12 String instructions

last modified by admin

on 2022/06/08 12:57

Table of Contents

LEN/string length detection

LEN(P)

After detecting the length of the character string specified in (s), store it after the device number specified in (d).

The data from the device number designated in (s) to the device number of 00H is treated as a character string.

-[LEN (s) (d)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s)	String or start number of the device storing the string	-	String	ANYSTRING_SINGLE
(d)	Store the device number of the detected character string length	-	Signed BIN 16 bit	ANY16

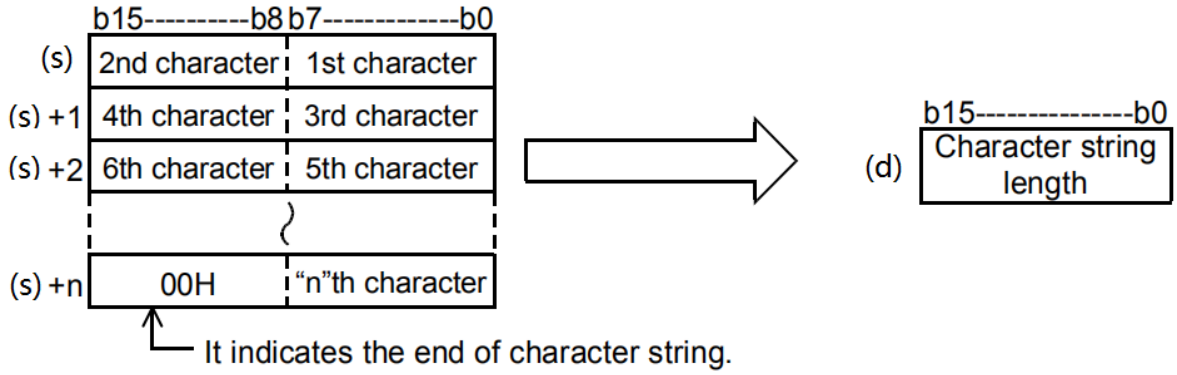
Device used

Instruction	Parameter	Devices									Offset modification	Pulse extension
		KnX	KnY	KnM	KnS	T	C	D	R	SD	[D]	XXP
LEN	Parameter 1	●	●	●	●	●	●	●	●	●	●	●
	Parameter 2		●	●	●	●	●	●	●	●	●	●

Features

After detecting the length of the character string specified in (s), store it after the device number specified in (d).

The data from the device number specified in (s) to the stored device number of 00H is treated as a character string.

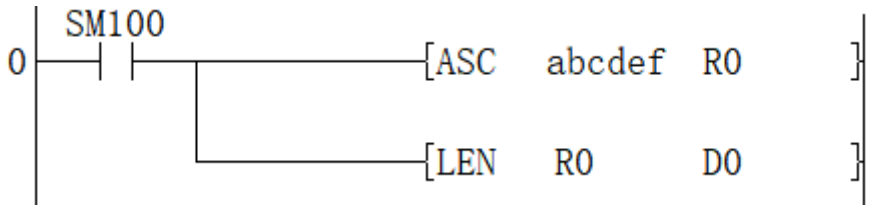


Error code

Error code	Content
4085H	(s) The read address exceeds the device range
408AH	(s) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s) When reading a character string, the maximum range of the device is read, but 00H is not found and the end

4086H	(d) When using offset, the offset address exceeds the device range

Example



For example, the above Circuit program

Use the asc instruction to write the string abcdef to the address starting from R0.

Then use the LEN instruction to determine the length. At this time, D0 will display 6.

LEFT/Extract from the left side of the string

LEFT(P)

For the character string data stored after the device number specified in (s), the data of (N) characters starting from the left side of the character string (the beginning of the character string) is stored in the device specified in (d) After numbering.

-[LEFT (s) (d) (N)]

Content, range and data type

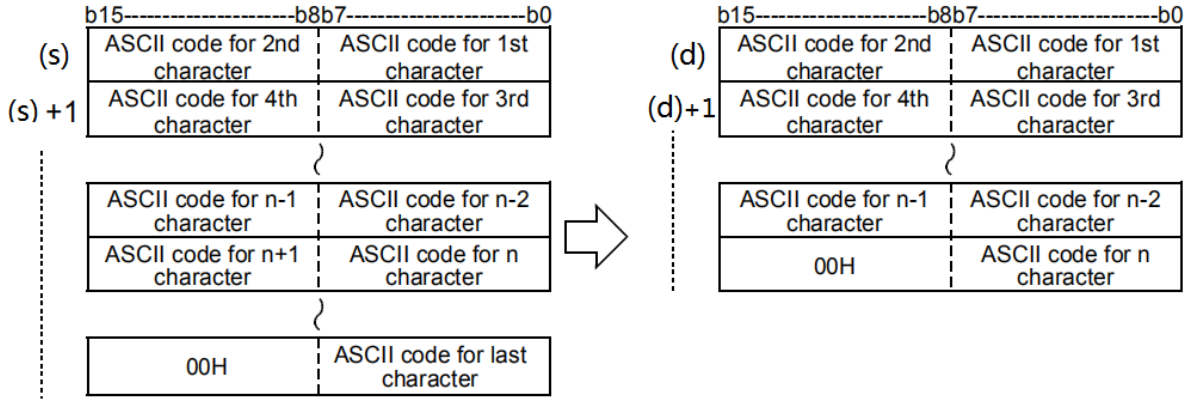
Parameter	Content	Range	Data type	Data type (label)
(s)	String or start number of the device storing the string	-	String	ANYSTRING_SINGLE
(d)	The start number of the device that stores the (N) character string from the left of (s)	-	String	ANYSTRING_SINGLE
(N)	Number of characters extracted	1 to 400	Signed BIN 16 bit	ANY16

Device used

Instruction	Parameter	Devices											Offset modification [D]	Pulse extension XXP
		KnX	KnY	KnM	KnS	T	C	D	R	SD	K	H		
LEFT	Parameter 1	●	●	●	●	●	●	●	●	●			●	●
	Parameter 2		●	●	●	●	●	●	●	●			●	●
	Parameter 3		●	●	●	●	●	●	●	●	●	●	●	●

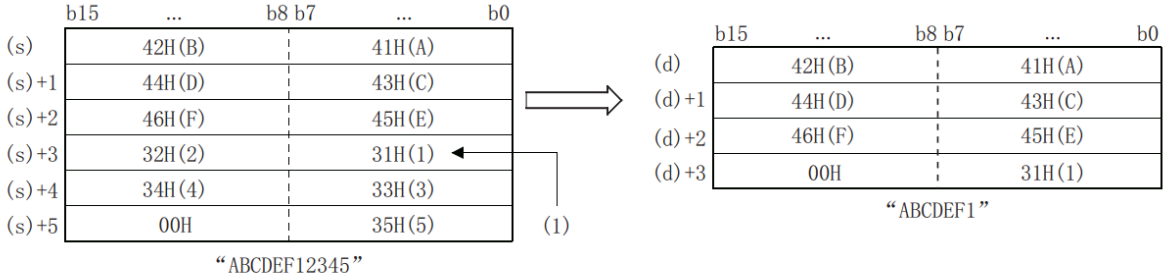
Features

For the character string data stored after the device number specified in (s), the data of (N) characters starting from the left side of the character string (the beginning of the character string) is stored in the device specified in (d) After numbering.



The character string specified in (s) is the data from the specified device to the position where "00H" is first detected in byte units.

(N)=7:



(1) :Ascall code for 7th character

The final NULL code (00H) representing the character string will be automatically appended to the end of the character string data.

If the number of extracted characters is an odd number, "00H" is stored in the upper byte of the device storing the final character. If the number of extracted characters is an even number, "0000H" is stored in the device after the device storing the final character.

When the number of characters specified in (N) is 0, the NULL code (00H) is stored in (d).

#Note:

When handling character codes other than ASCII codes, pay attention to the following points.

#The number of characters is handled in byte units (8 bits). Therefore, like the shifted JIS code, the character code of 1 character is represented by 2 bytes, and the number of characters of 1 character is "2".

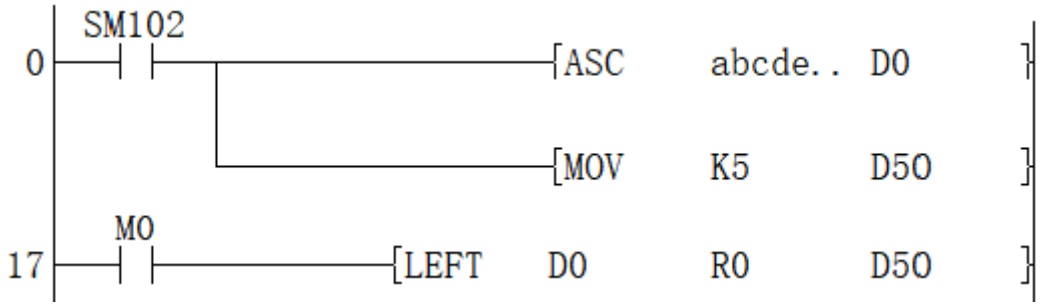
#When extracting a character string from a character string containing a character code representing one character in 2 bytes, such as the shift JIS code, the number of characters to be extracted should be considered in the unit of the character code of one character. If only 1 byte of the 2-byte character code is extracted, it will not be the expected character code, so be careful.

