

Ezi-MOTIONGATE

DeviceNet

PLC 예제 (MASTER K)



Fast, Accurate, Smooth Motion Control

Manual Version ; [ver 1.02.01]

적용 Plus-R Firmware version : 6.0.0.00 이상

초 판 : 2012 년 3 월 22 일

개정판 : 2012 년 7 월 31 일

www.fastech.co.kr

- 목 차 -

1. 안전상의 주의사항	6
2. 제품 사양 및 크기	9
2.1 개요	9
2.2 네트워크 별 제품 특성 표	10
2.3 제품 외형	11
2.3.1 제품 치수	11
2.3.2 각 부의 명칭	12
2.3.3 FASTECH RS485 통신 접속 커넥터	13
2.3.4 RS485 보오레이트 설정 및 종단저항 선택 스위치 (SW4)	14
2.3.5 상태 표시 LED (LED1..LED4)	15
2.3.6 네트워크 국번 설정 (SW1, SW2)	18
2.3.7 네트워크 통신속도 설정 (SW3)	18
2.3.8 전원 커넥터(DC POWER)	18
3. 설치 및 결선 방법	19
3.1 시스템 구성도	19
3.2 DeviceNet 네트워크의 결선	20
4. 동작 원리	21
4.1 DeviceNet 네트워크의 모션게이트 시스템 개요	21
4.1.1 DeviceNet 의 개요	21
4.1.2 모션게이트의 개요 및 네트워크 구성	23
4.2 제어명령 비트영역(Input Map)과 상태정보 비트영역(Output Map)	26
Input-Map 의 비트 구성	26
Output-Map 의 비트 구성	30

4.3 IO Map 의 동작 순서 및 동작 조건.....	33
4.3.1 IO-Map 의 비트명령 방법.....	33
4.3.2 IO-Map 의 제어 명령 준비 순서.....	35
5. IO-Map 의 사용 예제 (for DeviceNet).....	37
5.1. KGL for Windows 프로젝트 설정.....	37
5.2. PLC LADDER 프로그래밍 예제.....	41
■ 비트 영역 및 데이터 영역.....	41
■ CONNECT.....	42
■ ENABLE 명령과 E-ETOP 명령.....	44
■ ALARM 상태 확인.....	46
■ CANCEL 명령.....	48
■ HOLD 명령.....	50
■ RESPONSE TYPE 설정.....	52
■ 상태 정보 확인.....	56
■ CMD START 명령.....	60
5.2.1 조그 운전 명령.....	62
■ JOG Move – Speed Step Move or Speed Ratio Move.....	62
■ JOG Move – Speed Value Move.....	65
5.2.2 스텝 이동 명령.....	68
5.2.3 영점 이동 명령.....	71
5.2.4 위치 이동 명령.....	74
■ 상대 위치 이동.....	74
■ 절대 위치 이동.....	77
■ 위치 이동 명령에 대한 응답확인.....	79
5.2.5 PT 운전 명령.....	80
■ 일반 PT 운전.....	80
■ 싱글 PT 운전.....	83
■ PT 운전 명령에 대한 응답확인.....	85
5.2.6 원점 이동 명령.....	86
5.2.7 파라미터 설정.....	88
■ 파라미터 확인 명령.....	88
■ 파라미터 변경 명령.....	91

■ 파라미터 저장	94
5.2.8 위치 지정.....	96
5.2.9 알람 내역 확인 및 초기화.....	99
■ 알람 내역 확인	99
■ 알람 내역 초기화	101
5.2.10 모션게이트 버전 확인.....	103

예제 목차

예제 1. 각 축의 활성화 명령 및 응답 비트 확인.....	42
예제 2. 모터 활성화 명령과 비상정지 명령.....	44
예제 3. 알람상태 확인 및 알람해제 명령.....	46
예제 4. 실행취소 명령.....	48
예제 5. Axis - 0 의 일시정지 명령.....	50
예제 6. 응답 데이터 설정.....	52
예제 7. 응답 데이터 설정.....	56
예제 8. 모션 모드 의 CMD START 실행 방법.....	60
예제 9. 설정 모드 의 CMD START 실행 방법.....	61
예제 10. Speed Step Move , Speed Ratio Move 조그 운전 명령.....	62
예제 11. Speed Value Move 형태의 조그 운전 명령.....	65
예제 12. 스텝 이동 명령.....	68
예제 13. 영점이동 명령.....	71
예제 14. 상대위치 이동 명령.....	74
예제 15. 절대위치 이동 명령.....	77
예제 16. PT 운전 명령.....	80
예제 17. 싱글 PT 운전 명령.....	83
예제 18. 원점 이동 명령.....	86
예제 19. 파라미터 확인 명령.....	88
예제 20. 파라미터 변경 명령.....	91
예제 21. 파라미터 저장 명령.....	94
예제 22. 위치값 변경 명령.....	96
예제 23. 알람 내역 확인.....	99
예제 24. 알람 내역 초기화 명령.....	101
예제 25. 모션게이트의 버전 정보 요청 명령.....	103

※ 사용하시기 전에 ※

- 파스텍 Ezi-MOTIONGATE 를 구입해 주셔서 대단히 감사합니다.
- Ezi-MOTIONGATE 는 32bit 고성능 ARM 프로세서를 탑재한 Fieldbus to FASTECH protocol Gateway Unit 입니다.
- 이 사용자 설명서에는 Ezi-MOTIONGATE 의 취급 방법, 안전상의 주의 사항, 이상진단과 처리방법 및 사양 등이 기재 되어있습니다.
- 사용자 설명서를 잘 이해하신 후에 Ezi-MOTIONGATE 를 안전하게 사용하여 주십시오.
- 사용자 설명서를 다 읽으신 후에는 본 제품을 사용하는 사람이 언제든지 볼 수 있도록 잘 보관해 주십시오.



1. 안전상의 주의사항

◆ 일반 주의사항

- 사용자 설명서는 제품 개선이나 사양 변경 또는 사용자 설명서 자체를 이해하기 쉽게 하기 위하여 고지 없이 변경 될 수 있습니다.
- 사용자 설명서를 손상 또는 분실해서 새로 주문할 경우에는 구입하신 대리점이나 본사로 문의해 주십시오.
- 사용자 임의로 제품을 개조하는 것은 당사의 보증 범위 밖이므로 당사에서 책임지지 않습니다.

◆ 안전 주의사항

- 설치, 운전, 점검, 보수 등을 하기 전에는 반드시 사용자 설명서를 읽어서 그 내용을 충분히 숙지하신 후에 실시 해 주십시오, 또한, 기계에 관한 지식, 안전에 관한 정보나, 주의사항을 충분히 숙지하신 후 제품을 사용하여 주십시오.
- 사용자 설명서는 안전에 관한 주의사항의 정도를 **주의**와 **경고**로 구분하여 기재하고 있습니다.

 주의:	잘못 취급했을 경우 위험한 상황을 초래하여 중상 또는 경상을 입을 가능성이 있는 경우, 그리고 대물 손해만이 발생할 가능성이 있는 경우
 경고:	잘못 취급 하였을 경우 전기 감전 등의 위험한 상황을 초래하여, 사망 또는 중상을 입을 가능성이 있는 경우

- 기재된 내용 중에 주의에 해당하는 것일지라도, 상황에 따라서 중대한 결과를 야기시킬 가능성이 있습니다. 반드시 지켜 주십시오.

◆ 제품 상태



제품이 손상되어 있거나 또는 부품이 빠져있는지 확인하십시오.
비정상적인 제품을 설치, 운전할 경우 기계파손 또는 부상의 위험이 있습니다.

◆ 설치



운전 시에는 충분히 주의하십시오
떨어지면 제품이 파손되거나, 발에 떨어지면 부상의 위험이 있습니다.

제품을 취급할 장소에는 금속 등 불연 물질을 사용하여 주십시오.
화재가 날 위험이 있습니다.

여러 대의 Ezi-MOTIONGATE를 하나의 밀폐된 공간에 설치할 때는, 냉각장치 등을 설치하시어 주위 온도가 50°C이하가 되도록 해주십시오.
과열로 화재 또는 그 밖의 사고로 이어질 위험이 있습니다.



설치, 접속, 운전, 조작, 점검 및 고장 진단 작업은 적합한 자격을 가진 사람이 실시하여 주십시오.
화재, 부상, 장치 파손의 원인이 됩니다.

◆ 배선



드라이브의 전원 입력 전압은 정격 범위를 반드시 지켜 주십시오.
화재 및 고장의 원인이 됩니다.

접속은 배선도에 따라 확실히 실시하여 주십시오.
화재 및 오작동의 원인이 됩니다.



입력 전원이 OFF 되어 있는 것을 확인한 후에 작업해 주십시오.
감전 또는 화재의 위험이 있습니다.

본 Ezi-MOTIONGATE 케이스는 콘덴서에 의해 내부회로의 GND와 절연되어 있으므로, 반드시 접지를 시켜주십시오.
감전 또는 화재의 위험이 있으며, 제품 오작동의 원인이 됩니다.

◆ 운전 및 설정 변경

**주의**

드라이브의 보호기능이 작동하면 원인을 제거한 후에 보호 기능을 해제하여 주십시오.

원인을 제거하지 않고 운전을 계속하면 모터 및 드라이브가 오작동되어 부상, 장치 파손의 원인이 됩니다.

드라이브에 전원을 투입할 때에는 드라이브의 제어 입력을 모두 OFF로 한 후에 투입하여 주십시오.

모터가 가동되어 부상, 장치파손의 원인이 됩니다.

본 Ezi-MOTIONGATE 의 모든 값들은 출하 시 적절히 설정해 놓았습니다.

설정 변경 시에는 충분히 사용자 설명서를 숙지한 후 변경해 주십시오.

기계가 파손되거나 제품이 고장 날 수 있습니다.

◆ 보수 및 점검

**경고**

본 Ezi-MOTIONGATE는 주 회로 전원을 차단한 후, 충분히 시간이 경과한 후에 보수, 점검을 해주십시오.

콘덴서 전원이 남아 있으므로, 감전 등의 위험이 있습니다.

통전 중에는 배선 변경을 하지 마십시오.

감전 또는 제품 파손, 기계파손 등의 위험이 있습니다.

제품의 개조는 절대로 하지 마십시오.

감전 또는 제품파손, 기계파손 등의 위험이 있으며, 해당 제품은 당사의 A/S를 받을 수 없습니다.

설치 시 주의사항.

- 1) 실내에서 사용해야 하며, 실내 주의온도는 0°~55°C 에서 사용해야 합니다.
- 2) 케이스가 50°C 이상이 되면 외부에 방열을 시켜주어야 합니다.
- 3) 직사광선, 자석물체, 방사선 물체는 피해서 설치해 주어야 합니다.
- 4) 드라이브를 2대 이상 나란히 설치 시에는 수직방향은 20mm 이상, 수평방향은 50mm 이상 거리를 두고 설치해 주어야 합니다.

2. 제품 사양 및 크기

2.1 개요

- Ezi-MotionGate(이하 모션게이트)는 산업용 네트워크에서 슬레이브로 연결하여, FASTECH RS485 로 구성된 모션 드라이브를 마스터로 제어하는 모션 게이트웨이 장치 입니다.
- 모션게이트를 슬레이브로 연결하여 사용 가능한 가능한 최대 수량은 적용되는 산업용 네트워크에서 지원되는 수량입니다.
- 모션게이트에서 연결 가능한 모터 드라이브(Axis)는 적용되는 산업용 네트워크 마다 최대 15 까지의 모터 ID 를 부여할 수 있습니다.

2.2 네트워크 별 제품 특성 표

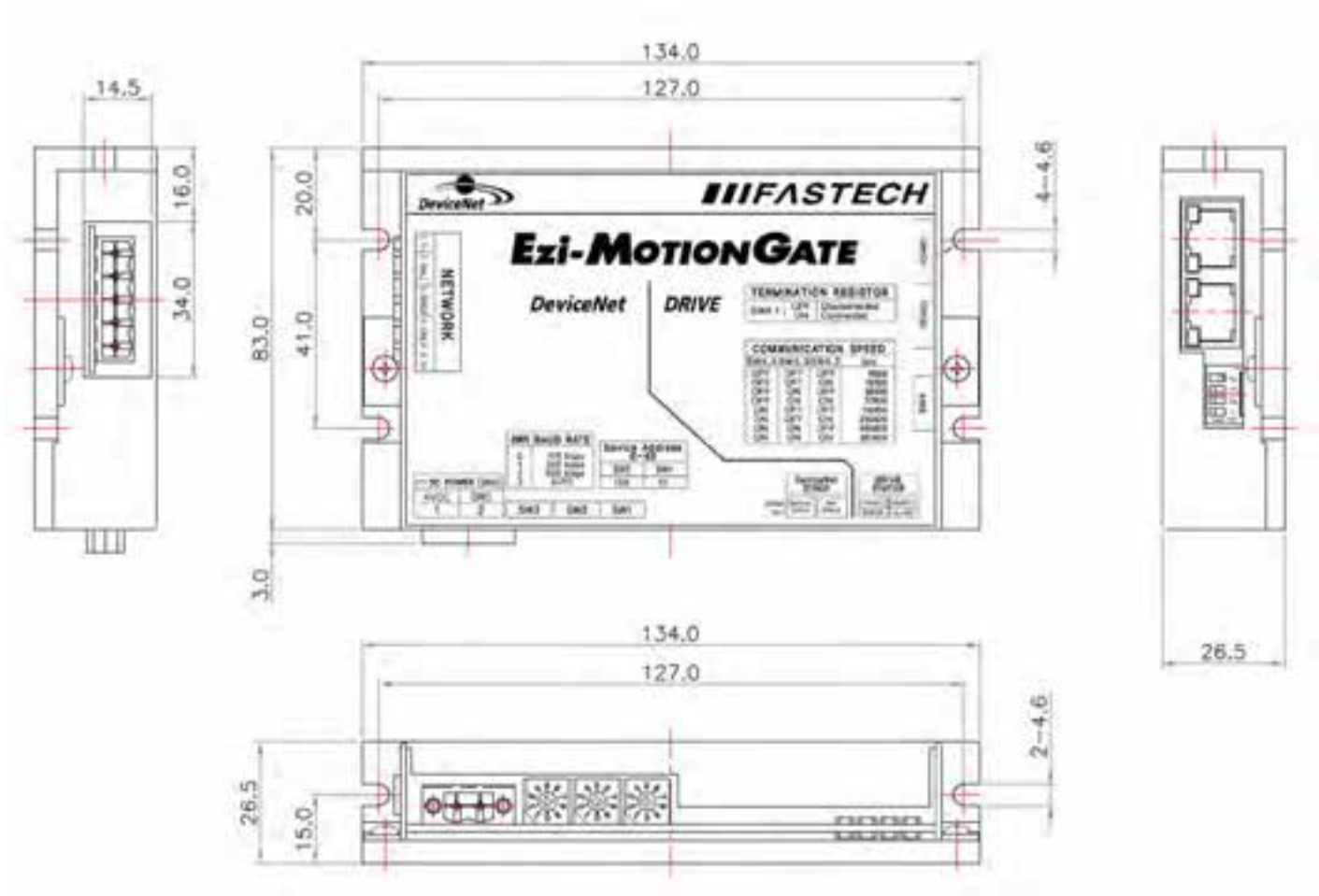
네트워크		필드버스(Fieldbus)								
		DeviceNET								
입력전압		24VDC ±10%								
제어방식		산업용 네트워크의 I/O 데이터를 사용하여 다축 제어가 가능한 모션 게이트웨이								
다 축 제어		1 Station - 16 Axis								
소비전류		최대 500mA								
환경	온도	사용 : 0~55℃ 보관 : -27~70℃								
	습도	사용 : 35~85℃ (결로는 없을 것) 보관 : -10~90℃(결로는 없을 것)								
	내 진동	0.5G								
기능	스위치 선택	네트워크 국번(Station No.) 설정, 네트워크 Baud-Rate 설정								
	LED 표시	네트워크 이상, 마스터 연결 이상, 드라이브의 Servo On 상태, 드라이브의 알람 상태								
특수 기능	조그 제어	4-Speed Step, Speed Ratio								
	스텝 이동 제어	4-Step Distance								
	통신 기능	Ezi-STEP Plus-R, Ezi-SERVO Plus-R series								
FASTECH RS485		Baud-Rate (bps)	9600	19200	38400	57600	115200	230400	460800	921600
		케이블 길이 (m)	1200	1150	1100	1000	1000	880	550	300
		RJ-45 커넥터 LED	YELLOW : RS485 송신 상태 (TX from MOTIONGATE) GREEN : RS485 수신 상태 (RX to MOTIONGATE)							
산업용 네트워크		적용된 네트워크의 사양에 따른 통신속도 및 케이블 길이								

NOTE 1: 케이블의 길이는 네트워크 상태가 최적일 때의 최대 연결거리입니다.

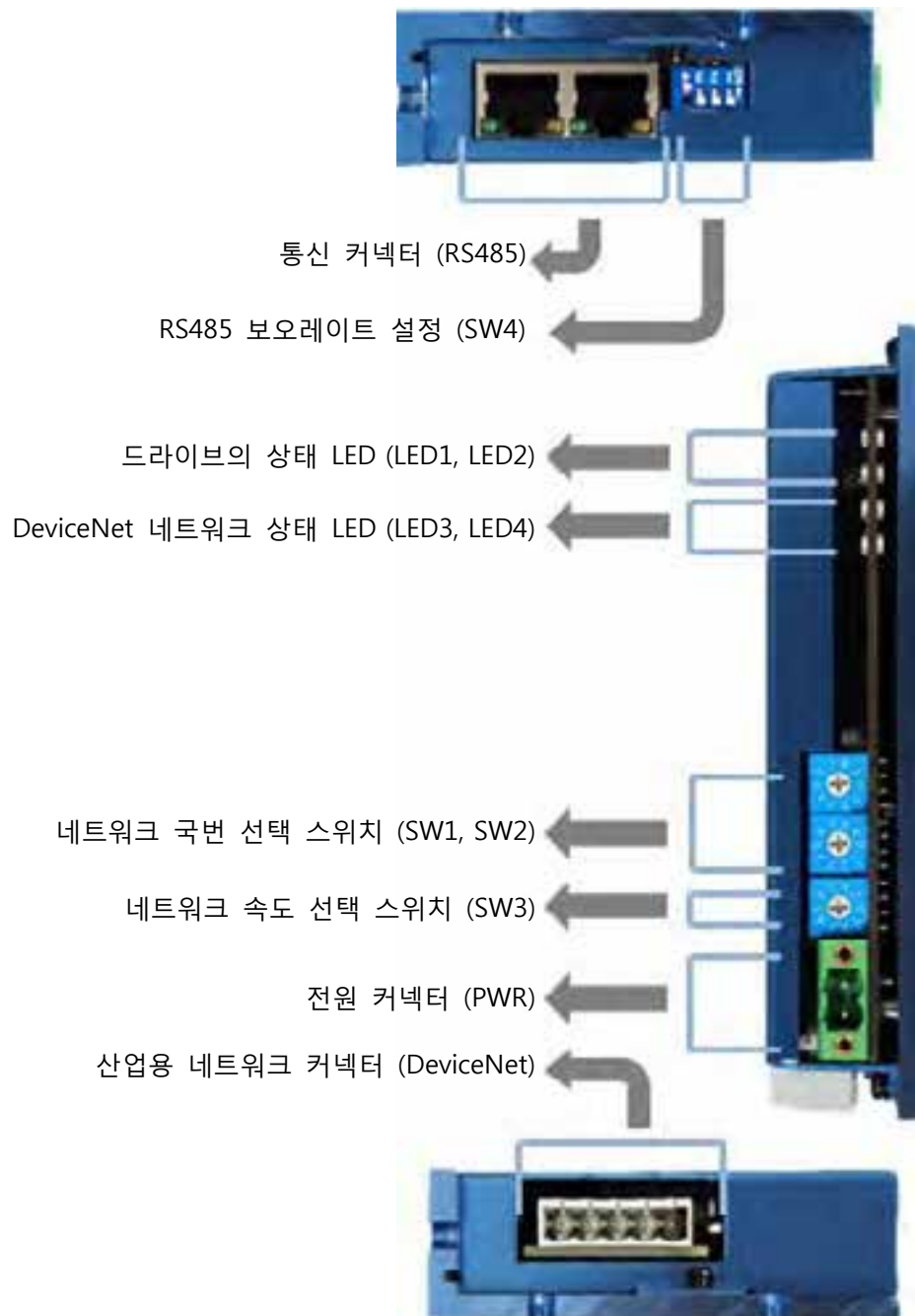
NOTE 2: 산업용 네트워크의 통신규약을 준수 합니다.

2.3 제품 외형

2.3.1 제품 치수

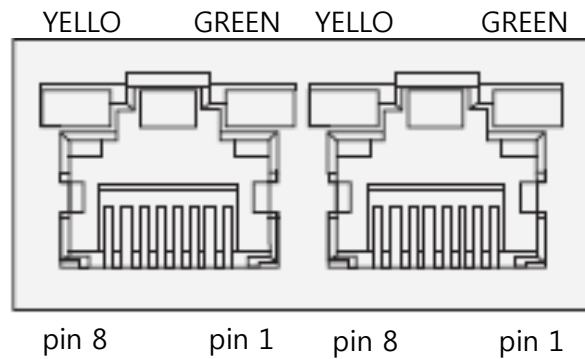


2.3.2 각 부의 명칭



2.3.3 FASTECH RS485 통신 접속 커넥터

통신 커넥터는 RJ-45 를 사용하여 연결합니다.



통신 커넥터의 핀 맵 (RS485)

핀 번호	기능
1, 2, 4, 5, 7, 8	GND
3	DATA +
6	DATA -
CASE	Frame GND


통신 커넥터의 LED

디스플레이	색상	발광 형태
RS485 TX	녹색	RS485 데이터를 송신 시 점멸
RS485 RX	황색	RS485 데이터를 수신 시 점멸

2.3.4 RS485 보오레이트 설정 및 종단저항 선택 스위치 (SW4)

SW4 는 모터 드라이브와 연결된 RS-485 통신 네트워크의 보오레이트(Baud-Rate : 통신속도) 설정을 위한 스위치 입니다. 만약 모션게이트가 네트워크 segment 의 가장 끝 단에 연결된 상태일 경우, 종단 저항을 사용 여부를 결정할 수 있습니다.

SW4.1 은 종단 저항의 사용을 결정하며, SW4.2~SW4.4 는 다음과 같이 통신속도를 설정 하는데 사용됩니다.

SW4.1	SW4.2	SW4.3	SW4.4	Speed baud[bps]	<p>*1 : 초기 설정 값</p>  <p>SW4.1 가 OFF : 종단 저항이 OFF 상태. SW4.1 가 ON : 종단 저항이 ON 상태.</p>
X	OFF	OFF	OFF	9600	
X	ON	OFF	OFF	19200	
X	OFF	ON	OFF	38400	
X	ON	ON	OFF	57600	
X	OFF	OFF	ON	115200 *1	
X	ON	OFF	ON	230400	
X	OFF	ON	ON	460800	
X	ON	ON	ON	921600	

**주의**

하나의 RS485 네트워크에 연결된 드라이브 모듈들의 통신 속도 설정 값은 모두 동일한 값으로 지정해 주어야 합니다.

2.3.5 상태 표시 LED (LED1..LED4)

상태 표시 LED 는 상황에 따라 LED1~LED4 를 동시에 동작 하거나, 단독적으로 동작하여 상태를 표시합니다.

LED 동작 표시 (LED1~LED4)

LED 번호	동작 상태	설명
LED1 LED2 LED3 LED4	소등	전원 OFF, 타임 아웃 상태, 네트워크 미 접속
	녹색, 적색 동시 점등	모션게이트에 전원공급으로 부팅되는 상태 국번 지정 스위치 또는 DeviceNet 보오레이트 선택 스위치를 변경 하여 재 부팅되는 상태 * 녹색과 적색이 동시 점등하므로 LED 의 색상은 주황색으로 보입니다.
	녹색 동시 점멸	모션게이트의 자기 진단 상태. ✓ 커넥터 미 연결 ✓ 잘못된 네트워크 보오레이트 설정 ✓ 잘못된 네트워크 국번 지정
LED3 LED4	녹색, 적색 동시 점등	모션게이트의 네트워크 장치가 인식 불가 상태 * 본사 또는 대리점으로 연락하여 조치를 취하십시오.

드라이브 상태 표시 LED (DRIVE STATUS)

LED 번호	LED 상태 정보	LED 이름	동작 상태	설명
LED1	드라이브의 접속 상태	ENABLE (녹색)	점등	CONNECT 된 드라이브 중 모터가 활성화 됨.
			점멸	CONNECT 된 드라이브 중 1 개 이상의 드라이브의 모터가 ENABLE 되지 않음
	드라이브 알람	ALARM (적색)	점멸	1 개 이상의 드라이브가 알람을 발생한 상태
LED2	모터 드라이브 연결 양호	CONNECT (녹색)	점등	모터 드라이브가 CONNECT 명령이 실행되어 모션게이트와 정상적으로 통신 중
		ERROR (적색)	소등	
	드라이브 접속 상태 오류	CONNECT (녹색)	소등	RS-485 네트워크에 연결된 모터 드라이브가 CONNECT 명령이 실행된 모터 드라이브가 없음.
		ERROR (적색)	점등	모터 드라이브와 모션게이트가 통신중인 상태가 아님. 드라이브와 통신되지 않음 RS485 네트워크의 통신연결이 끊김 보오레이트 설정 오류
	통신 에러	CONNECT (녹색)	점등	모터 드라이브와 통신 오류 발생 (CRC 에러 발생)
		ERROR (적색)	무작위 점멸	
	다축 연결 상태의 통신 에러	CONNECT (녹색)	점등	RS485 에 연결된 하나 이상의 모터드라이브에 CONNECT 명령에 응답 하지 않음. ✓ 네트워크의 연결이 끊김 ✓ 토폴로지 구성에 이상이 있음. ✓ 없는 모터드라이브의 IO-Map 영역에 CONNECT 명령을 실행함.
		ERROR (적색)	점등	

**주의**

드라이브 상태 표시 LED는 모션게이트와 드라이브의 통신 상태를 점검 후 모터
활성화 상태를 확인하도록 하십시오.

