL)		
	BEFORE LOCATION NUMBER 1	AFTER LOCATION NUMBER 1		BEFORE LOCATION NUMBER 1	AFTER LOCATION NUMBER 1
Y			Y		
OFF	OFF	OFF	OFF	OFF	OFF
ON	OFF	OFF	ON	OFF	ON
OFF	ON	OFF	OFF	ON	ON
ON	ON	ON	ON	ON	ON

**Figure 5-9. Push-Down Stack Logical Operations**

GEK-90842

**Rule Number 3:** The AND STR or OR STR instructions logically AND or OR location number 1 in the stack with location number 2 and writes the result into location number 1. In other words, these instructions put one group of contacts in series or parallel with another group of contacts. This instruction also moves the contents of positions 3 through 8 up one location.

**Rule Number 4:** An OUT Y instruction sets Y to reflect the status of location number 1 in the stack. If the status of location number 1 is ON (1), output Y will be turned off. If the status of location number 1 is OFF (0), output Y will be turned on.

pc-s1-83-0055

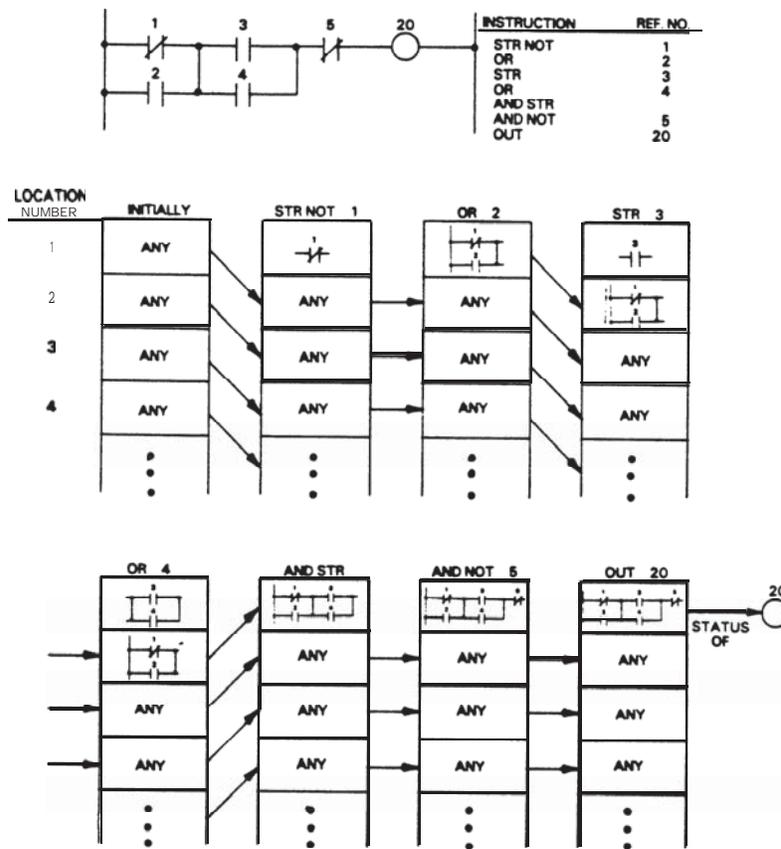


Figure 540. AND STR/OR STR Example Number 1

pc-s1-83-0056

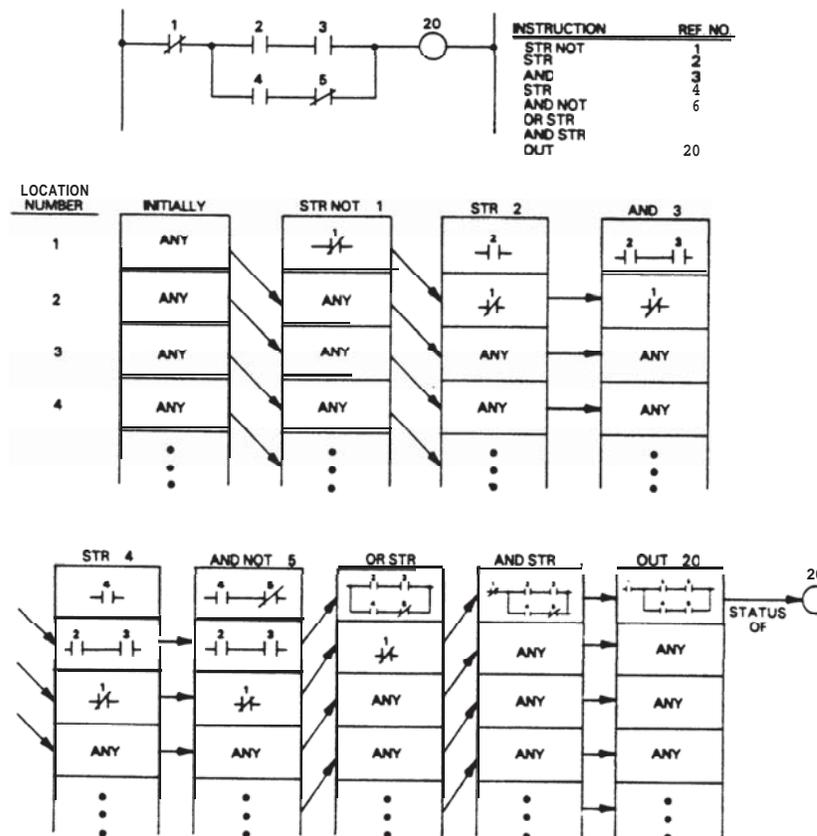


Figure 5-11. AND STR/OR STR Example Number 2

### Use of Retentive Coils as Latches

All coils referenced so far have not been retentive upon power failure. Thus if a non-retentive coil is ON when power is removed **from** the CPU, it will be OFF upon reapplication of power (if Switch 1 on the CPU module is **ON**). **There are 28** special coil references (340-373) that **are** retained upon power failure. Thus if they were ON prior to loss of power, they will be ON when power is restored; they will be OFF while power is OFF since no scanning is being performed. In addition to these 28 references, latches can be built using shift register references (400-577) that are not being used to perform shift register functions. Either retentive coils (340-373) or shift registers (400-577) can be used to control contacts, both normally open and normally closed wherever required. **This** special retentive function is derived **from** their use as coils to complete or store the results of relay logic rungs.

### Programming a Latched Relay

Latches require two separate functions or inputs similar to hard-wired relays, a **SET (turn ON)** and a **RESET (turn OFF)**. **These functions are commonly** referred to as latch and unlatch. If both are active, the last one in the scan will be controlling the state of the coil. To turn on a latch, the function SET

GEK-90842

followed by the retentive coil reference (for example, 340-373 or 400-577) is used instead of OUT, plus the coil number at the end of a logic rung. Once power flows to that coil, it will be energized (turned ON) and remain ON even if power flow to it should be interrupted.

To turn a latch off (de-energize) separate relay logic should be built ending in a RST (Reset) function and the [same] coil number. In many applications, it is useful to allow the system to clear (turn OFF) coils when power fails so that an orderly restart is possible when power is restored, whether that is 2 seconds or several hours, or several days later. Other applications require the control system to remember the status of key items such as part position, operations completed, elevators up or down, etc. The Series One and Series One Plus are provided with both standard and retentive coils to allow the user to tailor the program to fit their requirements.

Figure 5.12 illustrates the programming of retentive (latched) relays. Coil 340 is used only as an example. The logic to either set or reset a retentive coil can be of any convenient size similar to standard coil logic. When power flows through the top rung to the SET coil 340 function, it will be energized and remain ON even if power flow is removed. Anytime the power flows through the lower rung (regardless of path), coil 340 will be turned OFF. Since the set coil 340 is programmed before the reset, it is placed in earlier memory locations and the reset function will have overriding control if they are both energized.

pc-s1-83-0011

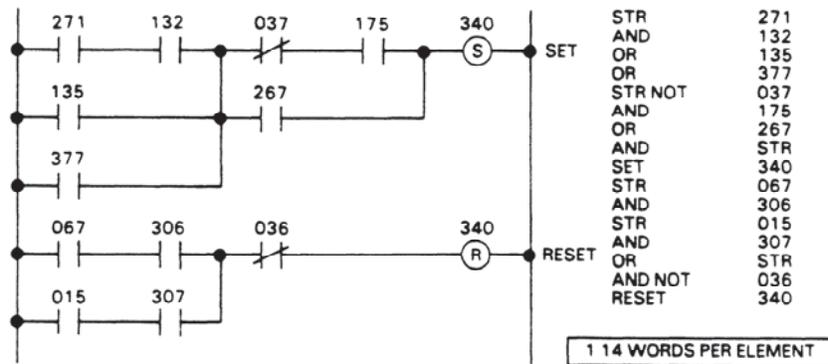


Figure 5-12. Example of Latching Logic

### Master Control Relay Functional Description

To control large quantities of coils on a supervisory basis, a Master Control relay function is provided. This function allows an efficient method of programming to turn off large quantities of coils in case some major permissive is not satisfied. One approach would be to incorporate a permissive contact in each rung controlling one of the coils. This may require many contacts and inefficient use of memory. Another method is to bracket those rungs over which master control is desired; this is similar to quotation marks around a direct statement (one at each end to define exactly where the statement starts and where it ends).

The permissive logic is first built starting to the left of the ladder diagram adjacent to the "hot" power rail. Then an MCS (Master Control Start) function is entered, followed by the logic it is to control, and ending with an MCR (Master Control Reset) function. The MCS and MCR are like quotations around the logic to be controlled. When power flows to the MCS, the logic enclosed will operate normally. If

there is no power flow to MCS, the coils will be forced to the OFF state regardless of the internal logic conditions, timers will be reset to zero, and counters will stop (frozen) but not reset. The amount of logic controlled by the Master Control relay function is limited only by the memory provided.

pc-s1-83-0012

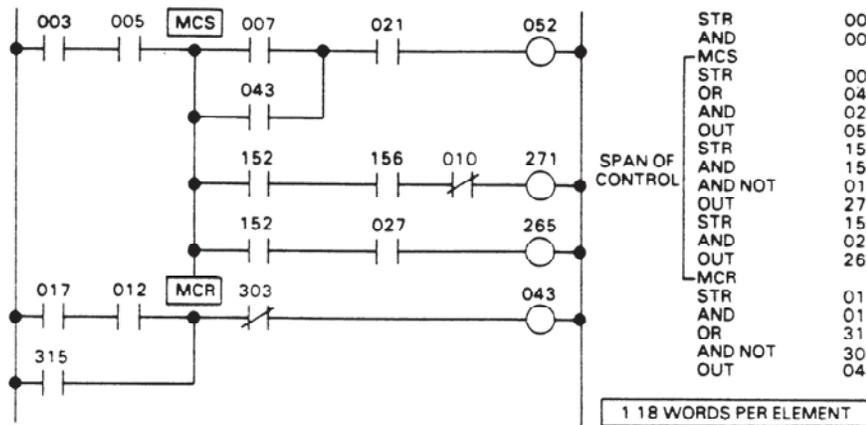


Figure 5-13. Example of Master Control Relay Logic

### Programming a Master Control Relay Function

Figure 5.13 illustrates the use of one pair of MCS/MCR functions. In many cases it is convenient to view the MCS function as defining a sub-left power rail whose connection to the main power rail is dependent upon some relay logic. The conditional logic (I003 - I125 in this example) is built first using the normal relay logic. The MCS function is entered without any reference to create the beginning point of the control. Logic is built using normal functions within the control area. Note the duplication of contact 152, which cost only one memory word, there is no need to count or limit the quantity of relay contacts used in PCs. To end the master control, the MCR function is entered returning control to the next power rail to the left and conventional logic can now be built. Unless both references 003 !and^ 005 are energized (ON) in this example, coils 052, 271, and 265 will be OFF. When these contacts are passing power, coils 052, 271, and 265 will respond to their normal logic.

### Multiple Master Control Relay Functions

Multiple master control functions are possible in any logic program. They can be embedded within the scope of the first MCS and the last MCR as necessary, as illustrated in Figure 5.14. The first group of coils under the Master Control of references 010 [or] 011 is 204-213 which includes two smaller groups (207-210 and 212-213). Group 207 and 210 are under the Master Control of 010 or 011 and 015, while group 212 and 213 are controlled by 010 or 011 and 022. Note the requirement for two successive MCRs to end both the group 212-213 and the larger group 204-213. To operate correctly there must be an equal number of MCS and MCR functions in your program.

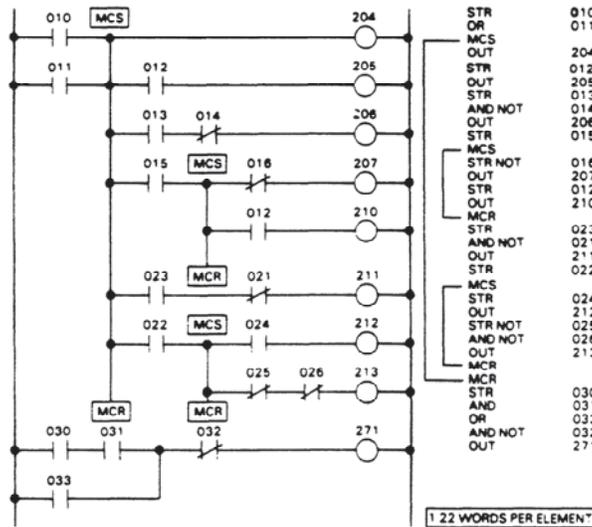


Figure 5-14. Multiple Master Control Relay Logic

**Disabling of Outputs**

In addition to Master Control functions, all outputs can be affected by an internal special function coil referenced as 376. When this coil is energized by user logic, all outputs in the hardware I/O structure will be turned OFF. Internal coils and statuses will still operate, but only internally. The only exception is to coils that are programmed as SET OUT functions rather than the OUT function. These will still be able to provide their normal output status (ON or OFF) to the I/O section.

Figure 5.15 illustrates the use of coil 376. In this example, when either reference 003 or 056 (inputs, outputs, or internal depending upon hardware I/O configuration) is energized, coil 376 will also be energized. This will cause all coils programmed with the OUT function, such as 010, to be turned OFF at their output module, although they will function normally internally. However, coils driven by the SET OUT function, such as 011, will not be affected by the status of coil 376. Table 5.7 summarizes the effect of coil 376 on the OUT and SET OUT functions.

**NOTE**

It is not recommended that the Output Disabler coil (376) be used with retentive coils (latches).