Error code

Error code	Content
4085H	(s) The read address exceeds the device range
408AH	(s) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s) When reading a character string, the maximum range of the device is read, but 00H is not found and the end

4084H	(N)<1 or (N)> string length
4086H	(d) The write address exceeds the device range

Example



From the "a b c d e f" starting from D0, take out 5 characters from the left to the R0 type. The character string of R0 is "a b c d e".

R0	1	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	ab
R1	1	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	cd
R2	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	e.
R3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	..

RIGHT/Extract from the right side of the string

RIGHT(P)

For the string data stored after the device number specified in (s), the data of (N) characters starting from the right side of the string (the end of the string) is stored in the device number specified in (d) after.

-[RIGHT (s) (d) (N)]

Content, range and data type

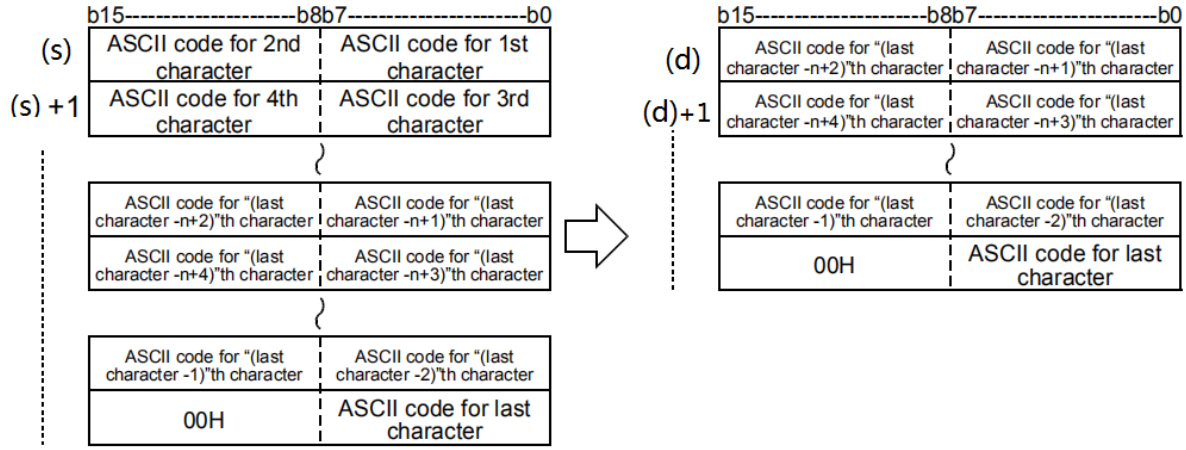
Parameter	Content	Range	Data type	Data type (label)
(s)	String or start number of the device storing the string	-	String	ANYSTRING_SINGLE
(d)	The start number of the device that stores the (N) character string from the right of (s)	-	String	ANYSTRING_SINGLE
(N)	Number of characters extracted	1 to 400	Signed BIN 16 bit	ANY16

Device used

Instruction	Parameter	Device											Offset modification	Pulse expansion
		KnX	KnY	KnM	KnS	T	C	D	R	SD	K	H	[D]	XXP
RIGHT	Parameter 1	●	●	●	●	●	●	●	●	●				
	Parameter 2		●	●	●	●	●	●	●	●				
	Parameter 3		●	●	●	●	●	●	●	●	●	●		

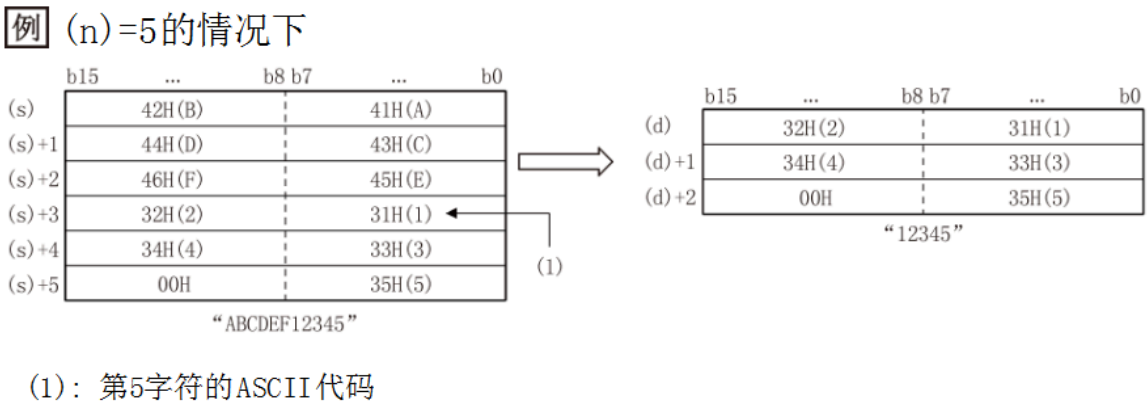
Features

For the string data stored after the device number specified in (s), the data of (N) characters starting from the right side of the string (the end of the string) is stored in the device number specified in (d) after.



The character string specified in (s) is the data from the specified device to the position where "00H" is first detected in byte units.

(N)=5:



The final NULL code (00H) representing the character string will be automatically appended to the end of the character string data.

If the number of extracted characters is an odd number, "00H" is stored in the upper byte of the device storing the final character. If the number of extracted characters is an even number, "0000H" is stored in the device after the device storing the final character.

When the number of characters specified in (N) is 0, the NULL code (00H) is stored in (d).

#Note:

When handling character codes other than ASCII codes, pay attention to the following points.

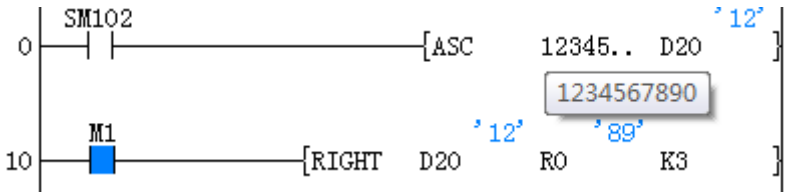
#The number of characters is handled in byte units (8 bits). Therefore, like the shifted JIS code, the character code of 1 character is represented by 2 bytes, and the number of characters of 1 character is "2".

#When extracting a character string from a character string containing a character code representing one character in 2 bytes, such as the shift JIS code, the number of characters to be extracted should be considered in the unit of the character code of one character. If only 1 byte of the 2-byte character code is extracted, it will not be the expected character code, so be careful.

Error code

Error code	Content
4085H	(s), (N) The read address exceeds the device range
408AH	(s) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s) When reading a character string, the maximum range of the device is read, but 00H is not found and the end
4084H	(N)<1 or (N)> string length
4086H	(d) The write address exceeds the device range

Example



Get 3 characters “890” from the right in the string "1234567890" and store them in R0

R0	0	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	89
R1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0.

Any extraction from MIDR/string

MIDR(P)

Store the data at any position in the character string data after the device number specified in (d).

-[MIDR (s1) (d) (s2)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s1)	String or start number of the device storing the string	-	String	ANYSTRING_SINGLE
(d)	Start number of the device storing the character string data of the operation result	-	String	ANYSTRING_SINGLE
(s2)	The start number of the device that stores the start character position and the number of characters (s2)+0: the position of the starting character, (s2)+1: the number of characters is signed	-	Signed BIN 16 bit	ANY16_ARRAY

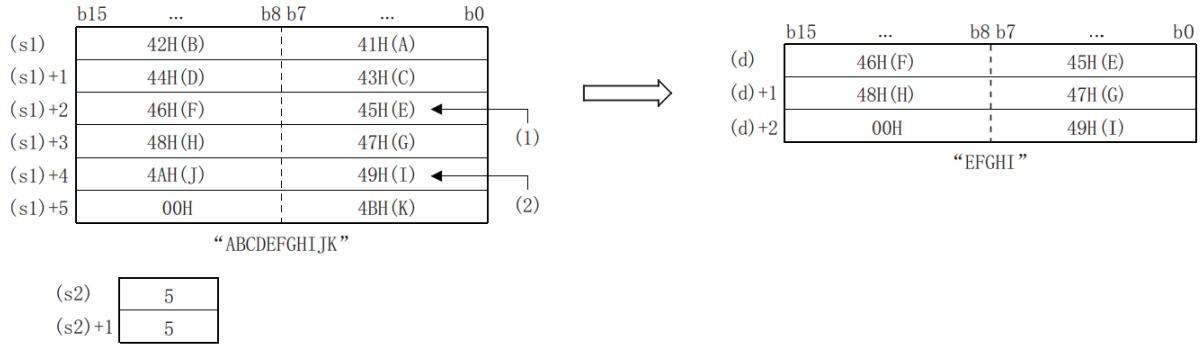
Device used

Instruction	Parameter	Devices									Offset modification [D]	Pulse extension XXP
		KnX	KnY	KnM	KnS	T	C	D	R	SD		
MIDR	Parameter 1	●	●	●	●	●	●	●	●	●	●	●

Parameter 2	●	●	●	●	●	●	●	●	●	●
Parameter 3	●	●	●	●	●	●	●	●	●	●

Features

For the character string data stored after the device number specified in (s1), the data of the character specified in (s2)+1 starting from the specified position in (s2) is stored to the device number specified in (d) and later .



