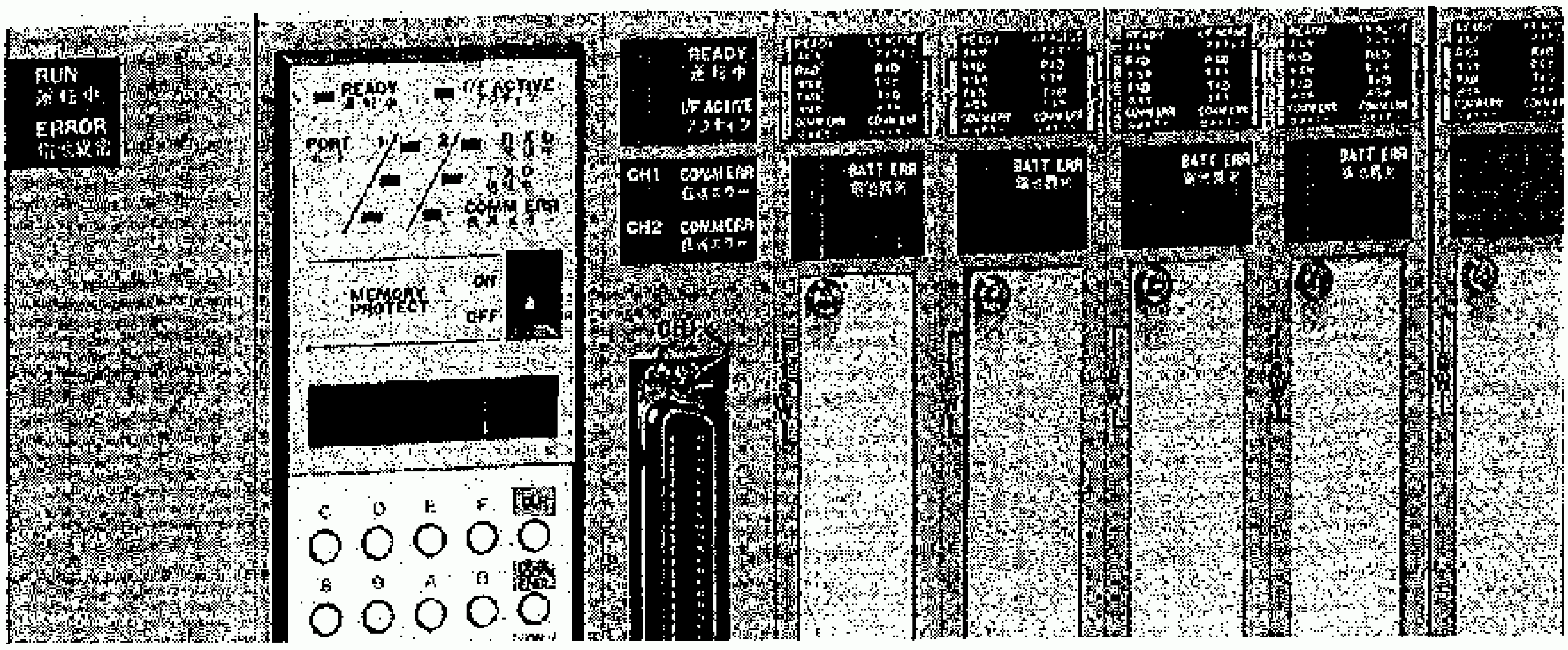
YASKAWA

MEMOCON-SC U84
DESCRIPTIVE INFORMATION

PROGRAMMABLE CONTROLLER
DESIGN AND MAINTENANCE



YASKAWA

MANUAL NO. SIE-C815-10.1F

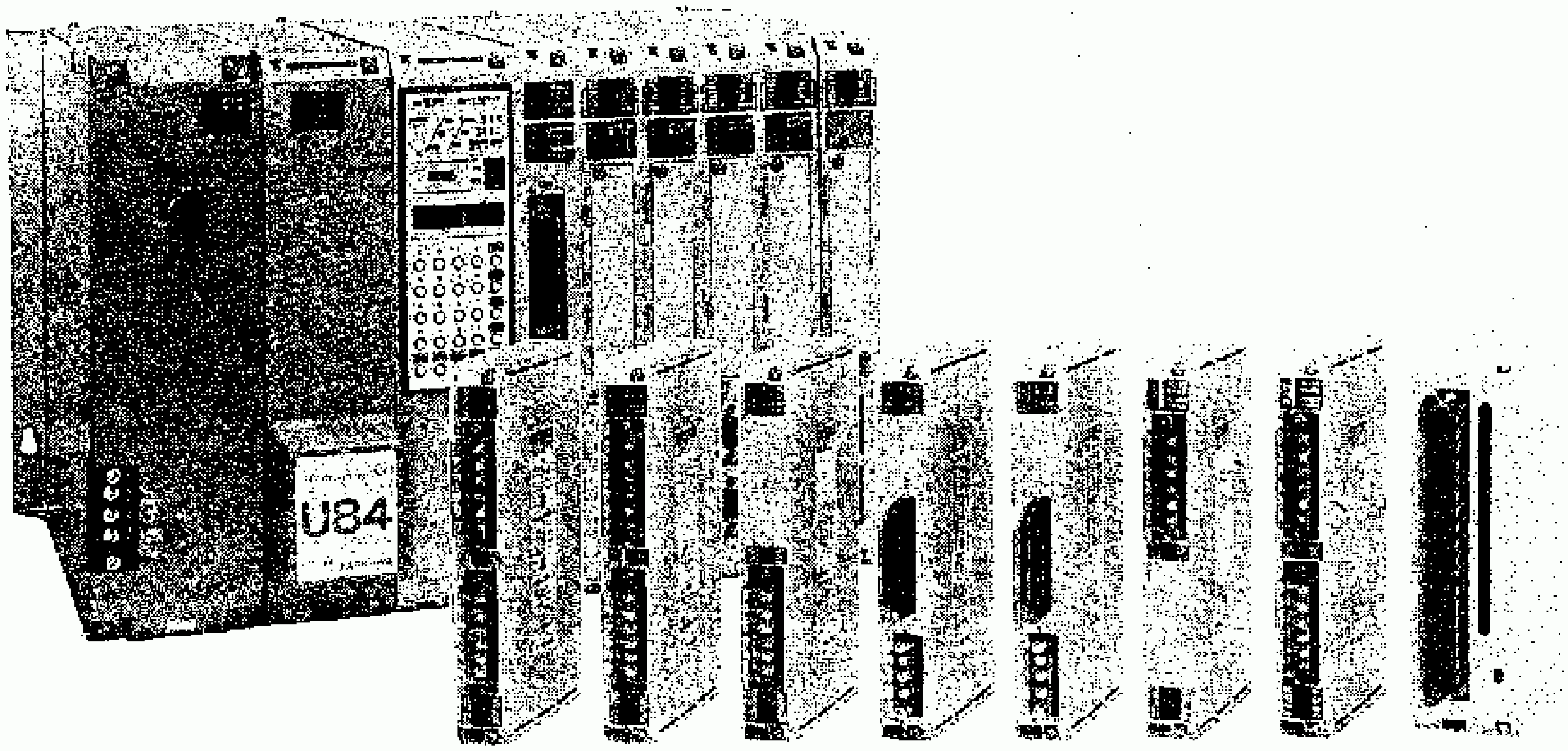
This manual, in the former part, provides the designer with the detailed information needed to design control systems utilizing the Memocon-SC U84 Controller.

In Sections 6 and 7, the manual describes as applications and instructions for handling and maintenance of the Memocon-SC U84 Controller.

The designer need not be familiar with digital equipment or computer technology to use this manual. The simple and versatile Ladder Line concept enables the designer to make use of Memocon-SC U84 Controller available functions, and to communicate his design easily to plant technicians.

List of standard auxiliary support equipment, designed to expand the power and versatility of the Controller system, is provided in Appendix A. The U84 dimensions are shown in Appendix B, and its layout and drilling plan in Appendix C.

584-202



584-202

Memocon-SC U84 and I/O Modules

CONTENTS

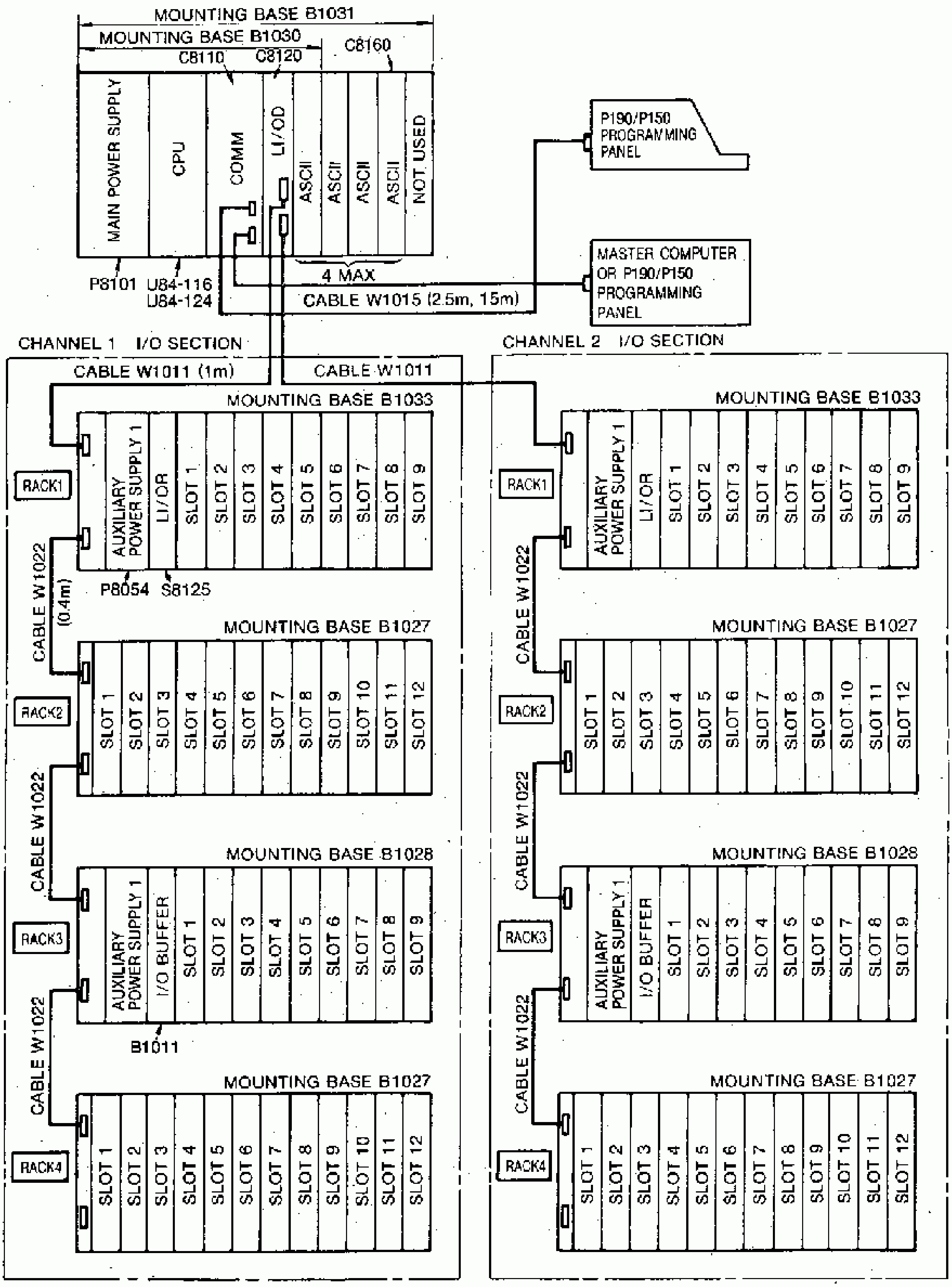
INDEX

INDEX (Cont'd)

SECTION 1
U84 CONFIGURATION

Table 1.1 U84 Configuration

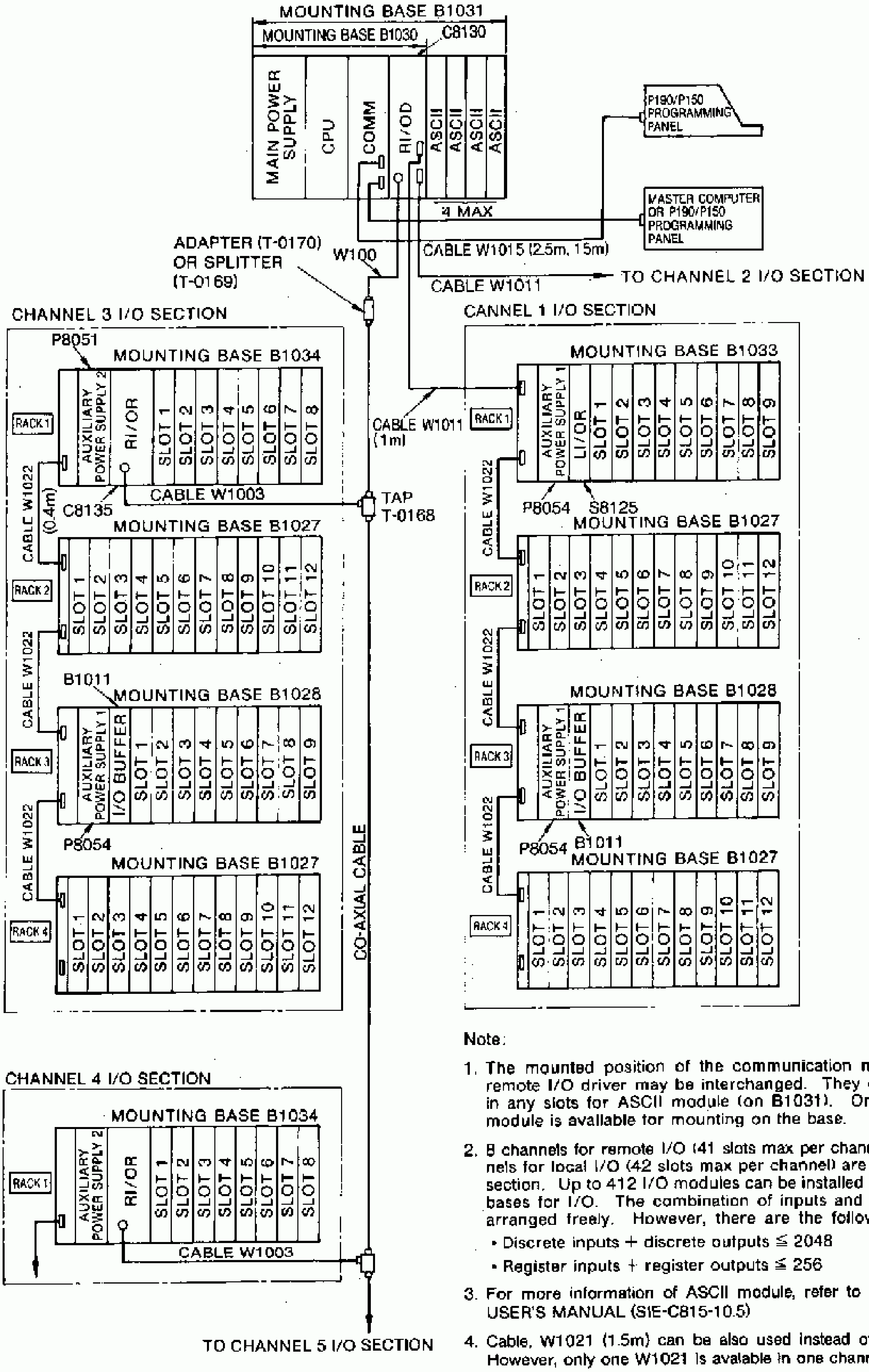
Component	Description
CPU Module (CPU)	The CPU module includes a logic solver and memory. The ladder logics are stored in the memory and solved according to input data sent from a local or remote I/O driver. The results are output to the local or remote I/O driver. The program memory is available in two types, 24k and 16k words. Regardless of the type of program memory, the U84 is capable of dealing with discrete inputs/outputs (ON/OFF signals) of up to 2,048 points and up to 256 register inputs/outputs (4-digit decimal or 16-bit binary data).
Power Supply Modules	The main power supply module provides DC power to the CPU module and peripheral modules including the communication module and local or remote I/O driver. The auxiliary power supply module 1 provides DC power to local I/O receiver, I/O buffer, and I/O modules. The auxiliary power supply module 2 provides DC power to remote I/O receiver and I/O modules.
Communication Module (COMM)	The communication module includes two RS-232C ports (MEMOBUS) for communication with the Programming Panel and a computer. By operating the front Register Access Panel (RAP), it is possible to display the status of the coils and input relays, to perform simulation (forced ON/OFF), to display and alter the contents of the registers, and to set and display communication parameters.
Local I/O Driver (LI/OD)	Receiving the reference from the CPU module, the local I/O driver transfers input data from the input module to the CPU module and output data from the CPU module to the output module for local I/O. The driver is used for two local I/O modules
Remote I/O Driver (RI/OD)	Receiving the reference from the CPU module, the remote I/O driver transfers input data from the input module to the CPU module and output data from the CPU module to the output module for remote I/O. The driver is used for eight remote I/O channels and two local I/O channels.
Local I/O Receiver (LI/OR)	The local I/O receiver is used for local I/O channels, and it exchanges signals between the local or remote I/O driver and the I/O modules.
Remote I/O Receiver (RI/OR)	The remote I/O receiver is used for remote I/O channels, and it exchanges signals between the remote I/O receiver and the I/O modules.
I/O Buffer Module	The I/O buffer module is used to install 22 or more I/O modules in one local I/O channel or 21 or more I/O modules in one remote I/O channel.
I/O Modules (1000 Series)	• Discrete signal modules One module is provided with inputs or outputs of 16 or 32 circuits. It is usable for numeric signals (by I/O allocation). • Numeric signal modules One module is provided with eight numeric inputs or outputs of 4-BCD digits or 16 bits in binary form (by I/O allocation). • Analog modules An A/D converter module has four circuits and a D/A converter module of two circuits. • Other modules Counter module, PID module, positioning module
Mounting Base	The CPU module, power supply module, peripheral modules, and I/O modules are mounted on a mounting base. The type of the mounting base varies with the type of module. The modules mounted on the base are connected to each other via a built-in mother board. Connections between mounting bases are made with cables.
Programming Panel	The programming panel permits storing a program, altering or deleting the stored program, monitoring status, and printing out a ladder diagram through a connected printer. Two types are available on request, CRT or plasma display.



Note:

1. The mounted position of the communication module and the local I/O driver may be interchanged. They can be installed in any slots for ASCII module. Only one of each module is available for mounting on the base.
2. 2 channels for local I/O (42 slots max per channel) are available in I/O section. Up to 84 I/O modules can be installed in the mounting bases for I/O. The combination of inputs and outputs can be arranged freely. However, there are the following limitations:
   - Discrete inputs + discrete outputs ≦ 2048
   - Register inputs + register outputs ≦ 256
3. For more information of ASCII module, refer to ASCII MODULE USER'S MANUAL (SIE-C815-10.5)
4. Cable, W1021 (1.5m) can be also used instead of W1022 (0.4m). However, only one W1021 is available in one channel.

Fig. 1.1 U84 Configurations



Note:

1. The mounted position of the communication module and the remote I/O driver may be interchanged. They can be installed in any slots for ASCII module (on B1031). Only one of each module is available for mounting on the base.
2. 8 channels for remote I/O (41 slots max per channel) and 2 channels for local I/O (42 slots max per channel) are available in I/O section. Up to 412 I/O modules can be installed in the mounting bases for I/O. The combination of inputs and outputs can be arranged freely. However, there are the following limitations:
   - Discrete inputs + discrete outputs ≦ 2048
   - Register inputs + register outputs ≦ 256
3. For more information of ASCII module, refer to ASCII MODULE USER'S MANUAL (SIE-C815-10.5)
4. Cable, W1021 (1.5m) can be also used instead of W1022 (0.4m). However, only one W1021 is avalable in one channel.

Fig. 1.2 U84 Configuration (When Remote I/O used)

SECTION 2
U84 SPECIFICATIONS

2.1 BASIC U84 SPECIFICATIONS

Table 2.1 Basic U84 Specifications

Items	Specifications
Power Supply	Single-phase 85 to 121 VAC, 47.5 to 63 Hz
Dissipated Power	280 VA (main power supply module)
Holding Time	10 ms
Ambient Temperature	0 to +55°C (excluding peripheral devices)
Storage Temperature	−20°C to +85°C (excluding lithium battery)
Humidity	5 % to 95 % relative (non-condensing)
Vibration-Resistance	In compliance with JIS* C 0911 (excluding peripheral devices)
Shock-Resistance	10 G max (excluding peripheral devices)
Environmental Condition	Free from explosive, inflammable, corrosive gases
Grounding	Grounding resistance: 100 Ω or less
Dielectric Strength	1500 VAC for 1 minute
Insulation Resistance	100 MΩ or more at 500 VDC
Noise Immunity	1500 Vp-p, pulse width: 1 μs, rising time: 1 ns

* Japanese Industrial Standard

2.2 CPU MODULE

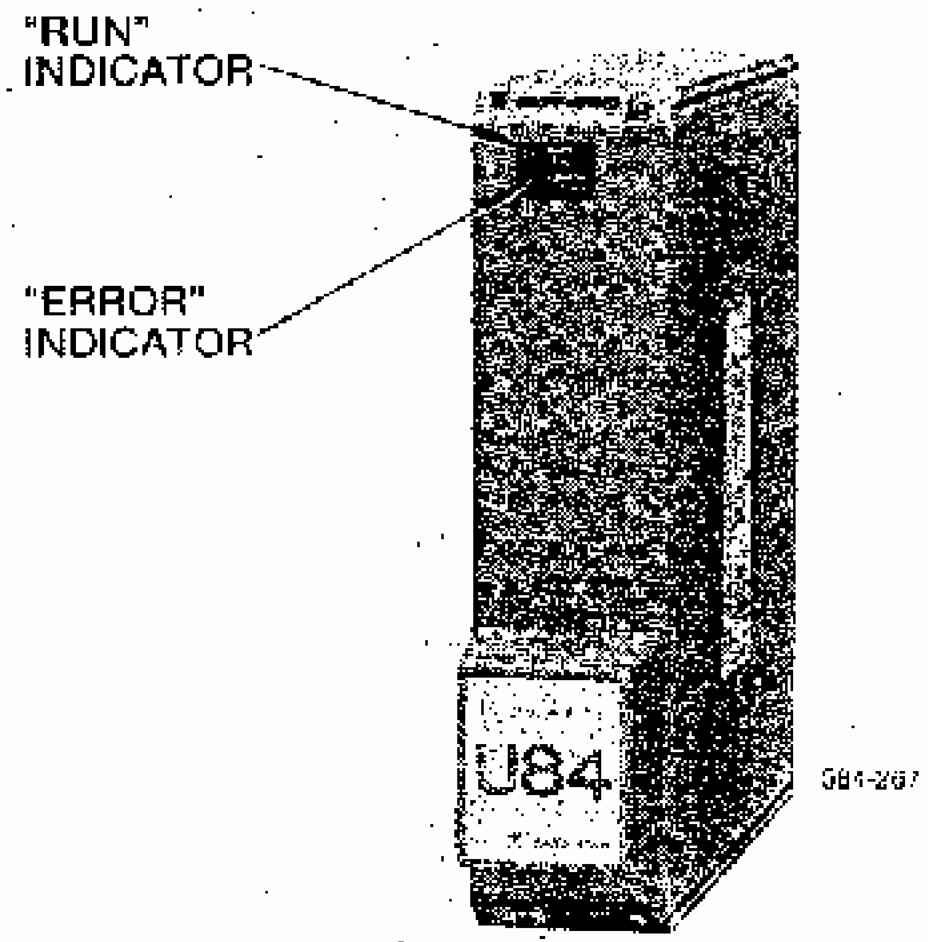


Fig. 2.1 CPU Module

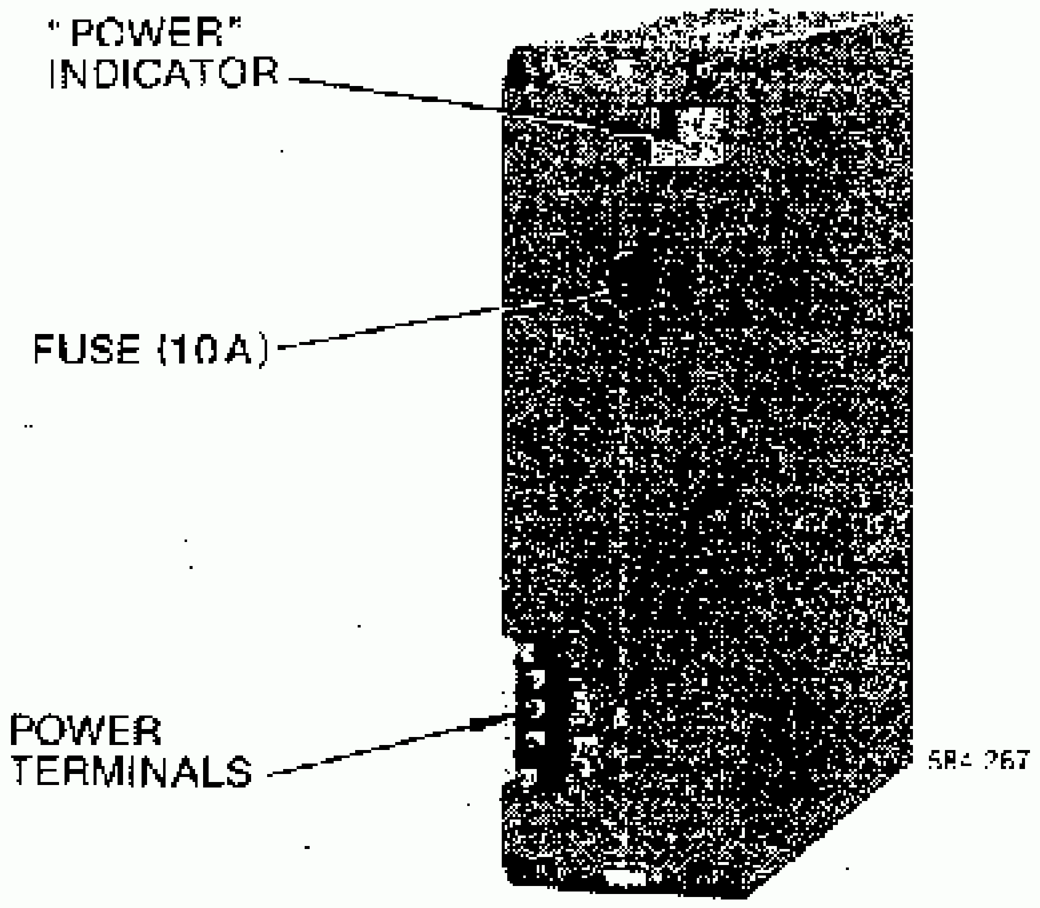
2. 2 CPU MODULE (Cont'd)

Table 2. 2 CPU Module Specifications

Items		Specifications
Type		• DDSCR-U84-116, 16 k words • DDSCR-U84-124, 24 k words
Control Method		Stored program and scan control
Programming		Relay ladder diagram symbology
Program Memory Size		16 k, 24 k CMOS RAM with battery back-up (24-bit per word)
Data Memory Size		9999 words, CMOS RAM with battery back-up (16 bit per word)
Scan Time		• 20 ms per 16 k words (typical) • 0.35 μs per word (basic instruction)
Logic Function	Relay	• Normally open contact, normally closed contact • Transitional contact (OFF to ON), or (ON to OFF) • Horizontal shunt, vertical shunt, vertical open • Coil, latched coil
	Timer	• Type: Seconds, tenths of seconds, hundredths of seconds • Maximum preset value: 4-digit decimal • Setting available from external device
	Counter	• Up counter, down counter • Maximum preset value: 4-digit decimal • Setting available from external device
	Arithmetic	• Addition, subtraction, multiply, divide (Arithmetic in 4-digit decimal)
	Move	R → T, T → R, T → T, BLKM, FIN, FOUT, SRCH, STAT
	Matrix	AND, OR, XOR, COMP, CMPR, MBIT, SENS, BROT
	Special Function	READ, WRITE (for ASCII module), Skip
	Square Arithmetic	Square root (SQRT), Double square root (DSQR)
	Double-precision Arithmetic	• Double-precision addition (DADD), Double-precision subtraction (DSUB), Double-precision multiply (DMUL), Double-precision divide (DDIV) • Arithmetic in 8 digit decimal
	Move with Index	• DIBT, DIBR • SIBT, SIBR
	Others	MRCT, TWST
Input/Output Points		• Discrete I/O points: Input + Output ≤ 2048 points • Register I/O points: Input + Output ≤ 256 points • Number of local channels: 2 (42 I/O modules max in use per channel) • Number of remote channels: 8 (41 I/O modules max in use per channel)
Diagnostic Function		• Checksum of memory • Watchdog timer checking • Battery monitoring • Parity checking • Internal code checking • Reference number checking • RTC checking, I/O allocation checking • CPU diagnostic, memory diagnostic
Backed-up Memory		• Kind: 1 lithium battery • Battery life: 5 years, at 25°C • Memory contents holding time: 1 year, at 25°C
Indicating Lamp		• RUN: Lights when CPU module is proper in operation. • ERROR: Lights when the output voltage of CMOS RAM back-up battery is low level, with AC power supply turned on.
Mounting Location		On mounting base B1030 or B1031
Dimensions in mm		69.5 × 300 × 232
Approx Weight		2 kg

2.3 POWER SUPPLY MODULES

(1) Main Power Supply Module



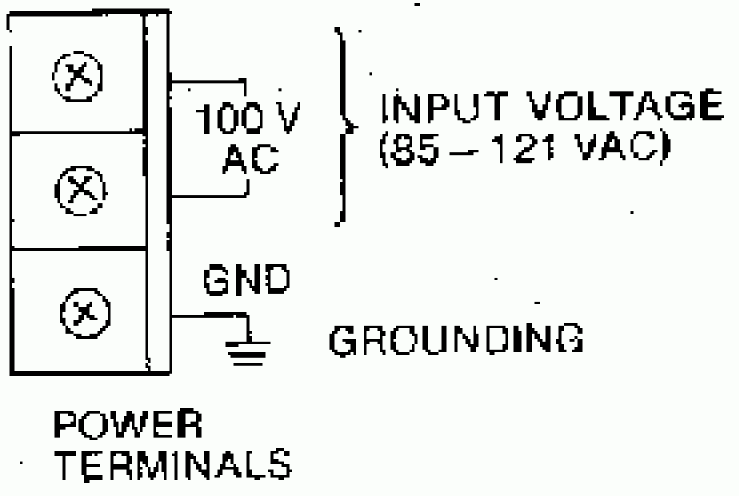
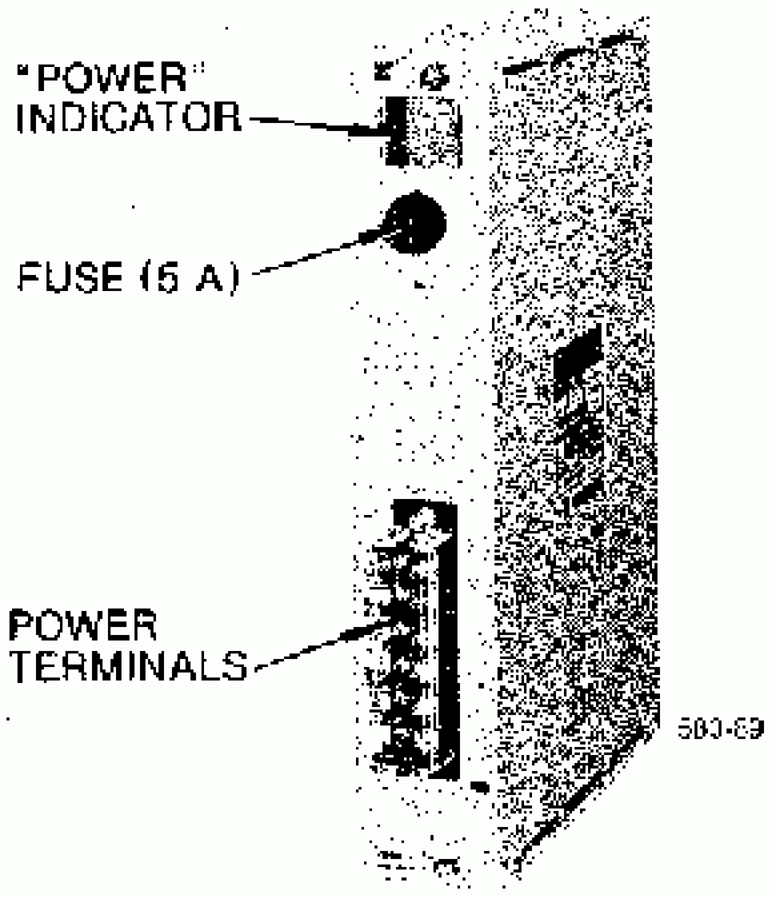


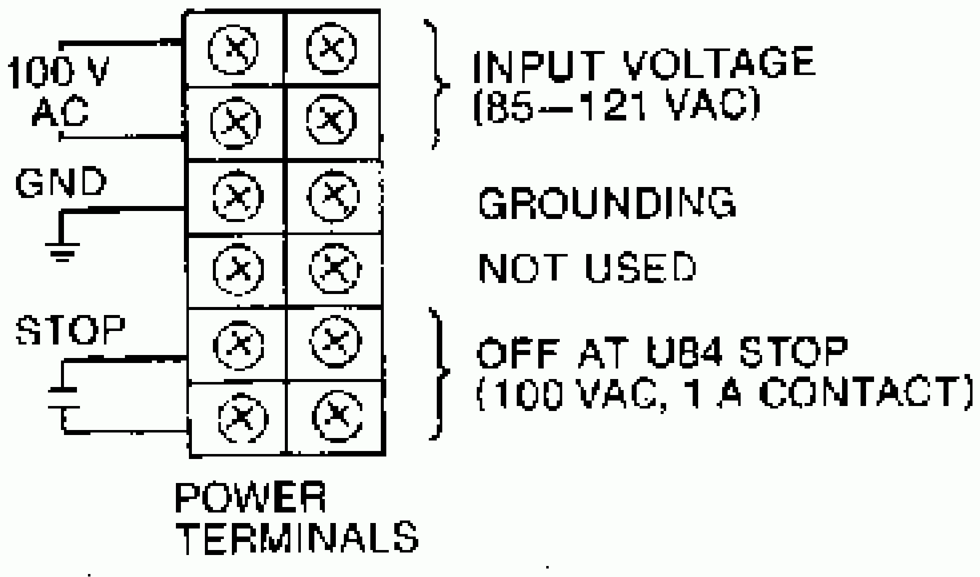
Fig. 2.2 Main Power Supply Module

Table 2.3 Main Power Supply Module Specifications

Items	Specifications
Type	JRMSP-P8101
Function	DC power supply for a CPU module, a communication module, a local or remote I/O driver module and four ASCII modules.
Input Voltage	Single-phase 85-121 VAC, 47.5-63 Hz, 280 VA
Transient Input Voltage	0-140 VAC (10 ms)
Inrush Current	50 A (peak) or less
Leakage Current	0.2 mA or less
Fuse	Glass tube fuse (10 A)
Indicating Lamp	POWER: Lights when power supply is proper.
Mounting Location	On mounting base B1030 or B1031
Dimensions in mm	90 (W) × 300 (H) × 232 (D)
Approx Weight	5 kg

(2) Auxiliary Power Supply Module 1





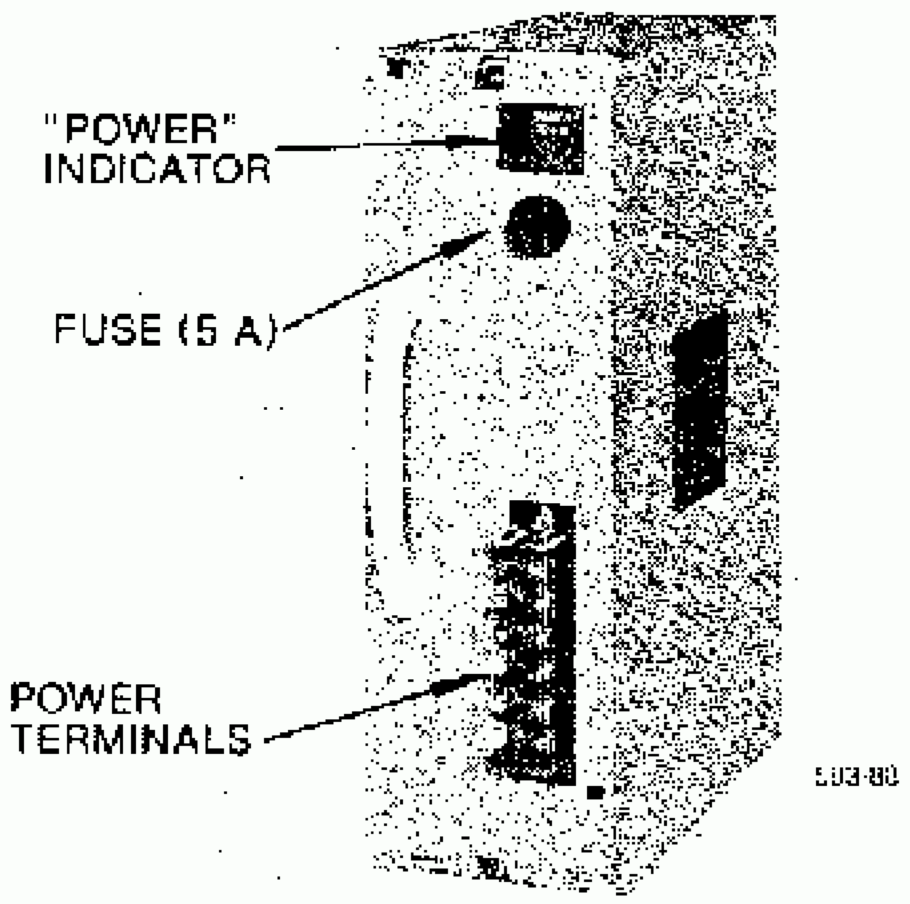
Note: Each designated pair of terminals are in common.

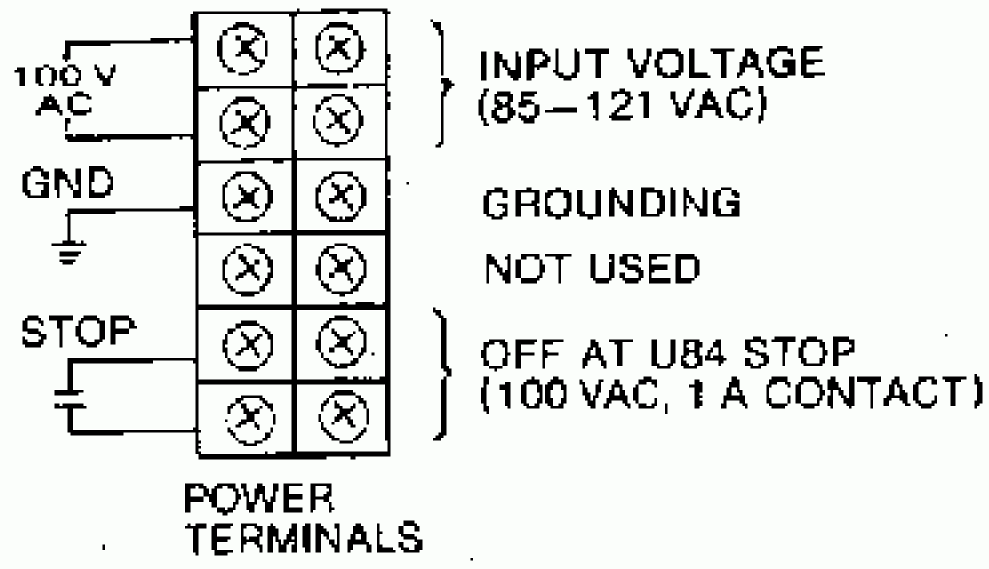
Fig. 2.3 Auxiliary Power Supply Module 1

Table 2.4 Auxiliary Power Supply Module 1 Specifications

Items	Specifications
Type	JRMSP-P8054
Function	DC power supply for local I/O receiver, I/O buffer module and I/O modules.
Input Voltage	Single-phase 85-121 VAC, 47.5-63 Hz, 90 VA
Transient Input Voltage	0-140 VAC (10 ms)
Inrush Current	30 A (peak) or less
Leakage Current	0.2 mA or less
Fuse	Glass tube fuse (5 A)
Indicating Lamp	POWER: Lights when power supply is proper.
Monitoring Contact	STOP: ON at U84 running, OFF at U84 stop (100 VAC, 1 A contact)
Mounting Location	On mounting bases B1033 and B1028
Dimensions in mm	45 (W) × 250 (H) × 160 (D)
Approx Weight	1.1 kg

(3) Auxiliary Power Supply Module 2





Note: Each designated pair of terminals are in common.

Fig. 2.4 Auxiliary Power Supply Module 2

Table 2.5 Auxiliary Power Supply Module 2 Specifications

Items	Specifications
Type	JRMSP-P8051
Function	DC power supply for remote I/O receiver and I/O modules.
Input Voltage	85-121 VAC, 47.5-63 Hz, 170 VA
Transient Input Voltage	0-140 VAC (10 ms)
Inrush Current	50 A (peak) or less
Leakage Current	0.2 mA or less
Fuse	Glass tube fuse (5 A)
Indicating Lamp	POWER: Lights when power supply is proper.
Monitoring Contact	STOP: ON at U84 running; OFF at U84 stop (100 VAC, 1 A contact)
Mounting Location	On mounting base B1034
Dimensions in mm	80 (W) × 250 (H) × 160 (D)
Approx Weight	2.8 kg

2.4 COMMUNICATION MODULE (COMM)

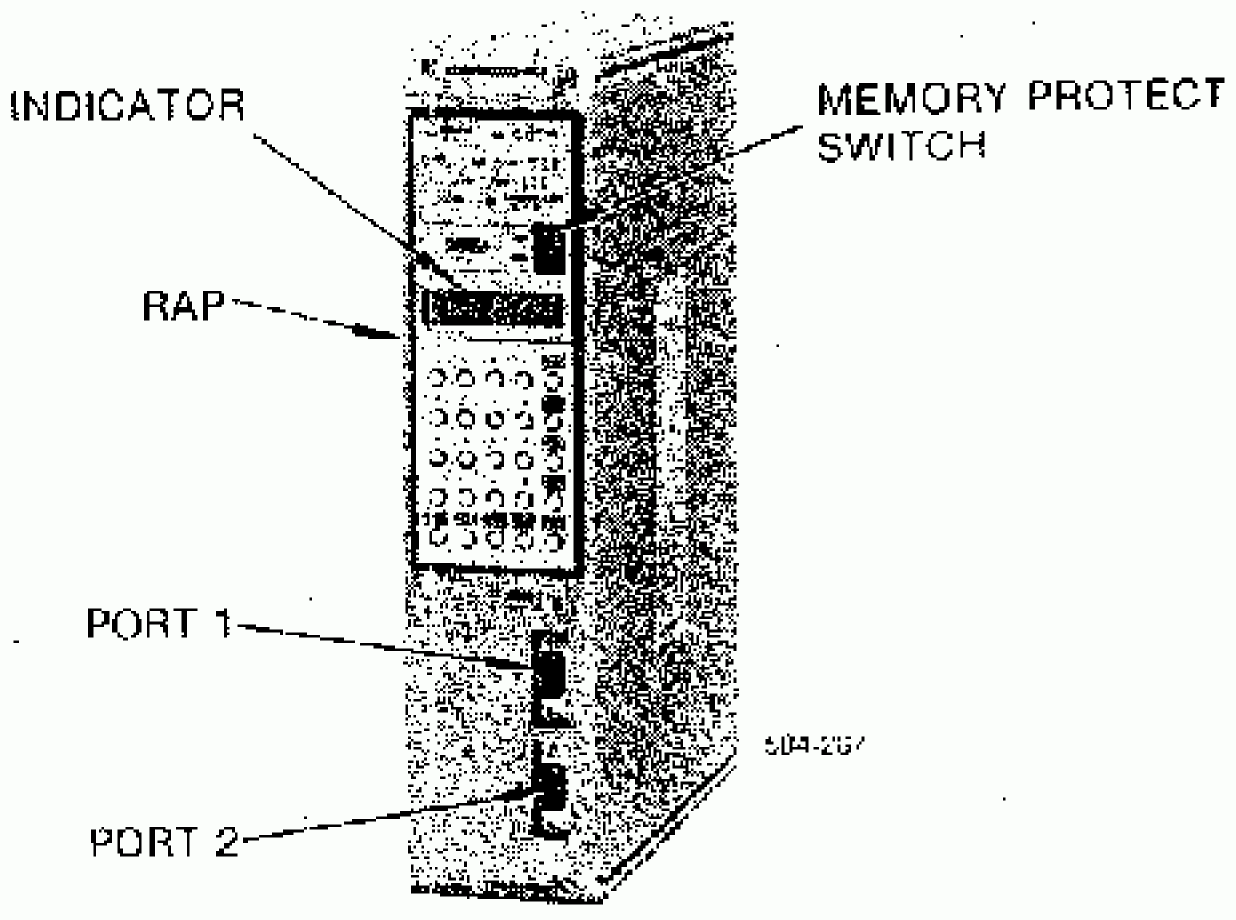


Fig. 2.5 Communication Module

Table 2.6 Communication Module Specifications

Items		Specifications
Type		JAMSC-C8110
Function		• For P190 CRT programming panel and MEMOBUS (2 ports per module) • Discrete I/O status indication, disable operation, register contents indication, set, and alternation (with register access panel)
Communication Port	No. of Ports	2 ports per module
	Communication Specification	EIA RS-232C
	Baud Rate	19200/9600/4800/2400/1200/600/300/150
	Data Bits	7 or 8 bits
	Parity	Even, odd or non
	Stop Bits	1 or 2 bits
	Protocol	MEMOBUS protocol
	Transmission Check	CRC-16 or LRC (checksum)
	Connector	D-SUB 9 pin
Mounting Location		On mounting bases B1030 or B1031
Dimensions in mm		69.5 (W) × 300 (H) × 232 (D)
Approx Weight		1.4 kg

2.5 LOCAL I/O DRIVER MODULE (LI/OD)

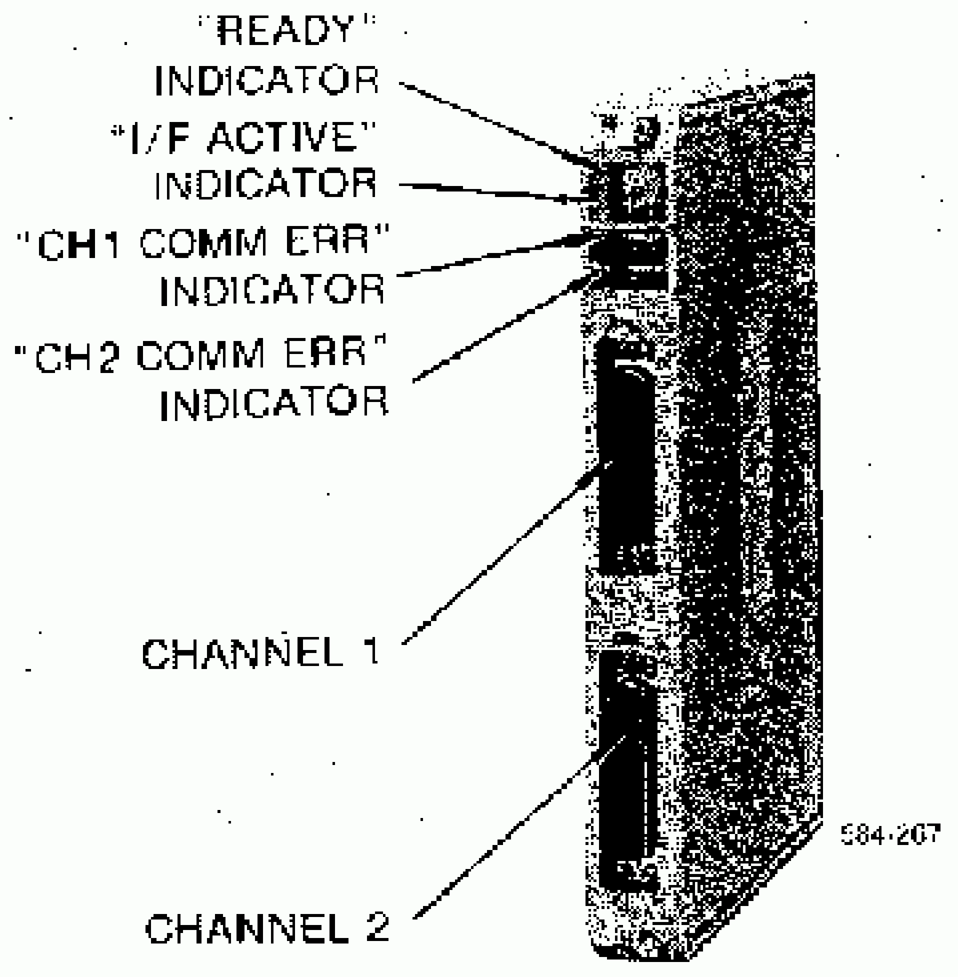


Fig. 2.6 Local I/O Driver Module

Table 2.7 Local I/O Driver Module Specifications

Items	Specifications
Type	JAMSC-C8120
Function	• Use for local I/O. • For connecting to two local I/O receiver modules and communicating I/O signals by a command from CPU module. • 2-local I/O receiver modules per module
Indicating Lamp	• READY: Lights when local I/O driver module is proper. • I/F ACTIVE: Lights when transmission from CPU module is proper. • CH1 COMM ERR: Lights when communication with channel 1 local I/O receiver is not proper. • CH2 COMM ERR: Lights when communication with channel 2 local I/O receiver is not proper.
Mounting Location	On mounting base B1030 or B1031
Dimensions in mm	34.5 (W) × 300 (H) × 232 (D)
Approx Weight	1.2 kg

2.6 ELECTRICAL REMOTE I/O DRIVER MODULE (RI/OD)

Table 2.8 Electrical Remote I/O Driver Module Specifications

Items	Specifications
Type	JAMSC-C8130
Function	For driving both of local I/O and remote I/O • Local I/O: Communicates with up to 2 local I/O receiver modules, by references from CPU module. • Remote I/O: Communicates with up to 8 remote I/O receiver modules, by references from CPU module.
Indicating Lamp	• REMOTE: Lights when remote I/O channel is ready to use. • READY: Lights when remote I/O driver module is proper. • I/F ACTIVE: Lights when transmission from CPU module is proper. • RXD: Lights when data from remote I/O receiver module is received. • TXD: Lights when data is sent to remote I/O receiver module. • REMOTE ERR: Lights when communication with remote I/O receiver module is not proper. • CH1 ERR: Lights when communication with channel 1 local I/O receiver module is not proper. • CH2 ERR: Lights when communication with channel 2 local I/O receiver module is not proper.
Mounting Location	On mounting base B1030 or B1031
Dimensions in mm	69.5 (W) × 300 (H) × 232 (D)
Approx Weight	2.2 kg

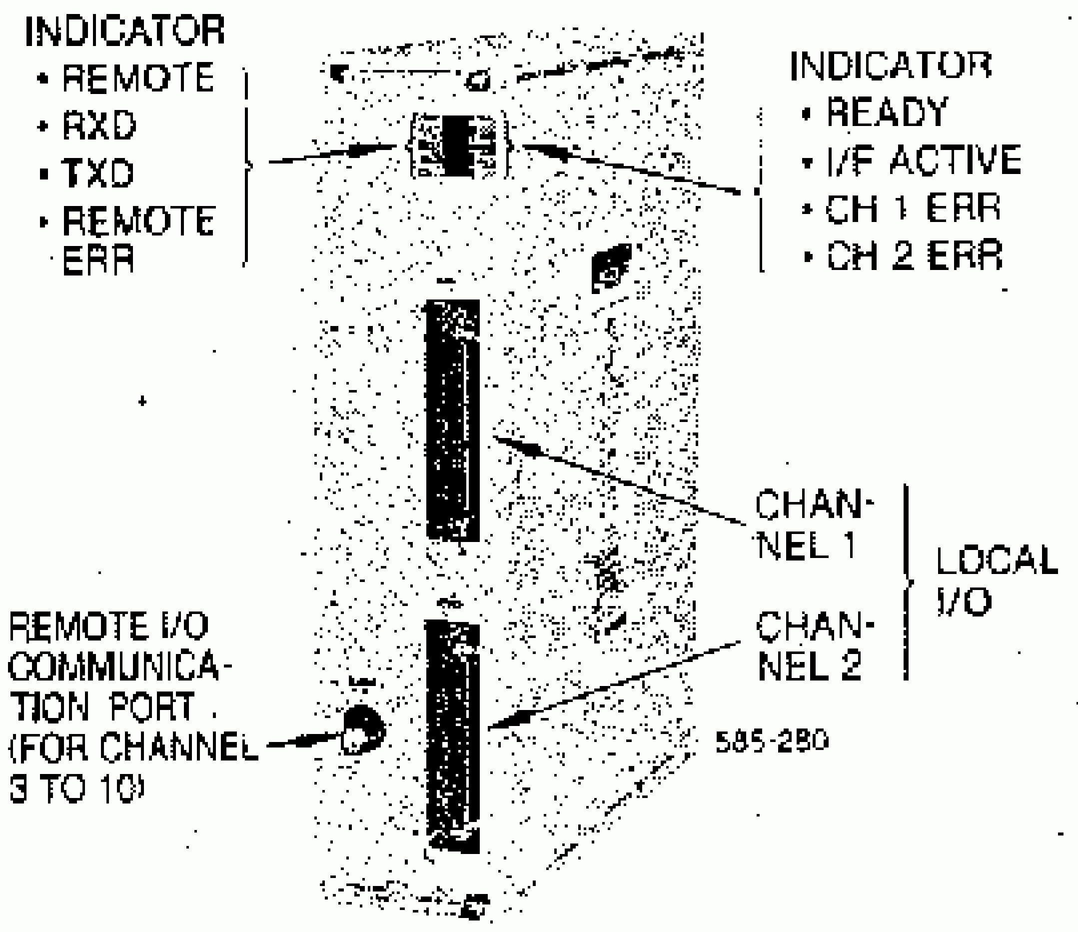


Fig. 2.7 Electrical Remote I/O Driver Module

2.7 OPTICAL REMOTE I/O DRIVER MODULE (RI/OD)

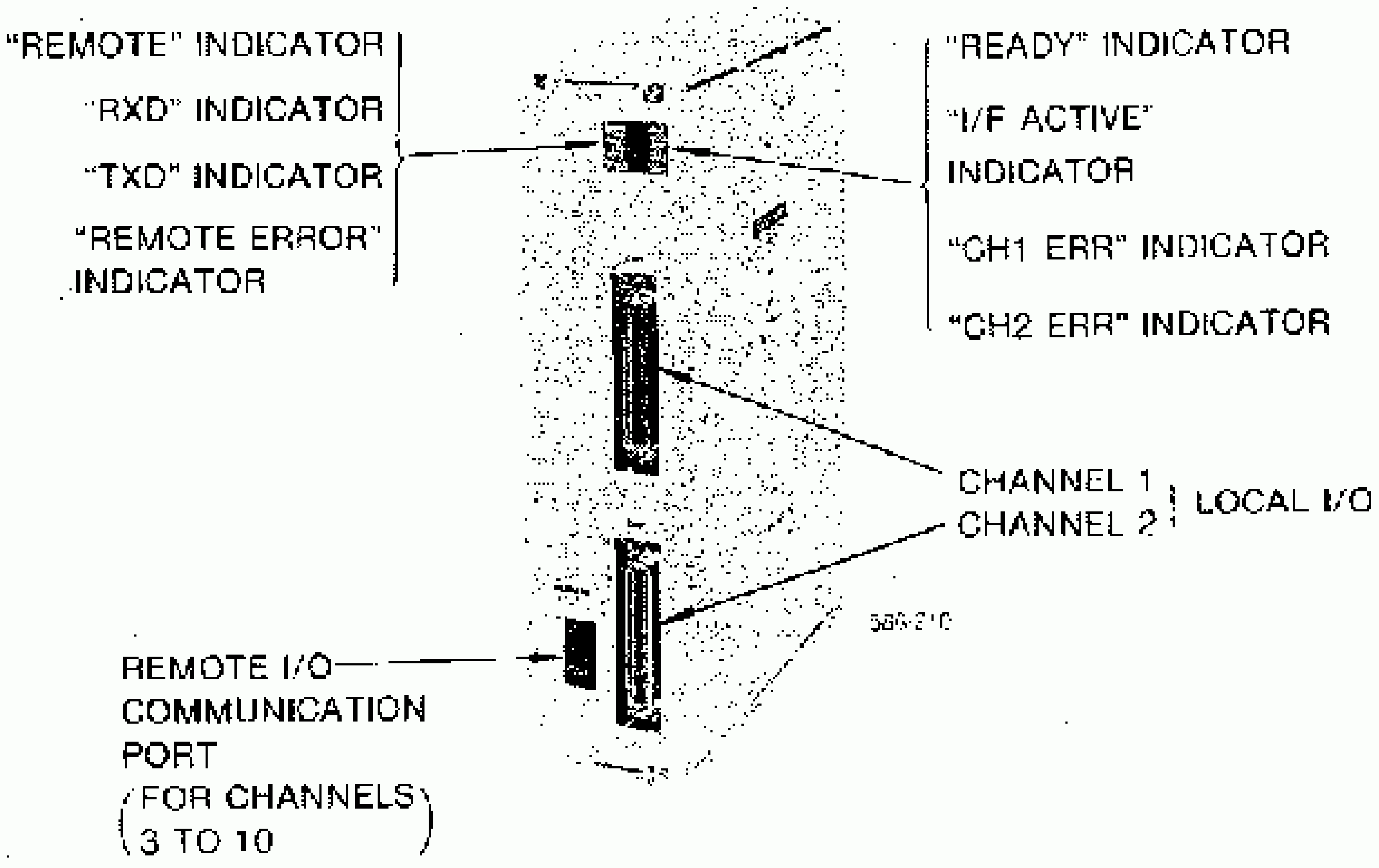


Fig. 2.8 Optical Remote I/O Driver Module

Table 2.9 Optical Remote I/O Driver Module Specifications

Items	Specifications
Type	JAMSC-C8140
Function	For driving both of local I/O and remote I/O • Local I/O: Communicates with up to 2 local I/O receiver modules, by references from CPU module. • Remote I/O: Communicates with up to 8 remote I/O receiver modules, by references from CPU module.
Indicating Lamp	• REMOTE: Lights when remote I/O channel is ready to use. • READY: Lights when remote I/O driver module is proper. • I/F: ACTIVE: Lights when transmission from CPU module is proper. • RXD: Lights when data from remote I/O receiver module is received. • TXD: Lights when data is sent to remote I/O receiver module. • REMOTE ERR: Lights when communication With remote I/O receiver module is not proper. • CH 1 ERR: Lights when communication with channel 1 local I/O receiver module is not proper. • CH 2 ERR: Lights when communication with channel 2 local I/O receiver module is not proper.
Mounting Location	On mounting base B1031 or B1030
Dimensions in mm	69.5(W) × 300(H) × 232(D)
Approx Weight	1.4kg

2.8 LOCAL I/O RECEIVER MODULE (LI/OR)

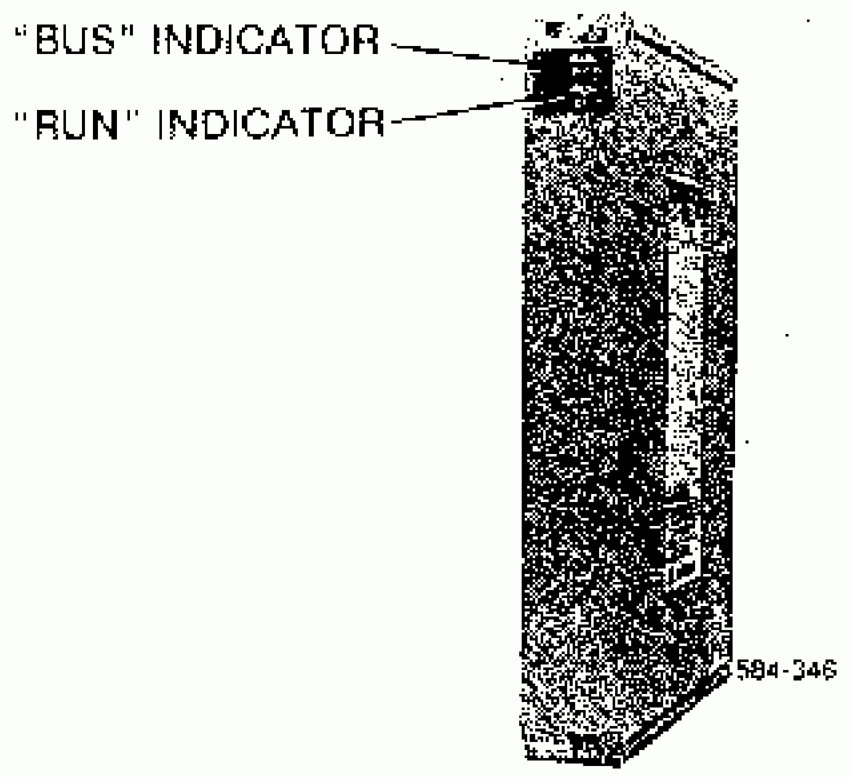


Fig. 2.9 Local I/O Receiver Module

Table 2.10 Local I/O Receiver Module Specifications

Items	Specifications
Type	JAMSC-S8125
Function	• Interface between local or remote I/O driver and I/O modules. • 42 I/O modules max
Indicating Lamp	• BUS: Lights when transmission from local I/O driver module is proper. • RUN: Lights when communication with I/O module is proper.
Mounting Location	On mounting base B1033
Dimensions in mm	34.5 (W) × 250 (H) × 160 (D)
Approx Weight	0.7 kg

2.9 ELECTRICAL REMOTE I/O RECEIVER MODULE (RI/OR)

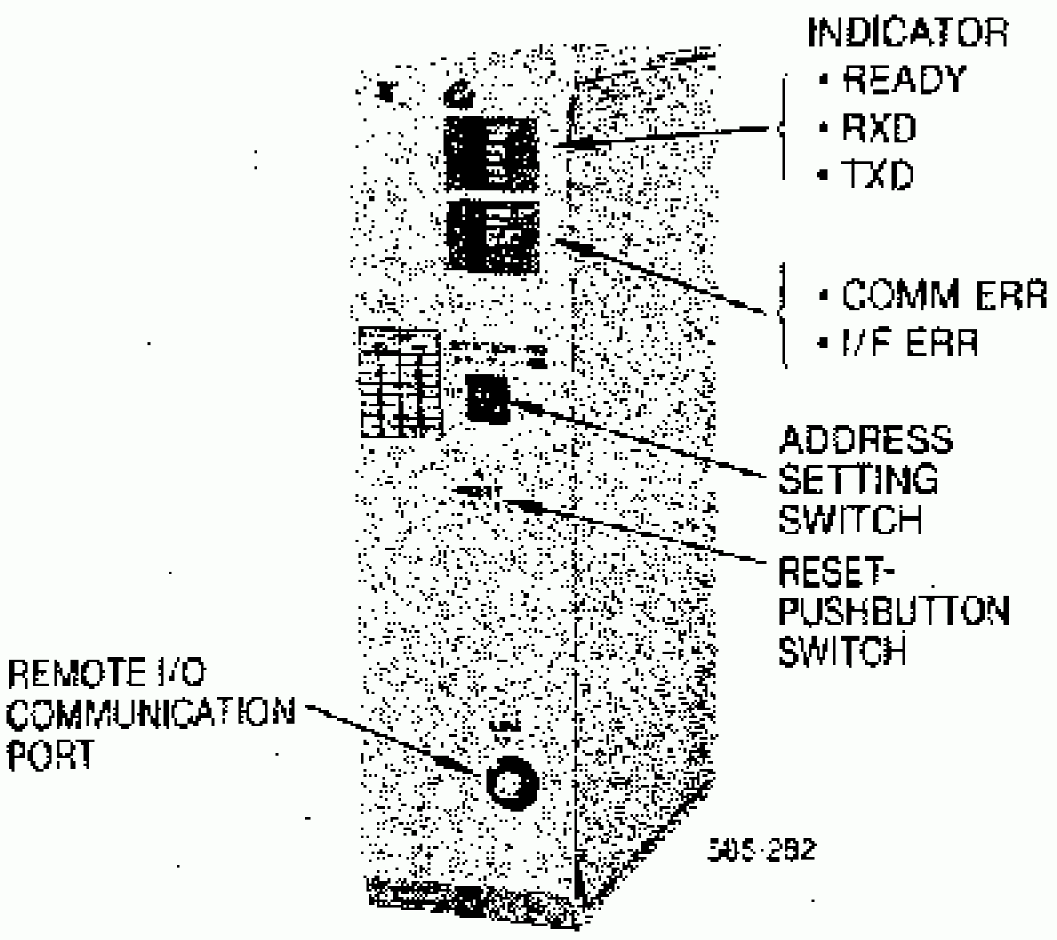


Fig. 2.10 Electrical Remote I/O Receiver Module

Table 2.11 Electrical Remote I/O Receiver Module Specifications

Items	Specifications
Type	JAMSC-C8135
Function	• Interface between remote I/O driver and I/O modules. • 41 I/O modules max
Indicating Lamp	• READY: Lights when remote I/O receiver is proper. • RXD: Lights when data from remote I/O driver module is received. • TXD: Lights when data is sent to remote I/O driver module. • COMM ERR: Lights when communication with remote I/O driver module is not proper. • I/F ERR: Lights when communication with I/O modules is not proper.
Address Setting Switch	Used to set station No. of remote I/O receiver module: Station No. Setting / I/O Channel No. 1 / 3 2 / 4 3 / 5 4 / 6 5 / 7 6 / 8 7 / 9 8 / 10 Ineffective setting: 0, 9, A, B, C, D, E, F
Reset-pushbutton Switch	Hard ware reset switch of remote I/O receiver module.
Mounting Location	On mounting base B1034
Dimensions in mm	65 (W) × 250 (H) × 160 (D)
Approx Weight	0.8 kg

2.10 OPTICAL REMOTE I/O RECEIVER MODULE (RI/OR)

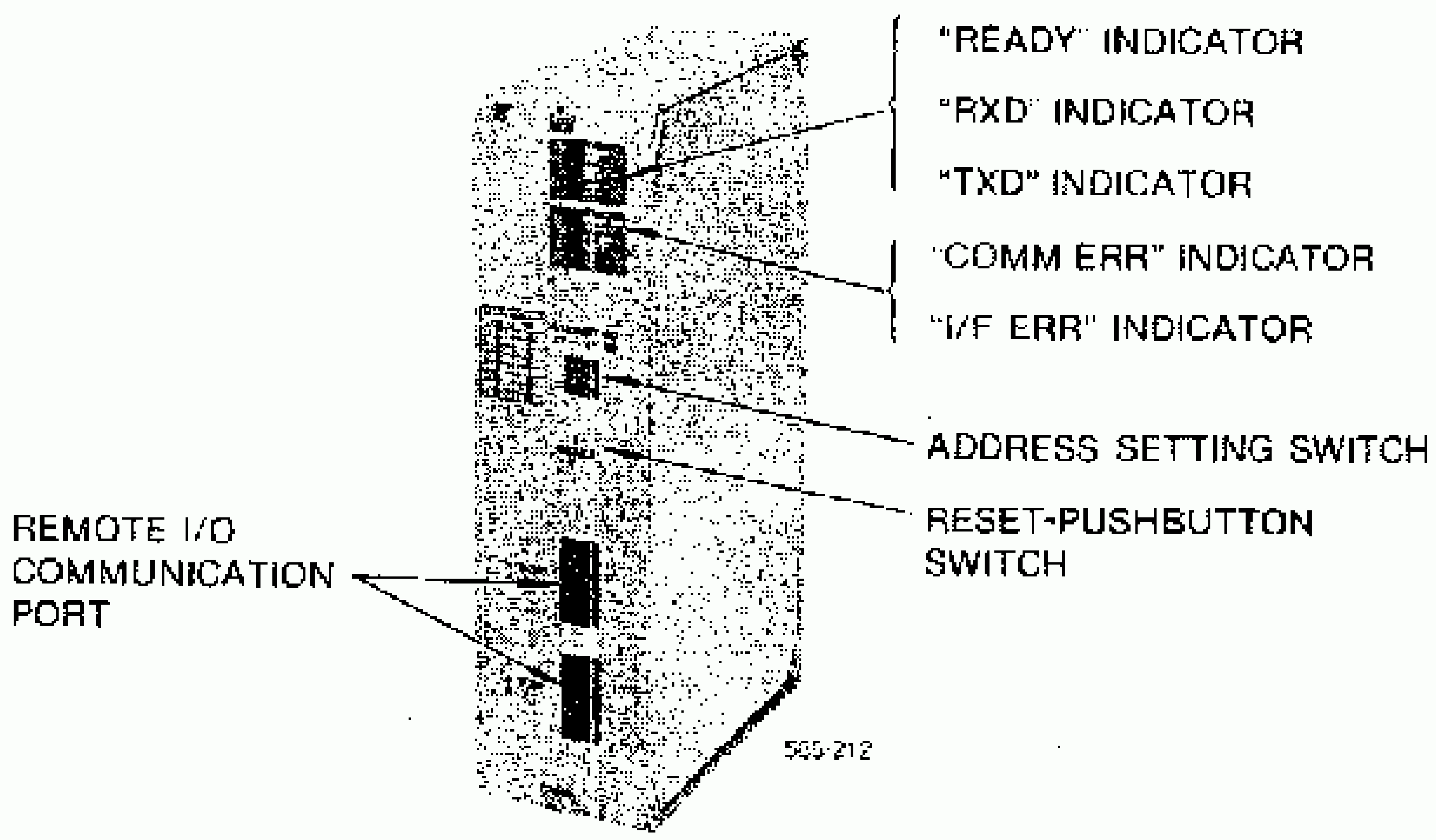


Fig. 2.11 Optical Remote I/O Receiver Module

Table 2.12 Optical Remote I/O Receiver Module Specifications

Items	Specifications	
Type	JAMSC-C8145	
Function	• Interface between remote I/O driver and I/O modules • 41 I/O modules max	
Indicating Lamp	• READY: Lights when remote I/O receiver is proper. • RXD: Lights when data from remote I/O driver module is received. • TXD: Lights when data is sent to remote I/O driver module. • COMM ERR: Lights when communication with remote I/O driver module is not proper. • I/F ERR: Lights when communication with I/O modules is not proper.	
Address Setting Switch	Used to set station No. of remote I/O receiver module	
	Station No. Setting	I/O Channel No
	1	3
	2	4
	3	5
	4	6
	5	7
	6	8
	7	9
	8	10
	Ineffective setting: 0, 9, A, B, C, D, E, F	
Reset-pushbutton Switch	Hardware reset switch of remote I/O receiver module	
Mounting Locatoin	On mounting base B1034	
Dimensions in mm	65(W) × 250(H) × 160(D)	
Approx Weight	0.8kg	

2.11 I/O BUFFER MODULE

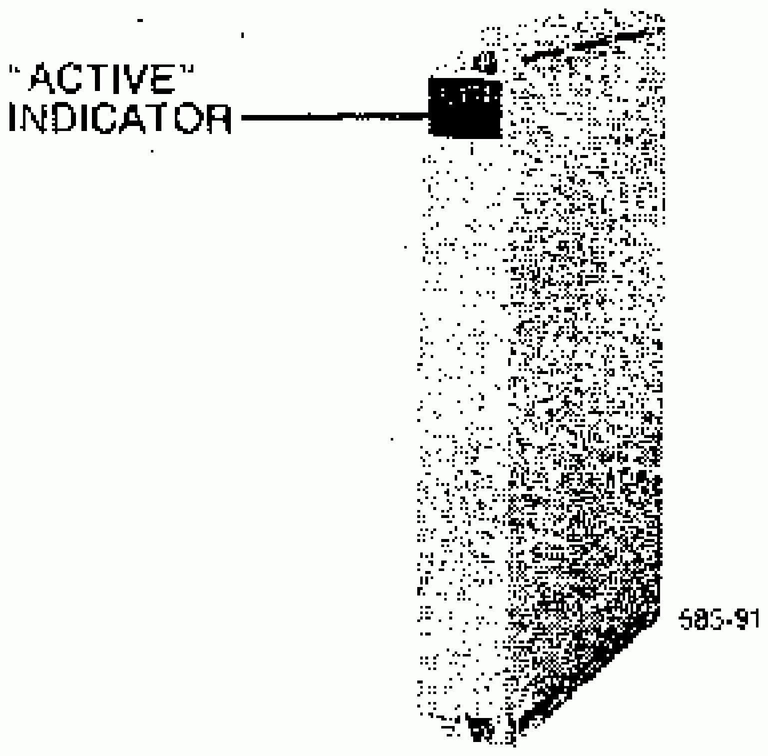


Fig. 2.12 I/O Buffer Module

Table 2.13 I/O Buffer Module Specifications

Items	Specifications
Type	JAMSC B1011
Function	• I/O bus buffer • When rack 3 or/and rack 4 is required due to numbers of I/O modules.
Indicating Lamp	ACTIVE: Lights (or flickers) when I/O signal transmission is proper.
Mounting Location	On mounting base B1028
Dimensions in mm	34.5 (W) × 250 (H) × 160 (D)
Approx Weight	0.7 kg

2.12 ASCII MODULE

Fig. 2.13 ASCII Module

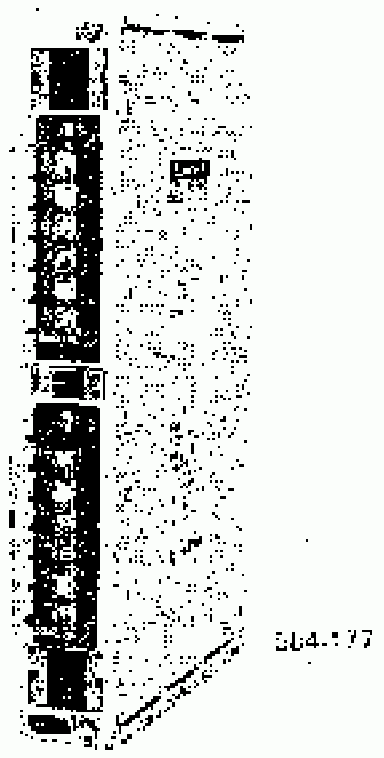
Table 2.14 ASCII Module Specifications

Items		Specifications
Type		JAMSC-C8160
Applicable PC		Memocon-SC U84
Number of Modules		4 Modules max connected
Memory Capacity		16 k words (32 k bytes) per module.
Memory Back-up		• One lithium battery • Battery life: 5 years at 25°C • Memory contents holding time: 1 year at 25°C
Message		• 512 messages max per module • 256 words (512 bytes) max per message
Number of Available Registers		512 registers max per message
ASCII Ports	Number of Ports	2 ports per module
	Transmission Mode	EIA RS-232C, half-duplex asynchronous
	Baud Rate	150/300/600/1200/2400/4800/9600/19200 bauds
	Data	5 to 8 bits
	Parity Check	Even parity, odd parity, no parity
	Stop Bits	1 or 2 bits
	Connector	Mini 9-pin D subconnector
Indicating Lamp	READY (Operating)	Lights when ASCII module operating normally.
	I/F ACTIVE (Active)	Lights during communication with CPU or communication module.
	RXD (Receiving)	Lights during receiving data via ODD or EVEN port.
	TXD (Transmitting)	Lights during transmitting data via ODD or EVEN port.
	COMM ERR (Communication Error)	Lights when error detected in data communication at ODD or EVEN port.
	BATT ERR (Battery Error)	Lights when memory back-up battery output voltage reduced (only when AC power supplied).
Diagnostic Function		• ROM checking • RAM checking • Checksum of memory • Watchdog timer checking • Battery voltage checking
Mounting Location		On mounting base B1031 (B1030*)
Dimensions in mm		34.5 (W) × 300 (H) × 232 (D)
Approx Weight		1 kg

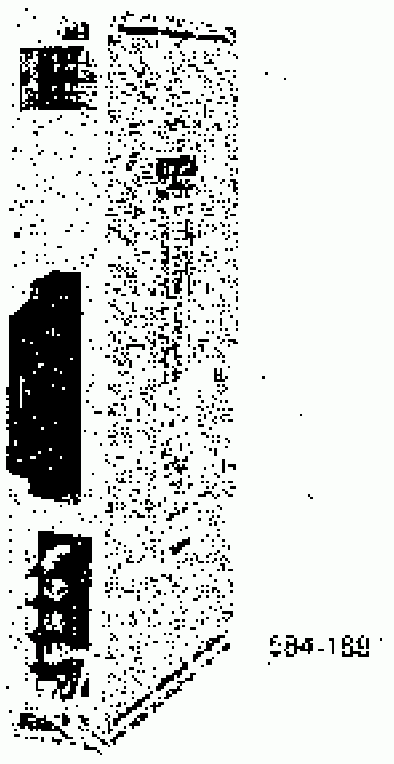
*Only one ASCII module can be mounted on this base when the local I/O driver is used.

2.13 I/O MODULES

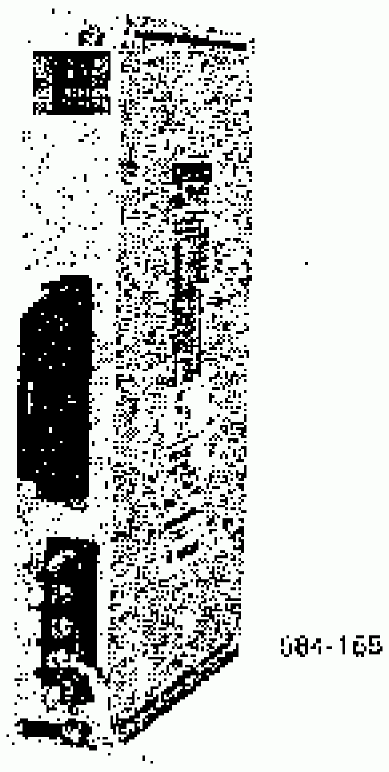
Input Module



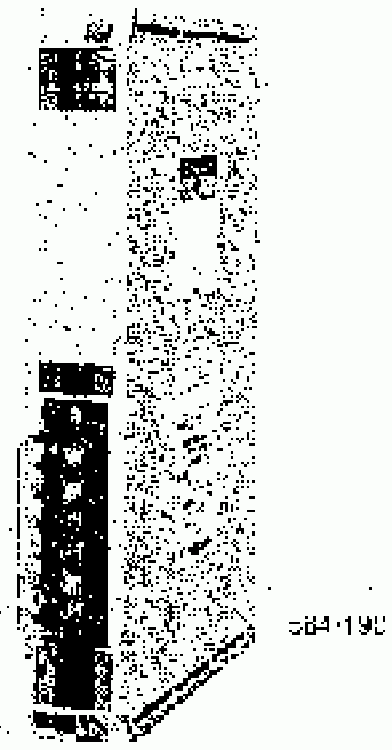
Output Module



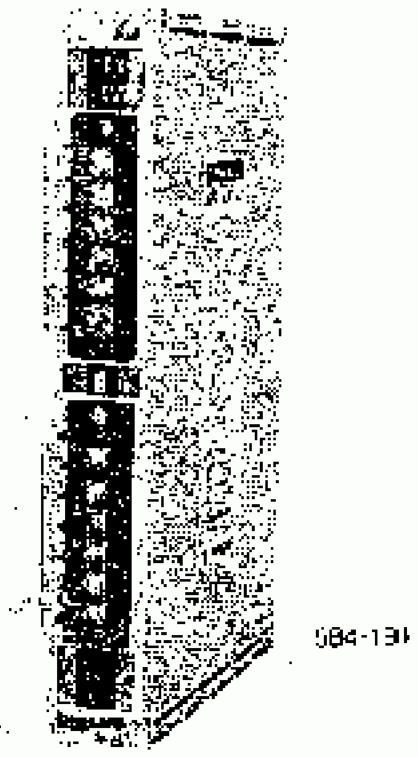
Register Input Module



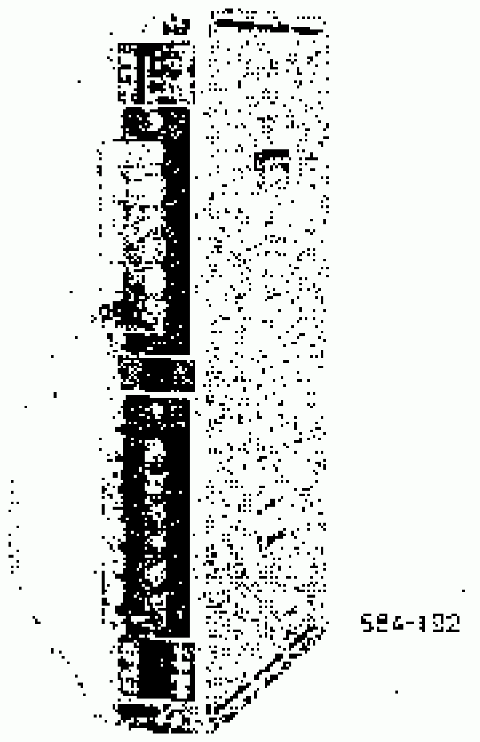
Register Output Module



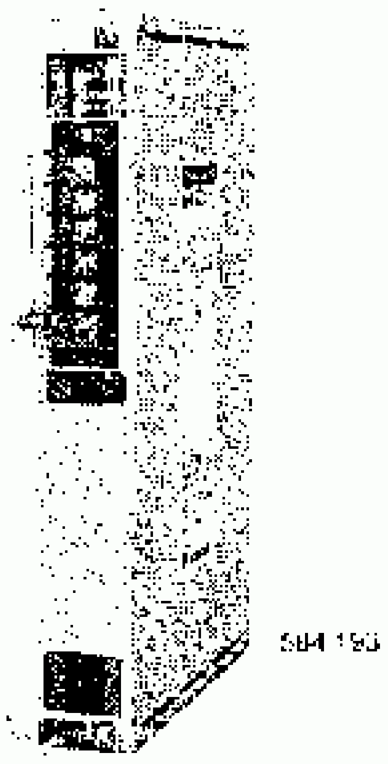
Analog Output Module



Analog Input Module



Preset Counter Module



Reversible Counter Module

Positioning Module

Fig. 2.14 I/O Modules

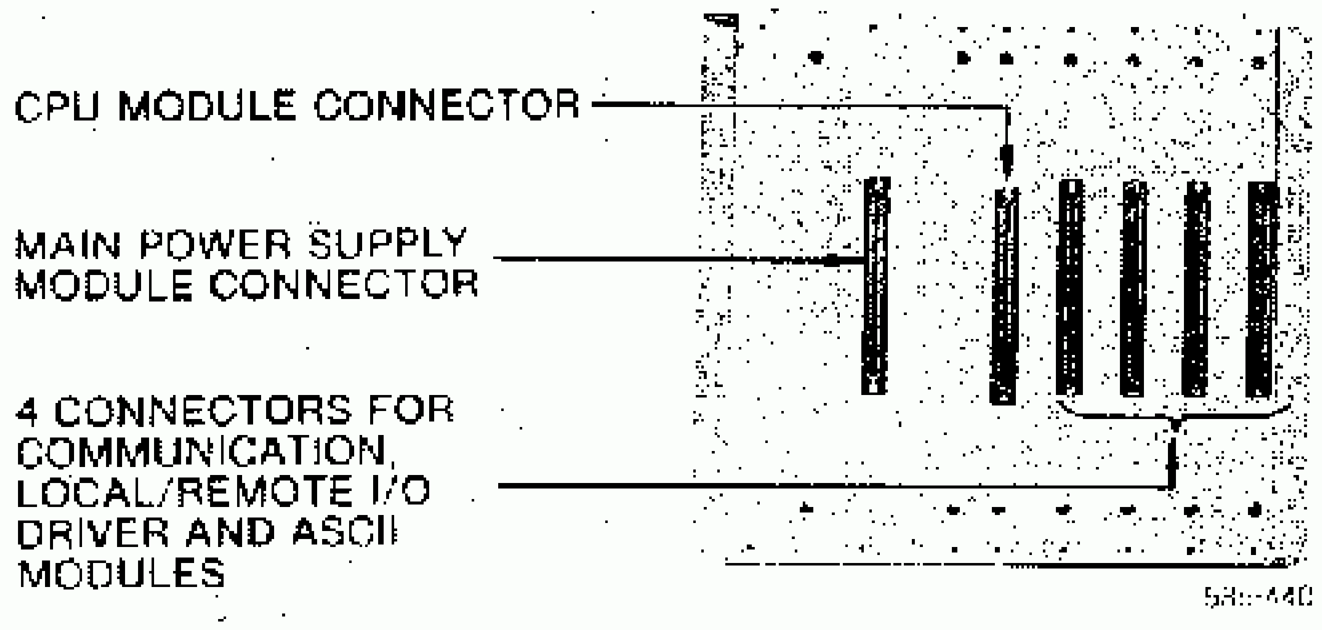
Table 2.15 I/O Module Specification

Modules			Type JAMSC-	Voltage	Current	Input Impedance	External Power Supply	Maximum Response Time	No. of I/O's	Module Span
Input	AC	100V	B1051B	80 to 130V	13mA	7.5 kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	16	1
Input	AC	200V	B1055	160V to 260V	6.5mA	30kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	16	1
Input	DC	5-12V	B1053	4.75 to 13.2V	5V: 4.5mA 12V: 11.5mA	1kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	16	1
Input	DC	48V	B1057	38 to 58V	10mA (48V)	4.8kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	16	1
Input	DC	24V	B1059C	19 to 29V	11mA (24V)	2.4kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	16	1
Input	DC	24V	B1061	20.4 to 26.4V	5mA (24V)	4.8kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	64	1
Input	DC	24V	B1063	20.4 to 26.4V	10mA (24V)	2.4kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	32	1
Input	DC	24V	B1065	19 to 29V	6mA (24V)	4kΩ	–	OFF → ON 10 ms ON → OFF 20 ms	32	1
Input	Register		B1071	–	–	12kΩ	External power: +24V at 200mA	64 ms	8	1
Input	Analog		B1073-1	0 to +10V	–	2MΩ	• Resolution: 1/1024 (10 bits) • External power: +15V 120mA, −15V 40mA.	–	4	1
Input	Analog		B1073-2	+1 to +5V	–	2MΩ	• Resolution: 1/1024 (10 bits) • External power: +15V 120mA, −15V 40mA.	–	4	1
Input	Analog		B1075-1	0 to +10V	–	2MΩ	• Resolution: 1/4096 (12 bits) • External power: +15V 120mA, −15V 40mA.	–	4	1
Input	Analog		B1075-2	+1 to +5V	–	2MΩ	• Resolution: 1/4096 (12 bits) • External power: +15V 120mA, −15V 40mA.	–	4	1
Output	AC	100V	B1050	80 to 130V	2A per output 5A per 8 outputs	–	Min. load current: 10mA (rms).	10 ms	16	1
Output	AC	200V	B1054	160 to 260V	1A per output 5A per 8 outputs	–	Min. load current: 10mA (rms).	10 ms	16	1
Output	DC	5-12V	B1052	4.75 to 13.2V	5V: 0.1A per output 12V: 0.3A per output	–	OFF current: less than 1mA.	1 ms	16	1
Output	DC	48V	B1056	38 to 58V	2A per output 5A per output	–	OFF current: less than 1mA.	1 ms	16	1
Output	DC	24V	B1058	19 to 29V	2A per output 5A per 8 outputs	–	OFF current less than 1mA.	1 ms	16	1
Output	DC	24V	B1060	20.4 to 26.4V	0.1A per output 0.4A per 8 outputs		OFF current: less than 1mA	1 ms	64	1
Output	DC	24V	B1062	20.4 to 26.4V	0.3A per output 0.6A per 4 outputs		OFF current: less than 1mA	1 ms	32	1
Output	DC	24V	B1064	19 to 29V	0.3A		OFF current: less than 0.2mA. (without fuse).	1 ms	32	1
Output	AC/DC*		B1090R	• 220VAC, 0.8A (PF: 0.4) • 110VAC, 1.2A (PF: 0.4) • 24VDC, 1A (TM†: 40ms)		–	Miniature relay, coil voltage: 24VDC	OFF → ON 7 ms max ON → OFF 3 ms max	16	1
Output	AC/DC*		B1094	• 220VAC, 1A (PF: 0.4) • 110VDC, 0.5A (TM†: 100ms)		–	Power reed relay coil voltage: 24VDC	OFF → ON 7 ms max ON → OFF 3 ms max	8	1
Output	Register		B1070	–	50mA		External power +24V at 200mA.	64 ms	8	1
Output	Analog		B1072B-1	0 to +10V	10mA	–	• Resolution: 1/1024 (10 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1072B-2	0 to +5V	10mA	–	• Resolution: 1/1024 (10 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1072B-3	-5 to +5V	10mA	–	• Resolution: 1/1024 (10 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1072B-4	−10 to +10V	10mA	–		–	2	1
Output	Analog		B1074-1	0 to +10V	10mA	–	• Resolution: 1/4096 (12 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1074-2	0 to +5V	10mA	–	• Resolution: 1/4096 (12 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1074-3	−5 to +5V	10mA	–	• Resolution: 1/4096 (12 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Output	Analog		B1074-4	−10 to +10V	10mA	–	• Resolution: 1/4096 (12 bits) • External power: +15V 100mA, −15V 40mA.	–	2	1
Motion Control	Reversible Counter		B1081C	–	–	–	Max count speed: 40 kpps. External power: +12V at 100mA.	–	1	1
Motion Control	Preset Counter		B1082C	–	–	–	Max count speed: 40 kpps. External power: +12V at 100mA.	–	1	1
Motion Control	Positioning		B1083C	–					1	2
PID			B1080	–					1	1
Power Supply			B1089	–			Power supply for PID module.	–	1	1

*Requires power supply for exciting the coil of relay built-in.
†Time constant

2. 14 MOUNTING BASE

(1) Mounting Base B1030



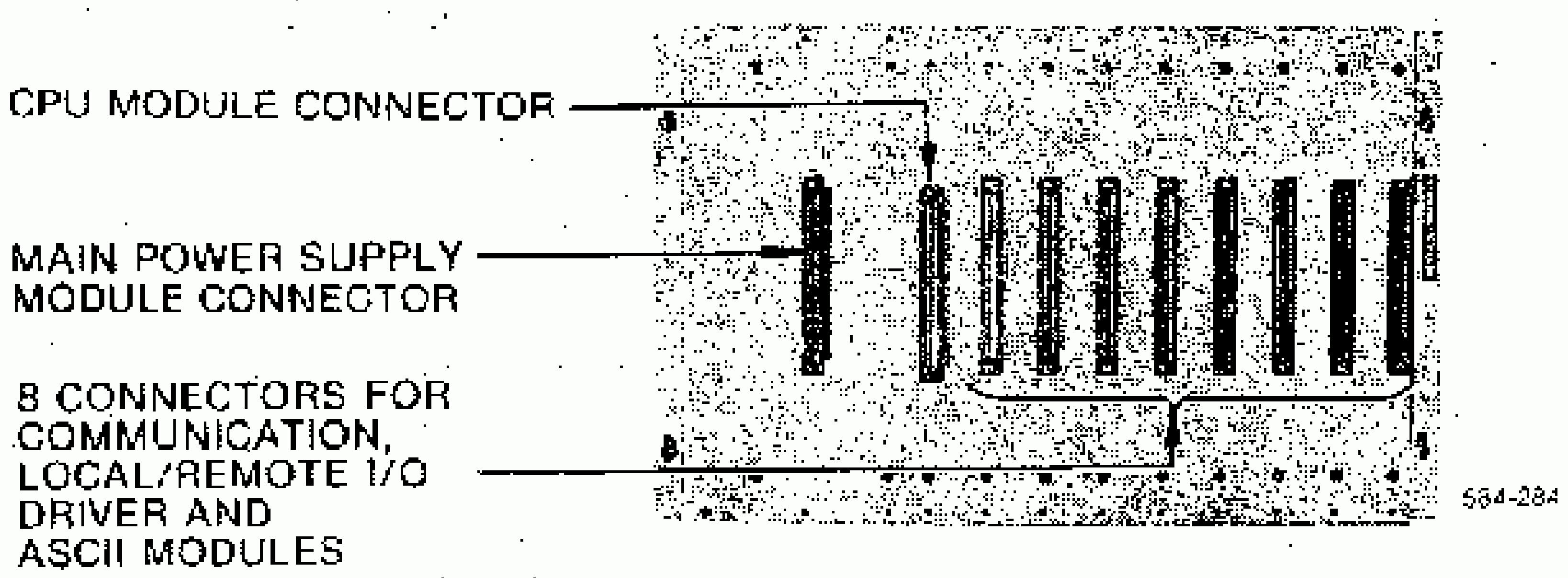
Note: Four connectors are available for communication, local/remote I/O driver or ASCII modules, any one of which can be used. One ASCII module can be mounted when local I/O driver is mounted.

Fig. 2. 15 Mounting Base B1030

Table 2. 16 Mounting Base B1030 Specifications

Items	Specifications
Type	JRMSI-B1030
Application	For mounting main power supply module, CPU module, communication module, local or remote I/O driver module and one ASCII module. Where remote I/O driver module is mounted, ASCII module is not used.
Dimensions in mm	340 (W) × 300 (H) × 28 (D)
Approx Weight	2.0 kg

(2) Mounting Base B1031



Note: Eight connectors are available for communication, local or remote I/O driver modules, any one of which can be used.

Fig. 2. 16 Mounting Base B1031

Table 2. 17 Mounting Base B1031 Specifications

Items	Specifications
Type	JRMSI-B1031
Application	For mounting main power supply module, CPU module, communication module, local or remote I/O driver module and up to 4 ASCII modules.
Dimensions in mm	480 (W) × 300 (H) × 28 (D)
Approx Weight	2.8 kg

(3) Mounting Base B1033

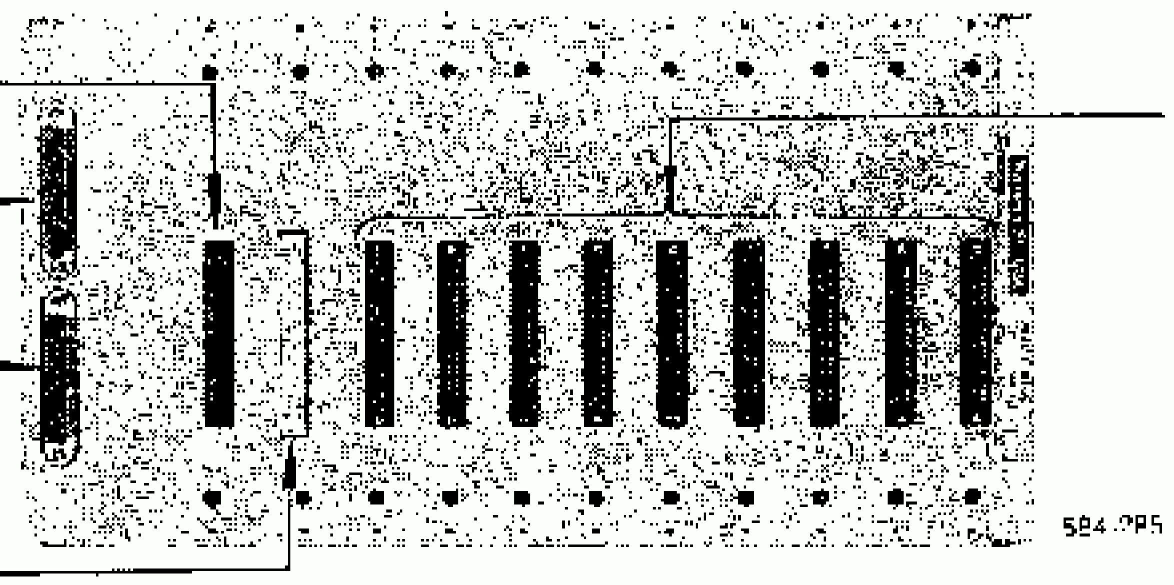


Fig. 2.17 Mounting Base B1033

Table 2.18 Mounting Base B1033 Specifications

Items	Specifications
Type	JRMSI-B1033
Application	For mounting auxiliary power supply module 1, local I/O receiver module and up to 9 I/O modules as I/O section rack 1.
Dimensions in mm	480 (W) × 250 (H) × 28 (D)
Approx Weight	2.5 kg

(4) Mounting Base B1034

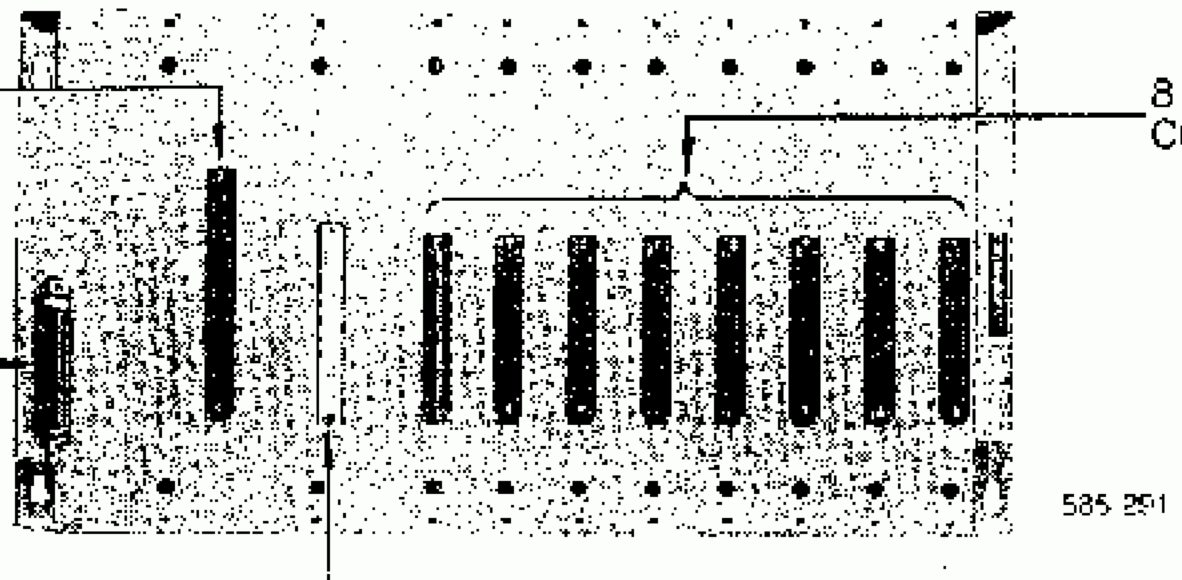


Fig. 2.18 Mounting Base B1034

Table 2.19 Mounting Base B1034 Specifications

Items	Specifications
Type	JRMSI-B1034
Application	For mounting auxiliary power supply module 2, remote I/O receiver module and up to 8 I/O modules as I/O section rack 1.
Dimensions in mm	480 (W) × 250 (H) × 28 (D)
Approx Weight	2.5 kg

(5) Mounting Base B1028

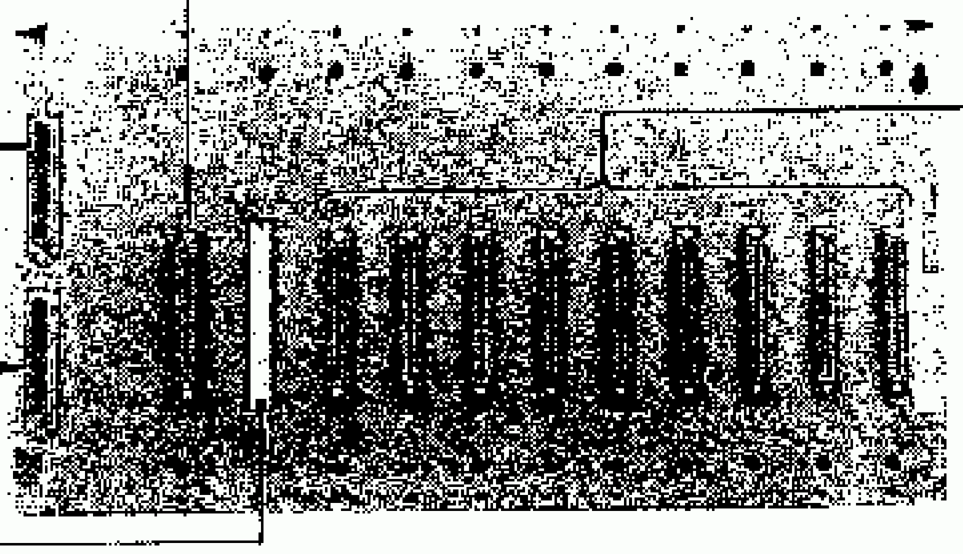


Fig. 2. 19 Mounting Base B1028

Table 2. 20 Mounting Base B1028 Specifications

Items	Specifications
Type	JRMSI-B1028
Application	For mounting auxiliary power supply module 1, I/O buffer module and up to 9 I/O modules as I/O section rack 3
Dimensions in mm	480 (W) × 250 (H) × 28 (D)
Approx Weight	2.5 kg

(6) Mounting Base B1027

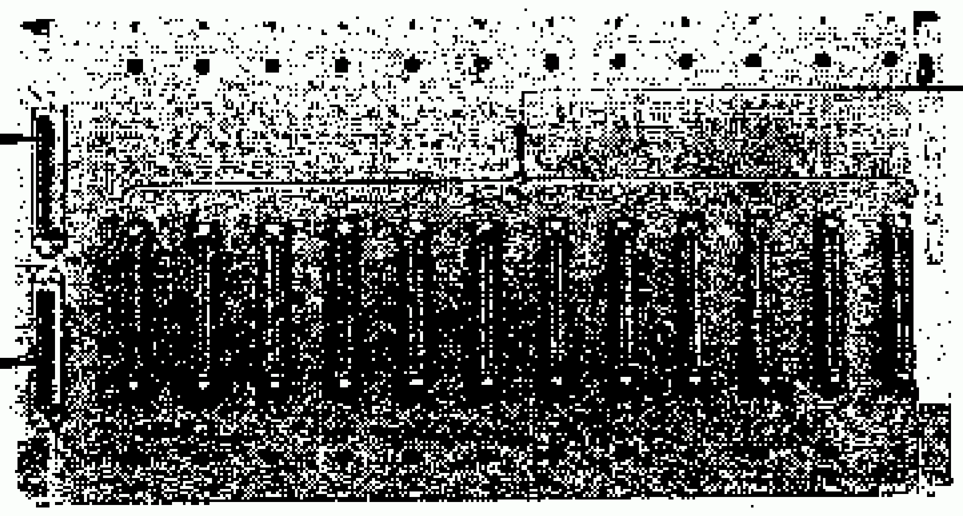


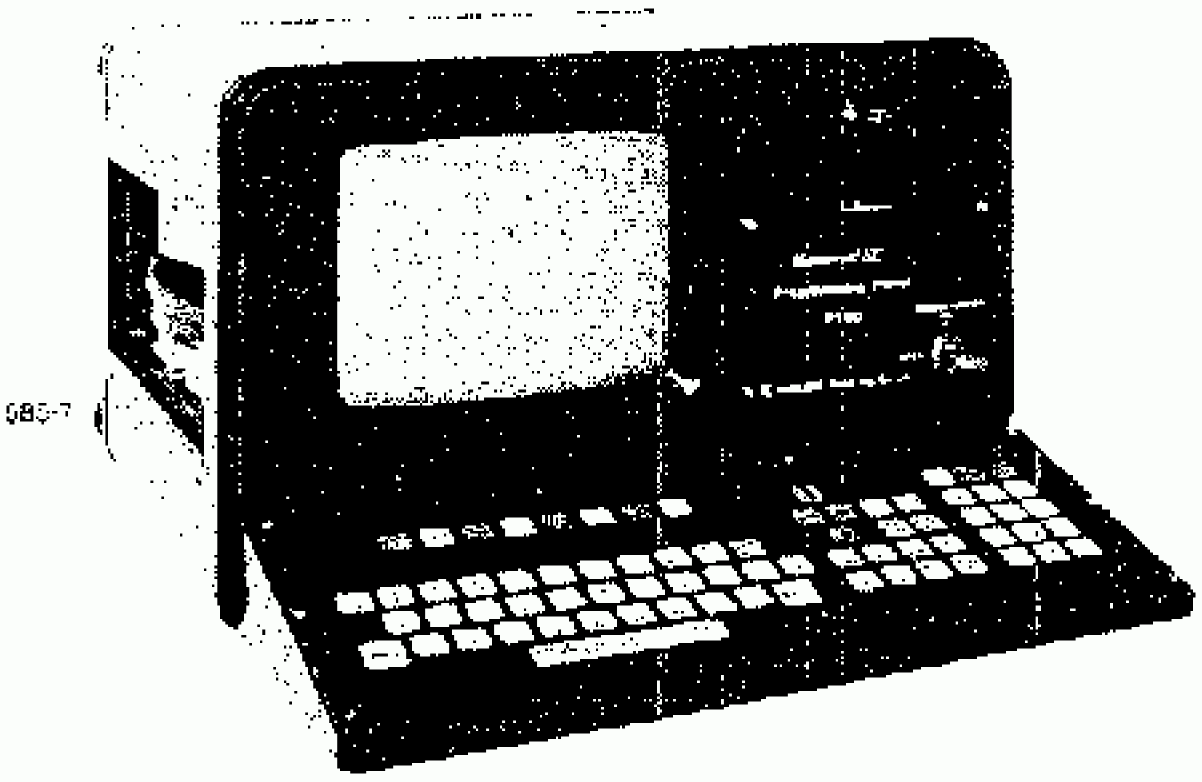
Fig. 2. 20 Mounting Base B1027

Table 2. 21 Mounting Base B1027 Specifications

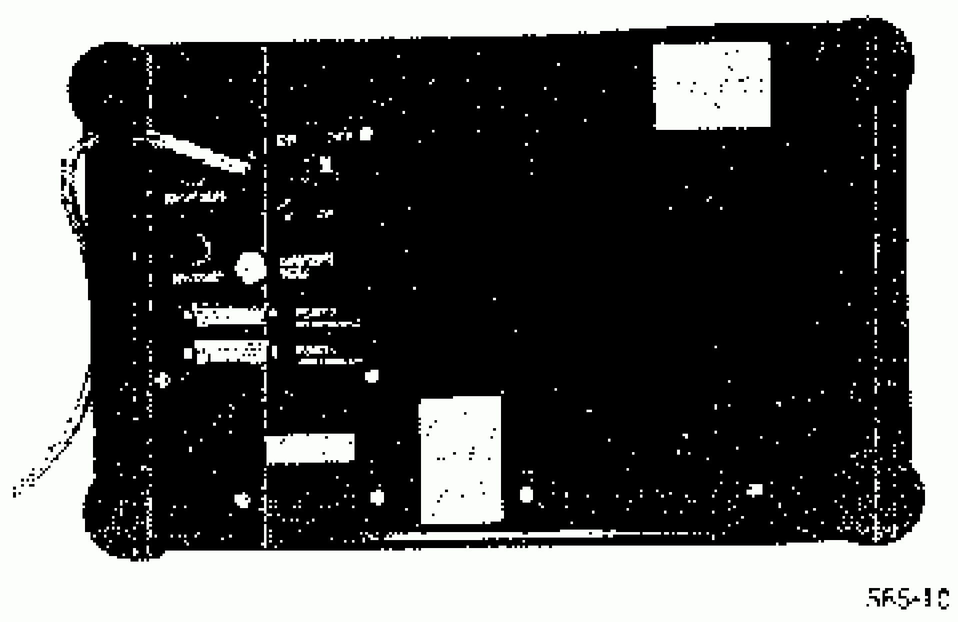
Items	Specifications
Type	JRMSI-B1027
Application	For mounting up to 12 I/O modules as I/O section racks 2 and 4
Dimensions in mm	480 (W) × 250 (H) × 28 (D)
Approx Weight	2.5 kg

2. 15 P190 CRT PROGRAMMING PANEL

For details of the P190 handling, refer to DESCRIPTIVE INFORMATION "Memocn-SC U84 P190 CRT Programming Panel User's Manual" (SIE-C815-10.2).



(a) Front



(b) Rear

Fig. 2. 21 P190 CRT Programming Panel

Table 2. 22 P190 Programming Panel Specifications

Items		Specifications
Type		DISCT-P190
Functions*		• Programming, adding, altering and deleting of logic and data • Logic entering, logic display • Load, dump and verify functions by mini-cartridge tape • Definition of system configuration • I/O allocation • Various monitorings
Attachments	CRT	9" (228.6mm)
	Keyboard	CRT label keys, function keyboard, numeric keyboard, ASCII keyboard, cursor control keys
	Tape Loader	Use of mini cartridge tape (Type DC 100 A)
	Video	• Monitor type: Black and white, raster scan CRT • Signal characteristics: EIA RS-170 • Video response: 10 MHz • Horizontal frequency: 16,041 Hz • Vertical frequency: 51.4 Hz
	Communication Port†	Two EIA RS-232C ports
Connection of P190 to Communication Module		Type W1015-1 (length: 2.5 m) or W1015-2 (length: 1.5 m) cable
Standard Specifications	Power	100 VAC ±15 V, single phase, at 50/60 Hz (47.5-63 Hz)
	Consumed Power	150 VA
	Ambient Temperature	−5°C to +45°C
	Storage Temperature	−20°C to +60°C
	Humidity	20 % to 80 % relative (non-condensing)
	Atmosphere	Free from explosive, inflammable, and corrosive gases.
	Grounding	Chassis grounding line is connected to mainframe grounding line via connecting cable of communication module.
	Approx Weight	15 kg

*A control program in a data cartridge tape must be loaded in the P190.

†A ladder diagram can be printed out through a printer connected to the P190.

Note: The P190 is also available for *Memocon-SC* R84, R84H, 584, 684, 684H and 484. EIA adaptor type DISCT-J470 is required for the 684, 484 and R84H (without MEMOBUS). The R84 requires EIA adapter type DISCT-J470 and cable adapter type JZMSZ-W710. When the power supply module type JRMSP-P7011 is used, this cable adapter is not required.

Data Cartridge Tape

Before using the P190 programming panel, a control program in a data cartridge tape must be loaded in the P190. There are the following data cartridge tapes. Select a suitable one according to the purpose.

(1) U84 Programmer Tape [TU84-S001]

Use this to make a program with the P190 or to perform I/O allocation.

(2) U84 Utility Tape [TU84-S002]

Use this to print out a ladder diagram through a printer connected to the P190.

(3) U84 ASCII Program Tape [TU84-S003]

This tape is used for storing ASCII message (to specify data I/O format) into ASCII module by P190, or for displaying or altering ASCII messages.

(4) U84 Tape Loader Tape [TU84-S004]

Use this to let the P190 function as a tape loader which includes the following functions.

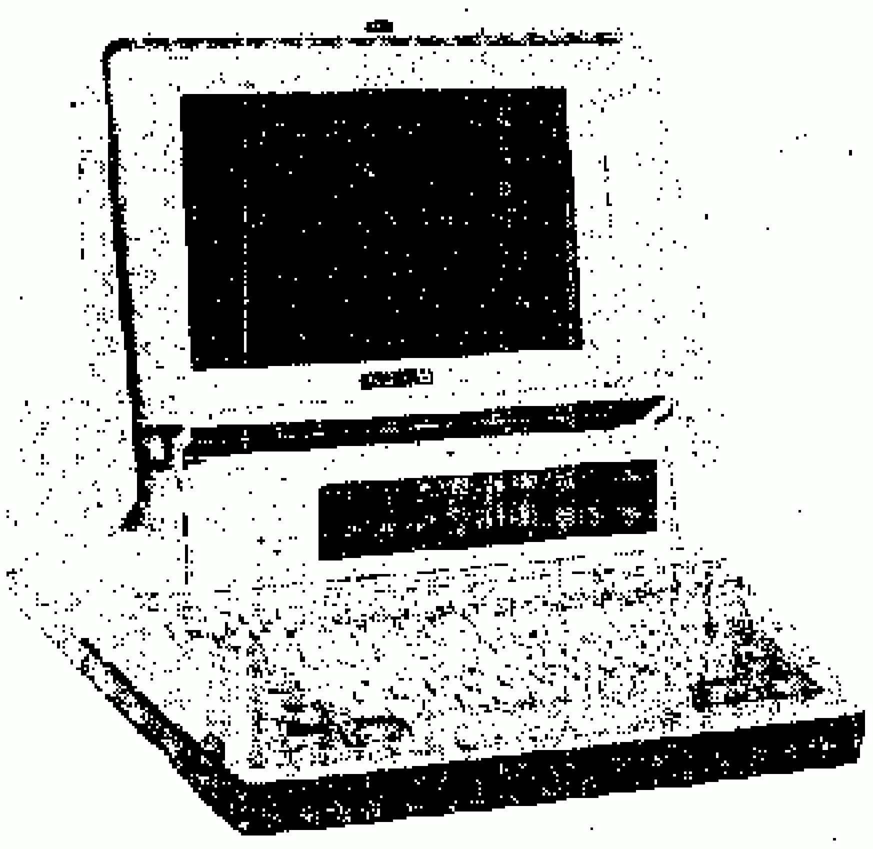
- Load: To load a ladder circuit, etc. in a tape to the CPU module
- Dump: To dump a ladder circuit, etc. in the memory of the CPU module to the tape.
- Verify: To verify the contents of a tape with those of the CPU module memory.

(5) Blank Tape [T190-000]

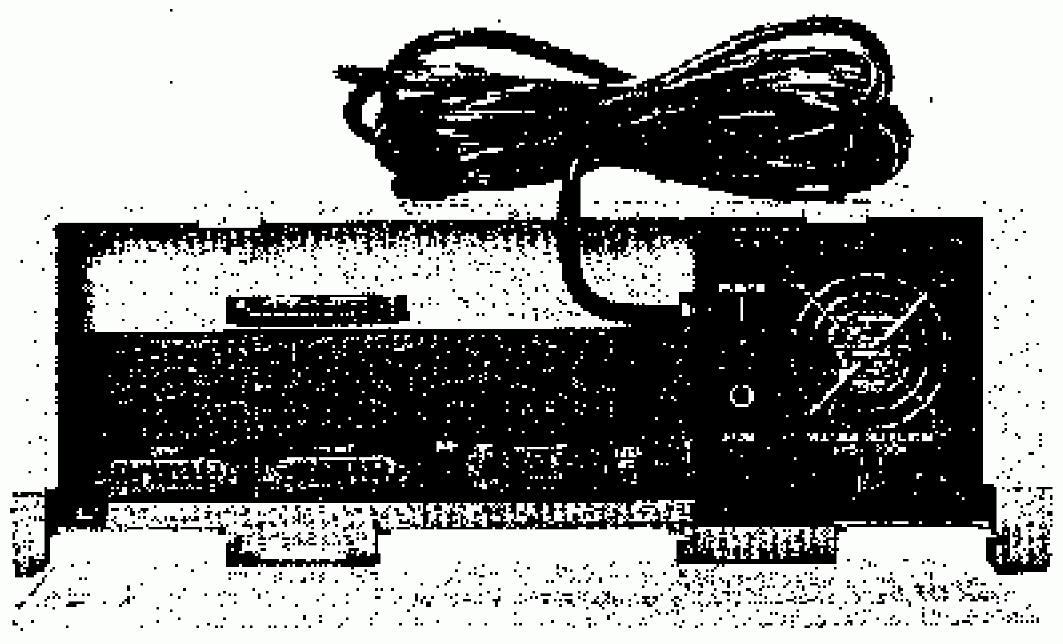
Use a blank tape to record the ladder programs stored in the CPU module memory. The contents of the registers and I/O allocation are recorded together.

2. 16 P150 PROGRAMMING PANEL

586 240



(a) Front



586 251

(b) Rear

Fig. 2. 22 P150 Programming Panel

Table 2.23 P150 Programming Panel Specifications

Items		Specifications
Type		DISCT-P150-10, DISCT-P150-11
Functions*		• Programming, adding, altering and deleting of logic and data • Logic entering, logic display • Load, dump and verify functions • Definition of system configuration • I/O allocation • Various monitorings, file control
Attachments	Graphic Display	Plasma display: 640 dots × 400 dots
	Keyboard	Label keys, function keyboard, numeric keyboard, ASCII keyboard, cursor control keys
	Floppy Disk Drive	Two 3.5-inch floppy disk drive
	Video	• Monitor type: Black and white, raster scan CRT (Type PC 8841 made by NEC Corp. is available.)
	Communication Port†	• One parallel port (in compliance with Centronics specification) • Two EIA RS-232C ports
Connection of P150 to Communication Module		Type W1006-1 cable (length: 5 m) or W1006-2 cable (length: 15 m)
Standard Specifications	Power	85 to 132 VAC/195 to 265 VAC, single phase, changeover at 50/60 Hz (47.5 to 63 Hz)
	Consumed Power	120 VA
	Ambient Temperature	+5°C to +40°C
	Storage Temperature	−20°C to +60°C
	Humidity	20 % to 80 % relative (non-condensing)
	Atmosphere	Free from explosive, inflammable, corrosive gases, and dust.
	Grounding	Chassis grounding line is connected to mainframe grounding line via connecting cable of communication module.
	Approx Weight	9 kg

*A control program in a 3.5-inch floppy disk must be loaded in the P150.
†A ladder diagram can be printed out through a printer connected to the P150.

3.5-Inch Floppy Disk

Before using the P150 programming panel, a control program in a 3.5-inch floppy disk must be loaded in the P150. Select one of the following 3.5-inch floppy disks suitable for the intended use.

(1) U84 Programmer FD (FU84-001)

This is used to make a program and control load, dump, verify and file functions with P150, and also used to perform I/O allocation.

(2) U84 Ladder Lister FD (FU84-002)

This is used to print out a ladder diagram through a printer connected to the P150.

(3) U84 ASCII Programmer FD (FU84-003)

This floppy disk is used to store ASCII messages (to specify data I/O format) into ASCII module by P150, or for display or alter, ASCII messages.

(4) U84 ASCII Message Lister FD (FU84-004)

This is used to print out an ASCII message through a printer connected to the P150.

(5) U84 Off-line Programmer FD (FU84-005)

This is used to make, alter or edit a U84 program, or for I/O allocation. Any message for coil or input relay can be displayed.

(6) Blank FD (F150-000)

A blank floppy disk is used to record the ladder circuits stored in the CPU module memory by using a programmer floppy disk. The contents of the registers and I/O allocation are recorded together.

SECTION 3
IMPORTANT MACHINE CONCEPTS

3.1 NETWORKS

The U84 program is composed in units of network. The multi-node format allows for up to ten elements of the program in each horizontal rung of the ladder diagram. Up to seven of these rungs can be combined into a network of relay contacts and other programming elements (timers, counters, etc.); each network can have up to seven coils placed at the extreme right of the network.

The networks are stored in the memory of the CPU module in the order of the network numbers.

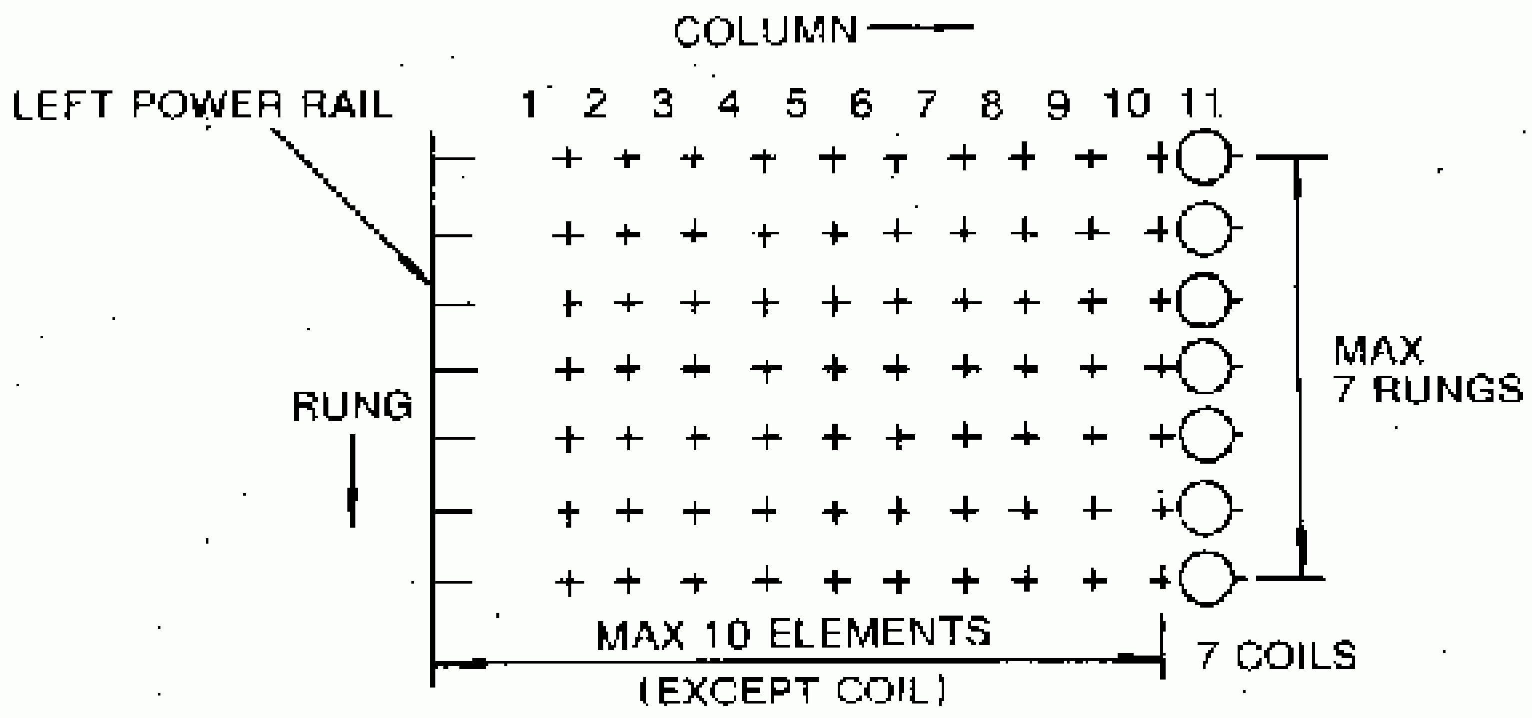


Fig. 3.1 Networks

3.2 CONTROLLER REFERENCE NUMBERS

Throughout the programming of the U84 Controller, five-digit reference numbers are utilized to build the user's logic. These references are divided into two broad categories: discretes and registers. Discrete references are used for individual items that can be either ON or OFF, such as limit switches, pushbuttons, relay contacts, motor starters, relay coils, solenoid valves, etc. Register references are used to store numerical values such as counts, times, analog values, etc.; all register references are four BCD digits long (maximum value 9999). Since there are four bits per BCD digit, registers can also be 16 bits of data.

Only four types of references are required to program the U84 Controller. Any specific reference can be used as many times as required by the particular application; there are no limitations on the number of times a reference is used. References are defined as shown in Table 3.1.

Table 3.1 The Number of Reference

Reference Number	Elements
0XXXX	Coils and their contacts
1XXXX	Input Relays
3XXXX	Input Registers
4XXXX	Holding Registers

(1) Coils

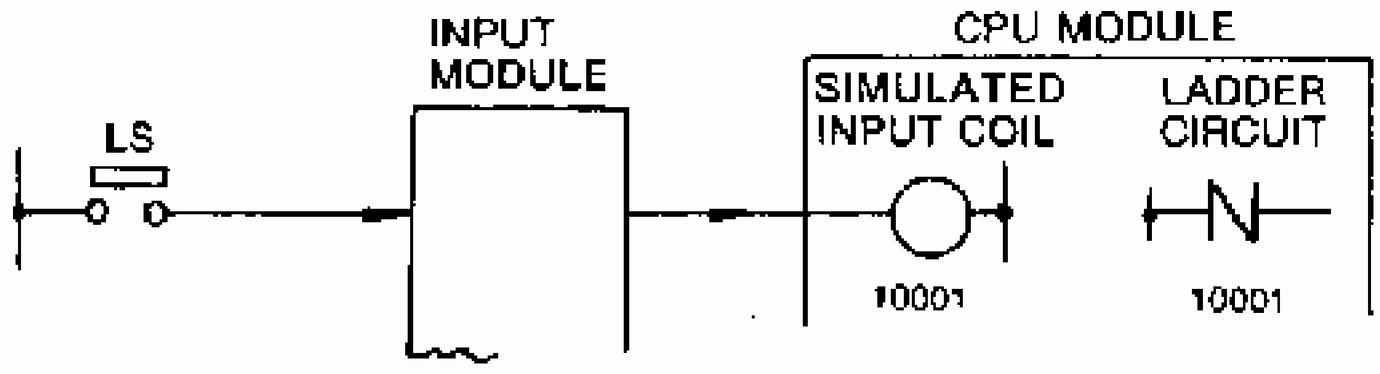
Other than the coils used in a relay circuit, coils can be used for any results of processing such as timer, counter, arithmetic operations, data move, matrix, etc.

Coils are divided into two types: normal coil (de-energized after a power failure) and latch coil (retentived after a power failure). They are divided into two groups from another standpoint. One is the internal coil (auxiliary relay) which is used only in a ladder circuit but not as an external output. The is the output coil which drives an external device via an output module.

The coils are identified by specific coil numbers. The coil contacts (having the same reference numbers as the coils) may be used repeatedly in any network.

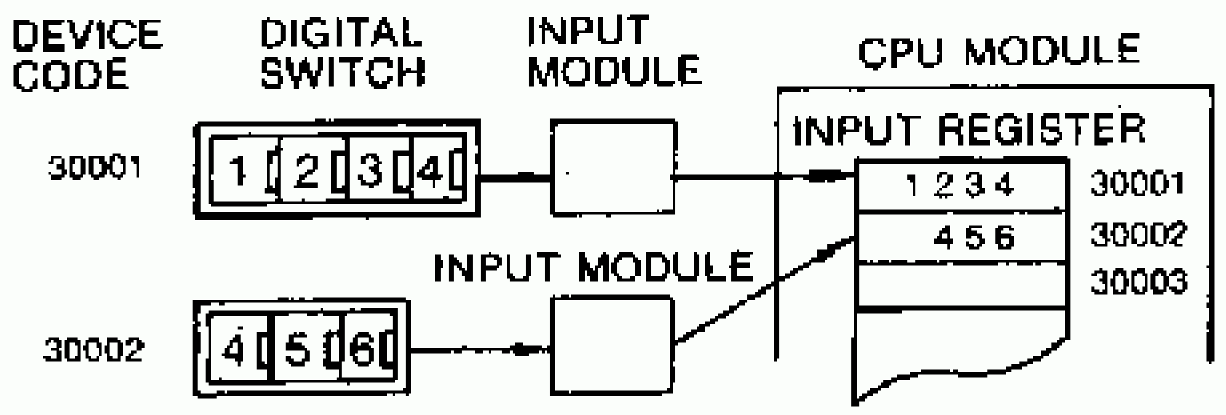
(2) Input Relays

An input relay is an ON/OFF signal (discrete input signal) entered through an input module. It may be considered as the contacts of a simulated input coil stored in the CPU module. The contacts of an input coil (input relay) may be used repeatedly in any network.



(3) Input Registers

An input register is memory (16 bits) for temporarily storing a numeric signal sent from an external device such as a digital switch, card reader, A/D converter, or computer. Each input register is identified by a reference number beginning with 3. The reference number may be considered as a device code given to the associated external device.

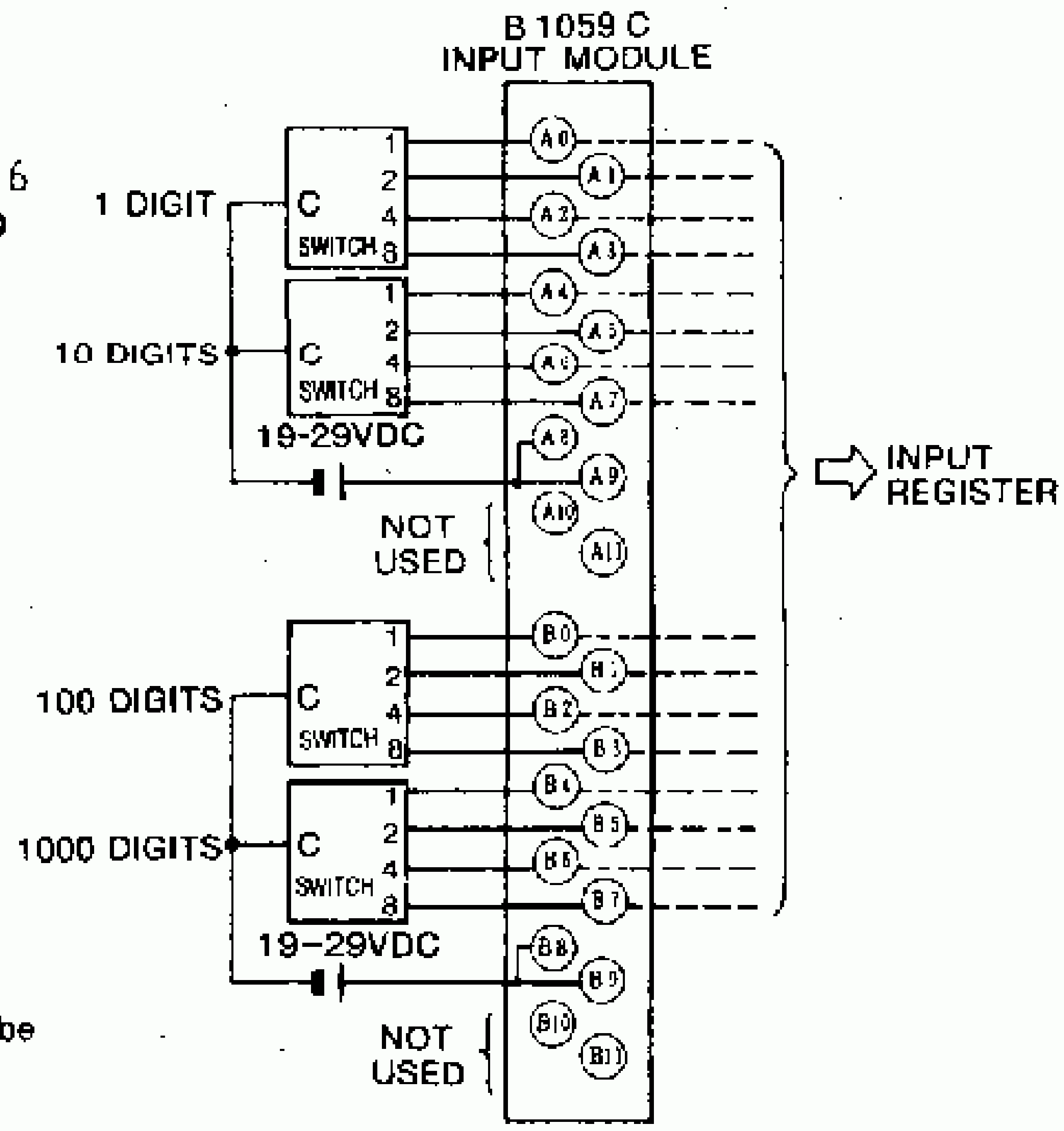


An input register holds numeric data (0-9999) of 1-4 digits in BCD form or 16 bits in binary form. The format of BCD or binary is defined by I/O allocation.

NOTE

The contents of any input register is binary in the memory. It is possible to refer to the contents of any input register but impossible to alter them in the ladder circuit.

Fig. 3.2 shows an example of connections of numeric input from digital switches.



Note: Numeric input (BCD) must be specified by I/O allocation.

Fig. 3.2 An Example of Connections of Numeric Input

(4) Holding Registers

The holding registers (of 16 bits each) hold the preset values and current values of the timer and counter, constants and results of arithmetic operations, transferred data, matrix data, or other constants needed for processing. Each holding register is identified by a reference number beginning with 4.

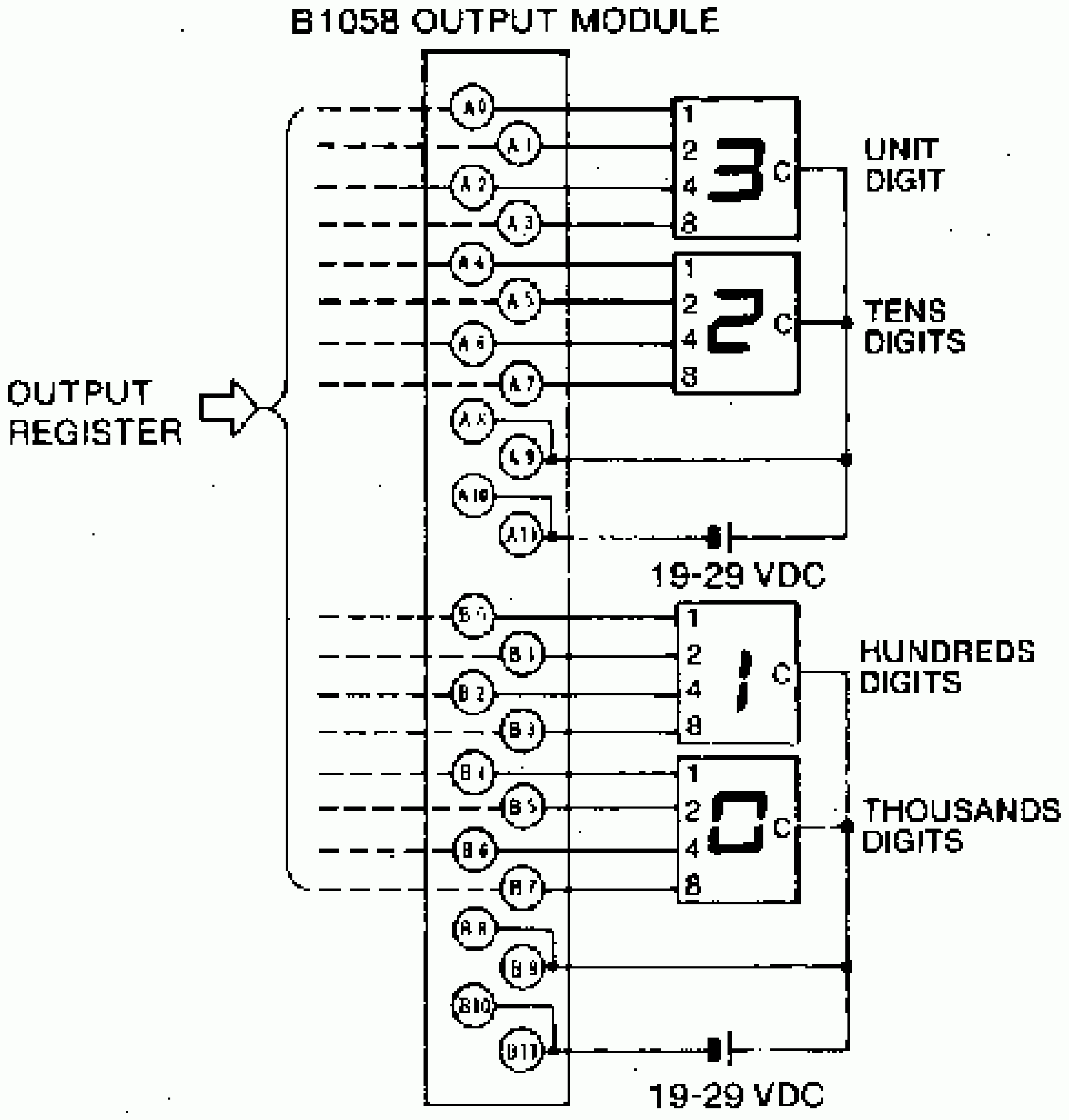
A holding register holds numeric data (0-9999) of 1-4 digits in BCD form or 16 bits in binary form.

NOTE

The internally contents of all holding registers are binary.

It is possible to refer to the contents of any holding register in a ladder circuit and even alter them. The contents are retentived during power failure.

The contents of the holding register can be output to the outside via an output module if necessary. At this time, the holding register may be called an output register. The output code is BCD or binary as defined by I/O allocation.



Note:
1. Numeric output (BCD) must be specified by I/O allocation.
2. The digital display is assumed to be of 24V interface, negative logic, and BCD code.

Fig. 3.3 Sample Connection of Numeric Output

(5) Range of Reference Numbers

The CPU module of the U84 comes in two types. One incorporates a program memory of 16k words and another 24k words. The range of reference numbers is the same regardless of the type of CPU module. Table 3.2 gives the classification of reference numbers.

Table 3.2 Range of Reference Numbers

Reference Number	Elements	Remarks
00001-02048	Output Coil and its contact	Available as internal coil
02049	Battery monitoring contact	ON when battery voltage is proper.
02050-04096	Internal coil and its contact	—
10001-12048	Input relay	—
30001-30256	Input register	—
40001-40256	Output register	Available as holding register
40257-49997	Holding register	—
49998	Storing of constant sweep set value	Available as holding register when not used for constant sweep.
49999	Storing of actual scan time	

3. 3 U84 INTERNAL PROCESS

Fig. 3.4 shows the U84 internal processing flow chart.

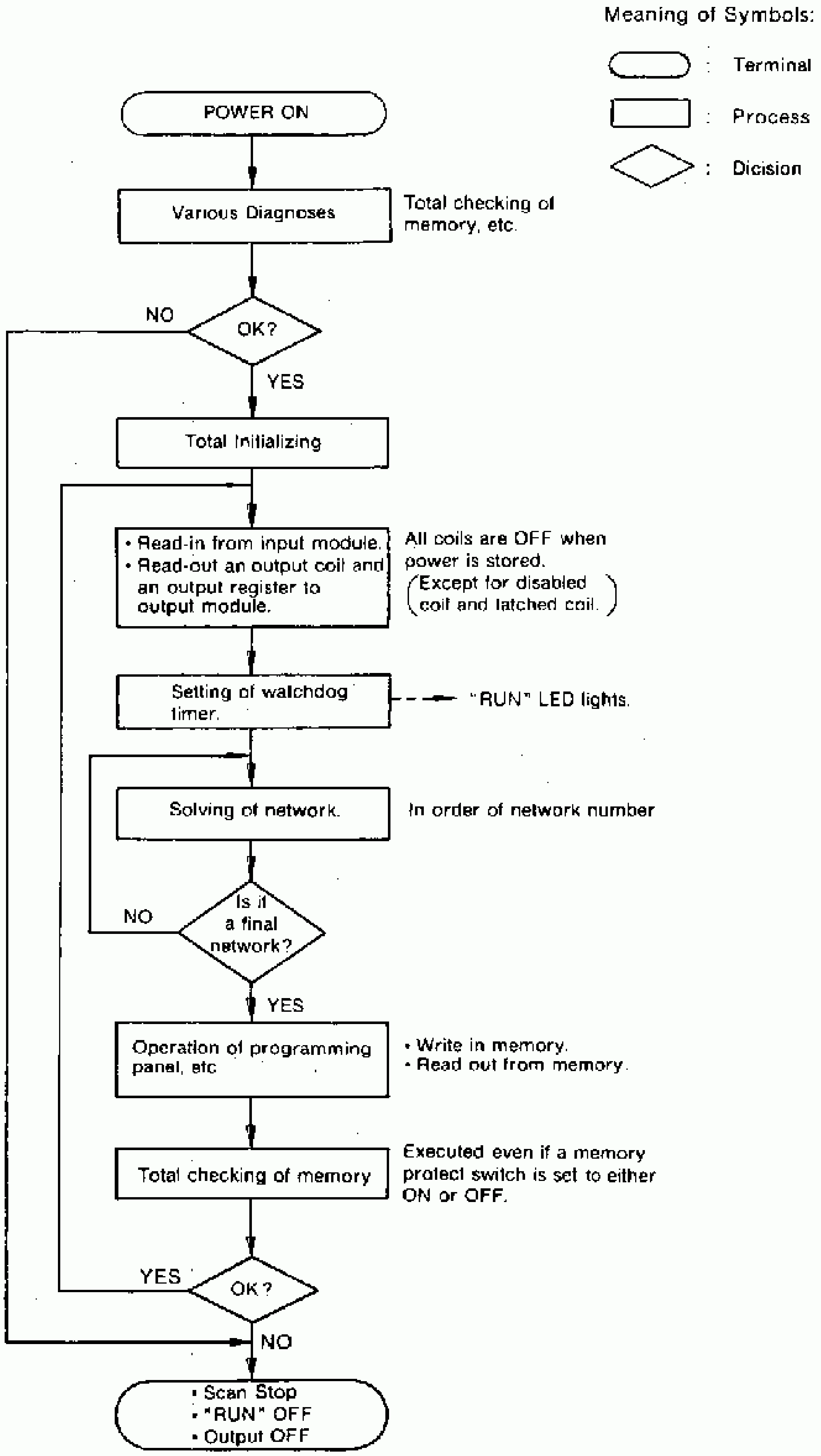


Fig. 3.4 U84 Internal Processing Flow Chart

(1) Power-up Sequence

When power is turned on, the system checks the contents of the memory and, if normal, initializes all input relays and coils, etc.

Table 3.3 shows the initialized statuses of input relays and coils.

Table 3.3 Initialized Statuses of Input Relays and Coils

Elements	Initialized Status
Coils	OFF except for latch coils and disabled coils
Input Relays, Input Registers	Latest status except for disabling input relays
Latch Coils, Disabling Inputs, Disabling Coils	Status held immediately before power failure
Holding Registers	Status held immediately before power failure

Note: Refer to 3.6 DISABLE FUNCTION.

The power-up sequence takes approximately 5 to 10 seconds.

(2) Scanning Cycle

The U84 Controller processes its logic data by solving networks in order of network number, beginning at network 1 and continuing until the last network programmed into the controller.

NOTE

Controller scans only memory actually programmed with logic.

A network is defined as a group of connected programmed elements; these elements can be relay contacts or numerical references. After all logics are solved, the inputs and outputs are all serviced together in one group. This completes one scan. As soon as one scan is completed, the next scan begins again with network 1.

(3) Watchdog Timer

A watchdog timer is set at the beginning of every scanning cycle. The timer remains ON for approximately 200 ms after it has been set. If the timer has not been set within a certain time interval, the system stops scanning, determining that there is something wrong with the scanning. At this time, the RUN LED on the CPU module comes OFF and all outputs become OFF. This is one of the self-diagnostic functions.

(4) Network Number

The ladder circuit is stored in the program memory in units of the network. The number of elements in a network must be in a range of 7 lines × 10 columns + 1 column (coil). The designer should designate the boundary of the network. The networks are numbered serially (1, 2, 3, and so on). As the designer designates the boundaries of the networks, the system assigns network numbers to them automatically. There are no limitations adding or deleting elements in a network, adding or deleting networks, or inserting a new network between adjacent networks. When some networks are added or deleted, the system controls the network numbers automatically.

NOTE

Network numbers may increase until the program memory becomes full.

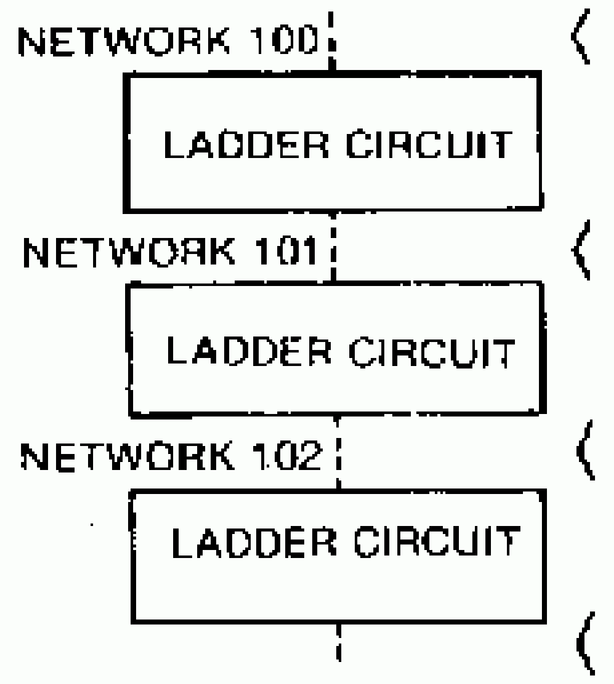


Fig. 3.5 Networks and Their Delimitations

Note: The designer designates the boundaries of each network (shown with <). It is actually performed by pressing the START NEXT key on the programming panel.

(5) Checksum

A serious error might occur if some contents of the memory, such as a ladder circuit or major system constants have been changed during operation and the changes cannot be detected. To prevent such problems, the system is provided with a self-diagnostic function called "Checksum." The total sum of the contents of the memory which should not be changed has been set. In every scanning cycle, the sum is calculated and compared with the previously setsum. If a discrepancy occurs, the U84 stops scanning as with any error. Simultaneously, all outputs connected to the output module are turned off. Register and analog outputs hold value before stopping the scan.

Total check covers the following.

- Ladder circuits
- I/O allocation table

NOTE

Total check is performed regardless of the memory protect switch position. When in OFF position, some ladder circuits may be altered through the programming panel and it will not be detected as an error by total check. Rather the total check sum is renewed and comparison will be made with reference to this new value.

3.4 SCANNING

(1) Solving a Ladder Circuit and Scanning

Each network is solved in order of column one to column ten and then to the coils. Within each column, the logic is solved from the top rung to the bottom rung of that column. The new results from each network (either coil status or register content) are immediately available for use by the next network or column. The scan is done by network number not by output coil number. Fig. 3.6 shows the sequence the U84 solves networks. The scanning technique is essential to the operation of the U84 controller and should be understood before proceeding. Table 3.4 shows the status of each element during scanning.

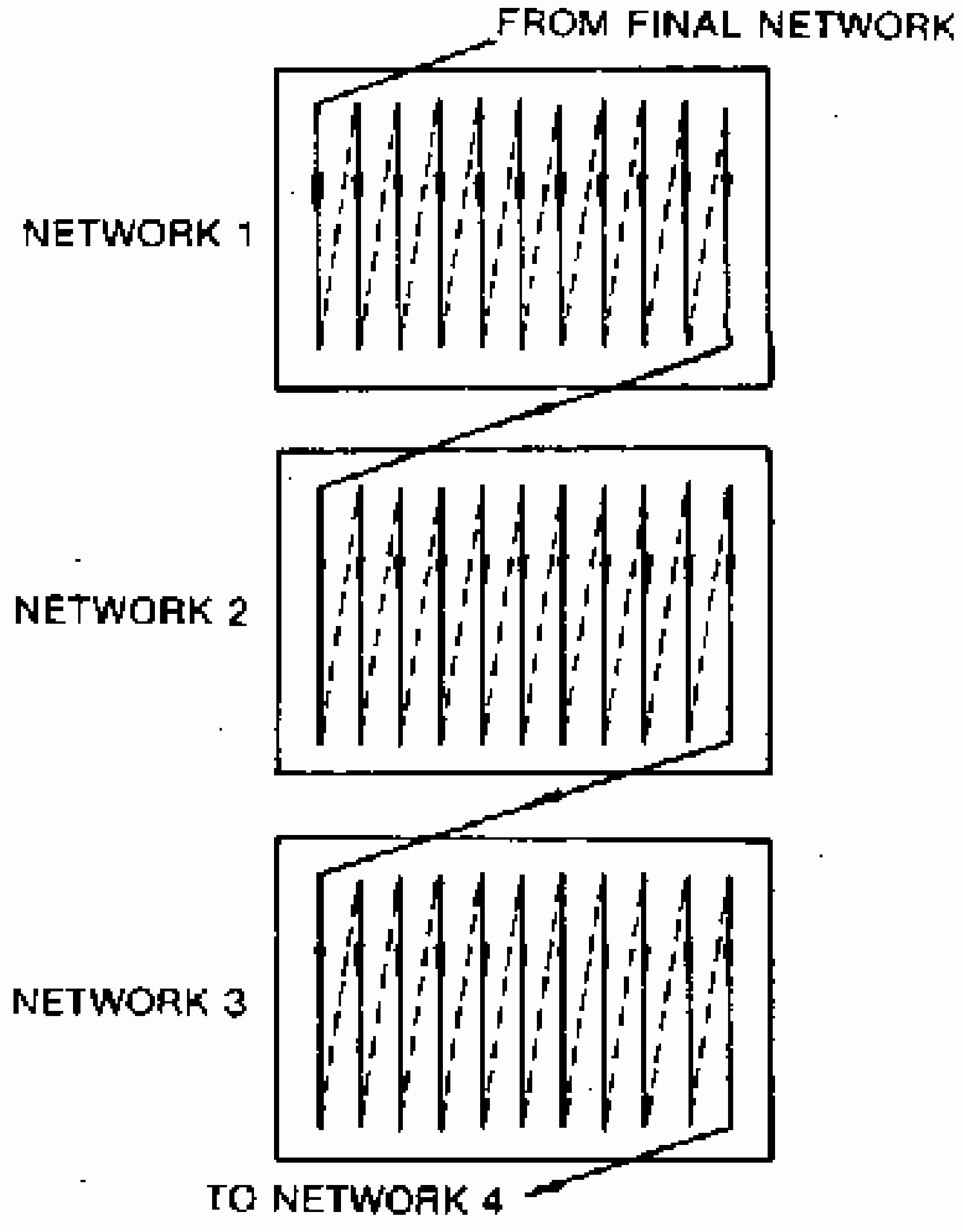


Fig. 3.6 Sequece of Solving Networks

3. 4 SCANNING (Cont'd)

Table 3. 4 Status of Each Element during Scanning

Element	Status
Input Relays (Except for Disabling Operation Inputs), Input Registers	Latest status after power-up sequence. Then input status, updated just when inputs are read in, remain unchanged until the next scanning.
Coils (Except for Latch Coils and Disabling Operation Coils)	OFF after power-up sequence. A coil is turned on or off according to the result solved in the first scanning status remains until the next scanning.
Latch Coils	After power-up sequence, status before power failure are restored. During scanning, the same as with coils.
Disabling Operation Inputs, Disabling Operation Coils	After power-up sequence, status before power failure are restored. Statuses are updated only when changed through the programming panel at the end of scanning.
Holding Registers	After power-up sequence, status before power failure are restored. During scanning, the contents vary according to changing the register contents using a ladder diagram.

Note: Input and output are actually performed when they are updated, as described in Fig. 3. 4.

As the U84 solves a ladder circuit by scanning, the ON/OFF status of coils and latch coils and the contents of the holding registers may change and the new status is referred to, not only in the subsequent networks, but in the same network for the elements which are to be solved later.

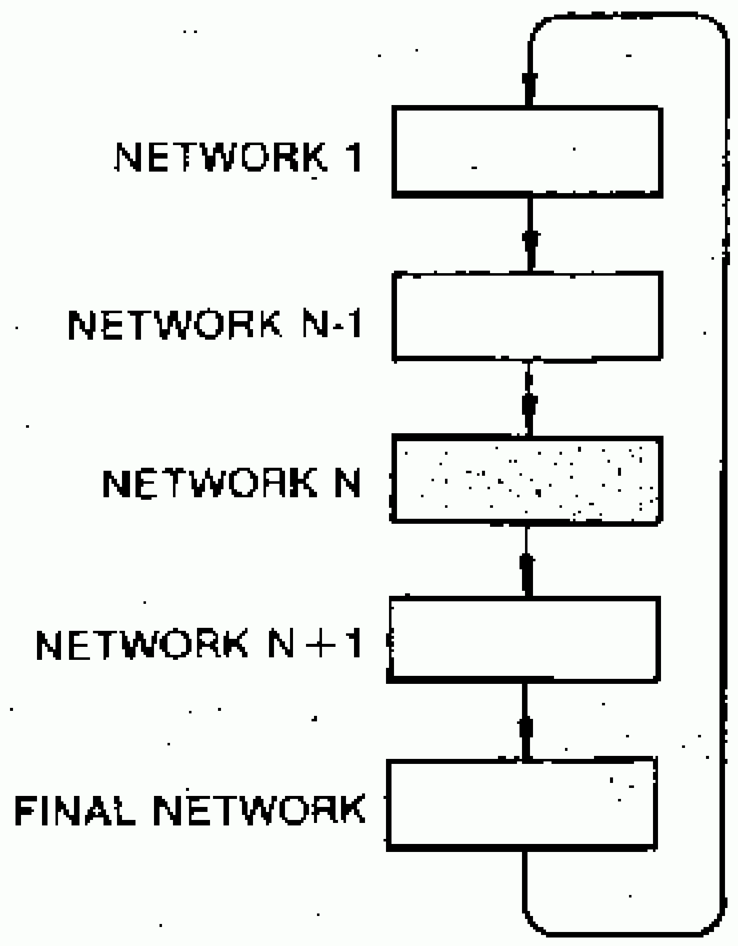


Fig. 3. 7 Example of Sequence of Solving Networks

In Fig. 3.7, the status of a coil solved in network N does not change between network N+1 and the final network nor between network 1 and network N-1. But the contents of a holding register may be changed at any time during a scanning cycle and the revised contents remain held until they are changed again.

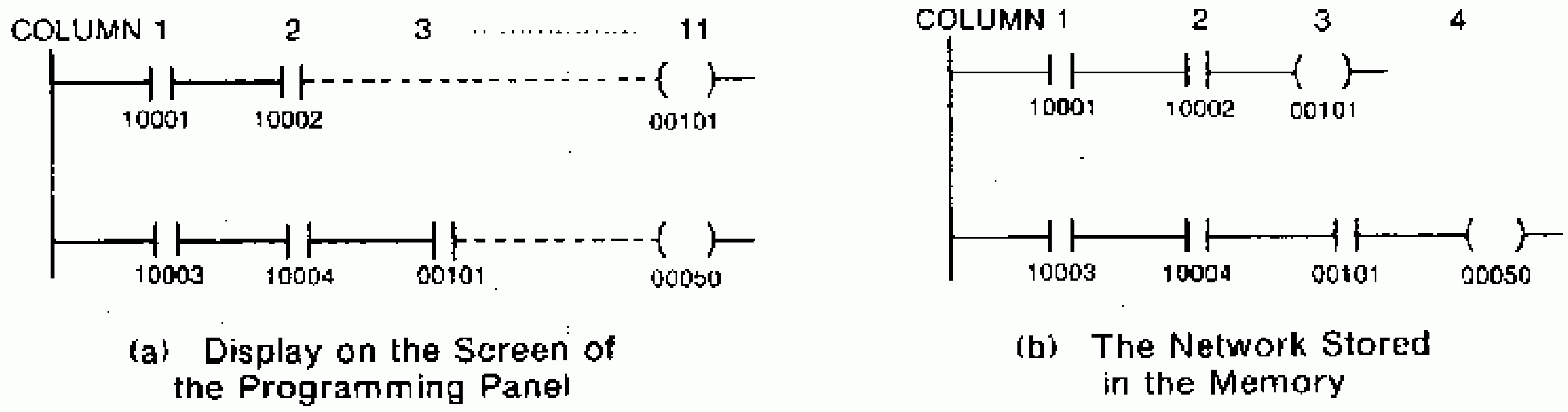
For example, a network is composed as shown in Fig.3.8 (a). This ladder diagram is held in the memory as shown in Fig.3.8 (b). The U84 solves the network as shown below where the elements are identified by the reference numbers.

10001 → 10003 → 10002 → 00101 (coil) → 00101 (contact)→ 00050 (coil).

In this example, a change in coil 00101 will be reflected immediately to line 2.

NOTE

If an element has been inserted between the contact 10002 and the coil 00101 on line 1, the change in coil 00101 will be reflected to line 2 during the next scanning cycle but not during the same scanning cycle as that of the change. If such a delay is a problem, remove line 2 from the network and place it in the next one.



Note: Columns 4-10 are not shown above. They are indicated with dotted lines as shown above also on the programming panel.

Fig. 3.8 Example of Sequence of Solving in a Network

(2) Scan Time

The scan time T(ms) of one scanning cycle is determined roughly by the following formula.

T = (Basic Time) + (Additional Time) + (Network Processing Time)

Where

- Basic time = 2 ms ── Processing time for self diagnosis
- Additional time = 5 ms ── I/O processing time
- Network processing time = {(number of networks) × 6.7 + Σ(processing time of each element)} ÷ 1000 (ms) ── See Table 3.5.

The processing time of each element varies during execution time and non-execution time as shown in Table 3.5. Therefore, the scan time usually varies. The U84 is provided with a "constant sweep" function which keeps the scan time fixed (refer to the manual of the Memocon-SC U84 P190 CRT programming panel for detail). This function uses two holding registers as follows.

- Holding register 49998: stores the preset value of constant sweep given in the unit of 10 ms between 10 and 200 ms.
- Holding register 49999: stores the actual scan time when constant sweep is executed. The value varies in the unit of 10 ms.

NOTE

If the preset value is smaller than the actual scan time, the actual time overrides the preset value.

3.4 SCANNING (Cont'd)

Table 3.5 Processing Time of Elements

Element (Function)	Condition	Processing Time (μs) Non Execution	Execution	Remarks	
Coil, Latched Coil	ENABLE	—	12.3	—	
	DISABLE	—	8.4		
Contact, Horizontal Open/Shunt	—	—	0.35		
Transitional Contact	—	—	6.0		
Timer	—	15.1	15.4		
Counter	—	22.1	23.8		
Addition	—	16.8	19.3		
Double-precision Addition	—	17.2	67.9		
Subtraction	—	16.8	20.3		
Doble-precision Subtraction	—	17.2	105		
Multiply	0 × 0	17.2	22.1	—	
	9999 × 9999		103		
Double-precision Multiply	0 × 0	17.2	27.0	—	
	99999999 × 99999999		401		
Divide	Quotient overflow	16.8	23.1	—	
	For remainder		106		
	For decimal part		186		
Double-precision Divide	Quotient overflow	11.9	23.5	—	
	For remainder		331		
	For decimal part		514		
R→T	—	19.6	24.5	—	
T→R	—	19.6	25.6		
T→T	—	19.6	25.6		
FIN		19.6	25.9 + 1.4 n	n: Table size	(1 ≦ n ≦ 100)
FOUT	Coil as destination	20.7	28.7		
	Register as destination		27.3		
BLKM	Coil as destination	18.2	27.0 + 4.9 n	n: Table size	(1 ≦ n ≦ 100)
	Register as destination		23.8 + 2.5 n		
STAT	Coil as destination	11.9	17.9 + 4.2 n	n: Table size	(1 ≦ n ≦ 21)
	Register as destination		15.8 + 2.1 n		
SRCH	Compare	19.3	27.0 + 2.1 n	n: Table size	(1 ≦ n ≦ 100)
	Non compare		23.5 + 2.1 n		
AND, OR, XOR	Coil as destination	18.6	25.6 + 4.6 n	n: Table size	(1 ≦ n ≦ 100)
	Register as destination		23.1 + 2.5 n		
COMP	Coil as destination	18.6	25.6 + 4.2 n	n: Table size	(1 ≦ n ≦ 100)
	Register as destination		23.1 + 2.1 n		
CMPR	Miscompare	20.0	26 + 2.1 m + 3.2 n	m: Bit number in miscompare n: Table size	(0 ≦ m ≦ 15) (1 ≦ n ≦ 100)
	Non miscompare		22.8 + 3.2 n		
MBIT	Coil as destination	22.1	37.5	—	
	Register as destination		36.1		
SENS	—	20.3	34.0	—	
BROT	Coil as destination	18.2	32.6 + 5.6 n	n: Table size	(1 ≦ n ≦ 100)
	Register as destination		31.2 + 3.5 n		
MROT	Shift	18.2	27.3 + 1.1 m + 2.8 n	m: Number of bits for shift n: Table size	(0 ≦ n ≦ 15) (1 ≦ n ≦ 100)
	Rotate		28.4 + 1.1 m + 2.8 n		
TWST	—	11.9	29.4 + 2.8 n	n: Table size	(1 ≦ n ≦ 100)
SQRT	√0	11.9	15.4	—	
	√9999		154		
DSQR	√0	11.9	16.8	—	
	√99999999		265		
DIBT	—	20.3	29.4 + 2.5 n	n: Table size	(1 ≦ n ≦ 100)
DIBR	—	20.3	29.4 + 2.5 n		
SIBT	—	20.3	28.4 + 2.5 n		
SIBR	—	20.3	26.6 + 2.5 n		

Note:
1. The processing time for a vertical short is zero.
2. The data given above simply provide you with a basis for calculating processing time. It is recommended to measure the actual processing time by the method shown in Fig. 4.17.

3.5 ALLOWABLE NUMBER OF MEMORY WORDS

The following formula gives the memory capacity in words needed for a network.

Memory Capacity (in Words) = 1 + (Number of Columns in Use) + (Total Number of Elements)

As an example, the detail of the network memory capacity is described using the ladder circuit of Fig. 3.9.

· Number of Columns in Use = 9 .... Columns 1-9

NOTE

The coils A, B, and C are stored at the locations of *1, *2, and *3 in the memory as shown in Fig. 3.8 (b).

· Total Number of Elements = 7 (Line1) + 5 (Line2) + 6 (Line3) + 4 (Line4) = 22

NOTE

On line 2, the locations of *4 and *5 (blank) and the vertical short do not occupy any memory. Horizontal shorts are counted as elements. As explained later, the timer and addition are counted as 2 and 3, respectively.

As a result,

· Memory Capacity = 1 + 9 + 22 = 32 (words)

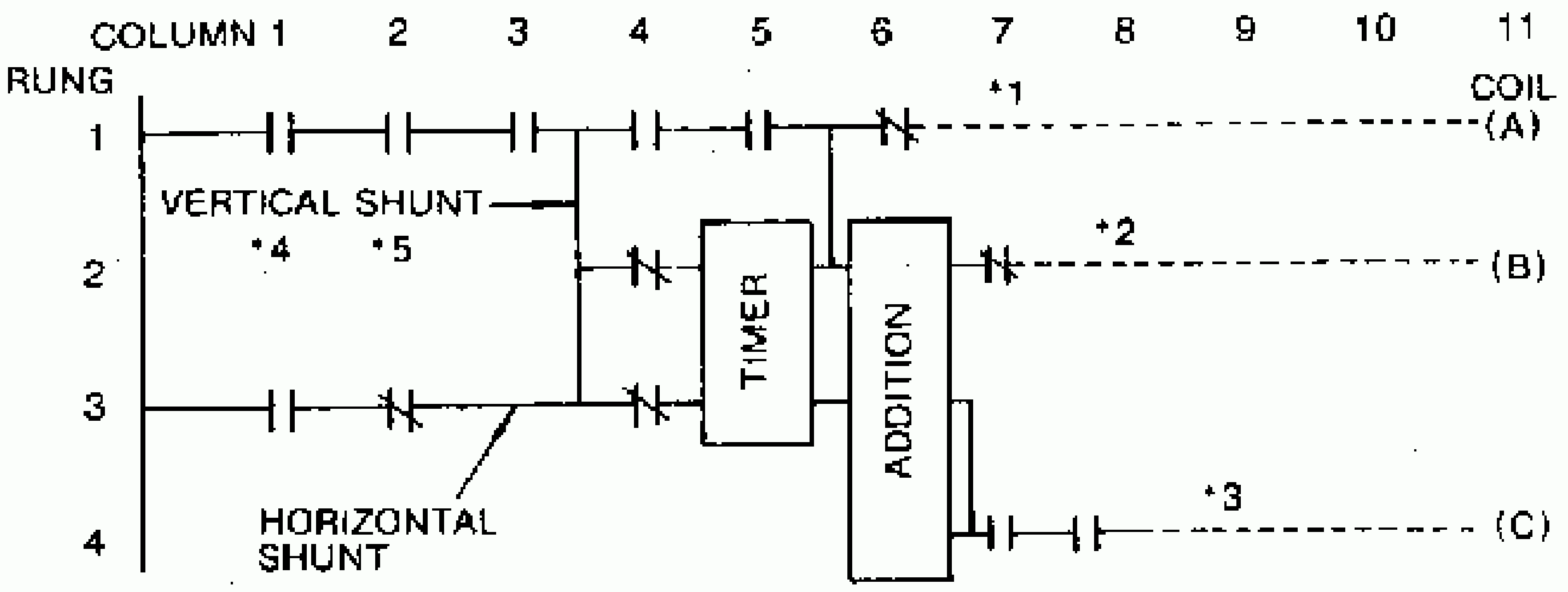
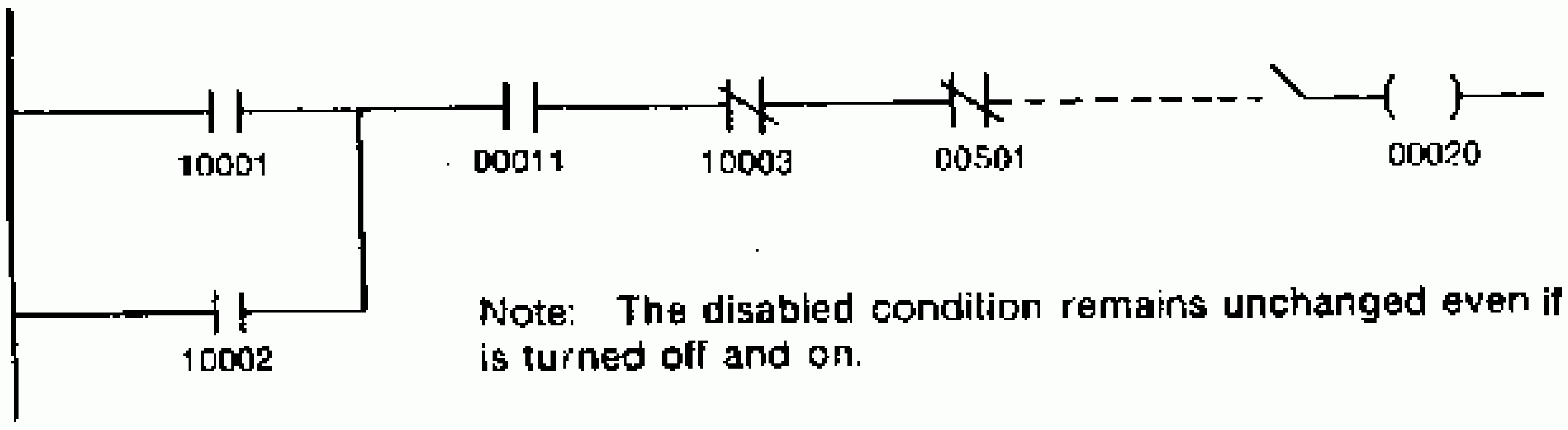


Fig. 3.9 An Example of Ladder Diagram for Calculating Memory Capacity Needed for Network

3.6 DISABLE FUNCTION

To simplify the checkout and maintenance of a control system using the U84 Controller, a special feature is incorporated into the Controller. This feature is called the Disable function. The Disable status is alterable only if Memory Protect is OFF. Any logic coil seelcted by the CRT's cursor can be disconnected from its logic by depressing the DISABLE pushbutton. If the coil was OFF when the pushbutton was depressed, it will remain OFF; if it was ON, it will remain ON. The coil is no longer controlled by the program in the Controller, but is now controlled by the operator via the CRT Programming Panel or RAP of a communication module. The coil can be toggled ON/OFF/ON/OFF by successively depressing the FORCE pushbutton.

Fig. 3.10 shows an example of a disabled coil display on the screen of the programming panel.



Note: The disabled condition remains unchanged even if power is turned off and on.

Fig. 3.10 Sample Display of a Disabled Coil

3.7 INPUT/OUTPUT ALLOCATION

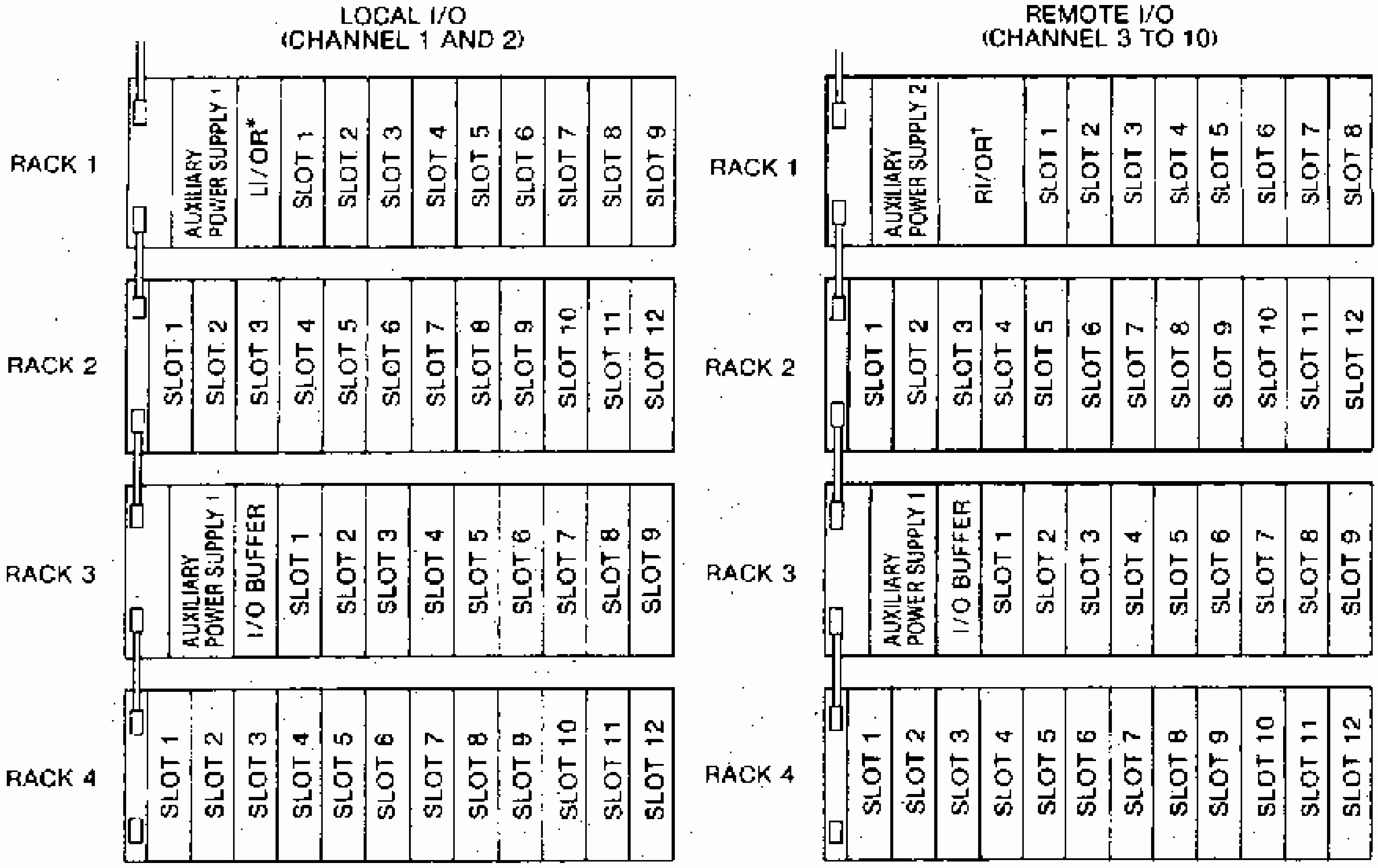
Since I/O module can be located at any module slot corresponding to I/O allocation, a variety of combination of I/O modules is available. Befor operating U84 Controller, be sure to set I/O allocation table to the CPU module memory using the programming panel.

I/O allocation is made independently to each location. A change of I/O allocation made to a location does not affect those for the other locations. For the operation of I/O allocation, refer to the Memocon-SC U84 P190 CRT Programming Panel User's Manual 2" (SIE-C815-10.2).

(1) I/O Section Configuration

The U84's I/O section consists of local I/O and remote I/O. 2 channels for local I/O (42 slots max per channel) and 8 channel for remote I/O (41 slots max per channel) are available in I/O section.

I/O modules may be installed at any location in a range of 2048 discrete input/output points and 256 register input/output points. Fig. 3.11 shows I/O section configuration of local and remote I/O for each one channel.



* Local I/O receiver module

† Remote I/O receiver module

Fig. 3.11 I/O Section Configuration

(2) I/O Modules Layout

I/O modules may be installed at any location of 10 channels in a range of 2048 discrete input/output points and 256 resister input/output points. They need not be installed in contiguous locations. However, it is recommended that the I/O modules be installed in groups (by input and output, voltage level, application, etc.). Fig. 3.12 shows a sample layout of I/O modules.

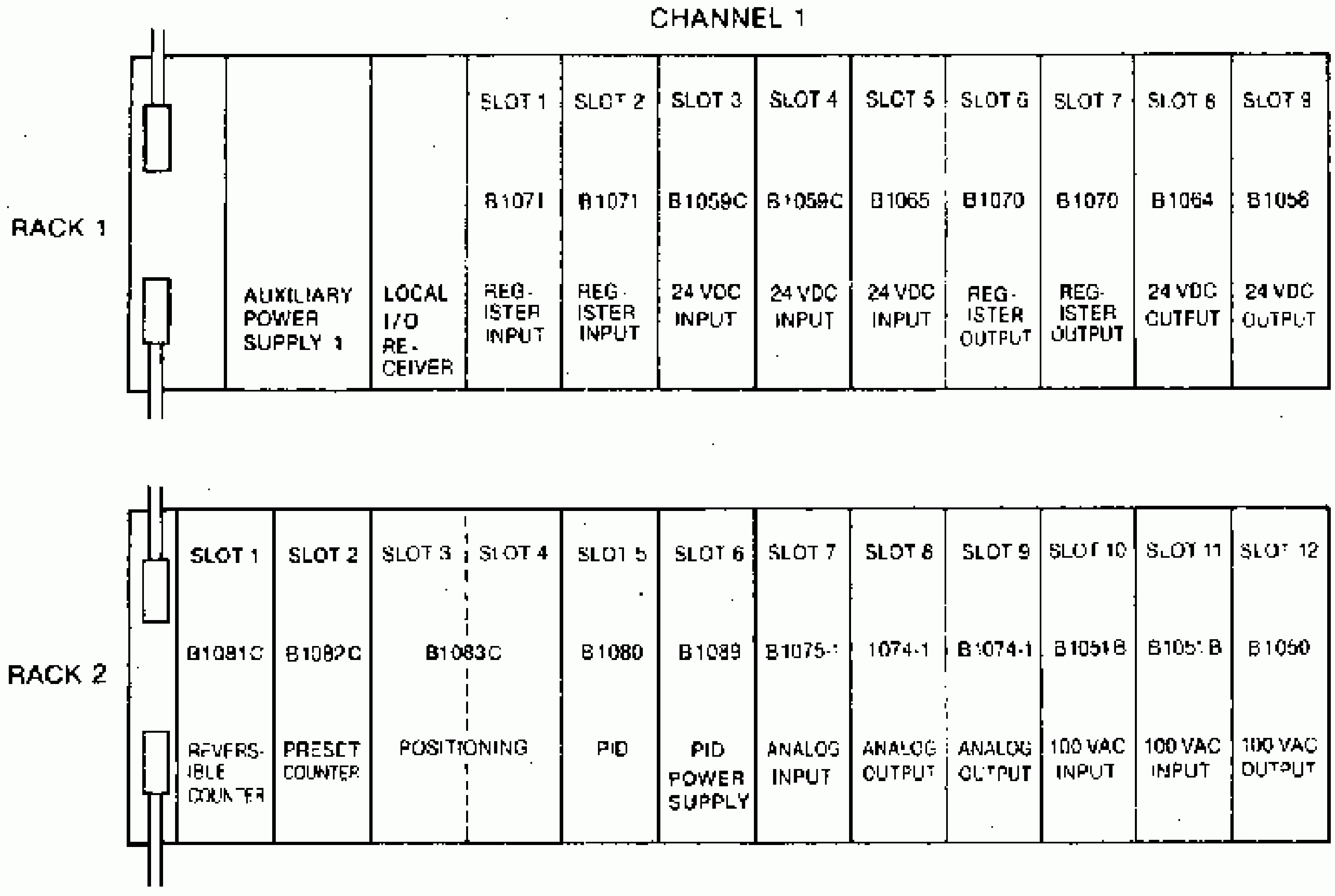


Fig. 3.12 Sample Layout of I/O Modules

(3) I/O Numbers

I/O signals include discrete inputs/outputs and register inputs/outputs (numerical value). The reference number is used as the I/O number.

Table 3.6 I/O Number List

Input/Output Type	I/O Number (Reference Number)
Discrete Input (Input Relay)	10001–12048
Register Input (Input Register)	30001–30256
Discrete Output (Output Coil)	00001–02048
Register Output (Output Register)	40001 40256

Note: The table above shows the range of I/O numbers as signable to each group of I/O signals. Note the following limitations:

- Discrete inputs + discrete outputs ≦ 2048
- Register inputs + discrete outputs ≦ 256

(4) I/O Module Location

The location of an I/O module is defined a channel number, a rack number, and a slot number.

① Local I/O

- Channel number: 1 or 2
- Rack number: 1-4
- Slot number: 1-9 (racks 1 and 3), 1-12 (racks 2 and 4)

② Remote I/O

- Channel mumber: 3-10
- Rack number: 1-4
- Slot number: 1-8(rack 1), 1-9(rack 3), 1-12(racks 2 and 4)

It is permissable to assign a pair of discrete input and output or a pair of register input and output to a slot. This is possible owing to the modules, such as the counter, PID, and positioning module, each of which deals with discrete inputs/outputs and register inputs/outputs (called a modular module). The number of input/output points allocated to a slot is given in Table 3.7.

Table 3.7 Number of Input/Output Points Allocated to a Slot

Input/Output Type	Allowable I/O Points in Allocation
Discrete Input	8, 16, 24, 32, 64
Discrete Output	8, 16, 24, 32, 64
Register Input	1–8
Register Output	1–8

Note:
1. Any I/O allocation is available in the range of I/O points given above. To a slot where a 16-point discrete output module is installed, for example, 8 or 16 discrete outputs must be allocated.
   If 24 or 32 points are allocated to the slot (it is possible through the P190 programming panel), the outputs may be degraded.
2. Up to 128 discrete I/O points can be allocated in units of 8 points for future expansion. At present, however, no I/O module can deal with more than 32 discrete inputs/outputs.
3. BCD or binary must be specified for register inputs and outputs. Both BCD and binary cannot be specified for the same number.

(5) Types of I/O Modules and I/O Allocation

The maximum number of I/O points for allocation is specified in accordance with the type of I/O module as shown in Table 3.8. Refer to Table 3.8 and the followings.

- Registers can be allocated, in both BCD and binary forms, to a discrete module.
- I/O allocation for the register module can be performed in BCD or binary form, but not in discrete form. If BCD and binary forms are mixed and allocated in the same slot, BCD is assigned first to the specified number of registers then binary to the remaining registers.
- Analog modules must be allocated in binary form, not in discrete form.
- Both discrete and register points must be allocated to counters, PID, and positioning modules. The register I/O allocation must be performed in binary form.

- Two slots are reserved for positioning modules and the connector is installed in the right slot. Allocation should therefore be made to the right slot but it is not necessary for the left slot.
- Allocation is not necessary for the power supply module because it does not deal with I/O signals.

Table 3.8 Number of I/O Points for Allocation by Module Type

Modules	Type	Number of Input Point			Number of Output Point		
		Discrete	BCD	Binary	Discrete	BCD	Binary
16-point Discrete Input Module	B1051B, B1055 B1053, B1057 B1059C	16	0	0	0	0	0
		0	1	0	0	0	0
		0	0	1	0	0	0
32-point Discrete Input Module	B1063 B1065	32	0	0	0	0	0
		0	2	0	0	0	0
		0	0	2	0	0	0
64-point Discrete Input Module	B1061	64	0	0	0	0	0
		0	4	0	0	0	0
		0	0	4	0	0	0
Register Input Module	B1071	0	8	0	0	0	0
		0	0	8	0	0	0
Analog Input Module	B1073-1, B1073-2 B1075-1, B1075-2	0	0	4	0	0	0
8-point Discrete Output Module	B1094	0	0	0	8	0	0
16-point Discrete Output Module	B1050, B1052 B1054, B1056 B1058, B1090	0	0	0	16	0	0
		0	0	0	0	1	0
		0	0	0	0	0	1
32-point Discrete Output Module	B1062 B1064	0	0	0	32	0	0
		0	0	0	0	2	0
		0	0	0	0	0	2
64-point Discrete Output Module	B1060	0	0	0	64	0	0
		0	0	0	0	4	0
		0	0	0	0	0	4
Register Output Module	B1070	0	0	0	0	8	0
		0	0	0	0	0	8
Analog Output Module	B1072B-1, B1072B-2 B1072B-3, B1072B-4 B-1074-1, B1074-2 B-1074-3, B1074-4	0	0	0	0	0	2
Reversible Counter Module	B1081C	8	0	2	8	0	2
Preset Counter Module	B1082C	16	0	2	16	0	8 (2)*
PID Module	B1080	8	0	2	8	0	3
Positioning Module	B1083C	16	0	4	16	0	4
Power Supply Module	B1089	0	0	0	0	0	0

*Varies depending on the mode in use.

(6) I/O Allocation Example

When you have fixed the layout of I/O modules, make an I/O allocation table as shown in Table 3.9 and enter the reference numbers and the numbers of points in pairs.