**Table 5-7. Effect of Coil 376 (Output Disabler)**

Internal Status	Coil 376 OFF		Coil 376 ON	
	Out	Set Out	Out	Set Out
OFF	OFF	OFF	OFF	OFF
ON	ON	ON	OFF	ON

pc-s1-83-0014

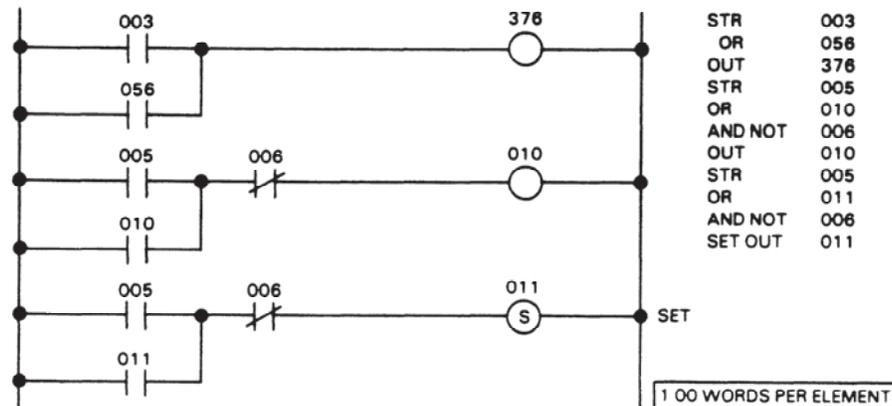


Figure 5-15. Sample Coil Disabler Logic

### Timer and Counter Functional Description

In addition to relay logic, there are a wide variety of functions that are standard with the Series One and Series One Plus. Timing and Counting are two of the most commonly used non-relay functions. These functions are special coils using references 600-677. Up to 64 total timers and/or counters can be built within the Series One or Series One Plus. They end a rung of logic similar to using an OUT function for relay logic. Unlimited contacts, both normally closed (timed OFF delay) and normally open (timed ON delay) can be referenced to any timer/counter coil. A preset value is used to indicate how many seconds, tenths of seconds, or counts, the function should delay before energizing its coil. When the coil is energized, time or counts continue to be recorded. All presets can be up to four decimal digits (0001 to 9999), which allow timers to be set from 0.1 to 999.9 seconds, and counters from 1 to 9999 counts. Timers can be set from 0.01 to 99.99 seconds with the Series One Plus 3.7K PC only. Timers are set to a .01 second duration by setting output reference 770 (SET 770). Caution must be used when doing this, since reference 770 being reset in the same program would cause the timers to revert back to .1 second duration.

### Special Timer/Counter References

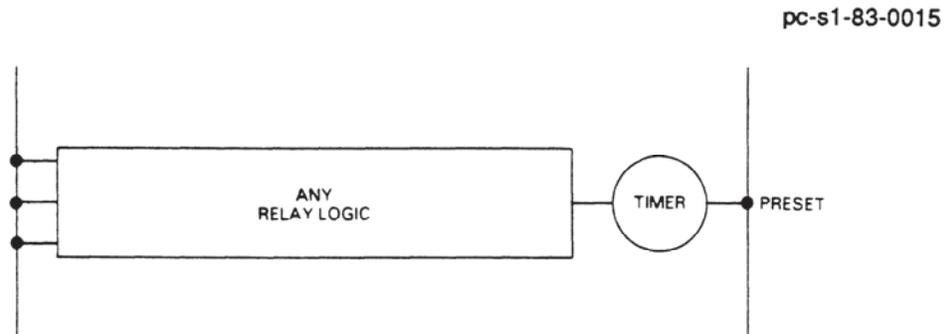
The four references 674-677 obtain their presets from external thumbwheels via the thumbwheel interface. Additionally, a Timer/Counter Setpoint unit can be used with the Series One Plus to enter the preset values. When a preset value is selected with the Timer/Counter Setpoint unit, the values are simultaneously entered into 8-bit registers, (two registers for each 4-digit BCD value). The registers can then be used to specify the preset value for any Timer or Counter. It should be noted that only 1 Thumbwheel Interface unit or 1 Timer/Counter Setpoint unit can be used with a series One Plus PC, not both at the same time. These four references cannot use internal presets; however, they will count up to 9999 and can be used to drive sequencers without the thumbwheel interface.

### Programming Timers

Timers (Figure 5.16) can be programmed in either seconds or tenths of seconds (or hundredths of seconds with a Series One Plus 3.7K PC) and require only one line of logic to control their operation. When power flows to the timer coil, it will begin recording time at the rate of one count every 0.1

GEK-90842

seconds or .01 seconds as applicable. The timer starts at 0000.0 and records time, towards the preset value. When the preset value is reached, an event will happen according to what has been programmed. Also, when the timer reaches the preset value, the timer will continue timing up, until reset by an interruption of power flow through its conditional contacts, or until it reaches its maximum value of 999.9 or 99.99, as applicable at which time it will stop. If power flow to a timer is interrupted or if the CPU should stop operating, the timer will be reset to zero. The basic clock driving all timers derives its accuracy from a crystal within the CPU, it does not depend upon the AC power line frequency.



**Figure 5-16. Sample of Timer Logic**

A timer can be programmed to be self resetting, by programming a closed contact referencing the timer coil. If programmed in this manner, when the preset value is reached, the timer coil will turn on, the closed contact referencing the coil will open and the timer will reset to 000.0 and start timing up again.

Figure 5.17 illustrates the programming of various types of timers. The relay logic is built in the normal manner up to the coil selection. The timer (TMR) function is selected along with a timer/counter reference (600-677). Each timer or counter must have its own unique coil to operate properly. After the timer is identified, the preset is entered.

In this example, coil 41 will be OFF until 15 seconds after timer 601 is energized. It will remain ON until either the power flow to 601 is interrupted or the CPU goes through a power up operation. Coil 42 is inverted from 41 in that it stays ON until 15 seconds after power flows continuously to the timer 601, when it goes OFF. Timer 602 delays 5.5 seconds after reference 27 is energized before it turns its coil ON, also energizing coil 43.

### Specifying Timer Preset Values

The Timer preset value can be either a fixed value entered after the T/C reference, the BCD value entered with the Timer/Counter Setpoint unit for the special Timer/Counter references (674-677), or the value in the registers that are assigned to the special Timer/Counter references. These registers are 564 and 565 for T/C reference 674; 566 and 567 for T/C reference 675; 570 and 571 for T/C reference 676; and 572 and 573 for T/C reference 677.

The preset value information is also valid for Counters.

a40016

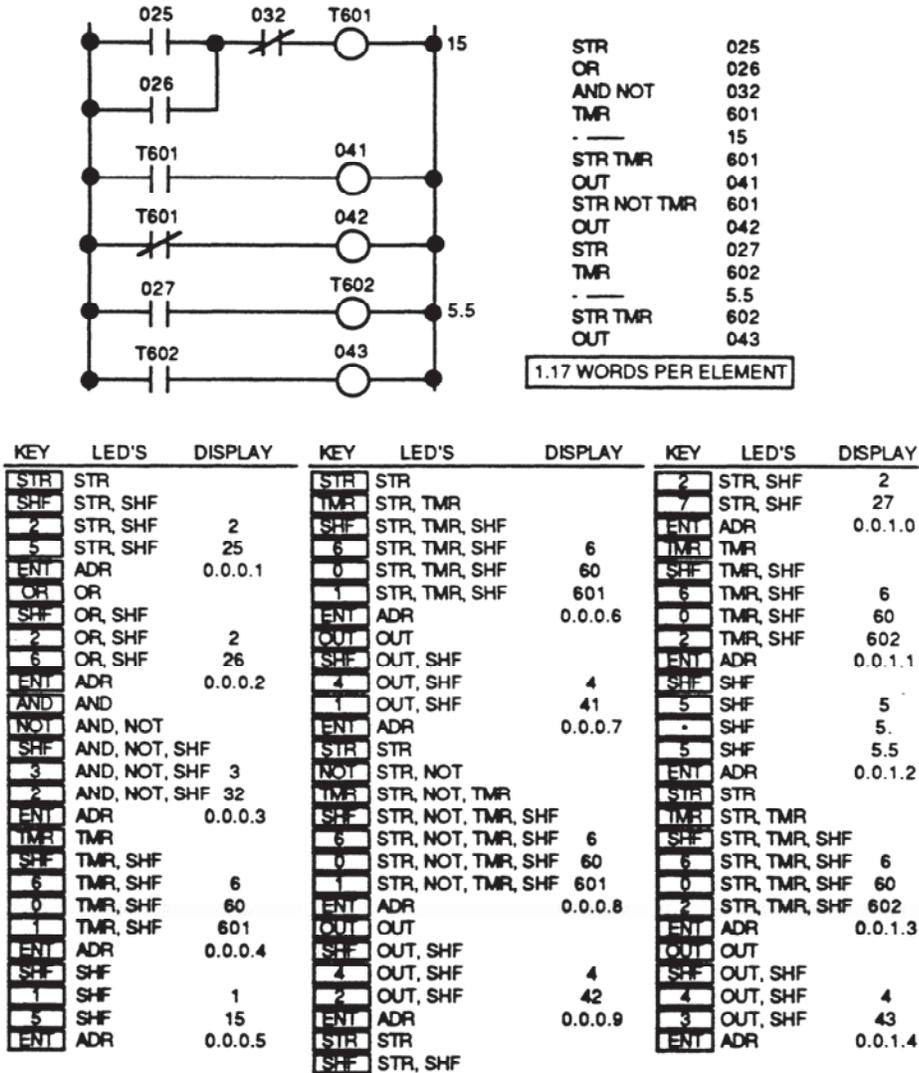


Figure 5-17. Example of Timer Logic

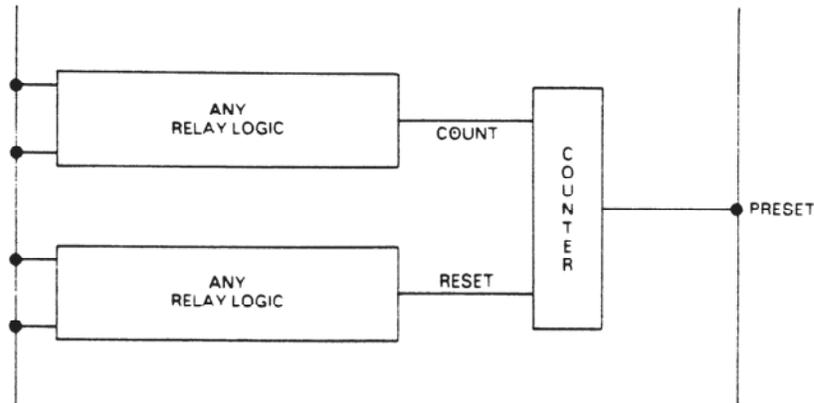
**Programming Counters**

Counters (Figure 5.18) operate similar to timers except that they require two rungs of relay logic to control their operation. The upper rung controls when the counter is incremented. When this rung goes from no power-flow to power-flow (OFF to ON), the counter is incremented by one. To cause another count to be recorded, power-flow must be interrupted and another OFF to ON transition must occur. All counters count up, starting at zero, towards a preset value. The ability to detect transitions and record counts is built into the counter function and requires no further programming by the user. The lower rung of logic (second STR function) controls the reset of the counter function. Whenever this rung supplies power flow to the counter, the counter will be reset to zero. If both rungs supply power flow, no

GEK-90842

counts are recorded and the counter is forced to zero. All counters are retentive upon loss of CPU power.

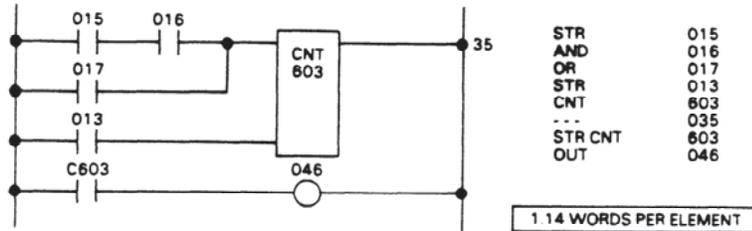
pc-s1-83-0017



**Figure 5-18. Example of Counter Logic**

Figure 5.19 illustrates the operation of counters. Again, the relay logic driving the counters is built as separate rungs using normal relay programming techniques. Counter 603 will count (increment) whenever references 015 and 016 are both ON or reference 017 is ON. If one parallel path is energized while the other is also ON, no additional count is recorded. When the count reaches 35 (its preset), coil 603 is energized, also energizing coil 46. Counting will continue beyond the preset value, and will continue counting until the counter is reset. Whenever reference 013 is energized, regardless of the count or top rung status or coil state, counter 603 will be reset to zero and held at that value until reference 013 is de-energized.

pc-s1-83-0018



KEY	ON LED'S	DISPLAY	KEY	ON LED'S	DISPLAY	KEY	ON LED'S	DISPLAY
STR	STR		STR	STR		STR	STR	
SHF	STR, SHF		SHF	STR, SHF		CNT	STR, CNT	
1	STR, SHF	1	1	STR, SHF	1	SHF	STR, CNT, SHF	
5	STR, SHF	15	3	STR, SHF	13	6	STR, CNT, SHF	6
ENT	ADR	0 0 0 1	ENT	ADR	0 0 0 4	0	STR, CNT, SHF	60
AND	AND		CNT	CNT		3	STR, CNT, SHF	603
SHF	AND, SHF		SHF	CNT, SHF		ENT	ADR	0 0 0 7
1	AND, SHF	1	6	CNT, SHF	6	OUT	OUT	
6	AND, SHF	16	0	CNT, SHF	60	SHF	OUT, SHF	
ENT	ADR	0 0 0 2	3	CNT, SHF	603	4	OUT, SHF	4
OR	OR		ENT	ADR	0 0 0 5	6	OUT, SHF	46
SHF	OR, SHF		SHF	SHF		ENT	ADR	0 0 0 8
1	OR, SHF	1	3	SHF	3			
7	OR, SHF	17	5	SHF	35			
ENT	ADR	0 0 0 3	ENT	ADR	0 0 0 6			

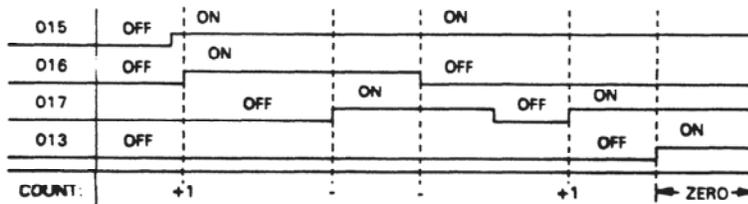


Figure 5-19. Example of Counters

Extending the Timer and Counter Range

Timers and counters can be connected in series or cascaded to extend their range beyond four digits. Figure 5.20 illustrates several techniques to extend preset ranges by utilizing multiple timer/counter

GEK-90842

functions. Timer 611 will record time as soon as references 206 and 225 are energized. After 800 seconds it energizes its coil and stops recording time. However, timer 612 starts as soon as 611 reaches its preset and continues for another 950 seconds. This is a total of 1750 seconds from the time T611 started. Whenever reference 206 or 225 is de-energized, timer 611 is reset to zero, de-energizing coil 611, and also resetting timer 612 (if necessary).

