Bulletin: 30598-758-01A1

Programming Guide

Date: FEBRUARY 1990

Subject: ***SY/MAX***®

CLASS 8005 MODEL 50 PROGRAMMABLE CONTROLLER

This guide provides supplemental information when using the Class 8005 Type PR-4 Programmer in conjunction with a Model 50 Progammable Controllers System. For specific information on such topics as advanced instructions, error codes, and troubleshooting, refer to the Model 50 Instruction Bulletin 30598-142-XX. For specific information concerning the Class 8010 Type SFW-50 Programming Software, refer to the Instruction Bulletin 30598-666-XX.

NOTE: The Class 8010 SY/MAX Type SFW-50 programming software equipment can also be used to program the Class 8005 Model 50 Programmable Controller. Other Class 8010 hardware and software cannot be used.

SQUARE D COMPANY

Dedicated to Growth • Committed to Quality

WARNING

THE APPLICATION OF THIS PRODUCT REQUIRES EXPERTISE IN THE DESIGN AND PROGRAMMING OF CONTROL SYSTEMS. ONLY PERSONS WITH SUCH EXPERTISE SHOULD BE ALLOWED TO PROGRAM, INSTALL, ALTER, AND APPLY THIS PRODUCT. POTENTIAL BODILY INJURY, DEATH, OR EQUIPMENT DAMAGE COULD RESULT IF THE PRODUCT IS IMPROPERLY APPLIED TO ANY EQUIPEMNT APPLICATION.

NOTICE

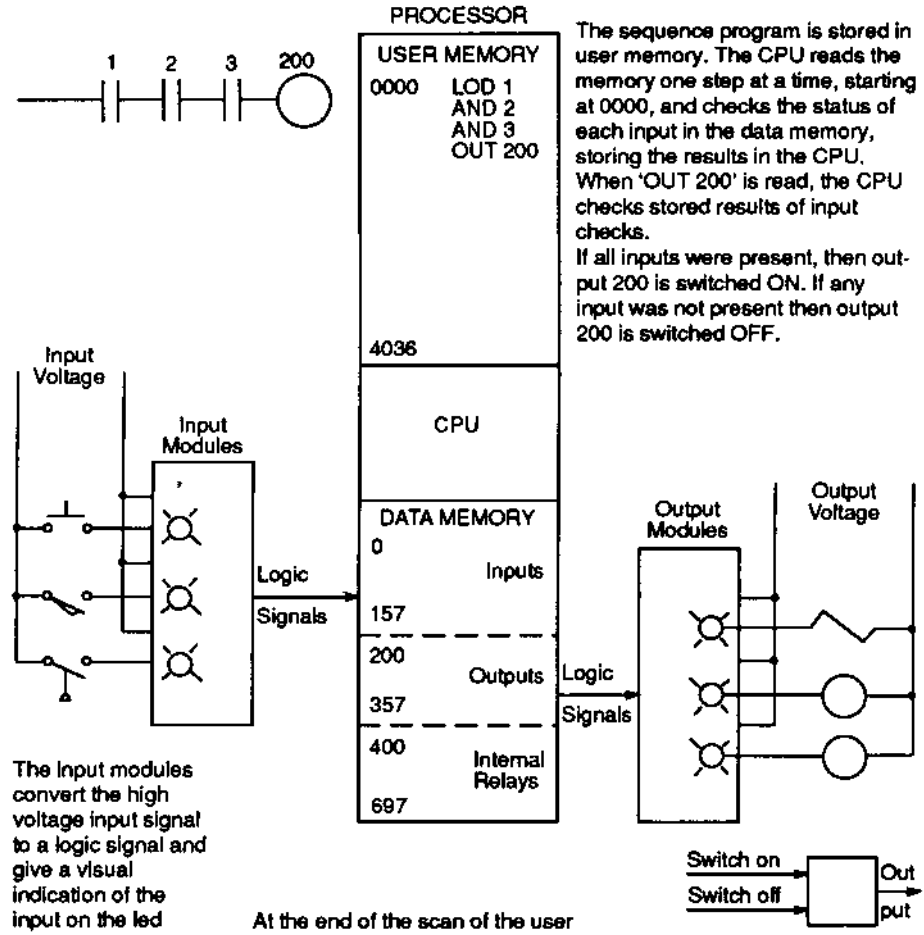
The products and services described in this manual are useful in a wide variety of different applications. Therefore, the user and others responsible for applying the products and services described herein are responsible for determining their acceptability for each application. While efforts have been made to provide accurate information within this mahual, the Square D Company assumes no responsibility for the application, completeness or usefulness of the information contained herein.

UNDER NO CIRCUMSTANCE WILL THE SQUARE D COMPANY OR ANY OF ITS SUBSIDIARIES BE RESPONSIBLE OR LIABLE FOR ANY DAMAGES OR LOSSES, INCLUDING INDIRECT OR CONSEQUENTIAL DAMAGES OR LOSSES, ARISING FROM EITHER THE USE OF ANY INFORMATION CONTAINED WITHIN THIS MANUAL OR THE USE OF ANY PRODUCTS OR SERVICES REFERENCED HEREIN.

No patent liability is assumed by the Square D Company with respect to the use of any of the information, products, circuits, programming or services referenced herein.

The information contained in this manual is subject to change without notice.

SV/MAX MODEL 50 PROGRAMMING GUIDE
TABLE OF CONTENTS



The sequence program is stored in user memory. The CPU reads the memory one step at a time, starting at 0000, and checks the status of each input in the data memory, storing the results in the CPU. When 'OUT 200' is read, the CPU checks stored results of input checks.
If all inputs were present, then output 200 is switched ON. If any input was not present then output 200 is switched OFF.

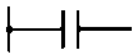
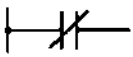
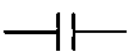
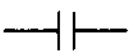
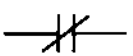
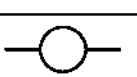
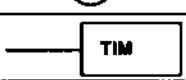
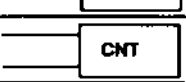
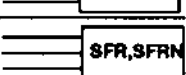
The input modules convert the high voltage input signal to a logic signal and give a visual indication of the input on the led indicator.

At the end of the scan of the user program, the status of the inputs is transmitted from the input modules to the data memory. Also the output status in the data memory is transmitted to the output modules.

Each output is controlled by a flip-flop which is switched ON or OFF by CPU.

