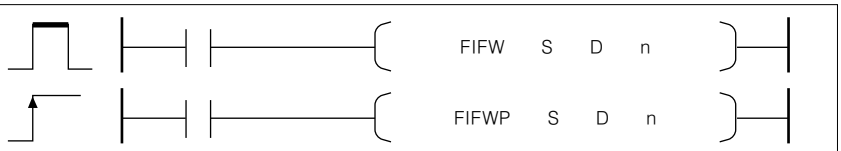
7.5 Data Table Operation Instruction

[7.5.1 Writing Data To The Data Table : FIFW, FIFWP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
FIFW(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D									O	O	O					



S	데이터 테이블에 삽입하고자 하는 데이터가 저장되어 있는 디바이스
D	데이터 테이블의 선두 어드레스

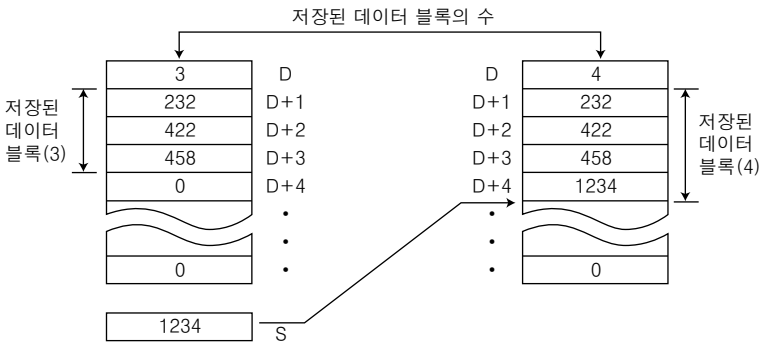
에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET D로 지정된 디바이스의 수가 해당 디바이스의 영역을 초과할 경우 SET

[FIFW]

1) 기능

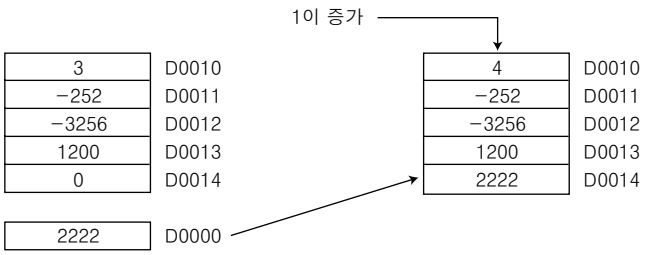
· S로 지정된 디바이스의 워드 데이터를 D로 지정된 데이터 테이블의 하단에 삽입하고 데이터 테이블의 워드 디바이스 수는 1이 증가됩니다.

· D로 지정된 디바이스의 수를 참조하여 데이터 테이블의 워드 디바이스의 수를 파악할 수 있습니다.



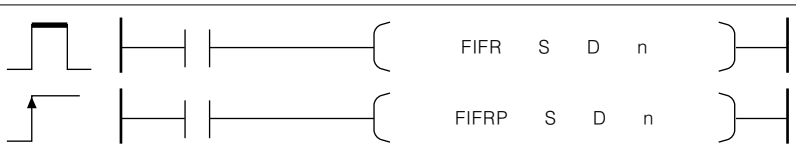
2) 프로그램 예

X0000이 On하면 D0000의 데이터를 D0010의 디바이스가 지정한 개수의 데이터 테이블 워드디바이스의 맨 하단에 삽입되며 테이블 내 워드디바이스 개수는 1이 증가한 상태로 D0010에 저장됩니다.



[7.5.2 Reading Oldest Data From Table : FIFR, FIFRP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
FIFR(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D									O	O	O					



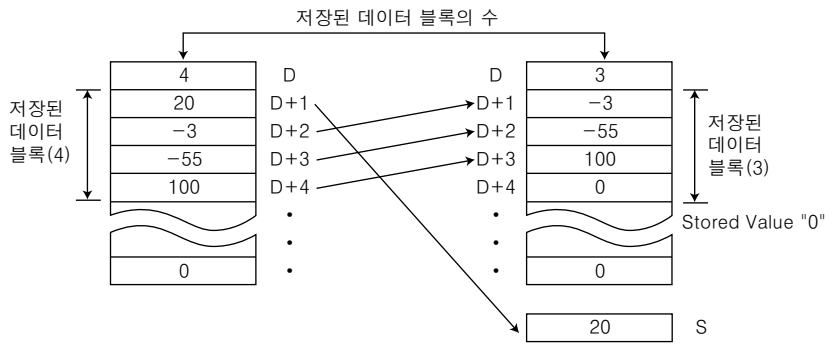
S	데이터 테이블로부터 추출한 데이터를 저장할 디바이스
D	데이터 테이블의 선두 어드레스

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET D로 지정된 디바이스의 수가 해당 디바이스의 영역을 초과할 경우 SET

[FIFR]

1) 기능

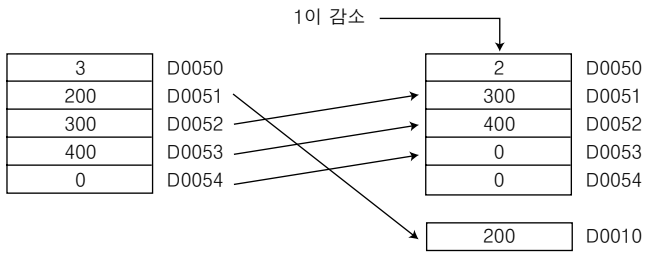
· D로 지정된 데이터 테이블에서 D+1로 지정된 디바이스(Oldest Data From Table)의 데이터를 S로 지정된 디바이스에 저장합니다.
· D로 지정된 디바이스의 수를 참조하여 데이터 테이블의 워드 디바이스의 수를 파악할 수 있습니다.
· D로 지정된 데이터 테이블의 워드 디바이스 수는 1 이 감소합니다.
· 데이터 테이블에서 D+1의 워드 데이터 빠지고 차례대로 하단의 데이터가 Shift되며 삽입됩니다.
· 데이터 테이블에서 맨 하단의 데이터는 0으로 채워집니다.



2) 프로그램 예

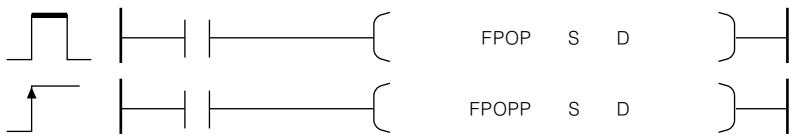
X0010이 On하면 D0050의 디바이스가 지정한 개수의 데이터 테이블에서 D+1로 지정된 워드디바이스D0051이 S로 지정된 워드 디바이스 D0010에 저장되며 차례대로 데이터가 상단으로 Shift되어집니다.

그리고 데이터 테이블의 맨 하단 D0053은 0으로 채워지며 테이블 내 워드디바이스 갯수는 1이 감소한 상태로 D0050에 저장됩니다.



[7.5.3 Reading Newest Data From Data Table : FPOP, FPOPP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
FPOP(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D									O	O	O					



S	데이터 테이블로부터 추출한 데이터를 저장할 디바이스
D	데이터 테이블의 선두 어드레스

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET D로 지정된 디바이스의 수가 해당 디바이스의 영역을 초과할 경우 SET

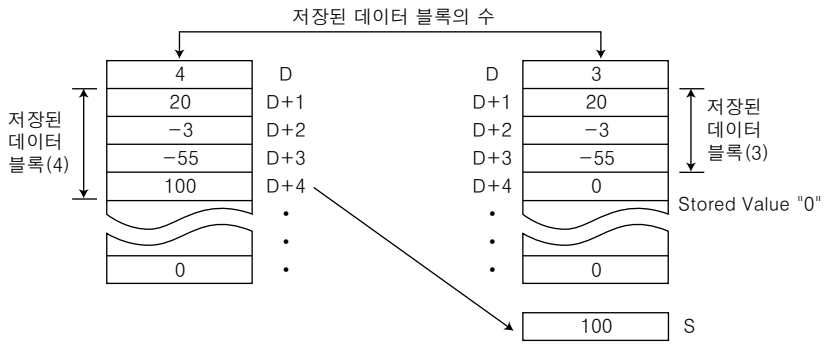
[FIFR]

1) 기능

· D로 지정된 데이터 테이블에서 D마지막으로 입력된 데이터를 읽어 S로 지정된 영역에 저장합니다.

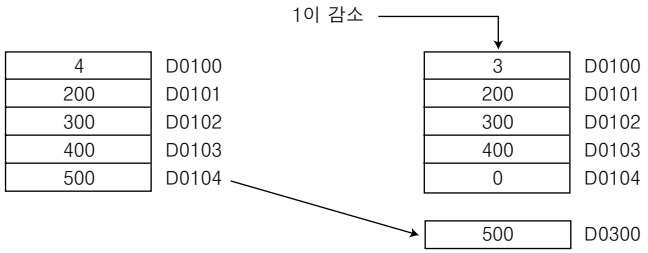
· D로 지정된 디바이스의 수를 참조하여 데이터 테이블의 워드 디바이스의 수를 파악할 수 있습니다.

· 명령이 실행되면 D로 지정된 데이터 테이블의 데이터 수는 1 이 감소하고, 마지막 데이터를 저장했던 영역의 데이터는 0이 됩니다.



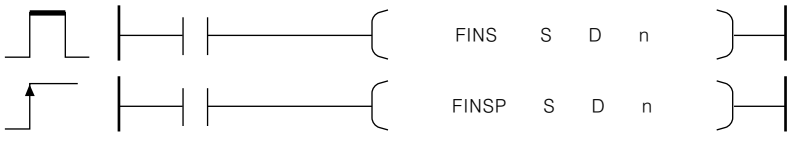
2) 프로그램 예

X00100이 On 하였을 때 D0100으로 지정된 데이터 테이블의 마지막 데이터 D0104의 값 500을 D300에 저장하는 프로그램



[7.5.4 Insert Data In Data Table : FINS, FINSP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
FINS(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	D									O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



S	데이터 테이블로부터 추출한 데이터를 저장할 디바이스
D	데이터 테이블의 선두 어드레스
n	삽입하는 데이터 테이블의 위치

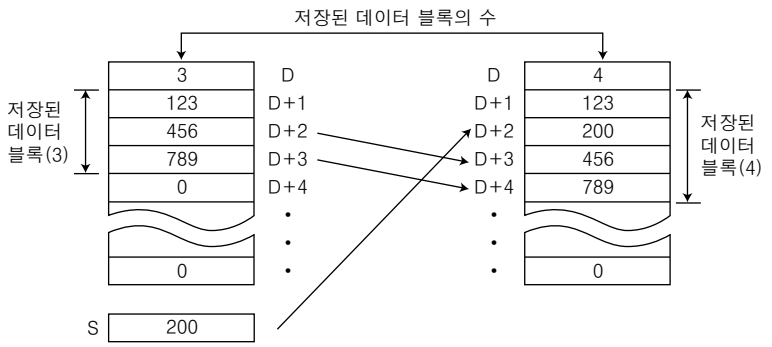
에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET D로 지정된 디바이스의 수가 해당 디바이스의 영역을 초과할 경우 SET

[FINS]

1) 기능

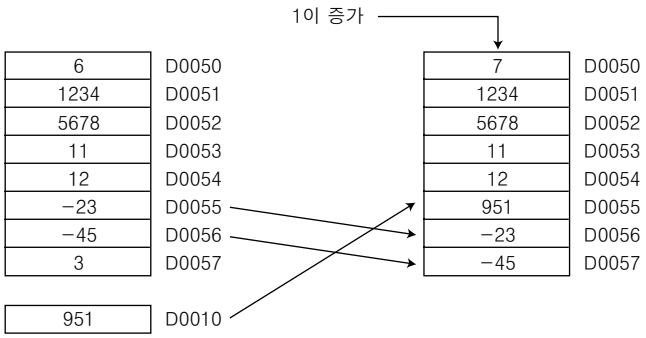
· S로 지정된 데이터를 D로 지정된 데이터 테이블의 n번째 위치(D+n)에 삽입하여 저장합니다.

· 명령이 실행되면, 데이터 테이블의 D+n 이후 데이터들은 1워드씩 증가된 위치로 Shift 되며, 데이터 테이블의 블록수는 1 증가합니다.



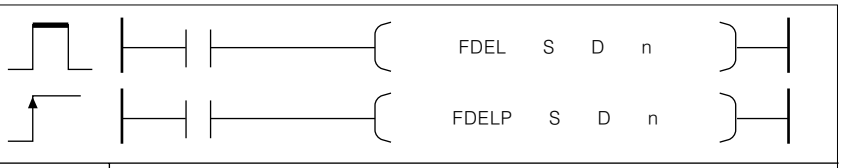
2) 프로그램 예

· X0010이 On 되었을 때, D0050으로 지정된 값 951을 D0050으로 지정된 데이터 테이블의 5번째 블록(D0055)에 삽입하는 프로그램



[7.5.5 Delete Data In Data Table : FDEL, FDELP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
FDEL(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	D									O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



S	데이터 테이블에서 삭제한 데이터를 저장할 영역의 어드레스
D	데이터 테이블의 선두 어드레스
n	삽입하는 데이터 테이블의 위치

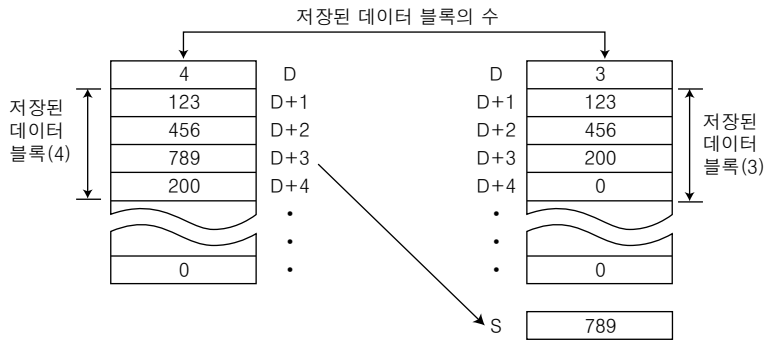
에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET D로 지정된 디바이스의 수가 해당 디바이스의 영역을 초과할 경우 SET

[FDEL]

1) 기능

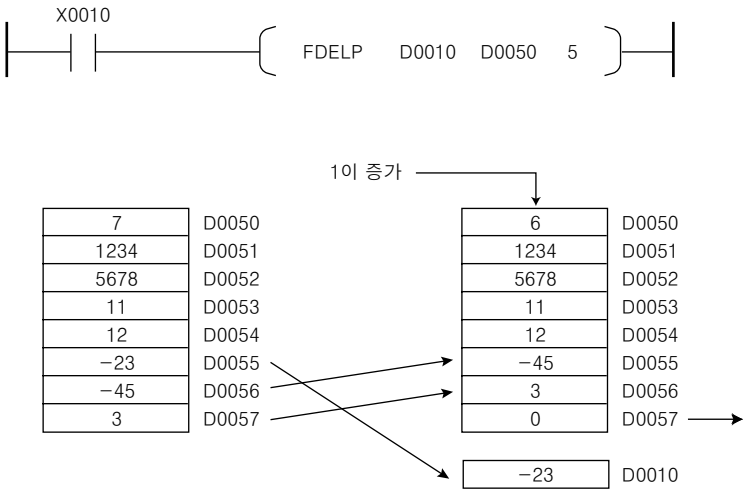
· D로 지정된 데이터 테이블의 n번째 블록(D+n)에 데이터를 삭제하고 삭제된 데이터를 S로 지정된 영역에 저장합니다.

· 명령이 실행되면, 데이터 테이블의 D+n 이후 데이터들은 1워드씩 감소된 위치로 Shift 되며, 데이터 테이블의 블록수는 1 감소합니다.



