OMRON

User Manual

ML3-IO Setup
(Oriental Drive)

MechatroLink IO Interface

Sep 25, 2015 (REV 1.0.0)

REVISION HISTORY			
VERSION	**DESCRIPTION**	**DATE**	**APPVD**
1.0.0	Manual 초본작성	2015.09.25	이명성

※ 본 매뉴얼은 필요에 따라 수시로 업데이트 될 수 있으며, 본사 홈페이지에서 최신항목을 다운 받아 사용하시기 바랍니다.

Table of Contents

1. INTRODUCTION..... 4
2. HARDWARE SETUP..... 4
   2.1. Driver Station ID Setting..... 5
   2.2. Remote I/O 점유 크기 및 전송 Byte 설정..... 5
   2.3. 호기(모터)번호 설정 및 통신속도 설정..... 6
   2.4. ML3 Master SW Setting..... 7
3. MECHATROLINK EXPLORER..... 9
4. VARIABLES..... 13
   4.1. M Variable Mapping..... 13
   4.2. Control Registers..... 15
   4.3. Remote I/O..... 16
   4.4. Remote Register..... 20

1. INTRODUCTION

본 매뉴얼은 UMAC 또는 Cruiser에 장착된 MechatroLink Master에서 Oriental 사의 ML3 Drive를 사용하기 위한 방법을 설명합니다.

Oriental 사의 ML3 Drive는 일반적인 ML3 Drive에서 사용하는 Servo Profile이 아닌 I/O Profile을 사용합니다.

ML3-IO를 사용하기 위해서는 반드시 아래 사항을 충족하여야 합니다.

ML3 Master Firmware Ver : 1.98 이상

PMAC Suit Ver : 6.3.33 이상 (Build date 9/24/2015 이후)

2. HARDWARE SETUP

- UMAC의 ML3 Master 또는 Cruiser의 ML3 포트와 NETC01-M3의 ML3 포트와 연결됩니다.
- Station(NETC01-M3)하나당 최대 8개의 모터를 연결할 수 있습니다.

2.1. Driver Station ID Setting

- NETC01-03의 STATION No. Rotary SW로 Station id를 설정합니다.

- 설정한 Station ID는 PMAC Suit를 사용해 ML3 Setup시 MechatroLink Explore를 통해 동일하게 설정되어야 합니다.

- Station ID를 변경하면 반드시 Drive 전원을 재인가 해주시기 바랍니다.

2.2. Remote I/O 점유 크기 및 전송 Byte 설정

SW2	내 용	출하 시 설정
No.1	Remote I/O 점유 크기를 설정합니다. OFF : 16Bit Mode(최대 8 축까지 접속 가능) ON : 8Bit Mode(최대 16 축까지 접속 가능)	OFF
No.2 No.3	전송 Byte 수를 설정합니다. No.2=OFF, No.3=OFF : 16Bytes No.2=OFF, No.3=ON : 32Bytes No.2=ON, No.3=OFF : 48Bytes No.2=ON, No.3=ON : 64Bytes	No.2=OFF No.3=ON

- NETC01-03의 **SW2는 반드시 16bit Mode, 48Byte 전송형식으로 설정**해 주시기 바랍니다.

2.3. 호기(모터)번호 설정 및 통신속도 설정

SW1	SW3-No.1	호기 번호 (Slave Address)
0	OFF	사용하지 않습니다
1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8

SW2	통신 속도(bps)
0	9600
1	19200
2	38400
3	57600
4	115,200
5~F	사용하지 않습니다

- 16Bit 모드 사용시 최대 사용 가능한 호기는 8개이며, SW1로 설정합니다.

- SW2로 설정하는 통신속도는 Oriental Dive와 모터간 RS485 통신 속도 입니다. ML3 통신 속도와는 무관합니다.

2.4. ML3 Master SW Setting

A. SW1: Base Address 설정 (Only for UMAC)

SW1 No.1~No.4를 사용하여 Base Address를 설정합니다.

SW1 No.5, No.6은 항상 On이 되어야 합니다.

1	2	3	4	Addr	1	2	3	4	Addr
On	Off	On	On	$68000	Off	On	On	On	$74000
		Off	On	$69000			Off	On	$75000
		On	Off	$6A000			On	Off	$76000
		Off	Off	$6B000			Off	Off	$77000

※ CRUISER에서는 Base Address가 $74000으로 고정되어 있습니다.

B. SW2: Servo Frequency 분주비 설정

- ML-III 사용시 Servo Frequency는 반드시 2의 배수로 설정해 주어야 합니다.
- **ML-III통신 주기는 2KHz입니다.** Servo Frequency가 2KHz가 아닐 경우 SW2를 사용하여 Clock을 분주해 주어야 합니다.

< CRUISER >

< UMAC ML-III Master>

Servo Frequency	Cruiser SW2
2kHz	0b0000
4kHz	0b0001
6kHz	0b0010
8kHz	0b0011
…	…
32kHz	0b1111

Servo Frequency	Umac SW2
2kHz	0b1111
4kHz	0b1110
6kHz	0b1101
8kHz	0b1100
…	…
32kHz	0b0000

C. SW3: Servo Frequency 출력 설정 (Only for UMAC)

- UMAC에서 Servo IC없이 ML-Ⅲ 카드 단독으로 제어하기 위해서는 SW 3으로 ML-III Servo Clock을 설정해 주어야 합니다.
- Servo IC가 있을 경우(ACC-24E2, 24E2S, 24E2A ...) SW3-3을 Receive로 설정해 주시기 바랍니다.

