Electricity and New Energy

Programmable Logic Controller

Basic Principles Using the Programming Software

Courseware Sample

36017-F0

Order no.: 36017-00
Second Edition
Revision level: 02/2015

By the staff of Festo Didactic

Internet: www.festo-didactic.com
e-mail: did@de.festo.com

Printed in Canada

ISBN 978-2-89289-839-2 (Printed version)
Legal Deposit – Bibliothèque et Archives nationales du Québec, 2005
Legal Deposit – Library and Archives Canada, 2005

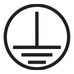
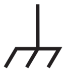
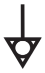
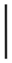
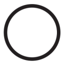
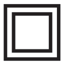
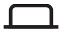
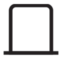
Information in this document is subject to change without notice and does not represent a commitment on the part of Festo Didactic. The Festo materials described in this document are furnished under a license agreement or a nondisclosure agreement.

Safety and Common Symbols

The following safety and common symbols may be used in this manual and on the equipment:

Symbol	Description
⚠ DANGER	**DANGER** indicates a hazard with a high level of risk which, if not avoided, will result in death or serious injury.
⚠ WARNING	**WARNING** indicates a hazard with a medium level of risk which, if not avoided, could result in death or serious injury.
⚠ CAUTION	**CAUTION** indicates a hazard with a low level of risk which, if not avoided, could result in minor or moderate injury.
CAUTION	**CAUTION** used without the *Caution, risk of danger* sign ⚠, indicates a hazard with a potentially hazardous situation which, if not avoided, may result in property damage.
	Caution, risk of electric shock
	Caution, hot surface
	Caution, risk of danger
	Caution, lifting hazard
	Caution, hand entanglement hazard
	Notice, non-ionizing radiation
⎓	Direct current
∿	Alternating current
≂	Both direct and alternating current
3∿	Three-phase alternating current
⏚	Earth (ground) terminal

Safety and Common Symbols

Symbol	Description
	Protective conductor terminal
	Frame or chassis terminal
	Equipotentiality
	On (supply)
	Off (supply)
	Equipment protected throughout by double insulation or reinforced insulation
	In position of a bi-stable push control
	Out position of a bi-stable push control

We invite readers of this manual to send us their tips, feedback, and suggestions for improving the book.

Please send these to did@de.festo.com.

The authors and Festo Didactic look forward to your comments.

Table of Contents

Introduction V

Courseware Outline

Programmable Logic Controller – Basic Principles Using the Programming Software VII

Sample Exercise Extracted from Programmable Logic Controller – Basic Principles Using the Programming Software

Ex. 7 Sequencer Instructions 3

Instructor Guide Sample Extracted from Programmable Logic Controller – Basic Principles Using the Programming Software

Unit 7 Sequencer Instructions 29

Introduction

The Lab-Volt Programmable Logic Controller (PLC) Training System enables students to develop competence in operating, programming, and troubleshooting modern PLC-controlled systems. This system is designed around a complete set of equipment, the 3240 series, intended for the teaching of PLCs.

The PLC Training System is based on the latest Allen-Bradley PLC, and is supported by Lab-Volt, Allen-Bradley, and Rockwell Software instructional materials. PLC programming is achieved by using the Windows®-based RSLogix 500 software from Rockwell Software. At the end of the training program, the students are able to use their freshly-acquired knowledge of PLC programming to achieve PLC control of various industrial applications implemented with the Mechanical Process Simulator, Model 3290, and of virtual industrial processes simulated with the PLC Simulation Software (P-SIM 2000) from A. Palmer Animated Training.

The PLC training program is divided into three courses titled Basic Principles, Industrial Applications, and Programming Software and Applications. Each course relies on a student manual providing theory and hands-on exercises. The manuals guide students from the basic principles of PLCs, where ladder programming is introduced, up to the PLC control of industrial applications where more advanced instructions and programming tools are required.

Courseware Outline

PROGRAMMABLE LOGIC CONTROLLER – BASIC PRINCIPLES USING THE PROGRAMMING SOFTWARE

Exercise 1 Familiarization with the PLC Trainer and RSLogix 500

Introduction to the Lab-Volt PLC Trainer. The RSLogix 500 Software. Running RSLogix and creating projects. Configuring system communications. Editing system preferences.

Exercise 2 Programming Basics

PLC ladder program versus hardwired ladder diagrams. Logical continuity and input/output devices. Series (AND) and parallel (OR) logic. Documenting a ladder program. Creating and printing reports.

Exercise 3 Online Operations

Data file organization. Instruction addressing. PLC input and output interfaces. Downloading a project to a PLC. PLC (processor) modes of operation.

Exercise 4 Latching Instructions

The PLC latching instructions: Output Latch (OTL) and Output Unlatch (OTU). Using latching instructions to maintain PLC outputs activated after the conditions that caused activation of these outputs no longer exist.

Exercise 5 Timer Instructions

The PLC timer instructions: Timer-On Delay (TON), Timer-Off-Delay (TOF), and Retentive Timer-On-Delay (RTO). The status bits of timer instructions. The Reset instruction. Using timer instructions to activate and deactivate PLC outputs during definite periods of time.

Exercise 6 Counter Instructions

The PLC counter instructions: count up (CTU) and count down (CTD). The status bits of counter instructions. The Reset instruction. Using counter instructions and their status bits to activate and deactivate PLC outputs after a specific number of events have occurred.

Courseware Outline

PROGRAMMABLE LOGIC CONTROLLER – BASIC PRINCIPLES USING THE PROGRAMMING SOFTWARE

Exercise 7 Sequencer Instructions

The PLC sequencer instructions: sequencer output (SQO) and sequencer compare (SQC). How to enter a sequencer instruction and its data table. The status bits of sequencer instructions. Using sequencer instructions to obtain sequential activation of the PLC outputs.

Exercise 8 Comparison Instructions

The PLC comparison instructions: Equal (EQU), Not Equal (NEQ), Less Than (LES), Less Than or Equal (LEQ), Greater Than (GRT), and Greater Than or Equal (GEQ). How to enter a comparison instruction. Using sequencer instructions driven by the accumulated value of a timer or counter instruction to perform sequential control of the PLC outputs.

Exercise 9 Shift Register Instructions/The Force Function

The PLC shift register instructions: bit shift left (BSL) and bit shift right (BSR). How to enter a shift register instruction. Using the Force function of the PLC to override the current status of PLC inputs or outputs, regardless of their actual status.

Appendix A Equipment Utilization Chart
B Mnemonics used for SLC 500 Instructions
C Boolean Algebra and Digital Logic
D Glossary of Terms
E Troubleshooting Procedures

Sample Exercise

Extracted from

Programmable Logic Controller

Basic Principles Using

the Programming Software

Exercise 7

Sequencer Instructions

EXERCISE OBJECTIVE

- To program and test PLC ladder programs that use sequencer instructions.

DISCUSSION

Introduction

PLC sequencer instructions are output instructions used to control sequential operations. They are employed in systems where devices must be turned on and off during definite periods of time, and in systems that perform a sequence of successive operations.

Sequencer Instructions of the Trainer PLC

The PLC on your trainer includes the following sequencer instructions:

- the sequencer output (**SQO**) instruction;

- the sequencer compare (**SQC**) instruction.

The SQO instruction transfers data from a programmed sequencer file, through a mask, to a destination file. This instruction is used for the sequential control of various devices in process or control operations.

