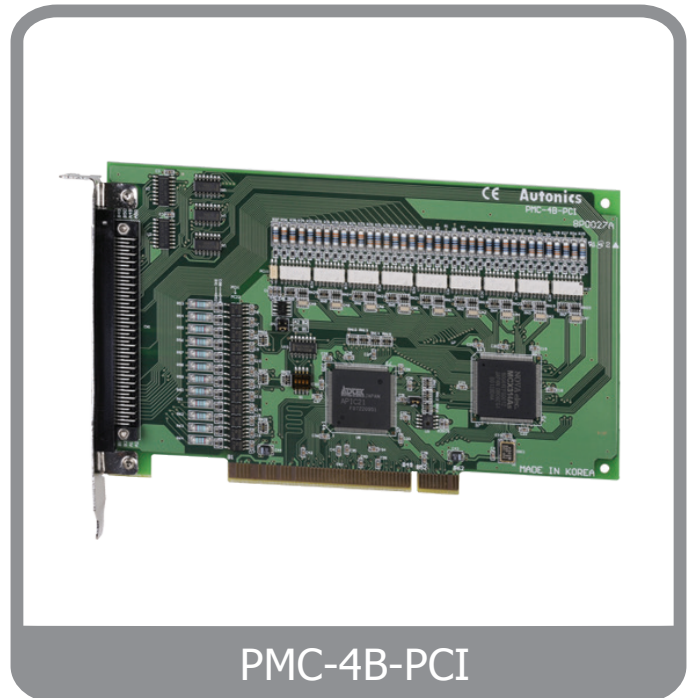


Motion Controller

# PMC-4B-PCI

사용자 매뉴얼

CE K



PMC-4B-PCI



# 제품 구입 감사 안내문

(주)오토닉스 제품을 구입해 주셔서 감사합니다.





먼저 안전을 위한 주의사항을 반드시 읽고 제품을 올바르게 사용하십시오.

본 설명서는 제품에 대한 안내와 바른 사용 방법에 대한 내용을 담고 있으므로 사용자가 쉽게 찾아 볼 수 있는 장소에 보관하여 주십시오.

# 사용자 매뉴얼 안내



- 사용자 매뉴얼의 내용을 충분히 숙지한 후에 제품을 사용하십시오.
- 사용자 매뉴얼은 제품 기능에 대해 자세하게 설명한 것으로, 사용자 매뉴얼 이외의 내용에 대해서는 보증하지 않습니다.
- 사용자 매뉴얼의 일부 또는 전부를 무단으로 편집 또는 복사하여 사용할 수 없습니다.
- 사용자 매뉴얼은 제품과 함께 제공하지 않습니다.  
당사 홈페이지([www.autonics.co.kr](http://www.autonics.co.kr))에서 다운로드 하여 사용하십시오.
- 사용자 매뉴얼의 내용은 해당 제품의 성능 및 소프트웨어 개선에 따라 사전 예고없이 변경될 수 있으면, 업그레이드 공지는 당사 홈페이지를 통해 제공해 드립니다.
- 당사에서는 사용자 매뉴얼의 내용을 조금 더 쉽게, 정확하게 작성하고자 많은 노력을 기울였습니다. 그럼에도 불구하고 수정해야 할 부분이나 질문사항이 있으시면 당사 홈페이지를 통하여 의견을 주시기 바랍니다.

# 사용자 매뉴얼의 공통 기호

기호	설명
 <b>Note</b>	해당 기능에 대한 보충 설명
 <b>Warning</b>	지시 사항을 위반할 경우 심각한 상해나 사망 사고의 위험이 있는 내용
 <b>Caution</b>	지시 사항을 위반할 경우 경미한 상해나 제품 손상이 발생할 수 있는 내용
 <b>Ex.</b>	해당 기능에 대한 예시
※1	주석 설명 표시

# 안전을 위한 주의사항

- '안전을 위한 주의사항'은 제품을 안전하고 올바르게 사용하여 사고나 위험을 미리 막기 위한 것이므로 반드시 지키십시오.
- 주의사항은 '경고'와 '주의'의 두 가지로 구분되어 있으며 '경고'와 '주의'의 의미는 다음과 같습니다.

 <b>Warning</b>	<b>경고</b>	지시 사항을 위반하였을 때, 심각한 상해나 사망 사고가 발생할 가능성이 있는 경우
 <b>Caution</b>	<b>주의</b>	지시 사항을 위반하였을 때, 경미한 상해나 제품 손상이 발생할 가능성이 있는 경우

## Warning

- 인명이나 재산상에 영향이 큰 기기(예: 원자력 제어장치, 의료기기, 선박, 차량, 철도, 항공기, 연소장치, 안전장치, 방범/방재장치 등)에 사용할 경우 반드시 2중으로 안전장치를 부착한 후 사용하십시오.  
화재, 인사사고, 재산상의 막대한 손실이 발생할 수 있습니다.
- 취급설명서에 기재된 일반 사양의 환경에서 사용하십시오. 부식성 가스, 인화성 가스가 있는 장소, 고온, 다습, 진동, 화재, 오동작, 제품의 손상 또는 열화의 원인이 됩니다.  
화재, 인사사고, 재산상의 막대한 손실이 발생할 수 있습니다.
- 제품을 개조하지 마십시오.  
화재, 인사사고, 재산상의 막대한 손실이 발생할 수 있습니다.
- 운전 중에 전원을 차단하지 마십시오.  
인명사고, 재산상의 손실, 오작동의 원인이 됩니다.
- 운전시에는 항상 비상 정지가 가능하도록 하십시오.  
장치 파손 및 인명 사고의 우려가 있습니다.
- 운전 중에 커넥터 및 점퍼핀을 분리하지 마십시오.  
인명사고, 재산상의 손실, 오작동의 원인이 됩니다.
- 제품의 폐기 시에는 산업 폐기물로서 처리하십시오.

