PLCs AND DCS IN PROCESS CONTROL AUTOMATION

Course Code: **20EI701**

Sharath H K, M.Tech.

Assistant Professor

Dept. of Mechanical Engineering,

MCE, Hassan.

PLC Instructions

Module – 3

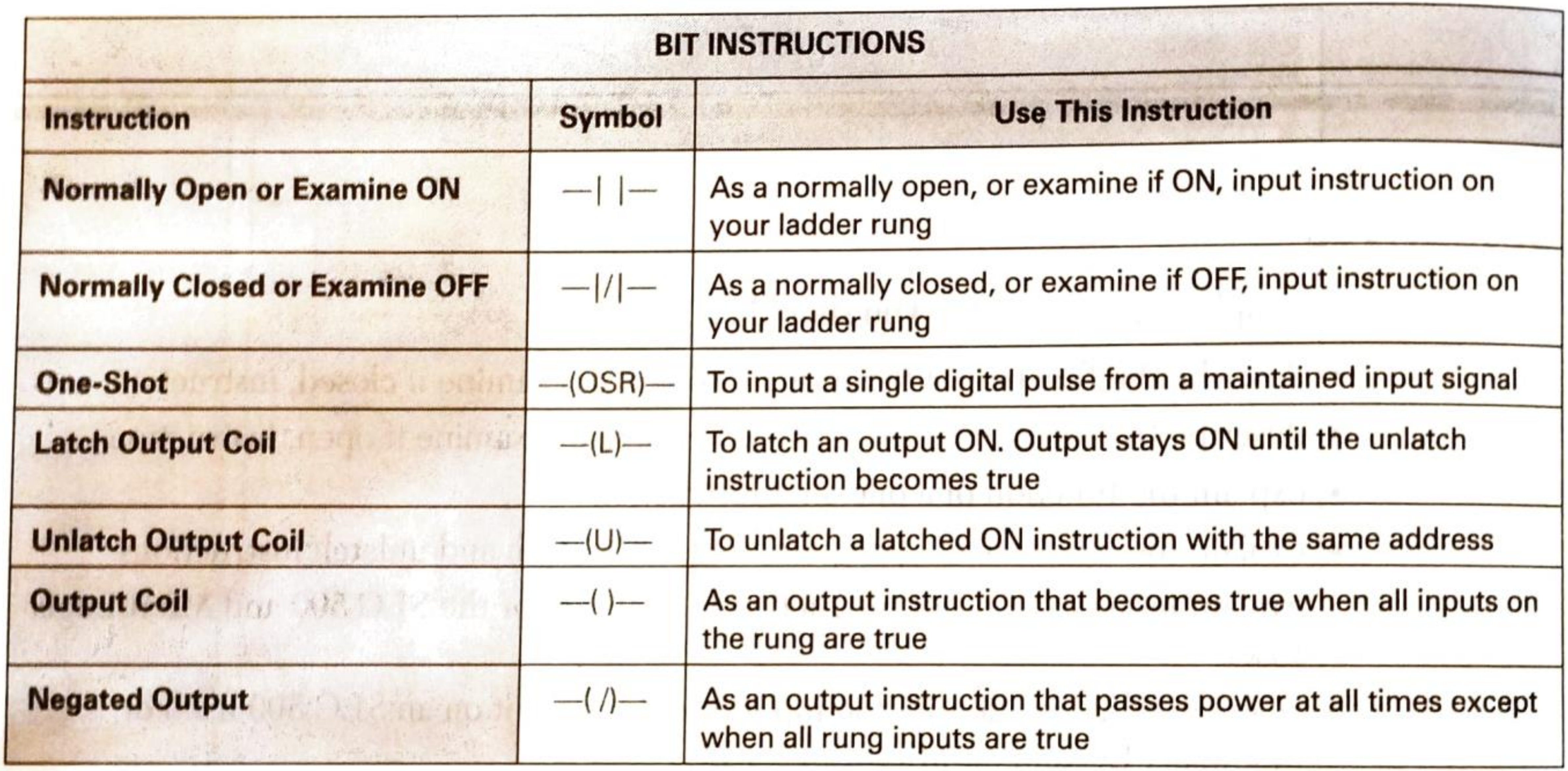
Contents

Module – 3

PLC Instructions: The Basic Relay Instructions, The Normally OPEN and normally CLOSED Instructions, the one shot Instruction, The output latching Instruction, the negated output Instruction. Understanding Relay instructions and the programmable controller Input Modules: Interfacing a Start-Stop Push Button station and Motor to a PLC. Construction of Ladder diagram for analytical problems. Timer instructions: Timer instructions, the ON – delay Timer and OFF delay Timer instructions, the retentive timer Instruction, The RESET Instruction. Counter Instructions: the Count Up and Countdown Instruction, The counter Reset and Clear Instruction, Combining Counters and Timers. Construction of Ladder diagram for analytical problems.

BIT or RELAY INSTRUCTIONS

- Contacts and coils are the basic symbols found on a ladder diagram.
- Normally open or normally closed contact symbols are programmed on a given rung to represent input conditions that are to be evaluated by the processor.
- Each output is represented by a coil symbol.
- Contacts and coils are also referred to as bit or relay instructions.
- Each input or output is represented by a separate bit in the input or output status file.
- Bits in PLC memory are typically input status file bits representing ON or OFF signals input into the status file from an input module.

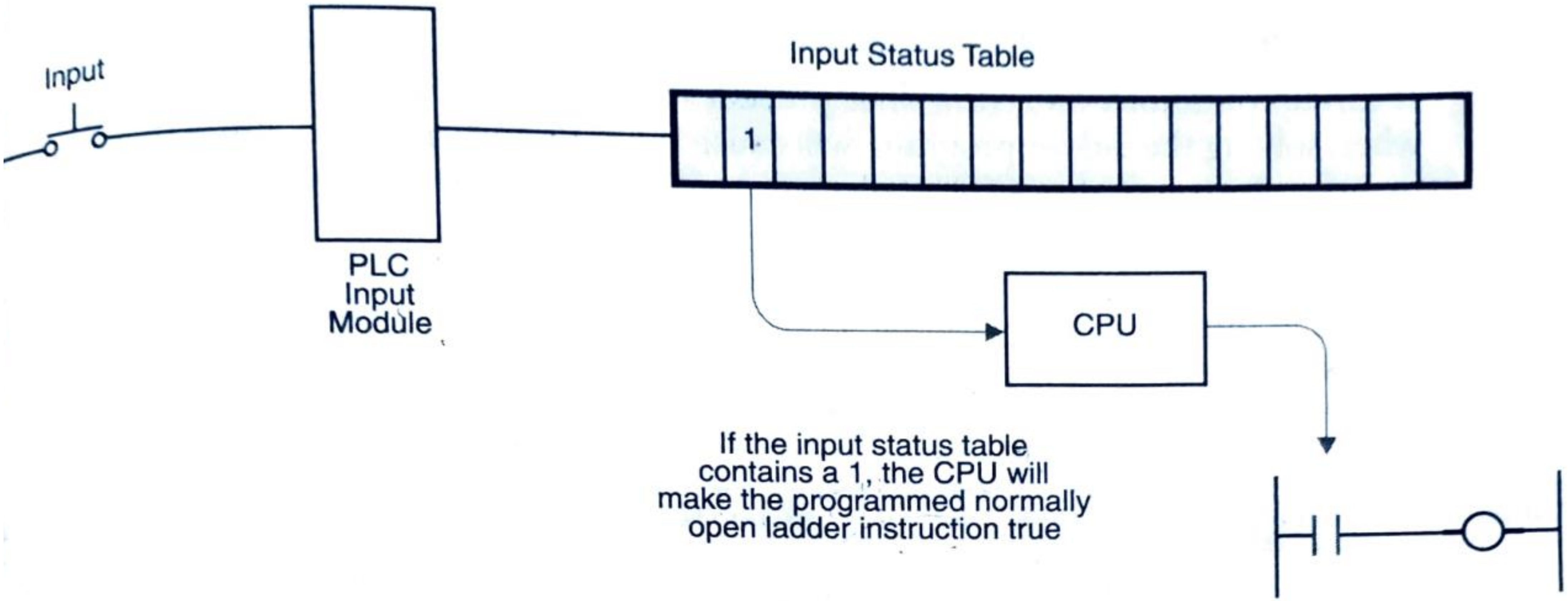


BIT INSTRUCTIONS		
Instruction	Symbol	Use This Instruction
Normally Open or Examine ON	—| |—	As a normally open, or examine if ON, input instruction on your ladder rung
Normally Closed or Examine OFF	—|/|—	As a normally closed, or examine if OFF, input instruction on your ladder rung
One-Shot	—(OSR)—	To input a single digital pulse from a maintained input signal
Latch Output Coil	—(L)—	To latch an output ON. Output stays ON until the unlatch instruction becomes true
Unlatch Output Coil	—(U)—	To unlatch a latched ON instruction with the same address
Output Coil	—( )—	As an output instruction that becomes true when all inputs on the rung are true
Negated Output	—(/)—	As an output instruction that passes power at all times except when all rung inputs are true

THE NORMALLY OPEN INSTRUCTION

- The SLC 500 and MicroLogix use the term "examine if closed" (XIC) to represent the normally open instruction.
- Other PLC manufacturers might use “examine if ON” to identify the normally open instruction.
- The examine if closed instruction tells the processor to test for an ON condition from the reference address bit.

THE NORMALLY OPEN INSTRUCTION (Continued..)



If the input status table contains a 1, the CPU will make the programmed normally open ladder instruction true

Figure 13-2 Normally open instruction interaction between field device, input module, input status table, and the CPU.

THE OUTPUT INSTRUCTION

- The output instruction is typically represented as an output coil.
- The SLC 500 family and MicroLogix refer to the output coil as an output energize instruction (OTE).
- Every rung must have a minimum of one output instruction, and it may have more than one output.
- Multiple outputs are programmed in parallel.
- The output instruction is always the last instruction before the right power rail.
- An output instruction represents the action that is to be taken when the solved input logic results in a logically true rung.

THE OUTPUT INSTRUCTION (Continued..)

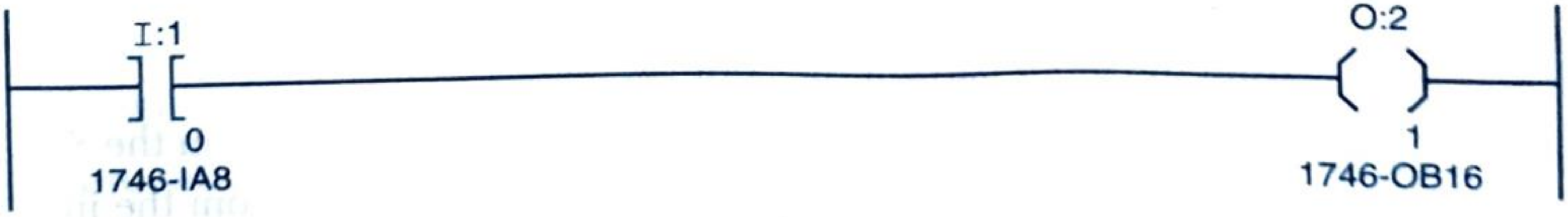


Figure 13-3 Ladder rung containing an examine if closed input instruction and an output enable instruction.

THE NORMALLY CLOSED INSTRUCTION

- The normally closed instruction is also called the examine off, or examine if open, depending on the particular PLC manufacturer.

- The SLC 500 and MicroLogix use the term "examine if open" (XIO) to identify this instruction.

- If the status table bit is found to be a 0, the device controlling this bit is in its ON state.

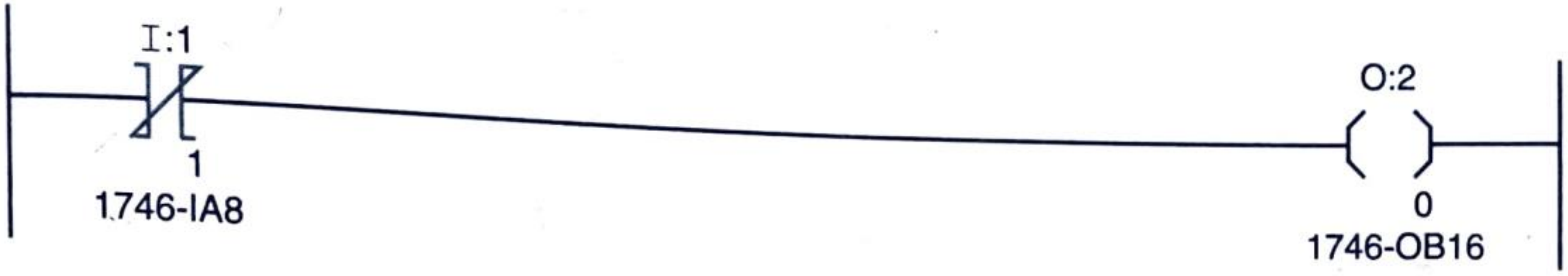
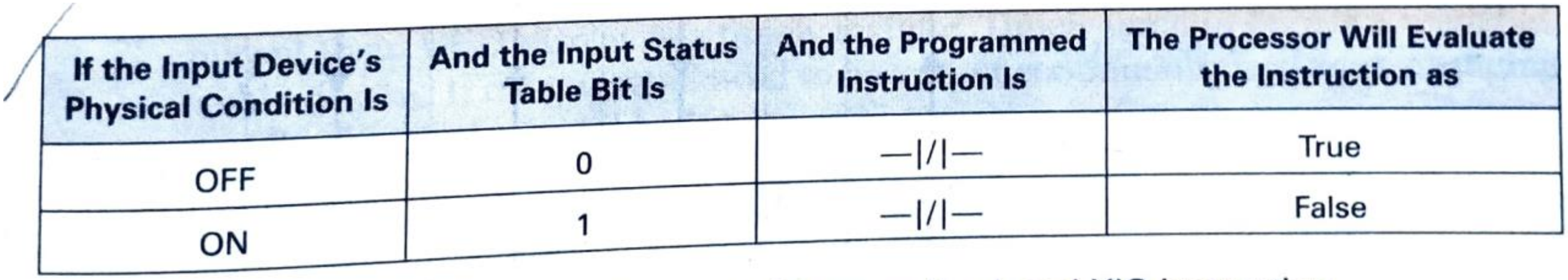


Figure 13-8 Normally closed, or examine if open, instruction controlling an OTE instruction.

THE NORMALLY CLOSED INSTRUCTION (Continued..)



If the Input Device's Physical Condition Is	And the Input Status Table Bit Is	And the Programmed Instruction Is	The Processor Will Evaluate the Instruction as
OFF	0	—|/|—	True
ON	1	—|/|—	False

Figure 13-9 Physical input conditions and the normally closed XIO instruction.

THE ONE-SHOT INSTRUCTION

- The one-shot rising instruction, OSR, is an input instruction that allows an event to occur only once.

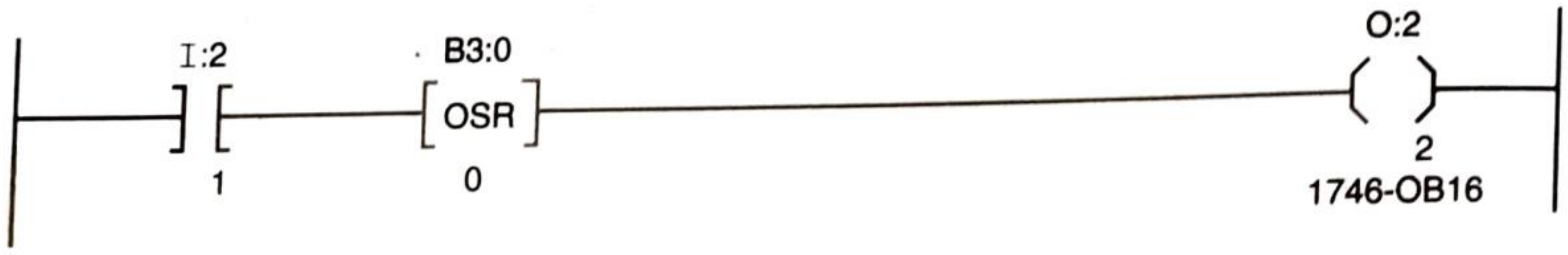


Figure 13-10 SLC 500 family one-shot ladder rung.

- The OSR instruction controls the one-shot output for output address 0:2/2
- The "rising" portion of the one-shot rising instruction means that the instruction is looking for an OFF-to-ON, or false-to-true, transition in the leading-edge pulse input.

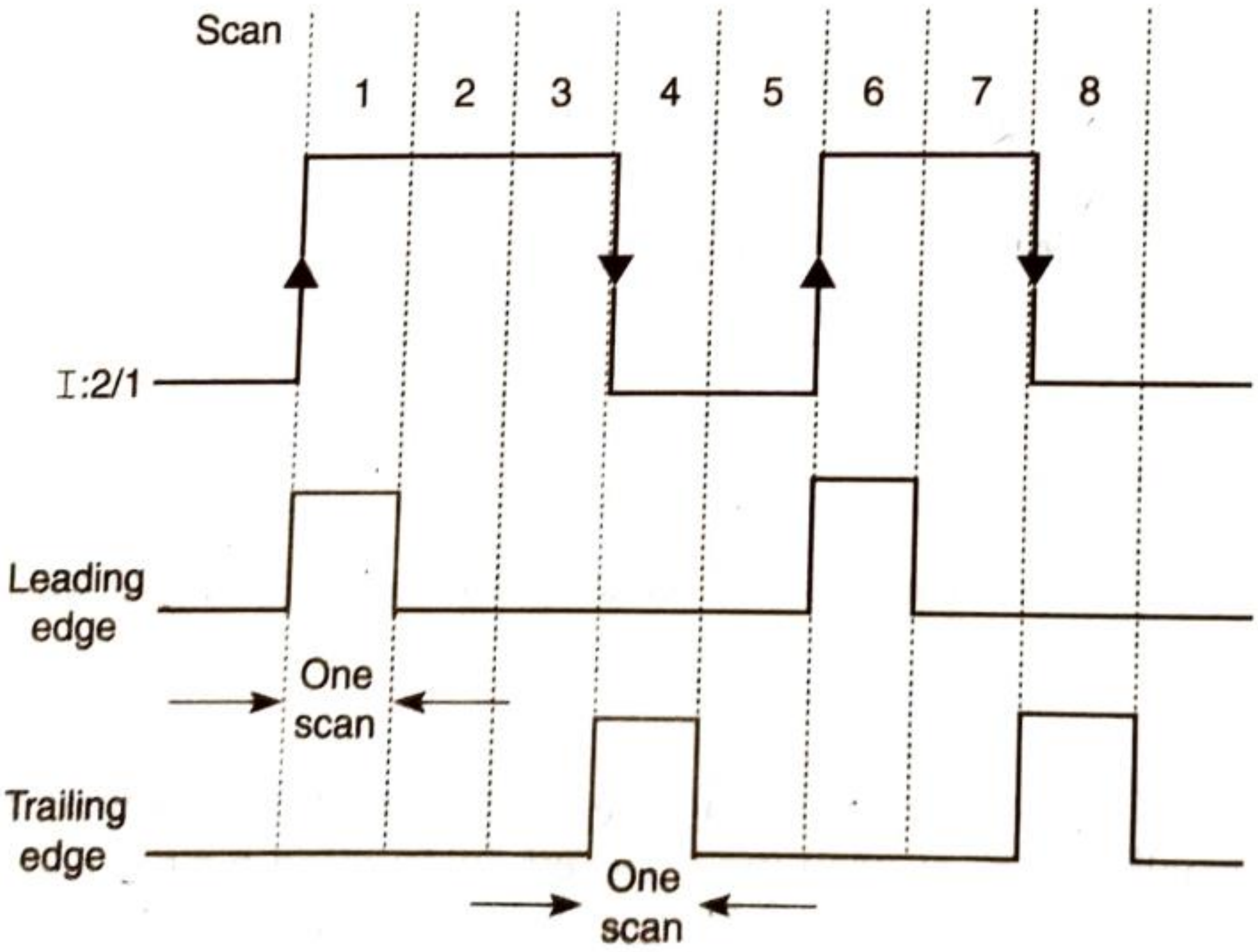


Figure 13-11 Leading-edge versus trailing-edge one-shot timing diagram.

THE ONE-SHOT INSTRUCTION (Continued..)

- A one-shot instruction operates in a manner similar to an output instruction.
- If a continuous path of input instructions on a ladder rung preceding the one-shot instruction is true, the one-shot instruction will be energized.
- When input instruction 1:2/1 goes from false to true, the one-shot (OSR) referenced output instruction goes true for one scan.
- Following the scan in which the OSR instruction is true, the referenced output instruction goes false.
- The OSR instruction holds the referenced output false even though the input logic to the OSR instruction may remain true.

THE ONE-SHOT INSTRUCTION (Continued..)

- The OSR instruction will only allow the referenced output to become true again after input logic goes false and then transitions from false to true again.

Application of One-Shot Instruction

- One-Shot Instruction is used to start an event triggered by a push-button input into PLC where the event has to happen only once per actuation of the push button, no matter how long it is held in.

THE ONE-SHOT INSTRUCTION (Continued..)