The reference number should be the first number allocated to the slot. For discrete I/O allocation, the reference numbers must begin with fixed numbers as follows.

- First number of discrete output = 00001 + 8n (n=0, 1, 2, ..., 255)
- First number of discrete input = 10001 + 8n (n=0, 1, 2, ..., 255)

The following limitations exist in relation to the range of the U84 reference numbers (see Table 3.2)

- First number of discrete output + number of points ≦ 02049
- First number of discrete input + number of points ≦ 12049
- First number of register input + number of points ≦ 30257
- First number of register output + number of points ≦ 40257

Table 3.9 shows the I/O allocation table of the sample layout of I/O modules shown in Fig. 3.12.

Table 3.9 Example of I/O Allocation Table

Channel	Rack	Slot	I/O Module	Function	Input Allocation					Output Allocation				
					Discrete		Register			Discrete		Resister		
					Reference	No. of Point	Reference	BCD	Binary	Reference	No. of Point	Reference	BCD	Binary
		1	B1071 Register input	BCD 4 digit 8 sets/module			30001	8						
		2	B1071 Register input	Binary 16-bit 8 sets/module			30009		8					
		3	B1059C 24 VDC, 16-point input	Binary 16-bit 1 set/module			39017		1					
		4	B1059C 24 VDC, 16-point input	Discrete 16 points	10001	16								
	1	5	B1065 24 VDC, 32-point input	Discrete 32 points	10017	32								
		6	B1070 Register output	Binary 16-bit 8 sets/module								4000'		8
		7	B1070 Register output	BCD 4 digit 8 sets/module								40009	8	
		8	B1064 24 VDC, 32-point output	Discrete 32 points						00001	32			
		9	B1058 24 VDC, 16 point output	Discrete 16 points						00033	16			
		1	B1081C Reversible counter	—	10049	8	30018		2	00049	8	40017		2
1		2	B1082C Present counter	—	10057	16	30020		2	00057	16	40019		8
		3	B1083C Positioning control	—										
		4		—	10073	16	30022		4	00073	16	40027		4
		5	B1080 PID	—	10089	8	30026		2	00080	8	40031		3
		6	B1089 DC power for PID	—										
	2	7	B1075-1 Analog input	—			30028		4					
		8	B1074-1 Analog output	—								40034		2
		9	B1074-1 Analog output	—								40036		2
		10	B1051B 100 VAC, 16 point input	Discrete 16 points	10097	16								
		11	B1051B 100 VAC, 16-point input	Discrete 16 points	10013	16								
		12	B1050 100 VAC, 16-point output	Discrete 16 points						00097	16			

SECTION 4
PROGRAMMING FUNCTIONS

4.1 PROGRAMMING FUNCTION LIST

U84 is provided with the programming functions as shown in Table 4.1.

Table 4.1 . Programming Funtion List

<table>
<tr><th>No.</th><th>Name</th><th>Description</th><th>Page</th></tr>
<tr><td>1</td><td>Relay</td><td>
• For programming a relay circuit using relay elements.
• There are 9 types of relay elements as follows:
<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Normally Open Contact</td><td>─┤ ├─</td><td>6</td><td>Vertical Shunt</td><td>│</td></tr>
<tr><td>2</td><td>Normally Closed Contact</td><td>─┤/├─</td><td>7</td><td>Vertical Open</td><td>Non</td></tr>
<tr><td>3</td><td>Transitional Contact (OFF to ON)</td><td>─┤↑├─</td><td>8</td><td>Coil</td><td>─( )─</td></tr>
<tr><td>4</td><td>Transitional Contact (ON to OFF)</td><td>─┤↓├─</td><td>9</td><td>Latched Coil</td><td>─(L)─</td></tr>
<tr><td>5</td><td>Horizontal Shunt</td><td>────</td><td></td><td></td><td></td></tr>
</table>
</td><td>41</td></tr>
<tr><td>2</td><td>Timer</td><td>
• For counting a time while any signal is ON.
• There are 3 types of timers as follows:
<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Timer (Seconds)</td><td>T 1.0</td></tr>
<tr><td>2</td><td>Timer (Tenths of Seconds)</td><td>T 0.1</td></tr>
<tr><td>3</td><td>Timer (Hundredths of Seconds)</td><td>T 01</td></tr>
</table>
</td><td>50</td></tr>
<tr><td>3</td><td>Counter</td><td>
• For counting the number when any signal is changed from OFF to ON.
• There are 2 types of counters as follows:
<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Up Counter</td><td>UCTR</td></tr>
<tr><td>2</td><td>Down Counter</td><td>DCTR</td></tr>
</table>
</td><td>57</td></tr>
<tr><td>4</td><td>Arithmetic</td><td>
• Arithmetic operation for numerical values in 4-digit decimal form or 8-digit.
• There are 8 types of arithmetic as follows:
<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th><th>Functions</th></tr>
<tr><td>1</td><td>Addition</td><td>ADD</td><td>Addition in 4-digit decimal</td></tr>
<tr><td>2</td><td>Subtraction</td><td>SUB</td><td>Subtraction in 4-digit decimal</td></tr>
<tr><td>3</td><td>Multiply</td><td>MUL</td><td>Multiply in 4-digit decimal</td></tr>
<tr><td>4</td><td>Divide</td><td>DIV</td><td>Divide in 4 digit decimal</td></tr>
<tr><td>5</td><td>Double-precision Addition</td><td>DADD</td><td>Addition in 8 digit decimal</td></tr>
<tr><td>6</td><td>Double-precision Subtraction</td><td>DSUB</td><td>Subtraction in 8 digit decimal</td></tr>
<tr><td>7</td><td>Double-precision Multiply</td><td>DMUL</td><td>Multiply in 8-digit decimal</td></tr>
<tr><td>8</td><td>Double-precision Divide</td><td>DDIV</td><td>Divide in 8-digit decimal</td></tr>
</table>
</td><td>6</td></tr>
<tr><td>5</td><td>Square Root</td><td>
• Square root operation for numerical values in 4-digit decimal form or 8-digit.
• There are types of square root as follows:
<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th><th>Functions</th></tr>
<tr><td>1</td><td>Square Root</td><td>SQRT</td><td>In 4-digit decimal</td></tr>
<tr><td>2</td><td>Double-precision Square Root</td><td>DSQR</td><td>In 8-digit decimal</td></tr>
</table>
</td><td>88</td></tr>
</table>

4.1 PROGRAMMING FUNCTION LIST (Cont'd)

Table 4.1 Programming Function List (Cont'd)

<table>
<tr><th>No.</th><th>Name</th><th>Description</th><th>Page</th></tr>
<tr><td>6</td><td>Move</td><td>

• A variety types of data move in multi-data groups.
• There are 8 types of data move as follows:

<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Register-to-Table</td><td>R→T</td></tr>
<tr><td>2</td><td>Table-to-Register</td><td>T→R</td></tr>
<tr><td>3</td><td>Table-to-Table</td><td>T→T</td></tr>
<tr><td>4</td><td>Block Move</td><td>BLKM</td></tr>
<tr><td>5</td><td>First In</td><td>FIN</td></tr>
<tr><td>6</td><td>First Out</td><td>FOUT</td></tr>
<tr><td>7</td><td>Table Search</td><td>SRCH</td></tr>
<tr><td>8</td><td>Get Controller System Status</td><td>STAT</td></tr>
</table>

</td><td>92</td></tr>
<tr><td>7</td><td>Block Move with Index</td><td>

• For specifying a data group transfered by a index, or a destination.
• There are 4 types of block move with index as follows:

<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Block Move 1 with Destination Index</td><td>DIBT</td></tr>
<tr><td>2</td><td>Block Move 2 with Destination Index</td><td>DIBR</td></tr>
<tr><td>3</td><td>Block Move 1 with Source Index</td><td>SIBT</td></tr>
<tr><td>4</td><td>Block Move 2 with Source Index</td><td>SIBR</td></tr>
</table>

</td><td>113</td></tr>
<tr><td>8</td><td>Matrix</td><td>

• For operating upon bit patterns within the matrix.
• There are 10 types of matrices as follows:

<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Logical AND</td><td>AND</td></tr>
<tr><td>2</td><td>Logical OR</td><td>OR</td></tr>
<tr><td>3</td><td>Logical Exclusive OR</td><td>XOR</td></tr>
<tr><td>4</td><td>Logical Complement</td><td>COMP</td></tr>
<tr><td>5</td><td>Logical Compare</td><td>CMPR</td></tr>
<tr><td>6</td><td>Logical Bit Modify</td><td>MBIT</td></tr>
<tr><td>7</td><td>Logical Bit Sense</td><td>SENS</td></tr>
<tr><td>8</td><td>Logical Bit Rotate</td><td>BROT</td></tr>
<tr><td>9</td><td>Logical Multi-Bit Rotate</td><td>MROT</td></tr>
<tr><td>10</td><td>Logical Byte Rearrangement</td><td>TWST</td></tr>
</table>

</td><td>128</td></tr>
<tr><td>9</td><td>Skip</td><td>

• Any networks are skipped and not solved.
• Symbol: SKP

</td><td>145</td></tr>
<tr><td>10</td><td>ASCII</td><td>

• There are 2 types of operating functions for processing ASCII message as follows:

<table>
<tr><th>No.</th><th>Type</th><th>Symbol</th></tr>
<tr><td>1</td><td>Read Command</td><td>READ</td></tr>
<tr><td>2</td><td>Write Command</td><td>WRITE</td></tr>
</table>

</td><td>—</td></tr>
</table>

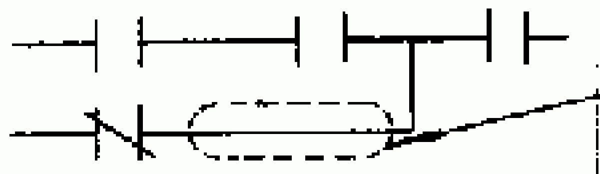
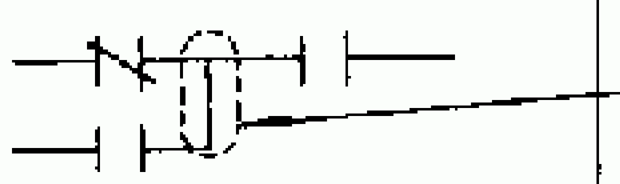
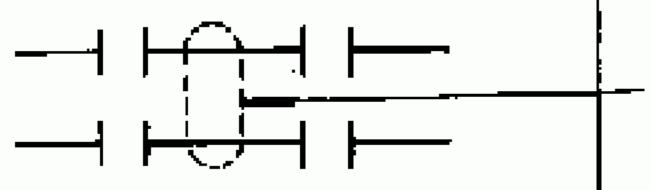
4. 2 RELAYS

4. 2. 1 Relay Logic Circuit

(1) Relay Elements

Table 4.2 shows relay elements for programming a relay circuit.

Table 4. 2 Relay Logic Elements

Element		Symbol (xxxxx: Reference Number)	Description	Reference Number
Contact	Normally Open	—| |— xxxxx	Operates while reference coil* is ON.	Input relay: 10001 to 12048 or Coil: 00001 to 04096
	Normally Closed	—|/|— xxxxx	Operates while reference coil is OFF.	
	Transitional Contact (OFF to ON)	—|↑|— xxxxx	Operates only in 1 scan cycle when reference coil is turned on	
	Transitional Contact (ON to OFF)	—|↓|— xxxxx	Operates only in 1 scan cycle when reference coil is turned off.	
Connection	Horizontal Shunt		Shunts between columns.	—
	Vertical Shunt		Shunts between lines.	
	Vertical Open		Opens the vertical shunt.	
Coil	Coil	—( )— xxxxx	De-energized when power is restored.	Coil: 00001 to 04096 (except for 02049)†
	Latched Coil	—(L)— xxxxx	Energized coil when power is restored.	

*Meaning of reference coil:

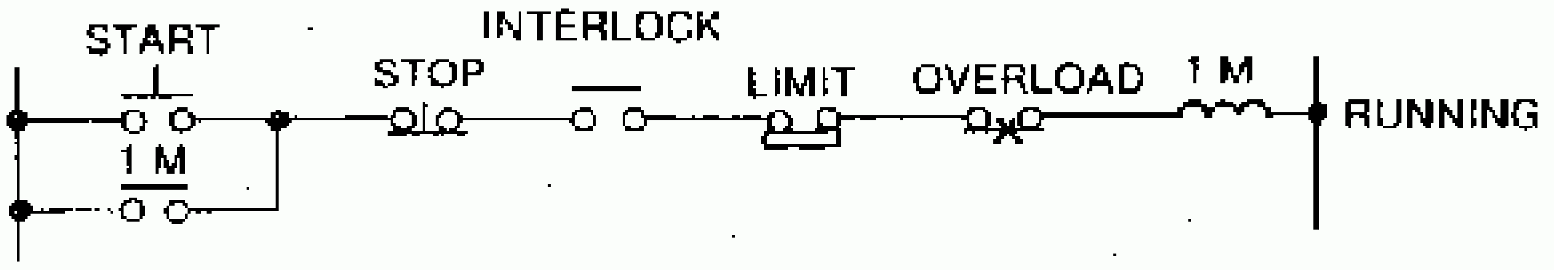
- Reference coil of —| |— 0XXXX is —( )— 0XXXX
- Reference coil of —| |— 1XXXX is input relay —| |— 1XXXX

†Coil 02049 cannot be programmed, because it is used as a battery monitoring coil. But its contact can be used in any network.

(2) Example Relay Logic Circuit

Fig. 4.1 shows an example of relay logic circuit.

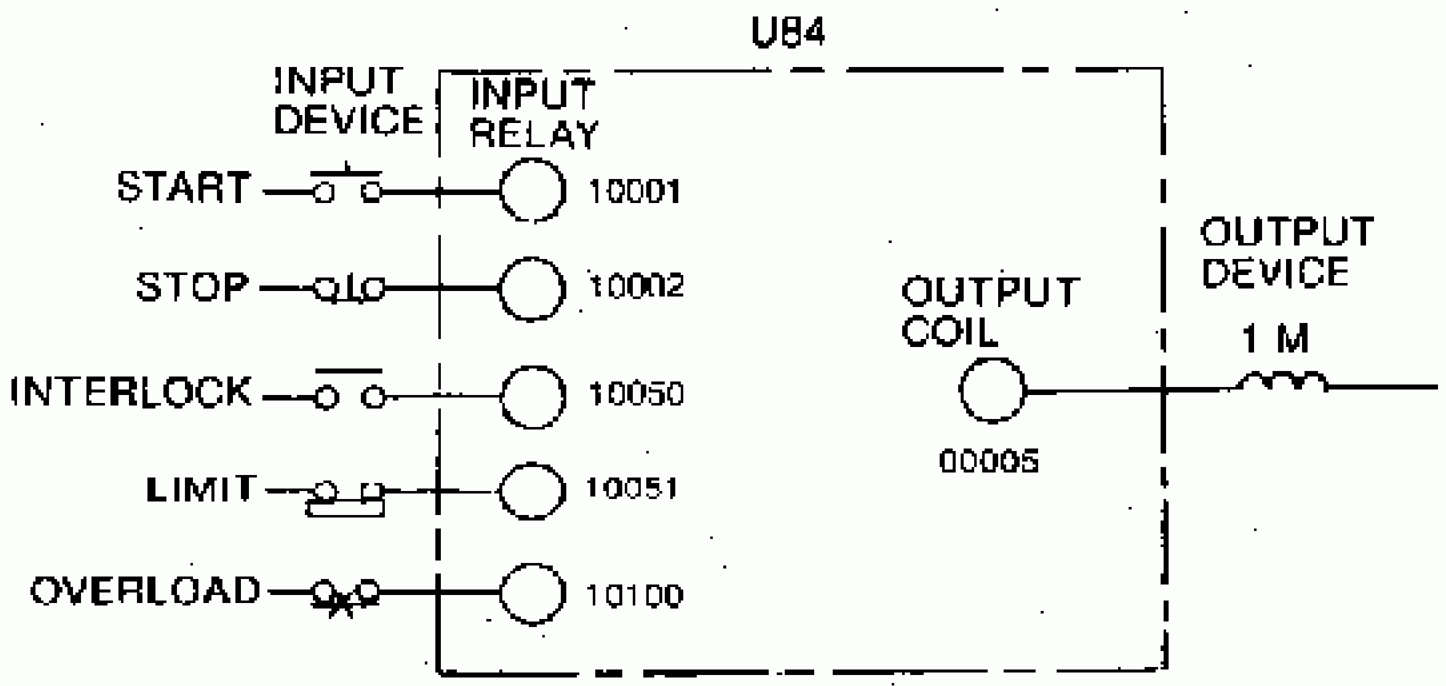
- If the logic in Fig. (a) is to be implemented in the U84 controller, the control elements must be connected to input circuits in the I/O configuration and outputs assigned.



(a) Example Relay Logic

⇩

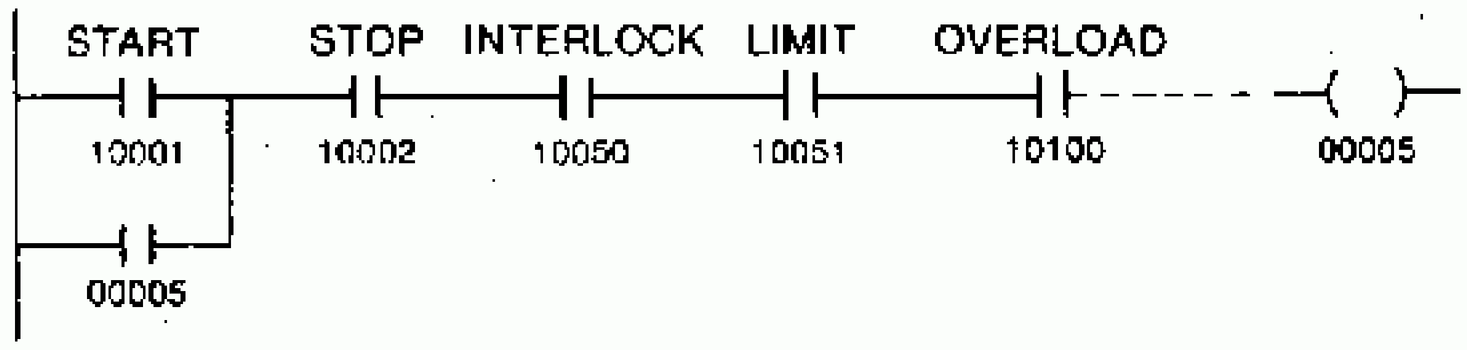
- Fig. (b) illustrates assumed input assignments and wiring details. Output number 00005 is assigned to operate the external device.



(b) Assumed I/O Wiring

⇩

- The resultant internal logic to be programmed by the user is shown in Fig. (c).



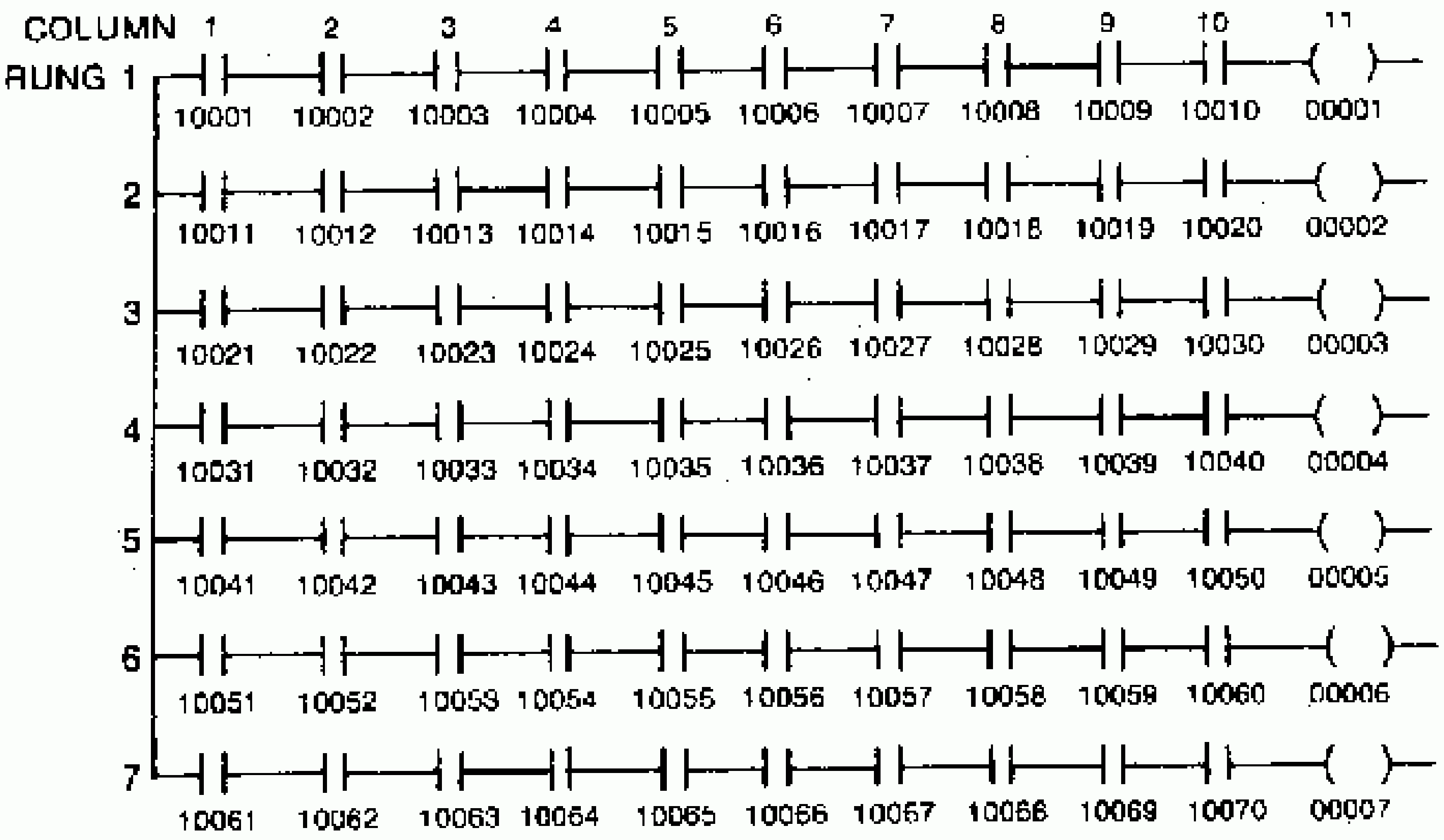
(c) Equivalent U84 Program

Note:
1. In this example, the stop, interlock, limit and overload are normally on for safety from failure. Thus they are normally open contacts in the ladder diagram. Whether a signal is a normally open (NO) or a normally closed (NC) contact should be decided in the design stage.
2. The reference number of a coil is given, like "00005," just as appearing on the CRT screen of the programming panel. On the drawing, "-( )" may be written instead of "00005"
   5

Fig. 4.1 Sample Relay Logic Circuit

4. 2. 2 Creating of Relay Circuits

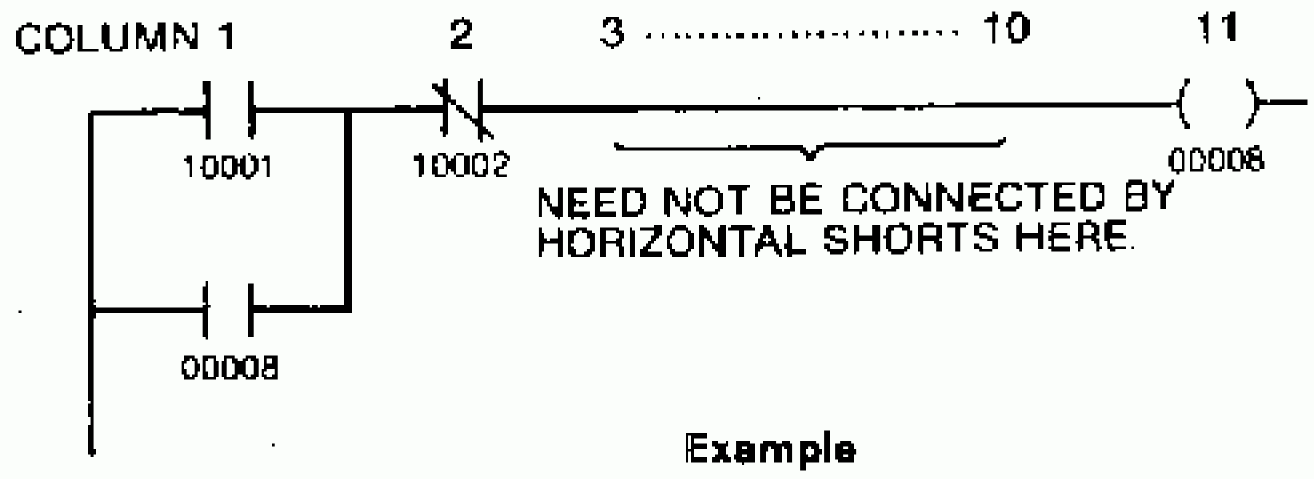
(1) In a network, contacts and horizontal short elements (shunts) may exist at any intersection of the matrix of 7 lines by 10 columns. Coils may exist on column 11. Thus up to 70 contacts and seven coils can be used in a network.



NET #1

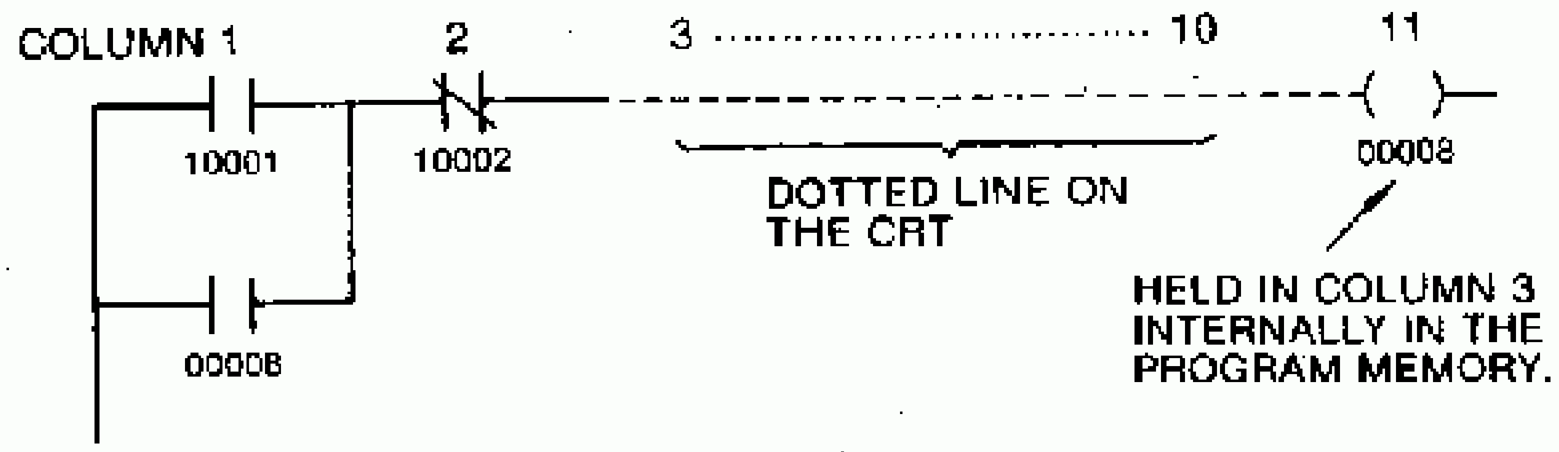
Example

(2) If a relay circuit consisting of contacts and a uniting element ends at column 2 (see below), columns 3 to 10 need not be connected by horizontal shorts before placing a coil in column 11.



Example

By placing a coil in column 3 on the screen of the programming panel, it will automatically appear in column 11 as shown below.

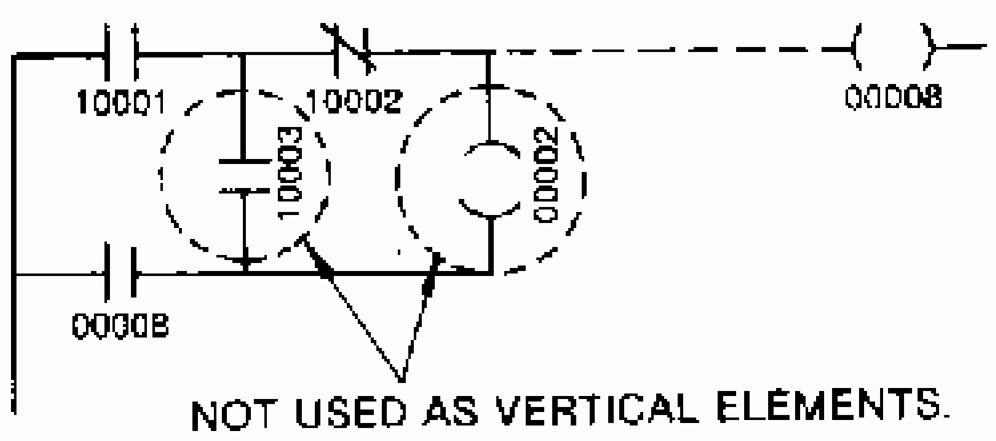


NET #2

Example

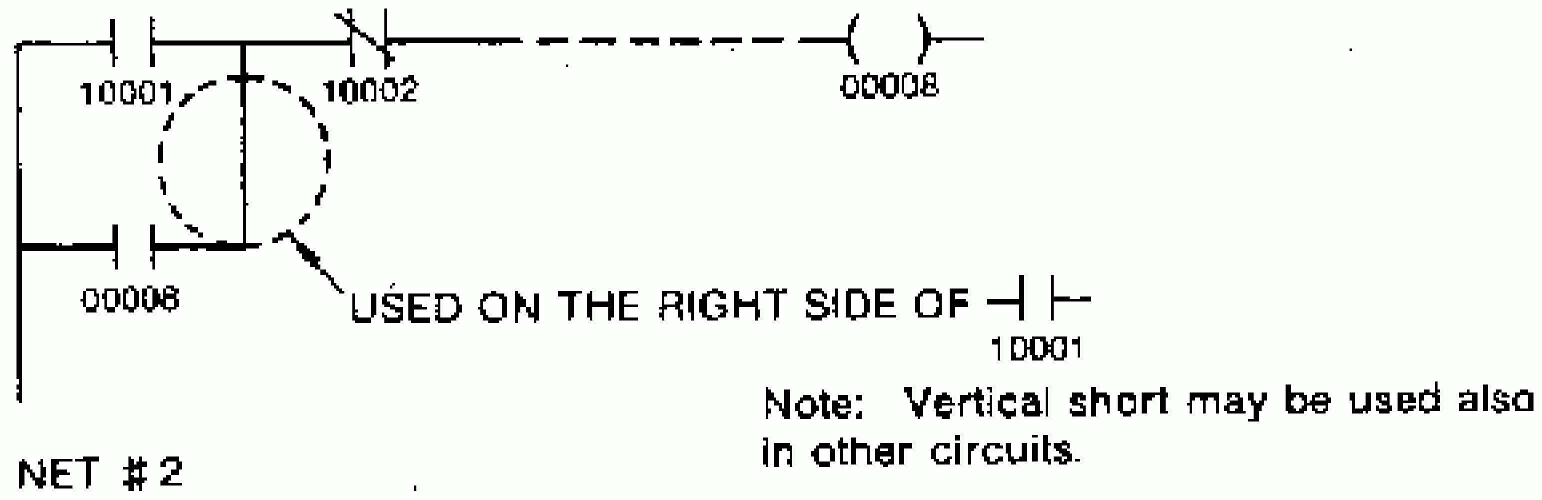
4. 2. 2 Creating of Relay Circuits (Cont'd)

(3) Contacts and coils are used as horizontal elements, but not as vertical elements.



Example

(4) A vertical short may exist on the right side of a contact or horizontal shorting element for downward connection (to the next line), but not on line 7 or with any coil.

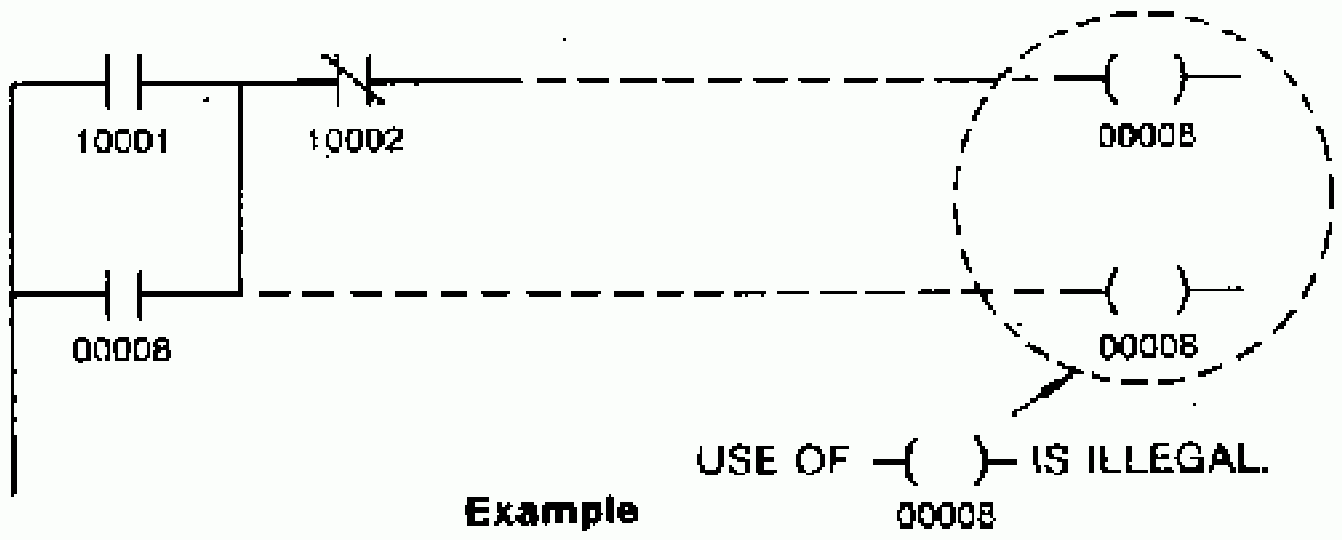


Example

(5) A vertical open is only used to cancel a vertical short.

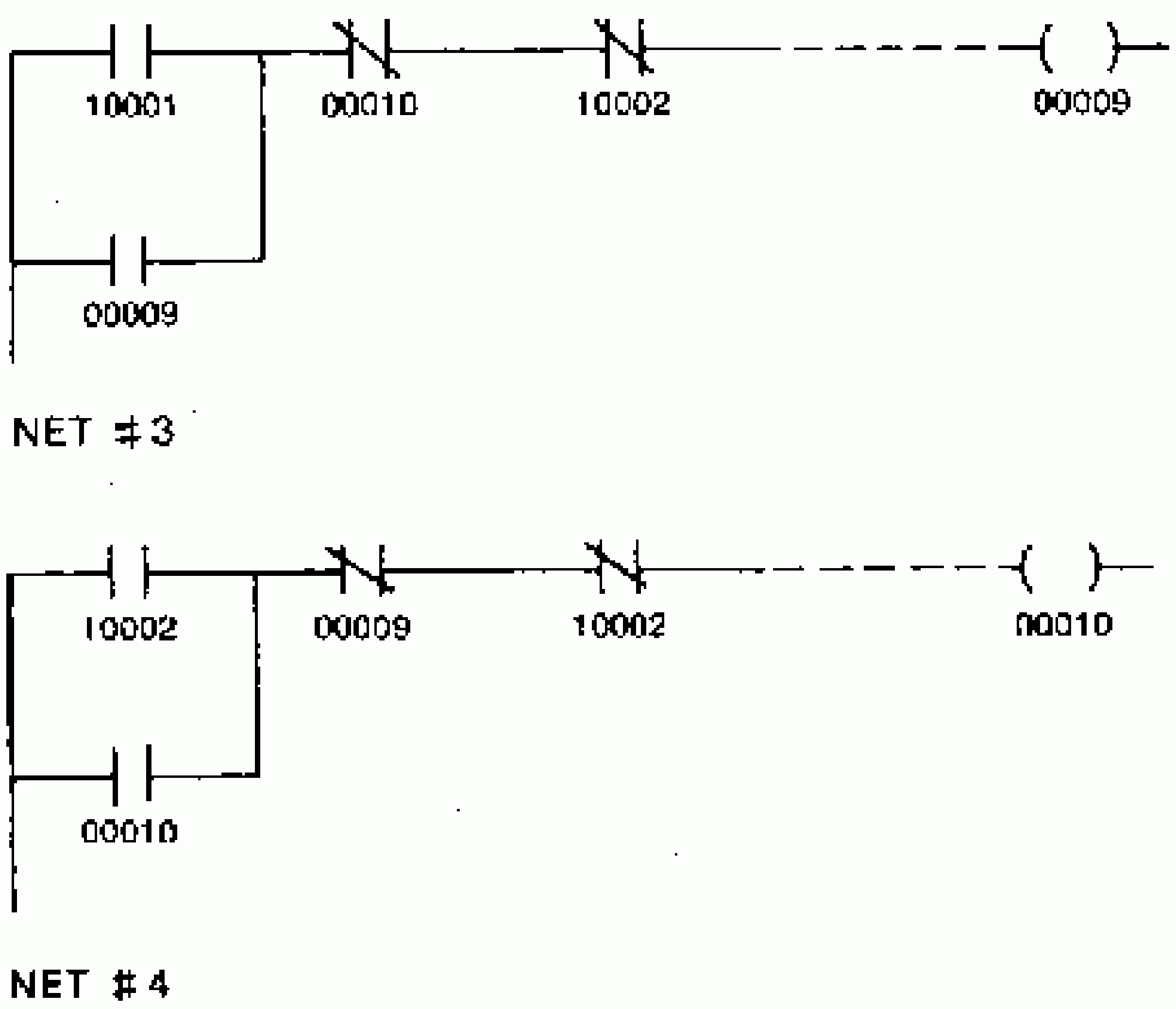
(6) Vertical shorts and opens are not placed at intersections of the 7 lines-by-10 columns matrix and therefore they occupy no memory locations.

(7) A reference number cannot be given to two or more coils.



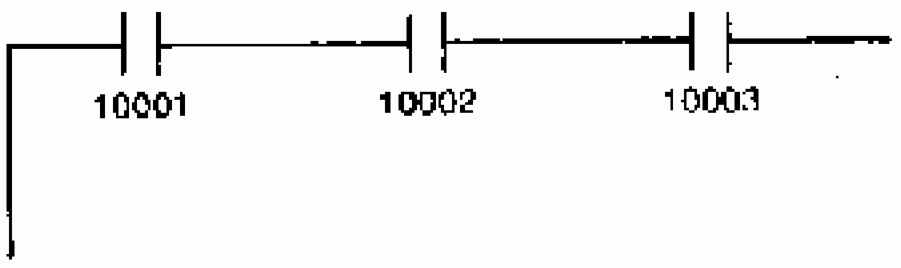
Example

(8) The contact of a coil and input relay (contact) may be used as a NO contact, a NC contact or transitional contact many times.



Example

(9) A relay circuit without coil is not useful, but not erroneous.

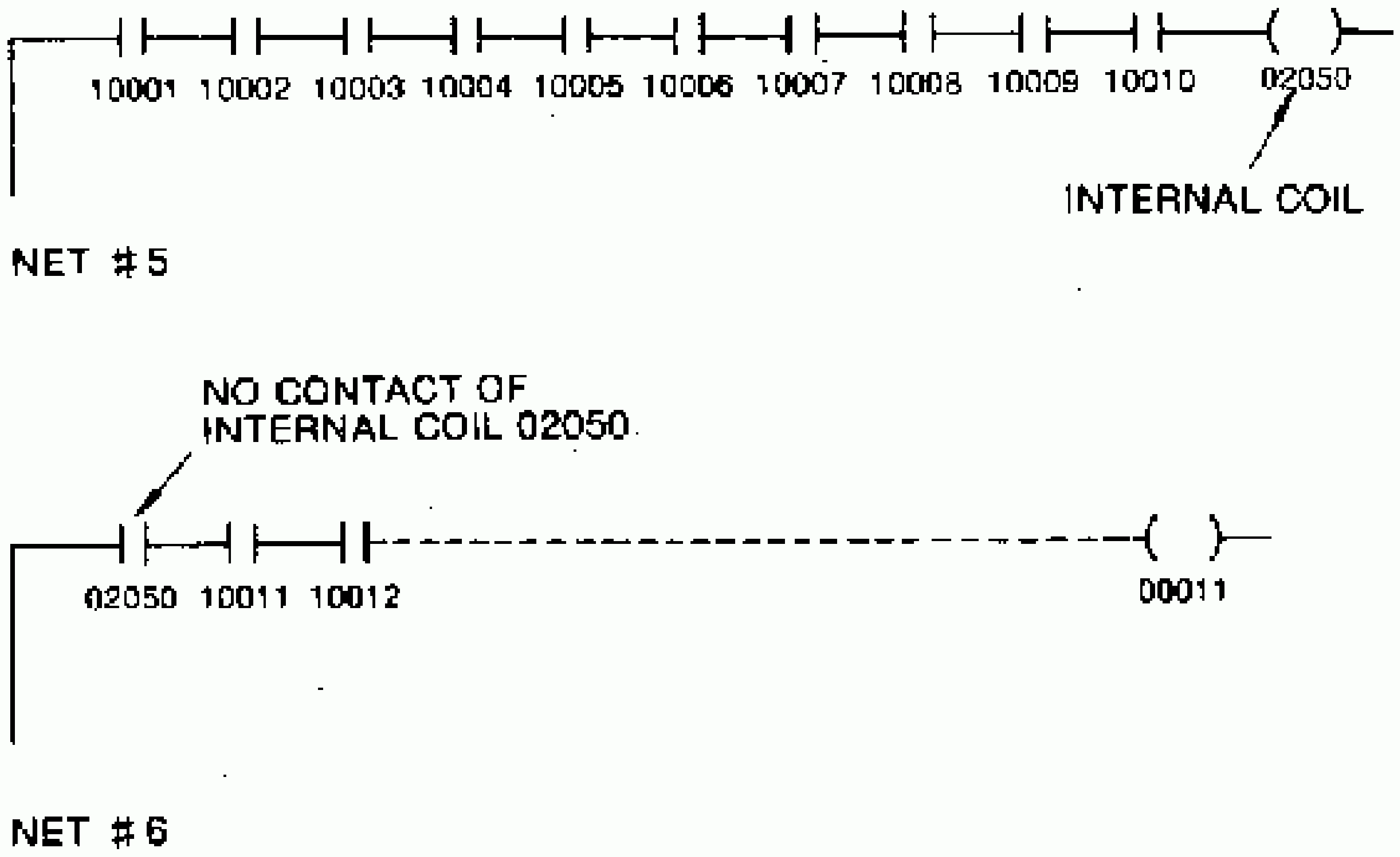


Example

(10) An internal coil is used to extend the number of elements, if more than 10 contacts arranged in series or more than seven contacts arranged in parallel are required.

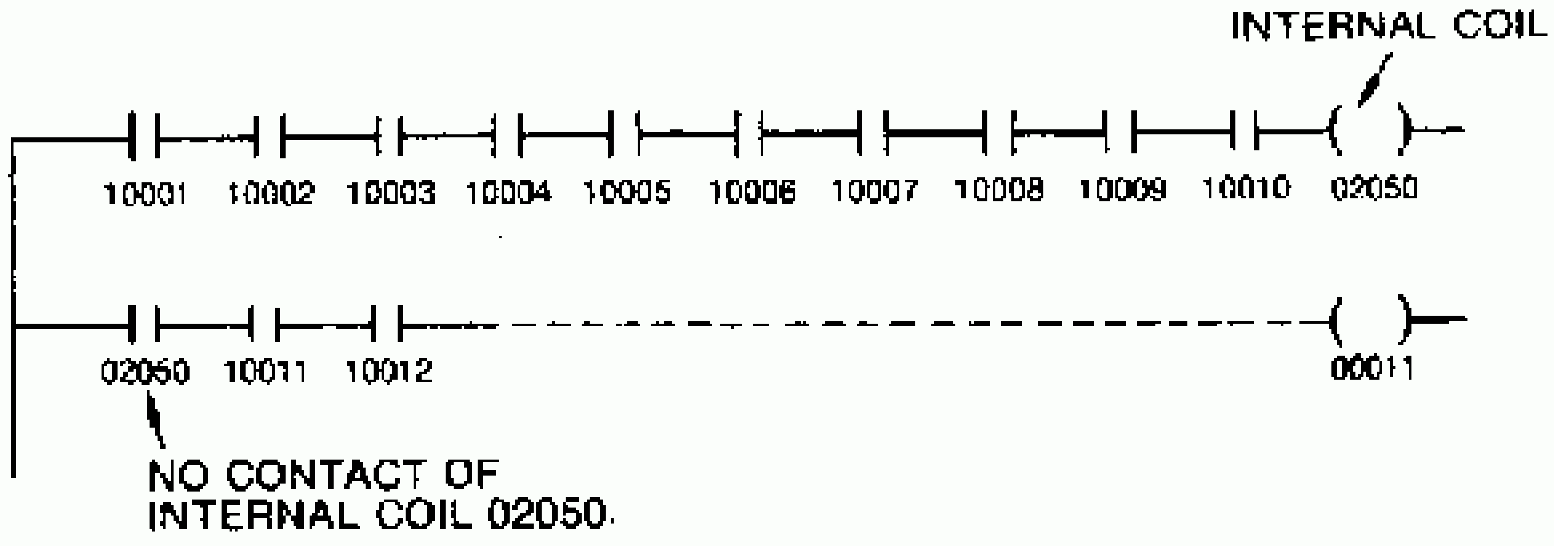
(a) Examples of Series Circuit

Examples 1 and 2 show how to make up coil 00011 with a series circuit of NO contacts 10001-10012.



Coil 00011 is turned on during the scanning cycle when all input relays 10001-10012 all are turned on.

Example 1



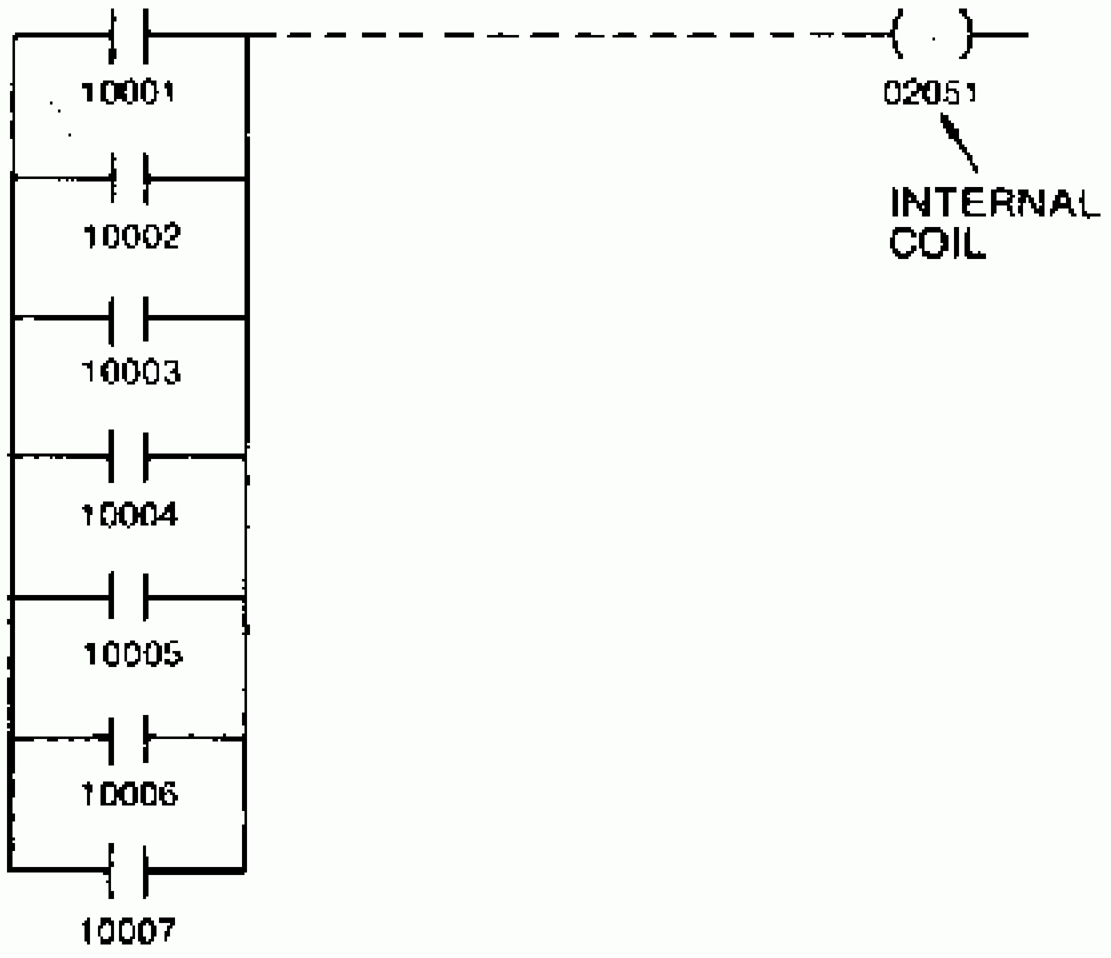
Coil 02050 is turned on during the scanning cycle when all input relays 10001-10012 are turned on, but coil 00011 is turned on during the next scanning cycle (with a delay of one scanning cycle).

Example 2

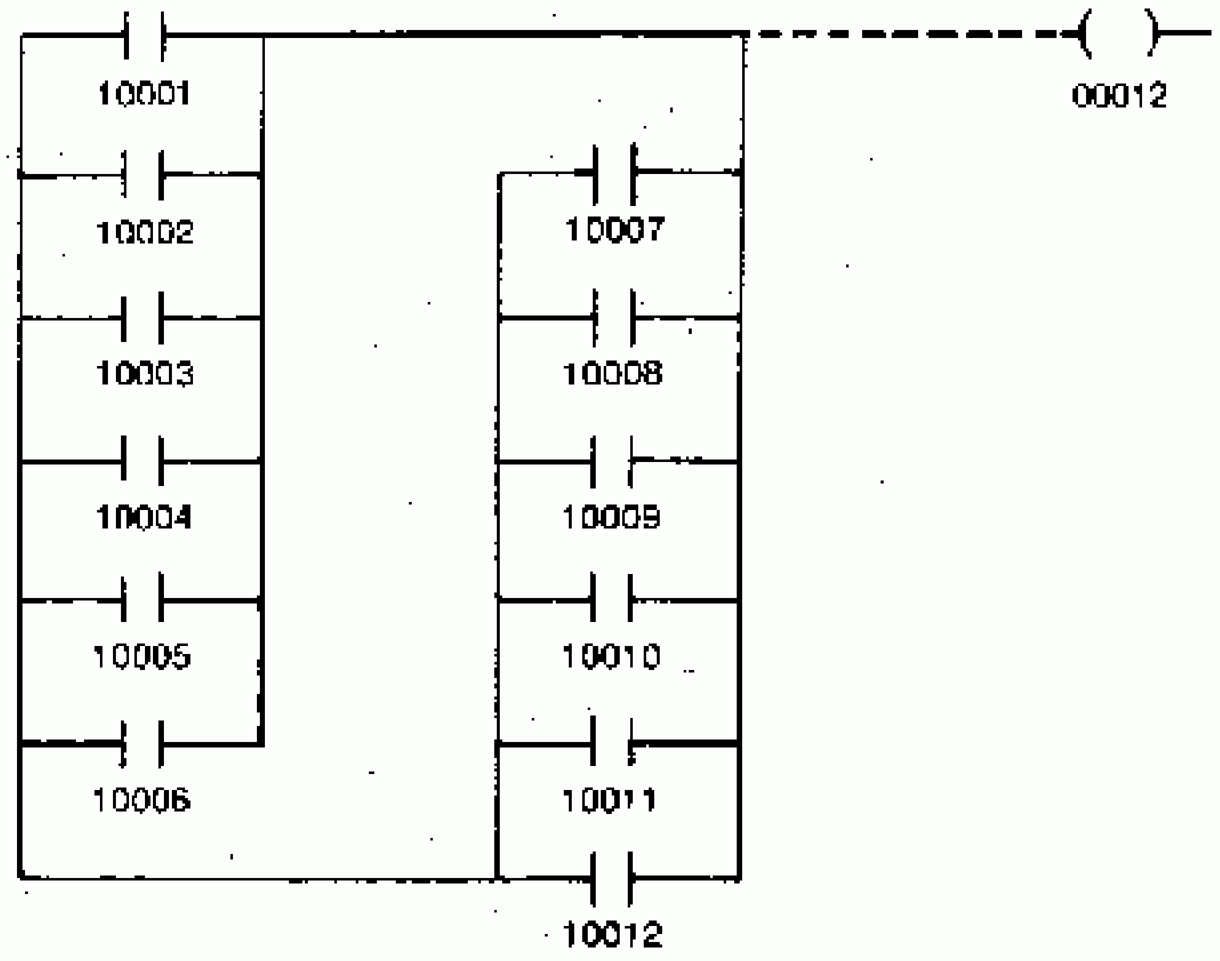
4. 2. 2 Creating of Relay Circuits (Cont'd)

(b) Examples of Parallel Circuit

Examples 1 and 2 show how to make up coil 00012 with a parallel circuit of NO contacts 10001-10012.

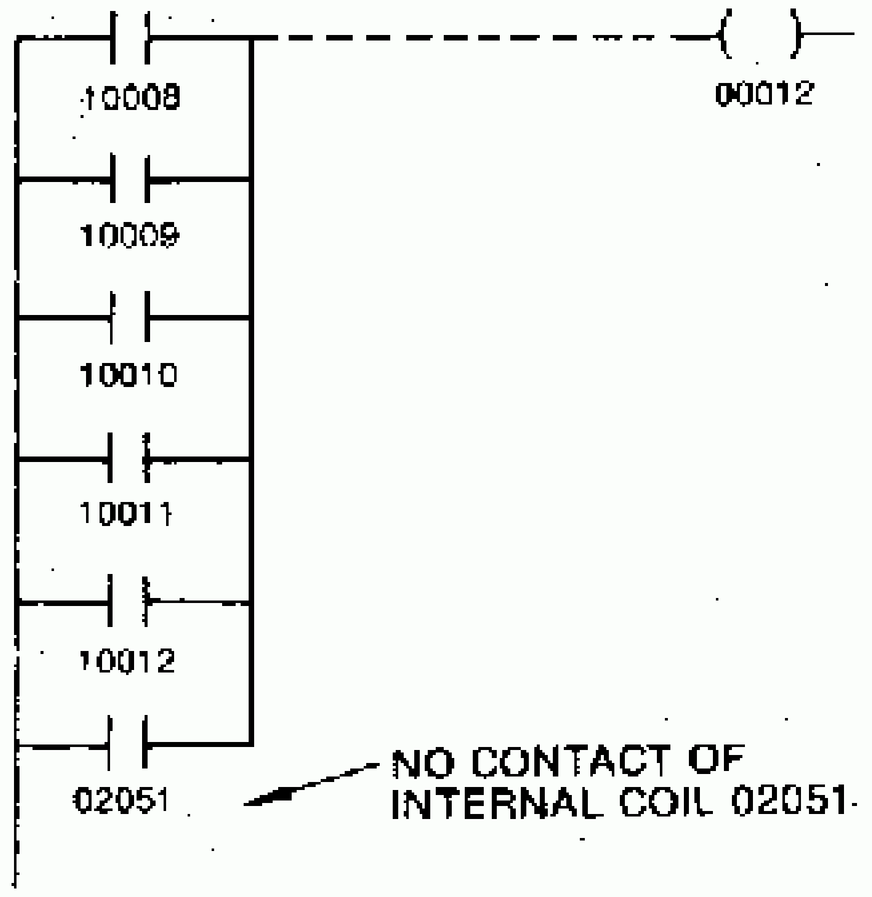


NET #7



Note. No internal coil is used in this case.

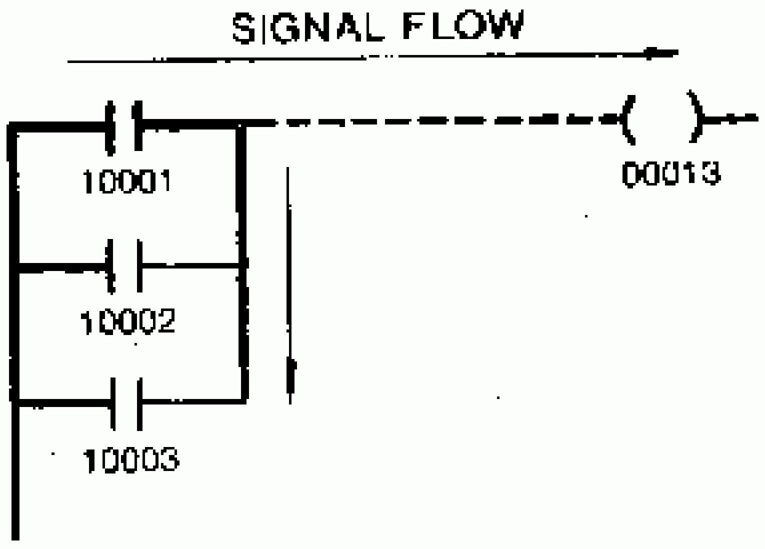
Example 2



NET #8

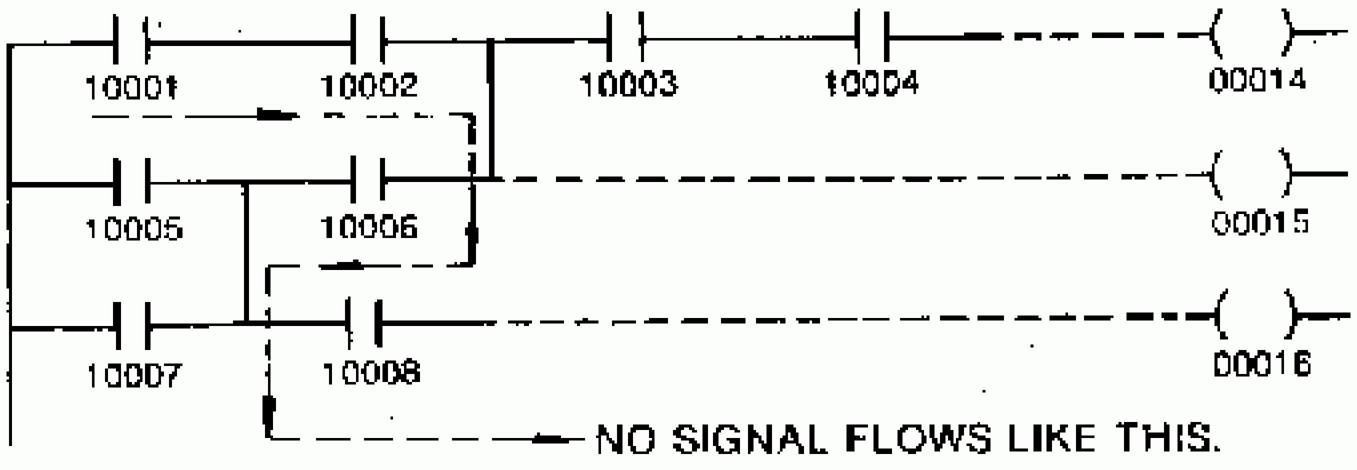
Example 1

(11) In a relay circuit, a signal always flows from the power rail (left side) to the coils (right side). Vertically it flows either from top to bottom or bottom to top.



When input relay 10001 is on, a signal flows as shown by arrows and coil 00013 is turned on. On the CRT screen of the P190 programming panel, flows of signal (called "power flows") are indicated by bold lines as shown here

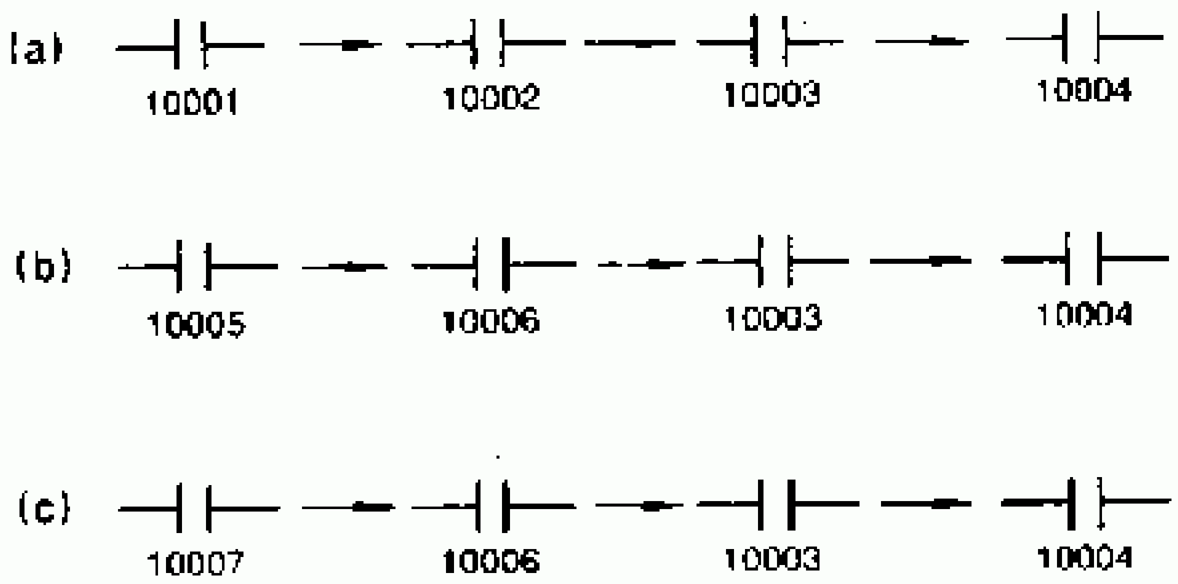
Example 1



NET #9

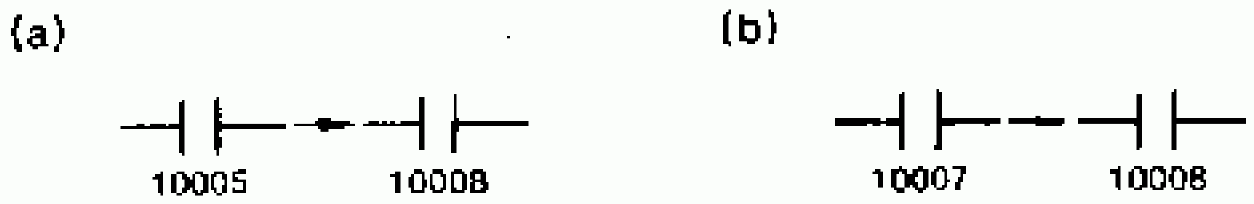
Example 2

(a) Paths to turn on coil 00014



(b) Paths to turn on coil 00015

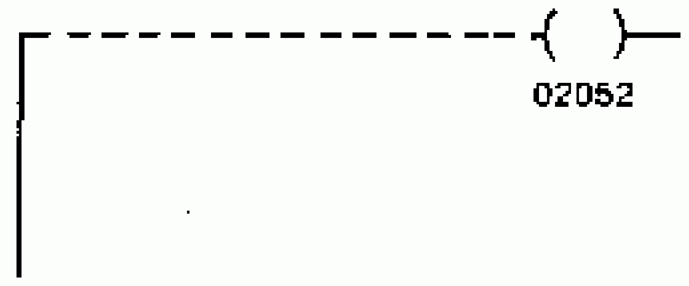
(c) Paths to turn on coil 00016



(d) Coil 00016 does not turn on through a path as shown with dotted line and therefore it is not necessary to consider a roundabout path.

4. 2. 3 Sample Application of Relays Circuits

(1) Normally-ON Circuit

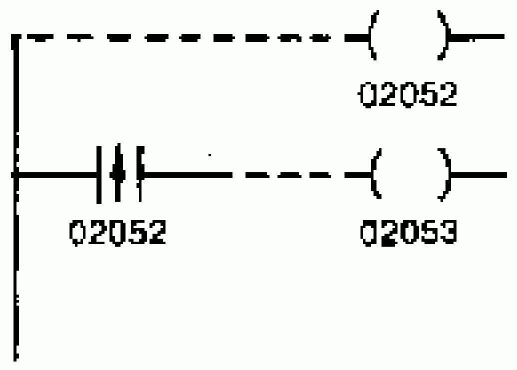


NET # 10

Example

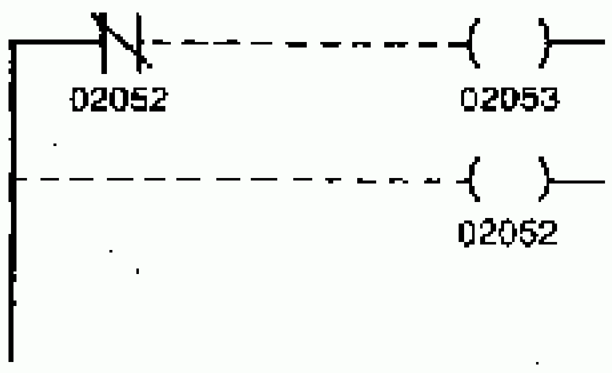
- Coil 02052 is ON as long as the U84 is operating properly.
- This is used, for example, in a pulse generating circuit for intialization.

(2) Pulse Generating Circuit for Initialization



NET # 10

Example 1

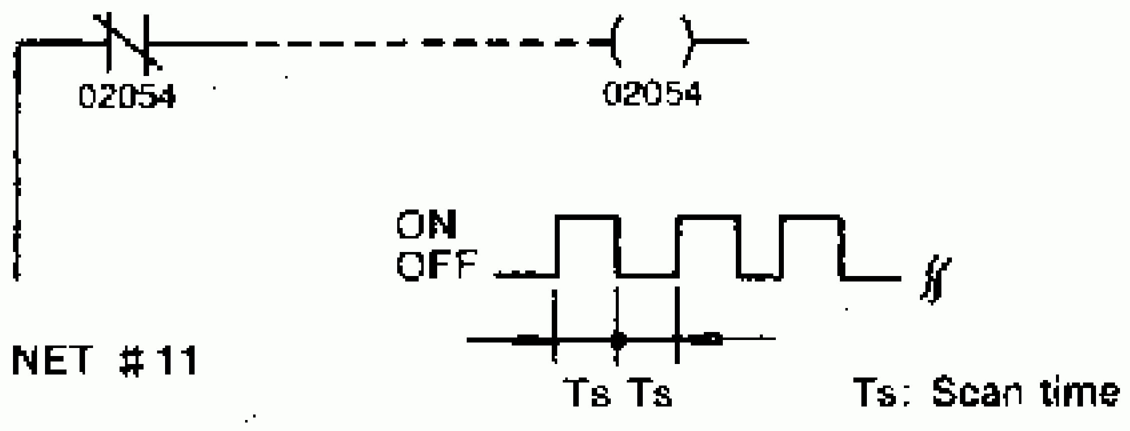


Example 2

- Coil 02053 is turned on only during the first scanning cycle after the U84 power is turned on. Examples 1 and 2 are equivalent.
- These are used to set/reset memory circuits, clear the current values of timers and counters, and preset arithmetic constants for initializing the internal logic after the U84 power is turned on.

Note: Program the circuits of examples 1 and 2 in a network preceding one using these circuits.

(3) Oscillator

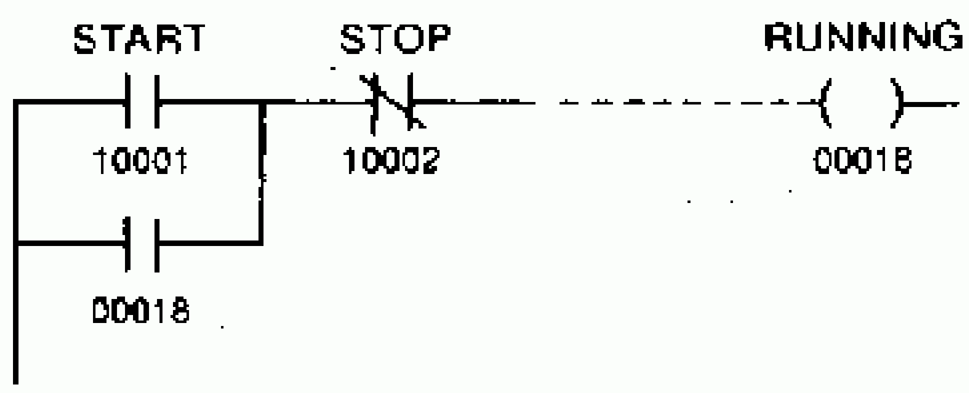


NET # 11

Example

- Coil 02054 repeats turning on and off every two scanning cycles.
- This is used to perform an arithmetic operation every two scanning cycles.

(4) Self-Holding Circuit (Memory Circuit)

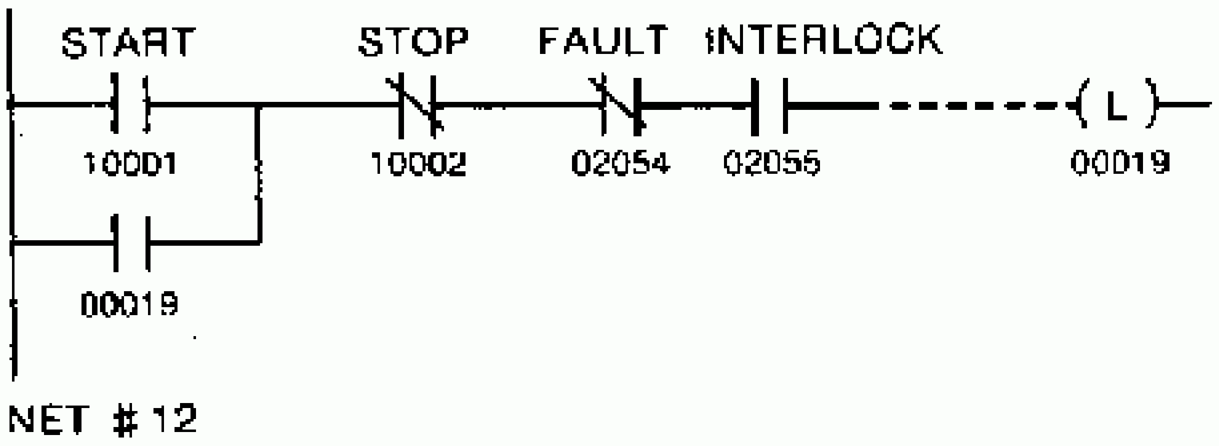


NET # 12

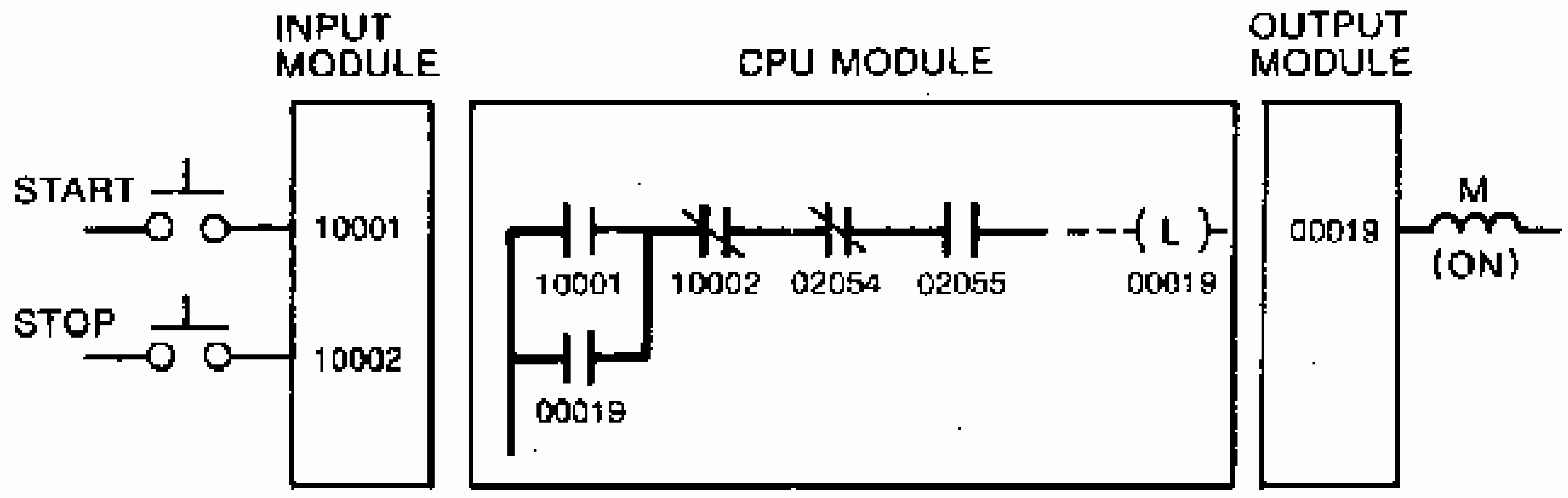
Example

- As in an ordinary relay circuit, an NO contact of coil is placed in parallel with memory set signal.
- When memory set signal (input relay 10001) comes ON, coil 00018 is turned on and self-held. When memory reset signal (input relay 10002) come ON, the coil is released from self-holding.
- This is used for start and stop operation with a pushbutton switch which resets automatically.

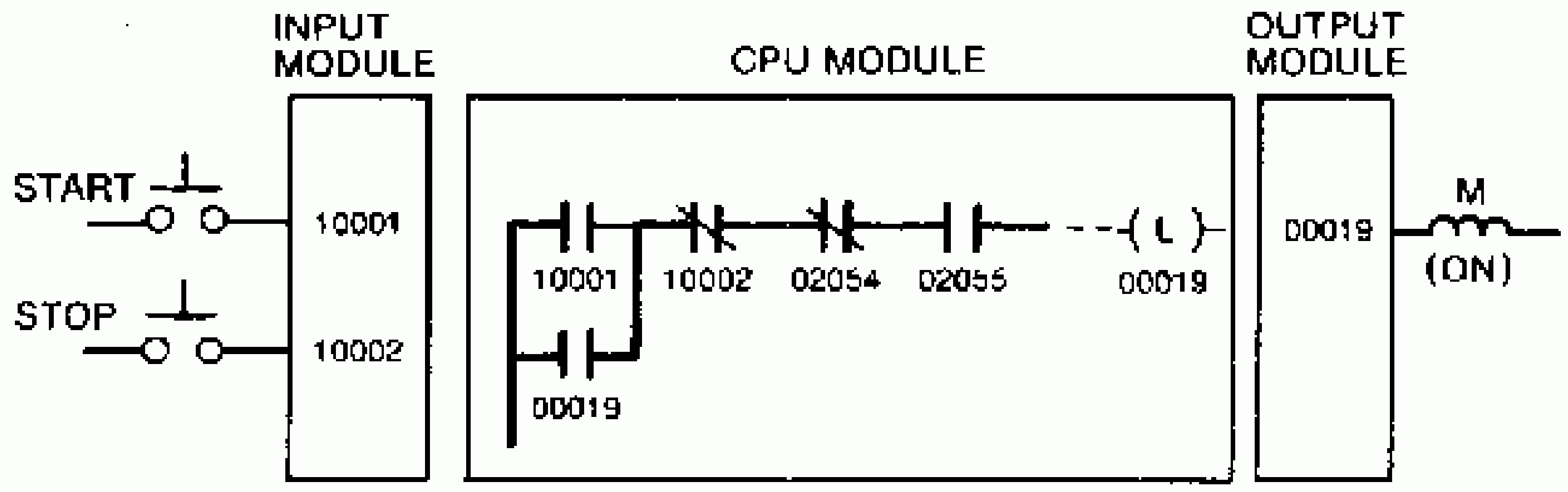
(5) Latch Relay Circuit
(Self-Holding Circuit with Memory during Power Failure)



When a latch coil is used in a self-holding circuit, the status before power failure will be restored after recovery. Any coil can be used as a latch coil.



(a) Before Power Failure



(b) After Recovery

Note: To use input signal as the reset (OFF) signal (10002 shown above) of a latch relay circuit, read it in as an NO contact then program it as an NC contact (to prevent a latch coil from resetting even if source power of I/O modules stop prior to that of the CPU at power failure).

To use a coil as the OFF condition (02054 above) or holding condition (02055 above) of a latch relay circuit, check if it is ON or OFF in reference to the network number during the first scanning cycle after power-on of the U84.

For example, if the network of latch coil 00019 appears later (has a greater network number) than that of coil 02054 and earlier than that of coil 02055 in the above example, the latch relay circuit will be reset if coil 02054 is ON during the first scanning cycle after power-on. It will also be reset unless coil 02055 is a latch coil and ON.

Example

(6) Transitional Contact Circuit

The transitional contact remains on only for one scanning cycle when the associated reference coil has turned on or off. Any input relay or coil may be used as a reference coil.

NOTE

The transitional contact to the associated reference coil in the skipped network is not operated correctly. The contact remains ON or OFF.

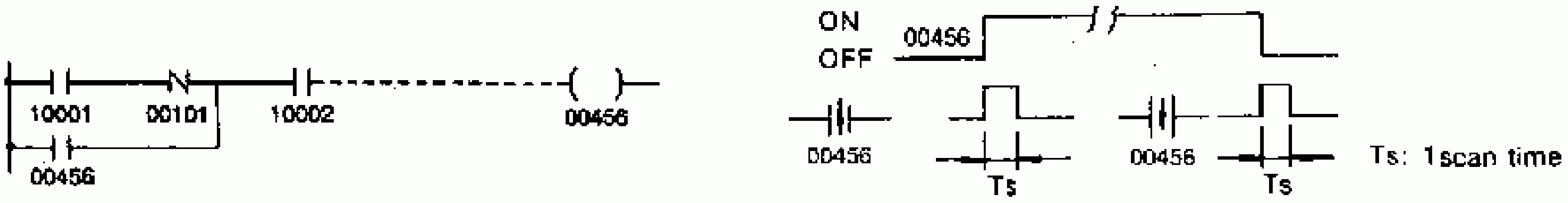


Fig. 4. 2 Transitional Contact

4.3 TIMERS

4.3.1 Types of Timers

(1) Types

Three types of timers are available as shown in Table 4.3. As many timers as desired used in a range of the program memory capacity and the number of holding registers.

Table 4.3 Types of Timers

Type	Symbol	Unit of Time	Limit of Count
Timer (Seconds)	T1.0	1 sec	1–9999 sec
Timer (Tenths of Seconds)	T0.1	0.1 sec	0.1–999.9 sec
Timer (Hundredths of Seconds)	T.01	0.01 sec	0.01–99.99 sec

(2) Unit of Time

A timer counts up in certain units of time. The 1-second timer, for example, counts in units of seconds.

(3) Limit of Count

A timer can count up to a certain limit of time. The 1-second timer, for example, counts in the range of 1 to 9999 seconds. The upper limits of count are presettable.

4.3.2 Timer Configuration

(1) Form

Fig. 4.3 shows the basic form of a timer. A timer is built vertically and needs two elements (top and bottom). Specify any of constant K, reference number 3XXXX of input register, or reference number 4XXXX of holding register, referring to Table 4.4. TXXX identifies a specific type of timer. Specify any of T1.0, T0.1, or T.01 referring to Table 4.3.

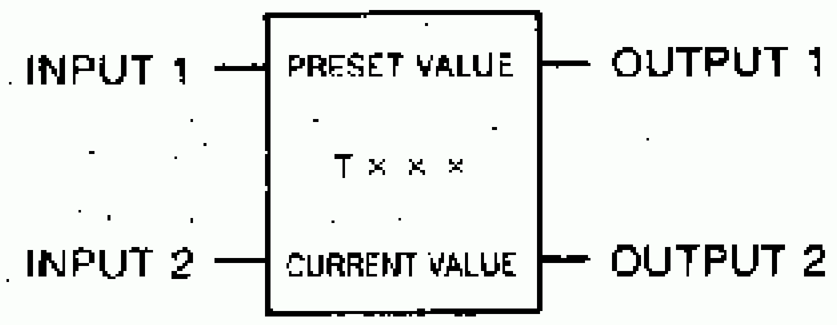


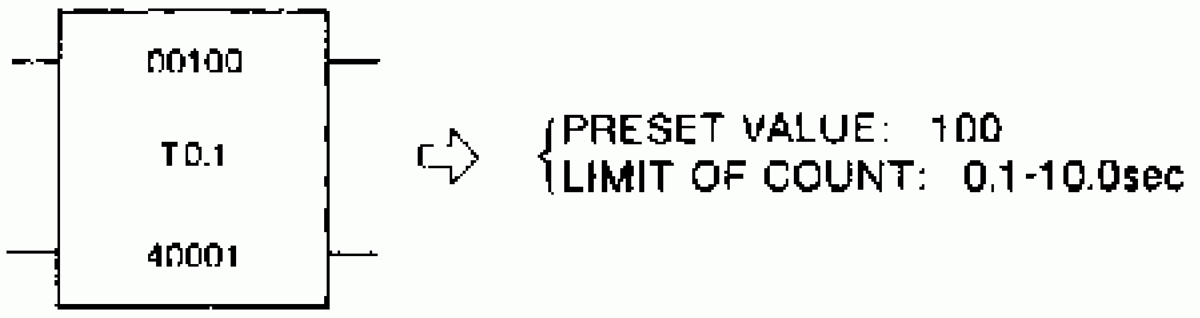
Fig. 4.3 Timer General Form

Table 4.4 Timer Elements

Element	Specified Numbers	Description
Top	Any of the following: Constant K (00001–09999) 3×××× (30001–30256) 4×××× (40001–49999)	The preset value of timer is constant K, or contents of 3×××× or 4××××.
Bottom	4×××× (40001–49999)	The current value of timer is stored in 4××××.

(2) Preset Value

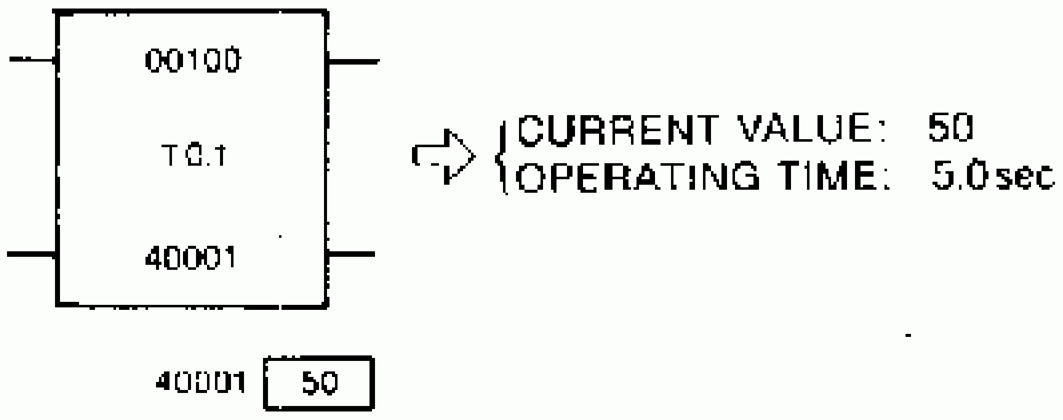
Designate a value of 1 to 9999 to determine the range (variable) of timer.



Example

(3) Current Value

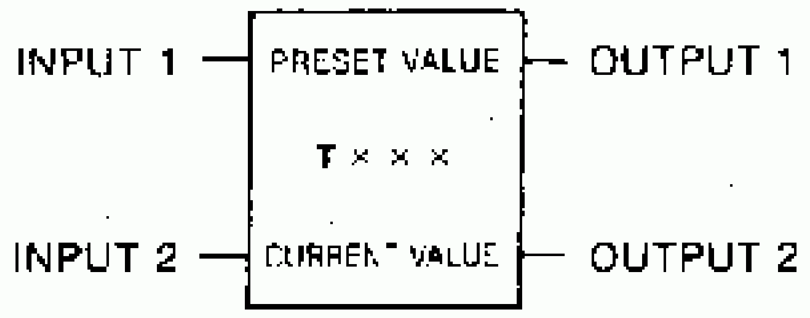
This is the value a timer has counted up.



Example

4. 3. 3 Function and Operation of Timer

(1) Timer Function



When input 2 is ON, the timer adds up the time intervals while input 1 is ON. When the current value becomes equal to the preset value, it stops counting with output 1 turned on and output 2 off. When input 2 is OFF, the timer does not count up regardless of the status of input 1. At this time, output 1 is kept off and output 2 is on (the current value is 0).

(2) Timer Operation

Fig. 4.4 and Table 4.5 show the timing chart and operation of the timer.

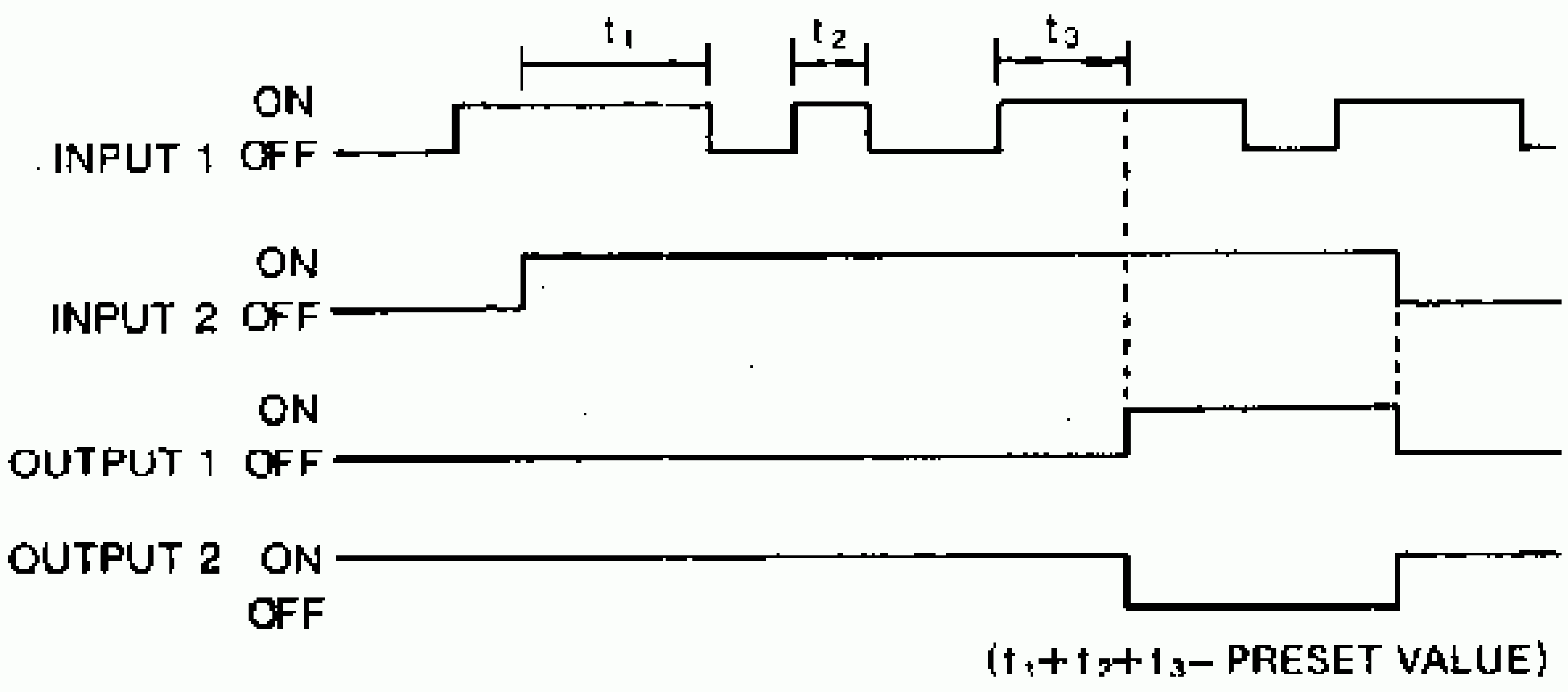


Fig. 4. 4 Timer Operation

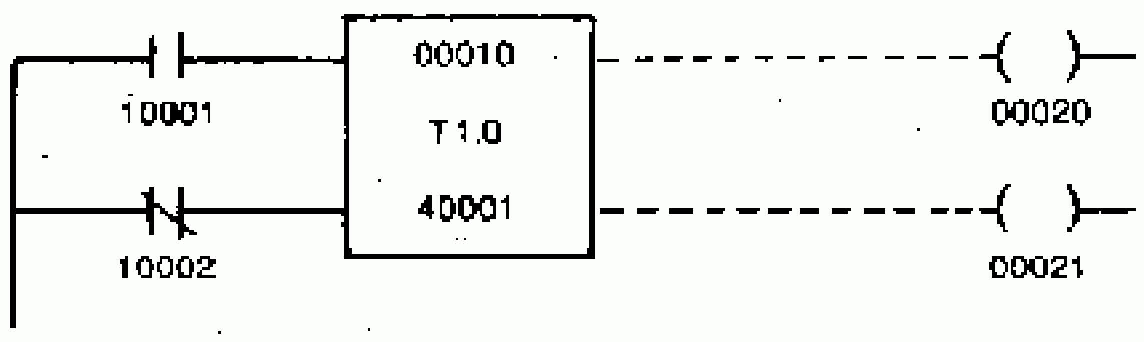
(2) Timer Operation (Cont'd)

Table 4.5 Timer Operation

Input Status		Timer Status		Current Value	Output Status	
Input 1	Input 2				Output 1	Output 2
ON OFF	OFF	Reset status		0	OFF	ON
ON	ON	Operating status	Current value < Preset value	Increase	OFF*	ON*
			Current value = Preset value	Preset value	ON	OFF
OFF	ON	Standstill status	Current value < Preset value	Constant	OFF	ON
			Current value = Preset value	Preset value	ON	OFF

*As a result of increasing current value, current value < preset value.

(3) Sample Timer



NET #13

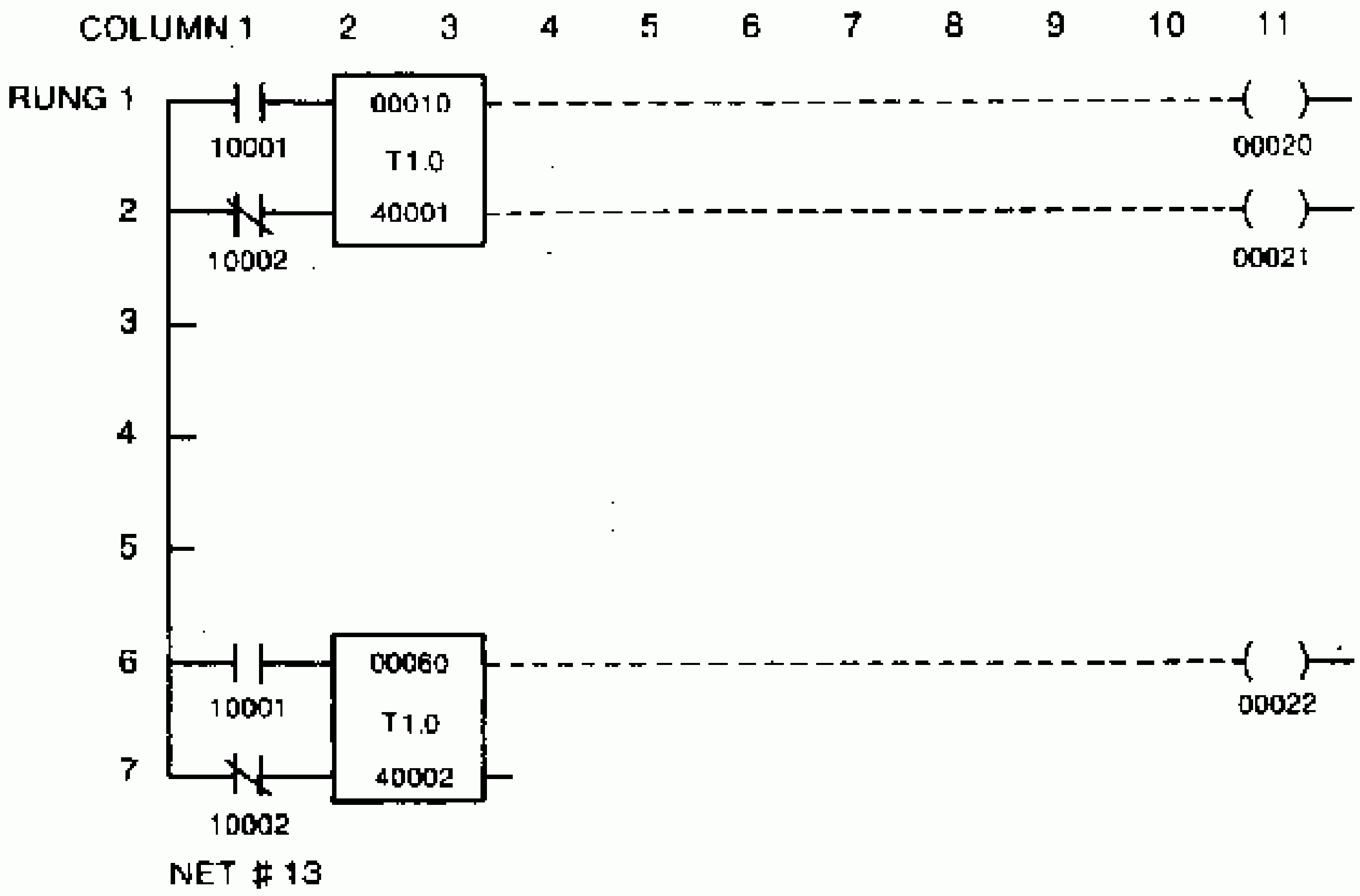
Example

- This timer counts up only when input 2 is ON (input relay 10002 is OFF).
- If input 1 (input relay 10001) is turned on while input 2 is ON, the timer is operated and the current value (contents of 40001) is increased by one every second.
- When the current value is equal to the preset value (10), the timer stops and output 1 (coil 00020) is turned on and output 2 (coil 00021) off.
- If input 2 is ON and input 1 is cycled ON-OFF-ON-OFF, the time intervals while input 1 is ON are added. See Fig.4.4.
- When input 2 is OFF, the timer does not count up regardless of the status of input 1. At this time, the current value is 0, output 1 OFF, and output 2 ON.

4. 3. 4 Programming Timer Circuit and Precautions

(1) Programming Timer Circuit.

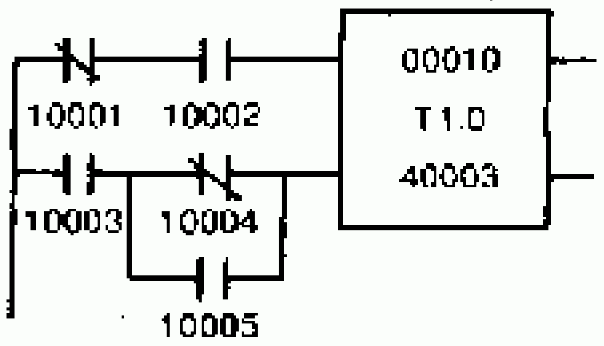
A timer needs two elements placed vertically (top and bottom) in a network. It can be used at any intersection of the 7 lines-by-10 columns matrix, but the top element (preset value) cannot be located on line 7.



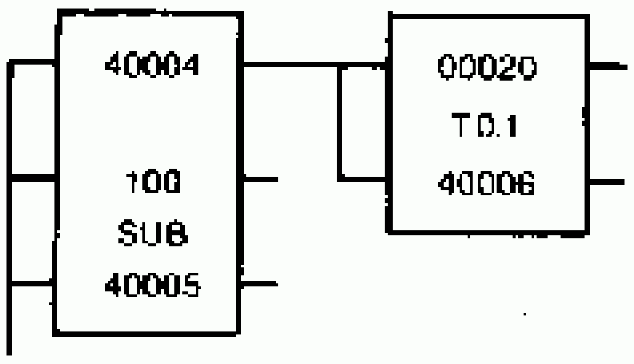
Example

(2) Timer Inputs

Inputs to the timer may be outputs of relays, other timers, counters, arithmetic operations, or data processing circuits.



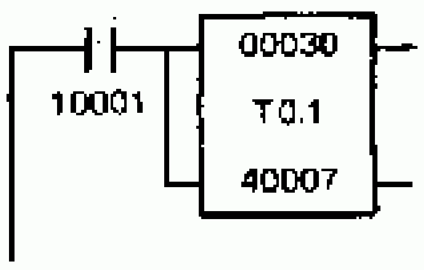
Example 1



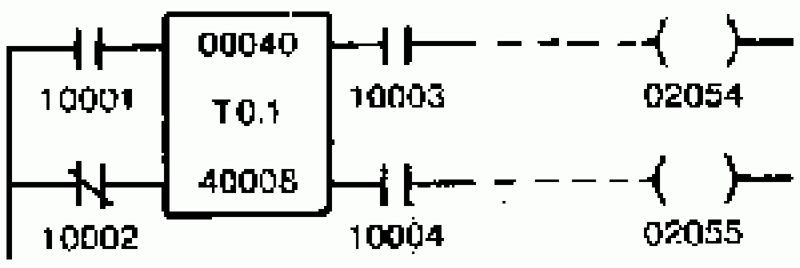
Example 2

(3) Timer Outputs

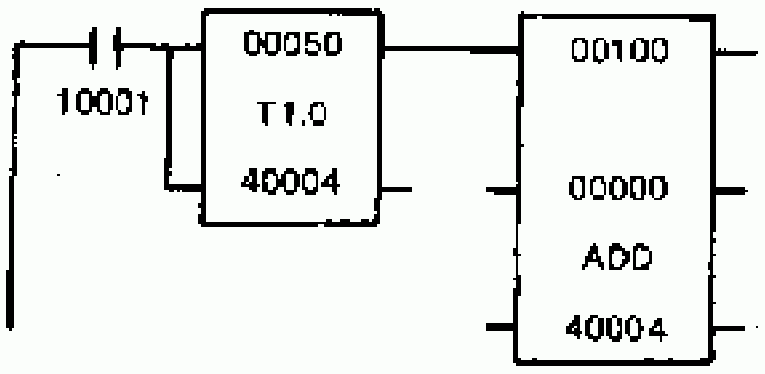
Coils may not be connected to outputs 1 and 2 of a timer. It is permitted to insert a relay contact to the right of an output or to connect an output directly to an input of a logic, except relays.



Example 1



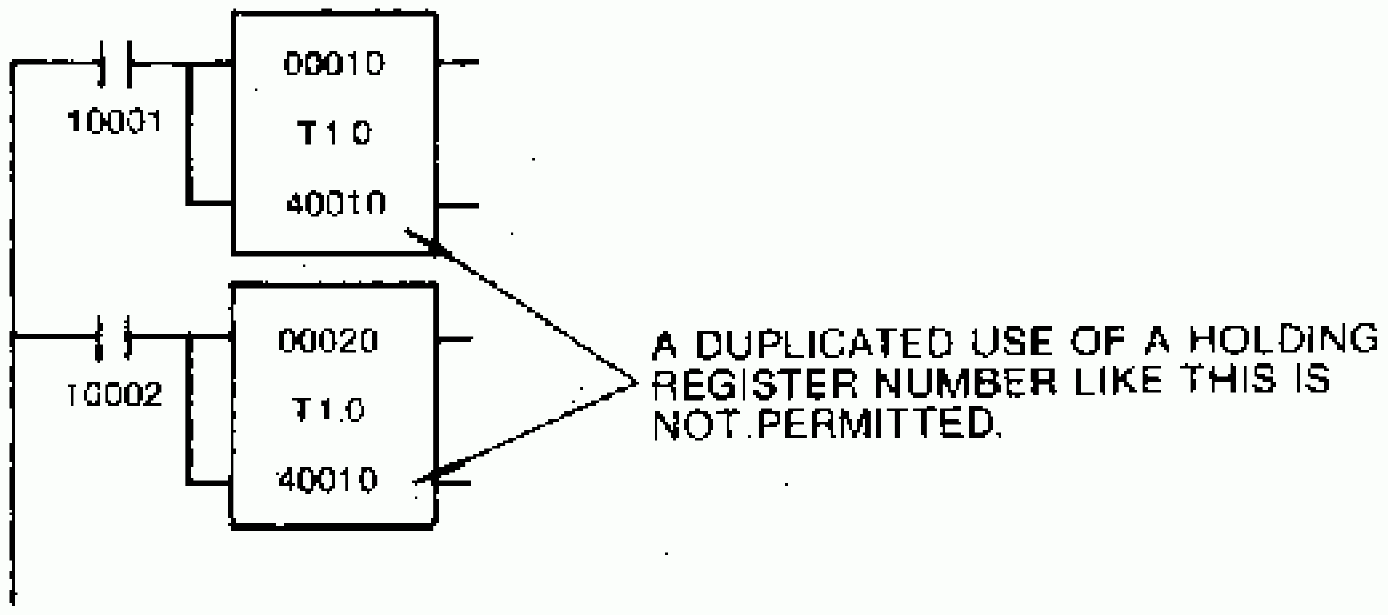
Example 2



Example 3

(4) Storing Timer Current Value

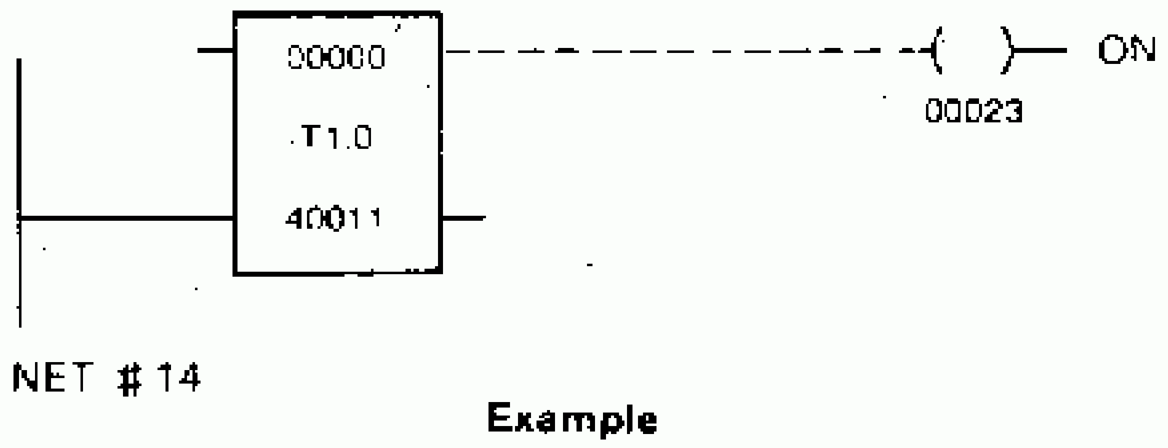
The register number of the holding register storing the current value of a timer cannot be the same as the register number of the holding register storing the current value of another timer or counter.



Example

(5) Preset Value of Timer

When the preset value of a timer is 0 and input 2 is ON, output 1 is ON (output 2 is OFF), regardless of on/off operation of input 1.



Example

(6) Relationship of Preset and Current Values

Ordinarily, the current value does not exceed the preset value, but it is possible to make the current value greater than the preset value by arithmetic operation or data transfer function. In such a case, the current value becomes equal to the preset value (output 1 is turned on) as soon as the timer circuit is solved.

(7) Timer Error

Error of the timer is given as follows.

Maximum value of error = Unit of preset time + 1 scan time

For example, if T1.0 is used for a 1-second timer, error may rise to 1 second (the timer may reach the limit a second earlier than the real time limit). Therefore, the use of T0.1 or T.01 is recommended in this case.

(8) Current Value of Timer at Power ON

During power failure, the timers of the U84 memorize the values before power failure. When the U84 is turned on, the current value of a coil will be reset to 0 or remain unchanged depending on the type and status of the reference coil of the contact used for input 2. Table 4.6 shows these relationships.

Table 4.6 Current Value of Timer at Power ON

Signal to Input 2				Current Value at Power ON
Type of Reference Coil		Status	Contact	
Input Relay Latch Coil		ON	NO	Value at power failure
			NC	0
		OFF	NO	0
			NC	Value at power failure
Coil (Not including Latch Coil)	n ≦ m	ON	NO	Value at power failure
			NC	0
		OFF	NO	0
			NC	Value at power failure
	n > m	OFF	NO	0
			NC	Value at power failure

Note

1. The number of the network including the reference coil of signal to input 2 is n and the number of the network including the timer circuit is m
2. If the signal to input 2 is composed of more than one contact, obtain the input 2 status from each contact status.
3. The current value of the timer, at the U84 power ON, is the value read when the timer circuit has been solved during the first scanning cycle after power is applied to the U84.

4.3.5 Application Timer Circuits

On-delay and off-delay timers can be obtained by vertically shorting timer inputs 1 and 2. Various timer circuits can be realized by using different signals for inputs 1 and 2.

(1) ON-delay Timer

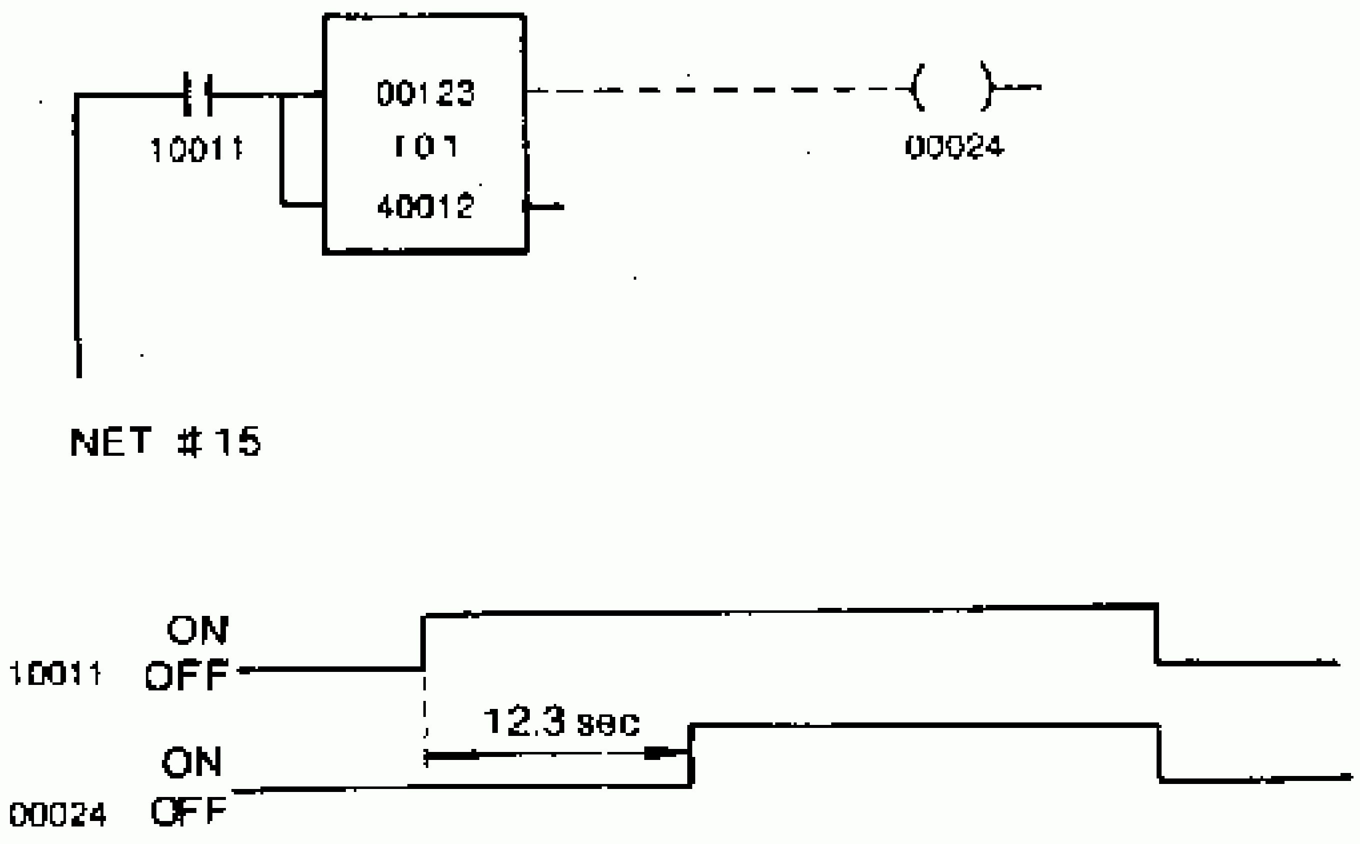


Fig. 4.5 Sample ON-delay Timer

(2) OFF-delay Timer

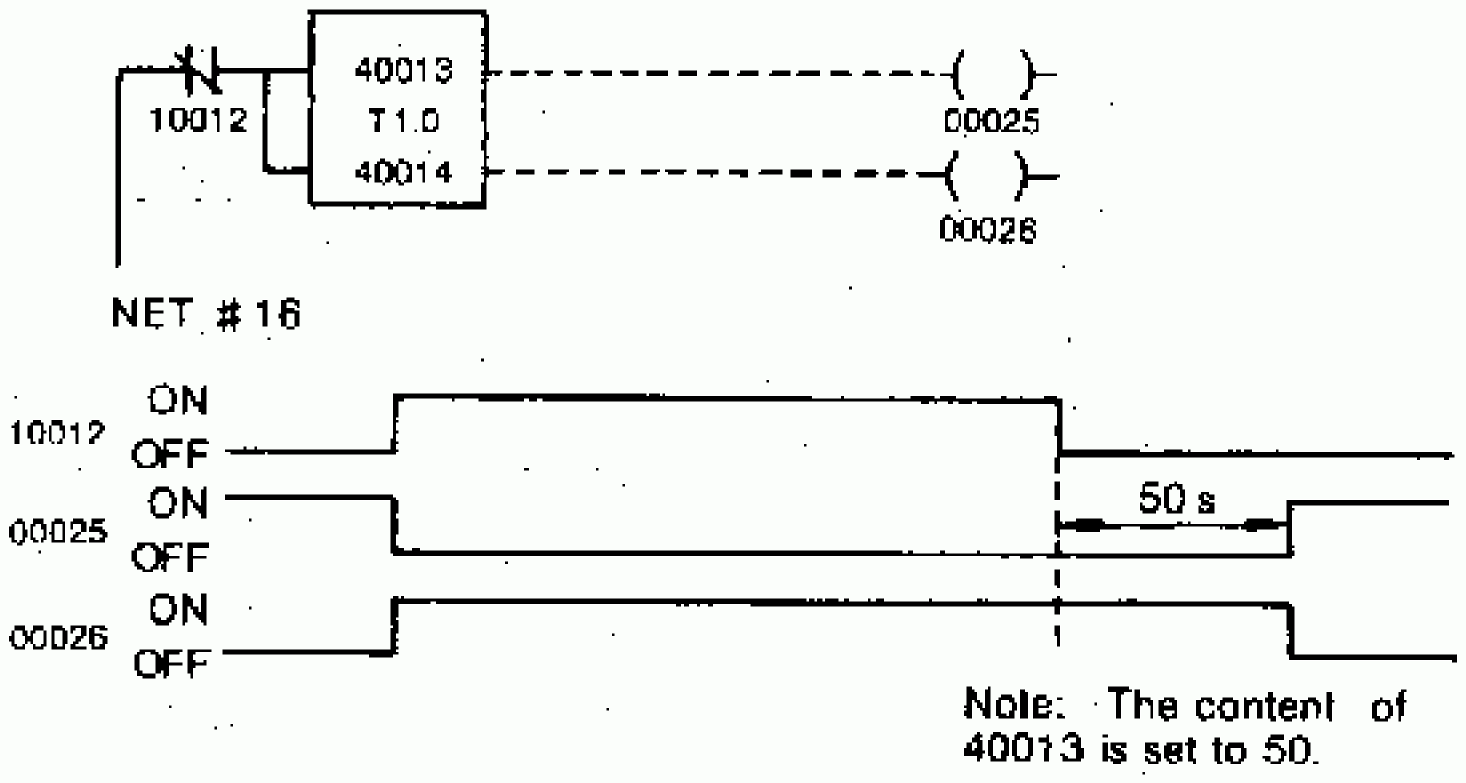


Fig. 4.6 Sample OFF-delay Timer

(3) One-Shot Timer

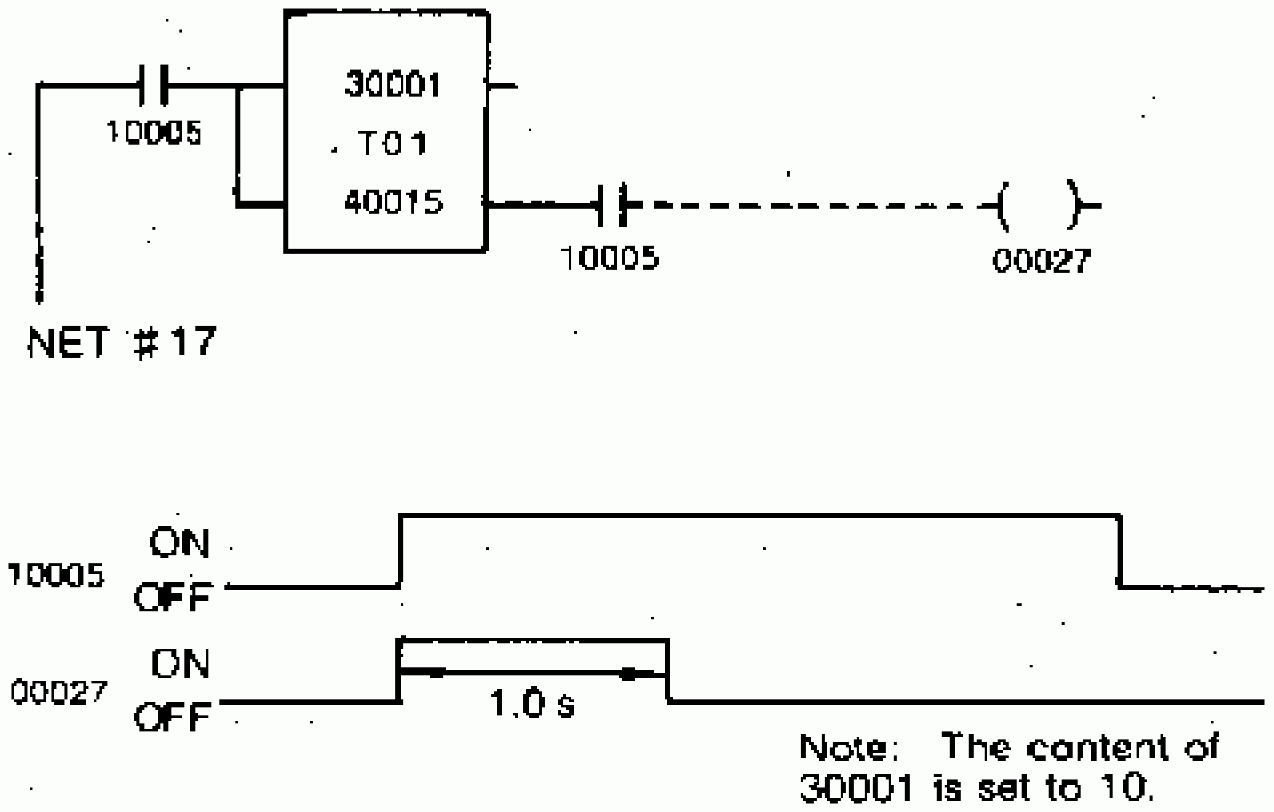


Fig. 4.7 Sample One-shot Timer

(4) Pulse Generating Circuit

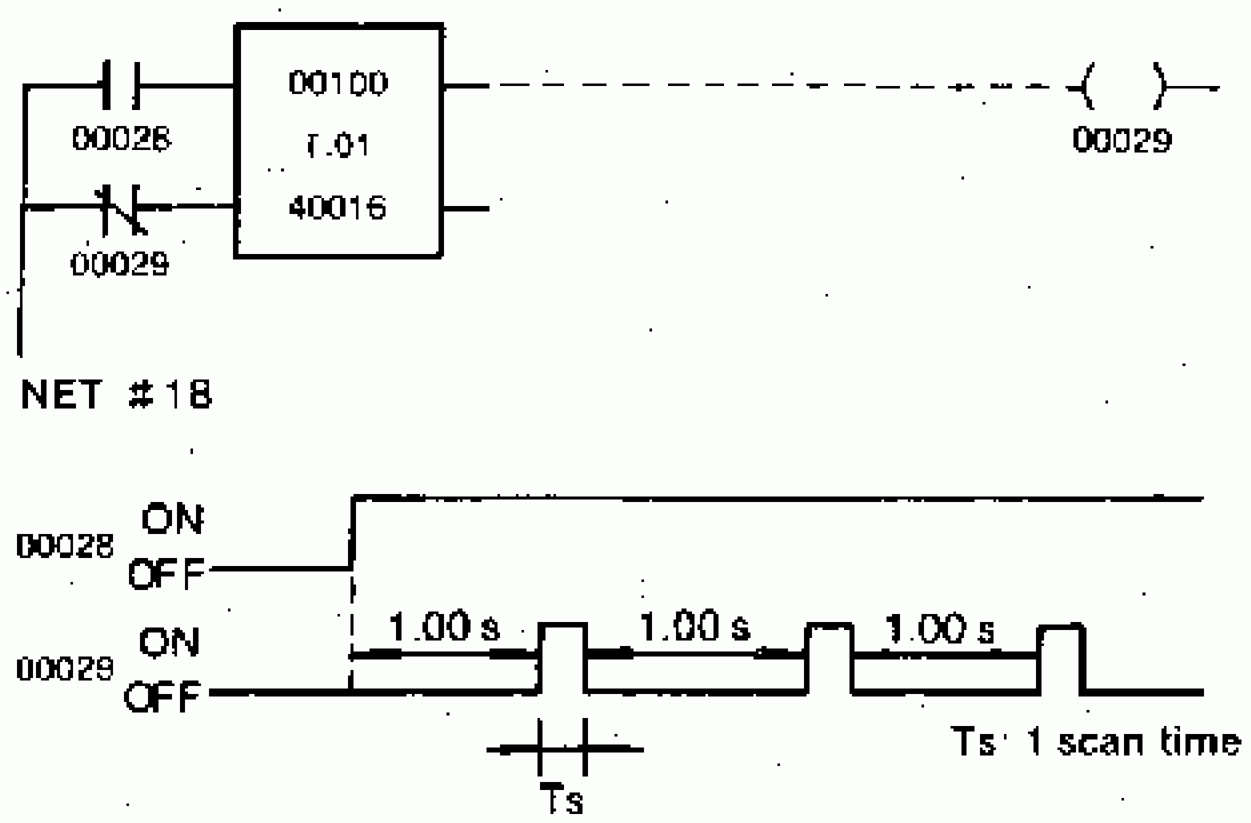


Fig. 4.8 Sample Basic Pulse Generating Circuit

(5) Flicker Relay

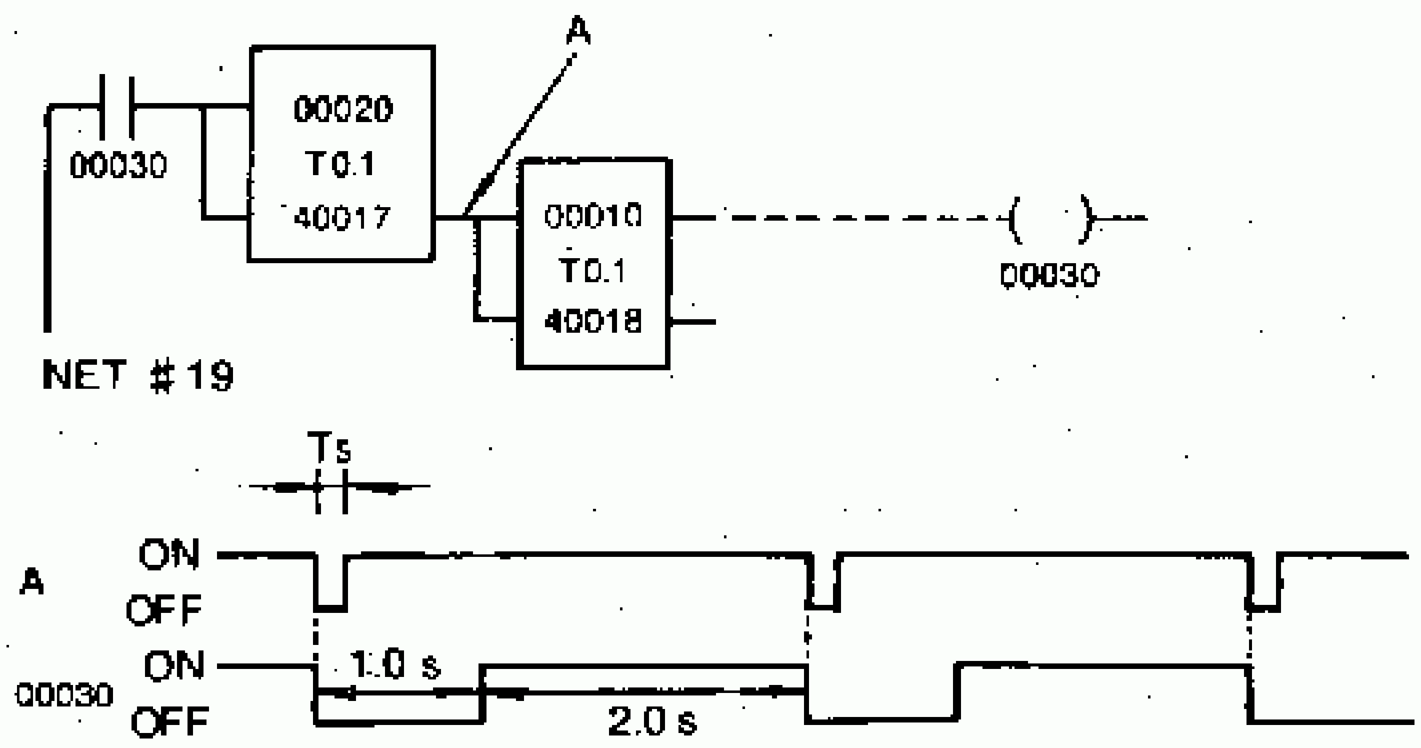


Fig. 4.9 Sample Flasher Relay

4.4 COUNTERS

4.4.1 Types of Counters

(1) Types

Two types of counters are available as shown in Table 4.7. As many counters as desired may be used, in a range of the program memory capacity and the number of holding registers.

Table 4.7 Type of Counters

Type	Symbol	Unit of Count	Limit of Count
Up Counter	UCTR	1 pulse	1–9999 pulses
Down Counter	DCTR		

(2) Unit of Count

An up or down counter increases or decreases the current count by one pulse.

(3) Limit of Count

An up or down counter increases or decreases in a range of pulses counted (1 to 9999 pulse). The upper limits of count are presettable.

4.4.2 Counter Configuration

(1) Form

Fig. 4.10 shows the basic form of a counter. A counter needs two elements placed vertically (top and bottom). Specify any of constant K, reference number 3××××of input register, or reference number 4×××× of holding register, referring to Table 4.8. ×CTR identifies a specific type of counter. Specify UCTR or DCTR in reference to Table 4.7.

Fig. 4.10 Counter General Form

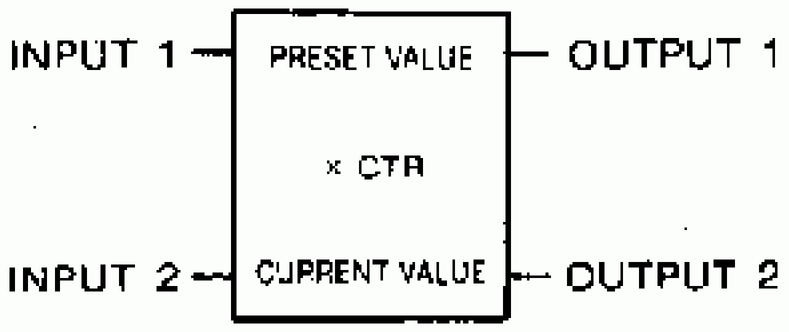
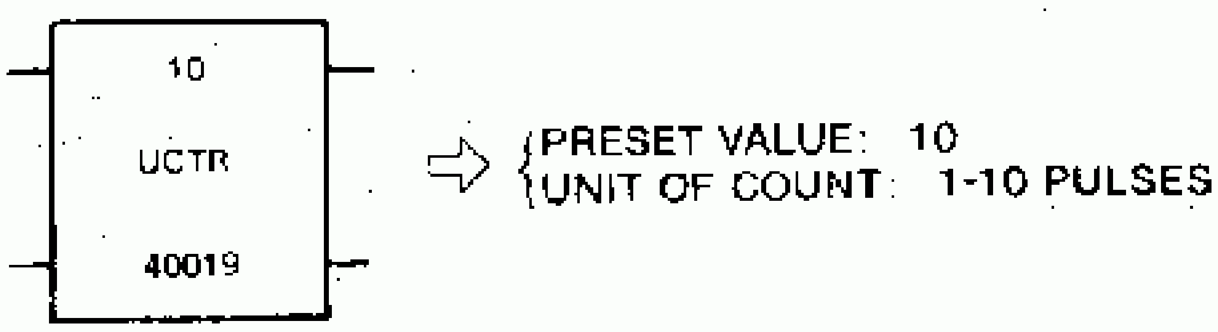


Table 4.8 Counter Elements

Element	Specified Numbers	Description
Top	Any of the following: Constant K (00001–09999) 3×××× (30001–30256) 4×××× (40001–49999)	The preset value of counter is constant K, or contents of 3×××× or 4××××.
Bottom	4×××× (40001–49999)	The current value of counter is stored in 4××××.

(2) Preset Value

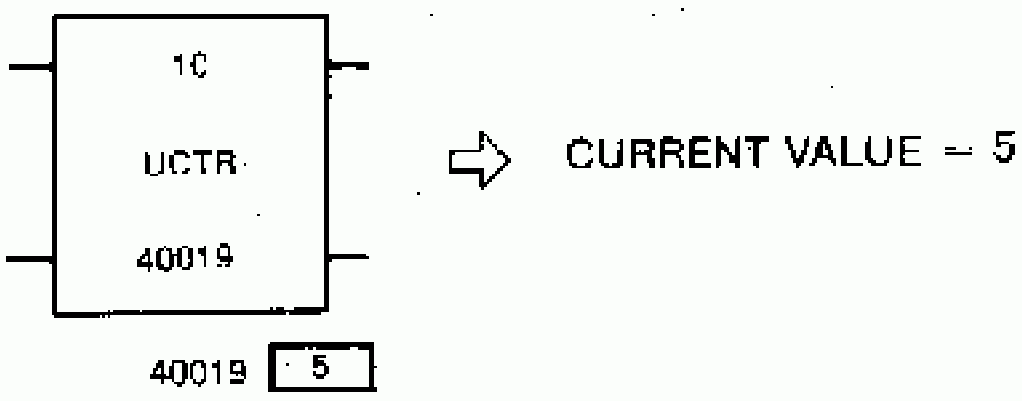
Specify a value of 1 to 9999 to determine the upper limit (variable) of count.



Example

(3) Current Value

This is the value a counter has counted.

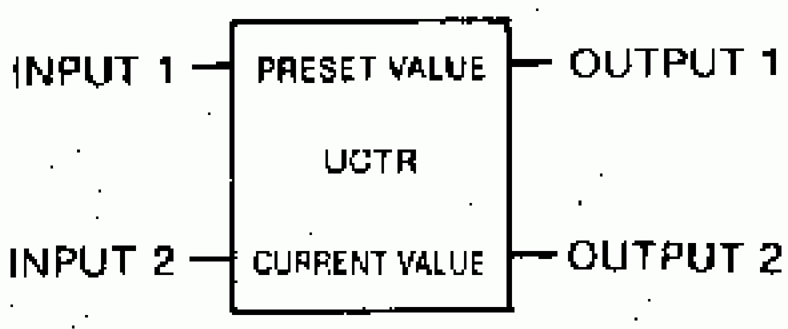


Example

4.4.3 Function and Operation of Counter

4.4.3.1 Up Counter

(1) Up Counter Function



When input 2 is ON, the up counter counts the number of times in which input 1 is turned from off to on. The current value is increased by one every counting.

When the current value becomes equal to the preset value, the up counter stops counting with output 1 turned on and output 2 off.

When input 2 is OFF, the up counter does not count even if input 1 is changed from OFF to ON. At this time, the current value is 0, output 1 is OFF and output 2 ON.

(2) Up Counter Operation

Fig. 4.11 and Table 4.9 show the timing chart and operation of the up counter.

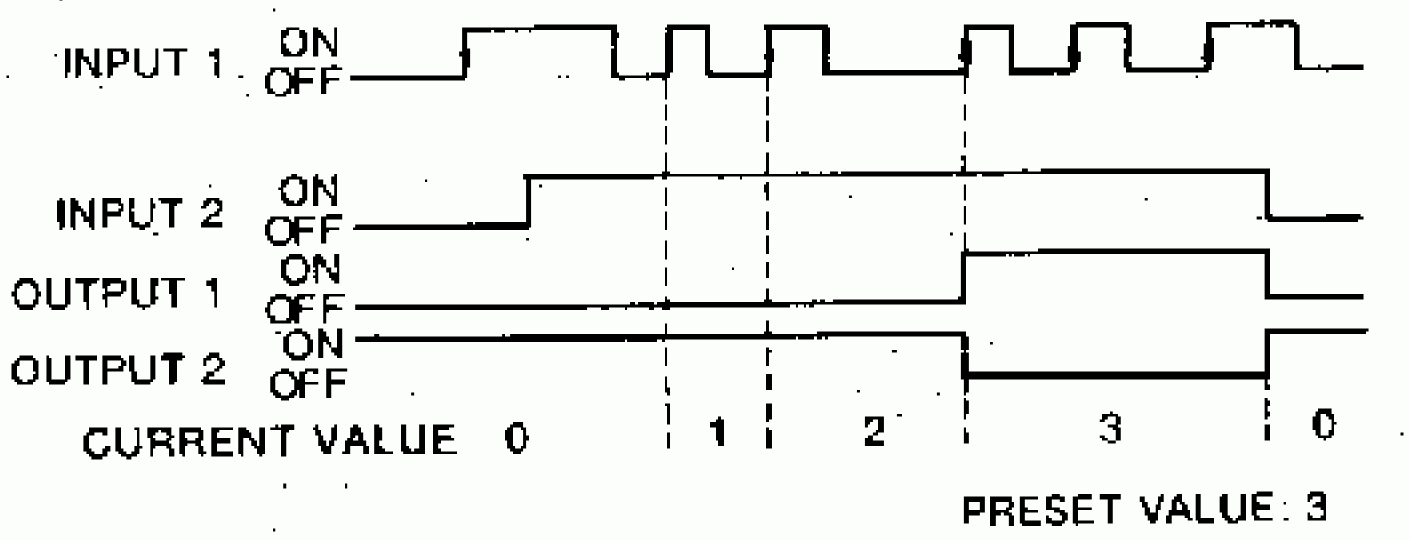


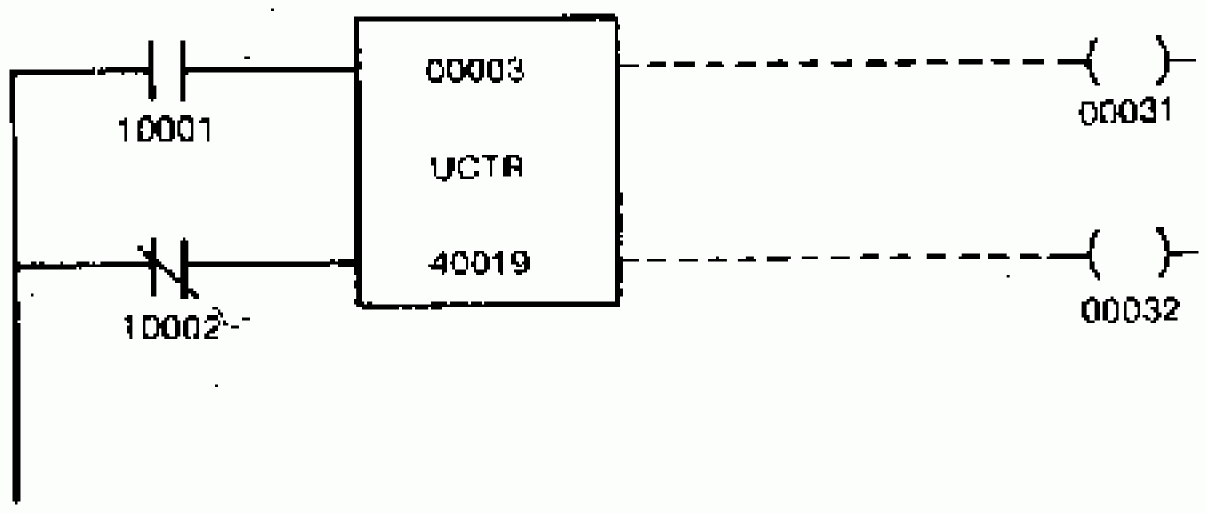
Fig. 4.11 Up Counter Operation

Table 4.9 Up Counter Operation

Input Status		Counter Status		Current Value	Output Status	
Input 1	Input 2				Output 1	Output 2
• ON Status • OFF Status • OFF→ON • ON→OFF	OFF	Reset status		0	OFF	ON
OFF→ON	ON	Operating status	Current value < Preset value	Increase (+1)	OFF*	ON*
			Current value = Preset value	Preset value	ON	OFF
ON→OFF	ON	Standstill status	Current value < Preset value	Constant	OFF	ON
			Current value = Preset value	Preset value	ON	OFF

*As a result of increasing current value, current value < preset value.

(3) Sample Up Counter

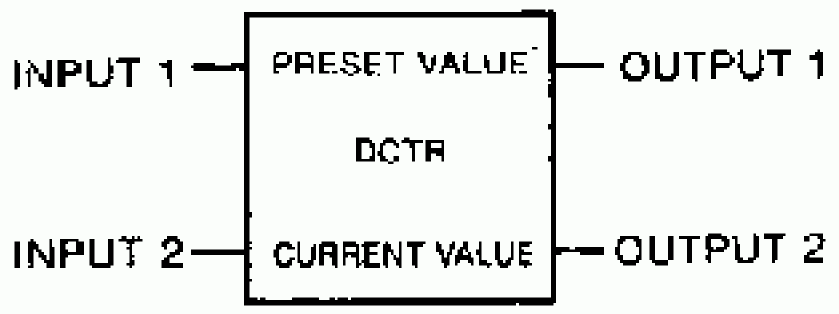


NET #20

- The up counter counts only when input 2 is ON (input relay 10002 is OFF).
- When input 1 (input relay 10001) is turned from off to on while input 2 is ON, the up counter counts and increases the current value (contents of 40019) by one.
- When the current value becomes equal to the preset value (3), the up counter stops counting and the output 1 (coil 00031) is turned on and output 2 (coil 00032) off.
- When input 2 is OFF (input relay 10002 is ON), the up counter does not count even if input 1 is changed from OFF to ON. At this time, the current value is 0, output 1 OFF, and output 2 ON.

4.4.3.2 Down Counter

(1) Down Counter Function



- When input 2 is ON, the down counter counts the number of times in which input 1 is turned form off to on. The current value is decreased by one every counting.
- When the current value becomes zero, the down counter stops counting with output 1 turned on and output 2 off.
- When input 2 is OFF, the down counter does not count even if input 1 is changed from OFF to ON. At this time, the current value becomes equal to the preset value with output 1 turned off and output 2 on.

(2) Down Counter Operation

Fig. 4.12 and Table 4.10 show the timing chart and operation of the down counter.

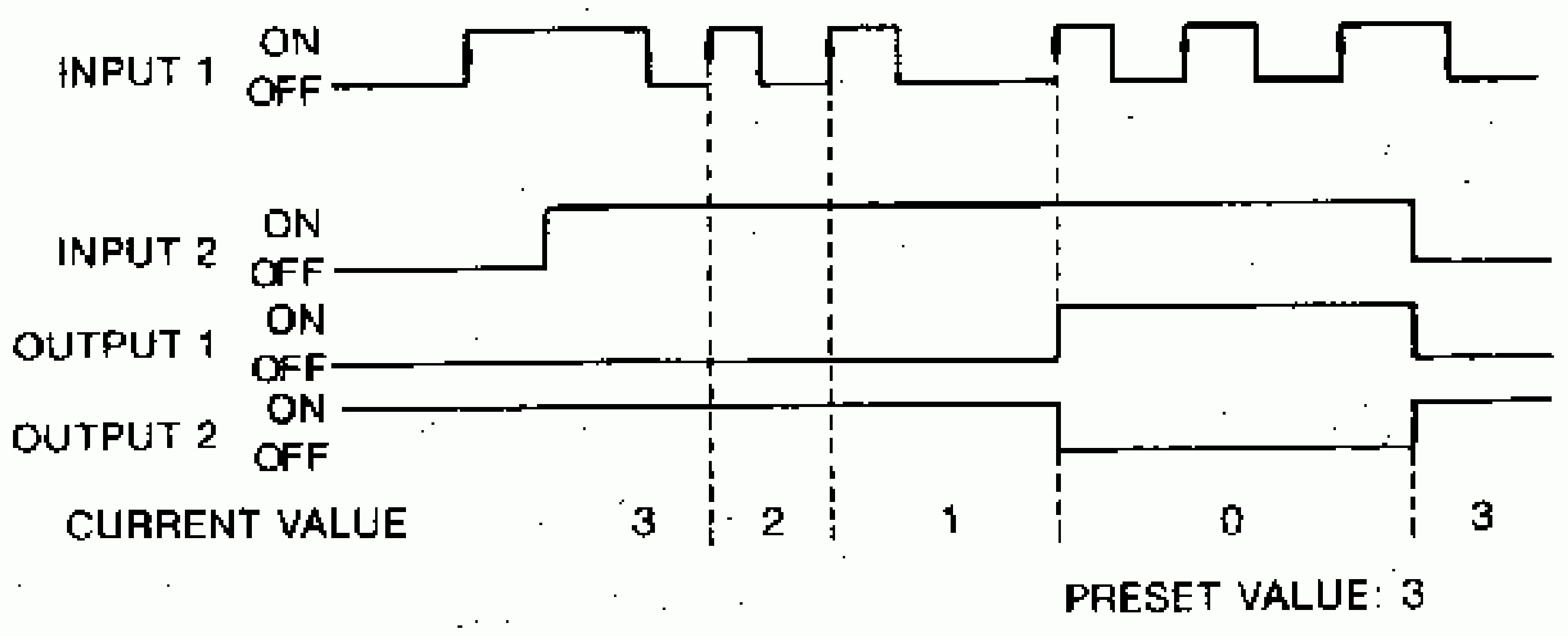


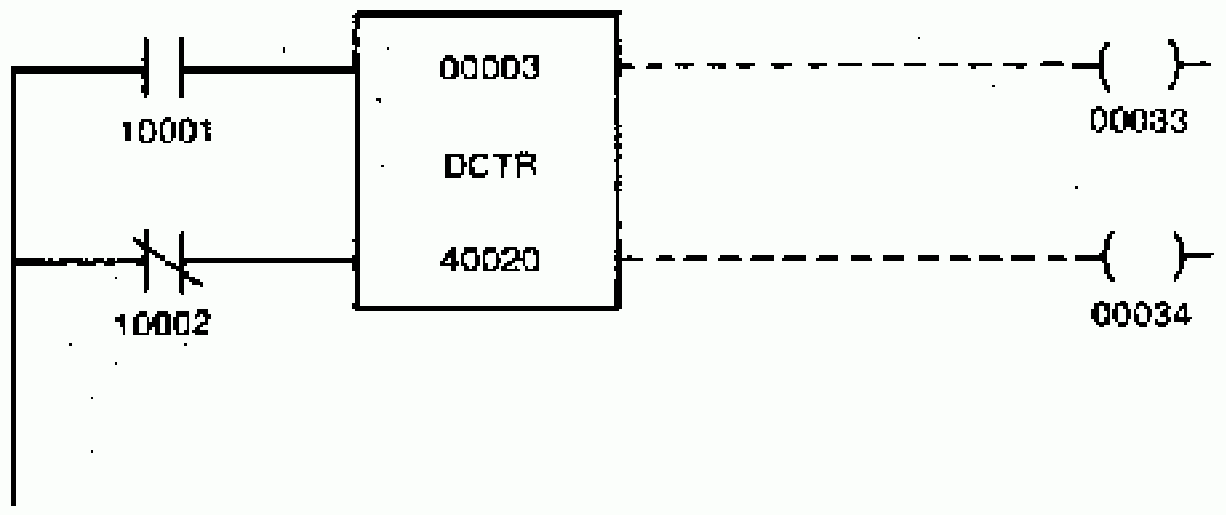
Fig. 4.12 Down Counter Operation

Table 4.10 Down Counter Operation

Input Status		Counter Status		Current Value	Output Status	
Input 1	Input 2				Output 1	Output 2
• OFF Status • ON Status • OFF→ON • ON→OFF	OFF	Reset status		Preset value	OFF	ON
OFF→ON	ON	Operating status	Current value > 0	Decrease (1)	OFF*	ON*
			Current value = 0	0	ON	OFF
• OFF Status • ON Status • ON→OFF	ON	Standstill status	Current value > 0	Constant	OFF	ON
			Current value = 0	0	ON	OFF

*As a result of decreasing current value, current value > 0

(3) Sample Down Counter



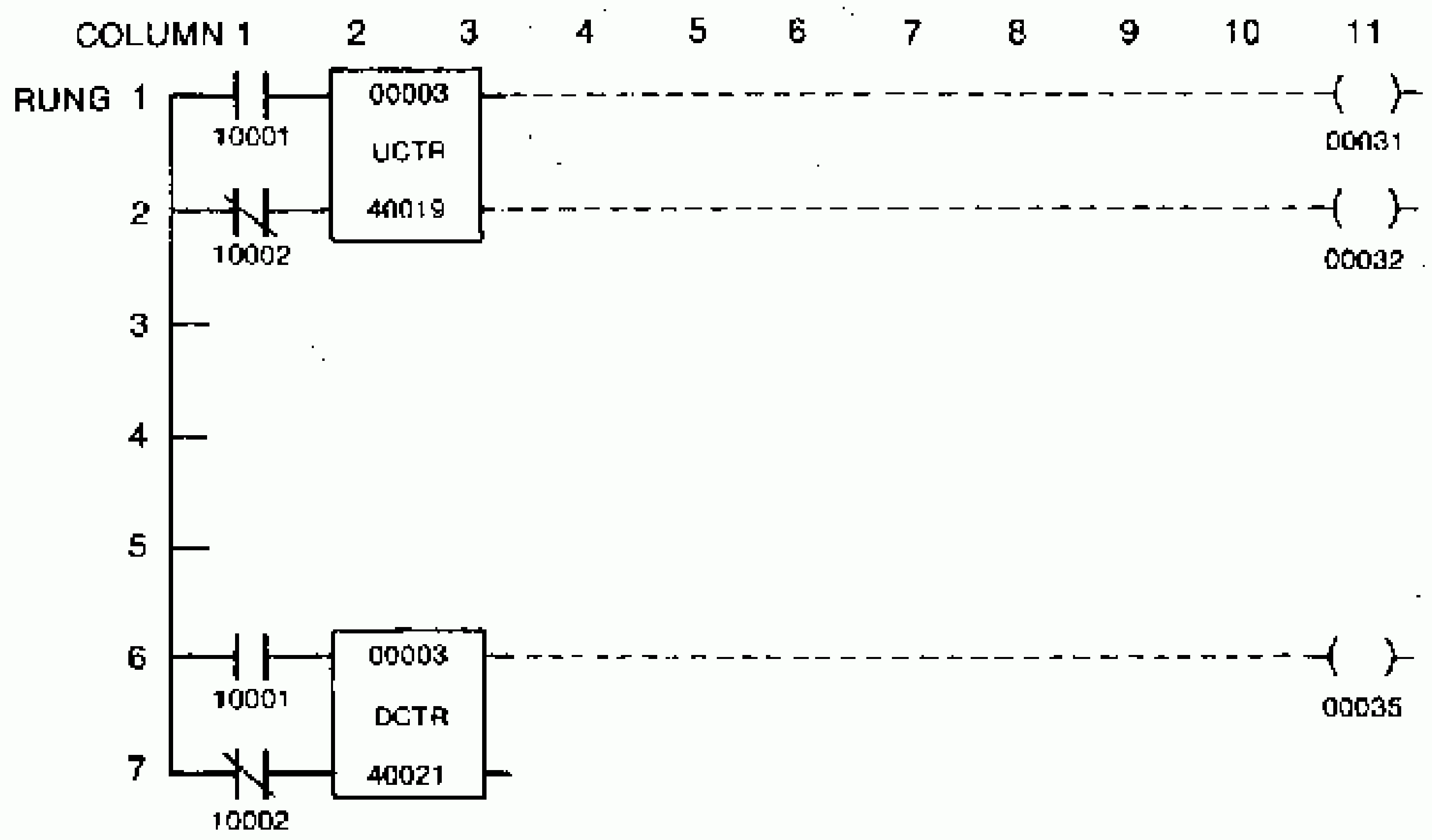
NET #21

- The down counter counts only when input 2 is ON (input relay 10002 is OFF).
- When input 1 (input relay 10001) is turned from off to on while input 2 is ON, the down counter decreases the current value (contents of 40020) by one.
- When the current value becomes zero, the down counter stops to counting and output 1 (coil 00033) is turned on and output 2 (coil 00034) off.
- When input 2 is OFF (input relay 10002 is ON), the down counter does not count even if input 1 is changed from OFF to ON. At this time, the current value becomes equal to the preset value and output 1 is turned off and output 2 on.

4. 4. 4 Programming Counter Circuit and Precautions

(1) Programming Counter Circuit

An up and down counter occupies two elements placed vertically (top and bottom) in a network. It can be uaed at any intersection of the 7 lines-by-10 columns matrix, but the top element (preset value) cannot be located on line 7.

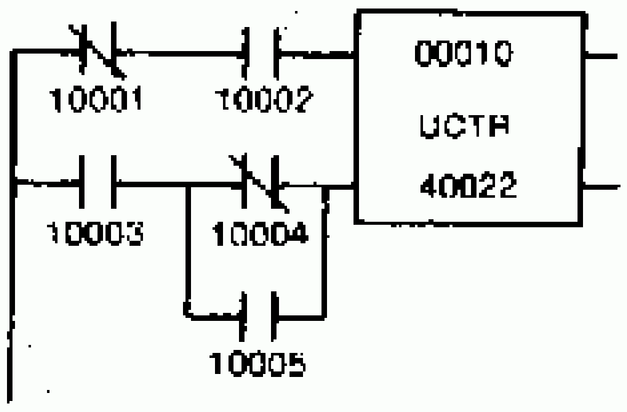


NET # 20

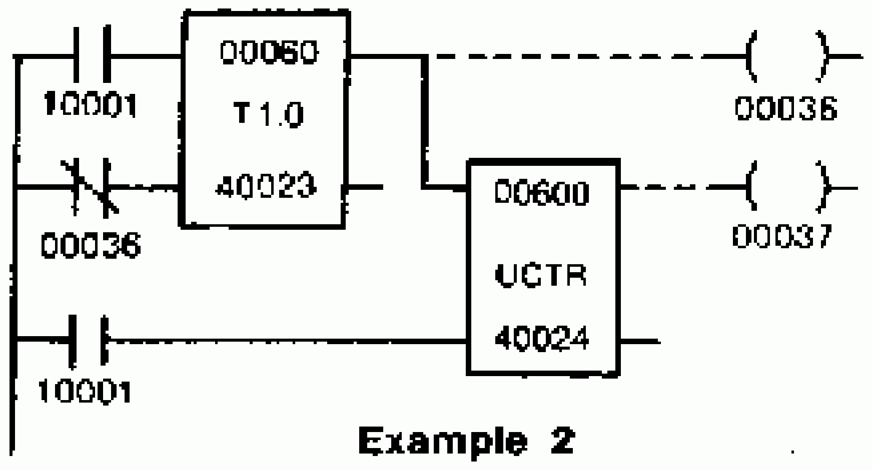
Example

(2) Counter Inputs

Inputs to the counter may be outputs of relays, other counters, timers, arithmetic operations, and data processing circuits.



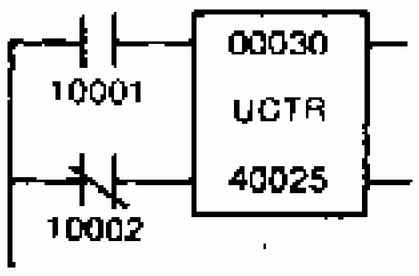
Example 1



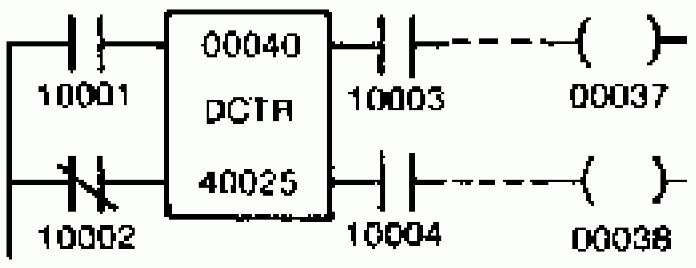
Example 2

(3) Counter Outputs

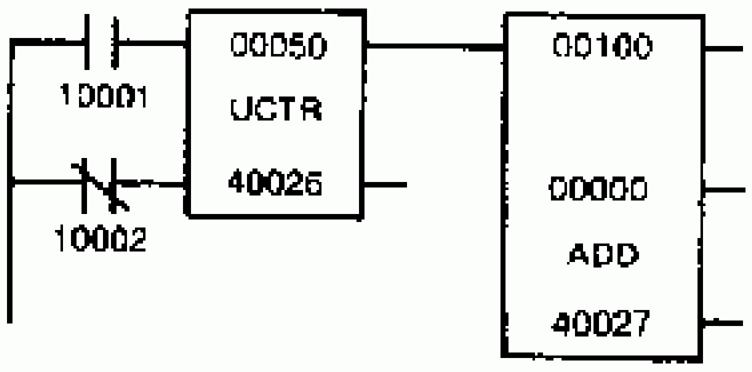
Coils may not be connected to outputs 1 and 2 of a counter. It is permitted to insert a realy contact to the right of an output or to connect an output directly to an input of a logic, except relays.



Example 1



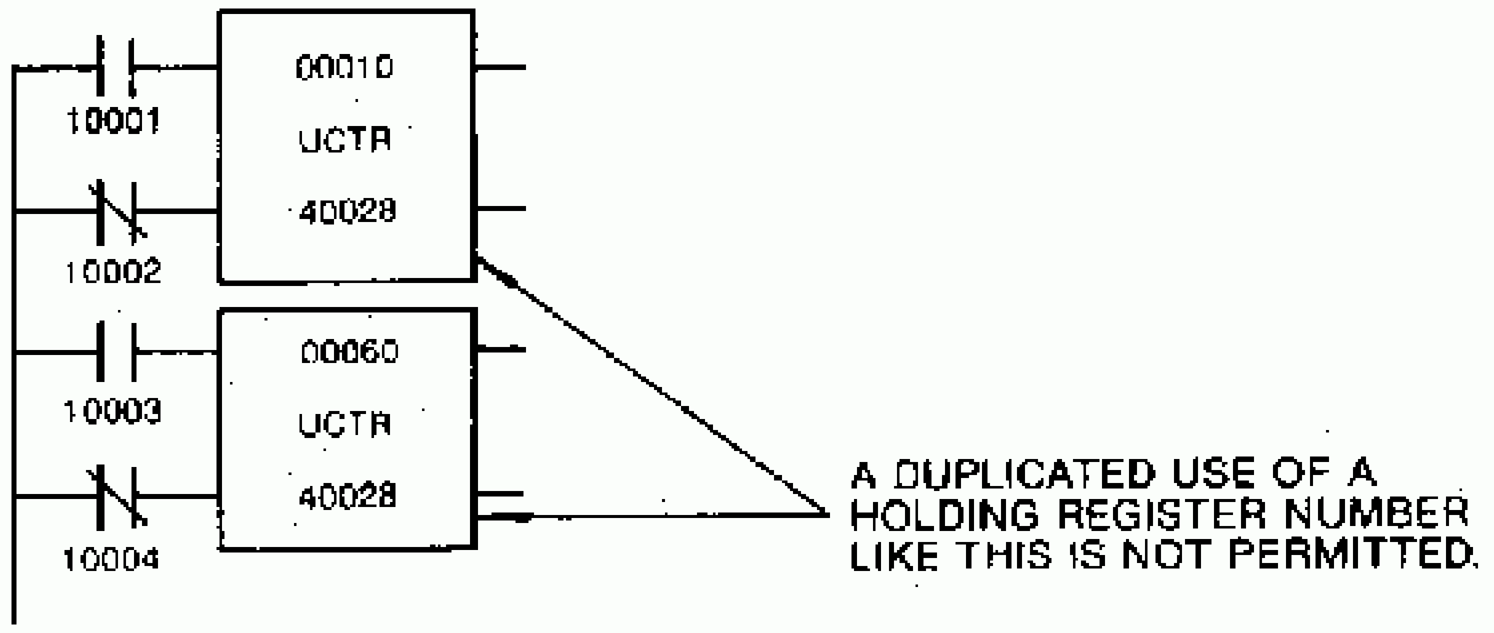
Example 2



Example 3

(4) Storing Counter Current Value

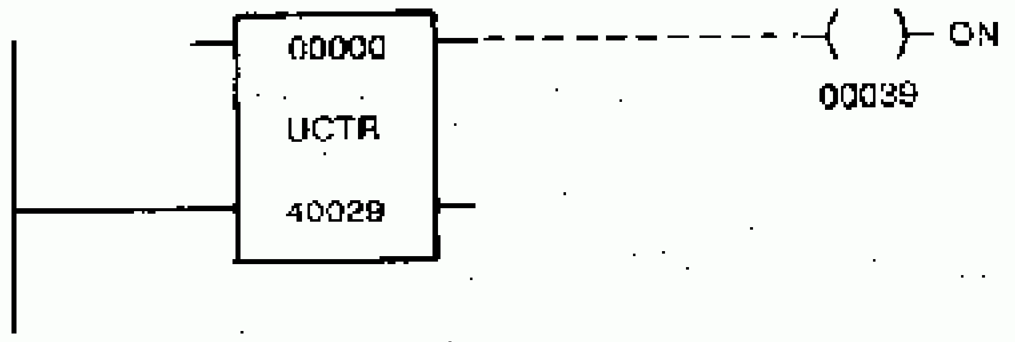
The register number of the holding register storing the current value of a counter cannot be the same as the register number of the holding register storing the current value of another counter or timer. In special cases when used as an up/down counter (see (5) of Application Circuits of Counter), a register number may be used for different counters or timers.



Example

(5) Preset Value of Counter

When the preset value of an up- or down-counter is 0 and input 2 is ON, output 1 is ON (output 2 is OFF), regardless of on/off operation of input 1.



Example

(6) Relationship of Preset and Current Values

Ordinarily, the current value does not exceed the preset value, but it is possible to make the current value greater than the preset value by arithmetic operation or data transfer function. In such a case, the current value becomes equal to the preset value as soon as the counter circuit is solved.

(7) Input Pulse Width

When operating a counter by signals (count pulses) fed from an external device, one scan time or more is required for each pulse width (ON or OFF). In addition, delay of response of the input module which receives the input pulses must be taken into consideration.

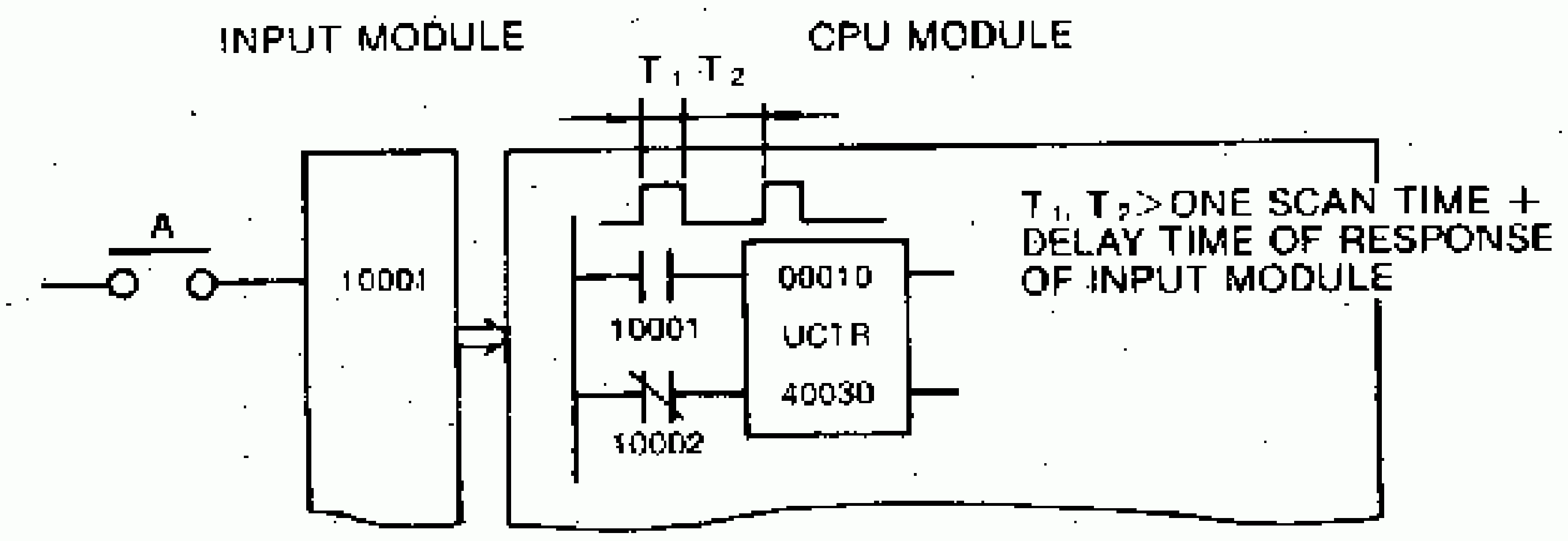


Fig. 4.13 Counter Operation by Signals fed from an External Device

(8) Current Value of Counter at Power ON

Like the timer, the counter keeps the current value even if power fails. When the U84 is turned on, the current value is determined by signals for input 2.

Table 4.11 Current Value of Counter at Power ON

Signal to Input 2				Current Value at Power ON	
Type of Reference Coil		Status	Contact	Up Counter	Down Counter
Input Relay Latch Coil		ON	NO	Value at power failure	Value at power failure
			NC	0	Preset value
		OFF	NO	0	Preset value
			NC	Value at power failure	Value at power failure
Coil (Not including Latch Coil)	n ≦ m	ON	NO	Value at power failure	Value at power failure
			NC	0	Preset value
		OFF	NO	0	Preset value
			NC	Value at power failure	Value at power failure
	n > m	OFF	NO	0	Preset value
			NC	Value at power failure	Value at power failure

Note:
1. The number of the network including the reference coil of signal to input 2 is n and the number of the network including the counter circuit is m.
2. If the signal to input 2 is composed of more than one contact, obtain the input 2 status from each contact status.
3. The current value of the counter, at the U84 power ON, is the value read when the counter circuit has been solved during the first scanning cycle after power is supplied to the U84.

4.4.5 Application Counter Circuits

(1) Large-Capacity Counter

- A large-capacity counter (having a preset value of 10000 or more) can be created by combining multiple counters.
- Fig. 4.14 shows a large-capacity counter (preset value = 100 × 150 = 15000) using two up counters.

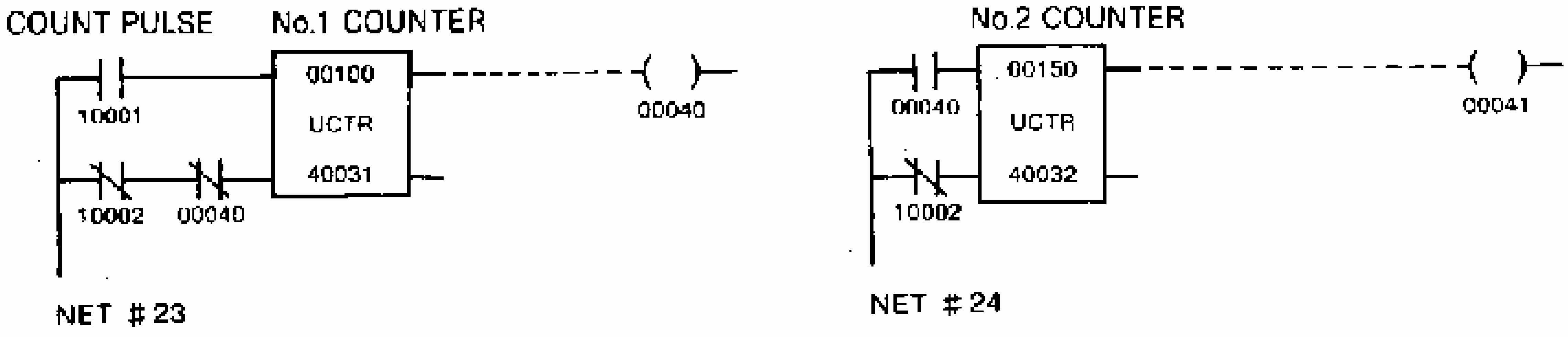
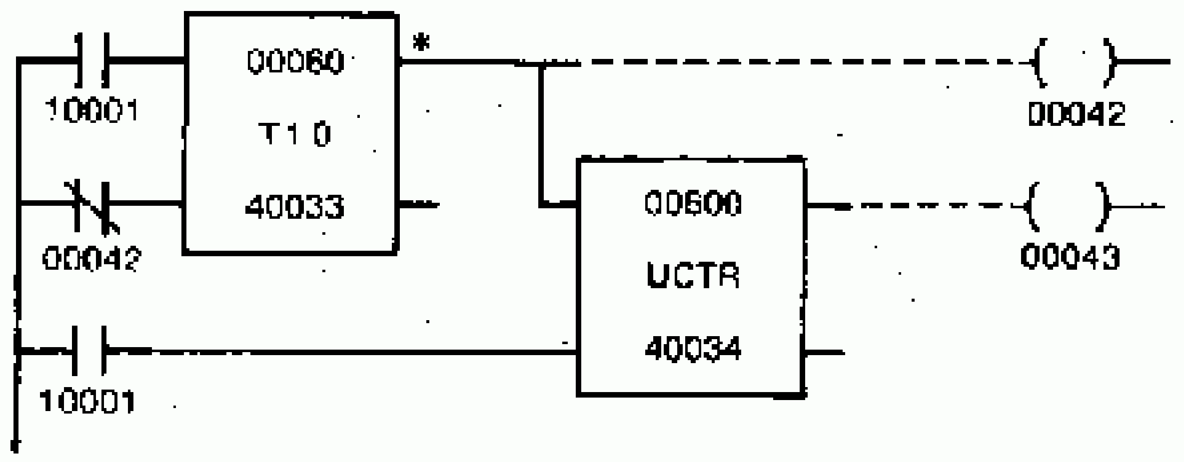


Fig. 4.14 Large-capacity Counter

- The No. 1 counter generates a pulse (ON time = 1 scan time) for the coil 00040 for every 100 input pulses.
- When the No. 2 counter counts 150 pulses output by the No. 1 counter, coil 00041 is turned on to indicate that 15000 pulses (100×150) have been counted.
- If the contents of 40031 is x (0 ≦ x ≦ 100) and the contents of 40032 is y (0 ≦ y ≦ 150), the current value (pulse count) of the large-capacity counter is x + 100y.

(2) Long-time Timer

- A long-time timer (having a preset value of 10000 or more) can be created by combining a timer and an up counter.
- Fig. 4.15 shows a long-time timer (preset value = 60 × 600 = 36000) using a 1-second timer and an up counter.



NET # 25

* Horizontal shunt is required.

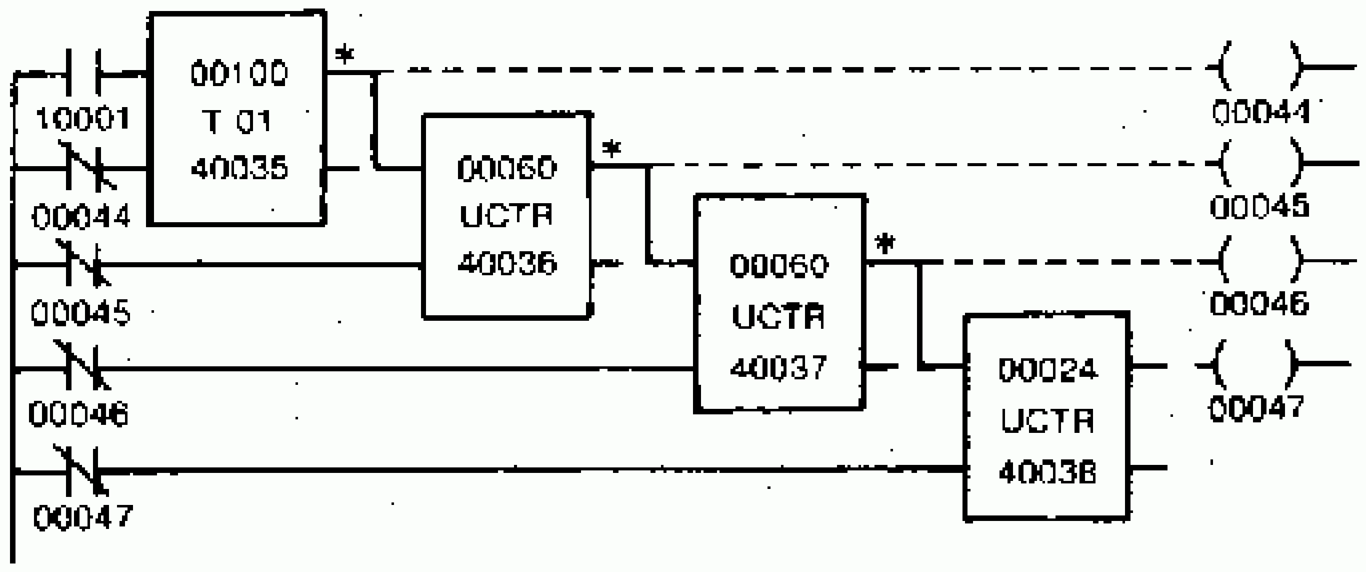
Fig. 4.15 Long-time Timer

- The 1-second timer generates a pulse (ON time = 1 scan time) for coil 00042 for every 60 seconds while input relay 10001 is ON.
- When the up counter counts 600 pulses output by the 1-second timer, coil 00043 is turned on to indicate that 36000 seconds (60×600) have elapsed.
- If the contents of 40033 is x (0 ≦ x ≦ 60) and the contents of 40034 is y (0 ≦ y ≦ 600), the current value of the long-timer is x + 60y (the number of seconds the timer has operated).

(3) Clock

- A clock circuit can be created by combining a timer with an up counter.
- Fig. 4.16 shows an example of a clock circuit.

START



NET # 26

* Horizontal shunts are required.

Fig. 4.16 Sample Timer/Counter Cascaded Logic

; while input relay 10001 is ON. The contents of 40036, 40037, e values in seconds, minutes, and hours, respectively. The ıded with a compensating circuit.

(4) Circuit for Measuring Scan Time

- It is possible to make up a circuit for measuring the scanning time of the U84 by combining an up counter with a 0.1-second timer.
- Fig. 4.17 shows a sample measuring circuit for scan time.

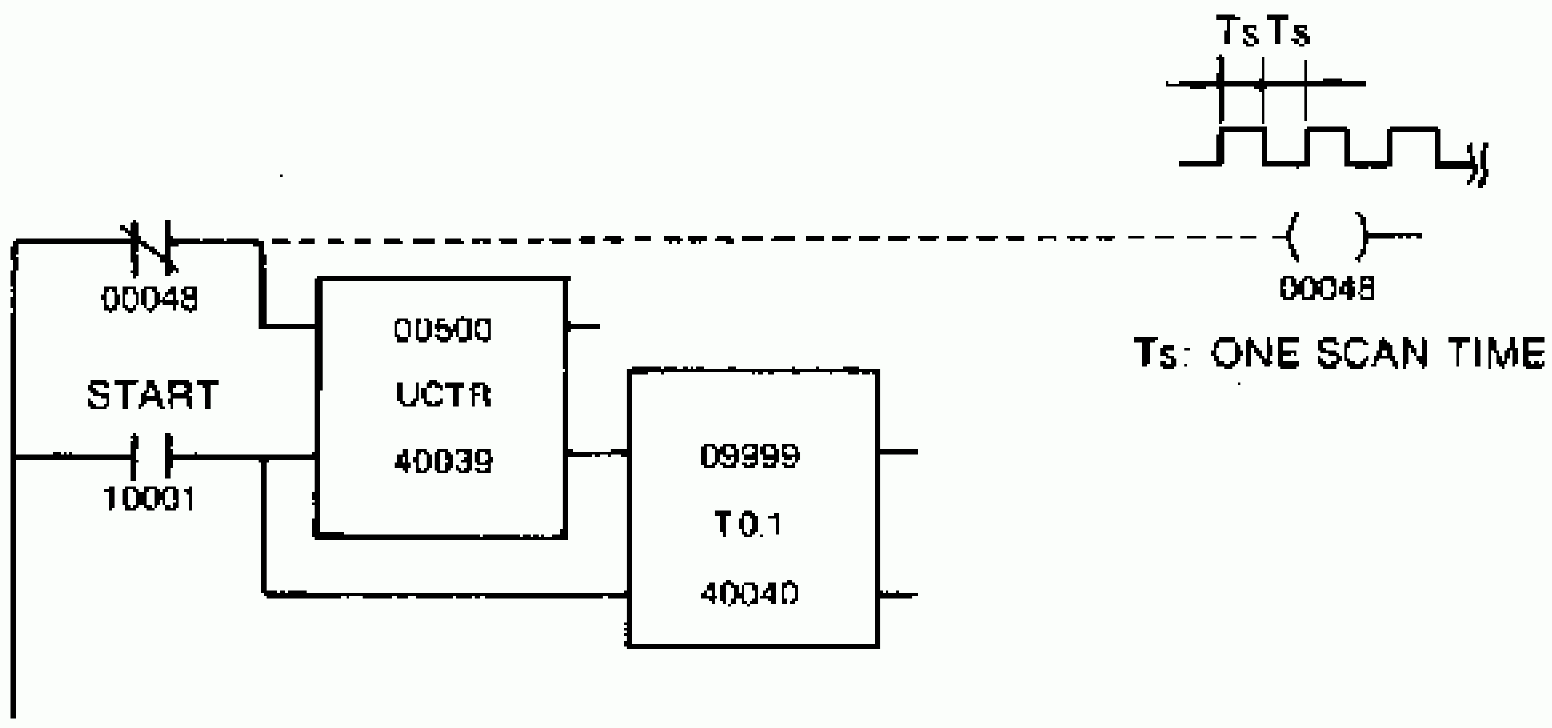


Fig. 4.17 A Circuit for Measuring Scan Time

- When input relay 10001 is turned on, both the up counter and the 0.1-second timer start to measure the U84's scanning time.
- Coil 00048 is turned on and off alternately every scanning cycle. When the up counter counts 500 pulses (alternations of ON and OFF), the output 2 is turned off and the 0.1-second timer stops.
- Now the timer's current value (contents of 40040) is the time taken to count 500 pulses or, the time of 1000 scanning cycles given in units of 0.1 second.
- If, for example, the value is 205, then the average value of one scan time is $205 \times 0.1 \div 1000 = 20.5$ milli-seconds.

(5) Up/Down Counter

- An up/down counter can be made by combining an up counter with a down-counter.
- Fig. 4.18 shows an example of an up/down counter.

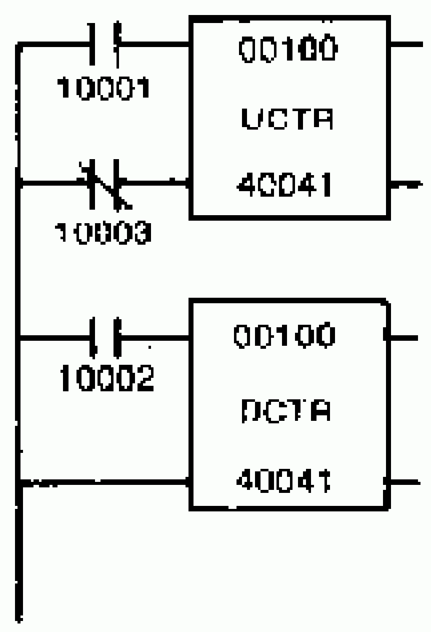


Fig. 4.18 Up/Down Counter

- The current value (contents of 40041) is increased by one each time input relay 10001 is turned from off to on.
- The current value is decreased by one each time input realy 10002 is turned from off to on.
- The current value is reset to zero when input relay 10003 is tuned on.

4. 5 ARITHMETIC FUNCTIONS

4. 5. 1 Types of Arithmetic Functions

- The arithmetic operations are addition, subtraction, multiply, and divide applied on operand V_1 by operand V_2. There are eight variations of arithmetic functions as described in Table 4.12.
- Hereafter, the following symbols are substituted for operands and the number of holding register storing the result of operation.
  - Operand: V_1 and V_2
  - Holding register number: R

  Example: $V_1 \times V_2 \rightarrow$ R(4 higher-place digits), R + 1(4 lower-place digits)

  ⬇

  This means that V_1 is multiplied by V_2 and the higher-place four digits of the result is placed in holding register R and the lower-place four digits in holding register R+1.

Table 4.12 Types of Arithmetic Functions

Type	Symbol	Operator	Range of Operand V_1	Range of Operand V_2	Reference Page
Addition	ADD	$V_1 - V_2$	0 – 9999		66
Double-precision Addition	DADD		0 – 99999999		68
Subtraction	SUB	$\|V_1 - V_2\|$	0 – 9999		70
Double-precision Subtraction	DSUB		0 – 99999999		72
Multiply	MUL	$V_1 \times V_2$	0 - 9999		74
Double-precision Multiply	DMUL		0 - 99999999		75
Divide	DIV	$V_1 : V_2$	0–9999 (0–99989999)*	0 – 9999	77
Double-precision Divide	DDIV		0 – 9999999899999999	0 – 99999999	80

*The range of V1 becomes as shown in parentheses when two successive input registers or holding registers are used.

4. 5. 2 Addition (ADD)

(1) Form

- Fig. 4.19 shows the form of addition (ADD).
- Addition operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.13, specify any of constant K, reference number 3XXXX of an input register, and reference number 4XXXX of a holding register for each of the top and middle elements.
- Only reference number 4XXXX of a holding register may be specified for the bottom element.
- ADD is the symbol denoting the addition.

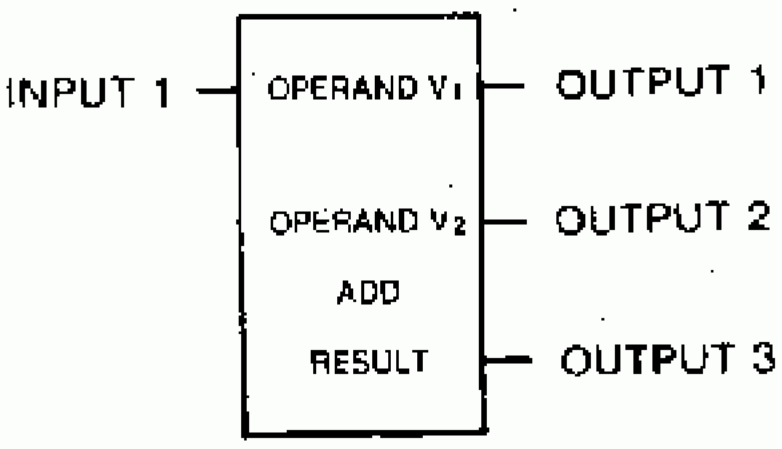
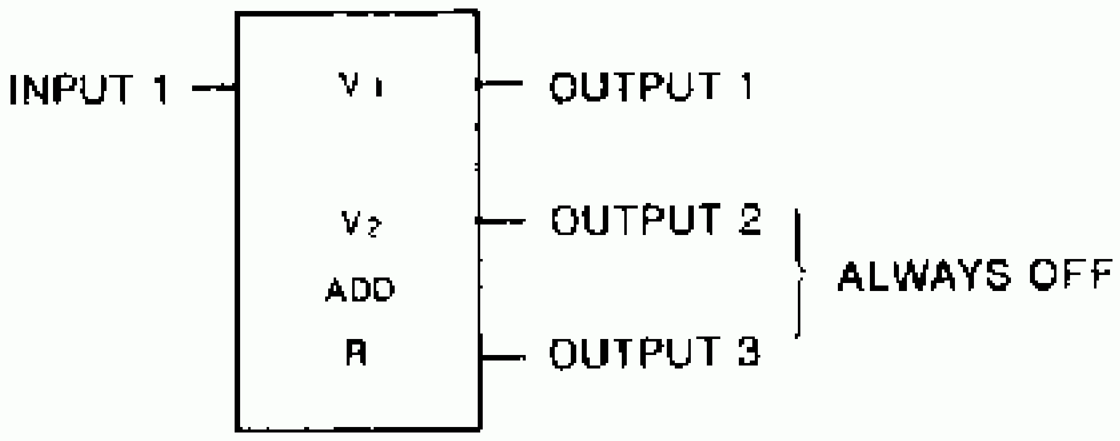


Fig. 4.19 ADD General Form

Table 4.13 Elements of Addition Function

Element Position	Specified Number	Description
Top	Any one of the following: • Constant K (00000—09999) • 3×××× (30001—30256) • 4×××× (40001—49999)	• When constant K is specified, the value is the operand (V_1 = 0 to 9999). • When 3×××× and 4×××× are specified, the contents are the operand (V_1 = 0 to 9999)
Middle		• When constant K is specified, the value is the operand (V_2 = 0 to 9999). • When 3×××× and 4××××are specified, the contents are the operand (V_2 = 0 to 9999).
Bottom	4×××× (40001—49999)	The result of addition function ($V_1 + V_2$ = 0 to 9999) is stored in 4××××.

(2) Addition Function



- By the addition (ADD), $V_1 + V_2$ is calculated when the input 1 is ON. The result is treated as follows.
  - (a) If $V_1 + V_2 \leqq 9999$,
    $V_1 + V_2$ is stored in R. The Output 1 remains OFF.
  - (b) If $V_1 + V_2 \geqq 10000$, $V_1 + V_2 - 10000$ is stored in R. The output 1 is turned on.
- The outputs 2 and 3 are always OFF.
- The result remains in R even after the input 1 is turned from on to off.

(3) Addition Operation

Table 4.14 shows an addition operation (ADD).

Table 4.14 Addition Operation

Input 1	Condition	Operation	Output 1
ON	$V_1 + V_2 \leqq 9999$	$V_1 + V_2 \rightarrow R$	OFF
	$V_1 + V_2 \geqq 10000$	$V_1 + V_2 - 10000 \rightarrow R$	ON
OFF	None	Not operated.	OFF

(4) Example—Addition

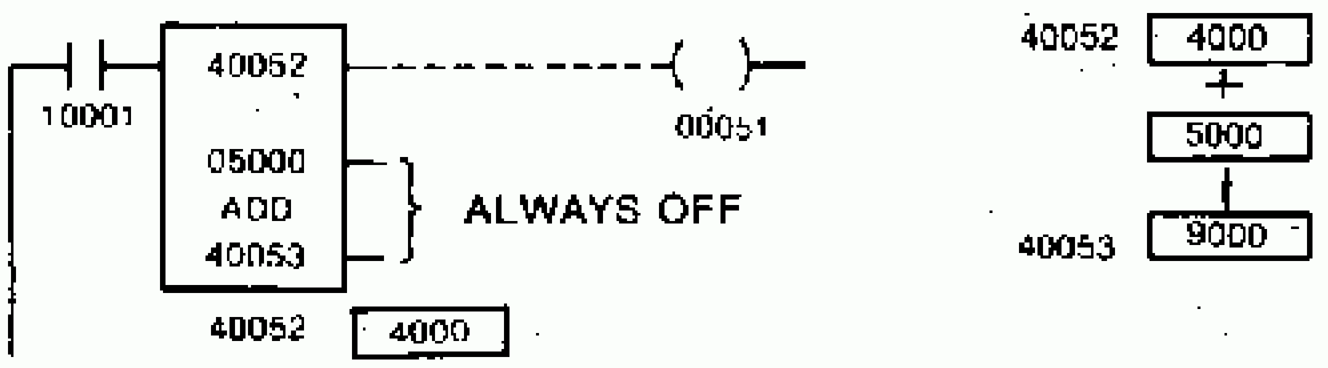
(a) NET #32 (b) ADD Operation

Example 1

ADD in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 remains OFF. The result remains in 40051 even after input relay 10001 is turned off.

