8.9 Character String Operation Instruction

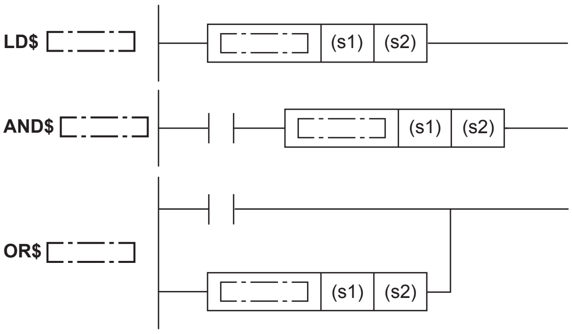
Comparing character strings

LD$□, AND$□, OR$□

FX5S FX5UJ FX5U FX5UC

These instructions perform a comparison operation between the character string data in the device specified by (s1) and later and the character string data in the device specified by (s2) and later. (Devices are used as a normally open contact.)

Ladder diagram



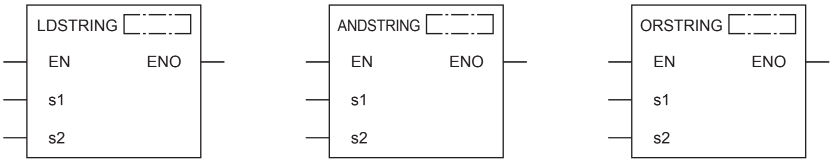
("=", "<>", ">", "<=", "<", ">=" enters □.)

Structured text[*1]

```
ENO:=LDSTRING_□(EN,s1,s2);
ENO:=ANDSTRING_□(EN,s1,s2);
ENO:=ORSTRING_□(EN,s1,s2);
```

("EQ", "NE", "GT", "LE", "LT", "GE" enters □.)[*2]

FBD/LD



("_EQ", "_NE", "_GT", "_LE", "_LT", "_GE" enters □.)[*2]

[*1] Supported by engineering tool version "1.035M" and later.

[*2] EQ is =, NE is <>, GT is >, LE is <=, LT is <, and GE is >=.

Setting data

■Descriptions, ranges, and data types

Operand	Description	Range	Data type	Data type (label)
(s1)	Comparison data or head device number where the comparison data is stored	—	Character string	ANYSTRING_SINGLE
(s2)	Comparison data or head device number where the comparison data is stored	—	Character string	ANYSTRING_SINGLE
EN	Execution condition	—	Bit	BOOL
ENO	Execution result	—	Bit	BOOL

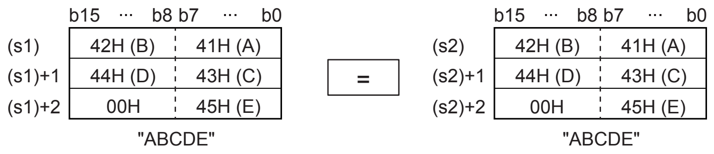
■Applicable devices

Operand	Bit	Word			Double word		Indirect specification	Constant			Others
	X, Y, M, L, SM, F, B, SB, S	T, ST, C, D, W, SD, SW, R	U□\G□	Z	LC	LZ		K, H	E	$	
(s1)	—	○[*1]	—	—	—	—	○	—	—	○	—
(s2)	—	○[*1]	—	—	—	—	○	—	—	○	—

[*1] T, ST, and C cannot be used.

Processing details

- These instructions perform a comparison operation between the character string data specified by (s1) and the character string data specified by (s2). (Devices are used as a normally open contact.)
- In the comparison operation, the ASCII codes of the character strings are compared one by one from the start of the strings.
- Character strings in the devices specified by (s1) and (s2) to a device that stores 00H are compared.
  - When all the character strings match, the comparison is considered as matched.



	b15 ··· b8	b7 ··· b0
(s1)	42H (B)	41H (A)
(s1)+1	44H (D)	43H (C)
(s1)+2	00H	45H (E)

"ABCDE"

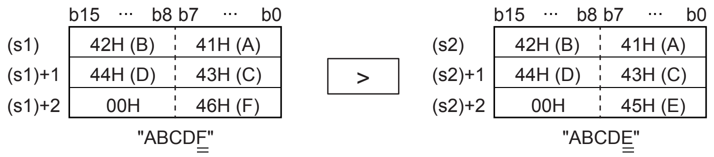
=

	b15 ··· b8	b7 ··· b0
(s2)	42H (B)	41H (A)
(s2)+1	44H (D)	43H (C)
(s2)+2	00H	45H (E)