- S
- **SW3-1,.2**

SW3-1	SW3-2	Servo Frequency
Off	Off	2Khz
Off	On	4Khz
On	Off	6Khz
On	On	8Khz

- **SW3-3**

SW3-3	Clock Direction
On	Transmit
Off	Receive

3. MECHTROLINK EXPLORER

- MechatroLink Explorer를 통해 ML3-IO Setup을 진행합니다. MechatroLink Explore를 사용하기 위해서는 반드시 PMAC Suit를 사용하셔야 합니다. Pewin32에는 MechatroLink Explorer가 포함되어 있지 않습니다.

A. MechatroLink Explorer 실행

- Menu → View → MechatroLink Explorer를 실행합니다.

- MechatroLink Explorer를 실행시키면 현재 설치된 ML3 Master Card의 정보가 나오게 됩니다. ML3 Master **카드가 없거나 Base Address가 올바르게 설정되지 않으면**(다른 카드와 주소가 충돌하면) **Master Card의 정보가 표시되지 않습니다.**

- Master Card의 정보 표시 형식은 Master #A[#B Ver #C]
#A : Master Card NO, #B : Base Add, #C : Firmware Version

- Master Card의 **Base Address는 Cruiser의 경우 $74000**으로 고정되며 **UMAC은 Master Card의 SW1 설정**을 따릅니다.

- MechatroLink IO Interface를 사용하기 위해서는 반드시 **Firmware Ver 1.98 이상을 사용**하여야 합니다.

B. Station ID 설정

- First(Port3,4에 연결했을 경우 Second)를 우클릭한 후 Station ID를 설정합니다.
- NET01-M3의 Rotary SW(STATION No.)로 설정한 Station ID에 맞게 ID(X10)과 ID(X1)을 설정합니다.

- NET01-M3이 연결된 Buffer의 **Profile을 Servo에서 I/O로 변경**합니다.
- 변경 완료 후 **반드시 Save to Flash 버튼을 눌러야만 ML3 Master Card에 정보가 저장**됩니다.

C. Slave Scan

- Master 항목을 우클릭한 후 Master Scan을 누르면 연결된 Slave(NETC01-03)이 표시됩니다.

- Slave가 **Scan되지 않을 경우 Station ID 설정을 및 케이블의 접속상태를 다시 확인**하시기 바랍니다.

D. Configuration 및 Download

- Master 항목을 우클릭한 후 Configuration을 누릅니다.

- ML3 Slave가 IO(NET01-M3) 만으로 구성된 경우 Interpolation motor setting 항목의 **Start Motor와 Counts는 영향을 미치지 않습니다.**

- ML3 Slave가 IO(NET01-M3) 만으로 구성된 경우 ML3 연결 및 해제를 제어하기 위한 Timer를 설정합니다. **Configuration에서 선택한 Timer는 PLC 프로그램 작성시 사용을 피해 주시기 바랍니다.**

- ML3 제어에 사용할 Command M variable과 Response M variable의 범위를 설정합니다. 설정된 M변수들은 반드시 사용을 피해주시기 바랍니다.
- 변수 설정이 끝난 후 Generate 버튼을 누르면 ML3 Setup을 위한 Project가 생성됩니다.

- Project 생성 후 Download 버튼을 누르면 ML3 Setup이 완료됩니다.
- Done 버튼을 누른 후 Terminal에서 Save와 Reset($$$)을 실행합니다.

4. VARIABLES

4.1. M variable Mapping

- PMAC의 ML3-IO는 **48Byte Data Format**을 지원합니다.

- NET01-M3의 Command 및 Response Register들은 PMAC의 **M Variable에 매핑하여 사용**합니다.

- 각 M Variable의 **크기는 16Bit**이고, **Station별 점유하는 M Variable의 개수는 25개** 입니다.

- **Station별 Offset은 25**입니다. 예를 들어 MechatroLink Explorer에서 Command Buffer의 Start Number를 M4000, Response Buffer의 Start Number를 M7000으로 설정했을 때

Station NO	CMD Buffer	RESP Buffer
ST 1	M4000~M4024	M7000~M7024
ST 2	M4025~M4049	M7025~M7049
ST 3	M4050~M4074	M7050~M7074
ST 4	M4075~M4099	M7075~M7099
…	…	…

- 각 Station 별 매핑된 M변수 목록은 MechatroLink Explorer에서 생성한 Project 에서 "MechatroLinkCommands.pmc" 파일을 열어 확인하실 수 있습니다.

- NET01-M3의 48Byte Data Format과 PMAC의 M Variable 매핑은 아래와 같습니다.

