TABLE 2. FI TRUTH TABLE

Step	Reset	Result
X	0	The coil de-energizes. The pointer is set to zero.
↑	1	The source moves to the position in the stack designated by the pointer. After the move, the pointer increments by 1.
1	1	The coil energizes when the pointer value equals the stack size and de-energizes when the pointer value is less than the stack size.

LO TRUTH TABLE

See Table 3.

TABLE 3. LO TRUTH TABLE

Step	Result
0	The coil de-energizes. No data transfers.
↑	The pointer decrements by 1. The destination transfers data at the pointed location.
1	The coil energizes when the pointer equals zero.

APPLICATIONS

The FILO Stack function monitors a system(s) for fault conditions and status when a fault occurs. Assuming that the system in question resets itself in the case of non-critical faults, the monitoring processor maintains a fault/fault status history. When hard or recurring faults occur, machine status is recalled in reverse order; the most recent fault status is recalled first. The processor is not part of the monitored system and is unaffected by system failures.

As shown in Figure 4, the status of selected switches and contacts is entered into the processor via IR0001 when a fault occurs in the operating system.

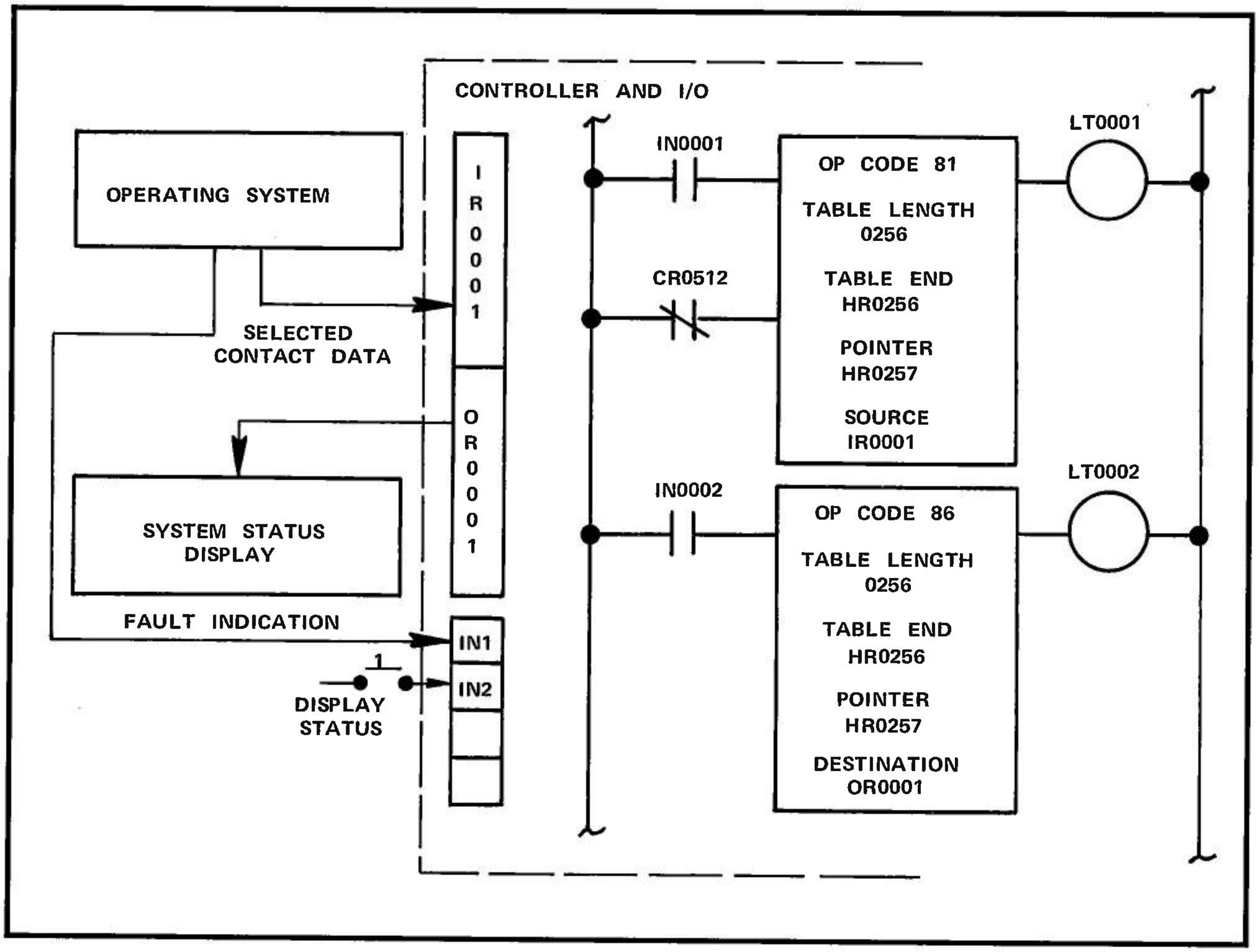


Figure 4. FILO Stack Application

MP — MULTIPLY

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 99,930,001). 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 (DB) 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.

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.

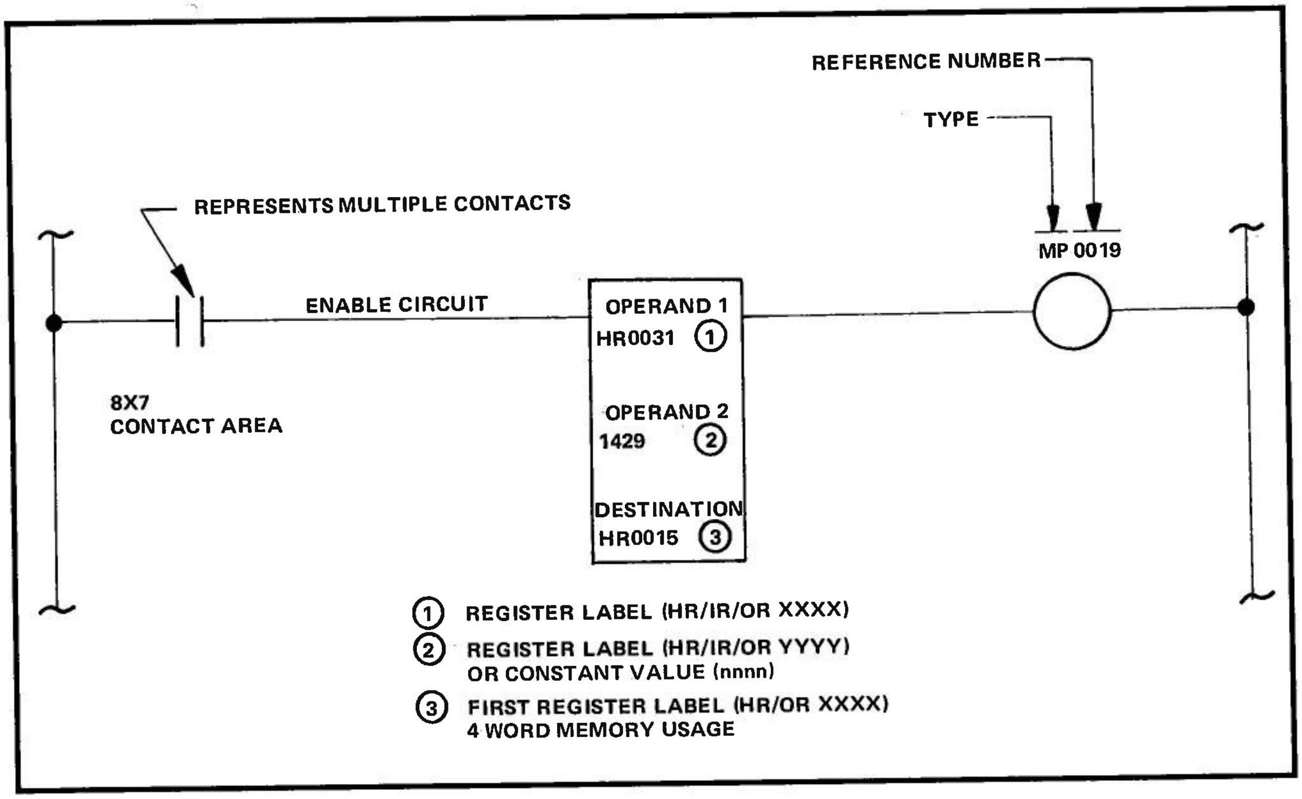

REFERENCE NUMBER
TYPE
REPRESENTS MULTIPLE CONTACTS
MP 0019
ENABLE CIRCUIT
OPERAND 1
HR0031 ①
8X7
CONTACT AREA
OPERAND 2
1429 ②
DESTINATION
HR0015 ③
① REGISTER LABEL (HR/IR/OR XXXX)
② REGISTER LABEL (HR/IR/OR YYYY) OR CONSTANT VALUE (nnnn)
③ FIRST REGISTER LABEL (HR/OR XXXX) 4 WORD MEMORY USAGE

Figure 1. Multiply (MP)

MP

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.

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 0000 through 9999. 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.
1	The coil is energized.

↑ = Transistion OFF to ON.

APPLICATIONS

The MP function is particularly useful when scaling factor inputs. For example, assume that a 0 through 10 V analog signal represents a true decimal range of 0 through 8190. The analog input module converts the 0 through 10 V range into a Binary-Coded-Decimal (BCD) range of 0 through 4095, allowing the processor to operate on the proper range of data. This converted, BCD analog data is then multiplied by 0002 in the processor.

Figure 2 shows the scaling of the input data by a factor of 0002. The 0 through 10 V (0 through 8190) external range is reproduced in the processor. Figure 3 shows a program using the MP function to square a number. Figure 4 shows a "times 10" application of this MP function. This configuration uses the second destination 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 this operation for each IN0004 closure.

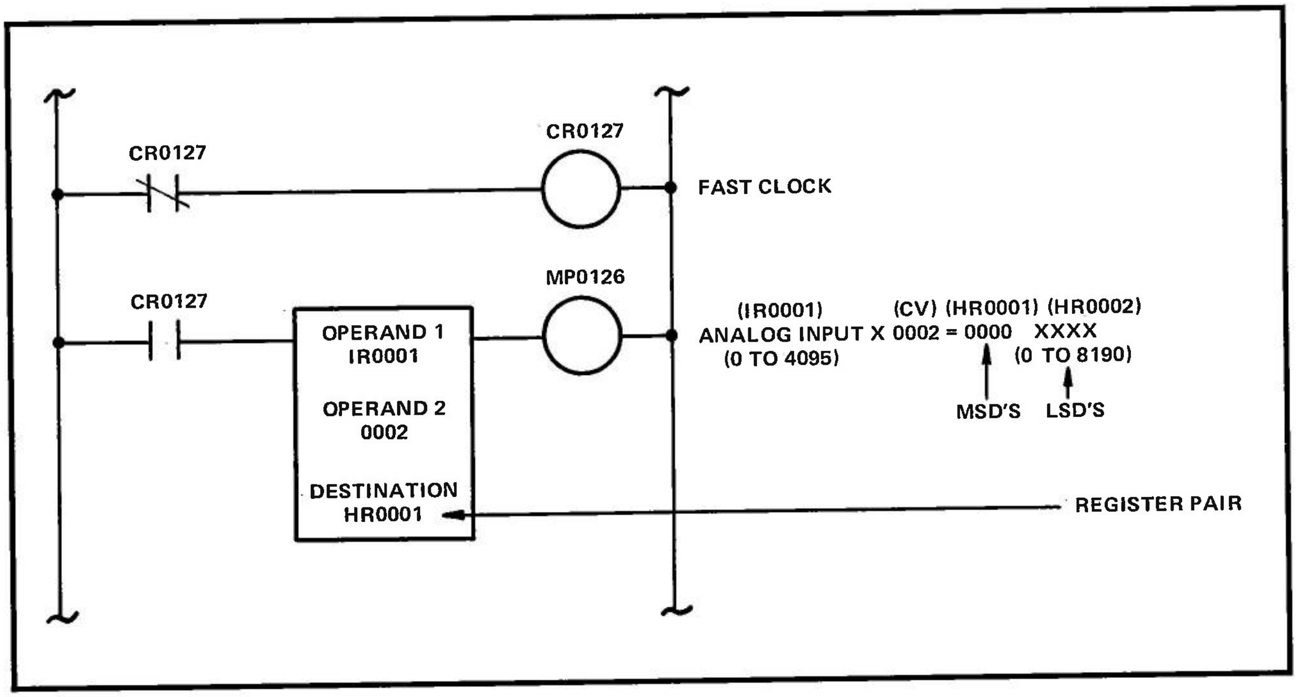


Figure 2. MP Scale Factoring

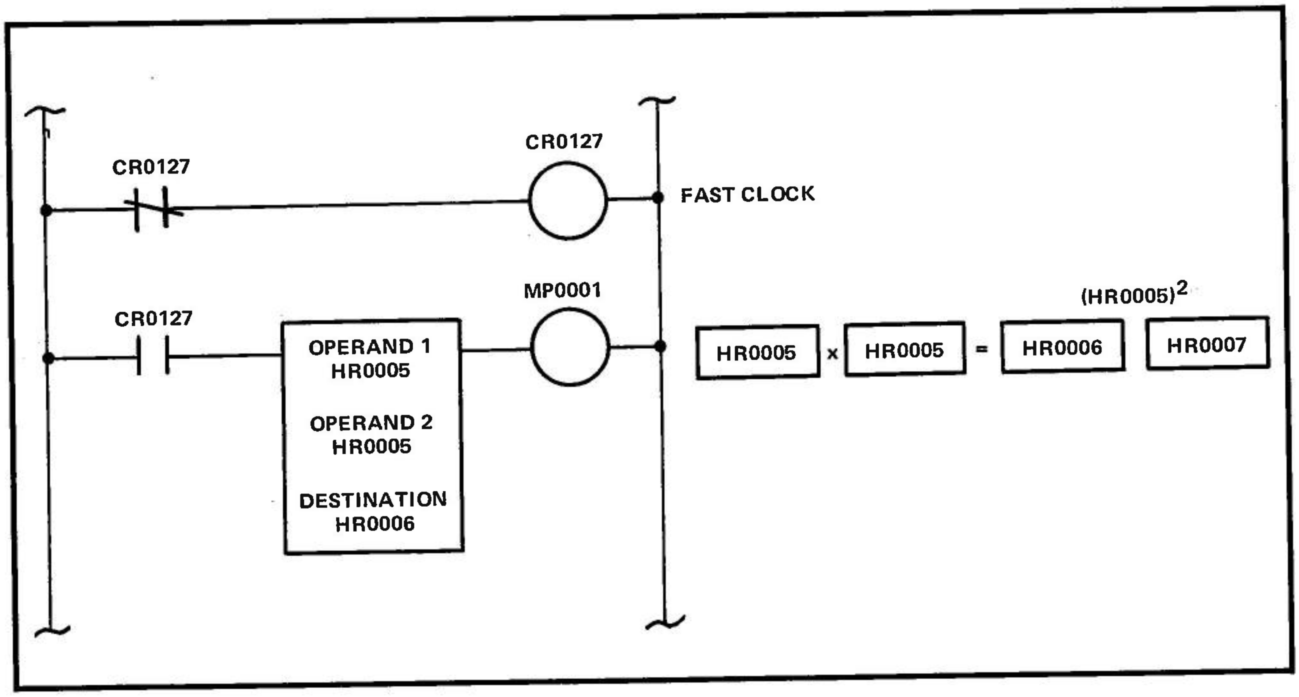


Figure 3. Number Squaring

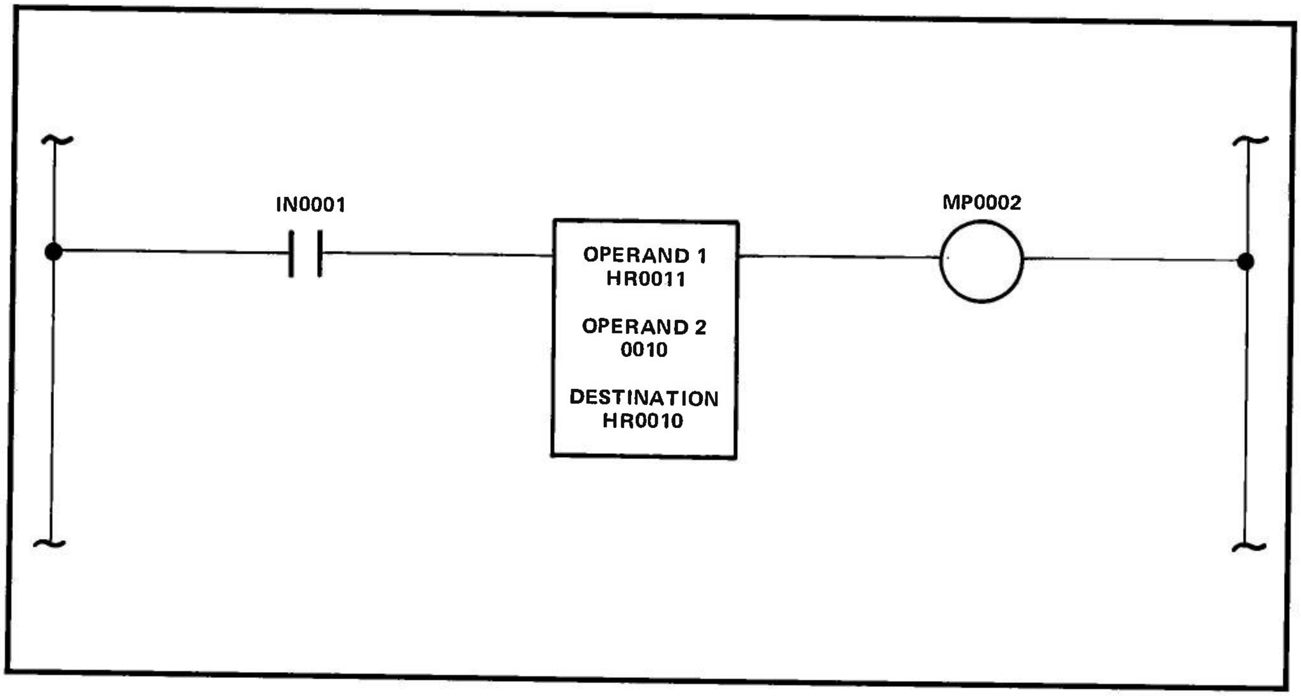


Figure 4. Times 10 Application Example

TABLE 2. TIMES 10 APPLICATIONS EXAMPLE FOR IN0001 CLOSURE

Condition	Contents of HR0010	Contents of HR0011
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

DESCRIPTION

The Master Control Relay (MR) function is also 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. When it does not conduct, the coils function 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.)
- Register data is frozen, unless the register is also affected by a coil that is not MR controlled.

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.

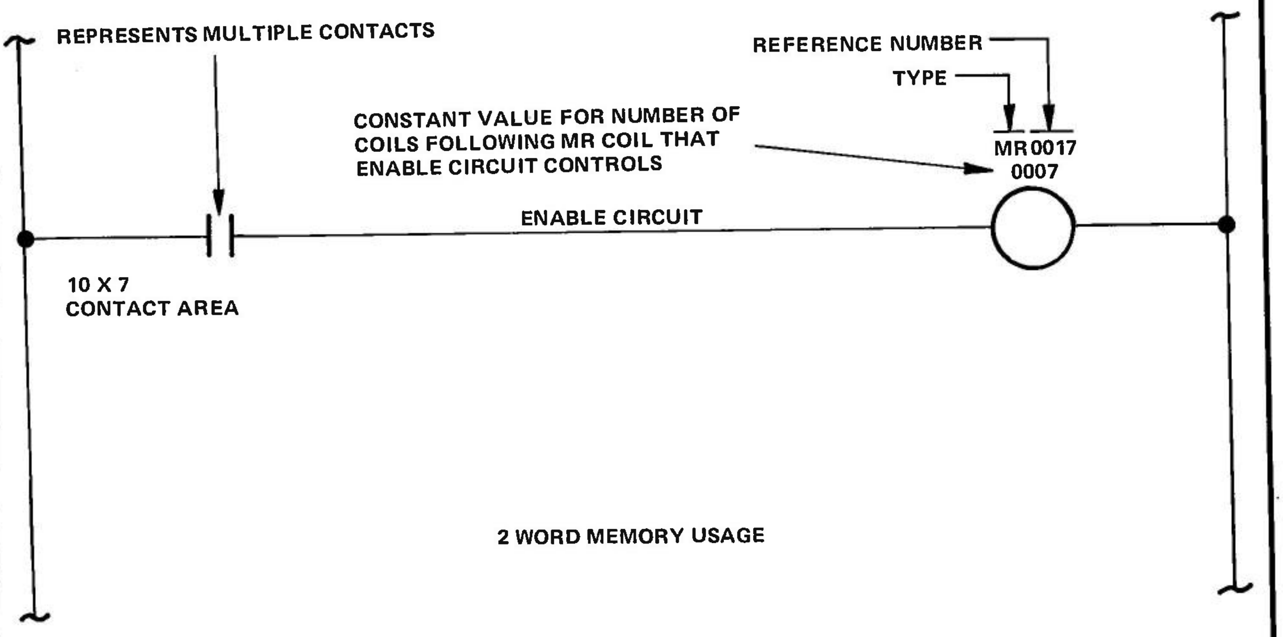

REPRESENTS MULTIPLE CONTACTS
REFERENCE NUMBER
TYPE
CONSTANT VALUE FOR NUMBER OF COILS FOLLOWING MR COIL THAT ENABLE CIRCUIT CONTROLS
MR 0017
0007
ENABLE CIRCUIT
10 X 7
CONTACT AREA
2 WORD MEMORY USAGE

Figure 1. Master Control Relay (MR)

The number of coils controlled by MR 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. When the enable circuit does not conduct, the specified number of coils following the MR coil are de-energized.

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 for to shut down power in an emergency. See Section 7 "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 power failure causes the processor to reset. However, in some cases, where programmed functions must be maintained to ensure system integrity, this reset is undesirable. Figure 2 shows an MR function application which prevents the processor reset.

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.

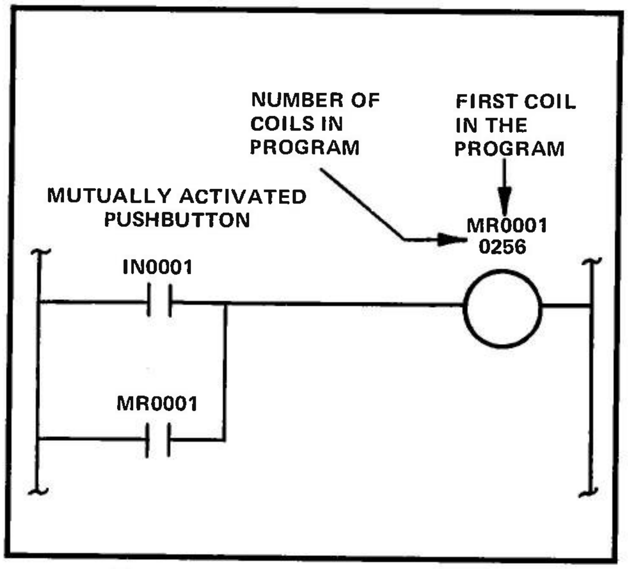

NUMBER OF COILS IN PROGRAM
FIRST COIL IN THE PROGRAM
MUTUALLY ACTIVATED PUSHBUTTON
IN0001
MR0001
0256
MR0001

Figure 2. Manual Reset Circuit

MV — MOVE

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 does 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.

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 moved. (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)

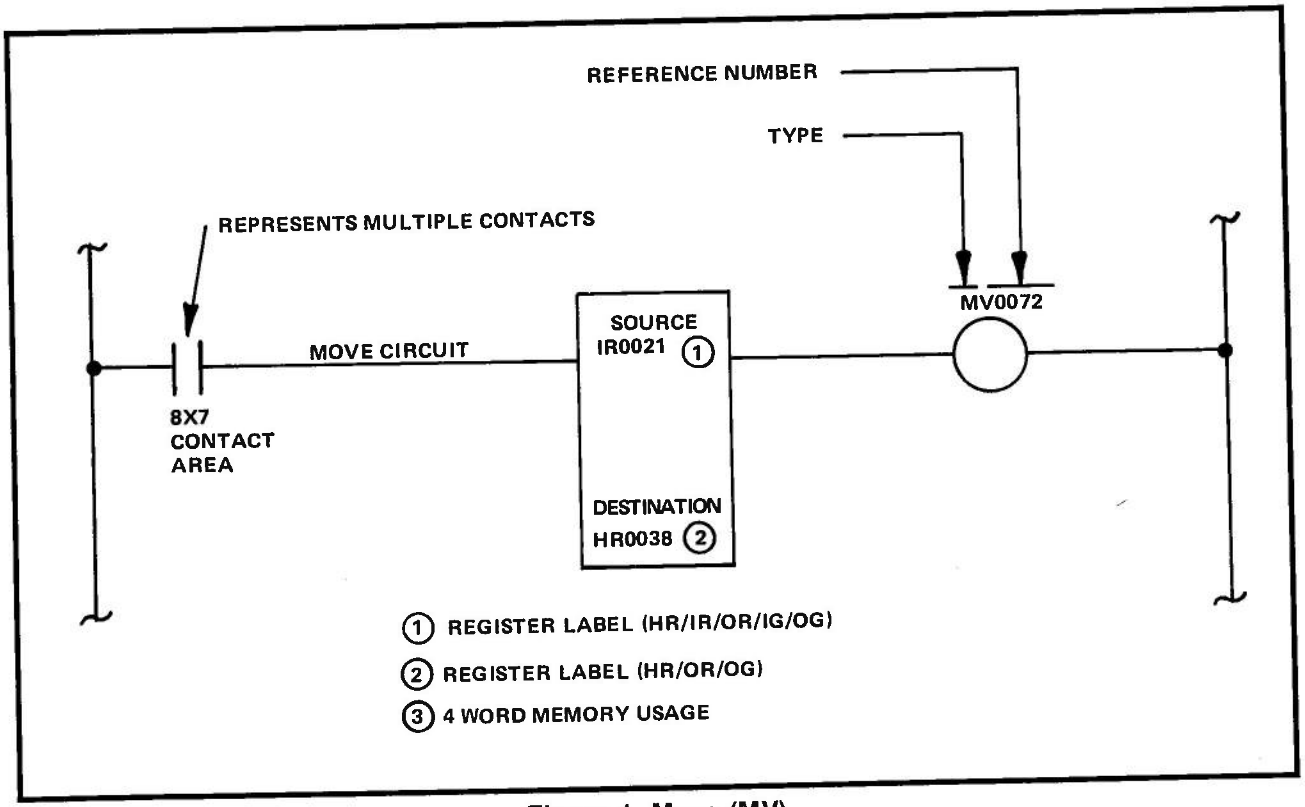


Figure 1. Move (MV)

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.