1. : The position of the 5th character (S2).
2. : ASCII code (S2)+1 of the 5th character.

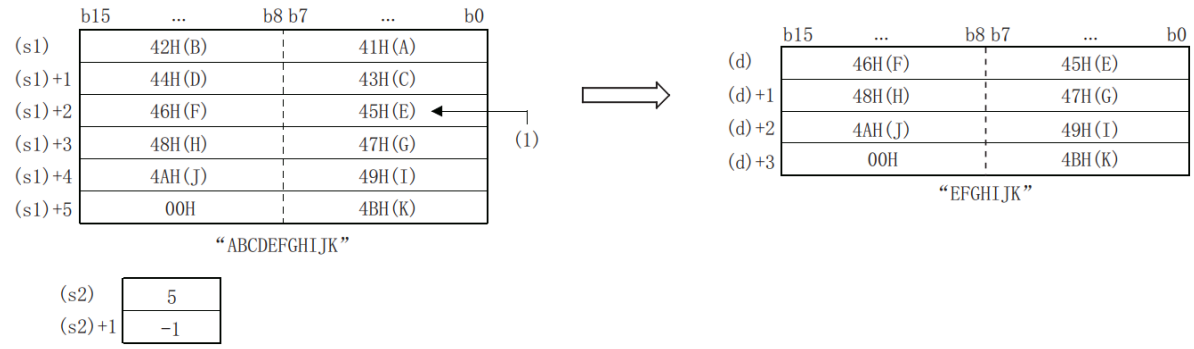
The character string specified in (s1) is the data from the specified device to the position where "00H" is first detected in byte units.

The final NULL code (00H) representing the character string will be automatically appended to the end of the character string data.

If the number of extracted characters "(s2)+1" is an odd number, "00H" is stored in the upper byte of the device storing the final character. If the number of extracted characters "(s2)+1" is an even number, "0000H" is stored in the device after the device storing the final character.

(s2) If the number of characters specified in +1 is 0, no processing is performed.

When the number of characters specified in (s2)+1 is -1, the data up to the final character data specified in (s1) is stored in the device specified in (d) and later.



(1): The position of the 5th character (S2).

#Note:

When handling character codes other than ASCII codes, pay attention to the following points.

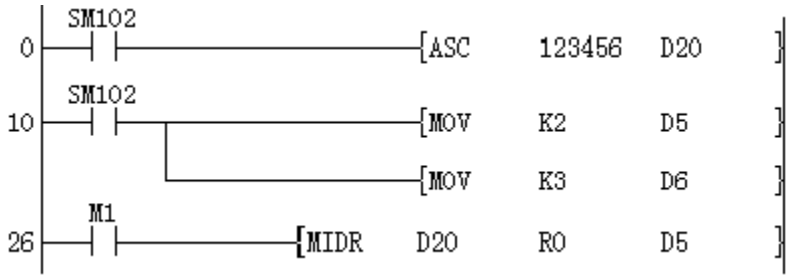
#The number of characters is handled in byte units (8 bits). Therefore, like the shifted JIS code, the character code of 1 character is represented by 2 bytes, and the number of characters of 1 character is "2".

#When extracting a character string from a character string containing a character code representing one character in 2 bytes, such as the shift JIS code, the number of characters to be extracted should be considered in the unit of the character code of one character. If only 1 byte of the 2-byte character code is extracted, it will not be the expected character code, so be careful.

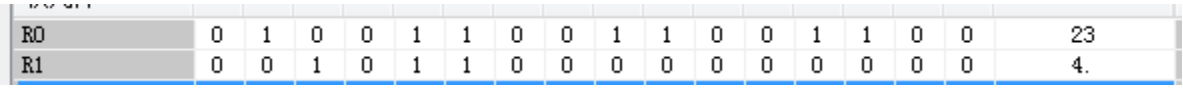
Error code

Error code	Content
4085H	(s1), (s2) The read address exceeds the device range
408AH	(s1) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s1) When reading a character string, the maximum range of the device is read, but 00H is not found.
4084H	(s2) When the value of +1 is -2 (including -2) or less. When the value of (s2) exceeds the number of characters in (s1). When the value of (s2) is negative. When the value of (s2)+1 exceeds the number of characters of (s1). When the value of (s2) and (s2) + 1 after the addition operation exceeds the number of characters of (s1).
4086H	(d) The write address exceeds the device range

Example



Get three characters "234" from the second character of the string "123456" into R0.



$MOV/ string transfer

$MOV(P)

Transfer the character string data specified in (s) to the device number specified in (d) and later.

-[$MOV (s) (d)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s)	Transmission string (maximum 255 characters) or the start number of the device storing the string	-	String	ANYSTRING_SINGLE

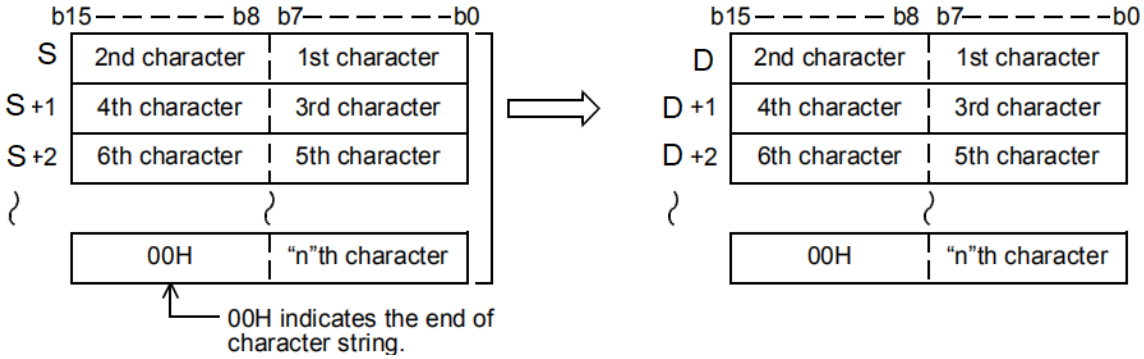
(d)	The start number of the device storing the transferred character string	-	String	ANYSTRING_SINGLE	

Device used

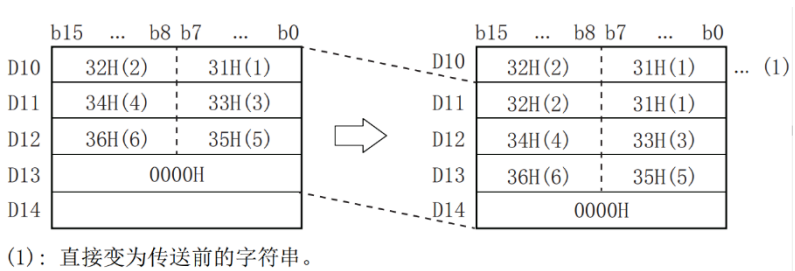
Instruction	Parameter	Device									Offset modification	Pulse expansion
		KnX	KnY	KnM	KnS	T	C	D	R	SD	[D]	XXP
$MOV	Parameter 1	●	●	●	●	●	●	●	●	●	●	●
	Parameter 2		●	●	●	●	●	●	●	●	●	●

Features

Transfer the character string data specified in (s) to the device number specified in (d) and later. In the transmission of a character string, the character string enclosed by the "" (double quotation marks) specified in (s) or the character string starting from the device number to the device number storing 00H is transmitted once.

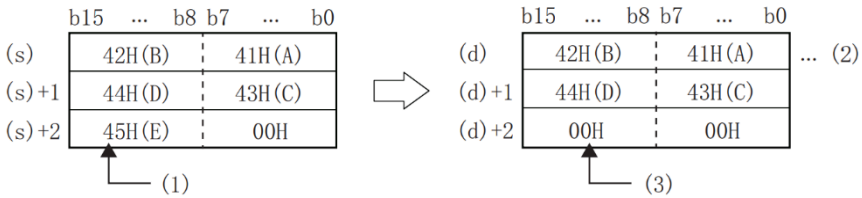


Even if the device range (s) to (s)+n storing the transferred character string data overlaps with the device range (d) to (d)+n storing the transferred character string data, it will be normal To process. For example, when the character string stored in D10 to D13 is transferred to D11 to D14, the situation is as follows.