4. 5. 2 Addition (Cont'd)

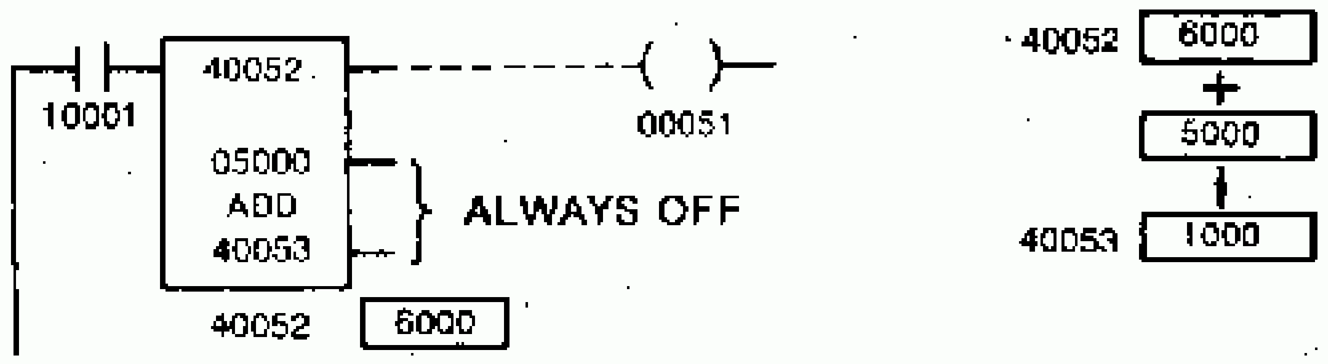
Example 2



(a) NET #31 (b) ADD Operation

ADD in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 remains OFF. The result remains in 40053 even after input relay 10001 is turned off.

Example 3



(a) NET #32 (b) ADD Operation

ADD in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 is turned on. The result remains in 40053 even after input relay 10001 is turned off.

4. 5. 3 Double-precision Addition (DADD)

(1) Form

- Fig. 4.20 shows the form of double-precision addition (DADD).

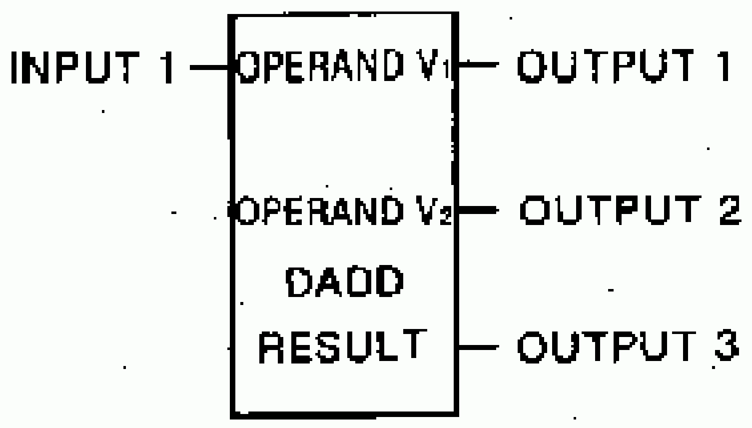


Fig. 4.20 DADD General Form

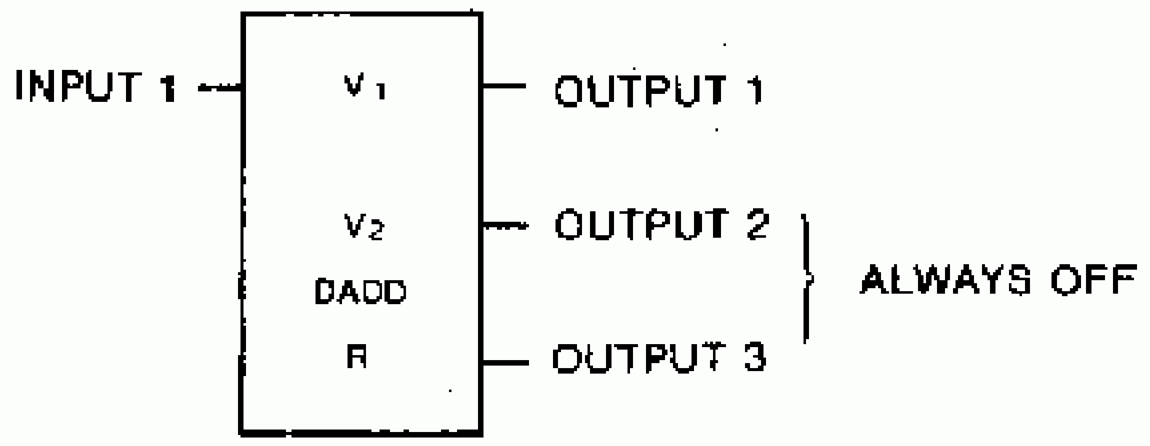
- Double-precision addition operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.15, specify either reference number 3XXXX of an input register or reference number 4XXXX of a holding register for each of the top and middle elements.
- Only reference number 4XXXX of a holding register may be specified for the bottom element.
- DADD is the symbol denoting the double-precision addition.

Table 4.15 Elements of Double-precision Addition Function

Element Position	Specified Number	Description
Top	Either one of the following: • 3×××× (30001−30255) • 4×××× (40001−49998)	Operand (V_1 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× [V_1H] 4××××+1 [V_1L] V_1H: High order 4 digits of V_1; V_1L: Low order 4 digits of V_1
Middle		Operand (V_2 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× [V_2H] 4××××+1 [V_2L] V_2H: High order 4 digits of V_2; V_2L: Low order 4 digits of V_2
Bottom	4×××× (40001−49998)	Result of operation ($V_1 + V_2$ = 0 to 99999999) is stored as follows. Example, 4×××× [$(V_1+V_2)H$] 4××××+1 [$(V_1+V_2)L$] $(V_1+V_2)H$: High order 4 digits of (V_1+V_2); $(V_1+V_2)L$: Low order 4 digits of (V_1+V_2)

(2) Double-precision Addition Function

- By the double-precision addition (DADD), $V_1 + V_2$ is calculated when the input 1 is ON. The result is treated as follows.



(a) If $V_1 + V_2 \leq 99999999$,

The four high order digits of $V_1 + V_2$ are stored in R and the four low order digits in R+1. The output 1 remains OFF.

(b) If $V_1 + V_2 \geq 100000000$,

The four high order digits of $V_1 + V_2 - 100000000$ are stored in R and the four low order digits in R+1. The output 1 is turned on.

- The outputs 2 and 3 are always OFF.
- The result remains in R and R+1 even after the input 1 is turned from on to off.

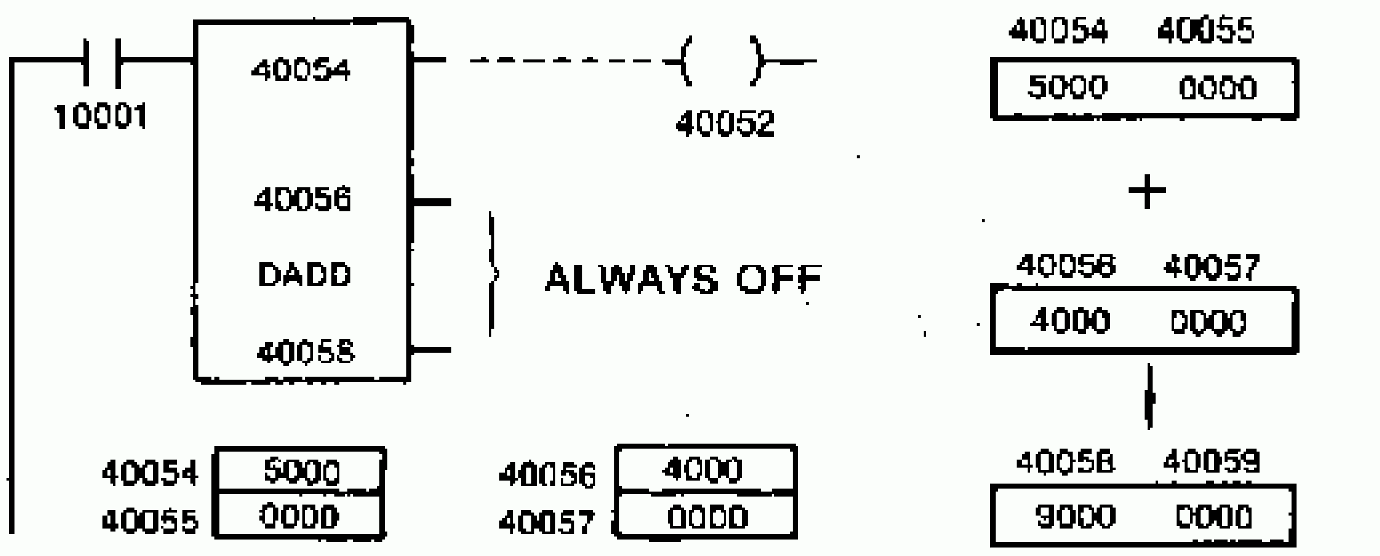
(3) Double-precision Addition Operation

Table 4.16 shows a double-precision addition operation (DADD).

Table 4.16 Double-precision Addition Operation

Input 1	Condition	Operation	Output 1
ON	$V_1 + V_2 \leq 99999999$	$V_1 + V_2 \rightarrow$ R (High order 4 digits), R+1 (Low order 4 digits).	OFF
ON	$V_1 + V_2 \geq 100000000$	$V_1 + V_2 - 100000000 \rightarrow$ R (High order 4 digits), R+1 (Low order 4 digits).	ON
OFF	None	Not operated.	OFF

(4) Example—Double-precision Addition

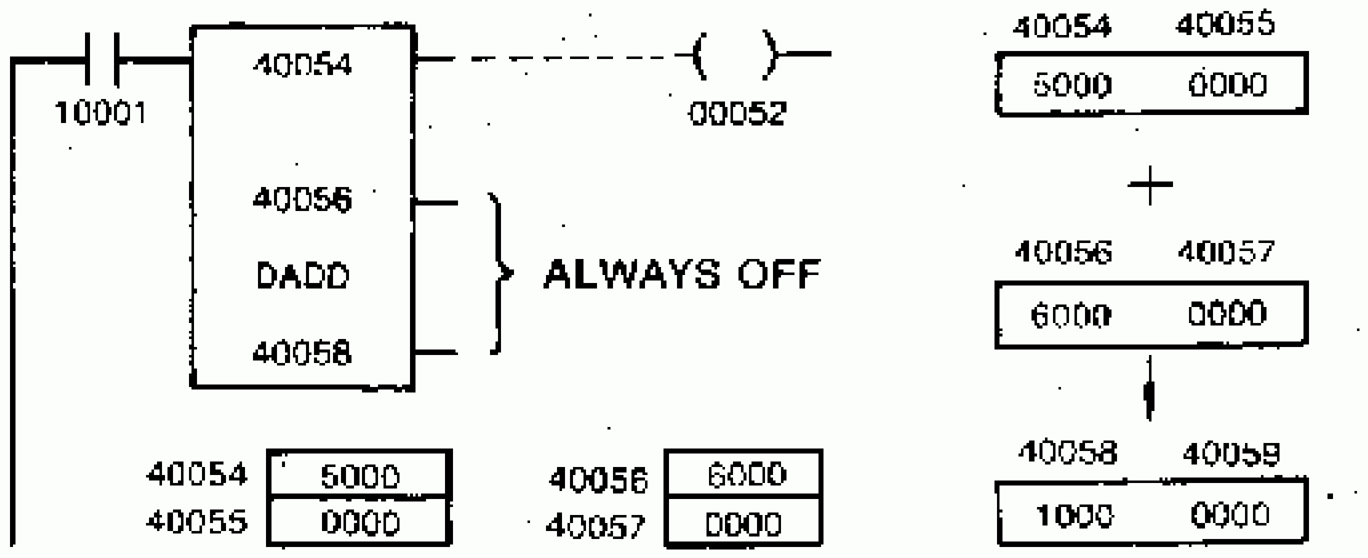


(a) NET #33 (b) DADD Operation

Example 1

DADD in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 remains OFF. The results remain in 40058 and 40059 even after input relay 10001 is turned off.

4. 5. 3 Double-precision Addition (DADD)(Cont'd)



(a) NET #33 (b) DADD Operation

DADD in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 (coil 00052) is turned on. The result remains in 40058 and 40059 even after input relay 10001 is turned off.

4. 5. 4 Subtraction (SUB)

(1) Form

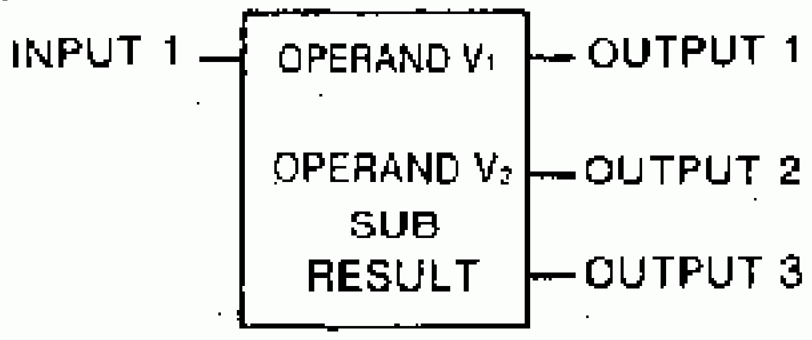


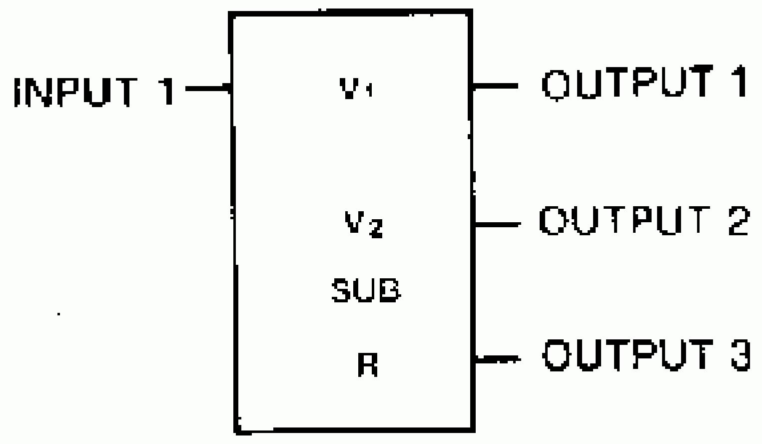
Fig. 4.21 SUB General Form

- Fig. 4.21 shows the form of subtraction (SUB).
- Subtraction operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.17, specify any of constant K, the reference number 3XXXX of an input register, and the reference number 4XXXX of a holding register for each of the top and middle elements.
- Only reference number 4XXXX of a holding register may be specified for the bottom element.
- SUB is the symbol denoting the subtraction.

Table 4.17 Elements of Subtraction Function

Element Position	Specified Number	Description
Top	Any one of the following: • Constant K (0000–9999) • 3×××× (30001–30256) • 4×××× (40001–49999)	• When constant K is specified, the value is the operand (V_1 = 0 to 9999). • When 3×××× and 4×××× are specified, the contents are the operand (V_1 = 0 to 9999).
Middle	Any one of the following: • Constant K (0000–9999) • 3×××× (30001–30256) • 4×××× (40001–49999)	• When contant K is specified, the value is the operand (V_2 = 0 to 9999). • When 3×××× and 4×××× are specified, the contents are the operand (V_2 = 0 to 9999).
Bottom	4×××× (40001–49999)	The result of subtraction function ($\lvert V_1 - V_2 \rvert$ = 0 to 9999) is stored in 4××××.

(2) Subtraction Function



- By the subtraction (SUB), $V_1 - V_2$ will be calculated when the input 1 is ON. The result is treated as follows.

  (a) If $V_1 > V_2$,

  $V_1 - V_2$ is stored in R and only the output 1 is turned on.

  (b) If $V_1 = V_2$,

  $V_1 - V_2 = 0$ is stored in R and only the output 2 is turned on.

  (c) If $V_1 < V_2$,

  $V_2 - V_1$ is stored in R and only the output 3 is turned on.

- The result remains in R even after the input 1 is turned from off to on.

(3) Subtraction Operation

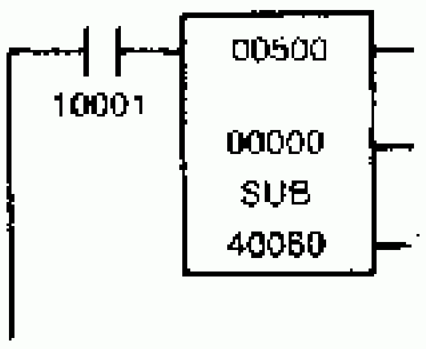
Table 4.18 shows a subtraction operation (SUB).

Table 4.18 Subtraction Operation

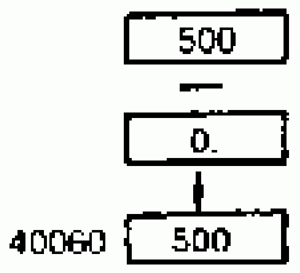
Input 1	Condition	Operation	Output 1	Output 2	Output 3
ON	$V_1 > V_2$	$V_1 - V_2 \rightarrow R$	ON	OFF	OFF
	$V_1 = V_2$	$0 \rightarrow R$	OFF	ON	OFF
	$V_1 < V_2$	$V_2 - V_1 \rightarrow R$	OFF	OFF	ON
OFF	None	Not operated.	OFF	OFF	OFF

(4) Example—Subtraction

Example 1



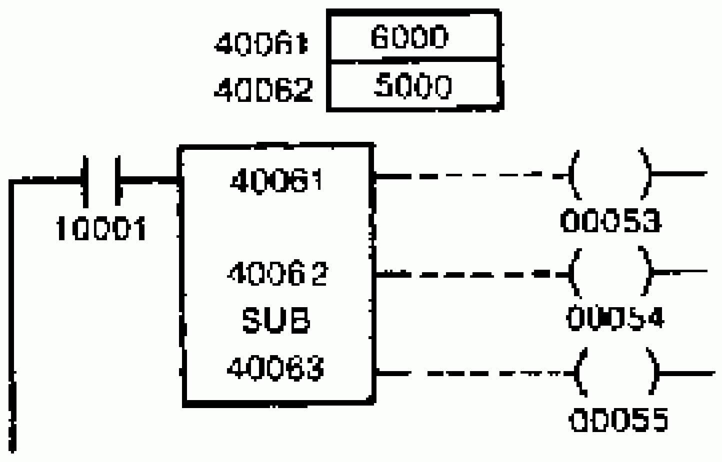
(a) NET ♯34



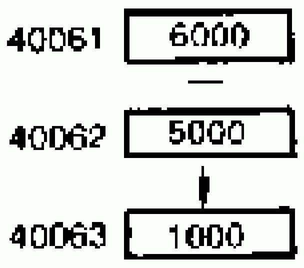
(b) SUB Operation

SUB in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 is turned on. The result remains in 40060 even after input relay 10001 is turned off.

Example 2



(a) NET ♯35

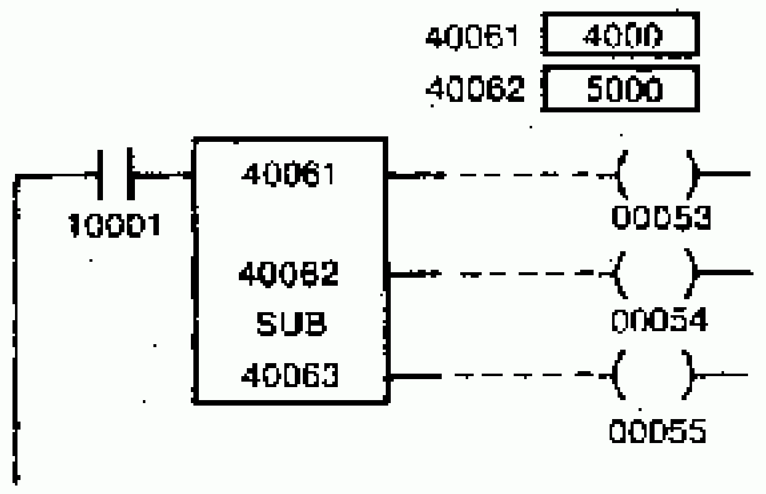


(b) SUB Operation

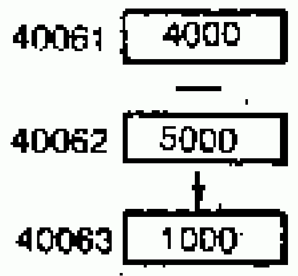
SUB in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 (coil 00053) is turned on. The result remains in 40063 even after input relay 10001 is turned off.

4.5.4 Subtraction (Cont'd)

Example 3



(a) NET #35



(b) SUB Operation

SUB in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 3 (coil 00055) is turned on. The result remains in 40063 even after input relay 10001 is turned off.

4.5.5 Double-precision Subtraction (DSUB)

(1) Form

- Fig. 4.22 shows the form of double-precision subtraction (DSUB).

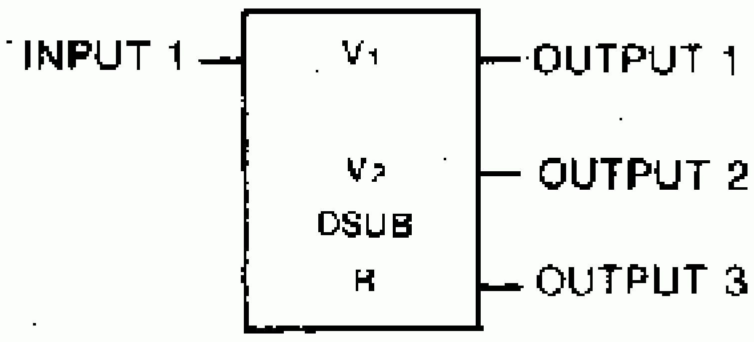


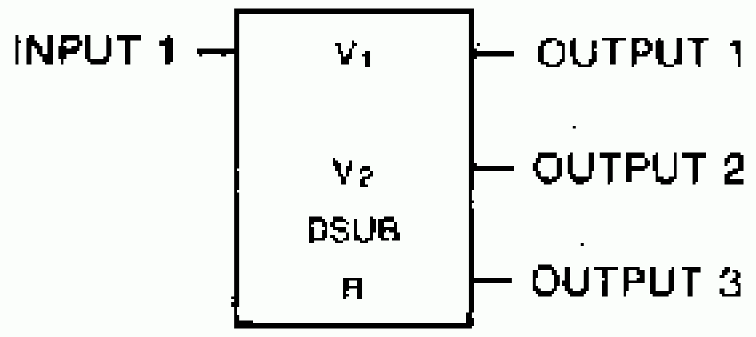
Fig. 4.22 DSUB General Form

- Double-precision subtraction operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.19, specify either reference number 3×××× of an input register or the reference number 4×××× of a holding register for each of the top and bottom elements.
- Only reference number 4×××× of a holding register may be specified for the bottom element.
- DSUB is the symbol denoting the double-precision subtraction.

Table 4.19 Elements of Double-precision Subtraction Function

Element Position	Specified Number	Description
Top	Either one of the following: • 3×××× (30001–30255) • 4×××× (40001–49998)	Operand (V_1 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× : V_1H ; 4××××+1 : V_1L V_1H: Higher-place 4 digits of V_1. V_1L: Lower-place 4 digits of V_1.
Middle		Operand (V_2 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× : V_2H ; 4××××+1 : V_2L V_2H: Higher-place 4 digits of V_2. V_2L: Lower-place 4 digits of V_2.
Bottom	4×××× (40001–49998)	Result of operation ($\lvert V_1 - V_2 \rvert$ = 0 to 99999999) is stored as follows. Example, 4×××× : $\lvert V_1 - V_2 \rvert H$; 4××××+1 : $\lvert V_1 - V_2 \rvert L$ $\lvert V_1 - V_2 \rvert H$: Higher-place 4 digits of $\lvert V_1 - V_2 \rvert H$. $\lvert V_1 - V_2 \rvert L$: Lower-place 4 digits of $\lvert V_1 - V_2 \rvert L$.

(2) Double-precision Subtraction Function



- By the double-precision subtraction (DSUB), $V_1 - V_2$ is calculated when the input 1 is ON. The result is treated as follows.

  (a) If $V_1 > V_2$,

  The four higher-place digits of $V_1 - V_2$ are stored in R and the four lower-place digits in R+1. Only the output 1 is turned on.

  (b) If $V_1 = V_2$,

  $V_1 - V_2 = 0$ is stored in R and R+1 and only the output 2 is turned on.

  (c) If $V_1 < V_2$,

  The four higher-place digits of $V_2 - V_1$ are stored in R and the four lower-place digits in R+1. Only the output 3 is turned on.

- The result remains in R and R+1 even after the input 1 is turned from on to off.

(3) Double-precision Subtraction Operation

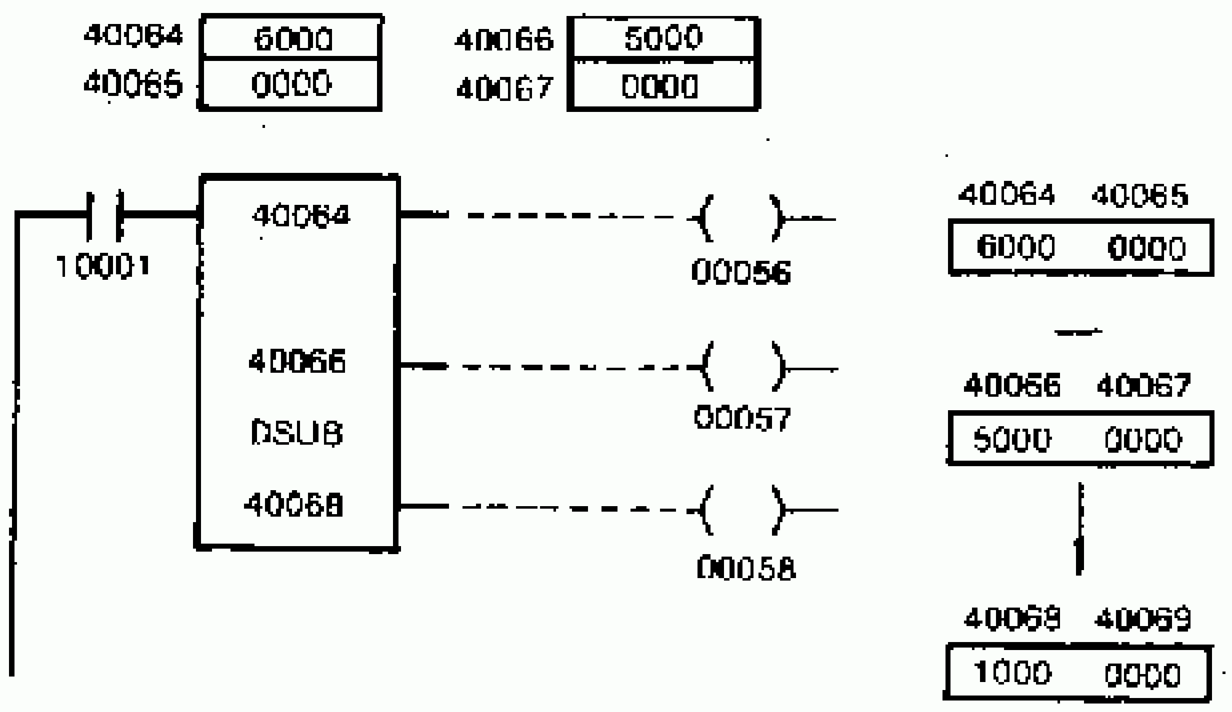
Table 4.20 shows a double-precision subtraction operation.

Table 4.20 Double-precision Subtraction Operation

Input 1	Condition	Operation	Output 1	Output 2	Output 3
ON	$V_1 > V_2$	$V_1 - V_2 \rightarrow R$ (Higher-place 4 digits), R+1 (Lower-place 4 digits)	ON	OFF	OFF
	$V_1 = V_2$	$0 \rightarrow R$, R+1	OFF	ON	OFF
	$V_1 < V_2$	$V_2 - V_1 \rightarrow R$ (Higher-place 4 digits), R+1 (Lower-place 4 digits)	OFF	OFF	ON
OFF	None	Not operated.	OFF	OFF	OFF

(4) Example — Double-precision Subtraction

Example 1

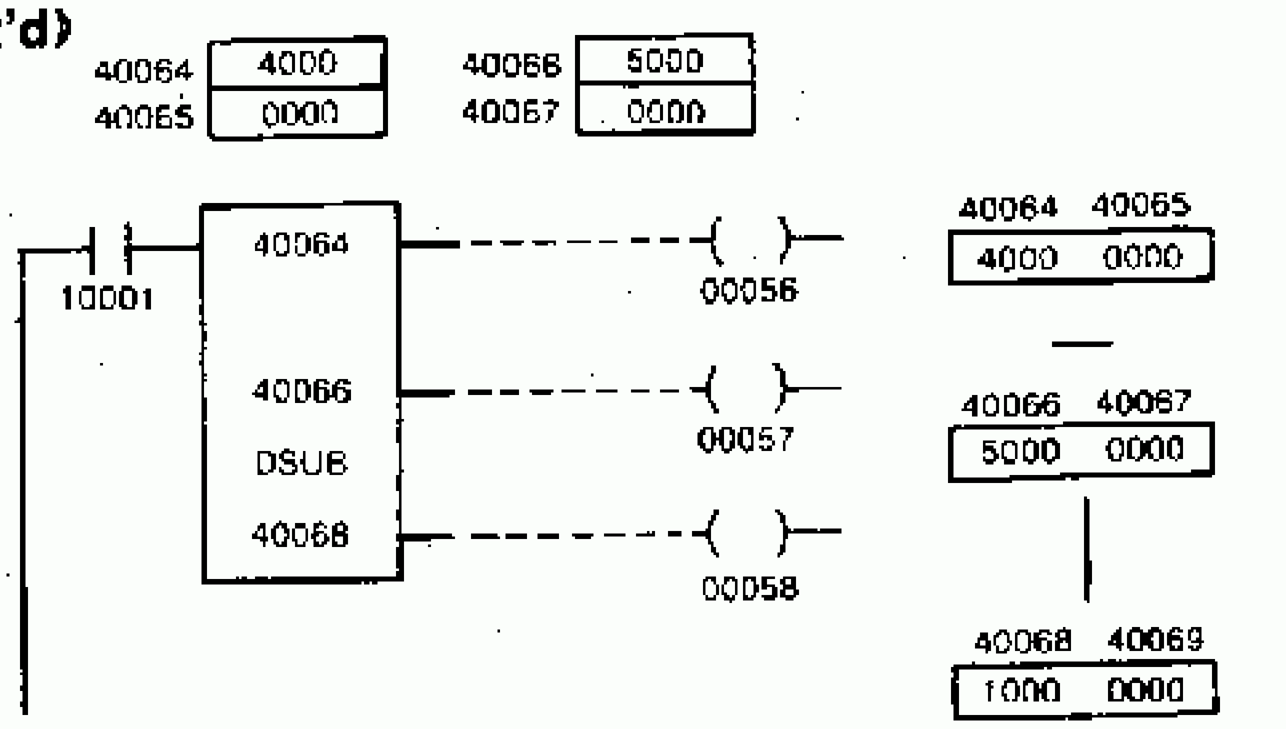


(a) NET #36 (b) DSUB Operation

DSUB in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 (coil 00056) is turned on. The result remains in 40068 and 40069 even after input relay 10001 is turned off.

4. 5. 5 Double-precision Subtraction (Cont'd)

Example 2



(a) NET #36 (b) DSUB Operation

DSUB in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 3 (coil 00058) is turned on. The result remains in 40068 and 40069 even after input relay 10001 is turned off.

4. 5. 6 Multiply (MUL)

(1) Form

- Fig. 4.23 shows the form of multiply (MUL).

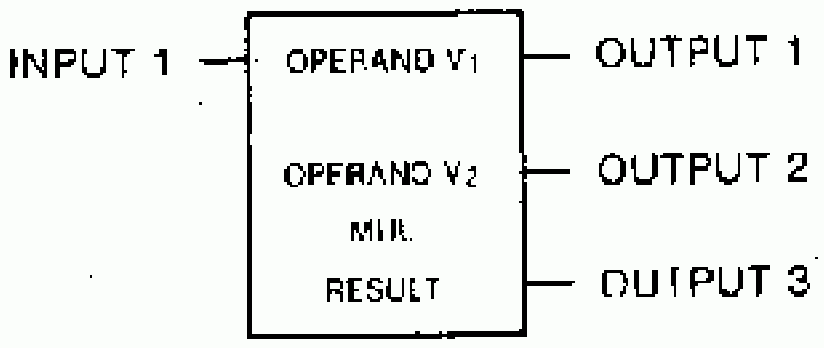


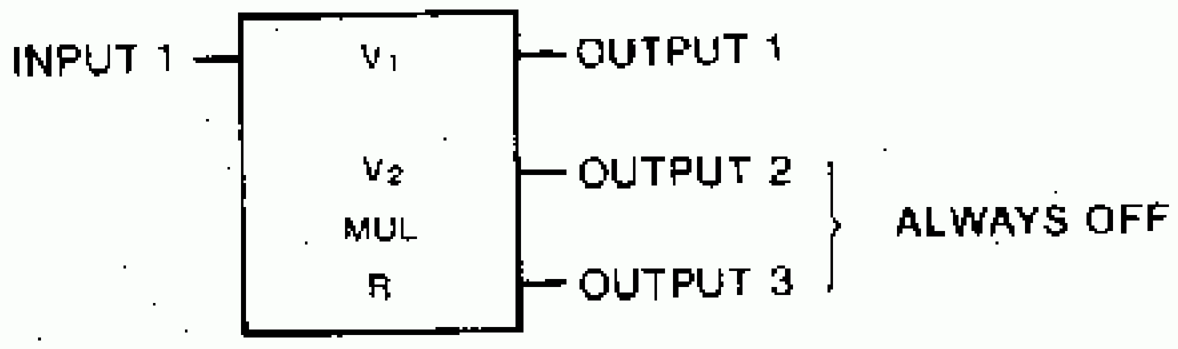
Fig. 4.23 MUL General Form

- Multiply operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.21, specify any of constant K, the reference number 3XXXX of an input register, and the reference number 4XXXX of a holding register for each of the top and middle elements.
- Only reference number 4XXXX of a holding register may be specified for the bottom element.
- MUL is the symbol denoting the multiply.

Table 4.21 Elements of Multiply Function

Element Position	Specified Number	Description
Top	Any one of the following: • Constant K (0000−9999) • 3××××(30001 30256) • 4××××(40001−49999)	•When constant K is specified, the value is the operand (V_1 = 0 to 9999). •When 3×××× and 4×××× are specified, the contents are the operand (V_2 = 0 to 9999).
Middle		•When constant K is specified, the value is the operand (V_2 = 0 to 9999). •When 3×××× and 4×××× are specified, the contents are the operand (V_2 = 0 to 9999).
Bottom	4××××(40001−49998)	Higher-place 4 digits of result of multiply function is stored in 4×××× and lower-place 4 digits in 4××××+1.

(2) Multiply Function



- By the multiply (MUL), $V_1 \times V_2$ is calculated when the input 1 is ON. The result is treated as follows. The four higher-place digits of $V_1 \times V_2$ are stored in R and the four lower-place digits in R+1. Output 1 is turned on.
- The result remains in R and R+1 even after the input 1 is turned from on to off.

(3) Multiply Operation

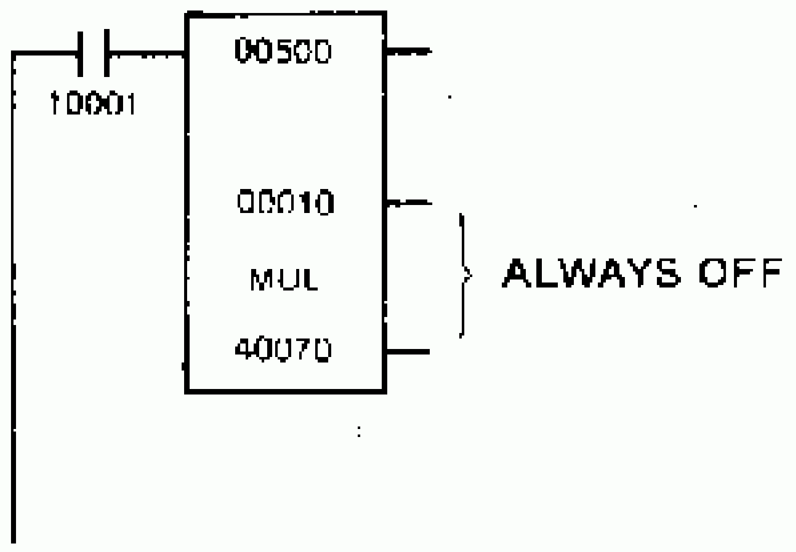
Table 4.22 shows a multiply operation (MUL).

Table 4.22 Multiply Operation

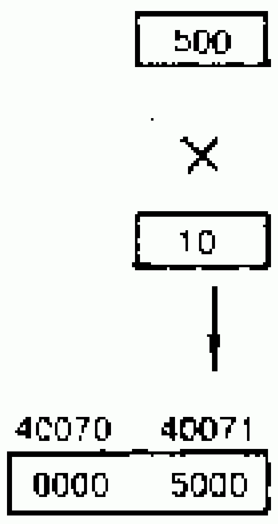
Input 1	Condition	Operation	Output 1
ON	None	$V_1 \times V_2 \rightarrow R$ (Higher-place 4 digits), R+1 (Lower-place 4 digits)	ON
OFF	None	Not operated.	OFF

(4) Example — Multiply

Example 1



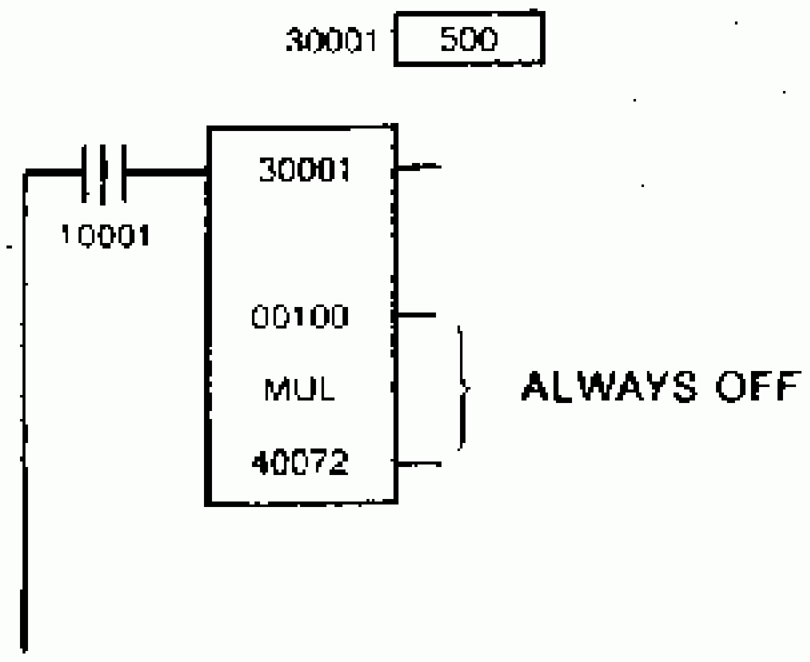
(a) NET #37



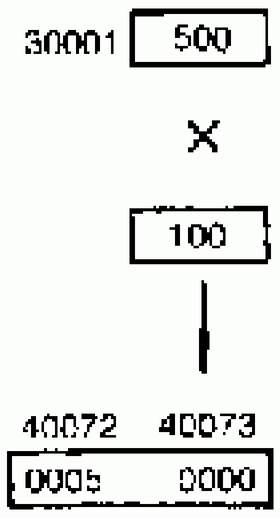
(b) MUL Operation

MUL in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 is turned on. The result remains in 40070 and 40071 even after input relay 10001 is turned off.

Example 2



(a) NET #38



(b) MUL Operation

MUL in (a) executes the operation of (b) only during the scanning cycle when input relay 10001 is turned on. The output 1 is turned on. The result remains in 40072 and 40073.

4.5.7 Double-precision Multiply (DMUL)

(1) Form

- Fig. 4.24 shows the form of double-precision multiply (DMUL).

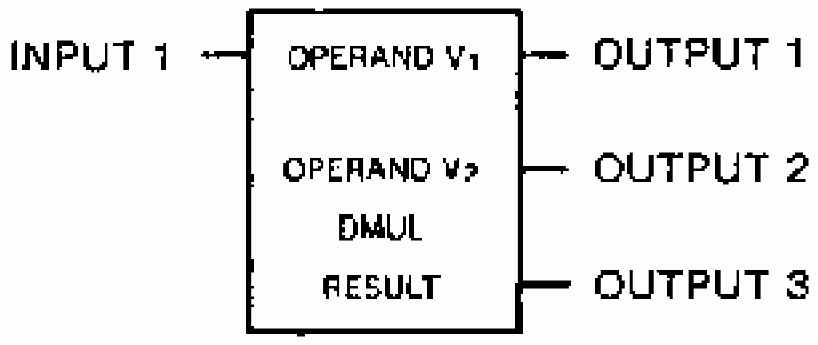


Fig. 4.24 DMUL General Form

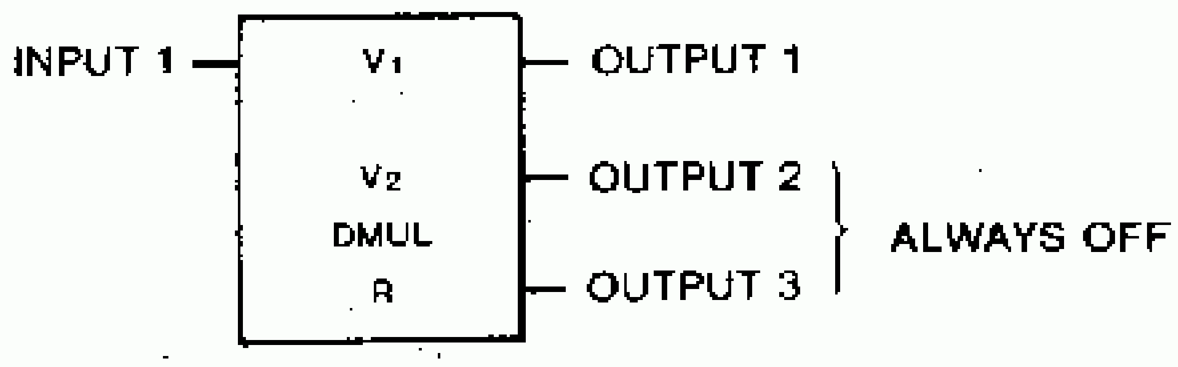
4.5.7 Double-precision Multiply (Cont'd)

- Double-precision multiply operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.23, specify either reference number 3XXXX of an input register or reference number 4XXXX of a holding register for each of the top and bottom elements.
- Only reference number 4XXXX of a holding register may be specifed for the bottom element.
- DMUL is the symbol denoting the double-precision multiply.

Table 4.23 Elements of Double-precision Multiply Function

Element Position	Specified Number	Description
Top	Either one of the following: • 3×××× (30001–30255) • 4×××× (40001–49998)	Operand (V_1 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× [V_1H] 4××××+1 [V_1L] V_1H: Higher-place 4 digits of V_1 V_1L: Lower-place 4 digits of V_1
Middle		Operand (V_2 = 0 to 99999999) is stored as follows. Where 4××××, 4×××× [V_2H] 4××××+1 [V_2L] V_2H: Higner-place 4 digits of V_2 V_2L: Lower-place 4 digits of V_2
Bottom	4×××× (40001–49996)	Result of operation ($V_1 \times V_2$ = 0 to 9999999999999999) is stored as follows. 4×××× [$(V_1 \times V_2)_1$] 4××××+1 [$(V_1 \times V_2)_2$] 4××××+2 [$(V_1 \times V_2)_3$] 4××××+3 [$(V_1 \times V_2)_4$] $(V_1 \times V_2)_1$: Most significant 4 digits of $(V_1 \times V_2)$ $(V_1 \times V_2)_2$: Higher-place 4 digits of $(V_1 \times V_2)$ $(V_1 \times V_2)_3$: Lower-place 4 digits of $(V_1 \times V_2)$ $(V_1 \times V_2)_4$: Least significant 4 digits of $(V_1 \times V_2)$

(2) Double-precision Multiply Function



- By the double-precision multiply (DMUL), $V_1 \times V_2$ is calculated when the input 1 is ON. The result is treated as follows.

  The most significant four digits of $V_1 \times V_2$ are stored in R, the second most significant four digits in R+1, the second least significant four digits in R+2, the least significant four digits in R+3, and the output 1 is turned on.

- The result remains in R, R+1, R+2, and R+3 after the input 1 is turned from on to off.

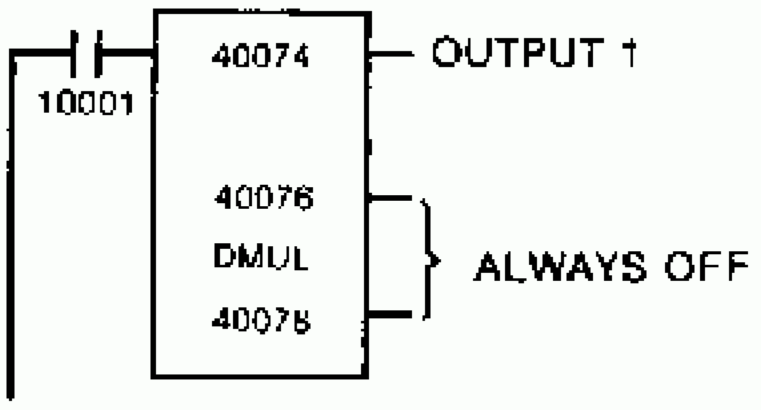
(3) Double-precision Multiply Operation

Table 4.24 shows a double-precision multiply.

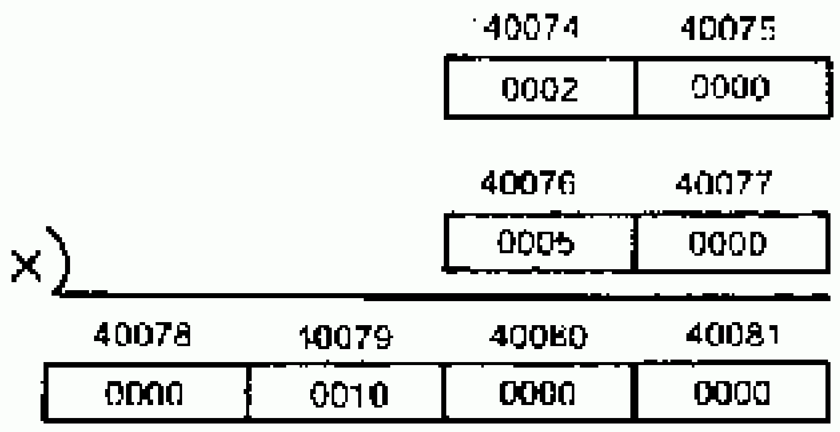
Table 4.22 Double-precision Multiply Operation

Input 1	Condition	Operation	Output 1
ON	None	$V_1 \times V_2 \rightarrow$ R (Most significant 4 digits), R+1 (2nd most significant 4 digits), R+2 (2nd least significant 4 digits), R+1 (Least significant 4 digits).	ON
OFF	None	Not operated.	OFF

(4) Example — Double-precision Multiply



(a) NET #39



(b) DMUL Operation

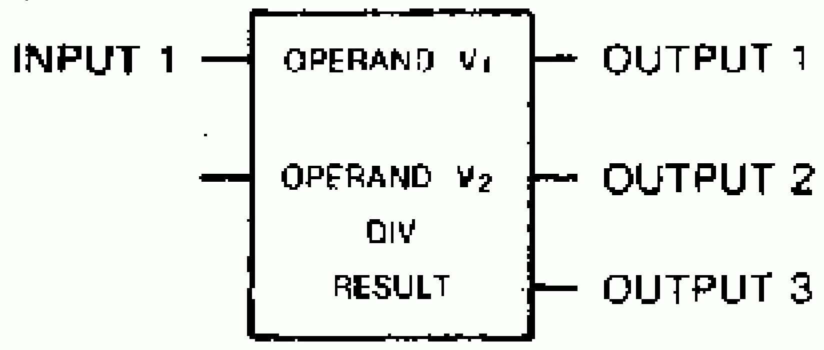
DMUL in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 is turned on. The result remains in 40078~40081 even after input relay 10001 is turned off

4. 5. 8 Divide (DIV)

(1) Form

· Fig. 4.25 shows the form of divide (DIV).

Fig. 4.25 DIV General Form



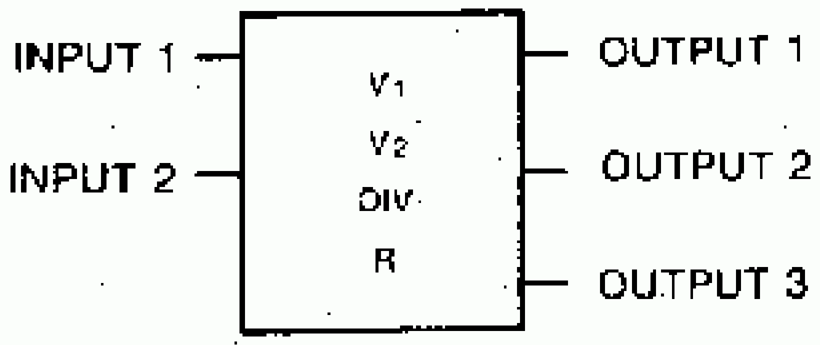
· Divide operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.25, specify any of constant K, reference number 3×××× of an input register, and reference number 4×××× of a holding register for each of the top and bottom elements.

· DIV is the symbol denoting the divide.

Table 4.25 Elements of Divide Funcjion

Element Position	Specified Number	Description
Top	Any one of the following: • Constant K (0—9999) • 3×××× (30001-30255) • 4×××× (40001—49998)	• When constant K is specified, the value is the operand (V_1 = 0 to 9999). • When 3×××× and 4×××× are specified, the operand (V_1 = 0 to 9998 9999) is stored as follows. Example, 4×××× [V_1H] 4××××+1 [V_1L] V_1H: Higher-place 4 digits of V_1 V_1L: Lower place 4 digits of V_1
Middle	Any one of the following: • Constant K (1—9999) • 3×××× (30001-30256) • 4×××× (40001—49999)	• When constant K is specified, the value is the operand (V_2 = 1 to 9999). • When 3×××× and 4×××× are specified, the contents are the operand (V_2 = 1 to 9999).
Bottom	4×××× (40001—49998)	Result of divide function ($V_1 \div V_2$) is stored as follows. Where input 2 is OFF: 4×××× [ ] The quotient of ($V_1 \div V_2$); 4××××+1 [ ] The remainder of ($V_1 \div V_2$) Where input 2 is ON: 4×××× [ ] The integer quotient of ($V_1 \div V_2$); 4××××+1 [ ] The decimal quotient of ($V_1 \div V_2$)

(2) Divide Function



By the divide (DIV), $V_1 \div V_2$ is calculated when the input 1 is ON. The result is treated as follows.

(a) If the input 2 is OFF,

The quotient of $V_1 \div V_2$ is stored in R and the remainder in R+1. Only the output 1 is turned on.

(b) If the input 2 is ON,

The integer part of the quotient of $V_1 \div V_2$ is stored in R and the decimal part (rounded off to the fifth decimal place) in R+1. Only the output 1 is turned on.

(c) The result remains in R and R+1 even after the input 1 is turned from on to off.

(d) In the following cases, divide operation is not executed and zero is placed in each of R and R+1.

- When $V_2 = 0$, the output 3 is turned on.
- If the quotient or the integer part of quotient overflows in R, the output 2 is turned on.

(3) Divide Operation

Tables 4.26 and 4.27 show a divide operation (DIV).

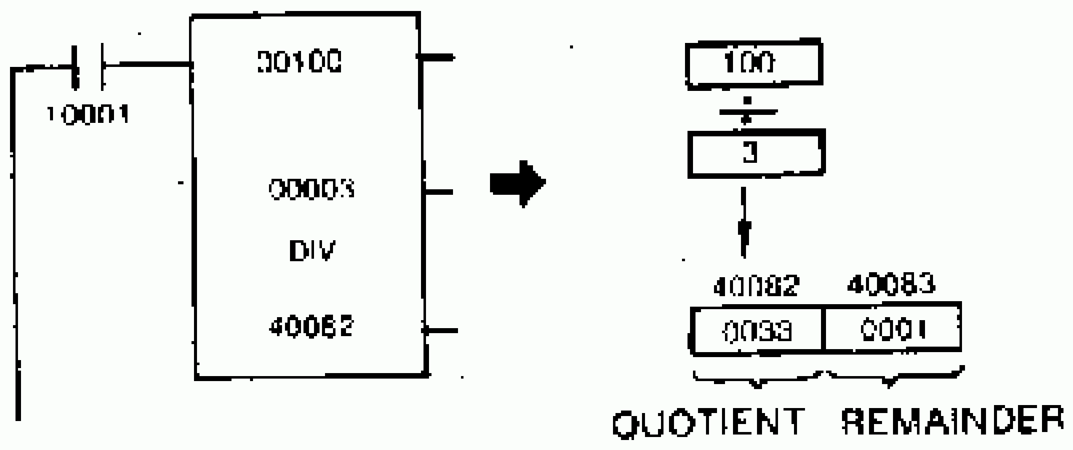
Table 4.26 Divide Operation (Constant in Top Place)

Input 1	Input 2	Condition	Operation	Output 1	Output 2	Output 3
ON	OFF	$V_2 \neq 0$	$V_1 \div V_2 \to$ R (Quotient) R+1 (Remainder)	ON	OFF	OFF
		$V_2 = 0$	$0 \to$ R R+1	OFF	OFF	ON
ON	OF	$V_2 \neq 0$	$V_1 \div V_2 \to$ R (Integer quotient) R+1 (Decimal quotient)	ON	OFF	OFF
		$V_2 = 0$	$0 \to$ R R+1	OFF	OFF	ON
OFF	ON OFF	None	Not operated.	OFF	OFF	OFF

Table 4.27 Divide Operation (Register in Top Place)

Input 1	Input 2	Condition	Operation	Output 1	Output 2	Output 3
ON	OFF	$V_2 \neq 0$, $V_{1H} < V_2$	$(V_{1H} \times 10000 + V_{1L}) \div V_2 \to$ R (Quotient) R+1 (Remainder)	ON	OFF	OFF
		$V_2 \neq 0$, $V_{1H} \geqq V_2$	$0 \to$ R R+1	OFF	ON	OFF
		$V_2 = 0$	$0 \to$ R R+1	OFF	OFF	ON
ON	ON	$V_2 \neq 0$, $V_{1H} < V_2$	$(V_{1H} \times 10000 + V_{1L}) \div V_2 \to$ R (Integer quotient) R+1 (Decimal quotient)	ON	OFF	OFF
		$V_2 \neq 0$, $V_{1H} \geqq V_2$	$0 \to$ R R+1	OFF	ON	OFF
		$V_2 = 0$	$0 \to$ R R+1	OFF	OFF	ON
OFF	ON OFF	None	Not operated.	OFF	OFF	OFF

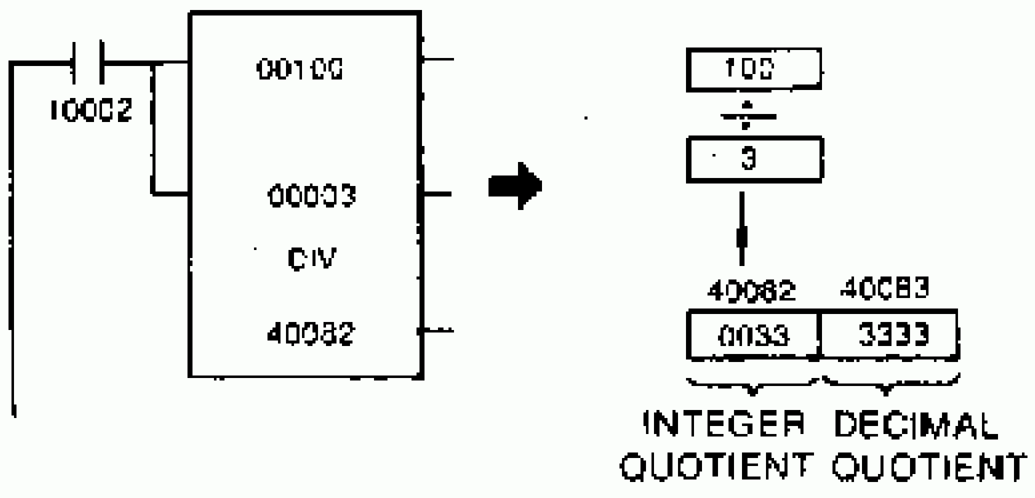
(4) Example—Divide



(a) NET #40 (b) DIV Operation

Example 1

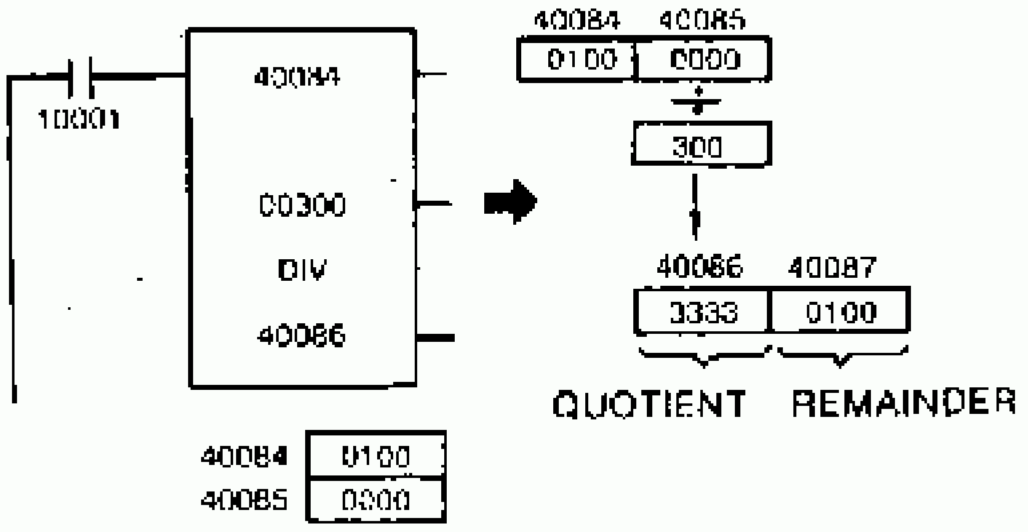
DIV in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 is turned on. The result remains in 40082 and 40083 even after input relay 10001 is turned off.



(a) NET #40 (b) DIV Operation

Example 2

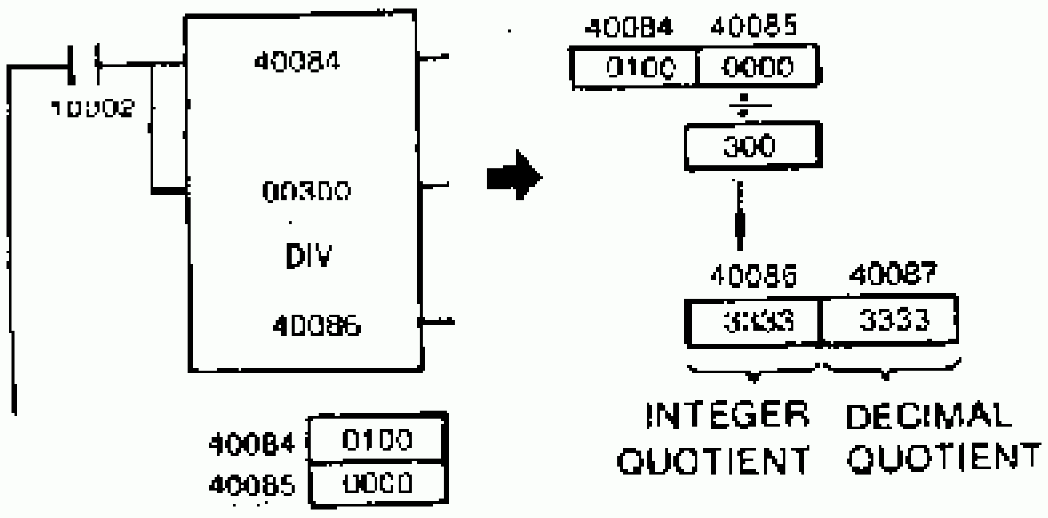
DIV in (a) executes the operation of (b) when input relay 10002 is ON. Only the output 1 is turned on. The result remains in 40082 and 40083 even after input relay 10002 is turned off.



(a) NET #41 (b) DIV Operation

Example 3

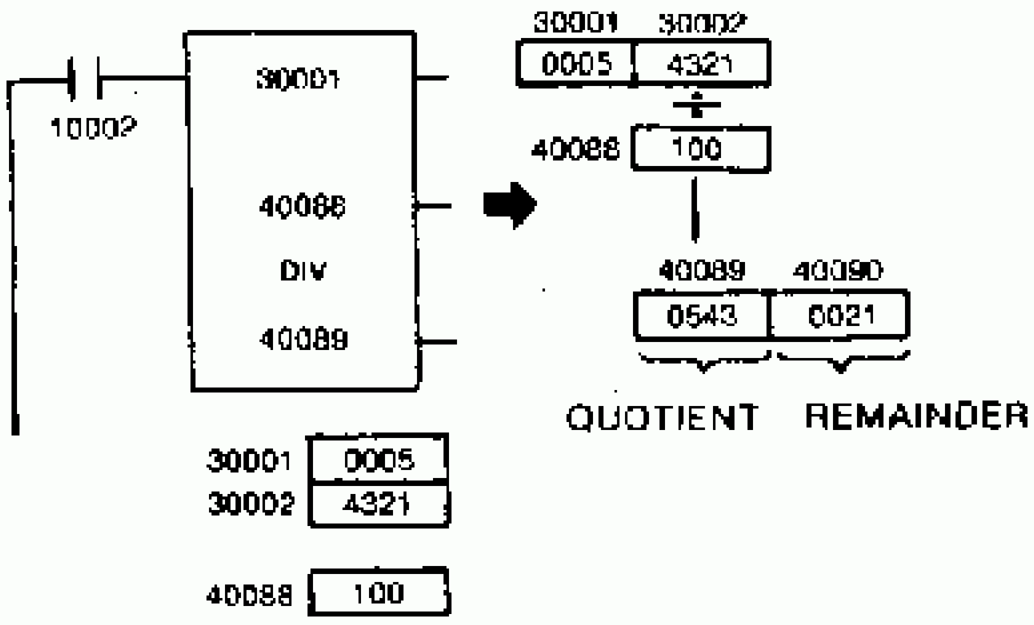
DIV in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 is turned on. The result remains in 40086 and 40087 even after input relay 10001 is turned off.



(a) NET #41 (b) DIV Operation

Example 4

DIV in (a) executes the operation of (b) when input relay 10002 is ON. Only the output 1 is turned on. The result remains in 40086 and 40087 even after input relay 10002 is turned off.

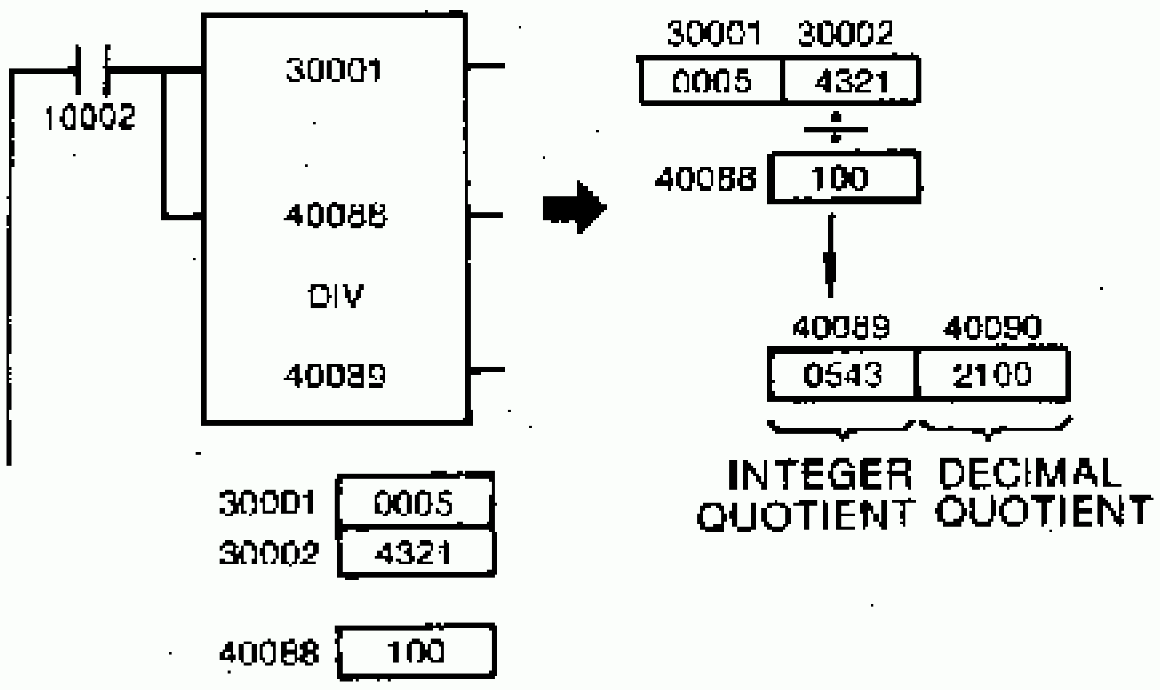


(a) NET #42 (b) DIV Operation

Example 5

DIV in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 is turned on. The result remains in 40089 and 40090 even after input relay 10001 is turned off.

4. 5. 8 Divide (Cont'd)



(a) NET #42 (b) DIV Operation

Example 6

DIV in (a) executes the operation of (b) when input relay 10002 is ON. Only the output 1 is turned on. The results remains in 40089 and 40090 even after input relay 10002 is turned off.

4. 5. 9 Double-precision Divide Function (DDIV)

(1) Form

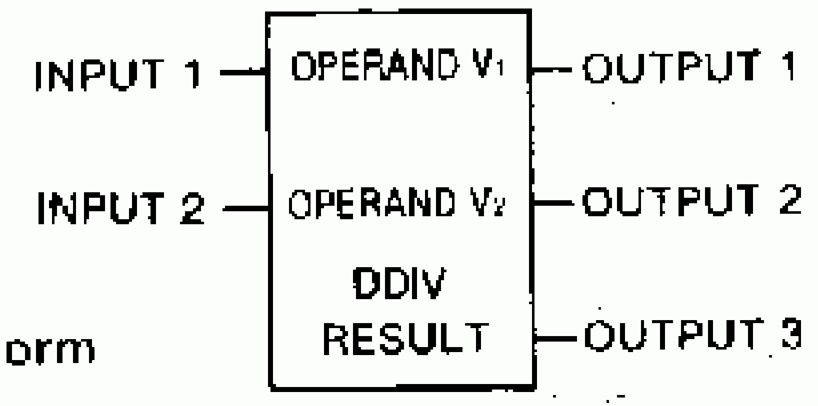


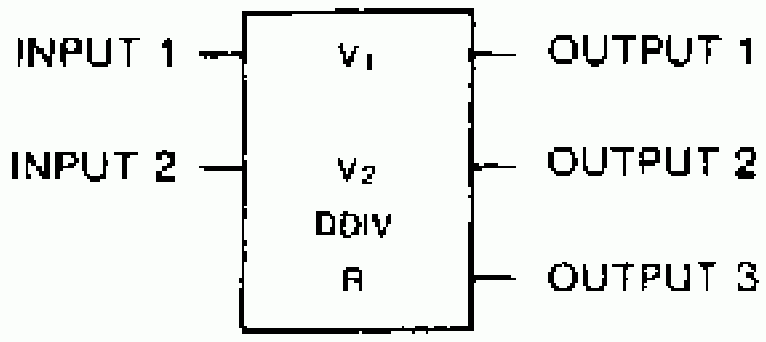
Fig. 4.26 DDIV General Form

- Fig. 4.26 shows the form of double-precision divide (DDIV).
- Double-precision divide operation requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.28, specify either the reference number 3×××× of an input register or 4××××of a holding register for each of the top and middle elements.
- Only reference number 4×××× of a holding register may be specified for the bottom element.
- DDIV is the symbol denoting the double-precision divide.

Table 4. 28 Elements of Double-precision Divide Function

Element Position	Specified Number	Description
Top	Either one of the following: • 3×××× (30001−30253) • 4×××× (40001−49996)	The operand (V_1 = 0 to 9999999999999999) is stored as follows Where 4××××. 4×××× : V_1H_1 / 4××××+1 : V_1H_2 / 4××××+2 : V_1L_1 / 4××××+3 : V_1L_2 V_1H_1: Most significant 4 digits of V_1 V_1H_2: 2nd most significant 4 digits of V_1 V_1L_1: 2nd least significant 4 digits of V_1 V_1L_2: Least significant 4 digits of V_1
Middle	Either one of the following: • 3×××× (30001−30255) • 4×××× (40001−49998)	The operand (V_2 = 1 to 99999999) is stored as follows. Where 4××××, 4×××× : V_2H / 4××××+1 : V_2L V_2H: Higher-place 4 digits of V_2 V_2L: Lower-place 4 digits of V_2
Bottom	4×××× (40001−49996)	The result of operation ($V_1 \div V_2$) is stored as follows. Where input 2 is OFF, 4×××× : Higher-place 4 digits / 4××××+1 : Lower-place 4 digits — Quotient of ($V_1 \div V_2$) 4××××+2 : Higher-place 4 digits / 4××××+3 : Lower-place 4 digits — Remainder of ($V_1 \div V_2$) Where input 2 is ON, 4×××× : Higher-place 4 digits / 4××××+1 : Lower-place 4 digits — Integer quotient of ($V_1 \div V_2$) 4××××+2 : Higher-place 4 digits / 4××××+3 : Lower-place 4 digits — Decimal quotient of ($V_1 \div V_2$)

(2) Double-precision Divide Function



By the double-precision divide (DDIV), $V_1 \div V_2$ is calculated when the input 1 is ON. The result is treated as follows.

(a) If the input 2 is OFF,

The four higher-place digits of the quotient of $V_1 \div V_2$ are stored in R and the four lower-place digits in R+1.
The four higher-place digits of the remainder of $V_1 \div V_2$ are stored in R+2 and the four lower-place digits in R+3.

Only the output 1 is turned on.

(b) If the input 2 is ON,

- The four higher-place digits of the integer part of the quotient of $V_1 \div V_2$ are stored in R and the four lower-place digits in R+1.
- The four high-place digits of the decimal part of the quotient of $V_1 \div V_2$ are stored in R+2 and the four lower-place digits in R+3.
- Only the output 1 is turned on.

(c) The result remains in each of R, R+1, R+2, and R+3 even after the input 1 is turned from on to off.

(d) In the following cases, the divide operation is not executed and zero is placed in each of R, R+1, R+2, and R+3.

- When $V_2 = 0$, the output 3 is turned on.
- If the quotient or the integer part of quotient overflows in R and R+1, the output 2 is turned on.

Example: Where $V_1 = 5000000000$, and $V_2 = 10$, the quotient 500000000 cannot be stored in R and R+1.

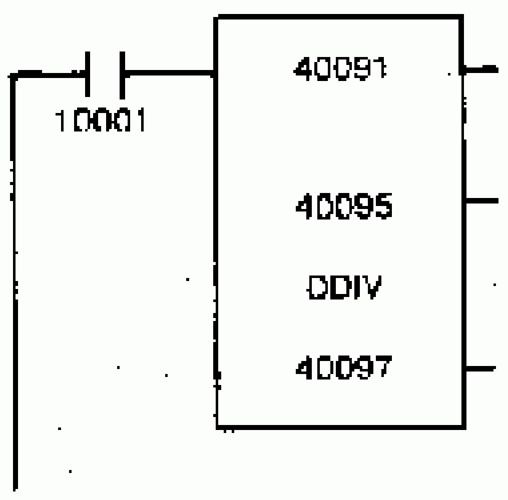
(3) Double-precision Divide Operation

Table 4.29 shows a double-precision divide operation (DDIV).

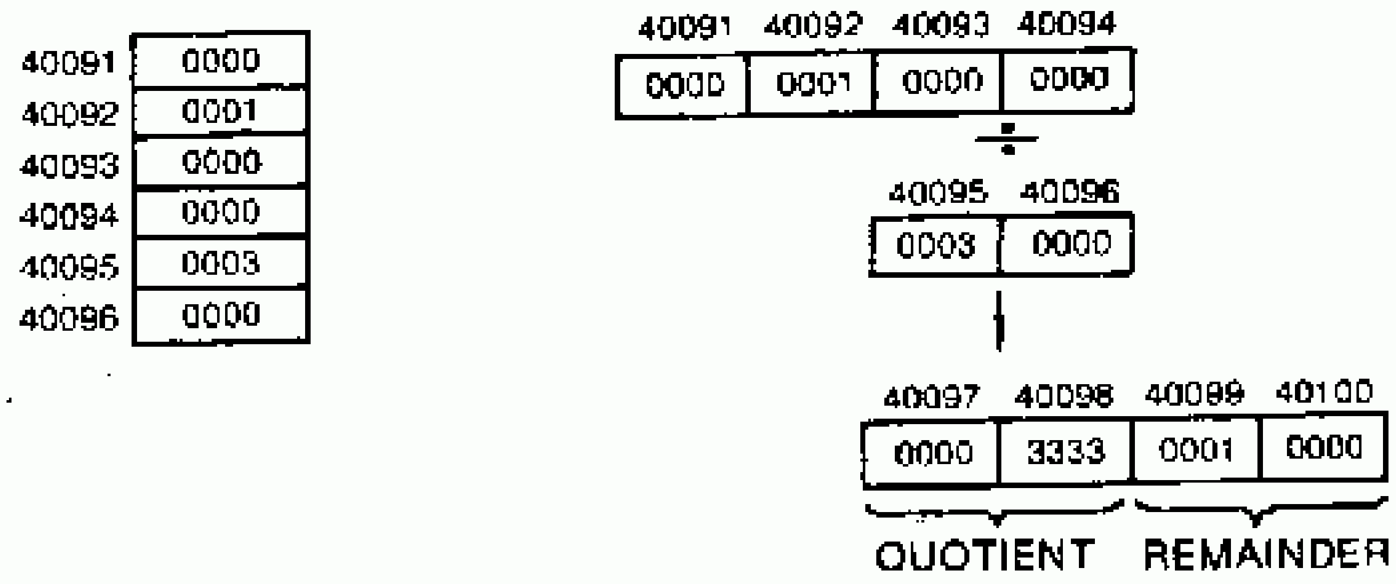
Table 4.29 Double-precision Divide Operation

Input 1	Input 2	Condition	Operation	Output 1	Output 2	Output 3
ON	OFF	$V_2=(V_{2H}\times10^4+V_{2L})\neq0$, $(V_{1HH}\times10^4+V_{1HL})<V_2$ (Higher-place 8 digits of V_1)	$(V_{1HH}\times10^{12}+V_{1HL}\times10^{8}+V_{1LH}\times10^{4}+V_{1LL})\div(V_{2H}\times10^{4}+V_{2L})$ → [V_1 ÷ V_2] R (Higher-place 4 digits of quotient), R+1 (Lower-place 4 digits of quotient), R+2 (Higher-place 4 digits of remainder), R+2 (Lower-place 4 digits of remainder).	ON	OFF	OFF
		$V_2\neq0$, $(V_{1HH}\times10^4+V_{1HL})\geq V_2$	0→R, R+1, R+2, R+3	OFF	ON	OFF
		$V_2=0$	0→R, R+1, R+2, R+3	OFF	OFF	ON
ON	ON	$V_2=(V_{2H}\times10^4+V_{2L})\neq0$, $(V_{1HH}\times10^4+V_{1HL})<V_2$ (Higher-place 8 digits of V_1)	$(V_{1HH}\times10^{12}+V_{1HL}\times10^{8}+V_{1LH}\times10^{4}+V_{1LL})\div(V_{2H}\times10^{4}+V_{2L})$ → [V_1 ÷ V_2] R (Higher-place 4 digits of integer quotient), R+1 (Lower-place 4 digits of integer quotient), R+2 (Higher-place 4 digits of decimal quotient), R+3 (Lower-place 4 digits of decimal quotient).	ON	OFF	OFF
		$V_2\neq0$, $(V_{1HH}\times10^4+V_{1HL})\geq V_2$	0→R, R+1, R+2, R+3	OFF	ON	OFF
		$V_2=0$	0→R, R+1, R+2, R+3	OFF	OFF	ON
OFF	ON, OFF	None	Not operated.	OFF	OFF	OFF

(4) Example—Double-precision Divide



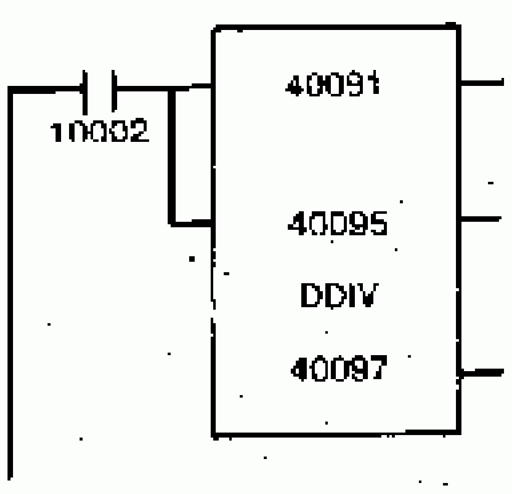
(a) NET #43

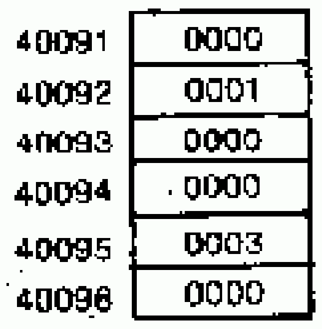


(b) DDIV Operation

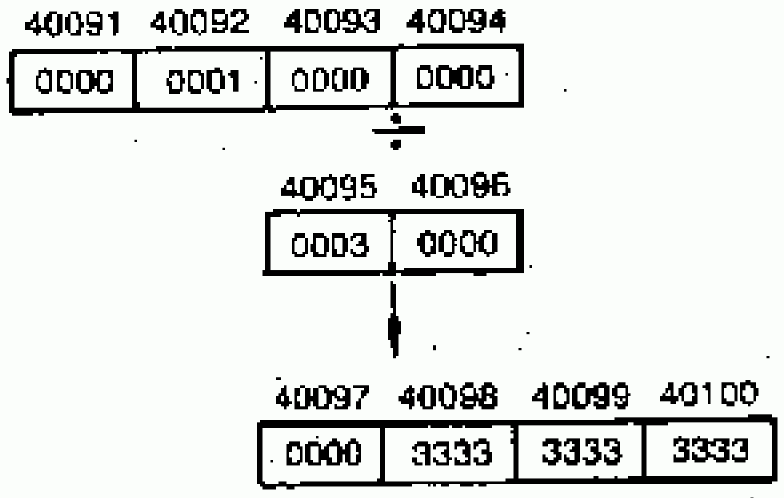
Example 1

DDIV in (a) executes the operation of (b) when input relay 10001 is ON. Only the output 1 is turned on. The result remains in 40097 to 40100 even after input relay 10001 is turned off.





(a) NET #43



(b) DDIV Contents

Example 2

DDIV in (a) executes the operation of (b) when input relay 10002 is ON. Only the output 1 is turned on. The result remains in 40097 to 40100 even after input relay 10002 is turned off.

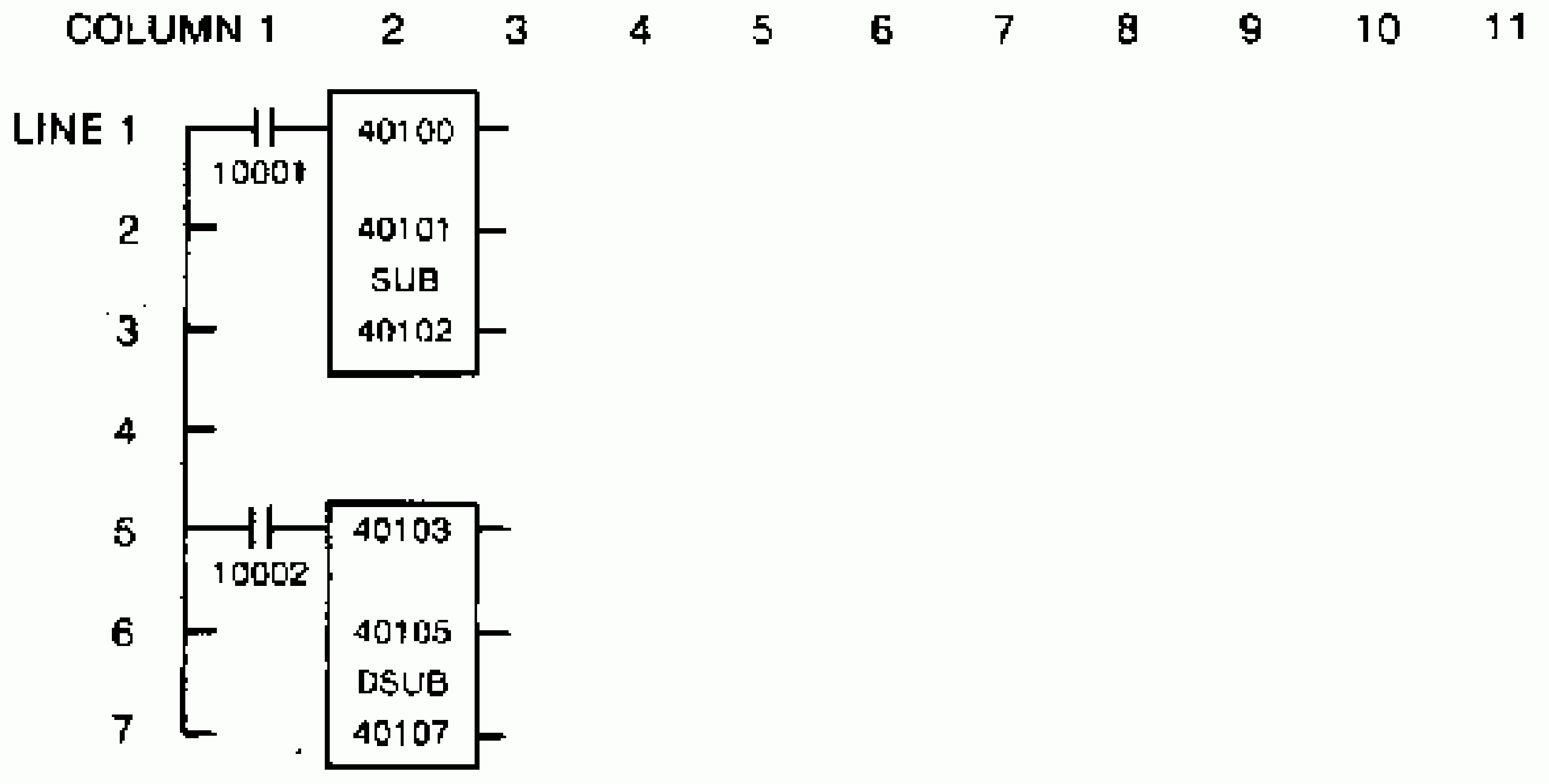
4. 5. 10 Programming Arithmetic Logic and Precautions

In all arithmetic operations, add, subtraction and multiply function require only input 1, and divide requires only inputs 1 and 2. But the P190 CRT programming panel gives display as each output element line which may be connected to each input element line.

(1) Programming Arithmetic Logic

All arithmetic operations require three elements placed vertically (top, middle, and bottom) in a network. They can be used at any intersection of the 7 lines-by-10 columns matrix, however the top element (operand) cannot be used on either line 6 nor line 7.

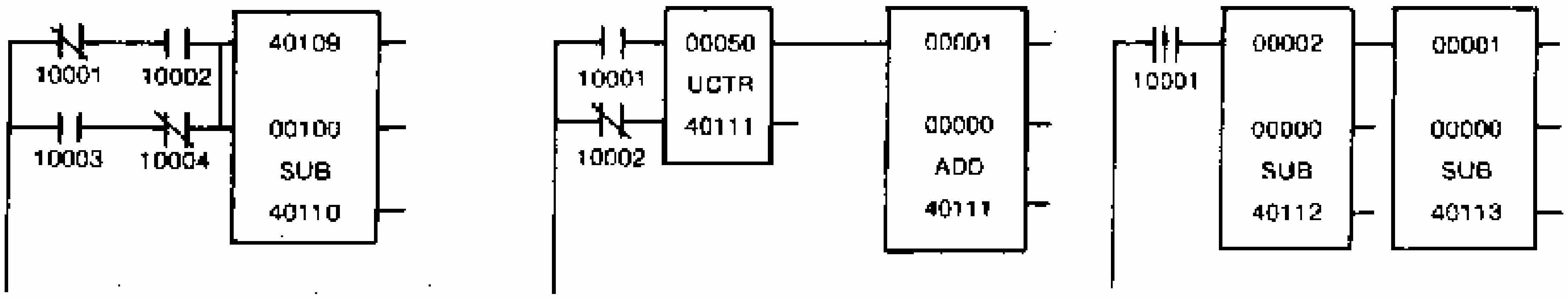
Example



NET # 44

(2) Inputs of Arithmetic Logic

Inputs to the arithmetic logic may be outputs of relays, timers, counters, data processing circuits and other arithmetic operations.

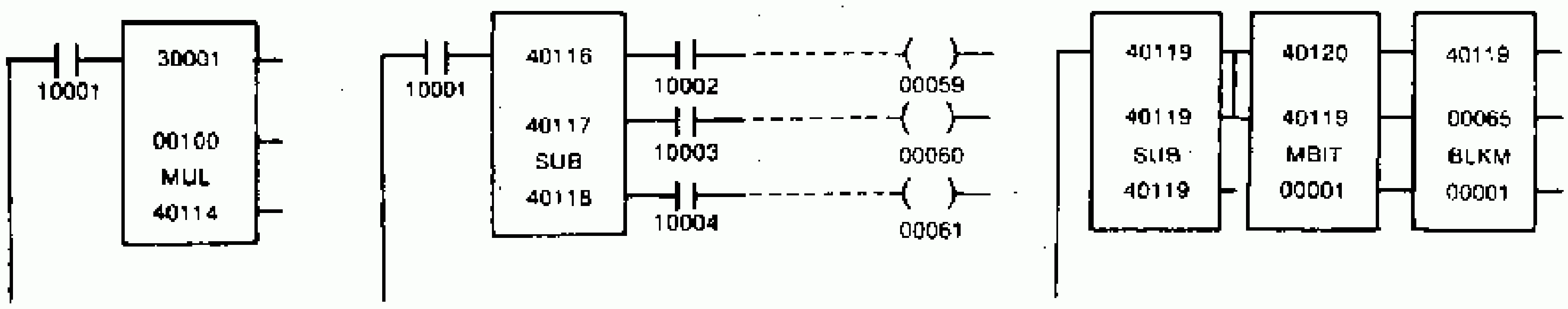


Example 1 **Example 2** **Example 3**

(3) Outputs of Arithmetic Logic

Coils need not be connected to three output nodes (1, 2, and 3) of an arithmetic function. A relay contact may be connected to the output nodes on the right or connect the output nodes directly to an input node of an arithmetic circuit, except relays.

If results of addition and double-precision addition operations exceed 9999 or 99999999, respectively, only output 1 is ON. When other arithmetic circuits are cascade-connected to outputs of addition or double-precision addition, use proper care.

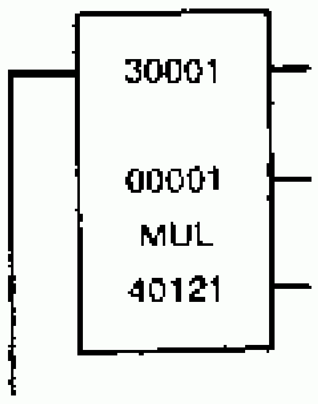


Example 1 **Example 2** **Example 3**

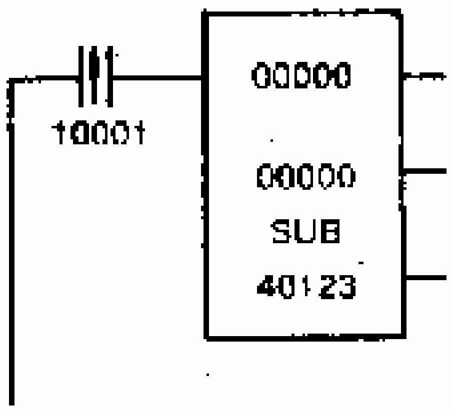
(4) Execution of Arithmetic Operations
(Only One Scanning Cycle)

To execute a constant arithmetic operation, connect the inputs directly to the power rail on the left. To execute it only in one scan, use a transitional contact as an input.

Example 1

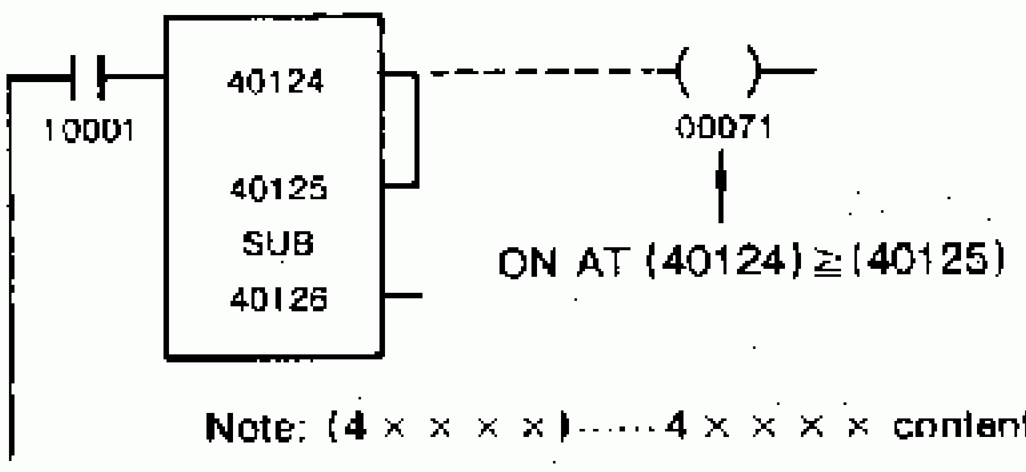


Example 2



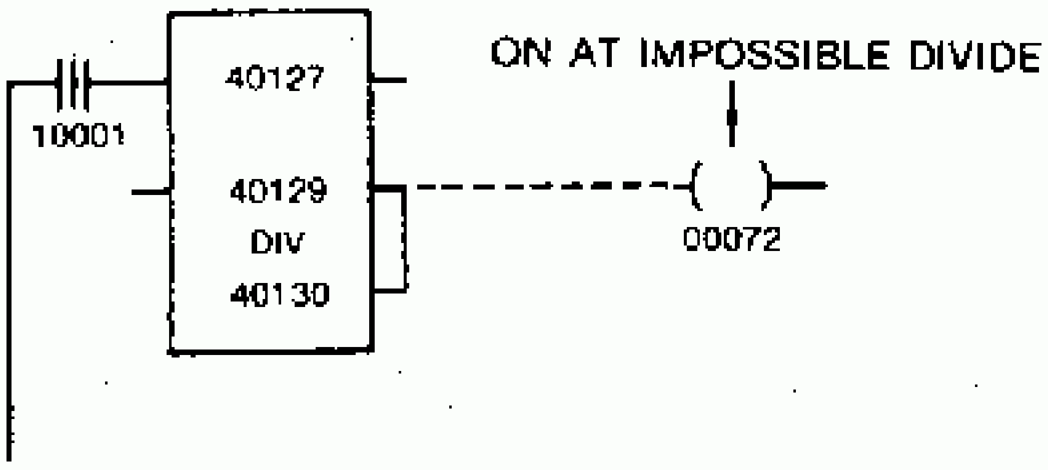
(5) ORed Outputs of Arithmetic Operations

Outputs of the subtraction, the double-precision subtraction, the divide, and double-precision divide can be ORed by using a vertical shunt on the side.

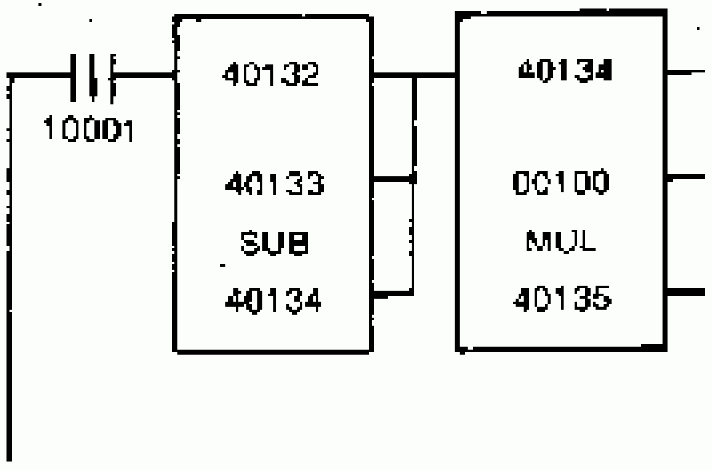


Note: (4 × × × ×)……4 × × × × contents

Example 1



Example 2



Multiply is executed regardless of the result of subtraction (cascade operation).

Example 3

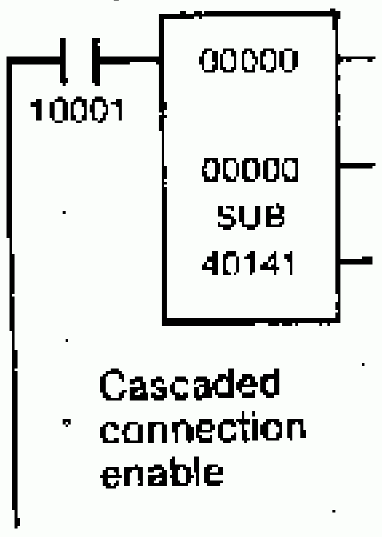
4.5.11 Example–Application Circuits of Arithmetic Functions

(1) Clearing Contents of Holding Registers to 0

- To clear one holding register to 0

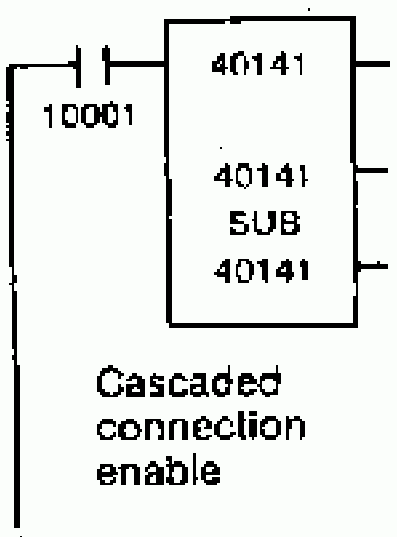
This is performed in three ways as shown in Examples 1 through 3 below. In all cases, the contents of 40141 become 0 when input relay 10001 is ON.

Fig. 4.27 Clearing Contents of Holding Registers to 0

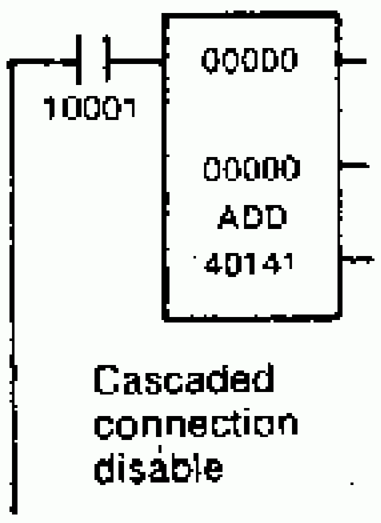


NET #44

Example 1



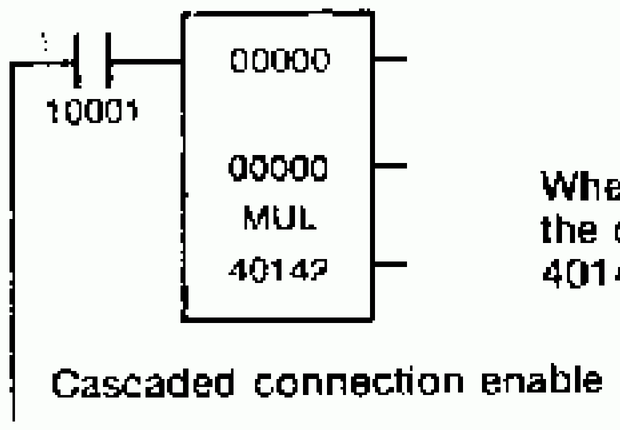
Example 2



Example 3

• To clear two successive hoding registers to 0

Fig. 4.28 Clearing Two Successive Holding Registers to 0



When input relay 10001 is ON, the contents of 40142 and 40143 become 0.

NET #45

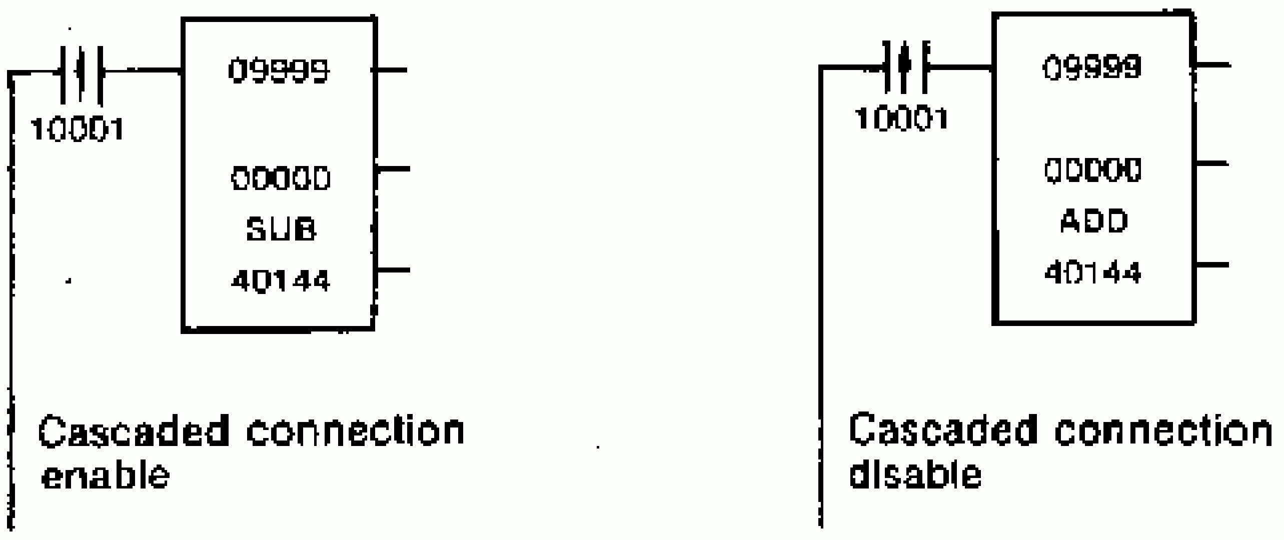
Example

(2) Setting Constant or Contents of a Register to Holding Register

• To set the constant in a holding register

This can be performed in two ways as shown in Examples 1 and 2 below. In both cases, 9999 is set in 40144 during the scanning cycle when input relay 10001 is turned on.

Fig. 4.29 Setting a Constant in a Holding Register



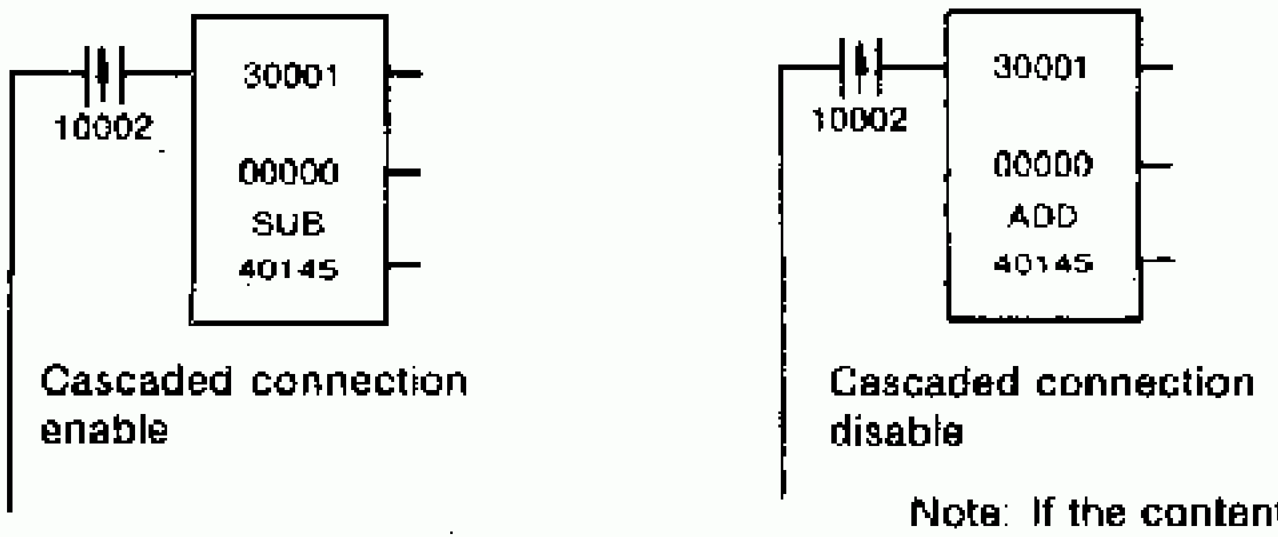
NET #46

Example 1 **Example 2**

• To set the content of a register in a holding register

This can be performed in two ways as shown in Examples 1 and 2 below. In both cases, the contents of 30001 is set in 40145 during the scanning cycle when input relay 10002 is turned on.

Fig. 4.30 Setting the Contents of a Register in a Holding Register



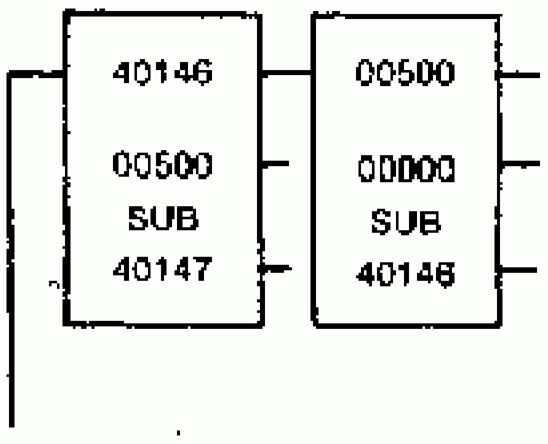
Example 1 NET #47 **Example 2**

Note: If the content of 30001 is greater than 10000, it cannot be set correctly to 40145. The value of 30001 minus 10000 is set to 40145. Use the method of Example 1 in this case.

(3) Limiting Contents of Holding Register

• To limit the maximum value

Fig. 4.31 Limiting the Maximum Value



Note: The maximum value of the contents of 40146 is limited to 500, i.e. (40146) ≦ 500.

Example NET #48

4. 5. 11 Example–Application Circuits of Arithmetic Functions (Cont'd)

- To limit the minimum value

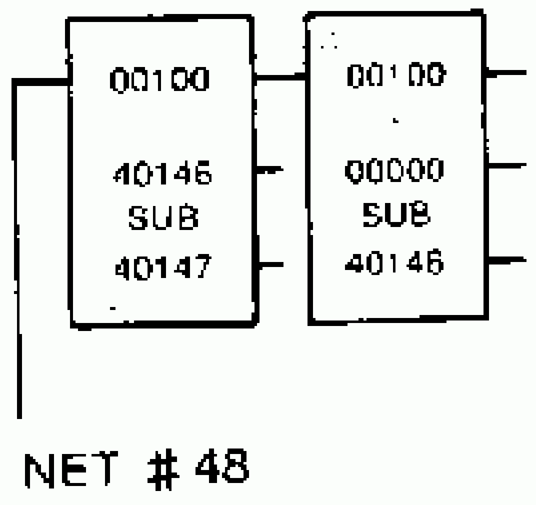


Fig. 4. 32 Limiting the Minimum Value

Note: The minimum value of the contents of 40146 is limited to 100. i.e. 100 ≦ (40146).

Example

- To limit the range

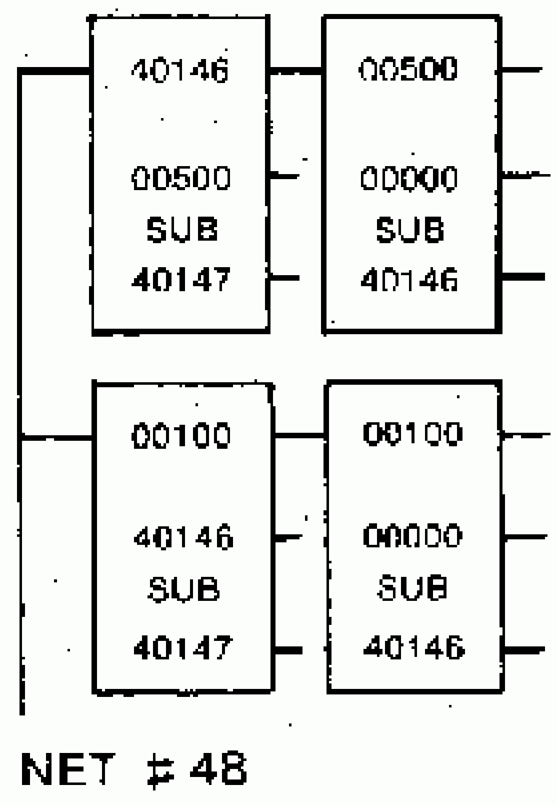


Fig. 4. 33 Limiting the Range

Note: The range of the contents of 40146 is limited in a range of 100 to 500. i.e. 100 ≦ (40146) ≦ 500.

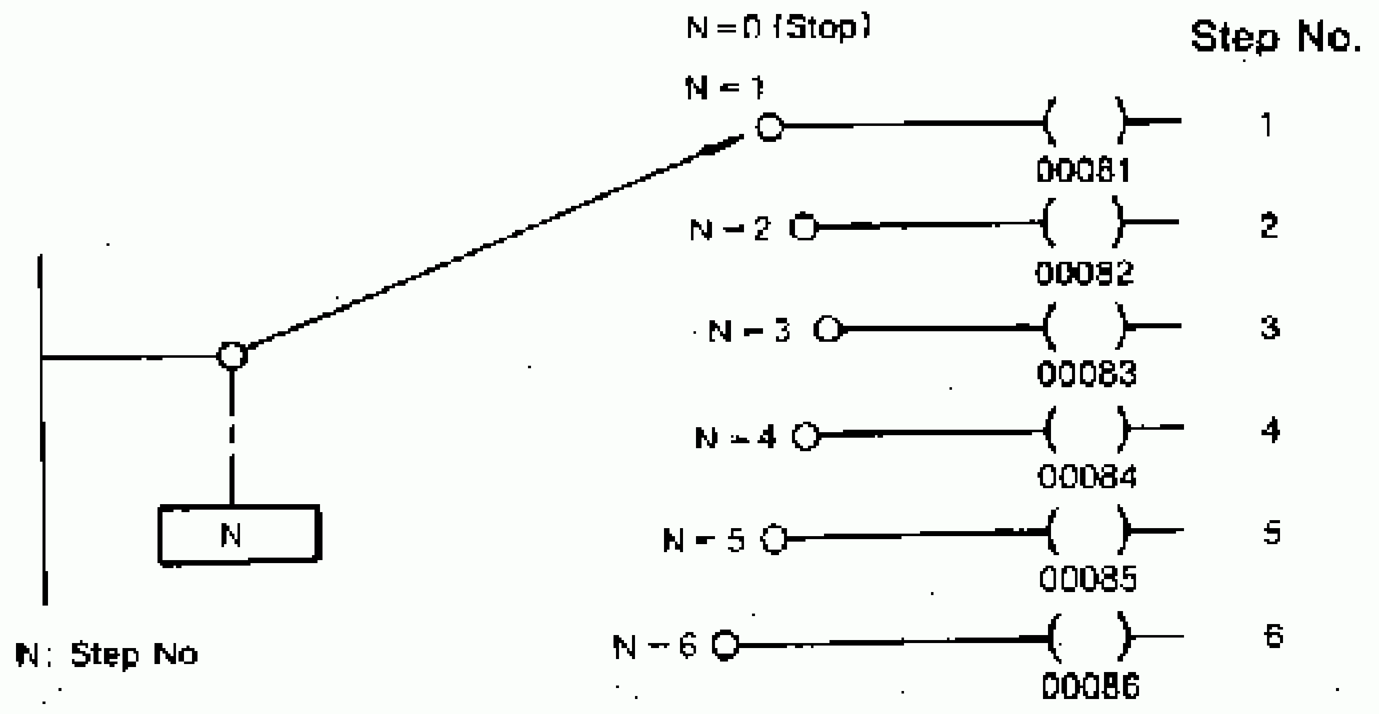
Example

(4) Stepping Switch (Having a few Steps)

A stepping switch having a few steps can be programmed in the following way.

Example: Stepping switch with 6 steps

(a) Step Circuitry



(b) Ladder Diagram

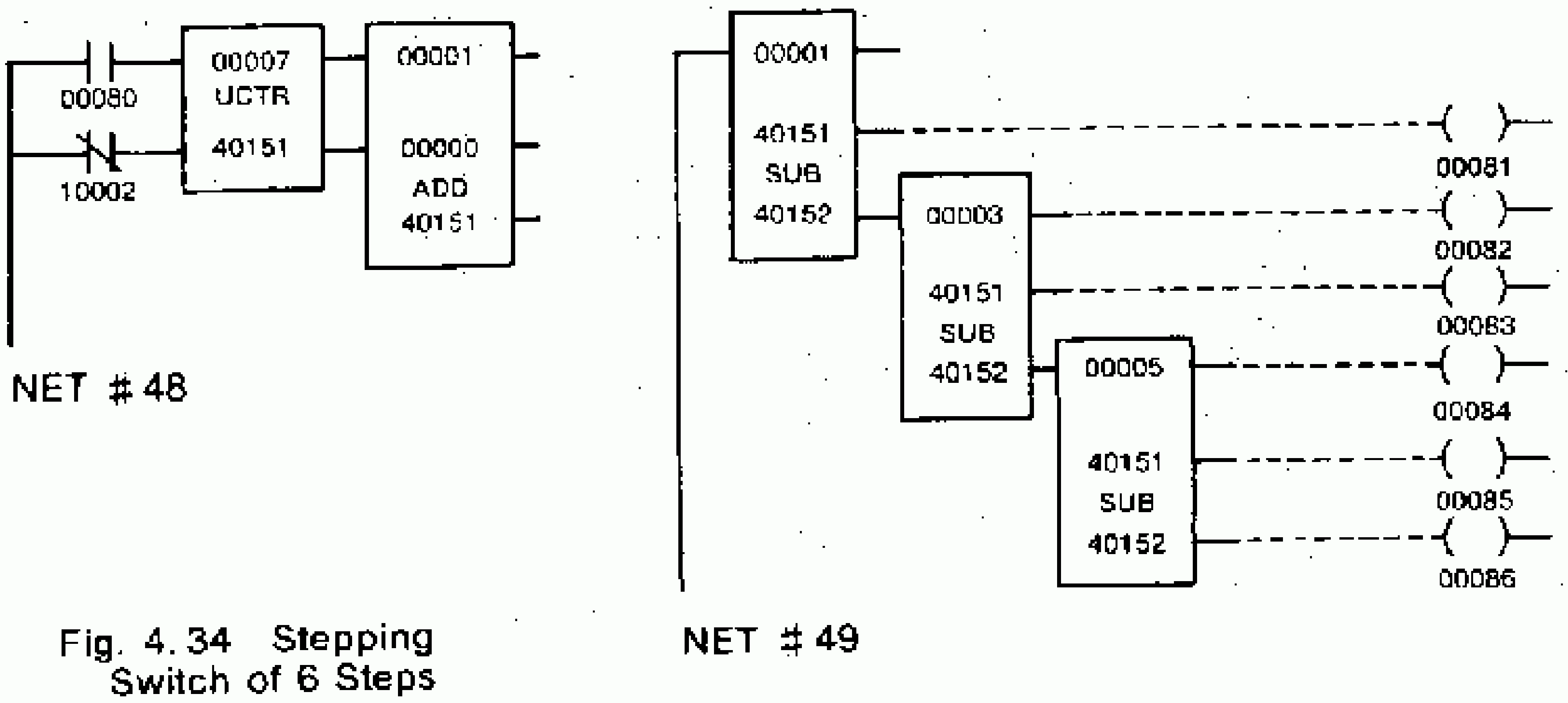


Fig. 4. 34 Stepping Switch of 6 Steps

(5) Scaling

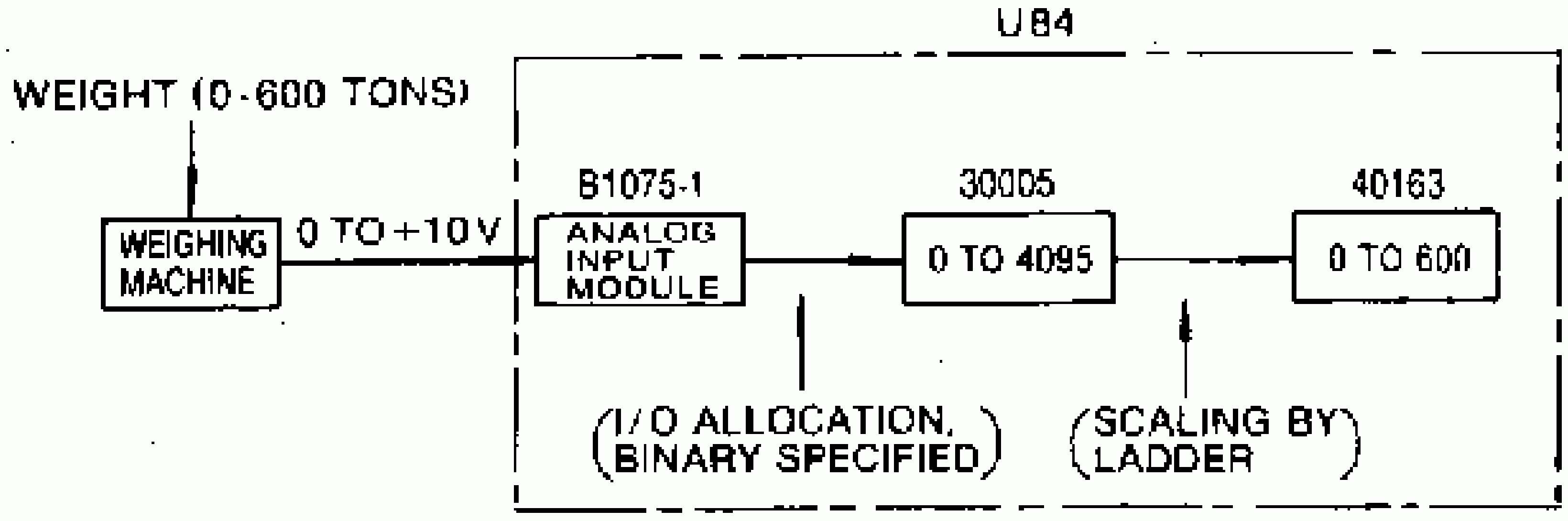
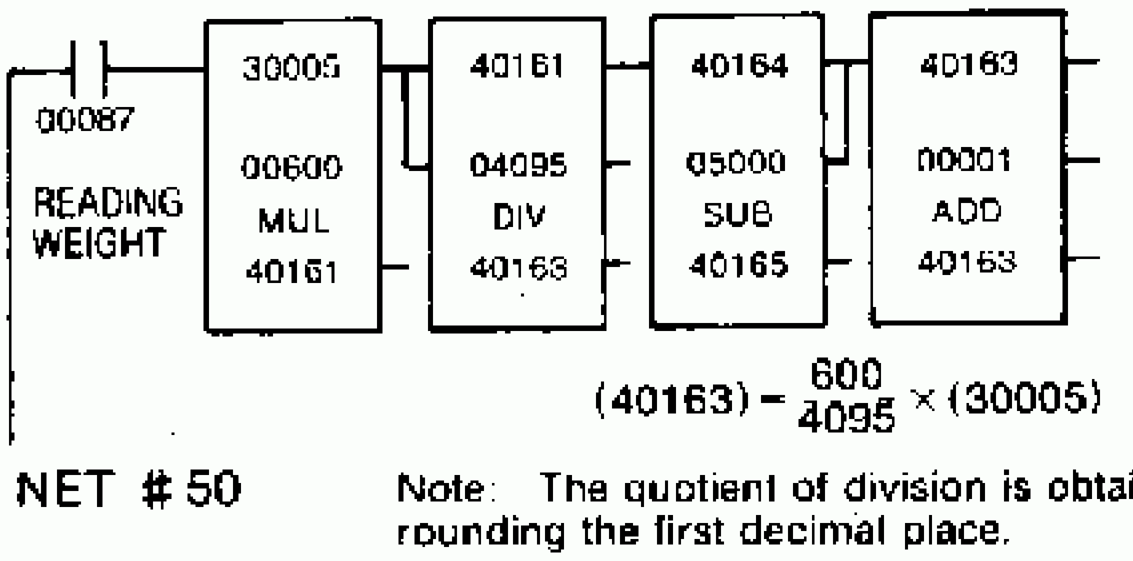


Fig. 4.35 Reading Analog Signal and Scaling

- Assume that a voltage signal of 0 to +10 V comes from the weighing machine to the analog input module B1075-1, as shown in Fig. 4.35, which converts the voltage signal to a numeric data of 0 to 4095 and enters it in input register 30005.
- The numeric data of 0 to 4095 entered in 30005 can be converted to a value of 0 to 600 (tons) by using the ladder circuit shown in Fig. 4.36.



$$(40163) = \frac{600}{4095} \times (30005)$$

Fig. 4.36 Scaling Circuit

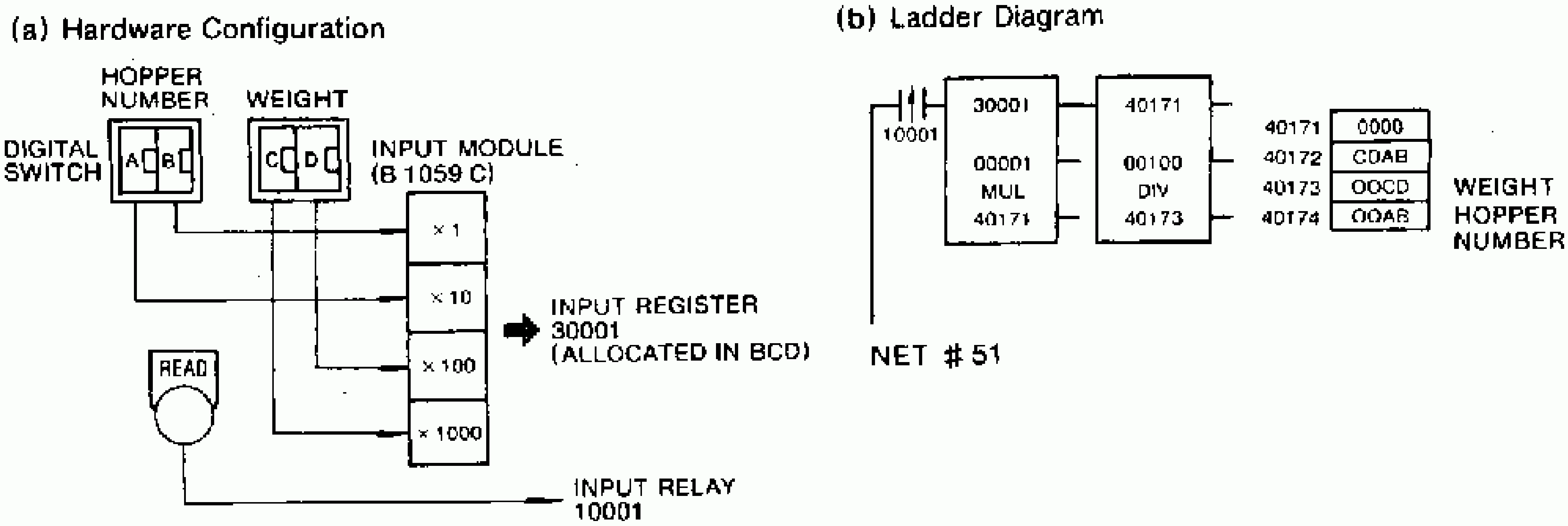
Note: The quotient of division is obtained by rounding the first decimal place.

(6) Saving I/O Modules

It is very easy to let the U84 read a value set by a digital switch or let a digital indicator display data stored in the U84 by allocating the register to the I/O module. The 1000-series I/O module can deal with 4-digit BCD data. Therefore, efficiency is the highest when it deals with four BCD digits.

The following is a method to let a single module process two data having different dimensions by using multiply and divide funcitons. Fig. 4.37 shows an example to let an input module read two numeric data of 2 digits each. Fig. 4.38 shows an example to let an output module output numeric data of 1- and 3-digit.

- Saving input module



Note: A, B, C and D are 0 to 9 and each digit is input by BCD code.

Fig. 4.37 Saving Input Module

4.5.11 Example–Application Circuits of Arithmetic Functions (Cont'd)

As the contents of 30001 is multiplied by the constant 1, 0 is entered in 40171 and CDAB (= C × 1000 + D × 100 + A × 10 + B) in 40172. We use the divide to produce a remainder. By CDAB ÷ 100 + CD with remainder AB, the weight and hopper number are separated in 40173 and 40174.

- Saving output module

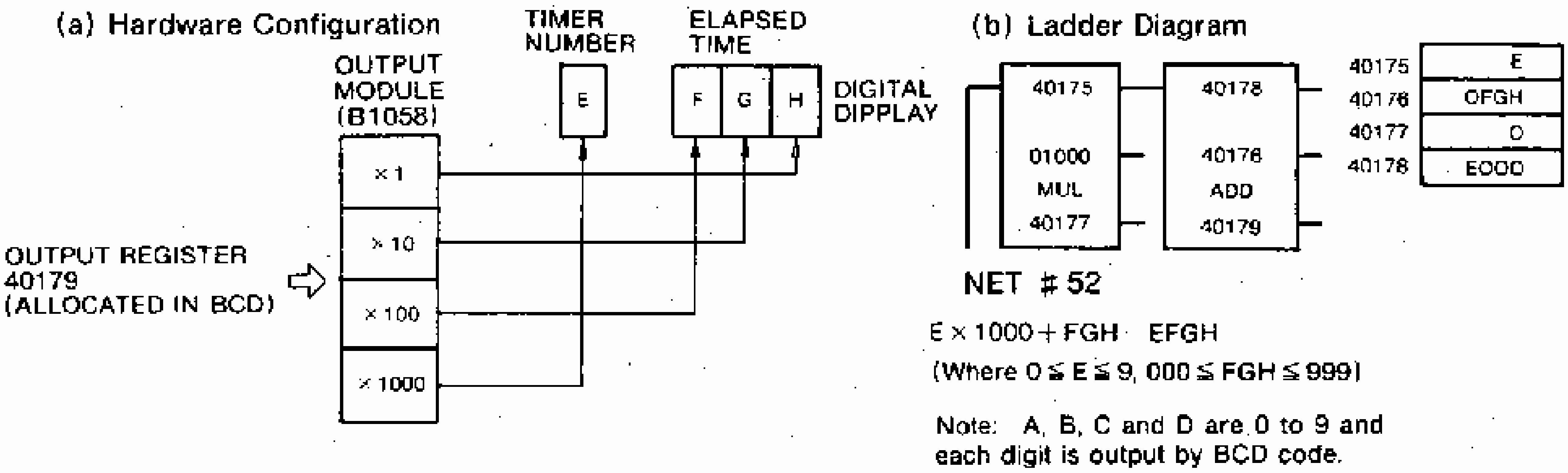


Fig. 4.38 Saving Output Module

4.6 SQUARE ROOT

4.6.1 Types of Square Root

Square root is the function that calculates the square root of the operand V which is a 4- or 8-digit decimal number. Two types of square root are available as shown in Table 4.30.

Table 4.30 Types of Square Root Function

Type	Symbol	Function	Range of Operand V	Reference Page
Square Root	SQRT	Calculation of $\sqrt{V}$	0 to 9999	88
Double-precision Square Root	DSQR	Calculation of $\sqrt{V}$	0 to 99999999	89

4.6.2 Square Root (SQRT)

(1) Form

- Fig. 4.39 shows the form of square root (SQRT).

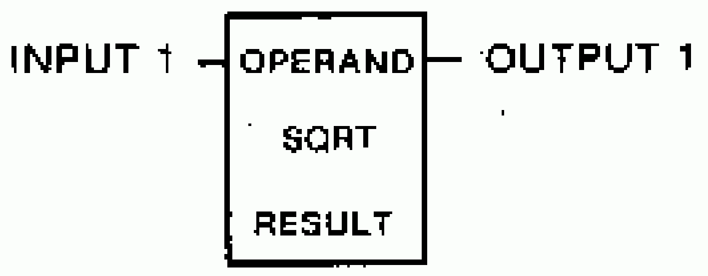


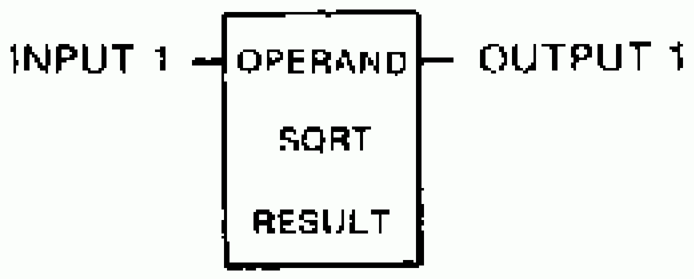
Fig. 4.39 SQRT General Form

- Square root operation requires two elements placed vertically (top and bottom). Referring to Table 4.31, specify either reference number 3×××× of an input register or reference number 4×××× of a holding register for each of the elements.
- SQRT is the symbol denoting the square root.

Table 4.31 Elements of Square Root

Element Position	Specified Number	Description
Top	Either one of the following: • 3×××× (30001–30256) • 4×××× (40001–49999)	The contents of 3×××× and 4×××× are the operand (V= 0 to 9999).
Bottom	4×××× (40001–49999)	The result of square root operation ($\sqrt{V}$) is stored as follows. 4××××: Integer part of $\sqrt{V}$ 4××××+1: Decimal part of $\sqrt{V}$

(2) Square Root Function



By the square root (SQRT), $\sqrt{V}$ is calculated when the input 1 is ON. The result is treated as follows. The integer part of $\sqrt{V}$ is stored in R and the decimal part (rounded off at the fifth decimal place) in R+1. The output 1 is turned ON. The result remains in R and R+1 even after the input 1 is turned from on to off.

(3) Square Root Operation

Table 4.32 shows a square root operation (SQRT).

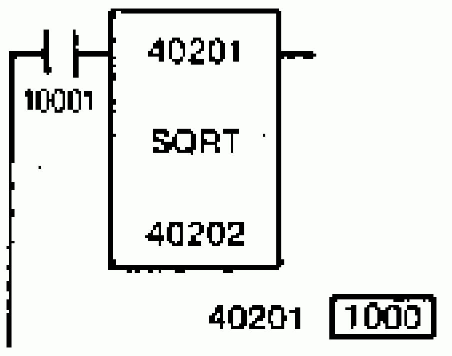
Table 4.32 Square Root Operation

Input 1	Operation	Output 1
ON	$\sqrt{V}$→R (Integer part), R+1 (Decimal part)*	ON
OFF	Not operated.	OFF

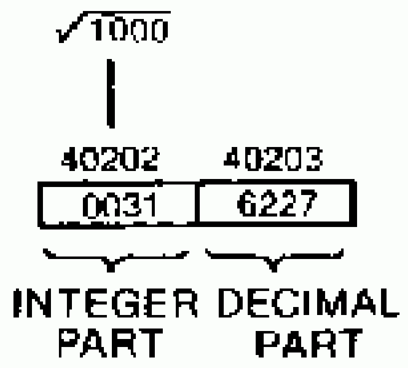
* Rounded off at the 5th decimal place.

(4) Example—Square Root

Example



(a) NET #52



(b) SQRT Operation

SQRT in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 is turned on. The result remains in 40202 and 40203 even after input relay 10001 is turned off.

4.6.3 Double-precision Square Root (DSQR)

(1) Form

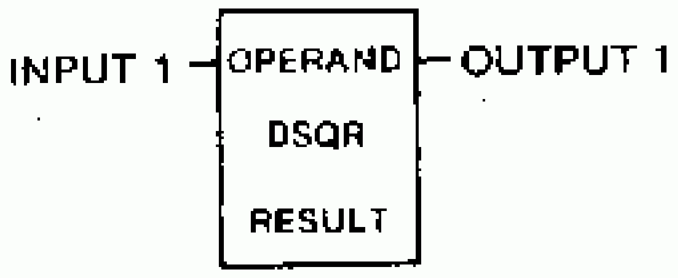


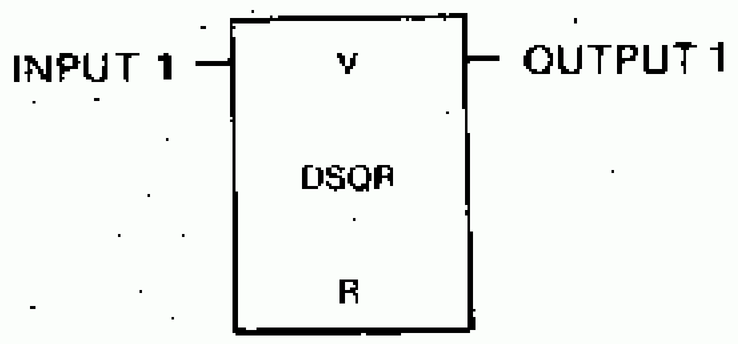
Fig. 4.40 DSQR General Form

- Fig. 4.40 shows the form of double-precision square root (DSQR).
- Double-precision square root operation requires two elements placed vertically (top and bottom). Referring to Table 4.33, specify either reference number 3×××× of an input register or reference number 4×××× of a holding register for each of the elements.
- DSQR is the symbol denoting the double-precision square root.

Table 4.33 Elements of Double-precision Square Root Function

Element Position	Specified Number	Description
Top	Either on of the following: • 3×××× (30001−30255) • 4×××× (40001−49998)	The operand (V = 0 to 99999999) is stored as follows Where 4××××, 4×××× [VH] 4××××+1 [VL] VH : Higher-place 4 decimal digits of V VL : Lower-place 4 decimal didits of V
Bottom	4×××× (40001−49998)	The result of double-precision square root operation ($\sqrt{V}$) is stored as follows. 4×××× [ ] 4××××+1 [ ] Integer part of $\sqrt{V}$ — Decimal part of $\sqrt{V}$

(2) Double-precision Square Root Function



By double-precision square root (DSQR), $\sqrt{V}$ is calculated when the input 1 is ON. The result is treated as follows. The integer part of $\sqrt{V}$ is stored in R and the decimal part (rounded off at the fifth decimal place) in R+1. The output 1 is turned on. The result remains in R and R+1 even after the input 1 is turned from on to off.

(3) Double-precision Square Root Operation

Table 4.34 shows a double-precision square root (DSQR).

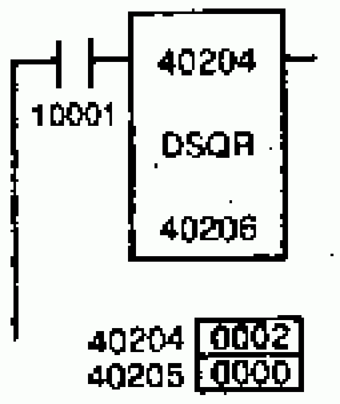
Table 4.34 Double-precision Square Root Operation

Input 1	Operation	Output 1
ON	$\sqrt{V}$→R (Integer part), R+1 (Decimal part)*	ON
OFF	Not operated.	OFF

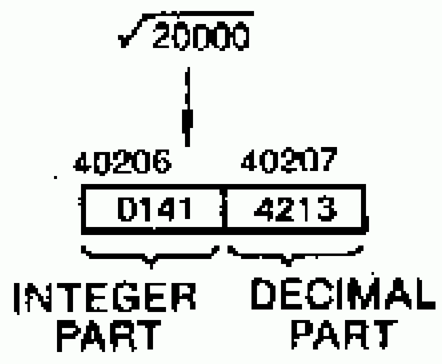
*Rounded off at the 5th decimal place.

(4) Example—Double-precision Square Root

Example



(a) NET #53



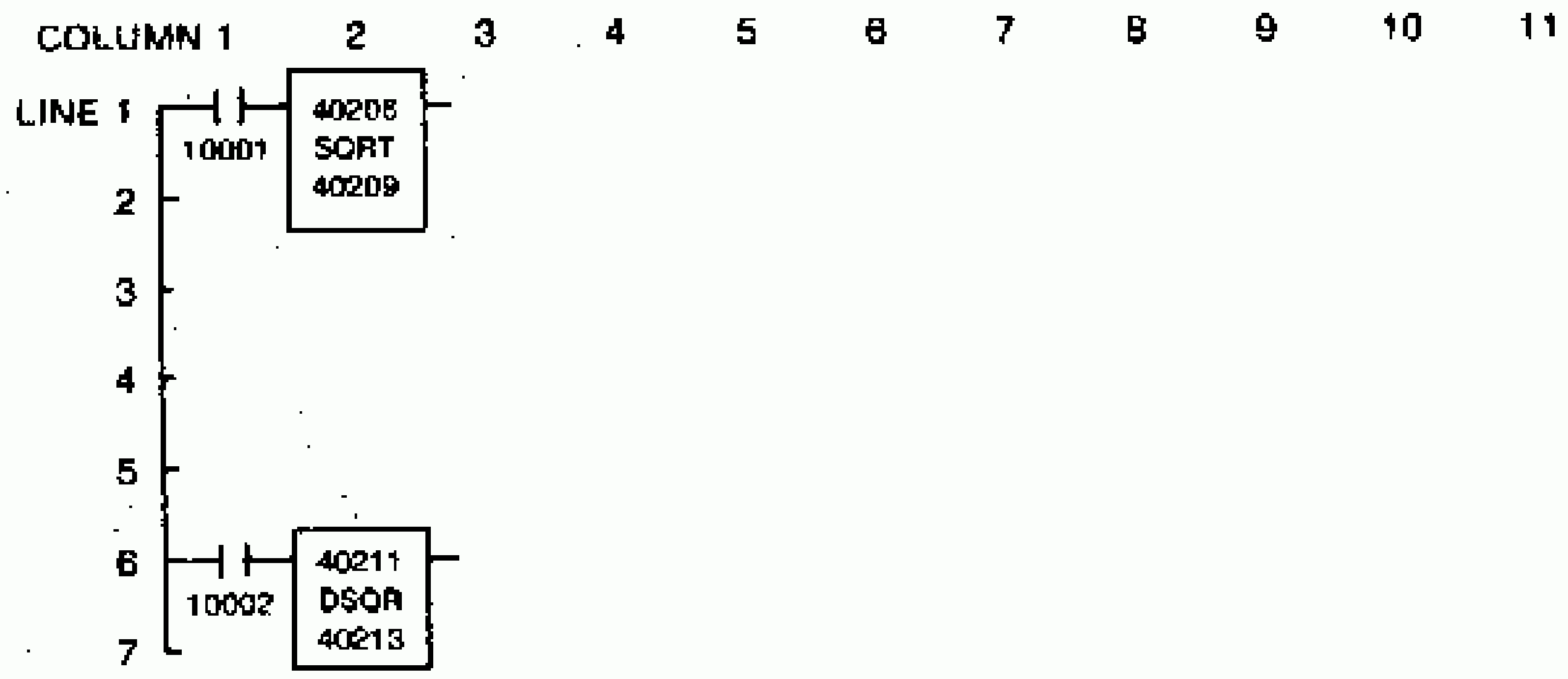
(b) DSQR Operation

DSQR in (a) executes the operation of (b) when input relay 10001 is ON. The output 1 is turned on. The result remains in 40206 and 40207 even after input relay 10001 is turned off.

4.6.4 Programming Square Root Circuit and Precautions

(1) Programming Square Root Circuit

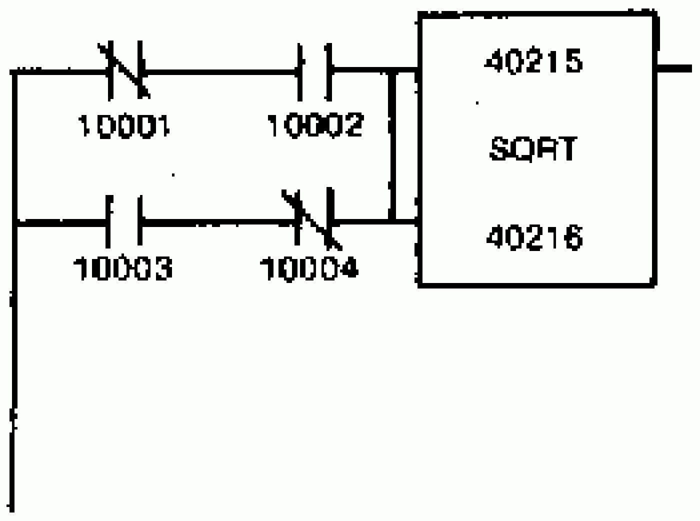
Square root operation requires two elements placed vertically (top and bottom). It can be used at any intersection of the 7 lines-by-10 columns matrix, but the top element (operand) cannot be used on line 7.



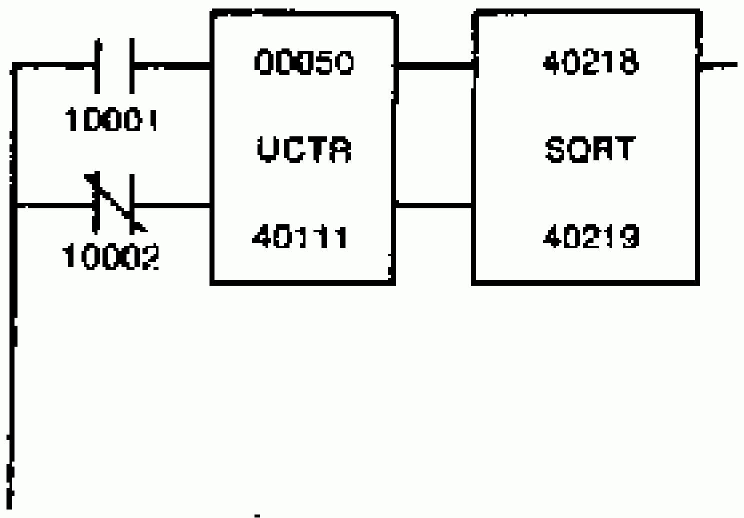
NET #53

(2) Input of Square Root

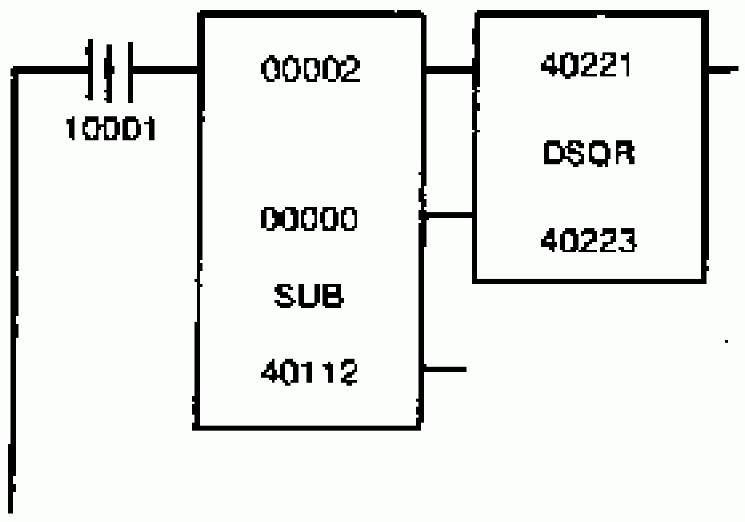
Input to square root may be the output of other square roots, relays, timers, counters, or data processing circuits.



Example 1



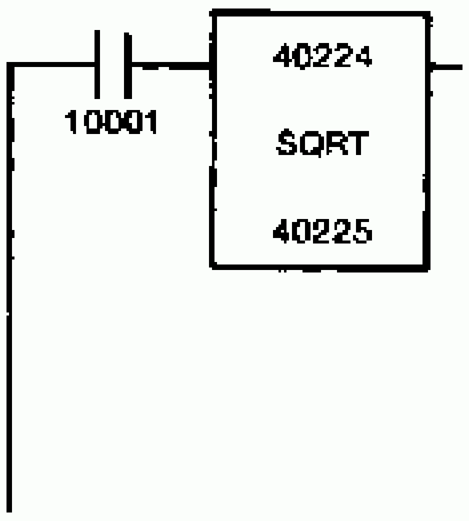
Example 2



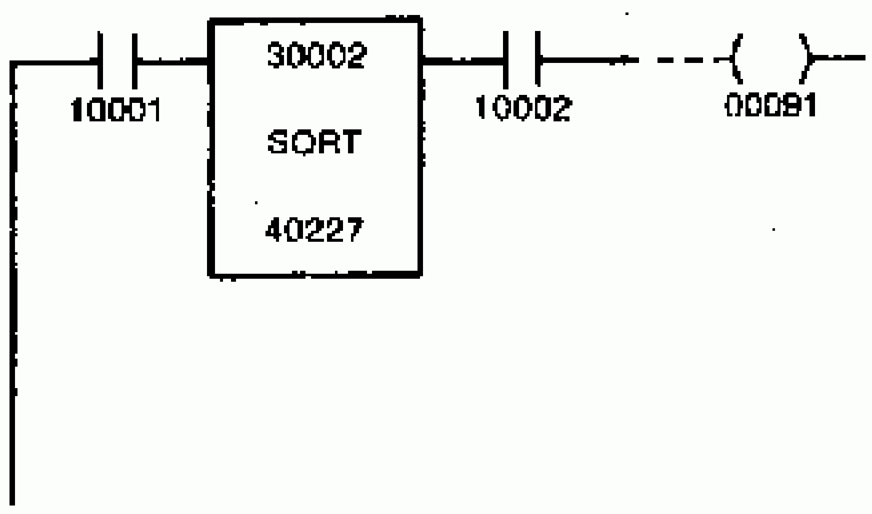
Example 3

(3) Outputs of Square Root

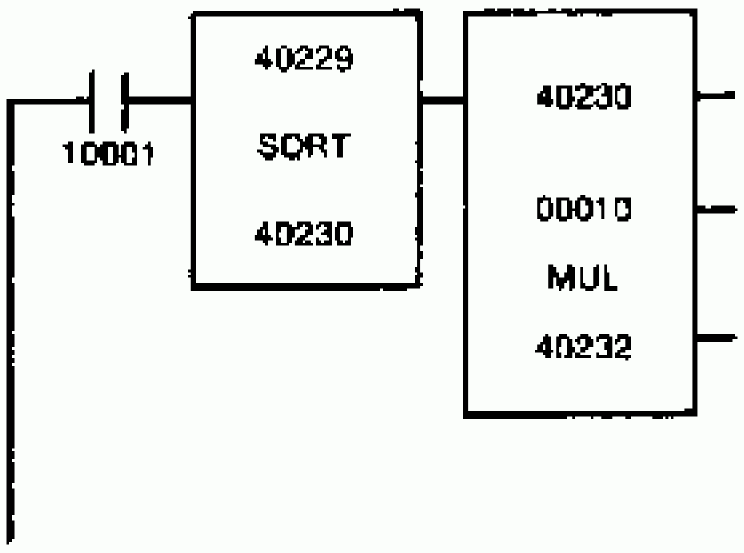
Coils need not be connected to two output nodes (1 and 2) of a square root. It is permitted to connect a relay contact to the output node at right or connect the output nodes directly to an input node of an arithmetic circuit, except relays.



Example 1



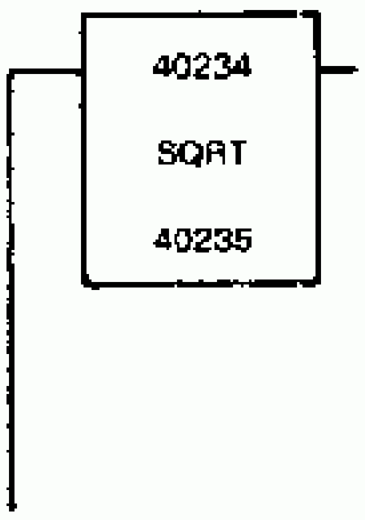
Example 2



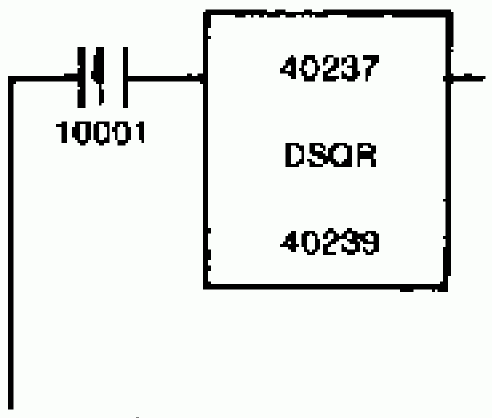
Example 3

(4) Execution of Square Root

To execute a square root constantly, connect the inputs directly to the power rail at left. To execute it only during one scanning cycle, use a transitional contact as an input.



Example 1



Example 2

4. 7 DATA MOVE

The previous section described the basic logic functions available with all U84 Controllers. These basic functions include the ability to simulate the operation of electro-mechanical relays, timers and counters, as well as to perform arithmetic functions (addition, subtraction, multiplication, and division), and square root functions.

This section discusses more advanced logic functions that are available with the U84 Controller. These optional functions greatly increase the power and sophistication of any control system designed around the U84 Controller. This advanced power allows the logic functions to be performed using significantly less memory.

4. 7. 1 Types of Data Move

Eight types of data move are available, as shown in Table 4.35.

Table 4. 35 Types of Data Move

Type	Symbol	Reference Page
Registor-To-Table Move	R→T	96
Table-To-Registor Move	T→R	100
Table-To-Table Move	T→T	101
Block Move	BLKM	102
First In	FIN	103
First Out	FOUT	104
Table Serch	SRCH	105
Get Controller System Status	STAT	108

4. 7. 2 Form of Data Move

Fig. 4.41 shows the form of data move.

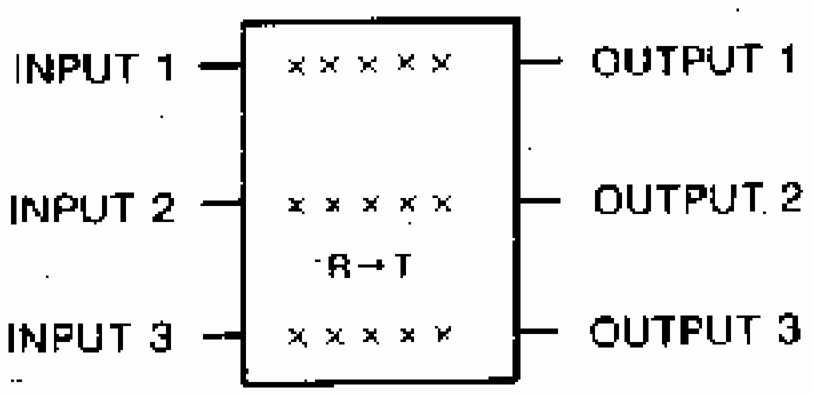


Fig. 4. 41 Move General Form

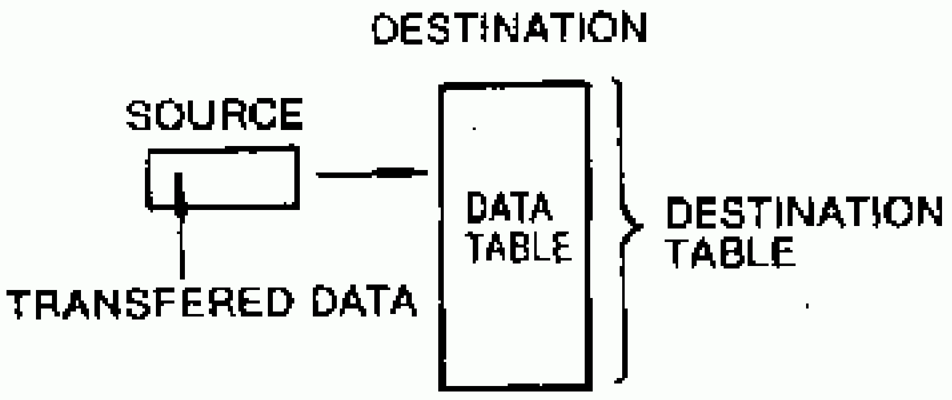
Like the arithmetic operations, data move operation requires three elements placed vertically (top, middle, and bottom). It can be used at any intersection of the 7 lines-by-10 columns matrix, but the top element cannot be used on lines 6 and 7. In Fig. 4.41, R → T indicates the type of data move.

(1) Elements and Their Meanings

- The top element is called source (of data) and the middle one destination (of data move). Generally, data is moved from the source to the destination. The bottom element denotes the size of the data table (table size) of the source, destination, or both.

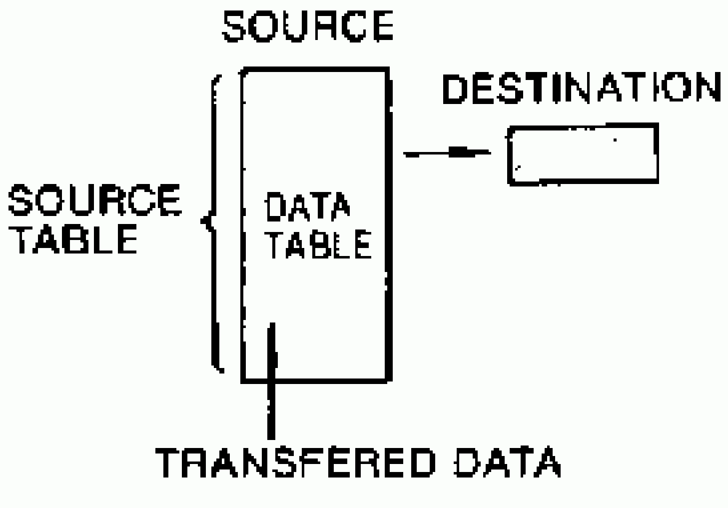
- There are three types of data move.

(a) Moving data from a source to a specified destination in a destination table (1:N move)



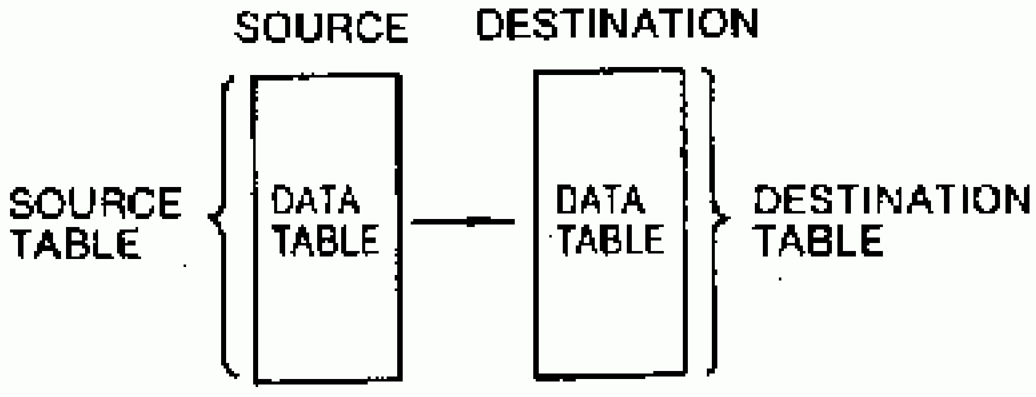
1 : N TYPE

(b) Moving data from a specified source in a source table to a destination (N:1 move)



N : 1 TYPE

(c) Moving data from the source table to the destination table (N:N move)

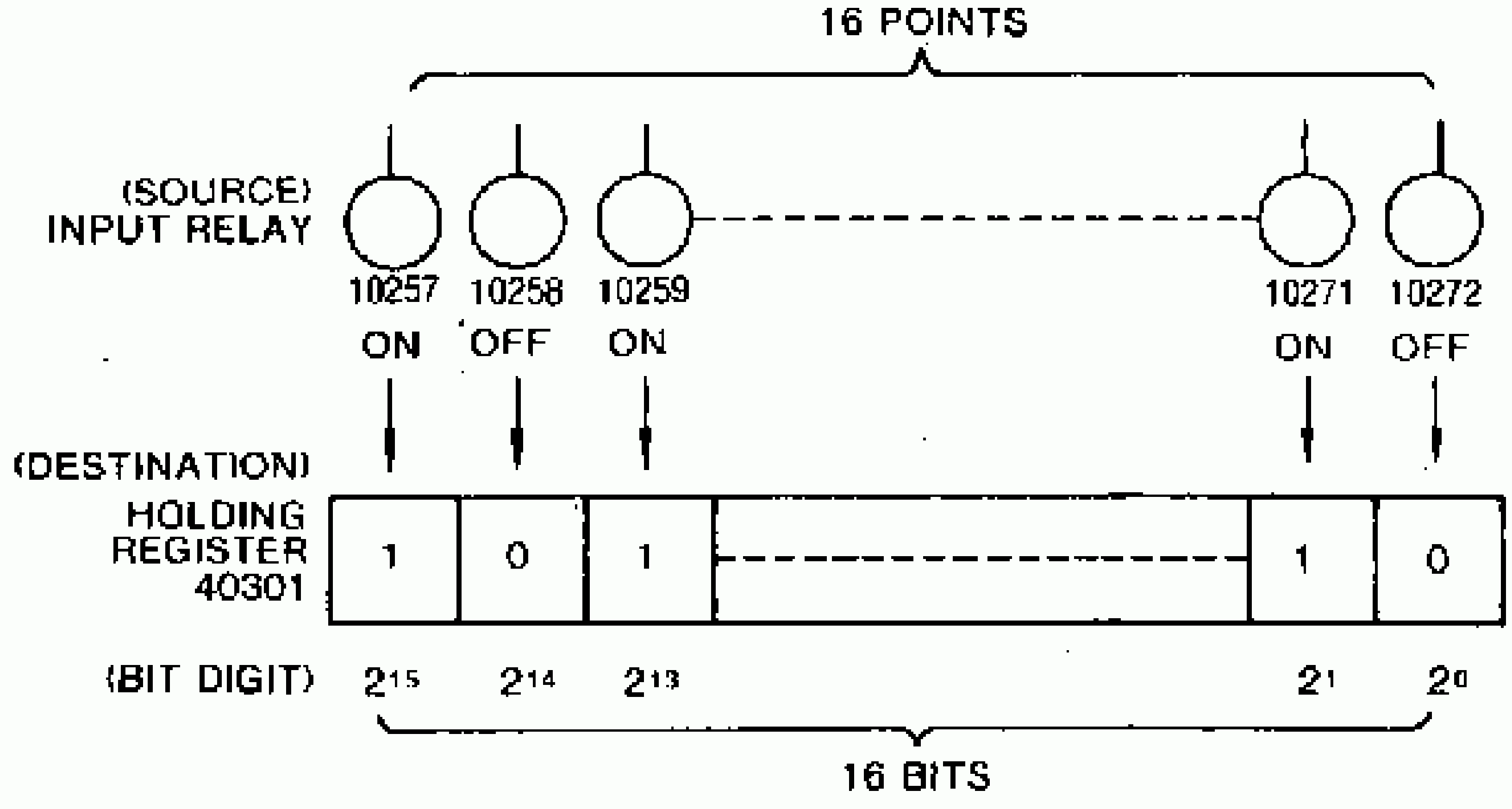


N : N TYPE

(2) Reference Numbers of Source and Destination (Source Reference and Destination Reference)

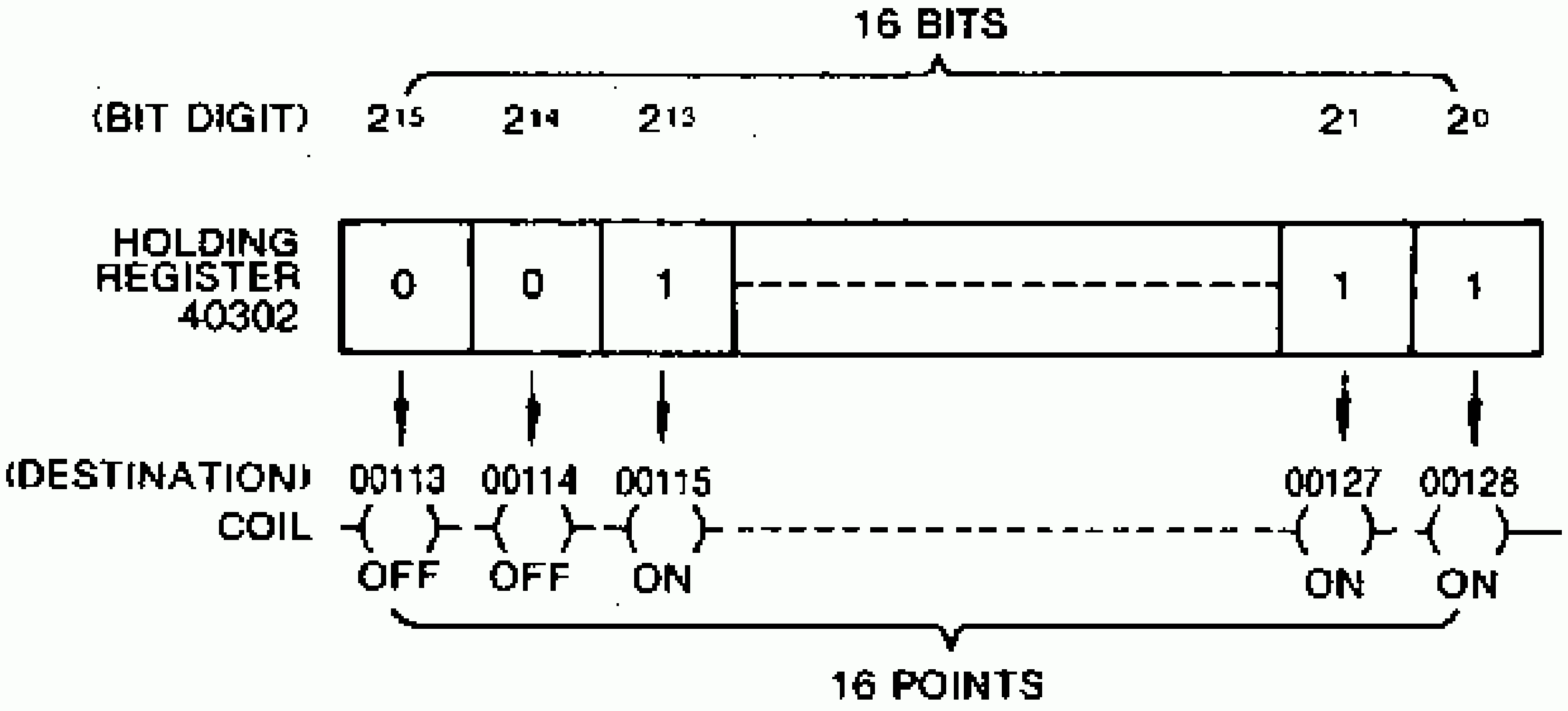
- The numbers specifiable as the reference numbers of the source and destination depend on the type of data move. See Table 4.36.
- When a register is specified, the contents of the register is moved as a value of 0-9999 or a 16-bit binary pattern ("1" and "0"). When coils or input relays are specified, 16 points of them are treated like a register with their ON and OFF status regarded as "1" and "0".

Example 1: Assume the source is 16 input relays 10257-10272 and the destination is 40301. Then the ON/OFF status of the input relays enters the corresponding bits of 40301, where ON is represented with "1" and OFF with "0".



(2) Reference Numbers of Source and Destination (Cont'd)

Example 2: Assume the source is 40302 and the destination is coils 00113-00128. Then "1" or "2" status of each coil of 40302 is set to each of 16 coils (113 to 128) corresponding to the coil, with ON/OFF status.



Note
1. When an input relay or a coil number is specified as a reference number, n=16 m+1 (m=0, 1, 2, . . .) is required where n is XXXX of 1XXXX or 0XXXX.
   Example: 10001 and 00049 are correct, but 10100 and 00002 are wrong.
2. Sixteen coils or input relays are treated as a group in data move even if some of them are not used.
3. When the destination is a group of coils, the number of the coil group must not be the same as one used in another circuit. The coil 02049 is used as a battery monitor and therefore coils 02049 to 02064 can not be included in the destination table.

(3) Table Size

- The table size is specified by a constant XXX. The range of the table size depends on the type of move. See Table 4.36.
- The table size is counted in units of register numbers for the register table and in units of 16 points for the discrete table.

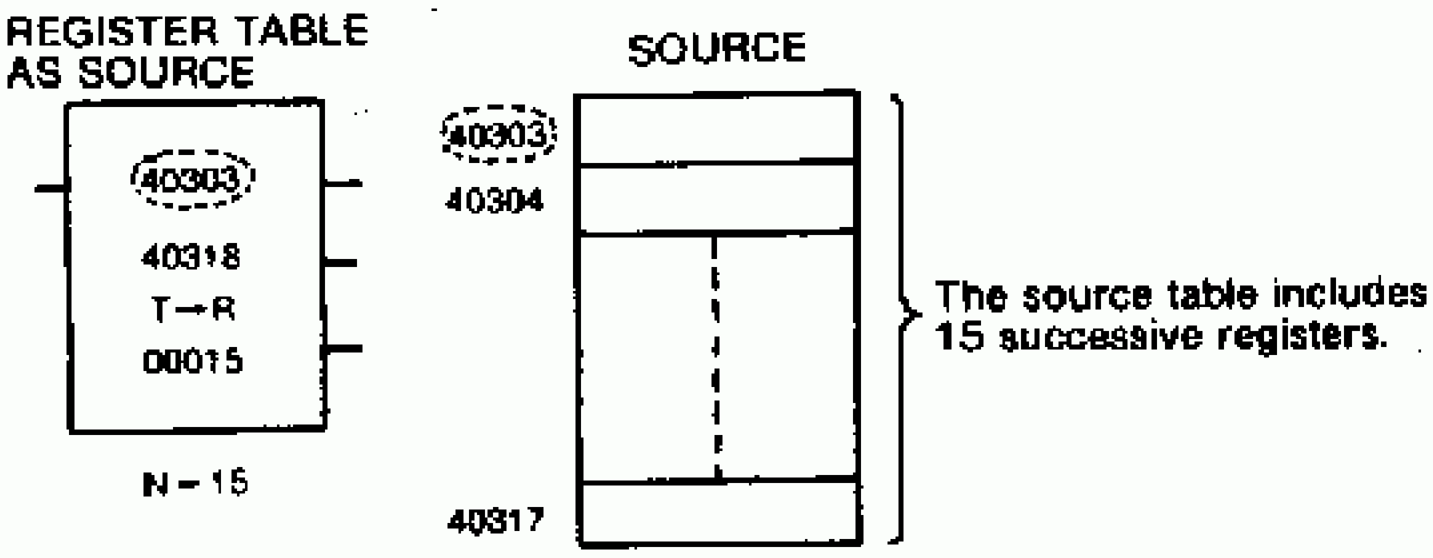
Example:

Data Table	Table Size
30005-30016	12
10001-10256	16

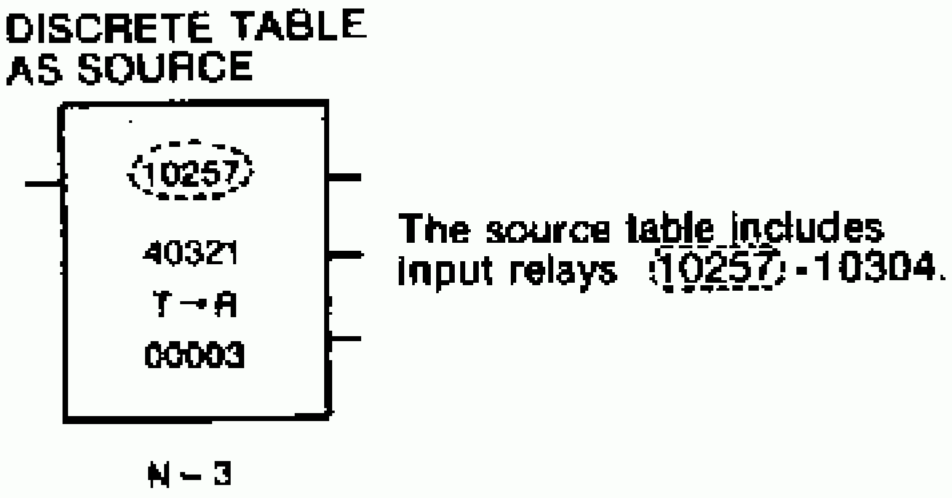
- Relationship of the reference numbers of the top and middle elements and table size N is as follows.

(a) When the source is a table (T → R, SRCH, FOUT):

The first (smallest) number of table is specified at the top as the reference number. The source table consists of N successive registers or N successive sets of discrete points (16 × N).



Example 1

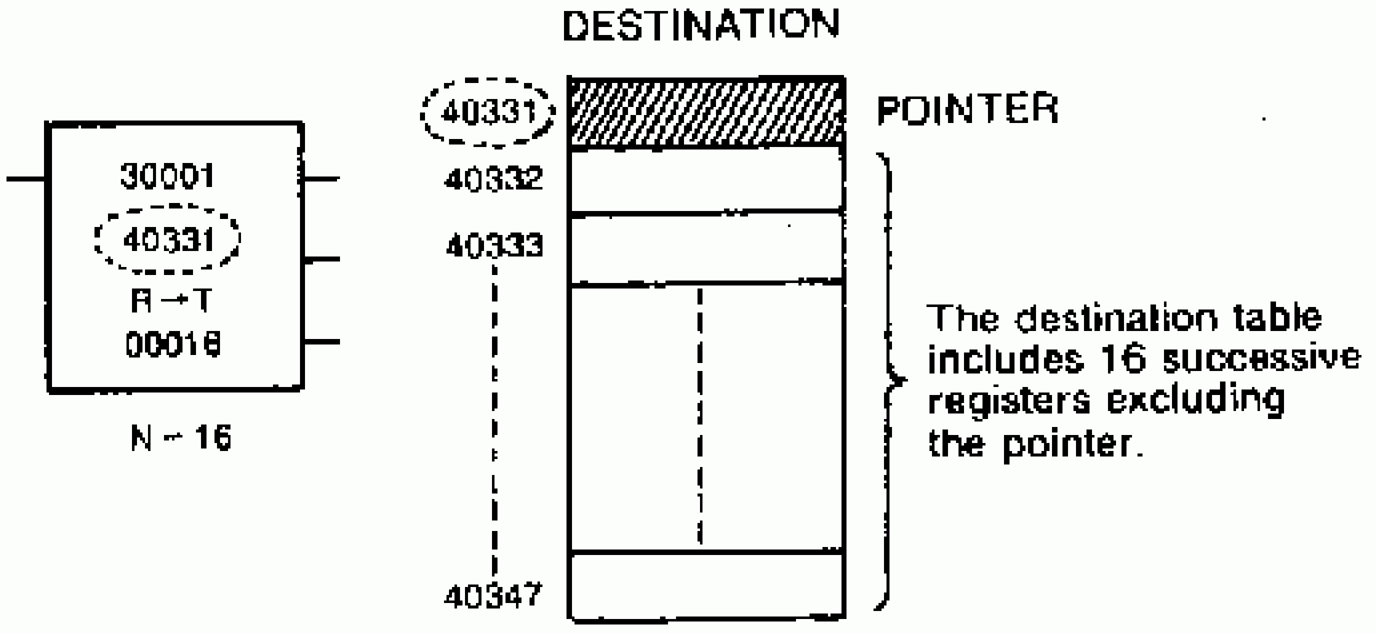


Example 2

(b) When the destination is a table (R → T, FIN):

The number of the holding register for a pointer is specified at the middle as the reference number. The destination table consists of N successive holding registers that begin from next to the pointer.

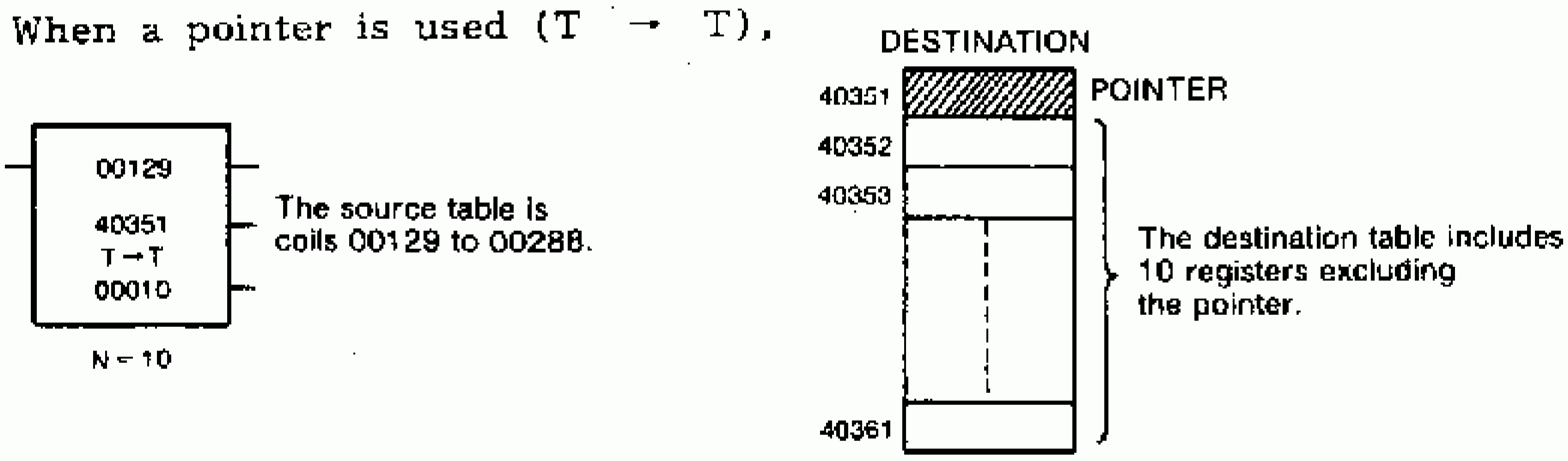
Example



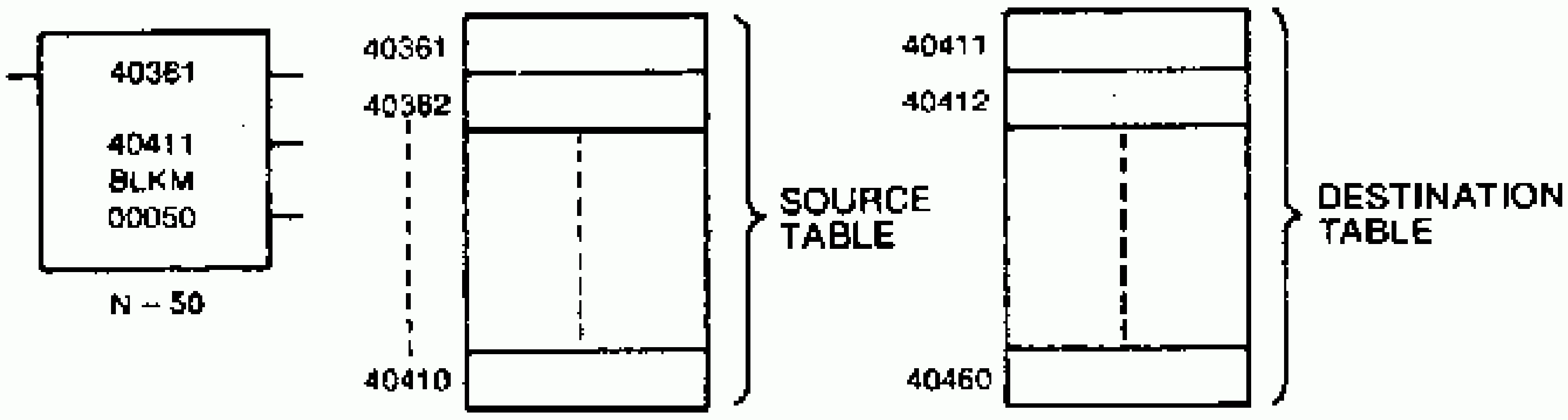
(c) When the source and the destination are tables (T → T, BLKM, STAT):

The table size of both of the source and the destination tables is N.

Example 1: When a pointer is used (T → T),



Example 2: When a pointer is not used (BLKM),



Note: For status read out (STAT), the source need not be specified, because it is fixed by a special memory in the U84 syetem.

(4) Pointer

- A pointer is used, except in block move (BLKM) and status read out (STAT). A holding register is used for the pointer and its contents specify the register (or a set of discrete points) to be operated in the source or destination table.
- The value of the pointer is 0, 1, 2, ..., N (N is a table size). The value plus the starting number of the table denotes the register number (or a set of 16 discrete points) to be moved.
- Ordinarily, the value of the pointer is the number already executed in the table. If the value is 20, for example, the 20th item in the table has been already executed and the 21st item is used for the next execution (except with FOUT). If the value of the pointer is 0, the execution will start from the top of the table.

Generally, the content of the pointer is incremented (decremented for FOUT) by one each time a move execution has been accomplished. It is possible to prohibit the automatic increment of the pointer.

(4) Pointer (Cont'd)

The value of the pointer can be forcibly altered by some functions other than relay. When the 50th item is required, regardless of the content of the pointer its content should be forcibly set to 49 by using arithmetic operations. As a result, it is possible to use the 50th item. The pointer preserves its value unless it is set externally.

4.7.3 Function and Operation of Data Move

Table 4.36 lists functions of data move.

(1) Register-to-Table Move (R → T)

(a) Form and Function

This is a move from a source to a destination table (1:N move). Figs. 4.42 and 4.43 show the form and function of the R → T.

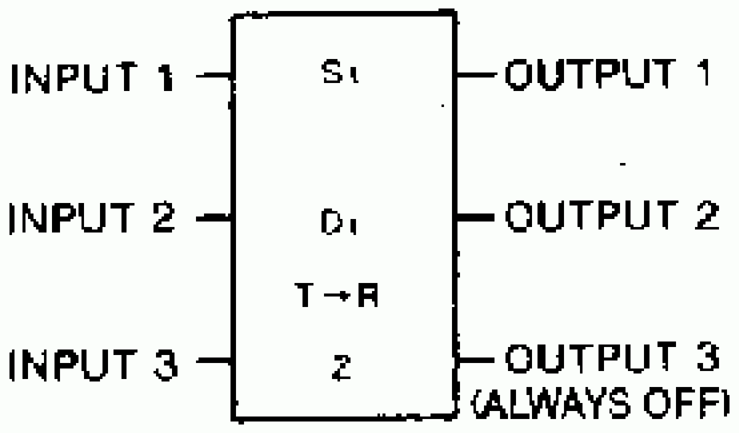


Fig. 4.42 R→T General Form

S1	Source reference	See Table 4.36
D1	Destination reference (pointer)	
Z	Destination table size	
R→T	Symbol denoting register→table	—

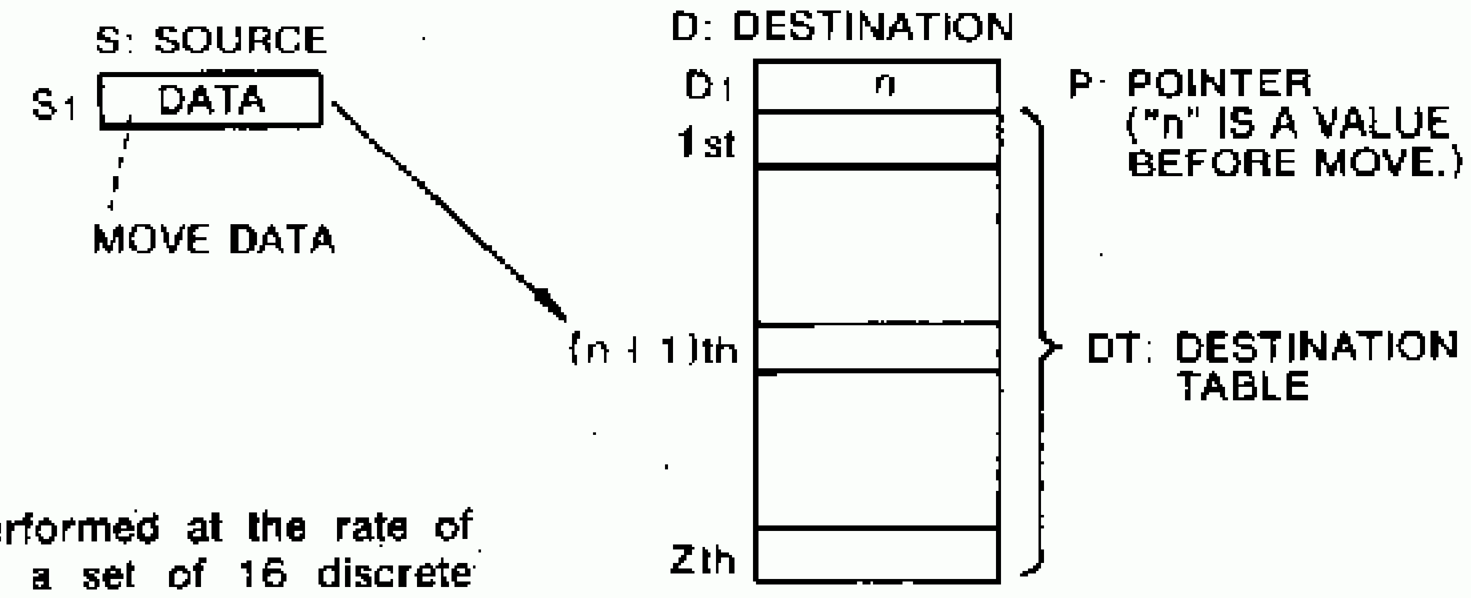


Fig. 4.43 R→T Function

Note:
Move will be performed at the rate of one register (or a set of 16 discrete points) per scanning cycle.

(b) Inputs

All three inputs are used with the R → T function. The input 1 controls the move. Every scan power flow is received at this node, the move is performed and the pointer incremented. Both the move and pointer incrementing occur during the solution of this Function Block. Incrementing the pointer will cause moves on future scans to occur into successive register locations. The pointer can not exceed the table length. Thus when the pointer is equal to the table length, it will stop incrementing and the move will stop operating. A transitional contact can be used to control the input 1 if a single move operation is desired.

The input 2, when receiving power flow, inhibits the incrementing of the pointer. Thus moves with power flow at both inputs 1 and 2 can be made continuously to the same register in the table, until either another function increments the pointer or the input 2 loses power flow. The input 3 resets the pointer to zero. Whenever this input receives power flow, the pointer is reset to zero regardless of its current value.

The bottom input, when energized, will cause the R→T move to go to the first register in the table if the top element is also energized.

(c) Outputs

The register-to-table function utilizes only the first two outputs. The output 3 has no significance and will be OFF (no power flow) under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever the pointer is equal to the table length, when the move function has reached the end of the table.