The SQC instruction compares the data from a source, through a mask, against the data in a programmed sequencer file for equality. This instruction is often used to monitor machine operating conditions or for diagnostic purposes.

To understand how the SQO and SQC instructions work, one must first consider the parameters associated with these instructions. These parameters are described below.

Parameters of the Sequencer Instructions of the Trainer PLC

When entering a sequencer instruction (SQO or SQC) with the trainer PLC, the following parameters must be programmed:

- **File** (SQO, SQC): address of the sequencer file. With an SQO instruction, the sequencer file stores data to be transferred to the destination file. With an

SQC instruction, the sequencer file stores reference data that is used for comparison to the data at the source address. The data contained in the sequencer file can be stored in a binary (Bx) data file or in an integer (Nx) data file.

- **Mask** (SQO, SQC): hexadecimal (h) code or the address of a word or file through which the sequencer instruction transfers (SQO) or compares (SQC) data. Mask bits that are set to logic state 0 will mask data. Mask bits that are set to logic state 1 will pass data.
- **Source** (SQC): address of the input word or file where the SQC instruction takes data for comparison to the data in its sequencer file (reference).
- **Destination** (SQO): address of the output word or file where the SQO instruction transfers data from its sequencer file.
- **Control** (SQO, SQC): 3-word register (R data file) that stores the status bits of the sequencer instruction, the length of its sequencer file, and the position (step) of the sequencer instruction in the sequencer file, as shown in Table 7-1.

Word	B_{15}	B_{14}	B_{13}	B_{12}	B_{11}	B_{10}	B_9	B_8	B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
0	EN		DN		ER			FD								
1	Length of sequencer file															
2	Position															

Table 7-1. R data file structure.

Status bits (word 0 of the control data file)

The status bits provide information on the sequential process. They can be used in a ladder program to control relay-type instructions.

– Enable [EN] – (bit number 15 of word 0): The EN bit pertains to both the SQO and SQC instructions. This bit is set to logic state 1 when the rung containing the sequencer instruction is true, indicating that this instruction is true. It is set to logic state 0 when the rung is false.

– Done [DN] – (bit number 13 of word 0): The Done (DN) bit pertains to both the SQO and SQC instructions. It is set to logic state 1 when the sequencer instruction steps to the last word of data in its sequencer file, causing this word to be transferred (SQO) or compared (SQC). The DN bit is reset to logic state 0 on the next false-to-true transition of the sequencer rung.

– Error [ER] – (bit number 11 of word 0): The ER bit pertains to both the SQO and SQC instructions. The ER bit is set to logic state 1 when the processor detects a negative position value, or a negative or zero length value.

– Found [FD] – (bit number 8 of word 0): The FD bit pertains to the SQC instruction only. This bit is set to logic state 1 when all the non-masked bits in the word or file at the source address match those of the corresponding reference word in the sequencer file. When the bits do not match, the FD bit is set to logic state 0. The FD bit is updated each time the processor evaluates the SQC instruction while the rung is true.

Length of the sequencer file (word 1 of the control data file)

This is the number of steps contained in the sequencer file, starting at position 1. The sequencer instruction automatically returns (wraps) to position 1 upon completion of each cycle.

Upon startup, that is, when the PLC is switched from the Program mode to the Run mode, the sequencer instruction is set to position 0 of its sequencer file. If the rung containing the sequencer instruction is true, this instruction transfers (SQO) or compares (SQC) the data present at position 0 of the sequencer file. If the rung containing the sequencer instruction is false upon startup, this instruction waits until the rung becomes true and then transfers (SQO) or compares (SQC) the data present at position 1 of the sequencer file.

Position (word 2 of the control data file)

This is the position (step) where the sequencer instruction currently is within its sequencer file.

The Reset (RES) instruction can be used to reset a sequencer. To do so, the Reset instruction must be programmed with the address of the R data file (control register) of the sequencer instruction. When made true, the Reset instruction resets the sequencer instruction to position zero of its sequencer file, and it resets all the sequencer status bits (except the FD bit) to logic state 0.

Operation of the SQO Instruction

Figure 7-1 shows how the SQO instruction works. When the rung containing the SQO instruction goes from false to true, this instruction steps to the next position in its sequencer file (file B10). The bits in the sequencer file are programmed to control PLC outputs 0 through 3. They are transferred, through the mask value (000FH), to the destination address (O:0.0).

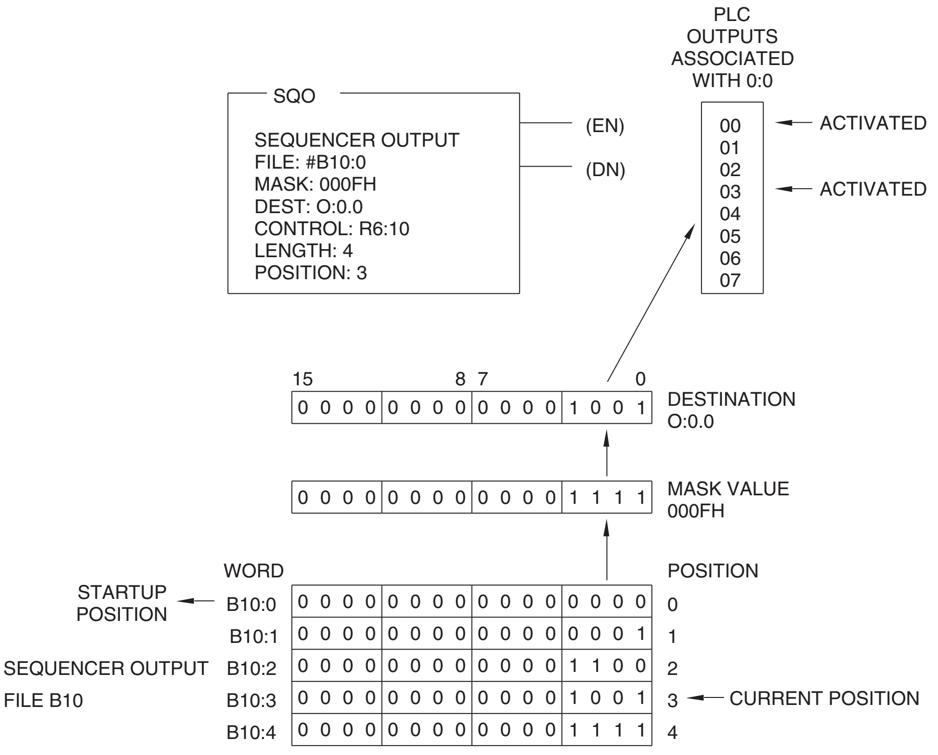
In Figure 7-1, for example, the SQO instruction is at position 3 of its sequencer file. Consequently, the word of data stored at address B10:3 is transferred to destination address O:0.0. Since bits 0 through 3 of the mask value are set to logic state 1, bits 0 through 3 of the word of data (1001) are allowed to pass to destination address O:0.0. This causes the output data file bits at addresses O:0/0 and O:0/3 to be set to logic state 1, causing PLC outputs 0 and 3 to be activated.

When the rung containing the SQO instruction becomes false, this instruction remains at position 3 of its sequencer file, so that PLC outputs 0 and 3 remains activated. When the rung becomes true again, the SQO instruction steps to position 4 of its sequencer file, which is the last position of this file. This causes the word of data stored at address B10:4 to be transferred to destination address O:0.0 through the mask value. This in turn causes PLC outputs 0 through 3 to become all activated.