Examples:

- A one-shot can be used to reset desired conditions in a single scan.
- A momentary push-button actuation could be used to increment speed on a motor. Speed would increment one step for each push of the button.
- One-shot instruction can be used with a math instruction to perform a calculation once per scan.
- A one-shot instruction can be used to bring in changing analog input data, which can be sampled at a predetermined rate.
- A push button or an internal bit and the one-shot instruction are used to send data to output display devices.

THE ONE-SHOT INSTRUCTION (Continued..)

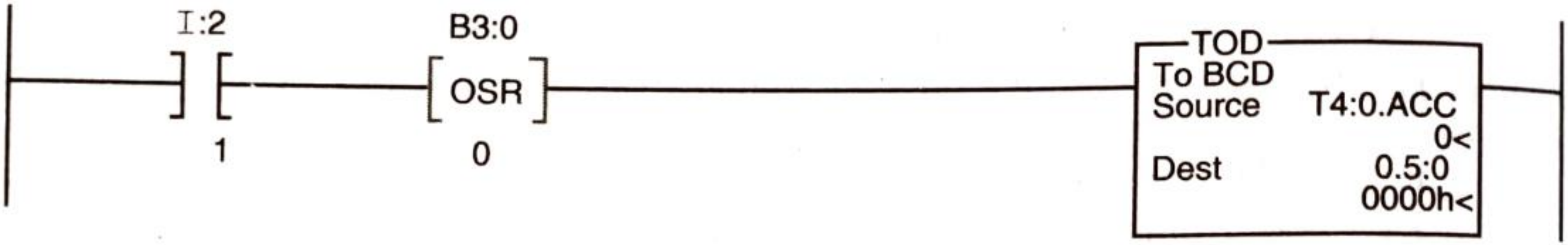


Figure 13-12 A rung of logic in which a one-shot rising instruction controls and converts an integer value to BCD instruction (TOD instruction).

- Figure 13-12 illustrates a rung where the OSR instruction will allow the conversion of integer data to BCD data.
- The PLC conversion instruction, TOD (to BCD), will be true for one scan.

THE ONE-SHOT INSTRUCTION (Continued..)

- The TOD instruction will convert decimal integer data from the accumulated value of the timer file 4, timer zero (T4:0.ACC) and send it as an output only once to the destination O:5.0.
- If the timer was running and displayed continuously, the display might change rapidly and be unreadable.
- The one-shot instruction allows the TOD instruction to send converted data out only when I:2/1 triggers the OSR instruction.
- A timer used in conjunction with a one-shot instruction could provide a stable display that could be updated every few seconds rather than appear as a blur of numbers.

THE ONE-SHOT INSTRUCTION (Continued..)

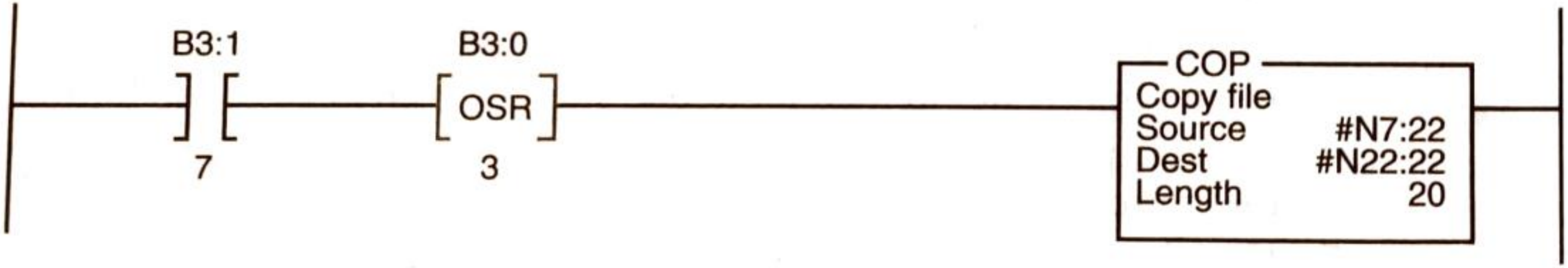


Figure 13-13 Internal bit B3:0/3 controlling a one-shot instruction to enable data to be copied only once per trigger of bit B3:0/3.

- A one-shot instruction could be used to trigger a copy instruction, as illustrated in Figure 13-13.
- The one-shot instruction will allow the copy instruction to copy data from the source to the destination only once per trigger.

THE OUTPUT-LATCHING INSTRUCTION

- An output-latching instruction is an output instruction used to maintain, or latch, an output ON even if the status of the input logic that caused the output to energize changes.
- The latched instruction will remain in a latched ON condition until an unlatch instruction with the same reference address is energized.
- Latch and unlatch instructions are always used in pairs. Each instruction is typically located on a separate rung.

THE OUTPUT-LATCHING INSTRUCTION (Continued..)

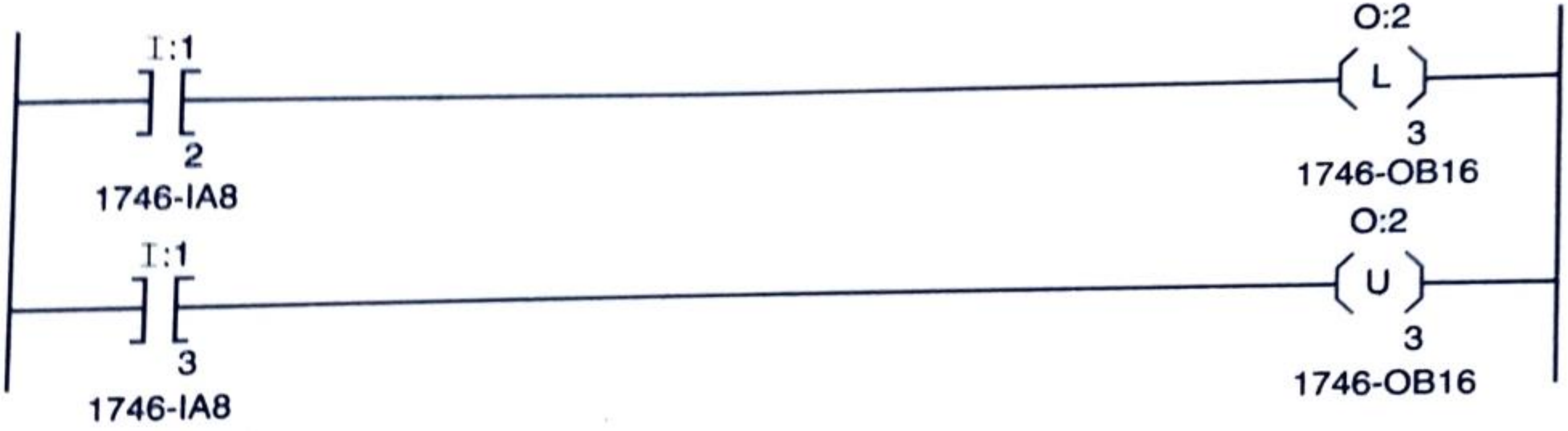


Figure 13-14 Latching and unlatching ladder logic.

- Figure illustrates two ladder rungs.
- The first rung contains the latching instruction, while the second rung contains the unlatching instruction.

THE OUTPUT-LATCHING INSTRUCTION (Continued..)

- When input 1:1/2 on the first rung is energized, the output-latch instruction, address O:2/3, is energized.
- The output-latch instruction will remain latched ON and will be unaffected, no matter how input 1:1/2 changes.
- When input I:1/3 is energized, the unlatch instruction will turn off output O:2/3.

THE OUTPUT-LATCHING INSTRUCTION (Continued..)

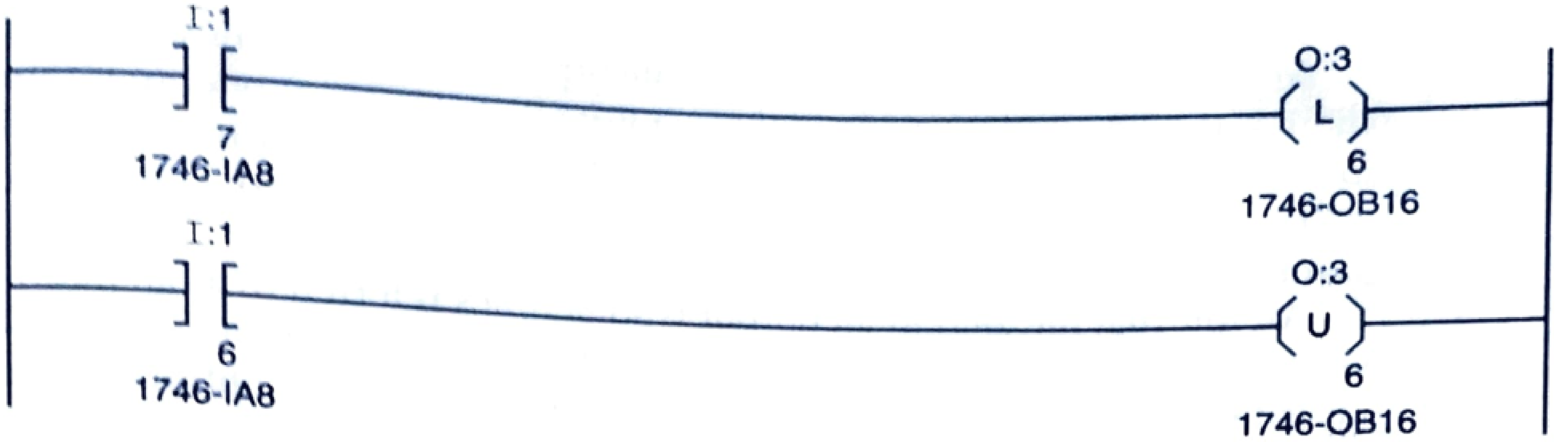


Figure 13-16 Latching instruction programmed before the unlatch instruction.

- Figure 13-16 illustrates a latching instruction programmed before the unlatch instruction.
- If both instructions are true, the last instruction programmed on the rung will take precedence over the other instruction.
- In this example, the output instruction will always be unlatched.

THE OUTPUT-LATCHING INSTRUCTION (Continued..)

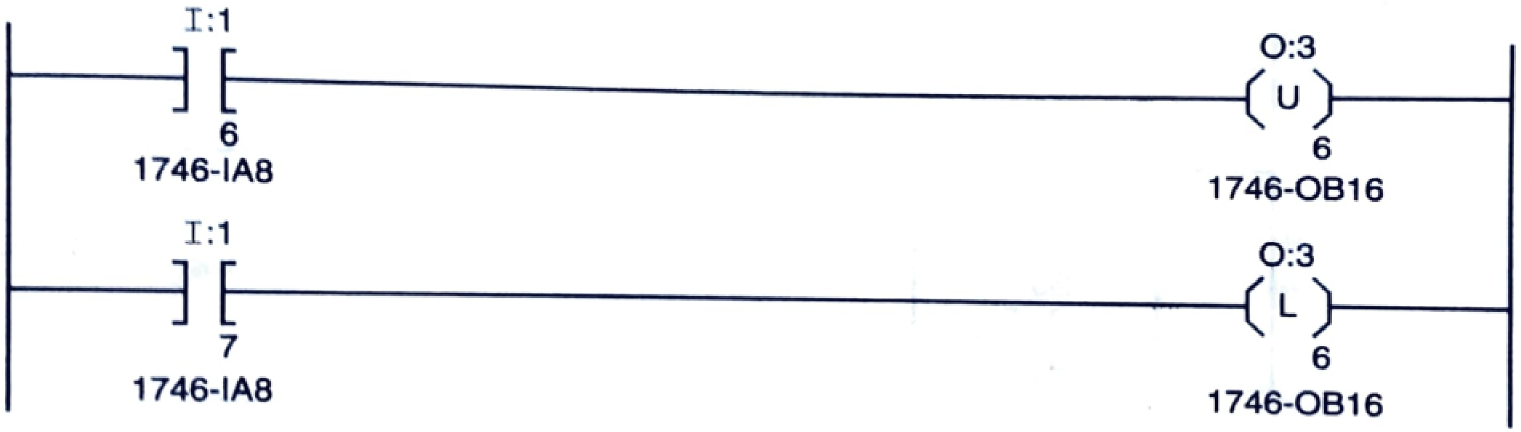
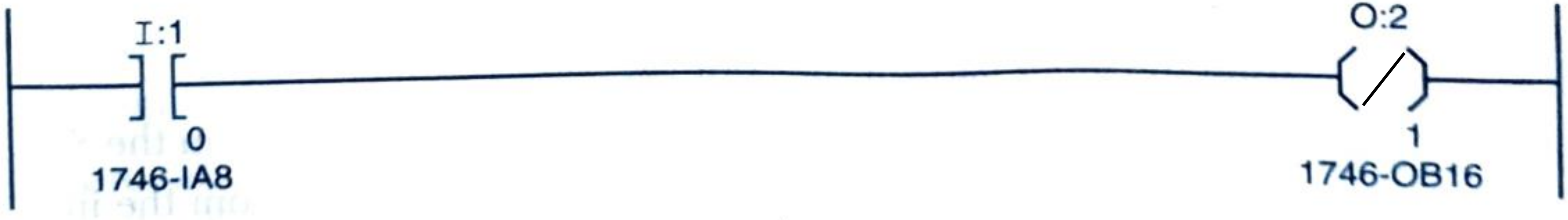


Figure 13-17 Latching instruction programmed after the unlatch instruction.

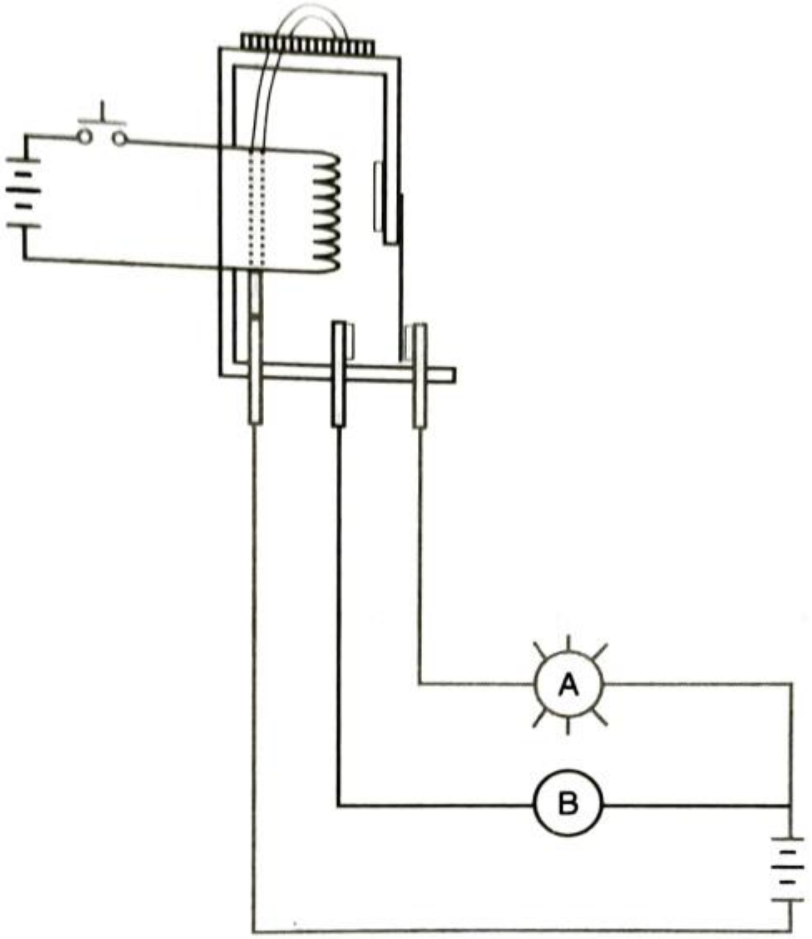
- In Figure 13-17, the unlatch instruction is programmed after the latch instruction.
- In this case, the last instruction-the latch instruction-will take precedence and keep the output latched, provided both the latch and unlatch rungs are true.

THE NEGATED OUTPUT INSTRUCTION

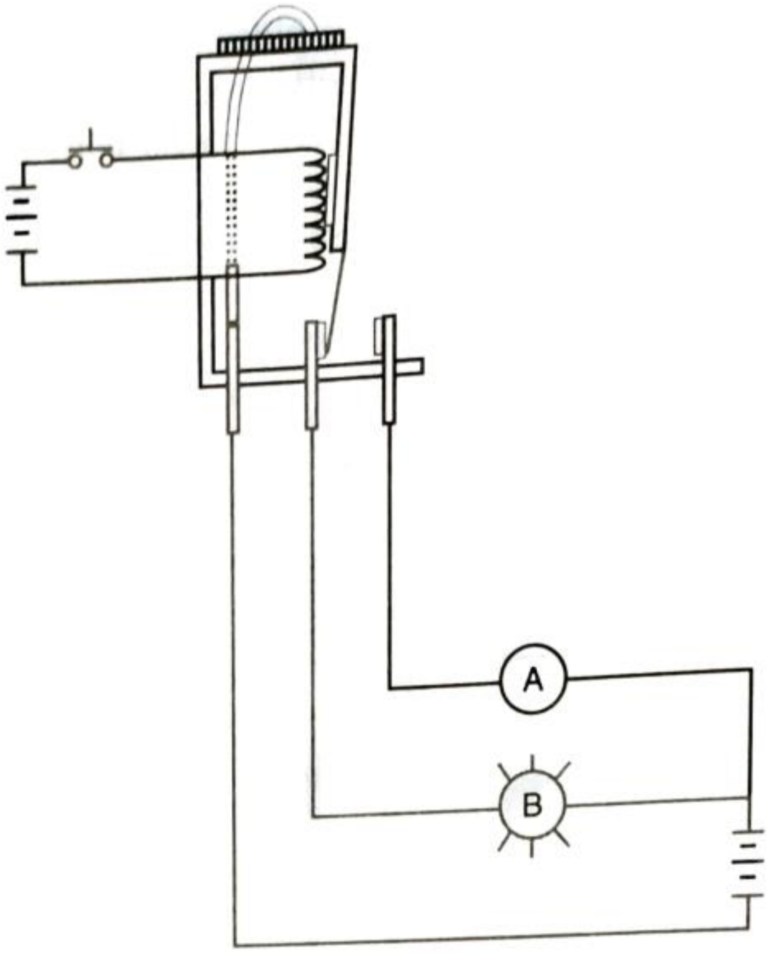
- The negated output instruction is the opposite of a normal output instruction.
- The negated output instruction is true when the rung's input logic is false.
- The negated output instruction is also referred to as a not output coil or instruction.



UNDERSTANDING RELAY INSTRUCTIONS

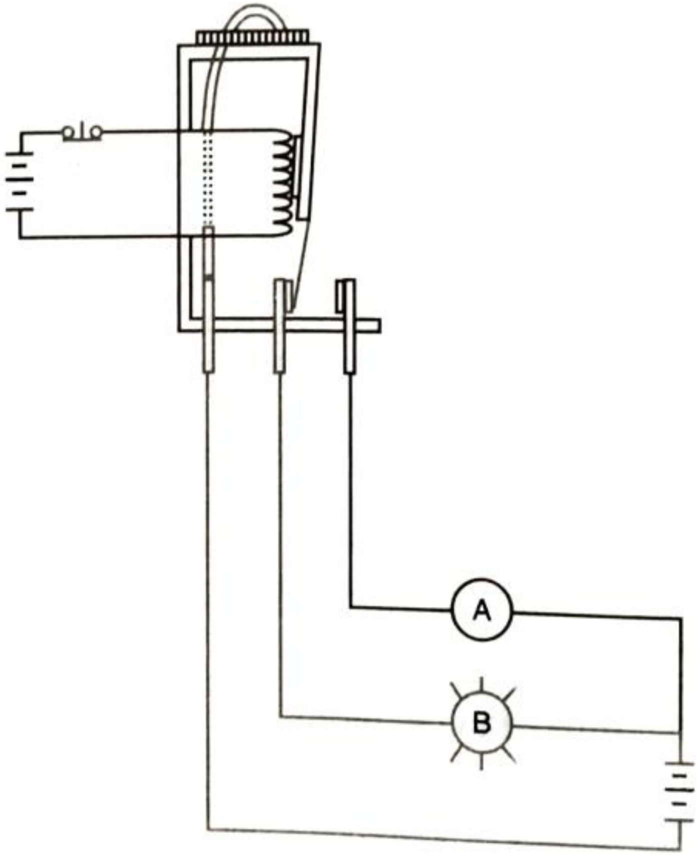


Normally open push button and nonenergized relay.

