Module 1

Data Comparison Instructions

Allen Bradley

ControlLogix / CompactLogix

Module 1:
Interpreting and Editing Data Comparison Instructions

Basic Data Comparison Instructions

Note: Logix 5000 refers to either a CompactLogix or
ControlLogix processor

RSLogix 5000 can refer to either RSLogix 5000 or Studio 5000 (Logix Designer) application software.

To view Comparison instructions, click the Comparison tab on the Language Element Toolbar.

Use the tab scroll arrows in line with the Instruction Category tabs to view instruction tabs not shown on the default view.

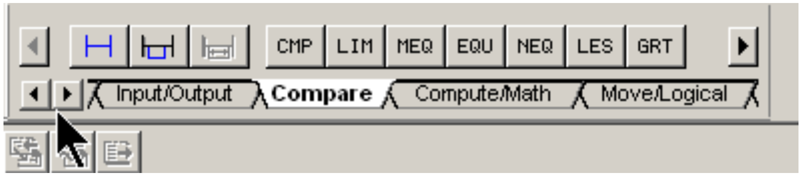


Figure 1-A Compare Instructions

To view all of the Comparison instructions use the left or right scroll arrows in-line with instruction types.

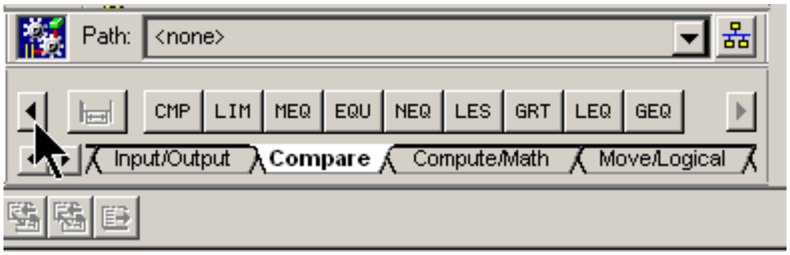


Figure 2-A Compare Instructions-Left Scroll Arrow

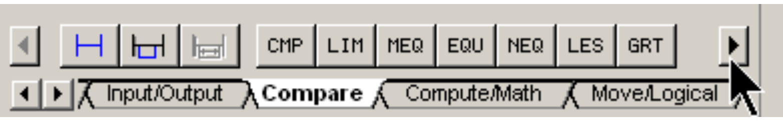


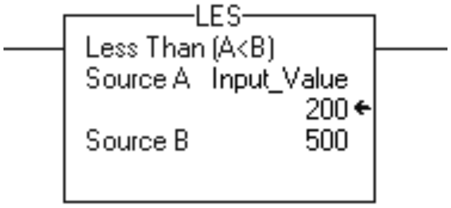
Figure 3-A Compare Instructions-Right Scroll Arrow

A Data Comparison Instruction will compare two data values, and if the instruction is true, it will pass logic power flow.

All Data Comparison instructions are input type of instructions.

There is no hard limit to the number of Data Comparison per Rung.

Source A in the instruction is compared to Source B. In the following illustration, the value in the Logix5000 tag 'Input_Value' (200) is compared to see if it is less than the constant of 500.



LES
Less Than (A<B)
Source A Input_Value
200
Source B 500

Figure 4-A. An LES Data Comparison Instructions.

Source A is usually the value in a memory location (tag) that has been created using the RSLogix 5000 Tag Editor. Valid data types that can be used are: SINT, INT. DINT, Real and String. (String tags should be used in the compare instructions; ASCII text cannot be entered as a constant.)

Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is the reference value). Valid data types that can be used are: SINT, INT. DINT, Real and String.

The data comparison instructions do not have status bits associated with them.

The following table shows the different Data Comparison Instructions and their mnemonics.

EQU	Equal To	(A = B)
LES	Less Than	(A < B)
LEQ	Less Than or Equal To	(A <= B)
GRT	Greater Than	(A > B)
GEQ	Greater Than or Equal To	(A >= B)
NEQ	Not Equal To	(A <> B)

Some more examples of data comparison instructions are shown in the following illustration.

In this example, if the value in tag ‘Timer_1.ACC’ is less than or equal to the value in tag ‘Low_Value’, the LEQ goes true and ‘Bit_Address.7’ tag True. If the value in tag ‘Timer_1.ACC’ is less than 700, the LES goes true and ‘Bit_Address.5’ is True.