2) 프로그램 예

· X0010이 On 되었을 때, D0100으로 지정된 데이터 테이블의 5번째 블록(D0055)의 데이터 −23을 삭제하고 D0010으로 지정된 영역에 저장하는 프로그램

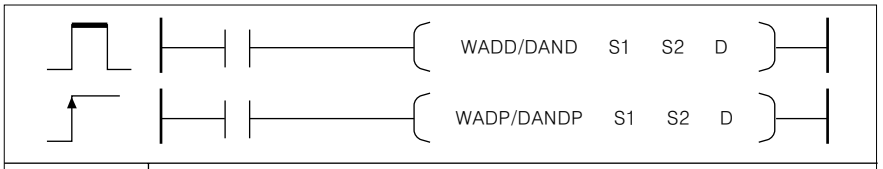


7.6 Logic operation instruction

Total Solution for Industrial Automation

[7.6.1 워드 And : WAND, WANDP, DAND, DANDP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
WAND(P) DAND(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O*		O	O	O	O	O	O				



S1, S2	지정된 데이터 또는 영역의 번호
D	Destinasion 영역의 번호

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[WAND]

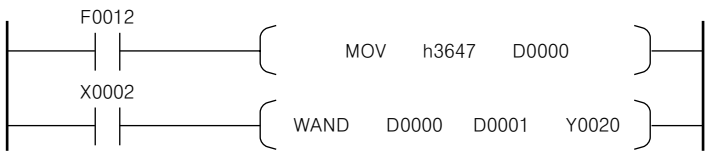
1) 기능

· S1으로 지정된 영역의 각 비트 데이터와 S2로 지정 된 영역의 각 비트 데이터를 AND 결합하여 D로 지정된 영역의 각 비트에 저장합니다.

· WAND(WANDP) 명령어는 16Bit, DAND(DANDP) 명령어는 32Bit의 데이터를 논리 연산 처리합니다 .

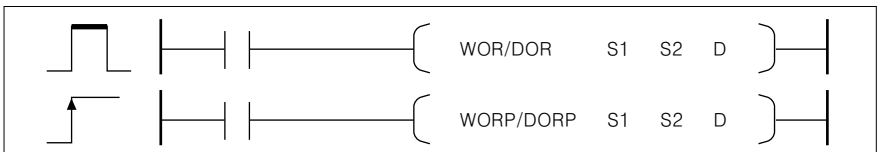
2) 프로그램

· 입력신호 D0000 와 D0001 의 데이터를 WAND 하여 Y0020 에 출력하는 프로그램



[7.6.2 워드 OR : WOR, WORP, DOR, DORP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
WOR(P) DOR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O*		O	O		O	O					



S1, S2	지정된 데이터 또는 영역의 번호
D	Destinasion 영역의 번호

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[WOR]

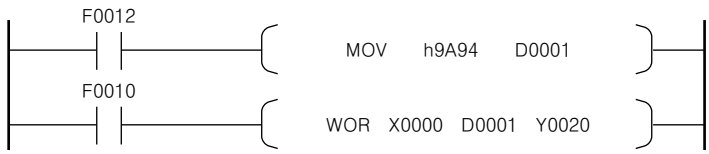
1) 기능

· S1으로 지정된 영역의 각 비트 데이터와 S2로 지정 된 영역의 각 비트 데이터를 OR 결합하여 D로 지정된 영역의 각 비트에 저장합니다.

· WOR(WORP) 명령어는 16Bit, DOR(DORP) 명령어는 32Bit 의 데이터를 논리 연산 처리합니다.

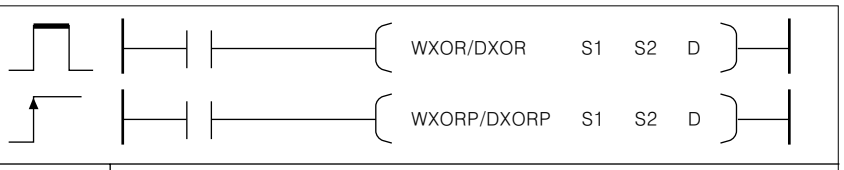
2) 프로그램 예

· 입력신호 X0000 와 D0001 의 데이터를 WOR 결합하여 Y0020에 출력하는 프로그램



[7.6.3 워드 Exclusive OR : WXOR, WXORP, DXOR, DXORP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
WXOR(P) DXOR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O*		O	O		O	O					



S1, S2	지정된 데이터 또는 영역의 번호
D	Destinasion 영역의 번호

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[WXOR]

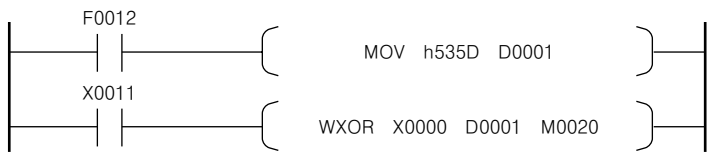
1) 기능

· S1으로 지정된 영역의 각 비트 데이터와 S2로 지정된 영역의 각 비트 데이터를 XOR 결합하여 D로 지정된 영역의 각 비트에 저장합니다

· WXOR(WXORP) 명령어는 16Bit, DXOR(DXORP) 명령어는 32Bit 의 데이터를 논리 연산처리 합니다.

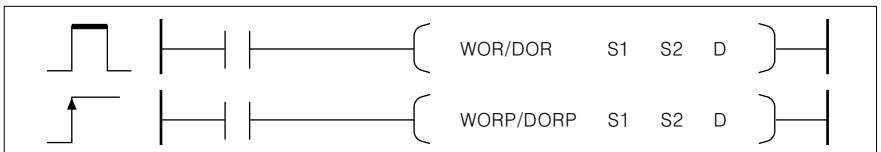
2) 프로그램

· 입력신호 X0000 와 D0001 의 데이터를 WXOR 결합하여 M0020 에 저장하는 프로그램



[7.6.4 워드 Exclusive NOR : WXNR, WXNRP, DXNR, DXNRP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
WXNR(P) DXNR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O*		O	O		O	O					



S1, S2	지정된 데이터 또는 영역의 번호
D	Destinasion 영역의 번호

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[WXNR]

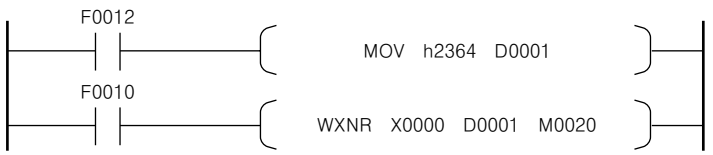
1) 기능

· S1으로 지정된 영역의 각 비트 데이터와 S2로 지정된 영역의 각 비트 데이터를 Exclusive NOR를 실행하여 D로 지정한 영역의 각 비트에 저장합니다.

· WXNR(WXNRP) 명령어는 16Bit, DXNR(DXNRP) 명령어는 32Bit의 데이터를 논리연산 처리합니다.

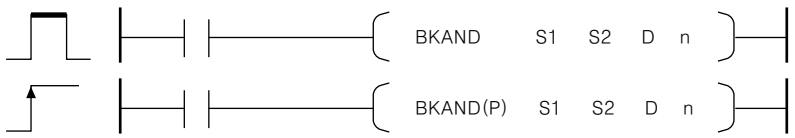
2) 프로그램 예

· 입력신호 X0000 와 D0001 의 데이터를 Exclusive NOR 결합하여 Y0020 에 출력하는 프로그램



[7.6.5 Block logical product : BKAND, BKANDP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BKAND(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	5	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



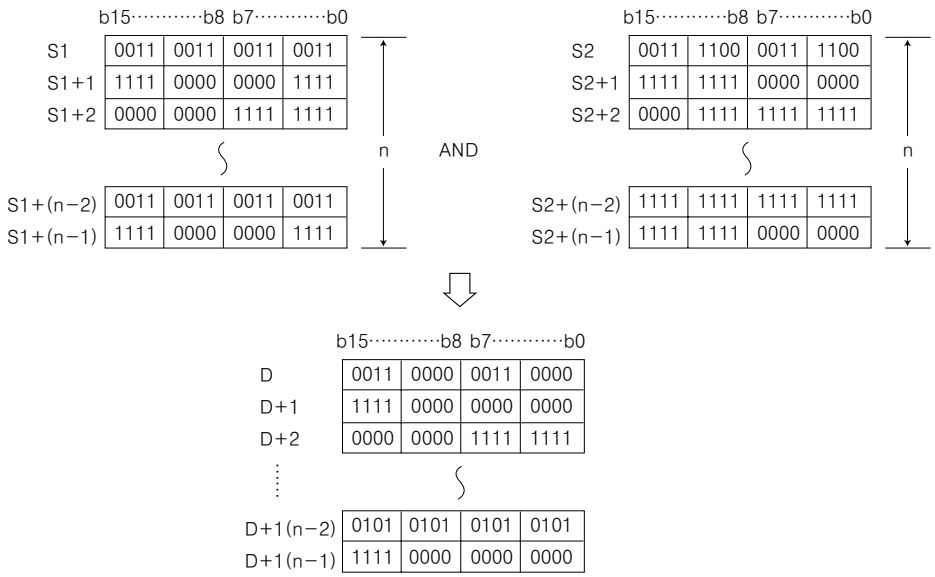
S1, S2	AND를 할 데이터가 저장되어있는 디바이스의 선두 어드레스
D	AND 결과를 저장하는 Device의 선두 어드레스
n	연산할 데이터의 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[BKAND]

1) 기능

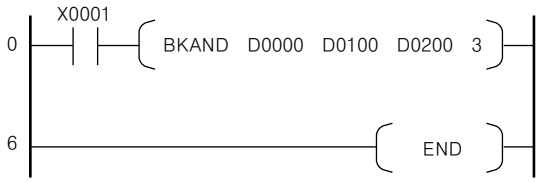
· S1으로 지정된 N크기의 Device와 S2로 지정된 n크기의 Device를 AND 하여 D로 지정된 Device에 저장합니다.



2) Program 예

· X0001이 ON되었을 때 D0000에서 3 만큼의 디바이스와 D0100에서 3 만큼의 디바이스를 AND하여 D0200에 저장하는 프로그램

[Ladder Mode]



[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	BKANDP	D0000
		D0100
		D0200
	3	
6	END	

	b15·····b8	b7·····b0		
D0000	1111	0000	1111	1111
D0001	1111	1111	0000	0000
D0002	0000	0000	1111	1111

AND

	b15·····b8	b7·····b0		
D0100	1111	1111	0000	0000
D0101	0000	1111	0000	0000
D0102	0000	1111	1111	1111

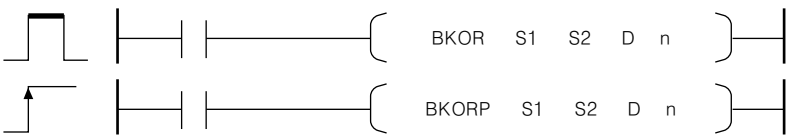
⇩

	b15·····b8	b7·····b0		
D0200	1111	0000	0011	0000
D0201	0000	1111	0000	0000
D0202	0000	0000	1111	1111

n 3

[7.6.6 Block logical sum operations : BKOR, BKORP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BKOR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	5	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				

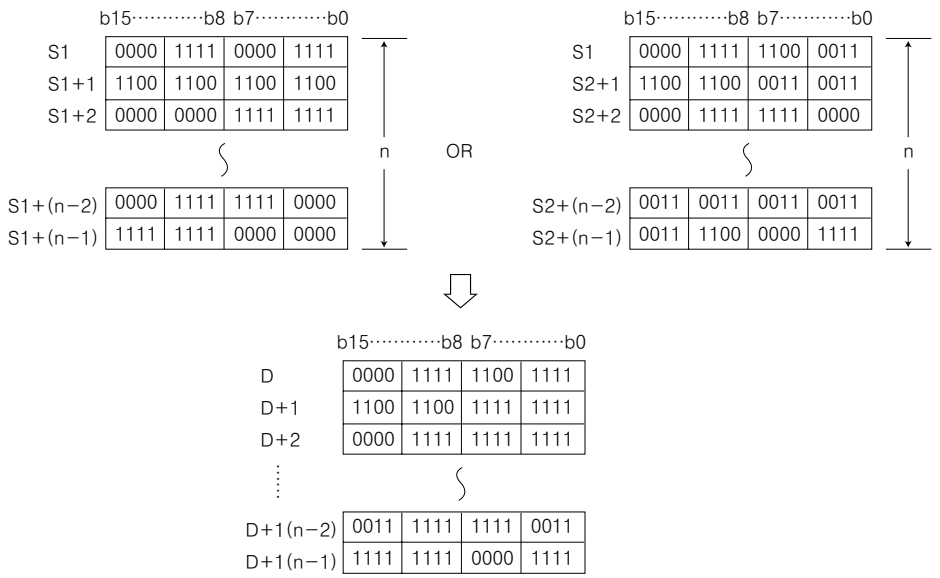


S1, S2	OR를 데이터가 저장되어있는 디바이스의 선두 어드레스
D	OR 결과를 저장하는 Device의 선두 어드레스
n	연산할 데이터의 수

[BKOR]

1) 기능

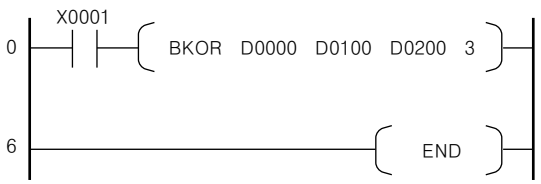
· S1으로 지정된 n크기의 Device와 S2로 지정된 n크기의 Device를 OR하여 D로 지정된 Device에 저장합니다.



2) Program 예

· X0001이 ON되었을 때 D0000에서 3 만큼의 디바이스와 D0100에서 3 만큼의 디바이스를 OR하여 D0200에 저장하는 프로그램

[Ladder Mode]



[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	BKANDP	D0000
		D0100
		D0200
		3
6	END	

	b15·····b8	b7·····b0		
D0000	0000	1010	1010	1010
D0001	0011	0011	0011	0011
D0002	1100	0000	0011	1100

OR

	b15·····b8	b7·····b0		
D0100	0011	0000	0000	0000
D0101	0000	1111	1111	0000
D0102	0000	1111	0000	1111

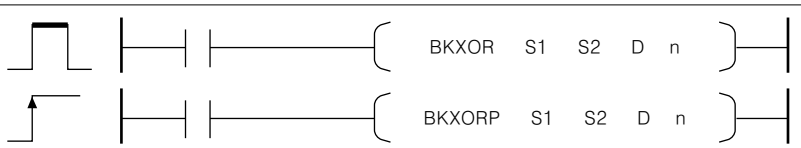
⇩

	b15·····b8	b7·····b0		
D0200	0011	1010	1010	1010
D0201	0011	1111	1111	0011
D0202	1100	1111	0011	1111

n 3

[7.6.7 Block exclusive OR operations : BKXOR, BKXORP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BKXNR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	5	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



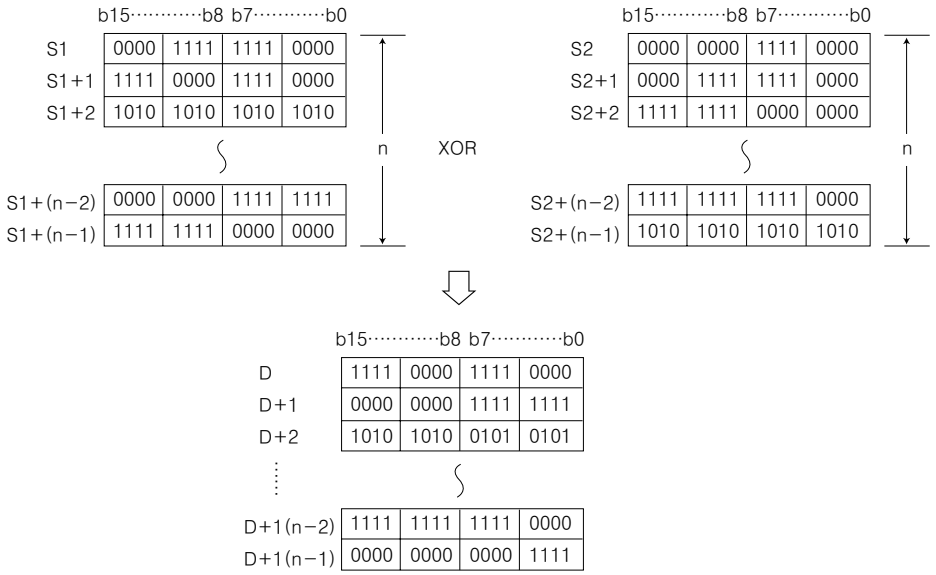
S1, S2	XOR를 데이터가 저장되어있는 디바이스의 선두 어드레스
D	XOR 결과를 저장하는 Device의 선두 어드레스
n	연산할 데이터의 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[BKXOR]

1) 기능

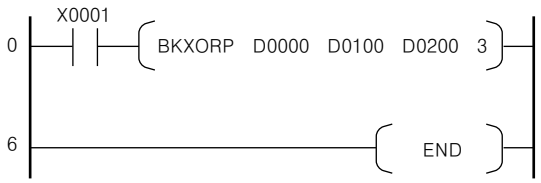
· S1으로 지정된 n크기의 Device와 S2로 지정된 n크기의 Device를 XOR 하여 D로 지정된 Device에 저장합니다.



