

# Converting decimal ASCII to 16-bit binary data

## DABIN(P)(\_U)

FX5S

FX5UJ

FX5U

FX5UC

These instructions convert the decimal ASCII data in the device areas specified by (s) and later to 16-bit binary data, and store the converted data in the device specified by (d).

Ladder diagram		Structured text	
		ENO:=DABIN(EN,s,d); ENO:=DABINP(EN,s,d);	ENO:=DABIN_U(EN,s,d); ENO:=DABINP_U(EN,s,d);
FBD/LD			

## Setting data

### ■Descriptions, ranges, and data types

Operand	Description	Range	Data type	Data type (label)
(s)	ASCII data or the head device where the ASCII data is stored	—	Character string	ANYSTRING_SINGLE
(d)	DABIN(P)	—	16-bit signed binary	ANY16_S
	DABIN(P)_U		16-bit unsigned binary	ANY16_U
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	\$	
(s)	—	○*1	—	—	—	—	○	—	—	○	—
(d)	○	○	○	○	—	—	○	—	—	—	—

\*1 T, ST, and C cannot be used.

## Processing details

- These instructions convert the decimal ASCII data in the device areas specified by (s) and later to 16-bit binary data, and store the converted data in the device specified by (d).
- The setting method of the decimal ASCII data to be set in (s) depends on the status of SM705 (Number of conversion digits selection).

Status of SM705*1	Setting method of (s)	Reference
OFF	Set (s) with a fixed number of digits (a sign + 5 digits in the numeric part).	Page 353 Setting method of (s) for when SM705 (Number of conversion digits selection) is off
ON	Set (s) with a desired number of digits (maximum: a sign + 5 digits in the numeric part).	Page 354 Setting method of (s) for when SM705 (Number of conversion digits selection) is on

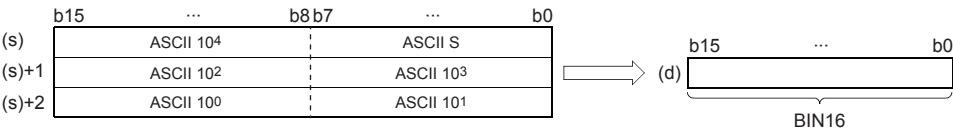
\*1 For the firmware version of the CPU module supporting SM705, refer to the following.

MELSEC iQ-F FX5 User's Manual (Application)

A CPU module which does not support SM705 operates in the same way as SM705 is off even if it is turned on.

■Setting method of (s) for when SM705 (Number of conversion digits selection) is off

Set decimal ASCII data with the fixed number of digits in (s) to (s)+2.

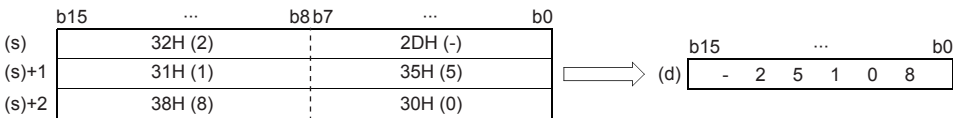


ASCII S: ASCII code for sign  
 ASCII 10<sup>0</sup>: ASCII code for ones place  
 ASCII 10<sup>1</sup>: ASCII code for the tens place  
 ASCII 10<sup>2</sup>: ASCII code for the hundreds place  
 ASCII 10<sup>3</sup>: ASCII code for the thousands place  
 ASCII 10<sup>4</sup>: ASCII code for the ten-thousands place

- The ASCII data in the device specified by (s) to (s)+2 is within the range from -32768 to 32767 when signed data is specified, and it is within the range from 0 to 65535 when unsigned data is specified.
- The data of (s)+3 or later is ignored.
- As sign data, set 20H (space) when the ASCII data is positive, and set 2DH (-) when the data is negative. (If a value other than 20H and 2DH is set, the data will be processed as positive data.)
- A value from 30H to 39H can be set in each place of ASCII code.
- If a value 20H or 00H is set in each place of ASCII code, the value will be processed as 30H.

**Ex.**

When the ASCII data, -25108 (signed), is specified by (s)



## ■Setting method of (s) for when SM705 (Number of conversion digits selection) is on

Set decimal ASCII data with a desired number of digits (including 00H (NULL code)) in (s). Note that 00H (NULL code) is not required to be set if the integral part has the maximum number of digits (5 digits).