DATA MEMORY CONFIGURATION

NAME	ALLOCATION NUMBERS		DESCRIPTION
Inputs	0 — 157		128 Inputs
Outputs	200 — 357		128 Outputs
Internal Relays	400 — 697		240 Internal Relays — A group of IR from 1 to 240 may be designated as retentive relays — See Function 6
Special Relays	700	Unused	
	701	Start control	
	702	Start control	
	703	All outputs off	
	704	Initialize pulse (turns on for one scan at start)	
	705	Unused	
	706	Numberical value error	
	707	(CY) Carry and Borrow	Math operations only
	710	Greater than (>) comparison	
	711	Equal to (=) comparison	
	712	Less than (<) comparison	
	713	1 - second timer reset	
	714	1 - second clock	
	715	100 millisecond clock	
	716	Timer/counter preset value changed	
	717	In - operation output	
Timer	0 — 79		0.1 — 999.9 second timer
Counter	0 — 44		0 — 999 counter
Reversible Counter	45		Dual pulse (up and down counts)
Reversible Counter	46		Single pulse (up/down selection)
Shift Register	0 — 127		128 Bit (bi-directional)
Single Output	0 — 95		96 Outputs
Data Register	800 — 899		100 Registers

INSTRUCTION	LADDER SYMBOL	NO. OF STEPS	FUNCTION
LOD		1	Store new rung (or branch) with N.O. contact
LOD NOT		1	Store new rung (or branch) with N.C. contact
AND		1	N.O. contact in series
AND NOT		1	N.C. contact in series
OR		1	N.O. contact in parallel
OR NOT		1	N.C. contact in parallel
AND LOD		1	Series connection with previously stored result
OR LOD		1	Parallel connection with previously stored result
OUT		1	Output
TIM		2	Timer
CNT		2	Counter
SFR, SFR NOT		2	Shift Register - forward Shift Register - reverse
MCS		1	Start of Master Control
MCR		1	End of Master Control
SOT		1	Off to on transitional output
SET		1	Sets an output, internal relay, or shift register
RST		1	Resets an output, internal relay, or shift register
JMP		1	Jumps a designated program area
JEND		1	Ends a jump program
END		1	Ends a program

PROGRAMMING PROCEDURE

A Model 50 processor, input module, output module and a PR-4 programmer can be used, with the input module connected to switches to simulate the inputs. Module output status indicators on the module can be used to indicate an output.

PROCEDURE with the PR-4 Programmer

1. Set processor to "HALT" via "HALT/RUN" switch (Input 0) or the PR-4 Programmer.

2. Set programmer switch to "PROGRAM".
3. Clear programmer memory and then processor memory by using following routines:-

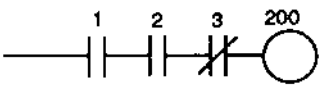
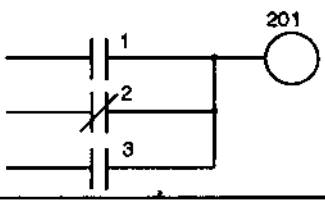
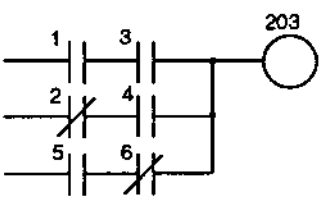
4. Program the relay circuit examples shown on pages 5 and 6.
5. Transfer program from programmer to RAM memory in processor:-

6. Set the processor to "RUN" via the "HALT/RUN" switch (Input 0) or the PR-4 programmer.

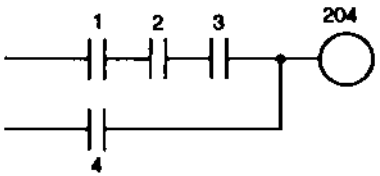
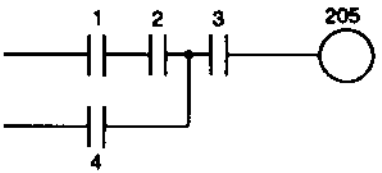
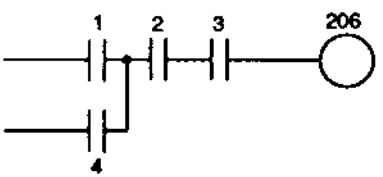
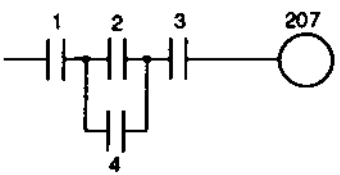
7. Check operation of program by operating switches and observing status LEDS on input and output modules.
8. Use monitor routines to display input/output status on programmer. See pages 24 to 29.
9. Procedure 1 to 8 can then be repeated for the examples of timers, counters, shift registers etc, on pages 11 to 26.

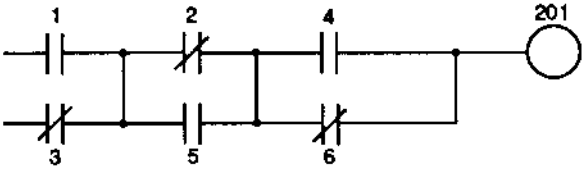
<u>Note</u>

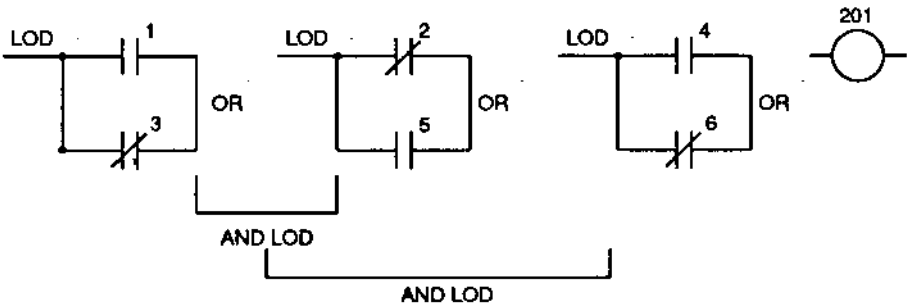
When adding a new program to an existing program in memory, ALWAYS search for next empty address in memory, (ie, the end of the existing program) using the following key routine:-

Ladder Diagram	Program Entry
Contacts in series  	LOD 1 ENTR AND 2 " AND NOT 3 " OUT 200 "
Contacts in parallel  	LOD 1 ENTR OR NOT 2 " OR 3 " OUT 201 "
Contracts in series/parallel  	LOD 1 ENTR OR NOT 5 " LOD NOT 3 " OR 4 " AND LOD " LOD 2 " OR NOT 6 " AND LOD " OUT 202 "
Contacts in series/parallel  	LOD 1 ENTR AND 3 " LOD NOT2 " AND 4 " OR LOD " LOD 5 " AND NOT 6 " OR LOD " OUT 203 "

RELAY CIRCUIT EXAMPLES

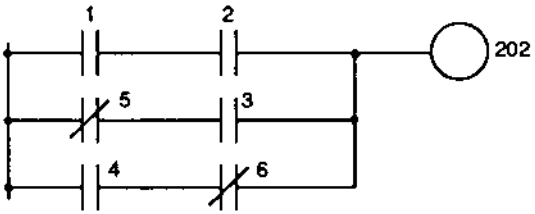
Ladder Diagram	Program Entry	
	LOD 1 AND 2 AND 3 OR 4 OUT 204	ENTR " " " "
	LOD 1 AND 2 OR 4 AND 3 OUT 205	ENTR " " " "
	LOD 1 OR 4 AND 2 AND 3 OUT 206	ENTR " " " "
	LOD 1 LOD 2 OR 4 AND LOD AND 3 OUT 207	ENTR " " " " "

