PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Latch Read (LR) function is used to determine if there are any discrete or register output module circuits not functioning. When the function is enabled, the latches on specified output register or discrete modules are read and compared to their corresponding OR or OG values in the PC. The results of the comparison may then be used to determine if there are any improper operating output circuits.

With the enable input conducting, the function reads the output latches on the module(s) whose reference numbers are associated with the source table. The function determined status of the output latches is then exclusively OR'ed (on a bit-per-bit basis) with the programmable controller's I/O image table and the results are placed in the destination table. Any non-zero bits in the destination table signify particular output circuits which are not functioning. For example, as shown in Figure 2, the PC determined states for outputs 1 through 32 do not correspond with each circuits module latch status as determined by the LR function. The destination register allow the user to easily identify that outputs 18 and 19 are not functioning properly. This function can also be used as a "Module Missing" detector. LR function symbology is shown in Figure 1.

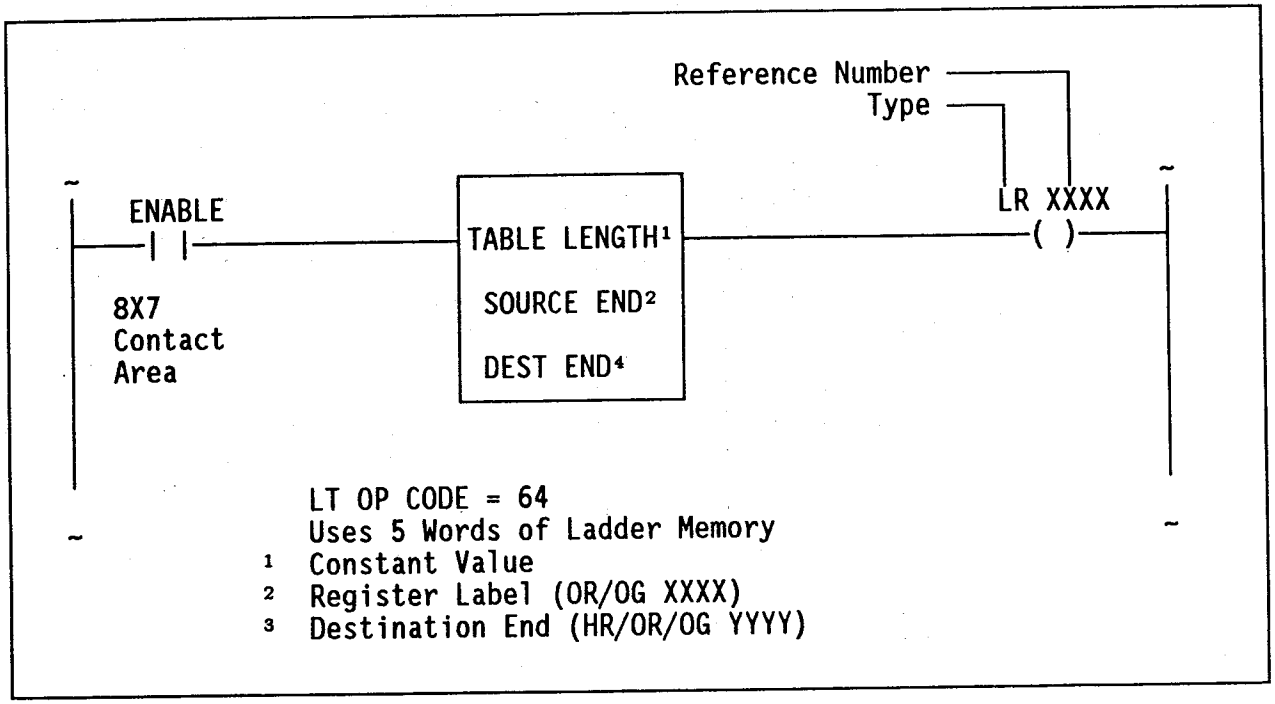


Figure 1. Latch Read Concept

Note

This function tests only to the logic side of the module's optocouplers. A faulted optocoupler or output driver will not be detected by the Latch Read.

SOURCE TABLE

BIT #	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
OG0002	0	0	1	1	1	1	1	1	0	0	0	0	0	0	1	1
OG0001	0	0	0	0	1	1	1	1	0	0	1	1	1	1	1	1

Status of CR 1 through CR 32 as determined by the PC

BIT #	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
OG0002	0	0	1	1	1	1	1	1	0	0	0	0	0	1	0	1
OG0001	0	0	0	0	1	1	1	1	0	0	1	1	1	1	1	1

Status of CR 1 through CR 32 as monitored by the Latch Read function

DESTINATION TABLE

BIT #	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0025 (CR17-32)	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
HR0024 (CR1-16)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The destination table shows that CR 18 and CR 19 are not functioning properly.

Figure 2. LR Function Operation

OP CODE

Op Code 64 defines the Literal (LT) as an LR function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - TABLE LENGTH

Table Length is a constant value which defines the number of OR's or OG's whose associated output latches are to be read.

OPERAND 2 - SOURCE END

Source Table End which is subject to the limits in Table 1 defines the type and number of the last register in the Latch Read Table. In the PC-1200, if the source end register or group does not correspond to real-world I/O (i.e., OG0064), the LR coil will turn on to signal there is a problem and operation will not occur.

TABLE 1. TABLE LENGTH AND SOURCE END LIMITS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128
[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.				

OPERAND 3 - DESTINATION END

Destination Table End defines the type and reference number (subject to the limits in Table 2) of the last register of the table containing the results of the LR function.

Note

The highest holding register address which can be referenced is dependent upon the memory size and the user program.

TABLE 2. DESTINATION END LIMITS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

COIL

In the PC-1100, the coil follows input contacts. In the PC-1200, the coil is energized if any output bits are detected to be in error.

LR TRUTH TABLE

See Table 3.

TABLE 3. LR TRUTH TABLE

Enable	Result
0	The destination registers remain unchanged.
1	The output latches on the modules associated with the Source Table are read and then XOR'ed with the data in the Source Table and the result is placed in the Destination Table each scan.

APPLICATIONS

The LR function is useful in critical applications where an improperly functioning output circuit could be hazardous. Using the program shown in Figure 3, outputs 1 through 64 are checked each scan for improperly operating circuits. If a malfunctioning circuit is present, the Search Matrix (SM) coil will energize. The status of the Search Matrix coil may then be used to initiate an alarm or a controller shutdown sequence.

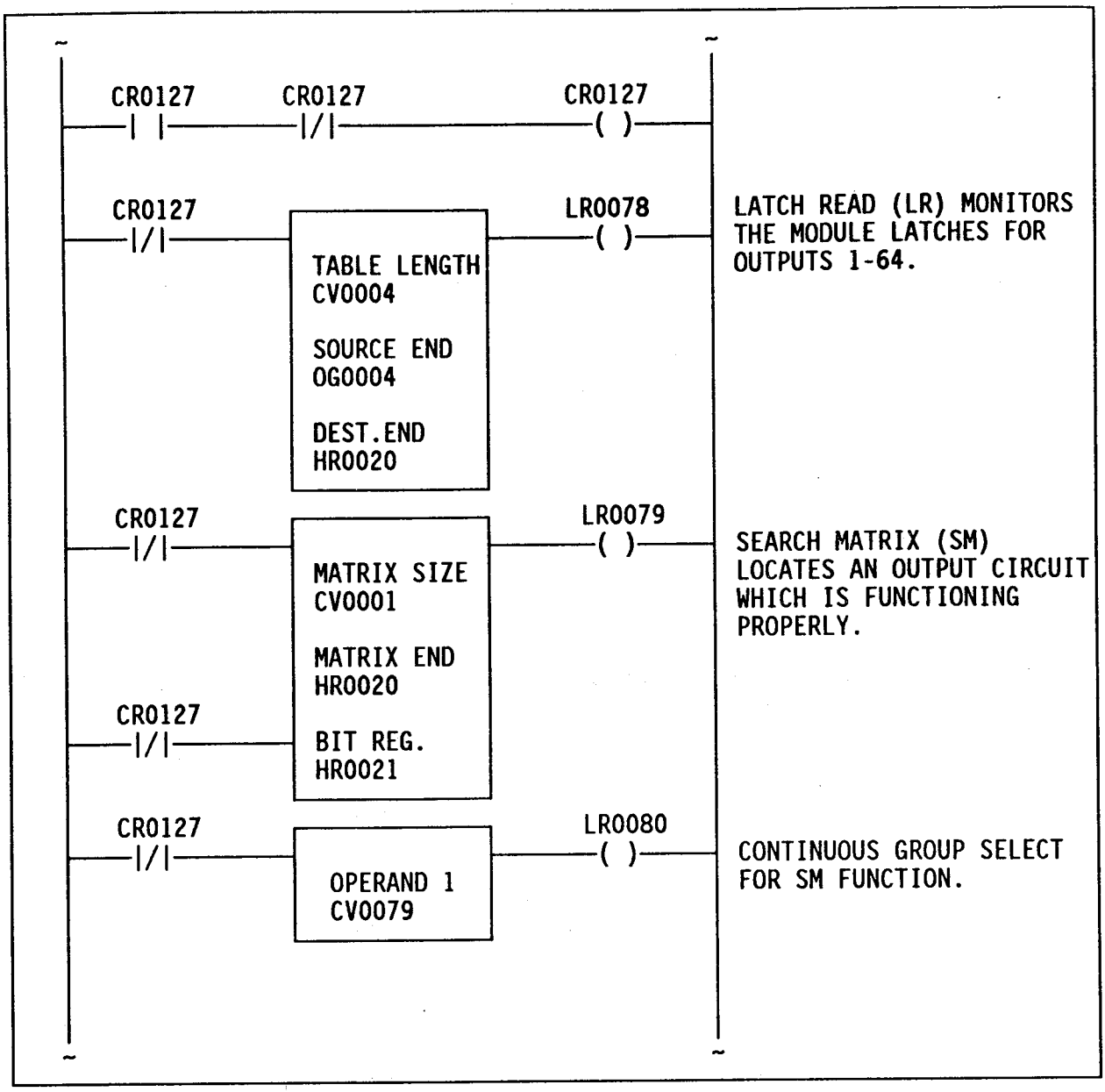


Figure 3. LR Application

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Lock Scan (LS) function was developed for use in applications where critical timing is important. With a lock scan circuit enabled, the function determines the actual scan time of the just-completed ladder scan time and I/O update (value in milliseconds) and stores it in a user-specified register. The function then compares the actual scan time of the just-completed scan to a user-entered desired scan time (in milliseconds). If the actual scan time is less than the desired scan time, the processor will wait the difference before beginning the next scan. If the actual scan time is greater than the desired scan time, the coil will energize and normal processing of the ladder diagram will continue. With the enable circuit not conducting, the coil is de-energized (function not performed) and the actual scan duration register is not altered.

n the event that more than one LS coil is programmed in a ladder diagram, scan calculation will be based on the last LS coil in the program. The actual scan duration time will be written into all LS coils used.

Figure 1 illustrates the concepts and symbols related to the Lock Scan function.

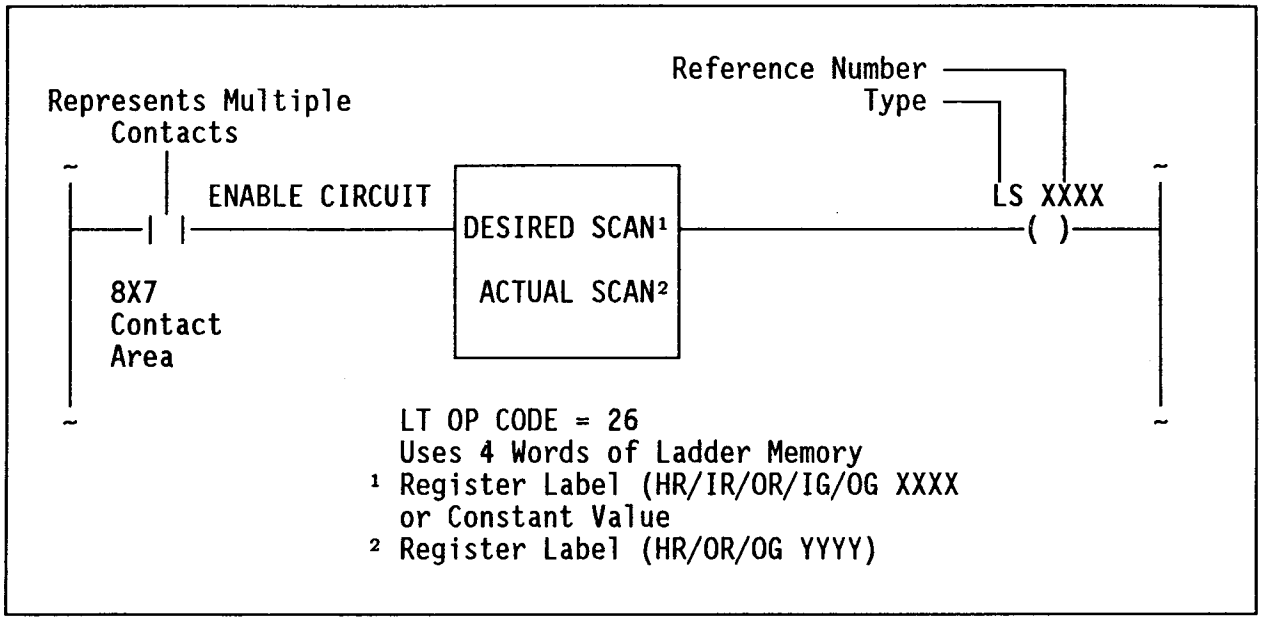


Figure 1. Lock Scan Function (LS) Concept

LS

OP CODE

Op Code 26 defines the Literal (LT) as being a Lock Scan (LS) function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - DESIRED SCAN

Processor scan time (in milliseconds) specified by the user. May be a Constant Value (CV) or may be specified in a:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

OPERAND 2 - ACTUAL SCAN

The actual duration (in milliseconds) of the just-completed ladder scan and I/O update. May be placed in a specified:

- Holding Register (HR)
- Output Register (OR)
- Output Group (OG)

COIL

When the enable input is off, the coil is de-energized, and the function is not performed. When the enable input is conducting, the coil remains de-energized, unless the desired scan time is less than the actual scan duration. (See Table 1.)

TABLE 1. TRUTH TABLE

Enable Input	Coil
0	Coil de-energized and function is not performed.
1	Function is performed and coil remains de-energized. If the desired scan time is less than the actual scan duration, the coil is energized, and normal processor scanning will continue.

APPLICATIONS

The Lock Scan (LS) function can be used, along with the Move (MV) function, to determine the "worst case" instantaneous scan time of a process. Consider the program shown in Figure 2. If appropriate, manually set the desired scan time to any value greater than that which has been "self-calculated" to suit the application. After running through several typical machine cycles, the desired scan time will be set to the worst case scan time.

Often the worst case instantaneous scan time can be reduced by careful evaluation of the program. By staggering the calculation of many special functions so that they are calculated over several scans rather than during a single scan, the worst case instantaneous scan time can be considerably improved.

Operand 2 (Actual Scan Time) always gives the instantaneous scan time. To calculate the average scan time, the program shown in Figure 3 may be used.

Figure 4 shows the Lock Scan used in a critical timing application. Assume that CR0010 is to energize every 150 msec. Using a counter and a Lock Scan function with a desired scan time of 50 msec, every third scan (150 msec) CR0010 will energize.

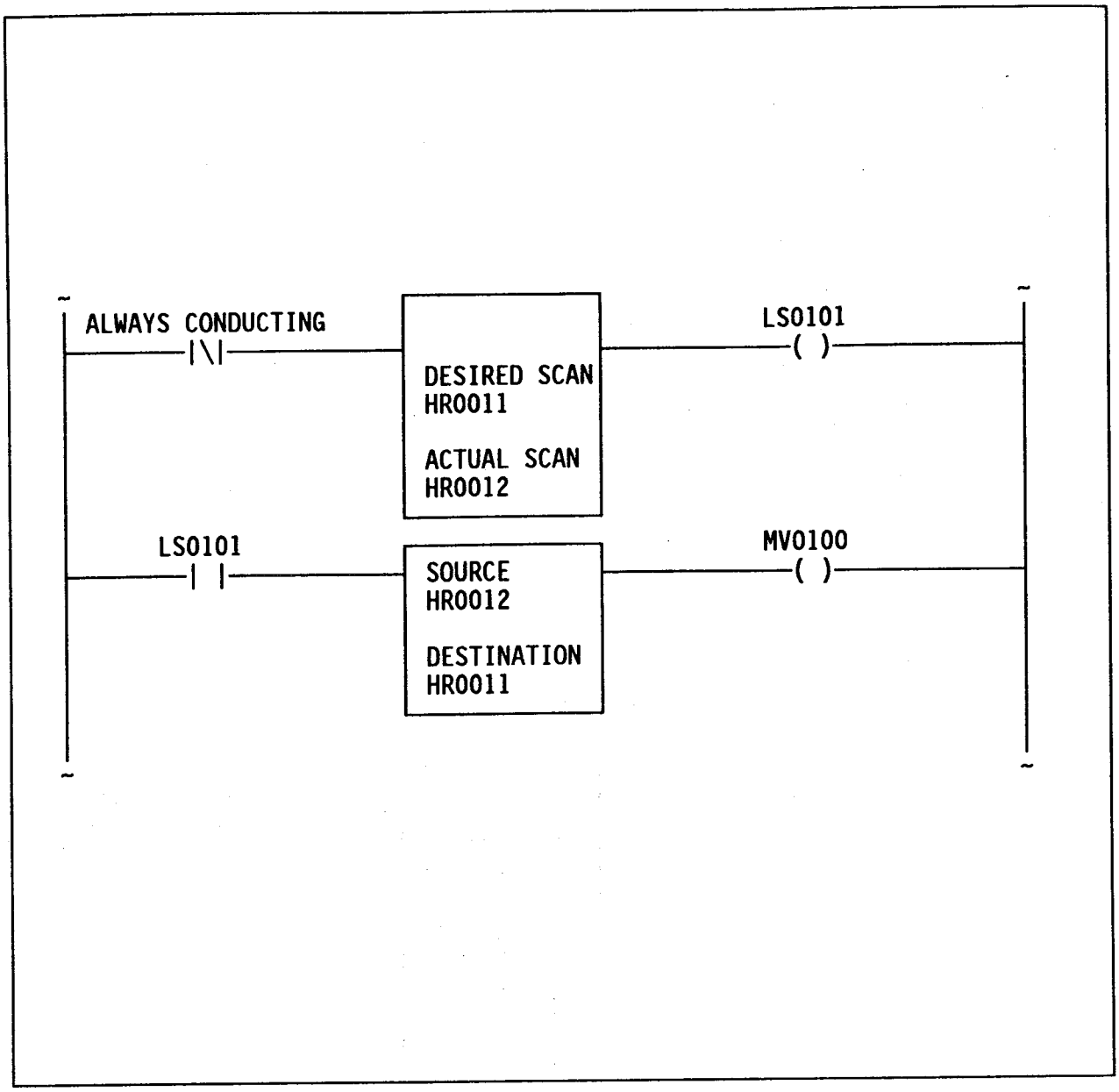


Figure 2. Determining Worst Case Instantaneous Scan Time

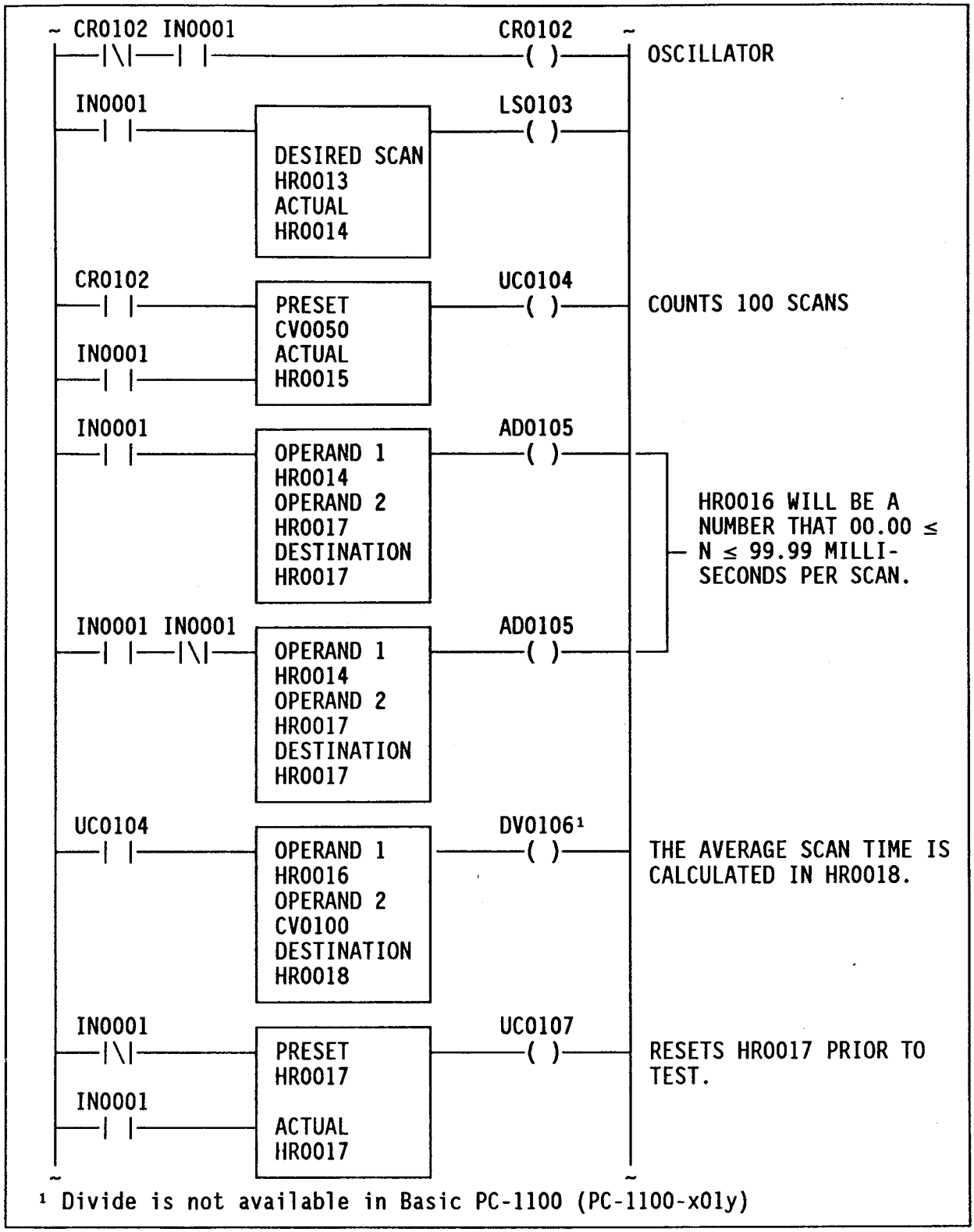
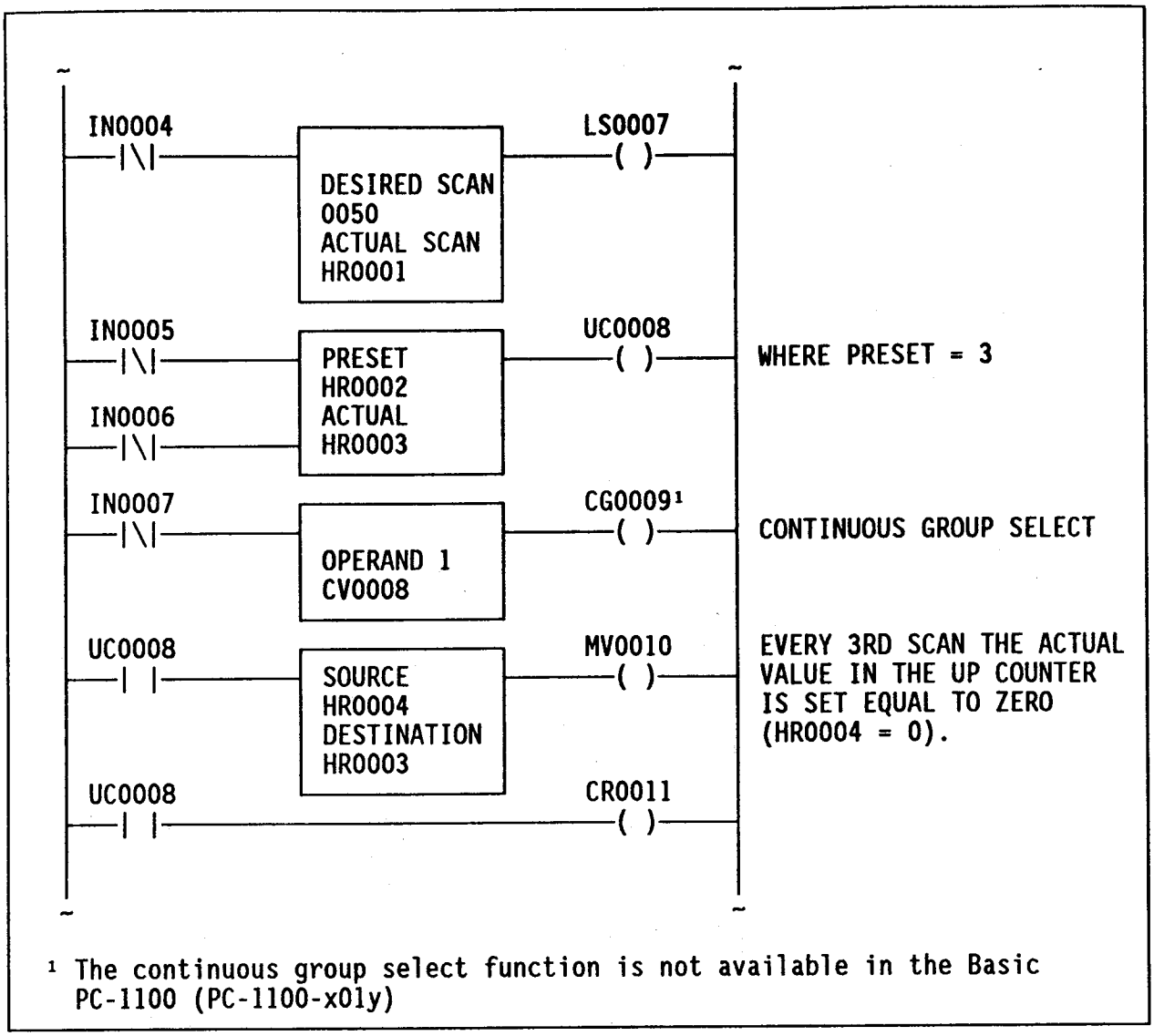


Figure 3. Average Time



[1] The continuous group select function is not available in the Basic PC-1100 (PC-1100-x01y)

Figure 4. Critical Timing Application

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Literal (LT) special function is a generic template. By assigning a unique number code (Op Code), the LT will operate as a specific special function. The LT template displays four generic Operands. These operands assume the characteristics of the special function specified by the Op Code. Figure 1 shows the Literal special function display.

Literals provide a method of entering new special functions into programmable controllers from older program loaders. As a generic version of the mnemonic special functions, the Literal special functions were created to prevent program loader obsolenscence,Special functions which are supported by newer

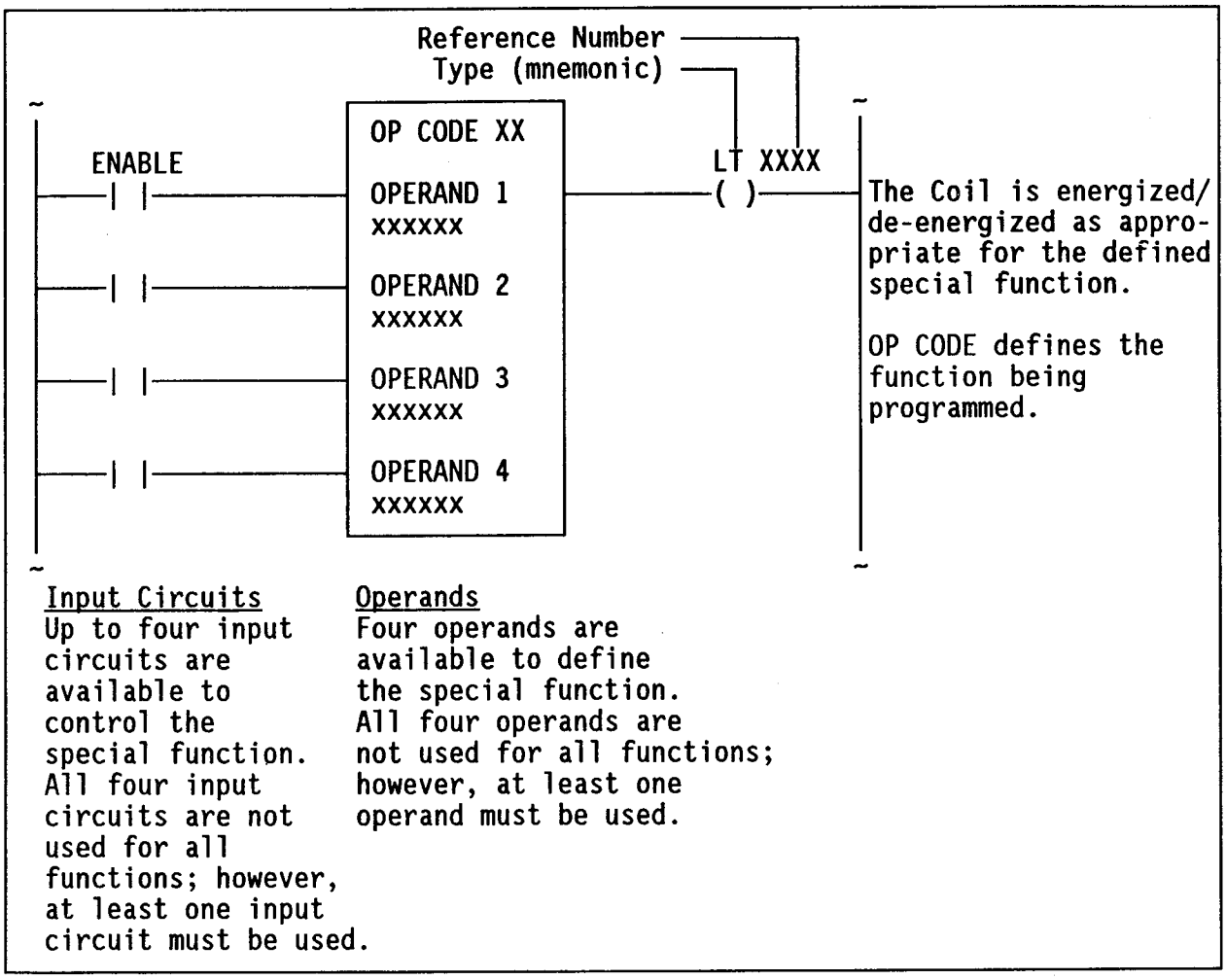


Figure 5-1. Literal (LT) Function

programmable controllers can be added using program loaders which do not support with mnemonic functions. This is especially true with the older program loaders, such as the CRT Program Loaders (NLPL-780) and the Mini Loaders (NLPL-789).

The Advanced Program Loader also supports Literals in anticipation of future special function releases. However, it is recommended that the mnemonic special function be used when available. The mnemonic system checks for proper operand entries, register types and number of inputs.

Caution should be used when assigning values to Literal special function operands. Values and registers entered into LT operands must conform to the specific special function being activated. Failure to do this could cause program or system misapplication.

Caution

Avoid using the LT special function when the mnemonic special function is available. Misapplication of LT could result in program or system misapplication.

Table 1 lists those special functions that can be represented by LT. Note that this may not be a complete list. Newly released special functions may be available through LT. Check with your Westinghouse representative.

As new special functions are released, they will be programmed originally as LT. Westinghouse will provide documentation on the proper implementation of the new special function.

Caution

Do not attempt to program LT without the direction of proper documentation. Program or system malfunctions could occur.

Some special functions can only be programmed using the mnemonic method. All program loaders support these mnemonic special functions. Table 2 lists these functions.

TABLE 1. SPECIAL FUNCTIONS SUPPORTED BY LT

Special Function	Mnemonic	OP CODE
AND Matrix	AM	90
ASCII Receive	AR	48
Ascending Sort	AS	58
ASCII Transmit	AT	91
Bit Operate	BO	61
Block Transfer	BT	60

TABLE 1. SPECIAL FUNCTIONS SUPPORTED BY LT (Cont'd)

Special Function	Mnemonic	OP CODE
Continuous Group Select	CG	15
Complement Matrix	CM	57
Configure Port	CP	27
FIFO Stack	FI/FO	81/85
FILO Stack	FI/LO	81/86
Indirect Move	IM	95
Loop Control	LC	94
Latch Read	LR	64
Lock Scan	LS	26
Move Byte	MB	63
N Bit Shift Registers	NR/NL	50/51
OR Matrix	OM	89
Open Table/Close Table	OT/CT	92/93
Port Transmit	PT	97
Restore Program Counter	RP	13
Reset Watchdog Timer	RW	14
Search Matrix	SM	56
Save Program Counter	SP	12
Square Root	SQ	19
Table Lookup	TL	82
Table Lookup Ordered	TO	83
Table-to-Register Move	TR	84
Unit Address	UA	28
Update Select	US	25
XOR Matrix	XM	88

TABLE 2. MNEMONIC SPECIAL FUNCTIONS

Special Function	Mnemonic
Add	AD
Bit Clear	BC
Binary to BCD	BD
Bit Follow	BF
Bit Set	BS
Control Relay	CR
BCD to Binary	DB
Down Counters	DC
Divide	DV
Equal To	EQ
Greater Than or Equal To	GE
Multiply	MP
Master Control Relay	MR
Move	MV
Subtract	SB

TABLE 2. MNEMONIC SPECIAL FUNCTIONS (Cont'd)

Special Function	Mnemonic
Skip	SK
Timer (Seconds)	TS
Timer (Tenths of second)	TT
Up Counters	UC
I/O Update Immediate	UI

SPECIFICATIONS

OP CODE

The number of the selected special function is entered into this field. The number in this field assigns a specific function to LT. Refer to Table 1.

INPUT CIRCUITS

Up to four input circuits are available to control the selected special function. All four are not necessary, however, at least one is required. Refer to the appropriate mnemonic special function description to determine the number of inputs that are required for the selected special function.

OPERAND 1, OPERAND 2, OPERAND 3, OPERAND 4

The number and purpose of the operands is determined by the specific special function being programmed. All four operands are not used for all special functions, however, at least one must be used. These operands take on the characteristics of the specific operands assigned to the specified special function. When completing these entries, refer to the specific special function description for proper values and limits.

COIL

The LT coil reference number is limited by the range supported by the particular programmable controller being used. The status of the coil is determined by the specific special function being implemented. Refer to the appropriate mnemonic special function description to determine the status of the LT coil.

In older program loaders, the coil will be labeled with "LT" and the appropriate reference number. In newer and upgraded program loaders, the coil will appear as LT until the special function is entered into the ladder program. Once entered, the program loader replaces "LT" with the appropriate special function mnemonic.

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Move Byte (MB) function transfers data from the upper, lower, or both upper and lower bytes of the source register (HR, IR, OR, IG, OG) to the upper, lower, or both upper and lower bytes of the destination register (HR, OR, OG). The transfer code register (Xfer Code) determines which byte(s) of the source register will be moved. It also determines the location the byte(s) will occupy in the destination register.

When an enable circuit is conducting, as in Figure 1, depending on the transfer code selected, one of the following data moves will take place.

- Lower byte of a source register is transferred to the lower byte of a destination register.
- Lower byte of a source register is transferred to the upper byte of a destination register.
- Upper byte of a source register is transferred to the lower byte of a destination register.
- Upper byte of a source register is transferred to upper byte of a destination register.
- Upper and lower bytes of a source register are transferred to the upper and lower bytes of a destination register, respectively.

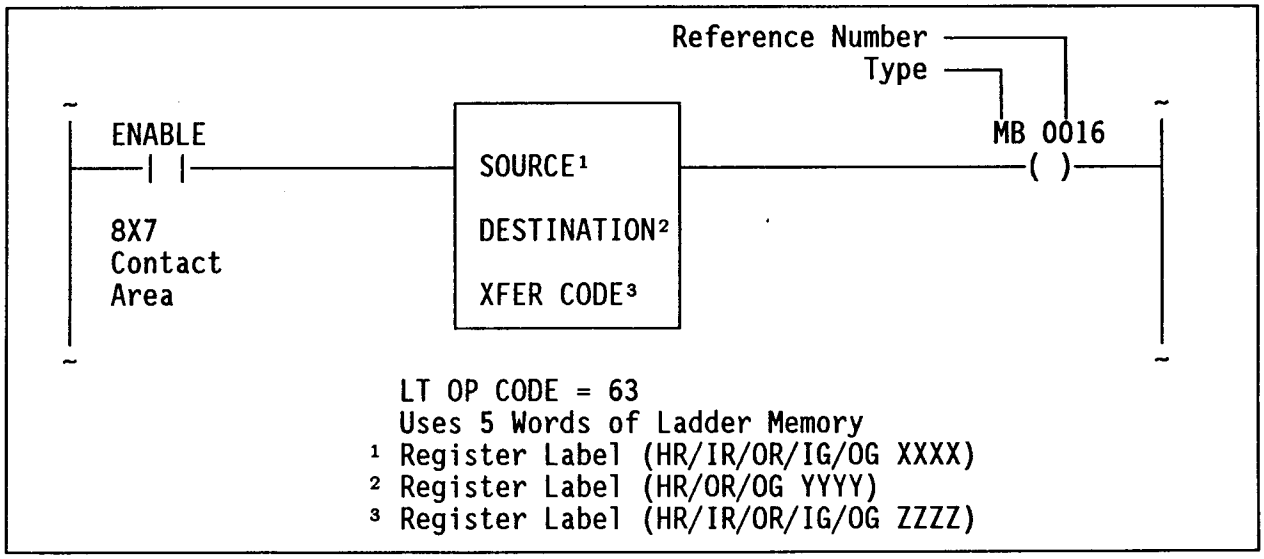


Figure 1. Move Byte Concept

- Upper and lower bytes of a source register are transferred to the lower and upper bytes of a destination register, respectively.
- Upper byte and lower byte of a single register exchange positions (that is, source location equals destination location).

When an enable circuit is not conducting, no data transfer is made, and the coil is de-energized. The coil is energized only when the enable circuit is energized and an invalid transfer code is used.

If the coil is forced, only its contacts (and output circuit, if any) are affected; the Move Byte function continues to operate according to the Move Byte circuit.

Figure 1 illustrates the concepts and symbols related to the Move Byte function.

OP CODE

Op Code 63 defines the Literal (LT) as being a Move Byte (MB) function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - ENABLE

Allows data to be copied from a source register to a destination register during each processor scan according to the Xfer Code.

OPERAND 2 - SOURCE

The location from which data is to be moved. (Data remains intact at this location.) This value may be held in a specified:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

OPERAND 3 - DESTINATION

The location to which data is to be moved. This location may be a specified:

- Holding Register (HR)
- Output Register (OR)
- Output Group (OG)

TABLE 1. TRANSFER CODE

Source Register			Destination Register	
Upper Byte	Lower Byte	Xfer Code	Upper Byte	Lower Byte
H	L	0	Not Altered	Not Altered
H	L	1	Not Altered	L
H	L	2	Not Altered	H
H	L	3	L	H
H	L	4	L	Not Altered
H	L	5	H	Not Altered
H	L	6	H	L
H	L	7-65535*	Not Altered	Not Altered
*7 thru 65535 are invalid Xfer Codes				

COIL

Coil is energized when the enable circuit is conducting and an invalid Xfer Code is used (i.e., 7 thru 65,535).

APPLICATIONS

The A/D Modules use one 16-bit input register to store the data from 2 input channels on the Module. (The lower byte contains one channel; the upper byte contains another. See Figure 2.) The Move Byte function can be used to transfer the 8-bit values from an input register containing data from two A/D channels to a single register with data from only one channel. The 8-bit values from the A/D Module may then be processed by the user ladder diagram. For example, a 4-channel A/D Module is set so that the two input registers used to store the A/D Module is set so that the two input registers used to store the A/D conversions are IR0003 and IR0004. The program shown in Figure 3 would move the data from each channel to a single holding register for processing.