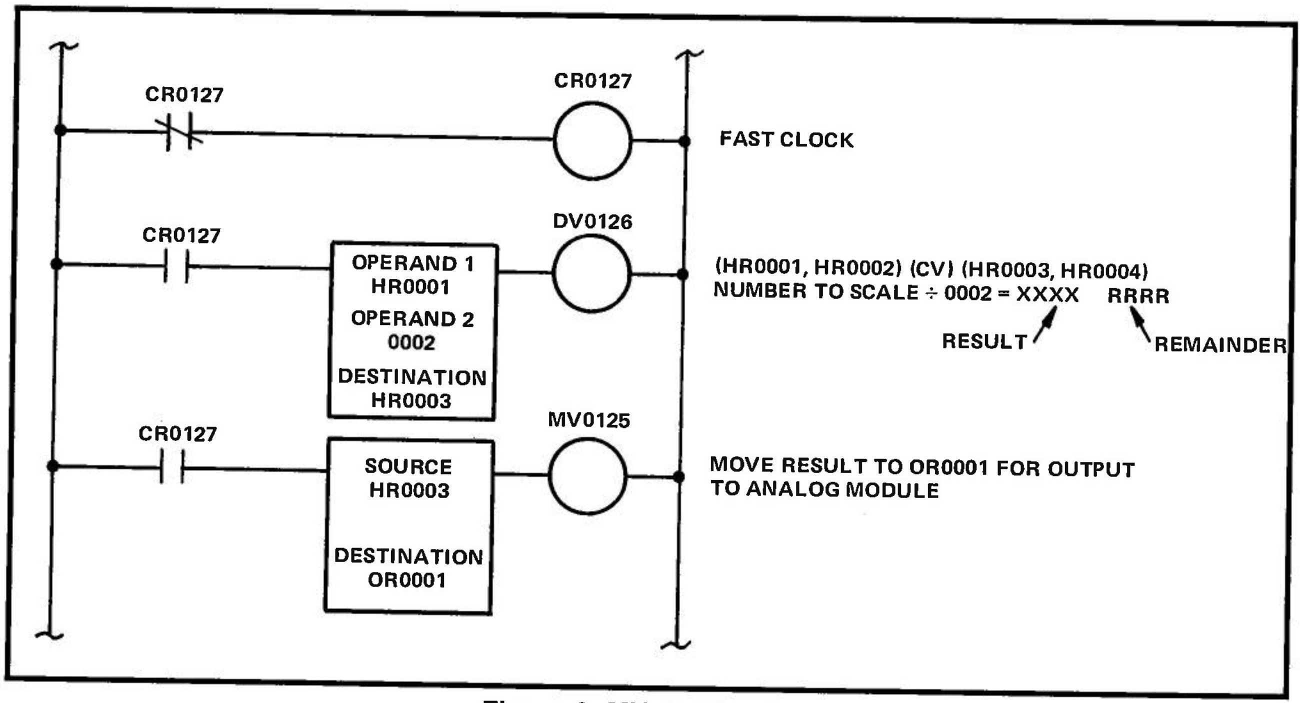


Figure 2. MV Application

NR/NL — N BIT SERIAL SHIFT REGISTERS

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 register 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 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 = Open

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

SPECIFICATIONS

COIL

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

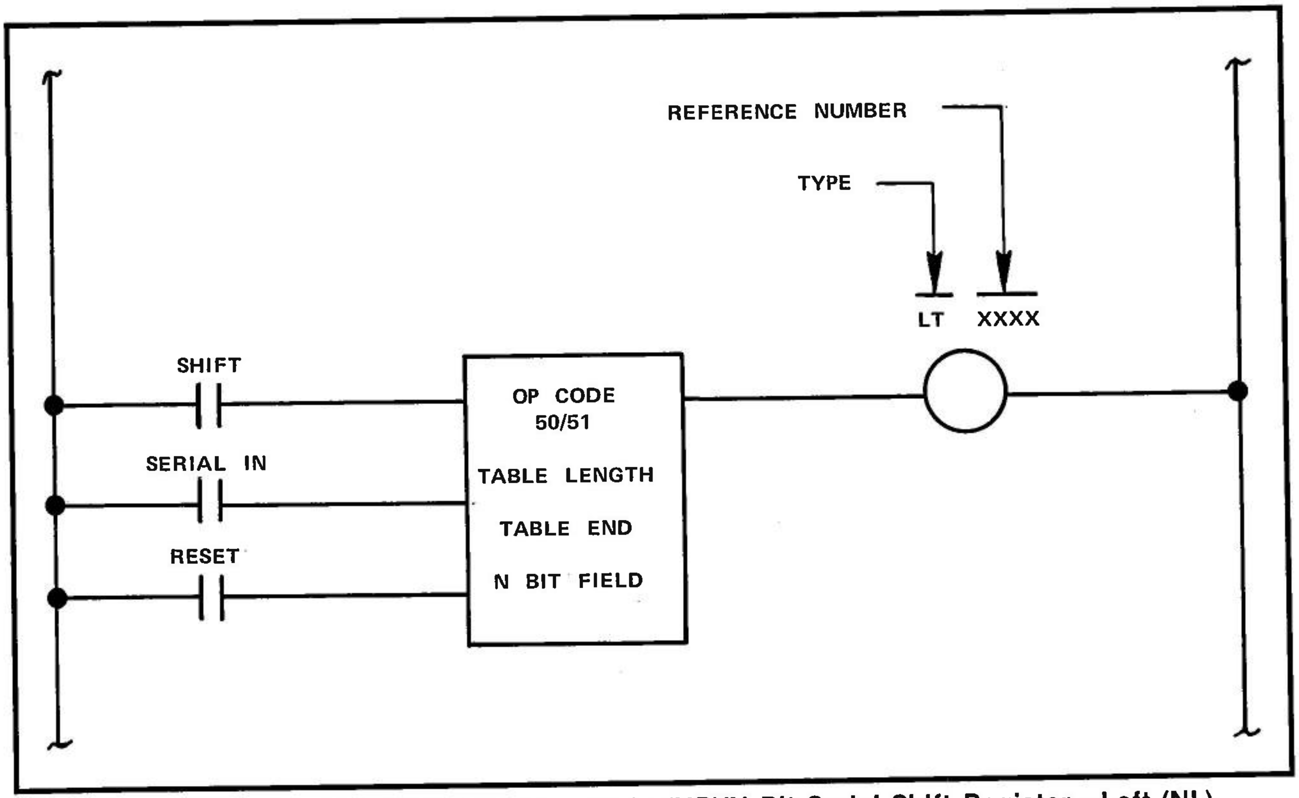


Figure 1. N Bit Serial Shift Register—Right (NR)/N Bit Serial Shift Register—Left (NL)

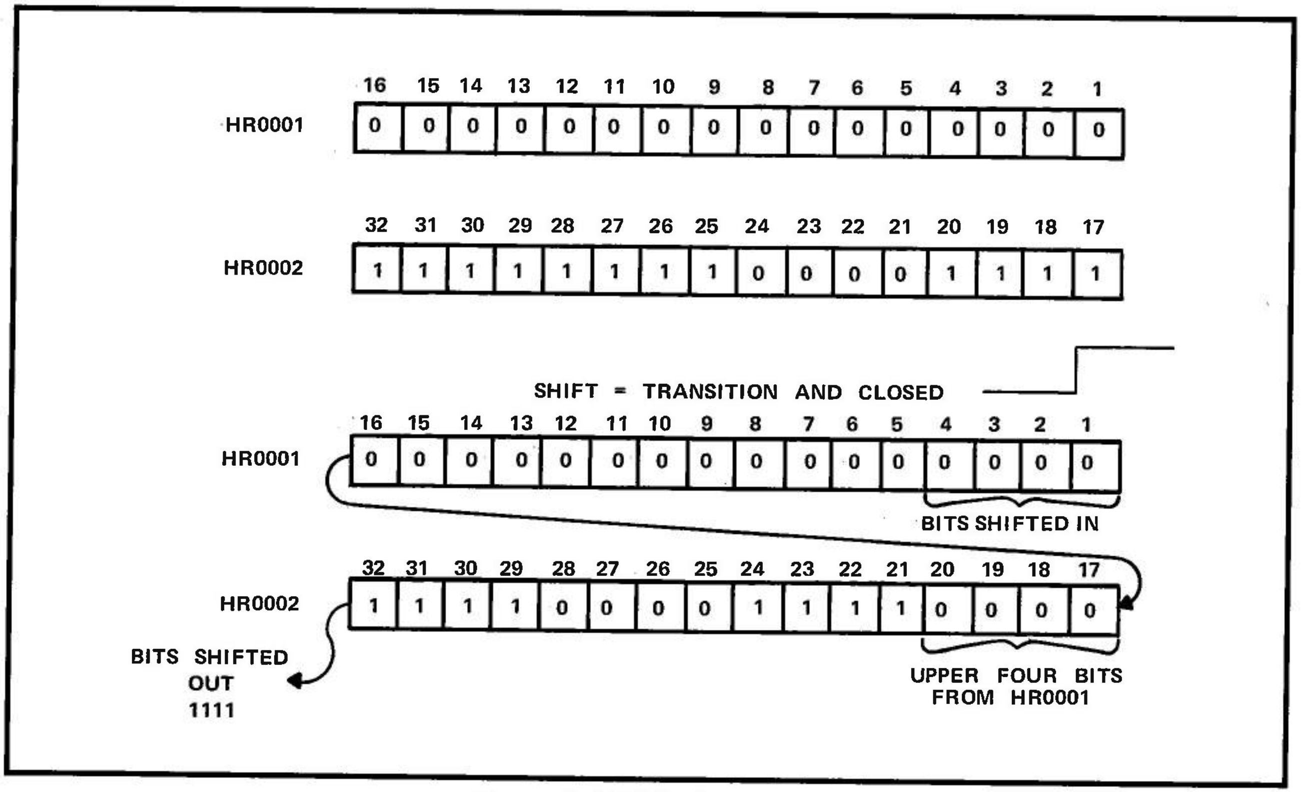


Figure 2. NR/NL Operation

OP CODE 50/51

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

Note

> When software changes allow, LT becomes NR or NL.

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 1. TABLE LENGTH/TABLE END LIMITS

Type	Limit
HR	⩽ 1792
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)

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.

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)

NR/NL TRUTH TABLE

See Table 2.

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.

TABLE 2. NR/NL TRUTH TABLE

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

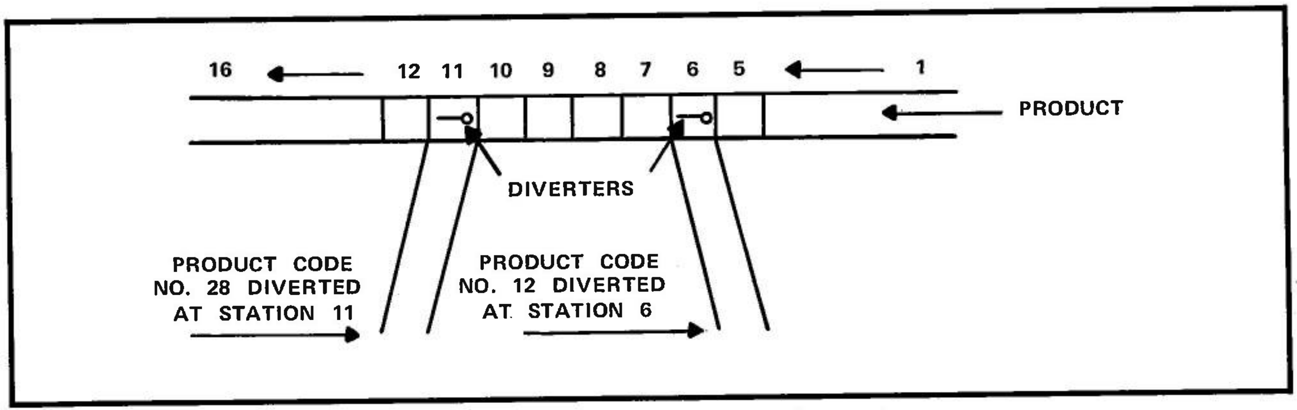
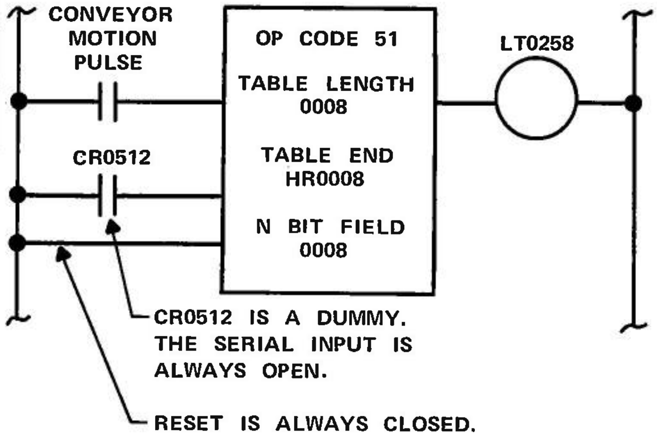
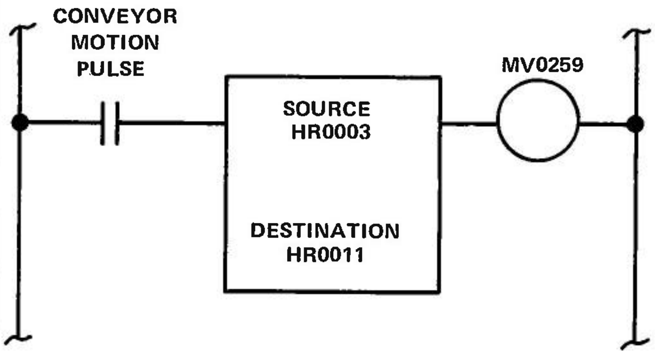

16
12
11
10
9
8
7
6
5
1
PRODUCT
DIVERTERS
PRODUCT CODE NO. 28 DIVERTED AT STATION 11
PRODUCT CODE NO. 12 DIVERTED AT STATION 6

Figure 3. Sorting and Diverting Operation



LT0258 IS AN N BIT SERIAL SHIFT LEFT OF 8 PLACES (EIGHT BINARY BITS) FOR EACH CYCLE OF THE CONVEYOR MOTION PULSE CIRCUIT. HR0001 TO HR0008 CONTAIN THE 16 CONVEYOR STATIONS, TWO PER REGISTER.

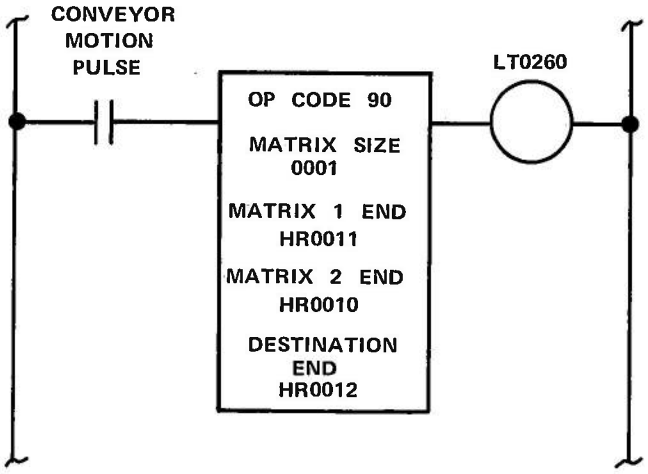
	BITS 1 THROUGH 8	BITS 9 THROUGH 16
HR0001	STATION 1	STATION 2
HR0002	STATION 3	STATION 4
HR0003	STATION 5	STATION 6
HR0004	STATION 7	STATION 8
HR0005	STATION 9	STATION 10
HR0006	STATION 11	STATION 12
HR0007	STATION 13	STATION 14
HR0008	STATION 15	STATION 16



MV0259 DUPLICATES THE CONTENTS OF HR0003.

	BITS 1 THROUGH 8	BITS 9 THROUGH 16
HR0003	STATION 5	STATION 6
	IN	
HR0011	STATION 5	STATION 6

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



LT0260 IS AN AND MATRIX FUNCTION USED TO MASK STATION 5 OUT OF A COMPARISON FOR THE VALUE OF STATION 6.

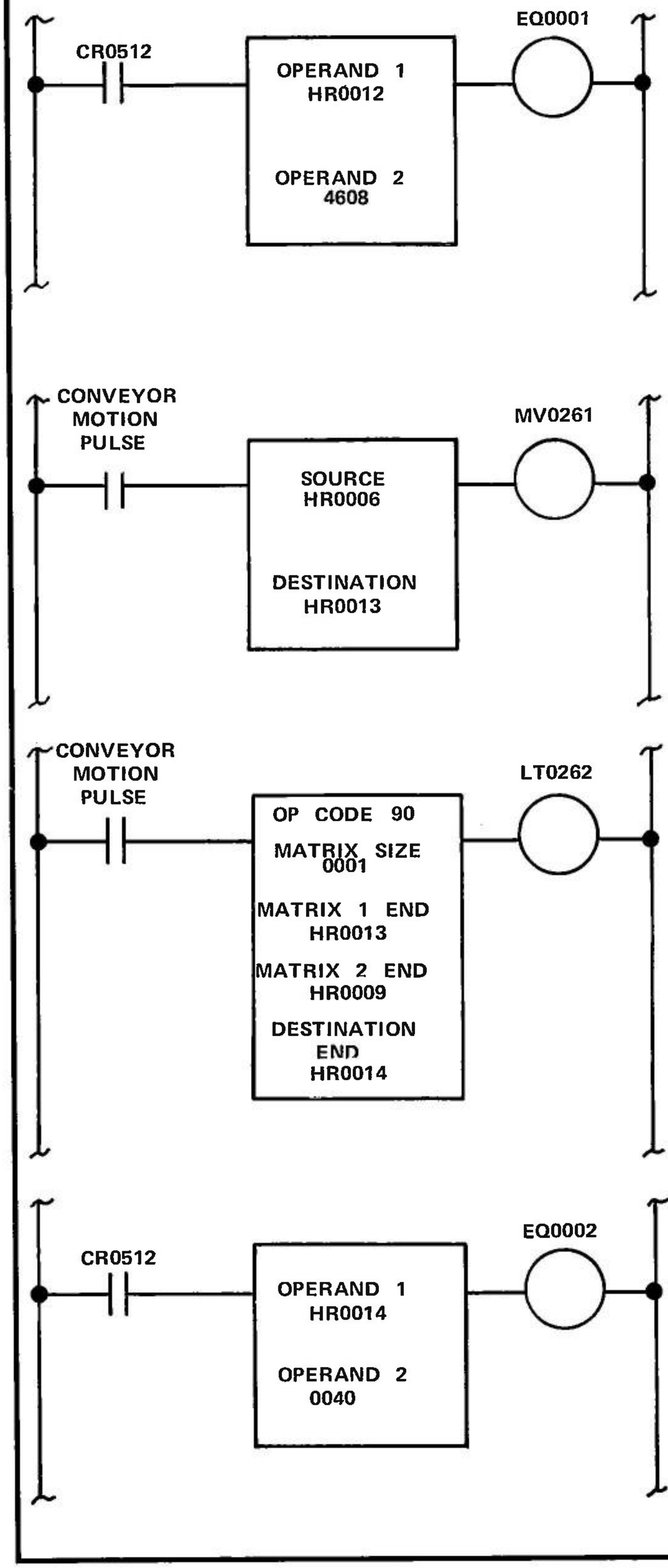
BIT 16 ↓ BIT 1 ↓

HR0010 = 1111111100000000_2

HR0011 = $XXXXXXXXXXXXXXXX_2$

HR0012 = $XXXXXXXX00000000_2$

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

00010010	000000002
STATION 6	

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.

MV0261 DUPLICATES THE CONTENTS OF HR0006.

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

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

LT0262 IS AN AND MATRIX FUNCTION 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 THROUGH 8, IS COMPARED TO A CONSTANT OF 40. 40 IS THE EQUIVALENT OF

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 STATION 11.

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

OM — OR MATRIX

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. OM function symbology 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.

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 bit in all three matrices.

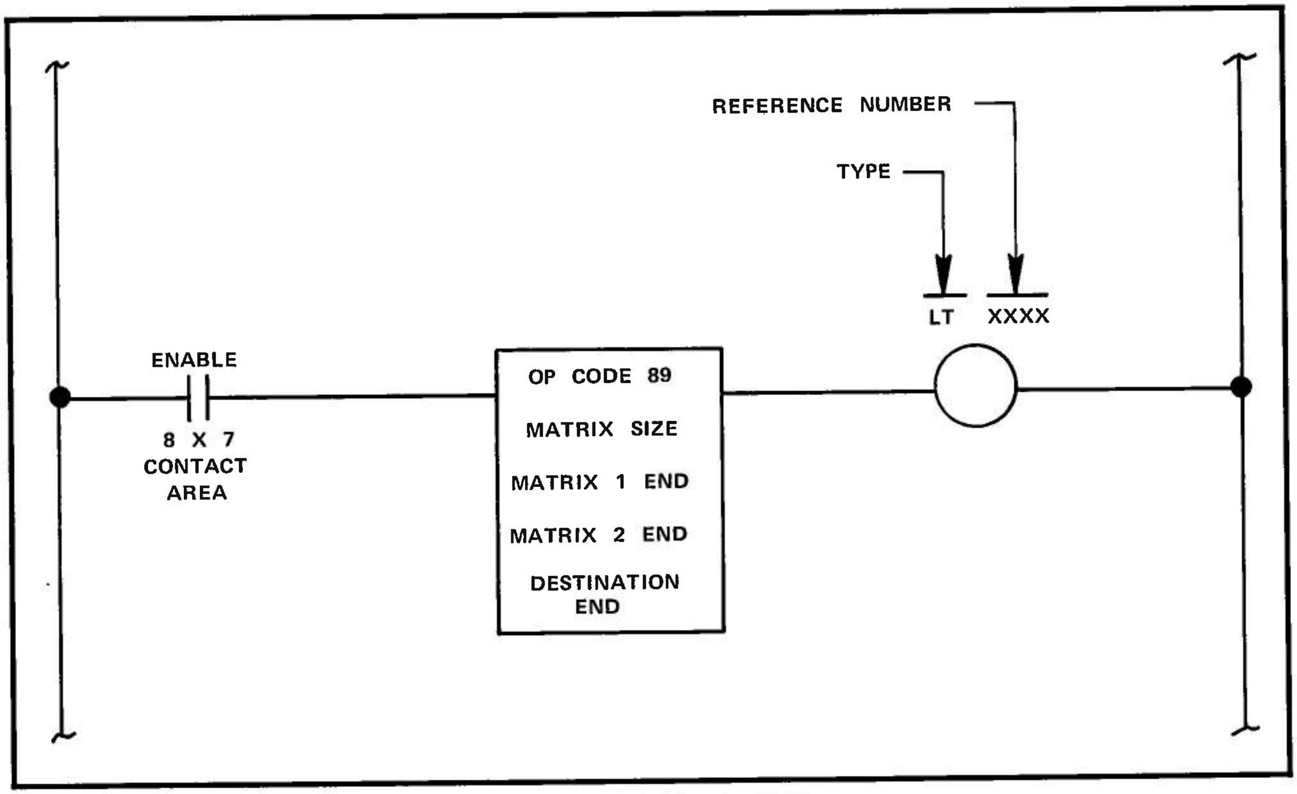


Figure 1. OR Matrix (OM)

SPECIFICATIONS

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.

OP CODE 89

The Op Code defines the Literal (LT) as an OM function.

Note

> When software changes allow, LT becomes OM.

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	Limit
HR	⩽ 1792
IR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
IG	⩽ 16 (PC-700) ⩽ 8 (PC-900 A/B)
OG	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)

DESTINATION END

The destination end defines the 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 DESTINATION END REGISTER

Type	Limit
HR	⩽ 1792
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
OG	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)

OM TRUTH TABLE

See Table 4.

Note

> The highest number holding register is limited by and dependent upon memory 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.
1	The coil is energized if the results are not zero.

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 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, LT0001 is energized as long as IN0001 is closed.

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

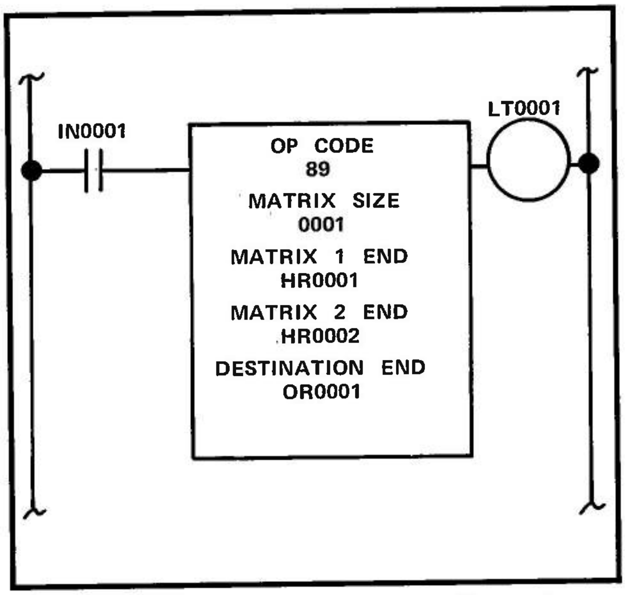


Figure 3. OM Ladder Diagram Example

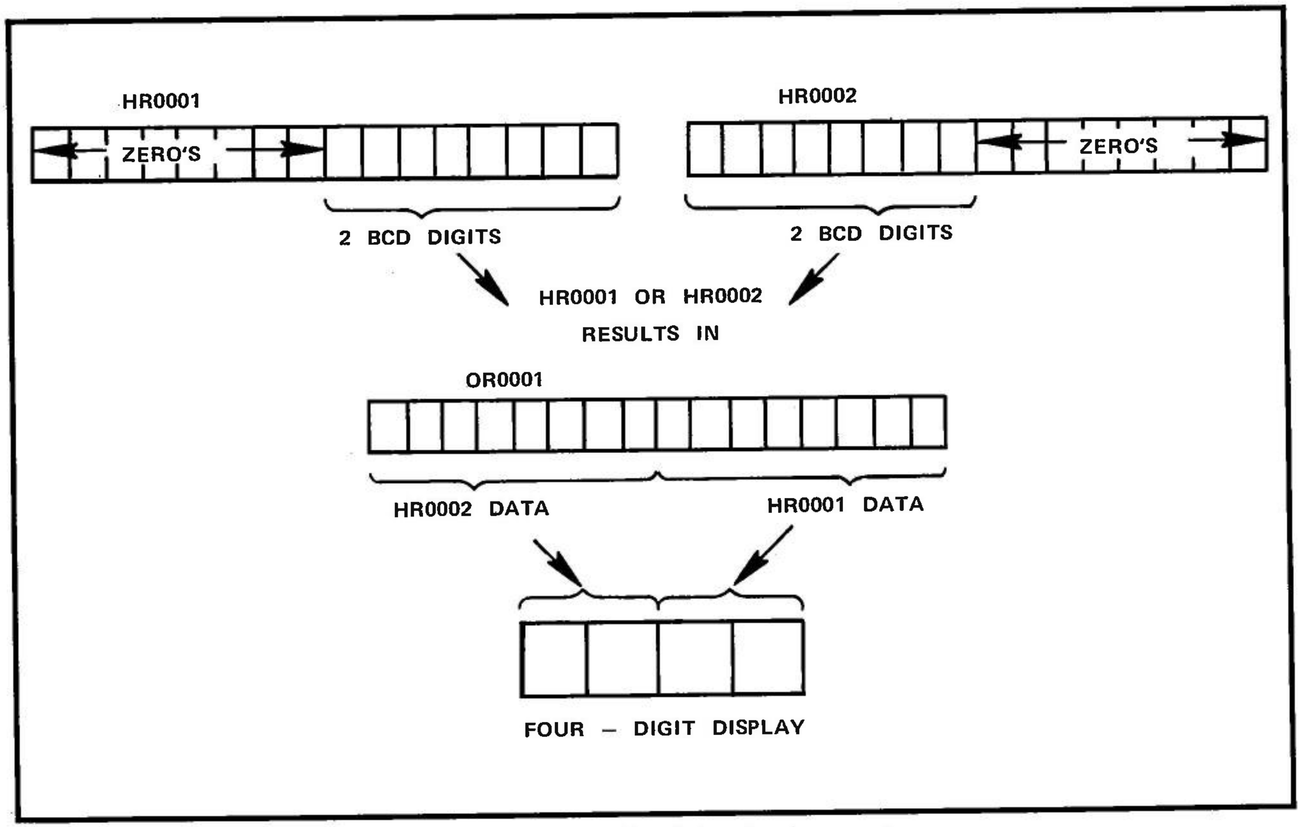


Figure 2. Output Register Assignments

MATRIX 1

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

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

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	0	0	0	0	0	0	0

MATRIX 1 END

MATRIX 2

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

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

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

MATRIX 2 END

DESTINATION MATRIX

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

OG0002	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

OG0003	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

DESTINATION END

Figure 4. A Pair of Matrices OR'ed

OT/CT — OPEN TABLE/CLOSE TABLE

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 below the pointed location are moved down 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 below the pointed location is moved up one location. The last register in the table is duplicated when the data is moved, as shown in Figure 3.

SPECIFICATIONS

OP CODE 92/93

The Op Code defines 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.

Note

> When software changes allow, LT becomes OT or CT.