2) 프로그램 예

· X0001이 ON되었을 때 D0000에서 3 만큼의 디바이스와 D0100에서 3 만큼의 디바이스를 XOR하여 D0200에 저장하는 프로그램

[Ladder Mode]



[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	BKXNRP	D0000
		D0100
		D0200
		3
6	END	

	b15·····b8	b7·····b0		
D0000	1010	1010	1010	1010
D0001	0101	0101	0101	0101
D0002	0000	0000	1111	1111

XOR

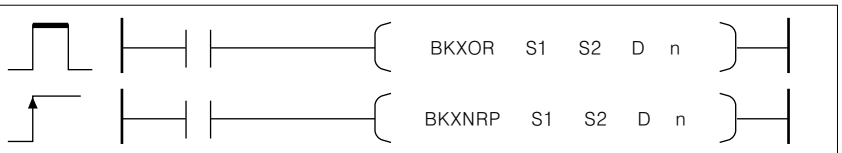
	b15·····b8	b7·····b0		
D0100	1111	1111	0000	0000
D0101	0000	0000	0000	1111
D0102	1010	1010	1010	1010

	b15·····b8	b7·····b0		
D0200	1010	1010	0101	0101
D0201	1010	1010	1010	0101
D0202	0101	0101	1010	1010

n 3

[7.6.8 Block non-exclusive logical sum operations : BKXNR, BKXNRP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BKXNR(P)	S1	O	O	O	O	O	O	O	O	O	O	O	O	5	O		
	S2	O	O	O	O	O	O	O	O	O	O	O	O				
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



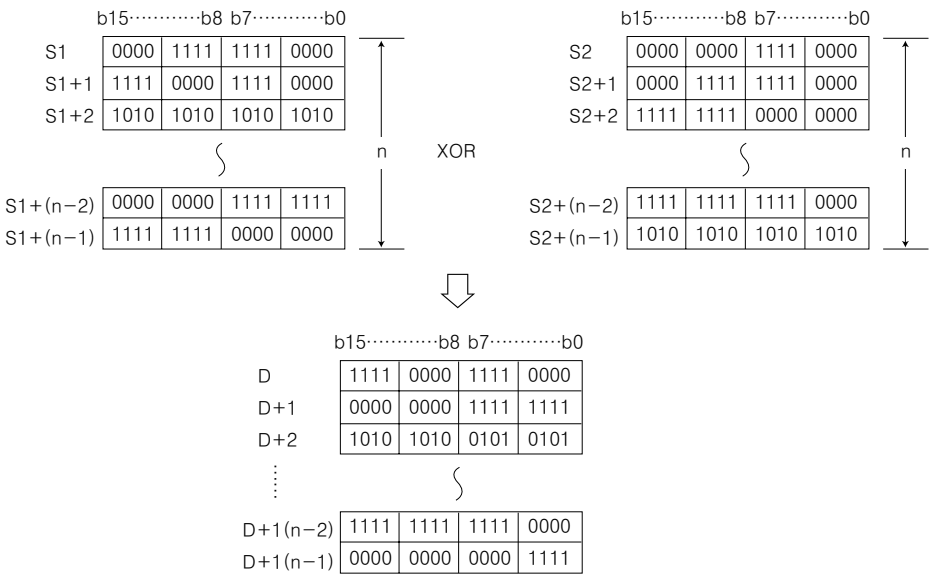
S1, S2	XOR를 데이터가 저장되어있는 디바이스의 선두 어드레스
D	XOR 결과를 저장하는 Device의 선두 어드레스
n	연산할 데이터의 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[BKXNR]

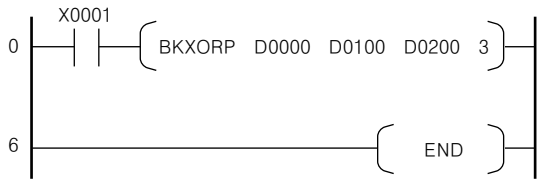
1) 기능

· S1으로 지정된 n크기의 Device와 S2로 지정된 n크기의 Device를 XNOR 하여 D로 지정된 Device에 저장합니다.



2) Program 예

· X0001이 ON되었을 때 D0000에서 3 만큼의 디바이스와 D0100에서 3 만큼의 디바이스를 XNR하여 D0200에 저장하는 프로그램



Steps	Instruction	Device
0	LD	X0001
1	BKXNRP	D0000
		D0100
		D0200
		3
6	END	

b15·····b8 b7·····b0

D0000	1010	1010	1010	1010
D0001	0101	0101	0101	0101
D0002	0000	0000	1111	1111

XNOR

b15·····b8 b7·····b0

D0100	1111	1111	0000	0000
D0101	0000	0000	0000	1111
D0102	1010	1010	1010	1010

b15·····b8 b7·····b0

D0200	1010	1010	0101	0101
D0201	1010	1010	1010	0101
D0202	0101	0101	1010	1010

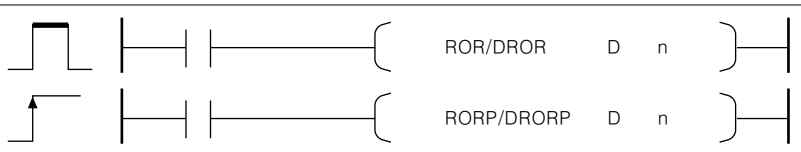
n 3

7.7 Rotation instruction

Total Solution for Industrial Automation

[7.7.1 Roate Rigth : ROR, RORP, DROR, DRORP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
ROR(P) DROR(P)	D	O		O	O	O		O	O	O	O	O		3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



D	우측방향으로 회전시킬 데이터가 저장되어 있는 영역
n	우측 방향으로 회전시킬 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 플래그를 SET하고 해당 명령어는 결과 처리되지 않습니다
캐리(F112)	우측 방향으로 회전중 캐리가 발생하면 캐리 플래그를 SET

[ROR]

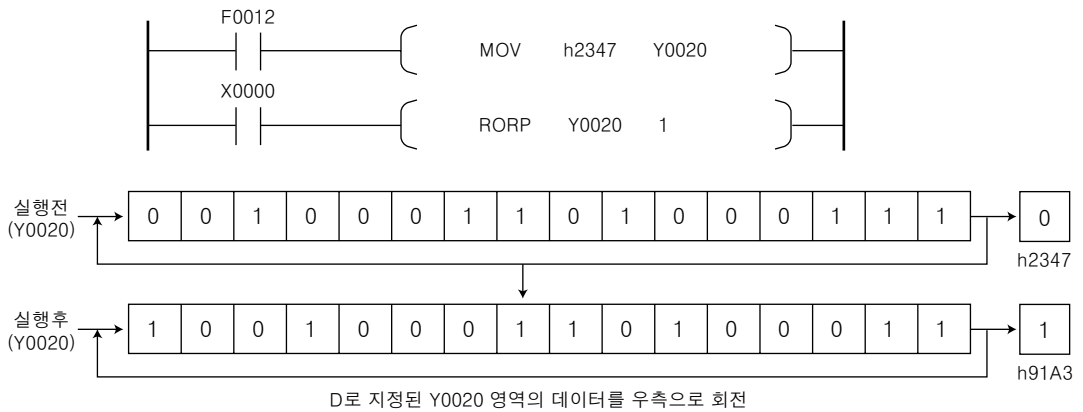
1) 기능

· 1워드 데이터 16개 비트를 n으로 지정된 bit 만큼 우측으로 회전하며 최하위 비트는 캐리 플래그 (F112)와 최상위 비트로 회전합니다. (1워드 내에서 회전)

· ROR(RORP) 명령어는 16Bit, DROR(DRORP) 는 32Bit 데이터를 연산 대상으로 합니다.

2) 프로그램 예

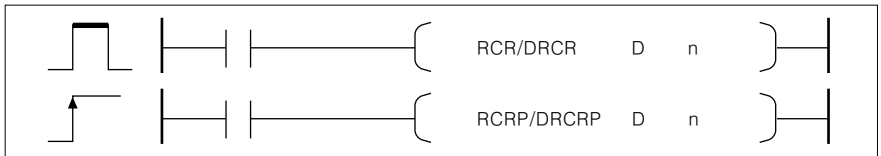
· 입력신호 X00000이 On 하였을 때 Y0020 의 데이터를 1 비트씩 우측으로 회전하며 , 캐리 플래그(F112)를 Set 하는 프로그램



D로 지정된 Y0020 영역의 데이터를 우측으로 회전

[7.7.2 Rotate Right with carry : RCR, RCRP, DRCR, DRCRP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
RCR(P) DRCR(P)	D	O		O	O	O		O	O	O	O	O		3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



D	우측방향으로 회전시킬 데이터가 저장되어 있는 영역
n	우측 방향으로 회전시킬 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 플래그를 SET하고 해당 명령어는 결과 처리되지 않습니다
캐리(F112)	회전중 캐리가 발생하면 캐리 플래그를 SET

[RCR]

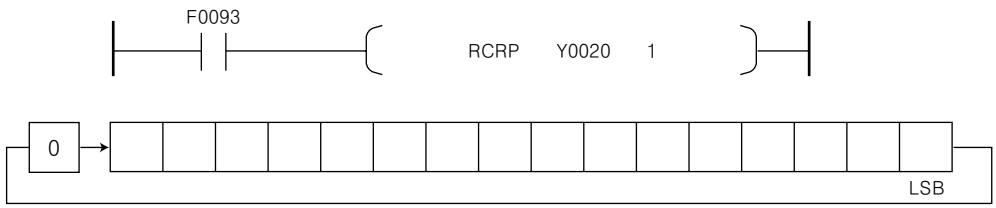
1) 기능

· 1워드 데이터 16개 비트를 n으로 지정된 bit 만큼 우측으로 회전하며 최하위비트는 캐리 플래그 (F112)로, 캐리 플래그 (F112)는 최상위 비트로 회전 (이동)합니다.

· RCR(RCRP) 명령어는 16Bit, DRCR(DRCRP) 명령어는 32Bit 데이터를 연산 대상으로 합니다.

2) 프로그램 예

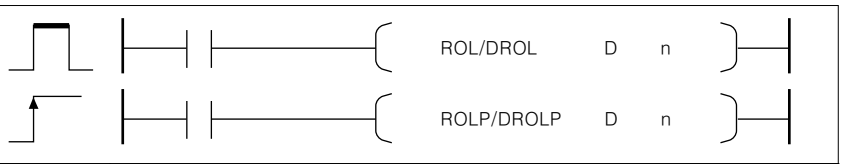
· 입력신호 F0093 의 클럭 주기인 1초마다 Y0020 의 데이터를 캐리 플래그 (F112)를 포함하여 우측방향 회전을 실행합니다.



1초마다 D로 지정된 Y0020영역의 데이터를 캐리 플러그(F112)를 포함하면서 우측으로 회전

[7.7.3 Rotate Left : ROL, ROLP, DROL, DROLP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
ROL(P) DROL(P)	D	O		O	O	O		O	O	O	O	O		3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



D	좌측방향으로 회전시킬 데이터가 저장되어 있는 영역
n	좌측 방향으로 회전시킬 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 플래그를 SET하고 해당 명령어는 결과 처리되지 않습니다
캐리(F112)	좌측 방향으로 회전중 캐리가 발생하면 캐리 플래그를 SET

[ROL]

1) 기능

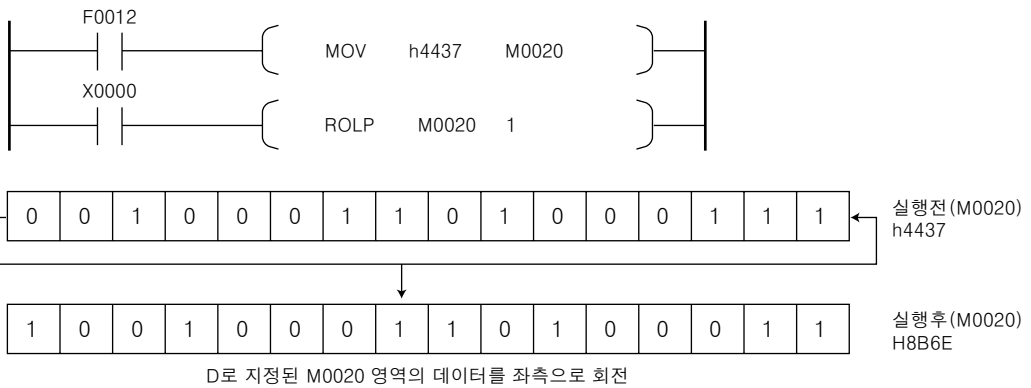
· 1워드 데이터 16개 비트를 n으로 지정된 bit 만큼 좌측으로 회전하며 최상위 비트는 캐리 플래그 (F112)와 최하위 비트로 회전합니다. (1워드 내에서 회전)

· ROL(ROLP) 명령어는 16Bit, DROL(DROLP) 명령어는 32Bit 데이터를 연산 대상으로 합니다.