pc-s1-83-0019

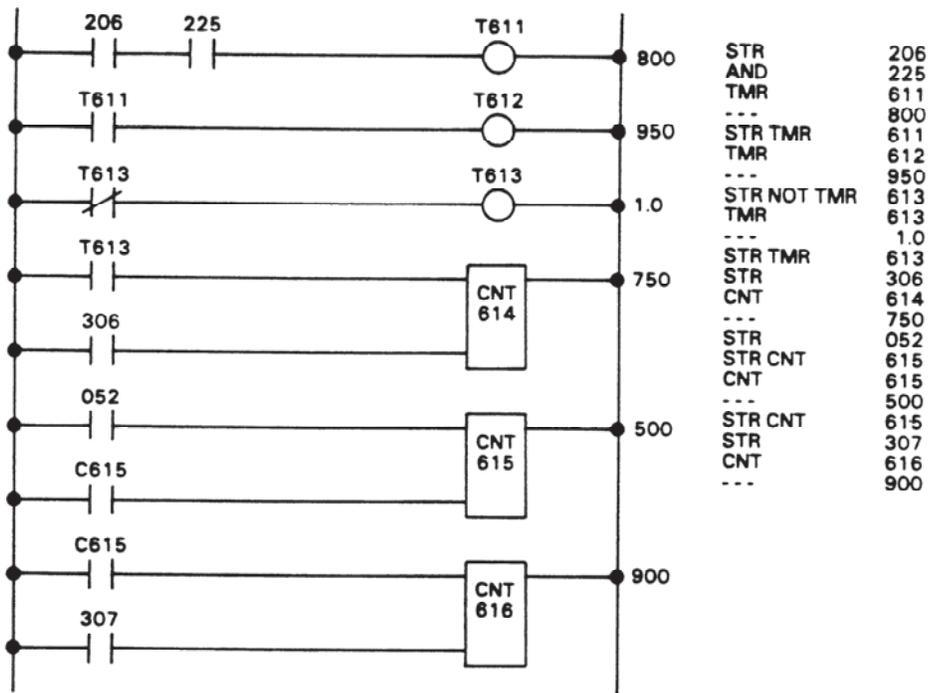


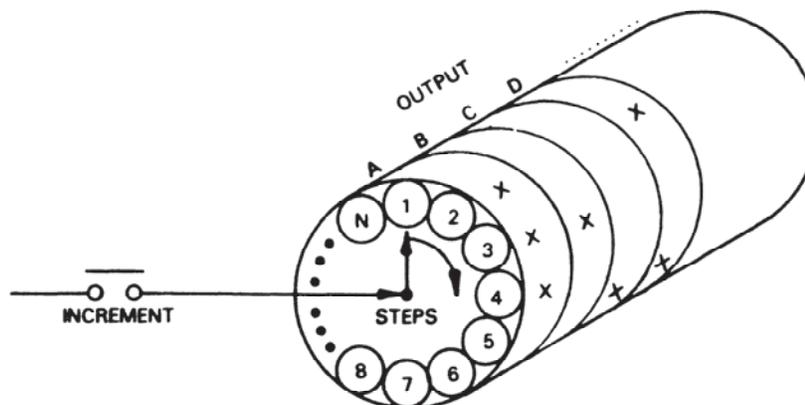
Figure 5-20. Example of Extended Counters

Another pair of functions are timer 613 and counter 614. Timer 613 is a self resetting timer with a preset of one second (10 tenths). It produces outputs each one scan long, every second from the time the CPU starts until it is turned OFF. Counter 614 counts these pulses up to 750 of them (750 seconds) until reset by reference 106. Since counters are retentive, using timer 613 to produce a one second clock that is counted by 614, results in a retentive timer. A similar result occurs if the top rung of counter 614 were referenced to internal coil 375 in lieu of 613. The third example in Figure 5.20 is counters 615 and 616. Counter 615 counts reference 52, up to 500, and then resets itself. Counter 616 records how many of these groups of 500 counts occur. Since counter 616 has a preset of 900, its coil will be energized only after 450,000 transitions of reference 52. At anytime, counter 615 has a representation of the small number of counts (0-499 or least significant portion) and counter 616 the larger values representing multiples of 500 (500-450,XXX) or most significant portion. Application techniques shown in Chapter 8 illustrate how a low order preset of 1000 can be obtained to produce a more conventional double precision counter.

## Sequencer Operation

All Series One and Series One Plus counters have one other very powerful feature which is applicable to all counters. They all can be programmed to act as sequencers. Each counter can control a 1000 step sequencer, which allows any Series One or Series One Plus PC to have sixty four 1000 step sequencers, each similar to a stepping switch or drum sequencer. Figure 5.21 illustrates the operation of one of these sequencers. Each sequencer moves from one position to the next as directed by some user defined signal shown here as a pushbutton. Counters move or are incremented once each time a count is recorded. Each sequencer starts at zero or home position when the counter is reset, and progresses through its steps one at a time in numerical order. At any time the position of the sequencer can be determined by the current count value contained in its storage. Although any sequencer can have up to 1000 steps, many applications are solved with a smaller number. To limit a sequencer to less than 1000 steps, the preset on the counter is used. As usual, each step of the sequencer can control any number of contacts.

pc-s1-83-0020



**Figure 5-21. Illustration of Sequencer Operation**

To illustrate the capabilities of the sequencer, refer to the example in Figure 5.22. The sequence map in this example is in the form of a timing diagram that has an overall cycle of 45 seconds. This time is subdivided into increments of 5 seconds per the requirements of this application. There are six outputs to be controlled during the sequence with the desired ON periods shown by horizontal black lines. Outputs are assigned and step numbers (starting at zero) are entered for each time period. To drive the sequencer (Figure 5.23), a timer (600) is entered to reset itself each 5 seconds. Input 030 is used to control the sequencer, which will not be incremented except when this input is energized. When energized, timer 600 will produce one pulse every 5 seconds. Counter 601 increments its count once each pulse, thus stepping the sequencer. Input 031 will reset the sequencer to home (zero) whenever it is energized, regardless of the count or position of the sequencer. The counter will go up to eight and then stop waiting for a reset signal.

GEK-90842

pc-s1-83-0021

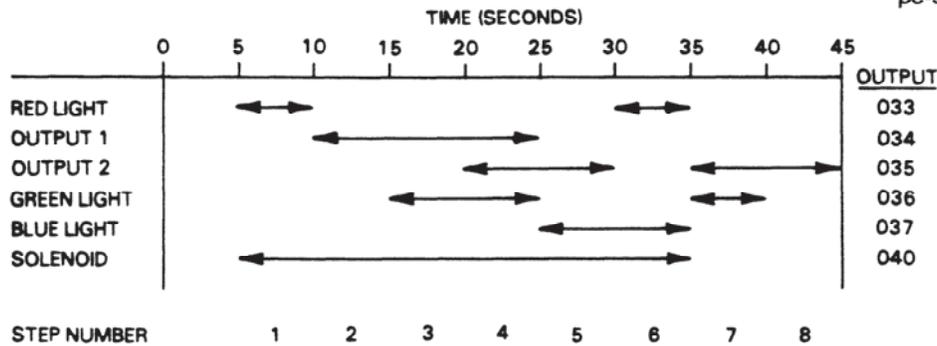


Figure 5-22. Example of Sequencer

The outputs from the sequencer are programmed using relay logic. However, the reference is first to the counter (601) controlling the sequencer and then to the current value that is going to control that contact. For example if a reference to counter 601 is desired that will pass power only when that counter has a current count of exactly one, the value 1 is entered after the reference to 601 on a normally open contact.

In this example, output 033 (red light) will be energized during 5 second intervals each for steps 0, 1, and 6. Similarly, output 034 (Output 1) will be energized at step 2, sealed and held until the beginning of step 5 (end of step 4). The other outputs are controlled using similar logic with parallel contacts or seal-in circuits.

If an output is ON for more steps than it is OFF, normally closed contacts would probably simplify your logic. Counters can also be used that are driven independently and not with a time base.

### Referencing Sequencer Contacts with a Series One Plus PC

When programming a Counter or Shift Register in a Series One Plus or Series One Plus 3.7K PC that has the IC610CPU105 CPU installed, a Drum Sequencer contact reference should not be used as the first contact (with a STR or STR NOT instruction) in the COUNT or RESET rung of a counter or the DATA, CLOCK, or RESET rung of a Shift Register. If a program using the step described above was entered into a Series One PC having an IC610CPU101 CPU (this would work correctly) and written to Tape, then later transferred to a Series One Plus PC with an IC610CPU105 CPU or a Series One Plus 3.7K PC with an IC610CPU106 CPU, the program would not execute properly.

Do not reference a Drum Sequencer step as the first contact in the rungs as described above. If a Drum Sequencer step is required as the first contact reference for a rung in a Counter or Shift Register, a programming technique that would work properly is to use the required Drum Sequencer step to turn on an internal coil in a separate rung, then reference that coil as the first contact in the Counter or Shift Register rung.

Printer Interface Unit  
CROSS REFERENCE PRINTOUT

OUTPUTS USED TABLE V2. 2

SERIES ONE, ONE PLUS/SR-20, SR-21

PAGE 0007

SHIFT REGISTER

400:	410: *	420: *	430: *	440:	450:	460:	470:
401:	411: *	421: *	431: *	441:	451:	461:	471:
402:	412: *	422: *	432: *	442:	452:	462:	472:
403:	413: *	423: *	433: *	443:	453:	463:	473:
404:	414: *	424: *	434: *	444:	454:	464:	474:
405:	415: *	425: *	435: *	445:	455:	465:	475:
406:	416: *	426: *	436:	446:	456:	466:	476:
407:	417: *	427: *	437:	447:	457:	467:	477:

500:	510:	520:	530:	540:	550:	560:	570:
501:	511:	521:	531:	541:	551:	561:	571:
502:	512:	522:	532:	542:	552:	562:	572:
503:	513:	523:	533:	543:	553:	563:	573:
504:	514:	524:	534:	544:	554:	564:	574:
505:	515:	525:	535:	545:	555:	565:	575:
506:	516:	526:	536:	546:	556:	566:	576:
507:	517:	527:	537:	547:	557:	567:	577:

TIMER/COUNTER

600:	610:	620:	630:	640:	650:	660:	670:	100:
601: *	611:	621:	631:	641:	651:	661:	671:	101:
602:	612:	622:	632:	642:	652:	662:	672:	102:
603: *	613:	623:	633:	643:	653:	663:	673:	103:
604:	614:	624:	634:	644:	654:	664:	674:	104:
605:	615:	625:	635:	645:	655:	665:	675:	105:
606:	616:	626:	636:	646:	656:	666:	676:	106:
607:	617:	627:	637:	647:	657:	667:	677:	107:

Figure 4-8. Sample Outputs Used Table Printout

Printer Interface Unit  
CROSS REFERENCE PRINTOUT

OUTPUT REGISTERS USED TABLE V2. 2

SERIES ONE PLUS/SR-21

PAGE 0006

OUTPUT

000:	001:	002:	003:	004:	005:	006:	007:	010:	011:	012:
013:	014:	015:		070:	071:	072:	073:	074:	075:	076:

INTERNAL RELAY

016:	017:	020:	021:	022:	023:	024:	025:	026:	027:
030:	031:	032:	033:	034:	035:	036:	037:		

SHIFT REGISTER

040:	041:	042:	043:	044:	045:	046:	047:
050:	051:	052:	053:	054:	055:	056:	057:

DATA REGISTER

400:	410:	420:	430:	440:	450:	460:	470:
401:	411:	421:	431:	441:	451:	461:	471:
402:	412:	422:	432:	442:	452:	462:	472:
403:	413:	423:	433:	443:	453:	463:	473:
404:	414:	424:	434:	444:	454:	464:	474:
405:	415:	425:	435:	445:	455:	465:	475:
406:	416:	426:	436:	446:	456:	466:	476:
407:	417:	427:	437:	447:	457:	467:	477:
500:	510: *	520:	530:	540:	550:	560:	570:
501:	511: *	521:	531:	541:	551:	561:	571:
502:	512:	522:	532:	542:	552:	562:	572:
503:	513:	523:	533:	543:	553:	563:	573:
504:	514:	524:	534:	544:	554:	564:	574:
505:	515:	525:	535:	545:	555:	565:	575:
506:	516:	526:	536:	546:	556:	566:	576:
507:	517:	527:	537:	547:	557:	567:	577:

Figure 4-9. Sample Outputs Used Table Printout (continued)

pc-s1-83-0022

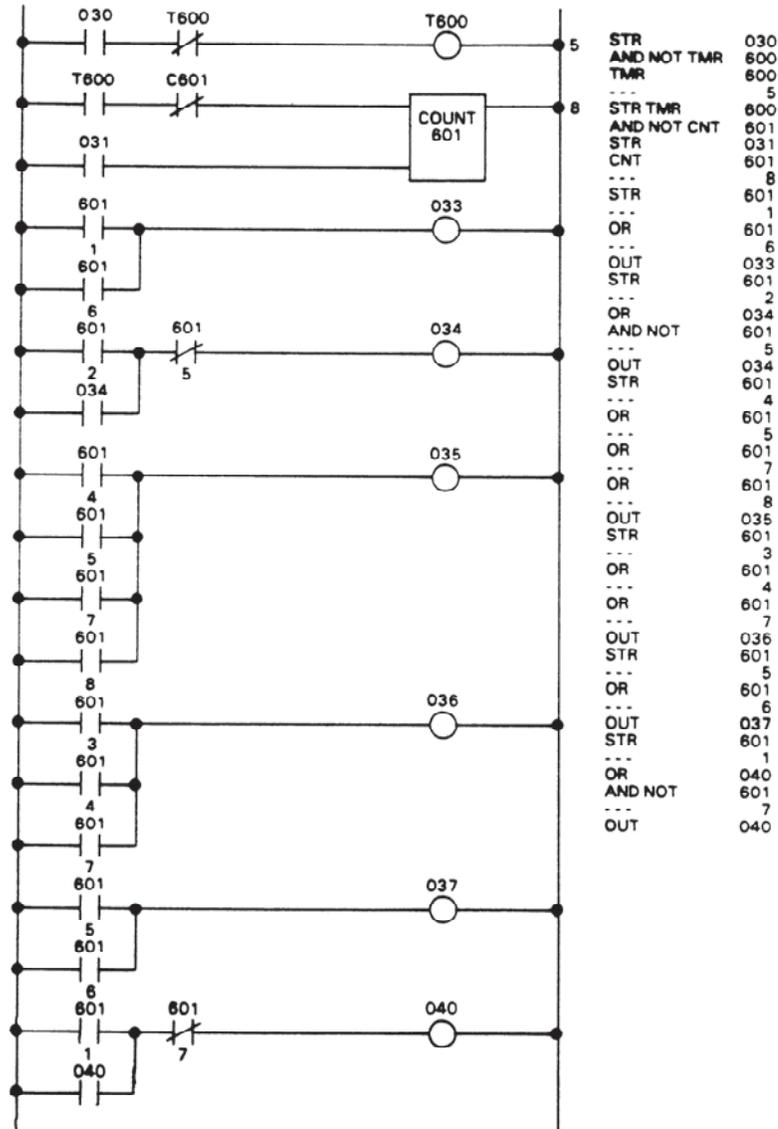


Figure 5-23. Typical Sequencer Logic

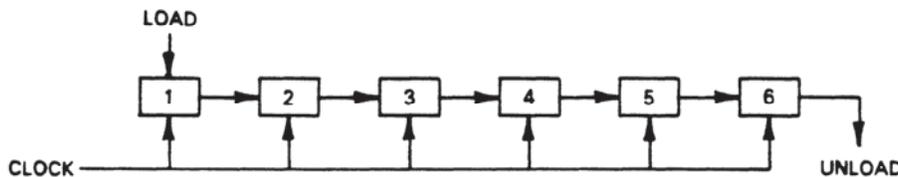
**Shift Register Functional Description**

Another powerful feature of all Series One and Series One Plus PCs is the ability to simulate the operation of shift registers. There are many physical devices that operate similar to shift registers such as an anchor chain, a conveyor belt, an indexing machine, a line of customers at a refund desk, etc.

