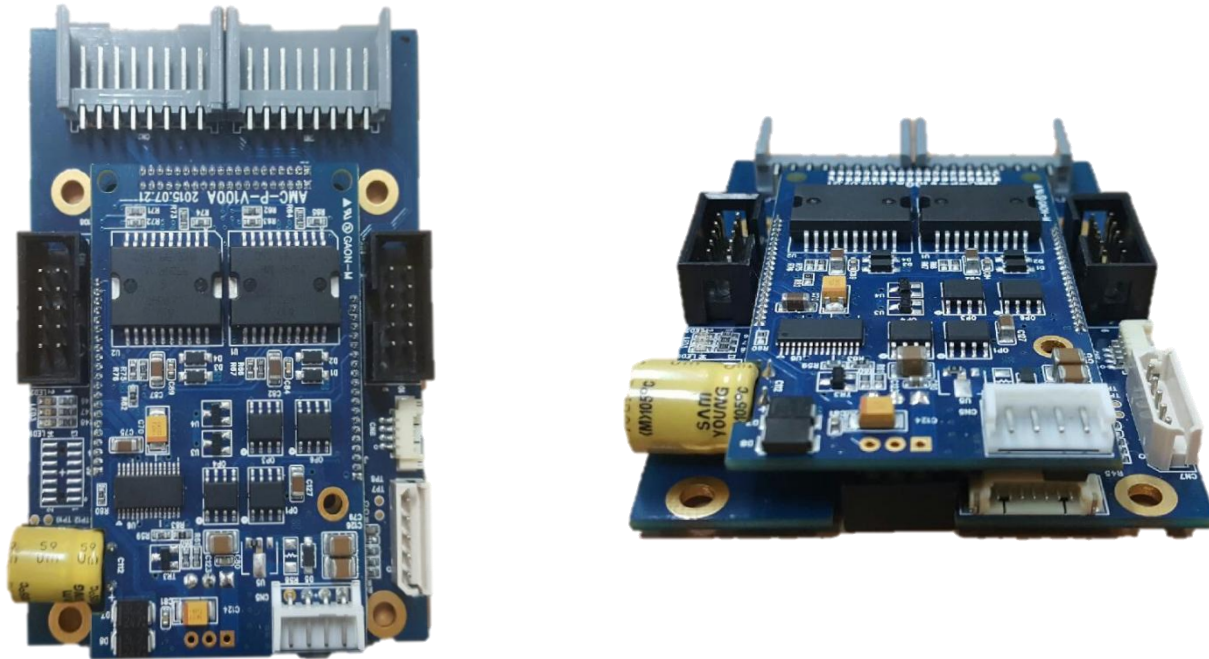


MBBL-2ACD

2축 BLDC 모터 드라이버 컨트롤러 사용 설명서



Motion Bank

1. 목 차



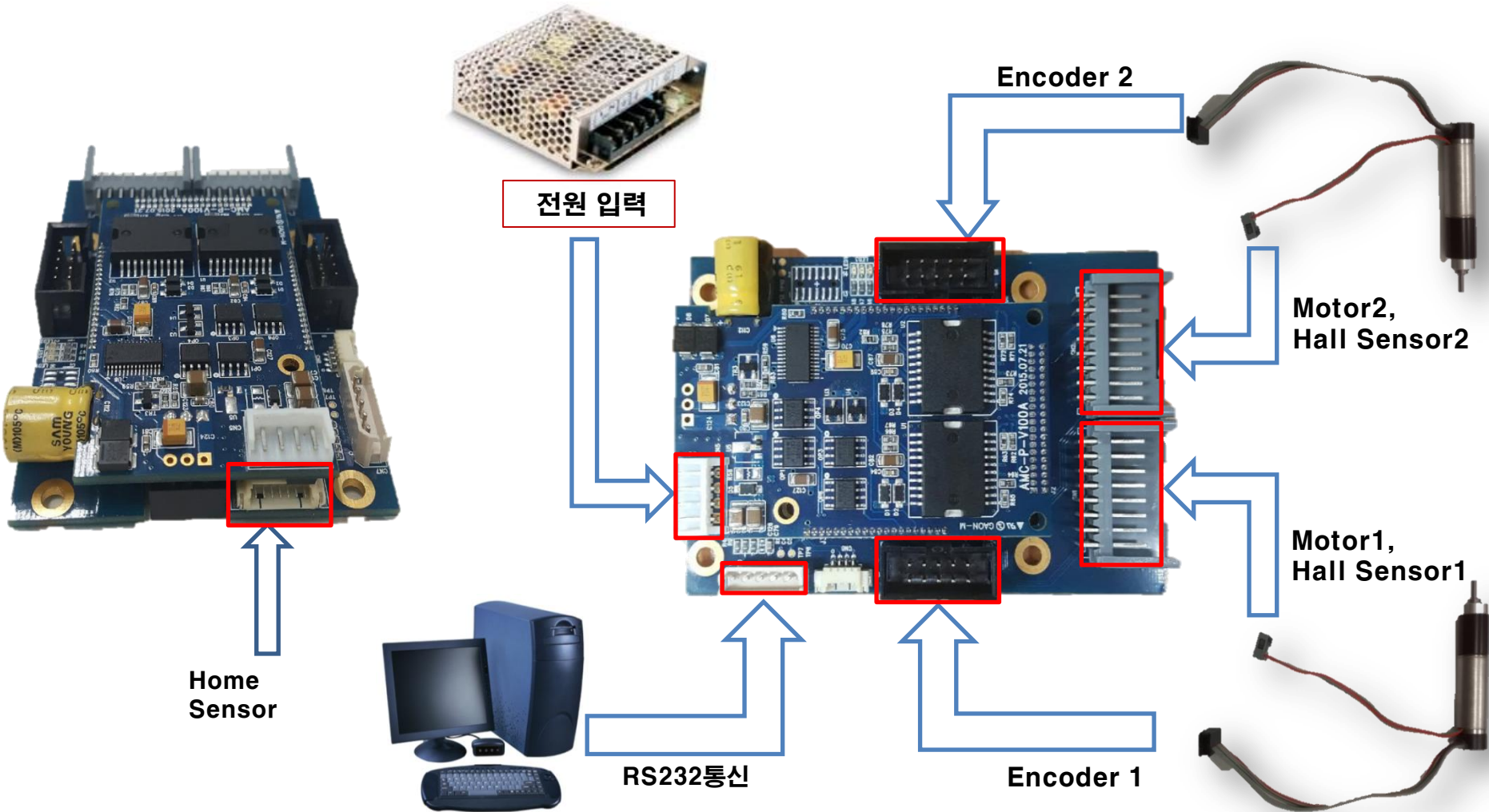
1. 목 차	1
2. 제품의 특징	3
3. 시스템의 구성	4
4. 제품의 구성	5
5. 제품의 배선	
- 5.1 전원 배선	7
- 5.2 통신 배선(RS232)	8
- 5.3 Motor /Hall sensor	9
- 5.4 Encoder 배선	10
- 5.5 Home Sensor 배선	11
6. 통신 프로토콜	
- 6.1 통신 개요	12
- 6.2 Port 확인 방법	13
- 6.3 Port 변경 방법	14
- 6.4 RS232 통신(하이퍼 터미널 세팅)	15
7. 명령 프로토콜 개요	16

8. 모터 구동 파라미터 설정	
- 8.1 S1명령어로 엔코더 타입 설정하기 -----	17
- 8.2 S1명령어로 엔코더 펄스 설정하기 -----	18
- 8.3 S1명령어로 최대 구동 전류 설정하기 -----	19
- 8.4 S1명령어로 최대 허용 전류 설정하기 -----	20
- 8.5 S2명령어를 이용한 위치 루프 게인 설정 -----	21
- 8.6 S2명령어를 이용한 속도 루프 게인 설정 -----	22
- 8.7 S2명령어를 이용한 속도 루프 시정수 설정 -----	23
- 8.8 S2명령어를 이용한 속도 LIMIT 설정 -----	24
- 8.9 S2명령어를 이용한 토크 LIMIT 설정 -----	25
9. 명령 프로토콜 설명 -----	26
10. 명령 프로토콜 예시	
- 10.1 Ss명령어를 사용한 위치 제어 예시 -----	36
- 10.2 ST명령어를 사용한 위치 제어 예시 -----	37
- 10.3 SS명령어를 사용한 위치 제어 예시 -----	38
- 10.4 SC명령어를 사용한 위치 제어 예시 -----	39
- 10.5 Sr명령어를 사용한 위치 제어 예시 -----	40
- 10.6 Sa/SV명령어를 사용한 속도 제어 예시 -----	41
- 10.7 SV명령어를 이용한 가/감속-시간 그래프 -----	42
- 10.8 ED명령어를 사용한 작동 중 정지 -----	43
- 10.9 홀 관련 통신 예시 -----	44
11. Fault 처리 -----	45

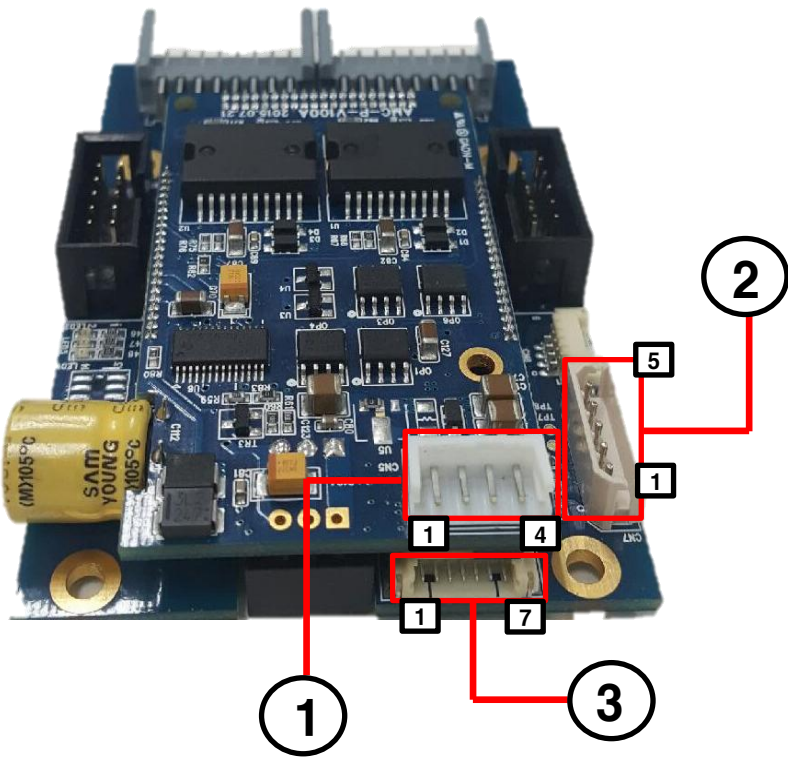
2.제품의 특징

- ❖ 8~24V 넓은 전원 입력 범위
- ❖ 실시간 위치/속도 제어기능
- ❖ 초소형 사이즈(60mm X 87mm X 15mm)
- ❖ 2축 BLDC모터 동시/독립제어가능
- ❖ Hall Sensor와 Encoder Sensor를 이용하여 BLDC모터 구동이 가능
- ❖ RS232 통신 방식 지원
- ❖ 파라미터 확인, 변경, 저장(EEPROM) 기능
- ❖ 검증된 모터 구동 전용 칩 사용(L6234)