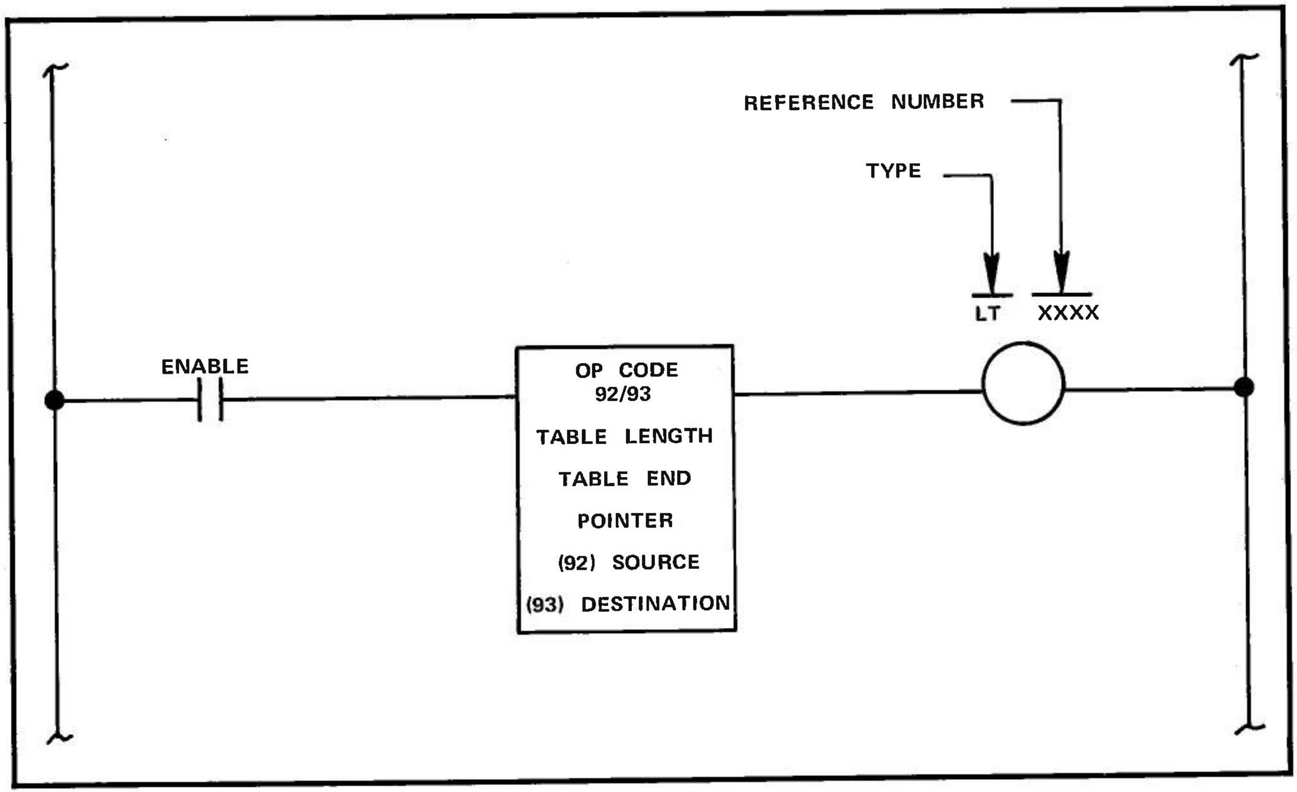


Figure 1. Open Table (OT)/Closed Table (CT)

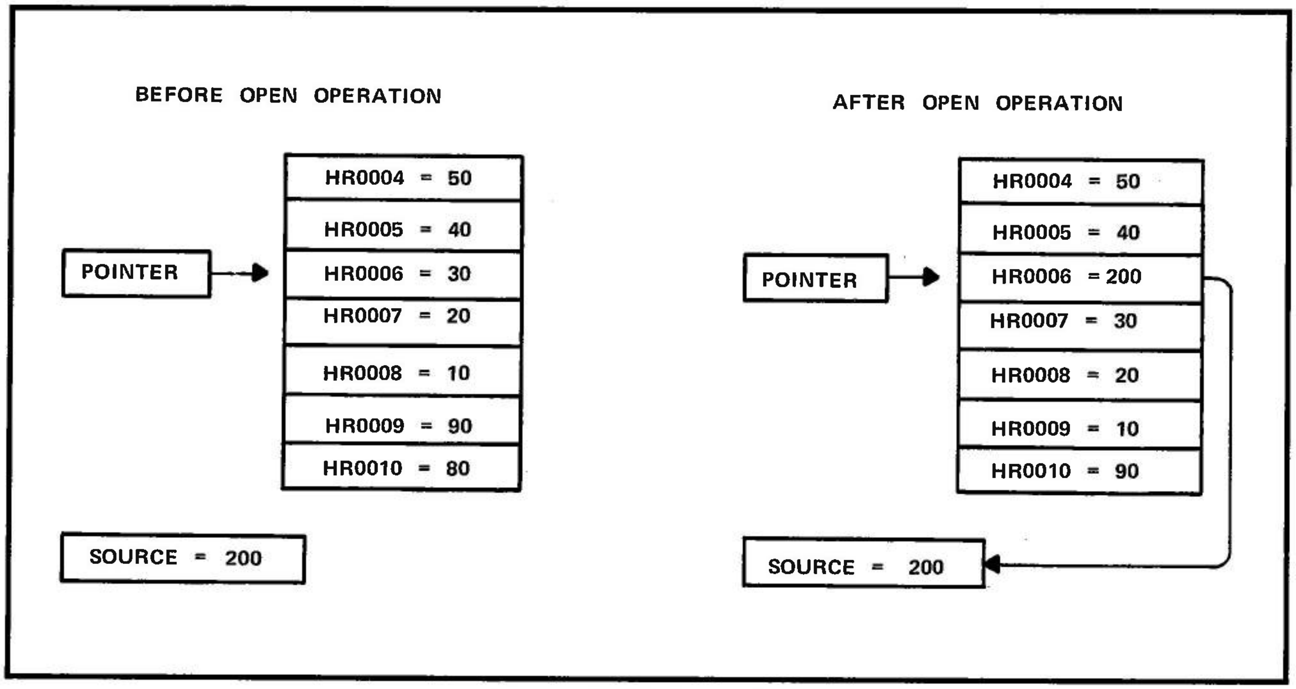


Figure 2. OT Operation

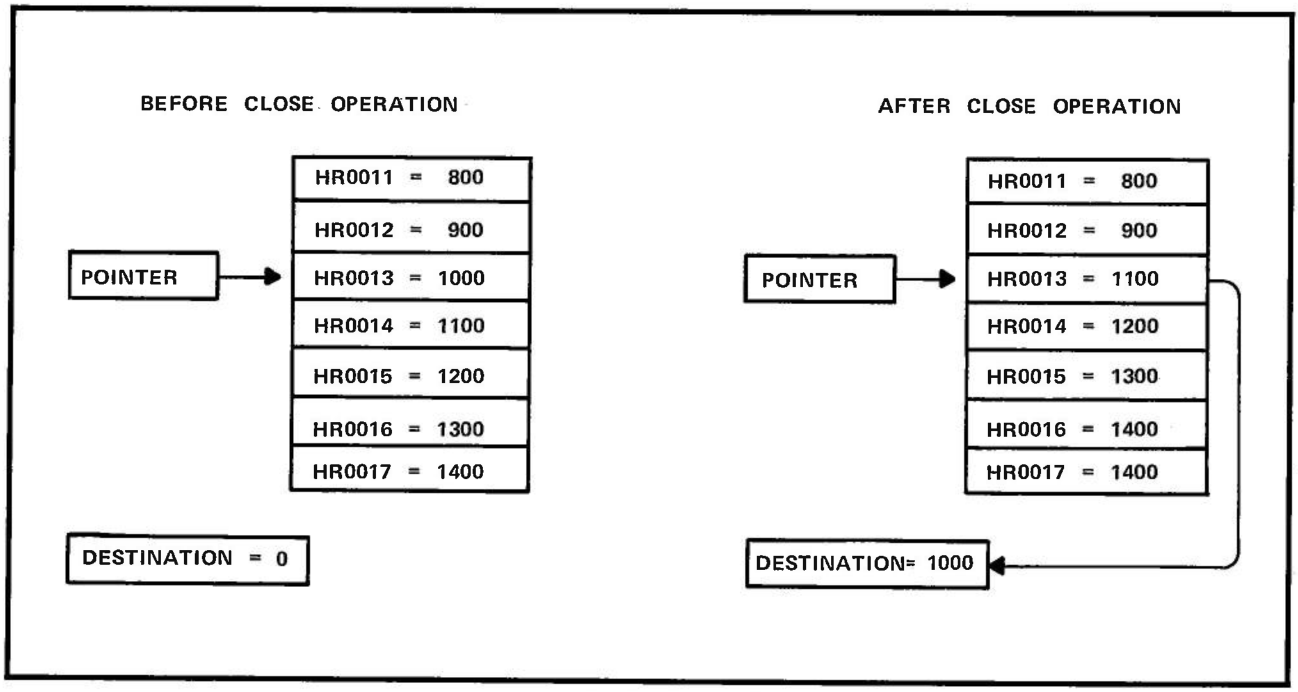


Figure 3. CT Operation

TABLE LENGTH

The table length is a constant value from 1 through 256 that determines the length of the table being opened or closed.

TABLE END

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

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)

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

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 down 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 below the pointed location are moved up one place in the table. Data in the last register in the table moves up 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.

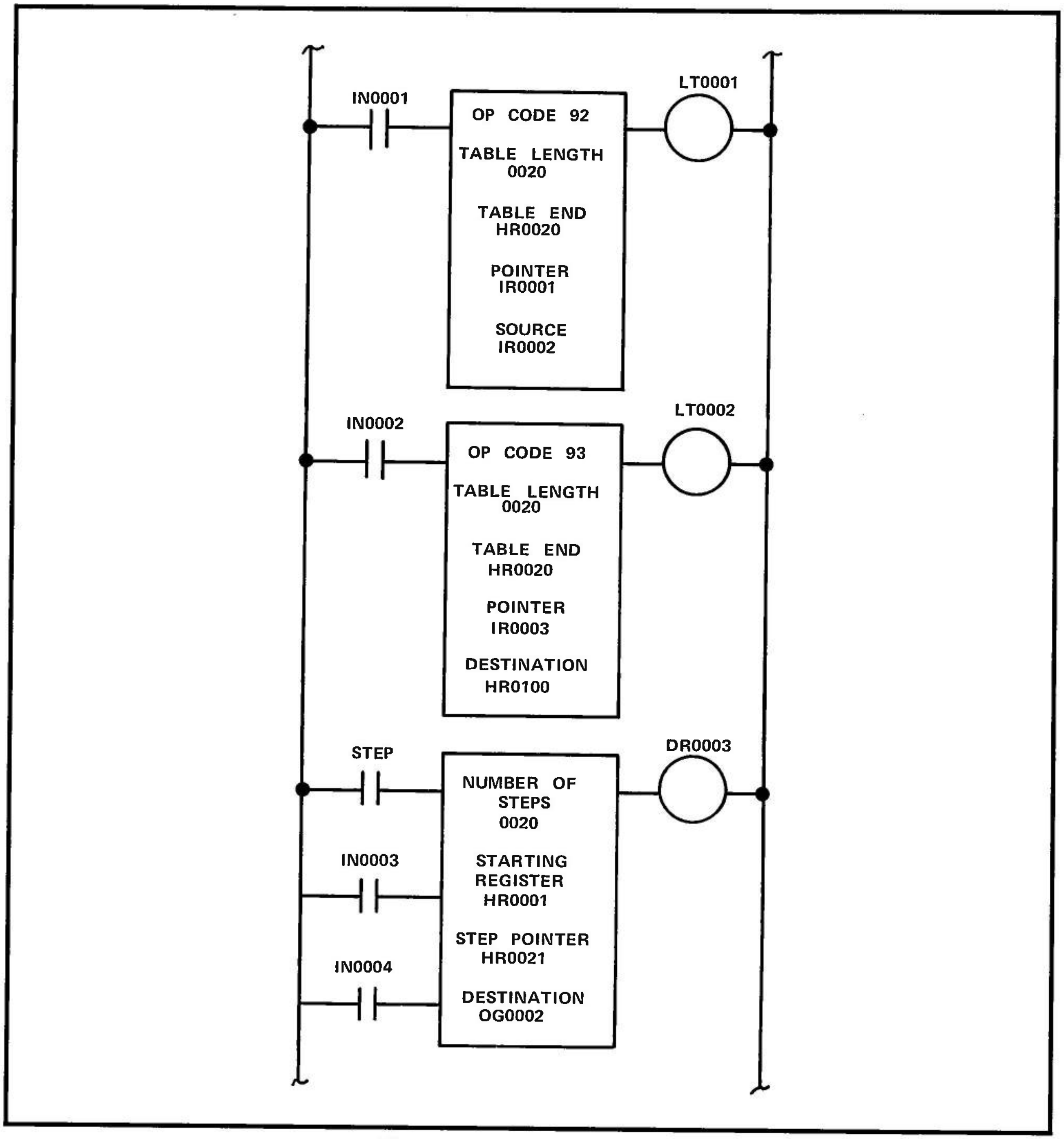


Figure 4. OT/CT Application

RP — RESTORE PROGRAM COUNTER

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.

SPECIFICATIONS

OP CODE 13

The Op Code defines the Literal (LT) as the RP function.

Note

When software changes allow, LT becomes RP.

OPERAND 1

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.

RP TRUTH TABLE

See Table 1.

APPLICATIONS

Applications for the SP and RP functions are described in this section.

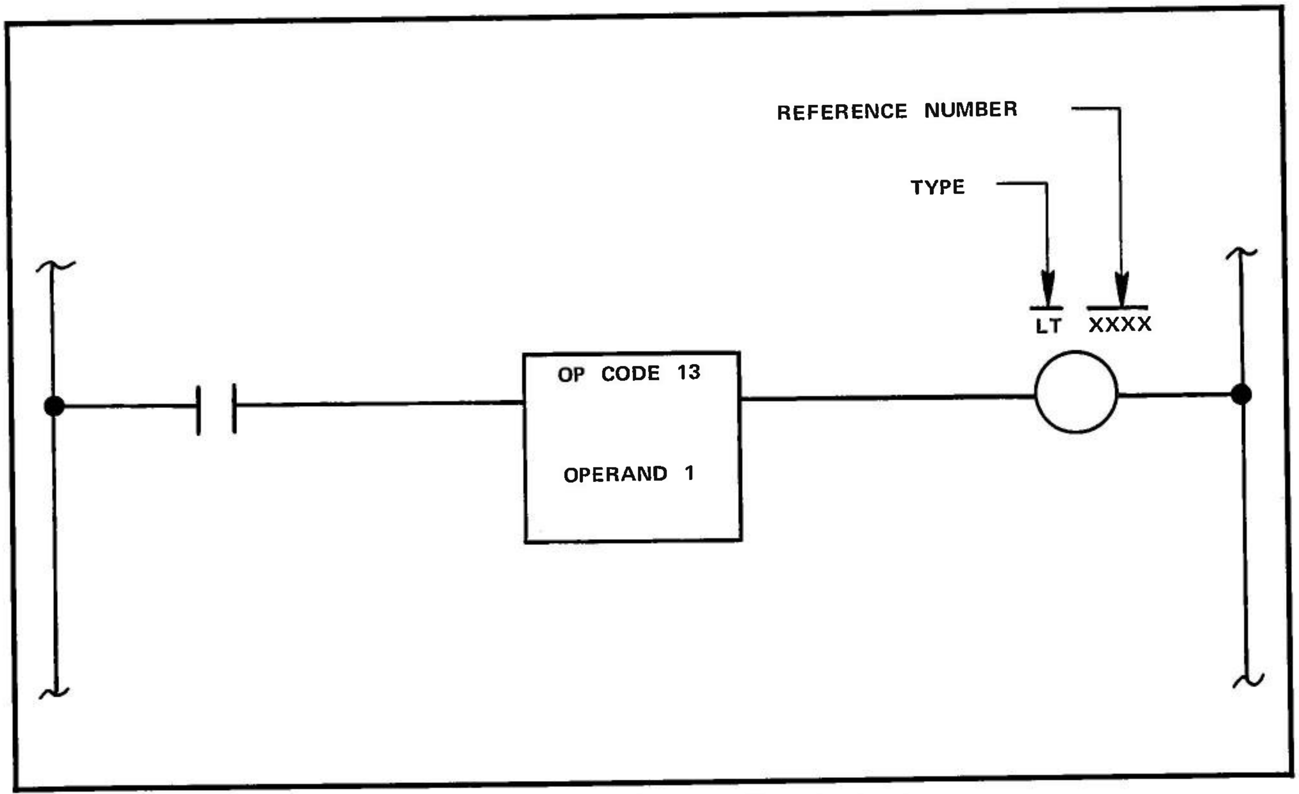


Figure 1. Restore Program Counter (RP)

TABLE 1. RP TRUTH TABLE

Enable	Result
0	The coil is de-energized. Ladder execution is unaffected.
1	Ladder execution starts at the program location stored in the register specified by Operand 1 unless that value does not point to ladder memory or point to the first contact of a ladder rung. If the number is invalid, no data is moved to the program counter and the coil is energized, indicating an error condition.

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 described in the RP module.

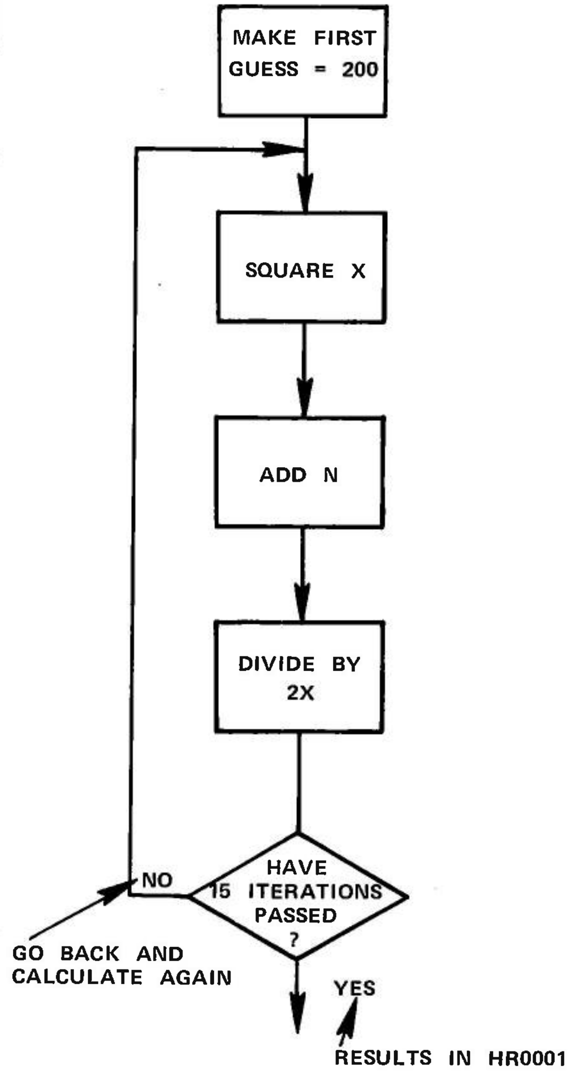


Figure 2. Square Root Flowchart

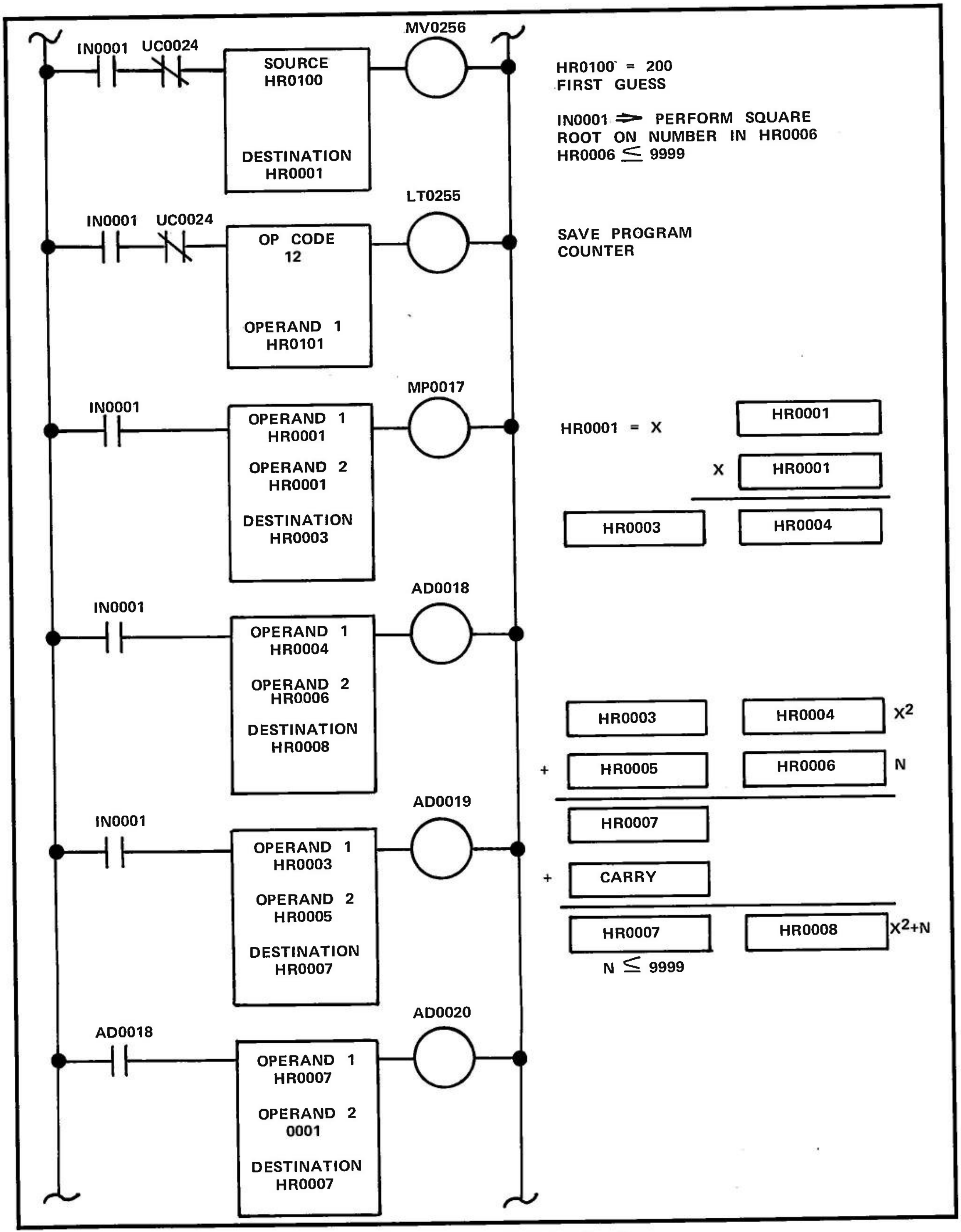


Figure 3a. Square Root Application

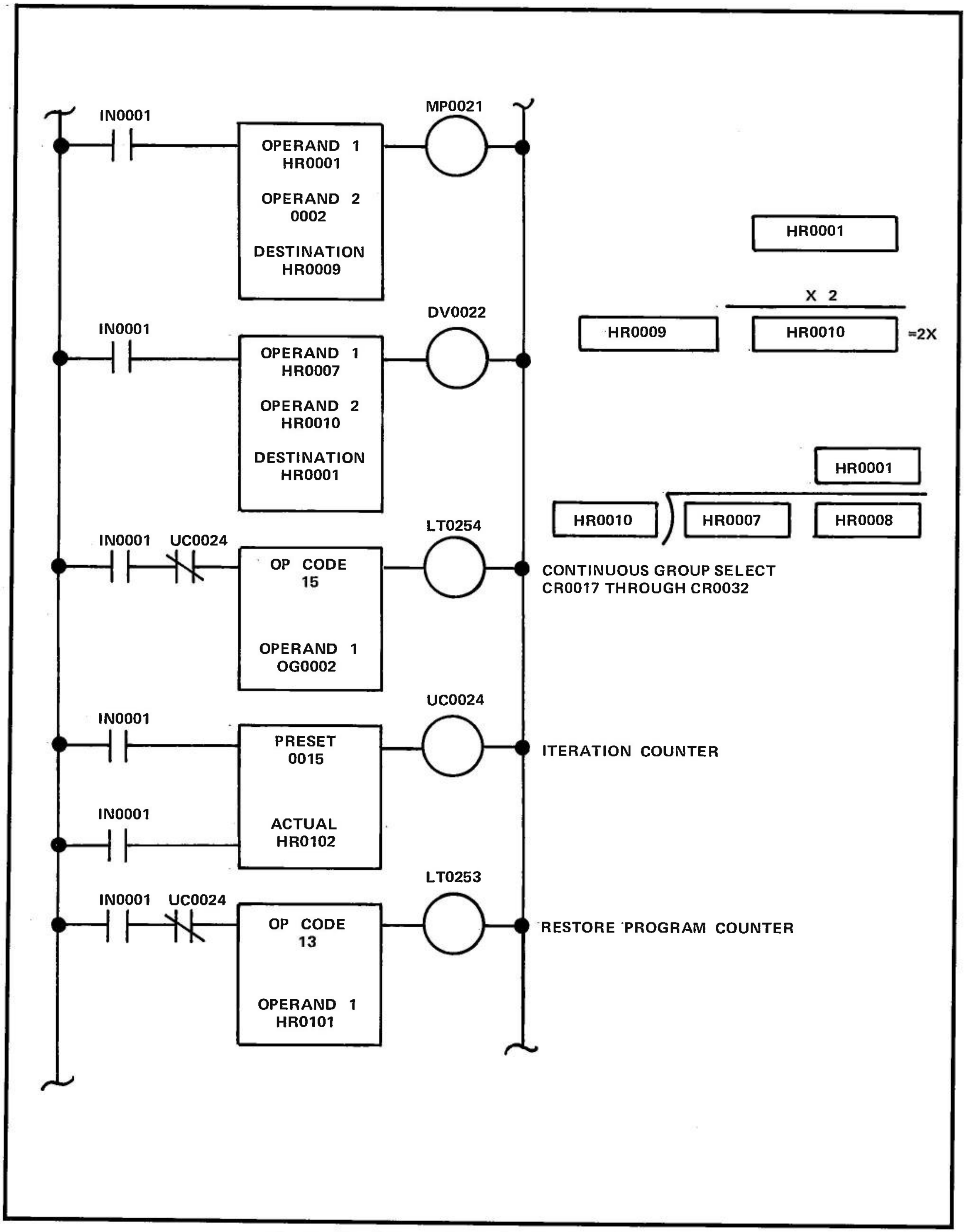


Figure 3b. Square Root Application (Cont'd)

RT — REGISTER-TO-TABLE MOVE

DESCRIPTION

The Register-to-Table Move (RT) function allows a source register to fill successive locations in a predefined table of registers. The relative location is determined by the value of the contents at a pointer location. RT function symbology is shown in Figure 1.

Figure 2 depicts the operation of the RT function in a simplified manner. The pointer operates under the control of three circuits; the step circuit causes the pointer to increment; the reset circuit holds the pointer to zero; and the enable circuit allows the pointer to increment and the data to transfer.

SPECIFICATIONS

COIL

The coil energizes when the pointer equals the table length minus one, and de-energizes when the pointer is greater or less than the table length minus one.

OP CODE 80

The Op Code defines the Literal (LT) as an RT function.

Note

When software changes allow, LT becomes RT.