DeviceNet 네트워크 상태 표시 LED (DeviceNet STATUS)

번호	LED 정보	동작 상태	내용	조치
LED 3	NS (Network Status)	OFF	DeviceNET 전원 미 공급	DeviceNet 네트워크에 24V 전원이 인가되지 않습니다. 네트워크 케이블 상태를 확인 하십시오
		녹색	네트워크 정상 연결	모션게이트와 상위제어기 간의 통신상태가 정상입니다.
		녹색 점멸 (1Hz)	네트워크 설정 오류	DeviceNet 의 파라미터의 설정을 확인 하십시오.
		적색	통신 실패	DeviceNet 네트워크 케이블 및 커넥터의 연결 상태를 확인 하십시오.
		적색 점멸 (1Hz)	타임 아웃	DeviceNet 커넥터가 연결되어있으나, 마스터와의 통신이 되지 않습니다. 상위제어기의 상태를 확인 하십시오
		녹/적 교차 점멸	자가 진단	DeviceNet 네트워크가 진단 상태 입니다.
LED 4	MS (Module Status)	OFF	전원 미 공급	모션게이트의 전원을 확인 하십시오
		녹색	정상 동작	모션게이트가 정상 상태입니다.
		녹색 점멸	Srandby 상태	모션게이트가 통신 속도 점검 및 상위제어기에 접속 허용을 기다리는 상태입니다.
		적색	복구 불능한 에러 발생	모션게이트의 고장이 의심 되므로, 대리점 또는 본사에 수리의뢰를 하십시오
		적색 점멸	복구 가능한 에러 발생	네트워크의 토폴로지와 상위 제어기의 파라미터를 확인하십시오.
		녹/적 교차 점멸	자가 진단	모션게이트가 진단 상태입니다.

**주의**

네트워크 상태 LED는 산업용 네트워크의 규약으로 인하여 네트워크의 상태에 따라 정해진 표현으로 동작합니다.

2.3.6 네트워크 국번 설정 (SW1, SW2)


DeviceNet 네트워크의 국번(Station No.)를 설정하는 로터리 스위치로, 네트워크에서 설정 하고자 하는 국번으로 지정 할 수 있습니다. DeviceNet 에서 지원하는 국번의 범위는 0~63 입니다.

스위치 값 (SW1)	ID번호 X10 (10자리)	스위치 값 (SW2)	ID번호 X1 (1자리)	
0	00	0	0	
1	10	1	1	
2	20	2	2	
3	30	3	3	
4	40	4	4	
5	50	5	5	
6	60	6	6	
7	N.C	7	7	
8	N.C	8	8	
9	N.C	9	9	

2.3.7 네트워크 통신속도 설정 (SW3)

사용되는 네트워크의 보오레이트를 설정하는 스위치 입니다. 상위 제어기의 네트워크 설정 값과 동일하도록 설정 합니다.

네트워크 통신속도 설정 (DeviceNet)

스위치 값 (SW3)	BAUD RATE	
0	125 kbps	
1	250 kbps	
2	500 kbps	
3	Auto Baud-rate	
4..9	N.C	

2.3.8 전원 커넥터(DC POWER)

전원 공급을 위한 커넥터 입니다.

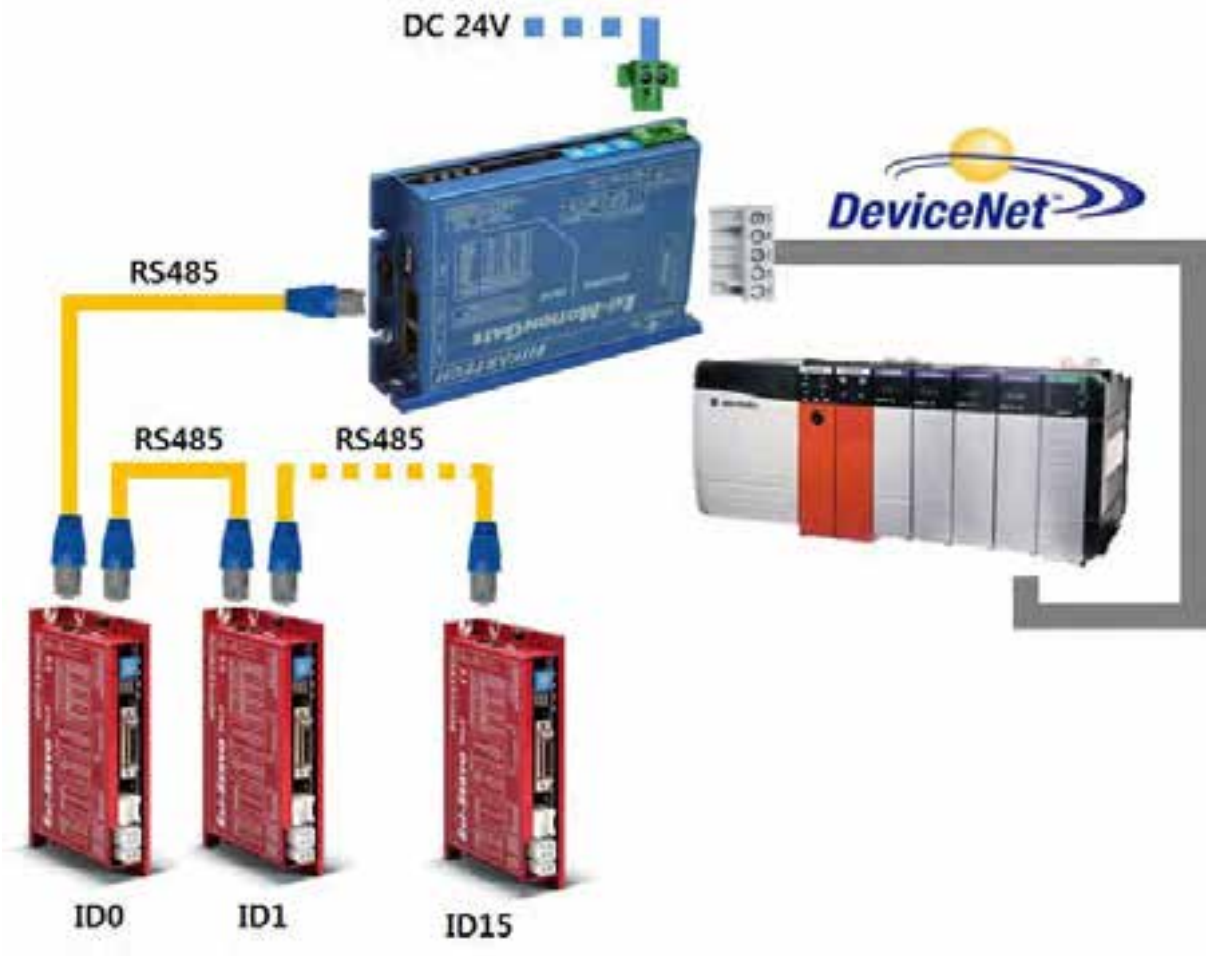
번호	기능	핀 배치도
1	입력전원 : 24VDC \pm 10%	
2	입력전원 : GND	

3. 설치 및 결선 방법

3.1 시스템 구성도





모션게이트의 배선도



3.2 DeviceNet 네트워크의 결선

DeviceNet 네트워크 케이블의 구조

DeviceNet 네트워크 케이블	DeviceNet 커넥터의 핀 맵												
 <table border="1" data-bbox="279 772 750 1120"> <thead> <tr> <th>심선 종류</th><th>데이터 명칭</th></tr> </thead> <tbody> <tr> <td>흑색</td><td>GND</td></tr> <tr> <td>적색</td><td>24V</td></tr> <tr> <td>백색</td><td>CAN_H</td></tr> <tr> <td>청색</td><td>CAN_L</td></tr> <tr> <td>드레인 와이어 또는 편조 실드</td><td>DRAIN</td></tr> </tbody> </table>	심선 종류	데이터 명칭	흑색	GND	적색	24V	백색	CAN_H	청색	CAN_L	드레인 와이어 또는 편조 실드	DRAIN	
심선 종류	데이터 명칭												
흑색	GND												
적색	24V												
백색	CAN_H												
청색	CAN_L												
드레인 와이어 또는 편조 실드	DRAIN												

⚠ 주의

모션게이트의 전원은 DeviceNet 케이블의 버스 전원과 독립적으로 사용됩니다. 만약 DeviceNet 커넥터의 전원만 공급될 경우 모션게이트의 전원은 ON 상태가 되지 않습니다.

또한 모션게이트의 네트워크 커넥터에 DeviceNet 버스 전원을 공급하지 않을 경우 동작 되지 않습니다.

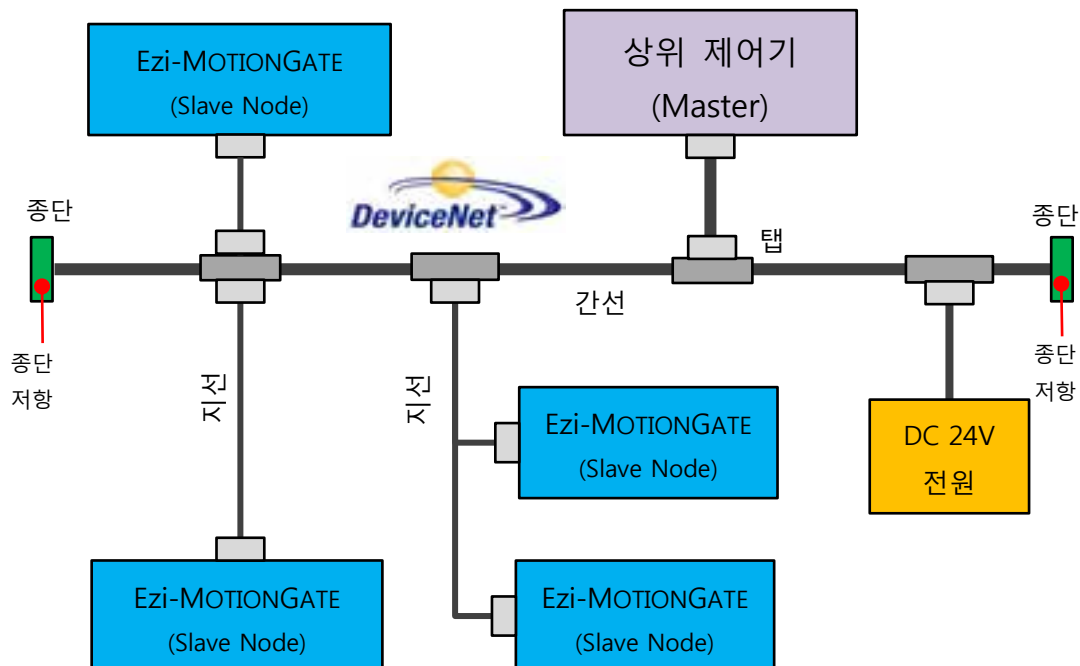
4. 동작 원리

4.1 DeviceNet 네트워크의 모션게이트 시스템 개요

4.1.1 DeviceNet 의 개요

DeviceNet 는 Allen Bradley 에 의하여 개발 되었으나, 현재 네트워크에 사양과 프로토콜이 공개되어 있으면 장치를 시스템에 접속하기 위하여 H/W 또는 S/W 의 라이선스를 구입할 필요가 없는 오픈 된 네트워크 규격 입니다.

DeviceNet 의 네트워크 장치는 4 선식 DeviceNet 케이블을 사용하여 CAN 구조의 데이터링크와 전원에 대한 연결이 허용되는 네트워크 토폴로지를 지닙니다. 또한 DeviceNet 의 네트워크는 3 개의 보오레이트(125kbps, 250kbps, 500kbps)로 최대 64 개의 노드를 지원 합니다.



하나의 DeviceNet 의 물리 구조는 간선에 네트워크 탭을 연결 하여 버스형태, Point-to-Point 연결하는 스타 형태, 그리고 망 구조 형태의 토폴로지를 구성 할 수 있습니다. 이러한 특징으로 현장 정보의 신속한 상위 연결이 가능하여 생산성이 향상되는 효과가 있습니다. 또한 단순화된 네트워크 토폴로지로 인하여 설치 및 유지보수가 간편합니다.

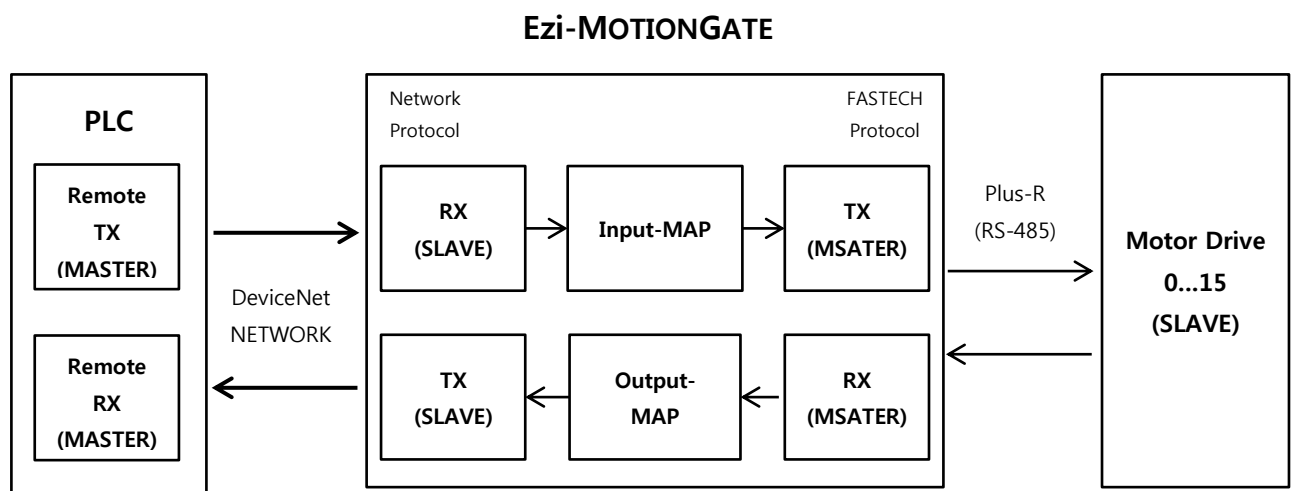
모션게이트의 DeviceNet 통신 사양

항목	사양			
리모트 송신 디바이스	128 byte (64 WORD)			
리모트 수신 디바이스	128 byte (64 WORD)			
통신 모드	Poll			
허용 국번 범위	0 ~ 63 국 (마스터 국번을 제외)			
통신속도와 케이블 연장 거리	케이블 유형	데이터 속도		
		125 kbps	250 kbps	500 kbps
	라운드 두꺼운 트렁크 케이블	500m	250m	100m
	라운드 얇은 트렁크 케이블	100m	100m	100m
	평면 트렁크 케이블	380m	200m	75m
	최대 드롭 길이	6m	6m	6m
	최대 누적 드롭 길이	156m	78m	39m

4.1.2 네트워크 구성

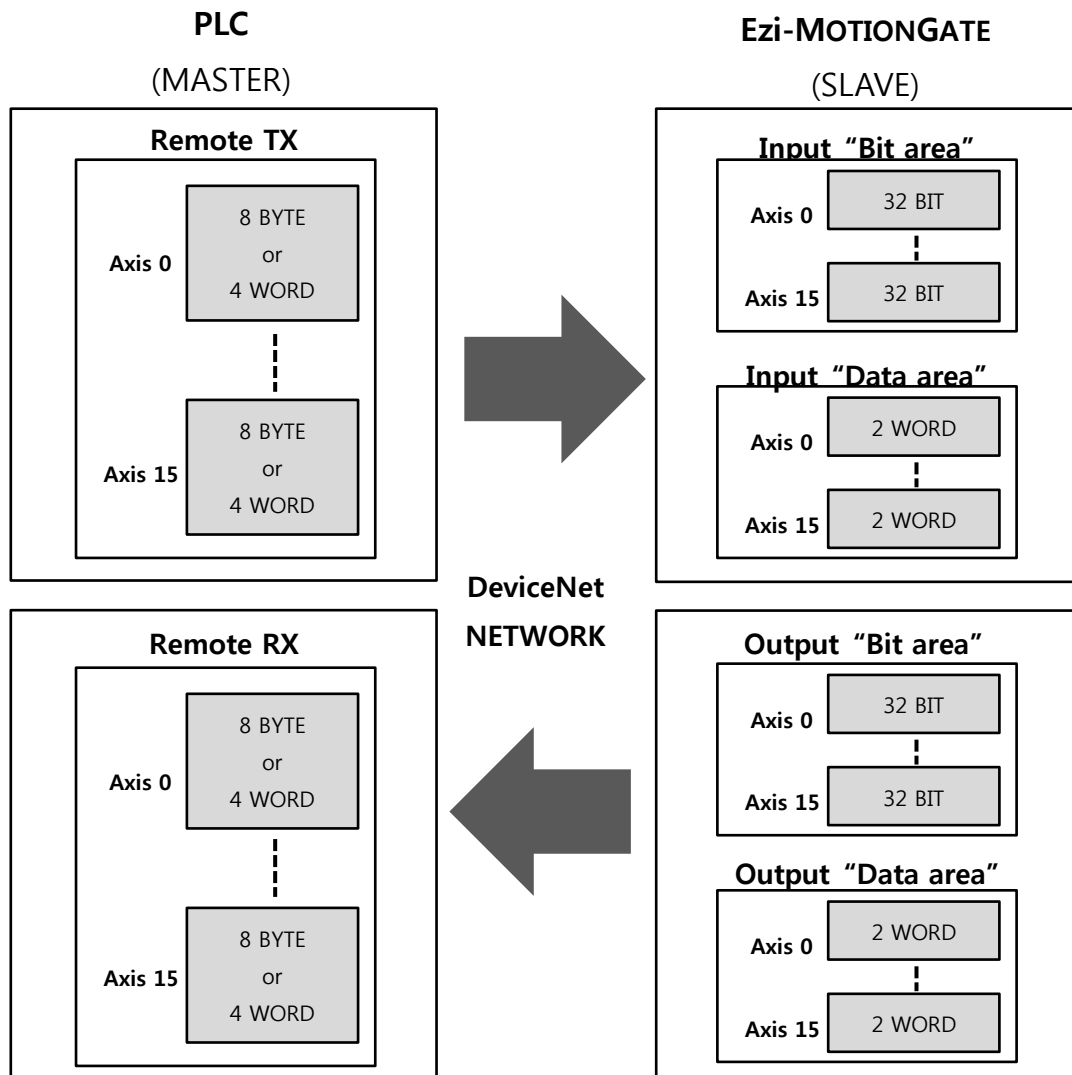
DeviceNet 네트워크에서 사용하는 모션게이트는 최대 16 개(Axis 0~15)의 드라이브를 지원합니다. (각 네트워크를 지원하는 모션게이트에 따라 달라집니다. 네트워크 별 지원 가능한 수량 확인은 [*2.2 네트워크 별 제품 특성표](#) 에서 확인할 수 있습니다.)

상위 제어기(PLC)는 모션게이트와 연결된 송수신 구간의 메모리 주소를 액세스 가능한 마스터 시스템이 되어야 합니다. DeviceNet 상위제어기의 리모트 데이터를 모션게이트가 CC-Link 네트워크로부터 수신하여, 수신된 데이터의 Input-Map 에 해당하는 축을 제어합니다. 각 축에서의 응답정보는 Output-Map 으로 구성하여 DeviceNet 네트워크로 리모트 데이터를 송신합니다.



DeviceNet 네트워크에서의 모션게이트에 연결된 각 모터드라이브에 대한 명령은 IO-Map 으로 구성된 데이터를 사용하여 명령 및 정보 요청을 합니다. 여기서 IO-Map 의 어드레스 영역은 각 축에 대한 제어 명령, 응답 정보 확인을 위한 영역입니다. IO-Map 의 데이터 구성은 한 축에 대하여 32bit 로 구성된 비트영역(Bit area)와 2 WORD(4 Byte)로 구성된 데이터영역(Data area)으로 나뉩니다.

상위 제어기(PLC)는 IO-Map 의 데이터 어드레스 영역으로 제어와 응답 데이터를 확인 할 수 있습니다.

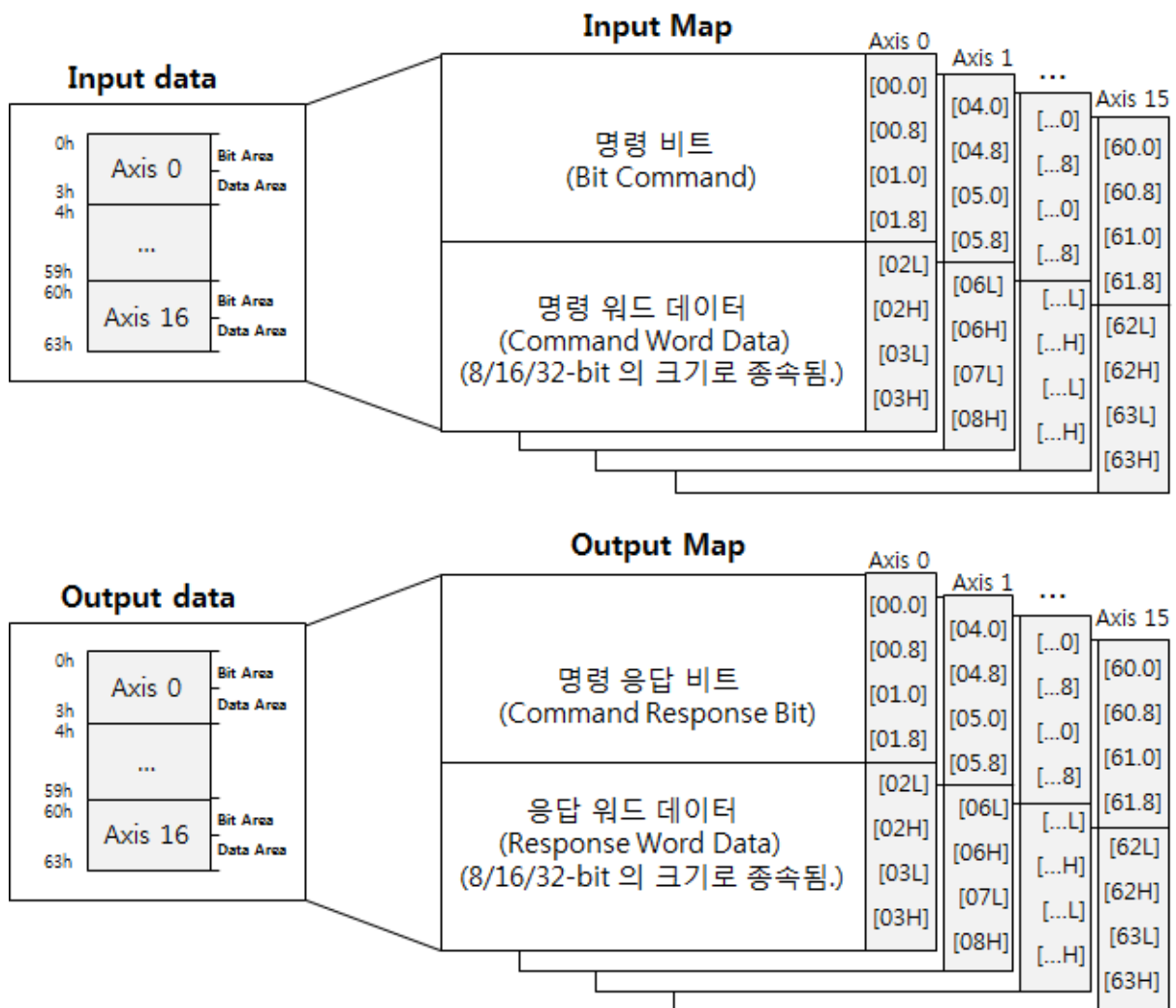


모션게이트의 IO-Map 데이터 어드레스 영역은 상위 제어기의 리모트 송수신 디바이스를 릴레이 접점으로 지정 시 128 바이트를 점유 하고, 리모트 송수신 데이터 레지스터 디바이스로 지정 시 64 워드를 점유합니다.(한 축에 8 바이트 또는 4 워드를 점유합니다.)

IO-Map 은 비트영역[0-3]과 데이터영역[4-7]으로 나누어 집니다.

Input-Map 의 비트영역은 모터축의 비트명령을 위한 구간으로 사용되며, 데이터영역은 비트명령에 해당하는 데이터 정보를 입력하는 구간으로 사용됩니다. Output-Map 의 비트영역은 해당 축의 상태플래그 또는 제어 명령에 대한 응답 비트로 사용되고, 워드영역은 Input-Map 의 명령으로 요청한 데이터가 지정 되는 구간입니다.

IO-Map 의 구조는 맨 처음의 주소에서 4 번째의 주소까지의 바이트 영역은 0 번축에 대한 IO-Map 이고 뒤로 1 번축의 IO-Map 으로 이어져 15 번축의 송신구간으로 8 바이트 간격으로 이어집니다.



NOTE 1: 상위 제어기의 디바이스 메모리 1WORD 는 16Bit 데이터로 2byte 의 크기를 갖고 있습니다. 그러므로 디바이스 메모리의 주소 0h 의 00.0~00.7 영역은 IO-Map 의 0Byte 영역을 점유하고, 00.8~00.15 는 1Byte 영역을 점유합니다.

NOTE 2: 데이터 영역이 사용하는 디바이스 점유 메모리는 2WORD 입니다. 따라서 데이터영역의 시작 주소에 대한 DWORD 주소를 사용할 수 있습니다.

4.2 제어명령 비트영역(Input Map)과 상태정보 비트영역(Output Map)

Input-Map의 비트 구성

Input-Map 은 모터 드라이브의 제어를 지령 하는 영역입니다. 명령에 대한 비트의 조합으로 모터 드라이브의 모션 제어의 선택 및 응답 정보 형식 설정, 파라미터 또는 PT 정보 등의 값을 설정할 수 있습니다.