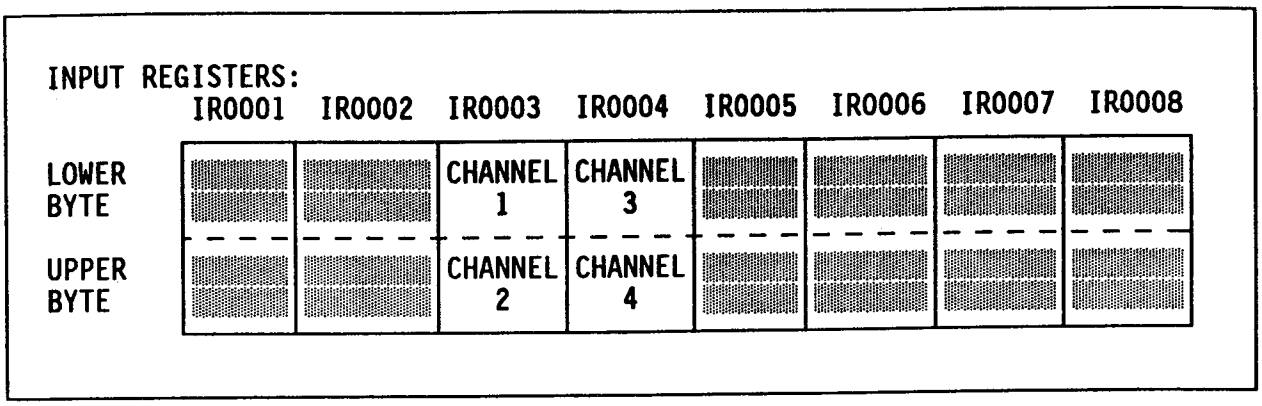


Figure 2. Values Location Before MB Operation

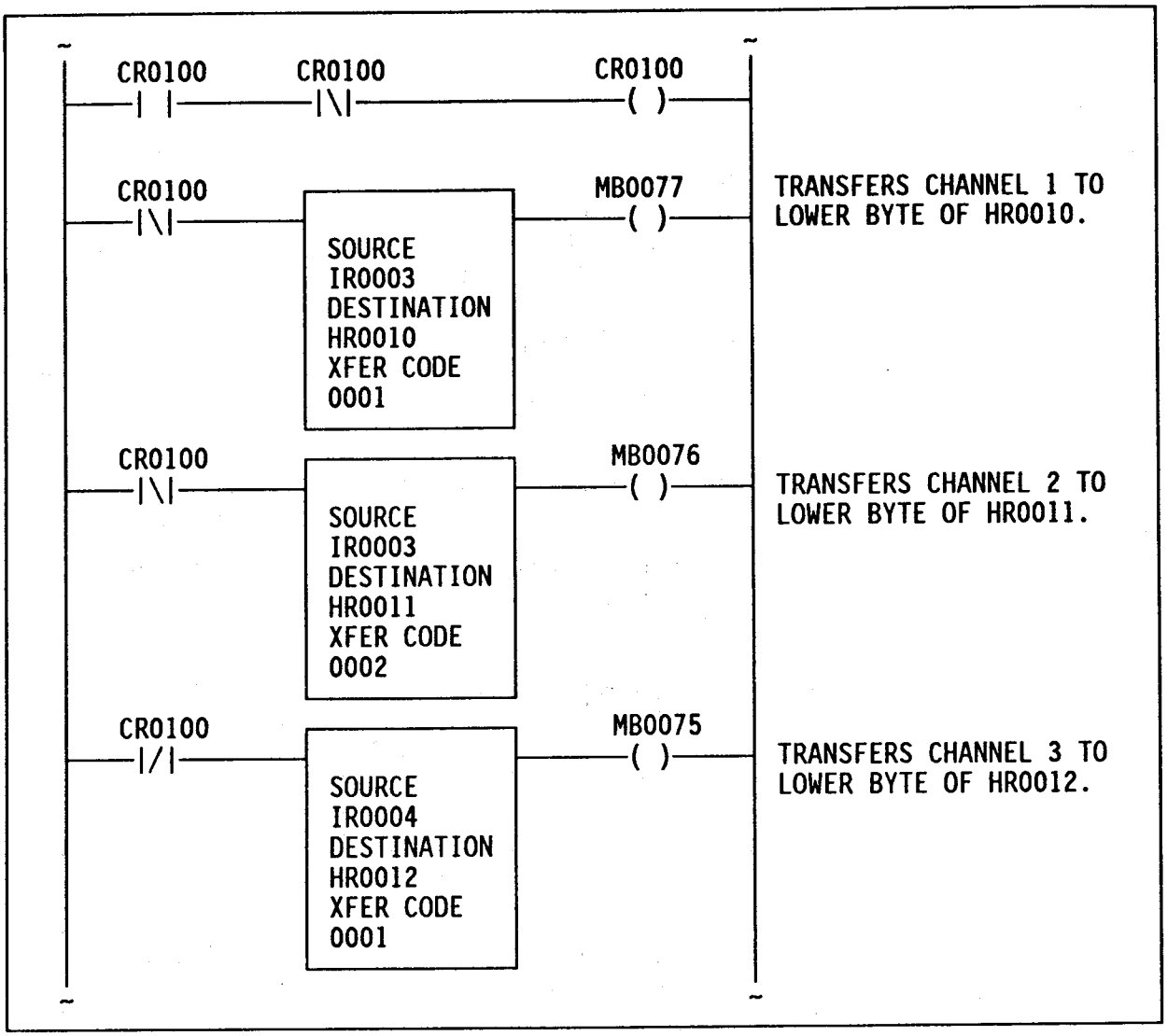


Figure 3. Move Byte Function Example

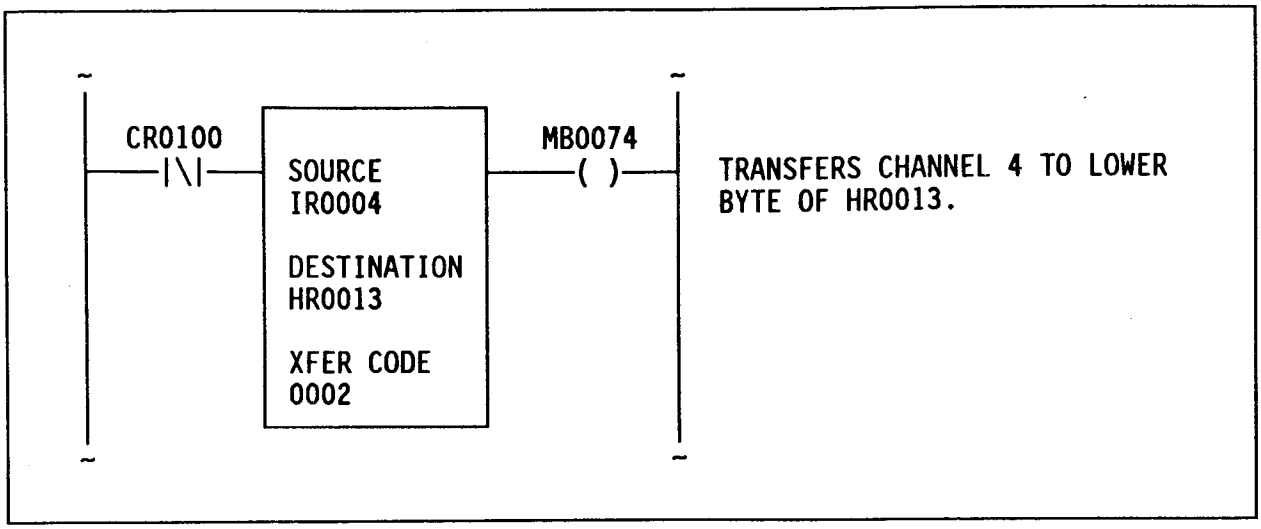


Figure 3. Move Byte Function Example (Cont'd)

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Multiply (MP) function multiplies two, four-digit decimal numbers (up to 9999). This multiplication results in a number of up to eight decimal digits (up to 99,980,001). MP function symbology is shown in Figure 1.

Operand 1 and Operand 2 are multiplied when the enable circuit changes from non-conducting to conducting. Operand 1 comes from an input, output, or holding register. Operand 2 comes from one of these registers or is programmed as a constant. The result is placed in a destination, which is composed of a consecutive pair of output or holding registers.

Although the limits on the operands and the result are decimal, multiplication is performed using equivalent binary numbers, since the register values for the operands and the result are in binary form. The Binary to Decimal (BD) function is used to convert binary numbers to Binary-Coded-Decimal (BCD); the Decimal to Binary (BD) function is used to convert BCD to binary form. Unlike the Add (AD) or Subtract (SB) functions, results exceeding 9999 are not evaluated as standard binary numbers, since the result is split between two registers.

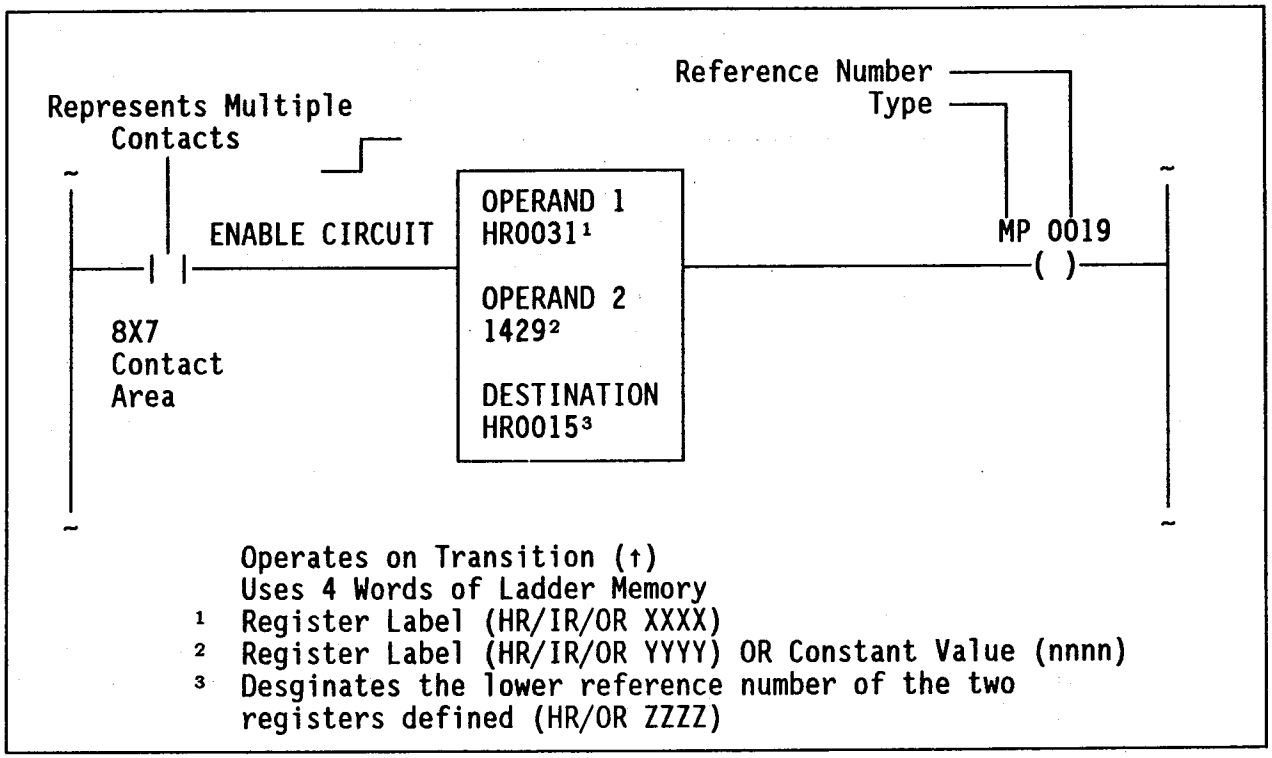


Figure 1. Multiply (MP)

The destination register pair is referenced by a single label, which automatically designates the register to which it refers, and the register of the same type having the next highest reference number. For example, 127 x 496 = 62,992. If Holding Register 25 is designated as the destination register, then 0006 is placed in HR0025, and 2992 is placed in HR0026.

In the PC-1100, the MP coil energizes when the enable circuit conducts, and de-energizes when the circuit does not conduct. Forcing an MP coil affects only the associated contacts and output circuits (if any); the MP function continues to operate under the enable circuit control.

In the PC-1200, the MP coil energizes only when the result of the multiply is greater than 655,359,999. An overflow will cause an invalid result. Forcing an MP coil affects only the associated contacts and output circuits (if any); the MP function continues to operate under the enable circuit control.

SPECIFICATIONS

OPERAND 1

This value is held in a specified Holding Register (HR), Input Register (IR), or Output Register (OR).

OPERAND 2

This value is a constant (0001 through 9999) or a value held in a specified Holding Register (HR), Input Register (IR), or Output Register (OR).

DATA SUMMARY

The operand range is for a PC-1100 is 0000 through 9999. The range for a PC-1200 is 0000 through 65535. Operand 1 x Operand 2 = Destination register pair contents.

DESTINATION

The destination is a consecutive pair of register locations; the first location contains the four most-significant digits of the result; the second location contains the four least-significant digits of the result. The destination is a specified pair of Holding Registers (HR), or Output Registers (OR).

Note

Both operands must be in binary form.

MP TRUTH TABLE

See Table 1.

TABLE 1. MP TRUTH TABLE

Enable Circuit	Result
0	None - The MP coil is de-energized.
↑	Multiplies - The result is placed in the destination register pair. The coil is energized. In the PC-1200, if the result is greater than 655,359,999 the coil energizes (error detected), otherwise the coil remains off.
1	In the PC-1100, the coil is energized. In the PC-1200, the coil is unchanged.
↑ = Transition OFF to ON.	

APPLICATIONS

The MP function is particularly useful when scaling factor inputs. Refer to the application example for the Divide (DV) special function. This example uses the MP function as well.

Figure 2 shows a program using the MP function to square a number.

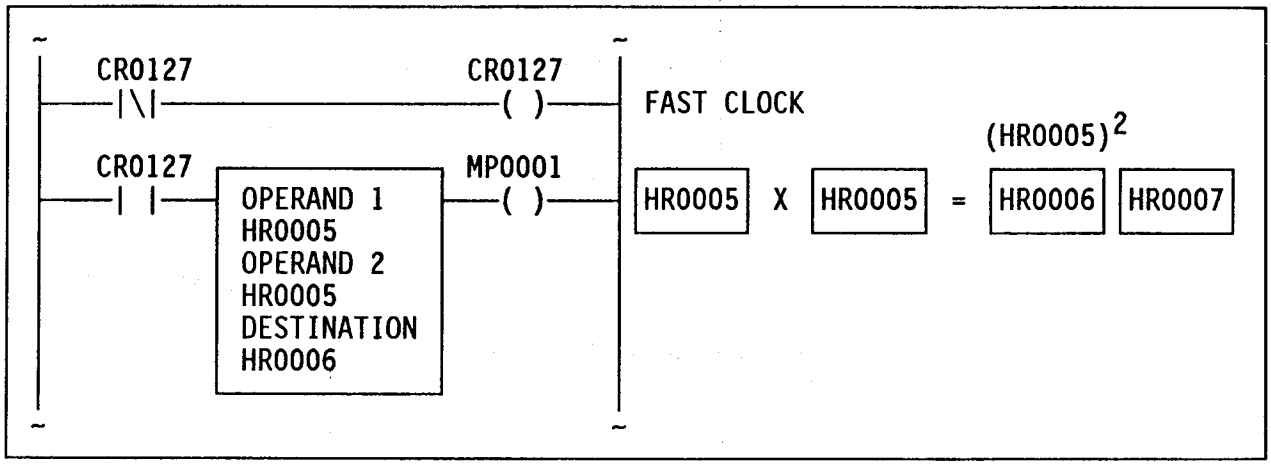


Figure 2. Number Squaring

Figure 3 shows a "times 10" application of this MP function. This configuration uses the second destinataion register as an operand. In this case, each time IN0001 is closed, HR0001 is multiplied by 10, and the result is placed in HR0010 and HR0011. Table 2 illustrates the operation for eahc IN0001 closure.

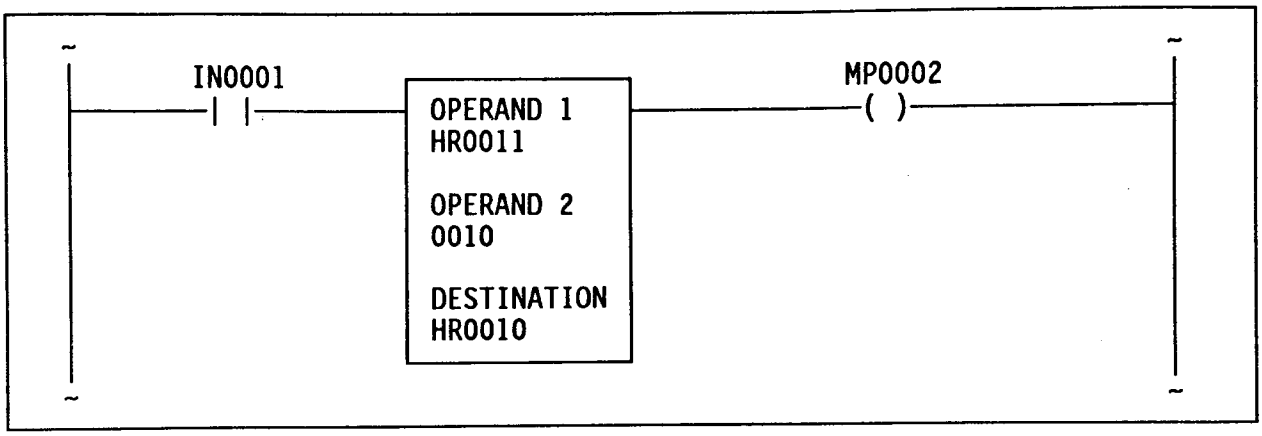


Figure 3. Times 10 Application Example

TABLE 2. TIMES 10 APPLICATIONS EXAMPLE FOR IN0001 CLOSURE

	Contents of HR0010	Contents of MR0011
Initial condition	0000	1234
First operation of IN0001	0001	2340
Second closure of IN0001	0002	3400
Third transition of IN0001	0003	4000
Fourth time IN0001 is closed	0004	0000

MR - MASTER CONTROL RELAY

Modified for PC-1200

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Master Control Relay (MR) function is a powerful programming tool. This function provides an "internal OFF" operation, disabling all or part of the programmed reference ladder diagram. MR function symbology is shown in Figure 1.

The MR function allows a prescribed condition or set of conditions to disable all or part of the programmed circuits into the processor. When the MR enable circuit conducts, all coils under MR control operate normally.

PC-1100 OPERATION

In the PC-1100, when the MR enable circuit does not conduct, the coil functions as follows:

- Control relay coils de-energize unless forced ON. (When the coils are forced ON, the associated outputs turn ON.)
- Special functions stop operating and their coils de-energize unless forced ON. (When the coils are forced ON, the associated outputs turn ON.)

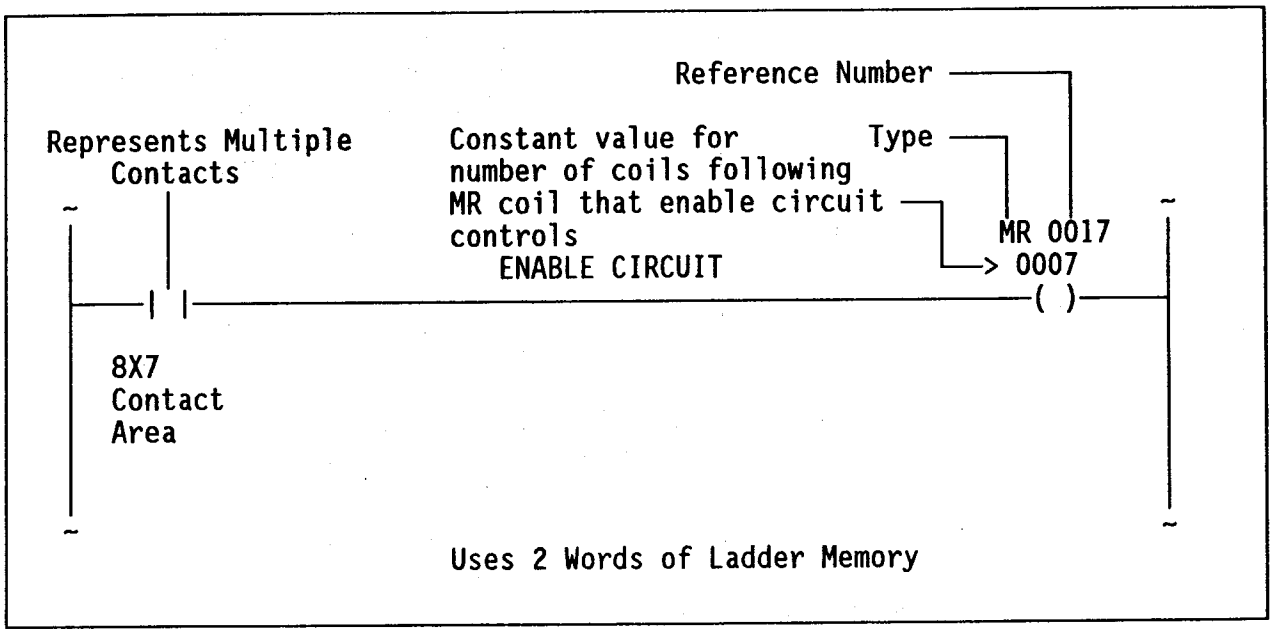


Figure 1. Master Control Relay (MR)

- Register data is frozen, unless the register is also affected by a coil that is not MR controlled.

PC-1200 OPERATION

In the PC-1200, the specified number of coils and special functions operate as if the left ladder rail is de-energized for the range of the MR function.

- Special functions in the MR range act as if their inputs lines are de-engergized. This means times and counters are reset, shift registers clear their register tables, TR and FI pointers are cleared, BF coils clear their programmed bits, etc. Power flow displayed on the program loader does not recognize the left rail is de-energized.

GENERAL

This function, like the Skip (SK) function, only depends on the condition of its contact circuits. Forcing the MR function forces only the MR coil, thus forcing its contacts and output circuits. Forcing the coils that control the contacts in the MR contact circuit forces the MR function.

The number of coils controlled by MR, or range, is specified by a preset constant (1 through 256). This constant indicates the number of coils that follow, and are controlled by, the MR function. The controlled coils are those programmed immediately after the MR coil. If the preset value is 20, the next 20 coils programmed are controlled by the MR contact circuit.

SPECIFICATIONS

ENABLE CIRCUIT

When the enable circuit conducts, normal processing is allowed. In the PC-1100, when the enable circuit does not conduct, the specified number of coils following the MR coil are de-energized. In the PC-1200, when the enable circuit does not conduct, power flow to the specified number of coils following the MR coil is removed. All special functions in this range are executed by the PC-1200 as if there were no power flow at any input.

COIL

When the coil energizes, normal processing is allowed. When the coil de-energizes, the specified number of coils following the MR coil are de-energized. Forcing the coil forces the associated contacts, not the functions.

CAUTION

The MR function does NOT replace an external, hard-wired Master Control Relay. An external relay must be provided to shut down power in an emergency. See Section 3 "Installation and Start-Up" for details.

NUMBER OF COILS

The number of coils specifies the number of coils to be disabled following the MR coil (1 through 256). If the end of the program is reached before the end of the range, functional control is terminated.

APPLICATIONS

A momentary power failure causes the processor to reset all CR coils and re-start the program. However, in some cases, this restart is undesirable. Figure 2 shows an MR function application which prevents automatic restart.

All coils, including MR0001, are turned OFF when a power failure occurs. Since MR0001 is sealed in by its own contacts, it does not re-energize (permit the coils following it to operate) until IN0001, a manual pushbutton, is pressed after power is restored.

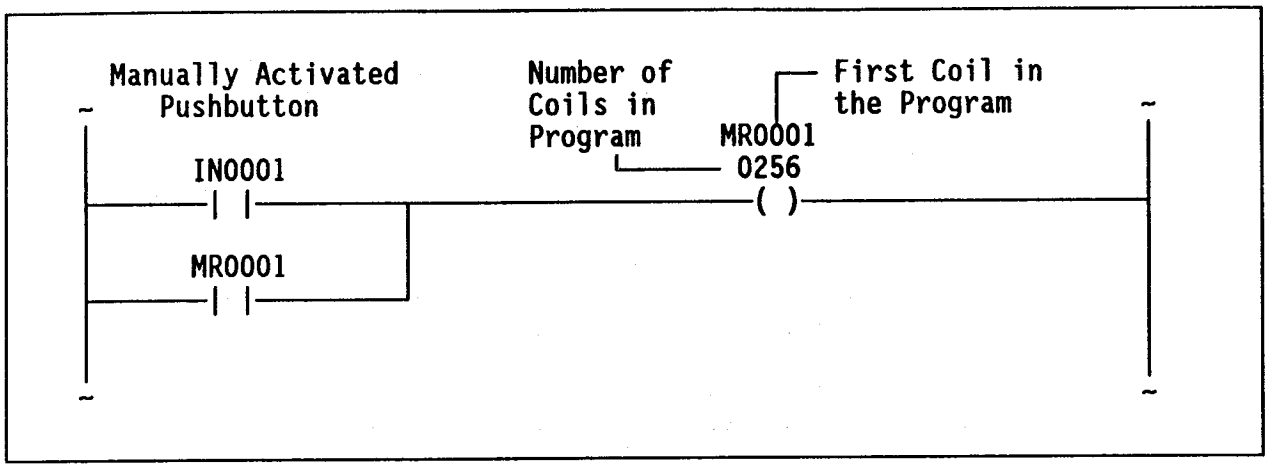


Figure 2. Manual Reset Circuit

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Move (MV) function transfers data from the source to the destination. MV function symbology is shown in Figure 1.

When the MV circuit conducts, the source data transfers to a prescribed destination on each processor scan, and the coil energizes. When the circuit doe not conduct, the destination register holds in its last state, and the coil de-energizes. If the coil is forced, only its contacts and any output circuit are affected; the MV function continues to operate according to the MV circuit.

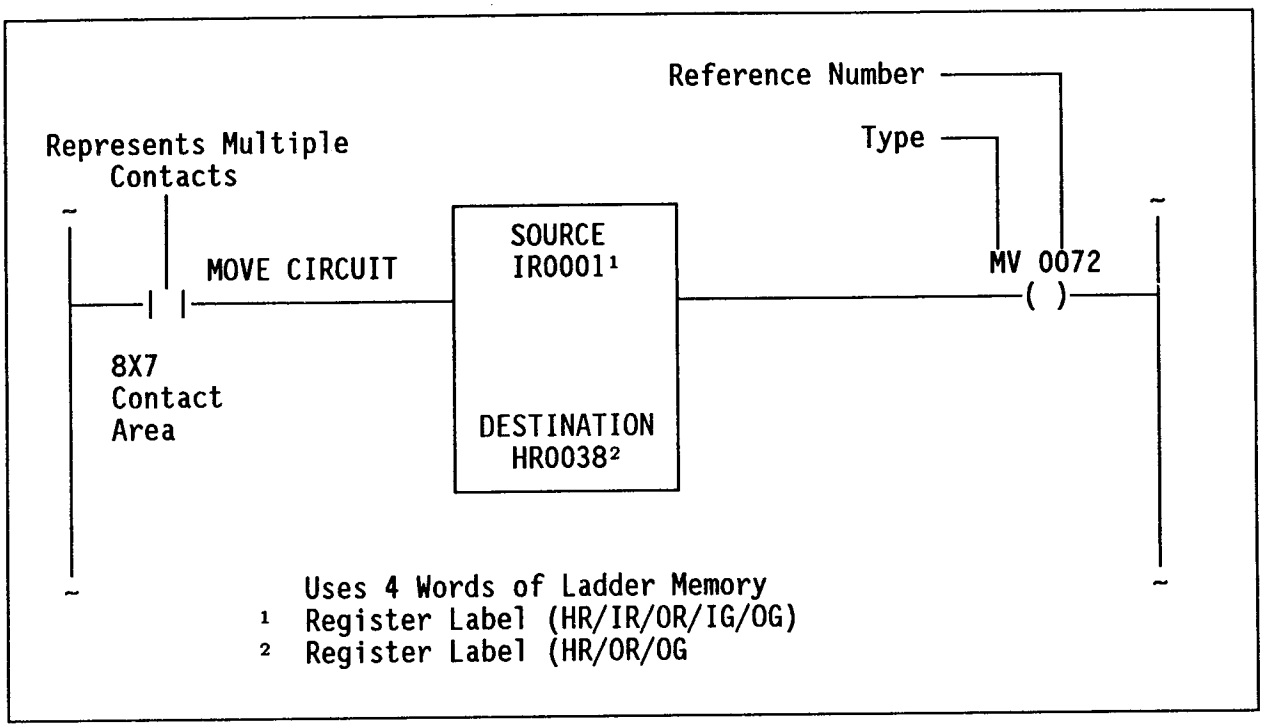


Figure 1. Move (MV)

SPECIFICATIONS

MV CIRCUIT

When the MV circuit conducts, data is copied from the source to the destination during each processor scan.

SOURCE

The source is the location from which data is copied. (Data remains intact at this location.) This value is held in a specified register or group:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

DESTINATION

The destination is the location to which data moves. This location is a specified register or group:

- Holding Register (HR)
- Output Register (OR)
- Output Group (OG)

COIL

The coil energizes when the MV circuit conducts, and de-energizes when the circuit does not conduct.

APPLICATIONS

The Divide (DV) function illustrates the usefulness of the MV function in conserving register space. Rather than using two output registers for the DV result and the remainder when only the result is desired, a pair of holding registers is used as the destination. The result is moved only to the output register as shown in Figure 2.

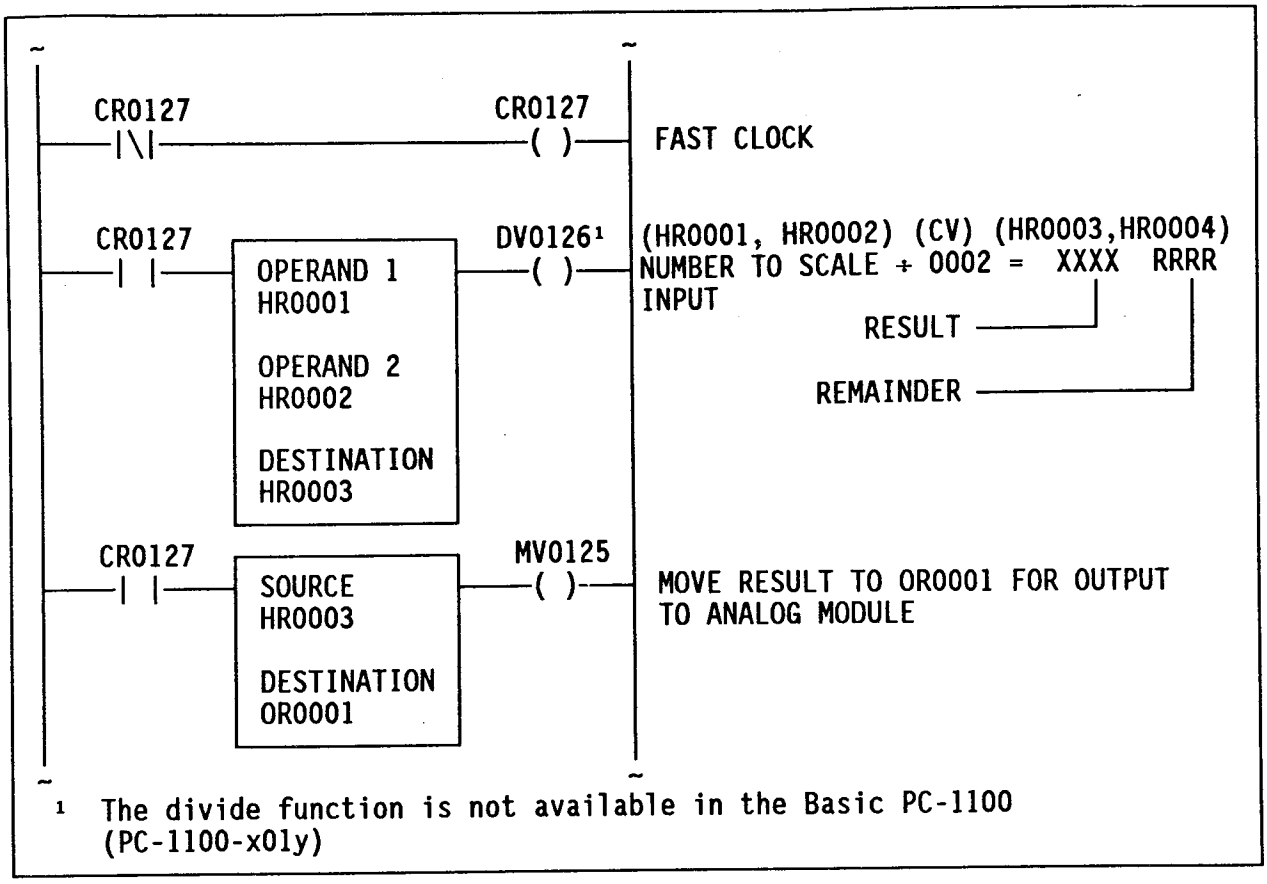


Figure 2. MV Application

NR/NL - N BIT SERIAL SHIFT REGISTERS

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The N Bit Serial Shift Register-Right (NR) and the N Bit Serial Shift Register-Left (NL) functions consist of 1 through 128, 16-bit registers (16 through 2048) bits) that can shift N (0 through 16) bits right (NR) or left (NL) at a time. NR/NL function symbology is shown in Figure 1.

The NR/NL functions are controlled by the shift, serial IN, and reset circuits. If the reset circuit does not conduct, the contents of the table are set to zero, and the function is inactive. If the reset circuit conducts, the contents of the register shift right or left when the shift circuit changes from non-conducting to conducting. If the serial IN circuit does not conduct, zeroes are shifted into the register, and when conducting, ones are shifted into the register.

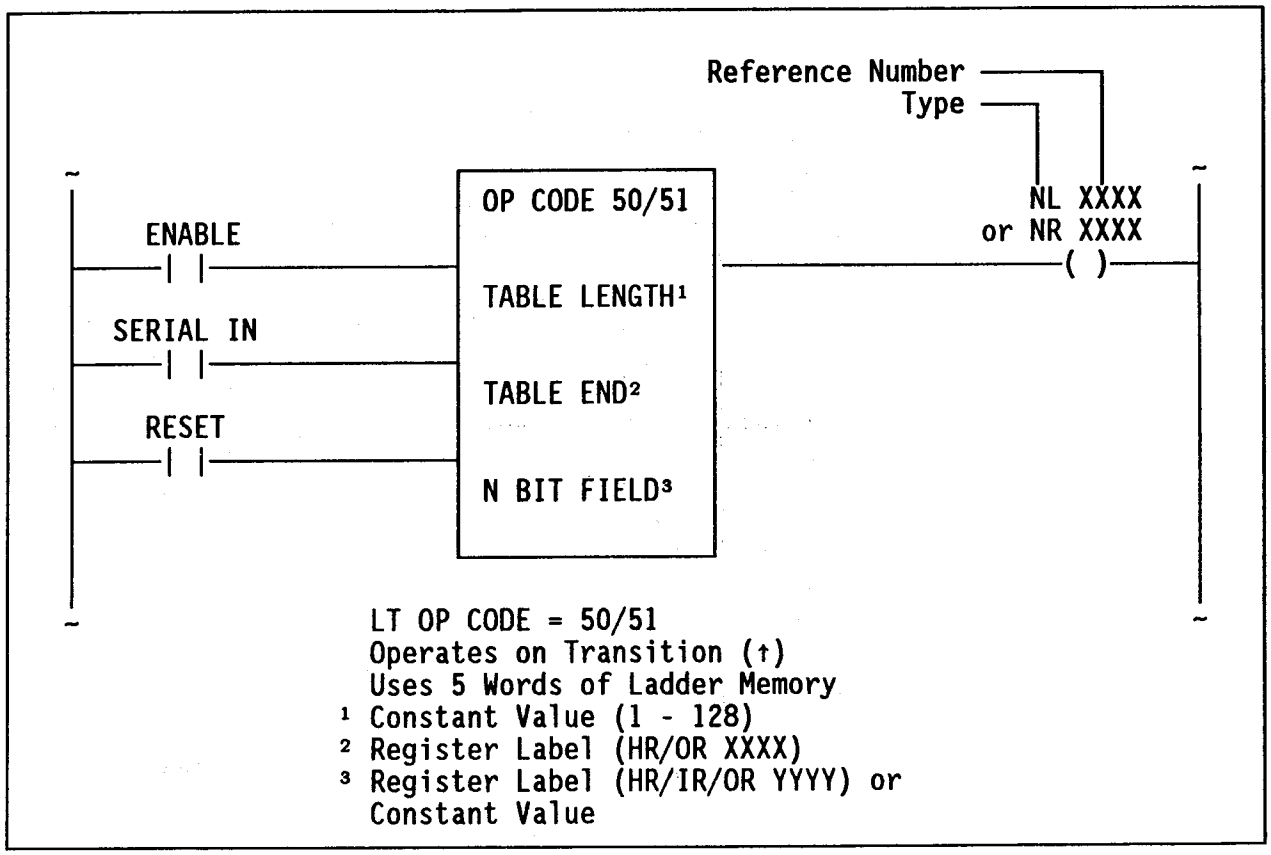


Figure 1. N Bit Serial Shift Register - Right (NR) or Left (NL)

Figure 2 is an example of the N Bit Serial Shift operation. With a shift register that is two holding registers long, the following conditions occur when four (N) bits shift to the left:

- Reset = Closed
- Serial IN = Open
- Shift = Closed

Right shifts work in a reverse manner. Bits shift from Bit 32 towards Bit 1.

OP CODE

Op Code 50 or 51 defines the Literal (LT) as the NR/NL function. Op Code 50 is the NR function; Op Code 51 is the NL function.

SPECIFICATIONS

TABLE LENGTH

The table length defines the number of registers making up the N Bit Shift Register. The range is 1 through 128 registers and is subject to the limits listed in Table 1.

Note

The highest number of holding registers available is dependent on memory size.

TABLE END

The table end defines the type and number of the last register in the NR/NL function. The table end is subject to the limits listed in Table 1.

TABLE 1. TABLE LENGTH/TABLE END LIMITS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

N BIT FIELD

The N Bit Field defines the number of bits (0 through 16) to be shifted. This number can be a constant or a value held in a specified register:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Constant Value (1 - 16)

COIL

The coil action is shown in the NR/NL Truth Table.

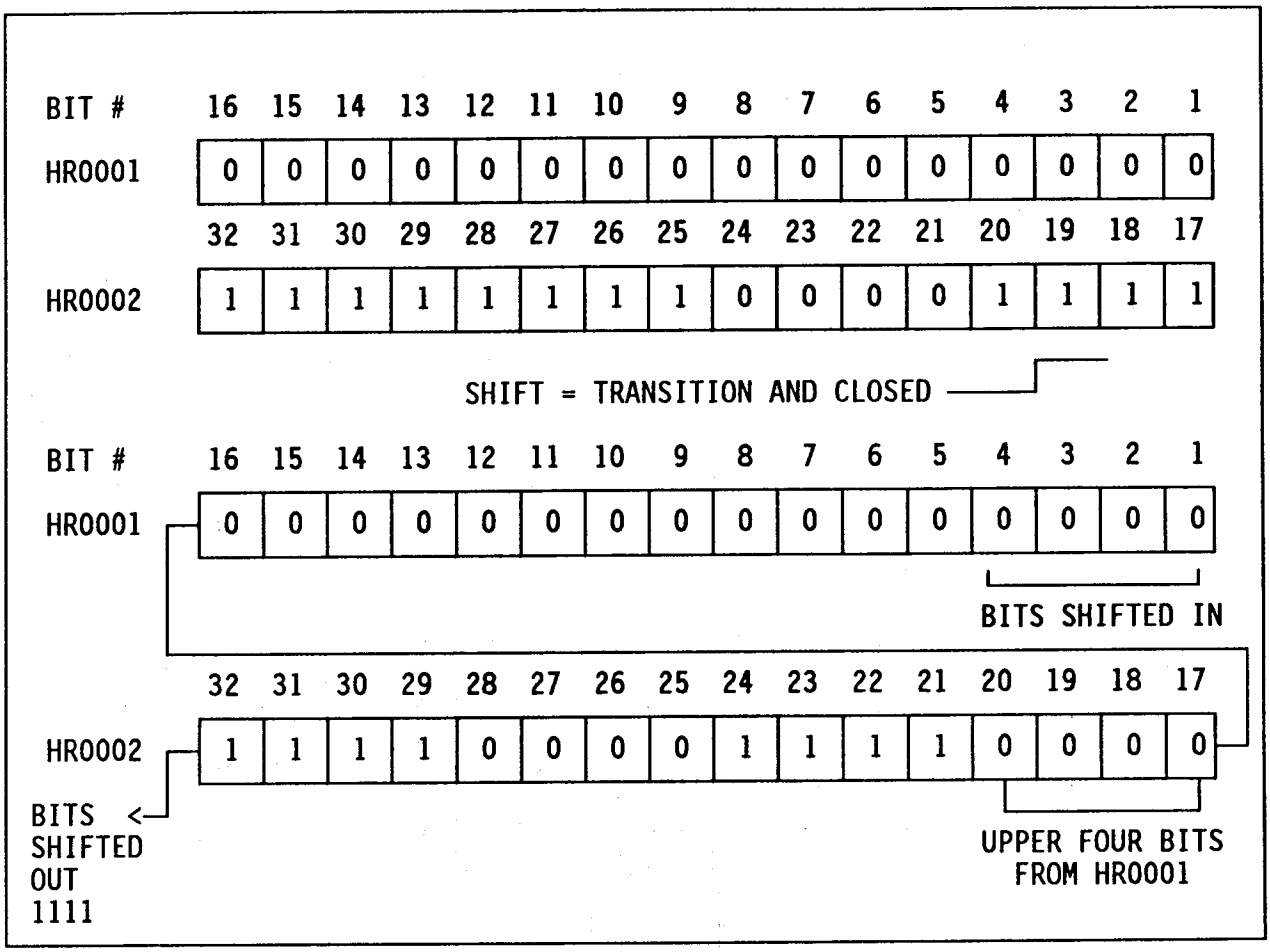


Figure 2. NR/NL Operation

NR/NL TRUTH TABLE

See Table 2.

TABLE 2. NR/NL TRUTH TABLE

Shift	Serial IN	Reset	Result
Don't Care	Don't Care	0	The coil de-energizes. The table is cleared to zero.
0	0	1	The coil and table states are dependent on previous operations.
↑	0	1	The number of bits specified by the N Bit Field are shifted into the table in a "zero" condition. If the specified number of bits is zero or greater than 16, no shift occurs and the coil energizes.
1	0	1	The table contents do not change. The coil energizes if the N Bit Field is zero or greater than 16.
↑	1	1	The number of bits specified by the N Bit Field are shifted into the table in a "one" condition. If the specified number of bits is zero or greater than 16, no shift occurs and the coil energizes.
1	1	1	The coil and table states are dependent on previous operations.

APPLICATIONS

The NR/NL functions are used to implement a conveyor system capable of sorting and diverting selected products at predetermined stations. Figure 3 depicts this operation.

The product code is entered into the system as two Binary-Coded-Decimal (BCD) digits (eight binary bits). A pulse is generated with each increment of conveyor motion. Figure 4 is the ladder diagram for monitoring the conveyor. All the stations are not implemented in this example; they can be implemented with additional programming.