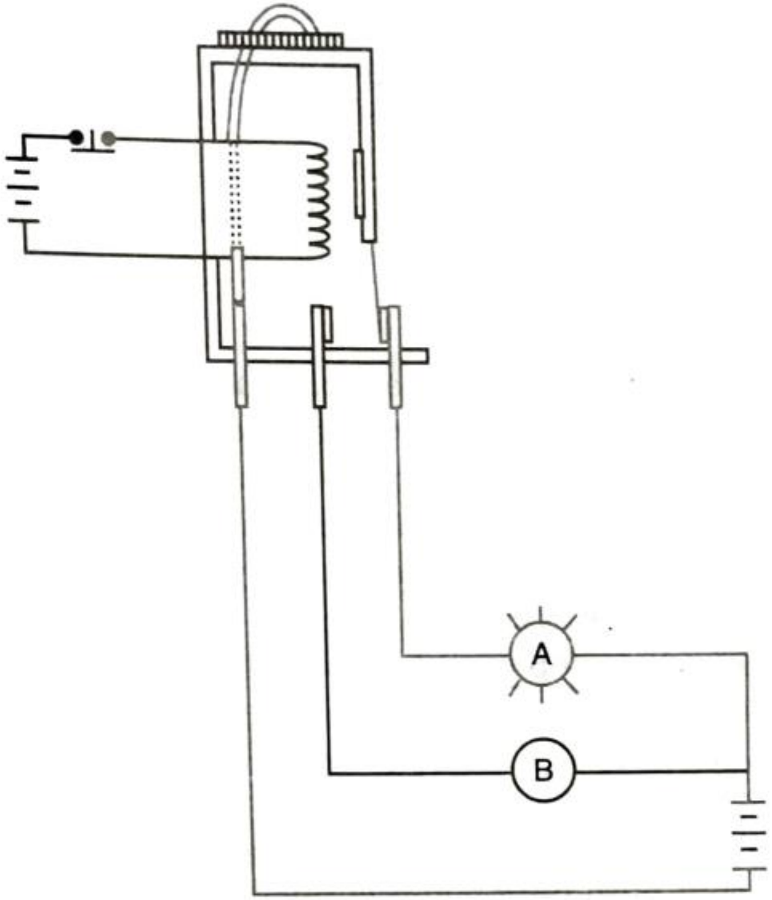


Normally open push button energizing relay coil and Pilot Light B.

UNDERSTANDING RELAY INSTRUCTIONS (Continued..)

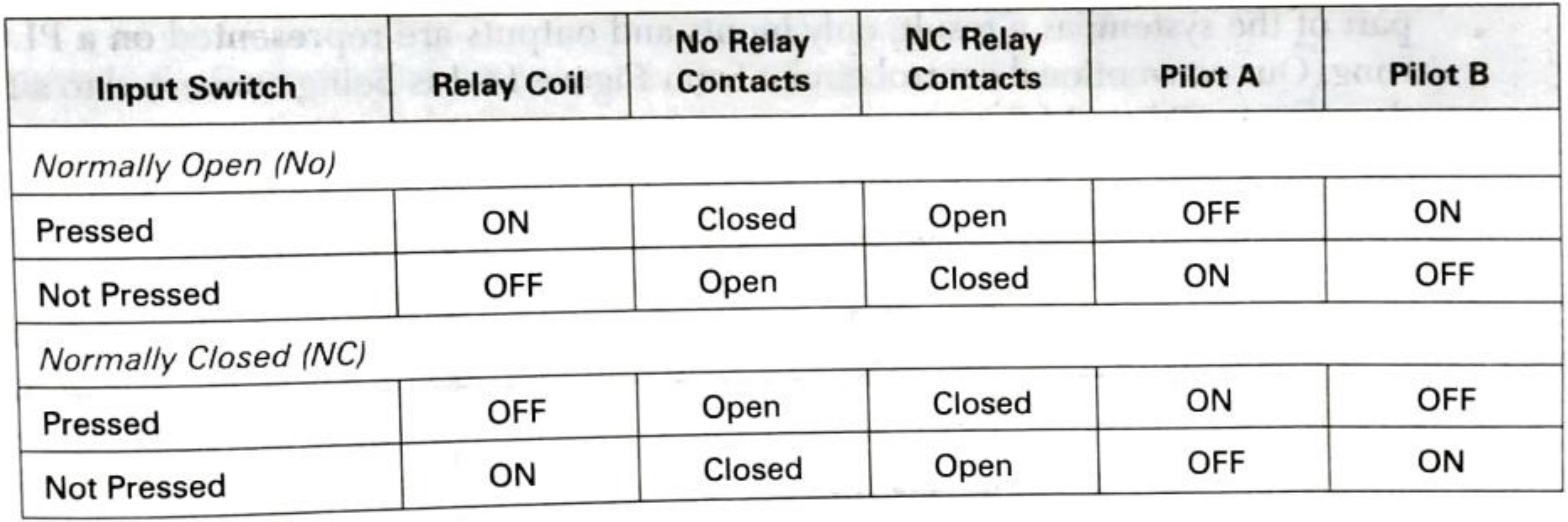


Normally closed push button energizing relay coil and Pilot Light B.



Normally closed stop push button depressed, causing an open input circuit.

UNDERSTANDING RELAY INSTRUCTIONS (Continued..)



Input Switch	Relay Coil	No Relay Contacts	NC Relay Contacts	Pilot A	Pilot B
Normally Open (No)					
Pressed	ON	Closed	Open	OFF	ON
Not Pressed	OFF	Open	Closed	ON	OFF
Normally Closed (NC)					
Pressed	OFF	Open	Closed	ON	OFF
Not Pressed	ON	Closed	Open	OFF	ON

Figure 14-7 Relay status in conjunction with input push buttons.

Interfacing a Start-Stop Push-button Station to a Programmable Controller

- The typical start-stop push button has one momentary, normally open, start push button and one momentary, normally closed, stop push button.
- Stop push buttons must fail safely. To fail safely, the stop push button must be normally closed.
- As a normally closed device, the stop push button will continually pass power until depressed.

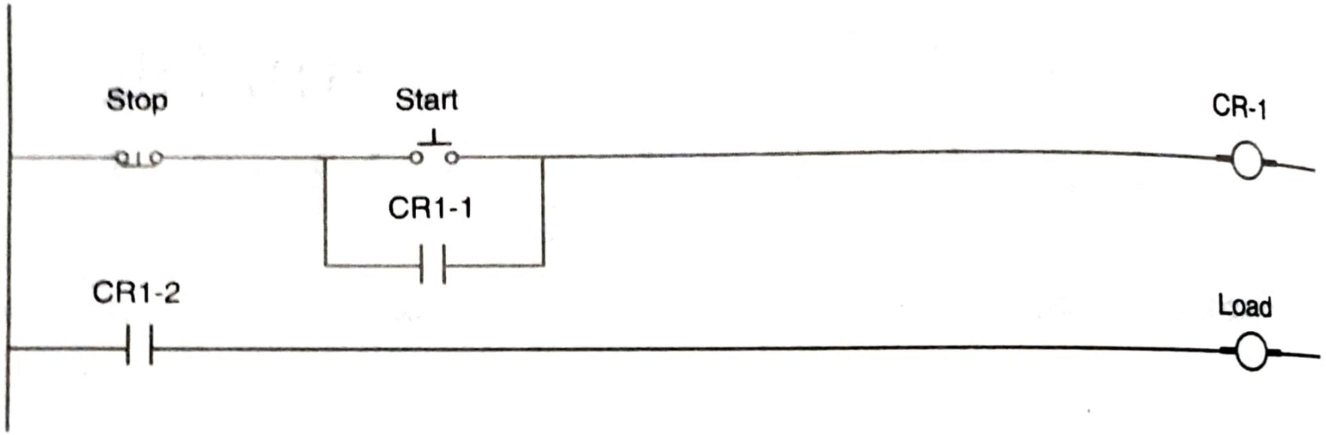


Figure 14-1 Typical hard-wired start-stop latching circuit.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

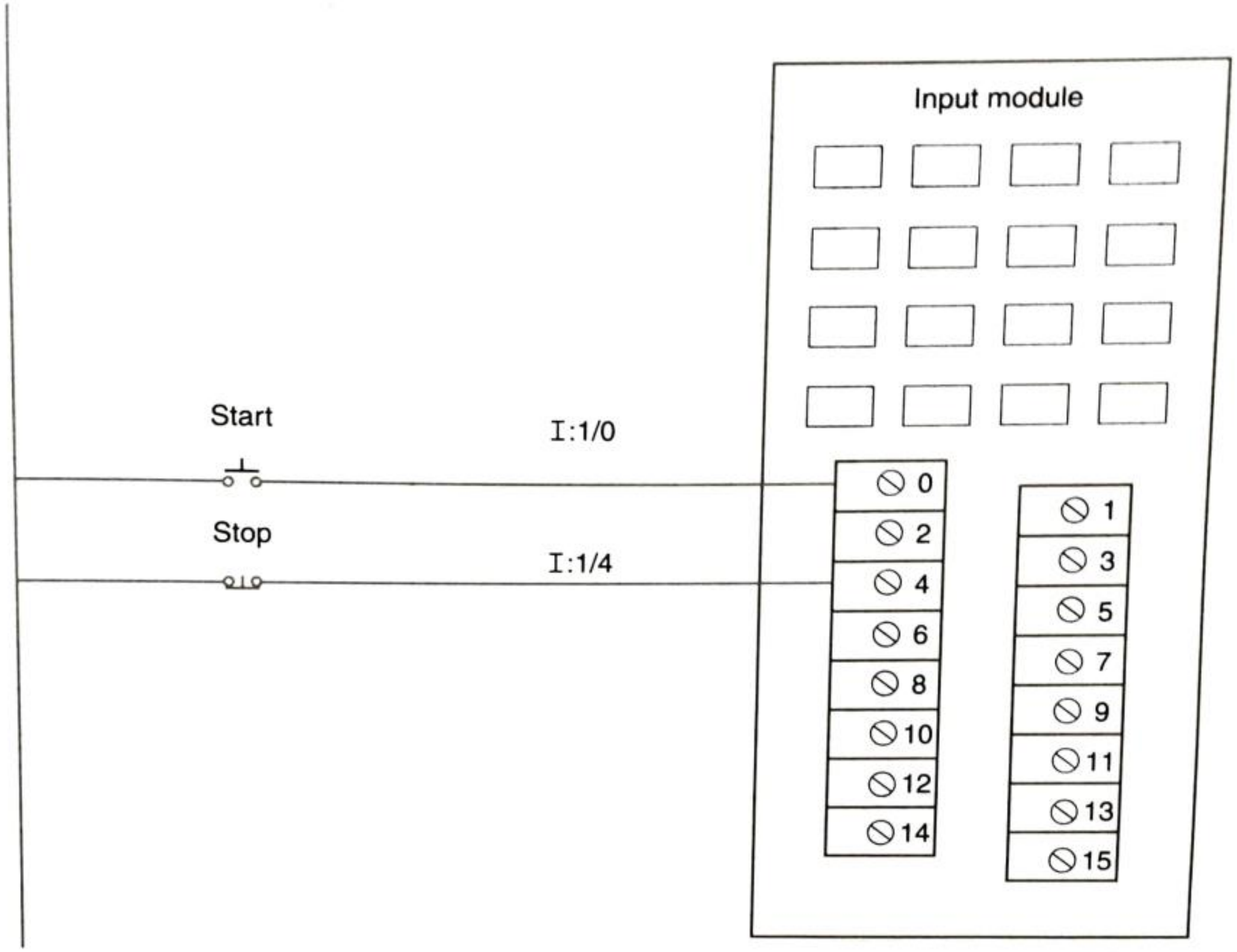
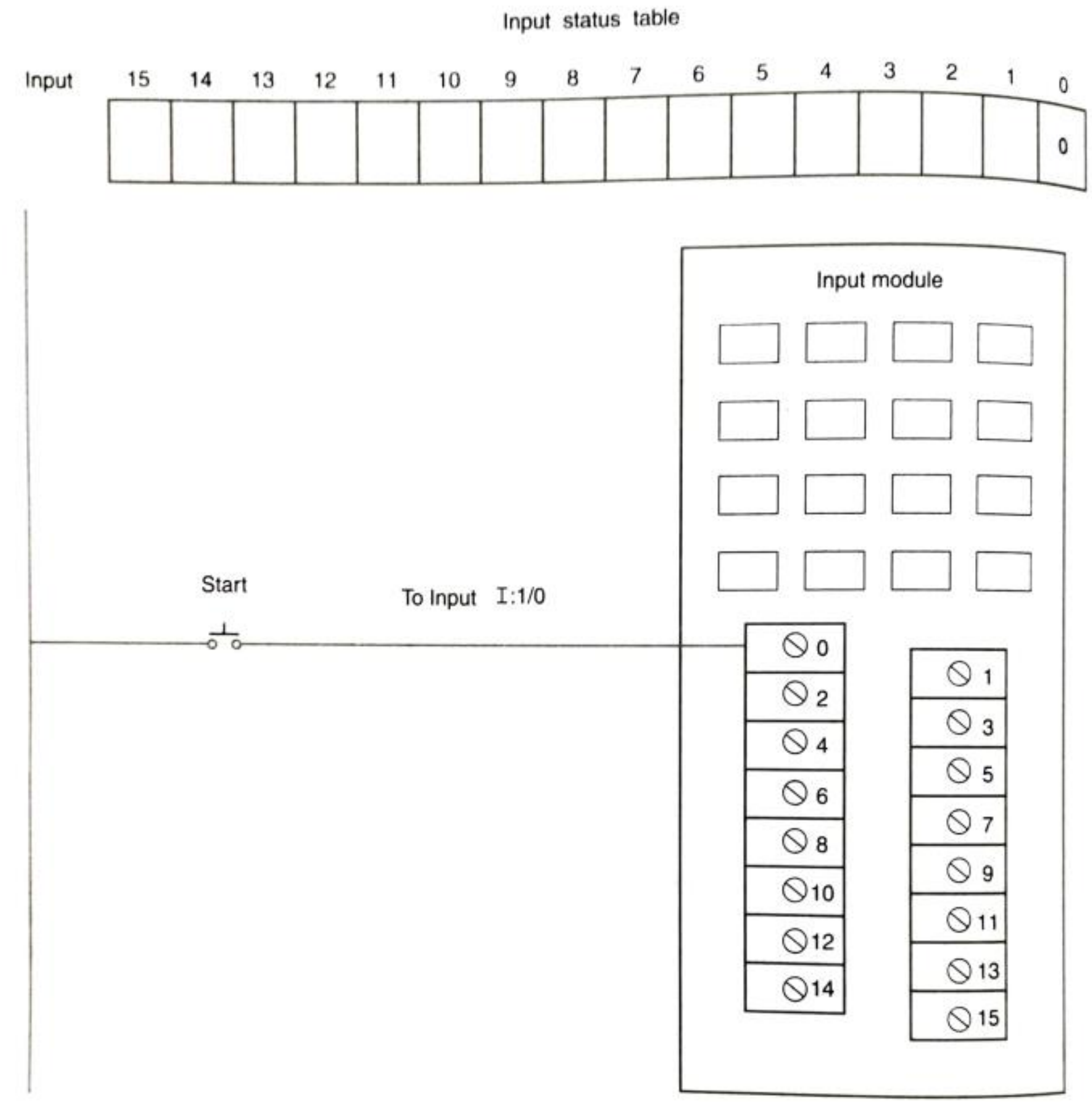


Figure 14-9 Separated inputs connected to PLC input screw terminals.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

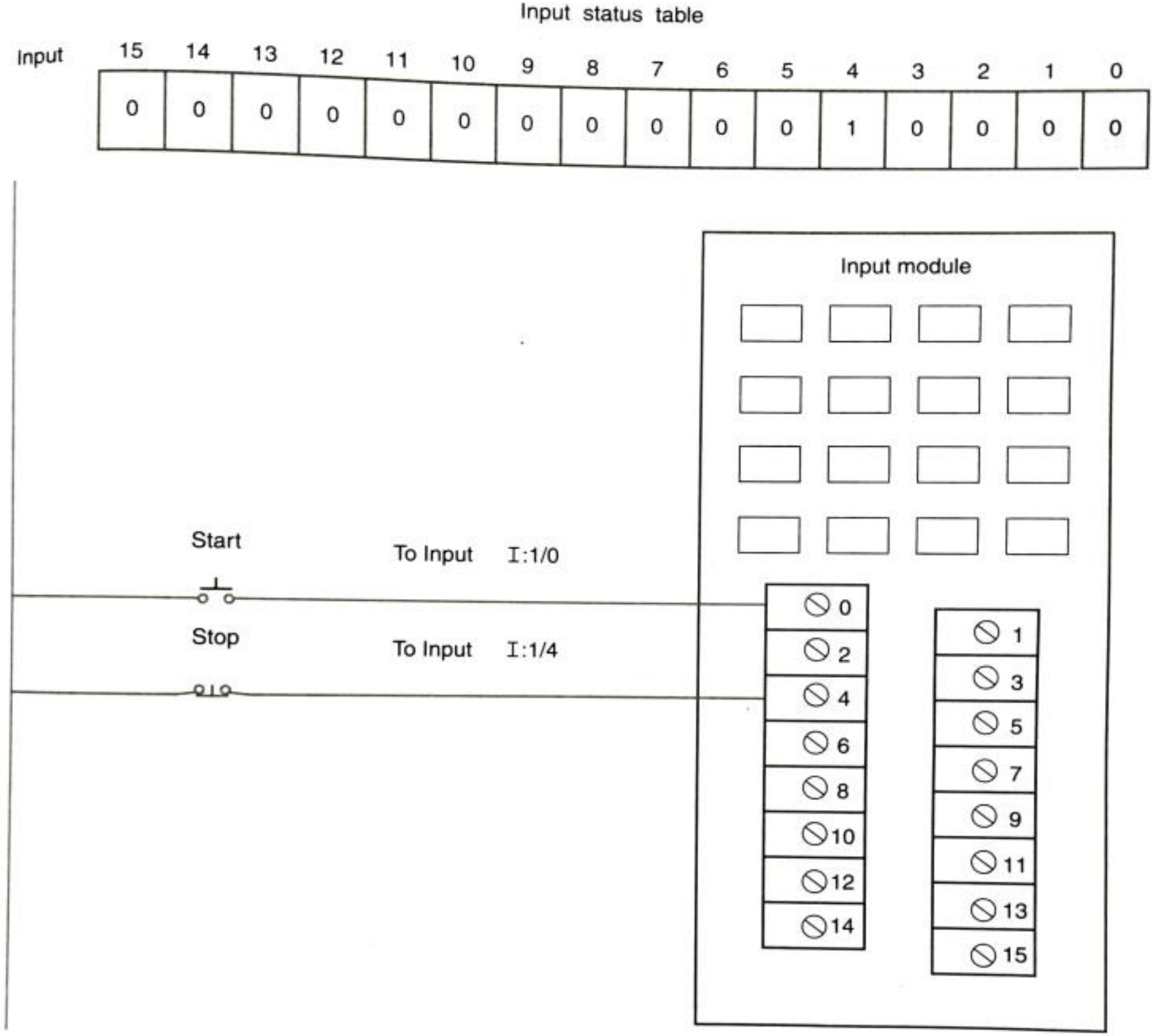


Input status table

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

Figure 14-10 Start push-button status reflected in the input status table. This OFF signal will be reflected as a 0 in this input status table position.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

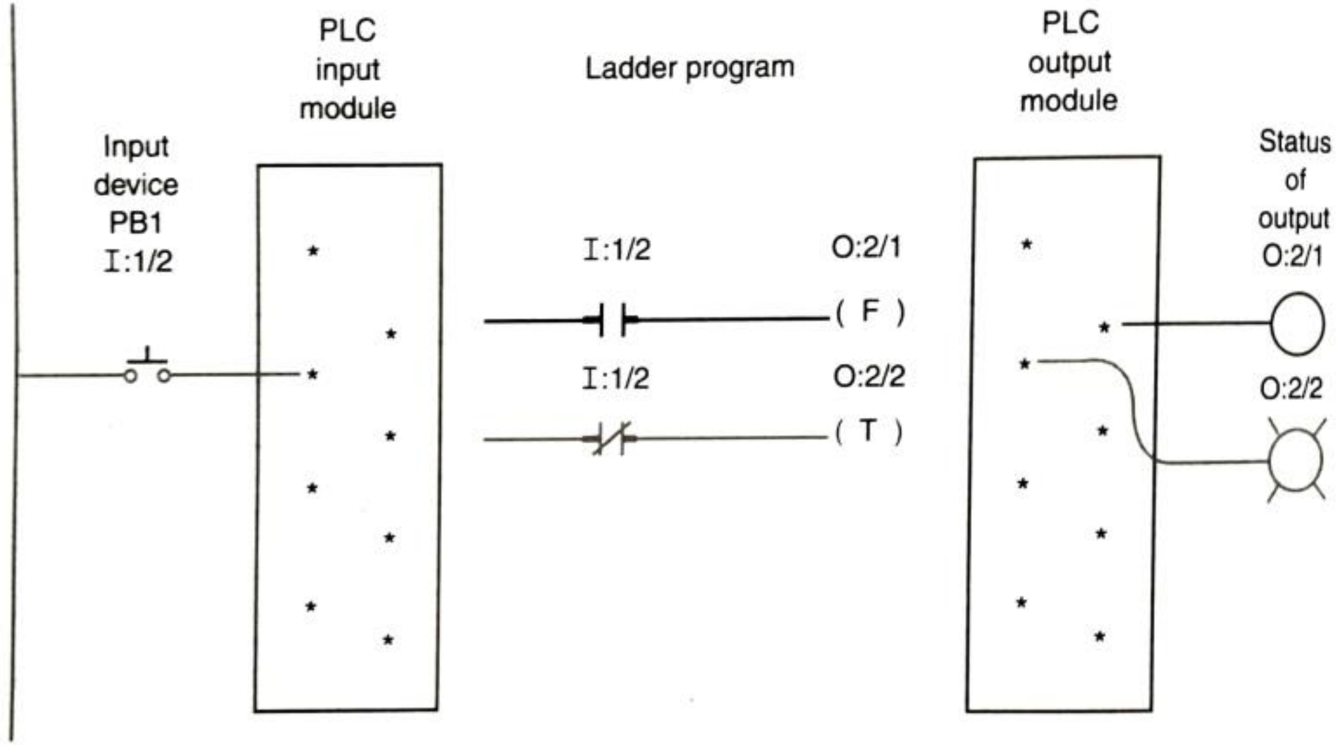


Input status table

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

Figure 14-11 Start-stop input bit status from input hardware in a nonenergized state.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)