On the next false-to-true transition of the rung containing the SQO instruction, this instruction automatically returns to position 1 of its sequencer file to start a new cycle. This causes the word of data stored at address B10:1 to be transferred to destination address O:0.0 through the mask value. This in turn causes PLC output 0 to be activated, and PLC outputs 1, 2, and 3 to be deactivated.

If the Reset instruction associated with the SQO instruction is made true at some point of the cycle, the SQO instruction will automatically return to position 0 of its sequencer file, and the following will occur:

- If the rung of the SQO instruction is true at the moment when the Reset instruction is made true, the word of data stored at position 1 of sequencer file B10 will be transferred to destination address O:0.0. This same word will be transferred on the next false-to-true rung transition, as the SQO instruction will step to position 1 of its sequencer file.
- If the rung of the SQO instruction is false at the moment when the Reset instruction is made true, the PLC output status will remain unchanged. When this rung becomes true again, the SQO instruction will step to position 1 of sequencer file B10, causing the word of data stored at this position to be transferred to destination address O:0.0.



SQO
SEQUENCER OUTPUT
FILE: #B10:0
MASK: 000FH
DEST: O:0.0
CONTROL: R6:10
LENGTH: 4
POSITION: 3

15	8 7		0	
0 0 0 0	0 0 0 0	0 0 0 0	1 0 0 1	DESTINATION O:0.0
0 0 0 0	0 0 0 0	0 0 0 0	1 1 1 1	MASK VALUE 000FH

WORD					POSITION
B10:0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0
B10:1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 1	1
B10:2	0 0 0 0	0 0 0 0	0 0 0 0	1 1 0 0	2
B10:3	0 0 0 0	0 0 0 0	0 0 0 0	1 0 0 1	3
B10:4	0 0 0 0	0 0 0 0	0 0 0 0	1 1 1 1	4

Figure 7-1. Operation of the SQO instruction.

Operation of the SQC Instruction

Figure 7-2 shows how the SQC instruction works. When the rung containing the SQC instruction goes from false to true, this instruction steps to the next position in its sequencer file (file B12). The bits in the sequencer file are programmed to monitor PLC inputs 0 through 3. They are compared, through the mask value (000FH), against the bits at the source address (I:0.0) for equality. When all the non-masked bits at the source address match the bits in the reference word of the sequencer file, the Found (FD) bit of the sequencer instruction is set to logic state 1. When the bits do not match, the FD bit is set to logic state 0.

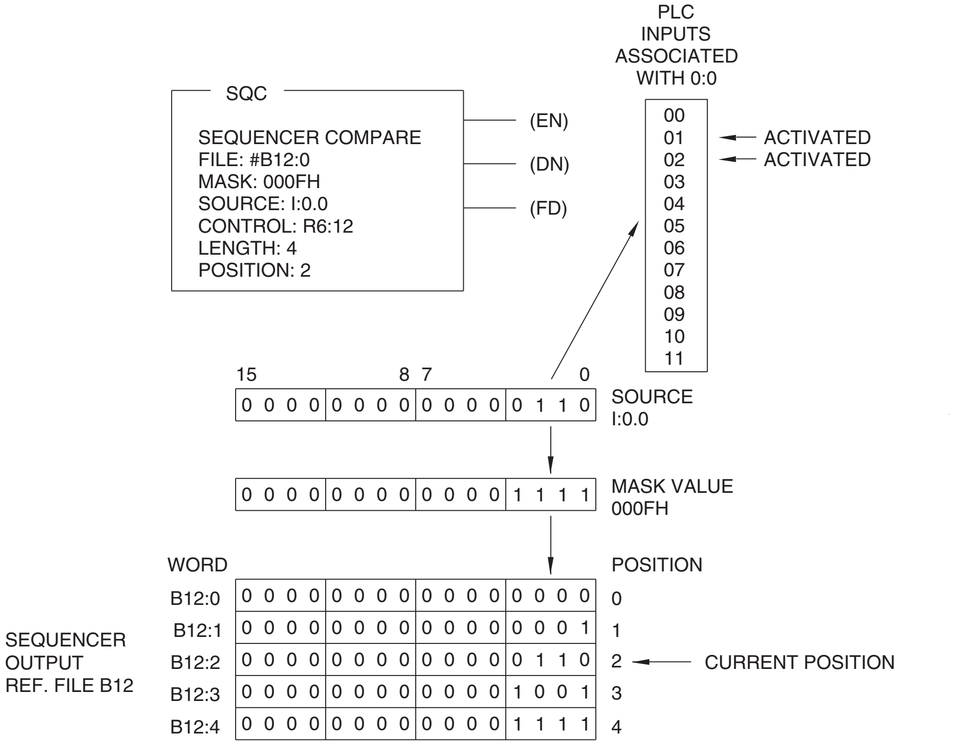


Figure 7-2. Operation of the SQC instruction.

In Figure 7-2, for example, the SQC instruction is at position 2 of its sequencer file. Consequently, the word of data stored at address B12:2 is compared, through the mask value, against the bits at source address I:0.0. Since PLC inputs 1 and 2 are activated, the bits at addresses I:0/1 and I:0/2 are set to logic state 1. Since bits 0 through 3 of the mask value are set to logic state 1, bits 0 through 3 at source address I:0.0 are non-masked. Consequently, all the bits at source address I:0.0 match those of the reference word at address B12:2, causing the FD bit to be set to logic state 1.

During all the time that the rung of the SQC sequencer remains true, the comparison is performed for every scan of the processor. This implies that, if PLC input 1 or 2 becomes deactivated while the rung is still true, the FD bit will be reset to logic state 0.

When the rung containing the SQC instruction becomes false, the FD bit is automatically reset to logic state 0. When the rung becomes true again, the SQC instruction steps to position 3 of its sequencer file and compares the word of data stored at address B12:3, through the mask value, against the bits at source address I:0.0. If PLC inputs 0 and 3 are activated, the FD will be set to logic state 1, otherwise it will be set to logic state 0.

On the next false-to-true transition of the rung containing the SQC instruction, the SQC instruction steps to position 4 and compares the word of data stored at address B12:4, through the mask value, against the bits at source address I:0.0. If PLC inputs 0 through 3 are all activated, the FD will be set to logic state 1, otherwise it will be set to logic state 0.

On the next false-to-true transition of the rung containing the SQC instruction, this instruction automatically returns to position 1 of its sequencer file to start a new cycle.

Procedure Summary

In this exercise, you will program and test two ladder programs: one that uses a sequencer output (SQO) instruction, and one that uses a sequencer compare (SQC) instruction. This will allow you to understand how each of these instructions and their status bits work. You will then use the acquired knowledge to create and test your own ladder program.

> **Note:** *The procedure given below applies to PLC Trainer model 3240. If you are using PLC Trainer model 3270, follow this procedure by adapting it in the usual way:*
>
> - *When asked to activate a PLC input, connect the corresponding terminal of the PLC input interface to the positive terminal of an external 24-V DC source, using a connection lead. (The common of the PLC input interface must be connected to the common of the 24-V DC source.)*
> - *When asked to deactivate a PLC input, remove the lead that connects the corresponding terminal of the PLC input interface to the 24-V DC source.*
> - *Observe the PLC output status indicators on the PLC module (as there are no PLC output lamps on this trainer front panel) to determine whether or not a PLC output is activated.*

EQUIPMENT LIST

Refer to the Equipment Utilization Chart, in Appendix A of this manual, to obtain the list of equipment required to perform this exercise.