2) 프로그램 예

· 입력신호 X0000 을 On 할 때마다 M0020 의 16 비트의 상태가 1 비트씩 좌측으로 이동하는

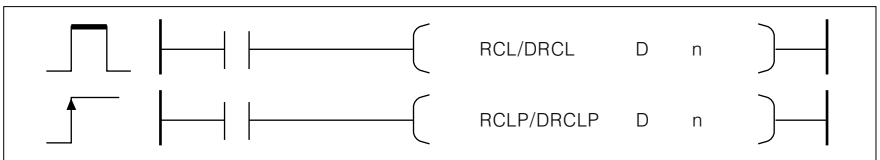
프로그램



D로 지정된 M0020 영역의 데이터를 좌측으로 회전

[7.7.4 Rotate left with carry : RCL, RCLP, DRCL, DRCLP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
RCL(P) DRCL(P)	D	O		O	O	O		O	O	O	O	O		3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



D	좌측방향으로 회전시킬 데이터가 저장되어 있는 영역
n	좌측 방향으로 회전시킬 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET
캐리(F112)	회전중 캐리가 발생하면 캐리 플래그를 SET

[RCL]

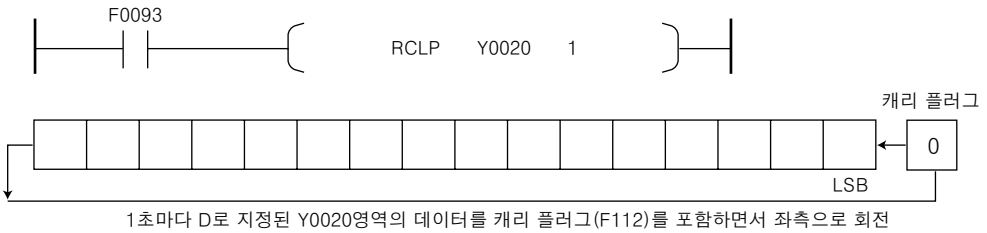
1) 기능

· 1워드 데이터 16개 비트를 n으로 지정된 bit 만큼 좌측으로 회전하며 최상위비트는 캐리 플래그 (F112)로, 캐리 플래그 (F112)는 최하위 비트로 회전(이동)합니다.

· RCL(RCLP) 명령어는 16Bit, DRCL(DRCLP) 명령어는 32Bit 데이터를 연산 대상으로 합니다.

2) 프로그램 예

· 입력신호 F0093 의 주기 클럭인 1초마다 Y0020의 데이터를 캐리 플래그 (F112)를 포함하여 좌측 방향 회전을 실행합니다.



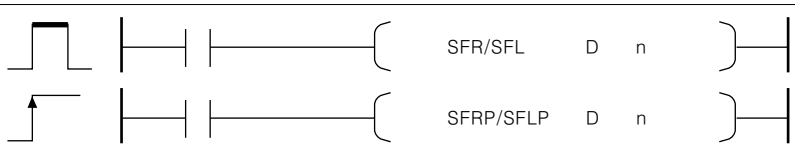
1초마다 D로 지정된 Y0020영역의 데이터를 캐리 플러그(F112)를 포함하면서 좌측으로 회전

7.8 Shift Instruction

Total Solution for Industrial Automation

[7.8.1 16Bit Data n Bit 우 Shift, 좌 Shift : SFR, SFRP, SFL, SFLP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
SFR(P) SFL(P)	D	O		O	O	O		O	O	O	O	O		3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



D	Shift될 Data를 저장하고 있는 device
n	Shift수

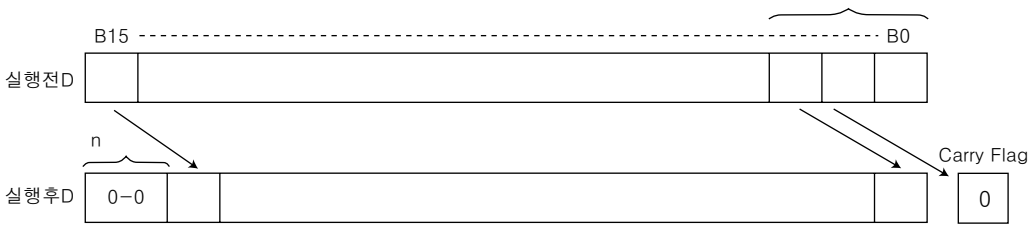
에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET
캐리(F112)	연산결과가 오버플로어면 SET

[SFR]

1) 기능

· D로 지정된 Device의 16비트 데이터를 우로 n 비트 이동합니다.

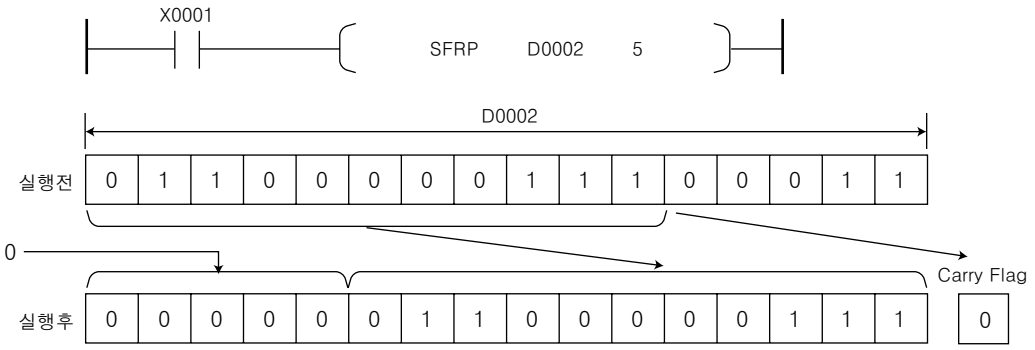
· 최상위부터 n비트분은 0으로 됩니다



· T, C 영역의 이동은 현재값(계수값 또는 카운터값)의 이동입니다. 설정값은 이동할 수 없습니다.

2) 프로그램 예

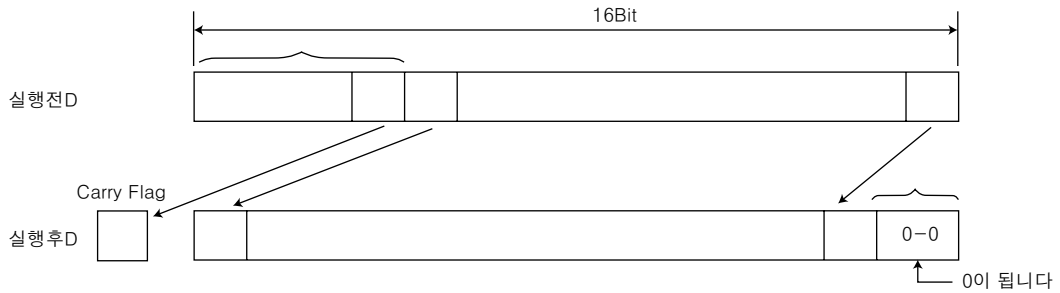
· X0001이 On했을 때 D0002의 내용을 5비트 우로 이동하는 프로그램



[SFL]

1) 기능

· D로 지정된 Device의 16비트 데이터를 좌로 n 비트 이동합니다

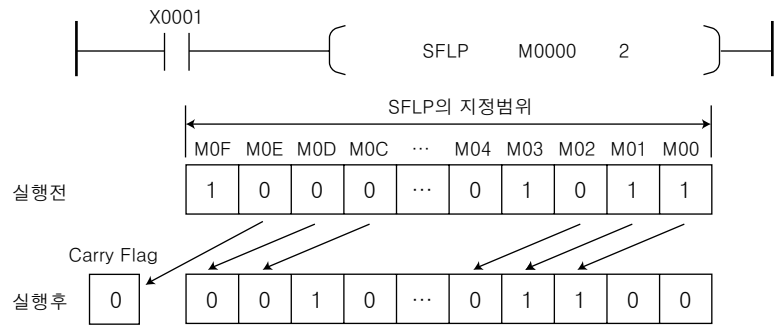


· 최하위부터 n 비트분은 0이 됩니다.

· T, C 영역의 이동은 현재값(계수값 또는 카운터값)의 이동입니다. 설정값은 이동할 수 없습니다.

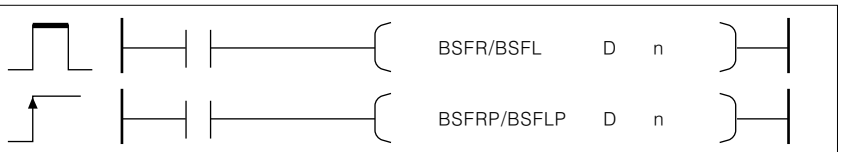
2) 프로그램 예

· X0001 On 했을 때 M0000의 데이터를 2비트 좌로 이동하는 프로그램



[7.8.2 16Bit Data 1 Bit 우 Shift, 좌 Shift : BSFR, BSFRP, BSFL, BSFLP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BSFR(P) BSFL(P)	D	○		○	○	○								3	○		○
	n	○	○	○	○	○	○	○	○	○	○	○	○				



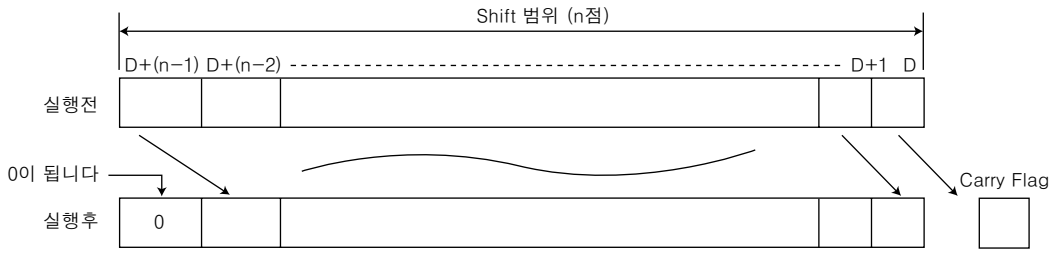
D	이동하는 Device의 선두 어드레스
n	Shift를 수행할 Device의 Bit수

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET
캐리(F112)	연산결과가 오버플로어면 SET

[BSFR]

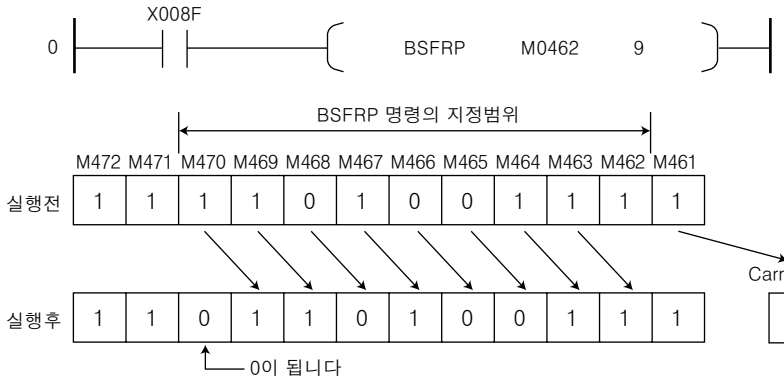
1) 기능

· D로 지정된 Device를 선두로 하여 n 개의 디바이스 bit를 우로 1비트 이동합니다.



2) 프로그램 예

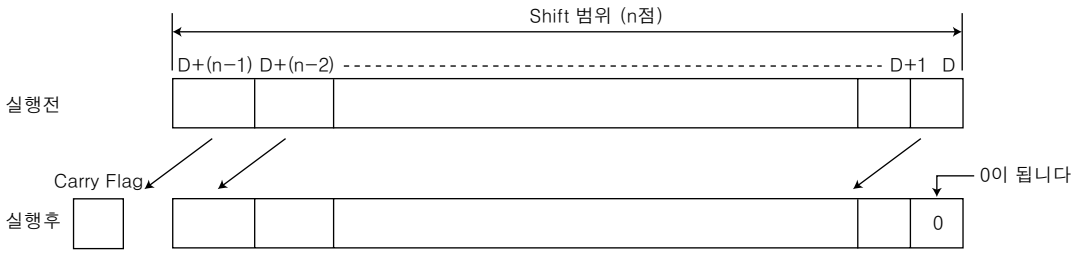
· X008F가 On 했을 때 M0462~M0470의 데이터를 우로 이동하는 프로그램



[BSFL]

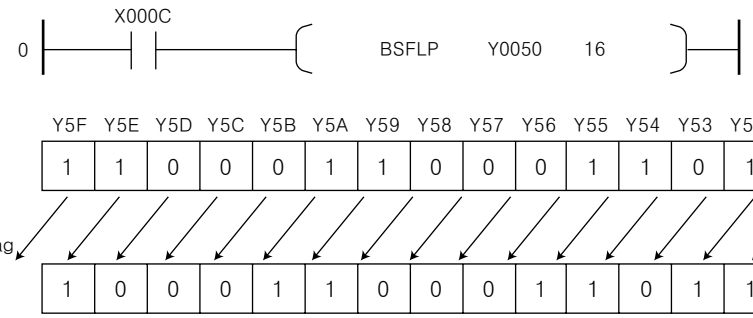
1) 기능

· D로 지정된 Device를 선두로 하여 n개의 디바이스 bit를 좌로 1비트 이동합니다.



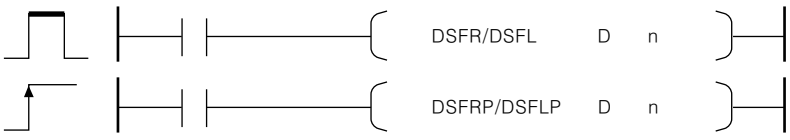
2) 프로그램 예

· X000C가 On했을 때 Y0050~Y005F의 출력을 좌로 이동하는 프로그램



[7.8.3 n Word Data 1 Word 우 shift, 좌 Shift : DSFR, DSFRP, DSFL, DSFLP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
DSFR(P) DSFL(P)	D	O		O	O		O	O	O	O	O	O	O	3	O		O
	n	O	O	O	O	O	O	O	O	O	O	O	O				



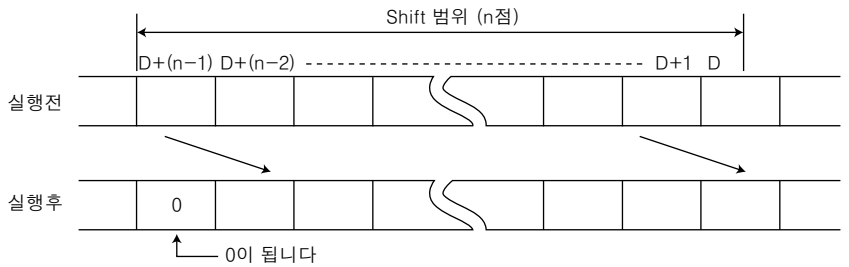
D	이동하는 Device의 선두 어드레스
n	이동되어질 Device의 수

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET
캐리(F112)	연산결과가 오버플로어면 SET

[DSFR]

1) 기능

· D로 지정된 Device를 선두로 하여 n점분을 우로 1점 이동합니다.

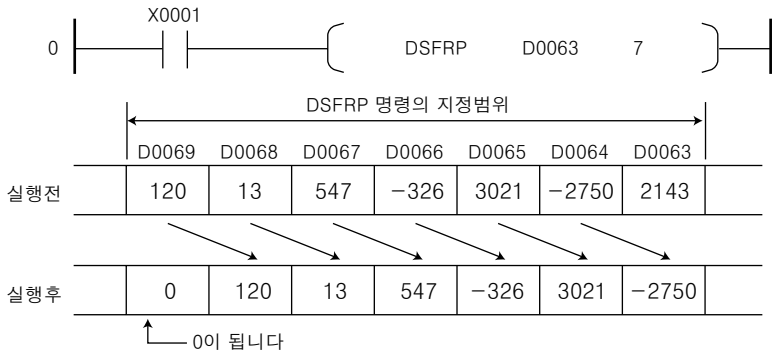


· 최상위 Device는 0이 됩니다.

· T, C 영역의 이동은 현재값(계수값 또는 카운터값)의 이동입니다. 설정값은 이동할 수 없습니다.

2) 프로그램 예

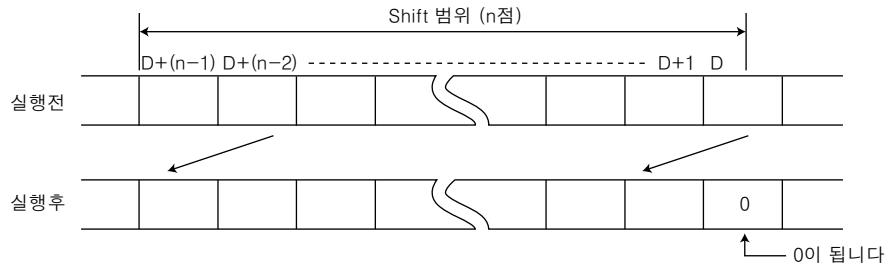
· X0001가 On했을 때 D0063~0069의 내용을 우로 이동하는 프로그램



[DSFL]

1) 기능

· D로 지정된 Device를 선두로 하여 n점분을 좌로 1점 이동합니다.

