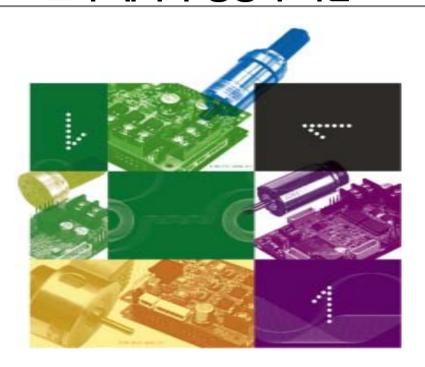
제어기 명령어 LIST

Network Based Digital Motor Controller

모터 제어기 명령어 파일





http://www.robocube.co.kr



제어기 명령어 LIST

1 명령어 구분 -----

각 명령어 뒤에 % + # ^ 표시는 명령어가 적용되는 MCU종류를 나타냅니다. 다음 표는 모터 제어기 별로 할당된 % + # ^ 기호를 보여줍니다. 단 각각의 명령어는 구동모터의 종류와 제어기 모델에 따라서 실제로는 적용이 되지 않는 경우가 있습니다.

	기호	+(TMS320F2811)	#(TMS320F28334/5)	^(TMS320F2809)	%(TMS320LF2406A)
	CUBE-DC2401-DIQ			0	
	CUBE-DC2402-DII	0			
DC	CUBE-DC2408-DID	0			
DC	CUBE-DC2408-SID	0			
	CUBE-DC2416-SIH	0			
	CUBE-DC2416-SDH	0			
	CUBE-BL2401-SIQ			0	
	CUBE-BL2402-DII	0			
	OUDE DI 2400 DID	○ (General)			
BLDC	CUBE-BL2408-DID		○ (Special)		
	CUBE-BL2408-SID	0			
	CUBE-BL2416-SIH	0			
	CUBE-BL2416-SDH	0			
STEP	CUBE-ST2402-DIIE		0		

2 명령어 검색방법 -----

명령어를 찾으실 때에는 찾기(Cntl + F)를 눌러 ";" + "명령어" 를 치시면 쉽게 찾으실 수 있습니다.

명령어에서 ";"는 명령어의 끝은 지정하는 문자로서 명령어 앞쪽에는 붙일 필요가 없습니다. (예를 들어서 다음 페이지에 있는 ;QS; 명령어를 사용할 때 앞쪽의 ";"를 빼고 QS; 만 입력하면 됩니다.

2008년 이전 구매 고객께서는 당사에 문의하여 주십시오.



3 명령어 형식

♣ 각 명령어는 맨앞에 1~3자의 명령어 문자(ASCII code)가 위치하며, 그 뒤에 필요한 파라메터 값이 ',' 를 seperator로 하여 나열된 다음, 마지막에 ';' 이나 '!'로 끝나는 형태로 되어있습니다. 파라메터값이 없거나 ?인 경우도 있습니다. (예로는 PA5100000,5200000; 또는 EsA55A! 또는 Qs; 또는 SA?;)

♣ 모터제어기로 명령이 전송되면 약 200us이내에 처리를 끝내는데, 이때 모터제어기로 전송된 명령의 마지막 끝이 ';' 이면 명령어에 대한 리턴값이 모터제어기로부터 출력됩니다. 이전에 전송된 명령의 처리가 끝난 이후에만 다음 명령어가 처리될수 있습니다. 따라서 모터제어기로부터 이전명령어에 대한 리턴값을 수신한 이후에 다음 명령을 보내야 안전합니다.

♣ 모터제어기로 전송된 명령의 마지막 끝이 '!' 이면 명령어에 대한 리턴값이 모터제어기로부터 출력되지 않습니다. 이 경우에는 이전 명령어의 마지막 값 ('!')을 보낸 다음, 반드시 최소 200uS 이후에 다음 명령의 전송을 개시하여야 합니다.

♣ 파라메터 값은 다음 표과 같은 종류의 값들이 될수 있습니다.

파라메터값 표현형식	파라메터의 의미	사용 예
dddd	16비트 또는 32비트의 부호없는 값을 십진수로 표현한 값	1234 또는 1234567 (제어기로부터 출력되는 값을 표현하는 경우일때 long 변수로서 해당값이 너무 크면 fffffff로 출력 됨)
iiii	16비트의 부호있는 값을 십진수로 표현한 값	-1234
1111	32비트의 부호있는 값을 십진수로 표현한 값	+1234567 (제어기로부터 출력되는 값을 표현하는 경우일때 long 변수로서 해당값이 너무 크면 fffffff로 출력됨)
ffff	부호와 소숫점이 있는 실수값을 십진수로 표현한 값	-0.3456
HL	8비트의 부호없는 값을 16진수로 표현한 값	7A (십진수로는 7*2 ⁴ +10=122에 해당함)
HLHL	16비트의 부호없는 값을 16진수로 표현한 값	1234 (십진수로는 1*2 ¹² +2*2 ⁶ +3*2 ⁴ +4=4660에 해당함)



파라메터값 표현형식	파라메터의 의미	사용 예
HLHLHL	24비트의 부호없는 값을 16진수로 표현한 값	12345B (십진수로는 1*2 ²⁰ +2*2 ¹⁶ +3*2 ¹² +4*2 ⁶ +5*2 ⁴ +11=1193051에 해당함)
HLHLHLHL	32비트의 부호없는 값을 16진수로 표현한 값	1234567C (십진수로는 1*2 ²⁸ +2*2 ²⁴ +3*2 ²⁰ +4*2 ¹⁶ +5*2 ¹² +6*2 ⁸ +7*2 ⁴ +12=305419900에 해당함)
LB	8비트의 부호없는 값을 한개의 바이트로 표현한 값	[0x0F] (여기서 [0x0F]는 00001111b 한바이트를 나타내며 십진수 로 15에 해당함)
HBLB	16비트의 부호없는 값을 두개의 바이트로 표현한 값으로 서 HB가 상위바이트임	[0x01][0x81] (십진수로는 1*256+128+1=385에 해당함)
HBMBLB	24비트의 부호없는 값을 3개의 바이트로 표현한 값	[0x01][0x02][0x03] (십진수로는 1*65536+2*256+128+3=66051에 해 당함)
HbMbLbHbMb Lb	12비트의 부호없는 값 2개를 총 3개의 바이트로 표현한 값	[0x51][0x42][0x03] (십진수로는 5*256+1*16+4=1300 과 2*256+0*16+3=515 두개의 숫자를 의미함)

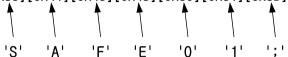
♣ 표에서 HB/HBLB/HBMBLB는 8/16/24비트의 값을 각각 1/2/3개의 바이트로 직접 표현한 것이며, HbMbLbHbMbLb는 12비트의 수 2개를 3개의 바이트로 표현한 것이고, 나머지의 경우는 십진수 또는 16진수로 변환한 값의 각 자리를 0/1/2/3/4/5/6/7/8/9/A/B/C/D/E/F를 나타내는 ASCII code인 [0x30]/[0x31]/[0x32]/[0x33]/[0x34]/[0x35]/[0x36]/[0x37]/[0x38]/[0x39]/[0x41]/[0x42]/[0x43]/[0x44]/[0x45]/[0x46]으로 표현한 것입니다.

예를 들어서 Swd4dd1,dAdd2; 라는 형식으로 기술되어 있는 명령어에 대하여, 파라메터값 dddd1, dddd2의 값으로 100과 120을 사용한다면 명령어가 Sw100,120;이 되는데, 이때 Sw100,120;는 다음 10개의 바이트로 표현되며 맨 앞에 있는 바이트부터 모터제어기로 전송되어야 합니다S' 'w' '1' '0' '0' ',' '1' '2' '0' ';' [0x53][0x77][0x31][0x30][0x30][0x30][0x32][0x30][0x38]



예를 들어서 SA<mark>HLHL</mark>; 라는 형식으로 기술되어 있는 명령어에 대하여. 파라메터값 HLHL의 값으로 FE01을 사용한다면 명령어가 SAFE01;이 되 는데, 이때 SAFE01;는 다음 7개의 바이트로 표현되며 맨 앞에 있는 바이트부터 모터제어기로 전송되어야 합니다.

[0x53][0x41][0x46][0x45][0x30][0x31][0x3B]



예를 들어서 PaHBMBLBHBMBLBCS; 라는 형식으로 기술되어 있는 명령어에 대하여. 파라메터값 HBMBLB의 값으로 0x123456 과 0xEDF018를 사용 한다면 명령어가 Pa[0x12][0x34][0x56][0xED][0xF0][0x18];이 되는데, 이때 Pa[0x12][0x34][0x56][0xED][0xF0][0x18];는 다음 9개의 바이트 로 표현되며 맨 앞에 있는 바이트부터 모터제어기로 전송되어야 합니다.

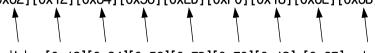
[0x50][0x61][0x12][0x34][0x56][0xED][0xF0][0x18][0x3B]



[0x12][0x34][0x56][0xED][0xF0][0x18] 'P'

예를 들어서 bHbMbLbHbMbLbHbMbLbHbMbLbCS; 라는 형식으로 기술되어 있는 명령어에 대하여. 파라메터값 HbMbLbHbMbLbHbMbLbHbMbLbHb 값으로 0x123. 0x456. 0xEDF 과 0x018를 사용한다면 명령어가 b[0x12][0x34][0x56][0xED][0xF0][0xf0][0x6E];이 되는데. [0x6E]는 앞선 6바이트의 합을 inverse한 값(checksum)으로서, b[0x12][0x34][0x56][0xED][0xF0][0x18][0x6E];는 다음 9개의 바이트로 표현되며 맨 앞에 있는 바이트 부터 모터제어기로 전송되어야 합니다.

[0x62][0x12][0x34][0x56][0xED][0xF0][0x18][0x6E][0x3B]



[0x12][0x34][0x56][0xED][0xF0][0x18] [0x6E]

♣ 명령마다 리턴값의 형식이 일부 다를수 있으므로 개별적인 명령어의 설명을 참조하기 바랍니다.

※

읽기 전에

♣ 명령어를 사용할때 ?를 사용하여 현재의 값이 얼마로 되어 있는지를 확인한 다음, 필요하다면 수정된 값을 명령어에 붙여서 다시 보내는 형식으로 사용하면 실수를 줄일수 있습니다.

예로서 Sw?; 를 모터제어기로 전송하면 제어에 사용하는 전류의 최대치를 리턴값으로 받아 볼수 있는데, 그 리턴값이 Sw800,900; 이라면 모터1의 제어전류 최대치가 800, 모터2의 제어전류 최대치가 900 이라는 뜻입니다. 그 값을 1000, 1000으로 바꾸고자 하면 Sw1000.1000;을 모터제어기로 보내면 됩니다.

♣ 당사에서는 CUBEMON 이라고 하는 PC utility를 제공하고 있습니다. 이를 사용하면 terminal 창을 이용하여 모터제어기로 보낼 명령을 입력하고 이에 따른 리턴값을 표시해주는 기능을 사용할수 있습니다. 또한 여러 종류로 설정된 4가지의 변수값을 1ms 단위로 실시간모니터해볼수 있는 기능도 이용할 수 있습니다.

♣ 본 설명서중에 RJM VERx 라는 명칭이 사용되고 있는데. 그 의미는 다음과 같습니다.

RJM_VER1 또는 RJM_VER2 또는 RJM_VER3 : 현재는 생산하고 있지 않은 초기 H/W 버젼의 DC 모터제어기를 가리킴.

RJM VER4 : 25W ~ 100W 급의 DC 모터제어기를 가리킴.

RJM_VER8 : 200W 급 이상의 DC 모터제어기를 가리킴.

RJM_VER5 : 2상 STEPPING 모터제어기를 가리킴.

RJM_VER6 또는 RJM_VER7_3PHASE_STEP: 3상 STEPPING 모터제어기를 가리킴.

RJM VER7 5PHASE STEP: 3상 STEPPING 모터제어기를 가리킴.

RJM_VER7 : BLDC 또는 AC 서보 모터제어기를 가리킴.



RS485 통신을 이용한 다중 마이컴 통신의 주의사항

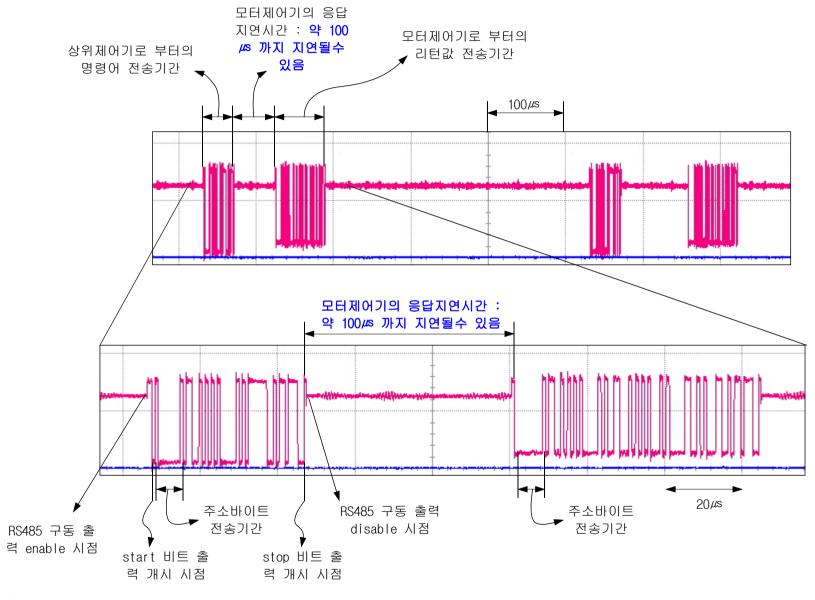
♣ TI사의 DSP제어기는 비동기통신포트(SCI)를 이용한 다중마이컴 통신을 위하여 다음과 같은 2가지 방식을 제공하고 있읍니다.

	Idle-Line Multiprocessor 모드	Address-Bit Multiprocessor 모드
1-byte format	Start LSB 2 3 4 5 6 7 MSB Parity Stop RS232 통신과 같이 start-bit/stop-bit/parity-bit가 붙은 8-bit 형식	Address IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
1-byte 값이 주소인지 데 이터인지를 구별하는 방법	이전 바이트와 현재 수신된 바이트 사이의 시간간격 이 10-bit 시간 이내이면 데이터로 판단하고, 10-bit 시간 이상이면 주소로 판단함.	현재 수신된 바이트의 9th bit가 1 이냐 0이냐에 따라서 주소인지, 데이터인지를 판단.
PC등 기존 기기들과의 호 환성	바이트와 바이트사이의 시간간격을 지킬수 있다면 호환성 있음.	호환성 없음.
로보큐브테크 채택방식	0	X

- ♣ 로보큐브테크는 기존 기기들과의 호환성 유지를 위하여 TI사의 DSP제어기의 Idle—Line Multiprocessor 모드를 채택하고 있습니다. 그러나 다중 제어기간의 통신에는 다음과 같은 주의가 필요합니다.
- ♣ 로보큐브테크 제어기 사용자의 RS485 통신시의 준수사항.
 - : 주소가 같은 데이터 블록안에 있는 데이터를 전송할 때는 바이트와 바이트 사이의 시간 간격이 10-bit 전송시간 이내가 되도록 하여야 합니다.
 - : 이를 준수하려면 주소가 같은 데이터 블록 안에 있는 데이터는 한꺼번에 준비하여야하며, 데이터블록안의 첫바이트 전송이 개시되면 나머지 데이터는 interrupt routine을 이용하여 background에서 전송하도록 하는 것이 필요합니다. (interrupt가 아니고 프로그램이 일일이 관여하여 데이터를 보내게 되면 통신이 제대로 안되거나 cpu time이 통신에 100% 점유되는 상태가 될수 있습니다.)
 - : 만약 바이트와 바이트 사이의 시간 간격이 클수 밖에 없는 경우라면 통신속도를 떨어뜨려 사용하는 것도 한가지 방법입니다. (예를 들면 1.25 Mbps의 경우 10-bit의 시간은 8uS 인데 비하여 9600 bps 인 경우는 10-bit의 시간이 1.04ms 로 매우 커집니다.)
 - : 같은 이유로 주소가 다른 데이터 블록 사이에는 데이터 블록과 데이터 블록 사이의 시간 간격이 10-bit 전송시간 이상이 되도록 하여야 합니다.

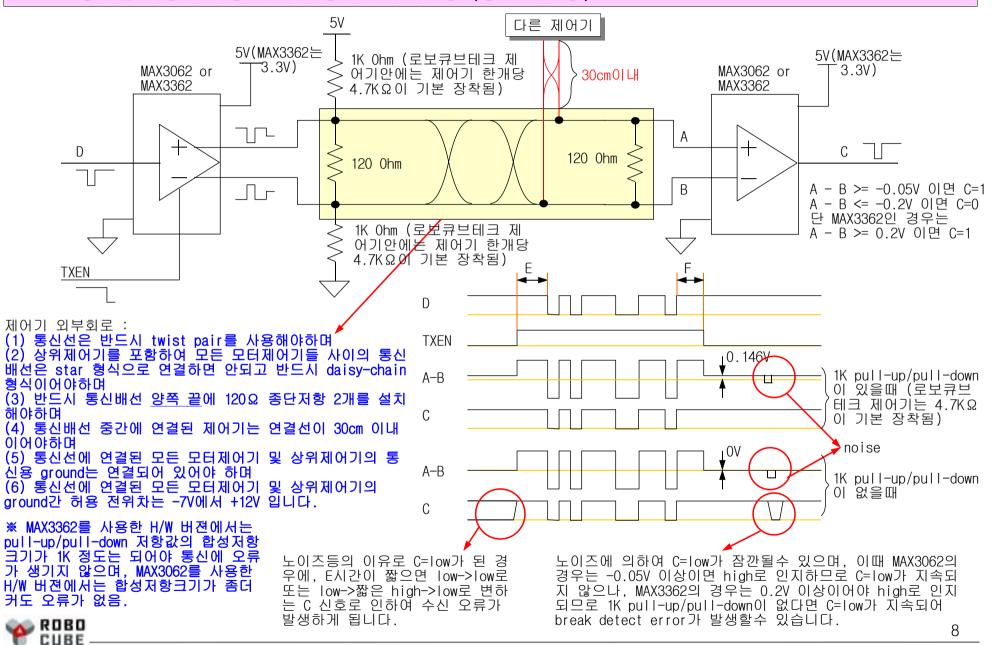


RS485 통신을 이용한 다중 마이컴 통신의 주의사항 (1.25Mbps RS485 출력 +단자 파형)

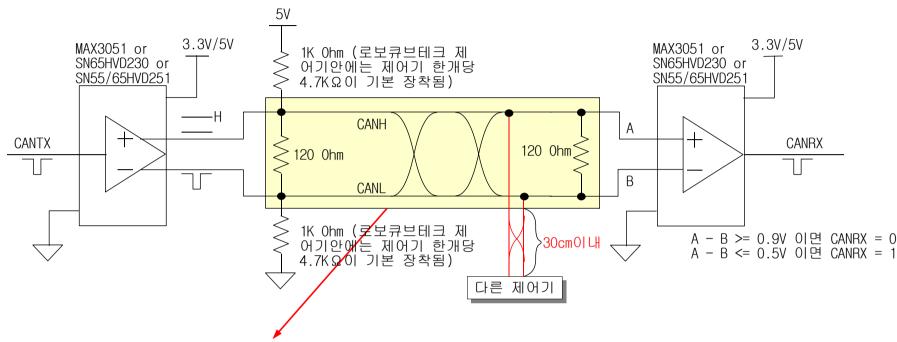




RS485 통신을 이용한 다중 마이컴 통신의 주의사항 (통신 배선상)



CAN 통신을 이용한 다중 마이컴 통신의 주의사항 (통신 배선상)



제어기 외부회로 :

(1) 통신선은 반드시 twist pair를 사용해야하며

(2) 상위제어기를 포함하여 모든 모터제어기들 사이의 통신배선은 star 형식으로 연결하면 안되고 반드시 daisy-chain형식이어야하며 (3) 반드시 통신배선 양쪽 끝에 120Ω 종단저항 2개를 설치해야하□

(3) 반드시 통신배선 <u>양쪽 끝</u>에 120Ω 종단저항 2개를 설치해야하며 (4) 통신배선 중간에 연결된 제어기는 연결선이 30cm 이내이어야하며

(5) 통진전에 연결된 모든 모터제어기 및 상위제어기의 통신용 ground는 연결되어 있어야 하며

(6) 통신선에 연결된 모든 모터제어기 및 상위제어기의 ground간 허용 전위차는 MAX3051인 경우 -7V에서 +12V이고, SN65HVD230의 경우 -7V에서 +16V이고, SN55/65HVD251의 경우는 -36V에서 +36V입니다. 순간 혀용 전위차는 SN65HVD230/SN65HVD251/SN65HVD251이 각각 ± 25V/±200V/±200V 입니다.

(7) 통신선에 연결된 모든 모터제어기 및 상위제어기의 ground간 전위차로서 통신가능한 전위차 범위는 MAX3051와 SN55/65HVD251 인 경우에 -7V에서 +12V이고, SN65HVD230 경우는 -2V에서 +7V 범위 입니다.

CANTX=0일 때 A - B >= 0.9V, CANRX = 0

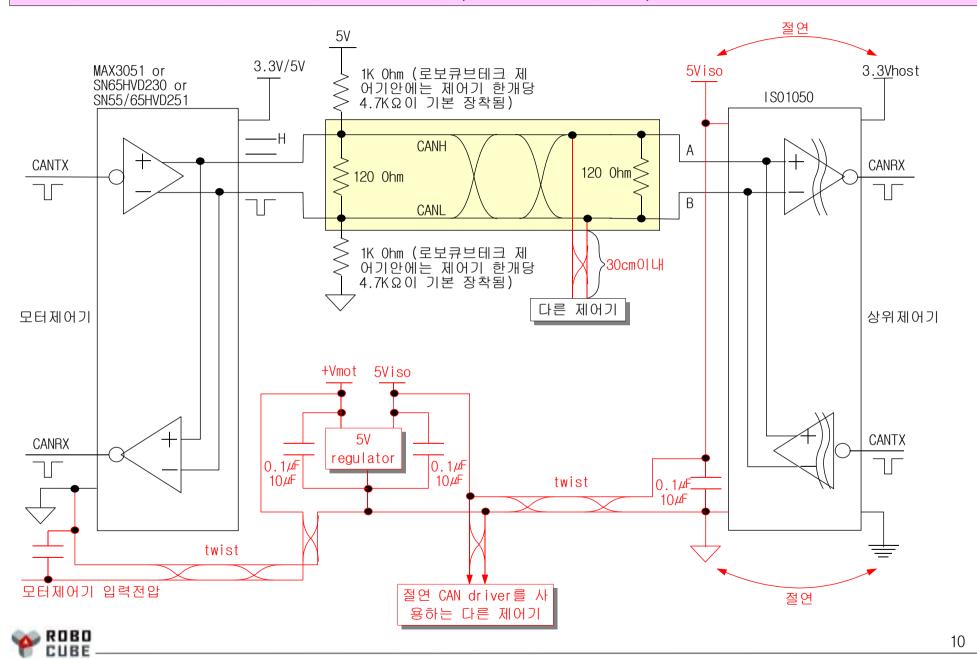
CANTX=1일 때

1K pull-up/pull-down이 없으면 A-B=0V<=0.5V => CANRX=1
1K pull-up/pull-down이 있으면 A-B=0.096V<=0.5V => CANRX=1
(1K pull-up/pull-down 이 있는 경우 noise margin이 0.096V
감소함)

※ CANH, CANL 출력단에 설치된 4.7KΩ pull-up/pull-down 저항은 RS485 통신을 할때는 통신상의 단말기가 모두 off되어 있는 경우 수신단의 H상태를 유지하기 위하여 반드시 필요하나, CAN 통신에서는 통신선상의 단말기가 많을 경우에 잘못된 값을 수신할 가능성이 존재하므로 이때는 좀더 큰 저항값을 사용하여야 함.



CAN 통신을 이용한 다중 마이컴 통신의 주의사항 (isolator를 사용할때)



명령어의 분류

분류	설명	페이지	비고
통신관련 파라메터의 설정	통신관련 파라메터의 설정 - 포트 및 주소	12	
	통신관련 파라메터의 설정 - 에러 및 복구	13	
모터의 기본 파라메터 설정과 저장	모터의 기본 파라메터 설정과 저장	14	
factory default 파라메터의 설정	factory default 파라메터의 설정	15	
	모터제어 파라메터의 설정 - 전류/속도/위치 게인	16	
	제어 파라메터의 설정 - BLDC 모터 추가게인 설정	17	
		18	
모터제어 파라메터의 설정	제어 파라메터의 설정 - 기타게인 설정	19/20/21	
그녀세이 퍼디메디크 글이	모터제어 파라메터의 설정 - 정지강성, 한계값, dead-time 및 펄스주기 속도계산	22	
	모터제어 파라메터의 설정 - compensator 및 filter	23	
	모터제어 파라메터의 설정 - 기타 파라메터1	24	
	모터제어 파라메터의 설정 - 기타 파라메터2	25	
작동개시, 작동중지 및 비상정지, 강제정지	작동개시, 작동중지 및 비상정지 관련 명령어	26	
75/1/N, 756/N & 066/N, 6/1/6/N	강제정지 명령어	27	
	전류구동 명령어	28	
	속도구동 명령어	29	
	point-to-point 변위구동 명령어	30	
	연속 point-to-point 변위구동 명령어	31	
모터제어 작동 실행	실시간 연속 변위구동 명령어	32	
	스테핑모터 구동관련 명령어	33	
	jog 작동 명령어	34	
	homing 작동 명령어	35	
	sequence 제어 명령어	36	
입출력 조작	출력 강제작동 명령어	37	
	입력 모니터 명령어	38	
초기화	초기화 명령어	39	
	과전압, 저전압 및 과전류 보호 명령어	40	
Fault 처리	과온도 보호 명령어	41	
	fault상태 모니터 명령어	42	
	제어작동상태 모니터 명령어	43	
제어작동상태 모니터	제어작동상태 변수 모니터 명령어	44	
M076841 144	제어작동상태 레지스터 및 메모리 읽기/쓰기 명령어	45	
	전류측정 명령어	46	
test 구동	test 구동 명령어	47	
튜닝	튜닝 명령어	48	
	튜닝 명령어 - analog type 센서 사용 BLDC모터의 자동 파라메터 설정	49	
text 입력/저장과 LCD 표시장치	text 입력/저장과 LCD 표시장치 관련 명령어	50	
데모작동	데모작동 명령어	51	
특정 목적의 기능	특정 목적의 기능관련 명령어	52	
F/T 센서	F/T 센서 명령어 - 1/2/3/4	53/54/55/56	
사용하지 않는 명령어	사용하지 않는 명령어	57	11
CUBE			

명령어의 분류 (통신관련 파라메터의 설정 - 포트 및 주소)

적용 CPU	명령어	return value	operation
+#^%	;SBIIII1;	SBIIII1; (SB?; 인 경우에만 echo 있음)	RS232C 직렬통신속도값을 설정함.
+#^%	;SB?;	35111177 (351.7 E 3 1 3 II 2 3 3 II 2 3 II 3 II 3 II 3 II	nozozo Teol Take eol.
+^%	;SbIIII1;	SbIIII1; (Sb?; 인 경우에만 echo 있음)	RS485 직렬통신속도값을 설정함.
+^%	;Sb?;	00111117 (00.7 E 0 1 011E 00110 ME)	10100 1232 1282 231.
#	;Sb1111,11112;	SbIIII1,IIII2; (Sb?; 인 경우에만 echo	RS485 와 RS232C extra 직렬통신속도값을 설정함.
#	;Sb?;	있음)	110-00 A 1102020 OXII u TESETENE ESS.
+#^	;Zbdddd1;	Zbdddd1;	CAN 통신속도를 설정함.
+#^	;Zb?;	ZDdddd 17	WW 80712 280.
+#^% +#^%	;SXHLHL; ;SX?;	SXHLHL; 또는 SXHLHLHLHL; 2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
+#^ +#^	;SXHLHLHLHL; ;SX?;	SXHLHLHLHL; 2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
+#^	;SXHLHLHLHL,HLHLHLHL; ;SX?;	SXHLHLHLHL, HLHLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
1.4400/	· CALII III ·		
+#^%	;SAHLHL;	SAHLHL;	모터제어기의 주소를 설정함.
+#^%	;SA?;		머러이에 대한 프로그르그는 보내 때 기오랜드 보고 오 포츠
+#^%	;SmHLHL;	SmHLHL;	명령어에 대한 return값을 보낼 때 사용하는 host의 주소
+#^%	;Sm?;		를 설정함.



명령어의 분류 (통신관련 파라메터의 설정 - 에러 및 복구)

적용 CPU	명령어	return value	operation
+#^%	;Stdddd1,dddd2,dddd3;	C+ dddd	PP/Pp/PQ/Pq 명령어의 update 주기와 PP/Pp/PQ/Pq 명령어
+#^%	;St?;	Stdddd1,dddd2,dddd3;	미수신의 연속허용횟수를 설정함.
+#^%	;QE;	QEHLHL,HLHL,HLHL,HLHL; (F2406인경우), QEHLHL,HLHL,HLHL,HLHL,HLHL; (F2811/F28334인 경우)	통신에러횟수를 읽어서 host로 전송함.
+#^	;QEE;	QEEHLHL, HLHL, HLHL, HLHL;	통신에러횟수를 읽어서 host로 전송함.
+#^%	;EC?;	ECHLHL,HLHL; (LF2406A의 경우)	RS232, RS485, CAN 통신과정에서 어떤 에러가 발생하였는
T#* %		ECHLHL, HLHL, HLHL; (F2811/F28334의 경우)	지를 읽어서 host로 전송함.
+#^%	;ECR;	ECR;	RS232, RS485, CAN 통신에러 플래그를 없애고 에러를 복구함.
+#^	;EcS;	EcS;	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전 송되어오는 broadcasting 데이터를 수신하도록 설정함.
+#^	;EcR;	EcR;	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전송되어오는 broadcasting 데이터를 수신하지 못하도록 설정함.



명령어의 분류 (모터의 기본 파라메터 설정과 저장)

적용 CPU	명령어	return value	operation
+#^%	;STHLHL;	STHLHL;	모터의 기본파라메터를 쉽게 설정하기 위하여 미리 준비되
+#^%	;ST?;	SINLINL,	어있는 motor type을 선정함.
+#^%	;SEA55A,HLHL,HLHL,HLHL,HL		
	HL;	SEHLHL, HLHL, HLHL, HLHL;	모터 1회전당 펄스수와 POLE PAIR 갯수를 설정함.
+#^%	;SE?;		
+#^	;XEdddd1,dddd2; or	XEdddd1,dddd2;	스테핑모터 또는 절대각엔코더를 사용하는 경우 엔코더의
+#^	;XE 1, 2;	, and the second	
+#^	;XE?;	or XEIIII1, IIII2;	4체배된 펄스수를 설정함.
+#^%	;SGA55A,dddd1,dddd2,dddd3		
	,dddd4;	SGdddd1,dddd2,dddd3,dddd4;	감속비와 최대회전속도를 설정함.
+#^%	;SG?;		
+#^% +#^%	;SXHLHL; ;SX?;	SXHLHL; 또는 SXHLHLHLHL; 2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
+#^ +#^	;SXHLHLHLHL; ;SX?;	SXHLHLHLHL; 2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
+#^ +#^	;SXHLHLHLHL,HLHLHLHL; ;SX?;	SXHLHLHLHL, HLHLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
+#^%	;EDA55A;	EDA55A;	DSP SRAM에 있는 파라메터 값을 default 값으로 바꿈. 이때 ST 명령어로 선정된 motor type에 따라서 파라메터의 기본값이 설정됨.
+#^%	;EsA55A;	EsA55A;	DSP SRAM에 있는 작동 파라메터 값을 한꺼번에 EEPROM에 저장함.



명령어의 분류 (factory default 파라메터의 설정)

적용 CPU	명령어	return value	operation
+#^	2012년 12월 19일 이전 버젼인 경우 ;Z0iiii1; ~ ZFiiii1; (단 Z3HLHL; Z4HLHL; Z5HLHL; Z6HLHL; Z7HLHL; Z8HLHL; Z9HLHL; ZAHLHL; ZBHLHL; ZCHLHL; 는 예외) (+#^) 2012년 12월 19일 S/W 버젼부터 ;Z0A55A,iiii1; ~ ZFA55A,iiii1; (단 Z3A55A,HLHL; Z4A55A,HLHL; Z5A55A,HLHL; Z6A55A,HLHL; Z7A55A,HLHL; Z8A55A,HLHL; Z9A55A,HLHL; ZAA55A,HLHL; ZBA55A,HLHL;	ZO(~F)HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH	0 ~ F 까지의 숫자로 지정된 위치의 FACTORY SETTING parameter RAM에 값 을 저장함.
+#^	;Zr;	Zrhlhl, Hlhl, Hlhl	16개의 FACTORY SETTING parameter 값 을 읽어서 보여줌 (RAM 영역의 값).
+#^	;ZsA55A;	ZsA55A;	메모리상의 FACTORY SETTING parameter 를 EEPROM으로 저장함.



명령어의 분류 (모터제어 파라메터의 설정 - 전류/속도/위치 게인)

적용 CPU	명령어	return value	operation
+#^%	;SQ또는		
	SqHLHLHLHL, HLHLHLHL, HLHLH	SQ또는	모터1의 파라메터 리밋과 위치/속도/전류 제어파라메터를
	LHL,HLHL,HLHLHL,HLHL,HL	SqHLHLHLHL, HLHLHLHL, HLHLHLHL, HLHL, HLHLH	설정함. (WQ 명령어로 속도 및 전류제어 파라메터를 설정
	HL, HLHLHLHL, HLHL, HLHL, HLH	LHL, HLHL, HLHL, HLHLHLHL, HLHL, HLHLHL	하는 경우는 제외)
	LHLHL;	HL;	
+#^%	;SQ?; 또는 ;Sq?;		
+#^%	;SR또는		
	SpHLHLHLHL, HLHLHLHL, HLHLH	SR또는	모터2의 파라메터 리밋과 위치/속도/전류 제어파라메터를
	LHL, HLHL, HLHLHLHL, HLHL, HL	SPHLHLHLHL, HLHLHLL, HLHLHLL, HLHL, HLHLH	설정함. (WR 명령어로 속도 및 전류제어 파라메터를 설정
	HL, HLHLHLHL, HLHL, HLHL, HLH	LHL, HLHL, HLHL, HLHLHLHL, HLHL, HLHLHL	하는 경우는 제외)
	LHLHL;	HL;	
+#^%	;SR?; 또는 ;Sp?;		
+#^	;AQ또는	AQ또는	
	AqHLHL, HLHLHLHL, HLHL, HLH	AGHLHL, HLHLHLHL, HLHL, HLHL, HLHLHLHL, HLHL	모터1의 위치/속도 제어파라메터를 설정함. (WQ 명령어로
	L,HLHLHLHL,HLHL;	:	속도제어 파라메터를 설정하는 경우는 제외)
+#^	;AQ?; 또는 ;Aq?;	,	
+#^	;AR또는	AR또는	
	Arhlhl, Hlhlhlhl, Hlhl, Hlh	Arhlhl, Hlhlhlhl, Hlhl, Hlhl, Hlhlhlhl, Hlhl	모터2의 위치/속도 제어파라메터를 설정함. (WR 명령어로
	L,HLHLHLHL,HLHL;		속도제어 파라메터를 설정하는 경우는 제외)
+#^	;AR?; 또는 ;Ar?;	,	
+#^	;AS또는		
	ASHLHL, HLHLHLHL, HLHL, HLHL	AS또는AsHLHL,HLHLHLL,HLHL,HLHLHL;	모터1,2의 전류 제어파라메터를 설정함. (WQ, WR 명령어로 전류제어 파라메터를 설정하는 경우는 제외)
	HLHL;		
+#^	;AS?; 또는 ;As?;		



명령어의 분류 (제어 파라메터의 설정 - BLDC 모터 추가게인 설정)

적용 CPU	명령어	return value	operation
+#^	;WQdddd1,dddd2,dddd3,dddd		모터1의 03/06번 위치/속도제어모드에서 속도게인과 ,
	4,dddd5;	WQdddd1,dddd2,dddd3,dddd4,dddd5;	RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터1의 전류제어
+#^	;WQ?;		파라메터를 설정함.
+#^	;WRdddd1,dddd2,dddd3,dddd		모터2의 03/06번 위치/속도제어모드에서 속도게인과 ,
	4,dddd5;	WRdddd1,dddd2,dddd3,dddd4,dddd5;	RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터2의 전류제어
+#^	;WR?;		파라메터를 설정함.
			BLDC 모터1,2에 있어서 Hall_A상 상승에지에 대한 U상 전
+#^	;WEdddd1,dddd2;	MEGGGGT GGGG GGGG GGGG	압 positive zero crossing 포인트의 위상지연 값을 설정
+#^	;WE?;	WEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	하며, 모터1,2의 POLE_PAIR 수, 모터1,2의
			MOTORx_POSITION_SCALE_FACTOR를 읽어서 host로 전송함.
	Wafer control이 아닌 BLDC		
	모터의 경우 (2012년 2월		
	18일 S/W 버젼부터 적용) :	Shdddd1,dddd2,dddd3,dddd4;	BLDC모터에서 holding torque를 만드는 파라메터를 설정
+#^	;Shdddd1,dddd2,dddd3,dddd	Sildada i, adduz, addus, addu4,	함.
	4;		
+#^	;Sh?;		
+#^	;Gtdddd1,dddd2;	C+dddd1 dddd2:	RJM_VER7의 Hall 센서입력의 ON/OFF를 검출하는 기준전압
+#^	;Gt?;	Gtdddd1,dddd2;	과 히스테리시스 진폭을 설정함.



명령어의 분류 (제어 파라메터의 설정 - 게인 스케쥴링 파라메터 설정)

적용 CPU	명령어	return value	operation
+#^	;AX or		
	AxA55A,dddd1,dddd2,dddd3,	AX or	
	dddd4,dddd5,dddd6,dddd7,d	AxA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dd	모터1의 상위 or 하위 속도에서 제어 게인을 높이거나 낮
	ddd8,dddd9,dddd10,dddd11,	dd6,dddd7,dddd8,dddd9,dddd10,dddd11,ddd	추는 파라메터를 설정함.
	dddd12,dddd13,dddd14;	d12,dddd13,dddd14;	
+#^	;AX or Ax?;		
+#^	;AY or		
	AyA55A, dddd1, dddd2, dddd3,	AY or	
	dddd4,dddd5,dddd6,dddd7,d	AyA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dd	모터2의 상위 or 하위 속도에서 제어 게인을 높이거나 낮
	ddd8,dddd9,dddd10,dddd11,	dd6,dddd7,dddd8,dddd9,dddd10,dddd11,ddd	추는 파라메터를 설정함.
	dddd12,dddd13,dddd14;	d12,dddd13,dddd14;	
+#^	;AY or Ay?;		
+#^	;AZA55A,dddd1,dddd2;	AZdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	SQ/SR/Sq/Sr/AQ/AR/Aq/Ar/WQ/WR 등의 명령으로 설정하는
+#^	;AZ?;	nzada i, addaz, adda , adda i, adda , adda ,	게인이 적용되는 중간 속도값을 설정함.
+#^	;AzA55A;	AzA55A;	AX/Ax/AY/Ay/AZ 명령에 의한 설정값을 EEPROM에 저장함.



명령어의 분류 (제어 파라메터의 설정 - 기타 게인 설정1)

적용 CPU	명령어	return value	operation
+#^ +#^	;WDA55A,dddd1,dddd2;;;WD?;	WDdddd1,dddd2;	미분값 계산을 실시하는 시간차를 설정함.
+#^ +#^	;Wddddd1,dddd2,dddd3,dddd 4,dddd5,dddd6; ;Wd?;	Wddddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	inner 속도제어루프가 있는 03번 위치제어모드의 속도제어 또는 Digital Hall sensor만을 사용하는 BLDC모터의 위치/ 속도 제어 또는 0A번 속도제어모드 또는 속도제어로서 SX 명령어로 설정하는 두번째 파라메터의 bit11이 0인 경우에 , 낮은 속도에서 위치/속도 제어게인(3번 위치제어모드의 경우는 속도게인만)을 낮게하는 파라메터를 설정함.
+#^ +#^	;WSdddd1,dddd2; ;WS?;	WSdddd1,dddd2;	위치제어모드에서 속도에 비례하는 damping 회전력을 발생 시키는 게인을 설정함.
+#^ +#^	;Wsdddd1,dddd2; ;Ws?;	Wsdddd1,dddd2;	DD motor 등에서 속도에 비례하는 damping current를 발생 시키는 게인을 설정함.
+#^ +#^	;Sfdddd1,dddd2; ;Sf?;	Sfdddd1,dddd2;	위치제어의 feedforward 게인을 설정함.



명령어의 분류 (제어 파라메터의 설정 - 기타 게인 설정2)

적용	를 CPU	명령어	return value	operation
+#^		BLDC 모터의 경우 ;MBdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터1의 경우, Hall 센서 A/B/C 상신호의 에지위치를 설정함.
+#^	\	BLDC 모터의 경우 ;Mb?;	Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어기로 전 송함.
+#^		BLDC 모터의 경우 ;MEdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6; ;ME?;	<pre>MEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;</pre>	엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터2의 경우, Hall 센서 A/B/C 상신호의 에지위치를 설정함.
+#^	Λ.	BLDC 모터의 경우 ;Me?;	Medddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall 센서를 사용하지 않는 BLDC 모터2의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어기로 전 송함.



명령어의 분류 (제어 파라메터의 설정 - 기타 게인 설정3)

적용 CPU	명령어	return value	operation
+#^	low resolution encoder의 DC 모터인 경우 ;MBdddd1,dddd2,dddd3,dddd 4; ;MB?;	MBdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터1의 경우, 엔 코더신호 A/B 상신호의 에지위치를 설정함.
+#^	low resolution encoder의 DC 모터인 경우 ;Mb?;	Mbdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터1의 경우 엔코 더신호의 에지간 간격값을 읽어서 상위제어기로 전송함.
+#^	low resolution encoder의 DC 모터인 경우 ;MEdddd1,dddd2,dddd3,dddd 4; ;ME?;	MEdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터2의 경우, 엔 코더신호 A/B 상신호의 에지위치를 설정함.
+#^	low resolution encoder의 DC 모터인 경우 ;Me?;	Medddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터2의 경우 엔코 더신호의 에지간 간격값을 읽어서 상위제어기로 전송함.
+#^ +#^	low resolution encoder의 DC 모터 또는 BLDC 모터인 경우 ;MFdddd1,dddd2; ;MF?;	MFdddd1,dddd2;	low resolution encoder를 사용하는 DC 모터 또는 BLDC 모터인 경우에 속도신호의 변화를 제한하는 범위를 설정함.



명령어의 분류 (제어 파라메터의 설정 - 정지강성, 한계값, dead-time 및 펄스주기 속도계산)

적용 CPU	명령어	return value	operation
+#^	;Sudddd1,dddd2;	Sudddd1,dddd2;	위치제어에서 정지위치 근방에서 강성을 높이기 위하여 제
+#^	;Su?;	Suddad 1, dddd2,	어 게인을 높게 하는 비율을 설정함.
+#^	;ABA55A,dddd1;		위치제어에서 정지하고 있을때, jitter와 소음을 줄이거나
+#^	;AB?;	ABdddd1;	또는 강성을 높이기 위하여, 위치제어 게인을 낮추거나 높
'''	,,,,,,		이기 위한 파라메터를 설정함.
+#^%	;SL11111,11112,11113,11114;	SLIIII1, IIII2, IIII3, IIII4;	위치제어모드에서 변위의 한계값을 설정함.
+#^%	;SL?;	02	יייי איייייייייייייייייייייייייייייייי
+#^%	;Swdddd1,dddd2;	Swdddd1,dddd2;	제어시 사용하는 전류의 리밋을 설정함.
+#^%	;Sw?;	onada i, adda_i	ANA ARREST STATES OF THE STATE
+#^	;Szdddd1,dddd2;	Szdddd1,dddd2;	제어시 사용하는 전류의 리밋을 0.01A 단위로 설정함.
+#^	;Sz?;	OZdada i, dadaz i	MON NOOL EHA GYE 0.01/ ENT EST.
+#^	;MRdddd1,dddd2,dddd3,dddd4;	MRdddd1,dddd2,dddd3,dddd4;	전류상승률을 제한하는 파라메터를 설정함.
+#^	;MR?;	mitada i, addaz, adda , adda ;	단유용용을 제한이는 피디메디를 들었다.
+#^	;AwA55A,dddd1,dddd2;	Awdddd1,dddd2;	사용가능한 최대크기의 PWM duty 비를 설정함.
+#^	;Aw?;	/ Waddi, adda_,	7,67,82 Adi 27,4 Tim dat, 172 201.
+#^	;ADA55A,iiii1,iiii2;		MOTOR 구동 Half-bridge의 dead-time을 보상하되, Z6 명령
+#^	;AD?;	ADiiii1,iiii2;	어로 설정하는 dead-time에 추가하여 보상하려는 보상량을
• 11			설정함.
+	;Acdddd1,dddd2;	Acdddd1,dddd2;	엔코더펄스 카운트 방식으로 속도를 계산할 때 moving
+	;Ac?;	//odddd i, dddd2 i	average 구간의 크기를 설정함.
			엔코더펄스 카운트 방식으로 속도를 계산할 때 moving
#	;Acdddd1,dddd2,dddd3,dddd4;	Acdddd1,dddd2,dddd3,dddd4;	average 구간의 크기와, 엔코더펄스의 주기를 사용하여 속
#	;Ac?;	, loadad i , addaz , adduo , adda ;	도를 계산할때 주기계산에 사용되는 엔코더펄스개수의 2에
			대한 지수를 설정함.



명령어의 분류 (제어 파라메터의 설정 - compensator 및 filter)

적용 CPU	명령어	return value	operation
+#^	;Sgiiii1,iiii2,iiii3,iiii		위치제어 또는 속도제어에서 phase lead 또는 phase lag
	4;	Sgiiii1,iiii2,iiii3,iiii4;	compensator의 파라메터를 설정함. (이 경우는 미분제어
+#^	;Sg?;		게인이 무시됨)
+#^%	;SFdddd1,dddd2,dddd3;	SFdddd1,dddd2,dddd3;	butterworth filter의 시정수와 변화율 상한값을 설정함.
+#^%	;SF?dddd1;	Si dada i, addaz, addas,	DULLET WOLLING 다 시승부와 단회을 당한없을 들었다.
#	;ALdddd1,dddd2,dddd3,dddd		
	4;	ALdddd1,dddd2,dddd3,dddd4;	28~31번 butterworth filterA의 시정수를 설정함.
#	AL?;		
+#^	;ANdddd1,dddd2;	ANdddd dddd:	모터1에 적용되는 Notch filter의 중심주파수와 Q factor
+#^	;AN?;	ANdddd1,dddd2;	를 설정함.
+#^	;Andddd1,dddd2;	Andddd1,dddd2;	모터2에 적용되는 Notch filter의 중심주파수와 Q factor
+#^	;An?;	Alladat, addaz,	를 설정함.



명령어의 분류 (제어 파라메터의 설정 - 기타 파라메터1)

적용 CPU	명령어	return value	operation
+#^	;Dddddd1;		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이
+#^	;Dd?;	Dddddd1;	는 모드에서 속도명령값 또는 전류명령값의 deadband 크기
• #	, ou: ,		를 설정함.
+#^	;DOA55A,dddd1,dddd2;		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이
+#^	;D0?;	DOdddd1,dddd2;	는 모드에서 속도명령값 또는 전류명령값이 0일때의 LSxC
T#* *	,00?,		포트 A/D 변환값을 설정함.
#	;MOiiii1,iiii2;		절대각센서를 사용하는 경우에 BLDC 모터의 Hall A/B/C상
#	;MO?;	MOiiii1,iiii2;	신호와 모터위상각신호의 위상차를 없애기 위한 위상
	(절대각센서를 사용하는 경	MOTITIT, TTTL2;	offset을 설정함.
	우)		OTISE(를 설정함.
+#^	• MA dddd - dddd0 - dddd0•		자동화용 H/W 버젼에서 상위명령값과 실제값의 차이가 설
+#^	;MAdddd1,dddd2,dddd3;	MAdddd1,dddd2,dddd3;	정값이내에 들어왔는지를 판단하고 mission 완료를 결정하
T#* *	;MA?;		는 경계값(threshold)을 설정함.
+#^	;WPdddd1,dddd2;	WID 44 44 4440 ·	Digital Hall sensor type의 RJM_VER7 모터에 대하여 정지
+#^	;WP?;	WPdddd1,dddd2;	희망 edge 번호를 지정함.
+#^	·Modddd dddd dddd dddd		Digital Hall sensor type의 RJM_VER7 모터에 대하여 감속
T#* '	;Wpdddd1,dddd2,dddd3,dddd	Wpdddd1,dddd2,dddd3,dddd4,dddd5;	완료후 안정화를 개시하는 지연시간과 안정화용 analog
1.44	4,dddd5;;;Wp?;		Hall sensor의 offset, 및 analog Hall 센서가 사용되는
+#^			위치오차 범위등을 설정함.



명령어의 분류 (제어 파라메터의 설정 - 기타 파라메터2)

적용 CPU	명령어	return value	operation
+#^	;Wciiii1,iiii2;	Wciiii1,iiii2;	유도모터의 경우에 ld 전류의 기준값을 설정함.
+#^	;Wc?;	WG11111,11112,	ㅠ도보다의 경우에 Nu 전류의 기준없을 들었음.
+#^	;Wqdddd1,dddd2;	Wqdddd1,dddd2;	모터1의 MAX_PERCENT_SLIP1, ELECTRIC_TIME_CONSTANT1 값
+#^	;Wq?;	wquada i, addaz,	을 설정함.
+#^	;Wrdddd1,dddd2;	Wrdddd1,dddd2;	모터2의 MAX_PERCENT_SLIP2, ELECTRIC_TIME_CONSTANT2 값
+#^	;Wr?;	widdai, dddd2,	을 설정함.
+#^	;WTdddd1,dddd2,dddd3,dddd		BLDC 모터에 있어서 cogging torque를 보상하는 파라메터
	4;	WTdddd1,dddd2,dddd3,dddd4;	를 설정함.
+#^	;WT?;		206.

명령어의 분류 (작동개시, 중지 및 비상정지 관련 명령어)

적용 CPU	명령어	return value	operation
+#^%	;PEHLHL;		
+#^	;PEA55A;	PEHLHL;	선택된 보드의 모터 출력을 ON 시킴.
+#^%	;PE?;		
+#^%	;SMHLHL;	SMHLHL;	위치, 속도, 전류제어 등의 작동모드를 설정함.
+#^%	;SM?;	OWI ICI IC	기시, 그노, 전파세이 승규 그중도그를 들었다.
+#^	;RBHLHL;	RBHLHL;	feedback 제어를 개시하도록 작동모드 설정에 필요한 조치
+#^	;RB?;	HOI ICI IC	를 자동으로 처리하게 함.
+#^%	;PDHLHL;		
+#^	;PDA55A;	PDHLHL;	선택된 보드의 모터 출력을 OFF 시킴.
+#^	;PD?;		
+#^%	;PPR; or ;PQR;	PPR; or PQR;	모터가 stall된 채로 정지된 상태를 해지함.
+#^%	;PRHLHL;	PRHLHL;	선택된 보드의 fault flag를 clear 시킴.
+#^	;PRA55A;	THE IC	연기단 보드의 Tault Hage Clear 자님.
+#^	;RrA55A;	N.A.	모터제어기를 리셋함.
+#^	;RSHLHL;	RSHLHL;	비상정지시의 정지 형태를 정의함.
+#^	;RS?;	TOTILITE?	000000000000000000000000000000000000000
+#^	;RsA55A;	RsA55A;	모터1과 모터2가 명령에 의한 비상정지를 실행함.
+#^	;RsA5A5;	RsA5A5;	모터1이 명령에 의한 비상정지를 실행함.
+#^	;Rs5A5A;	Rs5A5A;	모터2가 명령에 의한 비상정지를 실행함.
+#^	;Sodddd1,dddd2;		속도제어 및 위치제어 모드에서 리밋센서, 비상정지 명령
+#^	;So?;	Sodddd1,dddd2;	또는 비상정지 상황에 의한 강제 감속시의 감속율을 설정
'π	,00:,		함.



명령어의 분류 (강제정지 명령어)

적용 CPU	명령어	return value	operation
+#^%	;PsA55A;	PsA55A;	모터 1,2를 모두 현재의 위치에서 감속을 실시한 후에 정지함.
+#^	;PtA55A;	PtA55A;	보다 1,2들 보구 현재의 귀시에서 함독을 될지만 후에 정시함.
+#^%	;PsA5A5;	PsA5A5;	모터 1을 현재의 위치에서 감속을 실시한 후에 정지함.
+#^	;PtA5A5;	PtA5A5;	도니 T을 현재의 위치에서 음극을 들시한 후에 경치함.
+#^%	;Ps5A5A;	Ps5A5A;	모터 2를 현재의 위치에서 감속을 실시한 후에 정지함.
+#^	;Pt5A5A;	Pt5A5A;	도니 Z을 단체의 위치에서 음극을 들시된 후에 당시함.
+#^	;PeA55A;	PeA55A;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2 모두를 02번 위치제어모드로 바꾸고, 현재의 위치에 서 감속후 정지함.
+#^	;PeA5A5;	PeA5A5;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을 02번 위치제어모드로 바꾸고, 현재의 위치에서 감속 후 정지함.
+#^	;Pe5A5A;	Pe5A5A;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를 02번 위치제어모드로 바꾸고, 현재의 위치에서 감속 후 정지함.
+#^	;PvA55A;	PvA55A;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2 모두를 감속한 후에, 낮은 속도가되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
+#^	;PvA5A5;	PvA5A5;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을 감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
+#^	;Pv5A5A;	Pv5A5A;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를 감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.



명령어의 분류 (전류구동 명령어)

적용 CPU	명령어	return value	operation
+#^% +#^%	;SCiiii1,iiii2; ;SC?;	SCiiii1,iiii2; SCiiii1,iiii2,iiii3,iiii4;(RJM_VER5 STEP형 만 해당됨) SCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (RJM_VER6 3상STEP형 만 해당됨)	전류제어 모드에서 전류명령값을 설정함.
+#^	;Sciiii1,iiii2; ;Sc?;	Sciiii1,iiii2;	전류제어 모드에서 0.01A 단위로 전류명령값을 설정함.
+#^ +#^	;Pciiii1,iiii2; ;Pc?;	Pciiii1,iiii2;	digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우에, 전류제어 모드에서 전류명령값을 설정하되, 리턴값으로 현재 속도값을 host로 전송함.



명령어의 분류 (속도구동 명령어)

적용 CPU	명령어	return value	operation
+#^	;SVIIII1, IIII2;		
+#^	;SVr11111,11112;	SVIII1, IIII2;	속도제어 모드에서 속도명령값을 설정함.
+#^	;SV?;		
+#^	;PVIIII1,IIII2,dddd3;		속도제어 모드에서 속도명령값을 설정하되 설정값의
+#^	;PVrIIII1,III12,dddd3;	PVIII1, IIII2;	국도제어 모드에서 국도당당없을 결정하고 결정없고 checksum을 사용함.
+#^	;PV?;		CHECKSUM를 사용함.
+#^	;Pv 1, 2;		digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우
+#^	;Pv?;	Pv 1, 2;	에, 속도제어 모드에서 속도명령값을 설정하되, 리턴값으
1#	,FV!,		로 현재 위치값을 host로 전송함.
+#^%	;Sadddd1,dddd2,dddd3,dddd		
	4;	Sadddd1,dddd2,dddd3,dddd4;	속도제어 및 위치제어 모드에서 가속 및 감속율을 설정함.
+#^%	;Sa?;		



명령어의 분류 (point-to-point 변위구동 명령어)

적용 CPU	명령어	return value	operation
+#^%	;PAIIII1,IIII2;	PAIIII1, IIII2;	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
+#^%	;PA?;	FAIIII, 11112,	을 설정함.
+#^%	;PBHLHLHLHL,HLHLHL;	PBHLHLHLHL, HLHLHLHL;	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
+#^%	;PB?;	PONLINCIAL, NUMBER 1	을 설정함.
+#^%	;PaHBMBLBHBMBLBCS;	PaHBMBLBHBMBLBCS;	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
			을 설정함.
+#^	;P111111,11112;	PIIII1, IIII2;	위치 명령값의 증분을 설정하고 이동을 명령함.
+#^	;PI?;	,	
+#^	;P1 1;	P1 1;	모터 1에 대하여 위치 명령값의 증분을 설정하고 이동을
+#^	;P1?;	1111117	명령함.
+#^	;P2 1;	P2 1;	모터 2에 대하여 위치 명령값의 증분을 설정하고 이동을
+#^	;P2?;	FZ	명령함.
+#^%	;SSdddd1,dddd2;	SSdddd1,dddd2;	위치제어모드에서 Point-to-Point 이동시 이동속도값을 설
+#^%	;SS?;	SSdddd1,dddd2,	정함.
+#^%	;Ssdddd1,dddd2;	Codddd dddo.	PA, PB, Pa 명령에 의한 변위구동시의 가속/감속 기간을
+#^%	;Ss?;	Ssdddd1,dddd2;	설정함.
+#^	;Aldddd1,dddd2,dddd3,dddd		PA 또는 SV 명령에 의한 가감속시에 전류명령값의
	4;	Aldddd1,dddd2,dddd3,dddd4;	feedforward 값을 계산하는 scale factor를 설정함.
+#^	;AI?;		Teeurorwaru 없을 게신하는 Scale Tactor을 설정함.
+#^	;Aidddd1,dddd2;	Aiddd dddo:	PA 명령에 의한 가속시에 사다리꼴 모양의 가속 프로파일
+#^	;Ai?;	Aidddd1,dddd2;	을 만들기 위한 가속 변화율값을 설정함.
+#^	;ATdddd1;	ATAMA1.	PA 명령의 완료 시점에서 추가로 실행하는 기능을 위한 시
+#^	;AT?;	ATdddd1;	간파라메터를 설정함.



명령어의 분류 (연속 point-to-point 변위구동 명령어)

적용 CPU	명령어	return value	operation
+#^%	;SPdddd1,	SPdddd1,	좌표점을 정의함.
+#^%	;SP?dddd1;	SPudda1,11112,11113,	
+#^%	;PMdddd1,dddd2,dddd3;	PMdddd1 dddd2 dddd3:	현재위치로부터 dddd1점을 거쳐 dddd2점까지 dddd3의 속도
+#^%	;PM?;		로 이동시킴.
+#^%	;PME;	PME;	PM 에 의한 동작을 멈춤.

명령어의 분류 (실시간 연속 변위구동 명령어)

적용 CPU	명령어	return value	operation
+#^%	;SeHLHL;	SeHLHL;	motor1 과 motor2 에 대하여 PM, PP, PQ, Pp,
+#^%	;Se?;	SCHLIL,	Pq command를 적용가능토록 설정함.
+#^%	;SrHLHL,HLHL;		
+#^%	;SrHLHL,HLHL,HLHL,	SrHLHL, HLHL;	Pp/Pq/p 명령어에 대한 return값의 주소를 설
	HLHL;	SrHLHL, HLHL, HLHL;	정함.
+#^%	;Sr?;		
			모터제어기의 작동모드에 따라서
+#^%	;PP11111,11112;	PPIIII1,IIII2;	10msec/30msec 동안에 가야할 변위/속도/전류
+#^%	;PQ11111,11112;	PQIIII1, IIII2;	의 명령값을 설정하고 모터가 움직이도록 함.
			PP인경우는 10msec, PQ인 경우는 30msec임.
+#^% +#^%	;PpHBMBLBHBMBLBCS; ;PqHBMBLBHBMBLBCS;	2006년 10월 7일 이전 S/W 버젼 PpHBMBLBHBMBLBCS; PqHBMBLBHBMBLBCS; 2006년 10월 7일 S/W 버젼부터 PpLBHBLBLBHBLBHBLBCS; 또는 PpHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능) PqLBHBLBLBHBLBHBLBCS; 또는 PqHBLBHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능)	모터제어기의 작동모드에 따라서 10msec/30msec 동안에 가야할 변위/속도/전류 의 명령값을 설정하고 모터가 움직이도록 함. Pp인경우는 10msec, Pq인 경우는 30msec임.
+#^%	;pHBMBLBHBMBLBCS; (RS485의 경우) pHBMBLBHBMBLBCS (CAN의 경우)	RS485의 경우: pLBHBLBHBLBHBLBCS; 또는 pHBLBHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능) CAN의 경우: LBHBLBHBLBHBLB 또는 HBLBHBLBHBLBHBLB(2007년 9월 19일 S/W 버젼부터 사용가능)	모터제어기의 작동모드에 따라서 10msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터가 움직이도록 함.
+#^%	;PPE; or ;PQE;	PPE; or PQE;	PP, PQ, Pp, Pq 에 의한 동작을 멈춤.



명령어의 분류 (스테핑모터 구동관련 명령어)

적용 CPU	명령어	return value	operation
+#^% +#^%	RJM_VER5/6 STEP형인 경우: ;SHiiii1,iiii2,iiii3,iiii 4,iiii5,iiii6; ;SHiiii1,iiii2,iiii3,iiii 4,iiii5,iiii6,iiii7,iiii8 ; ;SH?;	SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (RJM_VER5 STEP형 만 해당됨) SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6,i iii7,iiii8;(RJM_VER6 3상STEP형 만 해당 됨)	RJM_VER5/6 STEP형의 host direct electrical angle 제어 모드에서 구동전류의 진폭과 전류 OFFSET을 설정함.
+#^%	스테핑모터의 경우 : ;Shdddd1,dddd2,dddd3,dddd 4; ;Sh?;	Shdddd1,dddd2,dddd3,dddd4;	스테핑모터에서 전류명령값의 최대치와 정지시의 전류치 비율을 설정함.
+#^ +#^ +#^	;XEdddd1,dddd2; or ;XEIIII1,IIII2; ;XE?;	XEdddd1,dddd2; or XEIIII1,IIII2;	스테핑모터 또는 절대각엔코더를 사용하는 경우 엔코더의 4체배된 펄스수를 설정함.
+#^	;Xedddd1,dddd2; ;Xe?;	Xedddd1,dddd2;	스테핑모터에서 DC-DC converter의 승압전압의 크기와 turn-off에 소요되는 시간을 설정함.



명령어의 분류 (jog 작동 명령어)

적용 CPU	명령어	return value	operation
+#^%	;J;	J;	모터1의 속도모드 조그 작동을 수행함.
+#^%	;K;	K;	모터2의 속도모드 조그 작동을 수행함.
+#^%	;SJHL,dddd1,dddd2,dddd3;	SJHL,dddd1,dddd2,dddd3;	속도모드의 조그동작에서 조그방향, 조그이동속도 및 조그
+#^%	;SJ?;		동작 지속시간을 정의함.
+#^%	;j;	j;	모터1의 위치모드 조그 작동을 수행함.
+#^%	;k;	k;	모터2의 위치모드 조그 작동을 수행함.
+#^%	;SjHL,dddd1,dddd2,dddd3;	SjHL,dddd1,dddd2,dddd3;	위치모드의 조그동작에서 조그방향, 조그이동속도 및 조그
+#^%	;Sj?;		동작 지속거리를 정의함.

명령어의 분류 (homing 작동 명령어)

적용 CPU	명령어	return value	operation
+#^%	;GHHL,HL,iiii1,iiii2;	CHAIL HILL IIII III III III	2개의 리밋센서와 1개의 홈센서를 사용한 홈 동작을 개시
+#^	;GH?;	GHHL, HL, 11111, 11112;	함. (0.01도 단위의 offset 변위)
+#^	;GhHL,HL, 1, 12;	CHUI HI IIII IIIO	2개의 리밋센서와 1개의 홈센서를 사용한 홈 동작을 개시
+#^	;Gh?;	GhHL, HL, IIII1, IIII2;	함. (4체배 엔코더 펄스단위의 offset 변위)
+#^	;GIHL,HL,iiii1,iiii2;	GIHL,HL,IIII1,III12;	2개의 리밋센서 만을 사용한 홈 동작을 개시함. (0.01도
+#^	;GI?;		단위의 offset 변위)
+#^	;GiHL,HL, 1, 2;	GiHL,HL,IIII1,IIII2;	2개의 리밋센서 만을 사용한 홈 동작을 개시함. (4체배 엔
+#^	;Gi?;		코더 펄스단위의 offset 변위)
+#^	;GJHL,HL,iiii1,iiii2;	 GJHL,HL, 1, 2;	한개의 홈센서 만을 사용한 홈 동작을 개시함. (0.01도 단
+#^	;GJ?;	GJNL, NL, IIIII, IIIIZ,	위의 offset 변위)
+#^	;GjHL,HL, 1, 2;	 GjHL,HL, 1, 2;	한개의 홈센서 만을 사용한 홈 동작을 개시함. (4체배 엔
+#^	;Gj?;	djnL,nL,11111,11112,	코더 펄스단위의 offset 변위)
+#^	;GKHL,HL,iiii1,iiii2;		한개의 홈센서 만을 사용한 홈 동작을 개시하되 홈센서가
+#^	;GK?;	GKHL, HL,	ON 되어 있는 상태에서만 개시함. (0.01도 단위의 offset
'π	, un: ,		변위)
+#^	;GkHL,HL, 1, 2;		한개의 홈센서 만을 사용한 홈 동작을 개시하되 홈센서가
+#^	;Gk?;	GkHL, HL,	ON 되어 있는 상태에서만 개시함. (4체배 엔코더 펄스단위
'π	, GK!		의 offset 변위)
+#^	;\$ 1, 2, 3, 4;	SIIII11, III12, III13, III14;	홈찾기 동작에서 변위의 한계값을 설정함.
+#^	;\$1?;		요夫가 ㅇㄱ에게 근거러 근겞을 글ㅇㅁ.
+#^	;GTdddd1,dddd2;	GTdddd1,dddd2;	리밋센서와 홈센서의 ON/OFF를 검출하는 경계전압을 설정
+#^	;GT?;	diadai, dada,	함.



명령어의 분류 (sequence 제어 명령어)

적용 CPU	명령어	return value	operation
+#^	;XPXAAAAAAAAAA;	VDVAAAAA AAAA	시퀜스제어 프로그램의 현재 포인터 위치에 시퀜스제어 프
T#* *	, APAAAAAAAAAA,	XPXAAAAAAAAAA;	로그램을 입력함.
			시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제
+#^	;XP?;	XPAAAAAAAAAA;	어 프로그램 한 스텝을 보여주고, 포인터 위치는 변경하지
			않음.
			시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제
+#^	;XP+;	XPAAAAAAAAAA;	어 프로그램 한 스텝을 보여주고, 포인터 위치를 다음 스
			텝위치로 이동함.
			시퀜스제어 프로그램의 포인터 위치를 강제 설정하고, 설
+#^	;XPdddd1;	XPAAAAAAAAA;	정된 포인터 위치의 시퀜스제어 프로그램 한 스텝을 보여
			주고, 포인터 위치는 변경하지 않음.
+#^	;XPp;	XPpdddd1;	현재의 시퀜스제어 프로그램 포인터 위치를 보여줌.
+#^	;XPsA55A;	XPsA55A;	메모리상의 시퀜스제어 프로그램을 EEPROM으로 저장함.
+#^	;XPT;	XPTHLHL, HLHL;	현재의 PORT_STATUS_FOR_X_CONTROL와 X_PORT_STATUS를 보
Ιπ			여줌.
+#^	;XPt;	XPtHLHL,HLHL;	현재의 X_timer0, X_timer0_period, X_counter0의 값을 보
Ιπ			여줌.
+#^	;XPP;	XPPHLHL, HLHL;	현재와 직전의 시퀜스제어 프로세스 카운트를 보여줌.
+#^	;XPF;	XPFHLHL;	현재의 X_STATUS_FLAG를 보여줌.
+#^	;XPS;	XPSHLHL, HLHL;	한스텝씩 시퀜스제어 프로그램을 실행시키는 모드일 때 한
T#	,,,,,	APSHLAL, ALAL,	스텝 씩 시퀜스제어 프로그램을 실행하게 함.



명령어의 분류 (출력 강제작동 명령어)

적용 CPU	명령어	return value	operation
+#^	;PWiiii1,iiii2; (VER4/8)		
+#^	;PWiiii1,iiii2,iiii3,iiii	;PWiiii1,iiii2; (VER4/8)	
	4; (VER5)	;PWiiii1,iiii2,iiii3,iiii4; (VER5)	모터의 PWM duty 출력을 강제로 설정함.
+#^	;PWiiii1,iiii2,iiii3,iiii	;PWiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;	포너의 FWW duty 물식을 당세도 일당함.
	4, iiii5, iiii6; (VER6/7)	(VER6/7)	
+#^	;PW?;		
+#^	;DXHLHL,HLHLHL;	DXHLHL, HLHL, HLHL, HLHL;	ON/OFF 출력과 7-seg 표기값을 설정하고 ON/OFF 입력과
+#^	;DX?;	DANLIL, ILIL, ILIL, ILIL, ILIL,	switch 입력을 읽어서 host로 전송함.
+#^	;XOHL,HL;	VOLII LII •	EXT_FAULT1_port와 EXT_FAULT2_port를 강제로 set 또는
+#^	;X0?;	XOHL, HL;	clear 함.
#^	;Xodddd1;	Voddadi.	스테핑모터에서 50V 승압회로와 역기전력 해소 pass를
#^	;Xo?;	Xodddd1;	ON/OFF 함.
#^	;Xodddd1;	Vodddd1:	380V 용 모터제어기에서 PRECHARGE_RELAY, MAIN_RELAY,
#^	;Xo?;	Xodddd1;	BRAKE_RELAY를 강제 설정함.



명령어의 분류 (입력 모니터 명령어)

적용 CPU	명령어	return value	operation
+#^%	;QS;	QSHL,HLHL,HLHL,HLHL; (스텝핑모터가 아닌 경우) QSHL,HLHL,HLHLHLHL,HLHLHLHL; (스텝핑모 터인 경우)	on/off 형식의 센서 상태를 읽어서 host로 전송함.
+#^	;Qs; (절대각센서를 사용하 는 경우가 아닐때)	QsHLHL, HLHL, HLHL;	인공안구의 경우는 팬/틸트방향의 절대각을 읽어서 host로 전송하며, 베스트모션의 경우는 경사각 센서값을 읽어서 host로 전송함.
#	;Qs; (절대각센서를 사용하 는 경우)	QsHLHLHLHL, HLHLHLHL;	절대각센서값을 읽어서 host를 전송함.
+#^%	;QA;	QAHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL	하위 8 channel의 12-bit AD 변환값을 읽어서 host로 전송함.
+#^%	;QB;	QBHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL	상위 8 channel의 12-bit AD 변환값을 읽어서 host로 전송함.
+#^	;Qa;	QaHLHL, HLHL, HLHL;	TLV2556의 12-bit A/D 변환값을 읽어서 host로 전송함.



명령어의 분류 (초기화 명령어)

적용 CPU	명령어	return value	operation
+#^%	;QEA55A;	QEA55A;	모터 1/2의 순간목표/명령/현재변위, 엔코더카운터값, 제
+#^	;QEA55A, <mark>HLHL,HLHL</mark> ;	QEA55A;	어오차등을 초기값으로 재설정함.
+#^%	;QEA5A5;	QEA5A5;	모터 1의 순간목표/명령/현재변위, 엔코더카운터값, 제어
+#^	;QEA5A5, <mark>HLHL</mark> ;	QEA5A5;	오차등을 초기값으로 재설정함.
+#^%	;QE5A5A;	QE5A5A;	모터 2의 순간목표/명령/현재변위, 엔코더카운터값, 제어
+#^	;QE5A5A, <mark>HLHL</mark> ;	QE5A5A;	오차등을 초기값으로 재설정함.
+#^	;RIA55A;	RIA55A;	모터 1/2의 적분오차값을 강제로 0으로 설정함.
+#^	;RIA5A5;	RIA5A5;	모터 1의 적분오차값을 강제로 0으로 설정함.
+#^	;R15A5A;	R15A5A;	모터 2의 적분오차값을 강제로 0으로 설정함.
+#^	;RrA55A;	N.A.	모터제어기를 리셋함.



명령어의 분류 (과전압, 저전압 및 과전류 보호 명령어)

적용 CPU	명령어	return value	operation
		Svdddd1,dddd2,dddd3,dddd4,dddd5; (2011	
+#^%	;Svdddd1,dddd2,dddd3;	년 9월 11일 이전 S/W 버젼)	MOSFET의 gate를 off 시키는 전압 강하량과 과전압 및 검
+#^%	;Sv?;	Svdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,	출시간의 크기를 설정함.
		dddd7; (2011년 9월 11일 S/W 버젼부터)	
+#^	;Sddddd1,dddd2,dddd3,dddd4;	Sddddd1,dddd2,dddd3,dddd4;	가속/감속시 저전압/과전압이 발생하는 것을 억제하기 위
+#^	;Sd?;	Sadada 1, adda2, adda3, addu4,	하여 가감속률을 제한하는 파라메터를 설정함.
+#^	;SDdddd1,dddd2;	SDdddd1,dddd2;	가속/감속시 저전압/과전압이 발생하는 것을 억제하기 시
+#^	;SD?;	SDadad F, adda2 ,	작하는 전압변동량을 설정함.
+#^	;ZPdddd1; (+#)	ZPdddd1;	역기전력 해소용 전력저항의 최대 ON time을 설정함.
+#^	;ZP?; (+#)	ZFuudu I,	작가선학 애조중 선탁자용의 최대 UN TIMB를 불당함.
+#^%	;SIdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;;SI?;	SIdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;	구동회로와 기구를 보호하기 위한 logic에서 사용하는 전 류리밋값과 time threshold를 설정함.
+#^%	;SWdddd1,dddd2;		과전류시에 HW적으로 회로를 차단하기 위한 과전류 설정값
+#^%	;SW?;	SWdddd1,dddd2;	을 설정함.
	2012년 10월 03일 이전 S/W		
+#^%	버젼;Sidddd1,dddd2;	Sidddd1,dddd2;	
+#^%	;Si?;		RJM_VER1과 RJM_VER2의 경우에는 과전류에러 fault 진단을
+#^ +#^ +#^	2012년 10월 03일 S/W 버젼부 터 ;Sidddd1,dddd2; 또는 ;Sidddd1,dddd2,dddd3,dddd4; ;Si?;	;Sidddd1,dddd2,dddd3,dddd4;	위한 1.024초 동안의 과전류검출 횟수를 설정하며, 그외의 경우는 SI 명령어로 설정한 전류치보다 실제전류가 연속적 으로 초과하기를 허용하는 최대시간을 설정함.
	;Sxdddd1,dddd2,dddd3,dddd4, dddd5,dddd6,dddd7,dddd8; F,Sx?;	Sxdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;	제어오차가 과도하게 큰 상태를 진단하기 위한 위치오차, 속도오차, 전류오차, 허용시간을 설정함. (RJM_VER5에는 40

Preliminary Rev. Jan/08/2013

명령어의 분류 (과온도 보호 명령어)

적용 CPU	명령어	return value	operation
+#^	;ZTdddd1,dddd2;	ZTdddd1,dddd2;	 모터1과 모터2의 최대허용 권선온도상승값을 설정함.
+#^	;ZT?;	Zidddi, dddz,	1313 1315 34VO CCC100WE E00.
+#^	;ZQdddd1,dddd2,dddd3,dddd4,	ZQdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,	모터1의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1,
1.44	dddd5,dddd6,dddd7,dddd8;	dddd7,dddd8;	Rth2, Tau1, Tau2)를 설정함.
+#^	;ZQ?;		
+#^	;ZRdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;	ZRdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,	모터2의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1,
+#^	;ZR?;	dddd7,dddd8;	Rth2, Tau1, Tau2)를 설정함.
			IRMCK201을 사용한 RJM_VER3에 있어서 모터1과 모터2의
+#^	;Zcdddd1,dddd2;	Zcdddd1,dddd2;	nominal current(IRMCK201 파라메터 설정에 사용한 값)를
+#^	;Zc?;	,	설정함.
+#^	;MTffff1,ffff2,ffff3;	NTIIII IIIO IIIO	O C 총 정 요. 네 시 피 기 메디르 성 정 하
+#^	;MT?;	MTffff1,ffff2,ffff3;	온도측정용 센서 파라메터를 설정함.
+#^	;Mtdddd1;	Mtdddd1,dddd2,iiii3;	온도측정용 센서를 사용하여 측정된 저항치 및 온도값을
+#^	;Mt?;	mtudut, addaz, 11110,	읽어내어 host로 전송함.
+#^	;QW;	QWIIII0, IIII1, IIII12,, IIII15, IIII16, II	
. ,,		117;	torque, efficiency를 읽어서 host로 전송함.
		Qwiiii0,iiii1,iiii2,,iiii5,iiii6,ii	모터1과 모터2의 loss power, winding temperature,
+#^	;Qw;	117;	housing temperature, ambient temperature를 읽어서 host
			로 전송함.



명령어의 분류 (fault상태 모니터 명령어)

적용 CPU	명령어	return value	operation
+#^%	;Q2;	Q2 HLHL,HLHL;	fault_status, fault1_status를 읽어서 host로 전송함.
+#^%	;Q3;	Q3 HLHL;	IRMCK201의 fault_status를 읽어서 host로 전송함.
+#^	;Qv;	QvHLHL, HLHL;	현재의 공급전압과 전압강하의 크기를 읽어서 host로 전송함.
+#^	;QJ;	QJHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL HL, HLHL;	faultBufPointer, faultBuf[0-7]을 읽어서 host로 전송함.
+#^	;Qj;	Q;HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH	faultBufPointer_save, faultBuf_save[0-14]을 읽어서 host로 전송함.
+#^	;QjA55A;	QjHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL HL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, H LHL, HLHL;	faultBufPointer_save, faultBuf_save[0-14]을 0으로 clear함.
+#^	;RJI;	RJn <aaaaaaaaaaa>;</aaaaaaaaaaa>	fault code를 저장한 RAM buffer를 읽어서 fault의 내용을 문자로 보여주는 index를 가장 최근의 fault code를 지정 하도록 초기화함.
+#^	;RJ;	RJn <aaaaaaaaaa>;</aaaaaaaaaa>	fault code를 저장한 RAM buffer를 읽어서 fault의 내용을 문자로 보여줌.
+#^	;Rjl;	Rjnold <aaaaaaaaaaaa>; 또는 Rjnnew<aaaaaaaaaaa< td=""><td>EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문 자로 보여주는 index를 가장 최근의 fault code를 지정하 도록 초기화함.</td></aaaaaaaaaaa<></aaaaaaaaaaaa>	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문 자로 보여주는 index를 가장 최근의 fault code를 지정하 도록 초기화함.
+#^	;Rj;	Rjnold <aaaaaaaaaaa>; 또는 Rjnnew<aaaaaaaaaa>;</aaaaaaaaaa></aaaaaaaaaaa>	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문 자로 보여줌.



명령어의 분류 (제어작동상태 모니터 명령어)

적용 CPU	명령어	return value	operation
+#^%	;QP;	QPHLHLHLHL, HLHLHLHL;	현재 변위값을 읽어서 host로 전송함.
+#^%	;Qp;	QpHLHLHLHL, HLHLHLHL;	순간목표변위값을 읽어서 host로 전송함.
+#^	;Qb9;	QbHBLBHBLBHBLBCS;	모터제어기에서 모터1/2의 순간 위치명령값과 현재 위치값을 읽어서 host로 전송함.
+#^	;Qq;	QqHLHLHLHL, HLHLHLHL;	option으로 부착된 엔코더카운트값을 읽어서 host로 전송함.
+#^%	;QV;	QVIIII1,IIII2; (F2406인경우), QVIIII1,IIII2; (F2811/F28334인 경우)	현재 속도값을 읽어서 host로 전송함.
+#^%	;Qe; (Force/Moment 센서가 아닌 경우)	Qe 1, 2, 3, 4, 5, 6;	error, error 적분값, error 미분값을 읽어서 host로 전송함.
+#^%	;QF;	QFiiii0,iiii1,iiii2,,iiii5,iiii6,iii i7;	filter 출력값을 읽어서 host로 전송함.
+#^%	;Q1;	Q1 HLHL;	operation_status를 읽어서 host로 전송함.
+#^%	;Q4;	Q4 HLHL, HLHL, HLHL;	프로그램 작성 날짜, version, macro 설정을 읽어서 host 로 전송함.
+#^	;Q5;	Q5 HLHL;	operation1_status를 읽어서 host로 전송함.
+#^	;Q6;	Q6 HLHL,HLHL;	conditional compiling 조건을 읽어서 host로 전송함.



명령어의 분류 (제어작동상태 변수 모니터 명령어)

적용 CPU	명령어	return value	operation
+#^% +#^%	;Qxdddd1; ;Qx?;	Qxdddd1;	1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는 8word를 host PC로 전송하는 데이터의 종류를 선택하는 변 수인 data_logging_select 변수값을 설정함.
+#^%	;Qxy;	없음	1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는 8word를 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;Qxyz;	없음	Qxy; 또는 Qz; 또는 Qzx; 또는 Qzy; 명령에 의한 모니터링 작동을 중지함.
+#^% +#^%	;QZHLHL,HLHL,HLHL,HLHL; ;QZ;	QZHLHL, HLHL, HLHL;	Qz 명령어로 1ms 마다 값을 host로 전송하는 4개의 변수, 또는 Qy 명령어로 host로 전송하는 8개 변수중에서 앞쪽 4 개의 변수에 대한 주소를 설정함.
+#^%	;Qz;	없음	QZ 명령어로 주소가 지정된 4word를 1msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;Qzx;	없음	QZ 명령어로 주소가 지정된 4word를 2msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;Qzy;	없음	QZ 명령어로 주소가 지정된 8word를 2msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;QYHLHL,HLHL,HLHL,HLHL; ;QY;	QYHLHL,HLHL,HLHL;	Qy 명령어로 host로 전송하는 8개 변수중에서 뒷쪽 4개의 변수에 대한 주소를 설정함.
+#^	;Qy;	RS232/485의 경우 : QyHBLBHBLBHBLBHBLBHBLBHBLBHBLB; CAN의 경우 : HBLBHBLBHBLBHBLBHBLBHBLBHBLB	QZ,QY 명령어로 주소가 지정된 8word를 binary 값의 형태로 host PC에 전송함.



명령어의 분류 (제어작동상태 레지스터 및 메모리 읽기/쓰기 명령어)

적용 CPU	명령어	return value	operation
+#^%	;QRHLHL,HL;	001111111111111111111111111111111111111	지정된 주소에서 시작하여 지정된 양만큼의 DSP SRAM 내용
T#* %		QRHLHL, HL, HLHL(, HLHL);	을 읽어서 host로 전송함.
+#^	;QQHLHL,HL;	QQHLHL(,HLHL);	지정된 주소에서 시작하여 지정된 양만큼의 DSP SRAM 내용
111	, MM ILI IL , I IL ,		을 읽어서 host로 전송함.
+#^%	;QTHLHL,HL,HLHL(,HLHL);	QTHLHL,HL,HLHL(,HLHL);	지정된 주소에서 시작하여 지정된 양만큼의 데이터를 DSP
1π /0	, X	WITHER IE, THE IE (, THE IE / ,	의 SRAM에 전송함.
			BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의
+#^%	;QrHL,HL;	QrHL,HL,HLHL;	지정된 주소로부터 register 값을 읽어서 host로 전송함.
			(BLDC 구동 chip IRMCK201 전용)
+#^%	;QtHL,HL,HL;	QtHL,HL,HL;	BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의
+#^%	;QtHL,HL,HLHL;	QtHL,HL,HLHL;	지정된 주소에 host로부터 전송된 데이터를 저장함과 동시
+#^%	;QtHL,HL?;	QtHL,HL,HLHL;	에 DSP의 SARAM 0x0a3e-0x0abd 번지에도 저장함. (BLDC 구
111 /0	, ACTIC, 11C: ,		동 chip IRMCK201 전용)

명령어의 분류 (전류측정 명령어)

적용 CPU	명령어	return value	operation
+#^%	;QC;	QCiiii1,iiii2; QCiiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) QCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) QCiiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)	모터에 흐르는 filtering된 상전류값을 읽어서 host로 전송함.
+#^%	;Qc;	Qciiii1,iiii2; Qciiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) Qciiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) Qciiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)	모터에 흐르는 filtering된 상전류값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.
+#^	;QK;	QKiiii1,iiii2;	모터에 흐르는 filtering된 총전류값을 읽어서 host로 전송함.
+#^	;Qk;	Qkiiii1,iiii2;	모터에 흐르는 filtering된 총전류값을 읽어서 암페어값으로 환산하여 host로 전송함.
+#^ +#^	;MC; ;MC?;	MCiiii1,iiii2,iiii3,iiii4;	BLDC 또는 AC servo 모터의 lq, ld 값을 읽어서 host로 전송함.
+#^ +#^	;Mc; ;Mc?;	Mciiii1,iiii2,iiii3,iiii4;	BLDC 또는 AC servo 모터의 lq, ld 값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.



명령어의 분류 (test 구동 명령어)

적용 CPU	명령어	return value	operation
+#^	;Tsdddd1,dddd2;	Tsdddd1,dddd2;	모터1의 step 구동을 위한 주기와 진폭을 설정하고 step
+#^	;Ts?;	Isaada I, aada 2,	구동을 실행하도록 함.
+#^	;Ttdddd1,dddd2;	Ttdddd1,dddd2;	모터2의 step 구동을 위한 주기와 진폭을 설정하고 step
+#^	;Tt?;	rtududi, duduz,	구동을 실행하도록 함.
+#^	;Tudddd1,dddd2;	Tudddd1,dddd2;	모터1의 사인파 구동을 위한 주기와 진폭을 설정하고 사인
+#^	;Tu?;	Tudada 1, dada2 ;	파 구동을 실행하도록 함.
+#^	;Tvdddd1,dddd2;	Tvdddd1,dddd2;	모터2의 사인파 구동을 위한 주기와 진폭을 설정하고 사인
+#^	;Tv?;	TVddddT,dddd2,	파 구동을 실행하도록 함.
+#^	;S1dddd1,dddd2;	\$1ddd1_ddd2:	OE/OF 위치제어모드에서 모터1의 sine파 변위 주기 및 진
+#^	;\$1?	S1dddd1,dddd2;	폭을 설정함.
+#^	;S2dddd1,dddd2;	S2dddd1,dddd2;	OE/OF 위치제어모드에서 모터2의 sine파 변위 주기 및 진
+#^	;\$2?	Szadad i, adduz,	폭을 설정함.
+#^% +#^%	RJM_VER5/6 STEP형이 아닌 경우 : ;SHiiii1,iiii2; ;SH?;	SHiiii1,iiii2;	host direct electrical angle 제어 모드에서 구동전류 (RJM_VER7형의 3상/5상 STEP의 경우로서 작동모드 09번이 아닐때) 또는 구동전압의 진폭 (BLDC 구동 chip IRMCK201 경우) 또는 PWM duty (DC 모터를 제외한 기타의 경우)를 설정함.
+#^ +#^	;WXiiii1,iiii2; ;WX?;	WXIIII1,IIII2;	RJM_VER7 모터의 electrical angle을 강제로 설정함.
+#^%	;Qddddd1;	Qddddd1;	Sleep_1micro(dddd1), Sleep_10micro(dddd1), Sleep_msec(1) 함수실행결과의 지연시간동안 CS_EEPROM_N_port에 low pulse를 출력함.



명령어의 분류 (튜닝 명령어)

적용 CPU	명령어	return value	operation	
+#^	;WKdddd1;	WK dada adda o	encoder/digital_Hall_IC_only type BLDC 모터의 자동 파	
+#^	;WK?;	WKdddd1,dddd2;	라메터 설정을 개시함.	
+#^	;WCdddd1,dddd2;	WCdddd dddo:	analog Hall sensor type의 RJM_VER7 모터에 대하여 위상	
+#^	;WC?;	WCdddd1,dddd2;	각 측정용 파라메터의 설정을 개시함.	
+#^	;Wkdddd1,dddd2; (2011년 6			
	월 14일 이전 S/W 버젼)	Wkdddd1,dddd2; (2011년 6월 14일 이전	RJM_VER7 BLDC 모터제어기에서, 모드8의 host direct	
+#^	;Wkdddd1,dddd2,dddd3,dddd	S/W 버젼)		
	4,dddd5; (2011년 6월 14일	Wkdddd1,dddd2,dddd3,dddd4,dddd5; (2011	angle control을 사용하여 자동튜닝 등의 구동을 할 때,	
	S/W 버젼부터)	년 6월 14일 S/W 버젼부터)	사용하는 PWM duty 값을 설정함.	
+#^	;Wk?;			
+#^	;WGdddd1,dddd2,dddd3,dddd		analog Hall concer type QL D.IM VED7 ELEIQL MLH coals	
	4,dddd5,dddd6;	WGdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall sensor type의 RJM_VER7 모터1의 센서 scale 과 offset을 설정함.	
+#^	;WG?;		파 OTISet들 설정함.	
+#^	;WHdddd1,dddd2,dddd3,dddd		analog Hall concer type OLD IM VED7 TIELDOL MILL coals	
	4,dddd5,dddd6;	WHdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall sensor type의 RJM_VER7 모터2의 센서 scale	
+#^	;WH?;		과 offset을 설정함.	
#	;WNdddd1,dddd2;	WNdddd1,dddd2;	RJM_VER7 analog Hall sensor type에서 구동가능한 영역에	
#	;WN?;	mydddd i , ddudz ,	서의 analog Hall sensor wave 총갯수를 설정함	
+#^%	;SZdddd1,dddd2;		Hall sensor pulse의 지정된 edge를 기준으로 하여 엔코더	
+#^%	;SZ?;	SZdddd1,dddd2;	Z펄스의 위치를 앤코더펄스 갯수로 읽어 냄. (BLDC 구동	
1# /0	,02!,		chip IRMCK201 전용)	



세어기 원립어 LIST

명령어의 분류 (튜닝 명령어 - analog type 센서 사용 BLDC모터의 자동 파라메터 설정)

적용 CPU	명령어	return value	operation
+#^	;WJdddd1,dddd2,dddd3,dddd 4,dddd5; ;WJ?; (제일메티컬의 경우)	WJdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함.
#^ #^	;WJdddd1; ;WJ?; (나노모션의 linear Hall sensor의 경우)	WJdddd1,dddd2;	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함.
+#^	;WJdddd1,dddd2,dddd3,dddd 4,dddd5; ;WJ?; (그 외의 경우)	WJdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함.
+#^	;WLdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6,dddd7; ;WL?dddd1;	WLdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;	3상 analog Hall 센서를 사용하는 경우에 A/B/C 상신호의 교차점값을 설정함.
+#^	;WLPdddd1;	WLP;	3상 analog Hall 센서를 사용하는 경우에 지정된 삼상 wave sector 번호부터 A/B/C 상신호의 교차점값을 자동 설정함.
+#^	;WLQ;	WLQ;	3상 analog Hall 센서를 사용하는 자석가동자의 경우에 - 방향쪽 센서와 +방향쪽 센서의 위상차를 자동 설정함.
+#^	;WLR;	WLR;	3상 analog Hall 센서를 사용하는 자석가동자의 경우에 순위가 높은 제어기의 +방향쪽 센서와 순위가 낮은 현재 제어기의 -방향쪽 센서 사이의 위상차를 자동 설정함.



명령어의 분류 (text 입력/저장과 LCD 표시장치 관련 명령어)

적용 CPU	명령어	return value	operation	
+#^	;XTAAAAAAAAAA;	VTAAAAA AAAA	text의 현재 포인터 위치에 text를 저장하며 저장된 text	
T#* *		XTAAAAAAAAAA;	를 보여줌.	
			text의 포인터 위치를 강제 설정하고 해당 위치에 있는	
+#^	;XT?dddd1;	XTdddd1,AAAAAAAAAA;	text 한 줄을 보여주고, 포인터 위치는 강제 설정된 값을	
			유지함.	
+#^	;XT?;	XTAAAAAAAAAA;	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고,	
ιπ	,,,,	Alamanaana)	포인터 위치는 변경하지 않음.	
+#^	;XT+;	XTAAAAAAAAAA;	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고,	
ιπ	,,,,	Alamanaana)	포인터 위치를 다음 줄 위치로 이동함.	
+#^	;XTp;	XTpdddd1;	현재의 text 포인터 위치값를 보여줌.	
+#^	;XTsA55A;	XTsA55A;	메모리상의 text 를 EEPROM으로 저장함.	
+#^	;DZHLHL,HLHL,HLHL;	DZHLHL, HLHL, HLHL;	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에	
+#^	;DZ?;		표시될 값의 주소를 설정함.	
+#^	;DFHLHL,HLHL,HLHL;	DFHLHL, HLHL, HLHL;	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에	
+#^	;DF?;		표시될 값의 type과 표시형식을 설정함.	
+#^	;DSfffff1,fffff2,fffff3,ffff		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에	
	4;	DSffff1,ffff2,ffff3,ffff4;	표시될 값의 scale factor를 설정함.	
+#^	;DS?;		THE WA SOUTH TROUBLE EST.	
+#^	;DTdddd1,AAAAAAAA;	DTdddd1, AAAAAAAA;	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에	
+#^	;DT?dddd1;		표시되는 값의 타이틀을 설정함.	



명령어의 분류 (데모작동 명령어)

적용 CPU	명령어	return value	operation
+#^	;DDdddd1,dddd2,dddd3; or		
+#^	;DDdddd1,dddd2,dddd3,dddd	DDdddd1,dddd2,dddd3; or	일정스텝으로 이동을 반복하는 데모작동의 파라메터를 설
	4;	DDdddd1,dddd2,dddd3,dddd4;	정함.
+#^	;DD?;		
+#^	;DEHLHL;	DEHLHL;	일정스텝으로 이동을 반복하는 데모작동의 작동모드를 설
+#^	;DE?;		정함.

명령어의 분류 (특정 목적의 기능관련 명령어)

적용 CPU	명령어	return value	operation
+#^	;WMHLHL; ;WM?;	WMHLHL;	Wafer 회전위치제어시에 위치모드로 정지한 한 상태에서 속도모드로 작동모드를 변경하며, 이에 따른 변수초기화 작동을 실시함.
+#^ +#^	Wafer control의 경우 (2012년 2 월 18일 S/W 버젼부터 적용): ;Shdddd1,dddd2,dddd3,dddd4; ;Sh?; (+#^)	Shdddd1,dddd2,dddd3,dddd4;	BLDC모터에서 holding torque를 만드는 파라메터를 설정함.
#	;MBdddd1,dddd2,dddd3,dddd4,ddd d5,dddd6,dddd7,dddd8,dddd9; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9;	brake actuator의 동작관련 파라메터를 설정함.
#	;Mbdddd1,dddd2,dddd3,dddd4,ddd d5,dddd6,dddd7; ;Mb?;	Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;	brake actuator의 동작관련 파라메터를 설정함.
#	;MEdddd1,dddd2,dddd3,dddd4;;ME?;	MEdddd1,dddd2,dddd3,dddd4;	brake actuator의 동작관련 파라메터를 설정함.
+#	;MBiiii1,iiii2,iiii3,iiii4,iii i5,iiii6,iiii7; ;MB?;	MBiiii1,iiii2,iiii3,iiii4,iiii5,iii i6,iiii7;	EPS의 동작관련 파라메터를 설정함.
#	;MBdddd1,dddd2,dddd3,dddd4,ddd d5,dddd6,dddd7; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;	자석가동자 제어기의 동작관련 파라메터를 설정함.
#	;Mbdddd1, 2; ;Mb?;	Mbdddd1, IIII2;	자석가동자 제어기의 동작관련 파라메터를 설정함.



명령어의 분류 (F/T 센서 명령어 - 1)

적용 CPU	명령어	return value	operation
+#^%	;Qb;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	channel 8-11(F2809는 0-3)의 A/D 변환값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가됨)
+#^	;Qb1;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	Butterworth filter #8-#11의 필터링한 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6축 F/T 센 서의 경우는 2채널이 추가됨)
+#^	;Qb2;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor에서 bridge amplifier #1-#4의 offset 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가됨)
+#^	;Qb3;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor에서 계산된 모멘트값과 힘값을 읽어서 host로 전송함.
+#^	;Qb4;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor의 D/A 출력 전압(bridge amplifier의 영점 조정용)을 A/D 변환한 값을 읽어서 host로 전송함.
+#^	;Qb5;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor에서 모멘트값과 힘값을 계산한 후, 추가 필터 링한 값을 읽어서 host로 전송함.
+#^	;Qe; (Force/Moment 센서인 경 우)	Qeiiii0,iiii1,iiii2,iiii3; or Qeiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)	X-moment, Y-moment, Z-force, extra load cell값에 대하 여 추가 필터링한 값을 읽어서 host로 전송함. (6축의 경 우는 X-moment, Y-moment, Z-force, X-force, Y-force, Z-moment 임)
+#^	;Qf;	Qfiiii0,iiii1,iiii2,iiii3; or Qfiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)	X-moment, Y-moment, Z-force, extra load cell를 읽어서 host로 전송함. (6축의 경우는 X-moment, Y-moment, Z-force, X-force, Y-force, Z-moment 임)



명령어의 분류 (F/T 센서 명령어 - 2)

적용 CPU	명령어	return value	operation
+#^%	;b;	RS485의 경우 : bHbMbLbHbMbLbHbMbLbHbMbLbCS; CAN의 경우 : bHbMbLbHbMbLbHbMbLbHbMbLb; (6축 F/T 센서의 경우는 3 byte가 추가됨)	Butterworth filter 8-11 (6축 F/T 센서의 경우는 8-13)의 필터링된 값을 읽어서 host로 전송함.
+#^	;SIA55A,dddd1,dddd2; ;SI?; (Force/Moment 센서인 경 우)	SIdddd1,dddd2;	F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는 0/1/2/3) 전압의 평균값을 설정하기 위한 filtering을 실시하고, 온도값과 함께 force_data_offset 변수값을 설정함.
+#^	;SI5AA5,dddd1,dddd2; ;SI?; (Force/Moment 센서인 경 우)	SIdddd1,dddd2;	F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는 0/1/2/3) 전압의 평균값을 설정하기 위한 filtering을 실시하고, 온도증가에 따른 전압상승의 기울기를 설정함.
+#^	;SIA5A5,0,0; ;SI?; (Force/Moment 센서인 경 우)	SIdddd1,dddd2;	F/T 센서에서 no load 상태일때 AD channel 8/9/10/11(내 장형의 경우는 0/1/2/3)의 변환값이 2048이 되도록 D/A 변 환기를 조절하여 만듦. (F/T센서의 앰프 offset을 2048로 맞추어주는 작동이며 총 5초가 걸림)
+#^	;MOdddd1,dddd2,dddd3,dddd 4; ;MO?; (F/T센서의 경우)	MOdddd1,dddd2,dddd3,dddd4;	F/T 센서에서 TLV5630 또는 DAC124S085 D/A converter의 출력값을 설정함.



명령어의 분류 (F/T 센서 명령어 - 3)

적용 CPU	명령어	return value	operation
+#^	;MTffff1,ffff2,ffff3;	MTffff1,ffff2,ffff3;	온도측정용 센서 파라메터를 설정함.
+#^	;MT?;	MIIIII, IIII2, IIII3,	근도국영웅 센지 파다메디를 일정함.
+#^	;Mtdddd1;	Mtdddd1,dddd2,iiii3;	온도측정용 센서를 사용하여 측정된 저항치 및 온도값을
+#^	;Mt?;	Mtdddd1,dddd2,11113,	읽어내어 host로 전송함.
+^	;MUffff1,ffff2,ffff3;	MUfffff1,ffff2,ffff3;	온도측정용 두번째 센서 파라메터를 설정함.
+^	;MU?;	MOTITI, 11112, 11113,	근도국증증 구단째 엔지 피디메디크 글중함.
+^	;Mudddd1;	Mudddd1,dddd2,iiii3;	온도측정용 두번째 센서를 사용하여 측정된 저항치 및 온
+^	;Mu?;	muddat, addaz, 11110,	도값을 읽어내어 host로 전송함.
+#^	;MPffff1,ffff2,ffff3,ffff		
	4, ffff5, ffff6, ffff7, ffff	MPffff1,ffff2,ffff3,ffff4,ffff5,ffff6,f	4개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용
	8, ffff9, ffff10, ffff11, fff	fff7, ffff8, ffff9, ffff10, ffff11, fffff12;	하여 X-moment, Y-moment, Z-force를 계산하는 변환행렬 값을 설정함.
	f12;	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	
+#^	;MP?;		
+#^	;Mpfffff1,ffff2,ffff3,ffff		6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용하여 X-moment, Y-moment, Z-force, X-force를 계산하는 변환행렬 값을 설정함.
	4, ffff5, ffff6, ffff7, ffff	;Mpfffff1,ffff2,ffff3,ffff4,ffff5,ffff6,	
	8,ffff9,ffff10,ffff11,fff	ffff7,ffff8,ffff9,ffff10,ffff11,ffff12;	
	f12;		
+#^	;Mp;		
+#^	;Mqffff1,ffff2,ffff3,ffff		
	4 , ffff5, ffff6, ffff7, ffff	;Mqffff1,ffff2,ffff3,ffff4,ffff5,ffff6,	6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용
	8,ffff9,ffff10,ffff11,fff	ffff7, ffff8, ffff9, ffff10, ffff11, ffff12;	하여 Y-force, Z-moment를 계산하는 변환행렬 값을 설정
	f12;		함.
+#^	;Mq;		



명령어의 분류 (F/T 센서 명령어 - 4)

적용 CPU	명령어	return value	operation
+#^	;MSiiii1,iiii2,iiii3,iiii 4;	MSiiii1,iiii2,iiii3,iiii4;	F/T 센서에서 증폭후 A/D 변환된 스트레인 값의 온도증가에 따른 변화율을 설정함.
+#^	;MS?;		에 떠는 근거들을 걸음다.
+#^	;MViiii1,iiii2;	MViiii1,iiii2;	F/T 센서에서 온도보상의 기준온도와 온도보상 기울기를
+#^	;MV?;	MV1111,11112,	자동 계산할 때 사용한 또 다른 온도값을 설정함.
+#^	;Moiiii1,iiii2,iiii3,iiii		F/T 센서에서 수직력의 auto zero rate, maximum allowed
	4, i i i i i 5;	Moiiii1,iiii2,iiii3,iiii4,iiii5;	drift, X 모멘트 coupling, Y 모멘트 coupling,
+#^	;Mo?;		FORCE_SENSOR_ERROR_DETECTION_TIME 파라메터를 설정함.
+^	;Msdddd1,dddd2;	Msdddd1,dddd2;	삼성메디컬용 F/T 센서의 X-moment, Z-force coupling을
+^	;Ms?;	Msdddd1, dddd2,	처리하는 파라메터를 설정함.
+^	;MBdddd1,dddd2;	MPdddd1 dddd2:	F/T 센서 drift offset 제거 동작관련 파라메터를 설정함.
+^	;MB?;	MBdddd1,dddd2;	F/I 센시 UIII UIISEL 세기 중액션인 파다메디를 설정함.



명령어의 분류 (사용하지 않는 명령어)

적용 CPU	명령어	return value	operation
+#^	;WBdddd1,dddd2;	WD addd ddddo:	RJM_VER7 제어시에 100KHz interrupt routine에서 사용하
+#^	;WB?;	WBdddd1,dddd2;	는 Butterworth filter의 시정수를 설정함.
%	;Scdddd1,dddd2;	Scdddd1,dddd2;	전류치를 암페어로 환산하는 scale factor를 설정함.
%	;Sc?;	Scadad I, addaz,	전류자들 함페어도 관련하는 Scale lactor를 설명함.
%	;PWiiii1,iiii2;	PWiiii1,iiii2;	모터의 PWM duty 출력을 강제로 설정함.
%	;PW?;	PWIIII, IIII2,	포디의 FWM duty 물럭들 경제도 설명함.
%	;SViiii1,iiii2;	SViiii1,iiii2;	속도제어 모드에서 속도명령값을 설정함.
%	;SV?;	3411111,11112,	그도세이 포트에서 극도증증없을 들었답.

명령어	구분	설명
	operation	on/off 형식의 센서 상태를 읽어서 host로 전송함.
		QSHL,HLHL,HLHL; (스텝핑모터가 아닌 경우) HL is two-byte hex code of a byte.
		QSHL,HLHL,HLHLHL,HLHLHL; (스텝핑모터인 경우) HL is two-byte hex code of a byte.
		1st byte : bit0 - 2 : low(0)/high(1) status of LS1A/B/C
		: bit3 - 5 : low(0)/high(1) status of LS2A/B/C
		: bit6 : low(0)/high(1) status of ENC1Z of motor1
		: bit7 : low(0)/high(1) status of ENC2Z of motor2
		2nd word : bit0 : SEL_EEPROM_N_port
		: bit1 : MOT_EN_N_port
	return value	: bit2 : RXEN_N_port
		: bit3 : TXEN_port
		: bit4 : OFF12V_port (intelligent power IC를 사용하는 일부모델은 해당없음)
;QS; (+#^%)		: bit5 : DISABLE1_IN_port (intelligent power IC를 사용하지 않는 일부모델은 해당없음)
		: bit6 : DISABLE2_IN_port (intelligent power IC를 사용하지 않는 일부모델은 해당없음)
		: bit7 : CS_IRMCK1_port (RJM_VER3 모델에만 해당됨)
		: bit8 : CS_IRMCK2_port (RJM_VER3 모델에만 해당됨)
		: bit9 :
		: bit10 :
		: bit11 : LS1D_port (일부모델은 해당없음, 모델별 입력콘넥터 포트 사양참조)
		: bit12 : LS2D_port (일부모델은 해당없음, 모델별 입력콘넥터 포트 사양참조)
		: bit13 : EXT_FAULT1_port
		: bit14 : EXT_FAULT2_port
		: bit15 :
		3rd word or long word : motor1 encoder (4체배 pulse, 스텝핑모터인 경우는 32-bit, 그외는 16-bit)
		4th word or long word : motor2 encoder (4체배 pulse, 스텝핑모터인 경우는 32-bit, 그외는 16-bit)



명령어	구분	설명
	comment	bit1에 해당하는 MOT_EN_N_port의 값이 0이면 모터가 enable 되어 있는 것이고, 1이면 모터가 disable 되어 있는 것임. 어 있는 것임. 스테핑모터의 경우는 엔코더 입력단자가 설치되어 있는 경우에 한하여 엔코더 값을 32bit로 확장한 값을 리턴함.
	operation	인공안구의 경우는 팬/틸트방향의 절대각을 읽어서 host로 전송하며, 베스트모션의 경우는 경사각 센서값을 읽어서 host로 전송함.
		QsHLHL,HLHL,HLHL; HL is two-byte hex code of a byte. HLHL는 word값을 4개의 hex code로 바꾼 값인데 high byte:low byte 순으로 표현됨.
;Qs; (+#^) (절대각센서를 사 용하는 경우가 아	return value	인공안구의 경우(F2811/F28334만 해당) 1st word : 팬방향의 절대각을 14bit로 표현한 값 (bit) 2nd word : 팬방향의 360도 회전수 3rd word : 틸트방향의 절대각을 14bit로 표현한 값 (bit) 4th word : 틸트방향의 360도 회전수
닐때)		베스트모션의 경우(F28334만 해당) 1st word : 경사각센서1의 경사각을 16bit로 표현한 값 (bit) 2nd word : 경사각센서1의 1bit status 값 3rd word : 경사각센서2의 경사각을 16bit로 표현한 값 (bit) 4th word : 경사각센서2의 1bit status 값
	comment	인공안구의 경우는 팬/틸트방향의 14bit 절대각을 읽은 값이며, 베스트모션의 경우 16bit의 경사각센서를 읽은 값임. 인공안구의 경우에 360도 회전수는 signed integer 임을 주의할 것.



명령어	구분	설명
;Qs; (#)	operation	절대각센서값을 읽어서 host롤 전송함.
	return value	QsHLHLHLHL, HLHLHLHL; HL is two-byte hex code of a byte.
(절대각센서를 사 용하는 경우)	comment	F28334로서 절대각센서를 사용하는 경우에 한함.
	operation	하위 8 channel의 12-bit A/D 변환값을 읽어서 host로 전송함.
;QA; (+#^%)	roturn volue	QAHLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL; HL is two-byte hex code of a byte.
, QA, (T# 70)	return value	HLHL는 word값을 4개의 hex code로 바꾼 값인데 high byte:low byte 순으로 표현됨.
	comment	각각의 값이 갖는 의미는 모델별로 다름.
	operation	상위 8 channel의 12-bit A/D 변환값을 읽어서 host로 전송함.
;QB; (+ # ^%)	return value	QBHLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL; HL is two-byte hex code of a byte.
, UD, (T# 70)		HLHL는 word값을 4개의 hex code로 바꾼 값인데 high byte:low byte 순으로 표현됨.
	comment	각각의 값이 갖는 의미는 모델별로 다름.
	operation	TLV2556의 12-bit A/D 변환값을 읽어서 host로 전송함.
;Qa; (+#^)	return value	QaHLHL, HLHL, HLHL;
	comment	TLV2556 12-bit A/D 변환기를 가지거나 RJM_VER7에서만 실제값이 리턴됨.



명령어	구분	설명
	operation	channel 8-11(F2809는 0-3)의 A/D 변환값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가
		됨)
		QbHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
		○S 는 앞선 8 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
		♣ F2809가 아닌 경우:
		1st word : A/D channel 8의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #1)
		2nd word : A/D channel 9의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #2)
		3rd word : A/D channel 10의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #3)
		4th word : A/D channel 11의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #4)
;Qb; (+#^%)	return value	5th word : A/D channel 12의 A/D값 (bit, 6축 F/T sensor의 경우 bridge amplifier #5)
		6th word : A/D channel 13의 A/D값 (bit, 6축 F/T sensor의 경우 bridge amplifier #6)
		♣ F2809인 경우:
		1st word : A/D channel 0의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #1)
		2nd word : A/D channel 1의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #2)
		3rd word : A/D channel 2의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #3)
		4th word : A/D channel 3의 A/D값 (bit, F/T sensor의 경우 bridge amplifier #4)
		5th word : A/D channel 4의 A/D값 (bit, 6축 F/T sensor의 경우 bridge amplifier #5)
		6th word : A/D channel 5의 A/D값 (bit, 6축 F/T sensor의 경우 bridge amplifier #6)
	comment	F/T 센서의 경우에 A/D 변환값은 4개(6축 F/T 센서의 경우는 6개)의 strain-guage bridge에서 증폭된 전압
		을 A/D 변환한 값임.



명령어	구분	설명
	operation	Butterworth filter #8-#11의 필터링한 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6
		축 F/T 센서의 경우는 2채널이 추가됨)
		QbHBLBHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
		CS 는 앞선 10 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
		1st word : butterworth filter #8의 A/D 변환값을 필터링한 값 (bit)
·Ob 1 · (1#A)	roturn value	2nd word : butterworth filter #9의 A/D 변환값을 필터링한 값 (bit)
;Qb1; (+#^)	return value	3rd word : butterworth filter #10의 A/D 변환값을 필터링한 값 (bit)
		4th word : butterworth filter #11의 A/D 변환값을 필터링한 값 (bit)
		5th word : butterworth filter #12의 A/D 변환값을 필터링한 값 (bit) (6축 F/T sensor의 경우)
		6th word : butterworth filter #13의 A/D 변환값을 필터링한 값 (bit) (6축 F/T sensor의 경우)
		7th word : DC_BUS_VOLT_filtered (65 bit/V) (6축 F/T sensor가 아니면 5th word 임)
	comment	F/T 센서인 경우 butterworth filter #8-#11(6축 F/T 센서의 경우는 #8-#13)는 bridge amplifier #1-#4(6
		축 F/T 센서의 경우는 #1-#6)의 전압값을 필터링함.