3.시스템의 구성

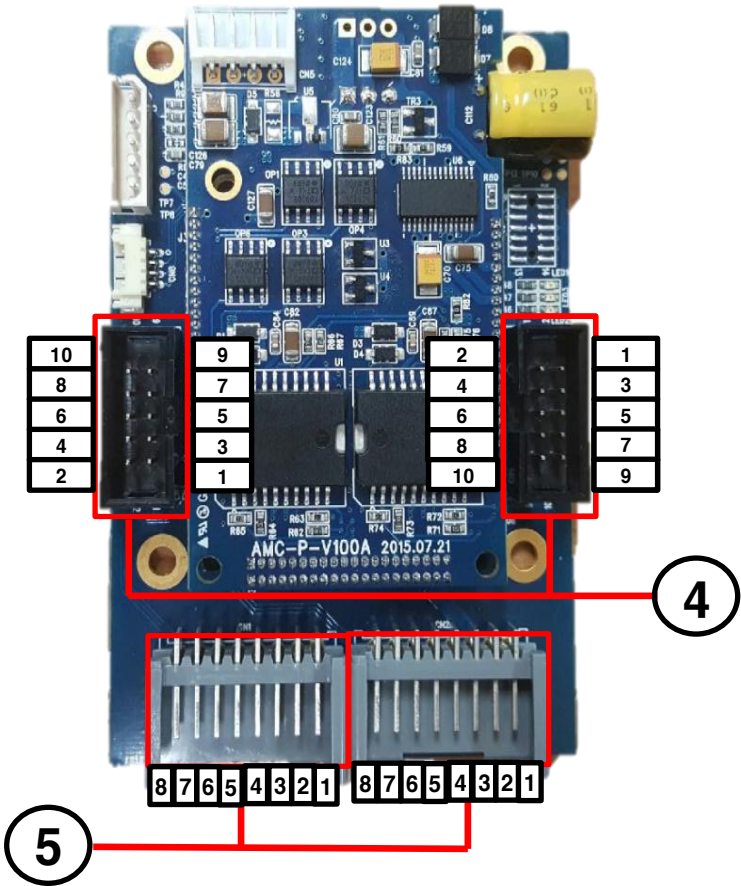


4.제품의 구성



분류	이름	설명	
1	Power Connector	1	GND
		2	DC 8~24V 입력
		3	DC 8~24V 입력
		4	GND
2	RS232 Connector	1	BOOT
		2	TXD
		3	RXD
		4	GND
		5	GND
3	Home Sensor Connector	1	VCC (3.3V)
		2	Hall Sensor1 IN
		3	Hall Sensor2 IN
		4	GND
		5	NC
		6	NC
		7	GND

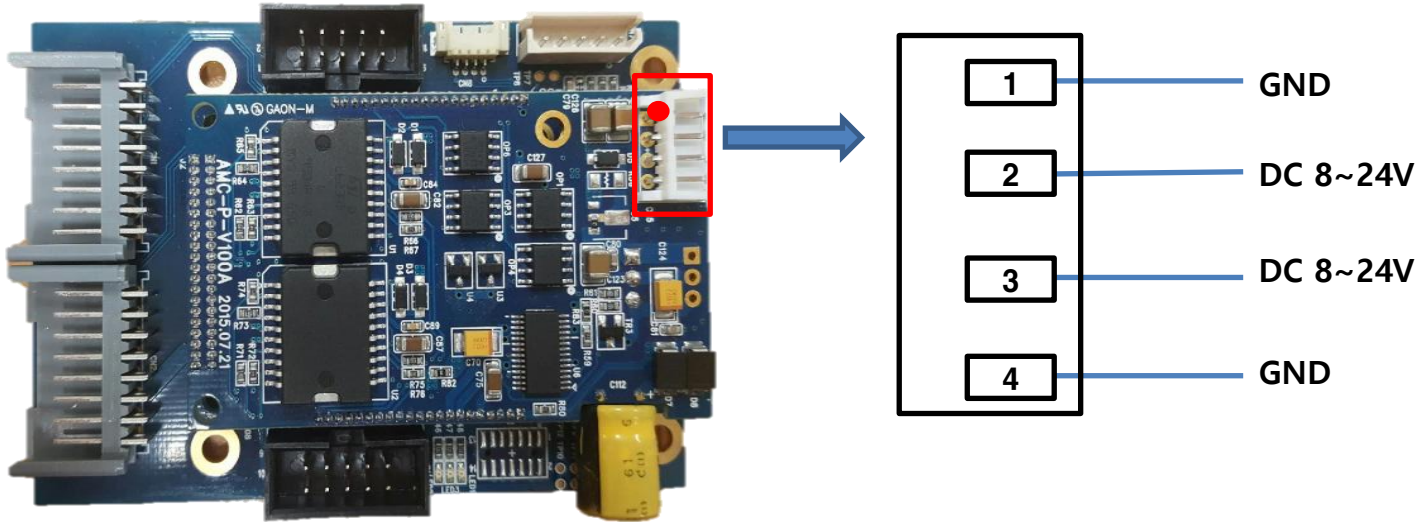
4.제품의 구성



분류	이름	설명	
4	Encoder Connector	1	NC
		2	VCC (5V)
		3	GND
		4	NC
		5	Encoder /A상
		6	Encoder A상
		7	Encoder /B상
		8	Encoder B상
		9	Encoder /I상
		10	Encoder I상
5	Motor/Hall Connector	1	Motor U상 출력
		2	Motor V상 출력
		3	Motor W상 출력
		4	VCC (5V)
		5	GND
		6	Hall Sensor 1상 입력
		7	Hall Sensor 2상 입력
		8	Hall Sensor 3상 입력

5.제품의 배선

5.1 전원 배선

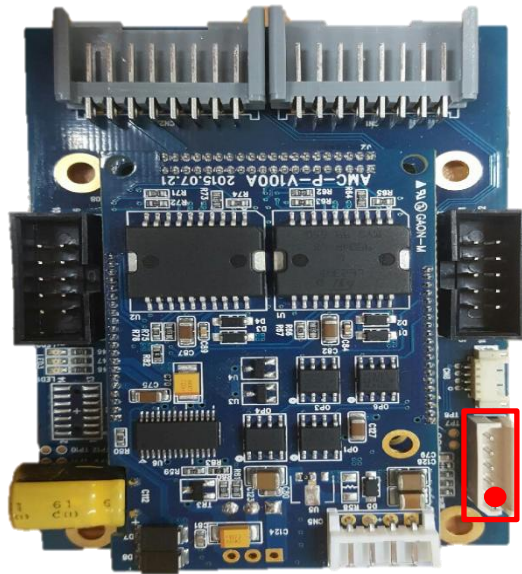


● 1번 핀을 의미합니다.

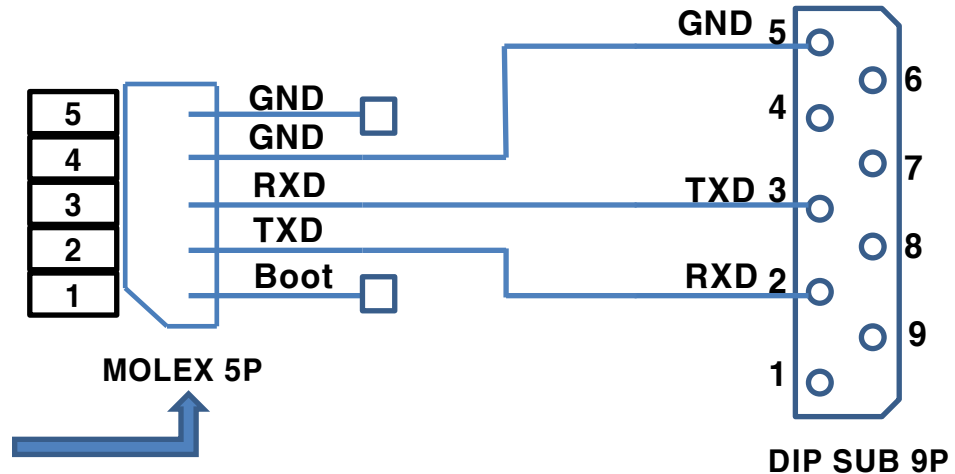
- 전원의 잘못된 결선은 제어기와 모터 파손의 원인이 되오니 배선을 확인 하신 후 결선하여 주시기 바랍니다.

5.제품의 배선

5.2 통신 배선(RS232)



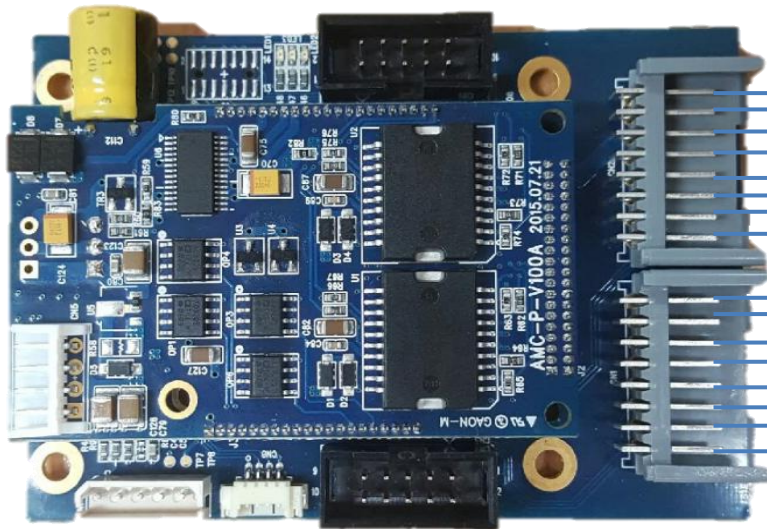
● 1번 핀을 의미합니다.



- 5pin RS232 커넥터의 1번 핀(Boot)은 Firmware Download를 위해 할당 된 핀으로서, 일반적인 통신시 1번 핀(Boot)은 연결하지 않아도 됩니다.

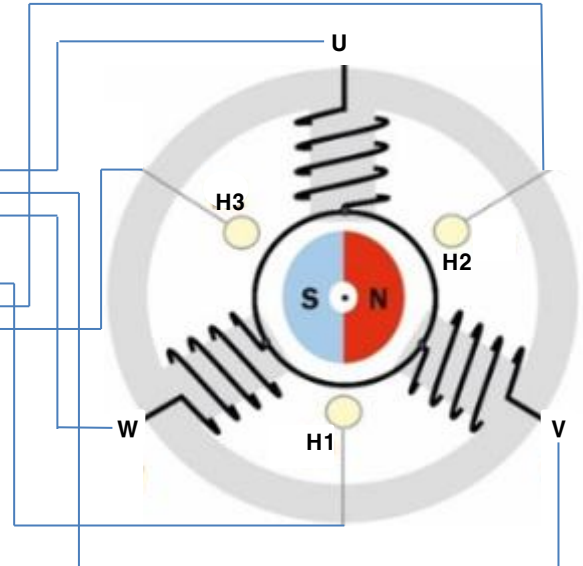
5.제품의 배선

5.3 Motor/Hall Sensor 배선



MOTOR U상
MOTOR V상
MOTOR W상
VCC (5V)
GND
Hall Sensor 1상
Hall Sensor 2상
Hall Sensor 3상