GEK-90842

There are several common features that can be used to describe the generalized features of a shift register. There is movement in all examples, normally in one direction. This movement can be defined as a group of fixed increments such as one link, one box position, one operation, or one person. The 2 order 1 of activities (pulling chain in, placing boxes on conveyor, loading parts to be worked on/tested, or adding people to the line) normally remains the same - first in, first out. In case of power failure in the plant or building, it is desirable that the shift register retain its content (part position, people order, boxes on conveyor, etc). However, in many cases it is also desirable that under logic control, the shift register be cleared such as start up on Monday morning. It is much easier for the user to clear a retentive shift register than it is to restore a cleared (volatile) content.

pc-s1-83-0023



**Figure 5-24. Typical Shift Register**

A shift register is a group of storage locations that are synchronized by a timing or clock signal. Figure 5.24 illustrates the operation of a typical shift register with six stages. Each stage can store a single bit which can represent a good or bad part, part or no part, ON or OFF, a one or zero, etc. Data is loaded into the first stage, incremented through the shift register one position per clock pulse, and then out the other end. When the clock signal goes from OFF to ON, each stage shifts one position towards the right. The content of stage 6 is unloaded, stage 5 content moves into stage 6, stage 4 into 5, etc. until stage 1 moves into stage 2. Stage 1 is then cleared, awaiting a load signal from another logic.

Within the Series One and Series One Plus, there are 128 retentive storage locations for use with shift registers. Since the Series One family logic functions are easy to use yet provide the user with unmatched flexibility, they allow many separate shift registers to be built of various lengths as long as the total quantity of stages does not exceed 128.

Each shift register requires several pieces of information to perform its function. Since it is a serial operation, there must be a definite beginning and a similar ending. At the start of an operation parts are put into the machine or boxes are put on the conveyor. A signal is required to indicate when a part is to be placed into the shift register at stage one. Similarly, there is a signal that indicates when the shift register should increment or move one position. The third and last signal is one to indicate when the shift register (all stages) should be cleared to zero or OFF conditions.

### Shift Register References

The definition of size for any shift register requires the user to define how many stages are desired, and which references are assigned to this function. The size of the shift register is usually established by the maximum size of the physical device it is controlling. For example, how many links are in the chain, how many boxes can be placed on the conveyor, how many positions has the index machine, or how many people is there space for? The references to stages of the shift registers within the Series One and Series One Plus are to the references 400 through 577. Each stage or position of any shift register should be unique. Storage locations should not be shared between different shift registers. However, multiple logic functions can be programmed to operate upon one shift register to produce bi-directional

shift registers, ring counters, First In First Out or Last In First Out stacks. Each shift register reference can control any number of relay contacts, both normally open and normally closed.

### Shift Register Operation

To illustrate the operation of serial shift registers, Figure 5.25 defines a theoretical problem using a sequential assembly machine. The machine has space for a maximum of 12 parts at any one time. These parts may be cups or other storage locations physically on the machine. The operations are performed while the machine is stationary and then all storage locations (cups) are indexed rapidly one position towards the right. There are three separate operations such as part assembly, welding, clamping, nuts tightened, painting and label placement, etc. Since it would be a waste of material and machine time to operate on locations without a part in place, there is a sensor on the first location to detect when a part is present. To the right is a testing position to determine good parts and bad parts for later separation. Each position of the machine is assigned a number for later reference to the shift register. Notice that positions 3, 5, 7, 10, and 12 are only holding locations to store parts in progress and are related to the mechanical layout of the machine.

pc-s1-83-0024

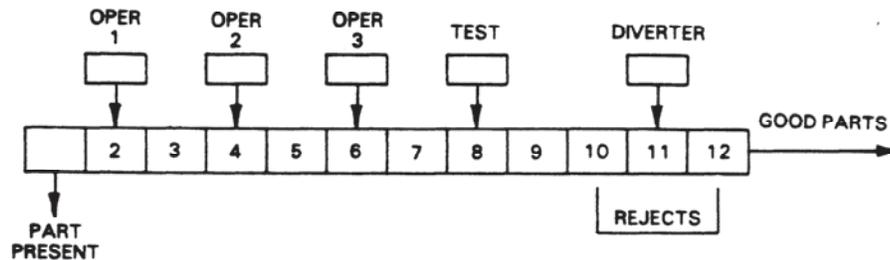


Figure 5-25. Example of Shift Register

Figure 5.26 illustrates the logic, including shift register, used to solve this problem. The following I/O references are assigned for use in this logic and can be changed as necessary to fit your application.

Part Present (Input)	045
Test OK (Input)	046
Reset/All (Input)	047
Operation 1 (Output)	130
Operation 2 (Output)	131
Operation 3 (Output)	132
Test (Output)	133
Diverter (Output)	134
Shift Permissive (Internal)	214

The timing of each operation allows the machine to index one position each 3 seconds. Thus timer 631 produces an output every 3 seconds as long as the internal permissive is satisfied. This coil (214) is developed elsewhere and indicates all operations have been completed and hardware is out of the way, allowing movement without damaging the machinery. The shift register itself has three parallel rungs each of which can be any combination of series and parallel contacts. The first is the input signal. Whenever this is supplying power flow the first stage of the shift register will turn ON and remains ON

GEK-90842

even if the input changes. The middle rung is the clock signal controlling the shifting of all stages. When this signal goes from OFF (no power flow) to ON (power flow), all stages will shift one position. This shifting is completed before other logic is solved. The bottom rung is the reset signal. Whenever this signal is supplying power flow, (all) stages of the shift register will be turned OFF (cleared).

The references used by this shift register and thus the number of stages, must be defined. The reference for the shift register itself is the location of the first stage and immediately following that is the reference for the last stage of that shift register. In our example, input 045 indicates a part present and loads the first stage (431) of the shift register.

Timer 631 (3 seconds) and coil 214 (permissive to shift) are both required before the shift occurs. Finally, input 047 is used to clear all shift register stages perhaps at the start of machine operation or otherwise under operator control. The shift register is built using references 431 through 444 (total 12 stages). The remaining relay logic merely connects the individual stages of the shift register to their output to exercise proper control. The only exception is the next to the last rung. This rung receives the test signal and resets stage 440 if the test is passed. Thus when the later stage (443) receives the data, it will not activate coil 134 to operate the diverter. Therefore, good parts are passed and the diverter operates only for bad parts, not for empty locations or good parts.

a42154

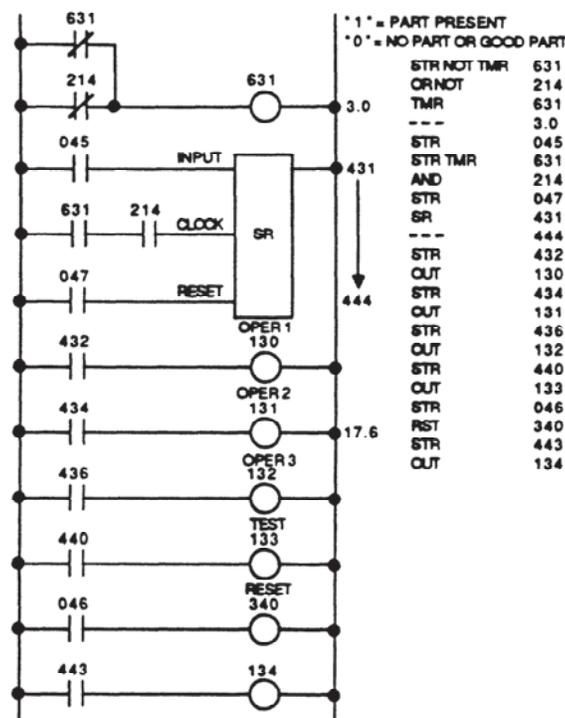


Figure 5-26. Example of Shift Register Logic

Normally many parts will be processed through this machine, continuously moving from input at position 1 to output from position 12. However, for the purpose of illustration, a single part will be examined as it moves down the machine. The part is placed in position 1, energizing input 045 and setting shift register stage 431. After three seconds, timer 631 energizes clocking the shift register causing the ON state for this part to move into stage 432. Then timer 631 will reset itself and stage 431 is cleared. When stage 432 is energized so is output 130 causing operation one to be performed on this part. An additional three seconds pass and another clock cycle causes the ON to move to stage 433. Stage 432 will be loaded from 431 which is OFF as long as additional parts are not processed. The above discussion assumes no delays are experienced with the permissive.

The above operation repeats itself as the part goes through 434 (operation 2), 435 (rest), 436 (operation 3), 437 (rest), and 440 (test) at three second intervals. While at the test position, output 133 is energized and the test OK input (046) is expected. If the part passes the test, input 046 will reset stage 440 making the remainder of the shift register act as if no part was present and allowing the part to pass straight on through. However, if the test is not passed, the ON stage moves to 441, 442, and 443 at three second intervals. At stage 443, it energizes output 134 causing the diverter to push off the defective part into the reject bin.

Other logic can be added to improve the capabilities of this program. For example, a timer could measure the delay from the manual cycle (631 energized) and receipt of the permissive (214). It would also alarm if this delay was excessive such as 5 or 8 seconds beyond the normal 3 seconds. Counters could also be installed to record both good and bad parts produced. This data can be readout and reset via the programmer whenever desired, such as each shift, day, week, etc.

### Forcing I/O References

An invaluable tool in the verification of user logic and field wiring is the ability to force I/O references. The operation of the I/O forcing operation is dependent on the I/O reference being forced. Inputs connected to external devices can be forced on or off for one solution of user logic. All other I/O references can be forced on or off indefinitely, but are overridden by user logic.

The most frequent user of the I/O forcing function is for verification of field wiring of outputs. Since user logic overrides the I/O forcing function it is recommended that this operation be performed with user program memory cleared. The key sequences for forcing I/O references are as follows:

- To force a specific I/O reference ON, enter the sequence: SET, SHF, XXX (I/O) reference), ENT.
- To force a specific I/O reference OFF, enter the sequence: RST, SHF, XXX (I/O) reference), ENT.

<b>WARNING</b>
----------------

**When forcing input points with the set or RST sequence, be aware that the physical state of the input may be overridden. If the forced set or RST occurs in the user logic program before the input is checked in the same I/O scan, the set or RST state will take precedence and would cause an output to be turned on or off at the wrong time.**

## SECTION 3

# Data Operation Instructions for the Series One Plus Programmable Controller

### Data Operations

The Series One Plus PC has, in addition to the basic ladder diagram functions, a group of instructions that include data moves, math functions, logical operations, conversion, and external fault diagnosis. These instructions provide the PC with the capability of performing various data operations. This group of instructions operate on multiple bits rather than one bit at a time. These instructions are not available with a Series One PC. For programs requiring many data operations, it is recommended that the Series One Plus 3.7K PC be used.

All of these instructions require that the data to be operated on be loaded (stored) into a 16-bit register called the accumulator. The data (numerical value) is processed in the CPU as BCD (Binary Coded Decimal). Each BCD value is represented by 4 corresponding bits in the accumulator. A BCD value can be any of the digits 0 through 9. When the specified operation is performed with the contents of the accumulator, the result of the operation is stored in the accumulator. This data can then be transferred to external outputs or stored internally to be used as needed. Additionally, data can be stored in registers (sixty-four 16-bit registers) and those registers referenced when programming the data operations. Each register can contain a 4-digit BCD number. Figure 5.27 is a block diagram showing how the data operations are performed.

Registers and the accumulator may also contain binary data, however arithmetic functions cannot be performed until the binary values are converted to BCD.

a40848

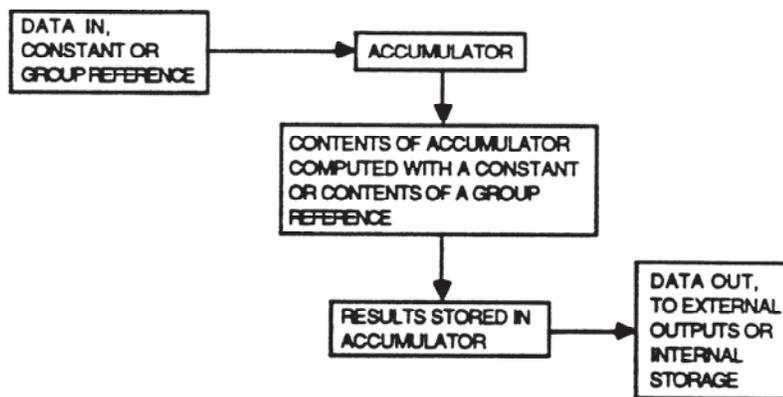


Figure 5-27. Data Operation Block Diagram

The contents of the accumulator do not change until new data is loaded into it with a D STR instruction or a computation takes place and the result is stored in the accumulator.

### Using References for Data Operations

When entering an instruction for a data operation, the instruction is first specified, then the data. The data can be a constant value or a group reference that contains the data to be used.

### Entering a Constant Value

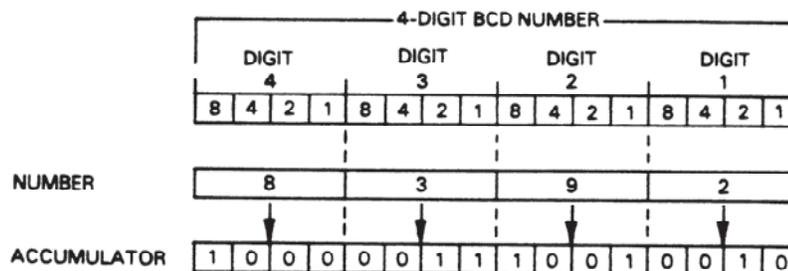
a40849



Where XXXX is a 4-digit BCD number

When a constant (4-digit BCD number) is entered as the data value, the bits of the BCD numbers will be stored in the corresponding position in the accumulator, i.e., the least significant digit in the lower 4 bits of the accumulator, and the most significant digit in the upper 4 bits of the accumulator.

a40850



### Entering a Group Reference



The I/O points, internal coils, and shift registers are treated as 8-bit registers when programming data operations. The references are then referred to as group references. By selecting the R key before a value, that value is operated on as an 8-bit or 16-bit register. For example, R 1 5 actually references 8 consecutive I/O points, 150 to 157. This is a convenient way of reading in data to the accumulator from input modules, or writing data from the accumulator to output modules for use as required by the user program. Certain instructions operate on 16 bits, in those cases the data will be read from or written to 16 consecutive references (one 16 circuit module or 2 adjacent 8 circuit modules). The group reference number for the references mentioned above can be either 1 or 2 digits.

GEK-90842

**Data Register References**

Data registers are referenced as 3-digit octal values with a valid range from 400 to 577. Although each individual register actually refers to one 8-bit byte, most registers are operated on as 16-bit words (2 consecutive registers). For example, if the key sequence R 4 0 2 is entered, the data operation uses data registers 402 and 403. If a 4-digit BCD value (for example 1234) were entered into the register, register 402 would contain the 2 least significant digits (34) and 403 the 2 most significant digits (12). A useful feature of the registers is that they can contain a value to be used as the preset for timers or counters. Unused T/C references 600-673 can be used as data registers. Each T/C reference operates on a 16-bit register.

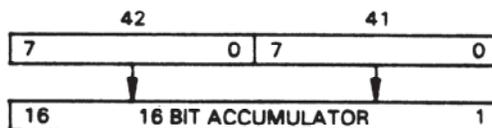
**Timer/Counter References**

The timer/counter references are also 3-digit octal values. Each of the timer/counter references represent a 16-bit register. Valid range for the timer/counter references is 600 to 677. These registers contain the accumulated value of a timer or counter. It is important to note that the timer/counter accumulated value is not the same as the accumulator read to or written from when programming any of the data operations.

**Example of Specifying a Group Reference**

An example of specifying a group reference and how it relates to the accumulator is shown below. Example: Reference specified is 41.

a40851



Contents of 41 and 42 (16 bits) are loaded into the accumulator

**NOTE**

Discrete bit 410 will be stored in the first position of the accumulator; discrete bit 427 will be stored in the last position (16) of the accumulator.

