

COMIZOA Motion Controller System

COMI-Motion Builder Manual



*COMputer Innovation
is Zoomed by Our Affection!*



저작권자 : **㈜커미조아**

Copyright (c) by COMIZOA CO.,LTD. All right reserved.

2003 년 9 월 23 일 초판 발행

이 사용자 설명서는 저작권법에 의해 보호되고 있습니다.

㈜커미조아의 사전 서면 동의 없이 사용자설명서의 일부 또는 전체를 어떤 형태로든 복사, 전
재할 수 없습니다.

Hardware Support : Hardware@comizoa.co.kr

Software Support : Software@comizoa.co.kr



㈜커미조아

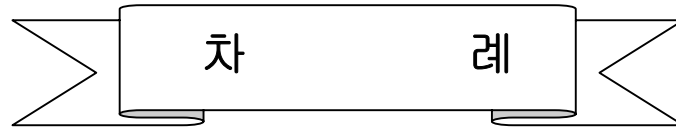
www.comizoa.co.kr

www.comizoa.com

Tel) 042 - 861 - 3301~3

Fax) 042 - 861 - 3304





COMI-MOTIONBUILDER MANUAL PART	7
1. 디바이스 로드 및 설정	8
1.1 디바이스 로드	8
1.2 Init & Config 설정	9
1.3 Interrupt Monitor Config	14
1.4 Source Build Option	16
1.5 3D Emulator Config	16
1.6 Velocity Config	17
2 모니터링 도구	19
2.1 Digital I/O	19
2.2 MIO Status	20
2.3 3D Emulator	21
2.4 Interrupt Monitor	24
3 모션 오퍼레이션	26
3.1 Single Axis Move	26
3.2 Multi Axis Move	28
3.3 Interpolation	29
3.4 Home Return	35
3.5 Manual Pulser	36
3.6 External Switch	37
3.7 Position Compare	38

3.8	<i>Listed Motion</i>	40
4	부가기능과 사용자 팁.....	42
4.1	모션관련 부가기능.....	42
4.2	GUI 부가기능.....	43

=====



시작하는 글

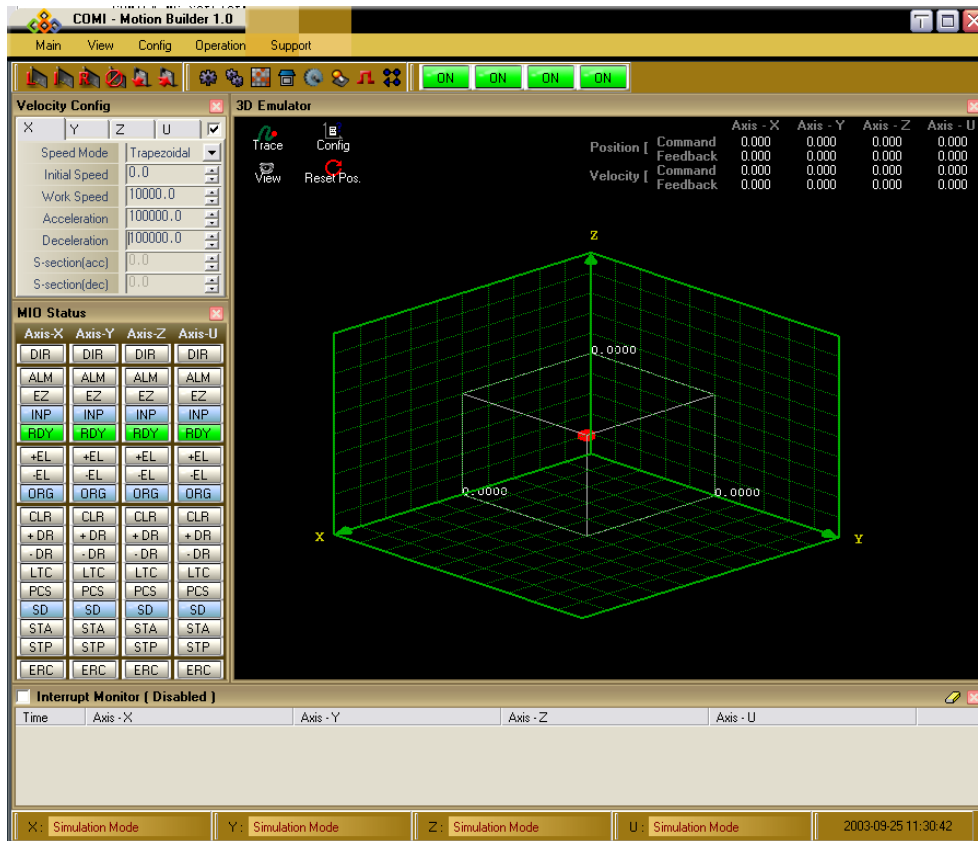
COMI-MotionBuilder 유틸리티 프로그램

본 매뉴얼에서는 COMI-MotionBuilder 유틸리티 프로그램을 설명합니다. COMI-MotionBuilder 유틸리티 프로그램은 (주) 커미조아 모션 제어용 디바이스 전용 유틸리티 프로그램으로서 모션 제어용 디바이스 구입시 무상으로 제공됩니다. 본 프로그램은 디바이스의 전반적인 기능 테스트를 손쉽게 수행할 수 있도록 하였으며, 구조물 없이도 3D 에뮬레이터를 이용한 경로추적 및 각종 모니터링이 가능한 가상구동을 통해, 알고리즘 설계시에 적은 리스크로 실험할 수 있도록 도움을 줄 수 있습니다. 또한, 코드 자동생성기능을 통하여, 보다 빠르고 쉬운 응용프로그램 제작 환경을 지원합니다.

COMI-MotionBuilder Manual Part

‘LX Series Manual -Motion Part- 2. 장치 제어기 및 소프트웨어 설치’를 참조하여 소프트웨어 설치를 마치면 COMI-MotionBuilder 라는 유틸리티 프로그램이 설치됩니다. 이 프로그램은 모션 제어용 디바이스에 적용되는 프로그램으로서 디바이스의 전반적인 기능을 테스트해볼 수 있도록 하며, 가상 구동 및 코드 생성, 모니터링 기능을 제공하는 프로그램입니다.

COMI-MotionBuilder 를 실행하면 다음과 같은 화면이 나타납니다.¹⁾



[그림 1] COMI-MotionBuilder 주 화면

¹⁾ 최초 실행시에는 [그림 1] 과 같은 화면이 나타나기 전에 ‘Config.cfg’ 와 ‘Layout.cfg’ 파일을 생성한다는 메시지가 뜨게 됩니다. 이는 사용자의 컴퓨터 환경과 설치 경로에 따라 로그파일을 생성하는 것으로, 정상적인 동작입니다.

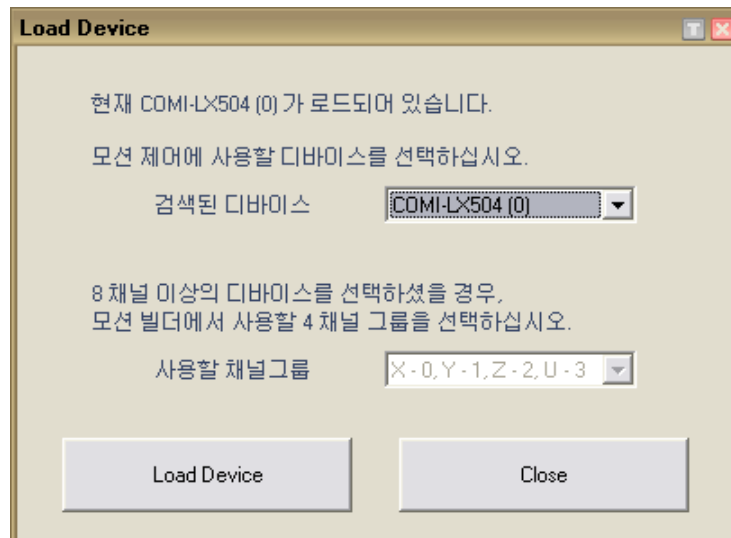
1. 디바이스 로드 및 설정

모션제어용 디바이스는 모터 드라이버를 통한 모션 제어와 모니터링에 앞서, 응용 프로그램으로 로드한 후, 각종 환경에 맞춰 알맞는 설정을 해주어야 합니다. 이번 장에서 설명되는 항목을 꼼꼼히 살펴 모터 드라이버의 특성과 I/O 환경을 제대로 설정하신 후 기구물과의 연결 및 구동을 하셔야, 기구물의 오작동으로 인한 파손을 미연에 방지하실 수 있습니다

1.1 디바이스 로드

COMI-MotionBuilder 프로그램은 실행시, 자동으로 (주) 커미조아의 모션제어용 디바이스를 검색하여 로드하여줍니다. 하지만, 하나의 시스템에 여러보드를 설치하셨을 경우이나, 다른 채널을 제어하고자 하는 경우에는 Load Device 창을 이용하여 사용하고자 하는 디바이스와 대상 채널을 선택하셔야 합니다.¹⁾

[그림 1] COMI-MotionBuilder 주 화면 상단의 Main 메뉴에서 ‘Load Device’ 를 클릭하시면, [그림 2] 와 같은 Load Device 창이 열립니다.



[그림 2] Load Device

¹⁾ COMI-MotionBuilder 은 x, y, z, u 축이라는 이름으로 모터에 연결된 네 채널을 제어, 모니터링합니다. 기본적으로는 디바이스의 0, 1, 2, 3 번 채널을 차례대로 x, y, z, u 축에 맵핑합니다.

자동으로 검색된 디바이스 콤보박스에서 사용하고자 하는 디바이스를 선택합니다. 동일한 이름의 디바이스의 경우, 괄호안의 인스턴스 번호로 각 디바이스를 구분합니다. 인스턴스 번호는 0/S 에 따라, 먼저 장착되어 드라이버가 먼저 설치된 순서로 낮은 인스턴스 번호를 부여받거나, PCI 슬롯의 위치에 따라 CPU 쪽 방향으로 가까운 쪽부터 낮은 인스턴스 번호를 부여받게 됩니다. 사용자의 0/S 에 따른 인스턴스 번호 부여방식을 확인하시기 바랍니다.

COMI-LX508 등의 8 채널 이상의 디바이스를 설치하셨을 경우, 0, 1, 2, 3 번 채널 외의 4, 5, 6, 7 번등의 채널을 x, y, z, u 축에 맵핑하여 사용하고자 하신다면, 사용할 채널그룹 콤보 박스를 이용하여 원하는 채널군을 선택합니다. 참고로, x, y, z, u 축은 채널번호에 맞춰 차례대로 맵핑됩니다.

사용할 디바이스와 채널군을 선택하셨으면, Load Device 버튼을 이용하여, 디바이스를 로딩합니다. 이 때에는 자동으로 예전에 로딩되었던 디바이스가 언로딩된 후, 현재 선택된 디바이스가 로딩됩니다.¹⁾ 만약 'INVALID_HANDLE_VALUE' 라는 메시지가 뜨게 되면 이는 정상적으로 디바이스가 로딩되지 못한 것을 의미합니다. 이 때에는 디바이스 설치 상태나, 드라이버 버전, dll 파일의 버전등을 체크하여, 제대로 로딩이 되도록 조치를 취하셔야 합니다.

1.2 Init & Config 설정

Config 메뉴의 'Device Init & Config' 를 통하여, 전반적인 디바이스의 설정이 가능합니다. 프로그램 실행 시, 저장되어 있던 설정값을 기준으로 디바이스 설정이 이루어지는데, 이는 'Device Init & Config' 를 통하여 변경할 수 있습니다. 또한, 이렇게 변경된 설정값은 자동으로 저장되어, 이는 다음 프로그램 실행시 디바이스 설정에 반영됩니다.

Main 메뉴의 'Device Reset' 을 이용하여 디바이스를 완전 초기화한 경우에도 'Device Init & Config' 를 통하여 디바이스를 재설정해 주셔야합니다.

[그림 1] COMI-MotionBuilder 주 화면 상단의 Config 메뉴에서 'Device Init & Config' 를 클릭하시면, [그림 3] 과 같은 Init & Config 창이 열립니다

¹⁾ 모든 디바이스는 응용프로그램에서 사용하기에 앞서 디바이스 로드과정을 통해, 프로그램으로 각각 로드된 후 사용되어야 합니다. 또한, 디바이스의 사용이 완료되거나 응용프로그램이 종료되기 전에 언로드과정을 통해 여러 설정과 자원을 해제하여야 합니다.

1. 디바이스 로드 및 설정

	Axis - X	Axis - Y	Axis - Z	Axis - U
입력 신호에 필터 로직 사용	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Command 펄스 출력 모드	출력모드 - 4	출력모드 - 4	출력모드 - 4	출력모드 - 4
Feedback 모드, Normal Low/High	1X A/B, Low	1X A/B, Low	1X A/B, Low	1X A/B, Low
출력 펄스 최대 주파수	655350	655350	655350	655350
펄스수 / 논리적 거리	1.00	1.00	1.00	1.00
펄스 출력 속도 / 논리적 속도	1.00	1.00	1.00	1.00
Feedback 펄스 / Command 펄스	1.00	1.00	1.00	1.00

Command / Feedback Alarm / Limit DR / SD / STA / STP CLR / CMP / INP / LTC

Generate Code Apply Close

[그림 3] Init & Config

기본적으로 모든 설정은 축별로 하게 되어 있습니다. 왼쪽의 체크박스를 이용하여 모든 축에 대한 설정을 같이 하실수도 있습니다.