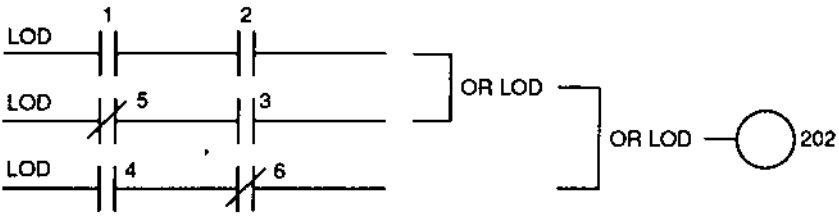




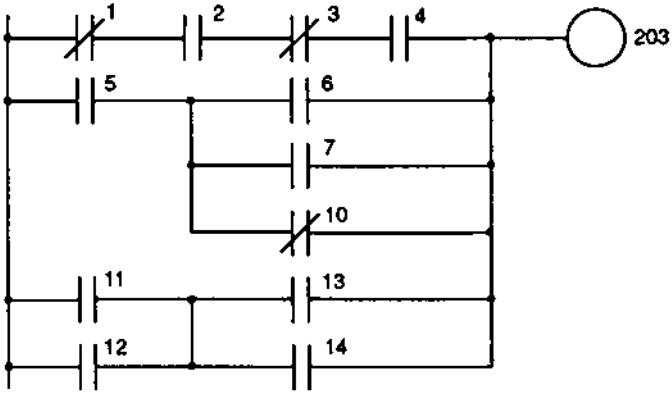
Program Entry	Explanation of function
LOD 1	Store new rung with N.O. contact 1.
OR NOT 3	N.C. contact 3 in parallel.
LOD NOT 2	Store new branch with N.C. contact 2.
OR 5	N.O. contact 5 in parallel.
AND LOD	Series connection with previously stored result.
LOD 4	Store new branch with N.O. contact 4.
OR NOT 6	N.C. contact 6 in parallel.
AND LOD	Series connection with previously stored result.
OUT 201	Output 201.

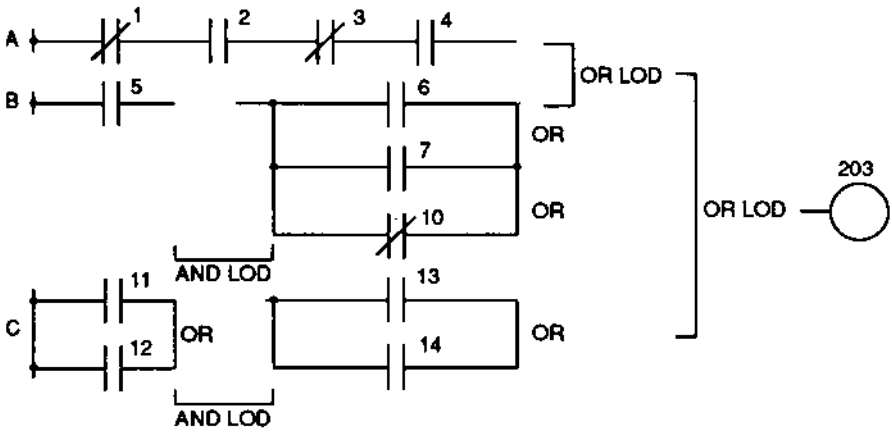
EXAMPLE OF USE OF "OR LOD" INSTRUCTION





Program entry	Explanation of function
LOD 1	Store new rung with N.O.contact 1.
AND 2	N.O. contact 2 in series.
LOD NOT 5	Store new branch with N.C. contact 5.
AND 3	N.O. contact 3 in series.
OR LOD	Parallel connection with previously stored result.
LOD 4	Store new branch with N.O. contact 4.
AND NOT 6	N.C. contact 6 in series.
OR LOD	Parallel connection with previously stored result.
OUT 202	Output 202.





The circuit can be divided into three parallel branches A, B and C. Each of these branches is programmed in turn and the "OR LOD" instruction used to link them together.

(Continued on next page).

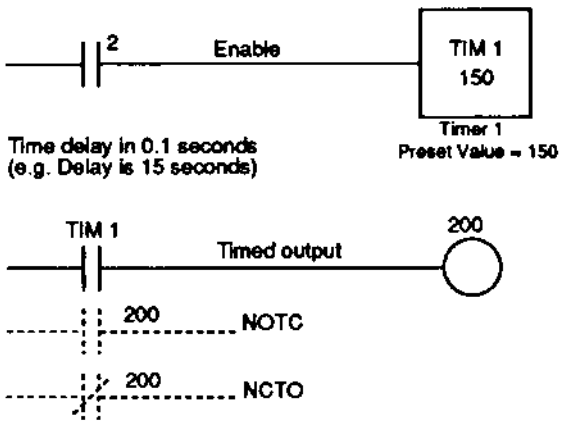
EXAMPLE OF COMPLEX RUNG PROGRAM

Continued from page 9

	Program entry	Explanation of function
A	LOD NOT 1	Store new rung with N.C. contact 1.
	AND 2	N.O. contact 2 in series.
	AND NOT 3	N.C. contact 3 in series.
	AND 4	N.O. contact 4 in series.
B	LOD 5	Store new branch with N.O. contact 5.
	LOD 6	Store new branch with N.O. contact 6.
	OR 7	N.O. contact 7 in parallel.
	OR NOT 10	N.C. contact 10 in parallel.
	AND LOD	Series connection with previously stored result.
C	OR LOD	Parallel connection with previously stored result.
	LOD 11	Store new branch with N.O. contact 11.
	OR 12	N.O. contact 12 in parallel.
	LOD 13	Store new branch with N.O. contact 13.
	OR 14	N.O. contact 14 in parallel.
	AND LOD	Series connection with previously stored result.
	OR LOD	Parallel connection with previously stored result.
	OUT 203	Output 203.

Time Delay 0.1 to 999.9 seconds	**Note.** This timer resets if A.C. power supply is interrupted.

Ladder Diagram

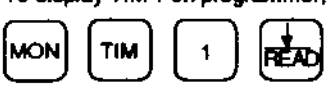


Program Entry

LOD 2	ENTR
TIM 1	"
150	"
LOD TIM 1	ENTR
OUT 200	"

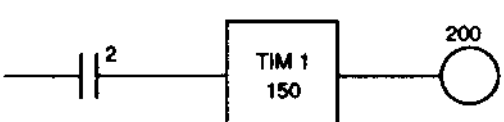
When contact 2 opens the number 150 is preset into TIM 1.
When contact 2 closes, the timing starts, and the number in TIM 1 reduces by one every 0.1 second until TIM 1 contents are zero. Output 200 will then energize.

To display TIM 1 on programmer, operate following keys:-



The programmer displays the remaining time of the delay during timing.