**Programming the Data Operation Instructions**

When a program is entered to perform any of the data operations, the first instruction must have at least one conditional contact entered immediately before it. To enter any of the data instructions, the F key must first be selected, then the Function number (F50, F63, F82, etc.) of the instruction. These numbers are assigned internally, by the CPU's operating system, to each of the data operation instructions and are listed in Table 5.8. After the F X X number has been entered, the program address will advance 1 step and the second word (data entry) should now be entered. The data entry is selected by entering SHF X X X X, for a constant value, or R X X X for a group reference.

Table 5-8. Data Operation Function Numbers

Instruction	Key Sequence	Instruction	Key Sequence
D•STR	F50	MULTIPLY	F73
D•STR1	F51	DIVIDE	F74
D•STR2	F52	D•AND	F75
D•STR3	F53	D•OR	F76
D•STR5	F55	SHIFT RIGHT	F80
D•OUT	F60	SHIFT LEFT	F81
D•OUT1	F61	DECODE	F82
D•OUT2	F62	ENCODE	F83
D•OUT3	F63	INV	F84
D•OUT5	F65	BIN (BCD TO BIN)	F85
CMPR	F70	BCD (BIB TO BCD)	F86
ADDITION	F71	EXTERNAL FAULT	
SUBTRACTION	F72	DIAGNOSIS	F20

The remainder of this chapter lists each of the data operation instructions and the information required for programming them. The format used to describe each of the instructions is as follows:

Mnemonic name for each instruction.

Function number that must be entered with the programmer (F50, F60, etc.)

Illustration of how each instruction operates in relation to the accumulator.

List of valid references (where required).

Keystrokes required to enter the instruction.

Program example to enhance the description of certain instructions.

### Special Function Coils for Data Operations

A group of 6 special purpose internal coils in the Series One Plus are used as indicators of certain conditions that may occur when some of the data operations are performed. When these conditions occur, the applicable coil is turned on as a "Flag" or indicator that the condition has occurred. These coils can be included in your program as conditional contacts to turn on an output or perform some function to indicate that a Flag is on. These special purpose coils and their meanings are:

COIL REFERENCE	FLAG INDICATION
772	>, Greater Than
773	=, Equal to
774	<, Less Than
775	Carry or Borrow
776	Zero
777	Overflow

GEK-90842

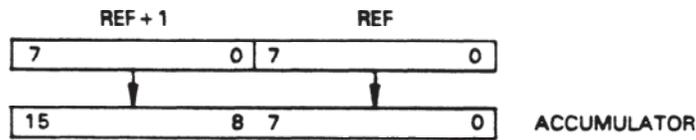
**D•STR (F50)**

D•STR (Data Store) is a 2 word instruction which, when executed, loads the accumulator with a 4 digit BCD constant or the contents of a specified 2 byte reference.

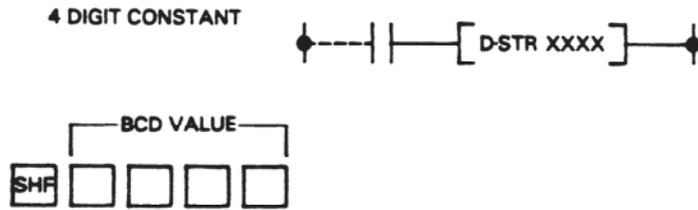
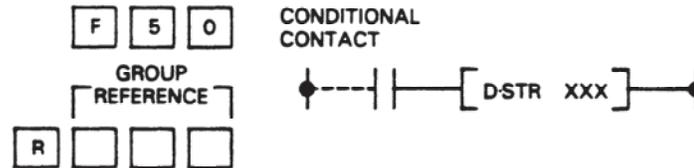
**NOTE**

When loading I/O References or other data, be sure that it is in BCD Format. If not it should be converted before using the data for arithmetic functions.

a40852



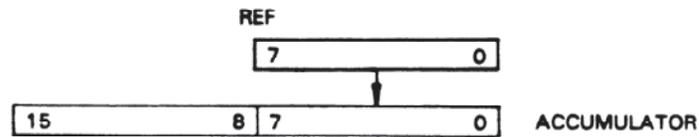
Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)



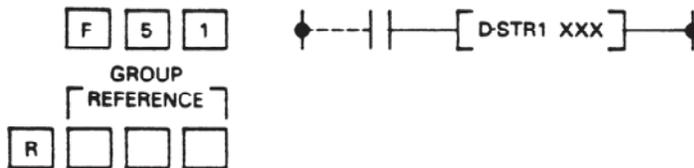
**D•STR1 (F51)**

This is a 2 word instruction which, when executed loads the lower 8 bits of the accumulator with the contents of a specified 1 byte reference. The upper 8 bits (8-15) will be zeros.

a40853



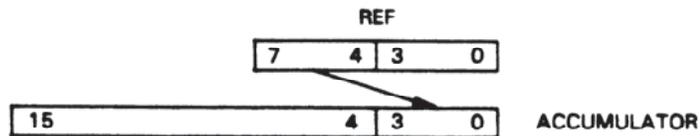
Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)



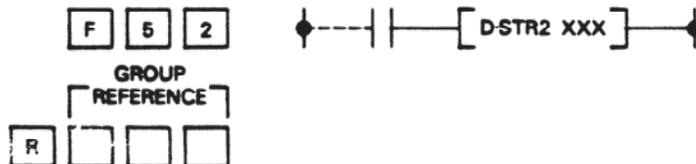
**D•STR2 (F52)**

This is a 2 word instruction which, when executed loads the lower 2 bits of the accumulator with the upper 4 bits of a specified 1 byte reference.

a40854



Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)

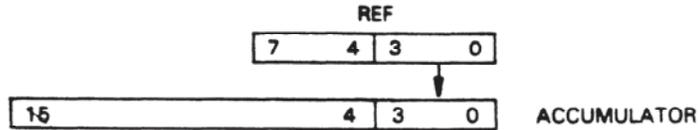


GEK-90842

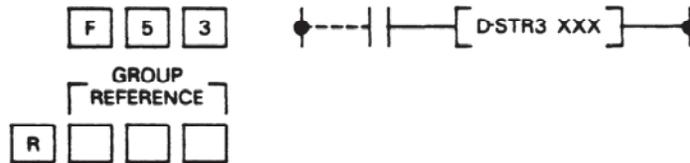
**D•STR3 (F53)**

This is a 2 word instruction which, when executed loads the lower 4 bits of the accumulator with the lower 4 bits of a specified 1 byte reference.

a40855



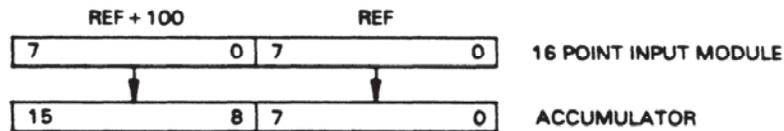
Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)



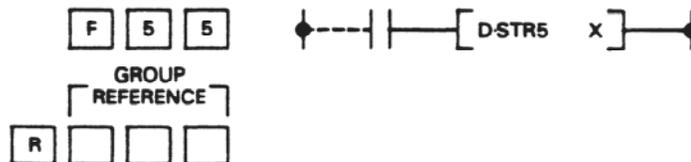
**D•STR5 (F55)**

This is a 2 word instruction which, when executed loads the accumulator with a BCD value from the 2 byte contents of a 16 circuit Input module.

a40856



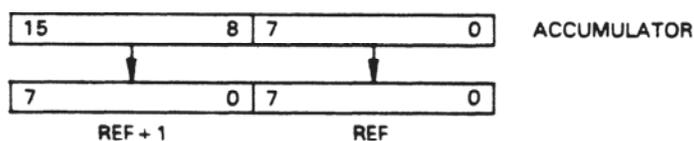
Valid Group Reference	
000 - 005	16 Point Input Module



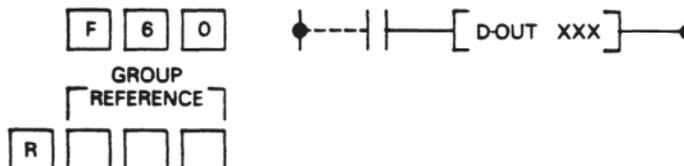
**D•OUT (F60)**

D•OUT (Data Out) is a 2 word instruction, which when executed transfers the contents of the accumulator (16 bits) to a specified 2 byte (group) reference (Output, Internal Relay, Shift Register coils, Data Registers or Timer/Counter accumulate value register).

a40857



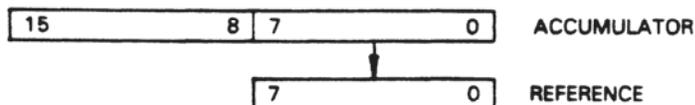
Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)



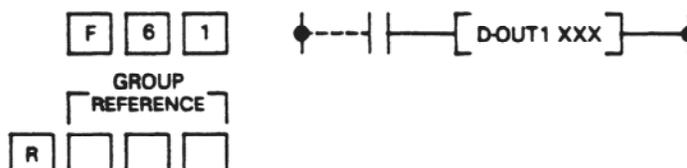
**D•OUT1 (F61)**

D•OUT1 is a 2 word instruction which, when executed will write the contents of the first 8 bits (0-7) of the accumulator to a specified reference.

a40858



Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)

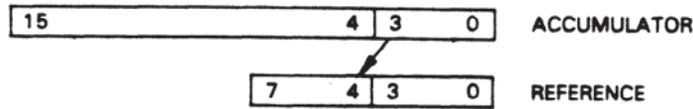


GEK-90842

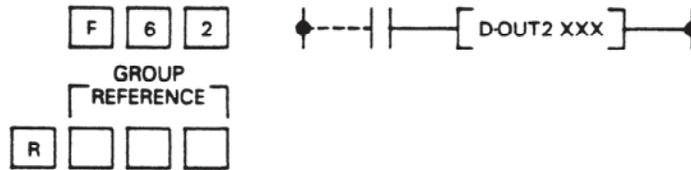
**D•OUT2 (F62)**

D•OUT2 is a 2 word instruction which, when executed will write the contents of the lower (first) 4 bits of the accumulator to the upper 4 bits of a specified reference.

a40859



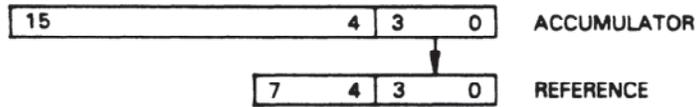
Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)



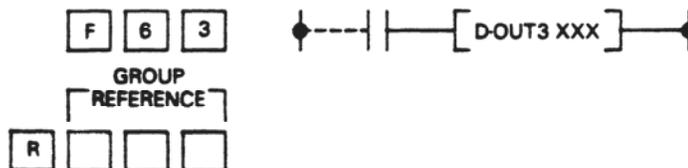
**D•OUT3 (F63)**

D•OUT3 is a 2 word instruction, which when executed will write the contents of the lower 4 bits of the accumulator to the lower 4 bits of a specified reference.

a40860



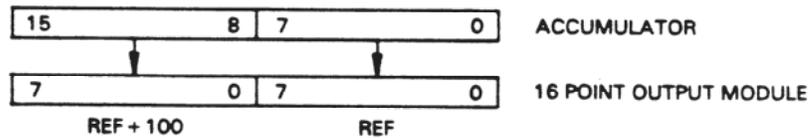
Valid Group References	
000 - 015	I/O Points
070 - 076	I/O Points
016 - 037	Internal Coils
040 - 057	Shift Register Coils
400 - 577	Data Registers (8-bit)



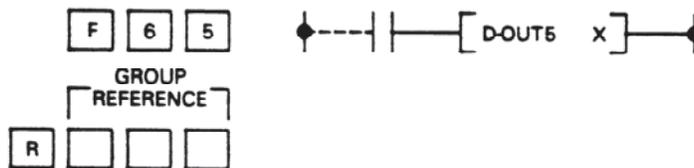
**D•OUT5 (F65)**

D•OUT5 is a 2 word instruction, which when executed will write the contents of the accumulator to a 2 byte reference which must be a 16 circuit Output module.

a40861



Valid Group Reference	
000 - 005	16 Point Output Module



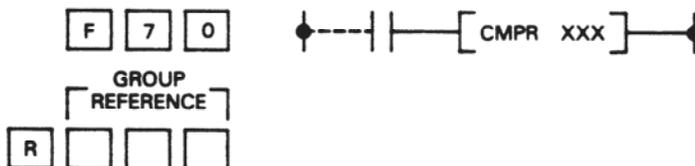
**CMPR (F70)**

The contents of a specified 2 group reference or a 4 digit BCD constant are compared to the contents of the accumulator. The result of the comparison will cause one of three special purpose internal coils to turn on, depending on whether the accumulator value is greater than (>), equal to (=) or less than (<) the value specified by the reference or constant.

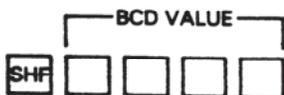
a40862

Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
500 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

FLAGS		
>	=	<
772	773	774

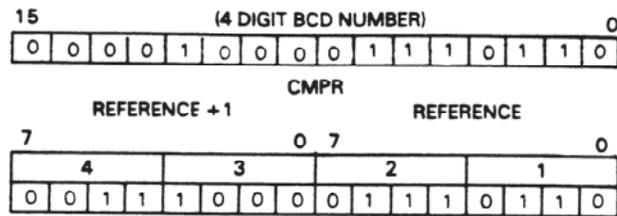


4 DIGIT CONSTANT



GEK-90842

• Compare sequence



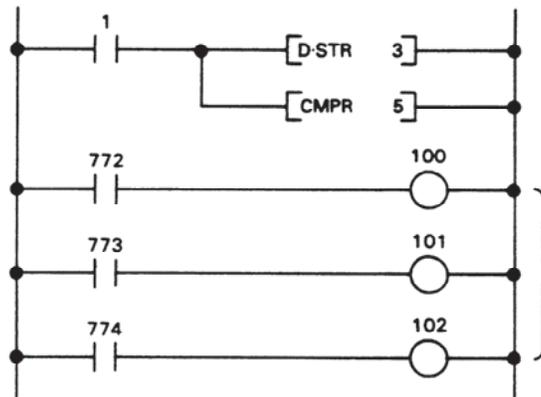
a40863

ACCUMULATOR CONTENTS

2 GROUP REFERENCE OR 4 DIGIT CONSTANT

- If Accumulator > Reference, then 772 turns ON.
- If Accumulator = Reference, then 773 turns ON.
- If Accumulator < Reference, then 774 turns ON.