Input-Map의 비트 영역(상위 4 바이트 영역)

BYTE offset	BIT	비트 이름	동작 레벨	Description
0	0	CONNECT	상승 엷지	<p>이 비트의 설정으로 해당 축의 사용여부를 결정하며, '1'로 세트 하였을 때 해당 축과 통신연결을 시도하며, 해당 축과 통신이 필요하지 않을 경우 이 비트의 설정을 '0'으로 설정합니다. '0'으로 설정 되었을 때 모션게이트는 해당 축과의 통신처리는 제외되며, 어떠한 명령도 실행 하지 않습니다.</p> <p>여러 축에 대한 명령이벤트가 동시에 발생되었을 시 모터 드라이브의 번호가 낮은 순에서 높은 순으로 처리가 됩니다. 한 축에 대한 이벤트의 처리가 완료되었다면, 다음 ID의 축에 대한 프로세스 작업으로 변경 됩니다.</p> <p>해당된 축에서 명령 또는 이벤트가 없을 경우, MotionGate 는 해당 축의 상태정보와 응답요청 항목에 대한 데이터를 수신 합니다.</p> <ul style="list-style-type: none"> - 해당 축의 상태정보(flags FLAG-define) - 지령 위치 값 (signed long 32-bit) - 실제 위치 값 (signed long 32-bit) - 위치 오차 값 (signed long 32-bit) - 현재 운전 속도 값 (signed long 32-bit) - 현재 운전중인 PT 번호 <p>NOTE 1: 모션게이트는 모터드라이브와 수시로 Fas_GetAllStatus() 함수의 명령을 실행 합니다.</p> <p>NOTE 2: 모터제어의 지연시간은 동일한 지연시간을 갖는 모터일 경우 연결 축수에 대한 배가 됩니다.</p>

BYTE offset	BIT	비트 이름	동작 레벨	Description
	1	ENABLE / IGNORED	상승 엷지	SERVO Drive : 해당 축의 상태를 모션동작이 가능한 상태로 전환합니다. 0 : Servo OFF 1 : Servo ON STEP Drive : 이 비트의 명령은 무시 됩니다.
	2	nESTOP	하강 엷지	모션 또는 모든 명령 실행 정지. (비상 정지) * 0: E-Stop 명령 실행, 1: E-Stop 명령 대기
	3	ALARM_RESET / MOTOR_FREE	상승 / 하강 엷지	SERVO Drive : 발생된 알람을 해제 할 때 사용 (상승 엷지 동작) * STEP Drive 는 MOTOR_FREE 비트를 '1'로 유지할 경우 STEP 드라이브는 Motor Free 상태가 유지되고, '0'으로 전환되는 하강엷지에서 스텝모터 알람 리셋 명령이 실행 됩니다.
	4	CMD_START	상승 엷지	위치이동 또는 PT 운전, 원점이동의 명령을 실행 시 사용
	5	-	-	-
	6	-	-	-
	7	MOTION / SETTING	H/L	MotionGate 의 Map 의 구성을 모션 및 설정으로 선택하는 비트 0: 모션제어 모드 1: 설정 모드
1	0	CMD_CODE0	H/L	모션 제어 모드일 때 0000(0): 일반 모션(Jog, Step, 영점 이동) 0001(1): 상대치 이동[Incrometal Move], 절대치 이동[Absolute Move] 0100(4): PT 운전 (PT 운전, 싱글 PT 운전) 0111(7): 원점 이동 (Origin) 설정 모드 일 때 0000(0): 명령 없음 0101(5): 버전 정보 확인 1000(8): 파라미터 요청 1001(9): 파라미터 쓰기 1010(10): 위치정보 변경 1100(12): 알람내역 요청 1101(13): 알람내역 삭제 1110(14): 파라미터 저장
	1	CMD_CODE1		
	2	CMD_CODE2		
	3	CMD_CODE3		

BYTE offset	BIT	비트 이름	동작 레벨	Description
	4	RESPONSE_TYPE0	H/L	<p>해당 축에 대한 RX 구간으로부터 수신을 원하는 응답데이터의 응답형식을 지정</p> <p>0000(0): 응답데이터를 요청하지 않음.</p> <p>0001(1): 지령 위치</p> <p>0010(2): 실제 위치</p> <p>0011(3): 위치 오차</p> <p>0100(4): 현재 속도</p> <p>0101(5): 운전중인 PT 번호</p> <p>1000(8): 현재 발생된 알람 번호</p> <p>*설정 모드에서는 사용되지 않습니다.</p>
	5	RESPONSE_TYPE1		
	6	RESPONSE_TYPE2		
	7	RESPONSE_TYPE3		
2	0	CANCEL	상승 엷지	모션에 대한 일반 정지
	1	HOLD	상승 엷지	모션 중 일시 정지
	2	-	-	-
	3	GO_ZERO_POS	상승 엷지	해당 축의 드라이브에서 지정된 영점으로 이동 (위치 값: 0)
	4	-JOG_MOV	상승 엷지	역방향 조그(JOG)운전 데이터 영역의 입력 값: 속도 비율, 속도 값, Speed Step 번호.
	5	+JOG_MOV	상승 엷지	정 방향 조그(JOG)운전 데이터 영역의 입력 값: 속도 비율, 속도 값, Speed Step 번호.
	6	-STEP_MOV	상승 엷지	<p>MotionGate 의 내부 파라미터 값인 위치 값과, 속도 값으로 가감이동</p> <p>데이터 영역의 입력 값: 위치 값의 번호(0~3)</p> <p>*사용자에 의하여 재 정의 가능</p>
	7	+STEP_MOV	상승 엷지	<p>MotionGate 의 내부 파라미터 값인 위치 값과, 속도 값으로 증감이동</p> <p>워드영역의 입력 값: 위치 값의 번호(0~3)</p> <p>*사용자에 의하여 재 정의 가능</p>

BYTE offset	BIT	비트 이름	동작 레벨	Description
3	0	INC/ABS	H/L	제어 방법이 위치이동일 때(CMD_CODE:0001) 상대치 이동 또는 절대치 이동을 선택하는 비트 0: 상대치 이동 1: 절대치 이동
	1	-	-	-
	2	SPD_MODE	H/L	제어 방법이 일반 모션일 때(CMD_CODE: 0000) Jog 이동 시 사용 0: 입력된 비율 값 또는 Speed Step 번호로 Jog 운전 1: 입력된 속도 값으로 Jog 운전
	3	-	-	-
	4	SINGLE_PT	H/L	제어 방법이 PT 운전일 때(CMD_CODE:0100), 일반 PT 운전, 또는 싱글 PT 운전을 선택하는 비트. 0: 일반 PT 운전 1: 싱글 PT 운전
	5	-	-	-
	6	-	-	-
	7	-	-	-

NOTE 1: Input-Map 은 모션게이트가 상위제어기로 명령을 입력 받는 영역 입니다.

NOTE 2: 위의 Input-Map 은 제어 명령 영역으로 상위 4 바이트에 대한 정보입니다.

NOTE 3: Input-Map 의 하위 4 바이트는 제어 명령에 대한 DWORD 데이터 입력 구간 입니다.

Output-Map의 비트 구성

Output-Map의 구간은 데이터 플래그와 비트명령에 대한 루프-백(Loop-Back: 되돌림)비트가 존재합니다. 루프-백 비트는 해당 비트의 명령 이벤트와 동일하게 반응하는 비트로 Input-Map의 비트 입력 여부를 확인할 수 있습니다. 상태 플래그는 해당 모터드라이브와 통신하여 수신된 데이터 정보를 기반으로 나타냅니다.

Output-Map의 비트 영역(상위 4 바이트 영역)

Byte offset	bit	비트 이름	동작 레벨	비트 유형	Description
0	0	CONNECTED	H	상태 비트	해당 축의 Plus-R과 연결되었을 경우 이 비트는 '1'로 세트
	1	ENABLED MOTOR_FREE_RESP (STEP)	H	상태 비트	해당축의 Servo ON 또는 스텝모터의 노말 상태일 때 '1'로 세트 *STEP Drive 일 때, Motor Free 명령에 대한 응답 비트가 됨.
	2	ESTOP_RESP	H	루프-백	Input-Map의 nESTOP 비트의 루프-백 비트로 비상정지 명령이 실행되면 '1'로 세트
	3	ALARM_ERROR	H	상태 비트	해당 축의 모터 드라이브에서 알람이 발생되었을 때 자동으로 '1'로 세트, 알람이 해제되면 '0'으로 클리어
	4	CMD_RESP	H	루프-백	Input-Map의 CMD_START 비트의 루프-백 비트
	5	OUT_RANGE	H	상태 비트	Input-Map의 데이터영역의 값이 해당 명령의 값의 범위에 맞지 않았을 때 '1'로 세트
	6	READY	H	상태 비트	현재 해당 축에 대한 명령이 실행 가능한 상태일 때 '1'로 세트 됩니다. 이 비트가 '0'인 상태에서 어느 명령도 동작하지 않습니다. NOTE 1: 세팅 모드에서 READY 비트가 '1'로 세트 되었을 때, 다른 축의 제어가 가능합니다.
	7	SET_MOV_RESP	H/L	루프-백	현재 Output-Map의 데이터가 세팅 모드일 때는 '1'로 세트 되고, 모션모드일 때 '0'으로 클리어

Byte offset	bit	비트 이름	동작 레벨	비트 유형	Description
	0	CMD_CODE_RESP0	H/L	루프-백	명령 받은 모션의 종류를 응답 0000(0): 일반 모션(Jog, Step, 영점 이동) 0001(1): 상대이동[Incrometal Move], 절대값이동[Absolute Move] 0100(2): PT 운전 (PT 운전, 싱글 PT 운전) 0111(3): 원점 이동 (Origin)
	1	CMD_CODE_RESP 1	H/L		
	2	CMD_CODE_RESP 2	H/L		
	3	CMD_CODE_RESP 3	H/L		
	4	RESPONSE_TYPE_RESP0	H/L	루프-백	Word 영역에 지정되는 되는 응답데이터의 정보를 응답 0000(0): 응답데이터를 요청하지 않음. 0001(1): 명령위치 0010(2): 실제 위치 0011(3): 위치 오차 0100(4): 현재 속도 0101(5): 운전중인 PT 번호 1000(8): 현재 발생된 알람 번호
	5	RESPONSE_TYPE_RESP1	H/L		
	6	RESPONSE_TYPE_RESP2	H/L		
	7	RESPONSE_TYPE_RESP3	H/L		
2	0	MOTIONNING	H/L	상태 비트	해당 축의 모션상태일 때 '1'로 세트
	1	HOLD_RESP	H/L	상태 비트	운전 중 HOLD 비트의 명령으로 일시 정지된 상태일 때 '1'로 세트
	2	-	-	-	-
	3	GO_ORIGIN_RESP	H	상태 비트	해당 축의 Plus-R 의 파라미터로 원점복귀를 실행 중 일 때, '1'로 세트
	4	-	-	-	-
	5	JOG_RESP	H	상태 비트	해당 축이 Jog 운전 중일 때
	6	-	-	-	-
	7	STEP_RESP	H	상태 비트	해당 축이 Step 운전 중일 때

Byte offset	bit	비트 이름	동작 레벨	비트 유형	Description
	0	PT_RUNNING	L/H	상태 비트	해당 축이 위치이동 중일 때
	1	MOV DIR	L/H	상태 비트	모터의 회전방향을 표시 0 : CW(+) 1 : CCW(-) * 이 비트를 확인 시 FLAG_IN_MOTION 비트가 '1'로 세트 되었을 때 갱신된 값으로 확인해야 합니다. 논리연산 (FLAG_IN_MOTION & FLAG_nDIR)
	2	INP	L/H	상태 비트	모터가 In position 이 완료되었을 때 '1'로 세트 됩니다. * 모터가 STEP 일 경우 이 비트는 동작되지 않습니다.
	3	ORIGIN_SENSOR	H	상태 비트	원점센서가 ON 이 된 경우 '1'로 세트 됩니다.
	4	SW_LIMIT_N	H	상태 비트	'-' 방향 프로그램 리미트를 초과한 경우 '1'로 세트 됩니다.
	5	SW_LIMIT_P	H	상태 비트	'+' 방향 프로그램 리미트를 초과한 경우 '1'로 세트 됩니다.
	6	HW_LIMIT_N	H	상태 비트	'-' 방향 리미트 센서가 ON 이 된 경우 '1'로 세트 됩니다.
	7	HW_LIMIT_P	H	상태 비트	'+' 방향 리미트 센서가 ON 이 된 경우 '1'로 세트 됩니다.

NOTE 1: Output-Map 은 모션게이트가 상위제어기에 응답 정보를 출력 하는 영역 입니다.

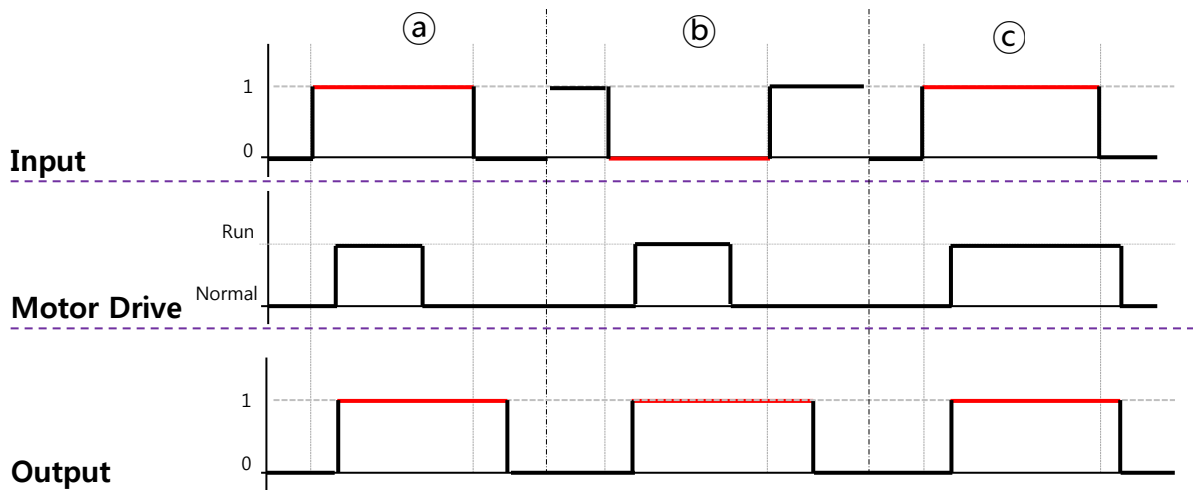
NOTE 2: 위의 Output-Map 은 상태 응답 비트 영역으로 상위 4 바이트에 대한 정보입니다.

NOTE 3: Output-Map 의 하위 4 바이트는 응답 데이터에 대한 DWORD 데이터로 수신되는 구간 입니다.

4.3 IO Map의 동작 순서 및 동작 조건

4.3.1 IO-Map의 비트명령 방법

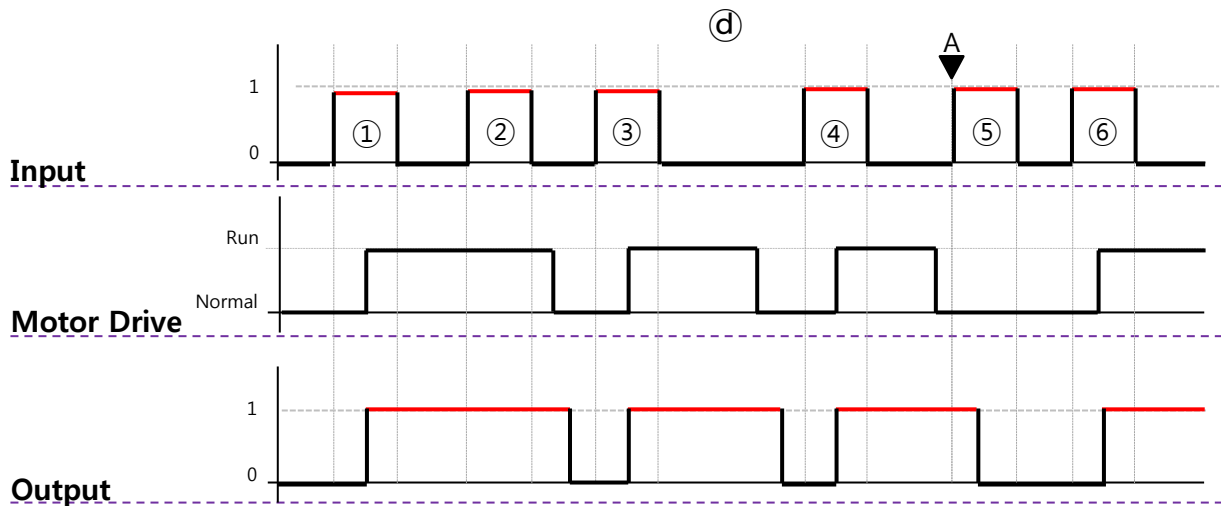
비트 명령은 상승엣지와 하강엣지 명령으로 구분됩니다.



Input의 상승 엣지의 명령 시작은 ㉠구간과 같이 '0' 상태에서 '1'로 변경되는 시점입니다. 이 명령을 받은 모션게이트는 해당 축으로 명령을 하달하고, 그 명령이 실행될 때 Output으로 명령에 대하여 응답합니다.

하강 엣지의 명령 시작은 ㉢의 구간은 와 같이 Input 명령이 '1' 상태에서 '0'으로 변경되는 시점입니다. 이 이벤트로 모션게이트는 해당 축으로 명령을 하달하고, 그 명령이 실행 될 때 Output으로 명령에 대하여 응답합니다.

㉢구간과 같은 비트 명령은 Input의 상승엣지 명령으로 모션게이트가 해당 축의 동작명령을 하달하여, 하강 엣지의 명령이 있을 때까지 지속적인 명령이 유지 되는 명령 입니다. 이 명령의 순서는 Input의 상승 엣지로 해당축의 동작이 되면, 동작에 대하여 Output으로 응답 합니다. 그리고 Input의 하강 엣지 명령으로 해당축의 동작이 정지 되면, Output으로 동작의 정지 대하여 응답됩니다.



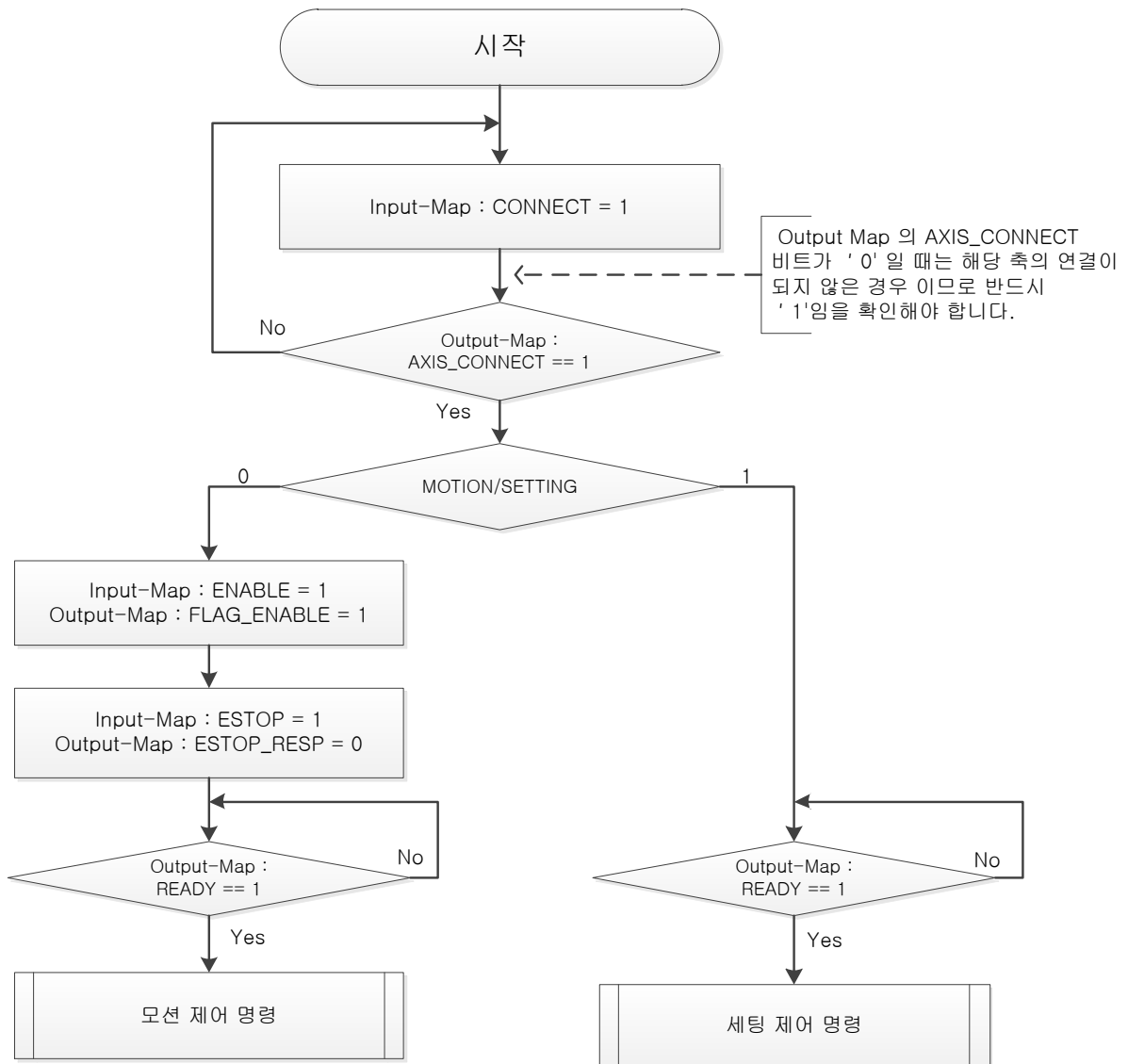
㉔구간은 Input 명령이 연속적으로 동작 했을 경우 입니다. 이러한 경우에는 ①의 명령으로 ㉔구간과 동일하게 명령이 시작됩니다. 이때 모션게이트가 동작 중일 때 입력된 명령 ② 으로 동작 되지 않습니다. 그리고 ①의 명령으로 동작한 모션게이트의 동작이 완료 된 후 입력된 ③의 명령으로는 동작이 실행됩니다.

④의 명령으로 실행된 동작이 완료되고 Output 의 응답하기 전의 시점인 A 에서 ⑤의 명령이 입력되면, 이 입력은 무시됩니다. 그러나 Output 이 응답된 후에 입력된 ⑥의 명령으로 동작합니다. 즉, Input 명령으로 모션게이트의 동작이 실행되고, Output 에 동작이 완료된 상태의 응답이 있을 때의 Input 명령이 유효 합니다.

4.3.2 IO-Map의 제어 명령 준비 순서

모션게이트는 명령을 실행할 때 아래의 순서의 과정이 필요 합니다.