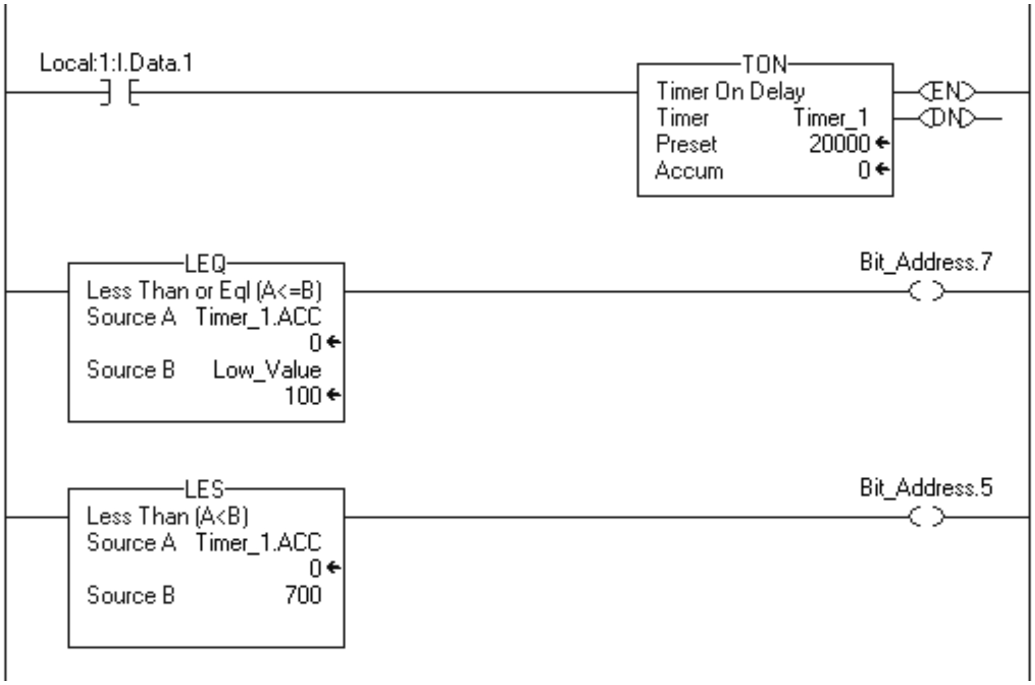


Figure 5-A. Data Comparison Instructions.

Unlike the older series of Allen-Bradley processors the ControlLogix / CompactLogix allows the comparison of ASCII strings. When strings are compared the hexadecimal values of the characters determine if one string is Equal to, Less than or Greater than another string. The hexadecimal codes for ASCII characters are available in several of Allen-Bradley's manual appendices.

Integer and Real data types can be compared to each other.

The following is an explanation of each of the Data Compare instructions:

LES

The LES instruction will test to see if Source A is less than Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is Source B value). In the following illustration, if the value in tag 'Input_Value' is less than 500, then 'Bit_Address.5'tag is True.

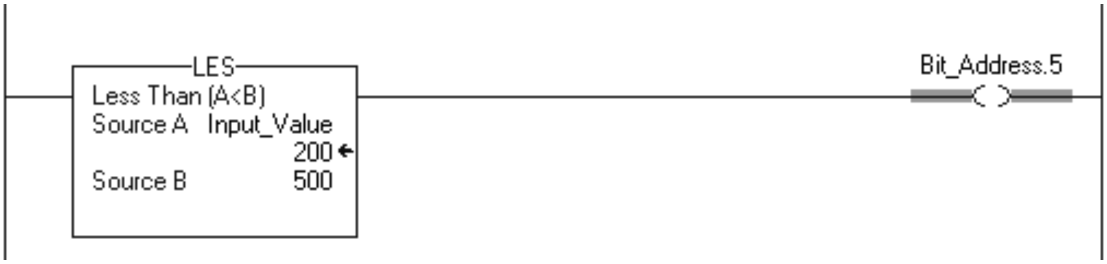


Figure 6-A. The LES Instruction.