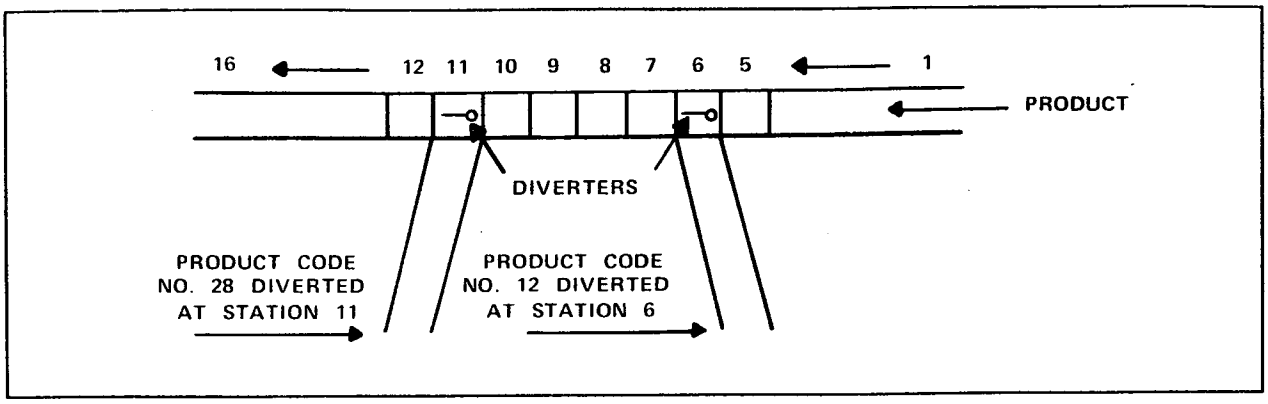
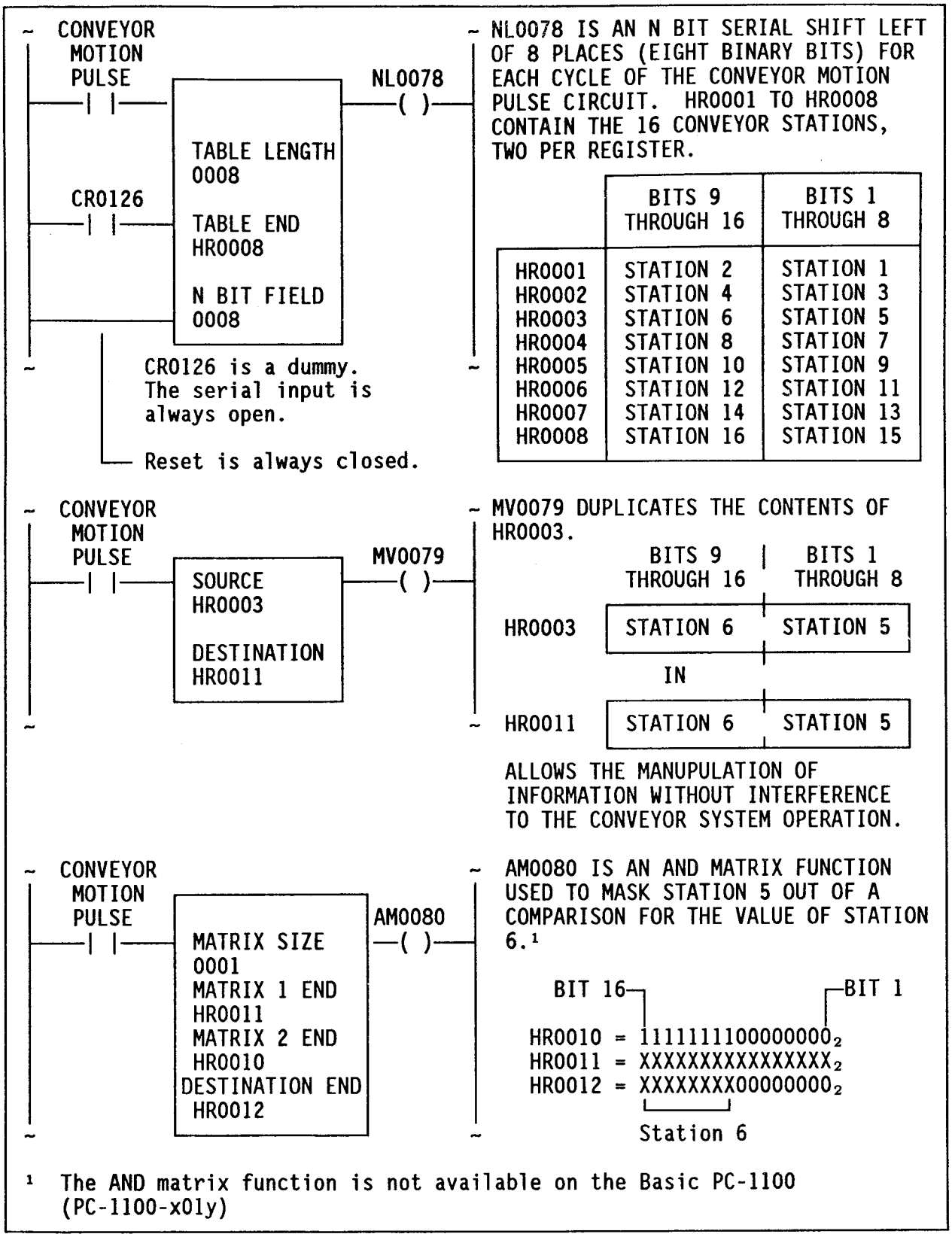
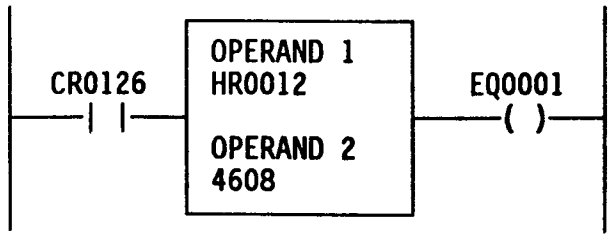


Figure 3. Sorting and Diverting Operation



1 The AND matrix function is not available on the Basic PC-1100 (PC-1100-x01y)

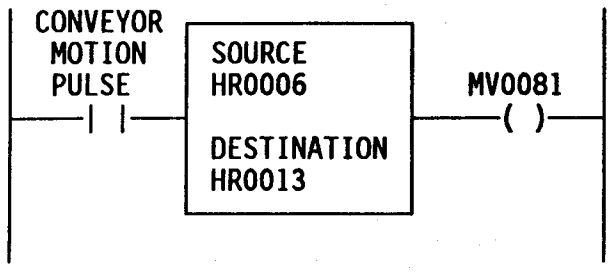
Figure 4a. NR/NL Application



THE CONTENTS OF HR0012, WHICH HAS DATA ONLY IN BITS 9 THROUGH 16, IS COMPARED TO A CONSTANT OF 4608. 4608 IS THE EQUIVALENT OF BCD 1200.

0001 0010 STATION 6	0000 0000

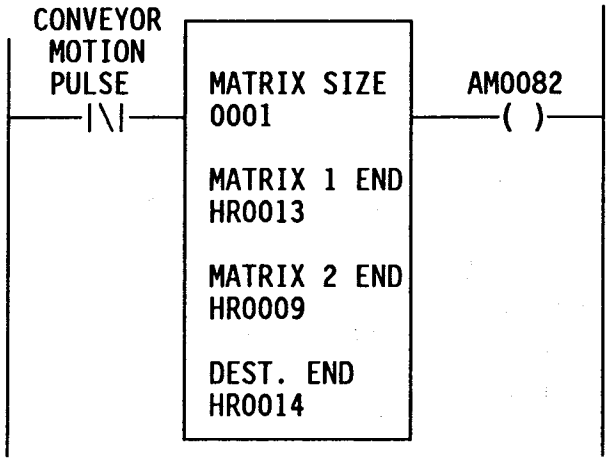
IF THE BCD PRODUCT CODE NO. 12 IS PRESENT, EQ0001 WILL ACTIVATE, CAUSING A DIVERTER TO REMOVE THE PRODUCT FROM THE LINE AT STATION 6



MV0081 DUPLICATES THE CONTENTS OF HR0006.

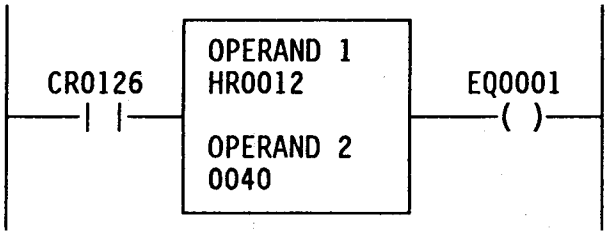
	BITS 9 THROUGH 16	BITS 1 THROUGH 8
HR0006	STATION 12	STATION 11
	IN	
HR0013	STATION 12	STATION 11

ALLOWS THE MANIPULATION OF INFORMATION WITHOUT INTERFERENCE TO THE CONVEYOR SYSTEM OPERATION.



AM0082 IS AN AND MATRIX FUNCTION[1] USED TO MASK STATION 12 OUT OF A COMPARISON FOR THE VALUE OF STATION 11.

	BIT 16	BIT 1
HR0009	0000 0000	1111 1111
HR0013	STATION 12	STATION 11
HR0014	0000 0000	STATION 11



THE CONTENTS OF HR0014, WHICH NOW HAS DATA ONLY IN BITS 1 THRU 8, IS COMPARED TO A CONSTANT OF 40. 40 IS THE EQUIVALENT OF BCD 28.

0000 0000	0010 1000 STATION 11

IF THE BCD PRODUCT CODE NO. 28 IS PRESENT, EQ0003 WILL ACTIVATE, CAUSING A DIVERTER TO REMOVE THE PRODUCT FROM THE LINE AT ST. 14

Figure 4b. NR/NL Application (Cont'd)

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The OR Matrix (OM) function logically OR's the contents of a pair of matrices on a bit-per-bit basis; then, it places the result in a destination matrix location. A matrix is defined as a table of up to 128 16-bit registers operated on a bit-by-bit basis. OM function symboloby is shown in Figure 1.

The OM operation occurs when the enable circuit changes from non-conducting to conducting. The contents of the original matrix are unaffected, as shown in Table 1.

OP CODE

Op Code 89 defines the Literal (LT) as an OM function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

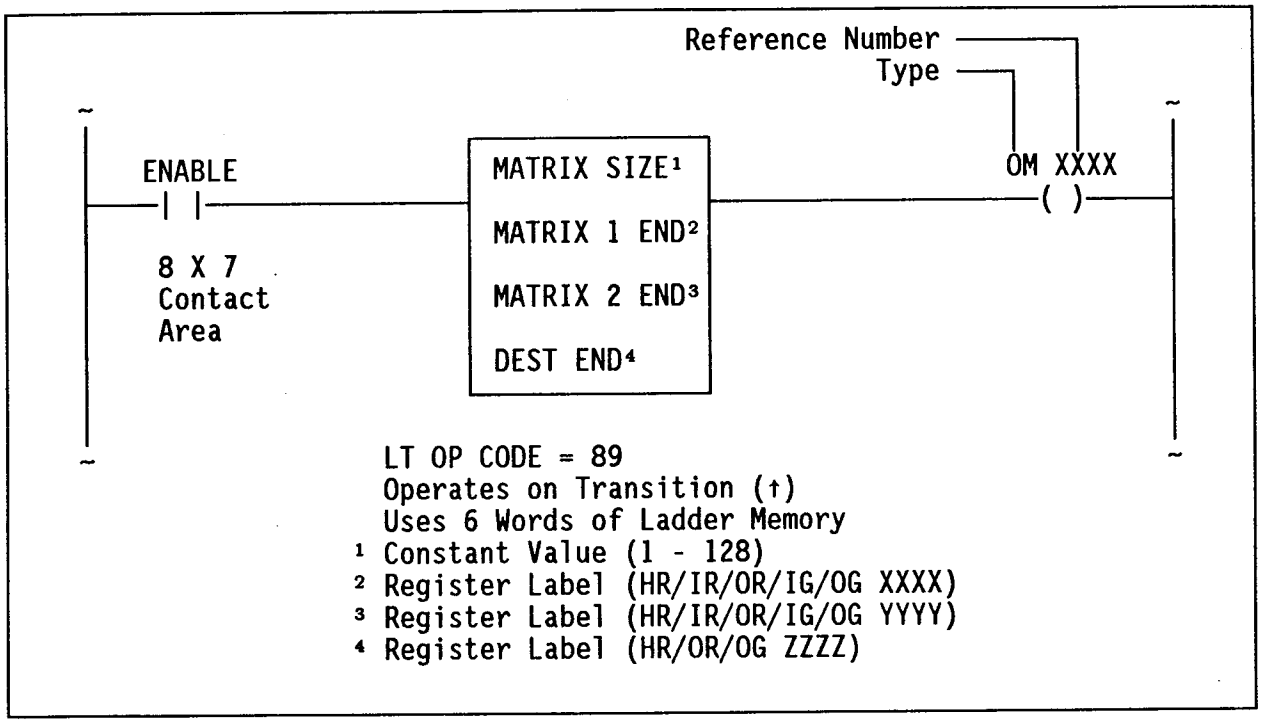


Figure 1. OR Matrix (OM)

TABLE 1. OM TRUTH TABLE SAMPLE

Matrix 1 Bit N	Matrix 2 Bit N	Destination Matrix Bit N
0	0	0
0	1	1
1	0	1
1	1	1
Note N is the same respective bit in all three matrices.		

SPECIFICATIONS

MATRIX SIZE

The matrix size is a constant value that defines the number of registers included in the matrix. The range is from 1 through 128 and is subject to the limitations cited under Matrices 1 and 2.

Matrix 1 End and Matrix 2 End define the type and number of the last register in Matrix 1 and Matrix 2 that will be OR'ed. See Table 2.

TABLE 2. OM END REGISTERS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
IG	≤ 4	≤ 4	≤ 8	≤ 16
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

DESTINATION END

The destination end defines tye type and number of the last register in the matrix that contains the results of the OM function. The type and number limitations are shown in Table 3.

TABLE 3. OM END REGISTERS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

COIL

The coil energizes when the enable circuit is conducting and the result of the OM operation is not zero. At all other times, the coil is de-energized.

OM TRUTH TABLE

See Table 4.

Note

The highest Holding Register reference number acceptable is dependent on memory and user program size.

TABLE 4. OM TRUTH TABLE

Enable	Result
0	The coil is de-energized. Matrix 1, Matrix 2, and the destination are unchanged.
↑	Matrix 1 and Matrix 2 are "OR"ed and the result is placed in the destination matrix. Matrix 1 and Matrix 2 are unaffected by the operation. Coil energized if destination is not zero.
1	The coil remains in state caused by the transition.

APPLICATIONS

The OM function is most helpful in the assembly of data for display. If, for example, a situation occurs that results in two-digit displays, valuable output register assignments can be conserved as shown in Figure 2.

In Figure 2, the lower-eight bits of HR0002 and the upper-eight bits of HR0001 must be zeroes.

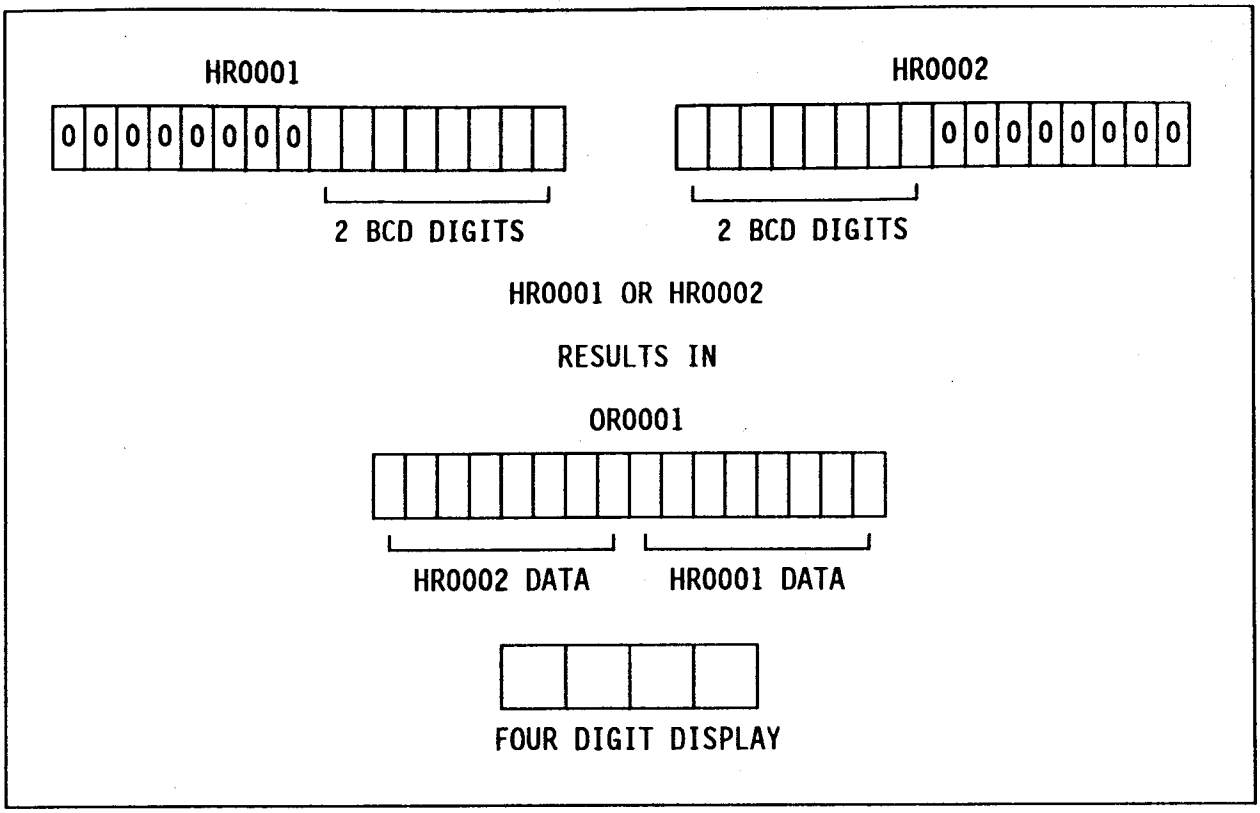


Figure 2. Output Register Assignments

Figure 3 shows the ladder diagram for the OM function. When IN0001 makes the transition from open to closed, the contents of HR0001 are OR'ed with the contents of HR0002 and then placed in OR0001. If the result is not zero, OM0001 is energized as long as IN0001 is closed.

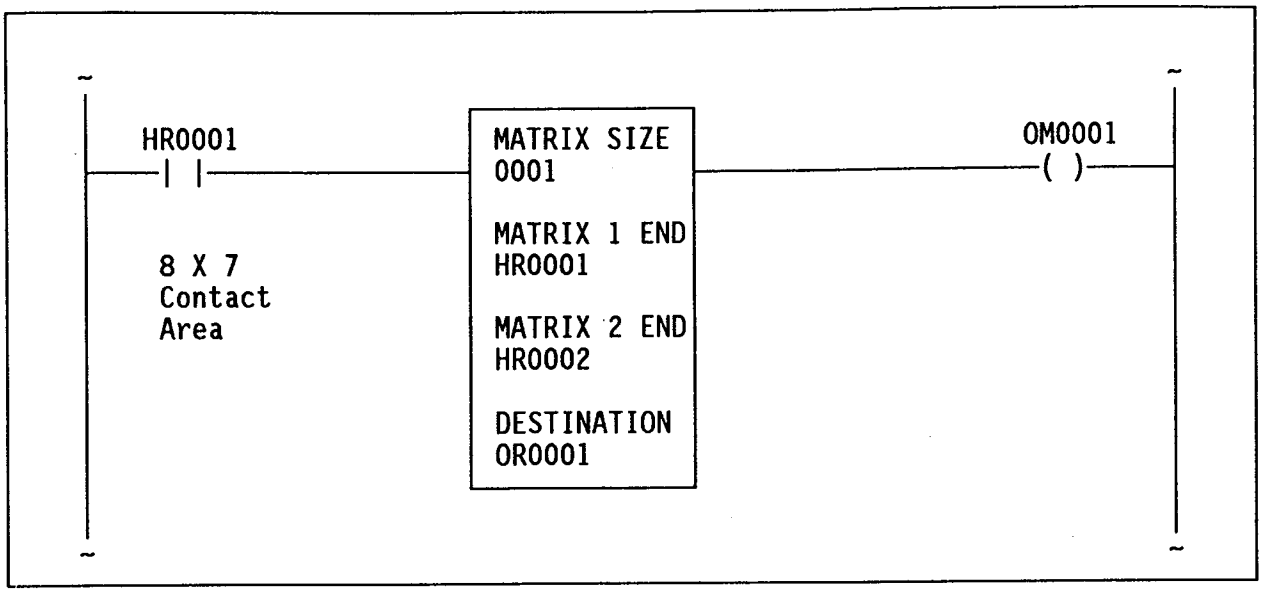


Figure 3. OM Ladder Diagram Example

Figure 4 shows a pair of matrices OR'ed together.

MATRIX 1 END — MATRIX 1 (BITS)

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0001	1	0	0	1	0	1	0	0	1	1	1	1	0	0	1	1

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0002	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MATRIX 2 END — MATRIX 2 (BITS)

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0004	0	1	1	0	0	0	0	0	0	0	1	0	1	1	0	0

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0005	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0006	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

DESTINATION END — MATRIX 3 (BITS)

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0001	1	1	1	1	0	1	0	0	1	1	1	1	1	1	1	1

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0002	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0003	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

Figure 4. A Pair of Matrices OR'ed

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Open Table (OT) and Close Table (CT) functions are used to insert or delete data from a table of registers. OT/CT function symbology is shown in Figure 1.

When a table is opened, data in the pointed location and all data in registers with higher numbers are moved up one position in the table. The new information in the source is inserted into the table and the data in the last table location is lost, as shown in Figure 2.

Note

If the table is expanded before OT is executed, data will be preserved.

When a table is closed, data in the pointed location is placed in the destination, and all data in registers with higher reference numbers are moved down one location. The last register in the table is duplicated when the data is moved, as shown in Figure 3.

OP CODE

Op Codes 92 and 93 define the Literal (LT) as the OT or CT function. Op Code 92 is for the OT function; Op Code 93 is for the CT function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

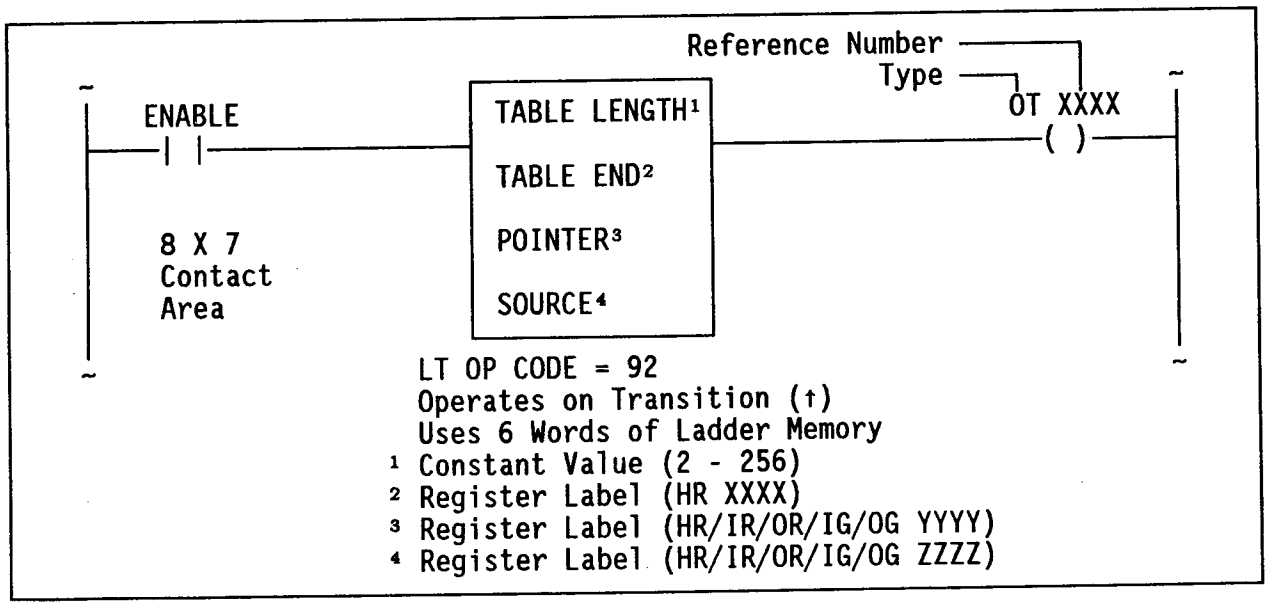


Figure 1a. Open Table (OT)

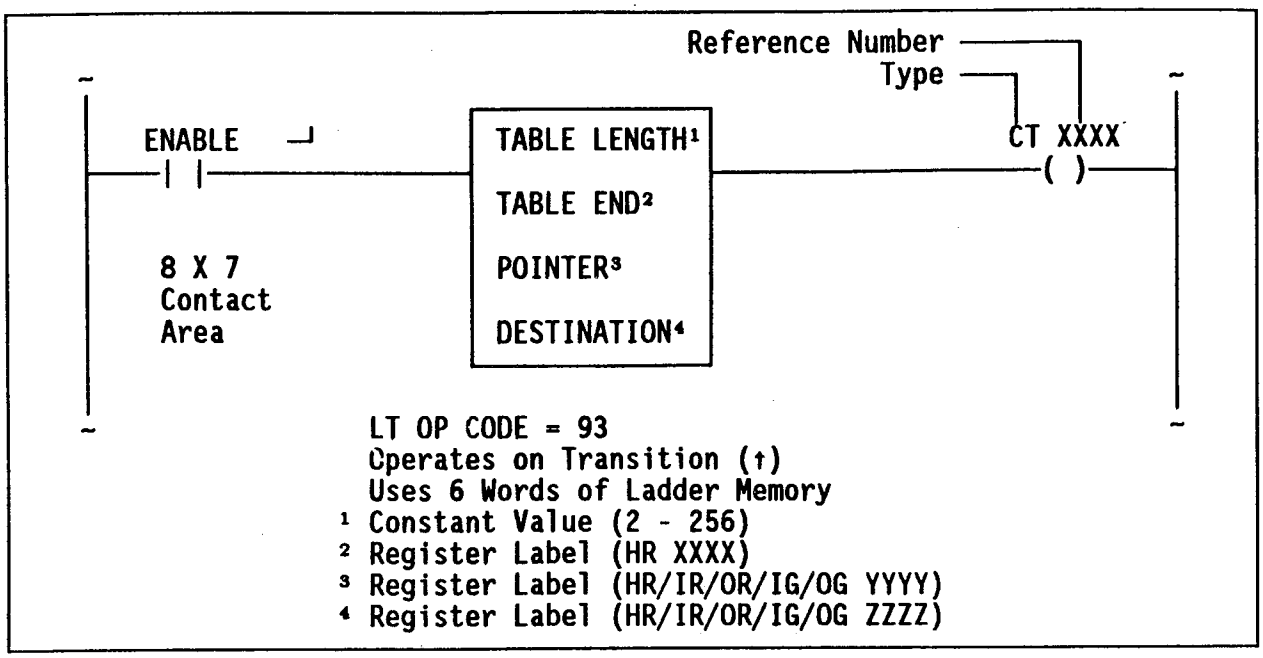


Figure 1b. Closed Table (CT)

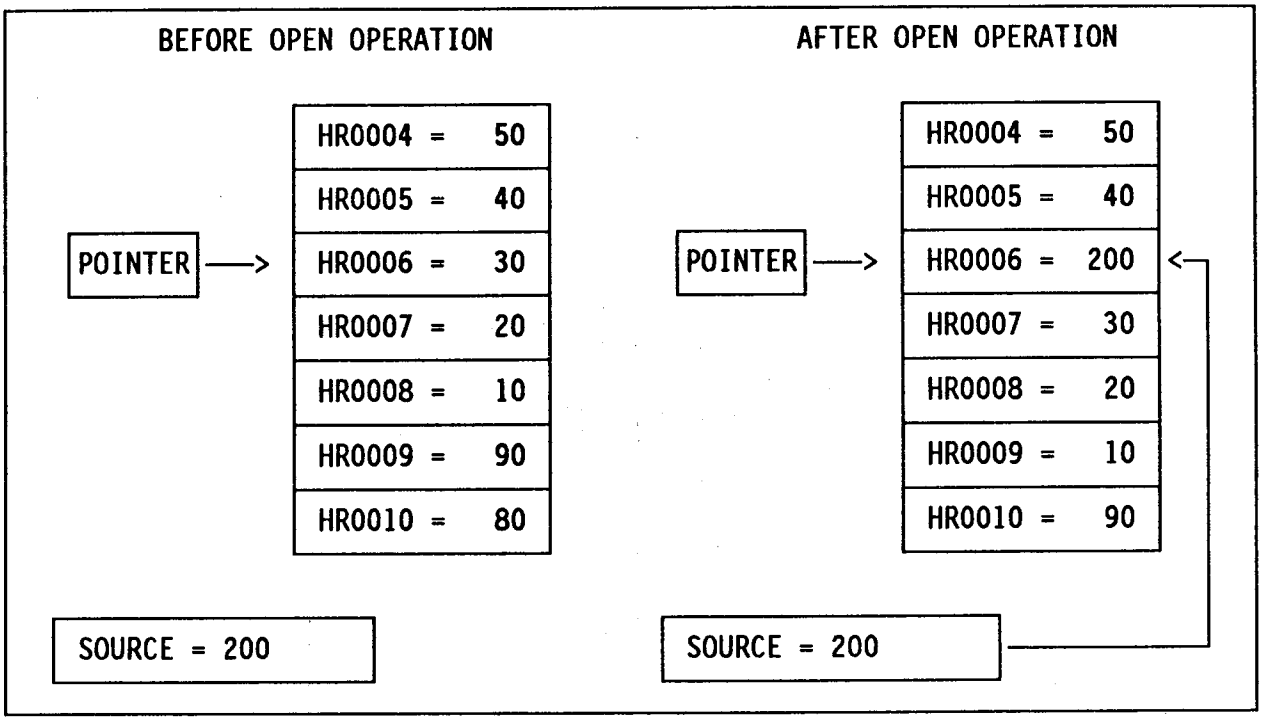


Figure 2. OT Operation

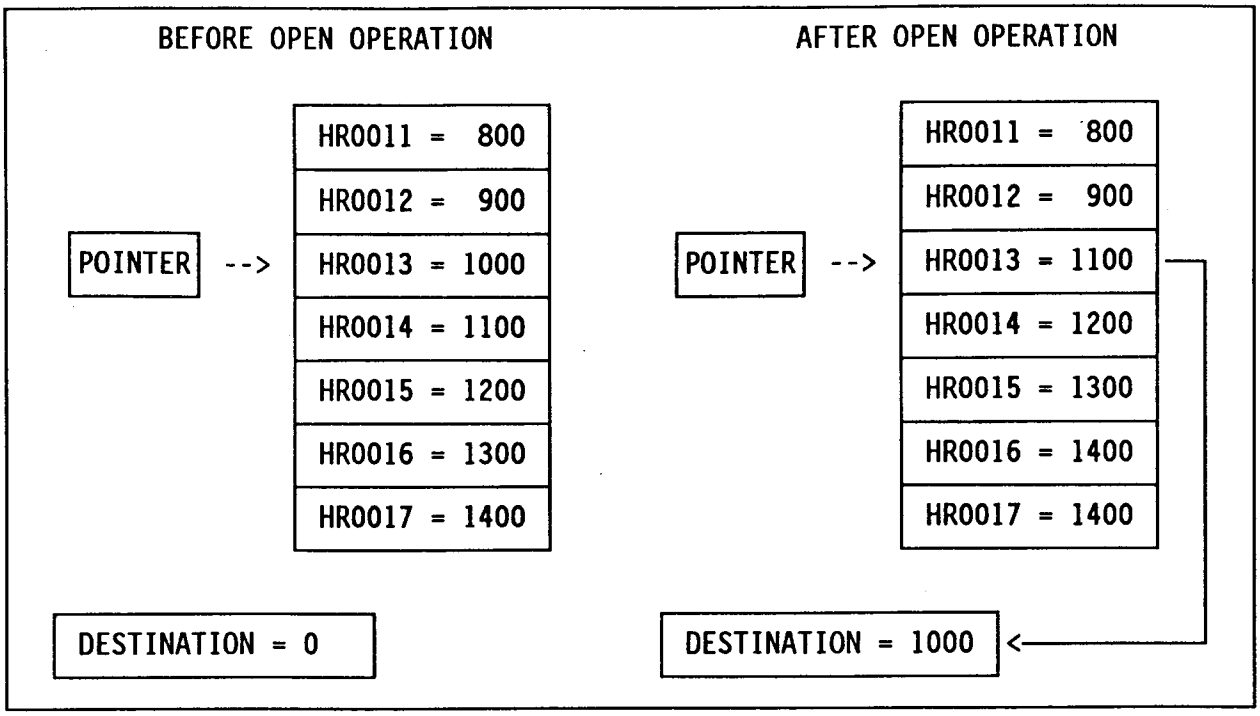


Figure 3. CT Operation

SPECIFICATIONS

OPERAND 1 - TABLE LENGTH

The table length is a constant value from 2 through 256 that determines the length of the table being opened or closed.

OPERAND 2 - TABLE END

The table end defines the holding register number of the last register in the table.

OPERAND 3 - POINTER

The pointer contains the location to be opened or closed. This location is a specified register or group:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

OPERAND 4 - SOURCE/DESTINATION

The source contains the data to be inserted in the table. The destination contains the data removed from the table. The source and destination are specified registers or groups:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

OT/CT TRUTH TABLE

See Table 1.

TABLE 1. OT/CT TRUTH TABLE (Cont'd)

Enable	Result
0	No data transfers and the coil de-energizes.
↑	OT: As long as the pointer is valid (≤ table length - 1), data in the table starting at the pointed location is shifted up one location, and data in the source register is placed in the open location. Data in the last register in the table is lost. The coil is turned ON.
	CT: As long as the pointer is valid (≤ table length - 1), data is moved from the pointed location to the destination. Table locations above the pointed location are moved down one place in the table. Data in the last register in the table moves down and also remains in the last location. The coil is turned on.
1	No data transfers. If the pointer is legal, the coil is turned ON.

APPLICATIONS

The OT and CT functions are used in any situation where tabular data is subject to change. The Drum Controller (DR) and Table-to-Register Move (TR) functions are functions to which the OT and CT functions can be applied, as shown in Figure 4. With the OT function, steps can be added to the DR function. With the CT function, steps can be deleted from the DR function. Together, a step may be removed, and a new step may be inserted at the old location. The table must be the same in all three functions.

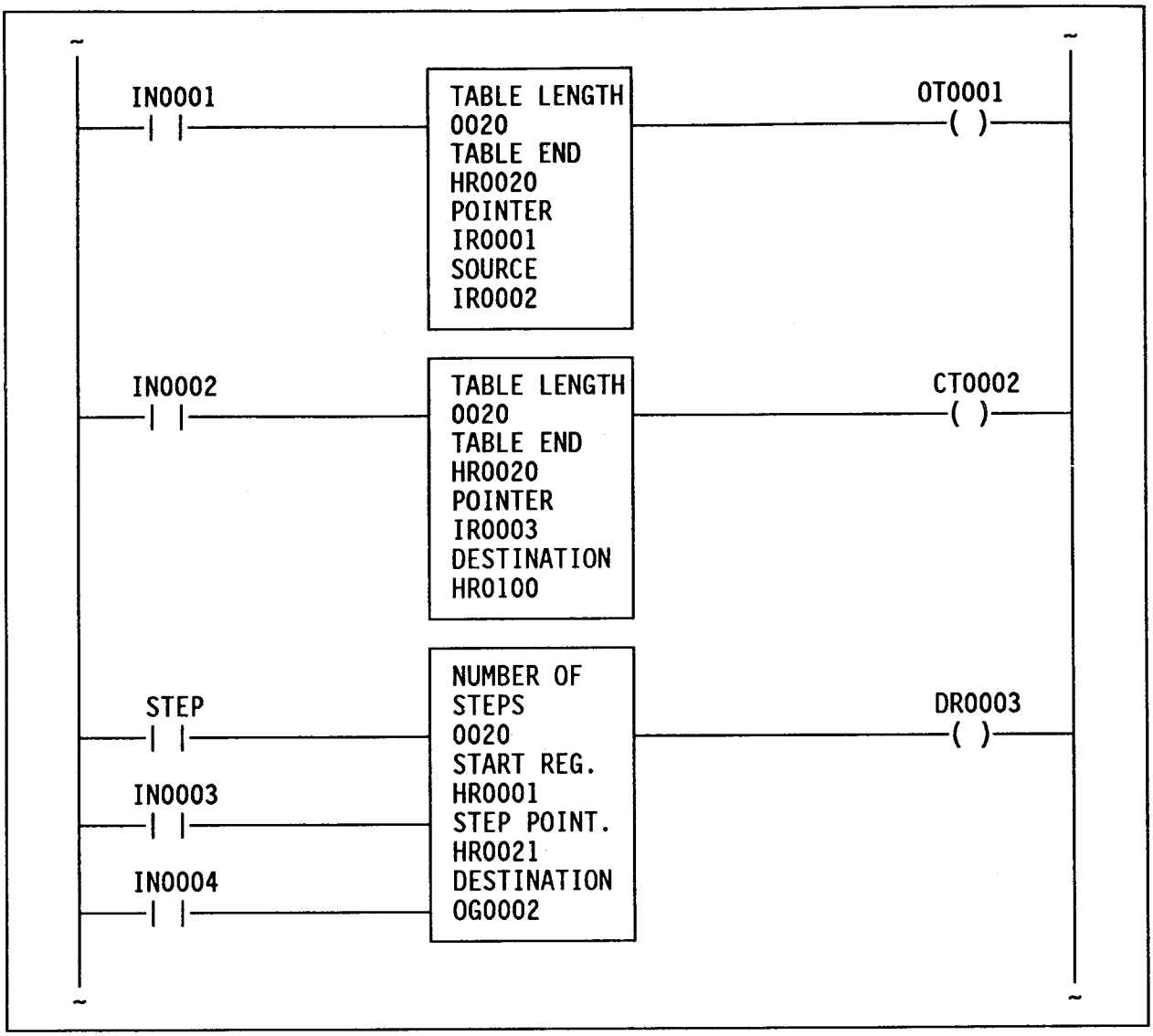


Figure 4. OT/CT Application

PT - PORT TRANSMIT

Modified for PC-1200

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: NOT SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

Port Transmit is one of three special functions used in networking. Refer to Section 3 for an overview of networking and to the CP (Configure Port) and UA (Unit Address) special function descriptions in this section for additional information.

Port Transmit (PT) designates the master PC on a Multi-Point network. The Master PC can read registers from and write registers to any Slave PC connected to the network. The user controls the communications through the ladder program in the Master by manipulating the PT function. Refer to Figure 1.

In the PC-1100 and PC-1200, four operands specify the number of registers to transfer, the location of the registers at the source and destination PCs, and the slave address. By programming the address register, this operand defines the location of one or several "indirect operands" relating to communication status, slave status, response time out and more. Refer to Tables 3a and 3b. These indirect operands aid in tailoring the network and in troubleshooting.

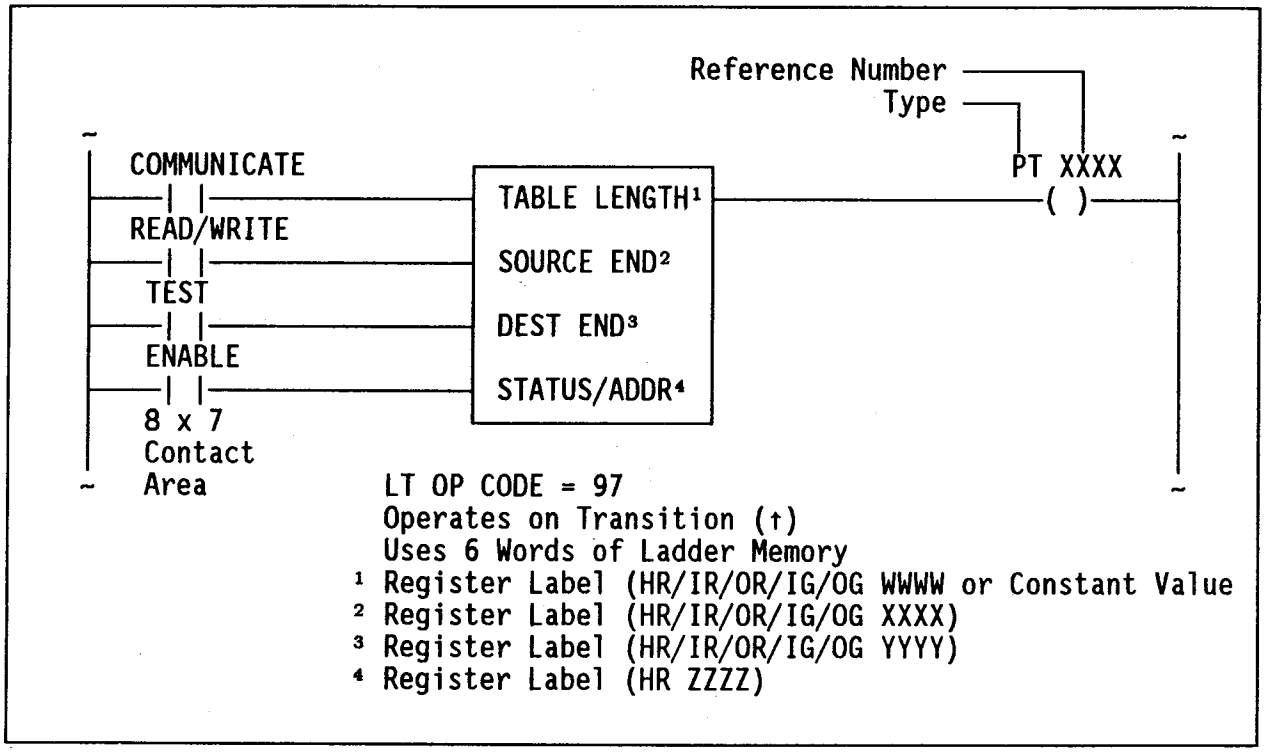


Figure 1. Port Transmit (PT)

PT initiates a communication session on a false-to-true transition of the COMMUNICATE input. Data transfer is independent of the processor scan. PT coil energizes during the communication session, which may last several scans. The coil is de-energized when data transfer is complete, or upon error detection.

To establish communications on the link, the Master transmits a six byte "set PC address" command. Each Slave on the link compares the address with it's own unique address that is programmed using the Unit Address (UA) special function. Only the Slave whose address matches will respond to the Master and data transfer is initiated. All communications on the network are error checked using the Westinghouse six byte protocol. Refer to the "Programmable Controller Communications Manual" (NLAM-B58) for more information.

An error on the link will be detected by the Master PC and an error bit will be set in the upper byte of the status register to identify the error. When an error is detected, data is not transferred.

OP CODE

Op Code 97 defines the Literal (LT) special function as a PT function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - TABLE LENGTH

The table length establishes the number of holding registers to be transmitted or received upon execution of the PT function. It can be defined by a constant value or allowed to vary by changing the data contained within a designated register. The table length register is subject to the limits given in Table 1.

Note

The highest holding register reference number acceptable depends on both the memory and user program size.

TABLE 1. TABLE LENGTH

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
CV	≤ 64	≤ 64	≤ 64	≤ 64
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
IG	≤ 4	≤ 4	≤ 8	≤ 16
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

OPERAND 2 - SOURCE END

The source end pointer is a register in which the reference number of the last holding register in the source table is found. This reference number must be at least as large as the table length and can be as large as EOP (end of program) minus 1.

- Read The holding registers within the Slave PC read by the Master PC.
- Write The holding resisters within the Master PC which are written to the Slave PC.

The source end pointer is subject to the limits given in Table 2.

Note

A reference number of zero will generate a user software fault in bit 7.

Change the source table end reference number only when transmission is complete, as indicated by the coil de-energizing.

TABLE 2. SOURCE END POINTER

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
IG	≤ 4	≤ 4	≤ 8	≤ 16
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

OPERAND 3 - DEST END

The destination end pointer is a register in which the reference number of the last holding register in the destination table is found. This reference number is subject to the same limitations as the source end pointer. Refer to Table 2.

- Read The holding registers within the Master PC which stores data from the source table within the Slave PC.
- Write The holding registers within the Slave PC which stores data from the source table within the Master PC.

OPERAND 4 - STATUS/ADDR

The Status/Address register, specified by Operand 4, selects the communications address of the Slave PC on the multipoint network with which the Master PC will establish communications.

The program loader displays this fourth operand as a pair - "Status/Address", however for clarity, the fourth operand defines the Slave Address register and marks the end of a consecutive pair of Holding Registers in the PC-1100 and the end of a consecutive table of four Holding Registers in the PC-1200. These extra registers are "indirect operands" used for extra network control and monitoring communications.