순서도 1. 모션 및 설정 제어 명령의 활성화 조건



※ 모션게이트의 명령

- ① Input-Map 의 CONNECT 비트를 '1'로 세트 하여 명령을 실행 <참고: [IO-Map 사용 설명서 *2.2.1](#)>
 - CONNECT 비트는 해당축의 사용을 선택하는 비트 이므로 반드시 '1'로 세트
 - Output-Map 의 AXIS_CONNECT 비트의 응답 상태가 '1'임을 확인
- ② Input-Map 의 MOTION/SETTING 비트를 선택 <참고 : [IO-Map 사용 설명서 *2.1](#)>
 - 모션 제어는 '0', 세팅 제어는 '1'로 선택 합니다.
- ③ 모션 제어는 Input-Map 의 ENABLE 비트와 ESTOP 비트를 '1'로 세트 <참고: [IO-Map 사용 설명서 *2.2.1](#)>
 - Output-Map 의 응답비트 FLAG_ENABLE 비트 '1'임을 확인
 - ESTOP_RESP 비트가 '0' 을 확인
- ④ 명령을 실행 하고자 할 때 Output-Map 의 READY 비트의 상태를 확인 <참고: [IO-Map 사용 설명서 * 2.7](#)>
 - 다른 명령이 실행 중 일 때 READY 비트는 '0' 상태로 유지 됨.
 - 모션 명령이 없을 때 READY 비트는 '1' 상태로 유지 됨.
 - 세팅 명령을 할 때, 해당 명령이 완료 될 때까지 '0' 상태로 유지 됨.
- ⑤ 드라이브의 모션 제어는 모션 명령에 대한 IO-Map 의 비트 조합으로 수행합니다.
 <참고: [IO-Map 사용 설명서 *2.1.1, *2.1.2](#)>
 - Input-Map 의 MOTION/SETTING 비트를 '0'으로 설정
 - 모션 제어의 명령 실행은 "CONNECT= 1, ENABLE=1, nESTOP=1" 을 반드시 설정 해야 함
- ⑥ 드라이브 및 모션게이트의 설정 값에 대한 확인 및 수정은 세팅 제어에 대한 IO-Map 의 비트 조합으로 수행 됨 <참고: [IO-Map 사용 설명서 *6. *7. *8. *9](#)>
 - Input-Map 의 MOTION/SETTING 비트를 '1'로 설정
 - 세팅 제어의 명령 실행은 "CONNECT= 1" 을 반드시 설정 해야 함

5. IO-Map 의 사용 예제 (for DeviceNet)

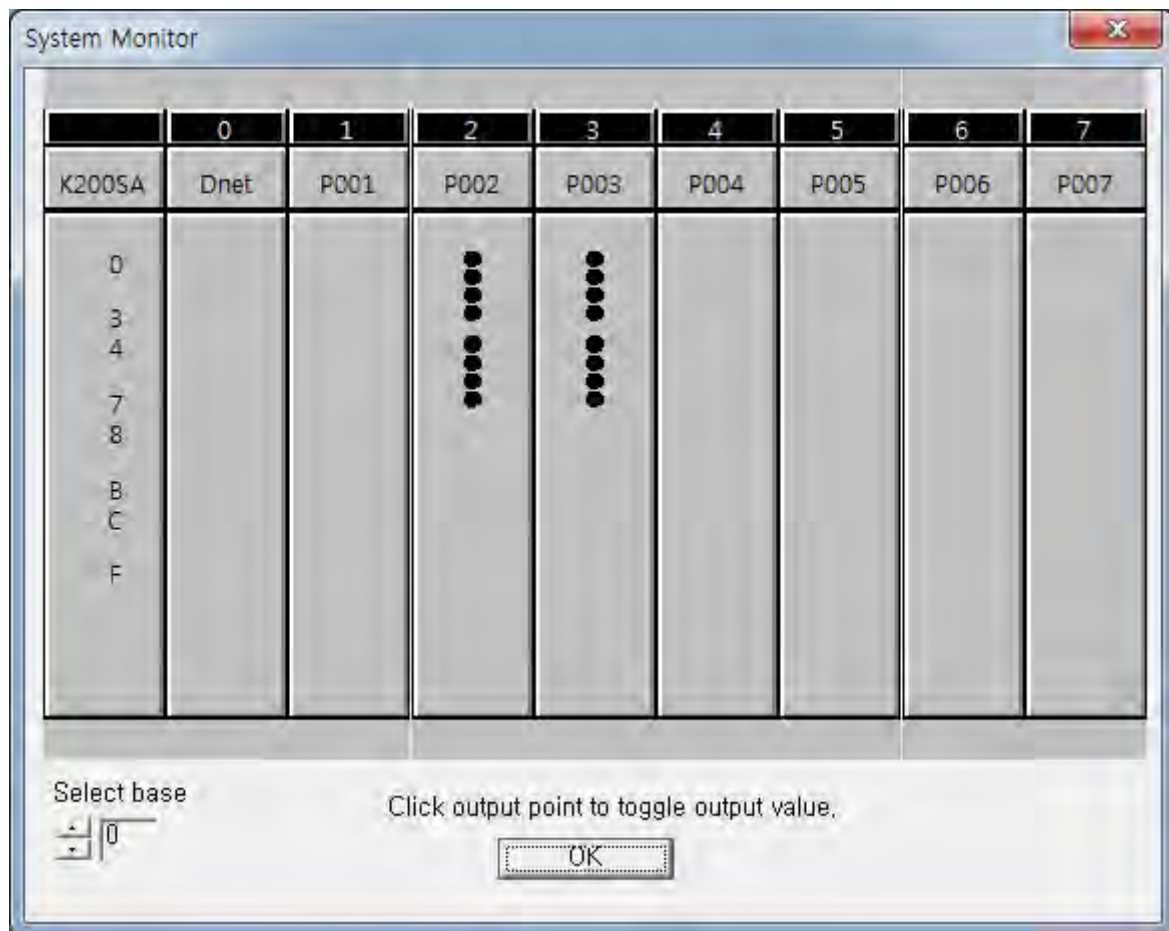
5.1. KGL for Windows 프로젝트 설정

- 네트워크 파라미터의 설정시의 주요 항목
 - 리모트 수신 : 모션게이트의 상태 및 응답 정보를 수신 받는 영역으로 Output-Map 의 정보가 매칭됩니다.
 - 리모트 송신 : 모션게이트로 비트 조합에 대한 명령을 전송하는 구간으로 Input-Map 의 정보가 매칭됩니다.

※ 본 메뉴얼에서의 예제는 다음과 같은 PLC 기반으로 제작 되었습니다.

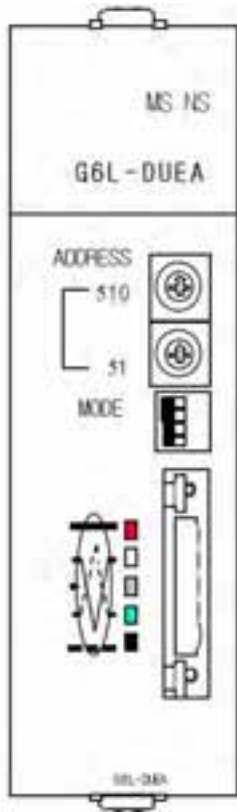
- PLC 시리즈 : MASTER K
- PLC 의 유형 : K200S
- DeviceNet 모듈 : G6L-DUEA
- PLC 다운로드 프로그램 : KGL for Windows

그림 5.1 예제에서 사용된 PLC 시스템



■ DeviceNet 모듈설정

- G6L-DUEA 의 MODE 스위치 1 번과 2 번을 변경하여 마스터로 설정합니다.
- MODE 스위치 3 번과 4 번으로 DeviceNet 의 네트워크 속도를 설정합니다.
(Ezi-MotionGate DeviceNet 의 스위치 SW.2 를 '3'으로 하였을 때 DeviceNet 의 통신속도를 자동으로 설정 됩니다.)
- ADDRESS 로터리 스위치로 0~63 범위의 DeviceNet 의 국번을 설정합니다.
(본 예제에서는 Master 의 국번을 '0', 모션게이트의 국번을 '1'로 설정 하였습니다.)



Master 또는 Slave Mode 설정

MODE. 1	MODE. 2	Operation status
OFF	OFF	Master Mode
ON	OFF	Slave Mode

네트워크 통신 속도 설정

MODE. 3	MODE. 4	Operation status
OFF	OFF	125kbps
ON	OFF	250kbps
OFF	ON	500kbps
ON	ON	

■ PLC Dnet 파라미터 설정

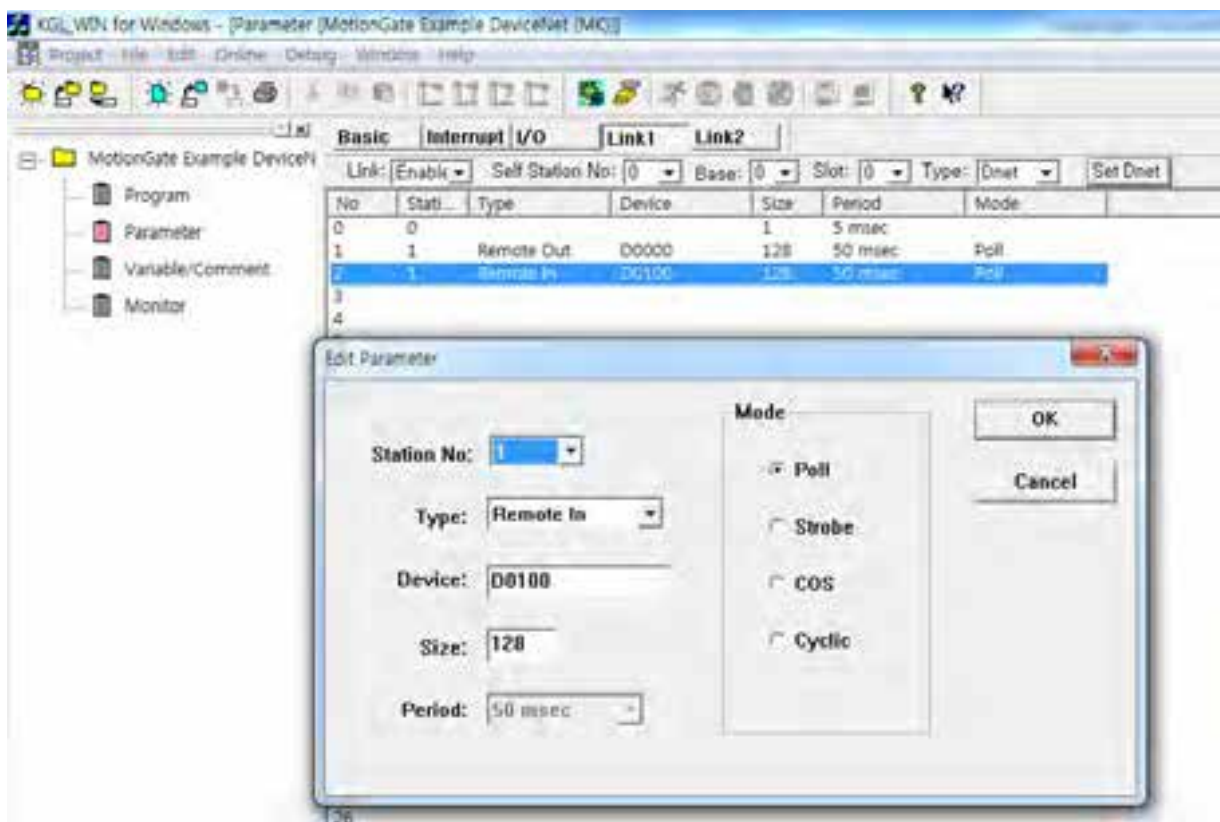
- PLC 프로그램 KGL for Windows 의 프로젝트에서 PLC 파라미터를 DeviceNet 모듈이 장착된 베이스와 슬롯번호에 맞추어 지정합니다. (본 예제에서는 DeviceNet(Dnet)을 0 번 국번으로 설정하였으며, 0 번 베이스의 0 번 슬롯에 장착 하였습니다.)
- 모든 슬레이브 장비의 데이터 스캔 후의 지연 시간인 스캔타임을 설정합니다.
- 프로그램에 따른 송신 횟수인 폴레이트를 지정 합니다. 여기에 설정된 값만큼 프로그램 스캔을 실행할 때 데이터 송신을 1 번 수행하게 됩니다.

그림 11.2 PLC 파라미터 설정 값



■ PLC 리모트 송수신 파라미터 설정

- 모션게이트의 국번(SW3 으로 지정한 번호)을 선택합니다.
- 리모트 송수신 디바이스의 시작주소를 지정합니다. (본 예제에서는 송신 디바이스를 D000, 수신 디바이스는 D100 으로 사용합니다.)
- 리모트 송수신 디바이스의 범위를 모션게이트에서 사용되는 데이터의 범위 (128byte)로 설정합니다. ([참고 4.1.2 네트워크 구성](#))
 - ◆ 송신 디바이스의 영역 : D000 ~ D063
 - ◆ 수신 디바이스의 영역 : D100 ~ D163
- 모션게이트의 DeviceNet 통신 모드는 Poll 방식이므로, 통신주기의 설정의 값은 의미가 없습니다.



5.2. PLC LADDER 프로그래밍 예제

■ 비트 영역 및 데이터 영역

그림 11. 7 리모트 송수신 디바이스의 주소범위

MotionGate [Output Map] -> PLC [Remote Receive]		PLC [Remote Transmit] -> MotionGate [Input Map]	
000h	Axis - 0	100h	Axis - 0
03Fh		103Fh	
040h	Axis - 1	1040h	Axis - 1
07Fh		107Fh	
080h	Axis - 2	1080h	Axis - 2
11Fh		111Fh	
120h	Axis - 3	1120h	Axis - 3
15Fh		115Fh	
160h	Axis - 4	1160h	Axis - 4
19Fh		119Fh	
200h	Axis - 5	1200h	Axis - 5
23Fh		123Fh	
240h	Axis - 6	1240h	Axis - 6
27Fh		127Fh	
280h	Axis - 7	1280h	Axis - 7
31Fh		131Fh	
320h	Axis - 8	1320h	Axis - 8
35Fh		135Fh	
360h	Axis - 9	1360h	Axis - 9
39Fh		139Fh	
400h	Axis - 10	1400h	Axis - 10
43Fh		143Fh	
440h	Axis - 11	1440h	Axis - 11
47Fh		147Fh	
480h	Axis - 12	1480h	Axis - 12
51Fh		151Fh	
520h	Axis - 13	1520h	Axis - 13
55Fh		155Fh	
560h	Axis - 14	1560h	Axis - 14
59Fh		159Fh	
600h	Axis - 15	1600h	Axis - 15
63Fh		163Fh	

NOTE: 각 축의 비트 주소는 DeviceNet 네트워크 설정 시 송수신 디바이스 주소의 옅셋 값입니다.

■ CONNECT

CONNECT 명령은 해당축의 사용을 결정시 사용합니다. 예제 2. 는 Axis-0 과 Axis-1 에 대한 CONNECT 명령에 대한 예제 입니다.

예제 1. 각 축의 활성화 명령 및 응답 비트 확인

✓ 제어 비트맵의 주소

- **Input-Map**
CONNECT – D000.0
- **Output-Map**
CONNECTED – D100.0

✓ 입출력 정보

- **입력신호**
CONNECT
- **출력신호**
CONNECT RESP

✓ IO-Map 의 명령 및 응답 형태

- **CONNECT 명령의 실행 전**

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

CONNECT bit = 0

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

CONNECTED bit = 0

- CONNECT 명령의 실행 후

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

CONNECT bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

CONNECTED bit = 1

✓ LADDER



NOTE: 예제 1.은 Axis-0 의 이후의 예제에 포함 되는 기본 예제이며, 이 명령이 실행된 상태에서 이후의 예제가 실행 가능합니다.

✓ 명령 순서

- ① CONNECT 의 ON 신호로, CONNECT 비트(D000)의 0 번 비트를 세트 합니다.
- ② Axis-0 의 접속이 정상 접속이 되었을 시 응답 데이터가 출력됩니다.
- ③ 0 번 축의 CONNECT 명령에 대한 응답은 D100 의 0 번 비트의 변화로 확인 됩니다.
- ④ 만약 접속이 되지 않았을 경우 D000 의 0 번 비트를 제외한 모든 4WORD 의 데이터 레지스터의 비트는 1 로 세트 되어 있습니다.
- ⑤ 접속 응답 신호 CONNECTED 의 비트 상태는 CONNECT Resp 코일로 확인 됩니다.

■ ENABLE 명령과 E-ETOP 명령

ENABLE 명령은 E-STOP 명령이 비활성화 된 상태에서 동작됩니다. 예제 2. 은 Axis-0 의 모터 활성화 명령과 E-STOP 명령에 대한 예제 입니다.

예제 2. 모터 활성화 명령과 비상정지 명령

✓ 제어 비트맵의 주소

- Input-Map

ENABLE – D000.1

E-STOP – D000.2

- Output-Map

ENABLED – D100.1

E-STOP_RESP – D100.2

✓ 입출력 정보

- 입력신호

ENABLE

E-STOP

- 출력신호

ENABLE RESP

E-STOP LoopBack

✓ IO-Map 의 명령 및 응답 형태

- ENABLE 명령을 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

ENABLE bit = 1

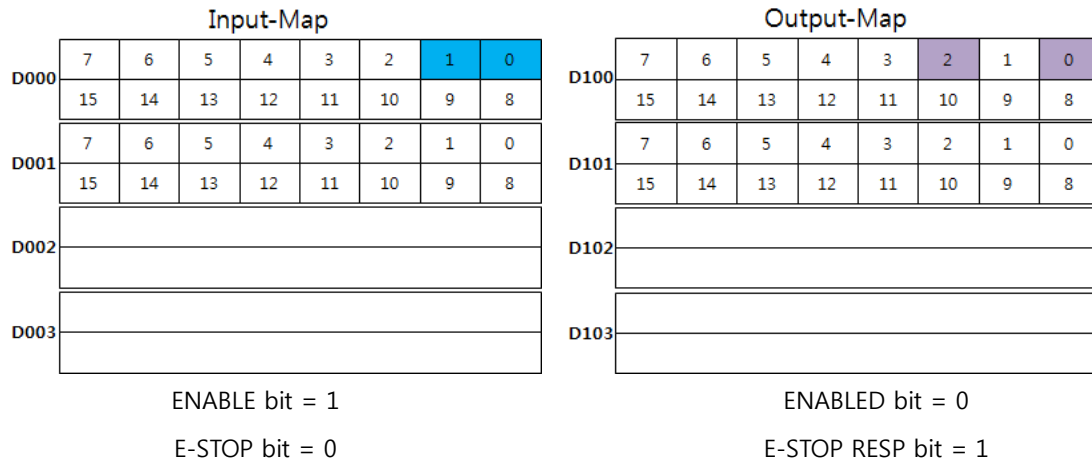
E-STOP bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

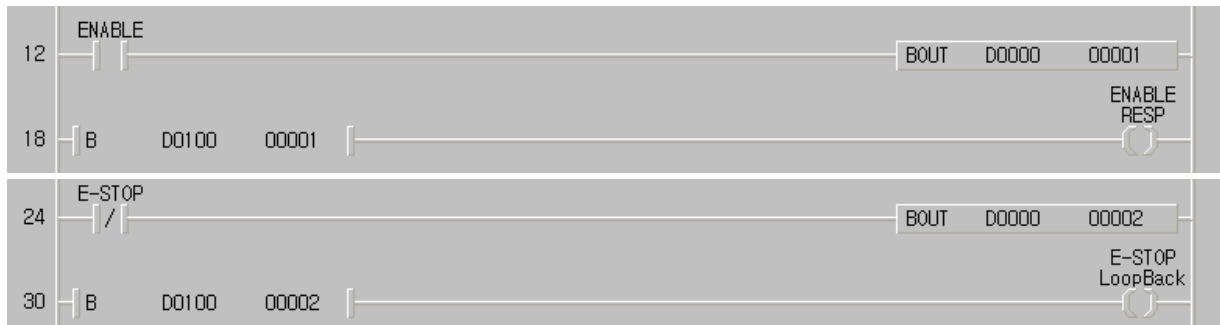
ENABLED bit = 1

E-STOP RESP bit = 0

- E-STOP 명령을 실행



✓ LADDER



NOTE: 예제 2. 는 Axis-0 의 ENABLE 명령과, E-STOP 에 대한 예제로, CONNECT 명령이 생략되었습니다.

✓ 명령 순서

- ① Axis-0 의 모터 활성화 명령은 E-STOP 명령이 실행되지 않는 상태에서 실행 되므로 E-STOP 비트를 '1' 상태로 유지합니다.
- ② E-STOP 명령은, B 점접 명령으로 구성하여 입력신호가 차단되었을 때 E-STOP 비트가 '0'이 되도록 구성합니다.
- ③ E-STOP 명령이 실행되면, 모션게이트의 ENABLE 명령은 자동으로 취소됩니다.
- ④ ENABLED 와 E-STOPED 의 비트 상태는 ENABLE RESP 와 E-STOP LoopBack 코일로 확인됩니다.

■ ALARM 상태 확인

알람상태는 ALM/ERR 비트로 확인할 수 있습니다. 예제 4. 는 Axis-0 의 알람상태 확인 방법과 알람 해제에 대한 예제 입니다.

예제 3. 알람상태 확인 및 알람해제 명령

✓ 제어 비트맵의 주소

- Input-Map

ENABLE – D000.1

E-STOP – D000.2

ALARM_RESET – D000.3

- Output-Map

ENABLED – D100.1

E-STOP_RESP – D100.2

ALARM_ERROR – D100.3

✓ 입출력 정보

- 입력신호

ALM_RST

- 출력신호

ALM_STAT

✓ IO-Map 의 명령 및 응답 형태

- 알람 발생 시

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

ALARM RESET bit = 0

ENABLE bit = 1

E-STOP bit = 1

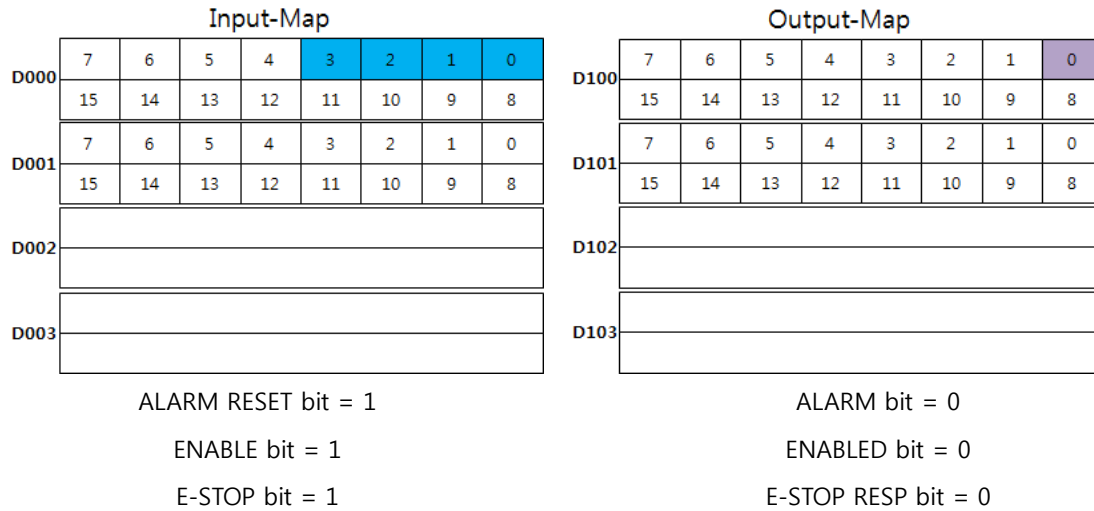
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

ALARM bit = 1

ENABLED bit = 0

E-STOP RESP bit = 0

- 알람 해제 명령을 실행



✓ LADDER



NOTE: 예제 3.의 래더는 예제 2.에 추가된 명령으로, CONNECT, ENABLE, E-STOP 명령이 생략되었습니다.

✓ 명령 순서

- ① 알람 상태는 ALARM/ERROR 비트로 응답됩니다.
- ② 발생한 알람은 ALM_RST의 Close 입력으로, 발생한 알람을 해제 할 수 있습니다.
- ③ 알람이 발생되면 모터는 비 활성화 되고 ENABLED 비트는 '0' 상태로 전환됩니다.

■ CANCEL 명령

실행취소 명령은 모션정지, 일시정지 명령의 취소, PT 운전 정지를 위하여 사용됩니다. 예제 4 는 Axis-0 의 실행취소 명령에 대한 예제 입니다.

예제 4. 실행취소 명령

✓ 제어 비트맵의 주소

- Input-Map

MOTION/SETTING – D000.7

CANCEL – D001.0

- Output-Map

MOTION/SETTING_RESP – D100.7

MOTIONING – D101.0

✓ 입출력 정보

- 입력신호

CANACLE

- 출력신호

MOTIONING

✓ IO-Map 의 명령 및 응답 형태

- CANCEL 명령을 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CANCEL bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

MOTIONING bit = 0

✓ LADDER



NOTE: 예제 4.의 래더는 예제 3.에 추가된 명령으로, 앞의 예제 내용이 생략 되었습니다.

✓ 명령 순서

- ① CANCEL 의 Close 입력으로 MOTION/SETTING 비트를 '0'으로 설정 하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② Input-Map 의 CANCEL 비트를 '1'로 설정하면, 실행 취소명령이 실행됩니다.
- ③ 만약 해당 축이 모션명령 또는 일시정지 명령이 실행중인 상태에서 CANCEL 명령을 실행 하였을 때, MOTIONING 비트와, HOLD RESP, PT RUNNING 비트가 OFF 됨을 확인할 수 있습니다.

※ MOTION/SETTING 모드 확인 방법

모션과 설정 모드는 IO-Map 의 MOTION/SETTING 비트로 명령하고, MOTION/SETTING_RESP 비트로 응답 확인을 합니다.

모션 모드에 대한 명령은 이 비트를 '0'으로 설정 하였을 때 Output-Map 의 [0.7] 비트가 '0'으로 루프-백 되어 모션 명령에 대한 응답임을 확인합니다.

설정 모드에 대한 명령은 이 비트를 '1'으로 설정 하였을 때 Output-Map 의 [0.7] 비트가 '1'으로 루프-백 되어 설정 명령에 대한 응답임을 확인합니다.

✓ LADDER



NOTE 1: 본 래더는 접점수의 증가 및 PLC 스텝의 증가를 방지하기 위한 PLC 회로입니다.

NOTE 2: 모든 예제의 래더에서 사용되는 M/Set Resp 는 MOTION/SETTING_RESP 비트 입니다.

■ HOLD 명령

일시정지 명령은 모션 명령을 일시적으로 정지하고, 재개 할 수 있도록 하는 명령입니다. 예제 5는 Axis - 0 의 일시정지 명령에 대한 예제 입니다.

예제 5. Axis - 0의 일시정지 명령

✓ 제어 비트맵의 주소

- Input-Map

MOTION/SETTING – D000.7

HOLD – D001.1

- Output-Map

MOTION/SETTING_RESP – D100.7

MOTIONING – D101.0

HOLD_RESP – D101.1

✓ 입출력 정보

- 입력신호

HOLD

- 출력신호

MOTIONING

HOLD Resp

✓ IO-Map 의 명령 및 응답 형태

- HOLD 명령을 실행하여 모션 일시정지

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

HOLD bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

HOLD_RESP bit = 1

MOTIONING bit = 0

- HOLD 명령을 해제하여 모션 재개

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

HOLD bit = 0

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

HOLD_RESP bit = 0

MOTIONING bit = 1

✓ LADDER



NOTE: 예제 5의 래더는 예제 4.에 추가된 명령으로, 앞의 예제 내용이 생략 되었습니다.

✓ 명령 순서

- ① HOLD 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② 일시정지 명령은 CMD_CODE 의 값과 무관하게 Input-Map 의 HOLD 비트가 '1'일 때 적용됩니다.
- ③ 모션게이트의 HOLD 명령이 실행된 상태에서 HOLD 입력이 Open 되면, Input-Map 의 HOLD 비트가 OFF 로 전환되어 일시정지 된 모션이 재개 됩니다.
- ④ 모션게이트의 HOLD 명령이 실행되면 모션 모드일 때만 Output-Map 의 HOLD_RESP 비트가 '1'로 상태가 됩니다. 따라서 두 Output-Map 의 비트를 AND 회로로 하여 확인 합니다.

■ RESPONSE TYPE 설정

RESPONSE TYPE(응답 데이터 설정) 설정은 모션모드(MOTIONION)상태에서 확인할 수 있습니다.
예제 6 은 Axis-0 의 RESPONSE TYPE 설정 방법에 대한 예제 입니다.