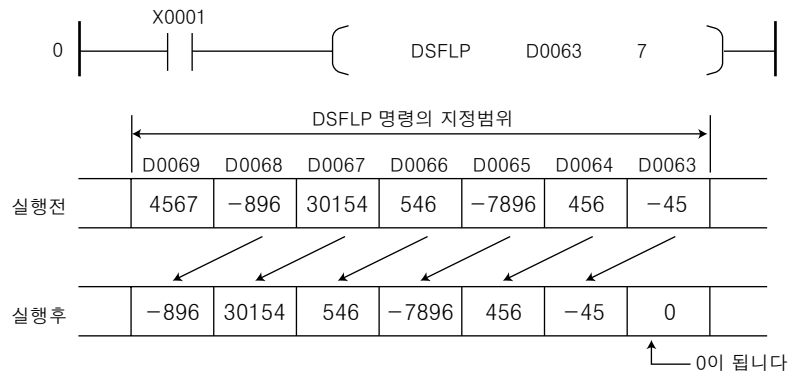


· 최하위의 Device는 0이 됩니다.

· T, C 영역의 이동은 현재값(계수값 또는 카운터값)의 이동입니다. 설정값은 이동할 수 없습니다.

2) 프로그램 예

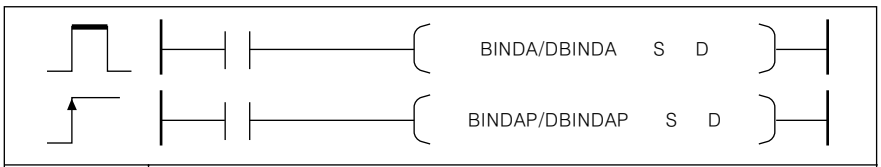
· X0001이 On했을 때 D0063~0069의 내용을 좌로 이동하는 프로그램



7.9 Character String Processing Instruction

[7.9.1 BIN 16bit 또는 32bit data를 10진 ASCII로 변환 : BINDA(P), DBINDA(P)]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BINDA(P) DBINDA(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					



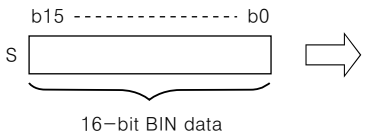
S	ASCII로 변환될 BIN데이터
D	변경된 데이터가 저장될 어드레스중 선두 어드레스

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[BINDA]

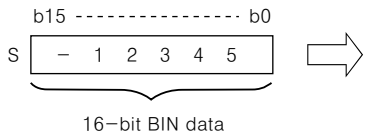
1) 기능

· S에 의해서 지정된 어드레스에 있는 BIN-16 bit data 번호들을 10진수(decimal)로 표시하고 각각의 ASCII code로 변환하여 D가 지정된 어드레스로부터 시작하여 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for ten Thousands place digit	Sign
D+1	ASCII code for hundreds plce digit	ASCII code for thousands place digit
D+2	ASCII code for onces digit	ASCII code for tens place digit

· 예를 들어, S에 있는 값이 -12345 라면 그 결과는 다음과 같이 D에 저장됩니다.



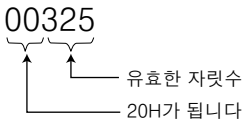
	b15 ---------- b8	b7 ---------- b0
D	31H(1)	2DH(-)
D+1	33H(3)	32H(2)
D+2	35H(5)	34H(4)

· S의 BIN data의 범위는 -32768 에서 32767 입니다.

· 그 결과는 다음과 같이 D에 저장됩니다.

1. BIN data가 양수이면 부호는 20H(space), 음수이면 2DH(-) 가 저장됩니다.

2. 유효한 숫자 왼쪽에 있는 0은 20H(space)로 저장됩니다.

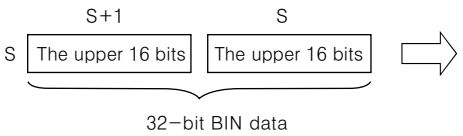


· D+3에 의해 저장된 어드레스에 저장될 데이터는 "0"이 됩니다.

[DBINDA]

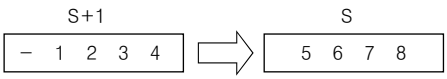
1) 기능

· S에 의해 지정된 어드레스에 있는 BIN 32-bit를 각각의 번호에 대해 "위치 0" 번호에 대해 10진수(Decimal)로 표시하고 ASCII code로 변환하여 D에 의해 지정된 어드레스에 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for Billions place digit	Sign
D+1	ASCII code for ten Millions place digit	ASCII code for Millions place digit
D+2	ASCII code for hundreds thousands place digit	ASCII code for ten Thousands place digit
D+3	ASCII code for Thousands place digit	ASCII code for ten Thousands place digit
D+4	ASCII code for tens place digit	ASCII code for Hundreds place digit
D+5	"20H"	ASCII code for ones digit

· 예를 들어, S에 있는 값이 -12345678 이라면 그 결과는 D에 다음과 같이 저장 됩니다.



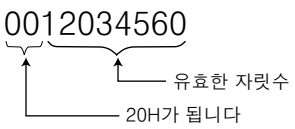
	b15 ---------- b8	b7 ---------- b0
D	20H(space)	2DH(-)
D+1	31H(1)	20H(space)
D+2	33H(3)	32H(2)
D+3	35H(5)	34H(4)
D+4	37H(7)	36H(6)
D+5	20H	38H(8)

· S의 BIN data 의 범위는 -2147483648 에서 2147483647 입니다.

· 그 결과는 다음과 같이 D에 저장됩니다.

1. BIN data가 양수이면 부호는 20H(space), 음수이면 2DH(-) 가 저장됩니다.

2. 유효한 숫자 왼쪽에 있는 0은 20H로 저장됩니다.

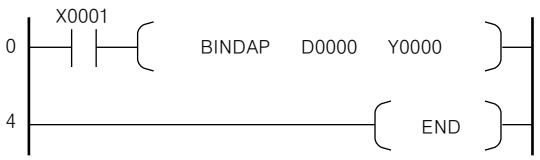


D+5에 의해 지정된 어드레스에 저장될 상위 8bit 데이터는 "0"이 됩니다.

2) 프로그램 예

· X0001이 ON 되었을 때 D0000의 BIN 16bit Data를 10진 ASCII로 변환하여 Y0000에 출력하는 프로그램

[Ladder Mode]

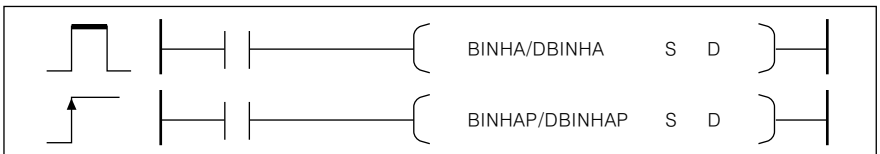


[List Mode]

Steps	Instruction	Device
0	LD	X0001
	BINDAP	D0000
		Y0000
4	END	

[7.9.2 Conversion from BIN 16bit or 32bit data to hexadecimal ASCII : BINHA(P), DBINHA(P)]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BINHA(P) DBINHA(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					

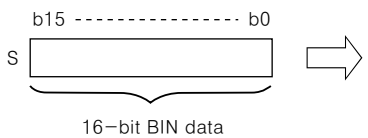


S	ASCII로 변환될 BIN데이터
D	변경된 데이터가 저장될 어드레스중 선두 어드레스

[BINHA]

1) 기능

· S에 의해서 지정된 어드레스에 있는 BIN-16 bit data 번호들를 16진수(hexadecimal)로 표시하기 위해서 ASCII code로 변환하여 D가 지정된 어드레스로부터 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for thrid digit	ASCII code for fourth digit
D+1	ASCII code for first digit	ASCII code for second digit

· 예를 들어, S에 있는 값이 02A6H 라면 그 결과는 다음과 같이 D에 저장됩니다.

	b15 ---------- b8	b7 ---------- b0
D	32H(2)	30H(0)
D+1	36H(6)	41H(A)

· S의 BIN data의 범위는 0에서 FFFFH 입니다.

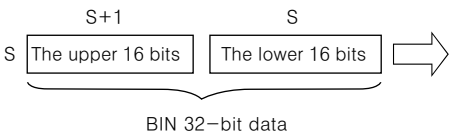
· D에 저장된 결과는 4 자리 hexadecimal 값으로 처리 된다. 유효한 값의 왼쪽에 있는 0들은 값이 "0" 으로 처리 됩니다.

· D+2에 의해 지정된 어드레스의 저장될 데이터는 "0"이 됩니다.

[DBINHA]

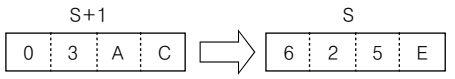
1) 기능

· S에 의해 지정된 어드레스에 있는 BIN 32-bit를 각각의 번호에 대해 16진수(hexadecimal)로 표시하고 ASCII code로 변환하여 D에 의해 지정된 어드레스에 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for seventh digit	ASCII code for eighth digit
D+1	ASCII code for fifth digit	ASCII code for sixth digit
D+2	ASCII code for third digit	ASCII code for fourth digit
D+3	ASCII code for first digit	ASCII code for second digit

예를 들어, S에 있는 값이 03AC625EH 이라면 그 결과는 D에 다음과 같이 저장 됩니다.



	b15 ---------- b8	b7 ---------- b0
D	33H(3)	30H(0)
D+1	43H(C)	41H(A)
D+2	32H(2)	36H(6)
D+3	45H(E)	35H(5)

· S의 BIN data 의 범위는 0 에서 FFFFFFFFH 입니다.

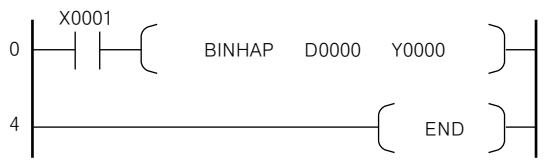
· D에 저장된 결과는 8자리의 hexadecimal값으로 처리됩니다.

· 유효한 자리의 왼쪽에 있는 모든 0은 "0"으로 처리됩니다.

2) 프로그램 예

· X0001이 ON 되었을 때 D0000의 BIN 16bit Data를 16진 ASCII로 변환하여 Y0000에 출력하는 프로그램

[Ladder Mode]

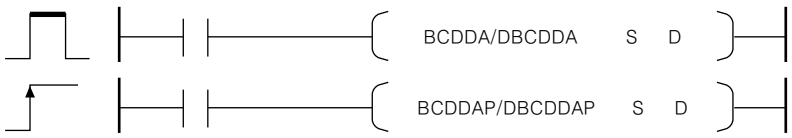


[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	BINHAP	D0000
		Y0000
4	END	

[7.9.3 Conversion from BCD 4digit and 8digit to decimal ASCII data : BCDDA(P), DBCDDA(P)]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
BCDDA(P) DBCDDA(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					



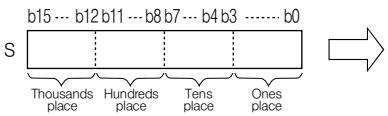
S	ASCII로 변환될 BIN데이터
D	변경된 데이터가 저장될 어드레스중 선두 어드레스

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[BCDDA]

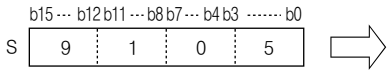
1) 기능

· S에 의해서 지정된 어드레스에 있는 BCD 4자리의 번호를 ASCII code로 변환하여 D가 지정된 어드레스로부터 시작하여 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for hundreds place digit	ASCII code for thousands place digit
D+1	ASCII code for ones digit	ASCII code for tens place digit

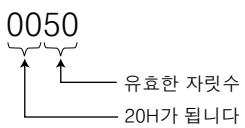
· 예를 들어, S에 있는 값이 9105라면 그 결과는 다음과 같이 D에 저장됩니다.



	b15 ---------- b8	b7 ---------- b0
D	31H(1)	39H(9)
D+1	35H(5)	30H(0)

· S의 BCD data의 범위는 0에서 9999입니다.

· 계산의 결과는 D에 저장되고, "유효한 자리" 왼쪽의 모든 0은 20H로 저장됩니다.

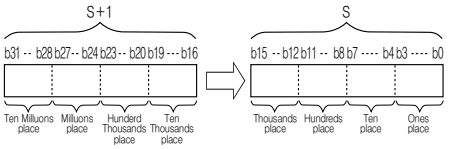


· D+2에 의해 지정된 어드레스에 저장될 데이터는 "0"이 됩니다.

[DBCDDA]

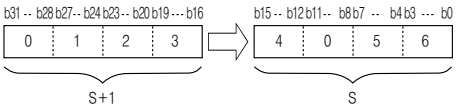
1) 기능

· S에 의해 지정된 어드레스에 있는 BCD 8자리의 각각의 번호를 ASCII code로 변환하여 D로 지정된 어드레스에 저장합니다.



	b15 ---------- b8	b7 ---------- b0
D	ASCII code for millions place digit	ASCII code for ten millions place digit
D+1	ASCII code for ten Thousands place digit	ASCII code for hundred thousands place digit
D+2	ASCII code for Hundreds place digit	ASCII code for Thousands place digit
D+3	ASCII code for Ones digit	ASCII code for Thousands place digit

· 예를 들어, S에 있는 값이 01234056 이라면 그 결과는 D에 다음과 같이 저장 됩니다.



	b15 ---------- b8	b7 ---------- b0
D	31H(1)	20H
D+1	33H(3)	32H(2)
D+2	30H(0)	34H(4)
D+3	36H(6)	35H(5)

· S의 BCD data 의 범위는 0 에서 99999999 입니다.

· 계산의 결과는 D에 저장되고 유효한 자리의 왼쪽에 있는 모든 0은 20H로 처리됩니다.

2) 연산 Error

· 다음의 경우에는 에러가 발생하며, 에러 Flag가 ON이 되고, Error Code는 F0050에 저장됩니다.