Input status table

													2	1	0
													0		

Output status table

													2	1	0
													1	0	

Figure 14-12 Normally open push button not activated.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

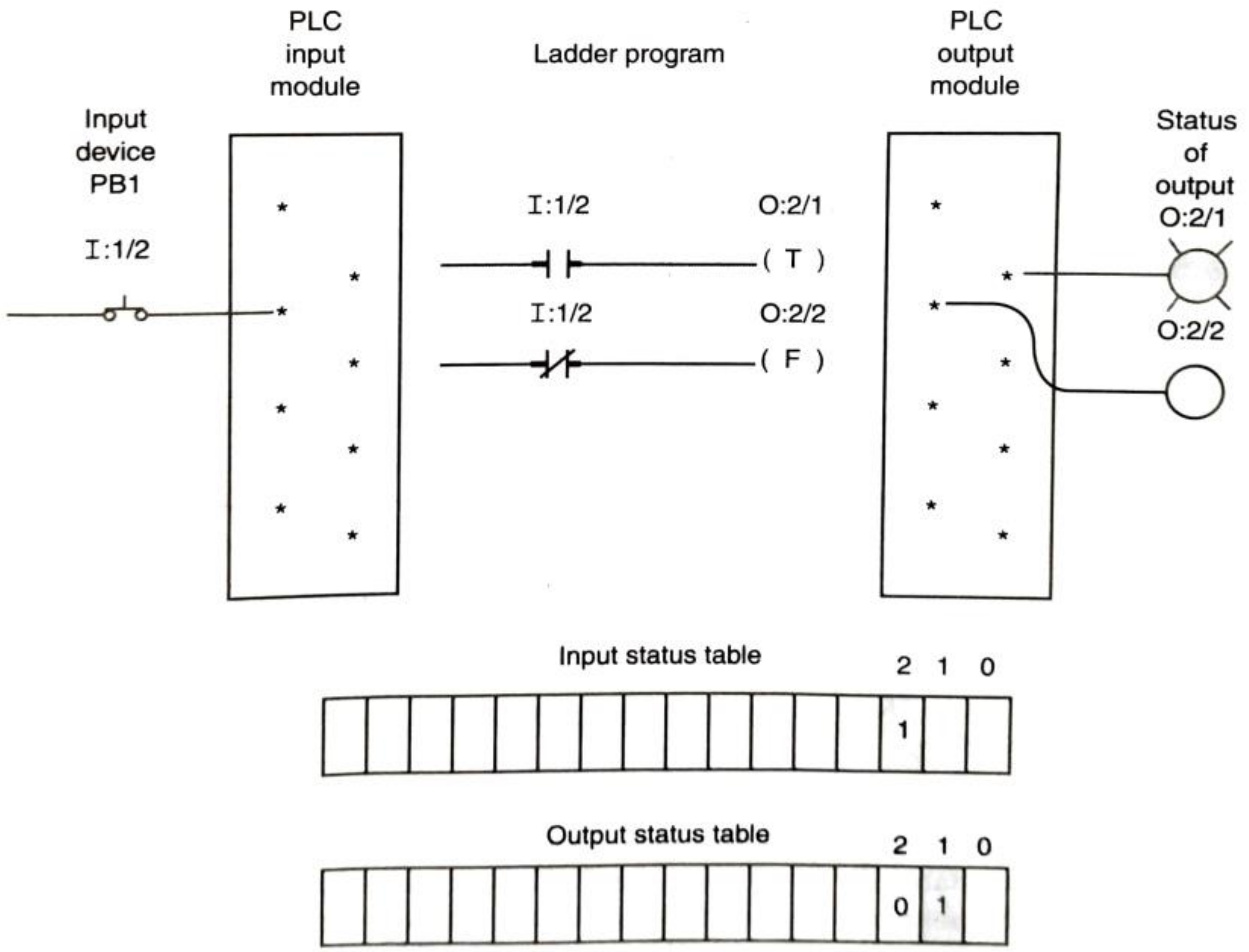
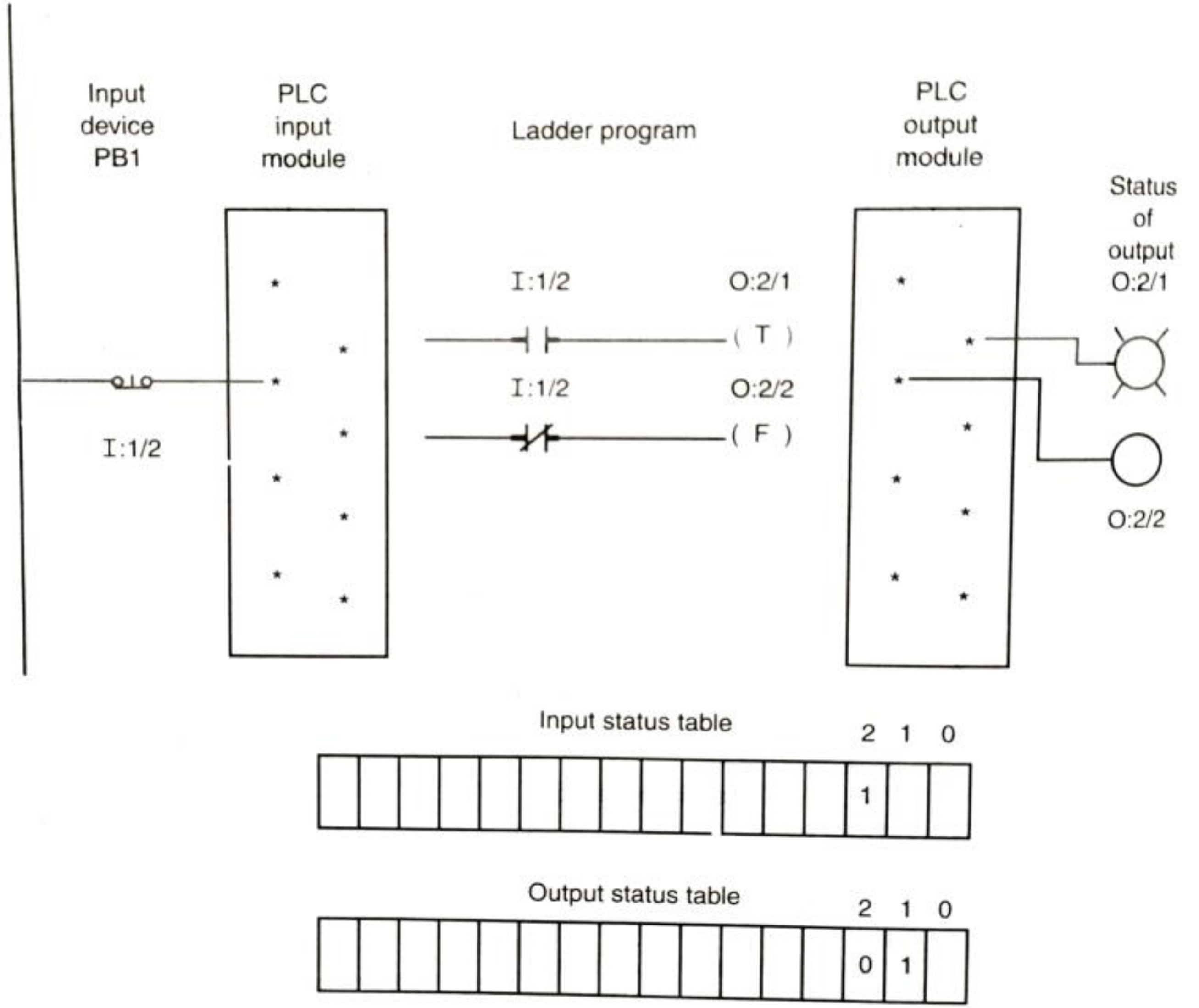


Figure 14-13 Normally open push-button input pressed, causing the normally open programming instruction to pass power, or become true.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)



Input status table

													2	1	0
													1		

Output status table

													2	1	0
													0	1	

Figure 14-14 Normally closed input push button providing a constant ON signal to the PLC input module.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

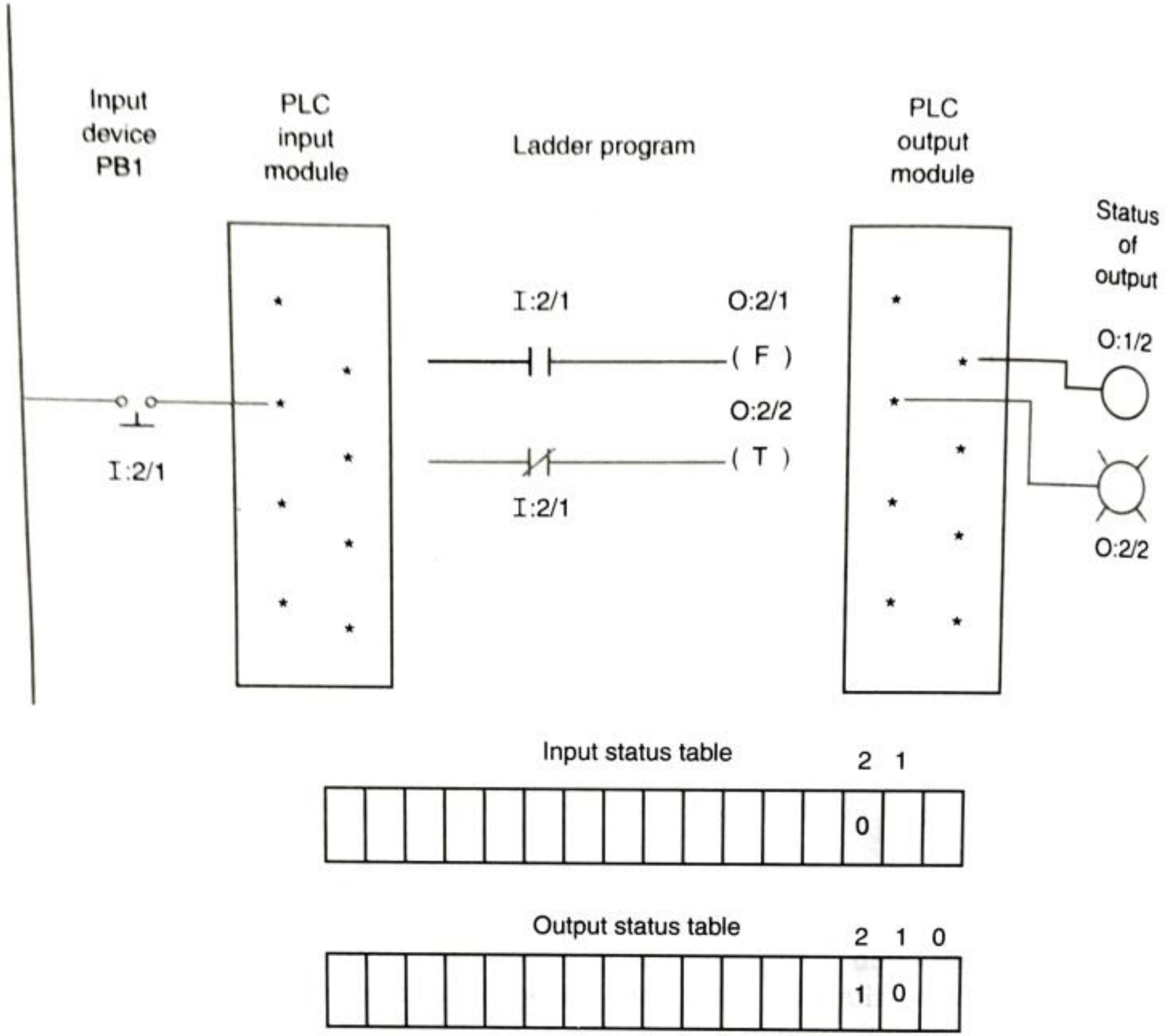


Figure 14-15 Normally closed push-button input to PLC being depressed by the operator. While being held open, no input signal is sent to the PLC input module.

Interfacing a Start-Stop Push-button Station to a Programmable Controller (Continued..)

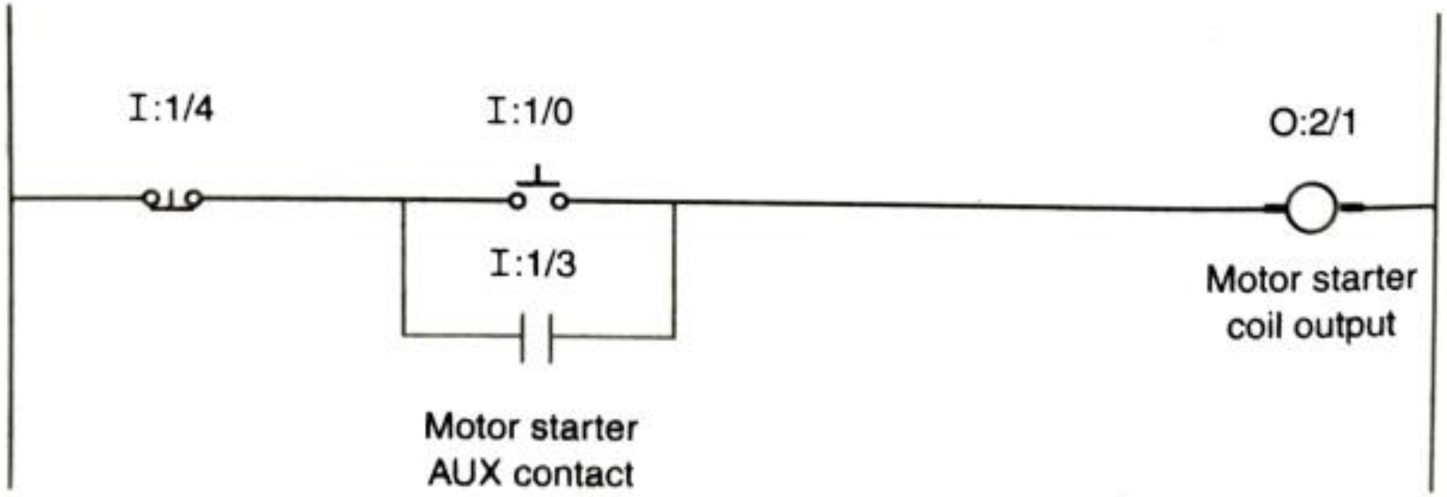


Figure 14-16 After start push button is released, the energized output will be latched through CR1-1.

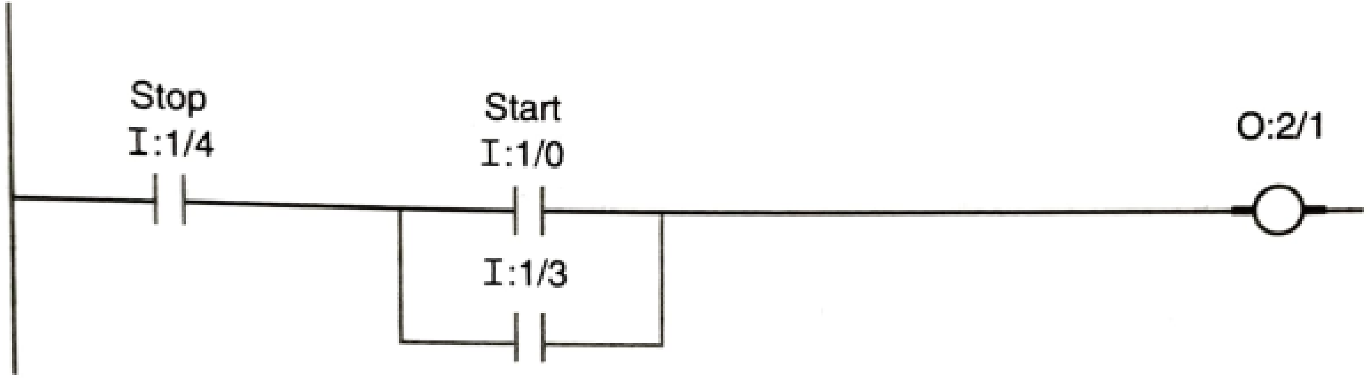


Figure 14-17 Conventional schematic start-stop logic from Figure 14-16 converted to a PLC-managed ladder program rung.

Motor Starter Circuits and the Overload Contacts

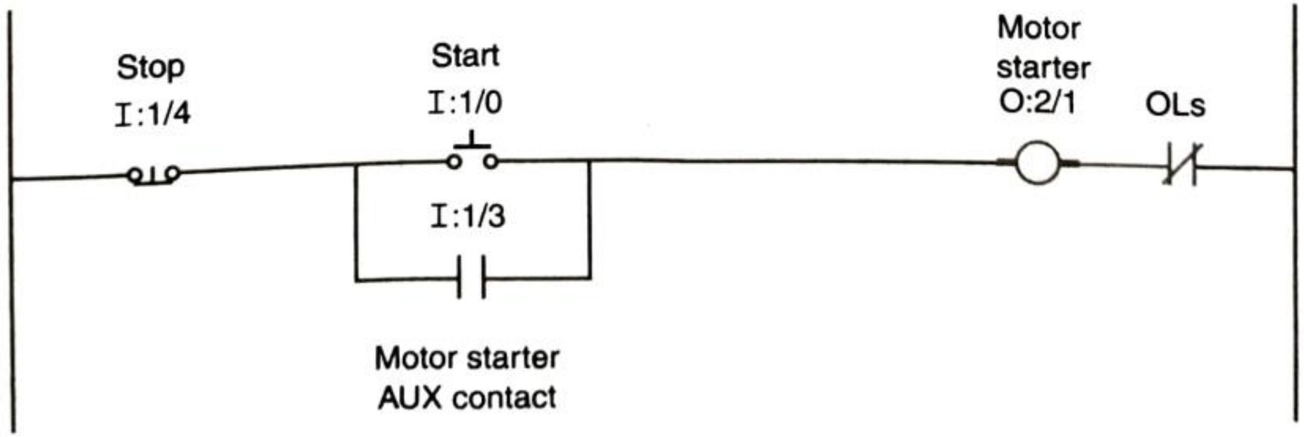


Figure 14-18 A conventional motor starter schematic diagram.

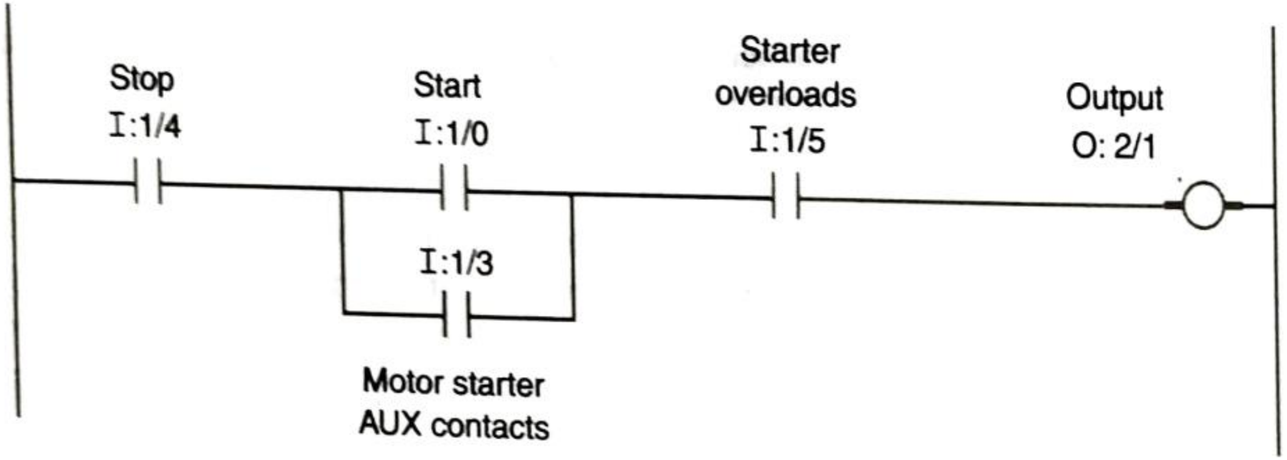


Figure 14-19 Conventional motor starter circuit converted for PLC system management.

Motor Starter Circuits and the Overload Contacts (Continued..)

