

§ iM - Σ 시리즈의 특징

iM - Σ 시리즈는 최신 32Bit High Performance Floating Point DSP 및 16Bit High Speed DSP에 의해 제어되는 Full Digital 방식의 고성능 Amplifier의 채용으로 초고속, 고정도 제어 및 Motion의 안정성을 구현하여 최상의 제어를 실현할 수 있는 다축 서보 컨트롤러입니다.

iM - Σ 시리즈의 주요 특징

모션 컨트롤, Amplifier 및 Sequence 기능을 내장하여 본 제품만으로도 완벽한 제어시스템 구성

Full Digital 방식의 2축 Module형 서보 앰프 채용

Module형 S/W의 채용으로 S/W 재 생산성 향상(전용기화 용이)

직렬 비동기 통신을 통한 원격 운전 및 프로그램 작성 및 각종 Data 작성 / 편집 가능

채널 구조에 의한 다축 동시 또는 개별 제어 가능(설비 투자 원가 절감)

모션 동작 중 Data 편집 및 각종 모니터링을 위한 Multi - Working 구조

동시 다양한 모터 운전 가능(Servo Motor, Linear Motor, Step Motor...)

Module간 CAN 통신에 의한 제어(제어축의 확장 용이, Noise 향상...)

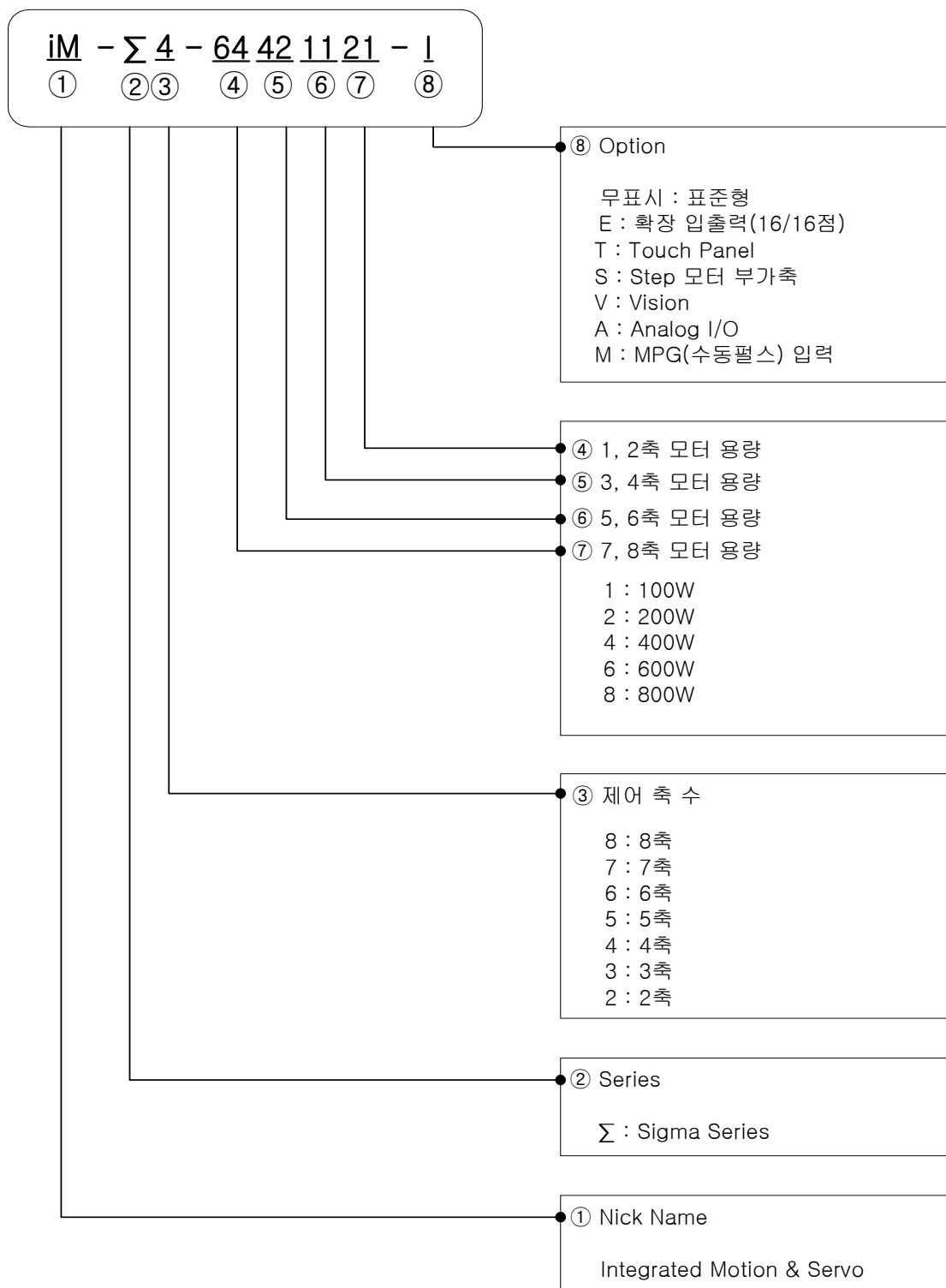
유지 보수의 편리성을 위한 Rack Style 구조

다양한 외부 Interface(Touch Panel, Vision, MPG)

iM - Σ 시리즈의 개요

1-1 iM - Σ 컨트롤러 개요

1-1-1 제품의 형명 구성



1-1-2 제품의 규격

제품에 대한 일반적인 규격은 아래와 같습니다.