PROCEDURE

Setting Up the Equipment

☐ 1. Connect the RS-232 serial port of the computer station to the Lab-Volt PC Interface, Model 3246. Connect the PC Interface to the communications port of the PLC on the PLC Trainer, using a 1747-C10 cable.

☐ 2. Turn on the computer and start RSLogix 500.

Turn on the PLC Trainer.

☐ 3. Connect the 120 (240)-V AC source of the PLC Trainer to the supply terminals for terminals 0 through 3, and to those for terminals 4 through 7 of the PLC output interface. This will allow the PLC output lamps on the trainer front panel to illuminate when the corresponding PLC outputs are activated.

Note: *Skip this step if you are using PLC Trainer model 3270.*

The SQO Instruction

☐ 4. Create a new project having the following processor name: EXERC_7.

The project tree of processor EXERC_7 and program file LAD 2 should be displayed in the RSLogix 500 window. Program file LAD 2 contains the main ladder program.

The next steps of this procedure consist in entering the ladder program of Figure 7-3.

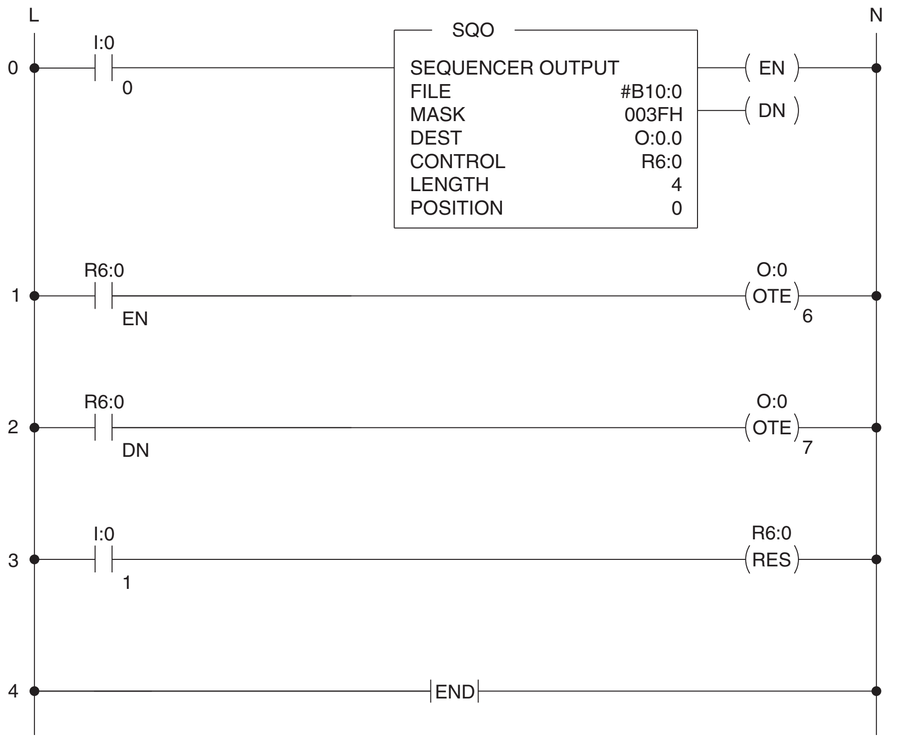


Figure 7-3. The sequencer output (SQO) instruction.

☐ 5. Select the **User** category of instructions by clicking the corresponding instruction category selection tab.

Insert a new rung into program file LAD 2. In this rung, enter instruction XIC I:0/0. Then, enter instruction SQO R6:0 by performing the following steps:

– Select the **File Shift/Sequencer** category of instructions by clicking the corresponding instruction category selection tab.
– Click the **Sequencer Output** button on the instruction toolbar to insert this instruction in the rung, type: **B10:0**, then press the mouse left button to accept this address. (The file indicator “#” will automatically be added to the front of this address).
– Double-click **Mask** within the SQO instruction, type: **003FH**, then press the mouse left button to accept value.
– Double-click **Dest** within the SQO instruction, type: **O:0.0**, then press the mouse left button to accept this address.
– Double-click **Control** within the SQO instruction, type: **R6:0**, then press the mouse left button to accept this address.
– Double-click **Length** within the SQO instruction, type: **4**, then press the mouse left button to accept this value.
– Leave the **Position** value of the SQO instruction set to **0**. This completes the parameter setting for instruction SQO R6:0.

You have now finished entering the instructions of rung 0 in the ladder program of Figure 7-3.

> **Note:** *The letter "H" that you entered after the mask value 003F specifies that this value is a hexadecimal code. This hexadecimal code corresponds to the following binary code: 0000 0000 0011 1111.*

☐ 6. Select the **User** category of instructions by clicking the corresponding instruction category selection tab.

Enter a new rung into program file LAD 2. In this rung, enter instruction XIC R6:0/EN. To do so, click the **Examine If Closed** button on the instruction toolbar, type: **R6:0/EN**, then click the mouse left button to enter this address.

Enter instruction OTE O:0/6.

You have now finished entering the instructions of rung 1 in the ladder program of Figure 7-3.

☐ 7. Enter a new rung into program file LAD 2. In this rung, enter instruction XIC R6:0/DN, then enter instruction OTE O:0/7.

You have now finished entering the instructions of rung 2 in the ladder program of Figure 7-3.

☐ 8. Enter a new rung into program file LAD 2. In this rung, enter instruction XIC I:0/1. Then, enter instruction RES R6:0. To do so, select the Timer/Counter category of instructions by clicking the corresponding instruction category selection tab. Click the **Reset** button on the instruction toolbar to insert this instruction in the rung, type: **R6:0**, then press the mouse left button to accept this address.

You have now finished entering the instructions of rung 3 in the ladder program of Figure 7-3.

☐ 9. Using the **Verify Rung** command, verify the rungs that have been edited. Correct the errors, if any.

The main ladder program in RSLogix 500 should be identical to that shown in Figure 7-3.

☐ 10. In the **Data Files** folder of processor EXERC_7, observe that a new file named B10 appears below File r8 - RESERVED. This file has been created because you entered B10:0 as the **File** address for instruction SQO R6:0. Data file **B10** is the sequencer file for this instruction.

Open data file **B10**. Observe that the corresponding window displays the logic state of the bits in the words at addresses B10:0 through B10:4. Each word corresponds to a sequencer position.

Enter 1's at the proper bit locations of data file **B10** so that it contains the same data as Table 7-2 below. When you have finished, close data file **B10**.

Note: *To enter a 1 at a bit location, double-click the bit location, type: 1 and then press the* ***Enter*** *key.*

SEQUENCER POSITION	WORD	BINARY DATA			
		15	8	7	0
0	B10:0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	B10:1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 1 1
2	B10:2	0 0 0 0	0 0 0 0	0 0 0 0	0 1 1 1
3	B10:3	0 0 0 0	0 0 0 0	0 0 1 0	1 0 1 0
4	B10:4	0 0 0 0	0 0 0 0	0 0 1 1	1 1 1 1

Table 7-2. Data table for instruction SQO R6:0.

☐ 11. Save the project in a project file named EXERC_7.RSS.

☐ 12. Make sure the system communications are properly configured.

☐ 13. Download project EXERC_7 to the PLC.

Go online and place the PLC in the Run mode.