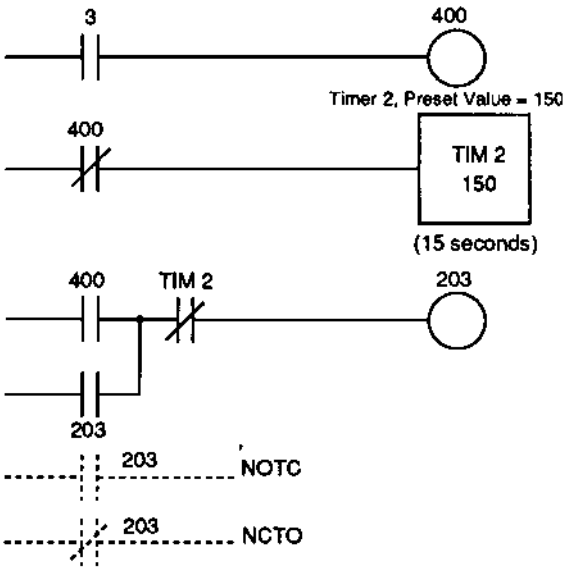
The above circuit may also be programmed as follows:-



LOD 2	ENTR
TIM 1	"
150	"
OUT 200	"

TIMING CIRCUITS - TIME DELAY AFTER DE-ENERGIZATION

Time Delay
0.1 to 999.9 seconds

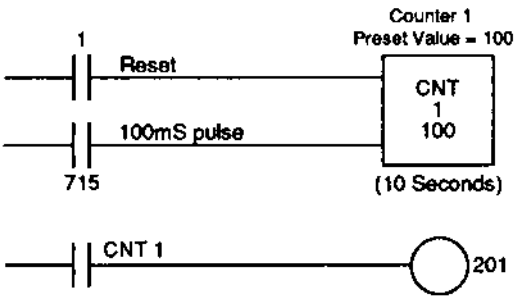
Ladder Diagram	Program Entry
	LOD 3 ENTR OUT 400 " LOD NOT 400 ENTR TIM 2 " 150 " LOD 400 ENTR OR 203 " AND NOT TIM 2 " OUT 203 "

When N.O. contact 3 closes, N.C. contact 400 opens and presets the number 150 into TIM 2. Also N.O. contact 400 closes energizing output 203 via N.C. contact TIM 2. When contact 3 opens, N.C. contact 400 closes starting the time delay. N.O. contact 400 opens but output 203 is maintained via N.O. contact 203.
When the timer times out, N.C. TIM 2 contact opens, de-energizing output 203.

Note
The internal relay 400 must be designated as a "Retentive type "relay" -See Function 6 (FUN 6) - page 35.

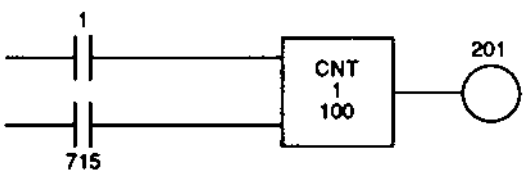
TIMING CIRCUIT - TIME DELAY AFTER ENERGIZATION WITH RETENTIVE COUNT ON POWER FAILURE

Time Delays
0.1 to 999.9 seconds (715 pulse)
1 to 9999 seconds (714 pulse)

Ladder Diagram	Program Entry
	LOD 1 ENTR LOD 715 " CNT 1 " 100 " LOD CNT 1 ENTR OUT 201 "

When contact 1 is closed the counter is reset and output 201 is off.
When contact 1 is open, the counter counts the 100ms pulses. When the count value equals the preset value, the output turns on.

The above circuit may also be programmed as follows:-

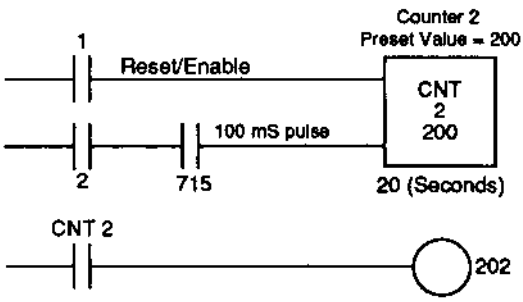


LOD 1	ENTR
LOD 715	"
CNT 1	"
100	"
OUT 201	"

Note
Counter 1 must be designated as a "Retentive type counter" - See Function 7 (FUN 7) - page 36.

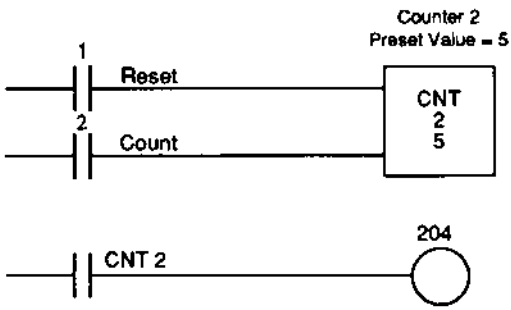
TIMING CIRCUITS - INTERRUPTIBLE TIMER (TIMER WITH HOLD)

Time Delays
0.1 to 999.9 seconds (715 pulse)
1 to 9999 seconds (714 pulse)

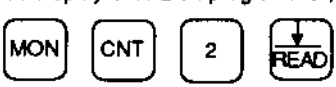
Ladder Diagram	Program Entry
	LOD 1 ENTR LOD 2 " AND 715 " CNT 2 " 200 " LOD CNT 2 ENTR OUT 202 "

When contact 1 closes, the counter is reset.
When contact 1 opens, the counter is enabled.
When contact 2 closes, the timing starts.
If contact 2 opens, the time is "held".
When counter 2 closes, the timing resumes from the "held" time.
When the timer times out, output 202 is energized.

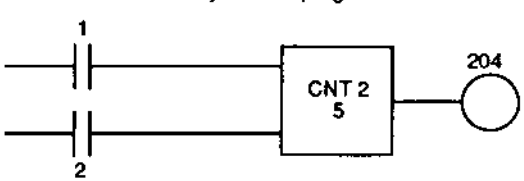
Preset values
0 to 9999

Ladder Diagram	Program Entry
	LOD 1 ENTR LOD 2 " CNT 2 " 5 " LOD CNT 2 ENTR OUT 204 "

When contact 1 is closed, the counter is reset.
When contact 1 is open, the counter is enabled.
Each time contact 2 closes, the counter increments by one.
When the counted value equals the preset value, the output 204 is energized.
To display CNT 2 on programmer, operate the following keys:-



The above circuit may also be programmed as follows:-

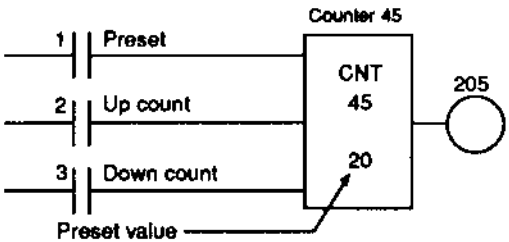


LOD 1 ENTR
LOD 2 "
CNT 2 "
5 "
OUT 204 "

REVERSIBLE COUNTER WITH UP AND DOWN PULSE INPUTS

Note. Only counter No. 45 may be used for this function	Preset values 0 to 9999

Ladder Diagram



Program Entry

LOD 1	ENTR
LOD 2	"
LOD 3	"
CNT 45	"
20	"
OUT 205	"

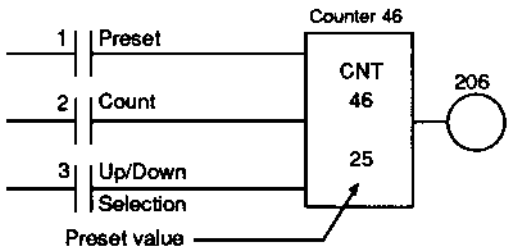
When contact 1 closes, the counter is preset to 20.
When contact 1 opens, the counter is enabled.
Each time contact 2 closes, the counter counts up by one.
Each time contact 3 closes, the counter counts down by one.

Note. The count starts from the preset value.

When counting up, the count will go 9999 → 0 → 1.
When counting down, the count will go 1 → 0 → 9999.
If the up and down pulses are on simultaneously, the counter may miss a count pulse.