제어 축수		2축	3/4축	5/6축	7/8축
전원	전원 전압	1Ø AC 220V, 50/60Hz, +10% ~ -15%			
	전원 용량[kVA]	Max 2	Max 3		
위치 검출기 형식	Servo Motor	Incremental (9/15선식) – Line Driver 출력형			
제어 S P E C	Program Tool	Operating Loader 또는 Personal Computer(IBM Compatible)			
	Position 입력	MDI(Manual Direct Input), DTI(Direct Teaching Input), MPG PC(IBM Compatible), Operating Loader, Touch Panel			
	프로그램 용량	모션 프로그램 (1,000스텝*100프로그램), 시퀀스 프로그램 (10개) 포인트 (1,000포인트*100프로그램)			
	공용 변수	공용 포인트 변수, 공용 실수형 변수, 공용 정수형 변수 : 각 256개 공용 내부 I/O : 256 Bytes(H/W I/O 포함)			
	제어 정도	엔코더의 ±1 Pulse 이내			
	통신 방식	외부통신 : RS232C or RS422, Module간 통신 : CAN Bus 방식			
입/출력	System I/O	기본 : 입력 32점 / 출력 32점 (System I/O : System 프로그램 사용자 작성)			
	User I/O	확장(IO Module 1개당) : 입력 16점 / 출력 16점, 24V Photo Coupler 확장 가능 최대 I/O : 입력 128점 / 출력 128점			
다이나믹 브레이크		기본 장착			
회생 방전 저항		기본 장착			
냉각 방식		강제 송풍 냉각			
안전 특성(보호)		IPM 이상, 과부하, 저/과전압, 추종 에러, 엔코더 에러, 전기각 검출 에러, 비상 정지, 과속도, 메모리 에러, H/W 리밋 에러 등			


* 사용되는 엔코더는 5V Line Driver 출력형으로 Incremental 형식입니다. 적용하는 모터에 따라 다음과 같은 형식이 있습니다.

적용 Motor	엔코더 형식(Incremental)
Dasatech(DA Series)	Incremental Encoder(9, 15선식)
Tamagawa(TBLi), 삼성(CSM..)	2048 or 2000 P/R(9,15선식)
LG-OTIS(CN,TF, ...)	2000 ~ 6000 P/R(15선식)

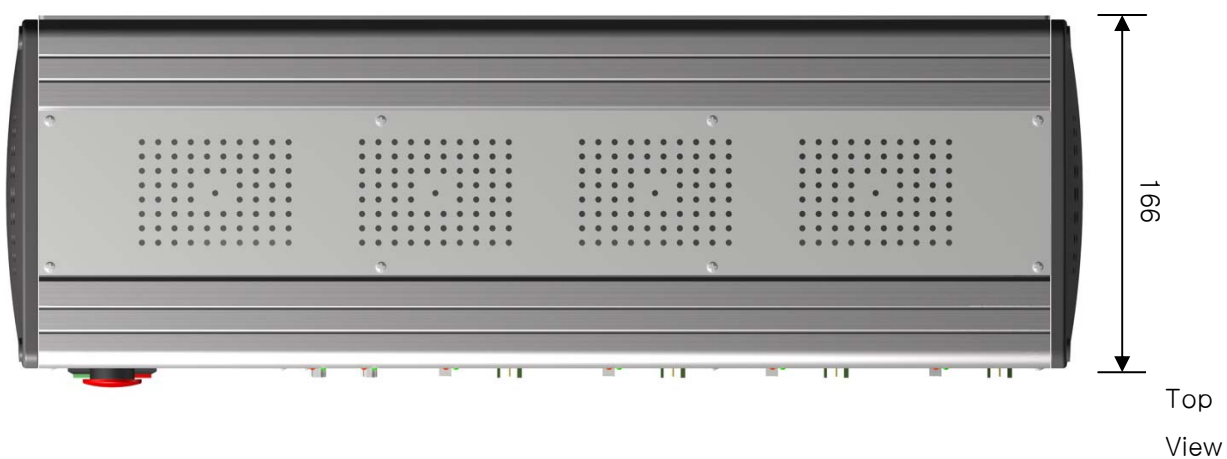
1-1-3 일반 사용 규격

제품의 일반 사용 규격은 아래와 같습니다. 아래 사용 규격을 불가피하게 만족시키지 못하는 상황이 발생하면 당사 고객지원팀으로 문의하여 주시기 바랍니다.