설정값의 종류에 따라 분류되어 있는 ‘Command/Feedback’ ‘Alarm/Limit’ ‘DR/SD/STA/STP’ ‘CLR/CMP/INP/LTC’ 탭을 이동하시며 각종 설정치를 변경하실 수 있으며 이러한 설정값은 [그림 3] 하단의 ‘Apply’ 버튼을 클릭함으로써, 실제로 디바이스에 설정하게 됩니다.

[그림 3] 하단의 ‘Generate Code’ 를 이용하면, 현재 설정되어 있는 값을 기준으로 샘플코드를 생성하여 줍니다. 사용자는 이 코드를 복사 또는 저장하여 프로그램 개발에 사용할 수 있습니다.

각 설정 항목에 대한 의미는 다음과 같습니다. 이를 숙지하시고 설정하시기 바라며, 보다 자세한 설명은 ‘LX Series Manual -Motion Part- 5. C/C++ 라이브러리’ 를 참조하시기 바랍니다.

입력 신호에 필터 로직 사용

각종 I/O 신호에 필터 로직을 적용할지를 설정하는 항목입니다. 필터 로직이 적용되면 펄스폭이 너무 짧은 입력 신호는 무시되게 됩니다.

Command 펄스 출력 모드 (Command / Feedback 탭)

Command 펄스의 출력 모드를 설정하는 항목입니다. 출력모드는 0 에서 5 까지 설정 가능하며 이 6 개의 출력모드의 의미는 다음과 같습니다.

Value	출력 형태			
	(+) 방향 운전 시		(-) 방향 운전 시	
	CW pin	CCW pin	CW pin	CCW pin
0		(High)		(Low)
1		(High)		(Low)
2		(Low)		(High)
3		(Low)		(High)
4		(High)	(High)	
5		(Low)	(Low)	

Feedback 모드, Normal Low/High (Command / Feedback 탭)

Feedback 펄스의 입력 모드를 설정합니다. Feedback 펄스는 실제 모터 또는 구조물의 구동거리를 체크하기 위하여 사용되며 거의 대부분 엔코더 입력을 사용합니다.

사용자는 1 채배, 2 채배, 4 채배에 해당하는 1X A/B, 2X A/B, 4X A/B 항목과, A 펄스 - 카운트 증가, B 펄스 - 카운트 감소에 해당하는 CW/CCW 항목 중 알맞은 항목을 선택하면 됩니다.

또한, Normal Low / Normal High 중 적절한 입력로직을 설정할 수 있습니다.

출력 펄스 최대 주파수 (Command / Feedback 탭)

모션에 적용할 수 있는 최저/최고 속도 범위를 설정하는 항목입니다. 최대 출력 주파수를 설정함으로써, 설정되어지는 최저/최고 속도의 범위는 다음과 같습니다.

최대 출력 주파수 값	출력 펄스의 주파수 범위(Hz)
6,553.5	0.1 to 6,553.5
13,107	0.2 to 13,107
32,767.5	0.5 to 32,767.5

1. 디바이스 로드 및 설정

65,535	1 to 65,535
131,070	2 to 131,070
327,650	5 to 327,650
655,350	10 to 655,350
1,310,700	20 to 1,310,700
3,276,750	50 to 3,276,750
6,553,500	100 to 6,553,500

펄스수 / 논리적거리 (Command / Feedback 탭)

논리적 단위 거리에 대한 펄스 수를 설정하는 항목입니다. 여기서 논리적 단위 거리라 함은 여러 이동에서 사용하는 거리 또는 위치에 대한 단위량을 의미합니다.

펄스 출력 속도 / 논리적 속도 (Command / Feedback 탭)

논리적 단위 속도에 대한 실제 펄스 출력 속도(PPS)를 설정하는 항목입니다. 여기서 논리적 단위 속도라 함은 속도 지정에서 사용하는 속도 또는 가속도에 대한 단위량을 의미합니다.

FeedBack 펄스 / Command 펄스 (Command / Feedback 탭)

Feedback 펄스와 Command 펄스의 분해능 비율(Resolution ratio)을 설정하는 항목입니다. 여기서 Feedback 펄스의 분해능이란 엔코더의 1 회전시에 발생하는 펄스 수를 의미합니다. 그리고 Command 펄스의 분해능이란 모터를 1 회전시키기 위해 필요한 Command 펄스수를 의미합니다.

Alarm 정지 방식, Active Low/High (Alarm / Limit 탭)

Alarm 신호 발생시의 반응 방식과 입력로직을 설정하는 항목입니다. Alarm 신호 발생시, 즉시 정지할 것인지 감속 후 정지할 것인지를 선택할 수 있으며 입력 신호가 Active Low 인지 Active High 인지 입력로직을 설정할 수 있습니다.

Soft Limit 사용 (Alarm / Limit 탭)

소프트웨어적인 Limit 를 사용할 것인지를 설정하는 항목입니다. 소프트웨어적인 Limit 은 리미트센서의 설치가 용이하지 않을 때 안전성을 위하여 소프트웨어적인 리미트를 설정하는 것입니다. 소프트웨어적인 Limit 은 Command pulse 카운터의 절대값이 지정한 +/- Limit 값보다 같거나 크게 되면 모션을 자동 정지하도록 합니다.

Positive Limit (Alarm / Limit 탭)

(+) 방향의 Limit 값을 설정하는 항목입니다.

Negative Limit (Alarm / Limit 탭)

(-) 방향의 Limit 값을 설정하는 항목입니다.

EL 신호 적용 방식, Active Low/High (Alarm / Limit 탭)

EL 신호 발생시의 반응 방식과 입력로직을 설정하는 항목입니다. EL 신호 발생시, 즉시 정지할 것인지 감속 후 정지할 것인지를 선택할 수 있으며 입력 신호가 Active Low 인지 Active High 인지 입력로직을 설정할 수 있습니다.

DR 신호 Active Low/High (DR / SD / STA / STP 탭)

+/- DR 신호의 입력 로직을 설정하는 항목입니다. DR 신호는 External Switch 오퍼레이션에서 사용됩니다.

SD 신호 사용, Active Low/High (DR / SD / STA / STP 탭)

SD 신호에 대한 사용여부와 입력로직을 설정하는 항목입니다. SD 신호는 감속의 시작점을 외부에서 입력하는 신호입니다.

SD 감속/감속정지, Latch (DR / SD / STA / STP 탭)

SD 신호 입력시에 반응 방식과 Latch 여부를 설정하는 항목입니다.

STA 신호 사용, 트리거 신호 유형 (DR / SD / STA / STP 탭)

STA 신호에 대한 사용 여부와 트리거 신호 유형을 설정하는 항목입니다. STA 신호는 외부에서 입력되는 모션 시작 신호로서, Low Level 이 되면 모션을 시작하는 트리거 유형과 Falling Edge 신호가 입력되면 모션을 시작하는 트리거 유형을 선택할 수 있습니다.

STP 신호 적용 방식 (DR / SD / STA / STP 탭)

STP 신호에 대한 사용 여부와 반응 방식을 설정하는 항목입니다. STP 입력핀에 Low Level 이 입력될 때, 즉시 정지할 것인지 감속 후 정지할 것인지를 선택할 수 있습니다.

CLR 대상 카운터 지정 (CLR / CMP / INP / LTC 탭)

CLR 신호의 대상 카운터를 설정하는 항목입니다. CLR 신호는 디바이스의 각종 카운터의 카운트값을 리셋합니다.

(주) 커미조아의 모션제어용 디바이스는 축당 네 개씩 카운터가 제공됩니다. 이 중에 CLR 신호에 의해 리셋될 카운트를 지정하는 것은 비트마스크 방법을 이용하는데, 이진수 0000 ~ 1111 까지의 숫자중에서 하나의 값이 설정되었을 때, 자신의 비트에 해당하는 값이 1 인 카운트가 CLR 신호의 대상 카운터로 지정됩니다.

각 비트는 최하위 비트부터 순서대로 Command Position, Feedback Position, Deviation, General 카운터를 의미합니다.

예를 들어, 이진수 '0101' 이(십진수 5) 설정되었을 경우, 최하위 비트인 bit0

1. 디바이스 로드 및 설정

과 세번째 비트인 bit2 이 '1' 이므로 Command Position 카운터와 Deviation 카운터가 CLR 신호의 대상카운터로 지정됩니다.

CLR 신호 유형 (CLR / CMP / INP / LTC 탭)

CLR 신호의 입력 유형을 설정하는 항목입니다.

CMP 펄스 폭 (입력값 * 1.5 us) (CLR / CMP / INP / LTC 탭)

CMP 출력 신호의 펄스폭을 설정하는 항목입니다. 입력되는 값에 1.5 us 이 곱해져서 설정됩니다. 만약, 0 을 입력하였을 경우, CMP 출력 신호의 펄스폭은 Command 출력 신호의 펄스폭과 같도록 설정됩니다.

CMP 신호는 위치 비교 출력 신호입니다.

CMP 신호 출력 로직 (CLR / CMP / INP / LTC 탭)

CMP 출력 신호의 출력 로직을 설정하는 항목입니다.

INP 신호 사용, Active Low/High (CLR / CMP / INP / LTC 탭)

INP 신호의 사용 여부와 입력 로직을 설정하는 항목입니다. INP 기능이 Enable 되면 Command 펄스 출력이 완료되도 INP 입력이 ON 이 되지 않으면 모션이 완료되지 않은 것으로 간주됩니다.

LTC 신호 유형, LTC2 대상 카운터 (CLR / CMP / INP / LTC 탭)

LTC 신호의 입력 로직과 LTC2 의 대상카운터를 설정하는 항목입니다. LTC 신호는 Position Latch 는 특정 순간에 Motion 의 위치 관련 카운터값을 래치(Latch)하여 읽을 수 있도록 하는 기능입니다.

1.3 Interrupt Monitor Config

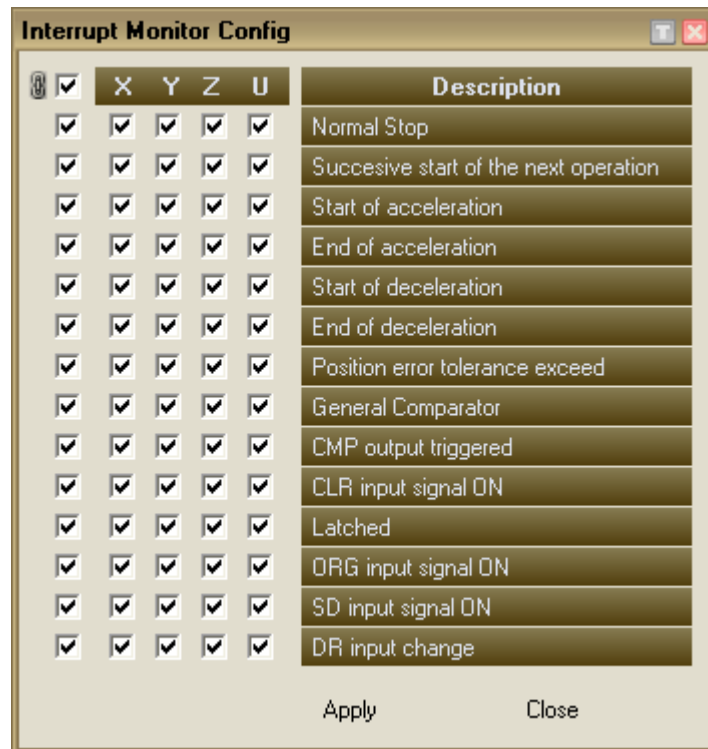
인터럽트는 특정 상황이 발생되었을 때 사용자(또는 프로그래머)에게 이것을 알려주기 위한 것입니다. 윈도우 환경에서는 일반 Application 레벨에서 인터럽트를 처리할 수 없으므로 이벤트를 통하여 Application 에게 인터럽트가 발생하였음을 알려줍니다.

㈜ 커미조아 모션제어용 디바이스의 드라이버 및 라이브러리에는 보다 정교하고 강력한 프로그래밍 지원을 위해 여러가지 상황에 대해 인터럽트를 지정하고 그 이벤트를 발생하도록 설정할 수 있습니다.

COMI-MotionBuilder 에서는 Interrupt Monitor Config 창을 통해 어떠한 인터럽트를 받을 것인지 설정하고, 2.4 절에서 소개될 모니터링 도구인 Interrupt Monitor 창을 통해 실제 발생하는 인터럽트에 대한 로깅을 할 수 있습니다. 자세한 설명은 2.4 절

을 참조하시기 바랍니다.

[그림 1] COMI-MotionBuilder 주 화면 상단의 Config 메뉴에서 ‘Interrupt Monitor Config’ 를 클릭하시면, [그림 4] 와 같은 Interrupt Monitor Config 창이 열립니다.



[그림 4] Interrupt Monitor Config

기본적으로 모든 인터럽트 항목 설정은 축별로 하게 되어 있습니다. 왼쪽의 체크박스를 이용하여 모든 축에 대한 설정을 같이 하실수도 있습니다. 모든 설정을 확인한 후, ‘Apply’ 버튼을 클릭하시면 설정이 적용됩니다. 또한, 이 설정값은 자동으로 저장되어 다음 프로그램 실행시에도 동일한 값이 적용됩니다.

위의 설정은 인터럽트 이벤트를 발생시킬 것인지, 인터럽트 이벤트 없이 공정을 수행할 것인지를 정하는 것입니다. 다시 말해서, ‘Start of acceleration’ 항목의 체크를 없앴다고 해서 가속구간 자체가 없어지는 것이 아니라 가속이 시작되는 시점에서의 이벤트 발생을 안 할 뿐이라는 뜻입니다.