명령어	구분	설명
	operation	F/T sensor에서 bridge amplifier #1-#4의 offset 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가됨)
		QbHBLBHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
		CS 는 앞선 10 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
	return value	1st word : bridge amplifier #1의 offset 값 (bit)
		2nd word : bridge amplifier #2의 offset 값 (bit)
;Qb2; (+#^)		3rd word : bridge amplifier #3의 offset 값 (bit)
		4th word : bridge amplifier #4의 offset 값 (bit)
		5th word : bridge amplifier #5의 offset 값 (bit) (6축 F/T sensor의 경우)
		6th word : bridge amplifier #6의 offset 값 (bit) (6축 F/T sensor의 경우)
		7th word : DC_BUS_VOLT_filtered (65 bit/V) (6축 F/T sensor가 아니면 5th word 임)
	comment	F/T 센서인 경우에만 적용됨.
		offset 값은 SI 명령어로 설정이 가능하며, 필터 시정수를 크게하여 일정시간 bridge amplifier #1-#4(6축
		F/T 센서의 경우는 #1-#6))의 전압값을 filtering한 값으로 설정됨.



명령어	구분	설명
	operation	F/T sensor에서 계산된 모멘트값과 힘값을 읽어서 host로 전송함.
		QbHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
		CS 는 앞선 8 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
		1st word : X-moment (0.01Nm)
	return value	2nd word : Y-moment (0.01Nm)
;Qb3; (+#^)	return varue	3rd word : Z-force (0.1N)
, QDO , (T#)		4th word : DC_BUS_VOLT_filtered (65 bit/V) or 첫번째 bridge-amplifier A/D 변환값을 필터링하고
		offset을 제거한 값 or X-force (0.1N, 6축 F/T 센서의 경우)
		5th word : Y-force (0.1N, 6축 F/T 센서의 경우)
		6th word : Z-moment (0.01Nm, 6축 F/T 센서의 경우)
	comment	F/T 센서인 경우에만 적용됨.
		주의 : Qb3; 명령에 대한 마지막 리턴값은 첫번째 amplifier와 나머지 3 amplifer로 분리하여 사용할때,
		DC_BUS_VOLT_filtered/ 값이 아니고 첫번째 amplifier 출력의 A/D 변환값에서 offset을 뺀 값임.
	operation	F/T sensor의 D/A 출력 전압(bridge amplifier의 영점 조정용)을 A/D 변환한 값을 읽어서 host로 전송함.
		QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
	return value	CS 는 앞선 8 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
		1st word : bridge amplifier #1의 영점조정용 D/A 출력전압의 A/D 변환값
;Qb4; (+#^)		2nd word : bridge amplifier #2의 영점조정용 D/A 출력전압의 A/D 변환값
		3rd word : bridge amplifier #3의 영점조정용 D/A 출력전압의 A/D 변환값
		4th word : bridge amplifier #4의 영점조정용 D/A 출력전압의 A/D 변환값
		5th word : bridge amplifier #5의 영점조정용 D/A 출력전압의 A/D 변환값 (6축 F/T 센서의 경우)
		6th word : bridge amplifier #6의 영점조정용 D/A 출력전압의 A/D 변환값 (6축 F/T 센서의 경우)
	comment	F/T 센서인 경우에만 적용됨.



명령어	구분	설명
	operation	F/T sensor에서 모멘트값과 힘값을 계산한 후, 추가 필터링한 값을 읽어서 host로 전송함.
		QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
		CS 는 앞선 8 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
		1st word : X-moment의 추가필터링값 (0.01Nm)
	* a *	2nd word : Y-moment의 추가필터링값 (0.01Nm)
	return value	3rd word : Z-force의 추가필터링값 (0.1N)
.Obc. (1#A)		4th word : DC_BUS_VOLT_filtered (65 bit/V) or 첫번째 bridge-amplifier A/D 변환값을 필터링하고
;Qb5; (+#^)		offset을 제거한 후 추가필터링값 값 or X-force의 추가필터링값 (0.1N, 6축 F/T 센서의 경우)
		5th word : Y-force의 추가필터링값 (0.1N, 6축 F/T 센서의 경우)
		6th word : Z-moment의 추가필터링값 (0.01Nm, 6축 F/T 센서의 경우)
		F/T 센서인 경우에만 적용됨.
		추가필터의 시정수는 SF9,xxxx,xxxx; 명령어로 설정하고 EsA55A; 명령어로 EEPROM에 저장할수 있음.
	comment	주의 : Qb3; 명령에 대한 마지막 리턴값은 첫번째 amplifier와 나머지 3 amplifer로 분리하여 사용할때,
		DC_BUS_VOLT_filtered/ 값이 아니고 첫번째 amplifier 출력의 A/D 변환값에서 offset을 뺀 값임.
		2012년 3월 9일 S/W 버젼부터 사용가능한 명령임.
	operation	모터제어기에서 모터1/2의 순간 위치명령값과 현재 위치값을 읽어서 host로 전송함.
		QbHBLBHBLBHBLBCS;
		HBLB 는 16bit word의 상위바이트 한개, 하위바이트 한개를 각각 binary 형태로 표현한 값.
;Qb9; (+#^)		CS 는 앞선 8 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
	return value	1st word : (모터1의 순간 위치명령값 - 500000)
		2nd word : (모터1의 현재 위치값 - 5000000)
		3rd word : (모터2의 순간 위치명령값 - 5000000)
		4th word : (모터2의 현재 위치값 - 5000000)
	comment	



명령어	구분	설명
	operation	Butterworth filter 8-11(6축 F/T 센서의 경우는 8-13)의 필터링된 값을 읽어서 host로 전송함.
		RS485의 경우 : bHbMbLbHbMbLbHbMbLbHbMbLbCS; (6축 F/T 센서의 경우는 3 byte가 추가됨)
		CAN의 경우: bHbMbLbHbMbLbHbMbLbHbMbLb; (6축 F/T 센서의 경우는 3 byte가 추가됨)
		HbMbLb 는 binary 형태의 12bit word를 가리키며 HbMbLbHbMbLb 로서 2개의 12bit가 모이면 8bit
		단위의 3byte가 됨.
		♡ 는 앞선 6 byte의 합의 inverse를 binary값 한 바이트로 표현한 것임.
;b; (+#^%)	return value	1st 12-bit : butterworth filter #8의 필터링된 값
		2nd 12-bit : butterworth filter #9의 필터링된 값
		3rd 12-bit : butterworth filter #10의 필터링된 값
		4th 12-bit : butterworth filter #11의 필터링된 값
		5th 12-bit : butterworth filter #12의 필터링된 값 (6축 F/T 센서의 경우)
		6th 12-bit : butterworth filter #13의 필터링된 값 (6축 F/T 센서의 경우)
	comment	Butterworth filter의 시정수를 바꾸려면 SF8,dddd1,dddd2; 명령을 사용하면 됨.

명령어	구분	설명
	operation	통신에러횟수를 읽어서 host로 전송함.
		QEHLHL,HLHL,HLHL,HLHL; (LF2406A의 경우)
		QEHLHL,HLHL,HLHL,HLHL; (F2811/F28334의 경우)
		1st word : Pa_receive_error_cnt (회)
	return value	2nd word : Pp_receive_error_cnt (회)
	return varue	3rd byte : PP_command_update_skip_cnt (회)
		4th word : RS232C or RS485(F2811/F28334는 해당없음) 통신에러 횟수 (회)
		5th word : CAN 통신에러 횟수 (회)
		6th word : RS485(LF2406 은 해당없음) 통신에러 횟수 (회)
;QE; (+#^%)	comment	1st word는 Pa 명령어를 수신했을 때 잘못된 checksum으로 수신되어, 값을 무시하고 Pa 명령어 끝에 :으로
, QL, (III /0)		회신한 횟수를 가리킴.
		2nd word는 Pp 명령어를 수신했을 때 잘못된 checksum으로 수신되어, 값을 무시하고 Pp 명령어 끝에 :으로
		회신한 횟수를 가리킴.
		3rd byte는 관절제어 PCB로 주어진 주기로 Pp 명령어가 보내어지지 않고 건너뛴 횟수를 가리킴.
		4th word는 RS232C or RS485(F2811/F28334는 해당없음) 통신도중에 RX-ERROR/BRKDKT/FE/0E/PE 에러가 발생
		하여 시리얼 통신 모듈을 자동으로 초기화한 횟수를 가리킴.
		5th word는 CAN 통신도중에 FE/BE/SA1/CRCE/SE/ACKE/BO/EP/EW/bus-off 에러가 발생하여 CAN 통신 모듈을
		자동으로 초기화한 횟수를 가리킴.
		6th word는 RS485(LF2406은 해당없음) 통신도중에 RX-ERROR/BRKDKT/FE/0E/PE 에러가 발생하여 시리얼 통신
		모듈을 자동으로 초기화한 횟수를 가리킴.



명령어	구분	설명
	operation	통신에러횟수를 읽어서 host로 전송함.
		QEEHLHL, HLHL, HLHL, HLHL;
		1st word : RS232C 통신에러 횟수 (회)
	roturn volue	2nd word : CAN 통신에러 횟수 (회)
	return value	3rd word : RS485 통신에러 횟수 (회)
		4th word : RS485 또는 CAN 통신의 수신에러 횟수 (회)
		5th word : RS485 또는 CAN 통신의 송신에러 횟수 (회)
	comment	1st word는 RS232C 통신도중에 RX-ERROR/BRKDKT/FE/OE/PE 에러가 발생하여 시리얼 통신 모듈을 자동으로
		초기화한 횟수를 가리킴.
;QEE; (+#^)		2nd word는 CAN 통신도중에 FE/BE/SA1/CRCE/SE/ACKE/BO/EP/EW/bus-off 에러가 발생하여 CAN 통신 모듈을
		자동으로 초기화한 횟수를 가리킴.
		3rd word는 RS485 통신도중에 RX-ERROR/BRKDKT/FE/0E/PE 에러가 발생하여 시리얼 통신 모듈을 자동으로 초
		기화한 횟수를 가리킴.
		4th word의 하위바이트는 RS485 또는 CAN 통신도중에 수신시간 초과에러가 발생한 횟수를 가리킴.
		상위바이트는 2nd word나 3rd word의 값과 같음. (단 bus-off error는 제외)
		5th word의 하위바이트는 RS485 통신도중에 TXEN 신호의 high 상태가 시간 초과되어 발생한 에러 횟수
		또는 CAN 통신도중에 송신시간 초과에러가 발생한 횟수를 가리킴.
		상위바이트는 RS485 통신도중에 송신시간 초과에러가 발생한 횟수를 가리킴. (단 TXEN 신호의
		high 상태 시간초과 에러는 제외)



명령어	구분	설명
		모터 1/2의 순간목표/명령/현재변위, 엔코더카운터값, 제어오차등을 초기값으로 재설정함.
;QEA55A; (+#^%)	operation	1st word : 모터1의 변위 기준값 5,000,000에 더해지는 변위 offset 값이며 FFFF는 -1에 해당함.
		2nd word : 모터2의 변위 기준값 5,000,000에 더해지는 변위 offset 값이며 FFFF는 -1에 해당함.
or	return value	QEA55A;
or		변위의 초기값은 5,000,000 4체배 pulse이며 엔코더카운터값 및 제어오차의 초기값은 0, 4체배 pulse임.
;QEA55A,HLHL,HLH		순간목표변위값(positionx_set), 명령변위값(positionx_cmd), 현재변위값(positionx)임.
L; (+#^)	comment	절대각센서를 사용하는 경우는 명령의 사용이 불가함.
L, (T#)		변위 offset 값은 2012년 4월 10일 S/W 버젼부터 적용되며, 일부 digital/analog Hall 센서만을 사용하는
		BLDC 모터의 경우는 문제가 있을수 있음.
	operation	모터 1의 순간목표/명령/현재변위, 엔코더카운터값, 제어오차등을 초기값으로 재설정함.
;QEA5A5; (+#^%)	operation	1st word : 모터1의 변위 기준값 5,000,000에 더해지는 변위 offset 값이며 FFFF는 -1에 해당함.
	return value	QEA5A5;
or		변위의 초기값은 5,000,000 4체배 pulse이며 엔코더카운터값 및 제어오차의 초기값은 0, 4체배 pulse임.
		순간목표변위값(positionx_set), 명령변위값(positionx_cmd), 현재변위값(positionx)임.
;QEA5A5,HLHL;	comment	절대각센서를 사용하는 경우는 명령의 사용이 불가함.
(+#^)		변위 offset 값은 2012년 4월 10일 S/W 버젼부터 적용되며, 일부 digital/analog Hall 센서만을 사용하는
		BLDC 모터의 경우는 문제가 있을수 있음.
	operation	모터 2의 순간목표/명령/현재변위, 엔코더카운터값, 제어오차등을 초기값으로 재설정함.
;QE5A5A; (+#^%)		1st word : 모터2의 변위 기준값 5,000,000에 더해지는 변위 offset 값이며 FFFF는 -1에 해당함.
	return value	QE5A5A;
or		변위의 초기값은 5,000,000 4체배 pulse이며 엔코더카운터값 및 제어오차의 초기값은 0, 4체배 pulse임.
		순간목표변위값(positionx_set), 명령변위값(positionx_cmd), 현재변위값(positionx)임.
;QE5A5A,HLHL;	comment	절대각센서를 사용하는 경우는 명령의 사용이 불가함.
(+#^)		변위 offset 값은 2012년 4월 10일 S/W 버젼부터 적용되며, 일부 digital/analog Hall 센서만을 사용하는
		BLDC 모터의 경우는 문제가 있을수 있음.



명령어	구분	설명
	operation	현재 변위값을 읽어서 host로 전송함.
		QPHLHLHLHL;
;QP; (+#^%)	return value	1st long word : motor1의 현재 변위값 (4체배 pulse, 32bit)
		2nd long word : motor2의 현재 변위값 (4체배 pulse, 32bit)
	comment	엔코더카운터값을 읽어서 32bit long word로 변환한 값을 읽어냄.
	operation	순간목표변위값을 읽어서 host로 전송함.
		QpHLHLHLHL, HLHLHLL;
;Qp; (+ # ^%)	return value	1st long word : motor1의 순간목표변위값 (4체배 pulse, 32bit)
, Qp, (T# /0)		2nd long word : motor2의 순간목표변위값 (4체배 pulse, 32bit)
	comment	순간목표변위값(positionx_set)을 읽어서 32bit long word로 변환한 값을 읽어냄.
		순간목표변위값은 1ms의 제어주기마다 설정되는 순간순간의 목표변위값을 가리킴.
	operation	option으로 부착된 엔코더카운트값을 읽어서 host로 전송함.
		QqHLHLHLHL, HLHLHLL;
;Qq; (+#^)	return value	1st long word : motor1의 현재 엔코더카운트값값 (4체배 pulse, 32bit)
		2nd long word : motor2의 현재 엔코더카운트값값 (4체배 pulse, 32bit)
	comment	option으로 부착된 엔코더카운터값을 읽어서 32bit long word로 변환한 값을 읽어냄.



명령어	구분	설명
	operation	현재 속도값을 읽어서 host로 전송함.
	return value	QViiii1,iiii2; (F2406인경우)
		QVIIII1,IIII2; (F2811/F28334인 경우)
		1st integer: 모터1의 현재 속도값 (rpm)
		2nd integer : 모터2의 현재 속도값 (rpm)
;QV; (+#^%)	comment	1msec 동안 입력된 엔코더 펄스수를 1개의 ring buffer에 1ms 마다 저장하고, 이를 1ms 마다 모두 더함으
		로써, 10msec 동안 입력된 엔코더 펄스수를 RPM으로 환산한 값임.
		4체배된 4096 pulse의 엔코더 경우에 1000rpm 이면 683 pulse 가 10ms 동안에 입력됨.
		이때 속도값의 분해능은 1.465rpm 이며, 시정수는 6ms 정도임.
		iiii1,iiii2는 부호가 있는 5자리 십진수임.
		1, 2는 부호가 있는 7자리 십진수임.
	operation	현재의 공급전압과 전압강하의 크기를 읽어서 host로 전송함.
	return value	QvHLHL;
;Qv; (+#^)		1st word: 현재의 filtering된 모터용 공급전압 크기 (0.01V)
		2nd word : 현재의 filtering된 모터용 공급전압의 전압강하 크기 (0.01V)
	comment	공급전압에 대한 필터의 시정수는 1ms 임. 공급전압을 IR2175S로 측정하는 경우는 2ms.



명령어	구분	설명
	operation	모터에 흐르는 filtering된 상전류값을 읽어서 host로 전송함.
		QCiiii1,iiii2; or QCiiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) or
		QCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) or
		QCiiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)
		1st integer : motor1 전류값 (bit) (STEP형은 coil1, RJM_VER3(BLDC)/7은 motor1의 +측 공급전류)
	return value	2nd integer : motor2 전류값 (bit) (STEP형은 coil2, RJM_VER3(BLDC)/7은 motor2의 +측 공급전류)
		3rd integer : coil3 전류값 (bit) (STEP형은 coil3)
		4th integer : coil4 전류값 (bit) (STEP형은 coil4)
		5rd integer : coil5 전류값 (bit) (STEP형은 coil5)
;QC; (+#^%)		6th integer : coil6 전류값 (bit) (STEP형은 coil6)
, QO, (1# 70)	comment	1A×25mΩ×11(앰프 증폭률) = 0.275V이므로 3.3/3.0V는 3.3/3.0V÷0.275V = 12/10.9A에 해당하며
		85.25/375.375 bit/A. (DC ver1.1)
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3/3.0V는 3.3/3.0V÷0.25V = 13.2/12A에 해당하며
		77.5/341.25 bit/A. (DC V1.2부터)
		1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3/3.0V는 3.3/3.0V÷0.1V = 33/30A에 해당하며 31/136.5
		bit/A. (Piggyback-DC/RJM_VER8부터)
		1A×5mΩ×50(앰프 증폭률)×(3.3/5) = 0.165V이므로 3.3/3.0V는 3.3/3.0V÷0.165V = 20/18.18A에 해당하
		며 51.15/225.225 bit/A. (Piggyback-BLDC형)
		1A×10mΩ×50(앰프 증폭률) = 0.5V이므로 3.3/3.0V는 3.3/3.0V÷0.5V = 6.6/6A에 해당하며 155/682.5
		bit/A. (mini-DC)



명령어	구분	설명
		1A×20(10/10)mΩ×50(앰프 증폭률) = 1.0/0.5/0.5V이므로 3.3/5.0/5.0V는 3.3/5.0/5.0V÷1.0/0.5/0.5V = 3.3/10/10A에 해당하며 310/409.5/409.5 bit/A. (STEP형) 1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 5.0V는 5.0V÷0.25V = 20A에 해당하며 204.75 bit/A. (Sensor-BLDC)
	comment	TMS320LF2406A 2상 STEP형은 motor1(coil1,coil2)/motor2(coil3,coil4) 또는 motor1(coil1,coil2,coil3)/solenoid(coil4)의 조합으로 사용이 가능하며, 3상 STEP형은 motor1(coil1,coil2,coil3)/motor2(coil4,coil5,coil6) 으로 코일이 사용됨. iiii1,iiii2,iiii3,iiii4,iiii5,iiii6은 부호가 있는 십진수임. SF 명령을 사용하여 필터 시정수를 조정할 수 있으며, 사용 필터 채널은 상전류값 순서대로 0/1/2/3/4/5/임.
		2008년 6월 이후의 모든 F2811/F28334 H/W 버젼에서는 다음과 같이 통일 되었음. 2mΩ인 경우는 81.9/81.9 bit/A (RJM_VER3 : IRMCK201을 사용한 BLDC제어기, RJM_VER8 : 고출력 DC) 5mΩ인 경우는 204.75/204.75 bit/A (RJM_VER7 : IRMCK201을 사용하지 않은 BLDC제어기) 10mΩ인 경우는 409.5/409.5 bit/A (RJM_VER4 : mini DC, RJM_VER5 : step motor) IRMCK201을 사용한 경우는 IRMCK201의 read group Iq 값이며, 4096일 때 정격전류값(RMS치)에 해당함. 2009년 7월15일 S/W 버젼부터는 IRMCK201을 사용한 경우도 전류감지저항에 따라서 scale factor가 정해짐.



명령어	구분	설명
	operation	모터에 흐르는 filtering된 상전류값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.
		Qciiii1,iiii2; or Qciiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) or
		Qciiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) or
		Qciiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)
		1st integer : motor1 코일 전류값 (0.01A) (STEP형은 coil1, RJM_VER3(BLDC)/7은 motor1의 +측 공급전류)
	return value	2nd integer : motor2 코일 전류값 (0.01A) (STEP형은 coil2, RJM_VER3(BLDC)/7은 motor2의 +측 공급전
	return varue	류)
		3rd integer : coil3 전류값 (0.01A) (STEP형은 coil3)
		4th integer : coil4 전류값 (0.01A) (STEP형은 coil4)
;Qc; (+#^%)		5rd integer : coil5 전류값 (0.01A) (STEP형은 coil5)
, ας, (1π /0)		6th integer : coil6 전류값 (0.01A) (STEP형은 coil6)
	comment	TMS320LF2406A 2상 STEP형은 motor1(coil1,coil2)/motor2(coil3,coil4) 또는
		motor1(coil1,coil2,coil3)/solenoid(coil4)의 조합으로 사용이 가능하며, 3상 STEP형은
		motor1(coil1,coil2,coil3)/motor2(coil4,coil5,coil6) 으로 코일이 사용됨.
		iiii1,iiii2,iiii3,iiii4,iiii5,iiii6은 부호가 있는 십진수임.
		SF 명령을 사용하여 필터 시정수를 조정할 수 있으며, 사용 필터 채널은 상전류값 순서대로 0/1/2/3/4/5/
		임.
		2009년 7월 15일 S/W 버젼부터 Z3 명령어로 전류감지저항값이 정상적으로 설정되어 있어야함.
		2009년 7월 15일 S/W 버젼부터 IRMCK201을 사용한 경우는 Zc 명령어로 nominal current가 정상적으로 설정
		되어 있어야함.



명령어	구분	설명
	operation	모터에 흐르는 filtering된 총전류값을 읽어서 host로 전송함.
	return value	QKiiii1,iiii2;
		스케일은 QC 명령에 준함.
		SF 명령을 사용하여 필터 시정수를 조정할 수 있으며, 사용 필터 채널은 총전류값 순서대로 14/15임.
;QK; (+ #^)		F2811/F28334 에서만 가능한 명령임.
	comment	
	Commone	2008년 6월 이후의 모든 F2811/F28334 H/W 버젼에서는 다음과 같이 통일 되었음.
		2mΩ인 경우는 81.9/81.9 bit/A (RJM_VER3 : IRMCK201을 사용한 BLDC제어기, RJM_VER8 : 고출력 DC)
		5mΩ인 경우는 204.75 <mark>/204.75</mark> bit/A (RJM_VER7 : IRMCK201을 사용하지 않은 BLDC제어기)
		10mΩ인 경우는 409.5 <mark>/409.5</mark> bit/A (RJM_VER4 : mini DC, RJM_VER5 : step motor)
	operation	모터에 흐르는 filtering된 총전류값을 읽어서 암페어값으로 환산하여 host로 전송함.
	return value	Qkiiii1,iiii2;
		스케일은 Qc 명령에 준함.
;Qk; (+#^)		SF 명령을 사용하여 필터 시정수를 조정할 수 있으며, 사용 필터 채널은 총전류값 순서대로 14/15임.
, UN, (T#T)	oommon t	F2811/F28334 에서만 가능한 명령임.
	comment	2009년 7월 15일 S/W 버젼부터 Z3 명령어로 전류감지저항값이 정상적으로 설정되어 있어야함.
		2009년 7월 15일 S/W 버젼부터 IRMCK201을 사용한 경우는 Zc 명령어로 nominal current가 정상적으로 설정
		되어 있어야함.



명령어	구분	설명
	operation	error, error 적분값, error 미분값을 읽어서 host로 전송함.
		Qe 1, 2, 3, 4, 5, 6;
:00: (1#/٩/)		1st long integer : 모터1의 위치에러 (4체배 pulse)
;Qe; (+#^%)		2nd long integer : 모터1의 위치에러 적분값 (4체배 pulse)
(Force/Moment 센	return value	3rd long integer : 모터1의 위치에러 미분값 (4체배 pulse)
서가 아닌 경우)		4th long integer : 모터2의 위치에러 (4체배 pulse)
저가 내린 경구)		5th long integer : 모터2의 위치에러 적분값 (4체배 pulse)
		6th long integer : 모터2의 위치에러 미분값 (4체배 pulse)
	comment	단위는 4체배된 엔코더 펄스수임.
	operation	X-moment, Y-moment, Z-force, extra load cell값에 대하여 추가 필터링한 값을 읽어서 host로 전송함. (6
	operation	축의 경우는 X-moment, Y-moment, Z-force, X-force, Y-force, Z-moment 임)
		Qeiiii0,iiii1,iiii2,iiii3; or
	return value	Qeiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)
		<mark>iiii0</mark> : X-moment의 추가필터링값 (0.01Nm 단위)
		<mark>iiii1</mark> : Y-moment의 추가필터링값 (0.01Nm 단위)
;Qe; (+#^)		iiii2 : Z-force의 추가필터링값 (0.1N 단위)
		iiii3 : 첫번째 amplifier에 연결된 load cell의 증폭후 A/D 변환값의 추가필터링값 - offset (bit 단위,
(Force/Moment 센		4095일때 3V) or X-force의 추가필터링값 (6축 F/T 센서의 경우)
서인 경우)		<mark>iiii4</mark> : Y-force의 추가필터링값 (6축 F/T 센서의 경우)
		<mark>iiii5</mark> : Z-moment의 추가필터링값 (6축 F/T 센서의 경우)
	comment	FT 센서용 amp 회로인 경우에만 해당됨.
		추가필터의 시정수는 SF9,xxxx,xxxx; 명령어로 설정하고 EsA55A; 명령어로 EEPROM에 저장할수 있음.
		iiii3는 6축 F/T 센서가 아닌 경우로서 외장형일때 첫번째 amplifier와 나머지 3개의 amplifier를 분리하
		여 사용하는 경우에만 의미있는 값이며, 이렇게 사용하는 것은 주문 사양임.
		2012년 3월 9일 S/W 버젼부터 사용가능한 명령임.



명령어	구분	설명
	operation	filter 출력값을 읽어서 host로 전송함.
		QFiiii0,iiii1,iiii2,iiii3,iiii4,iiii5,iiii6,iiii7;
		Oth integer : CMPR1의 filtering된 값 (bit, motor1의 PWM 출력값)
		1st integer : CMPR3의 filtering된 값 (bit, motor1의 전류 제한값 또는 설정값)
		2nd integer : CMPR2의 filtering된 값 (bit, motor2의 PWM 출력값)
		3rd integer : T1CMPR의 filtering된 값 (bit, motor2의 전류 제한값 또는 설정값)
		4th integer : A/D 변환한 MOTOR1의 전류치를 FILTERING한 값 (bit, DC V1.1 : 85.25/375.375 bit/A,
		DC V1.2부터 : 77.5/341.25 bit/A, Piggyback-DC/RJM_VER8부터 : 31/136.5 bit/A,
		BLDC : 51.15/225.225 bit/A, mini-DC : 155/682.5 bit/A, STEP : 310/682.5 bit/A,
		sensor BLDC: 204.75 bit/A)
	return value	5th integer : A/D 변환한 MOTOR2의 전류치를 FILTERING한 값 (bit, DC V1.1 : 85.25/375.375 bit/A,
;QF; (+#^%)		DC V1.2부터 : 77.5/341.25 bit/A, Piggyback-DC/RJM_VER8부터 : 31/136.5 bit/A,
		BLDC : 51.15/225.225 bit/A, mini-DC : 155/682.5 bit/A, STEP : 310/682.5 bit/A,
		sensor BLDC : 204.75 bit/A)
		6th integer : 모터용 공급전압값을 FILTERING한 값 (bit)
		(DC V1.2이전 : 414/956 bit,
		DC V1.2부터/mini-DC/Piggyback-DC : 281.8/650 bit/10V,
		Piggyback—BLDC: 724.8 bit/10V
		Sensor-BLDC: 638.9 bit/10V)
		7th integer: filtered value = butterworth filter(filter number, value to be filtered):
	comment	filtered_value = butterworth_filter(filter_number, value_to_be_filtered); 의 형식으로 사용하면 됨.
	COMMENT	의 성적으로 자용하면 됨. filter_number는 0 - 7중의 하나를 선택하며, 한가지 신호에 한개의 필터번호만 사용하여야함.



명령어	구분	설명
	operation	X-moment, Y-moment, Z-force, extra load cell를 읽어서 host로 전송함. (6축의 경우는 X-moment,
	operation	Y-moment, Z-force, X-force, Y-force, Z-moment 임)
		Qfiiii0,iiii1,iiii2,iiii3; or
		Qfiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)
		iiii0 : X-moment (0.01Nm 단위)
		iiii1 : Y-moment (0.01Nm 단위)
	return value	iiii2 : Z-force (0.1N 단위)
;Qf; (+#^)		iiii3 : 첫번째 amplifier에 연결된 load cell의 증폭후 A/D 변환값 - offset (bit 단위, 4095일때 3V) or
		X-force (6축 F/T 센서의 경우)
		iiii4 : Y-force (6축 F/T 센서의 경우)
		iiii5 : Z-moment (6축 F/T 센서의 경우)
		FT 센서용 amp 회로인 경우에만 해당됨.
	aamman t	iiii3는 6축 F/T 센서가 아닌 경우로서 외장형일때 첫번째 amplifier와 나머지 3개의 amplifier를 분리하
	comment	여 사용하는 경우에만 의미있는 값이며, 이렇게 사용하는 것은 주문 사양임.
		삼성의료기의 FT 센서인 경우는 무조건 4개의 값만 host로 전송됨.



명령어	구분	설명
	operation	모터1과 모터2의 input power, output power, output torque, efficiency를 읽어서 host로 전송함.
		QWiiii0,iiii1,iiii2,,iiii5,iiii6,iiii7;
		iiii0 : 모터1의 input power (0.1W 단위)
		iiii1 : 모터1의 output power (0.1W 단위)
		iiii2 : 모터1의 output torque (mNm 단위)
	return value	iiii3 : 모터1의 efficiency (% 단위)
		iiii4 : 모터2의 input power (0.1W 단위)
;QW; (+#^)		iiii5 : 모터2의 output power (0.1W 단위)
		iiii6 : 모터2의 output torque (mNm 단위)
		iiii7 : 모터2의 efficiency (% 단위)
		사전에 Z3 명령에 의하여 전류감지 저항값을 설정하여야 하며, ZQ/ZR 명령어에 의하여 모터의 특성파라메
	comment	터가 설정되어 있어야 하며, SX 명령어로 bit3(밑에서 4번째 비트)가 세트되어 있어야 계산이 실시됨.
		IRMCK201을 사용한 RJM_VER3의 경우는 Zc 명령에 의한 nominal current 값도 설정되어 있어야함.
		efficiency는 시정수 8ms의 필터링이 된 값임.
		IRMCK201을 사용한 RJM_VER3의 경우는 시정수 8ms의 필터링이 된 전류치를 사용함.



명령어	구분	설명
	operation	모터1과 모터2의 loss power, winding temperature, housing temperature, ambient temperature를 읽어서 host로 전송함.
;Qw; (+#^)	return value	Qwiiii0,iiii1,iiii2,,iiii5,iiii6,iiii7; iiii0 : 모터1의 손실전력 (0.1W 단위) iiii1 : 모터1의 권선온도 (0.1도 단위) iiii2 : 모터1의 모터케이스온도 (0.1도 단위) iiii3 : 모터1의 주변온도 (0.1도 단위) iiii4 : 모터2의 손실전력 (0.1W 단위) iiii5 : 모터2의 권선온도 (0.1도 단위) iiii6 : 모터2의 모터케이스온도 (0.1도 단위) iiii7 : 모터2의 주변온도 (0.1도 단위)
	comment	사전에 Z3 명령에 의하여 전류감지 저항값을 설정하여야 하며, ZQ/ZR 명령어에 의하여 모터의 특성파라메터가 설정되어 있어야 하며, SX 명령어로 bit3(밑에서 4번째 비트)가 세트되어 있어야 계산이 실시됨. IRMCK201을 사용한 RJM_VER3의 경우는 Zc 명령에 의한 nominal current 값도 설정되어 있어야함. 주변온도는 0도로 세팅되어 있음.



명령어	구분	설명
	operation	faultBufPointer, faultBuf[0-7]을 읽어서 host로 전송함.
		QJHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL;
	return value	1st word : faultBufPointer [0-7]
;QJ; (+#^)		2nd word - 9th word : faultBuf[0] - faultBuf[7]
		전원을 켠 이후에 발생한 오류에 대하여 flash 코드를 저장한 것임. 전원이 차단되면 이 flash 코드는
	comment	사라짐. 2012년 2월 10일 S/W 버젼부터 사용가능한 명령임.
		2012년 8월 18일 이전 S/W 버젼에서는 오류코드의 값이 >>4 되어서 리턴되는 오류가 있었음.
	operation	faultBufPointer_save, faultBuf_save[0-14]을 읽어서 host로 전송함.
		QjHLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH
	return value	1st word : faultBufPointer_save [0-14]
		2nd word - 16th word : faultBuf_save[0] - faultBuf_save[14]
;Qj; (+#^)		오류가 발생한 경우, flash 코드를 오류발생 순서대로 EEPROM에 저장한 값임. 따라서 과거의 flash 코
, (c), (Tir)		드도 가장 최근 것이 전원 투입/차단 횟수에 관계없이 남아 있음. 에러코드의 상위 8비트가 0 일때는
	comment	이번의 전원투입이후에 발생한 오류코드임을 의미하며, 0이 아닌값이면 클수록 더 이전에 발생한 오류코드
	Commert	임을 의미함. 2012년 2월 10일 S/W 버젼부터 사용가능한 명령임.
		2012년 8월 18일 이전 S/W 버젼에서는 오류코드의 값이 >>4 되어서 리턴되는 오류가 있었음.
		설정된 값은 EEPROM에 저장된 값임.
	operation	faultBufPointer_save, faultBuf_save[0-14]을 0으로 clear함.
;QjA55A; (+#^)		QjHLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH
	return value	1st word : faultBufPointer_save [0-14]
, wj noon, (T#)		2nd word - 16th word : faultBuf_save[0] - faultBuf_save[14]
	comment	2012년 2월 10일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
;Qddddd1; (+ #^%)	operation	Sleep_1micro(dddd1), Sleep_10micro(dddd1), Sleep_msec(1) 함수실행결과의 지연시간동안 CS_EEPROM_N_port에 low pulse를 출력함. 1st word : 지연시간 [0-100]
	return value	Qddddd1;
	comment	2406에서는 Sleep_1micro(dddd1) 지연동작은 하지 않음.
	operation	diagnosis mode를 개시함.
;QD; (+#^%)	return value	QD;
	comment	특별한 조치는 없음.
;QX; (+ #^ %)	operation	diagnosis mode를 종료함.
	return value	QX;
	comment	특별한 조치는 없음.

명령어	구분	설명
		1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는 8word를 host PC로 전송하는 데이터의 종류를 선택
	operation	하는 변수인 data_logging_select 변수값을 설정함.
		1st word : data_logging_select 변수값
	return value	Qxdddd1;
.0		data_logging_select 값에 따라서 host PC로 전송하는 값을 선택하고 보낼 데이터를 SCITXO[]에 저장하는
;Qxdddd1; (+#^%)		것은 cpu_joio.c에 이미 반영되어 있어야함.
.00. (1808)		data_logging_select 값에 따라 전송되는 변수값은 다음 페이지에 list되었음.
;Qx?; (+ # ^%)		2012년 1월 07일 S/W 버젼부터는 data_logging_select 값이 4096~8191인 경우는 0~4095인 경우와 같은 데
	comment	이터가 전송되는데, 4096~6143일때는 4word의 데이터를 2ms 마다 13 byte로 나누어서 보내며, 6144~8191일
		때는 8word의 데이터를 2ms 마다 16 byte로 나누어서 보냄.
		2012년 6월 16일 S/W 버젼부터는 data_logging_select 값이 500~599인 경우는 20KHz ~ 100KHz interrupt
		routine에서 4096word를 capture하며, PC로의 전송은 Qx9; 에 의하여 개시됨. 단 이 기능은 F/T 센서의
		경우는 해당없으며, Qx6;로 바뀌어서 실행됨.
	operation	1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는 8word를 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
	return value	none.
		전송되는 데이터는 Qxy; 명령을 전송해온 RS232C/RS485/CAN 채널로만 전송됨.
	comment	모터제어기의 RS232C 채널이 115200 baud rate의 전송속도로 설정이 되어 있을 때만 유효함.
;Qxy; (+ #^ %)		전송데이터의 종류는 Qx 명령어에 의하여 선택됨.
, QXy , (T# %)		Qxy; 명령어를 한번 보내면 전송개시, 한번 더 보내면 전송을 정지함.
		4 word의 값에 해당하는 총 8바이트를 binary 형태로 separator (;) 없이 1ms 마다 host로 전송하는데, 이
		때 한 word의 값은 low-byte, high-byte 순으로 전송함.
		2012년 1월 07일 S/W 버젼부터는 data_logging_select 값에 따라서 1ms당 4word 또는 2ms당 4word 또는
		2ms당 8word를 8byte 또는 13byte 또는 16byte로 나누어서 보냄.
		2012년 12월 27일 S/W 버젼부터는 Qxyz; 명령어로도 전송을 정지할수 있음.



명령어		구분	설명
		operation	Qxy; 또는 Qz; 또는 Qzx; 또는 Qzy; 명령에 의한 모니터링 작동을 중지함.
;Qxyz; (+	#^)	return value	none.
		comment	2012년 12월 27일 S/W 버젼부터 사용가능한 명령어임.

Qx, Qz, Qzx, Qzy 에 의한 실시간 데이터 모니터링

Qx0; ~ Qx2047; 는 Qx4096; ~ Qx6143; 과 같은 값을 PC로 전송하되,

Qx0; ~ Qx2047;는 1ms당 4word를 8byte로 나누어서 기존방식으로 PC에 전송하며,

Qx4096; ~ Qx6143;는 2ms당 4word를 보내는데 한 word당 3byte로 나누어서 보내고 마지막에는 0xaa를 붙여 보내기 때문에 총 13byte를 전송함.

(첫바이트는 word의 0~5번 비트까지, 두번째바이트는 word의 6~11번 비트까지, 세번째바이트는 word의 12~15번 비트까지를 가지고 있음)

Qx6144;~Qx8191; 의 경우는 2ms당 8word를 16byte로 나누어서 기존방식으로

PC에 전송하나 현재는 Qx6144; 인 경우에만 정상적인 8word가 PC로 전송됨.

Qz; 를 보내면 1ms당 4word를 8byte로 나누어서 기존방식으로 PC에 전송하며,

Qzx; 를 보내면 2ms당 4word를 13byte로 나누어서 보내며 (데이터 형식은 Qx4096; ~ Qx6143; 의 경우와 같음)

Qzy; 를 보내면 2ms당 8word를 16byte로 나누어서 기존방식으로

PC에 전송하며 8 word의 주소는 QZ 로 설정하는 4 word와 QY로 설정하는 4word를 합해서 설정됨



세어기 명령어 LIST

Qxdddd1 에 의하여 전송되는 변수값 (RJM_VER7인 경우)

	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
0/	모터1의 순간위치명령값 (position1_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터1의 현재 위치값(position1)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터2의 순간위치명령값 (position2_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터2의 현재 위치값(position2)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	
1/	모터1의 순간속도명령값 (speed1_cmd_old / SPEED_SCALE_FACTOR, 03번 모드 에서는 (speed1_long_cmd + speed1_long_cmd_feedforward) / 10, 06/08/09/0A번 모드에서는 speed1_long_cmd_old / 10) 단위크기는 RPM	모터1의 현재 속도값 (speed1_RPM, digital Hall 센서 만을 사용하는 경우는 필터링한 값임) 단위크기는 RPM	모터2의 순간속도명령값 (speed2_cmd_old / SPEED_SCALE_FACTOR, 03번 모드 에서는 (speed2_long_cmd + speed2_long_cmd_feedforward) / 10, 06/08/09/0A번 모드에서는 speed2_long_cmd_old / 10) 단위크기는 RPM	모터2의 현재 속도값 (speed2_RPM, digital Hall 센서 만을 사용하는 경우는 필터링한 값임) 단위크기는 RPM	
2/	모터1의 전류명령값 (current1_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 U상 전류의 A/D 변환값 (current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 V상 전류의 A/D 변환값 (current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 W상 전류의 A/D 변환값 (current3_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	
3/	모터2의 전류명령값 (current2_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 U상 전류의 A/D 변환값 (current4_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 V상 전류의 A/D 변환값 (current5_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 W상 전류의 A/D 변환값 (current6_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	



QxO; 에 대한 return 값은 위치센서로 analog 또는 digital potentiometer을 사용하는 경우에 /4를 하지 않고 원래값을 보내줌. 또 S1/S2/Ts/Tt/Tu/Tv 명령에 의한 구동일 경우에도 /4를 하지 않고 원래값을 보내줌.

	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
4/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당		
5/	resolver를 사용하는 경우는 모터1의 필터링된 cos*4	모터1의 필터링된 sin*4	모터1의 AD2_12_bit_Value (원 점)	모터1의 phase_angle1_old		
5/	유도모터의 경우는 ld1_AD	Iq1_AD	Imr 1>>16	speed1_RPM		
5/	그외는 모터1의 Hall 센서 A상 전압의 A/D 변환값 (0 - 4095)	모터1의 Hall 센서 B상 전압의 A/D 변환값 (0 - 4095)	모터1의 Hall 센서 C상 전압의 A/D 변환값 (0 - 4095)	모터1의 Hall센서 A상 상승에지 기준 위상각 (0 - 16383)		
6/	resolver를 사용하는 경우는 모터1의 필터링된 cos*4	모터1의 필터링된 sin*4	모터1의 electrical_angle1_cnt>>8	모터1의 phase_angle1		
6/	유도모터의 경우는 speed1_RPM	(4096L*Iq1_AD)/(int)(Imr1>>16)	FluxRate1	electrical_angle1_cnt>>8		
6/	그외는 모터2의 Hall 센서 A상 전압의 A/D 변환값 (0 - 4095)	모터2의 Hall 센서 B상 전압의 A/D 변환값 (0 - 4095)	모터2의 Hall 센서 C상 전압의 A/D 변환값 (0 - 4095)	모터2의 Hall센서 A상 상승에지 기준 위상각 (0 - 16383)		
7/	resolver를 사용하는 경우는 모터1의 필터링된 cos*4	모터1의 필터링된 sin*4	모터1의 AD2_12_bit_Value (원 점)	모터1의 (position1 & 0x3ffL)<<4; // 위상각 (linear resolver)		
7/	그외는 모터1의 전류명령값 (current1_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 U상 전류의 A/D 변환값 을 필터링한 값 (current1_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 V상 전류의 A/D 변환값 을 필터링한 값 (current2_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 W상 전류의 A/D 변환값 을 필터링한 값 (current3_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당		

	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
8/	resolver를 사용하는 TPC_FOCUSING 경우는 모터1의 현재 위치값(position1) 을 4로 나눈값 단위크기는 0.01953125년	모터1의 현재 위치값 (position1_ENCODER)을 4로 나눈 값 단위크기는 0.01953125/4m	Hall1C 입력 전압을 필터링한 값 (모터1의 외부로부터의 전류명령 값)	phase_angle1_old + ENC1Z_port * 1000;		
8/	그외는 모터2의 전류명령값 (current2_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 U상 전류의 A/D 변환값을 필터링한 값 (current4_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 V상 전류의 A/D 변환값을 필터링한 값 (current5_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 W상 전류의 A/D 변환값을 필터링한 값 (current6_AD_filtered) 2/5/10m요의 경우 81.9/204.75/409.5 가 1A에 해당		
10/	resolver를 사용하는 경우는 모터2의 필터링된 cos*4	모터2의 필터링된 sin*4	모터2의 AD10_12_bit_Value (원 점)	모터2의 phase_angle2_old		
10/	그외는 모터1의 입력 power(Pin1) 단위크기는 0.1W	모터1의 출력 power(Pout1) 단위크기는 0.1W	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 효율 단위크기는 1%		
11/	resolver를 사용하는 경우는 모터2의 필터링된 cos*4	모터2의 필터링된 sin*4	모터2의 electrical_angle2_cnt>>8	모터2의 phase_angle2		
11/	그외는 모터2의 입력 power(Pin2) 단위크기는 0.1W	모터2의 출력 power(Pout2) 단위크기는 0.1W	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	모터2의 효율 단위크기는 1%		
12/	resolver를 사용하는 경우는 모터2의 필터링된 cos*4	모터2의 필터링된 sin*4	모터2의 AD10_12_bit_Value (원 점)	모터2의 (position2 & 0x3ffL)<<4; // 위상각 (linear resolver)		
12/	그외는 모터1의 입력 power(Pin1) 단위크기는 0.1W	모터1의 출력 토크(Tout1) 단위크기는 mN	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 효율 단위크기는 1%		



	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
13/	resolver를 사용하는 TPC_FOCUSING 경우는 모터2의 현재 위치값(position1) 을 4로 나눈값 단위크기는 0.01953125년	모터2의 현재 위치값 (position1_ENCODER)을 4로 나눈 값 단위크기는 0.01953125년	Hall2C 입력 전압을 필터링한 값 (모터2의 외부로부터의 전류명령 값)	phase_angle2_old + ENC2Z_port * 1000;	
13/	모터2의 입력 power(Pin2)	모터2의 출력 토크(Tout2)	모터2의 현재 속도값	모터2의 효율	
	단위크기는 0.1W	단위크기는 mN	(speed2_RPM) 단위크기는 RPM	단위크기는 1%	
14/	공급전압 65 가 1V 에 해당함.	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당	
15/	공급전압 65 가 1V 에 해당함.	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	모터2의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당	
16/	모터1의 손실 power(Pin1-Pout1)	모터1의 권선 온도(Twinding1)	모터1의 하우징 온도(Thousing1)	모터1의 분위기 온도(Tambient1)	
	단위크기는 0.1W	단위크기는 0.1도	단위크기는 0.1도	단위크기는 0.1도	
17/	모터2의 손실 power(Pin2-Pout2)	모터2의 권선 온도(Twinding2)	모터2의 하우징 온도(Thousing2)	모터2의 분위기 온도(Tambient2)	
	단위크기는 0.1W	단위크기는 0.1도	단위크기는 0.1도	단위크기는 0.1도	
18/	모터1의 Hall 센서 A상 전압의	모터1의 Hall 센서 B상 전압의	모터1의 Hall 센서 C상 전압의	모터1의 U상 출력 위상각	
	A/D 변환값 (0 - 4095)	A/D 변환값 (0 - 4095)	A/D 변환값 (0 - 4095)	단위크기는 1 deg	
19/	모터2의 Hall 센서 A상 전압의	모터2의 Hall 센서 B상 전압의	모터2의 Hall 센서 C상 전압의	모터2의 U상 출력 위상각	
	A/D 변환값 (0 - 4095)	A/D 변환값 (0 - 4095)	A/D 변환값 (0 - 4095)	단위크기는 1 deg	



	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
20/	motor1_control_service_time + motor2_control_service_time * 100 단위크기는 1ms의 1%	mainLoopStep_cnt 0-9 범위의 값으로서 단위크기는 1ms	mainLoop_execution_time 단위크기는 1ms의 1%	maximum_interrupt_service_time 단위크기는 PWM 주기의 1%		
21/	motor1_error 위치제어 모드에서의 단위크기는 1 encoder pulse (엔코더를 안쓰 는 경우는 16384 펄스의 엔코더로 간주함) 속도제어 모드에서의 단위크기는 0.1 rpm 전류제어 모드에서는 해당없음	motor1_integral 위치제어 모드에서의 단위크기는 1 encoder pulse-msec (엔코더를 안쓰는 경우는 16384 펄스의 엔 코더로 간주함) 속도제어 모드에서의 단위크기는 0.1 rpm-sec 전류제어 모드에서는 해당없음.	motor1_derivative 위치제어 모드에서의 단위크기는 1 encoder pulse/ms (엔코더를 안쓰는 경우는 16384 펄스의 엔 코더로 간주함) 속도제어 모드에서의 단위크기는 0.1 rpm/ms 전류제어 모드에서는 해당없음	current1_cmd (01번 위치제어모 드에서는 motor1_PWM_set, 03번 위치제어모드에서는 speed1_long_cmd) 위치/속도제어 모드에서의 단위 크기는 AD 변환된 전류값의 단위 크기와 같음 전류제어 모드에서는 해당없음		
22/	motor2_error	motor2_integral	motor2_derivative	current2_cmd (01번 위치제어모 드에서는 motor2_PWM_set, 03번 위치제어모드에서는 speed2_long_cmd)		
23/	DC_BUS_VOLT_filtered - DC_BUS_VOLT_filtered_at_power_ up 65 가 1V 에 해당함	speed1_RPM 단위크기는 RPM	DC_BUS_VOLT_filtered 65 가 1V 에 해당함	BRAKE_RELAY_port * 1000 1000 이면 on, 0 이면 off		
24/	DC_BUS_VOLT_filtered	speed1_RPM	power_resistor_ON_timer	BRAKE_RELAY_port * 1000		
25/	DC_BUS_VOLT_filtered	DC_BUS_VOLT_filtered_at_power_ up	power_resistor_ON_timer	BRAKE_RELAY_port * 1000		



	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
30/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt		
31/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation2/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current2_control_limit - current2_control_limit_cnt		
32/	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation2/1 00 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt	current2_control_limit - current2_control_limit_cnt		
33/	(fault1_status >> 8) & 0x0001	모터1의 권선 온도(Twinding1) 단위크기는 0.1도	모터1의 권선 온도 리밋 (TempLimit1) 단위크기는 0.1도	current1_control_limit - current1_control_limit_cnt		
34/	(fault1_status >> 9) & 0x0001	모터2의 권선 온도(Twinding2) 단위크기는 0.1도	모터2의 권선 온도 리밋 (TempLimit2) 단위크기는 0.1도	current2_control_limit - current2_control_limit_cnt		
35/	LS1A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS1B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS1C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS1D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT1_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)		
36/	LS2A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS2B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS2C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS2D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT2_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)		



	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
37(모 터1), 38(모 터2)/	100 +	positionx_deceleration_at_limit_flg * 1 + speedx_deceleration_at_limit_f lg * 10	speedx_cmd_old / SPEED_SCALE_FACTOR	모터x의 순간위치명령값 (positionx_set)을 4로 나눈값		
43/	fault_status	fault1_status	IRMCK201_fault_status	MOT_EN_N_port * 1000 + operation_mode		
44/	faultBufPointer	faultBuf[faultBufPointer]	faultBuf[faultBufPointer-1]	faultBuf[faultBufPointer-2]		
45/	fault_save_timer	faultBufPointer_save	faultBuf_save[faultBufPointer_save]	faultBuf_save[faultBufPointer_ save-1]		
46/	fault_status	fault_save_timer	speed1_set	operation_mode		
48/	HALL_Sensor1_A_AD_value	HALL_Sensor 1_B_AD_value	HALL_Sensor 1_C_AD_value	phase_angle1		
49/	HALL_Sensor2_A_AD_value	HALL_Sensor2_B_AD_value	HALL_Sensor2_C_AD_value	phase_angle2		
50/	DC_BUS_VOLT_filtered_at_power_up	vfiltA[10]	DC_BUS_VOLT_slow_filtered	fault_status		
51/	total_current1_AD_filtered	motor1_current_max	current1_max_limit	fault_status		
52/	total_current2_AD_filtered	motor2_current_max	current2_max_limit	fault_status		



	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
53/	motor1_current_max	current1_over_timer	current1_over_flg + current1_over_K_flg * 10	fault_status		
54/	motor2_current_max	current2_over_timer	<pre>current2_over_flg + current2_over_K_flg * 10</pre>	fault_status		
55/	current1_cmd	Iq1_AD	motor1_moving_flg * 1000 + Idref1 + (MOTORx_PA_end_processing_time _set - MOTOR1_PA_end_timer)	ld1_AD		
56/	current2_cmd	Tq2_AD	motor2_moving_flg * 1000 + Idref2 + (MOTORx_PA_end_processing_time _set - MOTOR2_PA_end_timer)	Id2_AD		
60/	TPC 초점구동의 경우는 ADC_RESULT2_reg>>2	vfiltA[2]	current1_cmd	speed1_cmd / SPEED_SCALE_FACTOR		
60/	자동화용 H/W 버젼의 경우는 ADC_RESULT14_reg>>2 또는 ADC_RESULT6_reg>>2;	vfiltA[15] or vfiltA[11]	current1_cmd	speed1_cmd / SPEED_SCALE_FACTOR		
61/	TPC 초점구동의 경우는 ADC_RESULT10_reg>>2	vfiltA[5]	current2_cmd	speed2_cmd / SPEED_SCALE_FACTOR		
61/	Hall 센서형 전류센서를 사용하 는 경우는 current_offset_measurement_tim er	CURRENT1_SEN_P_at_power_up	CURRENT1_SEN_N_at_power_up	CURRENT2_SEN_P_at_power_up		
62/	current_offset_measurement_tim er	CURRENT1_SEN_P_at_power_up	CURRENT1_SEN_N_at_power_up	TOTAL_CURRENT1_at_power_up		



	RJM_VER7인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
63/	Hall 센서형 전류센서를 사용하 는 경우는 CANRXO_Buf[250]	CANRXO_Buf[251]	CANRXO_Buf [252]	current 1_AD		
63/	그외는 current_offset_measurement_tim er	CURRENT2_SEN_P_at_power_up	CURRENT2_SEN_N_at_power_up	TOTAL_CURRENT2_at_power_up		
64/	current_offset_measurement_timer	CURRENT1_SEN_P	CURRENT1_SEN_N	total_current1_AD		
65/	Hall 센서형 전류센서를 사용하 는 경우는 CURRENT1_SEN_P	CURRENT1_SEN_N	CURRENT2_SEN_P	CURRENT1_SEN_P + CURRENT1_SEN_N + CURRENT2_SEN_P - (CURRENT1_SEN_P_at_power_up + (CURRENT1_SEN_N_at_power_up + CURRENT2_SEN_P_at_power_up)		
65/	그외는 current_offset_measurement_tim er	CURRENT2_SEN_P	CURRENT2_SEN_N	total_current2_AD		
66/	automatic_boot_up_timer	operation_mode	speed1_RPM	position1		
67/	automatic_boot_up_timer	phase_angle1	position1_cmd	position1		
68/	total_current1_AD_filtered	motor1_current_max	<pre>current1_max_limit (=max(current1_limit, current1_control_limit) * 1.25)</pre>	min(current1_limit, current1_control_limit) * 1.25		
69/	total_current2_AD_filtered	motor2_current_max	<pre>current2_max_limit (=max(current2_limit, current2_control_limit) * 1.25)</pre>	min(current2_limit, current2_control_limit) * 1.25		



	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
72/	모터1의 전류명령값 (current1_cmd) 단위크기는 0.01A	모터1의 U상 전류의 A/D 변환값 (current1_AD) 단위크기는 0.01A	모터1의 V상 전류의 A/D 변환값 (current2_AD) 단위크기는 0.01A	모터1의 W상 전류의 A/D 변환값 (current3_AD) 단위크기는 0.01A	
73/	모터2의 전류명령값 (current2_cmd) 단위크기는 0.01A	모터2의 U상 전류의 A/D 변환값 (current4_AD) 단위크기는 0.01A	모터2의 V상 전류의 A/D 변환값 (current5_AD) 단위크기는 0.01A	모터2의 W상 전류의 A/D 변환값 (current6_AD) 단위크기는 0.01A	
74/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 단위크기는 0.01A	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 단위크기는 0.01A	
88/	모터1의 03/0A번 제어모드에서 속도의 PID 공통 조정게인 * I 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 P 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 I 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 D 조정게인 * 1000	
89/	모터1의 03/0A번 제어모드에서 속도의 PID 공통 조정게인 * I 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 P 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 I 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 D 조정게인 * 1000	
91/	스텝모터의 경우는 lead_angle1_set	lead_angle1	lead_angle1_set - lead_angle1	0	
91/	그외는 current1_cmd	Iq1_AD	current1_cmd - Iq1_AD	phase_angle1	
92/	스텝모터의 경우는 speed1_cmd_old / (7L * MOTOR1_NUMBER_of_POLE_PAIR)	(speed1_RPM_count * 600000L) / MOTOR1_PULSE_PER_REV_ENC	첫번째 변수 - 두번째 변수	motor1_error	
92/	그외는 speed1_long_cmd_old 또는 (speed1_cmd_old * 10) / SPEED_SCALE_FACTOR	speed1_RPM_10	첫번째 변수 - 두번째 변수	motor1_error	

	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
93/	스텝모터의 경우는 (position1_set - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	(position1 - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor1_error	
93/	그외는 position1_set - POSITION1_MID	(position1 - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor1_error	
95/	스텝모터의 경우는 lead_angle2_set	lead_angle2	lead_angle2_set - lead_angle2	0	
95/	그외는 current2_cmd	Iq2_AD	current2_cmd - Iq2_AD	phase_angle2	
96/	스텝모터의 경우는 speed2_cmd_old / (7L * MOTOR2_NUMBER_of_POLE_PAIR)	(speed2_RPM_count * 600000L) / MOTOR2_PULSE_PER_REV_ENC	첫번째 변수 - 두번째 변수	motor2_error	
96/	그외는 speed2_long_cmd_old 또는 (speed2_cmd_old * 10) / SPEED_SCALE_FACTOR	speed2_RPM_10	첫번째 변수 - 두번째 변수	motor2_error	
97/	스텝모터의 경우는 (position2_set - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	(position2 - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor2_error	
97/	그외는 position2_set - POSITION2_MID	(position2 - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor2_error	



	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
98/	모터1의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추가조정 게인 * 1000	모터1의 P 조정게인 * 1000	모터1의 I 조정게인 * 1000	모터1의 D 조정게인 * 1000	
99/	모터2의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추가조정 게인 * 1000	모터2의 P 조정게인 * 1000	모터2의 I 조정게인 * 1000	모터2의 D 조정게인 * 1000	
101/	모터1의 U상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터1의 V상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터1의 W상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터1의 Hall센서 A상 상승에지 기준 위상각 (0 - 16383), IRS2336을 사용한 경우는 fault 시에 10000을 더함	
102/	((RJM_VER7_ENCODER & IRF7779 & IRS2336)==1)&&(ABSOLUTE_ENC== 3)인 경우는 eQEP1_POSLAT_reg_save >> 14	eQEP1_POSLAT_reg_save & 0x3fff	phase_angle1	(position1 - POSITION1_MID) / (MOTOR1_POSITION_SCALE_FACTOR * 4)	
102/	그외 (((RJM_VER7_ENCODER & IRF7779)==1) ((RJM_VER8 & IRF7779)==1))&&(ABSOLUTE_ENC== 2)인 경우는 absENCA_duty PWM 절대각센서를 사용하는 경우로서 절대각센서 A의 low인 기간의 duty rario (-800 - +800)	absENCB_duty PWM 절대각센서를 사용하는 경우 로서 절대각센서 B의 low인 기간 의 duty rario (-7500 ~ +7500)	absPOSITION1>>2 PWM 절대각센서를 사용하는 경우 로서 절대각 계산값 / 4 (4체배된 4096펄스의 엔코더와 각변위 단위가 같음)	(position1 - POSITION1_MID) / (MOTOR1_POSITION_SCALE_FACTOR * 4) PWM 절대각센서를 사용하는 경우 로서 모터1의 각변위값 / 4 (4체배된 4096펄스의 엔코더 펄 스 단위)	
102/	그외의 경우는 모터2의 U상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터2의 V상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터2의 W상 PWM 출력값 (0 - PWM_PERIOD, PWM_PERIOD/2 일때 50% duty)	모터2의 Hall센서 A상 상승에지 기준 위상각 (0 - 16383)	



	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
103/	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 Hall 센서 A상 전압의 A/D 변환값 (0 - 4095)	모터1의 U상 출력 위상각 단위크기는 1 deg	cogging_torque_compensation1 0/4096/8192/16384이면 6.25/12.5/25% reduction	
104/	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	모터2의 Hall 센서 A상 전압의 A/D 변환값 (0 - 4095)	모터2의 U상 출력 위상각 단위크기는 1 deg	cogging_torque_compensation2 0/4096/8192/16384이면 6.25/12.5/25% reduction	
105/	current1_cmd	lq1	Vq1	Vd1	
106/	current2_cmd	lq2	Vq2	Vd2	
110/	motor1_error	current1_cmd	<pre>(position1_set - POSITION1_MID) - (position1_at_1ms - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR</pre>	(position1_at_1ms - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	
111/	motor2_error	current2_cmd	(position2_set - POSITION2_MID) - (position2_at_1ms - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	(position2_at_1ms - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	
112/	EPS의 경우는 모터1의 위치명령값 (position1_cmd)을 4로 나눈값	모터1의 순간위치명령값 (position1_set)을 4로 나눈값	모터1의 현재위치값(position1) 을 4로 나눈값	모터1의 vehicle_driving_motor_speed	
113/	position1_cmd - position1_set	speed1_RPM (Hall IC only 제어 의 경우는 필터링한 값)	speed1_current	speed1_residual	



	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
114/	speed1 command	speed1_RPM (Hall IC only 제어 의 경우는 필터링한 값)	current1_cmd	(MOTOR1_INERTIAL_RATIO * (long long)motor1_acc_rate) >> 12	
115/	speed2 command	speed2_RPM (Hall IC only 제어 의 경우는 필터링한 값)	current2_cmd	(MOTOR2_INERTIAL_RATIO * (long long)motor2_acc_rate) >> 12	
116/	notch filter MOTOR1_uk1_0	notch filter MOTOR1_yk1_0	notch filter MOTOR2_uk1_0	notch filter MOTOR2_yk1_0	
508/	current1_cmd	lq1	current1_cmd - Iq1	phase_angle1_old	
509/	current2_cmd	lq2	current2_cmd - lq2	phase_angle2_old	
0x800 +32	모터1의 순간위치명령값 (position1_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터1의 위치제어 오차 단위크기는 1 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함)	모터1의 위치제어 출력값 (01번 제어모드에서는 PWM값, 02번제어 모드에서는 전류명령값, 03번 제 어모드에서는 속도명령값, 스테 핑모터에서는 lead angle 명령 값)	모터1의 속도명령값 (03/0A번 제 어모드일때로서 feedforward 성 분 포함) 단위크기는 RPM*10	
	모터1의 실제 속도값 단위크기는 RPM*10	모터1의 속도제어오차 (03/0A번 제어모드일때) 단위크기는 RPM*10	모터1의 전류명령값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 실제전류값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	
0x800	모터2의 순간위치명령값 (position2_set)을 4로 나눈값	모터2의 위치제어 오차	모터2의 위치제어 출력값	모터2의 속도명령값 (03/0A번 제 어모드일때로서 feedforward 성 분 포함)	
+33	모터2의 실제 속도값	모터2의 속도제어오차 (03/0A번 제어모드일때)	모터2의 전류명령값	모터2의 실제전류값	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
0/	모터1의 순간위치명령값 (position1_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터1의 현재 위치값(position1)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터2의 순간위치명령값 (position2_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄스의 엔코더로 간주함), 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터2의 현재 위치값(position2) 을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	
1/	모터1의 순간속도명령값 (speed1_cmd_old / SPEED_SCALE_FACTOR, 03번 모드 에서는 (speed1_long_cmd + speed1_long_cmd_feedforward) / 10, 0A번 모드에서는 speed1_long_cmd_old / 10) 단위크기는 RPM	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터2의 순간속도명령값 (speed2_cmd_old / SPEED_SCALE_FACTOR), 03번 모드 에서는 (speed2_long_cmd + speed2_long_cmd_feedforward) / 10, 0A번 모드에서는 speed2_long_cmd_old / 10) 단위크기는 RPM	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	
2/	모터1의 전류명령값 (current1_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 권선전류의 A/D 변환값 (current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 전류명령값 (current2_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 권선전류의 A/D 변환값 (current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	
3/	모터1의 권선전류의 A/D 변환값 (current1_AD) 2/5/10m요의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 권선전류의 A/D 변환값을 filtering한 값(current1_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 권선전류의 A/D 변환값 (current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 권선전류의 A/D 변환값을 filtering한 값(current2_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	



QxO; 에 대한 return 값은 위치센서로 analog 또는 digital potentiometer을 사용하는 경우에 /4를 하지 않고 원래값을 보내줌. 또 S1/S2/Ts/Tt/Tu/Tv 명령에 의한 구동일 경우에도 /4를 하지 않고 원래값을 보내줌.

100

	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
4/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	
5/	LOW_RESOLUTION_ENCODER==1일때 모터1의 엔코더 A상 신호 * 4095	모터1의 엔코더 B상 신호 * 2000	phase_angle1 엔코더 A/B상 한사이클당 16384	speed1_RPM 단위크기는 RPM	
6/	LOW_RESOLUTION_ENCODER==1일때 모터2의 엔코더 A상 신호 * 4095	모터2의 엔코더 B상 신호 * 2000	phase_angle2 엔코더 A/B상 한사이클당 16384	speed2_RPM 단위크기는 RPM	
7/	LOW_RESOLUTION_ENCODER==1일때 모터1의 정속구간에서의 1ms당 엔코더 펄스수*10000 (speed1_set)	모터1의 1ms당 phase_angle1 변화량 (speed1_RPM_1ms_for_save_alt)	모터2의 정속구간에서의 1ms당 엔코더 펄스수*10000 (speed2_set)	모터2의 1ms당 phase_angle2 변화량 (speed2_RPM_1ms_for_save_alt)	
8/	LOW_RESOLUTION_ENCODER==1일때 Hall_IC1_period[0]	Hall_IC1_period[1]	Hall_IC1_period[2]	Hall_IC1_period[3]	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
10/	모터1의 입력 power(Pin1)	모터1의 출력 power(Pout1)	모터1의 현재 속도값	모터1의 효율	
	단위크기는 0.1W	단위크기는 0.1W	(speed1_RPM)	단위크기는 1%	
11/	모터2의 입력 power(Pin2)	모터2의 출력 power(Pout2)	모터2의 현재 속도값	모터2의 효율	
	단위크기는 0.1W	단위크기는 0.1W	(speed2_RPM)	단위크기는 1%	
12/	모터1의 입력 power(Pin1)	모터1의 출력 토크(Tout1)	모터1의 현재 속도값	모터1의 효율	
	단위크기는 0.1W	단위크기는 mN	(speed1_RPM)	단위크기는 1%	
13/	모터2의 입력 power(Pin2)	모터2의 출력 토크(Tout2)	모터2의 현재 속도값	모터2의 효율	
	단위크기는 0.1W	단위크기는 mN	(speed2_RPM)	단위크기는 1%	
14/	공급전압 65 가 1V 에 해당함.	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당	
15/	공급전압 65 가 1V 에 해당함.	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	모터2의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당	
16/	모터1의 손실 power(Pin1-Pout1)	모터1의 권선 온도(Twinding1)	모터1의 하우징 온도(Thousing1)	모터1의 분위기 온도(Tambient1)	
	단위크기는 0.1W	단위크기는 0.1도	단위크기는 0.1도	단위크기는 0.1도	
17/	모터2의 손실 power(Pin2-Pout2)	모터2의 권선 온도(Twinding2)	모터2의 하우징 온도(Thousing2)	모터2의 분위기 온도(Tambient2)	
	단위크기는 0.1W	단위크기는 0.1도	단위크기는 0.1도	단위크기는 0.1도	
20/	motor1_control_service_time + motor2_control_service_time * 100 단위크기는 1ms의 1%	mainLoopStep_cnt 0-9 범위의 값으로서 단위크기는 1ms	mainLoop_execution_time 단위크기는 1ms의 1%	maximum_interrupt_service_time 단위크기는 PWM 주기의 1%	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
21/	motor1_error 위치제어 모드에서의 단위크기는 1 encoder pulse 속도제어 모드에서의 단위크기는 0.1 rpm 전류제어 모드에서는 해당없음.	motor1_integral 위치제어 모드에서의 단위크기는 1 encoder pulse-msec 속도제어 모드에서의 단위크기는 0.1 rpm-sec 전류제어 모드에서는 해당없음.	motor1_derivative 위치제어 모드에서의 단위크기는 1 encoder pulse/ms 속도제어 모드에서의 단위크기는 0.1 rpm/ms 전류제어 모드에서는 해당없음.	current1_cmd (01번 위치제어모 드에서는 motor1_PWM_set, 03번 위치제어모드에서는 speed1_long_cmd) 위치/속도제어 모드에서의 단위 크기는 AD 변환된 전류값의 단위 크기와 같음 전류제어 모드에서는 해당없음.	
22/	motor2_error	motor2_integral	motor2_derivative	current2_cmd (01번 위치제어모 드에서는 motor2_PWM_set, 03번 위치제어모드에서는 speed2_long_cmd)	
23/	DC_BUS_VOLT_filtered - DC_BUS_VOLT_filtered_at_power_ up 65 가 1V 에 해당함.	speed1_RPM 단위크기는 RPM	DC_BUS_VOLT_filtered 65 가 1V 에 해당함	BRAKE_RELAY_port * 1000 1000 이면 on, 0 이면 off	
24/	DC_BUS_VOLT_filtered	speed1_RPM	power_resistor_ON_timer	BRAKE_RELAY_port * 1000	
25/	DC_BUS_VOLT_filtered	DC_BUS_VOLT_filtered_at_power_ up	power_resistor_ON_timer	BRAKE_RELAY_port * 1000	
30/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt	
31/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation2/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current2_control_limit - current2_control_limit_cnt	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
32/	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation2/1 00 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt	current2_control_limit - current2_control_limit_cnt	
33/	(fault1_status >> 8) & 0x0001	모터1의 권선 온도(Twinding1) 단위크기는 0.1도	모터1의 권선 온도 리밋 (TempLimit1) 단위크기는 0.1도	current1_control_limit - current1_control_limit_cnt	
34/	(fault1_status >> 9) & 0x0001	모터2의 권선 온도(Twinding2) 단위크기는 0.1도	모터2의 권선 온도 리밋 (TempLimit2) 단위크기는 0.1도	current2_control_limit - current2_control_limit_cnt	
35/	LS1A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS1B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS1C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS1D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT1_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)	
36/	LS2A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS2B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS2C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS2D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT2_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
터1), 38(모	<pre>positionx_plus_limit_over_flg * 1 + positionx_minus_limit_over_flg * 10 + motorx_emergency_stop_flg * 100 + motorx_emergency_stop_K_flg * 1000 + motorx_immediate_stop_action_f lg * 10000</pre>	positionx_deceleration_at_limit_flg * 1 + speedx_deceleration_at_limit_f lg * 10	speedx_cmd_old / SPEED_SCALE_FACTOR	모터x의 순간위치명령값 (positionx_set)을 4로 나눈값 위치센서로 analog 또는 digital potentiometer을 사용하는 경우에 /4를 하지 않고 원래값을 보내줌.	
43/	fault_status	fault1_status	IRMCK201_fault_status	MOT_EN_N_port * 1000 + operation_mode	
44/	faultBufPointer	faultBuf[faultBufPointer]	faultBuf[faultBufPointer-1]	faultBuf[faultBufPointer-2]	
45/	fault_save_timer	faultBufPointer_save	faultBuf_save[faultBufPointer_save]	faultBuf_save[faultBufPointer_ save-1]	
46/	fault_status	fault_save_timer	speed1_set	operation_mode	
50/	DC_BUS_VOLT_filtered_at_power_ up	vfiltA[10]	DC_BUS_VOLT_slow_filtered	fault_status	
51/	total_current1_AD_filtered	motor1_current_max	current1_max_limit	fault_status	
52/	total_current2_AD_filtered	motor2_current_max	current2_max_limit	fault_status	
53/	motor1_current_max	current1_over_timer	current1_over_flg + current1_over_K_flg * 10	fault_status	
54/	motor2_current_max	current2_over_timer	<pre>current2_over_flg + current2_over_K_flg * 10</pre>	fault_status	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
60/	모터1의 외부 analog 입력 전압 을 A/D 변환한 값	모터1의 외부 analog 입력 전압 을 A/D 변환한 값의 offset을 제 거하고 필터링한 값	current1_cmd 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	speed1_cmd / SPEED_SCALE_FACTOR 단위크기는 RPM	
61/	모터2의 외부 analog 입력 전압 을 A/D 변환한 값	모터2의 외부 analog 입력 전압 을 A/D 변환한 값의 offset을 제 거하고 필터링한 값	current2_cmd 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	speed2_cmd / SPEED_SCALE_FACTOR 단위크기는 RPM	
62/	CURRENT1_SEN_P_at_power_up	TOTAL_CURRENT1_at_power_up	CURRENT2_SEN_P_at_power_up	TOTAL_CURRENT2_at_power_up	
64/	CURRENT1_SEN_P	total_current1_AD	CURRENT2_SEN_P	total_current2_AD	
66/	automatic_boot_up_timer	operation_mode	speed1_RPM	position1	
67/	automatic_boot_up_timer	operation_mode	speed2_RPM	position2	
68/	total_current1_AD_filtered	motor1_current_max	<pre>current1_max_limit (=max(current1_limit, current1_control_limit) * 1.25)</pre>	min(current1_limit, current1_control_limit) * 1.25	
69/	total_current2_AD_filtered	motor2_current_max	<pre>current2_max_limit (=max(current2_limit, current2_control_limit) * 1.25)</pre>	min(current2_limit, current2_control_limit) * 1.25	
72/	모터1의 전류명령값 (current1_cmd) 단위크기는 0.01A	모터1의 권선전류의 A/D 변환값 (current1_AD) 단위크기는 0.01A	모터2의 전류명령값 (current2_cmd) 단위크기는 0.01A	모터2의 권선전류의 A/D 변환값 (current2_AD) 단위크기는 0.01A	
73/	모터1의 권선전류의 A/D 변환값 (current1_AD) 단위크기는 0.01A	모터1의 권선전류의 A/D 변환값 을 filtering한 값 (current1_AD_filtered) 단위크기는 0.01A	모터2의 권선전류의 A/D 변환값 (current2_AD) 단위크기는 0.01A	모터2의 권선전류의 A/D 변환값 을 filtering한 값 (current2_AD_filtered) 단위크기는 0.01A	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
74/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 단위크기는 0.01A	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 단위크기는 0.01A	
88/	모터1의 03/0A번 제어모드에서 속도의 PID 공통 조정게인 * I 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 P 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 I 조정게인 * 1000	모터1의 03/0A번 제어모드에서 속도의 D 조정게인 * 1000	
89/	모터1의 03/0A번 제어모드에서 속도의 PID 공통 조정게인 * I 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 P 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 I 조정게인 * 1000	모터2의 03/0A번 제어모드에서 속도의 D 조정게인 * 1000	
91/	current1_cmd	current 1_AD	current1_cmd - current1_AD	0	
92/	speed1_long_cmd_old 또는 (speed1_cmd_old * 10) / SPEED_SCALE_FACTOR	speed1_RPM_10	첫번째 변수 - 두번째 변수	motor1_error	
93/	position1_set - POSITION1_MID	(position1 - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor1_error	
95/	current2_cmd	current2_AD	current2_cmd - current2_AD	0	
96/	speed2_long_cmd_old 또는 (speed2_cmd_old * 10) / SPEED_SCALE_FACTOR	speed2_RPM_10	첫번째 변수 - 두번째 변수	motor2_error	
97/	position2_set - POSITION2_MID	(position2 - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor2_error	



	RJM_VER2/3/4/8인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
98/	모터1의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추 가조정게인 * 1000	모터1의 P 조정게인 * 1000	모터1의 I 조정게인 * 1000	모터1의 D 조정게인 * 1000	
99/	모터2의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추 가조정게인 * 1000	모터2의 P 조정게인 * 1000	모터2의 I 조정게인 * 1000	모터2의 D 조정게인 * 1000	
111/	BRAKE 제어기의 경우는 모터1의 속도값(speed1_RPM)	모터1의 전류값(current1_AD)	모터1의 speed1_cmd_old / SPEED_SCALE_FACTOR	모터1의 current1_control_limit	
112/	BRAKE 제어기의 경우는 모터1의 속도값(speed1_RPM)	모터1의 전류값(current1_AD)	모터1의 포텐시오메터 변위 A/D 값	모터1의 압력센서 A/D값	
114/	speed1 command	speed1_RPM (Hall IC only 제어 의 경우는 필터링한 값)	current1_cmd	(MOTOR1_INERTIAL_RATIO * (long long)motor1_acc_rate) >> 12	
115/	speed2 command	speed2_RPM (Hall IC only 제어 의 경우는 필터링한 값)	current2_cmd	(MOTOR2_INERTIAL_RATIO * (long long)motor2_acc_rate) >> 12	
116/	notch filter MOTOR1_uk1_0	notch filter MOTOR1_yk1_0	notch filter MOTOR2_uk1_0	notch filter MOTOR2_yk1_0	
508/	current1_cmd	offset을 제거한 전류 A/D 변환 값	current1_cmd - offset을 제거한 전류 A/D 변환값	0	
509/	current2_cmd	offset을 제거한 전류 A/D 변환 값	current2_cmd - offset을 제거한 전류 A/D 변환값	0	



세어기 명령어 LISI

	RJM_VER2/3/4/8인 경우			
dddd1 교	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
0x800 +32	모터1의 순간위치명령값 (position1_set)을 4로 나는 값 단위크기는 4 encoder pulse (엔코더를 안쓰는 경우는 16384 펄스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일때는 단위크기는 1 encoder pulse임	모터1의 위치제어 오차 단위크기는 1 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함)	모터1의 위치제어 출력값 (01번 제어모드에서는 PWM값, 02번제어 모드에서는 전류명령값, 03번 제 어모드에서는 속도명령값, 스테 핑모터에서는 lead angle 명령 값)	모터1의 속도명령값 (03/0A번 제 어모드일때로서 feedforward 성 분 포함) 단위크기는 RPM*10
	모터1의 실제 속도값 단위크기는 RPM*10	모터1의 속도제어오차 (03/0A번 제어모드일때) 단위크기는 RPM*10	모터1의 전류명령값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 실제전류값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당
0x800	모터2의 순간위치명령값 (position2_set)을 4로 나눈 값	모터2의 위치제어 오차	모터2의 위치제어 출력값	모터2의 속도명령값 (03/0A번 제 어모드일때로서 feedforward 성 분 포함)
+33	모터2의 실제 속도값	모터2의 속도제어오차 (03/0A번 제어모드일때)	모터2의 전류명령값	모터2의 실제전류값



	RJM_VER5/6인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
0/	모터1의 순간위치명령값 (position1_set)을 4로 나눈값 단위크기는 4/16384 바퀴임, 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1/16384 바퀴임	모터1의 현재 위치값(position1) 을 4로 나눈값 단위크기는 4/16384 바퀴임, 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1/16384 바퀴임	모터2의 순간위치명령값 (position2_set)을 4로 나눈값 단위크기는 4/16384 바퀴임, 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1/16384 바퀴임	모터2의 현재 위치값(position2) 을 4로 나눈값 단위크기는 4/16384 바퀴임, 단 S2/Tt/Tv 명령에 의한 작동중일 때는 단위크기는 1/16384 바퀴임	
1/	모터1의 순간속도명령값 (speed1_cmd_old / (70L * MOTOR1_NUMBER_of_POLE_PAIR)) 단위크기는 RPM	모터1의 현재 속도값 ((speed1_RPM_count * 60000L) / MOTOR1_PULSE_PER_REV_ENC) 단위크기는 RPM	모터2의 순간속도명령값 (speed2_cmd_old / (70L * MOTOR2_NUMBER_of_POLE_PAIR)) 단위크기는 RPM	모터2의 현재 속도값 ((speed2_RPM_count * 60000L) / MOTOR2_PULSE_PER_REV_ENC) 단위크기는 RPM	
2/	모터1의 A상 전류명령값 (current1_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 A상 권선전류의 A/D 변환값(current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 B상 전류명령값 (current2_cmd), 단 3/5상인 경 우는 current2_AD 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 B상 권선전류의 A/D 변 환값(current2_AD), 단 3/5상인 경우는 current3_AD 2/5/10m요의 경우 81.9/204.75/409.5 가 1A에 해당	
3/	모터1의 A상 권선전류의 A/D 변환값(current1_AD), 단 3상일때는 current2_cmd, 단 5상일때는 current1_cmd 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 A상 권선전류의 A/D 변환값을 filtering한 값 (current1_AD_filtered), 단 3/5 상일때는 current4_AD 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 B상 권선전류의 A/D 변 환값(current2_AD), 단 3/5상일 때는 current5_AD 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 B상 권선전류의 A/D 변 환값을 filtering한 값 (current2_AD_filtered), 단 3상 일때는 current6_AD, 단 5상일때 는 Iq1_AD 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	
4/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	



		RJM_VER5/62	! 경우	
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
7/	모터2의 A상(RJM_VER6는 모터1의 C상) 전류명령 (current3_cmd), 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 A상(RJM_VER6는 모터1의 C상) 권선전류의 A/D 변환값 (current3_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 B상(RJM_VER6는 모터2의 A상) 전류명령값(current4_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 B상(RJM_VER6는 모터2의 A상) 권선전류의 A/D 변환값 (current4_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당
8/	RJM_VER6의 경우는 모터2의 B상 전류명령값 (current5_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 B상 권선전류의 A/D 변 환값(current5_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 C상 전류명령값 (current6_cmd) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 C상 권선전류의 A/D 변 환값(current6_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당
8/	RJM_VER5의 경우는 모터1의 순간위치명령값 (position1_set)을 4로 나눈값 16384 일때 한바퀴임	모터1의 엔코더에 의한 현재 위 치값(position1_ENC)을 한바퀴당 16384로 환산한 후에 4로 나눈값 16384 일때 한바퀴임	모터2의 순간위치명령값 (position2_set)을 4로 나눈값 16384 일때 한바퀴임	모터2의 엔코더에 의한 현재 위 치값(position2_ENC)을 한바퀴당 16384로 환산한 후에 4로 나눈값 16384 일때 한바퀴임
14/	공급전압 65 가 1V 에 해당함.	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 현재 속도값 (speed1_RPM) 단위크기는 RPM	모터1의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당
15/	공급전압 65 가 1V 에 해당함.	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 현재 속도값 (speed2_RPM) 단위크기는 RPM	모터2의 부호있는 전류진폭 2/5/10mΩ의 경우 81.9/204.75/409.5 가 전류진폭 1A에 해당
20/	motor1_control_service_time + motor2_control_service_time * 100 단위크기는 1ms의 1%	mainLoopStep_cnt 0-9 범위의 값으로서 단위크기는 1ms	mainLoop_execution_time 단위크기는 1ms의 1%	maximum_interrupt_service_time 단위크기는 PWM 주기의 1%



	RJM_VER5/6인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
21/	motor1_error 02번 위치제어 모드에서의 단위 크기는 1 encoder pulse	motor1_integral 02번 위치제어 모드에서의 단위 크기는 1 encoder pulse-msec	motor1_derivative 02번 위치제어 모드에서의 단위 크기는 1 encoder pulse/ms	lead_angle1_set 4096일때 90도		
22/	motor2_error	motor2_integral	motor2_derivative	lead_angle2_set		
23/	DC_BUS_VOLT_filtered(모터공급 전압) - DC_BUS_VOLT_filtered_at_power_ up 65 가 1V 에 해당함.	speed1_RPM 단위크기는 RPM	DC_BUS_VOLT_filtered(모터공급 전압) 65 가 1V 에 해당함.	BRAKE_RELAY_port * 1000 + DC_DC_converter_voltage_transi ent_timer 1000 근방 이면 on, 0 근방이면 off		
24/	DC_BUS_VOLT_filtered (모터공급 전압)	speed1_RPM	(1 - CH_PUMP_EN_N_port) * 500 + power_resistor_ON_timer	BRAKE_RELAY_port * 1000 + DC_DC_converter_voltage_transi ent_timer		
25/	DC_BUS_VOLT_filtered (모터공급 전압)	DC_BUS_VOLT_filtered_at_power_ up	(1 - CH_PUMP_EN_N_port) * 500 + power_resistor_ON_timer	BRAKE_RELAY_port * 1000 + DC_DC_converter_voltage_transi ent_timer		
30/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt		
31/	(fault1_status >> 11) & 0x0001	controller_heat_dissipation2/1 00 단위크기는 0.1W	controller_heat_dissipation_li mit/100 단위크기는 0.1W	current2_control_limit - current2_control_limit_cnt		
32/	controller_heat_dissipation1/1 00 단위크기는 0.1W	controller_heat_dissipation2/1 00 단위크기는 0.1W	current1_control_limit - current1_control_limit_cnt	current2_control_limit - current2_control_limit_cnt		



	RJM_VER5/6인 경우					
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수		
33/	(fault1_status >> 8) & 0x0001	모터1의 권선 온도(Twinding1) 단위크기는 0.1도	모터1의 권선 온도 리밋 (TempLimit1) 단위크기는 0.1도	current1_control_limit - current1_control_limit_cnt		
34/	(fault1_status >> 9) & 0x0001	모터2의 권선 온도(Twinding2) 단위크기는 0.1도	모터2의 권선 온도 리밋 (TempLimit2) 단위크기는 0.1도	current2_control_limit - current2_control_limit_cnt		
35/	LS1A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS1B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS1C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS1D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT1_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)		
36/	LS2A_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 1000)	LS2B_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 2000)	LS2C_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 3000)	LS2D_port에 작용하는 전압의 A/D 변환값 또는 ON/OFF 입력값 + EXT_FAULT2_port * 5000 (analog의 경우는 4095 일때 3V, ON/OFF의 경우는 0 또는 4000)		
터1), 38(모	<pre>positionx_plus_limit_over_flg * 1 + positionx_minus_limit_over_flg * 10 + motorx_emergency_stop_flg * 100 + motorx_emergency_stop_K_flg * 1000 + motorx_immediate_stop_action_f lg * 10000</pre>	positionx_deceleration_at_limit_flg * 1 + speedx_deceleration_at_limit_f lg * 10	speedx_cmd_old / SPEED_SCALE_FACTOR	모터x의 순간위치명령값 (positionx_set)을 4로 나눈값		



	RJM_VER5/6인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
43/	fault_status	fault1_status	IRMCK201_fault_status	MOT_EN_N_port * 1000 + operation_mode	
44/	faultBufPointer	faultBuf[faultBufPointer]	faultBuf[faultBufPointer-1]	faultBuf[faultBufPointer-2]	
45/	fault_save_timer	faultBufPointer_save	faultBuf_save[faultBufPointer_save]	faultBuf_save[faultBufPointer_ save-1]	
46/	fault_status	fault_save_timer	speed1_set	operation_mode	
50/	DC_BUS_VOLT_filtered_at_power_ up	vfiltA[10]	DC_BUS_VOLT_slow_filtered	fault_status	
51/	total_current1_AD_filtered	motor1_current_max	current1_max_limit	fault_status	
52/	total_current2_AD_filtered	motor2_current_max	current2_max_limit	fault_status	
53/	motor1_current_max	current1_over_timer	<pre>current1_over_flg + current1_over_K_flg * 10</pre>	fault_status	
54/	motor2_current_max	current2_over_timer	current2_over_flg + current2_over_K_flg * 10	fault_status	
60/	모터1의 외부 analog 입력 전압 을 A/D 변환한 값	모터1의 외부 analog 입력 전압 을 A/D 변환한 값의 offset을 제 거하고 필터링한 값	current1_cmd 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	speed1_cmd / SPEED_SCALE_FACTOR 단위크기는 RPM	
61/	모터2의 외부 analog 입력 전압 을 A/D 변환한 값	모터2의 외부 analog 입력 전압 을 A/D 변환한 값의 offset을 제 거하고 필터링한 값	current2_cmd 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	speed2_cmd / SPEED_SCALE_FACTOR 단위크기는 RPM	
62/	CURRENT1_SEN_P_at_power_up	CURRENT1_SEN_N_at_power_up	CURRENT2_SEN_P_at_power_up	CURRENT2_SEN_N_at_power_up	
64/	CURRENT1_SEN_P	CURRENT1_SEN_N	CURRENT2_SEN_P	CURRENT2_SEN_N	
66/	automatic_boot_up_timer	operation_mode	speed1_RPM	position1	
67/	automatic_boot_up_timer	operation_mode	speed2_RPM	position2	



	RJM_VER5/6인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
68/	total_current1_AD_filtered	motor1_current_max	<pre>current1_max_limit (=max(current1_limit, current1_control_limit) * 1.25)</pre>	min(current1_limit, current1_control_limit) * 1.25	
69/	total_current2_AD_filtered	motor2_current_max	<pre>current2_max_limit (=max(current2_limit, current2_control_limit) * 1.25)</pre>	min(current2_limit, current2_control_limit) * 1.25	
72/	모터1의 A상 전류명령값 (current1_cmd) 단위크기는 0.01A	모터1의 A상 권선전류의 A/D 변 환값(current1_AD) 단위크기는 0.01A	모터1의 B상 전류명령값 (current2_cmd), 단 3/5상일때는 current2_AD 단위크기는 0.01A	모터1의 B상 권선전류의 A/D 변 환값(current2_AD), 단 3/5상일 때는 current3_AD 단위크기는 0.01A	
73/	모터1의 A상 권선전류의 A/D 변 환값(current1_AD), 단 3상일때 는 current2_cmd, 단 5상일때는 current1_cmd 단위크기는 0.01A	모터1의 A상 권선전류의 A/D 변 환값을 filtering한 값 (current1_AD_filtered), 단 3/5 상일때는 current4_AD 단위크기는 0.01A	모터1의 B상 권선전류의 A/D 변 환값(current2_AD), 단 3/5상일 때는 current5_AD 단위크기는 0.01A	모터1의 B상 권선전류의 A/D 변환값을 filtering한 값 (current2_AD_filtered), 단 3상일때는 current6_AD, 단 5상일때는 lq1_AD 단위크기는 0.01A	
74/	모터1의 현재 총전류의 A/D 변환 값(total_current1_AD) 단위크기는 0.01A	모터1의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current1_AD_filtered) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값(total_current2_AD) 단위크기는 0.01A	모터2의 현재 총전류의 A/D 변환 값을 필터링한 값 (total_current2_AD_filtered) 단위크기는 0.01A	



	RJM_VER5/6인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
91/	lead_angle1_set	lead_angle1	lead_angle1_set - lead_angle1	0	
92/	speed1_cmd_old / (7L * MOTOR1_NUMBER_of_POLE_PAIR)	(speed1_RPM_count * 600000L) / MOTOR1_PULSE_PER_REV_ENC	첫번째 변수 - 두번째 변수	motor1_error	
93/	(position1_set - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	(position1 - POSITION1_MID) / MOTOR1_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor1_error	
95/	lead_angle2_set	lead_angle2	lead_angle2_set - lead_angle2	0	
96/	speed2_cmd_old / (7L * MOTOR2_NUMBER_of_POLE_PAIR)	(speed2_RPM_count * 600000L) / MOTOR2_PULSE_PER_REV_ENC	첫번째 변수 - 두번째 변수	motor2_error	
97/	(position2_set - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	(position2 - POSITION2_MID) / MOTOR2_POSITION_SCALE_FACTOR	첫번째 변수 - 두번째 변수	motor2_error	
98/	모터1의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추가조정 게인 * 1000	모터1의 P 조정게인 * 1000	모터1의 I 조정게인 * 1000	모터1의 D 조정게인 * 1000	
99/	모터2의 PID 공통 조정게인 * I 조정게인 * I 위치제어 추가조정 게인 * 1000	모터2의 P 조정게인 * 1000	모터2의 I 조정게인 * 1000	모터2의 D 조정게인 * 1000	
114/	speed1_cmd_old / (70L * MOTOR1_NUMBER_of_POLE_PAIR)	speed1_RPM	current1_cmd	(MOTOR1_INERTIAL_RATIO * (long long)motor1_acc_rate) >> 12	
115/	<pre>speed2_cmd_old / (70L * MOTOR2_NUMBER_of_POLE_PAIR)</pre>	speed2_RPM	current2_cmd	<pre>(MOTOR2_INERTIAL_RATIO * (long long)motor2_acc_rate) >> 12</pre>	
116/	notch filter MOTOR1_uk1_0	notch filter MOTOR1_yk1_0	notch filter MOTOR2_uk1_0	notch filter MOTOR2_yk1_0	
508/	lead_angle1_set	lead_angle1	lead_angle1_set - lead_angle1	0	
509/	lead_angle2_set	lead_angle2	lead_angle2_set - lead_angle2	0	

116

	RJM_VER5/6인 경우			
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
0x800 +32	모터1의 순간위치명령값 (position1_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터1의 위치제어 오차 단위크기는 1 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함)	모터1의 위치제어 출력값 (lead angle 명령값) 4096일때 90도	모터1의 lead angle 값 4096일때 90도
	모터1의 실제 속도값 단위크기는 RPM*10	모터1의 lead angle 제어오차 4096일때 90도	모터1의 전류명령값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터1의 실제전류값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당
0x800 +33	모터2의 순간위치명령값 (position2_set)을 4로 나눈값 단위크기는 4 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함), 단 S1/Ts/Tu 명령에 의한 작동중일 때는 단위크기는 1 encoder pulse임	모터2의 위치제어 오차 단위크기는 1 encoder pulse (엔 코더를 안쓰는 경우는 16384 펄 스의 엔코더로 간주함)	모터2의 위치제어 출력값 (lead angle 명령값) 4096일때 90도	모터2의 lead angle 값 4096일때 90도
	모터2의 실제 속도값 단위크기는 RPM*10	모터2의 lead angle 제어오차 4096일때 90도	모터2의 전류명령값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당	모터2의 실제전류값 2/5/10mΩ의 경우 81.9/204.75/409.5 가 1A에 해당



		RJM_VER8 외장형 F/	Γ 센서인 경우	
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
2/	strain guage bridge1의 증폭된 전압값 (ADCRESULT8_reg >> 4)	strain guage bridge2의 증폭된 전압값 (ADCRESULT9_reg >> 4)	strain guage bridge3의 증폭된 전압값 (ADCRESULT10_reg >> 4)	strain guage bridge4의 증폭된 전압값 (ADCRESULT11_reg >> 4)
3/	strain guage bridge1의 증폭된 전압값을 필터링한 값 (vfilt[8])	strain guage bridge2의 증폭된 전압값을 필터링한 값 (vfilt[9])	strain guage bridge3의 증폭된 전압값을 필터링한 값 (vfilt[10])	strain guage bridge4의 증폭된 전압값을 필터링한 값 (vfilt[11])
4/	strain guage bridge1의 증폭된 전압값을 필터링하고 offset(온 도 offset 제외)을 제거한 값 (vfilt[8] - force_data_offset[0])	strain guage bridge2의 증폭된 전압값을 필터링하고 offset(온 도 offset 제외)을 제거한 값 (vfilt[9] - force_data_offset[1])	strain guage bridge3의 증폭된 전압값을 필터링하고 offset(온 도 offset 제외)을 제거한 값 (vfilt[10] - force_data_offset[2])	strain guage bridge4의 증폭된 전압값을 필터링하고 offset(온 도 offset 제외)을 제거한 값 (vfilt[11] - force_data_offset[3])
5/	온도보상 안된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상 안된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상 안된 compression force 단위크기는 0.1N	온도센서 온도값 단위크기는 0.01도
6/	strain guage bridge1의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[8] - force_data_offset[0] - temp_offset[0])	strain guage bridge2의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[9] - force_data_offset[1] - temp_offset[1])	strain guage bridge3의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[10] - force_data_offset[2] - temp_offset[2])	strain guage bridge4의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[11] - force_data_offset[3] - temp_offset[3])
7/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상된 compression force 단위크기는 0.1N	온도센서 온도값 단위크기는 0.01도
8/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도센서 저항값 단위크기는 Ohm	온도보상된 compression force 단위크기는 0.1N	온도센서 온도값 단위크기는 0.01도

6/7/8 번항은 온도센서가 장착된 H/W 버젼에서만 해당됨.



	RJM_VER8 외장형 F/T 센서인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
10/	온도보상 안된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상 안된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상 안된 compression force 단위크기는 0.1N	온도보상 안된 독립 1채널의 load_cell 값 단위크기는 0.1N	
11/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상된 compression force 단위크기는 0.1N	온도보상된 독립 1채널의 load_cell 값 단위크기는 0.1N	
14/	공급전압 65 가 1V 에 해당함.				
15/	공급전압 65 가 1V 에 해당함.				
20/	operation_mode	mainLoopStep_cnt 0-9 범위의 값으로서 단위크기는 1ms	mainLoop_execution_time 단위크기는 1ms의 1%	maximum_interrupt_service_time 단위크기는 10us의 1%	
43/	fault_status	fault1_status	IRMCK201_fault_status	MOT_EN_N_port * 1000 + operation_mode	
44/	faultBufPointer	faultBuf[faultBufPointer]	faultBuf[faultBufPointer-1]	faultBuf[faultBufPointer-2]	
45/	fault_save_timer	faultBufPointer_save	faultBuf_save[faultBufPointer_save]	faultBuf_save[faultBufPointer_save-1]	
46/	fault_status	fault_save_timer	speed1_set	operation_mode	



		RJM_VER7 내장형 F/	Γ 센서인 경우	
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
2/	strain guage bridge1의 증폭된	strain guage bridge2의 증폭된	strain guage bridge3의 증폭된	strain guage bridge4의 증폭된
	전압값 (ADCRESULTO_reg >> 4)	전압값 (ADCRESULT1_reg >> 4)	전압값 (ADCRESULT2_reg >> 4)	전압값 (ADCRESULT3_reg >> 4)
3/	strain guage bridge1의 증폭된	strain guage bridge2의 증폭된	strain guage bridge3의 증폭된	strain guage bridge4의 증폭된
	전압값을 필터링한 값	전압값을 필터링한 값	전압값을 필터링한 값	전압값을 필터링한 값
	(vfilt[8])	(vfilt[9])	(vfilt[10])	(vfilt[11])
4/	strain guage bridge1의 증폭된	strain guage bridge2의 증폭된	strain guage bridge3의 증폭된	strain guage bridge4의 증폭된
	전압값을 필터링하고 offset을	전압값을 필터링하고 offset을	전압값을 필터링하고 offset을	전압값을 필터링하고 offset을
	제거한 값 (vfilt[8] -	제거한 값 (vfilt[9] -	제거한 값 (vfilt[10] -	제거한 값 (vfilt[11] -
	force_data_offset[0])	force_data_offset[1])	force_data_offset[2])	force_data_offset[3])
5/	온도보상 안된 X 방향 모멘트값 (X_moment)	온도보상 안된 Y 방향 모멘트값 (Y_moment)	온도보상 안된 compression force	온도센서 온도값
	단위크기는 0.01Nm	단위크기는 0.01Nm	단위크기는 0.1N	단위크기는 0.01도
6/	strain guage bridge1의 증폭된	strain guage bridge2의 증폭된	strain guage bridge3의 증폭된	strain guage bridge4의 증폭된
	전압값을 필터링하고 offset(온	전압값을 필터링하고 offset(온	전압값을 필터링하고 offset(온	전압값을 필터링하고 offset(온
	도 offset 포함)을 제거한 값	도 offset 포함)을 제거한 값	도 offset 포함)을 제거한 값	도 offset 포함)을 제거한 값
	(vfilt[8] -	(vfilt[9] -	(vfilt[10] -	(vfilt[11] -
	force_data_offset[0] -	force_data_offset[1] -	force_data_offset[2] -	force_data_offset[3] -
	temp_offset[0])	temp_offset[1])	temp_offset[2])	temp_offset[3])
7/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상된 compression force 단위크기는 0.1N	온도센서 온도값 단위크기는 0.01도
8/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도센서 저항값 단위크기는 Ohm	온도보상된 compression force 단위크기는 0.1N	온도센서 온도값 단위크기는 0.01도



	RJM_VER7 내장형 F/T 센서인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
10/	온도보상 안된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상 안된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상 안된 compression force 단위크기는 0.1N	온도보상 안된 load_cell	
11/	온도보상된 X 방향 모멘트값 (X_moment) 단위크기는 0.01Nm	온도보상된 Y 방향 모멘트값 (Y_moment) 단위크기는 0.01Nm	온도보상된 compression force 단위크기는 0.1N	온도보상된 load_cell	
12/	6축인 경우는 strain guage bridge5의 증폭된 전압값 (ADCRESULT4_reg >> 4)	strain guage bridge6의 증폭된 전압값 (ADCRESULT5_reg >> 4)	DA output5 4095 가 3V 에 해당함.	DA output6 4095 가 3V 에 해당함.	
13/	6축인 경우는 strain guage bridge5의 증폭된 전압값을 필터링한 값 (vfilt[12])	strain guage bridge6의 증폭된 전압값을 필터링한 값 (vfilt[13])	strain guage bridge5의 증폭된 전압값을 필터링하고 offset을 제거한 값 (vfilt[12] - force_data_offset[4])	strain guage bridge6의 증폭된 전압값을 필터링하고 offset을 제거한 값 (vfilt[13] - force_data_offset[5])	
14/	공급전압 65 가 1V 에 해당함.				
15/	6축이 아닌 경우는 공급전압 65 가 1V 에 해당함.				
15/	6축인 경우는 온도보상 안된 X 방향 force값 (X_force) 단위크기는 0.1N	온도보상 안된 Y 방향 force값 (Y_force) 단위크기는 0.1N	온도보상 안된 Z 방향 모멘트값 (Z_moment) 단위크기는 0.01Nm	온도센서 온도값 단위크기는 0.01도	



	RJM_VER7 내장형 F/T 센서인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
16/	6축인 경우는 strain guage bridge5의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[12] - force_data_offset[4] - temp_offset[4])	strain guage bridge6의 증폭된 전압값을 필터링하고 offset(온 도 offset 포함)을 제거한 값 (vfilt[13] - force_data_offset[5] - temp_offset[5])	온도센서 온도값 단위크기는 0.01도	온도센서 온도값 증가량 단위크기는 0.01도	
17/	6축인 경우는 온도보상된 X 방향 force값 (X_force) 단위크기는 0.1N	온도보상된 Y 방향 force값 (Y_force) 단위크기는 0.1N	온도보상된 Z 방향 모멘트값 (Z_moment) 단위크기는 0.01Nm	온도센서 온도값 단위크기는 0.01도	
18/	temperature_FT_sensor_thermoco uple[0] - filtered hot junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[1] - filtered hot junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[2] - filtered hot junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[3] - filtered hot junction 온도 단위크기는 0.01℃	
19/	temperature_FT_sensor_thermoco uple[4] - unfiltered cold junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[5] - unfiltered cold junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[6] - unfiltered cold junction 온도 단위크기는 0.01℃	temperature_FT_sensor_thermoco uple[7] - unfiltered cold junction 온도 단위크기는 0.01℃	
20/	operation_mode	mainLoopStep_cnt 0-9 범위의 값으로서 단위크기는 1ms	mainLoop_execution_time 단위크기는 1ms의 1%	maximum_interrupt_service_time 단위크기는 10us의 1%	



	RJM_VER7 내장형 F/T 센서인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
43/	fault_status	fault1_status	IBMUKZUI TAHIT STATUS	MOT_EN_N_port * 1000 + operation_mode	
44/	faultBufPointer	faultBuf[faultBufPointer]	faultBuf[faultBufPointer-1]	faultBuf[faultBufPointer-2]	
45/	fault_save_timer	faultBufPointer_save	faultBuf_save[faultBufPointer_save]	faultBuf_save[faultBufPointer_save-1]	
46/	fault_status	fault_save_timer	speed1_set	operation_mode	



			RJM	M_VER7 내장형 F/	Γ 센서인 경우			
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	다섯번째 변수	여섯번째 변수	일곱번째 변수	여덟번째 변수
6144	ADCRESULTO_reg >> 4	ADCRESULT1_reg >> 4	ADCRESULT2_reg >> 4	ADCRESULT3_reg >> 4	ADCRESULT4_reg >> 4	ADCRESULT5_reg >> 4	ADCRESULT7_reg >> 4	ADCRESULT15_re g >> 4
6146	ADCRESULTO_reg >> 4	ADCRESULT1_reg >> 4	ADCRESULT2_reg >> 4	ADCRESULT3_reg >> 4	ADCRESULT4_reg >> 4	ADCRESULT5_reg >> 4	ADCRESULT7_reg >> 4	ADCRESULT15_re g >> 4
6147	vfilt[8]	vfilt[9]	vfilt[10]	vfilt[11]	vfilt[12]	vfilt[13]	temperature_FT _sensor	NTC_resistance
6148	vfilt[8]- force_data_off set[0]	vfilt[9]- force_data_off set[1]	vfilt[10]- force_data_off set[2]	vfilt[11]- force_data_off set[3]	vfilt[12]- force_data_off set[4]	vfilt[13]- force_data_off set[5]	temperature_FT _sensor	NTC_resistance
6149	온도보상 안된 X_moment	온도보상 안된 Y_moment	온도보상 안된 foot_force	온도보상 안된 X_force	온도보상 안된 Y_force	온도보상 안된 Z_moment	temperature_FT _sensor	NTC_resistance
6150	옾셋, 온도 보 상된 vfilt[8]	옾셋, 온도 보 상된 vfilt[9]	옾셋, 온도 보 상된 vfilt[10]	옾셋, 온도 보 상된 vfilt[11]	옾셋, 온도 보 상된 vfilt[12]	옾셋, 온도 보 상된 vfilt[13]	temperature_FT _sensor	NTC_resistance
6151	온도보상된 X_moment	온도보상된 Y_moment	온도보상된 foot_force	온도보상된 X_force	온도보상된 Y_force	온도보상된 Z_moment	temperature_FT _sensor	NTC_resistance
6152	DA output0	DA output1	DA output2	DA output3	DA output4	DA output5	temperature_FT _sensor	NTC_resistance
6153	온도보상되고 추가 필터링된 X_moment	온도보상되고 추가 필터링된 Y_moment	온도보상되고 추가 필터링된 foot_force	온도보상되고 추가 필터링된 X_force	온도보상되고 추가 필터링된 Y_force	온도보상되고 추가 필터링된 Z_moment	temperature_FT _sensor	NTC_resistance



Ts/Tt 명령에 의한 step파 test와 Tu/Tv 명령에 의한 sin파 test시의 Qxdddd1 사용

	Ts/Tt/Tu/Tv 명령에 의한 test시의 모니터링				
dddd1 값	위치제어모드 (01/02/03/04/0E/0F 모드)	속도제어모드 (05/06/0A 모드)	전류제어모드 (07 모드)	comment	
0	위치값	-	-	TMS320F28334/5에서 03번모드의 속도제어 튜닝은 0A번 모드에서 실시할것	
1	속도값		_		
75/76	speedx_RPM_10/speedx_RPM_period_10/speedx_	RPM_count_10/speedx_RPM_combined_10	-	TMS320F28334/5에서 카운트방식 /펄스주기방식/조합방식 속도값	
2/3	상전류값 DC의 경우는 Qx2;에서 모터1/ 의 상전류가 모두 표시됨				
4	총전류값				
93/97	위치제어 상황				
92/96	속도제어 상황				
91/95	전류제어상황				
55/56	currentx_cmd/lqx_AD/motorx_moving_flg * 1000 + Idrefx + (MOTORx_PA_end_processing_time_set - BLDC의 경우 전류값 MOTORx_PA_end_timer)/ldx_AD				
80/81	motorx_error/speedx_long_cmd + speedx_long_cmd_feedforward/motorx_error_ speed/currentx_cmd			TMS320F28334/5에서 03번 위치 제어모드 일때 위치/속도제어루 프의 오차 및 제어출력값	
82/83	motorx_error_speed/motorx_integral_speed/motorx_derivative_speed/currentx_cmd -			TMS320F28334/5에서 03번 위치 /OA번 속도제어모드 일때 속도 제어루프의 오차 및 제어출력값	
86/87	speedx_long_cmd_feedforward/speedx_long_cmd + speedx_long_cmd_feedforward/motorx_error_speed/currentx_cmd			TMS320F28334/5에서 03번 위치 /OA번 속도제어모드 일때	
114/115	속도명령/속도/전류명령/전류offset		-	가속에 따른 전류 offset	
77	speedx_RPM_10/gain_factorx		_	속도에 따른 게인 조정	



Ts/Tt 명령에 의한 step파 test와 Tu/Tv 명령에 의한 sin파 test시의 Qxdddd1 사용

	Ts/Tt/Tu/Tv 명령에 의한 test시의 모니터링					
dddd1 값	위치제어모드 (01/02/03/04/0E/0F 모드) 속도제어모드 (05/06/0A 모드) 전류제어모드 (07 모드) comment					
51/52	총전류/상전류/전류리밋1/오류코드 과전류보호					
53/54	상전류/과전류타이머/과전류flag/오류코드 과전류보호					
68/69	총전류/상전류/전류리밋high/전류리밋low 과전류보호			과전류보호		



명령어	구분	설명	
		Qz 명령어로 1ms 마다 값을 host로 전송하는 4개의 변수, 또는 Qy 5	명령어로 host로 전송하는 8개 변수중에
		서 앞쪽 4개의 변수에 대한 주소를 설정함.	
	operation	1st word : 첫번째 변수의 주소	
	operation	2nd word : 두번째 변수의 주소	
		3rd word : 세번째 변수의 주소	
		4th word : 네번째 변수의 주소	
	return value	QZHLHL, HLHL, HLHL;	
		QRHLHL,HL; 명령어로 설정가능한 변수의 주소범위를 벗어나면 reset	됨.
		설정가능한 주소범위는 다음과 같음.	
		1st/2nd/3rd/4th word : RAM address (TMS320LF2406A)	
;QZHLHL,HLHL,HLH		[0000 - 005F] : memory mapped registers [0060 - 0	07F] : heap memory
L,HLHL; (+#^%)			3FF] : stack
		_	[531] : peripheral memory
;QZ; (+#^%)		1st/2nd/3rd/4th : RAM address (TMS320F810/11/12)	
		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 0	
	comment		FFF] : LO/L1 RAM block (.ebss)
		[E000 - FFFF] : HO RAM block (.esysmem)	
		[E000 - FFFF]의 주소를 설정하면 실제로는 [3F8000 - 3F9FFF]의 주	소에 있는 값을 지칭하도록 내부에서
		처리됨.	
		1st/2nd/3rd/4th : RAM address (TMS320F28334/28335)	
		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 0	7FF] : M1 RAM block (.bss)
		[0880 - 793F] : peripheral memory	
		[COOO - DFFF] : L4/L5 RAM block (.ebss)	
		[E000 - FFFF] : L6/L7 RAM block (.esysmem)	



명령어	구분	설명
	operation	QZ 명령어로 주소가 지정된 4word를 1msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
	return value	none.
		전송되는 데이터는 Qz; 명령을 전송해온 RS232C/RS485/CAN 채널로만 전송됨.
;Qz; (+#^%)		모터제어기의 RS232C 채널이 115200 baud rate의 전송속도로 설정이 되어 있을 때만 유효함.
, QZ , (# /0)	comment	전송데이터의 주소는 QZ 명령어에 의하여 설정됨.
	Comment	Qz; 명령어를 한번 보내면 전송개시, 한번 더 보내면 전송을 정지함.
		1ms당 4word를 8byte로 나누어서 보냄.
		2012년 12월 27일 S/W 버젼부터는 Qxyz; 명령어로도 전송을 정지할수 있음.
	operation	QZ 명령어로 주소가 지정된 4word를 2msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
	return value	none.
		전송되는 데이터는 Qzx; 명령을 전송해온 RS232C/RS485/CAN 채널로만 전송됨.
;Qzx; (+#^)	comment	모터제어기의 RS232C 채널이 115200 baud rate의 전송속도로 설정이 되어 있을 때만 유효함.
, αΖΛ, (Ιπ)		전송데이터의 주소는 QZ 명령어에 의하여 설정됨.
		Qz; 또는 Qzx; 명령어를 한번 보내면 전송개시, 한번 더 보내면 전송을 정지함.
		2012년 1월 08일 S/W 버젼부터 적용되며 2ms당 4word를 13byte로 나누어서 보냄.
		2012년 12월 27일 S/W 버젼부터는 Qxyz; 명령어로도 전송을 정지할수 있음.
	operation	QZ/QY 명령어로 주소가 지정된 8word를 2msec 마다 host PC로 전송을 시작하게 하거나 전송을 중지 시킴.
	return value	none.
		전송되는 데이터는 Qzy; 명령을 전송해온 RS232C/RS485/CAN 채널로만 전송됨.
;Qzy; (+#^)		모터제어기의 RS232C 채널이 115200 baud rate의 전송속도로 설정이 되어 있을 때만 유효함.
, QZ y , (T#)	commont	전송데이터의 주소는 QZ 명령어와 QY 명령어에 의하여 설정됨.
	comment	Qz; 또는 Qzy; 명령어를 한번 보내면 전송개시, 한번 더 보내면 전송을 정지함.
		2012년 1월 08일 S/W 버젼부터 적용되며 2ms당 8word를 16byte로 나누어서 보냄.
		2012년 12월 27일 S/W 버젼부터는 Qxyz; 명령어로도 전송을 정지할수 있음.