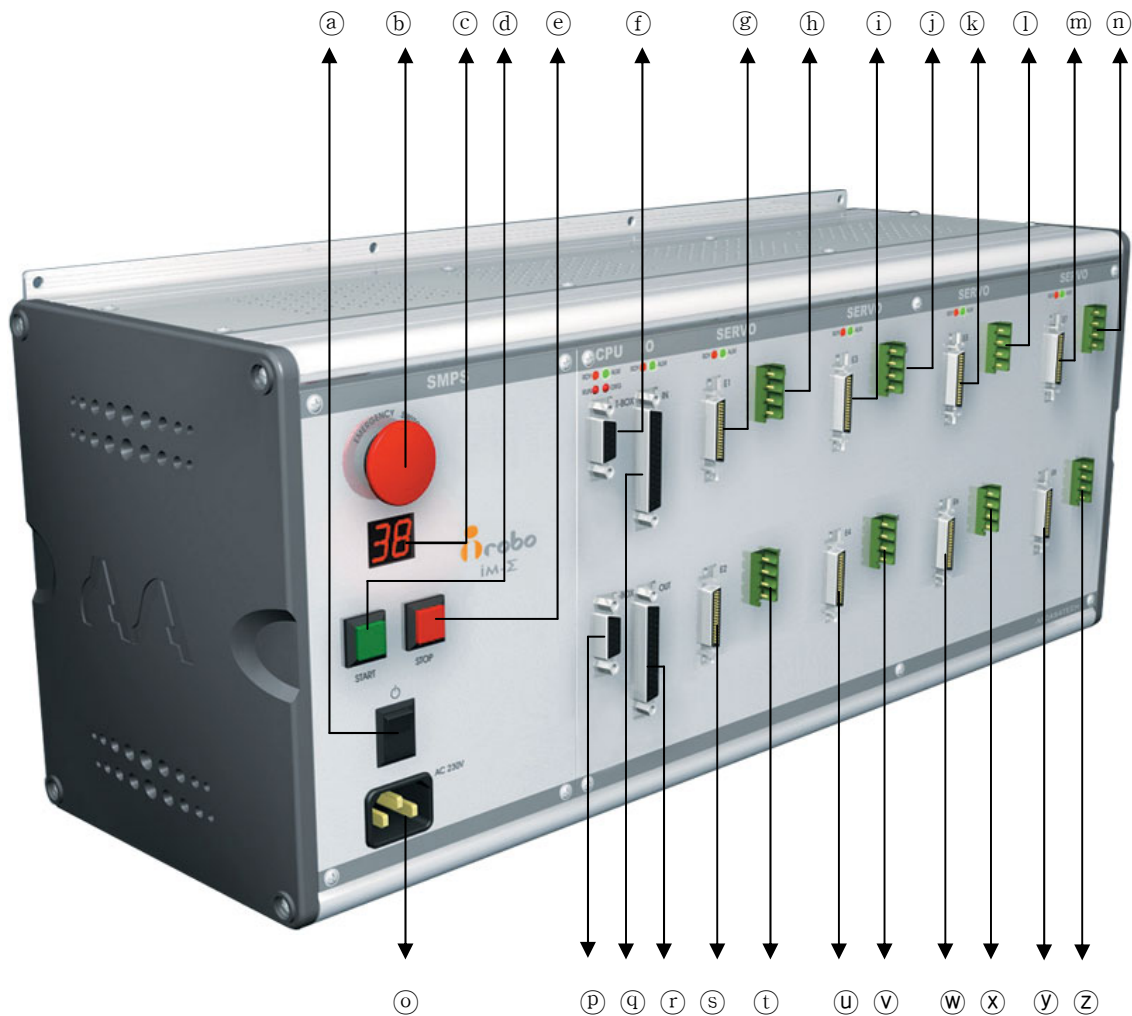
항 목	내 용
주위 환경	설치 장소에 부식성, 폭발성 가스가 없는 곳
사용 주위 온도	0 ~ 50℃
사용 주위 습도	20 ~ 80% (결로 현상이 없을 것)
보존 주위 온도	-15 ~ 60℃
보존 주위 습도	10 ~ 90% (결로 현상이 없을 것)
진동	0.5G (4.9m/s ²)

	임의로 일반 사용 규격을 위배하여 사용할 경우, 당사는 제품 불량에 대한 책임이 없습니다.
--	--

1-1-4 제품 외관 및 치수(8축 구성의 경우)