(d) Example

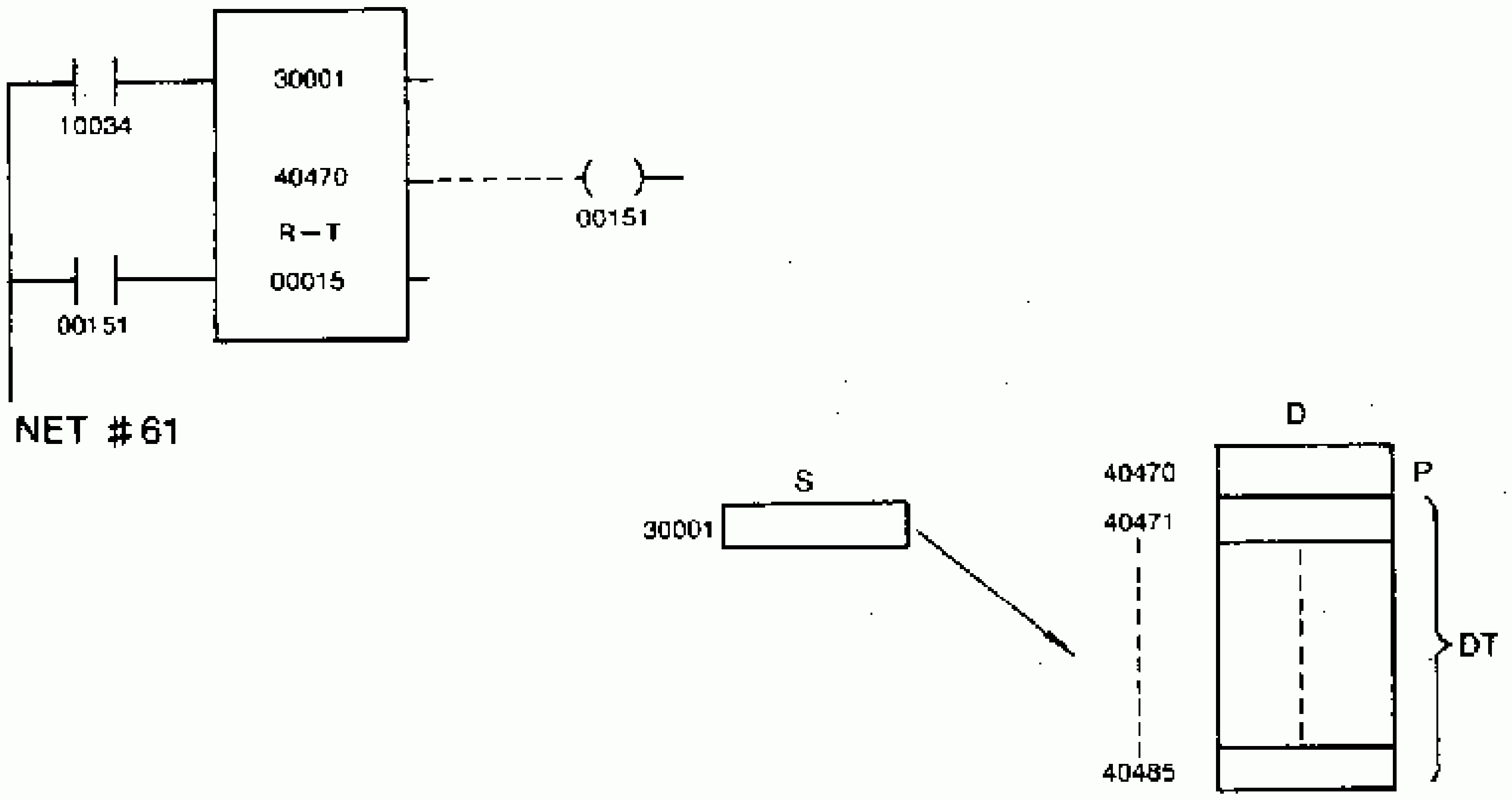
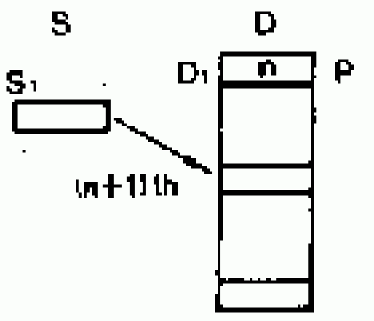
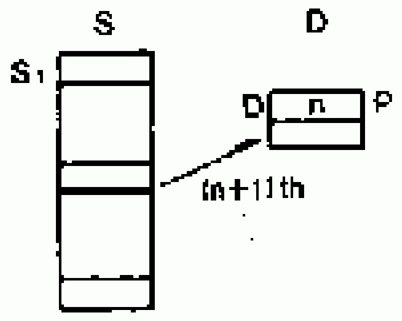
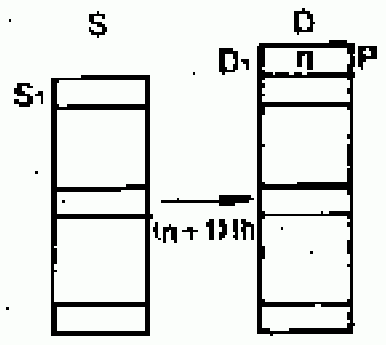
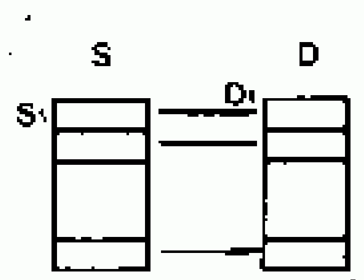


Figure above illustrates a typical Register-to-Table move function. As long as input 10034 is energized, the content of input register 30001 is moved into the table 40471-40485 at the rate of one entry per scan. The pointer is also automatically incremented. When the pointer reaches the value 15, the output 2 supplies power flow and energizes coil 00151. On the next scan, The input 3 will be energized since it is referenced to coil 00151. This input causes the pointer to be reset to zero, thus de-energizing coil 00151.

If the pointer is at the value eleven when input 10034 is energized, register 40482 receives the content of register 30001 and pointer goes to the value twelve.

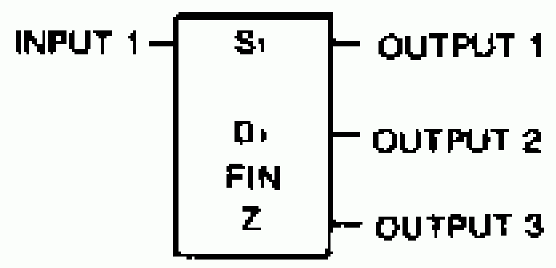
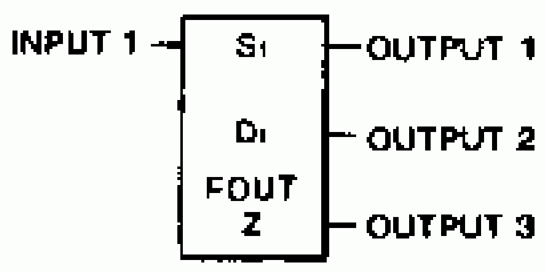
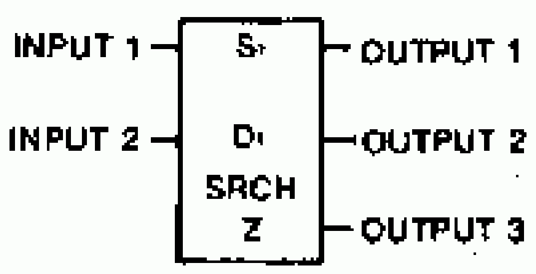
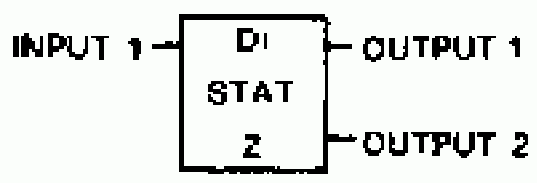
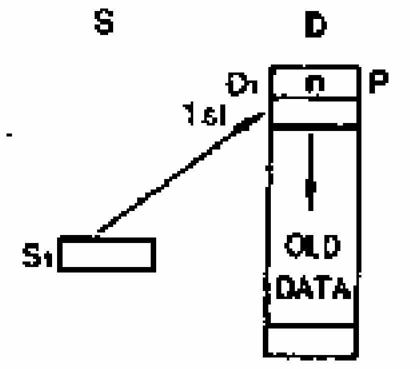
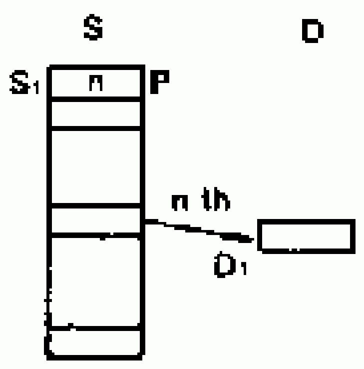
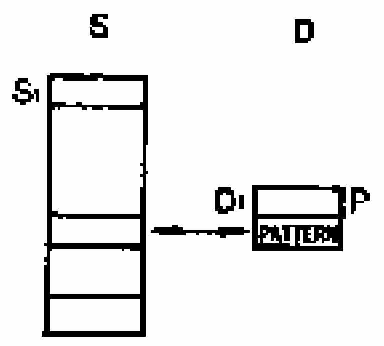
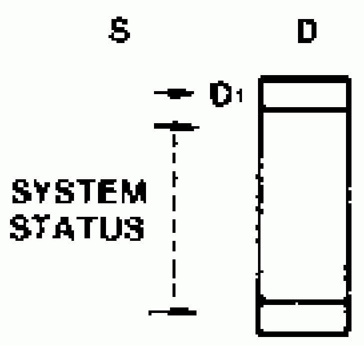
4.7.3 Function and Operation of Data Move (Cont'd)

Table 4.36 List of Data Move Function (Cont'd)

Item \ Function	Register-to-Table Move	Table-to-Register Move	Table-to-Table Move	Block Move
Reference Page	96	100	101	102
Form	INPUT 1 — S1 — OUTPUT 1 INPUT 2 — D1 R→T — OUTPUT 2 INPUT 3 — Z — OUTPUT 3	INPUT 1 — S1 — OUTPUT 1 INPUT 2 — D1 T→R — OUTPUT 2 INPUT 3 — Z — OUTPUT 3	INPUT 1 — S1 — OUTPUT 1 INPUT 2 — D1 T→T — OUTPUT 2 INPUT 3 — Z — OUTPUT 3	INPUT 1 — S1 — OUTPUT 1 D1 BLKM — OUTPUT 2 Z — OUTPUT 3
Explanation of Function S: Source D: Destination P: Pointer n: Value of Pointer before Move	 Stores the contents of a register (or 16 discrete points) in one scan.	 Reads the cotenls of a register (or 16 discrete points) in one scan.	 Moves the contents of a register (or 16 discrete points) in one scan.	 Moves all source data in one scan.
S_1: Source Reference	• 0××××(00001, 00017, ········, 04081) or 1××××(10001, 10017, ········, 12033) or 3××××(30001, 30002, ········, 30256) or 4××××(40001, 40002, ········, 49999) • n = 16m + 1 (m = 0, 1, ······) where n is ×××× of 0×××× or 1××××.			
D_1: Destination Reference	• 4××××(40001–49998) • Becomes pointer.			Any of following: • 4××××(40001–49999) • 0××××(00001, 00017, ······, 04081)*
Z: Table Size	1–999	1–999 (S_1 = 3××××, 4××××) 1–256 (S_1 = 0××××) 1–128 (S_1 = 1××××)		1–100
Input 1	Executes the move and increments pointer by 1.			Executes the move.
Input 2	Prohibits the changes of pointer.			Not used.
Input 3	Pointer becomes 0 (regardless of ON/OFF status of input 1).			Not used.
Output 1	Same as ON/OFF status of input 1.			
Output 2	ON when pointer = table size (regardless of ON/OFF status of input 1).			Always OFF
Output 3	Always OFF			
Cases where move is not executed even when input is ON.	Pointer ≧ Table size			—

*n = 16m + 1 (m = 0, 1, 2, ·····) where n is ×××× of 0××××.
Coil: 02049 is used as a battery monitor coil, therefore the destination table cannot include the coil group 02049—02064.

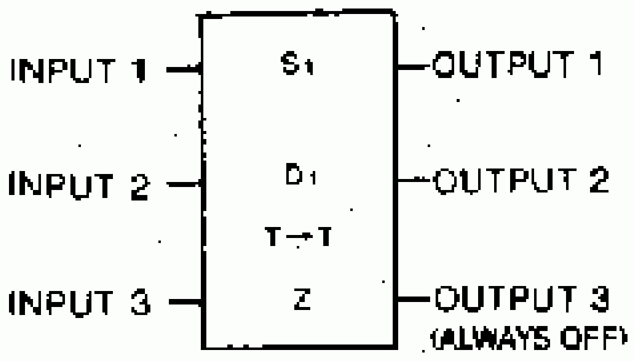
Note: All functions occur in one scan.

	First In	First Out	Table Search	Get Controller System Status
	103	104	105	108
				
	 First-in of the contents of a register (or 16 discrete points) in one scan.	 First-out of the contents of a register in one scan.	 Searches one matching in one scan.	 Moves all statuses in one scan.
		• 4XXXX (40001–49998) • Becomes pointer.	Any of following: • 3XXXX (30001–30256) • 4XXXX (40001–49999)	—
	• 4XXXX (40001–49998) • Becomes pointer.	Any of following: • 4XXXX (40001–49999) • 0XXXX (00001, 00017, ……, 04081)*	• 4XXXX (40001–49998) • Becomes pointer.	Any of following: • 4XXXX (40001–49999) • 0XXXX (00001, 00017, ……, 04081)*
	1–100	1–100	1–100	1–21
	Executes the move and increments pointer by 1.	Executes the move and decrements pointer by 1.	Searches from first source register.	Executes the move.
	Not used.		Searches source register next to the one indicated by the pointer.	Not used.
	Not used.			—
	Same as ON/OFF status of input 1			
	ON when pointer = table size (table full) (regardless of ON/OFF status of input 1).		ON when same pattern is found.	Always OFF
	ON when pointer = 0 (table empty) (regardless of ON/OFF status of input 1).		Always OFF	—
	Pointer ≧ Table size	Pointer = 0	—	

(2) Table-to-Register Move (T→R)

(a) Form and Function

This is the opposite function to the R → T, namely, a move from a source table to a destination (N:1 move). Figs. 4.44 and 4.45 show the form and function of the T → R.



S1	Source reference	See Table 4.36
D1	Destination reference(pointer)	See Table 4.36
Z	Source table size	See Table 4.36
T→T	Symbol denoting table→register	—

Fig. 4.44 T→R General Form

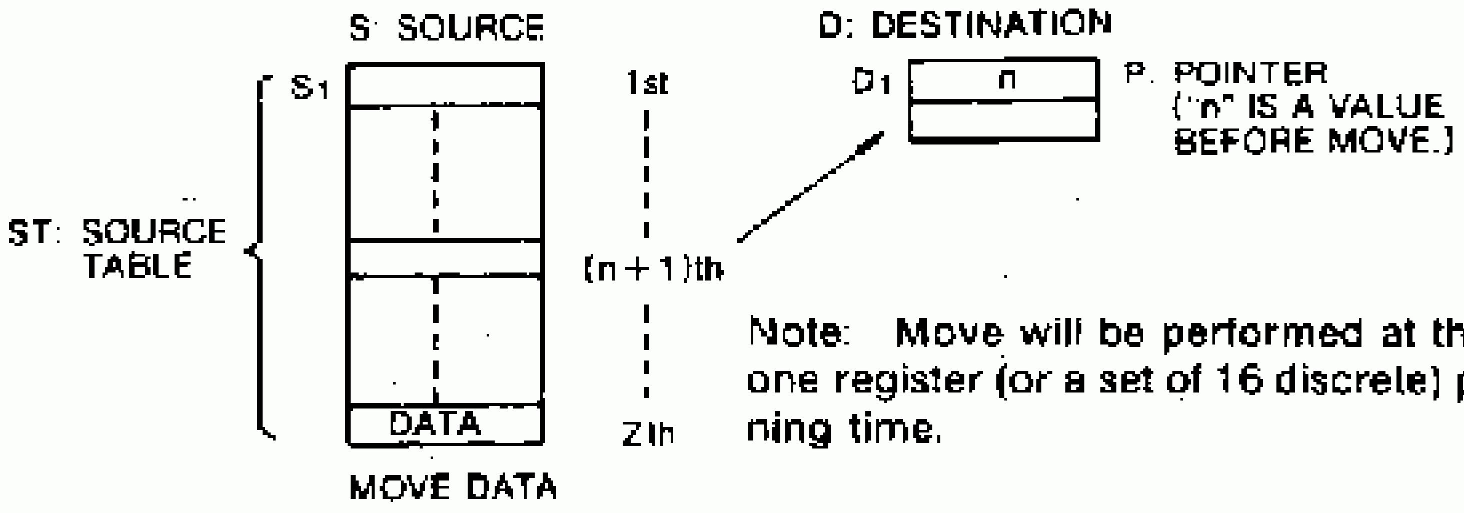


Fig. 4.45 T→R Function

Note: Move will be performed at the rate of one register (or a set of 16 discrete) per scanning time.

(b) Input

All three inputs are used with the T → R function. The input 1 controls the move. Every scan power flow is received at this node, the move is performed and the pointer incremented. Both the move and pointer incrementing occur during the solution of this Function Block. Incrementing the pointer will cause moves on future scans to occur from successive register locations. The pointer can not exceed the table length. Thus when the pointer is equal to the table length, it will stop incrementing and the move will stop operating. A transitional contact can be used to control the input 1 if a single move operation is desired.

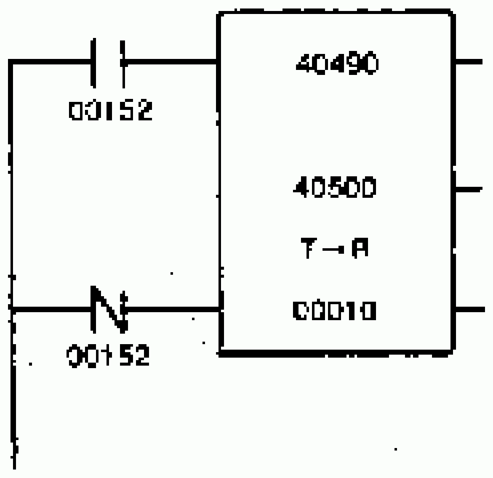
The input 2, when receving power flow; prohibits the incrementing of the pointer. Thus moves with power flow at both inputs 1 and 2 can be made continuously from the same register in the table until either another function increments the pointer or the input 1 loses power flow.

The input 3 resets the pointer to zero. Whenever this input receives power flow, the pointer is reset to zero regardless of its current value.
The input 1, when energized at the same time as the input 3 will cause the first element in the table to be moved into the destination register.

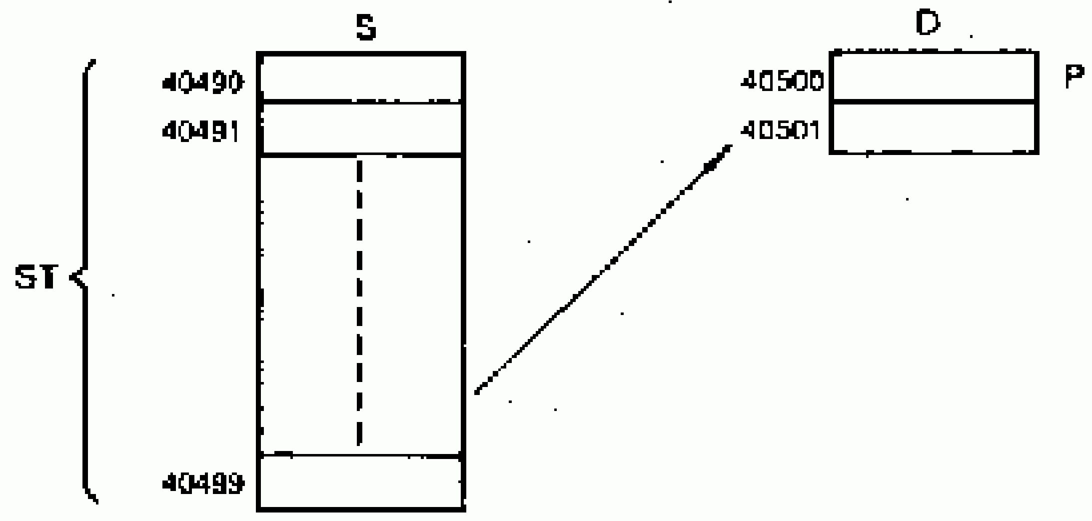
(c) Outputs

The table-to-register function utilizes only the first two outputs. The output 3 has no significance and will be OFF (no power flow) under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever the pointer is equal to the table length, when the move function has reached the end of the table.

(d) Example



NET #62

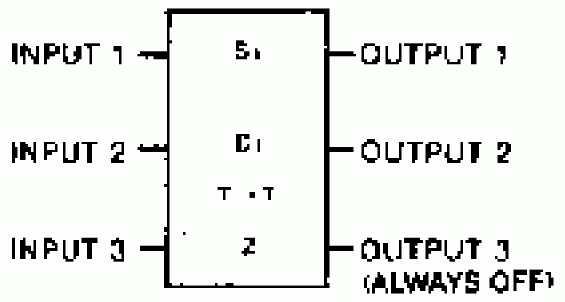


When coil 00152 is OFF, the input 3 receives power flow and the content of pointer becomes 0. Therefore the move is started from 40490 whenever the coil is ON.

(3) Table-to-Table Move (T→T)

(a) Form and Function

This is a move function from a source table to a destination table (N:N move). Figs. 4.46 and 4.47 show the form and function of the T→ T.



S1	Source reference	See Table 4.36
D1	Destination reference(pointer)	
Z	Source destination table sizes	
T→T	Symbol denoting table→table	–

Fig. 4.46 T→T General Form

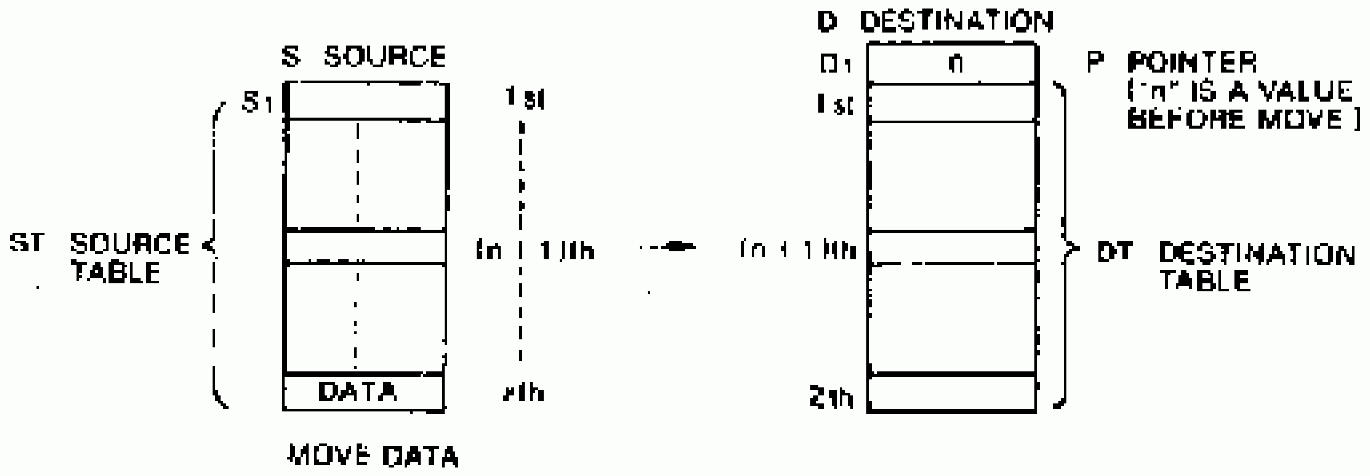


Fig. 4.47 T→T Function

NOTE

Move will be performed at the rate of one register (or a set of 16 discrete) per scanning cycle.

(b) Inputs

All three inputs are used with the T→T function. The input 1 controls the move. Every scan power flow is received at this node, the move is performed and the pointer incremented. Both the move and pointer incrementing occur during the solution of this Function Block. Incrementing the pointer will cause moves on future scans to occur at successive register locations. The pointer can not exceed the table length. Thus when the pointer is equal to the table length, it will stop incrementing and the move will stop operating.

A transitional contact will be used to control the input 1 if the single move operation is desired.

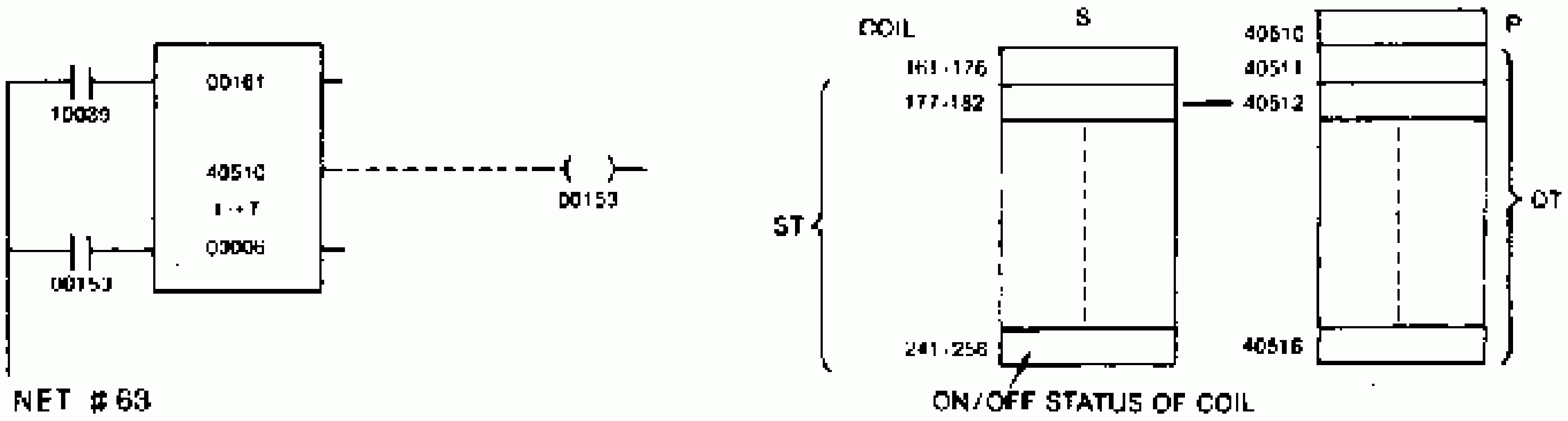
The input 2, when receiving power flow, prohibits the incrementing of the pointer. Thus moves with power flow at both inputs 1 and 2 can be made continuously between the same registers in the tables, until another function increments the pointer or the input 2 loses power flow. The output 3 can reset the pointer to zero. Whenever this input receives power flow, the pointer is reset to zero regardless of its current value.

The input 1, when energized at the same time as the input 3, will cause the first element in the source table to be copied to the first element in the destination table.

(c) Outputs

The table-to-table function utilizes only the first two outputs. The output 3 has no significance and will be OFF (no power flow) under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chanined horizontally within a network. The output 2 supplies power flow whenever the pointer is equal to the table length and indicates when the move function has reached the end of the table.

(d) Example



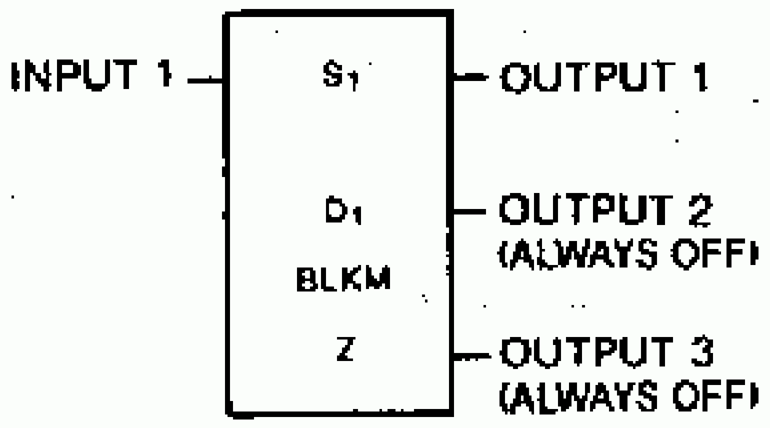
As long as input relay 10039 is energized, the ON/OFF status of coils 00161-00256 will be moved to 40511 and up at the rate of 16 points per scanning cycle. Coil 00153 is turned on in the scanning cycle of move to 40516. On the next scan, the move is started from the first one.

(4) Block Move (BLKM)

(a) Form and Function

This is another type of move from a source table to a destination table (N:N move). This function causes an entire table of registers to be copied into another table in one scan. The source is not altered, only copied. Since the BLKM function moves the entire table in one scan, it does not use a pointer register. Figs. 4.48 and 4.49 show the form and function of BLKM.

Fig. 4.48 BLKM General Form



S_1	Source reference	See Table 4.36.
D_1	Destination reference	
Z	Source, destination table sizes	
BLKM	Symbol denoting block move	—

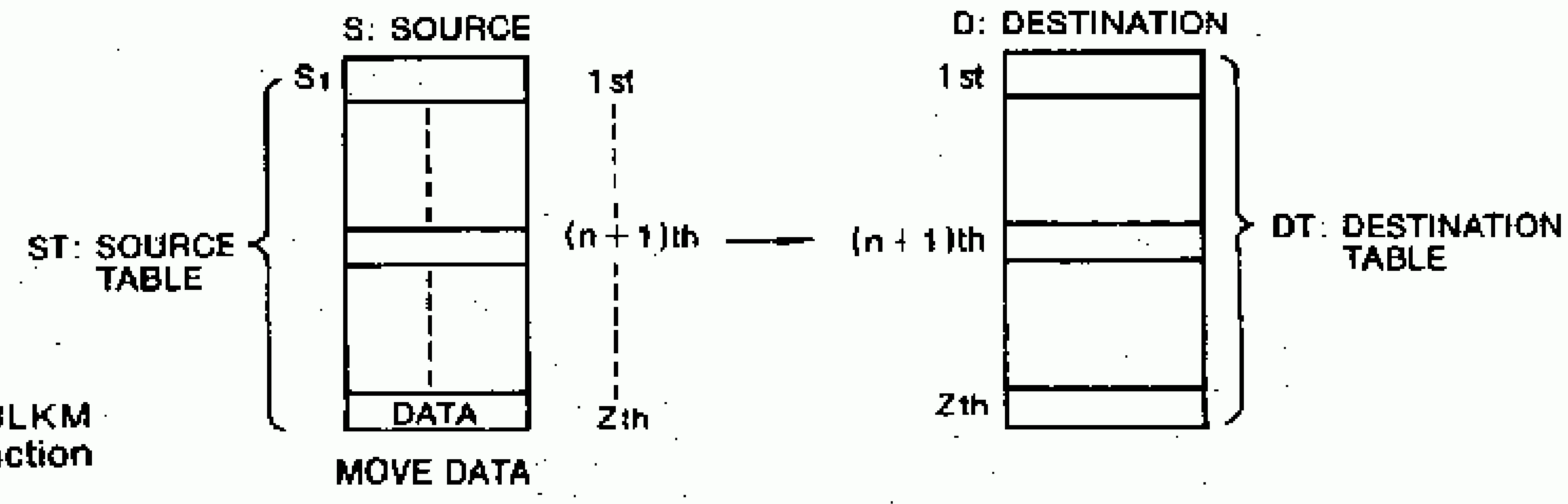


Fig. 4.49 BLKM Function

NOTE

All the registers (discrete) will be moved in one scanning cycle.

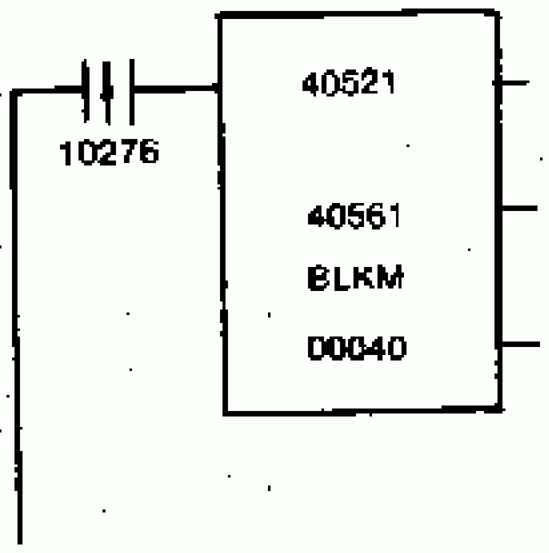
(b) Inputs

Only the input 1 is used with the BLKM function. When this input node receives power flow, the move is performed. Every scan, when enabled, the Block move will operate upon all registers.

(c) Outputs

The block move function utilizes only the output 1. The lower two outputs 2 and 3 have no significance and will be OFF under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network.

(d) Example



NET #64

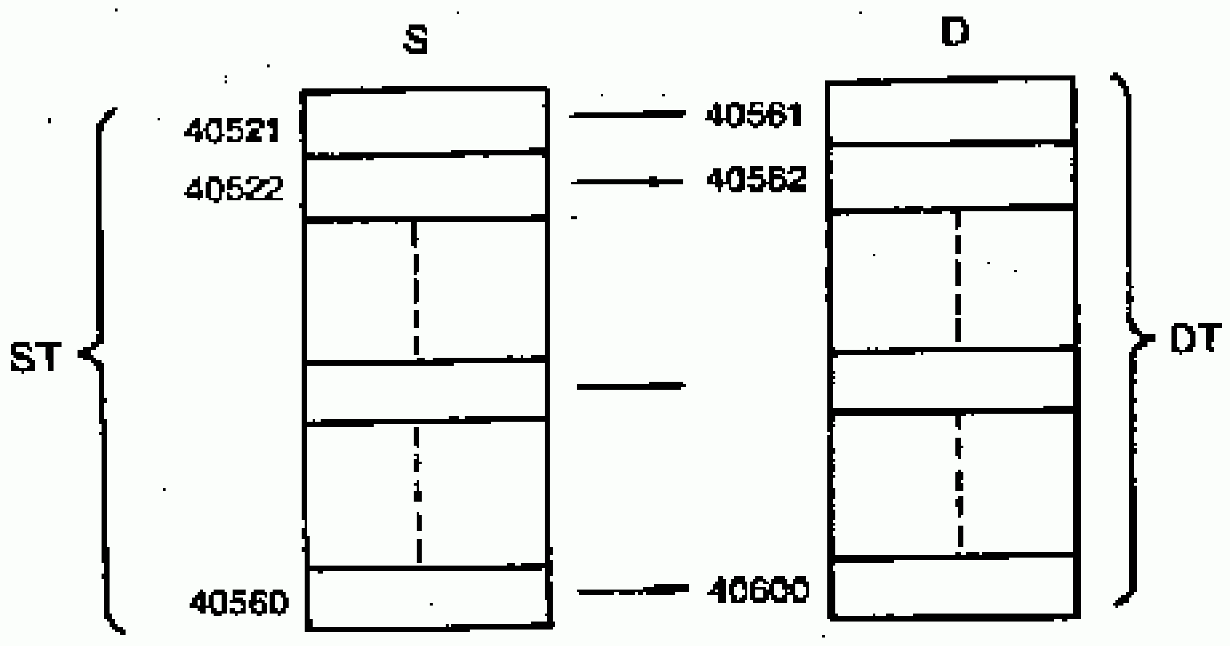


Figure illustrates a typical Block move function. As long as input 10276 is energized, the content of registers 40521-40560 will be moved into registers 40561-40600.

(5) First In (FIN)

(a) Form and Function

FIN is used in pair with a First Out (FOUT) (see (6)). This is a 1-to-N move like that of the R → T. The difference is that the data stored in the destination by FIN can be retrieved by FOUT in the order the data are stored. Fig. 4.50 and 4.51 show the form and function of FIN.

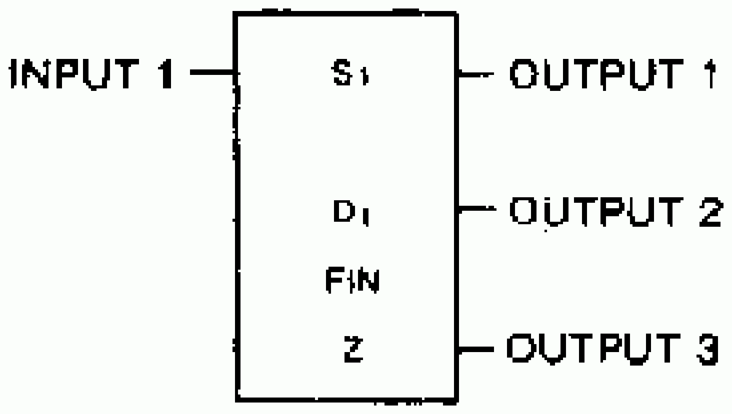


Fig. 4.50 FIN General Form

S1	Source reference	See Table 4.36.
D1	Destination reference(pointer)	
Z	Destination table size	
FIN	Symbol denoting first in (FIN)	—

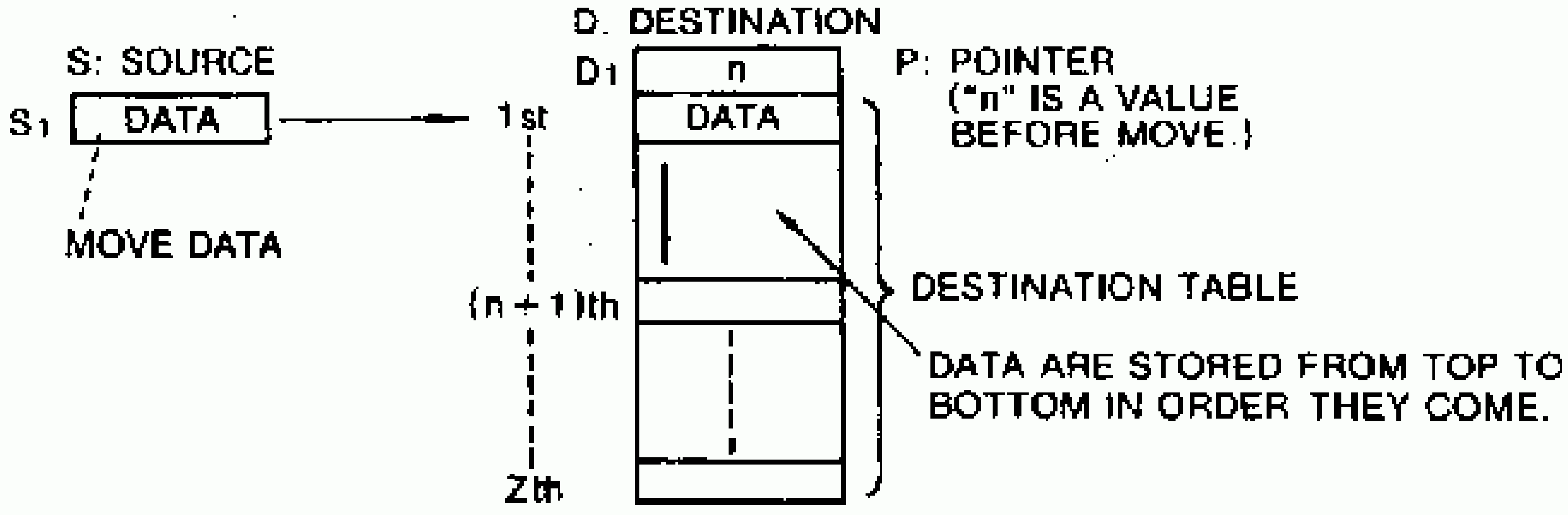


Fig. 4.51 FIN Function

NOTE

Move will be performed at the rate of one register (or a set of 16 discrete) per scanning cycle.

(b) Inputs

When the input 1 is ON, the contents of the destination table are shifted down by one, according to the content of pointer, the nth data (the oldest one) to the (n+1)th register, the (n-1)th data to the nth register, and so on, until the first data are shifted down to the second register. Then the source data is moved to the first register emptied. Thus the destination table is managed in such a manner that the data are stored in the table from top to bottom in the order they come. The pointer is incremented by one after the shift and move of data. This process is performed all in one scanning cycle, regardless of the table size. If n ≧ table size, no data will be moved even when the input 1 is ON. Inputs 2 and 3 are not used.

(c) Outputs

FIN function uses all three outputs. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever the table is full (pointer equal to table length); the output 3 supplies power flow whenever the table is empty (pointer equals zero). The outputs 2 and 3 do NOT require any inputs to receive power flow, they only require appropriate pointer values.

(d) Example

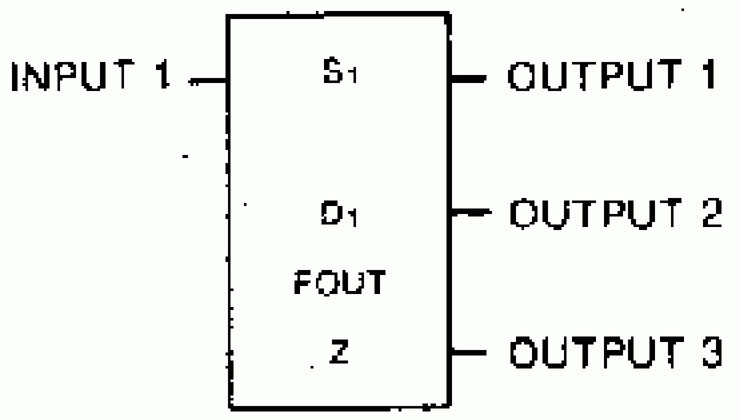
Example is shown in the section of FOUT.

(6) First Out (FOUT)

(a) Form and Function

This is a N-to-1 move that the destination table of FIN becomes the source table of FOUT. Figs. 4.52 and 4.53 show the form and function of FOUT.

Fig. 4.52 FOUT General Form

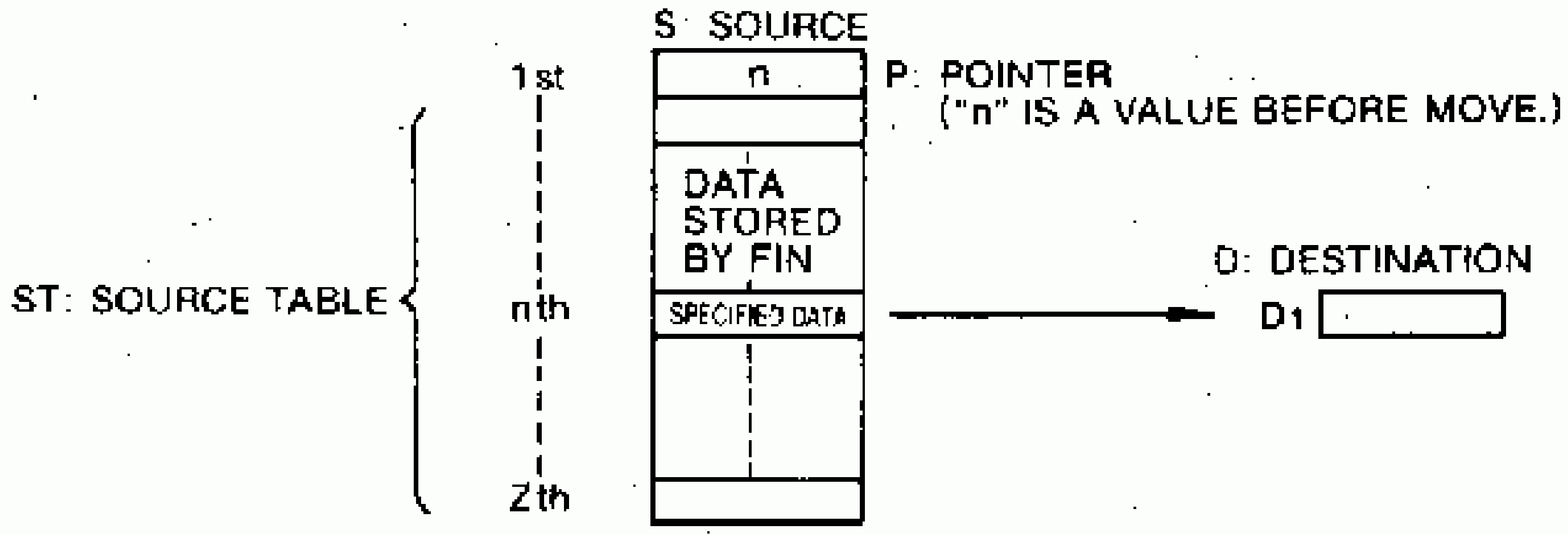


S_1	Source reference (pointer)	See Table 4.36.
D_1	Destination reference	
Z	Source table size	
FOUT	Symbol denoting first out (FOUT)	—

NOTE

Normally, the pointer is common to that of destination of FIN.
The table size is the same as that of FIN normally.

Fig. 4.53 FOUT Function



NOTE

Move will be performed at the rate of one register per scanning cycle. The data stored in a FIFO table will be retrieved by FOUT with the oldest one first, i. e. by the First in-First out principle.

(b) Inputs

When the input 1 is ON, the nth (but not the (n+1)th) contents of the source table are moved to the destination where n is the value of the pointer indicating the number of data stored. The pointer is decremented by one after move of the data. If n = 0, data will not be moved even when the input 1 is ON. The inputs 2 and 3 are not used.

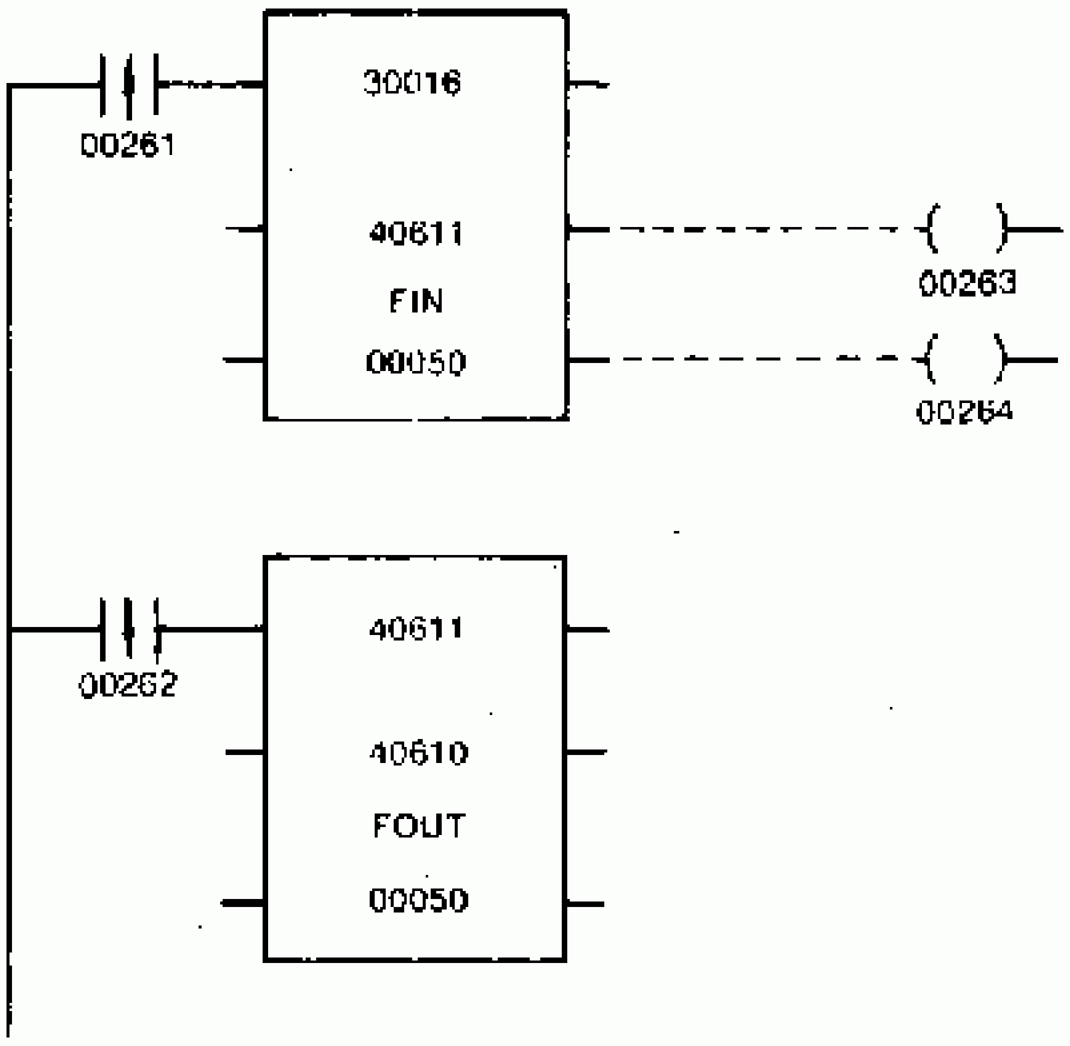
NOTE

The nth source register, whose contents are retrieved by FOUT, holds 0 unless new data is placed by FIN.

(c) Outputs

FOUT function uses all three outputs: each of the three outputs behaves in the same way on FIN function block. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever the table is full (pointer equal to table length); the output 3 supplies power flow whenever the table is empty (pointer equals zero). The outputs 2 and 3 do NOT require any inputs to receive power flow, they only require appropriate pointer values.

(d) Example



NET ♯ 65

Figure above illustrates a typical First-In—First-Out operation. When input 00261 is energized, the content of register 30016 is moved into the table (40612-40661) and the pointer incremented by one. Every scan this input is energized, a new entry is made into the table. Every time an insertion is made all the old data in the table is "pushed down" one register and the new data inserted at the top. Register 40612 therefore always contains the most recent entry into the table.

When input 00262 is energized, an entry stored in the table is removed, placed in register 40610, and the pointer is decremented by one. If the table was full (value 0050 in register 40611) coil 00263 would be energized and additional attempts to insert new data ignored. Coil 00264 is energized when the table is empty (zero in 40611) and additional attempts to remove data will be ignored.

If there were three valid entries in the table the content of register 40611 would be three and the entries stored in registers 40612 (newest), 40613, and 40614 (oldest). When input 00261 is energized, these entries are shifted down to registers 40613-40615, register 30016 copied into 40612, and the pointer incremented to four. If input 00262 is then energized, the value in register 40615 is placed in register 40610, and register 40611 decremented to three. All alterations of the register content and establishment of outputs will be accomplished in one scan for either FIN or FOUT operations.

NOTE

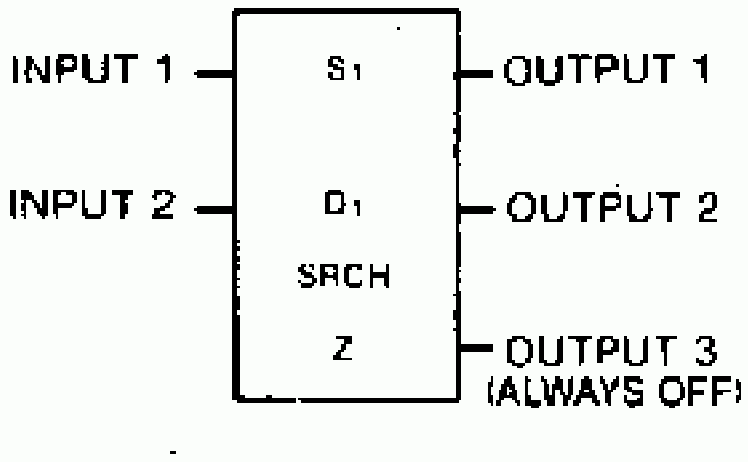
When FIN and FOUT are required during the same scanning cycle, the operation proceeds properly according to the order of logic solving.

(7) Table Search (SRCH)

(a) Form Function

This function searches a table of registers for a specified value. The source is not altered, only examined. If necessary, the SRCH function searches the entire table in one scan. It uses a pointer to indicate the location(s) within the table of registers which contain the value for which it is searching. This pointer is the only register whose value is altered by the SRCH function. Figs. 4.54 and 4.55 show the form and the function of the SRCH, respectively.

Fig. 4.54 SRCH General Form



S_1	Source reference	See Table 4.36
D_1	Destination reference	
Z	Source table size	
SRCH	Symbol denoting table search	—

(7) Table Search (Cont'd)

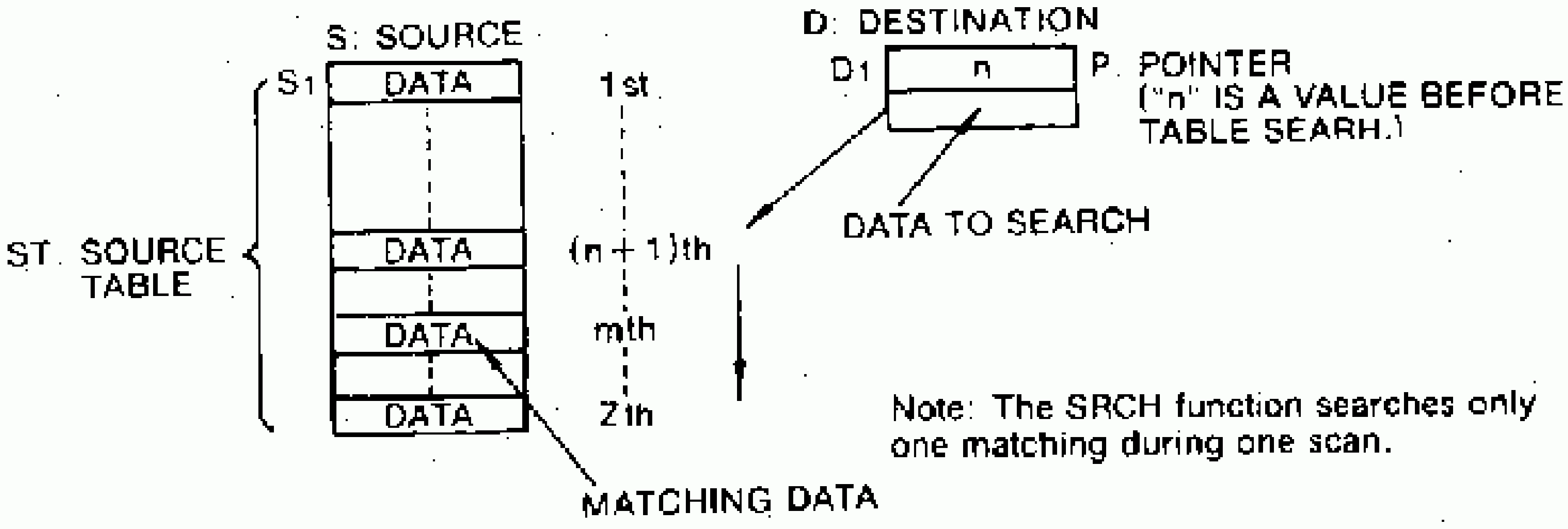


Fig. 4.55 SRCH Function

Before search, store the data to be searched in the location next to the destination pointer. Assume the value of the pointer is n when SRCH is required, then search is started from the (n+1)th source data. When a matching data is found at the mth location, the search will be completed with m placed in the pointer. If a matching data is not found, the search will be completed with 0 placed in the pointer.

SRCH searches for one piece of matching data during every scanning cycle. If all data of the source table matches, it takes N scanning cycles (N is the table size) to complete the search. If there is no matching data, the search is completed in one scanning cycle.

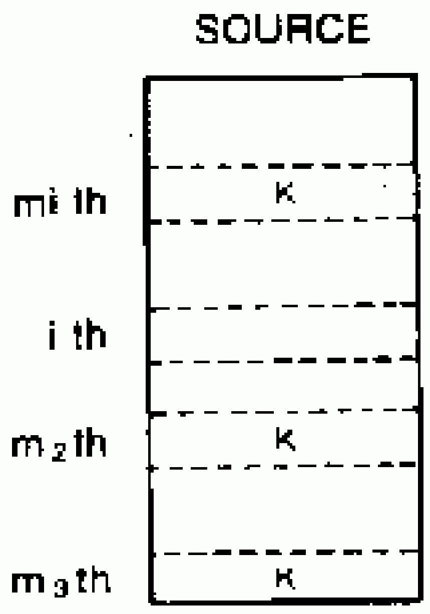
Where to start the next search depends on the status of inputs 1 and 2.

(b) Inputs

When only the input 1 is ON, n is set to 0 automatically before searching. SRCH always starts at the top of the source table. Even if there are many pieces of matching data in the source table, only the first one will be detected by search.

When the inputs 1 and 2 are ON, SRCH starts at the (n+1)th data of the source table when the pointer is n before searching. If the search is required from the i th data, set the pointer to i-1 and turn on the inputs 1 and 2.

If the m_1th, m_2th, and m_3th data of the source table are equal to the specified data K, set the pointer to i-1 and turn on the inputs 1 and 2. The search starts at the ith data and ends with $n = m_2$. The next search will start at the $(m_2 + 1)$th data and ends with $n = m_3$.



In any cases (only the input 1 is ON, or the inputs 1 and 2 are ON);

- If matching data is not found, the search of the scanning cycle is completed with n = N (N is the table size) then n is cleared to 0. (The search is not performed again from the top of the table during the same scanning cycle.)
- If the input is turned on when n ≥ N, n is automatically cleared to 0 before searching. As a result, the search is started at the top of source data. The input 3 is not used.

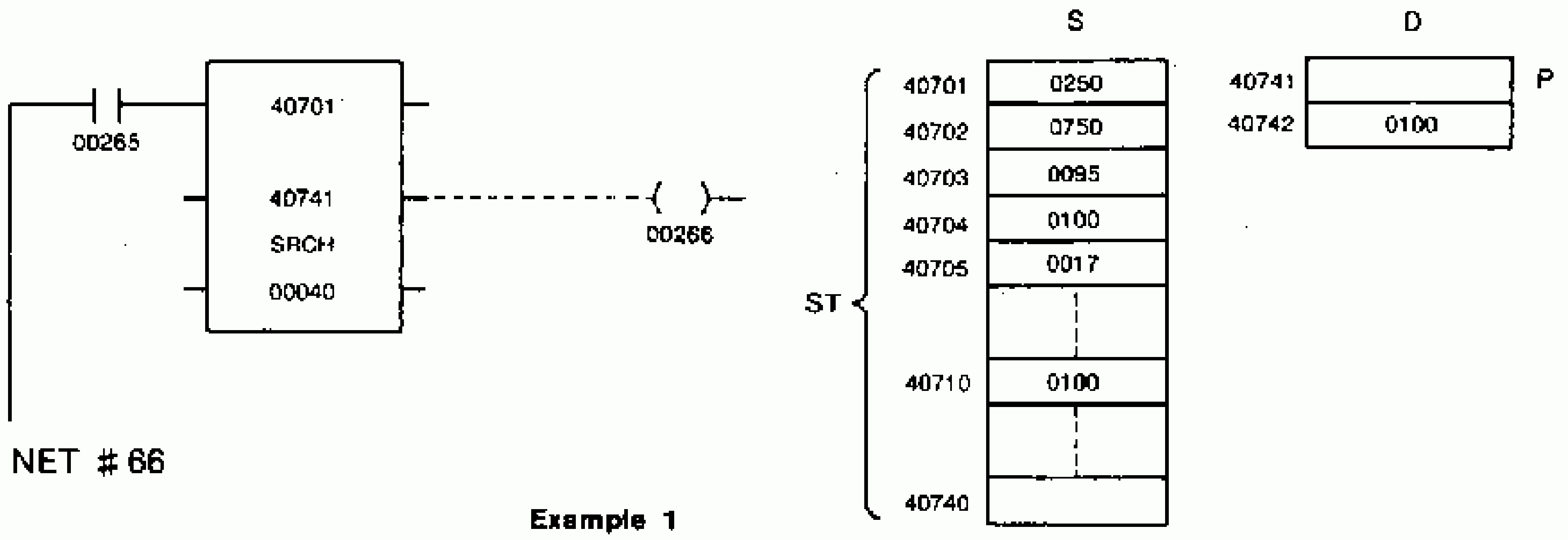
NOTE

The pointer is 0 whenever the input 1 is OFF.

(c) Outputs

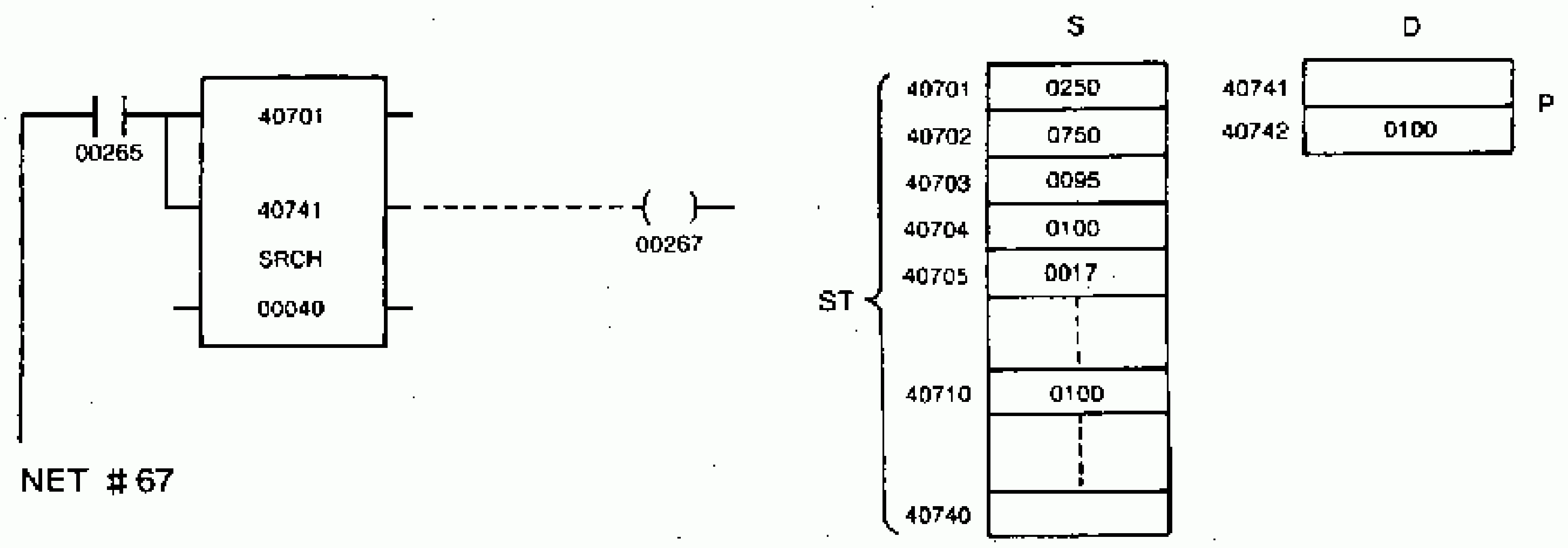
The SRCH function utilizes only the outputs 1 and 2. The output 3 has no significance and will be OFF (no power flow) under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever a match is found; if no match is found this output will not supply power flow and the pointer will contain the number zero.

(d) Examples.



Example 1

Figure above illustrates a typical SRCH operation. When coil 00265 is energized, the contents in the table 40701 to 40740 are compared to the value in register 40742. The first match found will stop the search, place the element number in register 40741, and energize coil 00266. Every scan that coil 00265 is energized, the SRCH will start at register 40701. If the value in register 40742 is 0100 and the table contents are: 0250, 0750, 0095, 0100, 0017, etc. the pointer will have the content 0004 after the search.



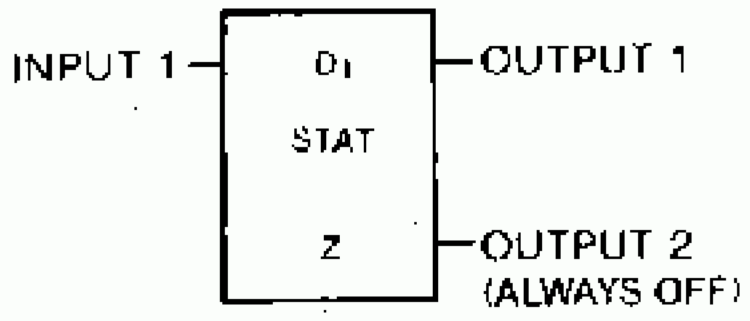
Example 2

To search for additional matches, both the top and middle inputs should receive power flow. This can be accomplished by adding a vertical connection after the control contact (reference 00265). If register 40741 contains the value zero when coil 00265 is energized in the modified network, the search begins with register 40701. Assuming the same numerical values, the search will stop at the fourth element, with the value 4 in register 40741 and coul 00267 energized. On the next scan with coil 00265 still energized, the search begins at the fifth element and will continue until either another value 0100 is encountered or the end of the table is reached. The state of coil 00267 will indicate which condition was reached after the second search.

(8) Get Controller System Status (STAT)

(a) Form and Function

This function provides the user with access to the U84 system status. The STAT is useful in the design of system diagnostics. Fig. 4.56 shows the form of STAT.



D1	Destination reference	See Table 4.36.
Z	Destination table size	
STAT	Symbol denoting the get controller system status	—

Fig. 4.56 STAT General Form

NOTE

The source is fixed by the U84's internal design and is not under user control.

Twenty-one registers worth of status can be obtained every scan. The length of the status move can be adjusted; however, it will always start with the U84 machine status.

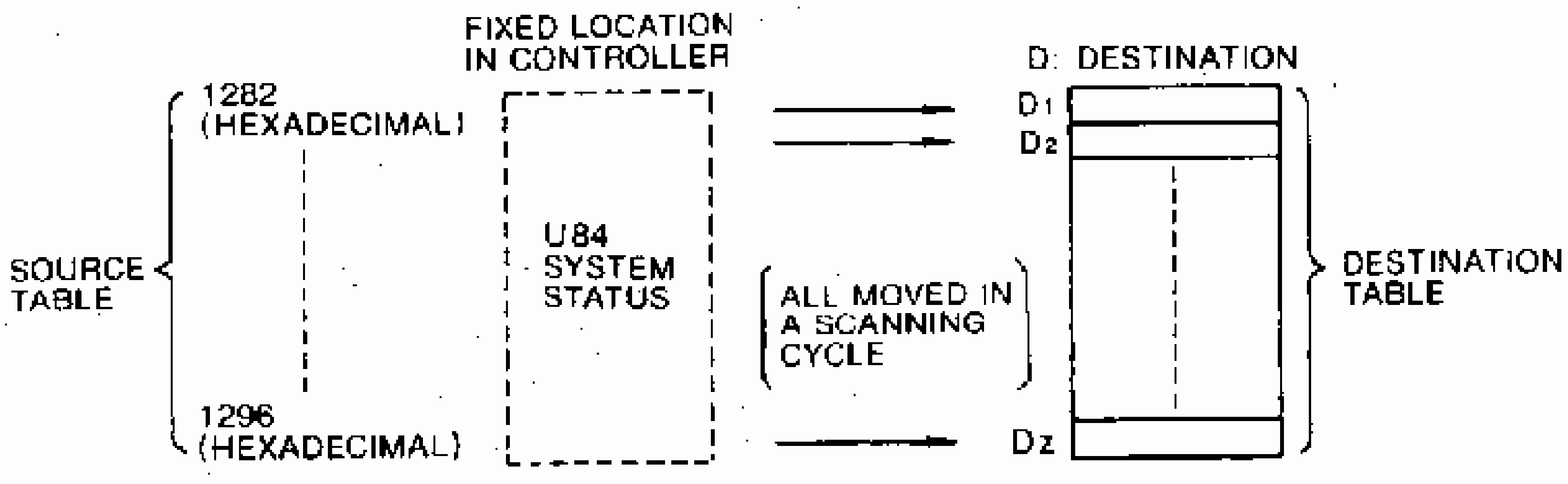


Fig. 4.57 STAT Function

(b) Inputs

Of the two available inputs, only the input 1 affects the operation of the STAT function. When this input node receives power flow, all statuses are moved to the destination in one scan.

(c) Outputs

Of the two available outputs, only the output 1 is used. The output 2 has no significance and will be OFF under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network.

(d) System Status

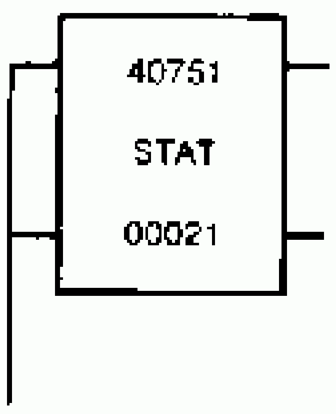
The moved system statuses are located in the destination as shown in Table 4.37.

Table 4 37 System Status

Order*	Content of System Status	Fixed Location (Hexadecimal)
1	Machine status	1282
2	Stop status	1283
3	Local I/O channel 1 (I/O CH #1) status	1284
4	Local I/O channel 2 (I/O CH #2) status	1285
5	Remote I/O CH1 (I/O CH #3) status	1286
6	Remote I/O CH2 (I/O CH #4) status	1287
7	Remote I/O CH3 (I/O CH #5) status	1288
8	Remote I/O CH4 (I/O CH #6) status	1289
9	Remote I/O CH5 (I/O CH #7) status	128A
10	Remote I/O CH6 (I/O CH #8) status	128B
11	Remote I/O CH7 (I/O CH #9) status	128C
12	Remote I/O CH8 (I/O CH #10) status	128D
13	Communication module status	128E
14	Local or remote I/O driver module status	128F
15	No. 1 ASCII module status	1290
16	No. 2 ASCII module status	1291
17	No. 3 ASCII module status	1292
18	No. 4 ASCII module status	1293
19	Not used (for future expansion)	1294
20	U84 dual stop status	1295
21	U84 dual machine status	1296

*If the destination is specified as 40101, order 1 is 40101 and order 2 is 40121.
Note: See Section 7.6 for details of system status.

(e) Example



NET #67

Status information can be obtained constantly because the input 1 is connected to the power rail at left. It is stored in 21 registers of 40751 to 40771, in the order shown in Table 4.37.

4.7.4 Programming Data Move Circuit and Precautions

- Inputs to the data move circuit may be outputs of relays, timers, counter, arithmetic operations, matrices, and other data move circuits.
- Coils need not be connected to three output nodes (1, 2 and 3) of a data move circuit. It is permitted to connect a relay contact to the output nodes at right or connect the output node directly to an input node of an arithmetic circuit, except relays.
- To execute an operation constantly, connect the input directly to the power rail at left. To execute it only during one scanning cycle, use a transitional contact as an input.
- The range of the source or destination table specified by a table size must be within the range of the reference numbers of input relays, coils, or registers.
- Since the output 1 supplies power flow whenever the input 1 receives power flow, cascaded operation is possible. It is possible to OR the outputs by connecting a vertical shunt element.

  When coil 0×××× is to be specified as destination, the number of the coil group cannot be the same as the number of a coil group used in another circuit. If it is attempted, the message "COIL IS USED" is displayed on the CRT screen of the programming panel and input operation will be rejected. Coil 02049 is used for battery monitor and the destination table cannot include coils 02049-02064.

  When coil 0×××× is specified as destination, the coil is turned on and off by data move function even if it is disabled.
- When a pointer is used, its value does not exceed the table size normally. If the value is made greater than the table size forcedly by an another circuit, move is not executed (SRCH executes it after resetting the pointer to 0) and the pointer becomes equal to the table size.
- The source remains unchanged data after the move.

4.7.5 Example-Application Circuits of Data Move

(1) Setting and Reading Data (Applications of R→T and T→R)

This is an example to set and read a time of externally-preset timers. Fig. 4.58 shows a sample layout of switches and indicator.

Setting:

Set a timer number and time by the digital switches and push the SET pushbutton switch to store the set time in the register associated with the timer number and indicate the time.

Reading:

Set a timer number and push the READ pushbutton switch to read out the set time corresponding to the timer number from a table and indicate the time.
It is assumed that the timer number will be N and the set time will be TN.

(a) Hardware Configuration

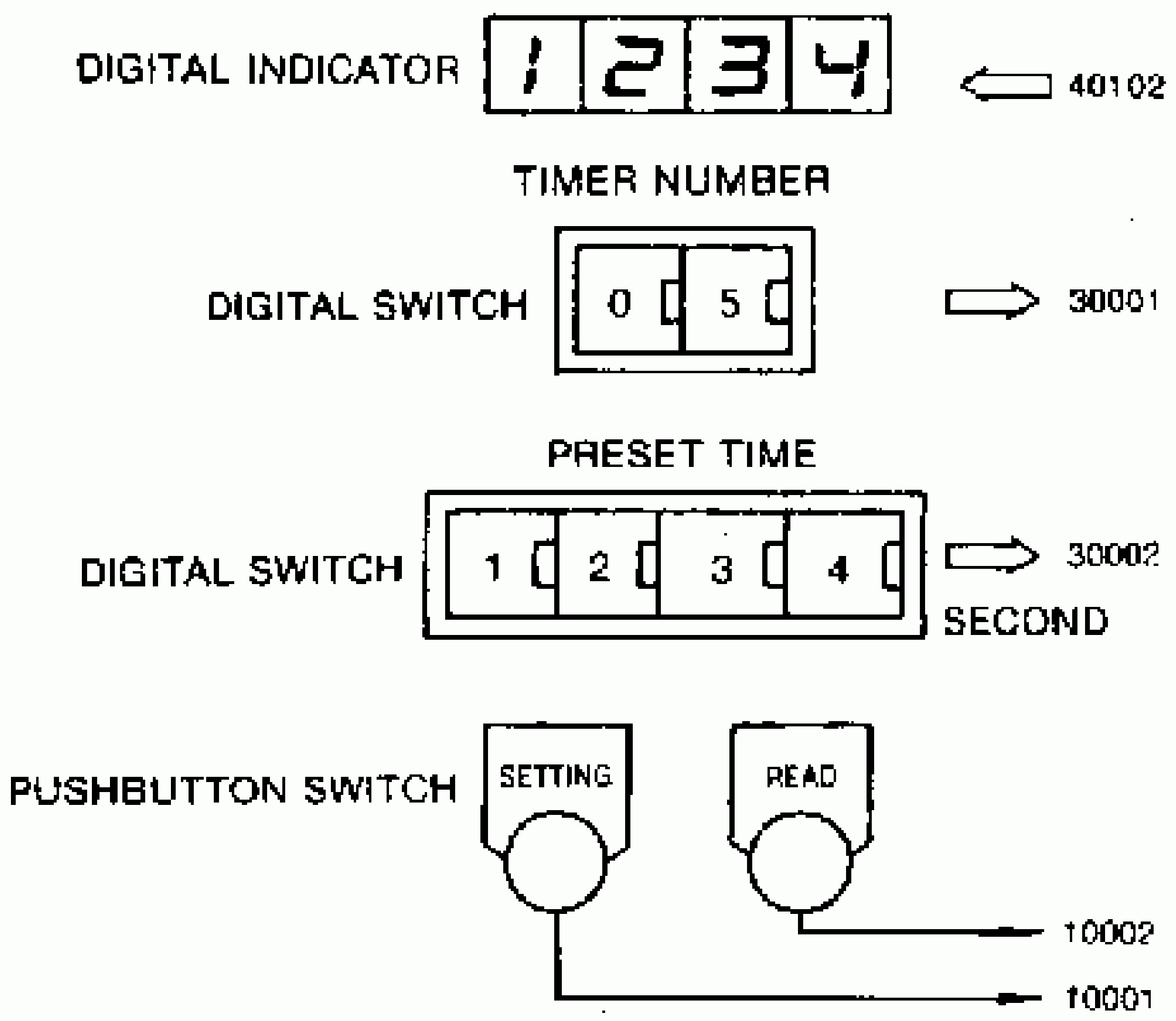
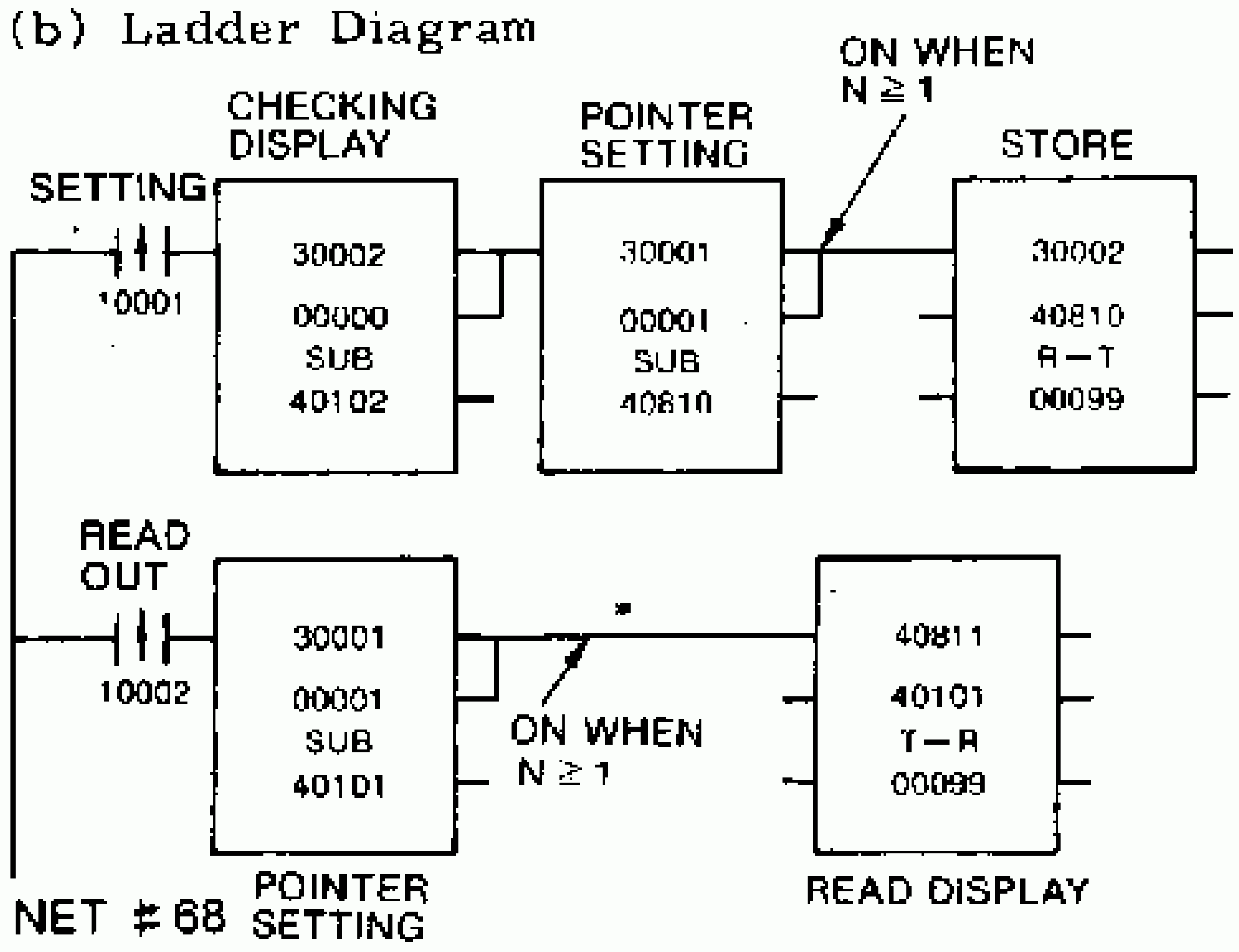


Fig. 4.58 External Devices and Assignments

(b) Ladder Diagram



* Horizontal shunt is needed.
Note: 40101 is used as the pointer but not as an output register.

Fig. 4.59 Setting/Reading Circuits

(c) Data Flow

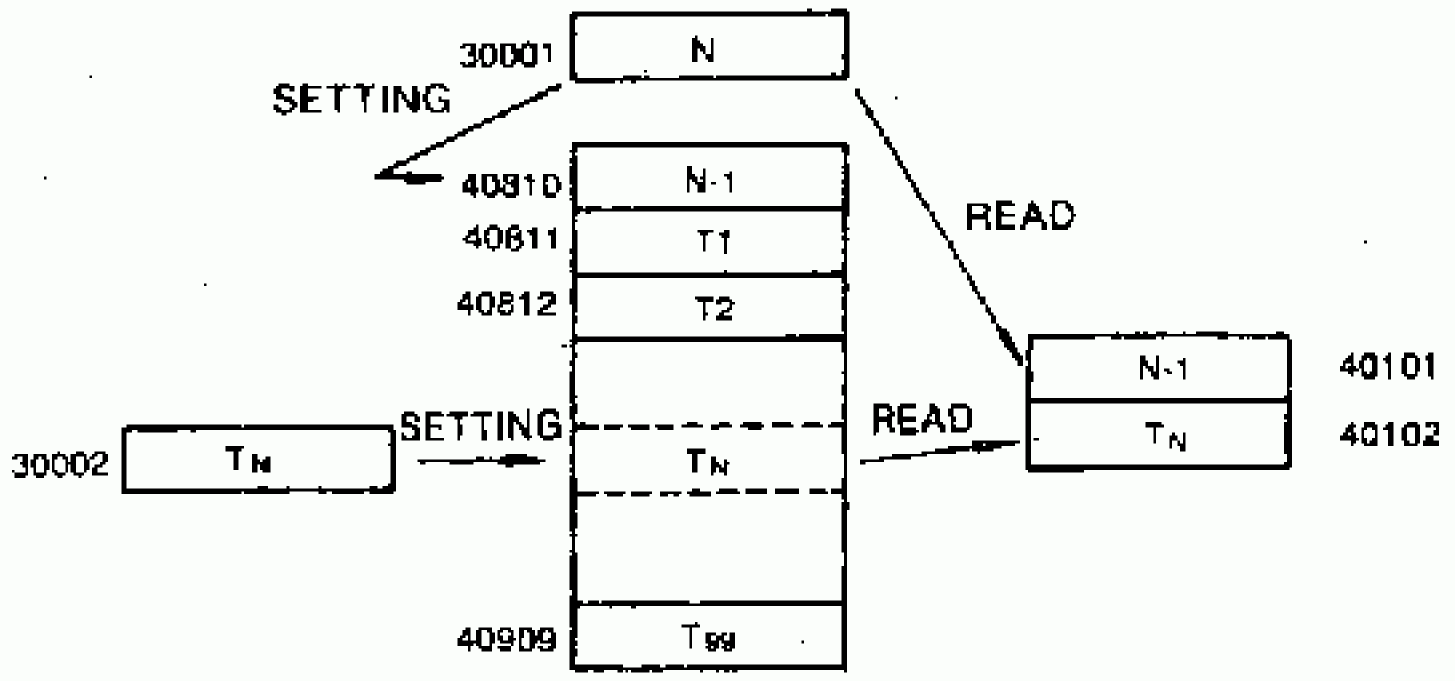
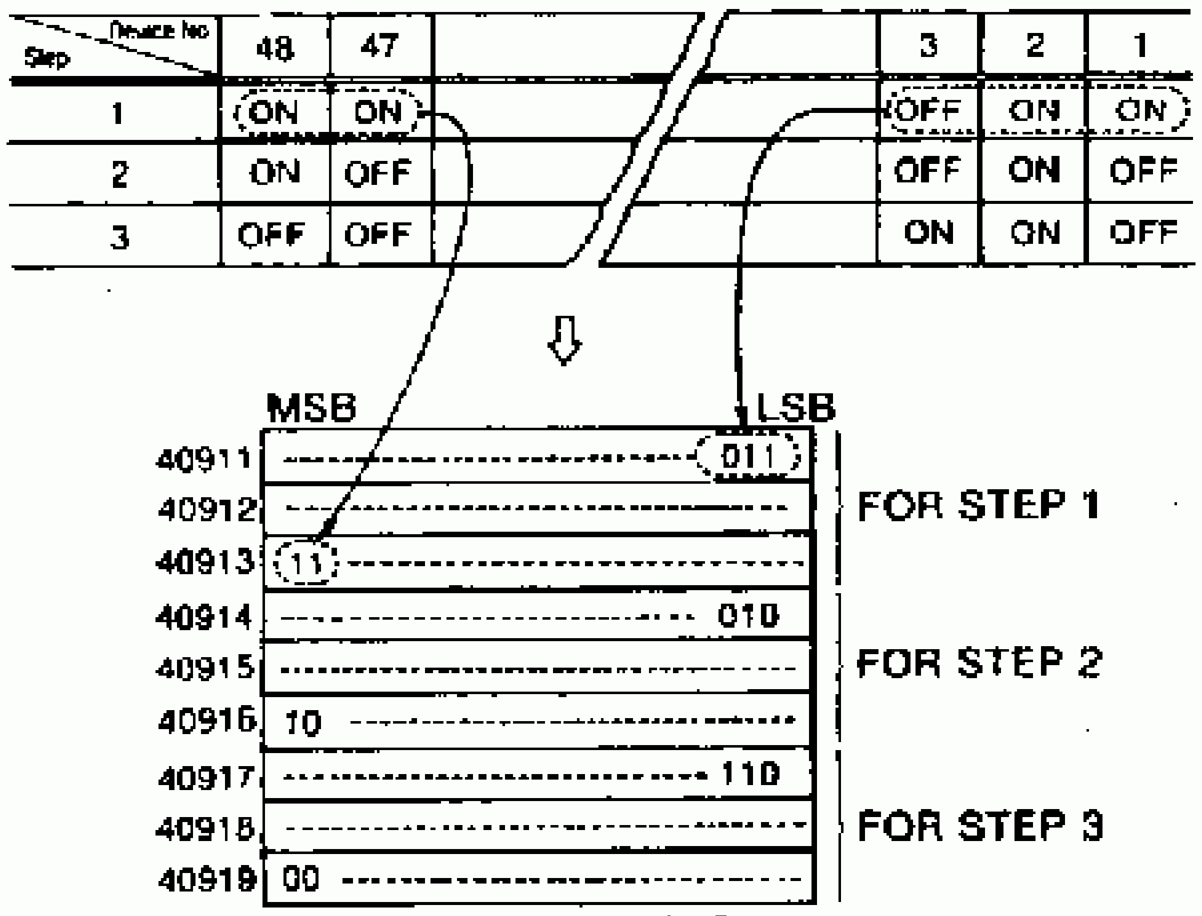


Fig. 4.60 Data Flow

(2) Move of Bit Pattern (Application of BLKM)

It is possible to execute a step-by-step sequence by storing the sequence of ON/OFF steps as a bit pattern in advance and taking out the bit pattern when executing. The example below is of three steps and with 48 output devices.

(a) Bit Pattern



(b) Ladder Diagram

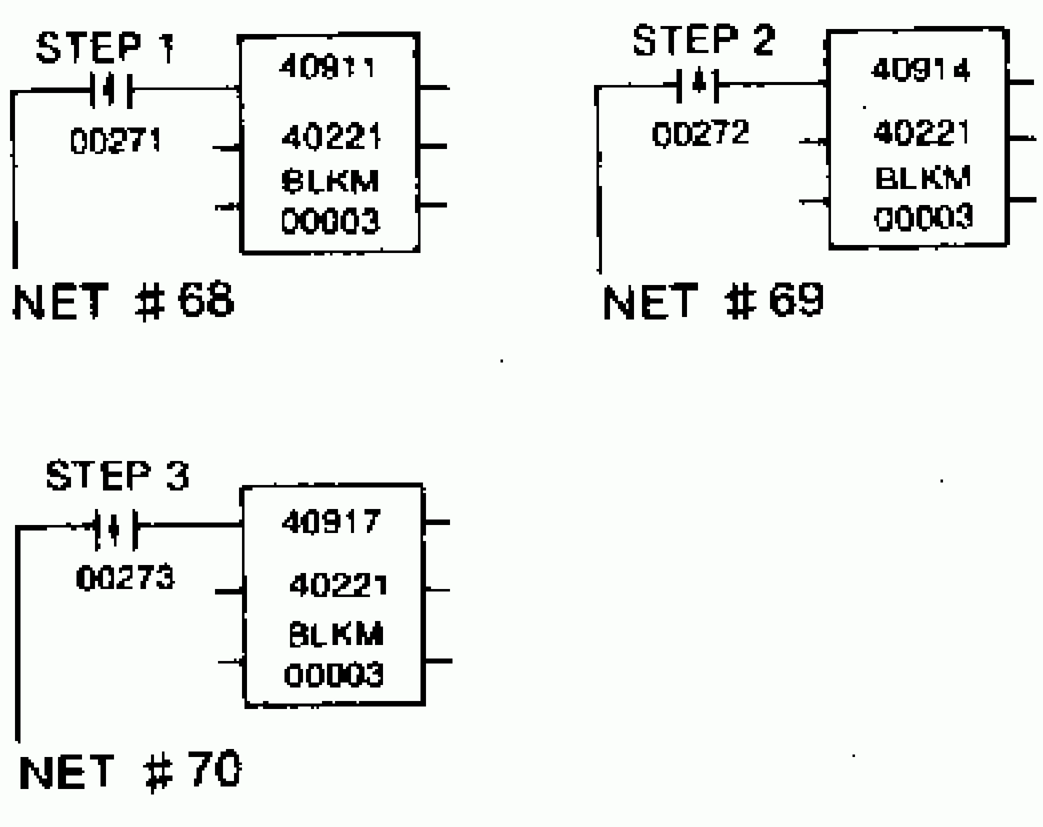
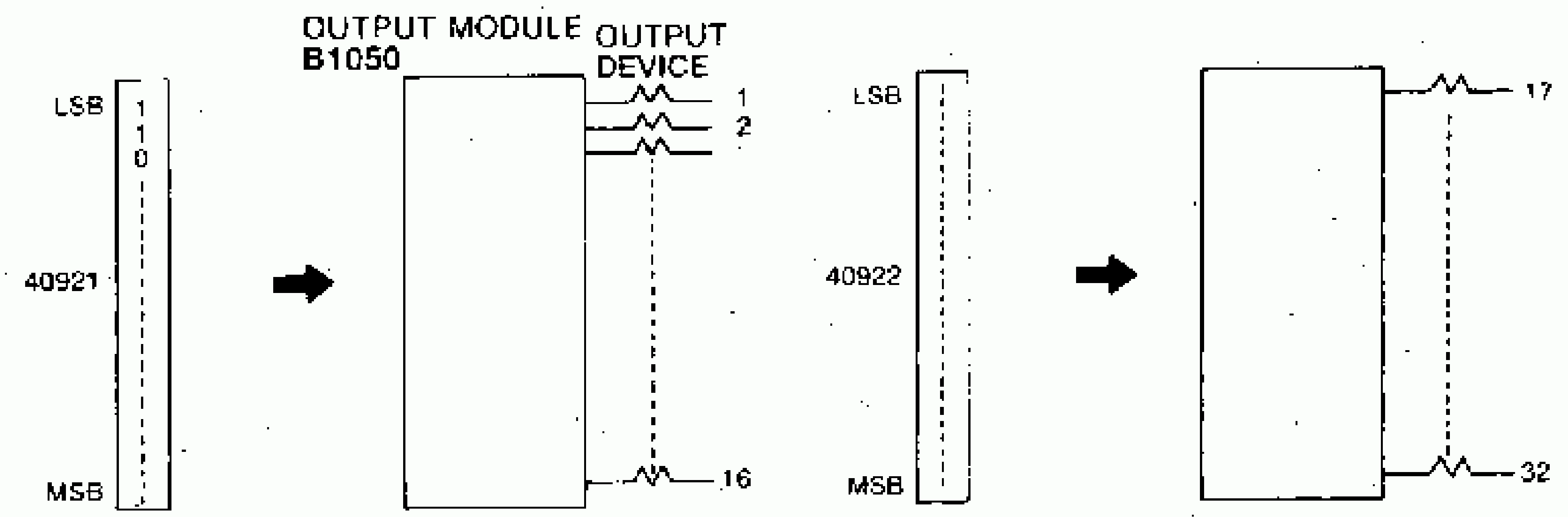
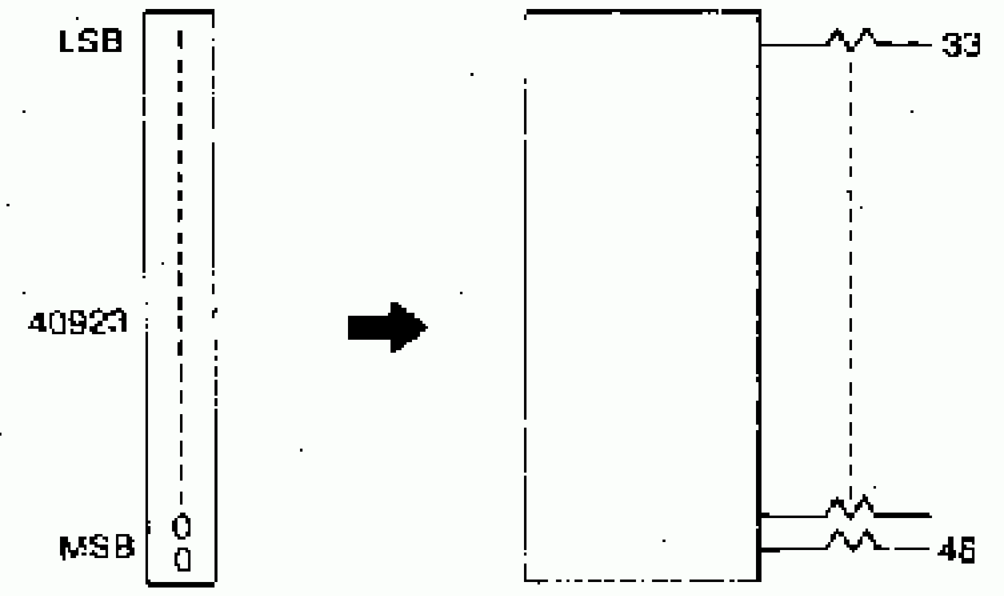


Fig. 4.61 Bit Pattern and the Ladder Diagram

(2) Move of Bit Pattern (Cont'd)

(c) Output of Bit Pattern





NOTE

1. Figure above shows the status of step 1.
2. 40221—40227 must be allocated in binary form.
3. Pay attention to the correspondence of LSB and MSB of register and the output device numbers.

(3) Reservation (Applications of FIN and FOUT)

In this example, articles are grouped by destination and placed on delivery tracks. It is assumed that branching and interruption do not happen during transportation.

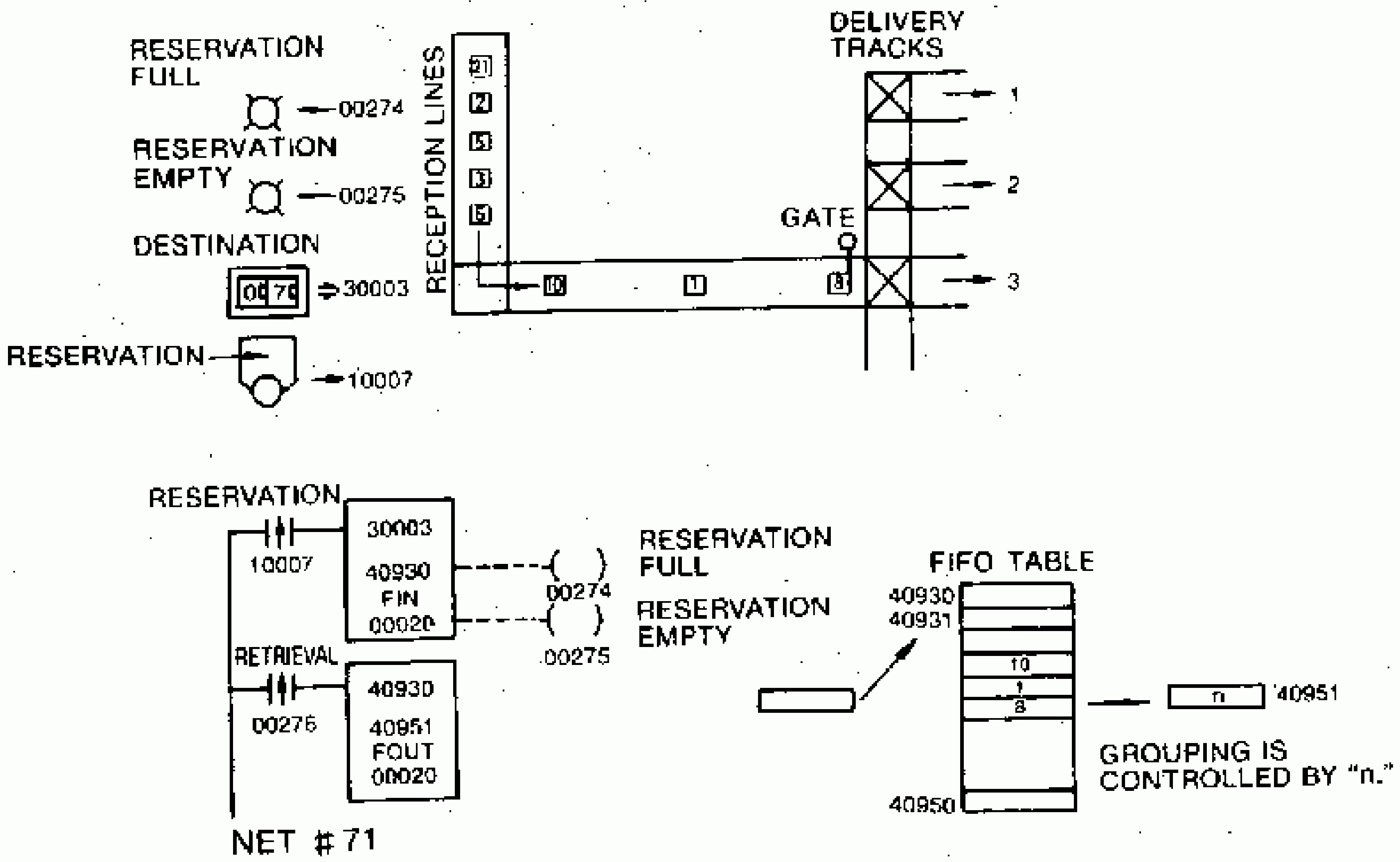
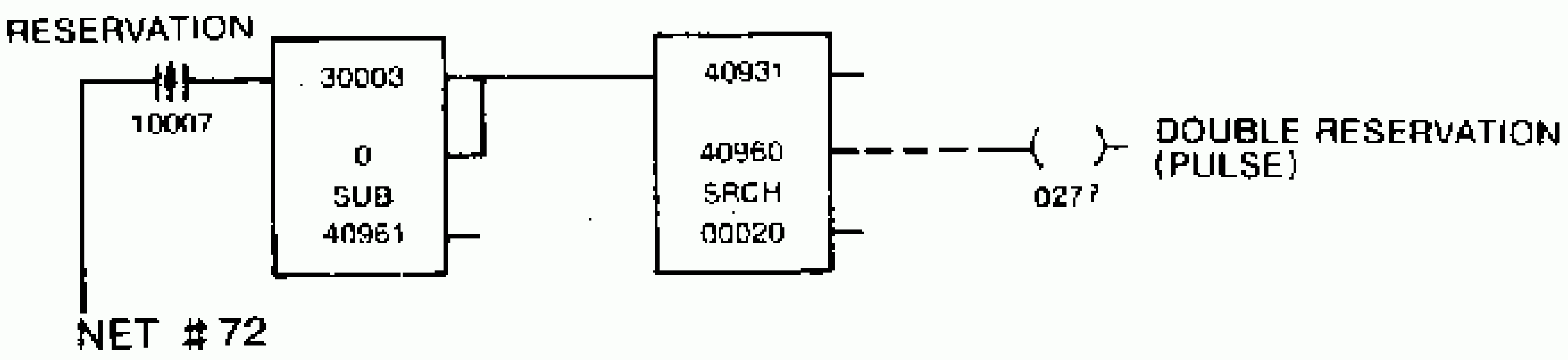


Fig. 4.62 Reservation Circuit

(4) Prevention of Double Reservation (Application of SRCH)

This is an example to prevent double reservation in the system shown in (3) above.



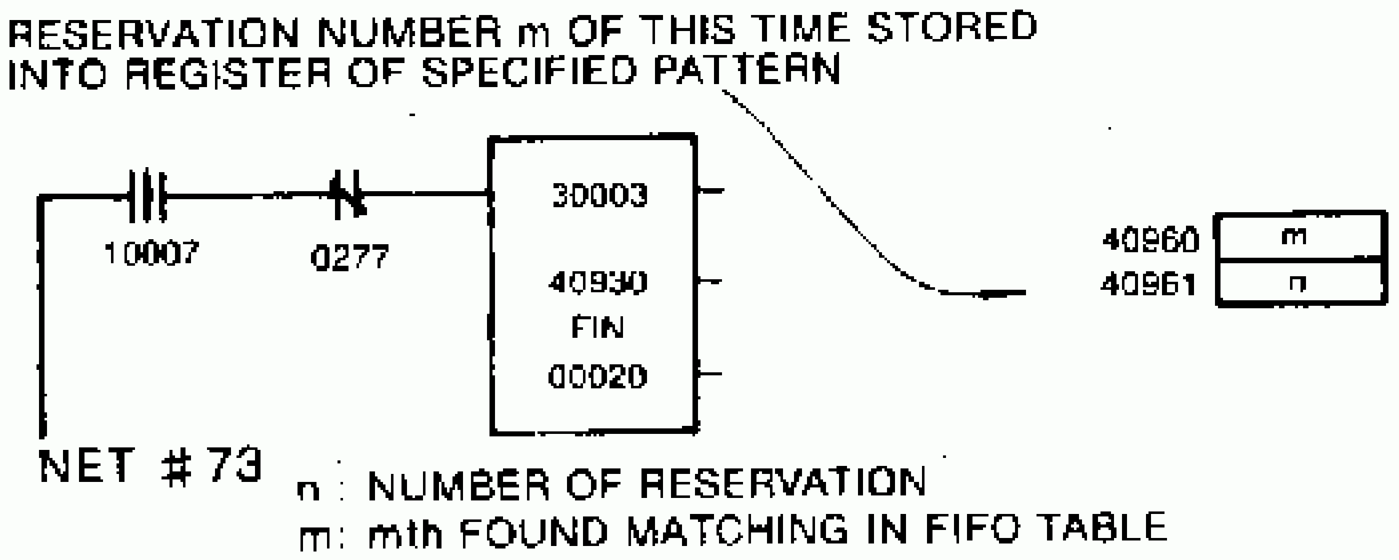


Fig. 4.63 Prevention Circuit of Double Reservation

4.8 INDEXED BLOCK MOVE

Indexed block move is an N-to-N move executed in one scanning cycle, like block move (BLKM) described in (4) of 4.7.3 In addition, there are following funcitons:

- The source or destination table can be specified according to the value (index) of the pointer.
- A group of input relays can be specified as the destination table.

4.8.1 Types of Indexed Block Move

Four types of the indexed block move are available as shown in Table 4.38.

Table 4.38 Types of Indexed Block Move

Type	Symbol	Description	Page
Block Move 1 with Destination Index	DIBT	Permits to specify destination table (input relay groups) according to pointer.	114
Block Move 2 with Destination Index	DIBR	Permits to specify destination table (holding register groups) according to pointer.	117
Block Move 1 with Source Index	SIBT	Permits to specify source table (coils, input relays and input register groups) according to pointer.	120
Block Move 2 with Source Index	SIBR	Permits to specify source table (holding register groups) according to pointer	123

4.8.2 Block Move 1 with Destination Index (DIBT)

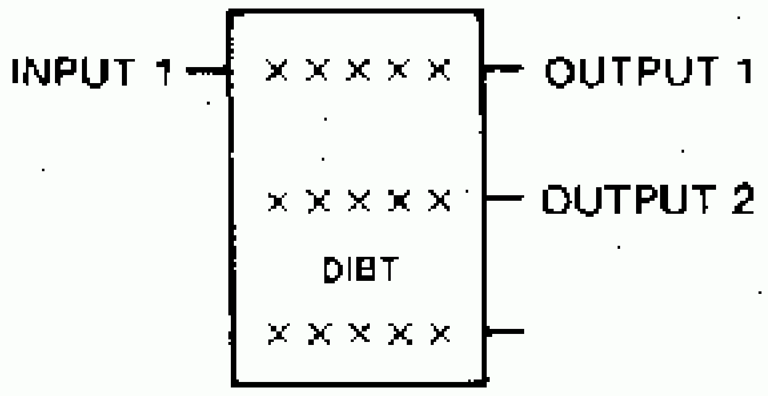


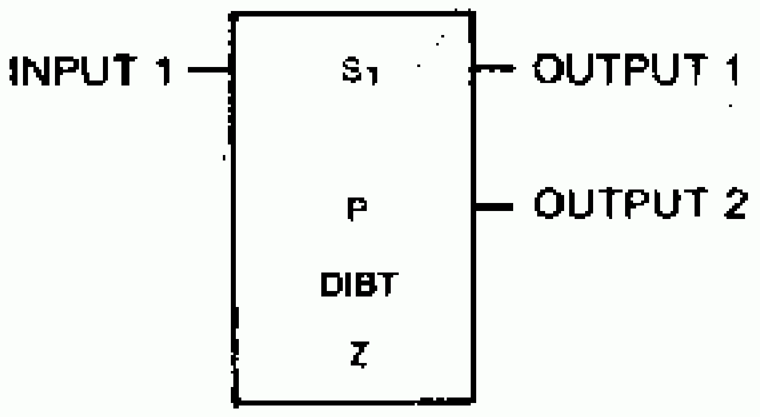
Fig. 4.64 DIBT General Form

(1) Form

Fig. 4.64 shows the form of the block move 1 with destination index (DIBT). Block move 1 with destination index requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.39, specify any of constant K, reference numbers 0×××× of a coil, 1×××× of an input relay, 3×××× of an input register, and 4×××× of a holding register for each of the elements.

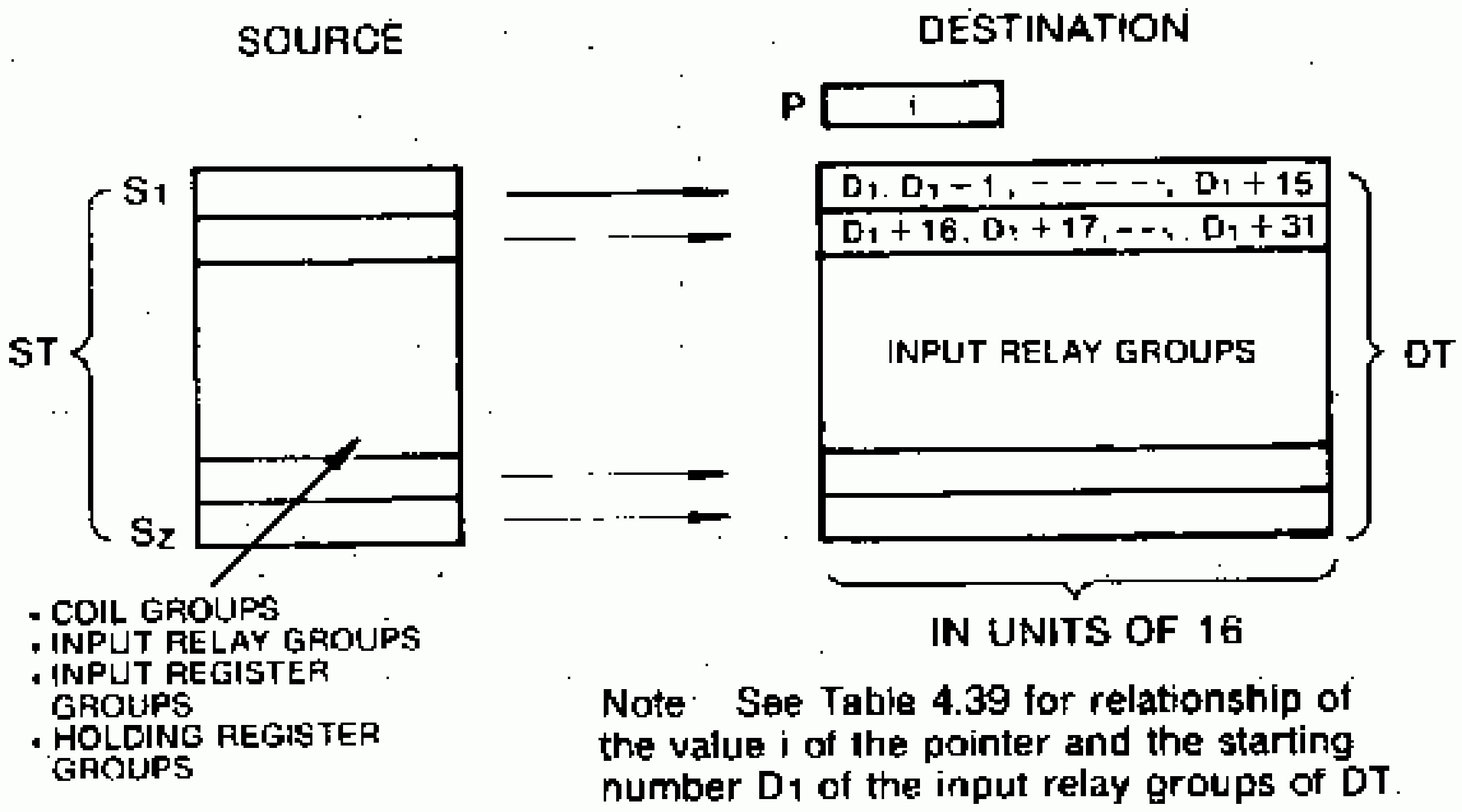
DIBT is the symbol denoting the block move 1 with destination index.

(2) Function and Operation



- By the block move 1 with destination index (DIBT), all data of the source table (ST) will be moved to the destination table (DT: input relay groups) specified with the value i of the pointer (P) when the input 1 is ON. The output 1 is turned on. The move will be completed in one scanning cycle.

Fig. 4.65 DIBT Fuction

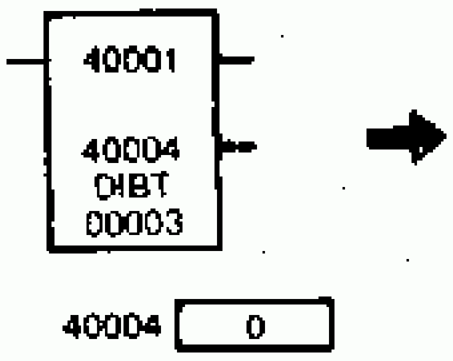


Note: See Table 4.39 for relationship of the value i of the pointer and the starting number D_1 of the input relay groups of DT.

- In the following cases, the move will not be executed and the output 2 is turned on.

(a) When the value i of the pointer is out of the range of 1001-1128:

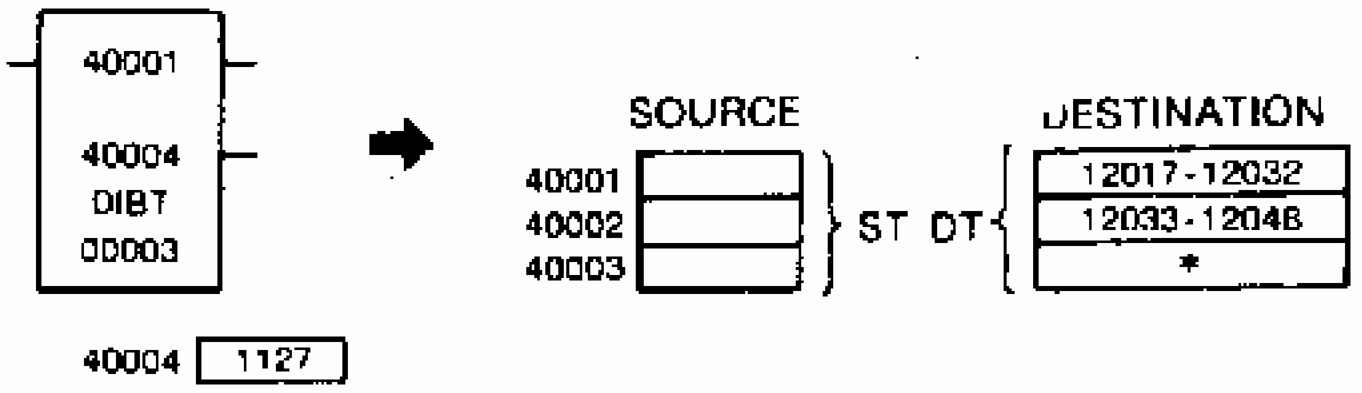
Example



There is no input relay group corresponding to i=0. Therefore, no destination tables corresponding to the source tables 40001-40003 exist.

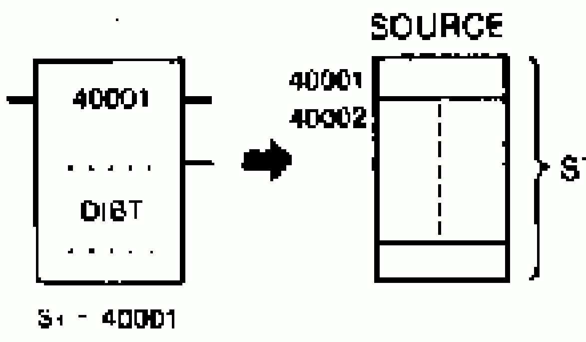
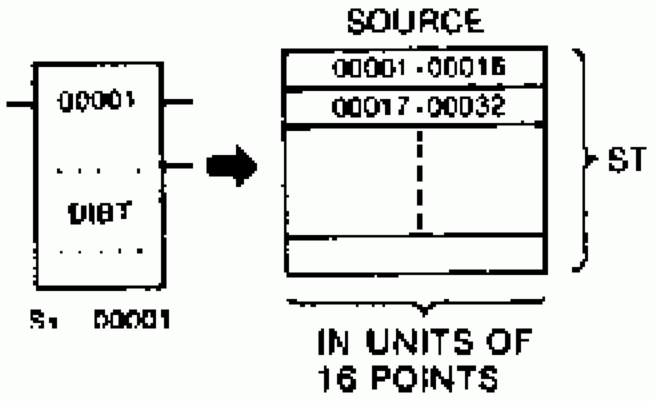
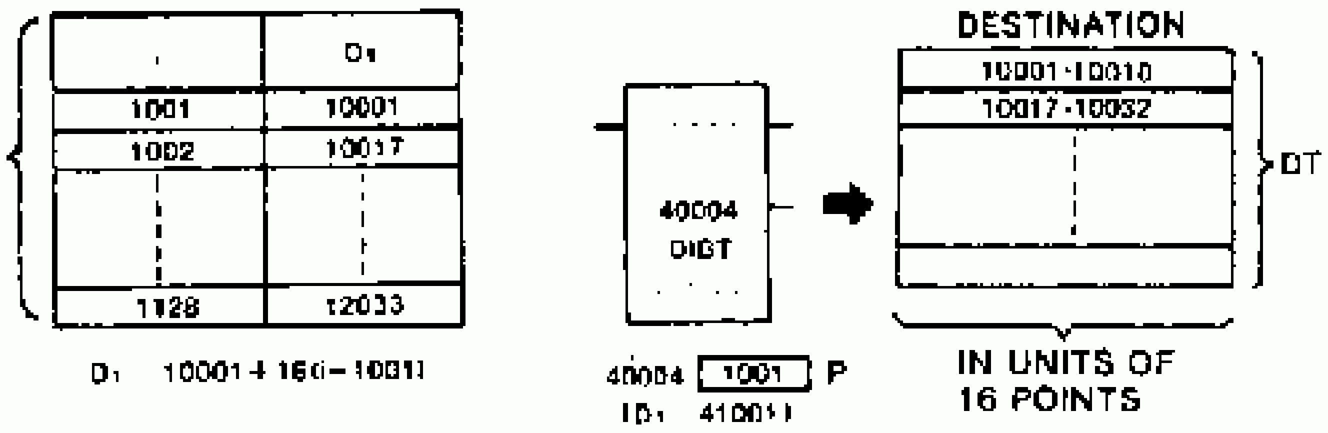
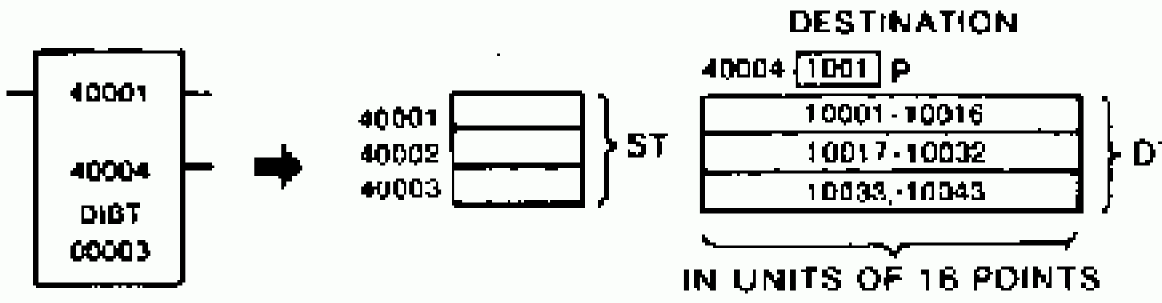
(b) When the value i of the pointer is within the range of 1001-1128 but no destination table to be specified by the value exists:

Example



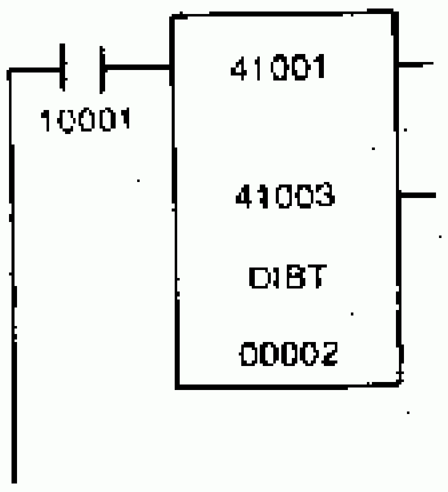
*No input relay groups of destination corresponding to the source 40003 exist.

Table 4.39 Elements of Block Move 1 with Destination Index

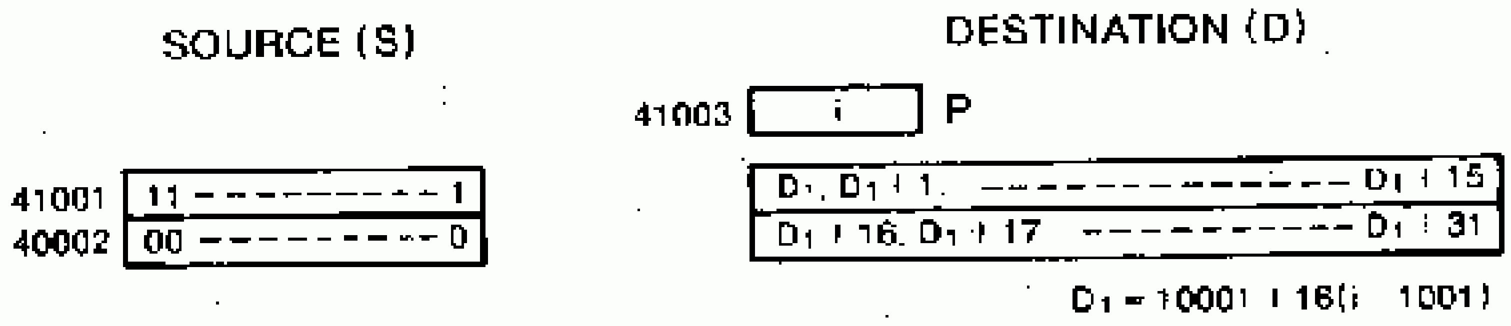
Element	Specified Number	Description
Top	Any of the following: • 0×××× (00001, 00017, 00033,··· 04081)* • 1×××× (10001, 10017, 10033, ··· 12033)* • 3×××× (30001–30256) • 4×××× (40001–49999)	Indicates starting number S_1 of source table (ST).  **Example 1**  **Example 2**
Middle	4×××× (40001–49999)	• Indicates pointer (P). It is possible to specify starting number of the destination table (DT: input relay group) by the value i of the pointer. • Relationship of the value i of the pointer and the starting number D_1 of the input relay groups of DT is shown below. • Relationship of i and D_1  **Example**
Bottom	Constant K (00001–00100)	Indicates the size of the source table (ST) and the destination table (DT) **Example**  **Example**

*n = 16 m + 1 (m = 0, 1, 2, ----) when n is ×××× of 0×××× or 1××××.
Note: Be sure to disable (DISABLE ON or DISABLE OFF) the input relay groups of the destination table.

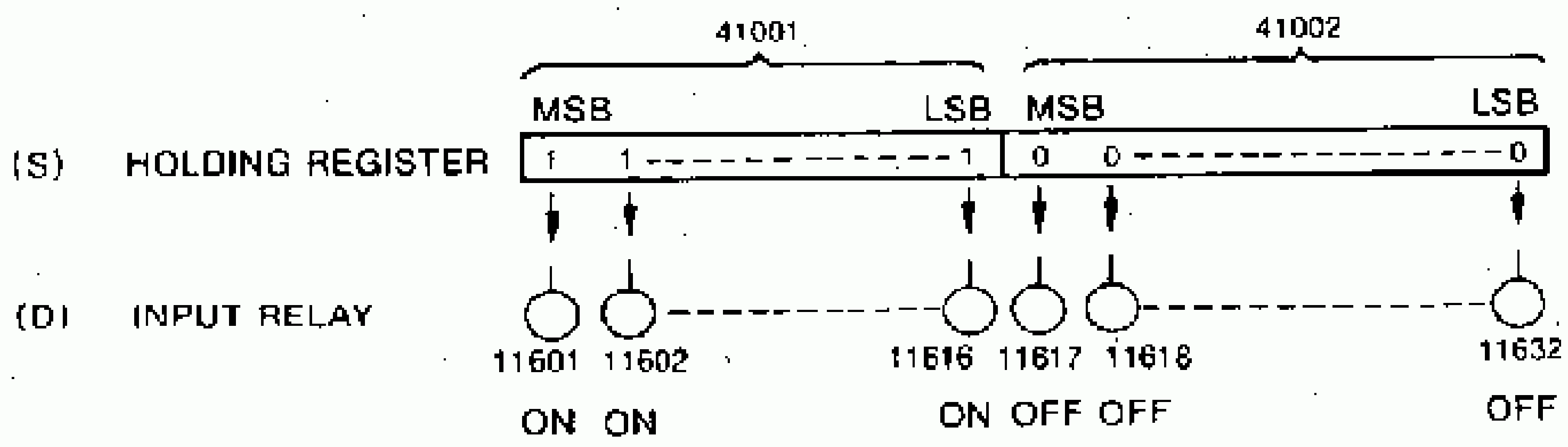
(3) Example



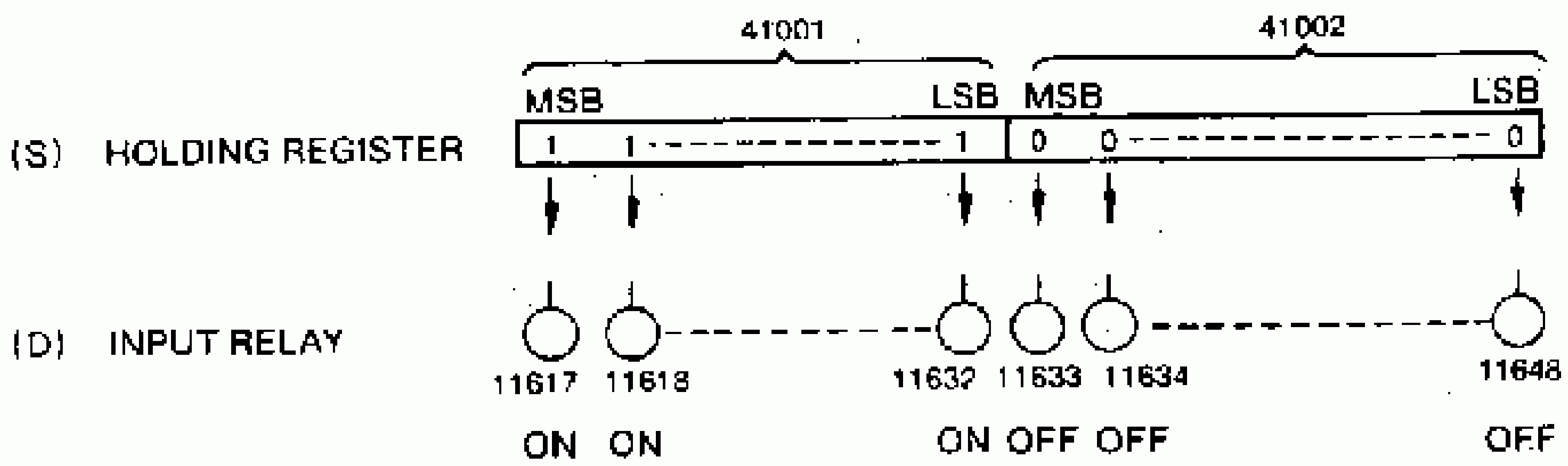
NET #81



- When i = 1101, D1 = 11601. If the input relay 10001 is turned on at this time, the bit patterns of the holding registers 41001 and 41002 are moved to the input relay groups 10601-10632 and the output 1 is turned on.



- When i = 1102, D1 = 11617. If the input relay 10001 is turned on at this time, the bit patterns of the holding registers 41001 and 41002 are moved to the input relay groups 11617-11648 and the output 1 is turned on.



- When i = 1128, D1 = 12033. Because the table size is 2, no input relay groups of destination corresponding to the holding register 41002 exist. Therefore, even if the input relay 10001 is turned on, the move will not be performed and the output 2 is turned on.

4. 8. 3 Block Move 2 with Destination Index (DIBR)

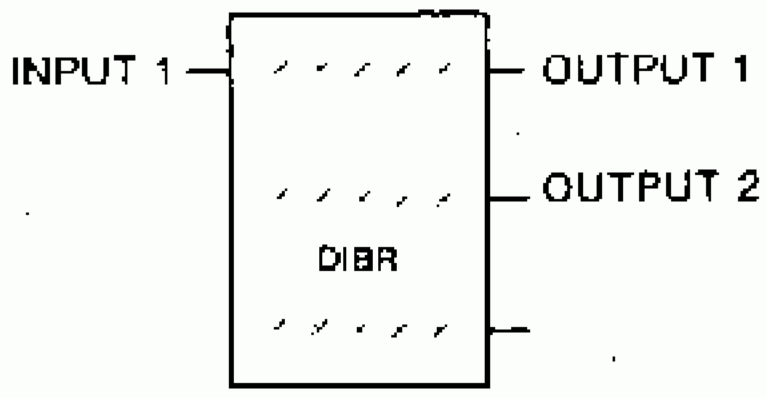
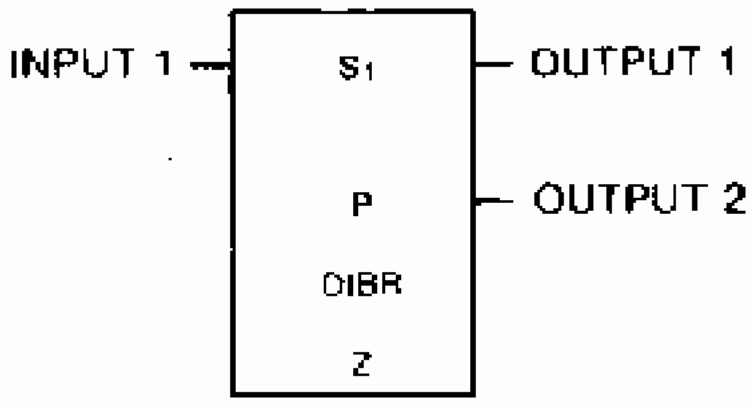


Fig. 4.66 DIBR General Form

(1) Form

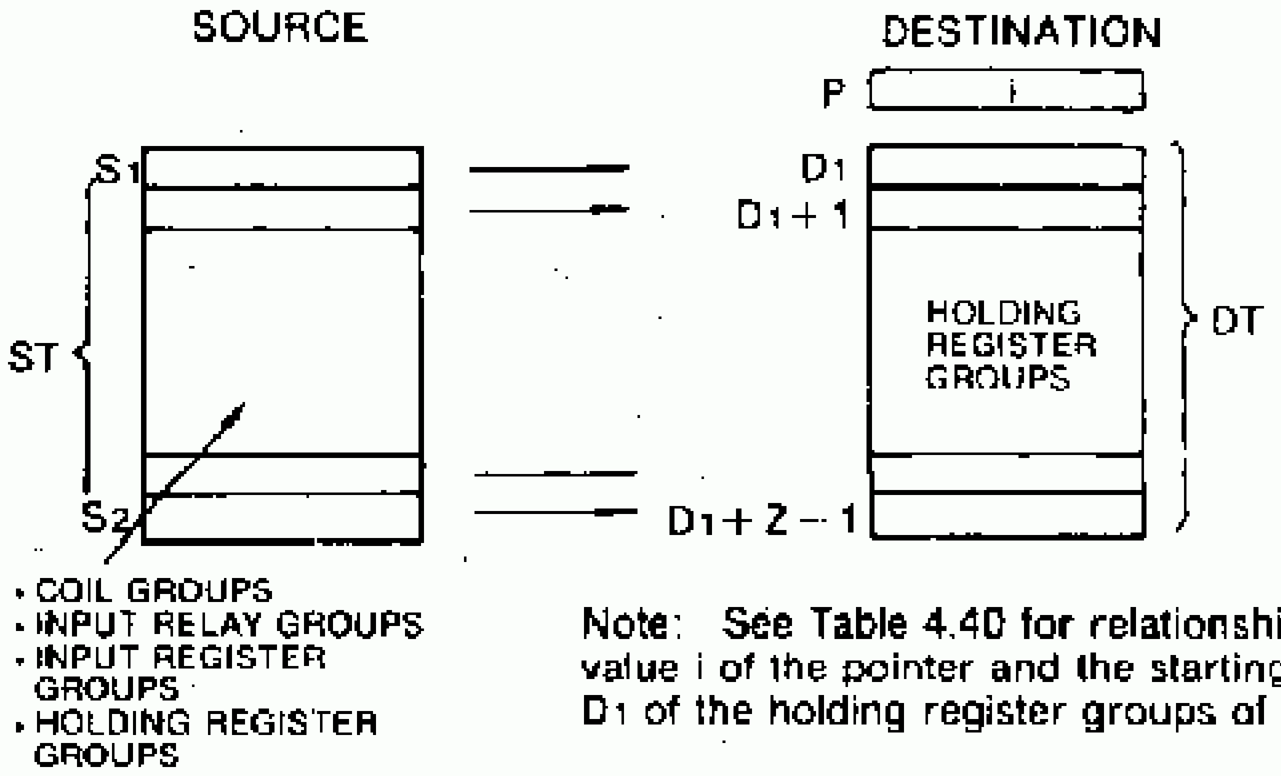
- Fig. 4.66 shows the form of the block move 2 with destination index (DIBR).
- Block move 2 with destination index requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.40, specify any of constant K, reference numbers 0XXXX of a coil, 1XXXX of an input relay, 3XXXX of an input register, and 4XXXX of a holding register for each of the elements.
- DIBR is the symbol denoting the block move 2 with destination index.

(2) Function and Operation



- By the block move 2 with destination index (DIBR), all data of the source table (ST) will be moved to the destination table (DT: holding register groups) specified with the value i of the pointer (P) when the input 1 is ON. The output 1 is turned on. The move will be completed in one scanning cycle.

Fig. 4.67 DIBR Function

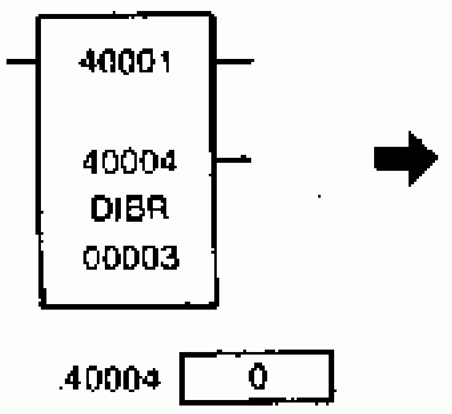


Note: See Table 4.40 for relationship of the value i of the pointer and the starting number D1 of the holding register groups of DT.

- In the following cases, the move will not be executed and the output 2 is turned on.

(a) When the value i of the pointer is out of the range of 1-9999.

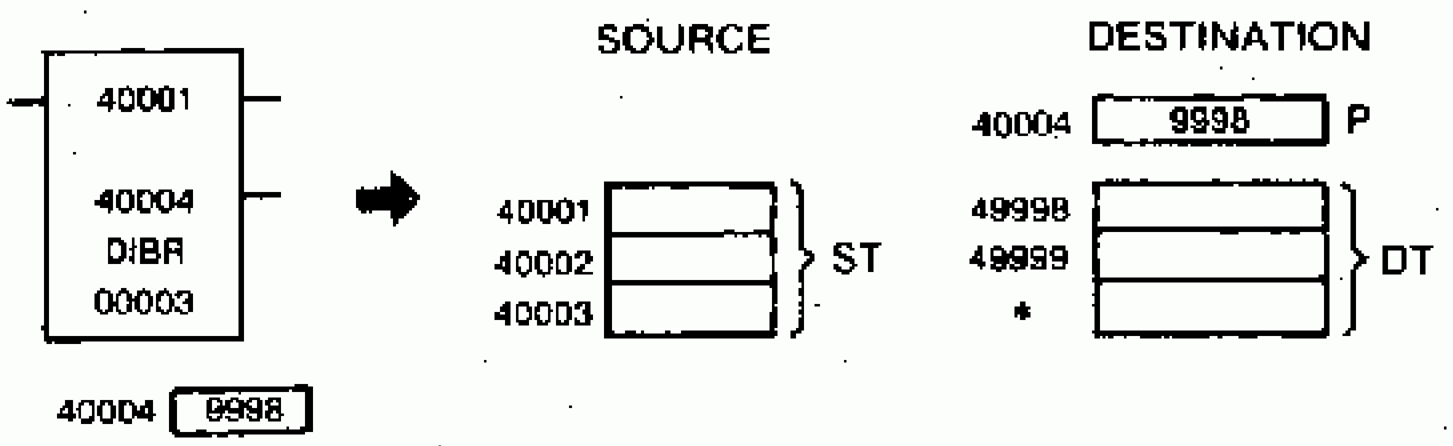
Example



There is no holding register corresponding to i=0. Therefore, no destination tables corresponding to the source tables 40001-40003 exist.

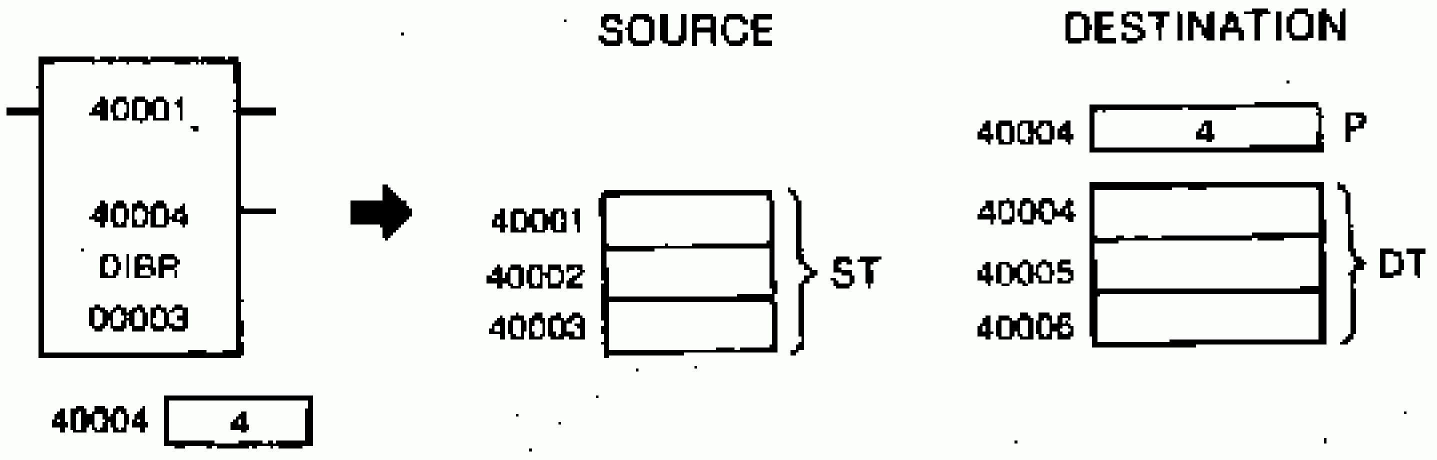
(2) Function and Operation (Cont'd)

(b) When the value i of the pointer is within the range of 1-9999 but no destination table to be specified by the value exists:



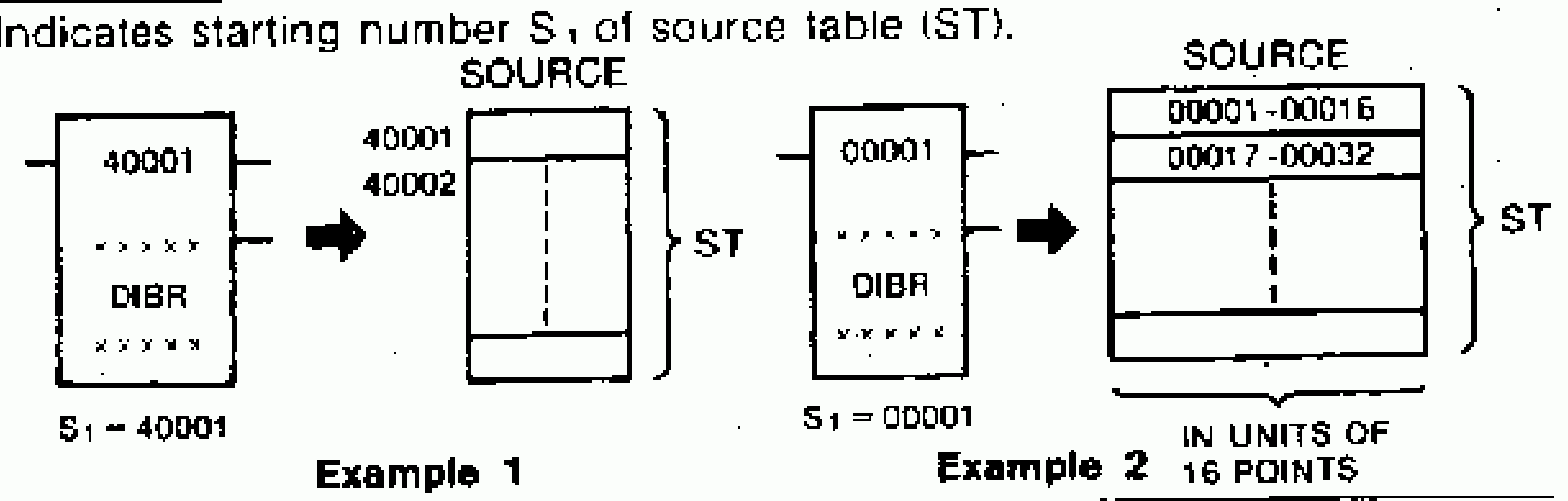
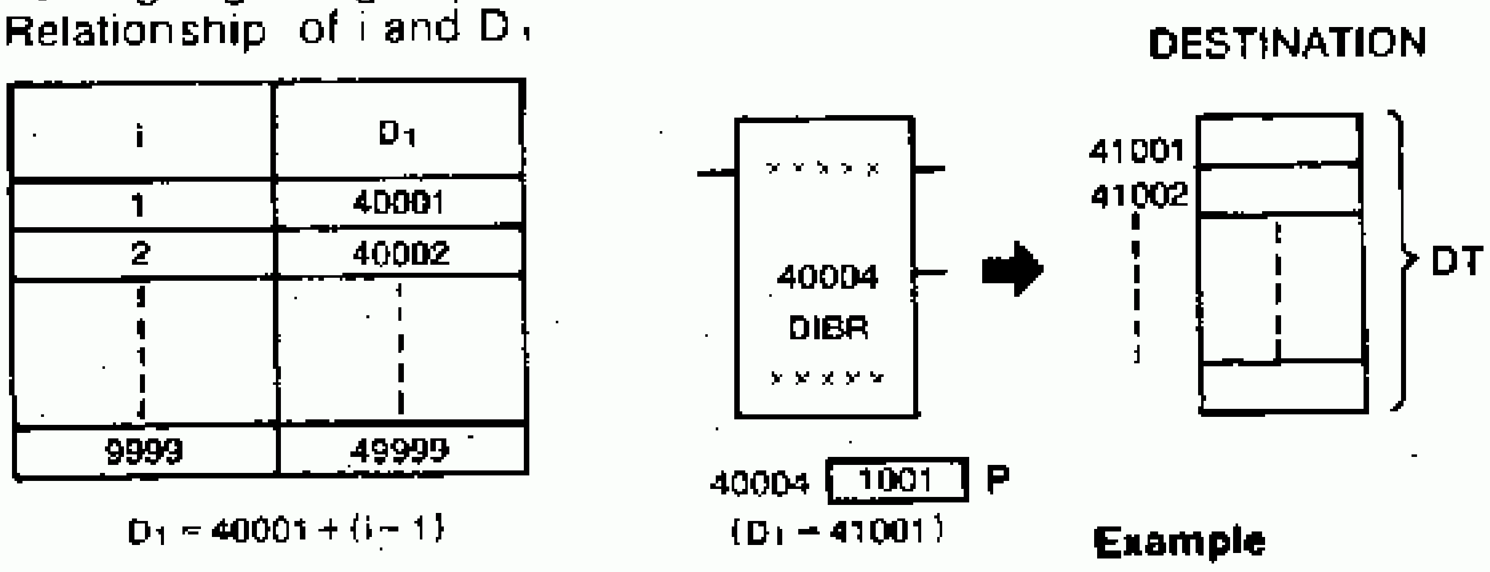
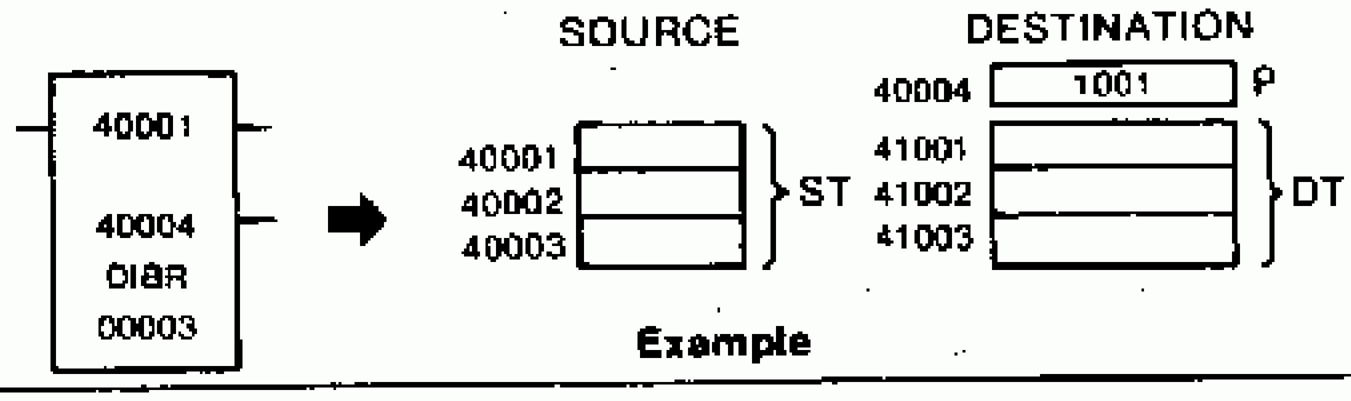
* No destination corresponding to the source 40003 exists.

(c) When the pointer is included in the destination table:



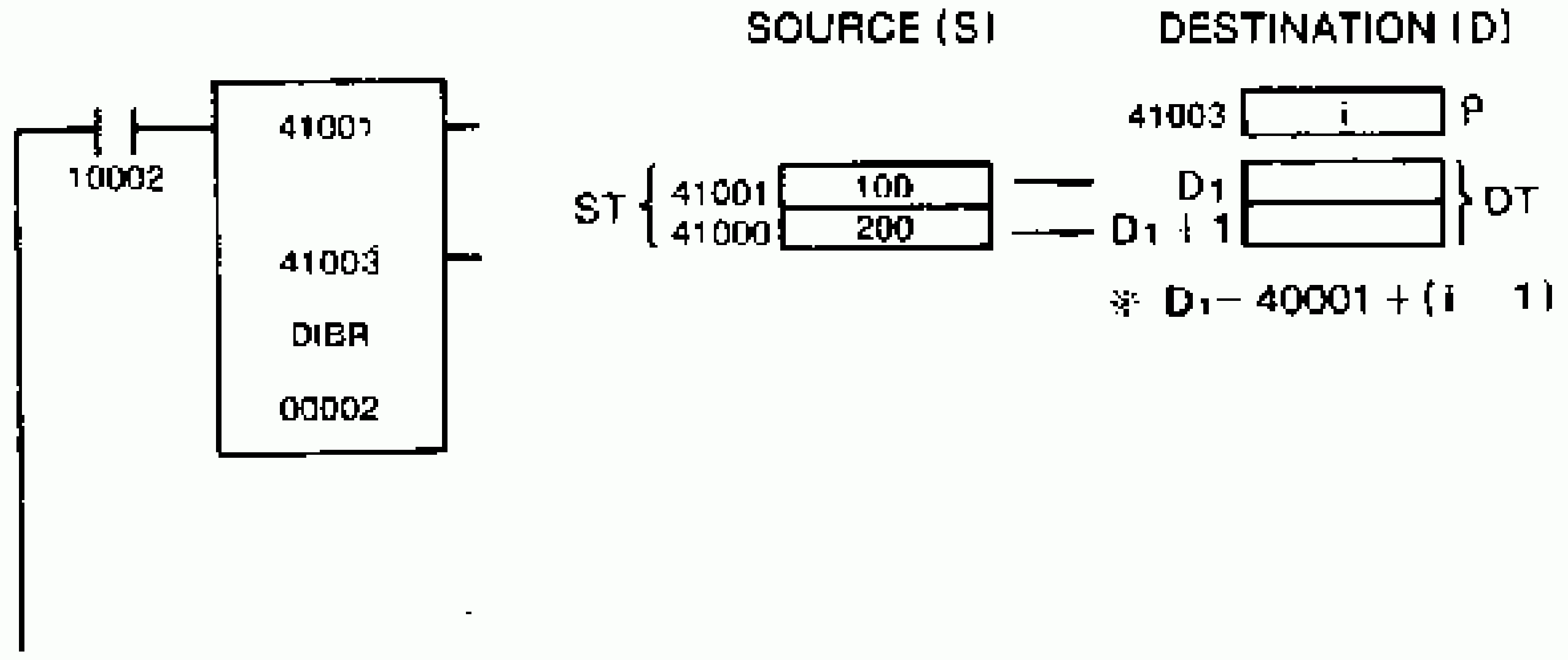
Note: The pointer 40004 is included in the destination table.

Table 4.40 Elements of Block Move 2 with Destination Index

Element	Specified Number	Description
Top	Any of the following: • 0×××× (00001, 00017, 00033, ····· 04081)* • 1×××× (10001, 10017, 10033,····· 12033)* • 3×××× (30001–30256) • 4×××× (40001–49999)	Indicates starting number S_1 of source table (ST). 
Middle	4×××× (40001–49999)	• Indicates pointer (P). It is possible to specify starting number of the destination table (DT: holding register groups) by the value i of the pointer • Relationship of the value i of the pointer and the starting number D_1 of the holding register groups of DT is shown below. • Relationship of i and D_1 
Bottom	Constant K (00001–00100)	Indicates the size of the source table (ST) and the destination table (DT). 

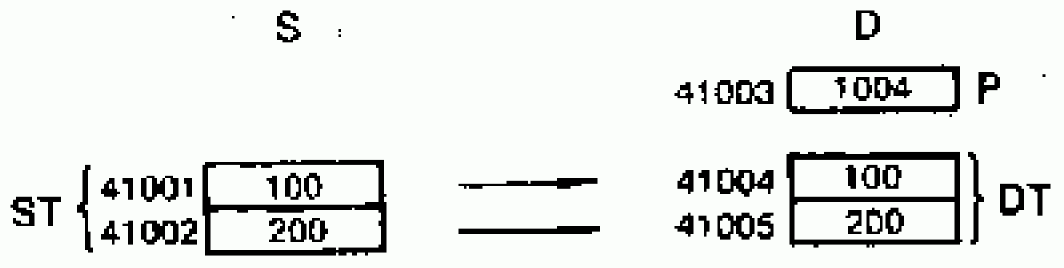
*n=16 m + 1 (m=0, 1, 2, ·····) when n is ×××× of 0×××× or 1××××.

(3) Example

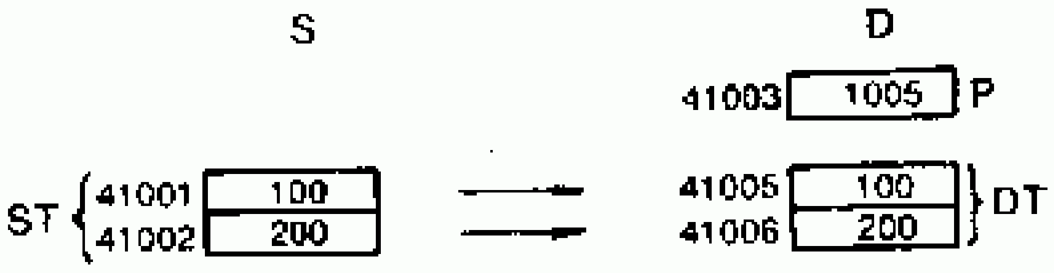


NET #82

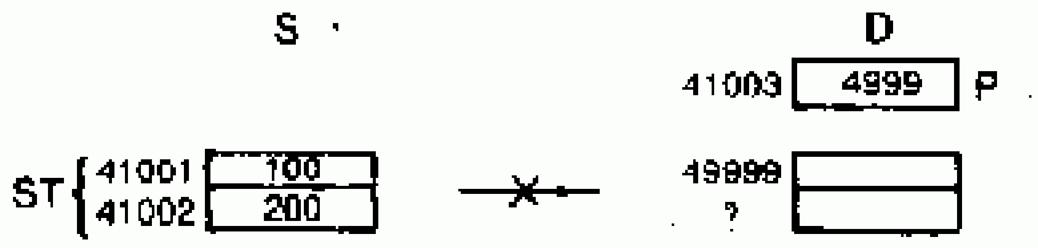
(a) When i = 1004, D1 = 41004. If the input relay 10002 is turned on at this time, the contents of the holding registers 41001 and 41002 is moved to the holding registers 41004 and 41005, and the output 1 is turned on.



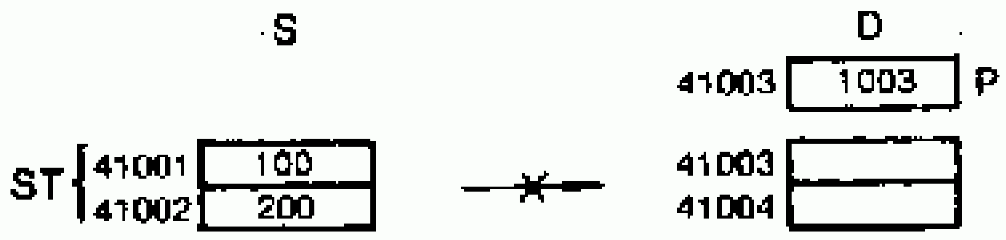
(b) When i = 1005, D1 = 41005. If input relay 10002 is turned on at this time, the contents of the holding registers 41001 and 41002 is moved to the holding registers 41005 and 41006, and the output 1 is turned on.



(c) When i = 9999, D1 = 49999. Because the table size is 2, no holding register of destination corresponding to the source holding register 41002 exists. Therefore, even if the input relay 10002 is turned on, the move will not be performed and the output 2 is turned on.



(d) When i = 1003, D1 = 41003. Because the pointer 41003 is included in the destination table, the move will not be performed and the output 2 is turned on even if the input relay 1002 is turned on.



4.8.4 Block Move 1 with Source Index (SIBT)

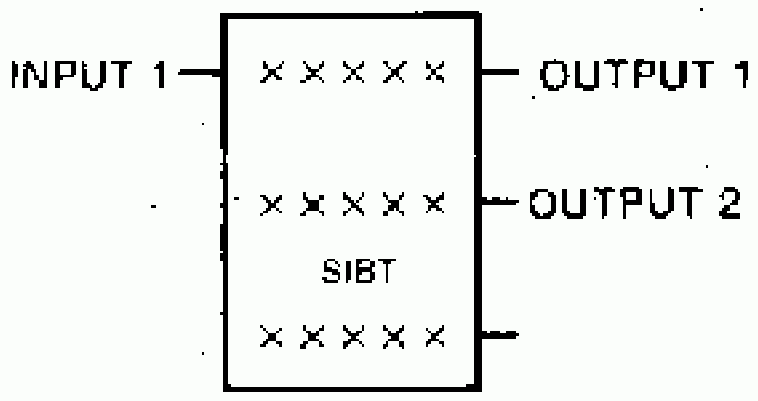
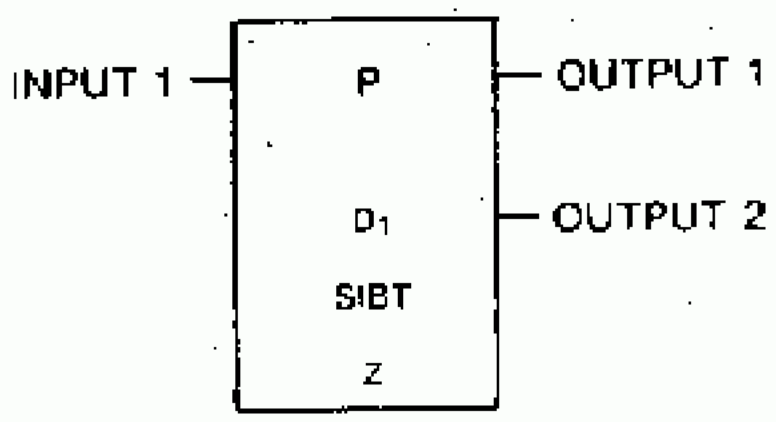


Fig. 4.68 SIBT General Form

(1) Form

Fig. 4.68 shows the form of the block move 1 with source index (SIBT). The Block move 1 with source index requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.41, specify either constant K or reference number 4×××× of a holding register for each of the elements. SIBT is the symbol denoting the block move 1 with source index.

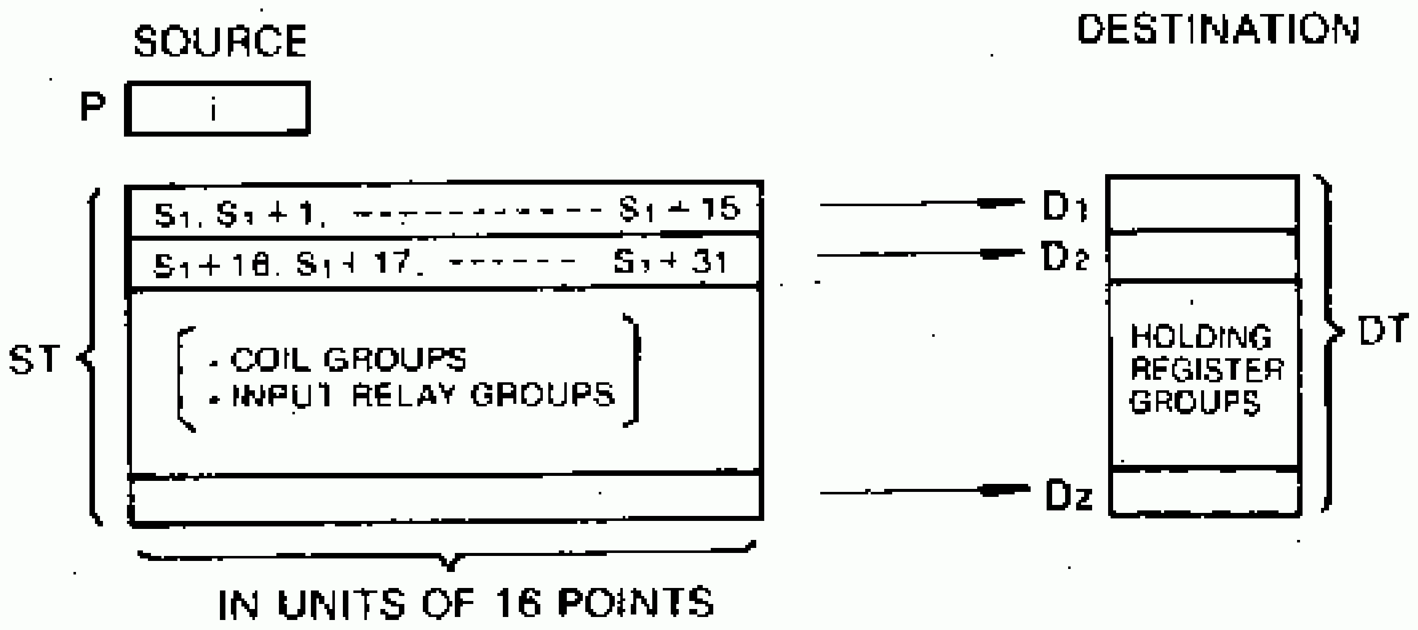
(2) Function and Operation



- By the block move 1 with source index (SIBT), all data of the source table (ST: coils, input relays, input register groups) specified by the value i of the pointer (P) will be moved to the destination table (DT: holding register groups) when the input 1 is ON. The output 1 is turned on. The move will be completed in one scanning cycle.

(a) When the source table is coil and input relay groups:

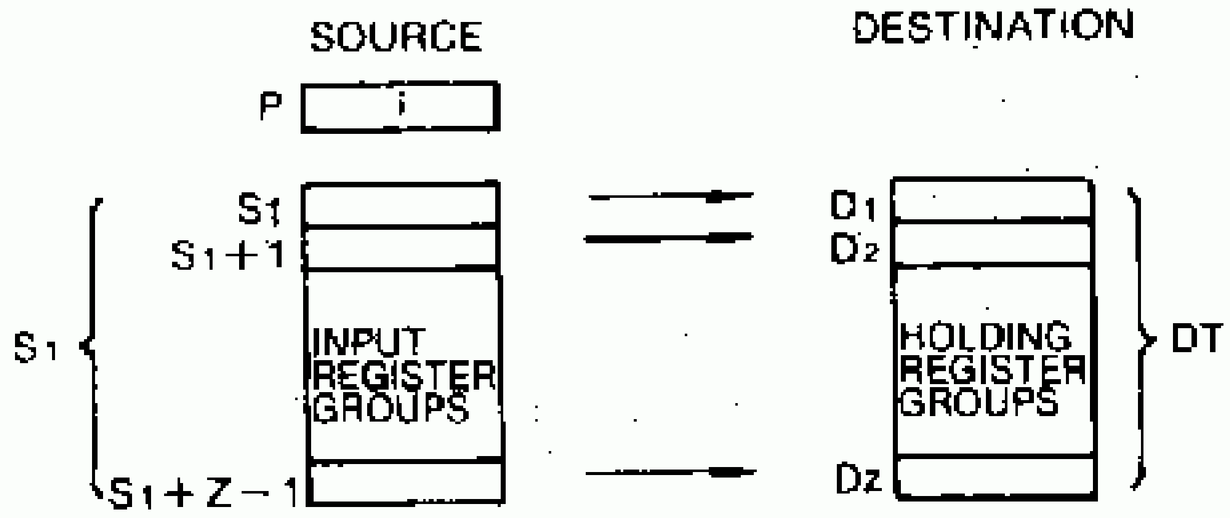
Fig. 4.69 SIBT (Move from 0×××× or 1××××) Function



Note: See Table 4.41 for relationship of the value i of the pointer and the starting number S_1 of the coil groups and input relay groups of ST.

(b) When the source table is input register groups:

Fig. 4.70 SIBT (Move from 3××××) Function

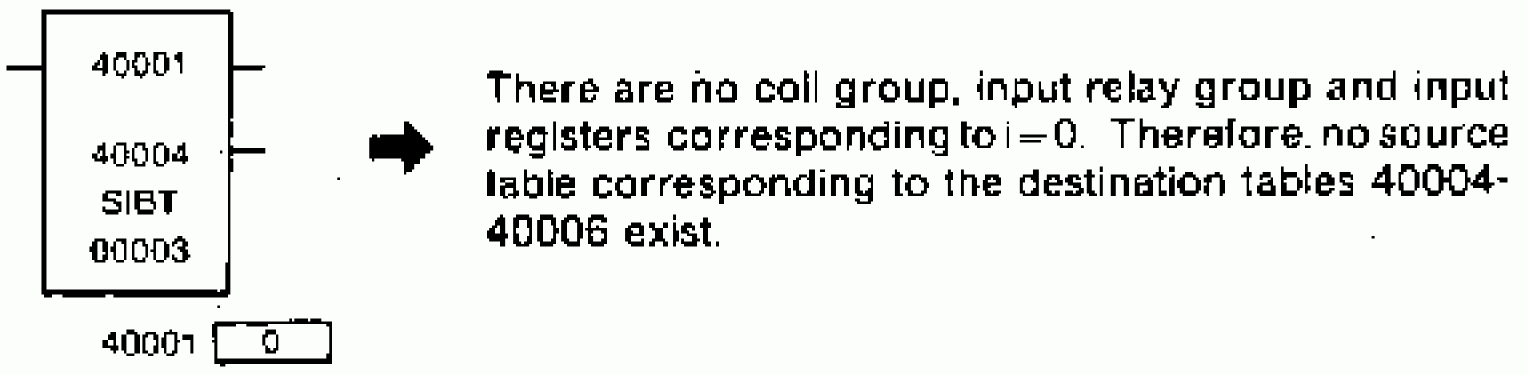


Note: See Table 4.41 for relationship of the value i of the pointer and the starting number S_1 of the input register groups of ST.

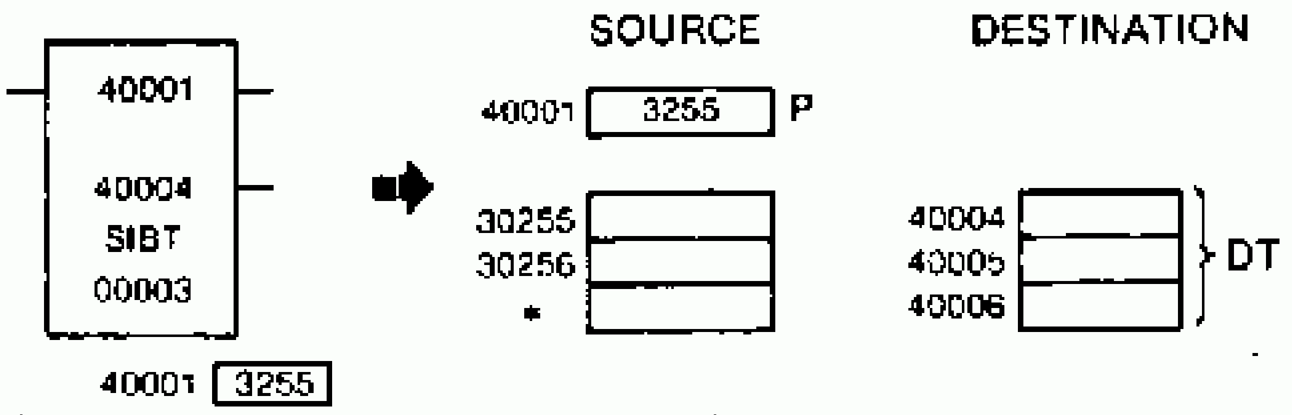
• In the following cases, the move will not be executed and the output 2 is turned on.

(a) When the value i of the pointer is out of the ranges of 1-256, 1001-1128, and 3001-3256:

Example

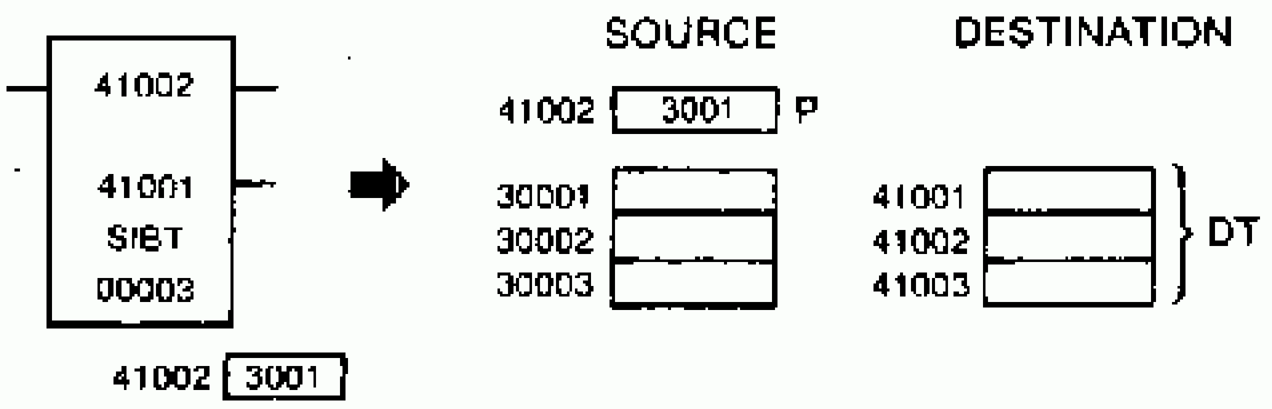


(b) When the value i of the pointer is within the ranges of 1-256, 1001-1128, or 3001-3256 but no source table specified by the value exists:



* No source input register corresponding to the destination 40006 exists.

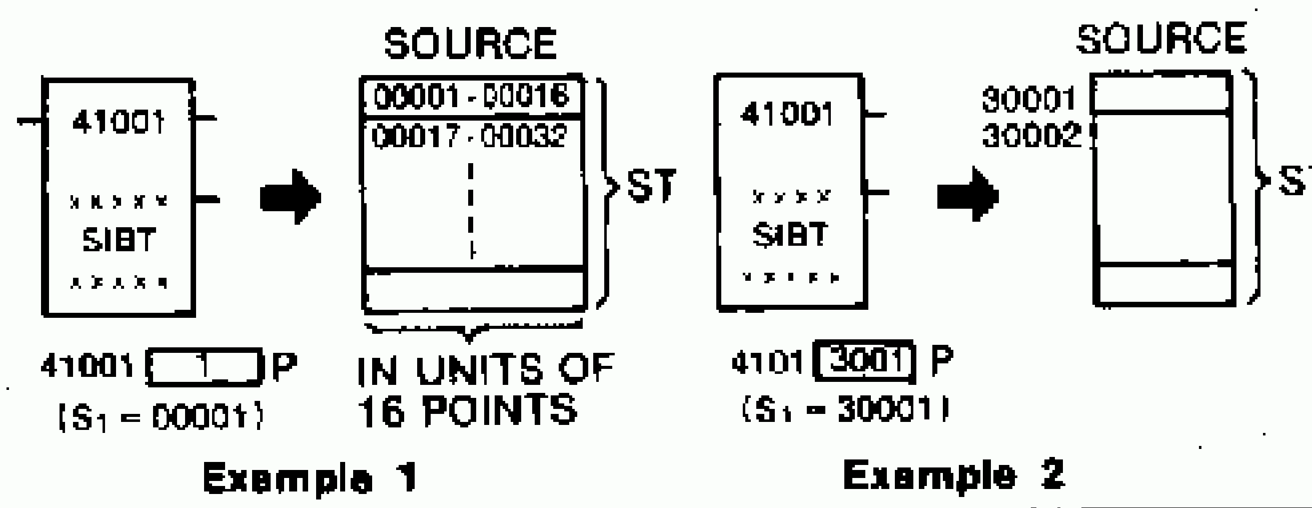
(c) When the pointer is included in the destination table:



Note: The pointer 41002 is included in the destination table.

Table 4.41 Elements of Block Move 1 with Source Index

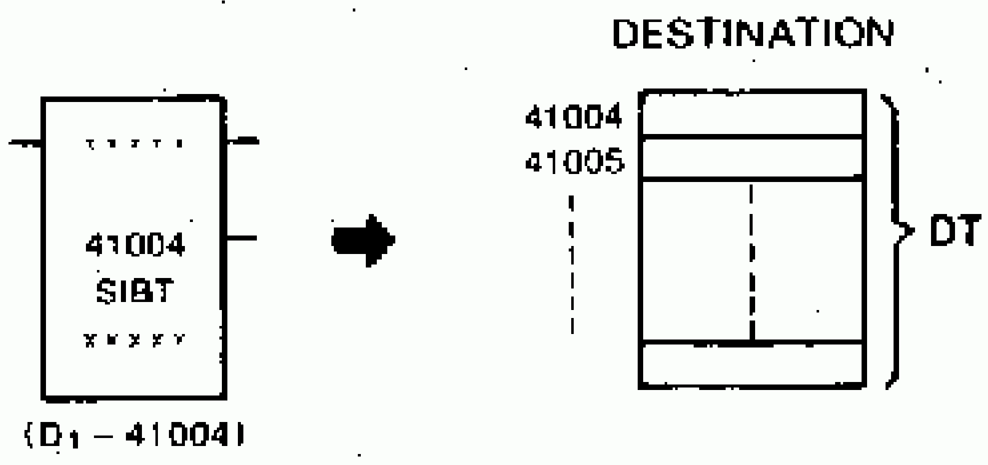
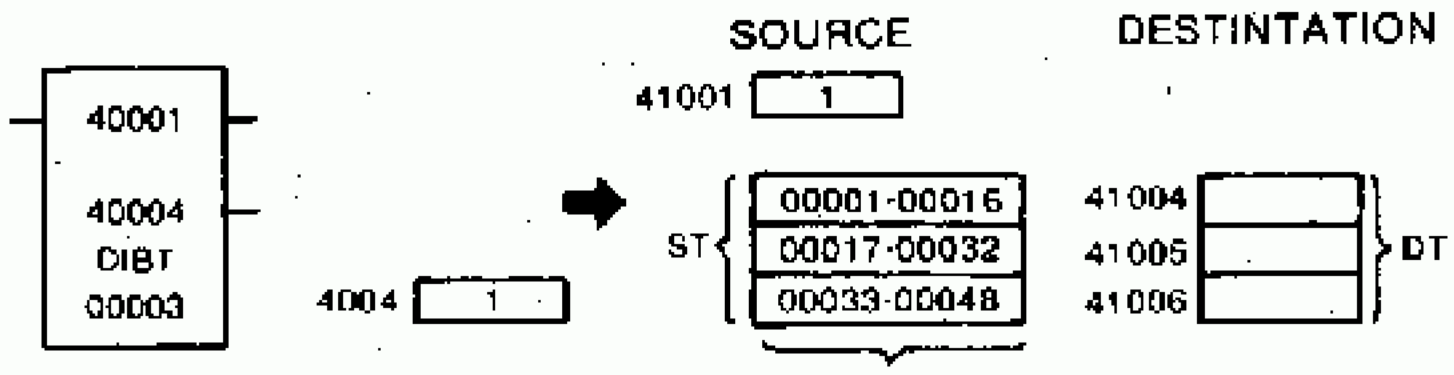
<table>
<tr><th>Element</th><th>Specified Number</th><th>Description</th></tr>
<tr><td>Top</td><td>4××××(40001−49999)</td><td>
• Indicates pointer (P). It is possible to specify starting number of the source table (ST: coil, input relay and input register groups) by the value i of the pointer.
• Relationship of the value i of the pointer and the starting number S_1 of coil, input relay and input register groups are shown below.
• Relationship of i and S_1
<table>
<tr><th>i</th><th>S_1</th><th>i</th><th>S_1</th><th>i</th><th>S_1</th></tr>
<tr><td>1</td><td>00001</td><td>1001</td><td>10001</td><td>3001</td><td>30001</td></tr>
<tr><td>2</td><td>00017</td><td>1002</td><td>10017</td><td>3002</td><td>30002</td></tr>
<tr><td>⋮</td><td>⋮</td><td>⋮</td><td>⋮</td><td>⋮</td><td>⋮</td></tr>
<tr><td>256</td><td>04081</td><td>1128</td><td>12033</td><td>3256</td><td>30256</td></tr>
</table>
$S_1 = 1 + 16(i - 1)$ $S_1 = 10001 + 16(i - 1001)$ $S_1 = 30001 + (i - 3001)$



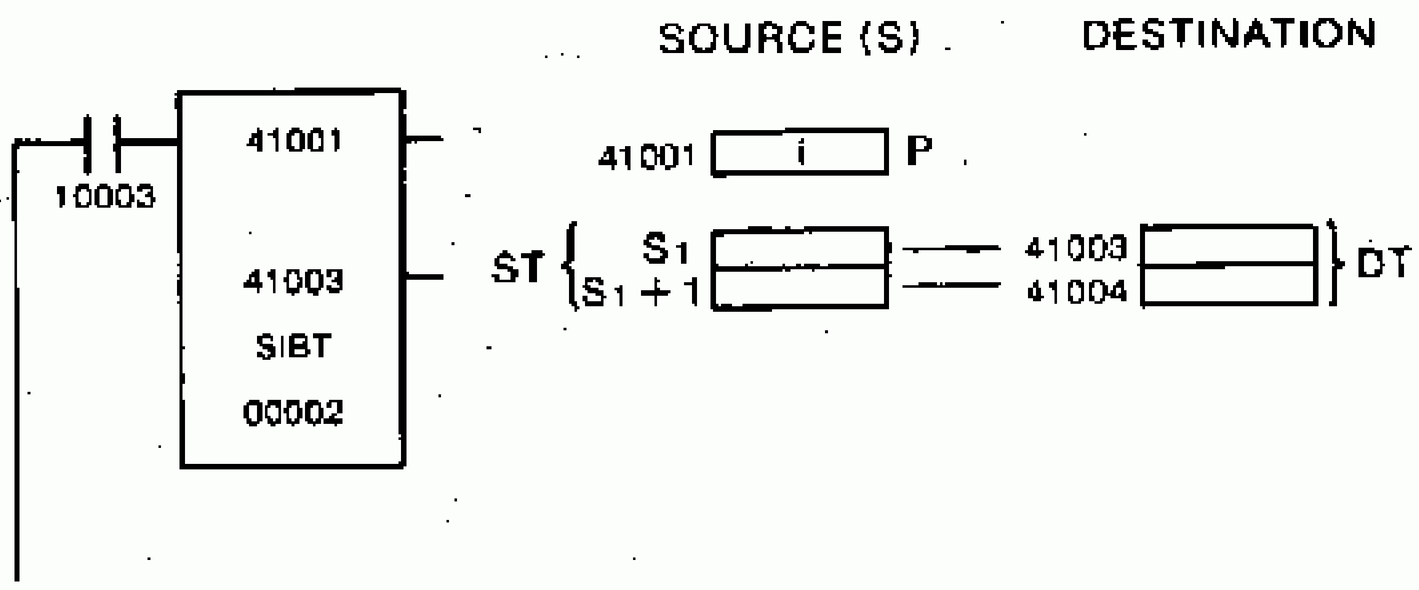
</td></tr>
</table>

(2) Function and Operation (Cont'd)

Table 4.41 Elements of Block Move 1 with Source Index (Cont'd)

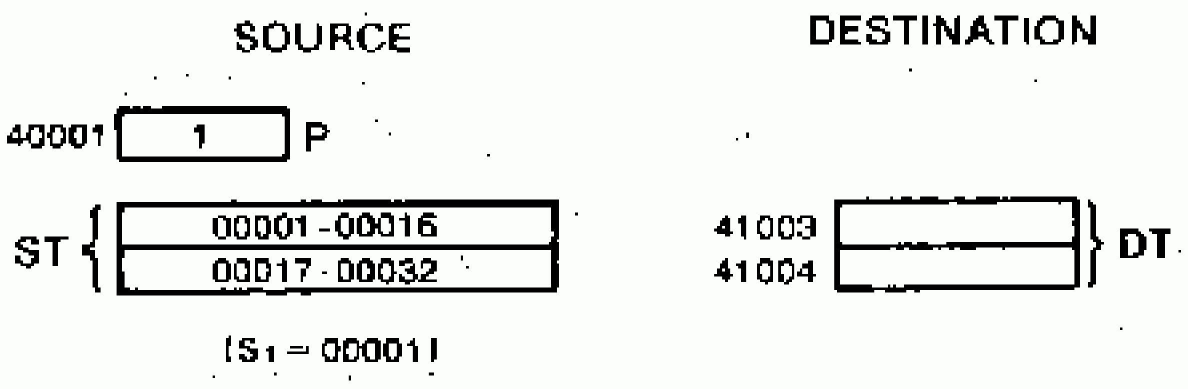
Element	Specified Number	Description
Middle	4 × × × × (40001–49999)	Indicates the starting number D_1 of the destination table (DT: holding register groups).  Example
Bottom	Constant K (00001–00100)	Indicates the size of the source table (ST) and the destination table (DT).  Example

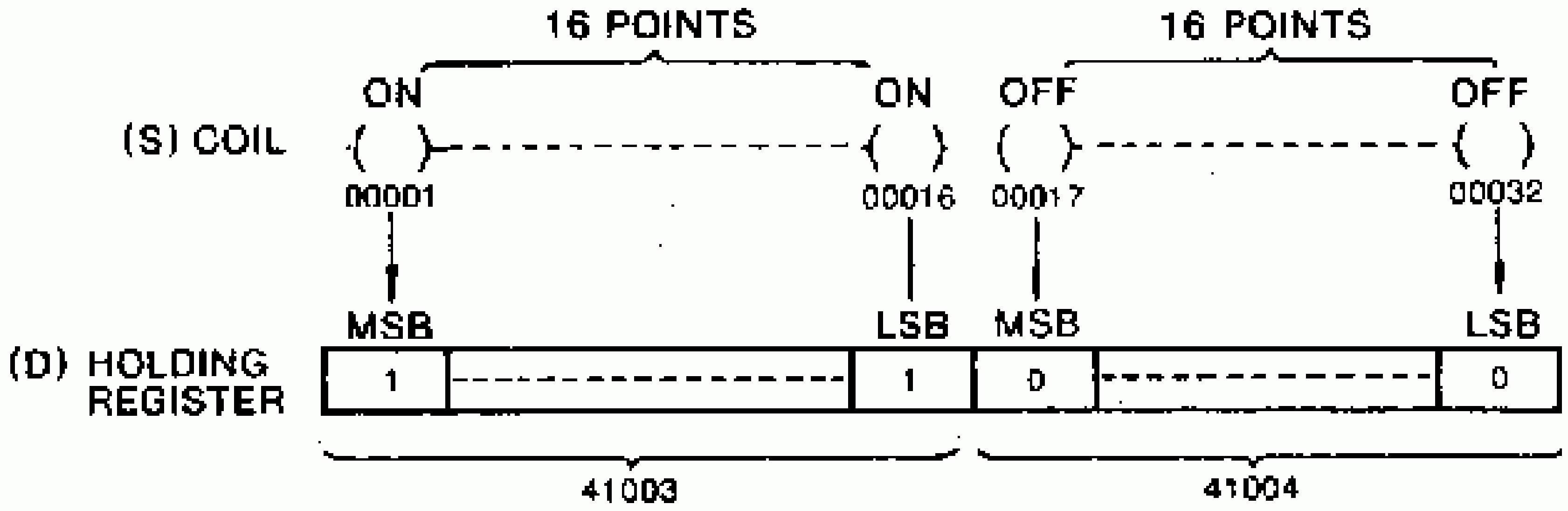
(3) Example



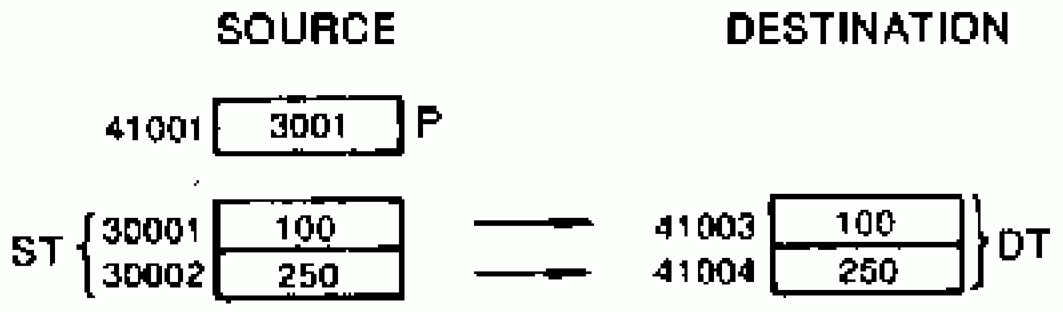
NET #63

(a) When i = 1, S1 = 00001. If the input relay 10003 is turned on at this time, the ON/OFF status of the coil groups 00001-00032 is moved to the holding registers 41003 and 41004 and the output 1 is turned on.

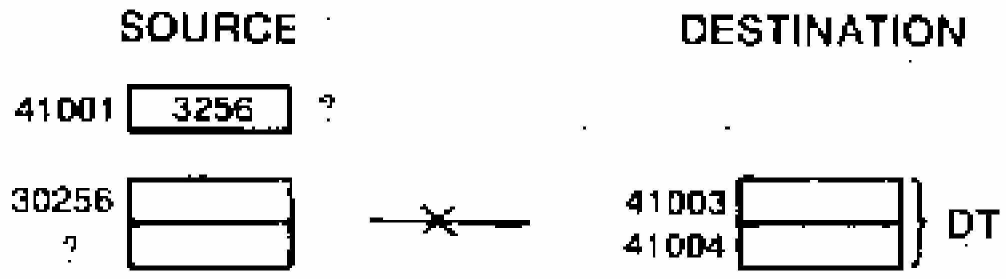




(b) When i = 3001, S1 = 30001. If the input relay 10003 is turned on at this time, the contents of the input registers 30001 and 30002 is moved to the holding registers 41003 and 41004, respectively.



(c) When i = 3256, S1 = 30256. Because the table size is 2, no source input register corresponding to the destination holding register 41004 exists. Therefore, even if the input relay 10003 is turned on, the move will not be performed and the output 2 is turned on.



Note: No source input register corresponding to the destination 41004 exists.

4.8.5 Block Move 2 with Source Index (SIBR)

(1) Form

- Fig. 4.71 shows the form of block move 2 with source index (SIBR).