The high byte in Operand 4 has two meanings:

PC-1100: If the slave addressed responds with a fault status, the slave's Fault Register High Byte will be displayed in the high byte of Operand 4.

PC-1200: The high byte indicates a retry number the master will reference in case of communication errors. Valid range is 0 - 255.

TABLE 3a. INDIRECT OPERANDS FOR PC-1100

Operand	Holding Register	High Byte	Low Byte
Indirect Operand 1	HR XXXX - 1	Status	Error code/Fault Low Byte Fault information available only in ADV units V3.6 or greater.
Operand 4	HR XXXX	Slave Fault - Fault information available only in ADV units V3.6 or greater.	Slave Address

TABLE 3b. INDIRECT OPERANDS FOR PC-1200

Operand	Holding Register	High Byte	Low Byte
Indirect Operand 3	HR XXXX-3	Slave response time-out (1 - 32767 mSec.)	
Indirect Operand 2	HR XXXX-2	Slave Fault, high byte	Slave Fault, low byte
Indirect Operand 1	HR XXXX-1	Status	Error code
Operand 4	HR XXXX	Retry count	Slave Address

INDIRECT OPERAND 1 - STATUS (PC-1100 and PC-1200)

The upper byte of the Status Register indicates the communication status. Bits 9 through 16 act as flags indicating what stage the communication session is in, Mode status of Slave, and error state if a data link or programming misapplication occurs. See Figure 2.

If the error bit 9 is set then the lower byte will contain an error code which can be compared to Tables 4 and 5 for troubleshooting. No data will be transferred.

If bit 10 is set then the slave is not in Run or Run Program Protect Mode and is either in stop or in fault. *Information will be transferred!* Stop or Fault mode may have information which is of interest to the user and the program must distinguish whether to discard information read or set an alarm.

If the slave responds in fault both the PC-1100 and PC-1200 will provide to the Master PC's program the fault register contents (PC-1100 Master ADV models V3.6 or greater; all PC-1200s). The PC-1100 splits the register with low byte placed in the low byte of the status register and high byte in the high byte of the address register. The PC-1200 places the fault register into Indirect Operand 2.

The PT function stage of execution is evident by the bit in the status register at a logic value of 1. The bits are defined in Figure 2.

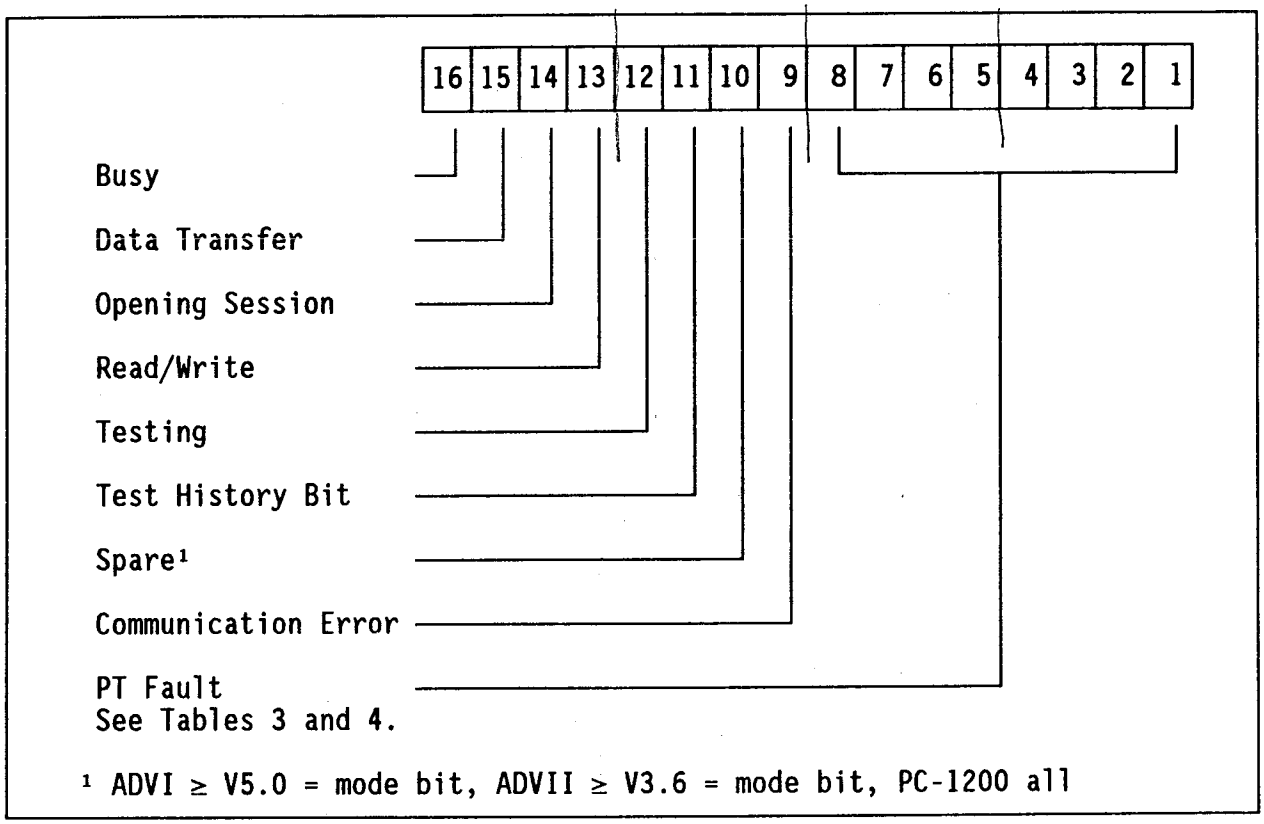


Figure 2. PT Status Register

TABLE 4. STATUS REGISTER ERROR CODES FOR MASTER PC

Error Code	Description
01H	Attempted to write into the write protected RAM area.
02H	Invalid data transfer detected.
03H	Checksum error detected.
04H	Command overrun.
06H	Uart overrun.
08H	Uart framing error detected.
09H	Uart parity error detected.
0BH	Invalid port transmit status detected.
0CH	Master time-out, transmitter buffer is not empty at the Master PC.

TABLE 5. STATUS REGISTER ERROR CODES FOR SLAVE PC

Error Code	Description
81H	Attempted to write to the Slave PC ladder while in run mode. *
82H	Invalid data transfer command detected. *
83H	Checksum error detected. *
84H	Command overrun. *
85H	Command aborted. *
86H	Uart overrun detected. *
87H	Invalid address in Slave PC. *
88H	Uart framing error detected. *
89H	Uart parity error detected. *

* = This error is associated with a two byte error code response from the Slave PC.

TABLE 5. STATUS REGISTER ERROR CODES FOR SLAVE PC (Cont'd)

Error Code	Description
8AH	Slave time-out (no response from the Slave PC).
8FH	Response received from the Slave PC upon initiating TEST.
C0H	The transmit time-out parameter exceeds 32,767 milliseconds. The user has specified, in CP, a transmit delay which when added to the time necessary to transmit the data at the selected baud rate exceeds 32,767 milliseconds
C1H	The receiver time-out parameter specified by the user, exceeds 32,767 milliseconds (See Indirect Operand 1, Operand 4)
C2H	PT function programmed incorrectly.

INDIRECT OPERAND 2 - SLAVE FAULT REGISTER (PC-1200 MASTER ONLY)

If the Slave addressed is in fault, the Slave's fault register will be displayed in Indirect Operand 2 until the next communication session.

INDIRECT OPERAND 3 - SLAVE RESPONSE TIME-OUT (PC-1200 MASTER ONLY)

The Response time-out is programmable to allow for variable time-outs depending on system needs. This register specifies the maximum time, in milliseconds that the Master will wait following the transmission of its request for the slave to reply.

The normal valid range for this parameter is 1 to 32,767 milliseconds. A value of 0 will default to the PC-1100 standard of 0.5 seconds.

Note

Any transmit delay programmed in the slave must be allowed for in the slave response time-out delay.

PC-1100s V3.6 or greater allow for programming this slave response time-out delay via the CP special function.

The CP special function provides for the programming of a complementary transmit delay useful in certain modem type communications. See the CP description in this Section.

CONTROL LINE DEFINITIONS

COMMUNICATE LINE

This line starts a communication session when a low to high transition is sensed. A circuit analogous to a fast clock coil can be programmed to keep the function cycling as fast as possible. A normally-closed contact is programmed in series with the communicate line, referenced to the functions own coil. See the example in Figure 3.

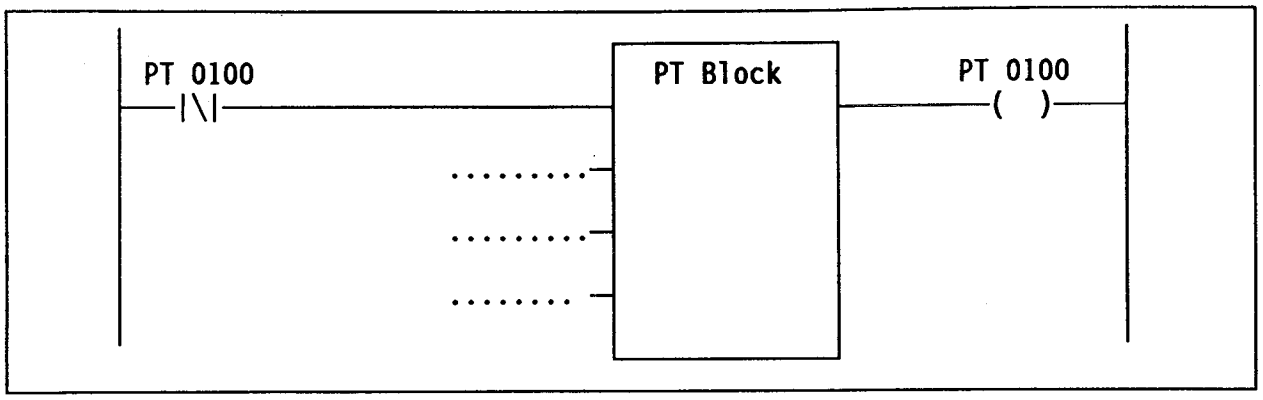


Figure 3. Cycling the PT Function

READ/WRITE LINE

A high state on this line indicates a Read transaction. Read is defined from the perspective of the Master where data originates from the Slave source and is transferred (read to) the Master destination registers.

A low state on this line indicates a Write transaction. Write is again defined from the perspective of the Master where data originates from the Master source registers and is transferred (written) to the Slave destination registers.

Note

Depending on the status of the Read/Write line the Source and Destination operands will switch reference between Master and Slave.

TEST LINE

This line is toggled with the Communicate line low and Enable High to test the status of the slaves on the network. It is meant as a troubleshooting guide to indicate improper set up of slaves on the network. The following events occur when test is invoked.

- The Master generates a synchronization pulse.
- The busy and testing bits are set to a logic one (high) in the status register.

- The coil energizes to indicate that testing is in progress.
- A special six byte "Set PC Address" command set to address = 0 is sent over the network with no response expected.

As a result of the test, if a response is received from any Slave, an error code is written to the low byte of the status register and the coil is de-energized to indicate that the test is complete. To clear the error, check all Slaves for proper UA programming and DIP switch settings.

If a response is not received within 2 seconds, the busy and testing bits are reset to zero, and the coil is de-energized, indicating that the test is complete.

ENABLE LINE

The Enable line must be high for the function to operate. Pulling Enable low in the middle of a communication session aborts the transaction.

PT TRUTH TABLE

See Table 6.

TABLE 6. PT TRUTH TABLE

Comm	Rd/Wr	Test	Enable	Results
X	X	X	0	The coil de-energizes. Any communication stops and the status register is cleared.
0	0	0	1	The coil status and communication sequence do not change.
↑[1]	0	0	1	The coil energizes. Data is moved from the Master PC to the Slave PC associated with the communications address stored in the Operand 4.
↑[1]	1	0	1	The coil energizes. Data is moved from the Slave PC associated with the communications address stored in Operand 4 to the Master PC.
0	X	↑[1]	1	The coil energizes. The multipoint network is synchronized. Network test is initiated.
↑[1]	X	↑[1]	1	The coil energizes. The multipoint network is synchronized. Network test is initiated.
↑[1]	X	1	1	The coil de-energizes. No new communication sequence can be initiated until TEST input line de-energizes.

[1] = Transition from Off to On.
X = Don't Care

APPLICATION EXAMPLE

OVERVIEW

The example in Figure 4 shows the core of a Master communication program. The key to its operation is the fast-clocking of the COMMUNICATE circuit with the PT coil. This method allows for one scan down time between each communication session which may take several scans.

The Table-to-Register-Move (TR) special functions step together during the down scan, resetting all the operands for the next session. When the end of the tables are reached, they wrap around for a continuous update through all slaves. The TR special functions create a matrix as depicted in Table 7. Each row represents one set of operands and Rd/Wr control for one communication session.

SEQUENCE OF OPERATION

Consider the PT function in Figure 4. It resides in the Master PC of the network. Upon a low to high transition of the COMMUNICATE circuit, with the TEST circuit low and the ENABLE circuit high, the following events occur in order.

Opening Session (Set PC Address)

1. The Status register is cleared of any error or fault codes.

2. The "Busy" and "Opening Session" bits in the status register are set to logic level of "1".

3. The Master PC generates a synchronization pulse on the RS-232 DTR line and through the RS-485 synchronization transceivers to cause all slaves to be ready for pending address information.

4. The status of the "RD/WR" input is latched.

5. The coil is energized to reflect that communication is in progress.

6. A six byte Set PC Address command is transmitted over the network. Each Slave compares its unique communication address with this command.

    a. If the address matches, Port B of the Slave PC is enabled for multipoint communications and a Set PC Address response is returned to the Master PC. In this example, Holding Register 17 may indicate a slave address of 1 to 5.

b. If the wrong response, corrupted response or no response is received within the slave response time-out limit, then the Master PC may retry based on the retry limit. If the Retrys and response limits expire the PT function indicates a communication error in the status register.

7. Once the correct Set PC Address Response is received then the Master PC will continue to the Data Transfer section.

Data Transfer (Block Read or Write)

1. The Opening Session bit in the status register is reset to zero.

2. The Data Transfer bit in the status register is set to a logic level 1.

3. A Six Byte Block Read or Write command is issued over the network in accordance with the latched state of the Rd/Wr circuit and the values within the Table Length, Source End, and Destination End operands.

   a. Block Read: If the response and data are not received within the time out period (default is 0.5 sec) an error is logged in the status register and the opening session is re-initiated. This sequence will repeat according to the retry limit until expired and the status register will reflect the last failure mode.

      If the response and data are received without error, the source table in the Slave PC is moved to the destination table in the Master PC. The Busy and Data Transfer bits are reset to zero, and the coil is de-energized to indicate that communications is complete.

   b. Block Write: If the response is not received within the time out period an error is logged in the status register and the opening session is re-initiated. This sequence will repeat according to the retry limit until expired and the status register will reflect the last failure mode.

      If the response is received without error, the source table in the Master PC is moved to the destination table in the Slave PC. The Busy and Data Transfer bits are reset to zero, and the coil is de-energized to indicate that communications is complete.

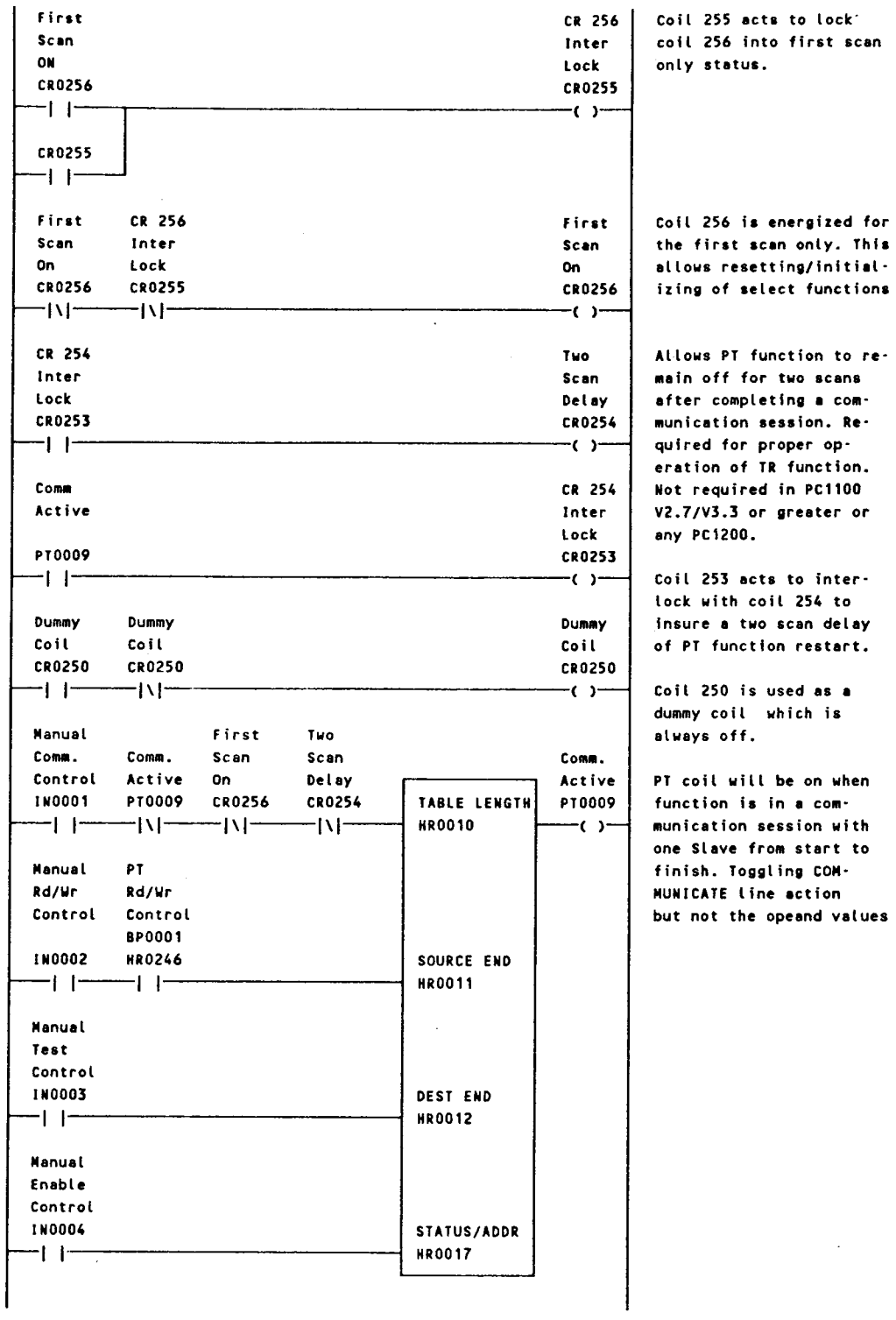


Figure 4a. PT Example

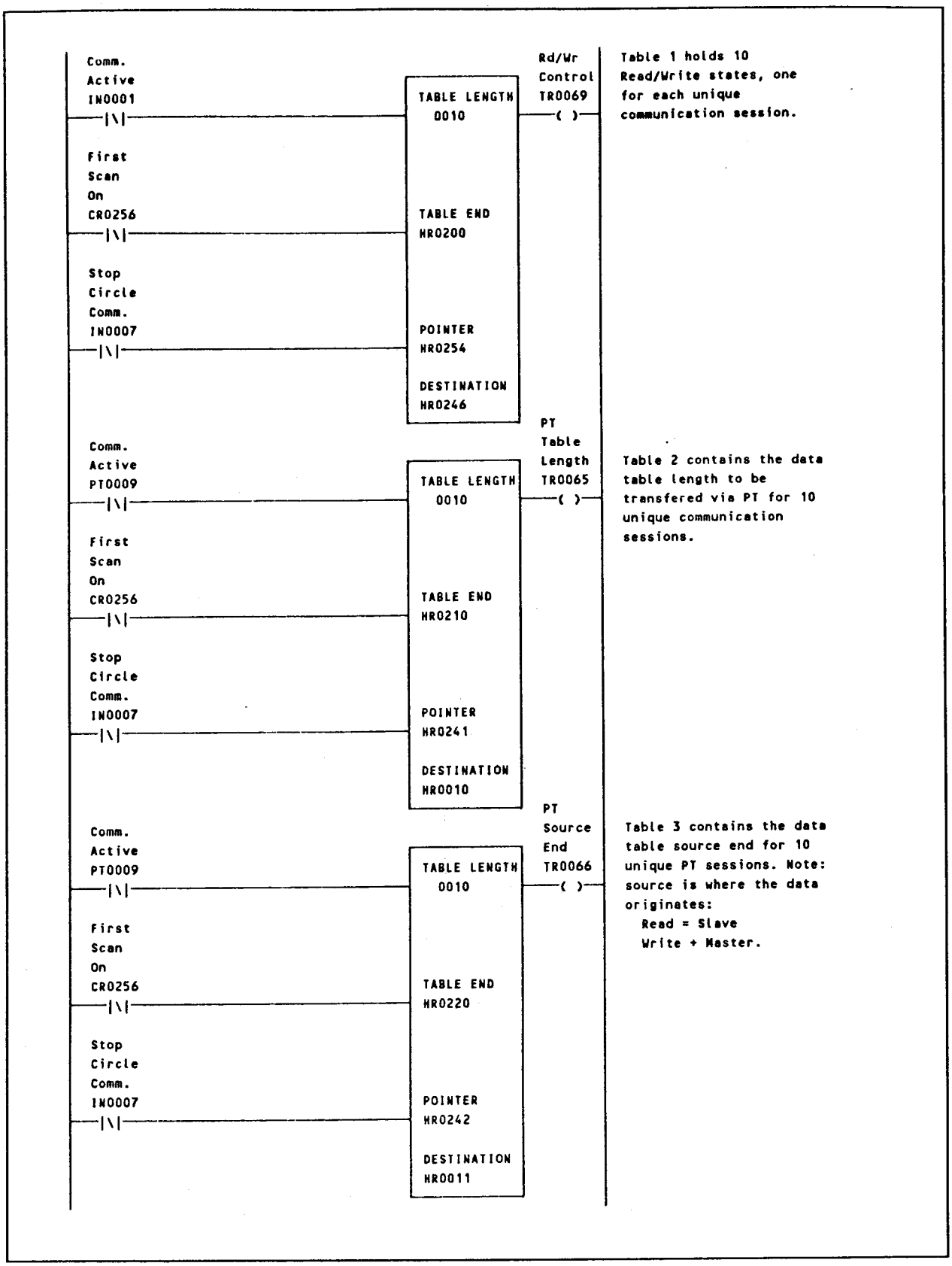


Figure 4b. PT Example

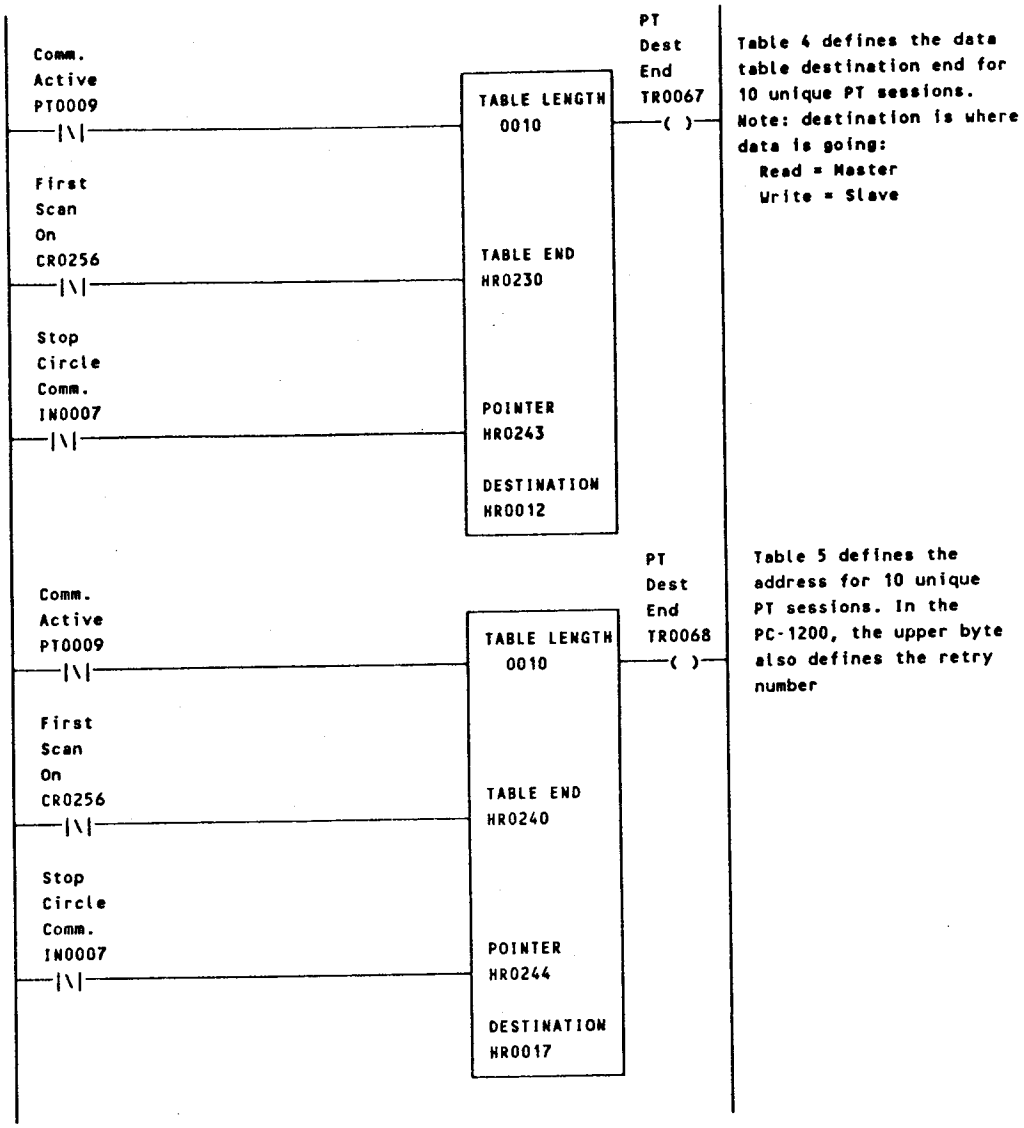


Figure 4c. PT Example

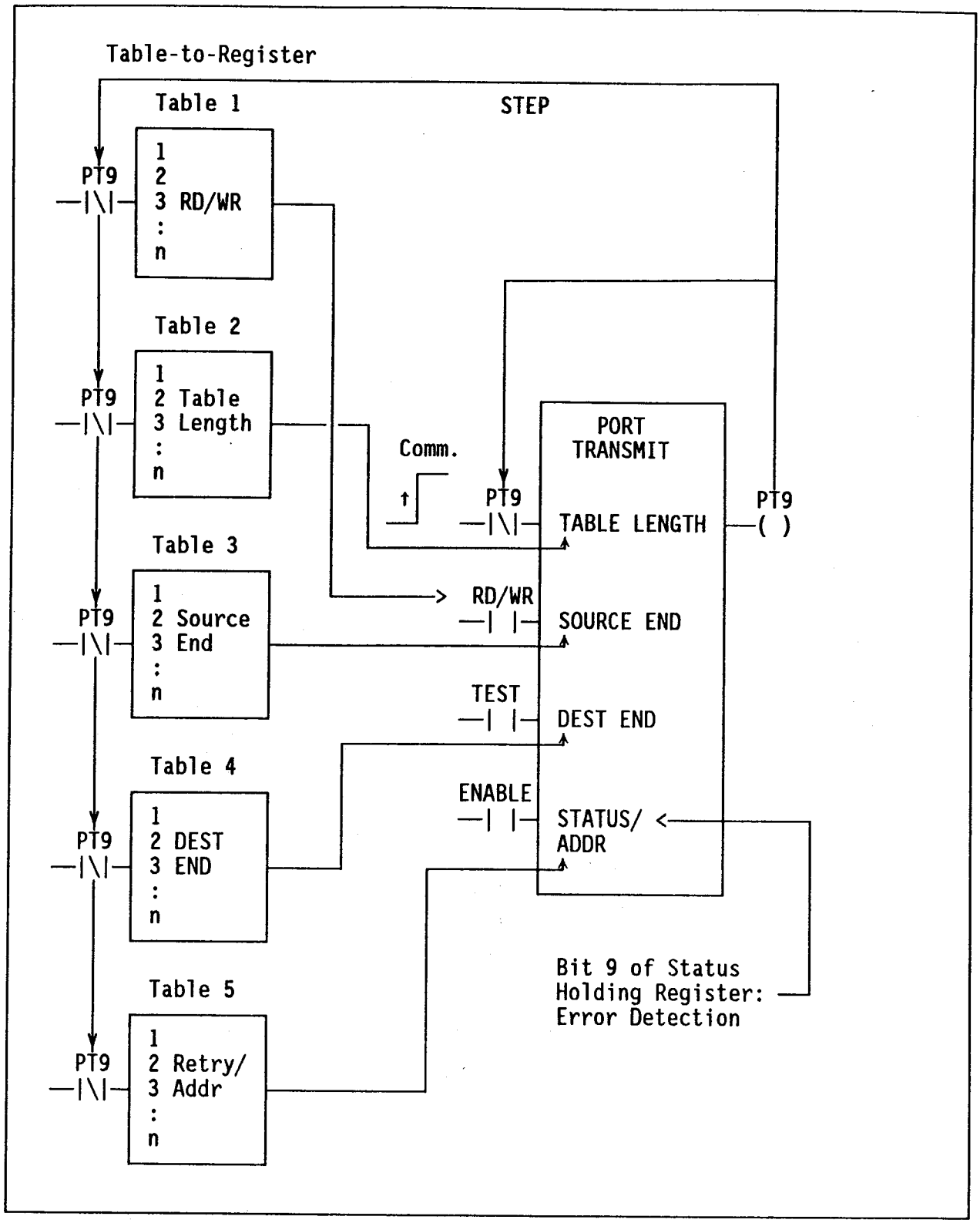


Figure 5. PT Example Block Diagram

In Table 7, each row represents one communication session. The retry variable is used only in the PC-1200. It is easiest to set this value with a hexadecimal number since the register is split in half byte entries. The retry rate for the PC-1100 is factory set at 3 retries and cannot be changed by the user.

TABLE 7. TR MATRIX FOR EXAMPLE

TR Pointer	Rd/ Wr	Table Length	Source End	Dest End	Retry High Byte	Addr Low Byte
00000	1	00020	00220	00420	02 H	01 H
00001	0	00020	00320	00320	02 H	01 H
00002	1	00010	00210	00430	01 H	02 H
00003	0	00010	00330	00310	01 H	02 H
00004	1	00040	00240	00470	01 H	03 H
00005	0	00040	00370	00340	01 H	03 H
00006	1	00020	00220	00490	00 H	04 H
00007	0	00010	00380	00310	00 H	04 H
00008	1	00010	00210	00500	03 H	05 H
00009	0	00020	00440	00320	30 H	05 H

MULTIPOINT NETWORK THROUGHPUT TIME

The multipoint network throughput time is measured from the moment the Port Transmit (PT) function initiates communications until the coil de-energizes. Four variables effect the time to transmit data:

- Baud rate
- Master PLC scan time
- Slave PLC scan time
- Length of data block to transfer

The maximum throughput time and typical throughput time can be calculated by using the following equations.

THROUGHPUT TIME EQUATIONS

Maximum = 3 ("Master" scan time) + 2 ("Slave" scan time) + 3W + X + 2.3 mS

Typical = 1.5 ("Master" scan time) + ("Slave" scan time) + 3W + X + 2.3 mS

where:

$$W = \frac{(\#\ \text{bits/byte})\ (6)}{(\text{baud rate})} + (.0012)(N) \quad \text{seconds}$$

$$N = \frac{(\#\ \text{bits/byte})\ (6)}{(\text{baud rate})\ (\text{scan time})} \quad \text{(round to the next highest integer)}$$

$$X = \frac{(\#\ \text{bits/byte})\ (\#\ \text{bytes/data block})}{(\text{baud rate})} + (.0012)(M) \quad \text{seconds}$$

$$M = \frac{(\#\ \text{bits/byte})\ (\#\ \text{bytes/data block})}{(\text{baud rate})\ (\text{scan time})} \quad \text{(round to the next highest integer)}$$

Calculation Notes

The # bits/byte transmitted is 12 (1 start bit, 8 data bits, 1 odd parity bit, and 2 stop bits) unless formatted otherwise by the Configure Port (CP) function.

The # bytes/data block transmitted is two times the PT table length plus the seven overhead bytes required by the Block Read and Block Write commands.

N and M represent correction factors for CPU overhead to handle the communications task. Choose the longest scan time between the "Master PC" and the "Slave PC" to calculate N and M.

SAMPLE THROUGHPUT CALCULATIONS

Assume 12 bits/byte, table length of 10 holding registers, "Master PC" scan time of 50 msec, and "Slave PC" scan time of 25 msec.

1200 Baud Example

N = 1.2 rounded up to 2.0

M = 5.4 rounded up to 6.0

W = .0624 seconds

X = .2772 seconds

Typical = 1.5(.050) + (.025) + 3(.0624) + .2772 + .0023
= 0.5667 seconds

Maximum = 3(.050) + 2(.025) + 3(.0624) + .2772 + .0023
= 0.6667 seconds

32.5 Kbaud Example

N = .04 assume 0.0

M = .20 assume 0.0

W = .0022 seconds

X = .0099 seconds

Typical = 1.5(.050) + (.025) + 3(.0022) + .0099 + .0023
= 0.1188 seconds

Maximum = 3(.050) + 2(.025) + 3(.0022) + .0099 + .0023
= 0.2188 seconds

To determine the throughput for a network made up of several slaves, perform the calculations above for each slave. Total all of the calculations. Add the time for one "master" scan for each slave.

1200 baud example for 10 slaves

Typical throughput for each slave is 0.5667 seconds from above. Assuming each slave is communicated to in turn, network throughput is:

Network Throughput = 10(.5667) + 10(.050) = 6.167 seconds

SYSTEM CHARACTERISTICS AND APPLICATION NOTES

1. A Slave addressed by the Master will continue to communicate if the Slave is in the stop program or fault mode. Communications with the UART circuitry are not disabled, but these conditions may affect the associated I/O. The application program can be designed to detect when the slave unit is not running.

2. When the Master detects a communications error during a session, it sets an error code in the status register of the PT function and no data is transferred. Cycling the COMMUNICATE input to the PT function will cause the next communication session to be attempted. The Link Test line may be used to assist in troubleshooting. The communicate line must be de-energized when doing a link test.

3. When a Slave is powered up or has a keyswitch transition from program to run, the PLC initializes the UART of the communications circuitry which may interfere with communications on the link. This interference may be detected as an error by the Master. This is a temporary event, however in general it is recommended that before a Slave is powered up or put into the run mode, communications should be halted by disabling the enable line of the PT function within the Master.

4. Each end of the RS-485 link must be terminated. (As explained in the NL-1075 Communications Adapter I.L 15753).

5. The Master will detect an error if a break or disconnection in the link exists. Whether the Master detects the fault depends on the location of the Slave that it is communicating to, and it's relation to the open point of the cable. If the Slave is located after the open, a communications error will result. If continuity exists, communications may continue to that Slave depending on the transmission line effects of the non-terminiated cable.

6. The APL or CRT loader may be used on the A port of any PC to monitor or change register data while communicating on the B port in either Single-Point or Multi-Point mode.

7. The characteristics of the RS-485 E.I.A. Standard restricts communications to 31 Slaves. Modem communications has the advantage, however, of allowing up to the maximum number of slaves the Unit Address function will support (255). Modem communication requires the use of an Asynchronous, multipoint type modem. The AT&T 202T or compatible modem is recommended. Executive software version 2.3 or higher supports modem communications by allowing a constant value of 3 to be assigned in the first operand of the Configure Port Function. This sets the B Port up for modems. (See function description for details).

8. The PC-1100 is shipped with the termination switches on the Communications Expansion Board in the ON or terminated position. They should remain in this position if the processor is to be used in the Single-Point mode. In Multi-Point mode using RS-485, the switches should be to the OFF position for all processors except the ones on each end of the network cable. In Multi-Point mode using RS-232 (modems), the switches should be set to ON.

9. To use a computer as a master on the link, the "set PC Address" command (a 6 byte protocol) must be sent and correctly received before data transfer can occur. After the master computer receives an error free response from the slave, communications can be established. One of the advantages of having a computer act as the master is that all the commands in the Communications Manual are valid after the PC Address has been set. Refer to the Communications Manual for further information. The "Set PC Address" command has the following format:

From Master PC

0A
Slave PC Address (UA)
00
00
00
Checksum

From Slave PC (no errors)

0A
Slave PC Address (UA)
Keyswitch Mode Data (program/run)
Fault Register LOW
Fault Register HI
Checksum

PROGRAMMING TIPS

1. A normally-closed contact (referenced to the PT coil) on the communicate input line to the PT function, will reset the line to low when communication is established (the coil is energized during data transfer). After data transfer is completed, the coil de-energizes making the line high initiating communications again. This gives the user the ability in certain situations, to create a program that will transmit or receive data at the rate that data transfer is completed. (Reference the PT description for more information on how the function works.)

2. The Status Register within the Port Transmit function is primarily used for debugging of the system. Bit 9 (Communications Error Bit) will be set whenever any of the error bits in the lower byte are set. Using a Bit Pick function on bit 9 of the status register will allow flexibility in detecting a communication fault.

3. Move functions with contacts referenced to the Read/Write enable logic will set up the register table pointers in the Port Transmit function. Two Move functions could be used to set the pointers for receiving data and two Move functions for sending data.

4. The Address Register in the Port Transmit Function selects the slave that the Master will establish communications with. A Table to Register function can be used to select each slave. Each time a new unit address is moved into the address register, the communicate line of the PT function would have to be toggled from non-conducting to conducting to establish the "set PC address" command with the slave that it is addressed to. A Table-to-Register function can be easily sequenced to select the communication addresses.

5. If the pointers or address register in the Port Transmit function are changed during a communication session (while the PT coil is energized), data may be transferred from/to incorrect addresses (data tearing). The logic for setting up pointers or the slave address must be done after the PT coil is de-energized.

6. Remember that the pointers and the address registers are indirect. This means that the "actual" pointer or value is held inside the operand register. In addition the Source and Destination mark the "End" of a table not the start. Before the processor can be put in the run mode the operand registers must contain a valid pointer or value. In a PC-1100, nonsensical operand contents will cause the processor to fault with bit 7 if placed in run. The PC-1200 traps the event by halting the communication session and indicating a programming failure failure in the status register.

7. Only one PT function is allowed in the Master program.

8. Only one Master may exist in a network.

RP - RESTORE PROGRAM COUNTER

Modified for PC-1200

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Restore Program Counter (RP) function is one of three Loop Back functions. the Loop Back functions are:

- Save Program Counter (SP)
- Restore Program Counter (RP)
- Reset Watchdog Timer (RW)

Loop Back functions are used in the development of programs that are capable of repeating segments of the ladder diagram. RP function symbology is shown in Figure 1.

OP CODE

Op Code 13 defines the Literal (LT) as the RP function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

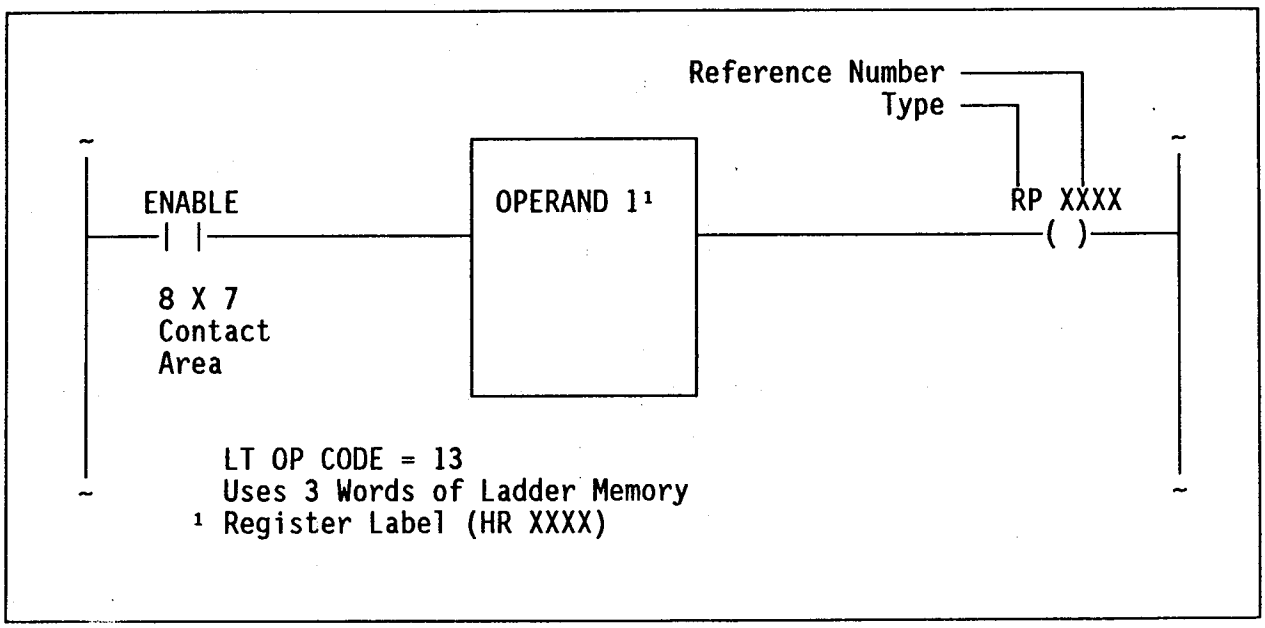


Figure 1. Restore Program Counter (RP)

SPECIFICATIONS

OPERAND 1

In the PC-1100, Operand 1 defines the holding register location from which the program counter will be taken. This is available from a previously selected SP function. The value in this location must point to the ladder memory and the first contact of a ladder function. If either or both of these conditions are not met, a "user software fault" occurs.

In the PC-1200, Operand 1 defines the holding register location from which the program counter will be taken. This is available from a previously selected SP function. The value in this location must point to the start of the network immediately following an SP function. If this condition is not met, and "Invalid Ladder Pointer" fault occurs.

RP TRUTH TABLE

See Table 1.

TABLE 1. RP TRUTH TABLE

Enable	Result
0	The coil is de-energized. Ladder execution is unaffected.
1	The coil is energized. Ladder execution starts at the program location stored in the specified registers. If the program location is invalid, the processor is placed into an "Invalid Ladder Program Pointer" fault.

APPLICATIONS

Applications for the SP and RP functions are described in this section. The Loop Back functions are especially useful in cases where a ladder segment requires several iterations during a single processor scan to complete an operation. Such an instance occurs when using a Newton approximation to perform a square root function The formula used for this calculation is:

$$\sqrt{N} = \frac{X^2 + N}{2X}$$

This formula is used in an iterative manner. An initial guess or trial number "X" is inserted into the equation; N is the number from which the square root is to be extracted. Next, the equation is solved and the result is inserted as a new trial number X. Then, the equation is solved again. Each time the equation is solved, the result approaches the square root of N. In most cases, 15 iterations should be sufficient to provide an adequate result. (See Figure 2.)