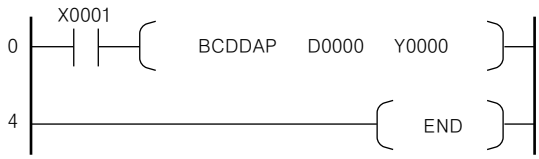
1. BCDDA 명령 수행 시 데이터가 0에서 9999를 벗어날 경우

2. DBCDDA 명령 수행 시 데이터가 0에서 99999999를 벗어날 경우

3) 프로그램 예

· X0001이 ON 되었을 때 D0000의 BCD 4 digit를 decimal ASCII를 변화하여 Y0000에 출력하는 프로그램

[Ladder Mode]

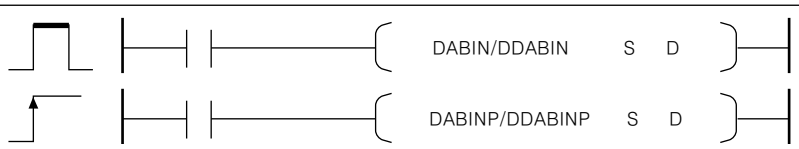


[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	BCDDAP	D0000
		Y0000
4	END	

[7.9.4 Conversion from decimal ASCII to BIN 16bit and 32bit data : DABIN(P), DDABIN(P)]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
DABIN(P) DDABIN(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					



S	BIN로 변경될 ASCII data가 저장될 어드레스 중 선두 어드레스
D	변경된 데이터가 저장될 어드레스중 선두 어드레스

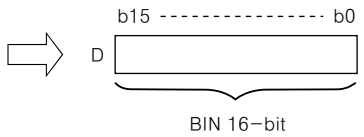
에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[DABIN]

1) 기능

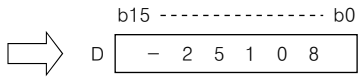
· S에 의해서 지정된 어드레스에 있는 decimal ASCII data를 D로 지정된 어드레스에 BIN 16-Bit data로 저장합니다.

	b15 ---------- b8	b7 ---------- b0
S	ASCII code for ten Thousands place digit	Sign data
S+1	ASCII code for Hundreds place digit	ASCII code for Thousands place digit
S+2	ASCII code for ones digit	ASCII code for tens Place digit



· 예를 들어 S의 ASCII code가 -25108H이면, 그 연산 결과는 다음과 같은 결과로 D에 저장됩니다.

	b15 ---------- b8	b7 ---------- b0
S	32H(2)	2DH(-)
S+1	31H(1)	35H(5)
S+2	38H(8)	30H(0)



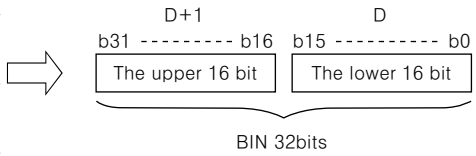
· S와 S+2에 지정된 ASCII data는 -32768 에서 32767 의 범위에 있습니다.
· Data가 양수이면 부호는 20H(space), 음수이면 2DH(-)가 저장됩니다.
· ASCII code의 범위는 "30H"에서 "39H" 사이에서는 어떤 각각의 위치도 설정 될 수 없습니다.
· ASCII code의 각각의 위치가 20H나 00H 라면 그것은 30H로 처리됩니다.

[DDABIN]

1) 기능

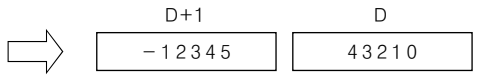
· S에 의해서 지정된 어드레스에 있는 Decimal ASCII data를 D로 지정된 어드레스에 BIN 32bit data로 저장합니다.

	b15 ---------- b8	b7 ---------- b0
S	ASCII code for billions place digit	Sign data
S+1	ASCII code for ten millions place digit	ASCII code for hundred millions place digit
S+2	ASCII code for hundred thousands place digit	ASCII code for millions place digit
S+3	ASCII code for thousands place digit	ASCII code for ten thousands place digit
S+4	ASCII code for tens place digit	ASCII code for hundreds place digit
S+5	(Ignored)	ASCII code for ones place digit



· 예를 들어 S의 ASCII code가 –1234543210 이면, 그 결과는 다음과 같은 결과로 D+1과 D에 저장됩니다.

	b15 ---------- b8	b7 ---------- b0
S	31H(1)	2DH(–)
S+1	33H(3)	32H(2)
S+2	35H(5)	34H(4)
S+3	33H(3)	34H(4)
S+4	31H(1)	32H(2)
S+5		30H(0)



· S 와 S+5에 지정된 ASCII data는 –2147483648 에서 2147483647의 범위에 있습니다.

· 추가로 S+5의 상위 byte에 저장되는 data는 무시합니다.

· 변환된 data의 값이 양수 이면 부호는 20H(space)가 되고, 음수 이면 2DH(–)가 됩니다.

· ASCII code의 범위는 "30H" 에서 "39H" 사이에서는 어떤 각각의 위치도 설정 될 수 없습니다.

· ASCII code 각각의 위치가 20H나 00H 라면 그것은 30H로 처리됩니다.

2) 연산 Error

· 다음의 경우에는 에러가 발생하며, 에러 Flag가 ON이 되고, Error code는 F0050에 저장됩니다.

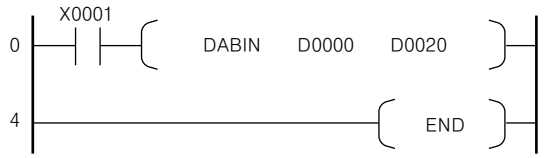
· S 에서 S+5 의 ASCII code가 30H로부터 39H와 20H 또는 00H 가 아닐 경우 에러 발생

· S 에서 S+5 의 ASCII data가 범위를 벗어났을 경우

3) 프로그램 예

· Decimal 5자리 ASCII data 그리고 D0000에서 D0002까지의 설정된 sign을 Bin으로 바꾸어 D0020에 결과를 저장하는 프로그램

[Ladder Mode]

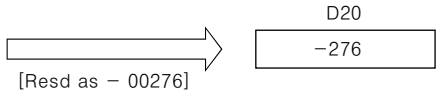


[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	DABIN	D0000
		D0020
4	END	

	b15 ---------- b8	b7 ---------- b0
D0000	20H(space)	2DH(–)
D0001	32H(2)	20H(space)
D0002	36H(6)	37H(7)

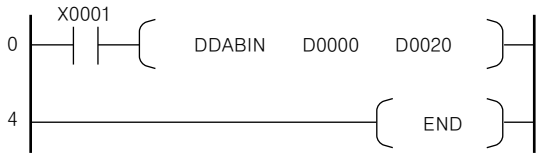
" ␣ ␣ 276"



[Resd as – 00276]

· Decimal 10자리 ASCII data 그리고 D0000~D0005까지 설정된 sign을 BIN으로 바꾸어 D0020과 D0021에 결과를 저장하는 프로그램

[Ladder Mode]

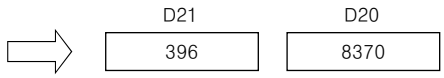


[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	DDABIN	D0000
		D0020
4	END	

	b15 ---------- b8	b7 ---------- b0
D0000	20H(space)	20H(space)
D0001	20H(space)	20H(space)
D0002	39H(9)	33H(3)
D0003	38H(8)	36H(6)
D0004	37H(7)	33H(3)
D0005		30H(0)

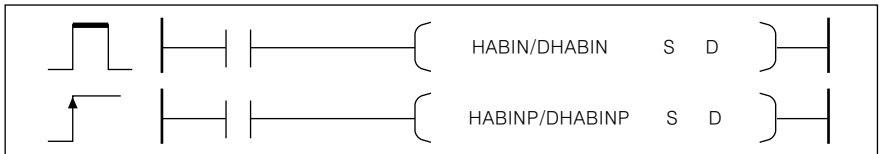
" ␣ ␣ ␣ ␣ 276"



[Resd as + 0003968370]

[7.9.5 Conversion from hexadecimal ASCII to BIN 16-bit data : HABIN(P), DHABIN(P)]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
HABIN(P) DHABIN(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					



S	BIN로 변경될 ASCII data가 저장될 어드레스 중 선두 어드레스
D	변경된 데이터가 저장될 어드레스중 선두 어드레스

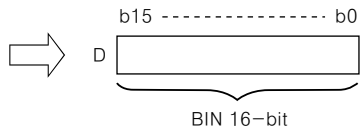
에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET

[HABIN]

1) 기능

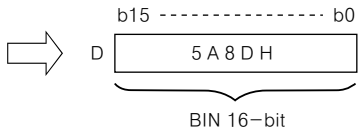
· S에 의해서 지정된 어드레스에 있는 16진(hexadecimal) ASCII data를 D로 지정된 어드레스 BIN 16-Bit data로 저장합니다.

	b15 ---------- b8	b7 ---------- b0
S1	ASCII code for third digit	ASCII code for fourth digit
S1+1	ASCII code for first digit	ASCII code for second digit



· 예를 들어 S의 ASCII code가 5A8DH이면, 그 연산 결과는 다음과 같은 결과로 D에 저장됩니다.

	b15 ---------- b8	b7 ---------- b0
S1	41H(A)	35H(5)
S1+1	44H(D)	38H(8)



· S 와 S+1에 지정된 ASCII data는 0000H 에서 FFFFH 의 범위에 있습니다.

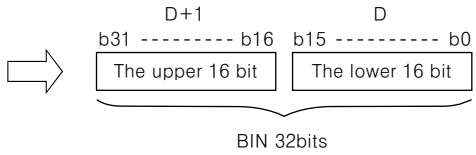
· ASCII code 들의 범위는 "30H" 에서 "39H" 와 "41H" 에서 "46H"의 범위에 있습니다.

[DHABIN]

1) 기능

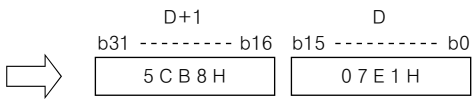
· S에 의해서 지정된 어드레스에 있는 16진(hexadecimal) ASCII data를 D로 지정된 어드레스에 BIN 32bit data로 저장합니다.

	b15 ---------- b8	b7 ---------- b0
S1	ASCII code for seventh digit	ASCII code for eighth digit
S1+1	ASCII code for fifth digit	ASCII code for sixth digit
S1+2	ASCII code for thrid digit	ASCII code for fourth digit
S1+3	ASCII code for first digit	ASCII code for second digit



· 예를 들어 S의 ASCII code가 5CB807E1H 이면, 그 결과는 다음과 같이 D+1과 D에 저장됩니다.

	b15 ---------- b8	b7 ---------- b0
D0000	43H(C)	35H(5)
D0001	38H(8)	42H(B)
D0002	37H(7)	30H(0)
D0003	31H(1)	45H(E)



· S 와 S+3에 지정된 ASCII data는 00000000H 에서 FFFFFFFFH의 범위에 있습니다.

· ASCII code들의 범위는 "30H" 에서 "39H" 와 "41H" 에서 "46H" 의 범위에 있습니다.

2) 연산 Error

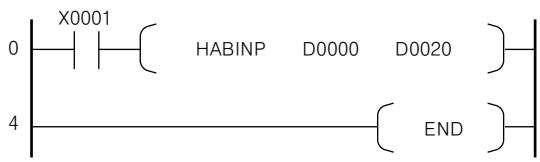
· 다음의 경우 에러가 발생하며, 에러 Flag가 ON이 되고, Error Code는 F0050에 저장됩니다.

· S 에서 S+3 의 어드레스에 있는 ASCII code가 30H 에서 39H와 "41H" 에서 "46H" 의 범위를 벗어날 경우

3) 프로그램 예

· D0000과 D0001에 저장된 16진(hexadecimal) ASCII data를 BIN data로 변경하고 D0020에 저장하는 프로그램

[Ladder Mode]

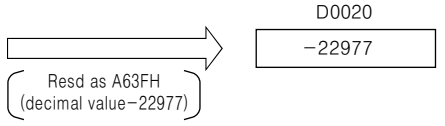


[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	HABINP	D0000
		D0020
4	END	

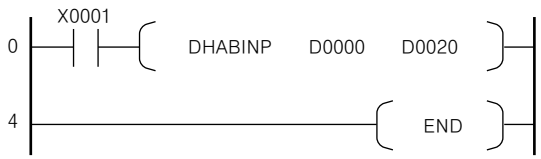
	b15 ---------- b8	b7 ---------- b0
D0000	36H(6)	41H(A)
D0001	46H(F)	33H(3)

"A63F"



· D0000과 D0003에 저장된 16진(hexadecimal) ASCII data를 BIN data로 변경하여 D0020과 D0021에 저장하는 프로그램

[Ladder Mode]



[List Mode]

Steps	Instruction	Device
0	LD	X0001
1	DHABINP	D0000
		D0020
4	END	

	b15 ---------- b8	b7 ---------- b0
D0000	46H(F)	34H(4)
D0001	32H(2)	44H(D)
D0002	37H(7)	38H(8)
D0003	30H(0)	35H(5)

"4FD28750"

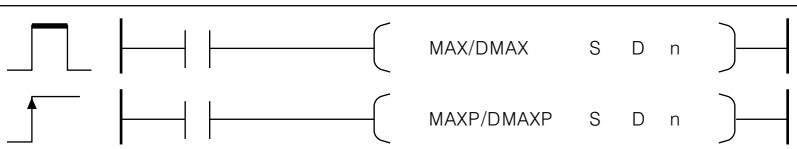
Resd as 4FD28750H
(decimal value1339197264)

D0021	D0020
13391	97264

7.10 Data Processing Instruction

[7.10.1 Maximum value search for 16- and 32-bit data : MAX, MAXP, DMAX, DMAXP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
MAX(P) DMAX(P)	S	O	O	O	O	O	O	O	O	O	O	O		4			
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				

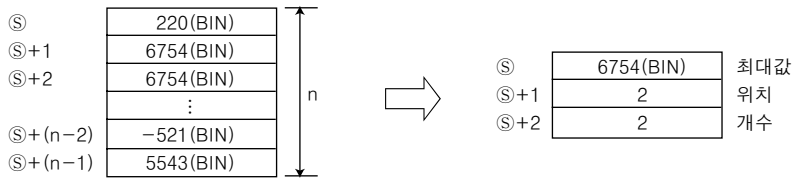


S	최대값을 찾기 위한 Device의 첫번째 번호
D	최대값을 찾은 결과를 저장하는 Device 첫번째 번호
n	찾아진 데이터 Block수

[MAX]

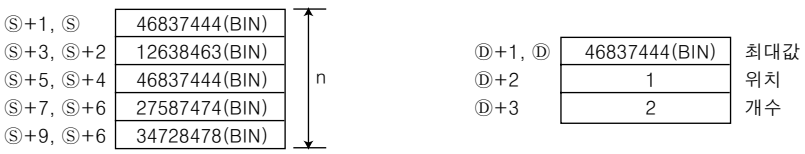
1) 기능

S로 지정된 Device 안에 있는 16bit 데이터의 n block에서 최대값을 찾아 D로 지정된 Device에 최대값을, D+1로 지정된 Device에 최대값이 위치한 첫번째 블록 번호를, D+2로 지정된 Device에 최대값을 갖는 블록의 개수를 저장합니다.



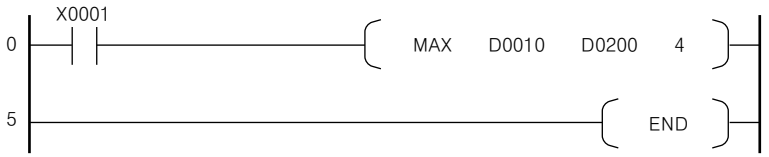
[DMAX]

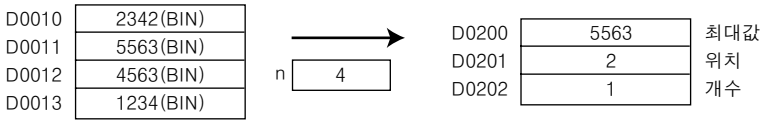
S로 지정된 Device 안에 있는 32Bit 데이터의 n block 에서 최대값을 찾아 D, D+1로 지정된 Device에 최대값을, D+2로 지정된 Device에 최대값이 위치한 첫번째 블록 번호를, D+3으로 지정된 Device에 최대값을 갖는 블록의 개수를 저장합니다.