☐ 14. Open data file **B10** to display the logic state of the bits in the words at addresses B10:0 through B10:4.

Place the **B10** data file window at the bottom of the RSLogix 500 window.

☐ 15. Open data file **R6 - CONTROL**. Observe that the corresponding window displays the status bits of instruction SQO R6:0, the length of sequencer file B10, and the position where instruction SQO R6:0 is within sequencer file B10.

What is the logic state of the EN status bit of instruction SQO R6:0? Why?

____

____

What is the current position of instruction SQO R6:0 within sequencer file B10? Why?

____

____

Place the **R6 - CONTROL** data file window at the bottom of the RSLogix 500 window.

□ 16. Activate PLC input 0. What happens to the EN status bit of instruction SQO R6:0 and to PLC output 6? Why?

____

____

____

____

What is the current position of instruction SQO R6:0 within sequencer file B10? Why?

____

____

□ 17. Which bits in the word at address B10:1 are set to logic state 1?

____

□ 18. According to the current status of PLC outputs 0 and 1, is the word of data at address B10:1 transferred to destination address O:0.0, through the mask value of instruction SQO R6:0? Explain.

□ 19. Deactivate PLC input 0. What happens to the EN status bit of instruction SQO R6:0 and to PLC output 6? Why?

Is instruction SQO R6:0 still at position 1 of sequencer file B10, causing PLC outputs 0 and 1 to be activated?

□ Yes □ No

□ 20. Activate PLC input 0. What is the current position of instruction SQO R6:0 within sequencer file B10? Why?

□ 21. Which bits in the word at address B10:2 are set to logic state 1?

□ 22. According to the current status of PLC outputs 0, 1, and 2, is the word of data at address B10:2 transferred to destination address O:0.0, through the mask? Explain.

____

____

____

____

____

□ 23. Deactivate and then activate PLC input 0. Are PLC outputs 1, 3, and 5 activated? Why?

____

____

____

____

____

□ 24. Deactivate and then activate PLC input 0. What happens to the DN status bit of instruction SQO R6:0 and to PLC output 7? Why?

____

____

____

Are PLC outputs 0 through 5 activated? Why?

____

____

____

□ 25. Deactivate and then activate PLC input 0. What happens to the DN status bit of instruction SQO R6:0 and to PLC output 7? Why?

____

____

____

____

Are PLC outputs 0 and 1 activated, indicating that instruction SQO R6:0 has returned to position 1 of sequencer file B10 to initiate a new cycle?

☐ Yes ☐ No

☐ 26. Deactivate and then activate PLC input 0. Instruction SQO R6:0 is now at position 2 of sequencer file B10, causing PLC inputs 0, 1, and 2 to be activated.

Deactivate and then activate PLC input 0. Instruction SQO R6:0 is now at position 3 of sequencer file B10, causing PLC inputs 1, 3, and 5 to be activated.

Activate PLC input 1. Are PLC outputs 0 and 1 activated? Why?

____

____

____

____

____

☐ 27. Deactivate PLC input 1. Does this cause instruction SQO R6:0 to step to position 1 of sequencer file B10, causing PLC outputs 0 and 1 to remain activated?

☐ Yes ☐ No

☐ 28. If the mask value of instruction SQO R6:0 were changed to "00F0H", what would be the status of PLC outputs 0 through 5 when this instruction is at positions 1, 2, 3, and 4 of sequencer file B10? Explain.

____

____

____

____

☐ 29. On the PLC Trainer, make sure that all PLC inputs are deactivated (all the pushbuttons released).

☐ 30. Place the PLC in the **Program** mode and go offline.

The SQC Instruction

□ 31. Modify the existing main ladder program of project EXERC_7 as indicated below in order to obtain the ladder program shown in Figure 7-4:

a. In rung 0, double-click the address of instruction XIC I:0/0, type the following address: **I:0/8**, then press the mouse left button to accept the new address.

b. In rung 0, replace instruction SQO R6:0 with a SQC (sequencer compare) instruction having the same address. To do so, select instruction SQO R6:0 and choose the **Change Instruction Type** command in the context-sensitive menu. Type **SQC** using the keyboard and press the **Enter** key.

Modify the parameter setting of the SQC instruction as follows:

– Double-click **File**, type **B12:0**, then press the mouse left button to accept this address.
– Double-click **Mask**, type: **00FFH**, then press the mouse left button to accept this value.
– Double-click **Source**, type: **I:0.0**, then press the mouse left button to accept this value.
– Leave the **Control** parameter set to **R6:0**, the **Length** parameter set to **4**, and the **Positio**n parameter set to **0**.

c. In rung 1, double-click the address of instruction OTE O:0/6, type the following address: **O:0/0**, then press the mouse left button to accept the new address.

d. In rung 2, double-click the address of instruction OTE O:0/7, type the following address: **O:0/1**, then press the mouse left button to accept the new address.

e. Insert a new rung below rung 2. In the newly created rung (rung 3), enter instruction XIC R6:0/FD. Then, enter instruction OTE O:0/2.

f. In rung 4, double-click the address of instruction XIC I:0/1, type the following address: **I:0/9**, then press the mouse left button to accept the new address.

Using the **Verify Rung** command, verify the rungs that have been edited. Correct any errors.

The main ladder program in RSLogix 500 should be identical to that shown in Figure 7-4.

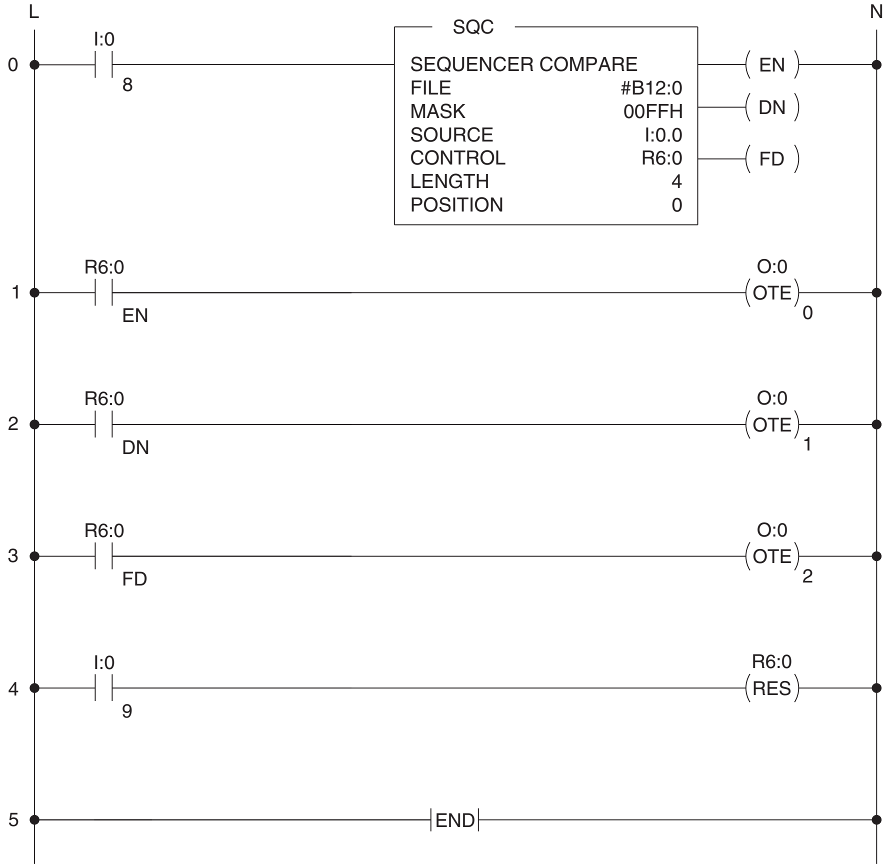


Figure 7-4. The SQC instruction.