The square root program is shown in Figure 3. See also the applications in the RW description.

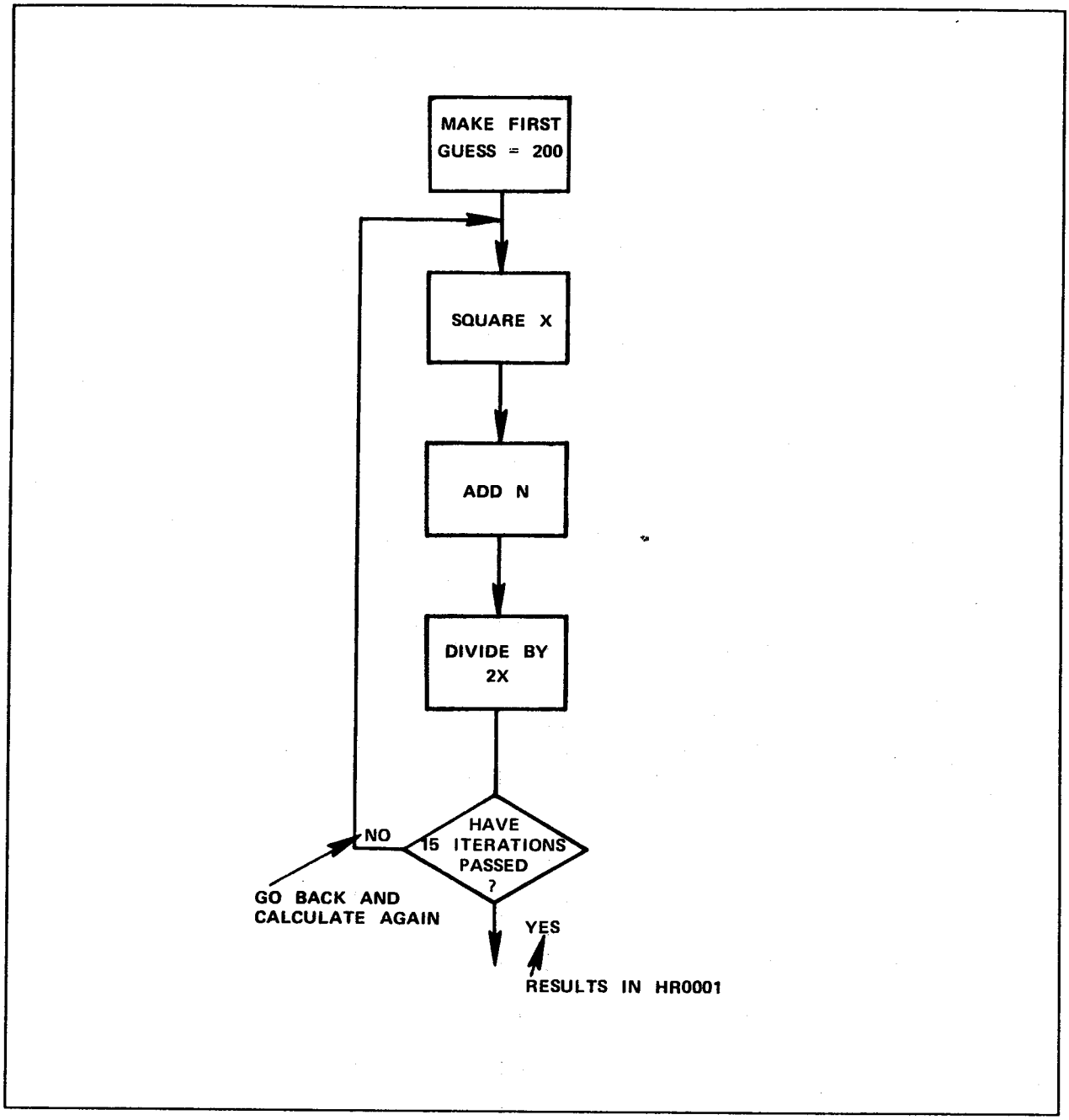


Figure 2. Square Root Flowchart

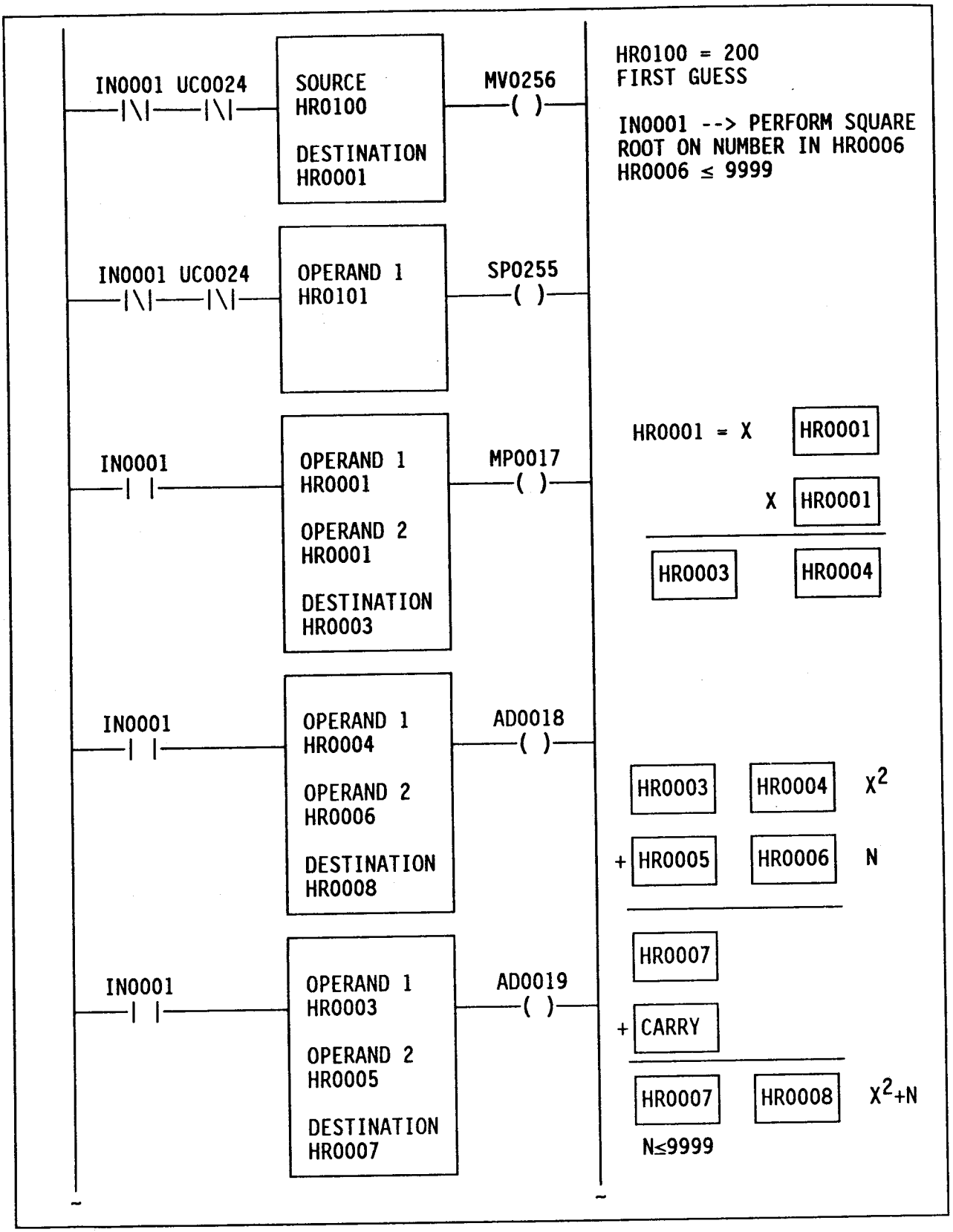


Figure 3a. Square Root Application

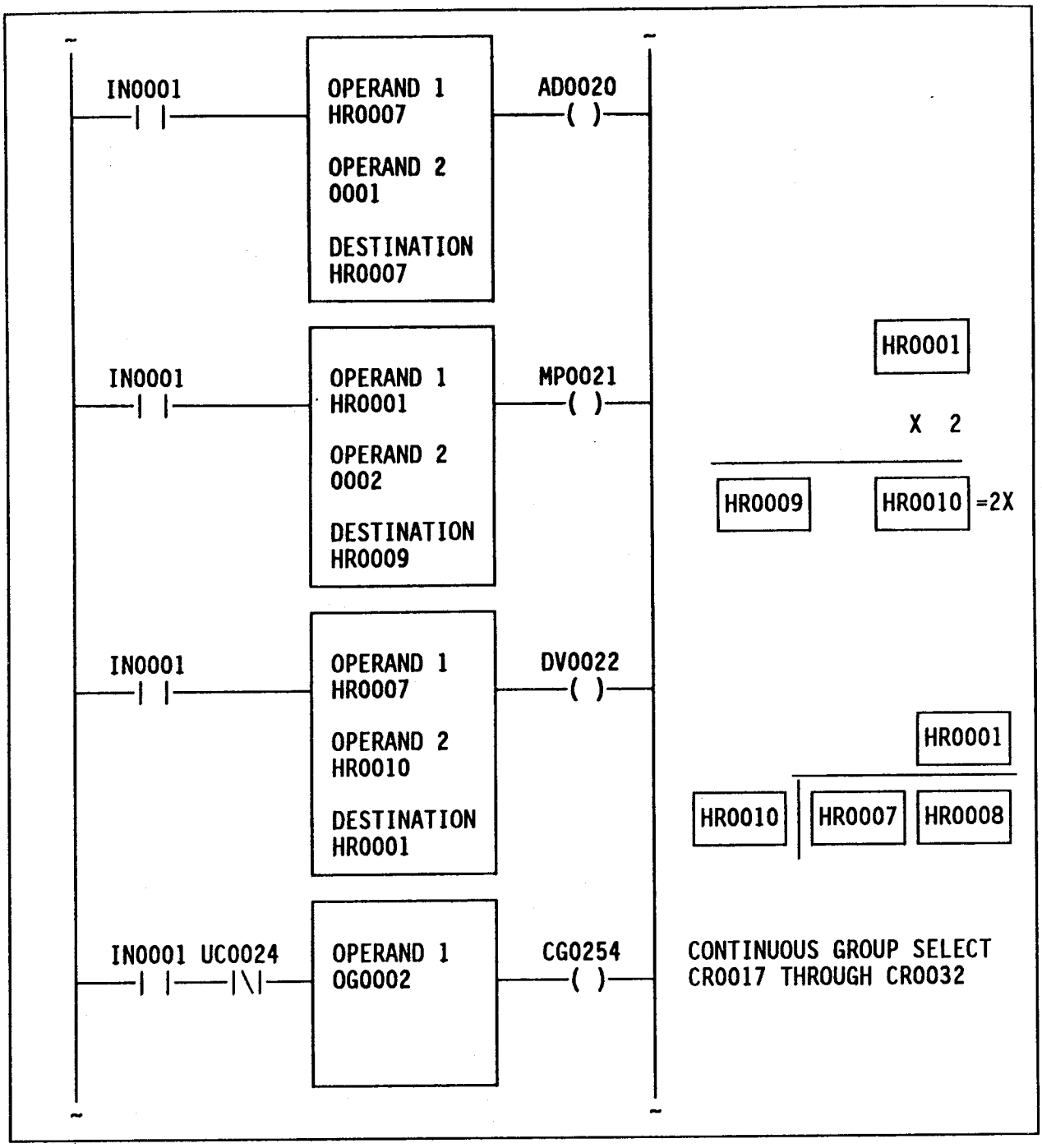


Figure 3b. Square Root Application

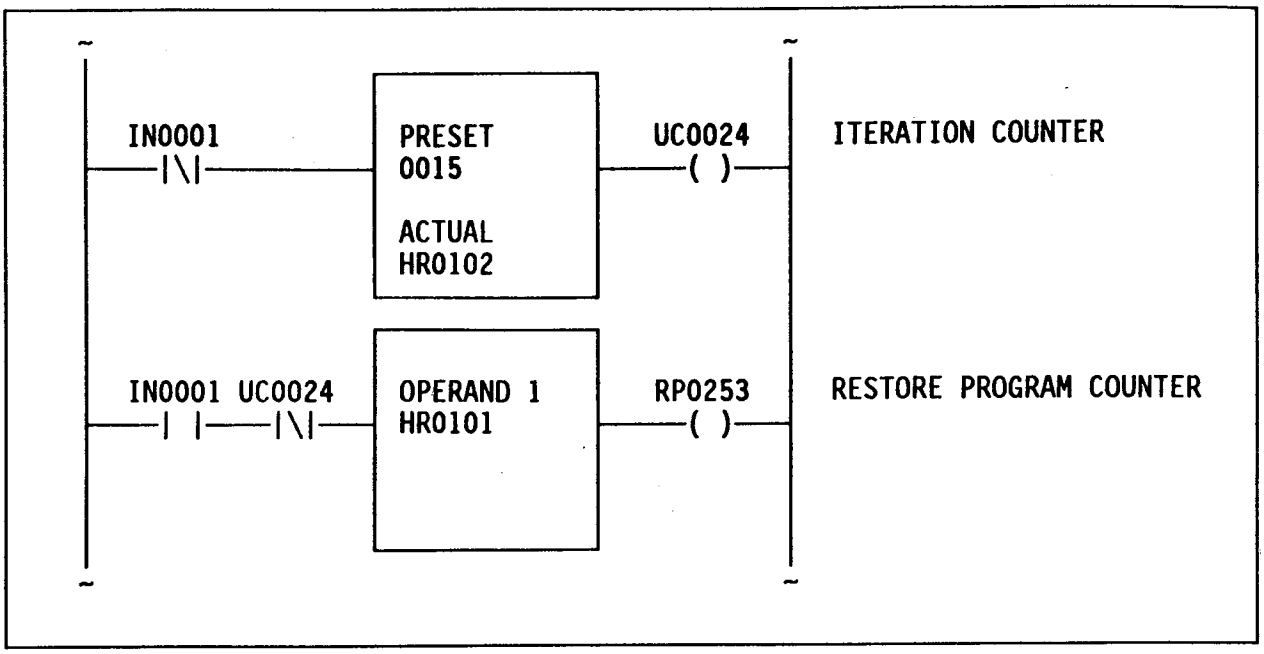


Figure 3c. Square Root Application

RW - RESET WATCHDOG TIMER

Modified for PC-1200

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Reset Watchdog Timer (RW) function is one of three Loop Back functions. The Loop Back functions are:

- Save Program Counter (SP)
- Restore Program Counter (RP)
- Reset Watchdog Timer (RW)

Loop Back functions are used in the development of programs that are capable of repeating segments of the ladder diagram. The RW function resets the Watchdog Timer and can allow scan times greater than the limit (default limit is 100 msec).

Note

In the PC-1100, use of the RW function will interfere will proper operation of the Timer functions (TS/TT). In the PC-1200, the Timers will operate properly as long as they are executed at least once every 16 seconds.

RW function symbology is shown in Figure 1.

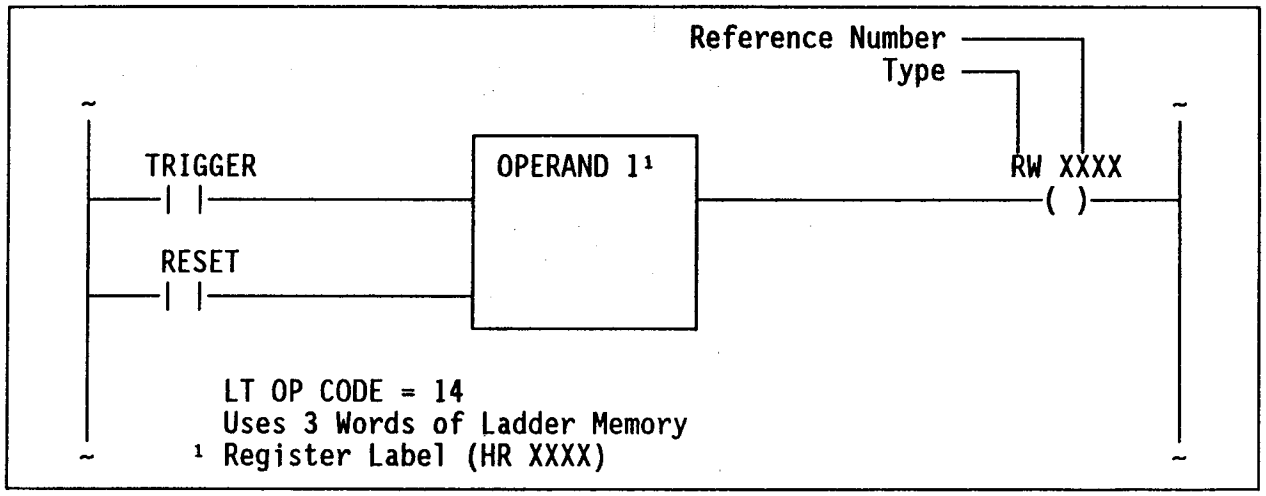


Figure 1. Reset Watchdog Timer (RW)

OP CODE

Op Code 14 defines the Literal (LT) as the RW function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1

Operand 1 is a holding register that accumulates the number of milliseconds that the RW function is active. For the PC-1100, a time base is approximated from this value using counters in place of timers.

RW TRUTH TABLE

See Table 1.

TABLE 1. RW TRUTH TABLE

Trigger	Reset	Result
0	0	Operand 1 is zeroed. The coil de-energizes. There is no effect on the Watchdog Timer.
0	1	Operand 1 depends on previous operations. The coil de-energizes. There is no effect on the Watchdog Timer.
1	0	Operand 1 is zeroed. The coil de-energizes. The Watchdog Timer is reset.
1 (PC-1100)	1	The Watchdog Timer is added to Operand 1. The Watchdog Timer is reset. The coil energizes if Operand 1 is ≥ 100. When the coil energizes, 100 msec have elapsed since the Watchdog Timer has been updated. Timers in program will be inaccurate.
1 (PC-1200)	1	The Watchdog Timer is reset. The time (in milli-seconds that the function is active) is accumulated in Operand 1. When Operand 1 ≥ 100, the coil is energized. Note: Operand 1 will roll over at 65,536. This will not affect the coil.

APPLICATIONS

In the PC-1100, an internal timer called the Watchdog Timer keeps track of the scan time and is designed to declare a fault condition when 100 msec have elapsed. The PC-1200 uses a hardware time to keep track of the scan time. This action is necessary to insure correct programmable controller operation. The SP, RP, and RW functions make it possible to put the program into a loop lasting longer than the desired 100 msec limit; the RW function can be used to prevent a scan overtime fault.

In the PC-1100, function timers are rendered inoperative when the trigger input to the function is closed. In the PC-1200, if a program loop executes for more than 16.383 seconds, other timers may become inaccurate.

Figure 2 illustrates the use of the RW function. In this system, a product is loaded from a large hopper into fiber drums. Each drum is filled with approximately 100 lb. of the product. When the fiber drum is in place, IN0001 closes and the fill loop is activated, providing that the emergency OFF switch is not activated and that the operation is not running overtime. If a fiber drum is not in place, the fill loop is disabled.

When the start PB is activated (IN3) the loop first updates the input from the scale IR0001. The loop then updates the input/output group to ensure that priority actions take place concerning IN0002, UC0002. The actual weight is compared to the desired weight (GE0019); when it is correct, the fill process stops. (See Figure 3.)

If the process takes 15 seconds or more, or if the emergency OFF switch is activated, the loop is stopped.

In Figure 3, the RW function is used to develop a timer within a loop.

In the PC-1100, timers may be effectively used only if they are not timing concurrently with the RW function. For example, in Figure 3, if heat sealable bags were used, additional timing for the sealing process would have been used following the fill loop.

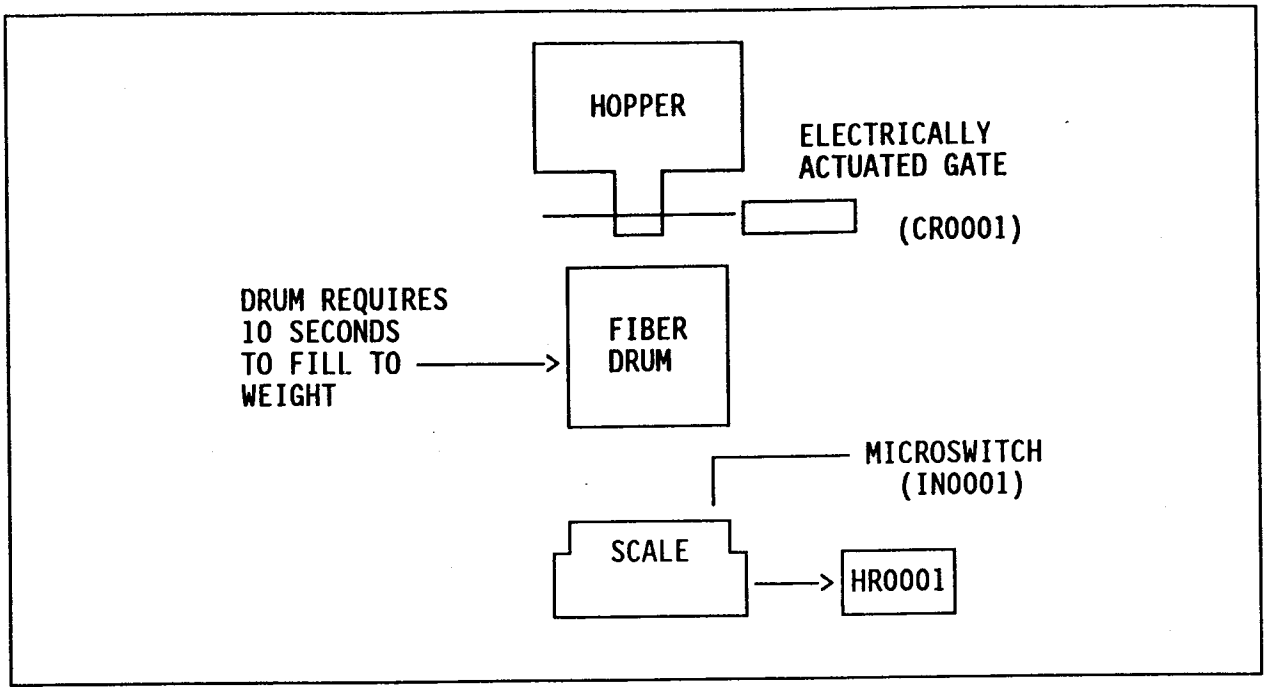


Figure 2. RW System Example

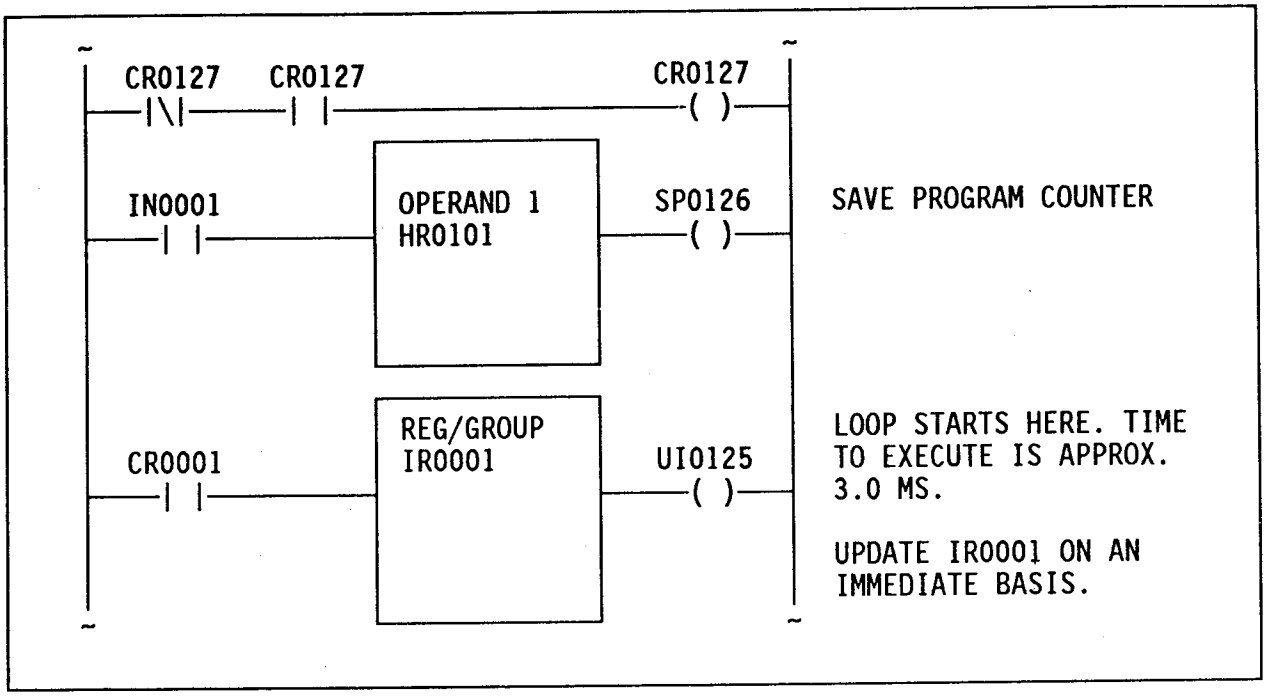


Figure 3a. RW Application

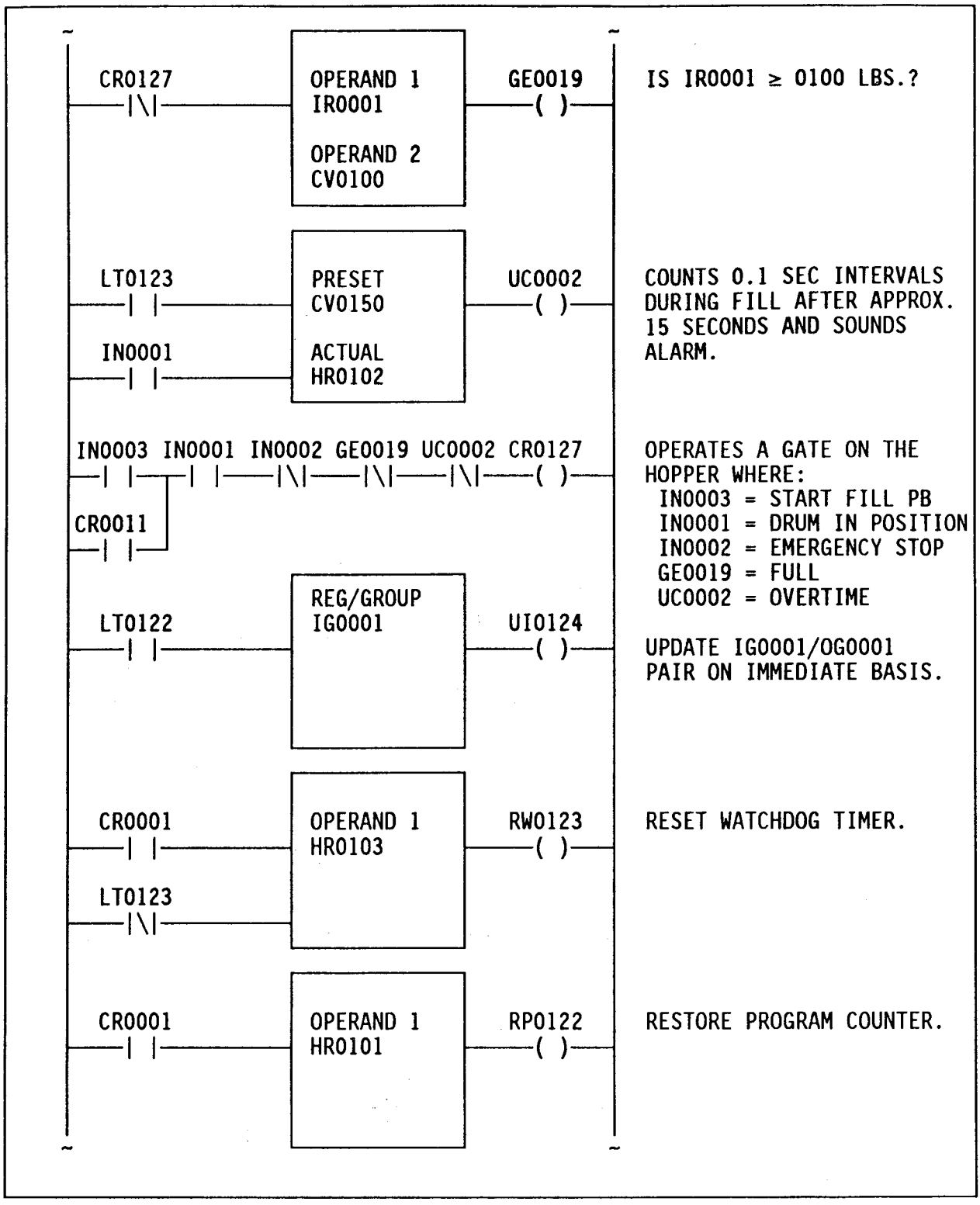


Figure 3b. RW Application

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Skip (SK) function is a powerful programming tool. When the SK function is executed, the processing of the entire ladder diagram or a portion thereof is skipped (bypassed). The skipped functions remain frozen in the state existing prior to the execution of SK. SK function symbology is shown in Figure 1.

SK allows a prescribed condition or set of conditions to determine when, and if, all or part of the circuits programmed into the processor are skipped. The SK coil energizes when the SK contact circuit is conducting and all coils under SK control are skipped: they are left in the state they were in prior to energizing the SK coil. When the circuit is not conducting, all coils under SK control operate normally.

The SK function depends upon the condition of its contact circuit. Forcing the SK coil forces only its contact and output circuits. Forcing the SK function requires the forcing of the coils or inputs that control the contacts in the SK contact circuit.

The coils controlled by the SK function are specified by a preset constant (1 through 256). This value indicates the number of coils following the SK coil. The controlled coils are those programmed immediately after the SK coil. If the preset value is 5, the next five coils programmed are controlled by the SK contact circuit. Special functions in the range of SK are not executed when SK is in effect.

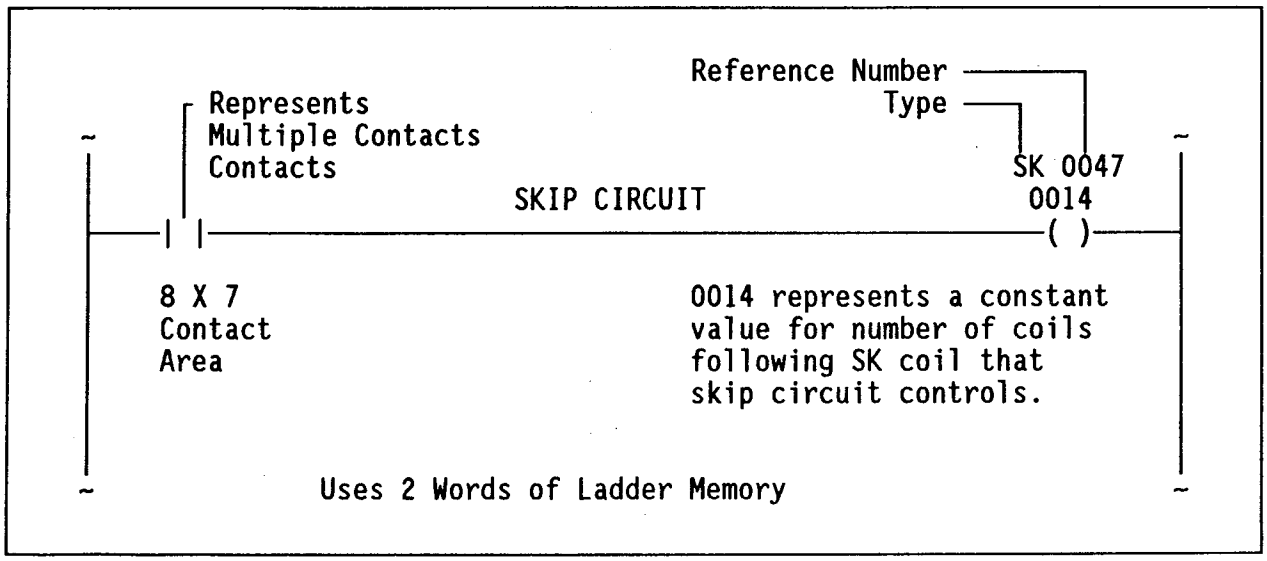


Figure 1. Skip (SK)

SPECIFICATIONS

SK CIRCUIT

When the SK circuit conducts, the specified number of coils following the coil are skipped. When the SK circuit is not conducting, normal processing is allowed.

NUMBER OF COILS

The number of coils specifies the number of coils to be skipped following the SK function (1 through 256). If the end of the program is before the number of coils specified, the range is terminated by the end of the program.

In the PC-1100, the portion of the program that is skipped is scanned by the processor but ignored. In the PC-1200, once the end of the skip is determined (and if no on-line changes are being made), the processor jumps over the skipped program section, reducing scan time.

Note that since timers in the PC-1200 accumulate elapsed time (up to approximately 16 seconds before rolling over to zero) and update the user's register when executed, they can continue to accumulate time even if skipped.

COIL

The coil energizes when the SK circuit conducts, and de-energizes when the circuit does not conduct.

APPLICATIONS

The SK function can be used when a series of outputs must be updated at predefined intervals (i.e., every 10 seconds). Figure 2 is an example of a program that controls this update by using the SK function.

In the PC-1200, the average scan time of a ladder can be greatly reduced by using the SK function to periodically skip non-critical sections of the ladder.

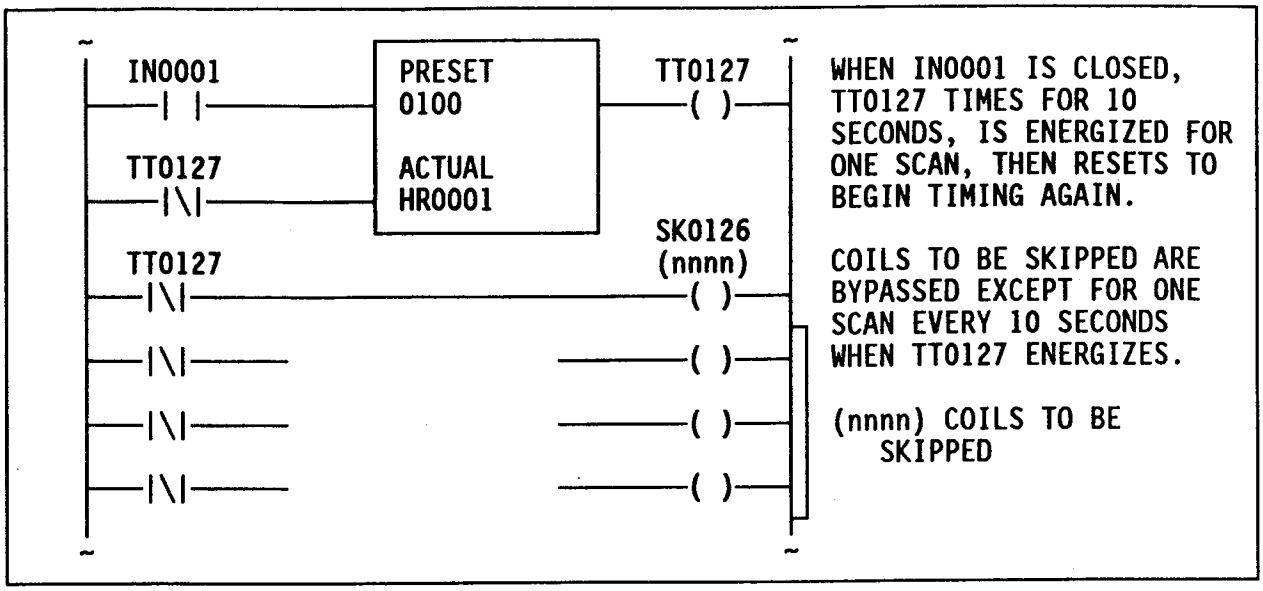


Figure 2. SK Application

SM - SEARCH MATRIX

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Search Matrix (SM) function is used to search bits that are 1's in a matrix and to report the bit position where the 1 is located. A matrix is defined as a table of registers handled on a bit-by-bit basis. Each time the SM function is operated, a bit register is used to hold the bit number of the bit that is a 1. SM function symbology is shown in Figure 1.

A pointer (bit register) holds the bit number of the bit found by the last search. On each open-to-closed transition of the step input (with the reset input conducting), the matrix is searched from the position following the pointer to the next set bit in the matrix. When a set bit is found, the coil is turned on, and the position of the located set bit is stored in the bit register. If no set bit is located between the pointer position and the end of the matrix, then the bit register is set to zero and the coil is turned off.

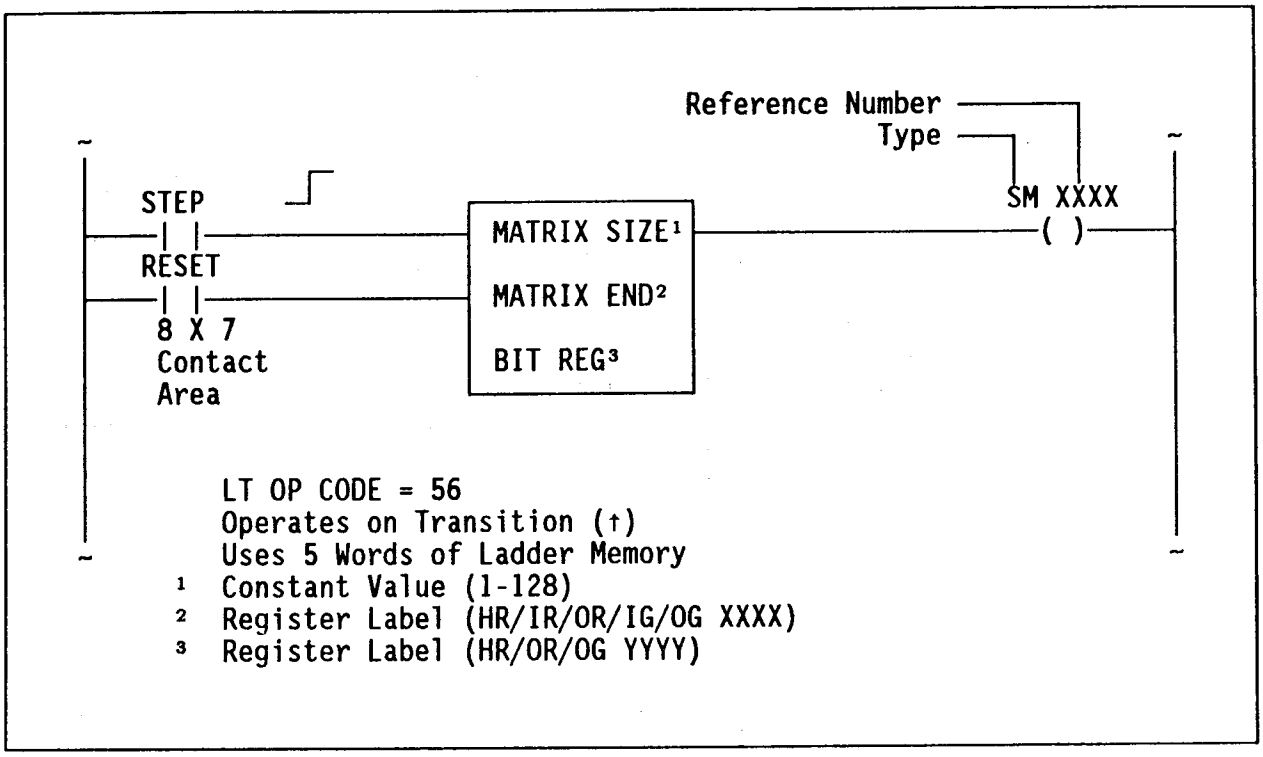


Figure 1. Search Matrix (SM)

For example, consider the matrix shown in Figure 2. A series of operations would yield the following values:

Operation No.	(Bit Register Value) Bit Number
1	0005
2	0007
3	0017
4	0022
5	0027
6	0028
7	0031
8	0032
9	0033
10	0036
11	0039
12	0000
13	0005
14	0007
•	•
•	•

In this example, the end of the matrix is encountered at operation number 12. At this point, the bit register value is set to zero and the coil is turned off.

MATRIX

HR0001	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
HR0002	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
	1	1	0	0	1	1	0	0	0	0	1	0	0	0	0	1
HR0003	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1

Figure 2. Matrix to be Searched

The matrix being examined is not affected by the SM function.

The SM function uses two input circuits, reset and step. When reset is open (non-conducting), the function is not operative and the bit register is set to zero. When reset is closed (conducting), the function operates on the transition of the step circuit from open (non-conducting) to closed (conducting). The coil energizes when:

1. Reset is closed (conducting).
2. The bit register is pointing to a non-zero bit.
3. The step input is closed (conducting).

OP CODE

Op Code 56 defines the Literal (LT) function as an SM function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - MATRIX SIZE

The matrix size is a constant value that defines the number of registers included in the matrix. The range is 1 through 128 and is limited as defined by the matrix end.

OPERAND 2 - MATRIX END

The matrix end defines the type and reference number of the last register in the matrix, as shown in Table 1.

Note

The highest Holding Register reference number acceptable is dependent upon the memory and user program size.

TABLE 1. SM END REGISTER

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
IG	≤ 4	≤ 4	≤ 8	≤ 16
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

OPERAND 3 - BIT REGISTER

When reset is conducting, the bit register points to the non-zero bits in the matrix. It advances one-at-a-time whenever the step circuit changes from OFF to ON. When reset is non-conducting, the bit register is zeroed. This register may be a specified:

- Holding Register (HR)
- Output Register (OR)
- Output Group (OG)

COIL

The coil is energized when the reset and step inputs are conducting and the bit register is pointing to a non-zero bit.

SM TRUTH TABLE

See Table 2.

TABLE 2. SM TRUTH TABLE

Step	Reset	Result
X	0	The coil is de-energized; the bit register equals zero.
0	1	The coil is de-energized; the bit register depends on previous operations.
↑	1	The bit register advances to the next non-zero bit and the coil is turned on. (The bit register is set to zero and the coil is turned off no set bit is found before the end of the matrix).
1	1	The coil and register are left in the state caused by the transition of step.
X = Don't care		

APPLICATIONS

The SM function is used as an internal monitoring program. As shown in Figure 3, if the actual state is equal to the desired state, no action is taken. If the actual state does not equal the desired state, the SM function points to the offending bit.