LEQ

The LEQ instruction will test to see if Source A is less than or equal to Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in Source B is the reference value). In the following illustration, if the value in tag ‘Input_Value’ is less than or equal to 800 (the value in tag ‘High_Value’), then ‘Bit_Address.7’ tag will be True.

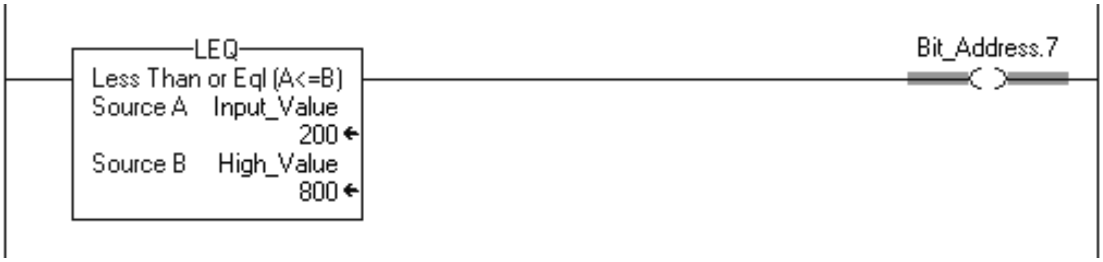


Figure 7-A. The LEQ Instruction.

EQU

The EQU instruction will test to see if Source A is equal to Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is the Source B value). In the following illustration, if the value in tag ‘Input_Value’ is equal to 335 (a constant), then ‘Bit_Address.3’tag will be True.

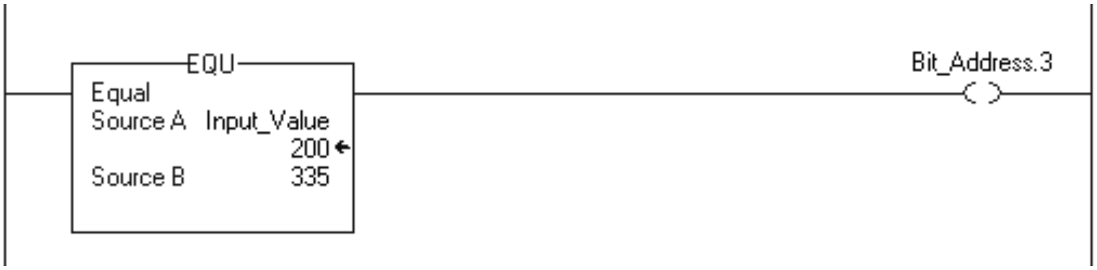


Figure 8-A. The EQU Instruction.

NEQ

The NEQ instruction will test to see if Source A is not equal to Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is the Source B value). In the following illustration, if the value in tag ‘Input_Value’ is not equal to 100 (the value in tag ‘Low_Value’), then ‘Bit_Address.4’ tag will be True.

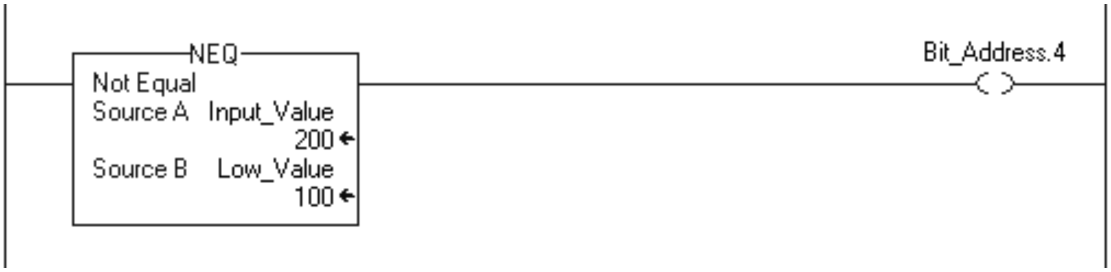


Figure 9-A The NEQ Instruction.

GRT

The GRT instruction will test to see if Source A is greater than Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is the Source B value). In the following illustration, if the value in tag ‘Input_Value’ is greater than 350 (a constant value), then ‘Bit_Address.6’ tag will be True.