□ 32. In the **Data Files** folder of processor EXERC_7, observe that a new file named **B12** appears below File B10. File B12 has been created because you entered B12:0 as the **File** address for instruction SQC R6:0. File B12 is the sequencer file for this instruction.

Open data file **B12**. Observe that the corresponding window displays the logic state of the bits in the words at addresses B12:0 through B12:4. Each word corresponds to a sequencer position.

Enter 1's at the proper bit locations of data file **B12** so that it contains the same data as Table 7-3 below. When you have finished, close data file **B12**.

SEQUENCER POSITION	WORD	BINARY DATA			
		15	8	7	0
0	B12:0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	B12:1	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1
2	B12:2	0 0 0 0	0 0 0 0	0 1 0 1	0 1 0 1
3	B12:3	0 0 0 0	0 0 0 0	1 1 1 1	1 1 1 1
4	B12:4	0 0 0 0	0 0 0 0	1 0 1 0	1 0 1 0

Table 7-3. Data table for instruction SQC R6:0.

□ 33. Using the **Save As** command in the **File** menu, save the new main ladder program in a project file named EXERC_7a.RSS.

> **Note:** *Do not forget to change the processor name while you are in the Save Program As dialog box.*

□ 34. Download project EXERC_7a to the PLC of the PLC Trainer. Go online and place the PLC in the **Run** mode.

□ 35. Open data file **B12** to display the logic state of the bits in the words at addresses B12:0 through B12:4.

Place the **B12** data file window at the bottom of the RSLogix 500 window.

□ 36. Open data file **R6 - CONTROL**. Observe that the corresponding window displays the status bits of instruction SQC R6:0, the length of sequencer file B12, and the position where instruction SQC R6:0 is within sequencer file B12.

What is the logic state of the EN status bit of instruction SQC R6:0 ? Why?

____

____

What is the current position of instruction SQC R6:0 within sequencer file B12? Why?

____

____

□ 37. Activate PLC input 8. What happens to the EN status bit of instruction SQC R6:0 and to PLC output 0? Why?

What is the current position of instruction SQC R6:0 within sequencer file B12? Why?

□ 38. Which bits in the word at address B12:1 are set to logic state 1?

□ 39. Activate PLC inputs 0 and 2. What happens to the FD status bit of instruction SQC R6:0 and to PLC output 2? Why?

□ 40. Deactivate PLC input 8. What happens to the EN and FD status bits of instruction SQC R6:0, and to PLC outputs 0 and 2? Why?

□ 41. Activate PLC input 8. Does this cause instruction SQC R6:0 to step to position 2 in sequencer file B12? Why?

____

____

□ 42. Leave PLC inputs 0 and 2 activated. Activate PLC inputs 4 and 6. Does the FD status bit of instruction SQC R6:0 go to logic state 1? Why?

____

____

____

____

□ 43. Deactivate PLC input 6. What happens to the EN and FD status bits of instruction SQC R6:0? Why?

____

____

____

□ 44. Deactivate and then activate PLC input 8. Instruction SQC R6:0 is now at position 3 of sequencer file B12. Which PLC inputs must now be activated in order for the FD status bit to be set to logic state 1? Explain why and verify your answer by activating the proper PLC inputs.

____

____

____

□ 45. Deactivate and then activate PLC input 8. What happens to the DN status bit of instruction SQC R6:0 and to PLC output 1? Why?

____

____

____

____

Which PLC inputs must now be activated in order for the FD status bit to be set to logic state 1? Explain why and verify your answer by activating the proper PLC inputs.

____

____

____

□ 46. Deactivate and then activate PLC input 8. What happens to the sequencer position? To the DN status bit of instruction SQC R6:0? Explain.

____

____

____

□ 47. Deactivate and activate PLC input 8 until instruction SQC R6:0 is at position 3 of sequencer file B12.

Activate PLC inputs 0 through 7 to make the FD status bit of instruction SQC R6:0 go to logic state 1.

Activate PLC input 9. What happens to the sequencer position? To the FD status bit? Why?

____

____

____

□ 48. Deactivate PLC input 9. Does this cause instruction SQC R6:0 to step to position 1 of sequencer file B12?

□ Yes □ No

□ 49. If the mask value of instruction SQC R6:0 were changed to "00F0H", what PLC inputs would you need to activate in order for the FD status bit of instruction SQC R6:0 to be set to logic state 1 when this instruction is at positions 1, 2, 3, and 4 of sequencer file B12? Explain.

____

____

____

____

□ 50. On the PLC Trainer, make sure that all PLC inputs are deactivated (all the pushbuttons released).

□ 51. Place the PLC in the **Program** mode. Go offline and close project EXERC_7a.RSS.

Creating a New Ladder Program

□ 52. Create (on paper) a ladder diagram that will control the activation of PLC outputs 0 through 7 in the following way:

- Upon startup, all the PLC outputs are deactivated.
- When PLC input 0 is activated, PLC output 0 is activated (step 1);
- 5 seconds later, PLC outputs 0 through 3 are activated (step 2);
- 5 seconds later, PLC outputs 0 through 3 are deactivated, while PLC outputs 4 through 7 are activated (step 3);
- 5 seconds later, PLC outputs 0 through 7 are activated (step 4);
- 5 seconds later, the cycle automatically repeats, starting from step 1, causing PLC output 0 to be activated.
- If PLC input 0 is deactivated, the PLC outputs that are activated at that moment remain activated. When PLC input 0 is reactivated, the cycle resumes from the step following the one at which it was interrupted.

Hints: use a sequencer output (SQO) instruction and a timer-on-delay (TON) instruction in your program. Use timer status bits to create transitions on the rung containing the SQO instruction and to reset the TON instruction.

□ 53. Create a new project having the following processor name: EXERC_7b. The project tree of processor EXERC_7b and program file LAD 2 should be displayed in the RSLogix 500 window.

Enter your ladder program in program file LAD 2.

Verify each rung, then save the project in a project file named EXERC_7b.RSS.

Download project EXERC_7b to the PLC of the PLC Trainer.

Go online and place the PLC in the **Run** mode.

□ 54. Test program operation and, if required, modify your program so that it operates properly. Once the program has been found operational, have the instructor check your work.

☐ 55. When you have finished, place the PLC in the **Program** mode and clear the PLC memory.

☐ 56. Close RSLogix 500. Turn off the computer.

☐ 57. On the PLC Trainer, make sure that all PLC inputs are deactivated (all the pushbuttons are released). Turn off the PLC Trainer. Remove all patch cords and return all the equipment.

CONCLUSION

In this exercise, you familiarized yourself with the following sequencer instructions of the trainer PLC: the sequencer output (SQO) instruction and the sequencer compare (SQC) instruction. You learned that these instructions have an R6 (control) file, as well as a sequencer file that stores data to be transferred (SQO) or compared (SQC)

- The SQO instruction transfers data from its sequencer file, through a mask, to a destination file on each false-to-true rung transition. Once the SQO instruction has reached the last position in its sequencer file, it automatically returns to position 1 on the next false-to-true rung transition.