명령어	구분	설명
		Qy 명령어로 host로 전송하는 8개 변수중에서 뒷쪽 4개의 변수에 대한 주소를 설정함.
		1st word : 다섯번째 변수의 주소
	operation	2nd word : 여섯번째 변수의 주소
		3rd word : 일곱번째 변수의 주소
		4th word : 여덟번째 변수의 주소
	return value	QYHLHL, HLHL, HLHL;
		QRHLHL,HL; 명령어로 설정가능한 변수의 주소범위를 벗어나면 reset됨.
		설정가능한 주소범위는 다음과 같음.
		1st/2nd/3rd/4th word : RAM address (TMS320LF2406A)
;QYHLHL,HLHL,HLH		[0000 - 005F] : memory mapped registers [0060 - 007F] : heap memory
L,HLHL; (+#^)		[0200 - 037F] : dual access RAM [0380 - 03FF] : stack
		[0800 - 0FFF] : single access RAM [7010 - 7531] : peripheral memory
;QY; (+#^)		1st/2nd/3rd/4th : RAM address (TMS320F810/11/12)
	comment	[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
	Commont	[0880 - 783F] : peripheral memory [8000 - 9FFF] : LO/L1 RAM block (.ebss)
		[E000 - FFFF] : HO RAM block (.esysmem)
		[E000 - FFFF]의 주소를 설정하면 실제로는 [3F8000 - 3F9FFF]의 주소에 있는 값을 지칭하도록 내부에서
		처리됨.
		1st/2nd/3rd/4th : RAM address (TMS320F28334/28335)
		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
		[0880 - 793F] : peripheral memory
		[COOO - DFFF] : L4/L5 RAM block (.ebss)



명령어	구분	설명
	operation	QZ,QY 명령어로 주소가 지정된 8word를 binary 값의 형태로 host PC에 전송함.
		RS232/485의 경우 : QyHBLBHBLBHBLBHBLBHBLBHBLBHBLBHBLB;
		CAN의 경우: HBLBHBLBHBLBHBLBHBLBHBLBHBLB
		HBLB 는 binary 형태의 16bit word의 상위바이트 한개, 하위바이트 한개를 가리킴.
		1st 2-byte : QZ로 주소가 지정된 첫번째 변수의 값
		2nd 2-byte : QZ로 주소가 지정된 두번째 변수의 값
;Qy; (+#^)	return value	3rd 2-byte : QZ로 주소가 지정된 세번째 변수의 값
		4th 2-byte : QZ로 주소가 지정된 네번째 변수의 값
		5th 2-byte : QY로 주소가 지정된 첫번째 변수의 값
		6th 2-byte : QY로 주소가 지정된 두번째 변수의 값
		7th 2-byte : QY로 주소가 지정된 세번째 변수의 값
		8th 2-byte : QY로 주소가 지정된 네번째 변수의 값
	comment	HBLB 는 16bit word의 상위,하위바이트를 가리키며 16bit의 binary 값임.



명령어	구분	설명
	operation	operation_status를 읽어서 host로 전송함.
;Q1; (+#^%)	return value	Q1 HLHL; bit0 : motor1_start_to_move_trajectory_flg bit1 : motor1_start_to_move_to_position_flg bit2 : motor1_moving_flg bit3 : motor1_feedback_mode bit4 : motor1_home_searching_status bit0 bit5 : motor1_home_searching_status bit1 bit6 : motor1_home_searching_status bit2 bit7 : motor1_home_searching_status bit3 bit8 : motor2_start_to_move_trajectory_flg bit9 : motor2_start_to_move_to_position_flg bit10 : motor2_moving_flg bit11 : motor2_feedback_mode bit12 : motor2_home_searching_status bit0 bit13 : motor2_home_searching_status bit1 bit14 : motor2_home_searching_status bit2 bit15 : motor2_home_searching_status bit3
	comment	motorx_start_to_move_trajectory_flg==1 이면 PM 명령이 수행중인 상태임. motorx_start_to_move_to_position_flg==1 이면 PP/PQ/Pp/Pq 명령이 수행중 또는 대기중 상태임. motor1_home_searching_status 는 홈동작 진행/성공여부임. (0이면 홈동작명령 실시된 바 없음, 1이면 성공, 2이면 실패, 3이면 리밋 찾는 중, 4이면 홈을 찾는 중, 5이면 홈을 지나치는 중, 6이면 offset 이동중, 7이면 리밋센서가 꺼지기를 기다리는 중 (2개의 리밋센서만 사용할 때), 8이면 홈를 찾는중 (한개의홈센서만 사용할 때), 9이면 홈센서가 꺼지기를 기다리는 중 (한개의 홈센서만 사용할 때) motor1_moving_flg, motor2_moving_flg는 작동모드 1/2/3/4/8에서만 유효함. 2012년 5월 22일 S/W 버젼부터는 작동모드 5/6에서도 유효하며, 모터의 실제움직임을 반영한 값이 아니고, 모터의 속도가 0이 되도록 제어하고 있는 상태임을 나타냄.



명령어	구분	설명
	operation	fault_status, fault1_status를 읽어서 host로 전송함.
;Q2; (+#^%)	return value	Q2 HLH_HLH: 1st word bit0 : motor1의 gate kill fault (DC/BLDC) 1st word bit1 : motor2의 gate kill fault (DC/BLDC) 1st word bit2 : 공급전압이 허용 low voltage 이하로 낮게 공급된 fault (DC/BLDC) 1st word bit3 : 공급전압이 허용 maximum voltage를 초과한 fault (DC/BLDC) 1st word bit4 : motor1의 maximum current를 초과한 fault (DC/BLDC) 1st word bit5 : motor2의 maximum current를 초과한 fault (DC/BLDC) 1st word bit6 : motor1의 제어 오차가 허용치를 넘는 fault (DC/BLDC) 1st word bit7 : motor2의 제어 오차가 허용치를 넘는 fault (DC/BLDC) 1st word bit8 : current1_over_flg (DC/BLDC) 1st word bit9 : current1_over_K_flg (DC/BLDC) 1st word bit10 : motor1_stall_K_flg (DC/BLDC) 1st word bit11 : motor1_stall_K_flg (DC/BLDC) 1st word bit13 : current2_over_flg (DC/BLDC) 1st word bit14 : motor2_stall_K_flg (DC/BLDC) 1st word bit15 : motor2_stall_K_flg (DC/BLDC) 2nd word bit1 : motor1_position_plus_limit_over 2nd word bit2 : motor1_position_minus_limit_over 2nd word bit3 : motor2_position_minus_limit_over 2nd word bit3 : motor2_position_minus_limit_over



명령어	구분	설명
		2nd word bit4 : IRMCK201 fault
		2nd word bit5 : RS232C communication fault
		2nd word bit6 : RS485 communication fault
		2nd word bit7 : CAN communication fault
		2nd word bit8 : 모터1의 권선온도가 ZT 명령어로 설정된 허용최대온도를 초과함.
		2nd word bit9 : 모터2의 권선온도가 ZT 명령어로 설정된 허용최대온도를 초과함.
		2nd word bit10 : 모터공급전압이 Z6로 설정된 전압의 125%를 초과함.
	return value	2nd word bit11: 모터제어기의 bridge1에서 발생하는 발열량이 방열판에서 감당 가능한 범위를 초과함.
		2nd word bit12: 모터제어기의 bridge2에서 발생하는 발열량이 방열판에서 감당 가능한 범위를 초과함.
		2nd word bit13: 12V용 EPS 모터제어기에 있어서 모터1의 과부하로 인하여 모터구동 전압이 10V 이하로
		떨어짐.
		2nd word bit14 : 과전류 검출기능이 있는 모터구동 IC를 사용하는 경우에 한하며, 모터1에서 과전류
		fault가 발생함.
		2nd word bit15 : 과전류 검출기능이 있는 모터구동 IC를 사용하는 경우에 한하며, 모터2에서 과전류 fault가 발생함.
		각각의 fault에 대한 처리는 별도의 document에 기술되었음.
		1st word bit0/1/2/3/4/5/6/7 은 fault 플래그를 지우려면 PRHLHL; 명령어를 보내면 됨.
		1st word bits/9/12/13 은 fault 상태가 제거되고 SI 명령어로 설정되는 시간을 경과하면 자동으로 지워
		집.
		o. 1st word bit10/11/14/15 는 fault 플래그를 지우려면 PPR; 명령어를 보내면 됨.
		2nd word bit0/1/2/3 은 fault 상태가 제거되면 바로 지워짐.
	comment	2nd word bit4 는 IRMCK201에 fault가 발생하였거나 또는 추가의 fault가 있음을 알려주는 비트이며 Q3;
		명령어를 사용하여 fault 상태를 알 수 있음. 이 IRMCK201을 사용한 경우에는 fault 플래그가 한번 발
		생하면 강제로 지우지 않는 한 남아있으며, fault 플래그를 없애고 fault를 복구하려면 PRHLHL; 명령어를
		보내면 됨.
		2nd word bit5/6/7 는 RS232, RS485, CAN 통신과정에서 fault가 발생하였음을 알려주는 비트이며, fault의
		상태를 알려면 EC?; 명령어를 보내면 됨. 이 fault 플래그는 한번 발생하면 강제로 지우지 않는 한 남
		아있으며, fault 플래그를 없애고 fault를 복구하려면 ECR; 명령어를 보내면 됨.
CUBE		133

명령어	구분	설명
		2nd word bit8/9 는 권선의 온도가 허용온도를 초과한 에러로서, 전류제어를 사용하는 상태일때는 전류값
		의 최대치를 줄이는 action을 자동으로 실시하게되며, 온도가 허용온도 이하로 내려가면 지연시간후 에러
		에서 복구됨.
		2nd word bit11/12 는 모터제어기의 발열량이 Z9 명령어로 설정하는 방열판용량으로부터 계산된 발열 허용
		범위를 초과한 에러로서, 전류제어를 사용하는 상태일때는 전류값의 최대치를 줄이는 action을 자동으로
	comment	실시하게되며, 발열량이 허용량 이하로 내려가면 시간지연후 에러에서 복구됨.
		2nd word bit13 은 모터1의 과부하로 인하여 모터구동 전압이 10V 이하로 떨어질 때 발생함. 전류제어
		에 사용하는 전류크기의 제한값 (Sw 명령어로 설정됨)을 점차 작게하며, 모터구동전압이 10V 이상이되면
		회복 시킴.
		2nd word bit14/15 은 과전류 검출기능이 있는 모터구동 IC를 사용하는 경우에 한하며, 과전류 fault 조건
		에서 발생하여 모터구동출력을 off 시킴.



명령어	구분	설명
	operation	IRMCK201의 fault_status를 읽어서 host로 전송함.
		Q3 HLHL;
		bit0 : motor1의 gate kill fault (BLDC IRMCK201)
		bit1 : motor1의 low voltage fault (BLDC IRMCK201)
		bit2 : motor1의 over voltage fault (BLDC IRMCK201)
		bit3 : motor1의 over speed fault (BLDC IRMCK201)
		bit4 : motor1의 execution time fault (BLDC IRMCK201)
		bit5 : motor1의 analog Hall sensor fault (RJM_VER7)
	return value	bit6 : motor1의 encoder fault (RJM_VER7)
;Q3; (+#^%)		bit7 : low control voltage fault (고전압제어기의 경우)
, ((O) (1# /0)		bit8 : motor2의 gate kill fault (BLDC IRMCK201)
		bit9 : motor2의 low voltage fault (BLDC IRMCK201)
		bit10 : motor2의 over voltage fault (BLDC IRMCK201)
		bit11 : motor2의 over speed fault (BLDC IRMCK201)
		bit12 : motor2의 execution time fault (BLDC IRMCK201)
		bit13 : motor2의 analog Hall sensor fault (RJM_VER7)
		bit14 : motor2의 encoder fault (RJM_VER7)
		bit15 : current offset fault at operation mode setting
	comment	각각의 fault나 에러에 대한 처리는 별도의 document에 기술되었음.
		bit5/6/7/13/14/15 와 연관관 오류는 IRMCK201을 사용하지 않는 제어기에서도 발생할수 있는 fault임.



명령어	구분	설명
	operation	프로그램 작성 날짜, version, macro 설정을 읽어서 host로 전송함. (2008년 10월6일 개정)
		Q4 HLHL, HLHL, HLHL;
		1st word high byte : Programming Year
		1st word low byte : Programming Month
	return value	2nd word high byte : Programming Date
		2nd word low byte : Programming Hour
		3rd word : Major Macro Definition
		4th word : Sub Macro Definition
		1st word가 0608 이면 2006년 8월을 가리킴.
		2nd word가 3004 이면 30일의 4시 작성 S/W 버젼임을 가리킴.
		3rd word의 highest 4bit : 1이면 : BIPEDAL_ROBOT 용 S/W
;Q4; (+#^%)		: 2이면 : ROBOT_JOINT_MOTOR 용 S/W-
		: 3이면 : ROBOT_INERTIAL_SENSOR 용 S/W-
		: 4이면 : ROBOT_EYE 용 S/W-
	comment	: 5이면 : SYSTEM MONITORING 용 S/W
		: 6이면 : HUMANOID 용 S/W
		: 7이면 : PTC HEATER 용 S/W
		: 8이면 : MAGNETIC BEARING 용 S/W
		: 9이면 : ABS CONTROL 용 S/W
		: a이면 : RACK type POWER AMP 용 S/W
		3rd word의 next 4bit : 0/1/2/3/4/5/6/7/8/9 :
		TMS320LF2406A / TMS320F2810 / TMS320F2811 or 12 / TMS320LF2403A /
		C8051F411 / TMS320LF2407A / TMS320F28335 / TMS320F28334 / TMS320F2809 /
		C8051F506



명령어	구분	설명
		3rd word의 bit7이 1이면 : RS232C_MODE (F2811/F28334의 경우는 항상 1)
		3rd word의 bit6이 1이면 : RS485_MODE (2811에서 EtherNet-IP or USB 사용시는 선택 불가능)
		3rd word의 bit5이 1이면 : CAN_MODE
		3rd word의 bit4이 1이면 : COM_TEST_MODE
		3rd word의 bit3/2이 3 일때 CAN 11-bit 주소모드로서 주소로 0-255만 사용
		2 일때 CAN 11-bit 주소모드로서 주소로 0-2047 사용, 2012년 10월 16일부터 적용
		1 일때 CAN 29-bit 주소모드로서 주소로 0-65535만 사용, 2012년 10월 16일부터 적용
		0 일때 CAN 11-bit 주소모드로서 주소로 0-255만 사용
		3rd word의 last 2bit : 0/1 : 기타 통신모드 없음/EtherNet-IP or USB (2811/28334 일때만)
		모터제어기의 경우
	comment	4th word의 highest 4bit :
		1/2/3/4/5/6 : ONE_MOTOR_MODE / TWO_MOTOR_MODE / BLDC_ONE_MOTOR_MODE / BLDC_TWO_MOTOR_MODE /
		BLDC_ONE_MOTOR_SENSOR / BLDC_TWO_MOTOR_SENSOR
		4th word의 next 4bit :
		1/2/3/4/5/6/7/8 : RJM_VER1/2/3/4/5/6/ <mark>7/8</mark>
		4th word의 next 4bit :
		1/2/3/4/5/6/7 : VER7_ENCODER+digital Hall / VER7_analog Hall only / VER7_digital Hall only /
		JEIL_MEDICAL / NANO_MOTION_RJM_VER6 / NANO_MOTION_LM_ANALOG / BEST_MOTION
		4th word의 lowest 4bit :
		0/1/2/3/4/5 : CLOCK_60MHz / CLOCK_80MHz / CLOCK_100MHz / CLOCK_120MHz / CLOCK_120MHz_1 /
		CLOCK_150MHzHz / CLOCK_120MHz / CLOCK_120MHz_1 / CLOCK_150MHz



명령어	구분	설명
	comment	관성센서의 경우
		4th word의 highest 4bit :
		1/2/3/4 : Ver1 / Ver2(Analog Device MEMS chip 채용) / Ver3(Fujitsu Gyro, Kionix Acc 채용) /
		Ver4(Fujitsu Gyro, VTI Acc 채용)
		4th word의 lowest 4bit :
		1/2/3/4/5 : CLOCK_80MHz / CLOCK_100MHz / CLOCK_120MHz / CLOCK_120MHz_1 / CLOCK_150MHz



명령어	구분	설명
	operation	operation1_status를 읽어서 host로 전송함.
		Q5 HLHL;
		bit0 : position1_plus_limit_over_flg
		bit1 : position1_minus_limit_over_flg
		bit2 : position1_deceleration_at_limit_flg
		bit3 : speed1_deceleration_at_limit_flg
		bit4 : position1_by_PA_cmd_execute_flg
		bit5:
;Q5; (+#^)		bit6 : linear_interpolation_execute_flg
, WO , (T#)	return value	bit7 : positionX_by_PA_cmd_execute_flg
		bit0 : position2_plus_limit_over_flg
		bit1 : position2_minus_limit_over_flg
		bit2 : position2_deceleration_at_limit_flg
		bit3 : speed2_deceleration_at_limit_flg
		bit12 : position2_by_PA_cmd_execute_flg
		bit13:
		bit14:
		bit15:
	comment	



명령어	구분	설명
	operation	conditional compiling 조건을 읽어서 host로 전송함.
		Q6 HLHL,HLHL; (2011년 9월 25일 이전 S/W 버젼)
		Q6 HLHL,HLHL; (2011년 9월 25일 S/W 버젼부터)
		1st word의 highest 4bit : ENCODER1_SCALE_FACTOR (모터1의 실제 엔코더 펄스수는 MOTOR1_PULSE_PER_REV
		* ENCODER1_SCALE_FACTOR 이며 ENCODER1_SCALE_FACTOR는 600, 7500, 72000을 나누어서 나누어지는
		수이어야하며, 통상값은 1로서 엔코더의 펄스수가 32767을 넘는 경우에만 1이 아닌 값을 설정함)
		1st word의 next 4bit : ENCODER2_SCALE_FACTOR (모터2의 실제 엔코더 펄스수는 MOTOR2_PULSE_PER_REV
		* ENCODER2_SCALE_FACTOR 이며 ENCODER2_SCALE_FACTOR는 600, 7500, 72000을 나누어서 나누어지는
		수이어야하며, 통상값은 1로서 엔코더의 펄스수가 32767을 넘는 경우에만 1이 아닌 값을 설정함)
		1st word의 next 4bit : SEQUENCE_CONTROL (0 = no sequence control program, 1 = sequence control
		program, 2 = 프로와텍)
		1st word의 last 4bit : RJM_VER7_UV_current (RJM_VER7만 해당, 0 = no U/V current sensing,1 = U/V
;Q6; (+#^)		current at AD3/4/11/12, 2 = U/V/W current at AD13/14/15, 3 = same as 1 but trapezoidal, 4 =
	return value	same as 2 but trapezoidal, 5 = same as 1 but total current + trapezoidal, 6 = same as 2 but
		total current + trapezoidal, 7 = 5상 스테핑모터의 경우, 8 = 삼상 스테핑모터의 경우)
		2nd word의 bit15 : FAULHABER_TYPE (0 = U 단자대 사용형식, 1 = Faulhaber type 스크류 터미널 형태)
		2nd word의 bit14: NO_DIP_SWITCH (0 = with DIP switch, 1 = no DIP switch)
		2nd word의 bit13 : SWAP_LIMIT_SW (RJM_VER3 TMS320F2810/11으로서 2007년 8월경에 제작된 모델에만
		해당, 0 = original type, 1 = swap limit switch1/switch2)
		2nd word의 bit12 : ADS7818 (RJM_VER3만 해당, 0 = 없는 경우, 1 = 있는 경우)
		2nd word의 bit11 : CURRENT_LOOP_50KHZ (RJM_VER5/6/4/8만 해당, 0 = 1KHz(RJM_VER5) /
		100KHz(RJM_VER4/8) CURRENT_LOOP, 1 = 50KHz(RJM_VER5/6 TWO MOTOR MODE, RJM_VER4/8) /
		100KHz(RJM_VER5/6 ONE MOTOR MODE) CURRENT_LOOP)
		2nd word의 bit10 : STEP_VER1PO (RJM_VER5만 해당, 0 = I_FAULTx at F8/F9, 1 = I_FAULTx at E0/E1)
		2nd word의 bit9 : RJM_VER7_L6234 (RJM_VER7만 해당, 0 = no L6234, 1 = L6234)
		2nd word의 bit8 : NEW_GAIN_RULE (0 = old gain, 1 = new gain)



명령어	구분	설명
		2nd word의 bit7/6 : ANGLE_RESOLVING_METHOD (RJM_VER7 analog type만 해당, 0 = no analog Hall sensor,
		1 = 아크사인/아크코사인법, 2 = 직선보간법, 3 = COS/SIN resolver)
		2nd word의 bit5 : WAFER_ROTATION (0 = non wafer control, 1 = NANO motion wafer control, RJM_VER7
		digital Hall IC only 인 경우에만 해당)
		2nd word의 bit4 : DD_MOTOR (0 = Direct Drive Motor가 아닌 경우, 1 = Direct Drive Motor인 경우)
		2nd word의 bit3 : AC_INDUCTION_MOTOR (0 = AC induction motor가 아닌 경우, 1 = AC induction motor인
		경우)
		2nd word의 bit2 : NO_HALL_BUT_HOME_SENSOR (0 = 통상의 경우, 1 = Encoder를 사용하지만 Hall IC는 없고
		대신하여 홈센서를 사용하는 경우)
		2nd word의 bit1 : HIGH_RESOLUTION_ANGLE (0 = 통상의 경우, 1 = sin/cos 계산시에 분해능을 16배 높임)
		2nd word의 bit0 : AUTOMATION (0 = 통상의 경우, 1 = 자동화용 H/W 버젼의 경우)
	return value	3rd word의 bit15 : RJM_VER7_MODULE (0 = 통상의 경우, 1 = RJM_VER7 MOSFET 모듈을 사용한 H/W 버젼으로
	Totalli value	서 DC 제어용으로 프로그램된 경우)
		3rd word의 bit14 : BIPOLAR_DRIVE (0 = 통상의 경우, 1 = DC 모터로서 bipolar 구동을 하는 경우)
		3rd word의 bit13 : RJM_VER4_PWM_AT_CH1245 (0 = PWM at PWM1+/PWM1-/PWM2+/PWM2-, 1 =
		PWM1+/PWM2+/PWM5+)
		3rd word의 bit12 : VERSION_380V (0 = IRAM136-3063B, 1 = IXYS-MUBW)
		3rd word의 bit11 : EtherNetIP_or_USB (0 = 통상의 경우, 1 = LAN/USB 를 사용하는 경우)
		3rd word의 bit10 :
		3rd word의 bit9 : DIFFERENTIAL_HALL_IC (0 = 통상의 경우, 1 = 380V H/W 버젼에서 differential Hall IC
		를 사용하는 경우)
		3rd word의 bit8 : RJM_VER7_Hall_IC_ENC (0 = 통상의 경우, 1 = BLDC에서 엔코터를 위치센서로 사용하는
		경우)
		3rd word의 bit7 : RJM_VER7_Hall_IC_analog (0 = 통상의 경우, 1 = BLDC에서 analog Hall 센서를 위치센
		서로 사용하는 경우)



명령어	구분	설명
		3rd word의 bit6 : RJM_VER7_Hall_IC_only (0 = 통상의 경우, 1 = BLDC에서 digital Hall 센서를 위치센서
		로 사용하는 경우)
		3rd word의 bit5/4 : ABSOLUTE_ENC (0 = 통상의 경우, 1 = Netzer 19bit Absolute Encoder 위치센서로 사
		용하는 경우)
		3rd word의 bit3 :
	return value	3rd word의 bit2 : LOW_RESOLUTION_ENCODER (0 = 통상의 경우, 1 = low resolution encoder를 위치센서로
	Teturn varue	사용하는 경우)
		3rd word의 bit1/0 : POTENTIOMETER (0 = potentiometer를 위치센서로 사용하지 않는 경우, 1 = LSxC 포트
		에 연결된 potentiometer 를 사용한 위치제어의 경우, 2 = digital potentiometer를 사용하는 경우)
		4th word의 highest 8bit : ENCODER1_SCALE_FACTOR
		4th word의 next 8bit : ENCODER2_SCALE_FACTOR
	comment	



명령어	구분	설명
		지정된 주소에서 시작하여 지정된 양만큼의 DSP SRAM 내용을 읽어서 host로 전송함.
		first word : RAM address (TMS320LF2406A)
		[0000 - 005F] : memory mapped registers [0060 - 007F] : heap memory
		[0200 - 037F] : dual access RAM [0380 - 03FF] : stack
		[0800 - 0FFF] : single access RAM [7010 - 7531] : peripheral memory
		first word : RAM address (TMS320F810/11/12)
	operation	[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
	operation	[0880 - 783F] : peripheral memory [8000 - 9FFF] : LO/L1 RAM block (.ebss)
		[E000 - FFFF] : HO RAM block (.esysmem)
;QRHLHL,HL;		first word : RAM address (TMS320F28334/28335)
(+#^%)		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
;QQHLHL,HL;		[0880 - 793F] : peripheral memory
(+#^)		[COOO - DFFF] : L4/L5 RAM block (.ebss)
		second byte : number of words [00 - 07]
	return value	QRHLHL, HL, HLHL(, HLHL);
	Tetuin varue	QQHLHL(,HLHL);
		return 값은 입력으로 들어온 값에다 RAM을 읽은 값들이 뒤에 덛붙여짐.
	comment	number of words 가 00이면 한 개의 word를 의미하며, 07일 때 최대 8 word 까지 읽을 수 있음.
		7010(0880) - 7531(783F) 영역의 주소에 있는 데이터를 읽을 때, 주소값이 정확치 않으면 DSP가 reset됨.
		TMS320F810의 경우 [E000 - FFFF] 주소 영역은 실제주소로는 [3F8000 - 3F9FFF] 주소영역에 해당함.
		[0600 - 07FF], [E000 - FFFF] 영역은 cpu_assm.asm에서 선언된 변수들이 저장되는 영역임.
		[8000 - 9FFF] 영역은 C에서 선언된 전역변수들이 저장되는 곳임.



명령어	구분	설명		
		지정된 주소에서 시작하여 지정된 양만큼의 데이터를 DSP의 SRAM에 전송함.		
		first word : RAM address (TMS320LF2406A)		
		[0800 - 0FFF] : single access RAM [7010 - 7531] : peripheral memory		
		first word : RAM address (TMS320F810/11/12)		
		[0600 - 07FF] : M1 RAM block (.bss) [0880 - 783F] : peripheral memory		
		[8000 - 9FFF] : LO/L1 RAM block (.ebss) [E000 - E0FF] : HO RAM block (.esysmem)		
	operation	[E200 - FFFF] : HO RAM block (.esysmem)		
		first word: RAM address (TMS320F28334/28335)		
		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)		
· OTI		[0880 - 793F] : peripheral memory		
;QTHLHL,HL,HLHL		[COOO - DFFF] : L4/L5 RAM block (.ebss)		
(,HLHL); (+#^%)		second byte : number of words [00 - 07]		
		following word : DSP의 SRAM에 쓸 데이터.		
	return value	QTHLHL, HL, HLHL(, HLHL);		
		return 값은 RAM에 값을 설정한 후 다시 읽은 값들이 뒤에 덛붙여짐.		
		number of words 가 00이면 한 개의 word를 의미하며, 07일 때 최대 8 word 까지 쓸 수 있음.		
		7010(0880) - 7531(783F) 영역의 주소에 데이터를 쓸 때, 주소값이 정확치 않으면 DSP가 reset됨.		
	comment	TMS320LF2406A의 경우 0800 - 0FFF 사이는 변수값이 저장되어 있는 장소로서 값을 쓰려면 주의를 요함.		
		TMS320F810의 경우 [E000 - FFFF] 주소 영역은 실제주소로는 [3F8000 - 3F9FFF] 주소영역에 해당함.		
		[0600 - 07FF], [E000 - FFFF] 영역은 cpu_assm.asm에서 선언된 변수들이 저장되는 영역임.		
		[8000 - 9FFF] 영역은 C에서 선언된 전역변수들이 저장되는 곳임.		



명령어	구분	설명
		BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의 지정된 주소로부터 register 값을 읽어서 host
	operation	로 전송함. (BLDC 구동 chip IRMCK201 전용)
	operation	first byte : register address [00 - 43]
		second byte : motor number [00 - 01]
;QrHL,HL; (+#^%)	return value	QrHL,HLHL;
		return 값은 입력으로 들어온 값에다 IRMCK201 register를 읽은 값이 뒤에 덛붙여짐.
	comment	register에서 읽힌 값이 DSP의 SARAM에 저장되지는 않음.
	Comment	return된 세번째 word의 low byte가 첫번째 바이트로 지정된 주소의 register값임.
		BLDC인 경우에 한하여만 사용이 가능한 명령임.
		BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의 지정된 주소에 host로부터 전송된 데이터를 저
;QtHL,HL,HL;		장함과 동시에 DSP의 SARAM 0x0a3e-0x0abd 번지에도 저장함. (BLDC 구동 chip IRMCK201 전용)
(+#^%)	operation	first byte : register address [00 - 70]
(111 /0)		second byte : motor number [00 - 01]
;QtHL,HL,HLHL;		third byte : register에 저장할 값 [byte 또는 word]
(+#^%)	return value	QtHL,HL,HL; 또는 QtHL,HL,HLHL;
(1π /0)		return 값은 호스트에서 입력으로 들어온 값임.
;QtHL,HL?;		00-7D번지 값은 DSP의 SARAM 0x0a3e-0x0abd 번지에도 저장됨.
(+#^%)	comment	?인 경우는 DSP의 SARAM 0x0a3e-0x0abd 번지에 저장된 word로 echo를 하며, echo된 word의 low byte가 첫
(π /0)		번째 바이트로 지정된 주소의 값에 해당함.
		BLDC인 경우에 한하여만 사용이 가능한 명령임.



Qr. Qt 명령을 사용한 IRMCK201 BLDC motor 구동 IC의 parameter의 읽기와 쓰기

- ◎ BLDC모터1/2의 parameter는 DSP의 SARAM 0x0a3e-0x0abd번지 (F2811의 경우는 0x3f823e-0x3f82bd, F28334의 경우는 0xE23e-E2bd) 에 IRMCK201의 register 0x00-0x7d 번지에 해당하는 값이 각각 저장된다.
- ◎ BLDC모터1/2의 Hall sensor에 대한 Z pulse의 offset(ZEntCnt)는 같은 종류의 모터라도 모터의 조립상태마다 다른 값인데 DSP의 SARAM 0x0a41/0x0a81번지 (F2811의 경우는 0x3f8241/0x3f8281, F28334의 경우는 0xE241/E281) 에 저장된다.
- ◎ BLDC모터1/2의 parameter는 Qtxx,xx,xx; 또는 Qtxx,xx,xxxx; 명령어로 바꿀수가 있으며 이때 DSP의 SARAM OxOa3e-OxOabd번지 (F2811의 경우는 0x3f823e-0x3f82bd, F28334의 경우는 0xE23e-E2bd) 뿐아니라 IRMCK201의 write register 0x00-0x7d 번지로도 값이 각각 저장된다. DSP의 EEPROM에 저장하려면 ESOAOO,0100; 명령어를 사용하여야한다.
- ◎ BLDC모터1/2의 parameter는 Qrxx,xx; 명령어로 읽을 수가 있으며 IRMCK201의 read register 0x00-0x43 번지의 값이 읽혀져서 host로 전송된다. 이때 DSP의 SARAM에 저장되지 않으며 IRMCK201의 read register의 주소는 IRMCK201의 write register 주소와 그 내용이 다르므로 주의하여야 한다.
- ◎ 전원이 최초 투입되면 BLDC모터의 parameter는 EEPROM에서 DSP의 0x0a3e-0x0abd번지 (F2811의 경우는 0x3f823e-0x3f82bd, F28334의 경우는 0xE23e-E2bd)로 읽혀들인 다음, 그 중에서 일부 parameter가 IRMCK201의 register 0x00-0x7d 번지로 각각 저장된다.
- ◎ DSP의 SARAM 0x0a3e-0x0abd번지 (F2811의 경우는 0x3f823e-0x3f82bd, F28334의 경우는 0xE23e-E2bd)에 있는 값은 QT 명령어로 강제 설정하지 않는 한 IRMCK201의 register에 있는 값과 항상 같다.



세어기 명령어 LIST

address	register name	description	비고
0x00	EncCntR(L)	Current value of 16-bit Quadrature Decoder counter.	
0x01	EncCntR(M)	Current value of 10-bit Quadrature becoder counter.	
0x03	EncStatus	spare Pwr0nHallC Pwr0nHallB Pwr0nHallA spare HallC HallB HallA (LSB)	
0x07	SysStatus	Start Stop spare PwrID PwrID GateKill FocEnbR PwmEnbR (LSB)	
0x08	RevCode(L)	IC Revision Code. Revision code format is "XX.XX", where each "X" is a 4-bit	
0x09	RevCode(M)	hexadecimal number.	
0x0a	DcBusStatus(L)	DC Bus Voltage. Data range is 0 - 4095, which corresponds to a DC bus voltage	
0x0b	DcBusStatus(M)	between 0 and 500 volts. 0x0b bit4 : Brake signal status. 0 = Brake signal active.	
0x0c	IfbVRawCurr(L)	Offset-corrected V and W phase raw current from the IR2175 current sensor. Values	
0x0d	IfbWRawCurr(L)	range from 0 - 4096, where 2048 corresponds to 0 current. The current feedback	
0x0e	fbVRawCurr(M) fbWRawCurr(M)	scale factors IdScI and IqScI in the CurrentFeedbackConfig write register group and the current sense resistor value determine the full scale current value.	
0x0f	Id(L)	Synchronous or rotating frame direct and quadrature current values in 2's complement	
0x10	ld(M)	representation. The full scale current values range from -16384 to 16383.	
0x11	Iq(L)	Synchronous or rotating frame direct and quadrature current values in 2's complement	
0x12	Iq(M)	representation. The full scale current values range from -16384 to 16383.	
0x13	Ud(L)	Synchronous or rotating frame direct and quadrature voltage values in 2's complement	
0x14	Ud(M)	representation. Data ranges are \pm VdLim for Ud and \pm VqLim for Uq.	
0x15	Uq(L)	Synchronous or rotating frame direct and quadrature voltage values in 2's complement	
0x16	Uq(M)	representation. Data ranges are \pm VdLim for Ud and \pm VqLim for Uq.	
0x17	UaStatFrmVolt(L)	Stationary frame Alpha and Beta voltage output component values.	
0x18	UbStatFrmVolt(L)	Data range is ± VdLim or ± VqLim (as specified in the CurrentLoopConfig write	
0x19	UaStatFrmVolt(M) UbStatFrmVolt(M)	register group), whichever is larger.	



세어기 명령어 LIST

address	register name	description	비고
0x1e	FItStatus	spare spare ExecTmFlt OvrSpdFlt OvFlt LvFlt GatekillFlt (LSB)	
0x26	Spd(L)	Current motor speed in SPEED units.	
0x27	Spd(M)	(See the description of SpdScl in the VelocityControl write register group.)	
0x30	IfbV0ffset(L)	Current feedback offset values from the last IFB Offset calculation. These values	이 값을 읽어서
0x31	IfbWOffset(M)	are automatically applied to each current feedback measurement value whenever the	write resister
0x32	fbVOffset(L) fbWOffset(M)	IfbOffsEnb bit in the SystemControl write register group is set.	0x10/11/12에 쓰고 저장하여야함.
0x34	EncCnt32bR_LW(L)		
0x35	EncCnt32bR_LW(M)	Current value of 32-bit Quadrature Decoder counter.	
0x36	EncCnt32bR_HW(L)	Current value of 32-bit Quadrature becoder counter.	
0x37	EncCnt32bR_HW(M)		
0x38	EeStatus	0x38 bit0 : I2C EEPROM Interface busy bit.	
0x39	EeDataR	EEPROM Data Register. Contains the data from the last EEPROM read operation.	
0x3a	EeAddrR	EEPROM Address read register shows the value stored in EEPROM at the offset of the EeAddrW write register (0x5D).	
0x3b	RegMapVer	Current register map version code.	
0x3c	ElecAngleStatus(L)	Electrical angle.	
0x3d	ElecAngleStatus(M)	crectificat angle.	
0x3e	SpdRef(L)	Speed PI controller reference input.	
0x3f	SpdRef(M)	Speed Fi Controller reference input.	
0x40	SpdErr(L)	Speed PI controller error.	
0x41	SpdErr(M)	opeed it controller error.	
0x42	IqRefR(L)	Speed PI controller output.	
0x43	IqRefR(M)	opeed it controller output.	



제어기 명령어 LIST

address	register name	EC 45 50W flat 모터	EC-powermax 30 100W	MAXON 136210 250W	FAULHARBER 4490 200W	
0x00	EncCntW(L)	0 (0x0000)	0 (0x0000)	0 (0x0000)	0 (0x0000)	Hall sensor의 값에 따라서 반드시 수동
0x01	EncCntW(M)					초기화하고 저장해야 함.
0x03	MaxEncCnt(L)	4095 (0x0fff)	3999 (0x0f9f)	1999 (0x07cf)	1999 (0x07cf)	4체배된 값 - 1
0x04	MaxEncCnt(M)	4095 (0X0111)	0999 (000191)	1999 (0x0/01)	1999 (0x0761)	4세메근 W 1
0x06	ZEncCnt(L)	277 (0x0115)	1583 (0x062f)	1833 (0x0729)	1750 (0x06d6)	Z 펄스의 위치값
0x07	ZEncCnt(M)	211 (0.0113)	1303 (0,0021)	1000 (0x0729)	1750 (0x0000)	2 글_ᅴ 커짒
0x09	EncAngScI(L)	32767 (0x7fff)	8389 (0x20c5)	8389 (0x20c5)	8389 (0x20c5)	
0x0a	EncAngScl(M)	02707 (0X7111)	0009 (0/2003)	0009 (0/2003)	0009 (0x2003)	
0x0b	EncCtr I	15 (0x0f)	15 (0x0f)	15 (0x0f)	15 (0x0f)	
0x0c	PwmCtrl	178 (0xb2) or 50(0x32)	50(0x32)는 gatekill 기능을 안쓰는 경우.			
0x0d		4296 (0x10c8)	4296 (0x10c8)	4296 (0x10c8)	4296 (0x10c8)	
0x0e	PwmPeriodConfig	PwmPeriod : 200 PwmConfig : 1				
0x0f	PwmDeadTm	20(0x14)	50 (0x32)	50 (0x32)	50 (0x32)	20 : 0.6 microsec
0x10	IfbOffsetV(L)					
0x11	IfbOffsetW(L)	IfbOffsetV: 0	IfbOffsetV: 0	IfbOffsetV: 0	IfbOffsetV: 0	반드시 수동 초기화
0x12	fbOffsetV(M) fbOffsetW(M)	IfbOffsetW : 0	IfbOffsetW : 0	IfbOffsetW : 0	IfbOffsetW : 0	하고 저장해야함.
0x13	IdScI(L)	5405 (0x151d)	4019 (0x0fb3)	1134 (0x046e)	3486 (0x0d9e)	
0x14	IdScI(M)	J-10J (UNIJIU)	HUID (UNUIDO)	1104 (040406)	OHOU (UNUUSE)	
0x15	IqScI(L)	5405 (0x151d)	4019 (0x0fb3)	1134 (0x046e)	3486 (0x0d9e)	
0x16	IqScI(M)	SHOS (UXISIU)	4019 (000100)	1134 (UXU40e)	OHOO (UXUUSE)	
0x17	SysCtrl	0	0	0	0	



제어기 명령어 LISI

address	register name	EC 45 50W flat 모터	EC-powermax 30 100W	MAXON 136210 250W	FAULHARBER 4490 200W	
0x18	IqRefW(L)	0	0	0	0	
0x19	lqRefW(M)	U	U	0	U	
0x1a	Kplreg(L)	2037 (0x07f5)	755 (0x02f3)	1219 (0x04c3)	1067 (0x042b)	
0x1b	Kplreg(M)	2007 (0x0713)	733 (0x0213)	1219 (080463)	1007 (0x0420)	
0x1c	KxIreg(L)	899 (0x0383)	476 (0x01dc)	595 (0x0253)	641 (0x0281)	
0x1d	Kxlreg(M)	099 (0,0000)	470 (000100)	393 (0X0230)	041 (0x0201)	
0x1e	IdRef(L)	0	0	0	0	
0x1f	ldRef(M)	U	U	U	U	
0x20	SlipGn(L)	0	0	0	0	
0x21	SlipGn(M)	U				
0x22	VqLim(L)	914 (0x0392)	1421 (0x058d)	1428 (0x0594)	1429 (0x0595)	
0x23	VqLim(M)	914 (0x0092)				
0x26	VdLim(L)	0 (0x00c0)	0 (0x00c0)	0 (0x00c0)	0 (0x00c0)	
0x27	VdLim(M)	0 (0x0000)				
0x31	VelCtrl	15 (0x0f)	15 (0x0f)	15 (0x0f)	13 (0x0d)	15 : SpdLpRate 7
0x32	KpSreg(L)	392 (0x0188)	57 (0x0039)	187 (0x0066)	187 (0x0066)	
0x33	KpSreg(M)	092 (000100)		107 (0x0000)		
0x34	KxSreg(L)	169 (0x00a9)	21 (0x015)	81 (0x0051)	69 (0x0045)	
0x35	KxSreg(M)	TOO (UNUUDO)	Z1 (UNU13)	01 (00001)	0 3 (0x0043)	
0x36	SregLimP(L)	5000 (0x1388)	12000 (0v2aa0)	8000 (0x1f40)	8000 (0x1f40)	
0x37	SregLimP(M)	3000 (0x1000)	12000 (0x2ee0)	0000 (UX1140)	0000 (UX1140)	
0x38	SregLimN(L)	-5000 (0xec78)	-12000 (0xd120)	-8000 (0xe0c0)	9000 (040000)	
0x39	SregLimN(M)	J000 (0x80/0)	12000 (OXU120)	OOOO (OVGOOO)	-8000 (0xe0c0)	



세어기 명령어 LIST

address	register name	EC 45 50W flat 모터	EC-powermax 30 100W	MAXON 136210 250W	FAULHARBER 4490 200W	
0x3a	SpdScI(L)	30517 (0x7735)	31250 (0x7a12)	62499 (0xf423)	62499 (0xf423)	TargetSpd 값이 rpm
0x3b	SpdScl(M)	30317 (0X7733)	31230 (UX/a12)	02499 (0X1423)	02499 (0X1423)	이 되도록 설정함.
0x3c	TargetSpd(L)	0	0	0	0	[rpm]
0x3d	TargetSpd(M)	U	U	U	U	[I DIII]
0x3e	SpdAccRate	16 (0x10)	64 (0x40)	32 (0x20)	32 (0x20)	작으면 위치제어 문
0x3f	SpdDecRate	16 (0x10)	64 (0x40)	32 (0x20)	32 (0x20)	제 생김. 1=83/SpdLpRate [Krpm/s]
0x42	FaultCtrl	2,1	2,1	2,1	2,1	
0x44	ModScI(L)	601 (0x0259)	601 (0x0259)	601 (0x0259)	601 (0x0259)	
0x45	ModScl(M)	001 (0x0259)	001 (0x0259)	001 (0x0259)	001 (0x0239)	
0x4e	PwmDataSel(L)		0	0	0	
0x4f	PwmDataSel(M)	0				
0x72	HallCBA001(L)	333 (0x014d)	333 (0v014d)	333 (0x014d)	667 (0x029b)	
0x73	HallCBA001(M)	00 (00000)	333 (0X014u)			
0x74	HallCBA010(L)	427 (0x01ab)	1667 (0x0683)	1667 (0x0683)	1333 (0x0535)	
0x75	HallCBA010(M)	427 (UNUTAD)	1007 (00000)	1007 (00000)	1000 (0x0000)	
0x76	HallCBA011(L)	0 (0x0000)	0 (0x0000)	0 (0x0000)	1000 (0x03e8)	
0x77	HallCBA011(M)	0 (00000)	0 (0x0000)	0 (0,0000)	1000 (0x00e0)	
0x78	HallCBA100(L)	256 (0x0100)	1000 (0x03e8)	1000 (0x03e8)	0 (0x0000)	
0x79	HallCBA100(M)	250 (0x0100)	1000 (0x03e0)	1000 (0x03e6)	0 (0x0000)	
0x7a	HallCBA101(L)	171 (0x00ab)	667 (0x029b)	667 (0x029b)	333 (0x014d)	
0x7b	HallCBA101(M)	171 (ΟλΟυαυ)	OUT (UNULBD)	007 (000230)	000 (000140)	
0x7c	HallCBA110(L)	341 (0x0155)	1333 (0x0535)	1333 (0x0535)	1667 (0x0683)	
0x7d	HallCBA110(M)	OTI (UXUIJU)	1000 (08000)	1000 (08000)	1007 (00000)	



세어기 명령어 LISI

address	register name	EC 45 50W flat 모터	EC-powermax 30 100W			
0x50	SysConfig	0	0	0	0	
0x52	VdSfwdCtrl(L)					
0x53	VqSfwdCtrI(L)					
0x54	VdSfwdCtrl(M) VqSfwdCtrl(M)					
0x55	ElecAngW(L)					
0x56	ElecAngW(M)					
0x58	EncCnt32bW_LW(L)					
0x59	EncCnt32bW_LW(M)					
0x5a	EncCnt32bW_HW(L)					
0x5b	EncCnt32bW_HW(M)					
0x5c	EeCtrl					
0x5d	EeAddrW					
0x5e	EeDataW					

명령어	구분	설명
		모터의 PWM duty 출력을 강제로 설정함.
	operation	1st integer : motor1의 PWM duty값 [-PWM_PERIOD ~ PWM_PERIOD] (0)
		2nd integer : motor2의 PWM duty값 [-PWM_PERIOD ~ PWM_PERIOD] (0)
;PWiiii1,iiii2;	return value	PWiiii1,iiii2;
(%)		PWM 주기값은 -PWM_PERIOD에서 PWM_PERIOD 범위에서만 유효한 값이며 PWM_PERIOD는 SG 명령으로 설정 가능
(10)		함. 단 SG 명령을 사용할 때는 특별한 sequence로 명령어를 내려보내야 하며 ST 명령어의 comment를 참
;PW?; (%)		조할 것.
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	comment	SM 명령에 의한 작동 모드가 0000일 때만 PW 명령으로 CMPR1/2_reg의 값에 적용되는데, CMPR1/2_reg의 값
		들을 직접 읽으려면 QR7417,01; 명령을 사용하면 됨.
		PW 명령어로 함부로 값을 설정하면 모터나 모터제어기에 손상이 갈수 있으므로 사용에 주의를 요함.
		iiii1,iiii2는 부호가 있는 십진수임.
;PWiiii1,iiii2;		모터의 PWM duty 출력을 강제로 설정함.
(VER4/8) (+#^)		1st - 6th integer : motor의 PWM duty값 [-PWM_PERIOD ~ PWM_PERIOD] (0)
	operation	VER4/8의 경우 : 1st/2nd 는 모터1/2의 PWM duty 설정값
;PWiiii1,iiii2,i		VER5의 경우: 1st/2nd 는 모터1의 PWM duty 설정값, 3rd/4th 는 모터2의 PWM duty 설정값
1113,11114;		VER6/7의 경우 : 1st/2nd/3rd 는 모터1의 PWM duty 설정값, 4th/5소/6th 는 모터2의 PWM duty 설정값
(VER5) (+#^)	return value	PWiiii1,iiii2; or PWiiii1,iiii2,iiii3,iiii4; or PWiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;
;PWiiii1,iiii2,i		PWM 주기값은 -PWM_PERIOD에서 PWM_PERIOD 범위에서만 유효한 값임.
iii3,iiii4,iiii		SM 명령에 의한 작동 모드가 0000일 때만 PW 명령으로 CMPR1/2/3/4/5/6_reg의 값에 적용되는데,
5, i i i i i 6;	comment	PW 명령어로 함부로 값을 설정하면 모터나 모터제어기에 손상이 갈수 있으므로 사용에 주의를 요함.
(VER6/7) (+#^)	Commert	CMPR1/2/3/4/5/6_reg의 값들을 직접 읽으려면 QR7417,02; 및 QR7517,02; 명령을 사용하면 됨. (2811의 경
() () () () ()		우) TMS320F28334/5의 경우는 주소가 달라서 6809/6849/6889/68C9/6909/6949 주소의 값을 읽으면 됨.
;PW?; (+#^)		iiii1, iiii2, iiii3, iiii4, iiii5, iiii6는 부호가 있는 십진수임.



명령어	구분	설명
	operation	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값을 설정함. 1st long integer : motor1의 변위명령값 [position1_minus_limit ~ position1_plus_limit] or 속도명령값 [5000000-32000/60000 ~ 5000000+32000/60000] or 전류명령값 [5000000-1023/4095 ~ 5000000+1023/4095] 2nd long integer : motor2의 변위명령값 [position2_minus_limit ~ position2_plus_limit] or 속도명령값 [5000000-32000/60000 ~ 5000000+32000/60000] or 전류명령값 [5000000-1023/4095 ~ 5000000+1023/4095]
	return value	PAIIII1, IIII2;
;PAIIII1,IIII2; (+#^%) ;PA?; (+#^%)	comment	position mode 로 설정된 모터는 변위명령값으로 설정되며, motor1의 경우는 position1_minus_limit에서 position1_plus_limit 범위에서만 유효하고, motor2의 경우는 position2_minus_limit에서 position2_plus_limit 범위에서만 유효함. 이 경우는 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2; 명령어로 설정된 속도로 이동함. 변위명령값은 5000000 이 기본값으로서 원점위치를 의미하며 항상 양수이어야함. speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을 5000000-32000/60000 ~ 5000000+32000/60000 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범위에서 설정됨. current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 5000000-1023/4095 ~ 5000000+1023/4095 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨. 이 명령어는 십진수를 long integer로, long integer를 십진수로 바꾸는 32bit 나누기 연산을 여러번 수행하므로, 명령어 처리에 약 300microsec(return을 안받으면 150micros)의 시간이 더 걸리기 때문에, PA 대신 PB 나 Pa 명령어를 사용하는 것이 더 효과적임.



명령어	구분	설명	
		모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값을 설정함.	
		1st long integer : motor1의 변위명령값 [00000000 - FFFFFFFF]	
		or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60]	
	operation	or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF]	
		2nd long integer : motor2의 변위명령값 [00000000 - FFFFFFFF]	
		or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60]	
;PBHLHLHLHL,HLHL		or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF]	
HLHL; (+#^%)	return value	PBHLHLHLH, HLHLHLHL;	
		전송데이터 형식과 설정값 범위를 빼고는 PA명령과 같음.	
;PB?; (+#^%)		position mode 로 설정된 모터는 변위명령값으로 설정되며 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2;	
		명령어로 설정된 속도로 이동함.	
	oommon t	speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을	
	comment	004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범위	
		에서 설정됨.	
		current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 004C4B40-3FF/FFF ~	
		004C4B40+3FF/FFF 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨.	



명령어	구분	설명		
	operation	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값을 설정함. 1st 24-bit long word: motor1의 변위명령값 [position1_minus_limit ~ position1_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] 2nd 24-bit long word: motor2의 변위명령값 [position2_minus_limit ~ position2_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] Pa 와; 를 빼고는 binary 값임.		
;PaHBMBLBHBMBLBC S; (+#^%)	return value	PaHBMBLBHBMBLBCS; HBMBLB 는 binary 형태의 24bit word를 가리키며 상위,중간,하위바이트를 가리킴. CS 는 앞선 6byte의 합의 inverse를 binary 한 바이트로 표현한 것임. return 값은 현재의 위치값임. checksum이 일치되지 않으면 수신값을 무시하고 return 값의 마지막은 :임. 이 경우 내부에서 수신에러 카운트 Pa_receive_error_cnt를 하나 증가시킴.		
	comment	전송데이터 형식을 빼고는 PA명령과 같음. position mode 로 설정된 모터는 변위명령값으로 설정되며 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2; 명령어로 설정된 속도로 이동함. 변위명령값은 4C4B40 이 기본값으로서 원점위치를 의미하며 항상 양수이어야함. speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을 004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범위에서 설정됨. current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨.		



명령어	구분	설명
		위치 명령값의 증분을 설정하고 이동을 명령함.
	operation	1st integer : motor1의 변위명령값의 증분 [-2147483648 ~ 2147483647]
		2nd integer : motor2의 변위명령값의 증분 [-2147483648 ~ 2147483647]
		PIIII1, IIII2;
;P111111,11112;	return value	1st long integer : motor1의 변위명령값
(+#^)		2nd long integer : motor2의 변위명령값
(1#)		position mode 로 설정된 경우에 한하며, motor1의 경우는 position1_minus_limit에서
;PI?; (+#^)		position1_plus_limit 범위에서만 유효하고, motor2의 경우는 position2_minus_limit에서
)FI:) (T#)		position2_plus_limit 범위에서만 유효함. 이 경우는 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2;
	comment	명령어로 설정된 속도로 이동함.
		리턴값은 증분값이 아니고 증감된 현재의 변위명령값임.
		이 명령어는 십진수를 long integer로, long integer를 십진수로 바꾸는 32bit 나누기 연산을 여러번 수행
		하므로, 명령어 처리에 약 300microsec(return을 안받으면 150micros)의 시간이 걸림.

명령어	구분	설명
	operation	모터 1에 대하여 위치 명령값의 증분을 설정하고 이동을 명령함.
	operation	1st integer : motor1의 변위명령값의 증분 [-2147483648 ~ 2147483647]
	return value	P111111;
;P1 1; (+#^)	return varue	1st long integer : motor1의 변위명령값
		position mode 로 설정된 경우에 한하며, position1_minus_limit에서 position1_plus_limit 범위에서만 유
;P1?; (+#^)		효함. 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2; 명령어로 설정된 속도로 이동함.
	comment	리턴값은 증분값이 아니고 증감된 현재의 변위명령값임.
		이 명령어는 십진수를 long integer로, long integer를 십진수로 바꾸는 32bit 나누기 연산을 여러번 수행
		하므로, 명령어 처리에 약 150microsec의 시간이 더 걸리기 때문에 주의를 요함.
	operation	모터 2에 대하여 위치 명령값의 증분을 설정하고 이동을 명령함.
	operation	1st integer : motor2의 변위명령값의 증분 [-2147483648 ~ 2147483647]
	return value	P211111;
;P2 1; (+#^)	Teturii varue	1st long integer : motor2의 변위명령값
		position mode 로 설정된 경우에 한하며, position2_minus_limit에서 position2_plus_limit 범위에서만 유
;P2?; (+#^)		효함. 변위명령값의 위치로 이동하되 ;SSdddd1,dddd2; 명령어로 설정된 속도로 이동함.
	comment	리턴값은 증분값이 아니고 증감된 현재의 변위명령값임.
		이 명령어는 십진수를 long integer로, long integer를 십진수로 바꾸는 32bit 나누기 연산을 여러번 수행
		하므로, 명령어 처리에 약 150microsec의 시간이 더 걸리기 때문에 주의를 요함.



명령어	구분	설명
	operation	선택된 보드의 모터 출력을 ON 시킴.
		1st word : board address or A55A
	return value	PEHLHL;
		보드의 주소값이 일치하지 않으면 명령이 무시됨. 주소값이 A55A인 경우에는 2012년 2월 8일 S/W 버젼
•DCM M • (1400)		부터 사용가능하며 주소값이 일치하지 않더라도 모터 출력을 ON 시킴.
;PEHLHL; (+#^%)		fault flag가 set 되어 있으면 명령이 무시됨. (단 위치 리밋을 벗어난 fault와 통신 fault에 의해서는 PE
;PEA55A; (+#^)		명령이 무시되지 않음)
, PEASSA, (T#*)		PE? 에 대한 리턴값은, 모터 출력이 ON되어 있을 때는 board address이며, OFF 되어 있을 때는 0000 임.
;PE?; (+ #^ %)	comment	72V 이상의 고전압 H/W 버젼에서는 PE 명령어로 서보를 enable 시킨후에 SM 명령어로 작동모드를 선택해
, FE! , (T# 70)		야, 0.55초후에 전류측정 offset을 정상적으로 설정하게 되며, 이후에 PA나 SV같은 구동명령을 사용 수 있
		음.
		precharge 릴레이가 있는 72V 이상의 고전압 H/W 버젼에서는 2012년 6월 30일 S/W 버젼부터 precharge 릴
		레이를 먼저 ON 한 다음 100ms 후에 PWM출력이 enable 됨.
		PEA55A; 명령은 2012년 2월 8일 S/W 버젼부터 사용가능함.
	operation	선택된 보드의 fault flag를 clear 시킴.
	operation	1st word : board address or A55A
	return value	PRHLHL;
		보드의 주소값이 일치하지 않으면 명령이 무시됨. 주소값이 A55A인 경우에는 2012년 2월 8일 S/W 버젼
		부터 사용가능하며 주소값이 일치하지 않더라도 모터 출력을 ON 시킴.
;PRHLHL; (+#^%)		모터 작동의 중지를 일으킨 fault flag와 관련 timer/counter를 clear 시킴. IRMCK201을 사용할 때는
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		IRMCK201 내부의 fault flag도 clear 시켜서 다시 모터를 기동시킬 수 있게 하며, 홈동작중이거나 홈동작
;PRA55A; (+#^)		중 에러가 발생한 경우에 홈동작을 중지하고 새로운 홈동작을 개시할 수 있는 상태로 설정함.
, ,	comment	fault flag중에서 motorx_stall_flg, motorx_stall_K_flg 는 PPR; 명령어에 의하여 clear 되며,
		currentx_over_flg, currentx_over_K_flg는 과전류상태가 벗어나야만 clear되며,
		위치 리밋을 벗어난 fault는 리밋 안쪽으로 들어와야만 clear되며,
		통신 fault는 ECR; 명령에 의하여 clear됨.
		보드의 주소값이 일치하지 않으면 명령이 무시됨.
NOUG		PRA55A; 명령은 2012년 2월 8일 S/W 버젼부터 사용가능함. 159
CUBE -		133

명령어	구분	설명
	operation	선택된 보드의 모터 출력을 OFF 시킴.
	operation	1st word : board address or A55A
	return value	PDHLHL;
		보드의 주소값이 일치하지 않으면 명령이 무시됨. 주소값이 A55A인 경우에는 2012년 2월 8일 S/W 버젼
		부터 사용가능하며 주소값이 일치하지 않더라도 모터 출력을 ON 시킴.
;PDHLHL; (+#^%)		2011년 11월 20일 S/W 버젼부터는 모터가 disable 될때 작동모드도 0으로 강제설정되게 하였음.
, OHE IL, (1# 70)		PD? 에 대한 리턴값은, 모터 출력이 OFF되어 있을 때는 board address이며, ON 되어 있을 때는 0000 임.
;PDA55A; (+#^)		precharge 릴레이가 있는 72V 이상의 고전압 H/W 버젼에서는 2012년 6월 30일 S/W 버젼부터 PWM출력을 먼
, 1 υπουπ, (1π)		저 disable 한 다음 200ms 후에 precharge 릴레이를 끔.
;PD?; (+#^)	comment	PDA55A; 명령은 2012년 2월 8일 S/W 버젼부터 사용가능함.
		만약 에너지를 저장할수 있는 제어대상을 구동하던 중에, 제어대상이 큰 에너지를 저장하고 있는 상태에서
		PD 명령을 수행하면, 저장된 에너지가 기전력을 발생시켜서 모터 공급전압의 강한 상승을 유발하는 경우에
		모터제어기가 손상될수 있으므로, PD 명령의 사용에 주의를 기하여야함. 이 경우에는 PD 명령을 사용하
		지 않더라도 구동중의 fault 상황에 의하여 모터출력이 off 될수 있는데, 충분히 큰 용량의 brake저항을
		사용하여 에너지를 열로 환원할 수 있게 구성하여야함.



명령어	구분	설명
		현재위치로부터 dddd1점을 거쳐 dddd2점까지 dddd3의 속도로 이동시킴.
	operation	dddd1 : 첫번째 경유지 point number [0 - 166]
	operation	dddd2 : 최종 도착지 point number [1 - 167]
		dddd3 : point 당 움직임시간 [1 - 10000] (ms)
		PMdddd1,dddd2,dddd3;
		dddd1 : 경유중 point number [0 - 166]
	return value	dddd2 : 최종 도착지 point number [1 - 167]
;PMdddd1,dddd2,d		dddd3 : point 당 움직임시간 [1 - 10000] (ms)
ddd3; (+#^%)		ddd2 > ddd1 이어야함
		(point간 움직임시간 * POINT간 거리)이 32768*65535를 넘으면 비정상적 작동 유발.
;PM?; (+#^%)		SM 명령에 의한 작동모드가 위치제어모드인 01/02/03 중에 하나이어야하고, Se <mark>HLHL</mark> ; 명령으로 해당모터의
		motorx_start_to_move_trajectory_flg가 세트 가능토록 설정된 모터 (전원 투입시 기본값은 세트 가능토록
		하는 것임) 만 명령이 유효함.
	comment	PM 명령의 수행전에 SP 명령어로 이동좌표점이 설정되어 있거나, EROBOO,0500; 명령어로 이전에 EEPROM에
		저장된 이동좌표점을 불러내어야 함.
		2012년 12월 03일 S/W 버젼부터 좌표점의 개수가 192개에서 168개로 줄었음.
		RJM_VER5/6 에서는 의미없는 명령임.
		자석가동자의 경우 사용하면 안되면 명령임.
	operation	PM 에 의한 동작을 멈춤.
;PME; (+#^%)	return value	PME;
		PM 의 명령이 일단 실행되면 최종 도착지 point까지 이동을 하는데 도중에 멈추려면 ;PME; 를 실행하여야
// III_/ (III / / / / / / / / / / / / / / / /	comment	함.
		RJM_VER5/6 에서는 의미없는 명령임.
		자석가동자의 경우 사용하면 안되면 명령임.



명령어	구분	설명
		모터제어기의 작동모드에 따라서 10msec/30msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터
		가 움직이도록 함. PP인경우는 10msec, PQ인 경우는 30msec임.
		1st long integer : motor1의 변위명령값 [position1_minus_limit ~ position1_plus_limit]
	operation	or 속도명령값 [5000000-32000/60000 ~ 5000000+32000/60000]
	operation	or 전류명령값 [5000000-1023/4095 ~ 5000000+1023/4095]
		2nd long integer : motor2의 변위명령값 [position2_minus_limit ~ position2_plus_limit]
		or 속도명령값 [5000000-32000/60000 ~ 5000000+32000/60000]
		or 전류명령값 [5000000-1023/4095 ~ 5000000+1023/4095]
;PP11111,11112;	return value	PP 1, 2; or PQ 1, 2;
(+#^%)		SeHLHL; 명령으로 해당모터의 motorx_start_to_move_to_position_flg가 세트 가능토록 설정된 모터 (전원
		투입시 기본값은 세트 가능토록 하는 것임) 만 명령이 유효하며, 10msec/30msec 이전에 다음 명령이 입력
;PQ 1, 2;		되어야 연속으로 모터가 움직임.
(+#^%)		10msec/30msec 명령갱신 주기는 St 명령으로 변경이 가능함.
		position mode 로 설정된 모터는 변위명령값으로 설정되며, motor1의 경우는 position1_minus_limit에서
	comment	position1_plus_limit 범위에서만 유효하고, motor2의 경우는 position2_minus_limit에서
	Commert	position2_plus_limit 범위에서만 유효함. 이 경우는 변위명령값의 위치로 이동하되 명령갱신주기 만큼
		의 시간 후에 도달하도록 변위명령값이 1ms 단위로 쪼개져서 증분됨. 변위명령값은 5000000 이 기본값
		으로서 원점위치를 의미하며 항상 양수이어야함.
		speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을
		5000000-32000/60000 ~ 5000000+32000/60000 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범
		위에서 설정됨.



명령어	구분	설명
		current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 5000000-1023/4095 ~
		5000000+1023/4095 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨.
		이 명령어는 십진수를 long integer로, long integer를 십진수로 바꾸는 32bit 나누기 연산을 여러번 수행
	comment	하므로, 명령어 처리에 약 300microsec(return을 안받으면 150micros)의 시간이 더 걸리기 때문에, PP/PQ
		대신 Pp/Pq 명령어를 사용하는 것이 더 효과적임.
		IIIIx는 최대 7자리의 10진수임.
		RJM_VER5/6 에서는 의미없는 명령임.



명령어	구분	설명
;PpHBMBLBHBMBLBC S; (+#^%) ;PqHBMBLBHBMBLBC S; (+#^%)	operation	모터제어기의 작동모드에 따라서 10msec/30msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터가 움직이도록 함. Pp인경우는 10msec, Pq인 경우는 30msec임. 1st 24-bit long word: motor1의 목표변위값 [position1_minus_limit ~ position1_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] 2nd 24-bit long word: motor2의 목표변위값 [position2_minus_limit ~ position2_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] Pp/Pq 와; 를 빼고는 binary 값임.
	return value	2006년 10월 7일 이전 S/W 버젼 : PpHBMBLBHBMBLBCS; PqHBMBLBHBMBLBCS; HBMBLB 는 binary 형태의 24bit word를 가리키며 상위,중간,하위바이트를 가리킴. CS 는 앞선 6byte의 합의 inverse를 binary 한 바이트로 표현한 것임. return 값은 현재의 위치값임. checksum이 일치되지 않으면 수신값을 무시하고 return 값의 마지막은 :임. 이 경우 내부에서 수신에러 카운트 Pp_receive_error_cnt를 하나 증가시킴. 2006년 10월 7일 S/W 버젼부터 : PpLBHBLBHBLBHBLBHBLBHBLBHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능) PqLBHBLBHBLBHBLBCS; 또는 PqHBLBHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능) LB 는 binary 형태의 8bit 바이트 한개를 가리킴. HBLB 는 binary 형태의 16bit word의 상위바이트 한개, 하위바이트 한개를 가리킴. CS 는 앞선 8byte 의 합의 inverse를 binary 한 바이트로 표현한 것임.



명령어	구분	설명
		p 명령의 리턴값 변수주소를 설정하는 명령으로 SrHLHL,HLHL;을 사용하였을 때는, p 명령에 대한 return
		값으로 첫번째 바이트는 LIMIT_SWITCH_IN, 두번째 바이트는 high byte of fault_status, 세번째 바이트는
		low byte of fault_status, 네번째 바이트는 low byte of fault1_status, 다섯번째 바이트는 high byte of
		*PpPq_return_value_ADDRESS1, 여섯번째 바이트는 low byte of *PpPq_return_value_ADDRESS1, 일곱번째 바
		이트는 high byte of *PpPq_return_value_ADDRESS2, 여덟번째 바이트는 low byte of
		*PpPq_return_value_ADDRESS2 임.
		2007년 9월 19일 S/W 버젼부터 p 명령의 리턴값 변수주소를 설정하는 명령으로 SrHLHL, HLHL, HLHL, HLHL; 을
		사용하였을 때는, p 명령에 대한 return 값으로 1st/2nd/3rd/4th word에 의하여 지정된 4개 주소내의 값이됨.
		6.
		checksum이 일치되지 않으면 수신값을 무시하고 return 값의 마지막은 :임. 이 경우 내부에서 수신에러
		카운트 Pp_receive_error_cnt를 하나 증가시킴.
	aamman t	전송데이터 형식을 빼고는 PP/PQ 명령과 같음.
	comment	SeHLHL; 명령으로 해당모터의 motorx_start_to_move_to_position_flg가 세트 가능토록 설정된 모터 (전원
		투입시 기본값은 세트 가능토록 하는 것임) 만 명령이 유효하며, 10msec/30msec 이전에 다음 명령이 입력
		되어야 연속으로 모터가 움직임.
		10msec/30msec 명령갱신 주기는 St 명령으로 변경이 가능함.
		position mode 로 설정된 모터는 변위명령값으로 설정되며, 변위명령값의 위치로 이동하되 명령갱신주기
		만큼의 시간 후에 도달하도록 변위명령값이 1ms 단위로 쪼개져서 증분됨. 변위명령값은 4C4B40 이 기본
		값으로서 원점위치를 의미하며 항상 양수이어야함.
		speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을 004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범위
		에서 설정됨.
		current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 004C4B40-3FF/FFF ~
		004C4B40+3FF/FFF 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨.
		RJM_VER5/6 에서는 의미없는 명령임.



명령어	구분	설명
	operation	PP, PQ, Pp, Pq 에 의한 동작을 멈춤.
;PPE; (+#^%)	return value	PPE; or PQE;
		PP, PQ, Pp, Pq 의 명령이 일단 실행되면 다음의 PP, PQ, Pp, Pq 를 계속 기다리는 상태가 된다. 이 상
;PQE; (+#^%)	comment	태를 빠져나오려면 ;PPE; 를 실행하여야함.
		RJM_VER5/6 에서는 의미없는 명령임.
	operation	모터가 stall된 채로 정지된 상태를 해지함.
;PPR; (+#^%)	return value	PPR; or PQR;
		motor1_stall_flg, motor2_stall_flg, motor1_stall_K_flg, motor2_stall_K_flg를 clear 시킴.
;PQR; (+#^%)	comment	다시 모터를 기동시킬 수 있게 됨.
		RJM_VER5/6 에서는 의미없는 명령임.



명령어	구분	설명
;pHBMBLBHBMBLBCS	operation	모터제어기의 작동모드에 따라서 10msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터가 움직이도록 함. 1st 24-bit long word : motor1의 목표변위값 [position1_minus_limit ~ position1_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] 2nd 24-bit long word : motor2의 목표변위값 [position2_minus_limit ~ position2_plus_limit] or 속도명령값 [004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60] or 전류명령값 [004C4B40-3FF/FFF ~ 004C4B40+3FF/FFF] p 와 ; 를 빼고는 binary 값임. HBMBLB 는 binary 형태의 24bit word를 가리키며 상위,중간,하위바이트를 가리킴.
; (RS485의 경우) (+#^%) pHBMBLBHBMBLBCS (CAN의 경우) (+#^%)		CS 는 앞선 6byte의 합의 inverse를 binary 한 바이트로 표현한 것임. RS485의 경우: pLBHBLBHBLBHBLBHBLBCS; 또는 pHBLBHBLBHBLBHBLBHBLBCS;(2007년 9월 19일 S/W 버젼부터 사용가능) CAN의 경우: LBHBLBHBLBHBLB 또는 HBLBHBLBHBLBHBLB(2007년 9월 19일 S/W 버젼부터 사용가능) LB 는 binary 형태의 8bit 바이트 한개를 가리킴. HBLB 는 binary 형태의 16bit word의 상위바이트 한개, 하위바이트 한개를 가리킴. CS 는 앞선 8byte 의 합의 inverse를 binary 한 바이트로 표현한 것임. p 명령의 리턴값 변수주소를 설정하는 명령으로 SrHLHL,HLHL;을 사용하였을 때는, p 명령에 대한 return 값으로 첫번째 바이트는 LIMIT_SWITCH_IN, 두번째 바이트는 high byte of fault_status, 세번째 바이트는 low byte of fault_status, 네번째 바이트는 low byte of fault_status, 대선째 바이트는 high byte of *PpPq_return_value_ADDRESS1, 여섯번째 바이트는 low byte of *PpPq_return_value_ADDRESS1, 일곱번째 바이트는 high byte of *PpPq_return_value_ADDRESS2 임. 2007년 9월 19일 S/W 버젼부터 p 명령의 리턴값 변수주소를 설정하는 명령으로 SrHLHL,HLHL,HLHL,HLHL;을 사용하였을 때는, p 명령에 대한 return 값으로 1st/2nd/3rd/4th word에 의하여 지정된 4개 주소내의 값이됨.



명령어	구분	설명
		checksum이 일치되지 않으면 수신값을 무시하고 return 값의 마지막은 :임. 이 경우 내부에서 수신에러
		카운트 Pp_receive_error_cnt를 하나 증가시킴.
		전송데이터 형식을 빼고는 PP/PQ 명령과 같음.
		SeHLHL; 명령으로 해당모터의 motorx_start_to_move_to_position_flg가 세트 가능토록 설정된 모터 (전원
		투입시 기본값은 세트 가능토록 하는 것임) 만 명령이 유효하며, 10msec 이전에 다음 명령이 입력되어야
		연속으로 모터가 움직임.
		10msec 명령갱신 주기는 St 명령으로 변경이 가능함.
	comment	position mode 로 설정된 모터는 변위명령값으로 설정되며, 변위명령값의 위치로 이동하되 명령갱신주기
	COMMETT	만큼의 시간 후에 도달하도록 변위명령값이 1ms 단위로 쪼개져서 증분됨. 변위명령값은 4C4B40 이 기본
		값으로서 원점위치를 의미하며 항상 양수이어야함.
		speed mode 로 설정된 모터는 변위명령값이 아니고 속도명령값으로 설정되며, 설정값을
		004C4B40-7D00/EA60 ~ 004C4B40+7D00/EA60 범위에서 주면 속도명령값은 -32000/60000 ~ 32000/60000 범위
		에서 설정됨.
		current mode 로 설정된 모터는 설정전류값을 설정하게 되며, 설정값을 004C4B40-3FF/FFF ~
		004C4B40+3FF/FFF 범위에서 주면 전류명령값은 -1023/4095 ~ 1023/4095 범위에서 설정됨.
		RJM_VER5/6 에서는 의미없는 명령임.



명령어	구분	설명
	operation	모터 1,2를 모두 현재의 위치에서 감속을 실시한 후에 정지함.
	return value	PsA55A;
;PsA55A; (+#^%)		PM/PP/PQ/Pp/Pq/p에 의한 위치제어 도중이었다면 이를 중지하고 현재의 위치로부터 감속하여 정지함.
		속도제어 중이었다면 감속을 실시한 후 정지하며 전류제어중이었다면 전류값으로 0으로 강제설정함.
;PtA55A; (+#^)	comment	단 감속률은 PsA55A;의 경우 Ss 명령어나 Sa 명령어로 설정된 감속도에 따르며, PtA55A;의 경우는 So 명령
		어로 설정된 감속도에 따름.
		2012년 1월 19일 S/W 버젼부터 사용가능한 명령임.
	operation	모터 1을 현재의 위치에서 감속을 실시한 후에 정지함.
	return value	PsA5A5;
;PsA5A5; (+#^%)		PM/PP/PQ/Pp/Pq/p에 의한 위치제어 도중이었다면 이를 중지하고 현재의 위치로부터 감속하여 정지함.
		속도제어 중이었다면 감속을 실시한 후 정지하며 전류제어중이었다면 전류값으로 0으로 강제설정함.
;PtA5A5; (+#^)	comment	단 감속률은 PsA55A;의 경우 Ss 명령어나 Sa 명령어로 설정된 감속도에 따르며, PtA55A;의 경우는 So 명령
		어로 설정된 감속도에 따름.
		2012년 1월 19일 S/W 버젼부터 사용가능한 명령임.
	operation	모터 2를 현재의 위치에서 감속을 실시한 후에 정지함.
	return value	Ps5A5A;
;Ps5A5A; (+#^%)		PM/PP/PQ/Pp/Pq/p에 의한 위치제어 도중이었다면 이를 중지하고 현재의 위치로부터 감속하여 정지함.
		속도제어 중이었다면 감속을 실시한 후 정지하며 전류제어중이었다면 전류값으로 0으로 강제설정함.
;Pt5A5A; (+#^)	comment	단 감속률은 PsA55A;의 경우 Ss 명령어나 Sa 명령어로 설정된 감속도에 따르며, PtA55A;의 경우는 So 명령
		어로 설정된 감속도에 따름.
		2012년 1월 19일 S/W 버젼부터 사용가능한 명령임.



명령어	구분	설명
;PeA55A; (+#^)	operation	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2 모두를 02번 위치 제어모드로 바꾸고, 현재의 위치에서 감속후 정지함.
	return value	PeA55A;
	comment	감속률은 Ss 명령어로 설정된 감속도에 따름.
		digital Hall 센서만을 사용하는 BLDC 모터제어에서만 유효함.
	operation	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을 02번 위치제어모
		드로 바꾸고, 현재의 위치에서 감속후 정지함.
;PeA5A5; (+#^)	return value	PeA5A5;
	comment	감속률은 Ss 명령어로 설정된 감속도에 따름.
		digital Hall 센서만을 사용하는 BLDC 모터제어에서만 유효함.
	operation	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를 02번 위치제어모
;Pe5A5A; (+#^)		드로 바꾸고, 현재의 위치에서 감속후 정지함.
	return value	Pe5A5A;
	comment	감속률은 Ss 명령어로 설정된 감속도에 따름.
		digital Hall 센서만을 사용하는 BLDC 모터제어에서만 유효함.



명령어	구분	설명
	operation	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2
		모두를 감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
;PvA55A; (+#^)	return value	PvA55A;
	comment	감속률은 Sa, Ss 명령어로 설정된 감속도에 따름.
		digital Hall 센서만을 사용하는 WAFER 구동용 BLDC 모터제어에서만 유효함.
	operation	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을
	орогаттоп	감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
;PvA5A5; (+#^)	return value	PvA5A5;
	comment	감속률은 Sa, Ss 명령어로 설정된 감속도에 따름.
	COMMOTT	digital Hall 센서만을 사용하는 WAFER 구동용 BLDC 모터제어에서만 유효함.
	operation	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를
		감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
;Pv5A5A; (+#^)	return value	Pv5A5A;
	comment	감속률은 Sa, Ss 명령어로 설정된 감속도에 따름.
	COMMOTT	digital Hall 센서만을 사용하는 WAFER 구동용 BLDC 모터제어에서만 유효함.
		digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우에, 속도제어 모드에서 속도명령값을 설정하되,
	operation	리턴값으로 현재 위치값을 host로 전송함.
		1st long integer : motor1 속도명령값 [-32000/60000 ~ 32000/60000] (0 rpm)
;Pv 1, 2;		2nd long integer : motor2 속도명령값 [-32000/60000 ~ 32000/60000] (0 rpm)
(+#^)	return value	PvIIII1,IIII2;
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\		1st long integer : motor1의 현재 변위값 (4체배 pulse, 32bit)
;Pv?; (+#^)		2nd long integer : motor2의 현재 변위값 (4체배 pulse, 32bit)
	comment	속도제어 모드에서만 사용이 가능함.
		리턴값은 엔코더 원점기준의 변위를 의미하므로 0 근방의 값인데 대하여, QP/PA 등의 명령이 5,000,000 근
		방의 값을 사용하는 것과 다르므로 주의를 해야 함.
		digital Hall sensor만을 사용하는 BLDC 모터의 경우는 해당사항 없음.



명령어	구분	설명
		digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우에, 전류제어 모드에서 전류명령값을 설정하되,
	operation	리턴값으로 현재 속도값을 host로 전송함.
;Pciiii1,iiii2;	operation	1st integer : motor1 전류명령값 [-4095 ~ 4095] (bit)
(+#^)		2nd integer : motor2 전류명령값 [-4095 ~ 4095] (bit)
(π)		Pciiii1,iiii2;
;Pc?; (+#^)	return value	1st integer: 모터1의 현재 속도값 (rpm)
,10:, (III)		2nd integer : 모터2의 현재 속도값 (rpm)
	comment	전류제어 모드에서만 사용이 가능함.
	Commont	digital Hall sensor만을 사용하는 BLDC 모터의 경우는 해당사항 없음.
		속도제어 모드에서 속도명령값을 설정하되 설정값의 checksum을 사용함.
		1st integer : motor1 속도명령값 [-60000 ~ 60000] (0 rpm)
;PVIIII1,III12,d	operation	2nd integer : motor2 속도명령값 [-60000 ~ 60000] (0 rpm)
ddd3; (+#^)		3rd integer : 'IIII1,IIII2' 의 각 글자code값을 더한 값 (예를들어 PV0,0,140; 에서 140은 '0'의 code
		값 48과 ','의 code 값 44, '0'의 code 값 48을 모두 더한 값임)
;PVr 1, 2,	return value	PVIII1, III12;
dddd3; (+#^)		PVr 명령의 경우는 속도오차가 큰 경우에 속도명령값을 재설정하고 적분오차를 0으로 초기화하는 작동을
		하지 않음.
;PV?; (+#^)	comment	PVr 명령의 경우는 2011년 9월 24일 S/W 버젼부터 사용가능한 명령임.
		F2811 <mark>/F28334</mark> 에서는 long integer를 입력할수있으며 속도명령값의 설정가능 범위가 [-60000 ~ 60000]
		(rpm)임.



명령어	구분	설명
		2개의 리밋센서와 1개의 홈센서를 사용한 홈 동작을 개시함. (GH는 0.01도 단위의 offset 변위이며 Gh는 4
		체배 엔코더 펄스단위의 offset 변위임)
		1st byte : 모터번호 [0이면 선택없음, 1이면 모터1, 2이면 모터2, 3이면 모터1/2]
		2nd byte : home 동작 개시 방향
	operation	[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2]
		iiii1 : 모터1이 홈센서를 찾은 후 offset 변위 [-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
		iiii2 : 모터2가 홈센서를 찾은 후 offset 변위 [-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
		1 : 모터1이 홈센서를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
;GHHL,HL,iiii1,i		IIII2 : 모터2가 홈센서를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
iii2; (+#^%)	return value	GHHL,HL,IIII1,IIII2; or GhHL,HL,IIII1,IIII2;
1112, (111 10)		GH 와 Gh 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.
;GhHL,HL, 1,		SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 home 동작 개시 방향의 리밋센서를 찾은 후
1112; (+#^)		(+방향이면 LSxA 입력이 0, -방향이면 LSxC 입력이 0이 될 때 리밋을 찾은 것으로 봄) 반대방향으로 움직
11127 (1117)		여서 홈센서를 찾은 후 (LSxB 입력이 0이 될 때 홈을 찾은 것으로 봄) 홈센서가 꺼질 때까지 움직여서 목
;GH?; (+#^)		표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함. 만약 리
, G. 11 , C. 11		밋센서보다 홈센서를 먼저 발견하게 되면 바로 반대방향으로 홈센서가 꺼질때까지 움직여서 목표변위, 엔
;Gh?; (+#^)		코더값등을 초기값으로 재설정하고 offset 변위만큼 이동함. 이때 홈센서가 꺼지기 전에 home 동작 개
, .	comment	시 반대 방향의 리밋센서가 ON 되어 있으면 홈센서와 리밋센서가 공용인 것으로 판단하여 이동방향을 바꾼
	oonorre	후에 홈센서가 꺼질때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 offset 변위만큼 이동
		함.
		SI 명령어로 설정된 위치 범위내에서만 리밋센서와 홈센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수
		행을 중지하고 정지함. SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.
		PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료
		되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.
		홈동작 진행/성공여부는 Q1; 명령으로 알수있으며 PRHLHL; 명령으로 홈동작을 중지시킬 수 있음.
		절대각센서를 사용하는 경우는 사용이 불가한 명령임.



명령어	구분	설명
		2개의 리밋센서 만을 사용한 홈 동작을 개시함. (GI는 0.01도 단위의 offset 변위이며 Gi는 4체배 엔코더
		펄스단위의 offset 변위임)
		1st byte : 모터번호 [0이면 선택없음, 1이면 모터1, 2이면 모터2, 3이면 모터1/2]
		2nd byte : home 동작 개시 방향
	operation	[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2]
		iiii1 : 모터1이 리밋센서를 찾은후 offset 변위 [-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
·CIUI UI ;;;;1 ;		iiii2 : 모터2가 리밋센서를 찾은후 offset 변위 [-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
;GIHL,HL,iiii1,i iii2; (+#^)		IIII1 : 모터1이 리밋센서를 찾은후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
1112, (##)		IIII2 : 모터2가 리밋센서를 찾은후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
;GiHL,HL, 1,	return value	GIHL,HL,IIII1,IIII2; or GiHL,HL,IIII1,IIII2;
1112; (+#^)		GI 와 Gi 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.
1112, (1π)		SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 home 동작 개시 방향의 리밋센서를 찾은 후
;GI?; (+#^)		(+방향이면 LSxA 입력이 0, -방향이면 LSxB 입력이 0이 될 때 리밋을 찾은 것으로 봄) 리밋센서가 꺼질 때
, (111)		까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이
;Gi?; (+#^)		동함.
, (111)	comment	SI 명령어로 설정된 위치 범위내에서만 리밋센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수행을 중지
	Comment	하고 정지함.
		SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.
		PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료
		되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.
		홈동작 진행/성공여부는 Q1; 명령으로 알수있으며 PRHLHL; 명령으로 홈동작을 중지시킬 수 있음.
		절대각센서를 사용하는 경우는 사용이 불가한 명령임.



명령어	구분	설명
		한개의 홈센서 만을 사용한 홈 동작을 개시함. (GJ는 0.01도 단위의 offset 변위이며 Gj는 4체배 엔코더
		펄스단위의 offset 변위임)
		1st byte : 모터번호 [0이면 선택없음, 1이면 모터1, 2이면 모터2, 3이면 모터1/2]
		2nd byte : home 동작 개시 방향
		[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2]
	operation	iiii1 : 모터1이 홈센서의 경계를 찾은 후 offset 변위
		[-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
;GJHL,HL,iiii1,i		iiii2 : 모터2가 홈센서의 경계를 찾은 후 offset 변위
iii2; (+#^)		[-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
1112, (1π)		1 : 모터1이 홈센서의 경계를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
;GjHL,HL, 1,		IIII2 : 모터2가 홈센서의 경계를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
1112; (+#^)	return value	GJHL,HL, 1, 2; or GjHL,HL, 1, 12;
1112, (1117)		GJ 와 Gj 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.
;GJ?; (+#^)		SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 홈센서를 찾은 후 (LSxB 입력이 0이 될 때
, 40., (11.)		홈을 찾은 것으로 봄) 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초
;Gj?; (+#^)		기값은 5000000임) offset 변위만큼 이동함. 만약 처음부터 홈센서가 ON이면 홈센서가 꺼질 때까지 움
, , , , , , , , , , , , , , , , , , , ,		직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함.
	comment	SI 명령어로 설정된 위치 범위내에서만 홈센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수행을 중지하
		고 정지함.
		SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.
		PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료
		되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.
		홈동작 진행/성공여부는 Q1; 명령으로 알수있으며 PRHLHL; 명령으로 홈동작을 중지시킬 수 있음.
		절대각센서를 사용하는 경우는 사용이 불가한 명령임.



명령어	구분	설명
		한개의 홈센서 만을 사용한 홈 동작을 개시하되 홈센서가 ON 되어 있는 상태에서만 개시함. (GK는 0.01도
		단위의 offset 변위이며 Gk는 4체배 엔코더 펄스단위의 offset 변위임)
		1st byte : 모터번호 [0이면 선택없음, 1이면 모터1, 2이면 모터2, 3이면 모터1/2]
		2nd byte : home 동작 개시 방향
		[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2]
	operation	iiii1 : 모터1이 홈센서의 경계를 찾은 후 offset 변위
;GKHL,HL,iiii1,i		[-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
iii2; (+#^)		iiii2 : 모터2가 홈센서의 경계를 찾은 후 offset 변위
		[-32000 ~ +32000, 감속기 출력기준의 0.01도 단위 각도]
;GkHL,HL, 1,		1 : 모터1이 홈센서의 경계를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
1112; (+#^)		IIII2 : 모터2가 홈센서의 경계를 찾은 후 offset 변위 [모터에 부착된 엔코더의 4체배 펄스기준]
	return value	GKHL, HL, 11111, 11112; or GkHL, HL, 11111, 11112;
;GK?; (+#^)		GK 와 Gk 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.
		만약 처음부터 홈센서가 OFF(LSxB 입력이 1 일 때)이면 명령을 무시하며, 홈센서가 처음부터 ON이면 SS 명
;Gk?; (+#^)		령어로 정의된 속도로 home 동작 개시방향으로 이동하여 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더
		값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함.
	comment	SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.
		PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료
		되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.
		홈동작 진행/성공여부는 Q1; 명령으로 알수있으며 PRHLHL; 명령으로 홈동작을 중지시킬 수 있음.
		절대각센서를 사용하는 경우는 사용이 불가한 명령임.



명령어	구분	설명
		리밋센서와 홈센서의 ON/OFF를 검출하는 경계전압을 설정함.
	operation	1st word : 0FF를 검출하는 경계전압 [100 - 280] (225 0.01V)
;GTdddd1,dddd2;		2nd wprd : 0N를 검출하는 경계전압 [20 - 200] (75 0.01V)
(+#^)	return value	GTdddd1,dddd2;
		1st word 보다 높은 전압일 경우 리밋센서나 홈센서는 OFF인 것으로 처리함.
;GT?; (+#^)	comment	2nd word 보다 낮은 전압일 경우 리밋센서나 홈센서는 ON인 것으로 처리함.
	Commert	비서로봇의 경우는 기본값이 2.63V와 1.31V로 설정되어 있음.
		⇔ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		RJM_VER7의 Hall 센서입력의 ON/OFF를 검출하는 기준전압과 히스테리시스 진폭을 설정함.
	operation	1st word : ON/OFF를 검출하는 기준전압 [25 - 400] (25/200 0.01V)
		2nd wprd : 히스테리시스 진폭 [5 - 200] (5/10 0.01V)
	return value	Gtdddd1,dddd2;
;Gtdddd1,dddd2;		1st word + 2nd word 보다 높은 전압일 경우 Hall 센서는 OFF인 것으로 처리함.
(+#^)		1st word - 2nd word 보다 낮은 전압일 경우 Hall 센서는 ON인 것으로 처리함.
		open collector 방식의 Hall sensor일때는 신호전압의 상승 속도가 매우 느리기 때문에, ON/OFF를 검출하
;Gt?; (+#^)	comment	는 기준전압과 히스테리시스 진폭을 작은 값으로 설정하여야함.
		F28334 H/W 버젼에서는 Gt 명령어로 설정된 값을 저장할 때 LF2406A/F2811에 비하여 16배 더 큰 값으로 저
		장되나 겉으로 들어나는 차이는 없음.
		나노모션의 analog Hall sensor를 사용하는 경우에는 225,10의 값을 사용할 것.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



홈동작 - 2개의 리밋센서와 1개의 홈센서를 쓰는 경우

♣ 홈동작을 하려면

1) SMHLHL; 명령어로 01/02/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없고 PWM사용, DC 모터전용) 상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, DC/BLDC) 상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어, BLDC 모터전용)

- 2) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 3) SSdddd1.dddd2; 명령어로 홈이동속도를 정의함.
- 4) GHHL,HL,iiii1,iiii2; 또는 GhHL,HL,iiii1,iiii2; 명령어로 홈동작을 개시함.
- 5) 홈동작의 상태는 Q1; 명령어로 확인할 수 있음.

의한 작동모드값이 01/02/03에서만 적용가능함.

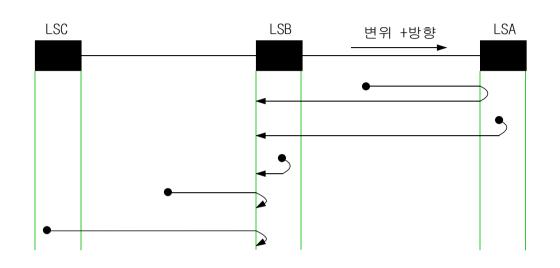
6) 홈동작도중에 중지하고자하면 PRHLHL; 명령어로서 모든 에러 플래그를 clear함.

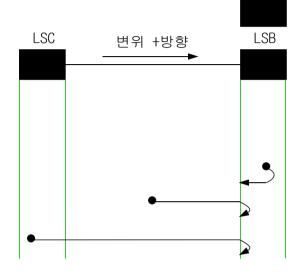
GH 와 Gh 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.

SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 home 동작 개시 방향의 리밋센서를 찾은 후 (+방향이면 LSxA 입력이 0, -방향이면 LSxC 입력이 0이 될 때 리밋을 찾은 것으로 봄) 반대방향으로 움직여서 홈센서를 찾은 후 (LSxB 입력이 0이 될 때 홈을 찾은 것으로 봄) 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함. 만약 리밋센서보다 홈센서를 먼저 발견하게 되면 바로 반대방향으로 홈센서가 꺼질때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 offset 변위만큼 이동함. 이때 홈센서가 꺼지기 전에 home 동작 개시 반대 방향의 리밋센서가 이지되어 있으면 홈센서와 리밋센서가 공용인 것으로 판단하여 이동방향을 바꾼 후에 홈센서가 꺼질때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 offset 변위만큼 이동함. SI 명령어로 설정된 위치 범위내에서만 리밋센서와 홈센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수행을 중지하고 정지함.

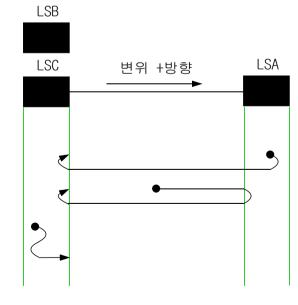
PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.

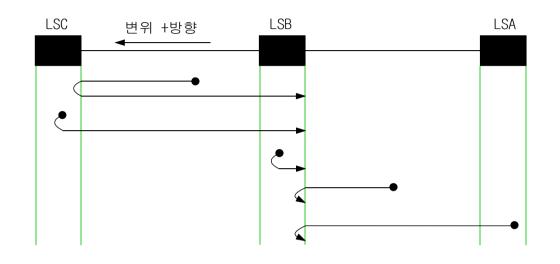


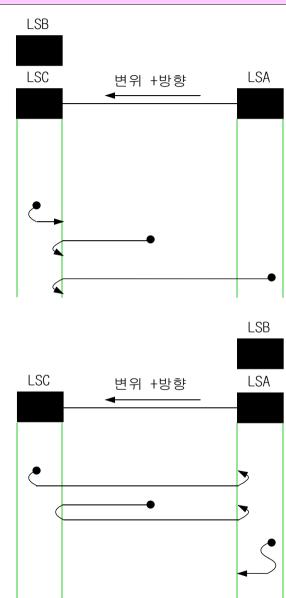




LSA









홈동작 - 2개의 리밋센서만을 쓰는 경우

♣ 홈동작을 하려면

1) SMHLHL; 명령어로 01/02/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없고 PWM사용, DC 모터전용) 상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, DC/BLDC) 상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어, BLDC 모터전용)

- 2) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 3) SSdddd1.dddd2; 명령어로 홈이동속도를 정의함.
- 4) GIHL,HL,iiii1,iiii2; 또는 GiHL,HL,iiii1,iiii2; 명령어로 홈동작을 개시함.
- 5) 홈동작의 상태는 Q1; 명령어로 확인할 수 있음.
- 6) 홈동작도중에 중지하고자하면 PRHLHL; 명령어로서 모든 에러 플래그를 clear함.
- GI 와 Gi 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.
- SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 home 동작 개시 방향의 리밋센서를 찾은 후 (+방향이면 LSxA 입력이 0, -방향이면 LSxB 입력이 0이 될 때 리밋을 찾은 것으로 봄) 리밋센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함.
- SI 명령어로 설정된 위치 범위내에서만 리밋센서와 홈센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수행을 중지하고 정지함.
- SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.

PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.



홈동작 - 1개의 홈센서만을 쓰는 경우로서 홈센서의 방향을 알때

♣ 홈동작을 하려면

1) SMHLHL; 명령어로 01/02/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없고 PWM사용, DC 모터전용) 상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, DC/BLDC) 상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어, BLDC 모터전용)

- 2) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 3) SSdddd1.dddd2; 명령어로 홈이동속도를 정의함.
- 4) GJHL,HL,iiii1,iiii2; 또는 GjHL,HL,iiii1,iiii2; 명령어로 홈동작을 개시함.
- 5) 홈동작의 상태는 Q1; 명령어로 확인할 수 있음.
- 6) 홈동작도중에 중지하고자하면 PRHLHL; 명령어로서 모든 에러 플래그를 clear함.

GJ 와 Gi 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.

SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 홈센서를 찾은 후 (LSxB 입력이 0이 될 때 홈을 찾은 것으로 봄) 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함. 만약 처음부터 홈센서가 ON이면 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼 이동함. SI 명령어로 설정된 위치 범위내에서만 홈센서를 찾으며 만약 범위를 벗어나면 즉시 명령의 수행을 중지하고 정지함.

SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.

PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.



홈동작 - 1개의 홈센서만을 쓰는 경우로서 홈센서 방향을 모를때

♣ 홈동작을 하려면

1) SMHLHL; 명령어로 01/02/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없고 PWM사용, DC 모터전용) 상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, DC/BLDC) 상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어, BLDC 모터전용)

- 2) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 3) SSdddd1.dddd2; 명령어로 홈이동속도를 정의함.
- 4) GKHL,HL,iiii1,iiii2; 또는 GkHL,HL,iiii1,iiii2; 명령어로 홈동작을 개시함.
- 5) 홈동작의 상태는 Q1; 명령어로 확인할 수 있음.
- 6) 홈동작도중에 중지하고자하면 PRHLHL; 명령어로서 모든 에러 플래그를 clear함.

GK 와 Gk 명령에서 offset변위의 기본단위는 각각 0.01도와 4체배 엔코더 펄스임.

만약 처음부터 홈센서가 OFF(LSxB 입력이 1 일 때)이면 명령을 무시하며, 홈센서가 처음부터 ON이면 SS 명령어로 정의된 속도로 home 동작 개시방향으로 이동하여 홈센서가 꺼질 때까지 움직여서 목표변위, 엔코더값등을 초기값으로 재설정하고 (초기값은 5000000임) offset 변위만큼이동함.

SM 명령에 의한 작동모드값이 01/02/03에서만 적용가능함.

PM/PP/PQ/Pp/Pq 명령에 의한 연속궤적동작을 안하고 있을 때로서, 홈동작이 한번도 안되었을 때 또는 완료되었을 때 또는 오류가 발생했을 때 : 즉 홈동작을 실행중이지 않을 때에 적용가능함.



제어모드의 설정 및 제어작동의 실행 (홈동작 - 1개의 홈센서만을 쓰는 Direct Drive 모터의 경우)

♣ 이 경우는 처음부터 홈센서가 ON 이어야 하며, 이 경우 전원을 켠 위치가 홈위치가 될 뿐아니라, 모터의 영구자석 위치도 판별하는 기준이됨.



명령어	구분	설명
	operation	모터1의 속도모드 조그 작동을 수행함.
;J; (+# ^%)	return value	J;
	commont	J; 명령어를 계속해서 보내어 오는 동안은 조그작동을 수행하며, J;의 입력 여부는 10ms 마다 체크됨.
	comment	모터1이 05/06 속도모드로 정의되어 있을때 만 가능함.
	operation	모터2의 속도모드 조그 작동을 수행함.
;K; (+#^%)	return value	K;
,N, (T# 70)	comment	K; 명령어를 계속해서 보내어 오는 동안은 조그작동을 수행하며, K;의 입력 여부는 10ms 마다 체크됨.
	COMMETT	모터2가 05/06 속도모드로 정의되어 있을때 만 가능함.
		속도모드의 조그동작에서 조그방향, 조그이동속도 및 조그동작 지속시간을 정의함.
		1st byte : jog_move_direction 변수에 조그동작시에 방향을 저장함.
	operation	[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2] (00)
	орегаттоп	dddd1 : jog_move_speed1 변수에 조그동작 속도를 설정함 [1 - MOTOR1_SPEED_MAX] (500 rpm)
		dddd2 : jog_move_speed2 변수에 조그동작 속도를 설정함 [1 - MOTOR2_SPEED_MAX] (500 rpm)
;SJHL,dddd1,dddd		dddd3 : jog_move_lasting_time 변수에 조그동작 지속시간 설정치를 저장함. [2 - 50] (10 10ms)
2,dddd3; (+#^%)	return value	SJHL,dddd1,dddd2,dddd3;
Σ, αααασ, (1π 70)		SJ 명령을 보낸후에 J;/K; 명령어를 보내면 가속을 거쳐서 jog_move_speed1/jog_move_speed2 의 속도로 이
;SJ?; (+#^%)		동을 함. 조그동작을 계속하려면 jog_move_lasting_time×10ms 이내에 J;/K; 명령어를 보내기를 반복해
,ου:, (۱π /0)		야 함. J;/K; 를 마지막으로 보낸 후에 jog_move_lasting_time×10ms 만큼을 기다린 다음, 감속을 하면
	comment	서 정지하고 조그작동을 멈춤.
	Commert	가속/감속율은 Sa 명령어로 설정된 값에 따름.
		조그동작 도중에 SJ 명령어를 보내면 조그속도를 바꿀 수 있음.
		2012년 6월 13일 이전 S/W 버젼에서는 +방향 조그이동속도를 크게 설정하지 못하는 문제가 있음.



명령어	구분	설명
	operation	모터1의 위치모드 조그 작동을 수행함.
;j; (+#^%)	return value	j;
	comment	j; 명령어를 계속해서 보내어 오는 동안은 조그작동을 수행하며, j;의 입력 여부는 10ms 마다 체크됨.
	COMMETT	모터1이 01/03 위치모드로 정의되어 있을때 만 가능함.
	operation	모터2의 위치모드 조그 작동을 수행함.
;k; (+# ^%)	return value	k;
, N, (1# 10)	comment	k; 명령어를 계속해서 보내어 오는 동안은 조그작동을 수행하며, k;의 입력 여부는 10ms 마다 체크됨.
	COMMETT	모터2가 01/03 위치모드로 정의되어 있을때 만 가능함.
		위치모드의 조그동작에서 조그방향, 조그이동속도 및 조그동작 지속거리를 정의함.
	operation	1st byte : jog_move_direction 변수에 조그동작시에 방향을 저장함.
		[0이면 음의 방향, 1이면 양의 방향이며 하위 4bit는 모터1, 상위 4bit는 모터2] (00)
		dddd1 : jog_move_speed1 변수에 조그동작 속도를 설정함 [1 - MOTOR1_SPEED_MAX] (500 rpm)
		dddd2 : jog_move_speed2 변수에 조그동작 속도를 설정함 [1 - MOTOR2_SPEED_MAX] (500 rpm)
;SjHL,dddd1,dddd		dddd3 : jog_move_step_size 변수에 조그동작 지속거리 설정치를 저장함. [2 - 1000] (200 pulse)
2,dddd3; (+#^%)	return value	SjHL,dddd1,dddd2,dddd3;
Σ, ααααο ; (1π 70)		Sj 명령을 보낸후에 j;/k; 명령어를 보내면 가속을 거쳐서 jog_move_speed1/jog_move_speed2 의 속도로 이
;Sj?; (+#^%)		동을 함. 조그동작을 계속하려면 jog_move_step_size 거리를 이동하기 전에 j;/k; 명령어를 보내기를
, Oj : , (m /0 /		반복해야 함. j;/k; 를 마지막으로 보낸 후에 jog_move_step_size 거리를 이동하는 동안 감속을 하면서
	comment	정지하고 조그작동을 멈춤.
	COMMICTIC	가속/감속시간은 Ss 명령어로 설정된 값에 따름.
		조그동작 도중에 Sj 명령어를 보내면 조그속도를 바꿀 수 있음.
		조그동작 도중에 비정상적인 가감속을 피하려면 가속/감속시간이 짧거나 jog_move_step_size가 크거나
		j;/k; 명령어의 입력주기가 짧아야 함.



JOG 작동

- ♣ JOG 동작을 하려면 (속도모드의 JOG)
 - 1) SMHLHL; 명령어로 05/06의 속도제어 모드를 선택함.

상위/하위바이트가 05이면 모터2/모터1이 속도제어모드(inner loop 없고 PWM사용(DC 2406), inner loop 전류제어(BLDC 2406, DC/BLDC 2811))

상위/하위바이트가 06이면 모터2/모터1이 속도제어모드(inner loop 없고 PWM사용(DC 2811), inner loop 속도제어(BLDC))

- 2) SJHL.dddd1.dddd2.dddd3; 명령어로 조그방향. 조그이동속도1/2 및 조그동작 지속시간을 설정함.
- 3) Sadddd1,dddd2,dddd3,dddd4; 가감속도를 설정함. (단위는 RJM_VER7인 경우에 rpm/ms이고 나머지는 rpm/100ms 임)
- 4) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 5) 모터1을 JOG 하려면 J; 를 입력하고, 모터2를 JOG 하려면 K; 를 입력함.

이때 조그동작 지속시간이내에 J; 나 K; 가 입력되면 입력 순간으로부터 조그동작 지속시간만큼 조그동작 정지시간이 연장됨.

- ♣ JOG 동작을 하려면 (위치모드의 JOG)
 - 1) SMHLHL; 명령어로 01/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없음, PWM사용, DC 모터전용)

상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, DC/BLDC, 현재 RJM_VER4(2811)/RJM_VER8 DC, BLDC 만가능)

상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어, BLDC 모터전용)

- 2) SJHL,dddd1,dddd2,dddd3; 명령어로 조그방향, 조그이동속도1/2 및 조그동작 지속시간을 설정함.
- 3) SjHL,dddd1,dddd2,dddd3; 명령어로 조그방향, 조그이동속도1/2 및 조그동작 지속거리를 설정함.

SJ 명령어를 이용하여 조그동작 지속시간을 설정한 다음에 Si 명령어를 사용하여 나머지 파라메터를 설정해야함.

- 4) Ssdddd1.dddd2; 가감속 시간을 설정함. (단위는 ms 임)
- 5) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 6) 모터1을 JOG 하려면 j; 를 입력하고, 모터2를 JOG 하려면 k; 를 입력함.

이때 조그동작 지속거리 만큼을 이동하기 전에 j; 나 k; 가 입력되면 현 위치로부터 조그동작 지속거리만큼 정지위치가 연장됨.



명령어	구분	설명
		모터제어기의 주소를 설정함. 1st word 2012년 10월 18일 이전 S/W 버젼인 경우 또는 SX 명령어 첫번째파라메터의 bit0 = 1 인 경우
		: 주소값 [0000 - 00FF] (807F) 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 첫번째파라메터의 bit0 = 0 인 경우
	operation	: 주소값 [0000 - 00FF] (807F) - SX 명령어 두번째파라메터의 bit15/14 = 11 일때 : 주소값 [0000 - 07FF] (007F) - SX 명령어 두번째파라메터의 bit15/14 = 10 일때
		: 주소값 [0000 - FFFF] (007F) - SX 명령어 두번째파라메터의 bit15/14 = 01 일때 : 주소값 [0000 - 00FF] (807F) - SX 명령어 두번째파라메터의 bit15/14 = 00 일때
;SAHLHL; (+#^%)	return value	SAHLHL;
;SA?; (+#^%)	comment	♣ 2012년 10월 18일 이전 S/W 버젼인 경우 또는 SX 명령어 첫번째파라메터의 bit0 = 1 인 경우와 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 두번째파라메터의 bit15/14 = 11 또는 00 일때는, 주소값의 상위 바이트는 반드시 하위바이트의 inverse이어야만 baord 주소의 설정이 유효함. ♣ 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 첫번째파라메터의 bit0 = 0 일때 SX 명령어 두번째파라메터의 bit15/14 값에 따라서 CAN 통신의 주소사용방법이 달라짐. bit15/14 값에 따라서 CAN 통신의 주소사용방법이 달라짐. bit15/14 = 11 일때 11-bit 주소모드로서 주소로 0-255만 사용 = 10 일때 11-bit 주소모드로서 주소로 0-255만 사용 (CAN2.0B), 2012년 10월 16일부터 적용 = 01 일때 29-bit 주소모드로서 주소로 0-255만 사용 (CAN2.0B), 2012년 10월 16일부터 적용 = 00 일때 11-bit 주소모드로서 주소로 0-255만 사용 ♣ 주소값으로서 254(0xFE) 및 255(0xFF)는 bradcasting용으로 사용하는 주소이므로 126(0x8E), 127(0x8F), 254(0xFE) 및 255(0xFF)는 제어기의 주소로 사용하면 안됨. ♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		위치, 속도, 전류제어 등의 작동모드를 설정함.
	operation	1st word : 작동모드 [0000 - FFFF] (0000)
	return value	SMHLHL;
		육 상위/하위바이트가
		01 : 모터2/모터1이 위치제어모드(inner loop 없고 PWM사용, DC 모터전용)
		02 : 모터2/모터1이 위치제어모드(inner loop 전류제어, DC(F2811/F28334)/BLDC/STEP(엔코더형))
		03 : 모터2/모터1이 위치제어모드(inner loop 속도제어)
		04 : 모터2/모터1이 속도제어모드(위치제어형식으로 속도제어, DC/BLDC/STEP(엔코더형))
		05 : 모터2/모터1이 속도제어모드(inner loop 전류제어, DC(F2811/F28334)/BLDC(F2811/F28334))
		06 : 모터2/모터1이 속도제어모드(inner loop 없고 PWM사용(DC), inner loop 속도제어(BLDC))
		07 : 모터2/모터1이 전류제어모드(DC/BLDC)
;SMHLHL; (+#^%)		08 : 모터2/모터1이 host electrical angle 제어모드(BLDC/STEP 모터전용)
CHILLIE (TI N)		09 : 모터2/모터1이 host electrical angle 제어모드(STEP모터의 속도모드전용)
;SM?; (+#^%)	comment	OA: 모터2/모터1이 속도제어모드(03번 위치제어에서 사용하는 속도제어와 같음)
, cm. , (m , o ,		OE : 모터2/모터1이 사인파 위치제어(DC/BLDC/STEP, 01 또는 03번 또는 08번 위치제어모드 사용,
		F2811/F28334에서 가능)
		OF: 모터2/모터1이 사인파 위치제어(DC/BLDC/STEP, 02번 위치제어모드 사용, F2811/F28334에서 가능)
		00 : 하위바이트가 00이면 모터2 off, 하위바이트가 00이면 모터1 off.
		♣ 04번 모드는 02번 위치제어모드를 사용하여 속도를 제어함, 단 LF2406 H/W 버젼에서는 01번 위치제어
		모드를 사용함.
		♣ 모드간 전환을 하는 경우에 이전 모드의 작동을 정지하고 새 모드의 작동을 개시함. 이때 위치/속도
		/전류/electrical-angle 등의 설정치는 0 또는 기본값으로 설정됨.
		♣ 380V 제어기의 경우에는 PE 명령어로 서보를 enable 시킨후에 SM 명령어로 작동모드를 선택해야, 0.55
		초후에 전류측정 offset을 정상적으로 설정하게 되며, 이후에 PA나 SV같은 구동명령을 사용 수 있음.
		♣ 작동모드로 03/0F가 선택되는 순간 IRMCK201의 가감속율을 9323 RPM/100ms 이상으로, RJM_VER7에서는
		1000*(10/8) RPM/ms이상으로 설정함. 이는 위치제어의 성능을 높이는데 필요한 사항임.