예제 6. 응답 데이터 설정

✓ 제어 비트맵의 주소

- Input-Map

MOTION/SETTING – D000.7

RESPONSE_TYPE 0~3 – D000.12~D000.15

- Output-Map

MOTION/SETTING_RESP – D100.7

RESPONSE_TYPE_RESP 0~3 – D100.12~D100.15

RESPONSE_DATA – D102~D103 (D102 [1 DWORD])

✓ 입출력 정보

- 입력신호

Response Type 0~3

- 출력데이터(DWORD)

Command Position

Actual Position

Position Error

Actual Velocity

Current PT No

Current Alarm No

✓ IO-Map 의 명령 및 응답 형태

- 응답 데이터를 요청하지 않음

Input-Map

D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

Output-Map

D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0000b

- 지령 위치 값(Command Position)을 요청

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0001b

RESPONSE_TYPE 0~3 = 0000b

RESPONSE_DATA = 0

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0001b

RESPONSE_DATA = Command Position

- 현재 위치 값(Actual Position)을 요청

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0010b

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0010b

RESPONSE_DATA = Actual Position

- 위치 오차 값(Position Error)을 요청

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0011b

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0011b

RESPONSE_DATA = Position Error

- 현재 속도 값(Actual Velocity)을 요청

		Input-Map							
D000	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D001	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0101b

		Output-Map							
D100	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D101	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0101b

RESPONSE_DATA = Actual Velocity

- 현재의 PT 번호 요청(Current PT No.)을 요청

		Input-Map							
D000	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D001	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0101b

		Output-Map							
D100	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D101	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0101b

RESPONSE_DATA = Current PT No.

- 현재의 알람정보(Current Alarm Info.)을 요청

		Input-Map							
D000	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D001	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D002									
D003									

MOTION/SETTING bit = 0

RESPONSE_TYPE 0~3 = 0100b

		Output-Map							
D100	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D101	7	6	5	4	3	2	1	0	
	15	14	13	12	11	10	9	8	
D102									
D103									

MOTION/SETTING_RESP bit = 0

RESPONSE_TYPE 0~3 = 0100b

RESPONSE_DATA = Current Alarm Info.

✓ LADDER

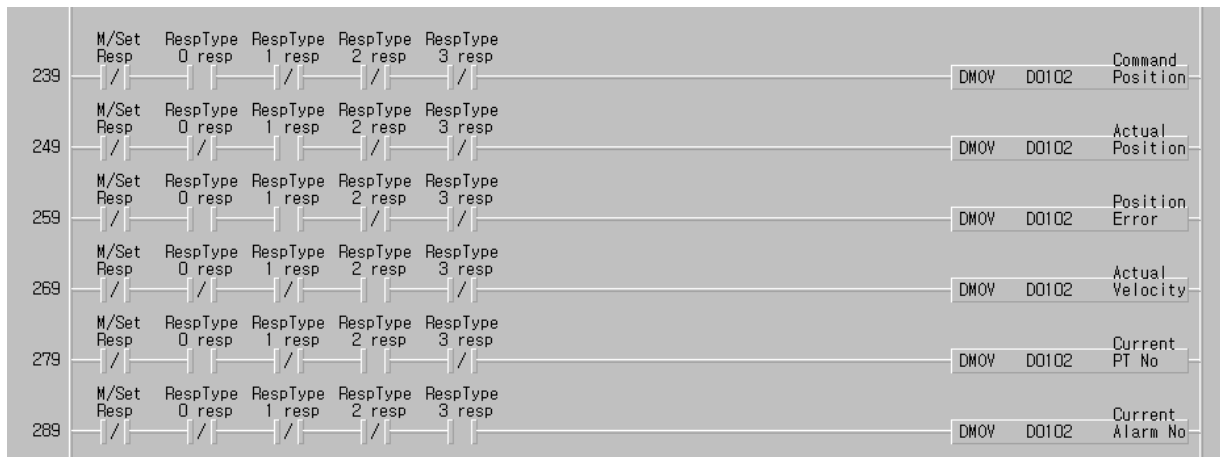
- Response Type 영역에 응답데이터 형식을 입력



- Response Type 의 응답 비트 정보를 간략화



- 간략화 된 비트정보를 받아 응답 데이터를 분류



NOTE: 예제 6.의 래더는 예제 5.에 추가된 명령으로, 앞의 예제 내용이 생략 되었습니다.

✓ 명령 순서

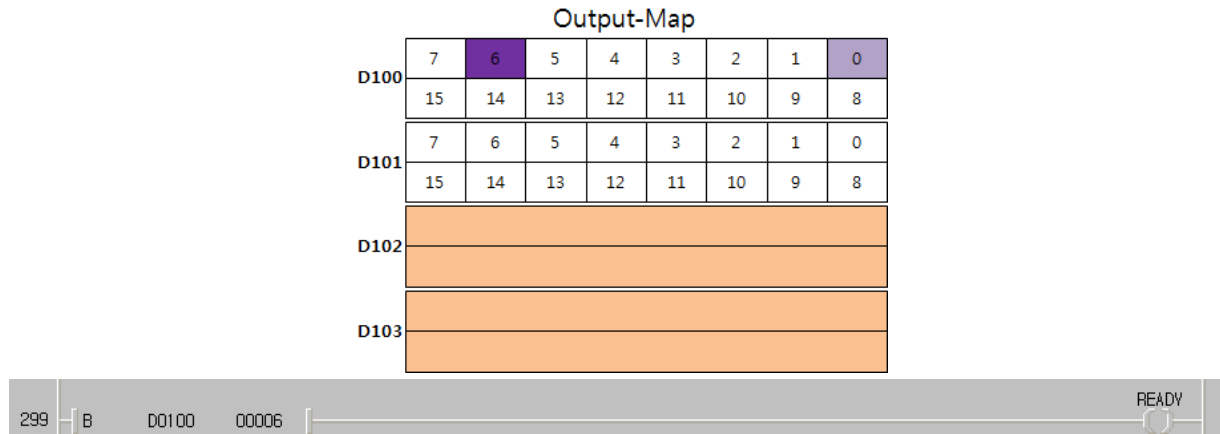
- ① 응답 데이터 설정은 모션 모드에서만 사용 가능하므로, MOTION/SETTING 비트가 0 상태 일 때를 인지하여 명령을 실행 합니다.
- ② Output-Map 의 데이터 영역으로 수신되는 데이터의 분류는 정보는 응답 비트의 조합으로 확인 됩니다. 여기서 응답 비트는 MOTION/SETTING_RESP 비트가 0 일 때 수신된 비트 정보만 유효합니다.

■ 상태 정보 확인

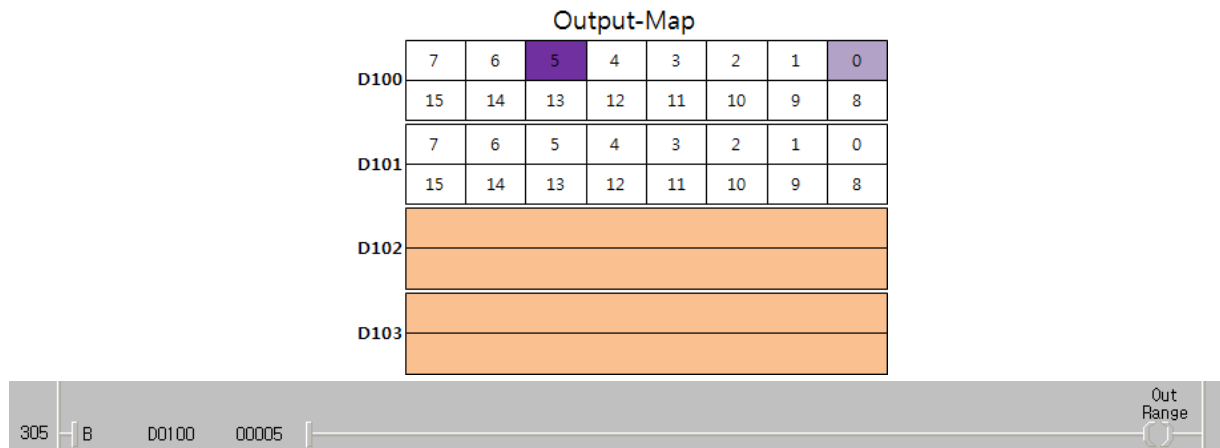
READY 비트, Out RANGE 비트를 제외한 상태정보 비트는 모션모드(MOTIONION)상태에서 확인할 수 있습니다. 예제 7 은 Axis-0 의 상태정보 확인 방법에 대한 예제 입니다.

예제 7. 응답 데이터 설정

- ✓ READY 비트[D100.06]: IO-Map 에 대한 명령이 가능 할 때 ON 됩니다.

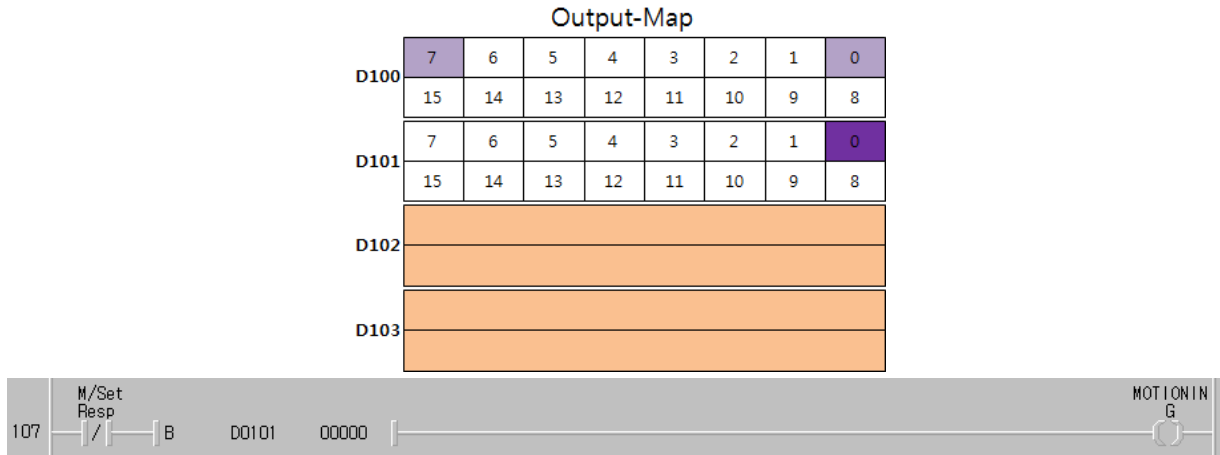


- ✓ Out. Range 비트[D100.05]: IO-Map 에 대한 명령시 데이터 범위를 초과 하였을 때 ON 됩니다.

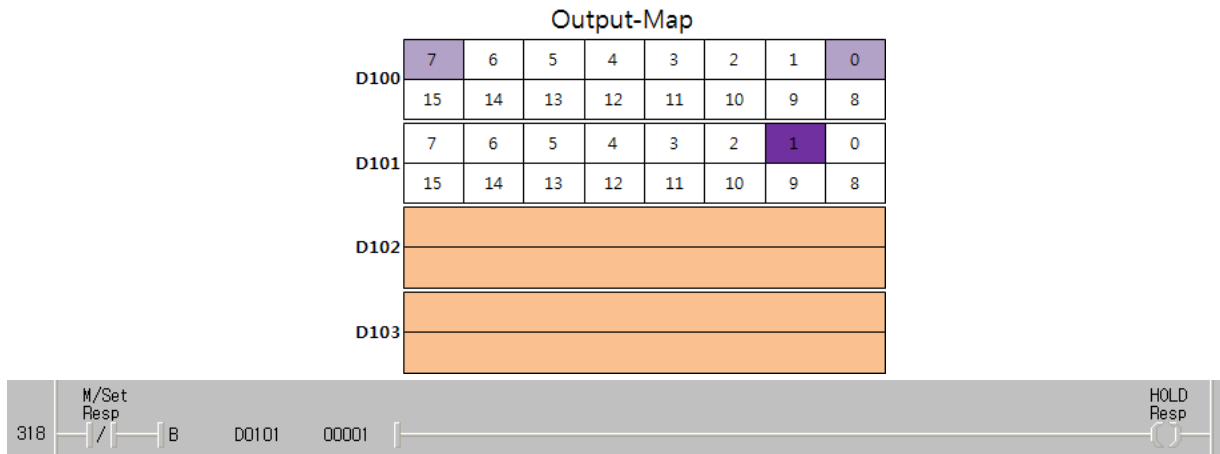


NOTE: READY 비트와 Out Range 비트는 모션모드와 설정 모드에서 동일한 조건으로 동작합니다.

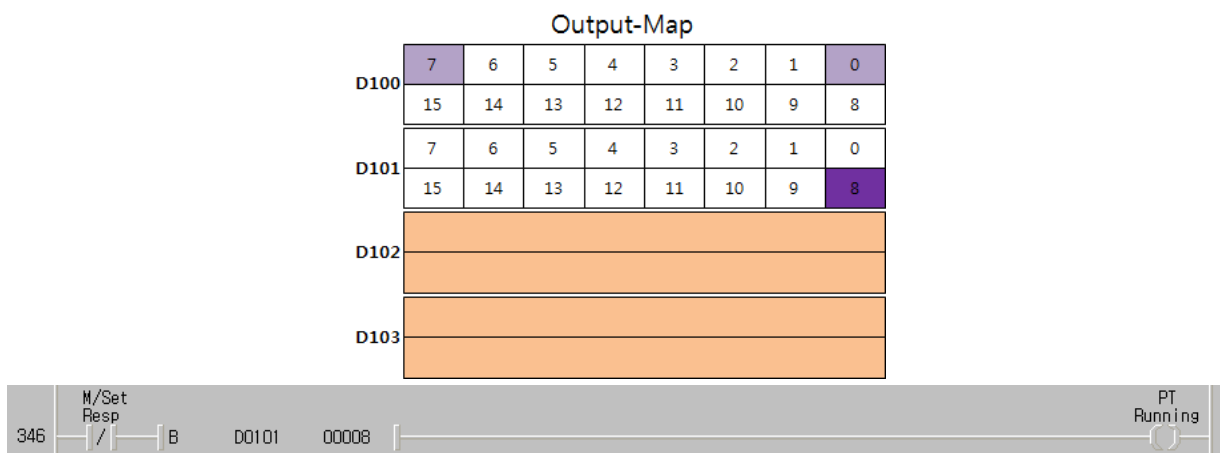
- ✓ MOTIONING 비트[D101.00]: 모터가 구동중일 때 ON 됩니다.



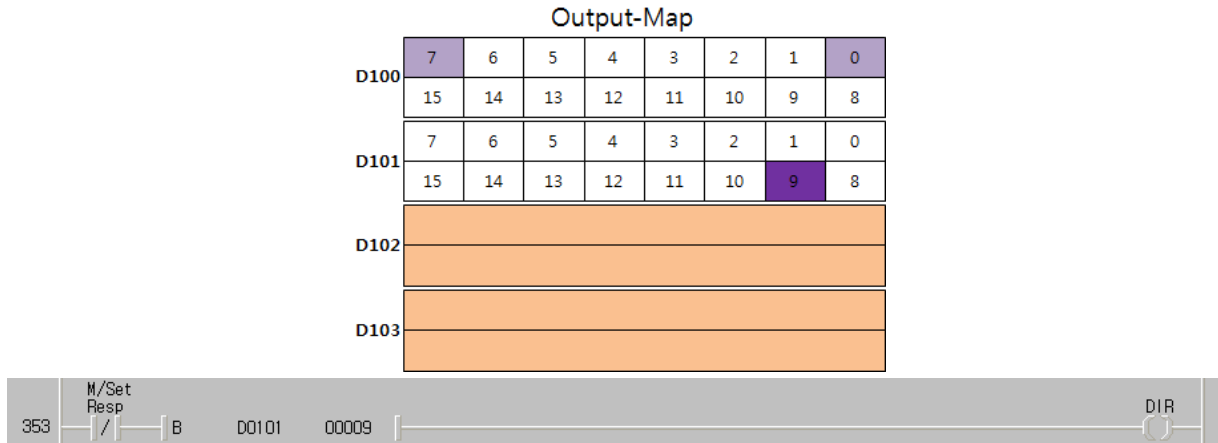
- ✓ HOLD RESP. 비트[D101.01]: 운전중 일시정지 상태로 되었을 때 ON 됩니다.



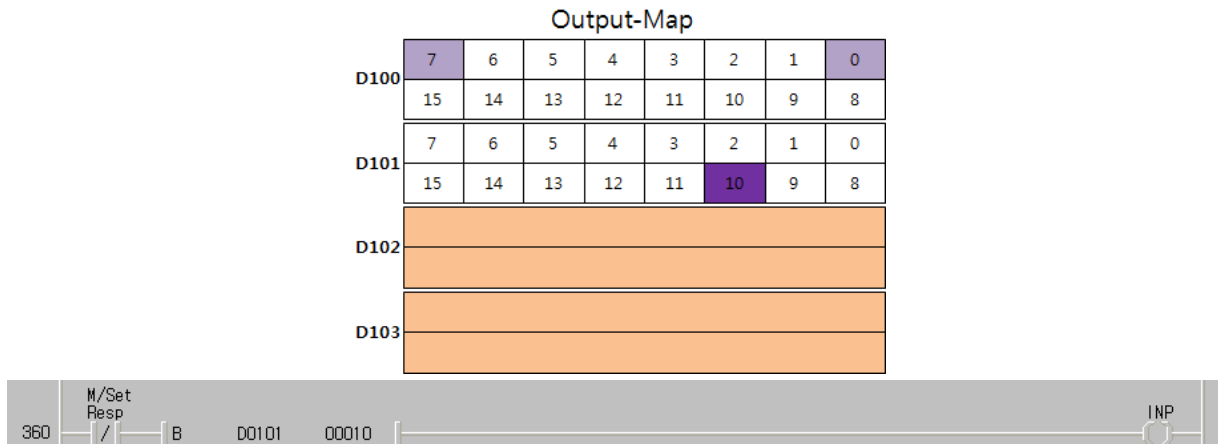
- ✓ PT RUNNING 비트[D101.08] : PT 운전 상태일 때 ON 됩니다.



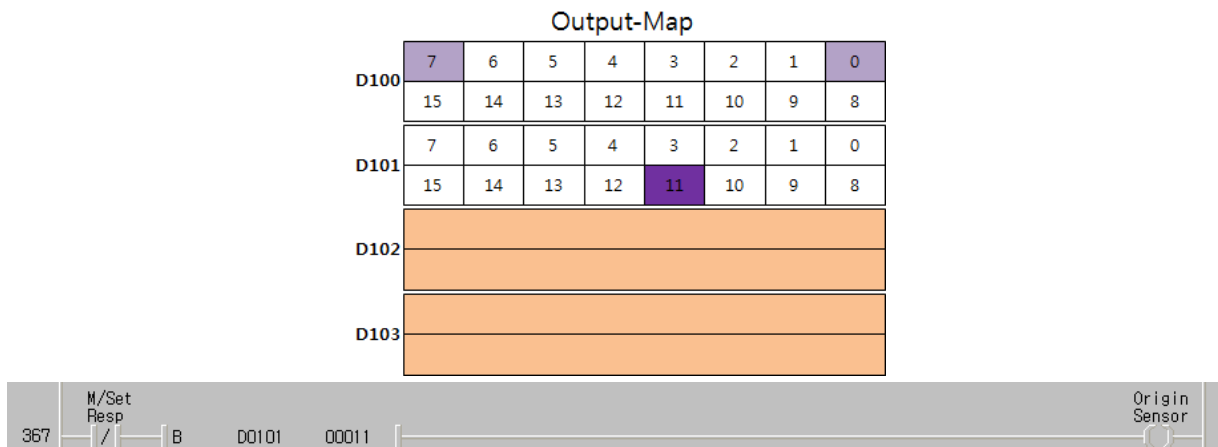
- ✓ DIR 비트[D101.09]: 모터의 회전방향이 정방향(CW)상태일 때 ON 됩니다.



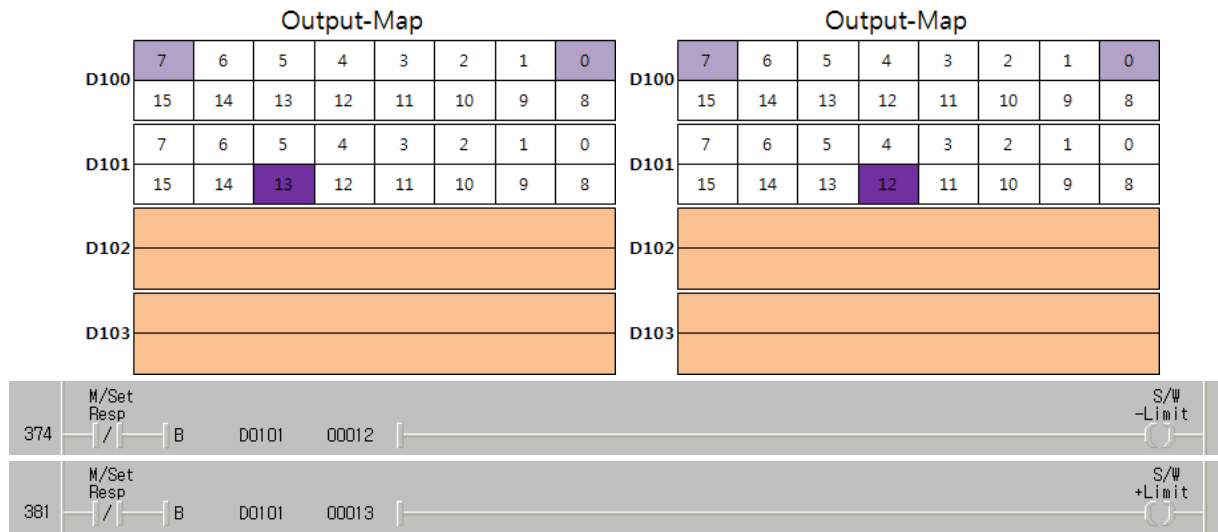
- ✓ INP 비트[D101.10]: Inposition 이 완료된 상태일 때 ON 됩니다.



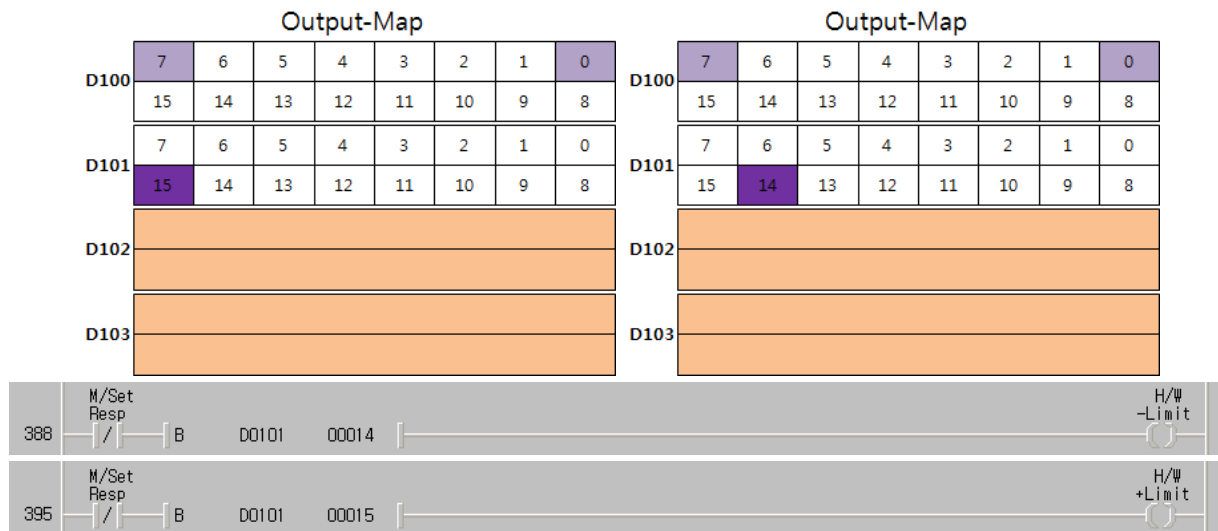
- ✓ OGRIGIN_SENSOR 비트[D101.11]: 원점센서가 ON 되어 있는 상태일 때 ON 됩니다.



- ✓ S/W -LIMIT 비트[D101.12]: -방향 프로그램 리미트를 초과한 경우 일 때 ON 됩니다.
- ✓ S/W +LIMIT 비트[D101.13]: -방향 프로그램 리미트를 초과한 경우 일 때 ON 됩니다.



- ✓ H/W +LIMIT 비트[D101.14]: +방향 리미트 센서가 ON 이 된 경우 일 때 ON 됩니다.
- ✓ H/W -LIMIT 비트[D101.15]: -방향 리미트 센서가 ON 이 된 경우 일 때 ON 됩니다.



NOTE: 상태정보의 확인은 IO-Map 이 모션모드에서 확인이 가능합니다.

■ CMD START 명령

CMD START 명령은 IO-Map 의 모션 모드(MOTION/SETTING = 0)인 위치이동, PT 운전, 원점이동과 설정 모드(MOTION/SETTING = 1)에서의 명령 실행으로 사용됩니다.

이 명령은 예제 8 와 같은 래더로 구성하여 사용합니다.