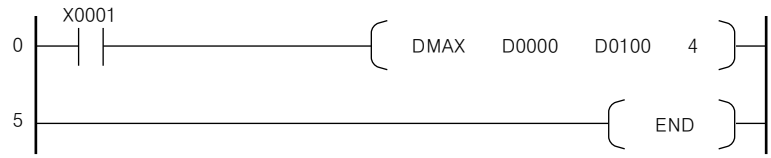
2) 프로그램 예

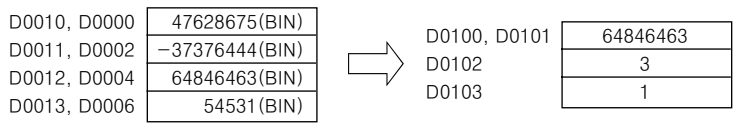
· X0001이 ON이 되었을 때 D0010부터 4만큼 저장된 데이터에서(D10~D13) 최대값을 찾아 최대값, 최대값의 위치, 최대값의 개수 순서로 D200에서 D202까지 저장하는 프로그램





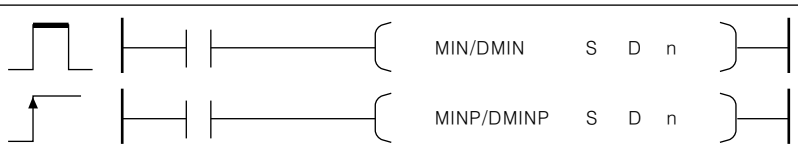
· X0001이 ON이 되었을 때 D0000에서 32bit 데이터 D0007까지의 최대값을 찾아 D0100에서 D0103까지 저장하는 프로그램





[7.10.2 Minimum value search for 16- and 32-bit data : MIN, MINP, DMIN, DMINP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
MIN(P) DMIN(P)	S	O	O	O	O	O	O	O	O	O	O	O		4			
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				

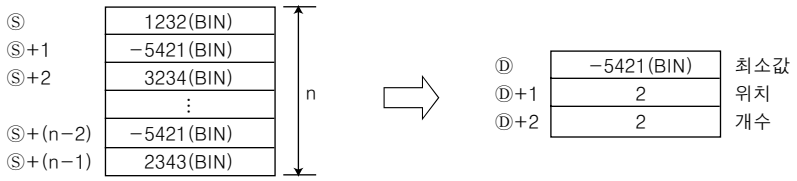


S	최대값을 찾기 위한 Device의 첫번째 번호
D	최대값을 찾은 결과를 저장하는 Device 첫번째 번호
n	찾아진 데이터 Block수

[MIN]

1) 기능

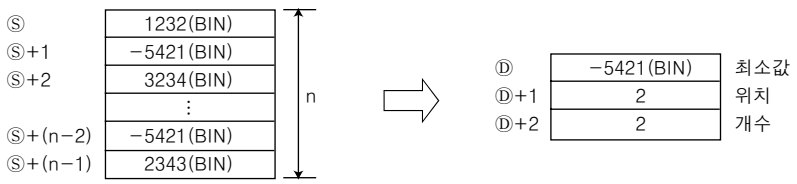
S로 지정된 Device 안에 있는 16bit 데이터의 n block에서 최소값을 찾아 D로 지정된 Device에 최소값을, D+1로 지정된 Device에 최소값이 위치한 첫번째 블록 번호를, D+2로 지정된 Device에 최소값을 갖는 블록의 개수를 저장합니다



[DMIN]

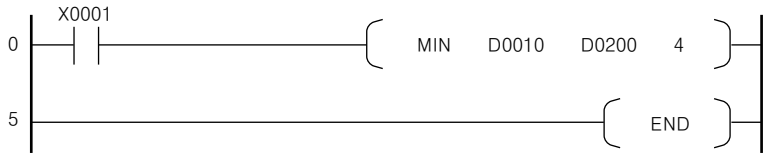
1) 기능

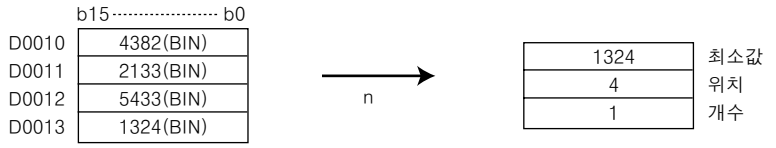
S로 지정된 Device 안에 있는 32bit 데이터의 n block에서 최소값을 찾아 D로 지정된 Device에 최소값을, D+2로 지정된 Device에 최소값이 위치한 첫번째 블록 번호를, D+3로 지정된 Device에 최소값을 갖는 블록의 개수를 저장합니다



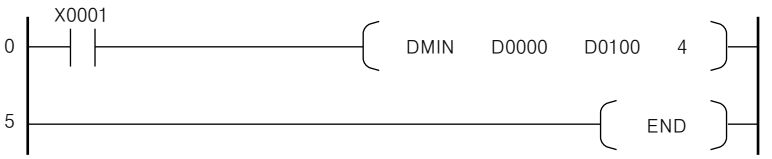
2) 프로그램 예

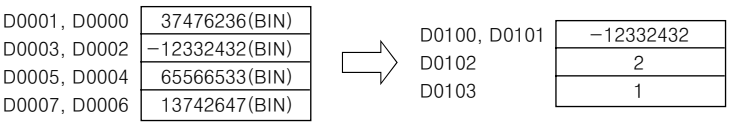
· X0001이 ON이 되었을 때 D0010부터 4만큼 저장된 데이터에서(D10~D13) 최소값을 찾아 최소값, 최소값의 위치, 최소값의 개수 순서로 D200에서 D202까지 저장하는 프로그램





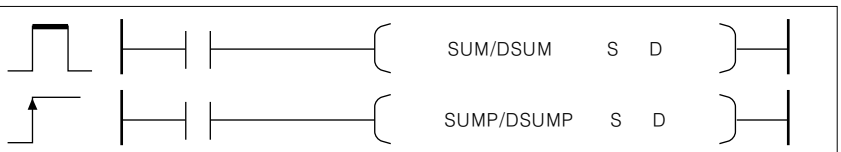
· X0001이 ON이 되었을 때 D0000에서부터 D0007까지 32bit 데이터 가운데 최소값을 찾아 D0100에서 D0103까지 저장하는 프로그램





[7.10.3 16-bit and 32-bit Data Checks : SUM, SUMP, DSUM,DSUMP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
SUM(P) DSUM(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3			
	D	O		O	O	O		O	O	O	O	O					



S	1로 SET되어진 비트를 계수 할 디바이스
D	S로 지정된 디바이스에서 계수한 비트의 수를 저장할 디바이스

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET 0으로 나누는 경우(S2의 값이 0인 경우)

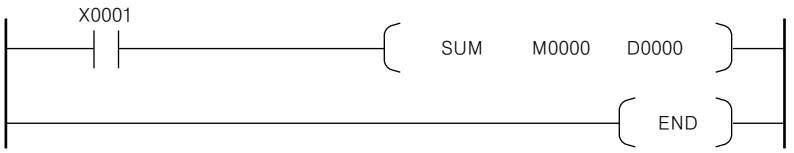
[SUM]

1) 기능

S로 지정된 영역 워드 데이터 영역에서 Set되어 있는 비트수를 계수하여 그 수를 D로 지정된 워드데이터 영역에 저장합니다.

2) 프로그램 예

X0001이 On되면 M0000의 워드데이터 중 1로 Set 되어 있는 비트를 계수하여 D0000에 저장하는 프로그램



M0010 워드데이터 (S로 지정된 영역)

b15															b0
0	0	1	1	1	1	0	0	1	0	1	0	0	1	1	1

1로 Set 되어진 비트데이터 수 : 9개

⇩

D0000 워드데이터 (D로 지정된 영역)

0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

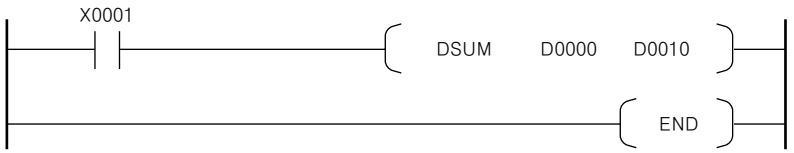
[DSUM]

1) 기능

S로 지정된 영역 더블워드 데이터 영역에서 Set되어 있는 비트수를 계수하여 그 수를 D로 지정된 워드데이터 영역에 저장합니다.

2) 프로그램 예

X0001이 On되면 D0000의 더블워드 데이터 중 1로 Set 되어 있는 비트를 계수하여 D0010에 저장하는 프로그램



D0000 더블워드데이터 (S로 지정된 영역)

b31																												b0
1	1	1	1	1	0	0	1	0	1	0	1	1	1	1	0	1	0	1	0	1	0	0	1	0	1	1	1	1	0	0	1

1로 Set 되어진 비트데이터 수 : 20개

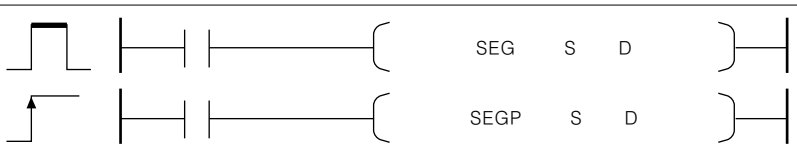
⇩

D0010 워드데이터 (D로 지정된 영역)

0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0

[7.10.4 7-segment decode : SEG, SEGP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
SEG(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	3	O		
	D	O		O	O	O		O	O	O	O	O					



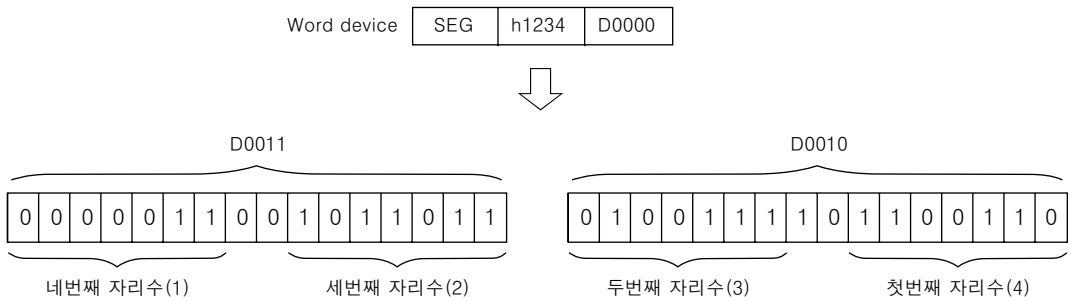
S	Decoded data에 저장하기 시작하는 선두 device 번호 혹은 decoded data
D	결과값을 저장하는 선두 device 번호

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

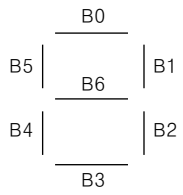
[SEG]

1) 기능

· S로 지정된 워드 디바이스의 데이터를 자리수별로 차례대로 7 Segment 형태의 Decode Data를 D로 지정된 영역에 더블 워드 데이터로 저장합니다.

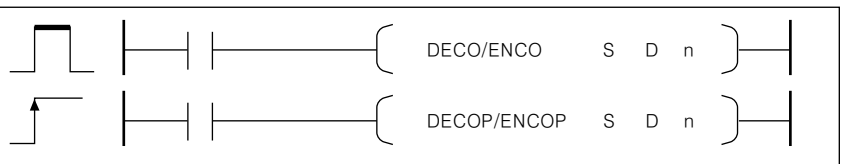


2) 7-segment decode display

S		Configuration of7 Segments	D							
Hexa decimal	-Bit Pattern		B7	B6	B5	B4	B3	B2	B1	B0
0	0000		0	0	1	1	1	1	1	1
1	0001		0	0	0	0	0	1	1	0
2	0010		0	1	0	1	1	0	1	1
3	0011		0	1	0	0	1	1	1	1
4	0100		0	1	1	0	0	1	1	0
5	0101		0	1	1	0	1	1	0	1
6	0110		0	1	1	1	1	1	0	1
7	0111		0	0	1	0	0	1	1	1
8	1000		0	1	1	1	1	1	1	1
9	1001		0	1	1	0	1	1	1	1
A	1010		0	1	1	1	0	1	1	1
B	1011		0	1	1	1	1	1	0	0
C	1100		0	0	1	1	1	0	0	1
D	1101		0	1	0	0	1	1	1	0
E	1110		0	1	1	1	1	0	0	1
F	1111		0	1	1	1	0	0	0	1

[7.10.5 8 ↔ 256 Bit Decode, Encode : DECO, DECOP, ENCO, ENCOP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
DECO(P) ENCO(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



S	Decode, Encode를 실행할 데이터 또는 데이터를 저장하고 있는 Device의 번호
D	Decode, Encode 결과를 저장하는 Device의 선두 어드레스
n	유효 비트 길이(1~8)

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[DECO]

1) 기능

· 8 → 256 Bit Decode

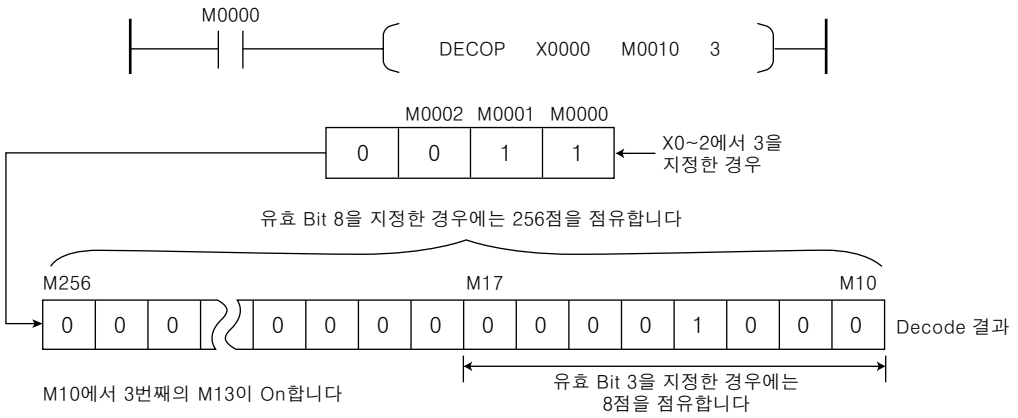
· S로 지정된 Device의 하위 n 비트를 Decode하고 그 결과를 D로 지정된 Device로부터 2^n 비트에 저장합니다.

· n은 1 ~ 8까지 지정 가능합니다.

· n=0인 때에는 무처리로 D로 지정된 Device로부터 2^n 비트의 내용은 변화하지 않습니다.

· Bit Decode는 1비트, Word Decode는 16비트로 취급합니다.

2) 프로그램 예



[ENCO]

1) 기능

· 256 → 8 Bit Encode

· S로부터 2^n 비트의 데이터를 Encode하여 D로 지정된 영역에 저장합니다.

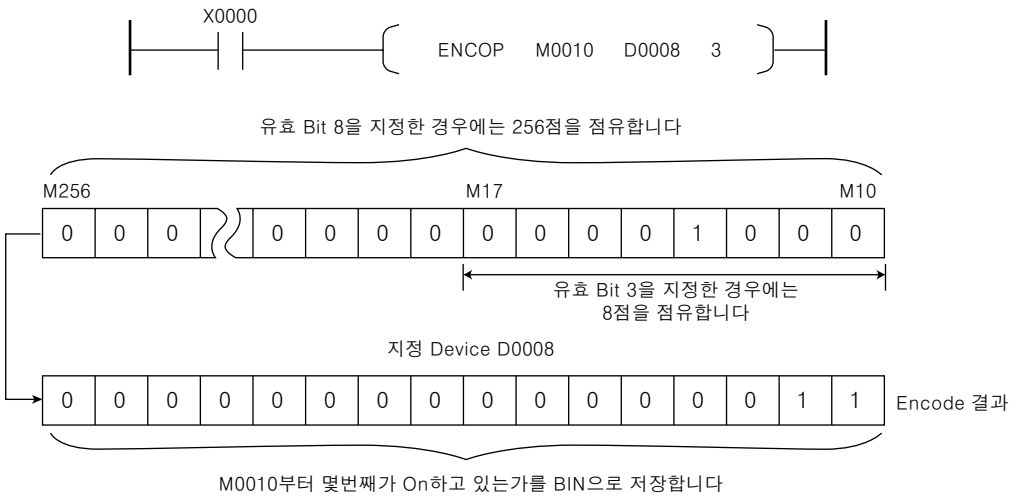
· n은 0 ~ 8까지 지정 가능합니다.