종 별	Command	Command M Variable	Response	Response M Variable
Control Register	DATA_RWA(20h)	Base+0 (ex:M4000)	DATA_RWA(20h)	Base+0 (ex:M7000)
	NONE	Base+1 (ex:M4001)	RWDT	Base+1 (ex:M7001)
	CMD_CTRL	Base+2 (ex:M4002)	CMD_STAT	Base+2 (ex:M7002)
	NONE	Base+3 (ex:M4003)	접속 Status	Base+3 (ex:M7003)
Remote I/O	호기번호[0] Remote I/O입력	Base+4 (ex:M4004)	호기번호[0] Remote I/O출력	Base+4 (ex:M7004)
	호기번호[1] Remote I/O입력	Base+5 (ex:M4005)	호기번호[1] Remote I/O출력	Base+5 (ex:M7005)
	호기번호[2] Remote I/O입력	Base+6 (ex:M4006)	호기번호[2] Remote I/O출력	Base+6 (ex:M7006)
	호기번호[3] Remote I/O입력	Base+7 (ex:M4007)	호기번호[3] Remote I/O출력	Base+7 (ex:M7007)
	호기번호[4] Remote I/O입력	Base+8 (ex:M4008)	호기번호[4] Remote I/O출력	Base+8 (ex:M7008)
	호기번호[5] Remote I/O입력	Base+9 (ex:M4009)	호기번호[5] Remote I/O출력	Base+9 (ex:M7009)
	호기번호[6] Remote I/O입력	Base+10 (ex:M4010)	호기번호[6] Remote I/O출력	Base+10 (ex:M7010)
	호기번호[7] Remote I/O입력	Base+11 (ex:M4011)	호기번호[7] Remote I/O출력	Base+11 (ex:M7011)
Remote Register	Register 호기 번호	Base+12 (ex:M4012)	Register 호기 번호 응답	Base+12 (ex:M7012)
	명령 Code + TRIG	Base+13 (ex:M4013)	명령 Code 응답 + TRIG 응답+STATUS	Base+13 (ex:M7013)
	DATA LOW	Base+14 (ex:M4014)	DATA LOW 응답	Base+14 (ex:M7014)
	DATA HIGH	Base+15 (ex:M4015)	DATA HIGH 응답	Base+15 (ex:M7015)
	Register 호기 번호	Base+16 (ex:M4016)	Register 호기 번호 응답	Base+16 (ex:M7016)
	명령 Code + TRIG	Base+17 (ex:M4017)	명령 Code 응답 + TRIG 응답+STATUS	Base+17 (ex:M7017)
	DATA LOW	Base+18 (ex:M4018)	DATA LOW 응답	Base+18 (ex:M7018)
	DATA HIGH	Base+19 (ex:M4019)	DATA HIGH 응답	Base+19 (ex:M7019)
	Register 호기 번호	Base+20 (ex:M4020)	Register 호기 번호 응답	Base+20 (ex:M7020)
	명령 Code + TRIG	Base+21 (ex:M4021)	명령 Code 응답 + TRIG 응답+STATUS	Base+21 (ex:M7021)
	DATA LOW	Base+22 (ex:M4022)	DATA LOW 응답	Base+22 (ex:M7022)
	DATA HIGH	Base+23 (ex:M4023)	DATA HIGH 응답	Base+23 (ex:M7023)

4.2. Control Register

A. DATA_RWA

매핑된 M Variable에 $20을 입력하면 Data Read Write가 시작됩니다.

Ex) M4000 = $20

B. RWDT

통신이 시작되면 $00~0F범위로 Data가 반복적으로 증가하며 통신이 멈추면 Data가 정지된 시점의 값으로 고정됩니다.

Ex) M7001 = $00 ~ $0F

C. CMD_CTRL

Alarm 발생시 매핑된 M Variable에 $08을 입력하면 Alarm을 Clear 합니다.

Ex) M4002 = $08

D. CMD_STAT

현재 CMD 상태를 모니터링 합니다. (Ex : M7002)

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
RCMD_ID		예약	예약	ALM_CLR_CMP	CMDRDY	D_WAR	D_ALM