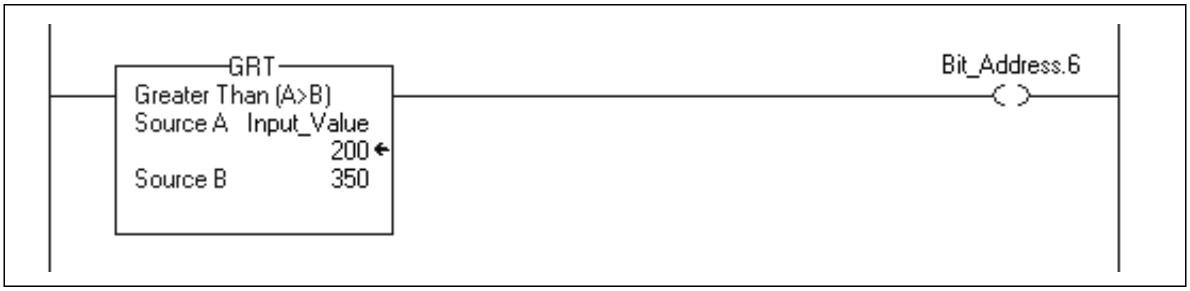


Figure 10-A. The GRT Instruction.

GEQ

The GEQ instruction will test to see if Source A is greater than or equal to Source B. Source A and Source B can be a constant number, or a tag located in the tag database (in this case, the value in that location is the Source B value).

In the following illustration, if the value in tag ‘Input_Value’ is greater than or equal to 800 (the value in tag ‘High_Value’), then ‘Bit_Address.8’ tag will be True.

Since Input_Value is not greater than or equal to High_Value the instruction is false and Bit_Address.8 will be False

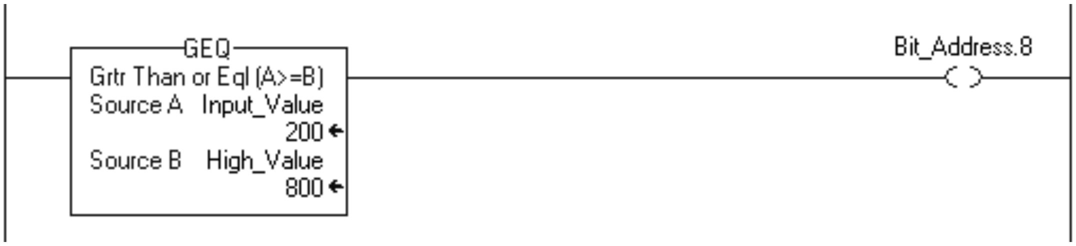


Figure 11-A. The GEQ Instruction.

The CMP and LIM Instructions

The CMP Instruction

The CMP instruction is an input type of instruction that compares values in an expression. When the expression is true, the CMP will be true and pass power flow.

The following illustration shows the CMP instruction. In this example, if the value in tag ‘Set_Point’ is less than or equal to the value in tag ‘High_Value’, then ‘Bit_Address.0’tag will be True.

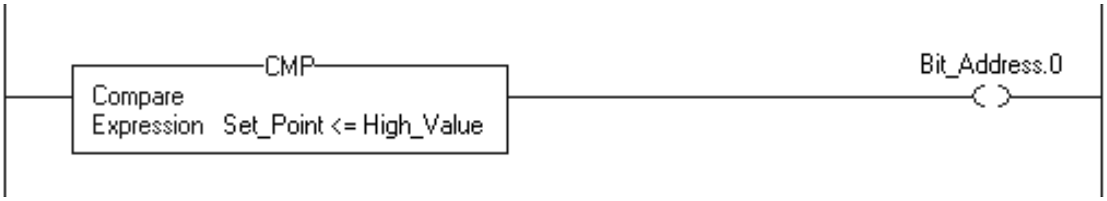


Figure 12-A. The CMP Instruction.

The disadvantage of using the CMP instruction is that it has a much greater execution time than the equivalent data compare instructions, and it uses more program memory to store the data than the data compares do. Also you cannot monitor the tags’ value as easily as with a basic compare instruction.

The advantage of using the CMP instruction is that it allows you to enter complex expressions in one instruction.

See page 11 for available Operators.

The following list will show you the valid operators that can be used in the CMP expression.