Note. Only counter No. 46 may be used for this function	Preset values 0 to 9999

Ladder Diagram



Program Entry

LOD 1	ENTR
LOD 2	"
LOD 3	"
CNT 46	"
25	"
OUT 206	"

When contact 1 closes, the counter is preset to 25.
When contact 1 is open, the counter is enabled.
When contact 3 closes, the "UP" mode is selected.
When contact 3 is open, the "DOWN" mode is selected.
Each time contact 2,closes, the counter counts up or down by one.

Note The count starts from the preset value.

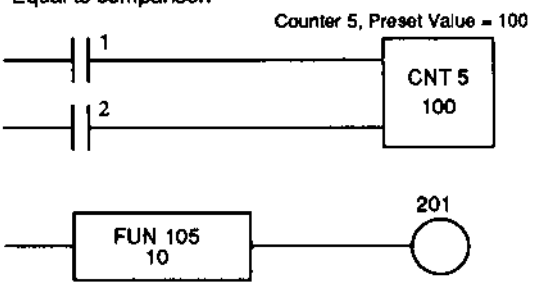
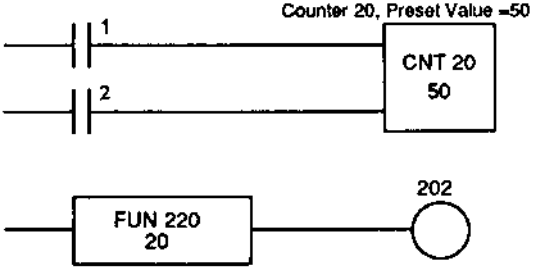
When the count value is zero, output 206 is energized.

When counting "UP" the count will go 9999 ⟶ 0 ⟶ 1
When counting "DOWN" the count will go 1 ⟶ 0 ⟶ 9999

COUNTER EQUAL TO AND EQUAL TO OR GREATER THAN COMPARISON INSTRUCTIONS

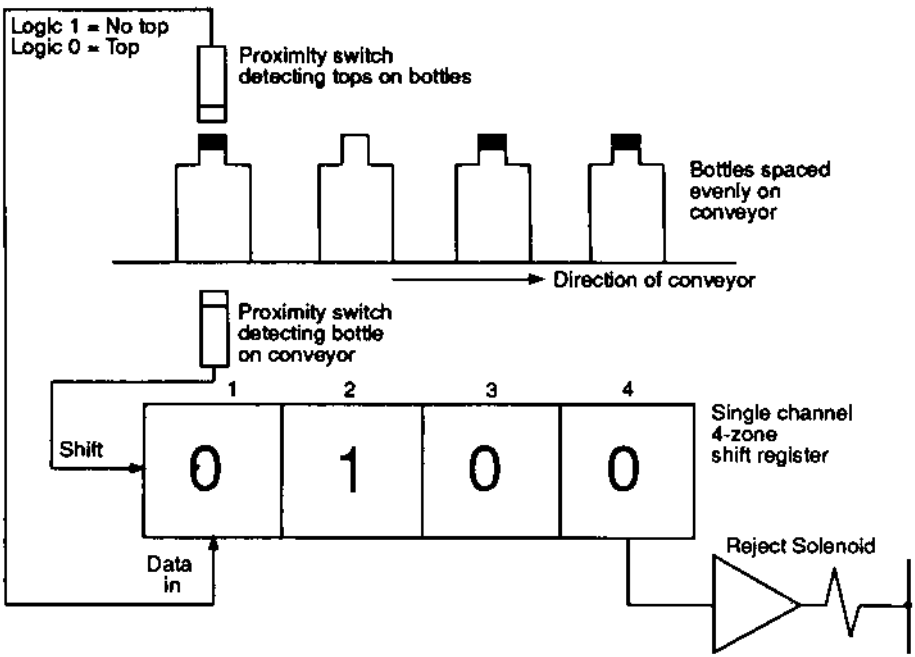
Counters 0 to 46 can perform the above comparisons by using following instructions:-

Equal to comparison:
FUN 100 to FUN 146 - Equivalent to counters 0 to 46.
Equal to or greater than comparison:
FUN 200 to FUN 246 - Equivalent to counters 0 to 46.

Ladder Diagram	Program Entry
Equal to comparison 	LOD 1 ENTR LOD 2 " CNT 5 " 100 " FUN 105 " 10 " OUT 201 "
Equal to or greater than comparison 	LOD 1 ENTR LOD 2 " CNT 20 " 50 " FUN 220 " 20 " OUT 202 "

The same "FUN" number can be used repeatedly for different preset values.

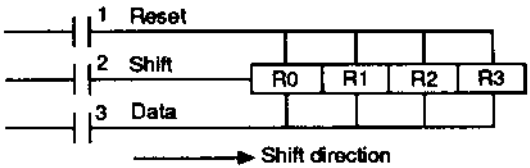
Example: Bottling Line



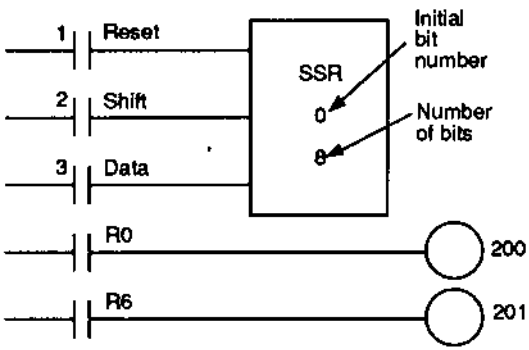
The shift register is an electronic model of what is happening on the conveyor. As each bottle moves over proximity switch, previous data is all shifted one zone to the right and either a logic 1 or a logic 0 is shifted into the first zone.
When the bottle with no top has moved to the fourth position on the conveyor, the logic 1 has moved to the fourth zone in the shift register where it operates the reject solenoid and removes the bottle from the conveyor.

FORWARD SHIFT REGISTER

Block Diagram



Ladder Diagram



Program Entry

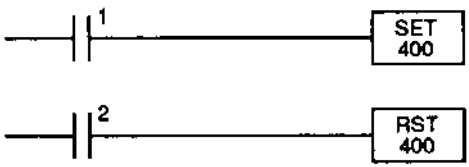
LOD 1	ENTR
LOD 2	"
LOD 3	"
SFR 0	"
8	"
LOD R0	"
OUT 200	"
LOD R6	"
OUT 201	"

When contact 1 is closed the shift register is reset. When contact 1 is open the shift register is enabled.
Each time contact 2 closes, the data in the shift register moves one bit position to the right and new data moves into bit 0.
If contact 3 is closed, when shift contact 2 closes, then a logic 1 is entered. If contact 3 is open, then a logic 0.

The above shift register bits may be monitored by operating following keys:-

MON SFR 0 READ

Note. Only internal relays can be used to give equivalent of mechanically latched relay in following circuit. Also Internal relays must be designated as "retentive type relays" - See Function 6 (FUN 6) page 35.