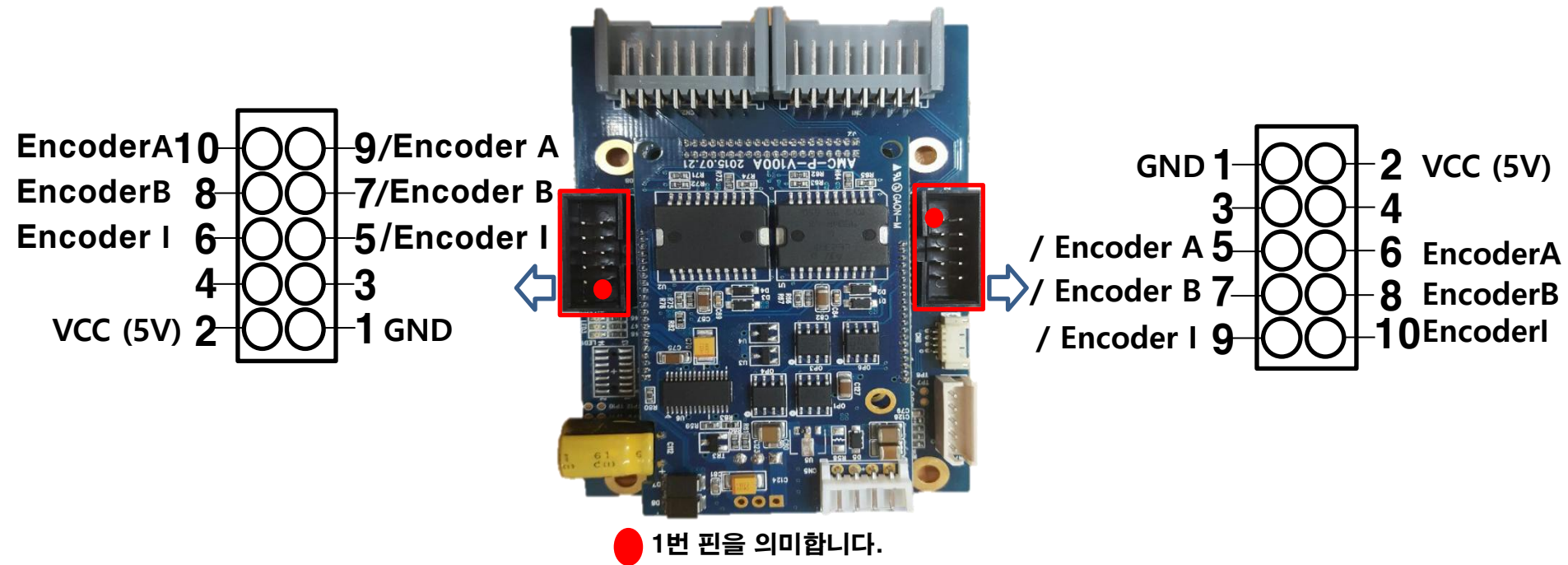
MOTOR U상
MOTOR V상
MOTOR W상
VCC (5V)
GND
Hall Sensor 1상
Hall Sensor 2상
Hall Sensor 3상



- 사용하시는 BLDC 모터 U, V, W상 3상을 정확히 확인 후 결선해 주십시오.
잘못된 결선은 제어기와 모터의 파손 원인이 됩니다.
- Hall Sensor의 배선을 확인 하신 후 정확히 결선해 주십시오.
잘못된 결선은 제어기와 모터의 파손 원인이 됩니다.

5.제품의 배선

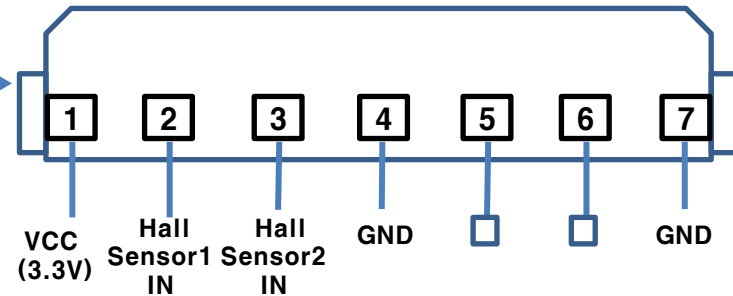
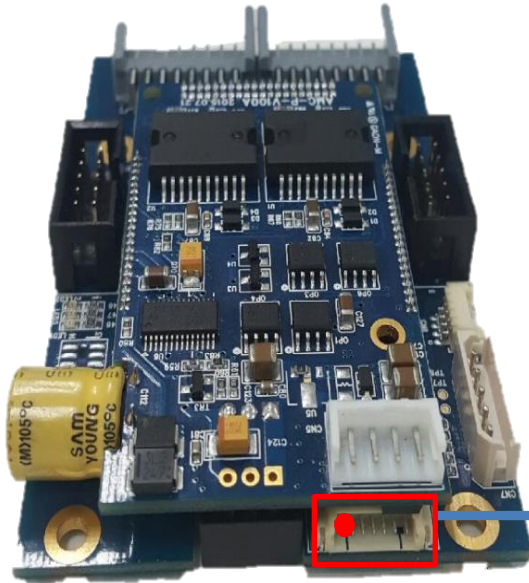
5.4 Encoder 배선



- Encoder의 정확한 결선을 위해 배선을 참고하여 주십시오. 잘못된 결선은 제어기와 모터의 파손 원인이 됩니다.

5.제품의 배선

5.5 Home Sensor 배선



● 1번 핀을 의미합니다.

- Home Sensor의 잘못된 결선은 제어기와 모터 파손의 원인이 되오니 배선을 확인 하신 후 결선하여 주시기 바랍니다.
- Hall Sensor1 IN, Hall Sensor2 IN 는 제어기의 홈 위치 검색용 센서 입력으로 사용 할 수 있습니다.

6. 통신 프로토콜



6.1 통신 개요

- 모션뱅크의 제어기와 통신하기 위해서는 제어기의 프로토콜을 준수하여 통신을 해야 합니다. 해당 제품은 ASCII 형태의 데이터를 사용하기 때문에 그에 맞는 명령어를 사용하셔야 합니다. ASCII는 0 ~ 255 (8 비트)를 사용합니다.

- Port는 PC사양에 맞게 설정하셔야 합니다.
(Port 확인방법 P13 참조)
(Port 변경방법 P14 참조)

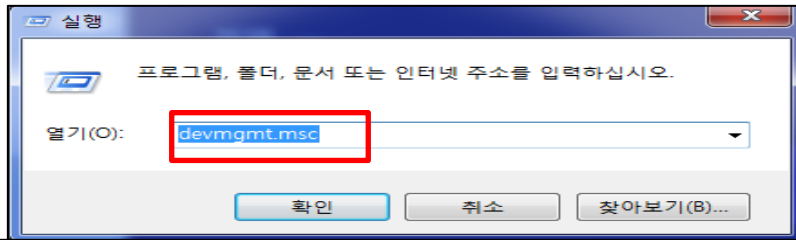
- 제품의 테스트 프로그램을 다운 받은 후 우측과 같은 셋팅 값을 설정해 주시기 바랍니다.
(셋팅 방법은 P15 참조)

Com Port Setting	
Baudrate (통신 속도)	19200
Data Bits (데이터 비트)	8
Stop Bits (정지 비트)	1
Parity (패리티)	None
Port	COM Number

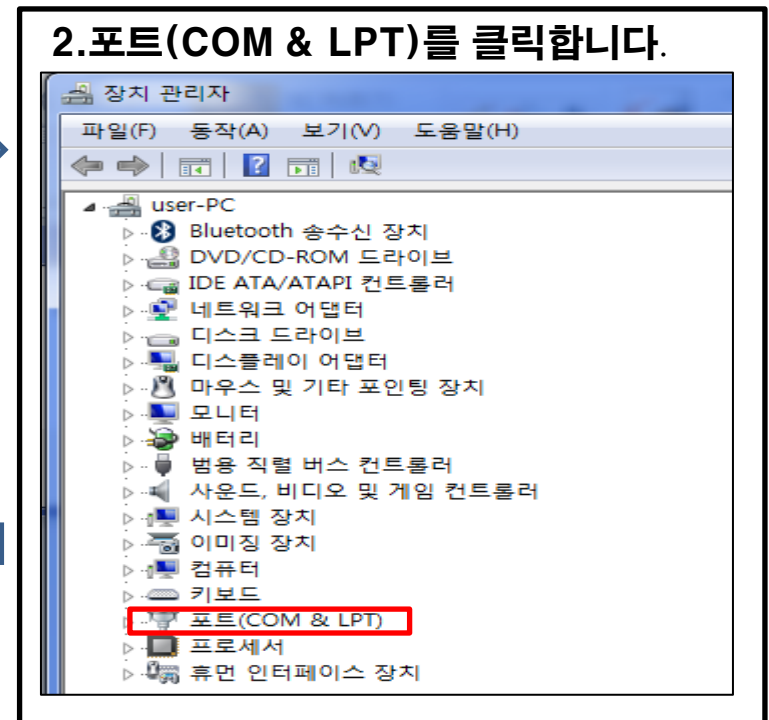
6.통신 프로토콜

6.2 Port 확인 방법

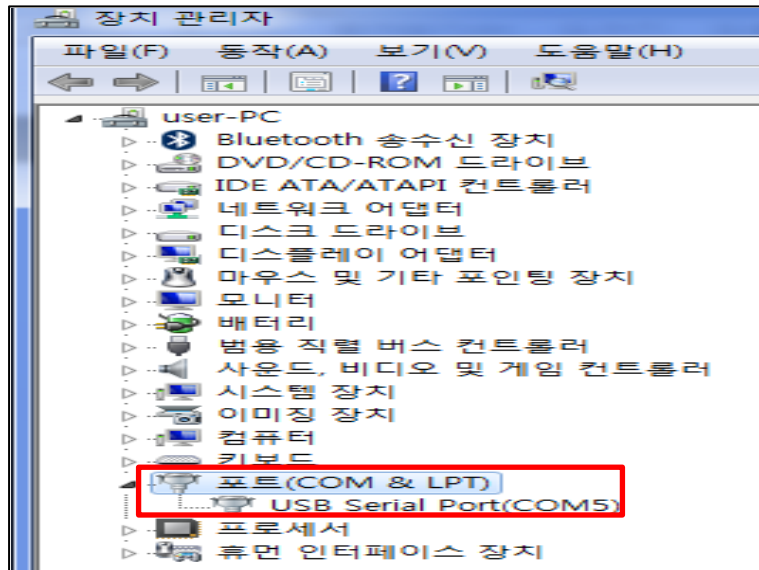
1. 실행 프로그램을 클릭한 후 devmgmt.msc를 입력합니다.



2. 포트(COM & LPT)를 클릭합니다.



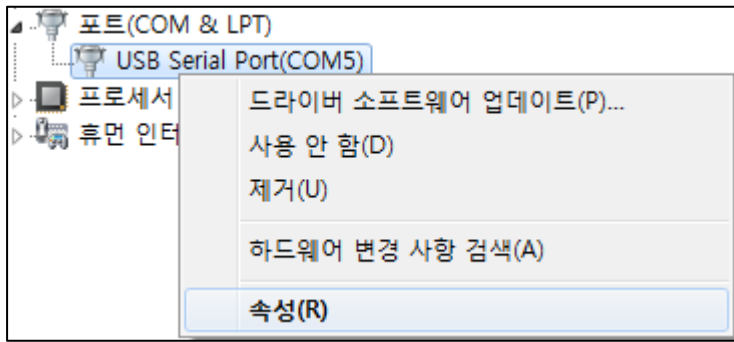
3. Port를 확인합니다.