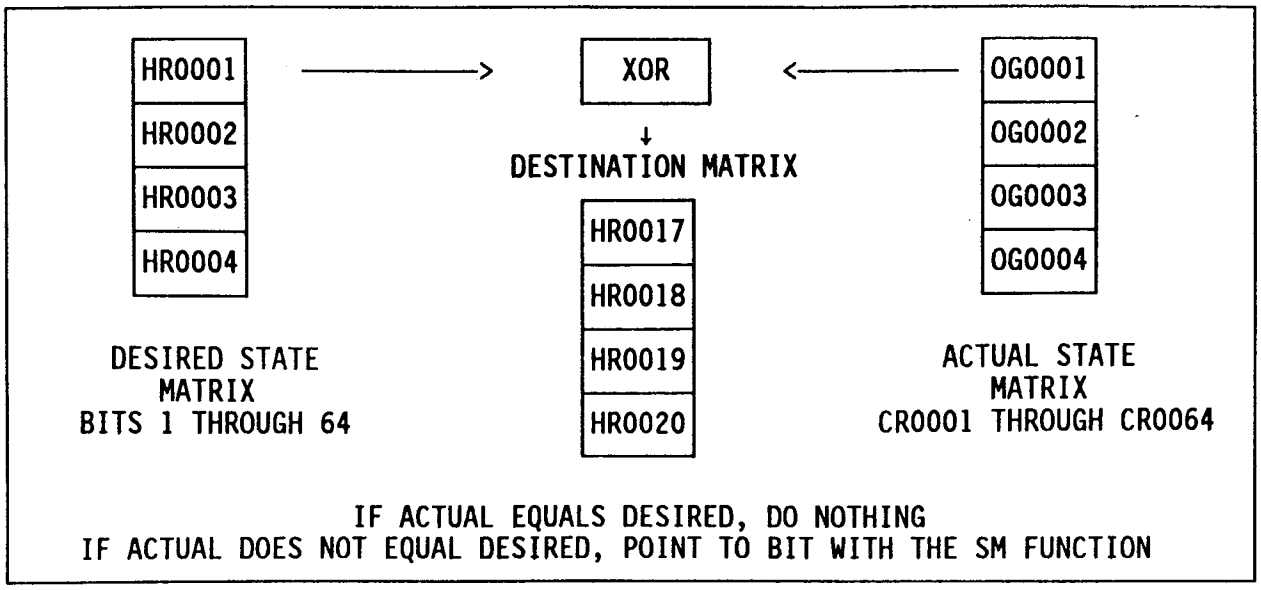


Figure 3. SM Function Equal-State Case

The ladder diagram for the SM function is shown in Figure 4. A non-zero result from the searched function enables IN0002 and causes the bit register to hold the number of each successive non-zero bit. IN0003 resets the function.

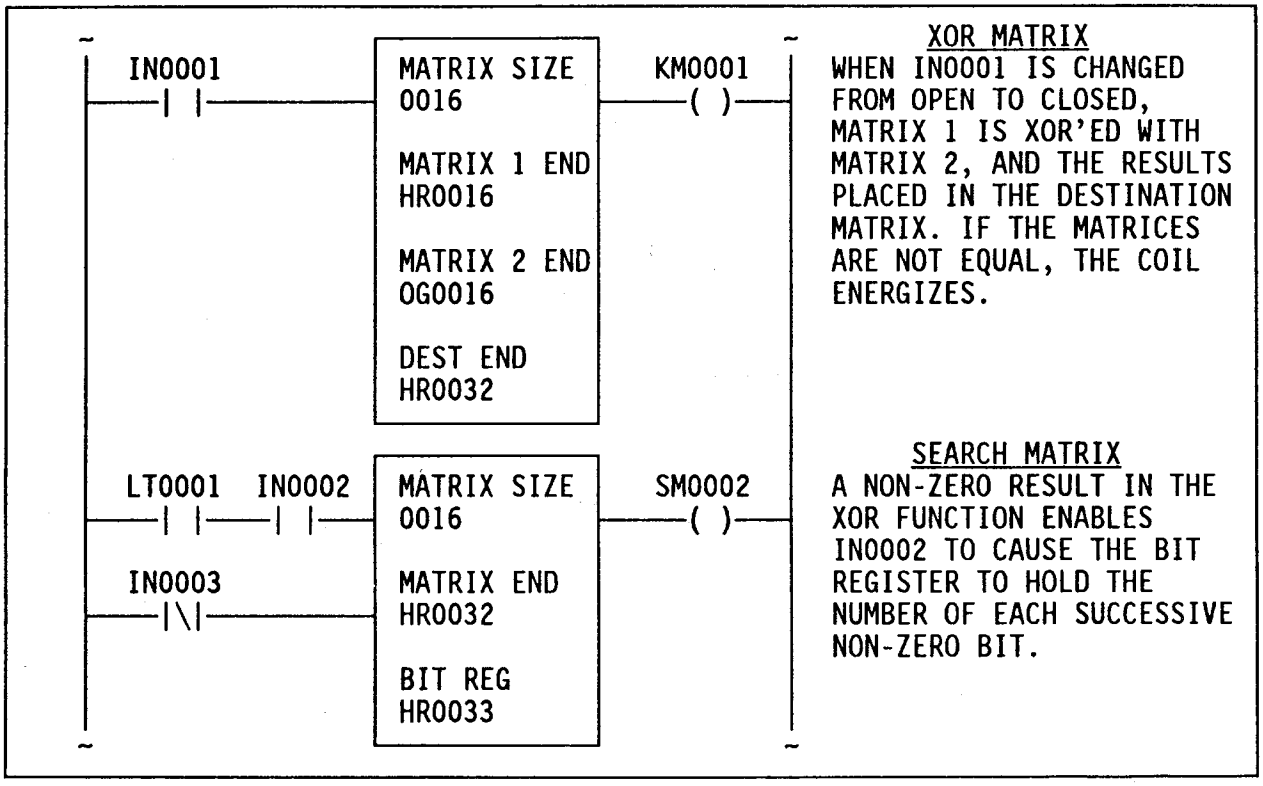


Figure 4. SM Application

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Save Program Counter (SP) function is one of three Loop Back functions. The Loop Back functions are:

- Save Program Counter (SP)
- Restore Program Counter (RP)
- Reset Watchdog Timer (RW)

Loop Back functions are used in the development of programs that are capable of repeating segments of the ladder diagram.

SP function symbology is shown in Figure 1.

OP CODE

Op Code 12 defines the Literal (LT) as the SP function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1

Operand 1 defines the holding register location to be used for storage of the program counter.

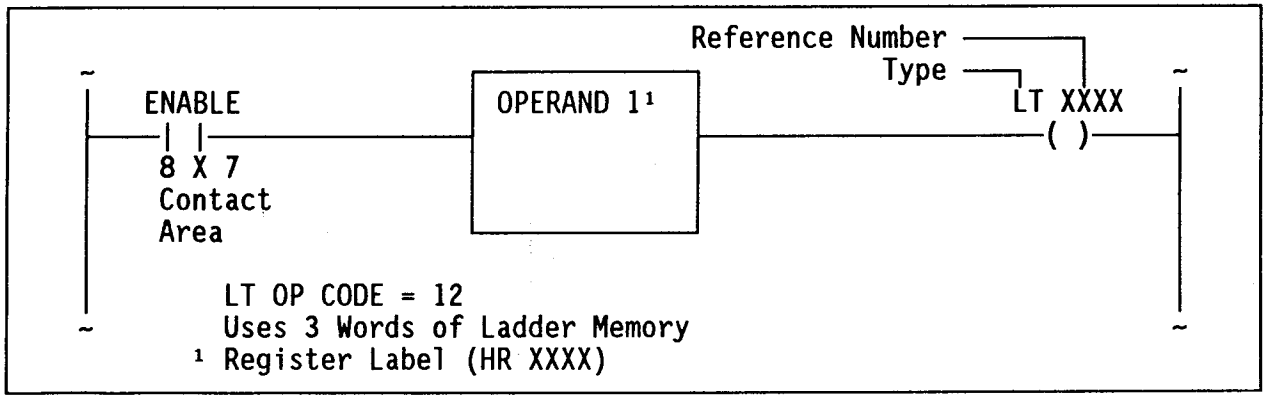


Figure 1. Save Program Counter (SP)

SP TRUTH TABLE

See Table 1.

TABLE 1. SP TRUTH TABLE

Enable	Result
0	The coil is de-energized. The ladder diagram execution is unaffected
1	The coil is energized. The contents of the program counter are duplicated in the register specified by Operand 1. When the program counter is restored, ladder execution begins with the first rung following the SP coil.

APPLICATIONS

See the "Applications" section of the RP function.

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

In the PC-1100, the Square Root (SQ) function allows the extraction of an eight-digit decimal number, up to 99,999,999. In the PC-1200, the Square Root (SQ) function allows the extraction of a nine-digit decimal number, up to 655,415,535. SQ function symbology is shown in Figure 1.

The number from which the root is taken is the source register. This register consists of a pair of registers: the first register contains the most significant digits; the second register contains the least significant digits. The four-digit result is placed in the destination register, a single location. The source register label denotes the first register of the pair. If the source register is HR0001, the pair is HR0001/HR0002. In the example shown in Figure 2, the source register is HR0001 and the destination register is HR0003.

The calculation is made when the calculate circuit changes from non-conducting to conducting. Although the source and destination are defined as decimal numbers, the actual square root is calculated in binary numbers because the root must be stored as a binary number. If the number is originally in Binary-Coded-Decimal (BCD) form, it is converted to binary by using the Decimal to Binary (DB) function. Likewise, if the result is desired in BCD form, the Binary to Decimal (BD) function is used.

In the PC-1200, if the number in the least significant register (that is, the second register in the source pair) is greater than 9999, the high order digit (number of 10,000's in the value) is in effect added to the high order register before the square root is extracted. For example, if HR0001 = 00010 and HR0002 = 10000, then HR0003 = the square root of 110,000. Any fractional part of the result is truncated.

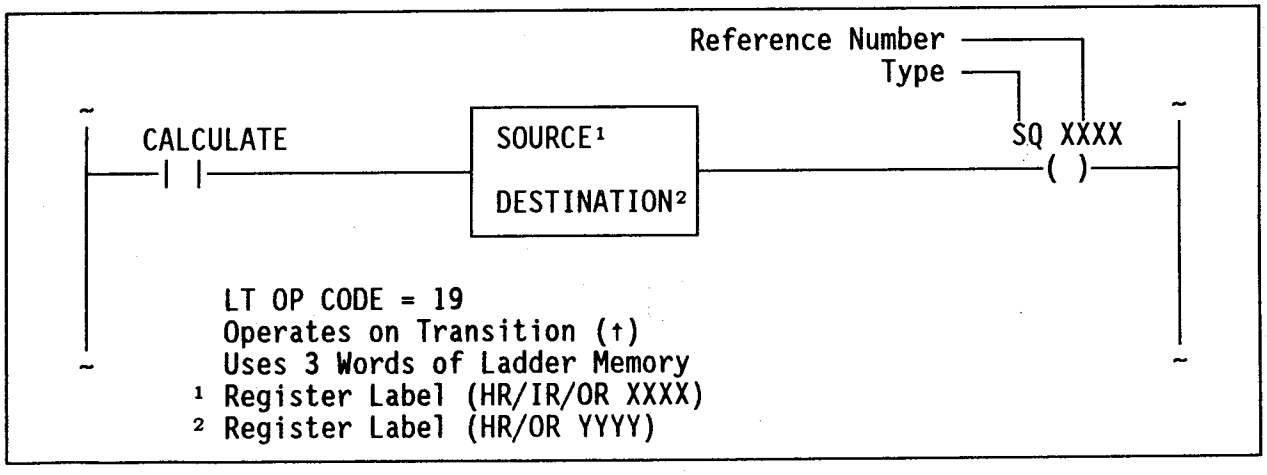


Figure 1. Square Root (SQ)

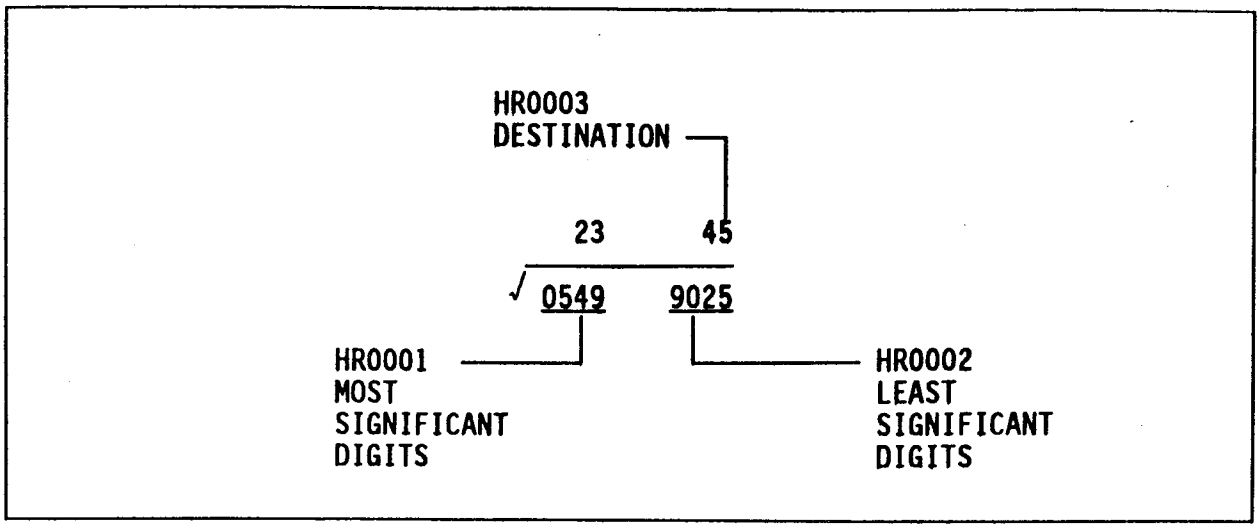


Figure 2. Source Register

The SQ coil energizes when the calculate circuit is conducting and de-energizes when the calculate circuit is non-conducting.

OP CODE

Op Code 19 defines the Literal (LT) as the SQ function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - SOURCE

The source designates the first register of a pair of registers that holds the number (0 through 99,999,999) from which the square root is taken. The first register in the pair holds the most-significant digits; the second register holds the least-significant digits. This pair is a set of specified registers:

- Holding Registers (HR)
- Input Registers (IR)
- Output Registers (OR)

OPERAND 2 - DESTINATION

The destination designates the location of the results from the SQ function calculation. This result is held in a specified register:

- Holding Register (HR)
- Output Register (OR)

SQ TRUTH TABLE

See Table 1.

TABLE 1. SQ TRUTH TABLE

Calculate	Result
0	The coil is de-energized. No calculations take place.
↑	The coil follows the calculate circuit. The square root is extracted from the source pair and the results are placed in the destination.
1	The coil is energized. No further calculations take place.

APPLICATIONS

The analog outputs of a kilowatt/kilovolt-ampere (kw/kVAR) reactive transducer are converted to digital data by analog input modules. Square roots are required to calculate kVA to further derive the Power Factor (PF) from these inputs. Figure 3 depicts the calculations of kVA and PF. Figure 4 is the ladder diagram using the SQ function.

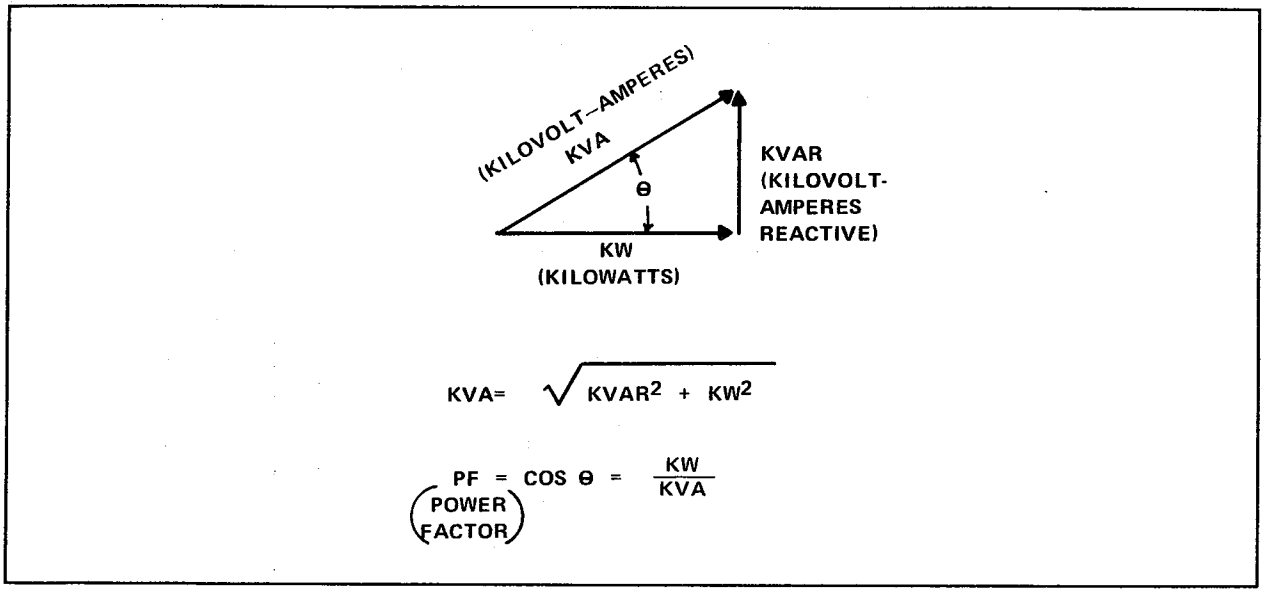


Figure 3. SQ Calculation

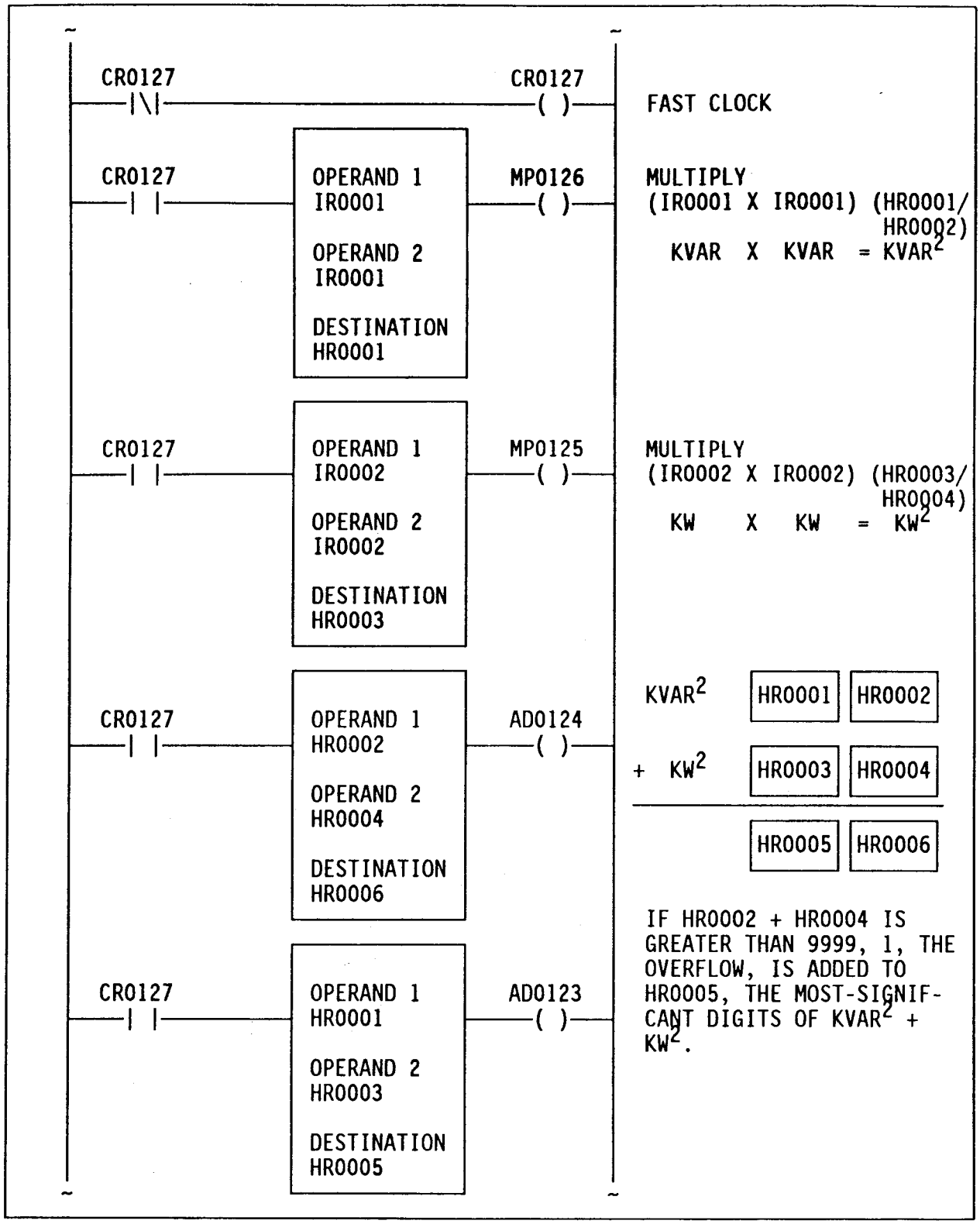


Figure 4a. SQ Application

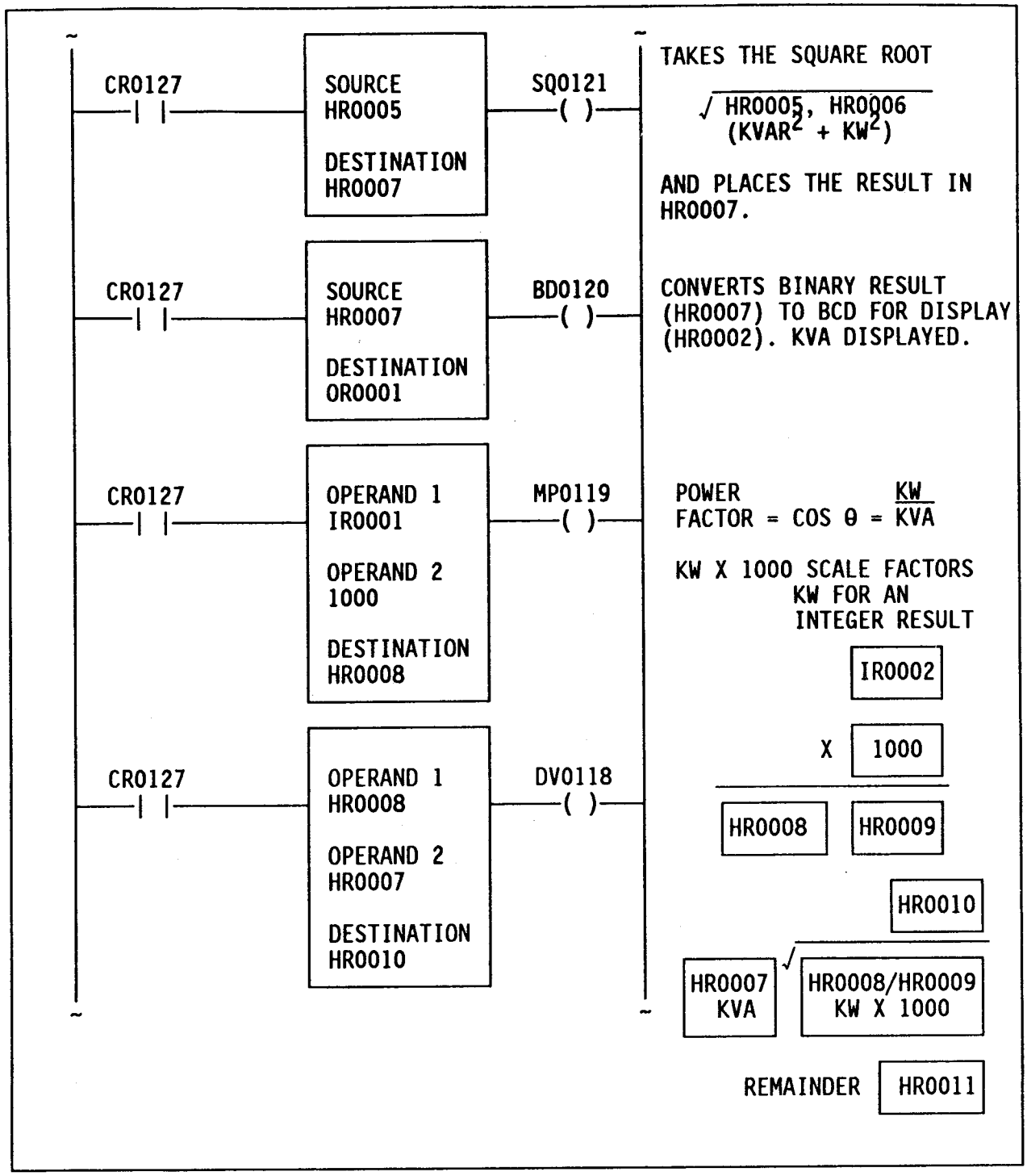


Figure 4b. SQ Application

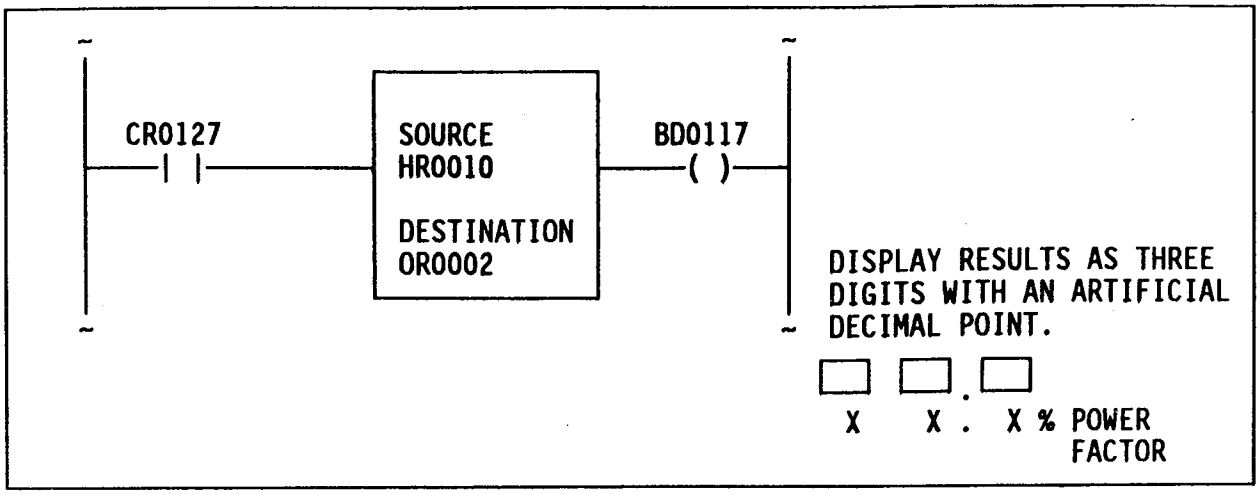


Figure 4c. SQ Application

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Table Lookup (TL) and Table Lookup Ordered (TO) functions are used to search a table of registers and locate the position of the register whose contents are greater than or equal to the source register contents. The TL function is used to find the registers whose contents are equal to the source destination, and the TO function is used to find the registers whose contents are greater than or equal to the source destination. TL/TO function symbology is shown in Figure 1.

OP CODE

Op Code 82 defines the Literal (LT) as TL. Op Code 83 defines the Literal (LT) as TO. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

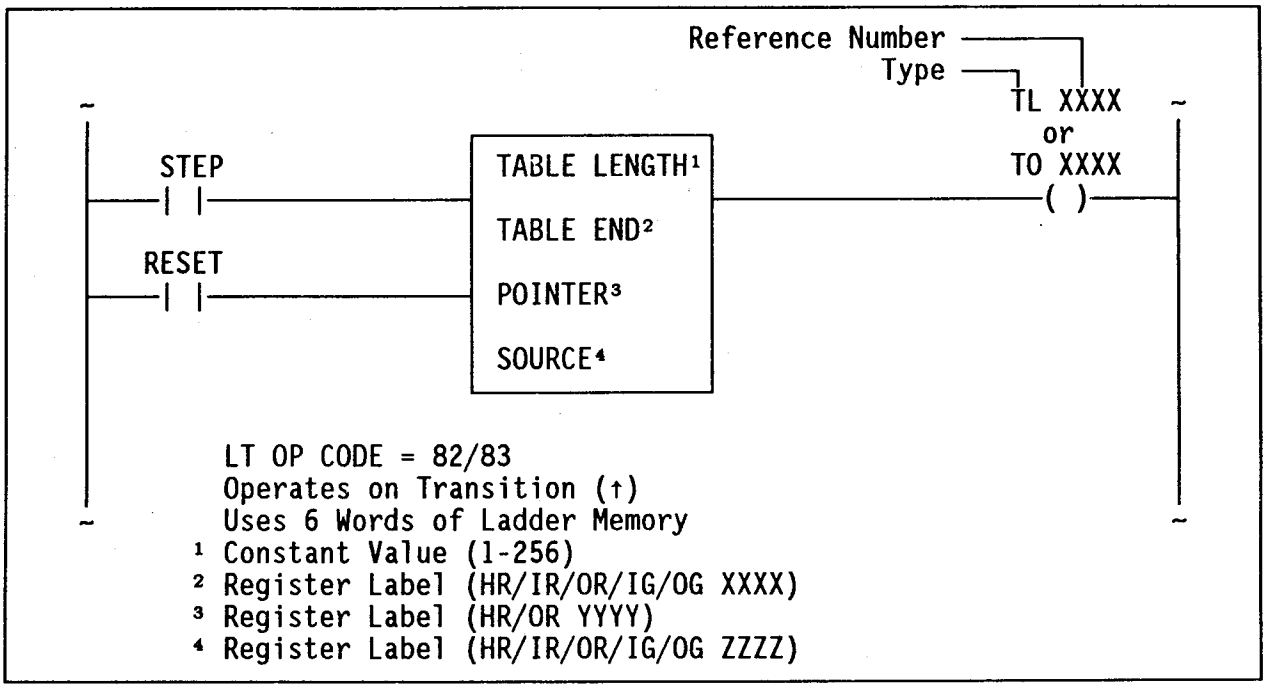


Figure 1. Table Lookup (TL)/Table Lookup Ordered (TO)

SPECIFICATIONS

OPERAND 1 - TABLE LENGTH

The table length is a constant value in the range of 1 through 256 that determines the length of the table being examined.

OPERAND 2 - TABLE END

The table end defines the number of the last holding register. This may be a specified:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

OPERAND 3 - POINTER

The pointer contains the current table location being examined. This may be a specified:

- Holding Register (HR)
- Output Register (OR)
- Output Group (OG)

OPERAND 4 - SOURCE

The source is the location of the data to which the table will be compared. This may be a specified:

- Holding Register (HR)
- Input Register (IR)
- Output Register (OR)
- Input Group (IG)
- Output Group (OG)

TL/TO TRUTH TABLE

See Table 1.

APPLICATIONS

The TO function can be used to find an angle between 0° through 89° when given the sine. A table is constructed that assumes a decimal point to the left and contains the sines of the angles from 0° through 89° in as many as 256 increments. Table 2 illustrates this concept using 90 increments.

TABLE 1. TL/TO TRUTH TABLE

Step	Reset	Result
X	0	The pointer is zeroed and the coil is de-energized.
0	1	The pointer remains in the previous position. The coil is de-energized.
↑	1	The search starts from the current pointer location plus one for the next location in the table that satisfies the function. Table = TL Function / Source Register Table ≥ TO Function / Source Register The new position is in the pointer location and the coil is energized. If no register is found satisfying the function, the pointer is set to zero, and the coil is de-energized.
1	1	The pointer remains in the same state as at the time of transition. The coil is energized.
X = Don't care		

TABLE 2. SINE VALUES AND CORRESPONDING ANGLES

Angle°	Sine
0	0000
1	0175
2	0349
3	0523
4	0698
•	•
•	•
89	9994
90	9998

If the sine value is placed in the source register and the instruction is executed, the pointer value is the approximate angle given by the sine. See Figure 2.

As shown in Figure 2, TO0125 is a TO function. For IN0001, the sine entered in IR0001 is converted to binary and input as the source for the TO function. The pointer value is the angle corresponding to the sine in IR0001.

If 0872 is placed in IR0001, and IN0001 is closed, the pointer goes to 5 to indicate that IR0001 is the sine of an angle of 5°. This is an approximation. A value of IR0001 between 872 and 1044 indicates an angle of 5°. A value equal to or greater than 1045 indicates an angle greater than 5°. A value less than or equal to 0871 indicates an angle less than 5°.

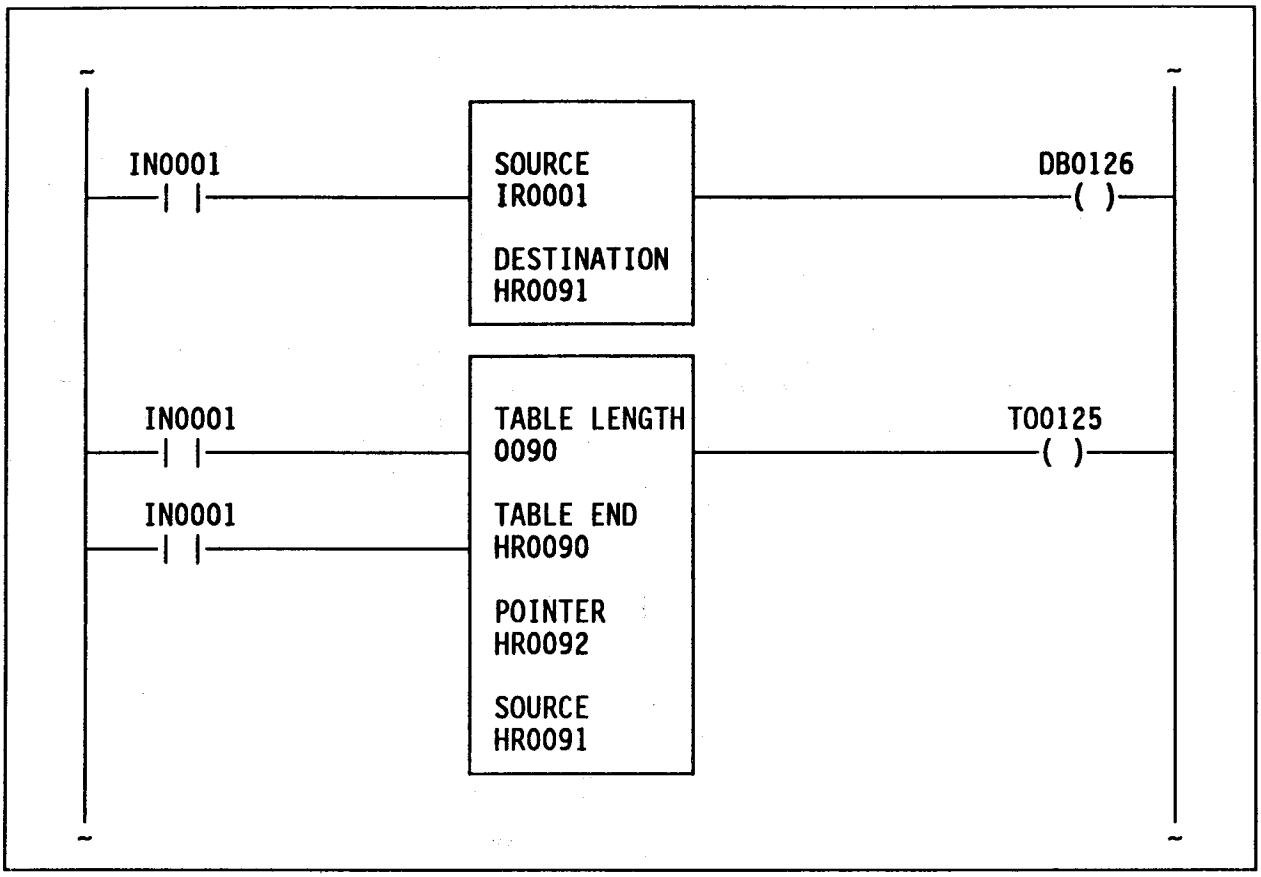


Figure 2. TO Application

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Table-to-Register Move (TR) function performs a drum sequencer task by allowing a table of registers to be transferred to a destination location, one register at a time. The table may contain from 1 to 256 registers. Figure 1 illustrates the concepts and symbols related to the TR function.

The TR function consists of three input circuits (step, reset, enable), a table of registers defined by table length and table end, a pointer location, and a destination location. With each low-to-high transition of the step circuit (assuming the reset and enable circuits are conducting), the pointer is incremented and the next register from the table will be transferred to the destination register. In the PC-1200, the pointer can be stepped without the enable line conducting. The register to be transferred is determined by the value contained in the pointer location. (See Figure 2.) With each transition on the step circuit, the pointer value increases by one, thereby allowing subsequent registers in the table to be transferred to the destination location. When the pointer value plus one equals the specified table length, the next non-conducting-to-conducting change in the step circuit will reset the pointer value to 0. Also, if the reset circuit is not conducting, the pointer value is zeroed.

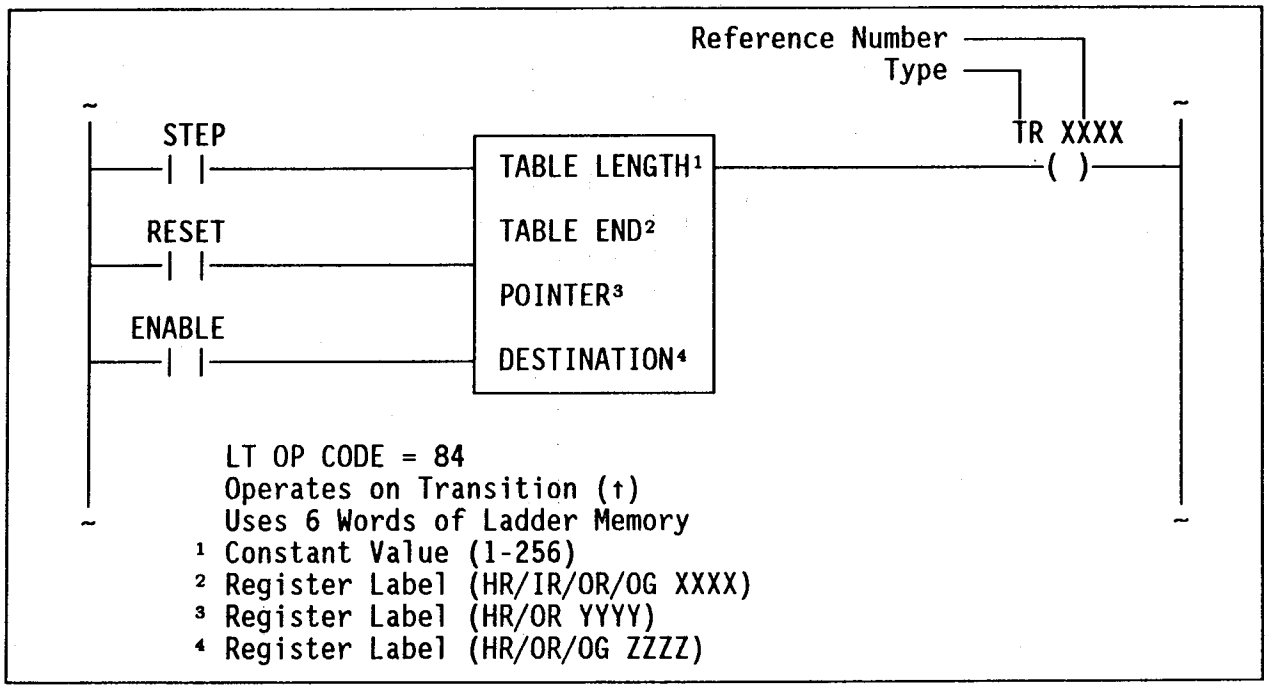


Figure 1. Table-to-Register Move Concept

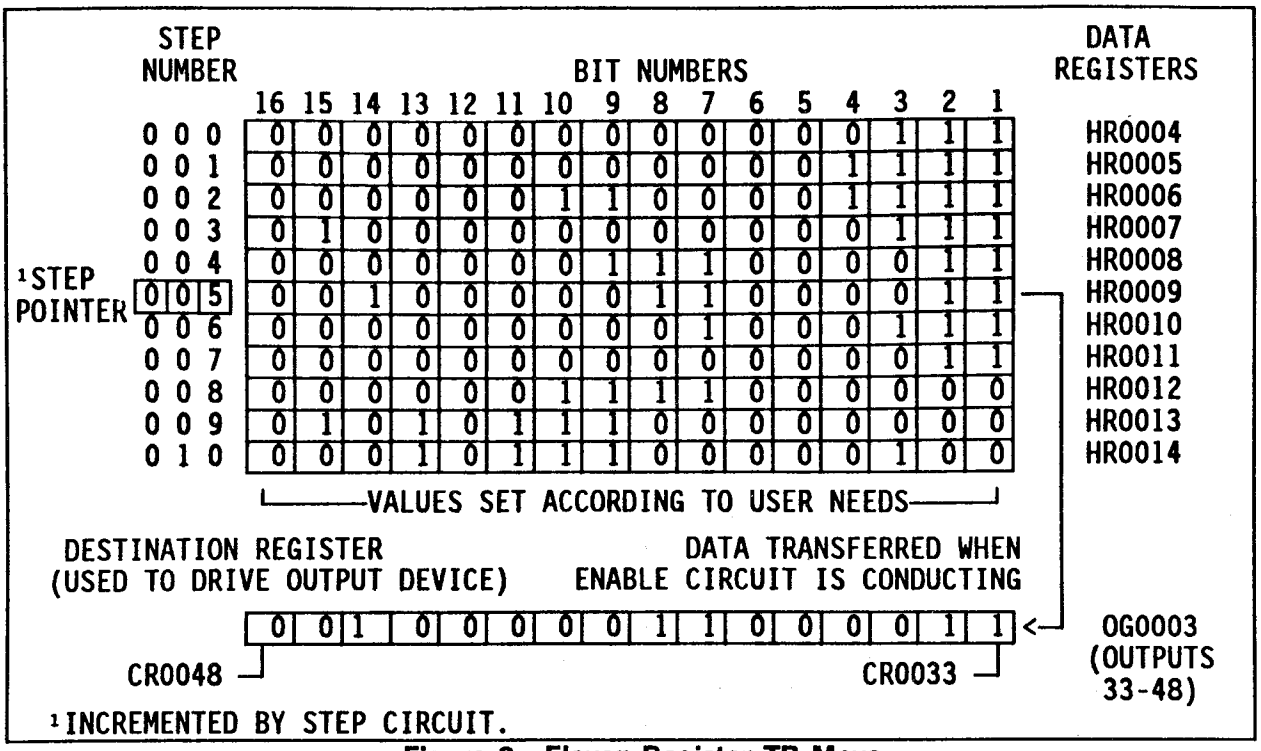


Figure 2. Eleven-Register TR Move

The enable circuit must be conducting for the function to move data from the table to the destination. If the enable circuit is not conducting, the destination will stay in the state it was in when the enable circuit changed from conducting to non-conducting.