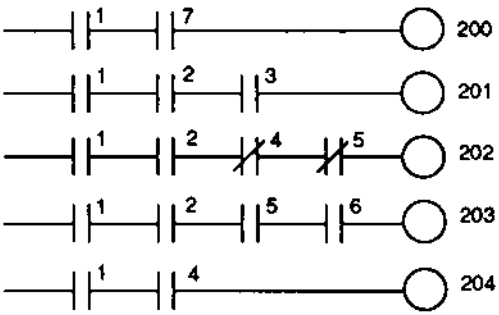
Ladder Diagram	Program Entry
	LOD 1 ENTRY SET 400 " LOD 2 " RST 400 "

If contact 1 closes momentarily, internal relay 400 will switch on.
If contact 2 closes momentarily, internal relay 400 will switch off.
If the power is interrupted, Internal relay 400 will "remember" whether it was switched on or switched off at the time of power interruption and will resume that condition when power is restored.

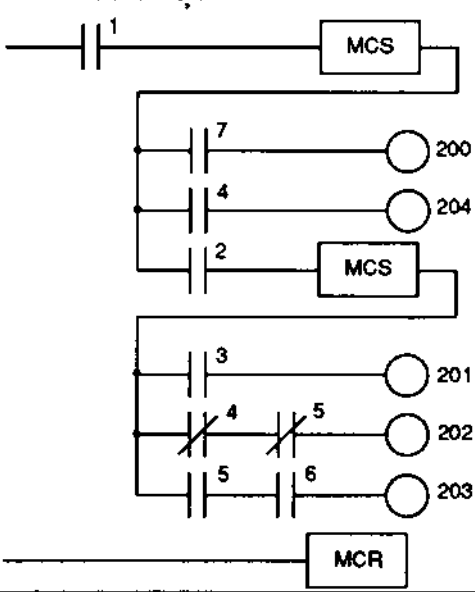
MASTER CONTROL RELAY - USING MCS AND MCR INSTRUCTIONS

Ladder Diagram

Without MCS and MCR



With MCS and MCR

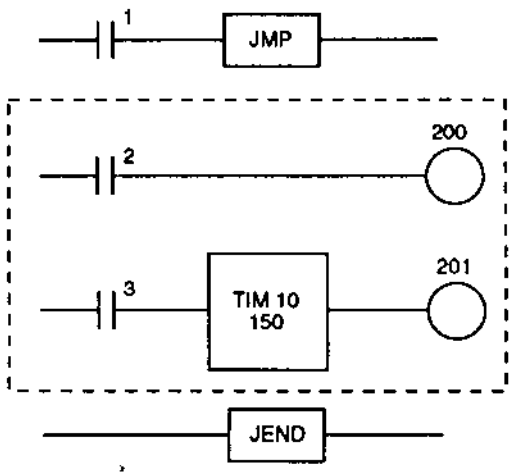


Program Entry

LOD 1	ENTR
MCS	*
LOD 7	*
OUT 200	*
LOD 4	*
OUT 204	*
LOD 2	*
MCS	*
LOD 3	*
OUT 201	*
LOD NOT 4	*
AND NOT 5	*
OUT 202	*
LOD 5	*
AND 6	*
OUT 203	*
MCR	*

Note

If the input to the MCS is off, all inputs following the MCS are forced off until the MCR instruction is executed.

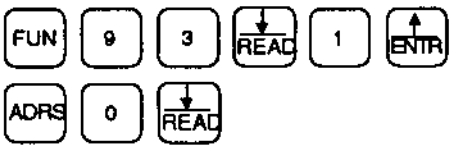
Ladder Diagram	Program Entry
	LOD 1 ENTR JMP * LOD 2 * OUT 200 * LOD 3 * TIM 10 * 150 * OUT 201 * JEND *

When contact 1 is open, the circuit within the dotted lines operates normally.
After contact 1 is closed, all outputs will remain in the state they were in at the time contact 1 closed and any input contact changes will be ignored.
The timed value in the timer will also be held at the value it was when contact 1 closed.

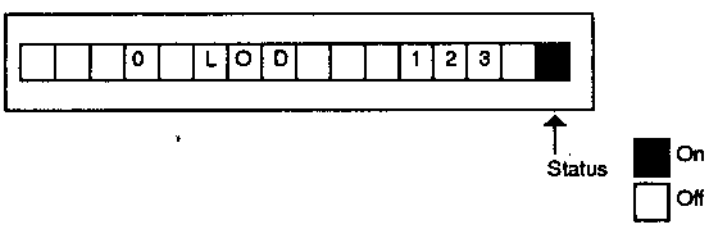
SEQUENTIAL MONITORING ROUTINE

The ON/OFF status of inputs/outputs/internal relays/timers/counters and shift register bits can be monitored by using the following routine.

Key Sequence:-

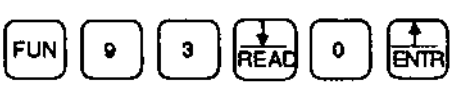


Display will show the address '0' instruction and status



Where a 'NOT' instruction is displayed, the status is reversed.

Note
Once this mode has been selected it can only be cancelled by interrupting the A.C. power to the processor or by using the following key sequence:-



Search for an address in user memory

e.g. search for address 4. Operate keys:-

This search routine will find the user memory address and display the instruction at that address.

The 

and 

keys

may then be used to move from specified memory address to higher or lower addresses.

User Memory

Address	Program
0	LOD 1
1	AND 4
2	AND 6
3	OUT 200
4	LOD 7
5	AND 5
6	OUT 202
7	LOD 2
4036	END

Note

If the sequential monitoring function, (FUN 93) - see page 24, has already been selected, then the I/O, internal relay, timer, counter and shift register bit status will also be displayed.

Search for input/output/TIM/CNT/SFR/special relay in program

e.g. search for "LOD 201"
operate keys:-

(This sets user memory at address 0)
Then operate keys:-

This search routine will find the first "LOD 201" in the program.

Each of the remaining "LOD 201" entries in the program may be found by repeating the routine:-

User Memory

Address	Program
0	LOD 1
1	AND 2
2	OUT 200
3	LOD 201
4	AND 4
5	AND 5
6	OUT 203
7	LOD 201
8	OR 6
9	OR 7
10	OUT 210
11	LOD 201
4036	END

When no more entries can be located, programmer will emit double 'beep'.

Note
If the sequential monitoring function, (FUN 93) - see page 24, has already been selected, then the I/O, internal relay, timer, counter and shift register bit status will be displayed.

Search for next empty address in memory.

Note
This key sequence is required whenever additional program is to be added to an existing program:-

Existing program

Address	Program
0	LOD 1
1	AND 4
2	AND 6
3	OUT 200
4	LOD 7
5	AND 5
6	OUT 202
7	END
8	END
	.
	.
	.
	.
	.
4036	

In this example the first instruction of the additional piece of program will be entered at memory address 7.

Note **IF THIS ROUTINE IS NOT FOLLOWED, THE ADDITIONAL PROGRAM MAY OVERWRITE THE EXISTING PROGRAM**

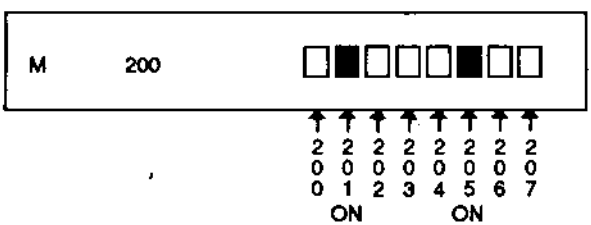
I/O AND INTERNAL RELAY MONITORING ROUTINE

Monitored status is displayed in units of 8 points starting with the designated number.
Monitor display is as follows:-

■ 'ON' display □ 'OFF' display

e.g. To monitor output 200 operate keys:-

Display will show:-



To display next 8 outputs operate 

Notes

1. Monitored data is updated and displayed every 100 msec.
2. Programmer can be in either 'Program' or 'Lock' mode when monitoring.