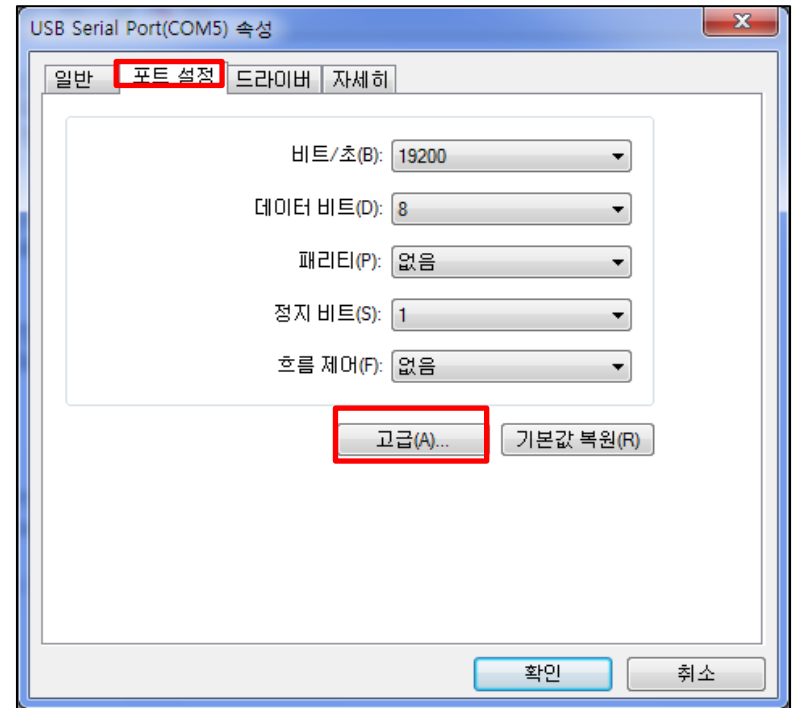
6. 통신 프로토콜

6.3 Port 변경 방법

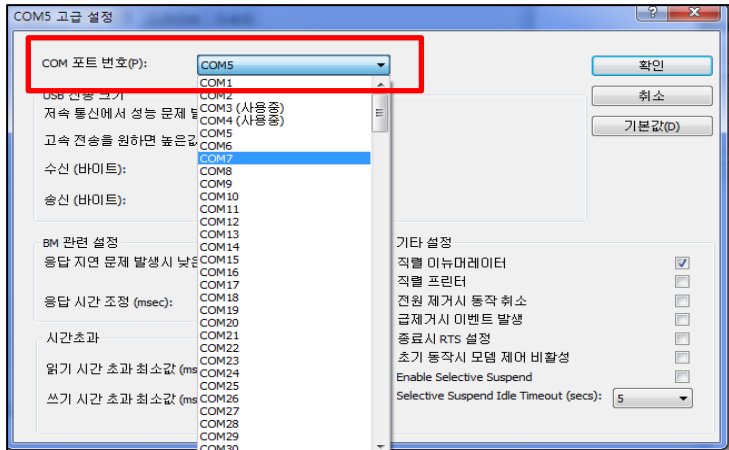
1. USB Serial Port를 마우스 오른쪽 버튼을 누르고 속성을 클릭합니다.



2. 속성 창이 뜨면 포트 설정을 클릭한 후에 고급(A)를 클릭합니다.



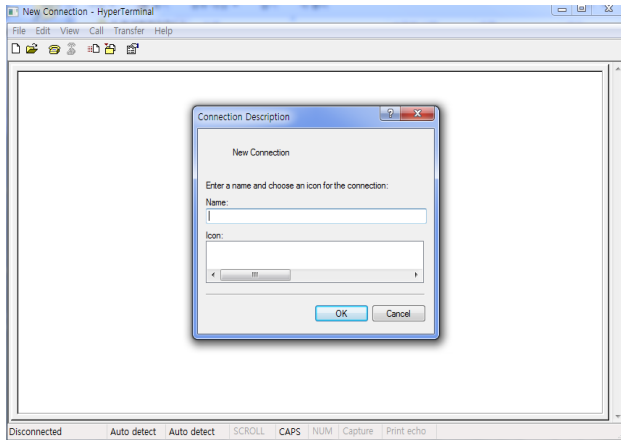
3. 고급 설정 창에서 원하는 COM 포트 번호를 설정한 후에 확인을 누릅니다.



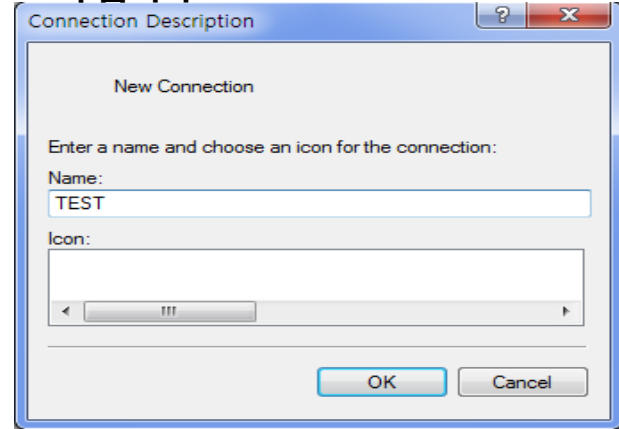
6.통신 프로토콜

6.4 RS232 통신(하이퍼 터미널 세팅 방법)

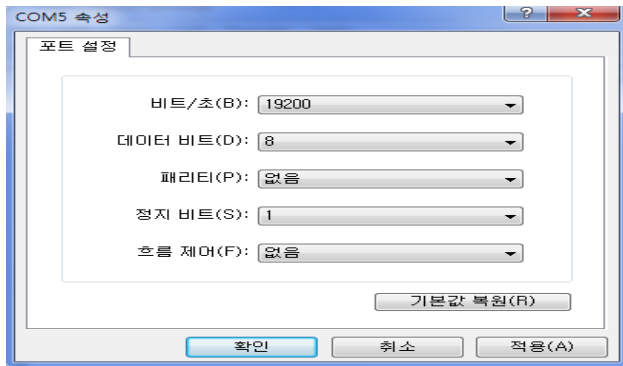
1.하이퍼 터미널을 엽니다.



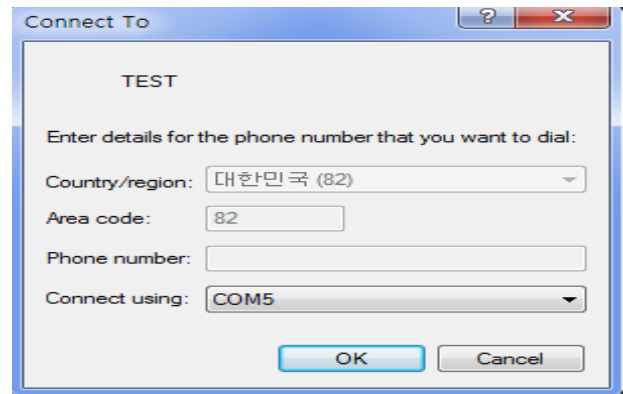
2.Name(이름)입력 후 확인 OK 버튼을 누릅니다.



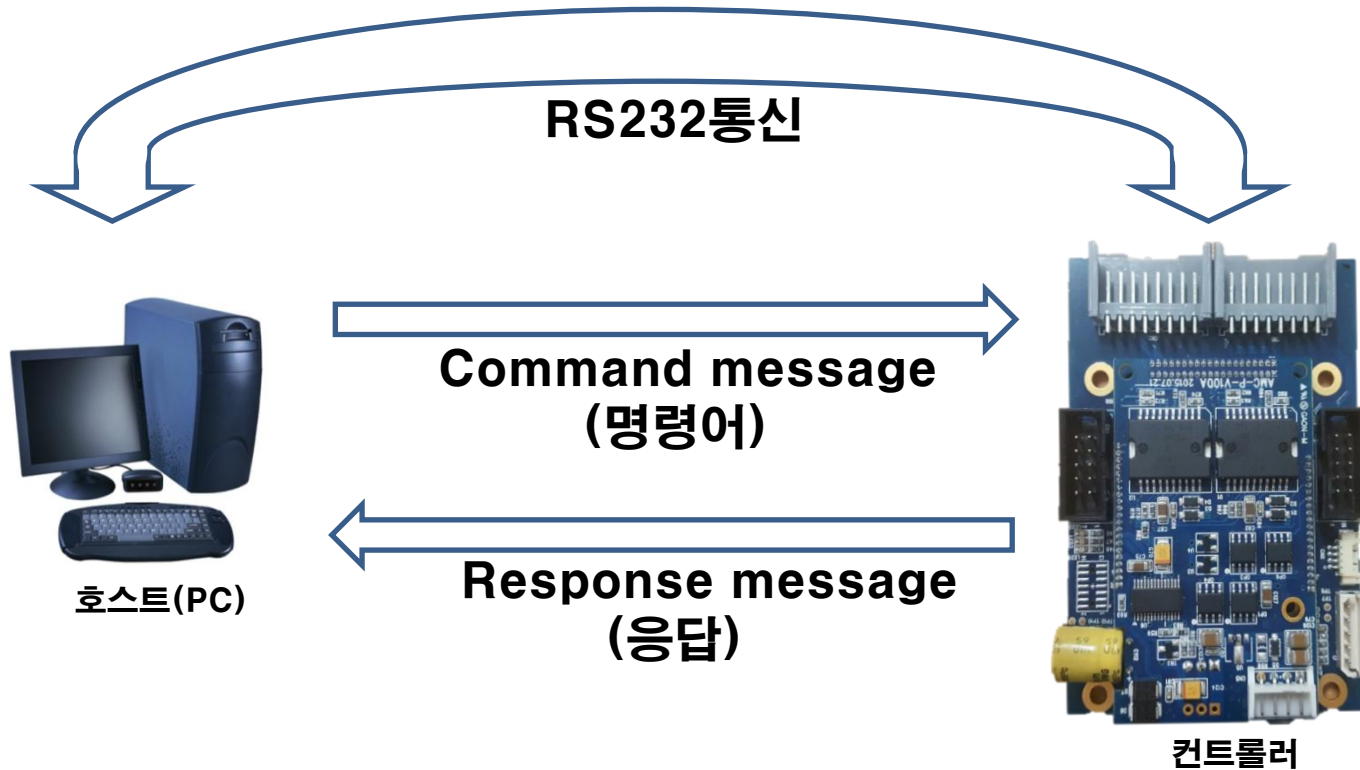
4.포트 등록정보 창에서 아래와 같이 설정 후 적용→ 확인을 누릅니다.
비트/초 : 19200, 흐름제어:없음



3.컨버터에 할당 된 Port를 선택하고 확인을 누릅니다.



7. 명령 프로토콜 개요



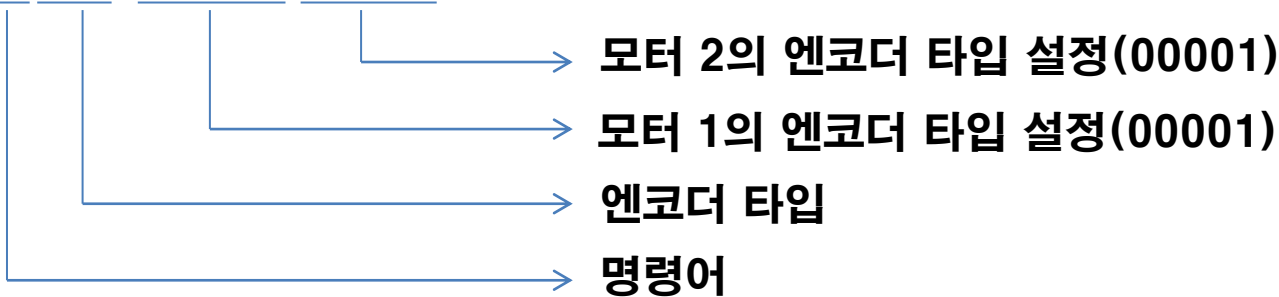
- 호스트가 Command Message를 컨트롤러에 전송하게 되면 컨트롤러는 Command Message에 상응 하는 Response Message를 호스트에 전송하게 됩니다.
- 호스트가 전송하는 Command Message의 종료 문자는 크게 3가지로 이루어져 있습니다.
 - ; 명령어를 사용하고 끝내는 통상적인 종료 문자입니다.
 - ? 설정한 명령어를 확인할 때 사용하는 종료 문자입니다.
 - ! 컨트롤러에서 발생된 예외상황에 의해 호스트에 일방적으로 송신하는 종료 문자입니다.