• Programming example, Compare



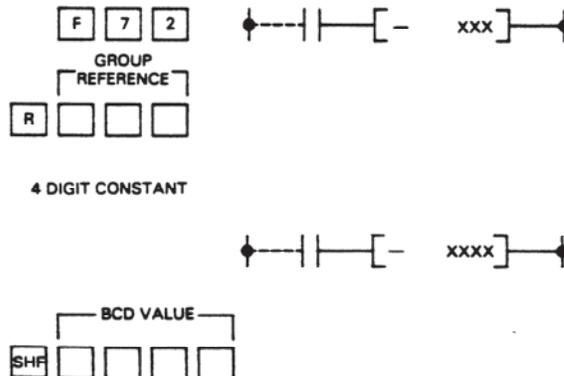
Addition (+), BCD 4 Digit (F71)

The contents of a 4 digit constant or a specified 2 group reference are added to the contents of the accumulator. The result of the addition is stored in the accumulator as a 4 digit BCD number. If the answer is >9999, the carry flag (Internal coil, 775) will turn on. If the answer is zero, the zero flag (776) will turn on. The flag coils can be used in a program as references for contacts to turn on an output connected to a light or other device to provide an indicator of the flag conditions.

a40864

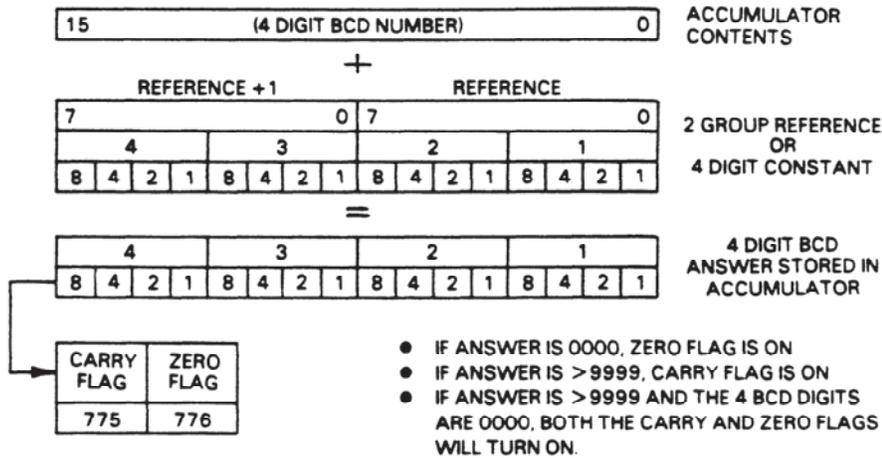
Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

FLAGS	
CARRY OR BORROW	ZERO
775	776

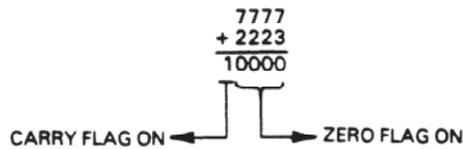


• Addition sequence

a40865



EXAMPLE:

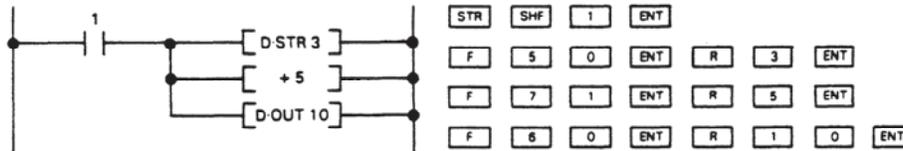


• Addition Programming Example

Example 1

4 BCD digits are loaded in accumulator from Inputs 0030-0047.

a40866



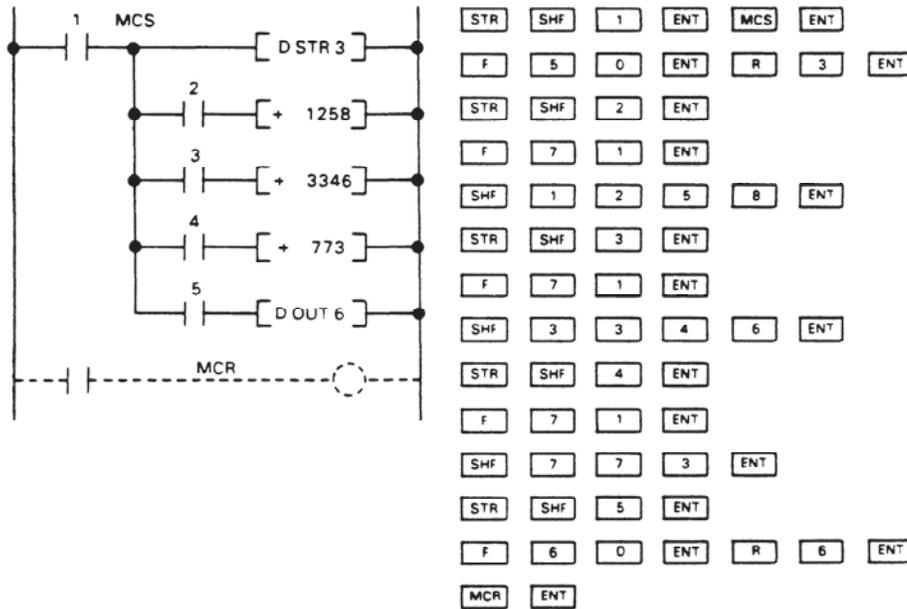
- The contents of the accumulator are added to data from Inputs 0050-0067.
- The answer is stored in the accumulator and also written to Outputs 0100 - 0117. If D OUT is specified as a Data Register, Internal coil, etc.; the answer would be stored at that location for further use in the program.

GEK-90842

Example 2

This example shows more than one constant value entered in the program. By closing the respective contact, a different number can be selected to be added to the contents of the accumulator.

a40867



When I1 is closed, MCS is energized and does not control the logic following it. The number selected by closing one of the contacts, I2, I3, or I4 is added to the contents of the accumulator (read from Inputs 0030-0047). When I5 is closed, the answer will be written to Outputs 0060-0077.

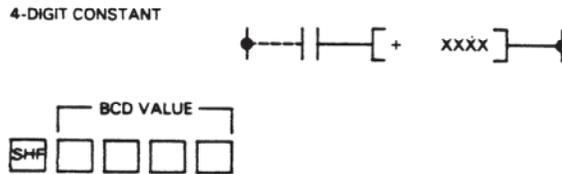
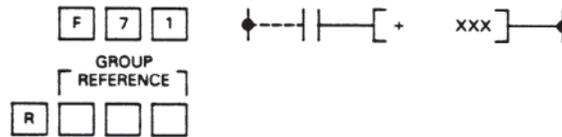
**Subtraction (-), BCD 4 Digit (F72)**

The contents of a specified 2 group reference or a 4 digit BCD constant value are subtracted from the contents (BCD) of the accumulator. If the answer after the subtraction is positive, it is written directly to the accumulator. If the answer is negative, the Borrow Flag (775) turns ON, the absolute value is subtracted from 10000 and the resulting value is written to the accumulator. If the answer after subtraction is zero, the Zero flag (776) turns ON.

a40868

Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

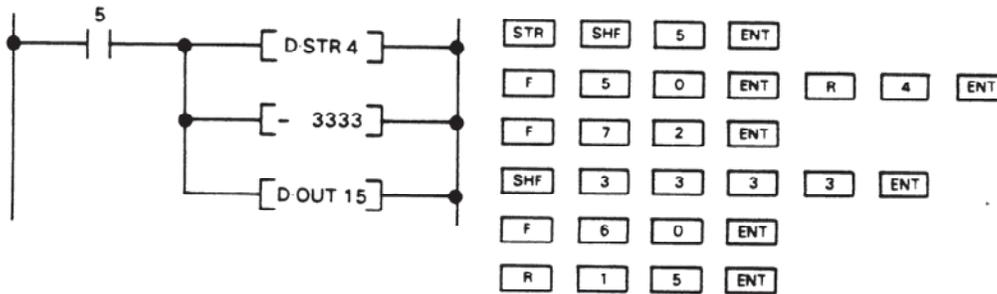
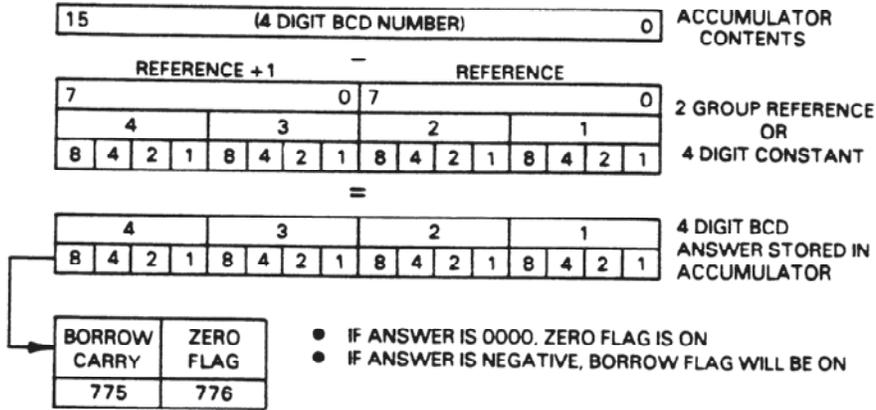
FLAGS	
CARRY	ZERO
775	776



GEK-90842

• Subtraction sequence

a40869



• Subtraction programming example

- Data is entered into the accumulator from Inputs 0040 - 0057.
- A constant value (3333) is subtracted from the contents of the accumulator.
- The answer is stored in the accumulator and is written to outputs 0150 - 0167.

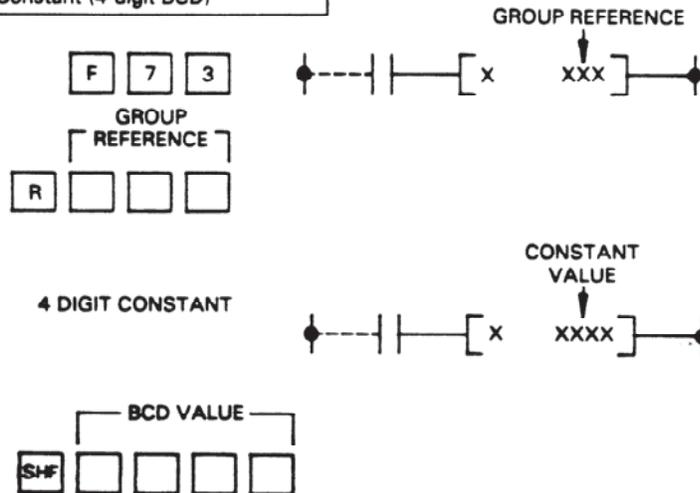
**Multiplication (X), BCD 4 Digit (F73)**

The contents of the accumulator are multiplied by the contents of a specified 2 group reference or a 4 digit BCD constant. The answer can be from 1 to 8 digits. The lower 4 digits of the answer are stored in the accumulator and digits 5-8 are stored in data registers 576 and 577, which are the auxiliary accumulator. If the answer is zero, the Zero flag (776) will turn ON.

a40870

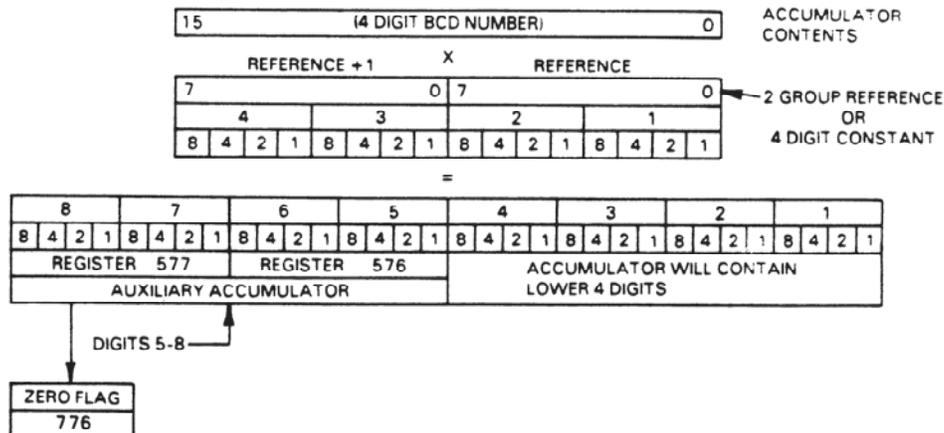
Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

FLAG
ZERO
776



• Multiplication sequence

a40871



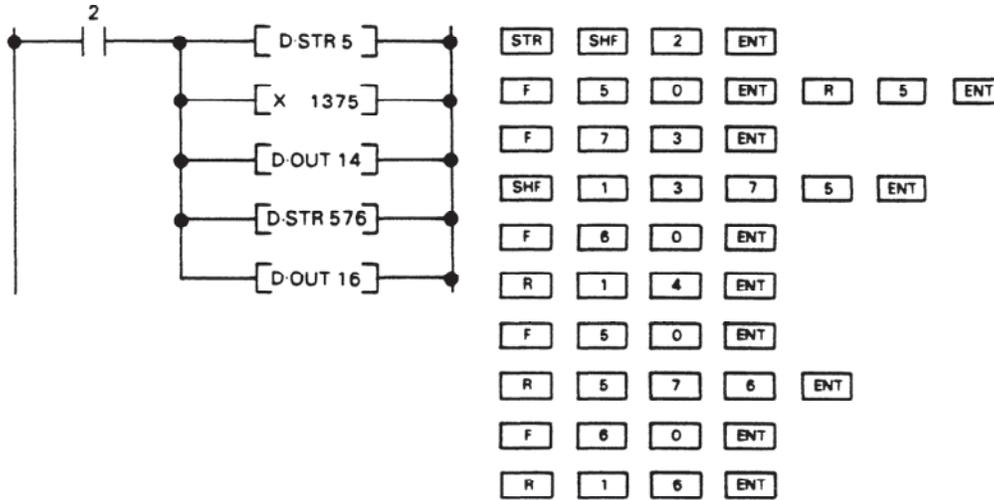
- The lower 4 digits (1, 2, 3, 4) of the answer will be stored in the accumulator. The upper 4 digits (5, 6, 7, 8) will be stored in the auxiliary accumulator (Data Registers 576 and 577).
- If the answer is 00000000, the Zero flag (776) is ON.

GEK-90842

• Multiplication programming example

Multiplying 4 digits by 4 digits with an answer containing 8 digits.

a40872



- A 4 digit BCD value is entered into the accumulator from Inputs 0050-0067.
- A constant value (1375) is entered as the multiplier.
- Assume that the answer is an 8 digit BCD number. The lower 4 digits are stored in the accumulator and transferred to Outputs 0140-0157. The upper 4 digits are stored in the auxiliary accumulator (Data Registers 576, 577) and transferred to Outputs 0160-0177.