· n=0인 때에는 무처리로 D의 내용은 변화하지 않습니다.

· Bit Device는 1비트, Word Device는 16비트로 취급합니다.

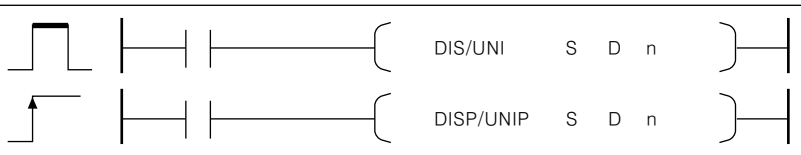
· 복수의 비트가 1일 때에는 최후 비트 위치에서 처리됩니다.

2) 프로그램 예



[7.10.6 16Bit Data의 분리, 결합 : DIS, DISP, UNI, UNIP]

명령		사용가능영역												스텝수	플래그		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리
DIS(P) UNI(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	4	O		
	D	O		O	O	O		O	O	O	O	O					
	n	O	O	O	O	O	O	O	O	O	O	O	O				



S	분리 결합하는 데이터가 저장되어 있는 Device의 선두 어드레스
D	분리, 결합하는 데이터를 저장하는 번호
n	분리 저장하는 Device 수(1~4), 결합 데이터 수(1~4)

에러(F110)	영역이 @D로 지정된 경우 영역 초과가 발생하면 SET

[DIS]

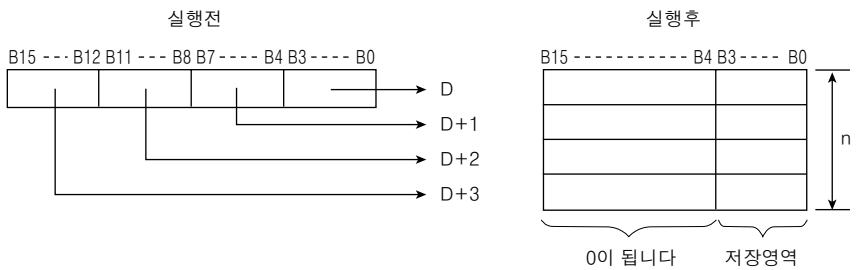
1) 기능

· S로 지정된 16비트 데이터의 하위 n자리(n자리 4비트)분의 데이터를 D로 지정된 Device에서 n점분의 하위 4비트에 저장합니다.

· D로 지정된 Device에서 n점분의 상위 12비트는 0이 됩니다.

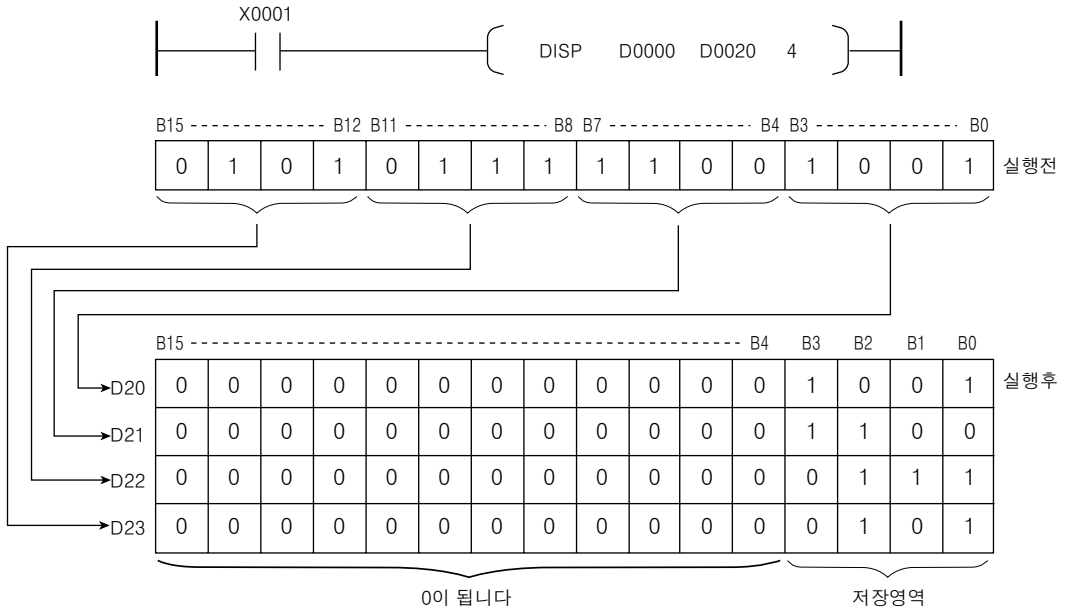
· n은 1 ~ 4까지 지정할 수 있습니다.

· n=0일 경우에는 무처리로 D의 Device에서 n점의 내용은 변화하지 않습니다.



2) 프로그램 예

· X0001이 On 되었을 때 D0000의 16비트 데이터를 4비트마다 D0020 ~ 0023에 저장하는 프로그램



[UNI]

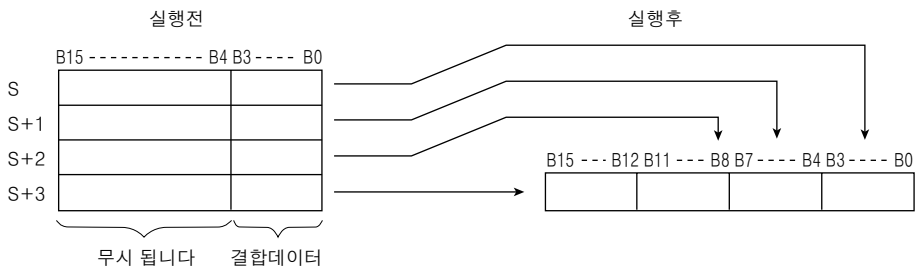
1) 기능

· S로 지정된 Device에서 n점분의 16비트 데이터의 하위 4비트를 D로 지정된 16비트 Device에 결합합니다.

· D로 지정된 Device의 상위(4-n)자리 비트는 0이 됩니다.

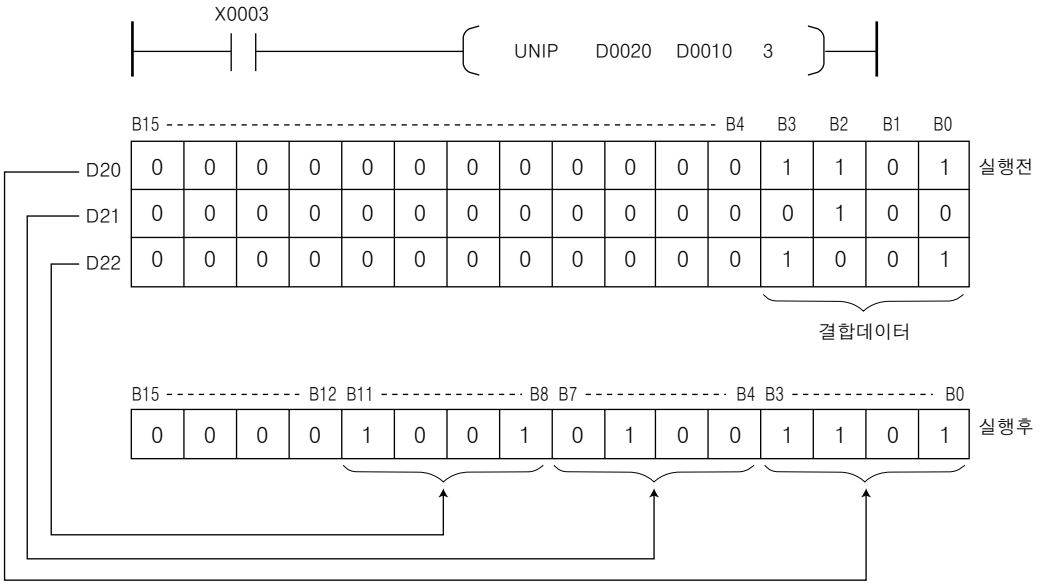
· n은 1 ~ 4를 지정할 수 있습니다.

· n=0일 경우에는 무처리로 D의 Device 데이터 내용은 변화하지 않습니다.



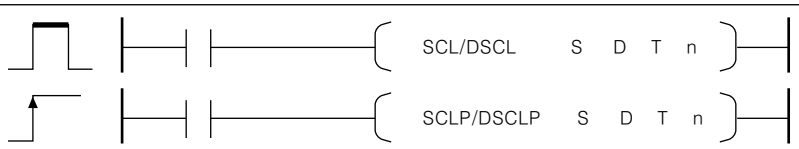
2) 프로그램 예

· X0003이 On 되었을 때 D0020 ~ 0022의 하위 4비트 데이터를 D0010에 저장하는 프로그램



[7.10.7 Data Scaling according to the specified linear function : SCL, SCLP, DSCL, DSCLP]

명령		사용가능영역												스텝수	플래그			사용가능 CPU		
		M	X	Y	K	L	F	T	C	Z	D	@D	정수		에러	제로	캐리	XP	CP	BP
SCL(P)	S	O	O	O	O	O	O	O	O	O	O	O	O	5	O			O	O	O
	D	O		O	O	O				O	O	O								
	n	O	O	O	O	O	O	O	O	O	O	O	O							



S	스케일링 변환할 데이터
D	스케일링 변환된 데이터
T	스케일 정보가 저장되어 있는 테이블
n	스케일링 변환할 데이터의 수

에러(F110)	영역이 @D로 간접 지정된 경우 영역 초과가 발생하면 SET Src.Start와 Src.End가 같은 경우 SET(명령어 처리하지 않음)

[SCL]

1) 기능

· S로 지정된 워드 데이터와 T로 지정된 스케일 정보가 저장되어 있는 테이블에 따라 스케일링 변환후 D로 지정된 영역에 저장합니다.

• 스케일 정보 테이블(총 4워드)

T	스케일링 할 Source Data의 시작 (Src. Start)
T + 1	스케일링 할 Source Data의 마지막 (Src. End)
T + 2	스케일링 된 Destination Data의 시작 (Dest. Start)
T + 3	스케일링 된 Destination Data의 마지막 (Dest. End)

[DSCL]

1) 기능

· S로 지정된 더블 워드 데이터와 T로 지정된 스케일 정보가 저장되어 있는 테이블에 따라 스케일링 변환후 D로 지정된 영역에 저장합니다.

· 스케일 정보 테이블(총 8워드)

T, T + 1	스케일링 할 Source Data의 시작 (Src. Start)
T + 2, T + 3	스케일링 할 Source Data의 마지막 (Src. End)
T + 4, T + 5	스케일링 된 Destination Data의 시작 (Dest. Start)
T + 6, T + 7	스케일링 된 Destination Data의 마지막 (Dest. End)

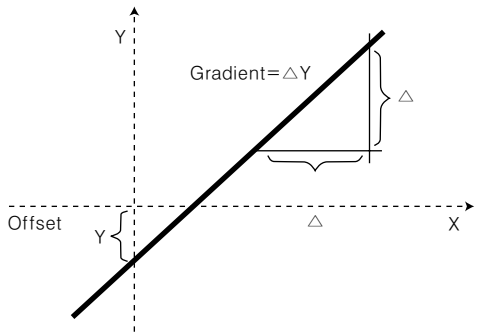
· 스케일링 공식

1. Destination = Source * Gradient(2) + Offset(3)

2. Gradient (기울기) = $\frac{\text{Dest.End} - \text{Dest.Start}}{\text{Src.End} - \text{Src.Start}}$

3. Offset (Y Intercept) = Dest.End – Gradient(2) * Src.Start

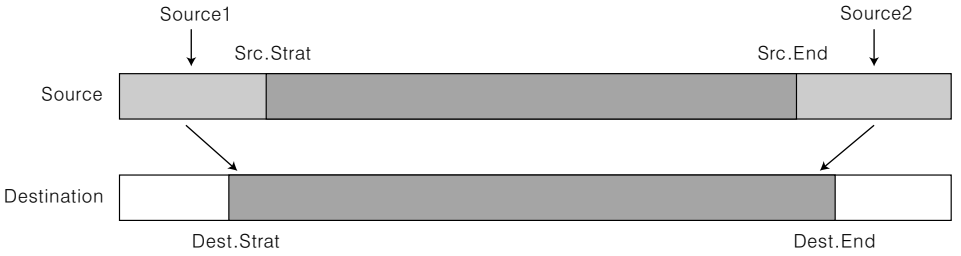
· Graph



· S로 지정된 스케일링 변환할 데이터가 스케일 정보의 Source Data의 범위를 벗어난 경우 Destnation Data의 최대 또는 최소값으로 표시됩니다.

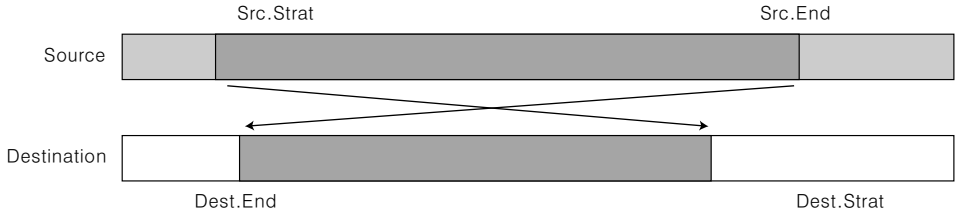
✓ Source1인 경우 Dest. Start로 스케일링

✓ Source2인 경우 Dest. End로 스케일링



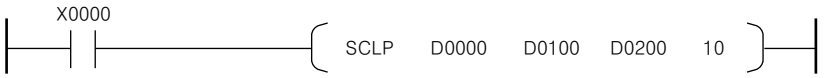
· Src.End가 Src.Start보다 작거나 Dest.End가 Dest.Start보다 작은 경우

✓ Gradient가 음수인 경우



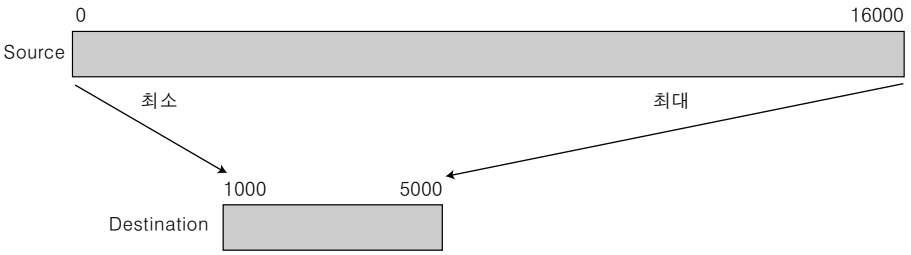
2) 프로그램 예

· X00000이 On 되었을 때 D0000~D0009부터 10워드의 데이터를 D0200에서 D0203의 정보테이블에 맞춰 스케일링 변환하여 D0100~D0109부터 10워드를 저장하는 프로그램



D0200에서 부터 4워드(스케일링 정보 테이블)

D0200	0	← 스케일링 할 Source Data의 최소값
D0201	16000	← 스케일링 할 Source Data의 최대값
D0202	1000	← 스케일링 할 Destination의 최소값
D0203	5000	← 스케일링 할 Destination의 최대값



Source Device	Scaling Value		Destination Device	Scaled Value
D00000	-192	→	D00100	1000
D00001	0	→	D00101	1000
D00002	500	→	D00102	1125
D00003	4000	→	D00103	2000
D00004	8000	→	D00104	3000
D00005	10000	→	D00105	3500
D00006	12000	→	D00106	4000
D00007	13500	→	D00107	4375
D00008	16000	→	D00108	5000
D00009	16192	→	D00109	5000