1.4 Source Build Option

COMI-MotionBuilder 는 보다 편리한 프로그래밍 환경과 빠른 개발 시간을 지원하기 위해 샘플 코드 생성기능을 지원합니다. 샘플코드의 언어와 주석 옵션등의 설정을 위한 설정창이나, COMI-MotionBuilder v1.0 beta edition 에서는 지원되지 않습니다.

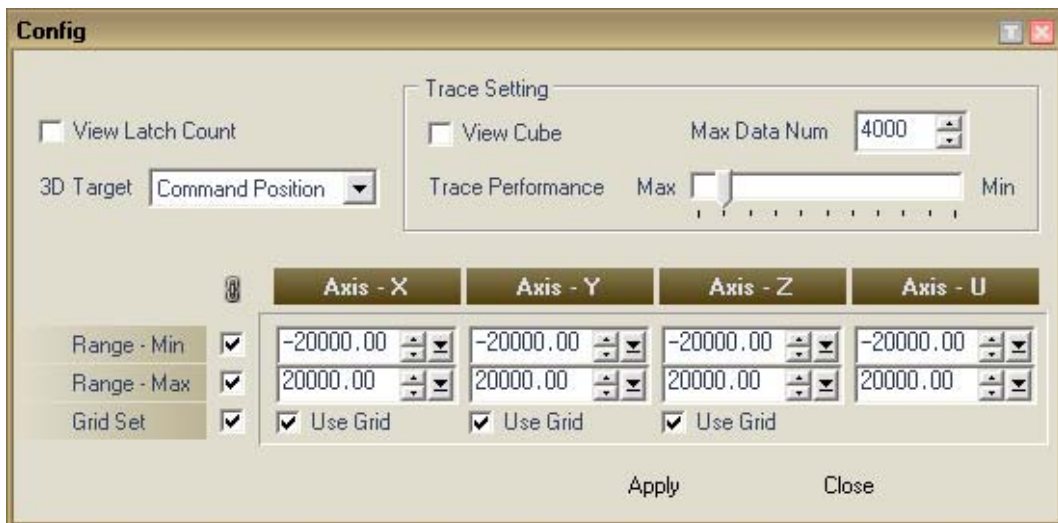
기본적으로 C++ 언어로 설정되어 있으며, 세부 주석을 포함합니다.

샘플 코드 생성을 위해서는 각 창에서 'Generate Code' 를 클릭하시면 됩니다. 윈도우 기본 텍스트 에디터인 메모장으로 열리는 C++ 코드는 텍스트 복사를 하셔서 사용하시던지 파일 자체를 저장하셔서 사용하실 수 있습니다.

1.5 3D Emulator Config

COMI-MotionBuilder 는 3D 그래프를 이용하여 강력한 경로추적 기능을 지원합니다. 이는 구조물 없이도 경로추적과 각종 모니터링이 가능한 가상구동을 통해, 알고리즘 설계시에 적은 리스크로 실험할 수 있도록 지원합니다.

[그림 1] COMI-MotionBuilder 주 화면 상단의 Config 메뉴에서 '3D Emulator Config' 를 클릭하시면, [그림 5] 와 같은 Config 창이 열립니다.



[그림 5] 3D Emulator Config

이 창을 통해 3D Emulator 의 타겟이나 경로추적에 대한 설정, 그래프의 축 설정등

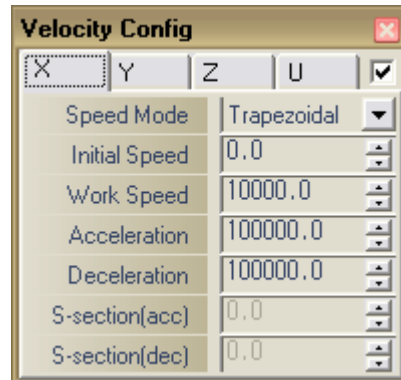
을 할 수 있습니다. 자세한 설명은 2.3 절을 참조하시기 바랍니다.

1.6 Velocity Config

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 'Velocity Config'를 클릭하시면, [그림 6] 와 같은 Velocity Config 창이 열립니다.

이 창을 통하여 각 축별로 속도/가속도를 지정할 수 있습니다.

우측 상단의 체크박스를 통하여 모든 축에 대한 설정을 동일하게 하실 수 있습니다. 반대로, 각 축별로 다른 설정을 하시려면 우측 상단의 체크를 해제하셔야 합니다.



[그림 6] Velocity Config

(※) 커미조아 모션제어용 디바이스는 3 가지의 속도 모드를 지원합니다. Speed Mode 콤보박스를 이용하여 3 가지의 속도 모드 중 하나를 택할 수 있으며, 각각의 속도 모드에 대한 설명은 다음과 같습니다.

Constant speed mode

Constant speed mode에서는 Motion을 수행할 때 가속/감속을 적용하지 않고 일정속도로 Motion을 수행합니다. 여기서 적용되는 일정 속도는 Work Speed에 해당합니다.

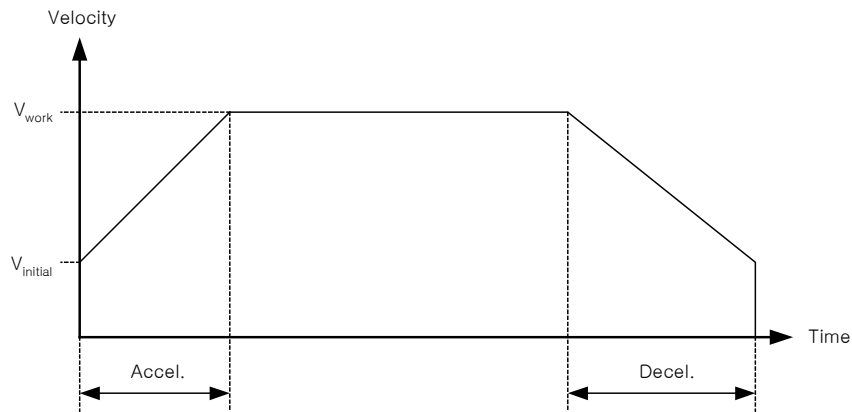
Trapezoidal speed mode

Trapezoidal speed mode에서는 Motion을 수행하는데 있어서 속도의 패턴을 [그림 7]과 같이 Linear acceleration → Working speed(constant) → Linear deceleration의 형태로 운용하는 모드입니다.

타사의 모션제어용 디바이스와는 달리 정지상태에서 가속하는 것과 달리 Initial Speed라는 속도에서 가속하는 것이 특징입니다. 이것은 기구물의 반응속도와 공정

1. 디바이스 로드 및 설정

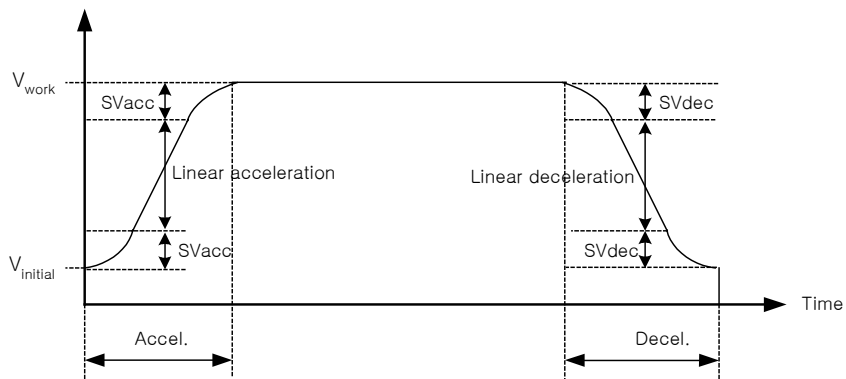
속도를 높여 기구물의 효율성을 극대화합니다.



[그림 7] Trapezoidal speed pattern

S-curve speed mode

S-curve speed mode에서는 Motion을 수행할 때 S자형 형태로 가속과 감속을 수행합니다. S-curve speed mode에서 가(감)속 구간은 [그림 8]과 같이 S-curve section과 Linear acceleration section으로 구성됩니다.



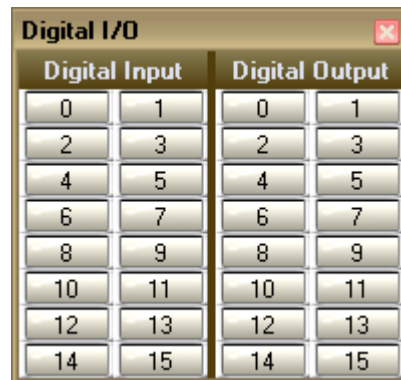
[그림 8] S-curve speed pattern

2 모니터링 도구

COMI-MotionBuilder 은 3D Emulator 를 포함한 강력한 모니터링 도구를 지원합니다. COMI-MotionBuilder 에서 제공하는 모니터링 도구는 COMI-MotionBuilder 에서 내린 제어 명령 뿐만 아니라 사용자가 제작한 별도의 프로그램에서 내린 제어 명령에 의한 모션 결과 또한 모니터링합니다. 이를 이용하여, 사용자는 자체 프로그램 개발시 모든 부분을 프로그래밍하지 않고도 COMI-MotionBuilder 의 모니터링 기능을 이용하여 손쉽게 실험할 수 있습니다.

2.1 Digital I/O

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 ‘Digital I/O’ 를 클릭하시면, [그림 9] 와 같은 Digital I/O 창이 열립니다.



[그림 9] Digital I/O

모션제어용 디바이스의 종류에 따라, 지원되는 Digital I/O 의 채널수는 차이가 있습니다. ‘LX Series Manual -Motion Part- 3. 하드웨어가이드’ 를 참조하시기 바랍니다.

COMI-LX501 등의 디바이스는 특수한 경우로 모터 드라이버 ON 명령 신호를 Digital Output 으로 대신하게 됩니다. 미리 예약되어 있는 서보 ON 신호나 서보 RDY 신호등에 의해 해당 디바이스의 결선에 다른 디바이스와 약간의 차이가 있으므로, 이 점을 숙지하여 결선하시기 바랍니다. 자세한 설명은 ‘LX Series Manual -Motion Part- 3. 하드웨어가이드’ 을 참조하시기 바랍니다.

COMI-MotionBuilder 내의 타이머가 주기적으로 스캔을 하면서, 디지털 신호의 입/출력 상태를 업데이트 합니다. Input 신호가 있을 경우, 해당 채널이 ON 색상으로 변경됩니다. Digital Output 의 경우, 해당 채널을 클릭할 때마다 ON / OFF 상태를 토

2. 모니터링 도구

클해가면서 Output 신호를 내보내게 됩니다.

2.2 MIO Status

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 ‘MIO Status’ 를 클릭하시면, [그림 10] 과 같은 MIO Status 창이 열립니다. 이 창을 통하여 각 축별로 현재 모션과 관련된 여러가지 I/O 상태를 확인할 수 있습니다.



[그림 10] MIO Status

입력 로직에 의해서 ON 이라고 판단되면 ON 색상으로 표시되게 됩니다. 즉, 0V / 5V 등의 신호의 전압 레벨보다는 Active Low / High, Normal Low / High 등의 입력 로직에 의해서 ON 이라고 판단된 경우에 ON 색상으로 표시되게 됩니다. 결국, 올바른 신호를 입력함에도 불구하고 MIO Status 에서 반전되어 표시되는 경우, 입력 로직 설정값을 확인하시기 바랍니다.

각 항목값의 의미는 다음표와 같습니다.

항목값	의미
-----	----

RDY	Servo ready signal input status(1=ON) 단, COMI-LX501에서는 지원되지 않음
ALM	Alarm signal status(1=ON)
+EL	Positive limit switch status(1=ON)
-EL	Negative limit switch status(1=ON)
ORG	Orgin switch status(1=ON)
DIR	Operating direction status(1=ON)
PCS	PCS signal input status(1=ON)
ERC	ERC pin output status(1=ON)
EZ	Index signal status(1=ON)
CLR	Clear input status(1=ON)
Latch	Latch signal input status(1=ON)
SD	Slow Down signal input status(1=ON)
INP	In-Position signal input status(1=ON)
DRP	+DR input signal status(1=ON)
DRN	-DR input signal status(1=ON)
STA	STA input signal status(1=ON)
STP	STP input signal status(1=ON)

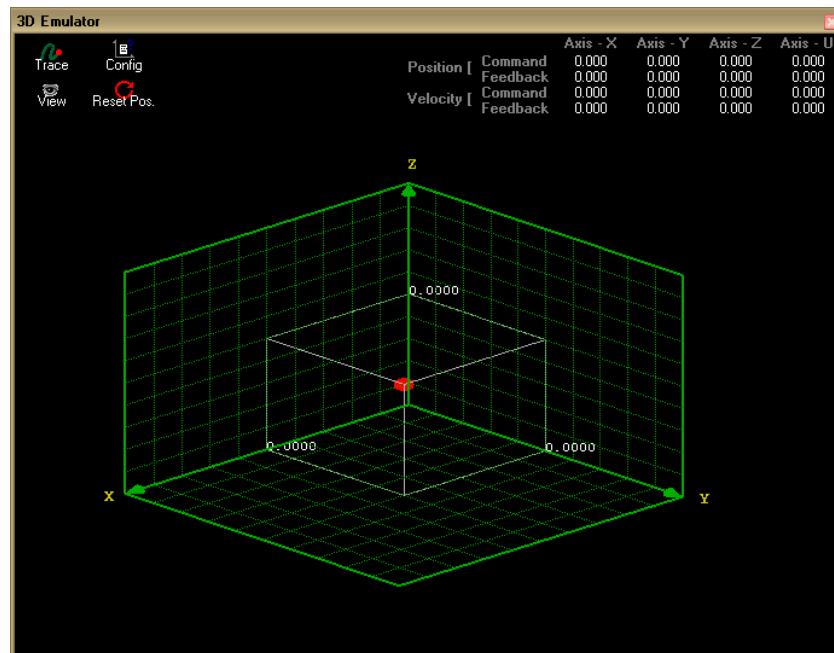
2.3 3D Emulator

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 ‘3D Emulator’를 클릭하시면, [그림 11]과 같은 3D Emulator 창이 열립니다.