The coil energizes only when the pointer value equals table length minus 1. Forcing a TR coil affects only the contacts associated with the coil; the TR function continues to operate normally.

OP CODE

Op Code 84 defines the Literal (LT) as being a Table-to-Register Move (TR). Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - TABLE LENGTH

Defines the number of registers in the table to be transferred. Range is from 1 to 256 and is limited-as indicated under table end.

OPERAND 2 - TABLE END

Defines the type and number of the last register in the table to be transferred in accordance with Table 1.

TABLE 1. LAST REGISTER TYPE, NUMBER

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

OPERAND 3 - POINTER

Register holding the number of the current register in the table that is being transferred to the destination. This may be a:

- Holding Register
- Output Register

OPERAND 4 - DESTINATION

The register into which information pointed by the pointer is placed. This may be a:

- Holding Register
- Output Register
- Output Group

TR TRUTH TABLE

See Table 2.

TABLE 2. TR TRUTH TABLE

Step	Reset	Enable	Coil Action/Results
0, 1 or ↑	0	0	Pointer is set to zero and held as long as reset is false. No data is moved.
↑	1	0	In PC-1100, pointer frozen at current value. No data moved. In PC-1200, no data moved, pointer increments.
0 or 1	0	1	Pointer set to zero. Contents of table start moved to destination
↑	1	1	Pointer incremented, data is transfered. If pointer is = (Table Length minus 1), the coil is energized. If pointer is ≥ table length, pointer is zeroed, and the data transferred.
0 or 1	1	1	Data transferred. Note: data is continuously moved both prior to and after the step function
↑	1	↑	PC-1100: Pointer does not increment, data moved. PC-1200: Pointer increments, data moved.

Note that the PC-1200 uses an independent counter (the step and reset inputs control the count in the pointer), and move is controlled by the enable input.

APPLICATIONS

In a simple application, the Table-to-Register Move function drives up to 16 discrete outputs through as many as 256 steps. Consider the program below in Figure 3.

A TR function does not have to use the step circuit to increment the pointer. Other functions may be used to alter the pointer value as shown in Figure 4.

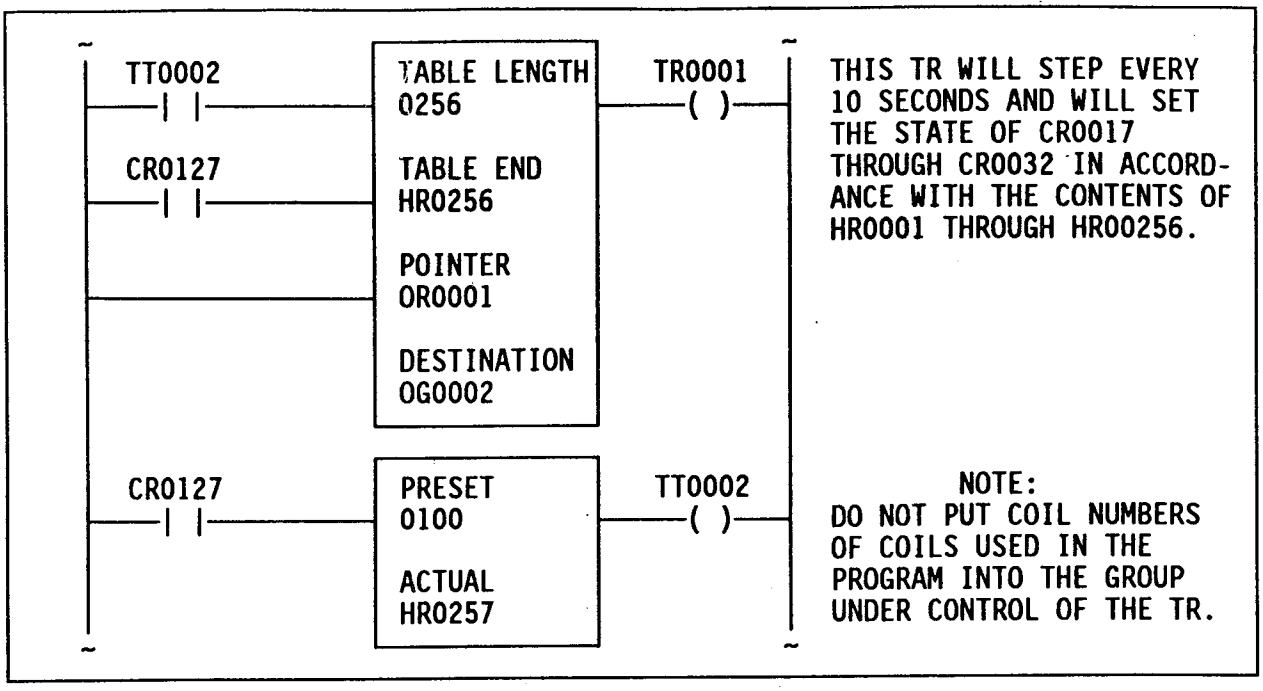


Figure 3. Example 256-Step Move

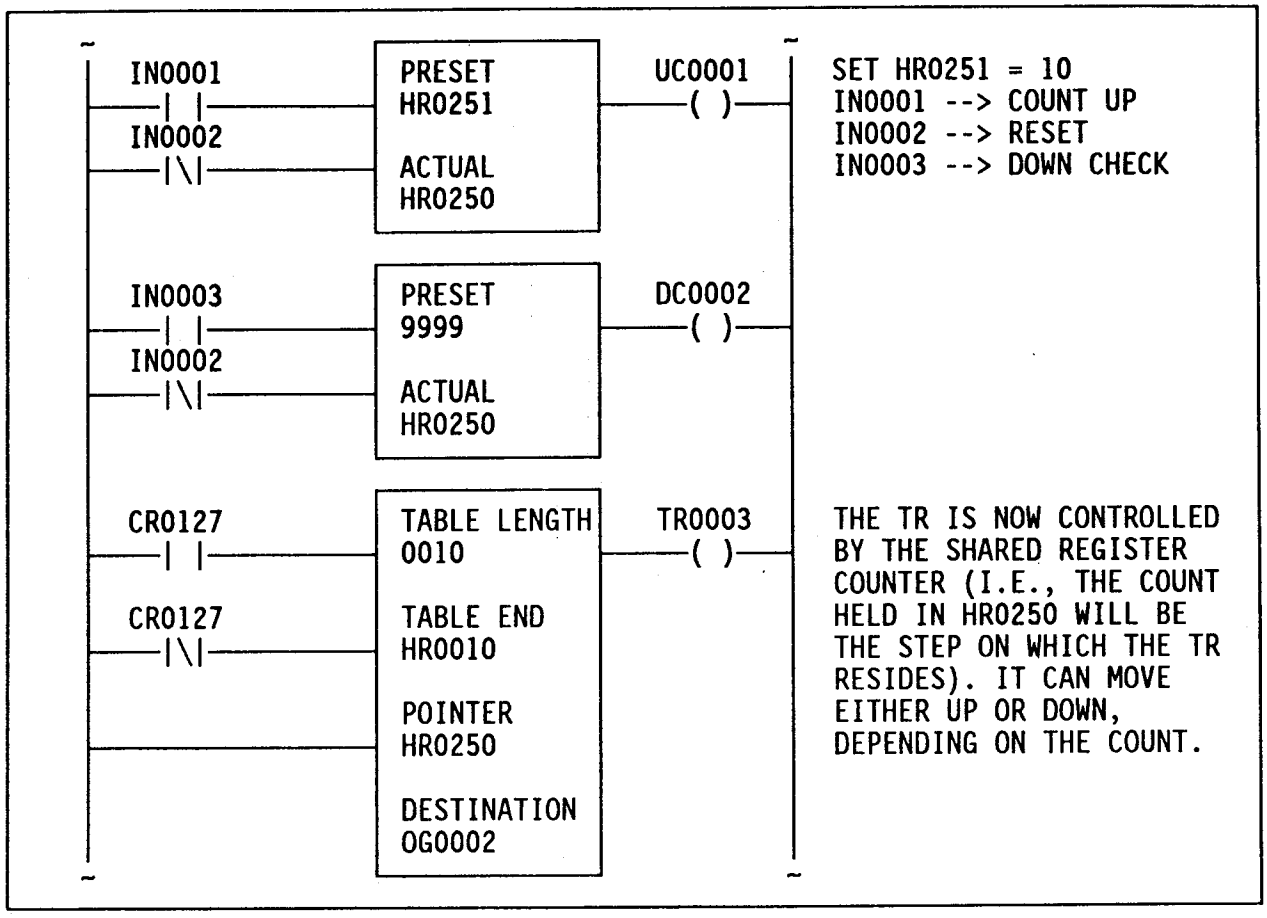


Figure 4. TR Application

In some situations it may be required to have a TR function to automatically operate at different intervals between steps. This can be accomplished using two TR functions, as shown in Figure 5.

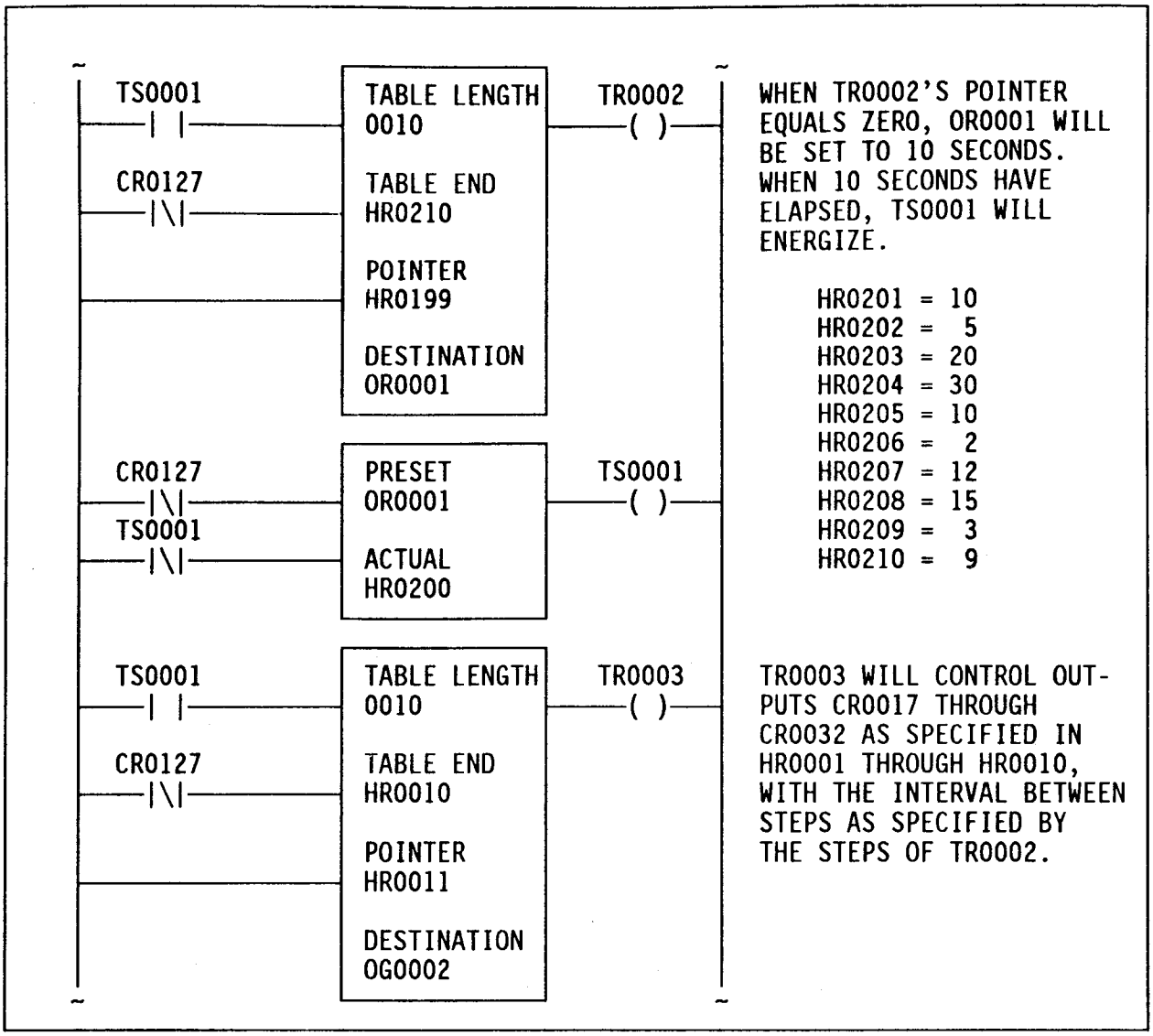


Figure 5. Application Using Two TR Functions

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

Timers are programmed to operate like any of the common types of electro-mechanical timers (pneumatic ON-and-OFF delay, motor driven timers, etc.). The timer is used, either independently or in conjunction with other functions, to develop complex timing chains. These timing chains have duty cycles structured to meet most machine cycle applications. Timer function symbology is shown in Figure 1.

The timing circuit and the enable circuit control the timers, which run only when the enable and timing circuits are conducting. If the timing circuit stops conducting, the timer retains the accumulated value as long as the enable circuit is conducting. When the enable circuit is not conducting, the timer is reset and held at 0000. Accumulated time is stored in the actual register (an assigned holding or output register). The associated coil is energized and its contacts are operated when the actual value equals a preset value, which is programmed as a constant along with the timer or comes from a specified register (holding, input, or output register) simultaneously with the enable circuit conducting.

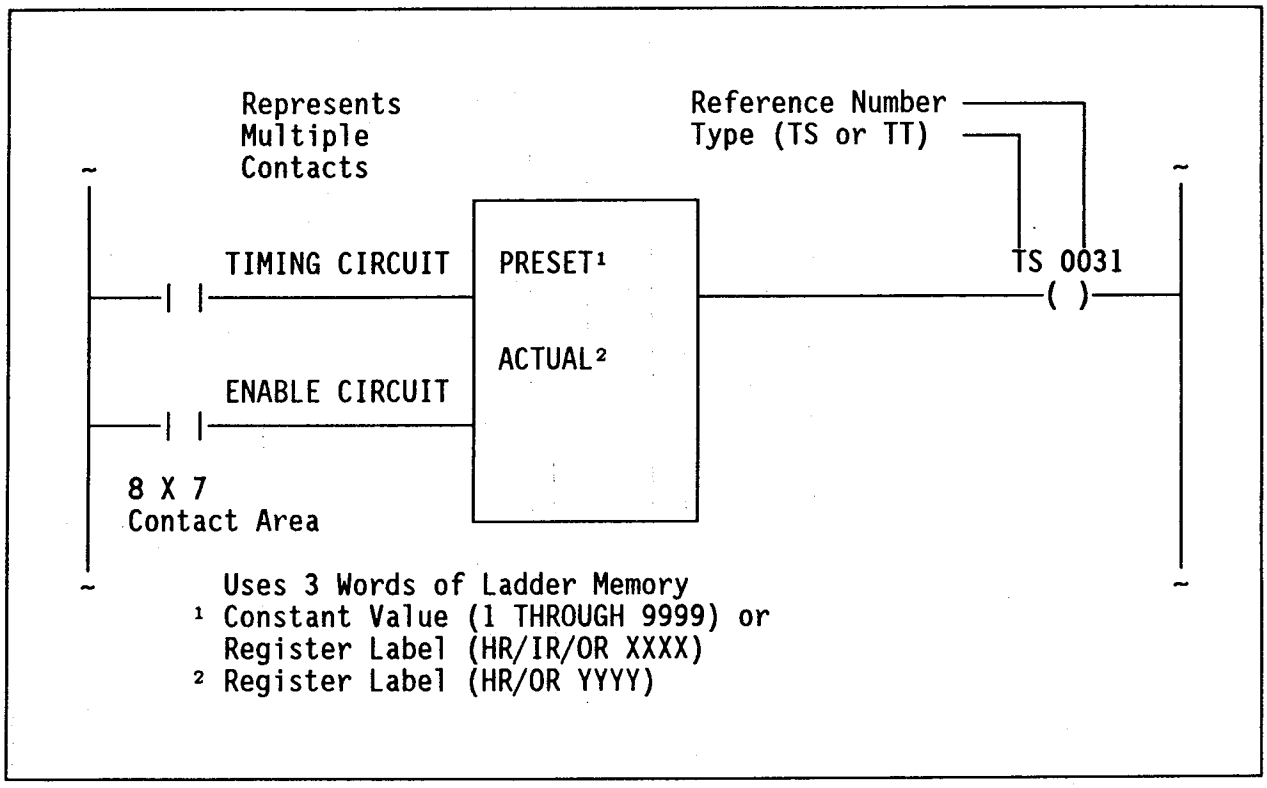


Figure 1. Timers (TS/TT)

In the PC-1100, the processor maintains a 1.0 second and a 0.1 second time base for all timers. All TS or TT functions are updated on the same scan, and the RW special function clears the time base.

In the PC-1200, each timer maintains its own high-resolution time base capable of accumulating up to approximately 16 seconds. The timers maintain accuracy within this 16-second time period when the SK function, RW function, or SP/RP functions are used.

ON DELAY CIRCUIT

An example of an ON Delay timer circuit is shown in Figure 2. The same input device (IN0001) controls both the timing and the enable circuits. The preset value is a constant value of 5 seconds, and the actual value is accumulated in HR0001. At Time 0, IN0001 is closed, both enabling the timer and starting the timing circuits. If left in this condition, TS0014 energizes after 5 seconds, as shown at Time 5.

Since this is an ON Delay, removal of the input causes the timer to immediately reset, turning OFF TS0014. If the input is again closed, as at Time 10, timing resumes. At Time 13, timing is interrupted. The ON delay is reset and a full delay results when next activated at Time 15.

OFF DELAY CIRCUIT

Figure 3 is an example of an OFF Delay timer circuit. In this circuit, IN0002 controls the timer. IN0002 is a normally-open (NO) switch. Its contacts in the timer control circuits are programmed to be normally-closed (NC). An additional ladder rung (CR0010) is required to implement the OFF Delay. Assuming that CR0010 is initially OFF and that IN0002 is not activated for some time, the TS0015 coil is energized, holding CR0010 OFF with the now open TS0015NC contacts. At Time 0, IN0002 is closed, resulting in the timer being reset, TS0015 being de-energized, and CR0010 being energized. The CR0010 coil is energized through the IN0002 contacts and the NC contacts of TS0015. At Time 5, IN0002 turns OFF, causing the timer to activate. The CR0010 coil remains energized via the CR0010 and TS0015 contacts until the timer times out. At Time 10 (timer time-out) the TS0015 coil energizes, opening its NC contacts and turning CR0010 OFF. CR0010 is de-energized after the OFF delay.

The remainder of the timing diagrams shows that the OFF delay is not cumulative (i.e., delay time must start over if the device is turned ON again before the OFF Delay elapses).

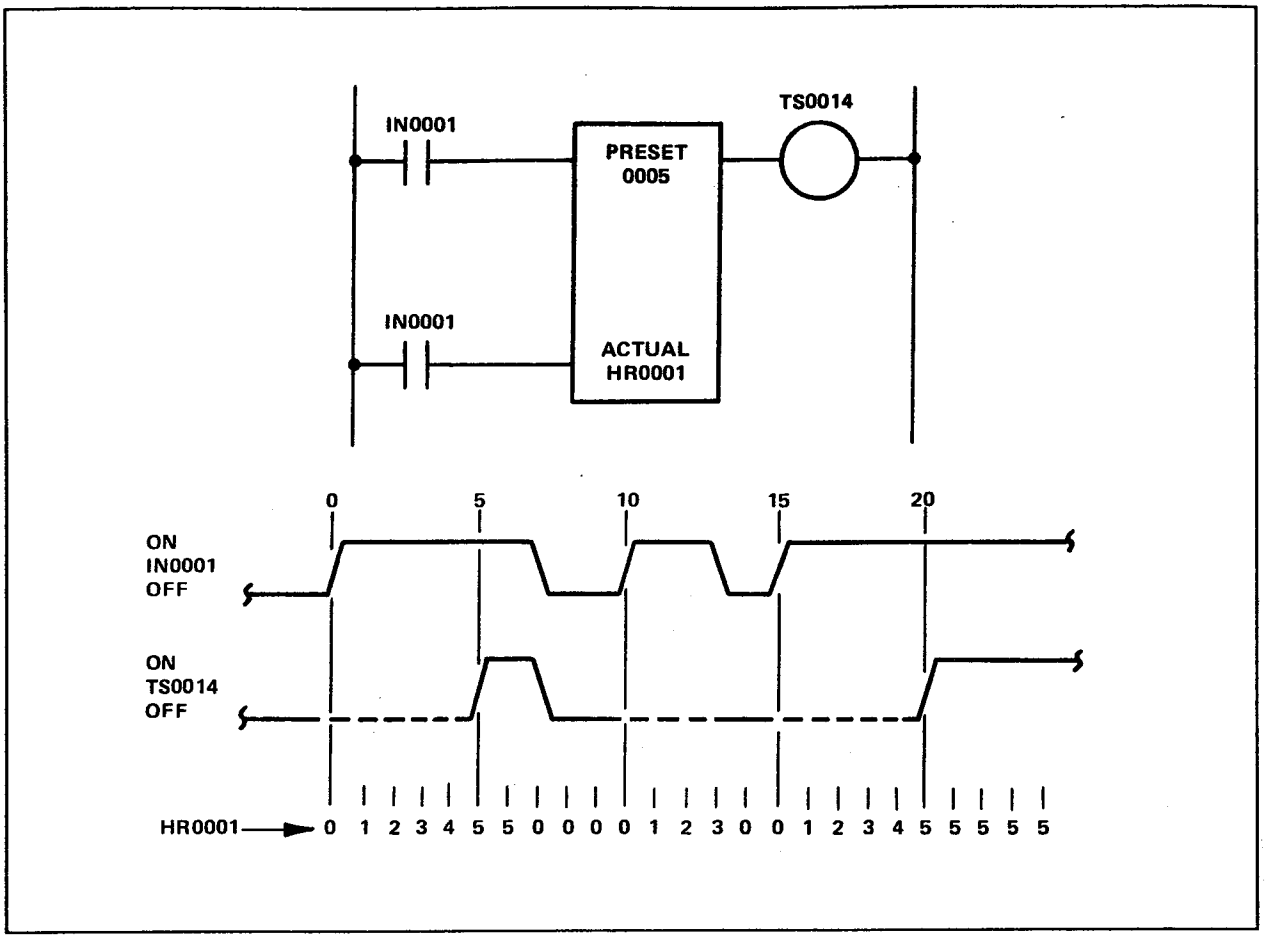


Figure 2. ON Delay Timer Circuit

SPECIFICATIONS

PRESET

The desired time value for the timer is the number of seconds (TS) or the number of tenths of seconds (TT). For example, if a TT timer is used, a preset of 0050 is equal to 5.0 seconds or 50 tenths of a second. This preset value may be a constant (0001 through 9999) or it may be a value held in a Holding Register (HR), Input Register (IR), or Output Register (OR). The range for using a register for preset is 1 through 65535.

ACTUAL

The current value of the timer is in seconds or tenths of seconds. This actual value is held in a Holding Register (HR) or Output Register (OR).

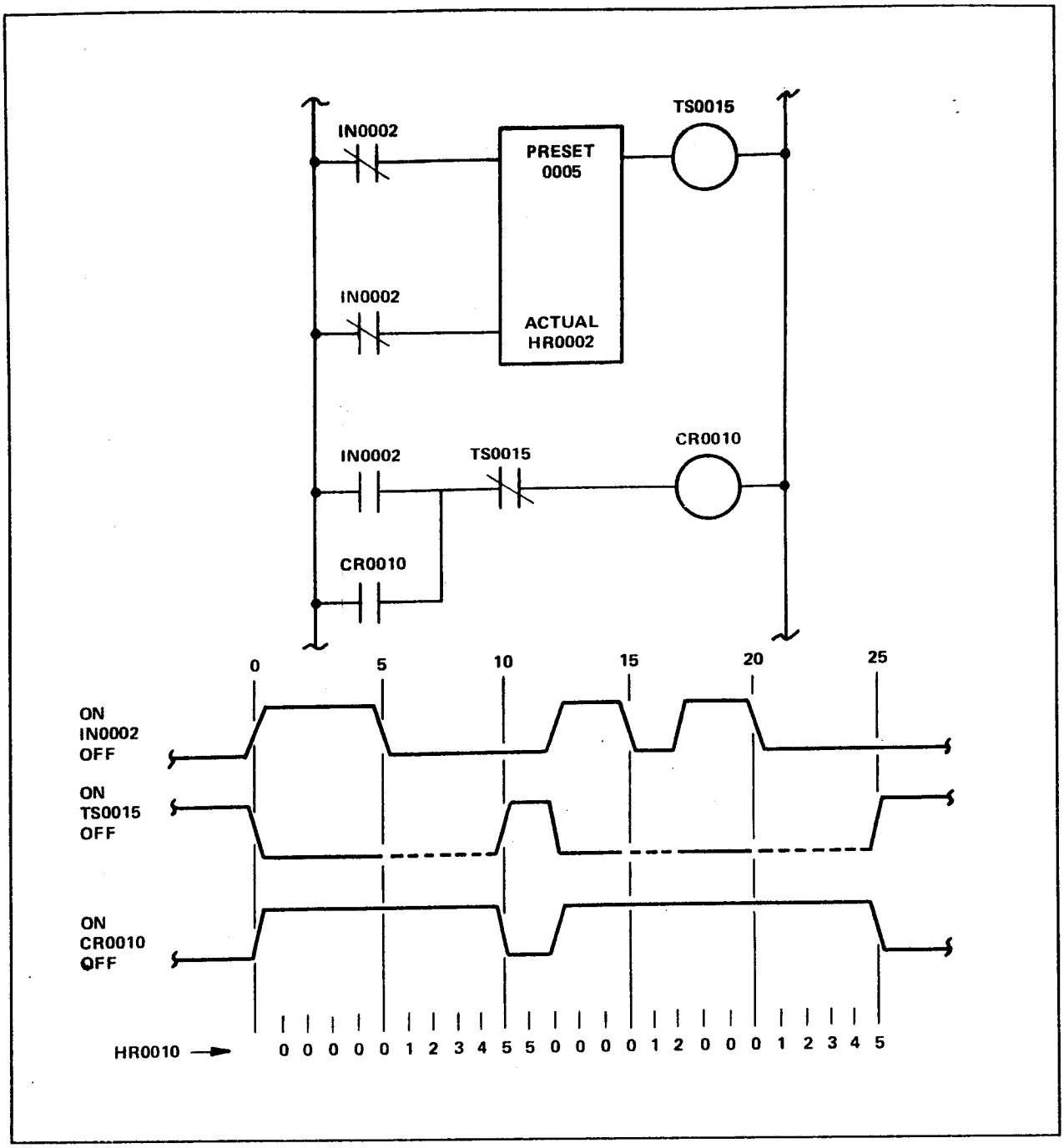


Figure 3. OFF Delay Timer Circuit

COIL

The coil is energized when actual value equals a preset value, with the enable circuit conducting.

TS/TT TRUTH TABLE

See Table 1.

TABLE 1. TS/TT TRUTH TABLE

Timing Circuit	Enable Circuit	Result
X	0	Actual is held to 0000.
0	1	Actual remains at current value.
1	1	Actual accumulates until actual equals preset.
X = Don't care		

APPLICATIONS

Figure 4 shows a timer circuit action under different conditions. The example in this figure is equivalent to a Motor Driven timer. At Time 0, the enable input (IN0004) and the timing input (IN0003) are closed. At Time 7, the enable input (IN0004) is opened, disabling the timer and setting the content of HR0006 to 0000. At Time 8, the enable circuit is closed and timing is started again. The timing process continues, and at Time 18, the actual value equals the preset value and the output coil (TT0012) is energized. The count of 10 is held in HR0006. At Time 20, both the timing (IN0003) and the enable (IN0004) circuits are opened, causing the TT0012 coil to de-energize and the actual value in HR0006 to reset to 0000. At Time 21, both the timing and the enable circuits are closed, causing the timer to accumulate time again. After two time periods (Time 23), the timing circuit (IN0003) is opened, causing the timer to stop but not to reset. HR0006 holds the last actual value until either the timing circuit is closed to resume timing or until the enable circuit is opened to reset the timer to 0000. At Time 25, the timing circuit closes and timing is resumed.

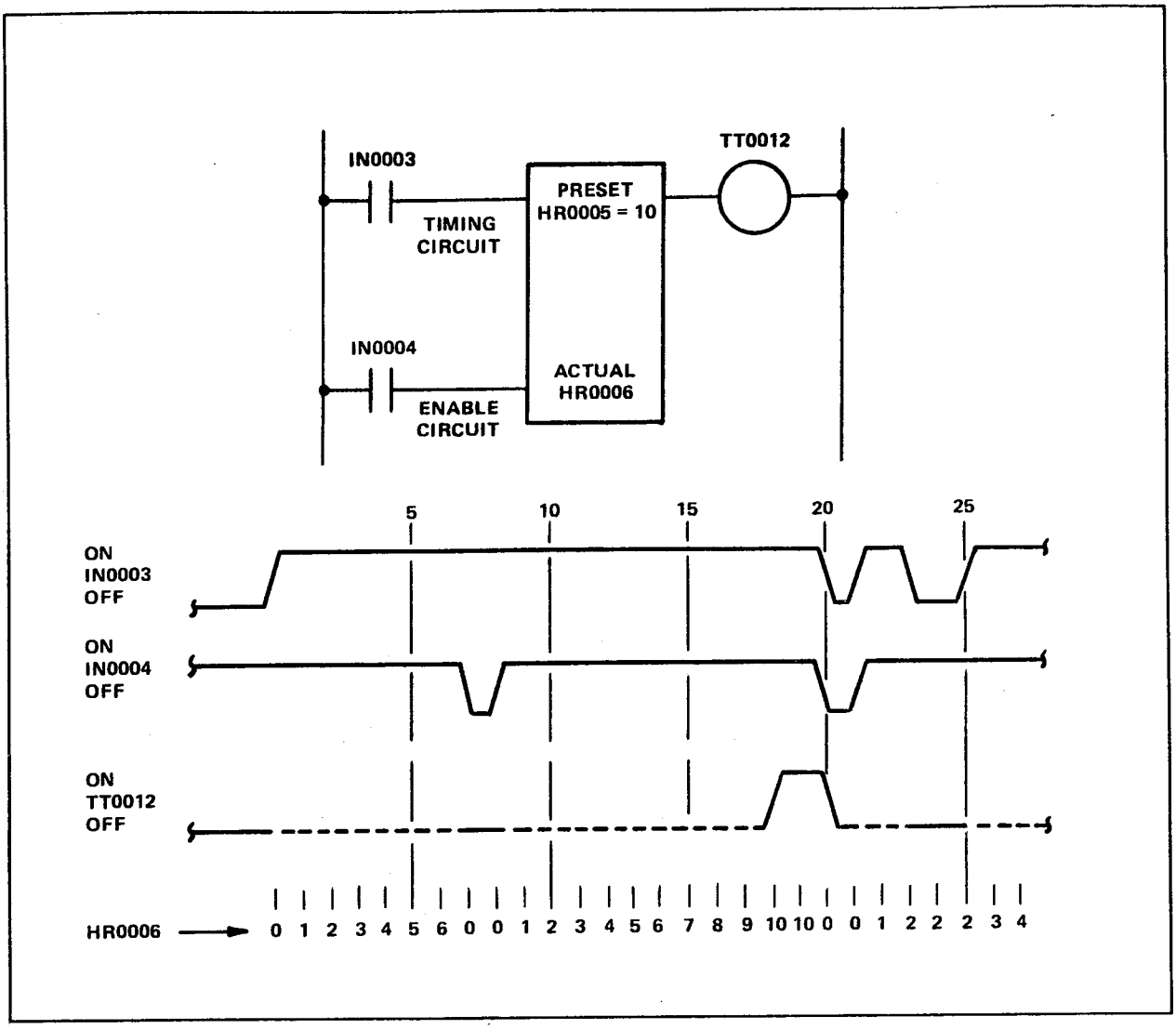


Figure 4. Motor-driven Timer

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: NOT SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Unit Address (UA) function defines the communications address of the PC in which it resides. It identifies this PC as a "Slave PC" and differentiates it from others when configured on a multi-point network. This address is compared to all "Set PC Address" commands initiated by the "Master PC". If the address matches, Port B is enabled for multi-point communications. [1]

Upon power-up and prior to solving the ladder logic, the executive memory is updated with the communications address. If more than one UA function is programmed, that which appears latest in the scan will be acknowledged. When the enable circuit transitions from off to on, the communications address is redefined according to Operand 1. UA function symbology is shown in Figure 1.

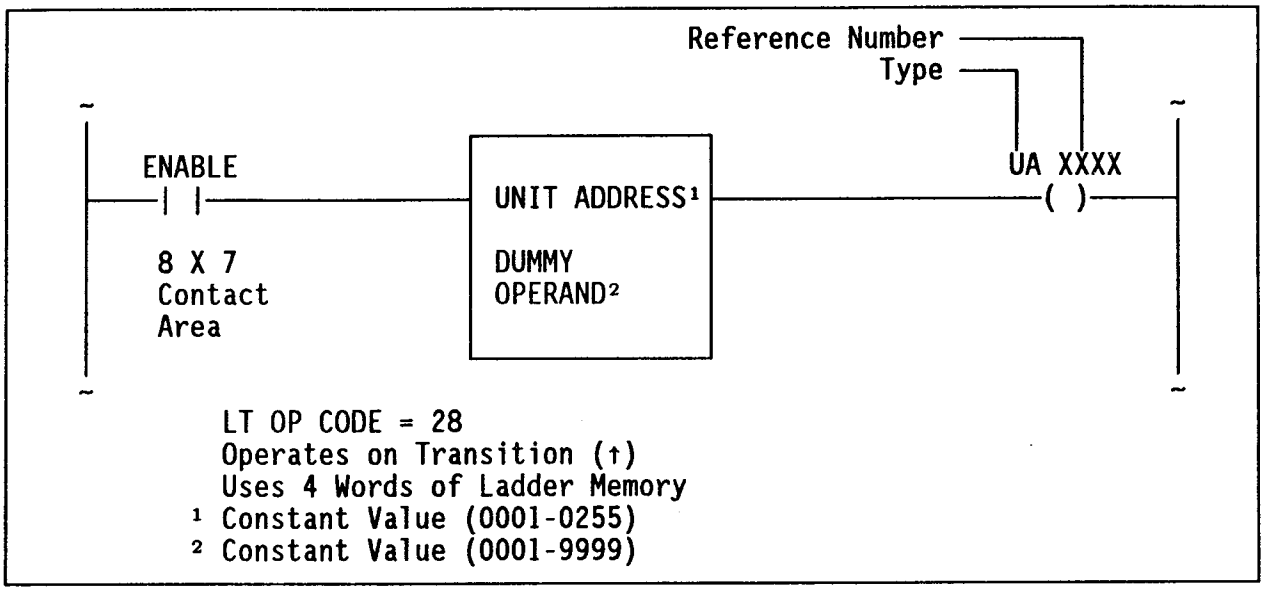


Figure 1. Unit Address (UA)

---

[1] See Paragraph 3-20 for an overview of PC-1100/1200 networking, and see the Instruction Leaflet for the Communications Adapter Module (IL-15753). For additional information on the related special functions (PT and CP) refer to their individual function descriptions in this section.

The controllers must be configured for multi-point communications as described in Paragraph 3-20. Configuration switch settings for the PC-1100 and PC-1200 are described in Paragraphs 3-17.

OP CODE

Op Code 28 defines the Literal (LT) as the UA function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - UNIT ADDRESS

The Unit Address is a constant value in the range of 1-255.

Note

If Operand 1 defines an invalid communications address then the PC generates a user software fault.

OPERAND 2 - DUMMY OPERAND

The Dummy Operand serves only to complete the UA function. It is defined by a constant value. (For clarity, set the dummy operand equal to the unit address.)

UA TRUTH TABLE

See Table 1.

TABLE 1. UA TRUTH TABLE

Enable	Result
0	The coil de-energizes. The communications address does not change.
↑	The coil energizes. The communications address is defined by Operand 1.
1	The coil status and communications address do not change.

When the controller is powered-up (or the keyswitch setting is changed from STOP/PROGRAM to RUN), it searches through the program for the UA function and sets the unit address. If more than one UA function is found, the last UA in the program is used as the default address.

In the PC-1200, the enable circuit must be energized to set the address. When the controller is powered-up, if no energized UA is found, the address will default to the invalid address "0". If a valid address is not set before the PT function is activated, the Slave controller will go into fault (bit 5 of the fault register).

APPLICATIONS

Consider the system configuration shown in Figure 2. The "Master PC" initiates the request to read or write register data of a "Slave PC" has its own unique address as established by the UA function This arbitrary address assignment identifies the Slave PCs with addresses 1 through 5. Any unique address from 1 through 255 could have been selected. Slave #4 is programmed with the UA function shown in Figure 3.

The "Master PC" can be configured in any position in the network. Note that the controllers which physically terminate the network require different switch settings (for the Port B communications switches). Also, for the PC-1200, the mode switch must be OPEN in the "Master PC" only. Refer to Paragraphs 3-17 and 3-20 for additional information on the configuration switch settings.

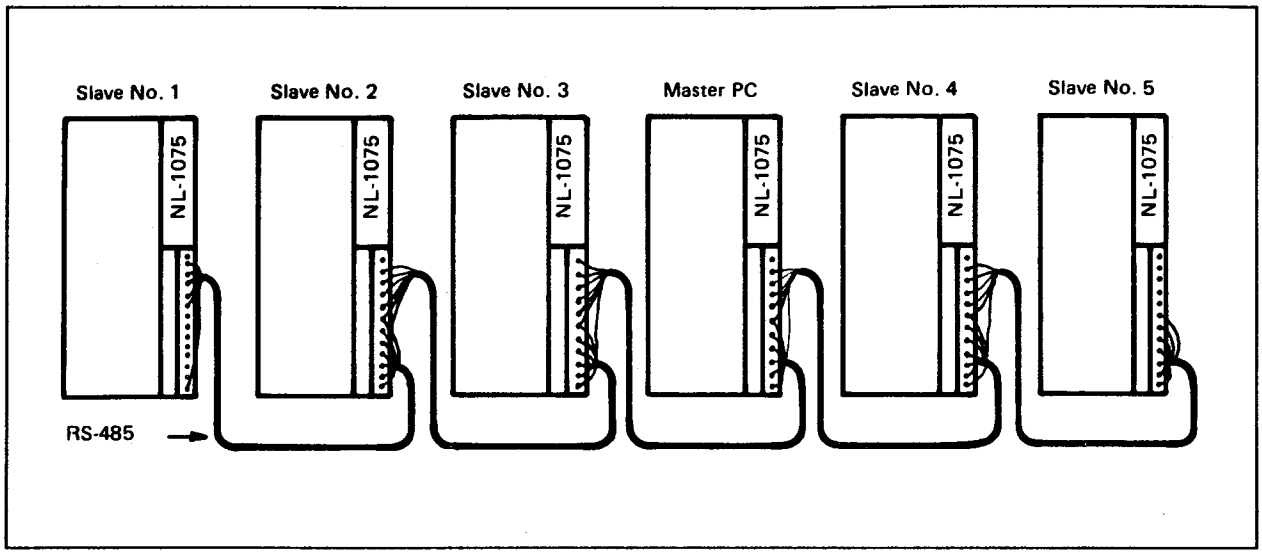


Figure 2. Multi-Point Network

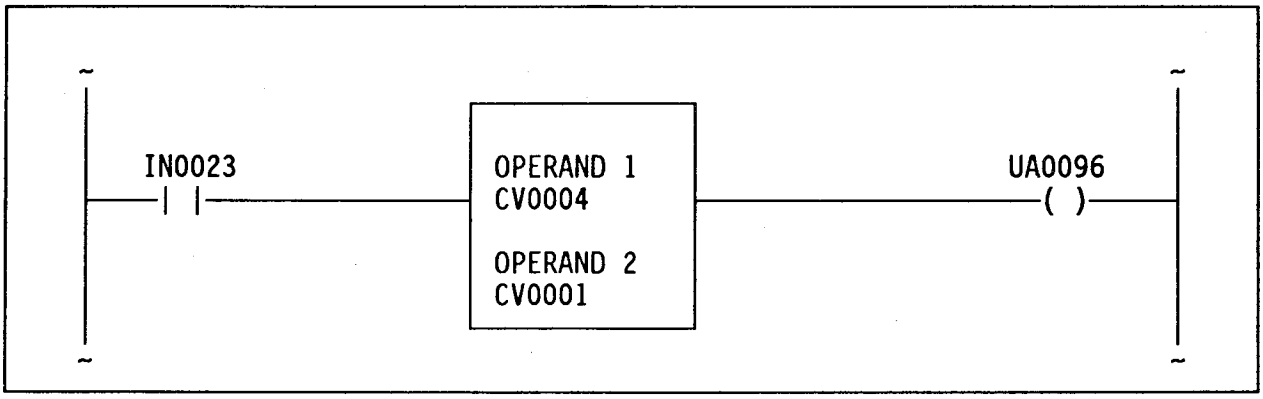


Figure 3. Unit Address

UC/DC-COUNTERS

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

A counter is similar to a timer, except that it does not operate on an internal clock and is dependent on external or program sources for counting circuit control. Up Counter (UC)/Down Counter (DC) function symbology is shown in Figure 1.

UP COUNTER (UC)

UP Counters (UCs) begin at 0000 and count to a maximum of 65535. The counter coil is energized upon reaching a preset value.

Two contact circuits control UCs: the counting circuit and the enable circuit. Counting is allowed when the enable circuit is conducting. The counter's accumulated value is reset and held at 0000 when the enable circuit is not conducting. During the change of the counting circuit from non-conducting to conducting, the UC increases the accumulated count by one and retains this count as long as the enable circuit is conducting.

The accumulated count is stored in the actual register (an assigned holding or output register). The coil is energized and its contacts are operated when the actual value equals or exceeds a preset value, which is programmed or comes from a register, simultaneously with the enable circuit conducting. The counter can continue past the preset value until 65535 is reached, and remains at that count until the enable circuit stops conducting.