1-1-5 제품 설명

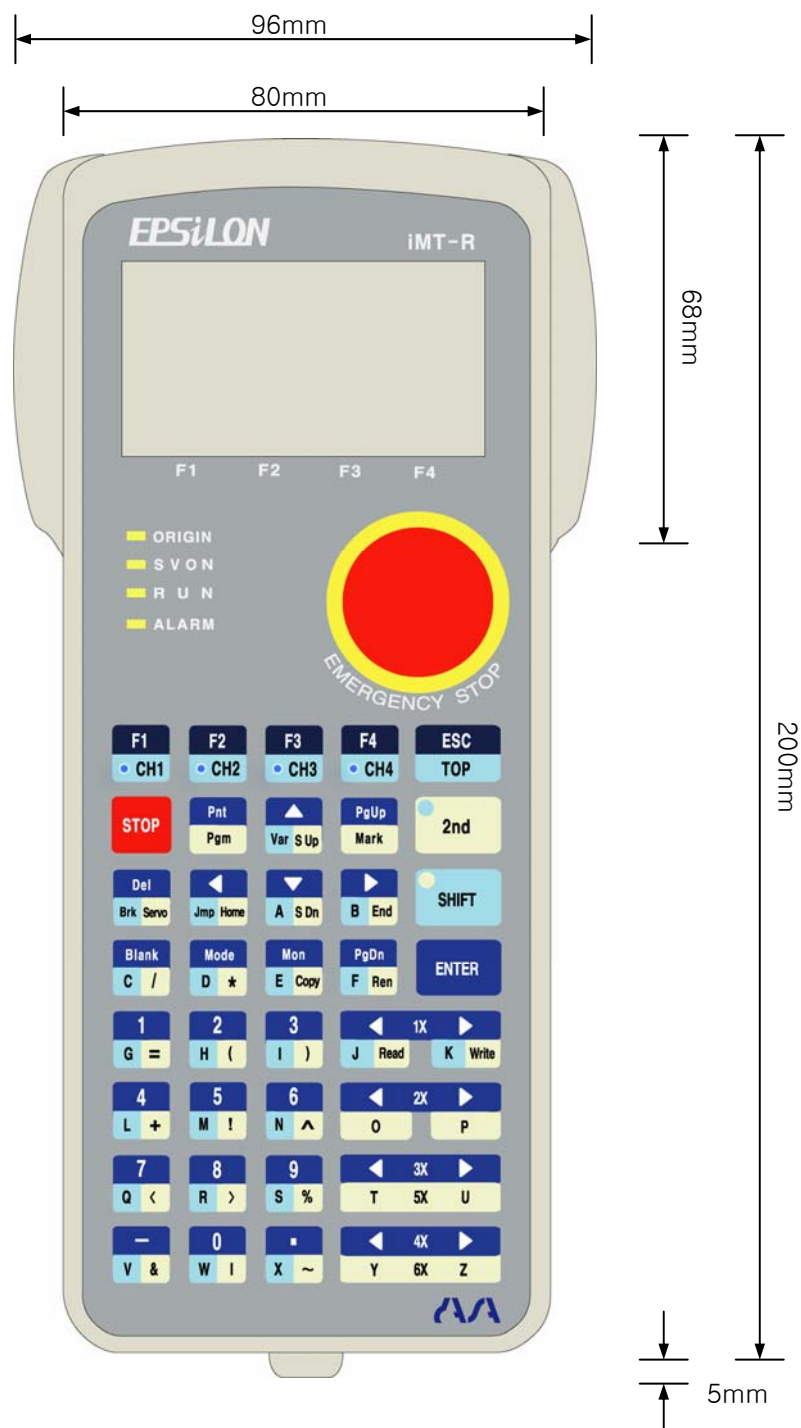


- ① 전원 Switch
; 컨트롤러의 전원 On/Off Switch
- ② Emergency Switch
; 비상 정지용 Switch(Normal Open, NO 점점)
- ③ 7 Segment 표시부
; 현재 모드, 동작 프로그램 No, Alarm Code Display용
- ④ Operation Switch #1
; Org/Motion Program Start Switch
- ⑤ Operation Switch #2
; Org/Motion Program Stop Switch
- ⑥ Teach Pendant Connector
; Teach Pendant 연결용 커넥터
- ⑦ 1축 Encoder/Sensor Connector
; 1축 엔코더, 센서 연결용 커넥터

- Ⓜ 1축 Motor Connector
; 1축 모터 연결용 커넥터
- Ⓜ 3축 Encoder/Sensor Connector
; 3축 엔코더, 센서 연결용 커넥터
- Ⓜ 3축 Motor Connector
; 3축 모터 연결용 커넥터
- Ⓜ 5축 Encoder/Sensor Connector
; 5축 엔코더, 센서 연결용 커넥터
- Ⓜ 5축 Motor Connector
; 5축 모터 연결용 커넥터
- Ⓜ 7축 Encoder/Sensor Connector
; 7축 엔코더, 센서 연결용 커넥터
- Ⓜ 7축 Motor Connector
; 7축 모터 연결용 커넥터
- Ⓜ 전원용 Cable 연결 단자
; 1 Φ ac 220V (+10%, -15%) 전원 공급용 Cable 연결 단자(제품 출하할 때 제공)
- Ⓜ RS-232/422 통신용 Connector
; PC, Touch Panel 연결 통신용 커넥터
- Ⓜ User Input Connector
; 사용자 입력 연결용 커넥터
- Ⓜ User Output Connector
; 사용자 출력 연결용 커넥터
- Ⓜ 2축 Encoder/Sensor Connector
; 2축 엔코더, 센서 연결용 커넥터
- Ⓜ 2축 Motor Connector
; 2축 모터 연결용 커넥터
- Ⓜ 4축 Encoder/Sensor Connector
; 4축 엔코더, 센서 연결용 커넥터
- Ⓜ 4축 Motor Connector
; 4축 모터 연결용 커넥터
- Ⓜ 6축 Encoder/Sensor Connector
; 6축 엔코더, 센서 연결용 커넥터
- Ⓜ 6축 Motor Connector
; 6축 모터 연결용 커넥터
- Ⓜ 8축 Encoder/Sensor Connector
; 8축 엔코더, 센서 연결용 커넥터
- Ⓜ 8축 Motor Connector
; 8축 모터 연결용 커넥터

1-2 오퍼레이팅 로더(Operating Loader)

1-2-1 Operating Loader의 외관



1-2-2 규격

- 1) 컨트롤러와 연결 케이블 길이는 3m 표준입니다(15m까지 연장 가능).
- 2) 컨트롤러와 통신은 RS232 통신 방식을 사용합니다(고정).