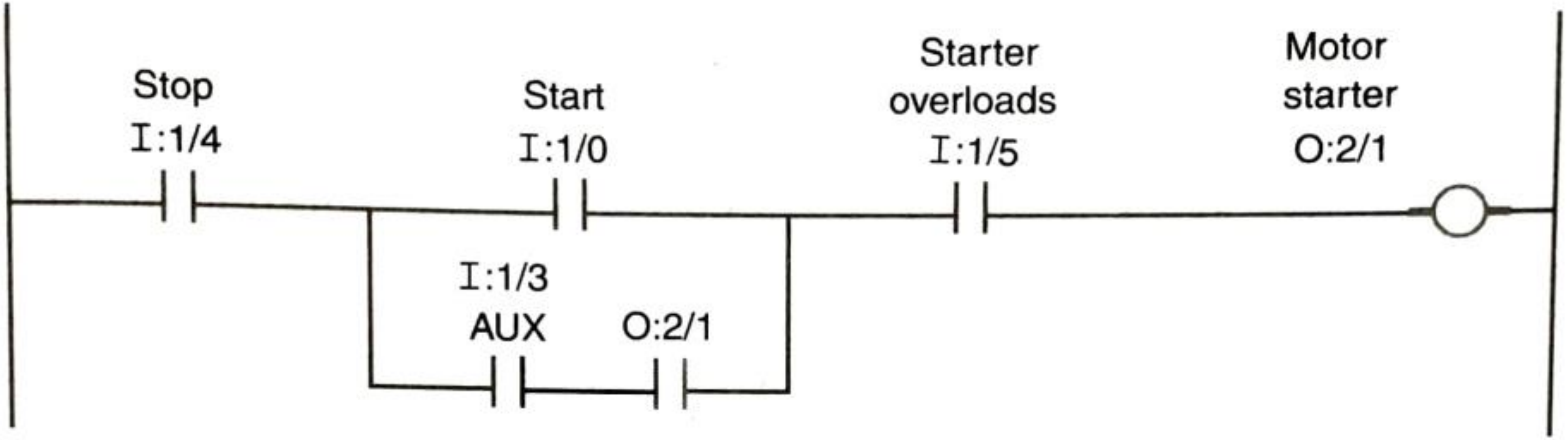


Figure 14-20 Recommended PLC ladder rung for PLC management of motor starters.

- When a PLC manages a motor starter, the logic represented in Figure 14-20 is usually recommended.

Motor Starter Circuits and the Overload Contacts (Continued..)

- This ladder rung differs by the inclusion of an additional input on the parallel branch.
- The added input instruction and its address, O:2/1, refer back to the output address instruction.
- As starters become physically larger, they become slower.
- As size increases, a starter's pickup and dropout times also increase.
- Mechanical sluggishness in larger motor starters could cause the auxiliary input to switch much more slowly than a smaller starter.

Motor Starter Circuits and the Overload Contacts (Continued..)

- A motor starter circuit has a normally open hardware auxiliary contact on the motor starter to seal the circuit in the ON state after the normally open start push button is pressed and released.
- This normally open auxiliary contact is a physical field input device that will be wired as a separate input to an unused input address.
- For this example, I:1/3 is chosen as auxiliary contact.

Motor Starter Circuits and the Overload Contacts (Continued..)

- As a result, the fast-scanning PLC may not unlatch the motor starter output even though the stop push button is depressed.
- To unlatch the rung, the latching auxiliary contact must open before the stop push button is released.
- To solve this problem, a normally open input instruction can be programmed on the latching parallel branch and refer it back to the output address.
- At the end of the first program scan, after the stop push button is depressed, the PLC will change input instruction O:2/1 to false, along with output O:2/1.

Motor Starter Circuits and the Overload Contacts (Continued..)

- With the rung now false, the slowness of the physical auxiliary contact will become irrelevant.
- The rung no longer has logical continuity because of the O:2/1 input instruction opening.

TIMER INSTRUCTIONS

- A timer consists of timer address, preset value, time base, and accumulated value.

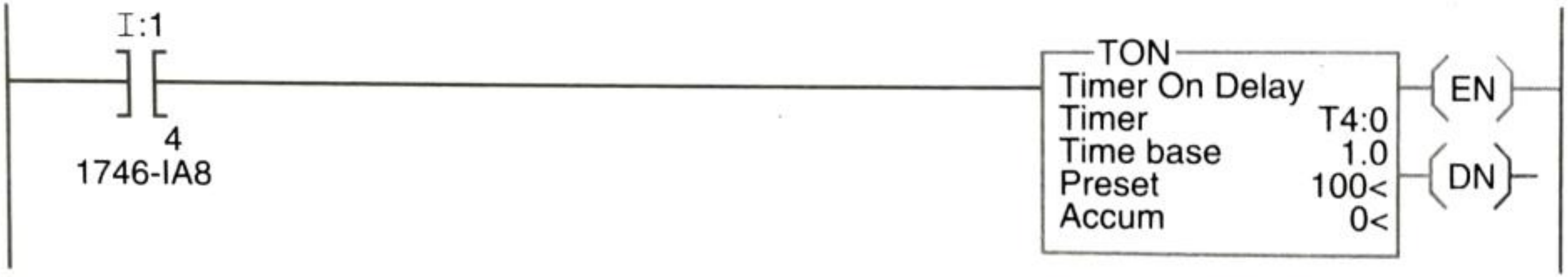


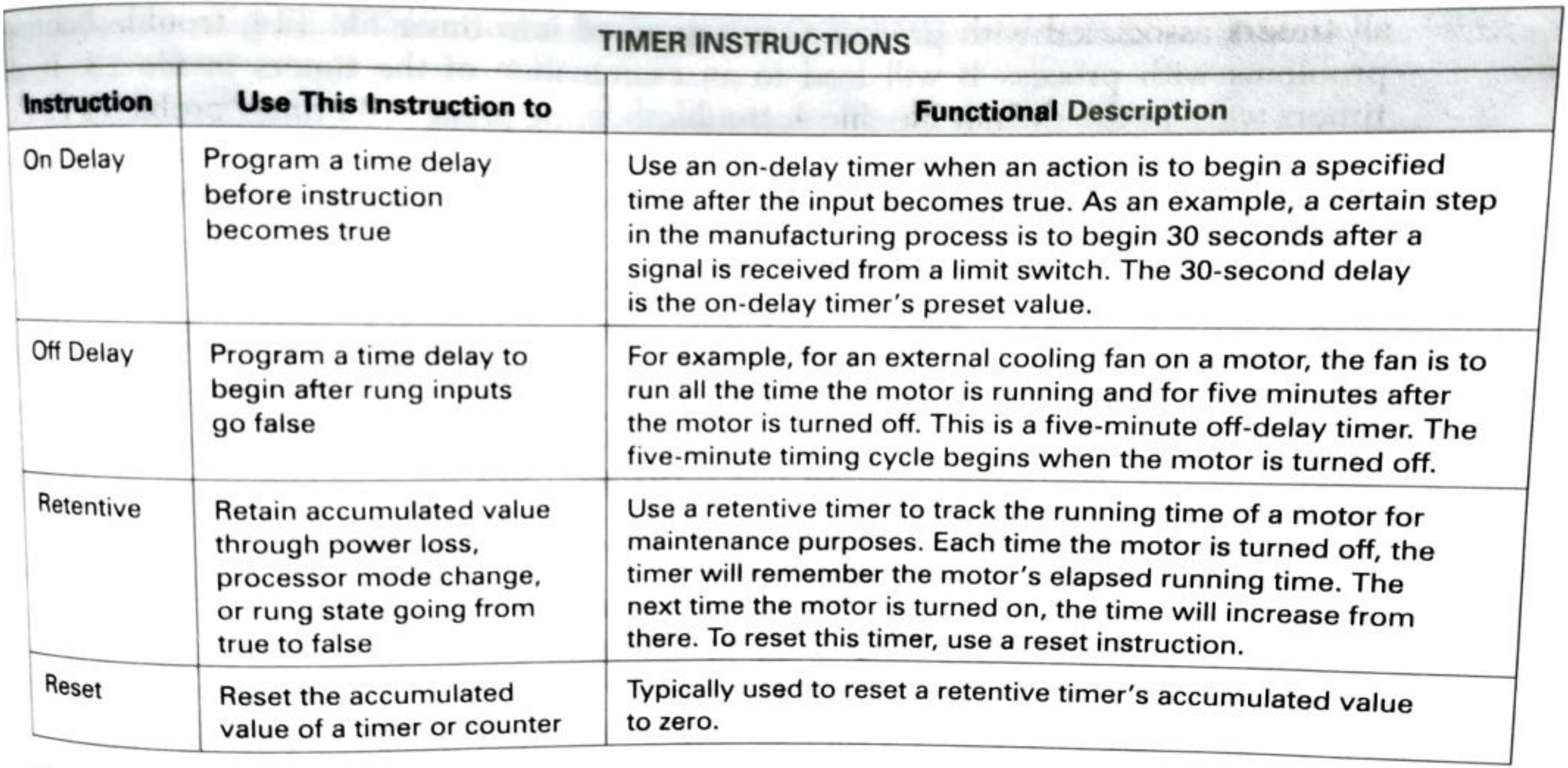
Figure 16-1 SLC 500 on-delay timer.

- In the ladder diagram with timer on delay (TON), the timer address is T4:0, time base is 1.0 seconds, preset value is 100, and the accumulated value is 0.

TIMER INSTRUCTIONS (Continued..)

- There are bits associated with the current state of the timer called status bits.
- The timer address is the timer's unique identifier in PLC memory.
- A timer's preset value is the length of time for which the timer is to run.
- The time base specifies at what rate the timer will increment.
- The time base is also referred to as the timer's accuracy.
- The accumulated value is the current elapsed time.
- The three types of timers are: on-delay, off-delay, and retentive.

TIMER INSTRUCTIONS (Continued..)



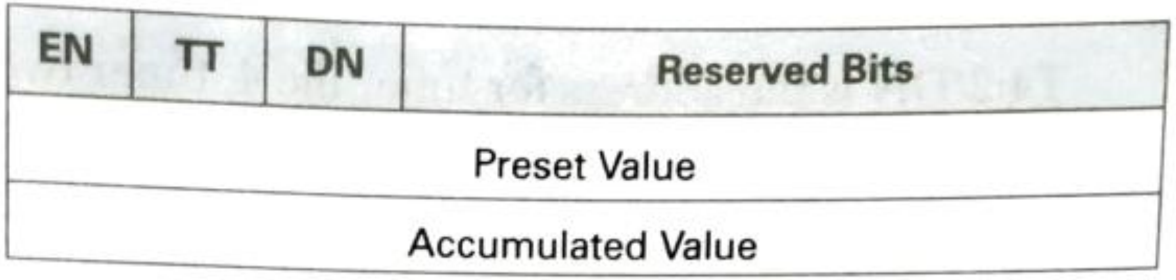
TIMER INSTRUCTIONS		
Instruction	**Use This Instruction to**	**Functional Description**
On Delay	Program a time delay before instruction becomes true	Use an on-delay timer when an action is to begin a specified time after the input becomes true. As an example, a certain step in the manufacturing process is to begin 30 seconds after a signal is received from a limit switch. The 30-second delay is the on-delay timer's preset value.
Off Delay	Program a time delay to begin after rung inputs go false	For example, for an external cooling fan on a motor, the fan is to run all the time the motor is running and for five minutes after the motor is turned off. This is a five-minute off-delay timer. The five-minute timing cycle begins when the motor is turned off.
Retentive	Retain accumulated value through power loss, processor mode change, or rung state going from true to false	Use a retentive timer to track the running time of a motor for maintenance purposes. Each time the motor is turned off, the timer will remember the motor's elapsed running time. The next time the motor is turned on, the time will increase from there. To reset this timer, use a reset instruction.
Reset	Reset the accumulated value of a timer or counter	Typically used to reset a retentive timer's accumulated value to zero.

Figure 16-2 Timer instructions common to most PLCs.

TIMER INSTRUCTIONS (Continued..)

Timer Element

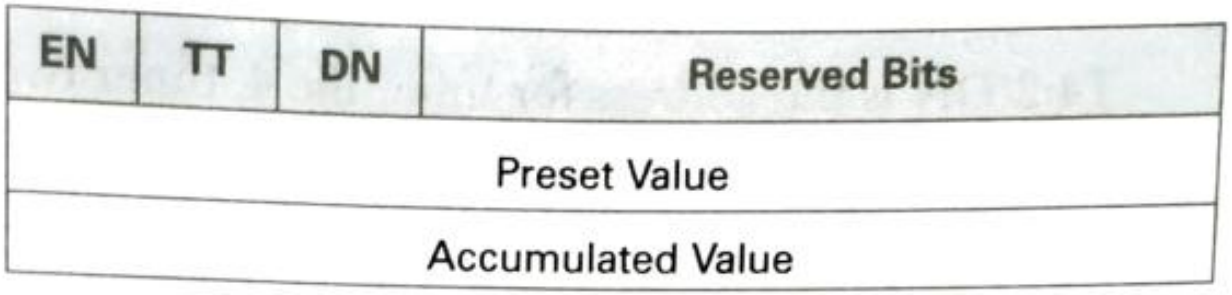
- A timer instruction is one element.
- A timer element is made up of three 16-bit words.
- Word zero contains the three status bits, Enable (EN), Timer-timing (TT), and Done (DN).
- Word one is for the preset value.
- Word two is for the accumulated value.



<table>
<tr><td>EN</td><td>TT</td><td>DN</td><td>Reserved Bits</td></tr>
<tr><td colspan="4">Preset Value</td></tr>
<tr><td colspan="4">Accumulated Value</td></tr>
</table>

Figure 16-5 One timer element is made of three 16-bit words.

TIMER INSTRUCTIONS (Continued..)



EN	TT	DN	Reserved Bits
Preset Value			
Accumulated Value			

One timer element is made of three 16-bit words.

- Word zero, bit 13, is the done bit. The done bit is set when the timer's accumulated value is equal to the preset value.

- Word zero, bit 14, is the timer-timing bit. This bit is set when the timer is timing.

- Word zero, bit 15, is the enable bit. This is set whenever the timer is enabled.

TIMER INSTRUCTIONS (Continued..)

Timer addressing

T4:2.ACC.

T = T identifies this as a timer file.

4 = This is timer file 4

:2 = This is timer two in file 4.

. = The point is called the word delimiter. The word delimiter separates the timer number from the sub element. The sub element is ACC, for accumulated value.

T4:2.PRE.

The sub element is PRE, for preset value.

TIMER INSTRUCTIONS (Continued..)

T4:2/DN is the address for timer file 4, timer two's done bit.

T4:2/TT is the address for timer file 4, timer two's timer-timing bit.

T4:2/EN is the address for timer file 4, timer two's enable bit.

THE ON-DELAY TIMER INSTRUCTION

- The on-delay timer instruction is used to program a time delay before an instruction becomes true.
- Timer addressing is as follows: T (Timer file number): (Timer element number).
- The timer address T4:0 is addressing timer file 4, timer element 0.
- The default timer file number is four.
- Each timer file may have up to 256 timer elements.

THE ON-DELAY TIMER INSTRUCTION (Continued..)

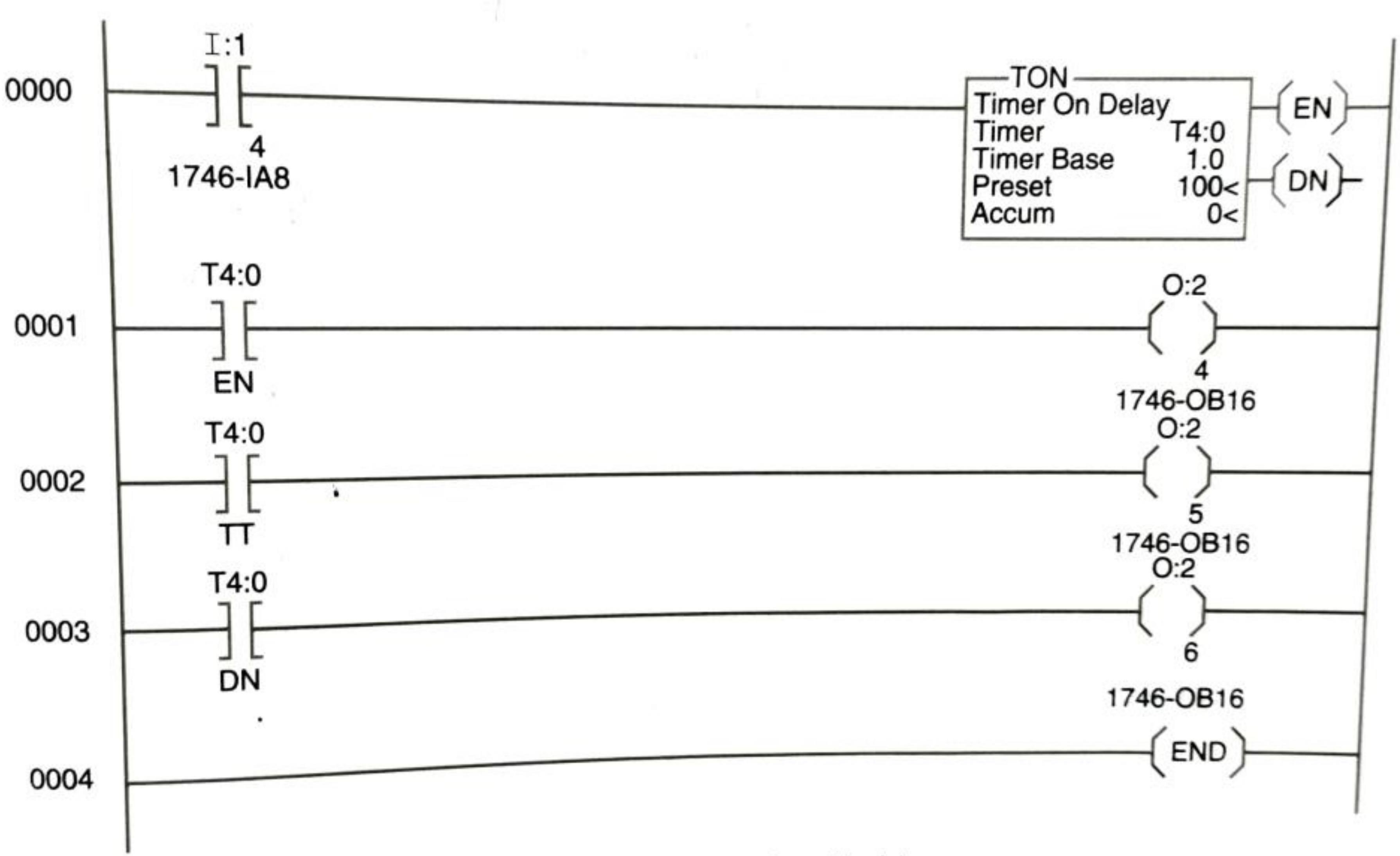


Figure 16-6 SLC 500 on-delay timer and its associated ladder rungs.

THE ON-DELAY TIMER INSTRUCTION (Continued..)

- As long as 1:1/4 is true, the timer on delay T4:0 will increment every 1 second toward the preset value of 100 seconds.
- The current number of seconds that have passed will be dis- played in the accumulated value portion of the instruction.
- When the accumulated value is equal to the preset, the timer's done bit will be energized or set.
- The timer's done bit is on rung 3.
- As long as the accumulated value is equal to the preset value, the done bit will remain set.

THE ON-DELAY TIMER INSTRUCTION (Continued..)

- As long as the done bit is set, the T4:0/DN, examine if closed, instruction on rung 3 will be true. With this instruction true, the output instruction 0:2/6 will also be true.
- An on-delay timer is not retentive. Any loss of continuity to the timer instruction on rung 0 will cause the timer to reset itself to an accumulated value of 0.

THE OFF-DELAY TIMER INSTRUCTION

- The off-delay timer instruction is used to program a time delay to begin after rung inputs go false.

Example:

- An external cooling fan on a motor is to run all the time the motor is running and for 100 seconds after the motor is turned off. This involves a 100-second off-delay timer. The 100-second timing cycle begins when the motor is turned off.

THE OFF-DELAY TIMER INSTRUCTION (Continued..)