(1): It directly becomes the character string before transmission.

When 00H is stored in the low byte of (s)+n, both the high byte and low byte of (d)+n will store 00H.



(1): The upper byte cannot be transmitted.

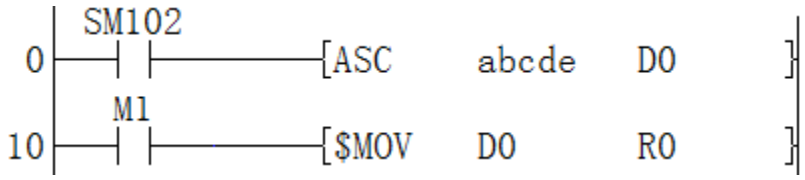
(2): It directly becomes the character string before transmission.

(3): The upper byte automatically stores 00H.

Error code

Error code	Content
4085H	(s) The read address exceeds the device range
408AH	(s) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s) When reading a character string, the maximum range of the device is read, but 00H is not found and the end
4086H	(d) The write address exceeds the device range

Example



Copy the string "a b c d e" in D0 to R0.

R0	1	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	ab
R1	1	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	cd
R2	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	e.

Arbitrary replacement in MIDW/string

MIDW(P)

For the string data stored after the device number specified in (s1), the data of the character specified in (s2)+1 is stored in the string data stored after the device number specified in (d) After the position specified in (s2).

-[MIDW (s1) (d) (s2)]

Content, range and data type

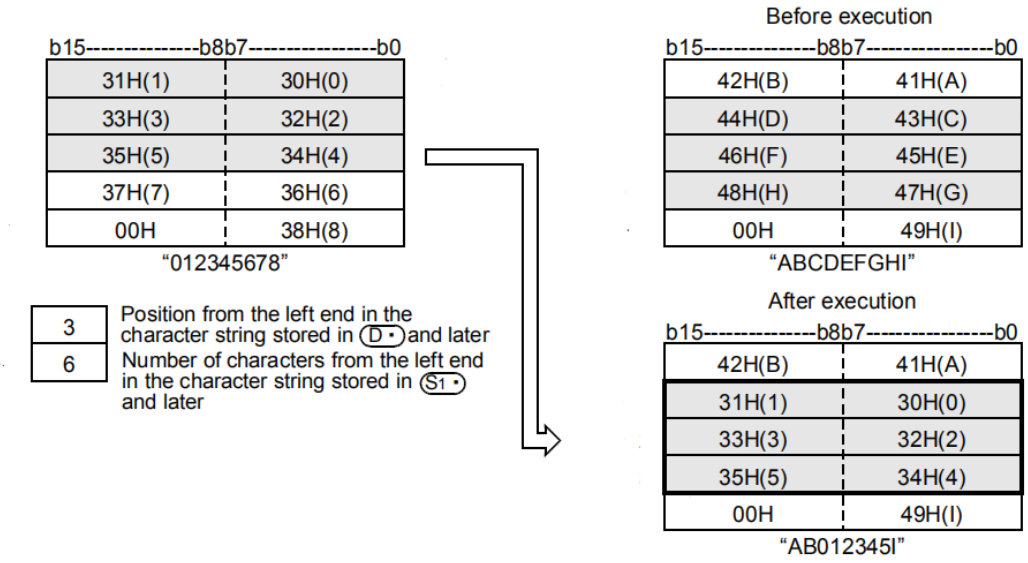
Parameter	Content	Range	Data type	Data type (label)
(s1)	String or start number of the device storing the string	-	String	ANYSTRING_SINGLE
(d)	Start number of the device storing the character string data of the operation result	-	String	ANYSTRING_SINGLE
(s2)	The start number of the device that stores the start character position and the number of characters (s2)+0: the position of the starting character, (s2)+1: the number of characters is signed	-	Signed BIN 16 bit	ANY16_ARRAY

Device used

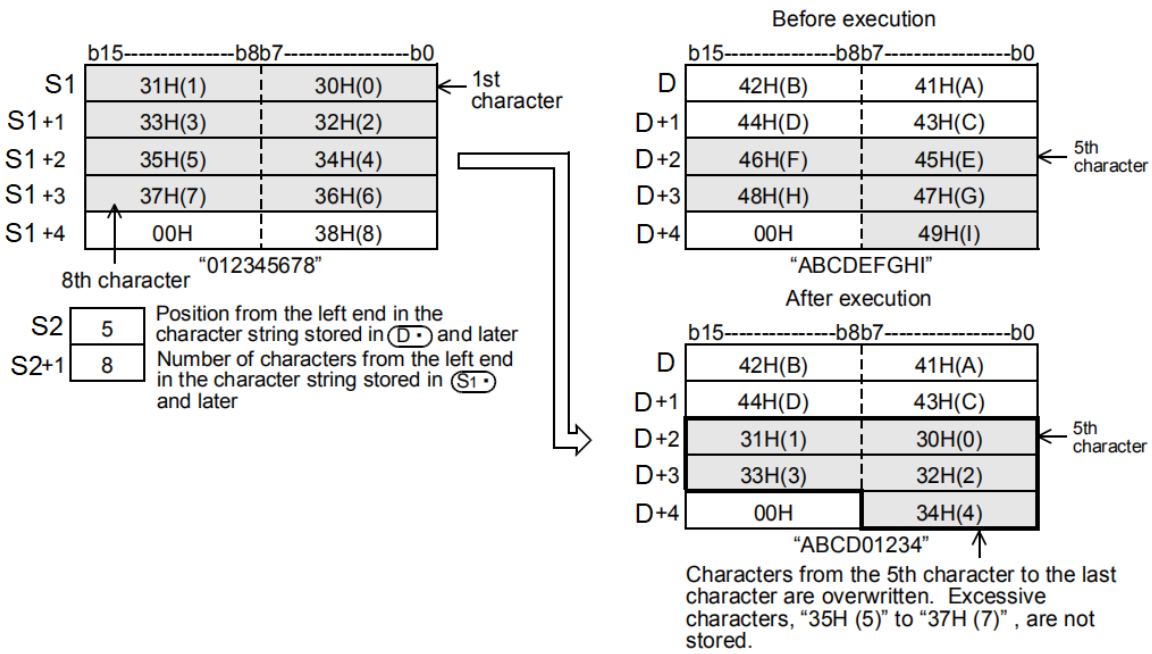
Instruction	Parameter	Device									Offset modification [D]	Pulse extension XXP
		KnX	KnY	KnM	KnS	T	C	D	R	SD		
MIDW	Parameter 1	●	●	●	●	●	●	●	●	●	●	●
	Parameter 2		●	●	●	●	●	●	●	●	●	●
	Parameter 3	●	●	●	●	●	●	●	●	●	●	●

Features

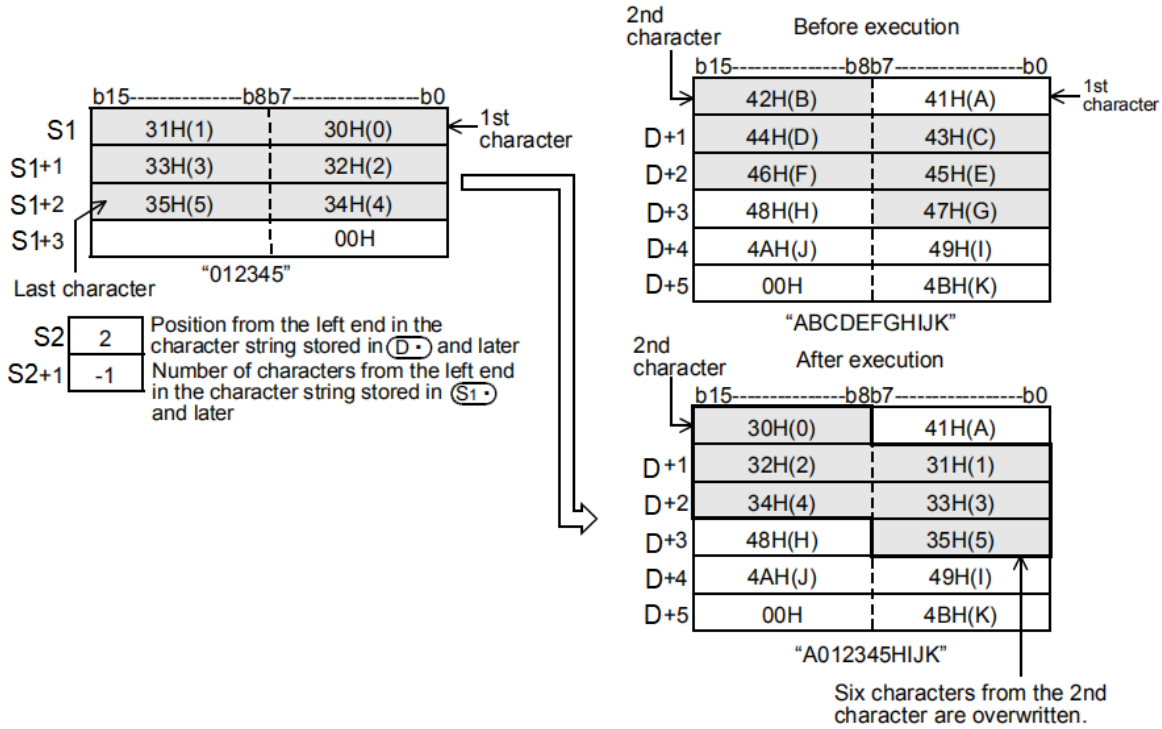
For the string data stored after the device number specified in (s1), the data of the character specified in (s2)+1 is stored in the string data stored after the device number specified in (d) After the position specified in (s2).