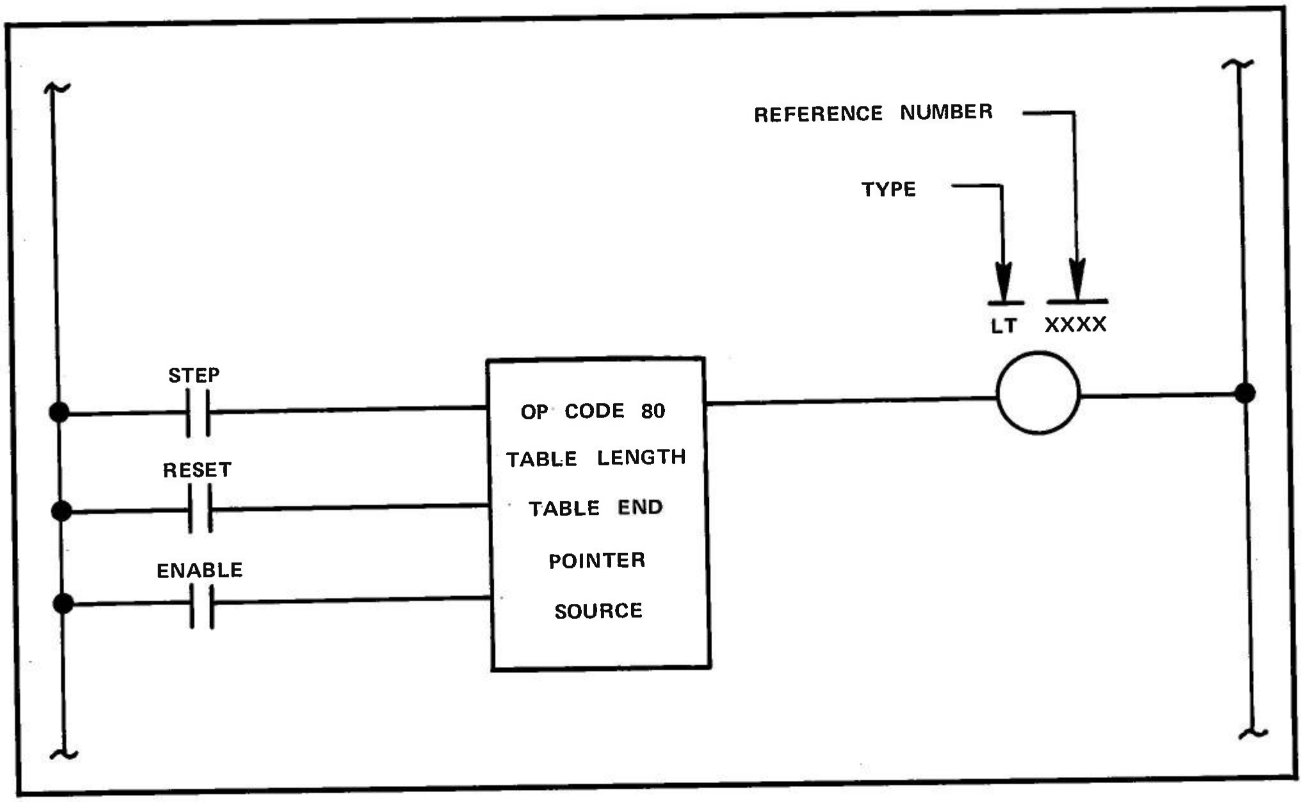
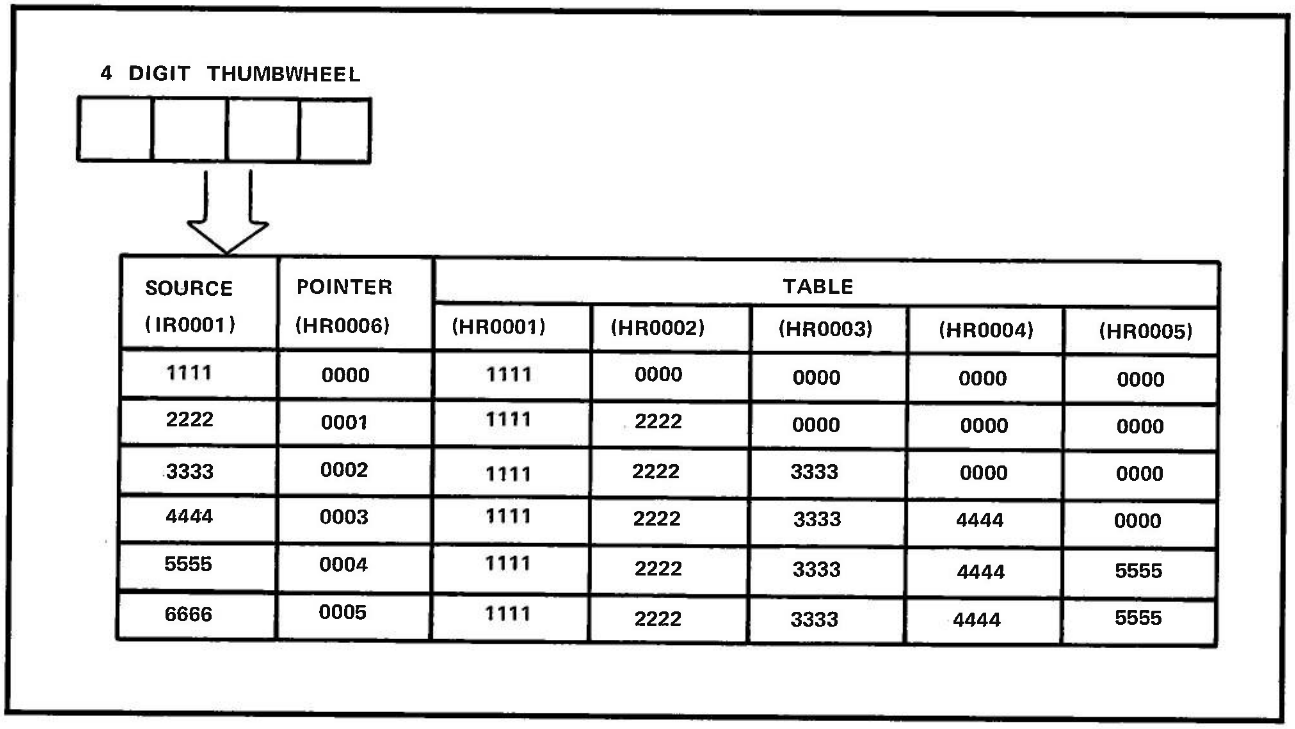


Figure 1. Register-to-Table Move (RT)



SOURCE	POINTER	TABLE				
(IR0001)	(HR0006)	(HR0001)	(HR0002)	(HR0003)	(HR0004)	(HR0005)
1111	0000	1111	0000	0000	0000	0000
2222	0001	1111	2222	0000	0000	0000
3333	0002	1111	2222	3333	0000	0000
4444	0003	1111	2222	3333	4444	0000
5555	0004	1111	2222	3333	4444	5555
6666	0005	1111	2222	3333	4444	5555

Figure 2. RT Example

TABLE LENGTH

The table length is a constant value that defines the number of registers in the table. This value ranges from 1 through 256, and is subject to the limits listed in Table 1.

TABLE 1. TABLE LENGTH/TABLE END LIMITS

Type	Limit
HR	⩽ 1792
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
OG	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)

TABLE END

The table end defines the type and number of the last register in the destination table, and is subject to the limits listed in Table 1.

Note

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

POINTER

The pointer contains the current table location into which data transfers. This location is a specified register:

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

SOURCE

The source is the location from which data transfers. It is a specified register or group:

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

RT TRUTH TABLE

See Table 2.

TABLE 2. RT TRUTH TABLE

Step	Reset	Enable	Result
0, ↑ or 1	0	0	The pointer is set to zero. No data transfer takes place.
0, ↑ or 1	0	1	The pointer is set to zero. The source contents transfer to the table start location.
↑	1	0	The pointer is incremented. No data transfer takes place.
0	1	1	The pointer value depends on previous operations. Data is transferred continuously.
↑	1	1	The pointer is incremented by 1. Data is moved from the source to the table, provided the pointer is ⩽ the table end. If the pointer is > the table end, no move occurs, and the pointer is zeroed. Data is transferred.
1	1	1	Data is transferred.

APPLICATIONS

The RT function is ideally suited for loading step data into the Drum Controller (DR) function. In the circuit in Figure 3, IN0001 is a momentary pushbutton, ENTER DATA; IN0002 is LOAD DATA and must be turned on to load information into the table; and IN0003 is a RELOAD DRUM CONTROLLER momentary pushbutton. An operator can enter new data into the Drum Controller table by entering the desired data in a thumbwheel set. This data is converted to binary information when the ENTER DATA pushbutton is pressed. The binary data is placed in HR0030, which is also the SOURCE register for the Table to Register Move, LT0130. If IN0002 is closed, pressing IN0001 will cause the data currently in HR0030 to be placed in HR0031, the first location in the table being loaded. When IN0001 is released, the POINTER will increment by 1. New data is loaded into the thumbwheel set and the procedure repeated, this time loading the data into HR0032. When the table is fully loaded, the operator presses IN0003, causing the full table of data to be placed in the Drum Controller table, HR0001 through HR0010, by the Block Transfer LT0133.

Additional contacts could have been inserted in the enable circuit of LT0133, the Block Transfer, to prevent the new table of data from being moved into the Drum Controller unless the machine being cycled is stopped or is between cycles.

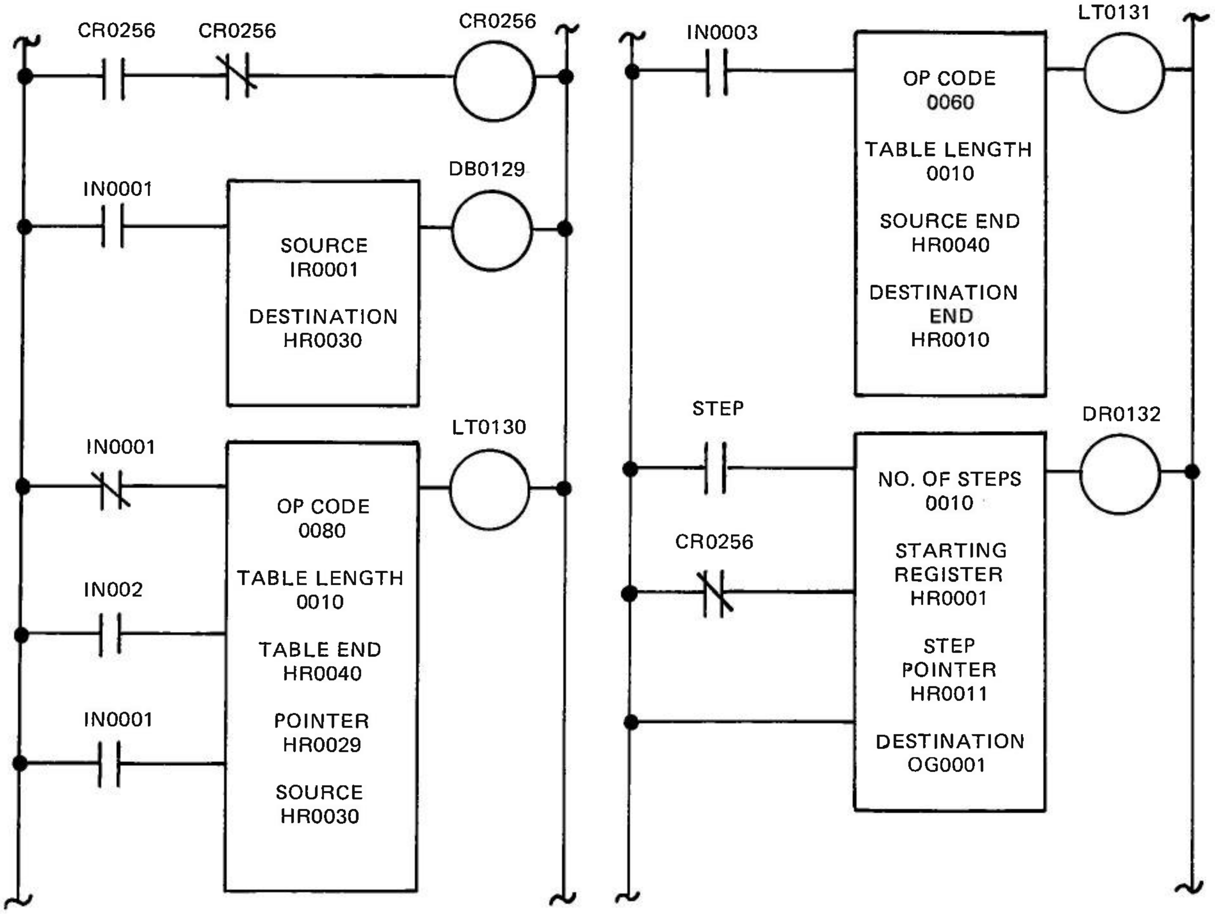


Figure 3. Drum Controller Table Loading Circuit

RW — RESET WATCHDOG TIMER

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.

RW function symbology is shown in Figure 1.

SPECIFICATIONS

OP CODE 14

The Op Code defines the Literal (LT) as the RW function.

Note

> When software changes allow, LT becomes RW.

OPERAND 1

Operand 1 is a holding register that accumulates the number of 10 msec intervals that have lapsed while the RW function is active. A maximum of 256 intervals is allowed.

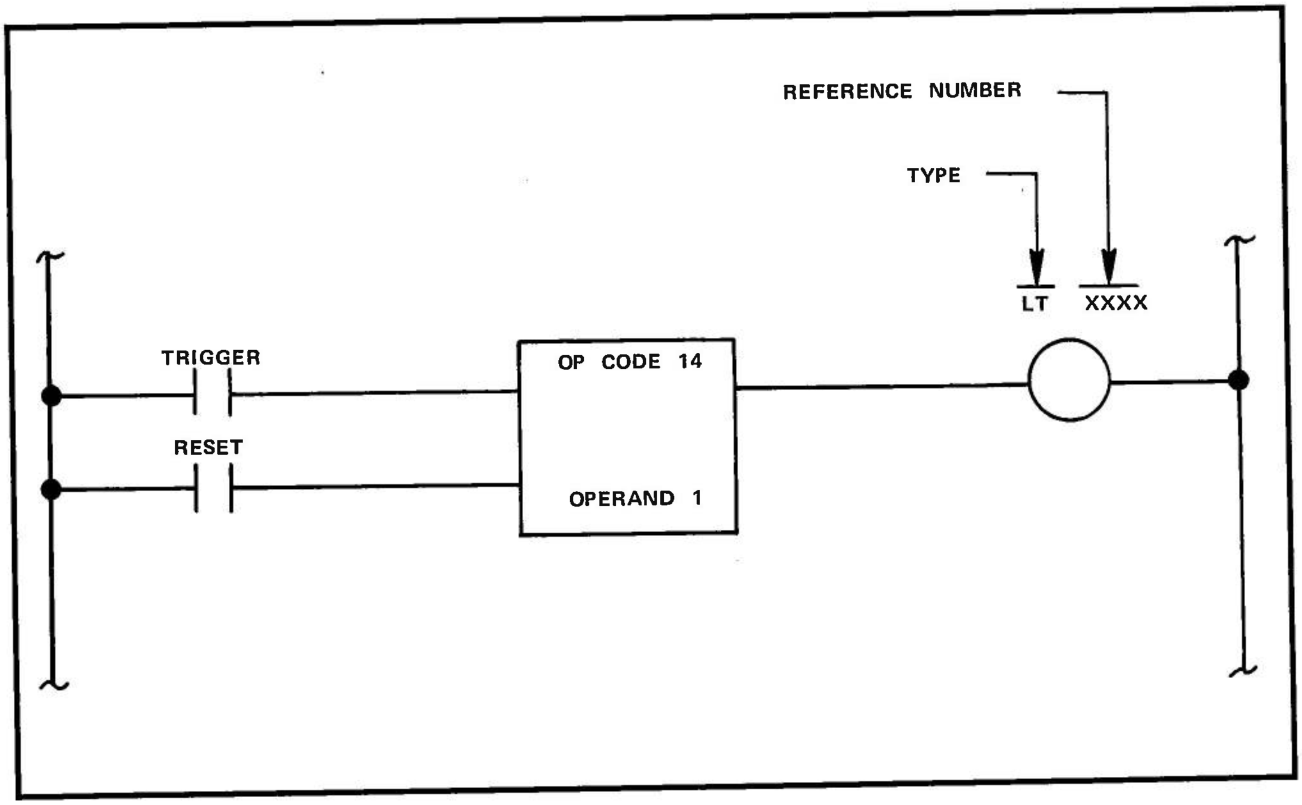


Figure 1. Reset Watchdog Timer (RW)

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	1	The Watchdog Timer is added to Operand 1. The Watchdog Timer is reset. The coil energizes if Operand 1 is > 10.0. When the coil energizes, 10 msec intervals have elapsed since the Watchdog Timer has been updated. Timers in program will be inaccurate.

APPLICATIONS

In the PC-900, 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. This action is necessary to maintain accurate timing at the timer tenths level. 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. (Special function timers are rendered inoperative when the trigger input to the function is closed.)

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 skipped.

When the fill loop is activated, the loop first updates the input group/output group to ensure that priority actions take place concerning IN0002, UC0002. Then, the loop updates the input from the scale IR0001. 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 30 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.

Timers may be effectively used if they are not timing concurrently with the loop. 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.

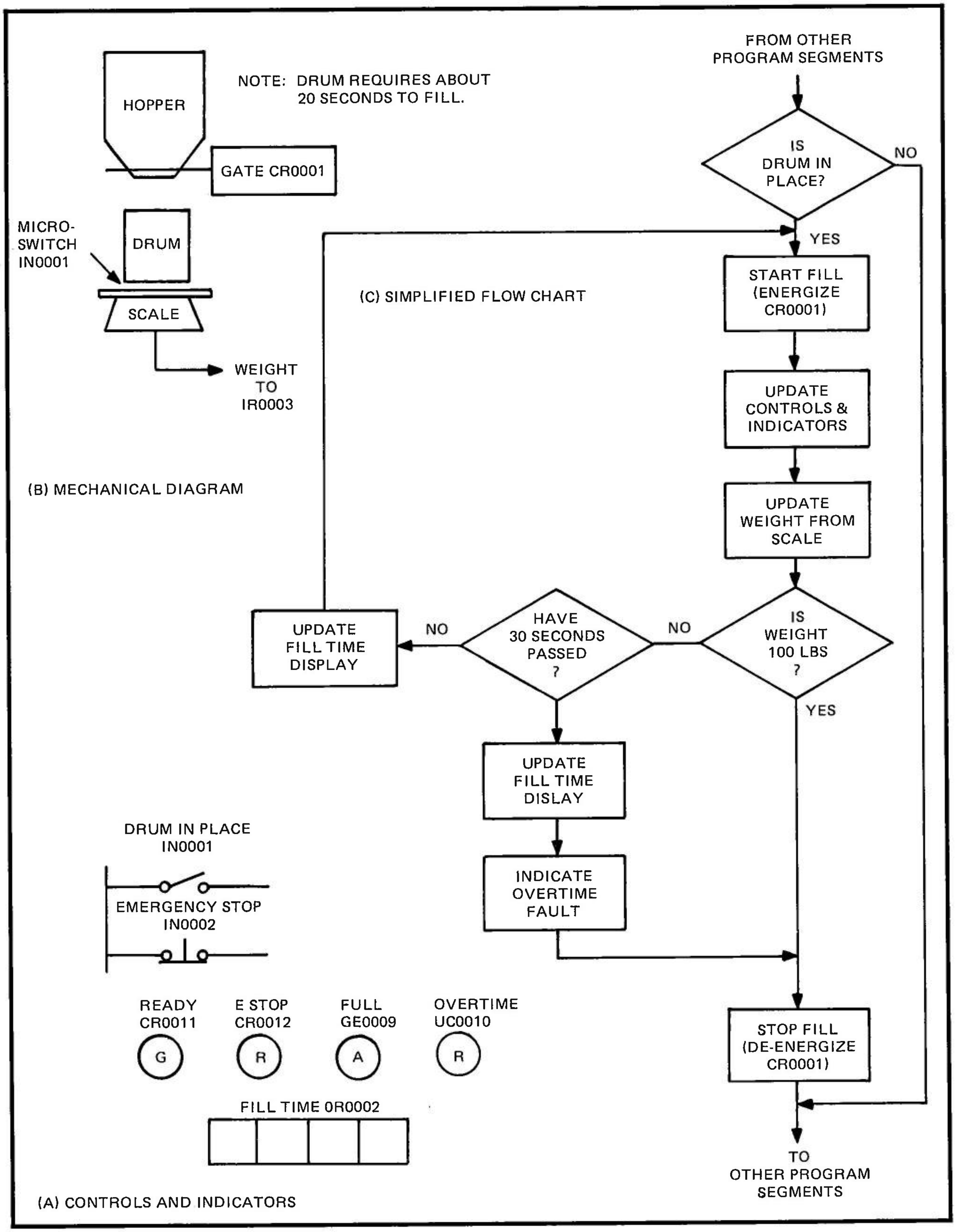


Figure 2. RW System Example

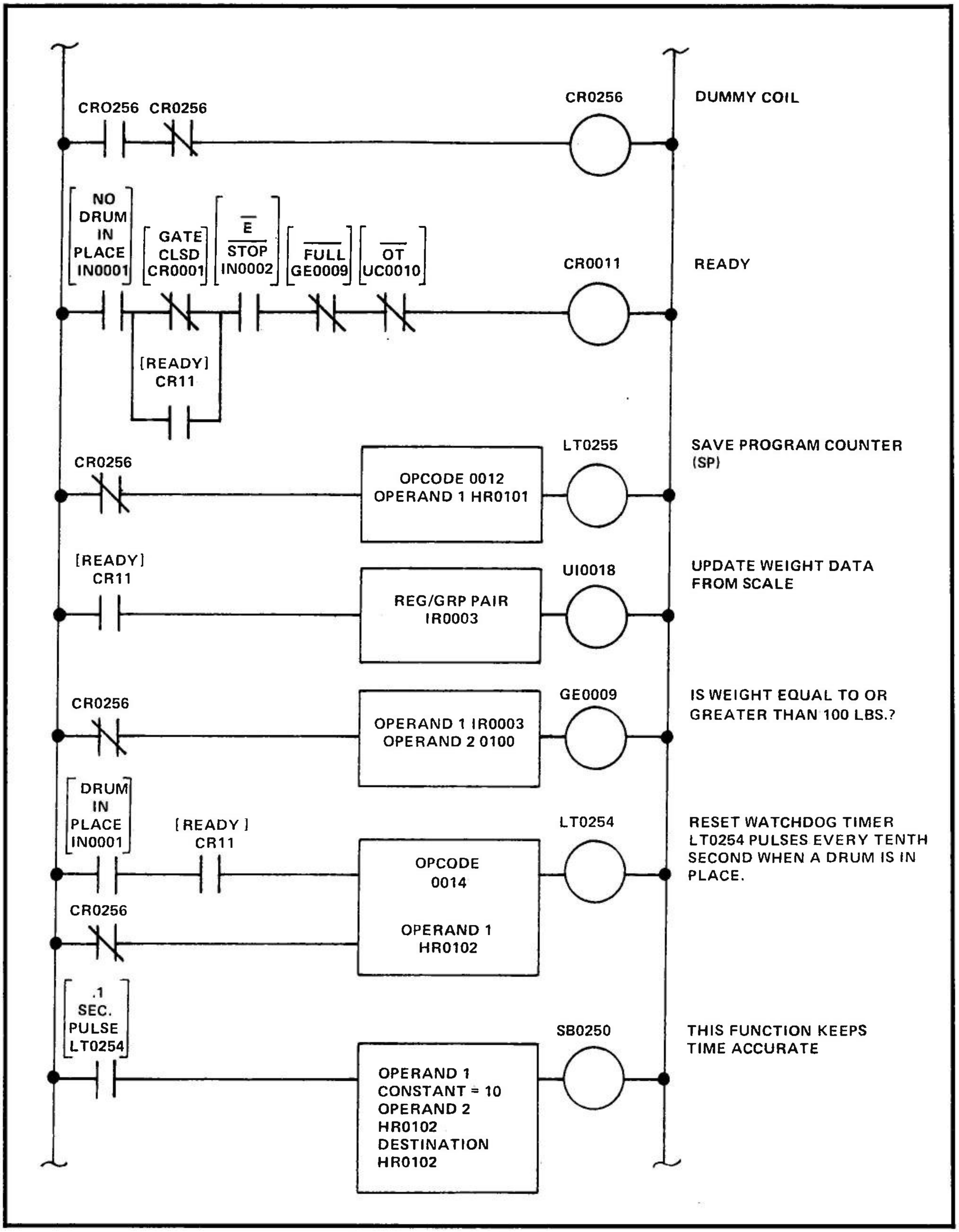
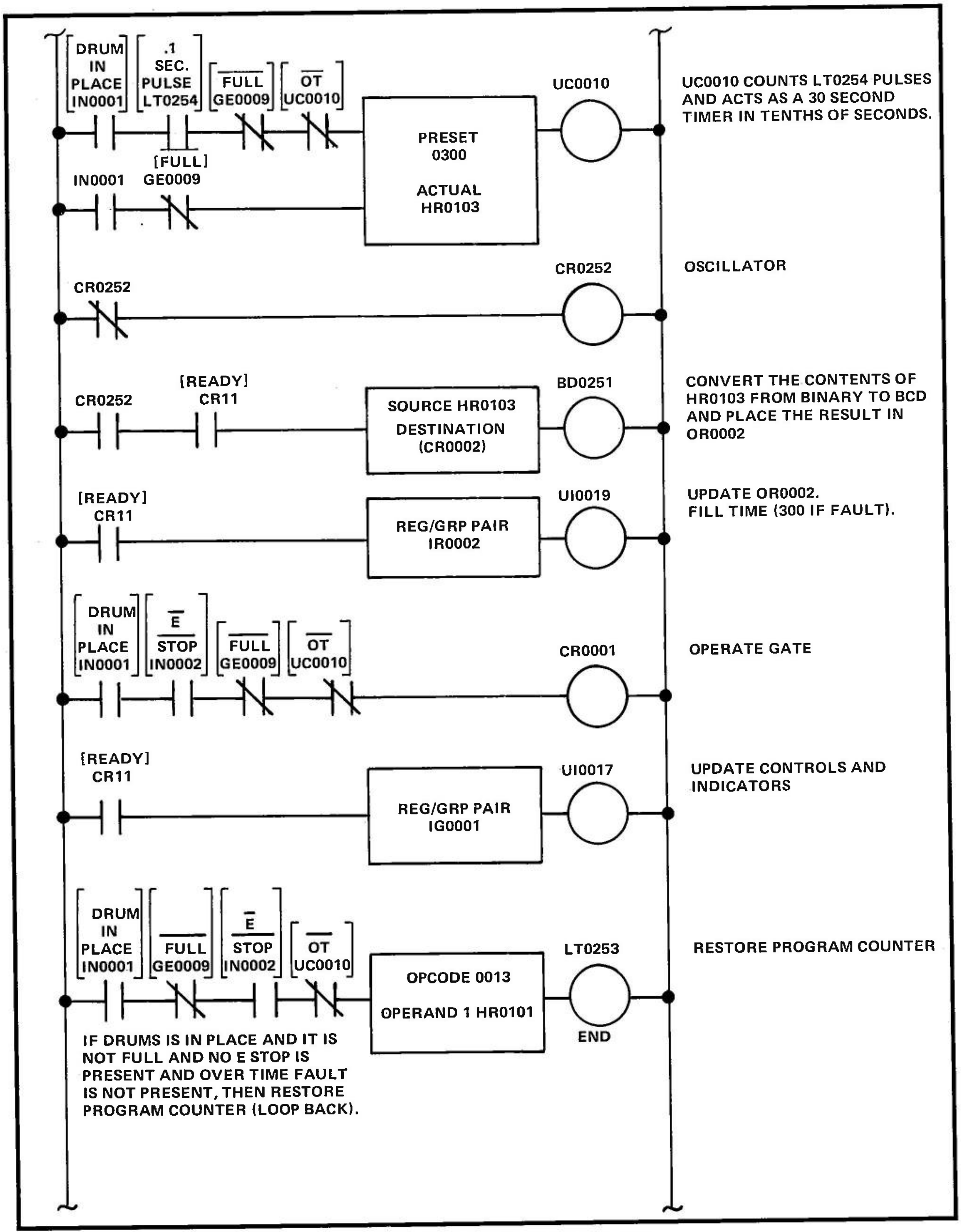


Figure 3a. RW Application



UC0010 COUNTS LT0254 PULSES AND ACTS AS A 30 SECOND TIMER IN TENTHS OF SECONDS.

OSCILLATOR

CONVERT THE CONTENTS OF HR0103 FROM BINARY TO BCD AND PLACE THE RESULT IN OR0002

UPDATE OR0002.
FILL TIME (300 IF FAULT).

OPERATE GATE

UPDATE CONTROLS AND INDICATORS

RESTORE PROGRAM COUNTER

IF DRUMS IS IN PLACE AND IT IS NOT FULL AND NO E STOP IS PRESENT AND OVER TIME FAULT IS NOT PRESENT, THEN RESTORE PROGRAM COUNTER (LOOP BACK).

Figure 3b. RW Application (Cont'd)

SK — SKIP

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.