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
COMM_ALM				CMD_ALM			

• Bit Field 의 설명

Bit Field	정 의	설 명
D_ALM	0 : Alarm 없음 1 : **NETC01-M3**가 Alarm 상태	**NETC01-M3**의 Alarm 상태를 나타내는 Bit입니다. **NETC01-M3** 고유의 Alarm이 발생하면 (COMM_ALM과 CMD_ALM 제외) D_ALM=1이 됩니다. COMM_ALM, CMD_ALM과는 독립되어 있습니다. ALM_CLR Command에 의해 **NETC01-M3**가 Alarm 상태로부터 정상적으로 회복되면 D_ALM=0이 됩니다.
D_WAR	0 : Warning 없음 1 : **NETC01-M3**가 Warning 상태	**NETC01-M3**의 Warning 상태를 나타내는 Bit입니다. **NETC01-M3**고유의 Warning이 발생하면 (COMM_ALM과 CMD_ALM 제외) D_WAR=1이 됩니다. COMM_ALM, CMD_ALM과는 독립되어 있습니다. ALM_CLR Command에 의해 **NETC01-M3**가 Warning 상태로부터 정상적으로 회복되면 D_WAR=0이 됩니다.
CMDRDY	0 : Command 접수 불가능 1 : Command 접수 가능	CMDRDY=0은 **NETC01-M3**가 Command 처리 실행 중임을 나타냅니다. 실행 중인 Command 처리를 계속하지만 DISCONNECT Command만은 CMDRDY의 값에 관계 없이 곧바로 실행됩니다. Command 실행의 완료 확인은 각 Command의 완료 확인 방법과 동일합니다. CMDRDY=0의 유지 시간은 **NETC01-M3**의 사양에 따라 규정되어 있습니다. 사양 값을 초과할 때에는 Master에서 Command Timeout을 검출합니다. Alarm이나 Warning의 상태라도 Command를 실행할 수 있다면 CMDRDY=1이 됩니다.
ALM_CLR_CMP	0 : Alarm Clear 미처리 1 : Alarm Clear 처리 완료	ALM_CLR_CMP=1은 CMD_CTRL.ALM_CLR=1을 받아 Alarm Clear처리가 완료된 것을 나타냅니다. CMD_CTRL.ALM_CLR을 「0」으로 하면 ALM_CLR_CMP를 해제할 수 있습니다.
RCMD_ID	Command Format의 CMD_ID의 Echo Back	Command의 변화에 대응한 Slave의 응답 Data를 나타냅니다. Command Format의 CMD_ID를 Echo Back합니다.
CMD_ALM	Command 이상 상태	COMM_ALM, D_ALM 및 D_WAR과는 독립되어 있습니다. Command에 이상이 발생한 경우 정상적인 Command를 접수하면 CMD_ALM은 자동으로 해제됩니다. CMD_ALM의 Alarm/Warning의 분류는 **NETC01-M3**의 사양에 의해 규정되어 있습니다. 자세한 내용은 30Page를 참조하십시오.
COMM_ALM	MECHATROLINK-Ⅲ 통신 이상 상태	CMD_ALM, D_ALM 및 D_WAR과는 독립되어 있습니다. COMM_ALM은 CMD_CTRL.ALM_CLR의 상승 Edge 또는 ALM_CLR Command로 Clear됩니다. 자세한 내용은 30Page를 참조하십시오.

E. 접속 Status

각 Station(NETC01-03)에 연결된 모터의 접속상태를 모니터링 합니다.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
7 호기	6 호기	5 호기	4 호기	3 호기	2 호기	1 호기	0 호기
15 호기	14 호기	13 호기	12 호기	11 호기	10 호기	9 호기	8 호기

접속이된 호기의 해당 Bit가 1로 표시되며, 접속되지 않은 호기는 0으로 표시됩니다.

Ex) 0호기~3호기 접속 : M7003 = $07

4.3. Remote I/O

- Remote I/O 입력을 통해 NETC01-03에 연결된 모터를 구동시키고 Remote I/O 출력을 통해 구동상태를 모니터링 합니다.

- 16bit mode로 1개의 Station당 최대 8개 호기(모터)에 대해 구동명령을 내릴 수 있습니다.

- 구동속도와 가속도, 목표 위치등은 Remote Register를 통해 입력한 Data 0~63로 정해집니다.

- Remote I/O M Variable은 Base Add +4 ~ Base Add +11 입니다.
  예를 들어 Command Buffer Start NO가 M4000, Response Buffer Start NO가 M7000일 경우 Remote I/O 입력 주소의 범위는 M4004(0호기) ~ M4011(7호기), Remote I/O 출력 주소의 범위는 M7004(0호기)~M7011(7호기) 입니다.

- **Remote I/O 입력** 16Bit의 기본구성은 아래와 같습니다. (필요에 따라 기능변경 가능)

• 8 축 접속 Mode 인 경우(16 bit Mode)

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
NET-IN15 [RVS]	NET-IN14 [FWD]	NET-IN13 [-JOG]	NET-IN12 [+JOG]	NET-IN11 [SSTART]	NET-IN10 [MS2]	NET-IN9 [MS1]	NET-IN8 [MS0]
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
NET-IN7 [미사용]	NET-IN6 [FREE]	NET-IN5 [STOP]	NET-IN4 [HOME]	NET-IN3 [START]	NET-IN2 [M2]	NET-IN1 [M1]	NET-IN0 [M0]

* [ ] 안은 초기값입니다.

- **Remote I/O 출력** 16Bit의 기본구성은 아래와 같습니다. (필요에 따라 기능변경 가능)

• 8 축 접속 Mode 인 경우(16 bit Mode)

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
NET-OUT15 [TLC]	NET-OUT14 [END]	NET-OUT13 [MOVE]	NET-OUT12 [TIM]	NET-OUT11 [AREA3]	NET-OUT10 [AREA2]	NET-OUT9 [AREA1]	NET-OUT8 [S-BSY]
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
NET-OUT7 [ALM]	NET-OUT6 [WNG]	NET-OUT5 [READY]	NET-OUT4 [HOME-P]	NET-OUT3 [START_R]	NET-OUT2 [M2_R]	NET-OUT1 [M1_R]	NET-OUT0 [M0_R]

* [ ] 안은 초기값입니다.

A. Remote I/O 입력 상세

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
NET-IN15 [RVS]	NET-IN14 [FWD]	NET-IN13 [-JOG]	NET-IN12 [+JOG]	NET-IN11 [SSTART]	NET-IN10 [MS2]	NET-IN9 [MS1]	NET-IN8 [MS0]
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
NET-IN7 [미사용]	NET-IN6 [FREE]	NET-IN5 [STOP]	NET-IN4 [HOME]	NET-IN3 [START]	NET-IN2 [M2]	NET-IN1 [M1]	NET-IN0 [M0]

* [ ] 안은 초기값입니다.