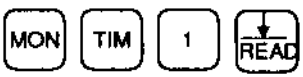
The above notes apply to all monitoring routines on the Model 50.

The monitor display shows timer or counter number, counted/timed value and on/off status. For both counter and timer, the On/Off status display is as follows:-

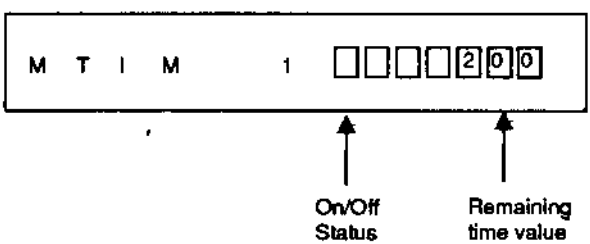
■ On (Timed out or count up)

□ Off (Timing or counting)

e.g. To monitor Timer 1 operate keys:-



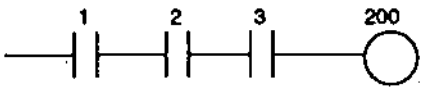
Display will show



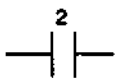
To monitor next timer (No. 2) operate 

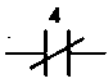
To change an input/output/internal relay timer/counter/special relay in the program.

Example



Program
LOD 1
AND 2
AND 3
OUT 200

To change 

(AND 2) to 

(AND NOT 4)

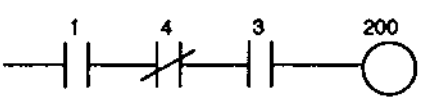
1. Set programmer to start of program (address 0)

2. Search for "AND 2" by first searching for "OUT 200"

To step program back until "AND 2" is displayed.

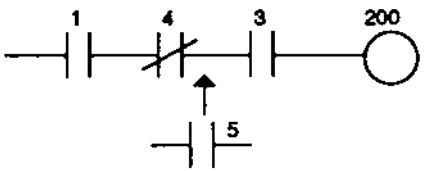
3. Operate following keys:-

Program
LOD 1
AND NOT 4
AND 3
OUT 200



To insert an input/output/timer/counter/special relay in the program.

Example

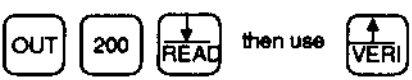


'AND 5' in rung

Program
LOD 1
AND NOT 4
AND 3
OUT 200

1. Set programmer to start of program (address 0)

2. Search for "AND 3" by first searching for "OUT 200"

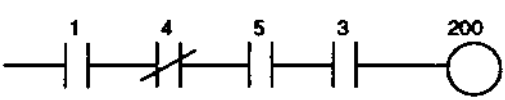


To step program back until "AND 3" is displayed.

3. Operate following keys:-

This will insert "AND 5" between "AND NOT 4" and "AND 3".

4. Final result.

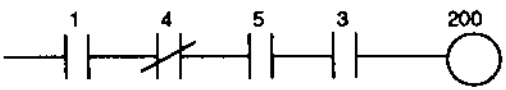


Program
LOD 1
AND NOT 4
AND 5
AND 3
OUT 200

PROGRAM EDITING - DELETE ROUTINE

To delete an Input/output/internal relay timer/counter/special relay in the program.

Example



Program
LOD 1
AND NOT 4
AND 5
AND 3
OUT 200

Delete "AND NOT 4"
and "AND 5"

1. Set programmer to start of program (address 0).

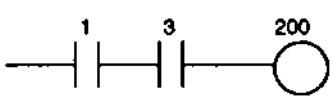
2. Search for "AND NOT 4" by first searching for "OUT 200"

to step back until "AND NOT 4" is displayed.

3. Operate following keys:-

This will delete two instructions in the program. The one displayed and the instruction following it. When deleting two address instructions (TIM, CNT, SFR etc) the two addresses will be deleted at one time as one instruction.

4. Final result

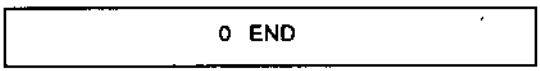


Program
LOD 1
AND 3
OUT 200

1. This routine clears the memory in the programmer.
   This routine must be carried out before entering a new program.
2. When the memory is cleared "END" instructions are automatically written at every step in the memory.

Key sequence to clear memory:-

Display will show:-

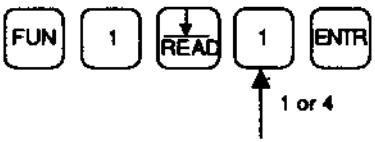


Notes

1. The clear memory routine can only be performed when the programmer mode switch is in "Program" position.
2. All function (FUN) settings are cleared by a clear memory routine.
3. The memory in the memory pack is **not** cleared by this routine.
4. After a memory clear, the program capacity of the programmer is set to 1K steps.

PROGRAM TRANSFER - TO/FROM CMOS - RAM MEMORY PACK

Set the programmer step capacity to same value as CMOS-RAM pack (1K or 4K) using following key sequence:-



Display will show:-

Transfer of program from programmer to CMOS-RAM pack

Check that RAM pack is installed in processor and processor is not in 'RUN'.
Key sequence is as follows:-

When writing is complete, the following message is displayed:-

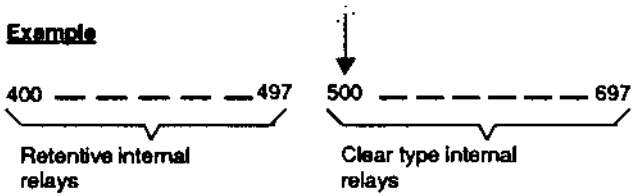
Transfer of program from CMOS-RAM pack to programmer

Key sequence is as follows:-

When writing is complete, the following message is displayed:-

It is possible to optionally assign the numbers of internal relays whose conditions are cleared (clear type relays) or maintained (retentive relays) when the processor goes to halt or the A.C. power is interrupted.
The internal relays are numbered from 400 to 697 (240 total).
Any number in this range can be specified as the number at which the clear type relays start.

Example



To set number 500, operate the following keys:-

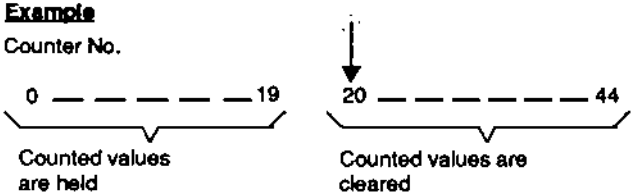
Notes

1. When all memories are cleared (programmer and processor), number "400" is set automatically; therefore, all internal relays are assigned as clear type relays.
2. The assignment of the number must be completed before the program is transferred to a memory pack.

ASSIGNMENT OF UP COUNTERS AS RETENTIVE OR CLEAR COUNTERS USING FUNCTION 7

It is possible to optionally assign the numbers of up counters (counters 0 to 44) whose counts are cleared (clear type counters) or maintained (maintain type counters) when the processor halts or the A.C. power is interrupted.
Any number in the range 0 to 44 can be specified as the number at which the clear type counters start.