명령어	구분	설명
		♣ SM 01/02/03/04/08/0E/0F 모드로 설정할때 가감속시간 MOTORx_ACC_DEC_PERIOD_SET을 설정하도록 하였음.
		♣ 2007년 10월 31일 현재 01,02(RJM_VER4(F2811/F28334)/RJM_VER8 DC,
		BLDC),03,04,05(RJM_VER4(F2811/F28334)/RJM_VER8 DC, BLDC(F2811/F28334)),06,07,08,0F 만이 구현되어 있음.
		♣ 2010년 7월3일 S/W 버젼부터 09번모드는 스테핑모터의 속도모드를 설정하는데 사용함. 스테핑모터의
		08번모드의 작동은 예전과 동일하며, BLDC모터의 경우 09번 작동모드는 08번 작동모드로 처리됨.
		♣ 03번 작동모드는 2012년 10월 3일 S/W 버젼부터 위치제어루프 내에 속도제어루프가 있는 위치제어모드로
		사용할수 있음. 속도제어루프 안에는 전류제어루프가 있음. Digital Hall 센서만을 사용하는 BLDC에는
		사용할수 없음.
		♣ 0A번 작동모드는 속도제어모드로서 2012년 10월 3일 S/W 버젼부터 사용할수 있음. 03번 제어모드에서
		속도제어루프의 게인을 설정하는데 쓸수 있음. Digital Hall 센서만을 사용하는 BLDC에는 사용할수 없음.
		2011년 11월 20일 S/W 버젼부터는 PE 명령어를 수행한 이후에만 SM 명령에 의한 작동모드 설정이 가능하게
		하였음.
		2011년 11월 27일 S/W 버젼부터 2012년 12월 20일 이전의 S/W 버젼에서는 위치모드 제어를 사용하던 도중에
		완전히 정지하지 않고 SM0000; 명령을 사용하여 감속을 하였다가, 다시 위치모드로 재설정하면 잠깐 재감속
		하는 위치명령이 발생하는 문제점이 있으므로 주의를 요함.
		리턴값이 SM0000: 인 경우에 작동모드 설정을 제대로 못했다는 뜻이며, 이러한 상황은
		(1) PWM 형식의 전류센서를 사용하는 경우 PE 명령을 수행할때 상전류/총전류값이 0에 해당하는 offset 값을
		측정하게 되는데, 이 전류 offset 값이 비정상적인 값을 갖을 때 발생하며,
		(2) 그외 AD8205/AD8206을 사용하는 경우는 전원 투입후 초기화 과정에서 상전류/총전류값이 0에 해당하는
		offset 값을 측정하게 되는데, 이 전류 offset 값이 비정상적인 값을 갖을 때도 발생함.
		(3) 사용할 수 없는 작동모드를 선택하였을 때는 해당 채널만 00값으로 리턴됨.
		(4) 펄스주기를 사용하여 속도계산을 하는 모드를 선택하였을때 (SX 명령어 두번째 파라메터의 BIT8이 0일
		때) PWM 주파수가 60KHz 보다 클때 (Z8 명령어로 설정하는 값은 하위 바이트가 06보다 클때)



명령어	구분	설명
		특히 총전류 측정회로에서 AD8205/AD8206의 전류 offset을 설정하는 저항이 0 0hm이 아닐 경우에는 0 0hm으
		로 교체해주어야 함. (모터제어기의 구버젼 하드웨어인 경우에 한함)
		♪ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



제어기 명령어 LIST

SM 명령어로 설정 가능한 작동 모드 (2012년 12월 26일 S/W 버젼부터 적용)

	00	01 위치제어 (PWM 직접 설정)	02 위치제어 (내부전류 제어루프)	03 위치제어 (내부속도 제어루프)	04 위치제어방 식의 속도 제어	05 속도제어 (내부전류 제어루프)	06 속도제어 (PWM 직접 설정)	07 전류제 어	08 (위치)속도 제어 (개루프)	09 속도제어	0A 속도제어 (내부전류 제어루프)	00 사인파 위 치제어	0E 사인파 위 치제어	0F 사인파 위 치제어
STEP (엔코더 없음)	○ (PWM 출력 강제설정 가 능)	_	_	_	_	_	_	_	○ (위치/속도명 령 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	_	○ (08번 모드 로 실행)	○ (08번 모드 로 실행)	○ (08번 모드 로 실행)
STEP (엔코더 있고 lead angle 제어 사 용)	○ (PWM 출력 강제설정 가 능)	-	O (lead angle 제어 실시)	_	○ (02번 모드 로 실행)	O (lead angle 제어 실시)	-	0	○ (위치/속도명 령 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	-	○ (08번 모드 로 실행)	○ (08번 모드 로 실행)	○ (02번 모드 로 실행)
DC	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	_	_	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (IRMCK201 을 사용)	○ (PWM 출력 강제설정 <mark>불</mark> 가능)		0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (03번 모드 로 실행)	○ (02번 모드 로 실행)
AC induction motor	○ (PWM 출력 강제설정 가 능)	_	_	0	○ (03번 모드 로 실행)	_	_	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	_	○ (03번 모드 로 실행)	○ (03번 모드 로 실행)	○ (03번 모드 로 실행)
BLDC (digital Hall IC 만 사 용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (digital Hall IC와 Encoder를 사 용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (3상 analog Hall 센서를 사용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (2상 resolver를 사 용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (PWM 방식 의 절대각센서 를 사용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)
BLDC (통신방식 의 절대각센서 를 사용)	○ (PWM 출력 강제설정 가 능)	0	0	0	○ (02번 모드 로 실행)	0	0	0	○ (속도명령만 수신)	○ (속도명령만 수신, 08번 모 드로 실행)	0	○ (03번 모드 로 실행)	○ (01번 모드 로 실행)	○ (02번 모드 로 실행)



명령어	구분	설명					
		모터의 기본파라메터를 쉽게 설정하기 위하여 미리 준비되어있는 motor type을 선정함.					
	operation 하위 byte : 모터1의 tpye [00 - FF]						
		상위 byte : 모터2의 tpye [00 - FF]					
	return value	STHLHL;					
		♣ EDA55A;가 후속으로 실행되어야 그 효과가 반영되는데, 이때 MOTOR1_PULSE_PER_REV,MOTOR1_GEAR_					
		RATIO, MOTOR2_PULSE_PER_REV, MOTOR2_GEAR_RATIO 가 설정되고 제어 파라메터도 설정됨.					
		♪ motor 상위바이트/하위 바이트가					
		00 : 4체배된 1024-PPR MAXON RE-max 29/24V/22W DC + 100:1 감속기 + max 8790 rpm,					
		01 : 4체배된 2048-PPR FAULHARBER 3257CR 24V/83W DC + 100:1 감속기 + max 5900 rpm,					
		02 : 4체배된 4096-PPR MAXON EC 45 flat 24V/50W(△) BLDC(8 pole pair) + 100:1 + max 6700 rpm,					
		03 : 4체배된 4000-PPR MAXON EC-powermax 30 24V/100W(△) BLDC(2 pole pair) + 100:1 + max 17800 rpm,					
;STHLHL; (+#^%)		04 : 4체배된 2000—PPR MAXON EC 45 24V/250W(△) BLDC(1 pole pair) + 100:1 + max 9090 rpm,					
		05 : 4체배된 2000-PPR FAULHARBER 4490 24V/200W(△) BLDC(1 pole pair) + 100:1 + max 9550 rpm,					
;ST?; (+#^%)		06 : analog Hall sensor FAULHARBER 3564 24V/100W(△) BLDC(1 pole pair) + 100:1 + max 11300 rpm,					
	comment	07 : analog Hall sensor FAULHARBER 2057 24V/60W(△) BLDC(1 pole pair) + 100:1 + max 26500 rpm,					
		08 : analog Hall sensor FAULHARBER 4490 24V/200W(△) BLDC(1 pole pair) + 100:1 + max 9550 rpm,					
		09 : 4체배된 2000-PPR MAXON EC 90 flat 24V/90W BLDC(12 pole pair) + 100:1 + max 3190 rpm,					
		OA : 4체배된 4096-PPR KOLLMORGEN RBEH 01212 24V/181W BLDC(4 pole pair) + 100:1 + max 3820 rpm,					
		OB : 4체배된 10000-PPR KOMOTEK DANQ-01EF1N2 24V/100W BLDC(4 pole pair) + 100:1 + max 5000(?) rpm,					
		또는 KOMOTEK 380V/3KW					
		OC : 4체배된 8192-PPR KOLLMORGEN RBEH 01211/13 24V/152W/203W BLDC(4 pole pair) + 100:1					
		+ max 5600/3240 rpm,					
		OD : 4체배된 8192-PPR KOLLMORGEN RBEH 02110 24V/241W BLDC(6 pole pair) + 100:1 + max 1470 rpm,					
		OE : 4체배된 8192-PPR KOLLMORGEN RBEH 00511 24V/52W BLDC(3 pole pair) + 100:1 + max 12800 rpm,					
		OF : 4체배된 4000-PPR MAXON RE35/40 48V/90W/150W DC + 100:1 감속기 + max 3810/7580 rpm,					



명령어	구분	설명
		10 : digital Hall-IC type AMT용 BLDC(7 pole pair) + 100:1 감속기 + max 6700 rpm,
		11 : 4체배된 2048-PPR MAXON EC 45 flat 24V/50W(△) BLDC(8 pole pair) + 100:1 + max 6700 rpm,
		12 : 4체배된 60000-PPR NANO MOTION linear 모터(2 pole pair) + 100:1 감속기 + max 3000 rpm,
		13 : 4체배된 4000-PPR 퍼스텍 BLDC(3 pole pair) + 100:1 감속기 + max 12800 rpm,
		14 : 4체배된 72000-PPR MAXON EC 90 flat 24V/90W BLDC(12 pole pair) + 100:1 + max 3190 rpm,
		15 : 4체배된 8000-PPR MITSUBISHI 380V/3KW BLDC(4 pole pair) + 100:1 + max 5000(?) rpm,
		16 : 4체배된 3276800-PPR YASKAWA DD-motor(10 pole pair) + 1:1 + max 120 rpm,
		그외 모터 type이 FFFF/FFE 값이 아닌 한 2048-PPR FAULHARBER 3257CR 24V/83W + 100:1 + max 5900
		rpm으로 기본값을 설정함.
	comment	♣ 모터 type이 FFFE 보다 작으면, PWM_PERIOD 값이 10≠S으로 설정되며 (PWM 출력주파수는 100KHz), 상기된
		필스수/감속비 설정에 따라 제어파라메터가 재설정됨.
		♣ 모터 type이 FFFF 이면, SEA55A 명령으로 1회전당 모터펄스값과 SG 명령으로 PWM_PERIOD/감속비의 변경이
		가능하고, 이어서 EDA55A; 명령을 수행하여 제어파라메터를 재설정하고, SA 명령어로 주소를 재설정하고,
		STFFFE; 명령을 수행한 다음, EsA55A; 명령으로 파라메터를 저장할 수 있음.
		🕏 모터 type이 FFFF 인 상태에서 1회전당 모터펄스값과 PWM_PERIOD/감속비를 변경치 않으면 직전에 EEPROM
		에 저장되었던 파라메터가 사용되나, 이때 실수로 SG 명령이 수행되면 PWM_PERIOD 값이 변경될 수 있음.
		따라서 파라메터의 저장 직전에 반드시 motor의 type을 FFFE 로 변경해야 함.
		⇔ 모터 type이 FFFE 이면, 1회전당 모터펄스값과 PWM_PERIOD/감속비의 변경이 안됨.
		♣ 모터 type 06/07/08/09은 F2811/F28334에서만 설정이 가능함.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



ST 명령을 이용한 모터 기본파라메터의 설정과 저장

- ♣ 모터의 기본파라메터를 설정하고 저장하려면 (상기 테이블에 존재하는 모터타입이 있는 경우)
 - 1) STHLHL; 명령을 실행하여 모터의 type를 지정함. (하위 8bit:motor1, 상위 8bit:motor2)
 - 2) EDA55A; 명령을 실행하며 파라메터를 초기화함. (이때 모터의 주소가 7F로 바뀜)
 - 3) SAHLHL; 명령을 실행하여 모터에 주소를 설정함.
 - 4) SB115200; 명령어를 사용하여 제어기 RS232C 통신의 기본 Baudrate를 바꾸어줌.
 - 5) ESOAOO.0100; 명령 또는 EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
 - 6) 전원을 껏다가 켬.
 - 7) ST?;, SA?;, SB?; 등을 사용하여 바뀐 파라메터가 반영되었는지를 확인함.
- ♣ 모터의 기본파라메터를 설정하고 저장하려면 (상기 테이블에 존재하는 모터타입이 없는 경우)
 - 1) STFFFF; 명령을 실행하고 나서 SE 명령과 SG 명령을 사용하여 모터 1회전당 엔코더펄스수, PWM_PERIOD, 감속비, 최대속도 등을 설정함.
 - 2) EDA55A; 명령을 실행하며 파라메터를 초기화함. (이때 모터의 주소가 7F로 바뀜)
 - 3) SAHLHL; 명령을 실행하여 모터에 주소를 설정함.
 - 4) SB115200; 명령어를 사용하여 제어기 RS232C 통신의 기본 Baudrate를 바꾸어줌.
 - 5) STFFFE; 명령을 실행하여 모터타입을 FFFE로 설정함.
 - 6) ESOAOO,0100; 명령 또는 EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
 - 7) 전원을 껏다가 켬.
 - 8) ST?;, SA?;, SB?; 등을 사용하여 바뀐 파라메터가 반영되었는지를 확인함.



모터 기본파라메터의 설정과 저장

♣ 모터의 기본파라메터를 설정하고 저장하려면 (상기 테이블에 존재하는 모터타입을 사용하면서 엔코더 펄스수나 무부하 최고속도, 기어비 등을 바꾸고자 하면)

- 1) STHLHL; 명령을 실행하여 모터의 type를 지정함. (하위 8bit:motor1, 상위 8bit:motor2)
- 2) EDA55A; 명령을 실행하며 파라메터를 초기화함. (모든 파라메터를 factory default 값으로 설정함)
- 3) SEA55A,HLHL,HLHL,HLHL,HLHL; 명령을 실행하며 4체배된 엔코더 펄스수와 pole pair 수를 HEX ASKII로 입력함. (DC/STEP 모터의 경우에 pole pair 수는 1이며, BLDC의 경우는 모터의 카타로그를 통하여 실제값을 입력하여야함.)
- 4) 만약 무부하 최고 속도나 감속비를 바꾸고자하면 SGA55A,dddd1,dddd2,dddd3,dddd4; 명령어로 바꾸고자 하는 값을 입력함.
- 5) EDA55A; 명령을 실행하여 파라메터를 다시 초기화함. (이 과정에서 바뀐 파라메터에 따른 초기화를 실시함)
- 6) SAHLHL; 명령어를 사용하여 제어기의 주소를 설정함.
- 7) SB115200; 명령어를 사용하여 제어기 RS232C 통신의 기본 Baudrate를 바꾸어줌.
- 8) ESOAOO,0100; 명령 또는 EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
- 9) 전원을 껏다가 켬.
- 10) ST?;, SE?;, SG?;, SA?;, SB?; 등을 사용하여 바뀐 파라메터가 반영되었는지를 확인함.

※ CUBE-BLxxxx-SIxR 모델의 경우는 엔코더 펄스수가 바뀌거나, pole pair가 바뀌는 경우는 RJMC 프로그램을 사용하여 파라메터의 갱신을 추가로 해주어야 함.



모터 기본파라메터의 설정과 저장

- 🕹 analog Hall sensor 타입 BLDC 모터의 홀센서 파라메터를 설정하고 저장하려면
 - 1) STHLHL; 명령을 실행하여 모터의 type를 지정함. (하위 8bit:motor1, 상위 8bit:motor2)
 - 2) EDA55A; 명령을 실행하며 파라메터를 초기화함. (모든 파라메터를 factory default 값으로 설정함)
 - 3) SEA55A,4000,4000,HLHL,HLHL; 명령을 실행하며 pole pair 수를 HEX ASKII로 입력함. (BLDC 모터의 카타로그를 통하여 실제값을 입력하여야함.)
 - 4) 만약 무부하 최고 속도나 감속비를 바꾸고자하면 SGA55A,dddd1,dddd2,dddd3,dddd4; 명령어로 바꾸고자 하는 값을 입력함.
 - 5) EDA55A; 명령을 실행하여 파라메터를 다시 초기화함. (이 과정에서 바뀐 파라메터에 따른 초기화를 실시함)
 - 6) SAHLHL; 명령어를 사용하여 제어기의 주소를 설정함.
 - 7) SB115200; 명령어를 사용하여 제어기 RS232C 통신의 기본 Baudrate를 바꾸어줌.
 - 8) ESOA00.0100; 명령 또는 EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
 - 9) 전원을 껏다가 켬.
 - 10) ST?;. SE?;, SG?;, SA?;, SB?; 등을 사용하여 바뀐 파라메터가 반영되었는지를 확인함.
 - 11) WJ1,10,20,08,12; (여기서 8,10은 8000 RPM을 정역방향 전류치를 같게 하는 튜닝을 실시하는 RPM으로 정한다는 뜻이며, 파라메터 설정 완료후에 12000 RPM을 정역 구동을 최종시험하는 RPM으로 정한다는 뜻임.)
 - 13) WJ?;의 리턴값이 WJ7,xx,xx,xx,xx,xx; 이면 성공적으로 파라메터 튜닝과 저장이 이루어 졌다는 뜻임.



세어기 원립어 LIST

ST 명령을 이용한 모터 기본파라메터의 설정

STHLHL;에 서 HL의 값	모터종류		엔코터 펄스수 (4체배된 값)	엔코더 안쓰는 경우	pole pair	Gear ratio	maximum speed [RPM]
00	DC	1024 pulse MAXON RE-max	1024	_	1	100	8790
01	DC	2048 pulse FAULHARBER	2048	_	1	100	5900
02	BLDC	4096 pulse MAXON EC 45 flat	4096	16384 (digital)	8	100	6700
03	BLDC	4000 pulse MAXON EC powermax	4000	16384 (digital)	2	100	17800
04	BLDC	2000 pulse MAXON EC 45	2000	16384 (digital)	1	100	9090
05	BLDC	2000 pulse FAULHARBER 4490 (△)	2000	16384 (digital)	1	100	9550
06	BLDC	analog Hall sensor FAULHARBER 3564 100W	_	16384 (analog)	1	100	11300
07	BLDC	analog Hall sensor FAULHARBER 2057 60W	_	16384 (analog)	1	100	26500
08	BLDC	analog Hall sensor FAULHARBER 4490 200W	_	16384 (analog)	1	100	9550
09	BLDC	2000 pulse MAXON EC 90 flat	2000	16384 (digital)	12	100	3190
0a	BLDC	4096 pulse KOLLMORGEN RBEH 01212 181W	4096	16384 (digital)	4	100	3820
0b	BLDC	10000 pulse KOMOTEK DANQ-01EF1N2 100W	10000	16384 (digital)	4	100	5000
0c	BLDC	8192 pulse KOLLMORGEN RBEH 01211/13 152/203W	8192	16384 (digital)	4	100	5600
0d	BLDC	8192 pulse KOLLMORGEN RBEH 02110 241W	8192	16384 (digital)	6	100	1470
0e	BLDC	8192 pulse KOLLMORGEN RBEH 00511 52W	8192	16384 (digital)	3	100	12800
Of	DC	4000 pulse MAXON RE35/40 48V 90/150W	4000	_	1	100	3810
10	BLDC	digital type AMT용	-	16384 (digital)	7	100	6700
11	BLDC	2048 pulse MAXON EC 45 flat 24V/50W(△)	2048	16384 (digital)	8	100	6700
12	LM	NANO MOTION linear 모터	60000 (15000 * 4)	16384 (analog)	2 (1 analog)	100	3000
13	BLDC	ENC type 퍼스텍	4000	16384 (digital)	3	100	12800
14	BLDC	2000 pulse MAXON EC 90 flat + linear scale	72000 (18000 * 4)	16384 (digital)	12	100	3190



세어기 명령어 UST

ST 명령을 이용한 모터 기본파라메터의 설정

STHLHL;에 서 HL의 값	모터종류		엔코터 펄스수 (4체배된 값)		pole pair	Gear ratio	maximum speed [RPM]
15	BLDC	8000 pulse MITSUBISHI 3KW (TMS28334/5 H/W 버젼 만 적용가능)	8000	16384 (digital)	4	100	5000
16	DD	3276800 pulse YASKAWA DD-motor	3276800	_	10	1	120
17	AC induction	8000 pulse 100W SPG induction motor	8000	_	2	100	1500
18 ~ FD	DC	default	2048	_	1	100	5900
any value	STEP	50 pole (200 step or 1.8도) 2상 스테핑모터	_	16384	50	1	3000
FE		사용자지정 파라메터인 경우	사용자지정	사용자지정	사용자 지정	사용자지정	사용자지정
FF		SE와 SG명령에 의한 파라메터 변경 가능 상태	SE 명령어로 설정	SE 명령어로 16384 값을 설정 해야함	SE 명 령어로 설정	SG 명령어 로 설정	SG 명령어로 설정



명령어	구분	설명
;SXHLHL; (+#^%) ;SX?; (+#^%)	operation	통신과 제어의 option을 선택하는 DIP switch의 값을 설정함. bit0 : DIP switch 1 : 통신모드의 설정 [CAN(0)/RS485(1)] bit1 : 직선보간 기능의 사용여부 [안함(0)/실시함(1)] (2406a에는 해당없음) bit2 : 모터권선의 온도보호 기능의 사용여부 [안함(0)/실시함(1)] (2406a에는 해당없음) bit3 : 모터의 입력/출력 power, 출력토크, 효율, 모터권선/하우징의 온도계산의 실시여부 [안함(0)/실시함(1)] (2406a에는 해당없음) bit4 : DIP switch 4 : 통신테스트 모드의 설정 bit5 : DIP switch 3 : 통신테스트 모드의 설정 bit5 : DIP switch 3 : 통신테스트 모드의 설정 bit4/5 : 외부의 20X4 text display에 선택된 4개의 변수값을 실시간으로 표시하는 기능의 사용여부 (2406a에는 해당없음) bit6 : DIP switch 2 : 통신모드의 설정 (2406a 에만 해당됨) bit7 : 홈 찾기를 위하여 사용하는 리밋센서를 선택함. (2406a에는 해당없음) bit8 : 모터2가 모터1을 추종하도록 제어하는 dummy모드의 사용여부 [안함(0)/실시함(1)] bit9 : 속도제어중 정지시에 적분항을 0으로 처리하는지의 여부 [안함(0)/실시함(1)] bit10 : positionx_limit(소프트웨어적인 가상 위치 리밋)의 적용하는지의 여부 [적용(0)/비적용(1)] bit11 : analog Hall sensor type의 RJM_VER7에서 Hall sensor parameter를 매 출발시마다 갱신을 하는지의 여부 [안함(0)/실시함(1)] (2406a에는 해당없음) bit12 : 속도모드에서 구동중에 위치리밋센서를 만나면 강제정지하도록 하는 기능의 사용여부 [안함(0)/실시함(1)] (2406a에는 해당없음) bit13 : LSxC입력단자의 전압에 따라서 속도명령 또는 전류명령을 설정 하는지의 여부 [안함(0)/실시함(1)] 엔코더가 있는 스텝모터의 경우는 위치명령도 엔코더에 의하여 설정되는지의 여부도 결정함. (2406a에는 해당없음) bit14 : sequence control mode에서 스텝동작 여부 [연속작동(0)/스텝작동(1)] (2406a에는 해당없음) bit15 : sequence control mode에서 스텝동작 여부 [연속작동(0)/스텝작동(1)] (2406a에는 해당없음)
CUBE	return value	SXHLHL; 또는 SXHLHLHLHL;(TMS320F2810/2811/2812/28334/28335 로서 2009년 8월 16일 S/W 버젼부터 적용됨) 됨) 2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHLL; 도 사용 가능함.

명령어	구분	설명
		♧ bit0/4/5/6의 경우는 DIP switch가 실제로 존재할 때 현재 DIP switch의 상태를 나타내며,
		bit1/2/3/7/8-15의 경우는 소프트웨어적으로 설정된 DIP switch의 상태를 나타냄. 따라서 DIP switch가
		실제로 존재하는 경우에는 bit0/4/5/6의 소프트웨어적 설정이 불가능함.
		♣ 2406a의 경우는 DIP switch 1/2의 ON(0)/OFF(1)에 따라서 통신모드가 선택되며, bitO/6(DIP switch
		1/2)가 1/1이면 RS485-only 통신 모드로 작동하며, 0/1이면 CAN-RS232C 통신 모드, 1/0이면 CAN-only 통
		신 모드, 0/0이면 RS232C-only 통신 모드로 작동함.
		F2811/F28334의 경우는 DIP switch 1의 ON(0)/OFF(1)에 따라서 통신모드가 선택되며, bitO(DIP switch 1)
		가 0이면 CAN 통신 모드로 작동하며, bit0(DIP switch 1)가 1이면 RS485 통신 모드로 작동함. RS232C
		통신은 항상 사용이 가능함.
		♣ F2811/F28334의 경우는 DIP switch 1이 0 일 때 CAN 통신 모드로 작동하며, LAN 이 장치된 제어기에서
		는 LAN 통신이 가능함. DIP switch 1이 1 일 때는 LAN 이 장치되었더라도 RS485 통신만 가능함.
		⇔ DIP switch 3/4의 ON(0)/OFF(1)에 따라서 통신테스트모드 설정되며, DIP switch 3/4가 1/0일때, 통신테
		스트모드가 설정됨. 이 경우는 받은 값을 2msec 후에 바로 되돌려주며, 수신에러가 발생하였을 때는 수
	comment	신에러 상태값(SCIxRXST_reg)을 되돌려줌.
		♣ 통신 모드의 경우는 SXHLHL; 로 값을 설정하고 ESOAOO,0100; 명령어로 EEPROM에 저장한 다음, 전원을
		재 투입하고 나면 바뀐 설정에 따른 동작을 실행하나, bit6의 경우는 SXHLHL; 로 값을 설정하는 순간부터
		<mark>동작에 반영됨</mark> . (이 설정은 F2811/F28334에서만 설정이 가능함)
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		♣ bit1이 1이면 직선보간 기능을 사용함. (모터작동모드가 02/05/07번 일때로서 2축제어기의 경우만 해당
		됨.) (이 설정은 F2811/F28334에서만 설정이 가능함)
		⇔ bit2가 1이면 모터권선의 온도보호 기능을 작동 시킴. (모터작동모드가 02/05/07번 일때만 사용가능)
		(이 설정은 F2811/F28334에서만 설정이 가능함)
		& bit3가 1이면 모터의 입력/출력 power, 출력토크, 효율, 모터권선/하우징의 온도계산을 실시함. (모터
		작동모드가 02/05/07번 일때만 사용가능) (이 설정은 F2811/F28334에서만 설정이 가능함)
		& bit4/5가 1/0이면 외부의 20X4 text display에 선택된 4개의 변수값을 실시간으로 표시하는 기능을 실
		시함. 기능설정에 사용하는 명령은 DZ/DF/DS/DT 임. (RS485_MODE, EtherNetIP_or_USB_MODE,
		COM_TEST_MODE가 아닐때만 사용가능) (이 설정은 F2811/F28334에서만 설정이 가능함)



명령어	구분	설명
		♣ bit7이 1이면 리밋센서로 LSxA(+리밋), LSxB(-리밋)를 사용하며, 0이면 리밋센서로 LSxA(+리밋),
		LSxC(-리밋)를 사용함. bit10이 0이어서 positionx_limit을 넘어설 때 강제 정지하는 기능에만 적용되
		며 홈동작에는 적용되지 않음. (이 설정은 F2811/F28334에서만 설정이 가능함)
		♣ bit8이 1이면 두번째 채널의 PWM은 첫번째 채널의 PWM 출력을 copy 하여 실행되며, 두번째 채널의 엔코
		더 입력에 의한 제어는 무시됨.
		♣ bit9이 1이면 속도제어중 속도명령이 0일때 적분제어항을 0으로 처리함으로써, 정지시에 적분제어항 때
		문에 원위치를 유지하려는 작동을 계속함으로써 모터에 과부하를 주는 일을 막을수 있음. 이렇게 하면
		속도명령이 0일때는 비례항 이외에는 반력을 만들지 않으나, 05번 속도제어 모드에서는 전류 offset에 따
		라서 모터가 한 방향으로 흐르는 문제가 발생할 수 있음. 제일메디컬의 경우는 무조건 0으로 처리함.
		♣ bit10이 0이면 위치모드에서 구동중에 위치리밋센서 또는 가상위치리밋을 만나면 강제정지하도록 함.
		& bit11이 1이면 analog Hall sensor type의 RJM_VER7 경우 Hall sensor parameter를 속도 0에서 출발할
		때 마다 매번 계산하여 반영하는데, 단 가속이 10과 같거나 작아야 하고, 06번 작동 모드여야 함. (이 설
		정은 F2811/F28334에서만 설정이 가능함).
		& bit12가 1이면 속도모드에서 구동중에 위치리밋센서 또는 가상위치리밋을 만나면 강제정지하도록 함.
		(이 설정은 F2811/F28334에서만 설정이 가능함).
		& bit13가 1이면 LSxC입력단자의 전압에 따라서 속도명령값(속도모드의 경우) 또는 전류명령값(전류모드
		의 경우)을 설정함. 속도 설정 범위는 SG 명령어로 설정되는 최대 속도 까지 임. 전류설정범위는 Sw
		명령어로 설정되는 전류제어리밋까지 임. 엔코더가 있는 스텝모터의 경우는 위치명령도 엔코더에 의하
		여 설정됨. 속도명령값을 설정하는 것인지, 전류명령값을 설정하는 것인지, 위치명령값을 설정하는 것
		인지는 SM 명령어의 의한 작동모드 설정에 따름. (이 설정은 F2811/F28334에서만 설정이 가능함)
		& bit14가 1이면 sequence control mode에서 스텝동작을 실시하며, 0이면 연속작동을 실시함. (이 설정은
		F2811/F28334에서만 설정이 가능함)
		& bit15가 1이면 sequence 제어가 실시되며, 0이면 sequence 제어를 실시하지 않음. (이 설정은
		F2811/F28334에서만 설정이 가능함)
		♣ 32비트의 리턴값은 TMS320F2810/2811/2812/28334/28335 로서 2009년 8월 16일 S/W 버젼부터 적용되며,
		TMS320LF2406A 또는 TMS320F2810/2811/2812/28334/28335 로서 2009년 8월 16일 이전 S/W 버젼은 16비트의
		리턴값이 적용됨.



명령어	구분	설명
		통신과 제어의 option을 선택하는 DIP switch의 값을 설정함.
		bit0-15 : SXHLHL; 의 경우와 같음.
		bit16 : 모터1의 엔코더에서 A/B 상의 신호를 서로 바꾸는 기능의 사용여부 [안함(0)/실시함(1)] (이 설정
		은 F2809/F28334에서만 설정이 가능하나, 2010년 7월 17일 S/W 버젼부터는 F2811에서도 가능하며, 2011년
		10월 16일 S/W 버젼부터는 절대각센서에도 적용됨)
		bit17 : 모터2의 엔코더에서 A/B 상의 신호를 서로 바꾸는 기능의 사용여부 [안함(0)/실시함(1)] (이 설정
		은 F2809/F28334에서만 설정이 가능함, 2010년 7월 17일 S/W 버젼부터는 F2811에서도 가능가능하며, 2011
		년 10월 16일 S/W 버젼부터는 절대각센서에도 적용됨)
		bit18 : 스테핑모터에서 50V 승압회로의 사용여부 [안함(0)/실시함(1)] (이 설정은 F28334에서만 설정이
		가능함) 속도모드에서 전압의 크기에 따라 가속/감속도를 조정하는 기능의 사용여부 [실시함(0)/안함(1)]
;SXHLHLHLHL;		(이 설정은 F28334에서만 설정이 가능함)
(+#^)		bit19 : 스테핑모터1에서 엔코더가 있는 경우에 lead angle control의 적용여부 [안함(0)/실시함(1)] (이
()	operation	설정은 F28334에서만 설정이 가능함)
;SX?; (+#^)		bit20 : 스테핑모터2에서 엔코더가 있는 경우에 lead angle control의 적용여부 [안함(0)/실시함(1)] (이
/G/(1/ (설정은 F28334에서만 설정이 가능함)
		bit21 : 스테핑모터에서 저속에서의 전류스위칭 소음을 줄이기 위하여 stepping motor의 전류제어게인을 모터속도에 따라서 가변하는 기능을 사용하는 여부 [안함(1)/실시함(0)] (이 설정은 F28334에서만 설정이
		고디국도에 따라서 가인하는 기능을 자용하는 여구 [인함(1)/설시함(0)] (이 설정는 F20034에서인 설정이 가능함)
		bit22 : 위치리밋초과 오류검출에 위치리밋센서와 가상위치리밋중 어느것을 사용하는지를 설정함 [가상위
		치리밋(1)/위치리밋센서(0)], 이 기능은 2011년 10월 01일 S/W 버젼부터 적용되며, 이전 S/W 버젼은 위치
		리밋센서만 사용함.
		bit24 : 위치제어에서 설정속도가 과도할 때 발생하는 문제를 처리 하는 여부 [안함(1)/실시함(0)] bit25 : 1ms main loop의 동기화와 위치/전류 명령값의 수신을 외부로부터의 CAN 데이터에 의하여 실시하
		는 여부 (2012년 12월 26일 S/W 버젼부터 정상 작동되며 CAN 통신에서만 가능) [안함(0)/실시함(1)]
		bit26 : 속도제어에서 감속에 따른 역기전력을 감소시키는 기능을 사용하는 여부 [안함(1)/실시함(0)]
		bit27 : 위치제어에서 PA/PB/Pa 명령수행시에 감속에 따른 역기전력을 감소시키는 기능을 사용하는 여부
		[안함(0)/실시함(1)]



명령어	구분	설명
		bit29:28: 00 일때 BLDC 모터제어에서 space vector PWM을 사용함. 01 일때 BLDC 모터제어에서 modified-SVPWM을 사용함. 10 일때 BLDC 모터제어에서 사인파 PWM을 사용함. 11 일때 BLOCK COMMUTATION 방식을 사용함.
		bit31 : 모터작동 불능 fault가 발생하였을때 모터제어기를 자동으로 위치제어모드로 재부팅하는 여부 [안함(0)/실시함(1)] (이 기능은 analog 센서타입의 BLDC 모터에는 적용되지 않음) bit31 : 일정변위를 연속으로 움직이게 하는 데모동작의 사용여부 [안함(0)/실시함(1)]
	return value	SXHLHLHLHL;
	Totalli valuo	2012년 5월 28일 S/W 버젼부터는 SXHLHLHLHL,HLHLHL;
		엔코더에서 A/B 상의 신호를 서로 바꾸는 것은 엔코더 카운트값이 증가하는 회전방향이 바뀐다는 것으로서, 물리적인 연결이 바뀌는 것은 아님. bit25가 1인 경우에 모터제어기는 상위로부터 총 8바이트를 수신하며 처음 7바이트와 8번째 바이트의 하위
	comment	4비트는 위치 또는 전류명령값 또는 자이로 각속도이며 8번째 바이트의 상위 4비트가 1/2/3/4 이면 위치명령/전류명령/각속도/사인파전류명령 임을 나타냄.
		bit26, bit27 과 관련한 파라메터는 SD 명령과 Sd 명령으로 설정됨. bit28, bit29로 BLDC 모터제어에서 사인파 PWM 제어 방식이 선택되었을때는 PWM 주파수를 80KHz 이내에서 만 사용할 것. (Z8 명령어 참조)
		SX 명령어에 의한 32비트의 설정은 TMS320F2810/2811/2812/28334/28335 로서 2009년 8월 16일 S/W 버젼부터 적용됨.
		bit31이 1 이더라도 모터작동 불능 fault의 사유가 아니고 정상적으로 전원을 투입한 경우라면 위치제어모드로 자동 부팅되지 않으며, analog 센서타입의 BLDC 모터의 경우는 원점을 감지하는 초기동작이 필요하므로 안전상 재부팅을 안하도록 하였음.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명					
;SXHLHLHLHL,HLHL HLHL; (+#^) ;SX?; (+#^)	operation	1st long word : SXHLHLHL; 의 경우와 같음. 2nd long word bit0/1/2/6 : 1 일때 모터1의 리밋센서 A/B/C/D 입력전압이 low 이면 active : 0 일때 모터1의 리밋센서 A/B/C/D 입력전압이 low 이면 active bit3/4/5/7 : 1 일때 모터2의 리밋센서 A/B/C/D 입력전압이 low 이면 active : 0 일때 모터2의 리밋센서 A/B/C/D 입력전압이 low 이면 active bit8 : 1 일때 03/05/06/0A번 제어모드에서 속도제어시에 speedx_RPM_combined_10을 사용함. (TMS320F28334/5 H/W 버젼에서만 펄스주기 방식에 의한 속도값이 적용되며 TMS320F2810/11/12에서는 펄스카운트 방식의 속도값이 적용됨) bit10/9 : 0 일때 03번 위치제어 sample time을 10ms 로 설정함. : 1 일때 03번 위치제어 sample time을 2ms 로 설정함. : 3 일때 03번 위치제어 sample time을 1ms 로 설정함. : 3 일때 03번 위치제어 sample time을 1ms 로 설정함. bit111 : digital Hall IC만을 사용하는 BLDC 모터를 제외한 경우에만 해당되며, bit110 0인 경우 05/06번 속도제어에서 속도제어루프에 저속에서 속도제어귀인을 낮추는 기능을 적용함. (Wd 명령어 참조) bit12 : 0 일때 1ms당의 펄스를 카운트하면서 1ms unit timer를 사용함.					
	return value	SXHLHLHLHL, HLHLHLL;					
	comment	bit 0~7의 기본값은 모든 비트가 1인 경우이며, 이때 해당 리밋센서의 입력전압이 low이면 리밋센서가 ON 된 것으로 판단함. 이러한 설정의 영향을 받는 경우는 리밋센서를 위치리밋센서로 사용하는 경우와 리밋센서입력으로 비상정지를 하는 경우등 리밋센서를 사용하는 모든 경우가 영향을 받음. 2012년 5월 28일 S/W 버젼부터 적용됨.					



명령어	구분	설명
;SXHLHLHLHL,HLHL HLHL; (+#^) ;SX?; (+#^)	comment	bit 8의 기본값은 10대 TMS320F28334/5를 사용한 H/W 버젼으로서 2012년 6월 12일 S/W 버젼부터, 이 비트가 0으로 설정되면 펄스주기에 의한 속도 계산값이 03/05번 제어모드에서 속도제어시에 사용되며 이 비트가 10기나 TMS320F2810/11/12에서는 03/05번 제어모드에서 속도제어시에 무조건 speedx_RPM_10을 사용함. 펄스주기에 의한 속도계산을 사용하려면 엔코더가 광학식인 것을 추천하며, Hall IC를 사용한 경우에는 펄스주기값의 fluctuation이 매우 심할수있음. bit10/9의 기본값은 3이며 2012년 10월 06일 S/W 버젼부터 유효함. bit11의 기본값은 1이며 2012년 10월 08일 S/W 버젼부터 유효함. bit12의 기본값은 1이며 2012년 10월 11일 S/W 버젼부터 유효함. bit13의 기본값은 1이며 2012년 10월 16일 S/W 버젼부터 유효함. CAN이나 RS485를 통하여 과도한 명령어 데이터가 전송되어 오는 경우는 사용하지 말고, 통신 테스트용으로만 사용할 것. bit15/14의 기본값은 3이며 2012년 10월 16일 S/W 버젼부터 유효함. 이 값은 EsA55A;에 의한 저장후 전원을 재투입하였을때 적용됨. 육 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



♣ TMS320LF2406A/TMS320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335의 작동모드는 16/32비트의 operation_mode_SWITCH 라는 변수명으로 설정되는데, SX 명령어로 설정하고 EsA55A; 명령어로 EEPROM에 저장할 수 있음. 단, DIP 스위치가 있는 일부 초기 H/W 버젼의 경우에 bit0/6/5/4는 DIP 스위치의 상태를 반영하고 있으며 나머지 비트는 EEPROM에 저장할 수 있음.

TMS	TMS320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335 (SX 명령어 첫번째 파라메터)							
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
bit - bit	U=Sequence 데어를 실해	1=한스텝씩 sequence 제어 를 실행함. 0=연속 sequence 제어 를 실행함. (2406a 에서는 해당없음)	1=LSxC입력단자 의 전압에 따라 서 속도명령 또 는 전류명령을 설정함. 엔코 더가 있는 스텝 모터의 경우는 위치명령도 엔코 더에 의하여 설 정됨. (2406a 에 서는 해당없음)	1=속도제어시에 (가상)위치리밋센 서를 만나면 정지 하고, 반대방향으 로만 속도값명령이 설정가능토록 함. 0=속도제어시에 (가상)위치리밋센 서를 사용안함. (2406a 에서는 해 당없음)	1=analog 센 서 타입의 RJM_VER7에서 Hall sensor 파라메터를 매 출발시 마 다 계산함. (2406a 에서 는 해당없음)	1=위치제어시의 (가상)위치리밋센 서를 적용안함. 0=위치제어시에 (가상)위치리밋센 서를 만나면 정지 하고, 반대방향으 로만 위치값 설정 이 가능하게함. (일부기능은 2406a 에서는 해당없음)	1=속도제어중 정 지시에 적분항을 0으로 처리함. (제일메디컬은 무 조건 0으로 처리) digital Hall IC 만 사용하는 경우 는 속도/위치제어 에 모두 적용되며 필수 선택사항임.	1=dummy 작동 모드 실행함. 0=dummy 작동 모드 실행안 함. (DC/BLDC 모터만 해당 되며, 2406a 에서는 해당 없음)
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
bit' - bit	사용함. N=1 QA 1 QC를	2번 스위치 (2406a 에서는 통신모드 설정 에 사용함)	3번 스위치 통신테스트모드 외부표시모드	4번 스위치 통신테스트모드 외부표시모드	1=모터 power 를 계산함. 0=모터 power 를 계산하지 않음.	1=모터권선온도 리 밋기능 사용함. 0=모터권선온도 리 밋기능 사용안함.	1=모터1과 모터2 가 직선보간운동 을 실시함. 0=모터1과 모터2 가 직선보간운동 을 실시하지 않 음.	1번 스위치 OFF(1)=RS485 ON(0)=CAN

* LSxC 포트를 analog 입력으로 사용할때, bit10에 의한 위치리밋의 적용에서 발생하는 문제를 해결하는데 bit7을 사용함.



•	TMS320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335 (SX 명령어 첫번째 파라메터)								
		bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	oit31 - oit24	1=데모작동을 실시함. 0=데모작동을 멈춤.	1=모터작동 불능 fault가 발생하였을때 모터제어기를 자동으로 위치제어모드로 재부팅함. (analog센서타입의 BLDC 모터에는 적용되지 않음) 0=재부팅 안함.	11=BLDC 모터제어에서 BLOCK COMMUTATION 방식을 사용함. 10=BLDC 모터제어에서 pure sine PWM 방식을 사용함. (수행시간 때문에 PWM 주파수를 80KHz 이내 에서만 사용할 것) D1=BLDC 모터제어에서 modified—SVPWM 방식을 사용함.		1 = 위치제어에 서 PA/PB/Pa 명 령수행시에 감 속에 따른 역기 전력을 감소시 키기 위하여, 감속도를 제한 하는 기능을 사 용함. 0=사용하지 않 음.	1 = 속도제어에 서 감속에 따른 역기전력을 감 소시키기 위하 여, 감속도를 제한하는 기능 을 사용하지 않 음. 0=사용함.	1=1ms main loop의 기동을 CAN 데이터가 수신될때 실시 함. (DC 모터로 서 CAN 통신에 서만 가능) 0=1ms main loop의 기동을 내부타이머로 실시함.	1=위치제어에서 설정속도가 과 도하여 위치제 어 오차가 매우 커질 때 발생하 는 문제를 처리 하지 않음. 0=처리함.
		bit23	bit22	bit21	bit20	bit 19	bit18	bit 17	bit16
	oit23 - oit16		1=위치리밋초과 오류검출에 가 상위치리밋만을 사용함. 0=위치리밋초과 오류검출에 위 치리밋센서만을 사용함.	1=저속에서 전 류스위칭 소음 을 줄이기 위하 여, stepping motor의 전류제 어게인을 모터 속도에 따라서 가변하는 기능 을 사용하지 않 음. 0=사용함.	1=stepping 모 터2의 lead angle 제어를 실시함. (반드 시 엔코더가 부 착되어 있어야 함) 0=stepping 모 터2의 lead angle 제어를 실시안함.	1=stepping 모 터1의 lead angle 제어를 실시함. (반드 시 엔코더가 부 착되어 있어야 함) 0=stepping 모 터1의 lead angle 제어를 실시안함.	1=stepping 모 터의 승압회로 를 사용함. 0=stepping 모 터의 승압회로 를 사용안함.	1=엔코더2의 A/B상 신호를 서로 바꿈. 0=엔코더2의 A/B상 신호를 서로 안바꿈.	1=엔코더1의 A/B상 신호를 서로 바꿈. 0=엔코더1의 A/B상 신호를 서로 안바꿈.



♣ 2012년 5월 28일 S/W 버젼부터 사용이 가능한 operation_mode_SWITCH_more 라는 변수명으로 설정되는데, SX 명령어로 설정하고 EsA55A; 명령어로 EEPROM에 저장할 수 있음. (SX 명령어로 설정하는 파라메터가 기존의 1개에서 2개로 늘었음)

1	TMS320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335 (SX 명령어 두번째 파라메터)								
		bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	אזומ	11=CAN 11-bit 주소로 0-255만 10=CAN 11-bit 주소로 0-2047 10월 16일부터 01=CAN 29-bit 주소로 0-65535 년 10월 16일부 00=CAN 11-bit 주소로 0-255만	사용. 주소모드로서 사용, 2012년 적용. 주소모드로서 만 사용, 2012 터 적용. 주소모드로서	0일때 CAN이나 RS485 통신 포트 를 통하여 입력 된 명령을 RS232C 포트로 전송함. 1일때 상기 작동 을 하지 않음.	0 일때 1ms당의 펄스를 카운트하 면서 1ms unit timer를 사용함 (TMS320F28334/5 H/W 버젼인 경우 로서 엔코더를 사용하는 경우에 만 적용 가능함) 1 일때 1ms당의 펄스를 카운트하 면서 1ms unit timer를 사용하 지 않음.	만을 사용하는 BLDC 모터를 제외	bit10:bit9의 값(03번 위치제어에/ 프의 sample time 설정함.	서 위치제어 루	1 일때 모터의 속도제어시에 speedx_RPM_10 을 사용하며, 0 일때 speedx_RPM_com bined_10(TMS32 0F28334/5 H/W 버젼에서만 펄 스주기 이용 방 식의 속도값이 고 그외는 카운 트방식의 속도 값)을 사용함
		bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	oit7 - oit0	1 일때 모터2 의 리밋센서 D 입력전압이 low 이면 active, O 일 때 high active	1 일때 모터1 의 리밋센서 D 입력전압이 low 이면 active, O 일 때 high active	1 일때 모터2의 리밋센서 C 입력 전압이 low 이면 active, 0 일때 high active	1 일때 모터2의 리밋센서 B 입력 전압이 low 이면 active, 0 일때 high active	1 일때 모터2의 리밋센서 A 입력 전압이 low 이면 active, 0 일때 high active	1 일때 모터1의 리밋센서 C 입력 전압이 low 이면 active, 0 일때 high active	1 일때 모터1의 리밋센서 B 입 력전압이 low 이면 active, 0 일때 high active	1 일때 모터1의 리밋센서 A 입 력전압이 low 이면 active, 0 일때 high active



♣ 2012년 5월 28일 S/W 버젼부터 사용이 가능한 operation_mode_SWITCH_more 라는 변수명으로 설정되는데, SX 명령어로 설정하고 EsA55A; 명령어로 EEPROM에 저장할 수 있음. (SX 명령어로 설정하는 파라메터가 기존의 1개에서 2개로 늘었음)

TMS320	TMS320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335 (SX 명령어 두번째 파라메터)								
	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24	
bit31 - bit24									
	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16	
bit23									
bit16									

명령어	구분	설명
		모터 1회전당 펄스수와 POLE PAIR 갯수를 설정함.
		1st word : motor1의 1회전당 펄스수 [1-32767] (MOTOR_TYPE에 따름)
	operation	2nd word : motor2의 1회전당 펄스수 [1-32767] (MOTOR_TYPE에 따름)
		3rd word : MOTOR1_NUMBER_of_POLE_PAIR [1-20] (MOTOR_TYPE에 따름)
		4th word : MOTOR2_NUMBER_of_POLE_PAIR [1-20] (MOTOR_TYPE에 따름)
		SEHLHL, HLHL, HLHL, (2012년 1월 24일 이전 S/W 버젼)
;SEA55A,HLHL,HLH	return value	SEHLHL, HLHL, HLHL, HLHL, (2012년 1월 24일 S/W 버젼부터)
L, HLHL, HLHL;		5th word : PWM_PERIOD (단위 크기는 1/120 uS, 2012년 1월 24일 S/W 버젼부터 적용됨)
(+#^%)		♣ 4체배된 엔코더 펄스수로 정의 됨.
(111 /6)		&기본값으로 MAXON RE-max 29/24V/22W DC 모터는 1024, FAULHARBER 3257CR 24V/83W DC 모터는 2048,
;SE?; (+#^%)		MAXON EC 45 flat 24V/50W(△) BLDC 모터는 4096, MAXON EC-powermax 30 24V/100W(△) BLDC 모터는 4000,
,OL:, (111 10)		MAXON EC 45 24V/250W(△) BLDC 모터는 2000, FAULHARBER 4490 24V/200W(△) BLDC 모터는 2000, MAXON EC
	comment	90 flat 24V/90W BLDC 모터는 2000임.
	Commert	✿ MOTORx_NUMBER_of_POLE_PAIR는 DC 모터의 경우는 무조건 1이며, BLDC 모터의 경우 pole pair의 수를 의
		미하며, STEP 모터의 경우는 90°/step각 을 의미함.
		♣ 이 명령은 EDA55A;를 실행한 이후이거나, ST 명령에 의한 MOTOR_TYPE 설정이 0xffff 이거나, 또는 모터
		를 enable 하지 않고 작동모드도 0일때에만 적용이 가능함.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명					
	operation	감속비와 최대회전속도를 설정함. 1st word: 모터 1의 Gear Ratio [1 - 2000] (MOTOR_TYPE에 따름) 2nd word: 모터 2의 Gear Ratio [1 - 2000] (MOTOR_TYPE에 따름) 3rd word: 모터 1의 최대속도 [10 - 32000/60000] (rpm, MOTOR_TYPE에 따름) 4th word: 모터 2의 최대속도 [10 - 32000/60000] (rpm, MOTOR_TYPE에 따름)					
;SGA55A, dddd1, dd	return value	SGdddd1,dddd2,dddd3,dddd4;					
dd2,dddd3,dddd4; (+#^%) ;SG?; (+#^%)	comment	♣ 이 명령은 EDA55A;를 실행한 이후 또는 모터 type이 0xfff 일때에만 적용이 가능함. 그러나 2011년 10월 16일 S/W 버젼부터는 PE 명령에 의한 작동개시를 하지 않고, 작동모드도 0일때에는 실행이 가능함. ♣ Gear Ratio는 모터에 연결된 감속기의 감속비를 의미하는데, 모터 출력축에 변위센서가 설치된 경우에는 그 값이 부정확하여도 큰 문제가 되지는 않으나, 감속기 출력단에 변위센서가 장착된 경우에는 정확한 값을 설정하여야함. Iinear 모터인 경우에는 감속기가 없는 구조이기는 하나, 직선 구동부 레일에 장착된 영구자석 N-S 극 pair의 총개수의 1/2 값으로 Gear Ratio 값을 설정하면 됨. ♣ 최대속도값은 2012년 12월 29일 이전 S/W 버젼에서는 1000부터, 그때 버젼 부터는 10부터 설정가능. ♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.					



명령어	구분	설명
;Sgiiii1,iiii2,i iii3,iiii4; (+#^) ;Sg?; (+#^)	operation	위치제어 또는 속도제어에서 phase lead 또는 phase lag compensator의 파라메터를 설정함. (이 경우는 미분제어 게인이 무시됨) 1st integer: COMPENSATOR_CENTER_FREQ1 [1 - 63] (50 Hz) 2nd integer: COMPENSATOR_ALPHA1 [-100 - 100] (0) 3rd integer: COMPENSATOR_CENTER_FREQ2 [1 - 63] (50 Hz) 4th integer: COMPENSATOR_ALPHA2 [-100 - 100] (0)
	return value	Sgiiii1,iiii2,iiii3,iiii4;
	comment	 ♣ COMPENSATOR_CENTER_FREQX는 대략 Hz 단위의 compensator 중심주파수를 의미함. ♣ COMPENSATOR_ALPHAX가 양수일 때는 phase lead compensator이며, 음수일 때는 phase lag compensator이고 0일 때는 compensator를 적용하지 않음. ♣ COMPENSATOR_ALPHAX가 양수일 때(phase lead compensator 일때)는 PI 만을 적용하고 PID를 적용하지 않음. ♣ COMPENSATOR_ALPHAX는 gain_increase(COMPENSATOR_ALPHAX가 양수일 때) 또는 gain_reduction(COMPENSATOR_ALPHAX가 음수일 때)을 설정하는 값임. ♣ COMPENSATOR_ALPHAX에 따라서 phase_lead/phase_lag 각의 크기도 값이 변화되며 COMPENSATOR_ALPHAX가 10/8/6/4/2일 때 약 55도/52도/46도/37도/20도의 phase_lead/phase_lag 각이 설정됨. ♣ F2811/F28334 에서만 가능한 명령임. ♣ 2008년 7월 23일 S/W 버젼부터 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. ♣ 2008년 10월 5일 S/W 버젼부터 COMPENSATOR_CENTER_FERQX 대신 COMPENSATOR_ALPHAX 값이 양수 또는 음수로 설정이 되도록 바꾸었음.



명령어	구분	설명
;SL 1, 2, 13, 4; (+#^%) ;SL?; (+#^%)	operation	위치제어모드에서 변위의 한계값을 설정함.
		1st long integer : motor1 설정변위의 상한값 (감속기출력 기준으로 +350도)
		2nd long integer : motor1 설정변위의 하한값 (감속기출력 기준으로 -350도)
		3rd long integer : motor2 설정변위의 상한값 (감속기출력 기준으로 +350도)
		4th long integer : motor2 설정변위의 하한값 (감속기출력 기준으로 -350도)
	return value	SLIII1, III12, III13, III14;
		♪ 상하한설정변위값은 motor1의 경우는 motor1의 현재위치값인 position1을 포함하는 범위이어야만 유효
	comment	하며, motor2의 경우 motor2의 현재위치값인 position2을 포함하는 범위에서이어야만 유효.
, SL: , (I# /0)		⇔ 상한값이 하한값보다는 커야만 유효함. 원점위치는 5,000,000임.
	Collinio	♪ 각종 위치명령값을 설정할때에만 크기를 제한하는 용도로 쓰임.
		♪ F2811/F28334의 경우 SX 명령어로 설정하는 bit10 값이 0이면 변위설정값의 제한이 실제로 가동됨.
		🕹 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
	operation	홈찾기 동작에서 변위의 한계값을 설정함.
		1st long integer : motor1 홈찾기 동작의 상한값 (감속기출력 기준으로 +350도)
		2nd long integer : motor1 홈찾기 동작의 하한값 (감속기출력 기준으로 -350도)
;\$11111,11112,1		3rd long integer : motor2 홈찾기 동작의 상한값 (감속기출력 기준으로 +350도)
1113,11114; (+#^		4th long integer : motor2 홈찾기 동작의 하한값 (감속기출력 기준으로 -350도)
) ;SI?; (+#^)	return value	\$11111,11112,11113,11114;
	comment	♣ 상하한설정변위값은 motor1의 경우는 motor1의 현재위치값인 position1을 포함하는 범위이어야만 유효
		하며, motor2의 경우 motor2의 현재위치값인 position2을 포함하는 범위에서이어야만 유효.
		♣ 상한값이 하한값보다는 커야만 유효함. 원점위치는 5,000,000임.
		축 홈찾기 동작을 할 때에만 크기를 제한하는 용도로 쓰임.
		♪ EEPROM에 저장되지 않으며 기본값은 SL 명령에 의하여 설정된 값임.



명령어	구분	설명
;SSdddd1,dddd2; (+#^%)	operation	위치제어모드에서 Point-to-Point 이동시 이동속도값을 설정함.
		1st word : motor1 속도명령값 [1 ~ 32000/60000] (1000/60 rpm)
		2nd word : motor2 속도명령값 [1 ~ 32000/60000] (1000/60 rpm)
;SS?; (+#^%)	return value	SSdddd1,dddd2;
	comment	♣ 단위는 rpm이며 RJM_VER6의 기본값은 60rpm, 나머지는 1000rpm임.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
;Ssdddd1,dddd2; (+#^%) ;Ss?; (+#^%)	operation	PA, PB, Pa 명령에 의한 변위구동시의 가속/감속 기간을 설정함.
		1st word : motor1의 가속/감속기간 [1 ~ 10000] (250 or 1000 ms)
		2nd word : motor2의 가속/감속기간 [1 ~ 10000] (250 or 1000 ms)
	return value	Ssdddd1,dddd2;
	comment	♣ 단위는 ms임.
		♣ 기본값은 STEP RJM_VER5의 경우 250ms이며 나머지의 경우는 1000ms임.
		⇔ SS 명령어로 정속구간에서의 속도를 먼저 설정해야만 Ss 명령어에 의한 가감속도 설정이 정상적으로 이
		루어짐. Sa 명령어로도 가감속도 설정이 가능한데 이때는 SS 명령어에 의한 정속구간에서의 속도와 무
		관하게 설정이 가능함.
		⇔ 엔코더의 펄스수가 적거나 가속/감속 기간이 큰 경우 Ss 명령어로 설정된 값과 실제의 가속/감속기간의
		오차가 있을수 있음. 이 경우에는 펄스수가 작은 경우에도 적용할 수 있는 firmware를 사용하여야함.
		♣ RJM_VER5의 경우는 500 이 최대 설정값임.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	가속/감속시 저전압/과전압이 발생하는 것을 억제하기 위하여 가감속률을 제한하는 파라메터를 설정함.
		1st word : motor1의 가속도 저감률 [0 ~ 65535] (100 rpm/100ms/V or 1 rpm/ms/V)
		2nd word : motor1의 감속도 저감률 [0 ~ 65535] (400 rpm/100ms/V or 4 rpm/ms/V)
		3rd word : motor2의 가속도 저감률 [0 ~ 65535] (100 rpm/100ms/V or 1 rpm/ms/V)
;Sddddd1,dddd2,d		4th word : motor2의 감속도 저감률 [0 ~ 65535] (400 rpm/100ms/V or 4 rpm/ms/V)
ddd3,dddd4; (+#^	return value	Sddddd1,dddd2,dddd3,dddd4;
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \		✿ RJM_VER7에서는 단위가 rpm/ms/V이며, 그외에는 rpm/100ms/V임. (확인요)
,	comment	♣ 설정값이 0이면 가감속도를 조정하는 일을 하지 않음.
;Sd?; (+#^)		♧ (operation_mode_SWITCH & 0x04000000L)==0L 일때 속도제어모드에서 가감속도 조정을 실시함.
,οα:, (۱π /		♧ (operation_mode_SWITCH & 0x08000000L)!=0L 일때 위치제어모드에서 가감속도 조정을 실시함.
		& 파라메터는 각각 BLDC1_PrmBuffer[36]/BLDC1_PrmBuffer[37]/BLDC2_PrmBuffer[36]/BLDC2_PrmBuffer[37]
		에 저장됨.
		♪ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		⇔ F2811 <mark>/F28334</mark> 에서만 가능한 명령임.
	operation	가속/감속시 저전압/과전압이 발생하는 것을 억제하기 시작하는 전압변동량을 설정함.
;SDdddd1,dddd2;		1st word : motor1의 전압변동량 [0 ~ 최고사용전압/4] (4 V, 제일메디컬은 1 V)
(+#^)		2nd word : motor2의 전압변동량 [0 ~ 최고사용전압/4] (4 V, 제일메디컬은 1 V)
(+# ')	return value	SDdddd1,dddd2;
;SD?; (+#^)		♣ 최고사용전압(Z6 명령어로 설정함)이 12V 보다 작거나 600V보다 크면 30V 로 취급함.
	comment	♪ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		♣ F2811 /F28334 에서만 가능한 명령임.



명령어	구분	설명
		OE/OF 위치제어모드에서 모터1의 sine파 변위 주기 및 진폭을 설정함.
;S1dddd1,dddd2;	operation	1st word : 주파수 [100 - 30000] (1000 0.001Hz)
(+#^)		2nd word : 진폭 [0 - 32767] (0 pulse)
(1#)	return value	S1dddd1,dddd2;
;S1?; (+ #^)		🕹 SM mode Of에서 sine파 위치제어가 실시됨.
, o i : , (i #)	comment	♣ 스테핑모터에서는 주기가 최대 1310ms 까지만 가능함.
		♣ F2811/F28334 에서만 가능한 명령임.
		OE/OF 위치제어모드에서 모터2의 sine파 변위 주기 및 진폭을 설정함.
;S2dddd1,dddd2;	operation	1st word : 주파수 [100 - 30000] (1000 0.001Hz)
(+#^)		2nd word : 진폭 [0 - 32767] (0 pulse)
(π)	return value	S2dddd1,dddd2;
;S2?; (+#^)		🗫 SM mode Of에서 sine파 위치제어가 실시됨.
, o.e., (''')	comment	🕹 스테핑모터에서는 주기가 최대 1310ms 까지만 가능함.
		🗘 F2811/F28334 에서만 가능한 명령임.
		속도제어 및 위치제어 모드에서 리밋센서, 비상정지 명령 또는 비상정지 상황에 의한 강제 감속시의 감속
;Sodddd1,dddd2;	operation	율을 설정함.
(+#^)		1st word : motor1 감속도 설정값 [1 ~ 60000] (9523 rpm/100ms or 10 rpm/ms)
(π)		2nd word : motor2 감속도 설정값 [1 ~ 60000] (9523 rpm/100ms or 10 rpm/ms)
;So?; (+#^)	return value	Sodddd1,dddd2;
,00:, (''')	comment	단위는 RJM_VER7/RJM_VER7에서는 rpm/ms, 나머지 H/W 버젼에서는 rpm/100ms임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		구동회로와 기구를 보호하기 위한 logic에서 사용하는 전류리밋값과 time threshold를 설정함.
		1st word : motor1 전류리밋 [0 - 1023/1638 or 8190(380V용)] (1638)
		2nd word : motor2 전류리밋 [0 - 1023/1638 or 8190(380V용)] (1638)
		3rd word : 모터1의 전류초과를 허용하는 시간 [1 - 50000] (1000 ms)
	operation	4th word : 모터2의 전류초과를 허용하는 시간 [1 - 50000] (1000 ms)
		5th word : 모터1의 전류초과를 허용하는 시간 대비 전류를 끄는 기간의 비율 [1 - 10] (5)
		6th word : 모터2의 전류초과를 허용하는 시간 대비 전류를 끄는 기간의 비율 [1 - 10] (5)
;SIdddd1,dddd2,d		7th word : 모터1의 전류초과시에 위치명령을 hold하기 까지의 시간 [1 - 10000] (500 ms)
ddd3, dddd4, dddd5		8th word : 모터2의 전류초과시에 위치명령을 hold하기 까지의 시간 [1 - 10000] (500 ms)
, dddd6, dddd7, ddd	return value	SIdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;
d8; (+#^%)		1A×25mΩ×11(앰프 증폭률) = 0.275V이므로 3.3V/3.0V는 3.3V÷0.275V = 12A/10.9A에 해당하며 85.25
αο, (۱π /ο)		/375.375 bit/A. (DC ver1.1)
;SI?; (+ # ^%)	comment	1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3V/3.0V는 3.3V÷0.25V = 13.2A/12A에 해당하며 77.5/341.25
,01:, (117 70)		bit/A. (DC V1.2부터)
(Force/Moment 센		1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3V/3.0V는 3.3V÷0.1V = 33A/30A에 해당하며 31/136.5 bit/A.
서가 아닌 경우)		(Piggyback-DC/RJM_VER8부터)
MOTORE 817		1A×5mΩ×50(앰프 증폭률)×3.3/5 = 0.165V이므로 3.3V/3.0V는 3.3V÷0.165V = 20A/18.18A에 해당하며
		51.15/225.225 bit/A. (Piggyback-BLDC)
		1A×10mΩ×50(앰프 증폭률) = 0.5V이므로 3.3V/3.0V는 3.3V÷0.5V = 6.6/6A에 해당하며 155/682.5
		bit/A. (mini-DC)
		1A×20(10)mΩ×50(앰프 증폭률) = 1.00V/0.5V이므로 3.3V/3.0V는 3.3V÷1.00V = 3.3A/6A에 해당하며
		310/682.5 bit/A. (STEP형)
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 5.0V는 5.0V÷0.25V = 20A에 해당하며 204.75 bit/A.
		(Sensor-BLDC)



명령어	구분	설명
	comment	따라서 DC V1.1 인 경우에 0 - 1023/4095의 값은 0 - 12/10.9A을 전류명령값을 의미함. 1[A]=85/375, 2[A]=170/751, 3[A]=256/1126, 4[A]=341/1502, 5[A]=426/1877, 6[A]=512/2252, 7[A]=597/2628, 8[A]=682/3003, 9[A]=768/3378, 10[A]=853/3754, 11[A]=938 일. DC V1.2부터는 1[A]=78/341, 2[A]=155/683, 3[A]=233/1024, 4[A]=310/1365, 5[A]=388/1706, 6[A]=465/2048, 7[A]=543/2389, 8[A]=620/2730, 9[A]=698/3071, 10[A]=775/3413, 11[A]=853/3754, 12[A]=930, 13[A]=1008 일. Piggyback-DC/RJM_VER8부터는 1[A]=31/137, 2[A]=62/273, 3[A]=93/410, 4[A]=124/546, 5[A]=155/683, 6[A]=186/819, 7[A]=217/956, 8[A]=248/1092, 9[A]=279/1229, 10[A]=310/1365, 11[A]=341/1502, 12[A]=372/1638, 13[A]=403/1775, 14[A]=434/1911, 15[A]=465, 16[A]=496 일. Piggyback-BLDC인 경우는 1[A]=51/225, 2[A]=102/450, 3[A]=153/676, 4[A]=205/901, 5[A]=256/1126, 6[A]=307/1351, 7[A]=358/1577, 8[A]=409/1802, 9[A]=460/2027, 10[A]=512/2252, 11[A]=563/2477, 12[A]=614/2703, 13[A]=665/2928, 14[A]=716/3153, 15[A]=767/3378 일. mini-DC 에서는 1[A]=155/683, 2[A]=310/1365 일. STEP 에서는 0.25[A]=78/171, 0.5[A]=155/341, 0.75[A]=233/512, 1[A]=310/683, 1.25[A]=388/853, 1.5[A]=465/1024, 1.75[A]=1194, 2[A]=1365, 2.25[A]=1536, 2.5[A]=1706, 2.75[A]=1878 일. Sensor-BLDC는 1[A]=205, 2[A]=410, 3[A]=614, 4[A]=819, 5[A]=1024, 6[A]=1229, 7[A]=1443, 8[A]=1638, 9[A]=1843 일.
		2008년 6월 이후의 모든 F2811/F28334 H/W 버젼에서는 다음과 같이 통일 되었음. 2mΩ인 경우는 81.9/81.9 bit/A (RJM_VER3 : IRMCK201을 사용한 BLDC제어기, RJM_VER8 : 고출력 DC) 5mΩ인 경우는 204.75/204.75 bit/A (RJM_VER7 : IRMCK201을 사용하지 않은 BLDC제어기) 10mΩ인 경우는 409.5/409.5 bit/A (RJM_VER4 : mini DC, RJM_VER5 : step motor)
		RJM_VER3/7/8 의 경우는 각각의 모터구동 BRIDGE 회로의 high side 쪽으로 총전류감지저항을 통해서 들어 가는 전류를 감지하고 이를 사용하여 전류리밋을 적용하나, RJM_VER1/2/4/5/6 의 경우는 코일에 흐르는 상 전류를 사용하여 전류 리밋을 적용함.



명령어	구분	설명
		2012년 4월 12일 S/W 버젼부터는 SI 명령어로 설정되는 전류 리밋은 상전류의 크기이며, 총전류는 그 값의 2/3가 리밋으로 적용됨. SI 명령어로 설정되는 전류 리밋값는 Z7 명령어로 설정되는 전류최대치보다 크지 않게 제한되며, Sw 명령어로 설정되는 제어전류 최대치보다 클수 없음. SI 명령어로 설정되는 전류 리밋의 125%에 해당하는 전류값을 단 한번이라도 초과하면 과전류 오류를 발생시키는데, 이 값이 Sw 명령어로 설정하는 전류치의 125%보다 작다면 Sw 명령어로 설정하는 전류치의 125%로 강제 설정됨.
	comment	2012년 5월 28일 S/W 버젼부터는 모터x의 전류초과를 허용하는 시간값(SI 명령어의 세번째/네번째 값)의 3/4 크기보다 Si 명령어로 설정하는 값이 작아지지 않게 (전원투입하여 마이컴 초기화 할때) 하였음.
		2012년 12월 19일 이전의 S/W 버젼에서는 1채널제어기에서 SI 명령으로 두번째값을 작게 설정하면 오류를 발생시키므로 첫번째 값과 같은 값을 설정해야함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는 0/1/2/3) 전압의 평균값을 설정하기 위한 filtering
	operation	을 실시하고, 온도값과 함께 force_data_offset 변수값을 설정함.
;SIA55A,dddd1,dd		1st word : 필터링 시간 [10 - 10000] (xxxxx ms)
dd2; (+#^)		2nd word : 필터 시정수 [4 - 1024] (xxxxx)
		SIdddd1,dddd2;
;SI?; (+#^)	return value	1st word : filtering 타이머의 남은 시간 (ms)
		2nd word : 필터 시정수 (ms)
(Force/Moment 센		필터시정수로 필터링시간동안 필터링을 행하여 force_data_offset 변수을 설정하며, 설정된 값은 Qb2; 명
서인 경우만 해당		령어로 읽어낼수있음.
됨)	comment	F/T 센서를 조립하였다가 풀었을때는 SIA5A5,0,0; 명령어로 F/T센서의 앰프 offset을 2048로 맞추어준 이
		후에 SIA55A,dddd1,dddd2; 명령을 사용할 것.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는 0/1/2/3) 전압의 평균값을 설정하기 위한 filtering
	operation	을 실시하고, 온도증가에 따른 전압상승의 기울기를 설정함.
;SI5AA5,dddd1,dd		1st word : 필터링 시간 [10 - 10000] (xxxx ms)
dd2; (+#^)		2nd word : 필터 시정수 [4 - 1024] (xxxxx)
uuzy (TII)		SIdddd1,dddd2;
;SI?; (+#^)	return value	1st word : filtering 타이머의 남은 시간 + 15000 (ms)
,01., (111		2nd word : 필터 시정수 (ms)
(Force/Moment 센		필터시정수로 필터링시간동안 필터링을 행하여 전압상승의 기울기를 설정하며, 설정된 값은 MS?; 명령어로
서인 경우만 해당		읽어낼수있음.
됨)	comment	SI5AA5 를 수행하기 전에 F/T 센서의 온도를 일정하게 유지한 상태로 SIA55A 명령을 먼저 수행하여야함.
_,		SI5AA5 를 수행할때도 F/T 센서의 온도를 일정하게 유지하되, 이전의 SIA55A 명령을 수행때와는 다른 온도
		에서 실시해야함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
;SIA5A5,0,0; (+#^)	operation	F/T 센서에서 no load 상태일때 AD channel 8/9/10/11(내장형의 경우는 0/1/2/3)의 변환값이 2048이 되도록 D/A 변환기를 조절하여 만듦. (F/T센서의 앰프 offset을 2048로 맞추어주는 작동이며 총 5초가 걸림) 1st word : 의미 없음. 2nd word : 의미 없음.
;\$1?; (+#^)	return value	Sldddd1,dddd2; 1st word : filtering 타이머의 남은 시간 + 30000 (ms) 2nd word : 필터 시정수 (ms)
(Force/Moment 센		Qx3 명령어로 모니터링을 하면 SIA5A5,0,0; 명령을 보낸후 5초이내에 4채널의 값이 모두 2048 근방의 값을
서인 경우만 해당됨)	comment	갖게 됨. F/T 센서를 조립하였다가 풀었을때는 반드시 이 명령어를 한번 수행하고 그 결과값을 EsA55A; 명령어로 저장해주어야함. 이때 F/T 센서에는 외력이 없는 상태이어야함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.

명령어	구분	설명
	operation	과전류시에 HW적으로 회로를 차단하기 위한 과전류 설정값을 설정함. 1st word : motor1 전류리밋 : DC의 경우 [0 - PWM_PERIOD] (PWM_PERIOD)
	return value	SWdddd1,dddd2;
;SWdddd1,dddd2; (+#^%) ;SW?; (+#^%)	comment	항상 설정이 가능하며 전원투입시의 기본값은 PWM_PERIOD(DC의 경우) 또는 400/0(Piggyback-BLDC나 TMS320LF2406A BLDC의 경우)임. EEPROM에 저장되지 않음. 최대값을 설정하면 3.3V/3.0V의 전압이 HW적인 과전류 차단회로로 공급되는데,이때 다음과 같이 적용됨 1A×25mΩ×11(앰프 종폭률) = 0.275V이므로 3.3V/3.0V는 3.3V÷0.275V = 12A/10.9A의 과전류 설정치에 해당함. (DC ver1.1) - 1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3V/3.0V는 3.3V÷0.25V = 13.2A/12A의 과전류 설정치에 해당함. (DC V1.2부터) - 1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3V/3.0V는 3.3V÷0.1V = 33A/30A의 과전류 설정치에 해당함. (Piggyback-DC) - 1A×5mΩ×50(앰프 증폭률) ×3.3/5 = 0.165V이므로 3.3V/3.0V는 3.3V÷0.165V = 20A/18.18A의 과전류 설정치에 해당함. (Piggyback-BLDC 또는 TMS320LF2406A BLDC) 따라서 DC V1.1 인 경우에 0 - PWM_PERIOD의 값은 0 - 12A/10.9A의 전류제한값을 의미함. DC V1.2에서는 0 - PWM_PERIOD의 값은 0 - 13.2A/12A의 전류제한값을 의미함. Piggyback-DC에서는 PWM_PERIOD/2 - PWM_PERIOD의 값은 0 - 16.5A/15A의 전류제한값을 의미함. Piggyback-DC에서는 PWM_PERIOD/2의 offset을 더한 값을 과전류설정값으로 설정하여야함. BLDC V1.1에서는 PWM_PERIOD/3.96의 offset을 더한 값을 과전류설정값으로 설정하여야함. BLDC V1.1에서는 PWM_PERIOD/3.96의 offset을 더한 값을 과전류설정값으로 설정하여야함. 구형 DC, Piggyback-BLDC, TMS320LF2406A BLDC에서만 과전류 차단기능이 있으며, 그외의 모델에서는 HW적인 과전류 차단기능이 없음.



operation	제어시 사용하는 전류의 리밋을 설정함. 1st word: motor1 제어전류리밋: (8190 or 6142 or 1638, 2009년 7월 15일 S/W 버젼부터는 comment 마지막부분 참조) 2nd word: motor2 제어전류리밋: (8190 or 6142 or 1638, 2009년 7월 15일 S/W 버젼부터는 comment 마지막부분 참조)
return value	Swdddd1,dddd2;
	전류를 사용하여 전류나 위치나 속도제어를 하는 경우에 사용하는 전류의 최대치를 설정함. IRMCK201을 사용하는 RJM_VER3인 경우는 4095 값은 정격전류를 의미함.
comment	2009년 7월 15일 S/W 버젼부터 모든 F2811/F28334 H/W 버젼에서는 다음과 같은 적용을 받음. KOMOTEK 380V모터의 경우 설정가능한 값의 범위는 -8190 ~ +8190 이며, 전류감지저항으로 10m0hm을 사용하는 경우 -20A ~ +20A 의 범위에 해당함. IRMCK201을 사용한 경우 설정가능한 값의 범위는 -12288 ~ +12288 이며, Zc 명령어로 설정되고 IRMCK201의 파라메터설정에 사용한 정격전류의 3배에 해당하는 범위임. 그 외의 경우 설정가능한 값의 범위는 -1638 ~ +1638 이며, 이는 Z3 명령어에 의하여 설정되고 실제 장착된 전류감지저항에 따라서 다른데, 2m0hm은 -20A ~ +20A 범위를 가리키며 5m0hm은 -8A ~ +8A 범위
	를 가리키고 10m0hm은 -4A ~ +4A 범위를 가리킴. 2012년 4월 10일 S/W 버젼부터는 Sw 명령어로 설정되는 제어전류 리밋값는 Z7 명령어로 설정되는 전류최대 치보다 크지 않게 제한됨. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
	return value



명령어	구분	설명
		제어시 사용하는 전류의 리밋을 0.01A 단위로 설정함.
	operation	1st word : motor1 제어전류리밋 : (comment 마지막부분 참조)
		2nd word : motor2 제어전류리밋 : (comment 마지막부분 참조)
	return value	Szdddd1,dddd2;
		2009년 7월 15일 S/W 버젼부터 적용되는 명령어임.
;Szdddd1,dddd2;		전류를 사용하여 전류나 위치나 속도제어를 하는 경우에 사용하는 전류의 최대치를 설정함.
(+#^)		Z3 명령어로 전류감지저항값이 정상적으로 설정되어 있어야함.
		IRMCK201을 사용한 경우는 Zc 명령어로 nominal current가 정상적으로 설정되어 있어야함.
;Sz?; (+#^)	comment	
	Collinett	KOMOTEK 380V모터의 경우 설정가능한 값의 범위는 전류감지저항으로 10m0hm을 사용하는 경우 -20A ~ +20A
		의 범위임. 그 외의 경우 설정가능한 값의 범위는 -1638 ~ +1638 이며, 이는 Z3 명령어에 의하여 설정
		되고 실제 장착된 전류감지저항에 따라서 다른데, 2m0hm은 -20A ~ +20A 범위이고 5m0hm은 -8A ~ +8A 범위
		이며 10m0hm은 -4A ~ +4A 범위임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
;Svdddd1,dddd2,d ddd3; (+#^%)	operation	MOSFET의 gate를 off 시키는 전압 강하량과 과전압 및 검출시간의 크기를 설정함. 1st word : 이 설정전압 이상의 전압강하가 발생하기를 3rd word 로 설정된 시간동안 연속되면 게이트를 off 시킴 [100 - 3000/22500(380V 모터구동기의 경우)] (800/10000 bit/0.01V) 2nd word : 이 설정전압 이상의 전압이 공급되기를 3rd word 로 설정된 시간동안 연속되면 게이트를 off 시킴 [800 - 9000/45000(380V 모터구동기의 경우)] (3400/4000(제일메디컬)/45000 bit/0.01V)
	return value	3rd word : 설정시간 [1 - 100] (10 ms) Svdddd1,dddd2,dddd3,dddd4,dddd5; (2011년 9월 11일 이전 S/W 버젼) Svdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7; (2011년 9월 11일 S/W 버젼부터) 4th word : 전원투입시의 filtering된 모터용 공급전압의 크기 (1 0.01V/bit) 5th word : 현재의 filtering된 모터용 공급전압의 크기 (1 0.01V/bit) 6th word : 현재의 slow filtering된 모터용 공급전압의 크기 (1 0.01V/bit) 7th word : 현재의 filtering된 모터용 또는 제어용(모터용 전원에서 제어용 전원을 내부에서 만드는 경우) 공급전압의 크기 (1 0.01V/bit)
;Sv?; (+#^%)	comment	이 설정값은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야함. 기본값은 800(8V)/10000(100V,380V 모터구동기의 경우), 3400(34V)/4000(40V,제일메디컬)/45000(450V,380V 모터구동기의 경우), 10(10ms) 임. DC motor에서는 AD7 입력전압을 사용하는데, 전압이상으로 gate가 off 되었을때, 복구하려면 PR 명령어를 사용하여야 함. BLDC motor에서도 적용되는데, 가변저항으로 설정된 전압을 사용하여 IRMCK201 자체에서 과전압 또는 저전압일때 gate를 자동 off 시키는 기능도 병행됨. (46.33V 이상이면 과전압, 13.56V 이하이면 저전압)이러한 과정을 통하여 과전류시에 발생하는 전압강하를 이용하여 DC와 BLDC의 구동회로를 보호할수있음. supply 전압의 허용 강하량은 통상 supply 전압의 1分에 해당하는 값을 설정하는 것이 좋음. supply 전압의 최대허용값은 통상 supply 전압의 1分에 해당하는 값을 설정하는 것이 좋음. 2nd word 값의 최고치가 2012년 2월 6일 S/W 버젼부터 6000에서 9000으로 상향조정되었음. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	전류제어 모드에서 전류명령값을 설정함.
		1st integer : motor1 전류명령값 [-1023/4095 ~ 1023/4095] (bit)
		2nd integer : motor2 전류명령값 [-1023/4095 ~ 1023/4095] (bit)
		SCiiii1,iiii2; or SCiiii1,iiii2,iiii3,iiii4;(RJM_VER5 STEP형 만 해당됨) or
		SCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;(RJM_VER6 3상STEP형 만 해당됨)
		1st integer : motor1 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil1)
	return value	2nd integer : motor2 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil2)
	Tetuiii value	3rd integer : coil3 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil3)
		4th integer : coil4 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil4)
		5rd integer : coil5 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER6 3상STEP형 만 해당, coil5)
;SCIIII1,IIII2;		6th integer : coil6 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER6 3상STEP형 만 해당, coil6)
(%)	comment	1A×25mΩ×11(앰프 증폭률) = 0.275V이므로 3.3V/3.0V는 3.3V÷0.275V = 12A/10.9A에 해당하며 85.25
		/375.375 bit/A. (DC ver1.1)
;SC?; (%)		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3V/3.0V는 3.3V÷0.25V = 13.2A/12A에 해당하며 77.5/341.25
		bit/A. (DC V1.2부터)
		1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3V/3.0V는 3.3V÷0.1V = 33A/30A에 해당하며 31/136.5 bit/A.
		(Piggyback-DC/RJM_VER8부터)
		BLDC 모터제어 chip인 IRMCK201에서 전류를 측정하고 스케일링하기 때문에, 전류감지저항값과 IR2175S 및
		전류스케일과 연관성이 있음. (Piggyback-BLDC형)
		1A×10mΩ×50(앰프 증폭률) = 0.5V이므로 3.3V/3.0V는 3.3V÷0.5V = 6.6/6A에 해당하며 155/682.5
		bit/A. (mini-DC)
		1A×20(10)mΩ×50(앰프 증폭률) = 1.00V/0.5V이므로 3.3V/5.0V는 3.3V÷1.00V = 3.3A/10A에 해당하며
		310/409.5 bit/A. (STEP형)
		다음페이지 참조!



명령어	구분	설명
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 5.0V는 5.0V÷0.25V = 20A에 해당하며 204.75 bit/A. (Sensor-BLDC) 따라서 DC V1.1 인 경우에 0 - 1023/4095의 값은 0 - 12/10.9A을 전류명령값을 의미함. 1[A]=85/375,
		2[A]=170/751, 3[A]=256/1126, 4[A]=341/1502, 5[A]=426/1877, 6[A]=512/2252, 7[A]=597/2628, 8[A]=682/3003, 9[A]=768/3378, 10[A]=853/3754, 11[A]=938 이며 10A 이상으로 설정하지 말것. DC V1.2부터는 1[A]=78/341, 2[A]=155/683, 3[A]=233/1024, 4[A]=310/1365, 5[A]=388/1706, 6[A]=465/2048, 7[A]=543/2389, 8[A]=620/2730, 9[A]=698/3071, 10[A]=775/3413, 11[A]=853/3754, 12[A]=930, 13[A]=1008 이며 11A 이상으로 설정하지 말것.
	comment	Piggyback-DC/RJM_VER8부터는 1[A]=31/137, 2[A]=62/273, 3[A]=93/410, 4[A]=124/546, 5[A]=155/683, 6[A]=186/819, 7[A]=217/956, 8[A]=248/1092, 9[A]=279/1229, 10[A]=310/1365, 11[A]=341/1502, 12[A]=372/1638, 13[A]=403/1775, 14[A]=434/1911, 15[A]=465, 16[A]=496 이며 14A 이상으로 설정하지 말 것.
		Piggyback-BLDC인 경우는 BLDC 모터파라메터 설정시에 주어야하는 정격전류값의 전류가 실제로 흐를 때, 측정된 전류값의 최종치가 4096이 되도록 하기 때문에, 전류명령값으로 4096/16=256를 설정하면 정격전류 를 명령하는 것이 됨.
		mini-DC 에서는 1[A]=155/683, 2[A]=310/1365 이며 2.5A 이상으로 설정하지 말것. STEP 에서는 0.25[A]=78/102, 0.5[A]=155/205, 0.75[A]=233/307, 1[A]=310/410, 1.25[A]=388/512, 1.5[A]=465/614, 1.75[A]=717, 2[A]=819, 2.25[A]=921, 2.5[A]=1024, 2.75[A]=1126 이며 1.5/2.75A 이상으로 설정하지 말것.
		Sensor-BLDC는 1[A]=205, 2[A]=410, 3[A]=614, 4[A]=819, 5[A]=1024, 6[A]=1229, 7[A]=1443, 8[A]=1638, 9[A]=1843 이며 9A 이상으로 설정하지 말것. 다음페이지 참조!



명령어	구분	설명
	comment	TMS320LF2406A RJM_VER5 STEP형은 motor1(coil1,coil2)/motor2(coil3,coil4) 또는
		motor1(coil1,coil2,coil3)/solenoid(coil4)의 조합으로 사용이 가능하며, RJM_VER6 3상STEP형은
		motor1(coil1,coil2,coil3)/motor2(coil4,coil5,coil6) 으로 코일이 사용됨.
		iiii1,iiii2,iiii3,iiii4,iiii5,iiii6은 부호가 있는 십진수임.



명령어	구분	설명
		전류제어 모드에서 전류명령값을 설정함.
		1st integer : motor1 전류명령값 [-1023/4095 ~ 1023/4095, 2009년 7월 15일 S/W 버젼부터는 comment 마
	operation	지막부분 참조] (bit)
		2nd integer : motor2 전류명령값 [-1023/4095 ~ 1023/4095, 2009년 7월 15일 S/W 버젼부터는 comment 마
		지막부분 참조] (bit)
		SCiiii1,iiii2; or SCiiii1,iiii2,iiii3,iiii4;(RJM_VER5 STEP형 만 해당됨) or
		SCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;(RJM_VER6 3상STEP형 만 해당됨)
		1st integer : motor1 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil1)
	return value	2nd integer : motor2 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil2)
	rotann varao	3rd integer : coil3 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil3)
;SCiiii1,iiii2;		4th integer : coil4 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER5/6 STEP형은 coil4)
(+#^)		5rd integer : coil5 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER6 3상STEP형 만 해당, coil5)
		6th integer : coil6 전류명령값 [-1023/4095 ~ 1023/4095] (bit) (RJM_VER6 3상STEP형 만 해당, coil6)
;SC?; (+#^)	comment	1A×25mΩ×11(앰프 증폭률) = 0.275V이므로 3.3V/3.0V는 3.3V÷0.275V = 12A/10.9A에 해당하며 85.25
		/375.375 bit/A. (DC ver1.1)
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3V/3.0V는 3.3V÷0.25V = 13.2A/12A에 해당하며 77.5/341.25
		bit/A. (DC V1.2부터)
		1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3V/3.0V는 3.3V÷0.1V = 33A/30A에 해당하며 31/136.5 bit/A.
		(Piggyback-DC/RJM_VER8부터)
		BLDC 모터제어 chip인 IRMCK201에서 전류를 측정하고 스케일링하기 때문에, 전류감지저항값과 IR2175S 및 전류스케일과 연관성이 있음. (Piggyback-BLDC형)
		전류스케일과 선편성이 있음. (Piggyback-bLb0성) 1A×10mΩ×50(앰프 증폭률) = 0.5V이므로 3.3V/3.0V는 3.3V÷0.5V = 6.6/6A에 해당하며 155/682.5
		bit/A. (mini-DC)
		다음페이지 참조!



명령어	구분	설명
		1A×20(10)mΩ×50(앰프 증폭률) = 1.00V/0.5V이므로 3.3V/5.0V는 3.3V÷1.00V = 3.3A/10A에 해당하며 310/409.5 bit/A. (STEP형) 1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 5.0V는 5.0V÷0.25V = 20A에 해당하며 204.75 bit/A. (Sensor-BLDC)
	comment	따라서 DC V1.1 인 경우에 0 - 1023/4095의 값은 0 - 12/10.9A을 전류명령값을 의미함. 1[A]=85/375, 2[A]=170/751, 3[A]=256/1126, 4[A]=341/1502, 5[A]=426/1877, 6[A]=512/2252, 7[A]=597/2628, 8[A]=682/3003, 9[A]=768/3378, 10[A]=853/3754, 11[A]=938 이며 10A 이상으로 설정하지 말것. DC V1.2부터는 1[A]=78/341, 2[A]=155/683, 3[A]=233/1024, 4[A]=310/1365, 5[A]=388/1706, 6[A]=465/2048, 7[A]=543/2389, 8[A]=620/2730, 9[A]=698/3071, 10[A]=775/3413, 11[A]=853/3754, 12[A]=930, 13[A]=1008 이며 11A 이상으로 설정하지 말것. Piggyback-DC/RJM_VER8부터는 1[A]=31/137, 2[A]=62/273, 3[A]=93/410, 4[A]=124/546, 5[A]=155/683, 6[A]=186/819, 7[A]=217/956, 8[A]=248/1092, 9[A]=279/1229, 10[A]=310/1365, 11[A]=341/1502, 12[A]=372/1638, 13[A]=403/1775, 14[A]=434/1911, 15[A]=465, 16[A]=496 이며 14A 이상으로 설정하지 말 것.
		Piggyback-BLDC인 경우는 BLDC 모터파라메터 설정시에 주어야하는 정격전류값의 전류가 실제로 흐를 때, 측정된 전류값의 최종치가 4096이 되도록 하기 때문에, 전류명령값으로 4096/16=256를 설정하면 정격전류를 명령하는 것이 됨. mini-DC 에서는 1[A]=155/683, 2[A]=310/1365 이며 2.5A 이상으로 설정하지 말것. STEP 에서는 0.25[A]=78/102, 0.5[A]=155/205, 0.75[A]=233/307, 1[A]=310/410, 1.25[A]=388/512, 1.5[A]=465/614, 1.75[A]=717, 2[A]=819, 2.25[A]=921, 2.5[A]=1024, 2.75[A]=1126 이며 1.5/2.75A 이상으로 설정하지 말것. 다음페이지 참조!



명령어	구분	설명
		Sensor-BLDC는 1[A]=205, 2[A]=410, 3[A]=614, 4[A]=819, 5[A]=1024, 6[A]=1229, 7[A]=1443, 8[A]=1638,
		9[A]=1 <mark>843</mark> 이며 9A 이상으로 설정하지 말것.
		TMS320LF2406A RJM_VER5 STEP형은 motor1(coil1,coil2)/motor2(coil3,coil4) 또는
		motor1(coil1,coil2,coil3)/solenoid(coil4)의 조합으로 사용이 가능하며, RJM_VER6 3상STEP형은
		motor1(coil1,coil2,coil3)/motor2(coil4,coil5,coil6) 으로 코일이 사용됨.
		iiii1,iiii2,iiii3,iiii4,iiii5,iiii6은 부호가 있는 십진수임.
		2008년 6월 이후의 모든 F2811/F28334 H/W 버젼에서는 다음과 같이 통일 되었음.
		2mΩ인 경우는 81.9 <mark>/81.9</mark> bit/A (RJM_VER3 : IRMCK201을 사용한 BLDC제어기, RJM_VER8 : 고출력 DC)
	comment	5mΩ인 경우는 204.75 <mark>/204.75</mark> bit/A (RJM_VER7 : IRMCK201을 사용하지 않은 BLDC제어기)
		10m요인 경우는 409.5/409.5 bit/A (RJM_VER4 : mini DC, RJM_VER5 : step motor)
		2009년 7월 15일 S/W 버젼부터 모든 F2811/F28334 H/W 버젼에서는 다음과 같은 적용을 받음.
		KOMOTEK 380V모터의 경우 설정가능한 값의 범위는 -8190 ~ +8190 이며, 전류감지저항으로 10m0hm을 사용하
		는 경우 -20A ~ +20A 의 범위에 해당함. IRMCK201을 사용한 경우 설정가능한 값의 범위는 -12288 ~
		+12288 이며, Zc 명령어로 설정되고 IRMCK201의 파라메터설정에 사용한 정격전류의 3배에 해당하는 범위
		임. 그 외의 경우 설정가능한 값의 범위는 -1638 ~ +1638 이며, 이는 Z3 명령어에 의하여 설정되고 실
		제 장착된 전류감지저항에 따라서 다른데, 2m0hm은 -20A ~ +20A 범위를 가리키며 5m0hm은 -8A ~ +8A 범위
		를 가리키고 10m0hm은 -4A ~ +4A 범위를 가리킴.



명령어	구분	설명
		전류치를 암페어로 환산하는 scale factor를 설정함.
	operation	1st word : motor1의 전류치 환산 factor [1 ~ 32767] (8192*0.01A/bit) (32768*0.01A/bit)
		2nd word : motor2의 전류치 환산 factor [1 ~ 32767] (8192*0.01A/bit) (32768*0.01A/bit)
	return value	Scdddd1,dddd2;
		1A×25mΩ×11(앰프 증폭률) = 0.275V이므로 3.3V/3.0V는 3.3V÷0.275V = 12A/10.9A에 해당하며
		8192*1200/1023 = 9609/8729. (DC ver1.1)
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 3.3V/3.0V는 3.3V÷0.25V = 13.2A/12A에 해당하며
		8192*1320/1023 = 10570/9602. (DC V1.2부터)
;Scdddd1,dddd2;		1A×2mΩ×50(앰프 증폭률) = 0.1V이므로 3.3V/3.0V는 3.3V÷0.1V = 33A/30A에 해당하며 8192★3300/1023 =
(%)	comment	26426/24006. (Piggyback-DC/RJM_VER8부터)
;Sc?; (%)		1A×5mΩ×50(앰프 증폭률)×3.3/5 = 0.165V이므로 3.3V/3.0V는 3.3V÷0.165V = 20A/18.18A에 해당하며
		8192*2000/1023 = 16016/14549. (Piggyback-BLDC)
		1A×10mΩ×50(앰프 증폭률) = 0.5V이므로 3.3V/3.0V는 3.3V÷0.5V = 6.6/6A에 해당하며 8192★660/1023 =
		5285/4801. (mini-DC)
		1A×20(10)mΩ×50(앰프 증폭률) = 1.00V/0.5V이므로 3.3V/5.0V는 3.3V÷1.00V = 3.3A/10A에 해당하며
		8192*330/1023 = 2643/8002 bit/A. (STEP형)
		1A×5mΩ×50(앰프 증폭률) = 0.25V이므로 5.0V는 5.0V÷0.25V = 20A에 해당하며 32768*2000/4095 =
		16004. (Sensor-BLDC)
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		전류제어 모드에서 0.01A 단위로 전류명령값을 설정함.
		1st integer : motor1 전류명령값 [-2000/2000 ~ 2000/2000, 2009년 7월 15일 S/W 버젼부터는 comment 마
	operation	지막부분 참조] (0.01A)
		2nd integer : motor2 전류명령값 [-2000/2000 ~ 2000/2000, 2009년 7월 15일 S/W 버젼부터는 comment 마
		지막부분 참조] (0.01A)
;Sciiii1,iiii2;	return value	Sciiii1,iiii2;
(+#^)		2009년 7월 15일 S/W 버젼부터 Z3 명령어로 전류감지저항값이 정상적으로 설정되어 있어야함.
(1#)		2009년 7월 15일 S/W 버젼부터 IRMCK201을 사용한 경우는 Zc 명령어로 nominal current가 정상적으로 설정
;Sc?; (+#^)		되어 있어야함.
	comment	2009년 7월 15일 S/W 버젼부터 모든 F2811/F28334 H/W 버젼에서는 다음과 같은 적용을 받음.
		KOMOTEK 380V모터의 경우 설정가능한 값의 범위는 전류감지저항으로 10m0hm을 사용하는 경우 -20A ~ +20A
		의 범위임. 그 외의 경우 설정가능한 값의 범위는 Z3 명령어에 의하여 설정되고 실제 장착된 전류감지
		저항에 따라서 다른데, 2m0hm은 -20A ~ +20A 범위이고 5m0hm은 -8A ~ +8A 범위를 이며 10m0hm은 -4A ~ +4A
		범위임.



명령어	구분	설명
;SViiii1,iiii2;		속도제어 모드에서 속도명령값을 설정함.
(%)	operation	1st integer : motor1 속도명령값 [-32000/60000 ~ 32000/60000] (0 rpm)
		2nd integer : motor2 속도명령값 [-32000/60000 ~ 32000/60000] (0 rpm)
;SVIIII1,IIII2;	return value	SViiii1,iiii2;, SVIIII1,IIII2; (F2811/F28334인 경우)
(F2811/F28334 <u>2</u>		SV 명령의 경우에 스테핑모터가 아닐때 속도오차가 크면 속도명령값을 재설정하고 적분오차를 0으로 초기
경우) (+#^)		화하는 작동을 내부에서 수행함. 따라서 같은 속도 명령을 주기적으로 주는 경우에도 속도변화가 발생
		할수 있음.
;SVr11111,11112;	comment	SVr 명령의 경우는 속도오차가 큰 경우에 속도명령값을 재설정하고 적분오차를 0으로 초기화하는 작동을
(F2811/F28334인	Comment	하지 않으므로, 속도 명령을 주기적으로 주는 경우에는 SVr 명령을 사용하여야함.
경우) (+#^)		SVr 명령의 경우는 2011년 9월 24일 S/W 버젼부터 사용가능한 명령임.
		F2811/F28334에서는 long integer를 입력할수있으며 속도명령값의 설정가능 범위가 [-60000 ~ 60000]
;SV?; (+#^%)		(rpm)임.
		속도제어 및 위치제어 모드에서 가속 및 감속율을 설정함.
		1st word : motor1 가속도 설정값 [1 ~ 60000] (9523 or 100 or 400 rpm/100ms or 1 rpm/ms)
	operation	2nd word : motor1 감속도 설정값 [1 ~ 60000] (9523 or 100 or 400 rpm/100ms or 1 rpm/ms)
		3rd word : motor2 가속도 설정값 [1 ~ 60000] (9523 or 100 or 400 rpm/100ms or 1 rpm/ms)
		4th word : motor2 감속도 설정값 [1 ~ 60000] (9523 or 100 or 400 rpm/100ms or 1 rpm/ms)
;Sadddd1,dddd2,d	return value	Sadddd1,dddd2,dddd3,dddd4;
ddd3, dddd4;		단위는 RJM_VER7/RJM_VER7에서는 rpm/ms, 나머지 H/W 버젼에서는 rpm/100ms임.
(+#^%)		기본값은 BLDC RJM_VER3 에서는 9523rpm/100ms, Wafer 회전제어기에서는 1rpm/100ms, 나머지 BLDC에서는 1
		rpm/ms, STEP RJM_VER5는 400 rpm/100ms, 그외의 경우는 100 rpm/100ms임.
;Sa?; (+#^%)		RJM_VER3 BLDC의 경우는 RMCK201의 가감속율을 9523 RPM/100ms 이상, RJM_VER7의 03번모드에서는 1000
	comment	RPM/ms이상으로 충분히 커야 위치제어에서의 불안정성을 없앨 수 있음. 이러한 이유로 BLDC의 경우에
		SM0303; 또는 SM0E0E; 명령을 수행할 때 일정값 이상이 되도록 강제로 바꿈.
		RJM_VER7에서 03/0E 작동모드에서는 Sa 명령에 의하여 가속 및 감속률을 설정하지는 않으나 위치제어 모드
		에서의 가감속에 필요한 펄스수는 계산함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		RJM_VER5/6 STEP형의 host direct electrical angle 제어 모드에서 구동전류의 진폭과 전류 OFFSET을 설정
		함.
		1st integer : motor1 구동전류 진폭 [1 ~ 2000] (200)
		2nd integer : motor2 구동전류 진폭 [1 ~ 2000] (200)
	operation	3rd integer : coil1 구동전류 offset [-800 ~ 800]
RJM_VER5/6 STEP	operation	4th integer : coil2 구동전류 offset [-800 ~ 800]
형인 경우 :		5th integer : coil3 구동전류 offset [-800 ~ 800]
		6th integer : coil4 구동전류 offset [-800 ~ 800]
;SHiiii1,iiii2,i		7th integer : coil5 구동전류 offset [-800 ~ 800]
1113,11114,1111		8th integer : coil6 구동전류 offset [-800 ~ 800]
5, i i i i i 6; (+#^%)	return value	SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (RJM_VER5 STEP형 만 해당됨) 또는
	Totalii varao	SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6,iiii7,iiii8; (RJM_VER6 3상STEP형 만 해당됨)
;SHiiii1,iiii2,i		RJM_VER5/6 STEP형에서 전류의 진폭과 전류 OFFSET은 310일때 1A에 해당함.
1113,11114,1111		RJM_VER5 STEP형을 2상스테핑 모터의 구동기로 사용하는 경우는 coil1/2 가 한개의 모터가 되고 coil3/4
5,11116,11117,11		가 또 한개의 모터가 되어 총 2개의 스테핑모터를 구동할 수 있음.
ii8; (+ #^%)		RJM_VER5 STEP형을 3상스테핑 모터의 구동기로 사용하는 경우는 coil1/2/3 가 해당코일이 되며, coil4는
		단독 코일로서 구동전류 offset에 해당하는 전류가 흐르도록 제어됨.
;SH?; (+#^%)	comment	RJM_VER6 3상STEP형인 경우는 첫번째 모터는 coil1/2/3 가 해당코일이 되며 두번째 모터는 coil4/5/6 이
		해당코일이 됨.
		F2811 <mark>/F28334</mark> 를 사용한 step motor 제어기(RJM_VER5)의 경우는 10mΩ으로 409.5 bit/A의 스케일을 갖음.
		step motor 제어기의 경우(RJM_VER5/RJM_VER6)는 1st interger 와 2nd interger는 ES 명령어로 EEPROM에
		저장될 수 있음.



명령어	구분	설명
RJM_VER5/6 STEP 형이 아닌 경우 : ;SHiiii1,iiii2; (+#^%)	operation	host direct electrical angle 제어 모드에서 구동전류 (RJM_VER7형의 3상/5상 STEP의 경우로서 작동모드 09번이 아닐때) 또는 구동전압의 진폭 (BLDC 구동 chip IRMCK201 경우) 또는 PWM duty (DC 모터를 제외한 기타의 경우)를 설정함. 1st integer: motor1의 구동전류 [1 ~ 2000] 또는 구동전압 진폭 [-2048 ~ 2047] 또는 PWM duty [-PWM_PERIOD ~ +PWM_PERIOD] 2nd integer: motor2의 구동전류 [1 ~ 2000] 또는 구동전압 진폭 [-2048 ~ 2047] 또는 PWM duty [-PWM_PERIOD ~ +PWM_PERIOD]
· CHO · (_ #/w)	return value	SHiiii1,iiii2;
;SH?; (+#^%)	comment	RJM_VER7형의 3상/5상 step motor 제어기의 경우는 1st interger 와 2nd interger는 ES 명령어로 EEPROM에 저장될 수 있음.
스테핑모터의 경우:	operation	스테핑모터에서 전류명령값의 최대치와 정지시의 전류치 비율을 설정함. dddd1 : CURRENT1_SATURATION [0 - 1000/2000] (300/400) dddd2 : CURRENT2_SATURATION [0 - 1000/2000] (300/400) dddd3 : CURRENT1_REDUCTION_RATIO [0 - 128] (8) dddd4 : CURRENT2_REDUCTION_RATIO [0 - 128] (8)
;Shdddd1,dddd2,d	return value	Shdddd1,dddd2,dddd3,dddd4;
ddd3,dddd4; (+#^%) ;Sh?; (+#^%)	comment	CURRENT1/2_REDUCTION_RATIO는 정지시의 전류감소를 설정하는 파라메터로서, 128이면 정지시전류는 구동시전류의 100%, 64이면 정지시전류는 구동시전류의 50%, 8이면 정지시전류는 구동시전류의 6.25% 임. 전류진폭이 CURRENT1/2_SATURATION 값을 초과하는 전류진폭을 SH 명령어로 설정하더라도 실제 전류제어과 정에서 전류의 진폭은 최대치로 제한됨. 따라서 진폭이 제한되는 경우에는 전류의 명령값이 sine파가 아니고 saturation 된 sine파가 됨. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.

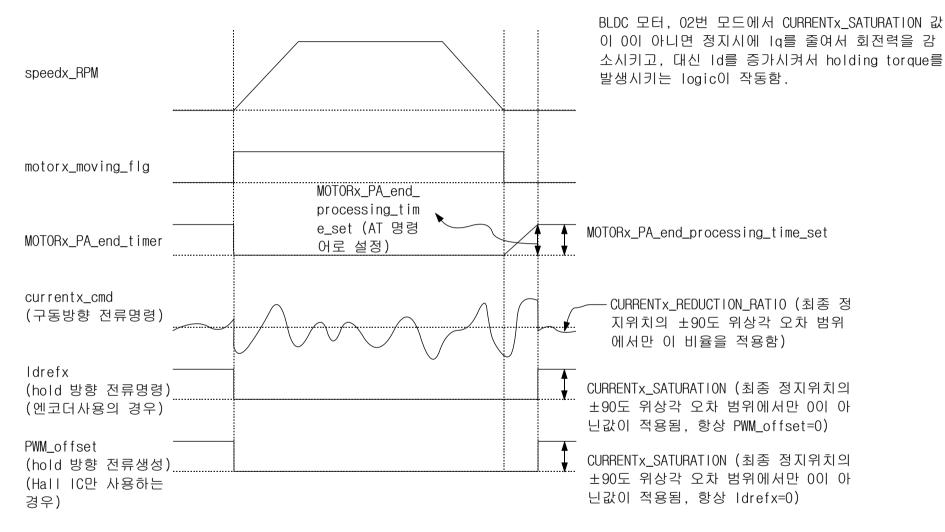


명령어	구분	설명
		BLDC모터에서 holding torque를 만드는 파라메터를 설정함.
		dddd1 : CURRENT1_SATURATION [0 - 2000] (0)
Wafer control의	operation	dddd2 : CURRENT2_SATURATION [0 - 2000] (0)
경우 (2012년 2월		dddd3 : CURRENT1_REDUCTION_RATIO [0 - 128] (32)
18일 S/W 버젼부		dddd4 : CURRENT2_REDUCTION_RATIO [0 - 32767] (100)
터 적용) :	return value	Shdddd1,dddd2,dddd3,dddd4;
48).		CURRENT1_SATURATION는 정지(position1_cmd==position1_set)후 MOTORx_PA_end_processing_time_set[ms] (AT
;Shdddd1,dddd2,d		명령어로 설정하는 시간값) 만큼의 시간이 경과한 이후에 적용하는 holding torque의 크기임.
ddd3, dddd4;		CURRENT2_SATURATION는 정지(position1_cmd==position1_set)후 CURRENT2_REDUCTION_RATIO[ms] 만큼의 시간
(+#^)		이 경과한 이후에 적용하는 holding torque의 크기임.
() ,	comment	CURRENT1_REDUCTION_RATIO는 정지(position1_cmd==position1_set)후 CURRENT2_REDUCTION_RATIO[ms] 만큼의
;Sh?; (+#^)		시간이 경과한 후에 회전방향으로의 회전력을 감소시키는 비율임. (예로 64이면 1/2, 32이면 1/4임)
,011: , (111)		WE 명령어 두번째 파라메터로 설정하는 값은 정지위치에서 필요한 holding 토크를 만드는 vector의 위상각
		보정량임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
Wafer control0		BLDC모터에서 holding torque를 만드는 파라메터를 설정함.
아닌 BLDC 모터의		dddd1 : CURRENT1_SATURATION [0 - 2000] (0)
경우 (2012년 2월	operation	dddd2 : CURRENT2_SATURATION [0 - 2000] (0)
18일 S/W 버젼부		dddd3 : CURRENT1_REDUCTION_RATIO [0 - 128] (32)
터 적용) :		dddd4 : CURRENT2_REDUCTION_RATIO [0 - 128] (32)
48).	return value	Shdddd1,dddd2,dddd3,dddd4;
;Shdddd1,dddd2,d		CURRENTx_SATURATION는 정지(position1_cmd==position1_set)후 MOTORx_PA_end_processing_time_set[ms] 만
ddd3,dddd4; (+#^)		큼의 시간이 경과한 후에 적용하는 holding torque의 크기임.
	comment	CURRENTx_REDUCTION_RATIO는 정지(position1_cmd==position1_set)후
		MOTORx_PA_end_processing_time_set[ms] 만큼의 시간이 경과한 후에 회전방향으로의 회전력을 감소시키는
;Sh?; (+#^)		비율임. (예로 64이면 1/2, 32이면 1/4임)
, σιι: , (ιπ /		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.