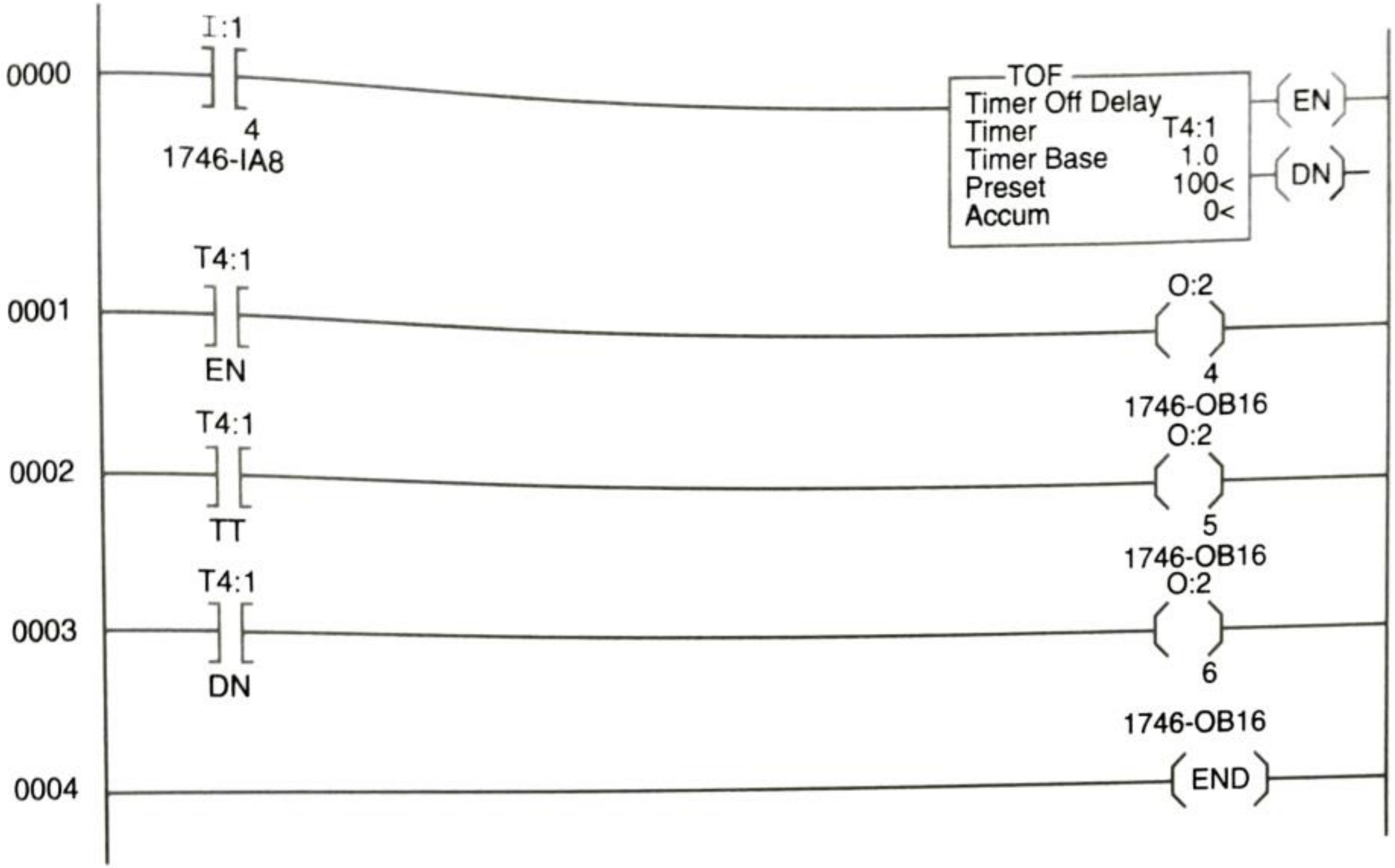


Figure 16-7 SLC 500 off-delay timer and its associated ladder rungs.

THE OFF-DELAY TIMER INSTRUCTION (Continued..)

- When I:1/4 becomes true, the TOF T4:1 instruction becomes true. The done bit is set as long as the rung is true. As a result, O:2/6, the external cooling fan output, energizes as soon as the rung goes from false to true.
- When the motor is turned off, rung 0 transitions from true to false, and the TOF instruction begins timing.
- The done bit and the external cooling fan's output (O:2/6) will re- main on, or true, for the preset value of 100 seconds.
- The time period between the point when the rung becomes false and the point when the 100-second preset time expires for T4:1 is called the delay after the input goes false, or the off delay.

THE OFF-DELAY TIMER INSTRUCTION (Continued..)

- The done bit and its associated output stay true until the off delay of 100 seconds expires. The time expires when the accumulated value reaches the preset value.

THE RETENTIVE TIMER INSTRUCTION

- This instruction is used to retain accumulated value through power loss, processor mode change, or change in the rung state from true to false.
- The retentive timer will measure the cumulative time period for which its rung is true.
- The retentive timer on RTO instruction behaves similar to the timer on delay with the exception that when the RTO instruction goes false, it will retain, or remember its accumulated value.

THE RETENTIVE TIMER INSTRUCTION (Continued..)

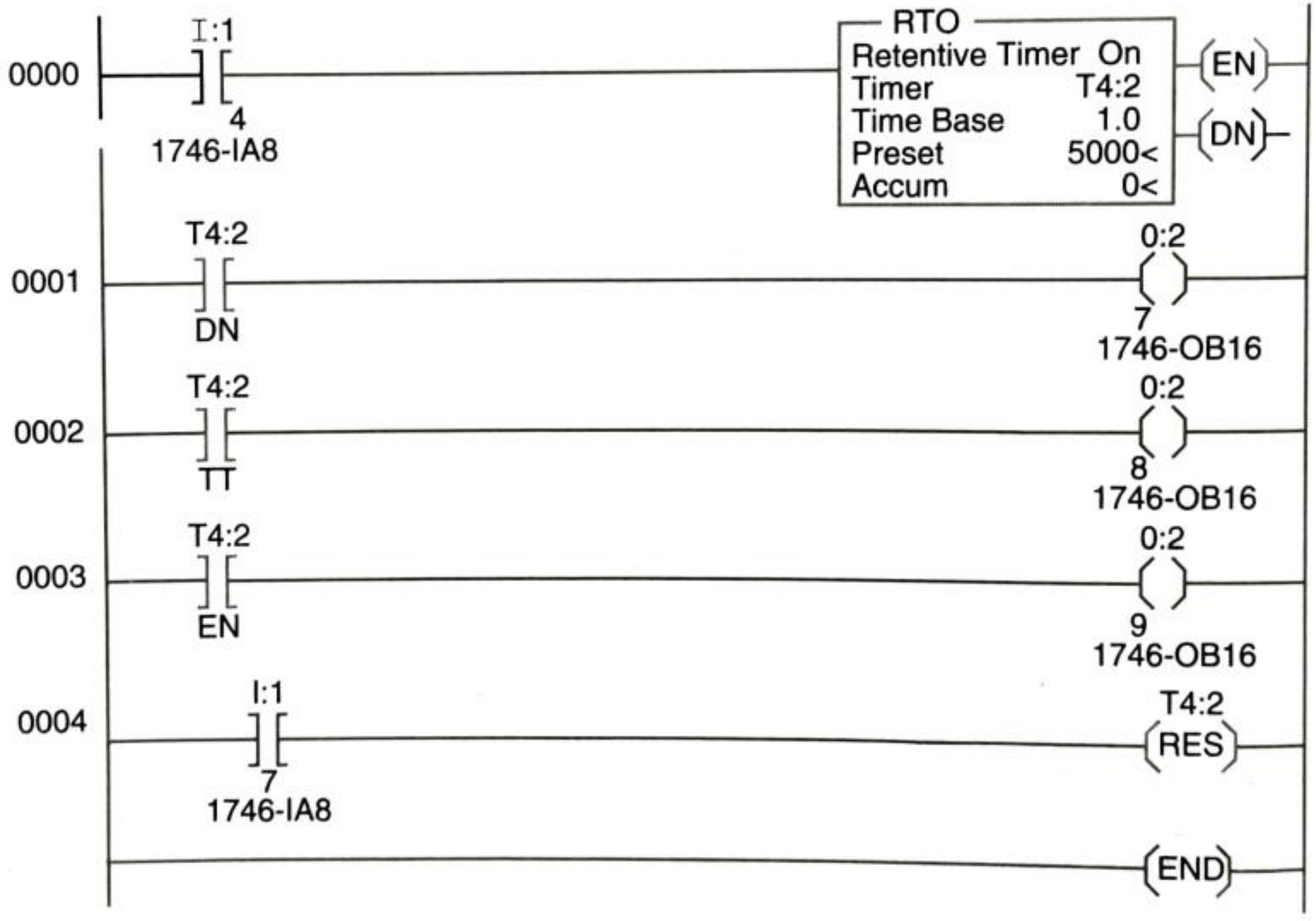


Figure 16-8 SLC 500 retentive timer and its associated ladder rungs.

THE RETENTIVE TIMER INSTRUCTION (Continued..)

- The timer done bit is set when the accumulated value is equal to the preset value.
- In this application, the done bit and output 0:2/7 could control the maintenance reminder pilot light.
- The timer-timing bit is on rung two. This bit is set any time the rung conditions are true and the timer times.
- The timer times whenever the rung is true and the accumulated value is less than the preset value.

THE RETENTIVE TIMER INSTRUCTION (Continued..)

- When the done bit is set, the timer- timing bit is reset as well. In Figure 16-8, the timer-timing bit will be true, or set, when- ever input 1:1/4 is true and as long as the accumulated value is less than the preset value of 5,000 seconds.
- Output O:2/8 will be on when the timer is timing between 0 seconds and 5,000 seconds.
- As the done bit is set and O:2/7 turns on, the timer-timing bit goes false and O:2/8 turns off.

THE RETENTIVE TIMER INSTRUCTION (Continued..)

- The timer enable bit, bit 15, is set, or true, anytime the timer instruction's rung 0 is true. As long as I:1/4 is true, the timer instruction is considered enabled. The enabled bit will be true when the timer-timing bit is true. The timer enable bit will stay set through the transition from the timer-timing bit to the timer done bit.
- As long as there is continuity through all input instructions to the timer instruction, no matter the relationship between the preset and accumulated values, the enable bit will be reset when the rung goes false.

THE RESET INSTRUCTION

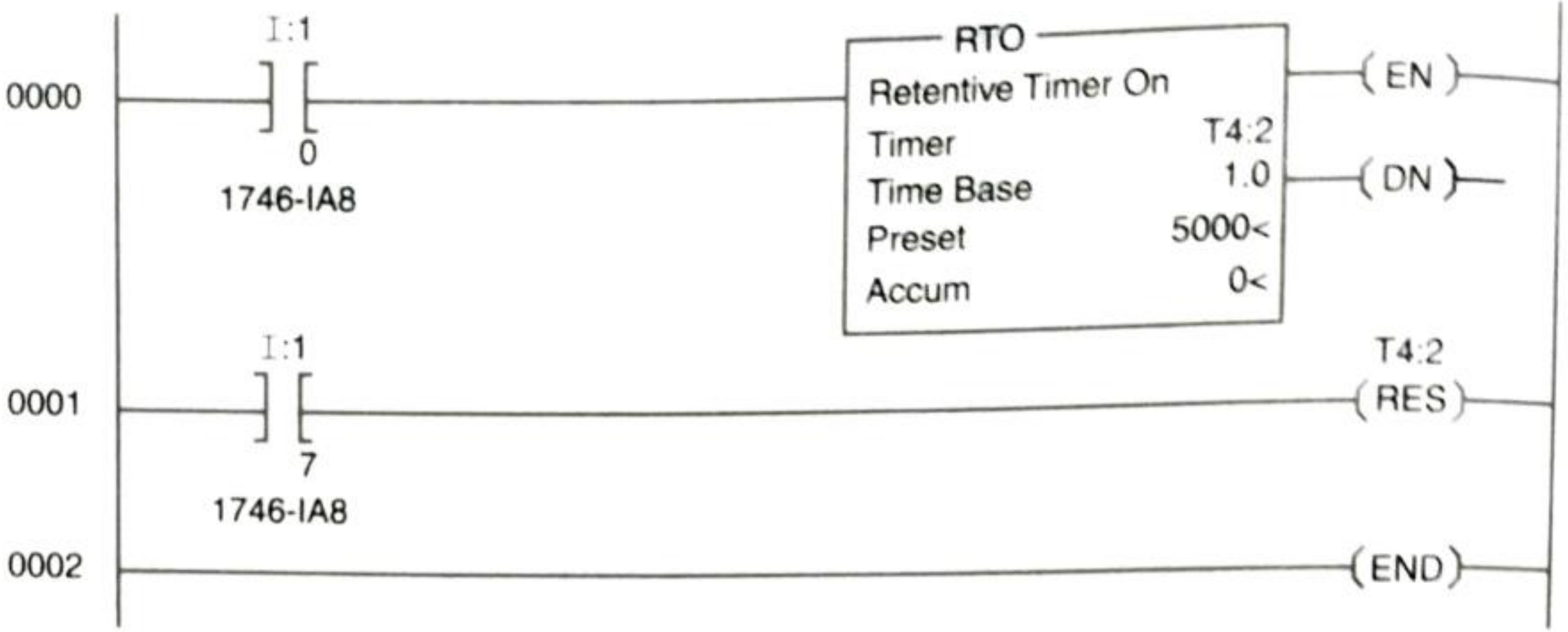
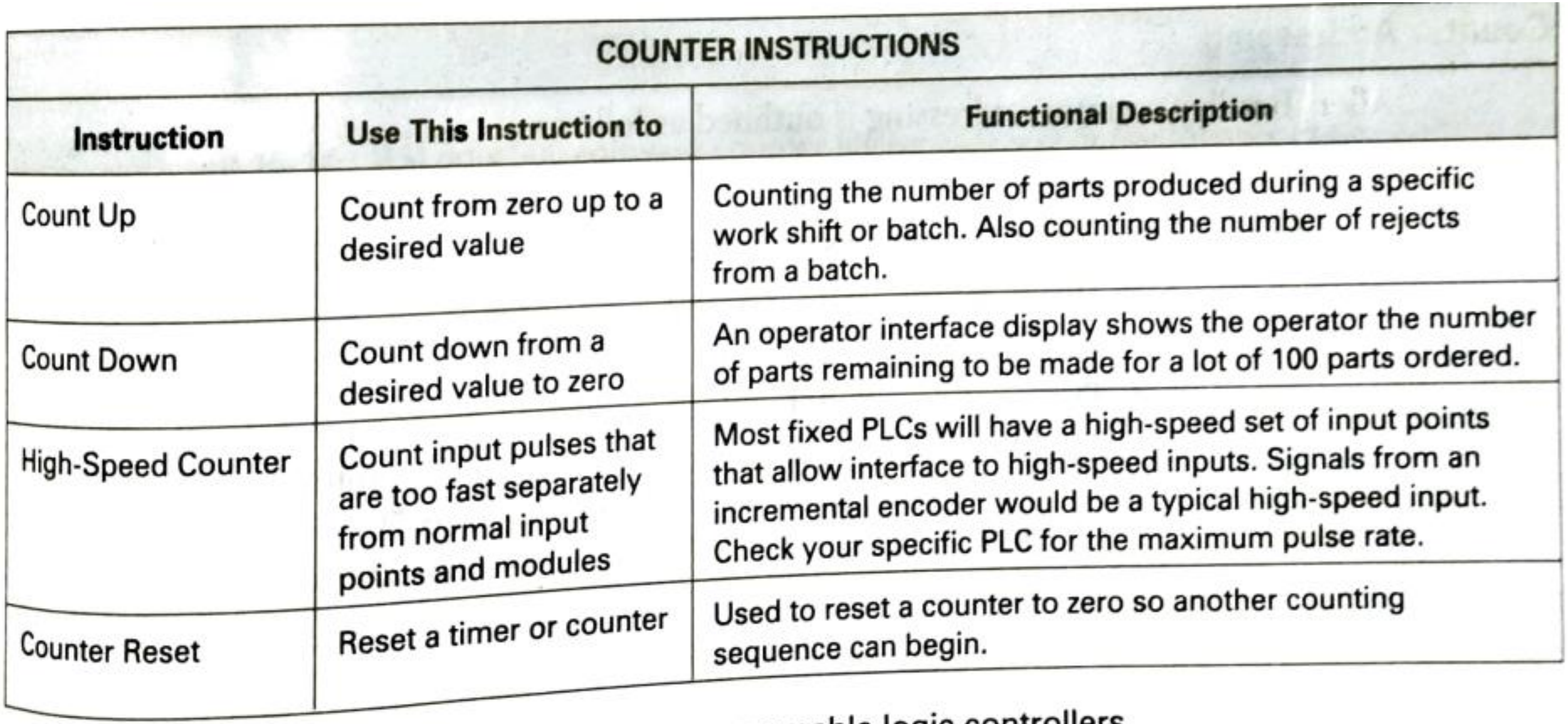


Figure 16-9 Retentive timer and its reset instruction.

THE RESET INSTRUCTION (Continued..)

- Use this instruction if you want to reset the accumulated value of a retentive timer.
- When a maintenance reminder pilot light comes on and a maintenance individual completes the required preventive maintenance, he or she will depress the push button connected to input I:1/7.
- Input I:1/7 is the reset maintenance timer signal. This will be a momentary normally open push-button field device. Pressing this button will reset the RTO's accumulated value back to 0.

PLC COUNTER INSTRUCTIONS



COUNTER INSTRUCTIONS		
Instruction	**Use This Instruction to**	**Functional Description**
Count Up	Count from zero up to a desired value	Counting the number of parts produced during a specific work shift or batch. Also counting the number of rejects from a batch.
Count Down	Count down from a desired value to zero	An operator interface display shows the operator the number of parts remaining to be made for a lot of 100 parts ordered.
High-Speed Counter	Count input pulses that are too fast separately from normal input points and modules	Most fixed PLCs will have a high-speed set of input points that allow interface to high-speed inputs. Signals from an incremental encoder would be a typical high-speed input. Check your specific PLC for the maximum pulse rate.
Counter Reset	Reset a timer or counter	Used to reset a counter to zero so another counting sequence can begin.

Figure 16-12 Common counters found in programmable logic controllers.

WORKING OF COUNTERS

- The counter instruction counts each time the input logic changes the rung from false to true.
- Input logic can be a signal coming from an external device, such as a limit switch or sensor, or a signal from internal logic.
- Each time the counter instruction sees a false-to-true rung transition, a count-up counter's accumulated value is incremented by one.
- Each time a count-down counter sees a false-to-true rung transition, its accumulated value is decremented by 1.
- Counting range is the numerical range within which a counter can count.
- The counters can count within the range of -32,768 to +32,767. This is the range of a 16-bit signed integer.

WORKING OF COUNTERS (Continued..)

- If a counter counts above +32,767, an overflow is detected and the overflow bit is set.
- Conversely, if a down-counter counts below -32,768, an underflow is detected and the underflow bit is set.
- In either an overflow or underflow, the counter will set the appropriate bit, wrap around, and begin counting from the other end of the counting range.
- To avoid overflowing or underflowing a counter, a reset instruction can be used to reset the counter back to 0.
- Counters are retentive. Assuming that the processor's battery is in good condition, a counter will retain its accumulated value and the on or off status of the done, overflow, and underflow bits through a power loss.

PLC COUNTER INSTRUCTIONS (Continued..)

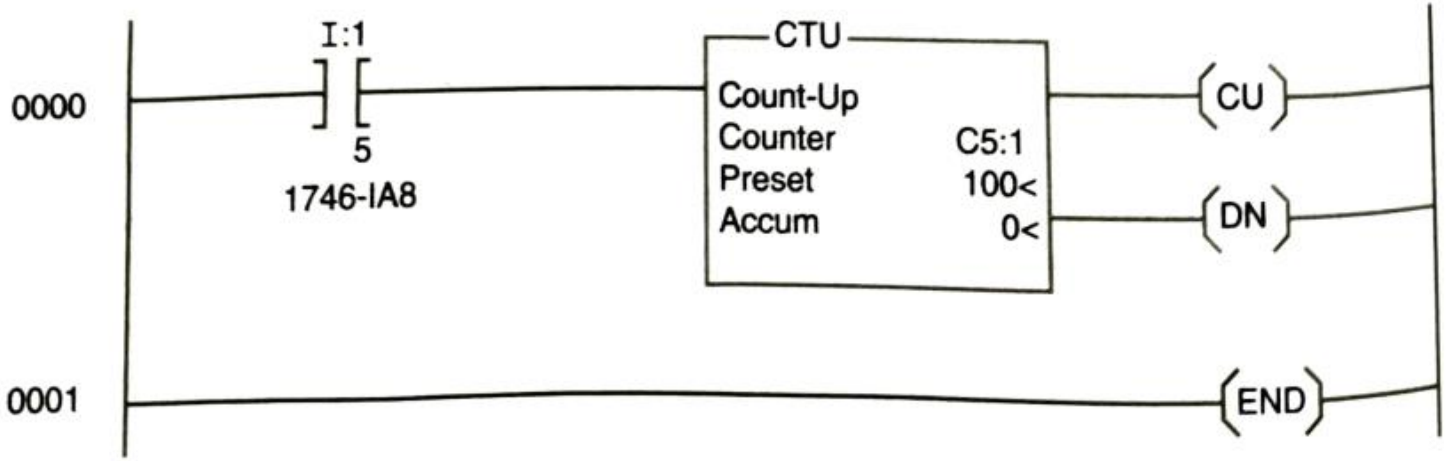


Figure 16-10 An RSLogix 5 or 500 count-up counter.