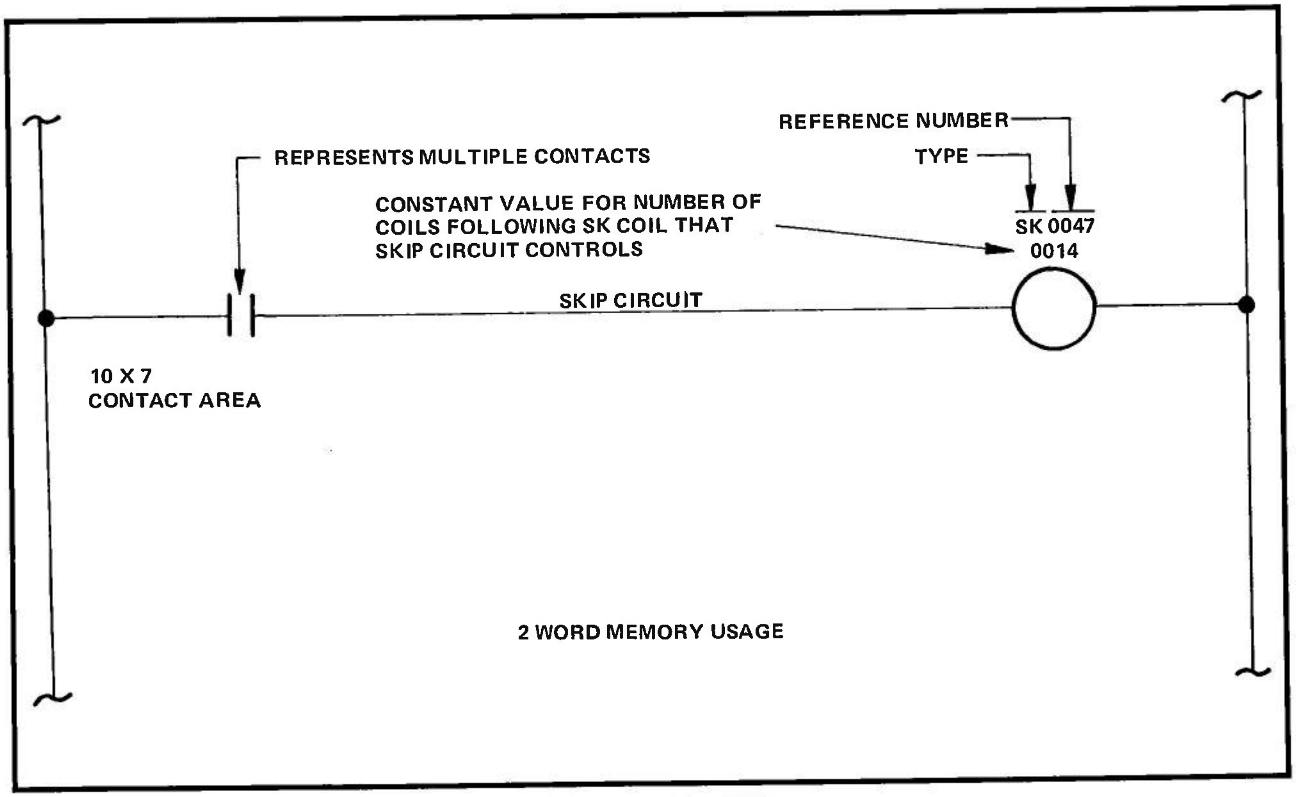

REFERENCE NUMBER
TYPE
REPRESENTS MULTIPLE CONTACTS
CONSTANT VALUE FOR NUMBER OF COILS FOLLOWING SK COIL THAT SKIP CIRCUIT CONTROLS
SK 0047
0014
SKIP CIRCUIT
10 X 7
CONTACT AREA
2 WORD MEMORY USAGE

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.

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.

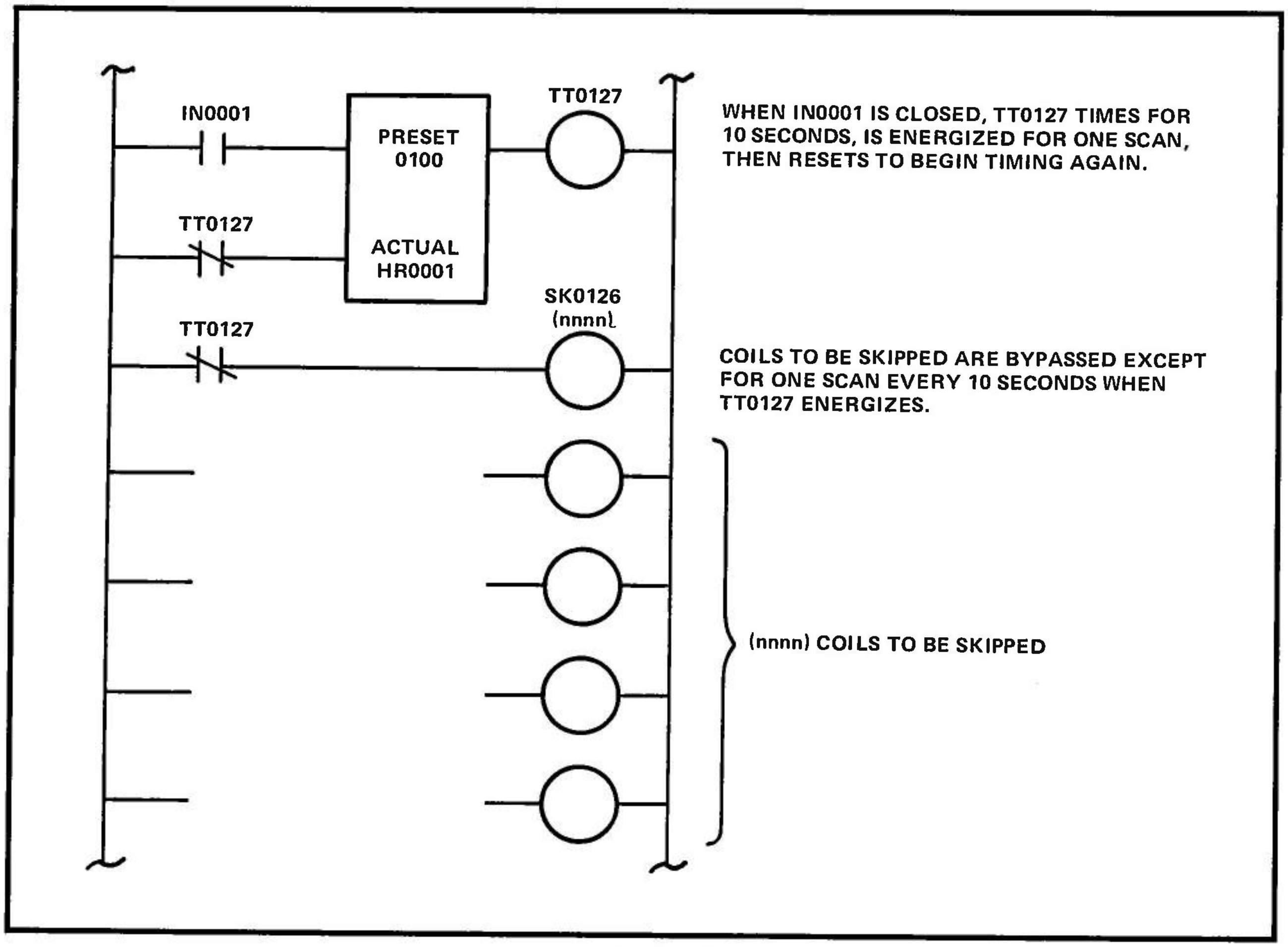

IN0001
TT0127
TT0127
PRESET
0100
ACTUAL
HR0001
TT0127
SK0126
(nnnn)
WHEN IN0001 IS CLOSED, TT0127 TIMES FOR 10 SECONDS, IS ENERGIZED FOR ONE SCAN, THEN RESETS TO BEGIN TIMING AGAIN.
COILS TO BE SKIPPED ARE BYPASSED EXCEPT FOR ONE SCAN EVERY 10 SECONDS WHEN TT0127 ENERGIZES.
(nnnn) COILS TO BE SKIPPED

Figure 2. SK Application

SM — SEARCH MATRIX

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. Each time the 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.

Consider the matrix shown in Figure 2. Each time the function is operated, a pointer (bit register) holds the bit number of the bit that is a 1 in the SM function as follows:

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
•	•
•	•

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

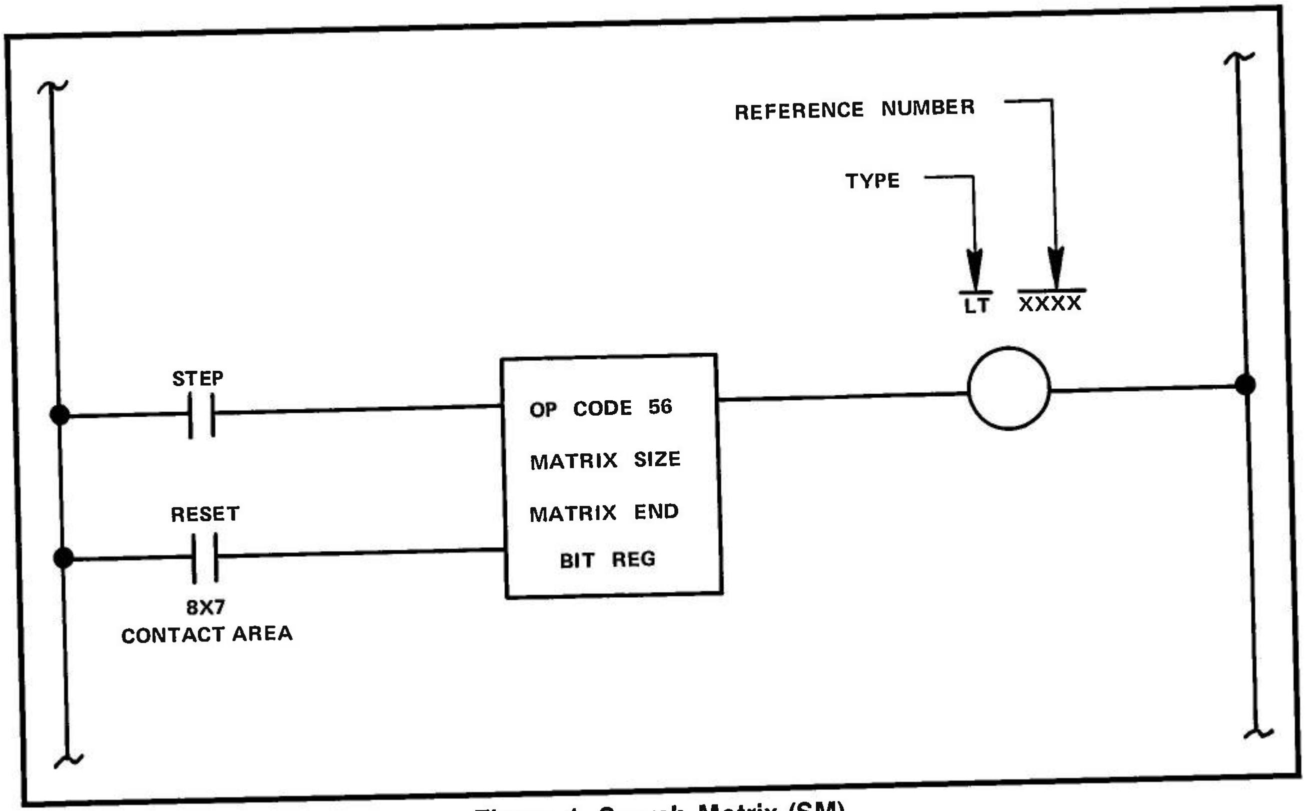


Figure 1. Search Matrix (SM)

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).

SPECIFICATIONS

COIL

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

OP CODE 56

The Op Code defines the Literal (LT) function as an SM function.

Note

> When software changes allow, LT becomes SM.

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.

MATRIX END

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

Note

> The highest number holding register is limited by and dependent upon the memory size.

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)

TABLE 1. SM END REGISTER

Type	Limit
HR	⩽ 1792
IR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)
IG	⩽ 16 (PC-700) ⩽ 8 (PC-900A/B)
OG	⩽ 32 (PC-700) ⩽ 8 (PC-900A) ⩽ 16 (PC-900B)

SM TRUTH TABLE

See Table 2.

TABLE 2. SM TRUTH TABLE

Step	Reset	Result
0	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, ↑	0	The coil is de-energized; the bit register equals zero.
↑	1	The bit register advances to the next non-zero bit.
1	1	The coil is energized; the bit register points to a non-zero bit.

MATRIX

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

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

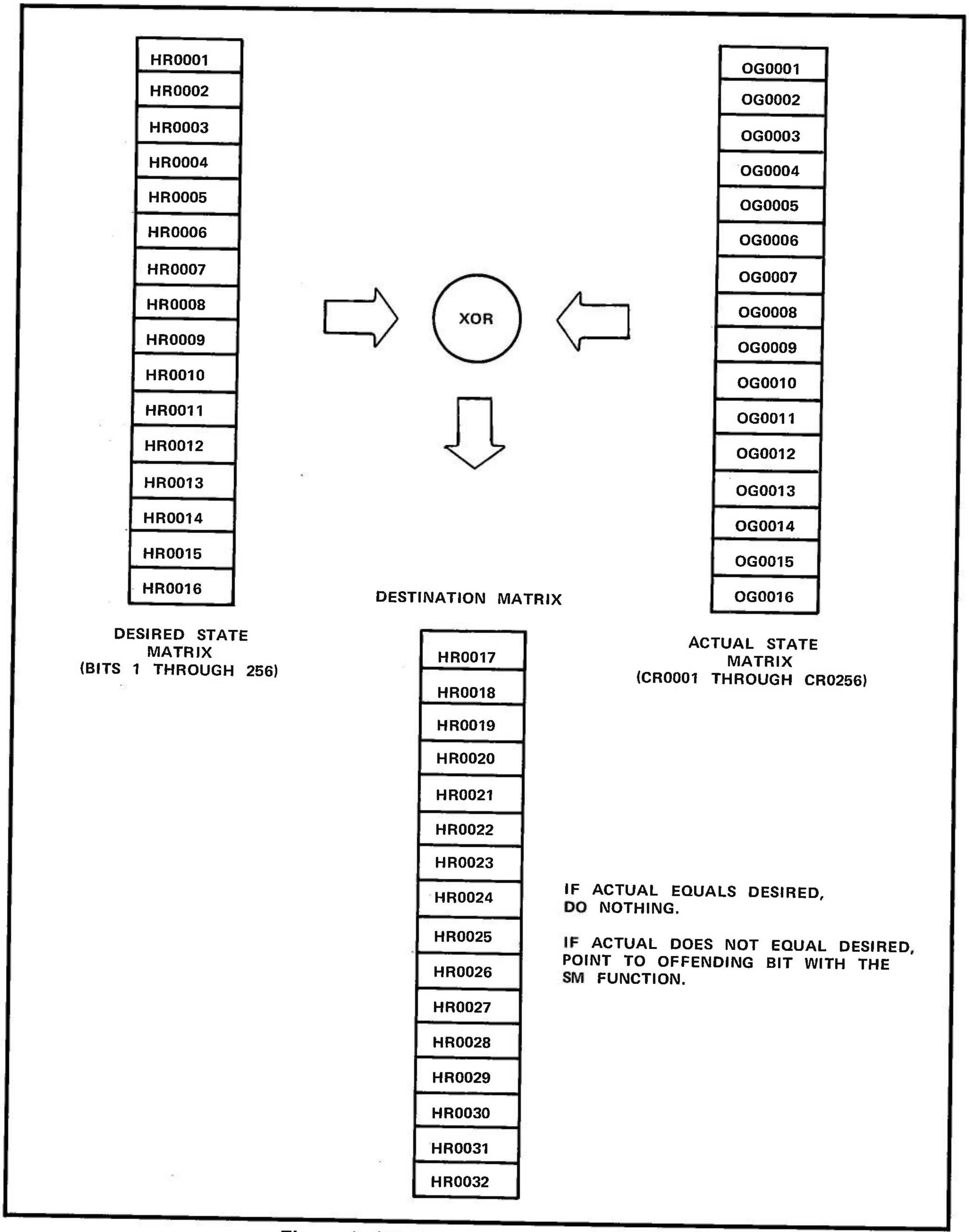
	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
HR0003	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1

Figure 2. Matrix to be Searched

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.

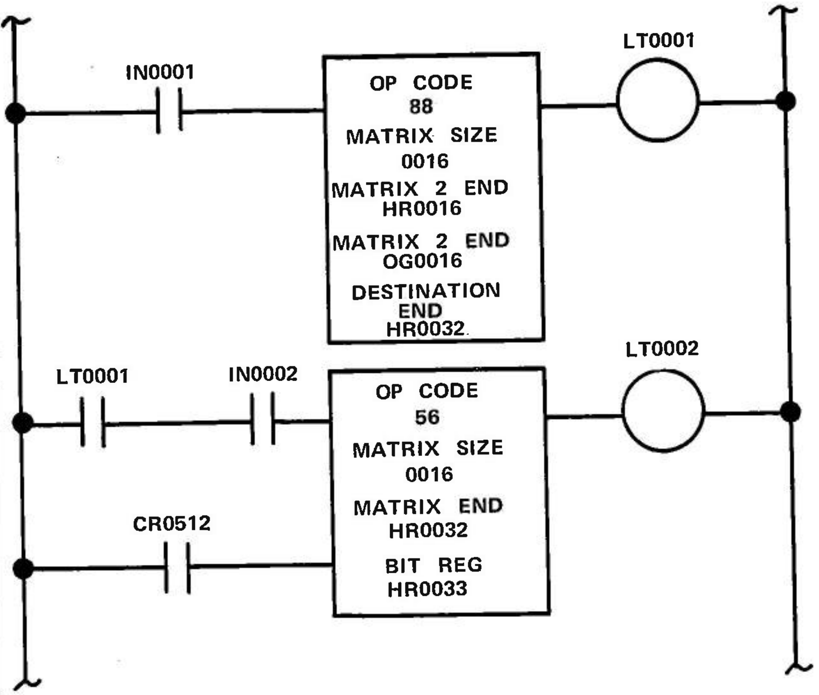
The ladder diagram for the SM function is shown in Figure 4. A non-zero result from the searched function enables IN0002 and causes the bit register to hold the number of each successive non-zero bit.



IF ACTUAL EQUALS DESIRED, DO NOTHING.

IF ACTUAL DOES NOT EQUAL DESIRED, POINT TO OFFENDING BIT WITH THE SM FUNCTION.

Figure 3. SM Function Equal-State Case



XOR MATRIX

WHEN IN0001 IS CHANGED FROM OPEN TO CLOSED, MATRIX 1 IS XOR'ED WITH MATRIX 2, AND THE RESULTS PLACED IN THE DESTINATION MATRIX. IF THE MATRICES ARE NOT EQUAL, THE COIL ENERGIZES.

SEARCH MATRIX

A NON-ZERO RESULT IN THE XOR FUNCTION ENABLES IN0002 TO CAUSE THE BIT REGISTER TO HOLD THE NUMBER OF EACH SUCCESSIVE NON-ZERO BIT.

Figure 4. SM Application

SP — SAVE PROGRAM COUNTER

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.

SPECIFICATIONS

OP CODE 12

The Op Code defines the Literal (LT) as the SP function.

Note

> When software changes allow, LT becomes SP.

OPERAND 1

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

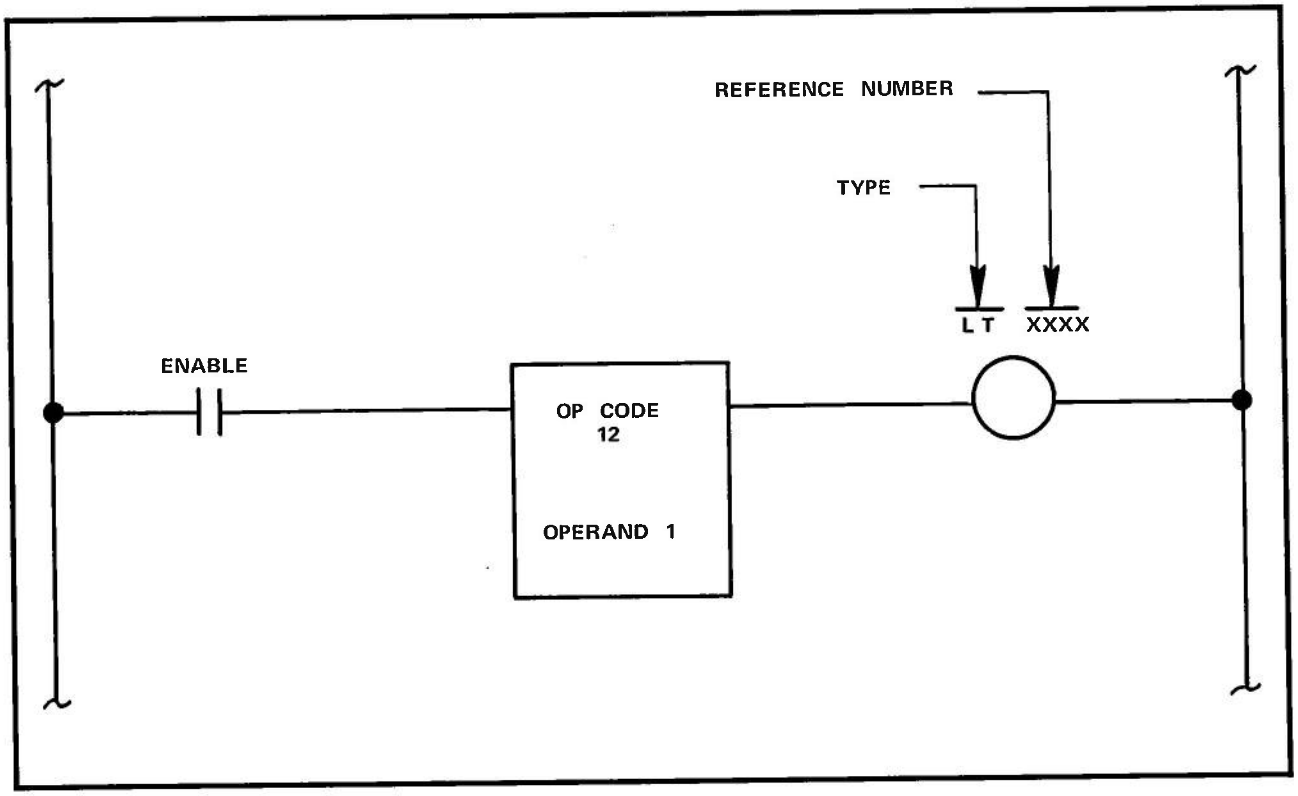


Figure 1. Save Program Counter (SP)

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.

SQ — SQUARE ROOT

DESCRIPTION

The Square Root (SQ) function allows the extraction of an eight-digit decimal number, up to 99,999,999. 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.

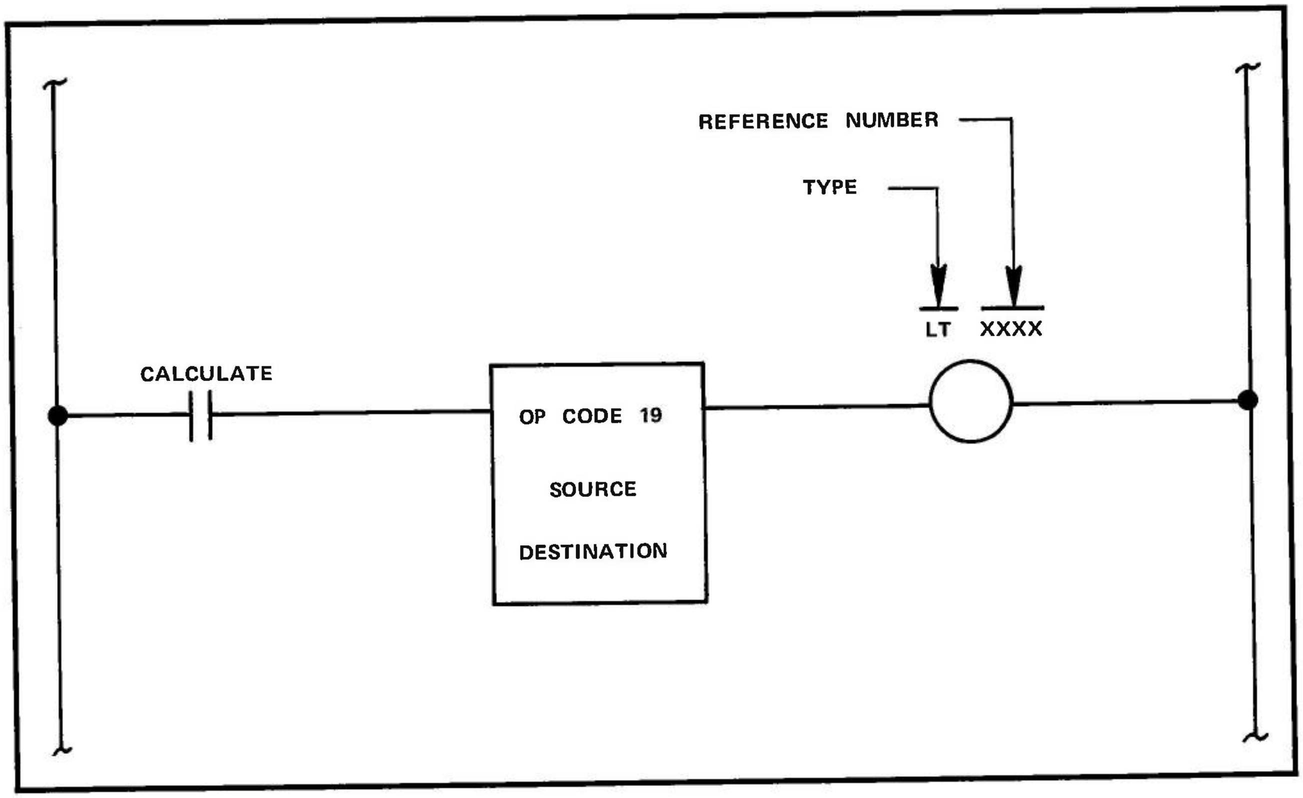


Figure 1. Square Root (SQ)

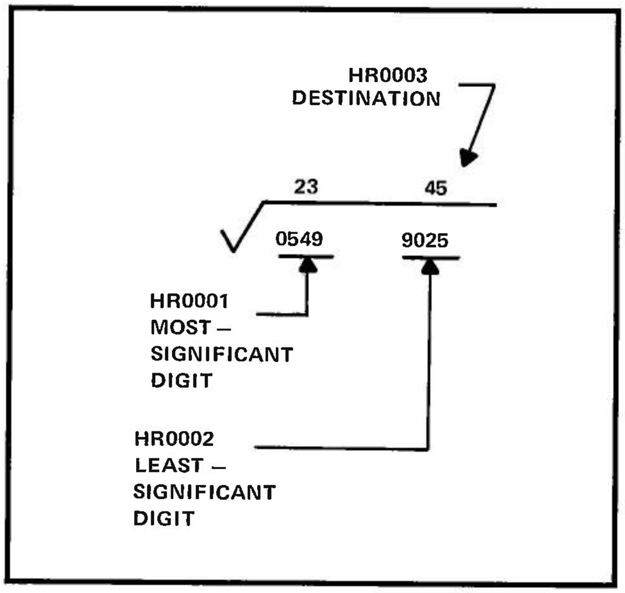


Figure 2. Source Register

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

SPECIFICATIONS

OP CODE 19

The Op Code defines the Literal (LT) as the SQ function.

Note

> When software changes allow, LT becomes SQ.

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 (HRs)
- Input Registers (IRs)
- Output Registers (ORs)

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 for the SQ function.