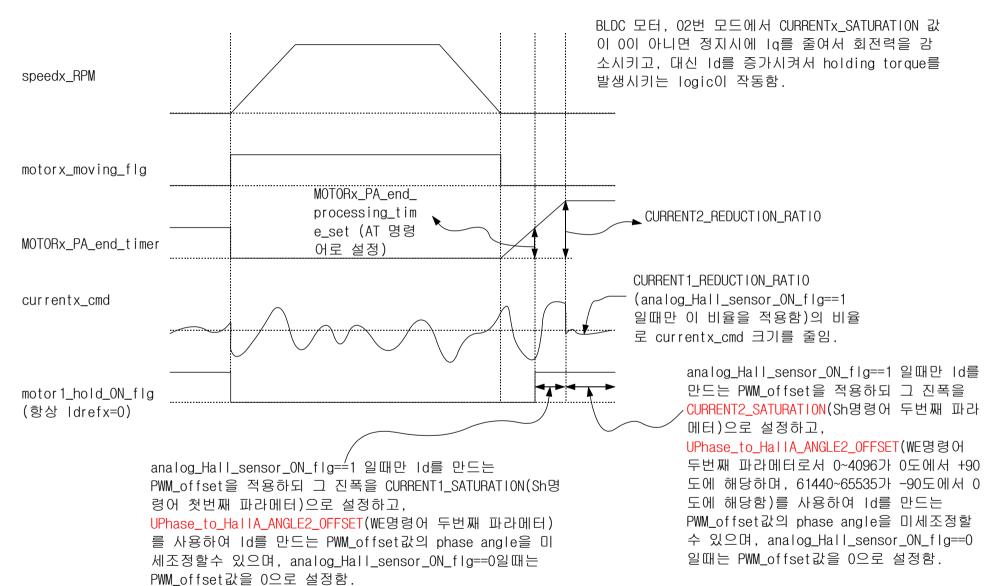


BLDC 모터에서 정지시 정지토크의 부여 (wafer control이 아닐때 - 2012년 2월 18일 S/W 버젼부터

적용)



240



ROBO

명령어	구분	설명
		좌표점을 정의함.
		dddd1 : point number [0 - 167]
	operation	IIII2 : motor1의 절대변위값 [position1_minus_limit ~ position1_plus_limit]
;SPdddd1,11112,1		IIII3 : motor2의 절대변위값 [position2_minus_limit ~ position2_plus_limit]
1113; (+#^%)	return value	SPdddd1, 11112, 11113;
		이렇게 설정된 좌표점은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야
;SP?dddd1;		함.
(+#^%)	comment	절대변위값은 상하한리밋범위 내에 있어야 유효함.
	Comment	2012년 12월 03일 S/W 버젼부터 좌표점의 개수가 192개에서 168개로 줄었음.
		자석가동자의 경우 사용할수 없음.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		위치제어의 feedforward 게인을 설정함.
	operation	1st word : motor1의 feedforward 게인 [0 - 10] (0)
;Sfdddd1,dddd2;		2nd word : motor2의 feedforward 게인 [0 - 10] (0)
(+#^)	return value	Sfdddd1,dddd2;
(π)		매 1ms마다 (positionx_set - position1_set_old)에 feedforward 게인을 곱한 값을 위치명령값으로 설정
;Sf?; (+#^)		함. 이렇게 함으로써 동적 상태에서 위치오차를 줄일수있음.
, oi : , (I#)	comment	기본값은 0이며 게인이 크면이 크면 진동이 발생할 우려가 큼.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F2811/F28334 에서만 가능한 명령임.



명령어	구분	설명		
		motor1 과 motor2 에 대하여 PM, PP, PQ, Pp, Pq command를 적용가능토록 설정함.		
		Oth bit : motor1_start_to_move_trajectory_flg 가 set (by PM) 가능하도록 설정함. (1)		
	operation	1st bit : motor2_start_to_move_trajectory_flg 가 set (by PM) 가능하도록 설정함. (1)		
		4th bit : motor1_start_to_move_to_position_flg 가 set (by PP, PQ, Pp, Pq) 가능하도록 설정함. (1)		
;SeHLHL; (+#^%)		5th bit : motor2_start_to_move_to_position_flg 가 set (by PP, PQ, Pp, Pq) 가능하도록 설정함. (1)		
, SCILLIL, (I# 70)	return value	SeHLHL;		
;Se?; (+#^%)	comment	이 설정값은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야함.		
, OC: , (1 m /0)		기본값은 0x0033임.		
		해당 bit가 세트되지 않으면 해당 모터는 PM 또는 PP, PQ, Pp, Pq 명령에 의한 작동이 되지 않음.		
		해당 bit가 세트되어 있어도 해당모터의 모드가 속도 모드이면 motorx_start_to_move_trajectory_flg,		
		motorx_start_to_move_to_position_flg 는 세트되지 않음.		
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
		위치제어에서 정지위치 근방에서 강성을 높이기 위하여 제어 게인을 높게 하는 비율을 설정함.		
;Sudddd1,dddd2;	operation	1st word : motor1의 게인 증가비율 [1 - 10] (1)		
(+#^)		2nd word : motor2의 게인 증가비율 [1 - 10] (1)		
	return value	Sudddd1,dddd2;		
;Su?; (+#^)	comment	설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
	Comment	F2811/F28334 에서만 가능한 명령임.		



명령어	구분	설명
;SQHLHLHLHL,HLHL HLHL,HLHLHL,HLH HL,HLHLHLHL	operation	모터1의 파라메터 리밋과 위치/속도/전류 제어파라메터를 설정함. (WQ 명령어로 속도 및 전류제어 파라메터를 설정하는 경우는 제외) 1st long word : MOTOR1_ERROR_SIZE_LIMIT (MOTOR1_PULSE_PER_REV * 10) 2nd long word : MOTOR1_INTEGRAL_START_ERROR_SIZE (MOTOR1_PULSE_PER_REV * 4) 3rd long word : MOTOR1_INTEGRAL_LIMIT (MOTOR1_PULSE_PER_REV * 100) 4th word : MOTOR1_INVERSE_OF_P_GAIN (MOTOR1_PULSE_PER_REV * 100) 5th long word : MOTOR1_INVERSE_OF_D_GAIN (MOTOR1_INTEGRAL_LIMIT*100L)/(50L*PWM_PERIOD) 6th word : MOTOR1_INVERSE_OF_D_GAIN (MOTOR1_INTEGRAL_LIMIT*100L)/(50L*PWM_PERIOD) 7th word : 속도제어시 MOTOR1_SPEED_MAX/60)*1000)/(250*PWM_PERIOD) 7th word : 속도제어시 MOTORV1_INVERSE_OF_P_GAIN 8th long word : 속도제어시 MOTORV1_INVERSE_OF_D_GAIN (F2811/F28334인 경우만) 10th word : 전류제어시 MOTORV1_INVERSE_OF_D_GAIN (F2811/F28334인 경우만) 10th word : 전류제어시 MOTORI1_INVERSE_OF_D_GAIN : 자동화용 H/W 버젼에서는 외부 analog 입력명령의 영점 offset 값. 11th long word : 전류제어시 MOTORI1_INVERSE_OF_I_GAIN : 자동화용 H/W 버젼에서는 외부 analog 입력명령의 개인 값. SQHLHILHLH_HLHLHLHLH_HLHLHLHLHLHLHLHLHLHLH
;SQ?; (+#^%) ;Sq?; (F2811/F28334인 경우만) (+#^)	(F2811/F28334인 경우만) Sq 명령일때는 게인값들은 50000/inverse_gain의 값들을 의미함. ((RJM_VER7_ENCODER RJM_VER7_ANALOG)==1)&&(TPC_FOCUSING==1) 인 경우에는 SQ 명령어로 설정하 막 2개의 파라마터가 각각 외부 입력전압으로부터 전류명령값으로 변환하는 offset과 게인의 의대 10/11th word는 자동화용 H/W 버젼의 제어기에서 (이 경우 전류게인은 WQ/WR 명령어로 설정함) 외인가되는 속도명령 또는 전류명령 전압값을 내부의 속도명령 및 전류명령값으로 변환하는데 사용메터임. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.	



명령어	구분	설명
		모터2의 파라메터 리밋과 위치/속도/전류 제어파라메터를 설정함. (WR 명령어로 속도 및 전류제어 파라메
		터를 설정하는 경우는 제외)
;SRHLHLHLHL,HLHL		1st long word : MOTOR2_ERROR_SIZE_LIMIT (MOTOR2_PULSE_PER_REV * 10)
HLHL,HLHLHLHL,HL		2nd long word: MOTOR2_INTEGRAL_START_ERROR_SIZE (MOTOR2_PULSE_PER_REV * 4)
HL,HLHLHL,HLH		3rd long word: MOTOR2_INTEGRAL_LIMIT (MOTOR2_PULSE_PER_REV * 100)
L,HLHL,HLHLHLHL,		4th word : MOTOR2_INVERSE_OF_P_GAIN (MOTOR2_PULSE_PER_REV*1000L)/(5L*PWM_PERIOD)
HLHL,HLHLHLHL;	operation	5th long word : MOTOR2_INVERSE_OF_I_GAIN (MOTOR2_INTEGRAL_LIMIT*100L)/(50L*PWM_PERIOD)
(%)	operation	6th word : MOTOR2_INVERSE_OF_D_GAIN
		(MOTOR2_PULSE_PER_REV*(MOTOR2_SPEED_MAX/60)*1000)/(250*PWM_PERIOD)
;SR또는		7th word : 속도제어시 MOTORv2_INVERSE_OF_P_GAIN
SpHLHLHLHL, HLHLH		8th long word : 속도제어시 MOTORv2_INVERSE_OF_I_GAIN
LHL,HLHLHLHL,HLH		9th word : 속도제어시 MOTORv2_INVERSE_OF_D_GAIN (F2811/F28334인 경우만)
L,HLHLHLHL,HLHL,		10th word : 전류제어시 MOTORi2_INVERSE_OF_P_GAIN
HLHL,HLHLHLHL,HL		11th long word : 전류제어시 MOTORi2_INVERSE_OF_I_GAIN
HL,HLHL,HLHLHLHL		SRHLHLHLH, HLHLHLH, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHLHLHL
; (F2811/F28334	return value	SR또는SpHLHLHLHL,HLHLHLL,HLHLHL,HLHL,HLHL,HLHL
인 경우) (+#^)		(F2811/F28334인 경우만)
		Sp 명령일때는 게인값들은 50000/inverse_gain의 값들을 의미함.
;SR?; (+#^%)		((RJM_VER7_ENCODER RJM_VER7_ANALOG)==1)&&(TPC_FOCUSING==1) 인 경우에는 SR 명령어로 설정하는 마지
		막 2개의 파라마터가 각각 외부 입력전압으로부터 전류명령값으로 변환하는 offset과 게인의 의미를 가짐.
;Sp?;	comment	(KOMOTEK_380V==1)&&(MOVING_MAGNET==1) 일때 SR 명령어로 설정하는 마지막에서 두번째 파라메터가 left
(F2811/F28334 <u>2</u>		analog Hall 센서에 대한 right analog Hall 센서의 위상차를 설정하는 값이며, 마지막 파라메터가 순위가
경우만) (+#^)		높은 제어기의 right analog Hall 센서에 대한 현 제어기의 left analog Hall 센서의 위상차를 설정하는
		값임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명		
;AQ또는 AqHLHL,HLHLHLHL, HLHL,HLHL,HLHLHL	operation	모터1의 위치/속도 제어파라메터를 설정함. (WQ 명령어로 속도제어 파라메터를 설정하는 경우는 제외) 1st word : MOTOR1_INVERSE_OF_P_GAIN (MOTOR1_PULSE_PER_REV*1000L)/(5L*PWM_PERIOD) 2nd long word : MOTOR1_INVERSE_OF_I_GAIN (MOTOR1_INTEGRAL_LIMIT*100L)/(50L*PWM_PERIOD) 3rd word : MOTOR1_INVERSE_OF_D_GAIN		
HL, HLHL; (+#^) ;AQ?; (+#^)		(MOTOR1_PULSE_PER_REV*(MOTOR1_SPEED_MAX/60)*1000)/(250*PWM_PERIOD) 4th word : 속도제어시 MOTORv1_INVERSE_OF_P_GAIN 5th long word : 속도제어시 MOTORv1_INVERSE_OF_I_GAIN 6th word : 속도제어시 MOTORv1_INVERSE_OF_D_GAIN		
;Aq?; (+ #^)	return value AQ또는AqHLHL,HLHLHLHLHL,HLHL,HLHL,HLHL,HLHL;			
,Aq?, (+#**)	comment	Aq 명령일때는 게인값들은 50000/inverse_gain의 값들을 의미함. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
;AR또는 ArHLHL,HLHLHLHL, HLHL,HLHL,HLHLHL HL,HLHL; (+#^) ;AR?; (+#^)	operation	모터2의 위치/속도 제어파라메터를 설정함. (WR 명령어로 속도제어 파라메터를 설정하는 경우는 제외) 1st word: MOTOR2_INVERSE_OF_P_GAIN (MOTOR2_PULSE_PER_REV*1000L)/(5L*PWM_PERIOD) 2nd long word: MOTOR2_INVERSE_OF_I_GAIN (MOTOR2_INTEGRAL_LIMIT*100L)/(50L*PWM_PERIOD) 3rd word: MOTOR2_INVERSE_OF_D_GAIN (MOTOR2_PULSE_PER_REV*(MOTOR2_SPEED_MAX/60)*1000)/(250*PWM_PERIOD) 4th word: 속도제어시 MOTORv2_INVERSE_OF_P_GAIN 5th long word: 속도제어시 MOTORv2_INVERSE_OF_I_GAIN 6th word: 속도제어시 MOTORv2_INVERSE_OF_D_GAIN		
;Ar?; (+#^)	return value	AR또는ArHLHL,HLHLHL,HLHL,HLHLHLHLHL,HLHL; Ar 명령일때는 게인값들은 50000/inverse_gain의 값들을 의미함. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		



명령어	구분	설명
;AS또는		모터1,2의 전류 제어파라메터를 설정함. (WQ, WR 명령어로 전류제어 파라메터를 설정하는 경우는 제외)
ASHLHL, HLHLHLHL,		1st word : 전류제어시 MOTORi1_INVERSE_OF_P_GAIN
HLHL, HLHLHLHL;	operation	2nd long word : 전류제어시 MOTORi1_INVERSE_OF_I_GAIN
(+#^)		3rd word : 전류제어시 MOTORi2_INVERSE_OF_P_GAIN
		4th long word : 전류제어시 MOTORi2_INVERSE_OF_I_GAIN
;AS?; (+#^)	return value AS또는AsHLHL,HLHLHLHLHLHLHLHL;	
	comment	As 명령일때는 게인값들은 50000/inverse_gain의 값들을 의미함.
;As?; (+#^)	COMMETT	설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.

모터 제어파라메터의 설정과 저장

- ♣ 모터의 제어파라메터를 설정하고 저장하려면
 - 1) SQ, Sq, SR, Sp, WQ, WR 명령어를 사용하여 제어게인을 바꾸어줌. (이때 SQ?;, Sq?; SR?;, Sp?; WQ?; WR?; 를 사용하여 현재값을 읽어내
 - 어 조금씩 바꾸는 형식으로 실시할 것)
 - 2) ESOAOO,0100; 명령 또는 EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
 - 3) 전원을 껏다가 켬.
 - 4) SQ?;, Sq?; SR?;, Sp?; WQ?; WR?; 등을 사용하여 바뀐 파라메터가 반영되었는지를 확인함.

제어기 H/W 버젼	모터 종류	위치제어 PID 게인 (01번, 02번, 03번 위치제어시에 사용)	속도제어 PID 게인 (05번 속도제어시에 사용)	속도제어 PID 게인 (06번 속도제어시에 사용)	전류제어 PI 게인 (07/08번 전류제어시에 사 용)	แฉ
RJM_VER 1,2,3(2406a)				SQ/Sq 또는 SR/Sp 명 령어의 7/8/9번째 파 라메터	SQ/Sq 또는 SR/Sp 명령어 의 마지막 2개 파라메터 (1KHz 제어주기)	
RJM_VER 3(2811),4,8	DC		SQ/Sq 또는 SR/Sp 명 령어의 7/8/9번째 파 라메터		SQ/Sq 또는 SR/Sp 명령어 의 마지막 2개 파라메터 (02번 위치제어시에도 사 용) (05번 속도제어시에도 사용)	
RJM_VER 3	- BLDC			WQ 또는 WR 명령어의 앞쪽 2개 파라메터 (03번 위치제어시에도 사용)	WQ 또는 WR 명령어의 마지 막 2개 파라메터 (02번 위	
RJM_VER 7				WQ 또는 WR 명령어의 앞쪽 3개 파라메터 (03번 위치제어시에도 사용)	치제어시에도 사용) (05번 속도제어시에도 사용)	
RJM_VER 5,6	STEP				SQ/Sq 또는 SR/Sp 명령어 의 마지막 2개 파라메터	



명령어	구분	설명
		PA 또는 SV 명령에 의한 가감속시에 전류명령값의 feedforward 값을 계산하는 scale factor를 설정함.
		1st word : motor1의 가속시 전류명령값 offset 계산용 scale factor [0 - 10000] (0)
	operation	2nd word : motor1의 감속시 전류명령값 offset 계산용 scale factor [0 - 10000] (0)
		3rd word : motor2의 가속시 전류명령값 offset 계산용 scale factor [0 - 10000] (0)
;Aldddd1,dddd2,d		4th word : motor2의 감속시 전류명령값 offset 계산용 scale factor [0 - 10000] (0)
ddd3,dddd4;	return value	Aldddd1,dddd2,dddd3,dddd4;
(+#^)		(motorx의 전류명령값 offset 계산용 scale factor) * 가감속도값 / 4096 의 크기를 가감속시의 전류
		feedforward 값으로 설정함. Qx114;Qx115; 를 사용하여 가감속시에 4번째값(전류명령 offset)이 3번째
;A1?; (+#^)		값(전류값)의 평균값이 되도록 AI 명령에 의한 파라메터를 수정함. Qx113; 으로 가감속을 확인할수 있
	comment	음. 02/03번 위치제어나 05번 속도제어에서만 사용이 가능함.
		2011년 11월 22일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F2811/F28334 에서만 가능한 명령임.
		PA 명령에 의한 가속시에 사다리꼴 모양의 가속 프로파일을 만들기 위한 가속 변화율값을 설정함.
	operation	1st word : motor1의 가속시 1ms당 가속변화량 [1 - 65535] (65535)
		2nd word : motor2의 가속시 1ms당 가속변화량 [1 - 65535] (65535)
;Aidddd1,dddd2;	return value	Aidddd1,dddd2;
(+#^)		단위크기는 (1000 * 1000 * 60) / (SPEED_SCALE_FACTOR * MOTOR1_PULSE_PER_REV) RPM/sec에 해당함.
		1보다 작은값의 설정이 불가하며, 큰값을 주는 경우는 사각모양의 가속 프로파일에 해당하는 가속을 실시
;Ai?; (+#^)	comment	함. Qx113; 의 네번째값으로 가감속 변화를 확인할수 있음.
		2012년 1월 30일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F2811 <mark>/F28334</mark> 에서만 가능한 명령임.



명령어	구분	설명		
	operation	모터1에 적용되는 Notch filter의 중심주파수와 Q factor를 설정함.		
		1st word : motor1의 Notch filter의 중심주파수 [0 - 5000] (500 0.1Hz)		
;ANdddd1,dddd2;		2nd word : motor1의 Notch filter의 Q factor [0 - 100] (0 0.1)		
(+#^)	return value	ANdddd1,dddd2;		
		전류명령값을 계산하고나서 MOTOR1_NOTCH_FILTER1_Q!=0 이면 notch filter를 적용하여 전류명령값을 최종		
;AN?; (+#^)	comment	결정함. 2011년 11월 24일 S/W 버젼부터 사용가능한 명령임.		
	Collilliont	설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
		F2811/F28334 에서만 가능한 명령임.		
		모터2에 적용되는 Notch filter의 중심주파수와 Q factor를 설정함.		
	operation	1st word : motor2의 Notch filter의 중심주파수 [0 - 5000] (500 0.1Hz)		
;Andddd1,dddd2;		2nd word : motor2의 Notch filter의 Q factor [0 - 100] (0 0.1)		
(+#^)	return value	Andddd1,dddd2;		
		전류명령값을 계산하고나서 MOTOR2_NOTCH_FILTER1_Q!=0 이면 notch filter를 적용하여 전류명령값을 최종		
;An?; (+#^)	comment	결정함. 2011년 11월 24일 S/W 버젼부터 사용가능한 명령임.		
	Commont	설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
		F2811/F28334 에서만 가능한 명령임.		
		28~31번 butterworth filterA의 시정수를 설정함.		
		1st word : 28번 butterworth filterA의 시정수. [0 - 4096] (0 ms)		
	operation	2nd word : 29번 butterworth filterA의 시정수. [0 - 4096] (0 ms)		
;ALdddd1,dddd2,d		3rd word : 30번 butterworth filterA의 시정수. [0 - 4096] (0 ms)		
ddd3,dddd4; (#)		4th word : 31번 butterworth filterA의 시정수. [0 - 4096] (0 ms)		
uuuo, uuuu+, (#)	return value	ALdddd1,dddd2,dddd3,dddd4;		
;AL?; (#)		butterworth filterA는 2차 필터로서 0-31 까지 총 32개가 있으며 1ms 마다 filter가 동작함.		
	comment	시정수 설정값을 2로 나눈 값이 ms 단위의 실제 시정수에 해당함. 0인 경우는 필터링하지 않음.		
		28/30번 butterworth filterA는 02번 제어모드에서 모터1/2의 전류명령값을 생성할때 low pass filter로		
		사용함. 2012년 8월 14일 S/W 버젼부터 사용가능한 명령임.		
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		



명령어	구분	설명		
	operation	PA 명령의 완료 시점에서 추가로 실행하는 기능을 위한 시간파라메터를 설정함.		
		1st word : PA 명령에 의한 위치이동 종료후 게인을 유지하는 기간 [1 - 30000] (100 ms)		
		(RJM_VER7_ENCODER==1)&&(ELECTRONIC_POWER_STEER==1) 일때는 가감속 완료후 오차의 적분값을 강제로 0이		
		되게하기 까지의 경과시간임.		
;ATdddd1; (+#^)	return value	ATdddd1;		
		이동중의 PID 게인이 PA 명령에 의한 위치이동 종료후에도 일정시간 유지되며, 설정시간이 경과된 이후에		
;AT?; (+ # ^)		는 비례, 적분 및 미분게인을 ABA55A 명령으로 설정된 비율로 낮추거나 높임. 새로운 PA 명령어로 이동		
	comment	을 시작하면 원래의 게인으로 자동 전환됨. 이렇게 함으로서 정지시의 흔들림이나 강성을 조절할수 있		
	Commert	음. 2011년 12월 10일 S/W 버젼부터 사용가능한 명령임.		
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.		
		F2811/F28334 에서만 가능한 명령임.		
		위치제어에서 정지하고 있을때, jitter와 소음을 줄이거나 또는 강성을 높이기 위하여, 위치제어 게인을		
	operation	낮추거나 높이기 위한 파라메터를 설정함. 2012년 1월 11일 S/W 버젼부터 사용가능한 명령임.		
		1st word : 정지시 게인을 낮추거나 높이기 위한 파라메터값 [0 - 800] (0)		
	return value	ABdddd1;		
		이동중의 PID 게인이 PA 명령에 의한 위치이동 종료후에도 일정시간 유지되며, 설정시간이 경과된 이후에		
		는 비례, 적분 및 미분게인을 ABA55A 명령으로 설정된 비율로 낮추거나 높여짐. 101~200, 201~300,		
;ABA55A,dddd1;		301~400, 401~500, 501~600, 601~700, 701~800 의 값은 1~100과 같게 취급되며, 0이면 정지시 게인의 조정		
(+#^)		이 없고, 1/10/100이면 0.1/1/10 배의 게인이 되도록 하는 것임. 201~300, 301~400, 601~700, 701~800		
, ,		인 경우는 정지시 추가로 전류게인을 반으로 줄이며, 401~500, 501~600, 601~700, 701~800인 경우는 정지		
;AB?; (+ # ^)		시 추가로 미분게인을 반으로 줄임. 새로운 PA 명령어로 이동을 시작하면 원래의 게인으로 자동 전환		
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	comment	됨. 이렇게 함으로서 정지시의 흔들림이나 강성을 조절할수 있음.		
		(KOMOTEK_380V & AUTOMATION)==1 이거나 ((RJM_VER7_ENCODER RJM_VER7_ANALOG) & AUTOMATION)==1 일때는		
		ABA55A 명령에 의한 값이 101~200, 301~400, 501~600, 701~800 범위에 있을때는 정지시에 01번 제어모드로		
		전환하며 (이때 KOMOTEK_380V==1이면 공급전압에 따라서 ABA55A 명령에 의한 값을 줄임), 이동중에는 02번		
		보드로 이동함. 01번 제어모드일때의 PWM 크기 최대치는 게인과 같은 비율로 크거나 작게 되며 단 25%		
		이하이거나 75%이상의 duty로 설정되지는 않음.		
NUUU		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. F2811/F28334 에서만 가능한 명령임.		

명령어	구분	설명		
		사용가능한 최대크기의 PWM duty 비를 설정함.		
;AwA55A,dddd1,dd	operation	1st word : 모터1의 사용가능한 최대크기의 PWM duty 비 [10 - 100] (100 1%)		
dd2; (+#^)		2nd word : 모터2의 사용가능한 최대크기의 PWM duty 비 [10 - 100] (100 1%)		
	return value	Awdddd1,dddd2;		
;Aw?; (+#^)	comment	2012년 7월 10일 S/W 버젼부터 사용가능한 명령임.		
	COMMETT	설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. F2811/F28334 에서만 가능한 명령임.		
		MOTOR 구동 Half-bridge의 dead-time을 보상하되, Z6 명령어로 설정하는 dead-time에 추가하여 보상하려는		
	operation	보상량을 설정함.		
	operation	1st word : 모터1의 dead-time을 추가보상하는 보상량 [-α ~ PWM_PERIOD/8] (-α 8.33ns)		
		2nd word : 모터2의 dead-time을 추가보상하는 보상량 [-α ~ PWM_PERIOD/8] (-α 8.33ns)		
	return value	ADiiii1,iiii2;		
;ADA55A, 1,		추가보상하는 보상량이 0인 경우는 Z6 명령어로 설정하는 dead-time 만큼의 보상을 하는 경우에 해당함.		
ii2; (+#^)		추가보상하는 보상량의 하한값 $-\alpha$ 는 $Z6$ 명령어로 설정하는 dead-time의 음의 값임. 이 경우에는 실질		
		적인 dead-time 보상은 0임.		
;AD?; (+#^)		Z6 명령어로 설정하는 dead-time 값이 0인 경우(모터드라이브 IC 또는 gate drive IC에서 dead-time을 부		
	comment	여하는 경우)는 추가보상하는 보상량의 범위가 -PWM_PERIOD/16 ~ PWM_PERIOD/8 이며 기본값은 0임.		
		2012년 8월 14일 S/W 버젼부터 사용가능한 명령이나 컴파일시에 option을 선택한 경우에 한하며, 통상의		
		경우에는 사용할수 없음. dead-time 보상에 CPU time을 많이 소모하므로 PWM 주파수를 60KHz 이내에서		
		사용하여야함.		
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. F2811/F28334 에서만 가능한 명령임.		



명령어	구분	설명
;Acdddd1,dddd2;	operation	엔코더펄스 카운트 방식으로 속도를 계산할 때 moving average 구간의 크기를 설정함.
		1st word : 모터1의 속도를 계산할 때 moving average 구간의 크기 [1 ~ 10] (10 ms)
(+)		2nd word : 모터2의 속도를 계산할 때 moving average 구간의 크기 [1 ~ 10] (10 ms)
	return value	Acdddd1,dddd2;
;Ac?; (+)		엔코더펄스를 카운트하여 속도를 계산하는데, moving average를 이용하여 속도계산값을 필터링함. 첫번
,AU:, (T)	comment	째와 두번째 파라메터값을 작은 값으로 지정할수록 속도계산값의 분해능은 나빠지나 시간지연은 감소함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. F28334/5 에서만 가능한 명령임.
		엔코더펄스 카운트 방식으로 속도를 계산할 때 moving average 구간의 크기와, 엔코더펄스의 주기를 사용
		하여 속도를 계산할때 주기계산에 사용되는 엔코더펄스개수의 2에 대한 지수를 설정함.
	operation	1st word : 모터1의 속도를 계산할 때 moving average 구간의 크기 [1 ~ 10] (10 ms)
	орегаттоп	2nd word : 모터2의 속도를 계산할 때 moving average 구간의 크기 [1 ~ 10] (10 ms)
		3rd word : 모터1의 주기계산에 사용되는 엔코더펄스 개수의 2에 대한 지수 [2 ~ 11] (2 multiple_of_2)
		4th word : 모터2의 주기계산에 사용되는 엔코더펄스 개수의 2에 대한 지수 [2 ~ 11] (2 multiple_of_2)
;Acdddd1,dddd2,d	return value	Acdddd1,dddd2,dddd3,dddd4;
ddd3,dddd4; (#)		엔코더펄스를 카운트하여 속도를 계산하는데, moving average를 이용하여 속도계산값을 필터링함. 첫번
dudo, dudu+, (#)		째와 두번째 파라메터값을 작은 값으로 지정할수록 속도계산값의 분해능은 나빠지나 시간지연은 감소함.
;Ac?; (#)		세번째와 네번째 파라메터값에 있어서 엔코더펄스 개수의 2에 대한 지수가 2인 경우는 체배하지 않은 엔코
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		더 펄스 한주기를 사용하는 것에 해당하고, 3인 경우는 체배하지 않은 엔코더 펄스 두주기를 사용하는 것
	comment	에 해당하고, 4인 경우는 체배하지 않은 엔코더 펄스 네주기를 사용하는 것에 해당함. 따라서 11인 경
	comment	우는 512주기를 사용하는 것에 해당함. 세번째와 네번째 파라메터값을 작은 값으로 지정할수록 속도계
		산값의 분해능은 나빠지나 시간지연은 감소함.
		세번째와 네번째 파라메터값이 최대속도에서 1ms당 full wave pulse 수가 1~15개 범위이면 2, 16~31개 범
		위이면 3, 32~63개 범위이면 4, 64~127개 범위이면 5, 128~255개 범위이면 6이 되는 것이 기본값임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음. F28334/5 에서만 가능한 명령임.



명령어	구분	설명
;AXA55A,dddd1,dddd2,dddd3,dddd6,dddd7,dddd8,dddd9,dddd11,dddd12,dddd13,dddd14	operation	모터1의 상위 속도에서 제어 게인을 높이거나 낮추는 파라메터를 설정함. 1st word : 모터1의 상위 속도값 [~ 60000] (- RPM) 2nd/3rd/4th word : 상위 속도에서의 P/I/D 위치게인 조정값 [50 ~ 20000] (1000 bit) 5th/6th/7th word : 01/02/04/05/06/0E/0F번 제어모드에서 상위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 8th/9th word : DC 또는 STEP 모터에서 상위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit) 10th/11th/12th word : 03/0A번 제어모드에서 상위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 13th/14th word : BLDC 모터에서 상위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit)
; (+#^)	return value	AXdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14;
;AX?; (+#^)	comment	2nd ~ 14th word의 값은 1000일때 게인을 조정하지 않는 것이며, 2000 이면 2배를 크게 하며, 500 이면 1/2배로 줄이는 것임. 2012년 12월 20일 S/W 버젼부터 문제 없이 적용됨. 1st word의 값을 65535로 설정하면 상위속도구간에서의 게인 조정이 실시되지 않음. 설정된 값은 AzA55A; 명령어로 EEPROM에 저장될 수 있음.
;AxA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14	operation	모터1의 하위 속도에서 제어 게인을 높이거나 낮추는 파라메터를 설정함. 1st word : 모터1의 하위 속도값 [~ 60000] (- ms) 2nd/3rd/4th word : 하위 속도에서의 P/I/D 위치게인 조정값 [50 ~ 20000] (1000 bit) 5th/6th/7th word : 01/02/04/05/06/0E/0F번 제어모드에서 하위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 8th/9th word : DC 또는 STEP 모터에서 하위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit) 10th/11th/12th word : 03/0A번 제어모드에서 하위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 13th/14th word : BLDC 모터에서 하위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit)
; (+#^)	return value	Axdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14;
;Ax?; (+#^)	comment	2nd ~ 14th word의 값은 1000일때 게인을 조정하지 않는 것이며, 2000 이면 2배를 크게 하며, 500 이면 1/2배로 줄이는 것임. 2012년 12월 20일 S/W 버젼부터 문제 없이 적용됨. 1st word의 값을 65535로 설정하면 하위속도구간에서의 게인 조정이 실시되지 않음. 설정된 값은 AzA55A; 명령어로 EEPROM에 저장될 수 있음.
CUBE		253

명령어	구분	설명
;AYA55A,dddd1,dddd2,dddd3,dddd6,dddd7,dddd8,dddd9,dddd11,dddd12,dddd13,dddd14	operation	모터2의 상위 속도에서 제어 게인을 높이거나 낮추는 파라메터를 설정함. 1st word : 모터2의 상위 속도값 [~ 60000] (- ms) 2nd/3rd/4th word : 상위 속도에서의 P/I/D 위치게인 조정값 [50 ~ 20000] (1000 bit) 5th/6th/7th word : 01/02/04/05/06/0E/0F번 제어모드에서 상위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 8th/9th word : DC 또는 STEP 모터에서 상위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit) 10th/11th/12th word : 03/0A번 제어모드에서 상위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 13th/14th word : BLDC 모터에서 상위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit)
; (+#^)	return value	AYdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14;
;AY?; (+#^)	comment	2nd ~ 14th word의 값은 1000일때 게인을 조정하지 않는 것이며, 2000 이면 2배를 크게 하며, 500 이면 1/2배로 줄이는 것임. 2012년 12월 20일 S/W 버젼부터 문제 없이 적용됨. 1st word의 값을 65535로 설정하면 상위속도구간에서의 게인 조정이 실시되지 않음. 설정된 값은 AzA55A; 명령어로 EEPROM에 저장될 수 있음.
;AyA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14	operation	모터2의 하위 속도에서 제어 게인을 높이거나 낮추는 파라메터를 설정함. 1st word : 모터2의 하위 속도값 [~ 60000] (- RPM) 2nd/3rd/4th word : 하위 속도에서의 P/I/D 위치게인 조정값 [50 ~ 20000] (1000 bit) 5th/6th/7th word : 01/02/04/05/06/0E/0F번 제어모드에서 하위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 8th/9th word : DC 또는 STEP 모터에서 하위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit) 10th/11th/12th word : 03/0A번 제어모드에서 하위 속도에서의 P/I/D 속도게인 조정값 [50 ~ 20000] (1000 bit) 13th/14th word : BLDC 모터에서 하위 속도에서의 P/I 전류게인 조정값 [50 ~ 20000] (1000 bit)
; (+#^)	return value	Aydddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9,dddd10,dddd11,dddd12,dddd13,dddd14;
;Ay?; (+#^)	comment	2nd ~ 14th word의 값은 1000일때 게인을 조정하지 않는 것이며, 2000 이면 2배를 크게 하며, 500 이면 1/2배로 줄이는 것임. 2012년 12월 20일 S/W 버젼부터 문제 없이 적용됨. 1st word의 값을 65535로 설정하면 하위속도구간에서의 게인 조정이 실시되지 않음. 설정된 값은 AzA55A; 명령어로 EEPROM에 저장될 수 있음.
CUBE		254

명령어	구분	설명
;AZA55A,dddd1,dd dd2; (+#^) ;AZ?; (+#^)	operation	SQ/SR/Sq/Sr/AQ/AR/Aq/Ar/WQ/WR 등의 명령으로 설정하는 게인이 적용되는 중간 속도값을 설정함. 1st word : 모터1의 중간 속도값 [~ 60000] (- RPM) 2nd word : 모터2의 중간 속도값 [~ 60000] (- RPM)
	return value	AZdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; 1st word : 모터1의 중간 속도값 [~ 60000] (- RPM) 2nd word : 모터2의 중간 속도값 [~ 60000] (- RPM) 3rd word : 모터1의 하위 속도값 [~ 60000] (- RPM) 4th word : 모터1의 상위 속도값 [~ 60000] (- RPM) 5rd word : 모터2의 하위 속도값 [~ 60000] (- RPM) 6th word : 모터2의 상위 속도값 [~ 60000] (- RPM)
	comment	1st word의 값을 65535로 설정하면 모터1에 대하여 게인 조정이 실시되지 않음. 2nd word의 값을 65535로 설정하면 모터2에 대하여 게인 조정이 실시되지 않음. 중간속도값은 상위속도보다 작아야 하고, 하위속도보다 커야함. 2012년 12월 30일 S/W 버젼부터 문제 없이 적용됨. 설정된 값은 AzA55A; 명령어로 EEPROM에 저장될 수 있음.
	operation	AX/Ax/AY/Ay/AZ 명령에 의한 설정값을 EEPROM에 저장함.
;AzA55A; (+#^)	return value	AzA55A;
	comment	



명령어	구분	설명
;Stdddd1,dddd2,d ddd3; (+#^%) ;St?; (+#^%)	operation return value comment	PP/Pp/PQ/Pq/p 명령어의 update 주기와 PP/Pp/PQ/Pq/p 명령어 미수신의 연속허용횟수를 설정함. 1st word : PP/Pp/p 명령어의 update 주기 [2-1000, 2002-3000] (10 ms) 2nd word : PQ/Pq 명령어의 update 주기 [2-1000, 2002-3000] (30 ms) 3rd word : PP/Pp/PQ/Pq/p 명령어 미수신의 연속허용횟수 [2-10] (3 회) Stdddd1,ddd2,dddd3; 기본값은 10ms, 30ms, 3회 이며 기본단위는 1ms 임. 2002-3000 범위의 값을 설정하고 EEPROM에 저장한 다음 재부팅하면 메인루프의 실행시간이 1.25ms가 되며, 이에 따라서 update 주기는 2.5ms부터 1250ms 까지 1.25ms 단위로 설정됨. 2002-3000 범위의 값을 설정하는 경우, 1st word 또는 2nd word 둘중에 한개만 2002-3000 범위일지라도 EEPROM에 저장하고 재부팅한다면 1.25ms가 기본단위가 됨. 1st word 와 2nd word 둘 모두가 2-1000 범위일 때, EEPROM에 저장하고 재부팅한다면 1ms가 기본단위가됨.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명	
		Pp/Pq/p 명령어에 대한 return값의 주소를 설정함.	
		1st word : Pp/Pq/p 명령어에 대하여 return 받을 첫번째 변수의 주소값. [범위는 comment 참조]	
	operation	2nd word : Pp/Pq/p 명령어에 대하여 return 받을 두번째 변수의 주소값. [범위는 comment 참조]	
		3rd word : p 명령어에 대하여 return 받을 세번째 변수의 주소값. [범위는 comment 참조]	
		4th word : p 명령어에 대하여 return 받을 네번째 변수의 주소값. [범위는 comment 참조]	
	return value	SrHLHL, HLHL; or SrHLHL, HLHL, HLHL;	
		SrHLHL,HLHL; 명령을 보내면 p 명령에 대한 return 값은 1st/2nd word로 지정된 2개 주소내의 값이 되	1며,
		SrHLHL,HLHL,HLHL,HLHL; 명령을 보내면 p 명령에 대한 return 값은 1st/2nd/3rd/4th word로 지정된 4기	개 주
		소내의 값이 됨. 이때 1st/2nd word는 QZ 명령어에 의해 설정되는 1st/2nd 주소값을 변경시킴.	
;SrHLHL,HLHL;		Pp/Pq 명령에 대한 return 값은 항상 1st/2nd word로 지정된 2개 주소내의 값이 됨.	
(+#^%)		TMS320LF2406A	
;SrHLHL,HLHL,HLH		[0000 - 005F] : memory mapped registers [0060 - 007F] : heap memory	
L,HLHL; (+#^%)		[0200 - 037F] : dual access RAM [0380 - 03FF] : stack	
		[0800 - 0FFF] : single access RAM [7010 - 7531] : peripheral memory	
;Sr?; (+#^%)	comment	TMS320F810	
	Commert	[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)	
		[0880 - 783F] : peripheral memory [8000 - 9FFF] : L0/L1 RAM block (.ebss)	
		[E000 - FFFF] : HO RAM block (.esysmem)	
		70A8(7108) - 70B7(7117) : A/D channel 0 - 15 (ADO CURRENT1_SEN+, AD1 CURRENT1_SEN-, AD2	
		CURRENT1_SET, AD3 LS1A, AD4 LS1B, AD5 LS1C, AD6 BLDC_CURR_SEN1, AD7 DC_BUS_VOLT, AD8 CURRENT2_S	SEN+,
		AD9 CURRENT2_SEN-, AD10 CURRENT2_SET, AD11 LS2A, AD12 LS2B, AD13 LS2C, AD14 BLDC_CURR_SEN2, AD1	15
		FET_TEMP)	
		기본값은 70AB(710B) : LS1A, 70B3(7113) : LS2A 임.	
		AD 변환값을 읽는 경우는 MSB에서 부터 10bit/12bit가 유효값임.	



명령어	구분	설명
	operation	Hall sensor pulse의 지정된 edge를 기준으로 하여 엔코더 Z펄스의 위치를 앤코더펄스 갯수로 읽어 냄. (BLDC 구동 chip IRMCK201 전용) 1st word: BLDC motor1 에서 출력되는 Hall sensor pulse의 지정된 edge 값 [0 - 6]
	operation	2nd word : BLDC motor2 에서 출력되는 Hall sensor pulse의 지정된 edge 값 [0 - 6] 0 이면 Z펄스의 위치를 앤코더펄스 갯수로 읽어 내는 작동을 하지 않음.
		SZdddd1,dddd2;
		1st word : BLDC motor1 - Hall sensor pulse의 지정된 edge 값 (SZdddd1,dddd2; 명령시)
		: BLDC motor1 - Hall sensor pulse의 지정된 edge를 기준으로 한 엔코더 Z펄스의
	return value	위치값 (SZ?; 명령시, 단위는 4체배된 엔코더펄스수)
;SZdddd1,dddd2;		2nd word : BLDC motor2 - Hall sensor pulse의 지정된 edge 값 (SZdddd1,dddd2; 명령시)
(+#^%)		: BLDC motor2 - Hall sensor pulse의 지정된 edge를 기준으로 한 엔코더 Z펄스의
		위치값 (SZ?; 명령시, 단위는 4체배된 엔코더펄스수)
;SZ?; (+#^%)		BLDC 모터제어기에 전원을 투입한후, SM 과 PE 명령어로 모터의 작동개시를 하지 않은 상태에서 명령을 수행토록 해야하며, 한번에 한개의 모터에 대해서만 실시하여야 함.
		SZdddd1,dddd2; 명령을 입력하면 작동 LED가 빠르게 깜빡거리기 시작함. 이어서 모터축을 들여다 볼 때
		반시계방향으로 모터축을 천천히 돌리면서 작동 LED의 초당 10회 깜빡임이 정상주기 (500ms, 초당 2회)로
	comment	돌아올 때까지 계속함. 이후에 SZ?; 명령어로 엔코더 Z펄스의 위치값을 읽음.
	Commert	SZ?; 에 대한 return 값은 SZdddd1,dddd2; 명령을 수행이 끝나서 얻어진 엔코더 Z펄스의 위치값임. 이
		러한 동작을 몇번 반복하여 읽은 엔코더 Z펄스의 위치값들이 거의 같은 상태가 되도록 만들고 최종값을 적
		절한 연산을 거쳐 변환한 다음, Qt06,00,HLHL; 또는 Qt06,01,HLHL; 명령어를 사용하여 모터제어기에 전송
		하고, ESOA00,0100; 명령어로 EEPROM에 저장하면 됨.
		BLDC인 경우에 한하여만 사용이 가능한 명령임.



명령어	구분	설명
	operation	RS232C 직렬통신속도값을 설정함.
		1 : 속도설정값 [9600,19200,38400,57600,115200, 625000,1250000,2500000] (115200 bps)
	return value	SBIIII1; (SB?; 인 경우에만 echo 있음)
;SB 1; (+#^%)		DIP 스위치 1번이 ON 되어 있는 상태에서 전원이 투입되었을 때(RS232C 모드)는 baud rate 설정 레지스터
, σοιτιτι, (τπ π)		값을 바꾸어서 실제 통신속도를 바꾸나, RS485모드인 상태에서는 EEPROM 저장용 parameter 만 설정함.
;SB?; (+#^%)		설정값이 정당한 baud rate가 아니면 설정값이 무시됨.
, OD: , (I'I' 10)	comment	전원투입시에 DIP 스위치 4번을 ON 하고 전원을 투입하면 default 값인 115200 으로 설정되며, 그 이외의
		경우는 EEPROM에 저장된 값으로 설정되나, 정당한 baud rate가 아니면 default 값인 115200으로 설정됨.
		IIIIx 는 unsigned long integer 임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		RS485 직렬통신속도값을 설정함.
	operation	11 1 : 속도설정값 [9600,19200,38400,57600,115200, 250000, 500000, 625000,1250000,2500000] (1250000
		bps)
	return value	SbIIII1; (Sb?; 인 경우에만 echo 있음)
;Sb 1; (+^%)		DIP 스위치 1번이 OFF 되어 있는 상태에서 전원이 투입되었을 때(RS485 모드)는 baud rate 설정 레지스터
		값을 바꾸어서 실제 통신속도를 바꾸나, RS232C모드인 상태에서는 EEPROM 저장용 parameter 만 설정함.
;Sb?; (+^%)		설정값이 정당한 baud rate가 아니면 설정값이 무시됨.
	comment	전원투입시에 DIP 스위치 4번을 ON 하고 전원을 투입하면 default 값인 1250000 으로 설정되며, 그 이외의
		경우는 EEPROM에 저장된 값으로 설정되나, 정당한 baud rate가 아니면 default 값인 1250000으로 설정됨.
		IIIIx 는 unsigned long integer 임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
명령어 ;Sb!!!!1,!!!12; (#) ;Sb?; (#)	operation return value	RS485 와 RS232C extra 직렬통신속도값을 설정함. : 속도설정값 [9600,19200,38400,57600,115200, 625000,1250000,2500000] (1250000 bps) : 속도설정값 [9600,19200,38400,57600,115200, 625000,1250000,2500000] (115200 bps) : 속도설정값 [9600,19200,38400,57600,115200, 625000,1250000,2500000] (115200 bps) Sb
	comment	만 설정함. 설정값이 정당한 baud rate가 아니면 설정값이 무시됨. RS485의 경우 전원투입시에 DIP 스위치 4번을 ON 하고 전원을 투입하면 default 값인 1250000 으로 설정되며, 그 이외의 경우는 EEPROM에 저장된 값으로 설정되나, 정당한 baud rate가 아니면 default 값인 1250000으로 설정됨.
		RS232C extra는 설정값이 정당한 baud rate가 아니면 115200 으로 설정됨. IIIIx 는 unsigned long integer 임. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		butterworth filter의 시정수와 변화율 상한값을 설정함.
	operation	1st word : filter 번호. [0 - 7/8/9] (번)
	operation	2nd word : filter의 시정수. [1 - 32767] (bit)
		3rd word : filter의 출력변화율 상한값. [1 - 32767] (bit)
	return value	SFdddd1,dddd2,dddd3;
		butterworth filter는 2차 필터로서 0-7/15 까지 총 8/16개가 있으며 1ms 마다 filter가 동작함.
;SFdddd1,dddd2,d		2nd word를 2로 나눈 값이 ms 단위의 실제 시정수에 해당함.
ddd3; (+#^%)		3rd word는 출력이 1ms 당 변할 수 있는 최대값을 지정함.
αααο, (1π λο)		6번 필터는 모터구동용 전압의 AD 변환값에 사용되며 시정수/변화율상한치의 기본값은 16/64 임.
;SF?dddd1;		7번 필터는 MOTOR1의 전류 AD 변환값에 사용되며 시정수/변화율상한치의 기본값은 256/4 임.
(+#^%)		2010년 12월 7일 S/W 버젼부터는 filter의 시정수로 1,2,4,8,16,32,64,128,256,512,1024,2048,4096,8192
(1π /0)	comment	값만 유효한데, 계산시간을 줄이기 위하여 shift operator를 사용함으로서 새로이 생겨난 제한임.
		0-7번, 9번 필터의 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		8번 필터의 설정값은 F2811/F28334에서만 유효하며 내부에 있는 8-15번 까지의 필터에 대하여 시정수 및
		출력변화율 상한값은 같은 모두 값을 사용함.
		8-15번 필터의 시정수 및 출력변화율 상한값의 기본값은 8,64이며 EEPROM에 저장이 안되는 값임.
		2012년 3월 9일 S/W 버젼부터 필터번호가 9인 경우는 F2811/F28334 H/W 버젼에서만 유효하며 F/T센서와 외
		부 analog 명령의 필터일에 사용하는 추가필터의 시정수을 설정하기 위한 것임.



명령어	구분	설명
		RJM_VER1과 RJM_VER2의 경우에는 과전류에러 fault 진단을 위한 1.024초 동안의 과전류검출 횟수를 설정하며, 그외의 경우는 SI 명령어로 설정한 전류치보다 실제전류가 연속적으로 초과하기를 허용하는 최대시간을 설정함.
	operation	RJM_VER1 또는 RJM_VER2의 경우 : 1st word : 이 횟수 이상으로 모터1의 과전류를 검출하였을때 MOTOR1_MAX_CURRENT_fault_flg를 세트함. [1 - 1024] (50 회) 2nd word : 이 횟수 이상으로 모터2의 과전류를 검출하였을때 MOTOR2_MAX_CURRENT_fault_flg를 세트함. [1 - 1024] (50 회)
2012년 10월 03일 이전 S/W 버젼 ;Sidddd1,dddd2; (+#^%)		그외의 경우 : 1st word : 이 횟수 이상으로 연속해서 모터1에서 SI 명령어로 설정한 전류치를 초과하는 경우 MOTOR1_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (500 ms, 380V 제어기의 경우는 50) 2nd word : 이 횟수 이상으로 연속해서 모터2에서 SI 명령어로 설정한 전류치를 초과하는 경우 MOTOR2_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (500 ms, 380V 제어기의 경우는 50)
.0:0. (.40%)	return value	Sidddd1,dddd2;
;Si?; (+#^%)	comment	이 설정값은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야함. RJM_VER1 또는 RJM_VER2의 경우에 기본값은 50(1024 번 중에 50번에 해당), 50(1024 번 중에 50번에 해당) 임. 그외의 경우는 500 또는 50ms로서 500 또는 50ms동안 SI 명령어로 설정한 전류치를 초과하는 경우 MOTORX_MAX_CURRENT_fault_flg를 세트하고 출력을 차단하며, 한번이라도 SI 명령어로 설정한 전류치를 초과하지 않으면 타이머를 0으로 clear함. 총전류에 대하여서는 SI 명령어로 설정한 전류치의 3/4 크기를 상한선으로 사용함. 380V 제어기가 아닌 경우에는 Si 명령에 의한 시간값을 SI 명령어로 설정하는 TIME_LIMIT_OF_CURRENT1_OVER 값의 3/4 보다 작지 않게 제한함. 2012년 5월 28일 이전 S/W 버젼에서는 RJM_VER1 또는 RJM_VER2의 DC motor에만 해당되었는데, 그 이후부터는 모든 제어기에 대하여 적용됨. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		RJM_VER1과 RJM_VER2의 경우에는 과전류에러 fault 진단을 위한 1.024초 동안의 과전류검출 횟수를 설정하
		며, 그외의 경우는 SI 명령어로 설정한 전류치보다 실제전류가 연속적으로 초과하기를 허용하는 최대시간
		을 설정함.
		RJM_VER1 또는 RJM_VER2의 경우 :
		1st word : 이 횟수 이상으로 모터1의 과전류를 검출하였을때 MOTOR1_MAX_CURRENT_fault_flg를 세트함.
		[1 - 1024] (50 회)
		2nd word : 이 횟수 이상으로 모터2의 과전류를 검출하였을때 MOTOR2_MAX_CURRENT_fault_flg를 세트함.
		[1 - 1024] (50 회)
2012년 10월 03일	operation	그외의 경우 :
S/W 버젼부터		1st word : 이 횟수 이상으로 연속해서 모터1에서 SI/Sw 명령어로 설정한 전류치중 작은 값의 125%를 초과
		하는 경우 MOTOR1_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (500 ms, 380V 제어기의 경우는 50)
;Sidddd1,dddd2;		2nd word : 이 횟수 이상으로 연속해서 모터2에서 SI/Sw 명령어로 설정한 전류치중 작은 값의 125%를 초과
(+#^%)		하는 경우 MOTOR2_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (500 ms, 380V 제어기의 경우는 50)
		3rd word : 이 횟수 이상으로 연속해서 모터1에서 SI/Sw 명령어로 설정한 전류치중 큰 값의 125%를 초과
또는		하는 경우 MOTOR1_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (3 ms)
		4th word : 이 횟수 이상으로 연속해서 모터2에서 SI/Sw 명령어로 설정한 전류치중 큰 값의 125%를 초과
Sidddd1,dddd2,dd		하는 경우 MOTOR1_MAX_CURRENT_fault_flg를 세트함. [1 - 10000] (3 ms)
dd3,dddd4; (+#^)	return value	Sidddd1,dddd2,dddd3,dddd4;
.0:0. (.400)		이 설정값은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야함.
;Si?; (+#^%)		RJM_VER1 또는 RJM_VER2의 경우에 기본값은 50(1024 번 중에 50번에 해당), 50(1024 번 중에 50번에 해당)
		임. 그외의 경우는 500 또는 50ms로서 500 또는 50ms동안 SI 명령어로 설정한 전류치를 초과하는 경우
		MOTORx_MAX_CURRENT_fault_flg를 세트하고 출력을 차단하며, 한번이라도 SI 명령어로 설정한 전류치를 초
	comment	과하지 않으면 타이머를 0으로 clear함. 총전류에 대하여서는 SI 명령어로 설정한 전류치의 3/4 크기를
		상한선으로 사용함. 380V 제어기가 아닌 경우에는 Si 명령에 의한 시간값을 SI 명령어로 설정하는 TIME_LIMIT_OF_CURRENT1_OVER 값의 3/4 보다 작지 않게 제한함.
		TIME_LIMIT_OF_CONNENTI_OVER 값의 3/4 보다 적지 않게 제한함. 2012년 5월 28일 이전 S/W 버젼에서는 RJM_VER1 또는 RJM_VER2의 DC motor에만 해당되었는데, 그 이후부터
		는 모든 제어기에 대하여 적용됨.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
KUDU		일당한 없는 다 당당이도 EEFNUMUN 사랑을 구 있습.

명령어	구분	설명
		제어오차가 과도하게 큰 상태를 진단하기 위한 위치오차,속도오차,전류오차,허용시간을 설정함. (RJM_VER5
		에는 해당없음)
		1st word : 모터1의 감속기 출력 기준의 허용 오차각. [1 - 1000] (10 도)
		2nd word : 모터2의 감속기 출력 기준의 허용 오차각. [1 - 1000] (10 도)
		3rd word : 모터1의 모터 출력축의 허용 속도오차. [1 - 10000]
		(1000 또는 SG 명렁어로 설정되는 MOTOR1_SPEED_MAX/5 rpm)
;Sxdddd1,dddd2,d	operation	4th word : 모터2의 모터 출력축의 허용 속도오차. [1 - 10000]
ddd3,dddd4,dddd5	operation	(1000 또는 SG 명렁어로 설정되는 MOTOR2_SPEED_MAX/5 rpm)
,dddd6,dddd7,ddd		5th word : 모터1의 허용 전류오차 [1 - 1000] (200 또는 SI 명렁어로 설정되는 current1_limit/2 bit)
d8; (+#^%)		6th word : 모터2의 허용 전류오차 [1 - 1000] (200 또는 SI 명렁어로 설정되는 current2_limit/2 bit)
		7th word : 모터1에서 이 시간 이상으로 연속으로 오류를 검출하였을 때 에러를 설정함.
;Sx?; (+#^%)		[1 - 50000] (10000 ms)
		8th word : 모터1에서 이 시간 이상으로 연속으로 오류를 검출하였을 때 에러를 설정함.
		[1 - 50000] (10000 ms)
	return value	Sxdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8;
	comment	이 설정값은 DSP의 SARAM에 저장되며 만약 EEPROM에 저장하려면 ;ESxxxx;명령어를 사용하여야함.
		기본값은 10도, 1000 또는 SG 명렁어로 설정되는 MOTOR1_SPEED_MAX/5 rpm, 200 또는 SI 명렁어로 설정되는
		current1_limit/2 bit, 10000ms 임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	명령어에 대한 return값을 보낼 때 사용하는 host의 주소를 설정함. 1st word 2012년 10월 18일 이전 S/W 버젼인 경우 또는 SX 명령어 첫번째파라메터의 bit0 = 1 인 경우 : 주소값 [0000 - 00FF] (FF00) 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 첫번째파라메터의 bit0 = 0 인 경우 : 주소값 [0000 - 00FF] (FF00) - SX 명령어 두번째파라메터의 bit15/14 = 11 일때 : 주소값 [0000 - 07FF] (0000) - SX 명령어 두번째파라메터의 bit15/14 = 10 일때 : 주소값 [0000 - FFFF] (0000) - SX 명령어 두번째파라메터의 bit15/14 = 01 일때 : 주소값 [0000 - 00FF] (FF00) - SX 명령어 두번째파라메터의 bit15/14 = 00 일때
	return value	SmHLHL;
;SmHLHL; (+#^%);Sm?; (+#^%)	comment	 ♣ 2012년 10월 18일 이전 S/W 버젼인 경우 또는 SX 명령어 첫번째파라메터의 bit0 = 1 인 경우와 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 두번째파라메터의 bit15/14 = 11 또는 00 일때는, 주소값의 상위 바이트는 반드시 하위바이트의 inverse이어야만 baord 주소의 설정이 유효함. ♣ 전원 투입시에 기본값은 FF00 또는 0000으로서 0번지를 return 값의 기본 host 주소로 지정함. ♣ 이 명령어에 대한 리턴값은 바뀐 주소의 host로 전송함. ♣ 전원을 깿다 켜지 않는 한 RS485 모드일 때 설정된 호스트 주소는 CAN에서는 적용되지 않으나, CAN 모드일 때 설정된 호스트 주소는 RS485 모드일때도 바로 적용됨. ♣ 2012년 10월 18일 S/W 버젼부터로서 SX 명령어 첫번째파라메터의 bit0 = 0 일때 SX 명령어 두번째파라메터의 bit15/14 값에 따라서 CAN 통신의 주소사용방법이 달라짐. bit15/14 값에 따라서 CAN 통신의 주소사용방법이 달라짐. bit15/14 = 11 일때 11-bit 주소모드로서 주소로 0-255만 사용 = 10 일때 11-bit 주소모드로서 주소로 0-255만 사용 = 01 일때 29-bit 주소모드로서 주소로 0-65535만 사용 (CAN2.0B), 2012년 10월 16일부터 적용 = 00 일때 11-bit 주소모드로서 주소로 0-255만 사용 ♣ 주소값으로서 254(0xFE) 및 255(0xFF)는 bradcasting용으로 사용하는 주소이므로 126(0x8E), 127(0x8F), 254(0xFE) 및 255(0xFF)는 host의 주소로 사용하면 안됨. ♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	모터1의 03/06/0A번 위치/속도제어모드에서 속도게인과 , RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터1의 전류제어파라메터를 설정함. 1st word : MOTOR1_SPEED_P_GAIN (32/20) 2nd word : MOTOR1_SPEED_I_GAIN (16/4/2) 3rd word : MOTOR1_SPEED_D_GAIN (8/2/1) 4th word : MOTOR1_CURRENT_P_GAIN (40) [0 - 1000] 5th word : MOTOR1_CURRENT_I_GAIN (180) [0 - 8000/32000]
	return value	WQdddd1,dddd2,dddd3,dddd4,dddd5;
;WQdddd1,dddd2,d ddd3,dddd4,dddd5 ; (+#) ;WQ?; (+#)	comment	처음 3개의 파라메터는 모터1의 03/06/0A번 위치/속도/속도제어모드에서의 속도제어게인을 설정함. 03 번 위치제어모드에서는 위치제어루프 내부의 1KHz 속도제어루프에서 사용하는 속도제어게인을 설정하며, 06번 속도제어모드에서는 BLDC에 한하여 T1UFINT_ISR에서 처리되는 10KHz 속도제어루프에서 속도제어게인을 설정하며, 0A번 속도제어모드에서는 1KHz 속도제어루프에서 사용하는 속도제어게인을 설정하는데 0A번 속도제어모드는 03번 위치제어루프 내부의 1KHz 속도제어루프와 같은 알고리즘이고 같은 게인을 사용함. 마지막 2개의 파라메터는 모터1로서 BLDC 모터와 RJM_VER7형의 3상 STEP모터인 경우에 07번 전류제어모드에서 전류제어게인을 설정함. 전류제어주기는 PWM 주파수와 같으며 전류제어 파라메터는 전류제어주파수(=PWM 주파수)와 무관하게 같은 값을 사용할수 있음. 07번 전류제어모드에서의 전류제어게인은 속도나 위치제어에서 내부 전류제어루프를 사용하는 경우에도 같은 값이 적용됨. 전류제어의 I-게인은 TMS320F2810/11/12 의 경우는 8000까지, TMS320F28334/5의 경우는 32000까지 설정 가능함. MOTOR1_SPEED_I_GAIN 은 2012년 12월 19일 S/W 버젼부터 0으로 설정하지 못하도록 하였음. BLDC1_PrmBuffer[25]/BLDC1_PrmBuffer[27]/BLDC1_PrmBuffer[13]/BLDC1_PrmBuffer[14]에도 설정된 값이 저장되며 이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음. F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
	operation	모터2의 03/06/0A번 위치/속도제어모드에서 속도게인과 , RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터2의 전류제어파라메터를 설정함. 1st word : MOTOR2_SPEED_P_GAIN (32/20) 2nd word : MOTOR2_SPEED_I_GAIN (16/4/2) 3rd word : MOTOR2_SPEED_D_GAIN (8/2/1) 4th word : MOTOR2_CURRENT_P_GAIN (40) [0 - 1000] 5th word : MOTOR2_CURRENT_I_GAIN (180) [0 - 8000/32000]
	return value	WRdddd1,dddd2,dddd3,dddd4,dddd5;
;WRdddd1,dddd2,d ddd3,dddd4,dddd5 ; (+#) ;WR?; (+#)	comment	처음 3개의 파라메터는 모터2의 03/06/0A번 위치/속도/속도제어모드에서의 속도제어게인을 설정함. 03 번 위치제어모드에서는 위치제어루프 내부의 1KHz 속도제어루프에서 사용하는 속도제어귀프에서 속도제어게인을 설정하며, 06번 속도제어모드에서는 BLDC에 한하여 T1UFINT_ISR에서 처리되는 10KHz 속도제어루프에서 속도제어게인을 설정하며, 0A번 속도제어모드에서는 1KHz 속도제어루프에서 사용하는 속도제어게인을 설정하는데 0A번 속도제어모드는 03번 위치제어루프 내부의 1KHz 속도제어루프와 같은 알고리즘이고 같은 게인을 사용함. 마지막 2개의 파라메터는 모터2로서 BLDC 모터와 RJM_VER7형의 3상 STEP모터인 경우에 07번 전류제어모드에서 전류제어게인을 설정함. 전류제어주기는 PWM 주파수와 같으며 전류제어 파라메터는 전류제어주파수(=PWM 주파수)와 무관하게 같은 값을 사용할수 있음. 07번 전류제어모드에서의 전류제어게인은 속도나 위치제어에서 내부 전류제어루프를 사용하는 경우에도 같은 값이 적용됨. 전류제어의 I-게인은 TMS320F2810/11/12 의 경우는 8000까지, TMS320F28334/5의 경우는 32000까지 설정 가능함. MOTOR1_SPEED_I_GAIN 은 2012년 12월 19일 S/W 버젼부터 0으로 설정하지 못하도록 하였음. BLDC2_PrmBuffer[25]/BLDC2_PrmBuffer[27]/BLDC2_PrmBuffer[13]/BLDC2_PrmBuffer[14]에도 설정된 값이 저장되며 이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음. F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		RJM_VER7 제어시에 100KHz interrupt routine에서 사용하는 Butterworth filter의 시정수를 설정함.
	operation	1st word : BUTTERWORTH_TIME_CONSTANT_for_MODE5 [0 - 4] (1)
;WBdddd1,dddd2;		2nd word : BUTTERWORTH_TIME_CONSTANT_for_MODE6 [0 - 4] (3)
(+#^)		WBdddd1,dddd2;
	return value	1st word : BUTTERWORTH_TIME_CONSTANT_for_MODE5
;WB?; (+#^)		2nd word : BUTTERWORTH_TIME_CONSTANT_for_MODE6
	comment	RJM_VER7에 한하여 적용됨.
	Commert	F2811 <mark>/F28334</mark> RJM_VER7에서만 가능한 명령임.
		BLDC 모터1,2에 있어서 Hall_A상 상승에지에 대한 U상 전압 positive zero crossing 포인트의 위상지연 값
		을 설정하며, 모터1,2의 POLE_PAIR 수, 모터1,2의 MOTORx_POSITION_SCALE_FACTOR를 읽어서 host로 전송
	operation	함.
		1st word : UPhase_to_HallA_ANGLE1_OFFSET (525/480/610/600 0.1도)
Wafer control의		2nd word : UPhase_to_HallA_ANGLE2_OFFSET (525/480/610/600 0.1도)
경우 (2012년 2월		WEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
18일 S/W 버젼부		1st word : UPhase_to_HallA_ANGLE1_OFFSET
터 적용):	return value	2nd word : UPhase_to_HallA_ANGLE2_OFFSET
		3rd word : MOTOR1_NUMBER_of_POLE_PAIR
;WEdddd1,dddd2;		4th word : MOTOR2_NUMBER_of_POLE_PAIR
(+#^)		5th word : MOTOR1_POSITION_SCALE_FACTOR
,,,		6th word : MOTOR2_POSITION_SCALE_FACTOR
;WE?; (+#^)		RJM_VER7에 한하여 적용됨.
	comment	Wafer control의 경우 WE명령어 두번째 파라메터로서 0~4096가 0도에서 +90도에 해당하며, 61440~65535가
		-90도에서 0도에 해당함)를 사용하여 ld를 만드는 PWM_offset값의 phase angle을 미세조정할수 있음.
		BLDC1_PrmBuffer[3]/BLDC1_PrmBuffer[4]/BLDC2_PrmBuffer[3]/BLDC2_PrmBuffer[4]에도 설정된 값이 저장되며
		이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음.
		F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		BLDC 모터1,2에 있어서 Hall_A상 상승에지에 대한 U상 전압 positive zero crossing 포인트의 위상지연 값
		을 설정하며, 모터1,2의 POLE_PAIR 수, 모터1,2의 MOTORx_POSITION_SCALE_FACTOR를 읽어서 host로 전송
	operation	함.
		1st word : UPhase_to_HallA_ANGLE1_OFFSET (525/480/610/600 0.1도)
		2nd word : UPhase_to_HallA_ANGLE2_0FFSET (525/480/610/600 0.1도)
Wafer control0		WEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
아닌 경우		1st word : UPhase_to_HallA_ANGLE1_OFFSET
WE 87		2nd word : UPhase_to_HallA_ANGLE2_OFFSET
;WEdddd1,dddd2;	return value	3rd word : MOTOR1_NUMBER_of_POLE_PAIR
(+#^)		4th word : MOTOR2_NUMBER_of_POLE_PAIR
(1117)		5th word : MOTOR1_POSITION_SCALE_FACTOR
;WE?; (+#^)		6th word : MOTOR2_POSITION_SCALE_FACTOR
,nc:, (iii)		RJM_VER7에 한하여 적용됨.
		Wafer control의 경우 WE명령어 두번째 파라메터로서 0~4096가 0도에서 +90도에 해당하며, 61440~65535가
	comment	-90도에서 0도에 해당함)를 사용하여 Id를 만드는 PWM_offset값의 phase angle을 미세조정할수 있음.
		BLDC1_PrmBuffer[3]/BLDC1_PrmBuffer[4]/BLDC2_PrmBuffer[3]/BLDC2_PrmBuffer[4]에도 설정된 값이 저장되며
		이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음.
		F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		analog Hall sensor type의 RJM_VER7 모터1의 센서 scale과 offset을 설정함.
		1st word : HALL_Sensor1_A_Scale (45621/47410/54142)
		2nd word : HALL_Sensor1_B_Scale (45792/46362/53092)
;WGdddd1,dddd2,d	operation	3rd word : HALL_Sensor1_C_Scale (46377/47077/55370)
ddd3,dddd4,dddd5		4th word : HALL_Sensor1_A_Offset (6518/7076/8038)
, dddd6; (+#^)		5th word : HALL_Sensor1_B_Offset (6558/6926/7852)
, duddo, (T#)		6th word : HALL_Sensor1_C_Offset (6652/7070/8317)
;WG?; (+#^)	return value	WGdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
, ma: , (m)		RJM_VER7에 한하여 적용됨.
	comment	BLDC1_PrmBuffer[57]/BLDC1_PrmBuffer[58]/BLDC1_PrmBuffer[59]/BLDC1_PrmBuffer[60]/BLDC1_PrmBuffer[61]/
	Commert	BLDC1_PrmBuffer[62]에도 설정된 값이 저장되며 이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음.
		F2811/F28334 RJM_VER7에서만 가능한 명령임.
		analog Hall sensor type의 RJM_VER7 모터2의 센서 scale과 offset을 설정함.
		1st word : HALL_Sensor2_A_Scale (45621/47410/54142)
		2nd word : HALL_Sensor2_B_Scale (45792/46362/53092)
;WHdddd1,dddd2,d	operation	3rd word : HALL_Sensor2_C_Scale (46377/47077/55370)
ddd3, dddd4, dddd5		4th word : HALL_Sensor2_A_Offset (6518/7076/8038)
,dddd6; (+#^)		5th word : HALL_Sensor2_B_Offset (6558/6926/7852)
, daddo, (in)		6th word : HALL_Sensor2_C_Offset (6652/7070/8317)
;WH?; (+ #^)	return value	WHdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
, mil. , (iii)	comment	RJM_VER7에 한하여 적용됨.
		BLDC2_PrmBuffer[57]/BLDC2_PrmBuffer[58]/BLDC2_PrmBuffer[59]/BLDC2_PrmBuffer[60]/BLDC2_PrmBuffer[61]/
	- Common t	BLDC2_PrmBuffer[62]에도 설정된 값이 저장되며 이를 통하여 ES 명령어에 의해 EEPROM에 저장될 수 있음.
		F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		analog Hall sensor type의 RJM_VER7 모터에 대하여 위상각 측정용 파라메터의 설정을 개시함.
	operation	1st word : 모터1의 파라메터 설정을 개시하지 않음(0)/개시함(1)/중지함(2)
		2nd word : 모터2의 파라메터 설정을 개시하지 않음(0)/개시함(1)/중지함(2)
		WCdddd1,dddd2;
;WCdddd1,dddd2;	return value	1st word : analog_Hall_sensor1_calibration_cnt
(+#^)		2nd word : analog_Hall_sensor2_calibration_cnt
(†#)		RJM_VER7에 한하여 적용됨.
;WC?; (+#^)		작동모드 0일 상태에서만 설정의 개시가 가능하며, 일단 개시되면 강제로 중지 시키지 않는 한 약 1+3/4바
, WO: , (1#)		퀴정도의 회전이 있어야, 파라메터 설정에 필요한 측정을 수행하고 파라메터 설정을 완료함.
	comment	파라메터 설정과정이 수행중일 때는 램프가 초당 10회 깜빡거리며, 설정이 완료되거나 취소되면 정상주기
		(500ms, 초당 2회)로 돌아옴.
		실행결과로 설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F2811 /F28334 RJM_VER7에서만 가능한 명령임.
		RJM_VER7 모터의 electrical angle을 강제로 설정함.
;WXiiii1,iiii2;	operation	1st integer : 모터1의 electrical angle [-360 - +360] (기본값 없음, 1도 단위)
(+#^)		2nd integer : 모터2의 electrical angle [-360 - +360] (기본값 없음, 1도 단위)
(π)	return value	WXiiii1,iiii2;
;WX?; (+ #^)		RJM_VER7에 한하여 적용됨.
, HA: , (#)	comment	작동모드 8이면서 속도 0일 때에만 설정의 의미가 있음.
		F2811 <mark>/F28334</mark> RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		RJM_VER7 analog Hall sensor type에서 구동가능한 영역에서의 analog Hall sensor wave 총갯수.
		1st word : 모터1의 analog Hall sensor wave 총갯수 [3 - 63] (7)
;WNdddd1,dddd2;		2nd word : 모터2의 analog Hall sensor wave 총갯수 [3 - 63] (7)
(#)	return value	WNdddd1,dddd2;
		RJM_VER7 analog Hall sensor에 한하여 적용됨.
;WN?; (#)	a a mm a n t	2012년 5월 1일 S/W 버젼부터 wave 총갯수가 128에서 64개로 줄었음.
	comment	설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F28334 RJM_VER7에서만 가능한 명령임.

명령어	구분	설명
;WJdddd1,dddd2,d	operation	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함. 1st word : 1 : [STEP 1] analog Hall sensor의 파라메터를 측정하여 저장하기를 개시함. 2 : [STEP 2] U상과 Hall sensor A상의 위상각차이를 측정하여 저장하기를 개시함. 3 : [STEP 3] 정역회전을 하면서 U상과 Hall sensor A상의 위상각 차이를 세밀하게 조정하여 저장하기를 개시함. 4 : [STEP 4] 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사함. 2nd word : STEP 3의 세밀한 조정시에 Hall sensor A상의 위상각을 업데이트하는 게인 [1 - 20] 3rd word : STEP 3의 세밀한 조정시에 정역운전의 최대 허용 횟수 [1-20] 4th word : STEP 3의 세밀한 조정시에 사용하는 RPM [0-40] 5th word : STEP 1/2/3의 성공적인 완료후 최종 시험을 실시하는 RPM [0-40]
ddd3,dddd4,dddd5; (제일메티컬의 경우)(+#^);WJ?;(+#^)	return value	WJdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; 1st word : 1 : analog Hall sensor의 파라메터를 측정중임. 2 : U상과 Hall sensor A상의 위상각차이를 측정중임. 3 : 정역회전을 하면서 U상과 Hall sensor A상의 위상각차이를 세밀하게 조정중임. 4 : 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사하는 중임. 0 : 측정동작을 개시하지 않았음. 7 : 측정동작을 완료하였음. 8 : U상과 Hall sensor A상의 위상각차이가 50 (5도) 이상 업데이트되는 오류가 있었음 9 : 측정동작에서 오류가 있었음. 2nd word : STEP 3의 세밀한 조정시에 Hall sensor A상의 위상각을 업데이트하는 게인 3rd word : STEP 3의 세밀한 조정시에 정역운전의 최대 허용 횟수로부터 down count된 현재값 4th word : STEP 3의 세밀한 조정시에 사용하는 RPM [0-40] 5th word : STEP 1/2/3의 성공적인 완료후 최종 시험을 실시하는 RPM [0-40] 6th word : 파라메터 측정 및 저장의 진척도 count



명령어	구분	설명
		작동모드가 0000일때만 개시가 가능함.
		STEP 1 측정이 정상적으로 이루어 졌으면 STEP 2 측정을 자동으로 실행하며, STEP 2 측정이 정상적으로 이
		루어 졌으면 STEP 3 측정을, STEP 3 측정이 정상적으로 이루어 졌으면 STEP 4 측정을 자동으로 실행함.
		매 스텝마다 측정이 정상이었으면 해당 파라메터를 매 스텝마다 EEPROM에 저장함.
		4th word, 5th word에서 0~40은 0~40000 RPM 을 의미함.
		리턴된 값의 첫번째 워드가 8일 때 U상과 Hall sensor A상의 위상각차이가 5도 이상 업데이트되는 오류가
		있었음.
		리턴된 값의 첫번째 워드가 9일 때 (측정동작에서 오류가 있었을 때), 리턴된 마지막 워드값이
		300 이면 아날로그 홀센서의 진폭이 적절하지 못한 경우임.
	comment	1050 이면 U상과 Hall sensor A상의 위상각 차이가 30도에서 90도 범위를 벗어 난 경우임.
		2400 이면 4th word로 지정된 +RPM을 200RPM 이상 미달 한 경우임.
		2700 이면 4th word로 지정된 -RPM을 200RPM 이상 미달 한 경우임.
		2701 이면 허용된 횟수 이내에서 STEP 3의 세밀한 조정을 실패하였음.
		3200 이면 5th word로 지정된 +RPM을 200RPM 이상 미달 한 경우임.
		3500 이면 5th word로 지정된 -RPM을 200RPM 이상 미달 한 경우임.
		3000 이상이면 측정된 파라메터의 저장은 모두 이루어진 경우임.
		측정된 analog Hall sensor의 파라메터값은 모터1/2에 대하여 각각 WG?; WH?; 의 리턴값임.
		측정된 U상과 Hall sensor A상의 위상각차이는 WE?; 명령어의 리턴값중에서 앞쪽의 2개임.
		설정된 파라메터 값은 측정시 오류가 없었다면 EEPROM에 저장되었음.
		F2811/F28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함.
	operation	1st word : 1 : [STEP 1] analog Hall sensor1의 파라메터를 측정하여 저장하기를 개시함.
		2 : [STEP 2] U상과 Hall sensor1 A상의 위상각차이를 측정하여 저장하기를 개시함.
;WJdddd1;		WJdddd1,dddd2;
(나노모션의		1st word : 1 : analog Hall sensor1의 파라메터를 측정중임.
linear Hall	return value	2 : U상과 Hall sensor1 A상의 위상각차이를 측정중임.
sensor의 경우)		0 : 측정동작을 개시하지 않았음.
(#^)		11 : 측정동작을 완료하였음.
		2nd word : 파라메터 측정 및 저장의 진척도 count
;WJ?; (#^)		작동모드가 0000일때만 개시가 가능함.
	comment	STEP 1 측정이 정상적으로 이루어 졌으면 STEP 2 측정을 자동으로 실행함.
		매 스텝마다 측정이 정상이었으면 해당 파라메터를 매 스텝마다 EEPROM에 저장함.
		28334 RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파라메터 설정을 개시함.
		1st word : 1 : [STEP 1] analog Hall sensor1의 파라메터를 측정하여 저장하기를 개시함.
		2 : [STEP 2] U상과 Hall sensor1 A상의 위상각차이를 측정하여 저장하기를 개시함.
		3 : [STEP 3] 정역회전을 하면서 모터1의 U상과 Hall sensor A상의 위상각 차이를 세밀하게
		조정하여 저장하기를 개시함.
		4 : [STEP 4] 모터1을 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사함.
		6 : [STEP 1] analog Hall sensor2의 파라메터를 측정하여 저장하기를 개시함.
	operation	7 : [STEP 2] U상과 Hall sensor2 A상의 위상각차이를 측정하여 저장하기를 개시함.
		8 : [STEP 3] 정역회전을 하면서 모터2의 U상과 Hall sensor A상의 위상각 차이를 세밀하게
;WJdddd1,dddd2,d		조정하여 저장하기를 개시함.
ddd3,dddd4,dddd5		9 : [STEP 4] 모터2를 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사함.
;		2nd word : STEP 3의 세밀한 조정시에 Hall sensor A상의 위상각을 업데이트하는 게인 [1 - 20]
(그외의 경우)		3rd word : STEP 3의 세밀한 조정시에 정역운전의 최대 허용 횟수 [1-20]
(+#^)		4th word : STEP 3의 세밀한 조정시에 사용하는 RPM [0-40]
		5th word : STEP 1/2/3의 성공적인 완료후 최종 시험을 실시하는 RPM [0-40]
;WJ?; (+#^)		WJdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
		1st word : 1 : analog Hall sensor1의 파라메터를 측정중임.
		2 : U상과 Hall sensor1 A상의 위상각차이를 측정중임.
		3 : 정역회전을 하면서 모터1의 U상과 Hall sensor A상의 위상각차이를 세밀하게 조정중임.
	return value	4 : 모터1을 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사하는 중임.
	rotum turus	6 : analog Hall sensor2의 파라메터를 측정중임.
		7 : U상과 Hall sensor2 A상의 위상각차이를 측정중임.
		8 : 정역회전을 하면서 모터2의 U상과 Hall sensor A상의 위상각차이를 세밀하게 조정중임.
		9 : 모터2을 설정 RPM으로 정역회전을 하면서 속도값이 미달되는 지를 검사하는 중임.
		0 : 측정동작을 개시하지 않았음.



분	설명
	11 : 측정동작을 완료하였음.
	12 : U상과 Hall sensor A상의 위상각차이가 50 (5도) 이상 업데이트되는 오류가 있었음
	13 : 측정동작에서 오류가 있었음.
	2nd word : STEP 3의 세밀한 조정시에 Hall sensor A상의 위상각을 업데이트하는 게인
	3rd word : STEP 3의 세밀한 조정시에 정역운전의 최대 허용 횟수로부터 down count된 현재값
	4th word : STEP 3의 세밀한 조정시에 사용하는 RPM [0-40]
	5th word : STEP 1/2/3의 성공적인 완료후 최종 시험을 실시하는 RPM [0-40]
	6th word : 파라메터 측정 및 저장의 진척도 count
	작동모드가 0000일때만 개시가 가능함.
	STEP 1 측정이 정상적으로 이루어 졌으면 STEP 2 측정을 자동으로 실행하며, STEP 2 측정이 정상적으로 이
	루어 졌으면 STEP 3 측정을, STEP 3 측정이 정상적으로 이루어 졌으면 STEP 4 측정을 자동으로 실행함.
	매 스텝마다 측정이 정상이었으면 해당 파라메터를 매 스텝마다 EEPROM에 저장함.
	4th word, 5th word에서 0~40은 0~40000 RPM 을 의미함.
	리턴된 값의 첫번째 워드가 12일 때 U상과 Hall sensor A상의 위상각차이가 5도 이상 업데이트되는 오류가 있었음.
	지었음. 리턴된 값의 첫번째 워드가 13일 때 (측정동작에서 오류가 있었을 때), 리턴된 마지막 워드값이
omment	300 이면 아날로그 홀센서의 진폭이 적절하지 못한 경우임.
	1050 이면 U상과 Hall sensor A상의 위상각 차이가 30도에서 90도 범위를 벗어 난 경우임.
	2400 이면 4th word로 지정된 +RPM을 200RPM 이상 미달 한 경우임.
	2700 이면 4th word로 지정된 -RPM을 200RPM 이상 미달 한 경우임.
	2700 이런 4HT WORD도 지정된 FIFM를 200AFM 이정 비를 된 정부점. 2701 이면 허용된 횟수 이내에서 STEP 3의 세밀한 조정을 실패하였음.
	27이 이런 어흥선 뜻부 이내에서 SIC 3의 세월인 조정을 할때하였음. 3200 이면 5th word로 지정된 +RPM을 200RPM 이상 미달 한 경우임.
	3200 이런 5th word로 지정된 -RPM을 200RPM 이상 미달 한 경우임.
	3000 이런 5HT WOLD도 지정된 FARM를 200AFM 이정 미필 된 경우함. 3000 이상이면 측정된 파라메터의 저장은 모두 이루어진 경우임.



명령어	구분	설명
		측정된 analog Hall sensor의 파라메터값은 모터1/2에 대하여 각각 WG?; WH?; 의 리턴값임.
	oommon t	측정된 U상과 Hall sensor A상의 위상각차이는 WE?; 명령어의 리턴값중에서 앞쪽의 2개임.
	comment	설정된 파라메터 값은 측정시 오류가 없었다면 EEPROM에 저장되었음.
		F2811 <mark>/F28334</mark> RJM_VER7에서만 가능한 명령임.



명령어	구분	설명
		encoder/digital_Hall_IC_only type BLDC 모터의 자동 파라메터 설정을 개시함.
		1st word : 1 : [STEP 1] encoder type BLDC 모터 1의 pole pair 수를 측정하여 저장하기를 개시함.
	operation	2 : [STEP 2] 모터 1의 U상과 Hall sensor A상의 위상각차이를 측정하여 저장하기를 개시함.
		6 : [STEP 1] encoder type BLDC 모터 2의 pole pair 수를 측정하여 저장하기를 개시함.
		7 : [STEP 2] 모터 2의 U상과 Hall sensor A상의 위상각차이를 측정하여 저장하기를 개시함.
		WKdddd1,dddd2;
		1st word : 1 : encoder type BLDC 모터 1의 pole pair 수를 측정중임.
		2 : 모터 1의 U상과 Hall sensor A상의 위상각차이를 측정중임.
		6 : encoder type BLDC 모터 2의 pole pair 수를 측정중임.
	return value	7 : 모터 2의 U상과 Hall sensor A상의 위상각차이를 측정중임.
;WKdddd1; (+#^)		0 : 측정동작을 개시하지 않았음.
$\mathbf{H}(\mathbf{u}\mathbf{u}\mathbf{u}\mathbf{u})$		11 : 측정동작을 완료하였음.
;WK?; (+#^)		12 : Hall sensor A/B/C 상이 음의 방향으로 회전함.
, ιιιχ: , (ι π)		13 : 엔코더 값이 음의 방향으로 증가함.
		14 : 계산된 pole pair 수가 0임.
		15 : 계산된 pole pair 수가 50 보다 큼.
		16 : Hall sensor의 low level 전위가 너무 높음.
		17 : Hall sensor의 high level 전위가 너무 낮음.
		18 : digital_Hall_IC_only type 이므로 STEP 2 측정을 하지 않았음.
		2nd word : 파라메터 측정 및 저장의 진척도 count
	comment	작동모드가 0000일때만 개시가 가능함.
		pole pair 수를 측정하기 전에 digital Hall sensor의 threshold를 자동으로 설정함.
		STEP 1 측정이 정상적으로 이루어 졌으면 STEP 2 측정을 자동으로 실행함.
		매 스텝마다 측정이 정상이었으면 해당 파라메터를 매 스텝마다 EEPROM에 저장함.



명령어	구분	설명
		리턴된 값의 첫번째 워드가 11이면 측정과 저장이 완료된 것임.
		digital_Hall_IC_only type BLDC 모터의 경우는 STEP 1의 Hall sensor threshold 측정만 실시함.
		측정된 pole pair 수는 SE?;의 리턴값중 마지막 2개의 값임.
		측정된 U상과 Hall sensor A상의 위상각차이는 WE?; 명령어의 리턴값중에서 앞쪽의 2개임.
		모터1의 MAX_PERCENT_SLIP1, ELECTRIC_TIME_CONSTANT1 값을 설정함.
;Wqdddd1,dddd2;	operation	1st word : MAX_PERCENT_SLIP1 (25 %)
(+#^)		2nd word : ELECTRIC_TIME_CONSTANT1 (4000 us)
	return value	Wqdddd1,dddd2;
;Wq?; (+#^)	comment	2012년 11월 30일 S/W 버젼부터 브레이크제어기/조향제어기/자석가동자에서는 적용안됨.
	Commert	설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		모터2의 MAX_PERCENT_SLIP2, ELECTRIC_TIME_CONSTANT2 값을 설정함.
;Wrdddd1,dddd2;	operation	1st word : MAX_PERCENT_SLIP2 (25 %)
(+#^)		2nd word : ELECTRIC_TIME_CONSTANT2 (4000 us)
	return value	Wrdddd1,dddd2;
;Wr?; (+#^)	comment	2012년 11월 30일 S/W 버젼부터 브레이크제어기/조향제어기/자석가동자에서는 적용안됨.
		설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		3상 analog Hall 센서를 사용하는 경우에 A/B/C 상신호의 교차점값을 설정함.
		1st word : 3상 analog Hall 신호의 삼상 wave sector 번호 (0 ~ 63)
;WLdddd1,dddd2,d		2nd word : C과 A상의 교차점중 큰 값 (0 ~ 4095)
ddd3, dddd4, dddd5	operation	3rd word : A과 B상의 교차점중 큰 값 (0 ~ 4095)
, dddd6, dddd7;	opor at ron	4th word : B과 C상의 교차점중 큰 값 (0 ~ 4095)
(+#^)		5th word : B과 C상의 교차점중 작은 값 (0 ~ 4095)
		6th word : C과 A상의 교차점중 작은 값 (0 ~ 4095)
;WL?dddd1; (+#^)		7th word : A과 B상의 교차점중 작은 값 (0 ~ 4095)
	return value	WLdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;
	comment	설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	3상 analog Hall 센서를 사용하는 경우에 지정된 삼상 wave sector 번호부터 A/B/C 상신호의 교차점값을
		자동 설정함.
		1st word : 3상 analog Hall 신호의 삼상 wave sector 번호 (0 ~ 63)
	return value	WLP;
		자석가동자를 -방향쪽 센서보다 더 -쪽으로 위치시킨 상태에서는 RrA55A;WLPO;를 차례로 입력하고 교차점
;WLPdddd1; (+#^)		자동설정을 개시하면 되며, -방향쪽의 위치리밋에 자석가동자가 닿은 상태에서 이미 -방향쪽 센서를 자석
THE dadd T		가동자가 덮고 있다면 그때의 삼상 wave sector 번호를 알아내어 RrA55A; WLPdddd1; 을 차례로 입력하고
	comment	교차점 자동설정을 개시해야함.
	Commert	-쪽 위치리밋에 닿아 있을때의 삼상 wave sector 번호를 알아내는 방법은 자석가동자를 +방향쪽 센서보다
		더 +쪽으로 위치시킨 상태에서 RrA55A;를 입력한 다음 -방향쪽으로 -방향 위치리밋까지 움직여서
		moving_magnet_sensor_cnt 변수값을 읽으면 됨.
		설정된 파라메터 값은 ESOC00,0300; 명령어로 EEPROM에 저장될 수 있음.
	operation	3상 analog Hall 센서를 사용하는 자석가동자의 경우에 -방향쪽 센서와 +방향쪽 센서의 위상차를 자동 설
		정함.
	return value	WLQ;
;WLQ; (+#^)		좌측 센서의 에지를 감지할수 있는 위치에서 RrA55A;를 실행한 후, 자석가동자를 움직여서 좌측 센서의 에
	comment	지만 검출한 상태를 만들고 나서, WLQ; 명령을 주고 자석가동자를 + 방향으로 서서히 움직여서 우측센서
		에지가 감지되도록 함. 설정된 값은 SR?; 의 리턴값에서 마지막 두번째 값임.
		설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
	operation	3상 analog Hall 센서를 사용하는 자석가동자의 경우에 순위가 높은 제어기의 +방향쪽 센서와 순위가 낮은
	орогаттоп	현재 제어기의 -방향쪽 센서 사이의 위상차를 자동 설정함.
;WLR; (+#^)	return value	WLR;
		순위가 높은 제어기의 -방향쪽 또는 +방향쪽 센서의 에지를 검출하도록 자석가동자를 움직인 상태에서 (단
	comment	이때 순위가 낮은 현재 제어기의 -방향쪽 또는 +방향쪽 센서의 에지가 검출되면 안됨) 순위가 낮은 현재
		제어기에 RrA55A; WLR; 명령을 차례로 주고 가동자를 손으로 + 방향으로 서서히 움직여서 현재 제어기의
		우측센서 에지가 감지되지 위치까지 이동시키면 됨. 설정된 값은 SR?; 의 리턴값에서 마지막 값임.
		설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		위치제어모드에서 속도에 비례하는 damping 회전력을 발생시키는 게인을 설정함.
;WSdddd1,dddd2;	operation	1st word : SPEED1_FEEDBACK_GAIN [0 ~ 30000] (bit)
(+#^)		2nd word : SPEED2_FEEDBACK_GAIN [0 ~ 30000] (bit)
	return value	WSdddd1,dddd2;
;WS?; (+#^)	comment	(RPM * SPEEDx_FEEDBACK_GAIN) / 4096 의 크기 만큼 전류명령값을 감소시킴.
	Collillett	설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		DD motor 등에서 속도에 비례하는 damping current를 발생시키는 게인을 설정함.
;Wsdddd1,dddd2;	operation	1st word : SPEED1_DAMPING_GAIN [0 ~ 30000] (bit)
(+#^)		2nd word : SPEED2_DAMPING_GAIN [0 ~ 30000] (bit)
(1#)	return value	Wsdddd1,dddd2;
;Ws?; (+#^)		(PWM 주파수의 기간동안 변화된 위치 펄스수 * SPEEDx_DAMPING_GAIN) / 4096 의 크기 만큼 전류명령값을
, mo: , (m)	comment	감소시킴.
		설정된 파라메터 값은 ES 명령어로 EEPROM에 저장될 수 있음.
;Wciiii1,iiii2;		유도모터의 경우에 ld 전류의 기준값을 설정함.
(+#^)	operation	1st word : 모터1의 ld 전류의 기준값 (0 bit)
(π)		2nd word : 모터2의 ld 전류의 기준값 (0 bit)
;Wc?; (+#^)	return value	Wciii1,iiii2;
,πο: , (π)	comment	유도모터의 경우에만 사용할수 있음.



명령어	구분	설명
	operation	BLDC 모터에 있어서 cogging torque를 보상하는 파라메터를 설정함. 1st word: 모터1의 cogging torque 보상 진폭 [0 ~ 1023] (0 bit) 2nd word: 모터1의 cogging torque 위상 [0 ~ 1023] (0 bit) 3rd word: 모터2의 cogging torque 보상 진폭 [0 ~ 1023] (0 bit) 4th word: 모터2의 cogging torque 위상 [0 ~ 1023] (0 bit)
	return value	WTdddd1,dddd2,dddd3,dddd4;
;WTdddd1,dddd2,d ddd3,dddd4; (+#^)) ;WT?; (+#^)	comment	cogging torque 보상 진폭이 0이면 cogging torque 보상을 하지 않는 것이며, cogging torque 보상 진폭이 0이 아니면, cogging torque 보상 진폭의 사인파 전류값을 feedback에 의한 전류 명령값에 더하여 전류제여 명령값이 되게함. cogging torque 보상 진폭의 단위는 SC명령에서 사용하는 전류크기의 단위와 같음. cogging torque 위상은 0~1023일때 0~360도의 위상각을 의미하며, 회전자기준으로 반시계방향으로 90도 앞선 위치를 기준으로한 위상각임. 2012년 6월 12일 이전 S/W 버젼에서는 cogging torque 보상 진폭설정에 문제가 있어서 제대로 작동하지 않음. 2012년 12월 27일 S/W 버젼부터는 TMS320F2811을 사용하는 H/W 버젼으로서 analog Hall 센서를 사용하는 경우에 WT 명령 적용이 되지 않음.



명령어	구분	설명
		RJM_VER7 BLDC 모터제어기에서, 모드8의 host direct angle control을 사용하여 자동튜닝 등의 구동을 할
		때, 사용하는 PWM duty 값을 설정함.
2011년 6월 14일		1st word : 모터1의 PWM duty 값 [0 ~ PWM_PERIOD/2] (bit)
이전 S/W 버젼 :	operation	2nd word : 모터2의 PWM duty 값 [0 ~ PWM_PERIOD/2] (bit)
;Wkdddd1,dddd2;		3rd word : analog Hall 센서의 파라메터 측정을 위한 시험속도 [20 ~ 1000] (rpm)
(+#^)		4th word : analog Hall 센서의 상전압의 교차점 전압(높은 전압쪽) 크기 [100 ~ 4000] (bit)
		5th word : analog Hall 센서의 상전압의 교차점 전압(낮은 전압쪽) 크기 [100 ~ 4000] (bit)
2011년 6월 14일	return value	Wkdddd1,dddd2; (2011년 6월 14일 이전 S/W 버젼)
S/W 버젼부터 :	Teturii varue	Wkdddd1,dddd2,dddd3,dddd4,dddd5; (2011년 6월 14일 S/W 버젼부터)
;Wkdddd1,dddd2,d		모터제어기가 TMS320F2810/11/12을 사용하였을 때, 기본값은 analog Hall 센서를 사용하는 경우로서 mini
ddd3,dddd4,dddd5		VER7인 경우는 45이며 나머지는 55이고, digital Hall 센서 또는 엔코더를 사용하는 경우로서 mini VER7인
; (+#^)		경우는 50이며 나머지는 60임.
	comment	모터제어기가 TMS320F28334/5를 사용한 경우는 TMS320F2810/11/12을 사용하였을 때의 2배 값임.
;Wk?; (+#^)		과도한 값을 사용하면 모터가 과열되니 주의할 것.
		2013년 1월 02일 S/W 버젼부터는 3rd word 값의 하한치가 200에서 20으로 바뀌었음.
		2012년 2월 11일 S/W 버젼부터는 1st/2nd word 의 경우 EEPROM에 저장되는 변수로 바뀌었음.
		Digital Hall sensor type의 RJM_VER7 모터에 대하여 정지희망 edge 번호를 지정함.
	operation	1st word : 모터1의 정지희망 edge 번호 [0 ~ 모터1의_pole_pair_수*6-1]
		2nd word : 모터2의 정지희망 edge 번호 [0 ~ 모터2의_pole_pair_수*6-1]
;WPdddd1,dddd2;	return value	WPdddd1,dddd2;
(+#^)		Digital Hall sensor type의 모터에서 05번 모드로 회전을 하다가 PeA5A5;/Pe5A5A;/PeA55A; 명령어로 정지
	comment	할때 정지를 희망하는 에지의 번호를 지정함.
;WP?; (+#^)		최초 전원 투입후에 모터를 enable 및 02번 모드 설정을 실시하고, PA5000000,5000000;을 수행하여 정지된
		위치에서 WP?;로 읽은 에지번호가 기준위치이며, 05번 모드로 회전을 하다가 PeA5A5;/Pe5A5A;/PeA55A; 명
		령어로 정지하면 같은 위치에 정지함. 만약 정지 위치를 바꾸고자 하면 WP 명령어로 정지희망 edge 변
		호를 바꾸어 주고 재기동 및 정지를 실시하면 됨.



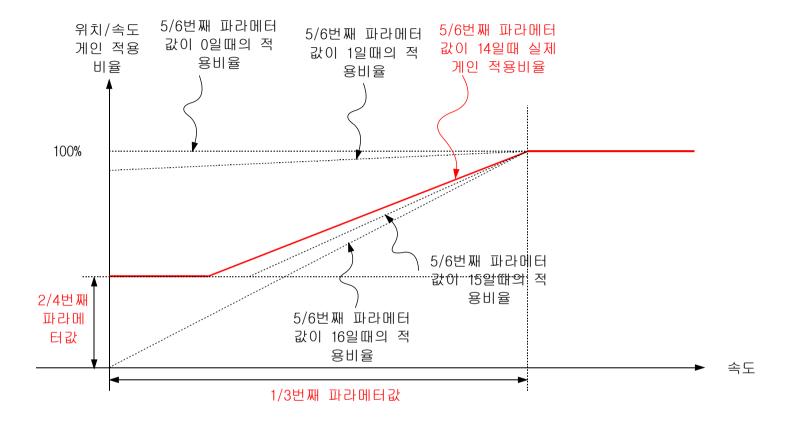
명령어	구분	설명
;Wpdddd1,dddd2,dddd3,dddd4,dddd5; (+#^)	operation	Digital Hall sensor type의 RJM_VER7 모터에 대하여 감속완료후 안정화를 개시하는 지연시간과 안정화용 analog Hall sensor의 offset, 및 analog Hall 센서가 사용되는 위치오차 범위등을 설정함. 1st word : 감속완료후 안정화를 개시하는 지연시간 [0 ~ 9999] (100 10ms) 2nd word : 안정화용 analog Hall sensor의 offset [0 ~ 4095] (2000 bit) 3rd word : analog Hall sensor를 적용하는 위치오차 범위 [0 ~ 8000] (1500 bit) 4th word : analog Hall sensor가 적용되는 기간에 사용하는 게인 factor [0 ~ 30000] (16384) 5th word : PvA55A;로 감속할때, 처음에는 속도모드로 감속하다가, 낮은 속도가 되면 위치모드로 정지하게 되는데, 그때의 경계속도값 [1 ~] (10)
	return value	Wpdddd1,dddd2,dddd3,dddd4,dddd5;
;Wp?; (+#^)	comment	Digital Hall sensor type의 RJM_VER7 BLDC 모터에서 WAFER_ROTATION==1인 경우에만 적용됨. 안정화용 analog Hall sensor는 모터2의 Hall sensor A입력 단자에 연결되어야하며 모터1의 Hall sensor A 입력과 상승하강 에지가 일치하여야함. 네번째 워드의 값이 16384이면 고속에서 적용하는 게인을 적용하는 것이며, 작을수록 게인을 적게하는 것임.
;WMHLHL; (+#^)	operation	Wafer 회전위치제어시에 위치모드로 정지한 한 상태에서 속도모드로 작동모드를 변경하며, 이에 따른 변수 초기화 작동을 실시함. 1st word : 작동모드 [0000 - FFFF] (0000)
	return value	WMHLHL;
;WM?; (+#^)	comment	♣ 1st word의 값이 0005 이어야만 초기화작동이 개시되며, 작동모드가 이미 0005 이면 초기화작동이 개시되지 않음. 1st word의 값이 0005 이외의 값이면 작동모드를 설정하지 않고 현재의 작동모드값을 보며주기만 함.



명령어	구분	설명
	operation	미분값 계산을 실시하는 시간차를 설정함. 1st word : 모터1의 제어에서 미분값 계산을 실시하는 시간차 [1/2/4/8] (2 ms) 2nd word : 모터2의 제어에서 미분값 계산을 실시하는 시간차 [1/2/4/8] (2 ms)
;WDA55A.dddd1.dd	return value	WDdddd1,dddd2;
;WDA55A,dddd1,dd dd2; (+#^) ;WD?; (+#^)	comment	♣ 1,2,4,8 만 유효한 값이며 그 이외의 값을 주면 2로 처리됨. ♣ 2012년 10월 6일 S/W 버젼부터 WD 파라메터의 값이 바뀌더라도 미분게인은 변하지 않도록 조치하였음. 따라서 2012년 10월 6일 이전 S/W 버젼을 사용하다 이후 S/W 버젼의 firmware로 upgrade 하면서 파라메터 는 이전의 것을 사용하는 경우로서 WD1,1; 또는 WD4,4; 또는 WD8,8;을 설정하여 사용하던 경우는 SQ/SR 명 령어에서 위치-inverse-D-gain을 작거나(*2 WD1,1;인경우) 크게(/2 WD4,4;인경우, /4 WD4,4;인경우) 바꾸 어주어야함. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.

명령어	구분	설명
명 60 M	operation	inner 속도제어루프가 있는 03번 위치제어모드의 속도제어 또는 Digital Hall sensor만을 사용하는 BLDC모 터의 위치/속도 제어 또는 0A번 속도제어모드 또는 속도제어로서 SX 명령어로 설정하는 두번째 파라메터의 bit11이 0인 경우에, 낮은 속도에서 위치/속도 제어게인(3번 위치제어모드의 경우는 속도게인만)을 낮게하 는 파라메터를 설정함. 1st word : 모터1의 위치/속도게인을 정상값으로 적용하는 모터속도의 최소값 [1 ~ 2000] (1000 rpm) 2nd word : 모터1의 위치/속도게인을 스케일링하는 스케일의 최소값 [1 ~ 100] (100 %) 3rd word : 모터2의 위치/속도게인을 정상값으로 적용하는 모터속도의 최소값 [1 ~ 2000] (1000 rpm) 4th word : 모터2의 위치/속도게인을 스케일링하는 스케일의 최소값 [1 ~ 100] (100 %) 5th word : 모터2의 위치/속도게인을 낮게하는 비율 [1 ~ 16] (16)
;Wddddd1,dddd2,d	return value	Wddddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
<pre>ddd3,dddd4,dddd5 ,dddd6; (+#^) ;Wd?; (+#^)</pre>	comment	inner 속도제어루프가 있는 03번 위치제어모드의 속도제어 또는 Digital Hall sensor만을 사용하는 BLDC모 터의 위치/속도 제어 또는 0A번 속도제어모드 또는 속도제어로서 SX 명령어로 설정하는 두번째 파라메터의 bit11이 0인 경우에, 속도에 비례하여 위치/속도게인(3번 위치제어모드의 경우는 속도게인만)을 스케일링하는데, 1st/3rd word에 해당하는 속도 이상에서는 scale factor는 1이며, 그 이하의 속도에서는 속도에 비례하여 scale factor 값이 낮아지는데 따라서 1보다 작은 값이 됨. scale factor는 2nd/4th word 로 지정한 값 이상으로 그 크기가 제한되며, 이 값이 100%이면 게인을 낮추는 작동을 하지 않는 것임. Qx77; 을 사용하면 적용되는 게인의 변화를 볼수 있음. (Qx77; 에서의 값이 16384일때 게인으로는 1.0에 해당함) Wafer 제어인 경우에는 기본값이 200rpm, 20%임. 모터x의 위치/속도게인을 낮게하는 비율이 16이면 속도에 비례하여 게인을 낮추는 것이고, 15이면 속도의 93.75% 비율로만 낮추는 것이며 (덜 낮추는 것임), 1이면 속도의 6.25% 비율로만 낮추는 것임 (가장 적게 낮추는 것임). 03/0A번 제어모드에서는 낮추는 비율을 1/2만큼 작게 적용함. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.





명령어	구분	설명
	operation	모터 1/2의 적분오차값을 강제로 0으로 설정함.
;RIA55A; (+#^)	return value	RIA55A;
	comment	
	operation	모터 1의 적분오차값을 강제로 0으로 설정함.
;RIA5A5; (+#^)	return value	RIA5A5;
	comment	
	operation	모터 2의 적분오차값을 강제로 0으로 설정함.
;RI5A5A; (+#^)	return value	R15A5A;
	comment	
	operation	모터제어기를 리셋함.
	return value	N.A.
;RrA55A; (+#^)	comment	전원투입시의 초기화과정이 다시 시작됨. 2011년 12월 4일 S/W 버젼부터 사용가능한 명령임. precharge 릴레이가 있는 72V 이상의 고전압 H/W 버젼에서는 2012년 6월 30일 S/W 버젼부터 PWM출력을 먼저 disable 한 다음 100ms(단 자동화용과 BRAKE_ACTUATOR는 200ms, ELECTRONIC_POWER_STEER는 10ms) 후에 precharge 릴레이를 끄면서 전원투입시의 초기화과정이 다시 시작됨. 상위제어기가 RrA55A; 명령을 보내지 않더라도, 모터제어기가 스스로 재부팅을 하는 경우는 다음과 같음. PWM 출력이 disable되어 있고, 작동모드 00인 경우로서 (BRAKE_ACTUATOR ELECTRONIC_POWER_STEER)==1 일때는 ((fault_status&0x041d)!=0) ((fault1_status&0x0410)!=0) 이거나 상위제어기에서 CAN message를 사용하여 리셋 코드를 보내어 왔을때 전류 offset값을 저장하고, CANRXO_Buf[253]를 증가시키고 RrA55A; 실행. (BRAKE_ACTUATOR ELECTRONIC_POWER_STEER)==0 일때는 ((fault_status&0x041d)!=0) ((fault1_status&0x0400)!=0) 이면서 (operation_mode_SWITCH & 0x40000000L)!=0)&&(RJM_VER7_Hall_IC_analog==0) 이면 전류 offset값을 저장하고, CANRXO_Buf[253]를 증가시키고 RrA55A; 실행. 자동 재부팅후에 PE 명령을 자동으로 실행한 다음, CANRXO_Buf[253]!=0 이면 저장된 전류 offset 값으로 적용 전류offset 값을 설정하고 ZB 명령어로 설정하는 하위바이트의 값에 따라서 작동모드를 설정함.

명령어	구분	설명
		비상정지시의 정지 형태를 정의함.
		bit0~3 : 1이면 모터1이 비상정지시에 작동모드를 0으로 설정하여 정지함. (PWM은 켜있음)
		2이면 모터1이 So 명령어로 설정하는 감속률로 비상정지를 실시함.
		3이면 모터1이 PWM 출력을 끄는 방식으로 비상정지를 실시함.
• DCUI UI • (1#A)	operation	bit4~7 : 1/2/4이면 리밋스위치 LS1A/B/C 가 low 일때 모터1이 비상정지를 실시함.
;RSHLHL; (+#^)		bit8~11 : 1이면 모터2가 비상정지시에 작동모드를 0으로 설정하여 정지함. (PWM은 켜있음)
;RS?; (+#^)		2이면 모터2가 So 명령어로 설정하는 감속률로 비상정지를 실시함.
,no:, (T#)		3이면 모터2가 PWM 출력을 끄는 방식으로 비상정지를 실시함.
		bit12~15 : 1/2/4이면 리밋스위치 LS2A/B/C 가 low 일때 모터2가 비상정지를 실시함.
	return value	RSHLHL;
	comment	비상정지는 Rs 명령에 의하여서도 실시될수 있으며 리밋스위치 입력에 의하여서도 실시될수 있음.
		2012년 1월 19일 S/W 버젼부터 사용가능한 명령임.
	operation	모터1과 모터2가 명령에 의한 비상정지를 실행함.
;RsA55A; (+#^)	return value	RsA55A;
,πολοολ, (1 π)	comment	비상정지의 형태는 RSHLHL; 에 따름.
		2011년 12월 4일 S/W 버젼부터 사용가능한 명령임.
	operation	모터1이 명령에 의한 비상정지를 실행함.
;RsA5A5; (+#^)	return value	RsA5A5;
, 113AJAJ, (1 11)	comment	비상정지의 형태는 RSHLHL; 에 따름.
	comment	2011년 12월 4일 S/W 버젼부터 사용가능한 명령임.
	operation	모터2가 명령에 의한 비상정지를 실행함.
;Rs5A5A; (+#^)	return value	Rs5A5A;
, 1130AJA, (1 11)	comment	비상정지의 형태는 RSHLHL; 에 따름.
		2011년 12월 4일 S/W 버젼부터 사용가능한 명령임.



명령어	구분	설명		
		feedback 제어를 개시하도록 작동모드 설정에 필요한 조치를 자동으로 처리하게 함.		
		1st word : A55A or 0202: 전류제어기반의 위치제어모드(02번 제어모드) 작동에 필요한 과정을 처리함.		
		: 5AA5 or 0303 : 속도제어기반의 위치제어모드(03번 제어모드) 작동에 필요한 과정을 처리함.		
		: A5A5 or 0505 : 전류제어기반의 속도제어모드(05번 제어모드) 작동에 필요한 과정을 처리함.		
		: 5A5A or 0606 : PWM제어기반의 속도제어모드(06번 제어모드) 작동에 필요한 과정을 처리함.		
		: AA55 or 0707 : 전류제어모드(07번 제어모드) 작동에 필요한 과정을 진행함.		
	operation	: 55AA or 0A0A : 전류제어와 펄스주기계산법 기반의 속도제어모드(0A번 제어모드) 작동에		
	operation	필요한 과정을 진행함. (2011년 12월 30일 S/W 버젼부터 사용가능함.)		
		: 0101 : 위치제어모드(01번 제어모드) 작동에 필요한 과정을 진행함.		
		: 0404 : 속도제어모드(04번 제어모드) 작동에 필요한 과정을 진행함.		
		: 0808 : host direct angle control 제어모드(08번 제어모드) 작동에 필요한 과정을 진행함.		
;RBHLHL; (+#^)		: 0909 : host direct angle control 제어모드(09번 제어모드) 작동에 필요한 과정을 진행함.		
, I IOI ILI IL , (III)		: OEOE : 사인파 제어모드(03번 제어모드 사용) 작동에 필요한 과정을 진행함.		
;RB?; (+#^)		: OFOF : 사인파 제어모드(02번 제어모드 사용) 작동에 필요한 과정을 진행함.		
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	return value	RBHLHL;		
		1st word : 처리과정을 보여주는 카운터, 카운터가 0이면 처리가 끝났음을 의미함.		
		전원투입후로서 모터의 PWM이 켜지지 않았고 (PE 명령이 수행되지 않은 것과 같음), 작동모드도 00일때만		
		적용이 가능한 명령임.		
		처리과정에는 약간의 시간이 소요되며, 모터제어기에 따라서는 전류 offset 설정 및 초기 위치 설정등의		
		과정을 실시하기도 함.		
	comment	위상각을 알수 없는 analog형 위치센서를 가지고있는 경우는 모터를 움직이는 초기화 동작이 실시될수 있		
	Comment	음.		
		2011년 12월 15일 S/W 버젼부터 사용가능한 명령임.		
		2012년 4월 10일 S/W 버젼부터 부팅을 완료할때 상전류의 크기가 1A 이상이면 작동모드를 00으로 강제로		
		바꾸어서 모터의 작동을 차단하며, RB0101/0202/0303/0404/0505/0606/0707/0808/0909/0E0E/0F0F;를 사용할		
		수 있음. 일부 작동모드는 정상작동하지 않는 것이 있을수 있음.		