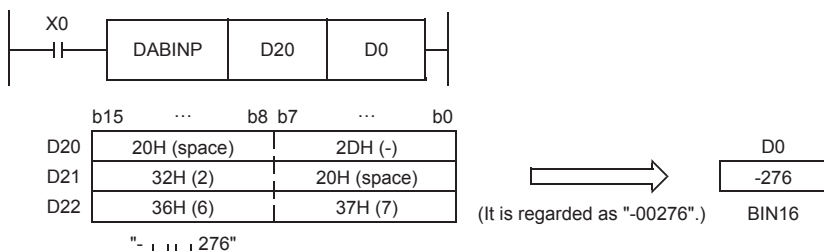
The following table lists the setting method of (s).

Value to be set in (s)	Data of (s) to (s)+2	Value to be set in (s)	Data of (s) to (s)+2																																								
<ul style="list-style-type: none"><li>• 0</li><li>• Positive value (1 digit in numeric part)</li></ul>	<ul style="list-style-type: none"><li>• Set 00H in the upper byte of (s).</li><li>• The data of (s)+1 or later is ignored.</li></ul> <table><tr><td>b15</td><td>...</td><td>b8 b7</td><td>...</td><td>b0</td></tr><tr><td>(s)</td><td colspan="2">00H</td><td colspan="2">ASCII 10<sup>0</sup></td></tr><tr><td>(s)+1</td><td colspan="4"></td></tr><tr><td>(s)+2</td><td colspan="4"></td></tr></table>	b15	...	b8 b7	...	b0	(s)	00H		ASCII 10 <sup>0</sup>		(s)+1					(s)+2					<ul style="list-style-type: none"><li>• Positive value (2 digits in numeric part)</li><li>• Negative value (1 digit in numeric part)</li></ul>	<ul style="list-style-type: none"><li>• Set 00H in the lower byte of (s)+1.</li><li>• The data of the upper byte of (s)+1 or later is ignored.</li></ul> <table><tr><td>b15</td><td>...</td><td>b8 b7</td><td>...</td><td>b0</td></tr><tr><td>(s)</td><td colspan="2">ASCII 10<sup>0</sup></td><td colspan="2">ASCII 10<sup>1</sup> / 2DH (-)</td></tr><tr><td>(s)+1</td><td colspan="2"></td><td colspan="2">00H</td></tr><tr><td>(s)+2</td><td colspan="4"></td></tr></table>	b15	...	b8 b7	...	b0	(s)	ASCII 10 <sup>0</sup>		ASCII 10 <sup>1</sup> / 2DH (-)		(s)+1			00H		(s)+2				
b15	...	b8 b7	...	b0																																							
(s)	00H		ASCII 10 <sup>0</sup>																																								
(s)+1																																											
(s)+2																																											
b15	...	b8 b7	...	b0																																							
(s)	ASCII 10 <sup>0</sup>		ASCII 10 <sup>1</sup> / 2DH (-)																																								
(s)+1			00H																																								
(s)+2																																											
⋮																																											
<ul style="list-style-type: none"><li>• Negative value (4 digits in numeric part)</li></ul>	<ul style="list-style-type: none"><li>• Set 00H in the upper byte of (s)+2.</li></ul> <table><tr><td>b15</td><td>...</td><td>b8 b7</td><td>...</td><td>b0</td></tr><tr><td>(s)</td><td colspan="2">ASCII 10<sup>3</sup></td><td colspan="2">2DH (-)</td></tr><tr><td>(s)+1</td><td colspan="2">ASCII 10<sup>1</sup></td><td colspan="2">ASCII 10<sup>2</sup></td></tr><tr><td>(s)+2</td><td colspan="2">00H</td><td colspan="2">ASCII 10<sup>0</sup></td></tr></table>	b15	...	b8 b7	...	b0	(s)	ASCII 10 <sup>3</sup>		2DH (-)		(s)+1	ASCII 10 <sup>1</sup>		ASCII 10 <sup>2</sup>		(s)+2	00H		ASCII 10 <sup>0</sup>		<ul style="list-style-type: none"><li>• Positive value (5 digits in numeric part)</li></ul>	<ul style="list-style-type: none"><li>• The data of the upper byte of (s)+2 or later is ignored. Since the number of digits is the maximum, 00H is not required to be set.</li></ul> <table><tr><td>b15</td><td>...</td><td>b8 b7</td><td>...</td><td>b0</td></tr><tr><td>(s)</td><td colspan="2">ASCII 10<sup>3</sup></td><td colspan="2">ASCII 10<sup>4</sup></td></tr><tr><td>(s)+1</td><td colspan="2">ASCII 10<sup>1</sup></td><td colspan="2">ASCII 10<sup>2</sup></td></tr><tr><td>(s)+2</td><td colspan="2"></td><td colspan="2">ASCII 10<sup>0</sup></td></tr></table>	b15	...	b8 b7	...	b0	(s)	ASCII 10 <sup>3</sup>		ASCII 10 <sup>4</sup>		(s)+1	ASCII 10 <sup>1</sup>		ASCII 10 <sup>2</sup>		(s)+2			ASCII 10 <sup>0</sup>	
b15	...	b8 b7	...	b0																																							
(s)	ASCII 10 <sup>3</sup>		2DH (-)																																								
(s)+1	ASCII 10 <sup>1</sup>		ASCII 10 <sup>2</sup>																																								
(s)+2	00H		ASCII 10 <sup>0</sup>																																								
b15	...	b8 b7	...	b0																																							
(s)	ASCII 10 <sup>3</sup>		ASCII 10 <sup>4</sup>																																								
(s)+1	ASCII 10 <sup>1</sup>		ASCII 10 <sup>2</sup>																																								
(s)+2			ASCII 10 <sup>0</sup>																																								
<ul style="list-style-type: none"><li>• Negative value (5 digits in numeric part)</li></ul>	<ul style="list-style-type: none"><li>• The data of (s)+3 or later is ignored. Since the number of digits is the maximum, 00H is not required to be set.</li></ul> <table><tr><td>b15</td><td>...</td><td>b8 b7</td><td>...</td><td>b0</td></tr><tr><td>(s)</td><td colspan="2">ASCII 10<sup>4</sup></td><td colspan="2">2DH (-)</td></tr><tr><td>(s)+1</td><td colspan="2">ASCII 10<sup>2</sup></td><td colspan="2">ASCII 10<sup>3</sup></td></tr><tr><td>(s)+2</td><td colspan="2">ASCII 10<sup>0</sup></td><td colspan="2">ASCII 10<sup>1</sup></td></tr></table>	b15	...	b8 b7	...	b0	(s)	ASCII 10 <sup>4</sup>		2DH (-)		(s)+1	ASCII 10 <sup>2</sup>		ASCII 10 <sup>3</sup>		(s)+2	ASCII 10 <sup>0</sup>		ASCII 10 <sup>1</sup>		ASCII 10 <sup>0</sup> : ASCII code for ones place ASCII 10 <sup>1</sup> : ASCII code for tens place ⋮ ASCII 10 <sup>4</sup> : ASCII code for the ten-thousands place																					
b15	...	b8 b7	...	b0																																							
(s)	ASCII 10 <sup>4</sup>		2DH (-)																																								
(s)+1	ASCII 10 <sup>2</sup>		ASCII 10 <sup>3</sup>																																								
(s)+2	ASCII 10 <sup>0</sup>		ASCII 10 <sup>1</sup>																																								