오른쪽 상단에는 네 개의 축에 대한 위치/속도값이 숫자로 표현됩니다. 각 축마다 네 개의 항목이 표시되며, Latch 카운터 기능을 체크하였을 경우, 여덟개의 항목이 표시됩니다.

3D 그래프는 x, y, z 축을 기준으로 그려지게 됩니다. 왼쪽 버튼 마우스 드래그를 하면 3D가 회전을 하고, 오른쪽 버튼 마우스 드래그를 하면 3D가 이동을 하게 됩니다. 또한, Ctrl 키를 누른 상태에서 왼쪽 버튼 마우스 드래그를 하면 3D 확대/축소 됩니다.

2. 모니터링 도구



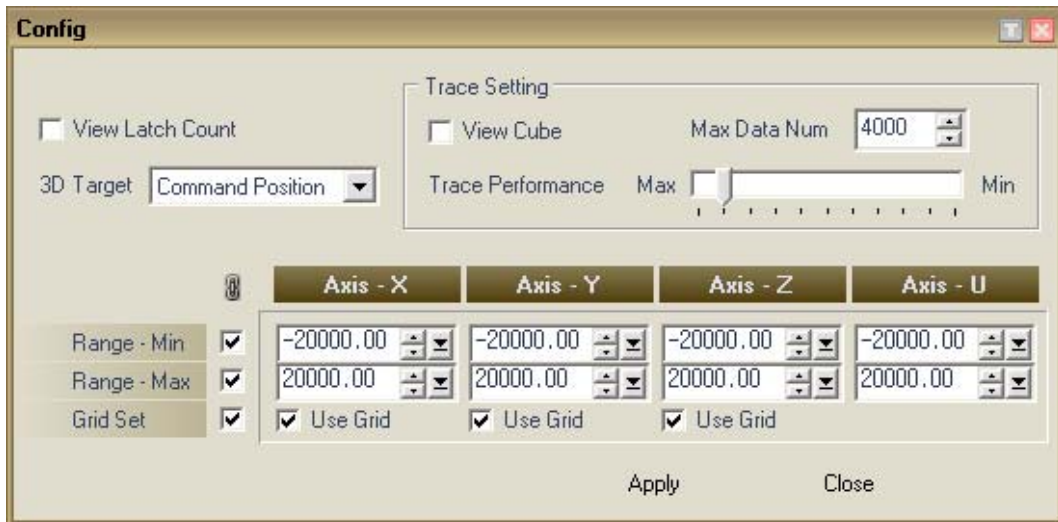
[그림 11] 3D Emulator

Trace

왼쪽 상단의 'Trace' 버튼을 클릭하시면 이동경로에 대해 자취를 남기게 됩니다. 이 때에, 예전에 남겨졌던 자취가 있다면 예전의 자취는 지우고 새로이 자취를 남기기 시작합니다. 다시 'Trace' 버튼을 클릭하시면 토글이 되면서 자취 남기던 것을 중지하게 됩니다.

Config

[그림 1] COMI-MotionBuilder 주 화면 상단의 Config 메뉴에서 '3D Emulator Config' 를 클릭하시거나, 왼쪽 상단의 'Config' 버튼을 클릭하시면 [그림 12] 와 같은 3D Emulator Config 창이 열립니다.



[그림 12] 3D Emulator Config

View Latch Count 를 체크하면, 3D Emulator 의 우측 상단에 래치 카운터값을 포함하여 8 개 항목이 표시됩니다. LATCH 신호가 입력되었을 때 래치되는 카운터는 다음과 같이 4 가지입니다.

- **LTC0** : 명령위치 카운터(Command position counter)
- **LTC1** : 실제위치 카운터(Feedback position counter)
- **LTC2** : 편차 카운터(Deviation counter) 또는 현재의 속도값(PPS)
- **LTC3** : General Counter

이 중 LTC2 카운터는 앞서 설명드린 Init & Config 설정창의 'CLR / CMP / INP / LTC 탭' 에서 대상 카운터를 지정할 수 있습니다.

3D Target 은 3D 그래프를 이용하여, 현위치를 표시하고 자취를 추적할 대상 카운터를 지정하는 콤보박스입니다. Command Position 을 선택하면, 모터 드라이버로 출력되고 있는 지령 펄스를 기준으로 3D Emulator 를 표시합니다. Feedback Position 을 선택하면, 실제로 엔코더를 통해 입력되는 펄스를 기준으로 3D Emulator 를 표시합니다.

Trace Setting 은 자취를 남기기위해 'Trace' 버튼을 토글 시킨 상태에서의 옵션입니다. View Cube 는 현위치를 표시하는 빨간 상자외에 현재 좌표를 안내해 주기 위한 정육면체 회색 안내선을 보이게 할 것인지를 설정합니다. Max Data Num 은 자취데이터갯수를 의미합니다. 이 개수가 많아지면 표시되는 자취의 길이가 길어집니다. 너무 많은 수를 입력하면 컴퓨터 사양에 따라서 부하가 걸릴 수 있으니, 적당한 값을 설정하시기 바랍니다. Trace Performance 는 max 로 갈수록 보다 세밀한 자취가 그려

2. 모니터링 도구

지는 반면 시스템 부하가 많아지고 min 으로 갈수록 시스템 부하가 적어지는 반면 자취의 세밀함은 적어집니다.

Range - max/min 을 이용하여 3D 그래프 축의 범위를 지정할 수 있습니다. COMI-MotionBuilder v1.0 beta edition 에서 U 축에 대한 범위는 무시됩니다. Grid 는 3D 그래프 공간을 나타내는 녹색 안내선을 뜻합니다.

View

3D 그래프를 어느 방향에서 바라볼 것인지를 선택합니다. 기본적으로 X-Y-Z 축을 입체적으로 바라보는 관점에서 X-Y 평면, X-Z 평면, Y-Z 평면을 바로 위에서 바라보는 관점을 택할 수 있습니다. 이 버튼을 클릭하여 바라보는 관점을 바꾸면 확대/축소나 이동한 값이 초기치로 리셋됩니다.

Reset Pos

카운터값을 0 으로 리셋합니다. 모든 축을 0 으로 리셋하거나 각 축을 선택하여 리셋하도록 선택하는 팝업 메뉴가 띄워집니다. 사용자는 팝업 메뉴를 클릭함으로써, 카운터를 리셋할 수 있는데, 이 버튼은 3D Emulator 에서 표시되는 수치만이 아니라 디바이스 카운터 자체를 리셋합니다.

2.4 Interrupt Monitor

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 ‘Interrupt Monitor’를 클릭하시면, [그림 13] 과 같은 Interrupt Monitor 창이 열립니다.

이 창을 이용하여 Interrupt 이벤트 발생을 시작하고 Interrupt 이벤트를 검사하여 로깅할 수 있습니다.



[그림 13] Interrupt Monitor

왼쪽 상단의 체크박스를 클릭하여 체크표시를 하면 Interrupt Monitor 가 Enable 되며 이벤트 발생을 시작합니다. Interrupt 이벤트가 발생되면 발생한 시간과 발생한 축별로 로그가 남게 됩니다. 이러한 로그는 오른쪽 상단의 지우개 모양의 버튼을 클릭하면 클리어됩니다.

어떠한 Interrupt 이벤트를 발생시킬 것인지는 앞서 설명드린 Interrupt Monitor Config 설정창에서 설정가능합니다. 하지만 예러에 관계된 인터럽트인 경우, 즉, 다음과 같은 인터럽트는 Interrupt Monitor Config 와 상관없이 발생하며 Interrupt Monitor 를 통해서 로깅됩니다.

인터럽트 발생 조건
Stop by +SL(Software limit) stop motion
Stop by -SL(Software limit) stop motion
Stop by General Comparator stop motion
+EL signal is turning ON stop motion
-EL signal is turning ON stop motion
ALM signal is turning ON stop motion
SD signal turning ON after deceleration
Abnormal operation data stop motion
PA/PB input buffer counter overflows
In-position counter counts beyond the range at the time of interpolation

3 모션 오퍼레이션

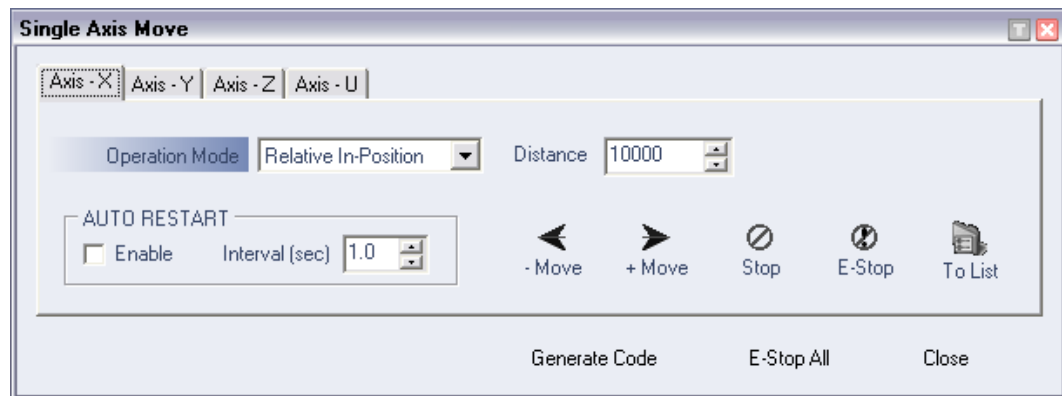
(주) 커미조아의 모션제어용 디바이스는 크게 8 가지 종류의 오퍼레이션을 지원합니다. COMI-MotionBuilder 는 이러한 기능을 각각의 폼에 나누어 구동할 수 있도록 하였습니다. 이러한 오퍼레이션은 서로 독립적으로 구동할 수도 있고 서로 연결하여 복합적으로 구동할 수도 있습니다.

모터드라이버가 ON 상태이며 ¹⁾, Alarm 이나 Limit 신호등의 제약이 없는 경우, 이번 절에서 설명될 오퍼레이션의 명령을 통해 실제로 모터가 동작하게 됩니다.

모든 오퍼레이션 창에는 Generate Code 버튼이 있습니다. 이 버튼들은 해당 오퍼레이션에 대해, 설정된 값을 기준으로 샘플코드를 생성하여 줍니다. 사용자는 이 코드를 복사 또는 저장하여 프로그램 개발에 이용하실 수 있습니다.

3.1 Single Axis Move

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 ‘Single Axis Move’ 를 클릭하시면, [그림 14] 와 같은 오퍼레이션 창이 열립니다.



[그림 14] Single Axis Move

이 창을 통해 x, y, z, u 네 축에 대해 각각 독립구동 명령을 내릴수 있습니다. 탭을 이용하여 구동 명령을 내릴 축에 대한 탭시트로 이동한 후, 여러 설정을 해주고 +/- Move 버튼을 클릭하면, 원하는 방향으로 대상축이 구동됩니다. 이 때, 네 축은 서로 완전 독립 상태로서, 다른 축의 구동 상태가 현재 대상축의 구동 상태에 영향을 미치지 않습니다. 즉, x 축에 Move 명령을 준 상태에서 y 축에 Move 명령을 주어

¹⁾ COMI-MotionBuilder 이 실행되면 기본적으로 디바이스를 로딩하고, 모터 드라이버에게 Servo ON 출력 신호를 보내내어 모터 드라이버를 ON 상태로 만듭니다.

도 상관 없으며, Move 의 종료 시점 또한 서로 영향을 받지 않습니다.

Operation Mode

Single Axis Move 오퍼레이션에서 지정할 수 있는 작동 모드에는 Velocity Move, Relative In-Position, Absolute In-Position 세 가지가 있습니다.

Velocity Move 는 일정한 속도를 지정해 주고 Move 명령을 내리면 정지 명령을 내리기 전까지 계속 지정된 속도로 구동되는 모드입니다. 구동 속도는 앞서 설명드린 Velocity Config 창을 통해 축별로 설정하실 수 있습니다.

Relative In-Position 모드는 현재 위치에서 지정된 상대 좌표값까지의 이동을 하는 모드입니다. Relative In-Position 모드를 선택하시면 Distance 항목이 생기며 이 항목에 지정된 값만큼 이동하게 됩니다.

Absolute In-Position 모드는 지정된 절대 좌표값까지의 이동을 하는 모드입니다. Absolute In-Position 모드를 선택하시면 Position1, Position2 항목이 생깁니다. - Move 버튼을 클릭하시면 Position1 항목에, +Move 버튼을 클릭하시면 Position2 항목에 지정된 절대좌표까지 이동하게 됩니다.

Auto Restart¹⁾

이동이 끝났을 경우, 지정된 Interval 만큼 멈춰있다가 자동으로 구동을 재시작해주는 기능입니다. 지정된 작동모드에 따라 적용되는 데에 약간의 차이가 있습니다.

Velocity Move 모드에서는 이 설정이 무시됩니다.

Relative In-Position 모드에서는 지정된 거리만큼을 자동으로 왕복하게 됩니다.

Absolute In-Position 모드에서는 Position1 항목과 Position2 항목에 지정된 절대 좌표 사이를 자동으로 왕복하게 됩니다.