명령어	구분	설명	
	operation	fault code를 저장한 RAM buffer를 읽어서 fault의 내용을 문자로 보여주는 index를 가장 최근의 fault	
		code를 지정하도록 초기화함.	
		RJn <aaaaaaaa>;</aaaaaaaa>	
	return value	n : fault code가 저장된 RAM buffer의 index로서 가장 최근에 발생한 오류코드의 index. (0 - 7)	
		AAAAAAAAAA : n의 index에 저장된 fault code의 내용	
;RJI; (+#^)		RJ; 명령어로 한개를 읽으면 1씩 감소하여 다음번에 RJ; 명령을 보내면 이전의 fault의 내용을 문자로 보	
		여줌. 전원을 켠다음 처음으로 RJ;명령을 보내면 가장 최근의 fault 내용을 보여줌.	
	comment	오류코드가 전혀 없으면 <no fault=""> 가 보여짐.</no>	
	Comment	전원을 투입할때 오류코드저장용 RAM 버퍼는 0으로 clear 됨. 따라서 RJ 명령어로 보여지는 오류코드는	
		전원투입 이후에 발생한 것임.	
		2012년 8월 18일 S/W 버젼부터 사용가능한 명령임.	
	operation	fault code를 저장한 RAM buffer를 읽어서 fault의 내용을 문자로 보여줌.	
	return value	RJn <aaaaaaaa>;</aaaaaaaa>	
		n : fault code가 저장된 RAM buffer의 index. (0 - 7)	
		AAAAAAAAAA : n의 index에 저장된 fault code의 내용	
;RJ; (+#^)		RJ; 명령어로 한개를 읽으면 1씩 감소하여 다음번에 RJ; 명령을 보내면 이전의 fault의 내용을 문자로 보	
,110, (1π)		여줌. 전원을 켠다음 처음으로 RJ;명령을 보내면 가장 최근의 fault 내용을 보여줌.	
	commont	오류코드가 전혀 없으면 <no fault=""> 가 보여짐.</no>	
	comment	전원을 투입할때 오류코드저장용 RAM 버퍼는 0으로 clear 됨. 따라서 RJ 명령어로 보여지는 오류코드는	
		전원투입 이후에 발생한 것임.	
		2012년 8월 18일 S/W 버젼부터 사용가능한 명령임.	



명령어	구분	설명
	operation	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문자로 보여주는 index를 가장 최근의 fault code를 지정하도록 초기화함.
		Rjnold <aaaaaaaaa>; 또는 Rjnnew<aaaaaaaaaa>;</aaaaaaaaaa></aaaaaaaaa>
		n : fault code가 저장된 EEPROM buffer의 index로서 가장 최근에 발생한 오류코드의 index (0~14).
	return value	_old : 현재의 전원투입 이전에 발생한 오류코드임을 의미함.
		_new : 현재의 전원투입 이후에 발생한 오류코드임을 의미함.
;RjI; (+#^)		AAAAAAAAAAA : n의 index에 저장된 fault code의 내용
		Rj; 명령어로 한개를 읽으면 1씩 감소하여 다음번에 Rj; 명령을 보내면 그 이전의 fault의 내용을 문자로
		보여줌. 전원을 켠다음 처음으로 Rj;명령을 보내면 가장 최근의 fault 내용을 보여줌.
		오류코드가 전혀 없으면 <no fault=""> 가 보여짐.</no>
	comment	전원을 투입할때 오류코드저장용 EEPROM 버퍼는 0으로 clear 되지 않음. 따라서 Rj 명령어로 보여지는
		오류코드는 전원투입 이후와 이전을 포함하여 가장 최근의 오류코드임.
		2012년 8월 18일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 EEPROM에 저장된 값임.
	operation	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문자로 보여줌.
		Rjnold <aaaaaaaaa>; 또는 Rjnnew<aaaaaaaaaa>;</aaaaaaaaaa></aaaaaaaaa>
		n: fault code가 저장된 EEPROM buffer의 index (0~14).
	return value	_old : 현재의 전원투입 이전에 발생한 오류코드임을 의미함.
		_new : 현재의 전원투입 이후에 발생한 오류코드임을 의미함.
. 21 / . #		AAAAAAAAAA : n의 index에 저장된 fault code의 내용
;Rj; (+#^)		Rj; 명령어로 한개를 읽으면 1씩 감소하여 다음번에 Rj; 명령을 보내면 그 이전의 fault의 내용을 문자로
		보여줌. 전원을 켠다음 처음으로 Rj;명령을 보내면 가장 최근의 fault 내용을 보여줌.
		오류코드가 전혀 없으면 <no fault=""> 가 보여짐.</no>
	comment	전원을 투입할때 오류코드저장용 EEPROM 버퍼는 0으로 clear 되지 않음. 따라서 Rj 명령어로 보여지는
		오류코드는 전원투입 이후와 이전을 포함하여 가장 최근의 오류코드임.
		2012년 8월 18일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 EEPROM에 저장된 값임.



명령어	구분	설명
;XSHL; (+#^)	operation	순서제어에서 command를 설정하거나 제어상태를 읽어냄. (프로아텍 전용)
	return value	XSHLHLHLHL;
;XS?; (+#^)	comment	XS로 설정하는 값의 bit0/1/2/3은 리턴값의 bit8/9/10/11에 반영됨.
		프로아텍 초기 S/W 버젼에서만 유효한 명령임.

♣ XS?; 명령어로 리턴되는 32bit 파라메터의 비트별 의미는 다음과 같음.

32bit 리턴값	단자	기능	설정 주체
LWORD.bit0	LS1A	forward limit sensor (0 if turned ON)	forward limit sensor
LWORD.bit1	LS1B	backward limit sensor (0 if turned ON)	backward limit sensor
LWORD.bit2	LS1C	move command switch (forward if 0, backword if 1)	move command switch
LWORD.bit3	LS2A	jam sensor (0 if turned ON)	jam sensor
LWORD.bit4	LS2B	speed select switch (low speed if 1, high speed if 0)	speed select switch
LWORD.bit5	LS2C	recover command switch (0 if turned ON)	recover command switch
LWORD.bit8		move forward command bit (1 if forward move)	move command switch
LWORD.bit9		move backword command bit (1 if backward move)	move command switch
LWORD.bit12		recovering status (1 if recovering status)	motor controller
LWORD.bit13		error status (1 if error status)	motor controller
LWORD.bit14		move forward status bit (1 if forward move status)	motor controller
LWORD.bit15		move backword status bit (1 if backward move status)	motor controller
LWORD.bit29	LS1D/FAULT1	red lamp	motor controller
LWORD.bit30	LS2D/FAULT2	buzzer	motor controller



명령어	구분	설명	
		순서제어에서 사용하는 X_speed1_low, X_speed1_mid, X_speed1_high 를 설정함. (프로아텍 전용)	
		1st integer : X_speed1_low [-32768 ~ 32767] (1000 rpm or 50 rpm)	
;XViiii1,iiii2,i	operation	2nd integer : X_speed1_medium [-32768 ~ 32767] (1000 rpm or 50 rpm)	
iii3; (+#^)		3rd integer : X_speed1_high [-32768 ~ 32767] (1000 rpm or 50 rpm)	
1110, (1#)	return value	XVIIII1, IIII2, IIIII3;	
;XV?; (+#^)		이 값은 자동화 시퀜스제어에서 사용하는 값임.	
, , , , , , , , , , , , , , , , , , ,	comment	프로아텍 초기 S/W 버젼에서만 유효한 명령임.	
	Commone	RJM_VER/6의 경우에 기본값이 50임.	
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.	
		EXT_FAULT1_port와 EXT_FAULT2_port를 강제로 set 또는 clear 함.	
	operation	1st hexcode : EXT_FAULT1_port의 출력값 [0(0FF) ~ 1(0N)] (0)	
;X0HL,HL; (+#^)		2nd hexcode : EXT_FAULT2_port의 출력값 [0(0FF) ~ 1(0N)] (0)	
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	return value	X0HL,HL;	
;X0?; (+#^)		자동화용 제어기를 제외하고는 ZC 파라메터로 설정하는 값의 bit0/bit1이 1일 때만 XO 명령에 의한 강제설	
	comment	정이 가능함. ZC 파라메터로 설정하는 값의 bit0/bit1이 0일 때는, 주문자의 요청에 의하여 해당되는	
		EXT_FAULTX 단자에 +5V가 연결되어 있는 상태이며, 이때 해당 단자는 50mA 이내의 전류를 제어기 밖에서	
		사용할수 있는 전원단자의 역할을 함.	
		스테핑모터에서 50V 승압회로와 역기전력 해소 pass를 0N/OFF 함.	
		1st word : 0 : 50V 승압회로 OFF, 역기전력 해소 pass OFF	
;Xodddd1; (#^)	operation	1 : 50V 승압회로 ON, 역기전력 해소 pass OFF	
·Vo2: (#^)		2 : 50V 승압회로 OFF, 역기전력 해소 pass ON	
;Xo?; (#^)	return value	others : 50V 승압회로 OFF, 역기전력 해소 pass OFF Xoddddd1;	
	comment	RJM_VER5 스테핑모터 H/W 버젼에서만 설정이 가능함.	



명령어	구분	설명	
		380V 용 모터제어기에서 PRECHARGE_RELAY, MAIN_RELAY, BRAKE_RELAY를 강제 설정함.	
		IRAM136을 사용한 제어기의 경우	
		1st word : 1 : PRECHARGE_RELAY on, MAIN_RELAY off, BRAKE_RELAY off	
		2 : PRECHARGE_RELAY off, MAIN_RELAY on, BRAKE_RELAY off	
		3 : PRECHARGE_RELAY off, MAIN_RELAY off, BRAKE_RELAY on	
	operation	others: PRECHARGE_RELAY off, MAIN_RELAY off, BRAKE_RELAY off	
	operation	IXYS사의 CIB 모듈을 사용한 경우	
		1st word : 2 : MAIN_RELAY on, 나머지 off	
		3 : BRAKE_RELAY on, 나머지 off	
		4 : Phase_BRAKE_RELAY on, 나머지 off	
		5 : Cool_FAN on, 나머지 off	
		others: MAIN_RELAY off, BRAKE_RELAY off, Phase_BRAKE_RELAY off, Cool_FAN off	
;Xodddd1; (#^)		Xodddd1;	
		IRAM136을 사용한 제어기의 경우	
;Xo?; (#^)		1st word : 0 : PRECHARGE_RELAY off, MAIN_RELAY off, BRAKE_RELAY off	
		1 : PRECHARGE_RELAY on, MAIN_RELAY xxx, BRAKE_RELAY xxx	
		2 : PRECHARGE_RELAY off, MAIN_RELAY on, BRAKE_RELAY xxx	
		3 : PRECHARGE_RELAY off, MAIN_RELAY off, BRAKE_RELAY on	
	return value	4 : 기타의 on/off 조합인 경우	
		IXYS사의 CIB 모듈을 사용한 경우	
		1st word: 0: MAIN_RELAY off, BRAKE_RELAY off, Phase_BRAKE_RELAY off, Cool_FAN off	
		2 : MAIN_RELAY on, BRAKE_RELAY xxx, Phase_BRAKE_RELAY xxx, Cool_FAN xxx	
		3 : MAIN_RELAY off, BRAKE_RELAY on, Phase_BRAKE_RELAY xxx, Cool_FAN xxx	
		4 : MAIN_RELAY off, BRAKE_RELAY off, Phase_BRAKE_RELAY on, Cool_FAN xxx	
		5 : MAIN_RELAY off, BRAKE_RELAY off, Phase_BRAKE_RELAY off, Cool_FAN on	
		6 : 기타의 on/off 조합인 경우	
	comment	380V 용 BLDC 모터제어기에서만 설정이 가능함.	



명령어	구분	설명
		스테핑모터에서 DC-DC converter의 승압전압의 크기와 turn-off에 소요되는 시간을 설정함.
;Xedddd1,dddd2;	operation	1st word : DC-DC converter의 승압전압의 크기 [24 - 60] (48 V)
(+#^)		2nd word : DC-DC converter의 off time [1-50] (5 10ms)
	return value	Xedddd1,dddd2;
;Xe?; (+#^)	oommont	2012년 4월 10일 S/W 버젼부터는 DC-DC converter의 승압전압 크기 기본값이 49V로 바뀌었음.
	comment	♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		스테핑모터 또는 절대각엔코더를 사용하는 경우 엔코더의 4체배된 펄스수를 설정함.
;XEdddd1,dddd2;	operation	1st word : motor1의 1회전당 펄스수 [1-65535] or [1-2147483647]
or (+#^)		2nd word : motor2의 1회전당 펄스수 [1-65535] or [1-2147483647]
;XE 1, 2;	return value	XEdddd1,dddd2; or XEIIII1,IIII2;
(+#^)		28334/5 H/W 버젼에서는 long 변수가 할당되기 때문에 1-2147483647 범위의 설정이 가능함.
(π)	comment	2810/2811 H/W 버젼에서는 16비트 변수가 할당되기 때문에 1-65535 범위의 설정이 가능함.
;XE?; (+#^)		LOW_RESOLUTION_ENCODER를 사용하도록 하는 S/W 버젼에서는 부팅시에 XE 명령에 의한 파라메터값을 SE 명
, AL: , (#)		령에 의한 파라메터값으로 강제 설정함.
		♣ 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	시퀜스제어 프로그램의 현재 포인터 위치에 시퀜스제어 프로그램을 입력함.
		AAAAAAAAAA : 현재 포인터 위치에 저장할 시퀜스제어 프로그램 한 스텝.
;XPXAAAAAAA	return value	XPXAAAAAAAAAA;
AAA; (+#^)		SX 로 설정되는 값의 최상위 비트가 0 일 때만 입력이 가능함.
Λ·Λ· (π	comment	포인터의 위치는 입력된 시퀜스제어 프로그램 바로 다음으로 변경됨.
	Comment	시퀜스제어 프로그램은 최대 640 워드, 1280자 까지 입력이 가능함.
		설정된 프로그램은 XPsA55A; 명령어로 EEPROM에 저장될 수 있음.
	operation	시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제어 프로그램 한 스텝을 보여주고, 포인터 위치는
	operation	변경하지 않음.
;XP?; (+ # ^)	return value	XPAAAAAAAAAA;
,Λι:, ('π')	return varue	AAAAAAAAAA : 현재 포인터 위치에 있는 시퀜스제어 프로그램 한 스텝.
	comment	시퀜스제어 프로그램 한 스텝이란 마지막이 ; 로 끝나는 범위 까지를 한 스텝으로 봄.
		포인터의 값은 변하지 않음.
	operation	시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제어 프로그램 한 스텝을 보여주고, 포인터 위치를
		다음 스텝위치로 이동함.
;XP+; (+#^)	return value	XPAAAAAAAAAA;
		AAAAAAAAAA : 현재 포인터 위치에 있는 시퀜스제어 프로그램 한 스텝.
	comment	포인터의 값은 다음 스텝의 시퀜스제어 프로그램 위치로 이동함.
		시퀜스제어 프로그램의 포인터 위치를 강제 설정하고, 설정된 포인터 위치의 시퀜스제어 프로그램 한 스텝
	operation	을 보여주고, 포인터 위치는 변경하지 않음.
;XPdddd1; (+#^)		dddd1 : 시퀜스제어 프로그램 영역의 글자 위치 [0 - 1278] (0 th)
, Al dudd i , (i #)	return value	XPAAAAAAAAAA;
		AAAAAAAAAA : dddd1 번째 글자부터 시작하는 시퀜스제어 프로그램 한 스텝.
	comment	dddd1은 짝수만 의미가 있음.



명령어	구분	설명		
	operation	현재의 시퀜스제어 프로그램 포인터 위치를 보여줌.		
	roturn volus	XPpdddd1;		
;XPp; (+#^)	return value	1st word : X_character_pointer		
	comment	시퀜스제어 프로그램의 입력이나 보기에 사용하는 포인터로서 한 글자가 1의 값에 해당하며 항상 짝수임.		
	operation	메모리상의 시퀜스제어 프로그램을 EEPROM으로 저장함.		
;XPsA55A; (+#^)	return value	XPsA55A;		
, AFSAUUA, (T#)	comment	총 640 word의 사용자 프로그램 저장영역에 저장되며 저장위치는 TMS320F2810/11/12에서는 0x3f8700 ~		
	Collinett	0x3f897f 이며 TMS320F28334/5에서는 0xE700 ~ 0xE97f 임.		
	operation	현재의 PORT_STATUS_FOR_X_CONTROL와 X_PORT_STATUS를 보여줌.		
		XPTHLHL, HLHL;		
		bit0 : LS1A 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit1 : LS1B 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit2 : LS1C 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
	return value	bit3 : LS2A 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit4: LS2B 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit5 : LS2C 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
;XPT; (+#^)		bit6 : MOT_EN_N_port 0 이면 모터가 enable 된 상태이고 1이면 disable된 상태임.		
		bit7 : OFF12V_port 0 이면 charge pump가 작동중이며 1이면 off된 상태임.		
		bit8 : DISABLE1_IN_port 0 이면 L6207이 정상이며 1이면 fault 상태임.		
		bit9 : DISABLE2_IN_port 0 이면 L6207이 정상이며 1이면 fault 상태임.		
		bit10 : LS1D_port 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit11 : LS2D_port 0 이면 GND 단자와 short 상태이고 1이면 open 상태임.		
		bit12 : EXT_FAULT1_port 0 이면 출력이 OFF 된 상태이며 1이면 ON 상태임.		
		bit13 : EXT_FAULT2_port 0 이면 출력이 OFF 된 상태이며 1이면 ON 상태임.		
	comment	X_PORT_STATUS은 시퀜스제어 프로그램으로 조작한 PORT_STATUS_FOR_X_CONTROL 값에 해당함.		



명령어	구분	설명
	operation	현재의 X_timer0, X_timer0_period, X_counter0의 값을 보여줌.
		XPtHLHL,HLHL;
;XPt; (+#^)	return value	1st word : X_timer0
, AFL, (I#)	return varue	2nd word : X_timer0_period
		3rd word : X_counter0
	comment	
	operation	현재와 직전의 시퀜스제어 프로세스 카운트를 보여줌.
		XPPHLHL, HLHL;
;XPP; (+#^)	return value	1st word : X_process_cnt
, XII , (Im)		2nd word : X_process_cnt_old
	comment	시퀜스제어 프로세스 카운트는 어디까지 작동이 진행되었는지와 직전의 작동상태를 보여주는 값임.
		직전의 시퀜스제어 프로세스 카운트는 한스텝씩 프로그램을 실행시키는 모드에서만 유효함.
	operation	현재의 X_STATUS_FLAG를 보여줌.
;XPF; (+#^)	return value	XPFHLHL;
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		1st word : X_STATUS_FLAG
	comment	시퀜스제어 프로그램으로 조작한 X_STATUS_FLAG 값을 보여줌.
	operation	한스텝씩 시퀜스제어 프로그램을 실행시키는 모드일 때 한 스텝 씩 시퀜스제어 프로그램을 실행하게 함.
		XPSHLHL, HLHL;
;XPS; (+#^)	return value	1st word : X_process_cnt
7/1 07 (111)		2nd word : X_process_cnt_old
	comment	SX 명령어로 설정되는 작동모드 스위치의 MSB-1 비트가 1 일 때 한 스텝씩 프로그램을 실행시키는 모드임.
		여기서 한 스텝이란 시퀜스제어 프로세스 카운트값이 변화되는 구간을 한 스텝이라고 정의함.



명령어	구분	설명		
	operation	text의 현재 포인터 위치에 text를 저장하며 저장된 text를 보여줌.		
	return value	XTAAAAAA:AAAA;		
;XTAAAAAAAA	Teturn varue	AAAAAAAAAA : 현재 포인터 위치에 저장할 text 한 스텝.		
AA; (+#^)		총 240 character가 저장 가능함.		
Λ, (π)	comment	! 글자와 한글은 사용하면 안됨. (SPACE"#\$‰'()*{ }~ 및 숫자, 대소문자 가능함)		
	Comment	?,+,p,s 는 첫글자로 사용하면 안됨.		
		설정된 text은 XTsA55A; 명령어로 EEPROM에 저장될 수 있음.		
		text의 포인터 위치를 강제 설정하고 해당 위치에 있는 text 한 줄을 보여주고, 포인터 위치는 강제 설정		
	operation	된 값을 유지함.		
		dddd1 : text 영역의 글자 위치 [0 - 222(쓰기할때)/254(읽기할때)] (0 th)		
;XT?dddd1; (+#^)	return value	XTdddd1,AAAAAAAAA;		
		AAAAAAAAAA : dddd1 번째 글자부터의 text 한 스텝.		
	comment	dddd1은 짝수만 의미가 있음.		
		한 줄이란 마지막이 ; 로 끝나는 범위 까지를 한 줄로 봄.		
	operation	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고, 포인터 위치는 변경하지 않음.		
;XT?; (+ #^)	return value	XTAAAAAA;		
7/(1:7	Totalli valuo	AAAAAAAAAA : text의 현재 포인터 위치에 있는 text 한 스텝.		
	comment	포인터의 값은 변하지 않음.		
	operation	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고, 포인터 위치를 다음 줄 위치로 이동함.		
;XT+; (+#^)	return value	XTAAAAAA;		
7/11/9 (11/1	return varue	AAAAAAAAAA : text의 현재 포인터 위치에 있는 text 한 스텝.		
	comment	포인터의 값은 다음 줄의 text 위치로 이동함.		



명령어	구분	설명	
	operation	현재의 text 포인터 위치값를 보여줌.	
·VTn· (1#^)	roturn volue	XTpdddd1;	
;XTp; (+#^)	return value	dddd1 : 현재의 text 포인터 위치.	
	comment	text의 입력이나 보기에 사용하는 포인터로서 한 글자가 1의 값에 해당하며 항상 짝수임.	
	operation	메모리상의 text 를 EEPROM으로 저장함.	
;XTsA55A; (+#^)	return value	XTsA55A;	
, ATSADDA, (T#**)	comment	총 112 word의 TEXT 저장영역에 총 224글자가 저장되며 저장위치는 TMS320F2810/11/12에서는 0x3f8980 ~	
		0x3f89ef 이며 TMS320F28334/5에서는 0xE980 ~ 0xE9ef 임.	

명령어	구분	설명
2012년 12월 19일		0 ~ F 까지의 숫자로 지정된 위치의 FACTORY SETTING parameter RAM에 값을 저장함.
이전 버젼인 경우		iiii1 : 저장하고자하는 십진수의 값 [0 - 65535] (단 Z3A55A,HLHL; Z4A55A,HLHL; Z5A55A,HLHL;
;Z0iiii1; ~	operation	Z6A55A,HLHL; Z7A55A,HLHL; Z8A55A,HLHL; Z9A55A,HLHL; ZAA55A,HLHL; ZBA55A,HLHL; ZCA55A,HLHL; 는 16진수
ZFiiii1;		의 값임, A55A는 2012년 12월 19일 S/W 버젼부터 적용됨.)
(단 Z3HLHL;	return value	ZO(~F)HLHL, HLHL, HLHL;
Z4HLHL; Z5HLHL;	return varue	HLHL,HLHL,,HLHL : 16개의 FACTORY SETTING parameter값
Z6HLHL; Z7HLHL;		♣ ZO 로 설정되는 값은 전압측정의 offset calibration 값임. 예를 들어서 30V의 전압이 Sv 명령어로
Z8HLHL; Z9HLHL;		2990로 읽혀졌다면 ZO-10;으로 offset 값을 설정하고 ZsA55A; 명령어로 저장하면 됨. 이 값은
ZAHLHL; ZBHLHL;		-400~+400 까지만 유효하며 기본값은 0임.
ZCHLHL; 는 예외)		♣ Z1으로 설정되는 값은 제일메디컬인 경우 속도명령값을 강제 수정하는 경계전압값임. 예를 들어서
(+#^)		31V의 전압이하에서 속도명령값을 강제로 줄이고자 한다면 Z13100;으로 경계전압값을 설정하고 ZsA55A; 명
		령어로 저장하면 됨. 이 값은 2700~3300 까지만 유효하며 기본값은 3000임.
2012년 12월 19일		Z1으로 설정되는 값은 제일메디컬이 아닌 경우 AD 변환기의 scale factor이며, 16384 값일 때 실질적인
S/W 버젼부터		scale factor는 1로 적용됨. 14746 ~ 18022 범위의 값만 유효하며, 이 범위를 벗어나면 16384로 처리
;Z0A55A,iiii1; ~		됨. 이러한 AD 변환기의 scale factor 조정은 TMS320F28334/28335를 사용한 H/W 버젼에서는 실시하지
ZFA55A, iiiii1;		않음. 제일메디컬의 경우는 16384 값이 적용됨.
	comment	♣ Z2로 설정되는 값은 제일메디컬인 경우 속도명령값을 강제 수정하는 수정비율임. 예를 들어서 1V의
Z4A55A, HLHL;		전압이 강하될 때마다 1600 rpm을 감소시킨다면 Z216;으로 수정비율을 설정하고 ZsA55A; 명령어로 저장하
Z5A55A, HLHL;		면 됨. 이 값은 10~22 까지만 유효하며 기본값은 16임.
Z6A55A, HLHL;		🖧 Z3로 설정되는 값은 모터제어기에 설치된 전류감지 저항값임. 예를 들어서 2mΩ의 총/상전류감지저
Z7A55A,HLHL;		항이 모터1에, 10mΩ의 총/상전류감지저항이 모터2에 설치되어 있다면 Z30A02; 명령어로 설정하고 ZsA55A;
Z8A55A, HLHL;		명령어로 저장하면 됨. 저항값은 1~20, 21~50이 유효한 값이며, 이 범위를 벗어난 경우에는 2 의 값으
Z9A55A, HLHL;		로 처리됨. 값이 1~20 범위 일때는 mΩ의 단위이며, 21~50 범위 일때는 0.01mΩ 단위임. 이 값은 임의
ZAA55A, HLHL;		로 바꾸면 안됨. 상전류 측정에 IR21771을 사용하면서 총전류 측정에 AD8205를 사용하는 1채널 제어기
ZBA55A, HLHL;		(IRF7779를 사용하는 72V 제어기와 DC 전압을 사용하는 72V 자동화용 제어기) 에서는 상위 8비트로서 총전
ZCA55A,HLHL; 는		류 측정용 저항값의 크기를 설정하도록 하여야함.(2012년 11월 30일 S/W 버젼부터)
예외) (+#^)		



명령어	구분	설명
		♣ Z4로 설정된 값은 하드웨어 버젼 정보를 나타냄.
		♣ Z5로 설정된 값은 serial number임.
		🗘 Z6로 설정된 값의 least significant 4bit는 PWM 출력의 deadband 크기를 설정하는 것으로서, 예를 들
		어서 Z60002; 명령을 수행하면 PWM deadband 크기가 0.1333 X 2 uS (IGBT를 사용하는 고전압 H/W 버젼의
		경우는 0.2666 X 2 uS) 가 되도록 설정하는 것이며, ZsA55A;로 저장한다음 전원 재투입 이후부터 반영됨.
		저장시에 그 값이 F 이면 2로 대치하여 저장함. 톰슨사 또는 TI사의 intelligent power IC를 사용한 경
		우는 0, IR사의 모듈 MOSFET를 사용한 경우는 4, IGBT를 사용하는 경우는 4 이상, 기타의 경우에 기본값은 2임.
		♣ Z6의 상위 12비트는 사용가능한 최대전압값(단위는 V)을 의미함. 24V 모터의 경우는 30(01E), 36V
		모터의 경우는 45(02D), 48V 모터의 경우는 60(03C)임. 12(00C)보다 작거나 500(1F4)) 보다 크다면 기
		본값 30(01E)으로 설정됨. MOSFET의 gate를 off하는 전압강하량의 설정값(Sv 명령의 첫번째값)은 사용
		가능 최대전압의 1/2 보다 크지 않게 제한됨.
	comment	♣ Z7으로 설정되는 값은 모터구동 최대전류값 (단위는 A)를 의미함. 예를 들어서 모터1은 최대 14A,
		모터2는 최대 8A 라면 Z7080E; 명령어로 설정하고 ZsA55A; 명령어로 저장하면 됨.
		제품 출하시에 Z6/Z7 으로 저장된 값은 사용자가 임의로 바꾸면 안됨.
		♣ Z8으로 설정되는 하위바이트는 모터제어용 PWM 주파수를 설정하는 값으로서 10KHz의 배수를 의미함.
		예를 들어서 100KHz의 PWM을 사용한다면 0A 값이어야 하며, 20KHz라면 02 값이어야 함. 02에서 0A의 범
		위값만 허용함. RJM_VER7으로서 digital Hall 센서만을 사용하는 경우에는 80KHz 까지만 설정이 가능
		함. 제어로직을 수행하는데 CPU time이 부족하면 부팅시에 PWM 주파수를 낮추어서 적용하므로 실제로
		적용되는 PWM 주파수가 얼마인지를 확인하려면 SE 명령의 리턴값으로서 제일 마지막값을 보면 됨. 예를
		들어서 1770 이 리턴되었다면 120000 / 6000 = 20KHz 인데 6000은 1770의 십진수 값임.
		♣ Z8으로 설정되는 상위바이트는 모터제어기의 발열량계산에서 모터 stall시의 전류 ripple에 의하여 증
		가되는 발열량의 배율을 설정함. 예를 들어서 0(00)/16(10)/32(20)/48(30)/64(40)이면 ripple이 없을때
		의 발열량의 1/2/3/4/5배가 모터 stall시에 발열한다고 설정하는 것임. (2010년 9월 22일 S/W 버젼부터 적
		용)



명령어	구분	설명
		♣ Z9로 설정되는 최상위 4bit는 모터제어기의 발열이 허용치를 넘는 허용시간(sec)를 가리키며, 나머지
		12bit는 방열판에서 허용하는 연속 발열량(W)을 가리킴. 기본 방열판의 허용 연속 발열량은 초소형의
		경우(25W급 제어기) 3W, 소출력형의 경우(50W급 제어기) 6W, 중출력형의 경우(100W급 제어기) 9W, 고출력
		형의 경우(200W급 제어기) 12W 임. (2010년 9월 22일 S/W 버젼부터 적용)
		♣ ZA로 설정된 값의 하위 8비트는 brake 저항의 저항값 (Ohm)을 의미하며 0x00이나 0xFF이면 설치되어 있
		지 않음을 의미함.
		♣ ZA로 설정된 값의 상위 8비트는 제어기에 공급되는 전압의 크기를 측정하는 분할저항의 저항값 (K0hm)
		을 의미하며, 공급전압이 48V 보다 낮은 경우는 통상 20 이지만, 48V인 경우에는 30 KOhm을 사용하여야하
		며 H/W 에도 30K0hm이 조립되어야함. 공급전압이 48V보다 높은 제어기의 경우는 50K0hm(100V 제어기)
		또는 200K0hm(380V 제어기)이 사용되지만, 20 으로 설정하여도 기본값이 50 또는 200의 값이 적용됨.
		또 공급전압이 48V보다 높은 제어기의 경우는 제어용 전원을 외부에서 공급받거나 내부의 전원회로를 사용
		하여 제어전원을 만드는데, 이때 전압분할에는 H/W 상에서 반드시 20K0hm을 사용하여야함. 스테핑모터
		의 경우 2가지 전압을 감지하는 회로가 적용되는데 분할저항값을 똑같게 조립하여야함.
	comment	♣ ZB로 설정된 값의 하위 8비트가 0xE1 또는 0xB4이면 상위로부터의 펄스 입력에 따라서 위치명령값이 설
		정되는 모드로서 0xE1 일때는 01번/02번 위치제어모드를 사용하며, 0xB4 일때는 03번 위치제어모드를 사용
		함. 0xD2이면 상위로부터의 analog 입력에 따라서 속도명령값이 설정되는 모드(제어모드는 05)이며,
		0xA5이면 상위로부터의 analog 입력에 따라서 속도명령값이 설정되는 모드(제어모드는 0A)이며, 0xC3이면 상위로부터의 analog 입력에 따라서 전류명령값이 설정되는 모드이며, 그 외는 RS232C/RS485/CAN/USB/LAN
		등의 통신에 의한 명령으로 작동되는 모드임. (이 기능은 자동화용 H/W 버젼으로서 2012년 3월 15일 S/W
		당의 응선에 되면 당당으로 작용되는 포트함. (이 가능은 자용화용 11/W 비년으로서 2012년 3월 15일 5/W 버젼부터 적용됨. 0xA5 인 경우는 2012년 12월 30일 S/W 버젼부터 적용됨)
		라면구다 국용점, 0xA3 한 경구는 2012년 12월 30월 37W 미년구다 국용점7 ♣ ZB로 설정된 값에서 하위 8비트의 바로위 4비트 값이 A이면 servo ON 입력 또는 RB 명령어로 부팅할때
		08번 모드로 가장 가까운 초기위치를 설정하고 부팅하며, ZB로 설정된 값에서 하위 8비트의 바로위 4비트
		값이 0이면 servo ON 입력 또는 RB 명령어로 부팅할때 08번 모드로 -방향 끝에 가장 가까운 초기위치를 설
		정하고 부팅함. 또 PE/SM 명령으로만 부팅하는 경우는 특정위치에서 해야함.
		♣ ZC로 설정된 값에서 하위 비트(bit0)의 값이 0이면, 모터제어기 외부입력 터미널중에서 FAULT1 상태를
		출력하는 단자에 5V 전원 출력이 연결되어 있음을 의미하며 이 경우에 FAULT1 출력 기능을 사용할수 없음.
		또 ZC로 설정된 값에서 bit1의 값이 0이면, 모터제어기 외부입력 터미널중에서 FAULT2 상태를 출력하는 단
- KUBU		자에 5V 전원 출력이 연결되어 있음을 의미하며 이 경우에 FAULT2 출력 기능을 사용할수 없음.

명령어	구분	설명
	comment	제품 출하시에 Z3/Z4/Z5/Z6/Z7/Z8/Z9/ZA/ZB/ZC 으로 저장된 값은 사용자가 임의로 바꾸면 안됨.
		설정된 값은 ZsA55A; 명령어로 EEPROM에 저장될 수 있음. (주의! EsA55A 아님)



Zx 명령에 의한 factory setting 의 설정

TMS320	320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335				
	ZO (십진수)	Z1 (십진수)	Z2 (십진수)	Z3 (16진수)	
기능	전압측정의 offset calibration 값	속도명령값을 강제 수정하는 경 계전압값 (제일메디컬인 경우만 해당됨)	속도명령값을 강제 수정하는 수 정비율 (제일메디컬인 경우만 해 당됨)	모터제어기에 설치된 전류감지 저항값	
세부내역				하위 8bit : 모터1의 상전류 감 지 저항값 (단위는 mΩ 또는 값 이 20보다 클 경우는 0.01mΩ) 상위 8bit : 모터2의 상전류 감 지 저항값 (단위는 mΩ 또는 값 이 20보다 클 경우는 0.01mΩ), 단 상전류 측정에 IR21771을 사 용하면서 총전류 측정에 AD8205 를 사용하는 1채널 제어기의 경 우는 총전류감지 저항의 크기	
	Z4 (16진수)	Z5 (16진수)	Z6 (16진수)	27 (16진수)	
기능	하드웨어 버젼 정보	serial number	deadband 및 사용 최대 전압값	모터구동 최대전류값 (단위는 A)	
세부내역			최하위 4bit : PWM 출력의 deadband 크기 (단위 크기는 0.1333uS 또는 IGBT를 사용하는 고전압 H/W 버젼의 경우 0.2666uS) 상위 12비트 : 사용가능한 최대 전압값 (단위는 V)	하위 8bit : 모터1의 최대 허용 상전류값 (단위는 A) 상위 8bit : 모터2의 최대 허용 상전류값 (단위는 A)	



Zx 명령에 의한 factory setting 의 설정

TMS320	S320F2810/TMS320F2811/TMS320F2812/TMS320F28334/TMS320F28335					
	Z8 (16진수)	Z9 (16진수)	ZA (16진수)	ZB (16진수)		
기능	모터제어용 PWM 주파수와 모터 stall시 발열 파라메터	연속 발열량과 허용 시간	brake 저항값과 공급전압 측정용 전압분할 저항값	전원 투입시의 자동부팅 모드		
세부내역	하위 8bit : 모터제어용 PWM 주파수 를 설정값 상위 8bit : 모터 stall시의 전류 ripple에 의하여 증가되는 발열량의 배율	발열량(단위는 W) 상위 4bit : 방열판	하위 8bit : brake 저항의 저항값 (단위는 0hm)을 의미하며 0x00이나 0xFF이면 설치되어 있지 않음을 의미함. 상위 8bit : 제어기에 공급되는 전압의 크기를 측정하는 분할저항의 저항값 (단위는 K0hm)	하위 8bit : 0xE1/0xB4/0xD2/0xA5/0xC3 일때 각각 (01/02번_위치제어)/03번_위치제어/05번_속도제어 /0A번_속도제어/07번_전류제어 모드로 전원 투입시 자동 부팅됨. 이때 위치명령은 상위제어기로 부터의 펄스입력이 사용되며, 속도 및 전류명령은 상위제어기로 부터의 analog입력이 사용됨. 상위 4bit : A이면 servo ON 입력 또는 RB 명령어로 부팅할때 08번 모드로 가장 가까운 초기위치를설정하고 부팅하며, 0이면 servo ON 입력 또는 RB 명령어로 부팅할때 08번 모드로 -방향 끝에 가장가까운 초기위치를 설정하고 부팅함때 08번 모드로 -방향 끝에 가장가까운 초기위치를 설정하고 부팅함.		
	ZC (16진수)	ZD (십진수)	ZE (십진수)	ZF (십진수)		
기능	fault 출력 단자를 통한 +5V 전원 출력설정					
세부내역	bit0: 0이면 모터제어기 외부입력 터미널중에서 FAULT1 상태를 출력하 는 단자에 5V 전원 출력이 연결되 어 있음을 의미하며 이 경우에 FAULT1 출력 기능을 사용할수 없음. bit1: 0이면 모터제어기 외부입력 터미널중에서 FAULT2 상태를 출력하 는 단자에 5V 전원 출력이 연결되 어 있음을 의미하며 이 경우에 FAULT2 출력 기능을 사용할수 없음.					



FACTORY SETTING parameter의 설정

♣ FACTORY SETTING parameter는 EEPROM의 제일마지막 16 word에 저장이 되며 ZO~ZF 명령어로 개발 설정할수있고, Zr 명령어로 읽어 낼수있으며, ZsA55A; 명령어로 EEPROM에 저장할수 있음.

♣ 전압측정의 offset calibration 값의 설정 (제일메디컬의 경우)

ZO 로 설정되는 값은 전압측정의 offset calibration 값임. 예를 들어서 30V의 전압이 Sv 명령어로 2990로 읽혀졌다면 ZO-10;으로 offset 값을 설정하고 ZsA55A; 명령어로 저장하면 됨. 이 값은 -200~+200 까지만 유효하며 기본값은 0임.

♣ 속도명령값을 강제 수정하는 경계전압값의 설정

Z1으로 설정되는 값은 제일메디컬인 경우 속도명령값을 강제 수정하는 경계전압값임. 예를 들어서 31V의 전압이하에서 속도명령값을 강제로줄이고자 한다면 Z13100;으로 경계전압값을 설정하고 ZsA55A; 명령어로 저장하면 됨. 이 값은 2700~3300 까지만 유효하며 기본값은 3000임.

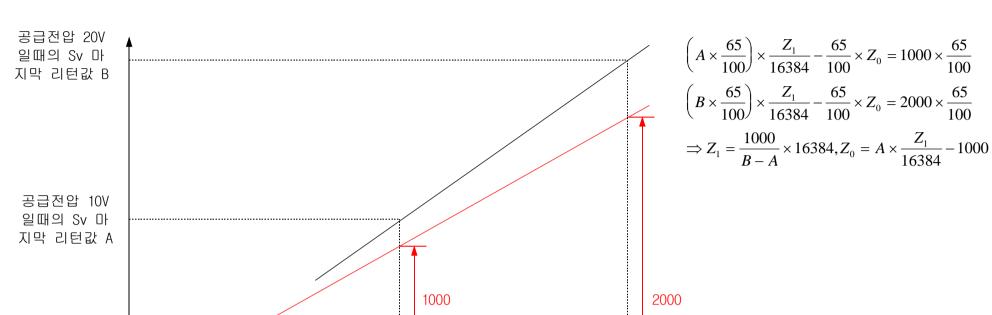
♣ 속도명령값을 강제 수정하는 수정비율의 설정

Z2로 설정되는 값은 제일메디컬인 경우 속도명령값을 강제 수정하는 수정비율임. 예를 들어서 1V의 전압이 강하될 때마다 1600 rpm을 감소시킨다면 Z216;으로 수정비율을 설정하고 ZsA55A; 명령어로 저장하면 됨. 이 값은 10~22 까지만 유효하며 기본값은 16임.

🗘 모터제어기에 설치된 전류감지 저항값의 설정

Z3로 설정되는 값은 모터제어기에 설치된 전류감지 저항값임. 예를 들어서 5mΩ의 전류감지저항이 모터1에, 10mΩ의 전류감지저항이 모터2에 설치되어 있다면 Z30A05; 명령어로 설정하고 ZsA55A; 명령어로 저장하면 됨. 이 값은 02,05,0A 만 유효함. 임의로 바꾸면 안됨. 같은 모터제어 채널이 있어서 총전류감지 저항과 상전류 감지 저항은 일치하여야함.





- ① Z0-1;Z1-1;ZsA55A;를 실행한 후, 전원을 껏다켜고
- ② Sv?;를 실행한 상태에서 10V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 A를 읽을것

공급전압 20V

③ Sv?;를 실행한 상태에서 20V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 B를 읽을것

공급전압 10V

- 4 Z1 parameter = 16384 * 1000 / (B A)
- ⑤ ZO parameter = (Z1 parameter * A / 16384) 1000
- 을 계산하여 ZO, Z1명령어를 사용하여 10진수로 입력한 다음
- ⑥ ZsA55A; 를 실행한 후, 전원을 껏다 켜고 Sv?; 명령어로 보정된 전압값 확인.

	명령어	구분	설명
		operation	16개의 FACTORY SETTING parameter 값을 읽어서 보여줌 (RAM 영역의 값).
	;Zr; (+#^)	return value	ZrHLHL, HLHL, HLHL;
:	,ZI , (T#)		HLHL,HLHL,,HLHL : 16개의 FACTORY SETTING parameter값
		comment	일곱번째값(Z6 명령어로 저장된 값)의 least significant 4bit의 값이 F 이면 2로 대치하여 저장함.
;Zs/		operation	메모리상의 FACTORY SETTING parameter 를 EEPROM으로 저장함.
	;ZsA55A; (+ #^)	return value	ZsA55A;
	,ZSADDA, (T#**)	comment	이 명령을 사용하면 FACTORY SETTING parameter 값이 변경되므로, 사용자는 특별한 사유가 없는 한 이 명
			령의 사용을 금함.

명령어	구분	설명
	operation	모터1의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1, Rth2, Tau1, Tau2)를 설정함.
;ZQdddd1,dddd2,d ddd3,dddd4,dddd5 ,dddd6,dddd7,ddd d8; (+#^)		ZQdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8; dddd1 : 권선저항(terminal resistance) (m0hm 단위) dddd2 : 권선인덕턴스(terminal inductance) (micro Henry 단위) dddd3 : 토크상수 (mNm/100A 단위, KOMOTEK 380V 경우는 mNm/10A) dddd4 : 역기전력상수 (micro Volt/RPM) dddd5 : 권선과 모터케이스 사이의 열전달저항 (0.001deg/W 단위) dddd6 : 모터케이스와 주변공기 사이의 열전달저항 (0.01deg/W 단위) dddd7 : 권선과 모터케이스 사이의 열전달 시정수 (0.1sec 단위) dddd8 : 모터케이스와 주변공기 사이의 열전달 시정수 (sec 단위)
;ZQ?; (+#^)	comment	powermax 30 200W 24V : 102, 16, 1350, 1408, 79, 530, 7, 848 4490 BS 200W 24V (Y) : 690, 220, 4186, 4384, 1350, 394, 290, 1756 4490 B 200W 24V (△) : 237, 76, 2383, 2495, 1350, 394, 290, 1756 3257 CR 24V: 1630, 270, 3770, 3950, 2000, 800, 170, 810 3863 C 24V: 620, 130, 3330, 3490, 1500, 600, 330, 843 EC45 FLAT 24V 50W : 978, 573, 3350, 3508, 4500, 425, 166, 212 KOMOTEK 380V용의 모터경우 토크상수의 단위는 mNm/10A 이므로 주의할 것. 설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	모터2의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1, Rth2, Tau1, Tau2)를 설정함.
;ZRdddd1,dddd2,d ddd3,dddd4,dddd5 ,dddd6,dddd7,ddd d8; (+#^)		ZRdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8; dddd1: 권선저항(terminal resistance) (m0hm 단위) dddd2: 권선인덕턴스(terminal inductance) (micro Henry 단위) dddd3: 토크상수 (mNm/100A 단위, KOMOTEK 380V 경우는 mNm/10A) dddd4: 역기전력상수 (micro Volt/RPM) dddd5: 권선과 모터케이스 사이의 열전달저항 (0.001deg/W 단위) dddd6: 모터케이스와 주변공기 사이의 열전달저항 (0.01deg/W 단위) dddd7: 권선과 모터케이스 사이의 열전달 시정수 (0.1sec 단위) dddd8: 모터케이스와 주변공기 사이의 열전달 시정수 (sec 단위)
;ZR?; (+#^)	comment	powermax 30 200W 24V : 102, 16, 1350, 1408, 79, 530, 7, 848 4490 BS 200W 24V (Y) : 690, 220, 4186, 4384, 1350, 394, 290, 1756 4490 B 200W 24V (△) : 237, 76, 2383, 2495, 1350, 394, 290, 1756 3257 CR 24V: 1630, 270, 3770, 3950, 2000, 800, 170, 810 3863 C 24V: 620, 130, 3330, 3490, 1500, 600, 330, 843 EC45 FLAT 24V 50W : 978, 573, 3350, 3508, 4500, 425, 166, 212 KOMOTEK 380V용의 모터경우 토크상수의 단위는 mNm/10A 이므로 주의할 것. 설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	모터1과 모터2의 최대허용 권선온도상승값을 설정함.
;ZTdddd1,dddd2;		ZTdddd1,dddd2;
(+#^)	return value	dddd1 : 모터1의 최대허용 권선온도상승값 (0.1도 단위)
(1π)		dddd2 : 모터2의 최대허용 권선온도상승값 (0.1도 단위)
;ZT?; (+#^)		온도 상승값은 주변온도를 기준으로 한 값인데, 주변온도는 0도로 고정되어 있음.
,Σι:, (ιπ /	comment	02/05/07번 작동모드에서만 정상적인 작동이 이루어짐.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
	operation	IRMCK201을 사용한 RJM_VER3에 있어서 모터1과 모터2의 nominal current(IRMCK201 파라메터 설정에 사용한
	operation	값)를 설정함.
		Zcdddd1,dddd2;
;Zcdddd1,dddd2;	return value	dddd1 : 모터1의 nominal current (0.01A 단위)
(+#^)		dddd2 : 모터2의 nominal current (0.01A 단위)
· · · · ·	comment	nominal current는 IRMCK201의 전류 scale factor를 정할때 사용하는 파라메터로서, 모터 카타로그상 연속
;Zc?; (+#^)		정격전류(RMS 값)인데, 100배의 값을 Zc 명령어로 설정하여야함.
,		※ IRMCK201의 작동파라메터를 다시 설정하기 어려울때 (설정 utility 프로그램이 없을때):
		QRE248,01; 의 리턴값 QRE248,01,efgh,ijlk; 을 읽어서 0xlkef 값을 뽑아내고
		27886044.6661633 / (상전류감지저항*0xlkef) 값을 계산하여 이 값을 Zc 명령어로 설정하면 됨.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
;ZPdddd1; (+#^)	operation	역기전력 해소용 전력저항의 최대 ON time을 설정함.
/_	return value	ZPdddd1;
;ZP?; (+ #^)		dddd1 : 역기전력 해소용 전력저항의 최대 ON time (10ms 단위)
	comment	설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	CAN 통신속도를 설정함.
		dddd1 : 통신속도설정값 [125,250,500,1000] (1000 Kbps)
	return value	Zbdddd1;
;Zbdddd1; (+#^)		전원 투입시에 DIP 스위치가 있는 경우에는 DIP 스위치 1번이 ON 되어 있을 때, DIP 스위치가 없는 경우에
,20uuu1, (+#)		는 SX 명령어로 설정되는 파라메터의 LSB가 0 일때 CAN 통신모드가 선택되며, 이 경우에 baud rate 설정
;Zb?; (+#^)		레지스터값을 바꾸어서 실제 통신속도를 바꿈. 설정값이 정당한 baud rate가 아니면 기본값 1000 Kbps
,ZU!, (T#**)	comment	로 설정됨.
		Zbdddd1; 명령어에 의하여 곧바로 CAN 통신속도가 바뀌는 것은 아니며, EsA55A; 명령어로 EEPROM에 저장한
		다음, 전원을 껐다 켜야지 바뀐 CAN 통신속도가 반영됨.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에 표시될 값의 주소를 설정함.
		1st word : 첫번째 변수의 주소
	operation	2nd word : 두번째 변수의 주소
		3rd word : 세번째 변수의 주소
		4th word : 네번째 변수의 주소
	return value	DZHLHL, HLHL, HLHL;
		표시될 값이 2-word 라도 첫번 word의 주소만 설정하면 됨.
		QRHLHL,HL; 명령어로 설정가능한 변수의 주소범위를 벗어나면 reset됨.
;DZHLHL,HLHL,HLH		설정가능한 주소범위는 다음과 같음.
L,HLHL; (+#^)		1st/2nd/3rd/4th : RAM address (TMS320F810/11/12)
Ε,ΠΕΠΕ, (Τπ)		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
;DZ?; (+#^)		[0880 - 783F] : peripheral memory [8000 - 9FFF] : LO/L1 RAM block (.ebss)
, oz:, (im /		[E000 - FFFF] : HO RAM block (.esysmem)
	comment	[E000 - FFFF]의 주소를 설정하면 실제로는 [3F8000 - 3F9FFF]의 주소에 있는 값을 지칭하도록 내부에서
		처리됨.
		1st/2nd/3rd/4th : RAM address (TMS320F28334/28335)
		[0400 - 05FF] : M1 RAM block (.stack) [0600 - 07FF] : M1 RAM block (.bss)
		[0880 - 793F] : peripheral memory
		[COOO - DFFF] : L4/L5 RAM block (.ebss)
		[E000 - FFFF] : L6/L7 RAM block (.esysmem)
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에 표시될 값의 type과 표시형식을 설정함.
		1st word : 첫번째 변수의 type과 표시형식
		2nd word : 두번째 변수의 type과 표시형식
		3rd word : 세번째 변수의 type과 표시형식
		4th word : 네번째 변수의 type과 표시형식
		각 word의
		상위바이트가 0이면 4자리 HexASKII로 값이 표시됨.
;DFHLHL,HLHL,HLH	operation	1이면 8자리 HexASKII로 값이 표시됨.
L,HLHL; (+#^)		2이면 5자리 unsigned integer로 값이 표시됨.
		3이면 6자리 signed integer로 값이 표시됨.
;DF?; (+#^)		4이면 7자리 unsigned long integer로 값이 표시됨.
		5이면 8자리 signed long integer로 값이 표시됨.
		6이면 9자리 double로 값이 표시됨.
		하위바이트가 0이면 표시대상 변수가 16bit 정수임.
		1이면 표시대상 변수가 32bit 정수임.
		2이면 표시대상 변수가 32bit 실수임.
	return value	DFHLHL, HLHL, HLHL;
	comment	표시되는 값은 4x20 영문 디스플레이의 11번부터 20번 column에 표시됨.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에 표시될 값의 scale factor를 설정함.
		1st double : 첫번째 변수의 scale factor
;DSfffff1,fffff2,f	operation	2nd double : 두번째 변수의 scale factor
fff3,ffff4; (+#^		3rd double : 세번째 변수의 scale factor
)		4th double : 네번째 변수의 scale factor
	return value	DSffff1,ffff2,ffff3,ffff4;
;DS?; (+#^)		DZ 명령어로 설정된 값에다 DS 명령어에 의한 scale factor를 곱한 값이 디스플레이에 표시됨.
	comment	원래의 값을 그대로 표시하려면 scale factor의 값으로 1.0을 설정하면 됨.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에 표시되는 값의 타이틀을 설정함.
	operation	1st word : 타이들의 번호 [0 - 3]
;DTdddd1,AAAAAA		next character : 최대 10자의 타이틀
AAA; (+#^)	return value	DTdddd1,AAAAAAAA;
		타이틀은 4x20 영문 디스플레이의 1번부터 10번 column에 표시됨.
;DT?dddd1; (+#^)	comment	설정된 값은 TEXT 저장영역의 마지막 40byte에 저장됨. (TMS320F2810/11/12에서는 0x3f89dc ~ 0x3f89ef 에
	Collinion	저장되며 TMS320F28334/5에서는 0xE9dc ~ 0xE9ef 에 저장됨)
		설정된 값은 XTsA55A; 명령어로 EEPROM에 저장될 수 있음. (주의! EsA55A; 아님)



명령어	구분	설명
		일정스텝으로 이동을 반복하는 데모작동의 파라메터를 설정함.
		1st word : set number_of_steps [1 - 20] (10 step)
;DDdddd1,dddd2,d	operation	2nd word : demo_step_size [1 - 60000] (40000/10000 pulse, 2012년 2월 4일 S/W 버젼부터는 100pulse가
ddd3; (+#^) or	operation	기본단위임)
;DDdddd1,dddd2,d		3rd word : demo_delay_time [1 - 100] (20 10ms)
ddd3,dddd4; (+#^		4th word : demo_offset (2 pulse) - 2011년 4월 15일 S/W 버젼부터만 적용됨.
)		DDddddd1,dddd2,dddd3; (2011년 4월 15일 이전 S/W 버젼) or DDddddd1,dddd2,dddd3,dddd4; (2011년 4월 15일
	return value	S/W 버젼부터) 2012년 2월 4일 S/W 버젼부터는 demo_step_size의 기본단위가 100 pulse 이므로 주의를
;DD?; (+#^)		요함. 특히 이전에 데모작동파라메터를 설정하여 사용하고 있었다면 에 100을 곱하여 스텝크기로 설정
		하므로 이전에 설정된 값을 100으로 나누어서 재설정하여야함.
	comment	설정된 값은 XTsA55A; 명령어로 EEPROM에 저장될 수 있음. (주의! EsA55A; 아님)
		일정스텝으로 이동을 반복하는 데모작동의 작동모드를 설정함.
		1st word : MSB(bit15) : 1이면 후진시에도 스텝동작을 실시함.
		bit14 : 전진중(0)인지 후진중(1)인지를 알려주는 상태비트로서 강제 설정이 되지 않음.
		bit13 : 1이면 LSxC 포트의 입력이 0일때 한 사이클의 데모작동이 개시됨.
	operation	bit12 : 1이면 데모작동이 개시될 때 bit7-bit0의 크기에 관계없이 저속/고속/스텝전후진
;DEHLHL; (+#^)		한 사이클만 수행하고 데모작동을 멈추며, 0이면 스텝전후진 한 사이클만 수행하고
		데모작동을 멈춤. 단, SX 명령어로 MSB를 세트한 경우에는 데모작동을 계속함.
;DE?; (+#^)		bit7-bit0 : 0이 아니면 데모작동을 개시하며 이 값의 횟수만큼 반복한 후에 0으로 자동
	return value	크리어됨. 단 bit12가 1일 경우에는 <mark>저속/고속/스텝전후진</mark> 한 사이클 데모만 실행
		하고 멈춤. 이 값을 0이 아닌값으로 다시 설정하면 데모작동을 다시 개시함.
		DEHLHL;
		1st word : bit14 : 전진중(0)인지 후진중(1)인지를 알려주는 상태비트임.
	comment	DE 명령어는 2012년 2월 4일 S/W 버젼부터만 적용됨.



명령어 :	구분	설명
		SX 명령어로 MSB를 세트하지 않더라도 DE 명령어로 bit13 또는 bit7-bit0 를 설정하여 데모작동을 개시할수 있음. 단 DE 명령어로 bit13 또는 bit7-bit0 를 사용한 경우는 정해진 횟수 또는 한사이클만 데모작동을 실시하고, 다시 트리거 되기를 기다리며, 데모작동이 개시될때 EXT_FAULT1_port(자동화용 H/W 버젼은 isolated 출력 0)를 HIGH가 되게 출력하며, 데모작동을 멈출때 EXT_FAULT1_port(자동화용 H/W 버젼은 isolated 출력 0)에 OV 가 약 100ms 출력되게 함. 이를 이용하면 두대의 모터제어기가 데모동작을 번갈 아 할수있음.
	comment	SX 명령어로 MSB를 세트할때 데모동작이 개시되며 SX 명령어로 MSB를 크리어할때까지는 데모작동을 계속함. 이때 DE0000; 이면 스텝전진/고속후진을 반복하며, DE8000; 이면 스텝전진/스텝후진을 반복하며, DE9000; 이면 저속/고속/스텝전후진을 반복함. SX 명령어로 MSB가 크리어 되어 있을때는 DE0002; 이면 스텝전진/고속후진을 2번 반복하고 정지하며, DE8002; 이면 스텝전진/스텝후진을 2번 반복하고 정지하며, DE9002; 이면 저속/고속/스텝전후진을 1번하고 정지함 (이 경우 마지막숫자는 0만 아니면 1인 경우와 같은 결과를 가짐). DE2000; 이면 LS1C 입력이 low가 될때 스텝전진/고속후진을 1번하고 정지하며, DEA000; 이면 LS1C 입력이 low가 될때 스텝전진/스텝후진을 1번하고 정지하며, DEA000; 이면 LS1C 입력이 low가 될때 스텝전진/스텝후진을 1번하고 정지하며, DE8000; 이면 LS1C 입력이 low가 될때 저속/고속/스텝전후진을 1번하고 정지함.
		SX 명령어로 MSB를 세트하고 EsA55A;로 저장을 하고난 다음, 재부팅하여 위치제어모드로 설정하여도 데모 작동이 개시되지 않음. (demo_cnt가 202로 고정됨) 따라서 SX 명령어로 데모를 개시하려고 하면 위치제 어모드로 설정하고난 다음 SX 명령어로 MSB를 세트하여야함. 중지하려면 SX 명령어로 MSB를 크리어하면 됨.
		두 모터제어기의 EXT_FAULT1_port(자동화용 H/W 버젼은 isolated 출력 0)와 LS1C 포트를 서로 cross 연결하고, 각 모터제어기에 DEB000; 을 설정한 다음 한쪽제어기의 LS1C 포트를 버튼스위치로 low로 강제 세트하면 두 모터제어기가 번갈아서 저속/고속/스텝전후진 작동을 실시함. 단 두 제어기의 전위차가 크면오류가 발생할수 있으므로 에러가 발생할때는 GT 명령어로 LS1C 포트의 on/off를 구별하는 threshold를 조정해볼 것.



명령어	구분	설명
;Dddddd1; (+#^)		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이는 모드에서 속도명령값 또는 전류명령값의
	operation	deadband 크기를 설정함.
		1st word : LSxC 포트의 A/D 변환값의 deadband 크기 [1 - 500] (16 bit)
;Dd?; (+#^)	return value	Dddddd1;
	comment	설정된 값은 XTsA55A; 명령어로 EEPROM에 저장될 수 있음. (주의! EsA55A; 아님)
		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이는 모드에서 속도명령값 또는 전류명령값이 0일때
		의 LSxC 포트 A/D 변환값을 설정함.
	operation	1st word : 모터1의 속도명령값 또는 전류명령값이 0이 되는 LS1C 포트의 A/D 변환값 [0 - 4095]
;DOA55A,dddd1,dd	operation	(기본값은 전원투입시의 LS1C 포트의 A/D 변환값)
dd2; (+#^)		2nd word : 모터 2의 속도명령값 또는 전류명령값이 0이 되는 LS2C 포트의 A/D 변환값 [0 - 4095]
		(기본값은 전원투입시의 LS2C 포트의 A/D 변환값)
;DO?; (+#^)	return value	D0dddd1,dddd2;
		기본값은 전원투입시의 LSxC 포트 A/D 변환값임. 이 값을 강제로 바꾸고자한다면 DOdddd1,dddd2; 명령
	comment	를 사용하여 바꿀수 있음. EEPROM에 저장되지는 않음.
		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이는 모드의 설정은 SX 명령어의 bit13을 참조.
		ON/OFF 출력과 7-seg 표기값을 설정하고 ON/OFF 입력과 switch 입력을 읽어서 host로 전송함.
	operation	1st word : 8-bit의 ON/OFF 출력 설정값
		2nd 24bit-word : 7-seg에 표시할 값 (이중에서 최상위 4-bit 1 character는 밝기 및 dot 0N/0FF 값임)
;DXHLHL,HLHLHL;		DXHLHL, HLHL, HLHL, HLHL;
(+#^)	return value	1st word : 8-bit의 ON/OFF 출력값
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\		2nd word : 7-seg의 표시값의 상위 3 character (이중에서 최상위 1 character는 밝기 및 dot ON/OFF 값
;DX?; (+#^)		임)
		3rd word : 7-seg의 표시값의 하위 2 character
		4th word : 8-bit의 ON/OFF 입력값
		5th word : DIP 스위치 값
	comment	자동화용 모터제어기에서만 쓸수 있는 명령어임.



명령어	구분	설명
		모터1의 step 구동을 위한 주기와 진폭을 설정하고 step 구동을 실행하도록 함.
	operation	1st word : motor1_period_of_step_response [0 - 2000] (0 ms)
		2nd word : motor1_amplitude_of_step_response [0 - 30000] (0 bit)
	return value	Tsdddd1,dddd2;
	Totalii valuo	1st word : 8의 배수로 재설정됨.
		위치/속도/전류제어모드에서 양방향으로 2nd word의 크기만큼 교대로, 반복적으로 step구동을 실시함.
;Tsdddd1,dddd2;		step구동을 중지하려면 1st word 값을 0으로 설정하면 됨.
(+#^)		1st word 값의 처음 1/8 동안은 구동명령이 0, 다음 1/4 동안은 구동명령이 +2nd word 값, 다음 1/4 동안
		은 구동명령이 0, 다음 1/4 동안은 구동명령이 -2nd word 값, 마지막 1/8 동안은 구동명령이 0임.
;Ts?; (+#^)		처음부터 과도한 크기의 2nd word 값을 설정하면 fault가 발생하거나, 제어전류의 크기가 전류리밋에 걸려
	comment	서 step 응답이 왜곡되므로 작은값 부터 시도하여야함.
		위치제어모드에서는 Qx93/97; 속도제어모드에서는 Qx92/96; 전류제어모드에서는 Qx91/95; 을 사용하여 1ms
		단위로 응답을 관찰할수 있음. 전류제어모드에서는 Qx2; 또는 Qx55/56; 를 사용하여 1ms단위의 전류제
		어 응답을 볼수 있으며, QxQx508/509; 와 Qx9;을 사용하여 전류제어 sample time 단위로 응답을 관찰할수
		도 있음.
		2012년 8월 19일 S/W 버젼부터 사용가능한 명령임.
		모터2의 step 구동을 위한 주기와 진폭을 설정하고 step 구동을 실행하도록 함.
;Ttdddd1,dddd2;	operation	1st word : motor2_period_of_step_response [0 - 2000] (0 ms)
(+#^)		2nd word : motor2_amplitude_of_step_response [0 - 30000] (0 bit)
(1π)	roturn volue	Ttdddd1,dddd2;
;Tt?; (+#^)	return value	1st word : 8의 배수로 재설정됨.
,ις, (ιπ /	comment	Ts 명령과 comment가 같음.
		2012년 8월 19일 S/W 버젼부터 사용가능한 명령임.



명령어	구분	설명
		모터1의 사인파 구동을 위한 주기와 진폭을 설정하고 사인파 구동을 실행하도록 함.
	operation	1st word : motor1_period_of_freq_response [0 - 2000] (0 ms)
		2nd word : motor1_amplitude_of_freq_response [0 - 30000] (0 bit)
	return value	Tudddd1,dddd2;
;Tudddd1,dddd2;		위치/속도/전류제어모드에서 2nd word의 크기에 해당하는 진폭으로 사인파 구동을 실시함.
(+#^)		사인파 구동을 중지하려면 1st word 값을 0으로 설정하면 됨.
(1111)		처음부터 과도한 크기의 2nd word 값을 설정하면 fault가 발생하거나, 제어전류의 크기가 전류리밋에 걸려
;Tu?; (+#^)		서 사인파 응답이 왜곡되므로 작은값 부터 시도하여야함.
, ια: , (ιπ)	comment	위치제어모드에서는 Qx93/97; 속도제어모드에서는 Qx92/96; 전류제어모드에서는 Qx91/95; 을 사용하여 1ms
		단위로 응답을 관찰할수 있음. 전류제어모드에서는 Qx2; 또는 Qx55/56; 를 사용하여 1ms단위의 전류제
		어 응답을 볼수 있으며, QxQx508/509; 와 Qx9;을 사용하여 전류제어 sample time 단위로 응답을 관찰할수
		도 있음.
		2012년 8월 19일 S/W 버젼부터 사용가능한 명령임.
	operation	모터2의 사인파 구동을 위한 주기와 진폭을 설정하고 사인파 구동을 실행하도록 함.
		1st word : motor2_period_of_freq_response [0 - 2000] (0 ms)
		2nd word : motor2_amplitude_of_freq_response [0 - 30000] (0 bit)
	return value	Tvdddd1,dddd2;
;Tvdddd1,dddd2;		위치/속도/전류제어모드에서 2nd word의 크기에 해당하는 진폭으로 사인파 구동을 실시함.
(+#^)		사인파 구동을 중지하려면 1st word 값을 0으로 설정하면 됨.
(1111)		처음부터 과도한 크기의 2nd word 값을 설정하면 fault가 발생하거나, 제어전류의 크기가 전류리밋에 걸려
;Tv?; (+#^)		서 사인파 응답이 왜곡되므로 작은값 부터 시도하여야함.
, IV:, (Im)	comment	위치제어모드에서는 Qx93/97; 속도제어모드에서는 Qx92/96; 전류제어모드에서는 Qx91/95; 을 사용하여 1ms
		단위로 응답을 관찰할수 있음. 전류제어모드에서는 Qx2; 또는 Qx55/56; 를 사용하여 1ms단위의 전류제
		어 응답을 볼수 있으며, QxQx508/509; 와 Qx9;을 사용하여 전류제어 sample time 단위로 응답을 관찰할수
		도 있음.
		2012년 8월 19일 S/W 버젼부터 사용가능한 명령임.



명령어	구분	설명
		온도측정용 센서 파라메터를 설정함.
	operation	1st word : constant (temp_C1)
	operation	2nd word : linear coefficient (temp_C2)
		3rd word : quadratic coefficient (temp_C3)
	return value	MTffff1,fffff2,ffff3;
		& 온도센서로는 NTC 온도센서를 사용하는 경우가 일반적이나, 특별한 경우 linear temprature 센서
		(TC1047)를 사용하는 경우가 있으므로 주의를 요함.
;MTffff1,ffff2,f		⇔ NTC 온도센서를 사용하는 경우 : 온도값은 Ohm 단위의 저항값 R을 사용하여 r = In(R)을 계산한 다음,
fff3; (+#^)		100000000.0 / {temp_C1 + temp_C2 * r + temp_C3 * r * r * r} - 23750 식을 이용하여 계산함. 계산
		된 온도값의 단위는 섭씨 0.01도 임.
;MT?; (+#^)		NTC 파라메터의 기본값은 temp_C1 = 1377.113100035604; temp_C2 = 234.029059587146; temp_C3 =
	comment	0.354431764337; 이며, 이는 -10도 55480 0hm, 25도 10000 0hm, 60도 2365 0hm 인 103k5(D20k) 모델의 파라메터에 해당함.
		다
		이용하여 온도값을 계산함. 계산된 온도값의 단위는 섭씨 0.01도 임. temp_C1의 기본값은 682.5로서
		0.5V에 해당하는 A/D 변환값이며, temp_C2의 기본값은 7.326으로서 1V당 100도 (0.01도 단위로는 10000)의
		값을 갖도록 하는 scale factor 값임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	anaration.	온도측정용 센서를 사용하여 측정된 저항치 및 온도값을 읽어내어 host로 전송함.
	operation	1st word : 온도측정을 실시하는 지의 여부 [0 ~ 1] (0 or 1)
		Mtdddd1,dddd2,iiii3;
	roturn volus	1st word : 온도측정을 실시하는 지의 여부 (0 or 1)
	return value	2nd word : NTC의 저항값 (0hm 단위) : NTC 온도센서를 사용하는 경우에 한함.
		3rd integer : 온도값 (0.01도 단위)
;Mtdddd1; (+#^)		♪ NTC 온도센서를 사용하는 경우 : NTC의 저항값은 전원투입후부터 계산을 실시함.
		온도값 계산은 TMS320F2809/TMS320F2810/11/12을 사용한 제어기로서 F/T 센서가 아닌 경우는 Mt1; 명령을
;Mt?; (+#^)		입력하여야만 개시되나, F/T 센서이거나 TMS320F28334/5를사용한 제어기의 경우는 전원투입후부터 자동 개
		시됨. 온도 측정을 멈추려면 MtO; 명령을 전송하면 됨.
	comment	NTC를 high-side에 위치시키고 low-side에 4.7K0hm 장착한 상태에서 3.3VA를 인가하여, NTC와 4.7K0hm에
		의하여 분할된 전압을 A/D 변환한 다음, NTC의 저항값을 계산함.
		측정된 NTC의 저항값과 MT 명령어로 설정된 파라메터를 사용하여, Steinhart-Hart 방정식으로부터 온도를
		계산함.
		♪ linear temprature 센서(TC1047)를 사용하는 경우 : 온도값계산을 항시 실시함.



명령어	구분	설명
	operation	온도측정용 두번째 센서의 파라메터를 설정함. 1st word : constant (temp2_C1) 2nd word : linear coefficient (temp2_C2) 3rd word : quadratic coefficient (temp2_C3) MUfffff1,fffff2,fffff3;
;MUfffff1,ffff2,f fff3; (+^) ;MU?; (+^)	comment	♣ 온도센서가 2개 장착되는 모델 (FT센서로서 외장형)에 한함. 이때 두번째 온도센서란 센서 연결용 screw terminal에서 가장 바깥에 위치한 온도 센서를 가리킴. 이 온도는 센서 연결용 screw terminal에서 가장 바깥에 위치한 오도 센서를 가리킴. 이 온도는 센서 연결용 screw terminal에서 가장 바깥에 위치한 strain-guage bridge의 온도 보상을 위하여 사용됨. ♣ 온도센서로는 NTC 온도센서를 사용하는 경우가 일반적이나, 특별한 경우 linear temprature 센서 (TC1047)를 사용하는 경우가 있으므로 주의를 요함. ♣ NTC 온도센서를 사용하는 경우: 온도값은 Ohm 단위의 저항값 R을 사용하여 r = In(R)을 계산한 다음, 100000000.0 / {temp2_C1 + temp2_C2 * r + temp2_C3 * r * r * r} - 23750 식을 이용하여 계산함. 계산된 온도값의 단위는 섭씨 0.01도 임. NTC 파라메터의 기본값은 temp2_C1 = 1377.113100035604; temp2_C2 = 234.029059587146; temp2_C3 = 0.354431764337; 이며, 이는 -10도 55480 Ohm, 25도 10000 Ohm, 60도 2365 Ohm 인 103k5(D20k) 모델의 파라메터에 해당함. ♣ linear temprature 센서(TC1047)를 사용하는 경우: (온도센서 A/D 변환값 - temp2_C1) * temp2_C2 식을 이용하여 온도값을 계산함. 계산된 온도값의 단위는 섭씨 0.01도 임. temp2_C1의 기본값은 682.5로서 0.5V에 해당하는 A/D 변환값이며, temp2_C2의 기본값은 7.326으로서 1V당 100도 (0.01도 단위로는 10000)의 값을 갖도록 하는 scale factor 값임.



명령어	구분	설명
	operation	온도측정용 두번째 센서를 사용하여 측정된 저항치 및 온도값을 읽어내어 host로 전송함.
		1st word : 온도측정을 실시하는 지의 여부 [0 ~ 1] (0 or 1)
		Mudddd1,dddd2,iiii3;
	return value	1st word : 온도측정을 실시하는 지의 여부 (0 or 1)
	Teturn varue	2nd word : NTC의 저항값 (0hm 단위) : NTC 온도센서를 사용하는 경우에 한함.
		3rd integer : 온도값 (0.01도 단위)
		♣ 온도센서가 2개 장착되는 모델 (FT센서로서 외장형)에 한함. 이때 두번째 온도센서란 센서 연결용
;Mudddd1; (+^)		screw terminal에서 가장 바깥에 위치한 온도 센서를 가리킴. 이 온도는 센서 연결용 screw terminal에
, wuddad i , (i)		서 가장 바깥에 위치한 strain-guage bridge의 온도 보상을 위하여 사용됨.
;Mu?; (+^)		♣ NTC 온도센서를 사용하는 경우 : NTC의 저항값은 전원투입후부터 계산을 실시함.
, wu: , (1)		온도값 계산은 TMS320F2809/TMS320F2810/11/12을 사용한 제어기로서 F/T 센서가 아닌 경우는 Mt1; 명령을
	comment	입력하여야만 개시되나, F/T 센서이거나 TMS320F28334/5를사용한 제어기의 경우는 전원투입후부터 자동 개
	Commert	시됨. 온도 측정을 멈추려면 MtO; 명령을 전송하면 됨.
		NTC를 high-side에 위치시키고 low-side에 4.7KOhm 장착한 상태에서 3.3VA를 인가하여, NTC와 4.7KOhm에
		의하여 분할된 전압을 A/D 변환한 다음, NTC의 저항값을 계산함.
		측정된 NTC의 저항값과 MT 명령어로 설정된 파라메터를 사용하여, Steinhart-Hart 방정식으로부터 온도를
		계산함.
		🕰 linear temprature 센서(TC1047)를 사용하는 경우 : 온도값계산을 항시 실시함.



온도센서 설정 과정

- 1. 전압 측정값의 calibration 실시
 - Z0-1;Z1-1;ZsA55A;를 실행한 후. 전원을 껏다켜고
 - Sv?;를 실행한 상태에서 10V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 A를 읽을것
 - Sv?;를 실행한 상태에서 20V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 B를 읽을것
 - Z1 parameter값 = 16384 * 1000 / (B A)
 - ZO parameter값 = (Z1 parameter값 * A / 16384) 1000
 - 을 계산하여 ZO. Z1명령어를 사용하여 부호있는 10진수로 입력한 다음
 - ZsA55A; 를 실행한 후
 - 전원을 껏다 켜고 Sv?; 명령어로 보정된 전압값 확인
- 2. 온도센서의 calibration 실시 (linear temperature sensor의 경우)
 - 첫번째 온도센서의 경우 Qx8;를 사용하여 네번째값인 온도값(0.01도 단위)을 보면서 MT 명령어의 첫번째 파라메터를 조정하여 현재 온도값과 같게 되게 맞추고
 - 두번째 온도센서의 경우 Qx22;를 사용하여 네번째값인 온도값(0.01도 단위)을 보면서 MU 명령어의 첫번째 파라메터를 조정하여 현재 온도값과 같게 되게 맞추고
 - EsA55A;로 저장
- 2. 온도센서의 calibration 실시 (NTC의 경우)
 - 다음 Matlab 코드를 이용하여 온도계산 파라메터를 계산함.

예를 들어서 -10℃에서 55480Ω, 25℃에서 10000Ω, 60℃에서 2365Ω 의 NTC라면

```
r = log(55480);
```

T1 = -10;

r2=log(10000);

T2=25;

r3 = log(2365);

T3=60;

a=[1 r1 r1*r1*r1;

1 r2 r2*r2*r2;

1 r3 r3*r3*r3]

b=[1/(T1+237.5);1/(T2+237.5);1/(T3+237.5)]

c=10000000*inv(a)*b

- 앞서 계산된 c 값 3개를 MT(첫번째 온도센서)/MU(두번째 온도센서) 명령어를 사용하여 입력하고
- EsA55A;로 저장



온도센서 설정 과정

- 3. 온도값의 확인
 - Q8;(첫번째 온도센서) 또는 Q22;(두번째 온도센서)를 사용하여 온도값의 정확도 확인

계산된 온도 값은 Mt(첫번째 온도센서), Mu(두번째 온도센서) 명령을 사용하여 읽어낼수 있음.

♣ 온도센서가 2개 연결가능한 외장형 F/T 센서인 경우에는, 단자대 마지막 단자에 연결된 온도센서가 두번째 온도 센서이며, 이보다 2개 단자 옆에 연결된 온도센서가 첫번째 온도센서임.



명령어	구분	설명
;MPffff1,ffff2,f		4개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용하여 X-moment, Y-moment, Z-force를 계산하는
fff3, ffff4, ffff5		변환행렬 값을 설정함.
,ffff6,ffff7,fff	operation	fffff1 - fffff4 : AD2FT11 - AD2FT14 (X-moment를 계산하기 위한 계수)
f8, ffff9, ffff10,		fffff5 - fffff8 : AD2FT21 - AD2FT24 (Y-moment를 계산하기 위한 계수)
ffff11, ffff12;		fffff9 - fffff12 : AD2FT31 - AD2FT34 (Z-force를 계산하기 위한 계수)
(+#^)	return value	MPffff1,ffff2,ffff3,ffff4,ffff5,ffff6,ffff7,ffff8,ffff9,ffff10,ffff11,ffff12;
(' '')		F/T 센서 회로에서만 유효한 명령임.
;MP? (+#^)	comment	4개의 계수는 순서대로 각각 bridge amplifier #3/#4/#1/#2의 전압값에 곱해짐.
7 Will . (1 II)		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
		6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용하여 X-moment, Y-moment, Z-force, X-force를
;Mpfffff1,ffff2,f		계산하는 변환행렬 값을 설정함. 2011년 11월 16일 S/W 버젼부터 사용가능한 명령임.
fff3,ffff4,ffff	operation	ffff1 - ffff2 : AD2FT15, AD2FT16 (X-moment를 계산하기 위한 계수)
5,ffff6,ffff7,ff	opor at ron	ffff3 - ffff4 : AD2FT25, AD2FT26 (Y-moment를 계산하기 위한 계수)
ff8,ffff9,ffff1		fffff5 - fffff6 : AD2FT35, AD2FT36 (Z-force를 계산하기 위한 계수)
0,ffff11,ffff12;		fffff7 - fffff12 : AD2FT41 - AD2FT46 (X-force를 계산하기 위한 계수)
(+#^)	return value	Mpffff1,ffff2,ffff3,ffff4,ffff5,ffff6,ffff7,ffff8,ffff9,ffff10,ffff11,ffff12;
	comment	6축 F/T 센서 회로에서만 유효한 명령임. (단 삼성메디컬의 경우 첫 4개의 파라메터가 보조모멘트 측정용)
;Mp? (+#^)		계수는 순서대로 각각 bridge amplifier #5/#6 또는 #3/#4/#1/#2/#5/#6의 전압값에 곱해짐.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
;Mqfffff1,ffff2,f		6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용하여 Y-force, Z-moment를 계산하는 변환행렬 값
fff3,ffff4,ffff	operation	을 설정함. 2011년 11월 16일 S/W 버젼부터 사용가능한 명령임.
5,ffff6,ffff7,ff	oper at ron	ffff1 - ffff6 : AD2FT51 - AD2FT56 (Y-force를 계산하기 위한 계수)
ff8,ffff9,ffff1		fffff7 - fffff12 : AD2FT61 - AD2FT66 (Z-moment를 계산하기 위한 계수)
0,ffff11,ffff12;	return value	Mqffff1,ffff2,ffff3,ffff4,ffff5,ffff6,ffff7,ffff8,ffff9,ffff10,ffff11,ffff12;
(+#^)		6축 F/T 센서 회로에서만 유효한 명령임.
	comment	계수는 순서대로 각각 bridge amplifier #3/#4/#1/#2/#5/#6의 전압값에 곱해짐.
;Mq? (+#^)		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		F/T 센서에서 TLV5630 또는 DAC124S085 D/A converter의 출력값을 설정함.
;MOdddd1,dddd2,d		1st word : channel #1의 D/A 출력값을 설정함. [0-4095] (1536 bit)
ddd3,dddd4; 또는		2nd word : channel #2의 D/A 출력값을 설정함. [0-4095] (1536 bit)
;MOdddd1,dddd2,d	operation	3rd word : channel #3의 D/A 출력값을 설정함. [0-4095] (1536 bit)
ddd3,dddd4,dddd		4th word : channel #4의 D/A 출력값을 설정함. [0-4095] (1536 bit)
5,dddd6; (+#^)		5th word : channel #5의 D/A 출력값을 설정함. [0-4095] (1536 bit) - 6축 F/T센서인 경우
		6th word : channel #6의 D/A 출력값을 설정함. [0-4095] (1536 bit) - 6축 F/T센서인 경우
;MO?; (+#^)	return value	MOddddd1,dddd2,dddd3,dddd4; 또는 MOdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
		F2809 내장형 또는 F2811 외장형으로 D/A 변환기가 장착된 F/T 센서 회로에서만 유효한 명령임.
(F/T센서의 경우)	comment	5번/6번째 값은 6축 F/T센서인 경우에만 해당됨.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		절대각센서를 사용하는 경우에 BLDC 모터의 Hall A/B/C상 신호와 모터위상각신호의 위상차를 없애기 위한
;MOiiii1,iiii2;	operation	위상 offset을 설정함.
(#)	operation	1st word : 모터1의 위상 offset [-32768 ~ 32767] (0 bit)
(#)		2nd word : 모터2의 위상 offset [-32768 ~ 32767] (0 bit)
;MO?; (#)	return value	MOiiii1,iiii2;
, WO: , (π)		F28334를 사용한 절대각센서를 사용한 모델에만 적용이 가능함. 모터를 손으로 서서히 돌리거나, 08번
(절대각센서를 사		모드(BLDC의 경우) 또는 07번 제어모드(DC 모터의 경우)로 낮은 속도로 모터를 구동하면서, Qx5; 명령어로
용하는 경우)	comment	Hall A/B/C상 신호와 모터위상각신호를 관찰하되, Hall A상 신호가 상승할때 모터위상각신호가 0이 되도록
0 VIL 0 T/		MO 명령을 사용하여 조정함.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
		F/T 센서에서 증폭후 A/D 변환된 스트레인 값의 온도증가에 따른 변화율을 설정함.
		1st integer : channel #1의 변화율. [-32768 ~ 32767]
		2nd integer : channel #2의 변화율. [-32768 ~ 32767]
;MSiiii1,iiii2,i	operation	3rd integer : channel #3의 변화율. [-32768 ~ 32767]
iii3,iiii4; 또는		4th integer : channel #4의 변화율. [-32768 ~ 32767]
;MSiiii1,iiii2,i		5th integer : channel #5의 변화율. [-32768 ~ 32767] - 6축 F/T센서인 경우
1113,11114,11115		6th integer : channel #6의 변화율. [-32768 ~ 32767] - 6축 F/T센서인 경우
,iiii6; (+ #^)	return value	MSiiii1,iiii2,iiii3,iiii4; 또는 MSiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;
		변화율은 (A/D 변환된 스트레인 값의 변화량) * 16384 / (0.01도 단위의 온도변화) 로 계산된 값이어야
;MS?; (+#^)		함.
	comment	온도증가량의 정의는 SI 명령어로 A/D 변환기 offset을 설정할때의 온도값을 기준으로한 변화량을 의미함.
		5번/6번째 값은 6축 F/T센서인 경우에만 해당됨.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.
		F/T 센서에서 온도보상의 기준온도와 온도보상 기울기를 자동 계산할 때 사용한 또 다른 온도값을 설정함.
	operation	1st integer : 온도보상의 기준온도 (0.01도 단위)
		2nd integer : 온도보상 기울기를 자동 계산할 때 사용한 또 다른 온도값 (0.01도 단위)
;MViiii1,iiii2;	return value	MViiii1,iiii2;
(+#^)		온도보상의 기준온도는 SIA55A; 명령에 의하여 앰프와 스트레인 게이지의 영점을 설정할때 자동으로 설정
		됨.
;MV?; (+#^)	comment	온도보상 기울기를 자동 계산할 때 사용한 또 다른 온도값은 SI5AA5; 명령에 의하여 자동으로 설정됨.
	Comment	MV 명령은 특별한 이유가 없는 한, 수동으로 값을 설정하는데 사용하지 마시오.
		2012년 3월 13일 S/W 버젼부터 사용가능한 명령임.
		설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



명령어	구분	설명
	operation	F/T 센서에서 수직력의 auto zero rate, maximum allowed drift, X 모멘트 coupling, Y 모멘트 coupling, FORCE_SENSOR_ERROR_DETECTION_TIME 파라메터를 설정함. 1st integer : FT_RETURN_HOME_RATE [0 ~ 32767] (10 0.1수직력단위크기/sec) 2nd integer : FT_DRIFT_ERROR_BOUND [0 ~ 32767] (200 수직력단위크기) 3rd integer : X_MOMENT_COUPLING [-32768 ~ 32767] (0 0.1% of 수직력) 4th integer : Y_MOMENT_COUPLING [-32768 ~ 32767] (0 0.1% of 수직력) 5th integer : FORCE_SENSOR_ERROR_DETECTION_TIME [0 ~ 65535] (2000 ms)
	return value	Moiiii1,iiii2,iiii3,iiii4,iiii5;
;Moiiii1,iiii2,i iii3,iiii4,iiii5 ; (+#^) ;Mo?; (+#^)	comment	♣ FT_RETURN_HOME_RATE는 기본값이 10 이며, 이는 1초에 수직력의 단위크기만큼 수직력의 크기를 줄인다는 뜻과 같음. 이렇게 함으로서 수직력에 drift 존재할 때 drift를 없애는 작용을 하도록 함. 이 작용은 drift 보상된 수직력의 크기가 두번째 파라메터인 FT_DRIFT_ERROR_BOUND 보다 작을 때만 실시됨. 단점으로는 수직력의 크기가 두번째 파라메터인 FT_DRIFT_ERROR_BOUND 보다 작은 상태로 시간이 지속되면, 수직력의 크기가 FT_RETURN_HOME_RATE의 비율로 줄어든다는 단점이 있음. 따라서 수직력이 return-to-zero의 특성을 가지고 있지 않을 때는 0이 아닌 값을 부여하는데 주의가 필요함. ♣ FT_DRIFT_ERROR_BOUND 는 FT_RETURN_HOME_RATE에 의하여 보정된 drift의 총량이 이 값을 초과할때 오류를 발생시키기 위한 값임. 단위는 수직력의 단위크기와 같음. drift 총량이 허용치를 초과하기를 FORCE_SENSOR_ERROR_DETECTION_TIME이상 경과하게 되면 수직력 및 모멘트의 값은 0으로 처리됨. 이 오류가 발생하면 23번 숫자를 램프로 flashing하며, Q2; 명령어의 리턴값에서 첫번째 워드의 bit6가 1로 세트됨. 오류에서 벗어나려면 SI 명령어로 offset 설정을 다시하면 됨. ♣ X/Y_MOMENT_COUPLING 값은 단위가 0.1% 이며, 수직력의 크기에 따라서 모멘트의 값이 변하는 경우 모멘트의 변화를 없애는데 사용하는 값임. 수직력크기 * (X/Y_MOMENT_COUPLING * 0.001) 만큼 모멘트의 크기를 뺌. ♣ 33번 숫자가 램프로 flashing하면 F/T센서의 4개 bridge amp 중 한개 이상이 비정상적인 값을 갖기를 FORCE_SENSOR_ERROR_DETECTION_TIME이상 경과한 것이며, 이 경우에도 수직력 및 모멘트의 값은 0으로 처리됨. 또한 Q2; 명령어의 리턴값에서 첫번째 워드의 bit7이 1로 세트됨. 정상범위로 값이 복귀되면 에 러상태는 정상작동상태로 자동복구됨. 설정된 값은 ES 명령어로 EEPROM에 저장될 수 있음.



F/T 센서 설정 과정

- 1. 전압 측정값의 calibration 실시
 - Z0-1;Z1-1;ZsA55A;를 실행한 후. 전원을 껏다켜고
 - Sv?;를 실행한 상태에서 10V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 A를 읽을것
 - Sv?;를 실행한 상태에서 20V 전압을 공급한 다음 (정확한 멀티메터로 제어기 전원공급단자에서 확인) Sv 마지막 리턴값 B를 읽을것
 - Z1 parameter값 = 16384 * 1000 / (B A)
 - ZO parameter값 = (Z1 parameter값 * A / 16384) 1000
 - 을 계산하여 ZO. Z1명령어를 사용하여 부호있는 10진수로 입력한 다음
 - ZsA55A; 를 실행한 후
 - 전원을 껏다 켜고 Sv?; 명령어로 보정된 전압값 확인
- 2. 온도센서의 calibration 실시 (linear temperature sensor의 경우)
 - Qx8;를 사용하여 네번째값인 온도값(0.01도 단위)을 보면서 MT 명령어의 첫번째 파라메터를 조정하여 현재 온도값과 같게 되게 맞추고 (온도센서가 2개 장착 가능한 외장형 모델인 경우는 단자대 마지막 단자 옆/옆의 단자에 온도센서를 연결해야함)
 - EsA55A;로 저장
- 2. 온도센서의 calibration 실시 (NTC의 경우)
 - 다음 Matlab 코드를 이용하여 온도계산 파라메터를 계산함.

예를 들어서 -10℃에서 55480Ω, 25℃에서 10000Ω, 60℃에서 2365Ω 의 NTC라면

```
r = log(55480);
```

T1 = -10;

r2 = log(10000);

T2=25;

r3=log(2365);

T3=60;

a=[1 r1 r1*r1*r1;

1 r2 r2*r2*r2;

1 r3 r3*r3*r3]

b=[1/(T1+237.5);1/(T2+237.5);1/(T3+237.5)]

c=1000000*inv(a)*b

- 앞서 계산된 c 값 3개를 MT 명령어를 사용하여 입력하고
- EsA55A;로 저장



F/T 센서 설정 과정

- 3. 온도값의 확인
 - Q8; 을 사용하여 온도값의 정확도 확인 (두번째값이 NTC 저항값(Ω 단위)이며 마지막 값이 온도값(0.01도 단위)임)
- 4. F/T센서의 calibration 실시
 - Qx3;를 사용하여 증폭된 전압값이 2000근방의 값이 되도록 MO 명령을 사용하여 offset을 조정함. (증폭된 전압값이 작으면 MO 명령어로 기본값 보다 큰 값을 입력하면 됨)
 - SI5000,256; 명령어를 사용하여 offset을 설정하고 (이때 F/T센서에는 5초간 힘이 인가되면 안되며, 온도는 일정하게 유지되야함)
 - -MS0,0,0,0;
 - 같은 온도를 유지하면서 아는 하중을 인가하여 Qx4;로 offset이 제거된 값을 기록하였다가 maxtix을 만든 다음 (이때 1번 채널을 다른 용도로 사용하는 H/W 버젼이라면 1번채널의 측정값(수직력1)은 2번채널의 측정값(수직력2)을 copy하여 사용할 것.)
 - MP 명령어로 matrix 값을 설정한 다음
 - FsA55A;로 저장
- 5. F/T 센서값의 확인
 - Qx5; 이용하여 아는 하중을 인가하면서 계산된 모멘트/force 값을 확인
- 6. F/T센서의 온도보상 실시
 - 좀더 높은 온도 (예 : 40도)를 유지하면서 Qx3;를 사용하여 증폭된 전압값을 기록
 - 이 값을 이용하여 MS 명령어로 보상값을 설정함
 - EsA55A;로 저장
- 7. F/T 센서값의 온도에 따른 확인
 - Qx5; 이용하여 계산된 모멘트/force 값을 온도를 변화시켜가면서 확인

계산된 온도 값은 Mt 명령을 사용하여 읽어낼수 있음.

계산된 F/T 값은 Qf; 명령을 사용하여 읽어내거나, Qb3; 명령을 사용하여 읽어 낼수 있음.



명령어	구분	설명
		삼성메디컬용 F/T 센서의 X-moment, Z-force coupling을 처리하는 파라메터를 설정함.
	operation	1st word : Z-force를 강제설정하는 X-moment의 크기 [0 - 30000] (1500 0.01Nm)
;Msdddd1,dddd2;		2nd word : X-moment가 클때 Z-force를 강제설정하는 Z-force 크기 [0 - 30000] (500 0.1N)
(+^)	return value	Msdddd1,dddd2;
		삼성메디컬에 적용되는 4-bridge F/T 센서에 한함. X-moment가 1st word로 설정하는 모멘트보다 크면
;Ms?; (+^)	oommon t	Z-force는 크기에 관계없이 2nd word로 설정되는 force로 강제 설정됨.
	comment	2011년 12월 19일 S/W 버젼부터 사용가능한 명령임.
		파라메터는 EsA55A; 명령으로 EEPROM에 저장이 가능함.



명령어	구분	설명
		자동화용 H/W 버젼에서 상위명령값과 실제값의 차이가 설정값이내에 들어왔는지를 판단하고 mission 완료
		를 결정하는 경계값(threshold)을 설정함.
	operation	1st word : 위치제어시의 threshold [1 ~ 32767] (100 pulse)
:MAdddd1,dddd2,d		2nd word : 속도제어시의 threshold [1 ~ 32767] (10 rpm)
ddd3; (+#^)		3rd word : 전류제어시의 threshold [1 ~ 32767] (50 0.01A)
Juuo, (T#)	return value	MAdddd1,dddd2,dddd3;
;MA?; (+#^)		상위명령값과 실제값의 차이가 설정값 이내에 들어오면 mission complete 출력을 ON 하며, 설정값의 2배
·WA: • (T#)		이상으로 커지면 mission complete 출력을 OFF 함.
	comment	속도나 전류제어의 경우는 명령값의 절대크기가 설정값의 2배 이내인 경우에 mission complete 출력을 0FF
		함.
		2012년 3월 22일 S/W 버젼부터 자동화용 H/W 버젼에서만 사용가능한 명령임.
	operation	BLDC 또는 AC servo 모터의 Iq, Id 값을 읽어서 host로 전송함.
		MCiiii1,iiii2,iiii3,iiii4;
:MC; (+#^)		1st integer : 모터1의 lq (bit)
(IIIO)	return value	2nd integer : 모터1의 ld (bit)
:MC?; (+#^)		3rd integer : 모터2의 Iq (bit)
, mo: , ('n')		4th integer : 모터2의 Id (bit)
	comment	단위크기의 의미는 QC 명령에 대한 리턴값의 의미와 같음.
	Commort	BLDC 또는 AC servo 모터에 한함. 2011년 8월 19일 S/W 버젼부터 사용가능한 명령임.
	operation	BLDC 또는 AC servo 모터의 Iq, Id 값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.
		Mciiii1,iiii2,iiii3,iiii4;
		1st integer : 모터1의 Iq (0.01A)
Mc; (+#^)	return value	2nd integer : 모터1의 Id (0.01A)
;Mc?; (+#^)		3rd integer : 모터2의 Iq (0.01A)
		4th integer : 모터2의 Id (0.01A)
	comment	BLDC 또는 AC servo 모터에 한함. 2011년 8월 19일 S/W 버젼부터 사용가능한 명령임.
		2012년 4월 10일 이전 S/W 버젼에서는 모터2의 전류스케일이 모터1의 전류스케일로 바뀌어져서 계산되는
Nooe Nooe		잘못이 있음. 337
CUBE —		აა <i>1</i>

명령어	구분	설명
;MRdddd1,dddd2,d ddd3,dddd4; (+#^	operation	전류상승률을 제한하는 파라메터를 설정함. 1st word: MOTOR1_CURRENT_RISING_TRIP [0 ~ 4914] (1638, 2m0hm을 전류감지저항으로 사용하는 경우는 20A) 2nd word: MOTOR1_CURRENT_RISING_INCREMENT [0 ~ 10000] (200, 2m0hm을 전류감지저항으로 사용하는 경우는 2442 A/s) 3rd word: MOTOR2_CURRENT_RISING_TRIP [0 ~ 4914] (1638, 2m0hm을 전류감지저항으로 사용하는 경우는 20A) 4th word: MOTOR2_CURRENT_RISING_INCREMENT [0 ~ 10000] (200, 2m0hm을 전류감지저항으로 사용하는 경우는 2442 A/s)
·MD2 · (+#^)	return value	MRdddd1,dddd2,dddd3,dddd4;
;MR?; (+#^)	comment	EPS가 아닌 경우에 한함. 2012년 10월 19일 S/W 버젼부터 적용됨. 1st/3rd 의 값이 1638이면 20A에 해당함. (2m0hm을 전류감지저항으로 사용하는 경우) 이 값보다 더 큰 전류명령값이 계산되는 경우 전류상승률제한 로직을 적용함. 2nd/4th word 의 값이 200이면 1ms당 2.442A의 상승을 허용한다는 말임. (2m0hm을 전류감지저항으로 사용하는 경우) 파라메터는 EsA55A; 명령으로 EEPROM에 저장이 가능함.



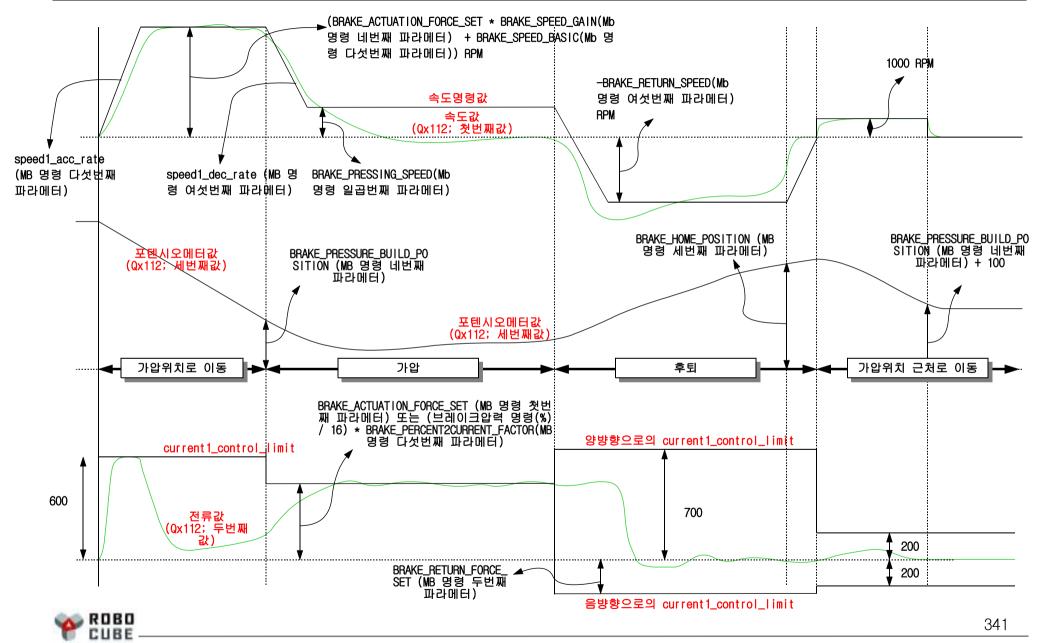
명령어	구분	설명
		brake actuator의 동작관련 파라메터를 설정함.
		1st word : 가압시의 전류설정값 [0 - 1000] (200 bit)
		2nd word : 복귀시의 전류설정값 [0 - 100] (50 bit)
		3rd word : 복귀위치의 포텐시오메터 값 [0 - 4095] (3200 bit)
		4th word : 가압개시위치의 포텐시오메터 값 [0 - 4095] (3050 bit)
		5th word : 가압시의 가속률 [1 - 60000] (6000 rpm/0.1sec)
	operation	6th word : 가압시의 감속률 [1 - 60000] (3000 rpm/0.1sec)
		7th word : %단위의 brake 압력 명령을 전류명령값으로 환산하는 scale factor [2*16 - 12*16]
		(7*16 bit, 2012년 7월 09일 S/W 버젼부터 적용)
		8th word : 마스터실린더 피스톤의 과도한 전진을 판단하는 위치값 [0 - 4095]
;MBdddd1,dddd2,d		(1300 bit, 2012년 7월 09일 S/W 버젼부터 적용)
ddd3, dddd4, dddd5		9th word : 마스터실린더 피스톤의 과도한 전진을 판단하는 위치직전의 구간크기
,dddd6,dddd7,ddd		[10 - 500] (100 bit,2012년 7월 17일 S/W 버젼부터 적용)
d8, dddd9; (#)	return value	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9;
		brake actuator에 한함. SV 또는 SVr 명령어로 +속도를 주면 가압이 개시됨. 가압이 개시되면 정상
;MB?; (#)		속도(가압시의 전류설정치에 따라서 다르게 설정됨)까지 가속하면서 전진하다가, 포텐시오메터의 가압개시
		위치에 도달하면 1000 RPM 까지 감속하면서, 가압시의 전류설정치를 전류상한치로 유지하면서 전진을 계속
		함. 이때 가압시의 전류설정치를 바꾸면 가압력이 그에 따라서 바뀜. SV 또는 SVr 명령어로 +속도를
		주기전에 가압시의 전류설정값을 바꾸면 가압된 결과압력값이 다르게 발생하는데, 100인 경우에 약
	comment	100bar, 200인 경우에 약 200bar, 250인 경우에 약 250bar가 발생함. 단 감속기의 정지마찰력 때문에
		정지상태의 최종 압력은 오차가 있음. SV 또는 SVr 명령어로 -속도를 주면 복귀가 개시됨. 복귀시의
		속도는 3000rpm이며 포텐시오메터의 복귀위치에 도달하면 정지함. 작동상태는 Qx2;로 전류치를, Qx1;으
		로 속도값을, Qx36로 포텐시오메터 변위를, Qx111;/Qx112;로 압력제어상태를 볼수 있음.
		피스톤의 위치가 8번째값에서 9번째값을 더한 위치에 도달하면 명령 압력값의 크기를 줄이기 시작하며, 8 법교가에 도당되면 201 된
		번째값에 도달하면 0이 됨.
		마지막 7개의 파라메터는 EsA55A; 명령으로 EEPROM에 저장이 가능함.



명령어	구분	설명
;Mbdddd1,dddd2,d ddd3,dddd4,dddd 5,dddd6,dddd7; (#) ;Mb?; (#)	operation	brake actuator의 동작관련 파라메터를 설정함. 1st word : CAN 통신이 끊어지고 비상정지가 ON 되어 있을때 비상정지 압력을 가하는 것을 enable(1), disable(0) 하는 파라메터값. 전원투입시의 기본값은 enable(1) 임. 2nd word : 브레이크 압력이 낮음을 판단하는 압력값 [0 - 200] (200 bar) 3rd word : 브레이크 압력이 낮음을 판단하는 피스톤 위치의 포텐시오메터 값 [0 - 4095] (1400 bit) 4th word : 가압시 압력설정값에 따른 속도증가비율 [0 - 50] (15) 5th word : 가압시 피스톤속도 기본값 [1 - 10000] (5000 rpm) 6th word : 리턴시 피스톤속도 [1 - 10000] (3000 rpm) 7th word : 브레이크 압력생성시기에서의 속도 [1 - 8000] (1000 rpm)
, MIO! , (#)	return value	Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd7;
	comment	brake actuator에 한함. CAN 통신이 끊어진 상태에서 비상정지 압력을 가하는 것이 disable 되어 있을때는, MB 명령어로 가압시의 전류설정값을 설정하고 SV 또는 SVr 명령어로 가압을 개시할수 있음.이 명령어는 2012년 7월 23일 S/W 버젼부터 적용됨.
;MEdddd1,dddd2,d ddd3,dddd4; (#)	operation	brake actuator의 동작관련 파라메터를 설정함. 1st word : CAN 통신이 끊어지고 비상정지가 ON 되기 까지의 지연시간 [10 - 65000] (5000 ms) 2nd word : 주전원이 차단되었다고 판단하는 경계전압 [1000 - 9000] (6500 0.01V) 3rd word : 주전원이 확실히 복구되었다고 판단하는 경계전압 [1000 - 10000] (7200 0.01V) 4th word : 주전원이 차단되었다고 판단한 후에 주전원 차단으로 최종 판정하기까지의 지연시간 [10 - 2000] (200 ms)
;ME?; (#)	return value	MEdddd1,dddd2,dddd3,dddd4;
	comment	brake actuator에 한함. CAN 통신이 끊어진 상태에서 비상정지 압력을 가하는 것이 disable 되어 있을때는, MB 명령어로 가압시의 전류설정값을 설정하고 SV 또는 SVr 명령어로 가압을 개시할수 있음.이 명령어는 2012년 8월 27일 S/W 버젼부터 적용됨.



Brake Actuator 작동 파형 (정상 작동시)



Qxdddd1 에 의하여 전송되는 변수값 (브레이크제어기의 경우)

	브레이크 제어기인 경우			
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
111/	speed1_RPM	current 1_AD	speed1_cmd_old / SPEED_SCALE_FACTOR	current1_control_limit
112/	speed1_RPM	current 1_AD	ADC_RESULT13_reg >> 4 (potentiometer)	ADC_RESULT12_reg >> 4 (pressure sensor)
115/	(UINT)(vfiltA[10] * 1.538461538) (공급전압)	brake_release_timer	ADC_RESULT13_reg >> 4 (potentiometer)	ADC_RESULT12_reg >> 4 (pressure sensor)
117/	ECU_fault_status	brake_engage_timer	ADC_RESULT13_reg >> 4 (potentiometer)	ADC_RESULT12_reg >> 4 (pressure sensor)

Brake Actuator 오류코드

오류코드	오류코드의 의미	발생원인	조치
0x8000	모터제어기에 구동 불가능한 fault가 발생하 였거나 정상 작동모드로 진입을 하지 못하고	모터제어기에 구동 불가능한 fault가 발생했을 경우	재부팅
UXOUUU	있다는 fault code	모터제어기에 구동 불가능한 fault는 없으나 정상작동 모드로 진입하지 못했을 경우	기다림
		브레이크 액이 부족할 경우	브레이크액 보충
0x4000	MB명령어 마지막에서 두번째 파라메터로 설정한 피스톤 마스터실린더 전진 위치보다 더 많이 전진했다는 fault code, 이 경우는 압력값	MB명령어 마지막에서 두번째 파라메터값이 너무 클 경우	MB명령어 마지막에서 두번째 파라메 터값을 작게 설정
	명령이 0으로 강제설정됨.	가압을 위한 전류 명령값이 너무 큰 경우	MB명령어 마지막에서 세번째 파라메 터값을 작게 설정
	MB명령어 마지막에서 두번째 파라메터에 마지	브레이크 액이 부족할 경우	브레이크액 보충
0x2000	막 파라메터를 더한 마스터실린더 피스톤 전 진 위치보다 더 많이 전진했다는 fault code, 이 경우는 압력값 명령이 전진변위량에 비례 적으로 줄어듦.	MB명령어 마지막 파라메터값이 너무 클 경우	MB명령어 마지막 파라메터값을 작게 설정
0x1000	정상 작동모드로 진입을 하지 못하고 있다는	부팅중이어서 정상작동 모드로 진입하지 못했을 경우	기다림
UX 1000	fault code	모터제어기에 구동 불가능한 fault가 발생했을 경우	재부팅
0x0800	과도하게 낮은 전압 상태가 되었음을 알리는 fault code	부하전류가 과도하게 흘러서 저전압 상태가 된 경우	Sw 명령어로 사용가능한 전류최대치 를 줄임
0x0400	모터의 코일 온도를 추정한 값이 과도하게 큰	부하전류가 과도하게 흘러서 모터의 온도가 과도하게 상승 함으로 추정될 때	Sw 명령어로 사용가능한 전류최대치 를 줄임
UXU4UU	값임을 알리는 fault code	모터 코일온도 추정관련 파라메터가 적절하지 않을 때	코일 온도추정 관련 ZQ 파라메터 수 정
0x0200	모터제어기의 구동소자 온도를 추정한 값이 과도하게 큰 값임을 알리는 fault code	부하전류가 과도하게 흘러서 모터제어기의 구동소자 온도가 과도하게 상승함으로 추정될 때	Sw 명령어로 사용가능한 전류최대치 를 줄임
	파포야계 단 없음을 될다는 Tault Code	모터 구동소자 온도 추정관련 파라메터가 적절하지 않을 때	관련 Z8/Z9 파라메터 수정
	피스톤이 정해진 위치까지 전진하였으나 설정	브레이크오일 부족이나 배관/마스터실린더 손상	배관/미스터실린더 점검
0x0080	되스폰이 성해진 위치까지 전전하였으나 설정된 압력보다 낮은 압력이 형성되었음	Mb명령어 두번째와 세번째 파라메터가 적절치 않을 때	Mb명령어 두번째와 세번째 파라메터 재설정



Brake Actuator 오류코드

오류코드	오류코드의 의미	발생원인	조치
0x0010	정상작동을 하던 중에 CAN 통신이 차단되어 브레이크를 비상정지하였다는 fault code	CAN 통신이 장시간 차단된 경우	너무 예민하다고 판단되면 ME 명령어 의 첫번째 파라메터를 크게함.
0x0020	브레이크가 가압되어 있는 상태에서, 정상작 동을 하던 중에, 공급전압이 현저히 떨어져서 주전원이 차단되었다고 판단하고, 브레이크를 후퇴시켰다는 fault code	브레이크가 가압되어 있는 상태에서 후퇴시키지 않고 주전 원을 차단한 경우	제어기가 손상이 되지 않은 경우는 무시하되, 손상이 되었다면 ME 명령 어의 2~4번 파라메터를 예민하게 보 호하도록 조정함.
	(0x0010 오류코드와 함께 0x0020 오류코드가 발생한 경우는 0x0020 오류코드에 의한 브레 이작동 해지가 우선함)	주전원 차단이 안된 상태에서 너무 예민하게 브레이크를 하 지	ME 명령어의 2~4번 파라메터를 덜 예 민하게 보호되도록 조정함.
0x0100	0x34번 메세지 브레이크압 설정값의 한도 초 과	CAN 통신 수신 데이터 오류	
0x0001	0x34번 메세지(braking command) 수신 time over	CAN 통신이 장시간 차단된 경우	
0x0008	0x36번 메세지(driving command) 수신 time over	CAN 통신이 장시간 차단된 경우	
0x0002	0x57번 메세지(steering status) 수신 time over	CAN 통신이 장시간 차단된 경우	
0x0004	0x58번 메세지(driving status) 수신 time over	CAN 통신이 장시간 차단된 경우	



명령어	구분	설명
;MBiiii1,iiii2,i	operation	EPS의 동작관련 파라메터를 설정함. 1st word : EPS_POSITION_SCALE_FACTOR [1 - 200] (85 pulse/0.1%) 2nd word : EPS_POSITION_REDUCTION_FACTOR [0 - 3000] (2730 0.00006105%/RPM) 3rd word : EPS_POSITION_REDUCTION_MIN [100 - 1000] (200 0.1%) 4th word : EPS_CENTER_OFFSET [-200 ~ 200] (0 0.1%) 5th word : EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE [0 ~ 100] (16) 6th word : MOTOR1_CURRENT_RISING_TRIP [0 ~ 4914] (2457, 0.5mOhm을 전류감지저항으로 사용하는 경우는 409) 7th word : MOTOR1_CURRENT_RISING_INCREMENT [0 ~ 10000] (10, 0.5mOhm. 전류감지저항으로 사용하는 경우는 2)
1113,11114,111115	return value	MBiiii1,iiii2,iiii3,iiii4,iiii5,iiii6,iiii7;
,iiii6,iiii7; (+#) ;MB?; (+#)	comment	EPS에 한함. 속도가 증가할수록 변위명령값의 최대치를 제한하는데, EPS_POSITION_REDUCTION_FACTOR 가 2730 이면 6000rpm 일때 변위명령값을 0%까지 제한하는 것에 해당하며, EPS_POSITION_REDUCTION_MIN 이 최소 제한값임. 작동상태는 Qx2;로 전류치를, Qx1;으로 속도값을, Qx112로 변위를 볼수 있음. 5th word 의 값이 16이면 모터구동전압이 10V 로부터 1V씩 떨어질때마다 10ms당 4(약 0.2A 에 해당함) 만큼의 크기로 전류리밋 값을 감소시킴. 6th word 의 값이 2457이면 20A에 해당함. (3m0hm을 전류감지저항으로 사용하는 경우) 이 값보다 더큰 전류명령값이 계산되는 경우 전류상승률제한 로직을 적용함. 만약 0.5m0hm을 전류감지저항으로 사용하는 경우는 409 값이 20A에 해당함. 7th word 의 값이 10이면 1ms당 81.4mA의 상승을 허용한다는 말임. (3m0hm을 전류감지저항으로 사용하는 경우) 만약 0.5m0hm. 전류감지저항으로 사용하는 경우는 2 값이 1ms당 97.7mA의 상승을 허용한다는 말임.