- The ASCII data in the device specified by (s) to (s)+2 is within the range from -32768 to 32767 when signed data is specified, and it is within the range from 0 to 65535 when unsigned data is specified.
- Set 2DH (-) to lower byte of (s)+0 as sign data when the ASCII data is negative. Set an ASCII code of the uppermost digit instead of setting sign data when the ASCII data is 0 or positive.
- A value from 30H to 39H can be set in each place of ASCII code.
- If the value is positive and the numeric part has 5 digits, the data of the upper byte of (s)+2 or later is ignored. If the value is negative and the numeric part has 5 digits, the data of (s)+3 or later is ignored.
- If a value 20H is set in each place of ASCII code, the value is processed as 30H. If a value 00H is set, the value is processed as the end of the decimal ASCII data.
- In the following cases, 0 is stored.
  - The first character is 00H (NULL).
  - The first character is 2DH (-) and the second character is 00H (NULL).

## Program example

In the program below, the signed decimal ASCII codes in five digits stored in D20 to D22 are converted into 16-bit binary data and stored in D0 when X0 is set to ON.



## Operation error

Error code (SD0/SD8067)	Description
2820H	The device specified by (s) exceeds the corresponding device range.
3401H	The sign data is other than 20H, 2DH.* <sup>1</sup>
	A value specified by (s) to (s)+2 for each place of the ASCII code is other than "30H" to "39H", "20H", and "00H".
	The ASCII data in the device specified by (s) to (s)+2 is out of the valid range (-32768 to +32767) (when a signed data is specified).
	The ASCII data in the device specified by (s) to (s)+2 is out of the valid range (0 to 65535) (when unsigned data is specified).

\*1 This error occurs when SM705 is on. When SM705 is off, sign data other than "2DH (-)" is ignored.