"ABCDE"

Instruction symbol in □	Result
$=	Conductive state
$<>	Non-conductive state
$>	Non-conductive state
$<=	Conductive state
$<	Non-conductive state
$>=	Conductive state

  - When the character strings are different, the string with a large character code is considered as the large one.



	b15 ··· b8	b7 ··· b0
(s1)	42H (B)	41H (A)
(s1)+1	44H (D)	43H (C)
(s1)+2	00H	46H (F)

"ABCDF"

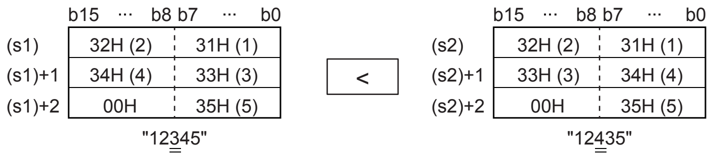
>

	b15 ··· b8	b7 ··· b0
(s2)	42H (B)	41H (A)
(s2)+1	44H (D)	43H (C)
(s2)+2	00H	45H (E)

"ABCDE"

Instruction symbol in □	Result
$=	Non-conductive state
$<>	Conductive state
$>	Conductive state
$<=	Non-conductive state
$<	Non-conductive state
$>=	Conductive state

  - When the character strings are different, the magnitude relation between them is determined based on the size of the first different character code.



	b15 ··· b8	b7 ··· b0
(s1)	32H (2)	31H (1)
(s1)+1	34H (4)	33H (3)
(s1)+2	00H	35H (5)

"12345"

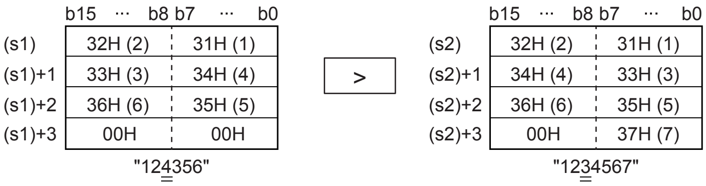
<

	b15 ··· b8	b7 ··· b0
(s2)	32H (2)	31H (1)
(s2)+1	33H (3)	34H (4)
(s2)+2	00H	35H (5)

"12435"

Instruction symbol in □	Result
$=	Non-conductive state
$<>	Conductive state
$>	Non-conductive state
$<=	Conductive state
$<	Conductive state
$>=	Non-conductive state

- When the length of the character string data differs for (s1) and (s2), the relative sizes of the character strings are determined based on the size of the first different character code.



	b15 ··· b8	b7 ··· b0
(s1)	32H (2)	31H (1)
(s1)+1	33H (3)	34H (4)
(s1)+2	36H (6)	35H (5)
(s1)+3	00H	00H

"124356"

>

	b15 ··· b8	b7 ··· b0
(s2)	32H (2)	31H (1)
(s2)+1	34H (4)	33H (3)
(s2)+2	36H (6)	35H (5)
(s2)+3	00H	37H (7)

"1234567"

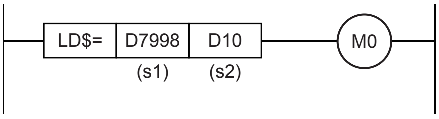
Instruction symbol in □	Result
$=	Non-conductive state
$<>	Conductive state
$>	Conductive state
$<=	Non-conductive state
$<	Non-conductive state
$>=	Conductive state

- If the character string specified by (s1) or (s2) has more than 16383 characters, the operation result is the non-conductive state.

Precautions

- In character string comparison operation, if the target device range does not have "00H", the values until the last number of the device are retrieved. Thus, even if the target device range does not have "00H", a comparison operation result is output when a mismatch between the acquired character strings is detected.

Ex.



Data of (s1)

D7998	42H(B)	41H(A)
D7999	44H(D)	43H(C)

Data of (s2)

D10	5A(Z)	41H(A)
D11	00H	43H(C)

- For the data specified by (s1) and (s2) as shown above, the second character is different between them. Thus, the operation result is non-conductive.

Operation error

There is no operation error.