Qxdddd1 에 의하여 전송되는 변수값 (EPS의 경우)

	RJM_VER7 내장형 F/T 센서인 경우			
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수
112/	<pre>(position1_cmd - POSITION1_MID) / (MOTOR1_POSITION_SCALE_FACTOR * 4)</pre>	<pre>(position1_set - POSITION1_MID) / (MOTOR1_POSITION_SCALE_FACTOR * 4)</pre>	(position1 - POSITION1_MID) / (MOTOR1_POSITION_SCALE_FACTOR * 4)	vehicle_driving_motor_speed

EPS 위치명령 조작 및 보호 작동

위치 명령값의 설정 및 제한 :

MSG_ID = 0x058 번으로 수신된 구동모터속도값 (0.1 rpm 단위, 수신된 세번째 바이트가 high-byte, 네번째 바이트가 low-byte),
MSG_ID = 0x035 번으로 수신된 조향각 명령값 (-1000 ~ +1000, 수신된 세번째 바이트가 high-byte, 네번째 바이트가 low-byte)에 대하여

1000 - (모터구동속도(rpm) * EPS_POSITION_REDUCTION_FACTOR) / 16384 값에 대하여 EPS_POSITION_REDUCTION_MIN 보다 작지 않게 제한한 값을 상한선으로 설정한 다음. 조향각 명령값이 이 상한선보다 크지 않게 제한함.

그런 다음 EPS_CENTER_OFFSET을 더한 값을 최종적인 조향각 명령값으로 설정함. (EPS_CENTER_OFFSET의 단위와 0x035 번으로 수신된 조향각 명령값의 단위는 같으며, -1000 ~ +1000의 범위는 조향중립위치로부터 -100% ~ +100%의 변위를 나타냄)

그런 다음 이 변위값에 EPS_POSITION_SCALE_FACTOR 을 곱하고 5,000,000을 더하여 제어기 내부의 위치명령으로 설정함. EPS_POSITION_SCALE_FACTOR 값은 85를 넘을수 없으며, EPS_CENTER_OFFSET 값이 큰 경우에는 EPS_POSITION_SCALE_FACTOR 값은 더 작아야 조향의 기구리밋에 닿지 않게됨.

전류상승률이 높을때의 전류명령값 제한 :

전류명령값 (제어기 내부에서 위치제어시에 발생시킨 상전류진폭의 명령값, 0.5m0hm을 사용하는 12V 제어기의 경우에 409 가 20A에 해당하며, 3m0hm을 사용하는 72V 제어기의 경우는 2457 이 20A에 해당함) 이 EPS_CURRENT_RISING_TRIP 보다 크면, 1ms 마다 EPS_CURRENT_RISING_INCREMENT 이상의 전류명령값 상승을 억제함.

3m0hm을 전류감지저항으로 사용하는 경우에 EPS_CURRENT_RISING_INCREMENT값이 10 이면 1s당 81.4A의 상승을 허용한다는 말이며, 0.5m0hm. 전류감지저항으로 사용하는 경우는 EPS CURRENT RISING INCREMENT값이 2 이면 1s당 97.7A의 상승을 허용한다는 말임.



전압강하가 클때의 제어전류 상한값 제한 :

0.5mOhm을 사용하는 12V 제어기의 경우에만 적용되는 로직이며, EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE 값이 16이면 모터구동전압이 10V 로부터 1V씩 떨어질때마다 10ms당 4(약 0.2A 에 해당함) 만큼의 크기로 전류리밋 값을 감소시킴.

```
if((UINT)EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE > 100) EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE = 16;
if((tempW7 = ((DC BUS VOLT 1V * 10) - vfiltA[10])) > 0) {
   if((fault1 status&0x2900)==0) {
      if(current1_cmd>=0) current1_control_limit_cnt = current1_control_limit - current1_cmd;
      else current1_control_limit_cnt = current1_control_limit + current1_cmd;
      current1 control limit cnt -= current1 control limit >> 2;
      if(current1_control_limit_cnt < 0) current1_control_limit_cnt = 0;</pre>
   if((fault1 status&0x2000)!=0) {
      tempU3 = ((UINT)tempW7 * EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE) >> 8;
      if(current1 control limit cnt < current1 control limit) current1 control limit cnt += tempU3;
      if(current1_control_limit_cnt > current1_control_limit) current1_control_limit_cnt = current1_control_limit;
   fault1_status = 0x2000;
else if((fault1_status&0x2000)!=0) {
   tempU3 = ((UINT)(-tempW7) * EPS_CURRENT_CONTROL_LIMIT_UP_DOWN_RATE) >> 8;
   if(tempU2==1) tempU3 = 0;
   else if(tempU2==2) tempU3 = 0;
   if(current1_control_limit_cnt > tempU3) current1_control_limit_cnt -= tempU3;
   else {
      current1_control_limit_cnt = 0;
      fault1_status &= 0xdfff;
```



EPS 오류코드

오류코드	오류코드의 의미	발생원인	조치
	모터제어기에 구동 불가능한 fault	모터제어기에 구동 불가능한 fault가 발생했을 경우	재부팅
0x8000	가 발생하였거나 정상 작동모드로 진입을 하지 못하고 있다는 fault code	모터제어기에 구동 불가능한 fault는 없으나 정상작동 모드로 진입하지 못했을 경우	기다림
0,,4000	모터의 절대각센서에 의한 위치값과 incremental encoder에 의한 위치값	절대각 센서의 노이즈나 파손 등으로 인한 count 오류	system shutdown 하고 케이블 및 절대각 센서점검
0x4000	의 차이가 2048이상임을 알리는	절대각 센서의 읽기 오류	단발성이면 무시
	fault code	기어의 파손이나 절대각 센서의 비선형 오차	기구 및 절대각센서 점검
	모터의 incremental encoder의 펄스 카운트 오차가 발생하여 Hall A/B/C	incremental encoder의 노이즈 등으로 인한 count 오 류	system shutdown 하고 케이블 및 incremental encoder 점검
0x2000	상 신호와 60도 이상의 위상차가 발생한 경우 (이 경우는 0x8000 오류 코드도 발생함)	Hall A/B/C 센서의 읽기 오류	단발성이면 무시
0x1000	정상 작동모드로 진입을 하지 못하	부팅중이어서 정상작동 모드로 진입하지 못했을 경우	기다림
0.000	고 있다는 fault code	모터제어기에 구동 불가능한 fault가 발생했을 경우	재부팅
		부하전류가 과도하게 흘러서 저전압 상태가 된 경우	Sw 명령어로 사용가능한 전류최대치를 줄임
0x0800	과도하게 낮은 전압 상태가 되었음 을 알리는 fault code	저전압발생 억제 관련 파라메터가 적절하지 않을 때	MB 명령어 마지막 두 파라메터를 작게 설정함
		외부 72V->12V DC-DC 변환기에 오류가 발생한 경우	DC-DC 변환기 점검 및 교체
0x0400	모터의 코일 온도를 추정한 값이 과 도하게 큰 값임을 알리는 fault	부하전류가 과도하게 흘러서 모터의 온도가 과도하게 상승함으로 추정될 때	Sw 명령어로 사용가능한 전류최대치를 줄임
	code	모터 코일온도 추정관련 파라메터가 적절하지 않을 때	코일 온도추정 관련 ZQ 파라메터 수정
0x0200	모터제어기의 구동소자 온도를 추정 한 값이 과도하게 큰 값임을 알리는	부하전류가 과도하게 흘러서 모터제어기의 구동소자 온도가 과도하게 상승함으로 추정될 때	Sw 명령어로 사용가능한 전류최대치를 줄임
UXUZUU	fault code	모터 구동소자 온도 추정관련 파라메터가 적절하지 않 을 때	관련 Z8/Z9 파라메터 수정



명령어	구분	설명
		자석가동자 제어기의 동작관련 파라메터를 설정함.
		1st word : NUMBER_OF_S_POLE_MAGNETS [1 - 63] (19 개)
		2nd word : START_CONTROL_MAGNET_SENSOR_CNT [1 - 384] (19 번)
	operation	3rd word : STOP_CONTROL_MAGNET_SENSOR_CNT [1 - 384] (17 번)
	operation	4th word : SWITCHOVER_MAGNET_SENSOR2_CNT [1 - 384] (4 번)
		5th word : NUMBER_OF_POLE_PAIRS_BETWEEN_SENSOR12_A_PHASE [1 - 63] (18 pairs)
;MBdddd1,dddd2,d		6th word : NUMBER_OF_POLE_PAIRS_BETWEEN_SENSOR11_A_PHASE [1 - 63] (19 pairs)
ddd3, dddd4, dddd5		7th word : NS_MAGNET_OFFSET [0 - 600] (200 bit)
, dddd6, dddd7;	return value	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;
(#)		자석가동자 제어기에 한함. START_CONTROL_MAGNET_SENSOR_CNT 는 삼상 wave sector의 번호로서 전류제
(117)		어가 개시되는 위치의 번호임. START_CONTROL_MAGNET_SENSOR_CNT 는 삼상 wave sector의 번호로서 전류
;MB?; (#)		제어가 정지되는 위치의 번호임. SWITCHOVER_MAGNET_SENSOR2_CNT 는 -방향쪽 센서와 +방향쪽 센서의 사
, , , , , , , , , , , , , ,		용이 교대되는 삼상 wave sector의 번호임. NUMBER_OF_POLE_PAIRS_BETWEEN_SENSOR12_A_PHASE 는 -방향
	comment	쪽 센서와 +방향쪽 센서의 A상 신호검출 센서사이의 사잇거리를 NS 자석 one-pair 폭으로 나눈 값임.
		NUMBER_OF_POLE_PAIRS_BETWEEN_SENSOR11_A_PHASE 는 순위가 높은 제어기의 -방향쪽 센서와 순위가 낮은 제
		어기의 -방향쪽 센서의 A상 신호검출 센서 사잇거리를 NS 자석 one-pair 폭으로 나눈 값임.
		NS_MAGNET_OFFSET은 Hall A/B/C 센서를 A/D 변환하였을때 자석을 검출하지 않은 위치에서의 A/D 변환 값이
		2200 + NS_MAGNET_OFFSET 보다 확실히 작다고 보장할수 있는 최소크기의 값임.
		EsA55A; 명령으로 EEPROM에 저장이 가능함.
		자석가동자 제어기의 동작관련 파라메터를 설정함.
;Mbdddd1,		1st word : 삼상 아날로그 Hall 센서의 상전압 cross point 근방에서 smoothing하는 기능을 enable (1 일
(#)	operation	때), disable (0 일때). 전원투입시의 기본값은 enable(1) 임.
		2nd long word : 제어가능범위내로서 전류명령값의 부드러운 천이를 적용하는 범위의 크기 [1 - 655360]
;Mb?; (#)		(16384 단위위치값)
	return value	Mbdddd1, IIII2;
	comment	자석가동자 제어기에 한함.



명령어	구분	설명
		F/T 센서 drift offset 제거 동작관련 파라메터를 설정함.
;MBdddd1,dddd2;	operation	1st word : FTdataBufSize [1 - 512] (300 word)
(+^)		2nd word : FTcompensationThresh1 [1 - 10] (2 bit)
(+^)	return value	MBdddd1,dddd2;
;MB?; (+^)		F/T 센서에 한함. FTdataBufSize 만큼의 A/D 변환 데이터를 ring buffering 하면서
	comment	FTcompensationThresh1 값을 넘는 변화가 있는지를 사용하여 drift offset 제거 동작을 실시함.
		EsA55A; 명령으로 EEPROM에 저장이 가능함.



5th word : Hall A상 상승에지로부터 C상 상승에지까지의 위치값 [1 - 65535] (10924 bit) ;MBdddd1,dddd2,d 6th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (13655 bit)	명령어	구분	설명
peration label peration perat			엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터1의 경우, Hall 센서 A/B/C 상신호의 에
Operation Opera			지위치를 설정함.
SHDC 모터의 경우 8LDC 모터의 경우 8LDC 모터의 경우 1MBdddd1,dddd2,d ddd3,dddd4,dddd 5,dddd6; (+#^) 1MB7; (+#^) Comment 6UDC 모터의 경우 3rd word : Hall A상 상승에지로부터 B상 상승에지까지의 위치값 [1 - 65535] (5462 bit) 4th word : Hall A상 상승에지로부터 C상 상승에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (13655 bit) return value MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; 6H지간 간격이 모두 같은 이상적인 경우, 1st~6th word의 값은 00000,02731,05462,08193,10924,1365501며, 이때 1st~6th word의 값의 사이간격은 2731임. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 0x5; 모니터하여 A상 상승에지후 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			1st word : Hall A상 상승에지의 위치값 [00000] (00000 bit)
Strd word : Hall A상 상송에지로부터 B상 상송에지까지의 위치값 [1 - 65535] (5462 bit) 4th word : Hall A상 상송에지로부터 A상 하강에지까지의 위치값 [1 - 65535] (8193 bit) 5th word : Hall A상 상송에지로부터 C상 상송에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상송에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상송에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상송에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (13655 bit) return value MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; (에지간 간격이 모두 같은 이상적인 경우, 1st~6th word의 값은 00000,02731,05462,08193,10924,1365501며, 이때 1st~6th word의 값의 사이간격은 2731일. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall A상신호의 상송에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상송에지를 기준으로하여 A상신호의 다음 상송에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 0x5; 모니터하여 A상 상송에지후 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 갈만하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능한. analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;		operation	2nd word : Hall A상 상승에지로부터 C상 하강에지까지의 위치값 [1 - 65535] (2731 bit)
Sth word : Hall A상 상승에지로부터 C상 상승에지까지의 위치값 [1 - 65535] (10924 bit) 6th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (13655 bit) ddd3,ddd4,ddd5,dddd4,ddd5,dddd6; return value MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; mord		operation	3rd word : Hall A상 상승에지로부터 B상 상승에지까지의 위치값 [1 - 65535] (5462 bit)
:MBdddd1,dddd2,d ddd3,dddd4,dddd5 feturn value MBdddd1,dddd2,dddd3,dddd4,dddd5; 5,dddd6; (+#^) Comment Comment (+#^) Comme	BLDC 모터의 경우		
return value MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6; 5,dddd6; (+#^) (MR) 간 간격이 모두 같은 이상적인 경우, 1st~6th word의 값은 00000,02731,05462,08193,10924,13655이며, 이때 1st~6th word의 값의 사이간격은 2731임. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. Operation Operation Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			
5,dddd6; (+#^) WN 간 간격이 모두 같은 이상적인 경우, 1st~6th word의 값은 00000,02731,05462,08193,10924,13655이며, 이때 1st~6th word의 값의 사이간격은 2731임. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. **Operation** **Operation** **Operation** **Operation** **Obddd1,dddd2,dddd3,dddd4,dddd5,dddd6;* **Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;**			
OI때 1st~6th word의 값의 사이간격은 2731임. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. **Operation** Operation** Operation** Operation** Omega Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	ddd3,dddd4,dddd	return value	
Comment Comment A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. **Operation** **Operation** **Operation** **Operation** **Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;* **Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;* **Ab신호의 에지들의 위치를 숫자화한 것으로서, A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링한 것임. 구동을 하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. **Mbddd1,dddd2,dddd3,dddd4,dddd5,dddd6;**	5,dddd6; (+#^)		
comment 호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임. 구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			·
구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. operation analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	;MB?; (+#^)		
셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. operation analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;		comment	
후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			
고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함. analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			
operation analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			
Operation 기로 전송함. Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;			
Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;		operation	
13t WOLU - HALL NO 000M/NETI 00 000M/N/N/1 /101/			
BLDC 모터의 경우 2nd word : Hall C상 하강에지로부터 B상 상승에지까지의 시간폭(bit)	BLDC 모터의 경우		
return value 3rd word : Hall B상 상승에지로부터 A상 하강에지까지의 시간폭 (bit)	DCD0 TCI-1 81	return value	
;Mb?; (+#^) 4th word : Hall A상 하강에지로부터 C상 상승에지까지의 시간폭 (bit)	;Mb?; (+#^)	. Starii Varao	
5th word : Hall C상 상승에지로부터 B상 하강에지까지의 시간폭 (bit)			
6th word : Hall B상 하강에지로부터 A상 상승에지까지의 시간폭 (bit)			
PWM 신호의 한펄스 주기가 시간폭의 1 bit 에 해당함. 단 2채널 H/W 버젼의 경우는 PWM 신호의 두펄스			
comment 주기가 시간폭의 1 bit 에 해당함.		comment	
TUBE 352	CUBE	l	352

명령어	구분	설명
		엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터2의 경우, Hall 센서 A/B/C 상신호의 에
		지위치를 설정함.
		1st word : Hall A상 상승에지의 위치값 [00000] (00000 bit)
	operation	2nd word : Hall A상 상승에지로부터 C상 하강에지까지의 위치값 [1 - 65535] (2731 bit)
	орегаттоп	3rd word : Hall A상 상승에지로부터 B상 상승에지까지의 위치값 [1 - 65535] (5462 bit)
BLDC 모터의 경우		4th word : Hall A상 상승에지로부터 A상 하강에지까지의 위치값 [1 - 65535] (8193 bit)
		5th word : Hall A상 상승에지로부터 C상 상승에지까지의 위치값 [1 - 65535] (10924 bit)
;MEdddd1,dddd2,d		6th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (13655 bit)
,	return value	MEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
5,dddd6; (+#^)		에지간 간격이 모두 같은 이상적인 경우, 1st~6th word의 값은 00000,02731,05462,08193,10924,13655이며,
		이때 1st~6th word의 값의 사이간격은 2731임. 모터가 +방향으로 회전할때, 1st~6th word의 값은 Hall
;ME?; (+#^)		A상신호의 상승에지를 기준으로하여 이어진 Hall A/B/C 상신호 에지들의 위치를 숫자화한 것으로서, A상신
	comment	호의 상승에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임.
		구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧
		셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5; 모니터하여 A상 상승에지
		후 첫번째 펼스폭과 곧이은 A상 상승에지후의 첫번째 펼스폭 크기의 비율을 사용하여 감속된 속도 계산하
		고 이를 감안하여 다시 스케일링하여야함. EsA55A; 명령으로 EEPROM에 저장이 가능함.
	operation	analog Hall 센서를 사용하지 않는 BLDC 모터2의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어
-		기로 전송함. Medddd1,dddd2,dddd3,dddd4,dddd5,dddd6;
		medddd7,dddd2,dddd3,dddd3,dddd5, 1st word : Hall A상 상승에지로부터 C상 하강에지까지의 시간폭 (bit)
BLDC 모터의 경우		2nd word : Hall C상 하강에지로부터 B상 상승에지까지의 시간폭 (bit)
	return value	3rd word : Hall B상 상승에지로부터 A상 하강에지까지의 시간폭 (bit)
;Me?; (+#^)	return varue	4th word : Hall A상 하강에지로부터 C상 상승에지까지의 시간폭 (bit)
, mo. , (m)		5th word : Hall C상 상승에지로부터 B상 하강에지까지의 시간폭 (bit)
		6th word : Hall B상 하강에지로부터 A상 상승에지까지의 시간폭 (bit)
		PWM 신호의 한펄스 주기가 시간폭의 1 bit 에 해당함. 단 2채널 H/W 버젼의 경우는 PWM 신호의 두펄스
	comment	주기가 시간폭의 1 bit 에 해당함.
CUBE		353

명령어	구분	설명
		low resolution encoder를 사용하는 DC 모터1의 경우, 엔코더신호 A/B 상신호의 에지위치를 설정함.
		1st word : Hall A상 상승에지의 위치값 [00000] (00000 bit)
	operation	2nd word : Hall A상 상승에지로부터 B상 상승에지까지의 위치값 [1 - 65535] (4096 bit)
low resolution		3rd word : Hall A상 상승에지로부터 A상 하강에지까지의 위치값 [1 - 65535] (8192 bit)
encoder의 DC 모		4th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (12288 bit)
터인 경우	return value	MBdddd1,dddd2,dddd3,dddd4;
		에지간 간격이 모두 같은 이상적인 경우, 1st~4th word의 값은 00000,4096,8192,12288이며, 이때 1st~4th
;MBdddd1,dddd2,d		word의 값의 사이간격은 4096임. 모터가 +방향으로 회전할때, 1st~4th word의 값은 엔코더 A상신호의
ddd3, dddd4;		상승에지를 기준으로하여 이어진 엔코더 A/B 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승
(+#^)		에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임.
	comment	구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Mb?; 명령에 대한 리턴값을 사용하여, 덧
;MB?; (+#^)		셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx5;를 모니터하여 A상 상승에
		지후 첫번째 펄스폭과 곧이은 A상 상승에지후의 첫번째 펄스폭 크기의 비율을 사용하여 감속된 속도 계산
		하고 이를 감안하여 다시 스케일링하여야함.
		EsA55A; 명령으로 EEPROM에 저장이 가능함.
	operation	low resolution encoder를 사용하는 DC 모터1의 경우 엔코더신호의 에지간 간격값을 읽어서 상위제어기로
low resolution	operation	전송함.
encoder의 DC 모		Mbdddd1,dddd2,dddd3,dddd4;
터인 경우		1st word : Hall A상 상승에지로부터 B상 상승에지까지의 시간폭 (bit)
GC 6T	return value	2nd word : Hall B상 상승에지로부터 A상 하강에지까지의 시간폭 (bit)
;Mb?; (+#^)		3rd word : Hall A상 하강에지로부터 B상 하강에지까지의 시간폭 (bit)
7 WID: 7 (TT /		4th word : Hall B상 하강에지로부터 A상 상승에지까지의 시간폭 (bit)
	comment	PWM 신호의 한펄스 주기가 시간폭의 1 bit 에 해당함.



명령어	구분	설명
		low resolution encoder를 사용하는 DC 모터2의 경우, 엔코더신호 A/B 상신호의 에지위치를 설정함.
		1st word : Hall A상 상승에지의 위치값 [00000] (00000 bit)
	operation	2nd word : Hall A상 상승에지로부터 B상 상승에지까지의 위치값 [1 - 65535] (4096 bit)
low resolution		3rd word : Hall A상 상승에지로부터 A상 하강에지까지의 위치값 [1 - 65535] (8192 bit)
encoder의 DC 모		4th word : Hall A상 상승에지로부터 B상 하강에지까지의 위치값 [1 - 65535] (12288 bit)
터인 경우	return value	MEdddd1,dddd2,dddd3,dddd4;
		에지간 간격이 모두 같은 이상적인 경우, 1st~4th word의 값은 00000,4096,8192,12288이며, 이때 1st~4th
;MEdddd1,dddd2,d		word의 값의 사이간격은 4096임. 모터가 +방향으로 회전할때, 1st~4th word의 값은 엔코더 A상신호의
ddd3, dddd4;		상승에지를 기준으로하여 이어진 엔코더 A/B 상신호 에지들의 위치를 숫자화한 것으로서, A상신호의 상승
(+#^)		에지를 기준으로하여 A상신호의 다음 상승에지 위치가 16384 값에 해당하도록 스케일링한 것임.
	comment	구동을 하지 않고 +방향으로 무부하상태로 일정속도로 돌렸을때, Me?; 명령에 대한 리턴값을 사용하여, 덧
;ME?; (+#^)		셈 및 스케일링을 한 값으로 설정하면 됨. 단 속도값이 일정하지 않다면 Qx6;를 모니터하여 A상 상승에
		지후 첫번째 펄스폭과 곧이은 A상 상승에지후의 첫번째 펄스폭 크기의 비율을 사용하여 감속된 속도 계산
		하고 이를 감안하여 다시 스케일링하여야함.
		EsA55A; 명령으로 EEPROM에 저장이 가능함.
	operation	low resolution encoder를 사용하는 DC 모터2의 경우 엔코더신호의 에지간 간격값을 읽어서 상위제어기로
low resolution		전송함.
encoder의 DC 모		Medddd1,dddd2,dddd3,dddd4;
터인 경우		1st word : Hall A상 상승에지로부터 B상 상승에지까지의 시간폭 (bit)
GC 6T	return value	2nd word : Hall B상 상승에지로부터 A상 하강에지까지의 시간폭 (bit)
;Me?; (+#^)		3rd word : Hall A상 하강에지로부터 B상 하강에지까지의 시간폭 (bit)
, mo: , (m /		4th word : Hall B상 하강에지로부터 A상 상승에지까지의 시간폭 (bit)
	comment	PWM 신호의 한펄스 주기가 시간폭의 1 bit 에 해당함.



명령어	구분	설명
law racelution	operation	low resolution encoder를 사용하는 DC 모터 또는 BLDC 모터인 경우에 속도신호의 변화를 제한하는 범위를 설정함. 1st word : Ss 또는 Sa 명령에 따른 가속도 설정치에 대한 실제 가속도의 제한 배율 [2 - 1000] (100 배) 2nd word : Ss 또는 Sa 명령에 따른 가속도 설정치에 대한 실제 가속도의 제한 배율 [2 - 1000] (100 배)
	return value	MFdddd1,dddd2;
low resolution encoder의 DC 모 터 또는 BLDC 모 터인 경우 ;MFdddd1,dddd2; (+#^) ;MF?; (+#^)	comment	low resolution encoder를 사용하는 DC 모터 또는 digital Hall IC만을 사용하는 BLDC 모터인 경우에 실제속도를 계산할때 일정 크기 이상의 속도 변화를 허용하지 않는데, Ss 또는 Sa 명령에 따른 가속도 설정치의 몇배까지만 허용하는지를 설정하는 명령임. 위치제어일 경우에는 Ss 명령어로 설정된 가속도를 기준으로 하며, 속도제어일 경우에는 Sa 명령어로 설정된 가속도를 기준으로하여 제한 배율을 적용함. p 명령을 사용하여 실시간 제어를 하는 경우에는 제어중 있을수 있는 최대 가속도 값을 Ss 또는 Sa 명령으로 설정하여야함. 가속도의 제한 배율값이 너무 적으면, 엔코더 또는 Hall 센서신호의 에지간 간격이 균일하지 않은 경우에속도 측정 오차가 커짐. 이를 방지하려면 MB/ME 명령어를 사용하여 보전을 해주어야함. 가속도의 제한 배율값을 모터제어기에서 적용할때는 Ss 또는 Sa 명령에 따른 가속도 설정치가 클수록 가속도의 제한 배율값이 작게 적용되도록 하고 있음. 이 때 최소값은 2이며, 정지시에는 4가 넘지 않게 제한함. EsA55A; 명령으로 EEPROM에 저장이 가능함.



명령어	구분	설명
	operation	현재의 각변위를 홈위치로 설정함.
		VHiiii1,iiii2;
	return value	iiii1 : panning 방향의 홈위치에서의 각변위값
;VH; (+#^)		iiii2 : tilting 방향의 홈위치에서의 각변위값
	comment	한바퀴의 각변위값은 0~16383 범위의 값을 갖음.
		analog 포텐시오메터나 digtal 포텐시오메터를 사용하는 경우와 인공안구에서만 적용가능한 명령임.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
	operation	홈위치의 각변위값을 설정함.
		<mark>iiii1</mark> : panning 방향의 홈위치에서의 각변위값 [0 - 16383]
;\0iiii1,iiii2;		<mark>iiii2</mark> : tilting 방향의 홈위치에서의 각변위값 [0 - 16383]
(+#^)	return value	V0iiii1,iiii2;
	comment	analog 포텐시오메터나 digtal 포텐시오메터를 사용하는 경우와 인공안구에서만 적용가능한 명령임.
		설정된 값은 EsA55A; 명령어로 EEPROM에 저장될 수 있음.
	operation	현재의 각변위값을 읽어내어 host로 전송함.
	return value	VPIIII1,IIII2;
;VP; (+#^)		1 : panning 방향의 현재 각변위값
		IIII2 : tilting 방향의 현재 각변위값
	comment	analog 포텐시오메터나 digtal 포텐시오메터를 사용하는 경우와 인공안구에서만 적용가능한 명령임.
;VAiiii1,iiii2; (+#^)		각변위값을 설정하여 설정 각변위 위치로 이동시킴.
	operation	iiii1 : panning 방향의 각변위 설정값 [-1365 ~ 1365]
		<mark>iiii2</mark> : tilting 방향의 각변위 설정값 [-1365 ~ 1365]
	return value	VAIIII1,IIII2;
	comment	인공안구에서만 적용가능한 명령임.



명령어	구분	설명
;VTAAAAAAAA AA; (+#^)	operation	영상안정화 PCB로 명령어를 전송하고 결과값을 수신함.
		AAAAAAAAAA : 영상안정화 PCB로 전송될 명령어
	return value	VTAAAAAA;
		AAAAAAAAAA : 영상안정화 PCB에서 리턴된 값
	comment	인공안구에서만 적용가능한 명령임.
	operation	일정시간동안 panning 및 tilting 방향으로 각변위를 일으킴.
		1,2,3 : 안구를 아래로 숙임.
		7,8,9 : 안구를 위로 올림.
;1; (+#^)		1,4,7 : 안구를 좌로 움직임.
;2; (+#^)		3,6,9 : 안구를 우로 움직임.
;3; (+#^)		1;
;4; (+#^)	return value	2;
;6; (+ #^)		3;
;7; (+ #^)		4;
;8; (+#^)		6;
;9; (+#^)		7;
		8;
		9;
	comment	인공안구에서만 적용가능한 명령임.



명령어	구분	설명
;ERHLHL,HLHL;	operation	파라메터 값을 EEPROM으로부터 읽어내어 DSP SARAM에 저장함. (EEPROM으로부터 읽은 값을 LF2406A의 경우
		는 0x0A00 - 0x0FFF, F2811의 경우는 0x3F8200 - 0x3F89FF, F28334의 경우는 0xE200 - 0xE9FF 번지에 저장
		함.)
		1st word : 파라메터를 EEPROM으로부터 읽어서 저장하는 SARAM의 시작주소 [0x0A00 - 0x0FFF/0x11FF]
(25LC040/080/160		2nd word : word 수 [0x0001 - 0x0600/0x0800]
(23LC040/080/180 /320/320A만 해 당) (+#^%)	return value	ERHLHL; HLHL;
	comment	EEPROM의 0번지(low byte)/1번지(high byte) 데이터가 SARAM의 0x0A00/0x3F8200/0xE200 번지로 읽혀져 저
37 (111 10)		장됨.
		마지막에 세미콜론 대신 콜론이 echo되면 시작주소나 word수가 not valid인 경우임.
		F2811 또는 F28334를 사용할때 25LC320/320A인 경우 2K word까지 사용이 가능하며, 시작주소/word 수가
		0x11FF/0x0800 으로 늘어남.
	operation	DSP SRAM에 있는 파라메터 값을 page단위(8 word 또는 16 byte)로 EEPROM에 저장함. (LF2406A의 경우는
		0x0A00 - 0x0FFF, F2811의 경우는 0x3F8200 - 0x3F89FF, F28334의 경우는 0xE200 - 0xE9FF 번지에 저장함)
		1st word : 파라메터를 EEPROM으로 저장하는 SARAM의 시작주소 [0x0A00 - 0x0FF8/0x11FF]
		2nd word : word 수 [0x0001 - 0x0600/0x0800]
;ESHLHL,HLHL;	return value	ESHLHL, HLHL;
(25LC040/080/160	comment	파라메터를 읽어내는 시작주소는 8 또는 16의 배수이어야함. (25LC320/25LC320A인 경우 16)
/320/320A만 해		word수도 8또는 16의 배수이어야함. (25LC320/25LC320A인 경우 16)
당) (+ <mark>#^</mark> %)		SARAM의 0x0A00/0x3F8200/0xE200 번지 데이터가 EEPROM의 0번지(low byte)/1번지(high byte)에 저장됨.
		마지막에 세미콜론 대신 콜론이 echo되면 시작주소나 word수가 not valid인 경우이거나, 저장하고 나서 다
		시 읽어낸 값이 다른 경우임.
		F2811 또는 F28334를 사용할때 25LC320/320A인 경우 2K word까지 사용이 가능하며, 시작주소/word 수가
		0x11FF/0x0800 으로 늘어남.



명령어	구분	설명		
	operation	DSP SRAM에 있는 작동 파라메터 값을 한꺼번에 EEPROM에 저장함.		
· E O A E E A · (1 # ^ W)	return value	EsA55A;		
;EsA55A; (+#^%)	aammant	LF2406A의 경우는 ES0A00,0100;을 수행한 것과 같으며, F2811의 경우는 ES0A00,01CO; 을 수행한 것과 같으		
	comment	며, F28334의 경우는 ES0A00,0200; 을 수행한 것과 같음.		
	operation	DSP SRAM에 있는 파라메터 값을 default 값으로 바꿈. 이때 ST 명령어로 선정된 motor type에 따라서		
	operation	파라메터의 기본값이 설정됨.		
;EDA55A; (+#^%)	return value	EDA55A;		
,EUASSA, (T# %)	comment	EEPROM에 저장을 하지는 않음.		
		F2811 또는 F28334의 경우는 EDA55A;수행시에 제어기주소가 정상이면 제어기주소/호스트주소		
		/operation_mode_SWITCH/DUMMY_MODE/SEQUENCE_CONTROL_MODE 값을 바꾸지 아니함.		



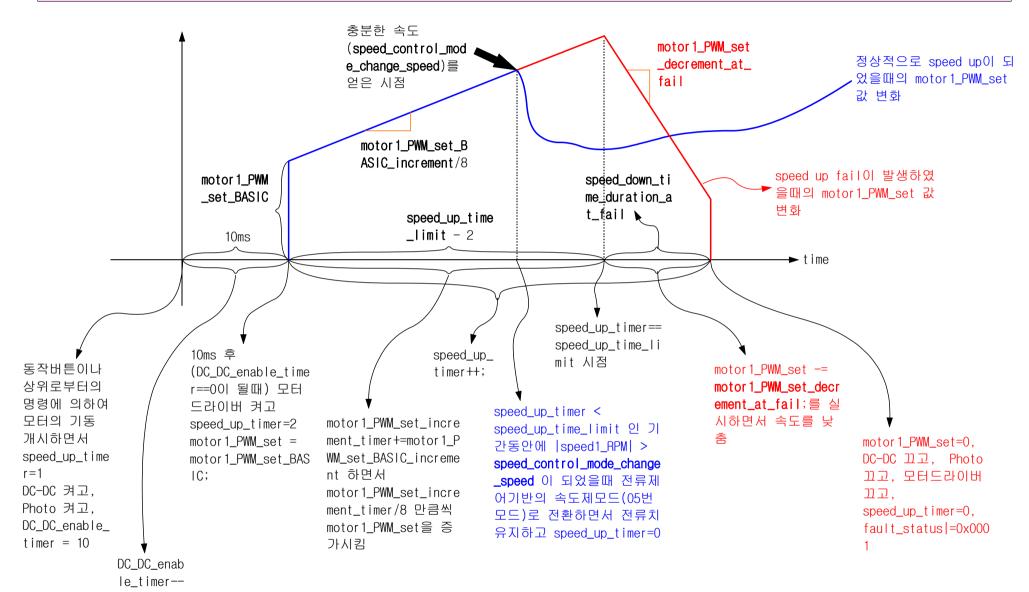
명령어	구분	설명
	operation	RS232, RS485, CAN 통신과정에서 어떤 에러가 발생하였는지를 읽어서 host로 전송함.
		ECHLHL,HLHL; (LF2406A의 경우)
		ECHLHL, HLHL; (F2811/F28334의 경우)
		1st word : SCIA RS232 통신에러
		bit2 : SCIA parity-error flag.
		bit3 : SCIA overrun-error flag.
		bit4 : SCIA framing-error flag. (an expected stop bit is not found.)
		bit5 : SCIA break-detect flag. (a break condition(ten continuous low bits) occurs.)
	return value	bit7 : SCIA receiver-error flag.
;EC?; (+#^%)		2nd word : CAN 통신에러
		bit0: At least one of the error counters reached the warning level of 96.
		bit1 : The CAN module is in error-passive mode.
		bit2: There is an abnormal rate of error occurrences on the CAN bus.
		bit3 : The CAN module did not receive an acknowledge.
		bit4 : The stuff bit rule was violated.
		bit5 : The CAN module received a wrong CRC.
		bit6 : The CAN module did not detect a recessive bit.
		bit7: The received bit does not match the transmitted bit outside of the
		arbitration field; or during transmission of the arbitration field, a
		dominant bit was sent but a recessive bit was received.
		bit8 : A Form Error occurred on the bus.



명령어	구분	설명		
		3rd word : SCIB RS232 통신에러 (F2811/F28334의 경우에만 해당됨)		
		bit2 : SCIB parity-error flag.		
	roturn volue	bit3 : SCIB overrun-error flag.		
	return value	bit4 : SCIB framing-error flag. (an expected stop bit is not found.)		
		bit5 : SCIB break-detect flag. (a break condition(ten continuous low bits) occurs.)		
		bit7 : SCIB receiver-error flag.		
		가장 최근에 있었던 RS232, RS485, CAN 통신에러상태를 읽어냄.		
	comment	통신에러가 발생하면 자동 복구되며, 이때의 에러상태를 저장하고 있다가, EC?의 명령에 따라 저장되었던		
	Comment	값을 보내줌.		
		통신 장치를 무조건 초기화하려면 ECR; 명령어를 사용함.		
	operation	RS232, RS485, CAN 통신에러 플래그를 없애고 에러를 복구함.		
;ECR; (+#^%)	return value	ECR;		
	comment			
	operation	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전송되어오는 broadcasting 데이터를 수신하도록		
	operation	설정함.		
;EcS; (+#^)	return value	EcS;		
	comment	전원을 켰을때의 기본상태는 수신하지 않은 것임.		
	Commert	2012년 6월 03일 S/W 버젼부터 사용가능한 명령임.		
	operation	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전송되어오는 broadcasting 데이터를 수신하지 못		
	operation	하도록 설정함.		
;EcR; (+#^)	return value	EcR;		
, 2011)		전원을 켰을때의 기본상태는 수신하지 않은 것임.		
	comment	브레이크제어기나 조향제어기에는 CAN 통신일때 선택적 broadcasting 기능이 해당 없음.		
		2012년 6월 03일 S/W 버젼부터 사용가능한 명령임.		

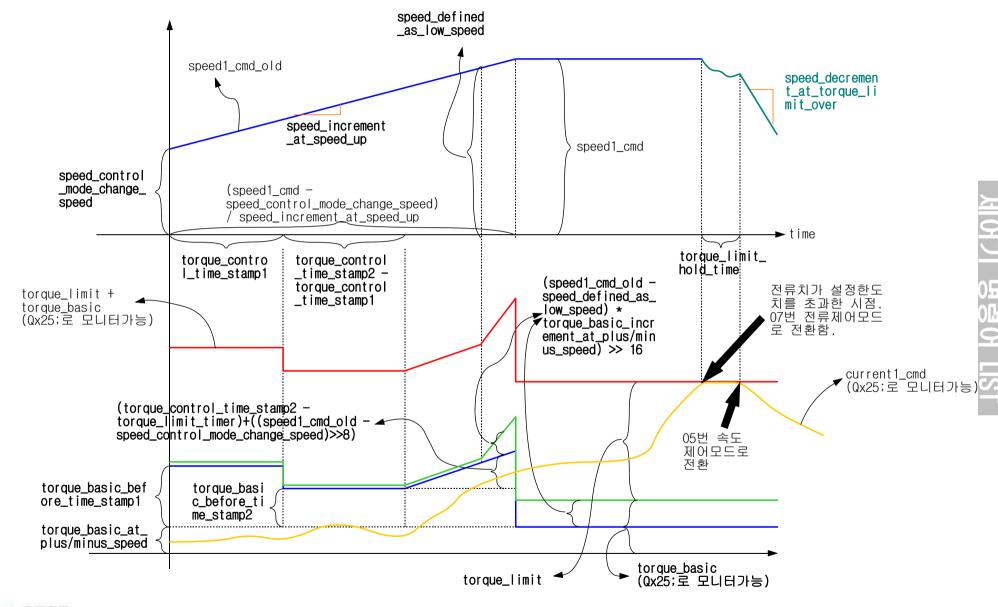


제일메디컬 전동 드라이버 (speed up procedure)



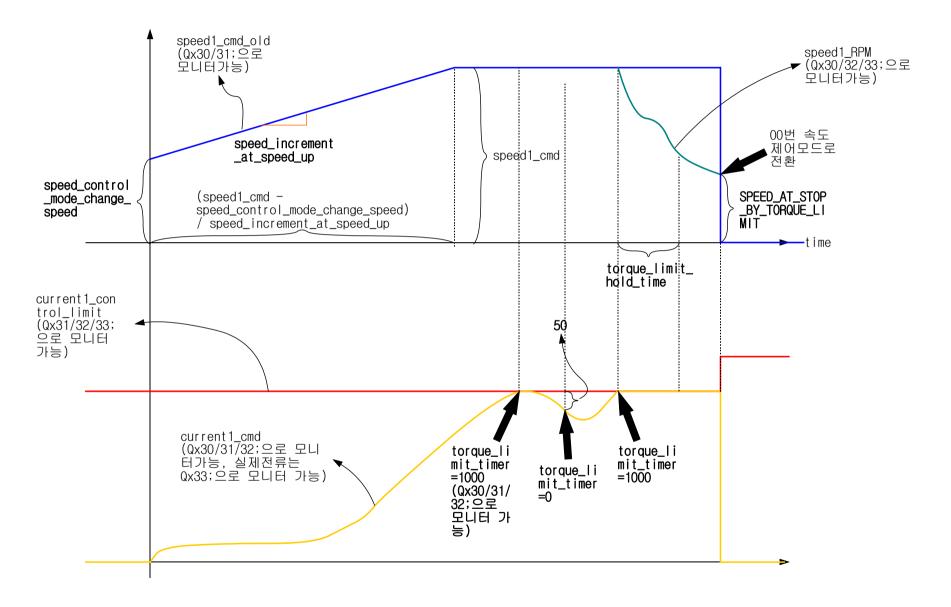


제일 메디컬 전동 드라이버 (정상적인 speed up 이후 procedure : 최초 S/W 버젼)





제일 메디컬 전동 드라이버 (정상적인 speed up 이후 procedure : 최종 S/W 버젼)





명령어	구분	설명		
		속도관련 파라메터를 설정함.		
	operation	1st word : speed_control_mode_change_speed (2000 rpm)		
	operation	2nd word : speed_defined_as_low_speed (gear_ratio * 30 rpm)		
		3rd word : speed_defined_as_high_speed (gear_ratio * 50 rpm)		
;WSdddd1,dddd2,d	return value	Sdddd1,dddd2,dddd3;		
ddd3;		speed_up_time_limit (Ww 명령어 첫번째 파라메터로 설정) 이내에 speed_control_mode_change_speed (WS 명령어		
		첫번째 파라메터로 설정) 이상 도달하지 못하면 E2 에러 발생됨.		
;WS?;	a a mmant	speed_defined_as_low_speed (WS 명령어 두번째 파라메터로 설정) 는 torque_basic(모터기본부하) 값을 계산하는데 사용함.		
	comment	speed_defined_as_high_speed (WS 명령어 세번째 파라메터로 설정) 는 현재 사용하고 있지 않음.		
		gear_ratio는 기본값이 275임.		
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
		최종 S/W 버젼에서는 마지막 두개의 파라메터를 사용하지 않음.		
		속도증분관련 파라메터를 설정함.		
	operation	1st word : speed_increment_at_speed_up (16 rpm/ms)		
	operation	2nd word : speed_decrement_at_speed_down (32 rpm/ms)		
		3rd word : speed_decrement_at_torque_limit_over (50 rpm/ms)		
;Wsdddd1,dddd2,d	return value	Wsdddd1,dddd2,dddd3;		
ddd3;		speed_increment_at_speed_up (Ws 명령어 첫번째 파라메터로 설정) 은 05번 속도제어모드로 정상적으로 진입하였		
		을 때 이후에 가속을 하는 가속률을 의미함.		
;Ws?;		speed_decrement_at_speed_down (Ws 명령어 두번째 파라메터로 설정) 은 05번 속도제어모드로 정상적으로 진입하		
	comment	였으나 토크리밋에 걸리지 않고 모터작동을 멈출때 감속률을 의미함.		
		speed_decrement_at_torque_limit_over (Ws 명령어 세번째 파라메터로 설정) 은 05번 속도제어모드로 정상적으로		
		진입하고 토크리밋에 의하여 작동을 멈출때 감속률을 의미함.		
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		



명령어	구분	설명		
		모터 기본부하를 계산하기 위한 파라메터를 설정함.		
		1st word : torque_control_time_stamp1 (20 ms)		
	operation	2nd word : torque_control_time_stamp2 (60 ms)		
	operation	3rd word : torque_basic_before_time_stamp1 (300 0.7326mA)		
		4th word : torque_basic_before_time_stamp2 (150 0.7326mA)		
;Wtdddd1,dddd2,d		5th word : torque_basic_min_after_time_stamp2 (80 0.7326mA)		
ddd3,dddd4,dddd5	return value	Wtdddd1,dddd2,dddd3,dddd4,dddd5;		
;		torque_control_time_stamp1 (Wt 명령어 첫번째 파라메터로 설정) 은 가속시간을 나타냄.		
		torque_control_time_stamp2 (Wt 명령어 두번째 파라메터로 설정) 은 가속후 과도기를 나타냄.		
;Wt?;		torque_basic_before_time_stamp1 (Wt 명령어 세번째 파라메터로 설정), torque_basic_before_time_stamp2 (Wt		
	comment	명령어 네번째 파라메터로 설정) 값은 가속에 따른 모터기본 부하를 설정함.		
	Commont	torque_basic_min_after_time_stamp2 (Wt 명령어 다섯번째 파라메터로 설정) 값은 torque_control_time_stamp2		
		(Wt 명령어 두번째 파라메터로 설정) 이후에 적용하는 최소값임.		
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
		최종 S/W 버젼에서는 사용하지 않는 명령어임.		
		모터 기본부하를 계산하기 위한 추가 파라메터를 설정함.		
		1st word : torque_basic_at_plus_speed (350 0.7326mA)		
	operation	2nd word: torque_basic_at_minus_speed (340 0.7326mA)		
		3rd word : torque_basic_increment_at_plus_speed (400 0.01118mA/1000RPM)		
;WTdddd1,dddd2,d		4th word: torque_basic_increment_at_minus_speed (350 0.01118mA/1000RPM)		
ddd3, dddd4;	return value	WTdddd1,dddd2,dddd3,dddd4;		
		torque_basic_at_plus_speed/torque_basic_at_minus_speed (WT 명령어 첫번째/두번째 파라메터로 설정) 값은 정방향/역방향으로 구동을 개시할때 적용하는 모터기본부하임 (감속기 마찰력등에 해당하는 부하).		
;WT?;		torque_basic_increment_at_plus_speed/torque_basic_increment_at_minus_speed (WT 명령어 세번째/네번째 파		
	comment	라메터로 설정) 값은 정방향/역방향으로 구동을 개시할때, 모터의 속도가 speed_defined_as_low_speed (WS 명령		
	COMMENT	어 두번째 파라메터로 설정) 를 초과하는 경우에 속도에 따라서 적용하는 모터기본부하임.		
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
		최종 S/W 버젼에서는 사용하지 않는 명령어임.		
CUBE		367		
- DUUL				

명령어	구분	설명	
		추가 파라메터를 설정함.	
		1st word: motor1_PWM_set_BASIC (200 0.04883%)	
	operation	2nd word: motor1_PWM_set_BASIC_increment (16 0.0061%/ms)	
	operation	3rd word : motor1_PWM_set_decrement_at_fail (10 0.04883%/ms)	
		4th word : gear_ratio (275)	
;WWdddd1,dddd2,d		5th word : torque_per_step (10 0.7326mA)	
ddd3,dddd4,dddd5	return value	WWdddd1,dddd2,dddd3,dddd4,dddd5;	
;		motor1_PWM_set_BASIC (WW 명령어 첫번째 파라메터로 설정), motor1_PWM_set_BASIC_increment (WW 명령어 두번	
		째 파라메터로 설정), motor1_PWM_set_decrement_at_fail (WW 명령어 세번째 파라메터로 설정) 은 초기 가속도중	
;ww?;		에 PWM값을 증가시키거나 speed_control_mode_change_speed (WS 명령어 첫번째 파라메터로 설정) 에 도달하지 못	
	comment	하고 감속할때 사용하는 값임. 첫번째와 두번째 값이 너무 작으면 E2 기동 불량 에러가 발생하며, 너무 크면 순간	
	Commert	전류소모가 커지며 토크관련 파라메터 설정에 주의를 요함.	
		gear_ratio (WW 명령어 네번째 파라메터로 설정) 은 감속기의 감속비를 나타냄.	
		torque_per_step (WW 명령어 다섯번째 파라메터로 설정) 은 사용하고 있지 않음.	
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.	
		시간관련 파라메터를 설정함.	
		1st word : speed_up_time_limit (1000 ms)	
	operation	2nd word : speed_down_time_duration_at_fail (200 ms)	
;Wwdddd1,dddd2,d	oporation	3rd word : automatic_power_off_time_set (6000 ms)	
ddd3, dddd4, dddd5		4th word : torque_limit_hold_time (250 ms)	
;		5th word : temperature_50deg (4000)	
	return value	Wwdddd1,dddd2,dddd3,dddd4,dddd5;	
;Ww?;		speed_up_time_limit (Ww 명령어 첫번째 파라메터로 설정) 이내에 speed_control_mode_change_speed (WS 명령어	
	comment	첫번째 파라메터로 설정) 이상 도달하지 못하면, speed_down_time_duration_at_fail (Ww 명령어 두번째 파라메터	
		로 설정) 시간 동안 감속한 후에 E2 에러를 발생시킴.	
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.	



명령어	구분	설명	
		전압관련 파라메터를 설정함.	
	operation	1st word : motor_drive_voltage_low (4000 mV)	
	operation	2nd word : LOW_BAT_THRESHOLD (3200 mV)	
		3rd word : LCD_BIAS_CHANGING_THRESHOLD (3300 mV)	
;WVdddd1,dddd2,d	return value	WVdddd1,dddd2,dddd3;	
ddd3;		모터구동전압(통상 7V)이 motor_drive_voltage_low (WV 명령어 첫번째 파라메터로 설정) 이하가 되면 E3	
		fault를 발생시키고 작동을 멈춤.	
;WV?;		밧데리전압이 LOW_BAT_THRESHOLD (WV 명령어 두번째 파라메터로 설정) 전압이하가 되면 LCD 표시판에 low	
	comment	voltage를 표시함.	
		밧데리전압이 LCD_BIAS_CHANGING_THRESHOLD (WV 명령어 세번째 파라메터로 설정) 전압이하가 되면 LCD의 bias	
		를 조정하여 contrast 비를 높여서 더 또렷하게 보이도록 함.	
	파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
기타 파라메터를 설정함.		기타 파라메터를 설정함.	
	operation	1st word : torque_limit_at_steps[5] (1300 0.7326mA)	
		2nd word : charge_detection_threshold (500 mV)	
;Wvdddd1,dddd2;	return value	Wvdddd1,dddd2;	
	torque_limit_at_steps[5] (Wv 명령어 첫번째 파라메터로 설정) 는 역회전시에 적용하는 최대부경		
;Wv?; 드는 전류치를 설정하는 값임.		드는 전류치를 설정하는 값임.	
	comment	charge_detection_threshold (Wv 명령어 두번째 파라메터로 설정) 보다 작은 전압이 감지되면 charge가 실시되	
		고 있다고 판단하고 작동전원을 차단함.	
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.	



į	명령어	구분	설명		
	WMddddd1,dddd2,d	operation return value	low speed일때 최대부하토크에 관련한 파라메터를 설정함. 1st word : torque_limit_at_steps_low_speed[0] (450 0.7326mA) 2nd word : torque_limit_at_steps_low_speed[1] (600 0.7326mA) 3rd word : torque_limit_at_steps_low_speed[2] (750 0.7326mA) 4th word : torque_limit_at_steps_low_speed[3] (900 0.7326mA) 5th word : torque_limit_at_steps_low_speed[4] (1050 0.7326mA) WMdddd1,dddd2,dddd3,dddd4,dddd5; torque_limit_at_steps_low_speed[0] (WM 명령어 첫번째 파라메터로 설정) 는 정회전시에 적용하는 최대부하		
:	ddd3,dddd4,dddd5	comment	토크설정값1.0 Ncm 에 해당하는 전류치를 설정함. torque_limit_at_steps_low_speed[1] (WM 명령어 두번째 파라메터로 설정) 는 정회전시에 적용하는 최대부하 토크설정값1.5 Ncm 에 해당하는 전류치를 설정함. torque_limit_at_steps_low_speed[2] (WM 명령어 세번째 파라메터로 설정) 는 정회전시에 적용하는 최대부하 토크설정값2.0 Ncm 에 해당하는 전류치를 설정함. torque_limit_at_steps_low_speed[3] (WM 명령어 네번째 파라메터로 설정) 는 정회전시에 적용하는 최대부하 토크설정값2.5 Ncm 에 해당하는 전류치를 설정함. torque_limit_at_steps_low_speed[4] (WM 명령어 다섯번째 파라메터로 설정) 는 정회전시에 적용하는 최대부하 토크를 만드는 전류치를 설정하는 값인데 SE 명령어로만 사용이 가능한 값임. 파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
	Wmdddd1,dddd2,d ddd3,dddd4,dddd5	operation	midium speed일때 최대부하토크에 관련한 파라메터를 설정함. 1st word: torque_limit_at_steps_mid_speed[0] (450 0.7326mA) 2nd word: torque_limit_at_steps_mid_speed[1] (600 0.7326mA) 3rd word: torque_limit_at_steps_mid_speed[2] (750 0.7326mA) 4th word: torque_limit_at_steps_mid_speed[3] (900 0.7326mA) 5th word: torque_limit_at_steps_mid_speed[4] (1050 0.7326mA) Wmdddd1,dddd2,dddd3,dddd4,dddd5;		
		comment	파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.		
		Sommor :	MUNICIONON, COLT FUNDIN VION VIOD.		



명령어	구분	설명
		high speed일때 최대부하토크에 관련한 파라메터를 설정함.
;Wndddd1,dddd2,d		1st word : torque_limit_at_steps_high_speed[0] (450 0.7326mA)
	operation	2nd word : torque_limit_at_steps_high_speed[1] (600 0.7326mA)
dudo, dudu4, duduo	operation	3rd word : torque_limit_at_steps_high_speed[2] (750 0.7326mA)
,		4th word : torque_limit_at_steps_high_speed[3] (900 0.7326mA)
;Wn?;		5th word : torque_limit_at_steps_high_speed[4] (1050 0.7326mA)
, WII: ,	return value	Wndddd1,dddd2,dddd3,dddd4,dddd5;
	comment	파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.
		high speed일때 최대부하토크에 관련한 파라메터를 설정함.
		1st word : SPEED_SCLAE_FACTOR (915 bit)
	operation	2nd word : SPEED_AT_STOP_BY_TORQUE_LIMIT (5000 rpm)
		3rd word : TIME_DELAY_FOR_FORCED_STOP (5000 ms)
		4th word : TORQUE_SCALE_FACTOR (256 0.390625%)
;WZdddd1,dddd2,d	return value	WZdddd1,dddd2,dddd3,dddd4;
ddd3, dddd4;		SPEED_SCLAE_FACTOR 값을 크게하면 실제속도는 감소함.
dudo, dudu-,		부하에 의한 감속이 개시된후 torque_limit_hold_time (Www 명령어로 설정) 이 경과한 다음, 모터구동을 차
;WZ?;		단하는데, 이때 모터의 속도가 SPEED_AT_STOP_BY_TORQUE_LIMIT 이하로 하강되어 있어야 차단함. 만약
,,,,		TIME_DELAY_FOR_FORCED_STOP 만큼의 시간이 경과하여도 모터의 속도가 SPEED_AT_STOP_BY_TORQUE_LIMIT 이
	comment	하로 하강되지 않았다면 강제로 감속을 개시함.
		TORQUE_SCALE_FACTOR 값은 WM/Wm/Wn 명령으로 설정한 값에 모두 공통적으로 적용되며,
		TORQUE_SCALE_FACTOR 값이 256일때 WM/Wm/Wn 명령으로 설정한 값을 그대로 사용하는 것에 히당하고,
		TORQUE_SCALE_FACTOR 값을 크게하면 실제토크가 증가함.
		파라메터는 FsA55A; 명령으로 EEPROM에 저장이 가능함.



Qxdddd1 에 의하여 전송되는 변수값 (제일메디컬)

	RJM_VER7인 경우				
dddd1 값	첫번째 변수	두번째 변수	세번째 변수	네번째 변수	
0	position1_set	position1_at_1ms	speed1_RPM_10	current1_cmd	
1	speed1_cmd_old [rpm]	speed1_RPM [rpm]	current1_cmd [0.7326mA]	current1_AD [0.7326mA]	
2	current1_cmd [0.7326mA]	current1_AD [0.7326mA]	ADCRESULT[1] [0.7326mA]	motor1_PWM_set [0.04883%]	
3	speed1_cmd_old [rpm]	speed1_RPM [rpm]	current1_AD [0.7326mA]	motor_voltage_scaled [mV]	
4	speed1_cmd_old [rpm]	speed1_RPM [rpm]	current1_AD [0.7326mA]	speed_pulse_low_duration [0.25 μ s]	
20	motor_current_scaled [mA]	PORT_STATUS	mainLoop_execution_time [%]	<pre>maximum_interrupt_service_time [%]</pre>	
21	motor1_error	motor1_integral	motor1_derivative	current1_cmd	
22	speed1_RPM	motor1_PWM_set	current1_cmd	current 1_AD	
23	speed_up_timer [ms]	fault_status	motor1_PWM_set [0.04883%]	speed1_cmd_old [rpm]	
24	speed_up_timer [ms]	operation_mode	motor1_PWM_set [0.04883%]	speed1_cmd_old [rpm]	
25	torque_limit_timer [ms]	torque_basic [0.7326mA]	torque_limit + torque_basic [0.7326mA]	current1_cmd [0.7326mA]	
26	BOTTON_RPM_press_timer	BOTTON_RPM_port	RPM_select	BOTTON_TORQUE_port	
27	ADCRESULT[0] (charge_voltage)	ADCRESULT[2] (motor voltage)	ADCRESULT[3] (motor temperature)	ADCRESULT[4] (bat voltage)	
28	charge_voltage_scaled [mV]	motor_voltage_scaled [mV]	motor_temperature_scaled [0.1℃]	bat_voltage_scaled [mV]	
29	charge_voltage_scaled [mV]	motor_current_scaled [mA]	motor_voltage_scaled [mV]	bat_voltage_scaled [mV]	



적용 CPU	명령어	return value	operation
+#^%	;QS;	QSHL,HLHL,HLHL,HLHL; (스텝핑모터가 아닌경우) QSHL,HLHL,HLHLHLHL,HLHLHLHL; (스텝핑모터인경우)	on/off 형식의 센서 상태를 읽어서 host로 전송함.
+#^	;Qs; (절대각센서를 사용하 는 경우가 아닐때)	QsHLHL, HLHL, HLHL;	인공안구의 경우는 팬/틸트방향의 절대각을 읽어서 host로 전송하며, 베스트모션의 경우는 경사각 센서값을 읽어서 host로 전송함.
#	;Qs; (절대각센서를 사용하 는 경우)	QSHLHLHLHL, HLHLHLHL;	절대각센서값을 읽어서 host를 전송함.
+#^%	;QA;	QAHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL HL;	하위 8 channel의 12-bit AD 변환값을 읽어서 host로 전송함.
+#^%	;QB;	QBHLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HLHL, HL	상위 8 channel의 12-bit AD 변환값을 읽어서 host로 전송함.
+#^	;Qa;	QaHLHL, HLHL, HLHL;	TLV2556의 12-bit A/D 변환값을 읽어서 host로 전송함.
+#^%	;Qb;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	channel 8-11(F2809는 0-3)의 A/D 변환값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가됨)
+#^	;Qb1;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	Butterworth filter #8-#11의 필터링한 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6축 F/T 센 서의 경우는 2채널이 추가됨)
+#^	;Qb2;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor에서 bridge amplifier #1-#4의 offset 값과 공급전압의 크기를 필터링한 값을 읽어서 host로 전송함. (6축 F/T 센서의 경우는 2채널이 추가됨)
+#^	;Qb3;	QbHBLBHBLBHBLBHBLBCS; (6축 F/T 센서의 경우는 4 byte가 추가됨)	F/T sensor에서 계산된 모멘트값과 힘값을 읽어서 host로 전송함.
+#^	;Qb4;	QbHBLBHBLBHBLBCS;	F/T sensor의 D/A 출력 전압(bridge amplifier의 영점 조 정용)을 A/D 변환한 값을 읽어서 host로 전송함.



적용 CPU	명령어	return value	operation	
	;Qb5;	QbHBLBHBLBHBLBCS;	F/T sensor에서 모멘트값과 힘값을 계산한 후, 추가 필터	
+#^		(6축 F/T 센서의 경우는 4 byte가 추가됨)	링한 값을 읽어서 host로 전송함.	
+#^	.0.0	QbHBLBHBLBHBLBHBLBCS;	모터제어기에서 모터1/2의 순간 위치명령값과 현재 위치값	
+#* .	;Qb9;	QDDBLBDBLBDBLBCS,	을 읽어서 host로 전송함.	
		RS485의 경우 :		
		bHbMbLbHbMbLbHbMbLbCS;	Butterworth filter 8-11 (6축 F/T 센서의 경우는 8-13)의	
+#^%	;b;	CAN의 경우 :	필터링된 값을 읽어서 host로 전송함.	
		bHbMbLbHbMbLbHbMbLb;	할다당한 없을 밝아서 NOSI도 선충함.	
		(6축 F/T 센서의 경우는 3 byte가 추가됨)		
		QEHLHL,HLHL,HLHL,HLHL; (F2406인경		
+#^%	;QE;	우),	통신에러횟수를 읽어서 host로 전송함.	
T#* 7/0	, WC ,	QEHLHL, HLHL, HLHL, HLHL;	중선에더봇구들 밝어서 NOSI도 신송함.	
		(F2811/F28334인 경우)		
+#^	;QEE;	QEEHLHL, HLHL, HLHL, HLHL;	통신에러횟수를 읽어서 host로 전송함.	
+#^%	;QEA55A;	QEA55A;	모터 1/2의 순간목표/명령/현재변위, 엔코더카운터값, 제	
+#^	;QEA55A, <mark>HLHL,HLHL</mark> ;	QEA55A;	어오차등을 초기값으로 재설정함.	
+#^%	;QEA5A5;	QEA5A5;	모터 1의 순간목표/명령/현재변위, 엔코더카운터값, 제어	
+#^	;QEA5A5, <mark>HLHL</mark> ;	QEA5A5;	오차등을 초기값으로 재설정함.	
+#^%	;QE5A5A;	QE5A5A;	모터 2의 순간목표/명령/현재변위, 엔코더카운터값, 제어	
+#^	;QE5A5A, <mark>HLHL</mark> ;	QE5A5A;	오차등을 초기값으로 재설정함.	
+#^%	;QP;	QPHLHLHLHL, HLHLHLHL;	현재 변위값을 읽어서 host로 전송함.	
+#^%	;Qp;	QpHLHLHLHL, HLHLHLHL;	순간목표변위값을 읽어서 host로 전송함.	
+#^	:0a:	0-1111111111111111111111111111111111111	option으로 부착된 엔코더카운트값을 읽어서 host로 전송	
T#	;Qq;	QqHLHLHLHL, HLHLHLHL;	함.	
+#^%	:0//:	QViiii1,iiii2; (F2406인경우),	현대 소드가운 이십니 boot로 전소하	
T#* 70	;QV;	QVIIII1,IIII2; (F2811/F28334인 경우)	현재 속도값을 읽어서 host로 전송함.	



적용 CPU	명령어	return value	operation
+#^	;Qv;	QvHLHL, HLHL;	현재의 공급전압과 전압강하의 크기를 읽어서 host로 전송 함.
+#^%	;QC;	QCiiii1,iiii2; QCiiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) QCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) QCiiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)	모터에 흐르는 filtering된 상전류값을 읽어서 host로 전송함.
+#^%	;Qc;	Qciiii1,iiii2; Qciiii1,iiii2,iiii3,iiii4; (2상 STEP형 만 해당됨) Qciiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (3상 STEP형 만 해당됨) Qciiii1,iiii2,iiii3,iiii4,iiii5; (5상 STEP형 만 해당됨)	모터에 흐르는 filtering된 상전류값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.
+#^	;QK;	QKiiii1,iiii2;	모터에 흐르는 filtering된 총전류값을 읽어서 host로 전송함.
+#^	;Qk;	Qkiiii1,iiii2;	모터에 흐르는 filtering된 총전류값을 읽어서 암페어값으로 환산하여 host로 전송함.
+#^%	;Qe; (Force/Moment 센서가 아닌 경우)	Qe 1, 2, 3, 4, 5, 6;	error, error 적분값, error 미분값을 읽어서 host로 전송함.
+#^	;Qe; (Force/Moment 센서인 경 우)	Qeiiii0,iiii1,iiii2,iiii3; or Qeiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)	X-moment, Y-moment, Z-force, extra load cell값에 대하여 추가 필터링한 값을 읽어서 host로 전송함. (6축의 경우는 X-moment, Y-moment, Z-force, X-force, Y-force, Z-moment 임)



적용 CPU	명령어	return value	operation
+#^%	;QF;	QFiiii0,iiii1,iiii2,,iiii5,iiii6,iii i7;	filter 출력값을 읽어서 host로 전송함.
+#^	;Qf;	Qfiiii0,iiii1,iiii2,iiii3; or Qfiiii0,iiii1,iiii2,iiii3,iiii4,iiii5; (6축 F/T 센서의 경우)	X-moment, Y-moment, Z-force, extra load cell를 읽어서 host로 전송함. (6축의 경우는 X-moment, Y-moment, Z-force, X-force, Y-force, Z-moment 임)
+#^	;QW;	QWiiii0,iiii1,iiii2,,iiii5,iiii6,iii i7;	모터1과 모터2의 input power, output power, output torque, efficiency를 읽어서 host로 전송함.
+#^	;Qw;	Qwiiii0,iiii1,iiii2,,iiii5,iiii6,iii i7;	모터1과 모터2의 loss power, winding temperature, housing temperature, ambient temperature를 읽어서 host 로 전송함.
+#^	;QJ;	QJHLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HL HL,HLHL;	faultBufPointer, faultBuf[0-7]을 읽어서 host로 전송함.
+#^	;Qj;	Q;HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH	faultBufPointer_save, faultBuf_save[0-14]을 읽어서 host로 전송함.
+#^	;QjA55A;	Q;HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HLH	faultBufPointer_save, faultBuf_save[0-14]을 0으로 clear함.
+#^%	;Qddddd1;	Qddddd1;	Sleep_1micro(dddd1), Sleep_10micro(dddd1), Sleep_msec(1) 함수실행결과의 지연시간동안 CS_EEPROM_N_port에 low pulse를 출력함.
+#^%	;QD;	QD;	diagnosis mode를 개시함.
+#^%	;QX;	QX;	diagnosis mode를 종료함.



적용 CPU	명령어	return value	operation
+#^%	;Qxdddd1;		1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는
+#^%	;Qx?;	Qxdddd1;	8word를 host PC로 전송하는 데이터의 종류를 선택하는 변
T# /0	, UX ()		수인 data_logging_select 변수값을 설정함.
			1msec 또는 2ms 마다 프로그램에서 설정된 4word 또는
+#^%	;Qxy;	없음	8word를 host PC로 전송을 시작하게 하거나 전송을 중지
			시킴.
+#^	;Qxyz;	없음	Qxy; 또는 Qz; 또는 Qzx; 또는 Qzy; 명령에 의한 모니터링
T#	, WAYZ,	W C	작동을 중지함.
+#^%	;QZHLHL,HLHL,HLHL,HLHL;		Qz 명령어로 1ms 마다 값을 host로 전송하는 4개의 변수,
+#^%	;QZ;	QZHLHL, HLHL, HLHL;	또는 Qy 명령어로 host로 전송하는 8개 변수중에서 앞쪽 4
1π /0	, WZ ,		개의 변수에 대한 주소를 설정함.
+#^%	;Qz;	없음	QZ 명령어로 주소가 지정된 4word를 1msec 마다 host PC로
ιπ /0	, 42,		전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;Qzx;	없음	QZ 명령어로 주소가 지정된 4word를 2msec 마다 host PC로
· ir			전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;Qzy;	없음	QZ 명령어로 주소가 지정된 8word를 2msec 마다 host PC로
· ir	, wzy,		전송을 시작하게 하거나 전송을 중지 시킴.
+#^	;QYHLHL,HLHL,HLHL;	QYHLHL, HLHL, HLHL;	Qy 명령어로 host로 전송하는 8개 변수중에서 뒷쪽 4개의
+#^	;QY;		변수에 대한 주소를 설정함.
		RS232/485의 경우 :	
+#^	;Qy;	QyHBLBHBLBHBLBHBLBHBLBHBLB;	QZ,QY 명령어로 주소가 지정된 8word를 binary 값의 형태
111	, wy ,	CAN의 경우 :	로 host PC에 전송함.
		HBLBHBLBHBLBHBLBHBLBHBLB	
+#^%	;Q1;	Q1 HLHL;	operation_status를 읽어서 host로 전송함.
+#^%	;Q2;	Q2 HLHL,HLHL;	fault_status, fault1_status를 읽어서 host로 전송함.
+#^%	;Q3;	Q3 HLHL;	IRMCK201의 fault_status를 읽어서 host로 전송함.



적용 CPU	명령어	return value	operation
+#^%	;Q4;	Q4 HLHL, HLHL, HLHL;	프로그램 작성 날짜, version, macro 설정을 읽어서 host 로 전송함.
+#^	;Q5;	Q5 HLHL;	operation1_status를 읽어서 host로 전송함.
+#^	;Q6;	Q6 HLHL,HLHL;	conditional compiling 조건을 읽어서 host로 전송함.
+#^%	;QRHLHL,HL;	QRHLHL,HL,HLHL(,HLHL);	지정된 주소에서 시작하여 지정된 양만큼의 DSP SRAM 내용
ιπ /0	, WI II IL IL , I IL ,		을 읽어서 host로 전송함.
+#^	;QQHLHL,HL;	QQHLHL(,HLHL);	지정된 주소에서 시작하여 지정된 양만큼의 DSP SRAM 내용
'π	, WWI ILI IL, I IL,		을 읽어서 host로 전송함.
+#^%	;QTHLHL,HL,HLHL(,HLHL);	QTHLHL,HL,HLHL(,HLHL);	지정된 주소에서 시작하여 지정된 양만큼의 데이터를 DSP
ιπ /0	, WITHERE, THE, THE ILL , THE ILL ,		의 SRAM에 전송함.
			BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의
+#^%	;QrHL,HL;	QrHL,HL,HLHL;	지정된 주소로부터 register 값을 읽어서 host로 전송함.
			(BLDC 구동 chip IRMCK201 전용)
+#^%	;QtHL,HL,HL;	QtHL,HL,HL;	BLDC 모터제어용 chip인 IRMCK201 chip에 있는 register의
+#^%	;QtHL,HL,HLHL;	QtHL,HL,HLHL;	지정된 주소에 host로부터 전송된 데이터를 저장함과 동시
+#^%	;QtHL,HL?;	QtHL,HL,HLHL;	에 DSP의 SARAM 0x0a3e-0x0abd 번지에도 저장함. (BLDC 구
111 /0	, WITE, TE;	W.L. II., I II. II.	동 chip IRMCK201 전용)



적용 CPU	명령어	return value	operation
% %	;PWiiii1,iiii2; ;PW?;	PWiiii1,iiii2;	모터의 PWM duty 출력을 강제로 설정함.
+#^	;PWiiii1,iiii2; (VER4/8)		
+#^	;PWiiii1,iiii2,iiii3,iiii	;PWiiii1,iiii2; (VER4/8)	
	4; (VER5)	;PWiiii1,iiii2,iiii3,iiii4; (VER5)	모터의 PWM duty 출력을 강제로 설정함.
+#^	;PWiiii1,iiii2,iiii3,iiii	;PWiiii1,iiii2,iiii3,iiii4,iiii5,iiii6;	모디의 PWM duty 물역들 경제도 설정함.
	4, iiiii5, iiiii6; (VER6/7)	(VER6/7)	
+#^	;PW?;		
+#^%	;PAIIII1,IIII2;	DALLILA LILIO:	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
+#^%	;PA?;	PAIIII1, IIII2;	을 설정함.
+#^%	;PBHLHLHLHL,HLHLHLH;		모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
+#^%	;PB?;	PBHLHLHLHL, HLHLHLHL;	을 설정함.
+#^%	;PaHBMBLBHBMBLBCS;	PaHBMBLBHBMBLBCS;	모터제어기의 작동모드에 따라서 변위/속도/전류의 명령값
111 /0	, F at IDIVIDEDI IDIVIDEDOS,	Fai IDMIDEDI IDMIDEDOS,	을 설정함.
+#^	;P11111,11112;	PIIII1, IIII2;	위치 명령값의 증분을 설정하고 이동을 명령함.
+#^	;PI?;	F111111,11112,	기시 강이없니 강한글 말이고 이용을 강하다.
+#^	;P1 1;	P1 1;	모터 1에 대하여 위치 명령값의 증분을 설정하고 이동을
+#^	;P1?;	F 111111,	명령함.
+#^	;P2 1;	P2 1;	모터 2에 대하여 위치 명령값의 증분을 설정하고 이동을
+#^	;P2?;	F211111,	명령함.
+#^%	;PEHLHL;		
+#^	;PEA55A;	PEHLHL;	선택된 보드의 모터 출력을 ON 시킴.
+#^%	;PE?;		
+#^%	;PRHLHL;	PRHLHL;	선택된 보드의 fault flag를 clear 시킴.
+#^	;PRA55A;	'HILIL,	변기면 포트의 Iduit Hay를 Cital 시점.



적용 CPU	명령어	return value	operation
+#^% +#^ +#^	;PDHLHL; ;PDA55A; ;PD?;	PDHLHL;	선택된 보드의 모터 출력을 OFF 시킴.
+#^% +#^%	;PMdddd1,dddd2,dddd3;;PM?;	PMdddd1,dddd2,dddd3;	현재위치로부터 dddd1점을 거쳐 dddd2점까지 dddd3의 속도로 이동시킴.
+#^%	;PME;	PME;	PM 에 의한 동작을 멈춤.
+#^% +#^%	;PP 1, 2; ;PQ 1, 12;	PPIIII1, IIII2; PQIIII1, IIII2;	모터제어기의 작동모드에 따라서 10msec/30msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터가 움직이도록 함. PP인경우는 10msec, PQ인 경우는 30msec임.
+#^% +#^%	;PpHBMBLBHBMBLBCS; ;PqHBMBLBHBMBLBCS;	2006년 10월 7일 이전 S/W 버젼 PpHBMBLBHBMBLBCS; PqHBMBLBHBMBLBCS; 2006년 10월 7일 S/W 버젼부터 PpLBHBLBLBHBLBHBLBCS; 또는 PpHBLBHBLBHBLBHBLBCS; (2007년 9월 19일부터 사용가능) PqLBHBLBHBLBHBLBHBLBCS; (2007년 9월 19일부터 사용가능)	모터제어기의 작동모드에 따라서 10msec/30msec 동안에 가야할 변위/속도/전류의 명령값을 설정하 고 모터가 움직이도록 함. Pp인경우는 10msec, Pq인 경우는 30msec임.
+#^%	;PPE; or ;PQE;	PPE; or PQE;	PP, PQ, Pp, Pq 에 의한 동작을 멈춤.
+#^%	;PPR; or ;PQR;	PPR; or PQR;	모터가 stall된 채로 정지된 상태를 해지함.
+#^%	;pHBMBLBHBMBLBCS; (RS485의 경우) pHBMBLBHBMBLBCS (CAN 의 경우)	RS485의 경우: pLBHBLBHBLBHBLBHBLBCS; 또는 pHBLBHBLBHBLBHBLBCS;(2007년 9월 19일부터 사용가능) CAN의 경우: LBHBLBHBLBHBLBHBLB 또는 HBLBHBLBHBLBHBLB(2007년 9월 19일부터 사용가능)	모터제어기의 작동모드에 따라서 10msec 동안에 가야할 변위/속도/전류의 명령값을 설정하고 모터 가 움직이도록 함.



적용 CPU	명령어	return value	operation
+#^%	;PsA55A;	PsA55A;	모터 1,2를 모두 현재의 위치에서 감속을 실시한 후에 정
+#^	;PtA55A;	PtA55A;	지함.
+#^%	;PsA5A5;	PsA5A5;	모터 1을 현재의 위치에서 감속을 실시한 후에 정지함.
+#^	;PtA5A5;	PtA5A5;	오다 1을 단체의 귀시에서 남극을 일시한 후에 성지함.
+#^%	;Ps5A5A;	Ps5A5A;	모터 2를 현재의 위치에서 감속을 실시한 후에 정지함.
+#^	;Pt5A5A;	Pt5A5A;	고니 Z을 현재의 위치에서 음극을 물시한 후에 당시함.
+#^	;PeA55A;	PeA55A;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2 모두를 02번 위치제어모드로 바꾸고, 현재의 위치에서 감속후 정지함.
+#^	;PeA5A5;	PeA5A5;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을 02번 위치제어모드로 바꾸고, 현재의 위치에서 감속후 정지함.
+#^	;Pe5A5A;	Pe5A5A;	digital Hall sensor만을 사용하는 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를 02번 위치제어모드로 바꾸고, 현재의 위치에서 감속후 정지함.
+#^	;PvA55A;	PvA55A;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1,2 모두를 감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.
+#^	;PvA5A5;	PvA5A5;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 1을 감속한 후 에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속 후 정지함.
+#^	;Pv5A5A;	Pv5A5A;	digital Hall sensor만을 사용하는 wafer 회전구동용 BLDC 모터에 있어서, 05번 속도제어 상태의 모터 2를 감속한 후에, 낮은 속도가 되면 02번 위치제어모드로 바꾸고, 감속후 정지함.



적용 CPU	명령어	return value	operation
+#^	;PvIIII1,IIII2; ;Pv?;	Pv 1, 2;	digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우에, 속도제어 모드에서 속도명령값을 설정하되, 리턴값으로 현재 위치값을 host로 전송함.
+#^	;Pciiii1,iiii2; ;Pc?;	Pciiii1,iiii2;	digital Hall sensor만을 사용하는 BLDC 모터가 아닌 경우에, digital Hall sensor만전류제어 모드에서 전류명령값을 설정하되, 리턴값으로 현재 속도값을 host로 전송함.
+#^ +#^ +#^	;PVIIII1,IIII2,dddd3; ;PVrIIII1,IIII2,dddd3; ;PV?;	PVIII1, III12;	속도제어 모드에서 속도명령값을 설정하되 설정값의 checksum을 사용함.

적용 CPU	명령어	return value	operation
+#^%	;GHHL,HL,iiii1,iiii2;		2개의 리밋센서와 1개의 홈센서를 사용한 홈 동작을 개시
+#^	;GH?;	GHHL, HL, 1,	함. (0.01도 단위의 offset 변위)
+#^	;GhHL,HL, 1,	GhHL,HL, 1, 12;	2개의 리밋센서와 1개의 홈센서를 사용한 홈 동작을 개시
+#^	;Gh?;	GIINL, NL, 11111, 11112,	함. (4체배 엔코더 펄스단위의 offset 변위)
+#^	;GIHL,HL,iiii1,iiii2;	 GIHL,HL, 1, 2;	2개의 리밋센서 만을 사용한 홈 동작을 개시함. (0.01도
+#^	;G1?;	dinL,nL,11111,11112,	단위의 offset 변위)
+#^	;GiHL,HL,	 GiHL,HL, 1, 2;	2개의 리밋센서 만을 사용한 홈 동작을 개시함. (4체배 엔
+#^	;Gi?;	dinc,nc,11111,11112,	코더 펄스단위의 offset 변위)
+#^	;GJHL,HL,iiii1,iiii2;	 GJHL,HL, 1, 12;	한개의 홈센서 만을 사용한 홈 동작을 개시함. (0.01도 단
+#^	;GJ?;	GUIL, IIL,	위의 offset 변위)
+#^	;GjHL,HL,	GjHL,HL, 1, 2;	한개의 홈센서 만을 사용한 홈 동작을 개시함. (4체배 엔
+#^	;Gj?;	GJIE, IIE, IIII, IIII2;	코더 펄스단위의 offset 변위)
+#^	;GKHL,HL,iiii1,iiii2;		한개의 홈센서 만을 사용한 홈 동작을 개시하되 홈센서가
+#^	;GK?;	GKHL, HL, 11111, 11112;	ON 되어 있는 상태에서만 개시함. (0.01도 단위의 offset
ιπ	, un: ,		변위)
+#^	;GkHL,HL, 1, 2;		한개의 홈센서 만을 사용한 홈 동작을 개시하되 홈센서가
+#^	;Gk?;	GkHL, HL, 1 1 1 2;	ON 되어 있는 상태에서만 개시함. (4체배 엔코더 펄스단위
'π	, un: ,		의 offset 변위)
+#^	;GTdddd1,dddd2;	GTdddd1,dddd2;	리밋센서와 홈센서의 ON/OFF를 검출하는 경계전압을 설정
+#^	;GT?;	Gladadi, addaz,	함.
+#^	;Gtdddd1,dddd2;	Gtdddd1,dddd2;	RJM_VER7의 Hall 센서입력의 ON/OFF를 검출하는 기준전압
+#^	;Gt?;	diadai, addaz,	과 히스테리시스 진폭을 설정함.



적용 CPU	명령어	return value	operation
+#^%	;J;	J;	모터1의 속도모드 조그 작동을 수행함.
+#^%	;K;	K;	모터2의 속도모드 조그 작동을 수행함.
+#^%	;SJHL,dddd1,dddd2,dddd3;	CILI dadat dada0 dada0:	속도모드의 조그동작에서 조그방향, 조그이동속도 및 조그
+#^%	;SJ?;	SJHL, dddd1, dddd2, dddd3;	동작 지속시간을 정의함.
+#^%	;j;	j;	모터1의 위치모드 조그 작동을 수행함.
+#^%	;k;	k;	모터2의 위치모드 조그 작동을 수행함.
+#^%	;SjHL,dddd1,dddd2,dddd3;	0:11 11110 11110	위치모드의 조그동작에서 조그방향, 조그이동속도 및 조그
+#^%	;Sj?;	SjHL,dddd1,dddd2,dddd3;	동작 지속거리를 정의함.

적용 CPU	명령어	return value	operation
+#^%	;SAHLHL;	SAHLHL;	ㅁ디펜이기이 조사를 성정하
+#^%	;SA?;	SATILL,	모터제어기의 주소를 설정함.
+#^%	;SMHLHL;	SMHLHL;	위치, 속도, 전류제어 등의 작동모드를 설정함.
+#^%	;SM?;	SWITER,	귀시, 극도, 선규제에 증의 극용도르를 결정함.
+#^%	;STHLHL;	STHLHL;	모터의 기본파라메터를 쉽게 설정하기 위하여 미리 준비되
+#^%	;ST?;	STHERE,	어있는 motor type을 선정함.
+#^%	;SXHLHL;	SXHLHL; 또는 SXHLHLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정
+#^%	;SX?;	2012년 5월 28일 S/W 버젼부터는	함.
ιπ /0	,0/1:	SXHLHLHLHL, HLHLHLHL;	0.
+#^	;SXHLHLHLHL;	SXHLHLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정
+#^	;SX?;	2012년 5월 28일 S/W 버젼부터는	함.
	, o.k.,	SXHLHLHLHL, HLHLHLHL;	.
+#^	;SXHLHLHLHL,HLHLHL;	SXHLHLHLHL, HLHLHLHL;	통신과 제어의 option을 선택하는 DIP switch의 값을 설정
+#^	;SX?;		함.
+#^%	;SEA55A,HLHL,HLHL,HLHL,HL		
	HL;	SEHLHL, HLHL, HLHL, HLHL;	모터 1회전당 펄스수와 POLE PAIR 갯수를 설정함.
+#^%	;SE?;		
+#^%	;SGA55A,dddd1,dddd2,dddd3		
	,dddd4;	SGdddd1,dddd2,dddd3,dddd4;	감속비와 최대회전속도를 설정함.
+#^%	;SG?;		
+#^	;Sgiiii1,iiii2,iiii3,iiii		위치제어 또는 속도제어에서 phase lead 또는 phase lag
	4;	Sgiiii1,iiii2,iiii3,iiii4;	compensator의 파라메터를 설정함. (이 경우는 미분제어
+#^	;Sg?;		게인이 무시됨)
+#^%	;SL11111,11112,11113,1111		
	4;	SLIII11, III12, III13, III14;	위치제어모드에서 변위의 한계값을 설정함.
+#^%	;SL?;		



적용 CPU	명령어	return value	operation
+#^	;\$11111,11112,11113,1111		
	4;	SIIII11, IIII2, IIII3, IIII4;	홈찾기 동작에서 변위의 한계값을 설정함.
+#^	;\$1?;		
+#^%	;SSdddd1,dddd2;	SSdddd1,dddd2;	위치제어모드에서 Point-to-Point 이동시 이동속도값을 설
+#^%	;SS?;	Soudda i, ddddz,	정함.
+#^%	;Ssdddd1,dddd2;	Ssdddd1,dddd2;	PA, PB, Pa 명령에 의한 변위구동시의 가속/감속 기간을
+#^%	;Ss?;	osudut, ddddz,	설정함.
+#^	;Sddddd1,dddd2,dddd3,dddd		 가속/감속시 저전압/과전압이 발생하는 것을 억제하기 위
	4;	Sddddd1,dddd2,dddd3,dddd4;	하여 가감속률을 제한하는 파라메터를 설정함.
+#^	;Sd?;		이어 가끔득절을 제한하는 파다메디를 들었음.
+#^	;SDdddd1,dddd2;	SDdddd1,dddd2;	가속/감속시 저전압/과전압이 발생하는 것을 억제하기 시
+#^	;SD?;	Sbaaa1, aaaa2,	작하는 전압변동량을 설정함.
+#^	;S1dddd1,dddd2;	S1dddd1,dddd2;	OE/OF 위치제어모드에서 모터1의 sine파 변위 주기 및 진
+#^	;\$1?	Stadadt, addaz,	폭을 설정함.
+#^	;S2dddd1,dddd2;	S2dddd1,dddd2;	OE/OF 위치제어모드에서 모터2의 sine파 변위 주기 및 진
+#^	;\$2?	ozudut, dadaz,	폭을 설정함.
+#^	;Sodddd1,dddd2;		속도제어 및 위치제어 모드에서 리밋센서, 비상정지 명령
+#^	;So?;	Sodddd1,dddd2;	또는 비상정지 상황에 의한 강제 감속시의 감속율을 설정
Ιπ	,50:,		함.
+#^%	;SIdddd1,dddd2,dddd3,dddd		
	4,dddd5,dddd6,dddd7,dddd8		
	;	SIdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,d	구동회로와 기구를 보호하기 위한 logic에서 사용하는 전
+#^%	;\$1?;	ddd7,dddd8;	류리밋값과 time threshold를 설정함.
	(Force/Moment 센서가 아닌		
	경우)		



적용 CPU	명령어	return value	operation
+#^	;SIA55A,dddd1,dddd2;		F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는
+#^	;\$1?;		0/1/2/3) 전압의 평균값을 설정하기 위한 filtering을 실
	(Force/Moment 센서인 경	SIdddd1,dddd2;	시하고, 온도값과 함께 force_data_offset 변수값을 설정
	우)		함.
+#^	;SI5AA5,dddd1,dddd2;		F/T 센서에서 AD channel 8/9/10/11(내장형의 경우는
+#^	;\$1?;	SIdddd1,dddd2;	0/1/2/3) 전압의 평균값을 설정하기 위한 filtering을 실
	(Force/Moment 센서인 경	Studdut, ddduz,	시하고, 온도증가에 따른 전압상승의 기울기를 설정함.
	우)		지어고, 근도증기에 따른 산업경공기 기울기을 들었답.
+#^	;S1A5A5,0,0;		F/T 센서에서 no load 상태일때 AD channel 8/9/10/11(내
+#^	;\$1?;		장형의 경우는 0/1/2/3)의 변환값이 2048이 되도록 D/A 변
	(Force/Moment 센서인 경	Studdut, ddduz,	환기를 조절하여 만듦. (F/T센서의 앰프 offset을 2048로
	우)	i	맞추어주는 작동이며 총 5초가 걸림)
+#^%	;SWdddd1,dddd2;	SWdddd1 dddd2:	과전류시에 HW적으로 회로를 차단하기 위한 과전류 설정값
+#^%	;SW?;		을 설정함.
+#^%	;Swdddd1,dddd2;	Swdddd1,dddd2;	제어시 사용하는 전류의 리밋을 설정함.
+#^%	;Sw?;	Owadda i, ddddz,	제에서 사용에는 단류의 다섯을 필요함.
+#^	;Szdddd1,dddd2;	Szdddd1,dddd2;	제어시 사용하는 전류의 리밋을 0.01A 단위로 설정함.
+#^	;Sz?;	Szuddu i , ddddz ,	MON NOOL ERA GYE 0.018 ENT EST.
		Svdddd1,dddd2,dddd3,dddd4,dddd5; (2011	
+#^%	;Svdddd1,dddd2,dddd3;	년 9월 11일 이전 S/W 버젼)	MOSFET의 gate를 off 시키는 전압 강하량과 과전압 및 검
+#^%	;Sv?;	Svdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,d	출시간의 크기를 설정함.
		ddd7; (2011년 9월 11일 S/W 버젼부터)	



적용 CPU	명령어	return value	operation
+#^% +#^%	;SC;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;	SCiiii1,iiii2; SCiiii1,iiii2,iiii3,iiii4;(RJM_VER5 STEP형 만 해당됨) SCiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (RJM_VER6 3상STEP형 만 해당됨)	전류제어 모드에서 전류명령값을 설정함.
% %	;Scdddd1,dddd2; ;Sc?;	Scdddd1,dddd2;	전류치를 암페어로 환산하는 scale factor를 설정함.
+#^ +#^	;Sciiii1,iiii2; ;Sc?;	Sciiii1,iiii2;	전류제어 모드에서 0.01A 단위로 전류명령값을 설정함.
% +#^ +#^ +#^%	;SViiii1,iiii2; ;SVIIII1,III12; (F2811/F28334인 경우) ;SVrIIII1,III12; (F2811/F28334인 경우) ;SV?;	SVIIII1,IIII2; SVIIII1,IIII2; (F2811/F28334인 경우)	속도제어 모드에서 속도명령값을 설정함.
+#^%	;Sadddd1,dddd2,dddd3,dddd 4; ;Sa?;	Sadddd1,dddd2,dddd3,dddd4;	속도제어 및 위치제어 모드에서 가속 및 감속율을 설정함.

적용 CPU	명령어	return value	operation
+#^% +#^% +#^%	RJM_VER5/6 STEP형인 경우: ;SHiiii1,iii12,iiii3,iiii 4,iiii5,iiii6; ;SHiiii1,iii12,iiii3,iiii 4,iiii5,iiii6,iiii7,iiii8 ; ;SH?;	SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; (RJM_VER5 STEP형 만 해당됨) SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6,i iii7,iiii8;(RJM_VER6 3상STEP형 만 해당 됨)	RJM_VER5/6 STEP형의 host direct electrical angle 제어 모드에서 구동전류의 진폭과 전류 OFFSET을 설정함.
+#^% +#^%	RJM_VER5/6 STEP형이 아닌 경우 : ;SHiiii1,iiii2; ;SH?;	SHiiii1,iiii2;	host direct electrical angle 제어 모드에서 구동전류 (RJM_VER7형의 3상/5상 STEP의 경우로서 작동모드 09번이 아닐때) 또는 구동전압의 진폭 (BLDC 구동 chip IRMCK201경우) 또는 PWM duty (DC 모터를 제외한 기타의 경우)를 설정함.
+#^% +#^%	스테핑모터의 경우 : ;Shdddd1,dddd2,dddd3,dddd 4; ;Sh?;	Shdddd1,dddd2,dddd3,dddd4;	스테핑모터에서 전류명령값의 최대치와 정지시의 전류치 비율을 설정함.
+#^ +#^	Wafer control의 경우 (2012년 2월 18일 S/W 버젼 부터 적용): ;Shdddd1,dddd2,dddd3,dddd 4; ;Sh?; (+#^)	Shdddd1,dddd2,dddd3,dddd4;	BLDC모터에서 holding torque를 만드는 파라메터를 설정함.



적용 CPU	명령어	return value	operation
	Wafer control이 아닌 BLDC		BLDC모터에서 holding torque를 만드는 파라메터를 설정
	모터의 경우 (2012년 2월		
	18일 S/W 버젼부터 적용) :	Shdddd1,dddd2,dddd3,dddd4;	
+#^	;Shdddd1,dddd2,dddd3,dddd	Silada i, adaz, adas, ada4,	함.
	4; (+#^)		
+#^	;Sh?; (+ #^)		
+#^%	;SPdddd1, 2, 3;	CD44441 11110 11110:	기교적은 적이라
+#^%	;SP?dddd1;	SPdddd1, IIII2, IIII3;	좌표점을 정의함.
+#^	;Sfdddd1,dddd2;	Sfdddd1,dddd2;	위치제어의 feedforward 게인을 설정함.
+#^	;Sf?;		귀시제어의 TeeuToTWaTu 게인들 설정함.
+#^%	;SeHLHL;	Colli III ·	motor1 과 motor2 에 대하여 PM, PP, PQ, Pp, Pq command
+#^%	;Se?;	SeHLHL;	를 적용가능토록 설정함.
+#^	;Sudddd1,dddd2;	Sudded dddo:	위치제어에서 정지위치 근방에서 강성을 높이기 위하여 제
+#^	;Su?;	Sudddd1,dddd2;	어 게인을 높게 하는 비율을 설정함.

적용 CPU	명령어	return value	operation
%	;SQHLHLHLHL,HLHLHLHL,HLHLH		
	LHL, HLHL, HLHLHLHL, HLHL, HLH		
	L,HLHLHLHL,HLHL,HLHLHL;	SQHLHLHLHL, HLHLHLHL, HLHLHLHL, HLHL, HLHLH	
+#^	;SQ또는	LHL, HLHL, HLHL, HLHLHLHL, HLHL, HLHLHLHL	
	SqHLHLHLHL, HLHLHLL, HLHLHL		모터1의 파라메터 리밋과 위치/속도/전류 제어파라메터를
	HL, HLHL, HLHLHLHL, HLHL, HLH	SQ또는	설정함. (WQ 명령어로 속도 및 전류제어 파라메터를 설정
	L, HLHLHLHL, HLHL, HLHL, HLHLH	SQHLHLHLHL, HLHLHLHL, HLHLHLL, HLHL, HLHLH	하는 경우는 제외)
	LHL; (F2811/F28334인 경	LHL, HLHL, HLHL, HLHLHLHL, HLHL, HLHL, HLHLHL	
+#^%	♀) ;SQ?;	HL; (F2811/F28334인 경우만)	
+#^	;Sq?; (F2811/F28334인 경		
'π	우만)		
%	;SRHLHLHLHL,HLHLHLHL,HLHLH		
	LHL, HLHL, HLHLHLHL, HLHL, HLH		
	L,HLHLHLHL,HLHL,HLHLHLHL;	00.	
+#^	;SR또는	SRHLHLHLHL, HLHLHLHL, HLHLHLHL, HLHL, HLHLHLHL	
	SpHLHLHLHL, HLHLHLHL, HLHLHL	LNL, NLNL, NLNL, NLNLNLL, NLNL, NLNLNLNL	모터2의 파라메터 리밋과 위치/속도/전류 제어파라메터를
	HL,HLHL,HLHLHLHL,HLHL,HLH	SR또는	설정함. (WR 명령어로 속도 및 전류제어 파라메터를 설정
	L,HLHLHLHL,HLHL,HLHLH	SPHLHLHL, HLHLHLHL, HLHLHLHL, HLHL, HLHLH	·
	LHL; (F2811/F28334인 경	LHL, HLHL, HLHL, HLHLHLHL, HLHL, HLHL, HLHLHL	
	우)	HL; (F2811/F28334인 경우만)	
+#^%	;SR?;		
+#^	;Sp?; (F2811/F28334인 경		
	우만)		



적용 CPU	명령어	return value	operation
+#^	;AQ또는		
	AqHLHL, HLHLHLHL, HLHL, HLHL,	AQ또는	모터1의 위치/속도 제어파라메터를 설정함. (WQ 명령어로
	HLHLHLHL,HLHL; (+#)	AqHLHL, HLHLHLL, HLHL, HLHL, HLHLHLHL, HLHL	속도제어 파라메터를 설정하는 경우는 제외)
+#^	;AQ?; (+#)	;	그도세이 퍼디메디를 걸음하는 용무는 세되/
+#^	;Aq?; (+#)		
+#^	;AR또는		
	Arhlhl, Hlhlhlhl, Hlhl, Hlhl,	AR또는	모터2의 위치/속도 제어파라메터를 설정함. (WR 명령어로
	HLHLHLHL,HLHL; (+#)	Arhlhl, Hlhlhlhl, Hlhl, Hlhl, Hlhlhlhl, Hlhl	속도제어 파라메터를 설정하는 경우는 제외)
+#^	;AR?; (+#)	;	12MM MUMUE EOOL OIL MA/
+#^	;Ar?; (+#)		
+#^	;AS또는		
	ASHLHL, HLHLHLHL, HLHL, HLHLH		모터1,2의 전류 제어파라메터를 설정함. (WQ, WR 명령어로
	LHL;	ASY = ASH HI	전류제어 파라메터를 설정하는 경우는 제외)
+#^	;AS?;		
+#^	;As?;		
+#^	;AIdddd1,dddd2,dddd3,dddd4		PA 또는 SV 명령에 의한 가감속시에 전류명령값의
	;	Aldddd1,dddd2,dddd3,dddd4;	feedforward 값을 계산하는 scale factor를 설정함.
+#^	;A1?;		
+#^	;Aidddd1,dddd2;	Aidddd1,dddd2;	PA 명령에 의한 가속시에 사다리꼴 모양의 가속 프로파일
+#^	;Ai?;		을 만들기 위한 가속 변화율값을 설정함.
+#^	;ANdddd1,dddd2;	ANdddd1,dddd2;	모터1에 적용되는 Notch filter의 중심주파수와 Q factor
+#^	;AN?;		를 설정함.
+#^	;Andddd1,dddd2;	Andddd1,dddd2;	모터2에 적용되는 Notch filter의 중심주파수와 Q factor
+#^	;An?;	·	를 설정함.
#	;ALdddd1,dddd2,dddd3,dddd4		
	;	ALdddd1,dddd2,dddd3,dddd4;	28~31번 butterworth filterA의 시정수를 설정함.
#	AL?;		



적용 CPU	명령어	return value	operation
+#^	;ATdddd1;	ATdddd1;	PA 명령의 완료 시점에서 추가로 실행하는 기능을 위한 시
+#^	;AT?;	ATada T,	간파라메터를 설정함.
+#^	;ABA55A,dddd1;		위치제어에서 정지하고 있을때, jitter와 소음을 줄이거나
+#^	;AB?;	ABdddd1;	또는 강성을 높이기 위하여, 위치제어 게인을 낮추거나 높
` ''	7710: 7		이기 위한 파라메터를 설정함.
+#^	;AwA55A,dddd1,dddd2; ;Aw?;	Awdddd1,dddd2;	사용가능한 최대크기의 PWM duty 비를 설정함.
+#^	;ADA55A, iiii1, iiii2; ;AD?;	ADiiii1,iiii2;	MOTOR 구동 Half-bridge의 dead-time을 보상하되, Z6 명령 어로 설정하는 dead-time에 추가하여 보상하려는 보상량을
			설정함.
+	;Acdddd1,dddd2;	Acdddd1,dddd2;	엔코더펄스 카운트 방식으로 속도를 계산할 때 moving
+	;Ac?;	Thousant, adda.	average 구간의 크기를 설정함.
#	;Acdddd1,dddd2,dddd3,dddd4		엔코더펄스 카운트 방식으로 속도를 계산할 때 moving
π	·	Acdddd1,dddd2,dddd3,dddd4;	average 구간의 크기와, 엔코더펄스의 주기를 사용하여 속
#	;Ac?;	nouddd i , ddddz , ddddo , dddd+ ,	도를 계산할때 주기계산에 사용되는 엔코더펄스개수의 2에
π	,nu:,		대한 지수를 설정함.

적용 CPU	명령어	return value	operation
+#^	;AX or		
	AxA55A,dddd1,dddd2,dddd3,d	AX or	
	ddd4,dddd5,dddd6,dddd7,ddd	AxA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dd	모터1의 상위 or 하위 속도에서 제어 게인을 높이거나 낮
	d8,dddd9,dddd10,dddd11,ddd	dd6,dddd7,dddd8,dddd9,dddd10,dddd11,ddd	추는 파라메터를 설정함.
	d12,dddd13,dddd14;	d12,dddd13,dddd14;	
+#^	;AX or Ax?;		
+#^	;AY or		
	AyA55A,dddd1,dddd2,dddd3,d	AY or	
	ddd4,dddd5,dddd6,dddd7,ddd	AyA55A,dddd1,dddd2,dddd3,dddd4,dddd5,dd	모터2의 상위 or 하위 속도에서 제어 게인을 높이거나 낮
	d8,dddd9,dddd10,dddd11,ddd	dd6,dddd7,dddd8,dddd9,dddd10,dddd11,ddd	추는 파라메터를 설정함.
	d12,dddd13,dddd14;	d12,dddd13,dddd14;	
+#^	;AY or Ay?;		
+#^	;AZA55A,dddd1,dddd2;	AZdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	SQ/SR/Sq/Sr/AQ/AR/Aq/Ar/WQ/WR 등의 명령으로 설정하는
+#^	;AZ?;	/ Zadaz, addaz, adda-, adda-, adda-,	게인이 적용되는 중간 속도값을 설정함.
+#^	;AzA55A;	AzA55A;	AX/Ax/AY/Ay/AZ 명령에 의한 설정값을 EEPROM에 저장함.

적용 CPU	명령어	return value	operation
+#^%	;Stdddd1,dddd2,dddd3;	011111111111111111111111111111111111111	PP/Pp/PQ/Pq 명령어의 update 주기와 PP/Pp/PQ/Pq 명령어
+#^%	;St?;	Stdddd1,dddd2,dddd3;	미수신의 연속허용횟수를 설정함.
+#^%	;SrHLHL,HLHL;	SrHLHL,HLHL;	
+#^%	;SrHLHL,HLHL,HLHL;	SrHLHL, HLHL, HLHL;	Pp/Pq/p 명령어에 대한 return값의 주소를 설정함.
+#^%	;Sr?;		
+#^%	;SZdddd1,dddd2;		Hall sensor pulse의 지정된 edge를 기준으로 하여 엔코더
+#^%	;SZ?;	SZdddd1,dddd2;	Z펄스의 위치를 앤코더펄스 갯수로 읽어 냄. (BLDC 구동
	,02:,		chip IRMCK201 전용)
+#^%	;SBIIII1;	SBIIII1; (SB?; 인 경우에만 echo 있음)	RS232C 직렬통신속도값을 설정함.
+#^%	;SB?;	COLLINIA (COLIN E OF WILL COMO ME)	TOLOGO TEOURITE EOO.
+^%	;Sb 1;	 Sb 1; (Sb?; 인 경우에만 echo 있음)	 RS485 직렬통신속도값을 설정함.
+^%	;Sb?;	ODITITION (OD:) E STUIL GOID XB)	INTO TEOUTAME EOD.
#	;Sb 1, 2;	SbIIII1,IIII <mark>2</mark> ; (Sb?; 인 경우에만 echo	 RS485 와 RS232C extra 직렬통신속도값을 설정함.
#	;Sb?;	있음)	INTO A INCLUCO WITH TEOURILE EOD.
+#^%	;SFdddd1,dddd2,dddd3;	SFdddd1,dddd2,dddd3;	butterworth filter의 시정수와 변화율 상한값을 설정함.
+#^%	;SF?dddd1;	or dadar, addaz, dadao,	CONTROL OF A CHE OCKE EOC.
	2012년 10월 03일 이전 S/W		
+#^%	버젼;Sidddd1,dddd2;	Sidddd1,dddd2;	
+#^%	;Si?;		RJM_VER1과 RJM_VER2의 경우에는 과전류에러 fault 진단을
	2012년 10월 03일 S/W 버젼부		위한 1.024초 동안의 과전류검출 횟수를 설정하며, 그외의
+#^	터		경우는 SI 명령어로 설정한 전류치보다 실제전류가 연속적
+#^	;Sidddd1,dddd2; 또는	;Sidddd1,dddd2,dddd3,dddd4;	으로 초과하기를 허용하는 최대시간을 설정함.
+#^	;Sidddd1,dddd2,dddd3,dddd4;		
1 11	;Si?;		
+#^%	;Sxdddd1,dddd2,dddd3,dddd4,	Sxdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,	제어오차가 과도하게 큰 상태를 진단하기 위한 위치오차,
	dddd5,dddd6,dddd7,dddd8;	dddd7,dddd8;	속도오차, 전류오차, 허용시간을 설정함. (RJM_VER5에는
+#^%	;Sx?;	dudu/, ddudo,	해당없음)
+#^%	;SmHLHL;	SmHLHL;	명령어에 대한 return값을 보낼 때 사용하는 host의 주소
+#^%	ESm?;		를 설정함. 395

적용 CPU	명령어	return value	operation
+#^	;WQdddd1,dddd2,dddd3,dddd		모터1의 03/06번 위치/속도제어모드에서 속도게인과 ,
	4,dddd5;	WQdddd1,dddd2,dddd3,dddd4,dddd5;	RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터1의 전류제어
+#^	;WQ?;		파라메터를 설정함.
+#^	;WRdddd1,dddd2,dddd3,dddd		모터2의 03/06번 위치/속도제어모드에서 속도게인과 ,
	4,dddd5;	WRdddd1,dddd2,dddd3,dddd4,dddd5;	RJM_VER7형의 3상/5상 STEP모터와 BLDC 모터2의 전류제어
+#^	;WR?;		파라메터를 설정함.
+#^	;WBdddd1,dddd2;	WBdddd1,dddd2;	RJM_VER7 제어시에 100KHz interrupt routine에서 사용하
+#^	;WB?;	mbadad i, addaz i	는 Butterworth filter의 시정수를 설정함.
			BLDC 모터1,2에 있어서 Hall_A상 상승에지에 대한 U상 전
+#^	;WEdddd1,dddd2;	WEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	압 positive zero crossing 포인트의 위상지연 값을 설정
+#^	;WE?;	meddal, addae, addal, addae, addae,	하며, 모터1,2의 POLE_PAIR 수, 모터1,2의
			MOTORx_POSITION_SCALE_FACTOR를 읽어서 host로 전송함.
+#^	;WGdddd1,dddd2,dddd3,dddd		analog Hall sensor type의 RJM_VER7 모터1의 센서 scale
	4, dddd5, dddd6;	WGdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	과 offset을 설정함.
+#^	;WG?;		
+#^	;WHdddd1,dddd2,dddd3,dddd		analog Hall sensor type의 RJM_VER7 모터2의 센서 scale
	4, dddd5, dddd6;	WHdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	과 offset을 설정함.
+#^	;WH?;		
+#^	;WCdddd1,dddd2;	WCdddd1,dddd2;	analog Hall sensor type의 RJM_VER7 모터에 대하여 위상
+#^	;WC?;	moddd i, dddae	각 측정용 파라메터의 설정을 개시함.
+#^	;WXiiii1,iiii2;	WX;;;;1,;;;2;	RJM_VER7 모터의 electrical angle을 강제로 설정함.
+#^	;WX?;		-
#	;WNdddd1,dddd2;	WNdddd1,dddd2;	RJM_VER7 analog Hall sensor type에서 구동가능한 영역에
#	;WN?;	mada i, adde i	서의 analog Hall sensor wave 총갯수를 설정함



적용 CPU	명령어	return value	operation
+#^	;WJdddd1,dddd2,dddd3,dddd		
	4,dddd5;	WJdddd1,dddd2,dddd3,dddd	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파
+#^	;WJ?;	4,dddd5,dddd6;	라메터 설정을 개시함.
	(제일메티컬의 경우)		
#^	;WJdddd1;		
#^	;WJ?;	WJdddd1,dddd2;	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파
	(나노모션의 linear Hall	wJadaa i, adaaz,	라메터 설정을 개시함.
	sensor의 경우)		
+#^	;WJdddd1,dddd2,dddd3,dddd		
	4,dddd5;	WJdddd1,dddd2,dddd3,dddd	analog Hall sensor type BLDC 모터의 linear interpolation 방식 자동 파
+#^	;WJ?;	4,dddd5,dddd6;	라메터 설정을 개시함.
	(그 외의 경우)		
+#^	;WKdddd1;	MICARA AAAAO	encoder/digital_Hall_IC_only type BLDC 모터의 자동 파라메터 설정을 개
+#^	;WK?;	WKdddd1,dddd2;	시함.
+#^	;Wqdddd1,dddd2;	Wadddd ddddo:	마다 1이 MAY DEDCENT CLID1 - ELECTRIC TIME CONCTANT1 가운 성정하
+#^	;Wq?;	Wqdddd1,dddd2;	모터1의 MAX_PERCENT_SLIP1, ELECTRIC_TIME_CONSTANT1 값을 설정함.
+#^	;Wrdddd1,dddd2;	Wrdddd1,dddd2;	미디OOL MAY DEDOENT OF IDO FLECTOR TIME CONSTANTS 가운 성정하
+#^	;Wr?;	wi dada i , dadaz ,	모터2의 MAX_PERCENT_SLIP2, ELECTRIC_TIME_CONSTANT2 값을 설정함.
+#^	;WLdddd1,dddd2,dddd3,dddd	MI dadat dadao dadao dada	그사 analag Hall 센크로 사용하는 결혼에 4/0/0 사사들이 교원적가의 성적
	4, dddd5, dddd6, dddd7;	WLdddd1, dddd2, dddd3, dddd	3상 analog Hall 센서를 사용하는 경우에 A/B/C 상신호의 교차점값을 설정
+#^	;WL?dddd1;	4,dddd5,dddd6,dddd7;	함.
+#^	·MI Dadadi	WI D.	3상 analog Hall 센서를 사용하는 경우에 지정된 삼상 wave sector 번호부
T#' '	;WLPdddd1;	WLP;	터 A/B/C 상신호의 교차점값을 자동 설정함.
1.44	· NAI O ·	WI O:	3상 analog HaⅡ 센서를 사용하는 자석가동자의 경우에 -방향쪽 센서와 +
+#^	;WLQ;	WLQ;	방향쪽 센서의 위상차를 자동 설정함.
			3상 analog Hall 센서를 사용하는 자석가동자의 경우에 순위가 높은 제어
+#^	;WLR;	WLR;	기의 +방향쪽 센서와 순위가 낮은 현재 제어기의 -방향쪽 센서 사이의 위
	п		상차를 자동 설정함.
CUB	E		397

적용 CPU	명령어	return value	operation
+#^	;WSdddd1,dddd2;		위치제어모드에서 속도에 비례하는 damping 회전력을 발생
+#^	;WS?;	WSdddd1,dddd2;	시키는 게인을 설정함.
+#^	;Wsdddd1,dddd2;		DD motor 등에서 속도에 비례하는 damping current를 발생
+#^	;Ws?;	Wsdddd1,dddd2;	시키는 게인을 설정함.
+#^	;Wciiii1,iiii2;	W	
+#^	;Wc?;	Wciiii1,iiii2;	유도모터의 경우에 ld 전류의 기준값을 설정함.
+#^	;WTdddd1,dddd2,dddd3,dddd		BLDC 모터에 있어서 cogging torque를 보상하는 파라메터
	4;	WTdddd1,dddd2,dddd3,dddd4;	를 설정함.
+#^	;WT?;		= 208.
+#^	;Wkdddd1,dddd2; (2011년 6		
	월 14일 이전 S/W 버젼)	Wkdddd1,dddd2; (2011년 6월 14일 이전	DIM VEDZ DIDE FIELDINI LI FIELDINI direct
+#^	;Wkdddd1,dddd2,dddd3,dddd	S/W 버젼)	RJM_VER7 BLDC 모터제어기에서, 모드8의 host direct angle control을 사용하여 자동튜닝 등의 구동을 할 때, 사용하는 PWM duty 값을 설정함.
	4,dddd5; (2011년 6월 14일	Wkdddd1,dddd2,dddd3,dddd4,dddd5; (2011	
	S/W 버젼부터)	년 6월 14일 S/W 버젼부터)	
+#^	;Wk?;		
+#^	;WPdddd1,dddd2;	WD 1111 1110.	Digital Hall sensor type의 RJM_VER7 모터에 대하여 정지
+#^	;WP?;	WPdddd1,dddd2;	희망 edge 번호를 지정함.
+#^	;Wpdddd1,dddd2,dddd3,dddd		Digital Hall sensor type의 RJM_VER7 모터에 대하여 감속
T#**		Woodadd dddd dddd dddd	완료후 안정화를 개시하는 지연시간과 안정화용 analog
. 44	4, dddd5;	Wpdddd1,dddd2,dddd3,dddd4,dddd5;	Hall sensor의 offset, 및 analog Hall 센서가 사용되는
+#^	;Wp?;		위치오차 범위등을 설정함.
+#^	;WMHLHL;		Wafer 회전위치제어시에 위치모드로 정지한 한 상태에서
		WMHLHL;	속도모드로 작동모드를 변경하며, 이에 따른 변수초기화
+#^	;WM?;		작동을 실시함.
+#^	;WDA55A,dddd1,dddd2;	WD-11-11 -1-10.	미터가 게시요 사기되는 기가되로 성적된
+#^	;WD?;	WDdddd1,dddd2;	미분값 계산을 실시하는 시간차를 설정함.