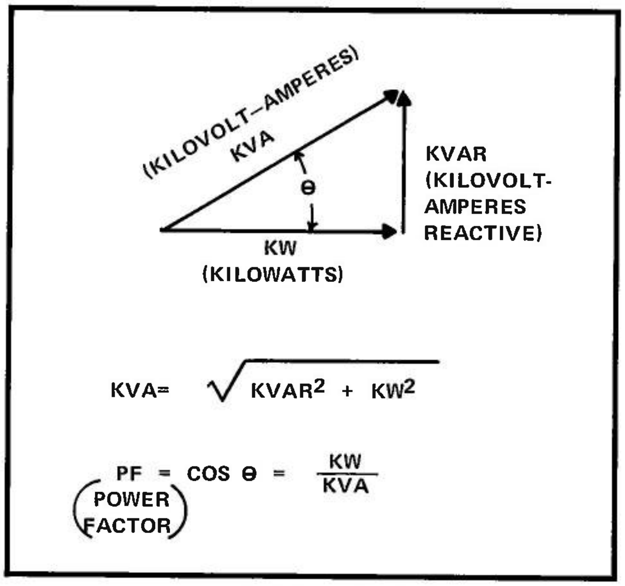
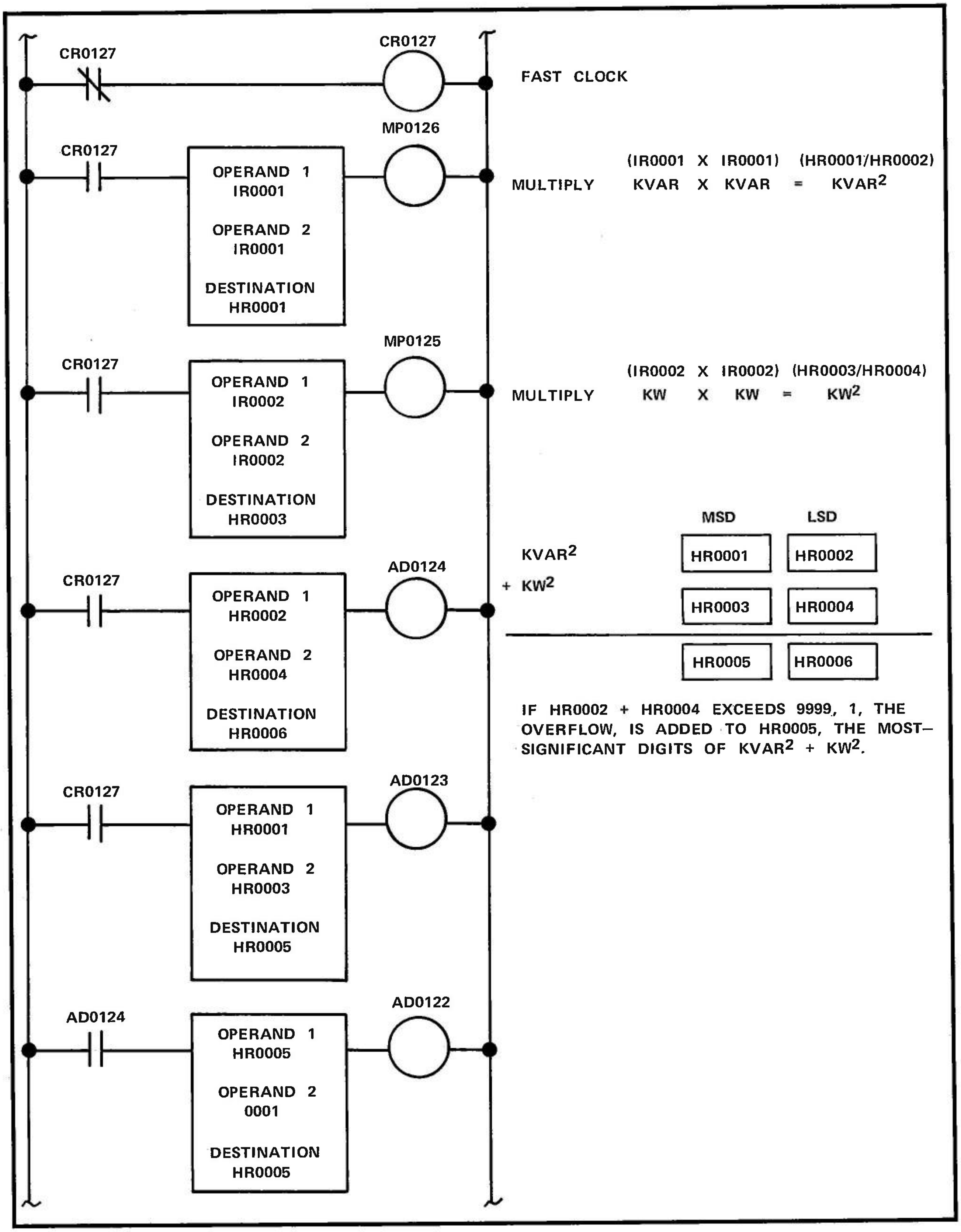
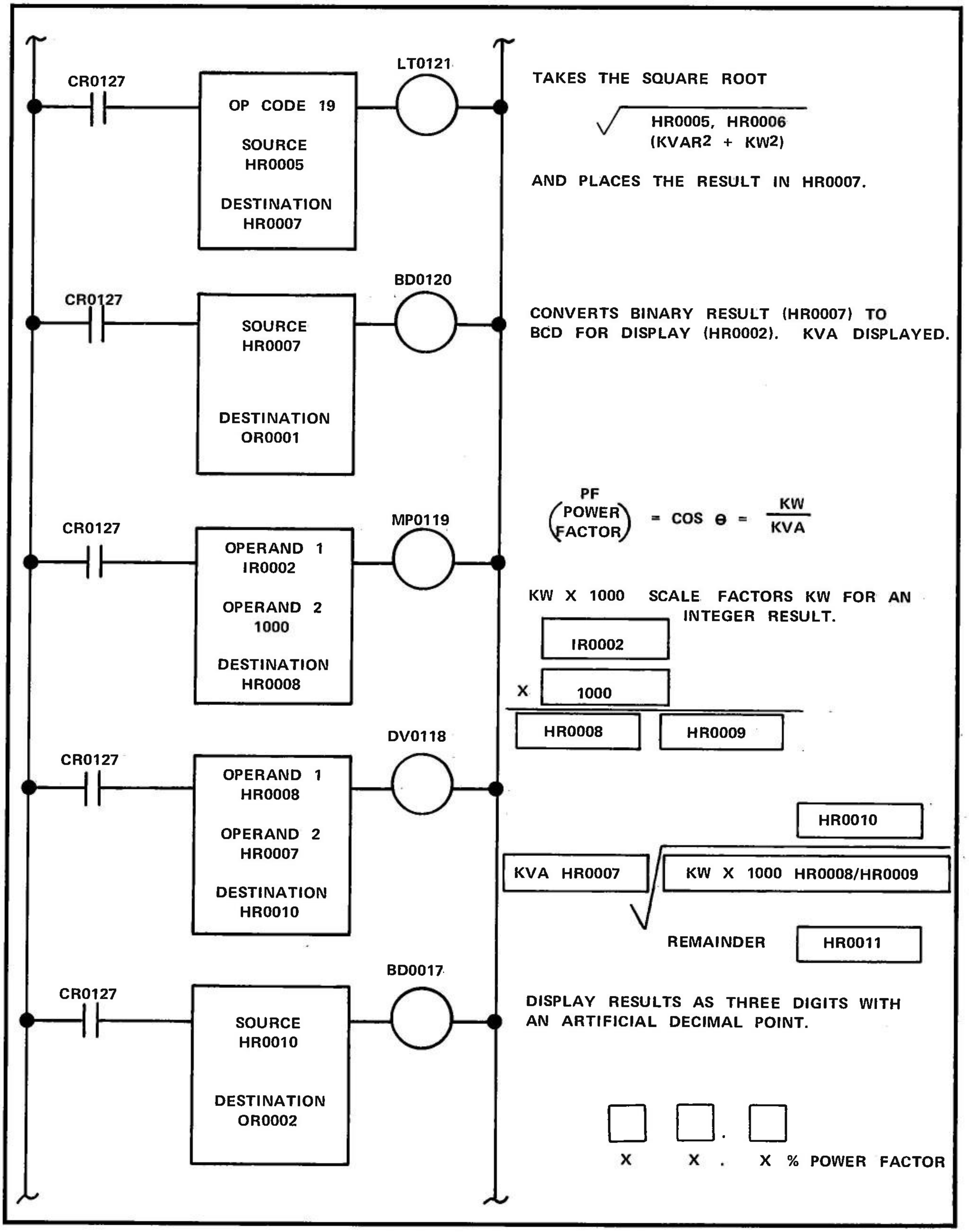


Figure 3. SQ Calculation



IF HR0002 + HR0004 EXCEEDS 9999, 1, THE OVERFLOW, IS ADDED TO HR0005, THE MOST–SIGNIFICANT DIGITS OF KVAR^2 + KW^2.

Figure 4a. SQ Application



TAKES THE SQUARE ROOT

$$\sqrt{\text{HR0005, HR0006 } (\text{KVAR}^2 + \text{KW}^2)}$$

AND PLACES THE RESULT IN HR0007.

CONVERTS BINARY RESULT (HR0007) TO BCD FOR DISPLAY (HR0002). KVA DISPLAYED.

$$\overset{\text{PF}}{\left(\begin{matrix}\text{POWER}\\\text{FACTOR}\end{matrix}\right)} = \cos\theta = \frac{\text{KW}}{\text{KVA}}$$

KW X 1000 SCALE FACTORS KW FOR AN INTEGER RESULT.

DISPLAY RESULTS AS THREE DIGITS WITH AN ARTIFICIAL DECIMAL POINT.

Figure 4b. SQ Application (Cont'd)

SR/SL — SHIFT REGISTERS

DESCRIPTION

The Shift Register functions provide the vehicle for serial shifting register data under program control. On command from the program, the contents of the register are shifted one place to the right by using the Shift Right (SR) function or one place to the left by using the Shift Left (SL) function. SR/SL function symbology is shown in Figure 1.

Three contact circuits control the SR/SL functions: the shift circuit, the serial data circuit, and the enable circuit. If registers are chained, these three lines control the chain as though it were a long single register. Shift operations are allowed only when the enable circuit is conducting. When the enable circuit is not conducting, all associated registers are cleared and the coil is turned OFF (each bit becomes a 0). Each time the shift circuit changes from nonconducting to conducting, the bits are shifted to the next position. The exit bit in the register shifts out of its position and determines the state of the coil. A logic 1 energizes the coil, and a logic 0 de-energizes the coil. The entry position in the register is filled depending on the condition of the serial data circuit. When the serial data contact is conducting, a logic 1 is entered, and when not conducting, a logic 0 is entered.

SR/SL functions operate on designated registers or groups of registers, and affect all bits in a register at once. When a shift command is received, each bit is moved to the next position; all bits move to the right or to the left. Bits are loaded at one end of the register and leave at the other end, as shown in Figure 2.

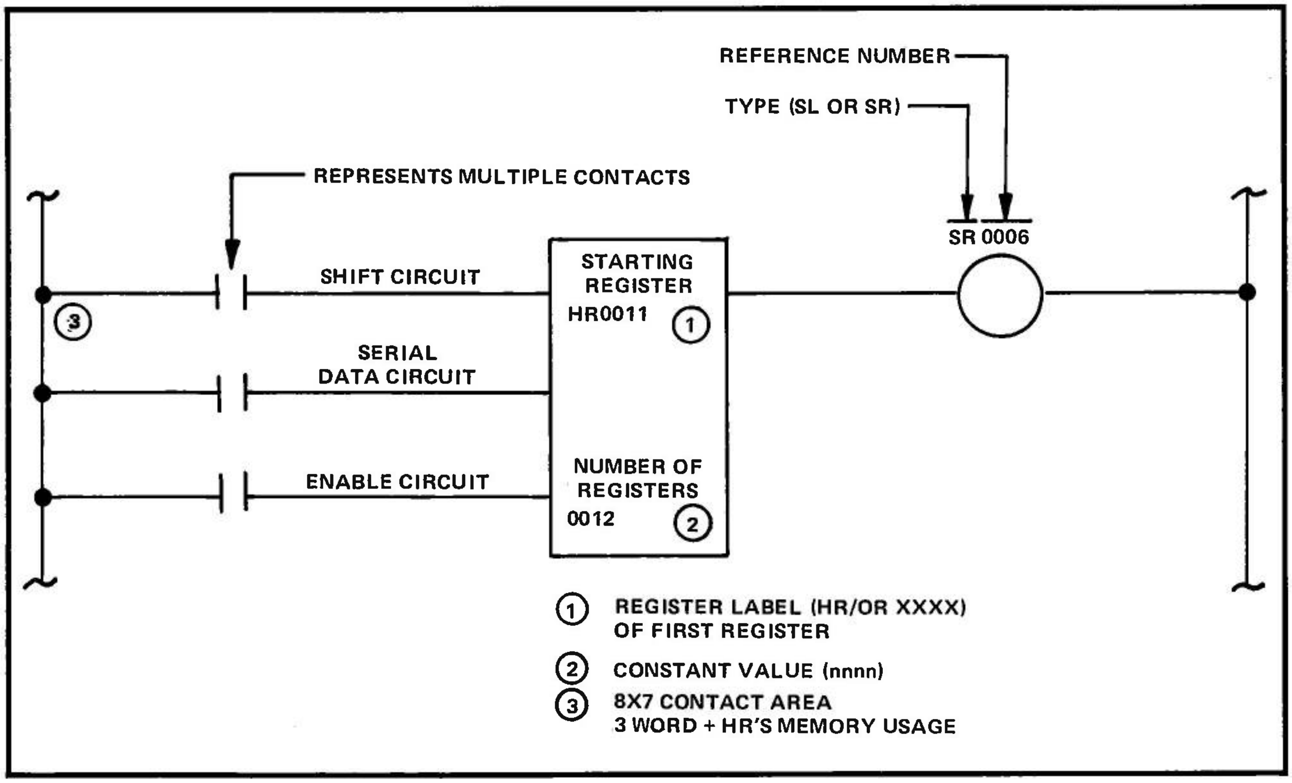


Figure 1. Shift Right (SR)/Shift Left (SL)

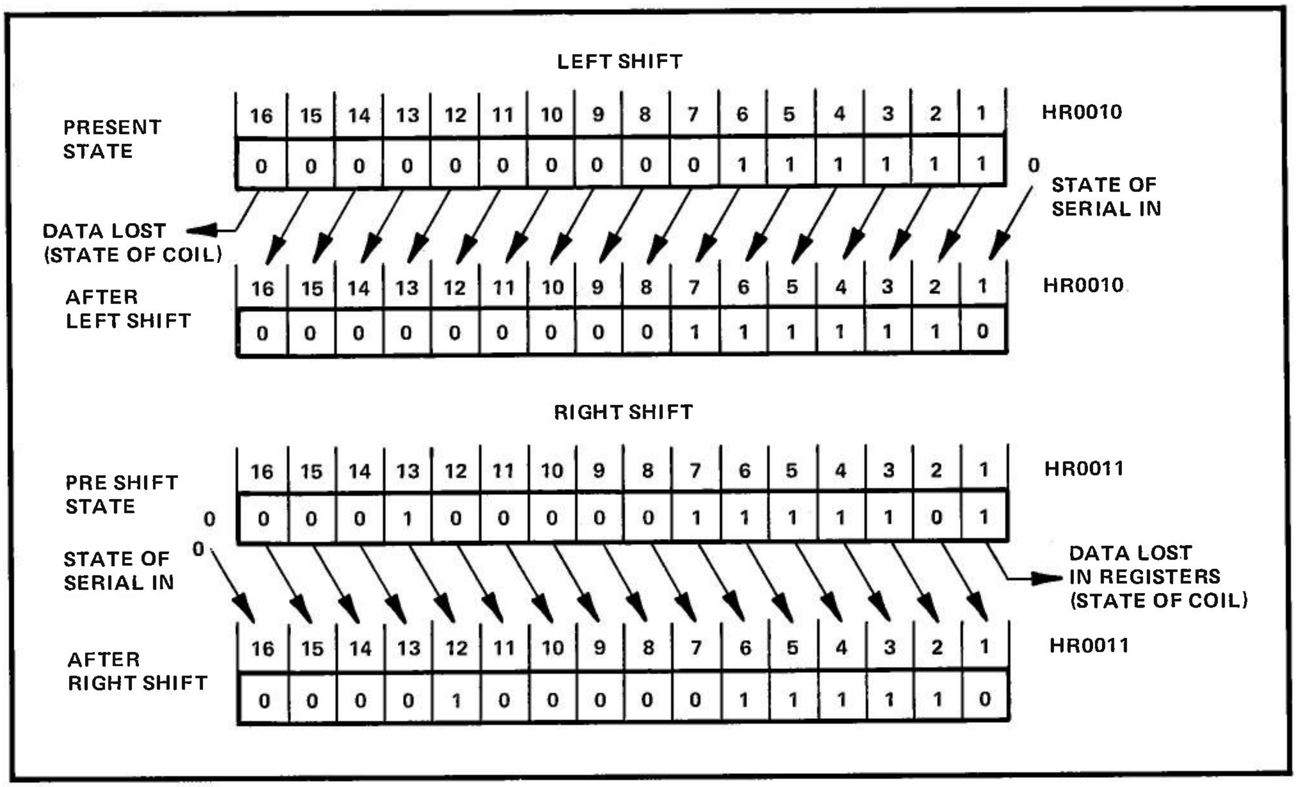


Figure 2. Left and Right Shifts

In addition to shifting the data in a register, the SR/SL functions provide a means for bringing and taking the serial information into and out of a register. When the shift command is given and the serial input circuit is closed, a 1 is placed in the appropriate shift register bit. If a 1 is shifted out of the register when a shift command is given, the associated coil is energized.

Shift registers are also chained with a limit of 128 Holding Registers (HRs) or 32 Output Registers (ORs) in length. Figure 3 shows the effects of chaining. When registers are chained, registers beyond the starting register assume a dual identity (i.e., HR0005, Bit 15 is also addressed as HR0004, Bit 31).

SPECIFICATIONS

SR/SL TRUTH TABLE

See Table 1.

TABLE 1. SR/SL TRUTH TABLE

Enable Circuit	Serial Data Circuit	Shift Circuit	Result
0	X	X	The contents of the shift register are zeroed. The coil is de-energized.
1	0, 1	0, 1	The coil reflects the state of bit shifted out. The contents of the shift register are frozen.
1	0	↑	A 0 is shifted into the register. The coil reflects the state of the bit shifted out.
1	1	↑	A 1 is shifted into the register. The coil reflects the state of the bit shifted out.
X = Don't care ↑ = Transition OFF to ON.			

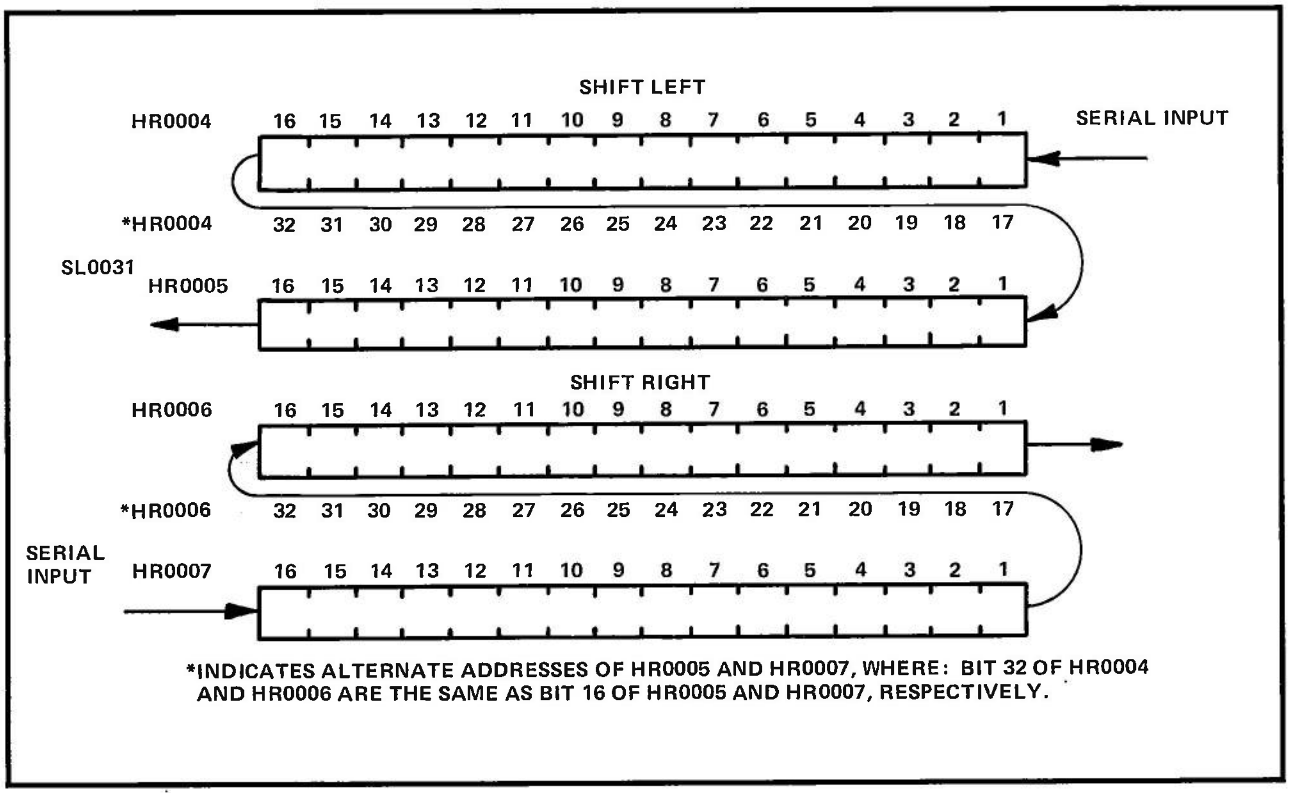


Figure 3. Chained Shift Registers

APPLICATIONS

Promotional and warning displays often use moving lights as attention-getting devices. The warning arrow at highway construction sites is one such example which illustrates basic shift register operation. The arrow is capable of signaling a move from right to left or from left to right. The warning arrow display configuration is shown in Figure 4. A ladder diagram program implementing this display is shown in Figure 5.

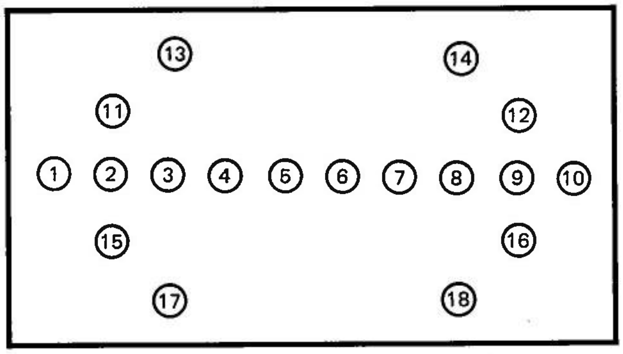
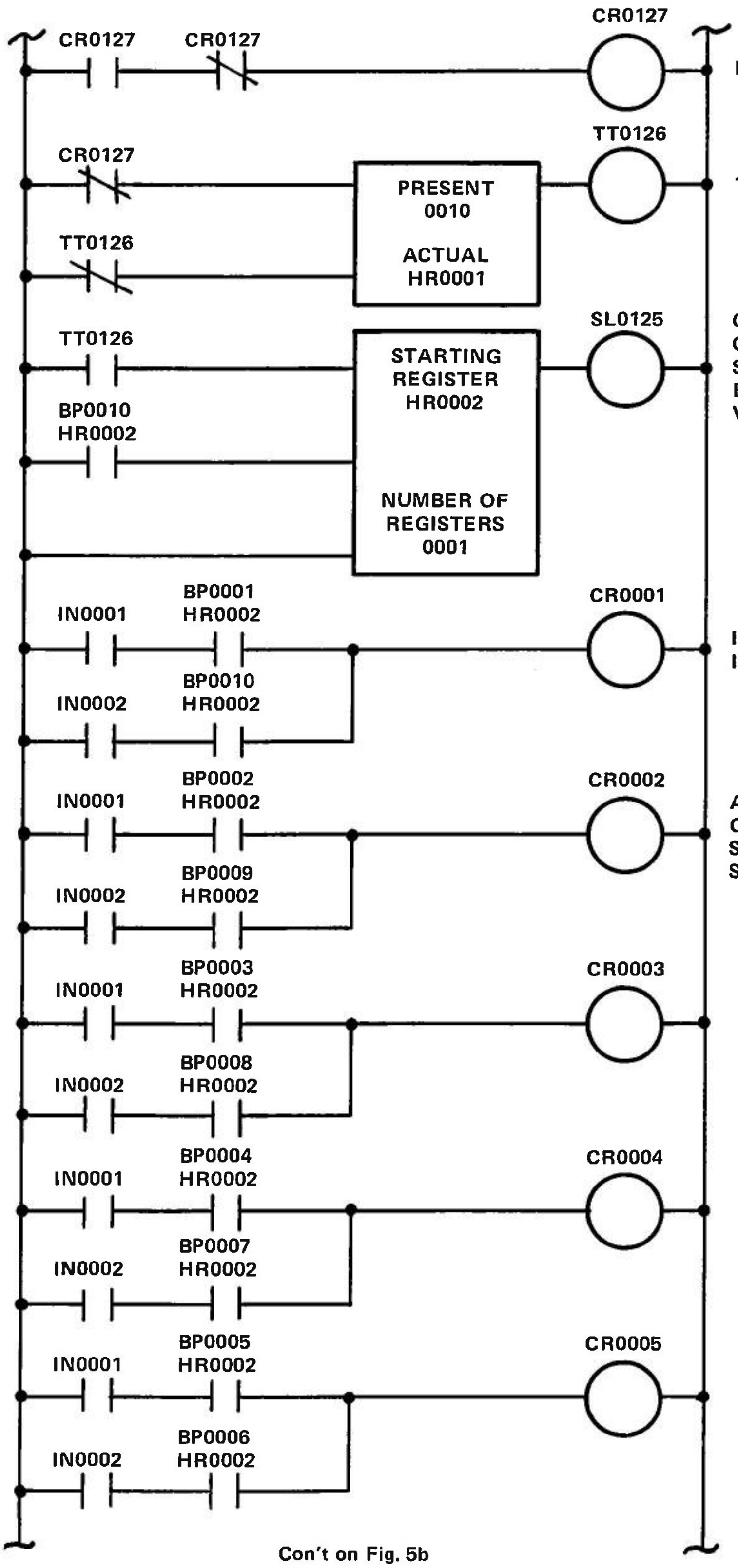


Figure 4. Warning Arrow Display



DUMMY COIL

1 SECOND SELF RESETTING TIMER

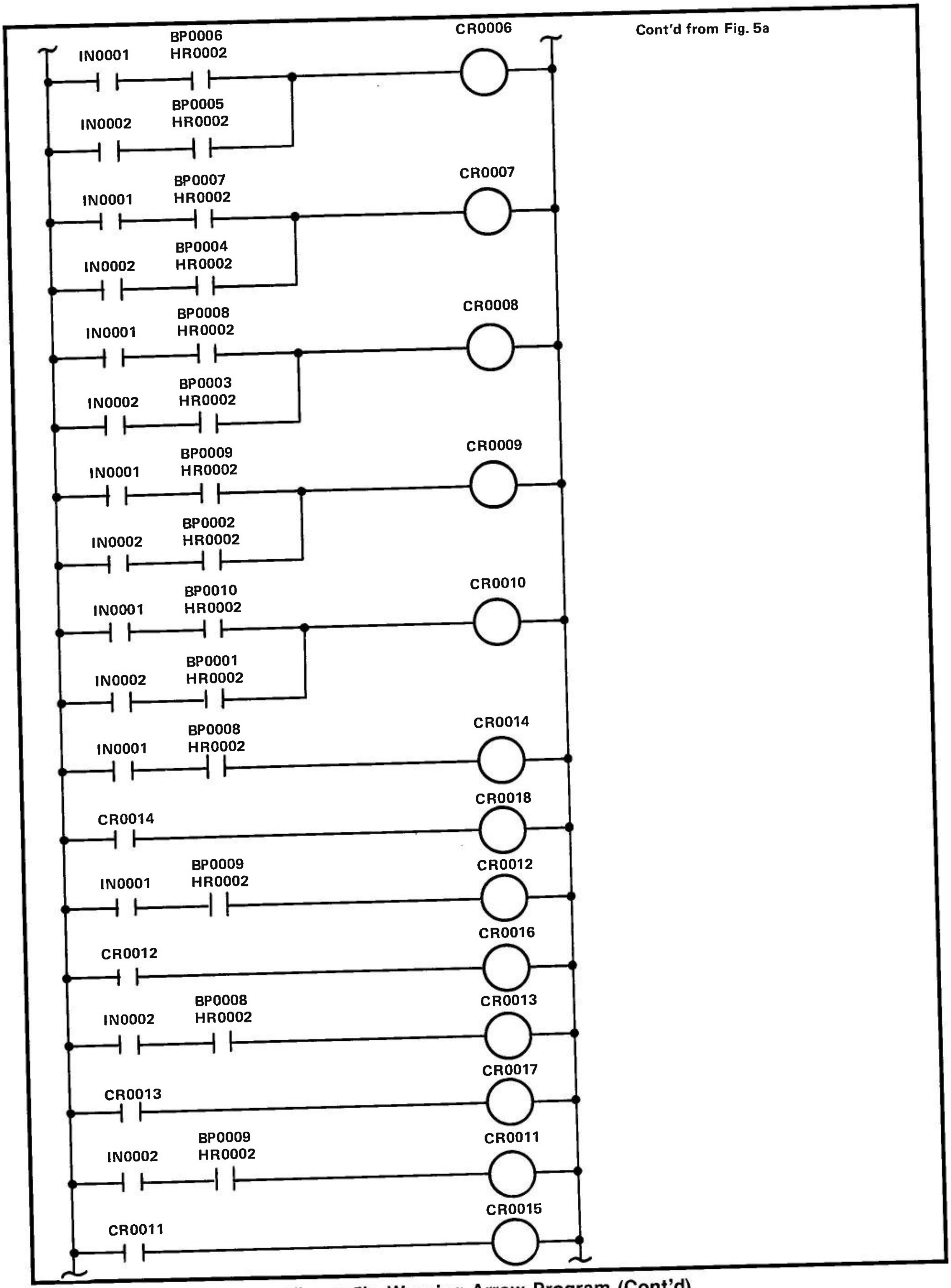
CIRCULATING SHIFT REGISTER. THE CONTENTS OF THE SHIFT REGISTER WILL SHIFT OUT AT BIT 10 AND RE-ENTER AT BIT 1. HR0002 MUST BE PRESET TO A VALUE OF 1 BEFORE INITIAL OPERATION.

IN0001 ⇒ LEFT-TO-RIGHT →
IN0002 ⇒ RIGHT-TO-LEFT →

AS THE SINGLE BIT IN HR0002 IS SHIFTED ONCE PER SECOND, THE LIGHTS WILL SEQUENCE AS DETERMINED BY THE SETTINGS OF IN0001 OR IN0002.