- The character string specified in (s1) or (d) is the data from the specified device to the position where "00H" is first detected in byte units.
- The final NULL code (00H) representing the character string will be automatically appended to the end of the character string data.
- If the number of characters specified in (s2)+1 is 0, no processing is performed.
- If the number of characters specified in (s2)+1 exceeds the last character of the character string data specified in (d), the data up to the last character of (d) is stored.



When the number of characters specified in (s2)+1 is -1, the data up to the final character data specified in (s1) is stored in the device specified in (d) and later.



#Note:

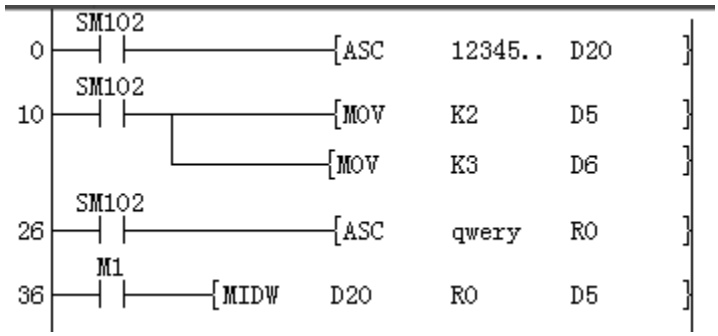
- When handling character codes other than ASCII codes, pay attention to the following points.
- The number of characters is handled in byte units (8 bits). Therefore, like the shifted JIS code, the character code of 1 character is represented by 2 bytes, and the number of characters of 1 character is "2".
- When extracting a character string from a character string containing a character code representing one character in 2 bytes, such as the shift JIS code, the number of characters to be extracted should be

considered in the unit of the character code of one character. If only 1 byte of the 2-byte character code is extracted, it will not be the expected character code, so be careful.

Error code

Error code	Content
4085H	(s1) (s2) (d) The read address exceeds the device range
408AH	(s1) (d) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s1) (d) When reading a character string, the maximum range of the device is read, but 00H is not found.
4084H	(s2) When the value of +1 is -2 (including -2) or less. When the value of (s2) exceeds the number of characters in (d). When the value of (s2) is negative. When the value of (s2)+1 exceeds the number of characters of (s1).
4086H	(d) The write address exceeds the device range

Example



Replace the three-character-length characters starting with the second character in the character string "q w e r y" stored in R0 with the first three characters in D20.

The result of R0 is "q123y".

R0	1	0	0	0	1	1	1	0	1	0	0	0	1	1	0	0	q1
R1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
R2	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	y.

STR/BIN 16-bit data → character string conversion

STR(P)

The BIN 16-bit data specified in (s2) is converted into a character string after a decimal point is added to the position specified in (s1), and stored in the device number specified in (d) or later.

-[STR (s1) (s2) (d)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s1)	The start number of the device that stores the number of digits of the converted value	-	Signed BIN 16 bit	ANY16_S_ARRAY
(s2)	Converted BIN data	-32768 to +32767	Signed BIN 16 bit	ANY16_S

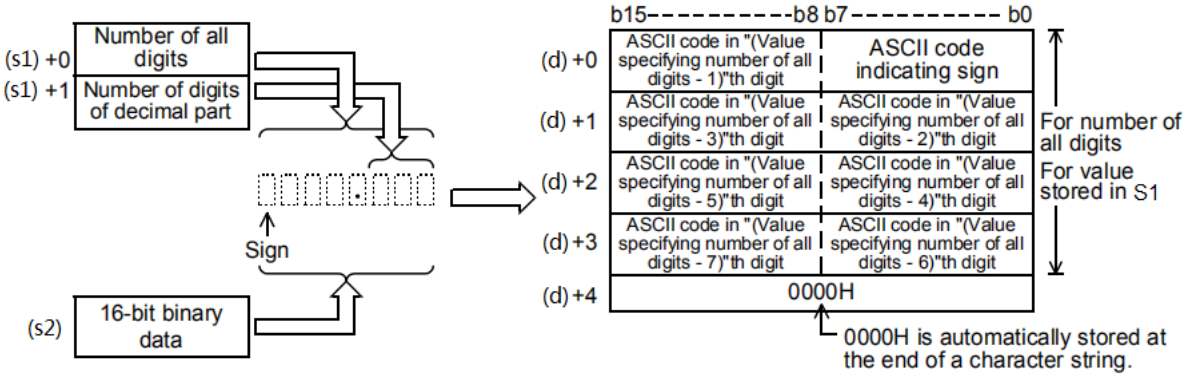
(d)	Start number of the device storing the converted character string	-	String	ANYSTRING_SINGLE

Device used

Instruction	Parameter	Device											Offset modification	Pulse extension
		KnX	KnY	KnM	KnS	T	C	D	R	SD	K	H	[D]	XXP
STR	Parameter 1	●	●	●	●	●	●	●	●	●			●	●
	Parameter 2	●	●	●	●	●	●	●	●	●	●	●	●	●
	Parameter 3					●	●	●	●	●			●	●

Features

The BIN 16-bit data specified in (s2) is converted into a character string after a decimal point is added to the position specified in (s1), and stored in the device number specified in (d) or later.



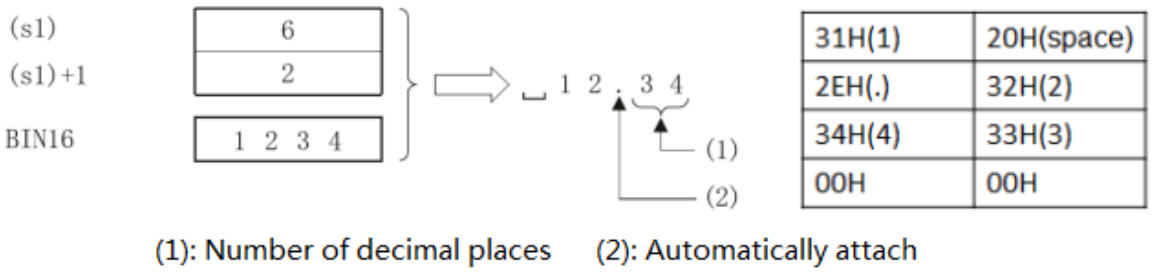
All digits that can be specified in (s1) are 2 to 8 digits.

The number of decimal places that can be specified in (s1)+1 is 0 to 5 digits. However, the setting should satisfy the condition that the number of decimal places ≤ (all digits-3).

The converted character string data will be stored in the device numbers after (d) as follows.

• In the sign, BIN 16-bit data will store 20H (blank) when it is positive, and 2DH (—) when it is negative.

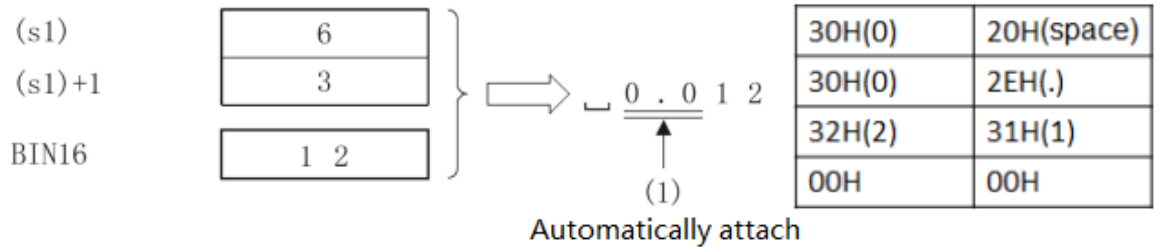
• When the number of decimal places is set to other than 0, 2EH(.) is automatically stored in the specified digit + 1 digit. When the decimal place is 0, 2EH(.) is not stored.