Stop / E-Stop / E-Stop All

해당축이 작동중인 상태에서 강제적으로 정지를 시키는 버튼들입니다.

Stop 버튼은 일반적인 정지로서, Velocity Config 창을 통해 설정된 속도모드에 따라 감속 후 정지하는 것을 뜻합니다.

E-Stop 버튼은 긴급한 상황에 정지 명령을 내리는 것으로서, 감속없이 즉시 정지하는 것을 뜻합니다. E-Stop 버튼은 해당축에 대해서만 즉시정지합니다.

¹⁾ COMI-MotionBuilder 프로그램 자체에서 지원하는 기능입니다. 모션 제어 디바이스의 드라이버나 라이브러리에는 일정기간 멈춰있다가 자동으로 재시작하는 기능이 없습니다.

3. 모션 오퍼레이션

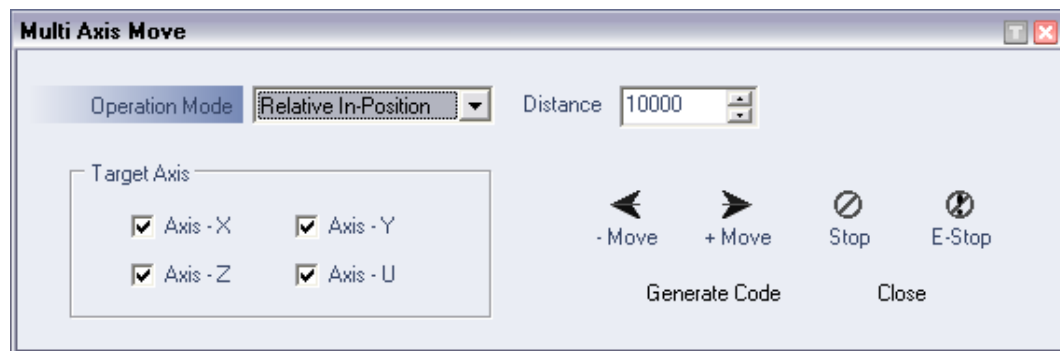
E-Stop All 버튼은 E-Stop 버튼처럼 감속없이 즉시 정지하는 점은 동일하나, E-Stop 버튼이 해당축만 즉시정지하는 것과는 달리 모든축을 즉시 정지합니다.

To List

현재의 Single Axis Move 구동 정보를 Listed Motion 오퍼레이션 창으로 보내주는 기능의 버튼입니다. 자세한 설명은 3.8 절을 참조하시기 바랍니다.

3.2 Multi Axis Move

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 'Multi Axis Move' 를 클릭하시면, [그림 15] 와 같은 오퍼레이션 창이 열립니다.



[그림 15] Multi Axis Move

이 창을 통해 여러축의 동시 구동 명령을 내릴 수 있습니다. Single Axis Move 오퍼레이션과 다른점은 여러축의 구동을 동시에 시작한다는 점입니다. Single Axis Move의 경우, 각 축마다 시작 명령을 내리면 약간의 딜레이가 발생합니다. 하지만, Multi Axis Move의 경우 각 축마다의 구동 정보를 미리 설정해놓고 동시에 클릭 신호를 주어 구동을 시작하므로 딜레이가 전혀 없습니다.

Single Aixs Move의 경우, 해당축의 지정명령이 완료되면 모션이 종료된 것으로 간주하지만, Multi Axis Move의 경우, 모든 축이 지정명령을 완료해야 모션이 종료된 것으로 간주합니다.

Operation Move

Multi Axis Move 오퍼레이션에서 지정할 수 있는 작동 모드에는 Velocity Move, Relative In-Position, Absolute In-Position 세 가지가 있습니다.

Velocity Move는 일정한 속도를 지정해 주고 Move 명령을 내리면 정지 명령을 내리

기 전까지 계속 지정된 속도로 구동되는 모드입니다. 구동 속도는 앞서 설명드린 Velocity Config 창을 통해 축별로 설정하실 수 있습니다.

Relative In-Position 모드는 현재 위치에서 지정된 상대 좌표값까지의 이동을 하는 모드입니다. Relative In-Position 모드를 선택하시면 Distance 항목이 생기며 이 항목에 지정된 값만큼 이동하게 됩니다. COMI-MotionBuilder 프로그램에서는 모든 축에 대해 동일한 상대 좌표값을 지정하게 되어 있으나, 실제 프로그램 개발시에는 축마다 다른 상대 좌표값을 지정하실 수 있습니다.

Absolute In-Position 모드는 지정된 절대 좌표값까지의 이동을 하는 모드입니다. Absolute In-Position 모드를 선택하시면 Position1, Position2 항목이 생깁니다. - Move 버튼을 클릭하시면 Position1 항목에, +Move 버튼을 클릭하시면 Position2 항목에 지정된 절대좌표까지 이동하게 됩니다. COMI-MotionBuilder 프로그램에서는 모든 축에 대해 동일한 절대 좌표값을 지정하게 되어 있으나, 실제 프로그램 개발시에는 축마다 다른 절대 좌표값을 지정하실 수 있습니다.

3.3 Interpolation

Multi Axis Move 오퍼레이션의 경우, 대상축들의 구동시작은 동시에 이루어지나, 각 축마다의 구동 완료시점은 각 축의 설정 및 환경에 따라 다르게 됩니다. 또한, 구동 시작 시점만 동일할 뿐 Single Axis Move 오퍼레이션의 경우와 마찬가지로 각 축은 서로 독립적이며, 다른 축의 구동에 영향을 받지 않습니다.

이에 반해, 두 개 이상의 축이 서로 연동되어져서 구동되는 것을 Interpolation 오퍼레이션이라 합니다. ㈜ 커미조아의 모션 제어 디바이스는 Linear Interpolation, Circular Interpolation, Helical Interpolation, Spline Interpolation 4 가지 종류의 보간 기능을 지원합니다.

COMI-MotionBuilder 프로그램에서는 Interpolation 오퍼레이션 창을 이용하여 이러한 네 가지 기능을 테스트해 볼 수 있습니다. [그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 'Interpolation' 을 클릭하시면, [그림 16] 과 같은 오퍼레이션 창이 열립니다. 기본적으로 Linear Interpolation 이 선택되어 있는 상태입니다. 탭을 이용하여, Circular, Helical, Spline 을 선택하시면 이에 해당하는 Interpolation 기능도 테스트해 보실 수 있습니다.

Interpolation 오퍼레이션 창은 다른 오퍼레이션 창과는 달리 속도 모드를 설정하는 항목이 있습니다. 왜냐하면, 다른 창들은 Velocity Config 창을 통하여 설정된 축별 속도모드를 사용하지만, Interpolation 의 경우 여러 축이 연동되어 구동되므로 이동 속도 또한 동일한 속도모드 안에서 서로 연동되어야 하기 때문입니다.

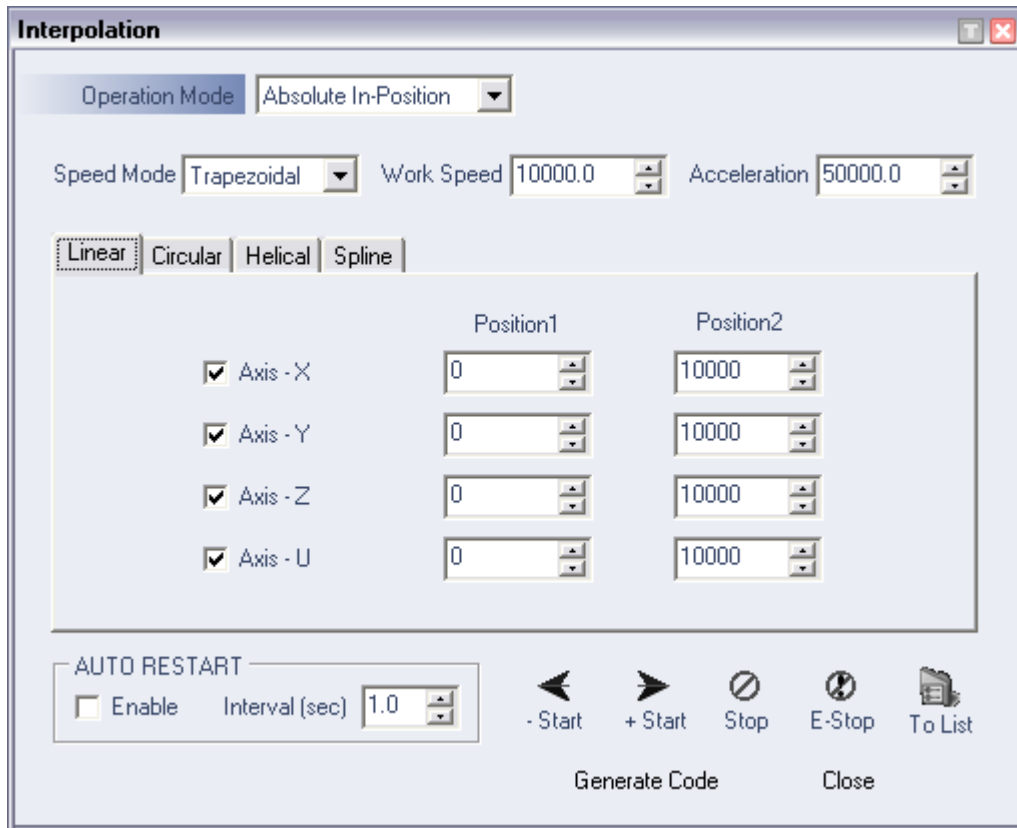
Linear Interpolation



3. 모션 오퍼레이션

2 축이상이 연동되어 직선보간 구동이 이루어지는 것을 Linear Interpolation 이라 합니다. ㈜ 커미조아의 모션 제어 디바이스는 2 ~ 4 축의 Linear Interpolation 을 지원합니다. Linear Interpolation 의 대상이 되는 축은 2 축 이상이며 채널번호가 연속적이지 않아도 됩니다.

Interpolation 오퍼레이션 창을 열거나, Linear 탭을 선택하였을 경우 [그림 16] 과 같이 Linear Interpolation 기능 테스트가 가능한 상태가 됩니다.



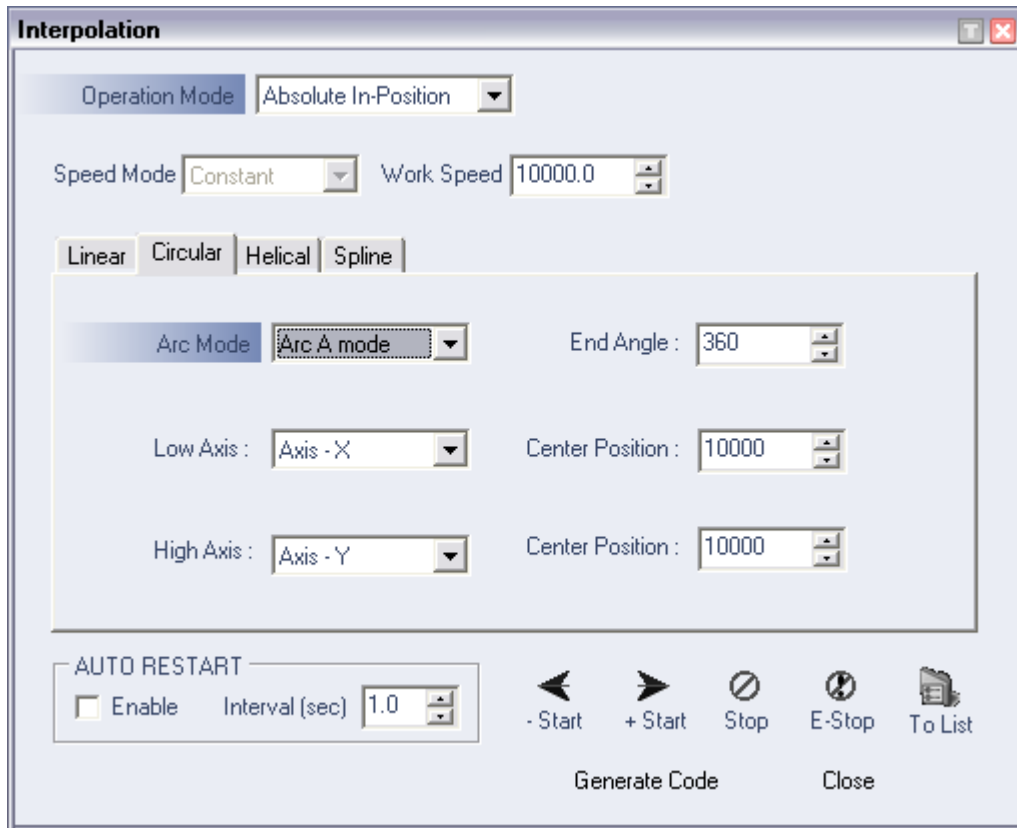
[그림 16] Linear Interpolation

Linear Interpolation 오퍼레이션에서 지정할 수 있는 작동 모드에는 Relative In-Position, Absolute In-Position 두 가지가 있습니다. 모드에 따라 상대좌표나 절대좌표 항목을 적절히 설정하신 후, Start 버튼을 클릭하시면 구동이 시작됩니다. Single Axis Move 오퍼레이션의 경우와 동일하게 AUTO RESTART 를 이용하시면 설정된 Interval 간격에 맞추어 왕복운동이 가능합니다.