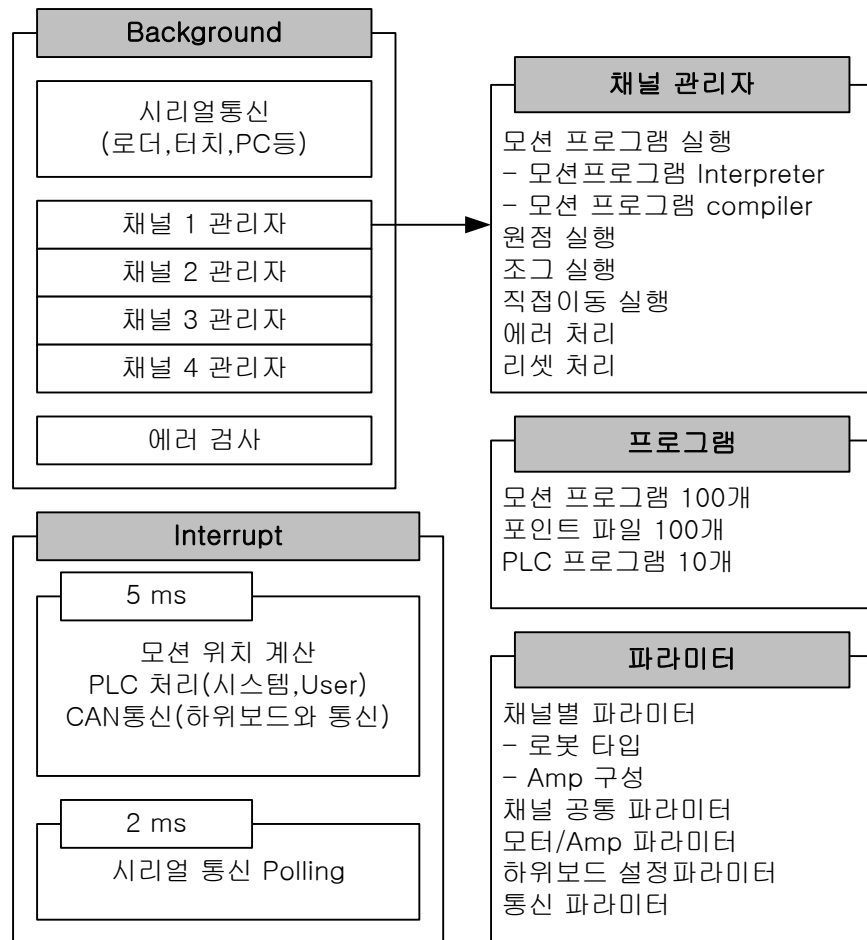
1-2-3 오퍼레이팅 로더 설명

- 1) LCD 창 : 컨트롤러와의 통신 내용이 표시됩니다.
- 2) LED : 컨트롤러의 현재 상태가 표시됩니다.
 - ORIGIN : 컨트롤러가 원점 완료 후 점등됩니다.
(다 채널 사용시 모든 채널의 원점 완료 후 점등됩니다.)
 - SVON : 모터가 서보 온이 되면 점등됩니다.
(다 채널 사용시 어떠한 채널의 모터라도 서보 온이 되면 점등됩니다.)
 - RUN : 모션 프로그램이 실행중일 때 점등됩니다.
(다 채널 사용시 어떠한 채널의 모션 프로그램이라도 실행중일 때 점등됩니다.)
 - ALARM : 컨트롤러로 비상 정지 신호가 입력되었을 경우 또는 컨트롤러에서 각종 Alarm이 발생하였을 경우 점등 됩니다.
(다 채널 사용시 어떠한 채널이라도 Alarm이 발생하였을 경우 점등됩니다.)
- 3) EMG S/W : 스위치를 누를 경우 비상 정지됩니다.
- 4) 조작용 Key : 컨트롤러의 각 화면 전환, 프로그램/포인트/변수 수정 및 파라미터 수정을 할 경우 사용됩니다.
조작 Key에 대한 설명은 “3장 오퍼레이팅 로더 조작”을 참조하십시오.

1-3 소프트웨어 구조 및 로봇 운전에 대한 일반사항

본 절에서는 로봇을 운전하기 위한 제어기 내부의 소프트웨어 구조와 특징, 사용법들에 대해 설명합니다. 각 소프트웨어 모듈의 특징을 이해하고 응용목적에 맞게 적절히 사용하기 바랍니다.