예제 8. 모션 모드의 CMD START 실행 방법

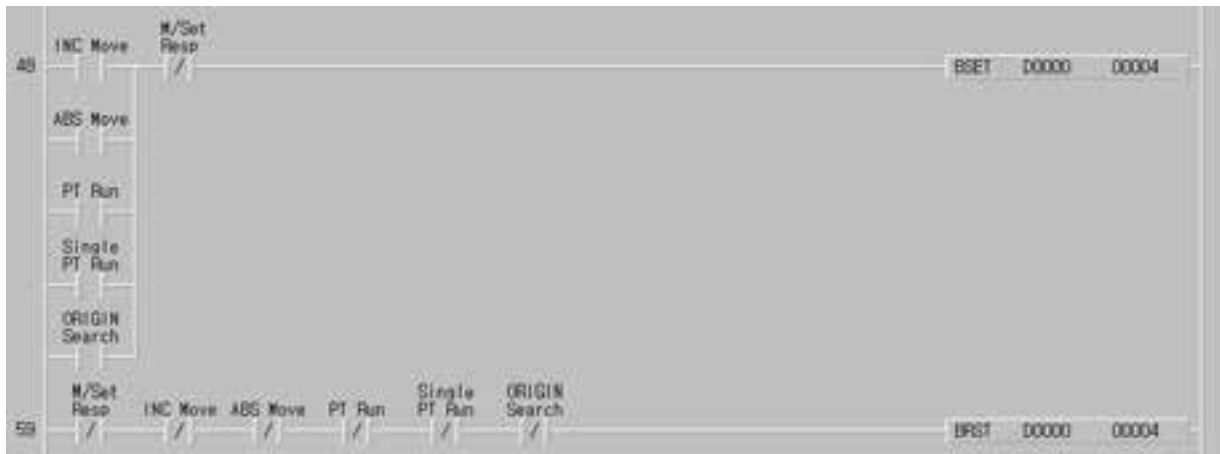
✓ IO-Map 의 명령 및 응답 형태

- 모션 모드에서의 명령 및 응답

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

✓ LADDER



NOTE : 예제 8 은 예제 15, 16, 18, 19, 21 에 추가된 명령으로, 해당 명령을 실행하기 위하여 추가된 예제입니다.

✓ 명령 순서

- ① CMD_START 비트가 필요한 제어의 입력신호를 OR 로 구성합니다.
- ② 입력신호와 MOTION/SETTING_RESP 의 N.C 신호를 AND 로 구성하여 CMD_START 비트를 '1' 로 세트 합니다.
- ③ 제어의 입력신호가 없을 때 CMD_START 비트가 '0'으로 리셋 되도록 합니다.

예제 9. 설정 모드의 CMD START 실행 방법

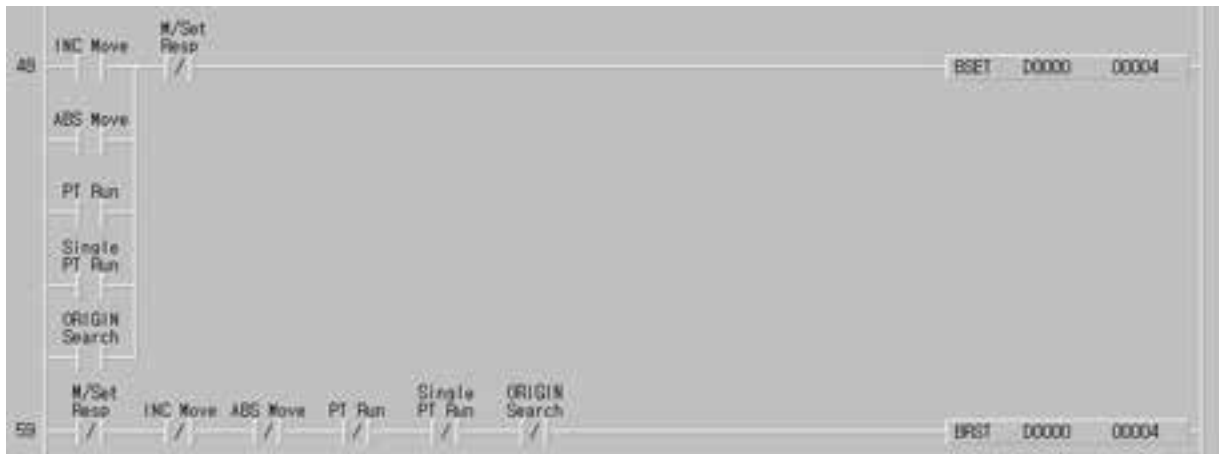
✓ IO-Map 의 명령 및 응답 형태

- 모션 모드에서의 명령 및 응답

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

✓ LADDER



NOTE : 예제 9 는 예제 15, 16, 18, 19, 21 에 추가된 명령으로, 해당 명령을 실행하기 위하여 해당 예제를 사용을 위해 추가된 예제입니다.

✓ 명령 순서

- ① CMD_START 비트가 필요한 제어의 입력신호를 OR 로 구성합니다.
- ② 입력신호와 MOTION/SETTING_RESP 의 N.O 신호를 AND 로 구성하여 CMD_START 비트를 '1' 로 세트 합니다.
- ③ 제어의 입력신호가 없을 때 CMD_START 비트가 '0'으로 리셋 되도록 합니다.

5.2.1 조그 운전 명령

조그 운전은 모션모드(MOTIONION)상태의 명령 코드(CMD_CODE) '0'에서 동작합니다. Speed Step Move 또는 Speed Ratio Move 과 Speed Value 의 조그 운전 명령 응답 확인은 JOG_Resp. 비트로 확인 가능합니다.

■ JOG Move – Speed Step Move or Speed Ratio Move

조그 운전의 Speed Step Move 과 Speed Ratio Move 는 운전 명령 방법은 동일합니다. 이 운전 방법은 파라미터 PN#104 『Use Jog Speed Ratio』의 설정 값으로 선택 됩니다. (*참고 6.1)

조그 운전의 Speed Step Move 는 저장된 0~3 의 속도 단계로 조그 운전하는 모션입니다.

조그 운전의 Speed Ratio Move 는 저장된 파라미터의 PN#105 『Move Speed for Jog Move: Ratio』의 비율로 운전합니다.

예제 10. Speed Step Move , Speed Ratio Move 조그 운전 명령

✓ 제어 비트맵의 주소

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 SPD.MODE – D001.10
 -JOG – D001.4
 +JOG – D001.5
 Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 JOG_RESP – D101.5

✓ 입출력 정보

- 입력 정보

JOG-
 JOG+
 JOG Speed Step No. (입력 범위 : 0~3 [DWORD])
 JOG Speed Ratio Value (입력 범위 : 1~255 [DWORD])

- 출력 정보

MOTIONING
 JOG Resp

✓ IO-Map 의 명령 및 응답 형태

- -JOG 명령 실행

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 0;

+JOG bit = 0, -JOG bit = 1

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 1, MOTIONING bit = 1

- +JOG 명령 실행

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 0;

+JOG bit = 1, -JOG bit = 0

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 1, MOTIONING bit = 1

- -JOG , +JOG 비트가 '0'인 상태

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 0;

+JOG bit = 0, -JOG bit = 0

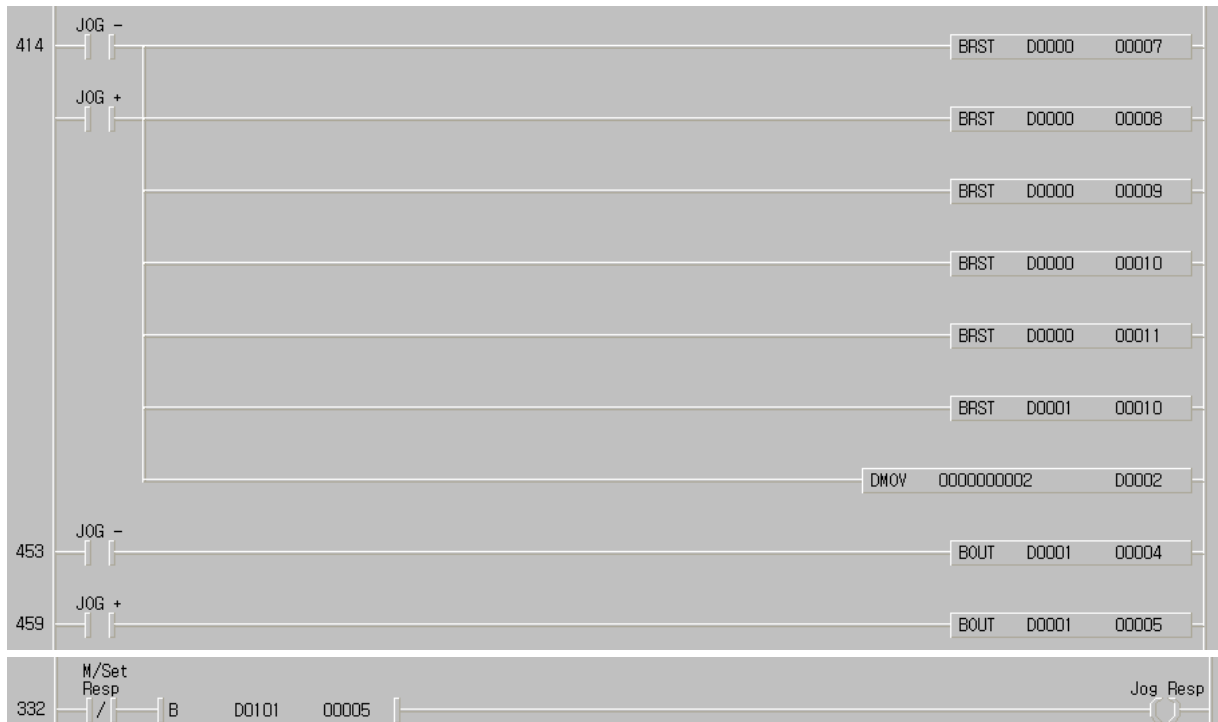
		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 0, MOTIONING bit = 0

✓ LADDER



NOTE 1: 예제 10 의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 파라미터 Pn#104 『Use Jog Speed Ratio』 값이 '1'일 때 Speed Ratio Move 모드에서 예제 10 의 래더를 실행 하면, Pn#105 『Move Speed for Jog Move: Ratio』 값의 2%의 속도로 운전합니다.

✓ 명령 순서

- ① JOG- 또는 JOG+ 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0000b 로 설정하여 일반모션 모드로 설정합니다.
- ③ SPD_MODE 비트가 '0'이 되도록 설정 합니다.
- ④ 파라미터 Pn#104 『Use Jog Speed Ratio』 가 '0'일 때 Speed Step 운전 모드 이므로 Speed Step 번호 0~3 의 값을 DMOV 명령으로 Command Data 영역에 입력합니다.
- ⑤ 파라미터 Pn#104 『Use Jog Speed Ratio』 가 '1'일 때 Speed Ratio 운전모드 이므로 속도 비율의 범위 1~255 를 DMOV 명령으로 Command Data 영역에 입력합니다.
- ⑥ Input-Map 의 -JOG 또는 +JOG 비트가 '1'이 되도록 설정합니다.
- ⑦ -JOG 또는 +JOG 비트가 '0'이 되면 조그 운전이 정지 되고, MOTIONING 비트는 '0'이 됩니다.
- ⑧ 조그 운전의 명령 응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 JOG_RESP 비트의 AND 조합으로 확인 합니다.

■ JOG Move – Speed Value Move

조그 운전의 Speed Value Move 는 입력된 값을 실제 속도에 적용하여 조그 운전하는 모션입니다.
예제 11 는 Axis-0 의 조그 운전의 Speed Value Move 에 대한 예제 입니다.

예제 11. Speed Value Move 형태의 조그 운전 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
CMD_CODE – D000.8~D000.11
SPD.MODE – D001.10
-JOG – D001.4
+JOG – D001.5
Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7
MOTIONING – D101.1
JOG_RESP – D101.5

✓ 입출력 정보

- 입력 정보

JOG-
JOG+
JOG Speed Value (입력 범위 : 1~500,000 pps [DWORD])

- 출력 정보

MOTIONING
JOG Resp

✓ IO-Map 의 명령 및 응답 형태

- -JOG 명령 실행

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 1;

+JOG bit = 0, -JOG bit = 1

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 1, MOTIONING bit = 1

- +JOG 명령 실행

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 1;

+JOG bit = 1, -JOG bit = 0

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 1, MOTIONING bit = 1

- -JOG , +JOG 비트가 '0'인 상태

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0000b , SPD_MODE bit = 1;

+JOG bit = 0, -JOG bit = 0

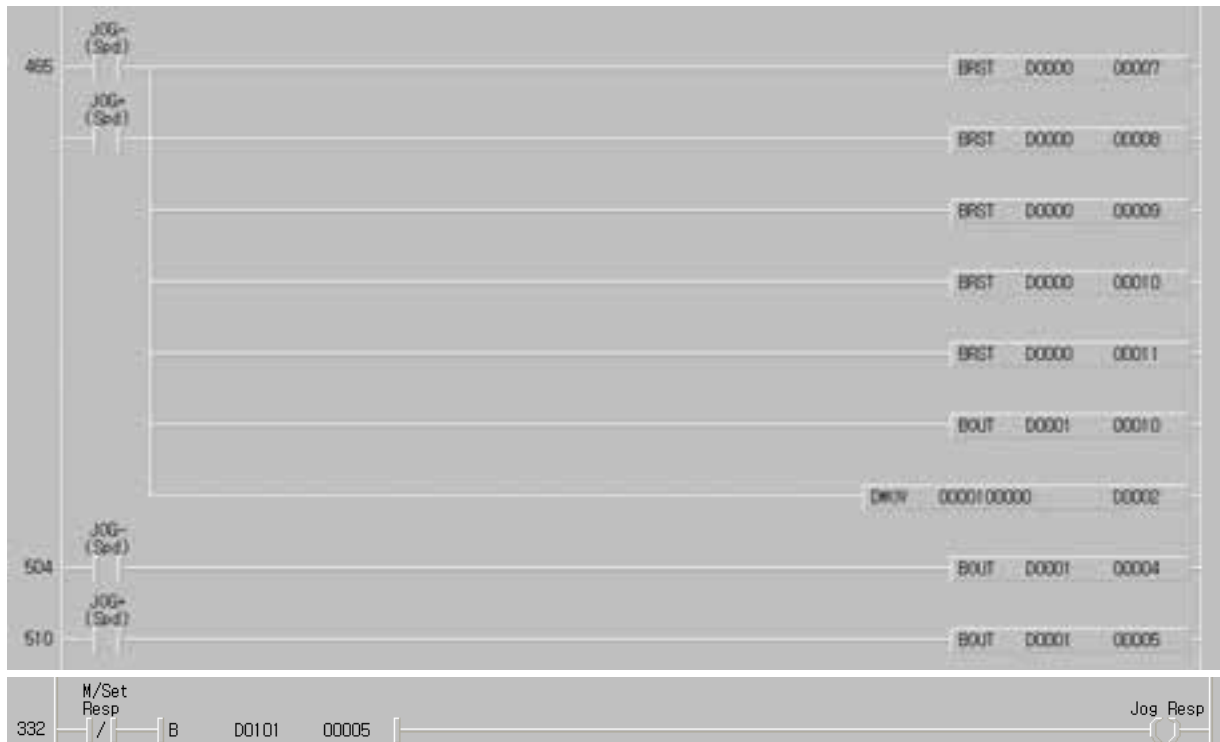
		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

JOG_RESP bit = 0, MOTIONING bit = 0

✓ LADDER



NOTE 1: 예제 11 의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 입력에 따른 방향으로 10,000pps 로 조그 운전을 하도록 구성되었습니다.

✓ 명령 순서

- ① JOG- 또는 JOG+ 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0000b 로 설정하여 일반모션 모드로 설정합니다.
- ③ SPD_MODE 비트가 '1'이 되도록 설정 합니다.
- ④ 조그 운전시의 실제 속도 값을 DMOV 명령으로 Command Data 영역에 입력합니다.
- ⑤ Input-Map 의 -JOG 또는 +JOG 비트가 '1'이 되도록 설정합니다.
- ⑥ -JOG 또는 +JOG 비트가 '0'이 되면 조그운전이 정지 되고, MOTIONING 비트는 '0'이 됩니다.
- ⑦ 조그 운전의 명령 응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 JOG_RESP 비트의 AND 조합으로 확인 합니다.

5.2.2 스텝 이동 명령

스텝 이동은 모션모드(MOTIONION)상태의 명령 코드(CMD_CODE) '0'에서 동작하며, 스텝이동 거리 번호 0~3 의 값을 지정하여 이동합니다. 예제 13 은 Axis-0 의 스텝이동에 대한 예제 입니다.

예제 12. 스텝 이동 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7

CMD_CODE – D000.8~D000.11

-STEP – D001.6

+STEP – D001.7

Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7

MOTIONING – D101.1

STEP_RESP – D101.5

✓ 입출력 정보

- 입력 정보

STEP-

STEP+

STEP Step Distance No. (입력 범위 : 0~3 [DWORD])

- 출력 정보

MOTIONING

Step Resp

✓ IO-Map 의 명령 및 응답 형태

- -SETP Move 명령 실행

Input-Map

	7	6	5	4	3	2	1	0
D000	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0000b

+STEP bit = 0, -STEP bit = 1

Output-Map

	7	6	5	4	3	2	1	0
D100	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

STEP_RESP bit = 1, MOTIONING bit = 1

- + SETP Move 명령 실행

Input-Map

	7	6	5	4	3	2	1	0
D000	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0000b

+STEP bit = 1, -STEP bit = 0

Output-Map

	7	6	5	4	3	2	1	0
D100	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

STEP_RESP bit = 1, MOTIONING bit = 1

- -STEP, +STEP 비트가 '0'인 상태

Input-Map

	7	6	5	4	3	2	1	0
D000	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0000b

+STEP bit = 0, -STEP bit = 0

Output-Map

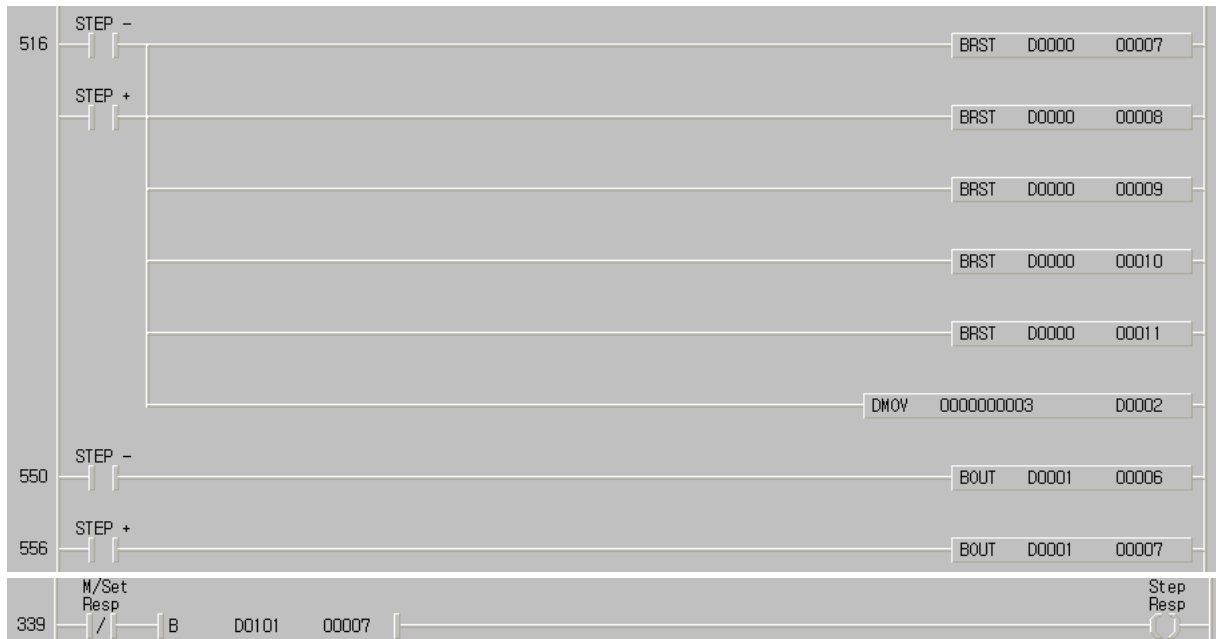
	7	6	5	4	3	2	1	0
D100	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

STEP_RESP bit = 0, MOTIONING bit = 0

✓ LADDER



NOTE 1: 예제 12 의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 입력에 따른 방향으로 10,000pps 로 조그 운전을 하도록 구성되었습니다.

✓ 명령 순서

- ① STEP- 또는 STEP+ 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0000b 로 설정하여 일반모션 모드로 설정합니다.
- ③ 한 스텝 이동거리의 저장 번호(STEP Distance No.)를 DMOV 명령으로 Command Data 영역에 입력합니다.
- ④ Input-Map 의 -STEP 또는 +STEP 비트가 '1'이 되도록 설정합니다.
- ⑤ 한 스텝의 이동 거리만큼 이동 후 MOTIONING 비트는 '0'이 됩니다.
- ⑥ 스텝 운전의 명령 응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 STEP_RESP 비트의 AND 조합으로 확인 합니다.

5.2.3 영점 이동 명령

영점 이동은 모션모드(MOTIONION)상태의 명령 코드(CMD_CODE) '0'에서 동작하며, 입력 데이터(D0)의 값은 무시되며 이동합니다. 예제 14 는 Axis-0 의 영점 이동에 대한 예제 입니다.

예제 13. 영점이동 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 GO_ZERO_POS – D001.3

- Output-Map

MOTION/SETTING_RESP – D100.7
 MOTIONING – D101.1
 GO_ZERO_POS_RESP – D101.3

✓ 입출력 정보

- 입력 정보

Go Zero Position

- 출력 정보

MOTIONING
 Go Zero POS Resp

✓ IO-Map 의 명령 및 응답 형태

- 영점 이동 명령을 실행

Input-Map							
D000	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D001	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D002	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D003	7	6	5	4	3	2	1
	0	1	2	3	4	5	6

MOTION/SETTING bit = 0

CMD_CODE = 0000b

GO_ZERO_POS bit = 1

Output-Map							
D100	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D101	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D102	7	6	5	4	3	2	1
	0	1	2	3	4	5	6
D103	7	6	5	4	3	2	1
	0	1	2	3	4	5	6

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

GO_ZERO_POS_RESP bit = 1,

MOTIONING bit = 1

- 영점 이동 중 GO_ZERO_POS 비트를 해제하여도, 영점 이동상태는 유지

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0000b

GO_ZERO_POS bit = 0

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

GO_ZERO_POS_RESP bit = 1,

MOTIONING bit = 1

- 영점 위치에서 GO_ZERO_POS bit 를 세트 하였을 때

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0000b

RESPONSE_TYPE = 0002b

GO_ZERO_POS bit = 0

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0000b

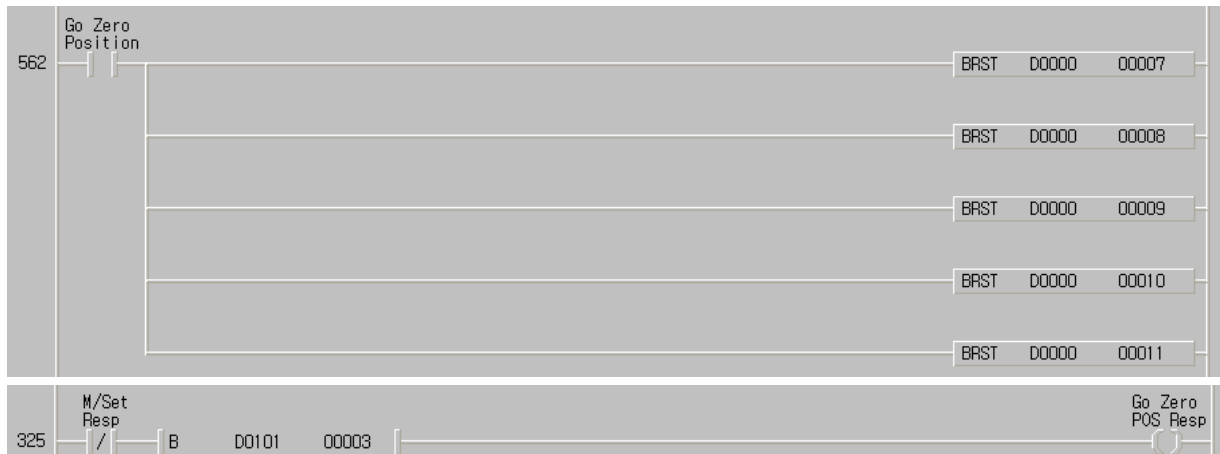
RESPONSE_TYPE_RESP = 0002b

GO_ZERO_POS_RESP bit = 1,

MOTIONING bit = 1

RESPONSE_DATA = Actual Position (0)

✓ LADDER



NOTE: 예제 13 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① Go Zero Position 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0000b 로 설정하여 일반모션 모드로 설정합니다.
- ③ Input-Map 의 GO_ZERO_POS 비트가 '1'이 되도록 설정합니다.
- ④ MOTIONING 비트는 영점이동이 시작되면 '1'이 되고, 영점위치에 도달하면 '0'으로 클리어 됩니다.
- ⑤ 영점이동의 명령 응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 GO_ZERO_POS_RESP 비트의 AND 조합으로 확인 합니다.

5.2.4 위치 이동 명령

상대위치 이동은 모션모드(MOTIONION/SETTING = 0)상태의 명령 코드(CMD_CODE) '1'에서 동작하며, 입력 데이터(D0)에 위치값으로 상대치 또는 절대치 이동을합니다.

■ 상대 위치 이동

상대 위치 이동은 입력된 위치 값으로 상대이동을 하는 명령입니다. 예제 14 은 Axis-0 의 상대위치 이동에 대한 예제 입니다.