Circular Interpolation

2 축이 연동되어 원호보간 구동이 이루어지는 것을 Circular Interpolation 이라 합니다. Circular Interpolation의 대상이 되는 축은 2 축이며 채널번호가 연속적이지 않아도 됩니다. 개념상 대상축중에 채널번호가 낮은 축을 x 축, 채널번호가 높은 축을 y 축으로 정하여, 구동방향이나 구동목표점등을 설정합니다.

Interpolation 오퍼레이션 창에서 Circular 탭을 선택하였을 경우 [그림 17] 와 같이 Circular Interpolation 기능 테스트가 가능한 상태가 됩니다.



[그림 17] Circular Interpolation

원호보간의 특성상 속도모드는 Constant 모드만이 지정가능합니다.

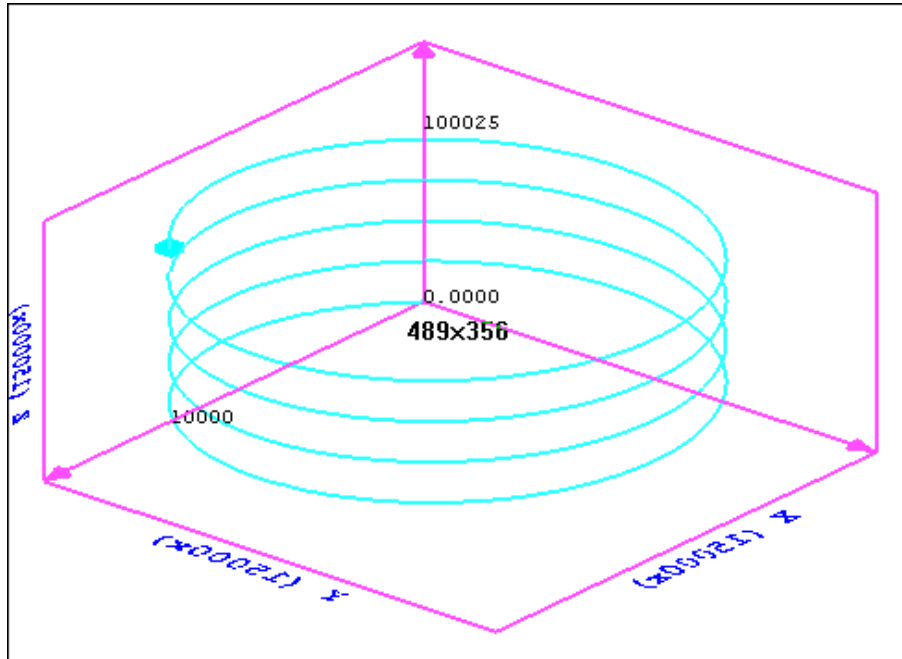
구동될 원호의 경로를 지정하는 방법은 크게 두가지로 나뉩니다. 각도를 이용하여 원호의 방향을 지정하는 ArcA mode 와 원호의 끝 종료점 좌표와 구동방향을 지정하는 ArcP mode 가 있는데, 이는 Arc Mode 콤보박스에서 선택하실 수 있습니다. 자세한 설명은 'LX Series Manual -Motion Part- 5.4.4 Coordinated Motion' 을 참조하시기 바랍니다.

3. 모션 오퍼레이션

Center Position 이나 Center Offset 은 현재 좌표로부터 원호보간의 중심점까지의 X/Y 축의 좌표값을 뜻하며, End Angle 은 원호보간할 각도, End Position 이나 End Distance 는 원호보간 끝 종료점의 좌표값을 뜻합니다.

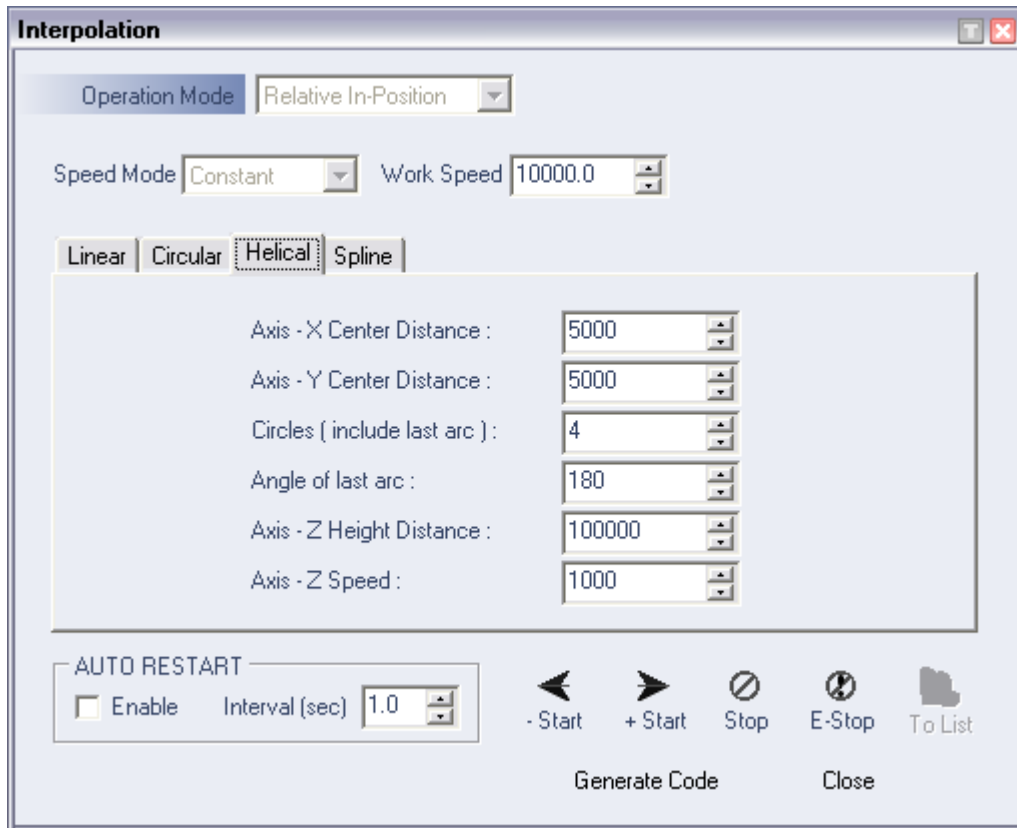
Helical Interpolation

Helical Interpolation 은 아래 그림과 같이 2 축 원호 보간과 1 축 직선 보간이 복합적으로 구동되는 기능입니다. 이것은 나사산의 가공등을 생각하시면 됩니다.



원호 보간 대상 2 축과 직선 보간 대상 1 축을 선택하시면, 개념상 원호 보간 대상축 중에 채널번호가 낮은 축을 x 축, 채널번호가 높은 축을 y 축, 직선 보간 대상 축을 z 축으로 정하여, 구동방향이나 구동목표점등을 설정합니다. COMI-MotionBuilder 프로그램의 경우, 채널 번호가 낮은 순서대로 x, y, z 축을 정하였으나 실제 프로그램 개발시에는 임의로 원호 보간 대상 2 축과 직선 보간 대상 1 축을 선택하실 수 있습니다.

Interpolation 오퍼레이션 창에서 Helical 탭을 선택하였을 경우 [그림 18] 과 같이 Helical Interpolation 기능 테스트가 가능한 상태가 됩니다.



[그림 18] Helical Interpolation

헬리컬보간의 특성상 작동모드는 Relative In-Position 모드만이 지정 가능하며, 속도모드는 Constant 모드만이 지정가능합니다.

Axis - X/Y Center Distance 항목은 현재 좌표로부터 원호보간의 중심점까지의 X/Y 축의 상대 좌표값을 뜻합니다.

Circles 항목은 마지막 ARC 를 포함하여 전체 구동 중에 포함되어야할 원의 수를 뜻합니다. 이 항목에서 지정한 수의 원이 그려지면 선형보간 대상 축에 지정한 거리에 도달하지 못하였더라도 모션은 완료됩니다.

Angle of last arc 항목은 마지막 ARC 의 각도입니다.

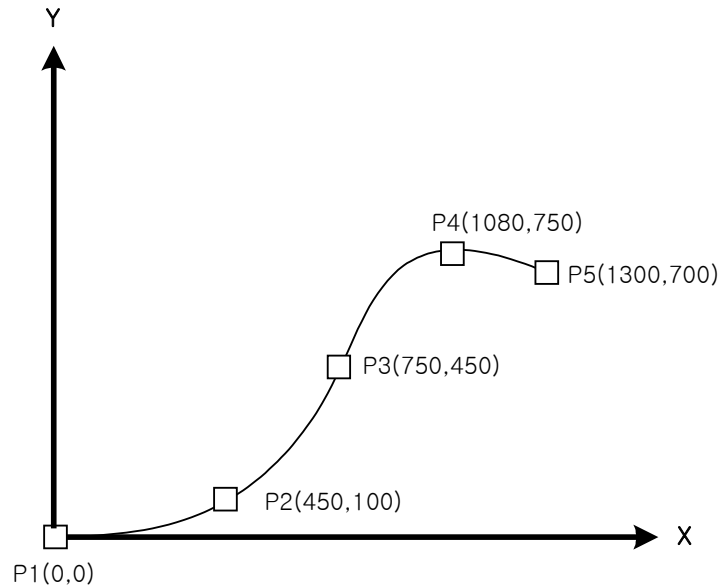
Axis - Z Height Distance 항목은 z 축이 이동해야할 거리를 상대좌표로 지정합니다. 지정한 수의 원이 그려지지 않았더라도 z 축이 지정한 거리에 도달하면 모션은 종료됩니다.

3. 모션 오퍼레이션

Axis - Z speed 항목은 z 축에 의한 직선 구동 속도입니다. 상단의 Work Speed는 헬리컬 보간중 x, y 축에 의한 원호보간의 속도를 뜻함에 유의하시기 바랍니다.

Spline Interpolation

Spline Interpolation은 아래 그림과 같이 사용자가 지정하는 데이터 포인트를 자유곡선으로 보간해주는 기능입니다. ㈜ 커미조아의 모션 제어 디바이스는 Cubic Spline Interpolation 기능을 지원하며, 2 축 뿐만 아니라 3 개 이상의 다수의 축 또한 보간이 가능하여 입체적인 자유곡선 보간이 가능합니다.



Interpolation 오퍼레이션 창에서 Spline 탭을 선택하시면, [그림 19]와 같이 Spline 데이터 포인트를 입력할 수 있는 상태가 됩니다. Spline 보간의 특성상 작동 모드는 Absolute In-Position 모드만이 지정 가능하며, 속도모드는 다른 모드도 가능하나 Constant 모드를 권장합니다.

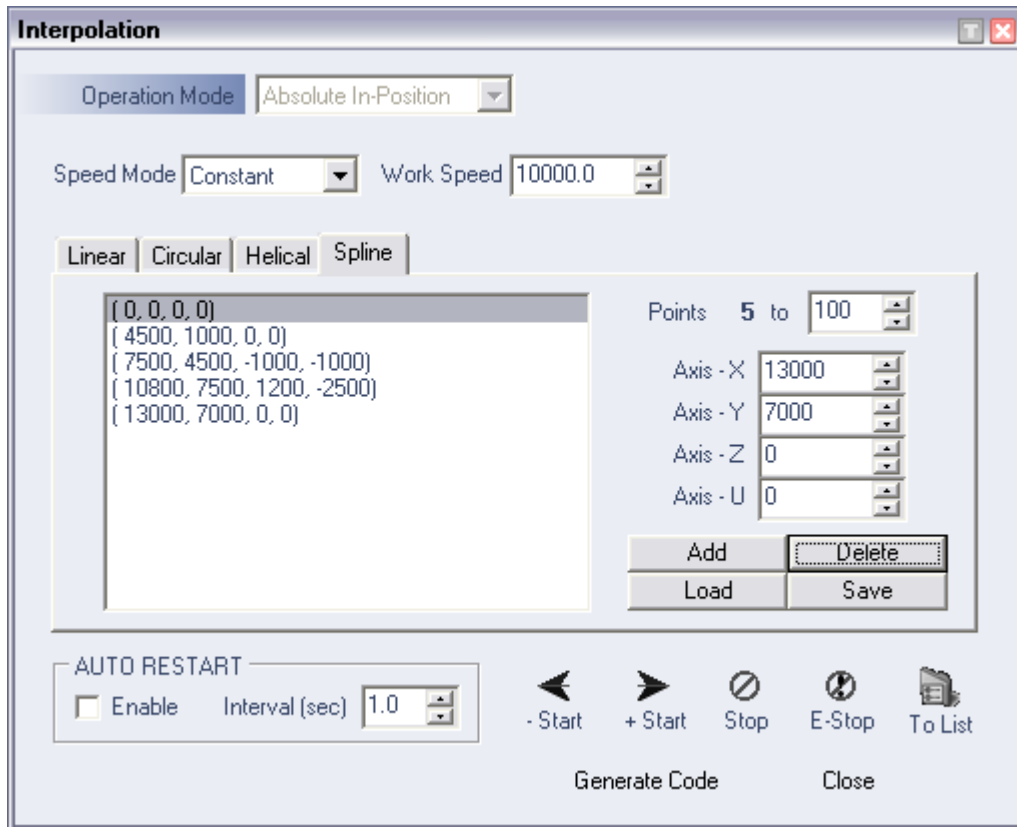
[그림 19] Spline Operation 창에는 'Points X to Y' 항목이 있습니다. X에는 현재 입력되어 있는 데이터 포인트 수가 표시됩니다. Y는 자유곡선의 정밀도라 할 수 있습니다. 전체 스플라인 보간의 이동 경로를 몇 개로 나누어서 처리할 것인지를 설정하는 항목입니다. 스플라인 보간은 모든 경로에 대해서 완전 곡선을 그리는 것이 아닙니다. 경로를 잘게 나누어서 그 사이 사이 구간은 직선운동을 하게 됩니다.