- The SQC instruction compares the data from a source file, through a mask, against the data in its sequencer file for equality. If there is equality, the Found (FD) status bit is set to logic state 1. When a false-to-true rung transition occurs, the SQC instruction steps to the next position in its sequencer file. Once the SQO instruction has reached the last position in its sequencer file, it automatically returns to position 1 on the next false-to-true rung transition.

REVIEW QUESTIONS

1. What is the sequencer file of an SQO or SQC instruction?

____

____

____

2. What is the mask value of an SQO or SQC instruction?

____

____

____

3. Briefly describe how an SQO instruction works.

4. When is the Found (FD) bit associated with the SQC instruction set to logic state 1?

5. Once an SQO instruction has reached the last position in its sequencer file, what happens to the sequencer position and to the Done (DN) bit on the next false-to-true transition of this instruction's rung?

Instructor Guide Sample

Extracted from

Programmable Logic Controller

Basic Principles Using

the Programming Software

EXERCISE 7 SEQUENCER INSTRUCTIONS

ANSWERS TO PROCEDURE QUESTIONS

□ 15. The EN status bit is at logic state 0 because the rung containing instruction SQO R6:0 (rung 0) is false.

Instruction SQO R6:0 is at position 0 of sequencer file B10. This occurs because immediately after startup, that is, when the PLC is switched from the Program mode to the Run mode, the SQO instruction is set to position 0 of its sequencer file.

□ 16. When PLC input 0 is activated, the EN status bit of instruction SQO R6:0 is set to logic state 1, since the rung containing this instruction (rung 0) becomes true. This causes instruction XIC R6:0/EN in rung 1 to become true, making instruction OTE O:0/6 in this rung true. This causes PLC output 6 to become activated.

Instruction SQO R6:0 is at position 1 of sequencer file B10. This occurs because the rung containing this instruction (rung 0) has gone from false to true, causing this instruction to step to the next position in sequencer file B10.

□ 17. Bits B10:1/0 and B10:1/1 are set to logic state 1.

□ 18. Yes, since PLC outputs 0 and 1 are activated. This occurs because the mask value of instruction SQO R6:0 is set to 003FH (0000 0000 0011 1111 in binary), thereby allowing bits 0 through 5 of the word at address B10:1 to pass to destination address O:0.0. Since bits 0 and 1 in this word are set to logic state 1, the bits at addresses O:0/0 and O:0/1 are set to logic state 1, causing PLC outputs 0 and 1 to be activated.

□ 19. When PLC input 0 is deactivated, the EN status bit of instruction SQO R6:0 is set to logic state 0, since the rung containing this instruction (rung 0) becomes false. This causes instruction XIC R6:0/EN in rung 1 to become false, making instruction OTE O:0/6 in this rung false. This causes PLC output 6 to become deactivated.

Yes.

□ 20. Instruction SQO R6:0 is at position 2 of sequencer file B10. This occurs because the rung containing this instruction (rung 0) has gone from false to true, causing this instruction to step to the next position in its sequencer file.

□ 21. Bits B10:2/0, B10:2/1, and B10:2/2 are set to logic state 1.

□ 22. Yes, since PLC outputs 0, 1, and 2 are activated. The reason why this occurs is that bits B10:2/0, B10:2/1, and B10:2/2 in the word at address B10:2 are set to logic state 1. Since these bits are allowed to pass to destination address O:0.0 through the mask, the bits at addresses O:0/0, O:0/1, O:0/2 are set to logic state 1. This causes PLC outputs 0, 1, and 2 to be activated.

□ 23. Yes. This occurs because the rung containing instruction SQO R6:0 (rung 0) has gone from false to true, causing this instruction to step to position 3 of sequencer file B10. Since bits B10:3/1, B10:3/3, and B10:3/5 in the word at address B10:3 are set to logic state 1, and since these bits are allowed to pass to destination address O:0.0 through the mask, the bits at addresses O:0/1, O:0/3, and O:0/5 are set to logic state 1. This causes PLC outputs 1, 3, and 5 to be activated.

□ 24. When PLC input 0 is deactivated and then activated, the DN status bit of instruction SQO R6:0 is set to logic state 1 since this instruction steps to the last position in sequencer file B10 (position 4).This causes instruction XIC R6:0/DN in rung 2 to become true, making instruction OTE O:0/7 in this rung true. This causes PLC output 7 to become activated.

Yes. The reason why this occurs is that bits B10:4/0 through B10:4/5 in the word at address B10:4 are all set to logic state 1. Since these bits are allowed to pass to destination address O:0.0 through the mask, the bits at addresses O:0/0 through O:0/5 are all set to logic state 1. This causes PLC outputs 0 through 5 to be activated.

□ 25. The DN status bit is reset to logic state 0, because the rung containing instruction SQO R6:0 (rung 0) goes from false to true. This causes instruction XIC R6:0/DN in rung 2 to become false, making instruction OTE O:0/7 in this rung false. This causes PLC output 7 to become deactivated.

Yes.

□ 26. Yes. This occurs because activation of PLC input 1 causes instruction XIC I:0/1 in rung 3 to become true, making instruction RES R6:0 in this rung true. This causes instruction SQO R6:0 to return to position 0 of sequencer file B10. Since the rung of instruction SQO R6:0 is true, the word of data stored at position 1 of sequencer file B10 is transferred to destination address O:0.0 through the mask. As a result, PLC outputs 0 and 1 are activated.

□ 27. Yes.

□ 28. If the mask value of instruction SQO R6:0 were changed to "00F0H" (0000 0000 1111 0000 in binary), bits 0 through 3 of the words in sequencer file B10 would be masked. Consequently, the status of PLC outputs 0 through 5 for each sequencer position would be as follows:

Position 1: PLC outputs 0 through 5 deactivated;
Position 2: PLC outputs 0 through 5 deactivated;
Position 3: PLC outputs 0 through 4 deactivated, PLC output 5 activated;
Position 4: PLC outputs 0 through 3 deactivated, PLC outputs 4 and 5 activated.

□ 36. The EN status bit is at logic state 0 because the rung containing instruction SQC R6:0 (rung 0) is false.

Instruction SQC R6:0 is at position 0 of sequencer file B12. This occurs because immediately after startup, that is, when the PLC is switched from the **Program** mode to the **Run** mode, the SQC instruction is set to position 0 of its sequencer file.

□ 37. When PLC input 8 is activated, the EN status bit of instruction SQC R6:0 is set to logic state 1, since the rung containing this instruction (rung 0) becomes true. This causes instruction XIC R6:0/EN in rung 1 to become true, making instruction OTE O:0/0 in this rung true. This causes PLC output 0 to become activated.

Instruction SQC R6:0 is at position 1 of sequencer file B12. This occurs because the rung containing this instruction (rung 0) has gone from false to true, causing this instruction to step to the next position in sequencer file B12.

□ 38. Bits B12:1/0 and B12:1/2 are set to logic state 1.

□ 39. When PLC inputs 0 and 2 are activated, the FD status bit of instruction SQC R6:0 is set to logic state 1. This makes instruction XIC R6:0/FD in rung 3 true, making instruction OTE O:0/2 in this rung true. This causes PLC output 2 to become activated.