CPU에서는 배경작업과 인터럽트 작업으로 나누어 일을 처리합니다. PLC나 모션 위치 궤적은 일정한 주기마다 갱신될 필요가 있으므로 5ms 인터럽트에서 수행됩니다. 배경작업은 인터럽트에서 실행되고 남은 시간에 실행되며 여러 개의 프로세스가 수행됩니다. 채널 관리자도 하나의 프로세스로서 수행되게 됩니다. PLC나 PC를 통해서 컨트롤러를 제어하는 경우, 약간 다른 절차에 의해서 명령이 수행됩니다. 먼저 PLC를 통하는 경우, I/O카드를 통해서 접점신호를 받게되고 이는 5ms인터럽트를 통해서 Main B'd에 전달됩니다. 그 후, System PLC를 거쳐서 Run이나 원점,조그등의 동작을 실행하게 되는데, 내부 시스템 접점을 검사하여 명령으로 받아들이는 작업은 배경작업으로 수행하게 됩니다. PC를 통하여 제어하게 되는 경우에는 시리얼 통신을 통하여 명령을 받게 되므로 배경작업을 통하여 명령을 받아서 배경작업을 통하여 명령이 처리되는 경우라 할 수 있습니다. 전체 시스템은 이런 명령체계에 잘 응답하도록 설계되었으므로 PLC에서 5ms동안의 Active시간을 갖는 D명령어를 사용해도 잘 동작하도록 되어 있습니다. 단, 모션 프로그램의 컴파일이나 포인트 Loading시에는 한 채널에서 배경작업을 독점하고 있게 되어 그 시간동안에는 다른 배경작업이 동작하지 않게 되니 주의하시기 바랍니다.



로봇의 운전은 원점 운전, 포인트 교시, 모션 프로그램 수행으로 이루어 집니다. 모션 프로그램과 포인트 파일은 각 100개씩 저장가능하며, 모션 프로그램 실행시에는 같은 번호의 포인트 파일이 존재할 경우 자동으로 불러오게 됩니다. 없을 경우에는 그냥 실행되며 이름은 달라도 무관합니다. 예를 들어 프로그램 번호가 10인 A.PGM이 있을 경우 프로그램을 선택하여 Run이 될 경우, 프로그램 번호가 9인 A.PNT와 프로그램 번호가 10인 B.PNT가 있다고 하면, 이름이 같은 A.PNT가 아니라 번호가 동일한 B.PNT가 불러지게 됩니다. 다른 번호의 포인트 파일을 사용하고자 할 경우에는 모션 프로그램 내에서 포인트 파일을 불러오는 명령어를 사용하여야 합니다. 이에 관한 사항은 <4장. 모션명령어>를 참고하시기 바랍니다. PLC 프로그램은 총 10개 까지 저장이 가능하며 외부 PLC 컨트롤러 없이 간단한 PLC기능을 수행할 수 있습니다.

1-3-1 원점 운전

로봇을 운전 할시에는 보통 원점운전을 거쳐 모션프로그램을 실행하는 단계로 이루어 집니다. 원점 운전이란 로봇의 특정 위치가 일정한 좌표값을 가지도록 하여, 포인트 좌표의 유효성을 보장하는 작업입니다.

증가형(Incremental) 엔코더를 사용하는 경우, 로봇이 이동함에 따라 모터의 엔코더에서는 이동한 변화량에 대한 정보만을 제공하기 때문에 로봇의 절대적인 원점을 알기 위해서는 센서등의 부가적인 장치를 사용하게 됩니다. 이때 정해진 위치는 0 또는 파라미터 설정을 통하여 임의의 좌표값으로 변경할 수 있고 포인트 좌표값의 기준이 됩니다.

포인트 값은 원점이 존재 할 때 의미를 가지며 원점이 바뀌었을 경우에는 같은 목표 좌표라 하더라도 로봇이 다른 위치로 이동하게 되기 때문에 주의를 요합니다. 원점은 모터의 변경이나 센서 위치의 변경 또는 원점 관련 파라미터의 변경에 의해서 변할 수 있습니다.

정밀성이 필요한 경우에는 원점운전시 로봇의 이동속도를 낮추어야 할 필요가 있습니다. CPU가 일정한 시간마다 센서 신호를 검사한다고 할 때, 속도가 빠른 경우 센서 신호를 검사하는 조밀성이 떨어지기 때문입니다. 원점 모드에는 모터의 C(Z)상을 사용하는 경우와 센서만을 사용하는 경우가 있는데, 모터의 C(Z)상을 사용하는 경우에는 CPU의 인터럽트를 사용하여 C상 신호시의 펄스를 기억하지만, 센서만을 사용하는 경우에는 배경작업에 의해서 센서를 감지하므로 약간의 오차가 더 발생할 수 있습니다. 원점운전 속도는 OrgSenSpd, OrgZSpd 파라미터에서 설정가능하며 자세한 내용은 7장 파라미터를 참고하십시오.