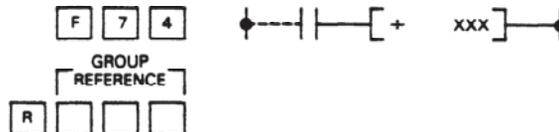
Division (÷), BCD 4 Digit (F74)

The contents of the accumulator are divided by the contents of a specified 2 group reference or a 4 digit BCD constant. The 4 digits of the quotient are stored in the accumulator and the 4 digits of the remainder are stored in the auxiliary accumulator (Data Registers 576, 577). If either the dividend or divisor are zero, the Zero flag (776) will turn ON. If the divisor is zero, the Overflow flag (777) will also turn ON.

a40873

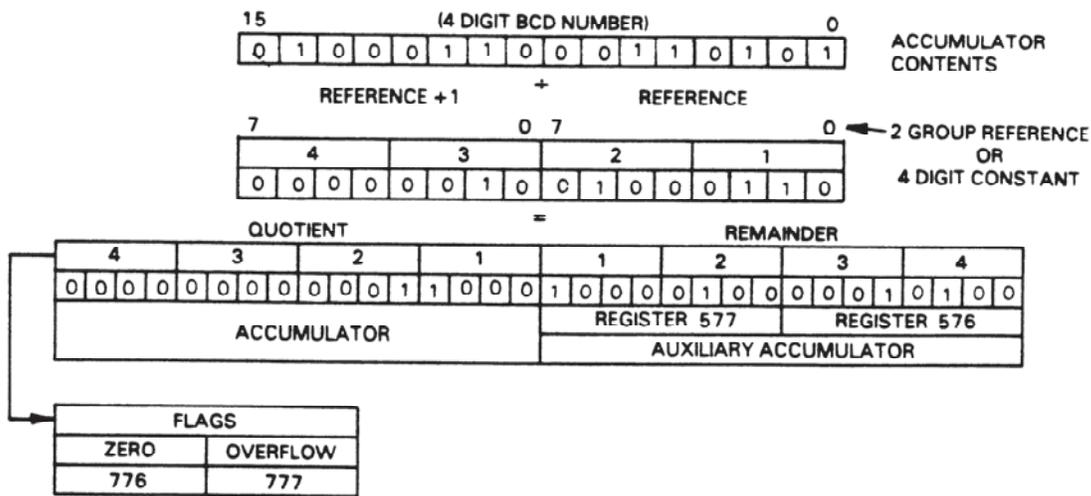
Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 676	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers
0000 - 9999	Constant (4-digit BCD)

FLAGS	
ZERO	OVERFLOW
776	777



• Division sequence

a40874

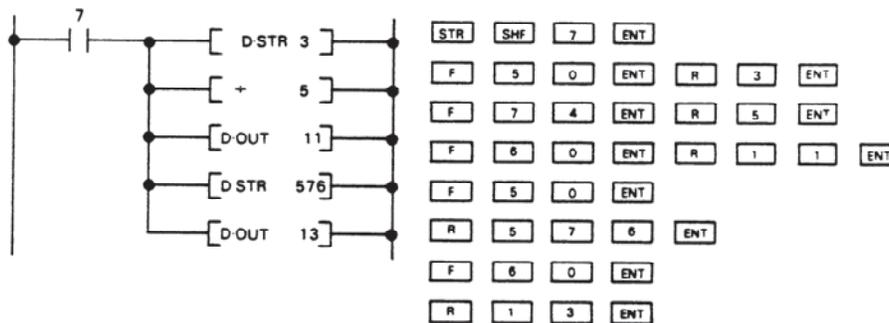


• The quotient (0 - 4 digits) will be stored in the accumulator as shown. the remainder (0 - 4 digits) will be stored in the auxiliary accumulator (Data Registers 577, 576) as shown.

• Division programming example

Division of a 4 digit number by another 4 digit number showing how the quotient and remainder can be written to Outputs.

a40875



- A 4 digit number is entered into the accumulator from Inputs 0030 - 0047.
- That number is divided by a number specified by Inputs 0050 - 0067.
- The 4 digits of the quotient are stored in the accumulator and are also written to Outputs 0110 - 0127.
- The 4 digits of the remainder are stored in the auxiliary accumulator (Data Registers 577, 576) and are written to Outputs 0130 - 0147.

GEK-90842

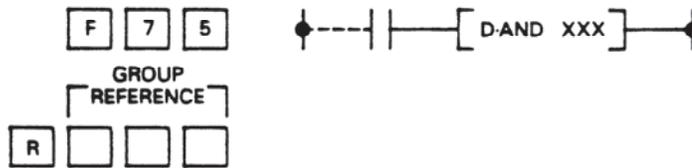
**D•AND (DATA AND) (F75)**

The contents of the accumulator are logically bit AND-ed with the contents of a specified 2 group reference or a 4 digit BCD constant. The result is stored in the accumulator. If the answer is 0000, the Zero flag (776) will turn ON.

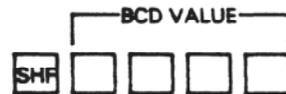
a40876

Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

FLAG
ZERO
776

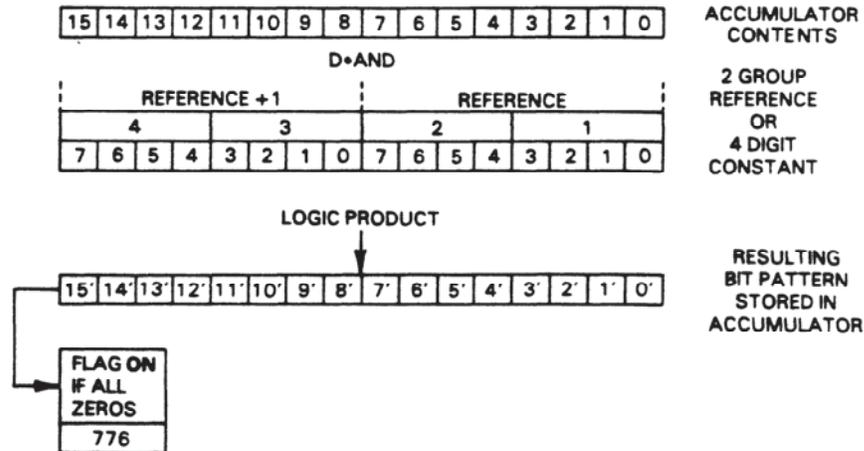


4 DIGIT CONSTANT



**D•AND Sequence**

a40877

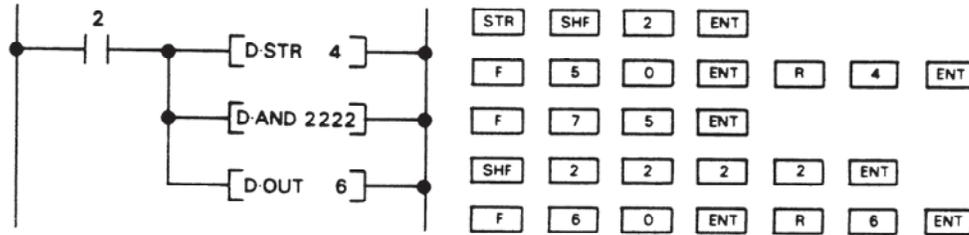


- Truth table for D•AND operation.  
 A = Accumulator contents  
 D = Contents of reference or constant

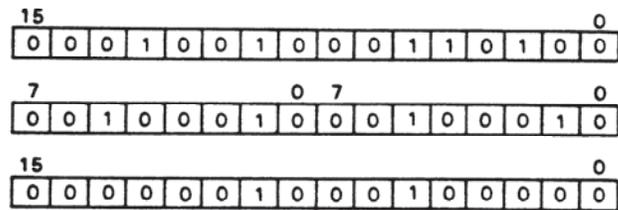
A	D	RESULT
0	0	0
0	1	0
1	0	0
1	1	1

• Programming example, D•AND

a40878



- Assume the data entered into the accumulator from Inputs 0040 - 0057 = 1234
- Is logically AND-ed with 2222
- The resulting answer stored in the accumulator would be 0220
- 0220 is written to Outputs 0060 - 0077
- Accumulator and reference contents for above operation are shown below



GEK-90842

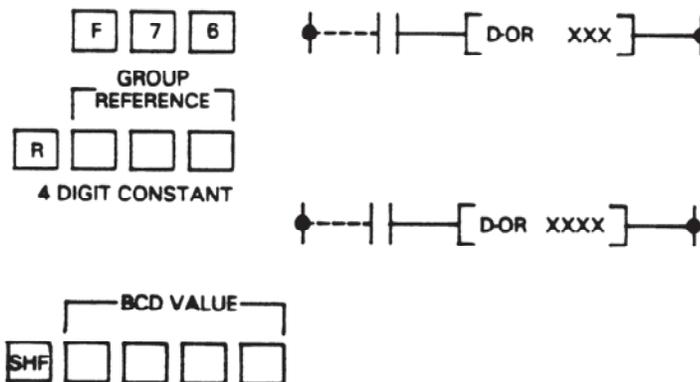
**D•OR (DATA OR) (F76)**

The contents of the accumulator are logically bit OR-ed with the contents of a specified 2 group reference or a 4 digit BCD constant. The result is stored in the accumulator. If the answer is 0000, the Zero flag (776) will turn ON.

Valid Group References	
000 - 014	I/O Points
070 - 075	I/O Points
016 - 036	Internal Coils
040 - 056	Shift Register Coils
400 - 576	Data Registers (8-bit)
600 - 677	T/C Accumulated Value Registers (16-bit)
0000 - 9999	Constant (4-digit BCD)

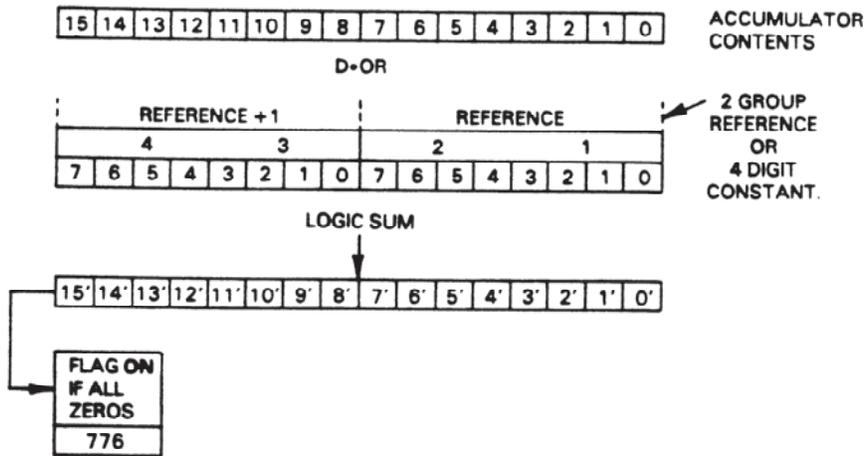
a40879

FLAG
ZERO
776



**D•OR Sequence**

a40880

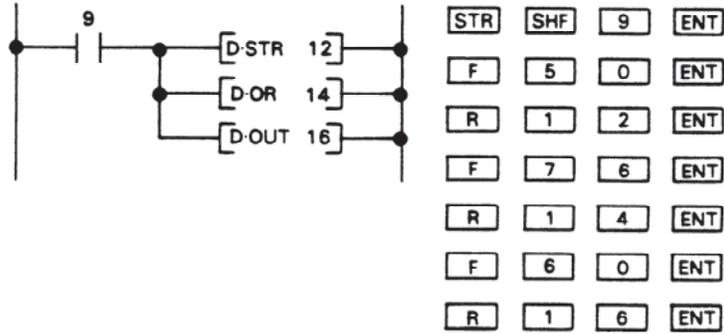


- Truth table for D•OR operation.  
 A = Accumulator contents  
 D = Contents of reference or constant

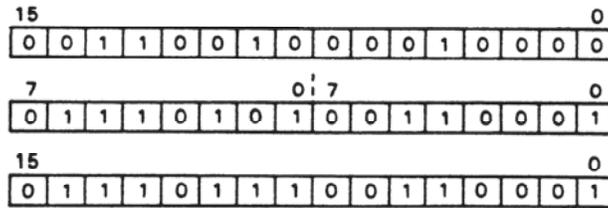
A	D	RESULT
0	0	0
0	1	1
1	0	1
1	1	1

• Programming example, D•OR

a40881



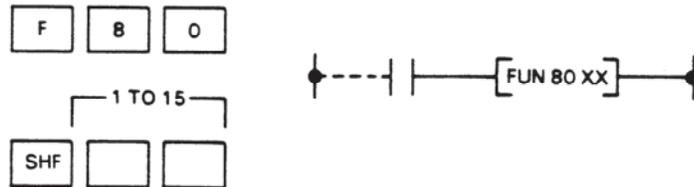
- Assume data entered into the accumulator from Inputs 0120 - 0137 = 3210
- Is logically OR-ed with data from Inputs 0140 - 0157 = 7531
- The resulting answer stored in the accumulator will be 7731
- 7731 is written to Outputs 0160 - 0177
- Accumulator and reference contents for above operation are shown below



GEK-90842

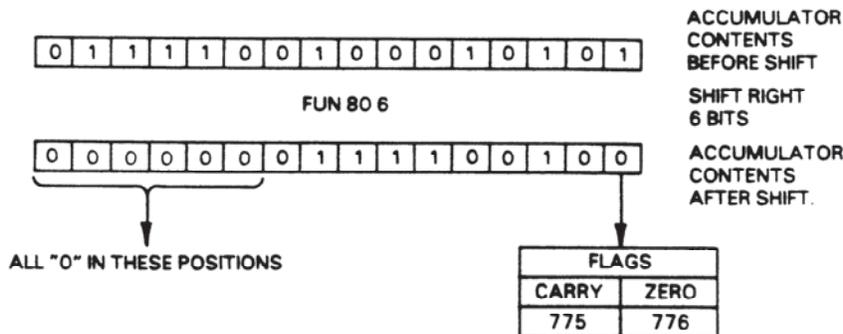
**Shift Right (F80)**

The contents of the accumulator are shifted to the right by the number of shifts (bits) specified by the instruction. The number of bits shifted can be from 1 to 15. The bit positions of bits shifted from the left are filled with zeros. If a '1' is shifted out of the accumulator as a result of the shift, the Carry flag (775) will turn on. If after the shift, the accumulator contains only zeros, the Zero flag (776) will turn on.



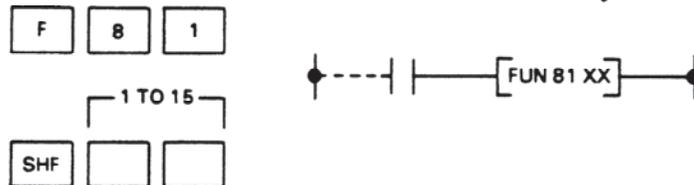
a40882

- FUN 80, Shift Right sequence (6 shifts in example)



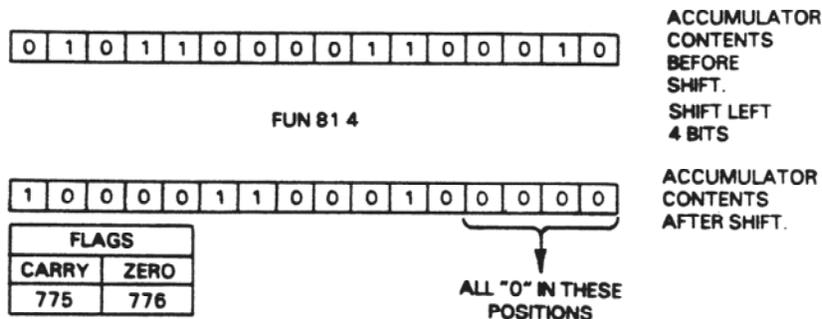
**Shift Left (F81)**

The contents of the accumulator are shifted to the left by the number of shifts (bits) specified by the instruction. The number of bits shifted can be from 1 to 15. The bit positions of bits shifted from the right are filled with zeros. If a '1' is shifted out of the accumulator as a result of the shift, the Carry flag (775) will turn on. If after the shift, the accumulator contains only zeros, the Zero flag (776) will turn on.



a40883

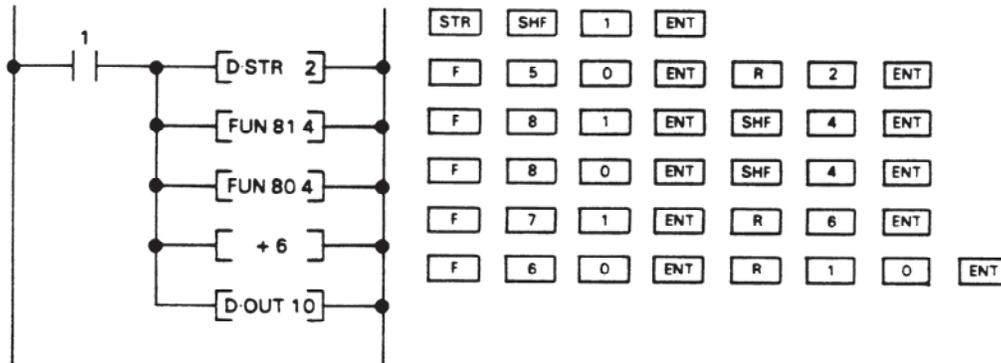
- FUN 81, Shift left sequence (4 shifts in example)



• Programming example using FUN 80 and FUN 81

This example shows a technique that can be used for deleting unnecessary data after it has been entered into the accumulator. In the example the fourth BCD digit will be deleted.

a40884



• Data entered into accumulator from Inputs 0020 - 0037.