8.모터 구동 파라미터 설정

8.1 S1을 이용한 엔코더 타입 설정

- S1001 명령은 엔코더 타입을 설정하는 명령어 입니다.
- 엔코더 타입은 오픈 컬렉터 방식과 라인드라이버 방식으로 나뉘며 이 제품은 라인드라이버 방식만을 지원합니다.
- 명령어 포맷

S1001,dddd,dddd;



■ S1을 이용한 엔코더 타입 설정 예시

*설정 전 기본값 : S1001,00001,00001;

	명령어(Command msg)	응답(Response msg)	설명
1	S1001,00001,00001;	S1001,00001,00001;	제어기의 엔코더 타입을 라인 드라이버로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S1001?	S1001,00001,00001;	설정된 엔코더 타입을 확인합니다.

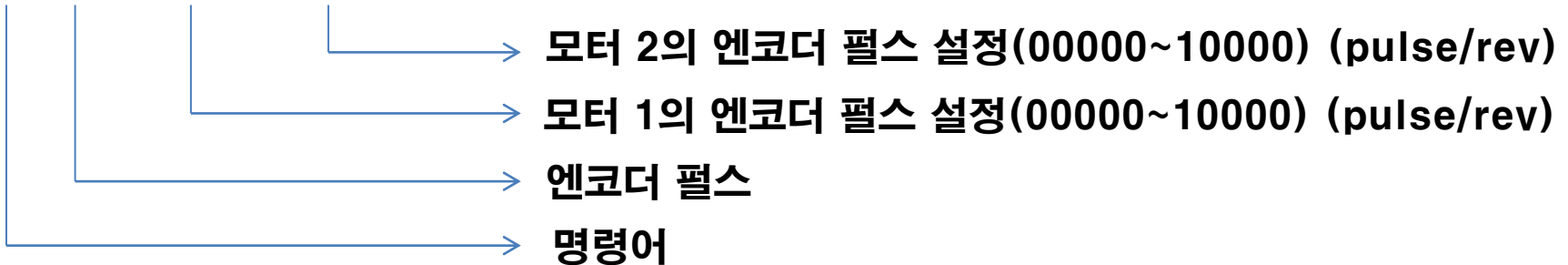
8.모터 구동 파라미터 설정

8.2 S1을 이용한 엔코더 펄스 설정

■ S1002 명령은 모터가 1회전할 때 엔코더 펄스 수를 설정하는 명령어 입니다.

■ 명령어 포맷

S1002,dddd,dddd;



■ S1을 이용한 엔코더 펄스 설정 예시

*설정 전 기본값 : S1002,00128,00128;

	명령어(Command msg)	응답(Response msg)	설명
1	S1002,00500,02500;	S1002,00500,02500;	모터 1의 엔코더 펄스 값을 500pulse/rev으로 설정하고 모터 2의 엔코더 펄스 값을 2500pulse/rev으로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S1002?	S1002,00500,02500;	설정한 엔코더 펄스 값을 확인합니다.

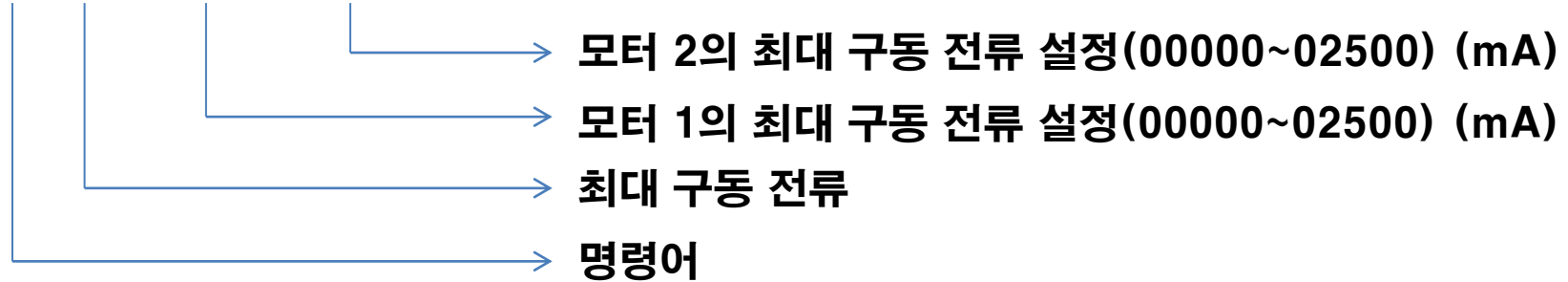
8.모터 구동 파라미터 설정

8.3 S1을 이용한 최대 구동 전류 설정

- S1003 명령은 제어 시 사용하는 최대 구동 전류값을 설정하는 명령어입니다.
- 제어기측에서 모터로 흐르는 전류 값을 제한할 수 있습니다.

■ 명령어 포맷

S1003,dddd,dddd;



■ S1을 이용한 최대 구동 전류 설정 예시

*설정 전 기본값 : S1003,00300,00300;

	명령어(Command msg)	응답(Response msg)	설명
1	S1003,00500,01500;	S1003,00500,01500;	모터 1의 최대 구동 전류값을 500mA로 설정하고 모터 2의 최대 구동 전류값을 1500mA로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S1003?	S1003,00500,01500;	설정된 최대 구동 전류 값을 확인합니다.

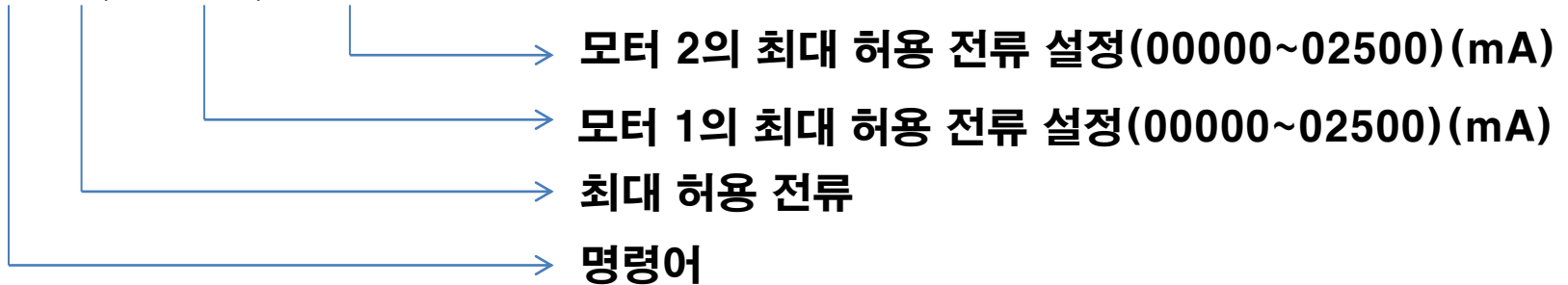
8.모터 구동 파라미터 설정

8.4 S1을 이용한 최대 허용 전류 설정

- S1004 명령은 최대 허용 전류를 설정하는 명령어 입니다.
- 제어기로 입력되는 허용 전류를 제한함으로써 구동회로를 보호하기 위한 명령어 입니다.

■ 명령어 포맷

S1004,dddd,dddd;



■ S1을 이용한 최대 허용 전류값 설정 예시

*설정 전 기본값 : S1004,00899,00899;

	명령어(Command msg)	응답(Response msg)	설명
1	S1004,00500,01500;	S1004,00500,01500;	모터 1의 허용 되는 전류 값을 500mA로 설정하고 모터 2의 허용 되는 전류 값을 1500mA로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S1004?	S1004,00500,01500;	설정한 최대 허용 전류 값을 확인합니다.

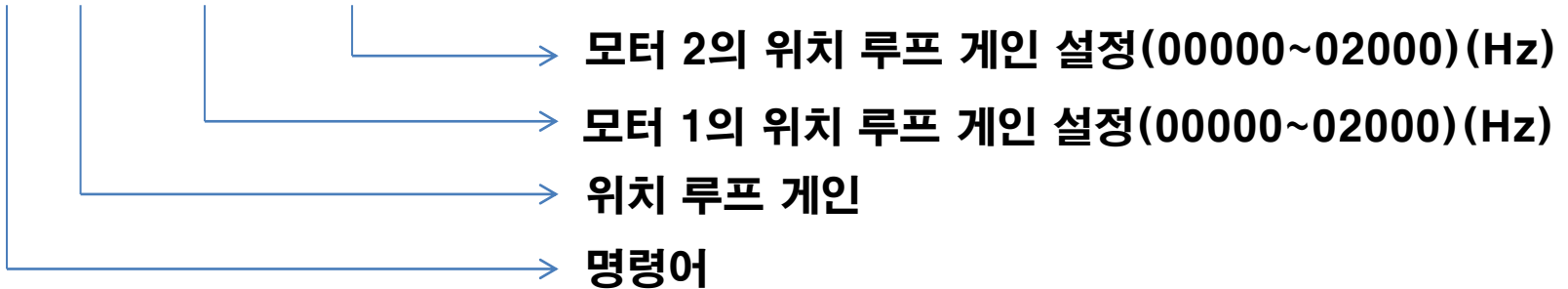
8.모터 구동 파라미터 설정

8.5 S2명령어를 이용한 위치 루프 게인 설정

■ S2001 명령은 위치 루프 게인을 설정하는 명령어입니다.

■ 명령어 포맷

S2001,dddd,dddd;



■ S2을 이용한 위치 루프 게인값 설정 예시

*설정 전 기본값 : S2001,00005,00005;

	명령어(Command msg)	응답(Response msg)	설명
1	S2001,01500,02000;	S2001,01500,02000;	모터 1의 위치 루프 게인 값을 1500Hz로 설정하고 모터 2의 위치 루프 게인 값을 2000Hz로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S2001?	S2001,01500,02000;	설정된 위치 루프 게인값을 확인합니다.

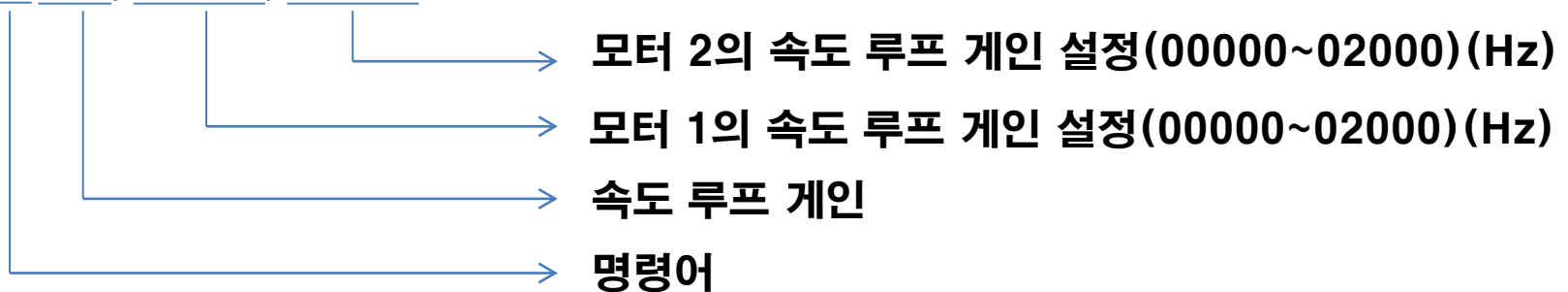
8.모터 구동 파라미터 설정

8.6 S2명령어를 이용한 속도 루프 게인 설정

■ S2002 명령은 속도 루프 게인을 설정하는 명령어 입니다.