Operator	Description	Optimal
+	add	DINT, REAL
-	subtract/negate	DINT, REAL
*	multiply	DINT, REAL
/	divide	DINT, REAL
=	equal	DINT, REAL
<	less than	DINT, REAL
<=	less than or equal	DINT, REAL
>	greater than	DINT, REAL
>=	greater than or equal	DINT, REAL
<>	not equal	DINT, REAL
**	exponent (x to y)	DINT, REAL
ABS	absolute value	DINT, REAL
ACS	arc cosine	REAL
AND	bitwise AND	DINT
ASN	arc sine	REAL
ATN	arc tangent	REAL
COS	cosine	REAL
DEG	radians to degrees	DINT, REAL
FRD	BCD to integer	DINT
LN	natural log	REAL
LOG	log base 10	REAL
MOD	modulo-divide	DINT, REAL
NOT	bitwise complement	DINT
OR	bitwise OR	DINT
RAD	degrees to radians	DINT, REAL
SIN	sine	REAL
SQR	square root	DINT, REAL
TAN	tangent	REAL
TOD	integer to BCD	DINT
TRN	truncate	DINT, REAL
XOR	bitwise exclusive OR	DINT

The LIM Instruction

The LIM (Limit Test) instruction is an input type of instruction that tests a value to see if it is within or outside of a data range. In the most common configuration of the instruction, if the value is within the data range, the LIM will be true, and pass power flow. The following illustration shows a LIM instruction.

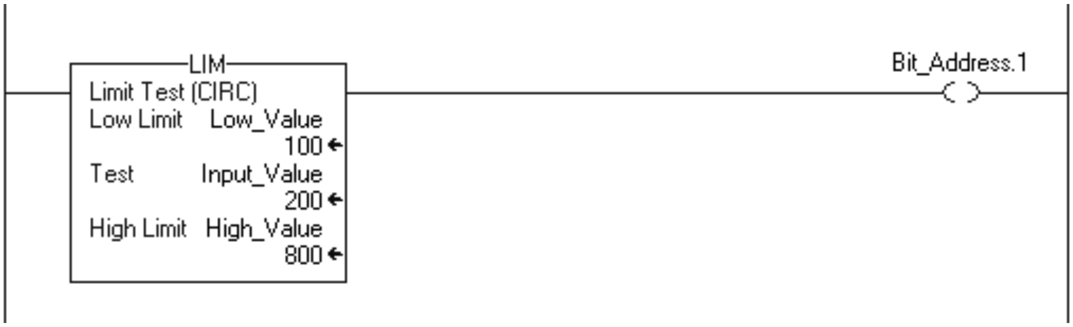


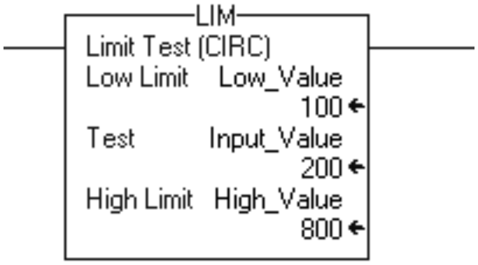
Figure 13-A. The LIM Instruction.

The Low Lim and High Lim parameters can be constants or tag memory locations. The Test value is the tag memory location that is under test.

The Low Lim and High Lim values are included in the data range, so if the test value is equal to one of the limits, the LIM is true.

The LIM instruction has what is termed as “Circular” capability, which means that the LIM can test positive and negative values to see if it is within a data range, or outside of a data range.

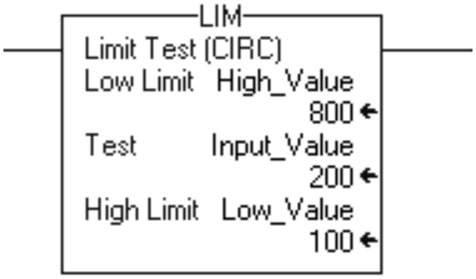
If the Low Lim value is less than the High Lim value, then the LIM will be true if the Test value is between the two limits.



```
LIM
Limit Test (CIRC)
Low Limit    Low_Value
                   100 ←
Test       Input_Value
                   200 ←
High Limit  High_Value
                   800 ←
```

Figure 14-A. The LIM instruction testing within a range.