•

• Shifted left, 4 bits.

•

• Shifted right, 4 bits.

•

• The fourth digit (5) has been deleted.

•

• The remaining value is added to the contents of Inputs 0060 - 0077. The result is stored in the accumulator and written to Outputs 0100 - 0117.

5	3	6	2
---	---	---	---

3	6	2	0
---	---	---	---

0	3	6	2
---	---	---	---

GEK-90842

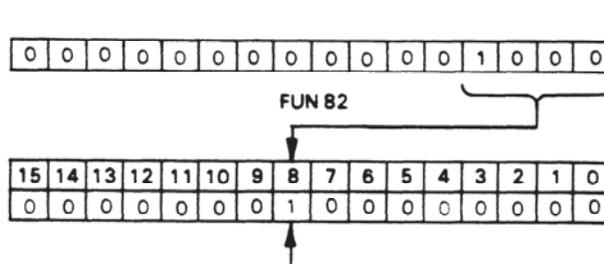
**Decode (F82)**

The lower 4 bits (least significant) of the accumulator are decoded to a decimal number from 0 to 15. A "1" is placed in the bit position in the accumulator that corresponds to the decoded decimal number. The upper 12 bits of the accumulator are disregarded for this operation.

a40885



• FUN 82 Decode sequence



LOWER 4 BITS OF ACCUMULATOR CONTAIN 1 BCD DIGIT.

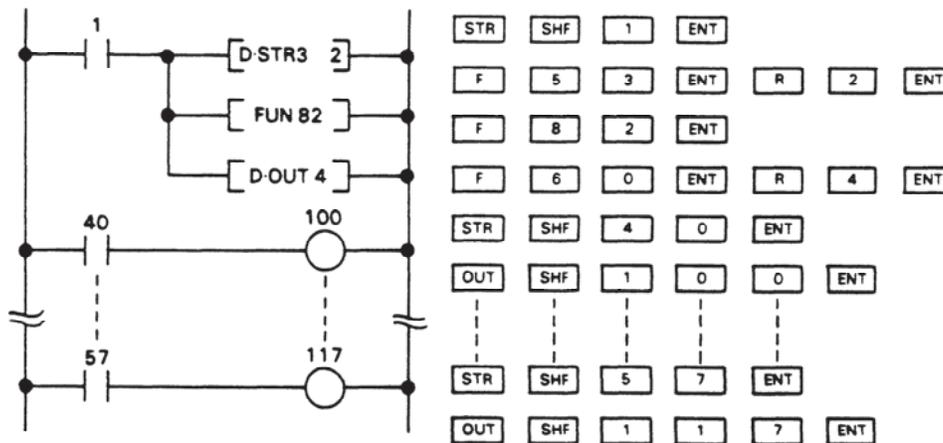
WHICH IS DECODED TO A DECIMAL NUMBER.

A "1" WILL BE PLACED IN THE ACCUMULATOR POSITION CORRESPONDING TO THAT DECIMAL NUMBER. ALL OTHER POSITIONS WILL CONTAIN ZEROS.

• If the BCD digit to be decoded is an 8, A "1" will be placed in the accumulator at that position as shown.

• Programming example, FUN 82

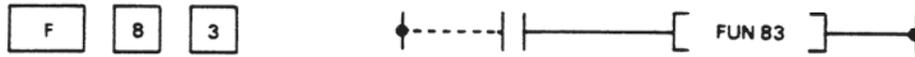
a40886



- A BCD digit (for example: 6) is loaded into the accumulator (lower 4 bits) from the lower 4 bits of the reference, Inputs 20, 21, 22, and 23. The bit pattern loaded in the accumulator would be 0 1 1 0.
- The BCD digit is decoded to a decimal number between 0 - 15 (in this example 6).
- That number is stored in the corresponding bit position in the accumulator.
- Outputs 0040 to 0057 correspond to the 16 bits of the accumulator. One of the outputs will be enabled when the accumulator position has a "1" placed in it. When the contact corresponding to the output reference is closed, a coil 100 - 117 will turn on.

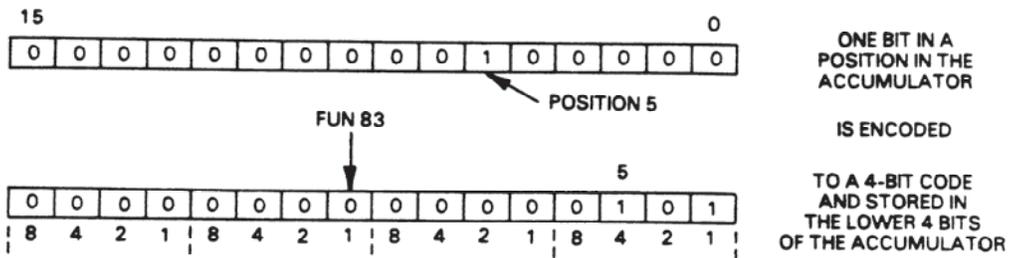
**Encode (F83)**

A bit in a position in the accumulator is encoded to a 4-bit code representing the position number (0-15) in the accumulator. The resulting 4-bit code is placed in the lower four bits of the accumulator.



a40887

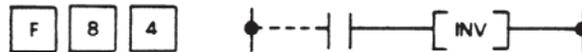
• FUN 83 Encode sequence (example)



- The upper 12 bits of the accumulator will contain zeros after the FUN 83 has been executed.

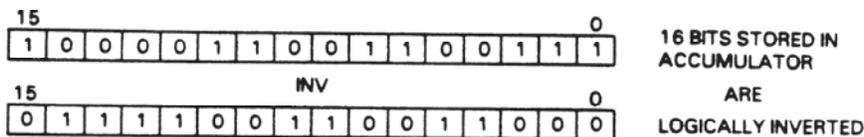
**INV (F84)**

The 16 bits stored in the accumulator are logically inverted; i.e., 1 to 0 and 0 to 1.

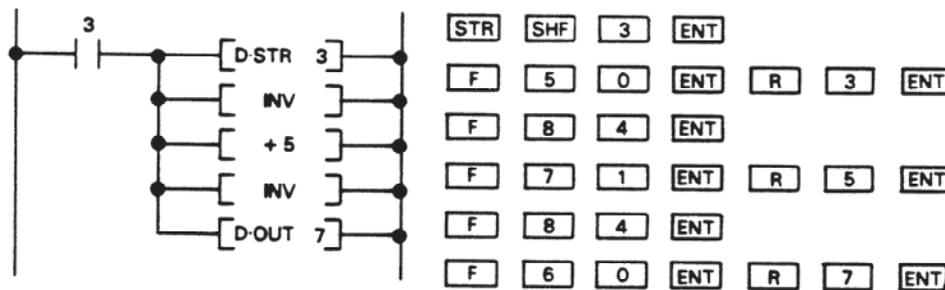


a40888

• Invert sequence



• Programming example, INV



- Data entered into the accumulator from Inputs 0030 - 0047 is inverted.
- The inverted data is added to data from Inputs 0050 - 0067. The answer is stored in the accumulator, then inverted again.
- Data is written to Outputs 0070 - 0107

GEK-90842

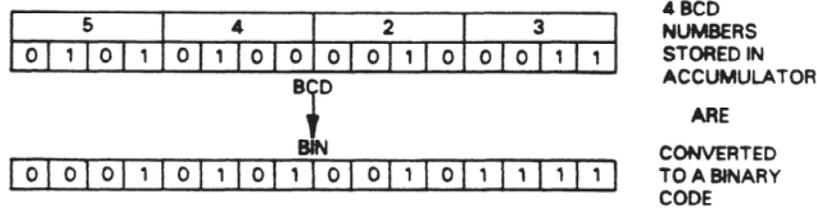
**BIN (BCD to Binary) (F85)**

The BCD value stored in the accumulator is converted to a binary code corresponding to that value.

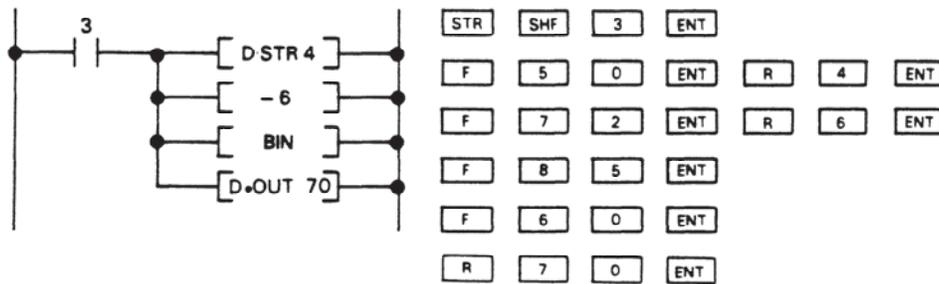
a40889



- Binary conversion sequence



- Programming example, BIN

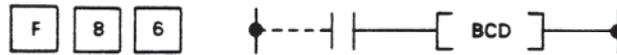


- Data is stored in the accumulator from Inputs 0040 - 0057.
- Data entered from Inputs 0060 - 0077 is subtracted from the value in the accumulator.
- The resulting answer is stored in the accumulator, where it is converted to a binary code.
- The binary code is written to Outputs 0700 - 0717.

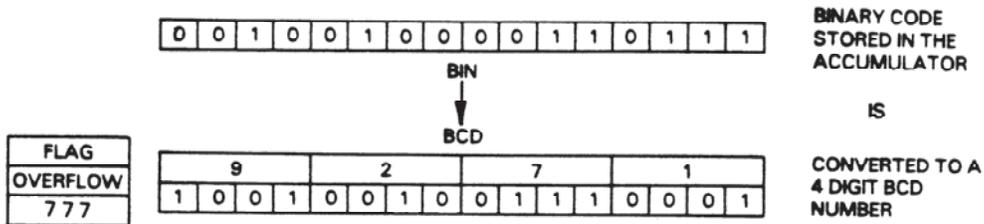
**BCD (Binary to BCD) (F86)**

A binary code stored in the accumulator is converted to a 4 digit BCD value. If the resulting BCD value is >9999, the overflow flag (777) will turn on. The flag coil can be used to turn on an output device as an indication of the overflow.

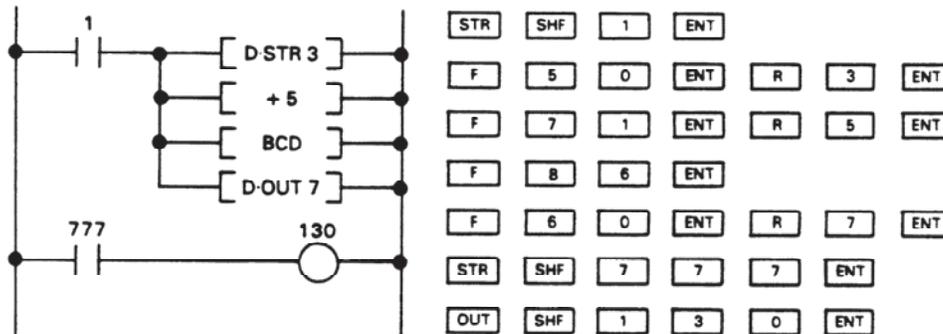
a40890



• BCD conversion sequence



• Programming example, BCD



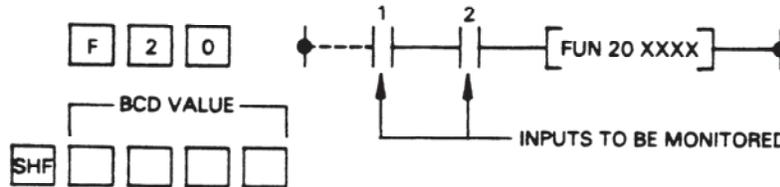
- Data is stored in the accumulator from Inputs 0030 - 0047.
- Data from Inputs 0050 - 0067 is added to the value in the accumulator.
- The resulting answer is stored in the accumulator, where it is converted to a 4 digit BCD number.
- The BCD number is written to Outputs 0070 - 0107.
- If an overflow occurs, the overflow Flag 777 will turn on Output 130.

GEK-90842

**External Fault Diagnosis (F20)**

This instruction provides a method of monitoring external devices for unwanted conditions such as switches that should not be on at the same time. When the instruction is programmed, a 4 digit BCD number is specified as a fault indication number. If the unwanted condition occurs, the 4 digit number will be displayed on the programmer data display. The 4 digit number can be a constant value or read from a group reference.

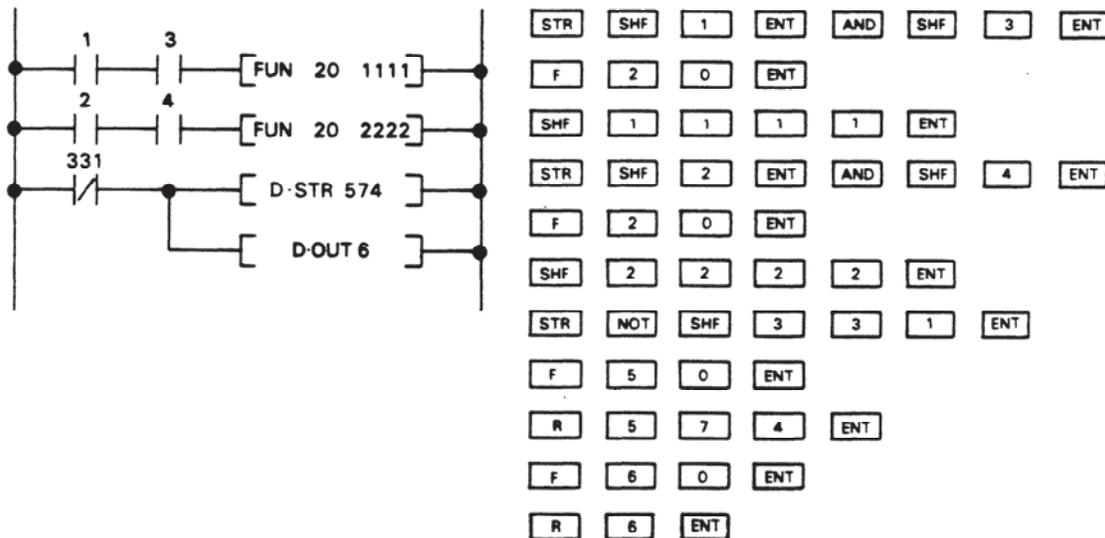
a40891



More than one FUN 20 instruction can be used in a program for detection of multiple fault conditions. Each condition to be monitored is assigned a different 4 digit number. If multiple faults occur simultaneously, the number displayed will be the one assigned by the last FUN 20 in the program. Data Registers 574 and 575 will contain the fault indicator number. When a fault number is displayed on the programmer, the fault number can be cleared from the display by depressing the CLR key. However, it is recommended that the actual fault be fixed before clearing the display.

• Programming example, FUN 20

a40892



- If devices connected to I1 and I3 are made at the same time, FUN 20 will be active, 1111 will be displayed on the data display on the programmer.
- If devices connected to I2 and I4 are made at the same time the data display will display 2222.
- The indicator number will be stored in 574 and 575 and that number will be written to Outputs 0060-0077.
- Fault indicator number display



- The physical fault should be cleared. Then clear the programmer display with the CLR key.