■ 명령어 포맷

S2002, ddddd, ddddd;



■ S2을 이용한 속도 루프 게인값 설정 예시

*설정 전 기본값 : S2002,00005,00005;

	명령어(Command msg)	응답(Response msg)	설명
1	S2002,01500,00500;	S2002,01500,00500;	모터 1의 속도 루프 게인 값을 1500Hz로 설정하고 모터 2의 속도 루프 게인 값을 500Hz로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S2002?	S2002,01500,00500;	설정한 속도 루프 게인값을 확인합니다.

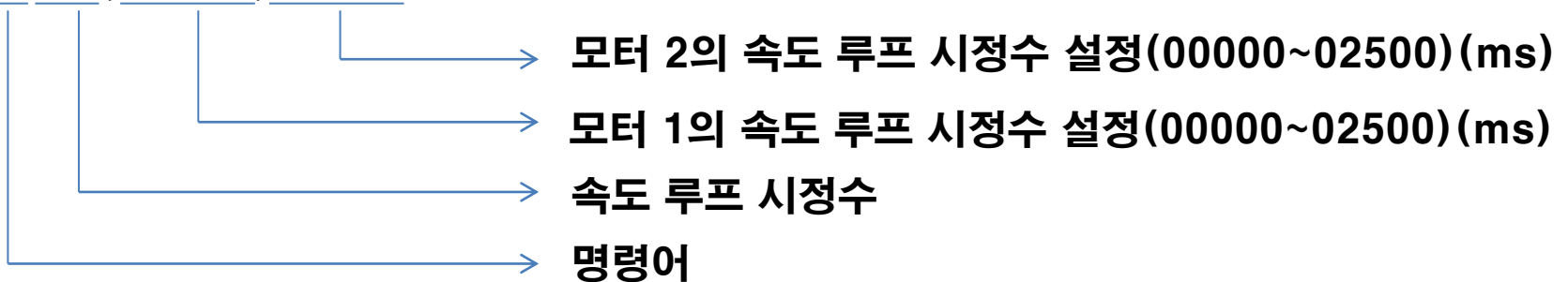
8.모터 구동 파라미터 설정

8.7 S2명령어를 이용한 속도 루프 시정수 설정

■ S2003 명령은 속도 루프 시정수를 설정하는 명령어 입니다.

■ 명령어 포맷

S2003,dddd,dddd;



■ S2을 이용한 속도 루프 시정수 설정 예시

*설정 전 기본값 : S2003,00030,00030;

	명령어(Command msg)	응답(Response msg)	설명
1	S2003,01500,00500;	S2003,01500,00500;	모터 1의 속도 루프 시정수를 1500ms 모터 2의 속도 루프 시정수를 500ms로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S2003?	S2003,01500,00500;	설정된 속도 루프 시정수값을 확인합니다.

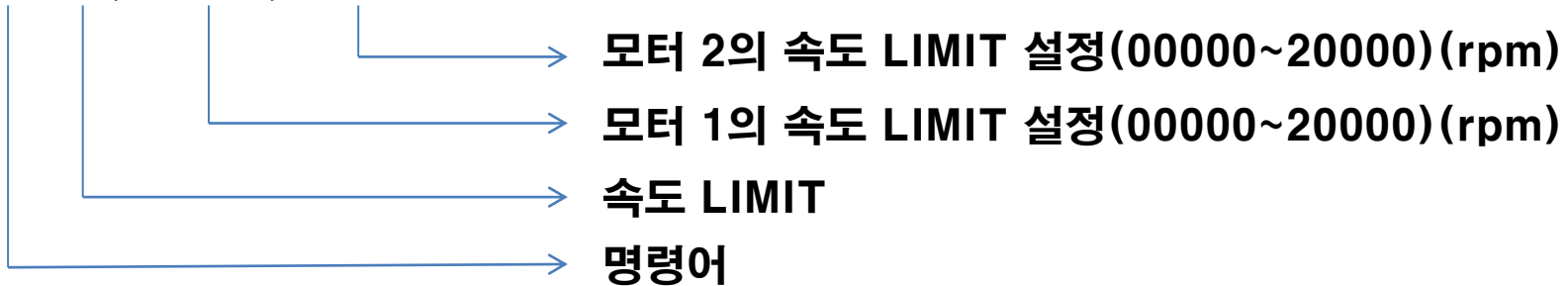
8.모터 구동 파라미터 설정

8.8 S2명령어를 이용한 속도 LIMIT 설정

■ S2004 명령은 속도 LIMIT을 설정하여 모터를 보호하기 위한 명령어 입니다.

■ 명령어 포맷

S2004,dddd,dddd;



■ S2을 이용한 속도 LIMIT설정 예시

*설정 전 기본값 : S2004,11000,11000;

	명령어(Command msg)	응답(Response msg)	설명
1	S2004,03500,04500;	S2004,03500,04500;	모터 1의 속도 LIMIT을 3500rpm 모터 2의 속도 LIMIT을 4500rpm으로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S2004?	S2004,03500,04500;	설정된 속도 LIMIT값을 확인합니다.

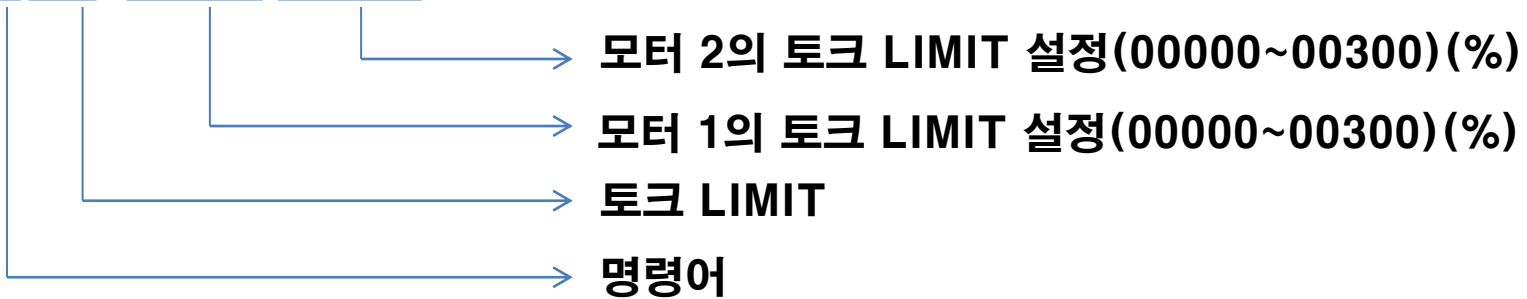
8.모터 구동 파라미터 설정

8.9 S2명령어를 이용한 토크 LIMIT 설정

■ S2005 명령은 토크 LIMIT을 설정하는 명령어 입니다.

■ 명령어 포맷

S2005,dddd,dddd;



■ S2을 이용한 토크 LIMIT 설정 예시

*설정 전 기본값 : S2005,00300,00300;

	명령어(Command msg)	응답(Response msg)	설명
1	S2005,00300,00200;	S2005,00300,00200;	모터 1의 토크 LIMIT을 300% 모터 2의 토크 LIMIT을 200%로 설정합니다.
2	SE;	SE;	파라미터를 저장합니다.
3	S2005?	S2005,00300,00200;	설정된 토크 LIMIT값을 확인합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명			
1	Q1?	Q1ABCD, ABCD;	<div style="text-align: center;"> </div> <p>· 현재 동작 상태 요청 >> 호스트(PC)의 요청에 의해 4자리로 구성된 2 데이터(모터1 ,모터2)를 전송합니다. 2개의 데이터는 현재 컨트롤러와 드라이버에 상태를 나타냅니다.</p>			
			A(모터 구동 가능 여부)	B(모터 구동 상태)	C(위치 도달 여부)	D(작동 이상 유무)
			E : enable-모터구동 가능 D : disable-모터구동 불가능	R : run-모터구동 상태 S : stop-모터정지 상태	I : Inposition - 위치에 도달 O : Outposition - 위치에 도달하지 못함	A : alarm-Fault발생 C : clear-정상
			<p>Ex) Q1EROC,ERIC</p> <div style="text-align: right;"> > 모터2 : 정상 > 모터2 : 위치에 도달 > 모터2 : 모터 구동상태 > 모터2 : 모터구동 가능 > 모터1 : 정상 > 모터1 : 위치에 도달하지 못함 > 모터1 : 모터 구동상태 > 모터1 : 모터구동 가능 </div>			

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
2	BRd;	BRd;	>> Baudrate를 설정합니다. 0 : 9600 1 : 19200 2 : 38400 3 : 57600 4 : 115200 *Baudrate 설정을 설정한 후 테스트 프로그램에 Baudrate도 재설정 해주시기 바랍니다. Ex)BR1; -Baudrate를 19200으로 설정합니다. *테스트 프로그램 Baudrate설정 P15참조
	BR?	BRd;	>> 설정한 Set Baudrate를 확인합니다.
3	PE;	PE;	>> Motor 출력 ON 명령에 의해 모터 출력을 ON합니다. *모터 구동을 하시려면 꼭 사용하여야 하는 명령어 입니다.
4	PD;	PD;	>> Motor 출력 OFF 명령에 의해 모터 출력을 OFF합니다.
5	ME;	ME;	>> Motion Enable 명령에 의해 설정한 motion을 시작합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
7	MD;	MD;	>> Motion Disable 명령에 의해 설정한 motion을 멈춥니다.
8	ED;	ED;	>> Emergency Stop 명령에 의해 motion을 급정지 합니다.
9	SMd;	SMd;	>> Motor의 제어 모드 설정 (위치 제어 모드/속도 제어 모드) 명령에 의해 모터의 제어모드를 설정합니다. 0 : 제어모드 비활성화 1 : 속도 제어 모드 2 : 이동속도를 설정하여 목표위치까지 도달하는 위치 제어 모드 3 : 이동시간을 설정하여 목표위치까지 도달하는 위치 제어 모드 Ex)SM1; -제어기의 제어모드를 속도 제어 모드로 설정합니다.
	SM?	SMd;	>> 현재 Motor의 제어모드 설정을 확인합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
10	TQ?	TQFFF, FFF;	<p>TQFFF,FFF;</p> <p>모터2의 정격 전류에 대한 비율 모터1의 정격 전류에 대한 비율</p> <p>>> 구동되고 있는 현재 모터 전류를 요청합니다.(단위 %/최대 100%) >> 호스트(pc)의 요청에 의해 현재 모터 전류를 정격 전류에 대한 비율로 표시합니다. Ex)TQ050,050; -전류가 정격전류의 50% 흐르고 있습니다.</p>
11	SV±dddd, ±dddd;	SV±dddd, ±dddd;	<p>SV±dddd,±dddd;</p> <p>모터2 속도 모터1 속도</p> <p>>> 속도 제어 모드 시 속도를 설정합니다. >> +는 정방향 -는 역방향으로 설정합니다. (단위는 rpm) Ex)SV+00100,-00200; -모터 1 : 정방향 100rpm, 모터 2 :역방향 200rpm</p>
	SV?	SV±dddd, ±dddd;	>> 설정한 속도값을 확인합니다.
12	Saaaaa, aaaa;	Saaaaa, aaaa;	<p>Saaaaa,aaaa;</p> <p>모터2의 가/감속 시간 모터1의 가/감속 시간</p> <p>>> 속도 제어 모드 시 가/감속 시간을 설정합니다.(단위는 ms) Ex)Sa10000,10000; -모터 1,2의 가감속 시간은 10000ms로 설정합니다.</p>
	Sa?	Saaaaa, aaaa;	>> 설정한 가/감속 시간을 확인합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
13	PAFFFFFFF, FFFFFFFF;	PAFFFFFFF, FFFFFFFF;	<p style="text-align: center;">PAFFFFFFF, FFFFFFFF;</p> <p>>> 위치 제어 모드 시의 목표 위치 설정합니다. (엔코더 초기값은 0x8000000) Ex) PA8000FFF, 7FFF000; -현재 엔코더 값 0x8000000을 기준으로 모터 1은 0xFFF(4095)만큼 정방향 이동, 모터 2는 0x1000(4096)만큼 이동합니다.</p>
	PA?	PAFFFFFFF, FFFFFFFF;	>> 설정한 목표 위치 값을 확인합니다.
14	SSdddd, dddd;	SSdddd, dddd;	<p style="text-align: center;">SSdddd, dddd;</p> <p>>> 위치 제어 모드시의 목표지점까지 이동을 할 때 이동속도값 설정 (단위는 rpm) >> 이동속도 설정에 의한 위치 제어 모드 시의 이동속도를 설정합니다. Ex) SS01000, 01000; -모터 1,2의 위치 제어 시의 속도를 1000rpm으로 설정합니다.</p>
	SS?	SSdddd, dddd;	>> 설정한 이동속도 값을 확인합니다.
15	SCdddd, dddd;	SCdddd, dddd;	<p style="text-align: center;">SCdddd, dddd;</p> <p>>> 위치 제어 모드시의 구동하고 있는 모터의 이동속도값을 변경합니다. Ex) SS01000, 01000; -모터 1,2의 위치 제어 시의 속도를 1000rpm으로 변경 설정합니다. (등속구간에서 적용)</p>

