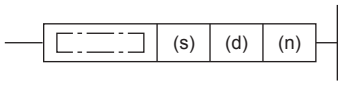
CRC calculation

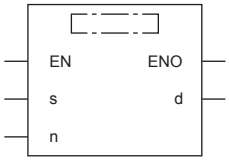
CRC(P)

FX5S FX5UJ FX5U FX5UC

These instructions calculate the CRC (cyclic redundancy check) value which is an error check method used in communication. In addition to CRC value, parity check and sum check are available. For obtaining the horizontal parity value and sum check value, the CCD(P) instruction is available. For the generation of CRC value (CRC-16), these instructions use "$X^{16} + X^{15} + X^2 + 1$" in a polynomial.

Ladder diagram	Structured text
	ENO:=CRC(EN,s,n,d); ENO:=CRCP(EN,s,n,d);

FBD/LD



Setting data

■Descriptions, ranges, and data types

Operand	Description	Range	Data type	Data type (label)
(s)	Head device number storing data for which the CRC value is generated	—	16-bit signed binary	ANY16
(d)	Device number storing the generated CRC value	—	16-bit signed binary	ANY16
(n)	Number of 8-bit (1-byte) data for which the CRC value is generated or the device number storing the number of data	1 to 32767	16-bit unsigned binary	ANY16_U
EN	Execution condition	—	Bit	BOOL
ENO	Execution result	—	Bit	BOOL

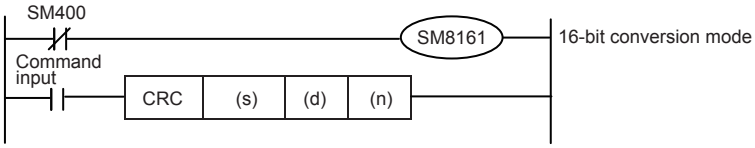
■Applicable devices

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)	○	○	○	—	—	—	○	—	—	—	—
(d)	○	○	○	—	—	—	○	—	—	—	—
(n)	○	○	○	○	—	—	○	○	—	—	—

Processing details

- These instructions generate CRC value for (n) 8-bit data (unit: byte) starting from a device specified in (s), and store to (d). The 16-bit conversion mode and 8-bit conversion mode are available for these instructions. For the operation in each mode, refer to the proceeding pages.
  - 16-bit conversion mode (while SM8161 is OFF)

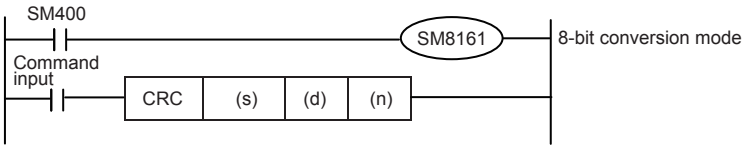
    In this mode, the operation is executed for high-order 8 bits (1 byte) and low-order 8 bits (1 byte) of a device specified in (s). The operation result is stored to one 16-bit device specified in (d).
    SM8161 is shared with the ASCI(P), HEXA(P), and CCD(P) instructions. SM8161 must always be off in the 16-bit mode.
    In the following program, conversion is executed as follows:



			Example) (s) = D100, (d) = D0, (n) = 6		
			Device	Contents of target data	
				8 bits	16 bits
Device storing data for which the CRC value is generated	(s)	Low-order byte	Low-order bits of D100	01H	0301H
		High-order byte	High-order bits of D100	03H	
	(s)+1	Low-order byte	Low-order bits of D101	03H	0203H
		High-order byte	High-order bits of D101	02H	
	(s)+2	Low-order byte	Low-order bits of D102	00H	1400H
		High-order byte	High-order bits of D102	14H	
	⋮	⋮	—		
	(s)+(n)/2-1	Low-order byte	—		
		High-order byte			
Device storing the generated CRC value	(d)	Low-order byte	Low-order bits of D0	E4H	41E4H
		High-order byte	High-order bits of D0	41H	

  - 8-bit conversion mode (while SM8161 is ON)

    In this mode, the operation is executed only for low-order 8 bits (low-order 1 byte) of a device specified by (s). With regard to the operation result, low-order 8 bits (1 byte) are stored to a device specified by (d), and high-order 8 bits (1 byte) are stored to a device specified by (d)+1.
    SM8161 is shared with the ASCI(P), HEXA(P), and CCD(P) instructions. SM8161 must always be on in the 8-bit mode.
    In the following program, conversion is executed as follows:



			Example) (s) = D100, (d) = D0, (n) = 6	
			Device	Contents of target data
Device storing data for which the CRC value is generated	(s)	Low-order byte	Low-order bits of D100	01H
	(s)+1	Low-order byte	Low-order bits of D101	03H
	(s)+2	Low-order byte	Low-order bits of D102	03H
	(s)+3	Low-order byte	Low-order bits of D103	02H
	(s)+4	Low-order byte	Low-order bits of D104	00H
	(s)+5	Low-order byte	Low-order bits of D105	14H
	⋮		—	
	(s)+(n)-1	Low-order byte	—	
Device storing the generated CRC value	(d)	Low-order byte	Low-order bits of D0	E4H
	(d)+1	High-order byte	High-order bits of D0	41H

Precautions

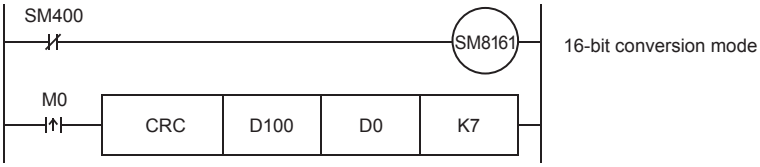
- In these instructions, "$X^{16}+X^{15}+X^2+1$" is used in a polynomial for generating the CRC value (CRC-16). There are many other standard polynomials for generating the CRC value. Note that the CRC value completely differs if an adopted polynomial is different. Major polynomials for generating the CRC value are shown below.

Name	Polynomial
CRC-12	$X^{12} + X^{11} + X^3 + X^2 + X + 1$
CRC-16	$X^{16} + X^{15} + X^2 + 1$
CRC-32	$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
CRC-CCITT	$X^{16} + X^{12} + X^5 + 1$

Program example

In the program example shown below, the CRC value of the data “0123456” stored in D100 to D106 is generated and stored to D0 when M0 turns ON.

- In the case of 16-bit conversion mode



	Contents of data		
	Device	Target data	
Device storing data for which CRC value is generated	D100: 3130H	Low-order byte	30H
		High-order byte	31H
	D101: 3332H	Low-order byte	32H
		High-order byte	33H
	D102: 3534H	Low-order byte	34H
		High-order byte	35H
	D103: 3736H	Low-order byte	36H
		—	—
Device storing generated CRC value	D0: 2ACFH	Low-order byte	CFH
		High-order byte	2AH

Operation error

There is no operation error.