1-3-2 모션 프로그램

원점 운전과 포인트 교시가 끝난 경우 모션프로그램 실행이 가능한 단계가 됩니다. 본 제어기는 범용 제어기로서 포인트 파일 뿐만이 아니라 모션 프로그램이 실제 로봇의 움직임에 중요한 역할을 담당하게 됩니다. 모션 프로그램은 실행되기 전에 컴파일되어 더 작은 명령어들로 제어기 내부에 저장되어집니다. 이 작은 명령어는 편의상 마이크로 명령어라고 부르도록 합니다. 여러 채널을 동시에 운전할 경우, 한 채널에서는 한 개의 마이크로 명령어를 수행하고 다른 채널이 실행될 수 있도록 합니다. 이에 대한 이해는 채널별 협조 운전시에 필요할 수 있습니다.

1-3-3 포인트 파일

포인트 교시된 내용은 포인트 파일에 저장됩니다. 모션프로그램에서는 포인트 파일에 저장된 포인트 값을 기본 포인트 파일이나 다른 포인트 파일을 불러와서 사용할 수 있습니다. 포인트가 포인트 파일에 저장시에는 포인트 번호가 함께 저장되지만 모션 프로그램에서 포인트 파일을 불러올 경우 번호없이 번호순서대로 메모리에 불러오게 되는데, 없는 포인트는 단순한 공백만 차지하게 됩니다. 따라서 메모리의 낭비를 가져오므로 긴 모션 프로그램과 많은 포인트 파일을 사용시는 포인트 파일에 의한 메모리 낭비를 막기위해서 비어 있는 포인트 없이 순서대로 포인트 번호를 사용하시기 바랍니다. 예를 들어, 포인트 번호 300번만 정의되어 있는 포인트 파일의 경우, 이 포인트 파일을 로딩할 경우 포인트 번호가 0~300이 모두 정의되어 있는 포인트 파일과 같은 크기의 메모리를 사용하게 됩니다.

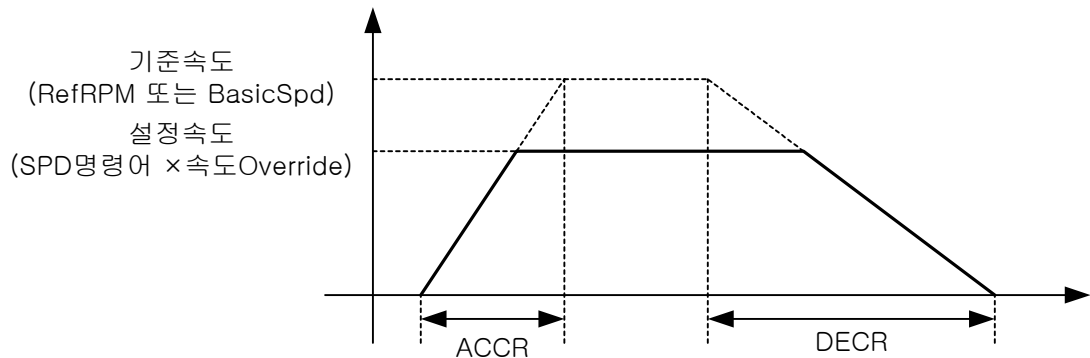
1-3-4 PLC 프로그램

PLC 기능을 사용하여 자동화 기능과 상위 제어기와 연동 및 하위 제어기 제어등의 작업을 수행할 수 있습니다. 전원을 켜를 때, 자동으로 원점을 잡고 모션 프로그램을 수행하는 기능은 PLC 프로그램을 통하여 실현할 수 있습니다. 원점운전이나 조그, 실행, 정지 등의 동작을 제어하는 시스템 접점은 다 채널구조로 인하여 채널마다 모든 접점을 할당할 경우, 사용하지 않는 포트도 발생할 수 있고 많은 포트가 필요하게 되므로 시스템 PLC라는 특권을 가진 PLC 프로그램을 통하여 외부 I/O 카드와 내부의 S/W 시스템 접점을 Mapping하도록 하였습니다. 자세한 사항은 <6장. 시퀀스 명령어>를 참고하십시오.

1-4 이동 명령어의 속도 계산 과정

로봇은 시작점에서 목표점까지 가속속 등의 속도 패턴을 가지고 이동하도록 되어 있습니다. 속도결정과 관련하여 이동 명령어들은 PTP(Point-To-Point)이동과 CP이동(직선, 원, 원호 보간)으로 분류됩니다. PTP이동에서는 각 축에 할당된 속도를 사용하여 각 이동거리를 속도로 나누어 최대의 시간이 소요되는 축에 맞추어 나머지 축들이 목표점에 동시에 도착할 수 있도록 속도가 조정됩니다. CP이동에서는 파라미터의 BasicSpd에 설정된 속도에 따라 정해진 궤적을 따라 이동할 때 각 축의 속도가 자연스럽게 결정됩니다. CP이동에 참여하는 축을 자세축이라고 하고 그에서 제외되는 축을 도구축이라고 합니다. CP이동시 도구축의 이동은 자세축의 시작점에서 목표점까지의 이동비율에 따라 비례적으로 이동합니다.