'Points X to Y' 항목 밑에 데이터 포인트를 입력하는 항목이 있습니다. 해당되는 축에 대해서 좌표를 입력하신 후, 'Add'를 클릭하시면 됩니다. Spline 보간 대상이 아닌 축들은 현재의 좌표값을 그대로 입력해 주시면 됩니다. 추가한 데이터 포인트

트들은 왼쪽에 리스트로 표시되어 집니다. 원하지 않는 데이터 포인트의 경우, 그 데이터 포인트 항목을 선택하신 후, Delete' 버튼을 클릭하시면 됩니다.

현재 입력되어 있는 데이터 포인트들은 'Save' 버튼 클릭으로 '*.lst' 라는 파일에 저장해 놓을 수 있습니다. 미리 저장해 놓은 데이터 포인트들은 'Load' 버튼 클릭으로 불러올 수 있습니다.

'To List' 버튼을 이용하여 Listed Motion 오퍼레이션 창으로 데이터를 넘겨줄 수 있습니다. 자세한 설명은 3.8 절을 참조하시기 바랍니다.

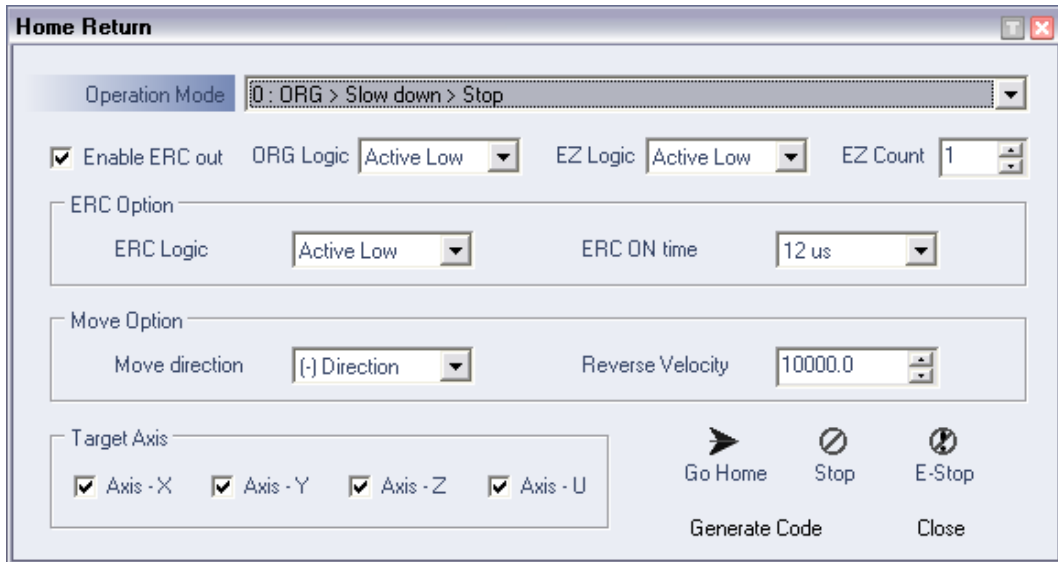


[그림 19] Spline Interpolation

3.4 Home Return

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 'Home Return'을 클릭하시면, [그림 20] 과 같은 오퍼레이션 창이 열립니다.

3. 모션 오퍼레이션



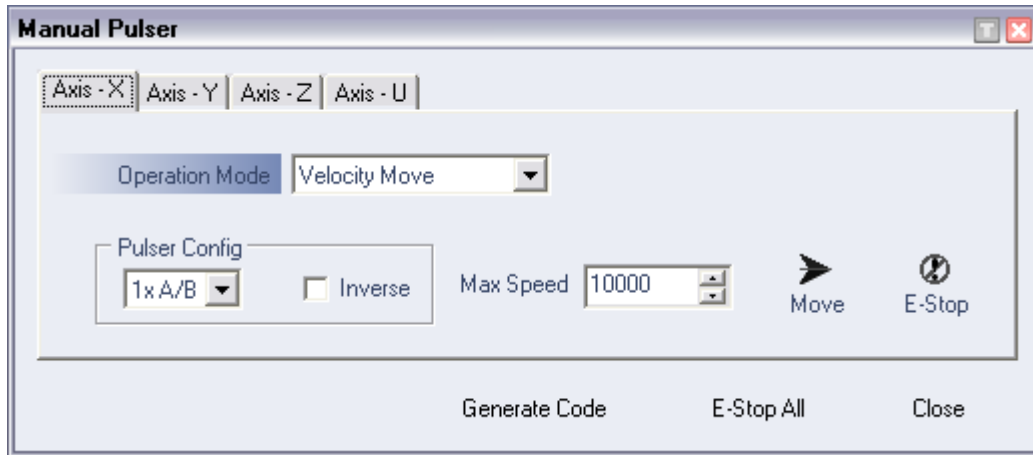
[그림 20] Home Return

이 창을 통해 축별로 원점 복귀 명령을 내릴 수 있습니다. (주) 커미조아의 모션 제어 디바이스는 13 가지의 원점 복귀 모드를 지원합니다. 또한, 많은 옵션을 통해 효율적이고 정확한 원점 복귀 기능을 지원합니다. 각 항목별 기능 및 자세한 사항은 ‘LX Series Manual -Motion Part- 5.4.6 원점 복귀(Home return) 함수’를 참조하시기 바랍니다.

3.5 Manual Pulser

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 ‘Manual Pulser’를 클릭하시면, [그림 21]과 같은 오퍼레이션 창이 열립니다.

Manual Pulser 오퍼레이션은 로터리 엔코더(Rotary Encoder)와 같은 장치를 이용하여 수동으로 모션을 제어하는 것을 의미합니다. 로터리 엔코더의 출력을 Manual Pulser의 입력단에 연결하여, 엔코더의 움직임에 맞추어 모션을 제어할 수 있습니다. 또한, 엔코더의 움직임을 그대로 따라하는 Velocity Move 외에 Home Move, Relative In-Position, Absolute In-Position 등의 작동모드를 추가적으로 지원하여, 원점복귀나 In-Position 구동에도 응용이 가능합니다.



[그림 21] Manual Pulser

Pulser Config 를 통하여, 입력되는 PA, PB 신호의 특성을 지정할 수 있습니다. 특히, 엔코더 입력인 1 채배 (1x A/B), 2 채배 (2x A/B), 4 채배 (4x A/B) 설정 외에 CW/CCW 모드를 이용하면, 다른 축의 모션을 그대로 수행하도록 설정할 수 있습니다.

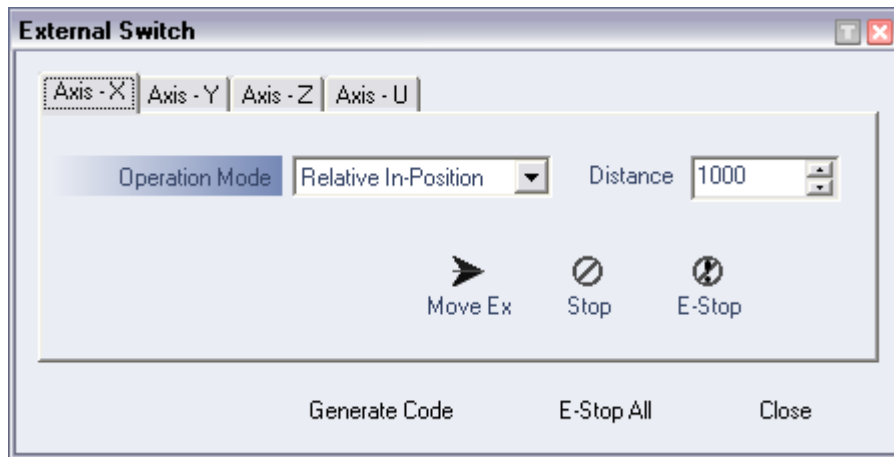
Max Speed 는 해당축의 최대 속도를 의미합니다. 기본적으로 Manual Pulser 오퍼레이션의 모션 속도는 PA, PB 입력신호의 주파수에 의해 결정됩니다. 이 때에, Max Speed 를 통하여 최대 속도를 제한하게 됩니다.

3.6 External Switch

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 'External Switch' 를 클릭하시면, [그림 22] 와 같은 오퍼레이션 창이 열립니다.

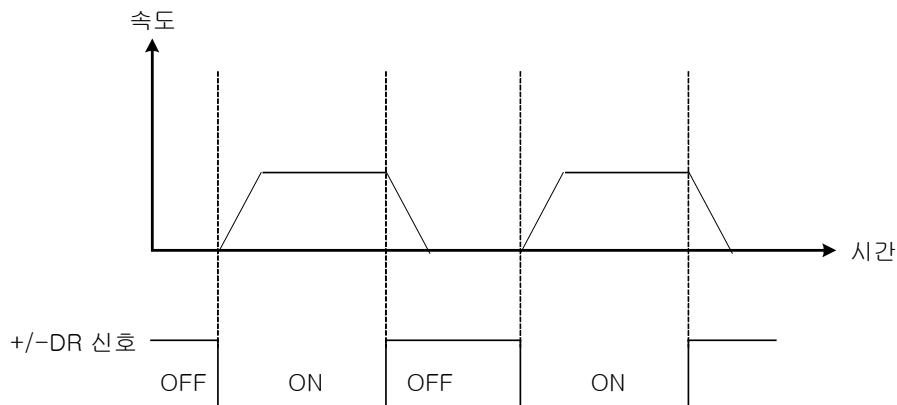
External Switch 오퍼레이션은 +DR 과 -DR 입력신호로 모션을 제어합니다. Velocity Move 와 Relative In-Position 중에서 작동모드를 선택한 후, 'Move Ex' 버튼을 통해 오퍼레이션 명령을 내리면, +DR 또는 -DR 신호가 ON 상태인 경우에만 모션을 구동하고 OFF 인 상태에서는 모션을 정지합니다. 자세한 설명은 'LX Series Manual - Motion Part- 5.4.8 External Switch Operation 모드 모션 제어 함수' 을 참조하시기 바랍니다.

3. 모션 오퍼레이션



[그림 22] External Switch

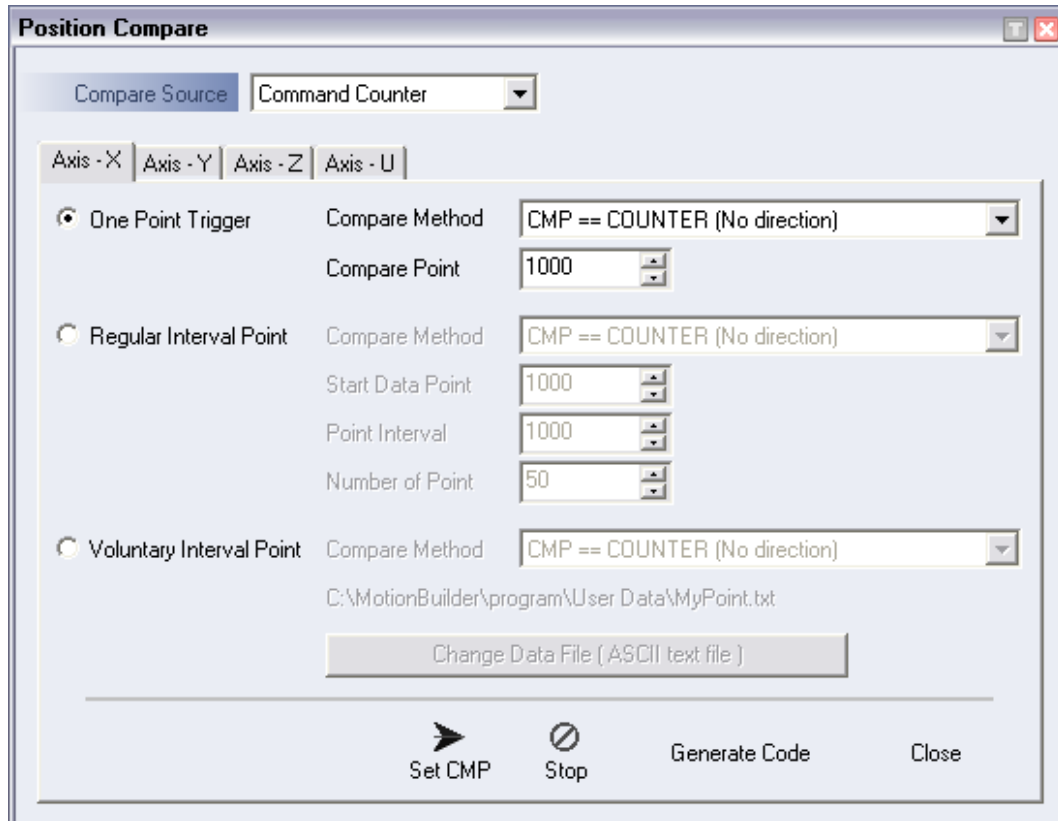
+DR 또는 -DR 신호와 모션 구동의 관계는 다음 그림과 같습니다.



3.7 Position Compare

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 'Position Compare' 를 클릭하시면, [그림 23] 과 같은 오퍼레이션 창이 열립니다.

Position Compare 오퍼레이션은 Command counter 또는 Position counter 등의 카운트값이 사용자가 지정한 조건에 만족되면 CMP 출력핀을 통하여 트리거 펄스를 출력해주는 기능입니다. 이 기능을 사용하면 모션을 구동하면서 연속적으로 원하는 위치에서 외부기기에 하드웨어적인 트리거 신호를 제공할 수 있습니다.