적용 CPU	명령어	return value	operation
			inner 속도제어루프가 있는 03번 위치제어모드의 속도제어
+#^	• Maaaaaa aaaaa aaaa		또는 Digital Hall sensor만을 사용하는 BLDC모터의 위치/
T#* *	;Wddddd1,dddd2,dddd3,dddd	WA	속도 제어 또는 OA번 속도제어모드 또는 속도제어로서 SX
1.44	4, dddd5, dddd6;	Wddddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	명령어로 설정하는 두번째 파라메터의 bit11이 0인 경우에
+#^	;Wd?;		, 낮은 속도에서 위치/속도 제어게인(3번 위치제어모드의
			경우는 속도게인만)을 낮게하는 파라메터를 설정함.
+#^	;RIA55A;	RIA55A;	모터 1/2의 적분오차값을 강제로 0으로 설정함.
+#^	;RIA5A5;	RIA5A5;	모터 1의 적분오차값을 강제로 0으로 설정함.
+#^	;RI5A5A;	R15A5A;	모터 2의 적분오차값을 강제로 0으로 설정함.
+#^	;RrA55A;	N.A.	모터제어기를 리셋함.
+#^	;RSHLHL;	RSHLHL;	비상정지시의 정지 형태를 정의함.
+#^	;RS?;	NOTICAL,	미성성자시의 성지 성대를 성의함.
+#^	;RsA55A;	RsA55A;	모터1과 모터2가 명령에 의한 비상정지를 실행함.
+#^	;RsA5A5;	RsA5A5;	모터1이 명령에 의한 비상정지를 실행함.
+#^	;Rs5A5A;	Rs5A5A;	모터2가 명령에 의한 비상정지를 실행함.
+#^	;RBHLHL;	001111111111111111111111111111111111111	feedback 제어를 개시하도록 작동모드 설정에 필요한 조치
+#^	;RB?;	RBHLHL;	를 자동으로 처리하게 함.
			fault code를 저장한 RAM buffer를 읽어서 fault의 내용을
+#^	;RJI;	RJn <aaaaaaaaaaa;< td=""><td>문자로 보여주는 index를 가장 최근의 fault code를 지정</td></aaaaaaaaaaa;<>	문자로 보여주는 index를 가장 최근의 fault code를 지정
			하도록 초기화함.
1.44	.0.1.		fault code를 저장한 RAM buffer를 읽어서 fault의 내용을
+#^	;RJ;	RJn <aaaaaaaaaaa;< td=""><td>문자로 보여줌.</td></aaaaaaaaaaa;<>	문자로 보여줌.
		Dinald<	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문
+#^	;Rj ;	Rjnold <aaaaaaaaa>; 또는</aaaaaaaaa>	자로 보여주는 index를 가장 최근의 fault code를 지정하
		Rjnnew <aaaaaaaaaa>;</aaaaaaaaaa>	도록 초기화함.
140	·D: ·	Rjnold <aaaaaaaaaaa>; 또는</aaaaaaaaaaa>	EEPROM에 저장된 fault code를 읽어서 fault의 내용을 문
+#^	;Rj;	Rjnnew <aaaaaaaaaa>;</aaaaaaaaaa>	자로 보여줌.
CUB	Ē		399

적용 CPU	명령어	return value	operation
+#^	;XSHL;	XSHLHLHLHL;	순서제어에서 command를 설정하거나 제어상태를 읽어냄.
+#^	;XS?;	ASHLILLILI,	(프로아텍 전용)
+#^	;XViiii1,iiii2,iiii3;	XViiii1,iiii2,iiii3;	순서제어에서 사용하는 X_speed1_low, X_speed1_mid,
+#^	;XV?;	AVIIIII, IIII2, IIII3,	X_speed1_high 를 설정함. (프로아텍 전용)
+#^	;XOHL,HL;	XOHL, HL;	EXT_FAULT1_port와 EXT_FAULT2_port를 강제로 set 또는
+#^	;X0?;	NOTIL , TIL ,	clear 함.
#^	;Xodddd1;	Xodddd1;	스테핑모터에서 50V 승압회로와 역기전력 해소 pass를
#^	;Xo?;	Aoudud ()	ON/OFF 함.
#^	;Xodddd1;	Xodddd1;	380V 용 모터제어기에서 PRECHARGE_RELAY, MAIN_RELAY,
#^	;Xo?;	Aoudud ()	BRAKE_RELAY를 강제 설정함.
+#^	;Xedddd1,dddd2;	Xedddd1,dddd2;	스테핑모터에서 DC-DC converter의 승압전압의 크기와
+#^	;Xe?;	Aedddd 1, dddd2,	turn-off에 소요되는 시간을 설정함.
+#^	;XEdddd1,dddd2; or	XEdddd1,dddd2;	스테핑모터 또는 절대각엔코더를 사용하는 경우 엔코더의
+#^	;XE 1, 2;	or XEIIII1, IIII2;	- 대 8 보다 보는 클레 국 전 보다를 제 8 하는 8 부 전 보다 크 4 체 배 된 펄 스 수를 설정함.
+#^	;XE?;	OI ALTITITIES	TAINIC ELTE ESB.
+#^	;XPXAAAAAAAAAA;	XPXAAAAAAAAAA;	시퀜스제어 프로그램의 현재 포인터 위치에 시퀜스제어 프
111	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	AI AAAAAAA	로그램을 입력함.
			시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제
+#^	;XP?;	XPAAAAAAAAA;	어 프로그램 한 스텝을 보여주고, 포인터 위치는 변경하지
			않음.
			시퀜스제어 프로그램의 현재 포인터 위치에 있는 시퀜스제
+#^	;XP+;	XPAAAAAAAAA;	어 프로그램 한 스텝을 보여주고, 포인터 위치를 다음 스
			텝위치로 이동함.
			시퀜스제어 프로그램의 포인터 위치를 강제 설정하고, 설
+#^	;XPdddd1;	XPAAAAAAAAA;	정된 포인터 위치의 시퀜스제어 프로그램 한 스텝을 보여
			주고, 포인터 위치는 변경하지 않음.



적용 CPU	명령어	return value	operation
+#^	;XPp;	XPpdddd1;	현재의 시퀜스제어 프로그램 포인터 위치를 보여줌.
+#^	;XPsA55A;	XPsA55A;	메모리상의 시퀜스제어 프로그램을 EEPROM으로 저장함.
+#^	;XPT;	XPTHLHL, HLHL;	현재의 PORT_STATUS_FOR_X_CONTROL와 X_PORT_STATUS를 보
Т#	, , , , , , , , , , , , , , , , , , , ,	Ar Inche, nche,	여줌.
+#^	;XPt;	XPtHLHL, HLHL;	현재의 X_timer0, X_timer0_period, X_counter0의 값을 보
T#	יארני		여줌.
+#^	;XPP;	XPPHLHL, HLHL;	현재와 직전의 시퀜스제어 프로세스 카운트를 보여줌.
+#^	;XPF;	XPFHLHL;	현재의 X_STATUS_FLAG를 보여줌.
+#^	;XPS;	XPSHLHL, HLHL;	한스텝씩 시퀜스제어 프로그램을 실행시키는 모드일 때 한
Т#	,,,,,		스텝 씩 시퀜스제어 프로그램을 실행하게 함.
+#^	;XTAAAAAAAAAA;	XTAAAAAAAAAA;	text의 현재 포인터 위치에 text를 저장하며 저장된 text
T#	, 17777777		를 보여줌.
			text의 포인터 위치를 강제 설정하고 해당 위치에 있는
+#^	;XT?dddd1;	XTdddd1, AAAAAAAAAA;	text 한 줄을 보여주고, 포인터 위치는 강제 설정된 값을
			유지함.
+#^	;XT?;	XTAAAAAAAAAA;	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고,
ιπ	,,,,	ATAMANA	포인터 위치는 변경하지 않음.
+#^	;XT+;	XTAAAAAAAAAA;	text의 현재 포인터 위치에 있는 text 한 줄을 보여주고,
· 11	,,,,,,	X1/VVVVV/VVV	포인터 위치를 다음 줄 위치로 이동함.
+#^	;XTp;	XTpdddd1;	현재의 text 포인터 위치값를 보여줌.
+#^	;XTsA55A;	XTsA55A;	메모리상의 text 를 EEPROM으로 저장함.



적용 CPU	명령어	return value	operation
+#^	2012년 12월 19일 이전 버 전인 경우 ;Z0iiii1; ~ ZFiiii1; (단 Z3HLHL; Z4HLHL; Z5HLHL; Z6HLHL; Z7HLHL; Z8HLHL; Z9HLHL; ZAHLHL; ZBHLHL; ZCHLHL; 는 예외) (+#^) 2012년 12월 19일 S/W 버젼 부터 ;Z0A55A,iiii1; ~ ZFA55A,iiii1; (단 Z3A55A,HLHL; Z4A55A,HLHL; Z5A55A,HLHL; Z6A55A,HLHL; Z7A55A,HLHL; Z8A55A,HLHL; ZBA55A,HLHL; ZAA55A,HLHL; ZBA55A,HLHL; ZAA55A,HLHL; E 예외)	ZO(~F)HLHL,HLHL,HLHL,HLHL,HLHL,HLHL L,HLHL,HLHL,HLHL,HLHL,HLHL,HLHL,HL HL,HLHL; HLHL,HLHL,	0 ~ F 까지의 숫자로 지정된 위치의 FACTORY SETTING parameter RAM에 값을 저장함.
+#^	;Zr;	Zrhlhl, Hlhl, Hlhl	16개의 FACTORY SETTING parameter 값을 읽어서 보여줌 (RAM 영역의 값).
+#^	;ZsA55A;	ZsA55A;	메모리상의 FACTORY SETTING parameter 를 EEPROM으로 저 장함.



적용 CPU	명령어	return value	operation
+#^	;ZQdddd1,dddd2,dddd3,dddd		
	4,dddd5,dddd6,dddd7,dddd8	ZQdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,d	모터1의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1,
	;	ddd7,dddd8;	Rth2, Tau1, Tau2)를 설정함.
+#^	;ZQ?;		
+#^	;ZRdddd1,dddd2,dddd3,dddd		
	4,dddd5,dddd6,dddd7,dddd8	ZRdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,d	모터2의 특성파라메터(저항, 인덕턴스, Kt, Kb, Rth1,
	,	ddd7,dddd8;	Rth2, Tau1, Tau2)를 설정함.
+#^	;ZR?;		
+#^	;ZTdddd1,dddd2;	ZTdddd1,dddd2;	모터1과 모터2의 최대허용 권선온도상승값을 설정함.
+#^	;ZT?;	2144441,44442,	TUIM TULA HUMO CCCTOOMS SOO.
+#^	;Zcdddd1,dddd2;		IRMCK201을 사용한 RJM_VER3에 있어서 모터1과 모터2의
+#^	;Zc?;	Zcdddd1,dddd2;	nominal current(IRMCK201 파라메터 설정에 사용한 값)를
111	,20:,		설정함.
+#^	;ZPdddd1; (+#)	ZPdddd1;	역기전력 해소용 전력저항의 최대 ON time을 설정함.
+#^	;ZP?; (+#)	Zi uddd i y	그거든 이 에도 이 근 기사이가 되네 에 대배를 들었다.
+#^	;Zbdddd1;	Zbdddd1;	CAN 통신속도를 설정함.
+#^	;Zb?;	Zbuddu i,	CAN 중인국工을 열정함.



적용 CPU	명령어	return value	operation
+#^	;DZHLHL,HLHL,HLHL;	N7U U U U U U U U .	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에
+#^	;DZ?;	DZHLHL, HLHL, HLHL;	표시될 값의 주소를 설정함.
+#^	;DFHLHL,HLHL,HLHL;	DFHLHL, HLHL, HLHL;	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에
+#^	;DF?;	OFFICIE, FICHE, FICHE,	표시될 값의 type과 표시형식을 설정함.
+#^	;DSfffff1,ffff2,ffff3,ffff		컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에
	4 ;	DSffff1,ffff2,ffff3,ffff4;	표시될 값의 scale factor를 설정함.
+#^	;DS?;		
+#^	;DTdddd1,AAAAAAAAA;	DTdddd1, AAAAAAAAA;	컴파일 테크놀로지사의 직렬통신 4x20 영문 디스플레이에
+#^	;DT?dddd1;	Diddui, nnnvvnn,	표시되는 값의 타이틀을 설정함.
+#^	;DDdddd1,dddd2,dddd3; or		
+#^	;DDdddd1,dddd2,dddd3,dddd	DDdddd1,dddd2,dddd3; or	일정스텝으로 이동을 반복하는 데모작동의 파라메터를 설
	4;	DDdddd1,dddd2,dddd3,dddd4;	정함.
+#^	;DD?;		
+#^	;DEHLHL;	DEHI HI:	일정스텝으로 이동을 반복하는 데모작동의 작동모드를 설
+#^	;DE?;	OLI ILI ILI	정함.
+#^	;Dddddd1;		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이
+#^	;Dd?;	Dddddd1;	는 모드에서 속도명령값 또는 전류명령값의 deadband 크기
'''	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,		를 설정함.
+#^	;DOA55A,dddd1,dddd2;		포텐시오메터로 속도명령 또는 전류명령을 받아서 움직이
+#^	;D0?;	DOdddd1,dddd2;	는 모드에서 속도명령값 또는 전류명령값이 0일때의 LSxC
111	,00:,		포트 A/D 변환값을 설정함.
+#^	;DXHLHL,HLHLHL;	DXHLHL, HLHL, HLHL, HLHL;	ON/OFF 출력과 7-seg 표기값을 설정하고 ON/OFF 입력과
+#^	;DX?;	UANLINE, NENE, NENE, NENE,	switch 입력을 읽어서 host로 전송함.



적용 CPU	명령어	return value	operation
+#^	;Tsdddd1,dddd2;	Toddad daddo:	모터1의 step 구동을 위한 주기와 진폭을 설정하고 step
+#^	;Ts?;	Tsdddd1,dddd2;	구동을 실행하도록 함.
+#^	;Ttdddd1,dddd2;	Ttdddd1,dddd2;	모터2의 step 구동을 위한 주기와 진폭을 설정하고 step
+#^	;Tt?;	TtadadT, adda2,	구동을 실행하도록 함.
+#^	;Tudddd1,dddd2;	Tudddd1,dddd2;	모터1의 사인파 구동을 위한 주기와 진폭을 설정하고 사인
+#^	;Tu?;	Tudada T, adda2,	파 구동을 실행하도록 함.
+#^	;Tvdddd1,dddd2;	Tvdddd1,dddd2;	모터2의 사인파 구동을 위한 주기와 진폭을 설정하고 사인
+#^	;Tv?;	Tvadad I, addaz,	파 구동을 실행하도록 함.

적용 CPU	명령어	return value	operation
+#^	;MTffff1,ffff2,ffff3;	MTffff1,ffff2,ffff3;	온도측정용 센서 파라메터를 설정함.
+#^	;MT?;	MIIIII, IIII2, IIII3,	근도국장중 센시 파다메니글 물장됨.
+#^	;Mtdddd1;	Mtdddd1,dddd2,iiii3;	온도측정용 센서를 사용하여 측정된 저항치 및 온도값을
+#^	;Mt?;	Mitada I, ada 2, 11113,	읽어내어 host로 전송함.
+^	;MUfffff1,ffff2,ffff3;	MUfffff1,ffff2,ffff3;	온도측정용 두번째 센서 파라메터를 설정함. -
+^	;MU?;	MOTITITI,	근도국항등 구인째 엔지 파다메니글 일정함.
+^	;Mudddd1;	Mudddd1,dddd2,iiii3;	온도측정용 두번째 센서를 사용하여 측정된 저항치 및 온
+^	;Mu?;	Muddad1, adda2,11113,	도값을 읽어내어 host로 전송함.
+#^	;MPfffff1,ffff2,ffff3,ffff		
	4 , ffff5, ffff6, ffff7, ffff	MPffff1, ffff2, ffff3, ffff4, ffff5, ffff6, f	4개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용하여 X-moment, Y-moment, Z-force를 계산하는 변환행렬 값을 설정함.
	8, ffff9, ffff10, ffff11, fff		
	f12;		
+#^	;MP?;		
+#^	;Mpfffff1,ffff2,ffff3,ffff		6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용 하여 X-moment, Y-moment, Z-force, X-force를 계산하는
	4 , ffff5, ffff6, ffff7, ffff	;Mpfffff1,ffff2,ffff3,ffff4,ffff5,ffff6,	
	8,ffff9,ffff10,ffff11,fff	ffff7, ffff8, ffff9, ffff10, ffff11, ffff12;	
	f12;		변환행렬 값을 설정함.
+#^	;Mp;		
+#^	;Mqffff1,ffff2,ffff3,ffff		
	4 , ffff5, ffff6, ffff7, ffff	;Mqffff1,ffff2,ffff3,ffff4,ffff5,ffff6,	6개의 strain guage 신호를 증폭하고 AD 변환된 값을 이용
	8, ffff9, ffff10, ffff11, fff	fffff7,ffff8,ffff9,ffff10,ffff11,fffff12;	하여 Y-force, Z-moment를 계산하는 변환행렬 값을 설정함.
	f 12;		
+#^	;Mq;		



적용 CPU	명령어	return value	operation
+#^	;MOdddd1,dddd2,dddd3,dddd4; ;MO?; (F/T센서의 경우)	MOdddd1,dddd2,dddd3,dddd4;	F/T 센서에서 TLV5630 또는 DAC124S085 D/A converter의 출력값을 설정함.
#	;M0;iii1,iiii2; ;M0?; (절대각센서를 사용하는 경 우)	MOiiii1,iiii2;	절대각센서를 사용하는 경우에 BLDC 모터의 Hall A/B/C상 신호와 모터위상각신호의 위상차를 없애기 위한 위상 offset을 설정함.
+#^	;MSiiii1,iiii2,iiii3,iiii4;;MS?;	MSiiii1,iiii2,iiii3,iiii4;	F/T 센서에서 증폭후 A/D 변환된 스트레인 값의 온도증가에 따른 변화율을 설정함.
+#^	;MViiii1,iiii2; ;MV?;	MViiii1,iiii2;	F/T 센서에서 온도보상의 기준온도와 온도보상 기울기를 자동 계산할 때 사용한 또 다른 온도값을 설정함.
+#^	;Moiiii1,iiii2,iiii3,iiii4, iiii5; ;Mo?;	Moiiii1,iiii2,iiii3,iiii4,iiii5;	F/T 센서에서 수직력의 auto zero rate, maximum allowed drift, X 모멘트 coupling, Y 모멘트 coupling, FORCE_SENSOR_ERROR_DETECTION_TIME 파라메터를 설정함.
+^	;Msdddd1,dddd2; ;Ms?;	Msdddd1,dddd2;	삼성메디컬용 F/T 센서의 X-moment, Z-force coupling을 처리하는 파라메터를 설정함.
+#^ +#^	;MAdddd1,dddd2,dddd3; ;MA?;	MAdddd1,dddd2,dddd3;	자동화용 H/W 버젼에서 상위명령값과 실제값의 차이가 설 정값이내에 들어왔는지를 판단하고 mission 완료를 결정하 는 경계값(threshold)을 설정함.
+#^ +#^	;MC; ;MC?;	MCiiii1,iiii2,iiii3,iiii4;	BLDC 또는 AC servo 모터의 Iq, Id 값을 읽어서 host로 전송함.
+#^	;Mc; ;Mc?;	Mciiii1,iiii2,iiii3,iiii4;	BLDC 또는 AC servo 모터의 lq, ld 값을 읽어서 0.01A 단위의 암페어값으로 환산하여 host로 전송함.
+#^	;MRdddd1,dddd2,dddd3,dddd4;;MR?;	MRdddd1,dddd2,dddd3,dddd4;	전류상승률을 제한하는 파라메터를 설정함.



적용 CPU	명령어	return value	operation	
#	;MBdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6,dddd7,dddd8 ,dddd9; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7,dddd8,dddd9;	brake actuator의 동작관련 파라메터를 설정함.	
#	<pre>;Mbdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6,dddd7; ;Mb?;</pre>	Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;	brake actuator의 동작관련 파라메터를 설정함.	
+#	;MBiiii1,iiii2,iiii3,iiii 4,iiii5,iiii6,iiii7; ;MB?;	MBiiii1,iiii2,iiii3,iiii4,iiii5,iiii6,iiii7;	EPS의 동작관련 파라메터를 설정함.	
#	;MBdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6,dddd7; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6,dddd7;	자석가동자 제어기의 동작관련 파라메터를 설정함.	
#	;Mbdddd1, 2; ;Mb?;	Mbdddd1, IIII2;	자석가동자 제어기의 동작관련 파라메터를 설정함.	
+^	;MBdddd1,dddd2; ;MB?;	MBdddd1,dddd2;	F/T 센서 drift offset 제거 동작관련 파라메터를 설정함.	
#	;MEdddd1,dddd2,dddd3,dddd 4; ;ME?;	MEdddd1,dddd2,dddd3,dddd4;	brake actuator의 동작관련 파라메터를 설정함.	



적용 CPU	명령어	return value	operation
+#^	BLDC 모터의 경우 ;MBdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6; ;MB?;	MBdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터1의 경우, Hall 센서 A/B/C 상신호의 에지위치를 설정함.
+#^	BLDC 모터의 경우 ;Mb?;	Mbdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall 센서를 사용하지 않는 BLDC 모터1의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어기로 전 송함.
+#^	BLDC 모터의 경우 ;MEdddd1,dddd2,dddd3,dddd 4,dddd5,dddd6; ;ME?;	MEdddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	엔코더를 사용하지 않고 digital Hall IC만을 사용하는 BLDC 모터2의 경우, Hall 센서 A/B/C 상신호의 에지위치를 설정함.
+#^	BLDC 모터의 경우 ;Me?;	Medddd1,dddd2,dddd3,dddd4,dddd5,dddd6;	analog Hall 센서를 사용하지 않는 BLDC 모터2의 경우 Hall 센서신호의 에지간 간격값을 읽어서 상위제어기로 전 송함.

적용 CPU	명령어	return value	operation
+#^	low resolution encoder의 DC 모터인 경우 ;MBdddd1,dddd2,dddd3,dddd 4; ;MB?;	MBdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터1의 경우, 엔 코더신호 A/B 상신호의 에지위치를 설정함.
+#^	low resolution encoder의 DC 모터인 경우 ;Mb?;	Mbdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터1의 경우 엔코 더신호의 에지간 간격값을 읽어서 상위제어기로 전송함.
+#^	low resolution encoder의 DC 모터인 경우 ;MEdddd1,dddd2,dddd3,dddd 4; ;ME?;	MEdddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터2의 경우, 엔 코더신호 A/B 상신호의 에지위치를 설정함.
+#^	low resolution encoder의 DC 모터인 경우 ;Me?;	Medddd1,dddd2,dddd3,dddd4;	low resolution encoder를 사용하는 DC 모터2의 경우 엔코더신호의 에지간 간격값을 읽어서 상위제어기로 전송함.
+#^ +#^	low resolution encoder의 DC 모터 또는 BLDC 모터인 경우 ;MFdddd1,dddd2; ;MF?;	MFdddd1,dddd2;	low resolution encoder를 사용하는 DC 모터 또는 BLDC 모터인 경우에 속도신호의 변화를 제한하는 범위를 설정함.

적용 CPU	명령어	return value	operation
+#^	;VH;	VHiiii1,iiii2;	현재의 각변위를 홈위치로 설정함.
+#^	;\0iiii1,iiii2;	V0iiii1,iiii2;	홈위치의 각변위값을 설정함.
+#^	;VP;	VP 1, 2;	현재의 각변위값을 읽어내어 host로 전송함.
+#^	;VAiiii1,iiii2;	VAiiii1,iiii2;	각변위값을 설정하여 설정 각변위 위치로 이동시킴.
+#^	;VTAAAAAAAAAA;	VTAAAAAAAAAA;	영상안정화 PCB로 명령어를 전송하고 결과값을 수신함.
+#^	;1;	1;	
+#^	;2;	2;	
+#^	;3;	3;	
+#^	;4;	4;	일정시간동안 panning 및 tilting 방향으로 각변위를 일으
+#^	;6;	6;	킴.
+#^	;7;	7;	
+#^	;8;	8;	
+#^	;9;	9;	

적용 CPU	명령어	return value	operation
+#^%	;ERHLHL,HLHL; (25LC040/080/160/320/320A 만 해당)	ERHLHL, HLHL;	파라메터 값을 EEPROM으로부터 읽어내어 DSP SARAM에 저장함. (EEPROM으로부터 읽은 값을 LF2406A의 경우는 0x0A00 - 0x11FF, F2811의 경우는 0x3F8200 - 0x3F89FF, F28334의 경우는 0xE200 - 0xE9FF 번지에 저장함.)
+#^%	;ESHLHL,HLHL; (25LC040/080/160/320/320A 만 해당)	ESHLHL, HLHL;	DSP SRAM에 있는 파라메터 값을 page단위(8 word 또는 16 byte)로 EEPROM에 저장함. (LF2406A의 경우는 0x0A00 - 0x11FF, F2811의 경우는 0x3F8200 - 0x3F89FF, F28334의 경우는 0xE200 - 0xE9FF 번지의 값을 EEPROM에 저장함.)
+#^%	;EsA55A;	EsA55A;	DSP SRAM에 있는 작동 파라메터 값을 한꺼번에 EEPROM에 저장함.
+#^%	;EDA55A;	EDA55A;	DSP SRAM에 있는 파라메터 값을 default 값으로 바꿈. 이때 ST 명령어로 선정된 motor type에 따라서 파라메터의 기본값이 설정됨.
+#^%	;EC?;	ECHLHL,HLHL; (LF2406A의 경우) ECHLHL,HLHL,HLHL; (F2811/F28334의 경우)	RS232, RS485, CAN 통신과정에서 어떤 에러가 발생하였는 지를 읽어서 host로 전송함.
+#^%	;ECR;	ECR;	RS232, RS485, CAN 통신에러 플래그를 없애고 에러를 복구함.
+#^	;EcS;	EcS;	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전 송되어오는 broadcasting 데이터를 수신하도록 설정함.
+#^	;EcR;	EcR;	RS485 또는 CAN을 통하여 모터제어기 주소 254(0xFE)로 전 송되어오는 broadcasting 데이터를 수신하지 못하도록 설 정함.



통신모드의 설정 (2012년 3월 15일 이전)

♣ 2406a의 경우, BLDC는 버젼 1.11 부터, One PCB형 DC는 버젼 1.21부터, 초소형 DC는 1.0부터 통신모드 설정용 DIP 스위치를 가지고 있음. 2811 및 자동화용은 DIP 스위치가 없으며 전원투입 시에 EEPROM에 저장된 스위치 설정에 따라 통신모드가 설정됨. (전원투입시의 스위치 설정에 따라 통신모드가 설정되므로, 작동 도중에 스위치를 바꾸어도 통신모드가 바뀌지는 않음)

TMS320LF2406A으로서 자동화용 모델이 아닌 경우 (FAULHABER_TYPE==0 으로 컴파일된 경우)				
통신모드	1번 스위치	2번 스위치	3번 스위치	4번 스위치
RS485 only	OFF (1)	OFF (1)	Х	Х
RS232, CAN	ON (O)	0FF (1)	X	X
CAN only	OFF (1)	ON (O)	X	X
RS232 only	ON (O)	ON (O)	Х	Х
통신 테스트 모드	Х	Х	OFF (1)	ON (O)

TMS320LF2406A으로서 자동화용 모델 (FAULHABER_TYPE==1 로 컴파일된 경우)				
통신모드	1번 스위치	2번 스위치	3번 스위치	4번 스위치
RS485 only	ON (O)	OFF (1)	Χ	Χ
RS232, CAN	OFF (1)	OFF (1)	X	Χ
CAN only	ON (O)	ON (O)	X	Χ
RS232 only	OFF (1)	ON (O)	Χ	Χ
통신 테스트 모드	X	X	OFF (1)	ON (O)

(FAULHABER_TYPE==1 또는 NO_DIP_SWITCH==1 로 컴파일된 경우) - H/W

TMS320F2810/11/12 또는 TMS320F28334/5로서 자동화용 모델

TMS320F2810/11/12 또는 TMS320F28334/5로서 자동화용 모델이 아닌 경우 (FAULHABER_TYPE==0 이고 NO_DIP_SWITCH==0 으로 컴파일된 경 우) - H/W 스위치에 의하여 통신의 작동 모드가 설정됨				
통신모드	1번 스위치	2번 스위치	3번 스위치	4번 스위치
RS485	OFF (1)	X	Х	Х
CAN	ON (O)	X	Х	Х
LAN	ON (O)	X	Х	Х
통신 테스트 모드	Χ	X	OFF (1)	ON (O)
외부표시 모드 (2009-11-05일 이 후 버젼)	X	Х	ON (O)	OFF (1)

스위치 대신에 EEPROM에 저장되는 S/W 스위치 (변수명은 operation_mode_SWITCH) 를 사용					
통신모드	1번 스위치 (bit0)	2번 스위치 (bit6)	3번 스위치 (bit5)	4번 스위치 (bit4)	
RS485	OFF (1)	X	X	Χ	
CAN	ON (O)	X	X	X	
LAN	ON (O)	X	Х	Х	
통신 테스트 모드	Χ	X	OFF (1)	ON (O)	
외부표시 모드 (2009-11-05일 이 후 버젼)	х	x	ON (O)	OFF (1)	



통신모드의 설정

- ♣ DIP switch가 없는 경우는 SX0070; 로 값을 설정하고 ESOA00,0100; 명령어로 EEPROM에 저장한 다음, 전원을 재 투입하고 나면 CAN 통신 모드로 작동하며, SX0071; 로 값을 설정하고 ESOA00,0100; 명령어로 EEPROM에 저장한 다음, 전원을 재 투입하고 나면 RS485 통신 모드로 작동함. (전원을 재투입하지 않으면 반영되지 않음)
- ♣ 통신 테스트 모드에서는 수신된 데이터를 처리하지 않고, 첫 바이트가 수신된 시간으로부터 2ms후에 그 때까지 받은 데이터를 모두 되돌려 줌. 만약 통신에러가 있었다면 통신 에러코드를 해당 바이트의 앞에 붙여서 되돌려줌. 만약 break detect 에러였다면 통신모듈을 초기화함.
- ♣ 통신 테스트 모드에서는 전원투입시에 기본 baud rate를 9600(RS232C)/1250000(RS485) 로 설정함.
- ♣ RS232C 모드의 통신속도와 RS485 모드의 통신속도는 각각 SB 나 Sb 명령어로 바꿀 수 있으며 ES 명령어로 EEPROM에 저장할수있음.
- ♣ 명령어의 리턴을 받지 않으려면. 명령어의 마지막에 ;대신 !를 보내거나 또는 RS485에서는 주소에 128을 더한 주소를 사용함.
- ♣ DIP 스위치의 3번스위치는 2406a 스테핑모터 드라이버의 경우 program 다운로드 설정 스위치로도 쓰임.
- ♣ TMS320F2810/11/12 버젼에서 USB나 LAN을 사용하려면 모터제어기에서 DIP 스위치나 SX 명령어를 사용하여 CAN 모드로 설정되어야 하며, 이 경우에 SCIB 는 FT232R이나 XPORT와 통신을 사용하는데 사용되는데, 이때 RS485 통신은 불가하며 CAN 통신은 사용이 가능하나 USB나 LAN을 동시에 사용하지만 않으면 됨. 그러나 TMS320F28334/5 버젼에서는 SCIC를 사용하며 RS485/CAN 통신과 별개로 사용할수 있음.
- ♣ USB의 경우 PC 쪽에서 FTDI device driver를 install 하거나 자동 찾기하면 install이 되며, com port로 잡히면 115200 baud rate로 통신속 도를 설정하고 사용하면 됨.
- ♣ LAN의 경우 Lantronix사의 DevideInstaller를 사용하여
 Web Confuration>Network에서 IP address, SubnetMask, Default Gateway를 설정하고
 Serial Setting>Port Setting에서 Baud Rate를 설정하고 Enable Packing을 체크하고
 OK 한다음 Apply Setting을 하여야하며
 - PC 쪽에서는 IP address와 포트 10001을 지정하여 연결하면 통신이 개시됨.



Ethernet-IP Xport 설정

PC 쪽 설정

- 1. Di32DL_4.2.0.1_Web.exe를 실행하여 DeviceInstaller 를 설치함.
- 2. dotnetfx.exe 를 실행하는데, 이미 있으면 설치를 중지함.
- 3. 1:1 케이블을 모터제어기와 허브 사이에 연결함.
- 4. DeviceInstaller를 실행하고 Search 버튼을 눌러서 Xport를 탐색함. (최초의 IP 설정시에만 해당됨)
- 5. Xport의 탐색이 이루어지면 DeviceInstaller 화면에 Xport 탐색결과가 표시되며 이를 더블 클릭함.
- 6. 이때 DeviceInstaller 오른편 화면에 Device Detail, Web Configuration, Telnet Configuration 탭이 생김.
- 7. Web Configuration 탭을 클릭하고 오른편의 녹색 화살표를 더블 클릭함.
- 8. 암호화면이 나오면 확인 버튼을 누름.
- 9. 만약 고정 IP가 없으면 바뀐화면의 Network을 클릭하여 현재의 IP를 확인함.
- 10. 만약 고정 IP가 있으면 Network을 클릭하여 IP address, Subnet Mask, Default Gateway를 입력하고, OK 버튼을 누른다음, Apply Settings를 클릭하고 저장이 될때까지 기다림.
- 11. DeviceInstaller를 닫고 다시 Web Configuration 탭 이후 녹색 화살표를 더블 클릭한 상태로 들어가서 Network 탭을 눌러서 바뀐 IP address를 확인함.
- 12. Serial Setting을 눌러서 RS232, 115200, Enable Packing을 설정하고, OK 버튼을 누른다음, Apply Settings를 클릭하고 저장이 될 때까지 기다림.
- 13. PC의 CubeMon에서 TCP/IP를 눌러서 Client Module 창이 생기는데, 여기에 9또는 10에서 설정된 IP address를 입력하고, 포트번호는 10001을 입력한 다음 연결 버튼을 누름.
- 14. 연결이 되면 연결버튼 옆의 LED가 깜빡임.
- 15. 전송데이터 영역에 SA?; + <enter> 를 누르면 서버응답창에 SAFE01;이 표시되면 정상임.

제어기쪽 설정

LAN 버젼의 프로그램을 다운로드한 다음, SAFE01;ST0303;SB115200;SX0070;EsA55A; 를 실행하고 전원을 껏다켬.



통신 TEST

♣ TMS320LF2406A 모터제어기에서 통신의 수신/회신 상황을 알려면

QR036C,04; 를 수행하면 RS485_회신_횟수/CAN_브로드캐스트명령_수신프레임_개수/CAN_회신프레임_개수/CAN_명령_수신프레임_개수/RS485_명령수신_횟수, CAN_회신프레임_개수=CAN_브로드캐스트명령_수신프레임_개수 또는 CAN_명령_수신프레임_개수이어야하며 각각의 값은 상위제어기에서 보낸 수와 같을 때 통신상의 오류가 없는 것임. QR036C,04;에서 036C 값과 리턴값은 소프트웨어 버젼에 따라서 바뀔수 있는 값이기 때문에 map file로부터 SCI_TX_cnt, CAN_RX3_cnt, CAN_TX5_cnt, CAN_RX1_cnt, SCI_RX_cnt 변수의 주소를 확인 하여야함.

💲 TMS320F2810/2811/2812 모터제어기에서 통신의 수신/회신 상황을 알려면

QR8274,04; 를 수행하면 RS485_회신_횟수/RS485_명령수신_횟수/CAN_브로드캐스트명령_수신프레임_개수/CAN_회신프레임_개수/CAN_명령 수신프레임_개수 를 알려주는데 RS485_회신_횟수=RS485_명령수신_횟수, CAN_회신프레임_개수=CAN_브로드캐스트명령_수신프레임_개수 또는 CAN_명령_수신프레임_개수이어야하며 각각의 값은 상위제어기에서 보낸 수와 같을 때 통신상의 오류가 없는 것임. QR8274,04;에서 8274 값과 리턴값은 소프트웨어 버젼에 따라서 바뀔수 있는 값이기 때문에 map file로부터 SCI_TX_cnt, SCI_RX_cnt, CAN_RX1_cnt, CAN_TX5_cnt, CAN_RX3_cnt 변수의 주소를 확인 하여야함.

⇔ TMS320F2811 보드(이족보행로봇용 SW 탑재)를 사용하여 CAN 통신테스트를 하려면

SM0001;QT8388,00,0001; 을 수행하면 10ms 주기로 01번 제어기로 SA?; 의 명령어를 1Mbps의 CAN 통신을 사용하여 연속으로 보냄. 송신/수신 상태를 보려면 QR8384,03; 를 수행하여 CAN_명령_송신횟수/CAN_수신횟수/RS485_명령_송신횟수/RS485_수신횟수 를 읽어내면 되는데, CAN_수신횟수=CAN_명령_송신횟수 이어야함. 연속송신을 정지시키려면 QT8388,00,0000; 을 수행하면 됨. QT8388,00,0001; 또는 QR8384,03;에서 8388/8384 값과 리턴값은 소프트웨어 버젼에 따라서 바뀔수 있는 값이기 때문에 map file로부터 CAN_TX_cnt, CAN_RX_cnt, SCI_TX_cnt, COM_check_flg 변수의 주소를 확인 하여야함.

♣ TMS320F2811 보드(이족보행로봇용 SW 탑재)를 사용하여 RS485 통신테스트를 하려면

QT8388,00,0002; 을 수행하면 10ms 주기로 01번 제어기로 SA?; 의 명령어를 1.25Mbps의 RS485 통신을 사용하여 연속으로 보냄. 송신/수신 상태를 보려면 QR8384,03; 를 수행하여 CAN_명령_송신횟수/CAN_수신횟수/RS485_명령_송신횟수/RS485_수신횟수 를 읽어내면 되는데, RS485_수신횟수=RS485_명령_송신횟수 이어야함. 연속송신을 정지시키려면 QT8388,00,0000; 을 수행하면 됨. QT8388,00,0002; 또는 QR8384,03;에서 8388/8384 값과 리턴값은 소프트웨어 버젼에 따라서 바뀔수 있는 값이기 때문에 map file로부터 CAN_TX_cnt, CAN_RX_cnt, SCI_TX_cnt, COM_check_flg 변수의 주소를 확인 하여야함.



모터제어기 및 호스트 주소의 설정

- ♣ 모터제어기의 주소를 바꾸려면
 - 1) SAHLHL; 명령을 실행하여 모터에 주소를 설정함. (하위바이트가 주소이며 상위바이트는 반드시 하위바이트의 inverse여야함)
 - 2) ESOAOO,0100; 명령어 또는 EsA55A; 로 EEEPROM에 저장함.

♣ 주소값으로서 254(0xFE) 및 255(0xFF)는 bradcasting용으로 사용하는 주소이므로 126(0x8E), 127(0x8F), 254(0xFE) 및 255(0xFF)는 제어 기의 주소로 사용하면 안됨.

- ♣ 호스트의 주소를 바꾸려면
 - 1) SmHLHL; 망령어로 호스트의 주소를 바꿈. (하위바이트가 호스트주소이며 상위바이트는 반드시 하위바이트의 inverse여야함) 이때 SmHLHL;에 대한 리턴값인 SmHLHL;은 바뀐 호스트 주소로 리턴함.
 - 2) RS485 모드일 때는 설정된 호스트의 주소를 첫 데이터로 하여 리턴값을 호스트로 전송함.
 CAN 모드일 때는 CAN transmit data format에서 ID 영역에 설정된 호스트의 주소를 탑재함.
 RS485 모드일 때 설정된 호스트 주소는 CAN 모드로 바뀌었을 때 적용되지 않으나, CAN 모드일 때 설정된 호스트 주소는 RS485 모드로 바뀌었을때 때 적용됨.
 - 3) ESOAOO,0100; 명령어 또는 EsA55A; 로 EEPROM에 저장함.
 - ♣ 주소값으로서 254(0xFE) 및 255(0xFF)는 bradcasting용으로 사용하는 주소이므로 126(0x8E), 127(0x8F), 254(0xFE) 및 255(0xFF)는 host 의 주소로 사용하면 안됨.
 - ※ 1.25Mbps의 RS485통신이 가능한 PCI카드를 사용하는 경우는 host주소로 0번을 사용하면 정상정인 통신이 안됨. (1.25Mbps의 설정은 제어판>serial_port의 속성에서 실시하며, 40MHz 오실레이터의 경우 16 sample, clock 체배 1.00으로 하면 1.25Mbps가 설정되며, 송신출력의 자동 off 기능을 사용하려면 DTR을 active high로 설정하여여하고, 프로그램에서는 DTR을 enable 하여야 함.)



RS485 통신을 이용한 다중 마이컴 통신

♣ TI사의 DSP제어기는 비동기통신포트(SCI)를 이용한 다중마이컴 통신을 위하여 다음과 같은 2가지 방식을 제공하고 있다.

	Idle-Line Multiprocessor 모드	Address-Bit Multiprocessor 모드
1-byte format	Start LSB 2 3 4 5 6 7 MSB Parity Stop Identifier mode (Normal normalizations) RS232 통신과 같이 start-bit/stop-bit/parity-bit가 붙은 8-bit 형식	Address Let Start LSD 2 3 4 5 6 7 MSD Address Party Stop Address
1-byte 값이 주소인지 데 이터인지를 구별하는 방법	이전 바이트와 현재 수신된 바이트 사이의 시간간격 이 10-bit 시간 이내이면 데이터로 판단하고, 10-bit 시간 이상이면 주소로 판단함.	현재 수신된 바이트의 9th bit가 1 이냐 0이냐에 따라서 주소인지, 데이터인지를 판단.
PC등 기존 기기들과의 호 환성	바이트와 바이트사이의 시간간격을 지킬수 있다면 호환성 있음.	호환성 없음.
로보큐브테크 채택방식	0	X

- ♣ 로보큐브테크는 기존 기기들과의 호환성 유지를 위하여 Idle-Line Multiprocessor 모드를 채택하고 있음.
- ♣ 로보큐브테크 제어기 사용자의 RS485 통신시의 준수사항.
 - : 주소가 같은 데이터 블록안에 있는 데이터를 전송할 때는 바이트와 바이트 사이의 시간 간격이 10-bit 전송시간 이내가 되도록 하여야함.
 - : 이를 준수하려면 주소가 같은 데이터 블록 안에 있는 데이터는 한꺼번에 준비하여야하며, 전송을 개시하면 나머지 데이터는 interrupt routine을 이용하여 background에서 전송하도록 하는 것이 필요함. (프로그램이 일일이 관여하여 보내는 형태면 통신이 제대로 안될수 있음)
 - : 만약 바이트와 바이트 사이의 시간 간격이 클수 밖에 없는 경우라면 통신속도를 떨어뜨려 사용하는 것도 한가지 방법임. (예를 들면 9600 bps, 이 경우 10-bit의 시간은 1.04ms 임)
- ♣ 통신에 binary 값을 사용하지 않는 경우의 해결방법.
 - : 통신에 binary값을 사용하지 않는 경우(Binary 값을 사용하는 Pa/Pp/p 등의 명령어를 사용하지 않는 경우)에 한하여 수신데이터의 마지막을 지칭하는 ';' 나 '!' 를 사용하여, 바이트와 바이트 사이의 시간 간격에 관계없이 다중마이컴 통신이 가능하게 할수 있음.



p/Pp/Pa 명령를 사용하여 위치/속도/전류값을 값을 갱신할 때, 갱신주기의 기본단위를 변경하려면

- ♣ p/Pp/Pq 명령의 갱신주기의 단위를 변경하려면 (TMS320F2810/11/12을 사용한 RJM_VER8 DC 모터제어기에서만 가능함)
 - 1) St?; 를 사용하여 현재의 설정값을 확인함. 만약 첫번째와 두번째의 리턴값이 2002~3000 사이의 값이라면 갱신주기를 1.25ms 단위로 설정하는 상태이며, 2~1000 사이의 값이라면 갱신주기를 1ms 단위로 설정하는 상태임.
 - 2) 갱신주기를 1.25ms 단위로 설정하려면 Stdddd1,dddd2,dddd3; 명령을 실행하되 dddd1/dddd2의 값은 2002~3000 사이의 값으로 설정하고, 이 경우에 2002~3000의 값은 2.5ms부터 1.25ms 단위로 1250ms 까지의 주기를 설정하는 것이 됨.
 - 갱신주기를 1ms 단위로 설정하려면 Stdddd1,dddd2,dddd3; 명령을 실행하되 dddd1/dddd2의 값은 2∼1000 사이의 값으로 설정하고, 이 경우에 2∼1000의 값은 2ms부터 1ms 단위로 1000ms 까지의 주기를 설정하는 것이 됨.
 - 3) EsA55A; 을 실행하여 EEPROM에 파라메터를 저장함.
 - 4) 전원을 껏다가 다시 켜면 p/Pp/Pq 명령의 갱신주기단위와 갱신주기가 직전에 설정된 대로 작동함. 만약 1.25ms 단위로 갱신주기를 바꾸었다면 LED의 깜빡이는 속도가 25% 정도 느려짐.



♣ 위치제어를 하려면

1) SMHLHL; 명령어로 01/03의 위치제어 모드를 선택함.

상위/하위바이트가 01이면 모터2/모터1이 위치제어모드(inner loop 없음, PWM사용, RJM_VER1/2/4(TMS320LF2406) DC 모터전용) 상위/하위바이트가 02이면 모터2/모터1이 위치제어모드(inner loop 전류제어, RJM_VER1/2/4(TMS320LF2406)는 사용불가) 상위/하위바이트가 03이면 모터2/모터1이 위치제어모드(inner loop 속도제어(BLDC), inner loop 없고 PWM사용(DC))

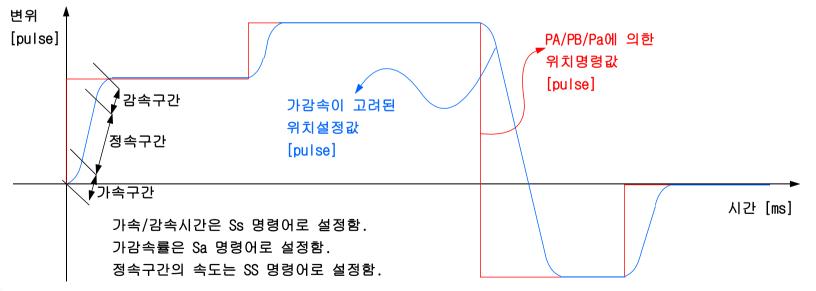
- 2) SSdddd1.dddd2; 명령어로 정속구간의 속도를 설정함. (단위는 rpm 임, PM/PP/PQ/Pp/Pq/p 명령어 사용 때는 유효하지 않음)
- 3) Ssdddd1,dddd2; 명령어로 가속/감속기간를 설정함. (단위는 ms이며, SS 명령어로 정속구간의 속도를 설정한 다음에 Ss 명령어에 의한 설정을 실행해야함. PM/PP/PQ/Pp/Pq/p 명령어 사용 때는 유효하지 않음)

또는 Sadddd1,dddd2; 명령어로 가감속도를 설정함. (단위는 rpm/100ms 임, PM/PP/PQ/Pp/Pq/p 명령어 사용 때는 유효하지 않음)

- 4) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 5) PA/PB/Pa/PM/PP/PQ/Pp/Pq/p 명령어를 사용하여 위치명령값을 전송함. (단위는 4체배된 엔코더펄스 수임)
- 6) 위치제어모드를 종료하려면 SMHLHL; 명령어로 00의 제어 모드를 선택하고,

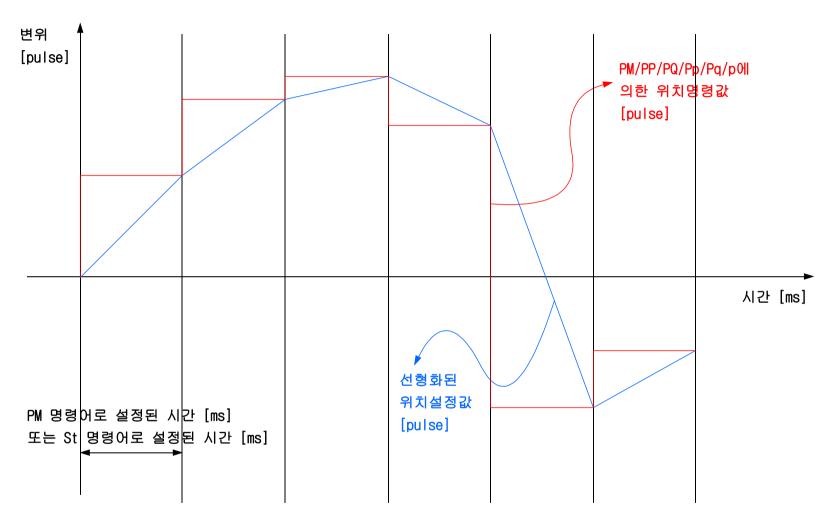
PDHLHL; 명령어로 해당 모터제어보드의 출력을 disable 시킴. (HLHL은 보드번호를 써야함)

♣ 만약 위치제어 이전에 속도제어를 하고 있었다면 QEA55A; QEA5A5; QE5A5A; 명령등으로 엔코더를 초기화하는 것이 필요할 수도 있음.





420



- ♣ St 명령어로 설정된 시간은 최소가 2ms 이며 PP/PQ/Pp/Pq/p 명령이 사용될 때 적용됨.
- ♣ 선형화된 위치설정값은 1ms 단위로 갱신됨.
- ♣ PM/PP/PQ/Pp/Pq/p 명령에 대해서는 별도의 가감속이나 속도제한이 적용되지 않기 때문에 사용자가 이를 고려하여 명령값을 내려보내야 함.



♣ DC 모터를 RJM_VER4의 01/03번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

$$PWM_set = \frac{100}{KP^{-1}}e_k + \frac{1}{KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{100}{KD^{-1}}(e_k - e_{k-4}) \quad \text{PWM_set는 } -600 ~ +600 ~ \text{범위의 값으로 제한되며, duty ratio로는 } -100\% ~ +100\% \text{에 해당함.} \qquad \text{KP}^{-1}/\text{KI}^{-1}/\text{KD}^{-1} 는 ~ SQ/SR ~ 명령어로 설정하고 저장할 수 있음.}$$

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 200%의 duty를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 50ms동안 모터 한바퀴일 때, 100%의 duty를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{\max}/60)/PWM$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 10%의 duty를 적용하는 것에 해당함.

PWM=600, PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도

♣ DC 모터를 RJM_VER4의 02번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

$$current_cmd = \frac{800}{KP^{-1}}e_k + \frac{1}{2*KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{12800}{KD^{-1}}(e_k - e_{k-4})$$
 current_cmd는 -current_control_limit ~ +current_control_limit 범위의 값으로 제한되며, 기본값은 $-1024 \sim +1024$ 인데, $10m\Omega$ 의 전류감지저항을 쓰는 경우 $-2.5A \sim +2.5A$ 에 해당함. current_control_limit의 크기는 Sw 명령어로 설정하고 저장할 수 있음. $KP^{-1}/KI^{-1}/KD^{-1}$ 는 SQ/SR 명령어로 설정하고 저장할 수 있음.

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*50)/PWM$ 이 기보가의 사용하며 의한 오랜지 모든 하비의의

 $^{-1}=(PPR*50)/PWM$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 23.4A의 전류를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, 2.9A의 전류를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{\max}/60)/PWM$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 18.75A의 전류를 적용하는 것에 해당함.

PWM=600, PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도



♣ DC 모터를 RJM_VER8의 01/03번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

 $PWM_set = \frac{100}{KP^{-1}}e_k + \frac{1}{KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{100}{KD^{-1}}(e_k - e_{k-4}) \quad \text{PWM_set는 } -600 ~ +600 ~ \text{범위의 값으로 제한되며, duty ratio로는 } -100\% ~ +100\% \text{에 해당함.} \qquad \text{KP}^{-1}/\text{KI}^{-1}/\text{KD}^{-1} 는 ~ SQ/SR ~ 명령어로 설정하고 저장할 수 있음.}$

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 200%의 duty를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 50ms동안 모터 한바퀴일 때, 100%의 duty를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{max}/60)/PWM$ 이 기본값을 사용하면 최대 $RPM(V_{max})$ 으로 회전할 때, 10%의 duty를 적용하는 것에 해당함.

PWM=600, PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도

♣ DC 모터를 RJM_VER8의 02번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

 $current_cmd = \frac{400}{KP^{-1}}e_k + \frac{1}{2*KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{3200}{KD^{-1}}(e_k - e_{k-4})$ $current_cmd$ 는 $-current_control_limit ~ +current_control_limit 범위의 값으로 제한되며, 기본값은 <math>-1229 \sim +1229$ 인데, $2m\Omega$ 의 전류감지저항을 쓰는 경우 $-15A \sim +15A$ 에 해당함. $current_control_limit$ 의 크기는 Sw 명령어로 설정하고 저장할 수 있음. $KP^{-1}/KI^{-1}/KD^{-1}$ 는 SQ/SR 명령어로 설

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*50)/PWM$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 59A의 전류를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*50) / PWM$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, 15A의 전류를 적용하는 것에 해당함.

정하고 저장할 수 있음.

 $KD^{-1} = (PPR*4)*(V_{max}/60)/PWM$ 이 기본값을 사용하면 최대 $RPM(V_{max})$ 으로 회전할 때, 23A의 전류를 적용하는 것에 해당함.

PWM=600, PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도



♣ BLDC 모터를 RJM_VER3의 03번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

$$speed_cmd = \frac{100}{KP^{-1}}e_k + \frac{1}{KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{100}{KD^{-1}}(e_k - e_{k-4})$$
 $speed_cmd \succeq -V_{max} \sim +V_{max}$ 범위의 값으로 제한되며, 단위는 RPM임. $KP^{-1}/KI^{-1}/KD^{-1} \succeq SQ/SR$ 명령어로 설정하고 저장할 수 있음.

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*200)/V_{\rm max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 50%의 $V_{\rm max}$ 를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*200)/V_{\max}$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, V_{\max} 를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4*2)*(V_{\max}/60)/V_{\max}$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 5%의 V_{\max} 를 적용하는 것에 해당함. PPR=모터 1회전시의 4체배된 엔코터 펄스수. V_{\max} =무부하침대회전속도

♣ BLDC 모터를 RJM_VER3의 02번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms)

$$current_cmd = \left\{\frac{800}{\mathit{KP}^{-1}}e_{\scriptscriptstyle k} + \frac{1}{\mathit{KI}^{-1}}\sum_{i=0}^{i=k}e_{\scriptscriptstyle i} + \frac{12800}{\mathit{KD}^{-1}}(e_{\scriptscriptstyle k} - e_{\scriptscriptstyle k-4})\right\}\frac{I_{con_max}}{V_{\max}} \quad \text{current_cmd는 -current_control_limit ~ +current_control_limit ~ +cu$$

current_cmd는 -current_control_limit ~ +current_control_limit 범위의 값으로 제한되며, 기본값은 -4095*2 ~ +4095*2 인데, IRMCK201의 정격전류 설정치의 2배에 해당함. current_control_limit의 크기는 Sw 명령어로 설정하고 저장할 수 있음. KP⁻¹/Kl⁻¹/KD⁻¹는 SQ/SR 명령어로 설정하고 저장할 수 있음.

 $e_{\nu} = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR * 200) / V_{max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 400%의 정격전류를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*200)/V_{max}$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, 100%의 정격전류를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4*2)*(V_{\max}/60)/V_{\max}$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 640%의 정격전류를 적용하는 것에 해당함.

PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도



♣ 엔코더가 부착된 BLDC 모터를 RJM_VER7의 03번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms이며, MAXON EC-90이 아닌 경우)

$$speed_cmd = \frac{100}{KP^{-1}}e_k + \frac{1}{KI^{-1}}\sum_{i=0}^{i=k}e_i + \frac{4}{KD^{-1}}(e_k - e_{k-4})$$
 speed_cmd는 $-V_{\max} \sim +V_{\max}$ 범위의 값으로 제한되며, 단위는 RPM임. KP $^{-1}/KI^{-1}/KD^{-1}$ 는 SQ/SR 명령어로 설정하고 저장할 수 있음.

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*1000)/V_{
m max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 10%의 $V_{
m max}$ 를 적용하는 것에 해당함.

 $K\!I^{-1} = (PPR*2000)/V_{
m max}$ 이 기본값을 사용하면 위치 오차가 2000ms동안 모터 한바퀴일 때, $V_{
m max}$ 를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{\max}/60)/(V_{\max}*25)$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 10%의 V_{\max} 를 적용하는 것에 해당함. PPR=모터 1회전시의 4체배된 엔코더 펄스수. V_{\max} =무부하최대회전속도

🗘 엔코더가 부착된 BLDC 모터를 RJM_VER7의 02번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms, MAXON EC-90이 아닌 경우)

$$\textit{current_cmd} = \left\{ \frac{100}{\textit{KP}^{-1}} e_k + \frac{1}{\textit{KI}^{-1}} \sum_{i=0}^{i=k} e_i + \frac{4}{\textit{KD}^{-1}} (e_k - e_{k-4}) \right\} \frac{I_{\textit{con_max}}}{V_{\text{max}}} \qquad \text{current_cmd는 -current_control_limit ~ +current_control_limit ~ +cu$$

current_cmd는 -current_control_limit ~ +current_control_limit 범위의 값으로 제한되며, 기본값은 -2457 ~ +2457 인데, 5mΩ의 전류감지저항을 쓰는 경우 -12A ~ +12A에 해당함. current_control_limit의 크기는 Sw 명령어로 설정하고 저장할 수 있음. KP⁻¹/KI⁻¹/KD⁻¹는 SQ/SR 명령어로 설정하고 저장할 수 있음.

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*1000)/V_{max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 1.2A의 전류를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*2000)/V_{max}$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, 1.2A의 전류를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{\max}/60)/(V_{\max}*25)$ 이 기본값을 사용하면 최대 $RPM(V_{\max})$ 으로 회전할 때, 1.2A의 전류를 적용하는 것에 해당함.

PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도



♣ 엔코더가 부착된 BLDC 모터를 RJM_VER7의 03번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms이며, MAXON EC-90인 경우)

$$speed_cmd = \frac{100}{\mathit{KP}^{-1}}e_{\mathit{k}} + \frac{1}{\mathit{KI}^{-1}}\sum_{i=0}^{\mathit{i=k}}e_{\mathit{i}} + \frac{100}{\mathit{KD}^{-1}}(e_{\mathit{k}} - e_{\mathit{k-4}}) \\ & \text{speed_cmd} \vdash -V_{\max} \sim +V_{\max} \text{ 범위의 값으로 제한되며, 단위는 RPM임.} \\ & \text{KP}^{-1}/\mathsf{KI}^{-1}/\mathsf{KD}^{-1} \vdash \mathsf{SQ/SR} \text{ 명령어로 설정하고 저장할 수 있음.} \\ \end{cases}$$

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*1000)/V_{
m max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 10%의 $V_{
m max}$ 를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*2000)/V_{max}$ 이 기본값을 사용하면 위치 오차가 2000ms동안 모터 한바퀴일 때, V_{max} 를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{\max} / 60) / (V_{\max}*25)$ 이 기본값을 사용하면 최대 RPM(V_{\max})으로 회전할 때, 250%의 V_{\max} 를 적용하는 것에 해당함.

PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도

🗘 엔코더가 부착된 BLDC 모터를 RJM_VER7의 02번 모드로 제어할 때, PID 제어 관계식 (제어주기는 1ms, MAXON EC-90인 경우)

$$\textit{current_cmd} = \left\{ \frac{100}{\textit{KP}^{-1}} e_k + \frac{1}{\textit{KI}^{-1}} \sum_{i=0}^{i=k} e_i + \frac{4}{\textit{KD}^{-1}} (e_k - e_{k-4}) \right\} \frac{I_{\textit{con_max}}}{V_{\textit{max}}} \qquad \text{current_cmd는 -current_control_limit ~ +current_control_limit ~ +cu$$

current_cmd는 -current_control_limit ~ +current_control_limit 범위의 값으로 제한되며, 기본값은 -2457 ~ +2457 인데, 5mΩ의 전류감지저항을 쓰는 경우 -12A ~ +12A에 해당함. current_control_limit의 크기는 Sw 명령어로 설정하고 저장할 수 있음. KP⁻¹/KI⁻¹/KD⁻¹는 SQ/SR 명령어로 설정하고 저장할 수 있음.

 $e_k = position_set - position$ 기본단위는 엔코더의 펄스를 4체배한 값임

 $KP^{-1} = (PPR*1000)/V_{max}$ 이 기본값을 사용하면 위치 오차가 모터 한바퀴일 때, 1.2A의 전류를 적용하는 것에 해당함.

 $KI^{-1} = (PPR*2000)/V_{max}$ 이 기본값을 사용하면 위치 오차가 200ms동안 모터 한바퀴일 때, 1.2A의 전류를 적용하는 것에 해당함.

 $KD^{-1} = (PPR*4)*(V_{max}/60)/(V_{max}*25)$ 이 기본값을 사용하면 최대 $RPM(V_{max})$ 으로 회전할 때, 1.2A의 전류를 적용하는 것에 해당함.

PPR=모터 1회전시의 4체배된 엔코더 펄스수, $V_{
m max}$ =무부하최대회전속도



- ♣ 위치제어의 PID제어 게인의 설정은

 - SQ 또는 SRHLHLHLHL, HLHLHLHL, HLHL, H
 - 의 4/5/6번째 파라메터에 의하여 설정 가능함.
 - Sq 또는 SpHLHLHLHL, HLHLHLHL, HLHL, H
 - 의 4/5/6번째 파라메터에 의하여도 설정 가능함. 이 경우는 50000/(SQ 또는 SR 명령에 의한 값) 에 해당하며, 값이 클수록 제어게인이 큰 형식으로 파라메터 설정이 가능함.
- ♣ RJM_VER4(2811)/RJM_VER8 DC 제어모드 02번에서는 전류제어를 사용하여 위치제어를 하게 되는데, Sw 명령에 의하여 전류명령값의 최대치가 제한된다. 이때 최대치 설정값이 지나치게 작으면 제어오차가 커지고, 지나치게 크면 기동시에 전류를 과다하게 사용함. 전류제어의 PI제어 게인은
 - SQ 또는 SRHLHLHLHL, HLHLHLHL, HLHL, HLH, HLHL, HL

 - 의 마지막 2개의 파라메터에 의하여 설정됨.

 - 의 마지막 2개의 파라메터에 의하여도 설정 가능함. 이 경우는 50000/(SQ 또는 SR 명령에 의한 값) 에 해당하며, 값이 클수록 제어게인이 큰 형식으로 파라메터 설정이 가능함.
- ♣ RJM_VER3로서 IRMCK201 을 사용하는 경우와 RJM_VER7의 경우, 02모드에서는 Sw 명령에 의하여 전류명령값의 최대치가 제한되는데, 이 값이 지나치게 작으면 제어오차가 커지고, 지나치게 크면 기동시에 전류를 과다하게 사용함. 전류제어의 PI제어 게인 MOTOR1_CURRENT_P_GAIN, MOTOR1_CURRENT_I_GAIN은 WQ 또는 WRdddd1,dddd2,dddd3,dddd4,dddd5;의 마지막 두 파라메터에 의하여 설정되며, EsA55A; 명령어에 의하여 BLDCx_PrmBuffer[13]와 BLDCx_PrmBuffer[14]에 저장될 수 있음. 또는



- ♣ RJM_VER7의 경우 Digital Hall IC만을 쓰는 경우는 02번 위치제어가 잘 안됨.
- 4 2811의 경우에 통신모드 설정용 DIP 스위치의 두번째 스위치가 전원 투입시에 ON(0)되어 있으면, SL 명령어로 설정된 각변위 이내로 각변위 명령이 제한되며, 이 경우는 작동도중에 통신모드 설정용 DIP 스위치를 바꾸어도 각변위 명령의 제한기능이 바뀌지 않음. 통신모드 설정용 DIP 스위치가 실제 존재하지 않고 EEPROM의 기억값으로만 존재하여도 각변위 명령의 제한기능은 작동되며, 이 경우에 SX 명령어로 operation_mode_SWITCH값의 0x0040에 해당하는 비트를 0으로 설정하면 각변위 제한기능이 설정되면서 제한기능이 작동개시되고, ESOAOO,0100; 명령어로 저장하게 되면 다시 설정할 필요가 없으나, EDA55A; 에 의한 기본값은 각변위 제한하는 것을 설정하는 것임.
- ♣ DIP switch가 없는 경우의 기본값은 기능을 사용하는 것임.
- ♪ 홈동작을 할 때는 위치 limit 기능 설정에 관계없이 홈찾기용 위치 limit 값을 사용함. 여기서 홈찾기용 위치 limit 값은 SI (에스엘) 명령에 의하여 설정할수있으며, 전원투입시에 이값은 SL 명령에 의한 위치 limit 값으로 초기화됨.
- ♣ SP 명령어에 의한 위치점 입력을 할 때는 위치 limit 기능 설정에 관계없이 위치 limit 값을 사용함.
- ♣ 위치 limit 기능이 설정되어 있고, 속도모드에서 위치 limit을 넘어간 상태에서 위치모드로 전환하였을 때, 위치 limit 기능이 곧바로 적용되지는 않으나, 새로운 위치값 설정명령이 입력될 때부터 위치 limit 기능이 적용됨.
- ♣ IRMCK201 BLDC나 RJM_VER7 BLDC의 경우 03번 모드 위치제어에서는 inner 속도제어 loop를 사용하므로 speedx_acc_rate, speedx_dec_rate의 영향을 받으며, 감가속값이 작으면 위치제어가 제대로 되지 않음. 따라서 모드의 설정시에 speedx_acc_rate, speedx_dec_rate를 큰 값으로 강제 설정함. 다시 속도모드로 전환할 때, 감가속도값이 지나치게 클수있으므로 가감속도값을 재설정 해주어야함.



♣ 속도제어를 하려면

1) SMHLHL; 명령어로 04/05/06의 속도제어 모드를 선택함.

상위/하위바이트가 04이면 모터2/모터1이 속도제어모드(위치제어형식으로 속도제어, 내부에서 실행될때는 DC motor의 경우 01번모드, IRMCK201 BLDC motor의 경우 02번모드, BLDC sensor type의 경우는 03번모드로 실행됨)

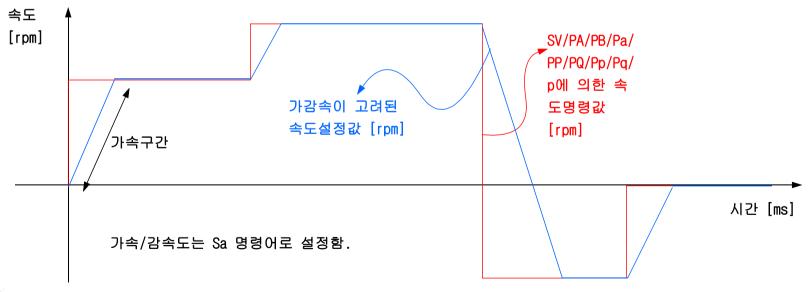
상위/하위바이트가 05이면 모터2/모터1이 속도제어모드(inner loop 없고 PWM사용(DC 2406), inner loop 전류제어(BLDC 2406, DC/BLDC 2811))

상위/하위바이트가 06이면 모터2/모터1이 속도제어모드(inner loop 없고 PWM사용(DC 2811), inner loop 속도제어(BLDC))

- 2) Sadddd1,dddd2,dddd3,dddd4; 가감속도를 설정함. (단위는 RJM_VER7인 경우에 rpm/ms이고 나머지는 rpm/100ms 임)
- 3) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 4) SViiii1,iiii2; 명령어를 사용하여 속도명령값을 전송함. (단위는 rpm 임)
- 5) 속도제어모드를 종료하려면 SMHLHL; 명령어로 00의 제어 모드를 선택하고,

PDHLHL; 명령어로 해당 모터제어보드의 출력을 disable 시킴. (HLHL은 보드번호를 써야함)

♣ 속도모드로 설정된 모터는 PA,PB,Pa,PP,PQ,Pp,Pq,p 명령을 수행할 때 속도값이 설정됨.





- ♣ DC/BLDC 05번 속도제어와 DC 모터의 06번 속도제어인 경우, PID제어 게인의 설정은 (미분게인은 F2811인 경우만 해당됨)

 - 의 7/8/9(F2811인 경우)번째 파라메터에 의하여 설정 가능함.
 - Sq 또는 SpHLHLHLHL, HLHLHLHL, HLHL, H
 - 의 7/8/9번째 파라메터에 의하여도 설정 가능함. 이 경우는 50000/(SQ 또는 SR 명령에 의한 값) 에 해당하며, 값이 클수록 제어게인이 큰 형식으로 파라메터 설정이 가능함.
- ♣ BLDC 모터 06번 속도제어의 PID제어 게인: MOTOR1_SPEED_P_GAIN, MOTOR1_SPEED_I_GAIN, MOTOR1_SPEED_D_GAIN은 WQ 또는 WRdddd1,dddd2,dddd3,dddd4,dddd5;의 앞쪽 두 파라메터 (PI 게인: IRMCK201을 사용한 BLDC제어의 경우) WQ 또는 WRdddd1,dddd2,dddd3,dddd4,dddd5;의 앞쪽 세 파라메터 (PID 게인: RJM_VER7 BLDC제어의 경우)에 의하여 설정되며, EsA55A; 명령어에 의하여 BLDCx_PrmBuffer[25],BLDCx_PrmBuffer[26],BLDCx_PrmBuffer[27](RJM_VER7 BLDC제어의 경우만)에 의하여 저장될 수 있음.
- ♣ RJM_VER4(2811)/RJM_VER8 DC 제어모드 05번에서는 전류제어를 사용하여 속도제어를 하게 되는데, Sw 명령에 의하여 전류명령값의 최대치가 제한된다. 이때 최대치 설정값이 지나치게 작으면 제어오차가 커지고, 지나치게 크면 기동시에 전류를 과다하게 사용함. 전류제어의 PI제어 게인은

 - 의 마지막 2개의 파라메터에 의하여 설정됨.
 - Sq 또는 SpHLHLHLHL, HLHLHLHL, HLHL, HLHL ; (F2811인 경우)
 - 의 마지막 2개의 파라메터에 의하여도 설정 가능함. 이 경우는 50000/(SQ 또는 SR 명령에 의한 값) 에 해당하며, 값이 클수록 제어게인이 큰 형식으로 파라메터 설정이 가능함.



- ♣ RJM_VER3로서 IRMCK201 을 사용하는 경우와 RJM_VER7의 경우, 05모드에서는 Sw 명령에 의하여 전류명령값의 최대치가 제한되는데, 이 값이 지나치게 작으면 제어오차가 커지고, 지나치게 크면 기동시에 전류를 과다하게 사용함. 전류제어의 PI제어 게인 MOTOR1_CURRENT_P_GAIN, MOTOR1_CURRENT_I_GAIN은 WQ 또는 WRdddd1,dddd2,dddd3,dddd4,dddd5;의 마지막 두 파라메터에 의하여 설정되며, EsA55A; 명령어에 의하여 BLDCx_PrmBuffer[13]와 BLDCx_PrmBuffer[14]에 저장될 수 있음.
- ♣ RJM_VER7의 경우 Digital Hall IC만을 쓰는 경우는 05번 속도제어가 잘 안됨.
- ♣ IRMCK201 BLDC나 RJM_VER7 BLDC의 경우 03번 모드 위치제어에서 속도모드로 전환할 때, 감가속도값이 지나치게 클수있으므로 가감속도값을 재설정 해주어야함.
- ♣ RJM_VER7 BLDC의 경우 06번 속도제어 모드에서 과전류 에러가 발생하였을 경우, 속도 명령값을 0으로 주면 에러상테에서 회복됨.
- ♣ 가속/감속시에 전원전압의 지나친 감소/증가를 억제하기 위한 기능이 있으며, Sddddd1,dddd2,,dddd3,,dddd4; 명령어를 사용하여 억제하는 정도를 설정할 수 있음.



제어모드의 설정 및 제어작동의 실행 (전류제어)

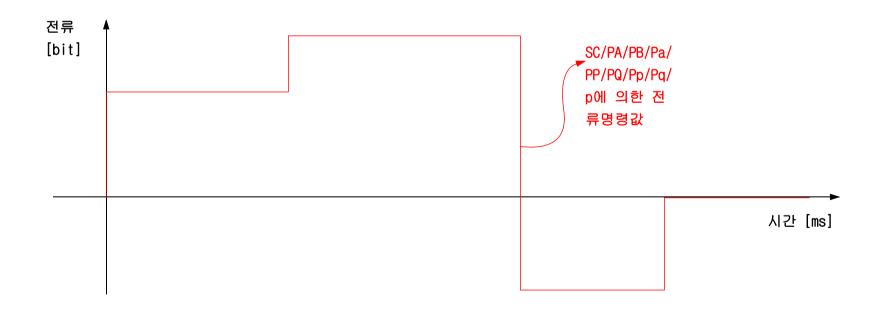
- ♣ 전류제어를 하려면
 - 1) SMHLHL; 명령어로 07의 전류제어 모드를 선택함.

상위/하위바이트가 07이면 모터2/모터1이 전류제어모드(DC 모터 or BLDC 모터 모두적용)

- 2) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함)
- 4) SCiiii1,iiii2; 명령어를 사용하여 전류명령값을 전송함. (단위는 bit 인데 -1023/4095 ~ +1023/4095 범위임)
- 5) 전류제어모드를 종료하려면 SMHLHL; 명령어로 00의 제어 모드를 선택하고,

PDHLHL; 명령어로 해당 모터제어보드의 출력을 disable 시킴. (HLHL은 보드번호를 써야함)

♣ 전류모드로 설정된 모터는 PA,PB,Pa,PP,PQ,Pp,Pq,p 명령을 수행할 때 전류값이 설정됨.





제어모드의 설정 및 제어작동의 실행 (전류제어)

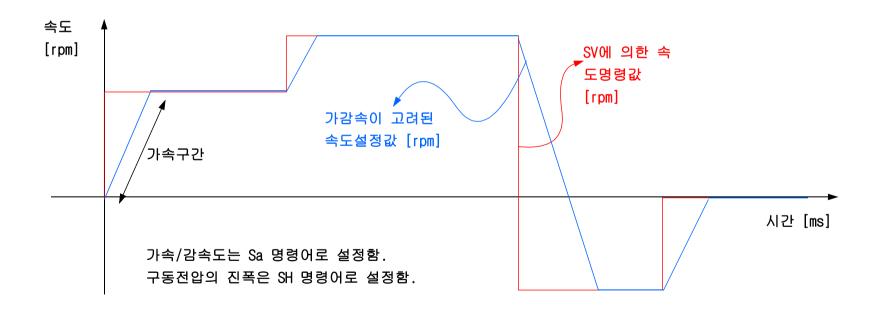
- ♣ DC모터 07번 전류제어의 미제어 게인의 설정은
 - SQ 또는 SRHLHLHLHL, HLHLHLHL, HLHL, HLH, HLHL, HL

 - 의 마지막 2개의 파라메터에 의하여 설정 가능함.
 - Sq 또는 SpHLHLHLHL, HLHLHLHL, HLHL, H
 - 의 마지막 2개의 파라메터에 의하여도 설정 가능함. 이 경우는 50000/(SQ 또는 SR 명령에 의한 값) 에 해당하며, 값이 클수록 제어게인이 큰 형식으로 파라메터 설정이 가능함.
- ♣ BLDC 모터 07번 전류제어의 PI제어 게인 MOTOR1_CURRENT_P_GAIN, MOTOR1_CURRENT_I_GAIN은
 WQ 또는 WRdddd1,dddd2,dddd3,dddd4,dddd5;
 의 마지막 두 파라메터에 의하여 설정되며, EsA55A; 명령어에 의하여 BLDCx_PrmBuffer[13]와 BLDCx_PrmBuffer[14]에 저장될 수 있음.
- ♣ RJM_VER4(2811)/RJM_VER8 DC의 경우 07번 전류제어는 100KHz의 T3PINT_ISR()에서 실시되며, 나머지 DC 모터의 경우는 1ms main loop에서 실시됨.
- ♣ RJM_VER7의 경우 07번 전류제어는 100KHz의 T3PINT_ISR()에서 실시됨.



제어모드의 설정 및 제어작동의 실행 (BLDC 모터의 host-direct-angle-control)

- 🗘 BLDC 모터의 host-direct-angle-control을 하려면
 - 1) SM 명령어로 08번 모드를 선택함.
 - 2) PE 명령어로 모터의 작동개시을 개시함.
 - 3) SHiiii1, iiii2; 명령어로 구동전압의 진폭을 설정함.
 - 4) SViiii1,iiii2; 명령어로 회전속도를 설정함. (단위는 rpm임, 단 pols pair의 역수비 만큼 모터출력축 실제회전속도는 줄어듦) 이때 가감속도는 Sadddd1,dddd2,dddd3,dddd4; 명령어로 설정할 수 있음.





제어모드의 설정 및 제어작동의 실행 (STEPPING 모터의 host-direct-angle-control)

- ♣ STEPPING 모터의 host-direct-angle-control을 하려면
 - 1) SM 명령어로 08번 모드를 선택함.
 - 2) PE 명령어로 모터의 작동개시을 개시함.
 - 3) SHiiii1,iiii2,iiii3,iiii4,iiii5,iiii6; 명령어로 구동전류의 진폭과 OFFSET을 설정함. (필요시) 기본값은 200(200),200(200),0,0,0,0 이며 200(200)/310(409.5) [A]의 명령전류 진폭으로 설정되었음을 의미함.
 - 4) Shdddd1,dddd2,dddd3,dddd4; 명령어로 구동전류의 제한크기와 정지시의 전류감소율을 설정함. (필요시) 기본값은 300(400),300(400),1,1 이며 300/310 [A]이상의 명령전류 진폭에 대하여는 300(400)/310(409.5) [A] 으로 제한된다는 뜻이며, 모터가 정지시에는 1/8 의 비율로 명령전류 진폭을 감소시킨다는 뜻임. 만약 SH 명령어로 명령전류진폭을 300(400) 이상으로 크게할수록 전류파형은 사인파에서 사각파형태로 변화됨.
 - 5) 속도제어를 하려면

SViiii1,iiii2; 명령어로 회전속도를 설정함. (100은 초당 100번 전류의 sine파를 교번하는 것과 같음) 이때 가감속도는 Sadddd1,dddd2,dddd3,dddd4; 명령어로 설정할 수 있음.

기본값은 9523,9523,9523,9523 이며 1초에 0 pulse-per-sec에서 95230 pulse-per-sec으로 속도를 올리는 정도의 가속을 의미함.

또는 위치제어를 하려면

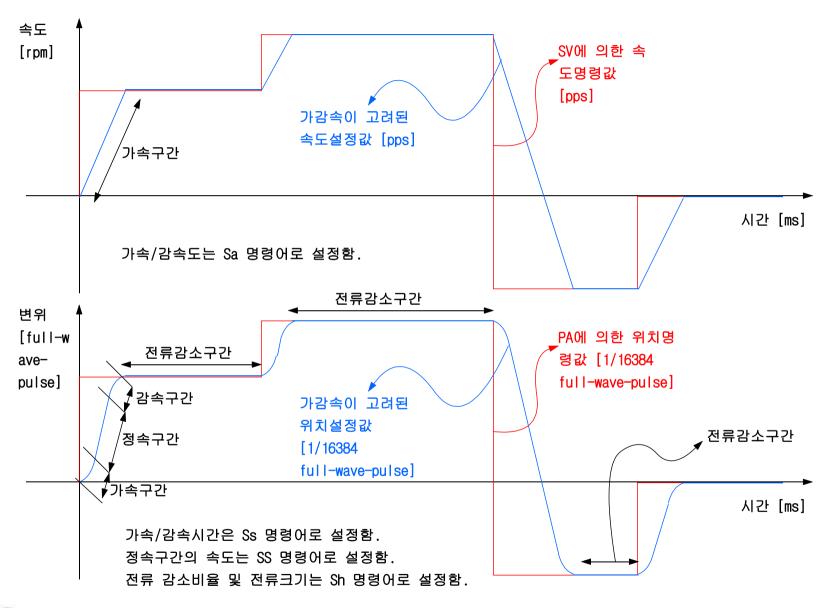
PAIIII1,IIII2; 명령어로 회전각을 설정함. (16384 는 한번 전류의 sine파를 교번하는 것과 같음) 이때 속도는 SSdddd1,dddd2; 명령어로 설정되며, 가감속 시간은 Ssdddd1,dddd2; 명령어로 설정할 수 있음. 기본값은 각각 49,49 와 0,0 이며 49 pulse-per-sec의 정속구간 속도와 0 ms의 가감속 시간을 의미함.

♣ QP 명령어로 현재의 회전 변위값을 읽을 수 있으며, 5,000,000의 기본값으로부터 변화된 값이 16384일 때, 전류의 sine파가 한번 교번된 회전변위(full-wave-pulse)에 해당함. 통상 스테핑모터에서 회전당 200pulse 라면 이는 회전당 200/4=50 full-wave-pulse 에 해당함. 이 full-wave-pulse 값은 SEA55A 명령어의 마지막 두개의 파라메터로 설정이 가능함 (16진수임).

- ♣ QC 명령어로 4개 코일의 전류치를 읽을 수 있음.
- ♣ SH 명령어로 설정된 전류진폭이 Sh 명령어로 설정된 제한치보다 크면 전류사인파는 saturation된 사인파가 됨.
- ♣ 구동력이 부족하면 SH500.500.0.0.0.0;Sh1000.1000.8.8;으로 명령전류 진폭을 키우거나 공급전압의 크기를 키워야함.
- ♣ 공급전압의 크기는 삼상스테핑모터의 경우 8V 정도가 적당하며, 2상스테핑모터의 경우는 24V 가 적당하나 회전력의 크기뿐 아니라 전류제어 안정성과 관련이 있으므로 적절한 전압의 선정이 필요함.



제어모드의 설정 및 제어작동의 실행 (위치제어 및 속도제어에서의 가감속 설정)





제어모드의 설정 및 제어작동의 실행 (위치제어 및 속도제어에서의 가감속 설정)

RJM_VEF	SV 명령어	Sa 명령어	Ss 명령어	SS 명령어
1/2/3(C C)/4/8	speedx_cmd=set_va lue[rpm]*SPEED_SC ALE_FACTOR;	<pre>speedx_acc_rate=set_value[rpm/100ms] speedx_acc_pulses=(speedx_acc_rate*SPE ED_SCALE_FACTOR*MOTORx_PULSE_PER_REV)/ 6000000L;</pre>	<pre>speedx_acc_pulses = speedx_set/ MOTORx_ACC_DEC_PERIOD_SET</pre>	<pre>speedx_set=(speedx_set_RPM*SPEED_SC ALE_FACTOR*MOTORx_PULSE_PER_REV)/60 000L;</pre>
3(BLDC)	speedx_cmd=set_va lue[rpm]*SPEED_SC ALE_FACTOR;	<pre>speedx_acc_rate=set_value[rpm/100ms] SpdAccRate=(speedx_acc_rate*((BLDCx_Pr mBuffer[VelCtrl/2]>>9)&0x0007)/8333L; speedx_acc_pulses=(speedx_acc_rate*SPE ED_SCALE_FACTOR*MOTORx_PULSE_PER_REV)/ 6000000L;</pre>	<pre>speedx_acc_pulses = speedx_set/ MOTORx_ACC_DEC_PERIOD_SET</pre>	<pre>speedx_set=(speedx_set_RPM*SPEED_SC ALE_FACTOR*MOTORx_PULSE_PER_REV)/60 000L;</pre>
5/6		<pre>speedx_acc_rate=set_value[rpm/100ms] speedx_acc_pulses=(speedx_acc_rate*SPE ED_SCALE_FACTOR*MOTORx_PULSE_PER_REV*M OTORx_NUMBER_of_POLE_PAIR)/6000000L;</pre>	<pre>speedx_acc_pulses = speedx_set/ MOTORx_ACC_DEC_PERIOD_SET</pre>	<pre>speedx_set=(speedx_set_RPM*10*SPEED _SCALE_FACTOR*MOTORx_PULSE_PER_REV* MOTORx_NUMBER_of_POLE_PAIR)/600000L ;</pre>
7	speedx_cmd=set_va lue[rpm]*SPEED_SC ALE_FACTOR;	<pre>speedx_acc_rate=set_value[rpm/ms] speedx_acc_pulses=(speedx_acc_rate*SPE ED_SCALE_FACTOR*MOTORx_PULSE_PER_REV)/ 60000L;</pre>	<pre>speedx_acc_pulses = speedx_set/ MOTORx_ACC_DEC_PERIOD_SET</pre>	<pre>speedx_set=(speedx_set_RPM*SPEED_SC ALE_FACTOR*MOTORx_PULSE_PER_REV)/60 000L;</pre>
비고		speedx_acc_rate=100ms 동안의 RPM 증가량 (RJM_VER7이 아닌 경우), =1ms 동안의 RPM 증가량 (RJM_VER7인 경우) RJM_VER7의 03/0e 작동 모드에서는 Sa 명령어로 speedx_acc_rate와 speedx_acc_pulses의 설정이 불가하나speedx_acc_pulses는 계산함.	speedx_acc_pulses=가속시 1ms 동안의 펄스수 증가량 *SPEED_SCALE_FACTOR (RJM_VER5/6이 아닌 경우), =가속시 1ms 동안의 펄스수 증 가량 *SPEED_SCALE_FACTOR*MOTORx_N UMBER_of_POLE_PAIR (RJM_VER5/6인 경우)	speedx_set=speedx_set_RPM에서의 1ms 동안 펄스수*SPEED_SCALE_FACTOR (RJM_VER5/6이 아닌 경우), =speedx_set_RPM에서의 1ms 동안 펄스 수*SPEED_SCALE_FACTOR*MOTORx_NUMBER _of_POLE_PAIR (RJM_VER5/6인 경우)



제어모드의 설정 및 제어작동의 실행 (사인파 위치제어)

- ♣ 사인파 위치제어 동작을 하려면 (2811/28334 버젼에서만 가능함)
 - 1) SMHLHL; 명령어로 Oe/Of의 사인파 위치제어 모드를 선택함.

상위/하위바이트가 Of이면 모터2/모터1이 사인파 위치제어모드가 선택되며 내부에서 실행될때는 DC motor의 경우 02번모드, IRMCK201 BLDC motor의 경우 02번모드, BLDC sensor type의 경우는 02번모드, STEP motor의 경우는 08번모드로 실행됨.

상위/하위바이트가 0e이면 모터2/모터1이 사인파 위치제어모드가 선택되며 내부에서 실행될때는 DC motor의 경우 01번모드, IRMCK201 BLDC motor의 경우 03번모드, BLDC sensor type의 경우는 03번모드, STEP motor의 경우는 08번모드로 실행됨.

- 2) S1dddd1,dddd2; 또는 S2dddd1,dddd2; 명령어로 모터1 또는 모터2의 사인파 주파수와 진폭을 설정함. 이때 진폭과 주파수는 설정순간에 즉각 변화되지 않고 위상과 진폭의 연속성을 유지하면서 서서히 변화됨. 변화의 크기는 주파수는 10ms당 0.01Hz, 진폭은 1ms당 1 pulse 임.
- 3) PEHLHL; 명령어로 해당 모터제어보드의 출력을 enable 시킴. (HLHL은 보드번호를 써야함) 이때 현 위치를 중립점으로 하여 설정주기의 설정 진폭의 사인파 위치제어 작동을 개시함. 이때 진폭과 주파수는 서서히 증가됨.
- 4) S1dddd1.dddd2; 또는 S2dddd1.dddd2; 명령어로 모터1 또는 모터2의 사인파 주기와 진폭을 바꿀 수 있음.



제어모드의 설정 및 제어작동의 실행 (channel 2 모터의 dummy 작동)

♣ DC/BLDC 모터의 경우에 channel 2 모터 구동회로가 dummy 모터 구동모드로 작동하는 것을 enable/disable 할 수 있음. (2811/28334 버젼에서만 가능함)

- 1) SX?; 명령어로 현재의 통신 및 작동 설정값을 확인함.
- 2) channel 2 회로가 dummy 모터의 구동을 하게 하려면, SX?; 명령어로 읽은 값에다 0x0100을 OR 한 값을 SXHLHL; 명령어로 설정함. channel 2 회로가 dummy 모터의 구동을 못 하게 하려면, SX?; 명령어로 읽은 값에다 0xFEFF를 AND 한 값을 SXHLHL; 명령어로 설정함.
- 3) EsA55A; 명령어로 EEPROM에 값을 저장함.
- 4) 전원을 껏다 켜면, dummy 모터 설정에 따른 동작을 개시하며, dummy 모터 모드로 작동이 enable 된 경우, 모터 제어 channel 2의 PWM duty는 channel 1의 PWM duty를 copy하는 형태로 모터를 구동함.



제어모드의 설정 및 제어작동의 실행 (sequence 제어 : 프로아텍 초기 버젼인 경우만 해당됨)

- ♣ sequence 제어의 기능을 enable 하려면
 - 1) sequence 작동에 필요한 프로그램이 입력되어 있다고 가정함.
 - 2) SX?; 명령어로 현재의 통신 및 작동 설정값을 확인함.
 - 3) 이미 sequence 제어모드로 작동중 이라면, SX?; 명령어로 읽은 값에다 0x7FFF을 AND 한 값을 SXHLHL; 명령어로 설정하고, SM0000; 명령어를 입력함. (sequence control mode를 끔, 이미 sequence 제어모드로 작동중인 상태였다면 이 작업을 생략하면 안됨)
 - 4) 필요하면 다른 파라메터들을 수정하거나, EDA55A; 명령어로 파라메터를 초기화한 후에 다른 파라메터들을 수정함.
 - 5) SX?; 명령어로 읽은 값에다 0x8000을 OR 한 값을 SXHLHL; 명령어로 설정함. (sequence control mode를 설정함)
 - 6) EsA55A; 명령어로 EEPROM에 값을 저장함.
 - 7) 전원을 껏다 켜면, 별도의 설정 없이 sequence 작동을 개시함.
- 🕹 sequence 제어의 기능을 disable 하려면
 - 1) SX?; 명령어로 현재의 통신 및 작동 설정값을 확인함.
 - 2) SX?; 명령어로 읽은 값에다 Ox7FFF을 AND 한 값을 SXHLHL; 명령어로 설정하고, SM0000; 명령어를 입력함. (sequence control mode를 끔, 생략하면 안됨)
 - 3) 필요하면 다른 파라메터들을 수정하거나, EDA55A; 명령어로 파라메터를 초기화한 후에 다른 파라메터들을 수정함.
 - 4) EsA55A; 명령어로 EEPROM에 값을 저장함.
 - 5) 전원을 껏다 켜도, sequence 작동을 하지 않음.



♣ 일반사항

- 1. 2811을 사용한 제어기에서만 가능함.
- 2. 시퀜스자동화 프로그램의 입력 : XP 명령어를 사용하여 RAM의 3F8700h ~ 3F89ffh-000080h 번지에 저장함.
- 3. 시퀜스자동화 프로그램의 EEPROM으로의 저장 : XPsA55A; 명령어를 사용하여 RAM에 있는 시퀜스자동화 프로그램을 EEPROM의 0x500 ~ 0x7ff-0x080 영역에 총 1536-256 byte를 저장함.
- 4. 시퀜스자동화 모드와 slave 모드의 전환 : 머신초기화 과정에서 EEPROM의 시퀜스자동화 프로그램을 RAM영역으로 복사하고나서, operation_mode_SWITCH의 최상위bit를 체크하여 최상위bit가 1이면 시퀜스자동화 모드가 개시됨.
- 5. 시퀜스자동화 모드의 실행 : mainLoopStep_cnt==4 일때 10ms 마다 X_PORT_STATUS = PORT_STATUS_FOR_X_CONTROL; 을 실행한 후에 시퀜스 자동화 프로그램을 해독하여 실행함.
- 6. step-by-step으로 프로그램을 시키고자 하면 operation_mode_SWITCH의 최상위-1 bit를 1로 저장하면 되며, step 작동을 실행하는 명령은 XPS; 임.
- 7. 주요 변수로는

PORT STATUS FOR X CONTROL : port의 ON/OFF 상태를 가지고 있는 16bit 변수로서 그 값은 매 10ms 마다 갱신됨.

X_PORT_STATUS : 매 10ms 마다 PORT_STATUS_FOR_X_CONTROL 값으로 갱신되며 시퀜스 제어 프로그램 내에서 조작이 가능함.

X_process_cnt : 시퀜스 제어 프로그램의 실행 과정을 알려주는 변수값으로서 시퀜스 제어 프로그램 내에서 조작이 가능함.

X STATUS FLAG : 시퀜스 제어 프로그램에 의하여 조작이 가능한 16bit의 변수

X_timer0 : 매 10ms 마다 1씩 증가하는 타이머 값임.

X_timerO_period : X_timerO의 주기값으로서 기본값은 100 (1초) 임.

X_counter0 : 카운터 값으로서 시퀜스 제어 프로그램 내에서 조작이 가능함.

8. 사용가능한 명령어의 형태 : 아래에 기술되어 있음

명령어는 IN, JPHLHL을 앞에 포함하거나 안 하거나 할수있으며, 이어서 뒤에는 blank일수도 있고, 제어기가 처리가능한 명령어일 수도 있음. IN, JPHLHL 이외의 명령어가 있을 경우에 process counter는 1이 자동 증가되며, 뒷부분에 있는 나머지 시퀜스 제어 프로그램은 처리하지 않음. IN, JPHLHL 의 명령어가 있을 경우에는 process counter를 변화시키며, 뒷부분에 있는 나머지 시퀜스 제어 프로그램은 처리하지 않음.



♣ Process counter와 관련한 명령어

```
XPLTHI HI ORHI HI :
                      X process cnt가 HLHL보다 작으면 X PORT STATUS 값에 HLHL을 OR 시킴.
                      X process cnt가 HLH 보다 작으면 X PORT STATUS 값에 HLHL을 AND 시킴.
XPLTHLHLANHLHL:
XPLTHLHLXOHLHL;
                       X_process_cnt가 HLHL보다 작으면 X_PORT_STATUS 값에 HLHL을 XOR 시킴.
XPLTHLHLEQHLHL명령어;
                       X process cnt가 HLHL보다 작으면 X PORT STATUS 값이 HLHL과 같을 때 명령어를 수행함.
XPLTHLHLSn명령어:
                      X process cnt가 HLHL보다 작으면 PORT STATUS FOR X CONTROL 값의 n 번째 비트가 1이면 명령어를 수행함.
XPLTHLHLRn명령어;
                      X process cnt가 HLHL보다 작으면 PORT STATUS FOR X CONTROL 값의 n 번째 비트가 0이면 명령어를 수행함.
XPLTHLHLLD;
                      X process cnt가 HLHL보다 작으면 PORT STATUS FOR X CONTROL 값을 X PORT STATUS 값에 복사함.
XPLTHLHLorHLHL;
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값에 HLHL을 OR 시킴.
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값에 HLHL을 AND 시킴.
XPLTHLHLanHLHL;
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값에 HLHL을 XOR 시킴.
XPI THI HI XOHI HI :
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값이 HLHL과 같을 때 명령어를 수행함.
XPLTHLHLeaHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 X_STATUS_FLAG 값의 n 번째 비트가 1이면 명령어를 수행함.
XPLTHLHLsn명령어;
XPLTHLHLrn명령어;
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값의 n 번째 비트가 0이면 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 X STATUS FLAG 값에 HLHL값을 복사함.
XPLTHLHLIdHLHL;
XPLTHLHLIN명령어;
                      X_process_cnt가 HLHL보다 작으면 X_process_cnt 값을 증가시키고 명령어를 수행함.
XPLTHLHLJPHLHL명령어;
                      X process cnt가 HLHL보다 작으면 X process cnt 값을 HLHL으로 세트하고 명령어를 수행함.
XPLTHLHLES;
                      X process cnt가 HLHL보다 작으면 이후의 명령을 해독하여 처리하지 않고 빠져나감.
XPI THI HI MI HI HI HI 명령어:
                      X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 <= (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLMGHLHLHL명령어;
                      X process cnt가 LLLL보다 작으면 (LLLL 번지의 값 >= (UINT)LLL)일 때 명령어를 수행함.
XPLTHLHLM<HLHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 < (UINT)HLHL)일 때 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 값 > (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLM>HLHLHL명령어;
                      X process cnt가 LLLL보다 작으면 (LLLL 번지의 값 == (UINT)LLL)일 때 명령어를 수행함.
XPLTHLHLM=HLHLHLHL명령어;
XPLTHLHLM+HLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 HLHL번지의 값을 1 증가시키고 명령어를 수행함.
XPLTHLHLM-HLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 HLHL번지의 값을 1 감소시키고 명령어를 수행함.
                      X process cnt가 LLLL보다 작으면 LLLL번지의 값을 0으로 clear 시키고 명령어를 수행함.
XPLTHLHLMRHLHL명령어;
```



```
XPLTHLHLTLHLHL명령어;
                      X process cnt가 HH 보다 작으면 (X timerO의 값 <= (UINT)HH )일 때 명령어를 수행함.
                      X process cnt가 HLH 보다 작으면 (X timerO의 값 >= (UINT)HLHL)일 때 명령어를 수행함.
XPI THI HI TGHI HI 명령어:
                      X_process_cnt가 HLHL보다 작으면 (X_timer0의 값 < (UINT)HLHL)일 때 명령어를 수행함.
XPI THI HI T<HI HI 명령어:
                      X process cnt가 베베보다 작으면 (X timerO의 값 > (UINT)베베)일 때 명령어를 수행함.
XPLTHLHLT>HLHL명령어;
XPLTHLHLT=HLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (X_timer0의 값 == (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLT+명령어;
                      X process cnt가 HLHL보다 작으면 X timer0의 값을 1 증가시키고 명령어를 수행함.
XPLTHLHLT-명령어:
                      X process cnt가 HLHL보다 작으면 X timer0의 값을 1 감소시키고 명령어를 수행함.
XPLTHLHLTR명령어;
                      X process cnt가 HHL보다 작으면 X timer0의 값을 0으로 clear 시키고 명령어를 수행함.
XPLTHLHLTPHLHL명령어;
                      X process cnt가 HLHL보다 작으면 X timer0의 period 값을 HLHL로 설정함.
XPLTHLHLCLHLHL명령어;
                      X process cnt가 HLH 보다 작으면 (X counter0의 값 <= (UINT)HLHL)일 때 명령어를 수행함.
XPI THI HI CGHI HI 명령어:
                      X process cnt가 H네보다 작으면 (X counter0의 값 >= (UINT)H네)일 때 명령어를 수행함.
XPLTHLHLC<HLHL명령어;
                      X process cnt가 베베보다 작으면 (X counter0의 값 < (UINT)베베 일 때 명령어를 수행함.
XPLTHLHLC>HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (X counter0의 값 > (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLC=HLHL명령어;
                      X process cnt가 H네보다 작으면 (X counter0의 값 == (UINT)H네)일 때 명령어를 수행함.
                      X process cnt가 LLL보다 작으면 X counter0의 값을 1 증가시키고 명령어를 수행함.
XPLTHLHLC+명령어:
XPLTHLHLC-명령어;
                      X process cnt가 LLL 보다 작으면 X counter이의 값을 1 감소시키고 명령어를 수행함.
XPLTHLHLCR명령어;
                      X process cnt가 H네보다 작으면 X counter0의 값을 0으로 clear 시키고 명령어를 수행함.
                      X_process_cnt가 HLHL보다 작으면 (LS1C의 AD 변환값 <= (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLBLHLHL명령어;
XPI THI HI BGHI HI 명령어:
                      X process cnt가 HLHL보다 작으면 (LS1C의 AD 변환값 >= (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLB<HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (LS1C의 AD 변환값 < (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLB>HLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (LS1C의 AD 변환값 > (UINT)HLHL)일 때 명령어를 수행함.
                      X process cnt가 베베보다 작으면 (LS1C의 AD 변환값 == (UINT)베베)일 때 명령어를 수행함.
XPLTHLHLB=HLHL명령어;
XPLTHLHLbLHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (LS2C의 AD 변환값 <= (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLbGHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (LS2C의 AD 변환값 >= (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLb<HLHL명령어:
                      X process cnt가 HLHL보다 작으면 (LS2C의 AD 변환값 < (UINT)HLHL)일 때 명령어를 수행함.
                      X_process_cnt가 HLHL보다 작으면 (LS2C의 AD 변환값 > (UINT)HLHL)일 때 명령어를 수행함.
XPLTHLHLb>HLHL명령어;
XPLTHLHLb=HLHL명령어;
                      X process cnt가 베베보다 작으면 (LS2C의 AD 변환값 == (UINT)베베)일 때 명령어를 수행함.
```



```
XPLTHLHLNLHLHLHL명령어;
                      X process cnt가 H.H.보다 작으면 (H.H.번지의 값 <= (int)H.H.)일 때 명령어를 수행함.
XPLTHLHLNGHLHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 값 >= (int)HLHL)일 때 명령어를 수행함.
XPI THI HI N<HI HI HI HI 명령어:
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 값 < (int)HLHL)일 때 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLN>HLHLHLHLB B OH;
XPLTHLHLN=HLHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 == (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLNIHLHLHL명령어;
                      X process cnt가 베베보다 작으면 (베베번지의 절대값 <= (int)베베)일 때 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 절대값 >= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLNOHLHLHLHL명령어;
                      X process cnt가 비비보다 작으면 (비비번지의 절대값 < (int)비비)일 때 명령어를 수행함.
XPLTHLHLNiHLHLHL명령어;
XPLTHLHLNoHLHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (HLHL번지의 절대값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLVLHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 값 <= (int)HLHL)일 때 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 값 >= (int)HLHL)일 때 명령어를 수행함.
XPI THI HI VGHI HI 명령어:
XPLTHLHLV<HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLV>HLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (speed1_RPM의 값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLV=HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 값 == (int)HLHL)일 때 명령어를 수행함.
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 절대값 <= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLVIHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (speed1_RPM의 절대값 >= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLVOHLHL명령어;
XPLTHLHLViHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 절대값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLVoHLHL명령어:
                      X process cnt가 HLHL보다 작으면 (speed1 RPM의 절대값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWLHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed2 RPM의 값 <= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWGHLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed2 RPM의 값 >= (int)HLHL)일 때 명령어를 수행함.
                      X_process_cnt가 HLHL보다 작으면 (speed2_RPM의 값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLW<HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed2 RPM의 값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLW>HLHL명령어;
XPLTHLHLW=HLHL명령어;
                      X process cnt가 HLHL보다 작으면 (speed2 RPM의 값 == (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWIHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (speed2_RPM의 절대값 <= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWOHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (speed2_RPM의 절대값 >= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWiHLHL명령어:
                      X process cnt가 HLHL보다 작으면 (speed2 RPM의 절대값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLWoHLHL명령어;
                      X_process_cnt가 HLHL보다 작으면 (speed2_RPM의 절대값 > (int)HLHL)일 때 명령어를 수행함.
```



```
XPLTHLHLvLHLHL명령어;
                       X process cnt가 HH 보다 작으면 (speed1 cmd old의 값 <= (int)HH)일 때 명령어를 수행함.
XPLTHLHLvGHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed1 cmd old의 값 >= (int)HLHL)일 때 명령어를 수행함.
XPI THI HI v<HI HI 명령어:
                       X process cnt가 HLHL보다 작으면 (speed1 cmd old의 값 < (int)HLHL)일 때 명령어를 수행함.
                       X process cnt가 HH 보다 작으면 (speed1 cmd old의 값 > (int)HH )일 때 명령어를 수행함.
XPLTHLHLv>HLHL명령어;
XPLTHLHLv=HLHL명령어;
                       X_process_cnt가 HLHL보다 작으면 (speed1_cmd_old의 값 == (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLvIHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed1 cmd old의 절대값 <= (int)HLHL)일 때 명령어를 수행함.
                       X process cnt가 HLHL보다 작으면 (speed1 cmd old의 절대값 >= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLvOHLHL명령어:
XPLTHLHLviHLHL명령어;
                       X process cnt가 HHL보다 작으면 (speed1 cmd old의 절대값 < (int)HHL)일 때 명령어를 수행함.
XPLTHLHLvoHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed1 cmd old의 절대값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLwLHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 값 <= (int)HLHL)일 때 명령어를 수행함.
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 값 >= (int)HLHL)일 때 명령어를 수행함.
XPI THI HI wGHI HI 명령어:
XPLTHLHLw<HLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLw>HLHL명령어;
                       X_process_cnt가 HLHL보다 작으면 (speed2_cmd_old의 값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLw=HLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 값 == (int)HLHL)일 때 명령어를 수행함.
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 절대값 <= (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLwIHLHL명령어;
XPLTHLHLwOHLHL명령어;
                       X_process_cnt가 LLLL보다 작으면 (speed2_cmd_old의 절대값 >= (int)LLL)일 때 명령어를 수행함.
XPLTHLHLwiHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 절대값 < (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLwoHLHL명령어;
                       X process cnt가 HLHL보다 작으면 (speed2 cmd old의 절대값 > (int)HLHL)일 때 명령어를 수행함.
XPLTHLHLHLHLHLHLHLHL명령어;
                           X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 <= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLmGHLHLHLHLHL명령어;
                           X process cnt가 LLLL보다 작으면 (HLHL번지의 값 >= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLm<HLHLHLHLHLHLHL명령어;
                           X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 < (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLm>HLHLHLHLHLHL명령어;
                           X_process_cnt가 HLHL보다 작으면 (HLHL번지의 값 > (ULONG)HLHLHLHL)일 때 명령어를 수행함.
                           X process cnt가 HLHL보다 작으면 (HLHL번지의 값 == (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLm=HLHLHLHLHLHL명령어;
XPLTHLHLm+HLHL명령어;
                           X_process_cnt가 HLHL보다 작으면 HLHL번지의 값을 1 증가시키고 명령어를 수행함.
XPLTHLHLm-HLHL명령어;
                           X_process_cnt가 HLHL보다 작으면 HLHL번지의 값을 1 감소시키고 명령어를 수행함.
XPLTHLHLmRHLHL명령어;
                           X process cnt가 HLHL보다 작으면 HLHL번지의 값을 0으로 clear 시키고 명령어를 수행함.
```



```
XPLTHLHLPLHLHLHL명령어;
                            X process_cnt가 HLHL보다 작으면 (position1 <= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLPGHLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position1 >= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPI THI HI P<HI HI HI HI 명령어:
                            X_process_cnt가 HLHL보다 작으면 (position1 < (ULONG)HLHLHLHL)일 때 명령어를 수행함.
                            X process cnt가 HH 보다 작으면 (position1 > (ULONG)HHHHH)일 때 명령어를 수행함.
XPLTHLHLP>HLHLHL명령어;
XPLTHLHLP=HLHLHLHL명령어;
                            X_process_cnt가 HLHL보다 작으면 (position1 == (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLQLHLHLHLH모명령어;
                            X_process_cnt가 HLHL보다 작으면 (position2 <= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLQGHLHLHLHL명령어:
                            X process cnt가 HLHL보다 작으면 (position2 >= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLQ<HLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position2 < (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLQ>HLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position2 > (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLQ=HLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position2 == (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLpLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position1 set <= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLpGHLHLHLHL명령어;
                            X_process_cnt가 HLHL보다 작으면 (position1_set >= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLp<HLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position1 set < (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLp>HLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position1 set > (ULONG)HLHLHLHL)일 때 명령어를 수행함.
                            X process cnt가 HLHL보다 작으면 (position1 set == (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLp=HLHLHLHL명령어;
XPLTHLHLqLHLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position2 set <= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLqGHLHLHLHL명령어;
                            X_process_cnt가 HLHL보다 작으면 (position2_set >= (ULONG)HLHLHLHL)일 때 명령어를 수행함.
                            X_process_cnt가 HLHL보다 작으면 (position2_set < (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLq<HLHLHL명령어;
XPLTHLHLa>HLHLHL명령어;
                            X process cnt가 HLHL보다 작으면 (position2 set > (ULONG)HLHLHLHL)일 때 명령어를 수행함.
XPLTHLHLq=HLHLHLHL명령어;
                            X_process_cnt가 HLHL보다 작으면 (position2_set == (ULONG)HLHLHLHL)일 때 명령어를 수행함.
```