Con't on Fig. 5b

Figure 5a. Warning Arrow Program



Cont'd from Fig. 5a

Figure 5b. Warning Arrow Program (Cont'd)

TL/TO — TABLE LOOKUP/TABLE LOOKUP ORDERED

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.

SPECIFICATIONS

OP CODE 82/83

The Op Code defines the Literal (LT) as TL or TO. Op Code 82 indicates TL; Op Code 83 indicates TO.

Note

When software changes allow, LT becomes TL or TO.

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.

TABLE END

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

POINTER

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

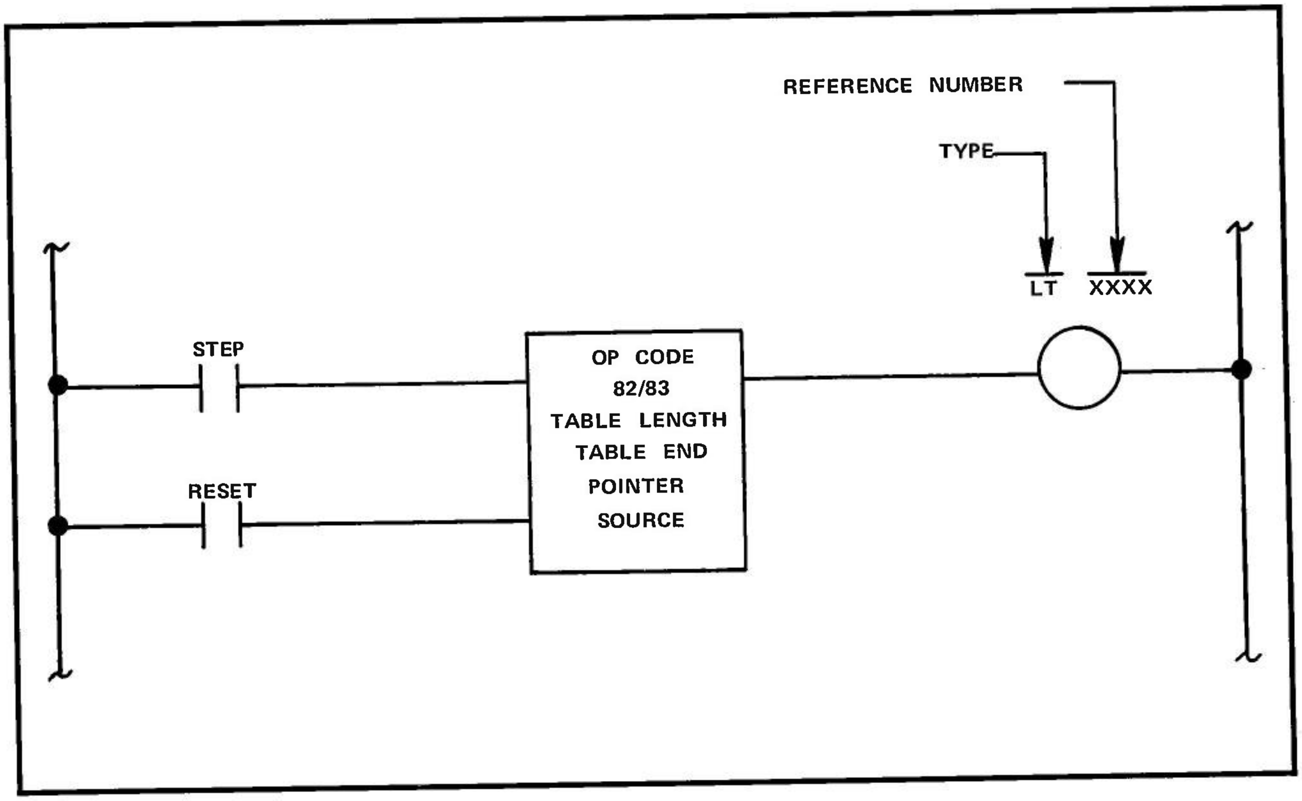


Figure 1. Table Lookup (TL)/Table Lookup Ordered (TO)

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

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.

TABLE 1. TL/TO TRUTH TABLE

Step	Reset	Result
Don't Care	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. TL Function Table = Source Register TO Function Table ⩾ 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 previous position. The coil is energized.

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 2. SINE VALUES AND CORRESPONDING ANGLES

Angle°	Sine
0	0000
1	0175
2	0349
3	0523
4	0698
5	0872
6	1045
•	•
•	•
•	•
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, LT0255 is a TO function. For IN0001, the sine centered 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 7100 is placed in IR0001, and IN0001 is closed, the pointer goes to 45 to indicate that IR0001 is the sine of an angle of 45°. This is an approximation. A value of IR0001 between 7071 and 7192 indicates an angle of 45°. A value equal to or greater than 7193 indicates an angle greater than 45°. A value less than or equal to 7070 indicates an angle less than 45°.

The TL function can find a number that exactly matches the number held in the source register and report its location using the pointer. Figure 3 shows a storage system that uses a Register-to-Table Move (RT) function to record the storage of data and a TL function to retrieve the location. See Figure 3.

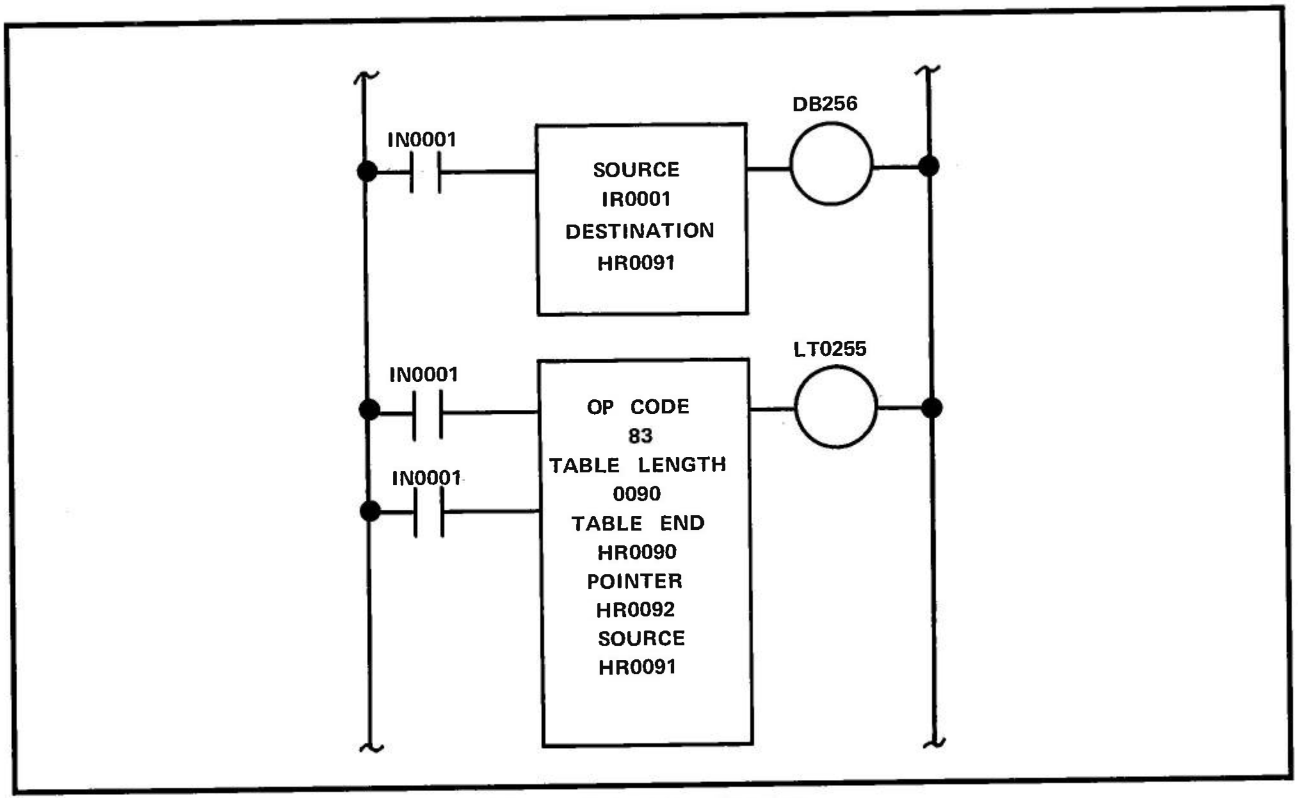
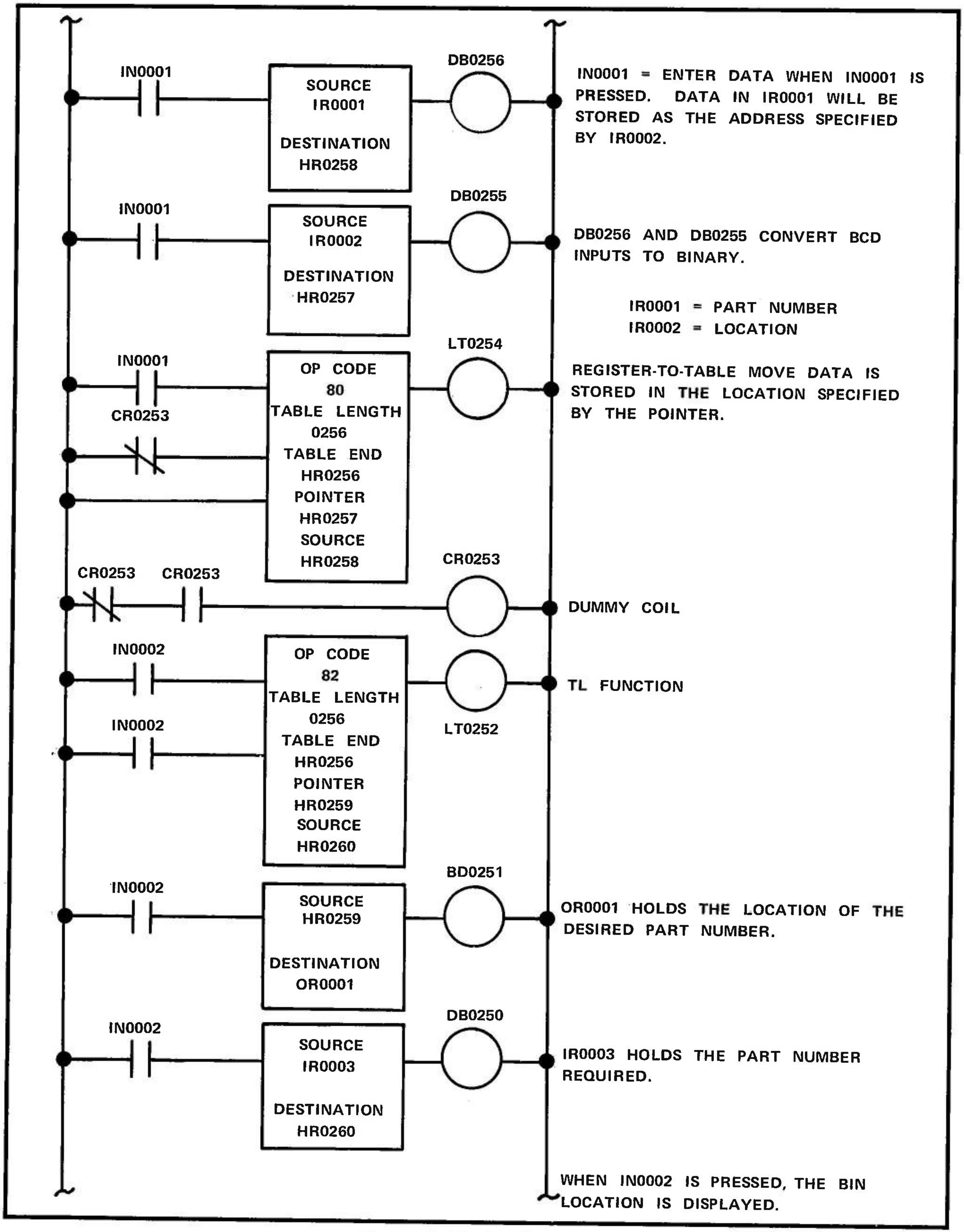


Figure 2. TO Application



IN0001 = ENTER DATA WHEN IN0001 IS PRESSED. DATA IN IR0001 WILL BE STORED AS THE ADDRESS SPECIFIED BY IR0002.

DB0256 AND DB0255 CONVERT BCD INPUTS TO BINARY.

IR0001 = PART NUMBER
IR0002 = LOCATION

REGISTER-TO-TABLE MOVE DATA IS STORED IN THE LOCATION SPECIFIED BY THE POINTER.

DUMMY COIL

TL FUNCTION

OR0001 HOLDS THE LOCATION OF THE DESIRED PART NUMBER.

IR0003 HOLDS THE PART NUMBER REQUIRED.

WHEN IN0002 IS PRESSED, THE BIN LOCATION IS DISPLAYED.

Figure 3. TL Application

TR — TABLE-TO-REGISTER MOVE

DESCRIPTION

The Table-to-Register Move (TR) function allows the contents of a table (successive register locations) to be moved into a destination register location. The relative location of the data being transferred is determined by the value of the contents of a pointer location.

SPECIFICATIONS

COIL

The coil energizes when the pointer equals the table length minus one, and de-energizes at all other times.

OP CODE 84

The Op Code defines the Literal (LT) as the TR function.

Note

> When software changes allow, LT becomes TR.

TABLE LENGTH

The table length defines the number of registers in the table to be transferred. This number ranges from 1 through 256, and is subject to the limits listed in Table 1.

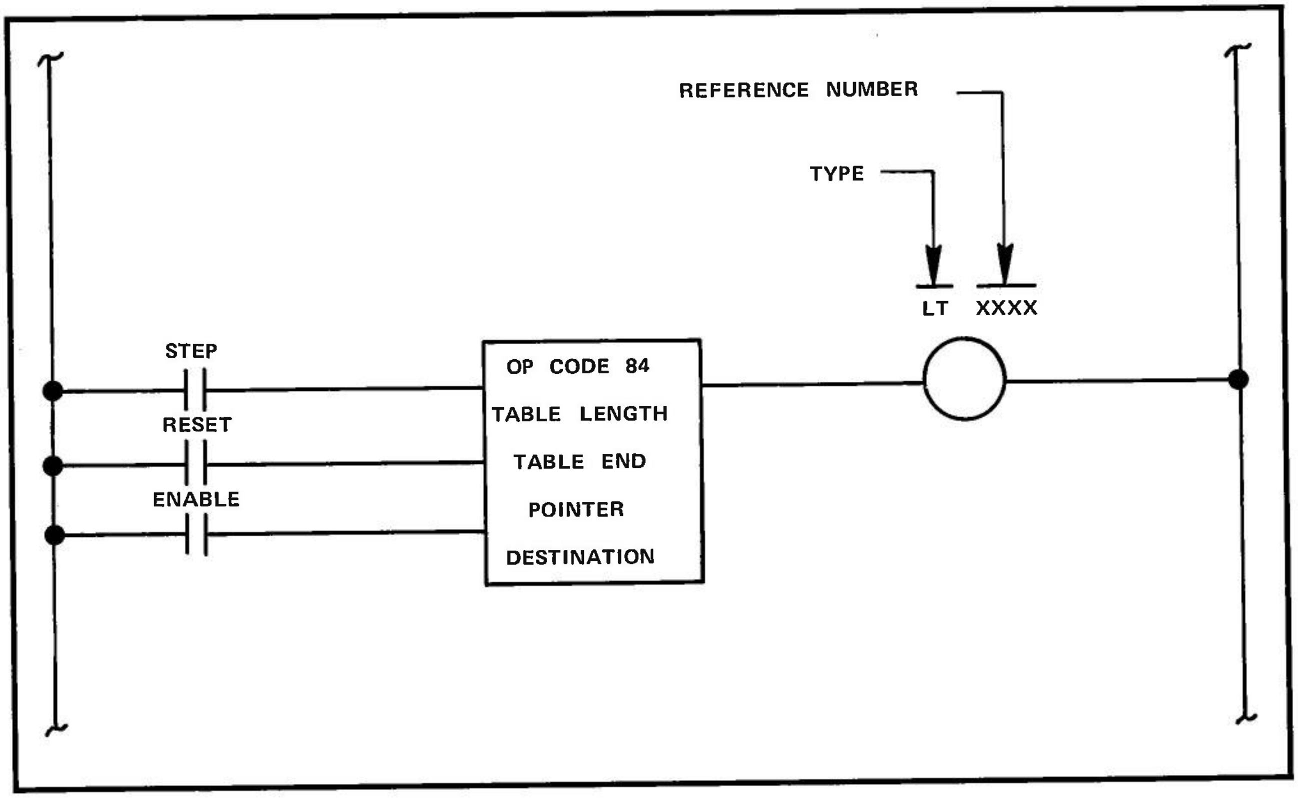


Figure 1. Table-to-Register Move (TR)

TABLE 1. TABLE LENGTH/TABLE END LIMITS

Type	Limit
HR	⩽ 1792
IB	⩽ 32 (PC-700) ⩽ 8 (PC-900 A) ⩽ 16 (PC-900 B)
OR	⩽ 32 (PC-700) ⩽ 8 (PC-900 A) ⩽ 16 (PC-900 B)
IG	⩽ 16 (PC-700) ⩽ 8 (PC-900 A/B)
OG	⩽ 32 (PC-700) ⩽ 8 (PC-900 A) ⩽ 16 (PC-900 B)

TABLE END

The table end defines the type and number of the last register in the table to be transferred. It is subject to the limits listed in Table 1.

Note

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

POINTER

The pointer holds the current table location from which data is being transferred to the destination. The pointer is a specified register:

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

DESTINATION

The destination holds the information indicated by the pointer, and is a specified register or group:

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

TR MOVE TRUTH TABLE

See Table 2.

TABLE 2. TR TRUTH TABLE

Step	Reset	Enable	Coil Action/Results
0, 1 or ↑	0	0	The pointer is set to zero and held as long as reset is false. No data is moved.
0, 1 or ↑	1	0	The pointer is frozen at the current value. No data is moved.
0, 1 or ↑	0	1	The pointer is set to zero. The contents of the table start are moved to the destination.
↑	1	1	The pointer is incremented. If the pointer equals the table length minus one, data is transferred and the coil is energized. If the pointer is ⩾ table length, the pointer is zeroed and data is transferred.
0 or 1	1	1	Data is transferred. (Data is continuously moved both prior to and after the step function.)

APPLICATIONS

The TR function can be used with its companion, the Register-to-Table Move (RT) function. Consider the example shown in Figure 2. Normally, the transmission of 256 inputs to the remote site requires four local units, four remote units, and four expander power supplies. If the function is not time critical, the circuits in Figures 3 and 4 are used.

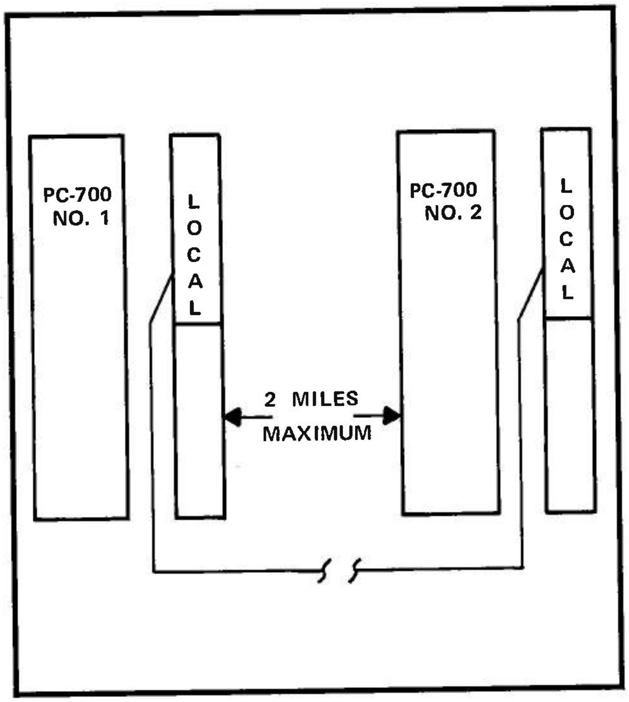
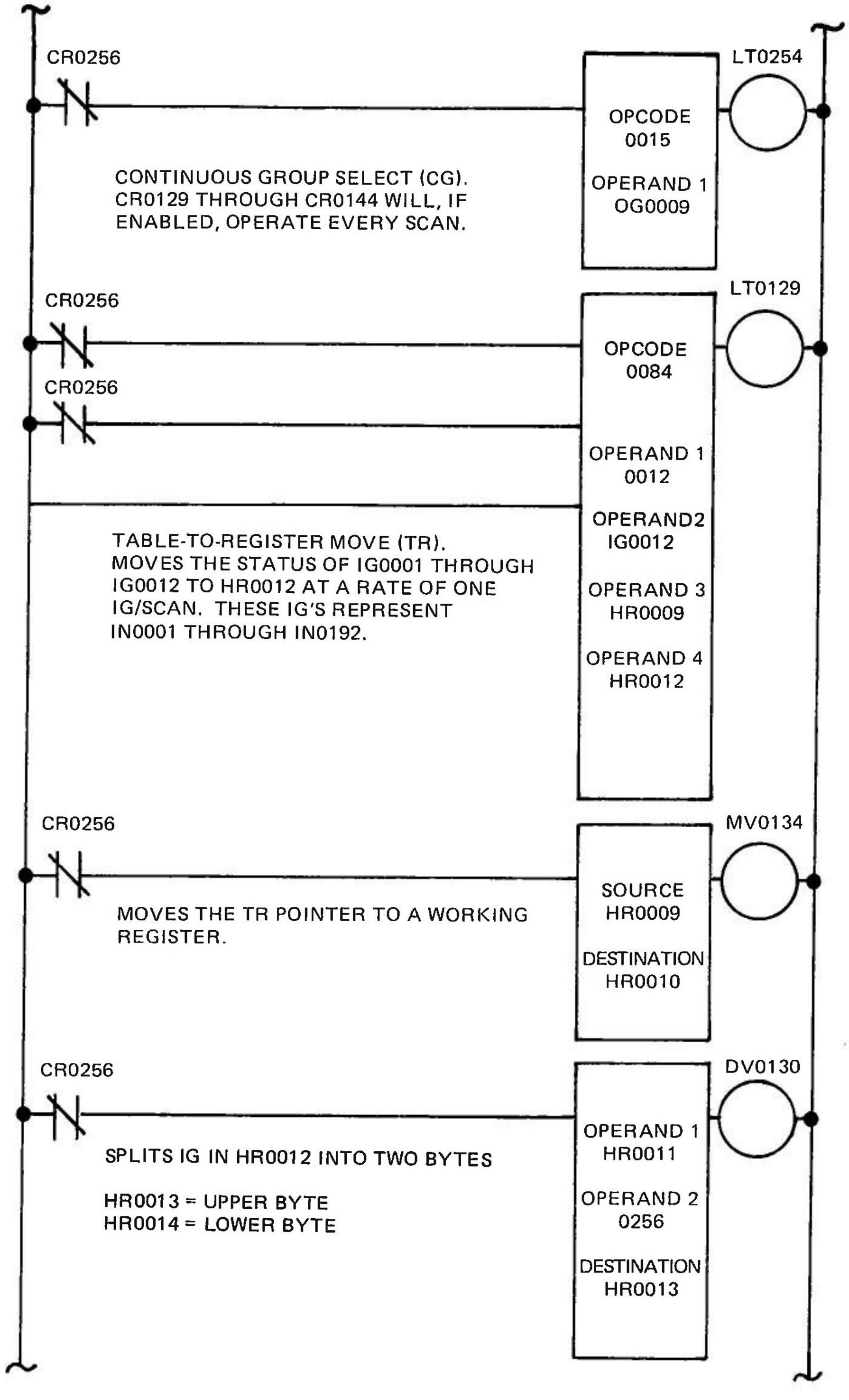


Figure 2. Interprocessor Data Exchange

Figure 3 is the program for PC No. 1. Using this program, PC No. 1 uses a Table-to-Register Move (TR) to move the status of IG0001 through IG0012 to a common register, from which it will be split into two bytes, each byte tagged and then sent to PC No. 2. This transfers IN0001 through IN0192. The Local Unit is set to Group 4 and is used to transfer IN0193 through IN0256 in the normal Local — Local fashion. The transfer rate is one IG/scan. Splitting the data and tagging it with the pointer ensures the integrity of the data at the receiving PC.

Figure 4 is the program for PC No. 2. Using this program, the data from PC No. 1 is reassembled and, using a Register-to-Table Move (RT), is stored for use by the PC. The two bytes, received as two separate words with a tag attached, are split back into untagged bytes while the tags are compared. If the tags are equal, then the Register-to-Table move is enabled and the received data is stored at the pointed location. The data transmitted will be faithfully reassembled in the receiving unit.

There are many options for this program and this example is intended to show a technique and possible application. These programs will faithfully transfer data, but may be too slow for some applications.



CONTINUOUS GROUP SELECT (CG). CR0129 THROUGH CR0144 WILL, IF ENABLED, OPERATE EVERY SCAN.

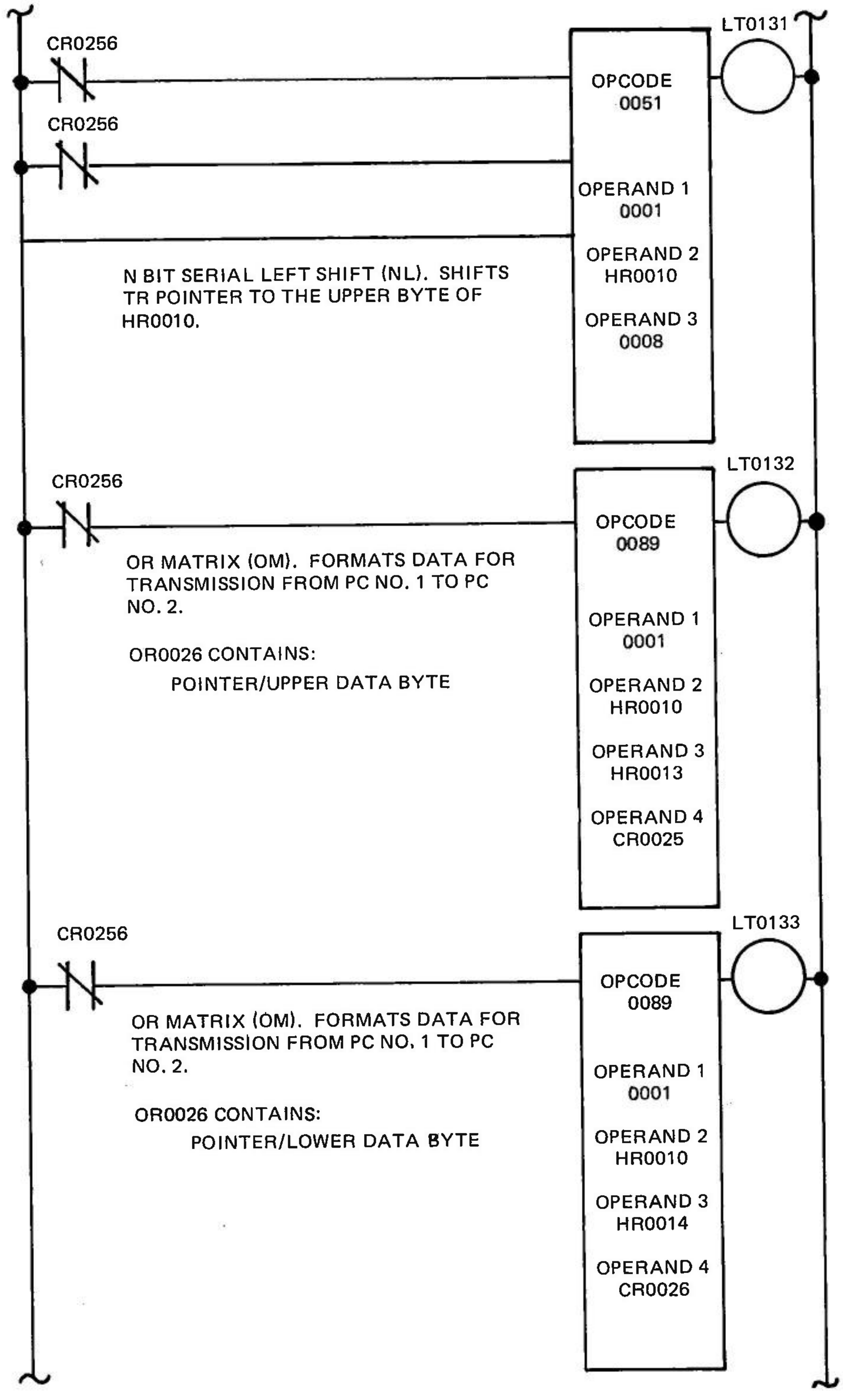
TABLE-TO-REGISTER MOVE (TR). MOVES THE STATUS OF IG0001 THROUGH IG0012 TO HR0012 AT A RATE OF ONE IG/SCAN. THESE IG'S REPRESENT IN0001 THROUGH IN0192.