신호명	기 능	설정 범위
미사용	입력 단자를 사용하지 않을 때 설정합니다.	-
FWD	+ 방향 연속 운전을 실행합니다.	0 : 감속 정지 1 : 운전
RVS	- 방향 연속 운전을 실행합니다.	
HOME	원점 복귀 운전을 실행합니다.	0 : 동작 없음 1 : 동작 개시
START	위치 결정 운전을 실행합니다.	
SSTART	순차 위치 결정 운전을 실행합니다.	
+JOG	+ 방향 JOG 운전을 실행합니다.	
-JOG	- 방향 JOG 운전을 실행합니다.	
MS0~MS5	I/O Parameter 로 설정한 운전 Data No.의 Direct 위치 결정 운전을 실행합니다.	
FREE	Motor 를 무여자로 하여 전자 Brake 를 해제합니다.	0 : 동작 없음 1 : Motor 무여자, 전자 Brake 해제
C-ON	Motor 의 여자/무여자를 전환합니다.	0 : Motor 무여자 1 : Motor 여자
STOP	Motor 를 정지시킵니다.	0 : 동작 없음 1 : Motor 정지
ALM-RST*	현재 Alarm 을 Reset 합니다.	0 : 동작 없음 1 : Alarm 해제
P-PRESET*	위치 Preset 을 실행합니다.	0 : 동작 없음 1 : Preset 실행
P-CLR*	절대 위치 이상 Alarm 을 Reset 합니다.	0 : 동작 없음 1 : Alarm 해제
HMI	**OPX-2A** 나 **MEXE02** 의 기능 제한을 해제합니다.	0 : 기능 제한 1 : 기능 제한 해제
R0~R15	범용 신호 RS-485 통신으로 제어할 때 사용합니다.	0 : OFF 1 : ON
M0~M5	6 개의 Bit ON/OFF 를 조합하여 운전 Data No.를 선택합니다. 조합에 대한 자세한 내용은 2-23 Page 를 참조하십시오.	0 : OFF 1 : ON (운전 Data No.는 0~63 까지 선택 가능)

B. Remote I/O 출력 상세

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
NET-OUT15 [TLC]	NET-OUT14 [END]	NET-OUT13 [MOVE]	NET-OUT12 [TIM]	NET-OUT11 [AREA3]	NET-OUT10 [AREA2]	NET-OUT9 [AREA1]	NET-OUT8 [S-BSY]
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
NET-OUT7 [ALM]	NET-OUT6 [WNG]	NET-OUT5 [READY]	NET-OUT4 [HOME-P]	NET-OUT3 [START_R]	NET-OUT2 [M2_R]	NET-OUT1 [M1_R]	NET-OUT0 [M0_R]

* [ ] 안은 초기값입니다.

신호명	기 능	읽기 내용
미사용	출력 단자를 사용하지 않을 때 설정합니다.	-
FWD_R	FWD 입력에 대한 응답을 출력합니다.	0 : OFF 1 : ON
RVS_R	RVS 입력에 대한 응답을 출력합니다.	
HOME_R	HOME 입력에 대한 응답을 출력합니다.	
START_R	START 입력에 대한 응답을 출력합니다.	
SSTART_R	SSTART 입력에 대한 응답을 출력합니다.	
+JOG_R	+JOG 입력에 대한 응답을 출력합니다.	
-JOG_R	-JOG 입력에 대한 응답을 출력합니다.	
MS0_R~MS5_R	MS0~MS5 입력에 대한 응답을 출력합니다.	
FREE_R	FREE 입력에 대한 응답을 출력합니다.	
C-ON_R	C-ON 입력에 대한 응답을 출력합니다.	
STOP_R	STOP 입력에 대한 응답을 출력합니다.	
R0~R15	범용 신호 R0~R15 의 상태를 출력합니다.	
M0_R~M5_R	M0~M5 입력에 대한 응답을 출력합니다.	
+LS_R	+LS 입력에 대한 응답을 출력합니다.	
-LS_R	-LS 입력에 대한 응답을 출력합니다.	
HOMES_R	HOMES 입력에 대한 응답을 출력합니다.	
SLIT_R	SLIT 입력에 대한 응답을 출력합니다.	
ALM	Alarm 발생 시에 출력됩니다. (A 접점 : Normally open)	0 : Alarm 없음 1 : Alarm 발생 중
WNG	Warning 발생 시에 출력됩니다.	0 : Warning 없음 1 : Warning 발생 중
READY	Driver 의 운전 준비가 완료되면 출력됩니다.	0 : 운전 불가능 1 : 운전 준비 완료
MOVE	Motor 운전 중에 출력됩니다.	0 : Motor 정지 1 : Motor 동작 중
END	위치 결정 운전이 완료되면 출력됩니다.	0 : Motor 동작 중 1 : Motor 동작 완료
HOME-P	Motor 위치가 원점에 있을 때 출력됩니다.	0 : 원점 이외 1 : 원점
TLC	부하가 Motor 의 Torque 사양 범위 외일 때 출력됩니다.	0 : Torque 범위 내 1 : Torque 범위 외
TIM	Motor 출력축이 7.2° 회전할 때마다 출력됩니다.	0 : OFF 1 : ON
AREA1~AREA3	Motor 가 Area 의 범위 내에 있을 때 출력됩니다.	0 : Area 범위 외 1 : Area 범위 내
S-BSY	Driver 가 내부 처리 상태일 때 출력됩니다.	0 : OFF 1 : ON