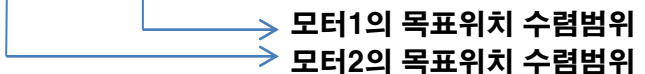
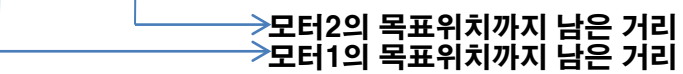
9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
16	STFFFFFFF, FFFFFFF;	STFFFFFFF, FFFFFFF;	<p>STFFFFFFF,FFFFFFF;</p> <p>>> 위치 제어 모드(SM3)일 때 목표지점 이동 시 목표지점까지의 이동시간 값을 설정합니다.(단위ms) Ex) ST00000FF,00000FF; -위치 제어 시의 목표지점까지의 이동시간을 255ms로 설정함, 모터 1,2의 속도는 목표 위치까지의 거리에 설정된 이동시간을 나눈 후 가감속을 시간을 포함하여 자동적으로 산출합니다.</p>
	ST?	STFFFFFFF, FFFFFFF;	>> 설정한 이동 시간 값을 확인합니다.
17	Ssdddd, dddd;	Ssdddd, dddd;	<p>Ssdddd,dddd;</p> <p>>> 위치 제어 모드(SM3)에서의 위치 구동시의 가/감속 시간을 설정합니다.(단위ms) >> 이동시간 설정에 의한 위치 제어 모드 시의 가감속 시간을 설정합니다. Ex)Ss00010,00010; -위치제어 시 모터 1,2의 가감속이 10ms에 이루어지도록 설정합니다.</p>
	Ss?	Ssdddd, dddd;	>> 설정한 가/감속 시간을 확인합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
18	SrFFFFFF, FFFFF;	SrFFFFFF, FFFFF;	<p style="text-align: right;">SrFFFFFF, FFFFF;</p>  <p>>> 위치 제어 모드일때 목표위치 수렴범위를 설정합니다.(단위 Encoder Count) Ex)Sr00100,00200; -모터 1은 목표 위치에 엔코더 펄스 기준 $\pm 0x100(256)$안으로 도달했을 때 도착으로 판단합니다. -모터 2는 목표 위치에 엔코더 펄스 기준 $\pm 0x200(512)$안으로 도달했을 때 도착으로 판단합니다.</p>
	Sr?	SrFFFFFF, FFFFF;	>> 설정한 목표위치 수렴 범위를 확인합니다.
19	TG?	TGFFFFFFFF, FFFFFFFF!	<p style="text-align: right;">TGFFFFFFFF, FFFFFFFF!</p>  <p>>> 위치 제어 모드에서 목표위치에 도달함을 알리고 목표위치와 남은 거리를 알립니다. >> 호스트의 요청에 의해 현재위치와 목표위치 사이에 오차(엔코더 펄스 기준)를 리턴 혹은 목표위치에 도달하는 순간 컨트롤러에서 호스트로 메시지를 전송합니다. Ex)수렴범위가 $0x100(256)$일 때 -TG0000FF(255),0000110(272)! 가 호스트로 전송되는 경우에는 모터 1이 수렴되었음을 알립니다. TG0000FF(256),0000F0(240)! 가 호스트로 전송되는 경우에는 모터1,2이 수렴되었음을 알립니다.</p>

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
20	QP?	QPFFFFFFF, FFFFFFF;	<p style="text-align: center;">QPFFFFFFF,FFFFFFF;</p> <p style="text-align: right;"> </p> <p>>> 현재의 위치 값 (단위 Encoder count) >> 호스트의 요청에 의해 현재의 위치값 전송합니다. (엔코더 초기값은 0x8000000)</p>
21	GsD,dddd;	GsD,dddd;	<p style="text-align: center;">GsD,dddd;</p> <p style="text-align: right;"> </p> <p>>> 홀 센서를 사용한 homing 동작 설정 >> 1리(D) 10 진수는 동작모드, 나머지 5자리(ddddd) 10진수는 홀 동작 시 속도입니다. (혹은 최대속도)(단위rpm)</p> <p>0 : 홀 위치 신호를 상승 엣지에서 검출하고, 다수의 상승 엣지 발생시에는 탐색방향 기준 첫번째 엣지를 홀 위치로 정합니다. 1 : 홀 위치 신호를 하강 엣지에서 검출하고, 다수의 하강 엣지 발생시에는 탐색방향 기준 첫번째 엣지를 홀 위치로 정합니다.</p> <p>Ex)Gs0,01000; -홀 위치 신호를 상승엣지에서 검출하며, homing동작시 모터 1,2는 1000rpm으로 동작하게 됩니다.</p> <p>*최저속도 : 1000rpm</p>
	Gs?	GsD,dddd;	>> 설정한 동작 모드와 홀 동작시 속도를 확인합니다.

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러->PC	설명
22	EAdd,FFFFFF,FFFFFF;	EAdd,FFFFFF,FFFFFF;	<p style="text-align: center;">EAdd,FFFFFF,FFFFFF;</p> <p> </p> <p> 모터1의 모드 설정 ← 모터2의 모드 설정 ← </p> <p> →모터2의 엔코더 카운터 값 →모터1의 엔코더 카운터 값 </p> <p> >> 엔코더 카운터 값 재설정(단위 Encoder count) >> 2자리의 10진수에 의해 모터선택 및 모드 설정, 첫째자리는 모터1, 둘째자리는 모터2, 나머지 7자리 16진수는 모터 1,2 엔코더 카운터 값입니다. </p> <p> 0 : 설정안함 1 : 엔코더 카운터 값 초기값으로 재설정 전송된 엔코더 카운터 값은 무시하고, 0x8000000로 설정합니다. 2 : 엔코더 카운터 값 전송된 엔코더 카운터 값으로 설정합니다 3 : 현재의 엔코더 카운터 값을 전송된 엔코더 카운터 값만큼 증가시킵니다. 4 : 현재의 엔코더 카운터 값을 전송된 엔코더 카운터 값만큼 감소시킵니다. </p> <p> Ex)-EA01,FFFFFF,FFFFFF; 모터 1은 엔코더 카운터 재설정 안함, 모터2는 엔코더 카운터 값을 0x8000000으로 재설정합니다. -EA22,0000FF(255),0000D4(212); 모터 1,2의 엔코더 카운터 값을 각각 0x0000FF(255),0x0000D4(212)으로 재설정합니다. -EA34,0000FD(253),0000F0(250); 모터 1은 현재의 엔코더 카운터 값에서 0x0000FD(253)만큼 증가시켜 재설정하고, 모터2는 현재의 엔코더 카운터 값에서 0x0000F0(250)만큼 감소시켜 재설정합니다. </p> <p> *최대값 : 0xFFFFFFFF; 최소값 : 0x0000000; </p>

9. 명령 프로토콜 설명



구분	명령어 Command message PC->컨트롤러	응답 Response Message 컨트롤러 ->PC	설명
23	Ghdd;	Ghdd;	<p style="text-align: right;">Ghdd;</p> <p style="text-align: right;"> </p> <p>>> 홈 센서를 사용한 homing 동작 수행 >> 첫째자리는 모터1, 둘째자리는 모터2, 설정된 homing 동작이 완료되면 호스트로 동작 완료 메시지를 보냅니다.</p> <p>0 : 선택안함 1 : 정방향으로 탐색하여 homing 동작 수행, homing 동작 완료 메시지를 전송합니다. 2 : 역방향으로 탐색하여 homing 동작 수행, homing 동작 완료 메시지를 전송합니다. 3 : homing 동작 완료 메시지</p> <p>Ex) Gh01; → Gh01; Gh03! (모터 2가 홈 동작 완료되면 전송) Gh02; → Gh02; GH03! (모터 2가 홈 동작 완료되면 전송) Gh11; → Gh11; Gh33! (모터1,2가 홈 동작 완료되면 전송) Gh21; → Gh21; Gh33! (모터1,2가 홈 동작 완료되면 전송)</p>
	Gh?	Ghdd!	>> Homing 동작 시 설정한 동작 수행 모드를 확인합니다.
24	Qh?	QhFFFFFF, FFFFFFFF;	<p style="text-align: right;">QhFFFFFF, FFFFFFFF;</p> <p style="text-align: right;"> </p> <p>>> 홈 센서 감지시점의 위치 값 요청합니다. (단위 Encoder count) >> 홈 센서 감지시점의 홈 위치 값 리턴합니다. (홈 위치 값은 산출 방법 Gs에 의해) >> 홈 위치가 검출되지 않은 상태인 경우에 값을 요청하면 값을 숫자대신 기호 등으로 표시합니다.</p> <p>Ex) Qh00000B8(184),00000B9(185); -회전에 의해 홈 위치가 감지된 시점이 모터 1에서는 엔코더 카운터 값이 0x00000B8(184)이고, 모터2에서는 엔코더 카운터 값이 0x00000B9(185)됩니다.</p>

10. 명령 프로토콜 예시

10.1 Ss명령어를 사용한 위치 제어 예시

- 현재의 엔코더 값을 기준으로 모터 1,2모두 정방향으로 0x60000만큼 가 감속이 6000ms에 이루어집니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM3;	모터 1,2를 이동시간 설정에 따른 위치 제어 모드로 설정합니다.
3	Ss06000,06000;	위치 제어 시 모터 1,2의 가 감속이 6000ms에 이루어지도록 설정합니다.
4	PA8060000,8060000;	모터 1,2모두 정방향으로 0x60000만큼 이동합니다.
5	ME;	명령에 의해 motion을 시작합니다.
6	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.2 ST명령어를 사용한 위치 제어 예시