The reason why the FD status bit is set to logic state 1 is that all the non-masked bits at source address I:0.0 match the bits in the reference word at address B12:1. Thus, the bits at addresses I:0/0 and I:0/2 are set to logic state 1 due to PLC inputs 0 and 2 being activated. These bits are non-masked since the mask value of instruction SQC R6:0 is 00FFH (0000 0000 1111 1111 in binary). Consequently, all non-masked bits at source address I:0.0 match those in the reference word at address B12:1, whose bits B12:1/0 and B12:1/2 are set to logic state 1.

□ 40. When PLC input 8 is deactivated, the EN and FD status bits of instruction SQC R6:0 are set to logic state 0, since the rung containing this instruction (rung 0) becomes false.

The fact that the EN bit goes to logic state 0 makes instruction XIC R6:0/EN in rung 1 false, thereby making instruction OTE O:0/0 in this rung false and causing PLC output 0 to become deactivated.

The fact that the FD bit goes to logic state 0 makes instruction XIC R6:0/FD in rung 3 false, thereby making instruction OTE O:0/2 in this rung false and causing PLC output 2 to become deactivated.

□ 41. Yes. This occurs because activation of PLC input 8 causes the rung containing instruction SQC R6:0 (rung 0) to go from false to true, causing this instruction to step to the next position in its sequencer file.

□ 42. Yes. The reason why the FD status bit is set to logic state 1 is that the bits at addresses I:0/0, I:0/2, I:0/4, and I:0/6 are set to logic state 1, due to PLC inputs 0, 2, 4, and 6 being activated. Consequently, the status of all the non-masked bits at source address I:0.0 match the bits in the reference word at address B12:2, whose bits B12:2/0, B12:2/2, B12:2/4, and B12:2/6 are set to logic state 1.

□ 43. The EN status bit remains set to logic state 1 since the rung containing instruction SQC R6:0 remains true. However, the FD bit is set to logic state 0 since the non-masked bits at source address I:0.0 no longer match the bits in the reference word at address B12:2, due to bit I:0/6 being set to logic state 0.

□ 44. PLC inputs 0 through 7 must be activated in order for the FD status bit of instruction SQC R6:0 to be set to logic state 1. This occurs because bits B12:3/0 through B12:3/7 in the reference word at address B12:3 are all set to logic state 1.

□ 45. When PLC input 8 is deactivated and then activated, the DN bit of instruction SQC R6:0 is set to logic state 1 since this instruction steps to the last position in sequencer file B12 (position 4). This makes instruction XIC R6:0/DN in rung 2 true, thereby making instruction OTE O:0/1 in this rung true and causing PLC output 1 to become activated.

PLC inputs 1, 3, 5, and 7 must be activated in order for the FD status bit of instruction SQC R6:0 to be set to logic state 1. This occurs because bits B12:4/1, B12:4/3, B12:4/5, and B12:4/7 in the reference word at address B12:4 are set to logic state 1.

□ 46. Instruction SQC R6:0 returns to position 1 of sequencer file B12 to initiate a new cycle. The DN status bit is reset to logic state 0 because the rung containing instruction SQC R6:0 (rung 0) goes from false to true.

□ 47. When PLC input 9 is activated, instruction SQC R6:0 returns to position 0 of sequencer file B12, while the FD status bit is reset to logic state 0. This occurs because activation of PLC input 9 makes instruction XIC I:0/9 in rung 4 true, making instruction RES R6:0 in this rung true.

□ 48. Yes.

□ 49. If the mask value of instruction SQC R6:0 were changed to "00F0H" (0000 0000 1111 0000 in binary), bits 0 through 3 of source address I:0.0 would be masked, so that you would need to activate the following PLC inputs in order for the FD status bit to be set to logic state 1:

Position 1: None
Position 2: PLC inputs 4 and 6
Position 3: PLC inputs 4 through 7
Position 4: PLC inputs 5 and 7

□ 52.

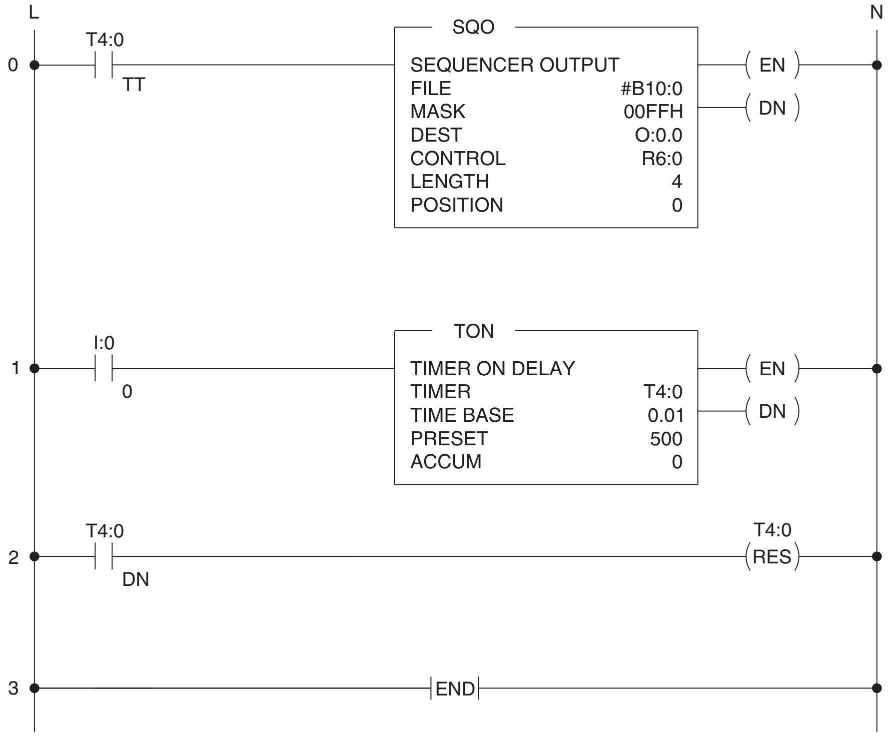


Figure 2. Suggested timer-driven SQO ladder program.

SEQUENCER POSITION	WORD	BINARY DATA			
		15	8	7	0
0	B10:0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	B10:1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 1
2	B10:2	0 0 0 0	0 0 0 0	0 0 0 0	1 1 1 1
3	B10:3	0 0 0 0	0 0 0 0	1 1 1 1	0 0 0 0
4	B10:4	0 0 0 0	0 0 0 0	1 1 1 1	1 1 1 1

Table 1. Data table for instruction SQO R6:0.

ANSWERS TO REVIEW QUESTIONS

1. The sequencer file is a file that stores data to be transferred to the destination file in the case of an SQO instruction, or reference data to be compared to the data at the source address in the case of an SQC instruction.

2. The mask value is a hexadecimal code or the address of a word or file through which the sequencer instruction transfers (SQO) or compares (SQC) data. Mask bits that are set to logic state 0 will mask data. Mask bits that are set to logic state 1 will pass data.

3. The SQO instruction transfers data from its sequencer file, through a mask, to a destination file on each false-to-true rung transition. Once the SQO instruction has reached the last position in its sequencer file, it automatically returns to position 1 on the next false-to-true rung transition.

4. The Found (FD) bit is set to logic state 1 when all the non-masked bits in the word or file at the source address match those of the corresponding reference word in the sequencer file. The FD bit is updated each time the processor evaluates the SQC instruction while the rung is true.

5. The Done (DN) bit passes from logic state 0 to logic state 1, and the SQO instruction returns to position 1 of its sequencer file to start a new cycle.