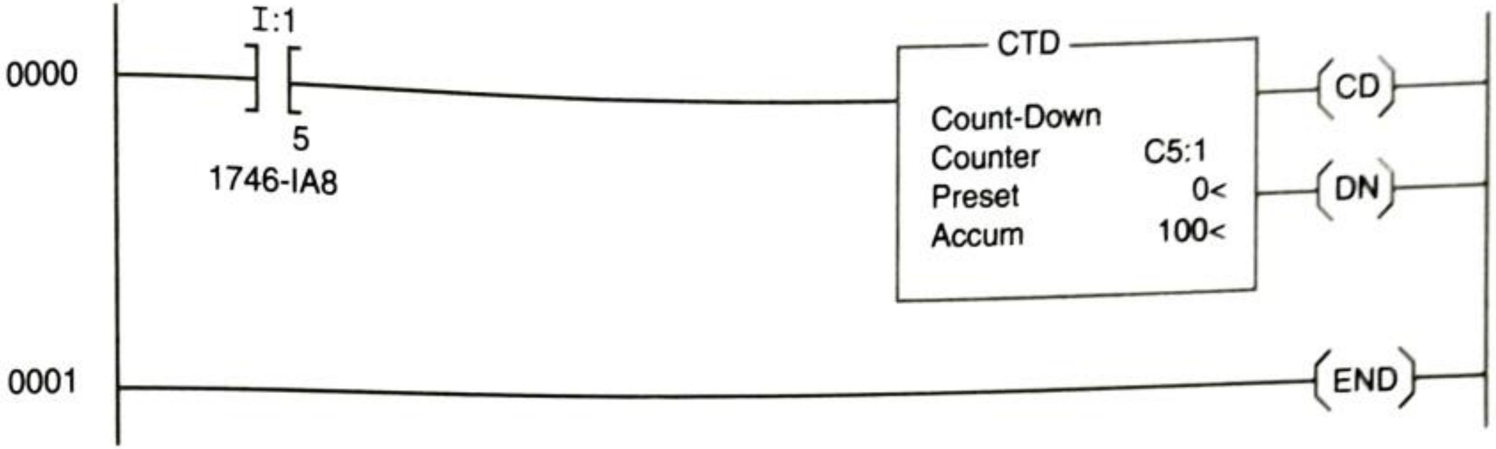
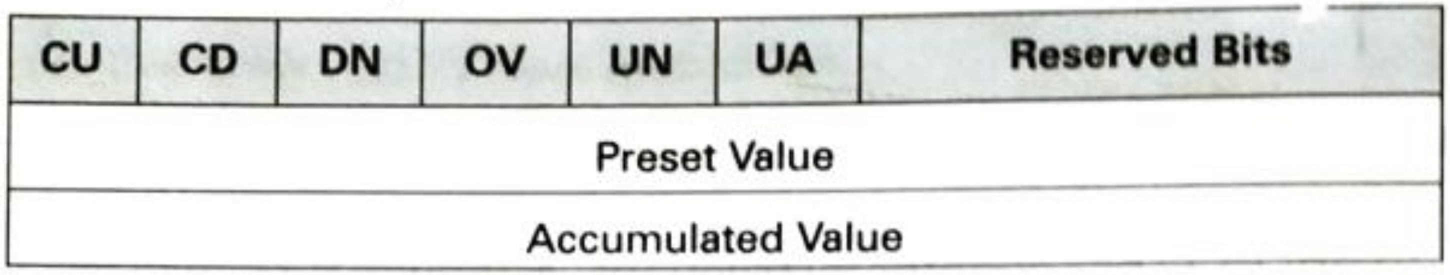


Figure 16-11 An SLC 500 count-down counter.

PLC COUNTER INSTRUCTIONS (Continued..)

- A up-counter counts from the accumulated value of 0 up to the preset value of 100.
- A down-counter counts from the accumulated value of 100 down to the preset value of 0.
- The default counter file is file 5.
- The counter file is used to store counter data.

PLC COUNTER INSTRUCTIONS (Continued..)

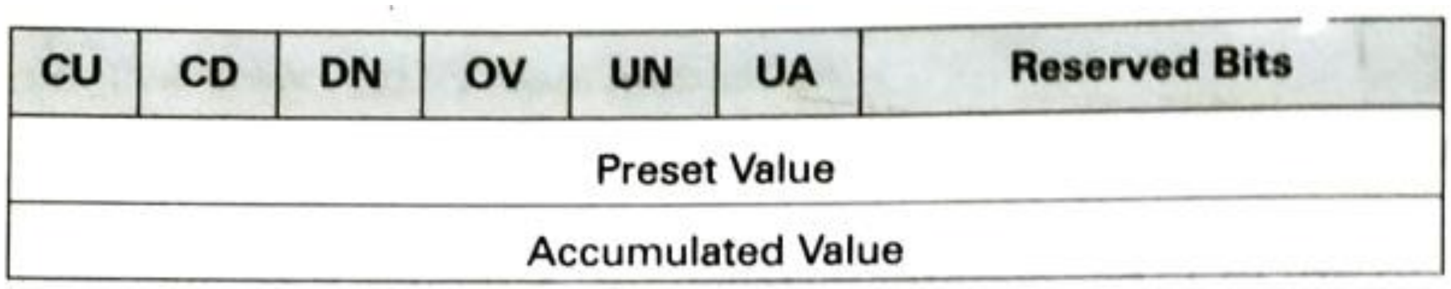


<table>
<tr><td>CU</td><td>CD</td><td>DN</td><td>OV</td><td>UN</td><td>UA</td><td>Reserved Bits</td></tr>
<tr><td colspan="7">Preset Value</td></tr>
<tr><td colspan="7">Accumulated Value</td></tr>
</table>

Figure 16-13 One timer element is made of three 16-bit words.

- Each counter consists of three 16-bit words and is called a "counter element."
- A counter instruction is one element.
- A counter element is made up of three 16-bit words.
- Word zero is for status bits. Status bits are CU, CD, DN, OV, UN, and UA.
- Word one is for the preset value.
- Word two is for the accumulated value.

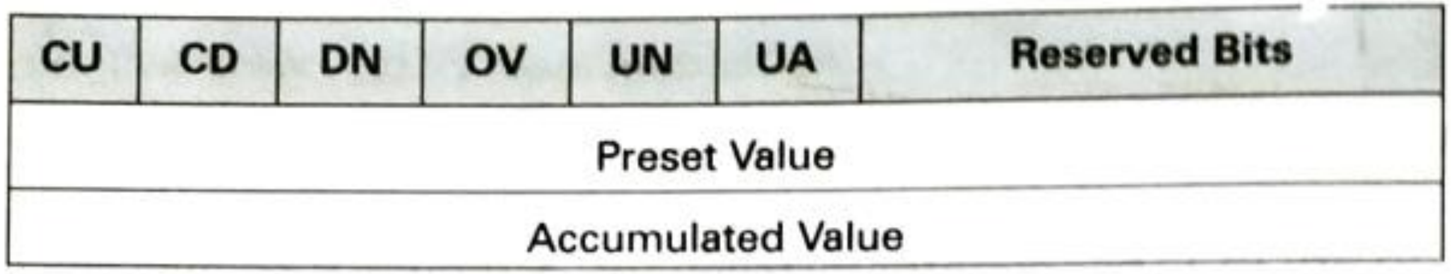
PLC COUNTER INSTRUCTIONS (Continued..)



<table>
<tr><td>CU</td><td>CD</td><td>DN</td><td>OV</td><td>UN</td><td>UA</td><td>Reserved Bits</td></tr>
<tr><td colspan="7">Preset Value</td></tr>
<tr><td colspan="7">Accumulated Value</td></tr>
</table>

- Counters have six status bits. Word zero, bits 0 through 9, are reserved for the processor.
- Word zero, bit 10, is the update accumulator bit, identified as UA.
- Bit 11 is the underflow bit, identified as UN. The underflow bit is set when the accumulated value of a count-down counter has reached the lowest possible accumulated value, that is -32,768. The counter will automatically wrap around and start counting down from the maximum positive value, +32,767.

PLC COUNTER INSTRUCTIONS (Continued..)



CU	CD	DN	OV	UN	UA	Reserved Bits
Preset Value						
Accumulated Value						

- Bit 12 is the overflow bit, identified as OV. The overflow bit is set when the accumulated value of a count-up counter has reached the highest possible accumulated value which is +32,767. The counter will automatically wrap around and start counting up from the maximum negative value, -32,768.

- Bit 13 is the done bit, identified as DN. The done bit is set when the counter's accumulated value is equal to or greater than the preset value.

- Bit 14 is the count-down-enabled bit, identified as CD. This bit is set when counting down and the rung conditions are true.

- Bit 15 is the count-up-enabled bit. The count-up-enabled bit is identified as CU. This bit is set when counting up and the rung conditions are true.

Counter Addressing

C5:3.

C = C identifies this as a counter file.

5 = This is counter file 5.

:3 = The colon is called the file separator. Counter 3 in counter file 5.

C5:12.ACC.

:12 = Counter 12 in counter file 5

. = The point is called the word delimiter. The subelement is ACC for the accumulated value.

C5:12.PRE.

The subelement is PRE for preset value.

PLC COUNTER INSTRUCTIONS (Continued..)

The addressing format for counter status bits are as follows:

C5:12/DN is the address for counter file 5, counter 12's done bit.

C5:12/CU is the address for counter file 5, counter 12's count-up-enabled bit.

C5:12/CD is the address for counter file 5, counter 12's count-down-enable bit.

THE COUNT-UP INSTRUCTION

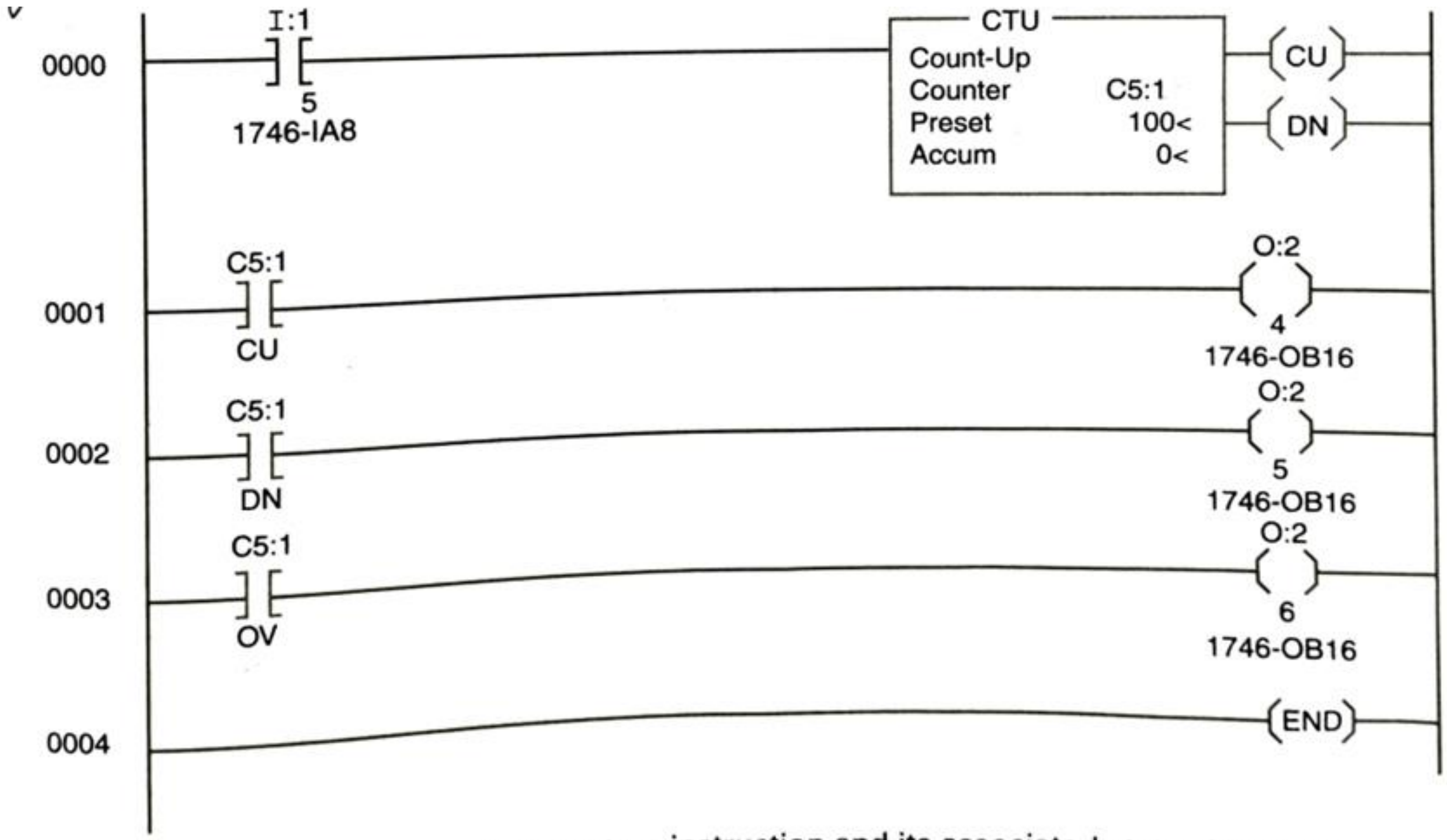


Figure 16-15 RSLogix 5 or 500 count-up instruction and its associated status bit rungs.

THE COUNT-UP INSTRUCTION (Continued..)

- Use the count-up instruction if you want a counter to increment one decimal value each time it registers a rung transition from false to true.
- Each time input 1:1/5 transitions from off to on, counter C5:1 will increment its accumulated value by one decimal value.
- The count-up overflow bit, OV, on rung 0003, is set whenever the count-up counter's accumulated value wraps from +32,767 around to -32,768.
- The done bit (DN, on rung 0002) is set when the accumulated value is equal to or greater than the preset value.
- The count-up-enabled bit, on rung 0001, is set when the rung conditions are true, or enabled.

THE COUNT-DOWN INSTRUCTION

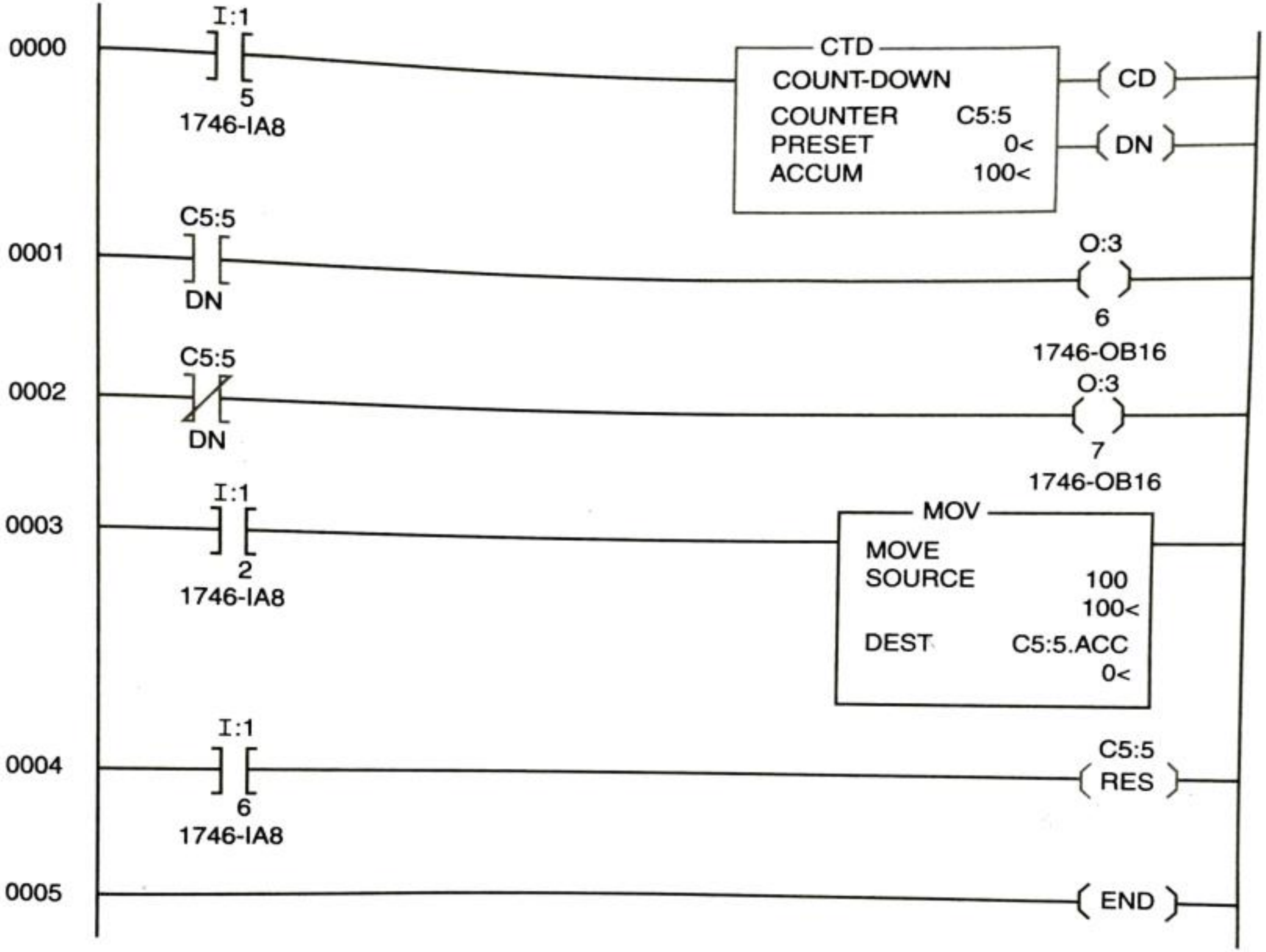


Figure 16-16 RSLogix 5 or 500 count-down counter and its associated status bits.

THE COUNT-DOWN INSTRUCTION (Continued..)

- Use this instruction if you want to count down over the range of +32,767 to -32,768. Each time the instruction sees a false-to-true transition, the accumulated value will be decremented by one count.
- The accumulated value for count-down counter C5:5 is set at 100. The preset value is 0. Each time input I:1/5 transitions from off to on, the accumulated value will decrement one decimal value.
- The done bit will be set, or on, during the entire count from 100 to 0. If the done bit is programmed as an examine if closed instruction (as on rung 0001 in Figure 16-16), the instruction will be true during the entire count-down sequence.

THE COUNT-DOWN INSTRUCTION (Continued..)

- If, on the other hand, the done bit was programmed as an examine if open instruction (as on rung 0002 in Figure 16-16), the output would be off, or false, during the count-down cycle. The done bit will change state when the counter's accumulated value transitions from 0 to -1.
- Rung 0004 in Figure 16-16 contains the reset instruction for count-down counter. Resetting the counter will cause the accumulated value to be reset to 0. To replace the accumulated value of 100 back into the counter's accumulator, the move instruction on rung 0003 will be used.
- The move source is integer 100. The move instruction's destination for the integer 100 is C5:5 ACC.

THE COUNTER RESET INSTRUCTION

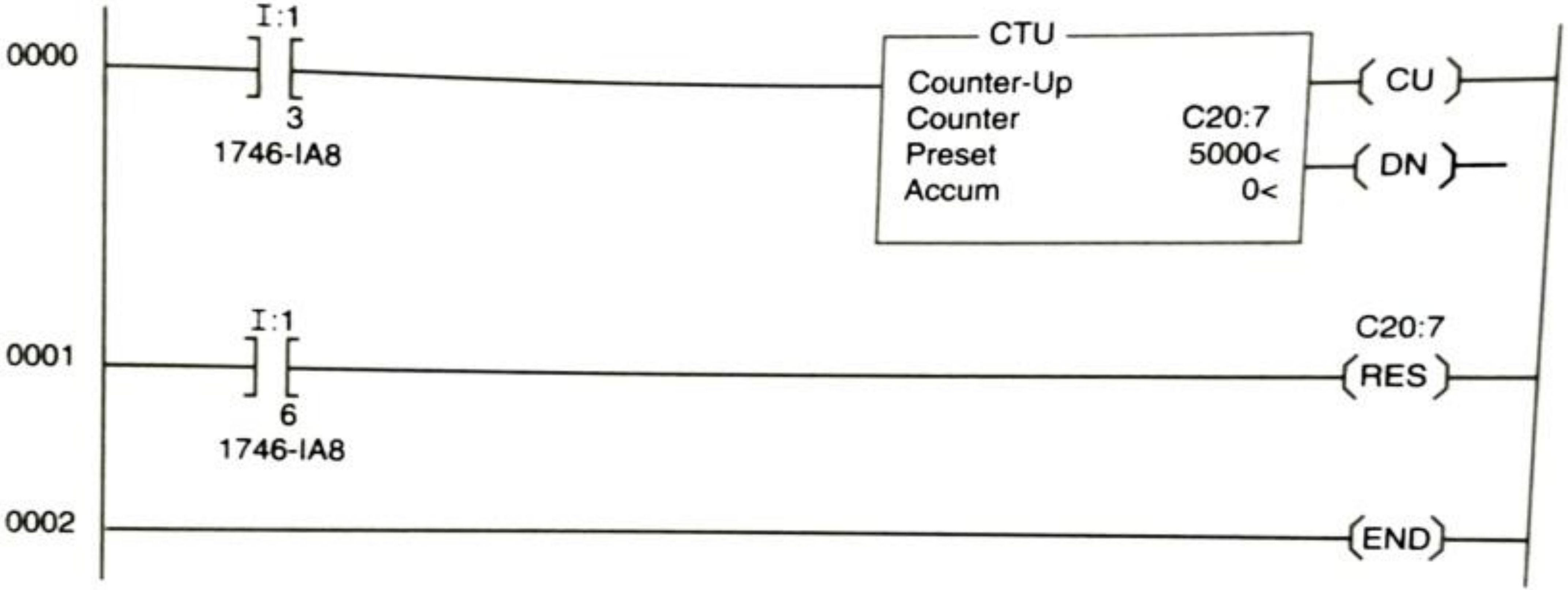


Figure 16-18 Reset instruction programmed for resetting count-up counter C20:7.

THE COUNTER RESET INSTRUCTION (Continued..)

- This instruction is used to reset a timer-on delay, retentive timer, count-up counter, or count-down counter's accumulated value to zero.
- The reset instruction's address must match the address of the timer or counter that is to be reset.
- Only one address is allowed per reset instruction.
- Resetting multiple timers or counters will require multiple reset instructions.
- Figure 16-18 illustrates a reset instruction on rung 0001, which will reset counter C20:7 on rung 0000.
- When input I:1/13 is set, or true, the reset instruction will reset the counter with the same address as the reset instruction.
- When resetting a timer or counter instruction, accumulated values and status bits are zeroed.