- 현재의 엔코더 값을 기준으로 모터 1은 정방향으로 0x60000만큼
모터 2는 정방향으로 0x10000만큼 이동시간 10ms안에 이동합니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM3;	모터 1,2를 이동시간 설정에 따른 위치 제어 모드로 설정합니다.
3	ST000000A,000000A;	위치 제어 모드 시 목표위치까지의 이동시간을 10ms로 설정합니다.
4	PA8060000,8010000;	모터 1은 정방향으로 0x60000만큼 모터 2는 정방향으로 0x10000만큼 이동합니다.
5	ME;	명령에 의해 motion을 시작합니다.
6	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.3 SS명령어를 사용한 위치 제어 예시

- 현재의 엔코더 값을 기준으로 모터 1은 정방향으로 0x70000만큼
모터 2는 정방향으로 0x60000만큼 3000rpm으로 회전하게 됩니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM2;	모터 1,2를 이동속도 설정에 따른 위치 제어 모드로 설정합니다.
3	SS03000,05000;	위치 제어 시 모터 1의 속도를 3000rpm으로 설정하고 모터 2의 속도를 5000rpm으로 설정합니다.
4	PA8070000,8060000;	모터 1을 정방향으로 0x70000만큼 모터 2를 정방향으로 0x60000만큼 이동합니다.
5	ME;	명령에 의해 motion을 시작합니다.
6	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.4 SC명령어를 사용한 위치 제어 예시

- 현재의 엔코더 값을 기준으로 모터 1,2모두 정방향으로 0x70000만큼 3000rpm으로 회전하다가 500rpm으로 회전 속도를 변경할 수 있습니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다
2	SM2;	모터 1,2를 이동속도 설정에 따른 위치 제어 모드로 설정합니다.
3	SS03000,03000;	위치 제어 시 모터 1,2의 속도를 3000rpm으로 설정합니다.
4	PA8070000,8070000;	모터 1,2모두 정방향으로 0x70000만큼 이동합니다.
5	ME;	명령에 의해 motion을 시작합니다.
6	SC00500,00500;	위치 제어 시 모터 1,2의 속도를 500rpm으로 변경 설정합니다.
7	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.5 Sr명령어를 사용한 위치 제어 예시

- 현재의 엔코더 값을 기준으로 모터 1,2모두 3000rpm으로 회전을 하며 목표 위치인 0x70000의 $\pm 0x200$ 안으로 도달했을 때 도착으로 판단합니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM2;	모터 1,2를 이동속도 설정에 따른 위치 제어 모드로 설정합니다.
3	SS03000,03000;	위치 제어 시 모터 1,2의 속도를 3000rpm으로 설정합니다.
4	PA8070000,8070000;	모터 1,2모두 정방향으로 0x70000만큼 이동합니다.
5	Sr00200,00200;	모터 1,2의 목표 위치에 도착 수렴범위를 $\pm 0x200$ 로 설정합니다.
6	ME;	명령에 의해 motion을 시작합니다.
6	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.6 Sa/SV명령어를 이용한 속도 제어 예시

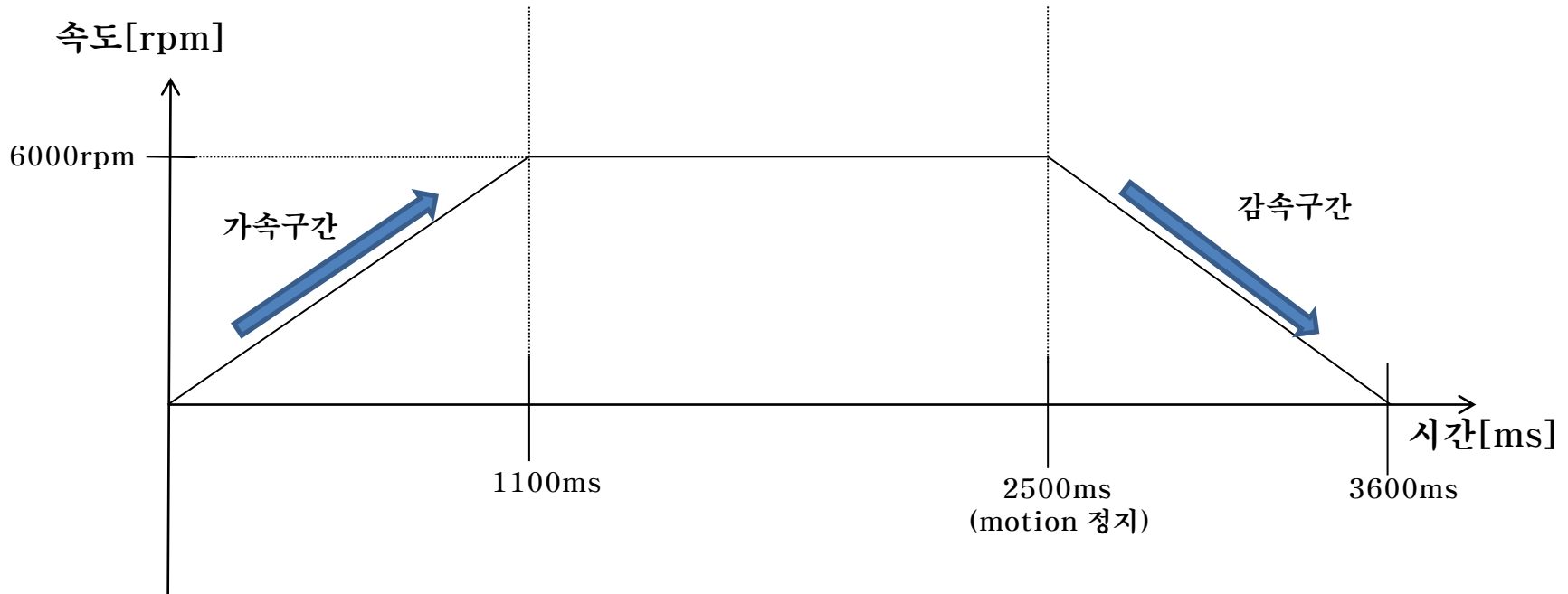
■ 아래 표와 같이 순서대로 명령을 입력하면 Sa로 설정된 가/감속률로 SV로 설정한 속도[rpm]로 회전하게 됩니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM1;	모터 1,2를 속도 제어 모드로 설정합니다.
3	Sa01100,02200;	모터 1의 가/감속 시간을 1100ms 모터 2의 가/감속 시간을 2200ms로 설정합니다. * 가속시간 : 0부터 1100ms까지 가속 감속시간 : motion 정지 부터 1100ms후 까지 감속
4	SV+06000,-06000;	모터1은 정방향 6000rpm, 모터2는 역방향 6000rpm 설정합니다.
5	ME;	명령에 의하여 Motion을 시작합니다.
6	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

10. 명령 프로토콜 예시

10.7 SV명령어를 이용한 가/감속-시간 그래프

■ 1100ms안에 6000rpm의 가/감속 속도제어를 나타내는 그래프입니다.



10. 명령 프로토콜 예시

10.8 ED명령어를 이용한 작동 중 정지

■ 아래 표와 같이 순서대로 명령을 입력하면 Sa로 설정된 가/감속률로 SV로 설정한 속도[rpm]로 회전하다가 ED로 모터의 작동을 급정지 합니다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	SM1;	모터 1,2를 속도 제어 모드로 설정합니다.
3	Sa00100,00200;	모터 1의 가감속 시간은 100ms 모터 2의 가감속 시간은 200ms로 설정합니다.
4	SV+01000,-01000;	모터1은 정방향 1000rpm, 모터2는 역방향 1000rpm 설정합니다.
5	ME;	명령에 의하여 Motion을 시작합니다.
6	ED;	명령에 의해 Motion을 급정지 합니다.

10. 명령 프로토콜 예시

10.9 홈 관련 통신 예시

■ 아래 과정은 모터 1,2가 정방향으로 탐색하여 homing 동작을 수행하고, homing 동작 완료 시 호스트로 완료 메시지를 보냅니다. 홈 동작 시 속도는 500rpm이다.

순차	명령어	설명
1	PE;	Motor 출력 ON합니다.
2	Gs1,00500;	홈 위치 신호를 하강 엣지에서 검출, 홈 동작 시 속도는 500rpm으로 설정합니다.
3	Gh11;	모터 1,2가 정방향으로 탐색하여 homing 동작을 수행하고, homing 동작이 완료되면 호스트로 동작 완료 메시지를 보냅니다.
4	MD;	명령에 의해 motion을 정지합니다. *motion을 정지하려고 할 때 사용해주시기 바랍니다.

11. Fault 처리

■ FAULT Code의 표시

제어 도중 갑자기 동작이 멈춘다거나 전류가 높다거나 하는 등 이상증상이 발생했을 시에는 Q2명령어를 통해 현재 컨트롤러의 Fault 상황을 나타냅니다.

	명령어 Commend Message	응답 Response Message	설명								
1	Q2?	Q2ABCD,ABCD;	<p style="text-align: center;">Q2ABCD,ABCD;</p> <div style="display: flex; justify-content: center; align-items: center;"> <div style="border-left: 1px solid black; border-right: 1px solid black; width: 100px; height: 100px; margin-right: 10px;"></div> <div style="text-align: center;"> <p>→ 모터2</p> <p>→ 모터1</p> </div> </div> <p>• 정의된 Fault Code(status) ▶ 호스트(PC)의 요청이나 상황 발생 시 4자리로 구성된 두 데이터를 전송합니다. 2개의 데이터는 현재 컨트롤러의 Fault 상황을 나타냅니다. Fault상황이 발생되면 종료문자<!>로 하여 데이터를 전송합니다. (기본 Fault는 과전류,과전압,속도에러,홀 센서 에러)</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th style="color: red;">A</th> <th style="color: red;">B</th> <th style="color: red;">C</th> <th style="color: red;">D</th> </tr> </thead> <tbody> <tr> <td>O : Over Voltage U : Under Voltage N : None</td> <td>H : Open Hall sensor C : Current sensing N : None</td> <td>C : Over Current L : Over Load S : Over Speed P : Over Position N : None</td> <td>E : Eeprom P : Parameter N : None</td> </tr> </tbody> </table>	A	B	C	D	O : Over Voltage U : Under Voltage N : None	H : Open Hall sensor C : Current sensing N : None	C : Over Current L : Over Load S : Over Speed P : Over Position N : None	E : Eeprom P : Parameter N : None
A	B	C	D								
O : Over Voltage U : Under Voltage N : None	H : Open Hall sensor C : Current sensing N : None	C : Over Current L : Over Load S : Over Speed P : Over Position N : None	E : Eeprom P : Parameter N : None								

11. Fault 처리

■ FAULT 표시의 예

	응답 Response Message	설명
1	Ex) Q2ONNN,NNCN!	>> 모터 1은 과전압으로 인한 Fault 상황 발생 모터 2는 과전류로 인한 Fault 상황 발생
2	Ex) Q2NNSN,NNLN!	>> 모터 1은 속도에러로 인한 Fault 상황 발생 모터 2는 과부화로 인한 Fault 상황 발생
3	Ex) Q2UNNN,NNPN!	>> 모터 1은 저전압으로 인한 Fault 상황 발생 모터 2는 위치에러로 인한 Fault 상황 발생

■ FAULT 처리의 예

	명령어 Commend Message	설명
1	Q2?	Q2? 명령어를 전송하여 리턴되는 데이터를 분석하여 현재 Fault 상태를 판단
2	PR:	Fault가 발생한 경우 원인을 제거 한 후 PR 명령을 사용하여 Fault Flag를 Clear함