예제 14. 상대위치 이동 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 INC/ABS – D001.8
 CMD_START – D000.4
 Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 MOTIONING – D101.1
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

INC Move
 Incremental Position (입력 범위 : -2,147,483,648 ~ -2,147,483,647 [DWORD])

- 출력 정보

MOTIONING
 POS MOV Resp

✓ IO-Map 의 명령 및 응답 형태

- INC Move 명령 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0001b

INC/ABS bit = 0

Command Data = 상대 위치 값

CMD_START bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0001b

CMD_RESP bit = 1, MOTIONING bit = 1

- INC Move 명령 후 CMD_START 비트의 해제

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0001b

INC/ABS bit = 0

Command Data = 상대 위치 값

CMD_START bit = 0

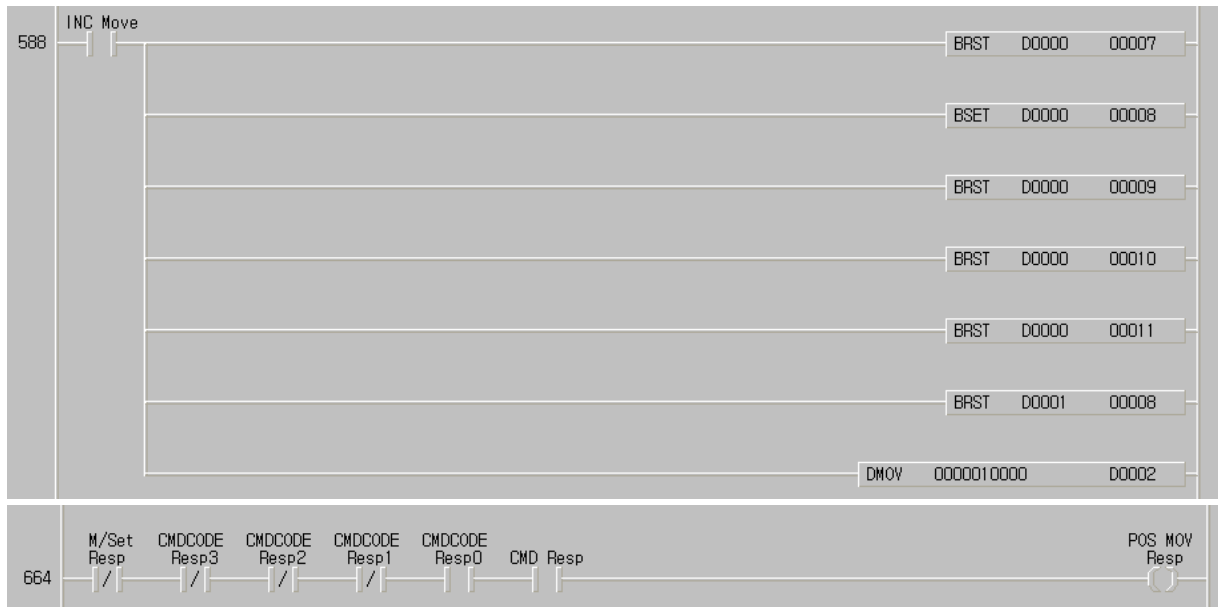
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0001b

CMD_RESP bit = 0, MOTIONING bit = 1

✓ LADDER



NOTE 1: 예제 12의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 10,000[pulse] 상대이동을 하도록 구성되었습니다.

✓ 명령 순서

- ① INC Move 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0001b 로 설정하여 위치이동 모드로 설정합니다.
- ③ INC/ABS 비트가 '0'이 되도록 설정 합니다.
- ④ 상대적 이동지점(Incremental Position)을 DMOV 명령으로 Command Data 영역에 입력 합니다.
- ⑤ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정합니다.
- ⑥ 상대 이동 지점으로 이동 후 MOTIONING 비트는 '0'이 됩니다.
- ⑦ 상대 이동의 명령 응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 CMD_RESP, CMD_CODE_RESP 비트의 AND 조합으로 확인 합니다.

■ 절대 위치 이동

절대 위치 이동은 입력 데이터(D002)에 위치값으로 절대치 이동을 합니다. 예제 15. 는 0 번축의 절대위치 이동에 대한 예제입니다.

예제 15. 절대위치 이동 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 INC/ABS – D001.8
 CMD_START – D000.4
 Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 MOTIONING – D101.1
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

ABS Move
 Absolute Position (입력 범위 : -2,147,483,648 ~ -2,147,483,647 [DWORD])

- 출력 정보

MOTIONING
 POS MOV Resp

✓ IO-Map 의 명령 및 응답 형태

- ABS Move 명령 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0001b

INC/ABS bit = 1

Command Data = 절대 위치 값

CMD_START bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0001b

CMD_RESP bit = 1, MOTIONING bit = 1

- ABS Move 명령 후 CMD_START 비트의 해제

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0001b

INC/ABS bit = 1

Command Data = 절대 위치 값

CMD_START bit = 0

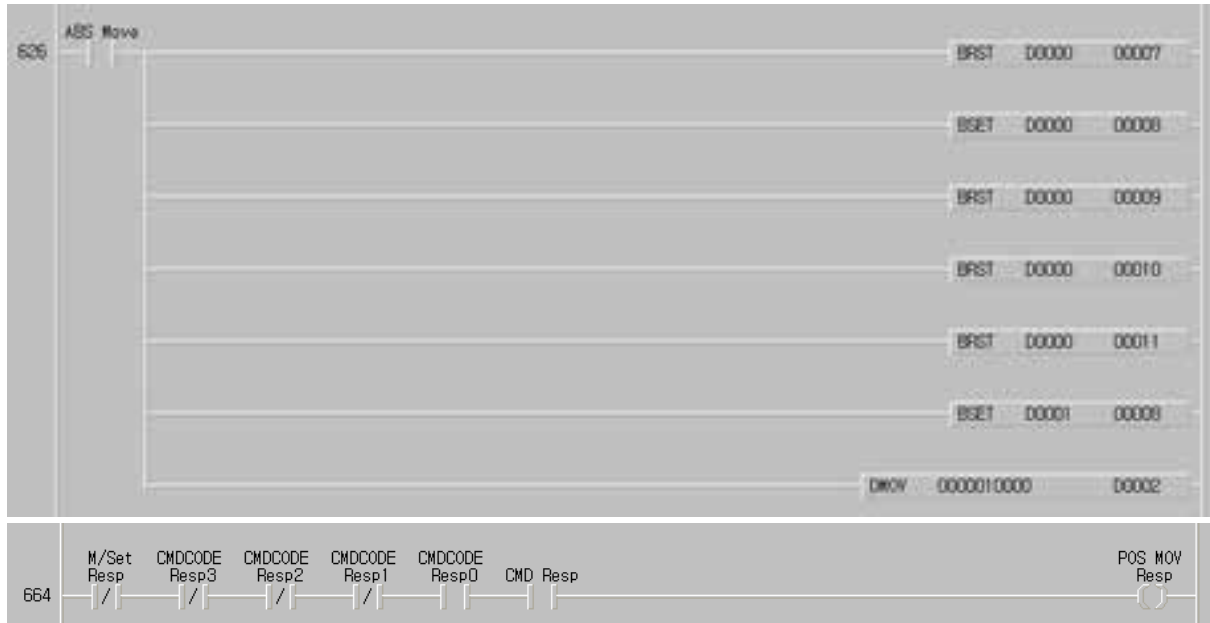
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0001b

CMD_RESP bit = 0, MOTIONING bit = 1

✓ LADDER



NOTE 1: 예제 15 의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 절대위치 10,000[pulse] 지점으로 이동 하도록 구성되었습니다.

✓ 명령 순서

- ① ABS Move 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0001b 로 설정하여 위치 이동 모드로 설정합니다.
- ③ INC/ABS 비트가 '1'이 되도록 설정 합니다.
- ④ 절대 위치(Absolute Position)을 DMOV 명령으로 Command Data 영역에 입력 합니다.
- ⑤ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정합니다.
- ⑥ 절대 위치 지점으로 이동 후 MOTIONING 비트는 '0'이 됩니다.
- ⑦ 절대 이동의 명령응답은 Output-Map 의 MOTION/SETTING_RESP 비트와 CMD_RESP, CMD_CODE_RESP 비트의 AND 조합으로 확인 합니다.

■ 위치 이동 명령에 대한 응답확인

상대 위치 이동과 절대 위치 이동 명령에 대한 응답 비트는 조그운전 또는 스텝이동, 영점 이동과 같이 IO-Map 으로 루프-백 되지 않습니다. 그러므로, Input-Map 에 설정한 비트의 응답 비트의 조합으로 명령을 확인 할 수 있습니다.

5.2.5 PT 운전 명령

PT 운전은 모션모드(MOTIONION/SETTING = 0)상태의 명령 코드(CMD_CODE) '4'에서 동작하며, 입력 데이터(D0)에 PT 항목의 번호를 입력하여 운전을 실행 합니다.

■ 일반 PT 운전

일반 PT 운전은 입력된 값(D002[DWORD])부터 PT 운전을 시작합니다. 예제 16 은 Axis-0 의 PT 운전 명령에 대한 예제입니다.

예제 16. PT 운전 명령

✓ 제어 비트맵 정보

- **Input-Map**
 - MOTION/SETTING – D000.7
 - CMD_CODE – D000.8~D000.11
 - SINGLE_PT – D001.12
 - CMD_START – D000.4
 - Command Data Area – D002~D003 (D002 [1 DWORD])
- **Output-Map**
 - MOTION/SETTING_RESP – D100.7
 - CMD_CODE_RESP – D100.8~D100.11
 - MOTIONING – D101.1
 - CMD_RESP – D100.4
 - PT_RUNUNG – D101.8

✓ 입출력 정보

- **입력 정보**
 - PT Run
 - PT No. (PT 항목이 저장 되어있는 번호 [DWORD])
- **출력 정보**
 - MOTIONING
 - PT CMD Resp
 - PT Running

✓ IO-Map 의 명령 및 응답 형태

- PT RUN 명령 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0100b

SINGLE_PT bit = 0

Command Data = PT 번호

CMD_START bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0100b

CMD_RESP bit = 1, PT_RUNNING bit = 1

- PT RUN 명령 후 CMD_START 비트의 해제

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0100b

SINGLE_PT bit = 0

Command Data = PT 번호

CMD_START bit = 0

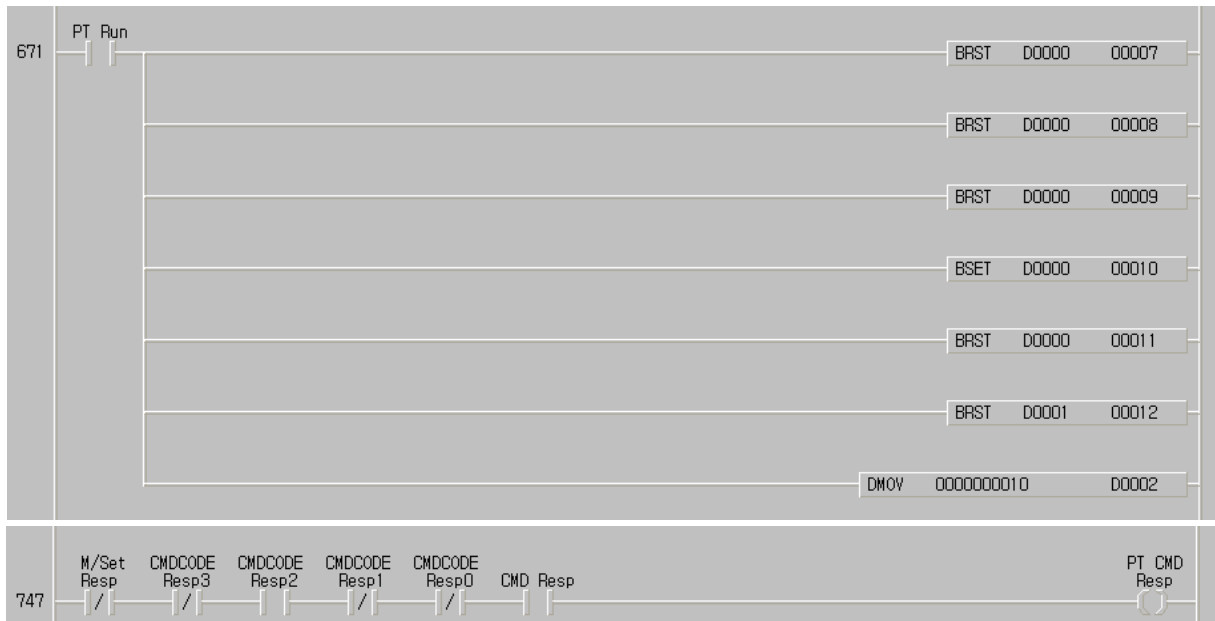
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0100b

CMD_RESP bit = 0, PT_RUNNING bit = 1

✓ LADDER



NOTE 1: 예제 16의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 PT 번호 10에서부터 PT운전을 시작하도록 구성되었습니다.

fff

✓ 명령 순서

- ① PT Run 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE의 값을 0100b로 설정하여 Position Table 운전 모드로 설정합니다.
- ③ SINGLE_PT 비트가 '0'이 되도록 설정 합니다.
- ④ 운전을 시작할 PT 번호를 DMOV 명령으로 Command Data 영역에 입력 합니다.
- ⑤ Input-Map의 CMD_START 비트가 '1'이 되도록 설정합니다.
- ⑥ PT 운전이 시작하면서 PT_RUNNING 비트는 '1'이 됩니다.
- ⑦ PT운전을 정지 하기 위해서는 예제 4의 CANCEL 명령을 실행합니다.

■ 싱글 PT 운전

싱글 PT 운전은 입력된 값(D002)부터 하나의 PT 항목에 대하여 운전합니다. 예제 19 는 0 번축의 싱글 PT 운전에 대한 예제입니다.

예제 17. 싱글 PT 운전 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 SINGLE_PT – D001.12
 CMD_START – D000.4
 Command Data Area – D002~D003 (D002 [1 DWORD])

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 MOTIONING – D101.1
 CMD_RESP – D100.4
 PT_RUNUNG – D101.8

✓ 입출력 정보

- 입력 정보

Single PT Run
 PT No. (PT 항목이 저장 되어있는 번호 [DWORD])

- 출력 정보

MOTIONING
 PT CMD Resp
 PT Running

✓ IO-Map 의 명령 및 응답 형태

- 싱글 PT RUN 명령 실행

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0100b

SINGLE_PT bit = 1

Command Data = PT 번호

CMD_START bit = 1

		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0100b

CMD_RESP bit = 1, PT_RUNNING bit = 1

- 싱글 PT RUN 명령 후 CMD_START 비트의 해제

		Input-Map							
D000		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D001		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D002									
D003									

MOTION/SETTING bit = 0

CMD_CODE = 0100b

SINGLE_PT bit = 1

Command Data = PT 번호

CMD_START bit = 0

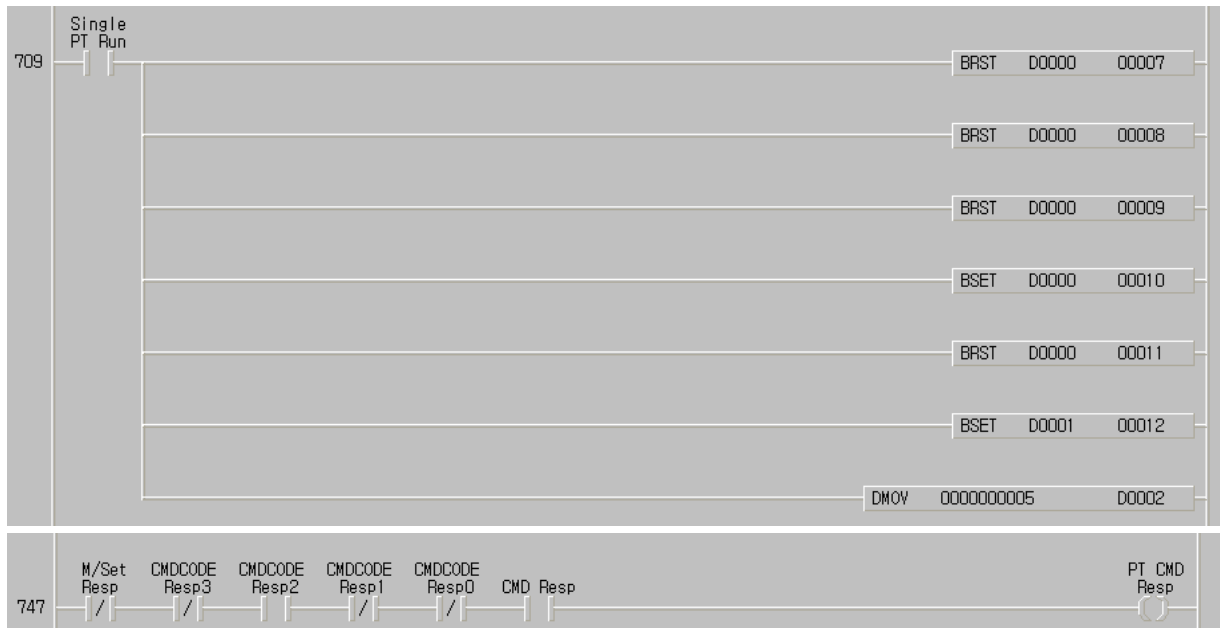
		Output-Map							
D100		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D101		7	6	5	4	3	2	1	0
		15	14	13	12	11	10	9	8
D102									
D103									

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0100b

CMD_RESP bit = 0, PT_RUNNING bit = 1

✓ LADDER



NOTE 1: 예제 15 의 래더는 앞의 예제에서 추가된 명령입니다.

NOTE 2: 본 예제에서는 PT 번호 5 번의 항목만을 운전을 하도록 구성되었습니다.

✓ 명령 순서

- ① Single PT Run 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0100b 로 설정하여 Position Table 운전 모드로 설정합니다.
- ③ SINGLE_PT 비트가 '1'이 되도록 설정 합니다.
- ④ 하나의 PT 운전을 실행할 PT 번호를 DMOV 명령으로 Command Data 영역에 입력 합니다.
- ⑤ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정합니다.
- ⑥ PT 운전이 시작하면서 PT_RUNNING 비트는 '1'이 됩니다.

■ PT 운전 명령에 대한 응답확인

PT 운전의 명령의 응답 확인 방법은 위치 이동 명령과 마찬가지로 IO-Map 으로 루프-백 되지 않습니다. 그러므로, Input-Map 에 설정한 비트에 대한 Output-Map 의 응답 비트의 조합으로 명령을 확인 할 수 있습니다.

5.2.6 원점 이동 명령

원점 이동 명령은 모션모드(MOTIONION/SETTING = 0)상태의 명령 코드(CMD_CODE) '7'에서 동작하며, 입력 데이터(D0)에 입력된 값과 무관하게 동작합니다. 예제 18. 은 Axis-0 의 원점이동 명령 대한 예제입니다.

예제 18. 원점 이동 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 MOTIONING – D101.1
 CMD_RESP – D100.4
 PT_RUNUNG – D101.8

✓ 입출력 정보

- 입력 정보

ORIGIN Search

- 출력 정보

MOTIONING
 Origin Mov Resp

✓ IO-Map 의 명령 및 응답 형태

- 원점 이동 실행

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D002								
D003								

MOTION/SETTING bit = 0

CMD_CODE = 0111b

CMD_START bit = 1

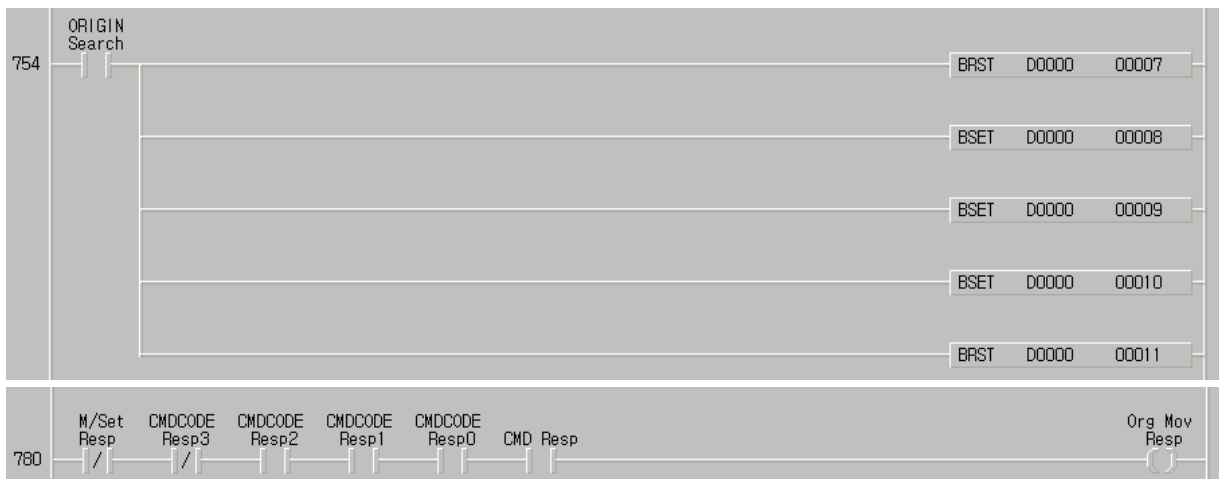
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102								
D103								

MOTION/SETTING_RESP bit = 0

CMD_CODE_RESP = 0111b

CMD_RESP bit = 1, PT_RUNNING bit = 1

✓ LADDER



NOTE 1: 예제 15 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① ORIGIN Search 입력의 Close 신호로 MOTION/SETTING 비트를 '0'으로 설정하여 IO-Map 을 모션모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0111b 로 설정하여 원점이동 명령 모드로 설정합니다.
- ③ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정합니다.
- ④ 원점이동 운전이 시작하면서 MOTOINING 비트는 '1'이 됩니다.
- ⑤ 원점이동은 드라이버의 파라미터에 지정된 방법으로 동작 합니다.

5.2.7 파라미터 설정

파라미터 설정은 해당 축의 모터 드라이브에 대한 파라미터와 PT 정보의 확인 및 설정, 저장을 할 수 있습니다.

■ 파라미터 확인 명령

파라미터 확인은 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '8' 에서 동작하며, 인덱스 영역(K4Y10)에 입력된 파라미터 값을 요청하는 명령입니다. 이때 입력 데이터(D0)에 입력된 값과 무관하게 동작합니다. 예제 19. 은 Axis-0 의 파라미터 확인 명령에 대한 예제입니다.

예제 19. 파라미터 확인 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 INDEX_VALUE– D001 (1WORD)
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 INDEX_VALUE_RESP – D101 (1WORD)
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

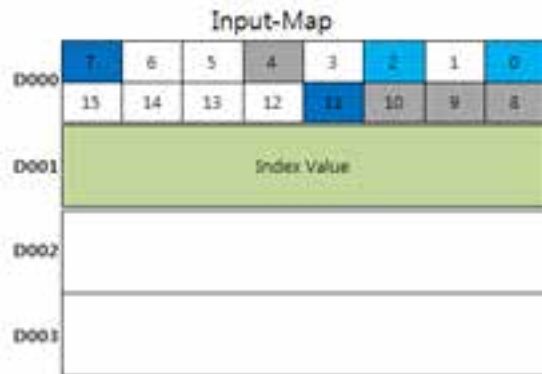
Read Parameter

- 출력 정보

Read Parameter Value

✓ IO-Map 의 명령 및 응답 형태

- 원점 이동 실행

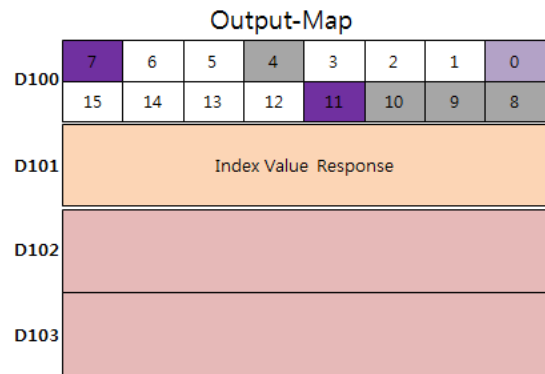


MOTION/SETTING bit = 1

CMD_CODE = 1000b

INDEX_VALUE = Index Value (Parameter No)

CMD_START bit = 1



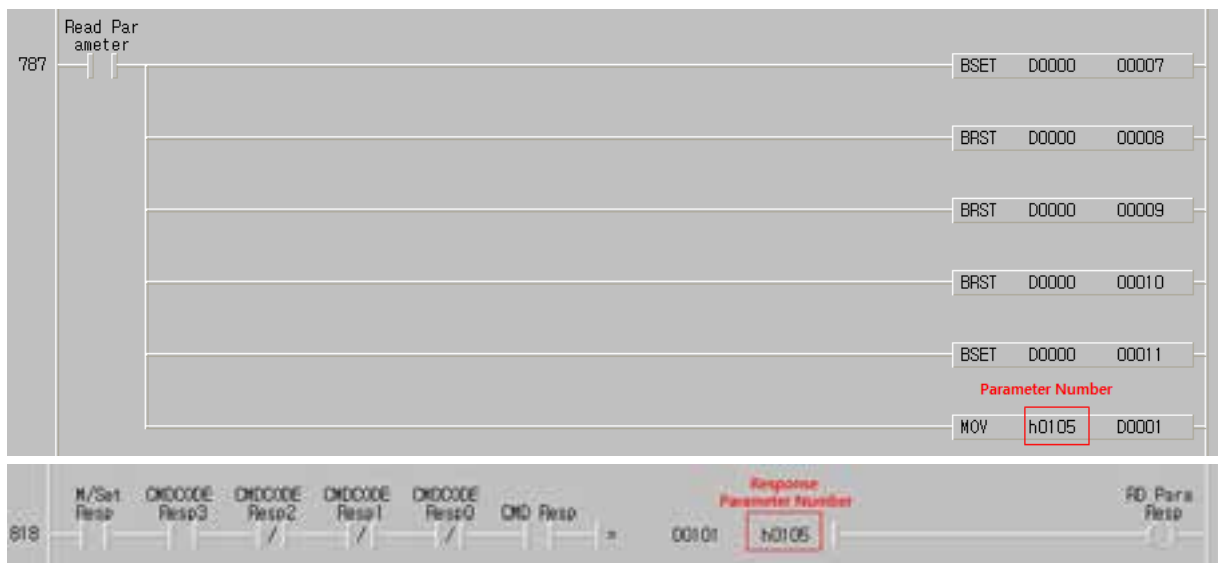
MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 1000b

INDEX_VALUE_RESP = Index Value (Parameter No)

CMD_RESP bit = 1

✓ LADDER

**NOTE 1:** 예제 19 의 래더는 앞의 예제에서 추가된 명령입니다.**NOTE 2:** 본 예제는 파라미터 PN#105 『Move Speed for Step Move 』의 인덱스 번호인 0x0105의 값을 요청한 래더입니다.