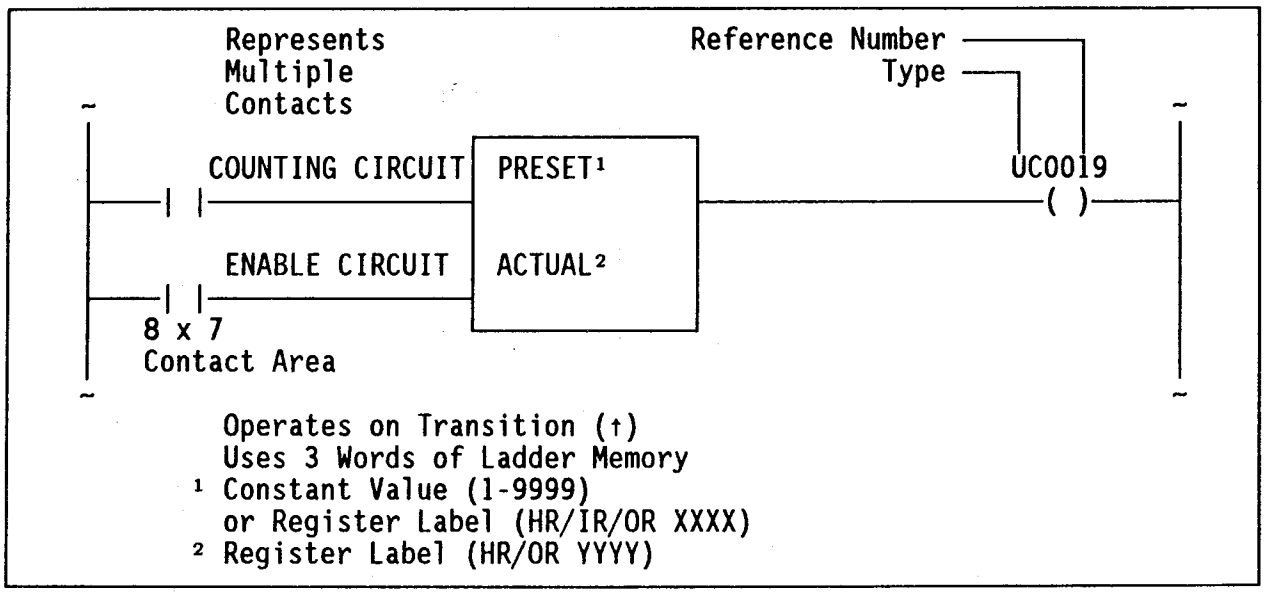


Figure 1. UP Counter (UC)/Down Counter (DC)

DOWN COUNTER (DC)

Down Counters (DCs) begin at a maximum (9999 if preset is a constant, 65535 if preset is a register), and decrement to 0000. The counter coil energizes upon reaching 0000.

The counting and the enable circuits control DC similarly to UC. Only two changes occur in the counting method. The accumulated value is reset and held at the preset value instead of 0000, and the count is decreased, not increased, by one when the counting circuit changes from non-conducting to conducting. The actual value is held at the preset value if enable is non-conducting. The count decreases on transition of the count input when the enable circuit is conducting.

Again, the count is stored in the actual register. Counting begins at the preset value, constant or variable. The coil is energized and its contacts are operated when the count reaches 0000. The count remains at 0000 until the enable circuit stops conducting.

SPECIFICATIONS

PRESET

UC is preset at the value at which the coil energizes.

DC is preset at the value at which down counting begins.

This value may be a constant (0001 through 9999) or it may be held in a Holding Register (HR), Input Register (IR), or Output Register (OR). The value is 1 through 65535 if it is held in a register.

ACTUAL

The current value is held in:

- Holding Register (HR)
- Output Register (OR)

COIL

In the up count, the coil is energized when the actual value is greater than or equal to preset value, with the enable circuit conducting. In the down count, the coil is energized when the actual value is equal to zero, with the enable circuit conducting.

UC/DC TRUTH TABLE

See Table 1.

TABLE 1. UC/DC TRUTH TABLE

Count	Enable	UC Result	DC Result
X	0	Actual is held at zero.	Actual is held at preset.
0	1	Actual is held at current value.	Actual is held at current value.
↑	1	Counter increments.	Counter decrements.
1	1	Actual is held at current value.	Actual is held at current value.
X = Don't care		Note When actual equals preset, the counter continues past preset.	Note When actual equals zero, the counter stops.

APPLICATIONS

An Up/Down counter can be used to maintain a count of the product within a zone of a conveyor, by programming UC and DC functions which share the same actual register. An example of this type of application is shown in Figure 2.

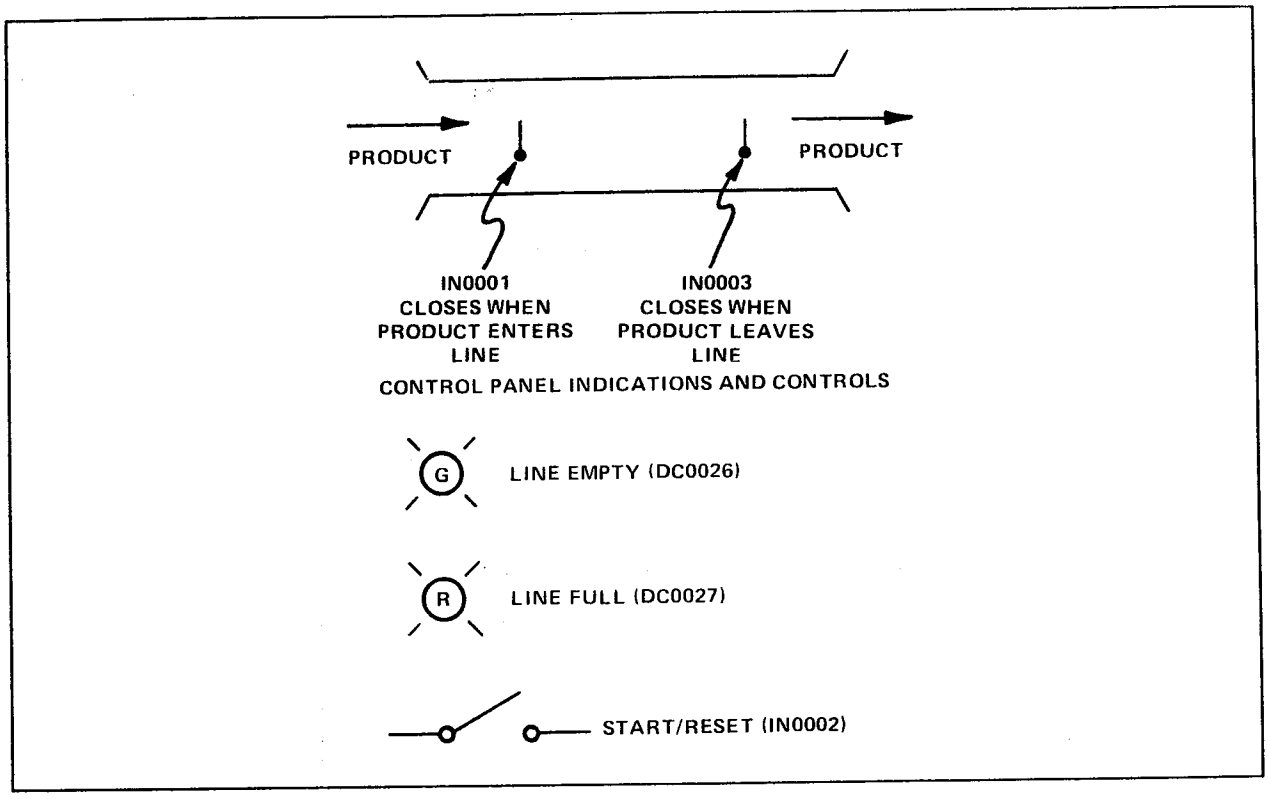


Figure 2. Shared Counter Application

In the shared register counter application (see Fig. 3), UC determines the actual value of DC. The DC function appears first in the program. If IN0002 is closed, both counters are active. When IN0001 is closed, HR0002 increments by one and continues to increment every time IN0001 is opened and closed. If IN0003 is opened and closed, the contents of HR0002 are decremented by one and continue to decrement until 0000 is reached.

Note that if IN0002 is opened, HR0002 will be set to 9999 by the DC function, then immediately reset to zero when the UC function is executed. The entire program will see HR0002 as zero when IN0002 is off.

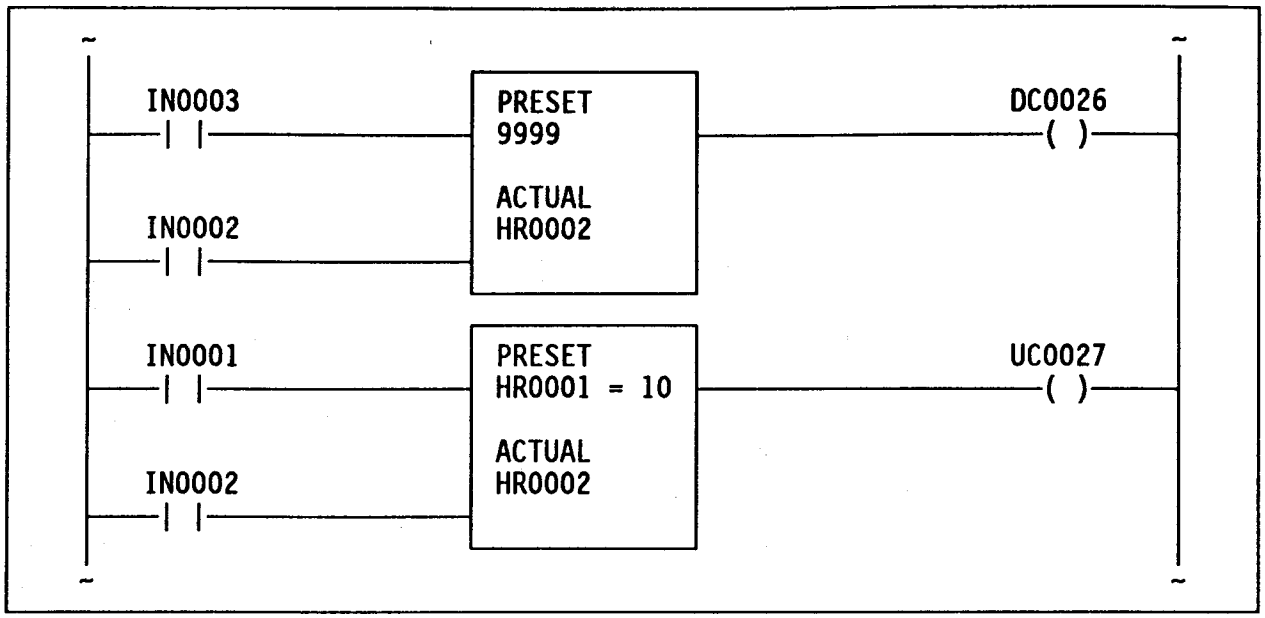


Figure 3. Shared Register Counter

UI - I/O UPDATE IMMEDIATE

Modified for PC-1200

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The I/O Update Immediate (UI) function updates the status of registers or groups when the UI function is executed in the logic scan, in addition to normal I/O update at the end of the logic scan. Energizing a UI enable input updates the status of the specified input register or group and its corresponding output register or group. De-energizing a UI enable input stops the update routine. UI function symbology is shown in Figure 1.

Note

The UI function cannot be programmed using the NLPL-789, NLPL-780 or NLPL-780P program loaders with software versions prior to 3.0.

SPECIFICATIONS

OPERATING DATA

The designation specifies the I/O pair for updating. This can be a pair of single registers or a group. One of the following may be used:

- IR: Input Register/Output Register
- IG: Input Group/Output Group

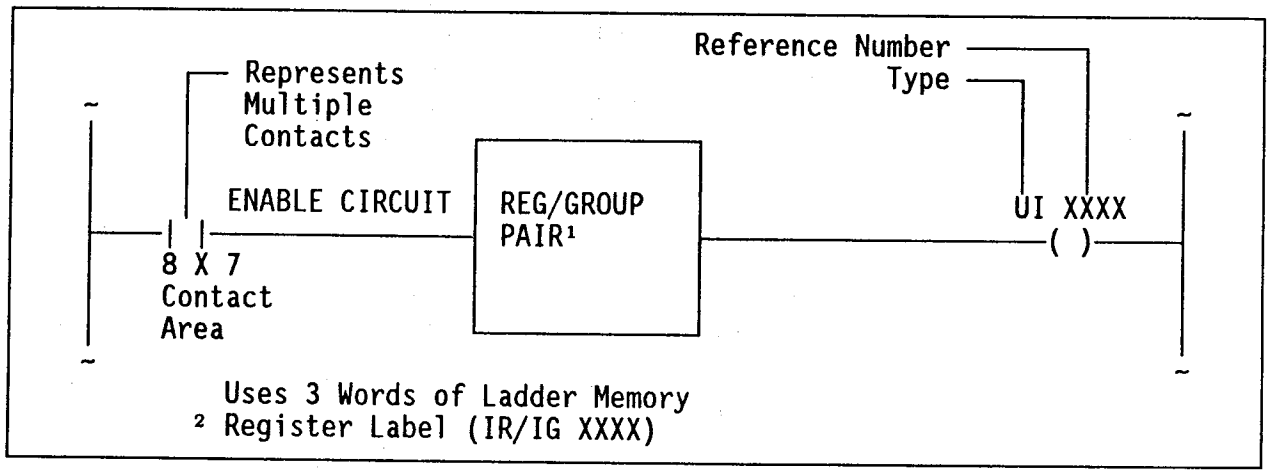


Figure 1. I/O Update Immediate (UI)

In the PC-1250, the Low and High I/O busses are updated simultaneously. With this function, the referenced group or register is updated simultaneously on *both buses*. For example, if group label IG0001 is referenced, the following will be updated: IG0001, OG0001, IG0009, and OG0009. Table 1 shows the Low and High bus groups and registers.

TABLE 1. UPDATE IMMEDIATE FUNCTION WITH DUAL I/O BUS (PC-1250 ONLY)

Reference Number	Updates on Low Bus	Updates on High Bus
IG0001 or IG0009	IG0001, OG0001	IG0009, OG0009
IG0002 or IG0010	IG0002, OG0002	IG0010, OG0010
•	•	•
•	•	•
•	•	•
IG0008 or IG0016	IG0008, OG0008	IG0016, OG0016
IR0001 or IR0065	IR0001, OR0001	IR0065, OR0065
IR0002 or IR0066	IR0002, OR0002	IR0066, OR0066
•	•	•
•	•	•
•	•	•
IR0064 or IR0128	IR0064, OR0064	IR0128, OR0128

ENABLE CIRCUIT

When conducting, the enable circuit energizes the UI function, and the specified pair of I/O registers or groups are updated. When not conducting, the enable circuit de-energizes the UI function and no immediate updating occurs during the logic scan.

COIL

When the coil is energized, the specified pair of I/O registers or groups is updated immediately. When the coil is de-energized, no updating occurs.

APPLICATIONS

The UI function can be used to periodically monitor the status of critical inputs. For example, if the user wants output GE0037 (CR0037) to activate immediately when IR0006 is greater than or equal to 1024, the program shown in Figure 2 can be repeatedly programmed at appropriate intervals throughout the ladder program. In this example, UI0126 is always ON when GE0037 is executed. If GE0037 is not turned ON, UI0126 is turned OFF for remainder of the scan.

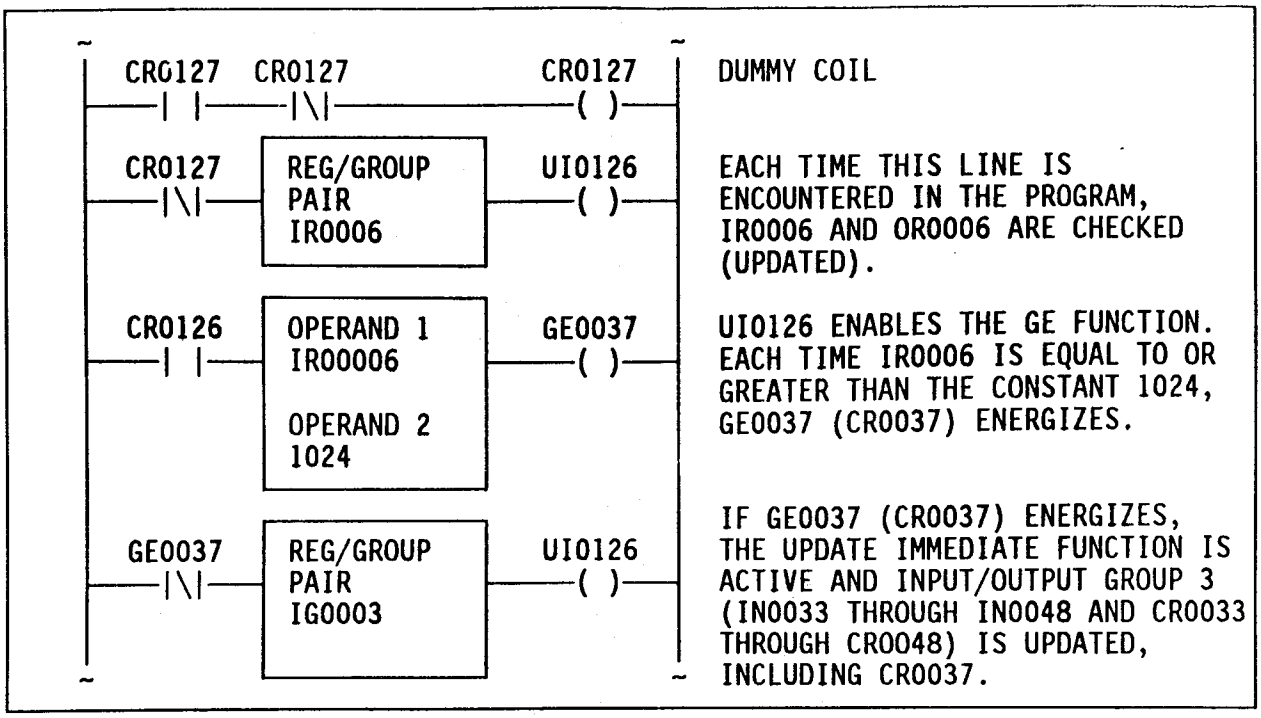


Figure 2. UI Operations

Figure 3 shows a method of counting faster than two times the scan rate. Input IN0020 is the count input.

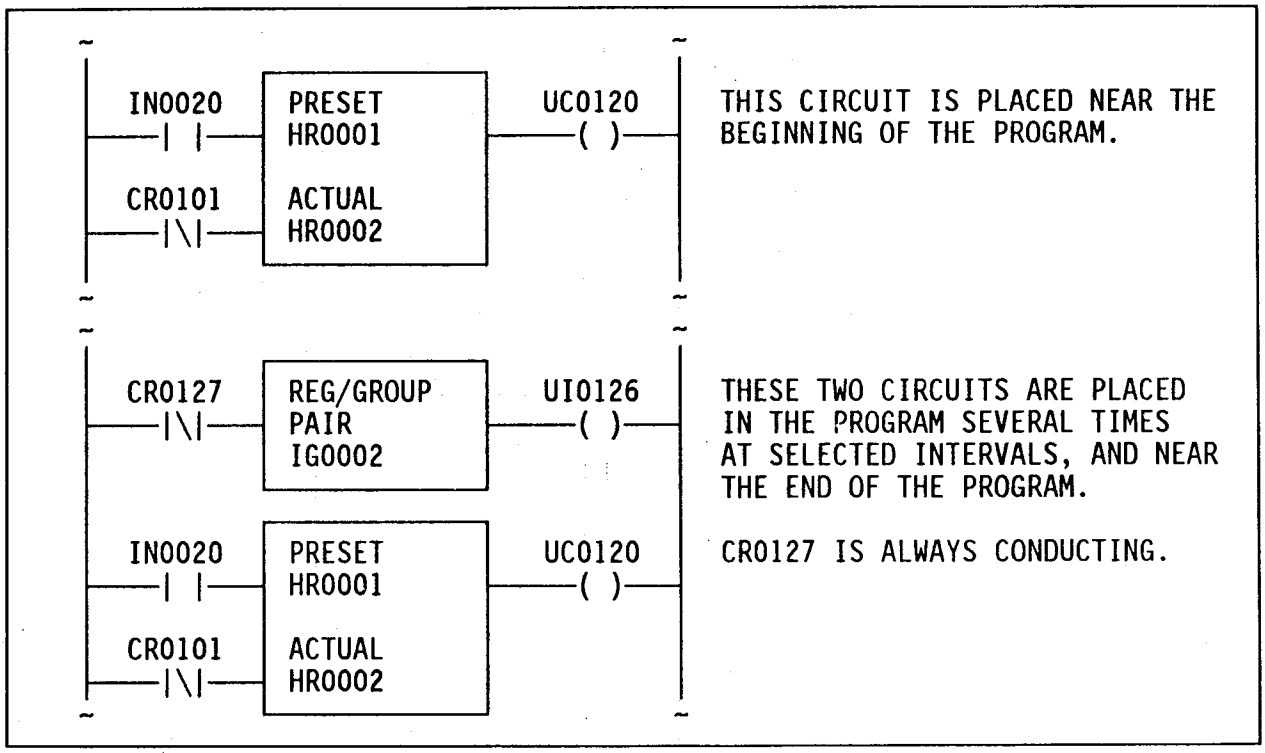


Figure 3. UI Application

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The Update Select (US) function allows the user to specify the number of quarters of I/O that are updated at the end of a processor scan. The I/O update cycle time is effectively decreased by eliminating the time spent updating unused I/O quarters. US function symbology is shown in Figure 1.

OP CODE

Op Code 25 defines the Literal (LT) as the US function. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

OPERAND 1 - I/O QUARTERS

A constant value determines the number of I/O quarters to be updated when the function is enabled. This value selects the number of quarters (of I/O groups and registers) to be updated at the end of the ladder scan.

If more than one US function is active in a ladder, the last active US will determine the amount of I/O quarters that will be updated.

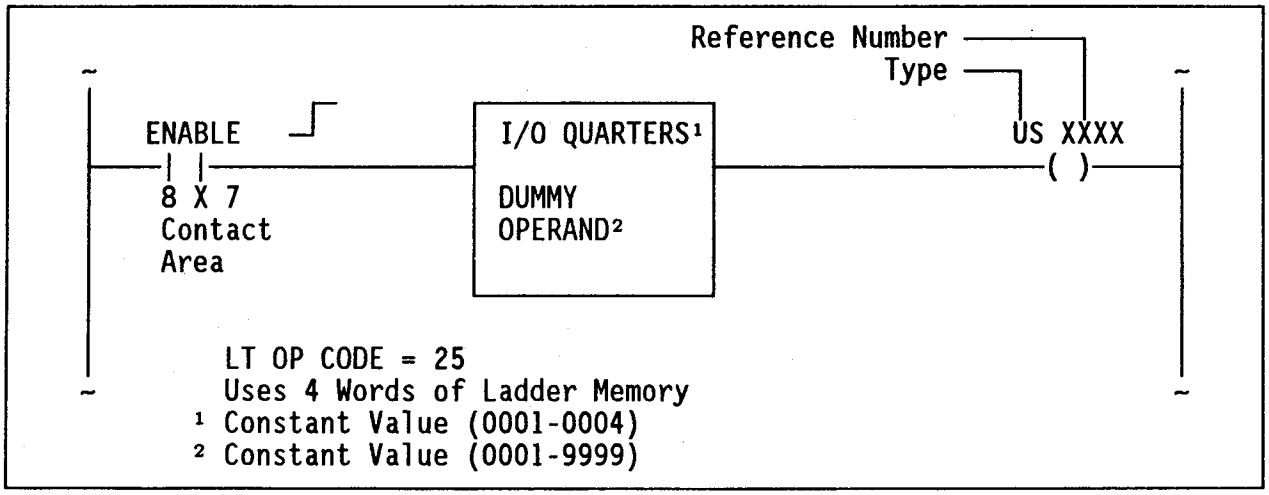


Figure 1. Update Select (US)

The I/O quarters value selects the number of quarters as follows:

1 = First Quarter
2 = First Half (Two Quarters)
3 = First Three Quarters
4 = All

The I/O reference numbers corresponding to the I/O quarters values are shown in Table 2.

TABLE 2. I/O QUARTERS

Controller	Number of I/O Quarters	IG and OG	IR and OR
PC-1100	1	1	1-2
	2	1-2	1-4
	3	1-3	1-6
	4	1-4	1-8
PC-1200-1020/40	1	1	1-8
	2	1-2	1-16
	3	1-3	1-48
	4	1-4	1-64
PC-1200-1041/42/43	1	1-2	1-16
	2	1-4	1-32
	3	1-6	1-48
	4	1-8	1-64
PC-1250 [1]	1	1-2 and 9-10	1-16 and 65-80
	2	1-4 and 9-12	1-32 and 65-96
	3	1-6 and 9-14	1-48 and 65-112
	4	1-16	1-128

[1] In the PC-1250, the corresponding quarters of the Low and High buses will be updated. If all quarters are not selected, then gaps will occur, due to the offset between the buses. For example, if Operand 1 = 1, then the following will be updated: IG/OG 1-2 and 9-10, IR/OR 1-16 and 65-80. The reference numbers between these ranges (in this example, IG/OG 3-8 and IR/OR 17-64) will not be updated.

OPERAND 2 - DUMMY OPERAND

The dummy operand is a constant value of 1 to complete the special function.

US TRUTH TABLE

See Table 3.

TABLE 3. US TRUTH TABLE

Enable	Result
0	The coil is de-energized and all I/O is updated.
1	The coil is energized and the specified I/O quarters are updated.

APPLICATIONS

If the user has a program whose inputs and outputs are in Group 1 only (Discrete Inputs 1 through 16, Discrete Outputs 1 through 16, Input Registers 1 through 2, Output Registers 1 through 2), the US function allows only the first quarter to be updated, increasing the processor scan rate. The program segment in Figure 2 shows this operation. US always enables, allowing only the first quarter of the I/O to update.

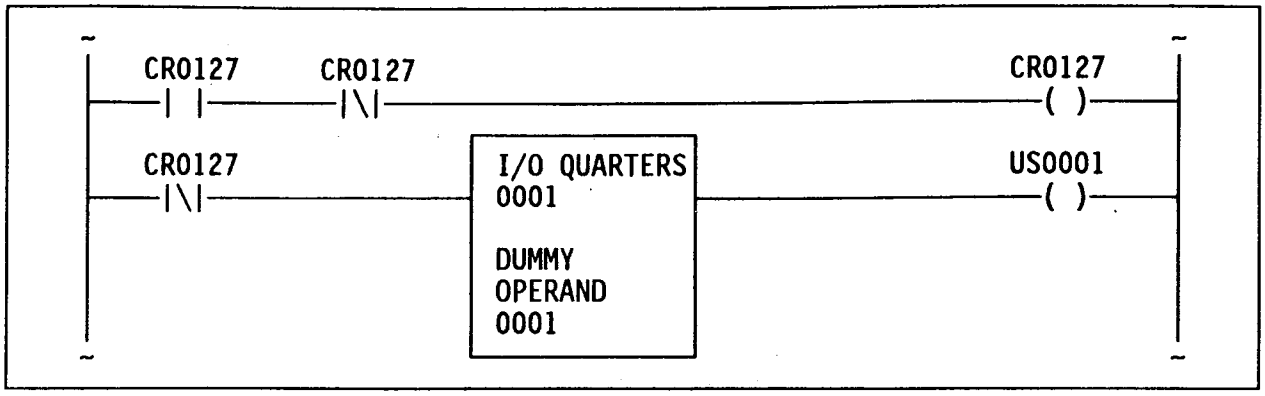


Figure 2. US Program

The US function can also be used to provide varying update frequency for different portions of the I/O. If some portions of I/O are not as critical as others, update frequency can be arranged accordingly. The ladder diagram in Figure 3 shown this operation. The self-resetting timer, TT0001, energizes for one scan every 10 seconds. The first half of the I/O is updated every scan. All of the I/O updates every 10 seconds.

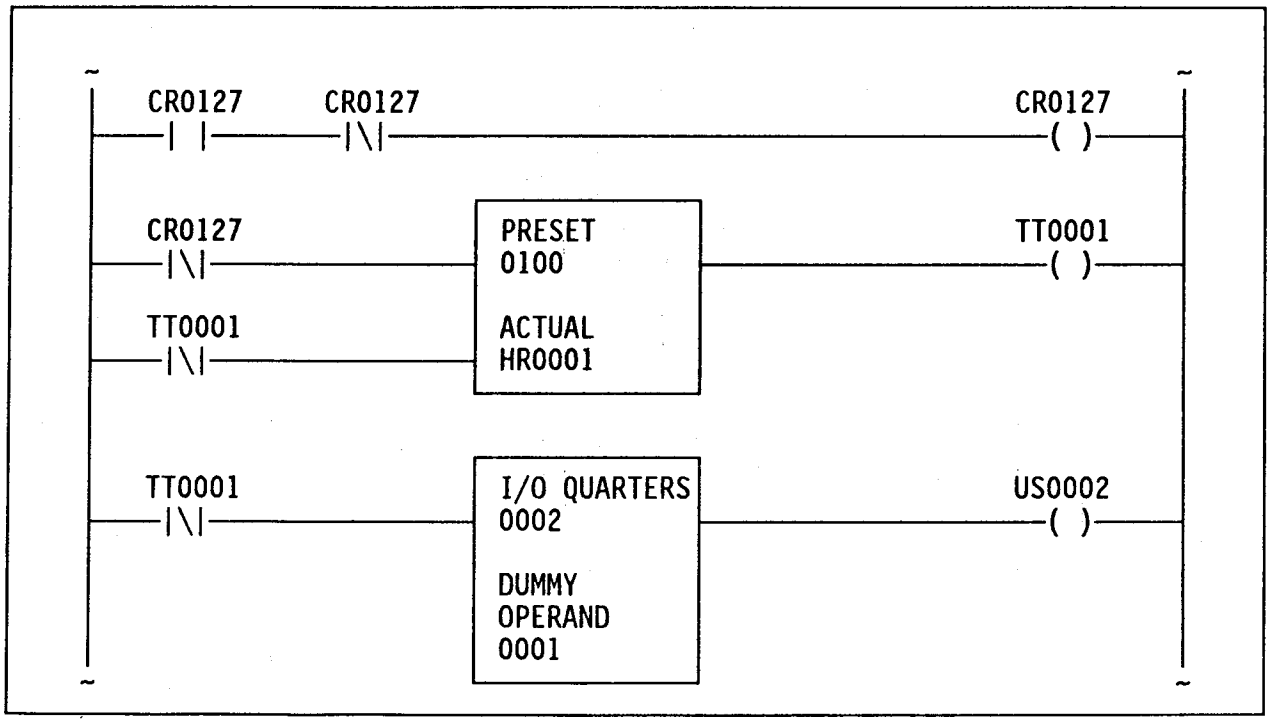


Figure 3. US for I/O at Differing Rates

XM-XOR MATRIX

PC-1100-x01y: NOT SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

The XOR Matrix (XM) function exclusively OR's the contents of a pair of matrices on a bit-per-bit basis; then, it places the result in a destination matrix location. A matrix is defined as a table of registers handled on a bit-by-bit basis. XM function symbology is shown in Figure 1.

The XM operation occurs when the enable circuit changes from non-conducting to conducting. The original pair of matrices is unaffected by the operation. See Table 1.

OP CODE

Op Code 88 defines the Literal (LT) as XM. Whether or not the Literal function should be used depends upon the capability of your program loader. It is recommended that the mnemonic function be used whenever possible. Refer to the Introduction in this Section and the LT function description.

SPECIFICATIONS

COIL

The coil energizes when the enable circuit is conducting and the result of the operation is non-zero.

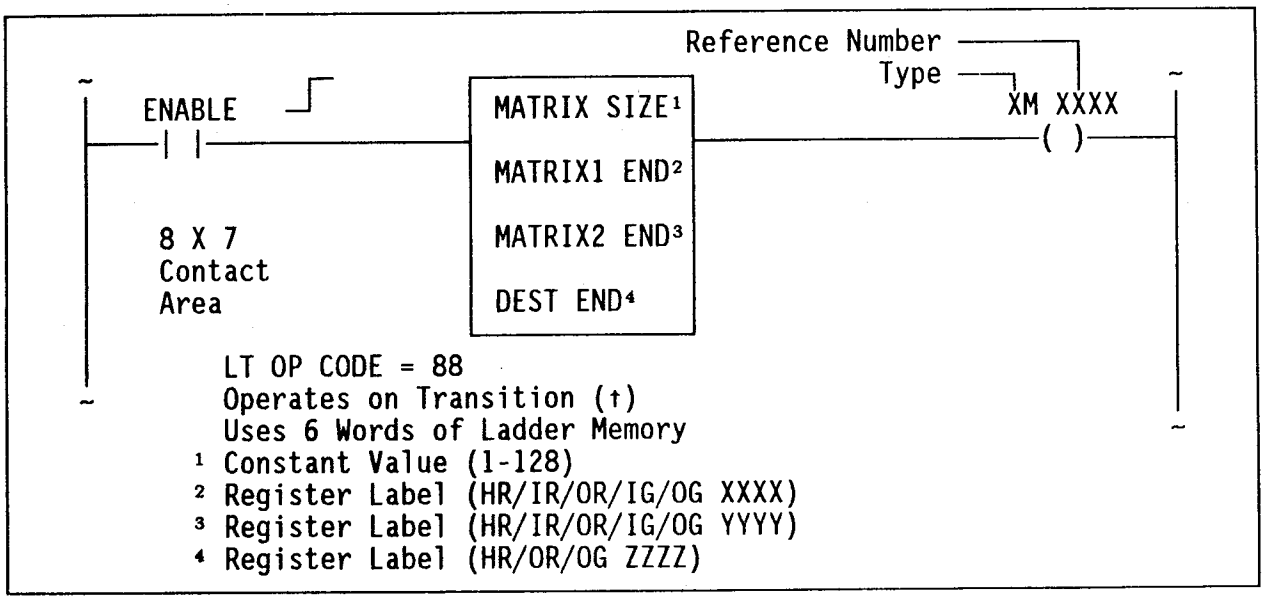


Figure 1. XOR Matrix (XM)

TABLE 1. XM TRUTH TABLE SAMPLE

Matrix 1 Bit N	Matrix 2 Bit N	Destination Matrix Bit N
0	0	0
0	1	1
1	0	1
1	1	0

OPERAND 1 - MATRIX SIZE

The matrix size is a constant value that defines the number of registers included in the matrix. The range is 1 through 128, which is subject to the limitations listed in Table 2.

TABLE 2. XM END REGISTERS

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
IR	≤ 8	≤ 32	≤ 64	≤ 128
OR	≤ 8	≤ 32	≤ 64	≤ 128
IG	≤ 4	≤ 4	≤ 8	≤ 16
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

OPERANDS 2/3 - MATRIX 1/MATRIX 2 END

Matrix 1/Matrix 2 end defines the type and number of the last register in Matrix 1 and Matrix 2 that will be exclusively OR'ed. The type and number are limited as indicated in Table 2.

OPERAND 4 - DESTINATION END

The destination end defines the type and number of the last register in the matrix containing the results of the XM function. The type and number limitations are shown in Table 3.

TABLE 3. XM DESTINATION END REGISTER

Type	PC-1100	PC-1200-1020/1040	PC-1200-1041/1041/1042	PC-1250
HR	≤ 1792 [1]	≤ 1792	≤ 1792	≤ 1792
OR	≤ 8	≤ 32	≤ 64	≤ 128
OG	≤ 8	≤ 32	≤ 64	≤ 128

[1] For the PC-1100, the maximum number of holding registers depends on memory size, as described in Section 4.

XM TRUTH TABLE

See Table 4.

TABLE 4. XM TRUTH TABLE

Enable	Result
0	The coil is de-energized. Matrix 1, Matrix 2, and the destination remain unchanged.
↑	XOR's Matrix 1 with Matrix 2, placing the results in the destination The coil energizes if the result is non-zero, and de-energizes if the result is zero.
1	The coil, as indicated above, results in the destination matrix.

APPLICATIONS

The XM function is used to make desired state versus actual state comparisons. If the actual state is equal to the desired state, the coil is de-energized when the enable circuit is conducting. If the actual state does not equal the desired state, the coil is energized with the enable circuit conducting. See Figure 2.

The ladder diagram for the XM function is shown in Figure 3. When IN0001 changes from open to closed, the XM function occurs. If IN0001 remains closed, the coil gives the following indications:

Energized = Matrix 1 ≠ Matrix 2
De-energized = Matrix 1 = Matrix 2

Figure 4 shows an example of a pair of matrices that are exclusively OR'ed.

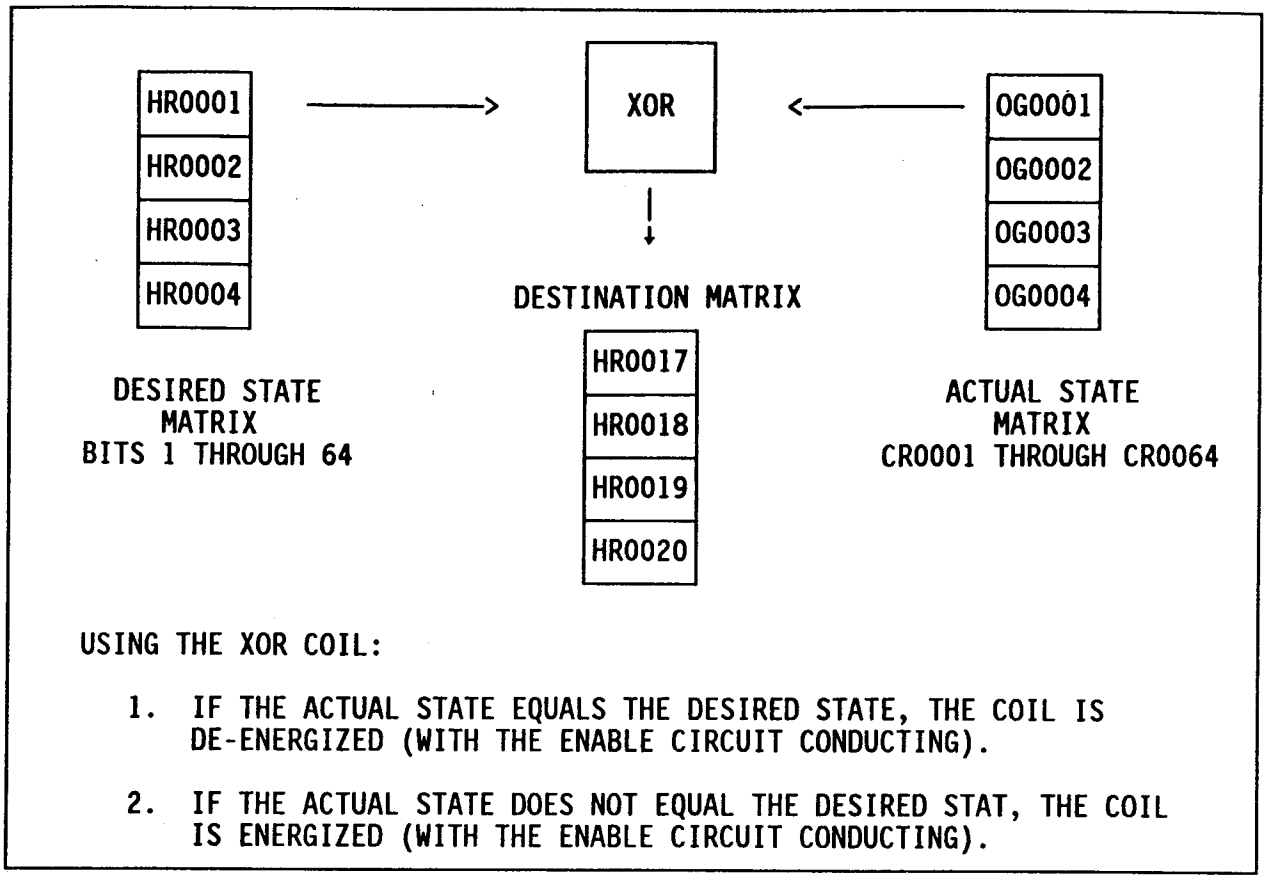


Figure 2. Desired State vs. Actual State Comparison

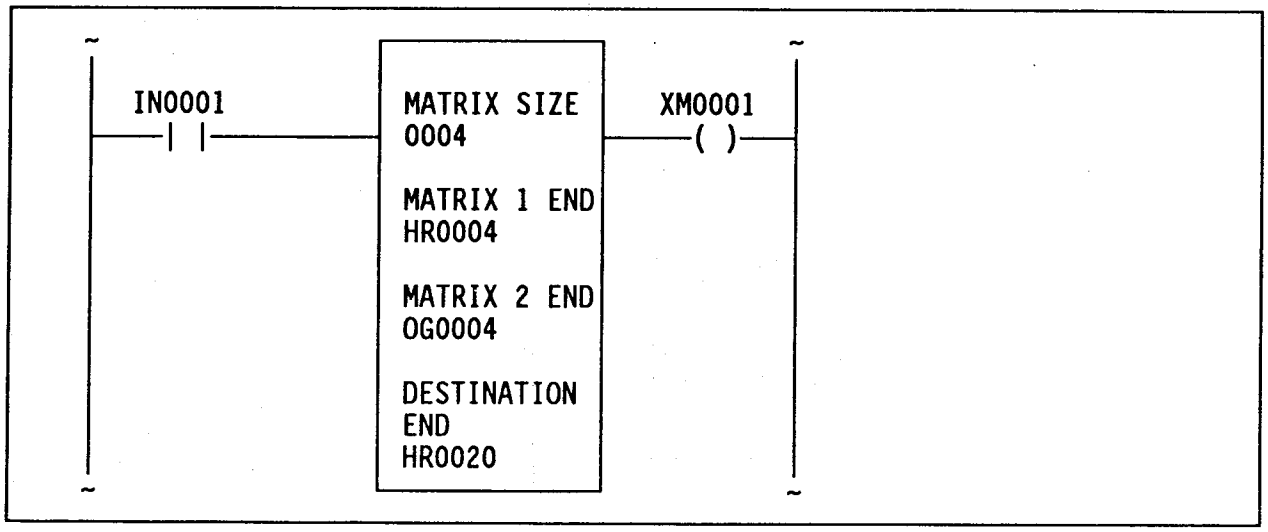


Figure 3. XM Application

MATRIX 1 END

MATRIX 1

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0001	1	0	0	1	0	1	0	0	1	1	1	1	0	0	1	1

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0002	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0003	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

MATRIX 2 END

MATRIX 2

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0004	0	1	1	0	0	0	0	0	0	0	1	0	1	1	0	0

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0005	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0006	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

DESTINATION END

MATRIX 3

	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
HR0001	1	1	1	1	0	1	0	0	1	1	1	1	1	1	1	1

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
HR0002	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 4. A Pair of Exclusively OR'ed Matrices

BATTERY STATUS COIL
Modified for PC-1200

PC-1100-x01y: SUPPORTED	PC-1100-x05y: SUPPORTED
PC-1100-x02y: SUPPORTED	PC-1200-x02y: SUPPORTED
PC-1100-x03y: SUPPORTED	PC-1200-x04y: SUPPORTED

DESCRIPTION

Battery Status Coil is a dedicated-function logic coil which is de-energized as long as the battery back-up system for the RAM memory is operating normally. The coil is energized when the battery backup system is de-activated or is capable of supporting the RAM for less than one week.

The dedicated coil is CR0128 in the PC-1100.

The dedicated coil is CR1024 in the PC-1200.

Note that it is possible to program the Battery Status Coil as a logic coil or reference number for a special function. However, the Battery Status Coil is reset at the end of each ladder scan (to reflect the status of the battery). For this reason, such use of this reference number (0128 in the PC-1100 or 1024 in the PC-1200) is not recommended.