C. Remote I/O 사용법

Ex) 정방향 연속운전

M4004 = $4000 : **0번 Data**에 저장된 속도, 가속도로 **0호기** 정방향 연속운전
M4004 = $4001 : **1번 Data**에 저장된 속도, 가속도로 **0호기** 정방향 연속운전
M4005 = $4000 : **0번 Data**에 저장된 속도, 가속도로 **1호기** 정방향 연속운전
M4005 = $4001 : **1번 Data**에 저장된 속도, 가속도로 **1호기** 정방향 연속운전

Ex) 역방향 연속운전

M4004 = $8000 : **0번 Data**에 저장된 속도, 가속도로 **0호기** 역방향 연속운전
M4004 = $8001 : **1번 Data**에 저장된 속도, 가속도로 **0호기** 역방향 연속운전
M4005 = $8000 : **0번 Data**에 저장된 속도, 가속도로 **1호기** 역방향 연속운전
M4005 = $8001 : **1번 Data**에 저장된 속도, 가속도로 **1호기** 역방향 연속운전

Ex) Jog 운전(단위운전)

- 아래 시퀀스에 맞춰 실행하여야 합니다.

```
M4004 = 0                          // Ready 출력이 ON 될 때까지 기다림
while(M7004 & $20 = 0) endwhile

M4004 = $1000                      // Jog + 이동 시작
while(M7004 & $20 != 0)endwhile

M4004 = 0                          // 이동이 완료 될 때까지 기다림
while(M7004 & $20 = 0) endwhile
```

Ex) 위치결정 운전

- 아래 시퀀스에 맞춰 실행하여야 합니다.

```
M4004 = 0                          // Ready 출력이 ON 될 때까지 기다림
while(M7004 & $20 = 0) endwhile

M4004 = $0008                      // 위치결정 운전 시작
while(M7004 & $20 != 0)endwhile

M4004 = 0                          // 이동이 완료 될 때까지 기다림
while(M7004 & $20 = 0) endwhile
```

4.4. Remote Register

- Remote Register 를 사용하여 모터 구동에 필요한 속도, 가속도, 위치정보등을 설정합니다.

- Remote Register 로 설정한 Data 는 Remote I/O 를 통해 모터 구동시 반영됩니다.

- 16Bit Data Mode 에서 최대 3 개 호기에 동시에 Data 전송이 가능하며, 3 개를 초과하는 호기에 대해 Data 를 전송하기 위해서는 순차적으로 명령이 인가되어야 합니다.

- Remote Register 입력시 Write Bit 와 Trigger Bit 가 반드시 인가 되어야 합니다.

- **Remote Register 를 전송하고 반드시 0 으로 Clear 한 후 다음 Data 를 전송해야 합니다.**

- Remote Register 출력을 통해 Motor 의 현재 위치, 속도등을 모니터링 하실 수 있습니다.

■ Remote Register 입력

• Command[**NETC01-M2**(**NETC01-M3**)→Driver]

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
–	TRIG	명령 Code					
DATA							

• 설 명

명 칭	내 용	설정 범위
명령 Code	Parameter 의 읽기와 쓰기, Monitoring 및 Maintenance 명령 Code 를 지정합니다.	–
TRIG	명령 Code 를 실행하는 Handshake 용 Trigger 입니다. TRIG 가 0 에서 1 이 되면 명령 Code 와 DATA 가 실행됩니다.	0 : 동작 없음 1 : 실행
DATA	Driver 에 기록하는 Data 입니다(Little-endian).	–

■ Remote Register 출력

• Response[Driver→**NETC01-M2**(**NETC01-M3**)]

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
STATUS	TRIG_R	명령 Code					
DATA_R							

• 설 명

명 칭	내 용	설정 범위
명령 Code	Command 의 명령 Code 를 반환합니다.	–
TRIG_R	명령 Code 실행 완료를 나타내는 Handshake 용 Trigger 입니다. 명령 Code 실행이 완료되면 TRIG_R 이 0 에서 1 로 바뀝니다.	0 : 미처리 1 : 실행 완료
STATUS	명령 Code 실행 결과를 나타냅니다.	0 : 정상 1 : 이상
DATA_R	Driver 에서 읽어온 Data 입니다(Little-endian).	–

A. 운전 Data

- 운전 Data 는 최대 64 개 까지 지정 가능하며, Remote I/O 출력시 구동에 사용할 Data 선택이 가능합니다.

- 운전 Data 는 정/역방향 연속이동, 위치결정 운전등에 사용됩니다.