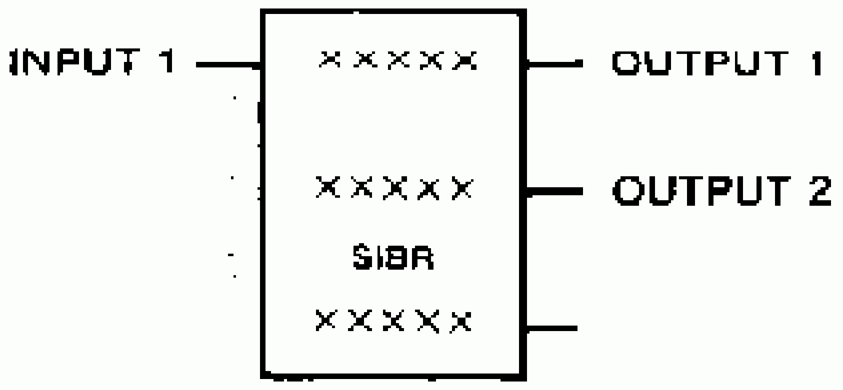
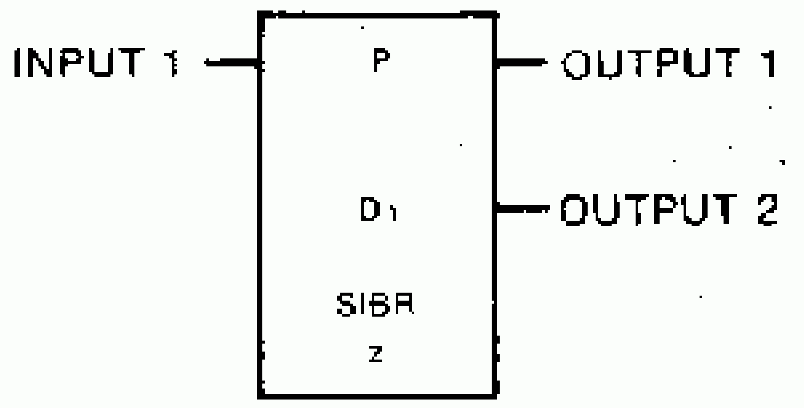


Fig. 4.71 SIBR General Form

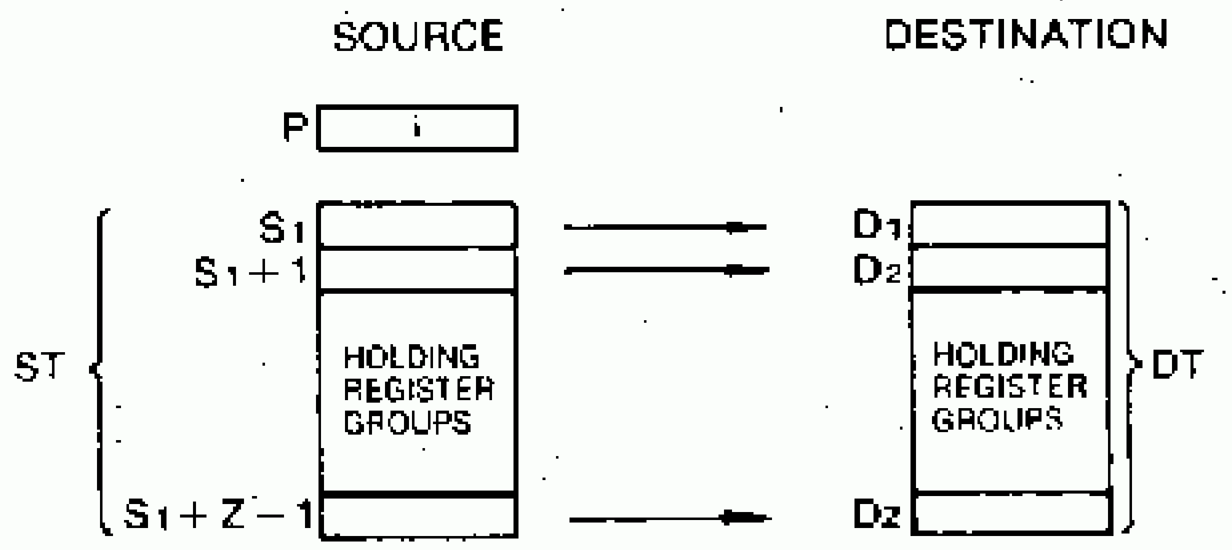
- The block move 2 with source index requires three elements placed vertically (top, middle, and bottom). Referring to Table 4.42, specify either constant K or reference number 4XXXX of a holding register for each of the elements.
- SIBR is the symbol denoting the block transfer 2 with source index.

(2) Function and Operation



- By the block move 2 with source index (SIBR), all data of the source table (ST: holding register groups) specified by the value i of the pointer (P) will be moved to the destination table (DT: holding register groups) when the input 1 is ON. The output 1 is turned on. The move will be completed in one scanning cycle.

Fig. 4.72 SIBR FUNCTION



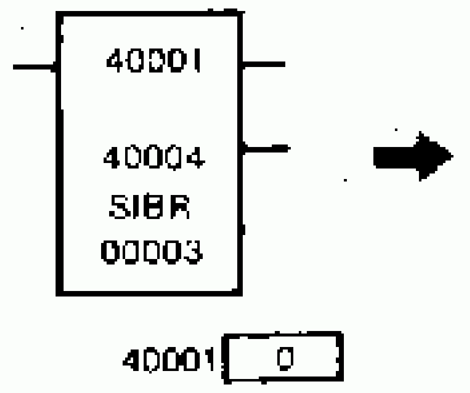
NOTE

See Table 4.42 for relationship of the value i of the pointer and the starting number S1 of the holding register groups of ST.

- In the following cases, the move will not be executed and the output 2 is turned on.

(a) When the value i of the pointer is out of the range of 1-9999:

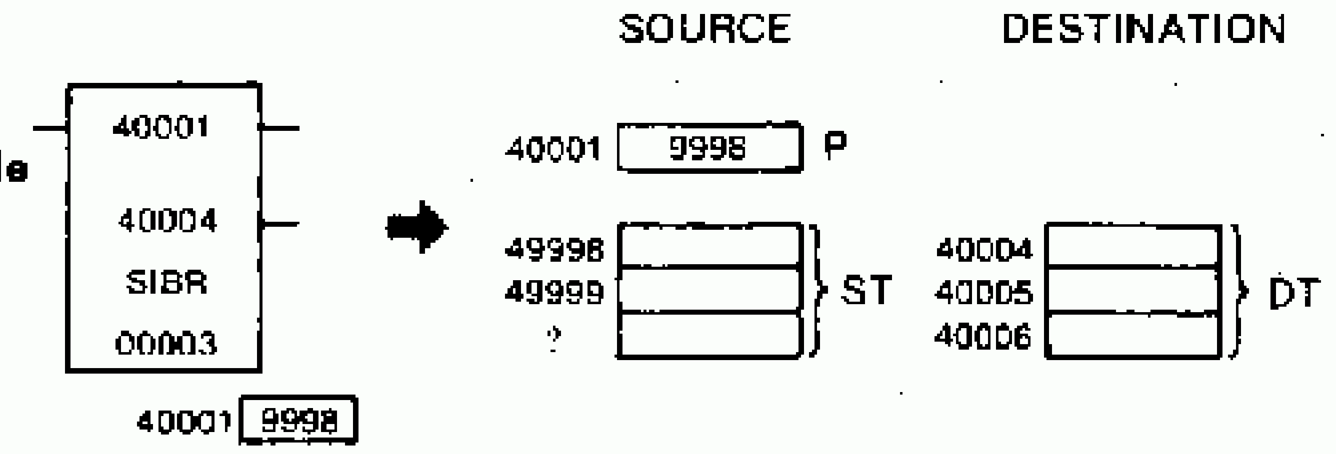
Example



No holding register corresponding to i=0 exists and therefore no source table corresponding to the destination table 40004-40006 exists.

(b) When the value i of the pointer is within the range of 1-9999 but no source table specified by the value exists:

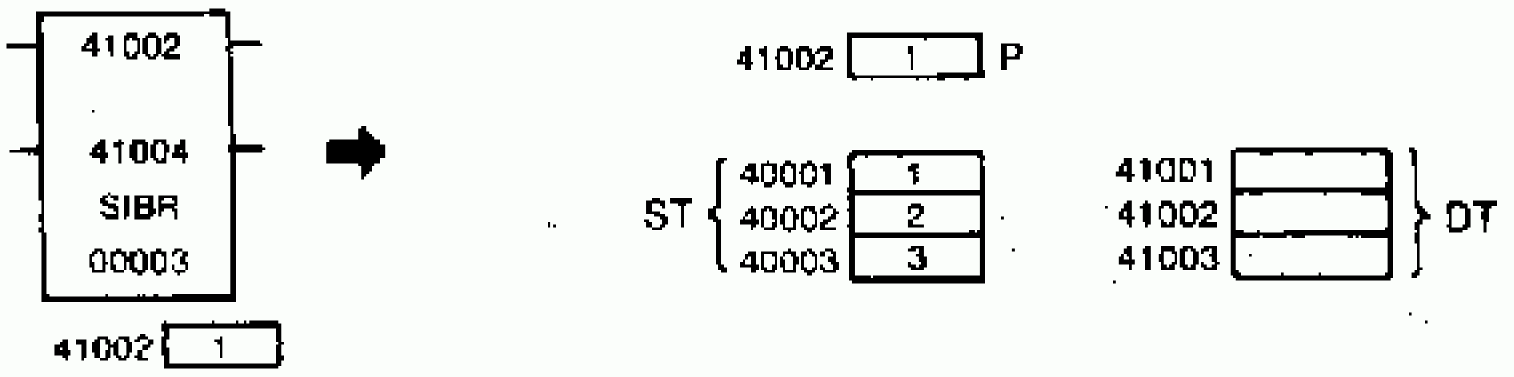
Example



Note: No source holding register corresponding to the destination 40006 exists.

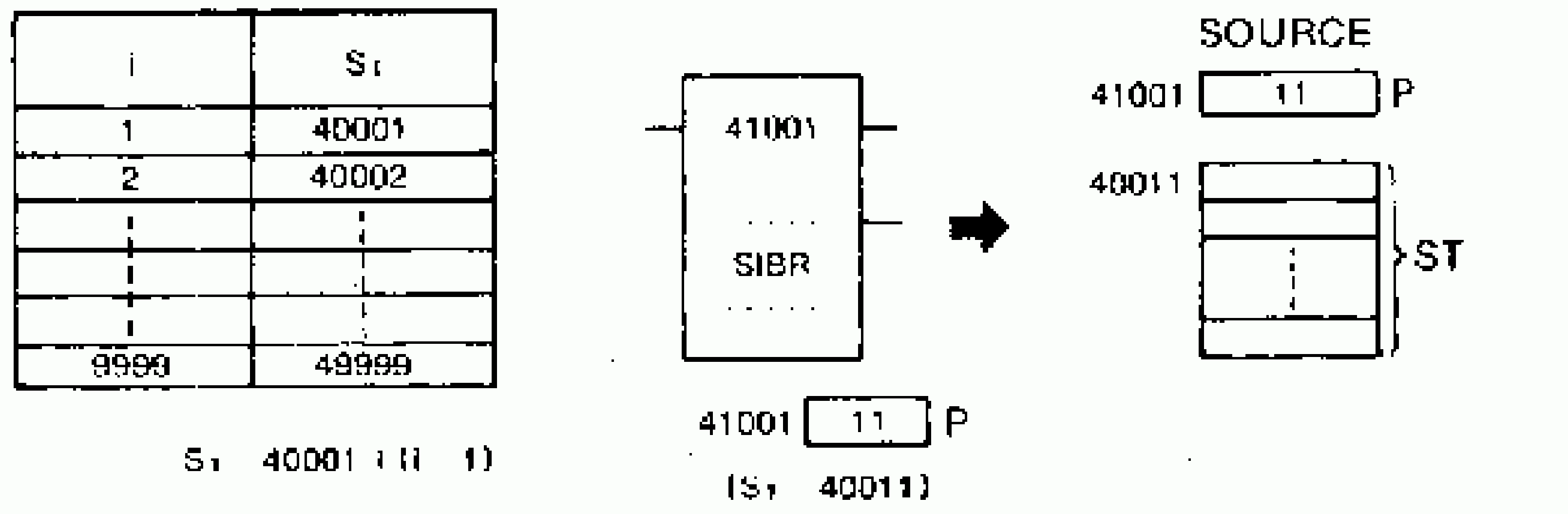
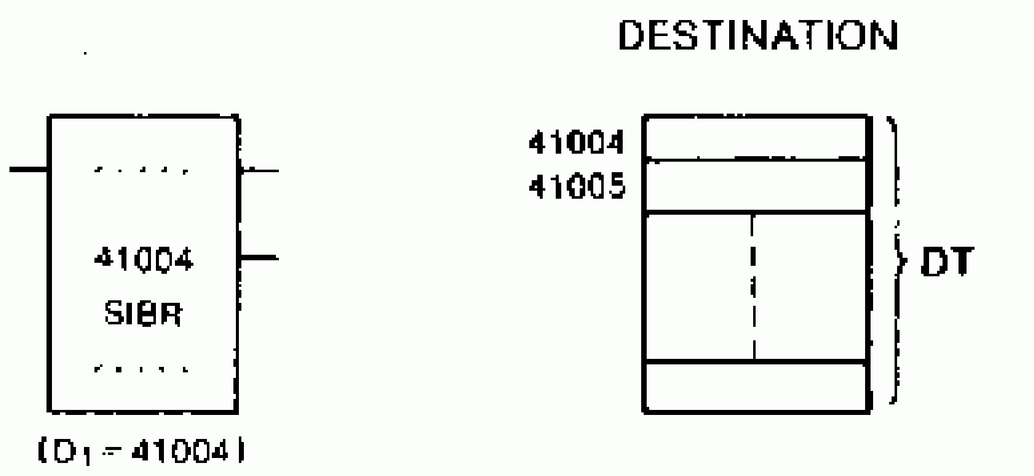
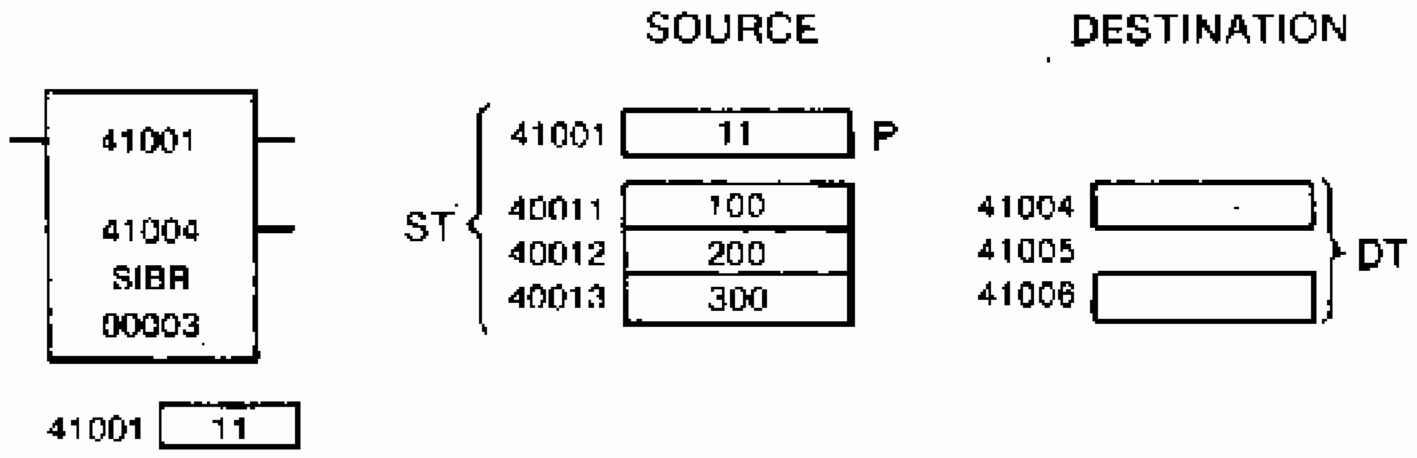
- When the pointer is included in the destination table:

Example

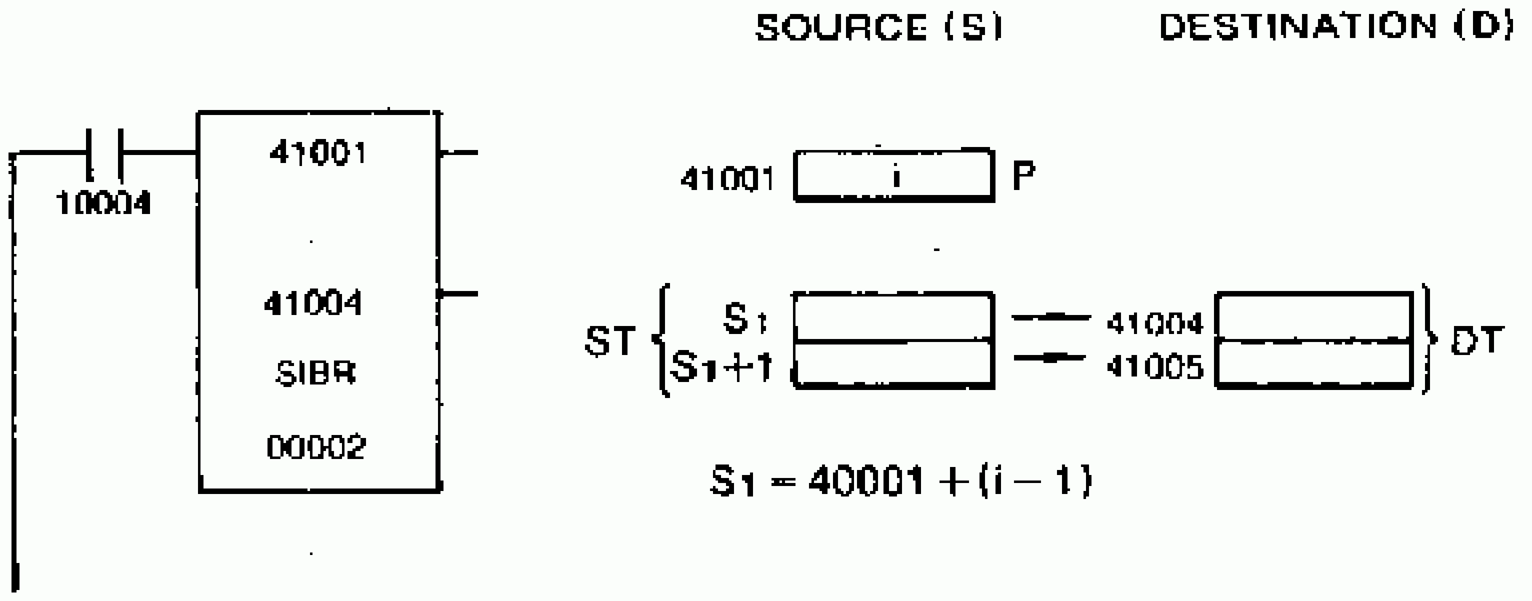


Note: The pointer 41002 is included in the destination table.

Table 4. 42 Elements of Block Move 2 with Source Index

Element	Specified Number	Description
Top	4 × × × × (40001–49999)	• Indicates pointer (P). It is possible to specify starting number of the source table (ST: holding register groups). • Relationship of the value i of the pointer and the starting number S_1 of the holding register groups is shown below • Relationship of i and S_1  S_1 = 40001 + (i − 1) Example
Middle	4 × × × × (40001–49999)	Indicates the starting number D_1 of the destination table (DT: holding register groups).  Example
Bottom	Constant K (00001–00100)	Indicates the size of the source table (ST) and the destination table (DT).  Example

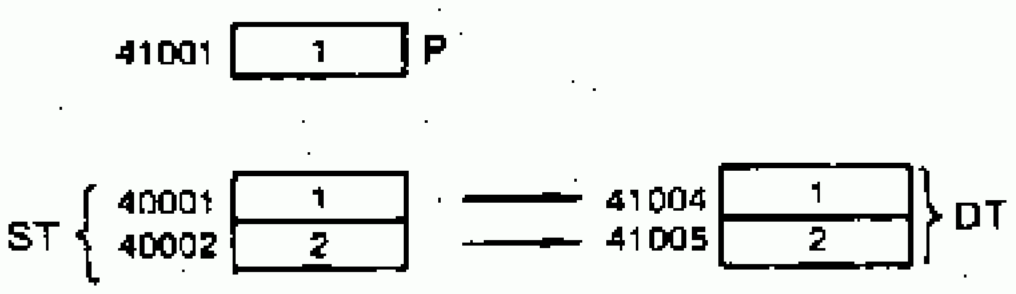
(3) Example



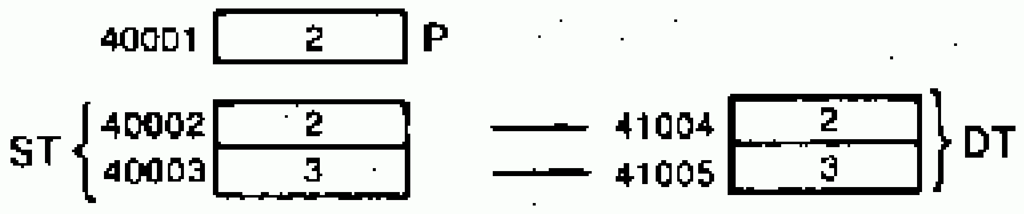
NET #84

• When i = 1, S1 = 40001. If the input relay 10004 is turned on at this time, the contents of the holding registers 40001 and 40002 is moved to the holding registers 40001 and 40002 is moved to the holding registers 41004 and 41005, respectively and the output 1 is turned on.

(3) Example (Cont'd)

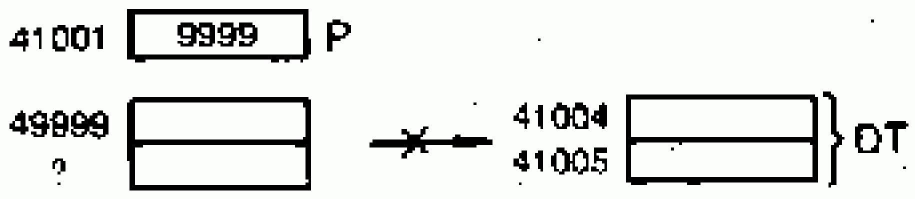


- When i = 2, S1 = 40002. If the input relay 10004 is turned on at this time, the block move will be performed as follows.



(S_1= 40002)

- When i = 9999, S1 = 49999. Because the table size is 2, no holding register corresponding to the destination 41005 exists. Therefore, even if the input relay 10004 is turned on, the move will not be performed and the output 2 is turned on.



(S_1 49999)

Note: No source holding register corresponding to the destination 41005 exists.

4. 8. 6 Programming Indexed Block Move Circuit and Precautions

(1) Inputs to the indexed block move circuit may be outputs of relays, timers, counters, arithmetic operations, data transfer matrixes, and other indexed block move circuits.

- Coils need not be connected to two output nodes, (1 and 2) of an indexed block move circuit. It is permitted to connect a relay contact to the output nodes at right or connect the output node directly to an input node of an arithmetic circuit, except relays.
- To execute the move constantly, connect the input directly to the power rail at left. To execute it only during one scanning cycle, use a transitional contact as an input.
- The range of the source or destination table specified by a table size must be within the range of the reference numbers of input relays, coils, or registers.
- It is possible to OR the outputs by connecting a vertical shunt element.
- The source or the pointer remain unchanged data or values after the move.
- Be sure to disable the input relay groups used as destination of the block move 1 with destination index (DIBT) (DISABLE ON or DISABLE OFF). There are following differences between the cases the input relay groups are disabled and enabled.

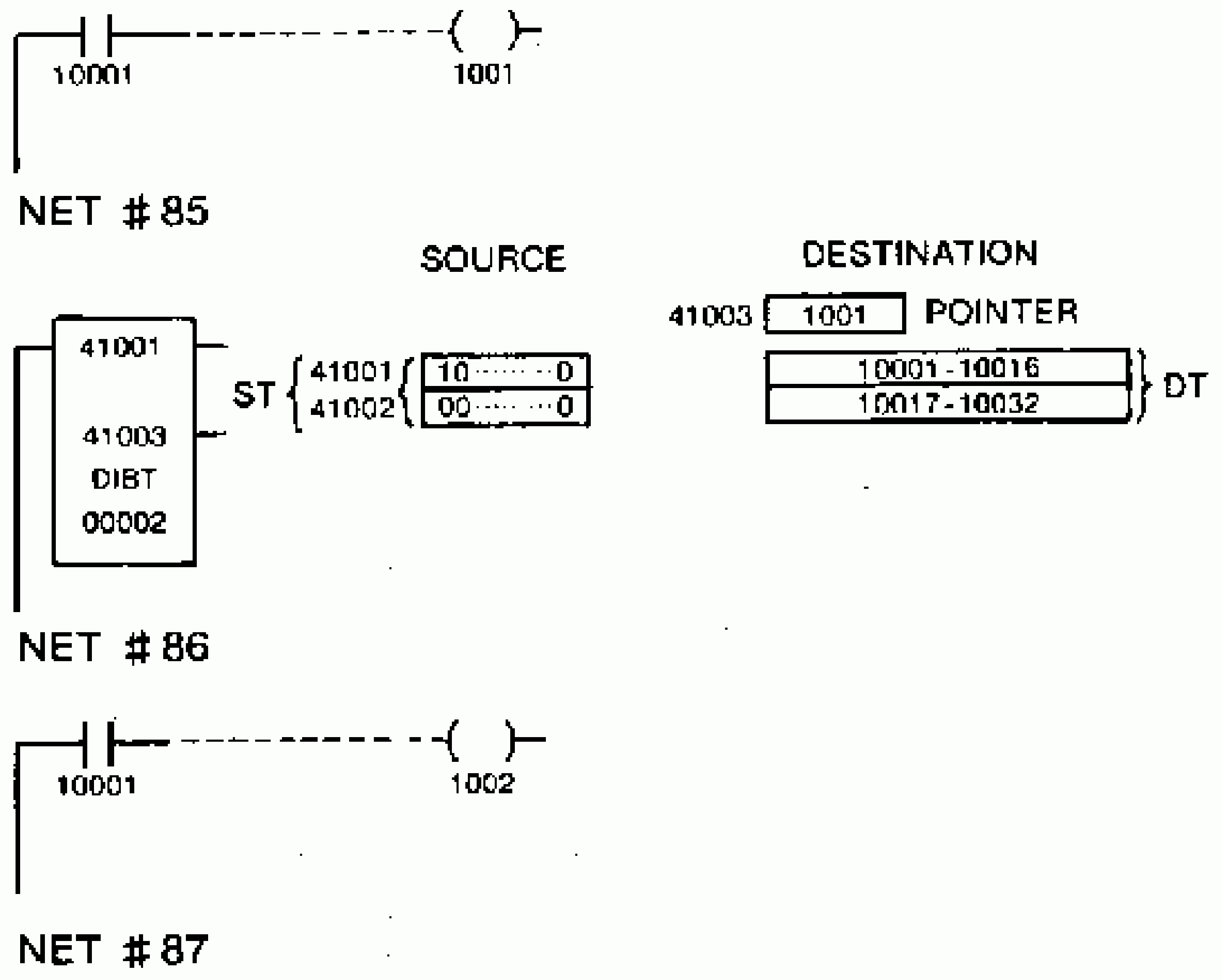
(a) When the input relay groups are disabled:

The input relays are turned on and off according to the results of execution of DIBT.

(b) When the input relay groups are enabled:

The input relays are turned on and off according to the results of execution of DIBT then all of them are turned off at the beginning of the next scanning cycle. But, if an input module is connected and I/O allocation is made, they are turned on and off according to the actual status of input signals at the beginning of the next scanning cycle.

Example



NOTE

If the input relay 10001 is disabled, the each normally open (NO) contact of the input relay 10001 of NET #85 and #87 is turned on. If the input relay 10001 is enabled, the NO contact of the input relay 10001 of NET #85 is turned off and that of NET #87 is turned on.

4.9 MATRIX

This function group allows matrices to be built in consecutively numbered registers. These matrices are similar to tables previously discussed in that they can be groups of input registers or holding registers depending upon the application requirements. In fact, these same registers can also be operated upon by Move functions. Whereas the Move functions operate upon individual registers as elements of tables, the matrix function will operate upon bit patterns within the matrix. Since all registers contain 16 bits, the size of a matrix in bits will be even multiples of 16 (e.g., 32, 48, 64, 80, etc.). A bit can have one of two states: ON (one) or OFF(zero). Bits within a matrix each have their own identification as illustrated in Fig. 4.73. A matrix of 100 registers contains 1600 bits.

	MSB	LSB
41001	1. 2 3. ------------------	. 16
41002	17 18 ------------------	. 32
41003	33. ------------------	. 48
41004	49. ------------------	. 64

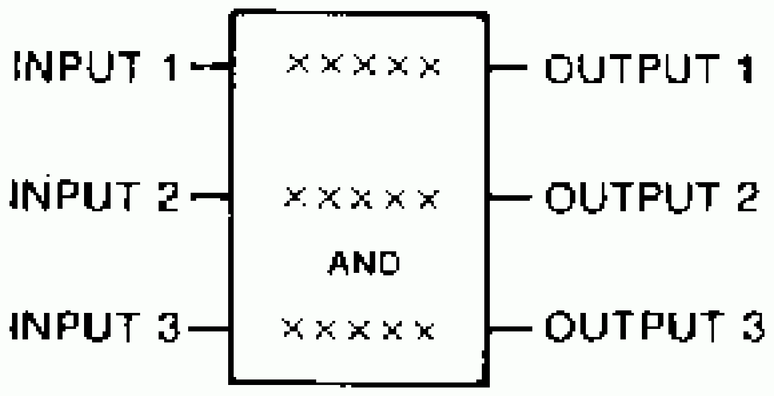
Fig. 4.73 Sample Matrix Bit Numbering

4.9.1 Types of Matrix

Table 4.43 Types of Matrix

Type	Symbol	Reference Page
Logical AND	AND	130
Logical OR	OR	130
Logical Exclusive OR	XOR	130
Logical Complement	COMP	134
Logical Compare	CMPR	134
Logical Bit Modify	MBIT	137
Logical Bit Sense	SENS	138
Logical Bit Rotate	BROT	140
Logical Multiple Bit Rotate	MROT	142
Logical Byte Rearrangement	TWST	144

4.9.2 Form of Matrix



The matrix requires three elments placed vertically (top, middle, and bottom), like arithmetic operations and data move. It can be used at any intersection of the 7 lines-by-10 columns matrix (except that the top element cannot be located on lines 6 and 7). AND written between the middle and bottom elements indicates the type of martix.

NOTE

Byte interchange (TWST) alone uses two elements (top and bottom places).

(1) Elements and Their Meanings

The top element is called source and the middle destination. The bottom indicates the size of the matrix table. Each has the same meaning as in the data move.

(2) Reference Number

(a) Numbers specified as reference numbers depend on the type of matrix. See Table 4.44.

(b) The content of a register (16-bit binary) for the register or the ON/OFF status for discrete signals is operated for the matrix, respectively.

NOTE

When the discrete I/O is specified as a reference number, $n=16m+1$ ($m=0, 1, 2, \ldots$) is required where n is ×××× of 1×××× or 0××××.

(3) Table Size

The constant ×××× is specified as the table size. The range of the fixed value depends on the type of matrix. See Tabale 4.44.

NOTE

Specify the number of registers (or of sets of 16 discrete signals) but not the number of bits as the table size.

Except for MBIT, SENS, and MROT, the source and destination tables have the same size (N : N).

(4) Pointer

Only CMPR, MBIT, and SENS functions use a pointer.

4. 9. 3 Function and Operation of Matrix

Tabale 4.44 lists the functions and operations of matrices.

(1) AND, OR, XOR

(a) Function

Since each logical function is provided with similar function block, these functions are described in this section. Fig. 4.74 shows the truth tables and equivalent relay circuits of AND, OR, and XOR applied to 1 bit signal.

Fig. 4. 74 Truth Tables

AND (A · B = C)

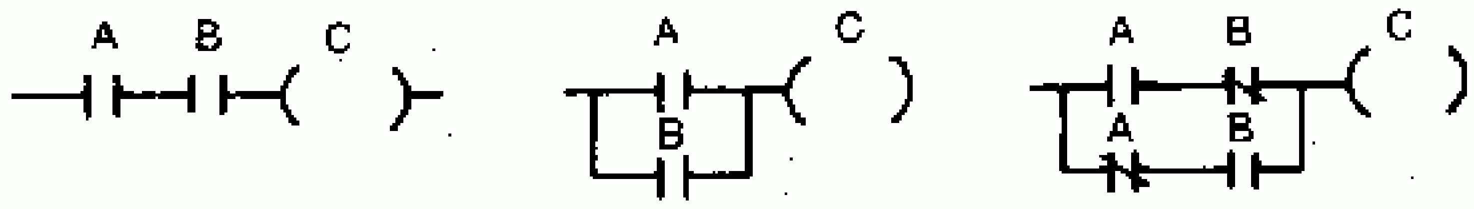
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR (A + B = C)

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

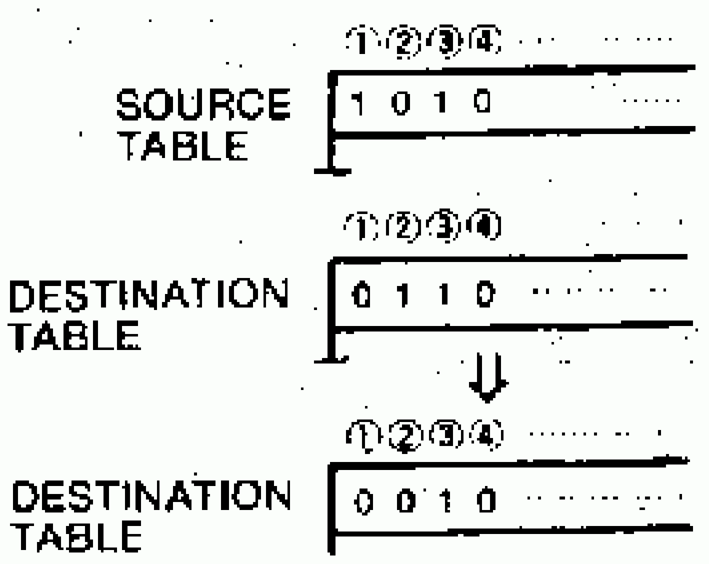
XOR (A ⊕ B = C)

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0



This function causes two matrices of equal length to be logically AND'ed, OR'ed or XOR'ed together and the results stored for reference by any other logic function. The result of the logical AND will be a one (ON) bit only if both bits (one from each matrix) are one bits; other-wise, the result will be zero (OFF) bit. One bit from each matrix with the same identification number will be combined in acoordance to the truth table.

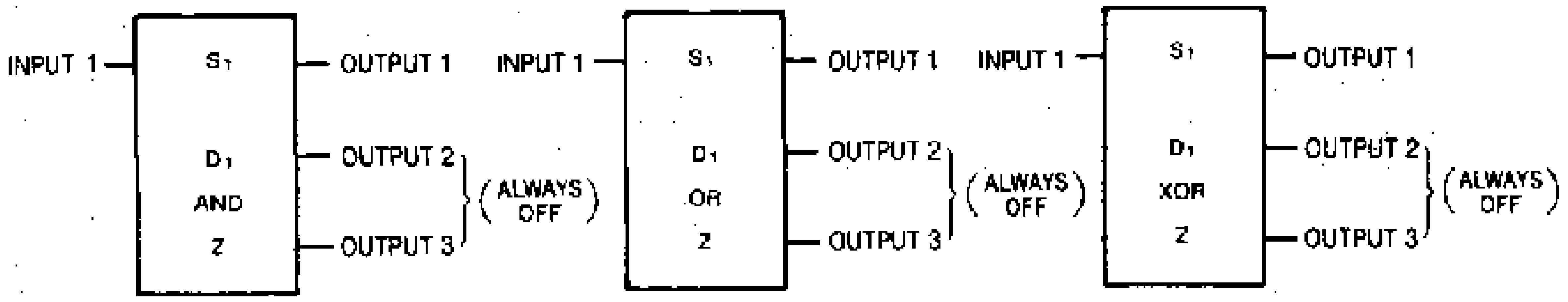
Example: AND



Note: The bit numbers are circled. AND is applied to the bit pair having the same bit number and the result is stored in the bit having the same number.

All bits in the matrices will be operated upon every scan the function is enabled. The Source matrix is not altered only copied. The current content of the Destination matrix is used for the AND, OR, or XOR operation and then replaced with the result of the operation. The Destination matrix is altered by the AND, OR, or XOR operation. The Matrix AND operation is useful to clear large groups of resisters to zero(when ANDed with a matrix of zeros) or to construct masks within the controller. The Matrix OR operation is useful to construct masks within the controller. The Matrix XOR operation is useful to detect differences between two matrices within one scan and, when operated upon the same matrix in both Source and Destination, to clear a matrix to zero. See Example 2.

(b) Form



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:
  Any of 0XXXX, 1XXXX, 3XXXX, and 4XXXX can be specified.
- Reference number (D1) of destination:
  Either 0XXXX or 4XXXX can be specified.
- Table size (Z): 1 to 100

(d) Inputs

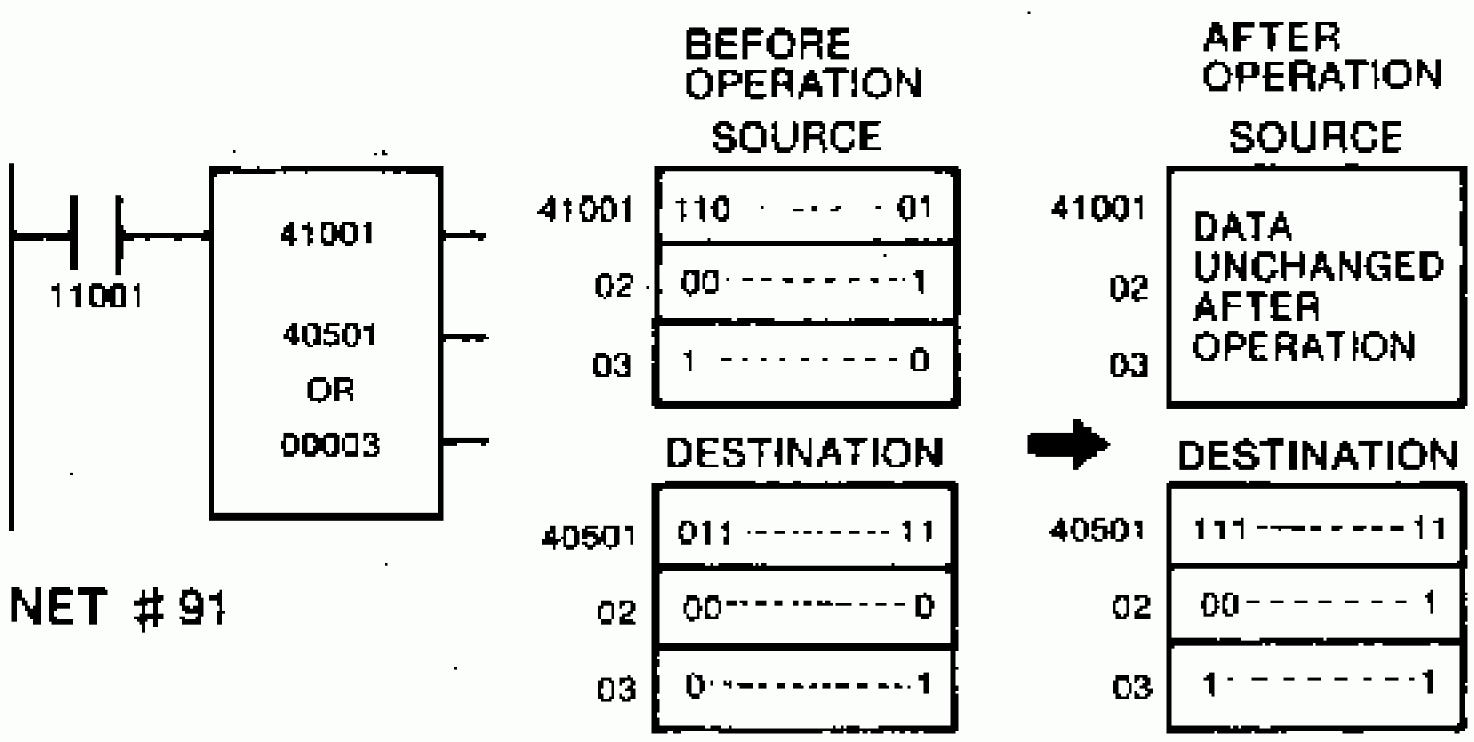
Only the input 1 is used with the these functions. When this input node receives power flow, the each operation is performed. Every scan, when enabled, the AND, OR or XOR will operate upon the entire content of the matrices. Transitional contact can be used if a single operation is desired.

(e) Outputs

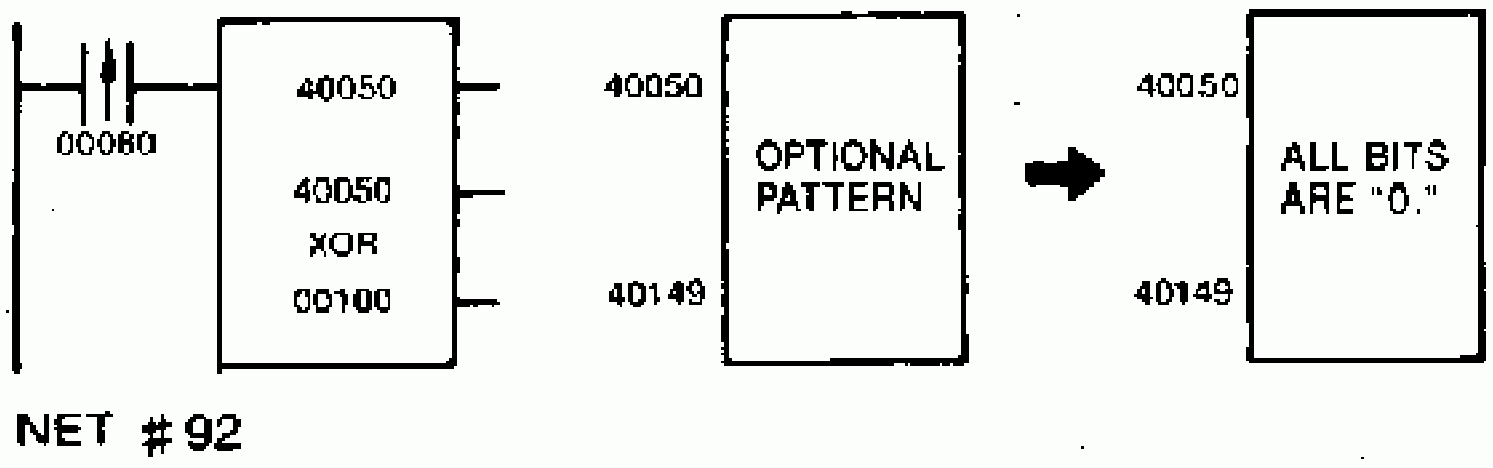
The each matrix function utilizes only the output 1. The lower two outputs have no significance and will be OFF under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus, the output 1 allows Function Blocks to be cascaded or chained horizontally within a network.

(f) Example

Example 1



Example 2

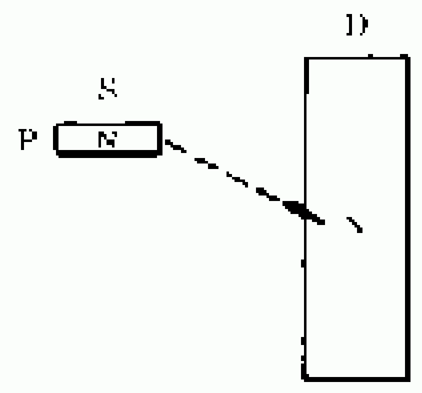
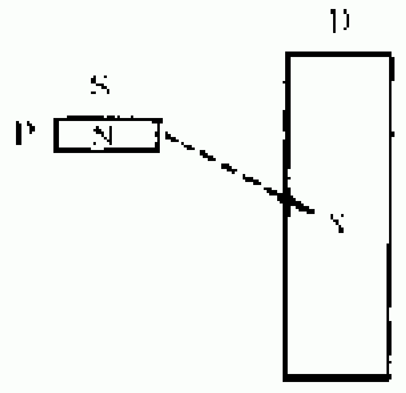
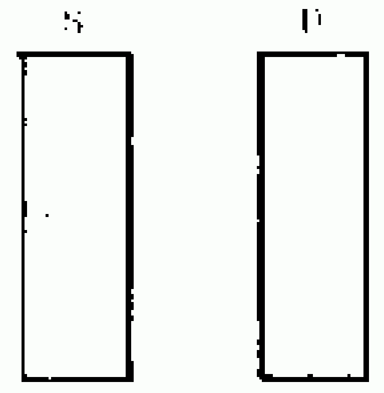
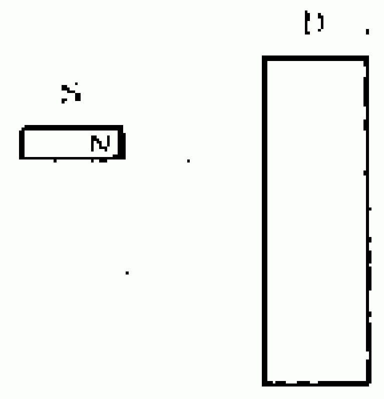
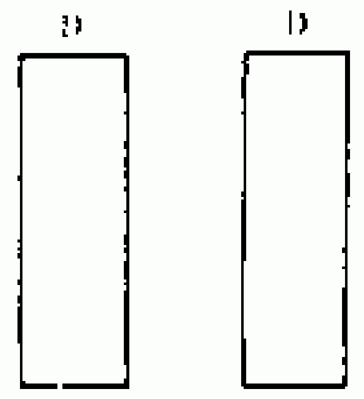


This XOR is used to clear the source and destination tables when operated upon the same matrix in both Source and Destination.

4. 9. 3 Function and Operation of Matrix (Cont'd)

Table 4. 44 List of Matrix Functions

Item \ Symbol	AND	OR	XOR	COMP	CMPR
Reference Page	130	130	130	134	134
Explanation of Data Form and Function S: Source D: Destination P: Pointer ( ): Contents	S D $(S)\cdot(D) \rightarrow D$ AND of all bits in one scan.	S D $(S)+(D) \rightarrow D$ OR of all bits in one scan.	S D $(S)\oplus(D) \rightarrow D$ XOR of all bits in one scan.	S D $\bar{(S)} \rightarrow D$ Inverts all bits in one scan.	S P D • Compares (S) and (D) bit by bit. • Finds one mismatching in one scan.
Source Reference	0×××× or 1×××× or 3×××× or 4××××	0×××× or 1×××× or 3×××× or 4××××	0×××× or 1×××× or 3×××× or 4××××	0×××× or 1×××× or 3×××× or 4××××	0×××× or 1×××× or 3×××× or 4××××
Destination Reference	0×××× or 4××××	0×××× or 4××××	0×××× or 4××××	0×××× or 4××××	4×××× (pointer)
Table Size	1–100	1–100	1–100	1–100	1–100
Input 1	Executes AND	Executes OR	Executes XOR	Executes COMP	Executes CMPR
Input 2	—	—		—	Sets pointer to 0 (regardless of ON/OFF status of input 1).
Input 3	—	—	—	—	—
Output 1	Same as ON/OFF status of input 1.				
Output 2	Always OFF				ON when mismatching is found.
Output 3	Always OFF				ON when mismatching is found with S's bit being 1 and D's bit being 0.
Where Operation is not executed when Input is ON.	—				

MBIT	SENS	BROT	MROT	TWST
137	138	140	142	144
				
• Sets/clears Nth bit of D. • Sets/clears a bit in one scan.	• Tests Nth bit of D. • Tests a bit in one scan.	• Shifts left or right all bits of S and places results in D. • Shifts entire table one bit in one scan.	• Shifts left or right all bits of D and places results in D. • Shifts entire table N bits in one scan.	• Interchanges higher-place byte and lower-place byte of each register of D and inverts bit arrangement. • Executes everything in one scan.
3XXXX or 4XXXX (pointer) or constant 1–9600	3XXXX or 4XXXX (pointer) or constant 1–9600	0XXXX or 1XXXX or 3XXXX or 4XXXX	4XXXX	—
0XXXX or 4XXXX	0XXXX or 1XXXX or 3XXXX or 4XXXX	0XXXX or 4XXXX	4XXXX	4XXXX
1–600	1–600	1–100	1–100	1–100
Executes MBIT	Executes SENS	Executes BROT	Executes MROT	Executes TWST
Sets to 1 when ON, clears to 0 when OFF. (Input 1 must be ON.)	1+N→N only when ON and S is 4XXXX. (Input 1 must be ON.)	Shifts left (LSB→MSB) when ON. Shifts right (MSB→LSB) when OFF.	Shifts left (LSB→MSB) when ON Shifts right (MSB→LSB) when OFF.	—
1+N→N only when ON and S is 4XXXX. (Input 1 must be ON.)	Sets N to 1 when ON (regardless of ON/OFF status of input 1).	When ON, Shift register When OFF, Shift register ← "0"	When ON, Shift register When OFF, Shift register ← "0"	—
			ON when input 1 is ON and MROT can execute.	Same as ON/OFF status of input 1.
ON when Nth bit is 1 after execution (to check input 2 ON).	ON when Nth bit is 1 after execution.	ON when carry is 1 after shift.	ON when input 1 is ON and MROT cannot execute.	Always OFF
ON when N exceeds maximum bit number decided by table size (regardless of ON/OFF status of input 1).		Always OFF		
N = 0 or N > maximum bit number decided by table size.		—	Number of shift (N): 16 bits or more	—

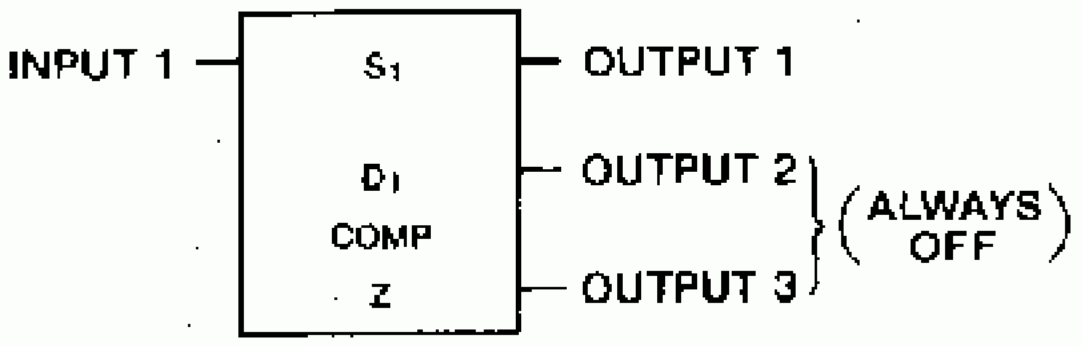
(2) COMP

(a) Function

This function causes the content of one matrix to be complemented (all ones replaced by zeros, and zeros by ones) and placed in another matrix for reference by any other function. The entire matrix is operated upon every scan the function is enabled by power flow to the input. The result of the Complement operation is placed in the Destination matrix; the previous content of this matrix is lost. The Source matrix is not altered only copied.

The Matrix Complement is useful to alter normally closed inputs to the same base as normally open inputs or to move masks within the controller.

(b) Form



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:

  Any of 0XXXX, 1XXXX, 3XXXX, and 4XXXX can be specified.

- Reference number (D1) of destination:

  Either 0XXXX or 4XXXX can be specified.

- Table size (Z): 1 to 100

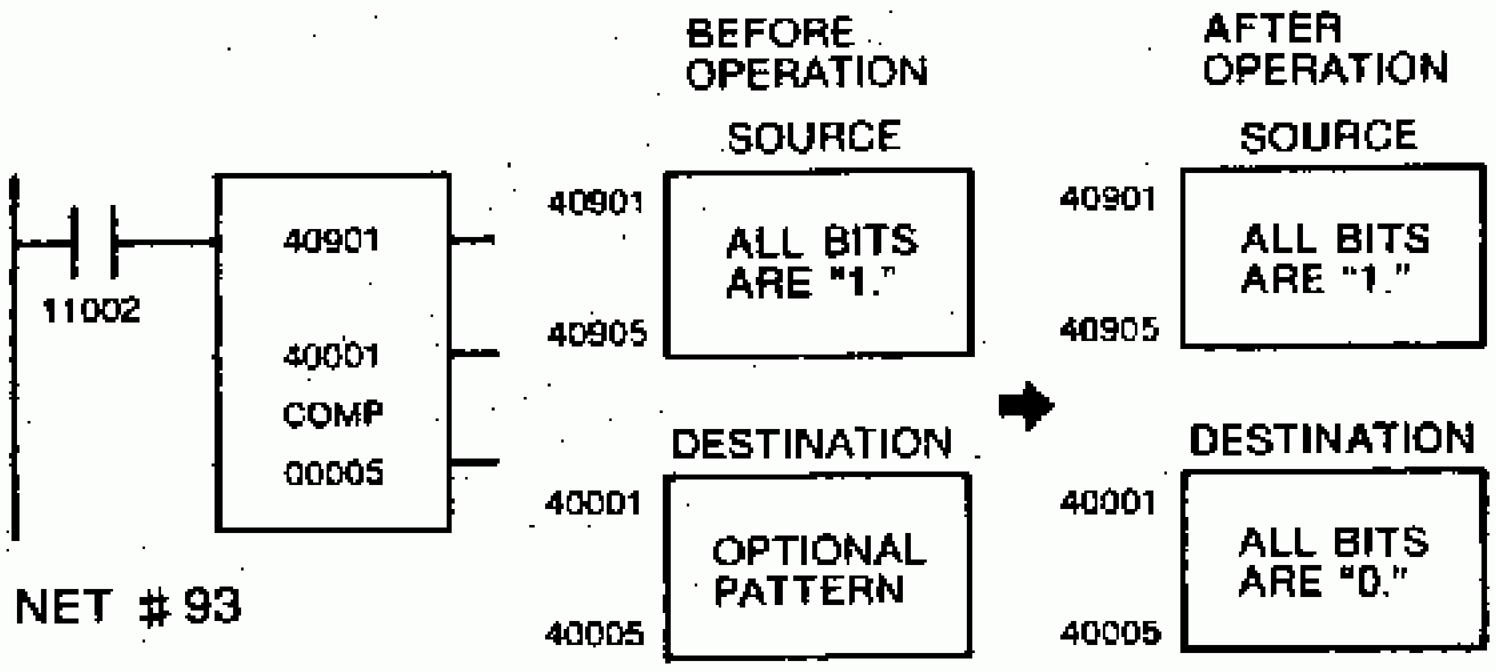
(d) Inputs

Only the input 1 is used with the COMPLEMENT function. When this input node receives power flow, the COMPLEMENT operation is performed. Every scan, when enabled, the COMPLEMENT will operate upon the entire content of both matrices. Transitional contacts can be used if a single operation is desired.

(e) Outputs

The COMPLEMENT matrix function utilizes only the output 1. The lower two outputs have no significance and will be OFF under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally with a network.

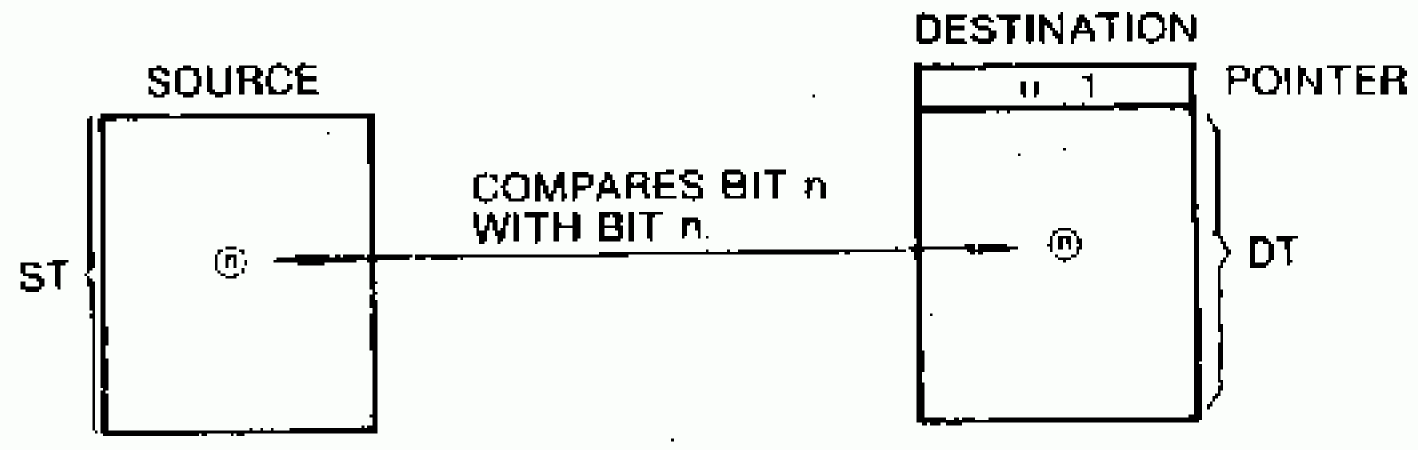
(f) Example



(3) Compare (CMPR)

(a) Function

This function causes two matrices to be compared on a bit-by-bit basis; their contents are not altered only examined. When enabled, the compare function will examine one bit from each matrix with the same identification number. If these bits agree (both zero or both ones) the next bit from each matrix is compared; however,



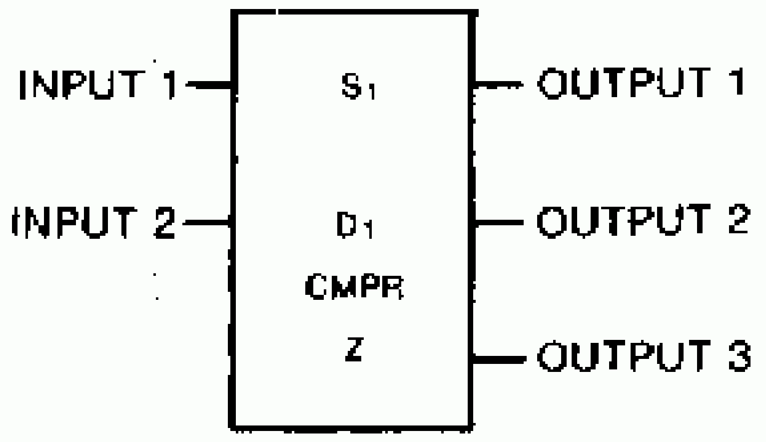
if they do not agree, the compare function will halt. Each scan the compare is performed, either the end of the matrix is located or a miscompare is encountered. If both matrices are identical, the entire length, up to 100 registers or 1600 bits (such as discrete inputs/outputs), is compared each and every scan.

An output is used to indicate the result of compare. It will be ON if a miscompare is detected and OFF if the end of the matrices is reached. A pointer is the only register whose content is altered by the Compare. This pointer is referred to by the Destination block and indicates which specific bit was responsible for the miscompare, if the operation is terminated by a miscompare (output ON). The pointer can also be used to cause the comparison to begin on bits other than at the beginning; the compare function always proceeds towards the end (high bit number) of the matrices. Before the Compare, the pointer will be incremented; if the pointer is at the end of the matrix or longer, it will be reset to one prior to beginning the compare operation.

NOTE

1. The contents of the pointer can be changed by another logic circuit. To start compare at bit i, set i−1 to the pointer before starting.
2. If there are no miscompares, the pointer will be at the end of the matrices when the comparison is completed.

(b) Format



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:

  Any of 0XXXX, 1XXXX, 3XXXX, and 4XXXX can be specified.

- Reference number (D1) of destination:

  Holding register 4XXXX is specified (it defines the pointer).

- Table size (Z): 1 to 100

(d) Inputs

Only the upper two inputs are used with the COMPARE function. Every scan the input 1 receives power flow, the Compare operation is performed. Up to 1600 pairs of bits can be compared each scan the input is enabled. Transitional contacts can be used if a single operation is desired.

The input 2 controls the reset of the pointer. When this input receives power flow, the pointer will be reset to zero prior to the comparison; otherwise, the comparison begins at the current value of the pointer plus one. If the pointer is at the end of the matrix or greater, it will be reset to zero prior to the comparison regardless of the state of this input. Matrices begin at bit number one (not zero), thus resetting implies the value one.

(e) Ouputs

The COMPARE matrix function utilizes all three available outputs. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Block to be cascaded or chained horizontally within a network. The output 2 will supply power flow if the comparison has detected a miscompare; this output will be OFF if either no compare is being performed (top input does not receive power flow), or if no miscompares have been detected and the end of the matrices have been reached. The output 3 senses the Source matrix bit if and only if a miscompare has been detected; this output will supply power flow when this bit is a one and be OFF if the bit is a zero. Using the lower two outputs, logic can be built to determine the exact bit configuration causing a miscompare (Source = one and Destination = zero, or vice versa).

(f) Example

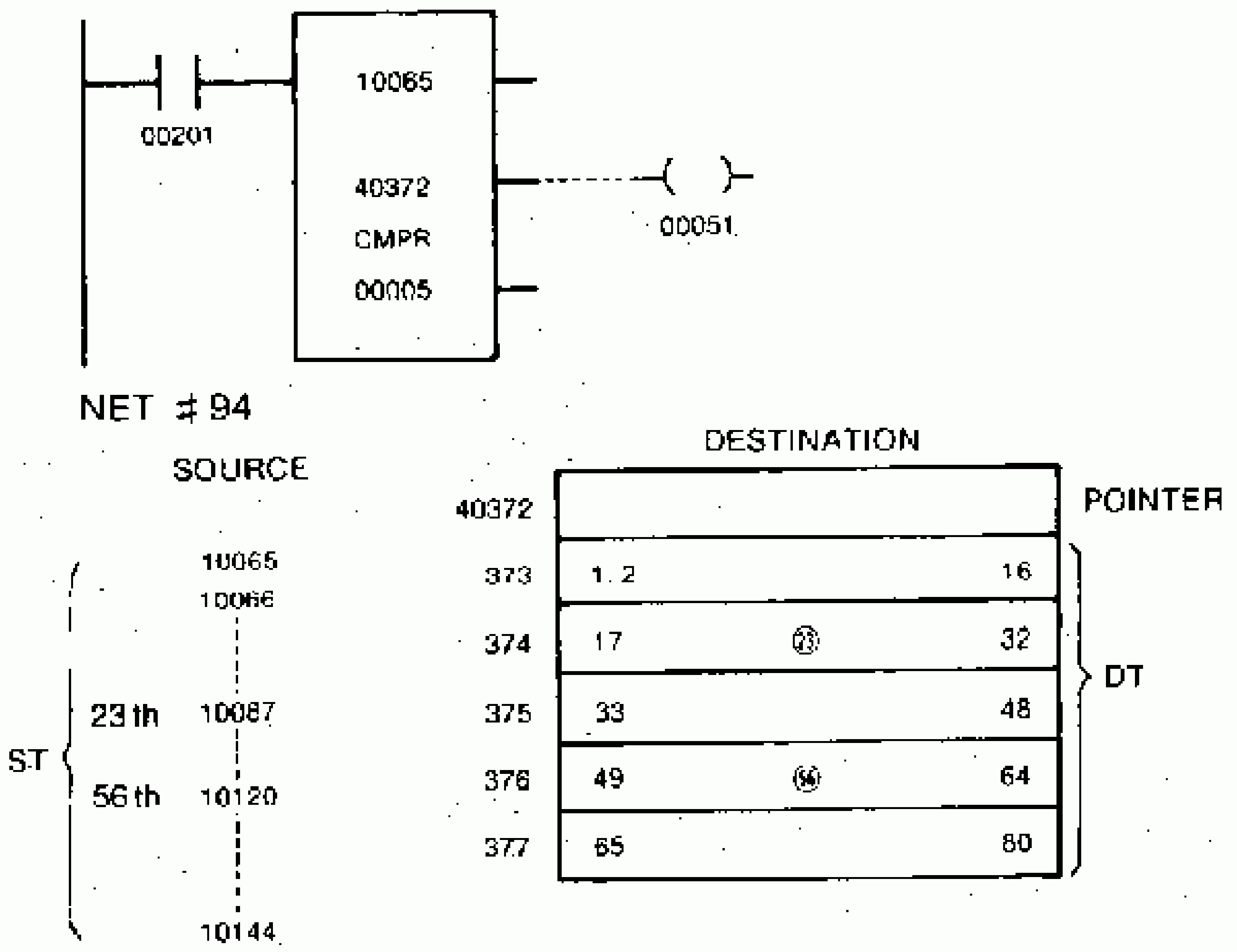


Figure above illustrates a typical COMPARE Matrix function. If the pointer (register 40372) contains the value zero or one with input 10056 is energized, the comparison begins at input 10065 and bit 1. The entire matrix all 80 bits will be compared to the inputs unless a miscompare is detected. However, if bit 23 in register 40372 is energized. If input 10087 is ON and bit 23 in the destination matrix (40373-40377) is a zero, coil 00051 will be energized.

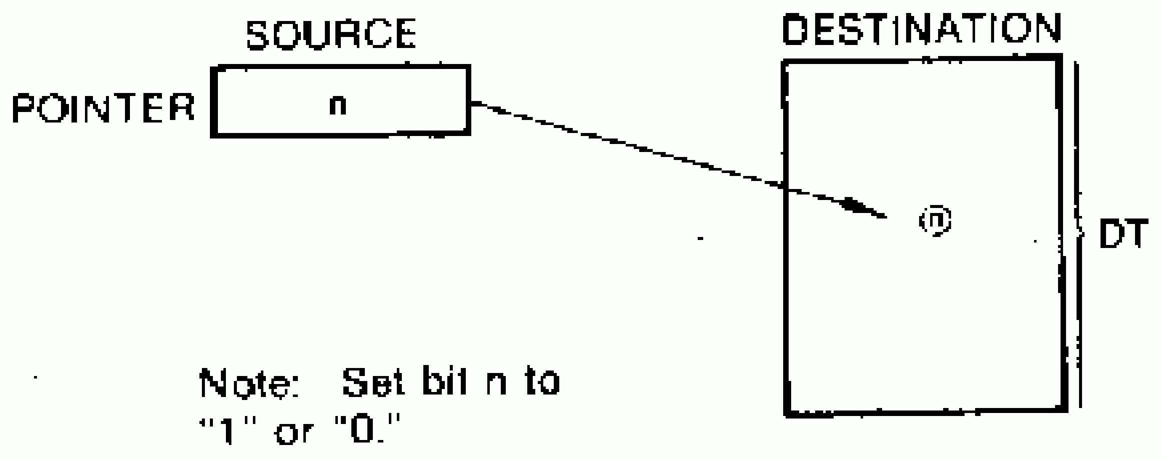
On the next scan with input 00201 still energized and the content of register 40372 not altered, the comparison will start at bit 24 and proceed towards the end of the matrix. If input 10120 is OFF and bit 56 set to a one value in the matrix 40373-40377, an additional miscompare is detected. The value in register 40372 is now 0056 and coil 00051 remains ON, 00051 since the Source bit is a zero (input 10120 OFF). The pointer value of the first miscompare is lost and replaced by the location of the second miscompare. To retain miscompare locations, they should be saved in another location, such as a table, with a register to table move function. This move function can be controlled by the coil 00051 in this example.

On the next scan, unless there are additional miscompares, the comparison begins at bit 57 and reaches the end of the matrix. The pointer will contain the value 81, coils 00051 will be OFF. On the next (fourth) scan, the comparison begins again and will detect bit 23 as a miscompare again unless the input or bit status is altered to prevent the miscompare. The compare function is very powerful and very fast. Unless action is taken, pointer values will be replaced by other values and repetitive miscompares detected. At best, all bits are compared every scan when they all agree, and at worst case one bit is compared each scan when they all disagree.

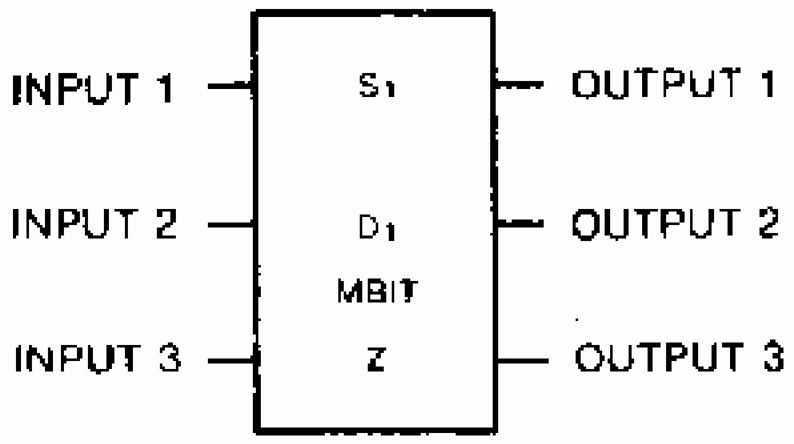
(4) Modify (MBIT)

(a) Function

This function allows individual bits in a matrix to be altered. Only one bit per scan can be affected by this function; all other bits retain their state. Bits can be either set to a one (ON) condition or cleared to a zero (OFF) condition. A pointer is used to indicate which bit is to be modified.



(b) Form



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source: 3××××, 4×××× or the fixed value of 1-9600 can be specified (it defines the pointer).

NOTE

1. If the pointer is 4××××, its content can be altered by another logic circuit.
2. MBIT performs nothing if the content of the pointer is 0 or exceeds the permissible maximum bit number specified by table size.
3. Where the value of the pointer may be a constant, it indicates a bit number.

- Reference number (D1) of destination: Either 0×××× or 4×××× can be specified.
- Table size (Z): 1 to 600 (corresponding to 16-9600 bits)

(d) Inputs

All three inputs are used with the Bit Modify function. Every scan the input 1 receives power flow, a bit is altered. Any of up to 9600 bits can be modified by this function. Transitional contacts can be used if a single operation is desired. The input 2 controls how that bit is to be modified. When this input receives power flow, the bit will be set to a one (ON) condition; no power flow results in the bit being cleared to a zero (OFF) condition. The current status (ON/OFF) of the bit has no effect on the result after this function. The input 3 when receiving power flow will cause the pointer, if stored in a holding register only, to be incremented after the bit is altered. The pointer is not incremented if it is a constant, stored in an input register, or there is no power flow to the input 3. The pointer, if incremented beyond the size of the matrix, will be reset to one automatically.

(e) Outputs

The Bit Modify matrix function utilizes all three outputs. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function blocks to be cascaded or chained horizontally within a network.

The output 2 supplies power flow whenever the bit is ON, one, after the operation is performed. Thus this output can be described as sensing the resultant bit or merely copying the state of the input 2 with a successful function. The output 3 will supply power flow if the pointer's magnitude is beyond the size of the matrix (pointer too large). Thus if the pointer in an input register or a holding register without automatic incrementing exceeds the matrix size, no operation is performed and this output is used to indicate the error condition.

(f) Example

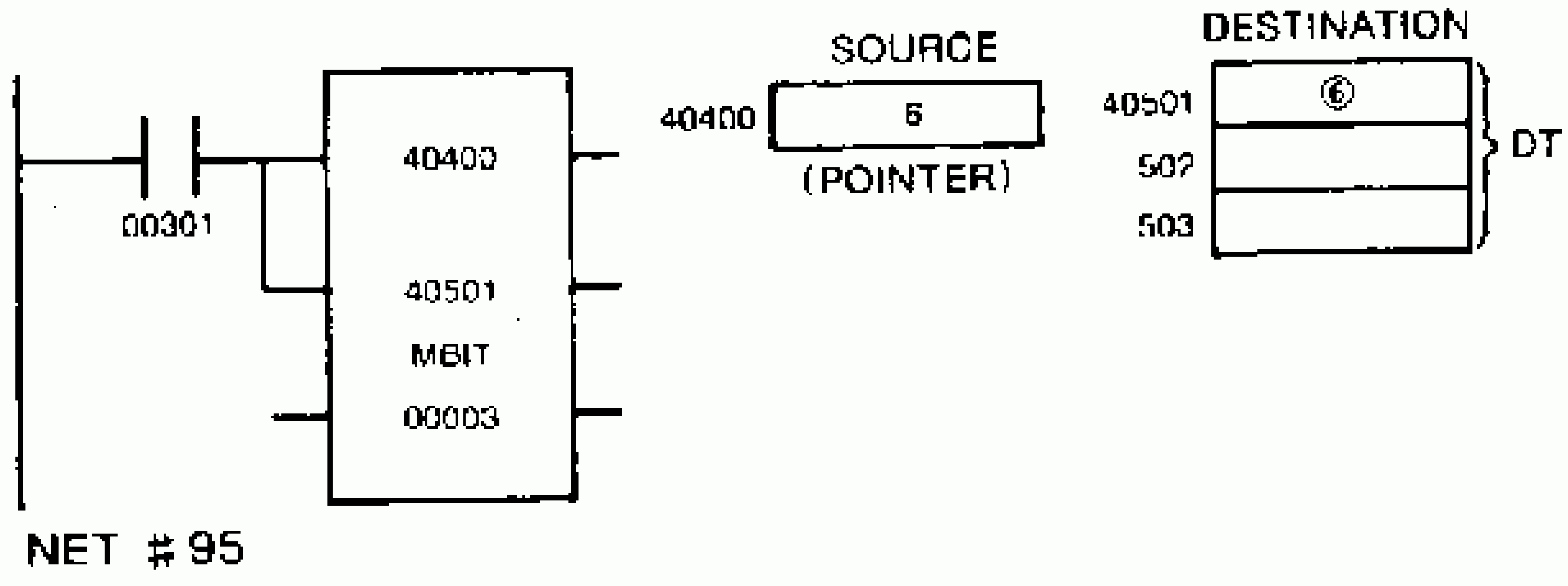
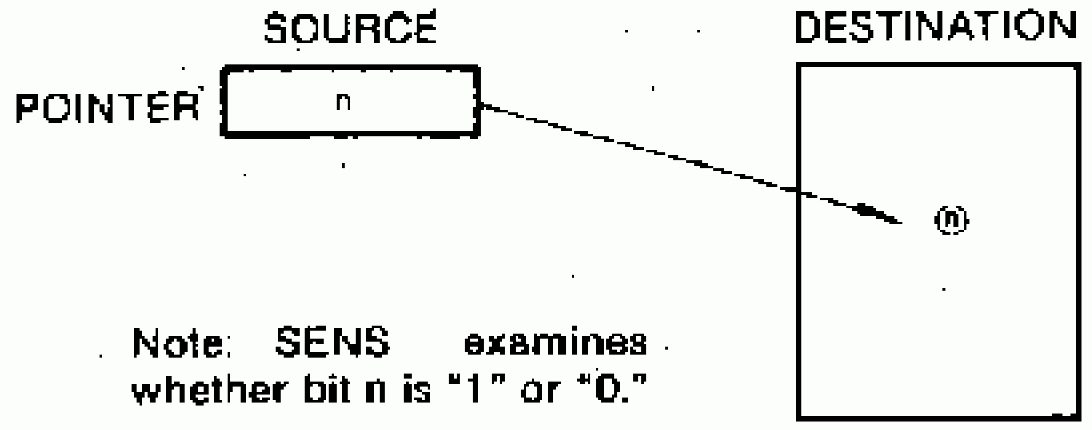


Figure above illustrates a typical Bit Modify Matrix function. If the pointer (register 40400) contains the value six, when coil 00301 is energized, the bit in the matrix 4501-40503 at location six will be set to a one. The control that sets the bit (in lieu of clearing the zero) is the vertical connection to the middle input. As long as coil 00301 is energized, bit six will be set every scan; unless ohter logic clears this bit, no change in the bit will be detectable.

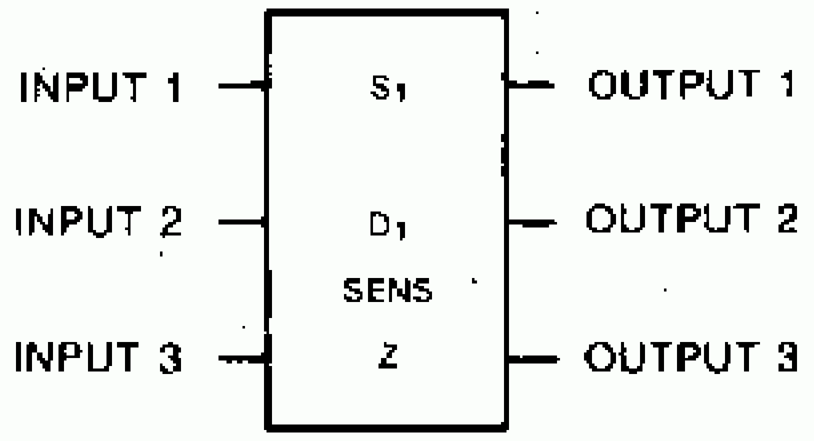
(5) Sense (SENS)

(a) Function

This function allows individual bits in a matrix to be examined, but not altered. An output is used to indicate one (ON) bits with power flow and a zero (OFF) bits without power flow. The operation of this function is similar to the Bit Modify previously discussed, except that no bits are modified. The status of only one bit can be obtained per scan.



(b) Form



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:
  3XXXX, 4XXXX or a constant of 1-9600 can be used (it defines the pointer).

NOTE

1. If the pointer is 4XXXX, its content may be specified by another logic circuit.
2. SENS performs nothing if the content of the pointer is 0 or exceeds the permissible maximum bit number determined by table size.
3. Where the value of the pointer may be a constant and it indicates a bit number.

· Reference number (DI) of destination:
Any of 0××××, 1××××, 3××××, and 4×××× can be specified.

· Table size (Z): 1 to 600 (corresponding to 16-9600 bits)

(d) Inputs

All three inputs are used with the Bit Sense function. Every scan the input 1 receives power flow, a bit is located in a matrix and its status is obtained. Any of up to 9600 bits can be obtained with this function. Transitional contacts can be used if a single operation is desired.

The input 2 controls the incrementing of the pointer. When this input receives power flow, the pointer, if stored in a holding register only, will be incremented after the bit is examined. Both the inputs 1 and 2 must receive power flow for the pointer to be incremented. The pointer is not incremented if it is a constant, stored in an input register, or there is no power flow to the input 2. The pointer, if incremented beyond the size of the matrix, will be reset to one automatically. The pointer is also reset to one if the input 1 receives power flow; this resetting is accomplished prior to sensing the bit as required by the top input.

(e) Outputs

The bit Sense matrix function utilizes all three outputs. The output 1 will supply power flow whenever the input 1 receives power flow. Thus the output 1 allows Function Blocks to be cascaded or chained horizontally within a network. The output 2 supplies power flow whenever the bit being sensed (addressed by the pointer) is an one (ON) bit; this output will not supply power flow whenever the bit is a zero (OFF) bit. The input 2 is the resultant of the sense function. The output 3 will supply power flow if the pointer's magnitude is beyond the size of the matrix (pointer too large). Thus if the pointer is a holding register with automatic pointer incrementing and it exceeds the matrix size, no operation is performed and this output is used to indicate the error condition.

(f) Example

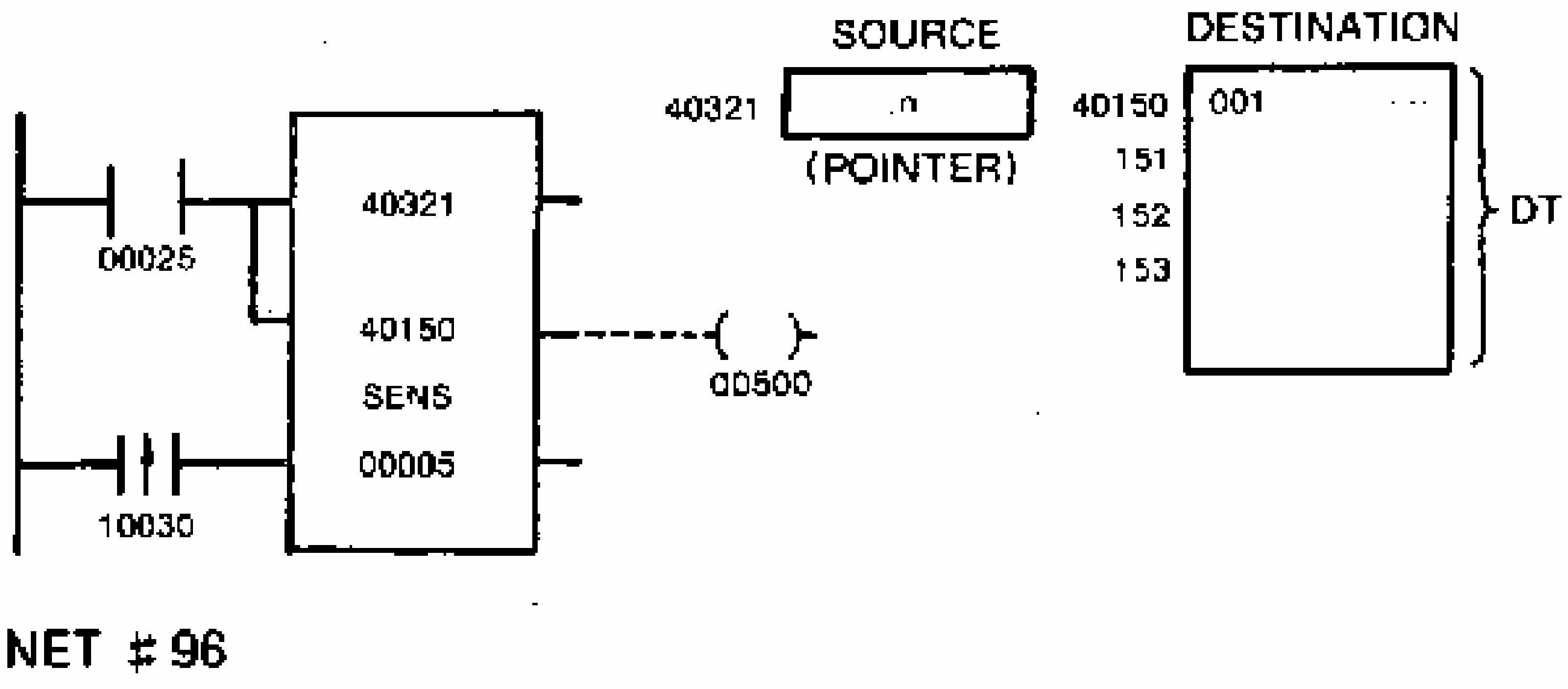


Figure above illustrates a typical Bit Sense Matrix function. If the pointer (register 40321) contains the value zero when input 00025 is energized, coil 00500 will be off during that first scan. Since there is a vertical connection for the power flow to input 2 the pointer will increment by one each scan input 00025 is energized.

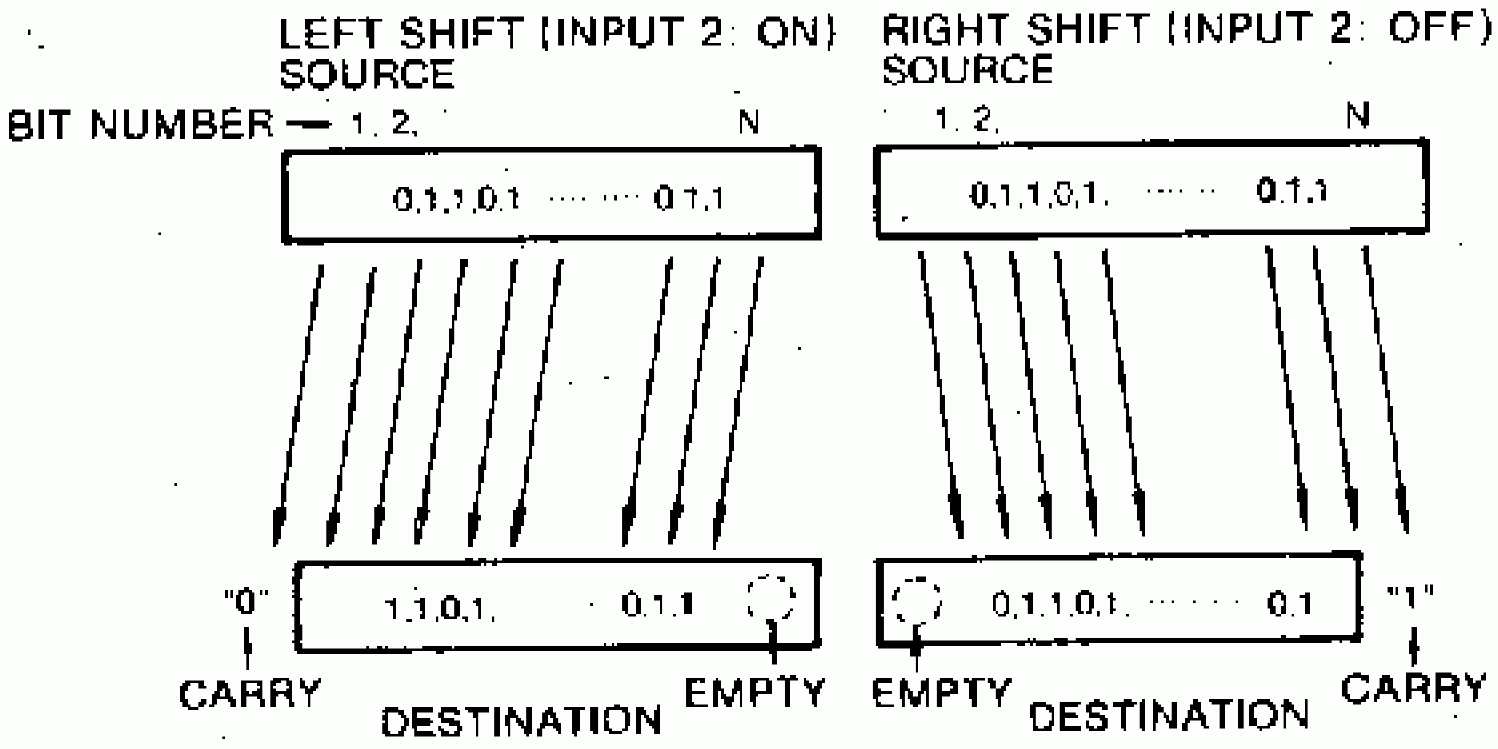
Thus on the second scan that this input is ON, the pointer will be at the value one, and the first bit in the matrix (registers 40321-40324) will be sensed. This bit is a zero, and coil 00500 remains OFF; this coil will also be OFF for the third scan while bit two is sensed. When the pointer is incremented to the value three, coil 00500 will be ON sensing bit three as a one bit.

As long as input 00025 is energized, the sense function "walks" through the matrix at the rate of one bit per scan; coil 00500 indicates the status of each bit. The sensing will return to bit one after bit 80 automatically if input 00025 is energized for a sufficiently long period. Whenever input 10030 is energized, the sensing will return to bit one regardless of the pointer's value; since a transitional contact is used, input 10030 must be deenergized and then re-energized to affect the pointer.

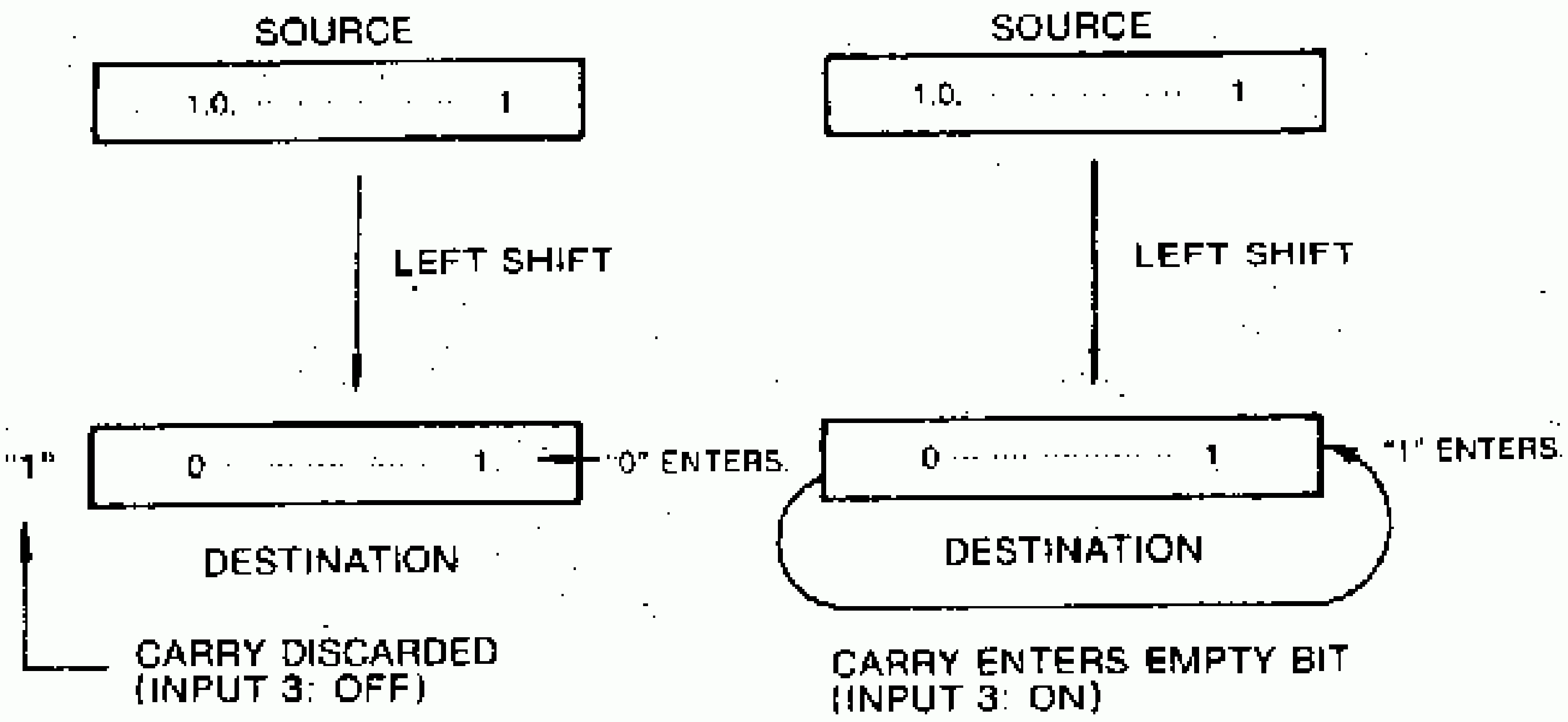
(6) Rotate (BROT)

(a) Function

BROT shifts all bits of the source table, one bit to the left or right, and stores the result in the destination table all in a scanning cycle. The direction of shift is determined by ON/OFF status of the input 2.

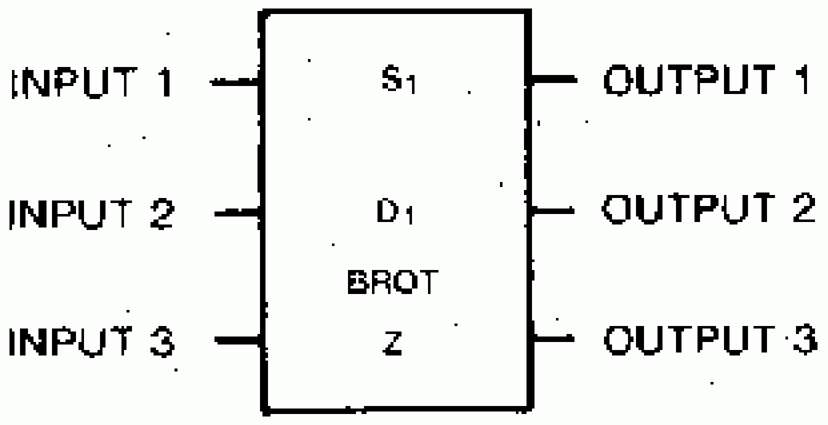


It is possible to select whether to fill the empty bit with 0 or with the carry, by ON/OFF status of input 3.



The source data remains unchanged unless the source and destination tables are the same. When the source and destination tables are the same, BROT realizes a 1-bit, N-stage (N is the total number of bits) shift register.

(b) Format



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:
  Any of 0××××, 1××××, 3××××, and 4×××× can be specified.
- Reference number (D1) of destination:
  Either of 0×××× or 4×××× can be specified.
- Table size (Z):
  1 to 100 (corresponding to 16-1600 bits)

(d) Inputs

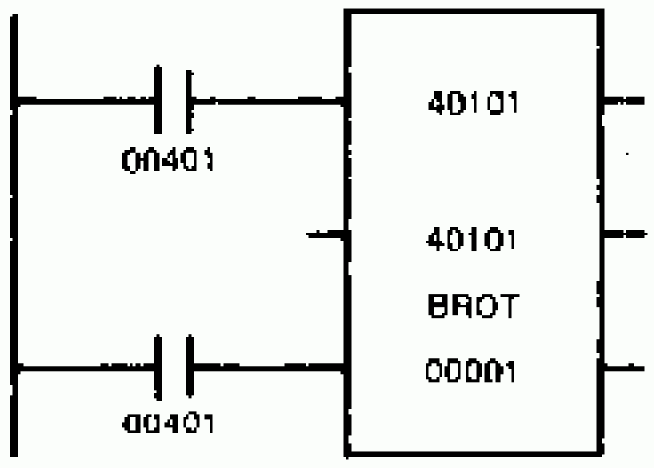
All three inputs are used with the Bit Rotate function. Every scan the input 1 receives power flow, all bits in the matrix will be rotated one position. Up to 1600 bits are moved with this function in one scan.

The input 2 controls the direction of the rotation. If this input receives power flow, the matrix will be rotated towards the left (bit 17 into 16, bit 16 into 15, ..., bit 3 into 2, bit 2 into 1, and bit 1 rotated out of the matrix). If the input 2 does not receive power flow, the matrix will be rotated towards the right (bit 1 into 2, bit 2 into 3, ... bit 15 into 16, bit 16 into 17, etc.); the last bit will be rotated out of the matrix. The input 3 controls what happens to the single bit location vacated by the rotate to create either a shift operation or a true rotate. When this input does not receive power flow, the bit rotated out is ignored and vacant location at the opposite end of the matrix will be filled with zero; this is a shift operation. If this input receives power flow, the bit rotated out is carried around unchanged and entered into the opposite end of the matrix; this is a true rotate operation.

(e) Outputs

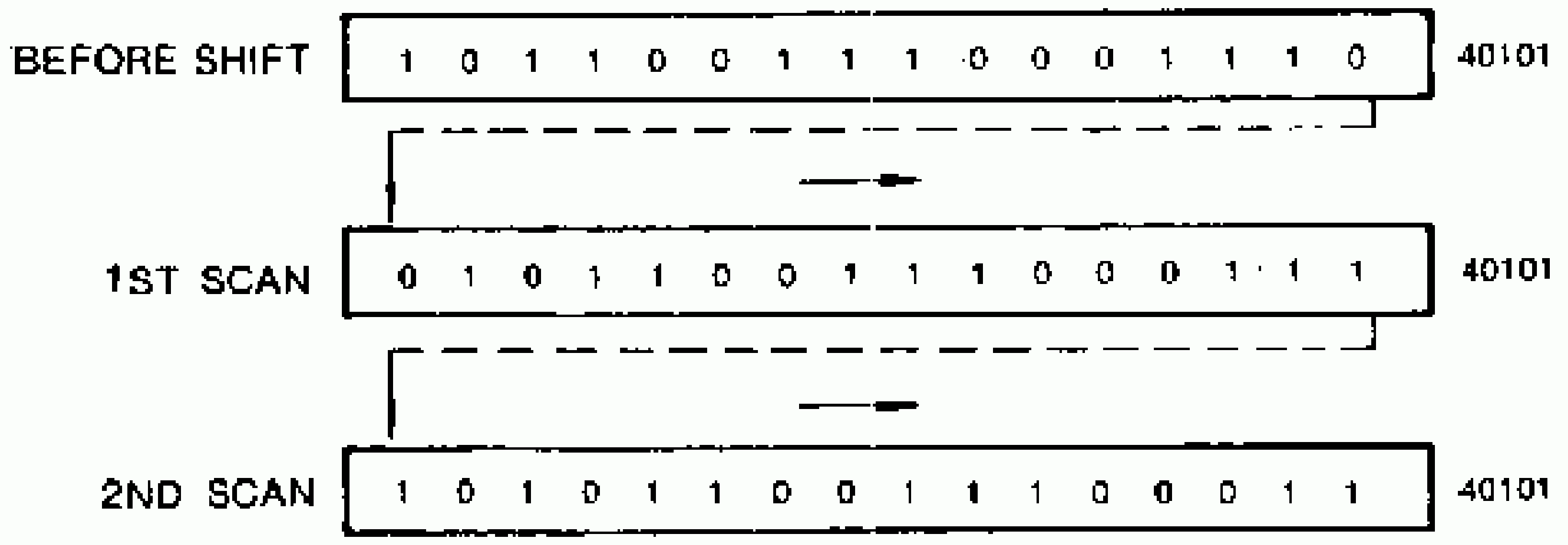
This function utilizes only the first two outputs. The output 3 has no significance and will be OFF (no power flow) under all conditions. The output 1 will supply power flow whenever the input 1 receives power flow. The output 2 supplies power flow when the carry is "1."

(f) Example



NET #97

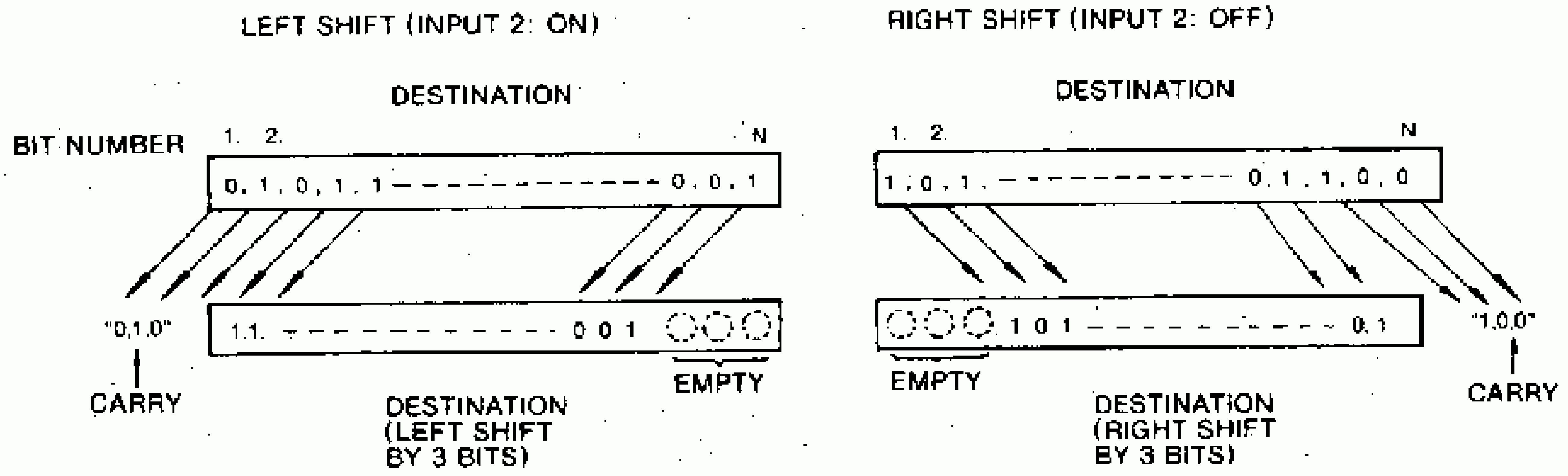
Note: This is an example of a 1-bit, 16-stage ring counter. (The source and destination tables are the same.) This shifts to the right as the input 2 is OFF.



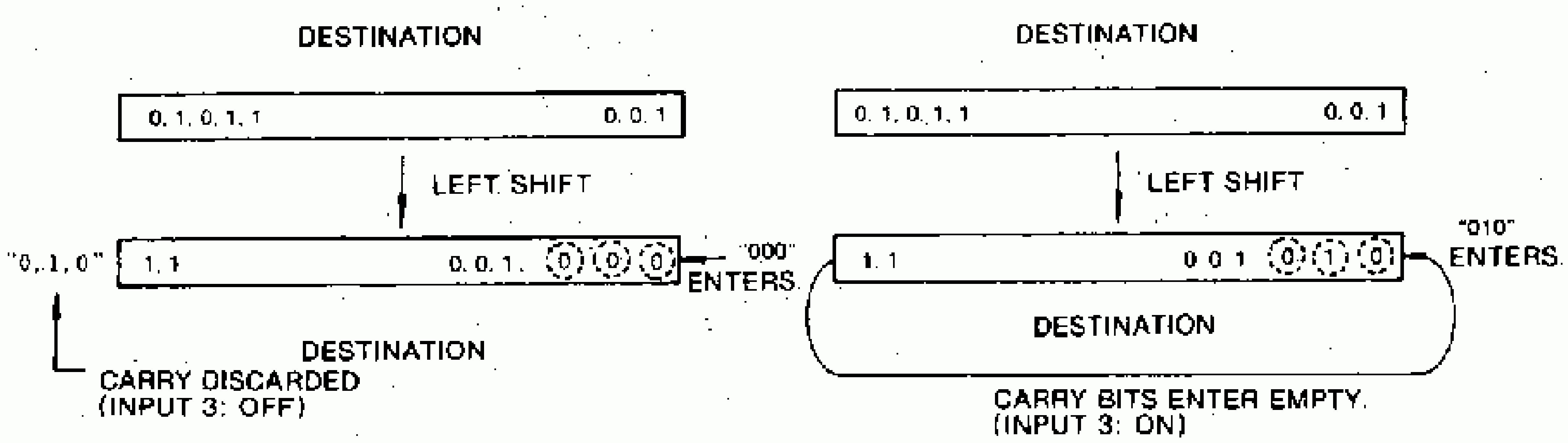
(7) Multi-Rotate (MROT)

(a) Function

MROT shifts all bits of the destination table, by the number of bits (1-15) stored in the source register to the left or right, and stores the result in the destination table all in a scanning cycle. Input 2 determines the direction of shift.

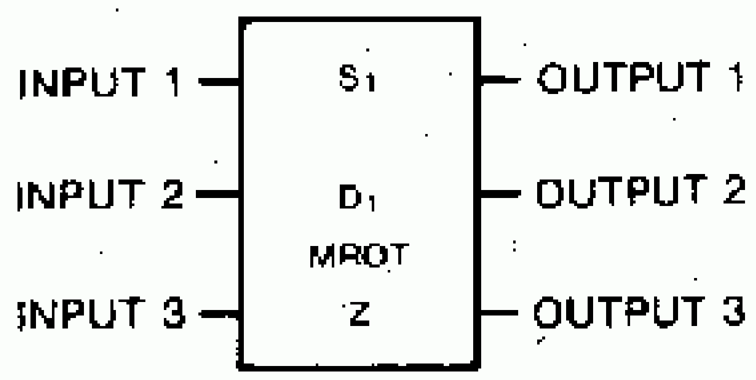


It is possible to select whether to fill the empty bits with 0's or with the carry bits, by ON/OFF status of input 3.



Unlike BROT, the destination data are changed by the shift successively. MROT realizes an n-bit (n is the number of bits to shift), N-stage (N is the total number of bits) shift register.

(b) Format



(c) Reference of elements (See Table 4.44.)

- Reference number (S1) of source:
  Only the holding register 4XXXX can be specified.
- Reference number (D1) of destination:
  Only the holding register 4XXXX can be specified.
- Table size (Z):
  1 to 100 (corresponding to 16-1600 bits)

(d) Inputs

When the input 1 receives power flow, MROT executes a shift. The input 2 determines the direction of shift as follows.

- When the inputs 1 and 2 are ON: Shift to left (to bit 1)
- When the input 1 is ON and 2 OFF: Shift to right (to bit N)

  The input 3 determines what should fill the empty bits as follows.

- When the inputs 1 and 3 are ON, the carry bits fill the empty bits.
- When input 1 is on and 3 off, 0's fill the empty bit.

(e) Outputs

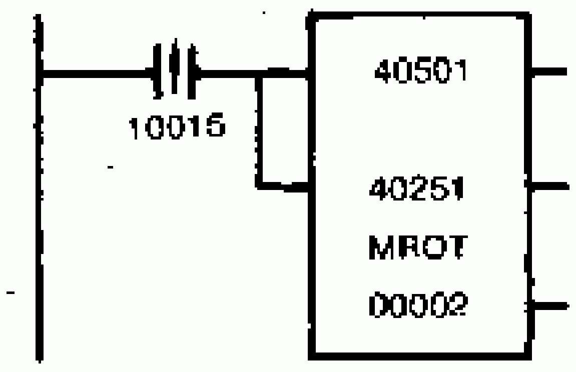
The output 1 is ON when the input 1 receives power flow and MROT executes a shift. The output 2 is ON when the input 1 receives power flow and MROT cannot execute a shift (the number of shift bits is 16 or more, or the source register is included in the destination table). The output 3 is always OFF.

NOTE

When number of shift specified in source register is 0, the shift is not operated, but the output 1 is ON.

(f) Example

Example 1



Note: This is an example of 5-bit left shift. 0's fill the empty bits as the input 3 is OFF.

NET #98

SOURCE

5	40501

SHIFT NUMBER DESTINATION

BEFORE SHIFT	0 1 1 1	0 0 1 0	1 1 0 0	0 1 0 1	40251
	0 0 0 1	0 1 1 0	1 1 1 0	1 0 1 1	40252

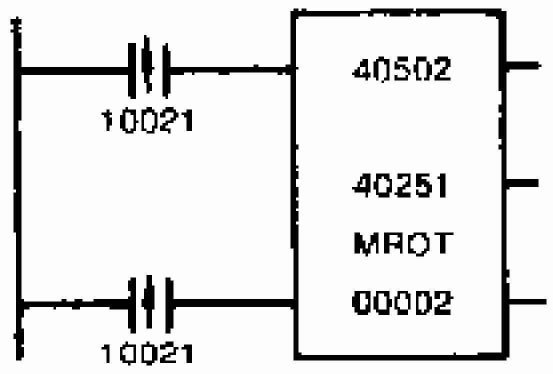
SOURCE

5	40501

SHIFT NUMBER DESTINATION

AFTER SHIFT	0 1 0 1	1 0 0 0	1 0 1 0	0 0 1 0	40251
	1 1 0 1	1 1 0 1	0 1 1 0	0 0 0 0	40252

Example 2



Note: This is an example of 8-bit right shift. The carry bits fill the empty bits as the input 3 is ON

NET #99

SOURCE

8	40502

SHIFT NUMBER DESTINATION

BEFORE SHIFT	0 1 1 1	0 0 1 0	1 1 0 0	0 1 0 1	40251
	0 0 0 1	0 1 1 0	1 1 1 0	1 0 1 1	40252

SOURCE

8	40502

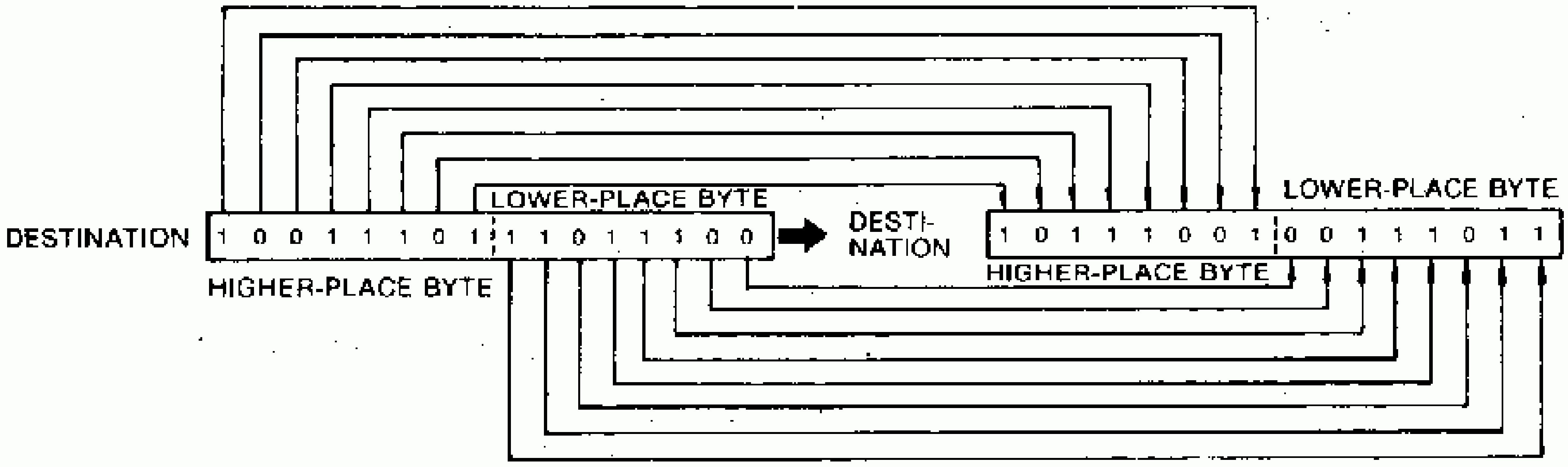
SHIFT NUMBER DESTINATION

AFTER SHIFT	1 1 1 0	1 0 1 1	0 1 1 1	0 0 1 0	40251
	1 1 0 0	0 1 0 1	0 0 0 1	0 1 1 0	40252

(8) Byte Rearrangement (TWST)

(a) Function

This function divides the registers in the destination table into the higher-place byte (8 bits) and the lower-place byte (8 bits). And bits in each byte are rearranged. All bits in the destination table are rearranged in one scan and stored in the destination table.



Before Byte Rearrangement

After Byte Rearrangement

When the statuses of coils and input relays are read with the U84 used as the MEMOBUS master, the information will come from a slave normally in such an order that a higher-place bit contains the status of a coil or input relay having a greater number. TWST can be used to rearrage the bits to the ordinary order.

SOURCE

8– 1	16– 9
24–17	32–25
40–33	48–41

Before Byte Rearrangement

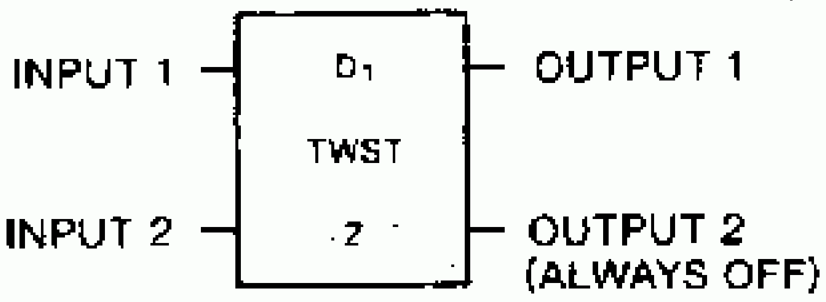
➡

SOURCE

1– 8	9–16
17–24	25–32
33–48	41–48

After Byte Rearrangement

(b) Form



(c) Reference of elements (See Table 4.44.)

- Reference number (D1) of destination:
  Only the holding register 4XXXX can be specified.
- Table size (Z): 1 to 100

(d) Inputs

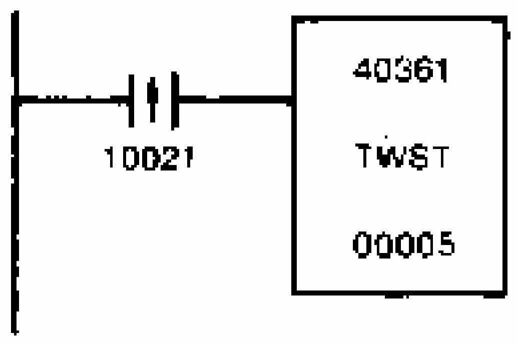
Only the input 1 is used for this function. When the input 1 receives power flow, the byte rearrangement operation can be performed.

(e) Outputs

TWST utilizes only the output 1 which will supply power flow whenever the input 1 receives power flow.

(f) Example

Example



DESTINATION

40361	0110	1010	1100	0001
40362	1110	0101	1101	1110
40363	0000	0101	1100	1100
40364	1110	1110	1011	0101
40365	0101	0011	1011	1111

Before Byte Rearrangement

➡

DESTINATION

40361	0101	0110	1000	0011
40362	1010	0111	0111	1011
40363	1010	0000	0011	0011
40364	0111	0111	1010	1101
40365	1100	1010	1111	1101

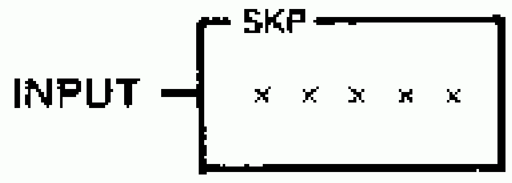
After Byte Rearrangement

4.10 SKIP

This function allows logic in groups of networks to be skipped and thus not sloved. Networks that are skipped will have their coils (if any) unchanged and register content unaltered; skipped networks are thus basically "frozen."

The Skip Function is useful to reduce the logic scan time. A skip of zero networks saves all of the time required to solve the logic remaining. A skip of non-zero networks reduces the time to solve the skipped logic to that of relay functions. The reduction in scan time can not exceed that of the logic remaining. Since all logic must be examined to count and determine quantity of networks (whose size does vary greatly), non-zero skips will save less time than a zero skip. In either case, the controller's over all scan time can not be reduced beyond that required to service I/O devices. In addition, the Skip Function can be used to select various groups of logic to be performed from a larger selection of logic.

(1) Form



Only one function block is used with the skip function, and input is available for control.

- Reference number

  The function block specifies the quantity of networks to be skipped. The value can be fixed quantity up to 9999 or a register reference (3 or 4XXXX

(2) Function and Operation

Assume the Skip instruction is stored in the network N and the number of networks to be skipped is J.

In a scanning cycle when the input is ON, processing of the J successive networks including network N (N, N+1, N+2, ..., N+J-1) is stopped.

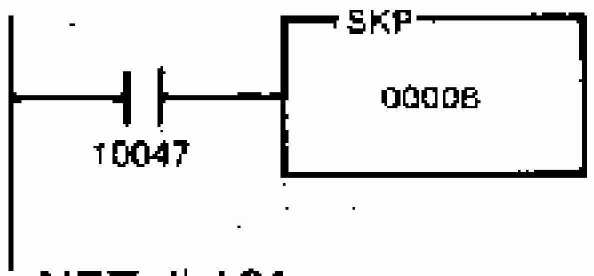
If J = 0 or J is greater than the number of networks following network N, processing of all networks following network N will be skipped.

In network N, all logics that follow the element of Skip according to the order of network solving (see (1) of 3.4) will be skipped.

NOTE

1. Power flow displayed on the P190CRT will be invalid for skipped networks. To prevent the user from misinterpreting data in skipped networks, an error message will appear at the lower left hand side of the P190 screen which says "POWER FLOW INVALID —NETWORK SKIPPED".
2. Note that the transitional contact of the coil included in a skipped network is not functioning properly.
(See (6) of 4. 2. 3.)

(3) Circuit Example



NET #101

When the input relay 10047 is ON, networks 101-108 will be skipped. If the network 105 is the last one, networks 101-105 will be skipped.

SECTION 5
INPUT/OUTPUT MODULES

5. 1 SPECIFICATIONS OF 1000 SERIES INPUT/OUTPUT MODULES

(1) B1051B 100 VAC Input Module

Table 5. 1 100 VAC Input Module Specifications

Items		Specifications
Type		JAMSC-B1051B
Number of Inputs		16-Inputs per module
Indicator		16-Input status LED's provided for each input, lighting up when input ON.
Electrical Character-istics	Input Conditions	ON level: ON at input voltage between 80 and 130 VAC continuous. OFF level: 30 VAC max
	Input Impedance	Approx 9 kΩ
	Input Current	Approx 11 mA (when supplying 100 VAC)
	Transient Voltage	200 VAC (1 cycle or less)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	Isolation Voltage	1500 VAC for 1 minute

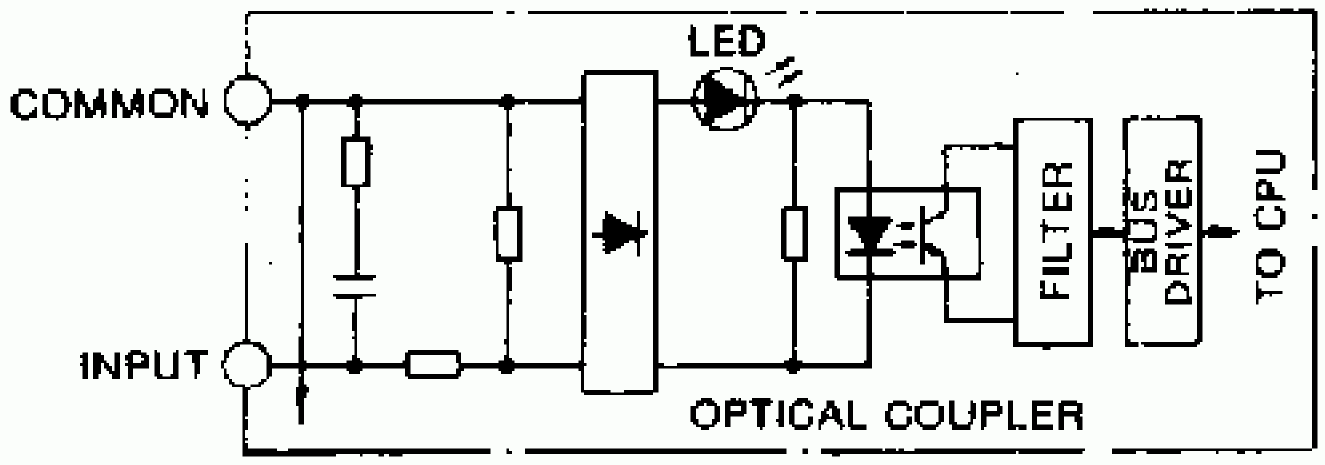


Fig. 5. 1 B1051B 100 VAC Input Module Simplified Schematic for One Circuit

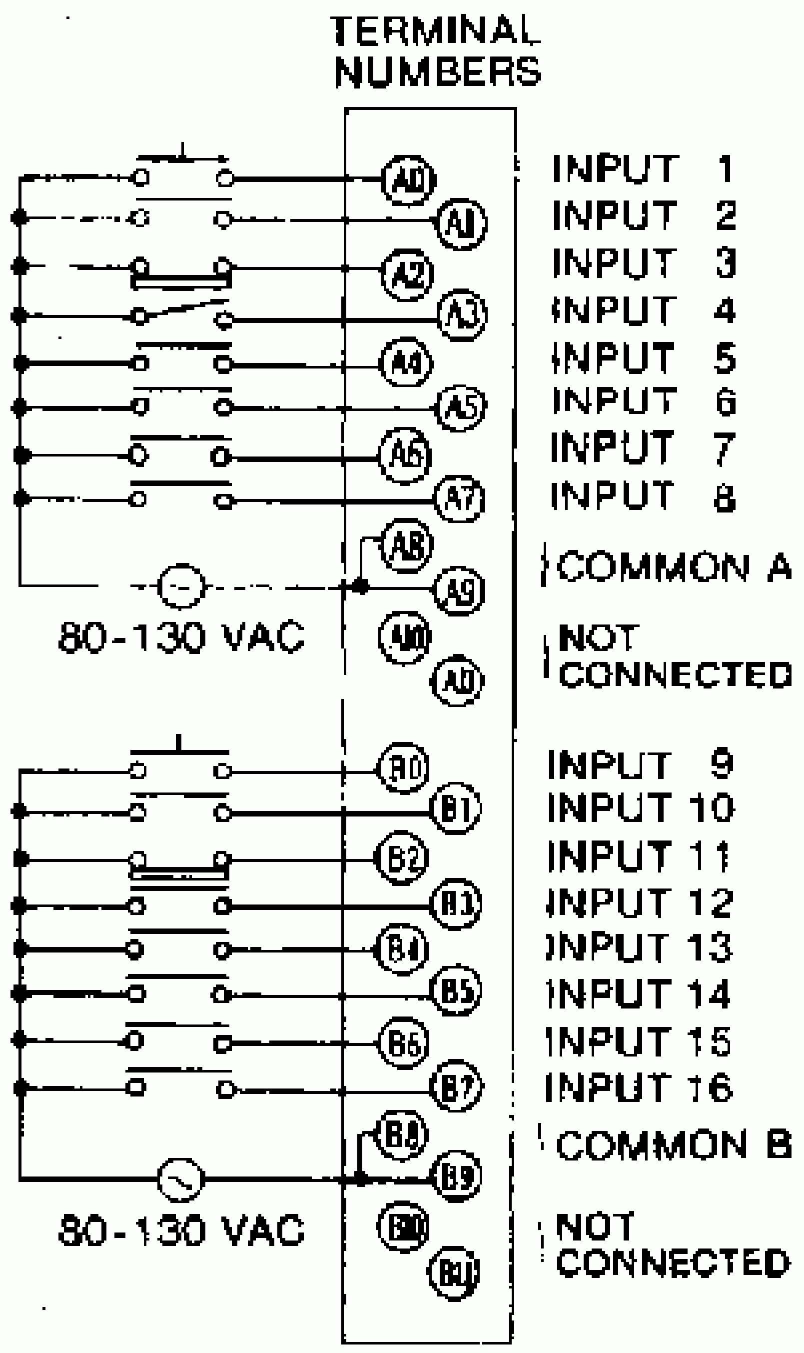


Fig. 5. 2 B1051B 100 VAC Input Module Terminal Numbering and Input Connections

(2) B1055 200 VAC Input Module

Table 5.2 200 VAC Input Module Specifications

Items		Specifications
Type		JAMSC-B1055
Number of Inputs		16-Inputs per module
Indicator		16-Input status LED's, provided for each input, lighting up when input ON.
Electrical Character-istics	Input Conditions	ON level: ON at input voltage between 160 and 260 VAC continuous. OFF level: 60 VAC max
	Input Impedance	Approx 23 kΩ
	Input Current	Approx 9 mA (when supplying 200 VAC)
	Transient Voltage	400 VAC (1 cycle or less)
	Response Time	OFF to ON: 10 ms maximum ON to OFF: 20 ms maximum
	Isolation Voltage	1500 VAC for 1 minute

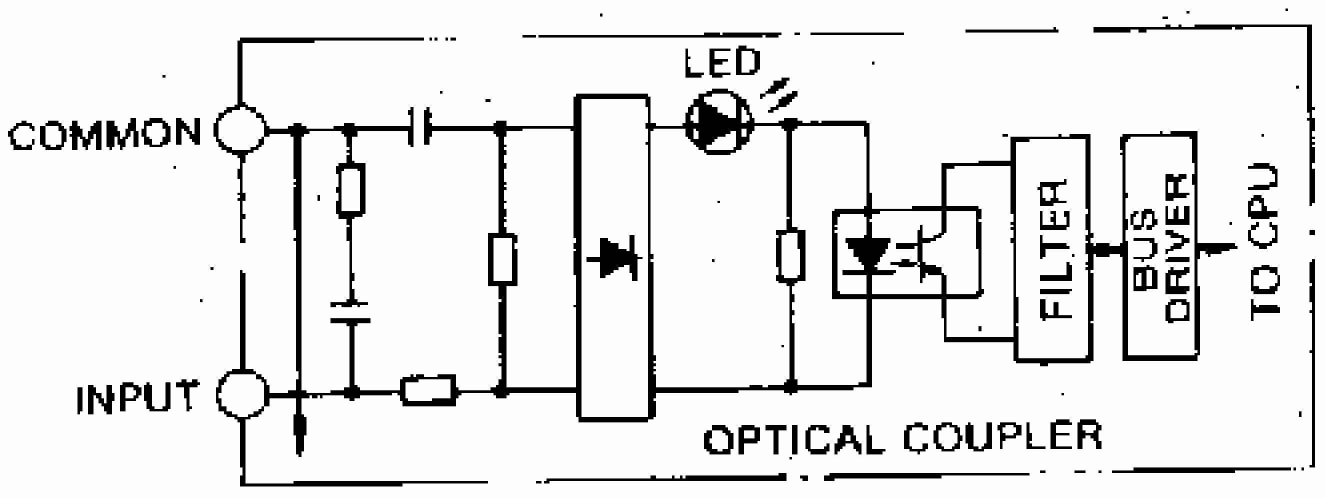


Fig. 5.3 B1055 200 VAC Input Module Simplified Schematic for One Circuit

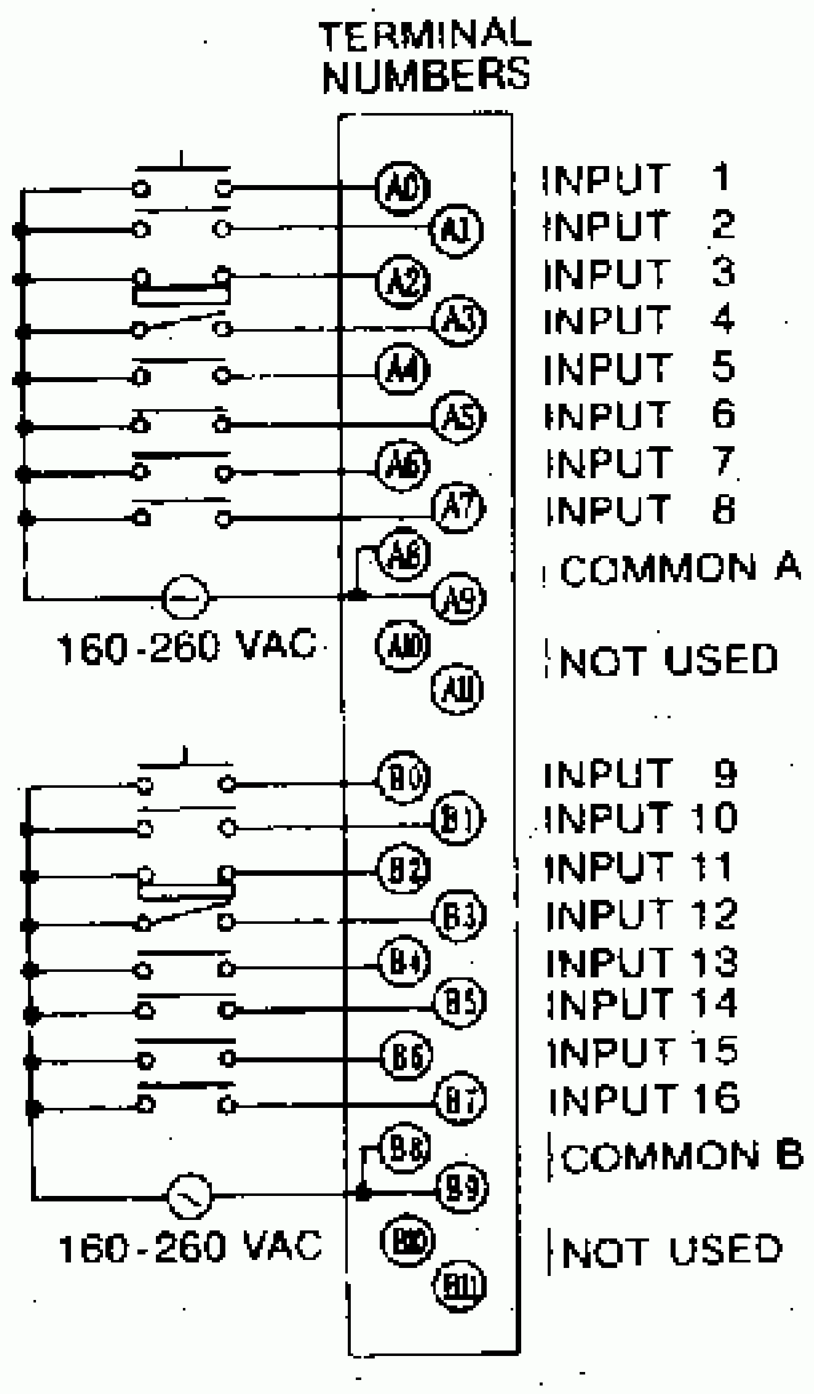


Fig. 5.4 B1055 200 VAC Input Terminal Numbering and Input Connections

(3) B1053 5 to 12 VDC Input Module

Table 5.3 5 to 12 VDC Input Module Specifications

Items		Specifications
Type		JAMSC-B1053
Number of Inputs		16-Inputs per module.
Indicators		16-Input status LED's, provided for each input, lighting up when input ON.
Electrical Characteristics	Input Conditions	External Source Voltage / ON Level / OFF Level: 5 VDC: 1.0 V max / 3.5 V min 12 VDC: 2.4 V max / 9.0 V min
	Input Impedance	Approx 1 k Ω
	Input Current	5 VDC: 4.5 mA; 12 VDC: 11.5 mA (measured at input terminals when input voltage is 0 V).
	Working Voltage	4.75 to 13.2 VDC
	Transient Voltage	15 VDC (Peak)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	External Power Supply Current (per Module)	External Source Voltage / Inputs OFF / Inputs ON: 5 VDC: 1 mA max / 180 mA max 12 VDC: 1 mA max / 470 mA max
	Isolation Voltage	1500 VAC for 1 minute

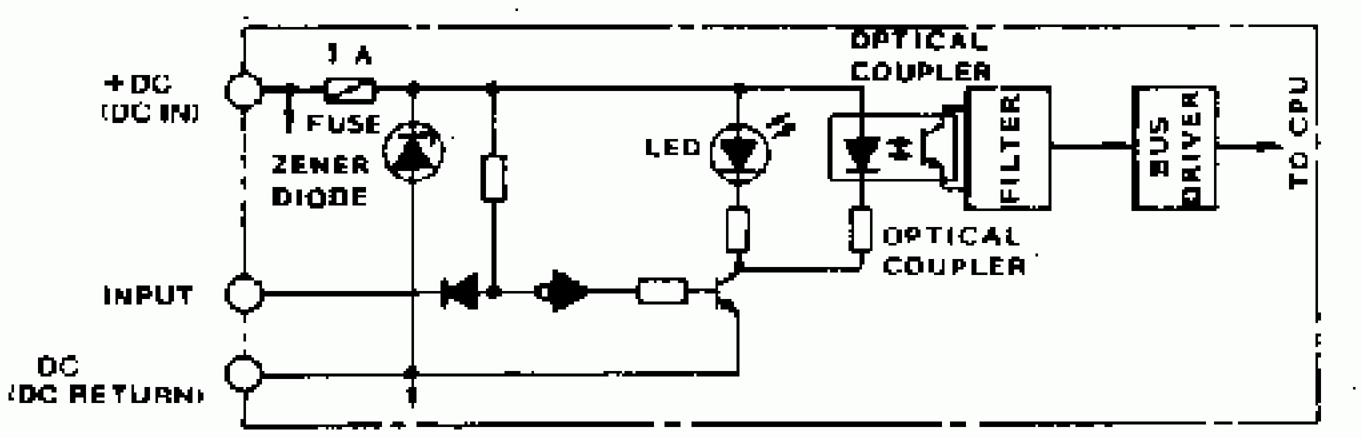


Fig. 5.5 B1053 5 to 12 VDC Input Module Simplified Schematic for One Circuit

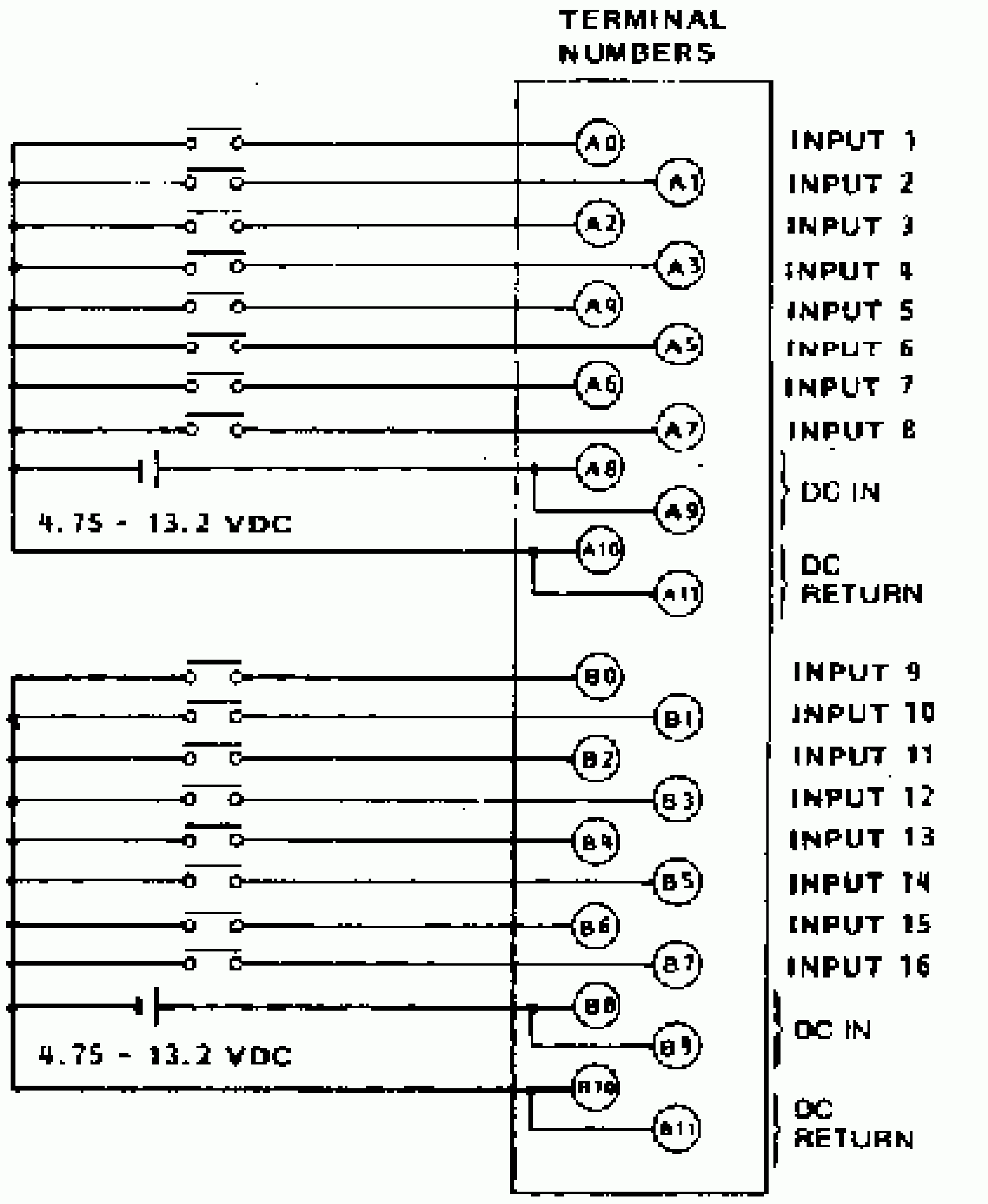


Fig. 5.6 B1053 5 to 12 VDC Input Module Terminal Numbering and Input Connections

(4) B1057 48 VDC Input Module

Table 5.4 48 VDC Input Module Specifications

Items		Specifications
Type		JAMSC-B1057
Number of Inputs		16-Inputs per module
Indicator		16-Input status LED's, provided for each input, lighting up when input ON.
Electrical Characteristics	Input Conditions	ON level: 16 VDC max (Negative logic) OFF level: 28 VDC min
	Input Impedance	Approx 4.8 kΩ
	Input Current	10 mA (measured at input terminals when input voltage is 0 V).
	Working Voltage	38 to 58 VDC
	Transient Voltage	70 V (Peak)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	External Power Supply Current (per Module)	External Source Voltage: 48 VDC; Inputs OFF: 1 mA max; Inputs ON: 160 mA
	Isolation Voltage	1500 VAC for 1 minute

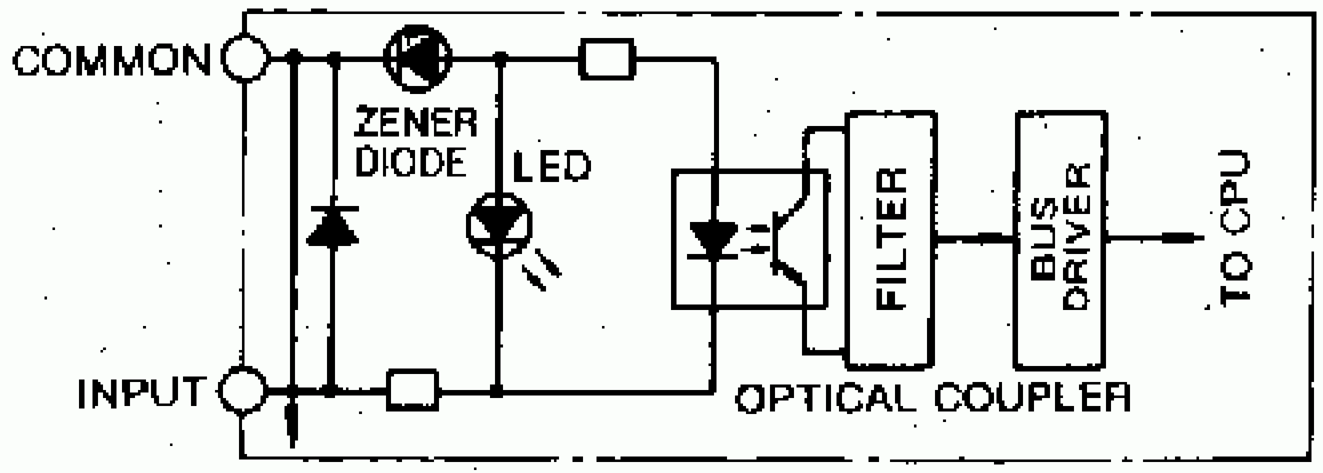


Fig. 5.7 B1057 48 VDC Input Module Simplified Schematic for One Circuit

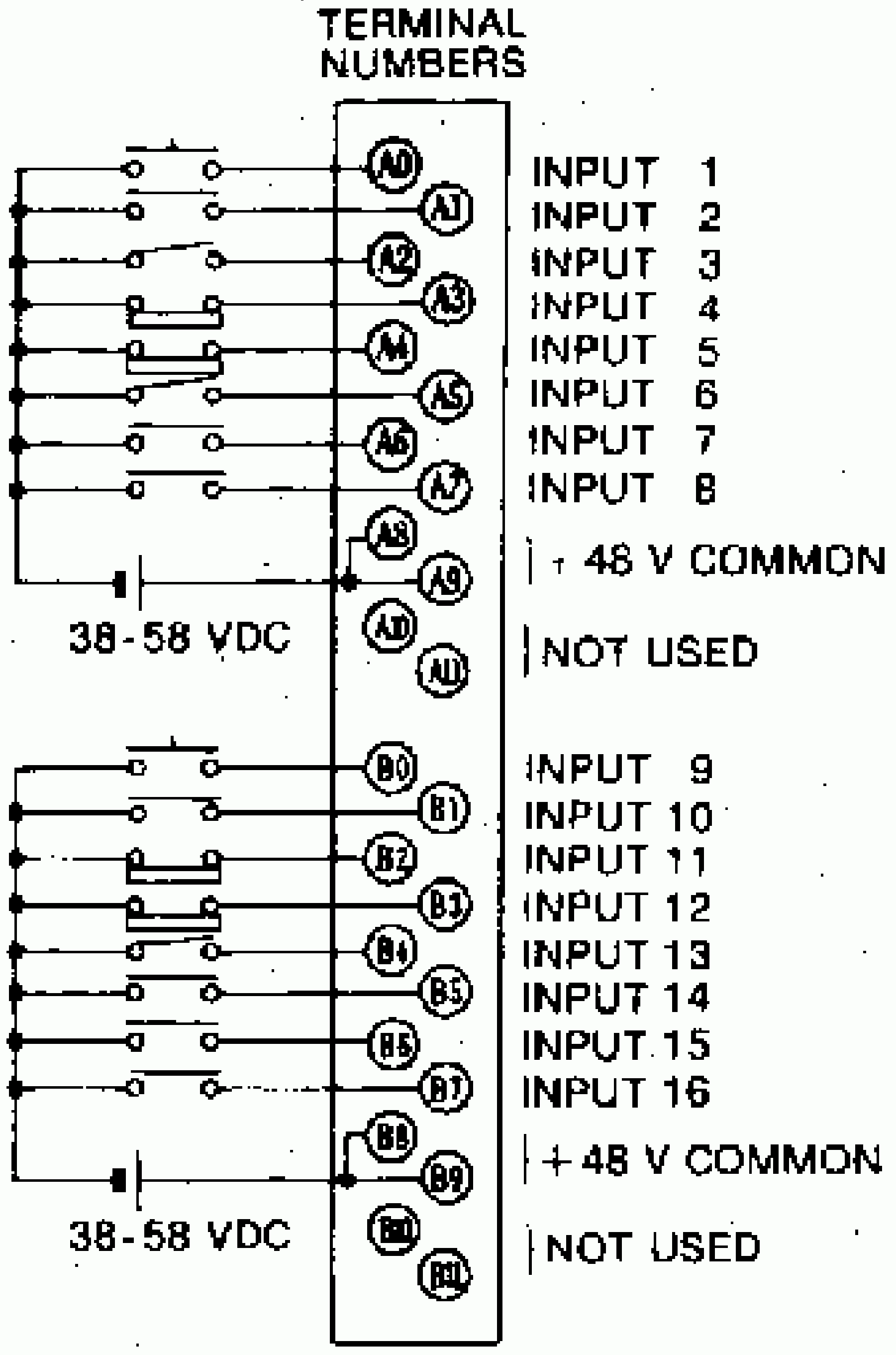


Fig. 5.8 B1057 48 VDC Input Module Terminal Numbering and Input Connections

(5) B1059C 24 VDC Input Module

Table 5.5 24 VDC Input Module Specifications

Items		Specifications		
Type		JAMSC-B1059C		
Number of Inputs		16-inputs per module		
Indicator		16-input status LED's, provided for each input, lighting up when input ON.		
Electrical Characteristics	Input Conditions	ON level: 8 VDC max (Negative logic) OFF level: 14 VDC min		
	Input Impedance	Approx 2.4 kΩ		
	Input Current	10 mA (measured at input terminals when input voltage is 0 V).		
	Working Voltage	19 to 29 VDC		
	Transient Voltage	35 V (Peak)		
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max		
	External Power Supply Current (per Module)	External Source Voltage	Inputs OFF	Inputs ON
		24 VDC	1 mA max	176 mA
	Isolation Voltage	1500 VAC for 1 minute		

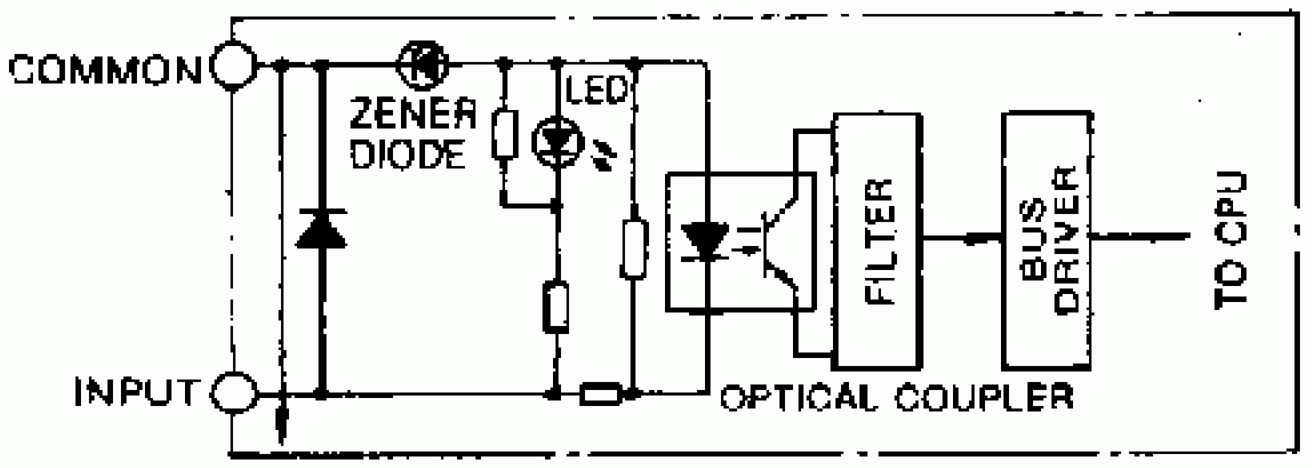


Fig. 5.9 B1059C 24 VDC Input Module
Simplified Schematic for One Circuit

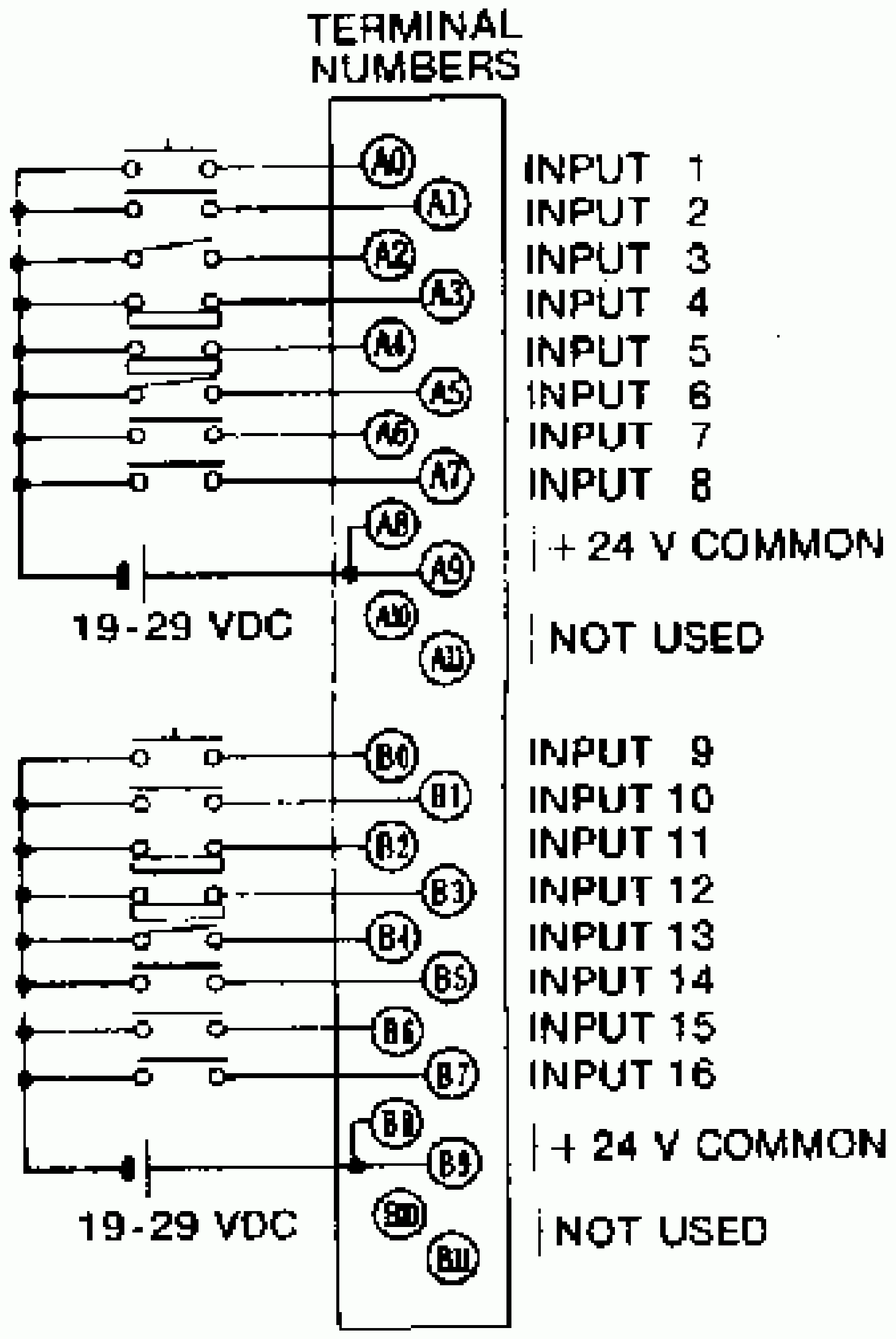


Fig. 5.10 B1059C 24 VDC Input Module
Terminal Numbering and
Input Connections

(6) B1061 24 VDC Input Module

Table 5.6 24 VDC Input Module Specifications

Items		Specifications
Type		JAMSC-B1061
Number of Inputs		64-Inputs per module
Indicators		Upper (1 to 32) inputs/lower (1 to 32) inputs status LED's selection, lighting up when input ON.
Electrical Characteristics	Input Condition	ON level: 6 VDC max (Negative logic) OFF level: 13 VDC min
	Input Impedance	Approx 4.8 kΩ
	Input Current	5 mA (measured at input terminals when input voltage is 0 V).
	Working Voltage	20.4 to 26.4 VDC
	Transient Voltage	35 V (Peak)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	External Power Supply Current (per Module)	External Source Voltage: 24 VDC — Inputs OFF: 360 mA max — Inputs ON: 1 mA max
	Isolation Voltage	1500 VAC for 1 minute

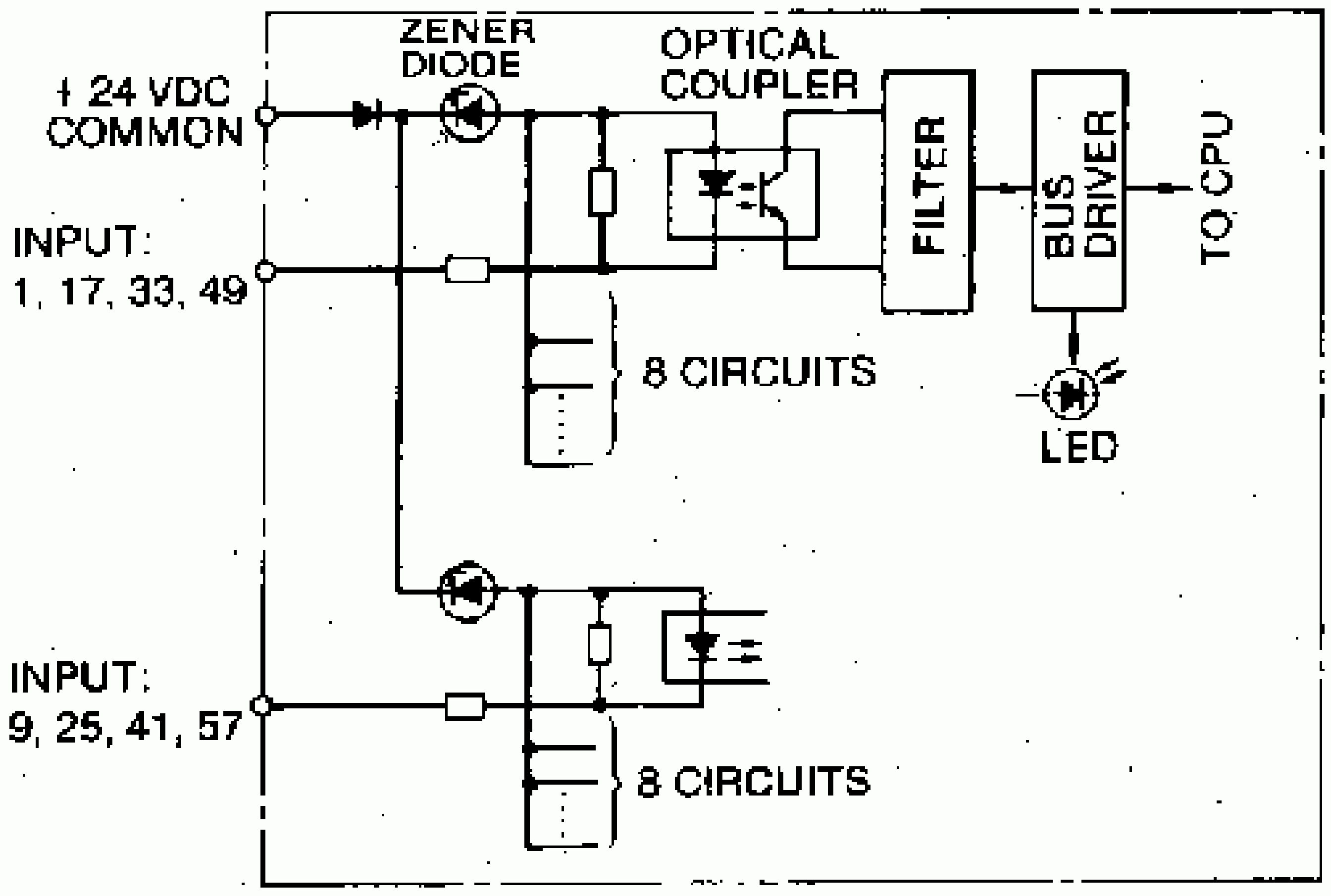


Fig. 5.11 B1061 24 VDC Input Module Simplified Schematic for One Circuit

Note: B1061 is provided with two connectors (connectors 1 and 2): These connectors have same terminal numbers and input connections.

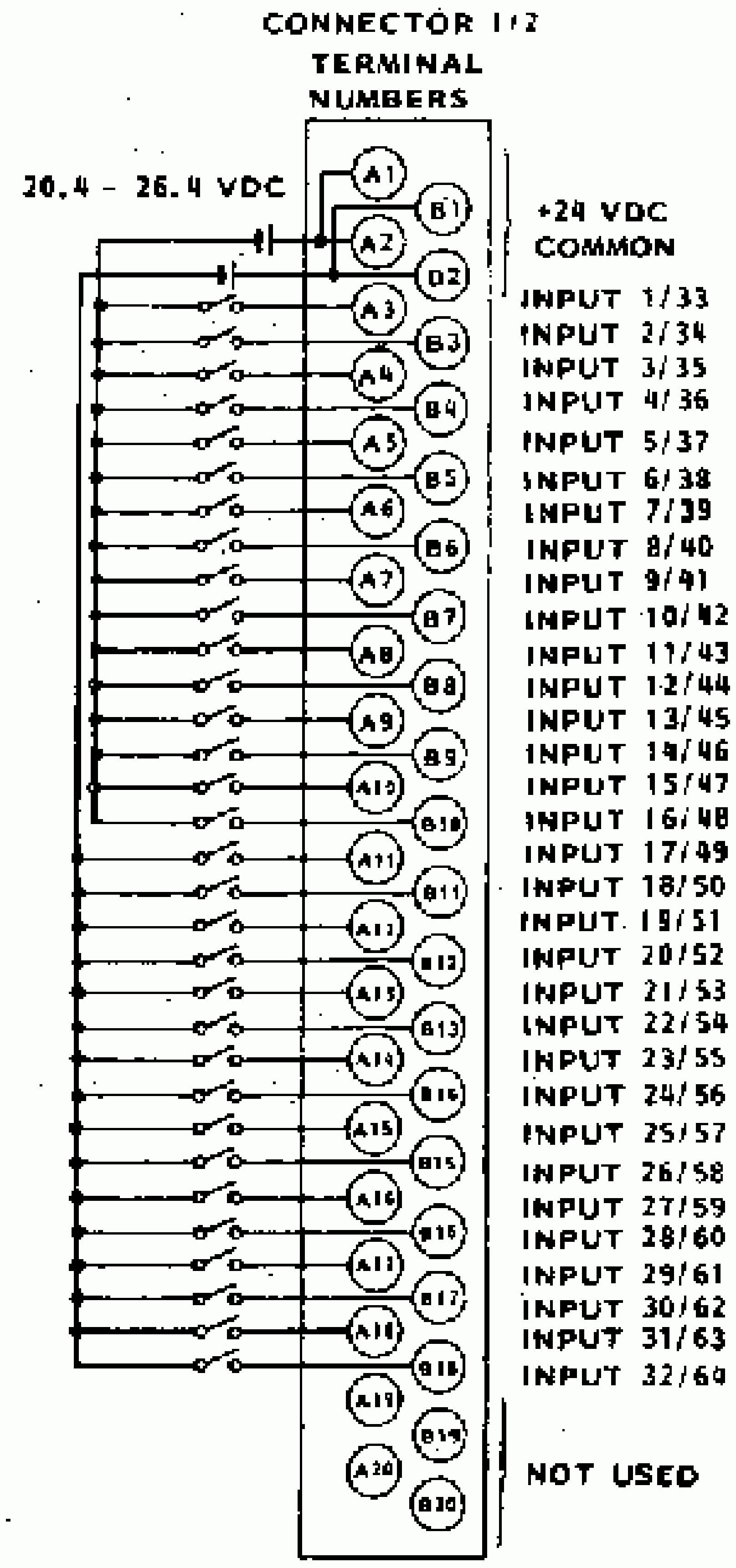


Fig. 5.12 B1061 24 VDC Input Module Terminal Numbering and Input Connections

(7) B1063 24 VDC Input Module

Table 5.7 24 VDC Input Module Specifications

Items		Specifications
Type		JAMSC-B1063
Number of Inputs		32-Inputs per module
Indicators		32-Input status LED's, provided for each input, lighting up when input ON.
Electrical Character-istics	Input Condition	ON level: 6 VDC max (Negative logic) OFF level: 13 VDC min
	Input Impedance	Approx 2.4 kΩ
	Input Current	10 mA (measured at input terminals when input voltage is 0 V.)
	Working Voltage	20.4 to 26.4 VDC
	Transient Voltage	35 V (Peak)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	External Power Supply Current (per Module)	External Source Voltage: 24 VDC; Inputs OFF: 320 mA max; Inputs ON: 1 mA max
	Isolation Voltage	1500 VAC for 1 minute

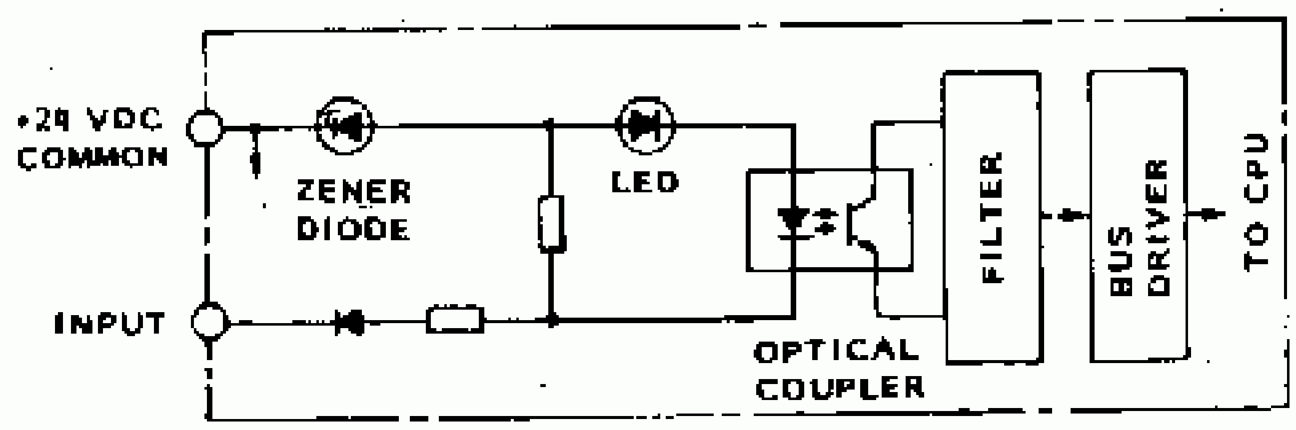


Fig. 5.13 B1063 24 VDC Input Module Simplified Schematic for One Circuit

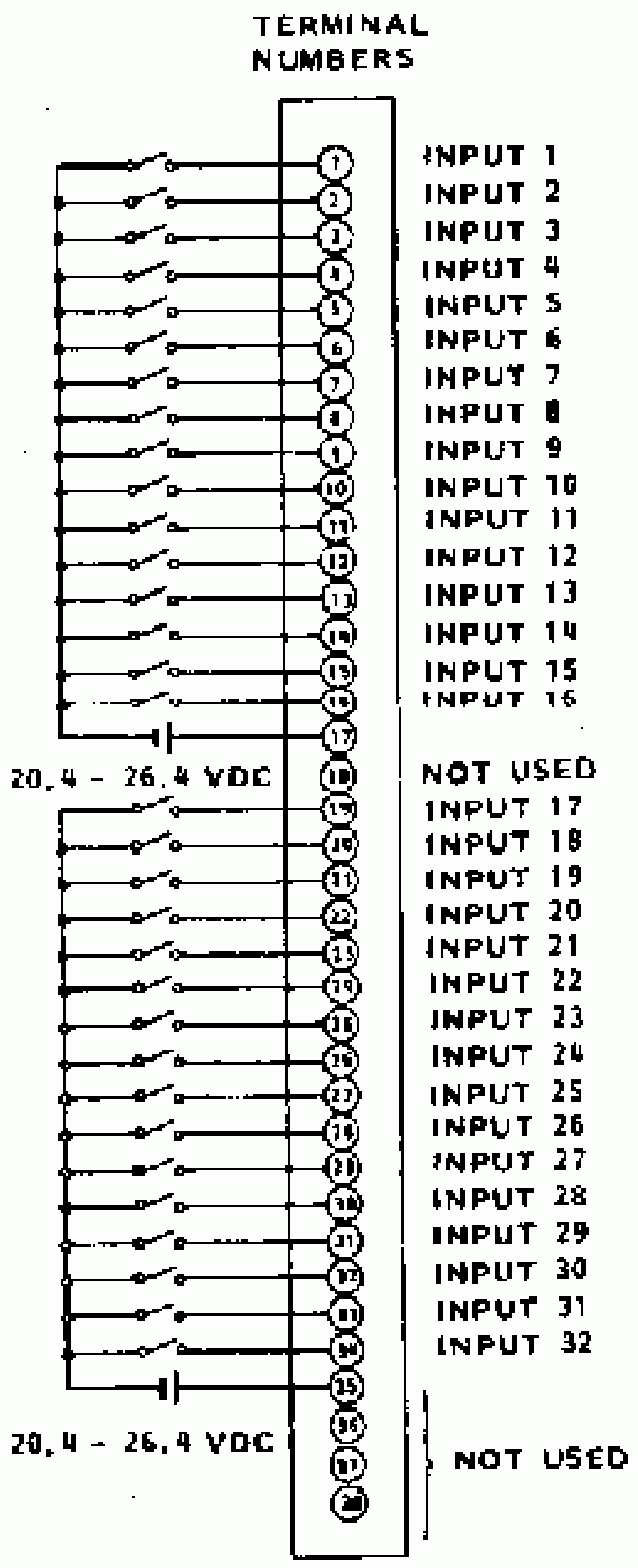


Fig. 5.14 B1063 24 VDC Input Module Terminal Numbering and Input Connections

(8) B1065 24 VDC Input Module

Table 5.8 24 VDC Input Module Specifications

Items		Specifications
Type		JAMSC-B1065
Number of Inputs		32-Inputs per module
Indicator		32-input status LED's, provided for each input, lighting up when input ON.
Electrical Characteristics	Input Condition	ON level: 8 VDC max (Negative logic) OFF level: 14 VDC min
	Input Impedance	Approx 4 kΩ
	Input Current	6 mA (measured at input terminals when input voltage is 0 V).
	Working Voltage	19 to 29 VDC
	Transient Voltage	35 V (Peak)
	Response Time	OFF to ON: 10 ms max ON to OFF: 20 ms max
	External Power Supply Current (per Module)	External Source Voltage: 24 VDC; Inputs OFF: 1 mA max; Inputs ON: 192 mA
	Isolation Voltage	1500 VAC for 1 minute

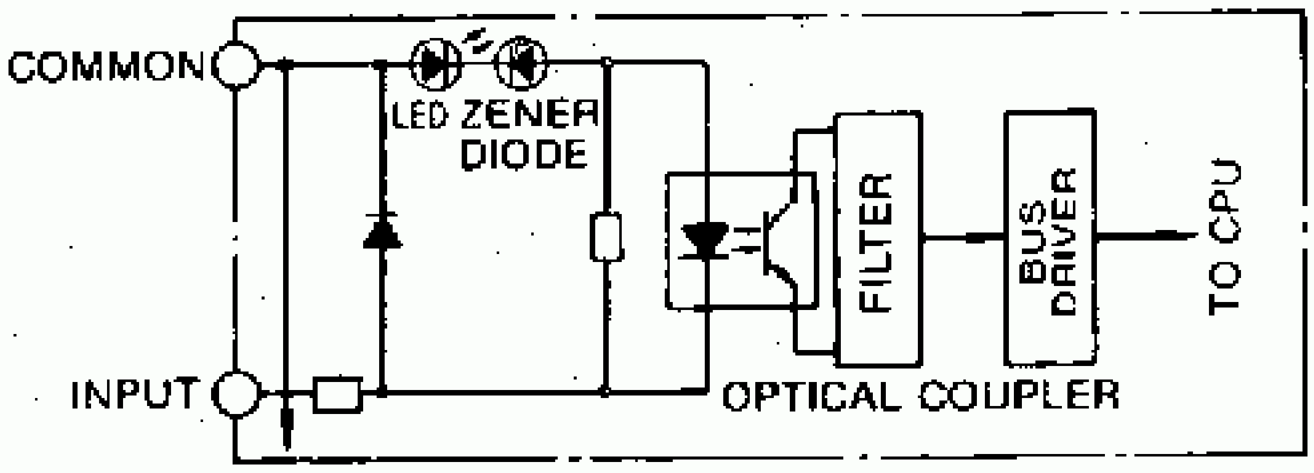


Fig. 5.15 B1065 24 VDC Input Module
Simplified Schematic for One Circuit

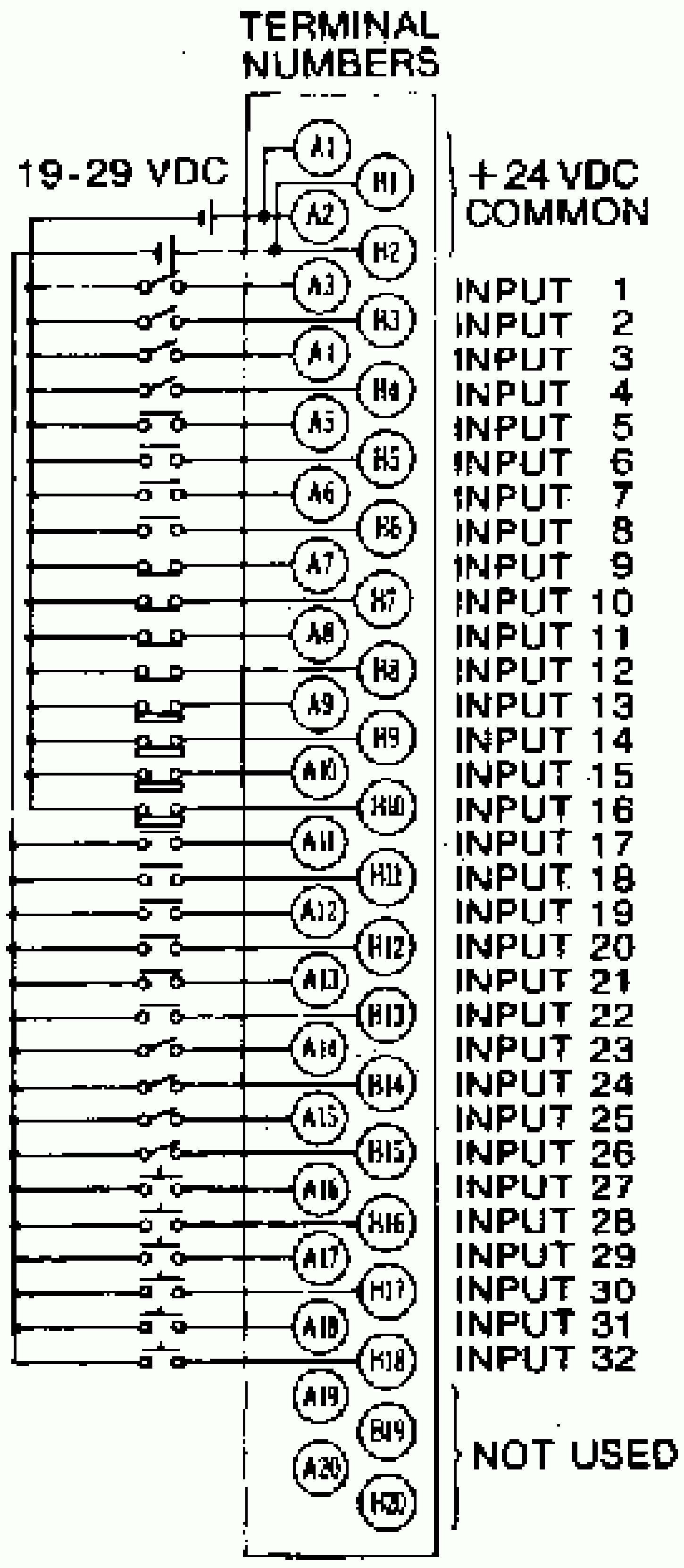


Fig. 5.16 B1065 24 VDC Input Module
Terminal Numbering and
Input Connections

(9) B1050 100 VAC Output Module

Table 5. 9 100 VAC Output Module Specifications

Items			Specifications
Type			JAMSC-B1050
Number of Outputs			16-Outputs per module
Indicators			16-Output status LED's, provided for each output, lighting up when output ON (at internal logic side). 2-Blown fuse LED's, provided for every 8 outputs, lighting up with fuse blown.
Fuse Rating			5 A (Glass tube fuse)
Electrical Character-istics	Load Voltage	Working Voltage	80 to 130 VAC
		Transient Voltage	200 VAC for 1 cycle
		Average ON Voltage	1.5 V rms (Load current: 2 A rms)
	Load Current	ON Current	2 A rms per output; 5 A per 8 outputs
		OFF Current	2 mA rms max
		Inrush Current	15 A (10 ms)
		Min Load Current	10 mA rms
	Response Time		OFF to ON: 10 ms max ON to OFF: 10 ms max
	Isolation Voltage		1500 VAC for 1 minute

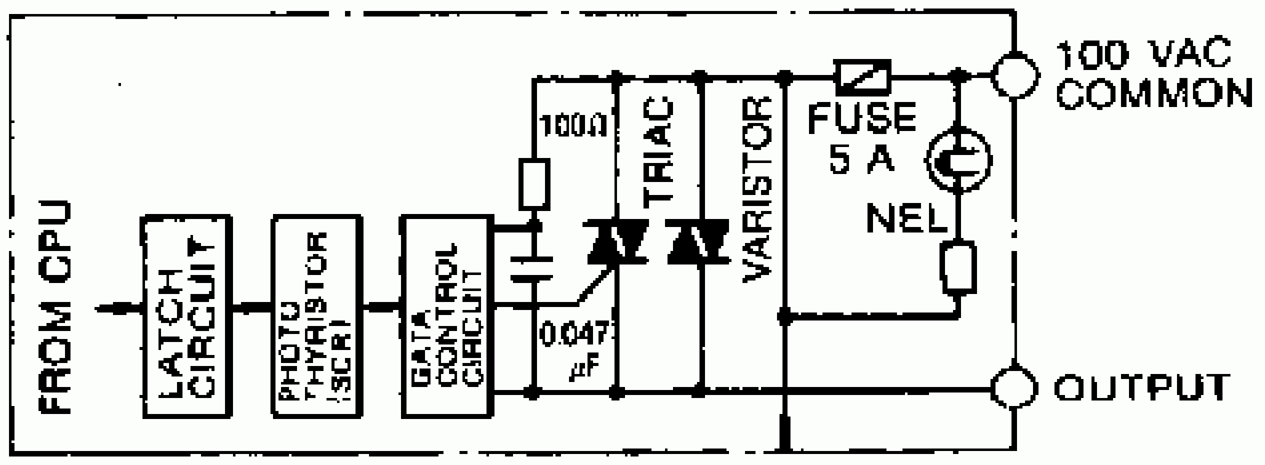


Fig. 5. 17 B1050 100 VAC Output Module Simplified Schematic for One Circuit

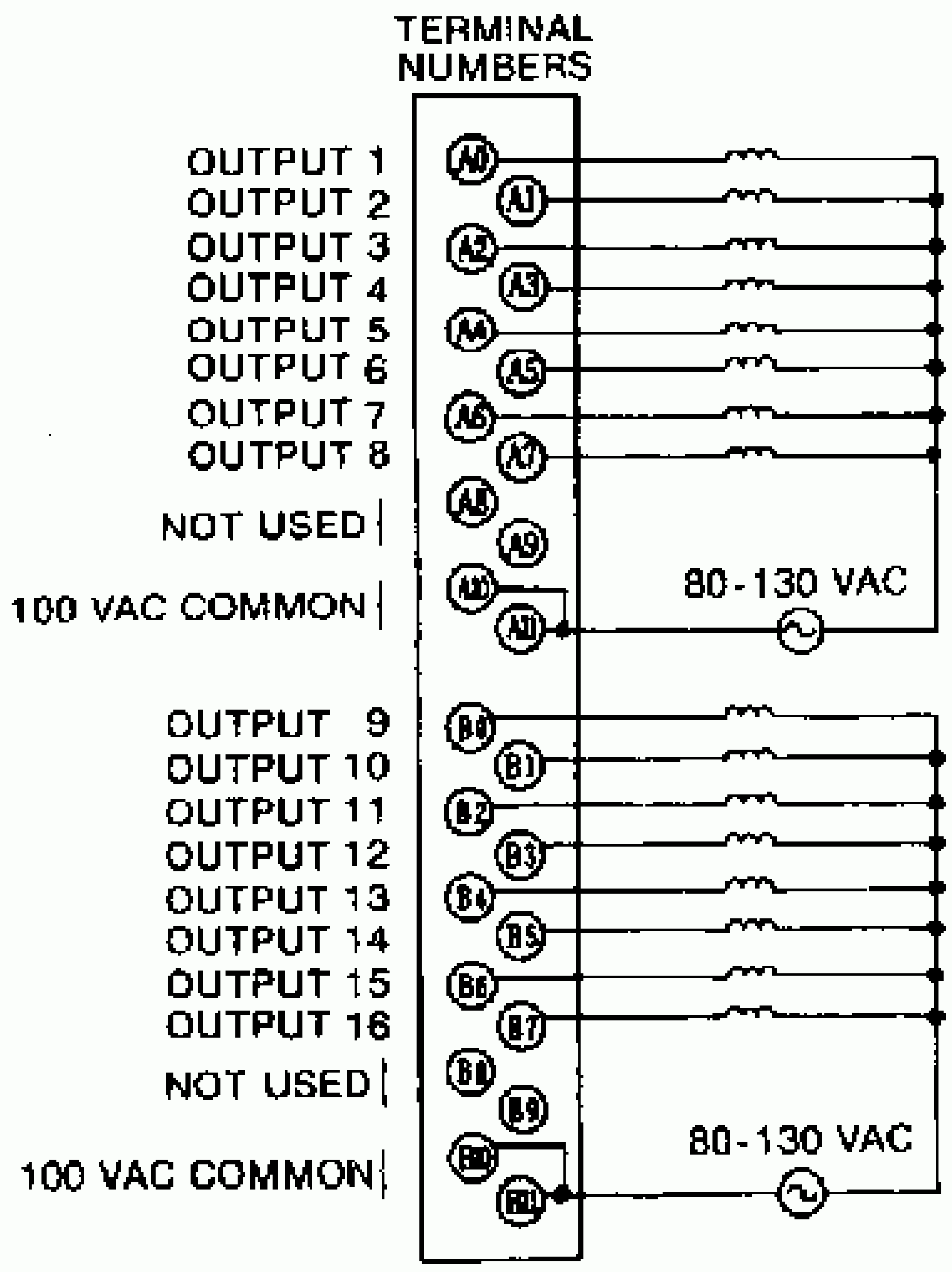


Fig. 5. 18 B1050 100 VAC Output Module Terminal Numbering and Output Connections

(10) B1054 200 VAC Output Module

Table 5.10 200 VAC Output Module Specifications

Items			Specifications
Type			JAMSC-B1054
Number of Outputs			16-Outputs per module
Indicators			16-Output staus LED's, provided for each output, lighting up when output ON (at internal logic side). 2-Blown fuse LED's, provided for every 8 outputs, lighting up with fuse blown.
Fuse Rating			5 A (Glass tube fuse)
Electrical Character-istics	Load Voltage	Working Voltage	160 to 260 VAC
		Transient Voltage	400 VAC for 1 cycle
		Average ON Voltage	1.5 V rms (Load current: 1 A rms)
	Load Current	ON Current	1 A rms per output, 5 A per 8 outputs
		OFF Current	2 mA rms max
		Inrush Current	15 A (10 ms)
		Min Load Current	10 mA rms
	Response Time		OFF to ON: 10 ms max ON to OFF: 10 ms max
	Isolation Voltage		1500 VAC for 1 minute

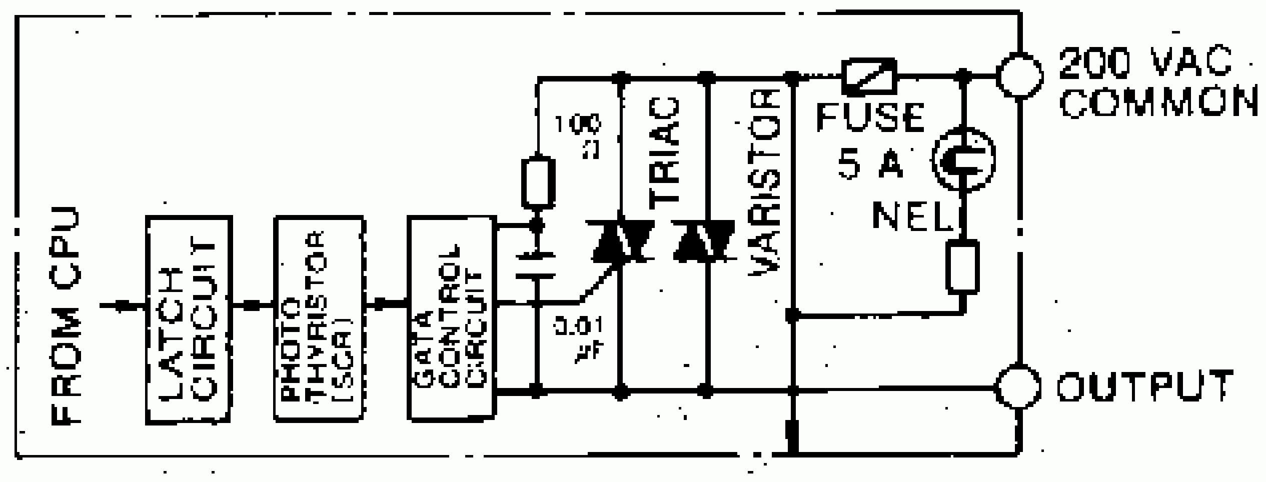


Fig. 5.19 B1054 200 VAC Output Module Simplified Schematic for One Circuit

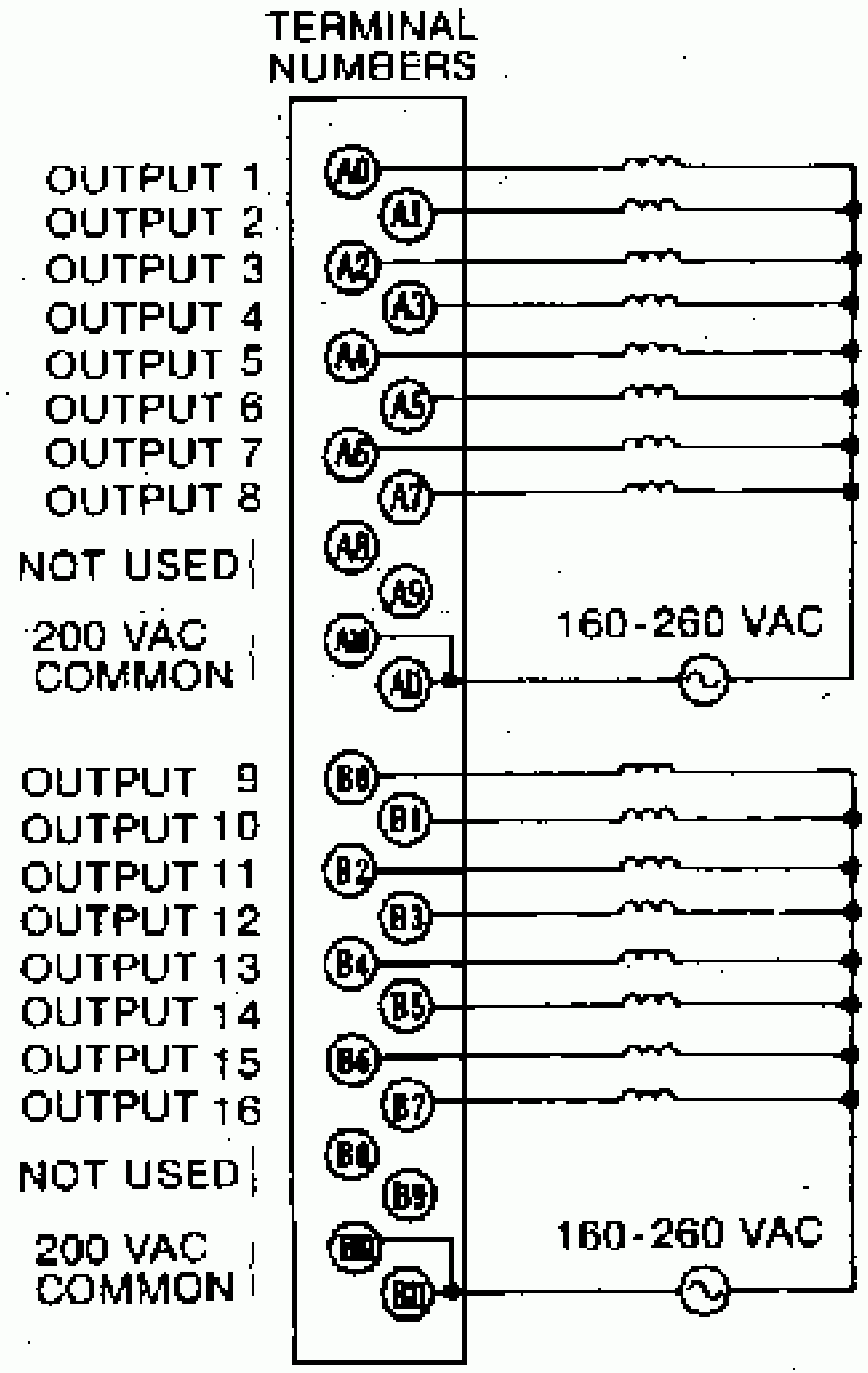


Fig. 5.20 B1054 200 VAC Output Module Terminal Numbering and Output Connections

(11) B1052 5 to 12 VDC Output Module

Table 5. 11 5 to 12 VDC Output Module Specifications

<table>
<tr><th colspan="3">Items</th><th>Specifications</th></tr>
<tr><td colspan="3">Type</td><td>JAMSC-B1052</td></tr>
<tr><td colspan="3">Number of Outputs</td><td>16-Outputs per module</td></tr>
<tr><td colspan="3">Indicators</td><td>16-Output status LED's, provided for each output, lighting up when output ON.
2-Blown fuse LED's provided for every 8 outputs, lighting up with fuse blown</td></tr>
<tr><td colspan="3">Fuse Rating</td><td>3 A (Glass tube fuse)</td></tr>
<tr><td rowspan="9">Electrical Character-istics</td><td rowspan="3">Load Voltage</td><td>Working Voltage</td><td>4.75 to 13.2 VDC</td></tr>
<tr><td>Transient Voltage</td><td>18 VDC (Peak)</td></tr>
<tr><td>Average ON Voltage</td><td>0.6 V max (Load current: 0.3 A)</td></tr>
<tr><td rowspan="3">Load Current</td><td>ON Current</td><td>5 VDC: 0.1 A per output
12 VDC: 0.3 A per output</td></tr>
<tr><td>OFF Current</td><td>0.2 mA max</td></tr>
<tr><td>Inrush Current</td><td>1 A (10 ms)</td></tr>
<tr><td colspan="2">Response Time</td><td>OFF to ON: 1 ms max
ON to OFF: 1 ms max</td></tr>
<tr><td colspan="2">External Power Supply Current (per Module)</td><td><table><tr><th>External Source Voltage</th><th>Outputs OFF</th><th>Outputs ON</th></tr><tr><td>5 VDC</td><td>1 mA max</td><td>110 mA max</td></tr><tr><td>12 VDC</td><td>1 mA max</td><td>320 mA max</td></tr></table></td></tr>
<tr><td colspan="2">Isolation Voltage</td><td>1500 VAC for 1 minute</td></tr>
</table>

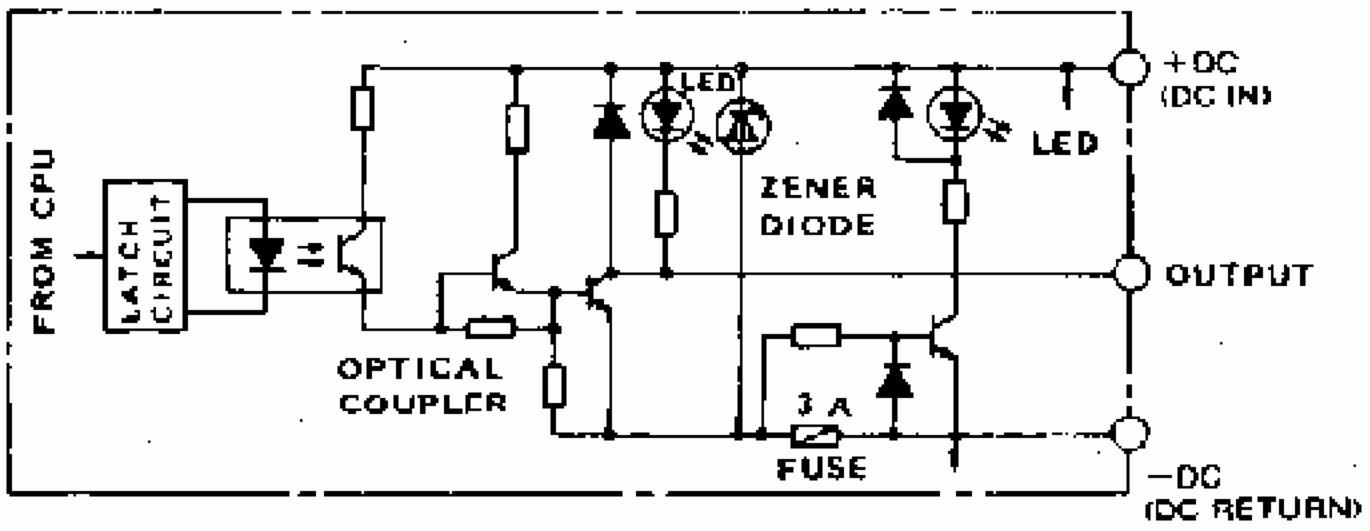


Fig. 5. 21 B1052 5 to 12 VDC Output Module Simplified Schematic for One Circuit

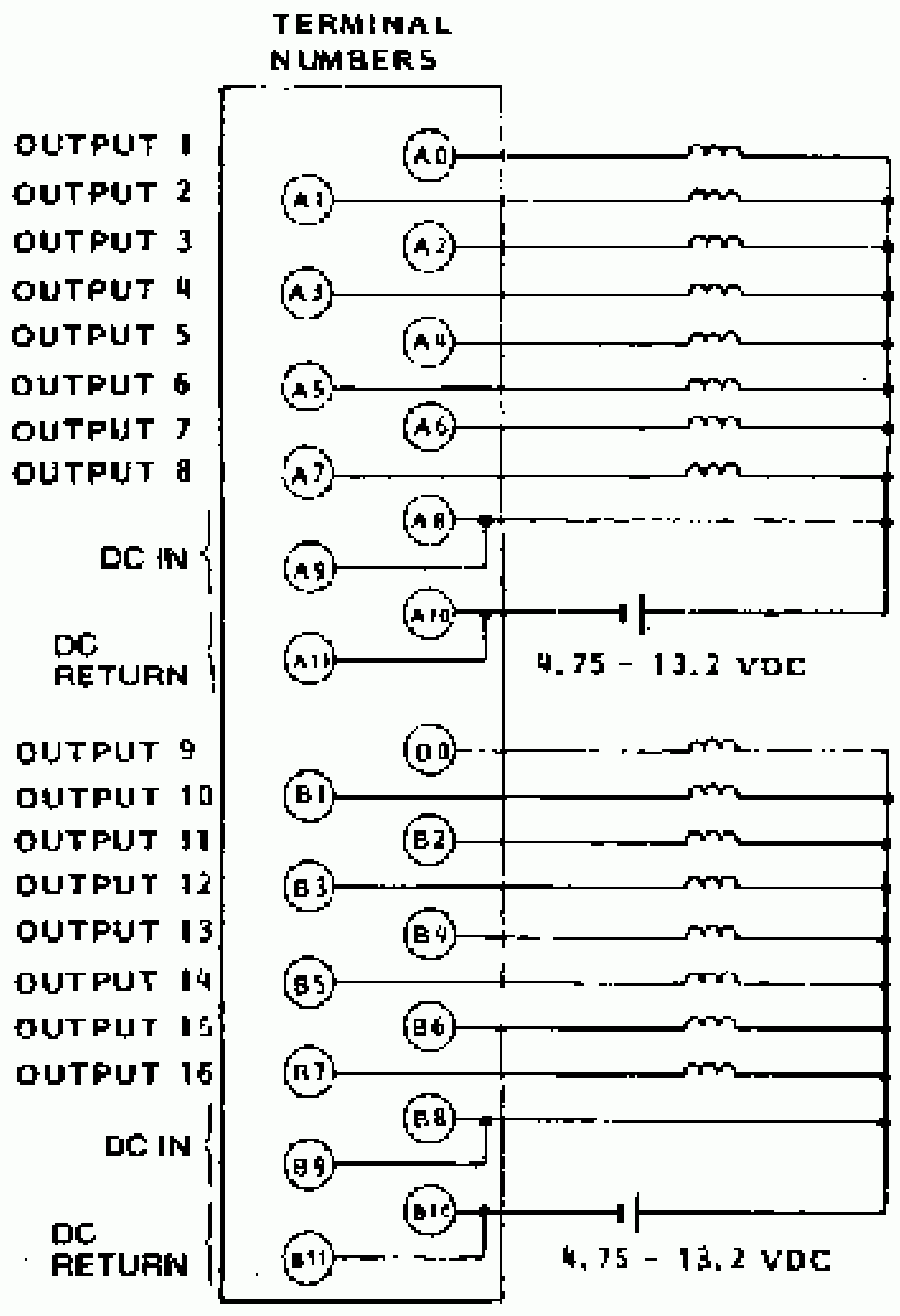


Fig. 5. 22 B1052 5 to 12 VDC Output Module Terminal Numbering and Output Connections

(12) B1056 48 VDC Output Module

Table 5.12 48 VDC Output Module Specifications

Items			Specifications
Type			JAMSC-B1056
Number of Outputs			16-Outputs per module
Indicators			16-Output status LED's, provided for each output, lighting up when output ON. 2-Blown fuse LED's, provided for every 8 outputs, lighting up with fuse blown.
Fuse Rating			5 A (Glass tube fuse)
Electrical Character-istics	Load Voltage	Working Voltage	38 to 58 VDC
		Transient Voltage	70 VDC (Peak)
		Average ON Voltage	1.5 V (Load current: 2 A)
	Load Current	ON Current	2 A per output; 5 A per 8 outputs
		OFF Current	1 mA max
		Inrush Current	7 A (10 ms)
	Response Time		OFF to ON: 1 ms max ON to OFF: 1 ms max External Source Voltage: 48 VDC — Outputs OFF: 20 mA, Outputs ON: 190 mA
	Isolation Voltage		1500 VAC for 1 minute

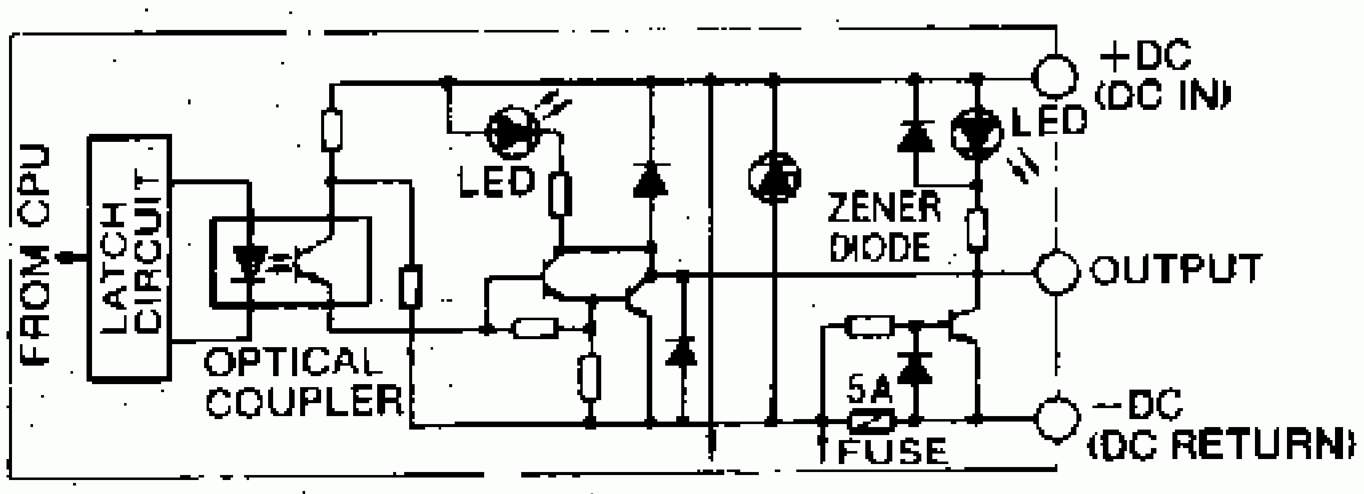


Fig. 5.23 B1056 48 VDC Output Module Simplified Schematic for One Circuit

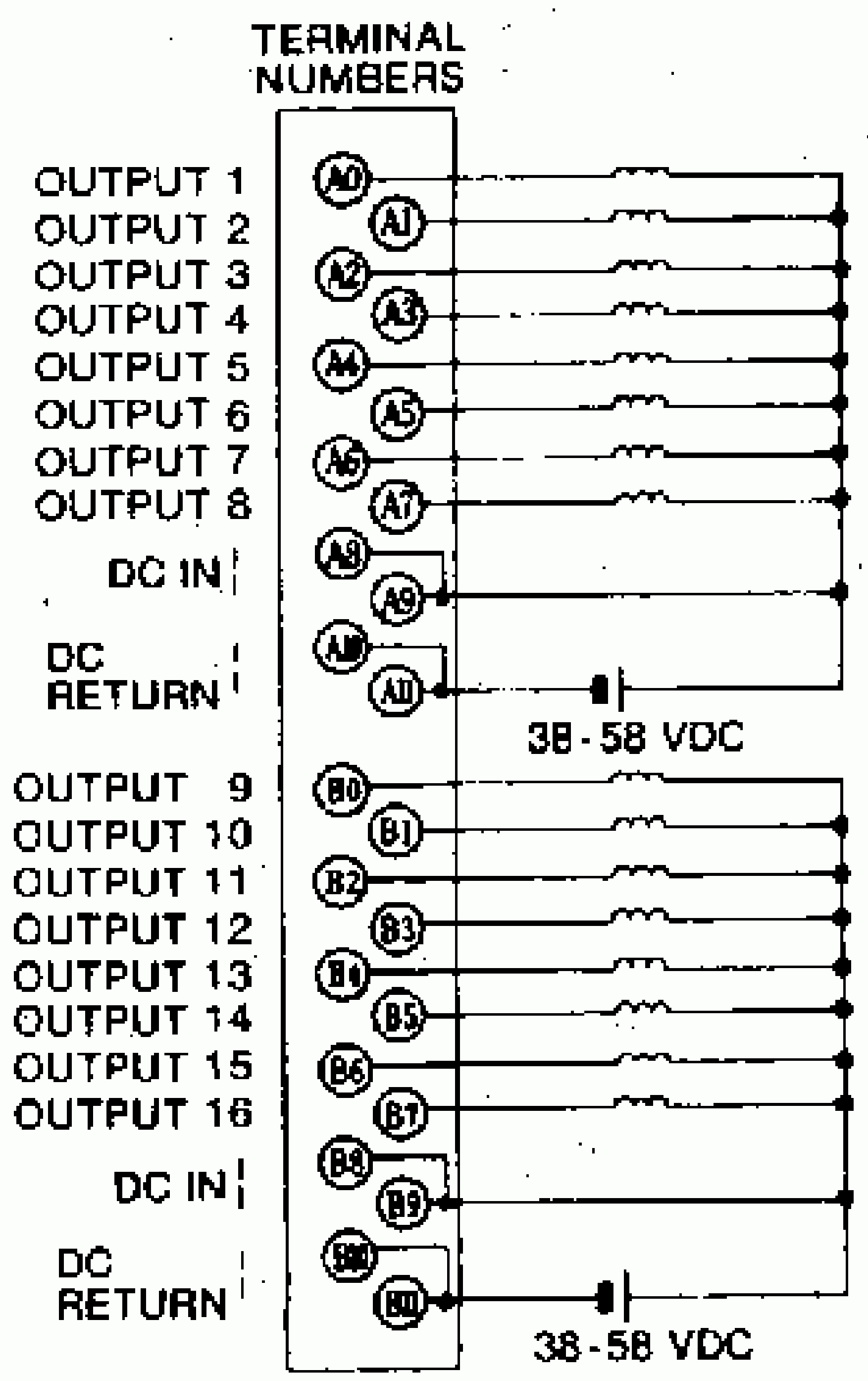


Fig. 5.24 B1056 48 VDC Output Module Terminal Numbering and Output Connections

(13) B1058 24 VDC Output Module

Table 5.13 24 VDC Output Module Sepcifications

Items			Specifications
Type			JAMSC-B1058
Number of Outputs			16-Outputs per module
Indicators			16-Output status LED's, provided for each output, lighting up when output ON. 2-Blown fuse LED's, provided for every 8 outputs, lighting up with fuse blown.
Fuse Rating			5 A (Glass tube fuse)
Electrical Character-istics	Load Voltage	Working Voltage	19 to 29 VDC
		Transient Voltage	35 VDC (Peak)
		Average ON Voltage	1.5 V (Load current: 2 A)
	Load Current	ON Current	2 A per output; 5 A per 8 outputs
		OFF Current	1 mA max
		Inrush Current	7 A (10 ms)
	Response Time		OFF to ON: 1 ms max ON to OFF: 1 ms max External Source Voltage: 24 VDC; Outputs OFF: 20 mA; Outputs ON: 190 mA
	Isolation Voltage		1500 VAC for 1 minute

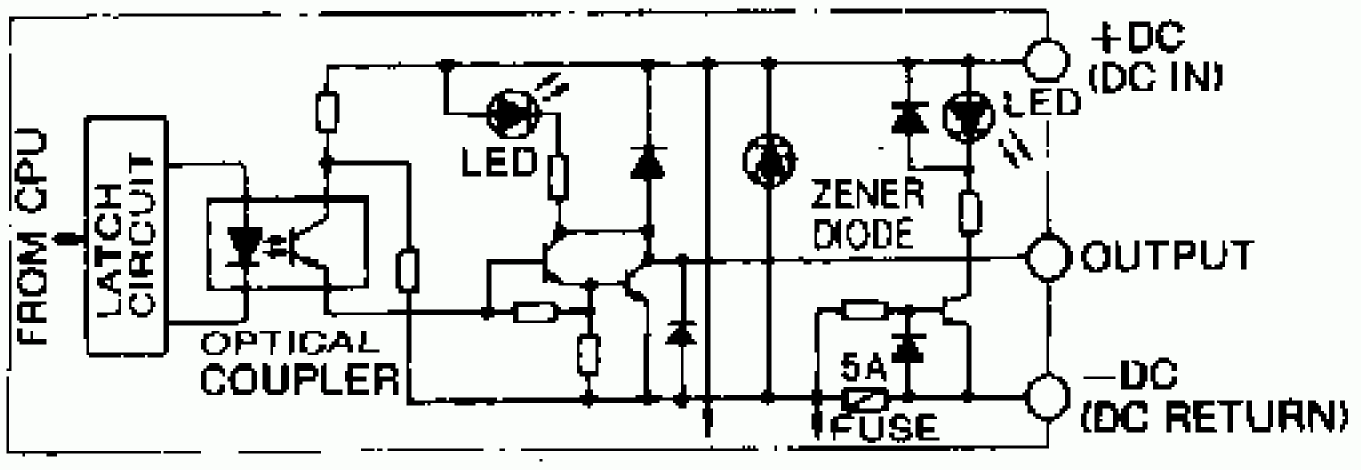


Fig. 5.25 B1058 24 VDC Output Module Simplified Schematic for One Circuit

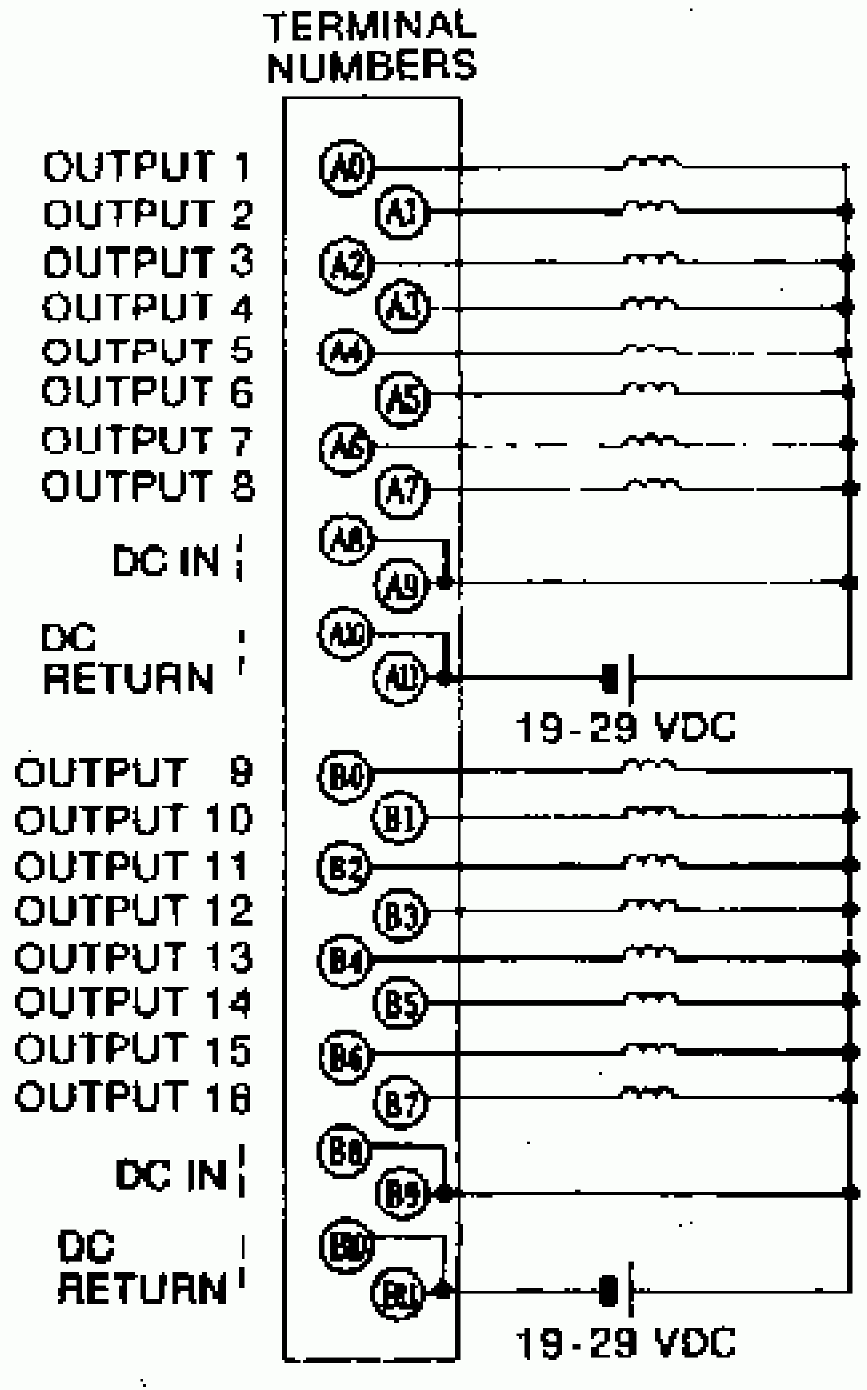


Fig. 5.26 B1058 24 VDC Output Module Terminal Numbering and Output Connections

(14) B1060 24 VDC Output Module

Table 5.14 24 VDC Output Module Specifications

Items			Specifications
Type			JAMSC-B1060
Number of Outputs			64-Outputs per module
Indicators			Upper (1 to 32) outputs/lower (1 to 32) outputs status LED's selection, lighting up when output ON.
Electrical Characteristics	Load Voltage	Working Voltage	20.4 to 26.4 VDC
		Transient Voltage	35 VDC (Peak)
		Average ON Voltage	2.1 V max (Load current: 0.1 A)
	Load Current	ON Current	0.1 A per output; 0.4 A per 8 outputs
		OFF Current	0.2 mA max
		Inrush Current	0.5 A
	Response Time		OFF to ON: 1 ms max ON to OFF: 1 ms max
	External Power Supply Current (per Module)		External Source Voltage: 24 VDC; Inputs OFF: 1 mA max; Inputs ON: 60 mA max
	Isolation Voltage		1500 VAC for 1 minute

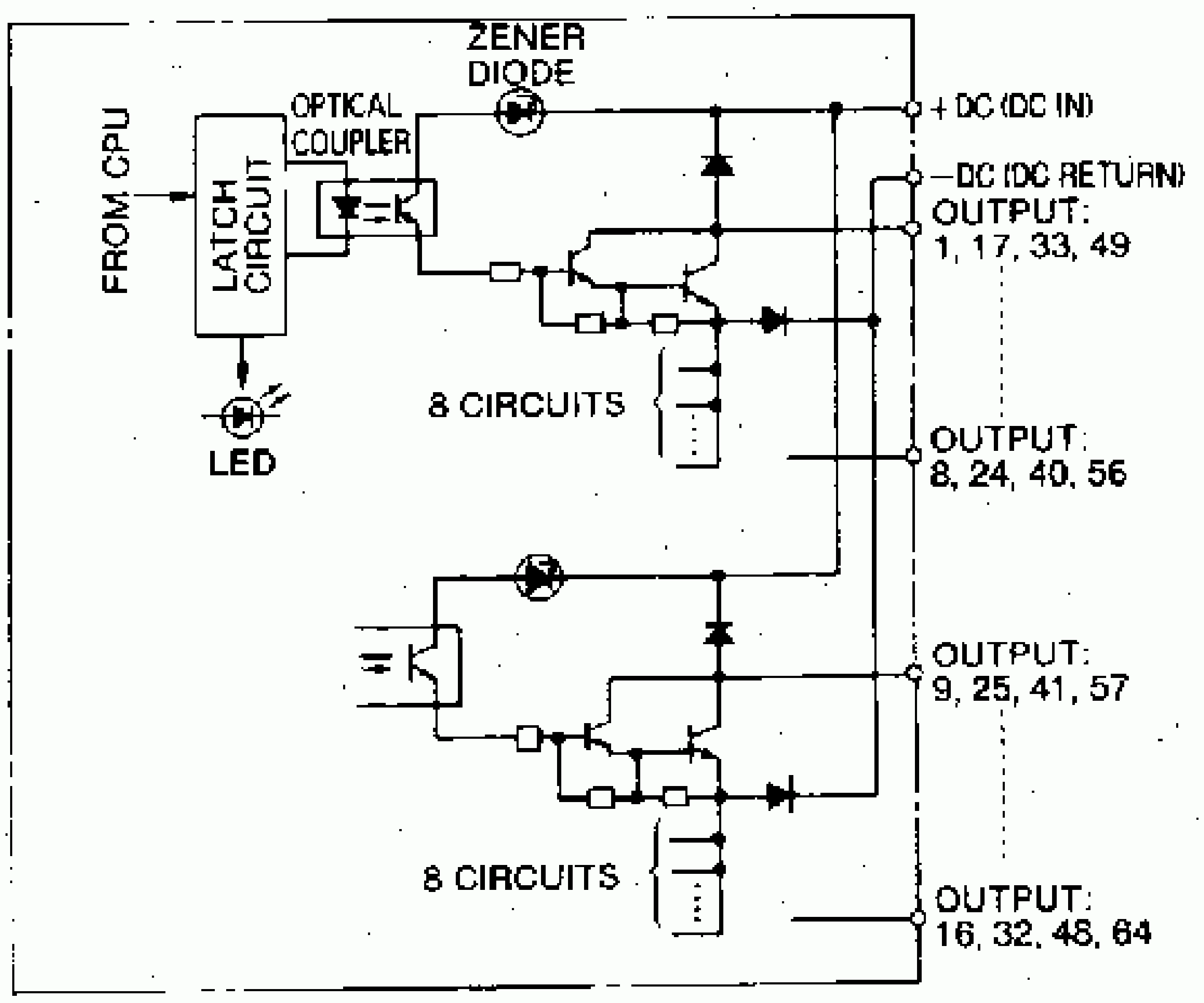


Fig. 5.27 B1060 24 VDC Output Module Simplified Schematic for One Circuit

Note: B1060 is provided with two connectors (connectors 1 and 2): These connectors have same terminal numbers and output connections.