(1): Number of decimal places

(2): Automatically attach

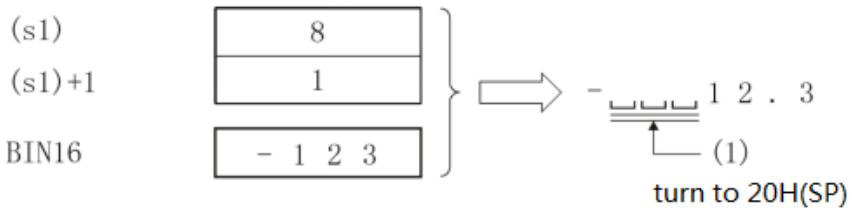
If the value of the decimal place is greater than the number of digits of the BIN 16-bit data, 0 is automatically appended and converted to "0.***" right-aligned.



30H(0)	20H(space)
30H(0)	2EH(.)
32H(2)	31H(1)
00H	00H

(1): Automatically attach

In the value of all digits, excluding the sign, and if the number of digits after the decimal point is greater than the number of BIN 16-bit data, 20H (blank) is stored between the sign and the value. If the digit of BIN 16-bit data is larger, it will be in error status.



20H(space)	2DH(-)
20H(space)	20H(space)
32H(2)	31H(1)
33H(3)	2EH(.)
00H	00H

(1): Change to 20H (SP).

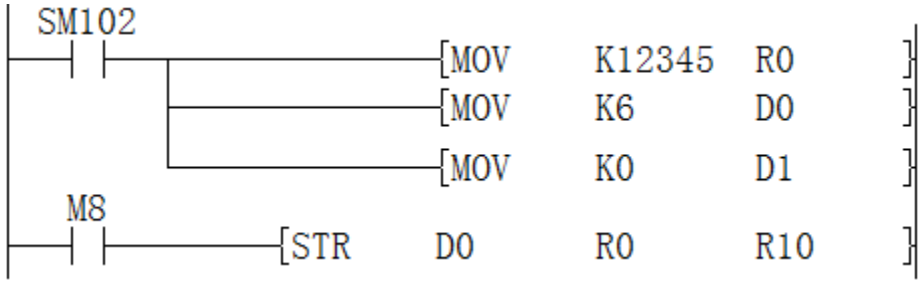
00H is automatically stored at the end of the converted character string.

• When the total digits are even digits, "0000H" is stored in the device after the device storing the final character. In the case of an odd number of digits, "00H" is stored in the upper byte (8 bits) of the device storing the final character.

Error code

Error code	Content
4085H	(s1), (s2) The read address exceeds the device range
4084H	(s1) or (s1+1) parameter setting value is out of range. E.g: 1. The value of (s1) is not in the range of 2-8 2. The value of (s1+1) is not in the range of 0-5 3. The value of (s1+1) is greater than the value of (s1) minus 3 4. When (s1+1) is 0, the number of digits specified in (s1) is less than the number of BIN 16-bit data specified in (s2)+1. When (s1+1) is not 0, the number of digits specified in (s1) is less than the number of BIN 16-bit data specified in (s2) + 2. (The number of digits of (s1) <the number of BIN 16-bit data that does not contain a sign of (s2) + the number of signs (+ or -) + the number of decimal points (.))
4086H	(d) When using offset, the offset address exceeds the device range

Example



After M8 is turned ON, according to the setting of all digits, 6 decimal places and 0 digits, it is converted into a character string "12345" (with a space before 1)

.

R10	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	1
R11	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	23
R12	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	45

DSTR/BIN 32-bit data → string conversion

DSTR(P)

The BIN 32-bit data specified in (s2) is converted into a character string after a decimal point is added to the position specified in (s1), and stored in the device number specified in (d) or later.

-[DSTR (s1) (s2) (d)]

Content, range and data type

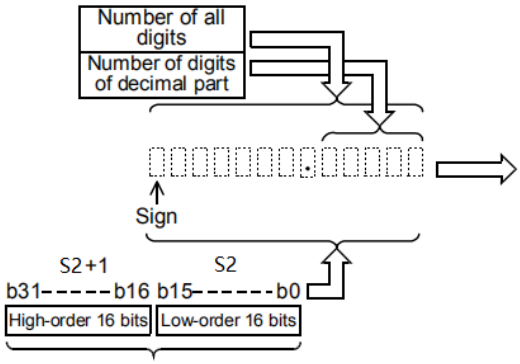
Parameter	Content	Range	Data type	Data type (label)
(s1)	The start number of the device that stores the number of digits of the converted value	-	Signed BIN 16 bit	ANY16_S_ARRAY
(s2)	Converted BIN data	-2147483648 to 2147483647	Signed BIN 32 bit	ANY32_S
(d)	Start number of the device storing the converted character string	-	String	ANYSTRING_SINGLE

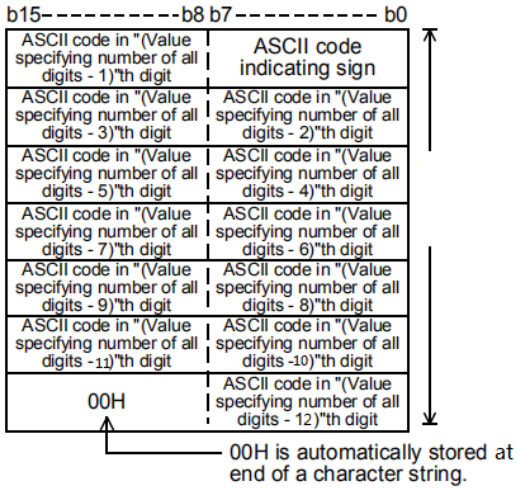
Device used

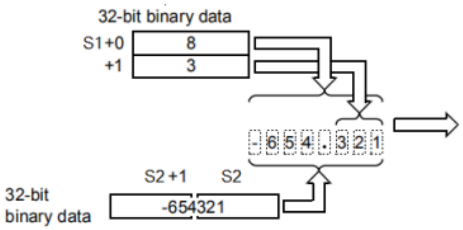
Instruction	Parameter	Device													Offset modification	Pulse extension
		KnX	KnY	KnM	KnS	T	C	D	R	SD	LC	HSC	K	H	[D]	XXP
DSTR	Parameter 1	●	●	●	●	●	●	●	●	●					●	●
	Parameter 2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	Parameter 3					●	●	●	●	●					●	●

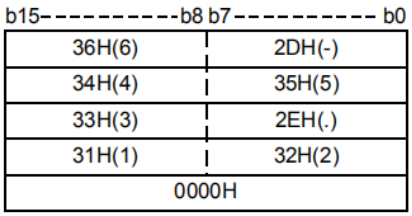
Features

The BIN 32-bit data specified in (s2) is converted into a character string after a decimal point is added to the position specified in (s1), and stored in the device number specified in (d) or later. -654.321 is specified in S2.









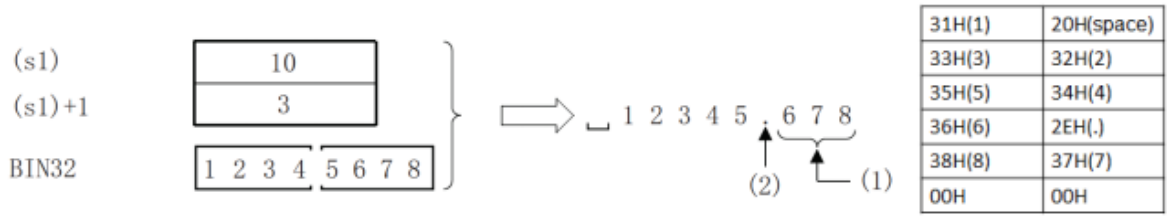
All digits that can be specified in (s1) are 2 to 13 digits.

The number of decimal places that can be specified in (s1)+1 is 0 to 10 digits. However, the setting should satisfy the condition that the number of decimal places ≤ (all digits-3).

The converted character string data will be stored in the device numbers after (d) as follows.

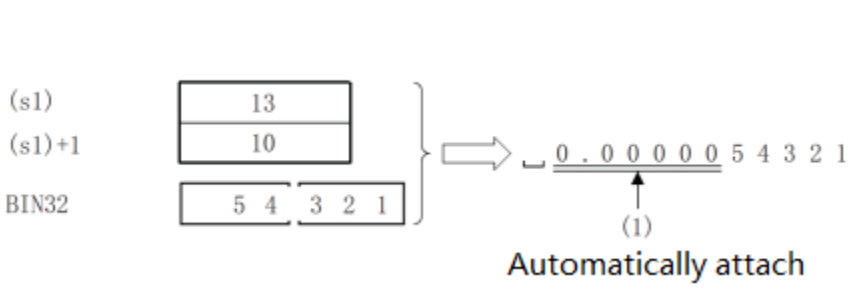
• In the sign, when the BIN 32-bit data is positive, 20H (blank) is stored, and when it is negative, 2DH (—) is stored.

• When the number of decimal places is set to other than 0, 2EH(.) is automatically stored in the specified digit + 1 digit. When the decimal place is 0, 2EH(.) is not stored.