If the Low Lim value is greater than the High Lim value, then the LIM will be true if the test value is outside of the two limits.



```
LIM
Limit Test (CIRC)
Low Limit   High_Value
                   800 ←
Test       Input_Value
                   200 ←
High Limit   Low_Value
                   100 ←
```

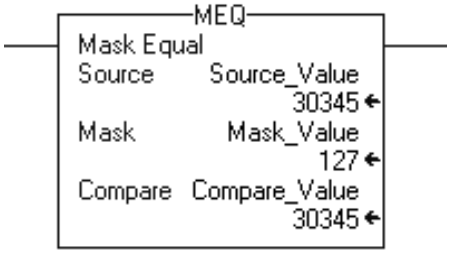
Figure 15-A. The LIM instruction testing outside the range.

The MEQ Instruction

The MEQ is an input type of instruction that compares bits in one tag with the bits in another tag. The MEQ has a mask tag, which can be used to mask out (filter) bits that do not need to be to compared.

When the bit pattern matches, the instruction is true and will pass power flow. If there is a mismatch of bits, the instruction is false.

The following illustration shows an MEQ instruction in a program.



Source Source_Value
30345

Mask Mask_Value
127

Compare Compare_Value
30345

Figure 16-A. The MEQ Instruction.

When the bits in the Source and Destination are bit for bit the same, on the bits where there is a "1" in the mask bit, the instruction will pass the power flow.

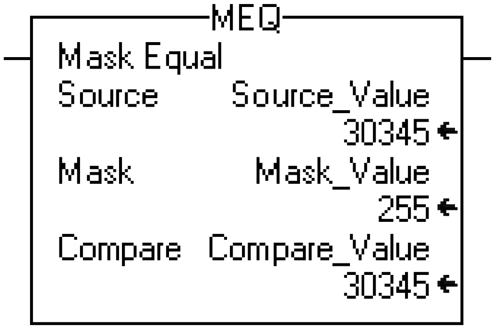
When entering a mask the programming software defaults to decimal display. If you need to enter the mask with another format, change to tag style. The following Table shows Prefixes that will precede the mask (or other tags) when the Style is not in decimal .

Prefix	Description
16#	Hexadecimal (ex. 16#0F0F)
8#	Octal (ex. 8#16)
2#	Binary (ex. 2#00110011)

Data types that can be used are SINT, INT and DINT. If you mix integer data types the upper bits of the smaller data type are filled with 0's by the instruction, so it is the same size as the largest data type.

Changing tags to Binary Style makes the instruction's data easier to interrupt.

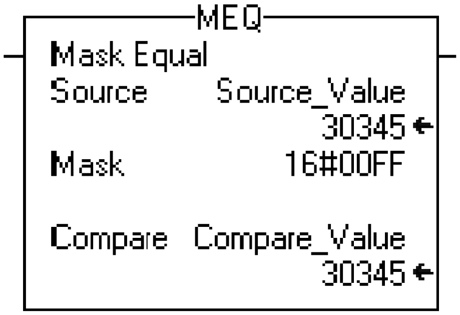
The following illustration will show how the bits are compared.



```
MEQ
Mask Equal
Source       Source_Value
                    30345 ←
Mask           Mask_Value
                      255 ←
Compare   Compare_Value
                    30345 ←
```

	Binary view	
Source_Value	0111 0110 1000 1001	Equals Decimal value of 30345
Mask_Value	0000 0000 1111 1111	Equals Decimal value of 255
Compare_Value	0111 0110 1000 1001	Equals Decimal value of 30345

Figure 17-A. The bit pattern in the MEQ Instruction.



```
MEQ
Mask Equal
Source       Source_Value
                    30345 ←
Mask             16#00FF

Compare   Compare_Value
                    30345 ←
```

	Binary view	
Source_Value	0111 0110 1000 1001	Equals Decimal value of 30345
Mask	0000 0000 1111 1111	Equals Hex value of 00FF
Compare_Value	0111 0110 1000 1001	Equals Decimal value of 30345

Figure 18-A. The Hex value for the Mask Data.