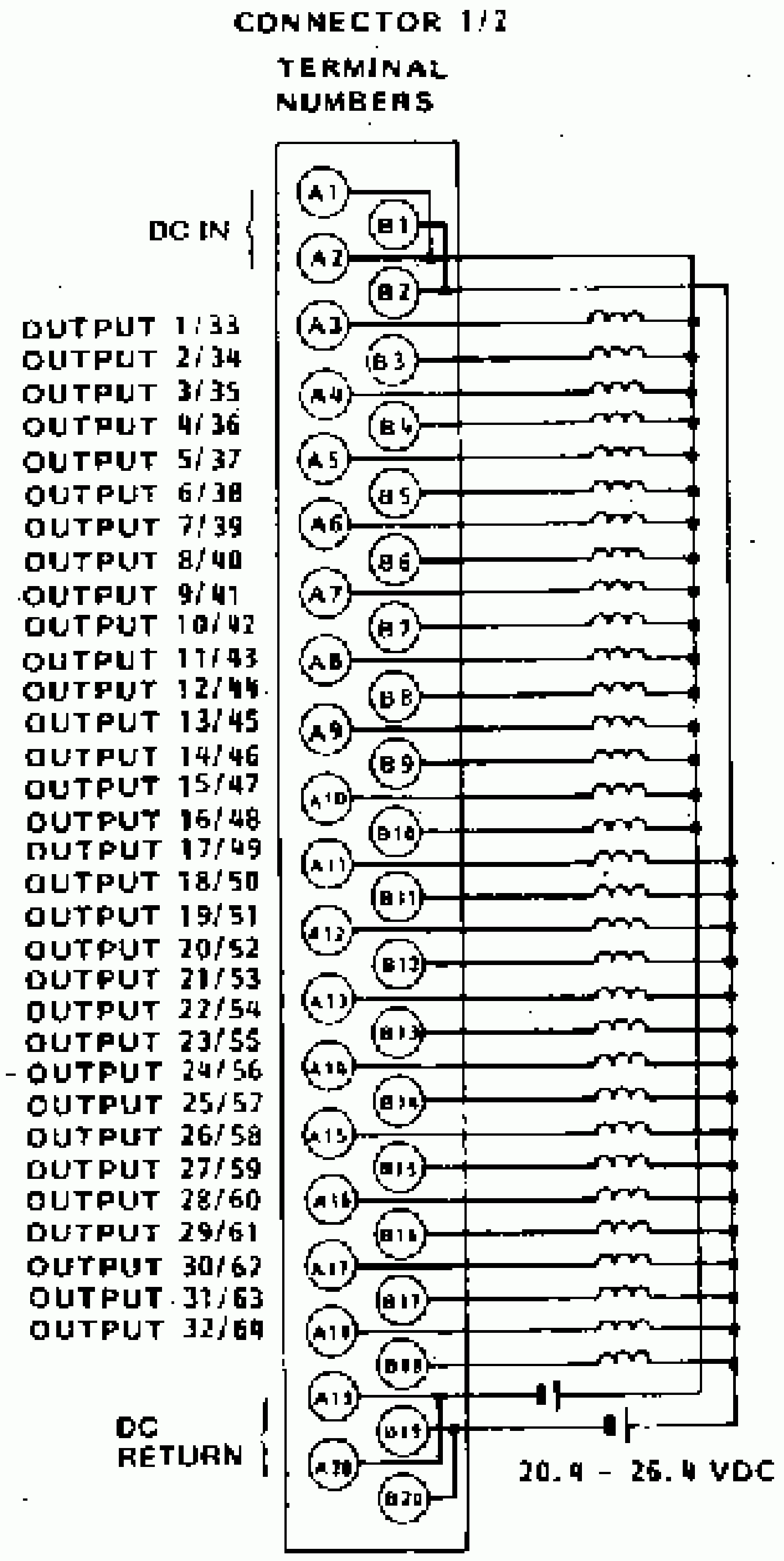


Fig. 5.28 B1060 24 VDC Output Module Terminal Numbering and Output Connections

(15) B1062 24 VDC Output Module

Table 5.15 24 VDC Output Module Specifications

Items			Specifications
Type			JAMSC-B1062
Number of Outputs			32-Outputs per module
Indicators			32-Output status LED's, provided for each output, lighting up when output ON.
Electrical Characteristics	Load Voltage	Working Voltage	20.4 to 26.4 VDC
		Transient Voltage	35 VDC (Peak)
		Average ON Voltage	2.1 V max (Load current: 0.3 A)
	Load Current	ON Current	0.3 A per output; 0.6 A per 4 outputs
		OFF Current	0.2 mA max
		Inrush Current	1 A (10 ms)
	Response Time		OFF to ON: 1 ms max ON to OFF: 1 ms max
	External Power Supply Current (per Module)		External Source Voltage: 24 VDC; Outputs OFF: 1 mA max; Outputs ON: 160 mA max
	Isolation Voltage		1500 VAC for 1 minute

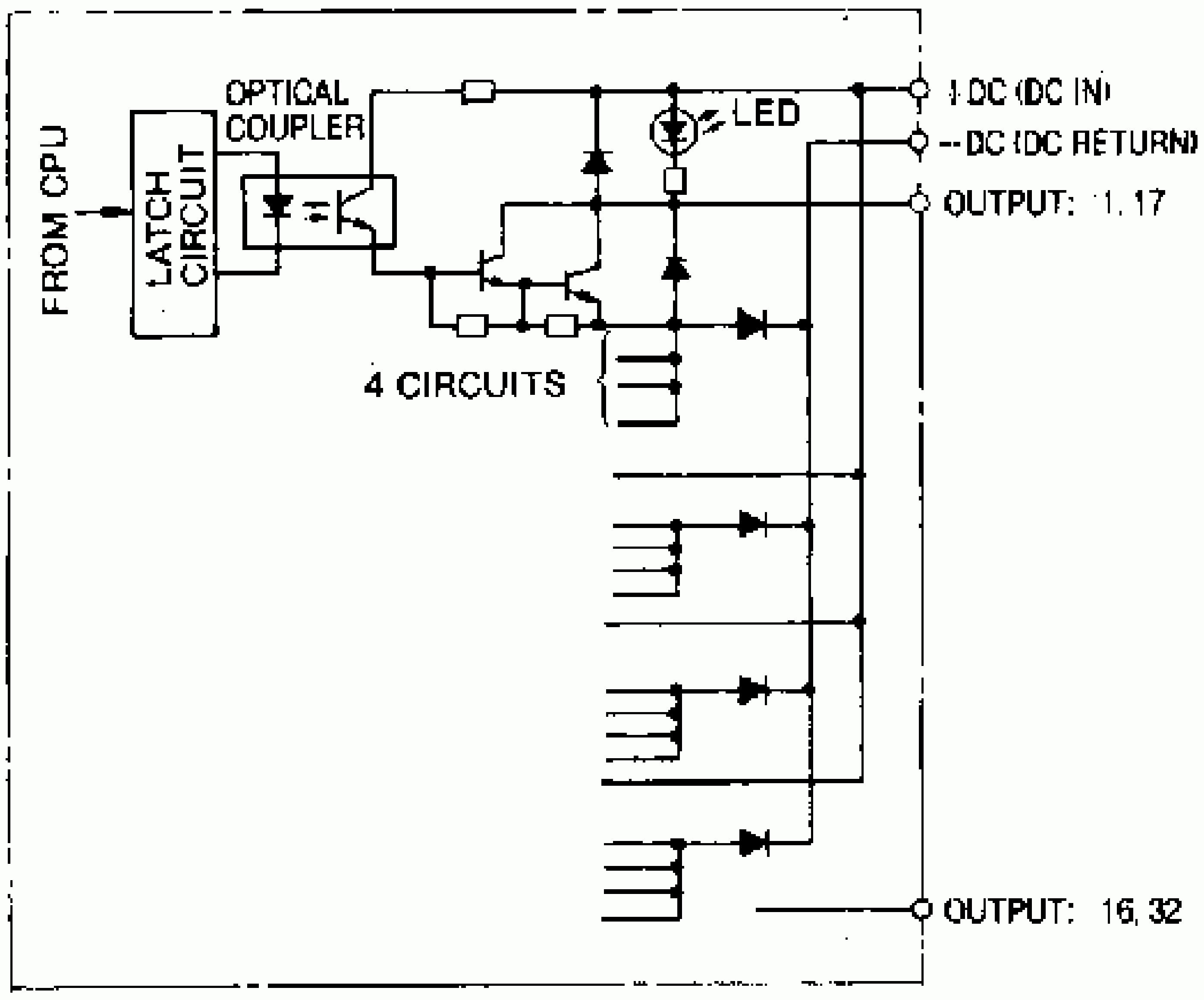


Fig. 5.29 B1062 24 VDC Output Module Simplified Schematic for One Circuit

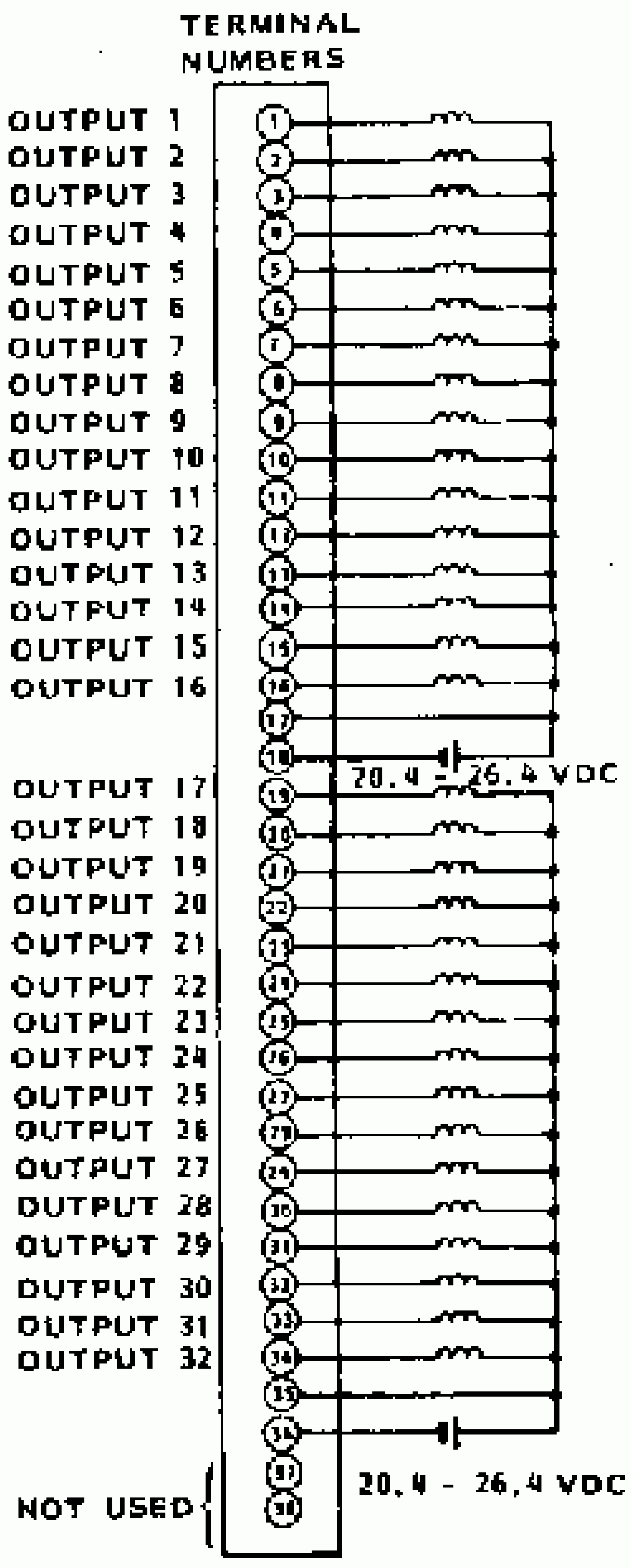


Fig. 5.30 B1062 24 VDC Output Module Terminal Numbering and Output Connections

(16) B1064 24 VDC Output Module

Table 5.16 24 VDC Output Module Specifications

Items			Specifications
Type			JAMSC-B1064
Number of Outputs			32-Outputs per module
Indicators			32-Output status LED's provided for each output, lighting up when output ON.
Fuse Rating			—
Electrical Characteristics	Load Voltage	Working Voltage	19 to 29 VDC
		Transient Voltage	35 VDC (Peak)
		Average ON Voltage	1.5 V (Load current: 0.3 A) (Active low)
	Load Current	ON Current	0.3 A per output, 0.6 A per 4 outputs
		OFF Current	0.2 mA max
		Inrush Current	1 A max for 10 ms
	Response Time		OFF to ON: 1 ms max ON to OFF: 1 ms max
	External Power Supply Current (per Module)		External Source Voltage: 24 VDC; Outputs OFF: 75 mA; Outputs ON: 120 mA
	Isolation Voltage		1500 VAC for 1 minute

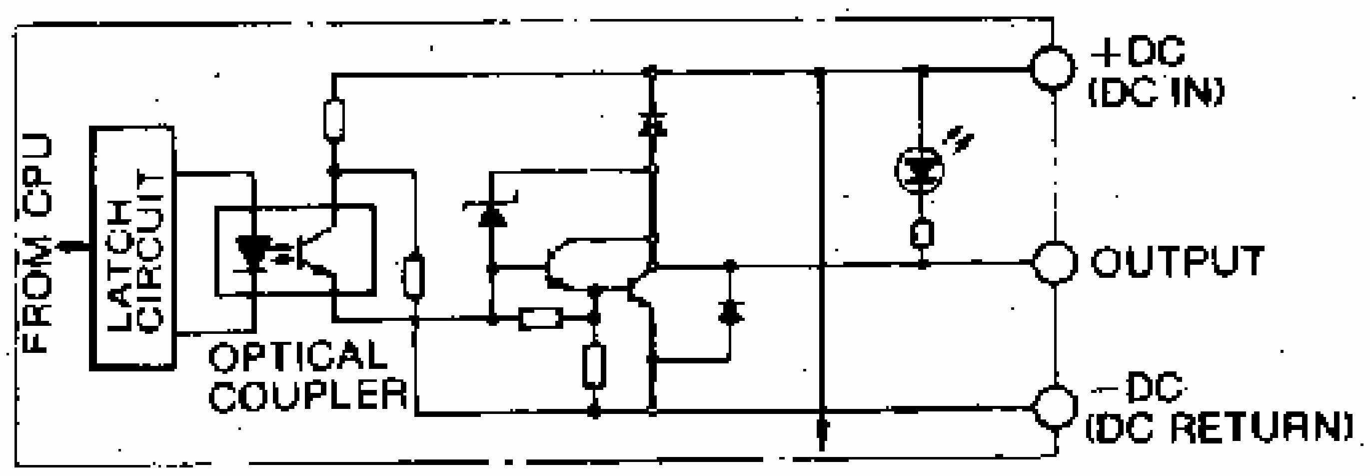


Fig. 5.31 B1064 24 VDC Output Module Simplified Schematic for One Circuit

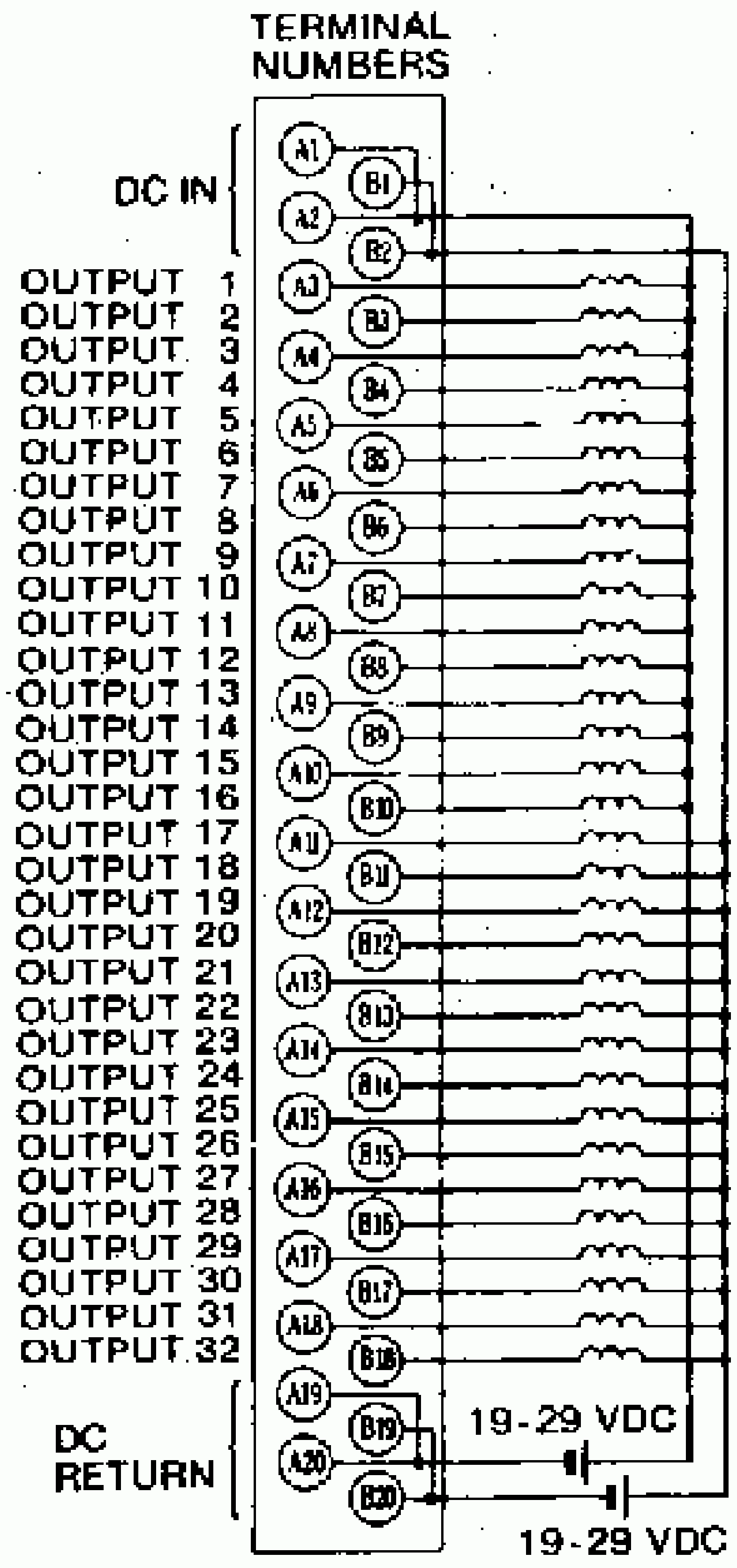


Fig. 5.32 B1064 24 VDC Output Module Terminal Numbering and Output Connections

(17) B1090B Relay Output Module

Table 5.17 Relay Output Module Specifications

Items			Specifications
Type			JAMSC-B1090B
Number of Outputs			16-Outputs per module
Indicators			16-Output status LED's provided for each output, lighting up when output ON.
Fuse Rating			-
Electrical Character-istics	Contact Ratings	Rated Voltage, Current	220 VAC, 0.8 A (Induction load, PF 0.4) 110 VAC, 1.2 A (Induction load, PF 0.4) 24 VDC, 1 A (Induction load, time constant 40 ms).
		Min Operational Voltage, Current	1 V, 10 mA, 100 mW
		Max Operational Voltage	250 VAC/30 VDC
		Max Closing Current	220 VAC, 15 A (PF 0.7)
		Max Interrupting Current	220 VAC, 15 A (PF 0.4) 24 VDC, 2 A (Time constant 40 ms).
		Contact Resistance	100 mΩ max
		Switching Life	100,000 switching min (at rated load)
	Response Time		OFF to ON: 7 ms max ON to OFF: 3 ms max
	External Power Supply for Coil		24 VDC ±10% (riple 5% or less). 450 mA (at all outputs ON).
	Insulation Voltage		1500 VAC for 1 minute

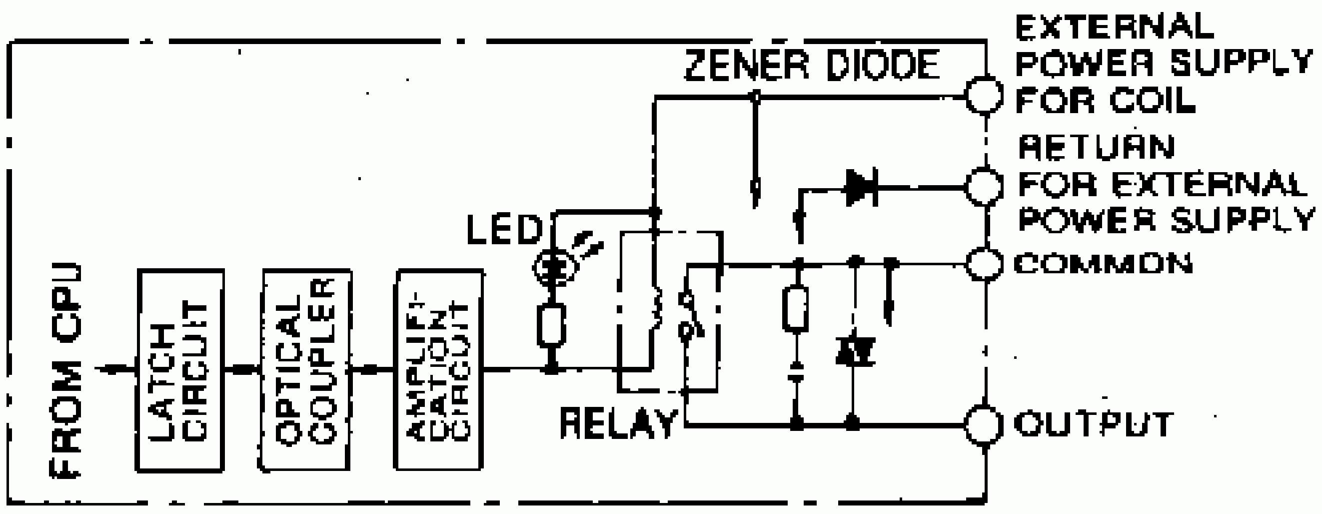


Fig. 5.33 B1090B Relay Output Module Simplified Schematic for One Circuit

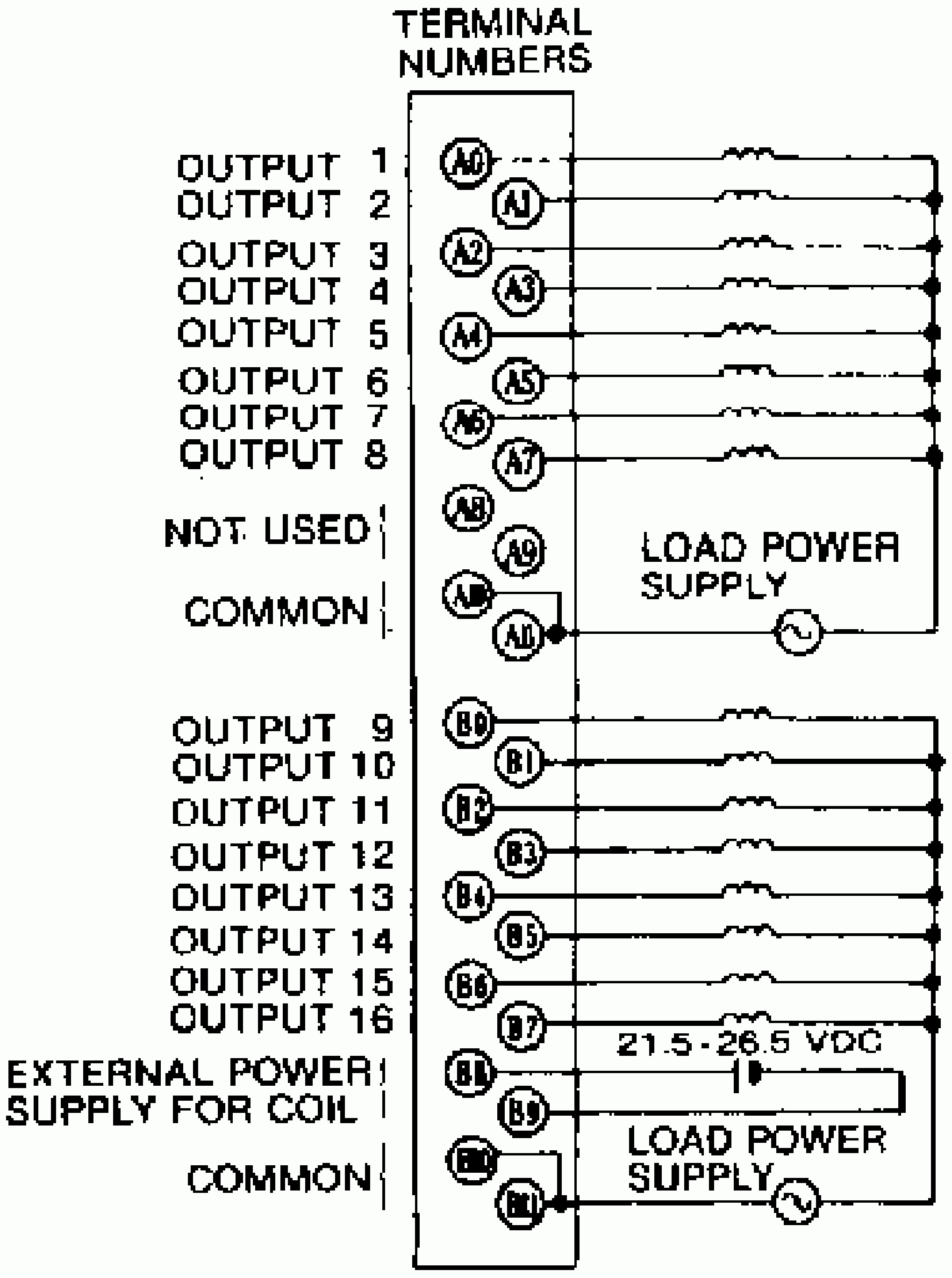


Fig. 5.34 B1090B Relay Output Module Terminal Numbering and Output Connections

(18) B1094 Power Reed Relay Output Module

Table 5.18 Relay Output Module Specifications

Items			Specifications
Type			JAMSC-B1094
Number of Outputs			8-Outputs per module
Indicators			16-Output status LED's provided for each output, lighting up when output ON.
Fuse Rating			1 A (Glass tube fuse) for external power supply.
Electrical Character-istics	Contact Ratings	Rated Voltage, Current	220 VAC, 1 A (PF 0.4) 110 VDC, 0.5 A (Time constant 100 ms)
		Min Operational Voltage, Current	1 V, 1 mA
		Max Operational Voltage	250 VAC/30 VDC
		Max Closing Current	220 VAC, 30 A (PF 0.7)
		Max Interrupting Current	220 VAC, 30 A (PF 0.4) 110 VDC, 0.6 A (Time constant 100 ms).
		Contact Resistance	100 mΩ max
		Switching Life	500,000 switching min (at rated load)
	Response Time		OFF to ON: 7 ms max ON to OFF: 3 ms max
	External Power Supply for Coil		24 VDC $^{+10\%}_{-20\%}$ (riple 5% or less). 270 mA (at all outputs ON).
	Insulation Voltage		1500 VAC for 1 minute

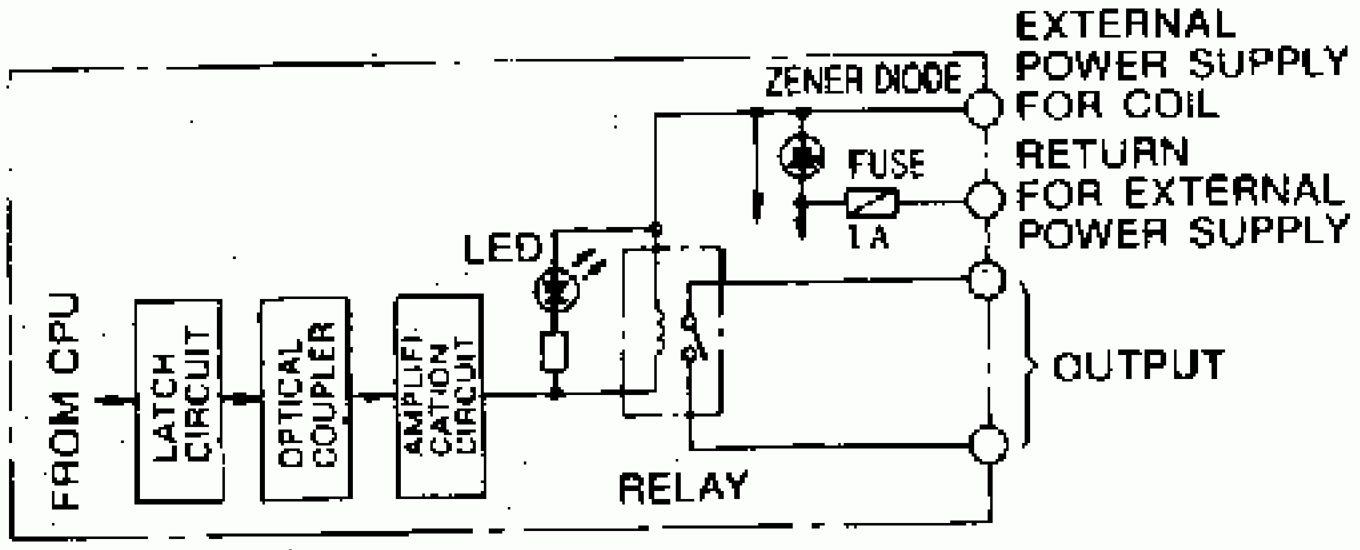


Fig. 5.35 B1094 Power Reed Relay Output Module Simplified Schematic for One Circuit

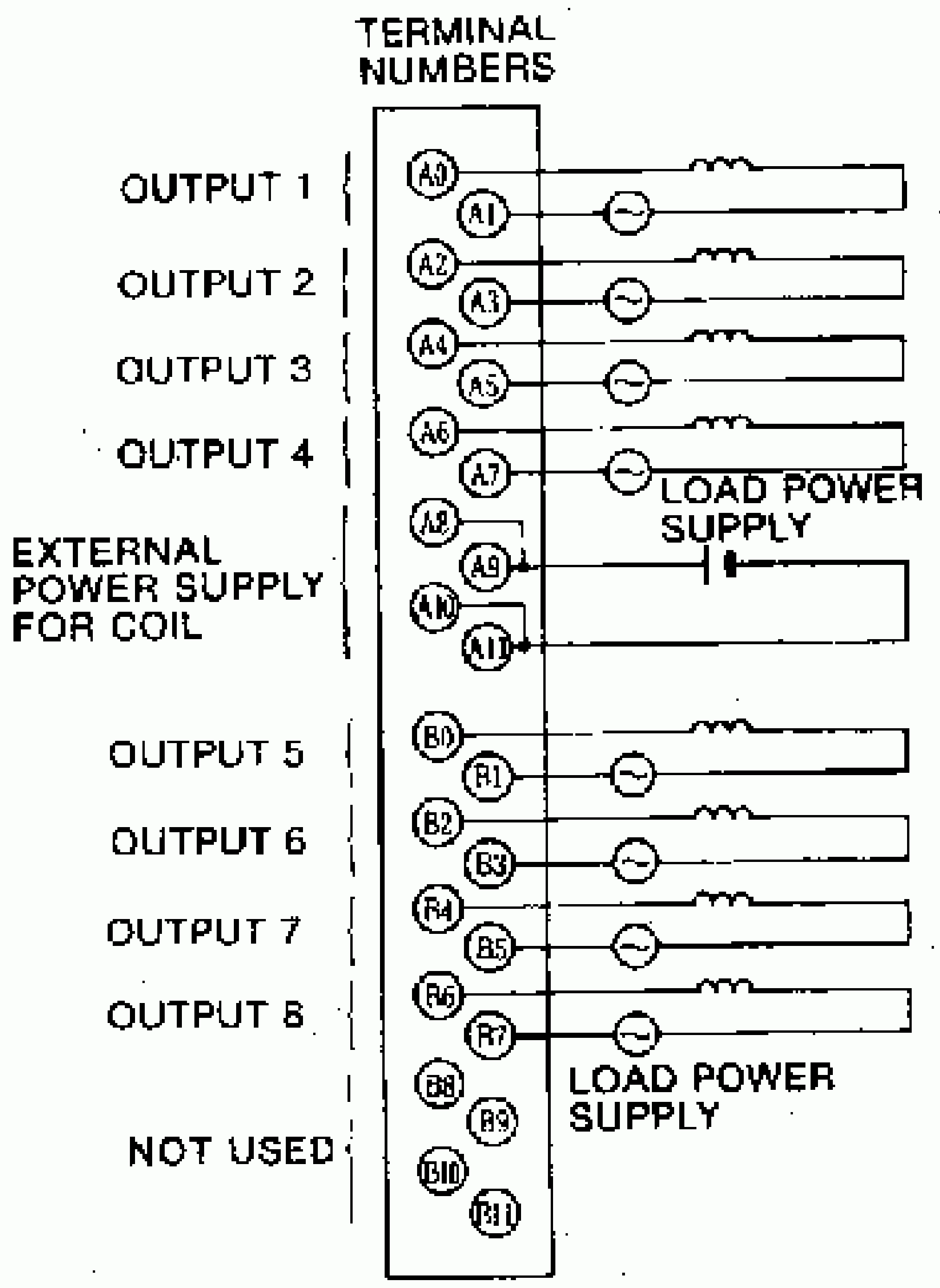


Fig. 5.36 B1094 Power Reed Relay Output Module Terminal Numbering and Output Connections

(19) B1071 Register Input Module

Table 5. 19 Register Input Module Specifications

Items			Specifications
Type			JAMSC-B1071
Number of Inputs			8-inputs per module
Input Numerical Value Range			BCD 4-digit or 16-bit binary
Indicator			BUS Lights (or flickers) while communicating with CPU. FIELD: Lights (or flickers) while reading numerical value data with external power supply turned on.
Fuse			1-fuse for external power supply. 0.25 A (LITTEL FUSE 275250)
Electrical Characteristics	Strobe Output	Logic	ACTIVE "HIGH"
		Strobe Cycle	64 ms
		Output Voltage	+12 V
		Output Current	Source current 50 mA max
	Numerical Value Input	Logic	ACTIVE "HIGH"
		Input Impedance	12 kΩ
		Input "H" Level	+8.5 to +12 V
		Input "L" Level	0 to +3.5 V
	External Power Supply		+24 VDC ±2 V at 0.2 A
	External Wiring Distance		30 meters max
	Isolation Voltage		500 VDC (continuous)

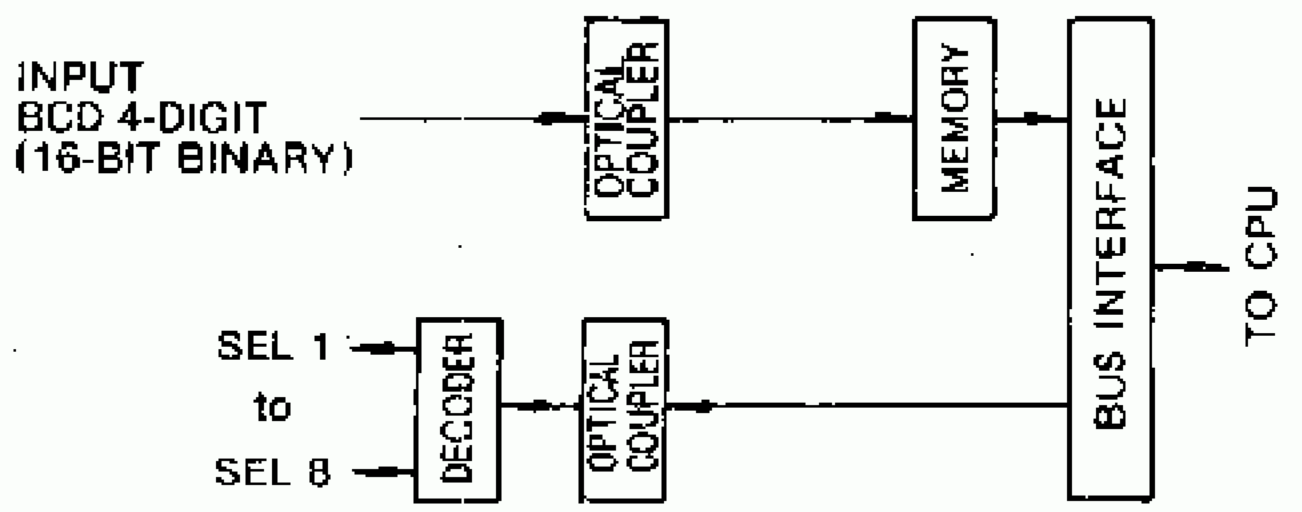
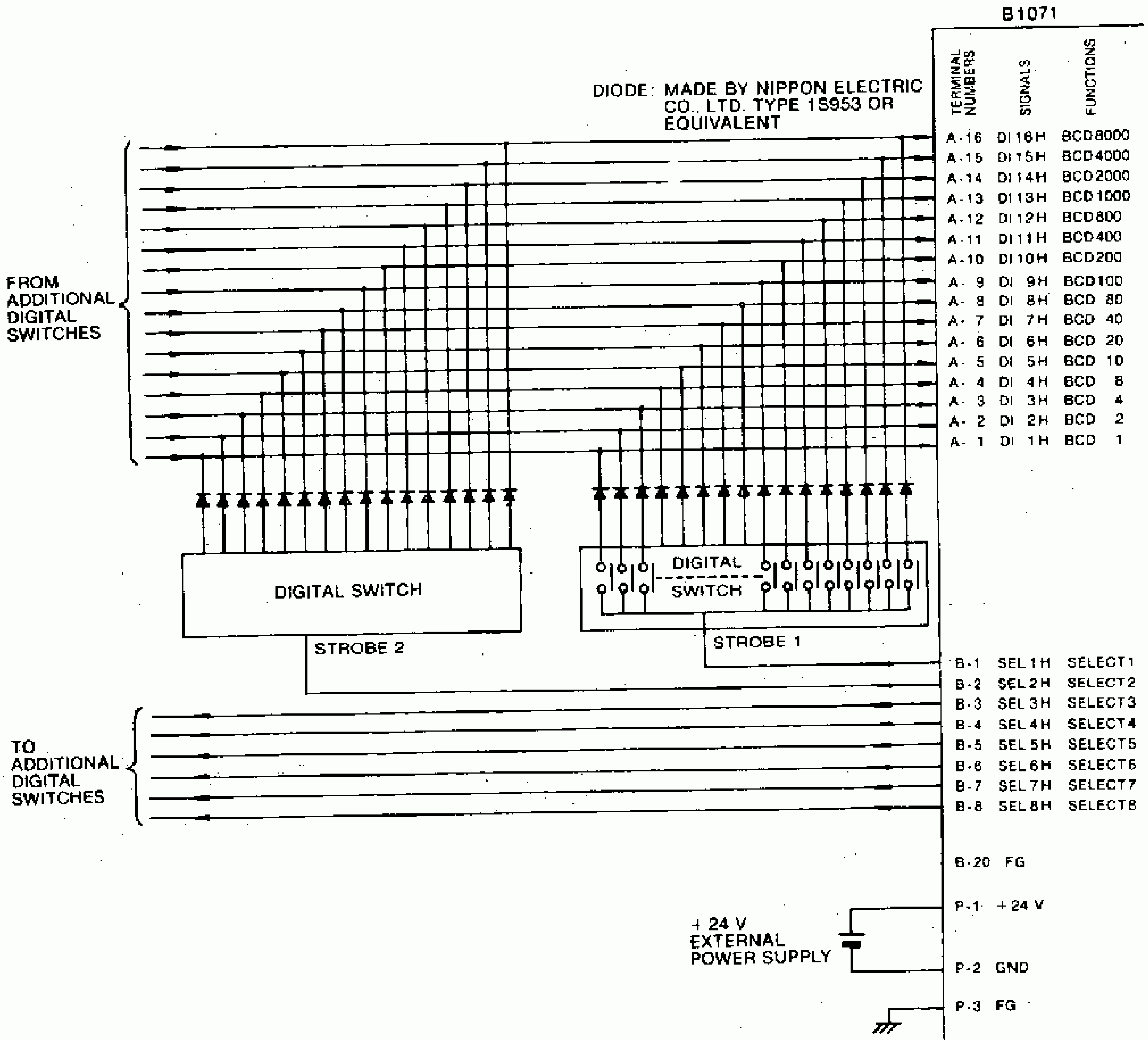


Fig. 5. 37 B1071 Block Diagram

(19) B1071 Register Input Module (Cont'd)



Note:

1. Connect only to the terminals shown above.
2. When the input data is binary, the terminal A-1 becomes the digit of 2^0 and the terminal A-16 becomes the digit of 2^{15}
3. The GNDs of the input data and the strobe output are connected to the GND (terminal P-2) of the external power supply in the module.

Fig. 5. 38 B1071 Connection Diagram

(20) B1070 Register Output module

Table 5. 20 Register Output Module Specifications

Items			Specifications
Type			JAMSC-B1070
Number of Outputs			8-Outputs per module
Output Numerical Value Range			BCD 4-digit or 16-bit binary
Indicator			BUS: Lights (or flickers) while communicating with CPU. FIELD: Lights (or flickers) while sending numerical value data with external power supply turned on.
Fuse			1-fuse for external power supply, 0.25 A (LITTEL FUSE 275250)
Electrical Character-istics	Strobe Output	Logic	ACTIVE "LOW"
		Strobe Cycle	64 ms
		Output Voltage	Open collector output, collector voltage 30 V max
		Output Current	Sink current 50 mA max
	Numerical Value Output	Logic	ACTIVE "LOW"
		Output Voltage	Open collector output, collector voltage 30 V max
		Output Current	Sink current 50 mA max
	External Power Supply		+24 VDC ±2 V at 0.2 A
	External Wiring Distance		30 meters max (depends on the equipment to be connected).
	Isolation Voltage		500 VDC (continuous)

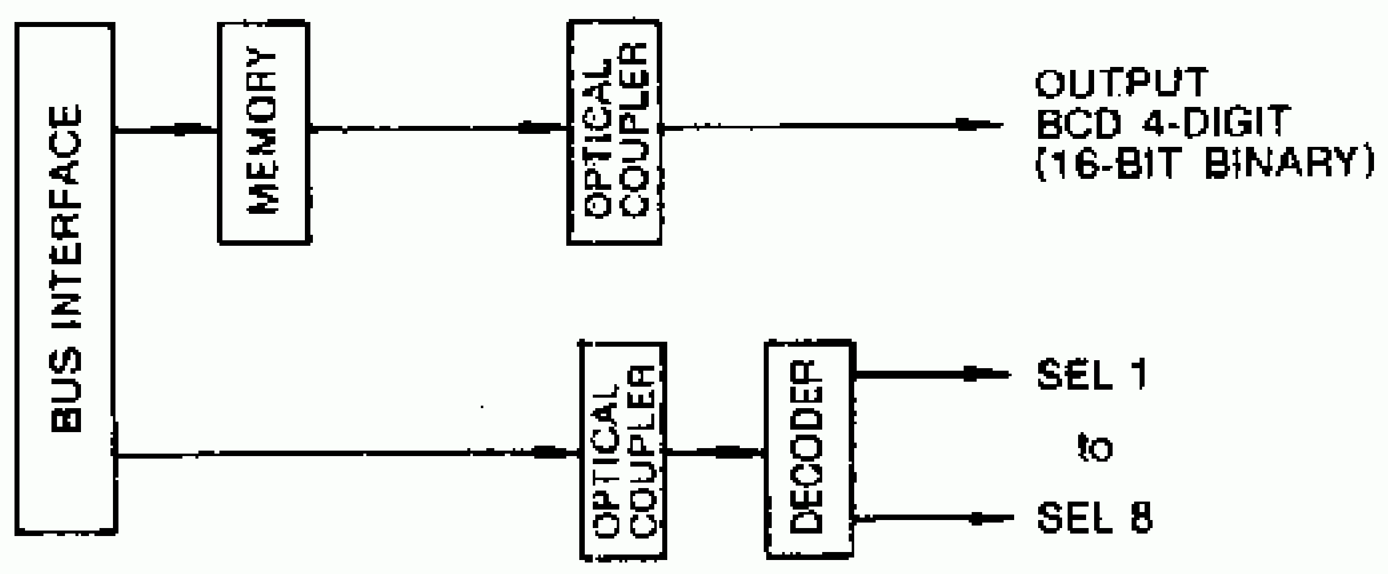
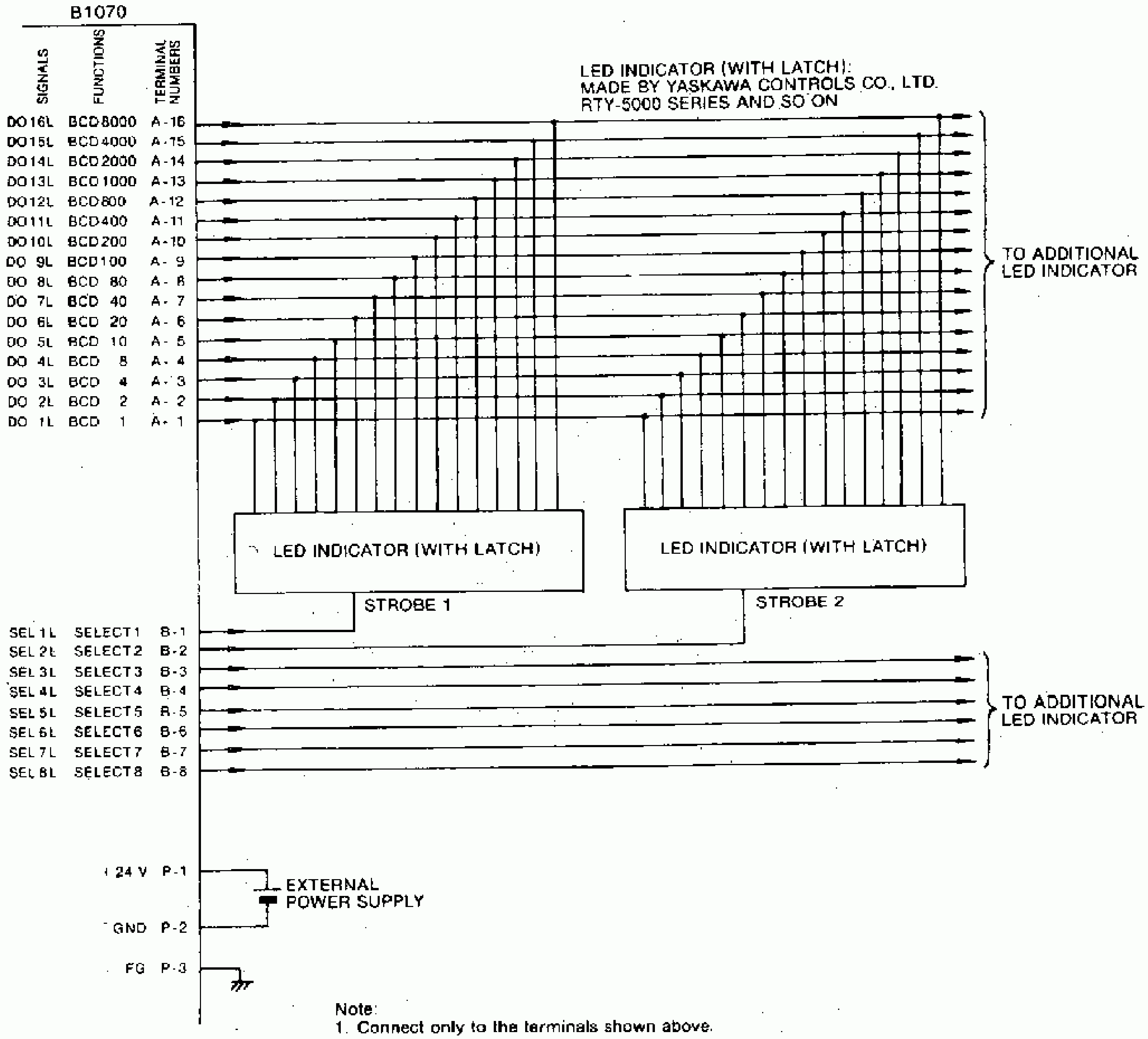


Fig. 5. 39 B1070 Block Diagram

(20) B1070 Register Output module (Cont'd)



Note:
1. Connect only to the terminals shown above.
2. When the output data is binary, the terminal A-1 becomes the digit of 2^0 and the terminal A-16 becomes the digit of 2^{15}.
3. The GNDs of the output data and the strobe output are connected to the GND (terminal P-2) of the external power supply in the module.

Fig. 5.40 B1070 Connection Diagram

(21) B1073 Analog Input (A/D Conversion) Module

Table 5.21 Analog Input Module Specifications

Items		Specifications
Type		JAMSC-B1073-1, -2
Number of Inputs		4-Inputs per module
Indicator		BUS: Lights (or flickers) while communicating with CPU. POWER: Lights when external power supply is turned on.
Fuse		2-fuses for external power supply, 0.25 A (LITTEL FUSE 275250)
Electrical Characteristics	Input Range	JAMSC-B1073-1: 0 to +10 V JAMSC-B1073-2 (optional) : +1 to +5 V
	Output	0-1023 (10 bits), stored into input register
	Input Impedance	2 MΩ (normal input)
	Resolution	1 bit in 1024
	Linearity Error	Less than 0.1 % of full scale
	Temperature Coefficient	0 to +10V range: Gain—40ppm/°C(full scale) Offset—10ppm/°C (full scale) +1 to +5V range: Gain—60ppm/°C (full scale) Offset—20ppm/°C (full scale)
	Accuracy	Less than 0.2 %, 25°C
	Max Allowable Input	±15 V
	Cross Talk between Input	−70 dB or above
	External Power Supply	+15 VDC ±0.5 V, 120 mA −15 VDC ±0.5 V, 40 mA
	Isolation Voltage	500 VDC (continuous)

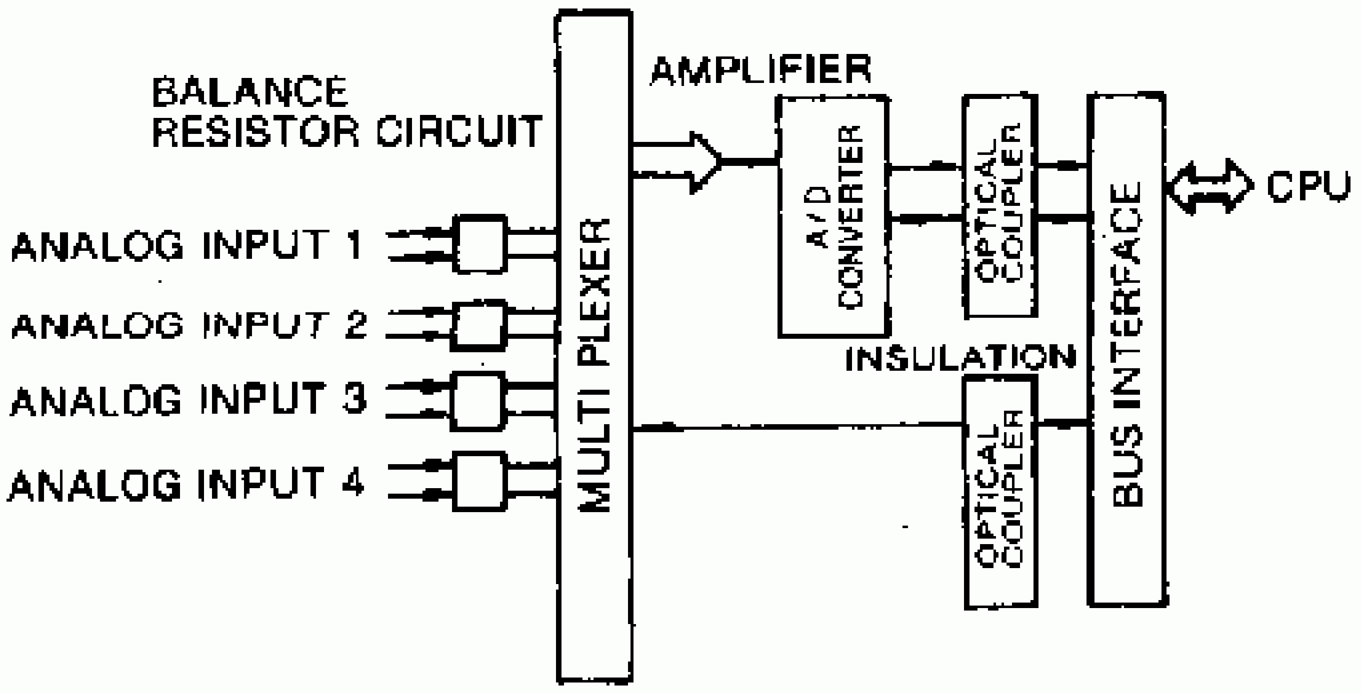


Fig. 5.41 B1073 Block Diagram

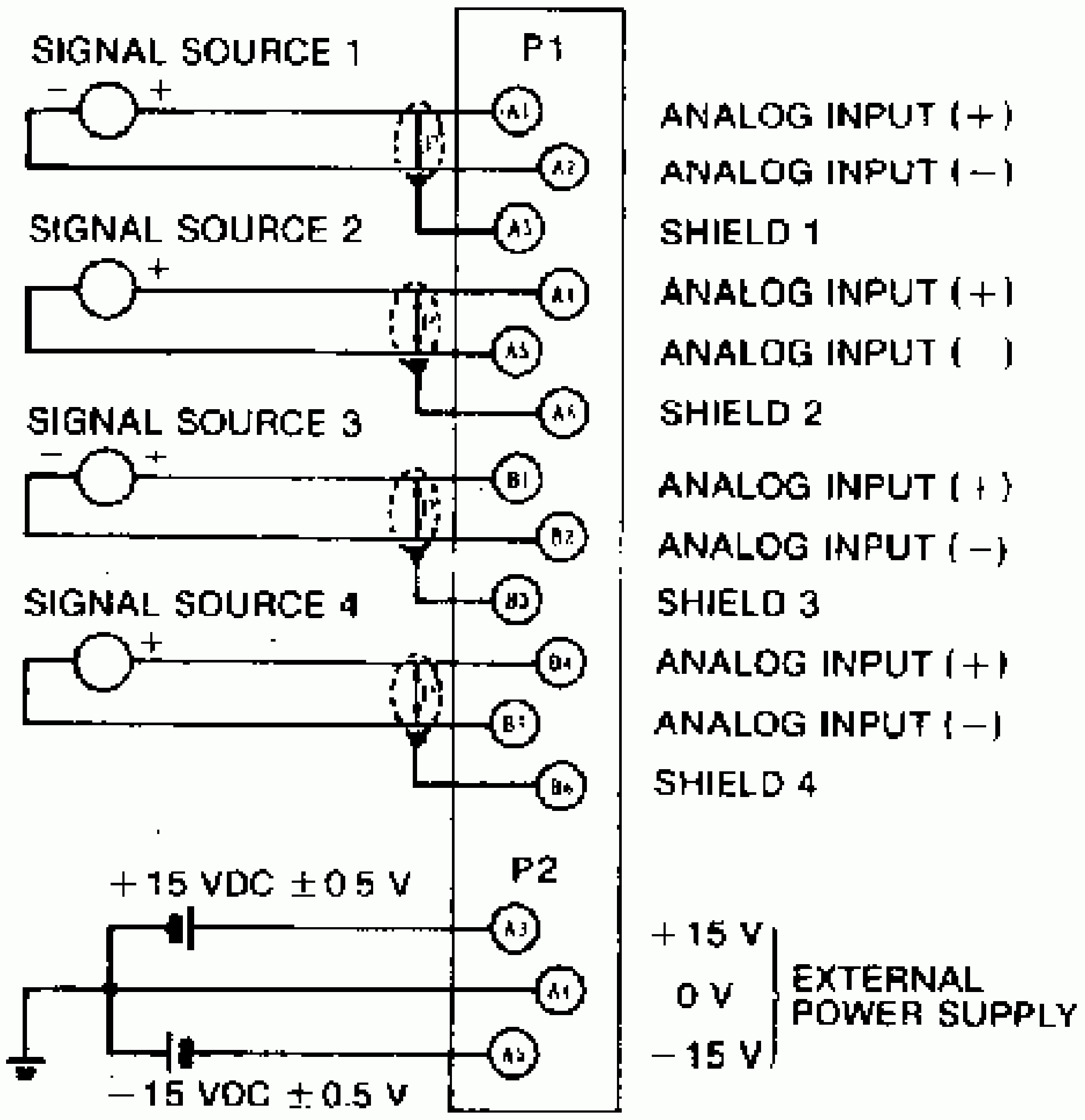


Fig. 5.42 B1073 Connection Diagram

Note: Terminals (A1), (A2), (A6) and (B1) to (B6) of connector P1 are not used.

(22) B1072B Analog Output (D/A Conversion) Module

Table 5.22 Analog Output Module Specifications

Items		Specifications
Type		JAMSC-B1072B-1, 2, -3, -4,
Number of Outputs		2-outputs per module
Indicator		No. 1 ACTIVE, No. 2 ACTIVE: Lights (or blinks) while communicating with CPU with external power supply turned on.
Fuse		2-fuses for external power supply, 0.25 A (LITTEL FUSE 275250)
Electrical Characteristics	Input	0-1024 (10 bits), contents of output register
	Output Range	JAMSC-B1072B-1: 0 to +10 V JAMSC-B1072B-2(optional): 0 to +5 V JAMSC-B1072B-3(optional): −5 to +5 V JAMSC-B1072B-4(optional): −10 to +10V
	Output Impedance	1 Ω or less
	Max Output Current	10 mA
	Resolution	1 bit in 1024
	Linearity Error	Less than 0.1 % of full scale
	Temperature Coefficient	0 to +10V, 0 to +5V range: Gain −40ppm/°C (full scale) Offset − 10ppm/°C (full scale) −5V to +5V, −10 to +10V range: Gain −40ppm/°C (full scale) Offset − 20ppm/°C (full scale)
	Accuracy	Less than 0.2 %, 25℃
	Cross Talk between Outputs	−80 dB or above
	External Power Supply	+15 VDC ±0.5 V, 60 mA per circuit −15 VDC ±0.5 V, 50 mA per circuit
	Isolation Voltage	500 VDC (continuous)

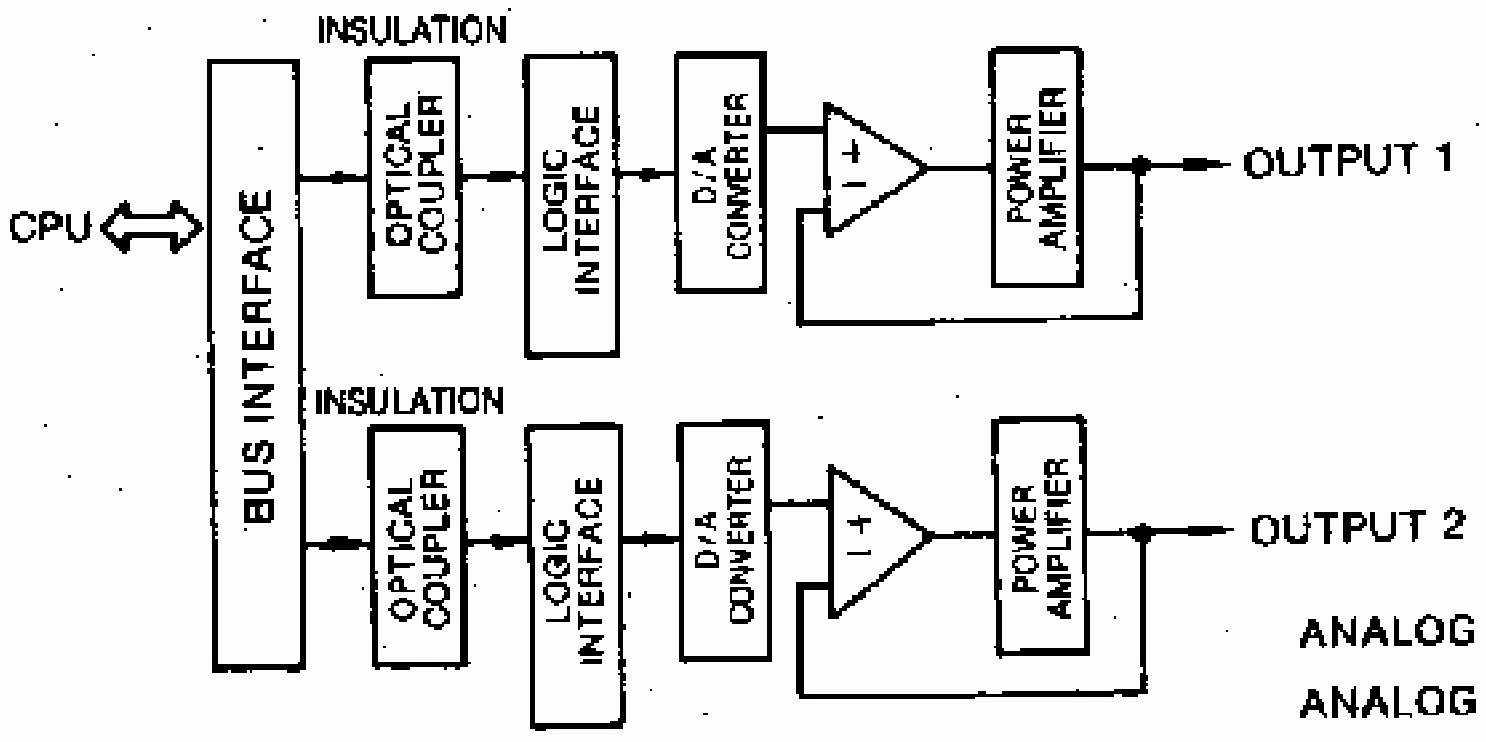


Fig. 5.43 B1072B Block Diagram

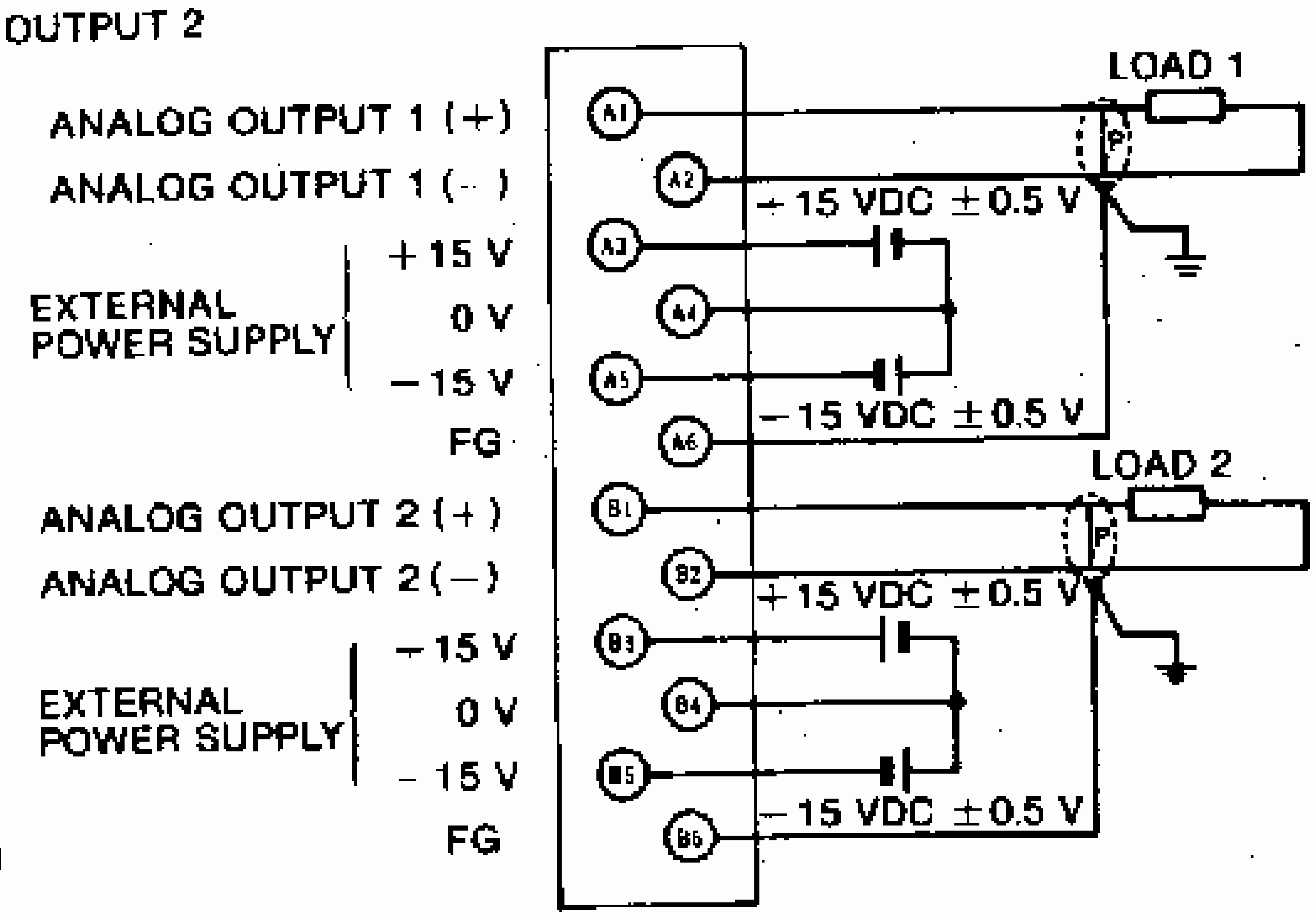


Fig. 5.44 B1072B Connection Diagram

Note: The shield of the cable should be connected to either the load side or the module side.

(23) B1075 Analog Input (A/D Conversion) Module

Table 5.23 Analog Input Module Specifications

Items		Specifications
Type		JAMSC-B1075-1, -2
Number of Inputs		4-Inputs per module
Indicator		BUS: Lights (or flickers) while communicating with CPU. POWER: Lights when external power supply is turned on.
Fuse		2-fuses for external power supply, 0.25 A (LITTEL FUSE 275250)
Electrical Characteristics	Input Range	JAMSC-B1075-1: 0 to +10 V JAMSC-B1075-2 (optional) : +1 to +5 V
	Output	0-4095 (12 bits), stored into input register
	Input Impedance	2 MΩ (normal input)
	Resolution	1 bit in 4096
	Linearity Error	Less than 0.1 % of full scale
	Temperature Coefficient	0 to +10V range: Gain—40ppm/°C (full scale) Offset—10ppm/°C (full scale) +1 to +5V range: Gain—60ppm/°C (full scale) Offset—20ppm/°C (full scale)
	Accuracy	Less than 0.2 %, 25°C
	Max Allowable Input	±15 V
	Cross Talk between Input	−70 dB or above
	External Power Supply	+15 VDC ±0.5 V, 120 mA −15 VDC ±0.5 V, 40 mA
	Isolation Voltage	500 VDC (continuous)

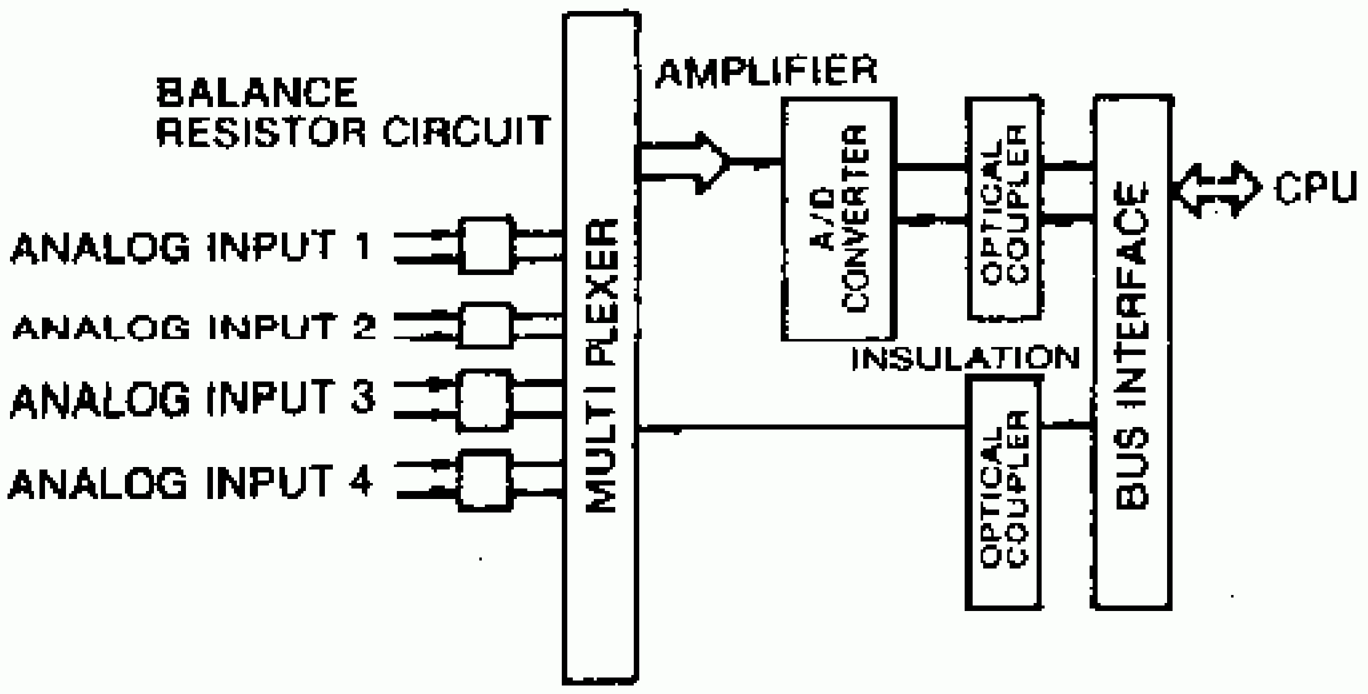


Fig. 5.45 B1075 Block Diagram

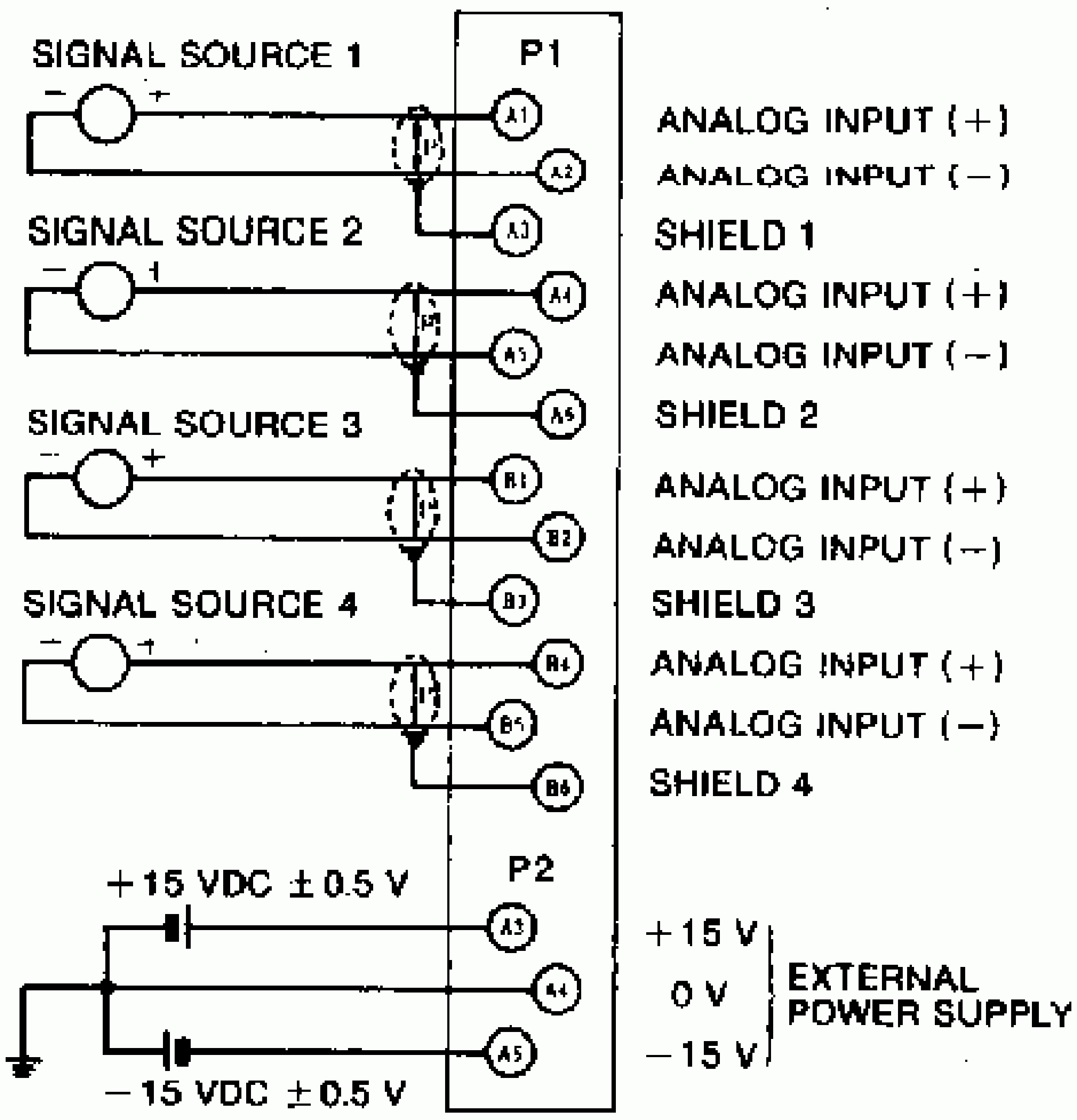


Fig. 5.46 B1075 Connection Diagram

Note: Terminals (A1), (A2), (A6) and (B1) to (B6) of connector P2 are not used.

(24) B1074 Analog Output (D/A Conversion) Module

Table 5. 24 Analog Output Module Specifications

Items		Specifications
Type		JAMSC-B1074-1, -2, -3, -4
Number of Outputs		2-outputs per module
Indicator		No. 1 ACTIVE, No. 2 ACTIVE: Lights (or flickers) while communicating with CPU with external power supply turned on.
Fuse		2-fuses for external power supply, 0.25 A (LITTEL FUSE 275250)
Electrical Characteristics	Input	0-4095 (12 bits), contents of output register
	Output Range	JAMSC-B1074-1: 0 to +10 V JAMSC-B1074-2 (optional): 0 to +5 V JAMSC-B1074-3 (optional): −5 to +5 V JAMSC-B1074-4 (optional): −10 to +10 V
	Output Impedance	1 Ω or less
	Max Output Current	10 mA
	Resolution	1 bit in 4096
	Linearity Error	Less than 0.1 % of full scale
	Temperature Coefficient	0 to +10V, 0 to +5V range. Gain−40ppm/°C (full scale) Offset−10ppm/°C (full scale) −5 to +5V, 10V to +10V range: Gain−40ppm/°C (full scale) Offset−20ppm/°C (full scale)
	Accuracy	Less than 0.2 %, 25℃
	Cross Talk between Outputs	−80 dB or above
	External Power Supply	+15 VDC ±0.5 V, 60 mA per circuit −15 VDC ±0.5 V, 50 mA per circuit
	Isolation Voltage	500VDC (continuous)

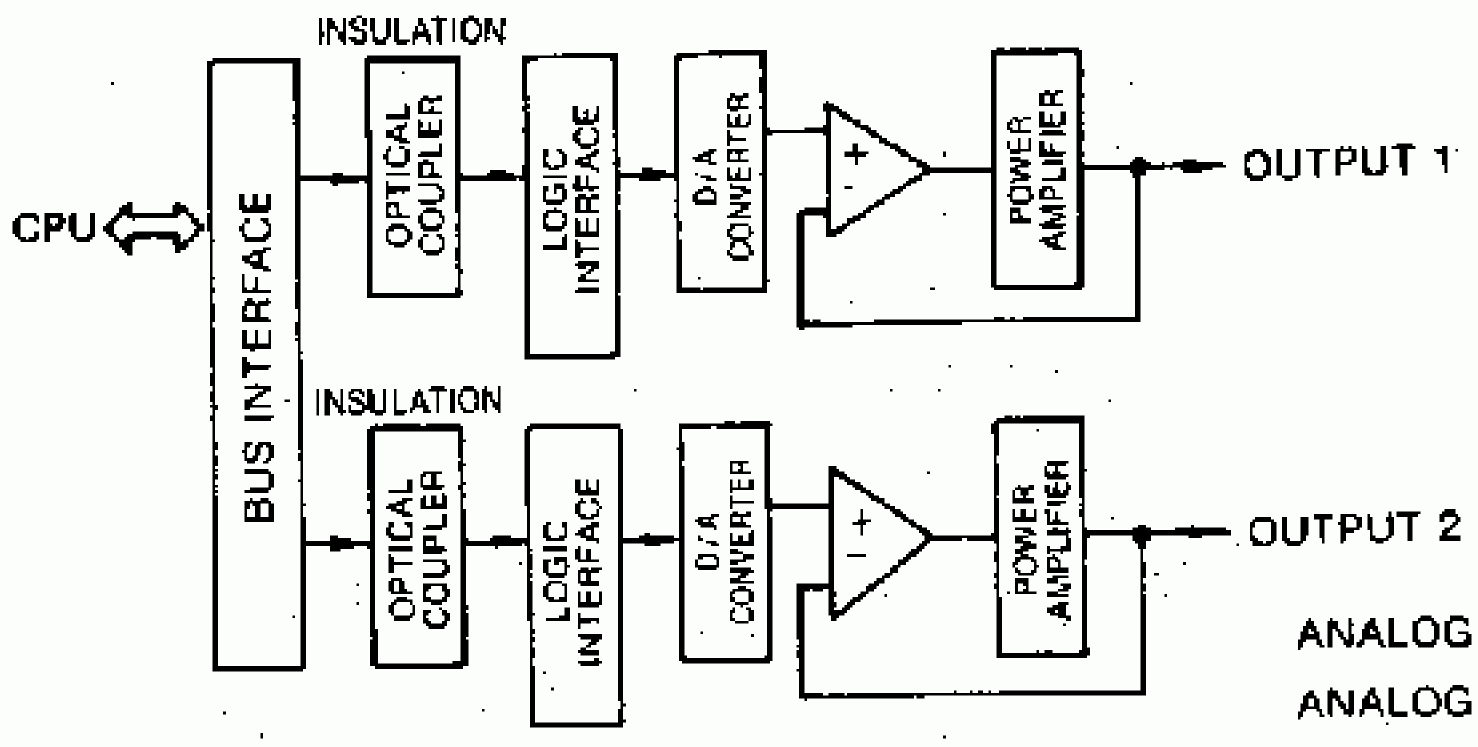


Fig. 5. 47 B1074 Block Diagram

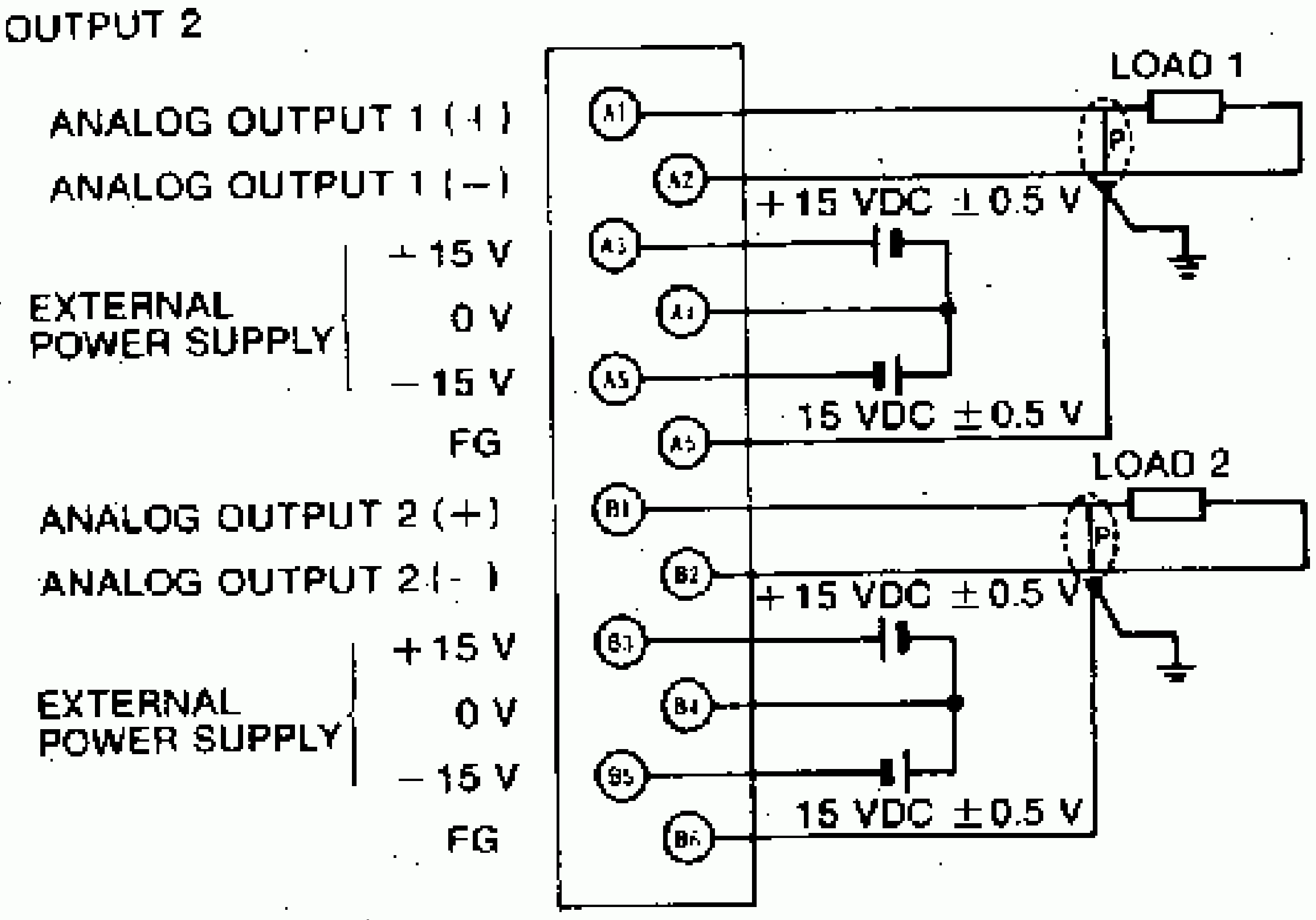


Fig. 5. 48 B1074 Connection Diagram

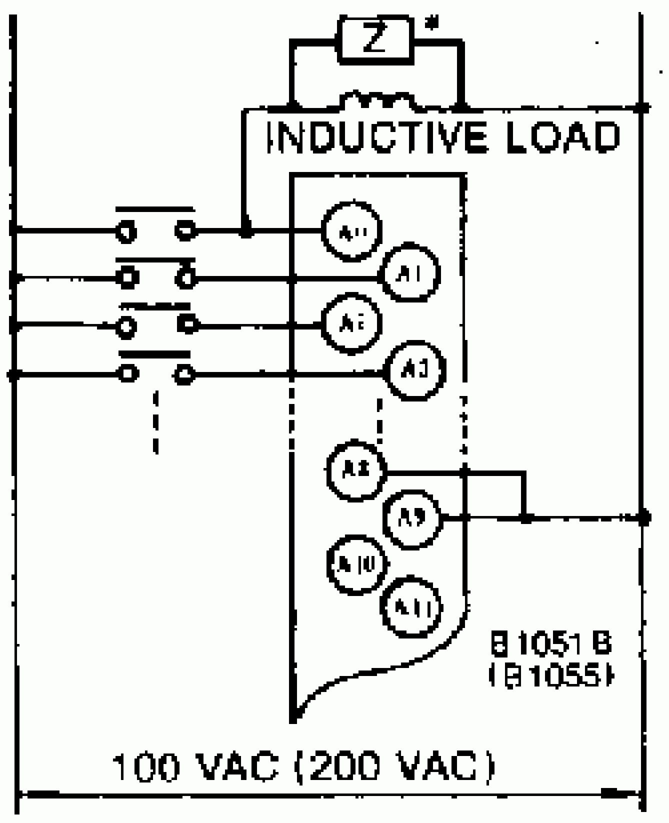
Note: The shield of the cable should be grounded at either the load side or the module side.

5. 2 PRECAUTIONS FOR USING INPUT/OUTPUT MODULES

5. 2. 1 Input Module

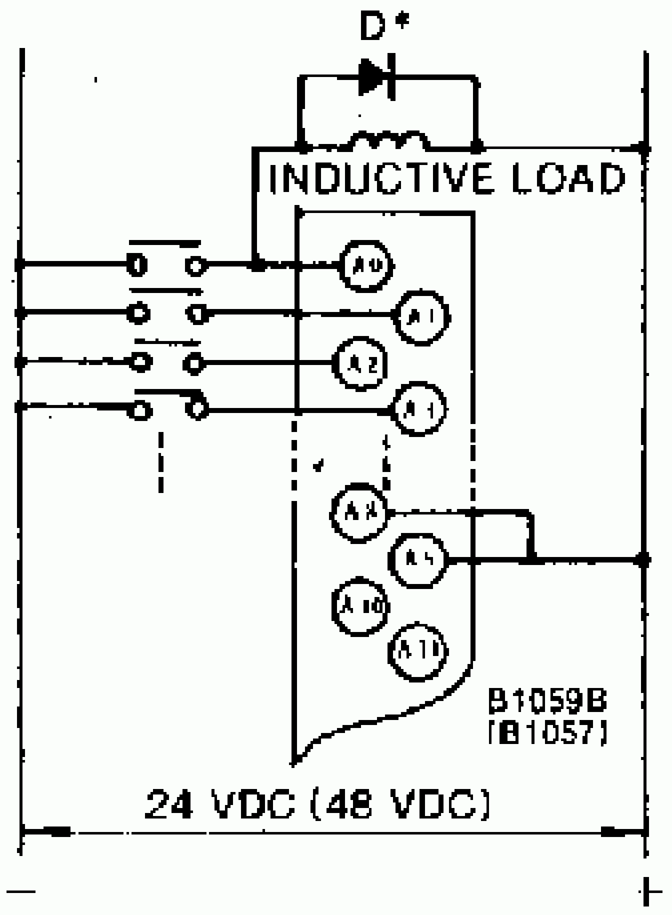
(1) Inductive Load

Where an inductive load is connected in parallel with the input module as shown in Figs. 5.49 and 5.50, connect a surge absorber or flywheel diode in parallel with the inductive load, respectively for the AC input module and DC input module. However, when the current flow through the inductive load connected to the DC input module is below 0.5 A, no flywheel diode is required to be connected.



*The surge absorber capacity should be selected corresponding to the load. It is recommended that type CR 50500 (made by Okaya Electric Industries Co.) or equivalent be used.

Fig. 5. 49 AC Input Module

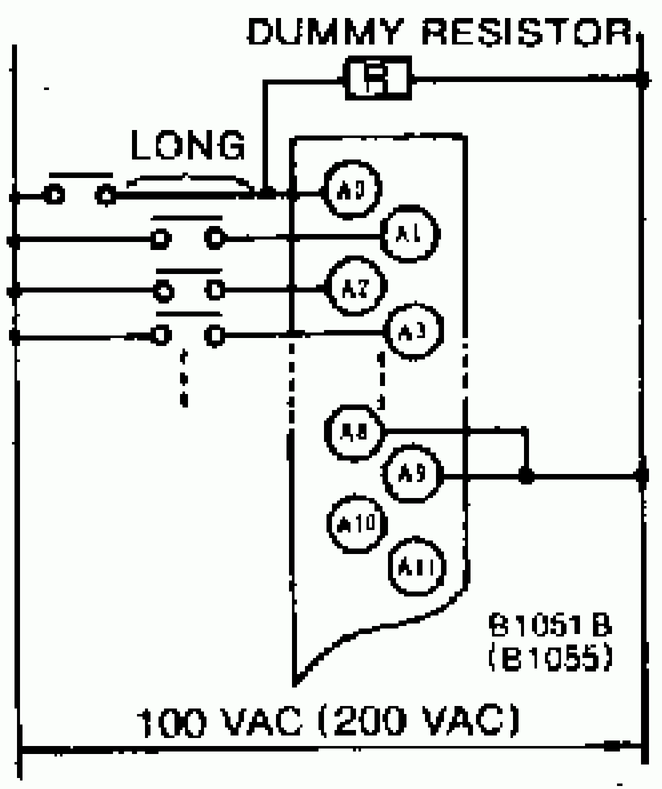


*The flywheel diode should be selected corresponding to the load. It is recommended that type F14 series (made by NEC) or equivalent be used.

Fig. 5. 50 DC Input Module

(2) Input Dummy Resistor

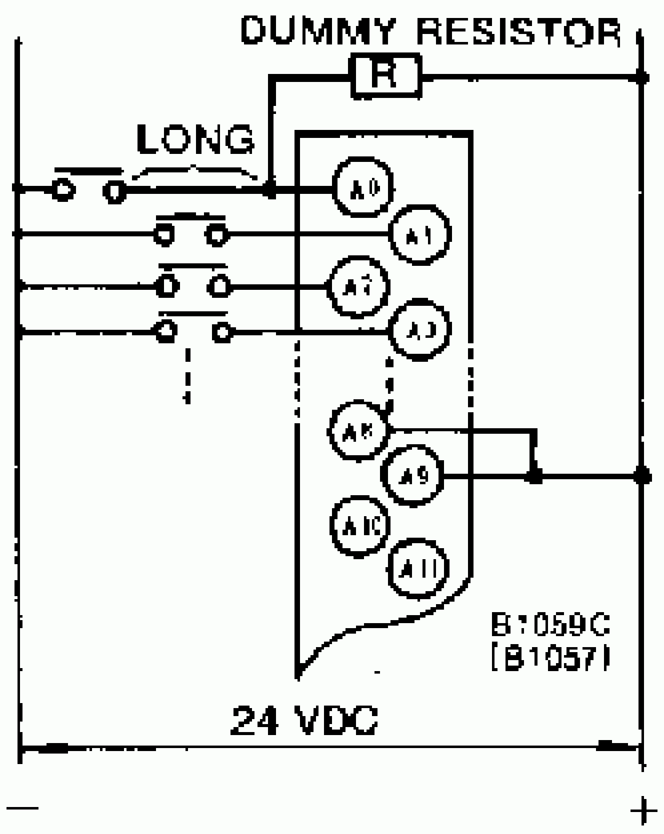
Where the external wiring is long or where there is an induction source in the vicinity, connect a dummy resistor in parallel to the input module, as shown in Figs. 5.51 and 5.52.



R: Input dummy resistor

- B1051B: 5 kΩ (10 W min)
- B1055: 10 kΩ (20 W min)

Fig. 5. 51 AC Input Module



R: Input dummy resistor

- B1059C: 2 kΩ (2 W min)
- B1057: 2 kΩ (2 W min)

Fig. 5. 52 DC Input Module

(3) Leakage Current in Input Equipment

Some input equipment (e.g. noncontact switches and limit switches with LED) has leakage current during the OFF state. If this equipment is connected to AC input modules, it may fail to maintain the voltage for an OFF condition which is an input due to leakage current, and input signals may not be cut off.

(Example) A non-contact switch with 5 mA of leakage current is connected to B1051B.

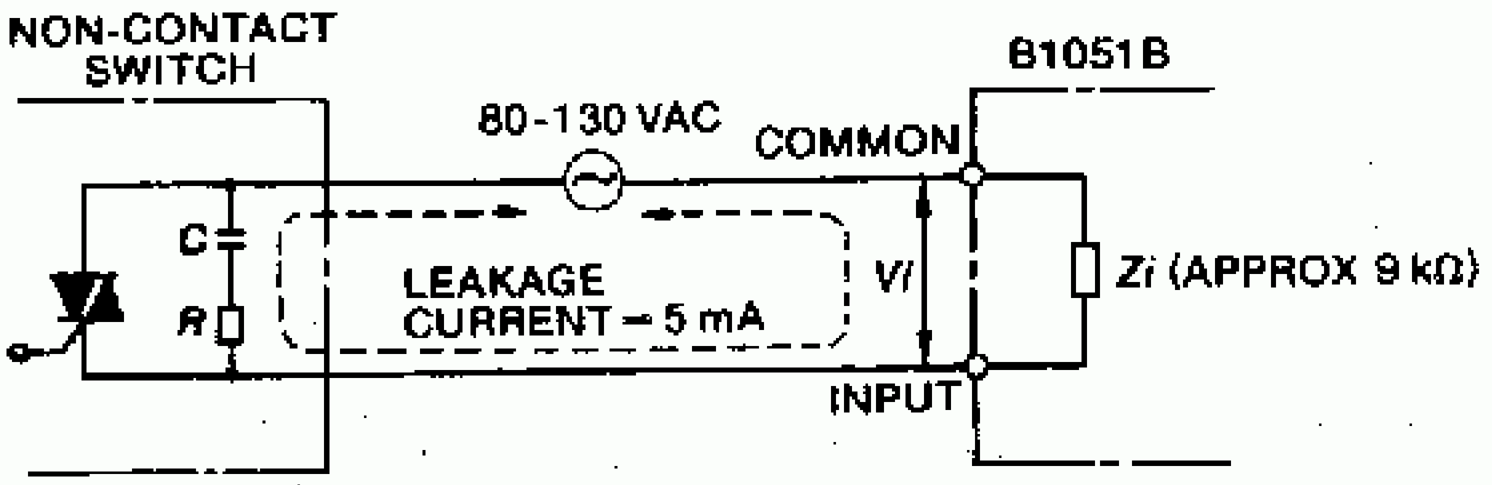


Fig. 5.53 Connection of a Non-contact Switch

If the leakage current is 5 mA, the input voltage (Vi) of B1051B becomes;

$$Vi = 5\ mA \times Zi = 5\ mA \times 9\ k\Omega = 45\ V$$

Since this does not satisfy an input condition (OFF voltage = 30 V or less), input signals may not be cut off. In this case, add a proper dummy resistor to the input terminal of B1051B.

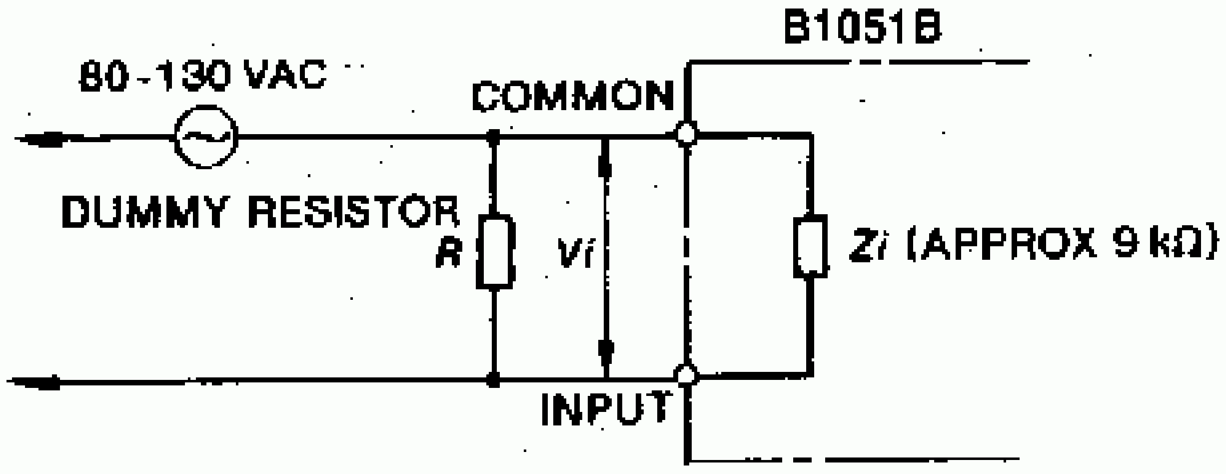


Fig. 5.54. Addition of a Dummy Resistor

The value of the dummy resistor R should be decided so that an input voltage Vi of B1051B becomes 30 V or less.

$$\frac{R \times Zi}{R + Zi} \times leakage\ current < 30\ V$$

$$\frac{R \times 9\ k\Omega}{R + 9\ k\Omega} \times 5\ mA < 30\ V \qquad \therefore R < 18\ k\Omega$$

Thus, the value of R becomes 18 kΩ or less. However, if the value is too small, heating value increases, resulting in the need of a resistor with large wattage.

Assume that the value of R is <u>15 kΩ</u>. Then the wattage W of the dummy resistor becomes;

$$W = \frac{(power\ source)^2}{R} = \frac{(100\ V)^2}{20\ k\Omega} = 0.67\ W \qquad \therefore W = 0.67\ W$$

Generally, the wattage W of the dummy resistor is taken to be <u>2 W</u> to provide a surplus wattage about three time more than required.

(4) ON/OFF Conditions of DC Input Module

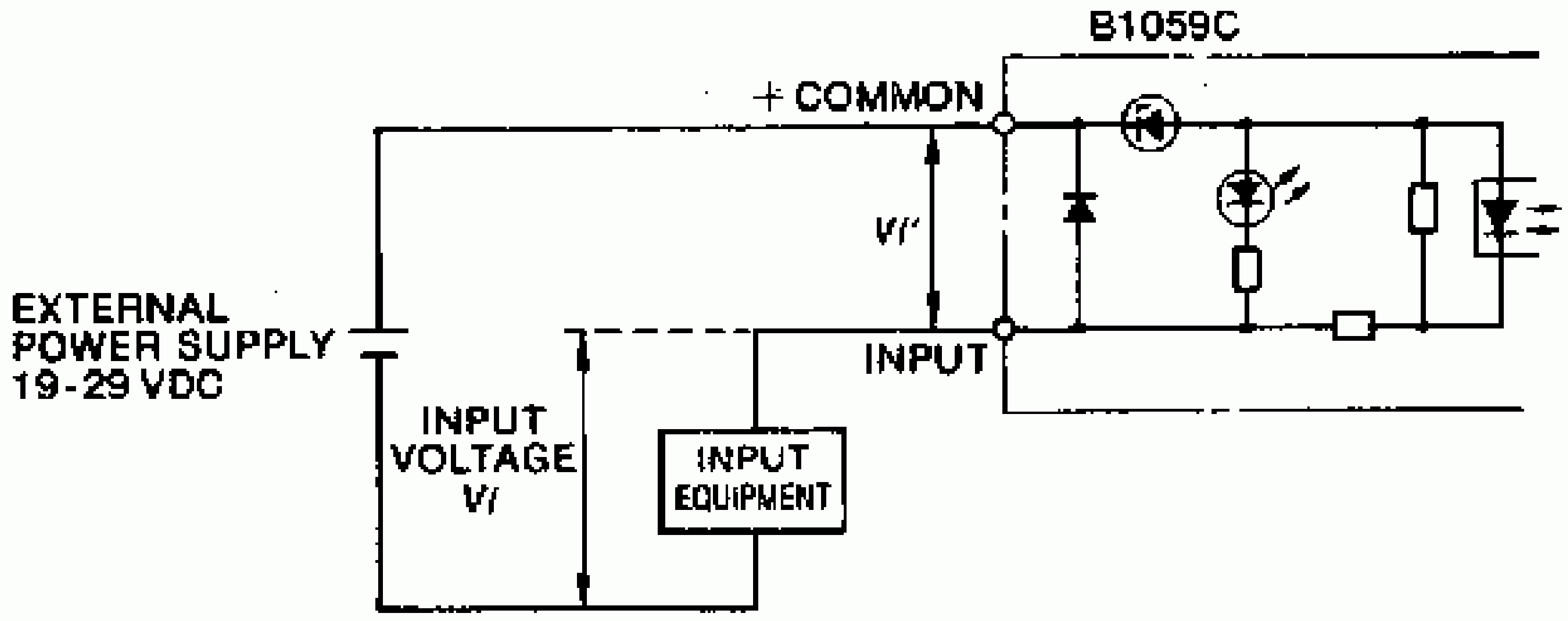


Fig. 5.55 ON/OFF Conditions of B1059C

Input conidtions of B1059C are;

ON level: 8 VDC or less

OFF level: 14 VDC or more

However, these values reflect the input voltage Vi in Fig. 5.43 when external voltage is 24 VDC. Accurate input conditions at external voltage Vo are;

ON level: Vo - 16 V or less

OFF level: Vo - 10 V or more

These conditions are summarized as follows:

Table 5.25 Relationship of External Power Supply Voltage and Input Conditions for Vi

	External Power Supply Voltage		
	19 VDC	24 VDC	29 VDC
ON Level	3 VDC or less	8 VDC or less	13 VDC or less
OFF Level	9 VDC or more	14 VDC or more	19 VDC or more

If input equipment with leakage current during OFF time is connected to other DC input modules (B1053, B1057, B1061, B1063, and B1065), input signals in OFF state may be called up as in ON state due to leakage current, or the input signal state indicator may turn on.

(Example) When a limit switch with LED is connected to B1059C

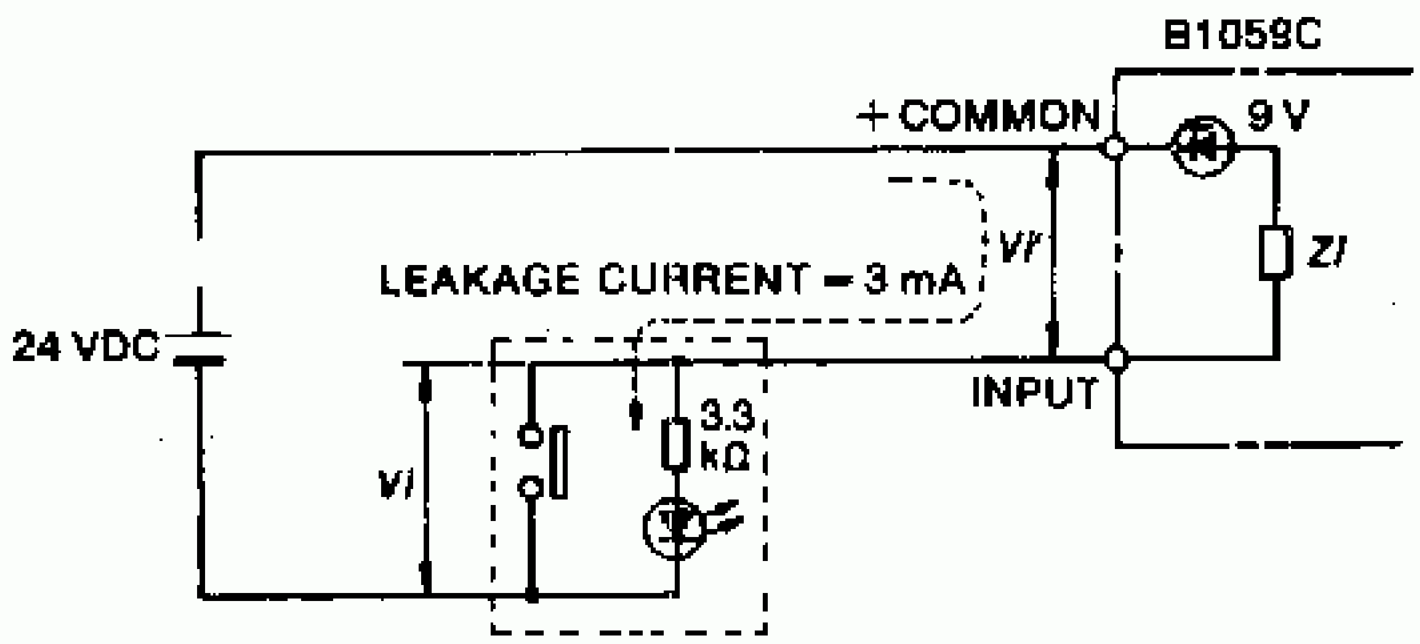


Fig. 5.56 Connection of a Limit Switch with LED

(4) ON/OFF Conditions of DC Input Module (Cont'd)

If a serial resistor of LED is 3.3 kΩ and the leakage current is 3 mA, then

$$Vi = 3.3\ k\Omega \times 3\ mA = 9.9\ V$$

This does not satisfy the input condition (OFF level = 14 V or more). Therefore, add a proper dummy resistor to the input terminal of B1059C so that the input condition is satisfied.

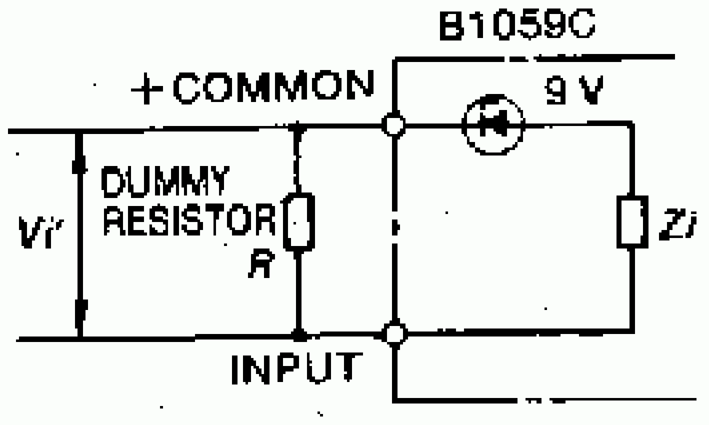


Fig. 5.57 Addition of a Dummy Resistor

The value of a dummy resistor R should be chosen such that the input voltage Vi' of B1059C becomes 10 V or less.

$$24\ V \times \frac{R}{3.3\ k\Omega + R} < 10\ V \qquad \therefore R < 2.35\ k\Omega$$

Thus, the value of the resistor becomes 2 kΩ.

Necessary wattage W is;

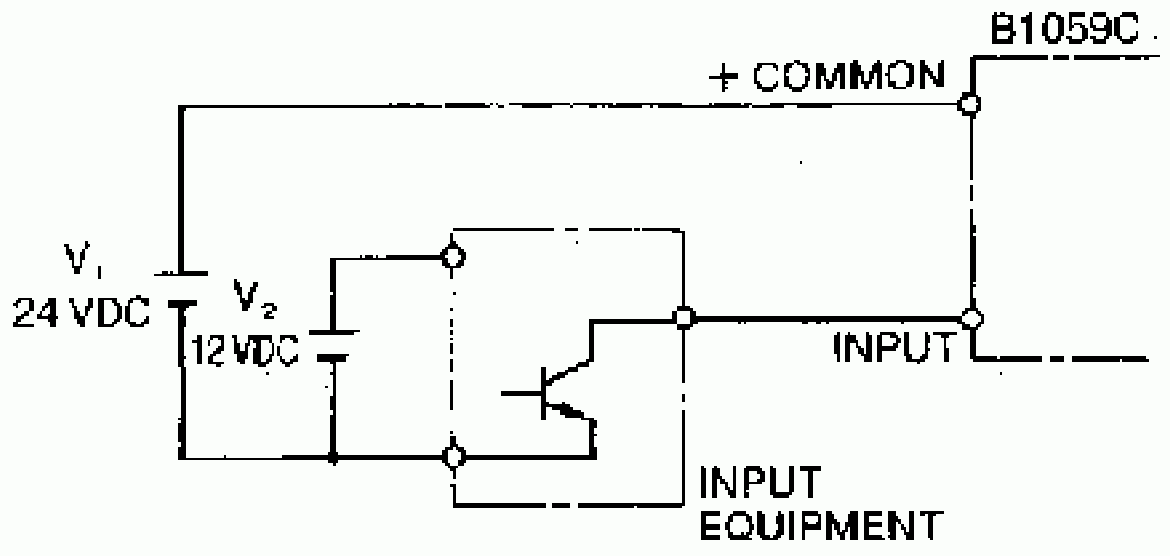
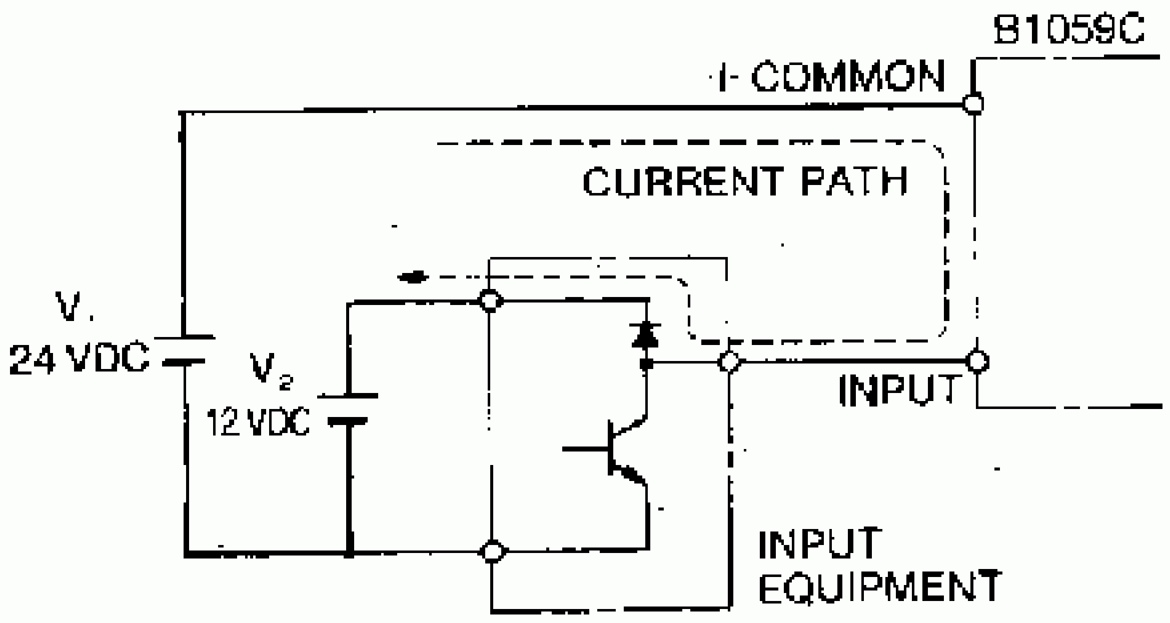
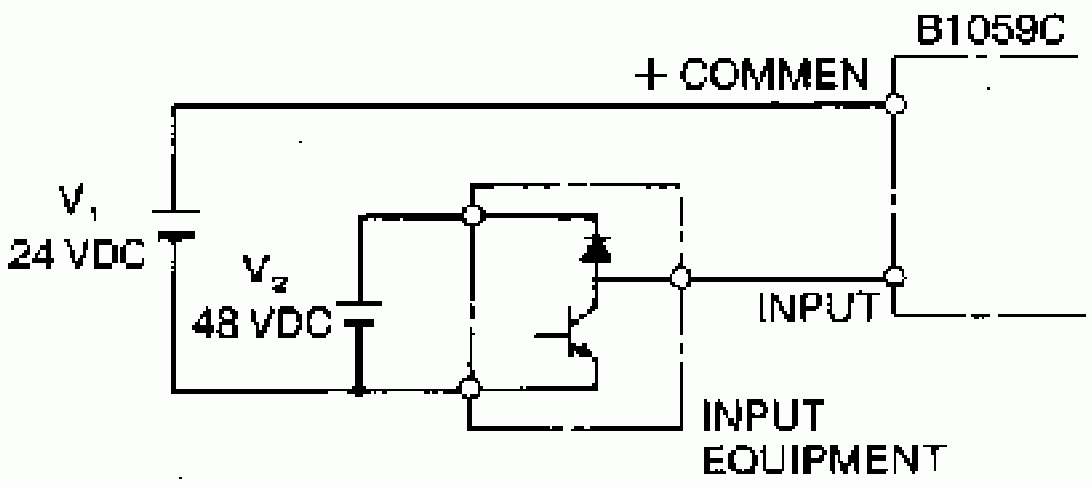
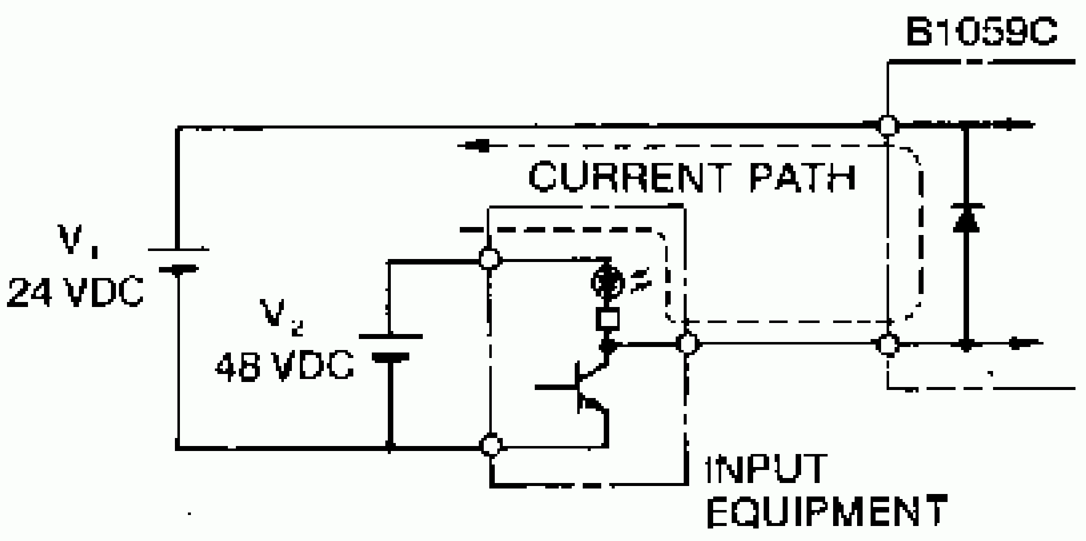
$$W = \frac{(power\ supply\ voltage)^2}{R} = \frac{(24\ V)^2}{2\ k\Omega} = about\ 300\ mW$$

In general, the wattage of a dummy resistor is taken to be 1W to provide a surplus wattage about three times more than required.

(5) Connection to Input Equipment with Different Voltage

Usually, power voltage of input equipment should be matched that of input modules. However, Table 5.26 shows possibilities of connecting input equipment having different voltages.

Table 5. 26 Possibilities of Connecting Input Equipment Having Different Voltages

Examples of Input Equipment	Connection Possibilities
① Open collector output ($V_1 > V_2$) 	Can be connected. However, the voltage resistance of the output transistor of the input equipment should be 40 V or more.
② With resistor, LED or diode ($V_1 > V_2$) 	Cannot be connected. When the the input equipment is OFF, current shown by a dotted line in the left figure may flow and input does not become OFF. Especially, in case of LED, reverse voltage may be applied during the OFF time to the equipment with LED and the LED may be broken.
③ With open collector or diode ($V_1 < V_2$) 	Can be connected.
④ With resistor or LED ($V_1 < V_2$) 	Cannot be connected. When the input equipment is OFF, current shown by a dotted line in the left figure may flow and the LED of the input equipment comes on dimly.

(6) Caution in Using B1061, B1063

Ambient temperature of B1061 (64-point) and B1063 (32-point) input modules depends on the external power supply voltage and the number of input points that are ON at the same time. Adjust the ambient temperature within a value shown in Fig. 5.58.

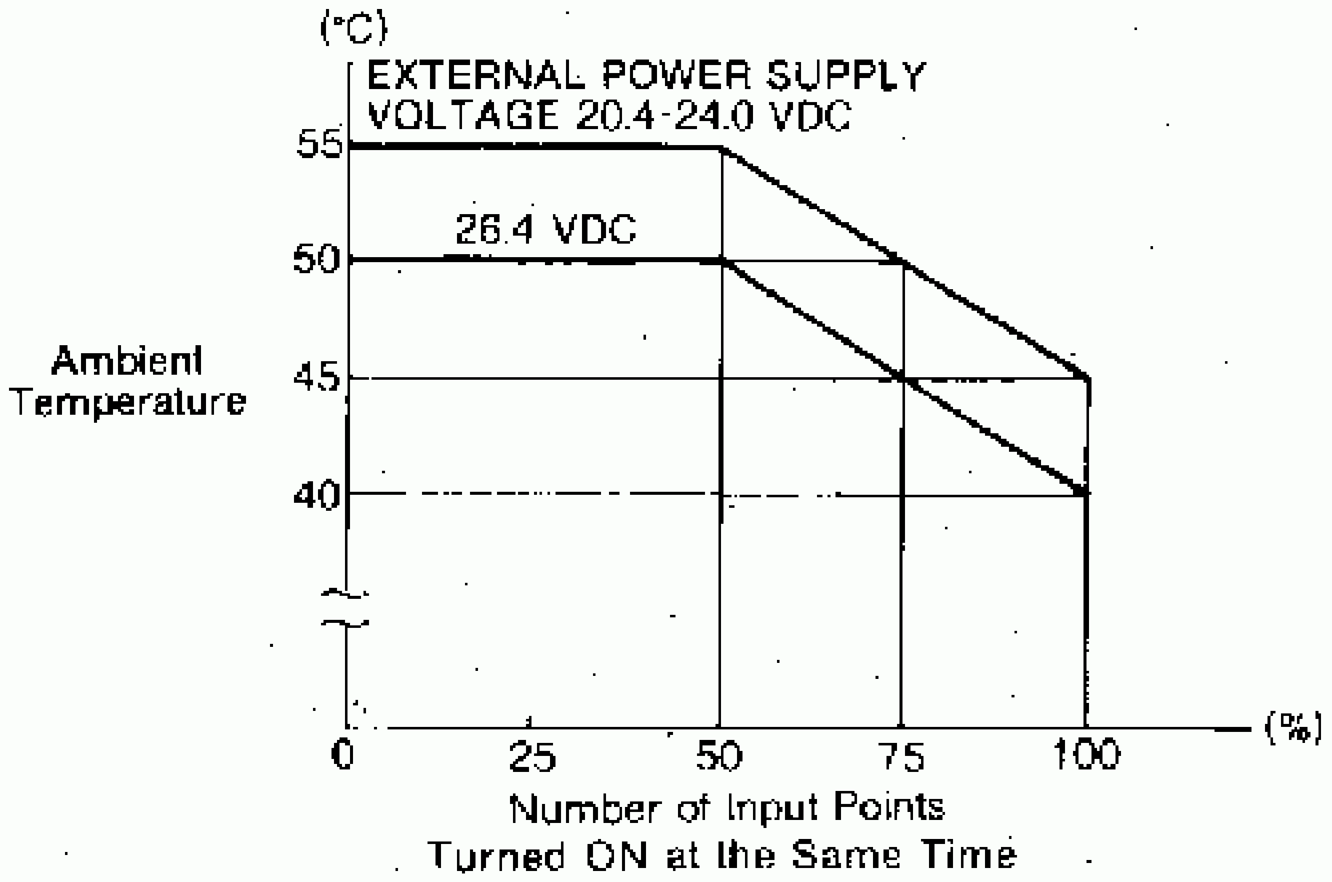


Fig. 5.58 Adjustment of Ambient Temperature

Table 5.27 shows the relationship between input status indicators and input signals of 64-point input module, B1061.

Table 5.27 Relationship between Input Status Indicators and Input Signals of B1061

Indicators	Input Signal No.		Indicators	Input Signal No.	
	Upper 32-point	Lower 32-point		Upper 32-point	Lower 32-point
A0	1	33	B0	17	49
A1	2	34	B1	18	50
A2	3	35	B2	19	51
A3	4	36	B3	20	52
A4	5	37	B4	21	53
A5	6	38	B5	22	54
A6	7	39	B6	23	55
A7	8	40	B7	24	56
A8	9	41	B8	25	57
A9	10	42	B9	26	58
AA	11	43	BA	27	59
AB	12	44	BB	28	60
AC	13	45	BC	29	61
AD	14	46	BD	30	62
AE	15	47	BE	31	63
AF	16	48	BF	32	64

For 32-point input module B1063, the relationship between input status indicators and input signals is as shown in Table 5.28.

Table 5.28 Relationship between Input Status Indicators and Input Signals of B1063

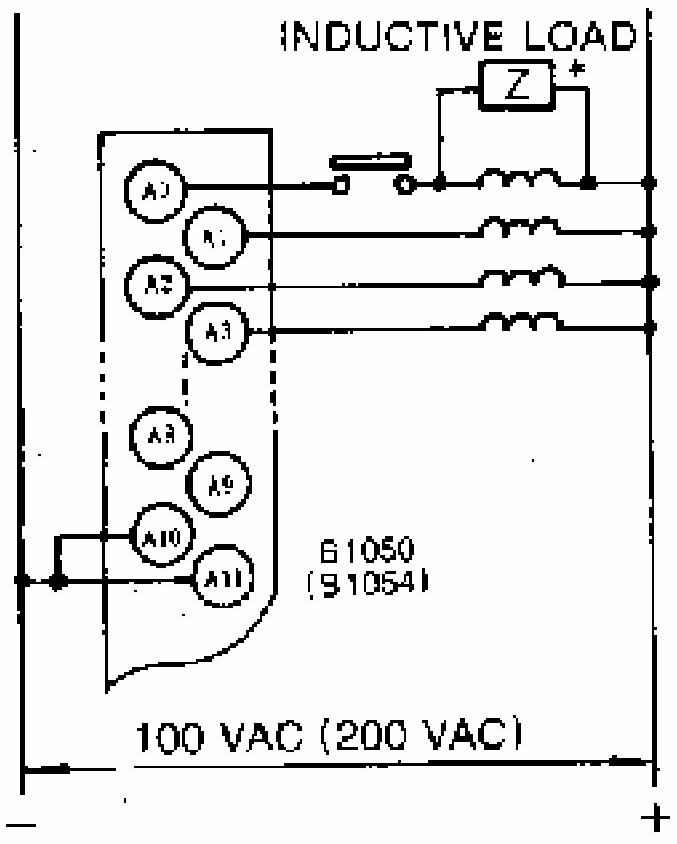
Indicators	Input Signal No.	Indicators	Input Signal No.
A0	1	B0	17
A1	2	B1	18
A2	3	B2	19
A3	4	B3	20
A4	5	B4	21
A5	6	B5	22
A6	7	B6	23
A7	8	B7	24
A8	9	B8	25
A9	10	B9	26
AA	11	BA	27
AB	12	BB	28
AC	13	BC	29
AD	14	BD	30
AE	15	BE	31
AF	16	BF	32

5. 2. 2 Output Module

(1) Connection to Contacts

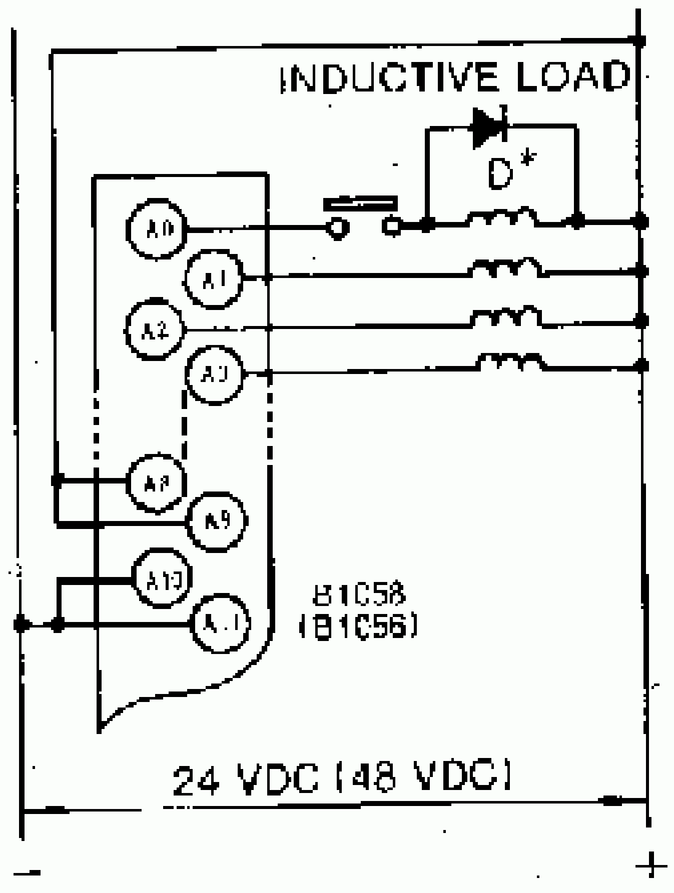
Where connecting contacts to an inductive load of the output module, as shown in Figs. 5.59 and 5.60, always connect a surge absorber or a flywheel diode in parallel to the inductive load.

Where connecting contacts in parallel to the output module, no surge absorber nor flywheel diode need be connected.



*The surge absorber capacity should be selected corresponding to the load. It is recommended that type CR 50500 (made by Okaya Electric Industries Co.) or equivalent be used.

Fig. 5. 59 AC Output Module



*The flywheel diode should be selected corresponding to the load. It is recommended that type F14 series (made by NEC) or equivalent be used.

Fig. 5. 60 DC Output Module

(2) Minimum Load Current

As the output switch of the AC output module, a triac is used. Since a triac cannot operate stably if the load is less than the specified minimum load current, make sure to use the load which is secure current levels above the minimum load current. If the minimum load current cannot be kept, connect a dummy resistor in parallel to the load so that the total load current is above the minimum load current.

(3) Maximum Load Current

Although an output point can accommodate a 2 A load, the total load for 8 output points must be up to 5 A. This should be taken into consideration for distributing loads.

(4) Output Fuse

The output fuse is used for preventing the trouble caused by shortcircuit of the load, but not for protecting the output element of the module.

(5) Output Status LED Indicator for 100 VAC Output Module

The output status LED indicator for the AC output module lights up by power supply for the internal logic circuit.

(6) Leakage Current from the Output Module

AC output module and relay contact output module contain a surge supressing circuit. Therefore, leakage current flows during OFF.

Table 5.29 Leakage Current in Output Modules

Output Module Type JAMSC-	Output Impedance during OFF (50 Hz)	Maximum Leakage Current
B1050	Approx 68 kΩ	Approx 2 mA at 130 VAC
B1054	Approx 145 kΩ	Approx 2 mA at 260 VAC
B1090B	Approx 145 kΩ	Approx 2 mA at 260 VAC

When a light-load relay is connected to these output modules, the relay does not turn off due to the current.

(Example) When load impedance is 6 kΩ and the load responds incorrectly due to 2 mA of leakage current.

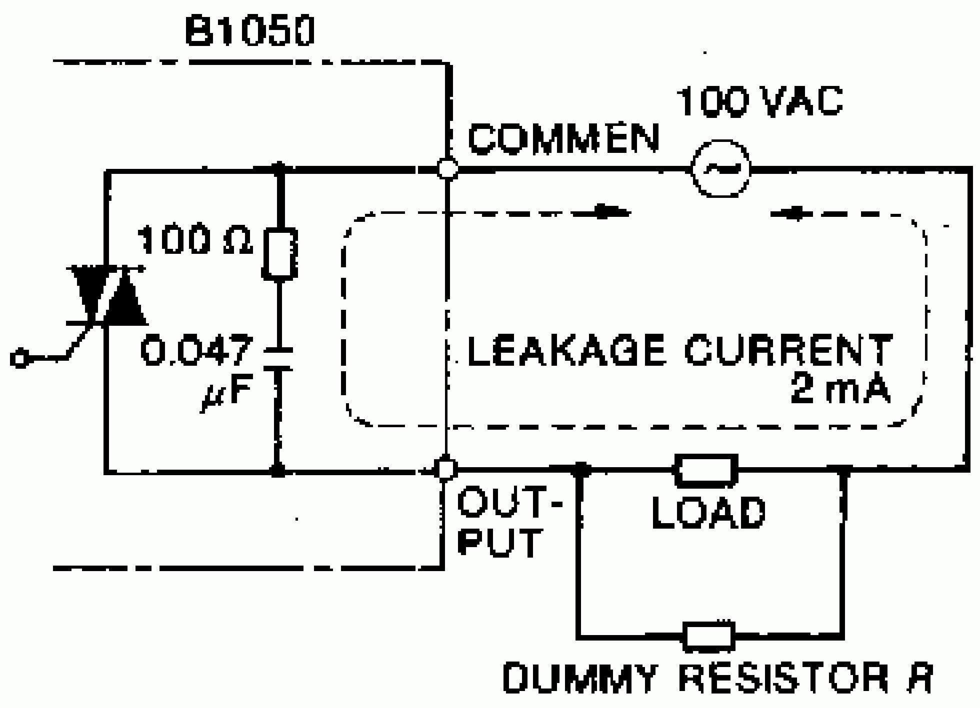


Fig. 5.61 Connection of Light Load

If 1 mA or less of current flow in the load does not cause any malfunciton, then the value of dummy resistor R becmoes;

$$2\ mA \times \frac{R}{R + 6\ k\Omega} < 1\ mA$$

$$\therefore R < 6\ k\Omega$$

Thus, make the valuc of R 5 kΩ.

Necessary wattage W is;

$$W = \frac{(power\ source\ voltage)^2}{R} = \frac{(100\ v)^2}{5\ k\Omega} = 2\ W$$

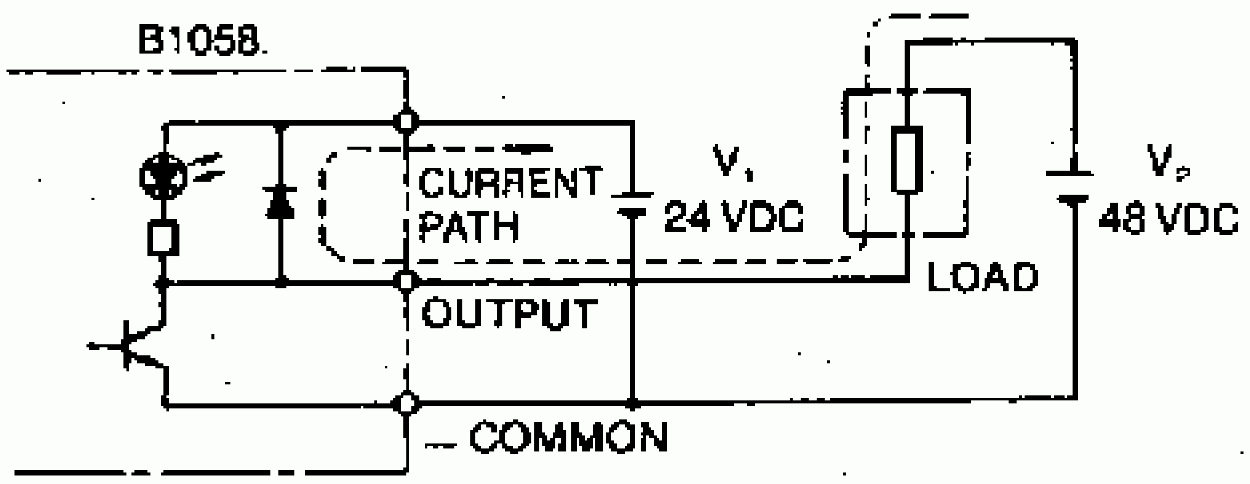
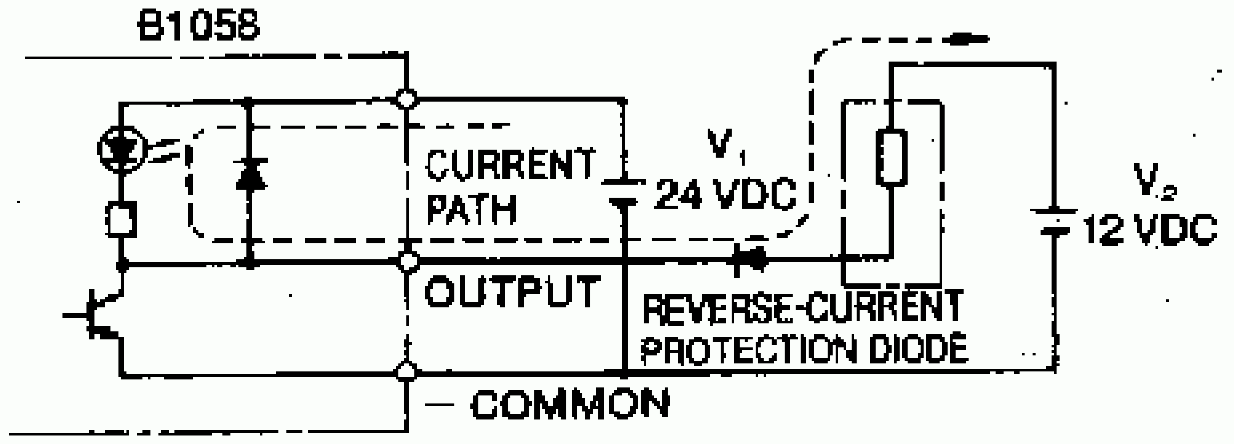
Generally, the wattage of the dummy resistor is taken to be 5 W to provide surplus wattage about 3 times more than required.

(7) Connection to a Load with different Voltage

In principle, power voltage of a load should be matched to that of an output module. Table 5.30 shows connection examples with loads having different voltages.

(7) Connection to a Load with different Voltage (Cont'd)

Table 5.30 Examples of Connections with Loads Having Different Voltages

Examples of Loads	Connection Possibilities
① High power voltage of load ($V_1 < V_2$)  	Cannot be connected. When B1058 is OFF, current shown by a dotted line in the left figure flows and the load does not go OFF completely. 35 V or more may be applied transiently to the output transistor.
② Low power voltage of load ($V_1 > V_2$)  	Cannot be connected. When B1058 is OFF, current shown by the dotted line flows and the output signal state indicator of B1058 comes on dimly. However, when a reverse-current protection diode is added as shown in the left figure, connection can be made.

(8) Connection of Solenoid with Diode

Solenoids with diodes have the advantage being driven by hlaf-wave rectification and less starting current. When solenoids with diodes are used as load of AC output module, be carefull of the following points.

(1) When output is OFF, overvoltage is applied to load:

When output of AC output module is OFF, current Ⓐ, shown by a dotted line, flows at half-cycle of the power supply to be a forward-biased diode for rectification, and is charged on capacitors. See Fig. 5.62.

With next half-wave, after polarity is reversed, diode for rectification is reverse-biased and current Ⓐ is blocked; discharge current Ⓑ, shown by a dotted line, flows from the capacitor. At this time, supply voltage and voltage charged on the capacitor are superimposed, and applied to the rectification solenoid with diode. The peak value of this voltage is approximately $2\sqrt{2}$ E (E: supply voltage). Rectification diode should require a withstand reverse voltage of $2\sqrt{2}$ E or more.

Connect a resistance of approximate multiples of 10 kΩ to several hundred kΩ on solenoid ends so that voltage applied to the solenoid is reduced. See Fig. 5.63.

(2) When output is ON, solenoid may not turn ON:

AC output module does not allow that output is ON even if receiving the ON signal before voltage of output ends are limited to approximately 50 V or less (for B1054). This is for reducing the inrush current when output is turned ON.

Where the solenoid with diode is connected, solenoid may not turn ON because voltage of output ends is not reduced to operation level by effect of voltage charged in the capacitor. Connect a resistance of approximate multiples of 10 kΩ to several hundred kΩ on solenoid ends so that voltage applied to solenoid is reduced. See Fig. 5.63.

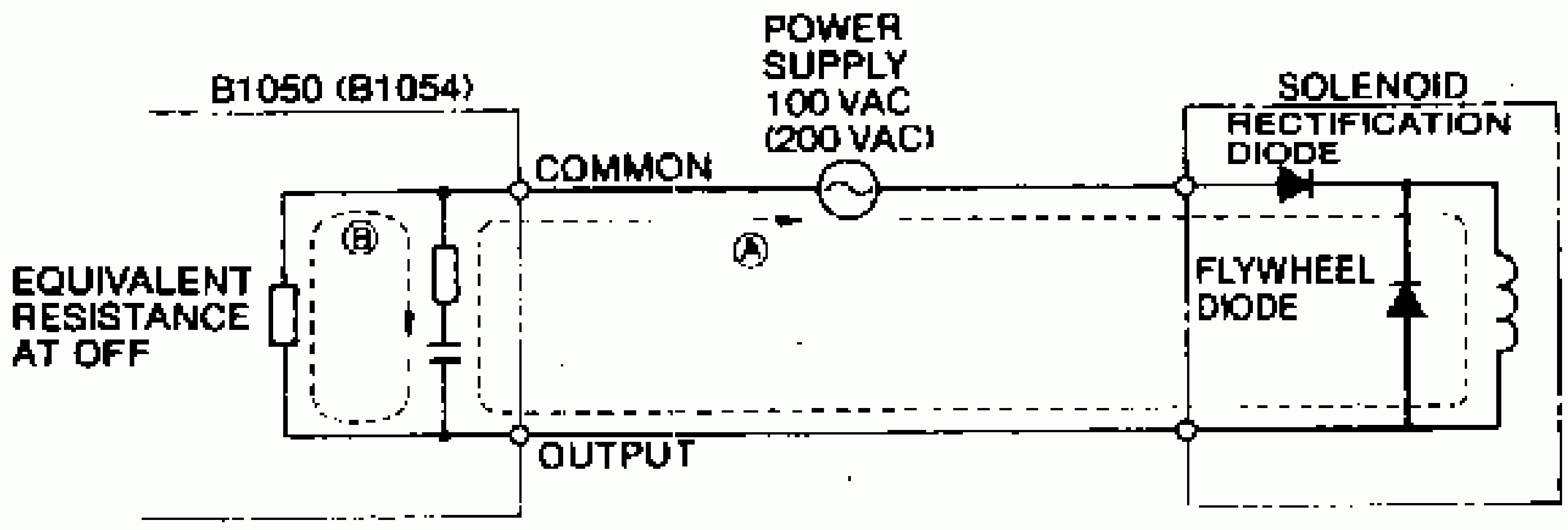


Fig. 5. 62 Connection of Solenoid with Diode

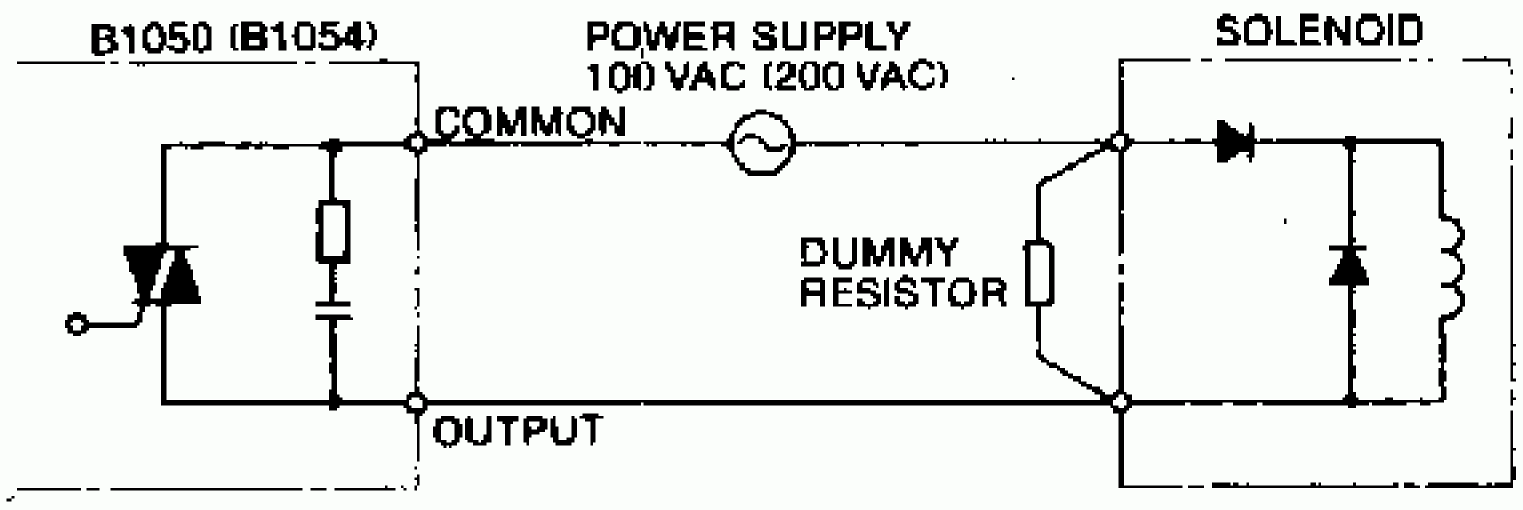


Fig. 5. 63 Connection of Dummy Resistor

(9) Caution in Using B1060, B1062

Ambient temperature of B1060(64-point) and B1062(32-point) output modules, depends on the external power supply voltage and the number of output points that are ON at the same time. Adjust the ambient temperature within a value shown in Fig. 5.64.

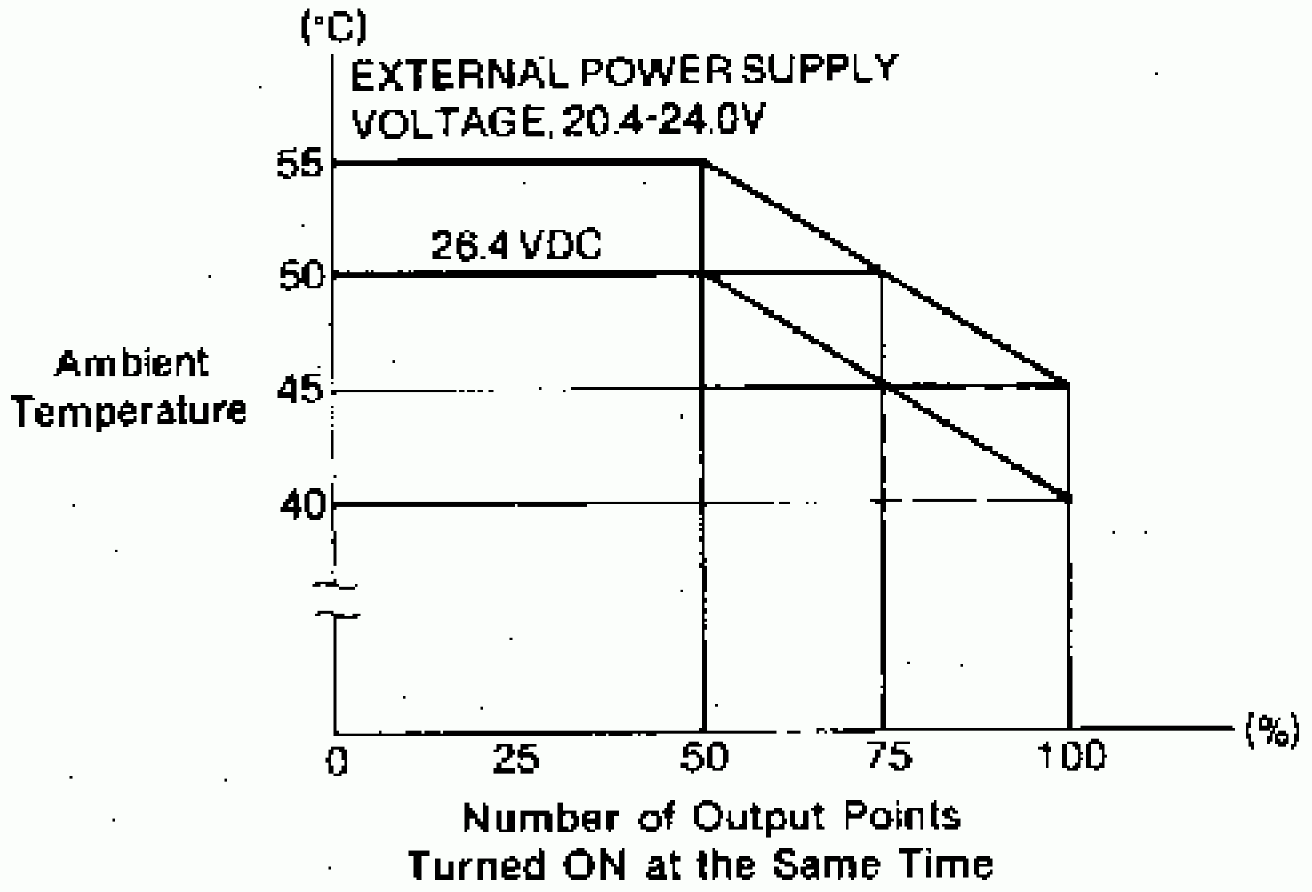


Fig. 5.64 Adjustment of Ambient Temperature

Table 5.31 shows the relationship between output status indicators and output signals of 64-point output module, B1060.

Table 5.31 Relationship between Output Status Indicators and Output Signals of B1060

Indicators	Output Signal No.		Indicators	Output Signal No.	
	Upper 32-point	Lower 32-point		Upper 32-point	Lower 32-point
A0	1	33	B0	17	49
A1	2	34	B1	18	50
A2	3	35	B2	19	51
A3	4	36	B3	20	52
A4	5	37	B4	21	53
A5	6	38	B5	22	54
A6	7	39	B6	23	55
A7	8	40	B7	24	56
A8	9	41	B8	25	57
A9	10	42	B9	26	58
AA	11	43	BA	27	59
AB	12	44	BB	28	60
AC	13	45	BC	29	61
AD	14	46	BD	30	62
AE	15	47	BE	31	63
AF	16	48	BF	32	64

For 32-point output module B1062, the relationship between output status indicators and output signals is as shown in Table 5.32.

Table 5.32 Relationship between Output Status Indicators and Output Signals of B1062

Indicators	Output Signal No.	Indicators	Output Signal No.
A0	1	B0	17
A1	2	B1	18
A2	3	B2	19
A3	4	B3	20
A4	5	B4	21
A5	6	B5	22
A6	7	B6	23
A7	8	B7	24
A8	9	B8	25
A9	10	B9	26
AA	11	BA	27
AB	12	BB	28
AC	13	BC	29
AD	14	BD	30
AE	15	BE	31
AF	16	BF	32

(10) Connection of B1094

Power reed relay output module, B1094 has a polarity on contact output as shown in Fig. 5.65. When DC power supply is used as a load, observe the correct polarity. If using an opposite polarity, electrical life of the contact may be shortened.

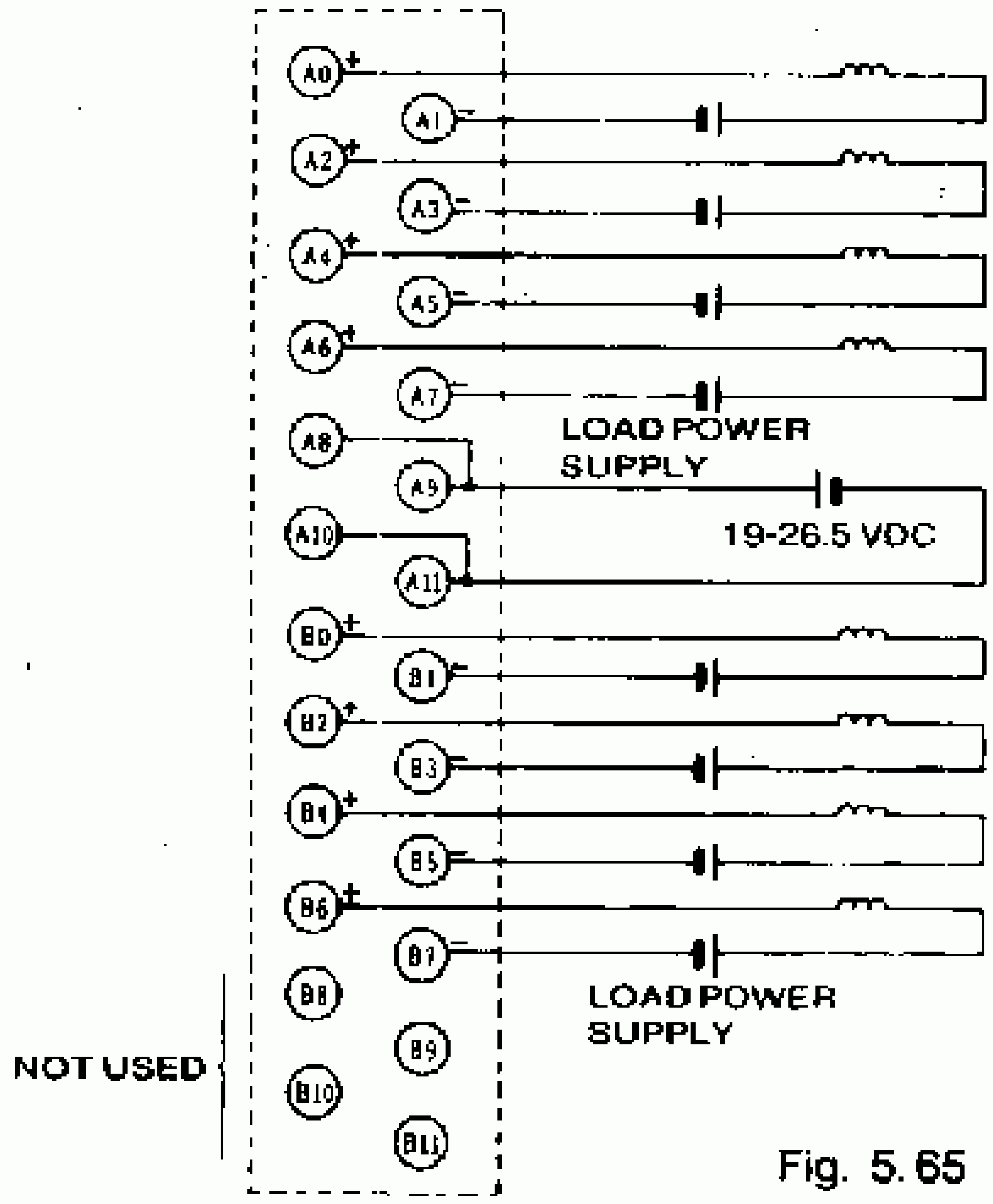


Fig. 5.65 Connection of B1094 and DC Load

5.2.3 Connection between Input/Output Modules

Where two or more U84 controllers are used in a system, and signals are exchanged between U84 controllers via input and output modules, connections should be as shown in Figs. 5.66 and 5.67. In this case, make sure to use modules of the same voltage rating, and connect a dummy resistor to the output module, to make stable operation.

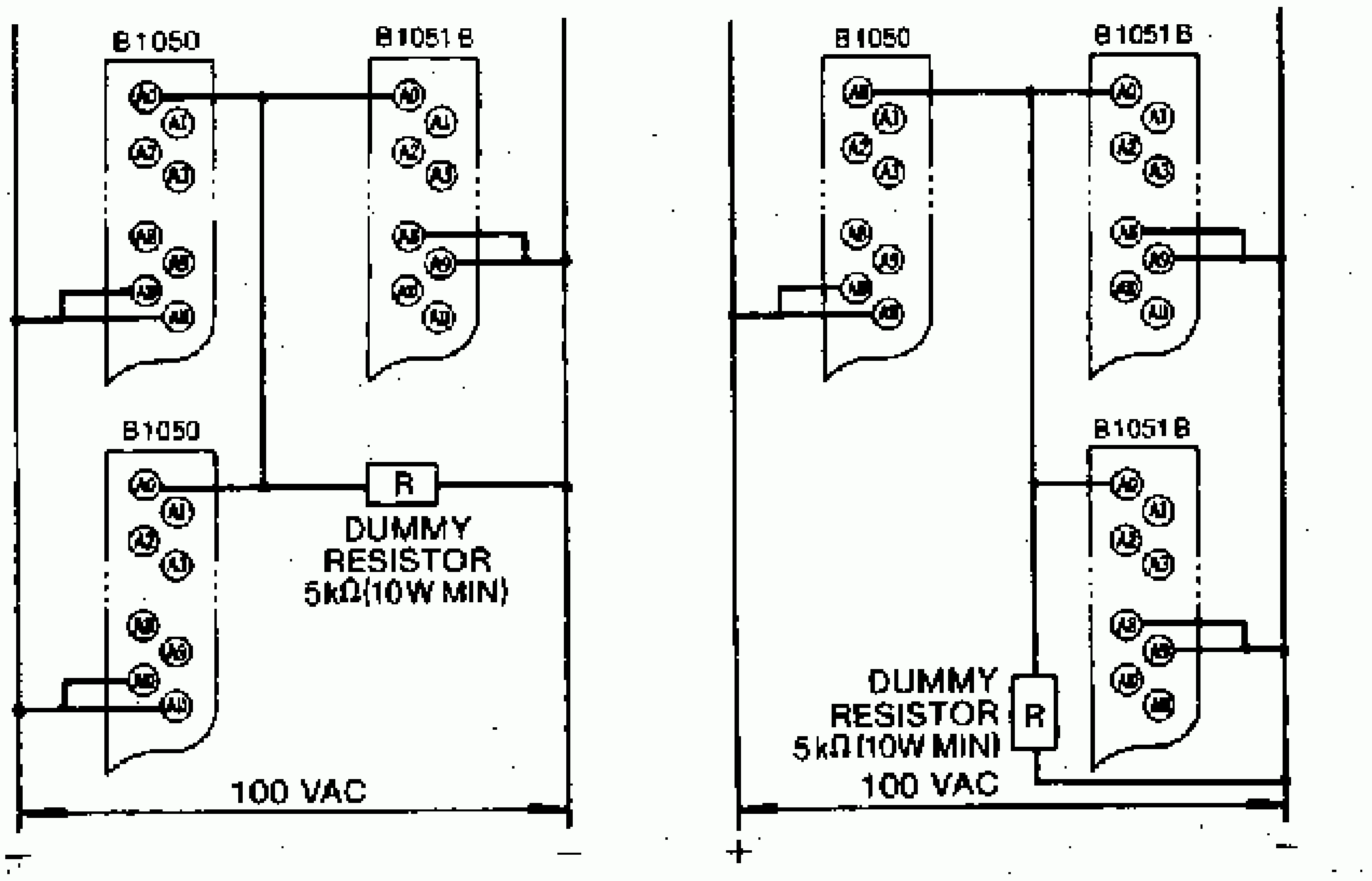


Fig. 5.66 Connection between AC Input and Output Modules

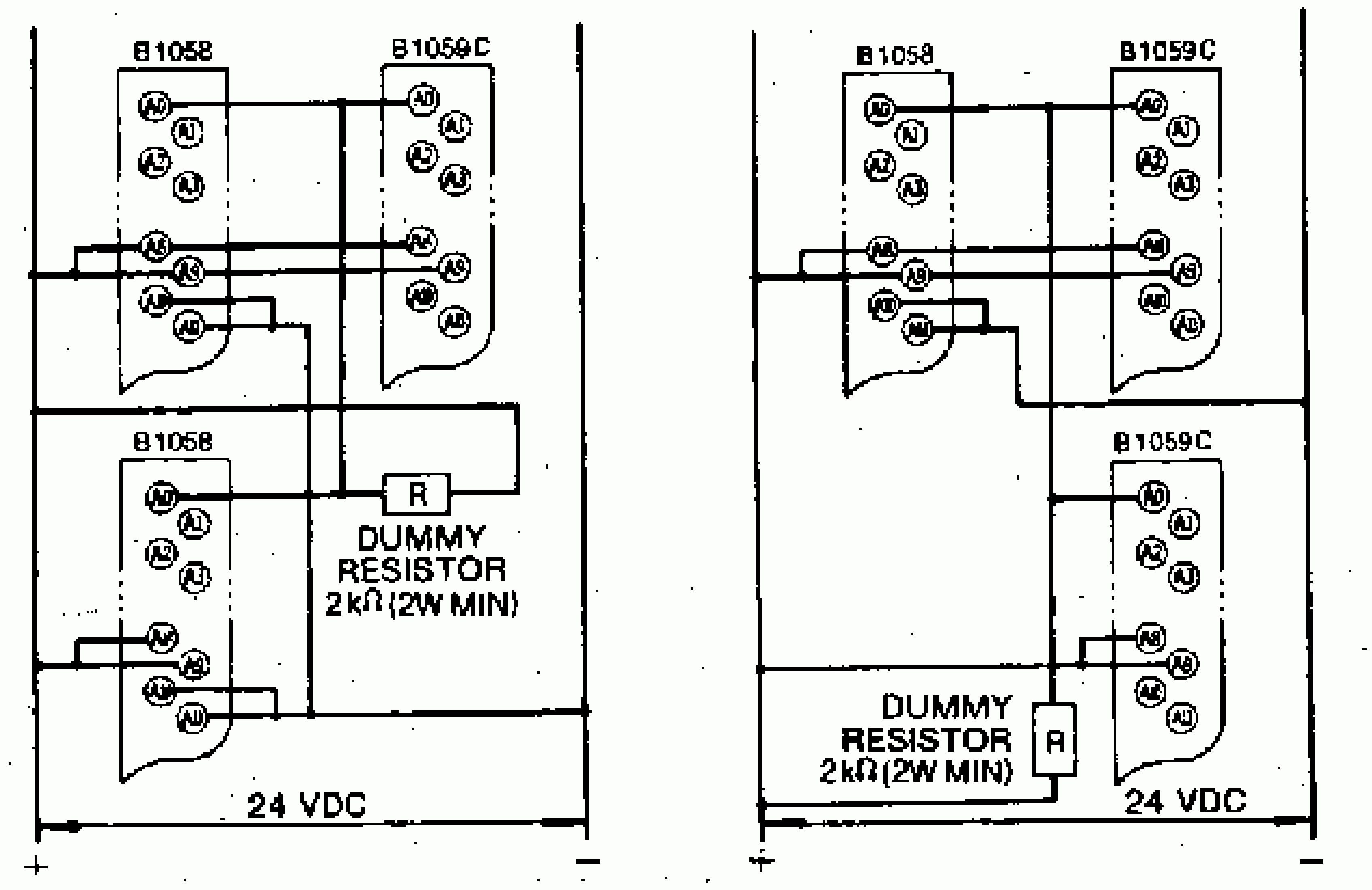


Fig. 5.67 Connection between DC Input and Output Modules

5. 2. 4 External Power Supply

General DC stabilized power supply should be used for DC I/O modules as an external power supply. Add a line filter on the AC input side of the DC stabilized power supply for special modules such as a register module, analog module, or counter module. Do not run the primary and the secondary side of the noise filter and the DC output side in the same wire duct.