1. : Number of decimal places
2. : Automatically attach

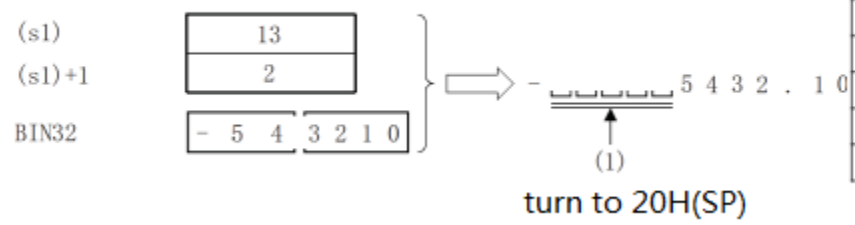
• If the value of the decimal place is greater than the number of digits in the BIN 32-bit data, 0 is automatically added and converted to "0.***" right-justified.



30H(0)	20H(space)
30H(0)	2EH(.)
30H(0)	30H(0)
30H(0)	30H(0)
34H(4)	35H(5)
32H(2)	33H(3)
00H	31H(1)

(1): Automatically attach

• If the sign is excluded from the value of all digits, and the number of digits after the decimal point is greater than the number of BIN 32-bit data, 20H (blank) is stored between the sign and the value. If the digit of BIN 16-bit data is larger, it will be in error status.



20H(space)	2DH(-)
20H(space)	20H(space)
20H(space)	20H(space)
34H(4)	35H(5)
32H(2)	33H(3)
31H(1)	2EH(.)
00H	30H(0)

(1): Change to 20H (SP)

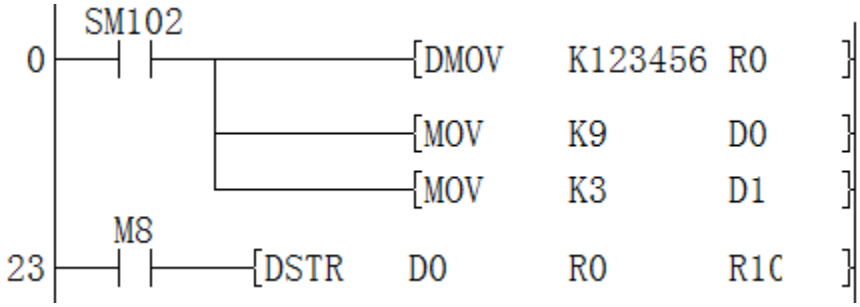
• 00H is automatically stored at the end of the converted character string.

• When the total digits are even digits, "0000H" is stored in the device after the device storing the final character. In the case of an odd number of digits, "00H" is stored in the upper byte (8 bits) of the device storing the final character.

Error code

Error code	Content
4085H	(s1), (s2) The read address exceeds the device range
4084H	(s1) or (s1+1) parameter setting value is out of range. E.g: 1. The value of (s1) is not in the range of 2 to 13. 2. The value of (s1+1) is not in the range of 0 to 10. 3. The value of (s1+1) is greater than the value of (S1) minus 3. 4. When (s1+1) is 0, the number of digits specified in (s1) is less than the number of BIN 16-bit data specified in (s2)+1. When (s1+1) is not 0, the number of digits specified in (s1) is less than the number of BIN 16-bit data specified in (s2) + 2. (The number of digits of (s1) <the number of BIN 16-bit data that does not contain a sign of (s2) + the number of signs (+ or -) + the number of decimal points (.))
4086H	(d) When using offset, the offset address exceeds the device range.

xample



As shown in the example

We need to convert 123456 into a floating point string with 9 lengths after the decimal point and 3 lengths,

The result of the conversion should be 123.456. The previous value will have two spaces to supplement the insufficient number.

R10	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	
R11	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	0	12
R12	1	1	0	0	1	1	0	0	0	1	1	1	0	1	0	0	3.
R13	0	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	45
R14	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	6.

$+/ Combination of strings

$+(P)

Connect the string data stored after the device number specified in (s2) to the string data stored after the device number specified in (s1), and store it after the device number specified in (d).

-[$+ (s1) (s2) (d)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s1)	Connection data or the start number of the device storing the data or a directly specified character string	-	String	ANYSTRING_SINGLE
(s2)	The connected data or the start number of the device storing the connected data or the directly specified character string	-	String	ANYSTRING_SINGLE
(d)	Start number of the device storing the connection result	-	String	ANYSTRING_SINGLE

Device used

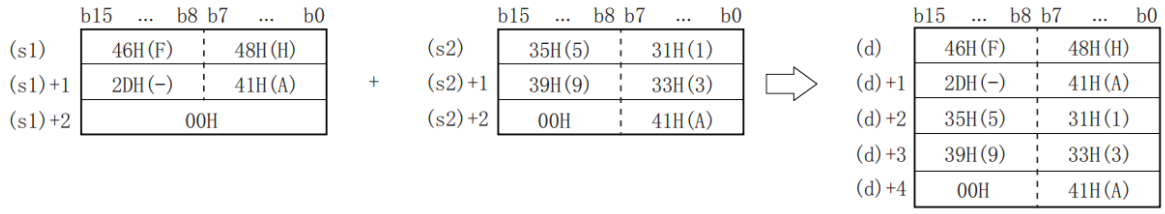
Instruction	Parameter	Device									Offset modification	Pulse extension
		KnX	KnY	KnM	KnS	T	C	D	R	SD	[D]	XXP
$+	Parameter 1	●	●	●	●	●	●	●	●	●	●	●

Parameter 2	●	●	●	●	●	●	●	●	●	●	●
Parameter 3		●	●	●	●	●	●	●	●	●	●

Features

Connect the string data stored after the device number specified in (s2) to the string data stored after the device number specified in (s1), and store it after the device number specified in (d).

The character strings of (s1) and (s2) start with the specified device number until the device number of 00H is stored.



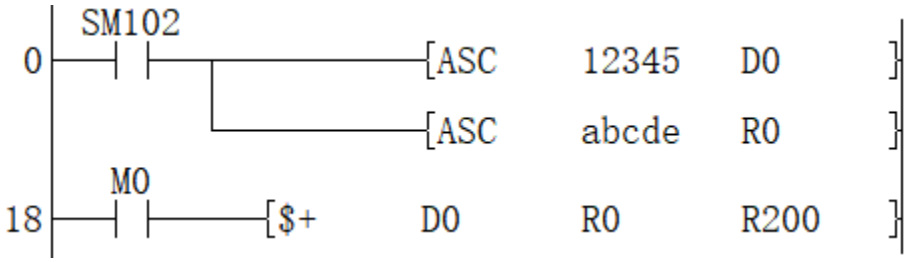
When merging character strings, 00H indicating the end of the character string specified in (s1) is ignored, and the character string specified in (s2) is connected at the final character of (s1).

If the character string is merged, 00H will be automatically appended at the end. If the number of characters after connection is an odd number, 00H is stored in the upper byte of the device that stores the final character, and if the number of characters after connection is an even number, the device after the device that stores the final character is stored 0000H will be stored.

Error code

Error code	Content
4085H	(s1) or (s2) read address out of device range
408AH	(s1) or (s2) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s1) or (s2) When reading a character string, the maximum range of the device is read, but 00H is not found.
4086H	(d) The write address exceeds the device range

Example



The result of combining the string "12345" and the string "abcde" is "12345abcde"

R200	1	0	0	0	1	1	0	0	0	1	0	0	1	1	0	0	12
R201	1	1	0	0	1	1	0	0	0	0	1	0	1	1	0	0	34
R202	1	0	1	0	1	1	0	0	1	0	0	0	0	1	1	0	5a
R203	0	1	0	0	0	1	1	0	1	1	0	0	0	1	1	0	bc
R204	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	0	de

INSTR/string search

INSTR(P)

Starting from the left (s3) character of the string data stored after the device number specified in (s2), search for the string data stored after the device number specified in (s1), and store the search result in In the device specified in (d).