MOVES THE TR POINTER TO A WORKING REGISTER.

SPLITS IG IN HR0012 INTO TWO BYTES

HR0013 = UPPER BYTE
HR0014 = LOWER BYTE

Figure 3a. PC No. 1 Program



N BIT SERIAL LEFT SHIFT (NL). SHIFTS TR POINTER TO THE UPPER BYTE OF HR0010.

OR MATRIX (OM). FORMATS DATA FOR TRANSMISSION FROM PC NO. 1 TO PC NO. 2.

OR0026 CONTAINS:

POINTER/UPPER DATA BYTE

OR MATRIX (OM). FORMATS DATA FOR TRANSMISSION FROM PC NO. 1 TO PC NO. 2.

OR0026 CONTAINS:

POINTER/LOWER DATA BYTE

Figure 3b. PC No. 1 Program (Cont'd)

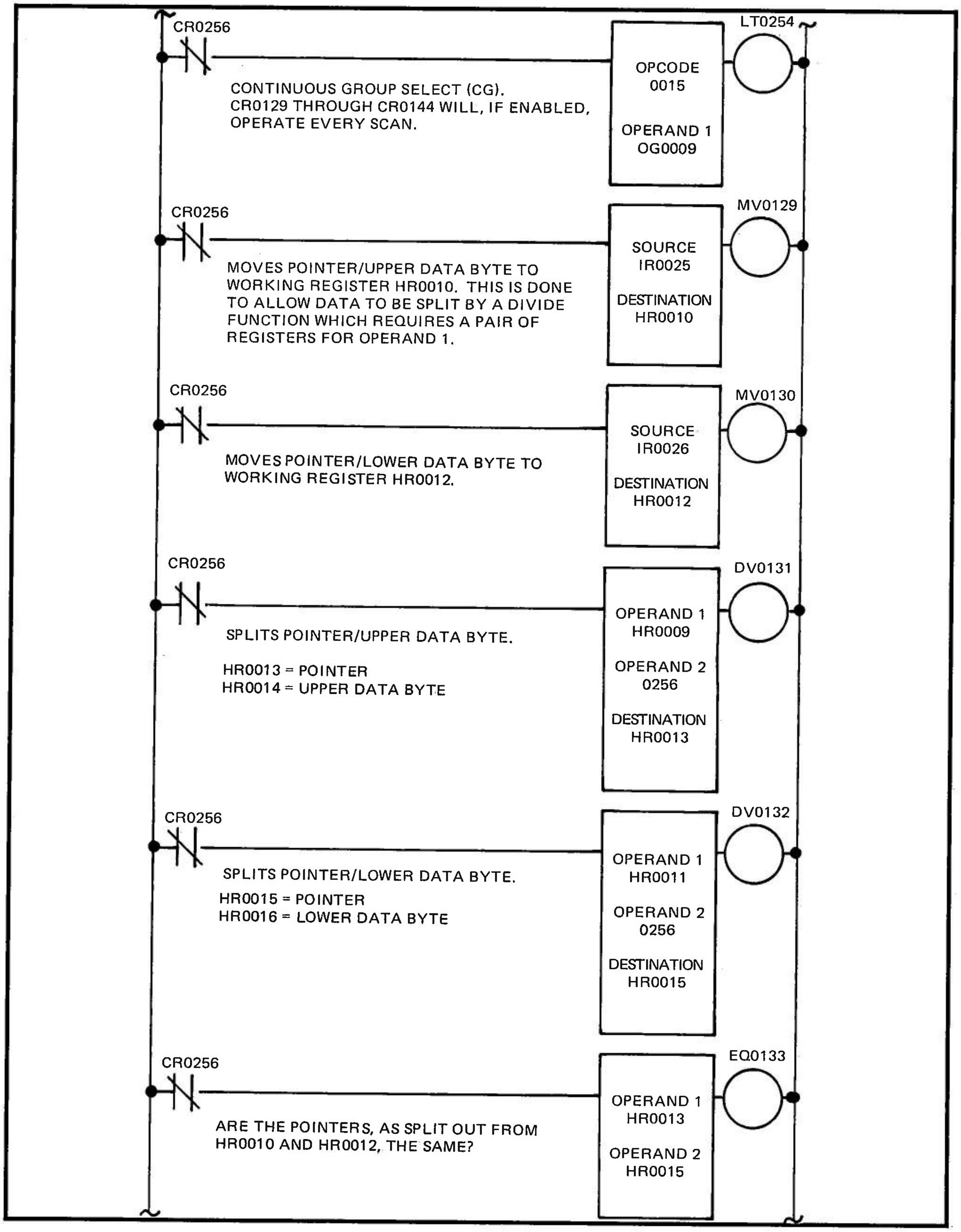
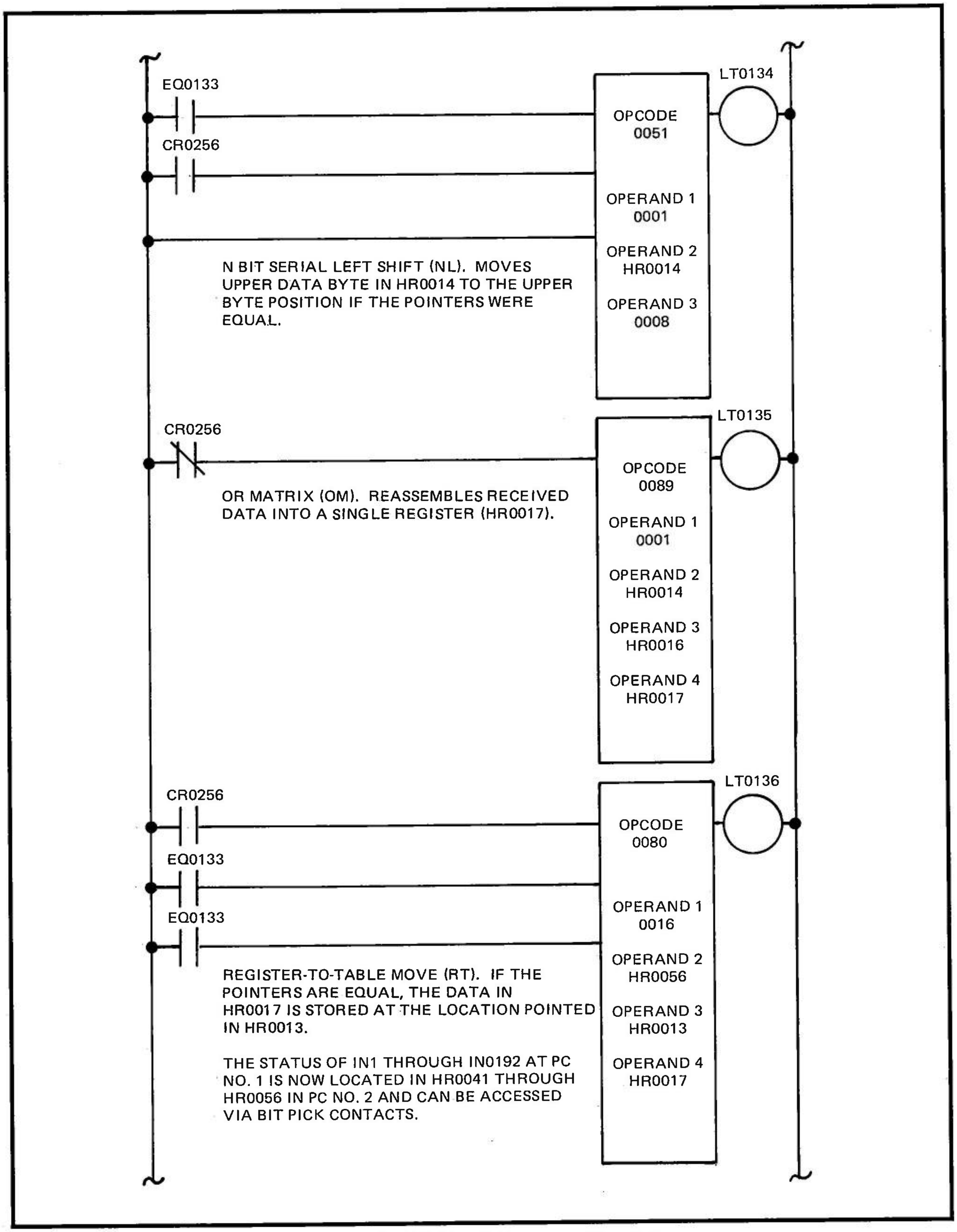


Figure 4a. PC No. 2 Program



N BIT SERIAL LEFT SHIFT (NL). MOVES UPPER DATA BYTE IN HR0014 TO THE UPPER BYTE POSITION IF THE POINTERS WERE EQUAL.

OR MATRIX (OM). REASSEMBLES RECEIVED DATA INTO A SINGLE REGISTER (HR0017).

REGISTER-TO-TABLE MOVE (RT). IF THE POINTERS ARE EQUAL, THE DATA IN HR0017 IS STORED AT THE LOCATION POINTED IN HR0013.

THE STATUS OF IN1 THROUGH IN0192 AT PC NO. 1 IS NOW LOCATED IN HR0041 THROUGH HR0056 IN PC NO. 2 AND CAN BE ACCESSED VIA BIT PICK CONTACTS.

Figure 4b. PC No. 2 Program (Cont'd)

TS/TT — TIMERS

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).

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.

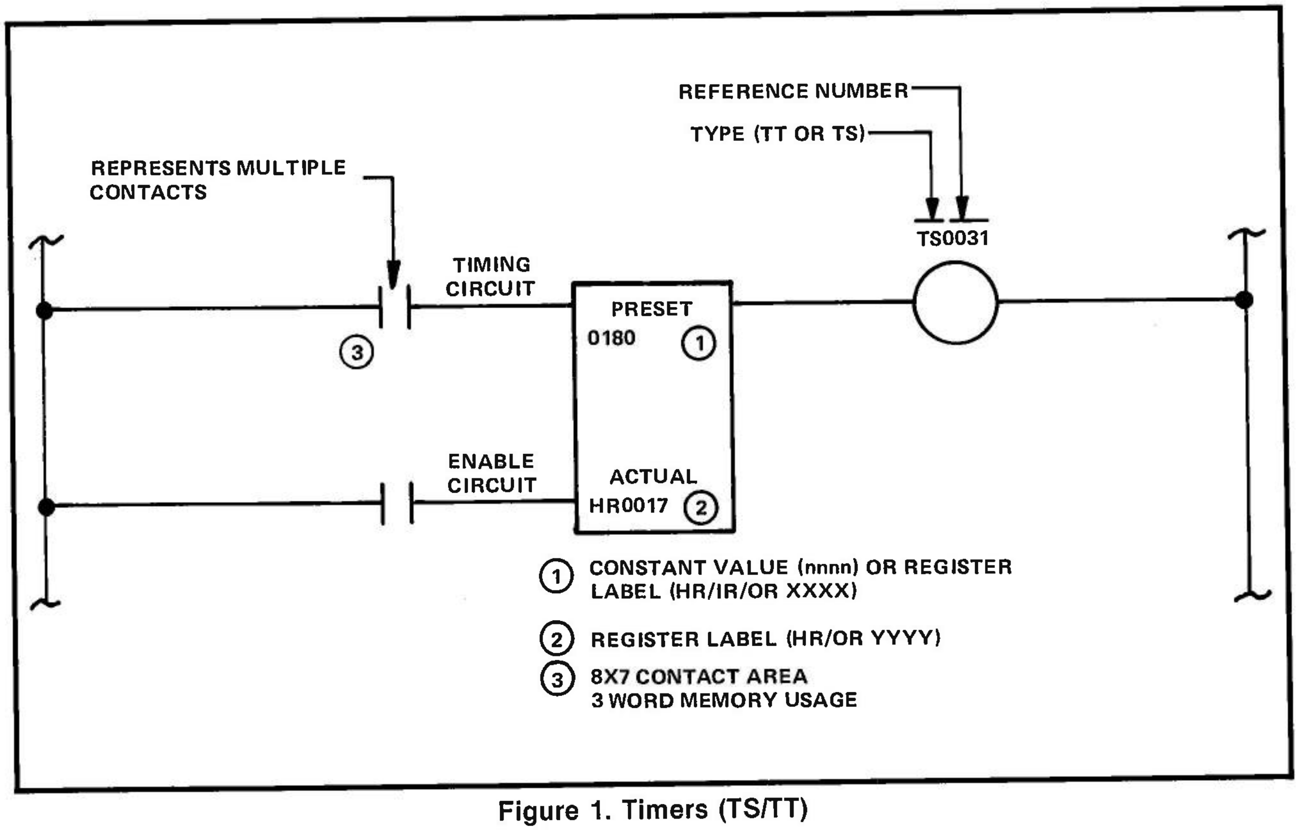


Figure 1. Timers (TS/TT)

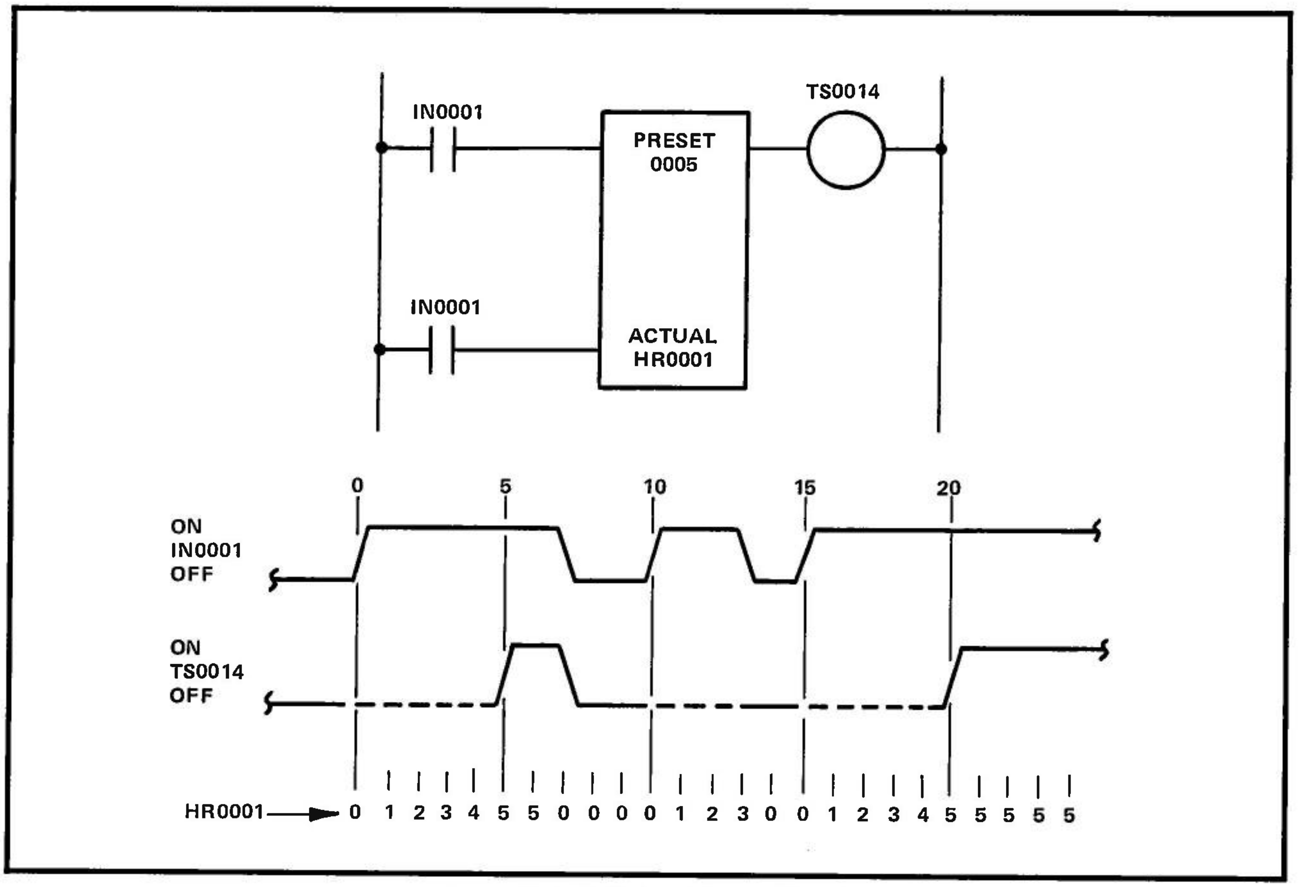


Figure 2. ON Delay Timer Circuit

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 TS0015 NC 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. CR0010 is then held energized by its own NO contacts and 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. De-energizing CR0010 causes a delayed OFF.

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).

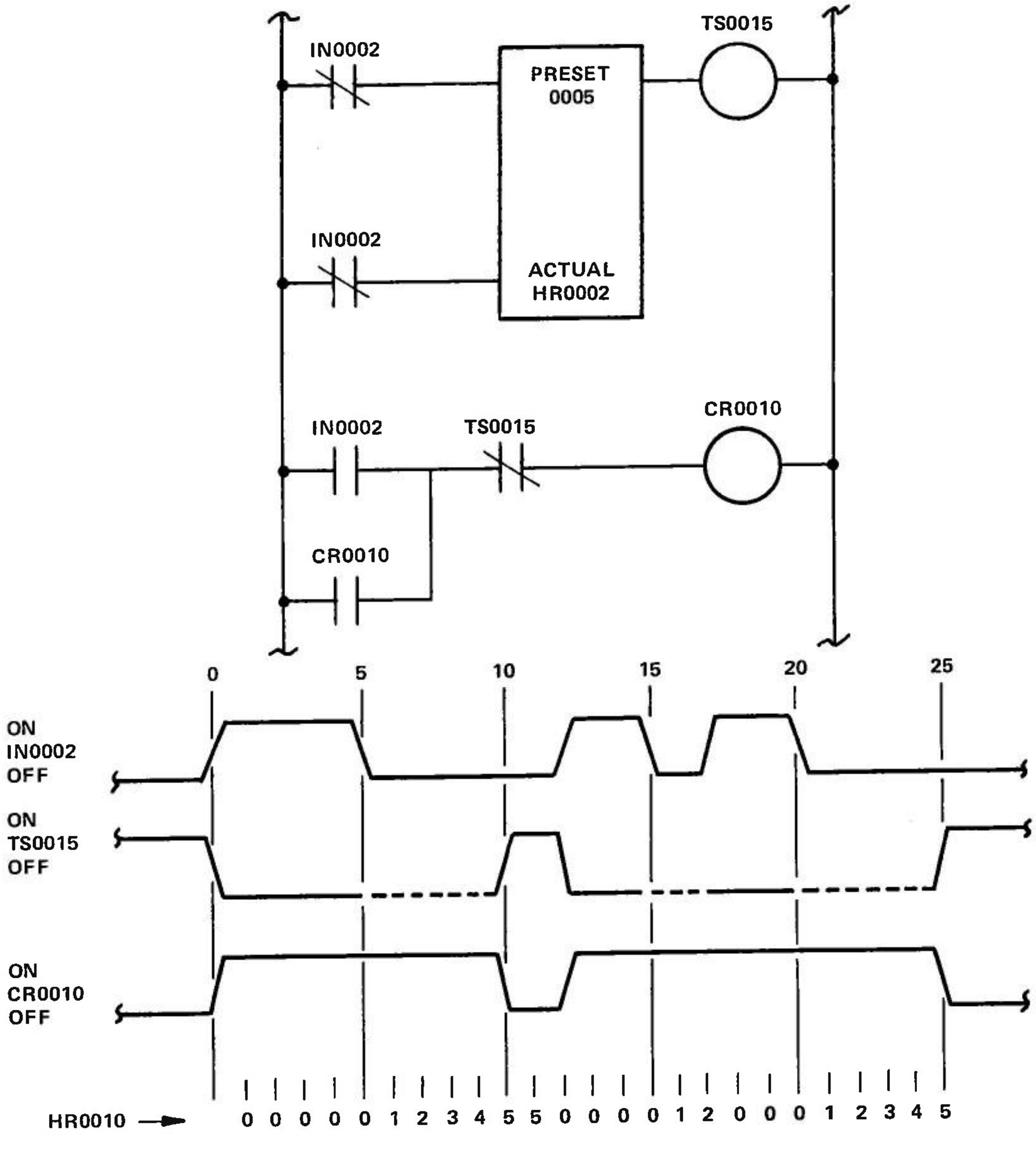


Figure 3. OFF Delay Timer Circuit

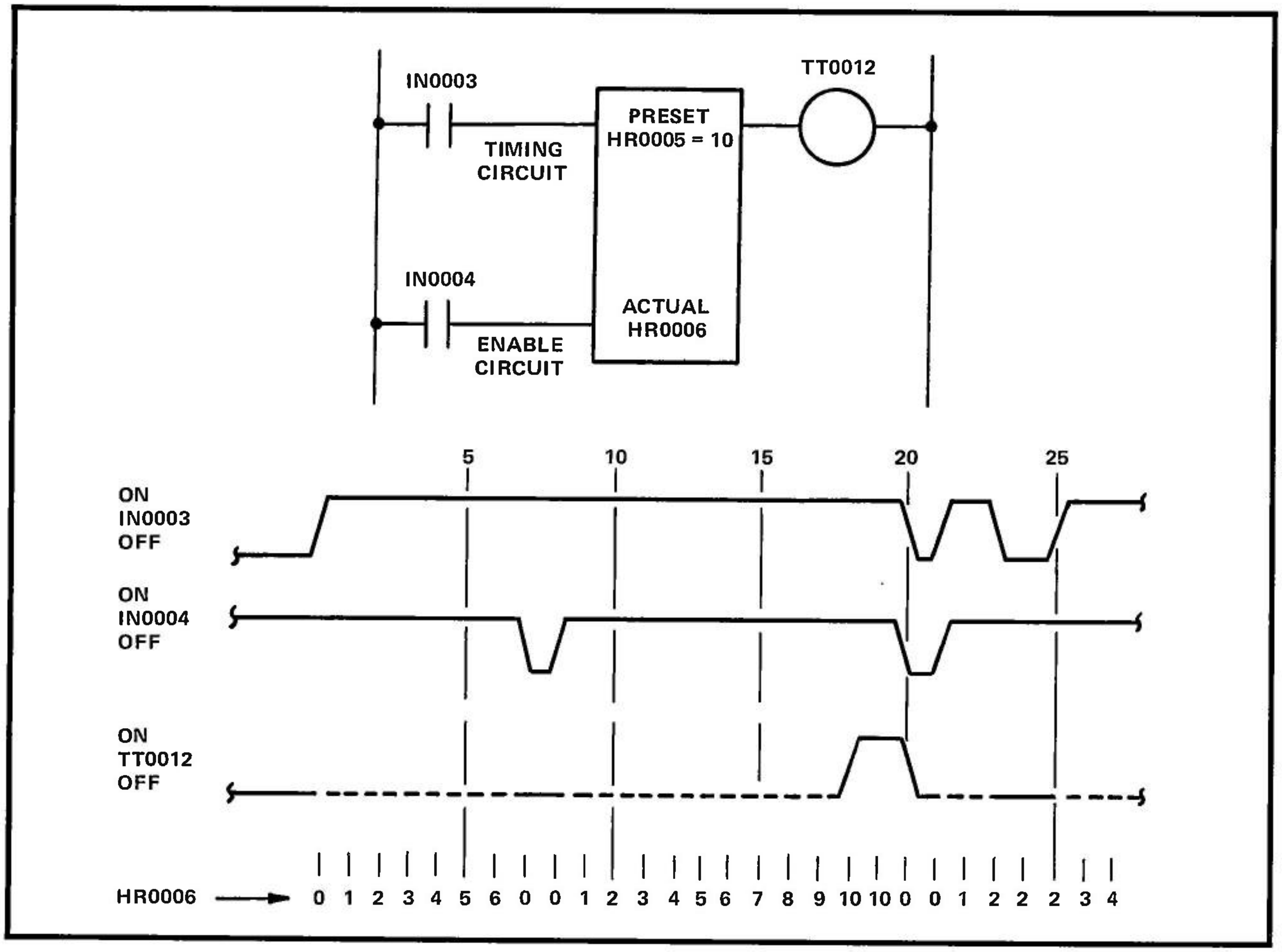


Figure 4. Motor Driven Timer

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).

COIL

The coil is energized when actual equals preset.

Note

> If preset is a register and is set to zero, actual equals preset at all times, and the coil is ON, regardless of the state of the enable circuit.

TS/TT TRUTH TABLE

See Table 1.

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,

TABLE 1. TS/TT TRUTH TABLE

Enable Circuit	Timing Circuit	Result
0	0	Actual is held to 0000.
0	1	Actual is held to 0000.
1	0	Actual is frozen at current value.
1	1	Actual accumulates until actual equals preset.

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.

UC/DC — COUNTERS

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 a loop 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. The counter continues past the preset value until 65535 is reached, and remains at that count until the enable circuit stops conducting.

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.

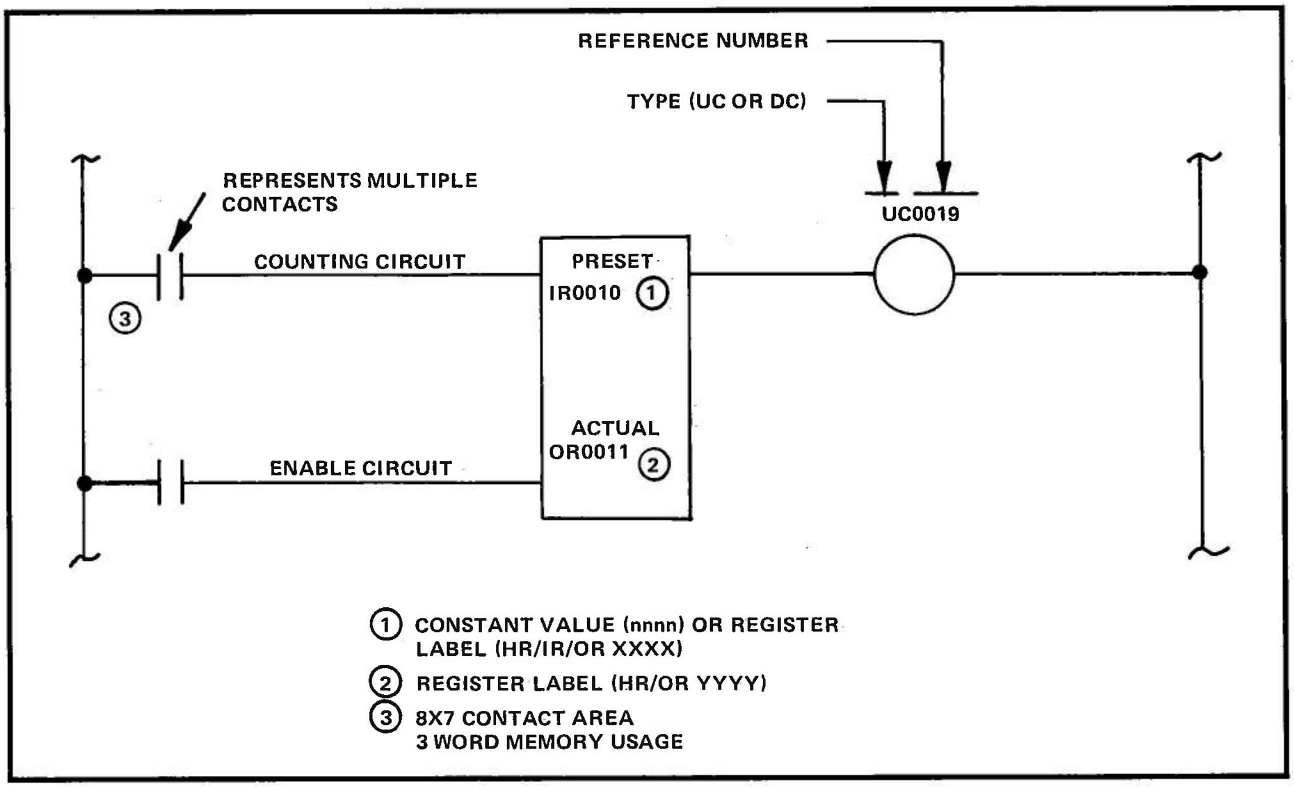


Figure 1. UP Counter (UC)/Down Counter (DC)