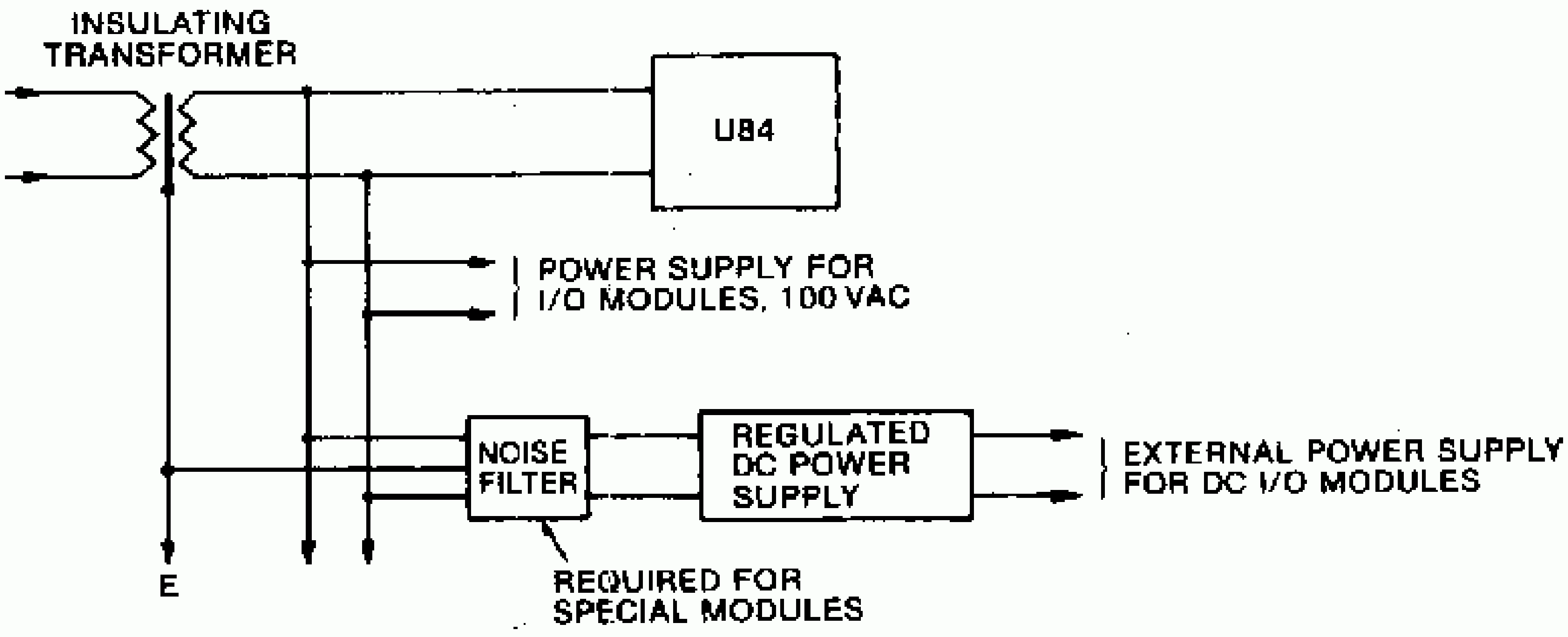


Fig. 5. 68 External Power Supply for DC I/O Modules

If it is necessary to use a simple DC power supply such as full wave rectification, minimize the ripple by adding a smoothing capacitor. The following should be observed:

- Instantaneous output voltage with ripple should always be within the range of the operation voltage of the DC I/O modules.
- Output voltage, including that of the power ON time and power OFF time, should never exceed the transient voltage of the DC I/O modules.
- Prevent the introduction of surge voltage by adding a line filter on the input side of a rectifying device.
- Where utilizing contactor, etc. in full-wave rectification output circuit, preventing the generation of surge voltage is an important circuit design concept. Add a surge suppressor between contactor output lines of module side as shown in Fig. 5.69.

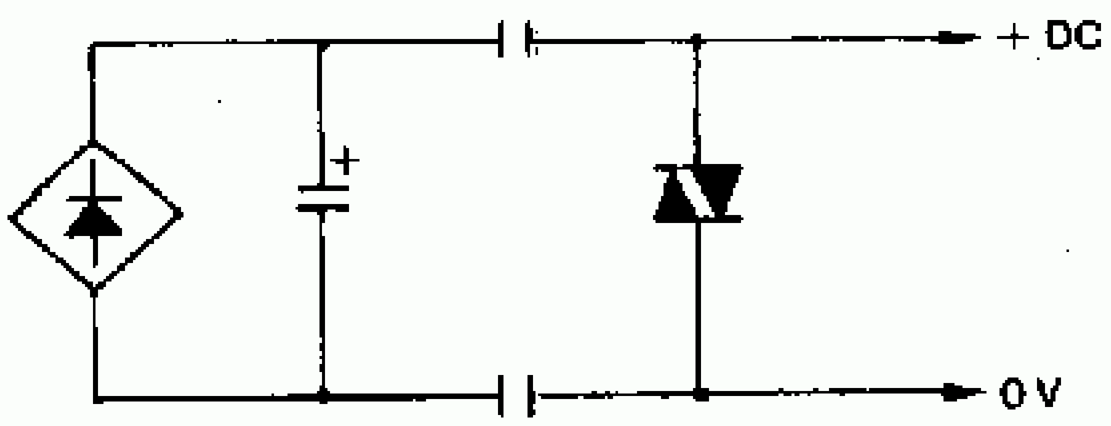


Fig. 5. 69

Precautions When Installing Modules

When installing I/O modules on racks 1, 2, 3 and/or 4, the number of modules to be installed must be limited so that the total consummed current of the modules to be used will not exceed the capacity of the internal power (supplied via a mounting base from a power supply module). Refer to Fig. 5.70. In this case, 16-and 32-point input modules are excepted.

(1) U84 Local I/O System

(2) U84 Remote I/O System

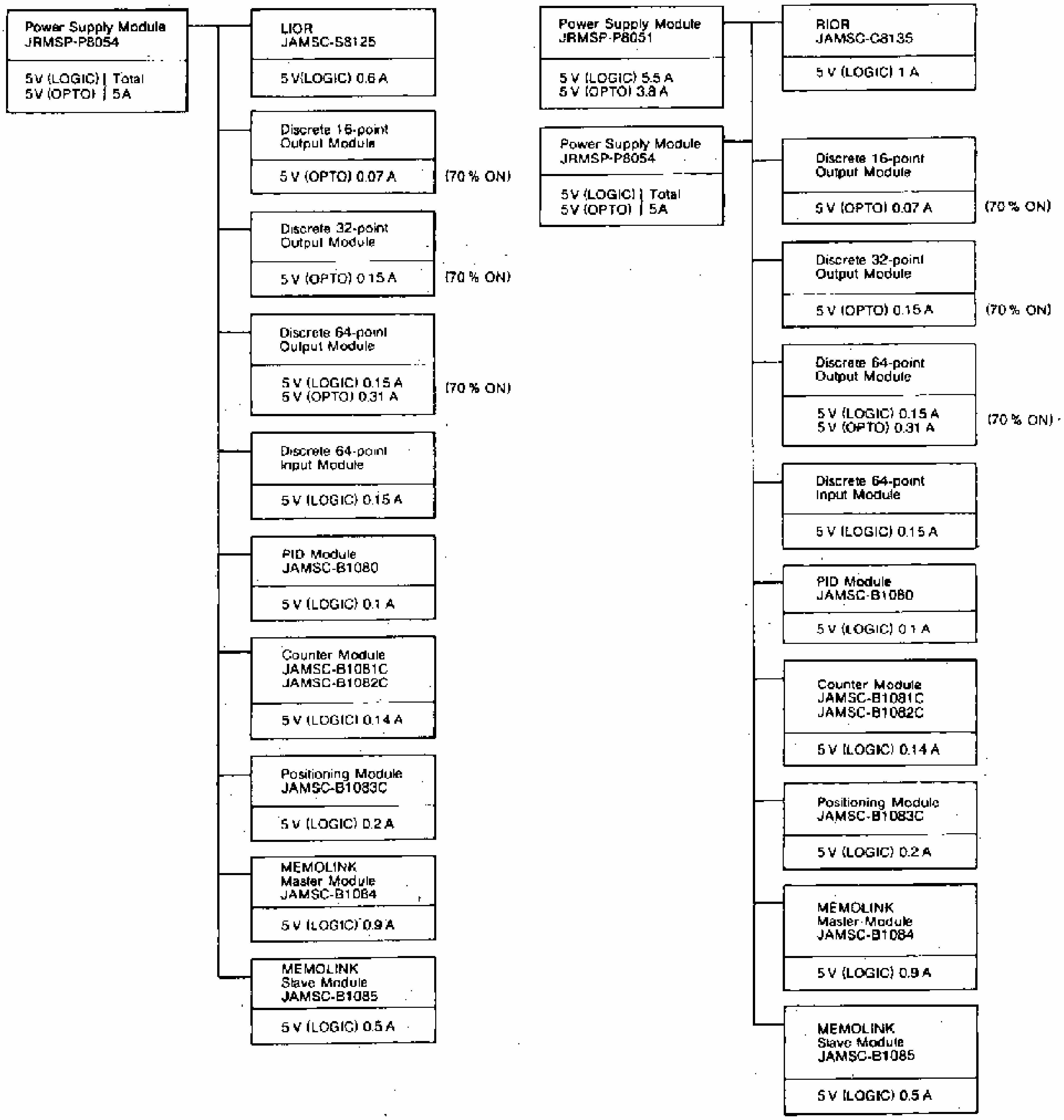


Fig. 5.70 Consumed Current of Each Module

SECTION 6
U84 APPLICATIONS

6.1 NOTES ON APPLICATIONS

Memocon-SC U84 should be used to meet your system specifications paying attention to the following points.

6.1.1 Backup Circuit

Since U84 is more reliable than relays, and also their eventual faults can be repaired quickly, it requires no backup circuit in ordinary systems. However, when it does require a backup circuit because of the special nature of the system, the selection of a proper backup method is an important consideration. An external manual circuit and standby U84 are sound methods.

6.1.2 Interlock

The U84 CPU is provided with a self-diagnosis function of stopping operation and turning output OFF when the stored ladder circuits (program) are destroyed, or when module cards develop faults. However, some faults and misoperations are not detected, and may destroy machines and devices. Where there is a possibility of such destruction, start and stop the system under redundant control such as external interlock and electric-mechanical redundant control.

6.1.3 Control Panel Layout

Lay out the control panel locations in consideration of the electrical environment conditions. For details, refer to Para. 6.3 "CONSTRUCTION, INSTALLATION AND WIRING OF CONTROL PANEL."

6.1.4 Network Processing

The network processing method of U84 has been explained in this manual. However, for reading the ON and OFF startes of all input signals into the CPU module of U84 correctly, the ON and OFF states of input signals must continue longer than the total delay time in the input module and one scan time. Therefore, when dealing with signals of short duration, special devices such as external memory circuits must be used, and for limit switch signals, the dog length must be sufficiently long. Refer to Fig. 6.1.

Output signals are also delayed up to the total of one scan time and the delay time within the output module. Therefore, for applications requiring a high degree of accuracy, some external devices are required. Refer to Fig. 6.2.

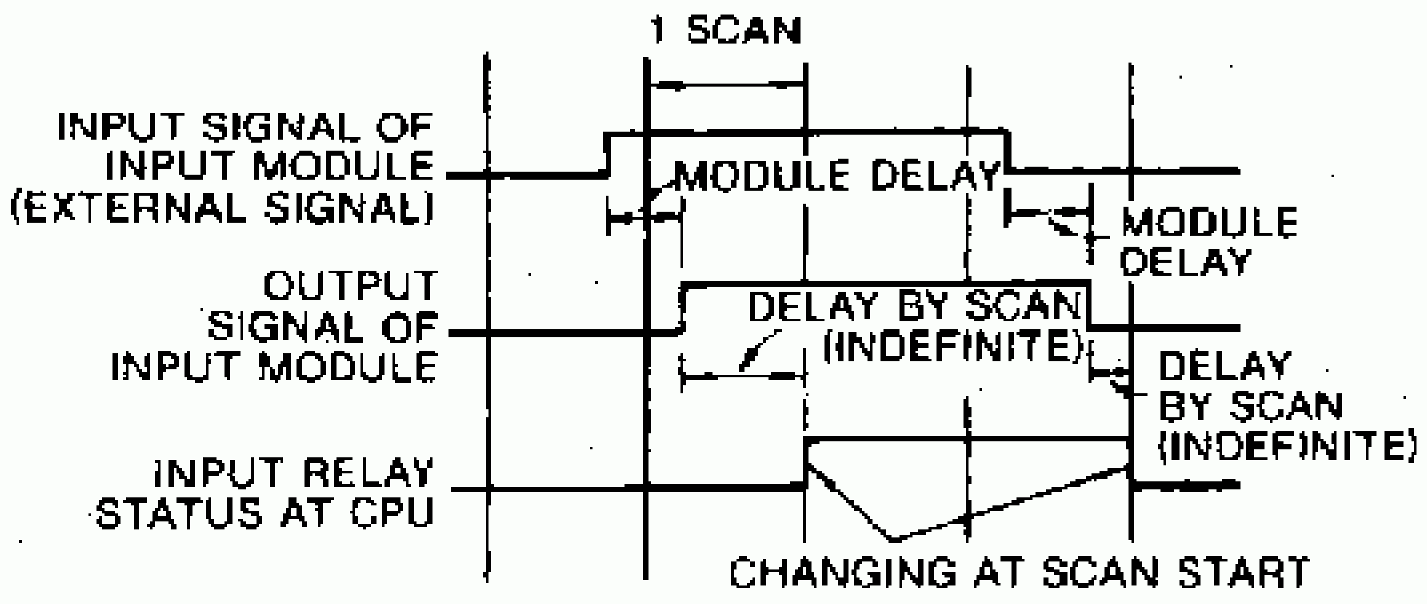


Fig. 6.1 Input Signal Delay

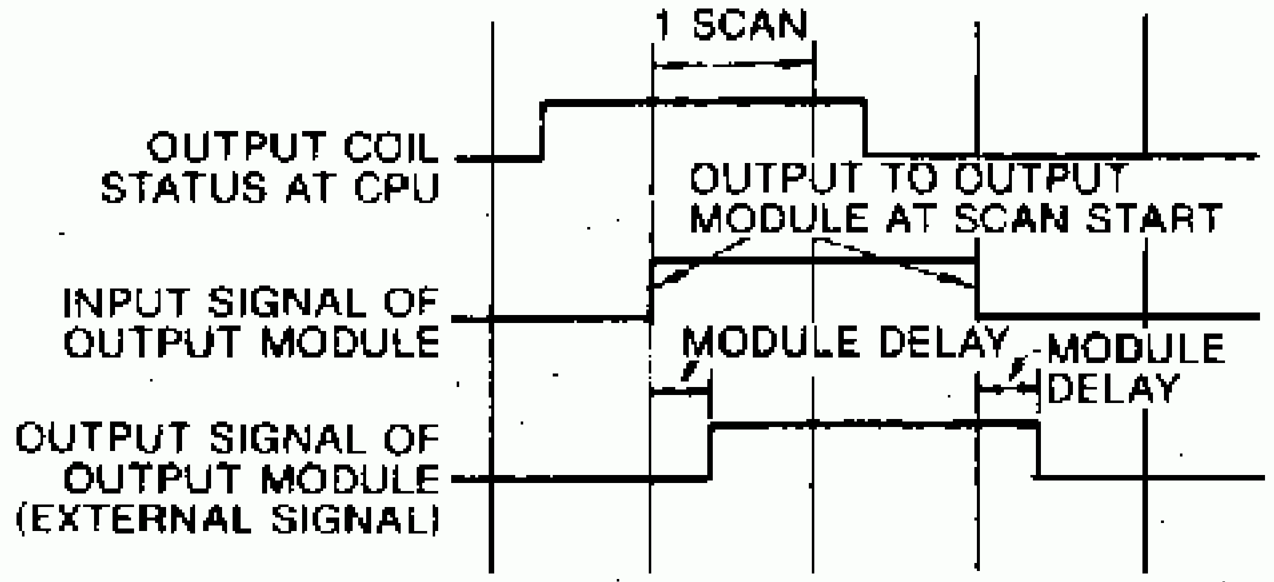


Fig. 6.2 Output Signal Delay

6.2 CALCULATION OF MEMORY CAPACITY

To find the required number of memories is a difficult task in composing any system. Exact numbers can only be determined from the intended ladder circuits, but here, the memory capacity of U84 may be roughly found as follows. Although more memories are required when the number of input and output signals increases, and when the sequence becomes more complex (more complex control),

40 × Number of Output Coils

is taken as a rough guideline regarding the number of memories (words).

Additional sequences may become required during trial and adjustment operations, and for this reason, some reserve memories should be prepared from the beginning.

6.3 CONSTRUCTION, INSTALLATION AND WIRING

The U84 systems are delivered with the CPU module, the power supply module, the input and output modules, the mounting base, the cable, etc. separated from each other.

These components must be installed in a control panel housing, the construction, layout, and wiring of which shall conform to the following standards. Where more strict conditions for the wiring, etc. are specified by separate specifications, etc., there should be given priority. For further details not sepcified below, applicable statutes or regulations apply.

6.3.1 Construction of Control Panel

For the control panel, the following construction is recommended.

- Enclosed steel housing, self-standing (or wall-mounting)
- Dustproof, or semi-dustproof
- Cooling: Where the panel-interior temperature (U84 ambient temperature) rises above 55°C, ceiling fan or other cooling devices must be used. A cooling fan should in principle be used to discharge air from the panel interior.

The respective modules have the heating value given in Table 6.1, the heating value for input and output modules applies when all the 16 points are simultaneously ON.

Table 6.1 Heating Value of Modules

Module (Type)		Heating Value W
CPU Module	U84—124	24
	U84—116	
Main Power Supply Module	(P8101)	150
Auxiliary Power Supply Module 1	(P8054)	25
Auxiliary Power Supply Module 2	(P8051)	60
Communication Module	(C8110)	10
Local I/O Driver Module	(C8120)	10
Electrical Remote I/O Driver Module	(C8130)	16
Optical Remote I/O Driver Module	(C8140)	16
ASCII Module	(C8160)	10
Local I/O Receiver Module	(S8125)	4
Electrical Remote I/O Receiver Module	(C8135)	6
Optical Remote I/O Receiver Module	(C8145)	6
I/O Buffer Module	(B1011)	1
100 VAC Input Module	(B1051B)	5
200 VAC Input Module	(B1055)	5
5-12 VDC Input Module	(B1053)	5
48 VDC Input Module	(B1057)	7
24 VDC Input Module	(B1059C)	4
24 VDC Input Module	(B1061, B1063)	8
24 VDC Input Module	(B1065)	5
100 VAC Output Module	(B1050)	10
200 VAC Output Module	(B1054)	10
5-12 VDC Output Module	(C1052)	7
48 VDC Output Module	(B1056)	10
24 VDC Output Module	(B1058)	10
24 VDC Output Module	(C1060)	8
24 VDC Output Module	(C1062)	13
24 VDC Output Module	(B1064)	10
Relay Output Module	(B1090B)	10
Power Reed Relay Output Module	(B1094)	5
Register Input Module	(B1071)	6
Register Output Module	(B1070)	6
Analog Input (A/D) Module	(B1073, B1075)	4
Analog Output (D/A) Module	(B1072B, B1074)	4
Reversible Counter Module	(B1081C)	10
Preset Counter Module	(B1082C)	10
Positioning Module	(B1083C)	8
PID Module	(B1080)	2
Power Supply Module	(B1089)	12

Note:

1. The heat generation from the mounting base is negligible.
2. The heating value of input and output modules when they are OFF (all points) is below 2W.

- Dimension

  Determine the size, etc. of the control panel by referring to the dimension of the each unit modules (Appendix B) and the U84 panel mounting dimension (Appendix C).

- Layout of mounting base

  Install these in the relative positions as shown in Appendix C, taking the cooling and other conditions into consideration.

6.3.2 Device Configuration in Control Panel

The modules of the U84 are mounted on six types of mounting bases. The mounting base is determined by the type of module. Each module must be installed on a fixed location of the mounting base (see Fig. 2.12-2.17).

When installing the U84 in a control panel, determine the device configuration by considering the layout of other devices and the following. In Appendix C, a sample layout of the U84 in a control panel is shown.

(1) Mounting Bases and I/O Cables

Mounting bases must be connected in the order shown in Fig. 6.3

The I/O cables shown in Table 6.2 are used for connection between mounting bases.

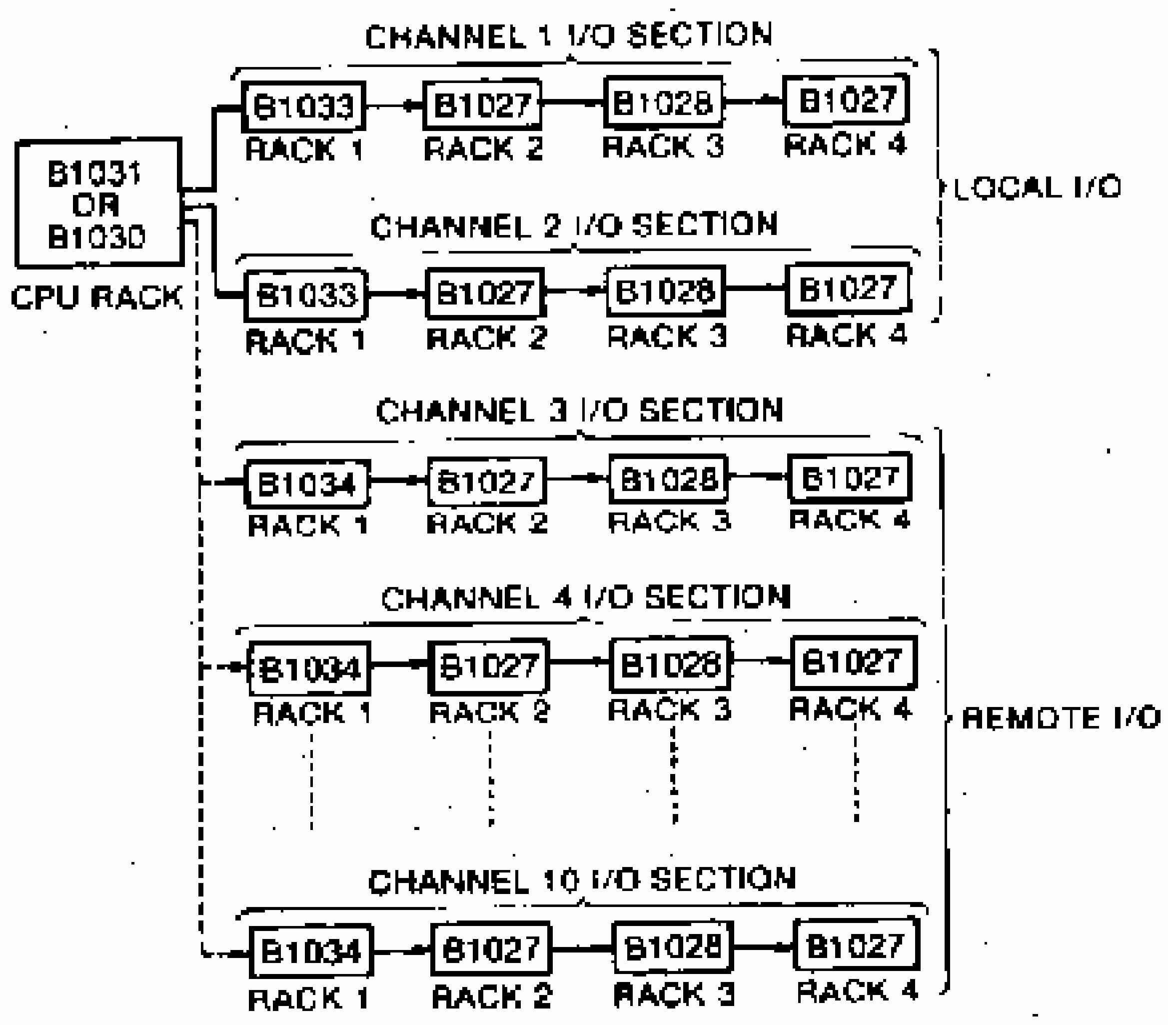


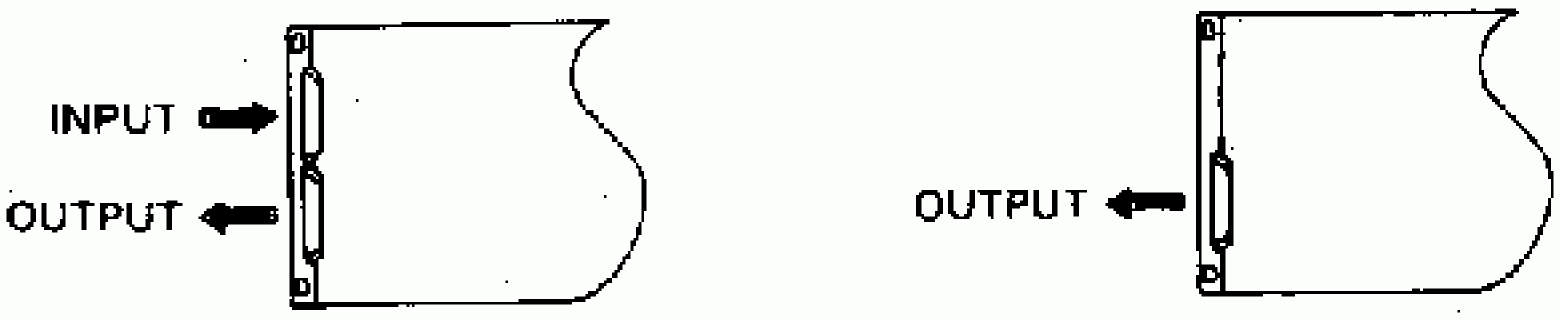
Fig. 6.3 Mounting Base Connection

Table 6.2 I/O Cable Specifications

Type JZMSZ-	Length m	Application
W1011-1	1	Used for connecting a local or remote I/O driver module connector on the mounting base B1031 to the connector on the upper place of the B1033.
W1011-2	5	
W1021	1.5	Used for connecting across each mounting base (B1033, B1034, B1027, B1028), respectively.
W1022	0.4	

Note: Only one W1021 can be used for a channel.

As for the two connectors provided on the left side of the B1033, B1034, B1027, and B1028, one on the top side is used for input lines and the other on the bottom side for output lines (see Fig. 6.4). B1034 is provided only one connector on the bottom side for output lines.



(a) B1033, B1027, B1028 (b) B1034

Fig. 6.4 Connectors for Connecting across Each Mounting Base (B1033, B1034, B1027, B1028) respectively

To determine the layout of mounting bases, take into consideration the order of connection (Fig. 6.3) and the types and lengths of the cables. The cables must be connected so that signal flows from the output to the input.

(2) Weight (Table 6. 3)

Table 6. 3 Weight of Module

Module (Type)		Approx Weight kg
CPU Module	(U84-124, U84-116)	2.0
Main Power Supply Module	(P8101)	5.0
Auxiliary Power Supply Module 1	(P8054)	1.1
Auxiliary Power Supply Module 2	(P8051)	2.8
Communication Module	(C8110)	1.4
Local I/O Driver Module	(C8120)	1.2
Electrical Remote I/O Driver Module	(C8130)	2.2
Optical Remote I/O Driver Module	(C8140)	1.4
ASCII Module	(C8160)	1.0
Local I/O Receiver Module	(S8125)	0.7
Electrical Remote I/O Driver Module	(C8135)	0.8
Optical Remote I/O Receiver Module	(C8145)	0.8
I/O Buffer Module	(B1011)	0.7
16-point Discrete Input Module	(B1051B, B1055, B1053, B1057, B1059C)	0.8
32-point Discrete I/O Module	(B1063, B1065, B1062, B1064)	0.8
64-point Discrete I/O Module	(B1061, B1060)	0.8
8-point Discrete Output Module	(B1094)	1.1
16-point Discrete Output Module	(B1050, B1054, B1052, B1056, B1058, B1090B)	1.1
Register I/O Module	(B1071, B1070)	0.8
Analog I/O Module	(B1073, B1075, B1072B, B1074)	1.0
Reversible Counter Module	(B1081C)	0.8
Preset Counter Module	(B1082C)	0.9
Positioning Module	(B1083C)	1.8
PID Module	(B1080)	1.0
Power Supply Module	(B1089)	1.0
Mounting Base	(B1030)	2.0
Mounting Base	(B1031)	2.8
Mounting Base	(B1033, B1034, B1027, B1028)	2.5
I/O Cable	(W1011-1)	0.3
I/O Cable	(W1011-2)	1.4
I/O Cable	(W1021)	0.5
I/O Cable	(W1022)	0.3

(3) Electrical Noise

- Avoid installing U84 together with elements or wires carrying high-voltage and large current power* in the same panel.
- When installing U84 together with the low-voltage main circuit† in the same panel, install the elements and wires related to the low-voltage main circuit as far apart from the U84 and its wiring as possible.
- Do not bind the U84 wiring together with general control circuit‡ wiring.
- Install the mounting base to a solid steel panel (frame). Never install these on a insulator. When the panel (frame) is painted, remove the paint from the area around the mounting holes before installing the base, in order to secure good grounding, and to prevent noise.

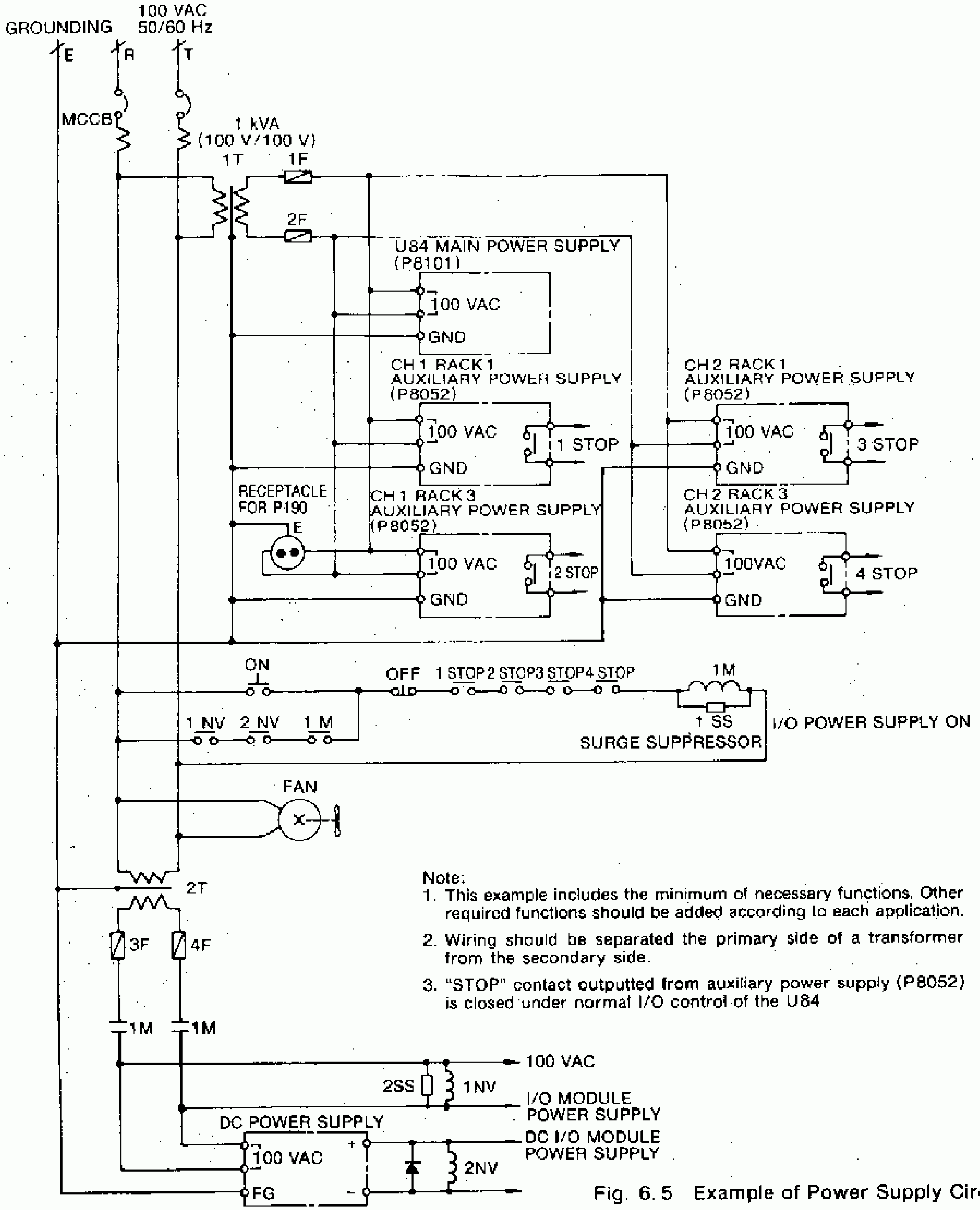
*Above 600 VAC, 750 VDC or 800 A

†Below 600 VAC or 750 VDC, with a current above 20 A

‡Below 600 VAC or 750 VDC, with a current below 20 A

(4) Power Supply Circuit

- Fig. 6.5 shows the example of power supply circuit.
- When the power supply is in an unfavorable condition, connect a line filter (noise filter) or an insulating transformer to the power supply line of a power supply module, input module and output module.
- The voltage and capacity of the power supply depend on the types of input and output modules and the connected loads. Determine these factors by referring to the I/O specifications in SECTION 5.
- U84 starts deciphering processes immediately when the power supply is turned ON. In some systems, the power module of U84 may have to be energized only after connecting the input and output power supply and determining the input and output states.



Note:
1. This example includes the minimum of necessary functions. Other required functions should be added according to each application.
2. Wiring should be separated the primary side of a transformer from the secondary side.
3. "STOP" contact outputted from auxiliary power supply (P8052) is closed under normal I/O control of the U84

Fig. 6.5 Example of Power Supply Circuit

(5) Wiring in Panel

The wiring related to U84 in the panel is in types shown in Table 6.4. Use the wires of the listed sizes.

Table 6.4 Types of Wiring in Panel

Type of Wiring	Wire Size mm^2	Description
Power Supply	1.25	To be connected to the power supply terminal "100 VAC" of power supply module, via circuit breaker, etc.
I/O Signal	0.3—1.25*	To be connected to input and output signal lines and input and output module terminals (two 1.25 mm^2 wires can be connected to one terminal).
Grounding	1.25	Connection between the GND terminal of the power supply module and the control panel housing (ground).

Note: For 32-point discrete module and register module, the wire size should be 0.3 mm^2 or less.

6.3.3 Grounding Wire

- The GND terminal of the power supply module should be connected to the control panel housing, as shown in Fig. 6.6 at E, and connecting point E should be connected to a ground pole.
- The grounding wire between point E and the ground pole should be larger than 8 mm^2 in the cross-sectional area, and should be as short as possible.
- The grounding resistance should be 100 Ω or less. (Ordinary building frames may be used. However, do not use a ground wire or ground pole in common with power lines, motors, etc.)

NOTE

When metal ducts, metal tubes or wiring racks are used, ground them in accordance with the accepted technical standards.

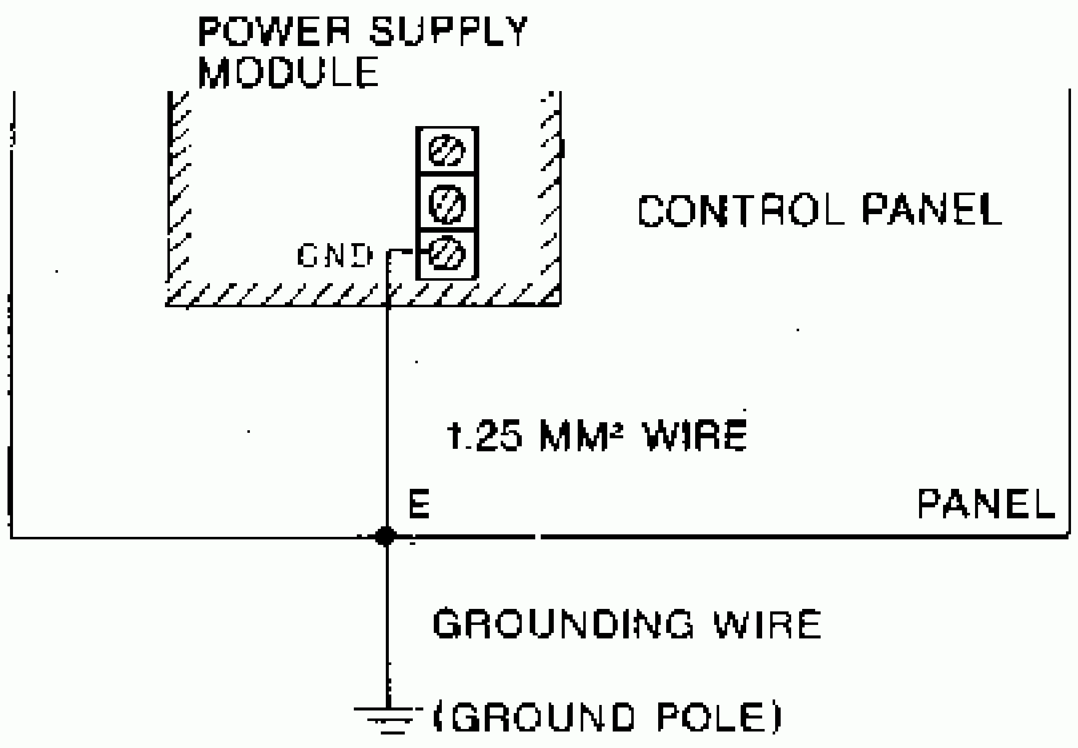


Fig. 6.6 Grounding Wire

6. 3. 4 External Wiring

(1) Cables for Input and Output Signal Lines

Cables to be used as external input and output signal lines should be selected in full consideration of the environmental conditions, mechanical strength, electrical noise, wiring length, operational voltage, etc. Isolate the input and output signal lines from each other and select cables on the basis of the guidelines given in Table 6.5.

(2) Installation of Input and Output Signal Line Cables

Since the input and output signal lines are low-voltage control circuit lines, separate these lines from ordinary control circuit lines and the main circuit lines as far as possible. Keep the space of minimum 10 centimeters between the lines. If they cannot separated, use the totally shielded cables and place the iron plate between the lines to separate them completely.

Table 6. 5 I/O Signal Line Cable Installation

Wiring Distance	Description
30 m max	• DC output signal lines and DC input signal lines may be contained in the same duct. AC output signal lines and AC input signal lines also may be contained in the same duct. • DC input/output signal lines and AC input/output signal lines should be contained in their respective ducts, separately.
30—300 m	• DC input signal lines, DC output signal lines, AC input signal lines and AC output signal lines should be contained in their respective ducts, separately. • Where induced voltage is high, connect a dummy resistor as described in 5.2 PRECAUTIONS FOR USING INPUT/OUTPUT MODULES, or use the totally shielded cables.
300 m min	• Do not use cables over 300 m, in view of the rush current to the output module. • Where the wiring distance is over 300 m, use a relay, and limit the wiring length between the relay and the control panel within 300 m.

6. 4 SPARE PARTS

Generally, it is recommended that the following spare parts should be stocked.

- CPU module: one unit
- Main power supply module: one unit
- Auxiliary power supply module 1: one unit
- Auxiliary power supply module 2: one unit
- Communication module: one unit
- Local I/O driver module: one unit
- Remote I/O driver module: one unit
- ASCII module: one unit
- Local I/O receiver module: one unit
- Remote I/O receiver module: one unit
- I/O buffer module: one unit
- Input and output modules: At least one unit each of all the used modules. Where many modules are used, 3% of the used number of modules are recommended as a spare parts quantity.

SECTION 7
U84 HANDLING AND MAINTENANCE

7. 1 U84 INSTALLATION PROCEDURE

The mounting bases, modules, and I/O cables are shipped separately. Follow the procedures described below when installing the U84 in a control panel.

7. 1.1 Installation of Mounting Bases

According to (1) of 6.3.2, determine the layout of mounting bases and drilled holes. Install wire ducts as necessary. There are four types of mounting bases. Install the mouning base by fastening four M5 screws through the four holes provided.

NOTE

The mounting base connectors are covered. When installing the mounting base, leave the cover installed over the connectors to protect them from foreign matter.

7. 1. 2 Installation of Modules

Install modules of the U84 on the fixed mounting bases. Fig. 7.1 shows how to install a module on a mounting base.

Remove the connector cover. Fit the guide posts of the module into the guide holes of the mounting base and push the module in. Then fasten the module to the mounting base with the M4 screws provided with the module.

NOTE

Do not remove the connector cover if no module is to be installed.

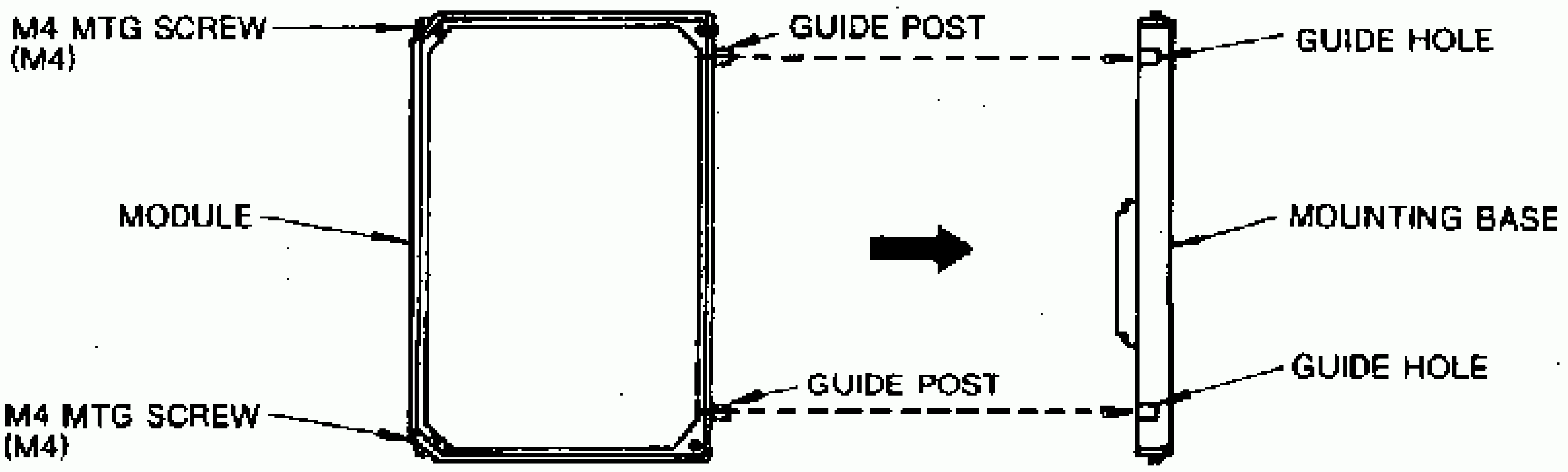


Fig. 7. 1 Module Mounting

7.1.2 Installation of Modules (Cont'd)

The type of mounting base and the mounting location are determined depending on module types. Figs. 7.2 to 7.7 show mounting place of each module on the mounting base.

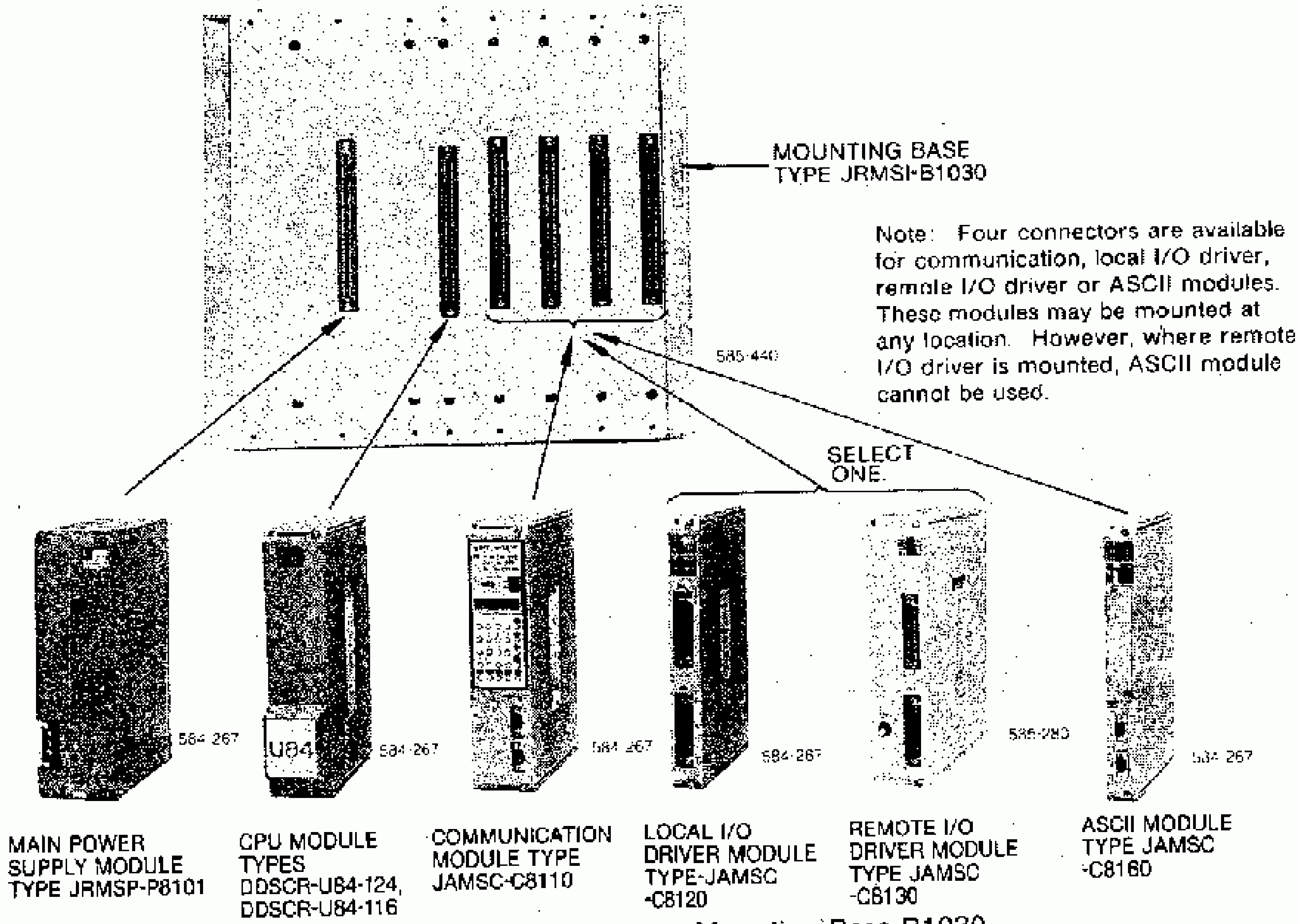


Fig. 7.2 Module Mounting on Mounting Base B1030

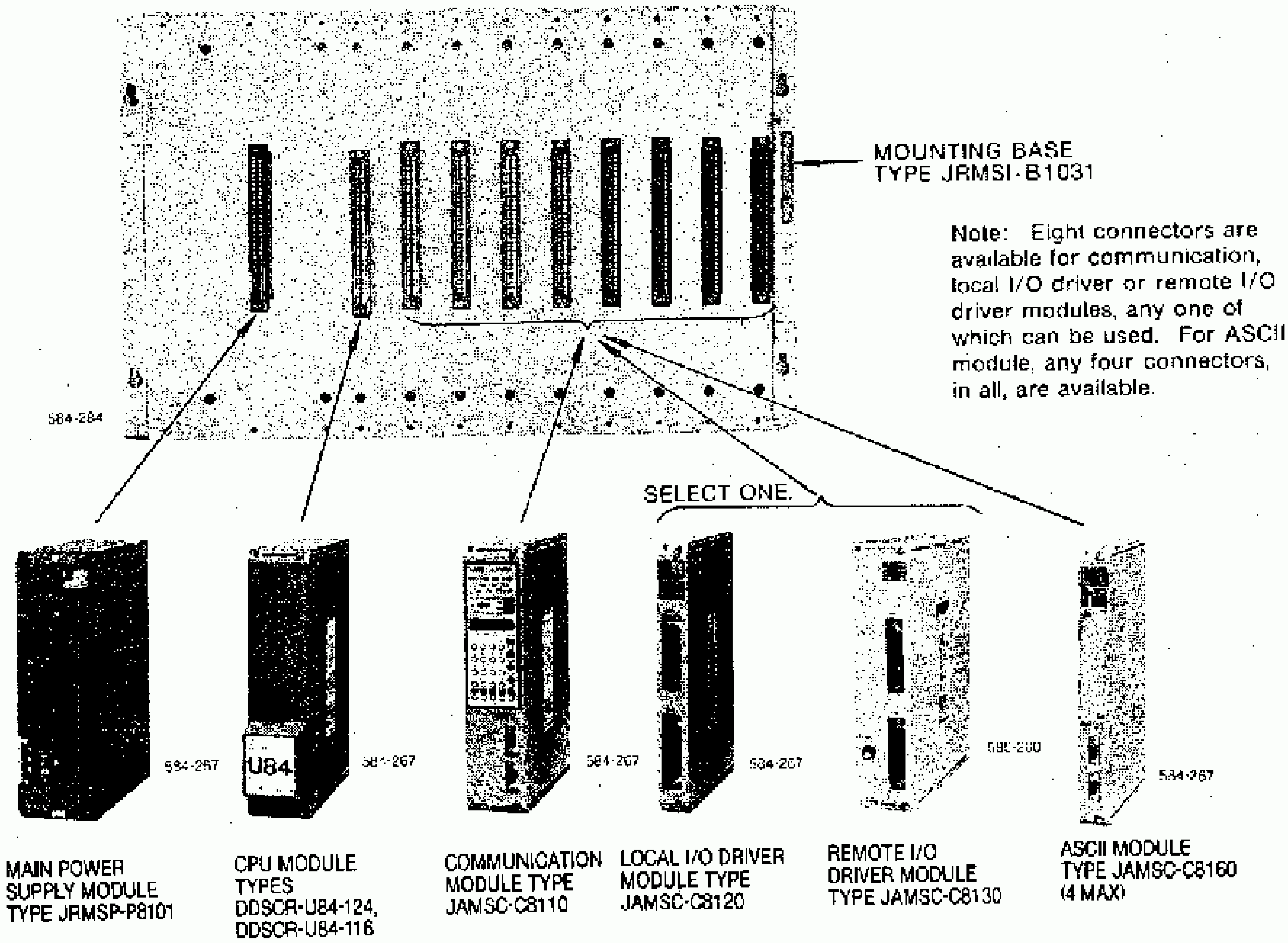


Fig. 7.3 Module Mounting on Mounting Base B1031

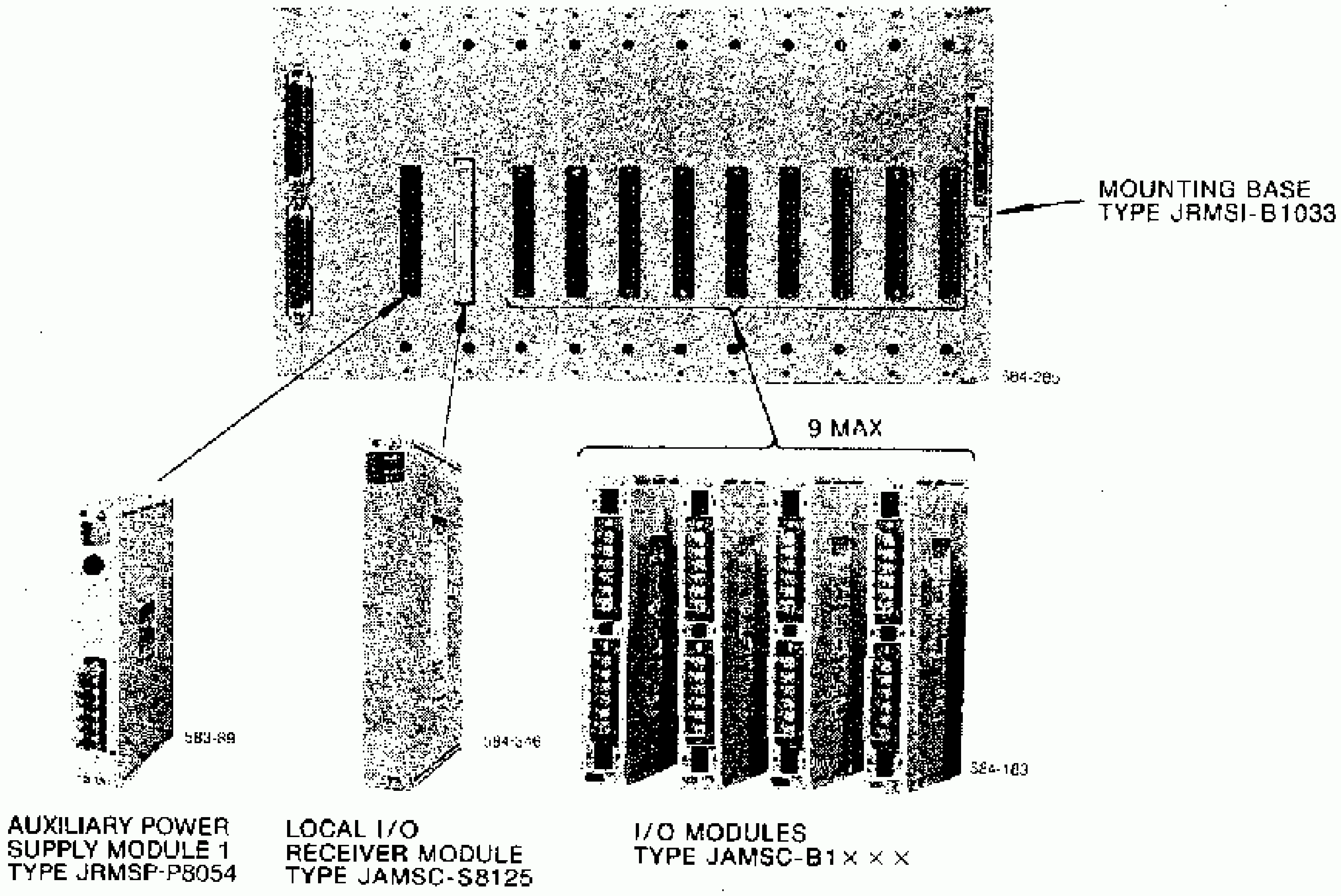


Fig. 7.4 Module Mounting on Mounting Base B1033

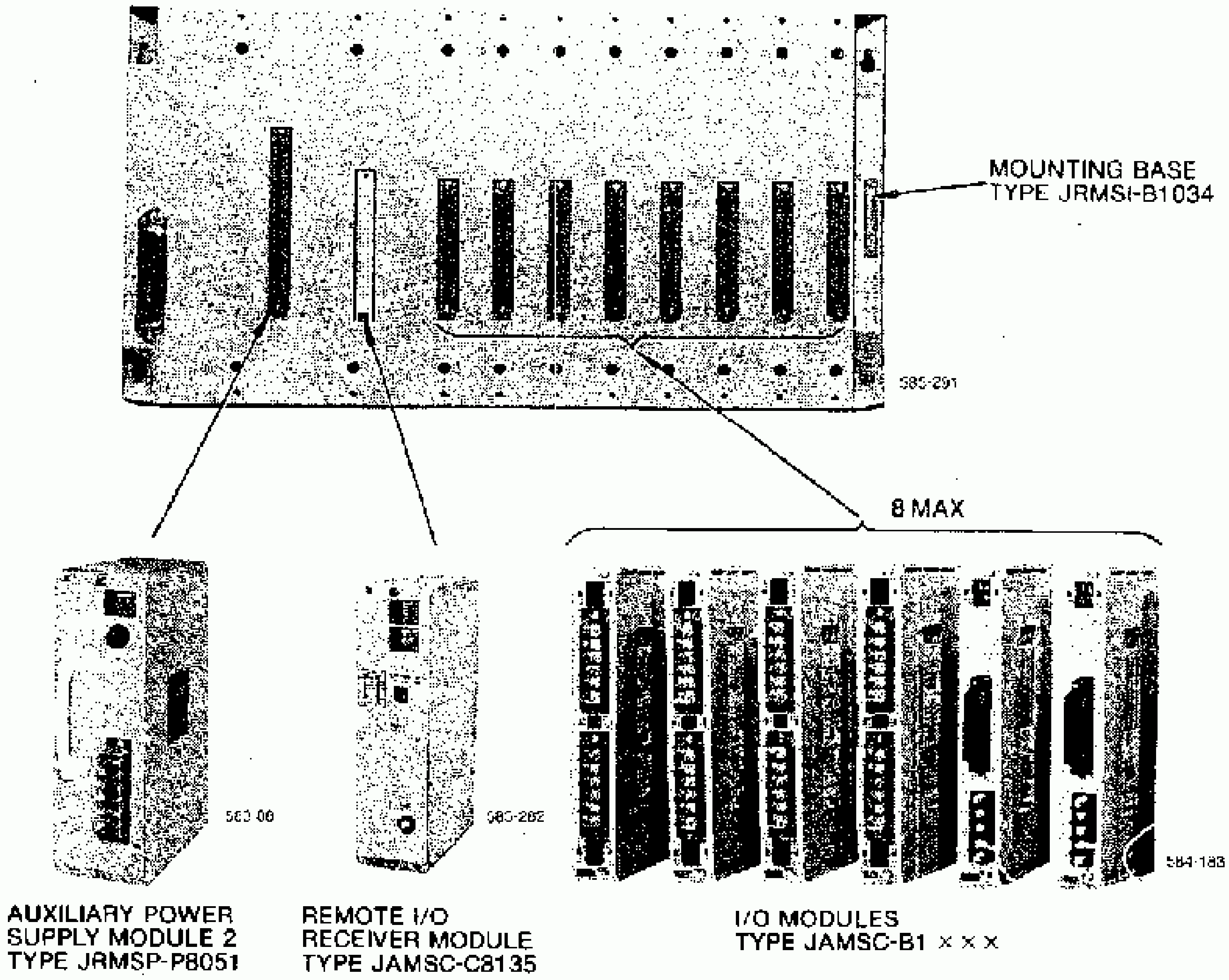


Fig. 7.5 Module Mounting on Mounting Base B1034

7.1.2 Installation of Modules (Cont'd)

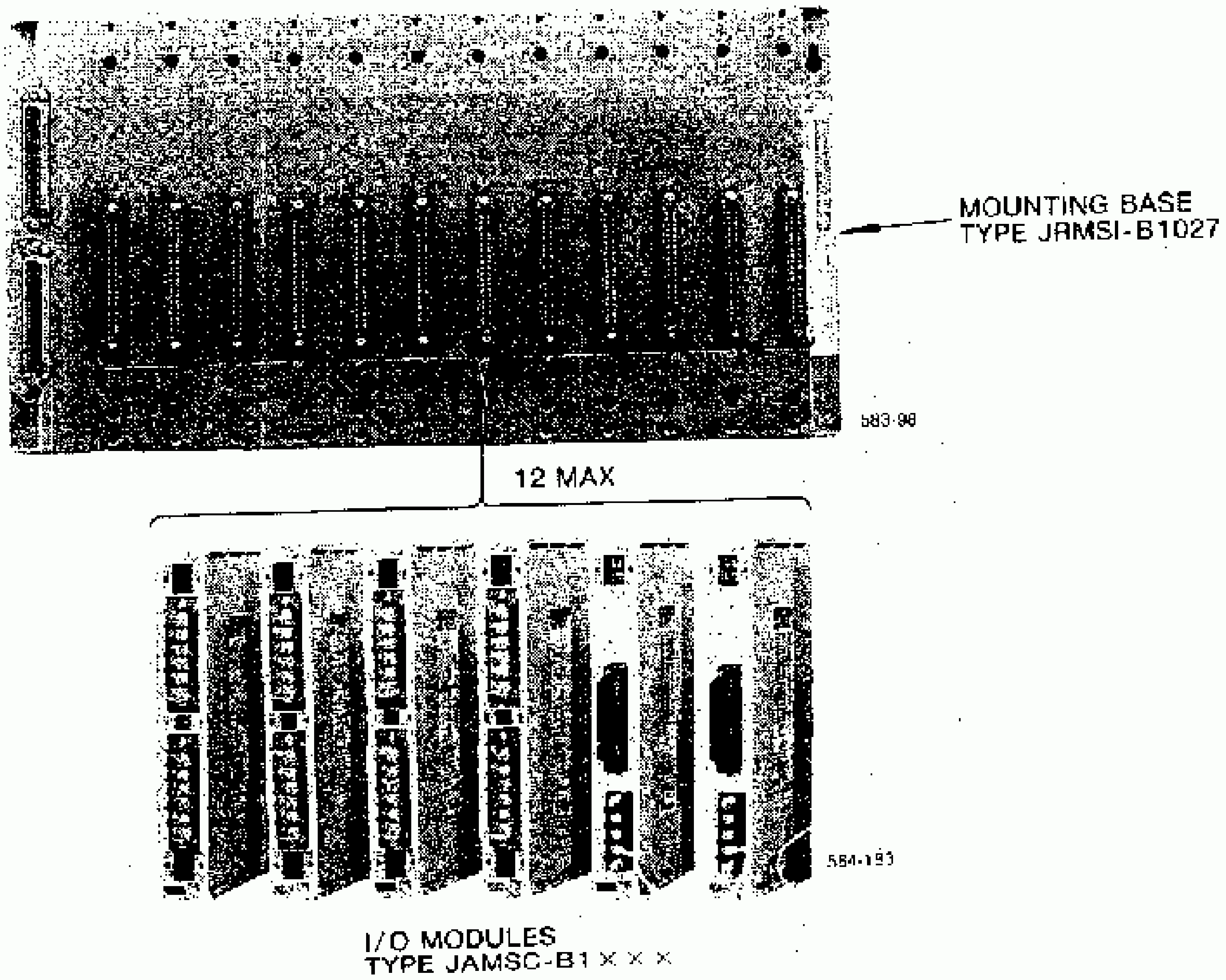


Fig. 7.6 Module Mounting on Mounting Base B1027

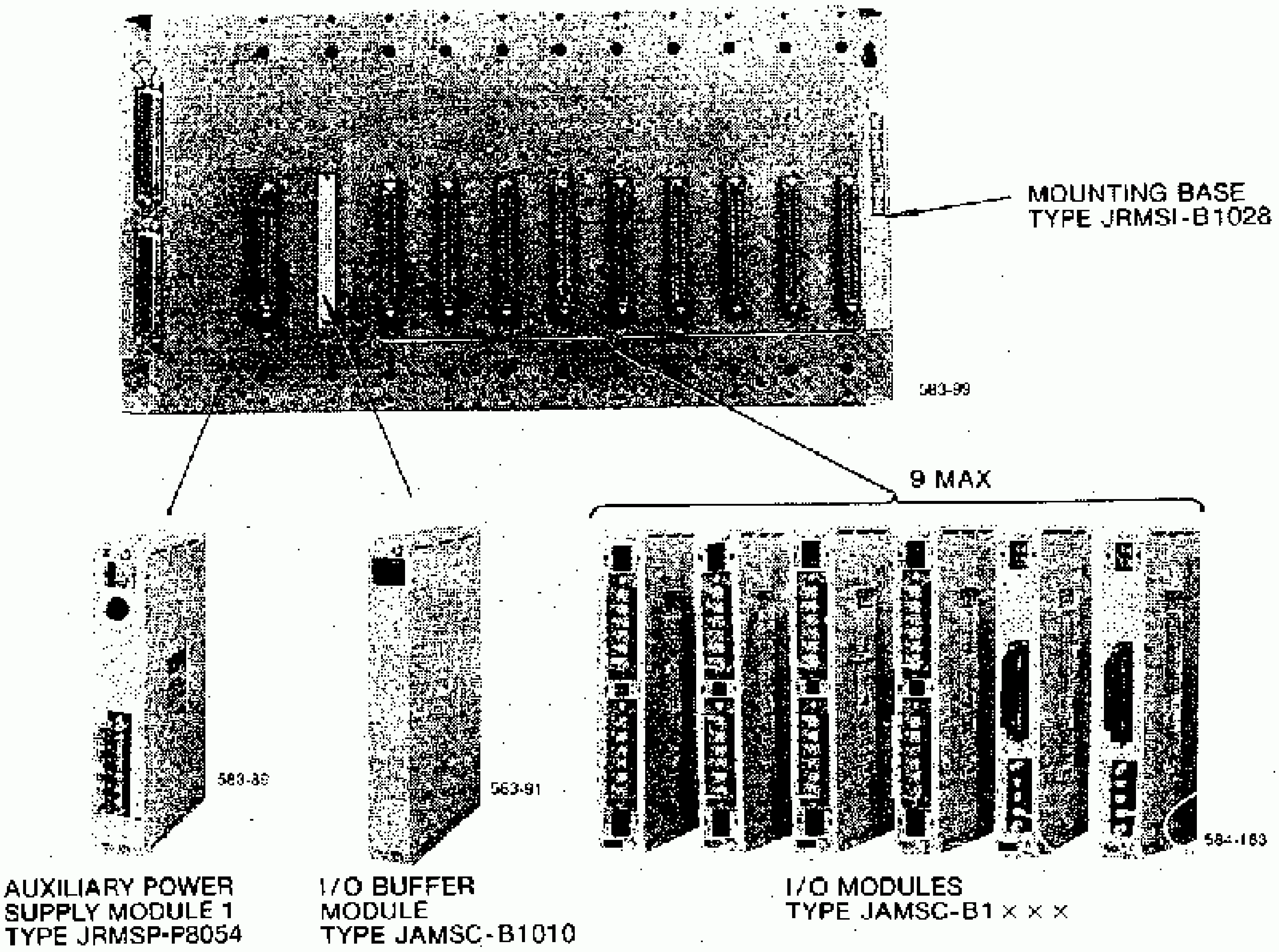


Fig. 7.7 Module Mounting on Mounting Base B1028

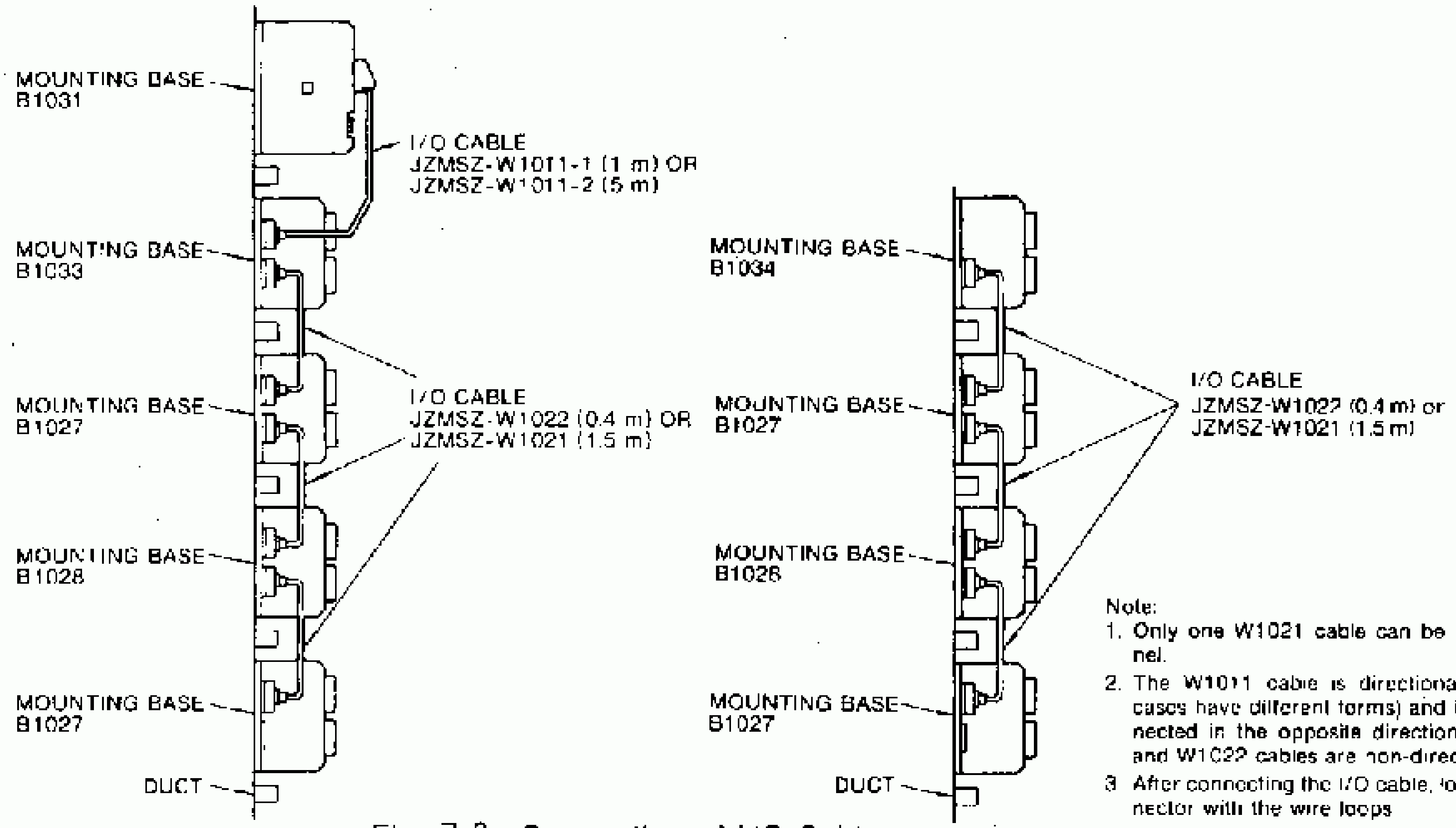
7.1.3 Connection of I/O Cables

Next, connect the mounting bases with I/O cables.

Two connectors are provided on the left side of the mounting base except B1030 and B1031 mounting bases. The connector on the top side is for input lines and that on the bottom side is for output lines.

Remove the connector cover and make connections in the order shown in Fig. 6.4. Connect the cables in the correct direction so that signal flows from output to input.

On the B1030/B1031 and B1033 mounting bases, connect the CH1 or CH2 connector of the local I/O driver installed on the B1030/B1031 mounting base with upper connector of the B1033 mounting base. Fig. 7.8 shows connection of I/O cables.



Note:
1. Only one W1021 cable can be used for a channel.
2. The W1011 cable is directional (the connector cases have different forms) and it cannot be connected in the opposite direction but the W1021 and W1022 cables are non-directional.
3. After connecting the I/O cable, lock the cable connector with the wire loops.

Fig. 7.8 Connection of I/O Cables

7.1.4 Wiring of Modules

The power supply module and I/O modules have terminal blocks and connectors for connecting the power lines and I/O signal lines.

Four types of terminal blocks and one type of connector are used. Table 7.1 shows the names of modules having terminal blocks and a connector.

Table 7.1 Terminal Blocks and Connectors for Wiring

Type	No. of Pins	Lead Size	Connecting Method	Module Type	Q'ty	Remarks
Terminal Block	12	1.25 mm²	Pressure terminal • R type: R1.25–4 • Fork type: 2–4Y, F1.25–4	B1050, B1051B, B1052, B1053 B1054, B1055, B1056, B1057 B1058, B1059C, B1073, B1075 B1080, B1082C, B1089, B1090B B1094	2	Removable from module.
				B1072B, B1074, B1081C, P8052 P8051	1	
	3			B1070, B1071	1	
	36			B1083C	1	
	3			P8101	1	Unremovable
	38		Pressure terminal • R type: R1.25–3.5 • Fork type: 2–3.5Y, F1.25–3.5	B1062, B1063	1	Removable from module.
Connector	40	0.3 mm²	Soldering	B1064, B1065, B1070, B1071	1	–
				B1060, B1061	2	

Note: Module types B1070 and B1071 require both of terminal block and connector.

(1) Wiring of Terminal Block

The transparent acrylic cover of the external wiring connector terminals on the faceplate of the I/O module should be removed when connecting the wire to the connector terminal. When the 6-pin terminal block cover is removed, follow Fig. 7.9. When the 3-pin terminal cover is removed, remove two mounting screws, one on the top and one on the bottom of the cover.

Where I/O module provided with connector terminal with non-transparent cover is used, the cover is hinged. The terminal block cover of B1083C (positioning module) is also hinged.

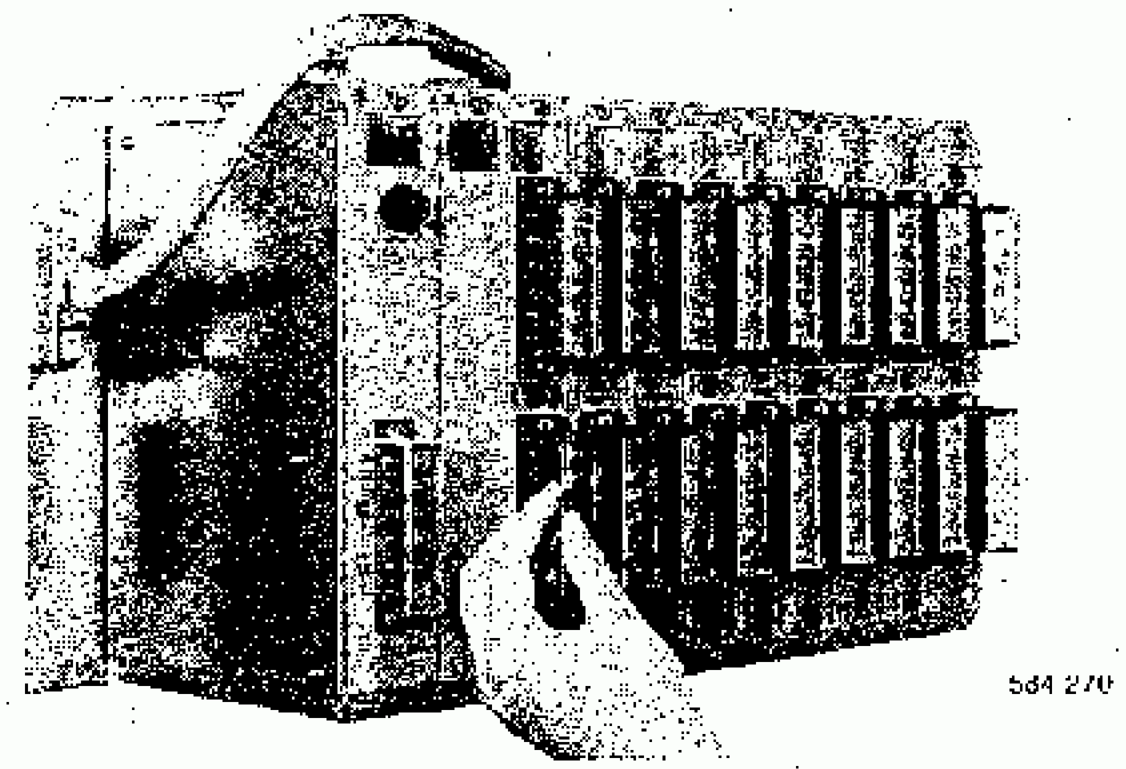


Fig. 7.9 Removing of Transparent Cover of External Wiring Connector Terminals

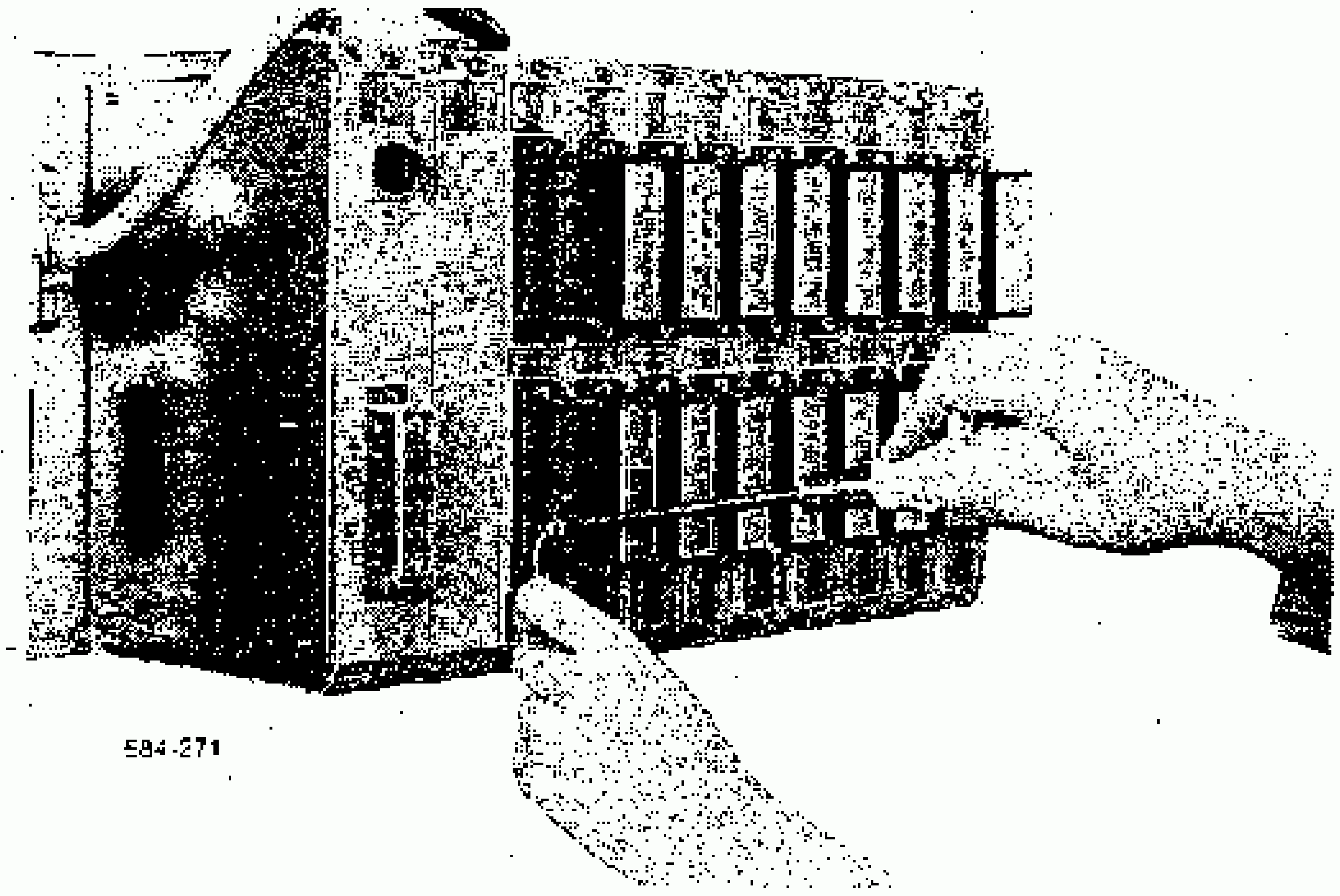


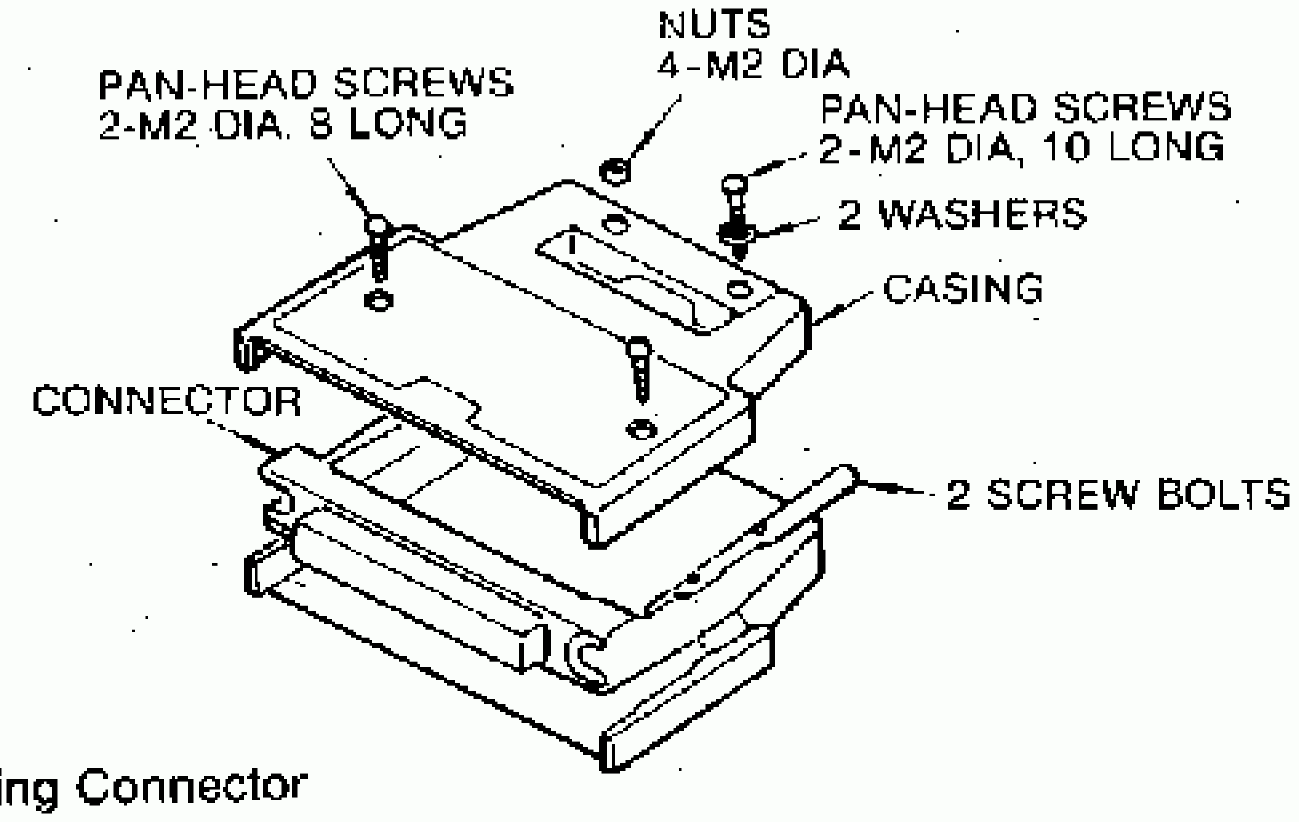
Fig. 7.10 Connecting the Wire to the Connector Terminal

(2) Wiring of Connector

The Connector is attached to the I/O module.

(a) Connector case removal

Separate the connector case from the connector by removing the four screws, as shown in Fig. 7.11.



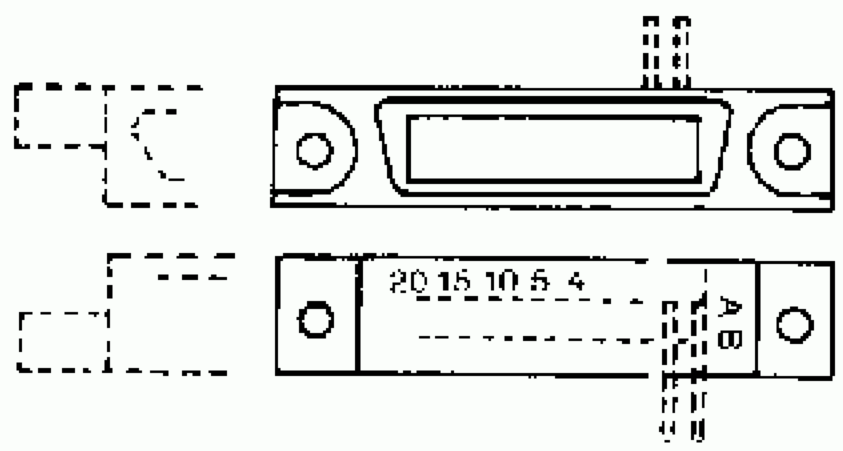
Note:
1. After connecting the wires, secure them to the case by a cable clamp, then assemble the connector casing and connector by the four screws.
2. If the number or size of wires is small, wind several turns of vinyl tape around the wire bundle to make the cable clamp tight.

Fig. 7.11 Removing Connector

(b) Lead connection

- Use wires of size 0.3 mm^2 or less.
- Put an insulating tube over the soldered portion to prevent shortcircuiting by wires, or soldering burrs.
- The terminal numbers are indicated on the wiring side of the connector as shown in Fig. 7.12.

Fig. 7.12 Indication of Terminal Numbers



7.2 MODULE REPLACEMENT

The indicator lamps of the modules, the register access panel (RAP) of the communication module, and the programming panel permit locating a defective module.

It is possible to locate and replace the defective module so quickly that system down time will be minimized.

Follow the procedures given below to replace a module.

(1) Turn off supply power.

Remove the AC power from the U84 and external supply power from the I/O modules. If I/O modules are replaced with the power ON, outputs might be erroneous.

(2) Remove the terminal block and connector. [Fig. 7.13 (a)]

It is not necessary to remove the wires separately. Rather, remove the terminal block and connector. Remove the two screws, top and bottom.

Remove the power lines, wired to the P8101 (main power supply module), B1070 (register output module), and B1071 (register input module), one by one.

(3) Loosen the screws fastening the module. [Fig. 7.13 (b)]

Loosen the two screws, top and bottom, fastening the module to the mounting base, until they are released from the base.

(4) Remove the module. [Fig. 7.13 (c)]

Holding the top and bottom of the module, pull it out toward you.

(5) Install a new module.

Fit the guide posts of the module into the guide holes of the mounting base and push the module in so that the guide posts slip into the holes completely. Then fasten the module to the mounting base with the screws provided with the module.

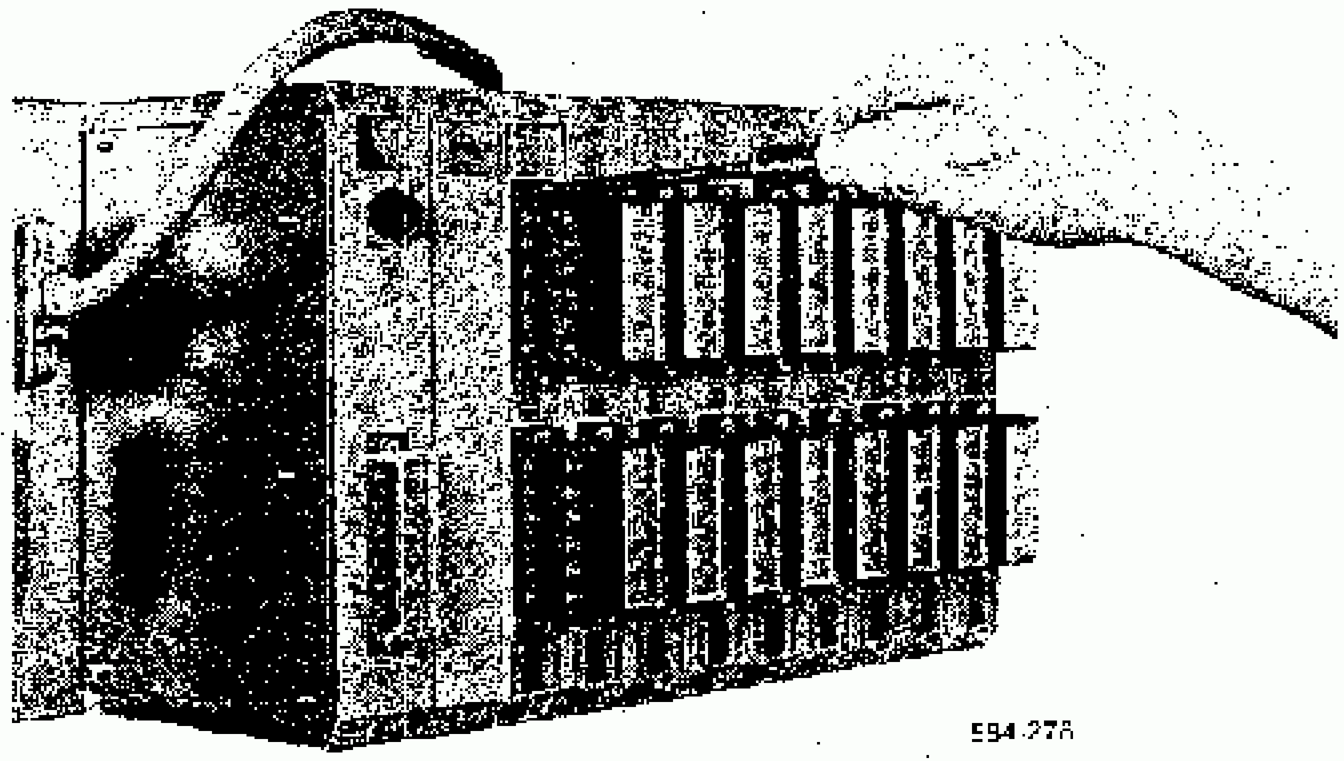
(6) Install the terminal block and connector.

Replace the terminal block, connector, and wires removed in step (2).

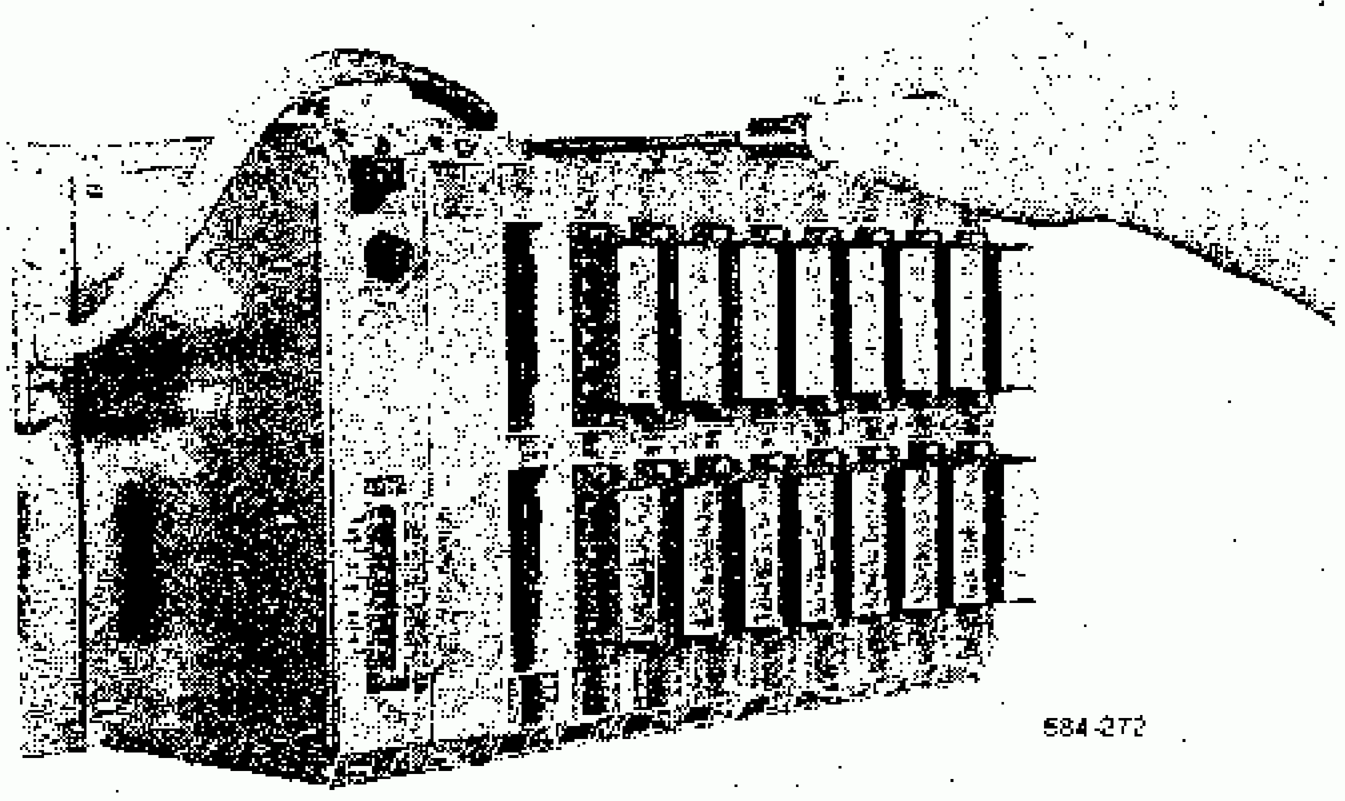
(7) Turn on power.

Make sure that the module and wiring are correct, before turning on power.

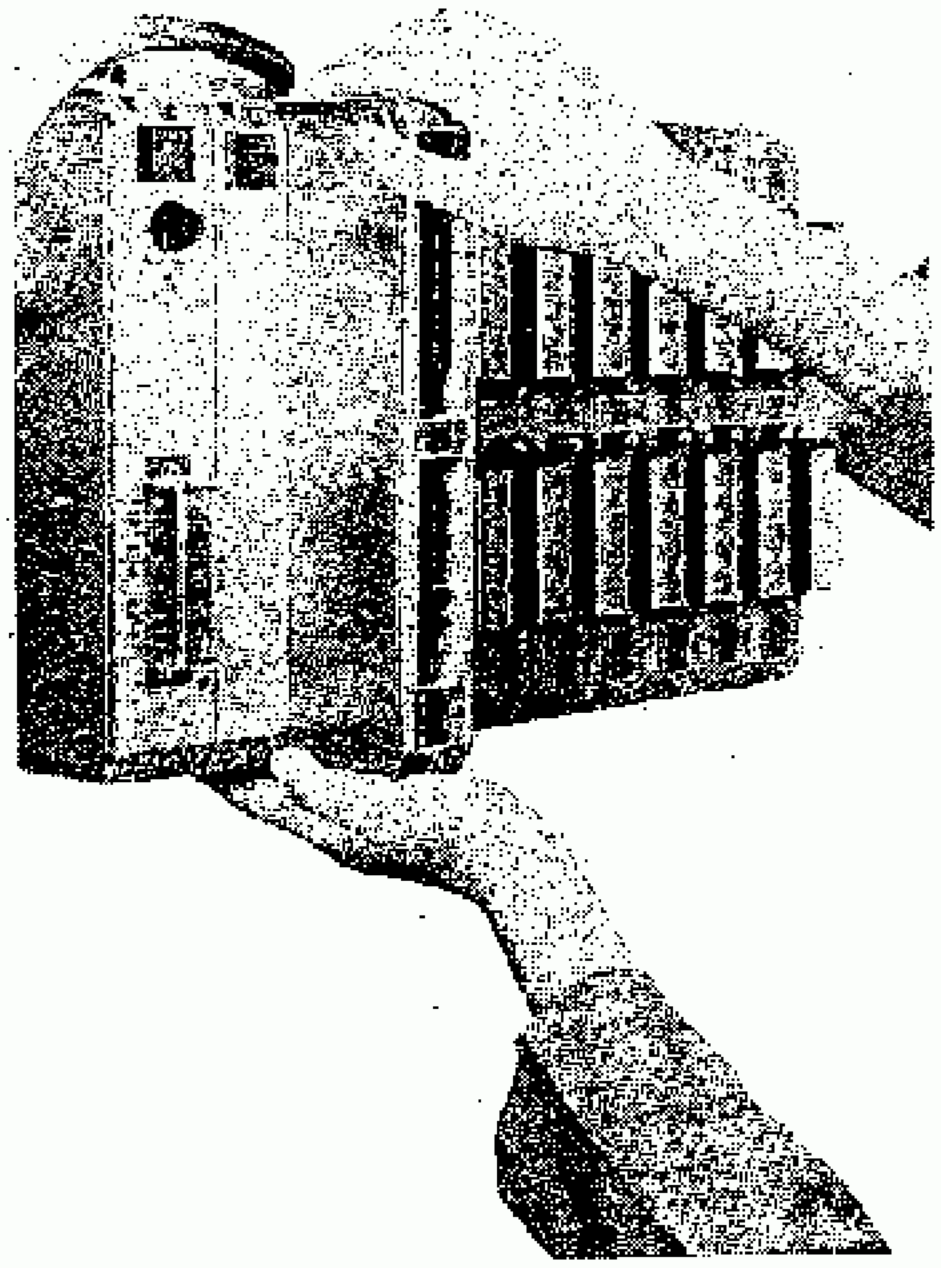
(7) Turn on power. (Cont'd)



(a) Removing Connector Terminal Block



(b) Loosening Module Mounting Screws



(c) Pulling out Module

Fig. 7.13 Module Replacement

7.3 BATTERY REPLACEMENT

Back-up power for the U84 memory is provided by battery. It is recommended to replace the battery every two years.

NOTE

Battery service life varies with the environmental conditions (temperature and humidity) and working time (AC power failure time).

(1) Battery Specifications

Table 7.2 gives the specifications of the battery used in the U84.

Table 7.2 Battery Specifications

Item	Specifications
Name	Lithium battery
Type	BR-CT2N (with wiring tab)
Manufacturer	Matsushita Battery Industry Co., Ltd.
Nominal Voltage	3 V
Nominal Capacity	5000 m Ah
Ambient Temperature	0°C to +55°C
Storage Temperature	−20°C to +45°C
Life	• Warranty: 5 years at 25°C • Actual protective period for memory: 1 year at 25°C
Approx Weight	50 g

Note: When the battery in the Table above is required, contact Yaskawa representative.

(2) Battery Replacement Interval

The battery will last for a maximum of five years or until the total time of AC power failure reaches a year. If the ERROR (battery failure) lamp of the CPU module lights, replace the battery within one month.

Approximate standards for battery replacement are given below.

Depending on U84 average daily operating hours

- 12 hours/day: replace every two years.
- 16 hours/day: replace every three years.
- 20 hours/day: replace every four years.

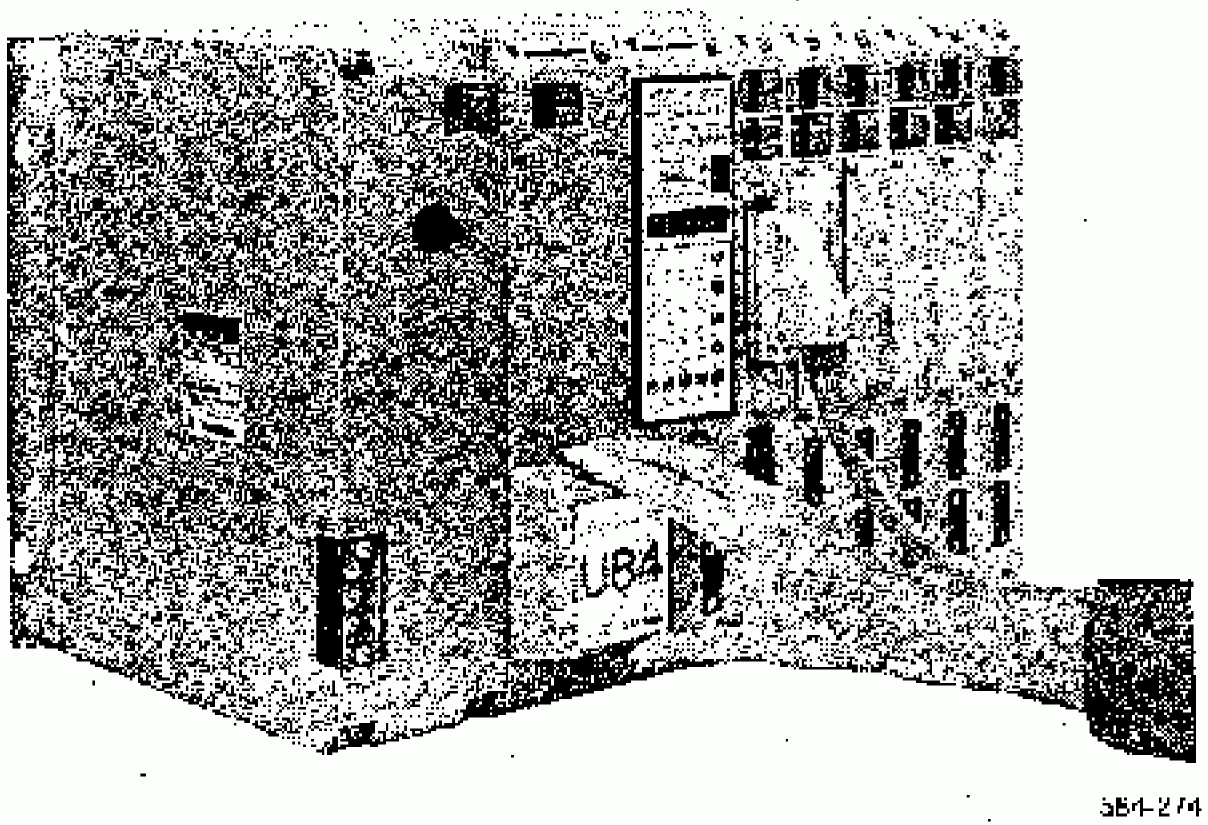
(3) Battery Replacement Procedures

Be sure to replace the battery with the CPU module connected to the base and receiving AC power to the main power supply module. If the battery is removed with no AC power supplied, the memory contents will be destroyed. But if the battery is removed with no AC power supplied for a short time (within 30 minutes), the memory contents will not be destroyed.

Replace the battery using the following procedures:

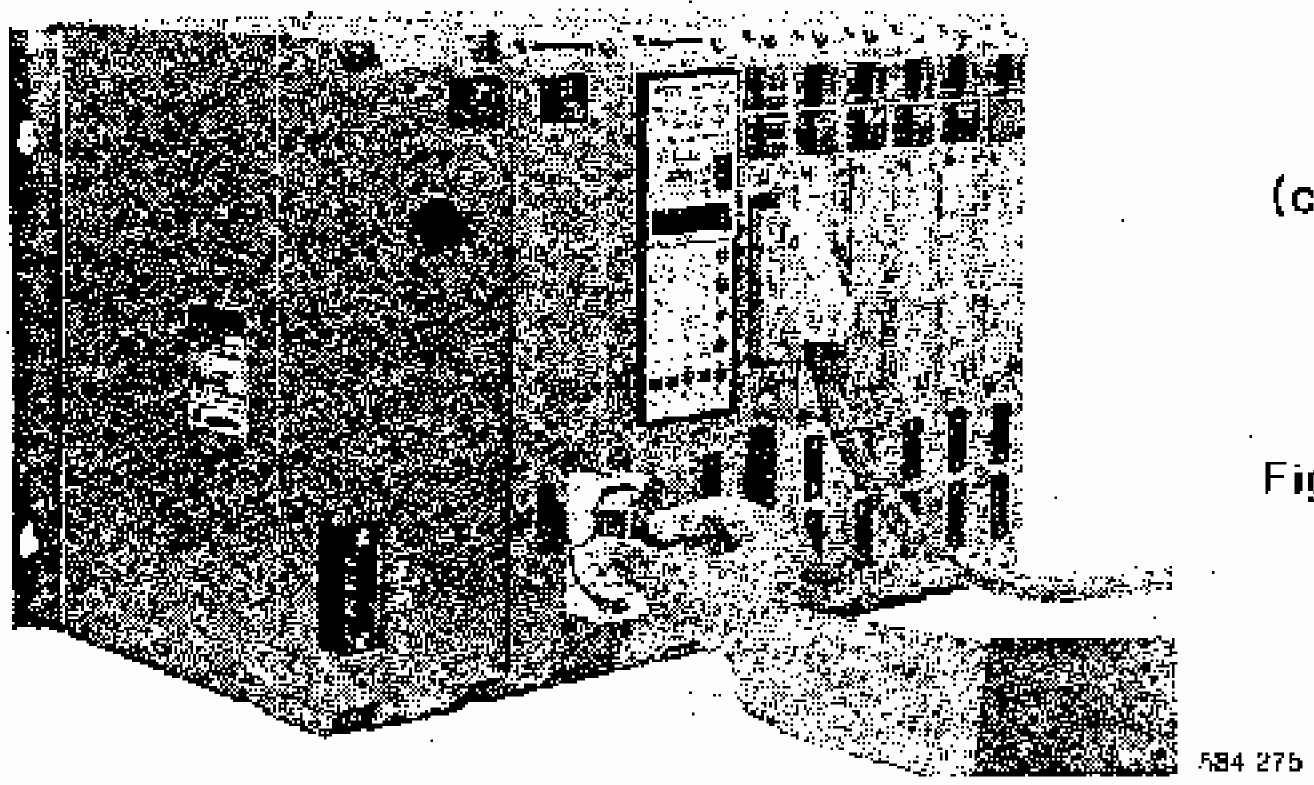
1. Preheat the soldering iron.
2. Confirm that the POWER lamp of the main power supply module is lit (AC power ON).
3. Remove the battery cover. See Fig. 7.14(a).
4. Take out the battery from the battery case, then separate the connector attached at the ends of the leads from the CPU module. See Figs. 7.14(b) and (c).
5. Remove the leads from the soldering tabs on the battery, using the preheated soldering iron.
6. Solder the removed leads to the soldering tabs on the new battery, using care to observe the polarity. (Lead colors Red:Positive; Black:Negative)
7. Place the new battery in the battery case and insert the connector attached at the ends of the leads into the CPU module connector.
8. Confirm that the ERROR (battery trouble) lamp of the CPU module goes off.
9. Install the battery cover.
10. Turn off the power supply for the soldering iron.
    This completes the battery replacement.

(3) Battery Replacement Procedures (Cont'd)



(a) Removing Battery Cover

(c) Separating Connector from CPU Module



(b) Taking out Battery from Battery Case

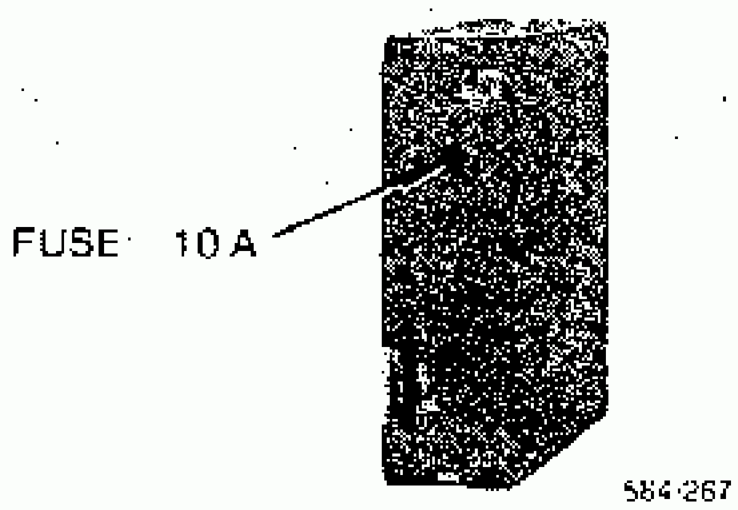
Fig. 7.14 Battery Replacement

7.4 FUSE REPLACEMENT

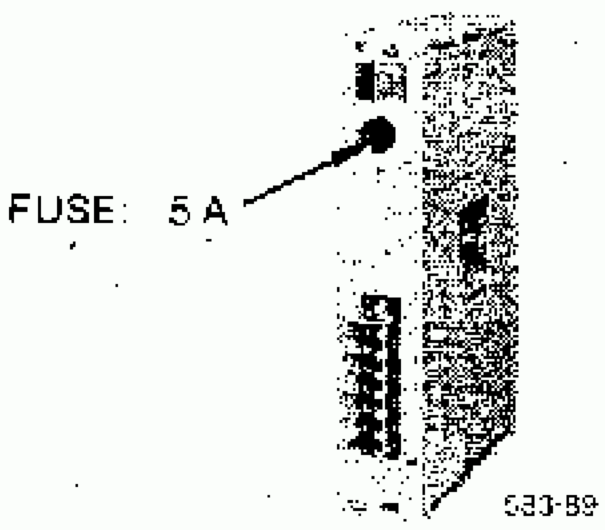
7.4.1 Fuse Replacement of Power Supply Module

The main power supply module and the auxiliary power supply module contain one fuse in the fuse holder respectively. The fuse is a commercially available glass tube fuse. The fuse capacity is 10A for main power supply module and 5A for auxiliary power supply module.

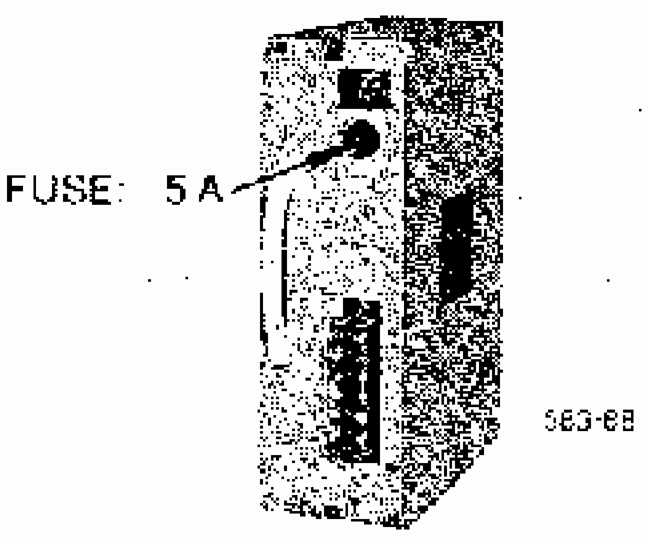
When the fuse blows, "POWER" indicator is turned off. Replace the fuse with AC power supply turned off.



Main Power Supply Module



Auxiliary Power Supply Module 1



Auxiliary Power Supply Module 2

Fig. 7.15 Fuses of Power Supply Modules

7.4.2 Fuse Replacement of Output Module

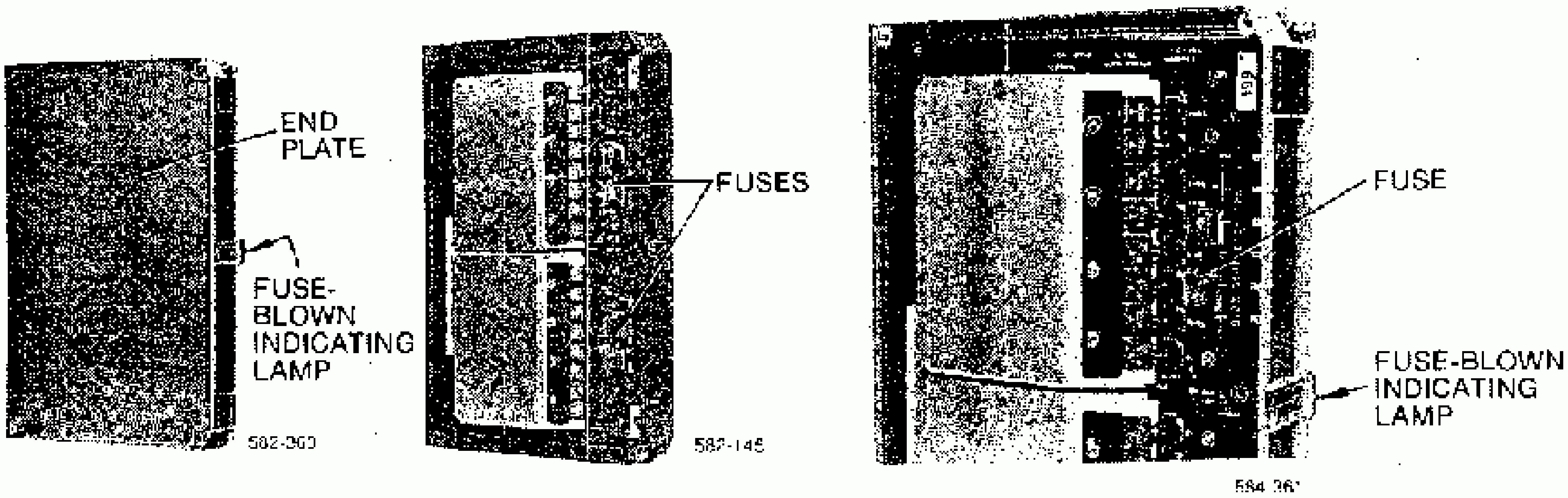
Four types of output modules have one output fuse per eight circuits (two per module). The types of output modules with two fuses are given below:

- B1050: 100VAC, 16-point output
- B1054: 200VAC, 16-point output
- B1056: 48 VDC, 16-point output
- B1058: 24 VDC, 16-point output

The fuse is a commercially available glass tube fuse with a 5A capacity. If a fuse blows, Fuse-blown Indicating Lamp lights. Replace the fuse after removing the output module from mounting base and the end plate at the left side of the module. (Fig. 7.16)

NOTE

The output fuse is used for preventing any trouble caused by a shortcircuit, but not for protecting the output elements of the module.



(a) Removed Output Module

(b) With Endplate Removed

(c) Location of New Fuse to be Replaced

Fig. 7.16 Replacing Output Module Fuse

7.5 USE OF REGISTER ACCESS PANEL

The register access panel (RAP) of the communication module displays the on/off states of coils and input relays, disabling operation (DISABLE ON/OFF), the contents of registers and holding registers.
It also permits setting or modifying the contents of registers.

Fig. 7.17 shows the register access panel. Immediately after supply of power starts or the U84 starts to run it indicats as follows.

SC U84

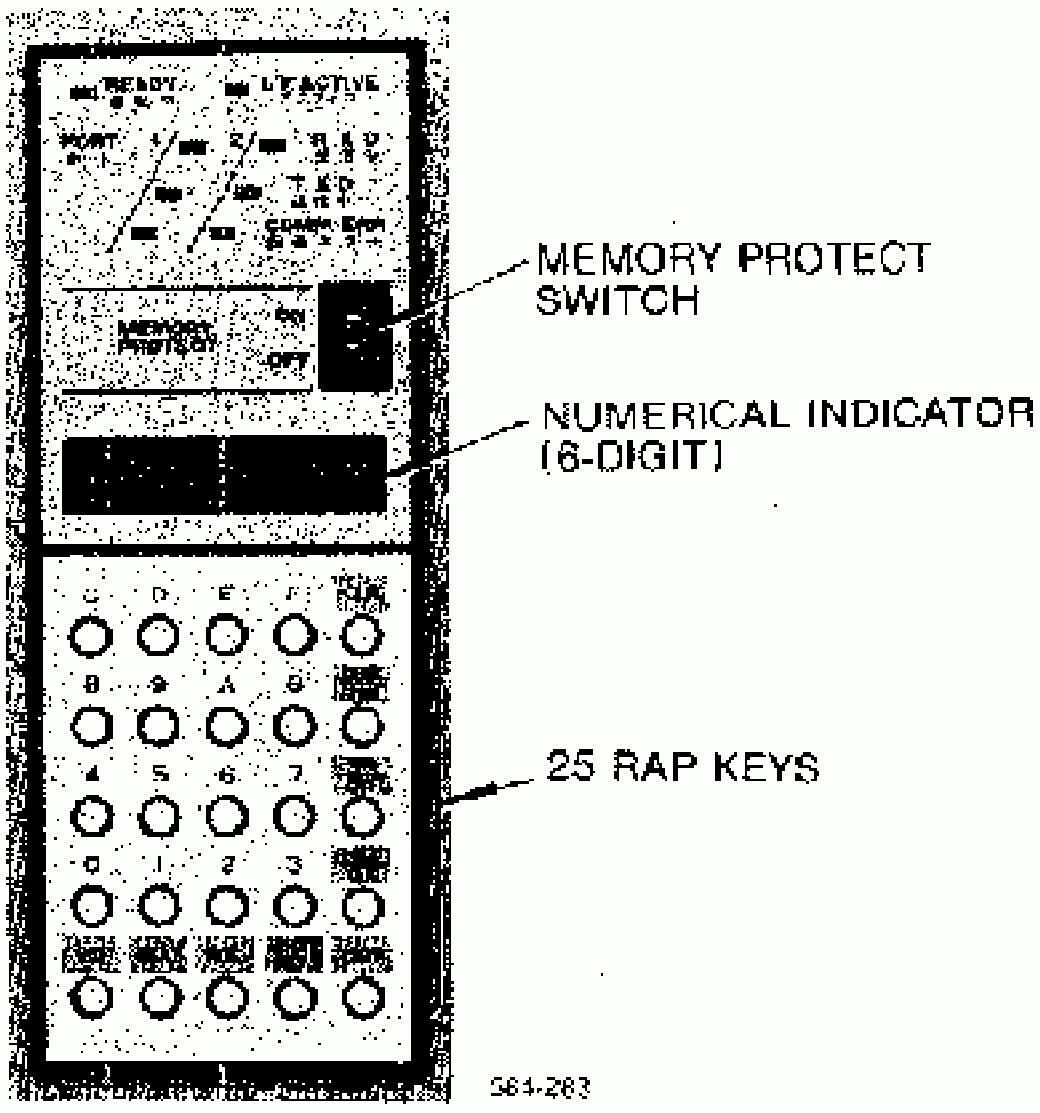


Fig. 7.17 Register Access Panel

7.5.1 RAP Keys

Table 7.3 RAP Keys

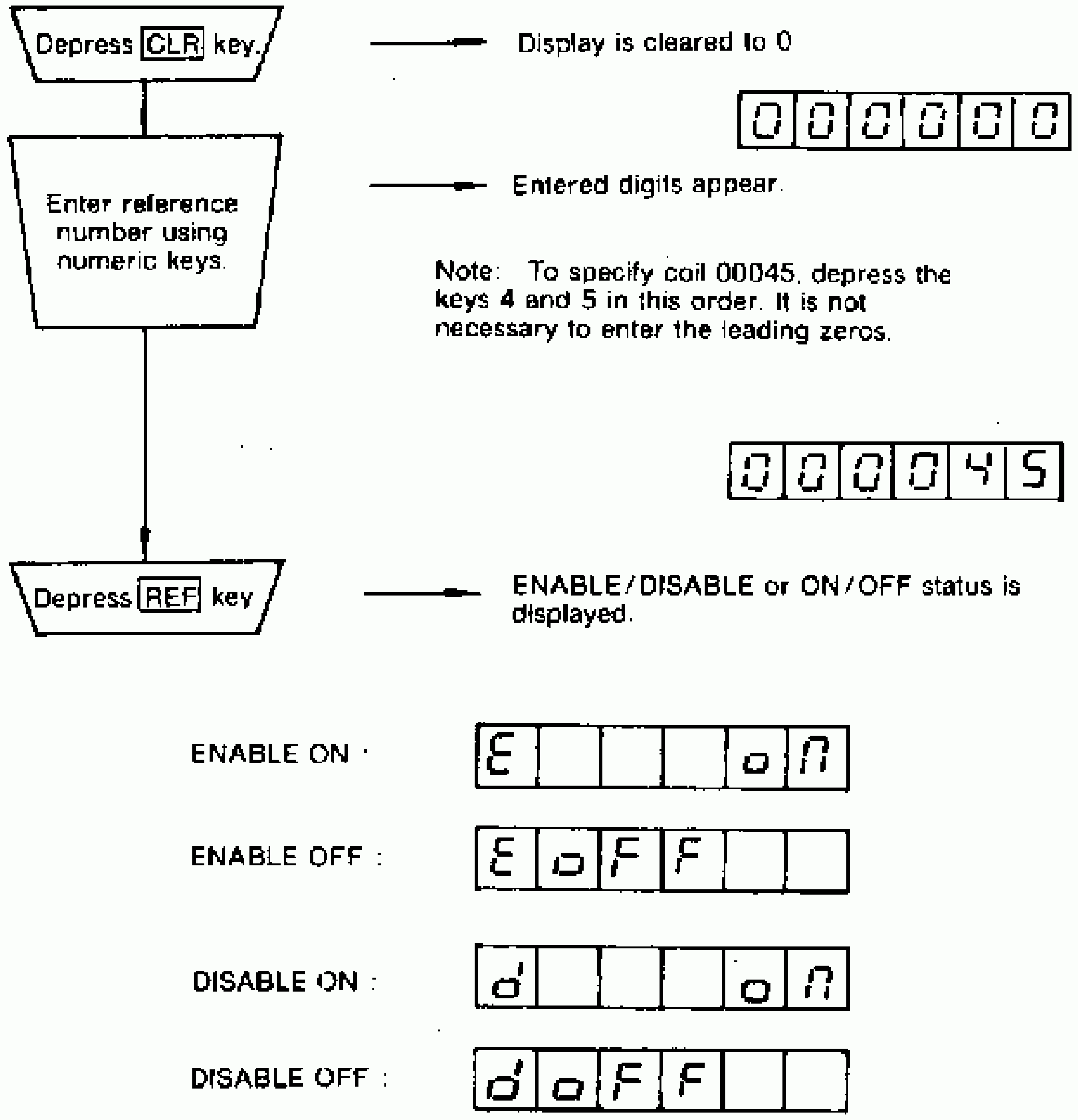
Key	Description
[0] to [F]	Numeric keys. Use to enter numbers and numeric data. [A] to [F] are hex keys.
[CLR]	Push to clear the display. Use to reset error indication or cancel key entry.
[DSBL/ENBL]	Use to disable or enable coil or input relay. Push to disable and push again to enable (the functions alternate when pressed). Effective when memory protect switch is OFF.
[ON/OFF]	Turns on/off disabled coil or input relay. ON/OFF functions alternate when pressed. Effective when memory protect switch is OFF.
[HEX/DEC]	Selects hexadecimal (hex) or decimal for input data and displayed data. Hex and decimal alternate when pressed.
[ENTR]	Push to enter input data to holding register or set port parameter. Effective when memory protect is OFF.
[WHAT REF?]	Push to find reference number or address currently monitored.
[NEXT]	Push to monitor contents of reference number or address next to those currently monitored.
[PREV]	Push to monitor contents of reference number or address preceding those currently monitored.
[REF]	Push to monitor contents of reference number or address set.

7.5.2 Operation

(1) Displaying ON/OFF Status of Input Relay 1×××× or Coil 0××××

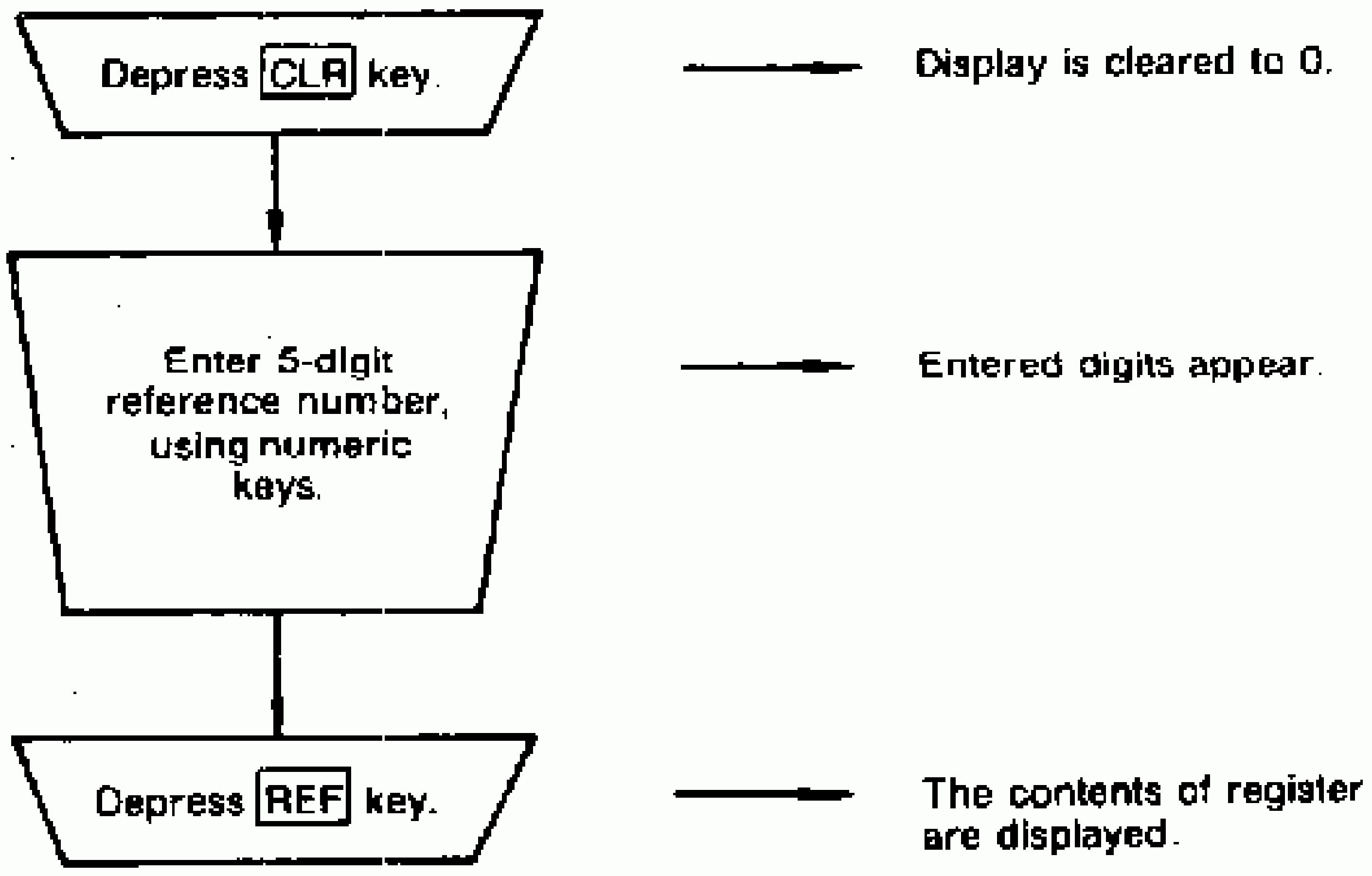
Set reference number, and the status for example, of the specified coil will be displayed enabled (observing ladder logic) or disabled (turned on or off independently of ladder logic).

- The status display is refreshed in every scanning cycle.
- If you push [NEXT] ([PREV]) at this time, the reference number is incremented (decremented) by one and the status of the input relay or coil identified by the updated reference number appears.



(2) Displaying Contents of Input Register 3 × × × × or Holding Register 4 × × × ×

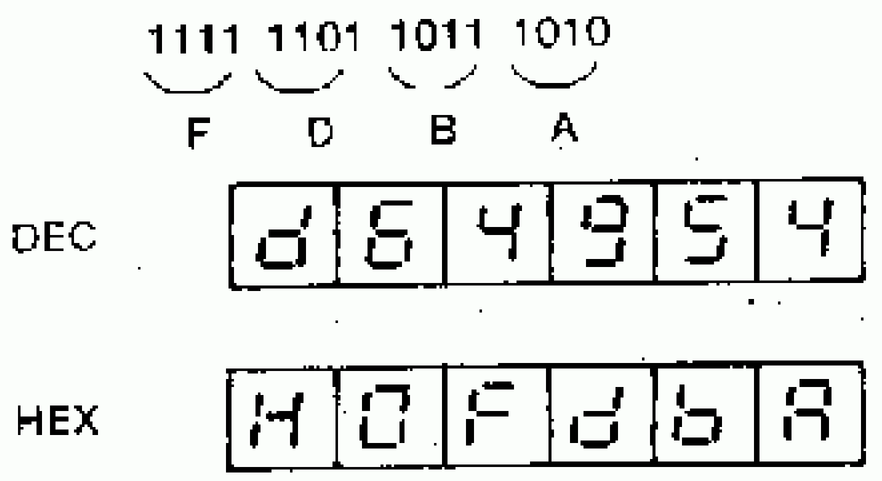
Set a reference number, and the contents of the register will be displayed. It is possible to select decimal or hex for the displayed data.



- The data will be refreshed in every scanning cycle.
- The reference number is incremented (decremented) by one every time you push NEXT (PREV).
- Display of the data changed over in decimal form and hex form slternately as you push HEX/DEC . The decimal mode is selected first.

(2) Displaying Contents of Input Register 3×××× or Holding Register 4×××× (Cont'd)

Example: The decimal number 64954 is FDBA in hexadecimal form.



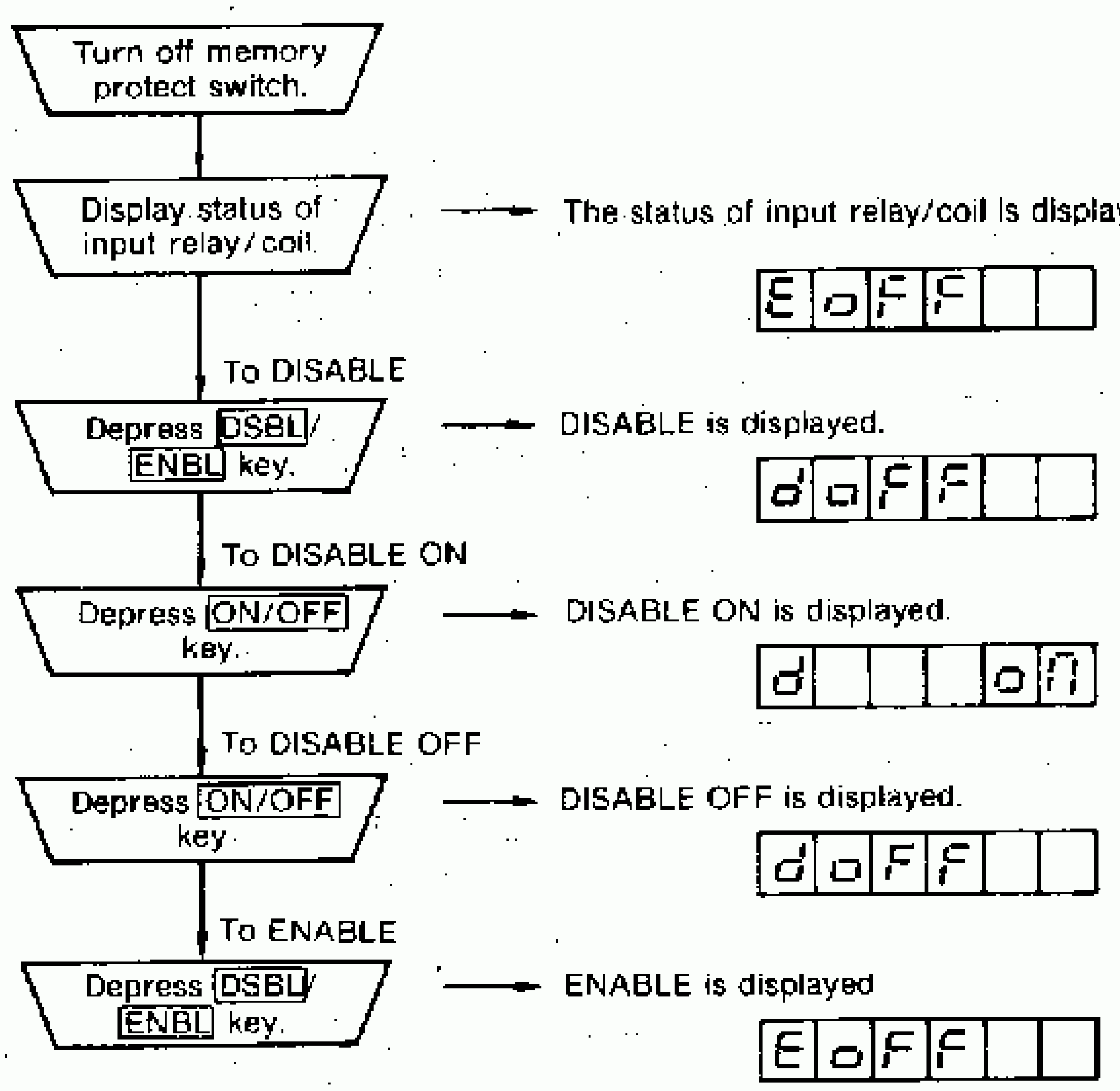
Note: The maximum number the 16-bit register can hold is 65535 ($= 2^{16} - 1$).

(3) DISABL/ENABL of Input Relay or Coil

It is possible to DISABLE or ENABLE the input relay or coil whose status is read by the method of (1).

As soon as you disable an input relay or coil, the input relay becomes independent of the actual signal and the coil of the ladder circuit, holding the ON or OFF status it has kept so far.

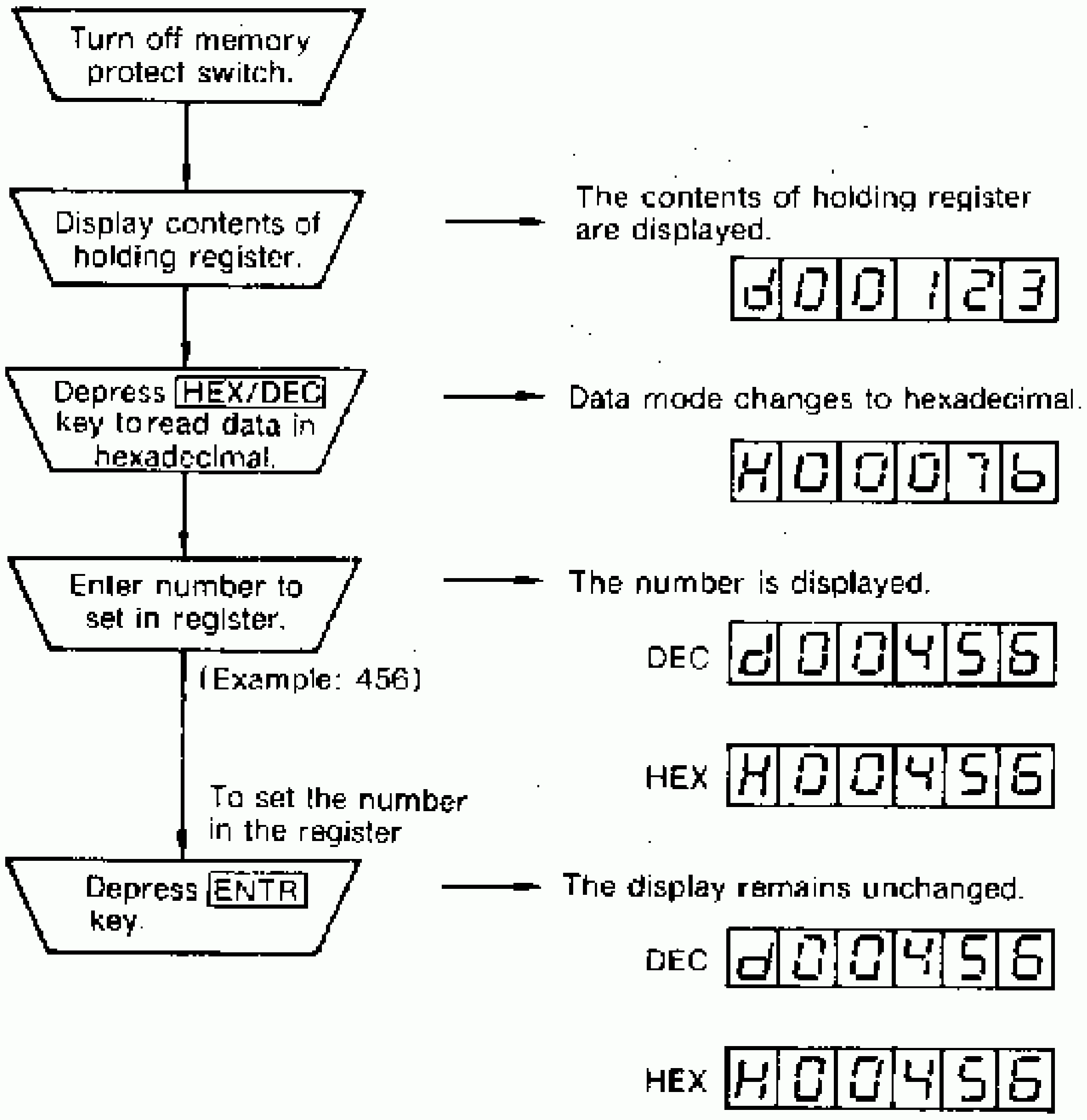
The disabled input realy and coil can be turned on and off unconditionally. When enabled, the input relay is activated according to the actual signal and the coil to the logic of the ladder circuit.



- [DSBL/ENBL] and [ON/OFF] change mode alternately when pushed.
- The reference number is incremented (decremented) by one every time you push [NEXT] ([PREV]).

(4) Setting or Altering of Holding Register

It is possible to set a value in the holding register whose contents are read by the method of (1).



- The range of value which can be set in a holding register is 0-65535 (decimal) or 0-FFFF (hex).
- [HEX/DEC] changes mode alternately when pushed.
- The reference number is incremented (decremented) by one every time you push [NEXT] ([PREV]).

(5) Displaying of Communication Parameters

It is possible to display a communication parameter between the communication module and a device (the programming panel, etc.) connected to a port by entering a special number which is described below. 6 and 000 are fixed.

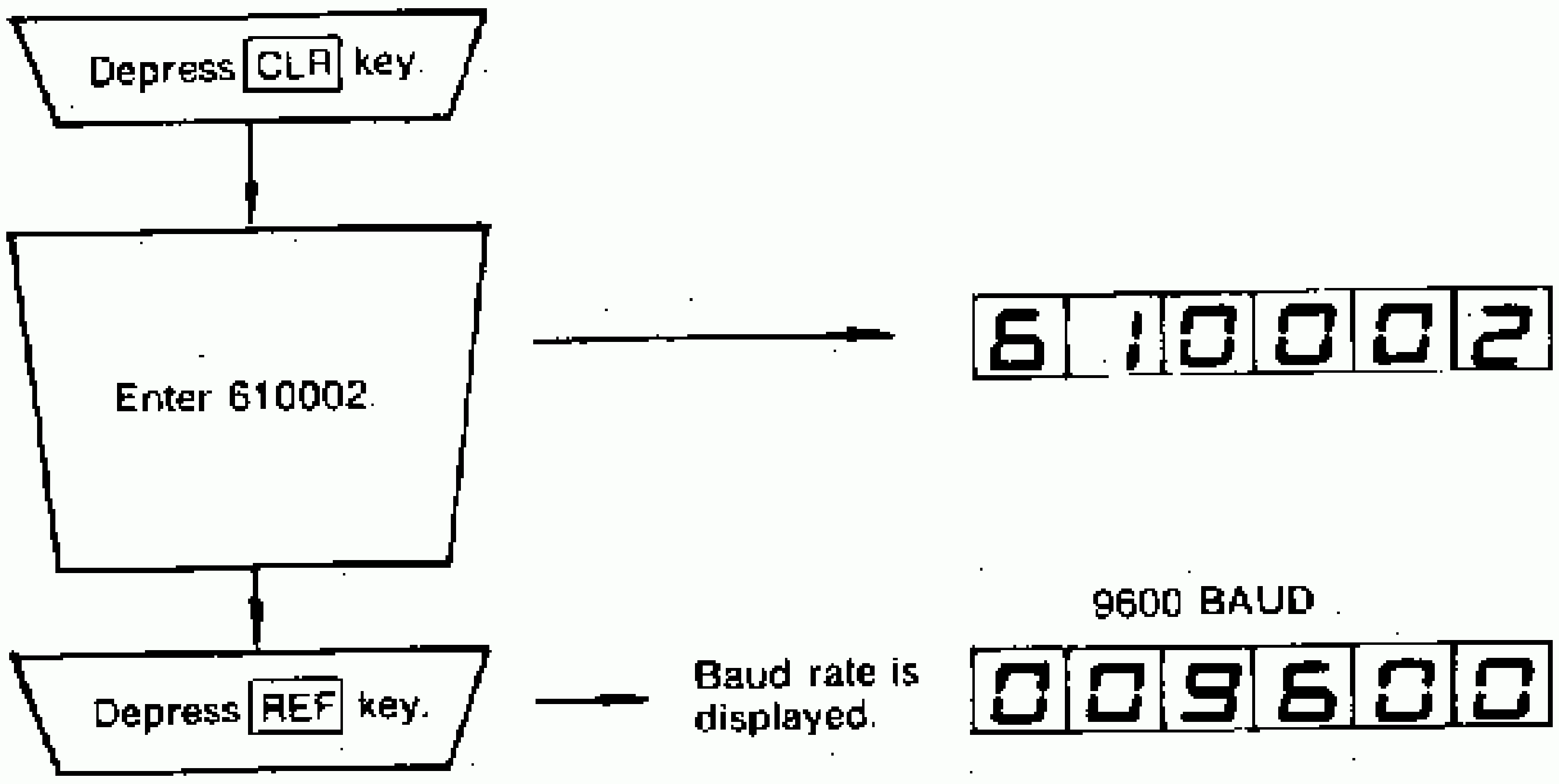
6X000Y

Parameter Y = 1 Device address
2 Baud rate
3 Use of parity
4 Parity when used
5 Number of stop bits
6 Communication mode
7 Preset value of port delay timer

Port Number
X = 1 Port 1
X = 2 Port 2
X = 3, X = 4 For future expansion

(5) Displaying of Communication Parameters (Cont'd)

Operate as follows to look at the baud rate of the device connected to port 1, for example.



Communication parameters associated with Y are as follows.

Y = 1 Device address: 1 to 247

Y = 2 Baud rate: 150, 300, 600, 1200, 2400, 4800, 9600, or 19200

Y = 3 ENABLE ... parity check performed

DISABLE ... parity check not performed

Y = 4 EVEN ... even parity

ODD .... odd parity

Y = 5 Number of stop bits: 1 or 2

Y = 6 Communication mode

bbbbbb .. RTU

ASCII ... ASCII

Y = 7 Preset value of port delay timer (U84 delays response by this value): 0 to 255 (correspond to 0 to 2550 msec)

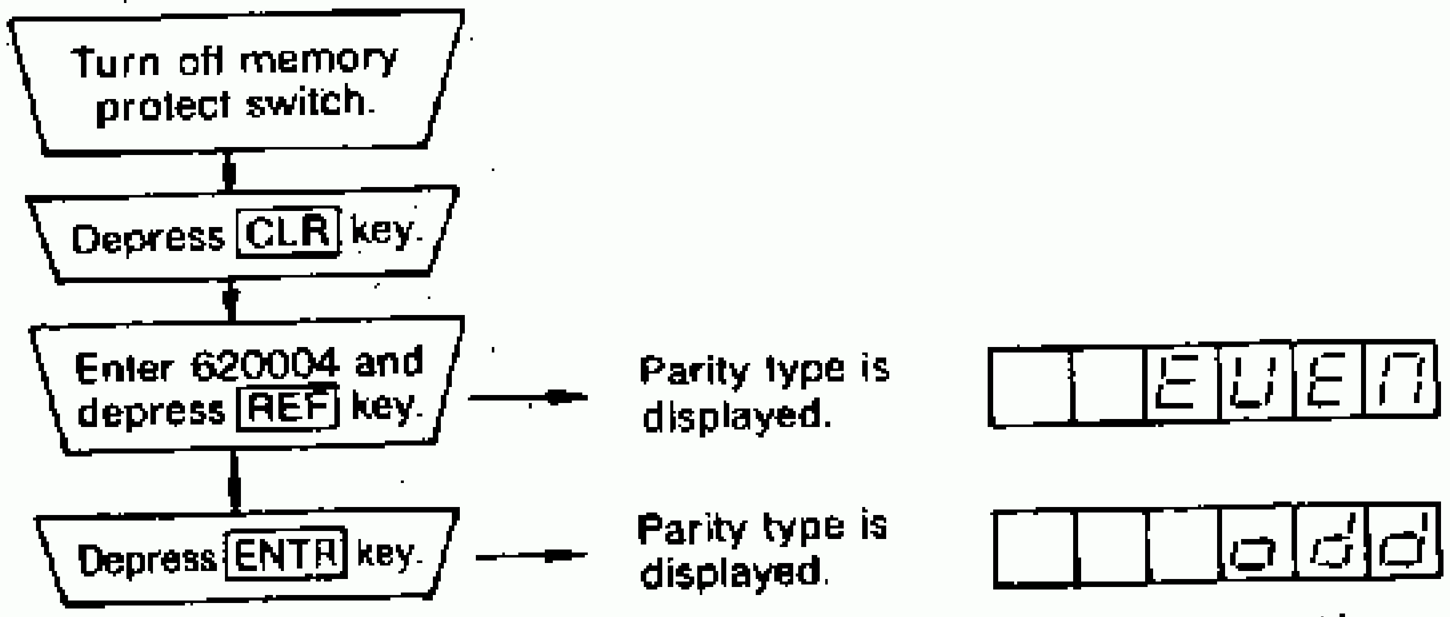
- The value of Y is incremented (decremented) by one every time you push [NEXT] ([PREV]).

(6) Setting or Altering of Communication Parameters

To set a device address or baud rate, enter the number to be set and push [ENTR]. Then the number is set.

In other cases, push [ENTR] only. Then one of two choices changes to the other.

To change even parity of port 2 to odd, for example;



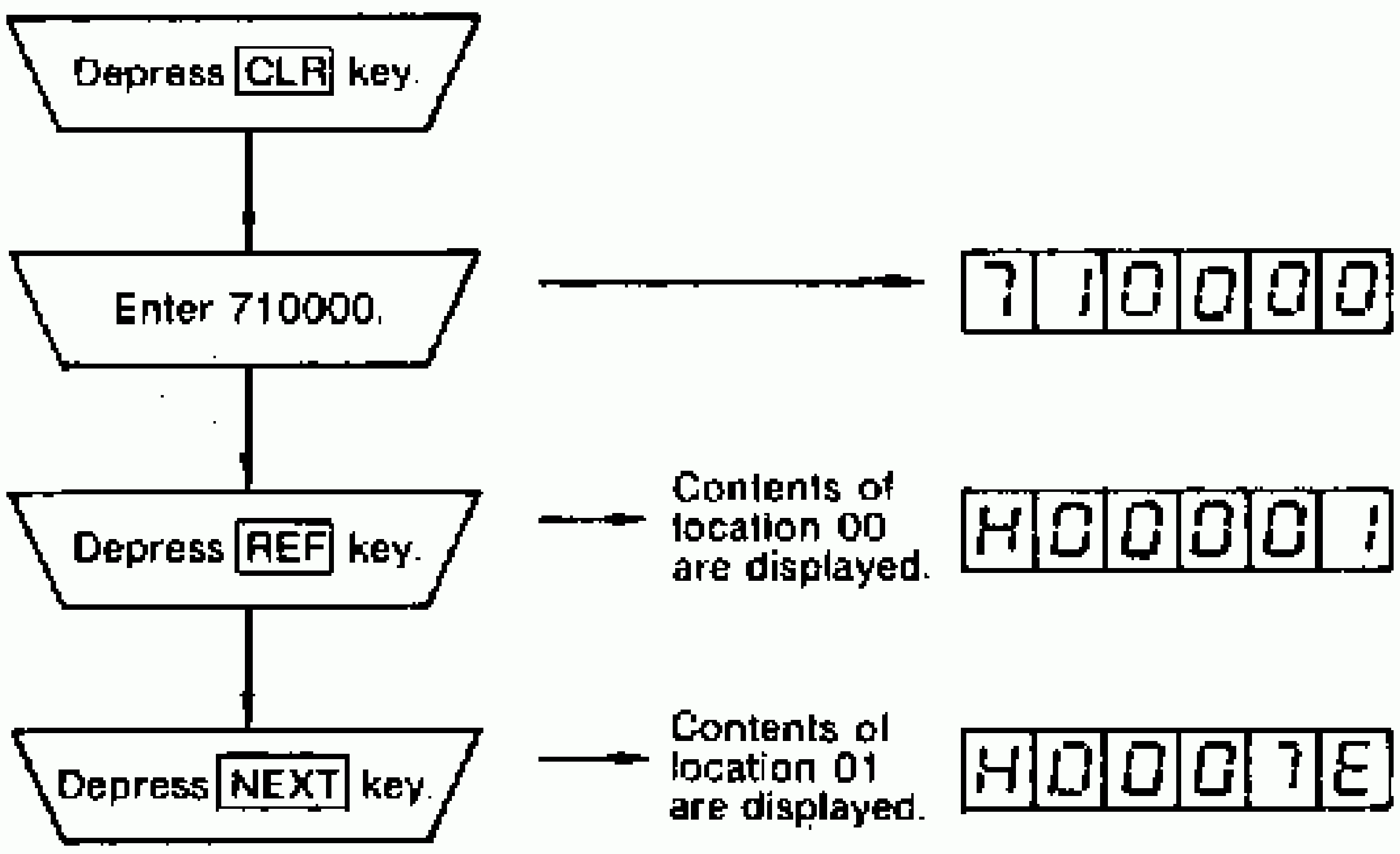
- The value of Y is incremented (decremented) by one every time you push [NEXT] ([PREV]).

(7) Displaying of Contents of Communication Buffer

As in communication parameters, it is possible to display the contents of the communication buffer of a MEMOBUS port by entering a special number which is described below. 7 and 0 are fixed.

```
7X0YZZ
 | | └── Buffer Address (hex)
 | |       ZZ = 00-FF
 | |
 | └─Selects Transmission or Reception Buffer
 |
Port Number          Y = 0   Receiving buffer
X = 1   Port 1       Y = 1   Transmitting buffer
X = 2   Port 2
X = 3 | For future
X = 4 | expansion
```

Operate as follows to look at the contente of transmitting buffer of port 1, for example.



- The data is displayed in hex form at first.
- The buffer address ZZ is incremented (decremented) by one every time you push [NEXT] ([PREV]).

(8) Displaying of U84 System Status

It is possible to display the U84 system status by entering a special number which is described below. 3 and 0 are fixed.

30××××
└──── Address (hex) of Location Storing System Status

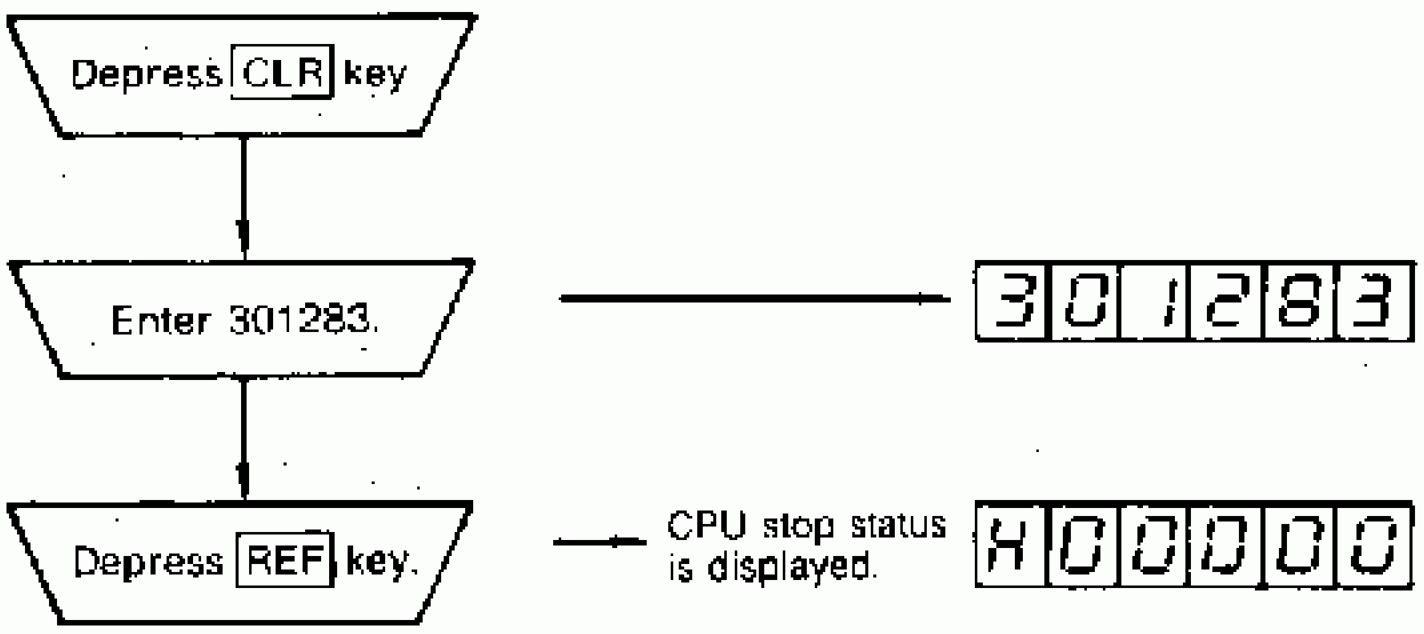
Table 7.4 shows the addresses of the locations where the U84 system status is stored.

Table 7.4 U84 System Status

Address (Hexadecimal)	System Status	Order	Page
1282	U84 machine status	1	216
1283	U84 stop status	2	217
1284	Local I/O CH1 (I/O CH #1) status	3	217
1285	Local I/O CH2 (I/O CH #2) status	4	217
1286	Remote I/O CH1 (I/O CH #3) status	5	217
1287	Remote I/O CH2 (I/O CH #4) status	6	217
1288	Remote I/O CH3 (I/O CH #5) status	7	217
1289	Remote I/O CH4 (I/O CH #6) status	8	217
128A	Remote I/O CH5 (I/O CH #7) status	9	217
128B	Remote I/O CH6 (I/O CH #8) status	10	217
128C	Remote I/O CH7 (I/O CH #9) status	11	217
128D	Remote I/O CH8 (I/O CH #10) status	12	217
128E	Communication module status	13	217
128F	Local or remote I/O driver module status	14	217
1290	No. 1 ASCII module status	15	217
1291	No. 2 ASCII module status	16	217
1292	No. 3 ASCII module status	17	217
1293	No. 4 ASCII module status	18	217
1294	For future expansion	19	—
1295	U84 dual stop status	20	
1296	U84 dual machine status	21	

Note: The status shown above may be read with a STAT Command in a ladder circuit. The order means that read with STAT.

Operate as follows to read the CPU stop status, for example.



- The data is displayed in hex form at first.
- The address ×××× is incremented (decremented) by one every time you push NEXT (PREV).

7.5.3 Error Codes

(1) Wrong Operation

Table 7.5 summarizes the codes indicating wrong operations. Wrong data would not be stored in memory, but the normal state can be restored by recovery operation. Note that no error code will be displayed on RAP if you make a similar mistake on the programming panel: it appears only when you operate RAP.

Table 7.5 Error Codes of Wrong Operations

Error Code	Error Description	Example
EEE001	Function code error (MSD is not 0, 3, 6, nor 7.)	100001, [REF]
EEE002	Auxiliary function code error (Second MSD is not 0, 1, 2, 3, or 4.)	050000, [REF]
EEE003	Reference number outside permissible range.	015000, [REF]
EEE004	Entered data outside permissible range.	To enter a number exceeding 65536 in a holding register.
EEE005	Input relay/coil not disabled	To turn ON/OFF enabled coil 00001
EEE006	Setting not permitted	To enter a number in input register 3x x x x
EEE007	Memory protect ON	To depress [ENTR] key when memory protect is ON.
EEE008	Other wrong operations	–
EEE009	For future expansion	–

Recovery Operation

If error code is 1, 2, or 3, push [CLR] then error indication is disappeared and the six 0s are displayed. Enter the correct number and push [REF].

If error code is 4, 5, 6, or 7, push [CLR] then the display is returned to the indication before the occurrence of error.

NOTE

[CLR] not only clears the digital and character displays to 0 but also cancels the function specified. Before making a new entry, it is recommended to push [CLR] twice in order to cancel the data and function entered.

(2) Errors of Communication Module

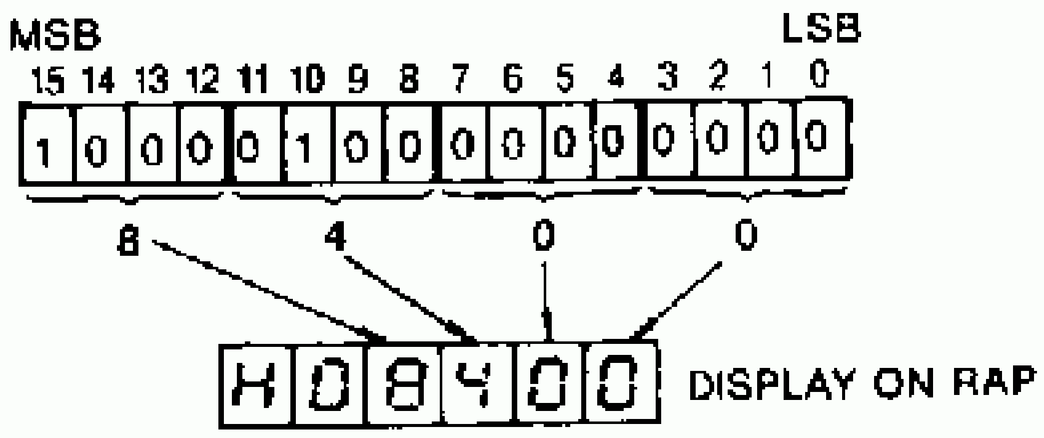
If the communication module is defective, replace it with new one.

Table 7.6 Error Codes

Error Code	Error Description
DEAD 05	PROM total check error after power ON
DEAD 10	RAM (work address) error
DEAD 20	RAM (for interface) error
DEAD 40	Watchdog timer error
DEAD 50	Communication parameter error

7.6 U84 SYSTEM STATUS

The U84 system status is stored in fixed address of the CPU memory as described in par. 7.5(8) and can be indicated on RAP. The system status is indicated by bit patterns or codes. A 16-bit binary data can be displayed in hex form as follows.



Each hex digit is displayed by four binary digits as follows. The relationship between the binary and hex indications permits us to know what bits are 1 from the indication on RAP.

Hexadecimal	Binary	Hexadecimal	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

(1) U84 Machine Status (Address 1282)

Bit	Description
Bit 0 (0001)	No allocation
Bit 1 (0002)	
Bit 2 (0004)	
Bit 3 (0008)	For future expansion
Bit 4 (0010)	Watchdog timer error of No. 1 ASCII module
Bit 5 (0020)	Watchdog timer error of No. 2 ASCII module
Bit 6 (0040)	Watchdog timer error of No. 3 ASCII module
Bit 7 (0080)	Watchdog timer error of No. 4 ASCII module
Bit 8 (0100)	Watchdog timer error of communication module
Bit 9 (0200)	Watchdog timer error of local or remote I/O driver module
Bit 10 (0400)	Scan stop
Bit 11 (0800)	CPU No. in dual system (0: No. 1, 1: No. 2)
Bit 12 (1000)	Battery voltage drop
Bit 13 (2000)	No allocation (always 1)
Bit 14 (4000)	Program memory size: (0: 16 k words, 1: 24 k words)
Bit 15 (8000)	Dual flag (0: single, 1: dual)

(2) U84 Stop Status (Address 1283)

Bit	Description
Bit 0 (0001)	For future expansion
Bit 1 (0002)	
Bit 2 (0004)	Local or remote I/O driver module error
Bit 3 (0008)	System bus error
Bit 4 (0010)	Status memory parity error
Bit 5 (0020)	Status memory total check error
Bit 6 (0040)	Program memory parity error
Bit 7 (0080)	Program memory total check error
Bit 8 (0100)	Total check error in I/O allocation table
Bit 9 (0200)	Logic solver error
Bit 10 (0400)	Real time clock error
Bit 11 (0800)	Watchdog timer error
Bit 12 (1000)	Illegal network data error
Bit 13 (2000)	No start of network
Bit 14 (4000)	No end of logic
Bit 15 (8000)	U84 system stop

Note: This status alone will be displayed also on the P190 programming panel as follows.

STOPPED U84 SYSTEM ERROR: ×××× (hexadecimal)

(3) Local I/O CH1 Status (Address 1284) and CH2 Status (Address 1285)

Status Codes (Hexadecimal)	Description
0000	CH1 or CH2 normal
0001	I/O bus error
0002	I/O busy time out
0004	I/O busy error
0008	No allocation
0010	
0020	

(4) Remote I/O CH1 to CH8 Status (Address 1286 to 128D)

Status Codes (Hexadecimal)	Description
0000	Channel normal
0001	I/O bus error
0002	I/O busy time out
0004	I/O busy error
0008	No response from remote I/O receiver
0010	No I/O data in remote I/O receiver
0020	Communication error with remote I/O receiver

(5) Communication Module Status (Address 128E)

Status Codes (Hexadecimal)	Description
00A0	Module normal

(6) Local or Remote I/O Driver Module Status (Address 128F)

Status Codes (Hexadecimal)	Description
0080	Module normal
0081	ROM error
0082	RAM error
0083	I/O allocation table error
0084	I/O channel error
0085	For internal check
0086 to 008E	No allocation
008F	Watchdog timer error

(7) No. 1 ASCII Module Status (Address 1290)

Status Codes (Hexadecimal)	Description
0010	Module normal (at a standstill)
0018	Module normal (at a RUN)
0011	RAM (message format) error
0012	RAM (message pointer) error
0013	RAM (for interface) error
0014	ROM error
0015	RAM (port parameter) error
0016	Low battery voltage (at a standstill)
001E	Low battery voltage (at a RUN)
0017	No allocation
0019 to 001D	
001F	

(8) No. 2 to No. 4 ASCII Module Statuses (Address 1291 to 1293)

Only 10-digit number of each status codes is different from those of No.1 ASCII module:

- For No.2 ASCII module—××2×
- For No.3 ASCII module—××3×
- For No.4 ASCII module—××4×

7.7 TROUBLESHOOTING FLOW CHART

In order that repairs can be made as easily and quickly as possible without a knowledge of details, the U84 is maintained in basic units of modules, that is, by module replacement.

If a failure occurs, the first thing to do is to identify the problem accurately and follow the prescribed procedure for maintenance. Make use of the general U84 troubleshooting flow charts provided in this section, Figs. 7.18 to 7.24.

Maintenance of the U84 requires spare parts listed on Section 6.4.

NOTE

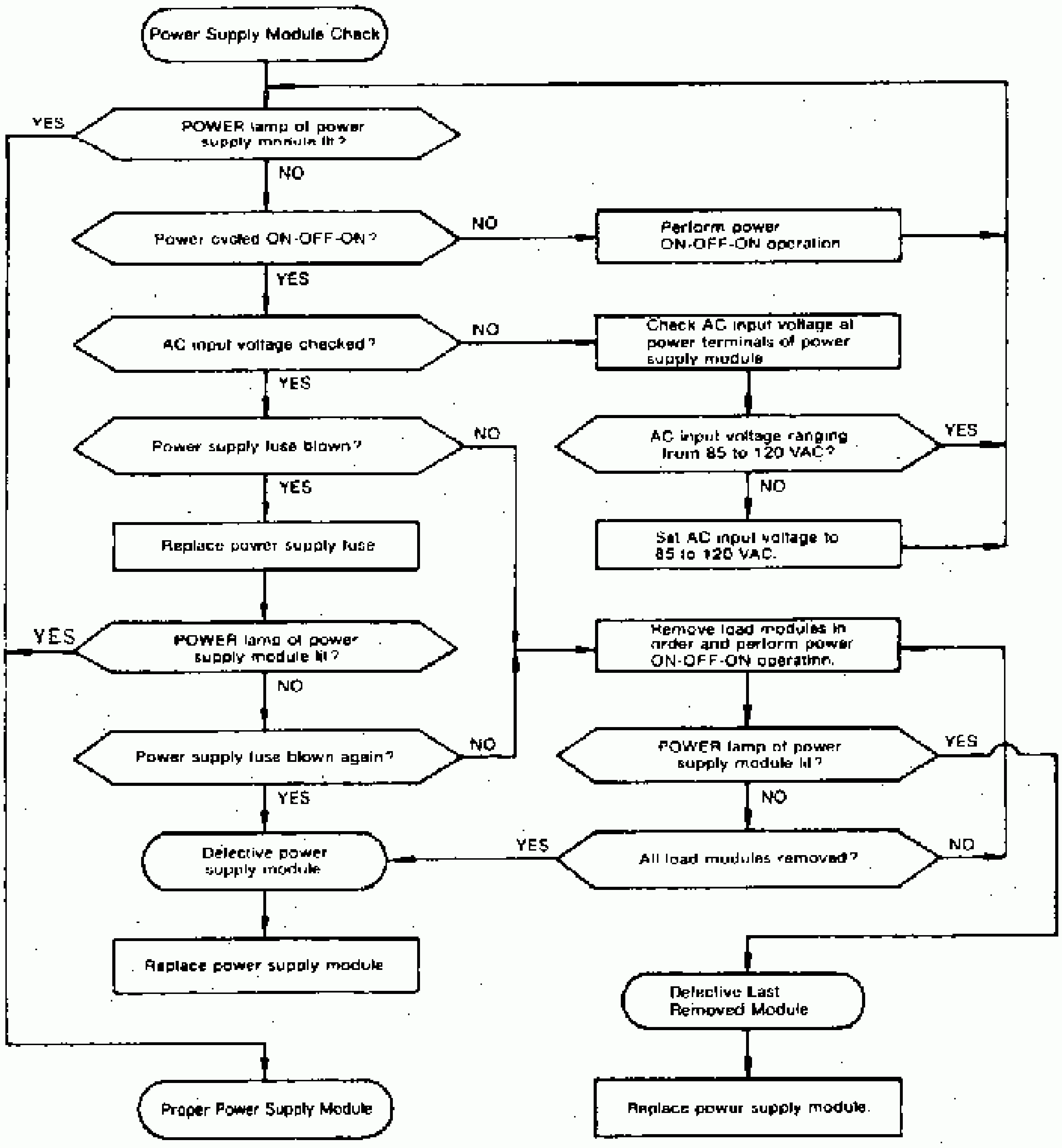
1. The symbols used in the flow charts have the following meanings:

   ▭ : Action　　⬭ : Teminal or comment

   ⬡ : Decision　　○ : Connector

2. Power on / off operations are omitted from the flow charts. Normally, power will be turned off before a maintenance action, and turned on after the action has been completed.

(1) Power Supply Module Check

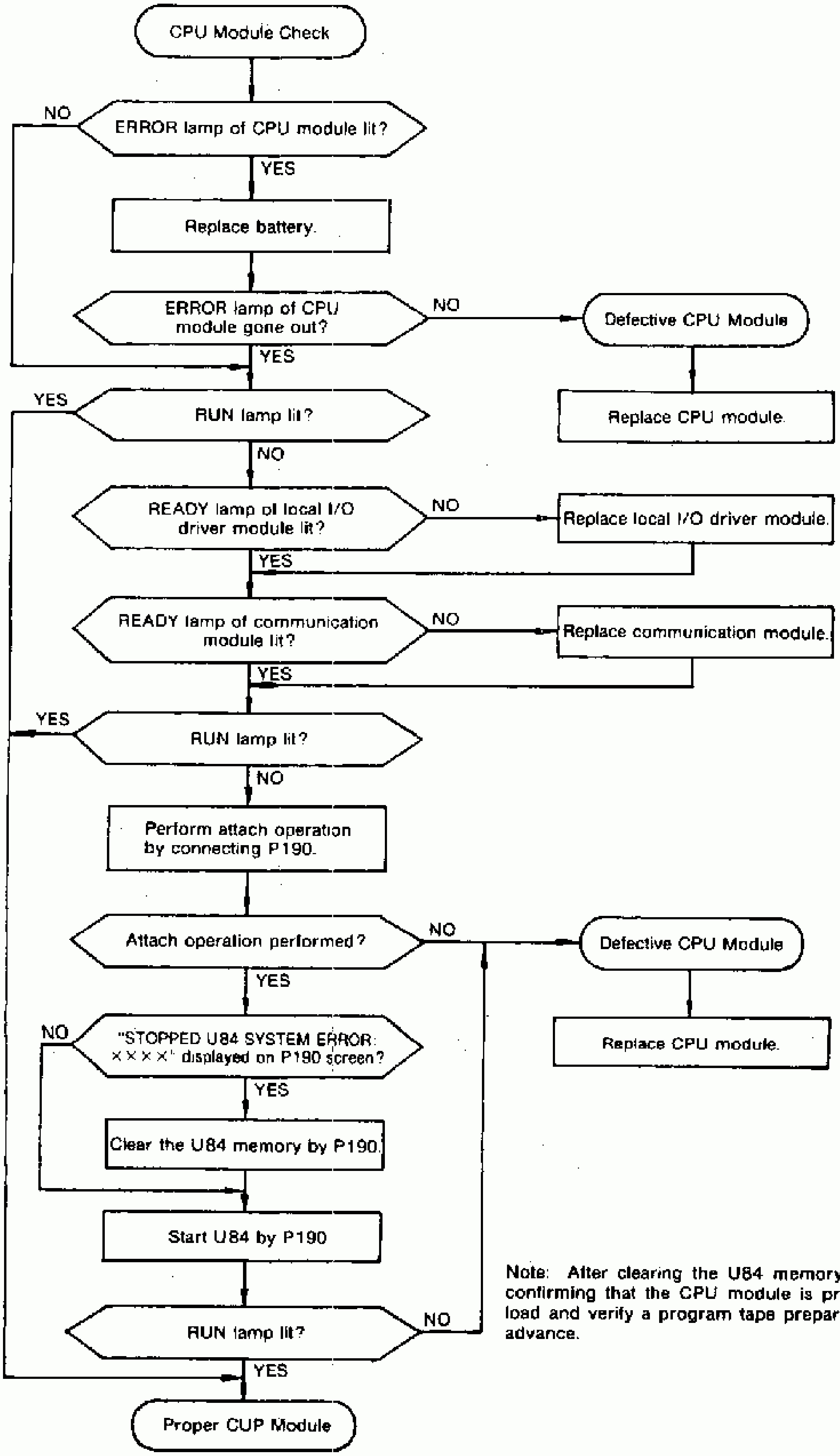


Note: The load modules include the following.

- With main power supply modules: CPU, local I/O driver, communication module
- With auxiliary power supply modules: Local I/O receiver, I/O buffer, I/O modules

Fig. 7.18 Power Supply Module Check

(2) CPU Module Check



Note: After clearing the U84 memory and confirming that the CPU module is proper, load and verify a program tape prepared in advance.