THE CLEAR INSTRUCTION

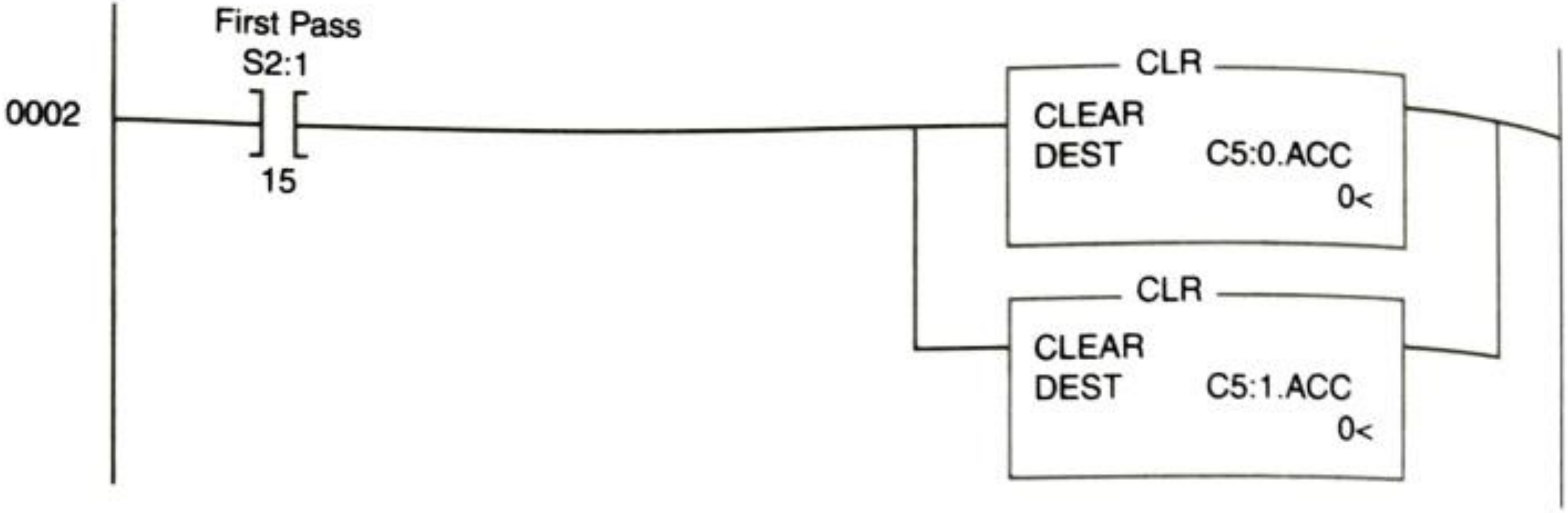


Figure 16-19 Using the clear instruction to set C5:0.ACC and C5:1.ACC to zero.

THE CLEAR INSTRUCTION (Continued..)

- The clear instruction (CLR) is an output instruction that can also be used to set the accumulated value of a timer or counter to zero.
- CLR can be used to set the destination of most data words, in addition to timers or counters, to zero.
- Figure 16-19 illustrates a rung of logic using parallel clear instructions to clear the accumulators of counters 5:0 and C5:1 to zero.

COMBINING TIMERS & COUNTERS

- Timers and counters can be programmed to work together.
- Timers or counters can be connected, or cascaded, together to increase the time or count.
- One counter can be used to count the number of cycles another counter has completed.
- There are many applications and ways in which timers and counters can be programmed to work together.

COMBINING TIMERS & COUNTERS (Continued..)

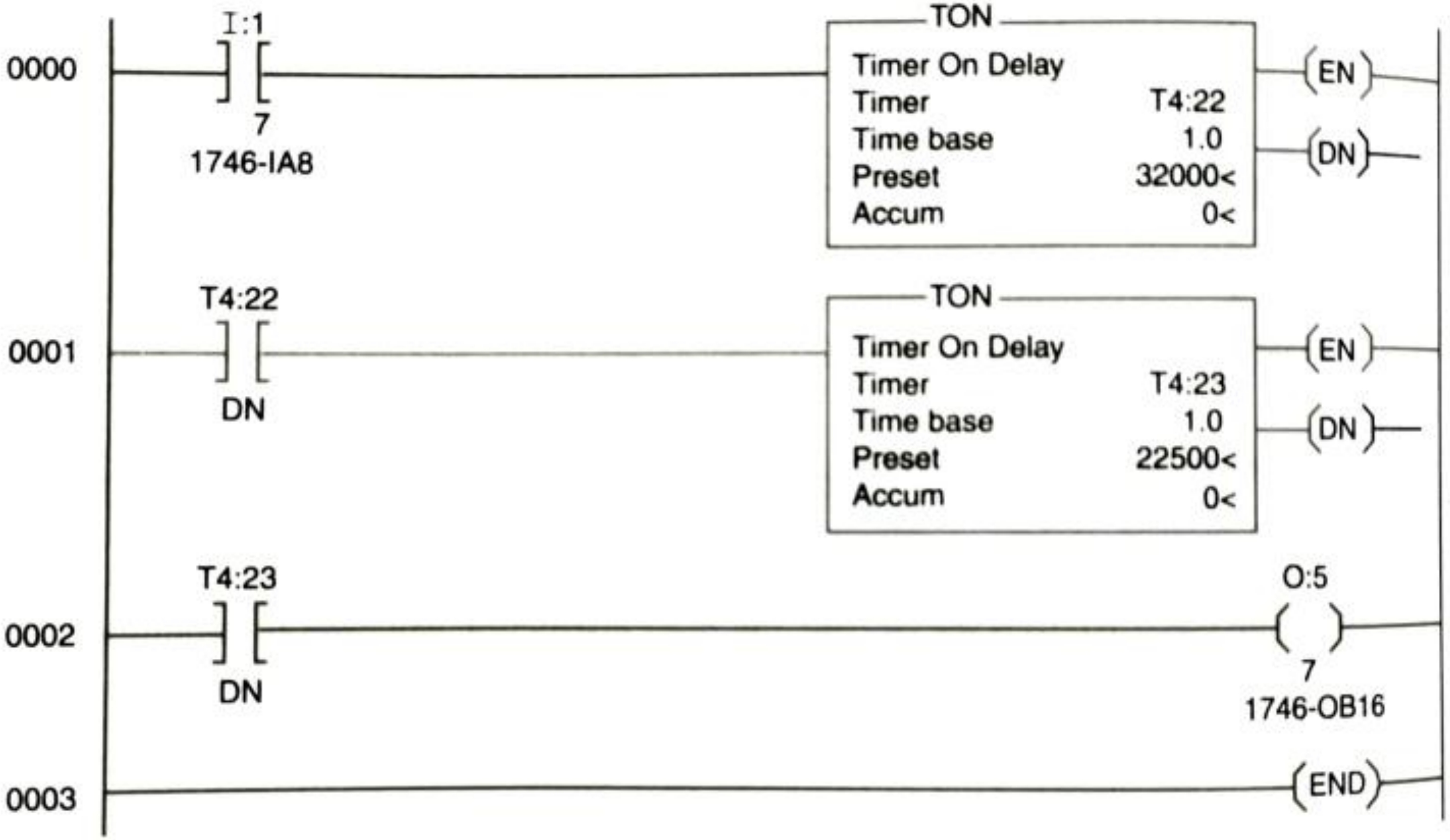


Figure 16-20 Two timers cascaded to lengthen timing.

COMBINING TIMERS & COUNTERS (Continued..)

- Figure 16-20 illustrates two timers cascaded together so as to lengthen the time that can be counted.
- The maximum preset of this timer is 32,767 seconds.
- Timer on rung zero times for 32,000 seconds.
- Timer T4:23, on rung one, continues for an additional 22,500 seconds.
- When the total time has elapsed, T4:23's done bit on rung 0002 will energize output 0:5/7.
- Thus, cascading two timers together is one method of lengthening a timing cycle.

COMBINING TIMERS & COUNTERS (Continued..)

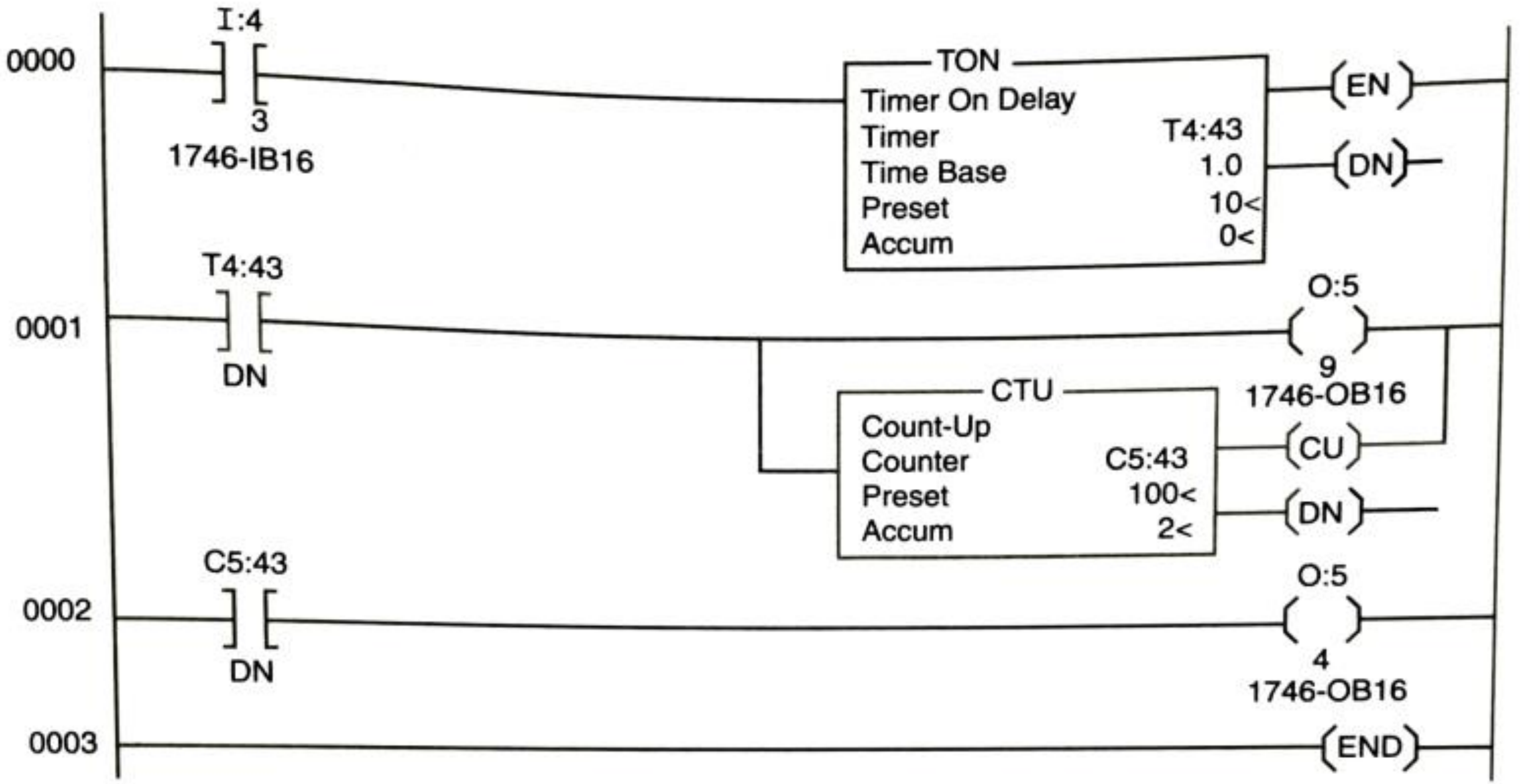


Figure 16-21 Counter C5:0 tracking the number of timing cycles.

COMBINING TIMERS & COUNTERS (Continued..)

- Figure 16-21 illustrates a counter working in conjunction with a timer.
- The timer preset value is 10 seconds.
- Every 10 seconds, the T4:43 DN bit on rung 0001 will pulse at output O:5/9. Counter C5:43, a parallel output, will count the number of timing cycles.

COMBINING TIMERS & COUNTERS (Continued..)

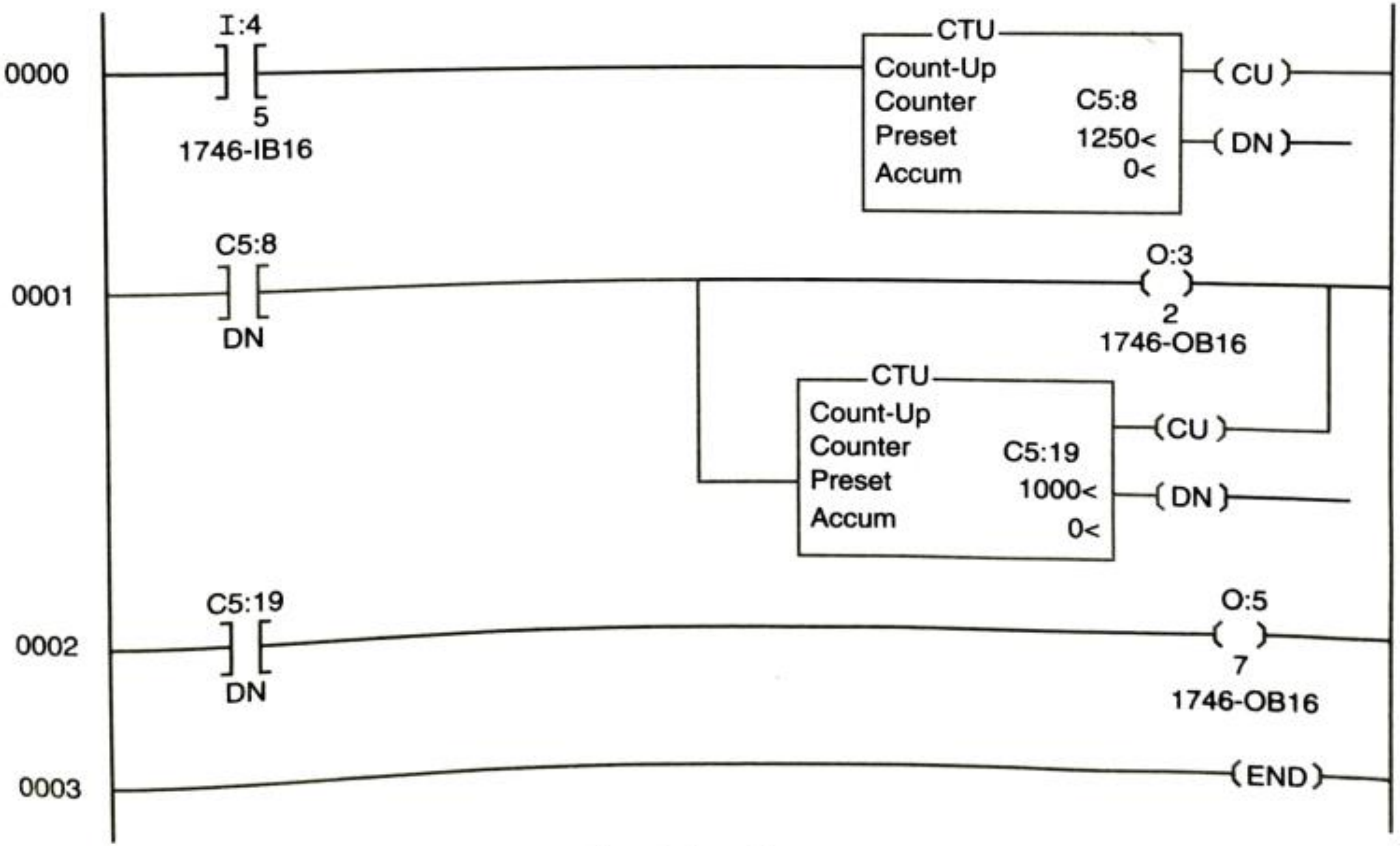


Figure 16-22 Counters cascaded together to lengthen count.

COMBINING TIMERS & COUNTERS (Continued..)

- Figure 16-22 illustrates two counters cascaded together to lengthen the total count.
- Counter C5:8 counts up to 1,250. When C5:8 reaches its preset value 1,250, C5:8's done bit is set. Setting the done bit will energize output O:3/2 and increment counter C5:19.
- After C5:8 has gone through 1,000 cycles, done bit C5:19 will energize output O:5/7.
- In this example, O:5/7 is energized at a count of 1,250,000.

COMBINING TIMERS & COUNTERS (Continued..)

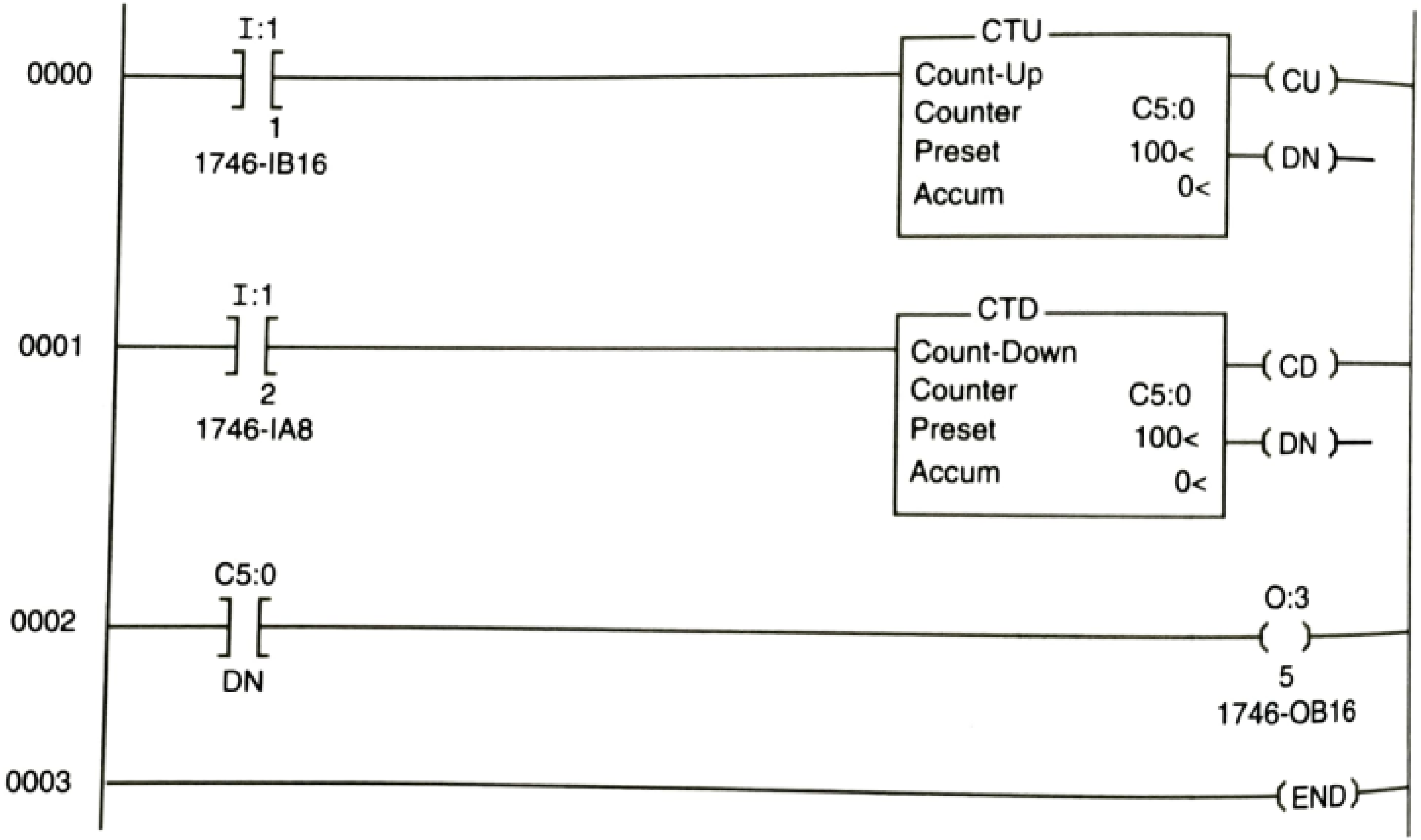


Figure 16-23 Using a CTU and a CTD with the same address to create an up- and down-counter.

COMBINING TIMERS & COUNTERS (Continued..)

- Figure 16-23 illustrates a separate count-up instruction and a separate count-down instruction with the same address, C5:0.
- Input I:1/1 will increment CTU C5:0. Input I:1/2 will decrement CTD C5:0.
- Since each of these counters has the same address, the accumulator will reflect the state of the counter addresses as C5:0.
- Since there is an up-counter and a down-counter addressed as C5:0, the accumulator will be shared between the two counter instructions.
- Because the accumulator is shared, it will reflect any count seen by either counter instruction.

References

1. **Introduction to Programmable Logic Controllers**, Garry Dunning, 3rd edition, Cengage learning.
2. **Working of an electromagnetic relay**

https://youtu.be/cunddFiQzrk?si=958ljaG-YALX-l0c