PTP이동과 CP이동시 자세축의 속도는 합성거리에 대해서 합성된 속도를 계산하여 각 축의 이동비율에 따라 배분하는 방식으로 계산됩니다. 합성속도는 사다리꼴의 속도 패턴을 갖습니다. 가속과 감속시의 기울기는 ACCR, DECR 2개의 모션 명령어에 의해서 결정됩니다.



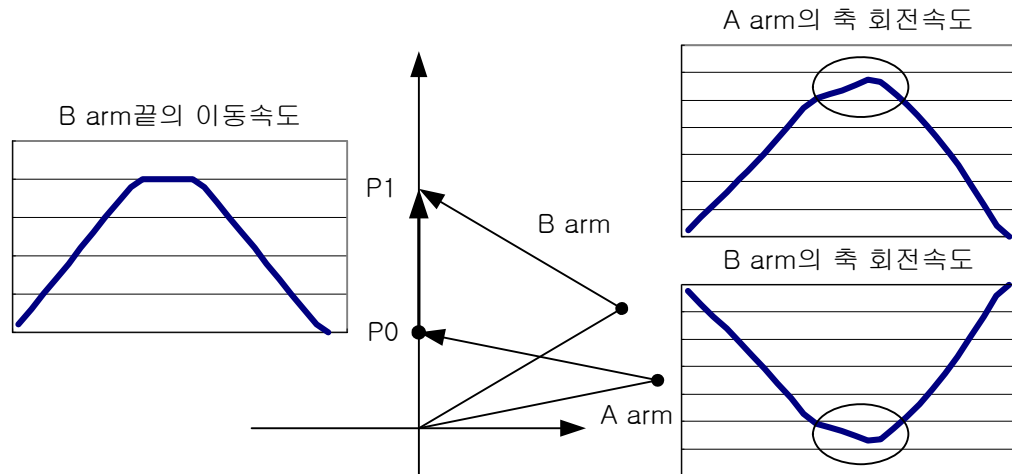
1-4-1 PTP 이동시 속도 패턴

PTP이동시 축별 이송속도를 계산하는 과정은 다음과 같습니다. 합성속도의 속도 패턴이 사다리꼴 형태이므로 이를 각 축에 배분한 속도도 사다리꼴 형태를 갖습니다.

- └▶ 시작점(pulse):<0,0,0,0> 목표점(pulse):<200,400,100,0>
 RefRPM:<3000,2000,1500,1000>
- └▶ 이송시간:<0.067,0.2,0.067,0> => 최대값은 0.2로 Y축이 기준축
- └▶ 합성거리 = $L = \sqrt{(200-0)^2 + (400-0)^2 + (100-0)^2 + (0-0)^2} = 100\sqrt{21} = 458.3$
- └▶ 축별이동비율 = $\langle 200/L, 400/L, 100/L, 0/L \rangle = \langle 0.436, 0.873, 0.218, 0 \rangle$
- └▶ 합성이송속도 = 기준축의 RefPRM * 모션설정 속도 / 기준축의 이동비율
 = $2000 * 1 / 0.873 = 2291$
- └▶ 축별이송속도 = 축별이동비율 * 모션설정 속도 * 합성이송속도
 = $\langle 0.436, 0.873, 0.218, 0 \rangle * 1 * 2291 = \langle 999, 2000, 499, 0 \rangle$
 Y축은 자신의 RefRPM 속도로 이동

1-4-2 CP 이동시 속도 패턴

직교로봇의 경우는 각축이 한 개의 좌표변화를 전담하고 있으므로 PTP이동에서의 속도와 CP 이동에서의 속도 패턴이 동일합니다. 따라서, 보다 정확한 이해를 위해 Scara의 예를 들어서 각 축의 속도는 어떻게 변하는지 설명합니다.



P0에서 P1으로 B Arm의 끝이 사다리꼴의 속도 패턴을 가지고 이동할 때, A,B Arm 각 축의 회전속도는 사다리꼴의 속도와는 다른 모양이 됩니다. 이와 같이 같은 CP속도라고 하더라도 작업영역의 어느 부분을 어느 방향으로 이동하느냐에 따라서 축별 회전속도는 달라지게 됩니다.

1-4-3 도구축의 속도 패턴

CP이동에 참여하는 축을 자세축, 참여하지 않는 축을 도구축이라하고, CP이동시 도구축은 자세축의 이동비율에 맞추어 이동하게 됩니다.

기구부의 형태가 XYZT이고 $\langle 0,0,0,0 \rangle$ 에서 $\langle 100,200,300,50 \rangle$ 으로 이동할 때, 자세축이 $\langle 0,0,0 \rangle$ 에서 $\langle 100,200,300 \rangle$ 으로 보간이동할 때, T축은 0에서 50까지 이동하게 됩니다. 주의할 점은 자세축의 시작점과 목표점이 동일하거나 매우 작는데 반해 도구축의 좌표값의 차이가 큰 경우, 이동비율이 순간적으로 0%에서 100%로 변하게 되므로 도구축의 이동속도가 매우 빨라지게 되어 Following Error나 Over Speed등의 에러가 발생할 수 있습니다.