-[INSTR (s1) (s2) (d) (s3)]

Content, range and data type

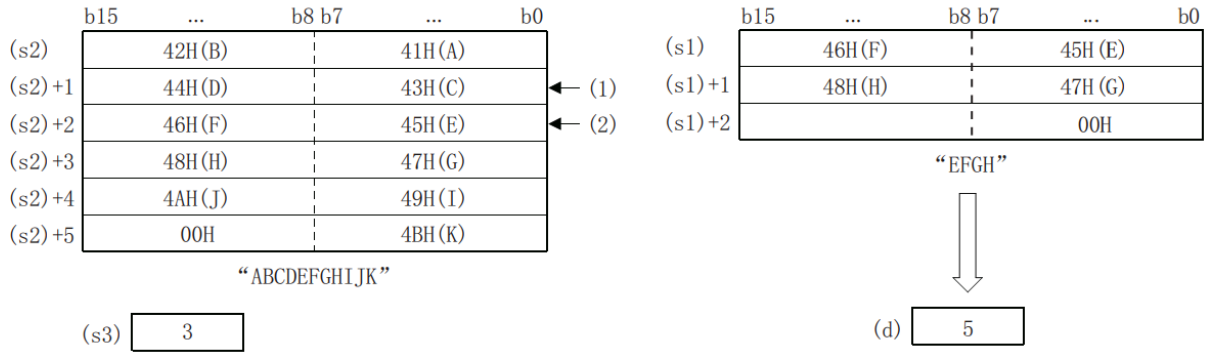
Parameter	Content	Range	Data type	Data type (label)
(s1)	Search string or the start number of the device storing the search string	-	String	ANYSTRING_SINGLE
(s2)	The searched character string or the start number of the device storing the searched character string	-	String	ANYSTRING_SINGLE
(d)	Start number of the device storing the search result	-	Signed BIN 16 bit	ANY16
(s3)	Search start position	1 to 400	Signed BIN 16 bit	ANY16

Device used

Instruction	Parameter	Device											Offset modification	Pulse extension
		KnX	KnY	KnM	KnS	T	C	D	R	SD	K	H	[D]	XXP
INSTR	Parameter 1					●	●	●	●	●			●	●
	Parameter 2					●	●	●	●	●			●	●
	Parameter 3	●	●	●	●	●	●	●	●	●			●	●
	Parameter 4	●	●	●	●	●	●	●	●	●	●	●	●	●

Features

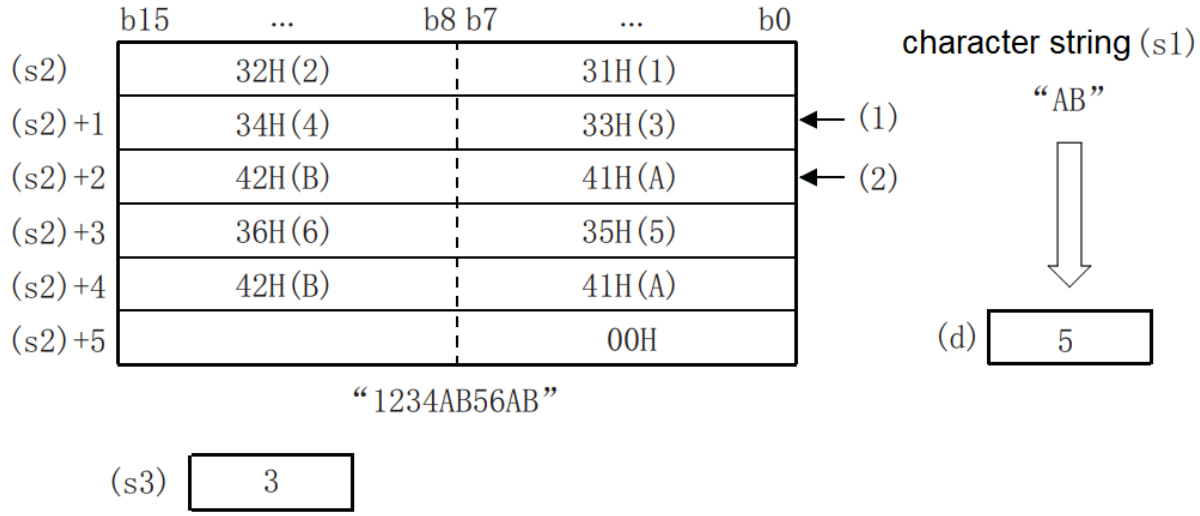
Starting from the left (s3) character of the string data stored after the device number specified in (s2), search for the string data stored after the device number specified in (s1), and store the search result in In the device specified in (d). The search result will store the first character from the start character of the string data specified in (s2).



(1): Search start position (S3): 3rd character

(2): The fifth character from the start character

- If there is no matching character string data, 0 is stored in (d).
- If the search start position (s3) is "0", no processing is performed.
- The searched character string (s1) can be directly specified.



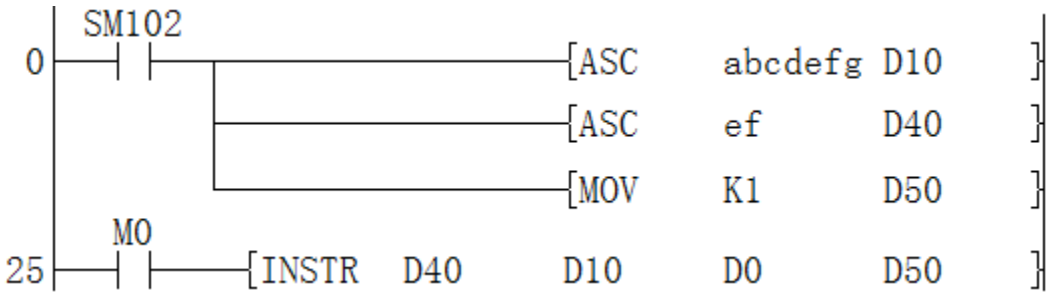
(1): Search start position (s3): 3rd character

(2): The fifth character from the start character

Error code

Error code	Content
4085H	(s1), (s2), (s3) The read address exceeds the device range
408AH	(s1), (s2) The length of the read string exceeds, and the continuous length of the string exceeds 400 characters
408BH	(s1), (s2) When reading a character string, the maximum range of the device is read, but 00H is not found.
4086H	(d) The write address exceeds the device range
4084H	(s3) <0 or (s3)>=string length

Example



Search for the string "ef" in the continuous string "abcdefg" from the first to the fifth position.

ASC/ASCII data input

ASC

A command to convert a character string of half-width/English numbers into ASCII code.

Used to select and display multiple messages on the external display.

-[ASC (s) (d)]

Content, range and data type

Parameter	Content	Range	Data type	Data type (label)
(s)	32-character half-width English numbers input from the computer	-	String (ASCI code only)	ANY_ASC
(d)	Start word device number for storing ASCII data	-	BIN16 bit	ANY16_S

Device used

Instruction	Parameters	Device					Offset modification	Pulse extension
		T	C	D	R	SD	[D]	XXP
ASC	Parameter 1							
	Parameter 2	●	●	●	●	●	●	

Features

1. 16-bit arithmetic (ASC)

After converting the half-width, English, and numeric character strings specified in (S) into ASCII codes, they are transferred to (D) in sequence.

• Process A to Z, 0 to 9, and half-width characters of Signs in (S). (Full-width character strings are not processed.)When programming with a programming tool, enter a character string.

• The converted ASCII code is stored in (D) every 2 characters/1 byte in the order of low 8 bits and high 8 bits.

Extensions

After SM161 is turned ON, the extended function becomes effective. At this time, the half-width/alphanumeric character string specified in S is converted into ASCII code, and then it is transmitted to the lower 8 bits (1 byte) of D in sequence.

#Note:

(1) Number of occupied points of the device

1) When the extended function is OFF

— D occupies the number of characters ÷ 2 points (if not evenly divisible, the decimal point is rounded up.)

2) When the extended function is ON

-The number of points occupied by D is the same as the number occupied by characters.

(2) When using etc.

The extended function flag SM161 is a flag bit common to other instructions.

When using the above instructions and ASC instructions, please note that the SM161 ON or OFF program is written before the ASC instruction so as not to affect it.

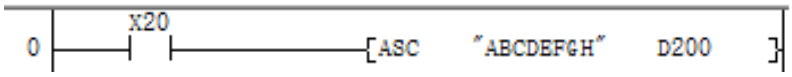
Error code

Error code	Content
4085H	The output result of reading application instruction(s) exceeds the device range

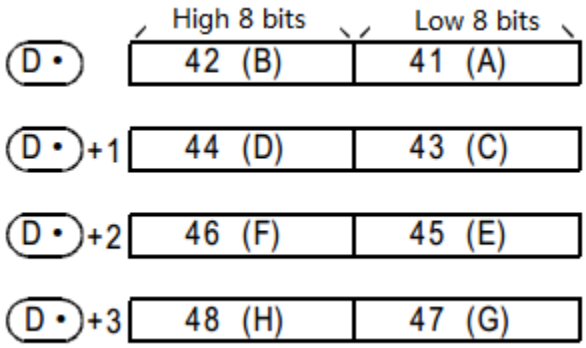
4086H	(D) The output result exceeds the device range in writing application instructions

Example

1. Procedure



When X20 = ON, the assignment of D200 to D203:



If the special register SM161 is set to ON, each ASCII character occupies a 16-bit variable after conversion, as shown in the figure below, the high byte of each variable is filled with 0 (hexadecimal):

	D•		S
	High 8 bits	**Low 8 bits**	**String**
D•	00	41	A
D• +1	00	42	B
D• +2	00	43	C
D• +3	00	44	D
D• +4	00	45	E
D• +5	00	46	F
D• +6	00	47	G
D• +7	00	48	H