Fig. 7.19 CPU Module Check

(3) Local I/O Driver (LI/OD) Module Check

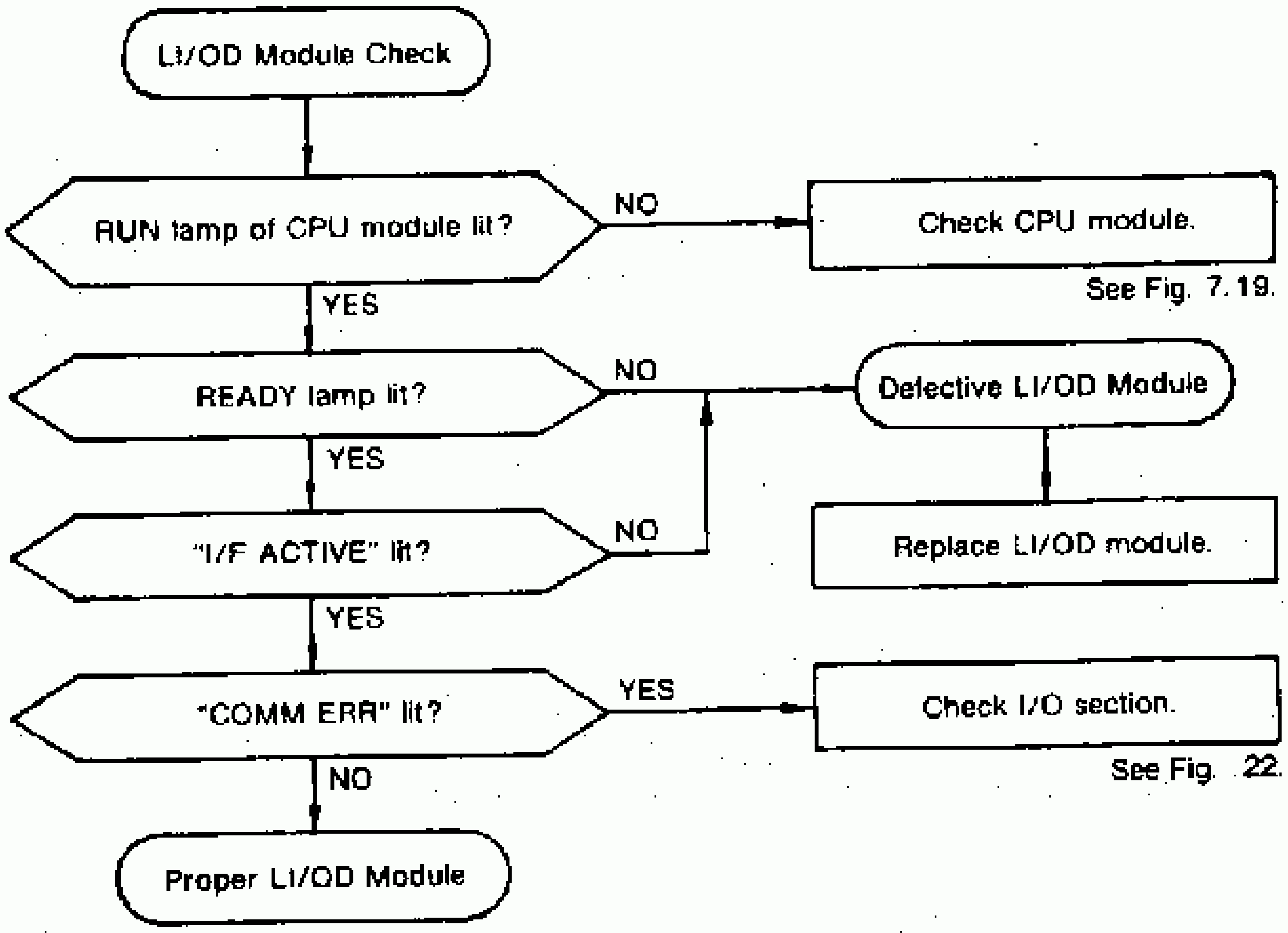


Fig. 7.20 Local I/O Driver Module Check

(4) Communication Module (COMM) Check

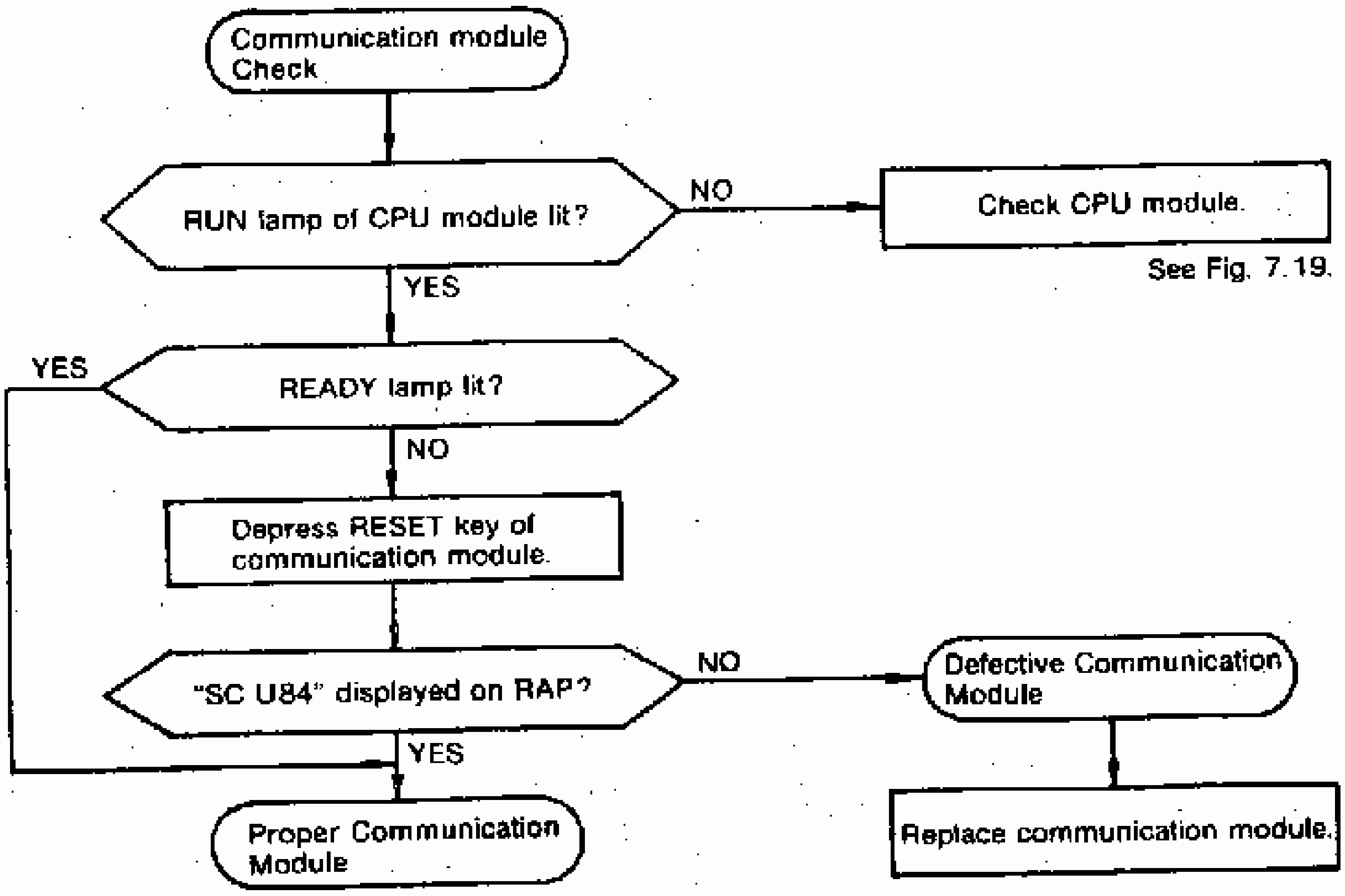


Fig. 7.21 Communication Module Check

(5) I/O Section Check

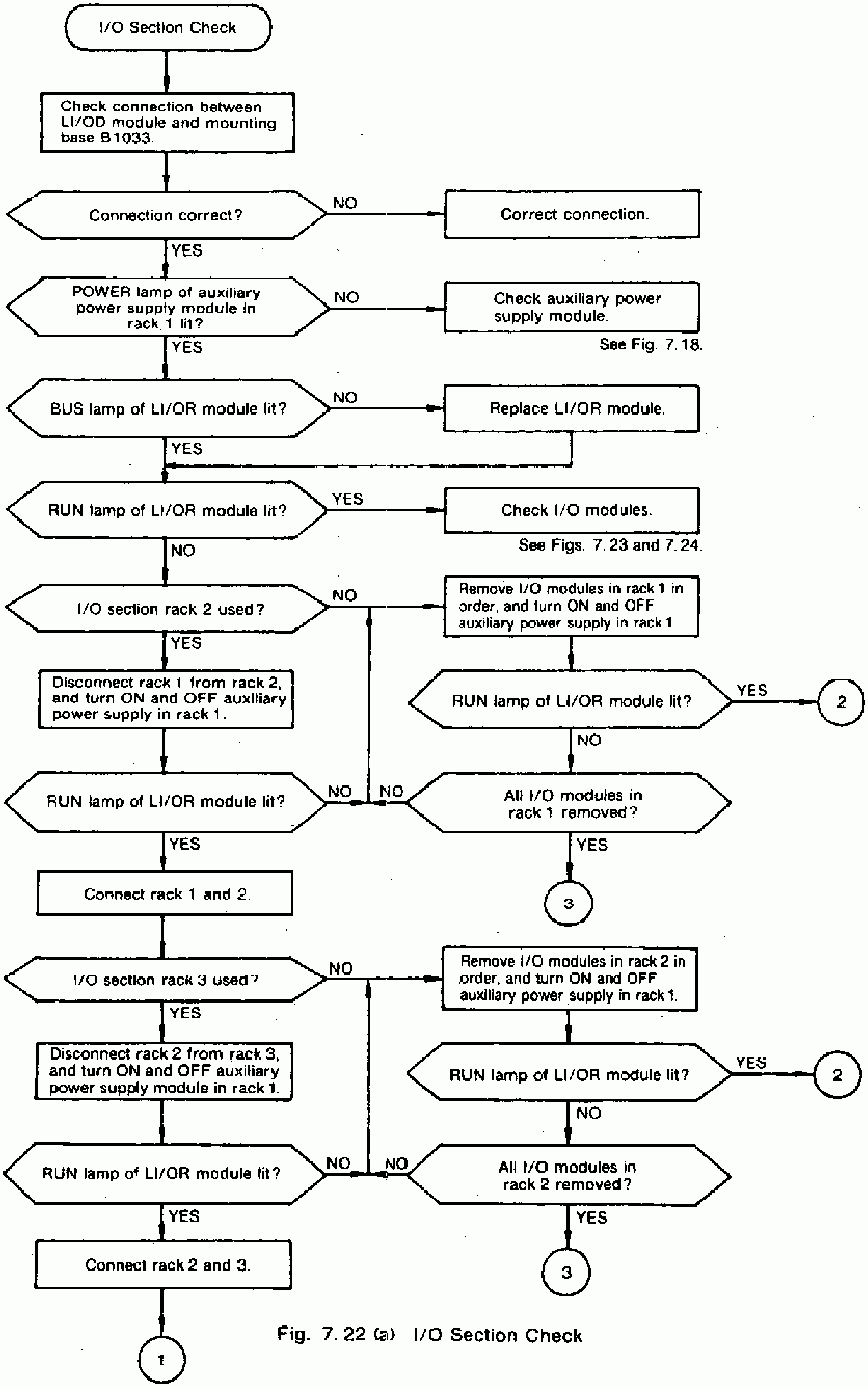


Fig. 7.22 (a) I/O Section Check

(5) I/O Section Check (Cont'd)

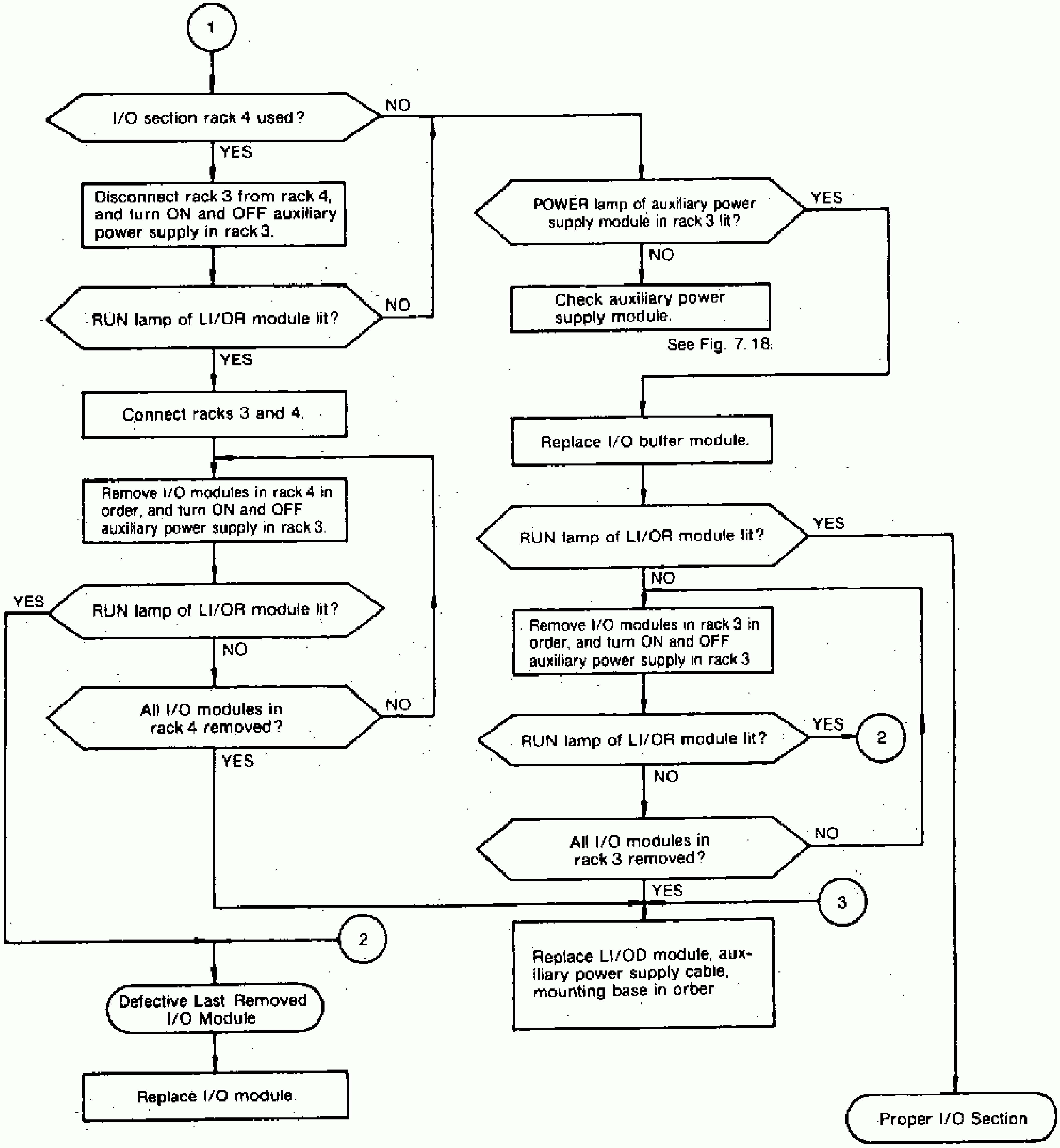


Fig. 7.22 (b) I/O Section Check

(6) Input Module Check

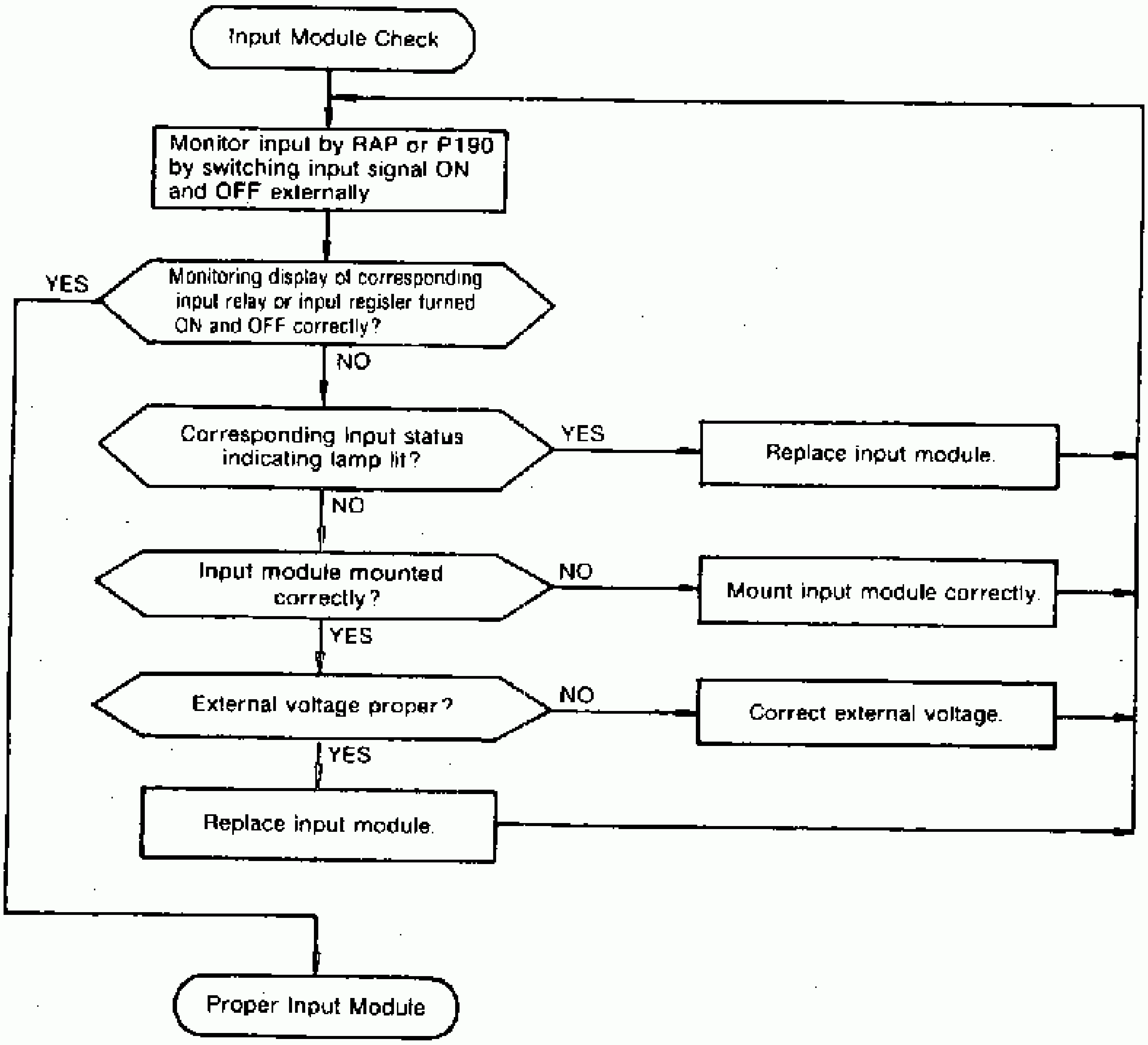


Fig. 7. 23 Input Module Check

(7) Output Module Check

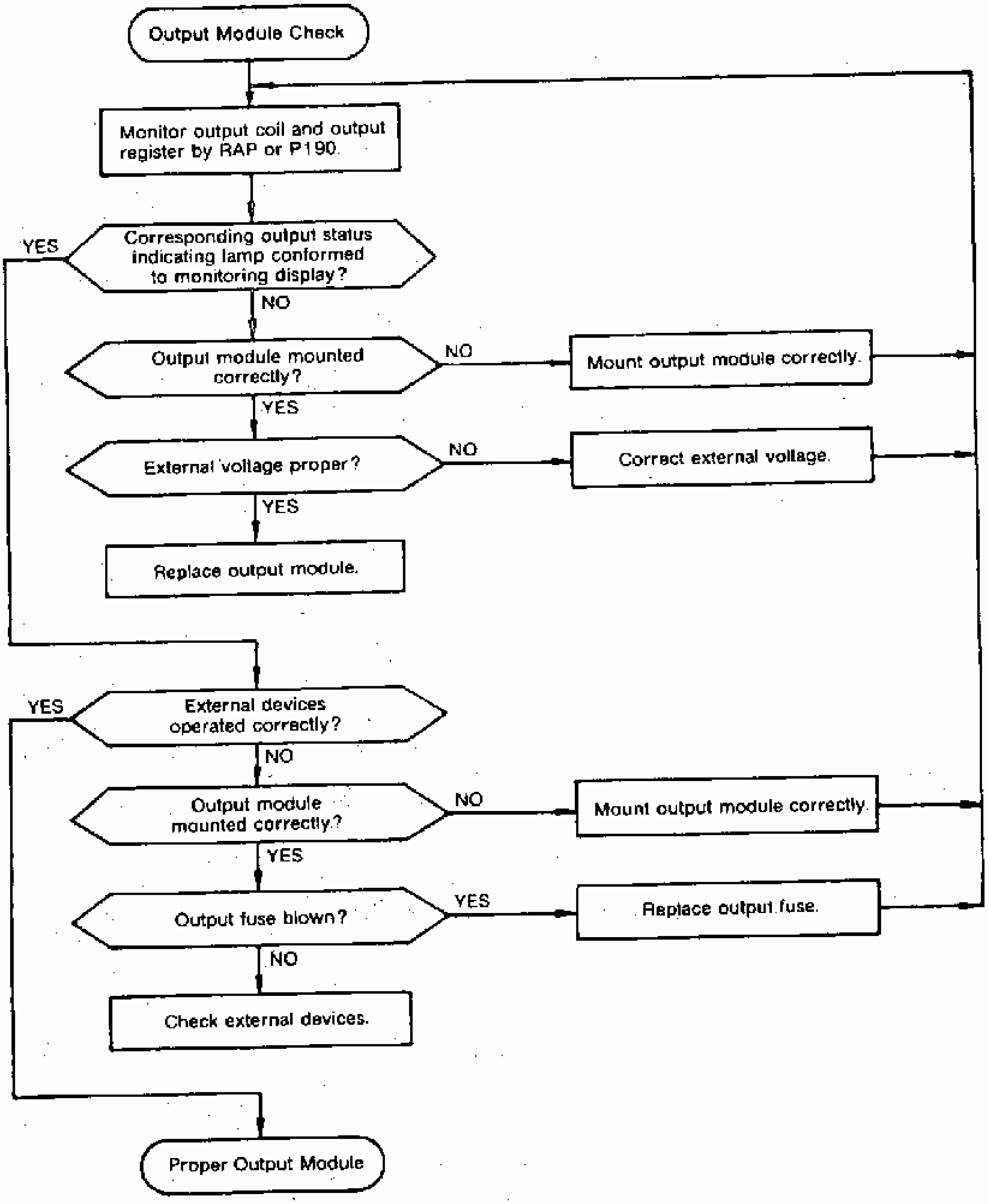


Fig. 7.24 Output Module Check

APPENDIX A
U84 COMPONENTS LIST

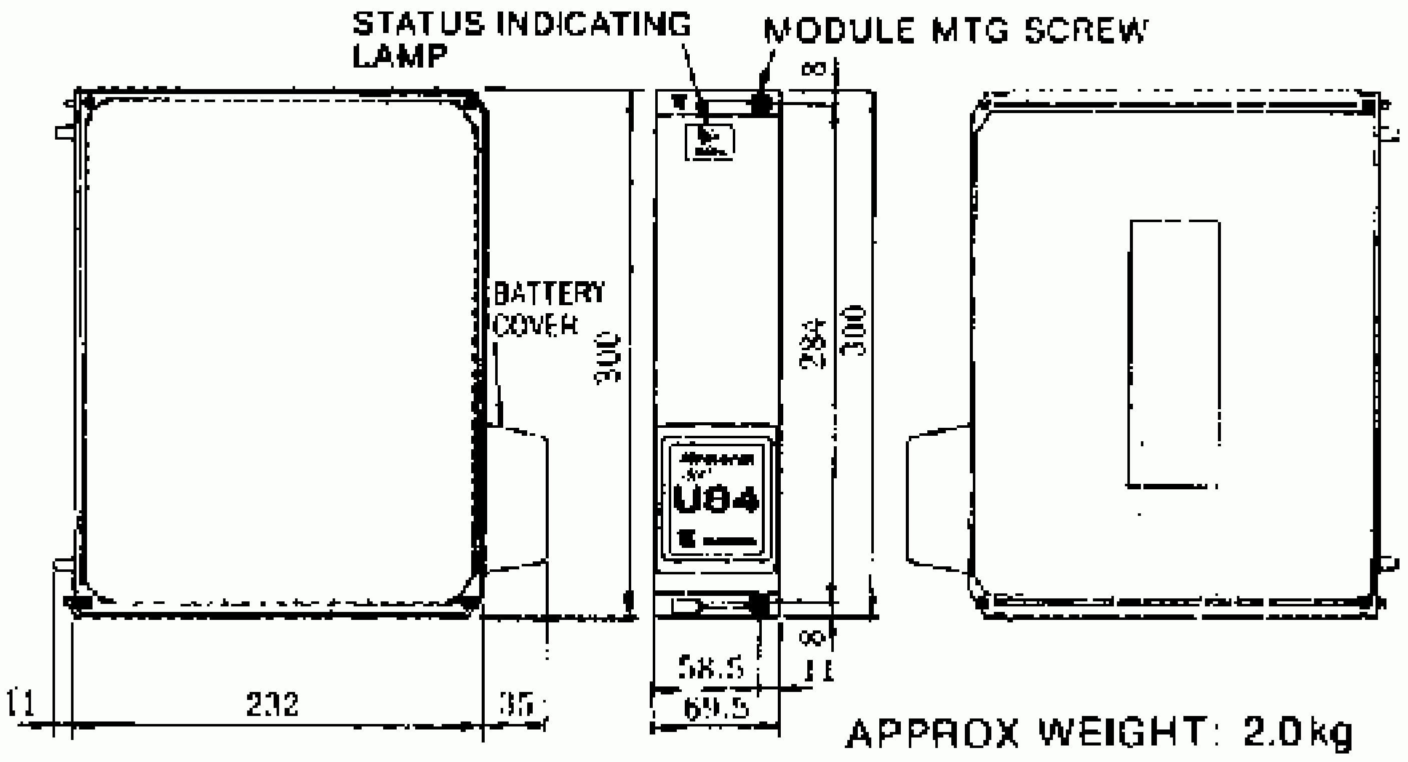
Component		Type	Function or Application	Remarks
CPU Module		DDSCR-U84-116	16 K-word memory size of processor	24 bits/word
		DDSCR-U84-124	24 K-word memory size of processor.	
Switching Module		JAMSC-C8150	Monitoring and switching of dual CPUs	-
Main Power Supply Module		JRMSP-P8101	DC power supply for CPU, switching and peripheral modules.	—
Auxiliary Power Supply Module 1		JRMSP-P8054	DC power supply for remote I/O receiver and I/O modules.	—
Auxiliary Power Supply Module 2		JRMSP-P8051	DC power supply for local I/O receiver, I/O buffer and I/O modules.	—
Peripheral Modules	Communication Module	JAMSC-C8110	For P190 CRT programming panel and MEMOBUS	• With register access panel. • 2 ports per module
	Local I/O Driver Module	JAMSC-C8120	Interface between CPU and local I/O receiver	—
	Electrical Remote I/O Driver Module	JAMSC-C8130	Interface between CPU and remote I/O receiver	—
	Optical Remote I/O Driver Module	JAMSC-C8140	Interface between CPU and optical remote I/O receiver	—
	ASCII Module	JAMSC-C8160	• For control of ASCII devices • For U84 used as MEMOBUS master	2 ports per module
	I/O Link Module	JAMSC-C8180	For connection between U84s	—
	Dual Selection Module	JAMSC-C8150	For selection of 2 CPUs	—
Mounting Base		JRMSI-B1030 JRMSI-B1031	• For CPU, main power supply, switching and peripheral modules • For dual selection, power supply and peripheral modules	—
		JRMSI-B1032	For CPU and main power supply modules in dual system	-
		JRMSI-B1033	For local I/O receiver, aux power supply 1 and I/O modules	9 I/O modules max
		JRMSI-B1034	For remote I/O receiver, aux power supply 2 and I/O modules.	8 I/O modules max
		JRMSI-B1027	For I/O modules	12 I/O modules max
		JRMSI-B1028	For aux power supply 1, I/O buffer and I/O modules	9 I/O modules max
Local I/O Receiver Module		JAMSC-S8125	Interface between local I/O driver and I/O modules	
Electrical Remote I/O Driver Module		JAMSC-C8135	Interface between remote I/O driver and I/O modules	-
Optical Remote I/O Receiver Module		JAMSC-C8145	Interface between optical remote I/O driver and I/O modules	—
I/O Buffer Module		JAMSC-B1011	I/O bus buffer	—
I/O Cable		JZMSZ-W1021	For connection between mounting bases B1027 and B1028.	Cable length: 1.5 m
		JZMSZ-W1022		Cable length: 0.4 m
		JZMSZ-W1011-1	For connection between local or remote I/O driver and mounting base B1033.	Cable length: 1 m
		JZMSZ-W1011-2		Cable length: 5 m
Dual System Cable		JZMSZ-W1013	For connection between switching module and mounting base B1032 (Each 2 cables used)	Cable length: 0.4 m
		JZMSZ-W1014		
Remote I/O Cable (Coaxial Cable)		JZMSZ-W1003-1	For connection between remote I/O driver/receiver and main cable (Type BNC connector for remote I/O driver or receiver, and type F male connector for main cable).	Cable length: 2 m
		JZMSZ-W1003-2		Cable length: 5 m
		JZMSZ-W1003-3		Cable length: 10 m
		JZMSZ-W1004	For connection between remote I/O driver and receiver (Type BNC connectors used for both ends).	Cable length: 10 m
		JZMSZ-W453-001	For connection between tap and splitter, or tap and tap (Type F male connectors used for both ends).	Cable length: 2 m
		JZMSZ-W453-002		Cable length: 5 m
		JZMSZ-W453-003		Cable length: 10 m
Tap		T-0168	For signal branch (one direction)	dB loss: −10 dB
Splitter		T-0169	For signal branch (both direction)	dB loss: −3 dB
Adapter		T-0170	For connection between type F male connectors	—
I/O Module		JAMSC-B1051B	100 VAC input	• 16 circuits per module (8 circuits for B1094) • With signal status indicating lamp
		JAMSC-B1053	5−12 VDC input	
		JAMSC-B1055	200 VAC input	
		JAMSC-B1057	48 VDC input	
		JAMSC-B1059C	24 VDC input	
		JAMSC-B1090B	Relay contact output	
		JAMSC-B1094	Power relay contact output	

Component	Type	Function or Application	Remarks
I/O Module (Cont'd)	JAMSC-B1050	100 VAC output	• 16 circuits per module • With signal status indicating lamp • With fuse and fuse-blown indicating lamp (1 fuse per 8 circuits)
	JAMSC-B1052	5–12 VDC output	
	JAMSC-B1054	200 VAC output	
	JAMSC-B1056	48 VDC output	
	JAMSC-B1058	24 VDC output	
	JAMSC-B1060	24 VDC output	• 64 circuits per module • With status indicating lamp and indicating selection switch
	JAMSC-B1061	24 VDC input	
	JAMSC-B1062	24 VDC output	• 32 circuits per module • Without fuse and fuse-blown indicating lamp
	JAMSC-B1063	24 VDC input	
	JAMSC-B1064	24 VDC output	
	JAMSC-B1065	24 VDC input	
	JAMSC-B1070	Register output	• 8 circuits per module • With no indicating lamp and fuse.
	JAMSC-B1071	Register input	
	JAMSC-B1072B-1, -2	Analog output (D/A) 0 to +10 V, 0 to +5 V	• 2 circuits per module • Resolution: 10 bits
	JAMSC-B1072B-3, -4	Analog output (D/A) −5 to +5V, −10 to +10V	2 circuits per module
	JAMSC-B1073-1, -2	Analog input (A/D) 0 to +10 V, +1 to +5 V	• 4 circuits per module • Resolution: 10 bits
	JAMSC-B1074-1, -2	Analog output (D/A) 0 to +10 V, 0 to +5 V	• 2 circuits per module • Resolution: 12 bits
	JAMSC-B1074-3, -4	Analog output (D/A) −5 to +5 V, −10 to +10 V	
	JAMSC-B1075-1, -2	Analog input (A/D) 0 to +10 V, +1 to +5 V	• 4 circuits per module • Resolution: 12 bits
	JAMSC-B1081C	Reversible counter	1 circuit per module
	JAMSC-B1082C	Preset counter	
	JAMSC-B1080	PID control	
	JAMSC-B1083C	Positioning control	
	JAMSC-B1089	DC power supply for PID module	4 PID modules max
Programming Panel	DISCT-P190	Storing and checking of program, monitoring, loading and dumping.	
	DISCT-P150		Plasma display
Mini-cartridge Tape for Data: U84 Programmer Tape	TU84-S001	Storing, altering and checking of program, monitoring	
Mini-cartridge Tape for Data: U84 Utility Tape	TU84-S002	Printing out of ladder list	
Mini-cartridge Tape for Data: U84 ASCII Tape	TU84-S003	Storing of ASCII messages	
Mini-cartridge Tape for Data: U84 Loader Tape	TU84-S004	Loading, dumping, verifying	
Mini-cartridge Tape for Data: Blank Tape	T190-000	Dumping of user's application program	
Floppy Disk: U84 Programmer FD	FU84-001	Storing, altering and checking of program, monitoring, loading and dumping.	
Floppy Disk: U84 Ladder Lister FD	FU84-002	Printing out of ladder list	
Floppy Disk: U84 ASCII Programmer FD	FU84-003	Storing, altering and displaying of ASCII messages.	
Floppy Disk: U84 ASCII Message Lister FD	FU84-004	Printing out of ASCII message	
Floppy Disk: U84 Off-line Programmer FD	FU84-005	Creating, altering and editing of program by P150 without CPU.	
Floppy Disk: Blank FD	F150-000	Dumping of user's application program	
Modem	DISCT-J478	FSK modem for MEMOBUS	
Interface Cable	JZMSZ-W1015-1, -2	For connection between followings: • P190 and communication module • U84 master and R84H-M/684H slave • 584 master and U84 slave	Cable length: 2.5 m Cable length: 15 m
	JZMSZ-W181	For connection between P190 and printer.	Cable length: 2.5 m
	JZMSZ-W1018-1, -2	For connection between ASCII module and ASCII devices.	Cable length: 5 m Cable length: 15 m
	JZMSZ-W1007-1	For connection between ASCII module and J478 modem.	Cable length: 5 m For remote use of ASCII devices.
	JZMSZ-W1008-1	For connection between J478 modem and ASCII devices.	
	JZMSZ-W1009-1	For connection between P190 and J478 modem W1007 for printer.	Cable length: 5 m For remote use of printer
	JZMSZ-W1015-T1, T2	For connection between P150 and communication module.	Cable length: 2.5 m Cable length: 15 m
	JZMSZ-W1006-1, -2	For connection between computer and U84 slave.	Cable length: 5 m Cable length: 15 m
	JZMSZ-W1017-1, -2	For connection between followings: • U84 master and J478 modem • U84 slave and J478 modem	Cable length: 5 m Cable length: 15 m
	JZMSZ-W1019-1, -2	For connection between U84 master and U84 slave.	Cable length: 5 m Cable length: 15 m

APPENDIX B

DIMENSIONS in mm

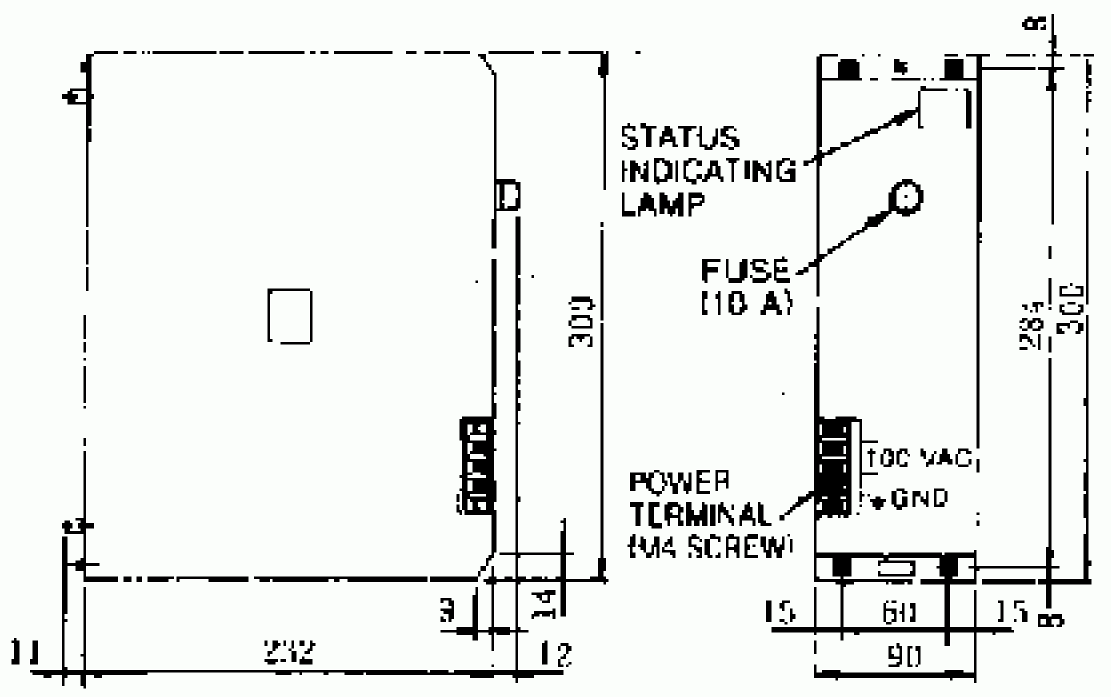
(1) CPU Module Types DDSCR-U84-124, -U84-116



APPROX WEIGHT: 2.0kg

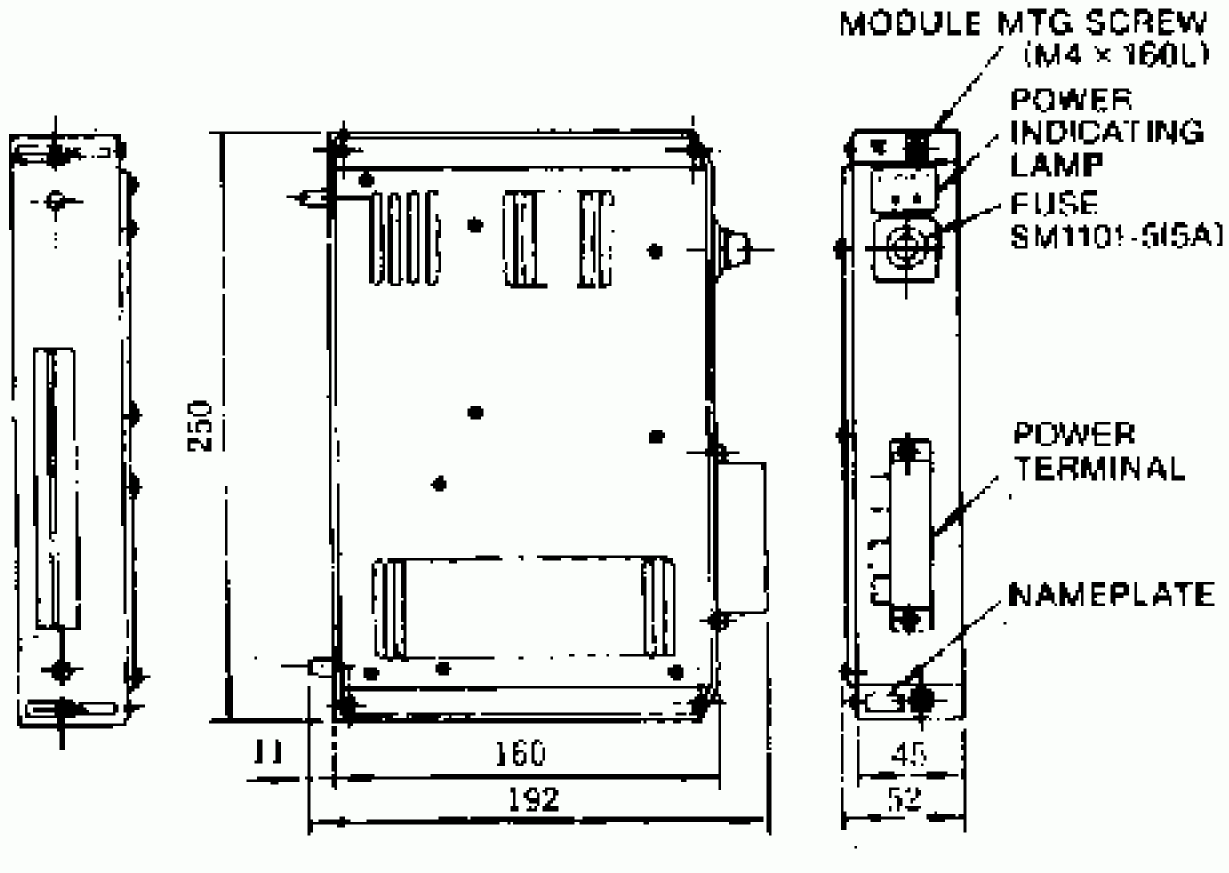
PROGRAM MEMORY	TYPE
16k	DDSCR-U84-116
24k	DDSCR-U84-124

(2) Main Power Supply Module Type JRMSP-P8101



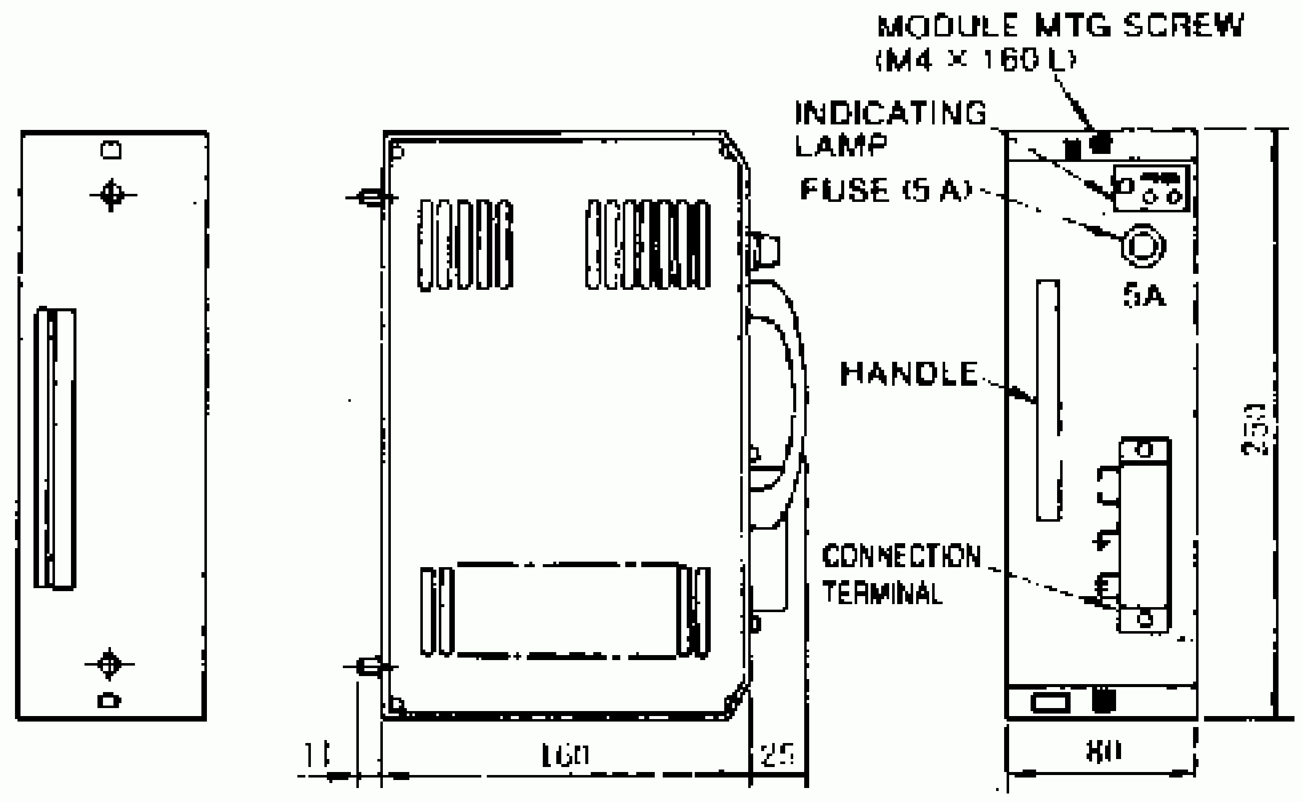
APPROX WEIGHT: 5.0kg

(3) Auxiliary Power Supply Module 1 Type JRMSP-P8054



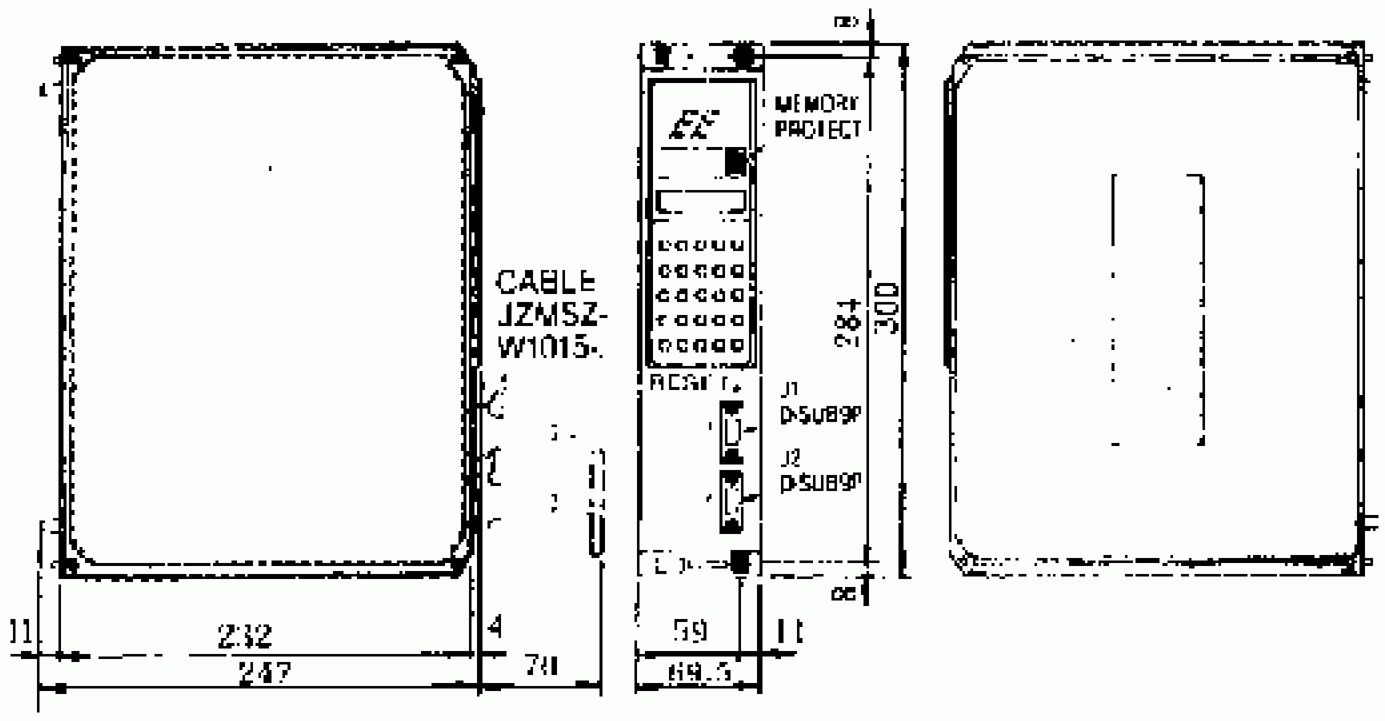
APPROX WEIGHT: 1.1kg

(4) Auxiliary Power Supply Module 2 Type JRMSP-P8051



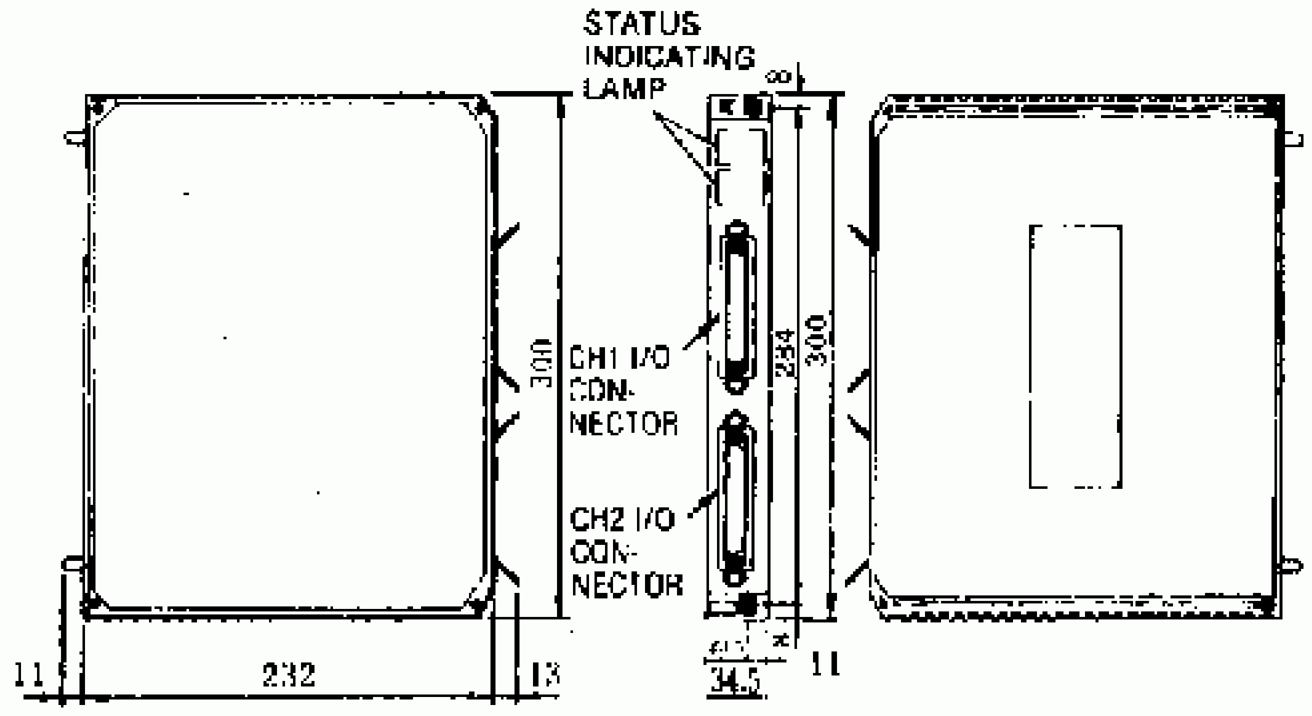
APPROX WEIGHT: 2.8kg

(5) Communication Module Type JAMSC-C8110



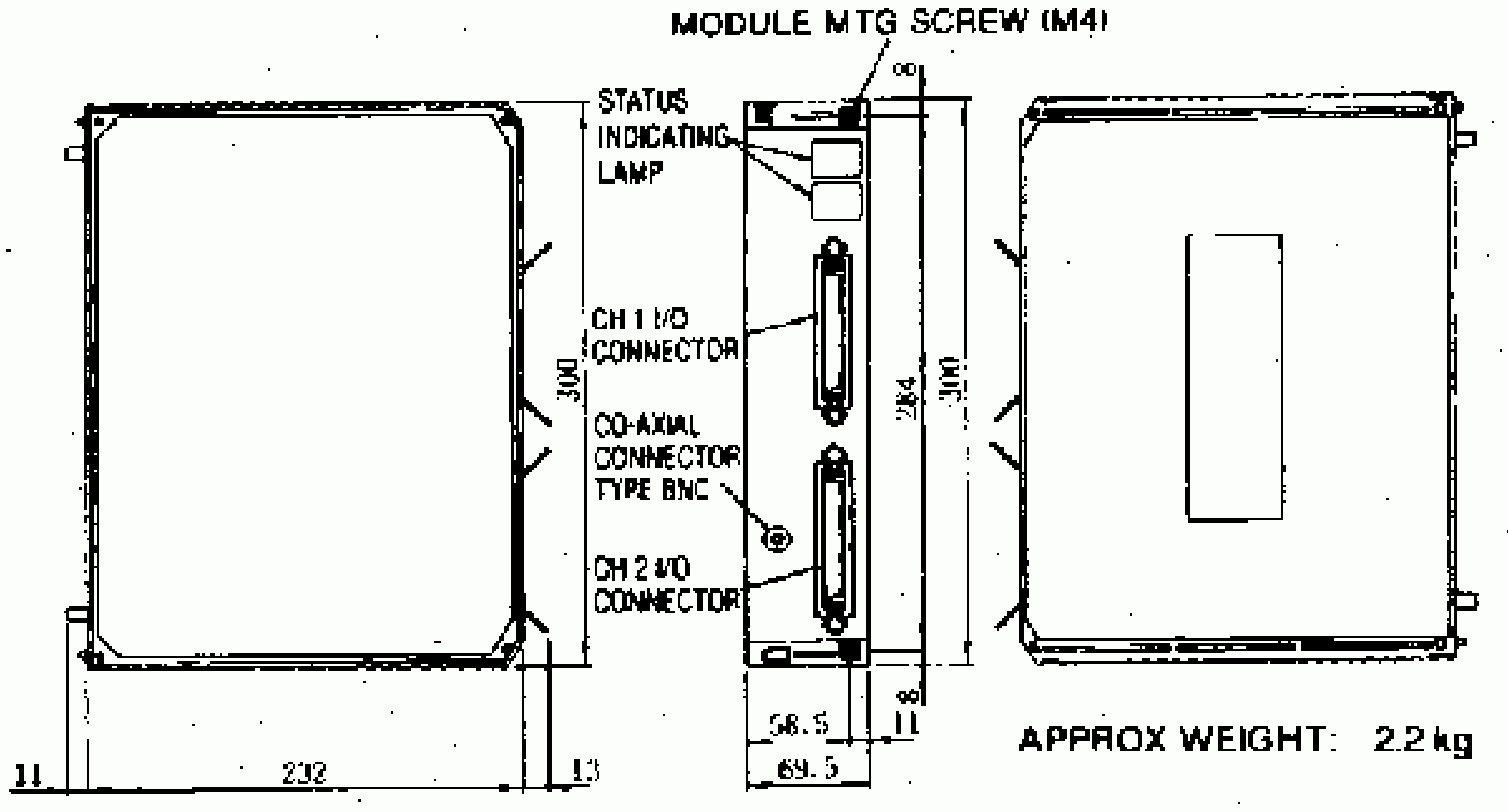
APPROX WEIGHT: 1.4kg

(6) Local I/O Driver Module Type JAMSC-C8120

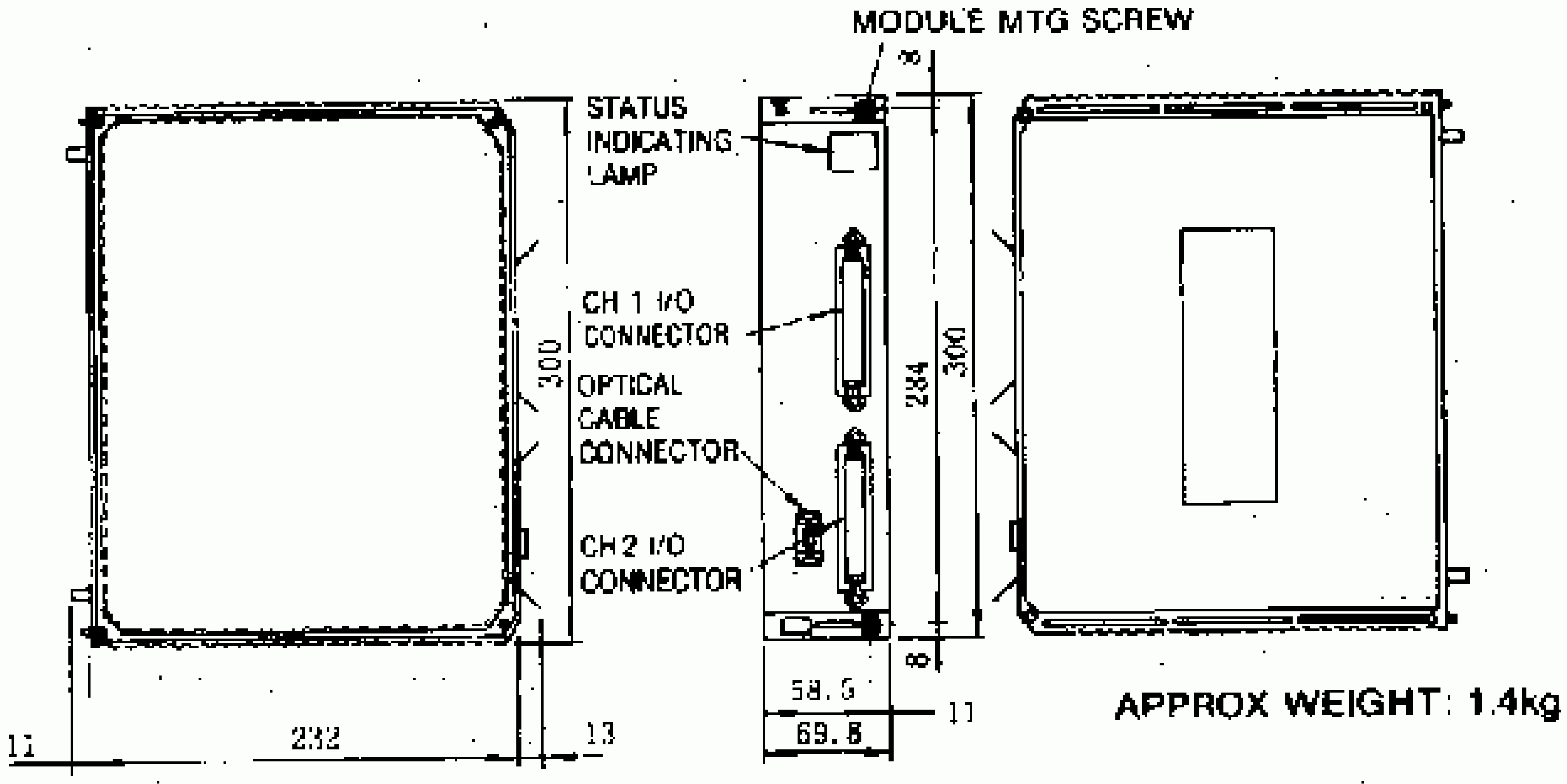


APPROX WEIGHT: 1.2kg

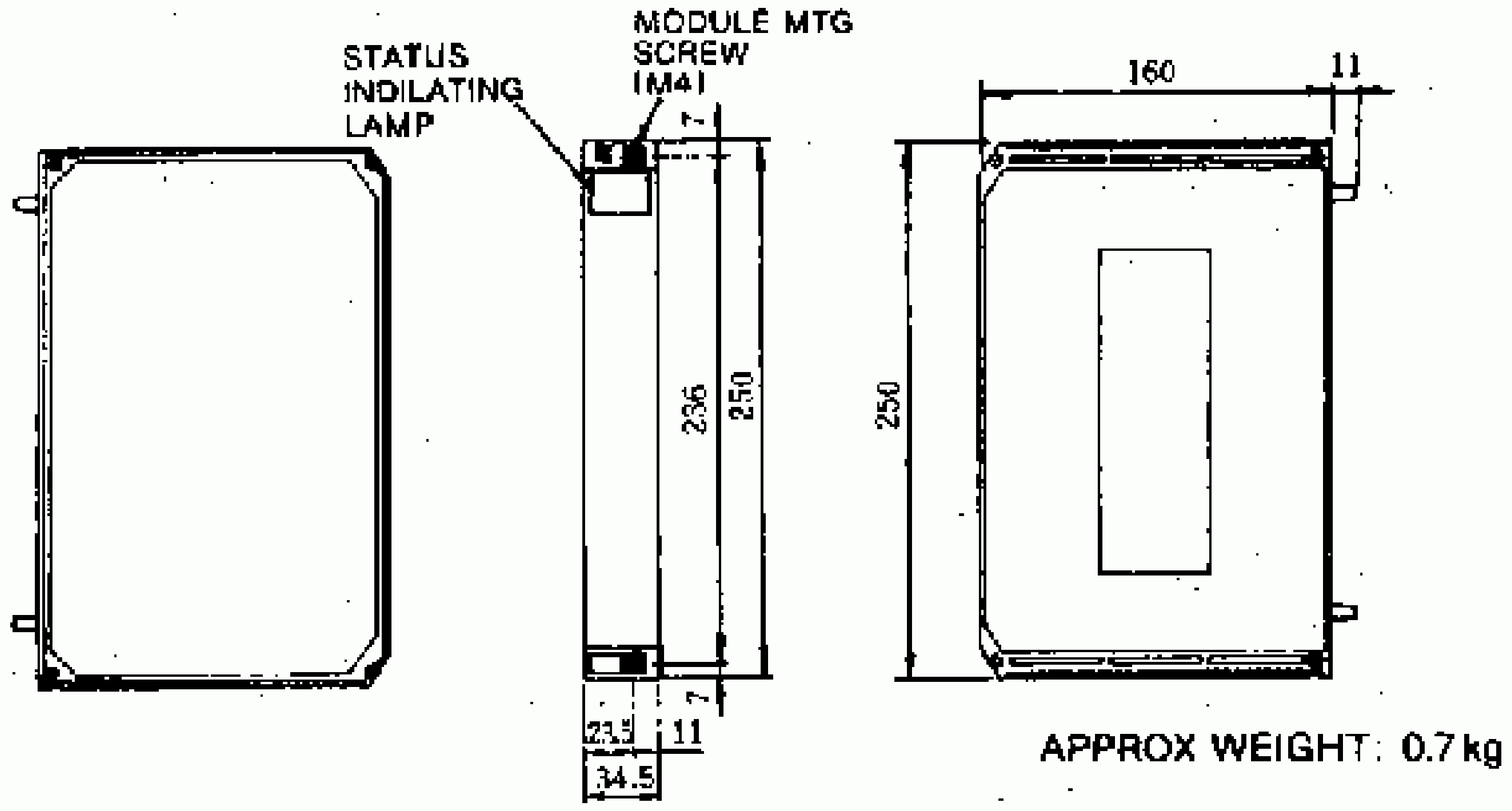
(7) Electrical Remote I/O Driver Module Type JAMSC-C8130



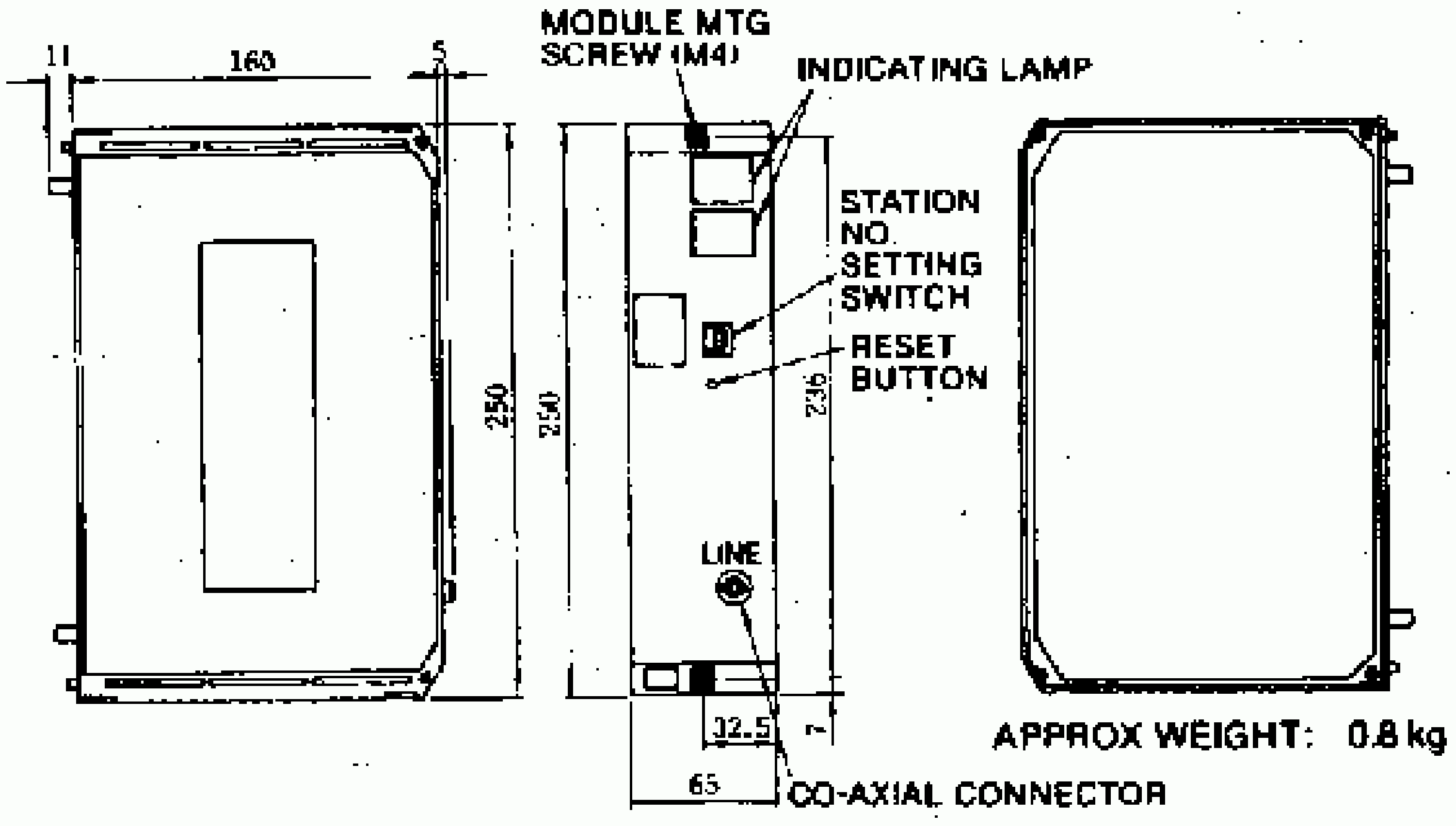
(8) Optical Remote I/O Driver Module Type JAMSC-C8140



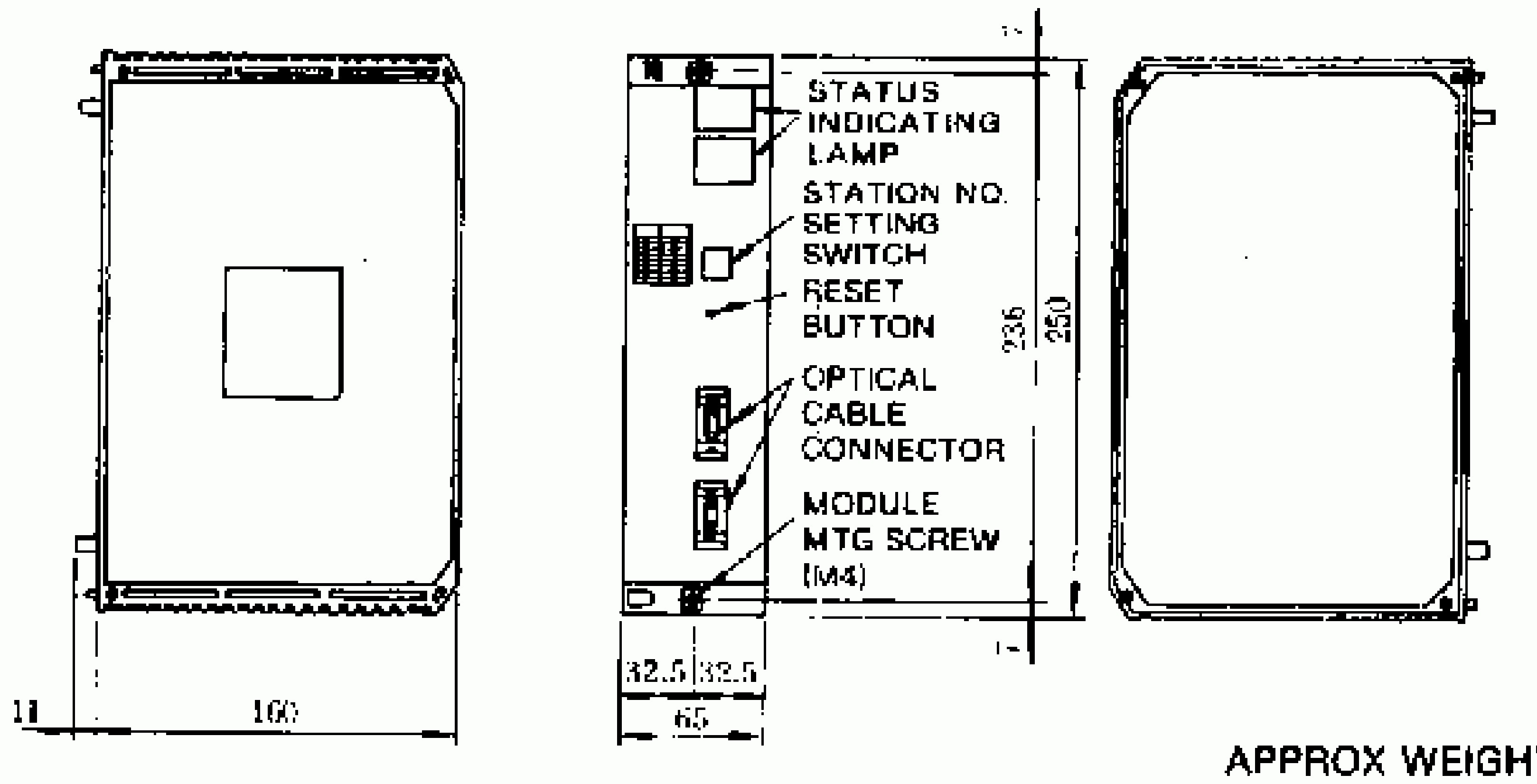
(9) Local I/O Receiver Module Type JAMSC-C8125



(10) Electrical Remote I/O Receiver Module Type JAMSC-C8135

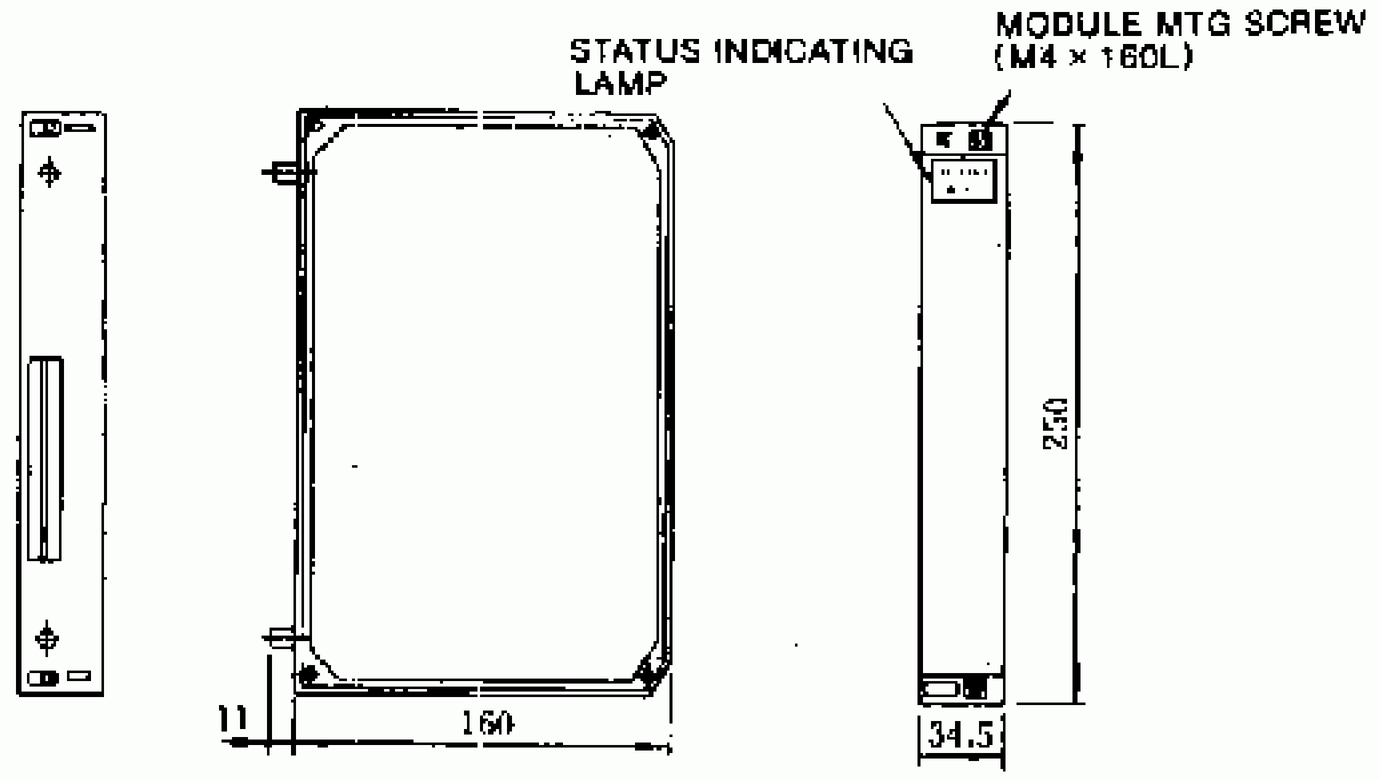


(11) Optical Remote I/O Receiver Module Type JAMSC-C8145



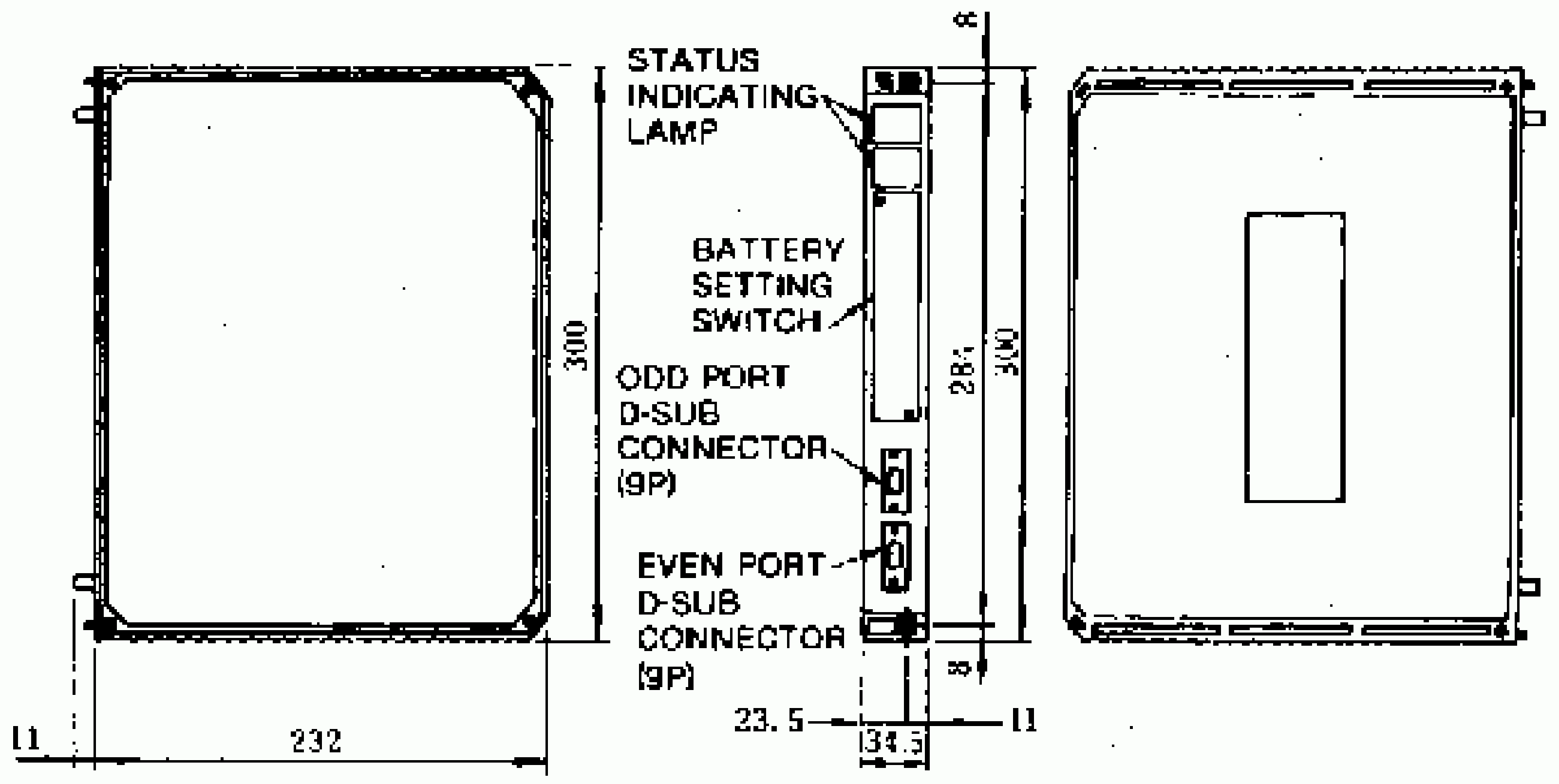
APPROX WEIGHT: 0.8kg

(12) I/O Buffer Module Type JAMSC-B1011



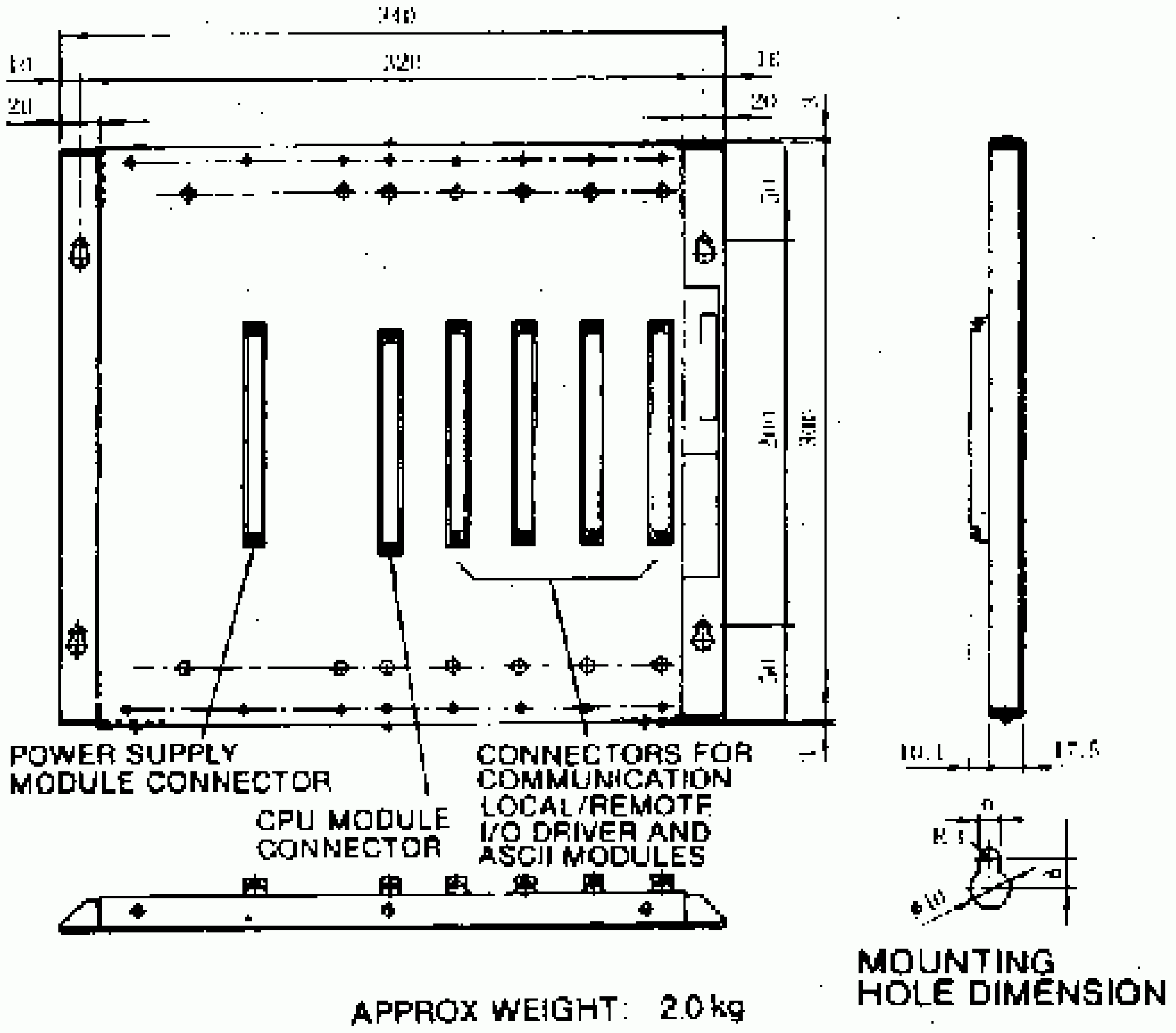
APPROX WEIGHT: 0.7kg

(13) ASCII Module Type JAMSC-C8160

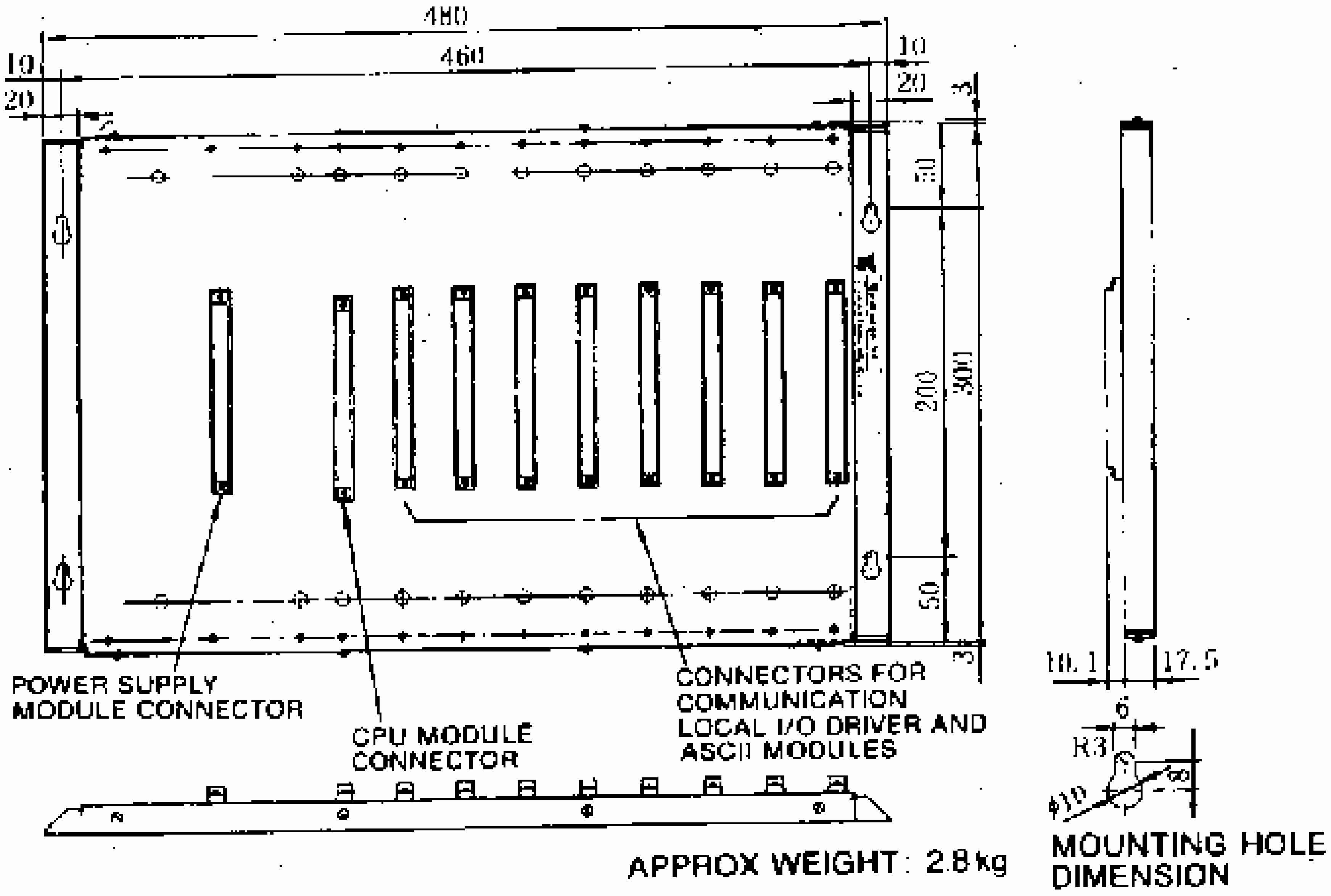


APPROX WEIGHT: 1.2kg

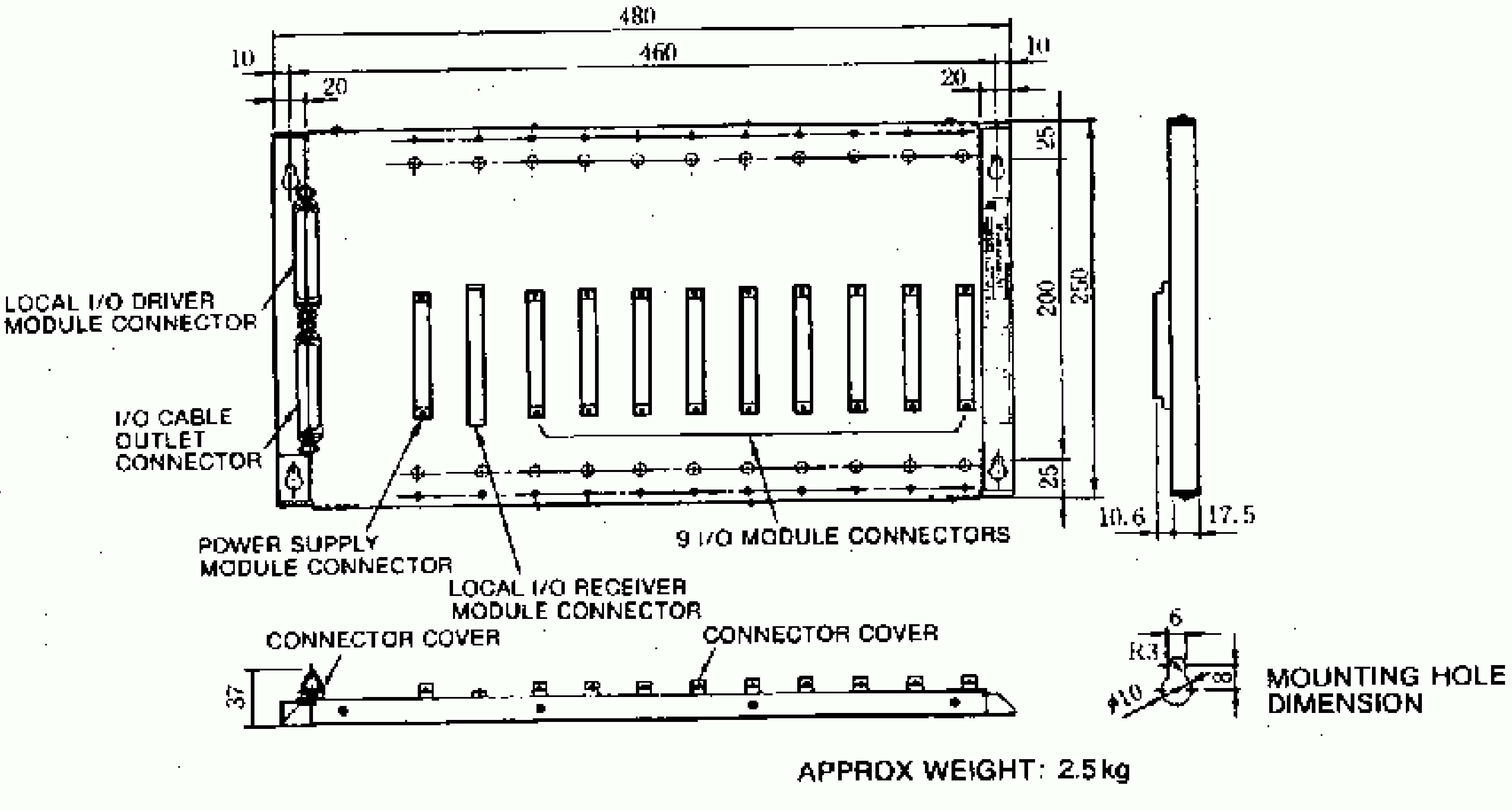
(14) Mounting Base Type JRMSI-B1030



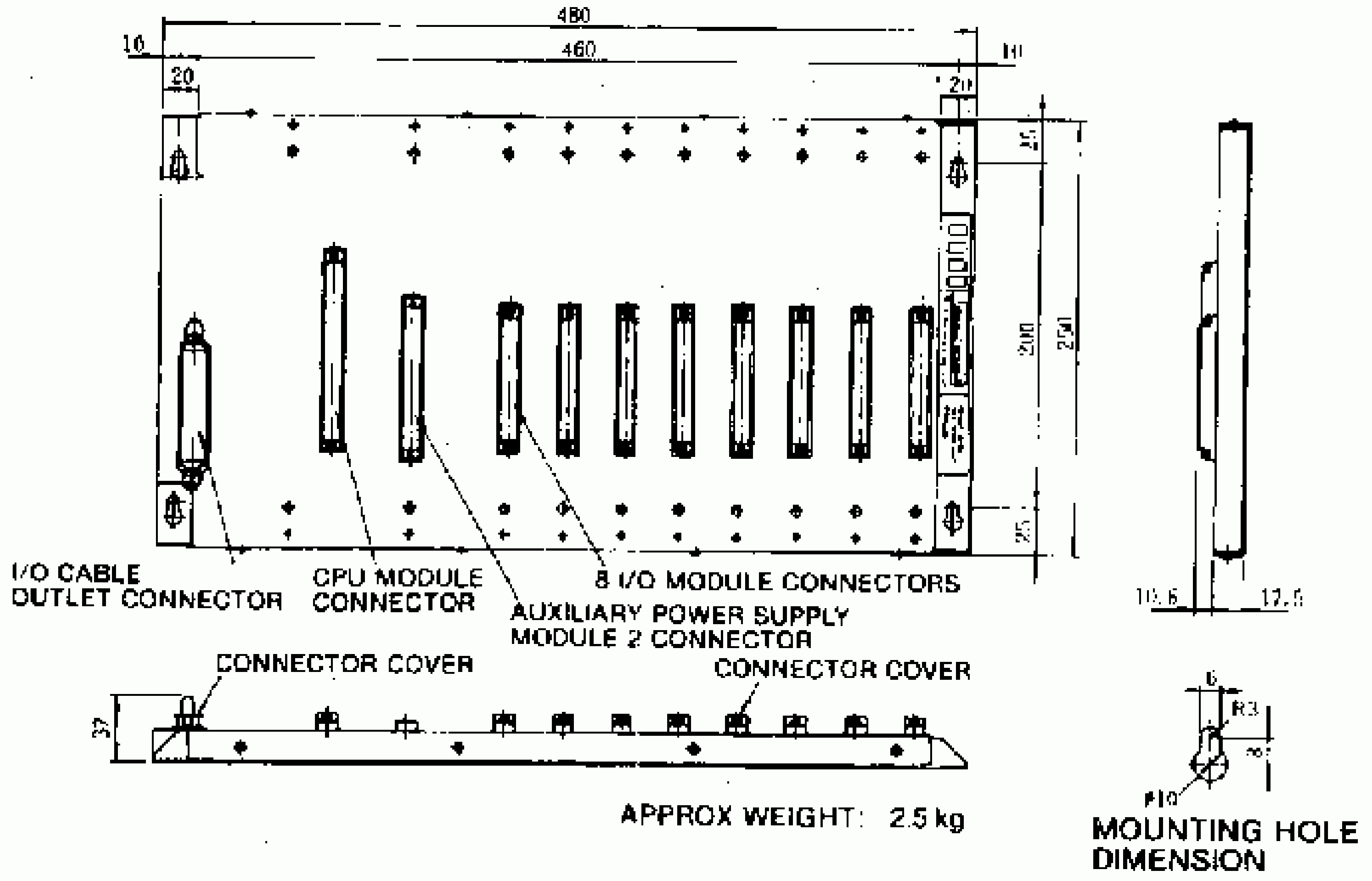
(15) Mounting Base Type JRMSI-B1031



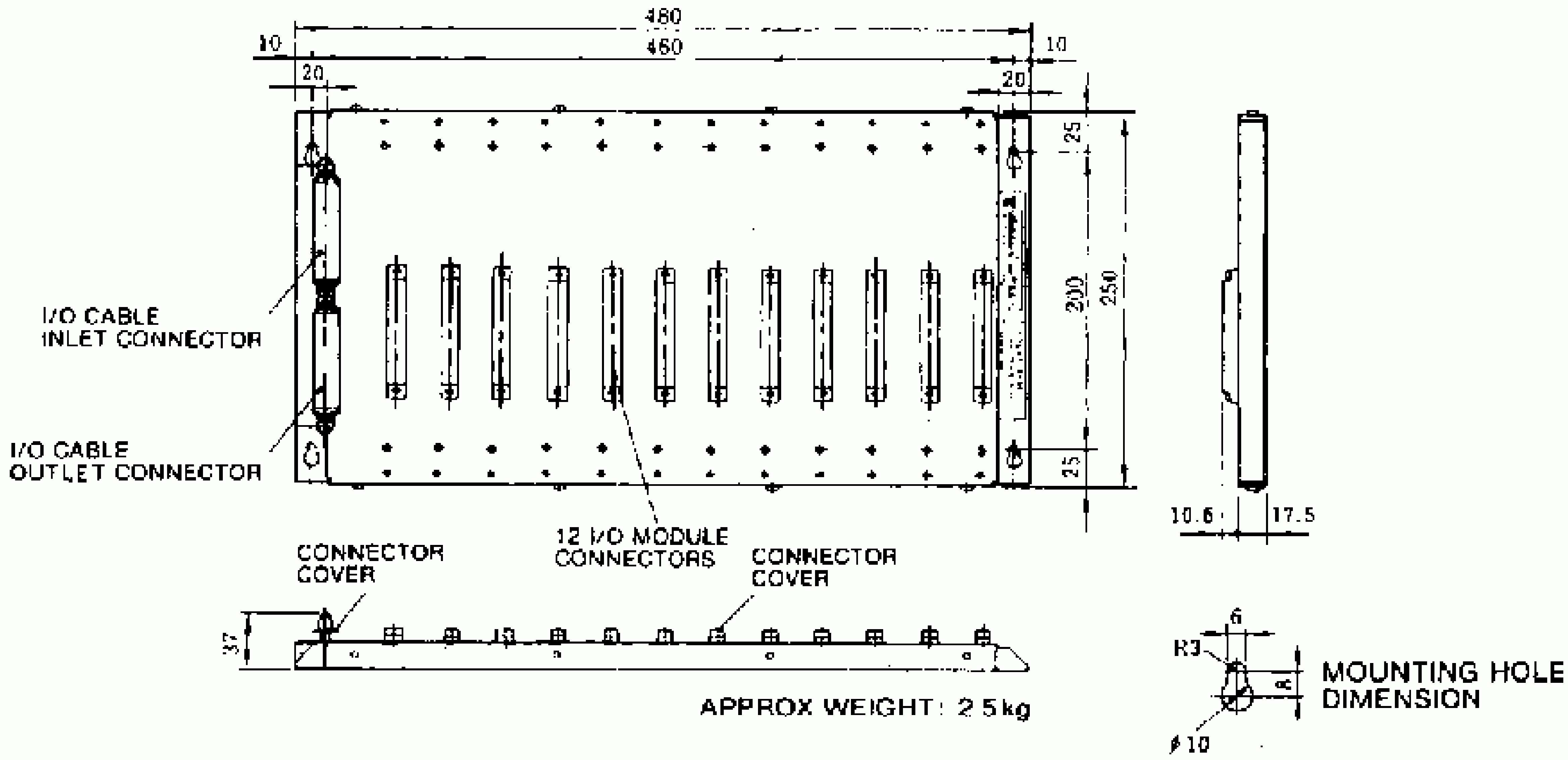
(16) Mounting Base Type JRMSI-B1033



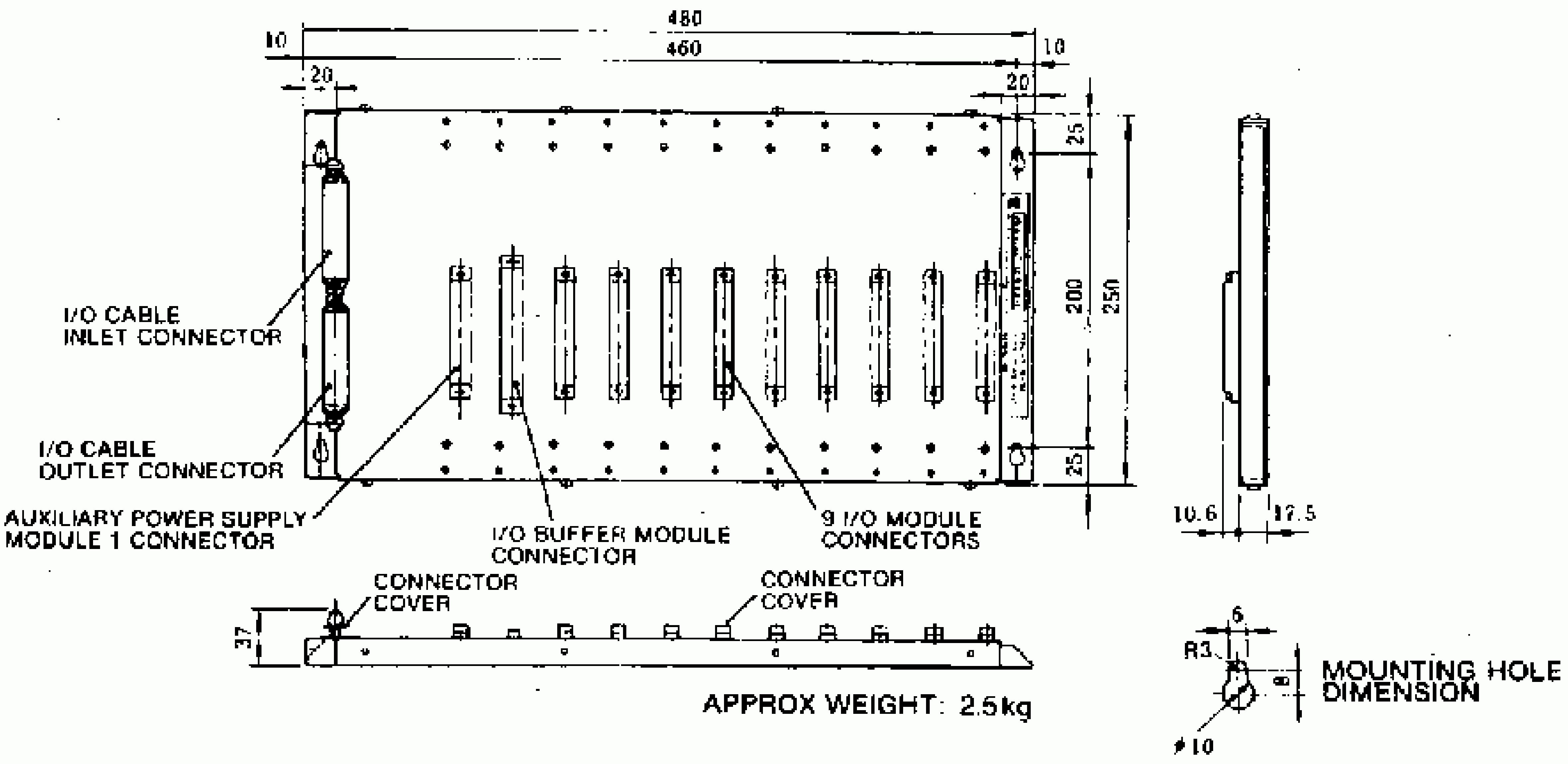
(17) Mounting Base Type JRMSI-B1034



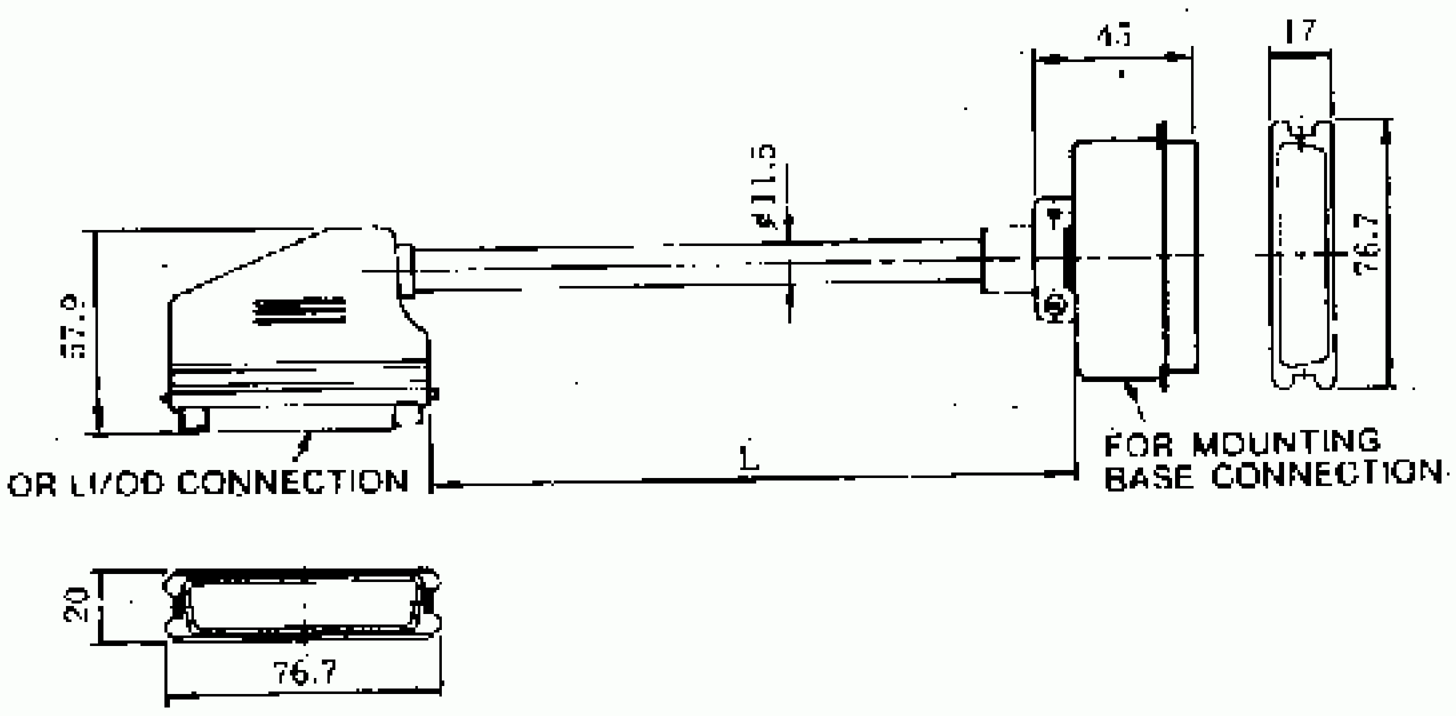
(18) Mounting Base Type JRMSI-B1027



(19) Mounting Base Type JRMSI-B1028

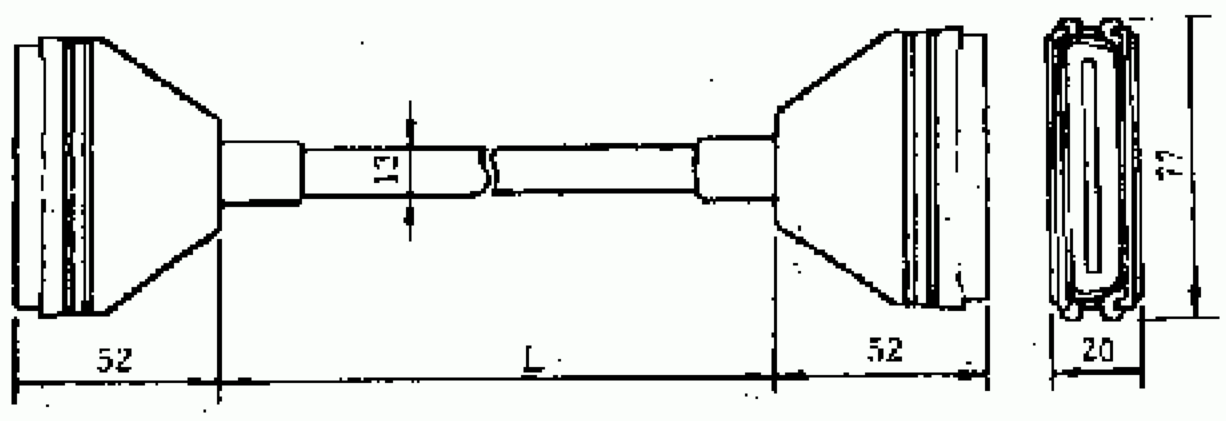


(20) I/O Cable Type JZMSZ-W1011



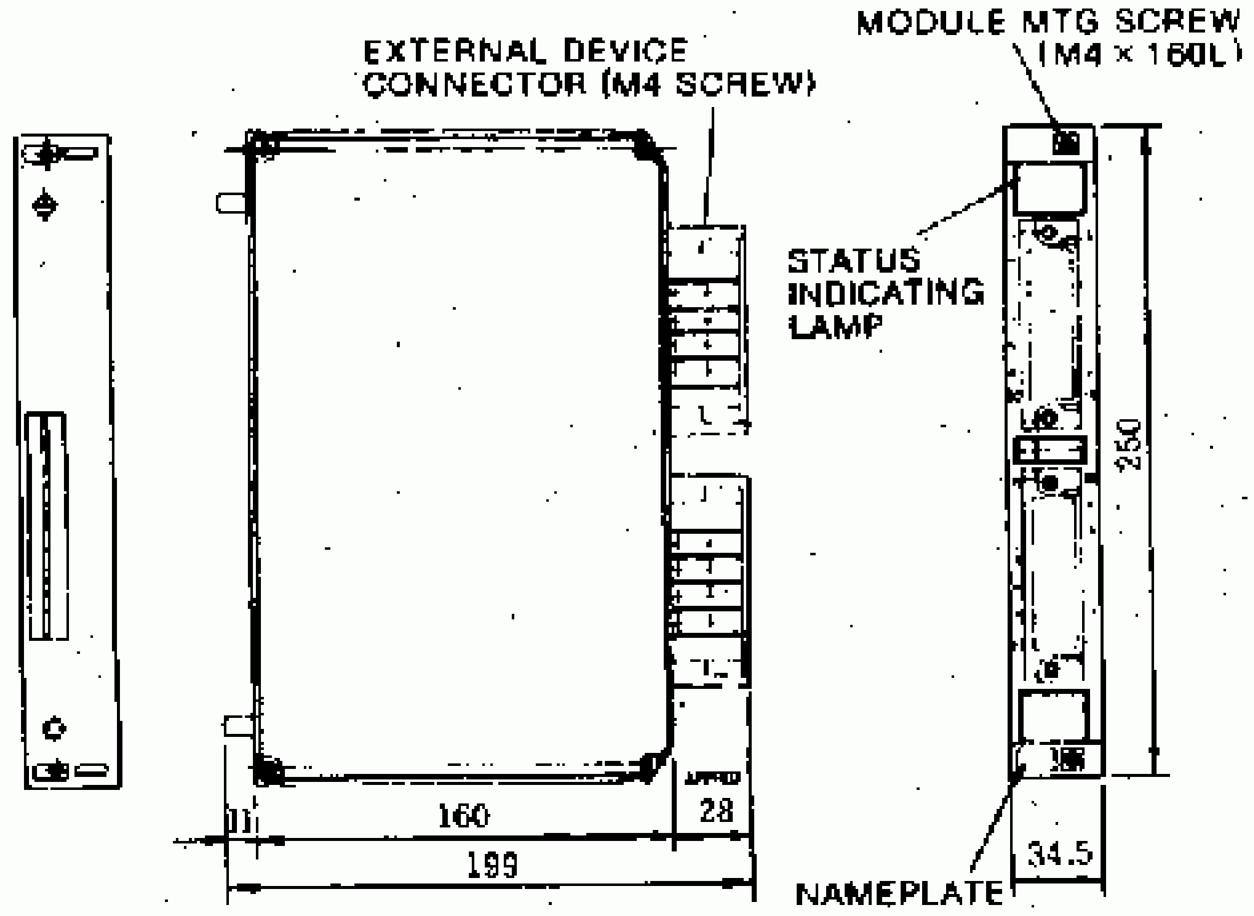
Type	Length L	Approx Weight
JZMSZ-W1011-1	1000	0.3kg
JZMSZ-W1011-2	5000	1.4kg

(21) I/O Cable Types JZMSZ-W1021 -W1022



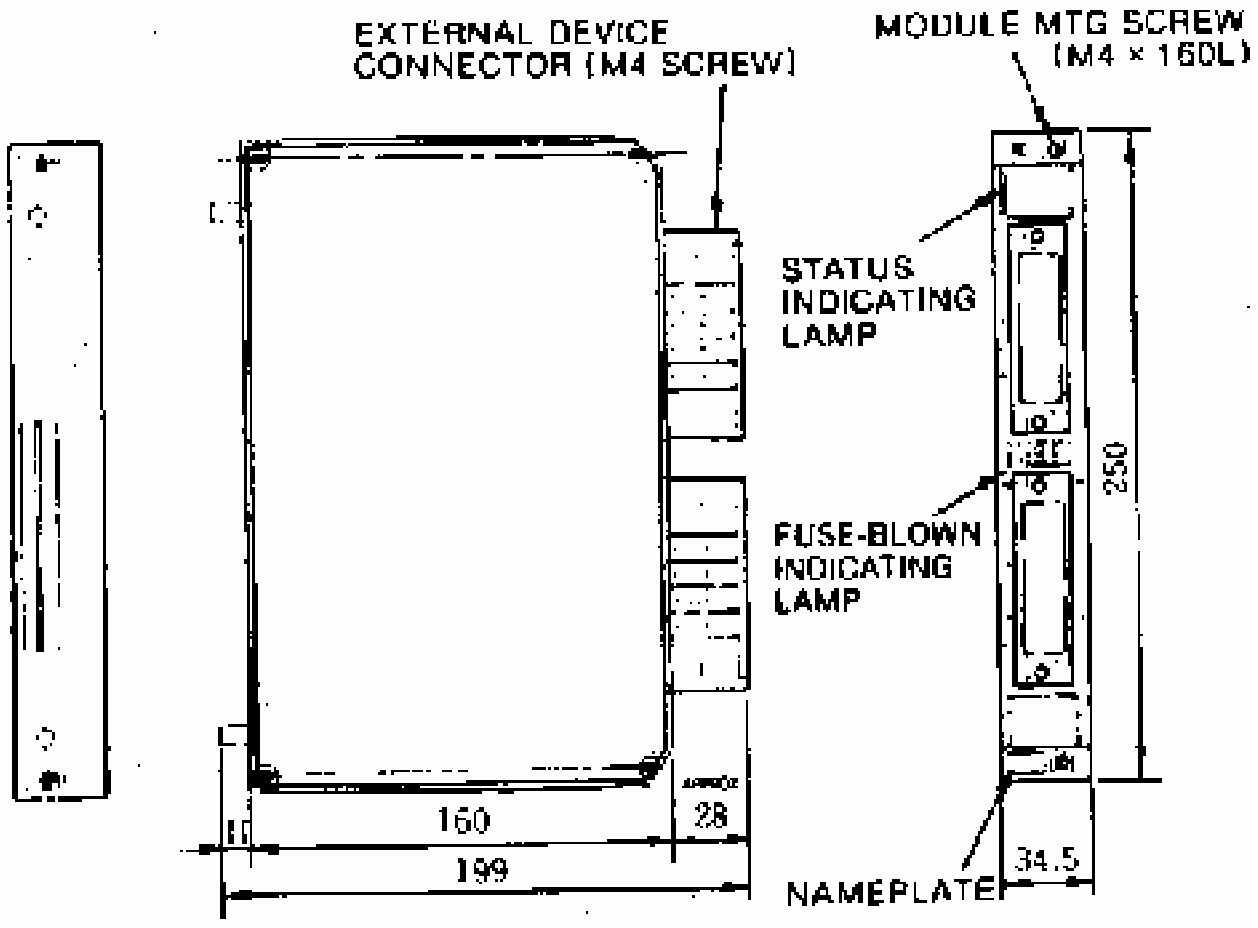
Type	Length L	Approx Weight
JZMSZ-W1021	1500	0.5kg
JZMSZ-W1022	400	0.3kg

(22) Input Module Types JAMSC-B1051B, -B1053, -B1055, -B1057, -B1059C



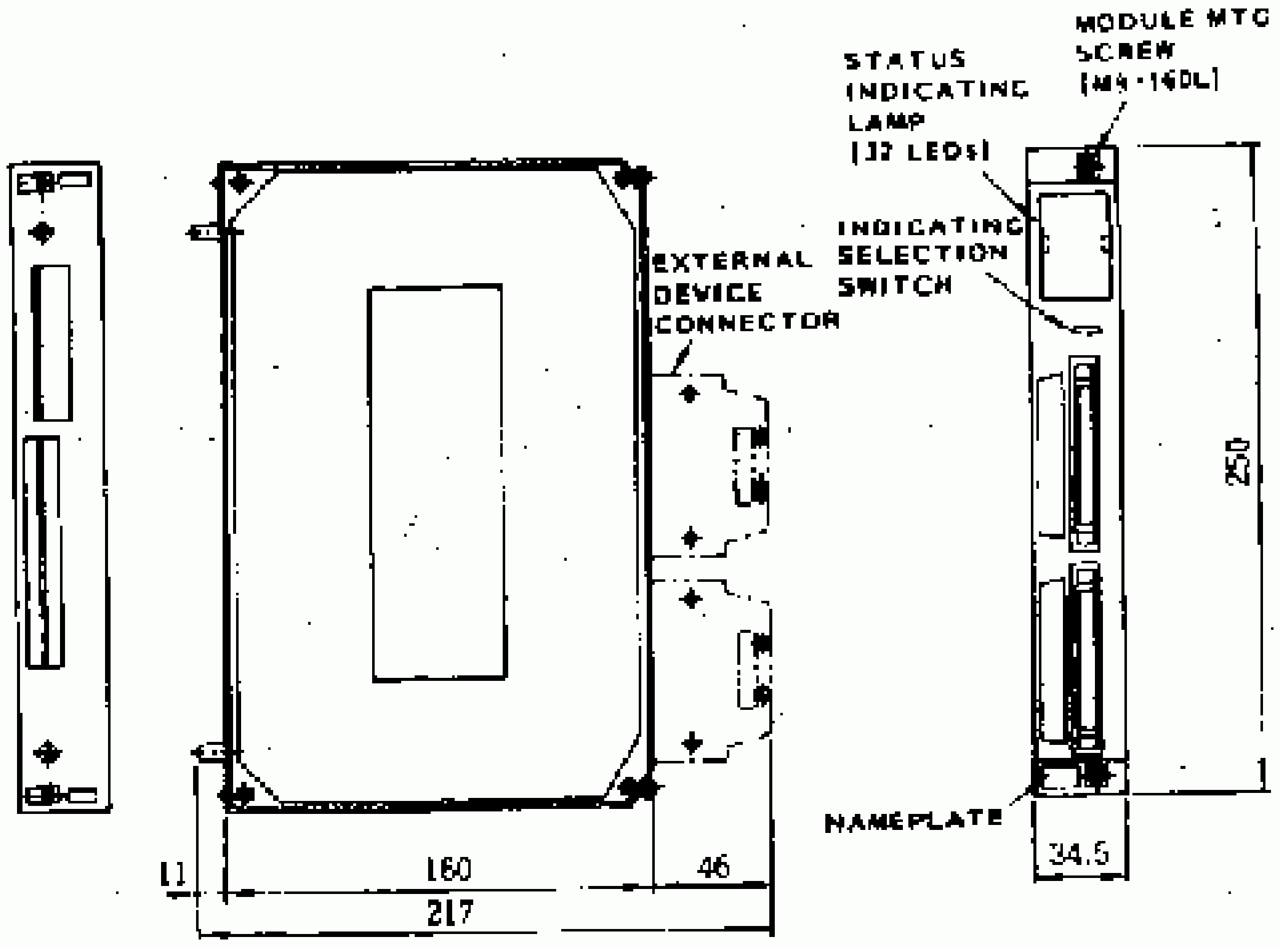
APPROX WEIGHT: 0.8kg

(23) Output Module Types JAMSC-B1050, -B1052, -B1054, -B1056, -B1058, -B1090B, -B1094



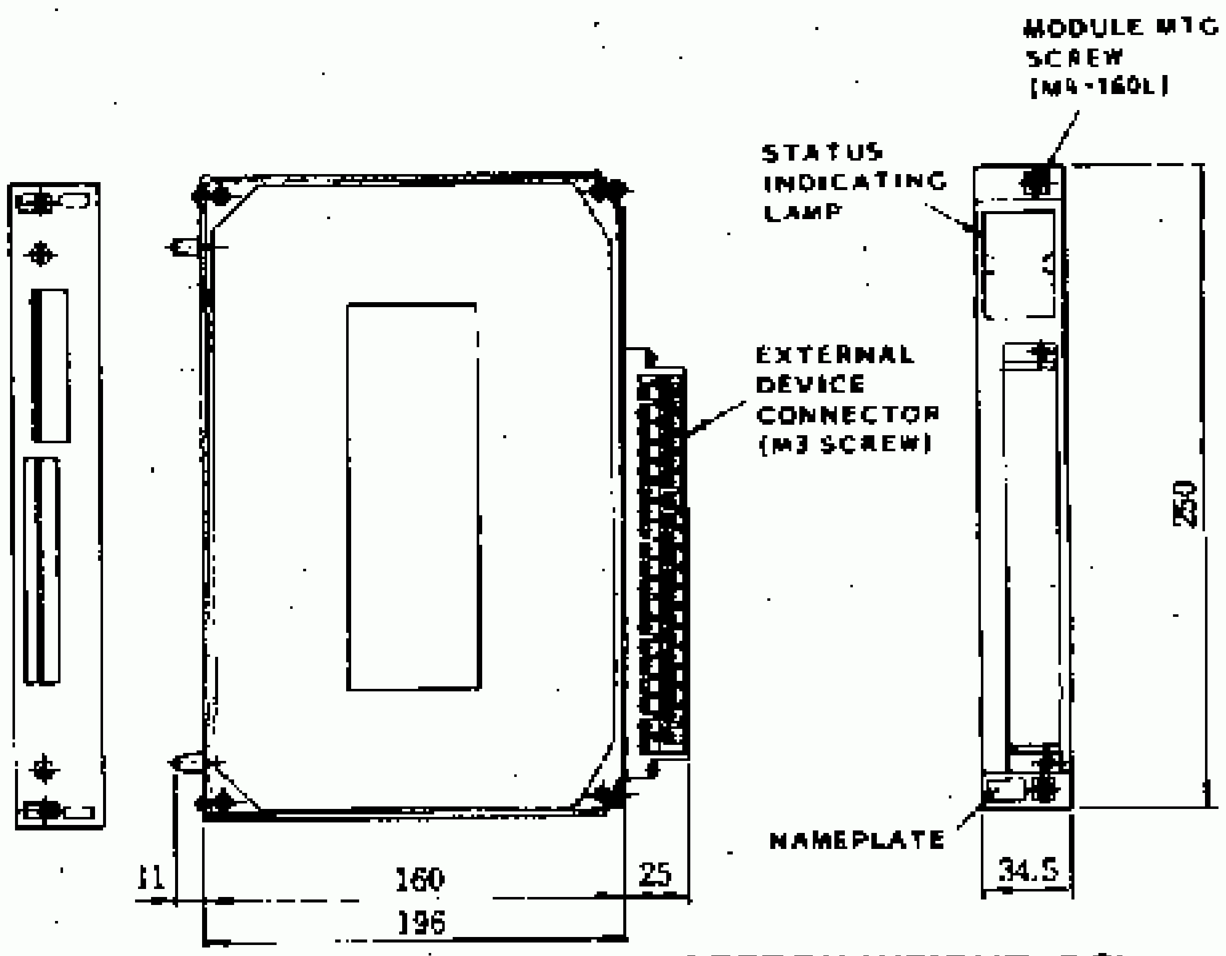
APPROX WEIGHT: 1.1kg

(24) Input Module Type JAMSC-B1061 Output Module Type JAMSC-B1060



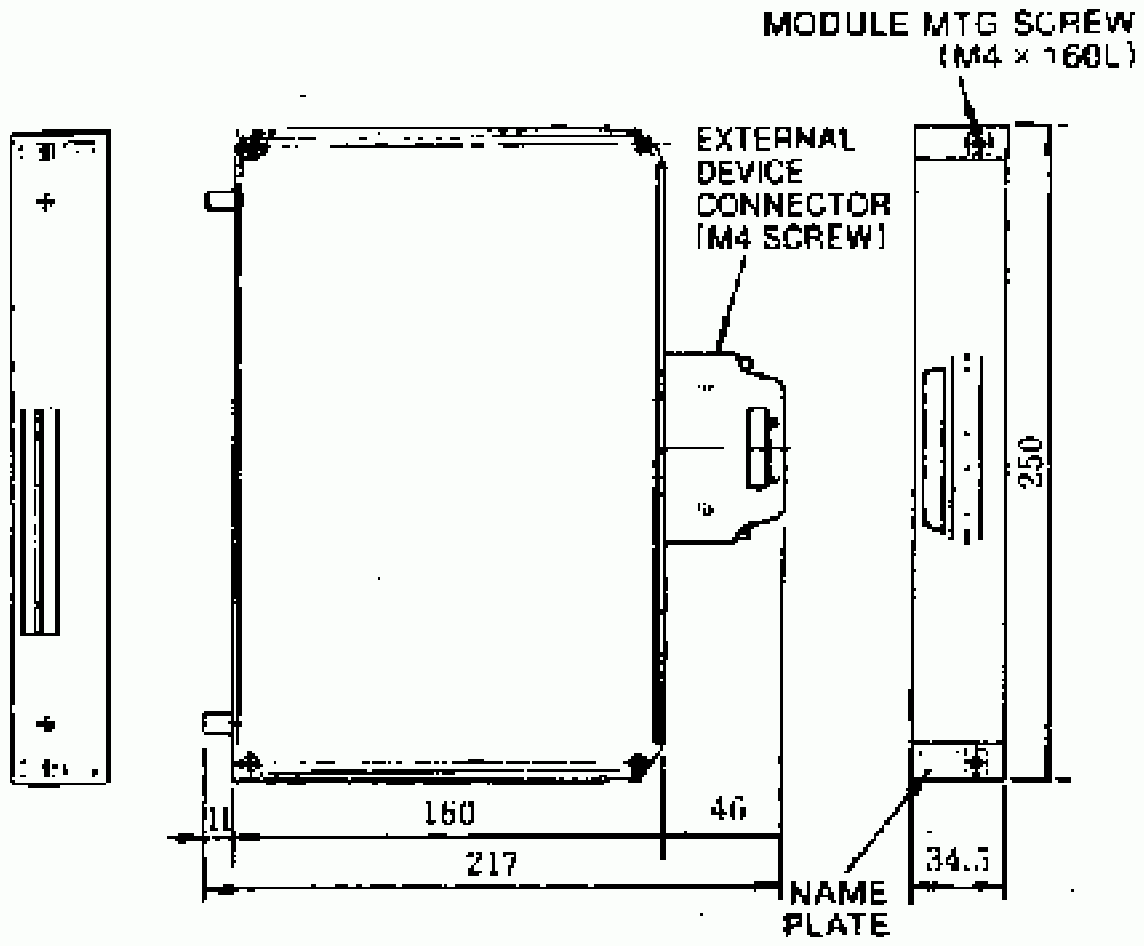
APPROX WEIGHT: 0.8 kg

(25) Input Module Type JAMSC-B1063 Output Module Type JAMSC-B1062



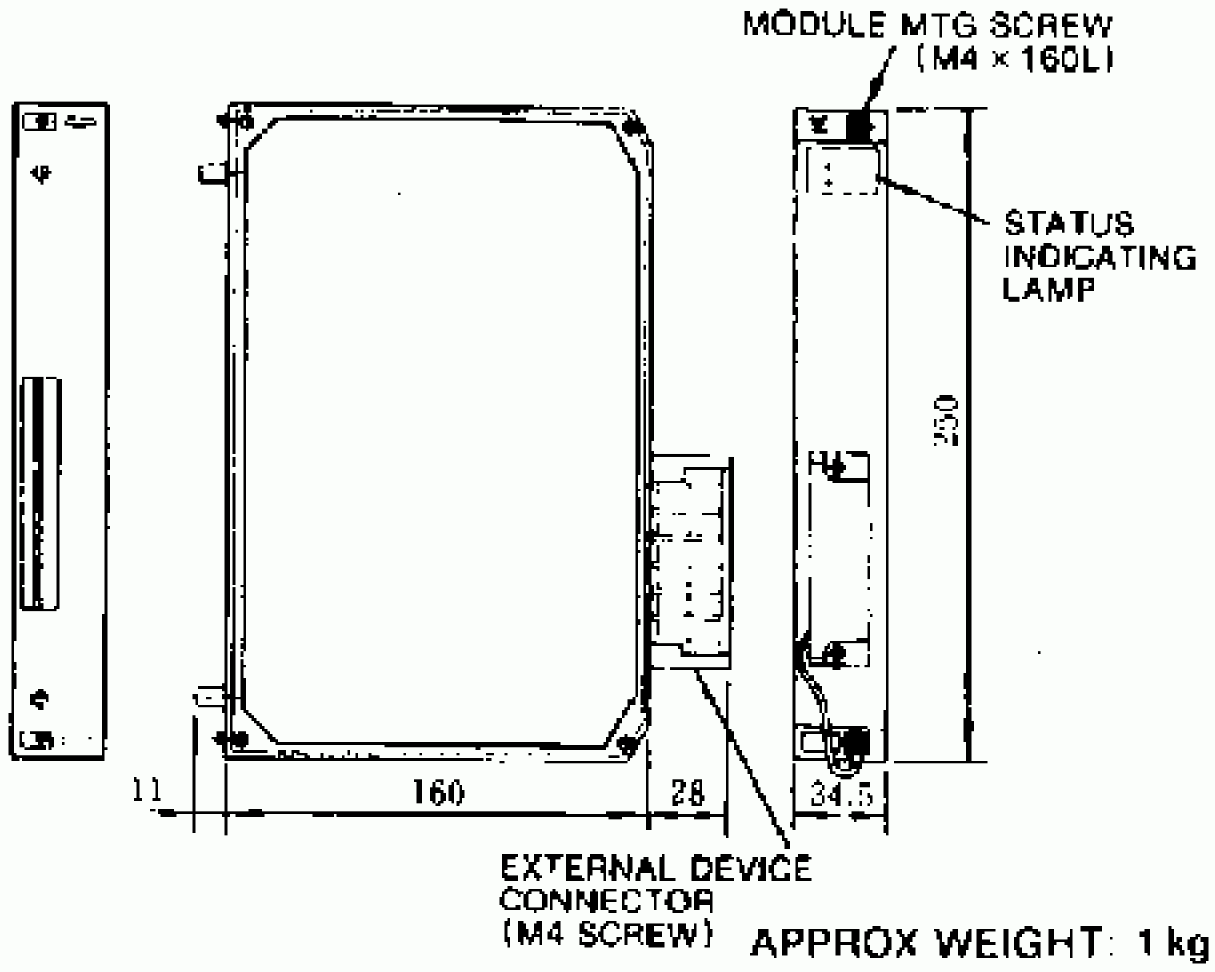
APPROX WEIGHT: 0.8kg

(26) Input Module Type JAMSC-B1065
Output Module Type JAMSC-B1064



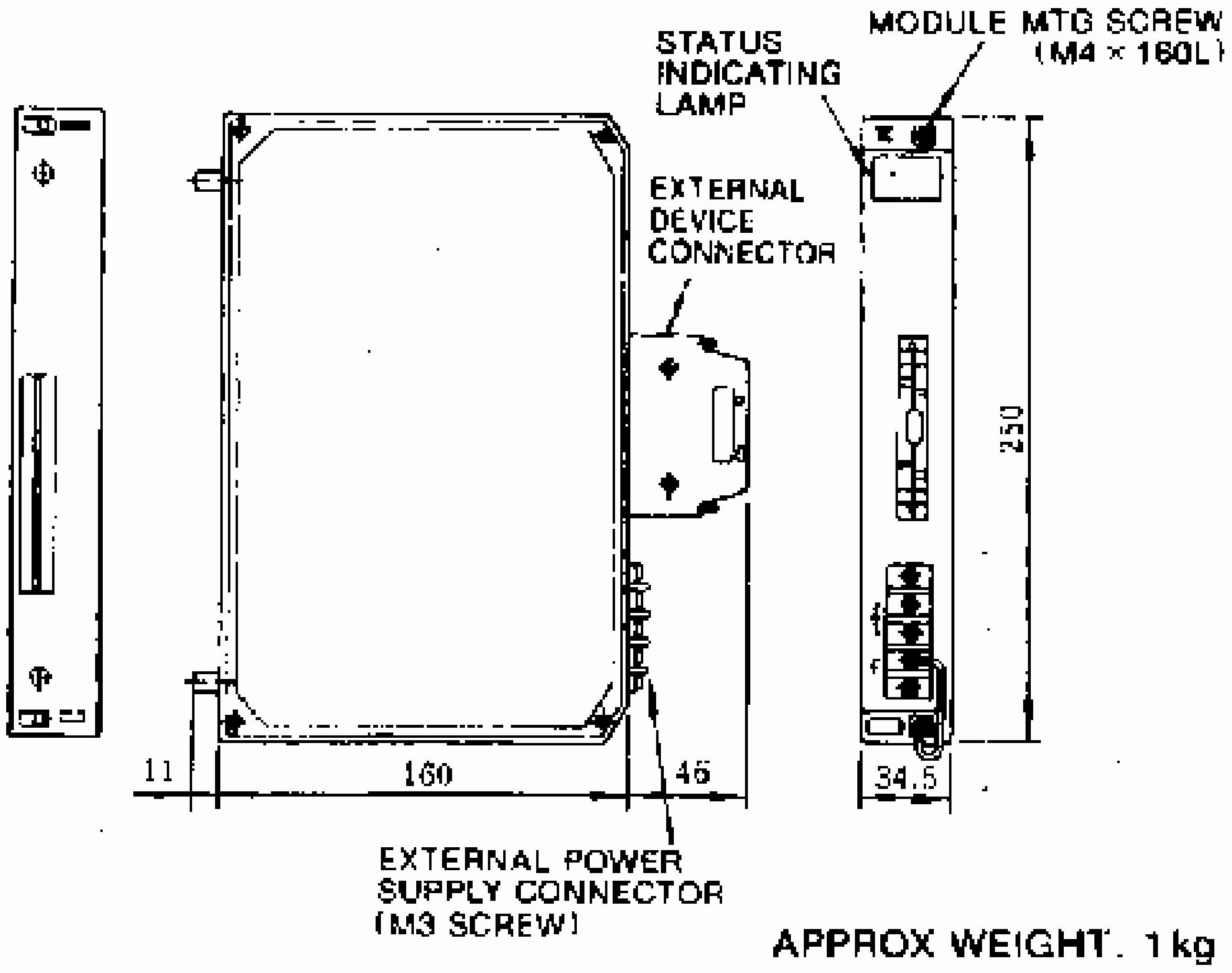
APPROX WEIGHT: 0.8kg

(29) Output Module Types JAMSC-B1072B-1, -B1072B-2, -B1072B-3, -B1072B-4, -B1074-1, -B1074-2, -B1074-3, -B1074-4



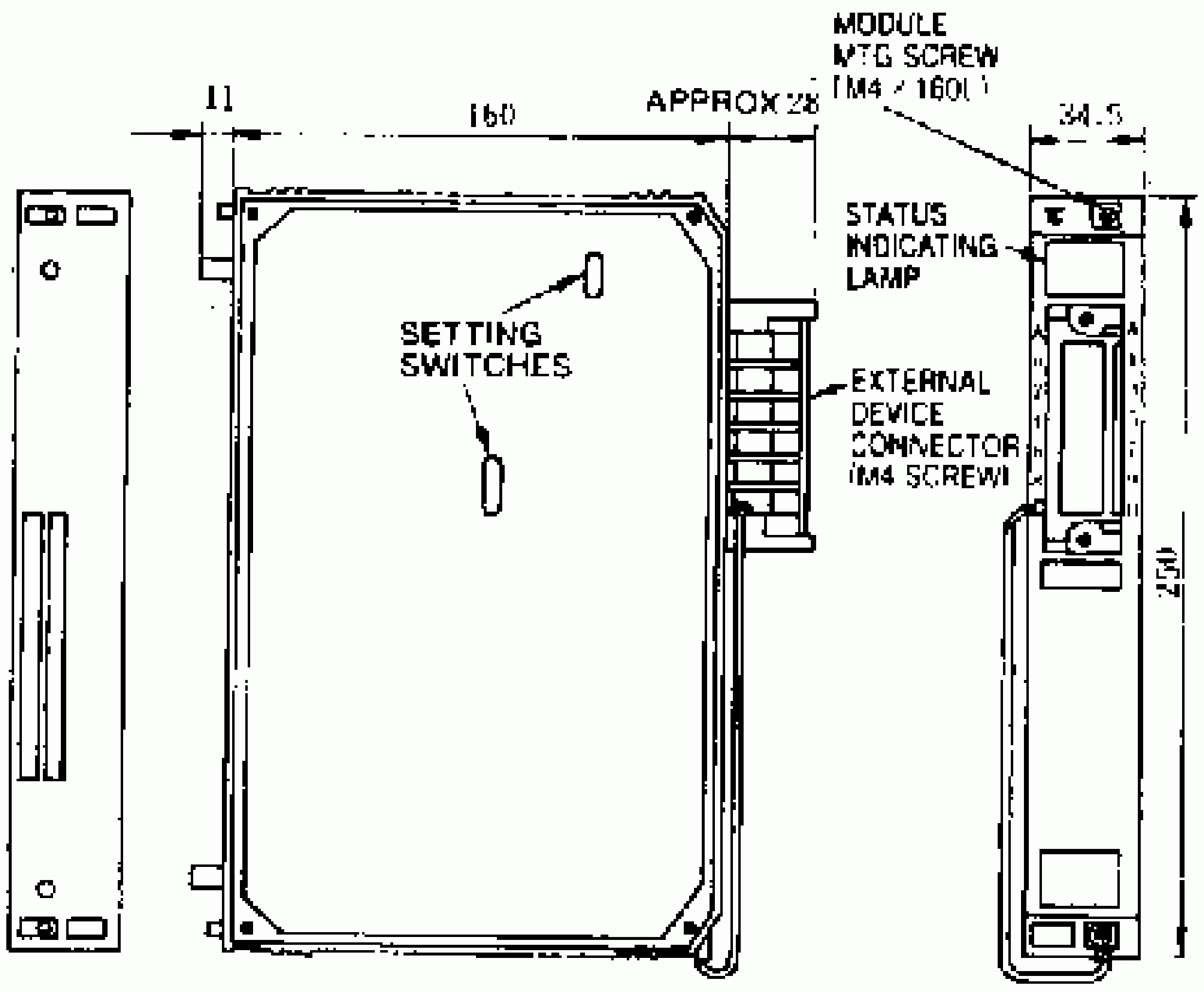
APPROX WEIGHT: 1kg

(27) Input Module Type JAMSC-B1071
Output module Type JAMSC-B1070



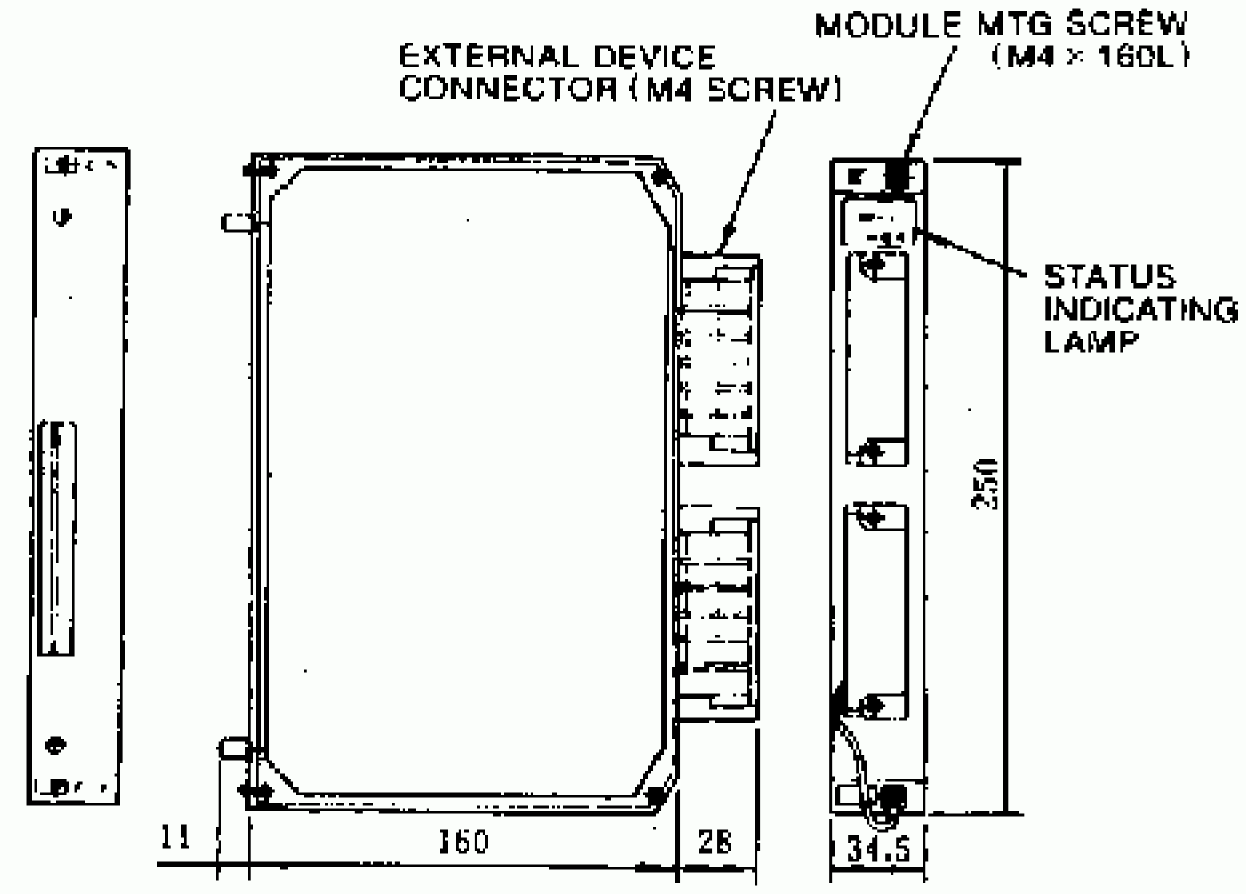
APPROX WEIGHT: 1kg

(30) Reversible Counter Module Type JAMSC-B1081C



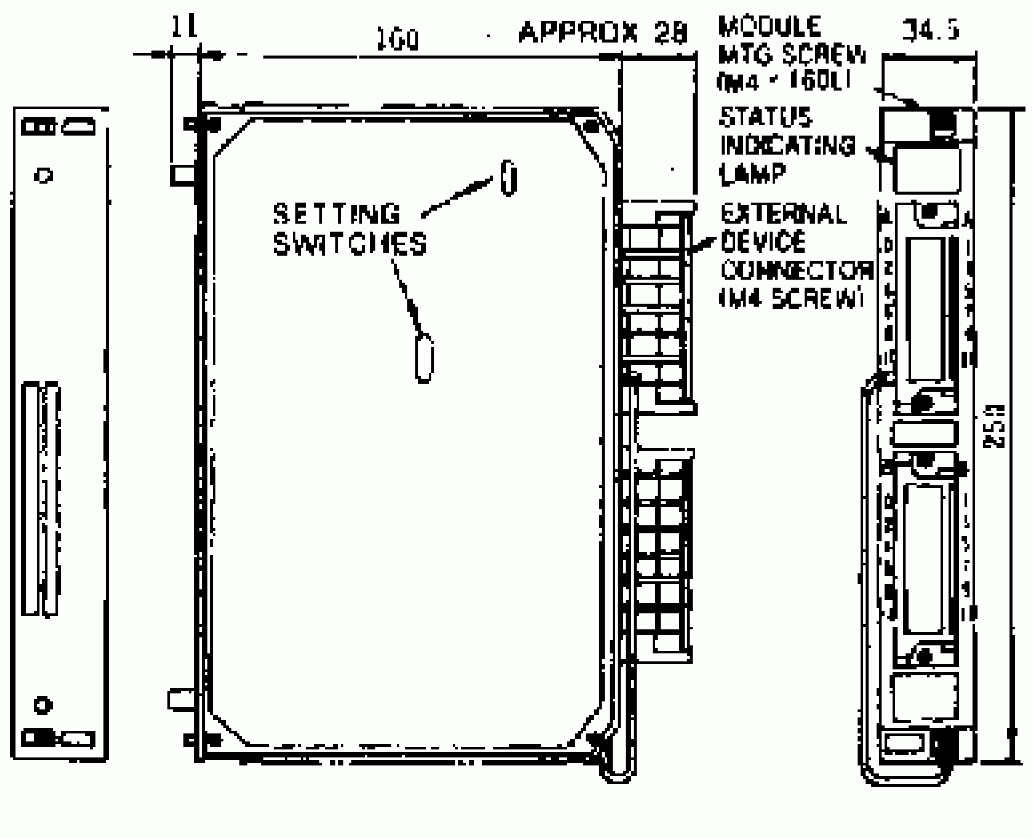
APPROX WEIGHT: 0.8kg

(28) Input Module Types JAMSC-B1073-1, -B1073-2, -B1075-1, -B1075-2



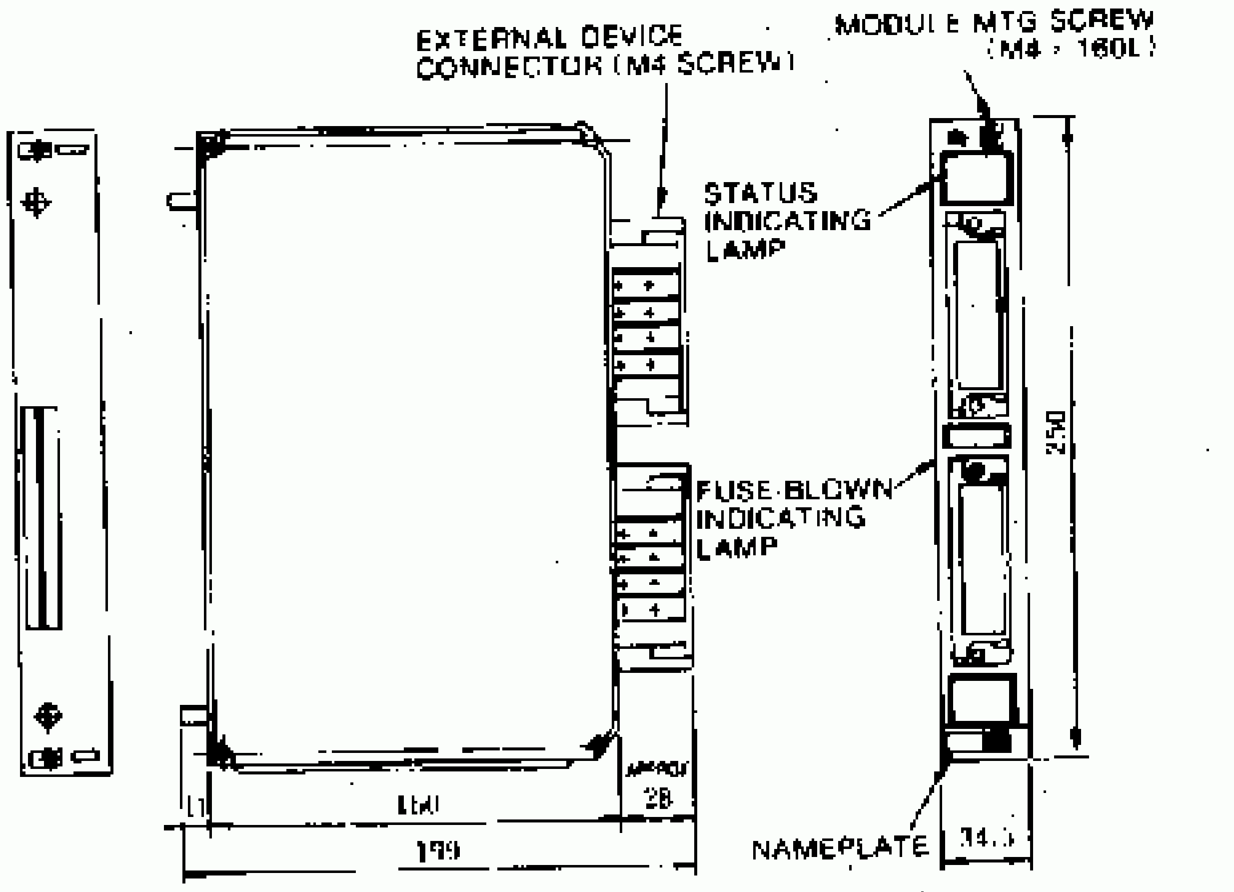
APPROX WEIGHT: 1kg

(31) Preset Counter Module Type JAMSC-B1082C



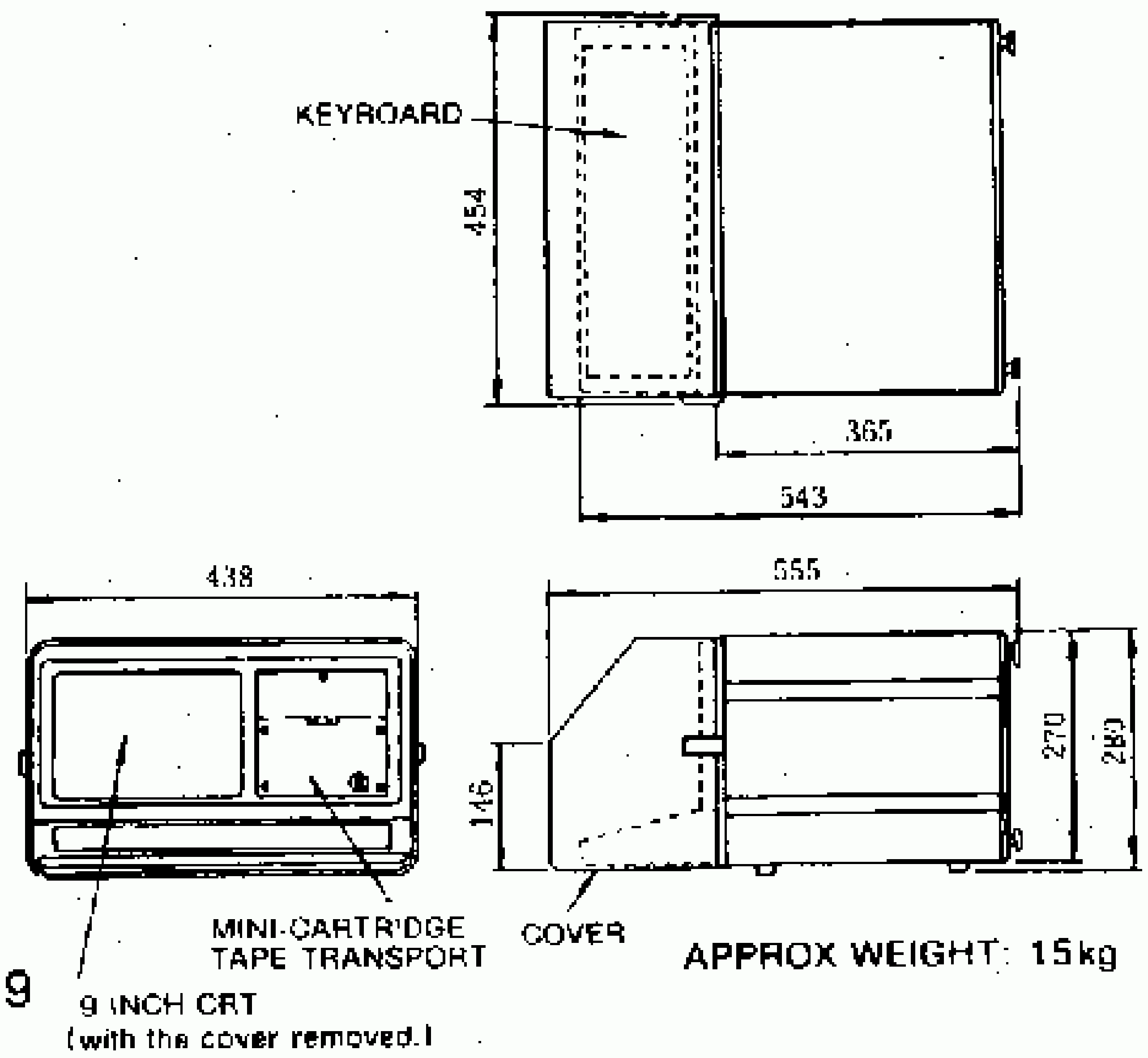
APPROX WEIGHT: 0.9kg

(32) PID Module Type JAMSC-B1080



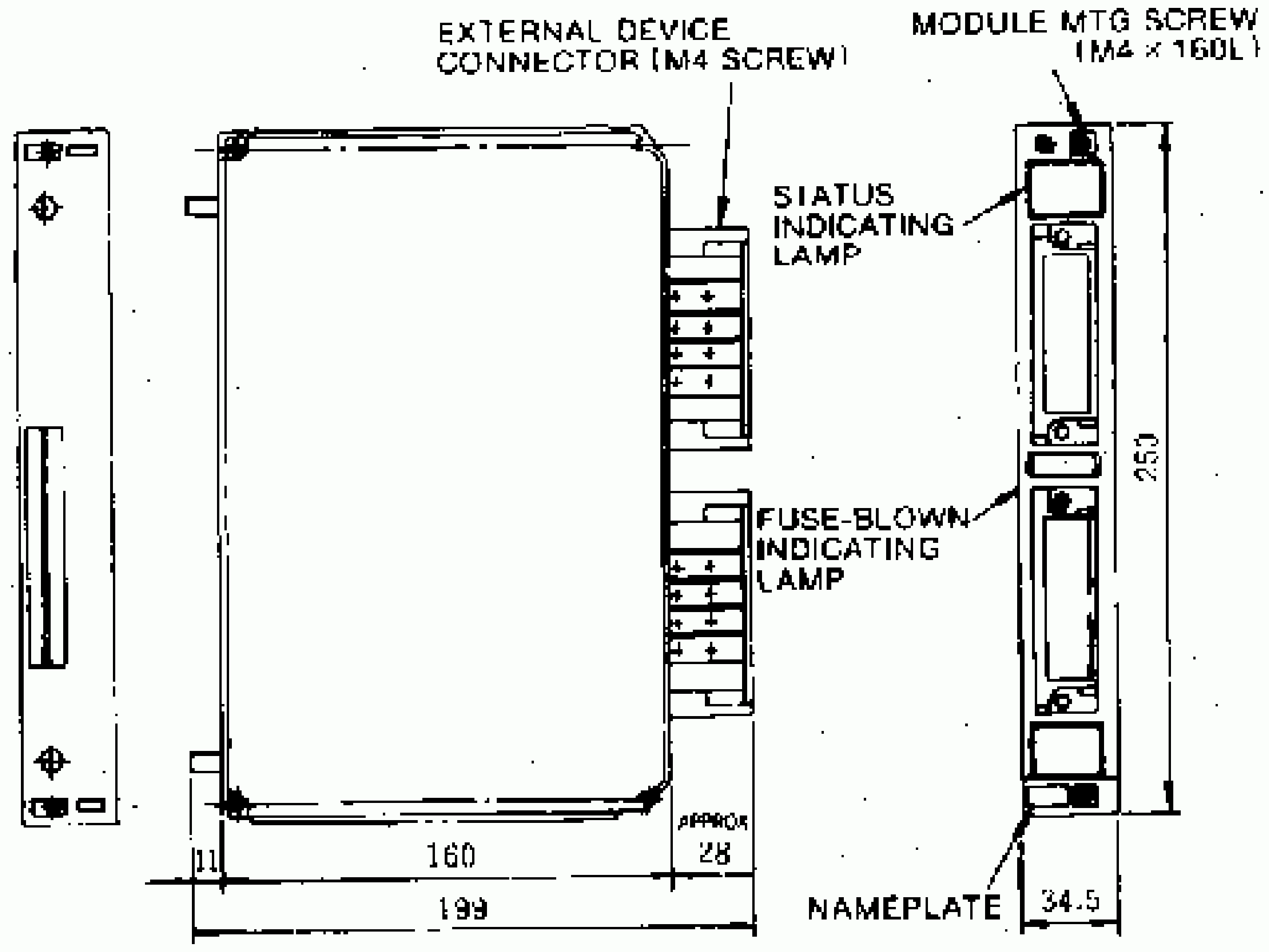
APPROX WEIGHT: 1.0kg

(35) P190 Programming Panel Type DISCT-P190



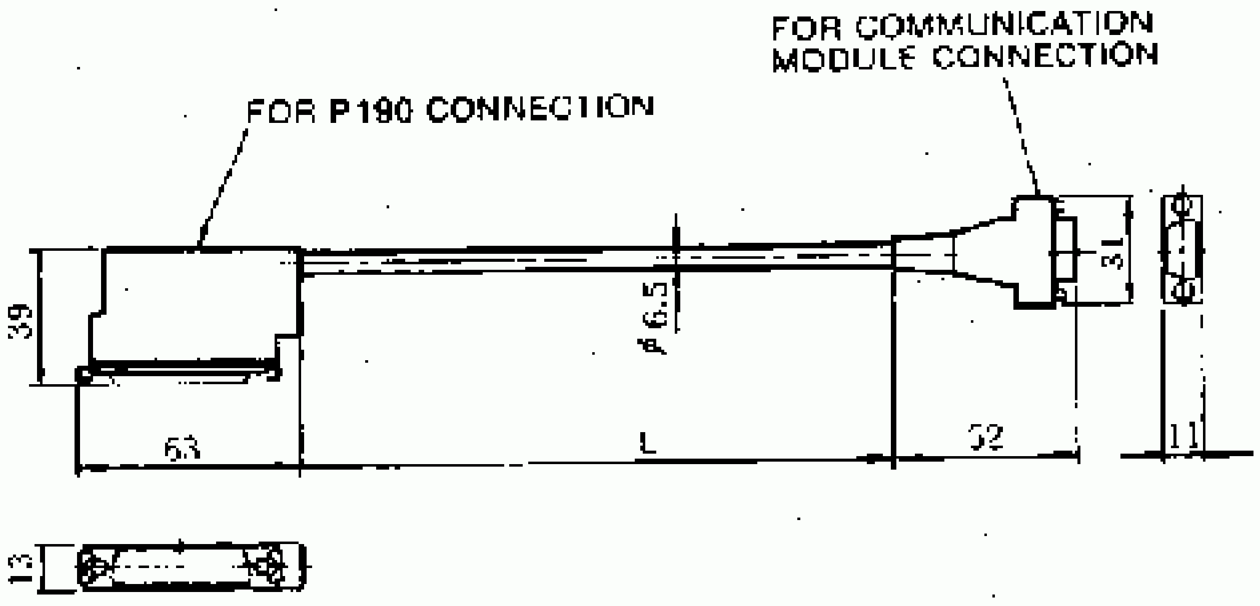
APPROX WEIGHT: 15kg

(33) Power Supply Module Type JAMSC-B1089



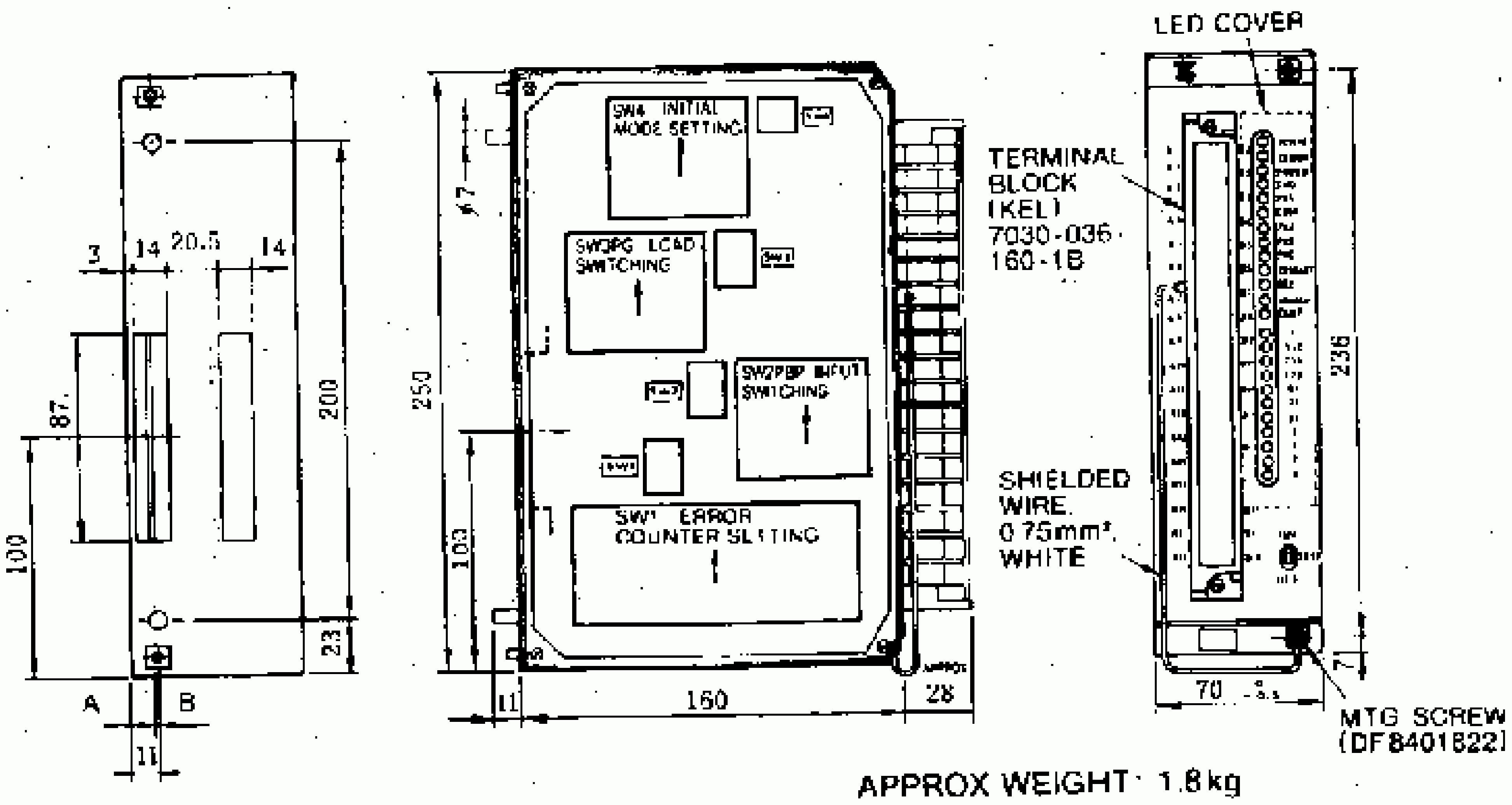
APPROX WEIGHT: 1.0 kg

(36) Interface Cable Type JZMSZ-W1015



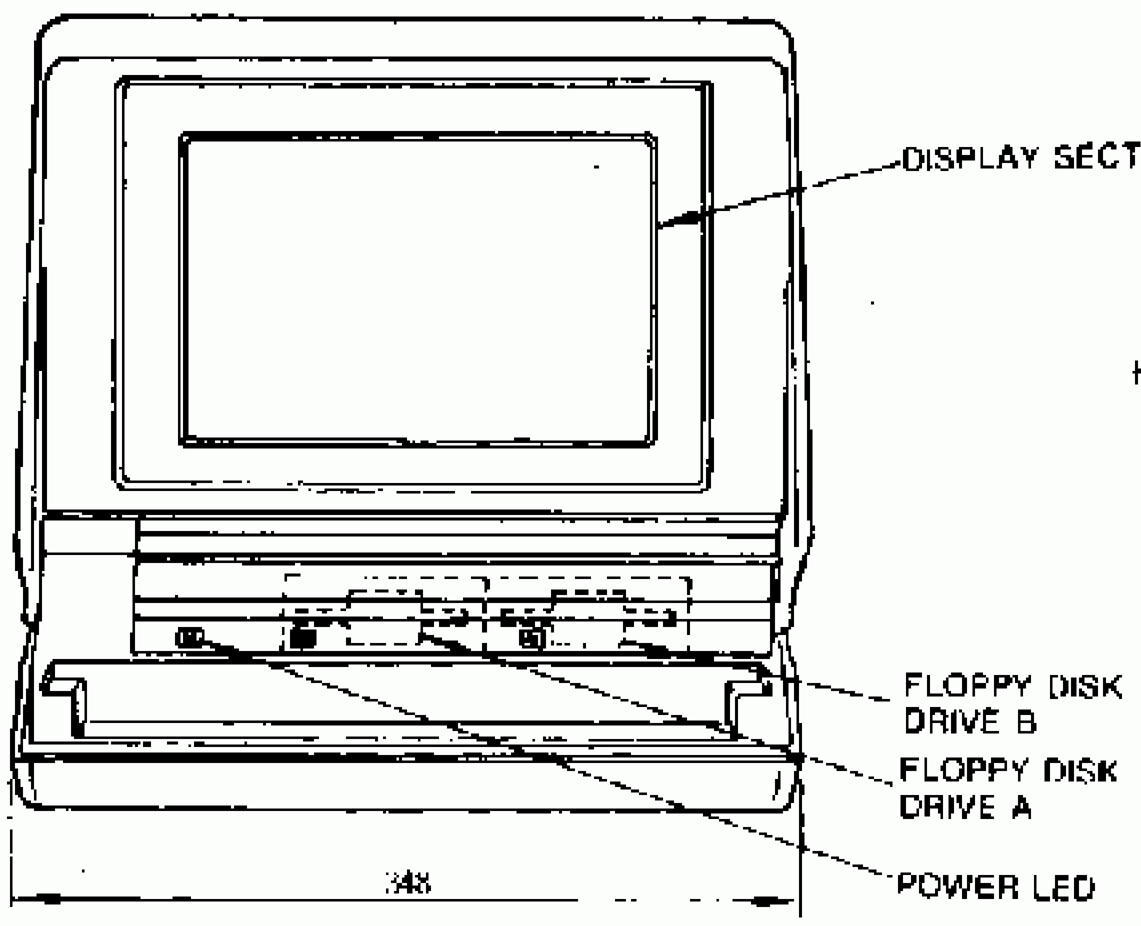
Type	Length L	Approx Weight
JZMSZ-W1015-1	2500	0.2 kg
JZMSZ-W1015-2	15000	1.0 kg

(34) Positioning Module Type JAMSC-B1083C

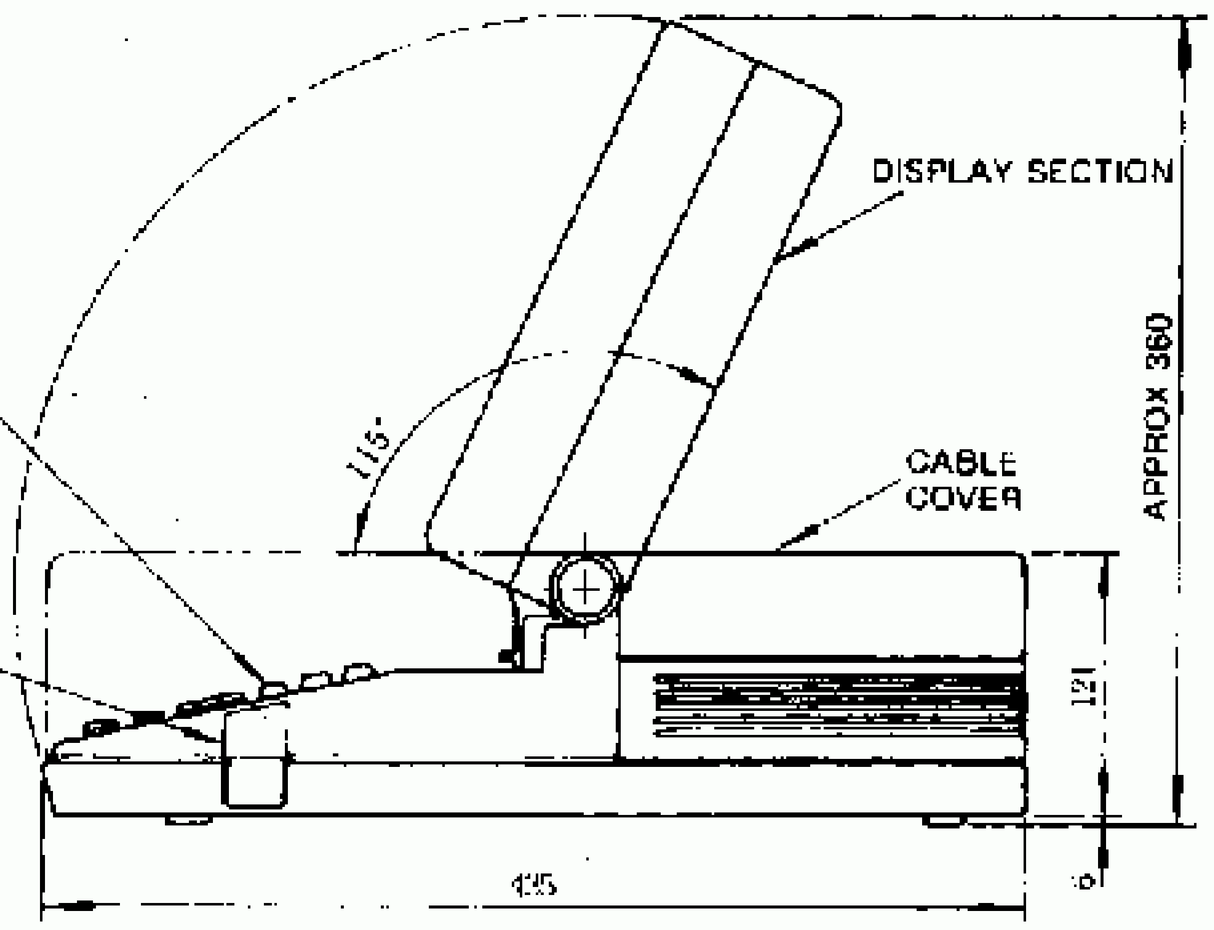


APPROX WEIGHT: 1.8kg

(37) P150 Programming Panel Type DISCT-P150

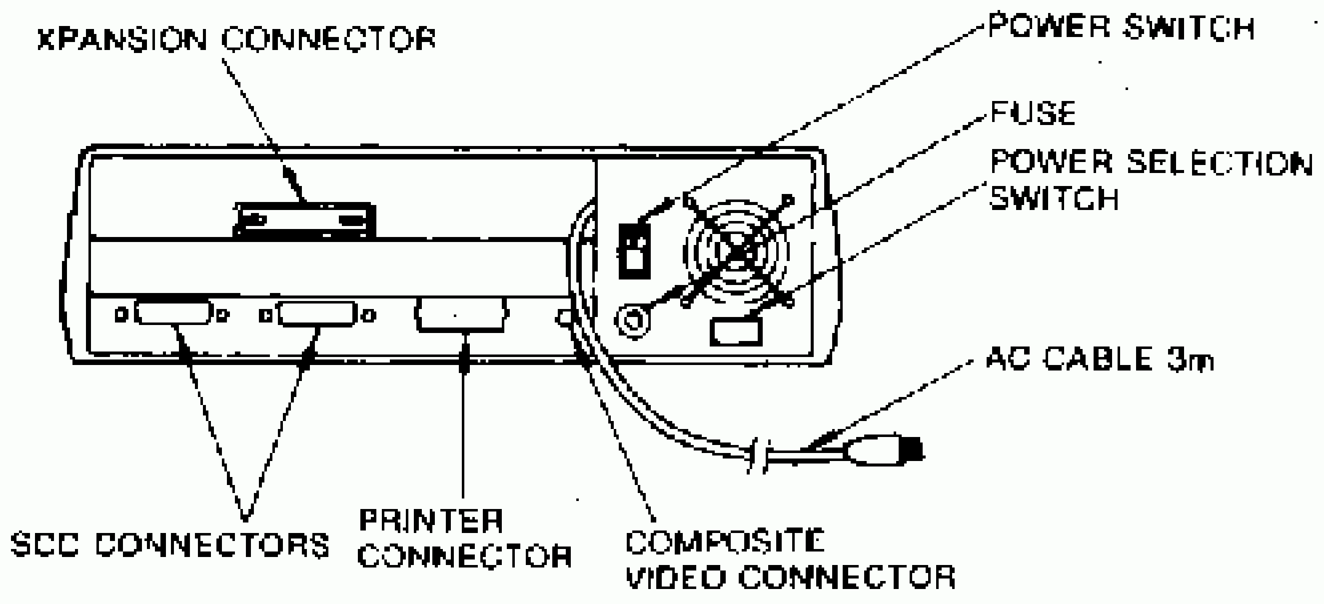


Front View



Side View

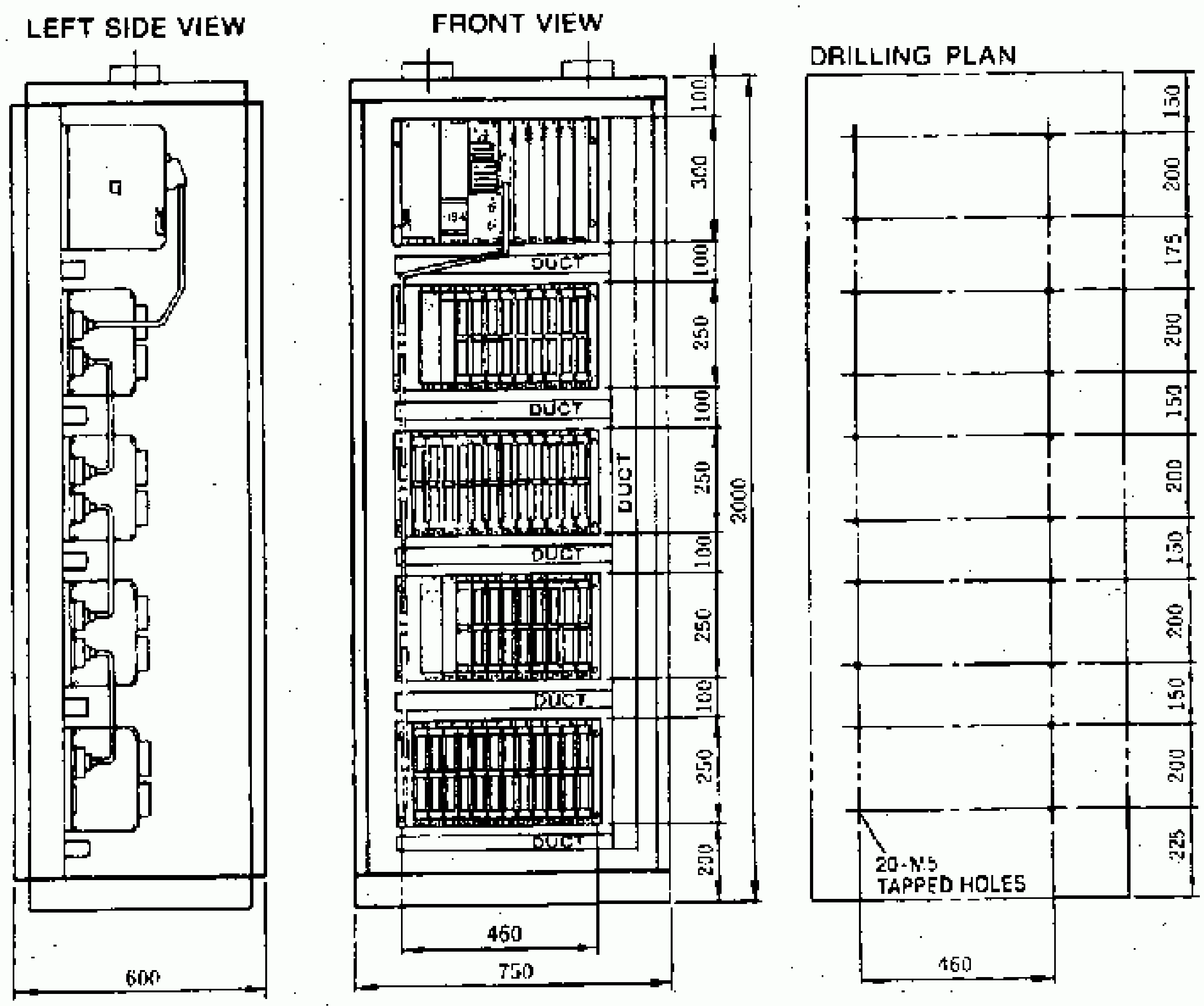
APPROX WEIGHT: 9kg



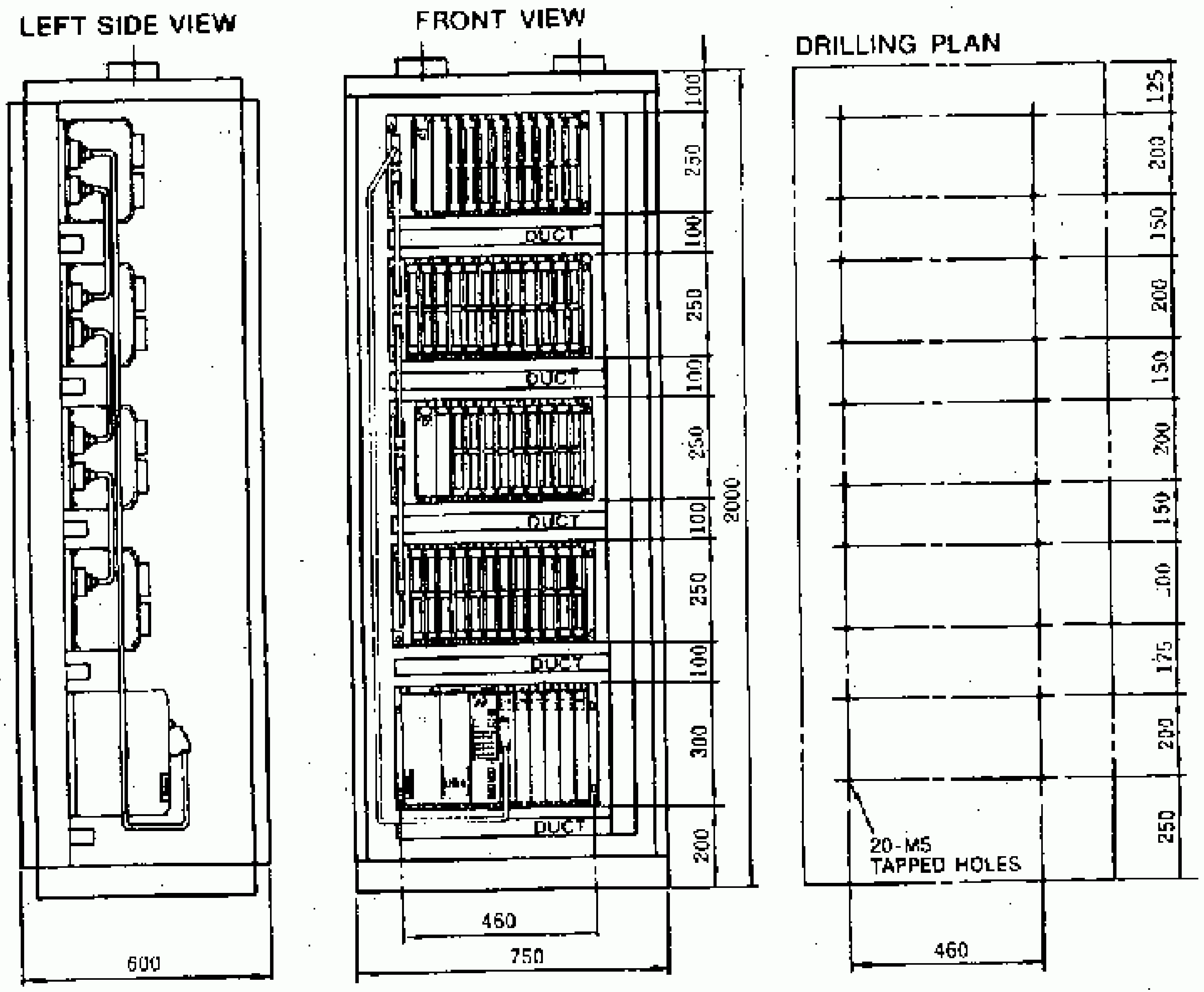
Rear View (With Display Section Closed)

APPENDIX C
U84 LAYOUT AND DRILLING PLAN in mm

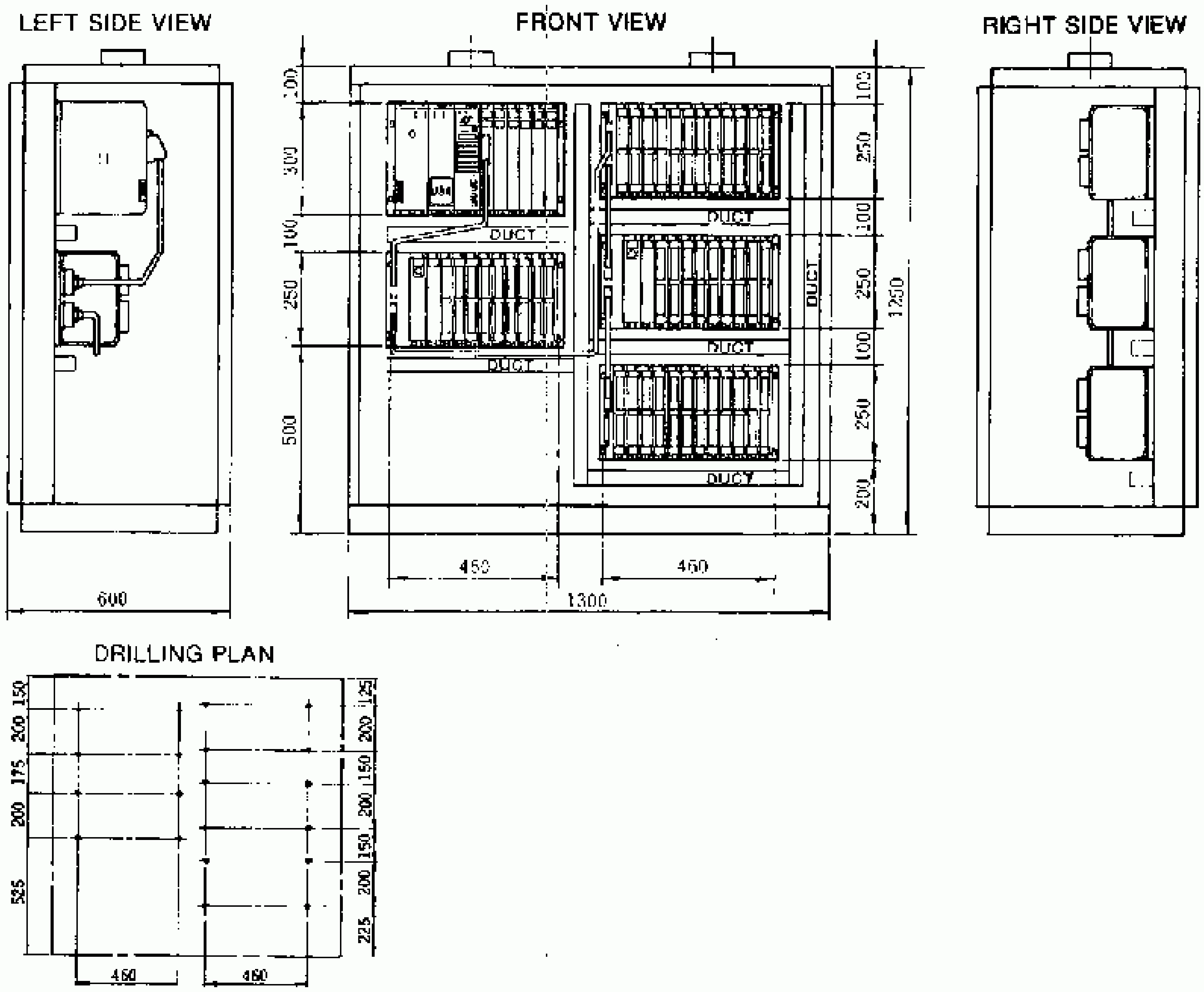
Example 1



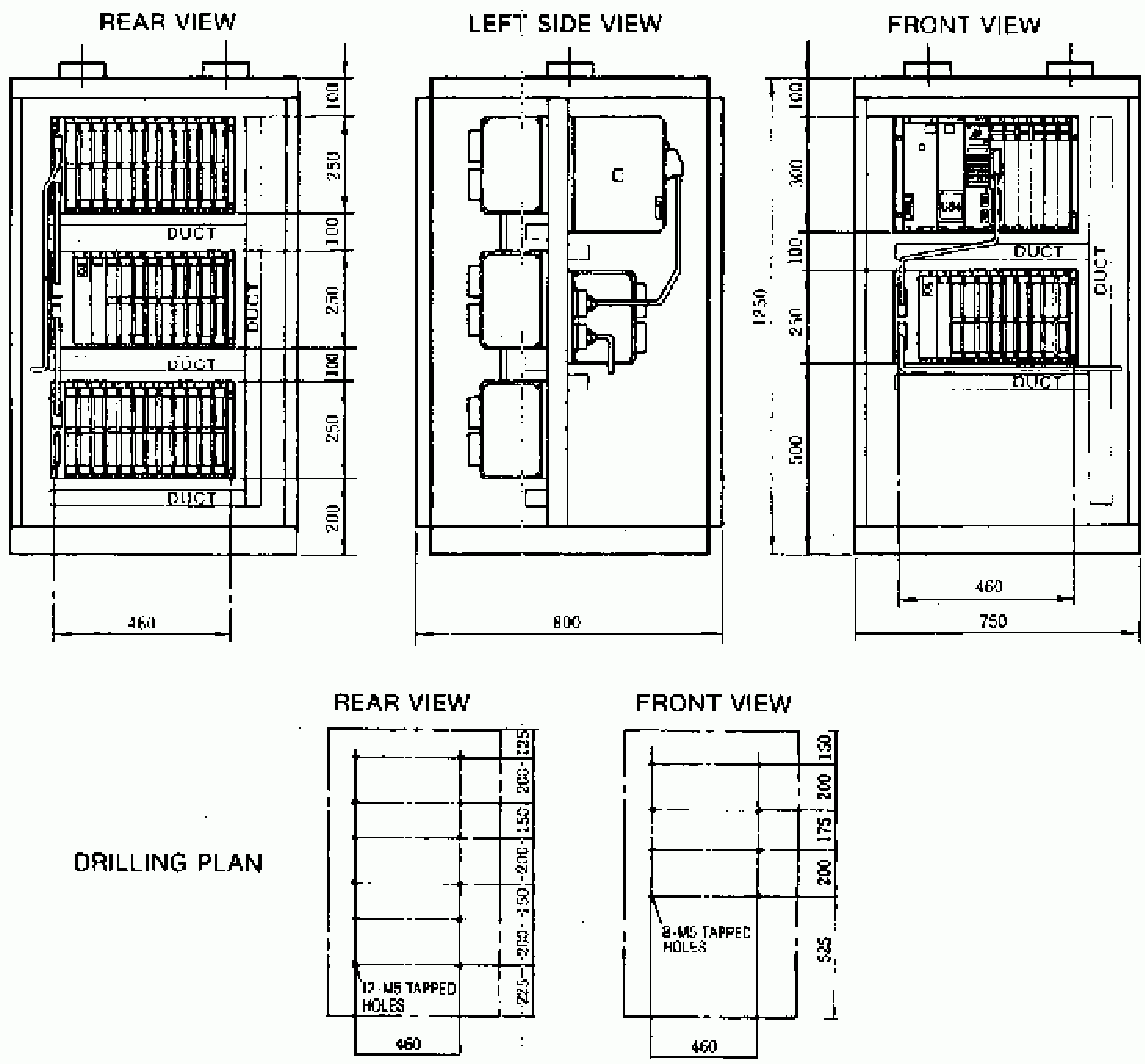
Example 2



Example 3



Example 4



MEMOCON-SC U84
DESCRIPTIVE INFORMATION

TOKYO OFFICE
New Pier Takeshiba South Tower, 1-16-1, Kaigan, Minatoku, Tokyo 105-6891 Japan
Phone 81-3-5402-4511 Fax 81-3-5402-4580

YASKAWA ELECTRIC AMERICA, INC.
2121 Norman Drive South, Waukegan, IL 60085, U.S.A.
Phone 1-847-887-7000 Fax 1-847-887-7370

MOTOMAN INC. HEADQUARTERS
805 Liberty Lane West Carrollton, OH 45449, U.S.A.
Phone 1-937-847-6200 Fax 1-937-847-6277

YASKAWA ELÉTRICO DO BRASIL COMÉRCIO LTDA.
Avenida Fagundes Filho, 620 Bairro Saude-Sao Päulo-SP, Brazil CEP: 04304-000
Phone 55-11-5071-2552 Fax 55-11-5581-8795

YASKAWA ELECTRIC EUROPE GmbH
Am Kronberger Hang 2, 65824 Schwalbach, Germany
Phone 49-6196-569-300 Fax 49-6196-888-301

Motoman Robotics Europe AB
Box 504 S38526 Torsås, Sweden
Phone 46-486-48800 Fax 46-486-41410

Motoman Robotec GmbH
Kammerfeldstraße 1, 85391 Allershausen, Germany
Phone 49-8166-900 Fax 49-8166-9039

YASKAWA ELECTRIC UK LTD.
1 Hunt Hill Orchardton Woods Cumbernauld, G68 9LF, United Kingdom
Phone 44-1236-735000 Fax 44-1236-458182

YASKAWA ELECTRIC KOREA CORPORATION
Kfpa Bldg #1201, 35-4 Youido-dong, Yeongdungpo-Ku, Seoul 150-010, Korea
Phone 82-2-784-7844 Fax 82-2-784-8495

YASKAWA ELECTRIC (SINGAPORE) PTE. LTD.
151 Lorong Chuan, #04-01, New Tech Park Singapore 556741, Singapore
Phone 65-282-3003 Fax 65-289-3003

YASKAWA ELECTRIC (SHANGHAI) CO., LTD.
4F No.18 Aona Road, Waigaoqiao Free Trade Zone, Pudong New Area, Shanghai 200131, China
Phone 86-21-5866-3470 Fax 86-21-5866-3869

YATEC ENGINEERING CORPORATION
Shen Hsiang Tang Sung Chiang Building 10F 146 Sung Chiang Road, Taipei, Taiwan
Phone 886-2-2563-0010 Fax 886-2-2567-4677

YASKAWA ELECTRIC (HK) COMPANY LIMITED
Rm. 2909-10, Hong Kong Plaza, 186-191 Connaught Road West, Hong Kong
Phone 852-2803-2385 Fax 852-2547-5773

BEIJING OFFICE
Room No. 301 Office Building of Beijing International Club, 21
Jianguomenwai Avenue, Beijing 100020, China
Phone 86-10-6532-1850 Fax 86-10-6532-1851

TAIPEI OFFICE
Shen Hsiang Tang Sung Chiang Building 10F 146 Sung Chiang Road, Taipei, Taiwan
Phone 886-2-2563-0010 Fax 886-2-2567-4677

SHANGHAI YASKAWA-TONGJI M & E CO., LTD.
27 Hui He Road Shanghai China 200437
Phone 86-21-6531-4242 Fax 86-21-6553-6060

BEIJING YASKAWA BEIKE AUTOMATION ENGINEERING CO., LTD.
30 Xue Yuan Road, Haidian, Beijing P.R. China Post Code: 100083
Phone 86-10-6293-2782 Fax 86-10-6232-1536

SHOUGANG MOTOMAN ROBOT CO., LTD.
7, Yongchang-North Street, Beijing Economic Technological Investment & Development Area,
Beijing 100076, P.R. China
Phone 86-10-6788-0551 Fax 86-10-6788-2878

YASKAWA ELECTRIC CORPORATION

YASKAWA

MANUAL NO. SIE-C815-10.1F

© Printed in Japan January 2001 85-3
99-7③