Review Questions

1. T F A hexadecimal value of “00FF” is equal to a value of “255” in decimal.

2. The mnemonic for the Less Than or Equal to instruction is:

a) LES

b) LOE

c) LQE

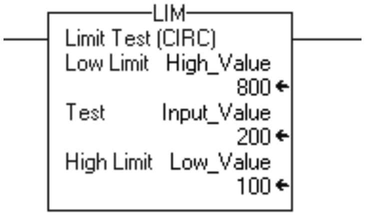
d) LEQ

3. T F A maximum of 10 data comparison instructions can be in series in a rung of PLC logic.

4. T F In a GRT instruction, Source A is compared to Source B, and will be true if Source A is greater than Source B.

5. T F The LIM instruction has a circular limit capability, which means that it will work across the full Integer numbering range.

6. **In the following Limit Test instruction, what value in tag ‘Input_Value’ will make the instruction false?**



```
LIM
Limit Test (CIRC)
Low Limit   High_Value
                 800 ←
Test       Input_Value
                 200 ←
High Limit   Low_Value
                 100 ←
```

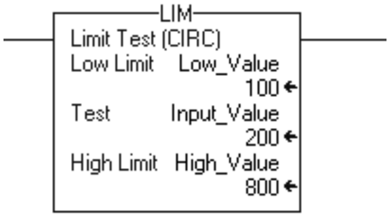
a) -40,060

b) 567

c) 1200

d) +42,000

7. **In the following Limit Test instruction, what value in tag “Input_Value” will make the instruction true?**



```
LIM
Limit Test (CIRC)
Low Limit    Low_Value
                 100 ←
Test       Input_Value
                 200 ←
High Limit  High_Value
                 800 ←
```

a) -40,060

b) 356

c) 1200

d) +42,000

8. T F The CMP instruction can do the same comparison as a LES instruction.

9. The mask value in an MEQ instruction will view the mask tag value in a _________ format (style) while in the ladder view by default.

a) BCD/HEX

b) Decimal

c) Binary

d) Octal

10. The mnemonic for a Not Equal to data comparison instruction is:

a) NEQ

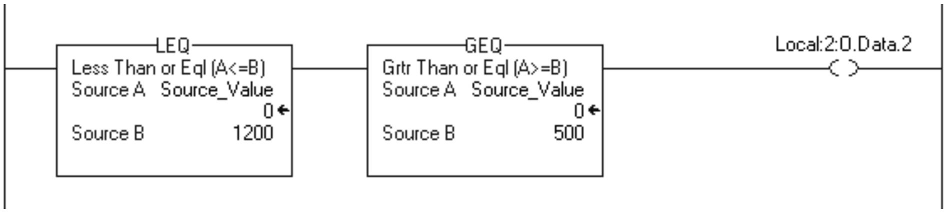
b) EQN

c) NTE

d) NOE

11. T F Integer and Floating Point type of data can be accurately compared to each other in a data compare instruction.

12. **In the following rung of logic, what value in tag Source_Value will shut the output off?**



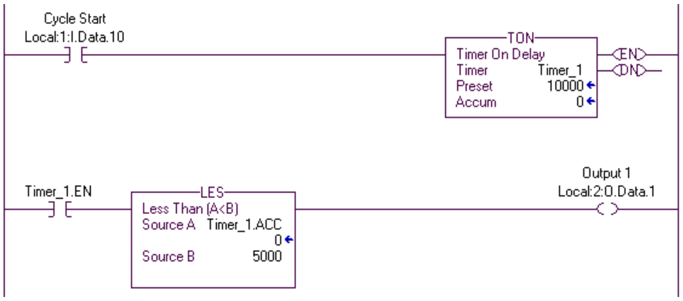
a) 550

b) 410

c) 1000

d) 1200

13. **Choose the answer that will best explain how the output responds after the "Cycle_Start" input is turned on.**



a) Output1 goes True (On) and stays on for 5 seconds.

b) Output1 goes True (On) and stays on for 10 seconds.

c) Output1 never comes on.

d) Output1 comes on after 5 seconds and stays on until the input is shut off.

Review Questions Answers

1. T

2. d

3. F

4. T

5. T

6. b

7. b

8. T

9. b

10. a

11. T

12. b

13. a

DOL DISCLAIMER:

This product was funded by a grant awarded by the U.S. Department of Labor’s Employment and Training Administration. The product was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties, or assurances of any kind, express or implied, with respect to such information, including any information on linked sites and including, but not limited to, accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability, or ownership.

This work is licensed under a Creative Commons Attribution 4.0 International License.