[그림 23] Position Compare

Compare Source 항목을 통해 비교하고자 하는 대상카운터를 선택하실 수 있습니다.

Position Compare 오퍼레이션창에서 CMP 트리거 신호를 출력할 위치를 지정하는 방법에는 크게 세 가지가 있으며, 이는 라디오 버튼을 통해 선택할 수 있습니다.

One Point Trigger

하나의 위치만을 지정하고자 할 때에는 'One Point Trigger'를 선택하시면 됩니다. Compare Method를 통해 CMP 출력 조건을 선택할 수 있습니다. Compare Point 값과 Compare Source 항목을 통해 선택된 대상카운터값 사이의 관계가 Compare Method에 의해 선택된 조건을 만족하면 CMP 신호가 출력됩니다.

Regular Interval Point

여러 위치값이 일정한 간격일 때에는 'Regular Interval Point'를 선택하시면 됩니다. 비교할 처음 위치값과 위치값들의 간격 그리고, 비교할 위치의 개수를 통하여 여러 위치값을 쉽게 설정하실 수 있습니다.

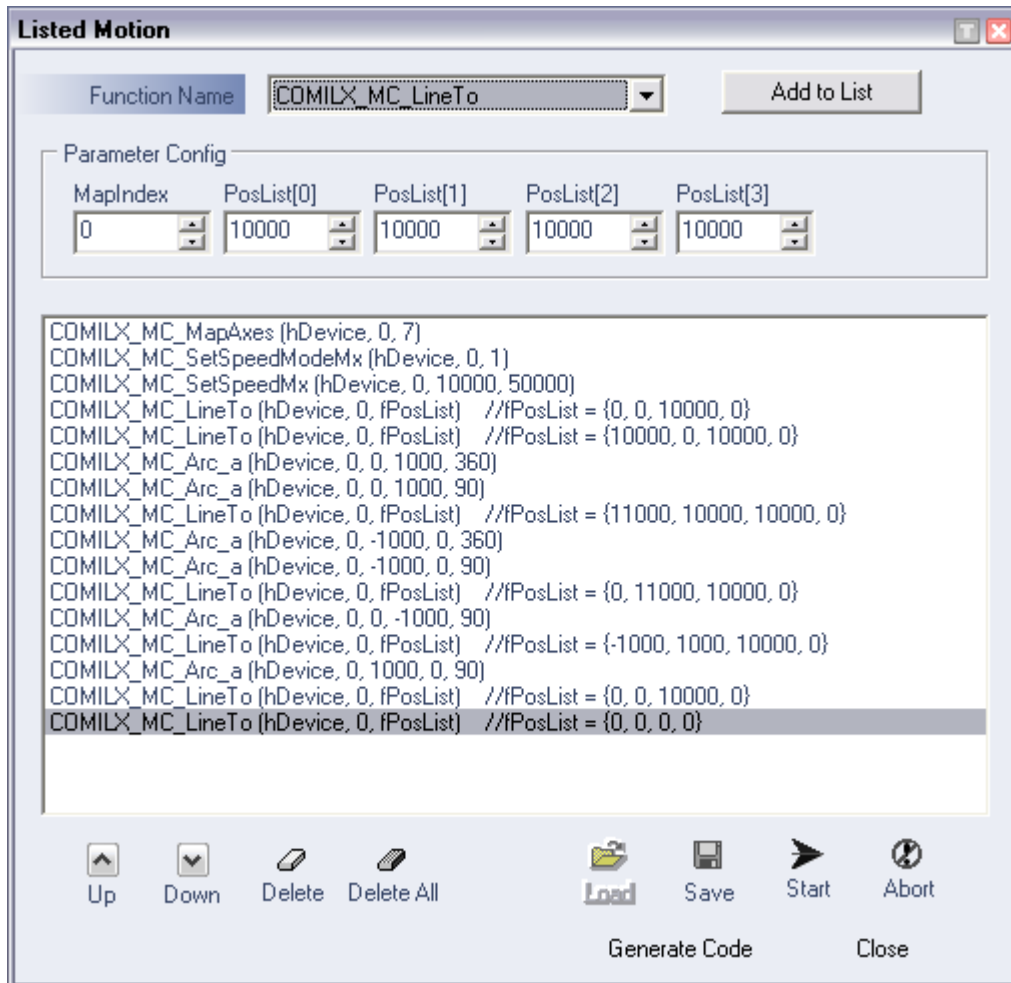
3. 모션 오퍼레이션

Voluntary Interval Point

여러 위치값이 임의의 간격일 때는 ‘Voluntary Interval Point’ 를 선택하시면 됩니다. 메모장등의 에디터를 이용하여, 원하는 위치값들을 텍스트로 저장하신 후, ‘Change Data File’ 을 클릭하여 저장한 파일명을 지정하여 주시면 됩니다.

3.8 Listed Motion

[그림 1] COMI-MotionBuilder 주 화면 상단의 Operation 메뉴에서 ‘Listed Motion’ 을 클릭하시면, [그림 24] 와 같은 오퍼레이션 창이 열립니다.



[그림 24] Listed Motion

Listed Motion 오퍼레이션은 여러 작업을 리스트에 등록시킨 후에 일괄적으로 처리하는 기능입니다. 하나의 작업과 그 다음 작업간의 지연시간없이 연속적인 작업을 할 수 있으며, 작업속도의 연속성을 확보할 수 있습니다.

‘Function Name’ 항목에서 원하는 작업을 선택하시면, 이 작업에 필요한 옵션들이 ‘Parameter Config’ 에 나타납니다. 적절히 설정하신 후, ‘Add List’ 버튼을 클릭하시면 리스트에 추가됩니다.

Up/Down 버튼을 통해 작업의 순서를 변경하실 수 있습니다. 또한 Delete/Delete All 버튼을 통해 작업을 리스트에서 삭제하실 수 있습니다.

리스트에 등록이 끝나면 ‘Start’ 버튼을 이용하여 Listed Motion 오퍼레이션을 시작하고, ‘Abort’ 버튼을 이용하여 강제적으로 종료할 수 있습니다.¹⁾

Load/Save 버튼을 통해 리스트에 등록된 작업들을 파일에 저장하거나, 가져올 수 있습니다.

¹⁾ ‘Abort’ 버튼을 이용하여 정지 명령을 내리지 않고, 다른 창의 ‘Stop’ 버튼등을 이용하여 정지명령을 내리면, Listed Motion 자체가 종료되는 것이 아니라, 현재 구동중인 해당 작업만이 종료됩니다.

4 부가기능과 사용자 팁

이번 장에서는 앞서 설명되지 않은 COMI-MotionBuilder 프로그램의 모션 관련 부가기능과 사용자 편의를 위해 추가된 GUI 부가기능에 대해 설명합니다.

추후에 Trouble Shooting 등의 사용자 팁등이 4.3 절에 추가될 예정입니다. (주) 커미조아에 고객등록하시면, COMI-MotionBuilder 프로그램의 업그레이드와 매뉴얼의 추가/수정본에 대한 지원을 받으실 수 있습니다.

4.1 모션관련 부가기능

Servo ON/OFF

(주) 커미조아의 모션 제어디바이스는 Servo ON 출력 단자를 제공합니다.¹⁾ 이러한 출력단자를 통하여 서보를 ON/OFF 할 수 있습니다.

[그림 1] COMI-MotionBuilder 주 화면 상단의 Main 메뉴의 Servo ON/OFF 의 하위 메뉴를 이용해서서, Servo ON 신호의 출력상태를 토글하실 수 있습니다. 또한, 'Servo ON/OFF' 툴바를 이용하여, 출력상태의 토글 및 모니터링이 가능합니다. Servo ON 신호가 나가지 않는 상태에서는 'OFF' 가 표시되고, Servo ON 신호가 나가고 있는 경우에는 'ON' 이 표시됩니다. 더불어, 'ON' 상태에서 모터 드라이버로부터 알람 신호가 들어오는 경우에는 'FAULT' 가 표시되고, 현재 대상축이 모션을 수행중일 경우에는 'RUN' 이 표시됩니다. 이러한 LED 는 왼쪽부터 x, y, z, u 축을 각각 의미합니다. 해당 축 LED 를 클릭하시면 Servo ON 신호의 출력상태가 토글됩니다.



기본적으로, COMI-MotionBuilder 프로그램이 처음 시작될 때와 'Init&Config' 창의 'Apply' 버튼을 클릭하였을 경우, 자동으로 네 축 모두에 Servo ON 출력신호를 내보내도록 제작되어 있습니다.

Device Reset

[그림 1] COMI-MotionBuilder 주 화면 상단의 Main 메뉴의 Device Reset 을 클릭하시면, 디바이스를 리셋하실 수 있습니다.¹⁾ 이 때에는 모션 제어 디바이스의 모든

¹⁾ COMI-LX501 의 경우, Servo ON/OFF 출력단자가 없습니다. Digital Output 출력 단자를 이용하실 수 있습니다.

¹⁾ Standard 툴바를 이용하여 Device Reset 을 실행하실 수도 있습니다.

상태가 초기치로 리셋됩니다. 각종 설정과 출력 상태, 카운터를 포함한 모든 것이 리셋됩니다. 단, Device Reset 을 하였다고 해서 모터 드라이버의 상태까지 리셋되는 것은 아닙니다.

Emg Stop All

[그림 1] COMI-MotionBuilder 주 화면 상단의 Main 메뉴의 Emg Stop All 을 클릭하면, 모든 축을 감속없이 정지시킬 수 있습니다.²⁾ 수행중이던 Interpolation 오퍼레이션이나 Listed Motion 오퍼레이션이 있다면 그 오퍼레이션 또한 중지합니다.

4.2 GUI 부가기능

Load / Save Layout Profile

COMI-MotionBuilder 프로그램은 사용자가 환경에 맞게 레이아웃을 자유로이 조절할 수 있습니다. 또한, 이러한 레이아웃을 상황에 따라 각각의 프로파일로 저장하여 사용할 수 있습니다. [그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴의 Load / Save Layout Profile 을 이용하여 현재 보이는 레이아웃을 *.lay 라는 파일에 저장하거나, 이미 저장되어 있는 *.lay 파일로부터 프로파일을 읽어 화면 레이아웃을 이에 맞추어 배치하실 수 있습니다.

Skin Mode

COMI-MotionBuilder 프로그램은 그 자체가 독립적으로 사용될 수도 있지만, 사용자 응용 프로그램과 동시에 실행되는 것을 기본으로 하여 제작되었습니다. 사용자는 모션 제어와 모니터링에 대한 모든 부분을 코딩하지 않고도, 꼭 필요한 부분만 코딩한 후, COMI-MotionBuilder 와 연동하여 디버깅함으로써, 보다 빠르고 쉬운 프로그램 개발을 할 수 있습니다.

다른 프로그램과 같이 실행될 때를 대비하여, COMI-MotionBuilder 프로그램이 차지하는 화면의 공간을 적절히 조절하기 위해 만들어진 것이 Skin Mode 입니다. [그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴의 Skin Mode 를 이용하여 현재의 레이아웃을 스킨모드로 변경하실 수 있습니다. 스킨모드로 변경하시면, 메인 폼이 사라지고 메인 폼에 종속되었던 모니터링 도구 창과 Velocity Config 창이 독립적으로 이동하게 됩니다.

스킨모드를 해제하시려면, 모니터링 도구 창과 Velocity Config 창 중에 하나라도 닫기 버튼을 클릭하시면 됩니다.

Always Stay Top

²⁾ Standard 툴바를 이용하여 Emg Stop All 을 실행하실 수도 있습니다.

4. 부가기능과 사용자 팁

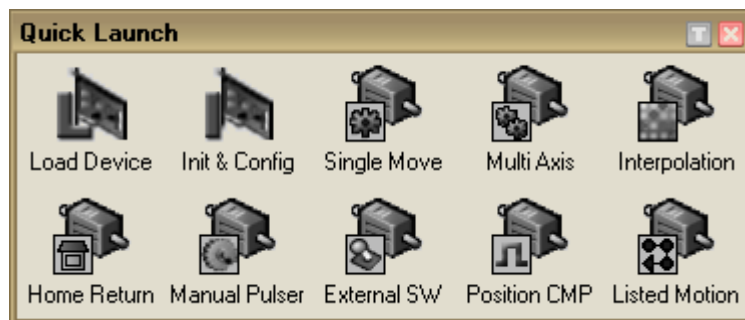
Always Stay Top 기능을 이용하면, 원하는 창이 다른 창에 의해 가려지지 않도록 설정하실 수 있습니다.



각 폼의 오른쪽 상단에 위치한 'T' 표시를 클릭하면 그레이스케일의 버튼이 칼라로 바뀌면서 Always Stay Top 모드로 설정됩니다. 이렇게 설정된 창은 다른 창이나 다른 프로그램에 의해 가려지지 않습니다. 하지만, Always Stay Top 을 여러 창에 설정하시면 먼저 설정된 창은 다른창에 가려질 수 있음을 유의하시기 바랍니다.

Quick Launch

[그림 1] COMI-MotionBuilder 주 화면 상단의 View 메뉴에서 'Quick Launch' 을 클릭하시면, [그림 25] 와 같은 창이 열립니다.



[그림 25] Quick Launch

창을 편리하게 띄울 수 있는 아이콘들이 정렬되어 창입니다. 사용자의 화면 상황에 맞게 크기나 위치를 변경하실 수 있습니다. 변경된 가로, 세로비에 따라 아이콘이 재 정렬되며, 이러한 변경은 자동으로 저장되어, 다음 실행시에도 반영됩니다.