명령 Code		내 용	설정 범위	초기값
읽기	쓰기			
0200h ~ 023Fh	1200h ~ 123Fh	위치 No.0 ~ 위치 No.63	-8,388,608~8,388,607 step	0
0240h ~ 027Fh	1240h ~ 127Fh	운전 속도 No.0 ~ 운전 속도 No.63	0~1,000,000 Hz	1000
0280h ~ 02BFh	1280h ~ 12BFh	운전 방식 No.0 ~ 운전 방식 No.63	0 : Incremental 1 : Absolute	0
02C0h ~ 02FFh	12C0h ~ 12FFh	운전 기능 No.0 ~ 운전 기능 No.63	0 : 단독 1 : 연결 2 : 연결 2 3 : 누름	0
0300h ~ 033Fh	1300h ~ 133Fh	가속 No.0 ~ 가속 No.63	1~1,000,000(1=0.001 ms/kHz 또는 1=0.001 s)*1*2	1000
0340h ~ 037Fh	1340h ~ 137Fh	감속 No.0 ~ 감속 No.63		
0380h ~ 03BFh	1380h ~ 13BFh	누름 전류 No.0 ~ 누름 전류 No.63	0~1000 (1=0.1 %) *3	200
03C0h ~ 03FFh	13C0h ~ 13FFh	순차 위치 결정 No.0 ~ 순차 위치 결정 No.63	0 : 무효 1 : 유효	0
0400h ~ 043Fh	1400h ~ 143Fh	Dwell 시간 No.0 ~ Dwell 시간 No.63	0~50000(1=0.001 s)	0

Ex) 호기(모터)0에 대한 Data0 운전속도, Data1 운전속도 전송

```
////////////////////////////////////////////////////
// 호기(모터) 선택 및 CMD Code 초기화
////////////////////////////////////////////////////
M4012 = 0
M4013 = 0
while(M7013 & $4000 != 0)   endwhile

////////////////////////////////////////////////////
// Data 0 속도($240) 입력
////////////////////////////////////////////////////
M4014 = 1000 & $FFFF
M4015 = (1000&$FFFF0000)/$10000
M4013 = $240 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile

////////////////////////////////////////////////////
// Data 1 속도($241) 입력
////////////////////////////////////////////////////
M4014 = 2000 & $FFFF
M4015 = (2000&$FFFF0000)/$10000
M4013 = $241 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile
```

Ex) 호기(모터)0에 대한 Data0 운전방식, 위치 전송

```
////////////////////////////////////////////////////
// 호기(모터) 선택 및 CMD Code 초기화
////////////////////////////////////////////////////
M4012 = 0
M4013 = 0
while(M7013 & $4000 != 0)   endwhile

////////////////////////////////////////////////////
// Data 0 운전방식($280) 입력
////////////////////////////////////////////////////
M4014 = 1
M4015 = 0
M4013 = $280 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile

////////////////////////////////////////////////////
// Data 0 위치($200) 입력
////////////////////////////////////////////////////
M4014 = -30000 & $FFFF
M4015 = (-30000&$FFFF0000)/$10000
M4013 = $200 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile
```

B. 운전 Parameter

- 운전 Parameter 는 Jog 이동에 관련된 항목과 기동속도(구동 시작속도)등을 설정합니다.

명령 Code		내 용	설정 범위	초기값	반영*1
읽기	쓰기				
0140h	1140h	공통 가속	1~1,000,000(1=0.001 ms/kHz 또는 1=0.001 s)*2	1000	B
0141h	1141h	공통 감속			
0142h	1142h	기동 속도	0~1,000,000 Hz	500	
0143h	1143h	JOG 운전 속도	1~1,000,000 Hz	1000	
0144h	1144h	JOG 가감속	1~1,000,000(1=0.001 ms/kHz 또는 1=0.001 s)*2	1000	
0145h	1145h	JOG 기동 속도	0~1,000,000 Hz	500	
0146h	1146h	가감속 선택	0 : 공통 1 : 독립	1	
0147h	1147h	가감속 단위	0 : ms/kHz 1 : s	0	C
0820h	1820h	자동 복귀 동작	0 : 무효 1 : 유효	0	
0821h	1821h	자동 복귀 운전 속도	1~1,000,000 Hz	1000	B
0822h	1822h	자동 복귀 가감속	1~1,000,000(1=0.001 ms/kHz 또는 1=0.001 s)*2	1000	
0823h	1823h	자동 복귀 기동 속도	0~1,000,000 Hz	500	
0824h	1824h	JOG 이동량	1~8,388,607 step	1	

Ex) 호기(모터)0에 대한 Jog 이동량 설정

```
//////////////////////////////////////////////////
// 호기(모터) 선택 및 CMD Code 초기화
//////////////////////////////////////////////////
M4012 = 0
M4013 = 0
while(M7013 & $4000 != 0)   endwhile
//////////////////////////////////////////////////
// Jog 이동량($824) 입력
//////////////////////////////////////////////////
M4014 = 500 & $FFFF
M4015 = (500&$FFFF0000)/$10000
M4013 = $824 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile
```

C. 원점 복귀 Parameter

- 원점 복귀에 관련된 Parameter 를 설정합니다.

- Index(Encoder C 상) 방식 원점 검색은 지원하지 않습니다.