✓ 명령 순서

- ① Read Parameter 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1000b 가 되게 하여 파라미터 요청 명령 모드로 설정합니다.
- ③ INDEX_VALUE 에 요청할 파라미터 또는 PT 항목의 인덱스 값을 입력합니다.
- ④ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 파라미터 값을 요청합니다.
- ⑤ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 그리고 Input-Map 에 입력한 인덱스 값과 응답된 인덱스 값을 비교 결과를 AND 하여 응답 데이터를 확인합니다.

■ 파라미터 변경 명령

파라미터 변경은 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '9' 에서 동작하며, 인덱스 영역(K4Y10)에 작성된 파라미터 번호에 입력 데이터영역(D0)에 지정된 값으로 변경하는 명령입니다. 예제 23 은 0 번축의 파라미터 변경 명령에 대한 예제입니다.

예제 20. 파라미터 변경 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 INDEX_VALUE – D001 [1 WORD]
 COMMAND_WORD_DATA – D001~D002 [1 DWORD]
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 INDEX_VALUE_RESP – D101 (1WORD)
 CMD_RESP – D100.4
 RESPONSE_DATA – D102~D103 (D102 [1 DWORD])

✓ 입출력 정보

- 입력 정보

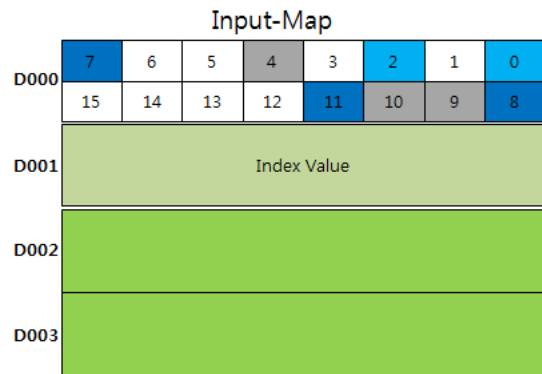
Write Parameter

- 출력 정보

Write Parameter Value

✓ IO-Map 의 명령 및 응답 형태

- 파라미터 값 변경 실행



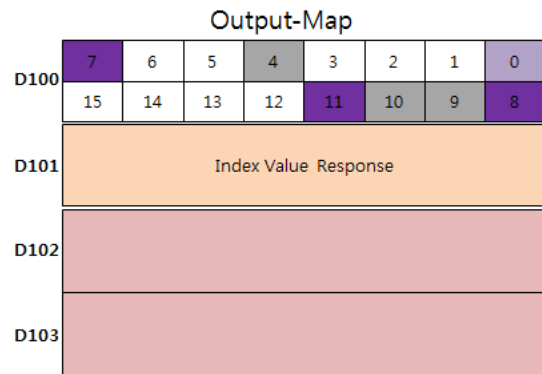
MOTION/SETTING bit = 1

CMD_CODE = 1001b

INDEX_VALUE = Index Value (Parameter No)

COMMAND_WORD_DATA = Parameter Value

CMD_START bit = 1



MOTION/SETTING_RESP bit = 1

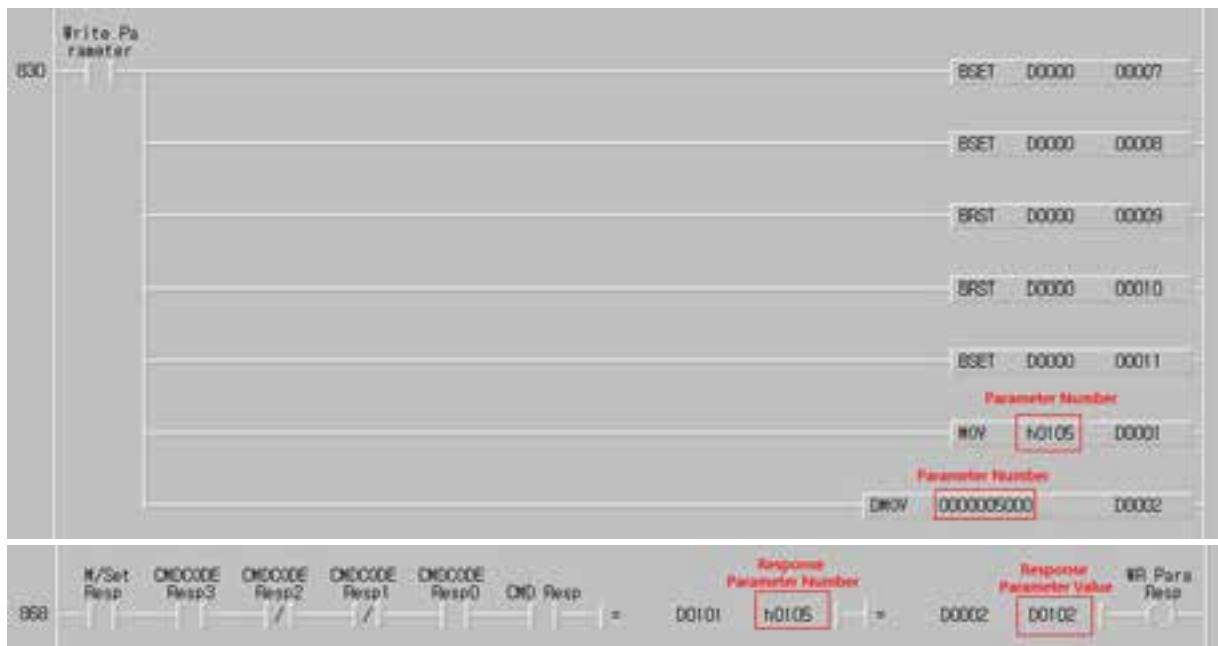
CMD_CODE_RESP = 1001b

INDEX_VALUE_RESP = Index Value (Parameter No)

RESPONSE_DATA = Parameter Value

CMD_RESP bit = 1

✓ LADDER

**NOTE 1:** 예제 20 의 래더는 앞의 예제에서 추가된 명령입니다.**NOTE 2:** 본 예제는 파라미터 PN#105 『Move Speed for Step Move 』의 인덱스 번호인 0x0105 의 값을 5000pps 로 변경하는 래더입니다.

✓ 명령 순서

- ① Write Parameter 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1001b 가 되게 하여 파라미터 변경 명령 모드로 설정합니다.
- ③ INDEX_VALUE 에 변경할 파라미터 또는 PT 항목의 인덱스 값을 입력합니다.
- ④ COMMAND_WORD_DATA 에 변경할 파라미터의 값을 입력합니다.
- ⑤ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 파라미터 값을 요청합니다.
- ⑥ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 그리고 Input-Map 에 입력한 인덱스 값과 응답된 인덱스 값을 비교 결과를 AND 하여 응답 데이터를 확인합니다.
- ⑦ 파라미터의 변경명령이 실행되면, RESPONSE_DATA 영역으로 변경된 파라미터의 값이 응답합니다.

■ 파라미터 저장

파라미터 저장은 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '14' 에서 동작하여 ROM 영역에 파라미터 값을 저장 합니다. 예제 21. 은 0 번축의 파라미터 저장 명령에 대한 예제입니다.

예제 21. 파라미터 저장 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 INDEX_VALUE – D001 [1 WORD]
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 INDEX_VALUE_RESP – D101 (1WORD)
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

Save Parameter

- 출력 정보

Save Parameter Resp

✓ IO-Map 의 명령 및 응답 형태

- 파라미터 저장 명령

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	0							
D002								
D003								

MOTION/SETTING bit = 1

CMD_CODE = 1110b

CMD_START bit = 1

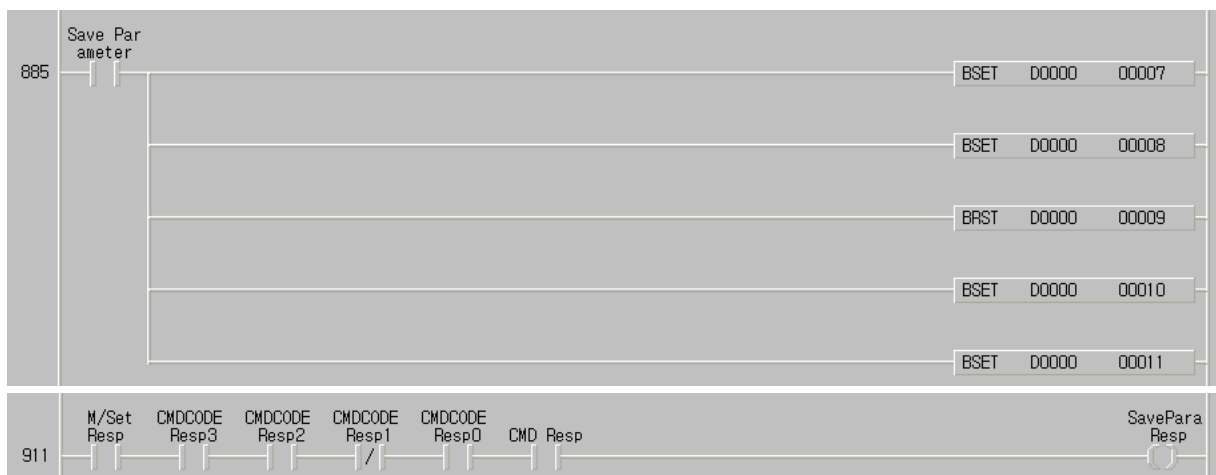
Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	0							
D102								
D103								

MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 1110b

CMD_RESP bit = 1

✓ LADDER

**NOTE :** 예제 21 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① Save Parameter 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1110b 가 되게 하여 파라미터 저장 명령 모드로 설정합니다.
- ③ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 파라미터 값을 요청합니다.
- ④ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 을 AND 하여 확인합니다.

5.2.8 위치 지정

위치 지정 명령은 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '10' 에서 동작하여 추종중인 위치값(Command Position)을 변경합니다. 예제 22. 은 Axis-0 의 추종중인 위치값 변경 명령에 대한 예제입니다.

예제 22. 위치값 변경 명령

✓ 제어 비트맵 정보

- **Input-Map**
 MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 COMMAND_WORD_DATA – D001~D002 [1 DWORD]
 CMD_START – D000.4
- **Output-Map**
 MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 CMD_RESP – D100.4
 RESPONSE_DATA – D102~D103 (D102 [1 DWORD])

✓ 입출력 정보

- **입력 정보**
 Set Position
 Set Current Position Data
- **출력 정보**
 Response Current Position

✓ IO-Map 의 명령 및 응답 형태

- 현재 위치 지정 명령

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	0							
D002								
D003								

MOTION/SETTING bit = 1

CMD_CODE = 1010b

COMMAND_WORD_DATA = Set Current Position

CMD_START bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	0							
D102								
D103								

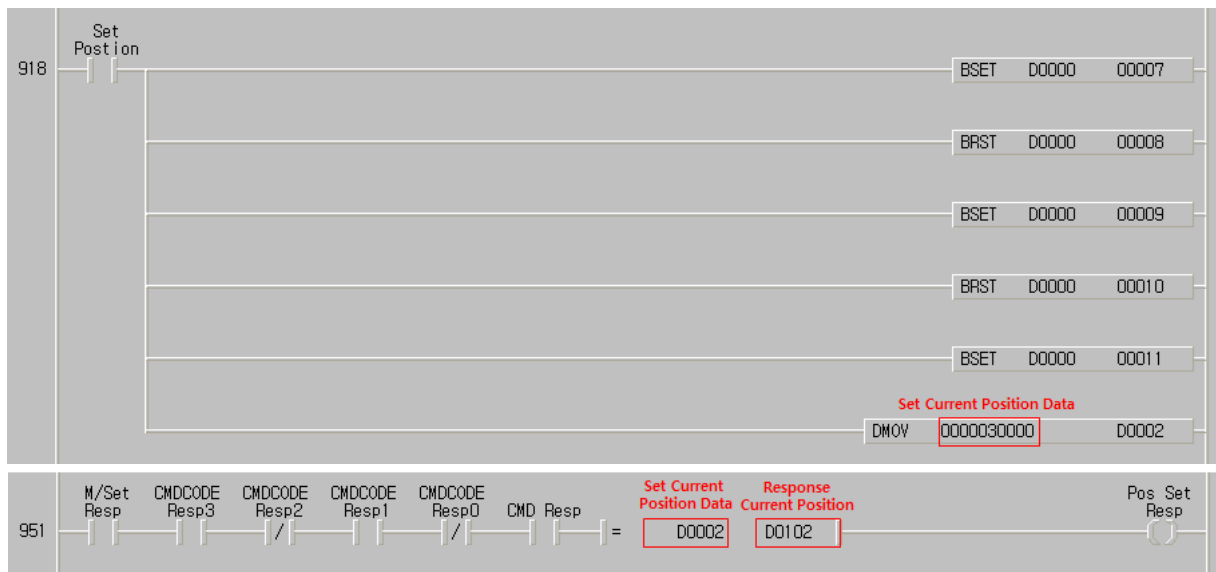
MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 1010b

RESPONSE_DATA = Response Current Position

CMD_RESP bit = 1

✓ LADDER

**NOTE 1:** 예제 22 의 래더는 앞의 예제에서 추가된 명령입니다.**NOTE 2:** 본 예제는 현재 위치를 30,000[pulse] 지점으로 설정하는 명령입니다.

✓ 명령 순서

- ① Set Position 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정 하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1010b 가 되게 하여 파라미터 변경 명령으로 설정합니다.
- ③ COMMAND_WORD_DATA 에 변경할 위치 값을 입력합니다.
- ④ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 위치 지정을 실행합니다.
- ⑤ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 그리고 Input-Map 에 입력한 위치 지정 값과 응답된 현재 위치 값을 비교한 결과를 AND 하여 응답 데이터를 확인합니다.
- ⑥ 현재 위치 지정 명령을 실행 하면 지정된 값으로 드라이브에 저장이 되었다가, 다시 현재 위치를 읽어 응답합니다. 따라서, 이 명령을 실행 시 모터의 회전축이 동작 되지 않도록 한 상태에서 실행해야 합니다.

5.2.9 알람 내역 확인 및 초기화

모션게이트는 최근에 발생한 알람 부터 4 개에 대한 알람내역을 관리 할 수 있습니다.

■ 알람 내역 확인

알람 내역 확인은 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '12' 에서 동작하여 해당축에 발생한 알람 내역(Alarm History)를 확인 할 수 있습니다. 예제 26. 은 Axis-0 의 알람 내역 확인에 대한 예제입니다.

예제 23. 알람 내역 확인

✓ 제어 비트맵 정보

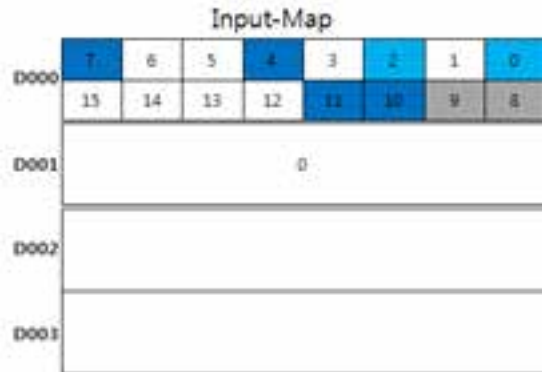
- **Input-Map**
 MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 CMD_START – D000.4
- **Output-Map**
 MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 CMD_RESP – D100.4
 RESPONSE_DATA – D102~D103 (D102 [1 DWORD])

✓ 입출력 정보

- **입력 정보**
 ALM History
- **출력 정보**
 ALM His Resp

✓ IO-Map 의 명령 및 응답 형태

- 알람 내역 확인 명령 실행



MOTION/SETTING bit = 1

CMD_CODE = 1100b

CMD_START bit = 1



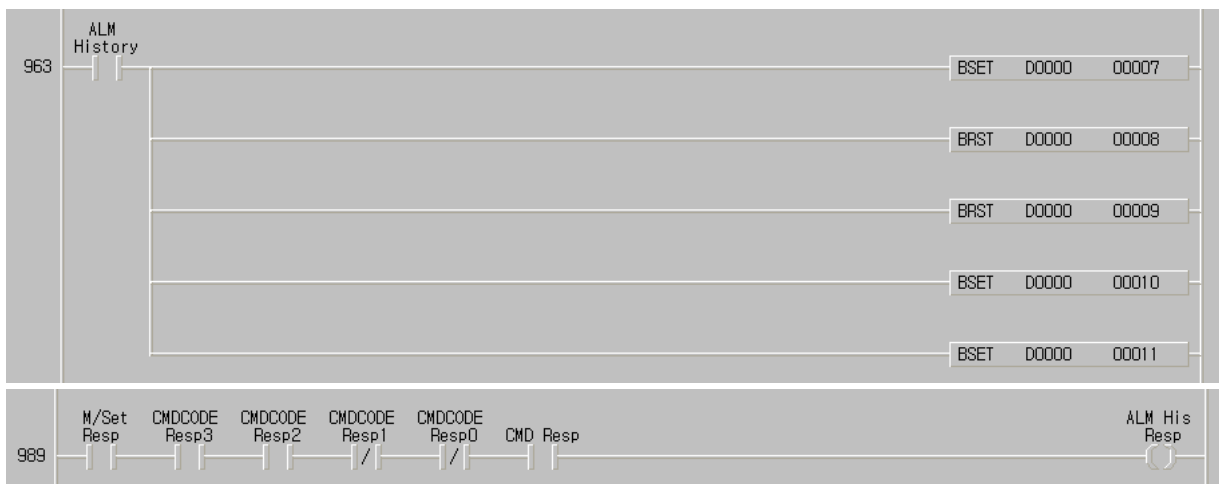
MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 1100b

CMD_RESP bit = 1

RESPONSE_DATA = Alarm History

✓ LADDER

**NOTE:** 예제 23 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① ALM History 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정 하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1100b 가 되게 하여 알람내역 요청 명령으로 설정합니다.
- ③ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 위치 지정을 실행합니다.
- ④ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 를 AND 로 구성 하여 확인합니다.
- ⑤ 알람 내역 요청 명령을 실행 하면 가장 최근의 알람코드부터 4 개의 알람내역을 응답합니다.

■ 알람 내역 초기화

알람 내역 초기화는 설정모드(MOTIONION/SETTING=1)상태의 명령 코드(CMD_CODE) '13' 에서 동작하여 해당축에 발생한 알람 내역(Alarm History)를 확인 할 수 있습니다. 예제 27 은 Axis-0 의 알람 내역 초기화에 대한 예제입니다.

예제 24. 알람 내역 초기화 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

Reset ALM His

- 출력 정보

ALM His CLR Resp

✓ IO-Map 의 명령 및 응답 형태

- 알람 내역 초기화 명령 실행

Input-Map							
	7	6	5	4	3	2	1
D000	15	14	13	12	11	10	9
D001	0						
D002							
D003							

MOTION/SETTING bit = 1

CMD_CODE = 1110b

CMD_START bit = 1

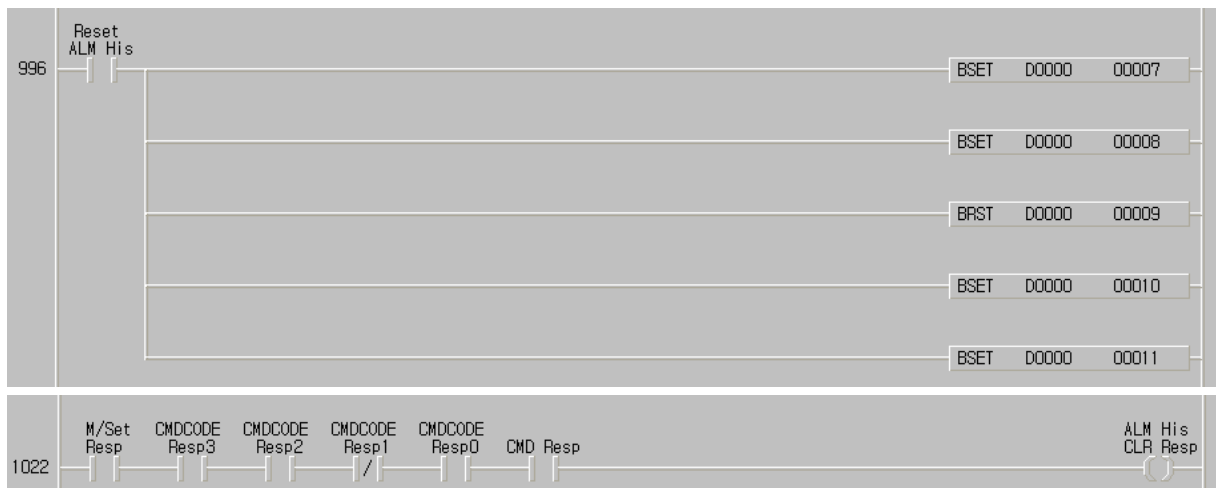
Output-Map							
	7	6	5	4	3	2	1
D100	15	14	13	12	11	10	9
D101	7	6	5	4	3	2	1
	15	14	13	12	11	10	9
D102							
D103							

MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 1110b

CMD_RESP bit = 1

✓ LADDER

**NOTE :** 예제 21 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① Reset ALM His 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정 하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 1101b 로 하여 알람내역 초기화 명령으로 설정합니다.
- ③ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 알람내역을 초기화합니다.
- ④ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 을 AND 하여 확인합니다.

5.2.10 모션게이트 버전 확인

모션게이트의 버전 확인 명령은 어느 축에서 명령을 실행하여도 모션게이트의 버전 정보를 얻을 수 있습니다. 이 명령으로 설정모드(MOTIONION/SETTING=1)상태의 명령 코드 (CMD_CODE) '5' 에서 동작하여 확인할 수 있습니다. 예제 25. 는 모션게이트의 버전 정보 요청 방법에 대한 예제입니다.

예제 25. 모션게이트의 버전 정보 요청 명령

✓ 제어 비트맵 정보

- Input-Map

MOTION/SETTING – D000.7
 CMD_CODE – D000.8~D000.11
 CMD_START – D000.4

- Output-Map

MOTION/SETTING_RESP – D100.7
 CMD_CODE_RESP – D100.8~D100.11
 CMD_RESP – D100.4

✓ 입출력 정보

- 입력 정보

Read Ver Info

- 출력 정보

Read Ver Resp

✓ IO-Map 의 명령 및 응답 형태

- 버전 정보 요청 명령

Input-Map								
D000	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D001	0							
D002								
D003								

MOTION/SETTING bit = 1

CMD_CODE = 0101b

CMD_START bit = 1

Output-Map								
D100	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D101	7	6	5	4	3	2	1	0
	15	14	13	12	11	10	9	8
D102	Release No							
	Bug Fix							
D103	Minor Version							
	Major Version							

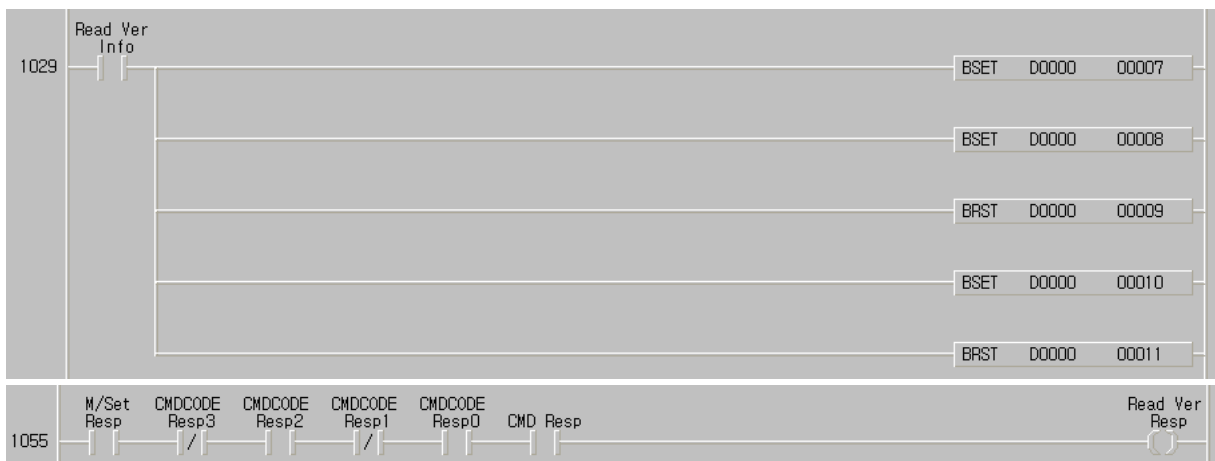
MOTION/SETTING_RESP bit = 1

CMD_CODE_RESP = 0101b

CMD_RESP bit = 1

RESPONSE_DATA = Version Information

✓ LADDER

**NOTE :** 예제 21 의 래더는 앞의 예제에서 추가된 명령입니다.

✓ 명령 순서

- ① Read Ver Info 입력의 Close 신호로 MOTION/SETTING 비트를 '1'이 되도록 설정 하여 IO-Map 을 설정모드 상태로 전환 합니다.
- ② CMD_CODE 의 값을 0101b 로 하여 버전 정보 요청 명령 모드로 설정합니다.
- ③ Input-Map 의 CMD_START 비트가 '1'이 되도록 설정하여 버전 정보 요청 명령을 실행 합니다.
- ④ 응답 데이터는 MOTION/SETTING_RESP, CMD_CODE_RESP, CMD_RESP 을 AND 하여 확인합니다.



- 사용자설명서의 일부 또는 전부를 무단 기재하거나 복제하는 것은 금지되어 있습니다.
- 손상이나 분실 등으로 사용자설명서가 필요할 때에는 본사 또는 가까운 대리점에 문의하여 주십시오.
- 사용자설명서는 제품의 개량이나 사양변경 및 사용자 설명서의 개선을 위해서 예고 없이 변경되는 경우가 있습니다.
- Ezi-MOTIONGATE 는 국내에 등록된 FASTECH Co.LTD,의 등록상표입니다.

© Copyright 2008 FASTECH Co.,Ltd.
All Rights Reserved. Jul 25 2012 Rev.01.02.00

www.fastech.co.kr