Example



To set number 20, operate following keys:-

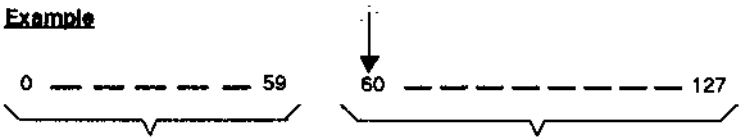
To assign all counters as maintain type, set number to '45'.

Notes

1. When all memories are cleared (programmer and processor), number "0" is set automatically; therefore, all counters are assigned as clear type counters.
2. Reversible counters 45 and 46 are both maintain type: however, the counted value can be cleared automatically on power up by programming a 704 contact in the preset line.
3. The assignment of the number must be completed before the program is transferred to a memory pack.

It is possible to optionally assign the number of shift register bits whose status is cleared, (clear type bits) or retained (retentive type bits) when the processor halts or the A.C. power is interrupted.
Any number in the range 0 to 127 can be specified as the number at which the clear type bits start.

Example



To set bit number 60, operate keys:-

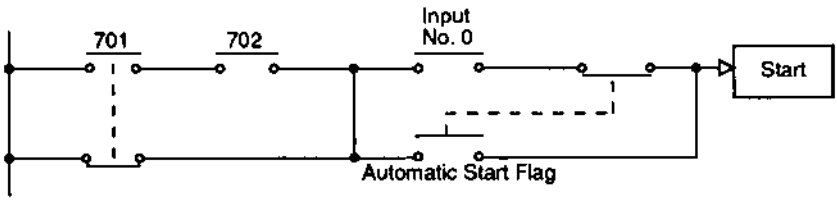
To assign all shift register bits as retentive, set number 127.

Notes

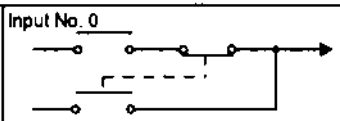
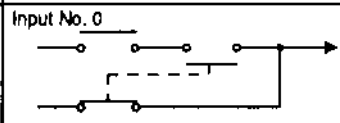
1. When all memories are cleared (programmer and processor), number "0" is set automatically; therefore, all shift register bits are assigned as clear type.
2. The assignment of the number must be completed before the program is transferred to a memory pack.

PROCESSOR START/STOP CIRCUIT EXPLANATION

The Start/Stop circuit of the processor is as shown below:-



The automatic start flag is controlled by function 61 as follows:-

Function 61 set to 0	Start flag OFF	Input No. 0 
Function 61 set to 500	Start flag ON	Input No. 0 

Special internal relays 701 and 702 can be turned ON and OFF directly from the program loader by using the 'SET' and 'RST' functions.

For example, internal relay 701 can be set by operating

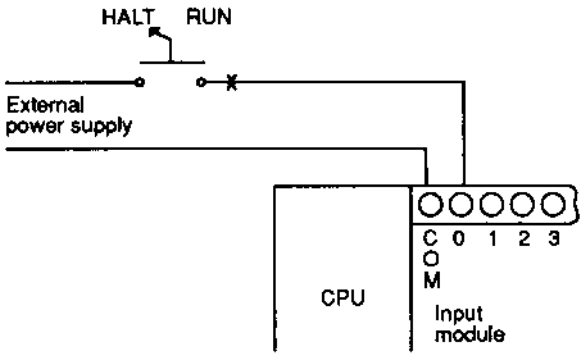
TRS SET 701 ENTR

Internal relay 701 can be reset by operating

TRS RST 7Ø1 ENTR

The two main methods of processor start/stop using the above functions are as follows:
1. Start/stop using input 0 with HALT/RUN switch (see page 39).
2. Start/stop using relays 701 and 702 (see page 40).

Input 0 is wired to a switch as shown below.



When the HALT/RUN switch is put to RUN, input 0 is turned ON, and the processor switches to run (RUN LED is ON).
When the HALT/RUN switch is put to HALT, input 0 is turned OFF, and the processor switches to HALT.
When the switch is in RUN, the processor will automatically go into RUN if the A.C. power is switched off and then restored.

Note
The above circuit will only operate as described, if Function 61 (FUN 61) is set to 0.
Function 61 will automatically set to 0, if the programmer memory and the processor memory are cleared (this operation is normally performed before commencing programming).
If it is necessary to set FUN 61 to 0, the key sequence is as follows:-

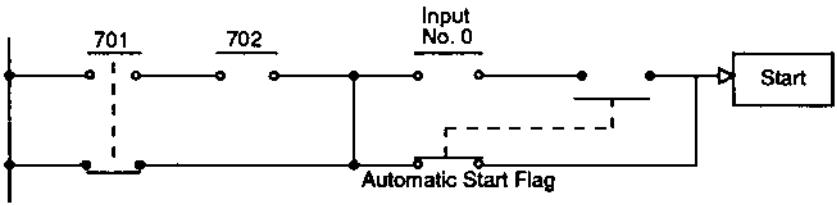
When the HALT/RUN switch is put to RUN, there will be a delay before the processor goes into run. The delay is: 1K steps 1sec; 4K steps 4 sec.

PROCESSOR START/STOP OPERATION USING SPECIAL INTERNAL RELAYS 701 AND 702

This method of Start/Stop operation can only be used after FUN 61 is set to 500:-

FUN 61 READ 500 ENTR

This operation sets the automatic start flag ON. The Start/Stop circuit is now as shown.



The above circuit as shown, with special internal relays 701 and 702 switched off, will automatically start the processor (RUN LED ON) as soon as power is switched on to the processor.
To stop the processor (RUN LED OFF), operate following keys:-

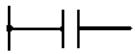
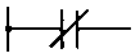
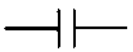
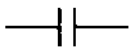
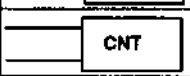
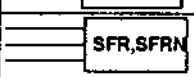
TRS SET 701 ENTR

TRS RST 702 ENTR

To restart the processor operate the following keys:-

TRS RST 701 ENTR

If the processor is in "RUN", at the time the A.C. power to the processor is interrupted, it will switch back to "RUN" when A.C. power is restored to the processor.

INSTRUCTION	LADDER SYMBOL	PROGRAM ENTRY
LOD		LOD, (DATA NUMBER), ENTR
LOD NOT		LOD, NOT, (DATA NUMBER), ENTR
AND		AND, (DATA NUMBER), ENTR
AND NOT		AND, NOT, (DATA NUMBER), ENTR
OR		OR, NOT, (DATA NUMBER), ENTR
OR NOT		OR, NOT, (DATA NUMBER), ENTR
OUT		OUT, (DATA NUMBER), ENTR
TIM		TIM, (TIMER NUMBER), ENTR
CNT		COUNTER, (COUNTER NUMBER), ENTR
SFR, SFR NOT		SFR, (INITIAL BIT NUMBER), ENTR, (# OF BITS)
MCS		MCS, ENTR
MCR		MCS, ENTR
SOT		SOT, (SINGLE OUTPUT NUMBER), ENTR
SET		SET, (DATA NUMBER), ENTR
RST		RST, (DATA NUMBER), ENTR
JMP		END, ENTR
JEND		JEND, ENTR
END		END, ENTR