명령 Code		내 용	설정 범위	초기값	반영*1
읽기	쓰기				
0160h	1160h	원점 복귀 방법	0 : 2 Sensor 방식 1 : 3 Sensor 방식 2 : 누름 방식	1	B
0161h	1161h	원점 복귀 운전 속도	1~1,000,000 Hz	1000	
0162h	1162h	원점 복귀 가감속	1~1,000,000(1=0.001 ms/kHz 또는 1=0.001 s)*2	1000	
0163h	1163h	원점 복귀 기동 속도	1~1,000,000 Hz	500	
0164h	1164h	원점 복귀 Offset	-8,388,608~8,388,607 step	0	
0165h	1165h	원점 복귀 개시 방향	0 : -측 1 : +측	1	
0166h	1166h	원점 복귀 SLIT Sensor 검출	0 : 무효 1 : 유효	0	
0167h	1167h	원점 복귀 TIM 신호 검출			
0168h	1168h	누름 운전 원점 복귀 운전 전류	0~1000 (1=0.1 %)	1000	

Ex) 호기(모터)0에 대한 원점 검색 방향 설정

```
//////////////////////////////////////////////////
// 호기(모터) 선택 및 CMD Code 초기화
//////////////////////////////////////////////////
M4012 = 0
M4013 = 0
while(M7013 & $4000 != 0)   endwhile
//////////////////////////////////////////////////
// 원점 검색방향 설정($165) 입력
//////////////////////////////////////////////////
M4014 = 1
M4015 = 0
M4013 = $165 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile

M4013 = 0
while(M7013 & $4000 != 0)   endwhile
```

D. Monitoring Command

- Monitoring Command 를 사용하여 Remote Register 출력으로 특정 정보를 Monitoring 할 수 있습니다.

명령 Code	내 용	설 명
2040h	현재의 Alarm	발생 중인 Alarm Code 를 나타냅니다.
2041h	Alarm 이력 1	Alarm 이력 1~10 을 나타냅니다.
2042h	Alarm 이력 2	
2043h	Alarm 이력 3	
2044h	Alarm 이력 4	
2045h	Alarm 이력 5	
2046h	Alarm 이력 6	
2047h	Alarm 이력 7	
2048h	Alarm 이력 8	
2049h	Alarm 이력 9	
204Ah	Alarm 이력 10	
204Bh	현재의 Warning	발생 중인 Warning Code 를 나타냅니다.
204Ch	Warning 이력 1	Warning 이력 1~10 을 나타냅니다.
204Dh	Warning 이력 2	
204Eh	Warning 이력 3	
204Fh	Warning 이력 4	
2050h	Warning 이력 5	
2051h	Warning 이력 6	
2052h	Warning 이력 7	
2053h	Warning 이력 8	
2054h	Warning 이력 9	
2055h	Warning 이력 10	
2057h	통신 Error Code 이력 1	지금까지 발생한 통신 Error Code 이력 1~10 을 나타냅니다.
2058h	통신 Error Code 이력 2	
2059h	통신 Error Code 이력 3	
205Ah	통신 Error Code 이력 4	
205Bh	통신 Error Code 이력 5	
205Ch	통신 Error Code 이력 6	
205Dh	통신 Error Code 이력 7	
205Eh	통신 Error Code 이력 8	
205Fh	통신 Error Code 이력 9	
2060h	통신 Error Code 이력 10	
2061h	현재의 선택 Data No.	선택된 운전 Data No.를 나타냅니다.
2062h	현재의 운전 Data No.	위치 결정 운전 중의 운전 Data No.를 나타냅니다. 연결 운전과 순차 위치 결정 운전에서 사용할 수 있습니다. 정지 중에는 마지막으로 운전한 Data No.가 표시됩니다.
2063h	지령 위치	지령 위치를 나타냅니다.
2064h	지령 속도	지령 속도를 나타냅니다.
2066h	Feedback 위치	Feedback 위치를 나타냅니다.
2067h	Feedback 속도	Feedback 속도를 나타냅니다.
2069h	Dwell 의 남은 시간	연결 운전 2 에서 사용하는 Dwell 시간의 잔여량을 나타냅니다.
206Ah	Direct I/O, 전자 Brake 상태	Direct I/O 와 전자 Brake 의 상태를 나타냅니다. 할당은 다음 표를 참조하십시오.

Ex) 호기(모터)0, 호기(모터)1에 대한 현재 위치 모니터링

```
//////////////////////////////////////////////////
// 0호기(모터) 선택 및 CMD Code 초기화
//////////////////////////////////////////////////
M4012 = 0
M4013 = 0
while(M7013 & $4000 != 0)   endwhile

//////////////////////////////////////////////////
// 호기0의 현재위치($2066)를 P1000에 저장
//////////////////////////////////////////////////
M4013 = $2066 + $4000 + 1000
while(M7013 & $4000 = 0)    endwhile
p1000 = M7015*$10000 + M7014

//////////////////////////////////////////////////
// 1호기(모터) 선택 및 CMD Code 초기화
//////////////////////////////////////////////////
M4016 = 1
M4017 = 0
while(M7016 & $4000 != 0)   endwhile

//////////////////////////////////////////////////
// 호기1의 현재위치($2066)를 P1001에 저장
//////////////////////////////////////////////////
M4017 = $2066 + $4000 + 1000
while(M7017 & $4000 = 0)    endwhile
p1001 = M7019*$10000 + M7018
```