## Caution

- 전원이 인가된 상태에서 결선 및 점검, 보수를 하지 마십시오.  
감전 및 오동작의 원인이 됩니다.

- 당사 기술자 이외에는 제품을 수리하지 마십시오.  
감전이나 화재의 우려가 있습니다. 수리가 필요할 시에는 당사로 문의하십시오.
- 반드시 정격/성능 범위에서 사용하십시오.  
제품의 수명이 짧아지는 원인이 되며 화재의 우려가 있습니다.
- 청소 시 물, 유기용제를 사용하지 마십시오.  
감전, 화재, 제품의 손상의 우려가 있습니다.
- 본 제품의 내부로 먼지나 배선 찌꺼기가 유입되지 않도록 주의하십시오.  
감전, 화재, 제품의 손상의 원인이 됩니다.

### (1) 취급 시 주의사항

- 모션 컨트롤러 구동 전 주의사항
  - ① 사용환경에 적절하게 위치 좌표 및 각종 파라미터들을 설정한 후 모션 컨트롤러를 구동시키십시오.
  - ② 조그 혹은 연속 모드를 사용 시, 시스템의 속도를 서서히 증가시키면서 적절한 구동속도를 설정하십시오.
- ID Select S/W 입력 시 주의사항
  - ① 본 제품을 한 PC 에 여러장 사용할 때, 다른 보드와 스위치 설정을 다르게 하여 사용하십시오.
  - ② 총 16 개의 보드를 동시 사용 가능합니다.
- 본 제품은 아래의 환경조건에서 사용할 수 있습니다.
  - ① 실내 사용
  - ② 오염 등급 II(Pollution Degree II)
  - ③ 고도 2000m 이하
  - ④ 설치 카테고리 II(Installation Category II)

### (2) 제품의 보관



#### Warning

본 제품을 사용 후, 보관시에는 PC 와 I/O 케이블을 분리하여 정전기 방지용 포장지로 포장하여 정격 및 사양에서 표기된 온도 및 습도 이내에서 보관하십시오.

### (3) PC 로의 설치

기판의 Edge connector 를 PC 의 PCI bus connector 에 정확하게 삽입하여, 취부 금속부를 나사로 조이십시오.

설치 작업은 반드시 PC 의 전원을 차단한 후에 실행하십시오.



#### Warning

- 반드시 PCI bus connector 에 장착하여 사용하십시오.  
장치 파손 및 화재나 감전, 부상의 우려가 있습니다.



**Caution**

- 전원 입력은 반드시 절연 트랜스를 장착하여 사용하십시오.  
화재나 감전, 부상의 우려가 있습니다.

**(4) 입/출력 신호의 접속**

외부 전원이나 입력/출력 신호와의 접속에 있어서 극성을 반대로 하거나 정격범위를 넘는 전압/전류를 가하면 회로 소자를 파괴하거나 동작의 신뢰성을 저하시키는 원인이 됩니다. 충분히 배선을 확인한 후에 접속하십시오.



**Warning**

- 접속은 반드시 접속도를 기초로 해서 배선하십시오.  
화재나 감전 및 제품 파손의 우려가 있습니다.
- 외부 전원의 이상, 컨트롤러의 고장이라도, 반드시 시스템 전체가 안전하도록 컨트롤러의 외부에 안전 보호 장치를 설치하십시오.  
화재나 감전 및 제품 파손의 우려가 있습니다.
- 리미트(한계) 스위치를 반드시 설치하십시오.  
인명사고, 재산상의 손실이 발생할 수 있습니다.
- 긴급 정지 스위치를 반드시 설치하십시오.  
인명사고, 재산상의 손실이 발생할 수 있습니다.



**Caution**

- 설치 및 배선 작업 등을 할 때에는 반드시 전원을 차단하고 설치하십시오.  
감전, 제품 손상의 우려가 있습니다.
- 설치 및 배선 작업 등을 할 때에는 배선끼리 서로 단락되지 않도록 주의하십시오.  
감전, 제품 손상의 우려가 있습니다.
- 사용하지 않는 단자는 아무것도 결선하지 말고 다른 단자와 단락되지 않도록 주의하십시오.  
감전, 제품 손상의 우려가 있습니다.

**(5) 특수 용어 설명**

- active (Active):  
어떤 신호에 있어, 그 신호가 갖는 기능이 유효한 상태에 있는 것
- drive (Drive)  
Pulse 열 입력의 서보 모터 또는 스테핑 모터의 driver 구동 디바이스에 대하여, 모터를 회전시키기 위한 pulse 를 출력하는 동작



- 정량 pulse drive  
지정된 pulse 양 만큼 pulse 를 출력하는 drive
- 연속 pulse drive  
정지신호가 active 될 때 까지 drive pulse 출력을 계속 출력하는 drive
- CW  
시계 방향 (clockwise 의 약어)
- CCW  
반시계 방향 (counter clockwise 의 약어)
- 보간 노드  
연속 보간을 구성하는 하나 하나의 보간 drive
- 가가속도  
단위 시간당의 가속도/감속도의 증가/감소율
- 2 의 보수  
2 진수에 있어서 부의 값의 표현방법

**Ex.**

16bit 길이의 data 로는, -1 은 FFFFh, -2 는 FFFEh, -3 은 FFFDh, .... -32768 은 8000h 로 표현합니다.

### (6) 특수문자 기호 설명

- n○○○○: X, Y, Z, U 의 각 축의 신호명을 n○○○○ 라고 기술하고 있습니다.  
이 때, "n" 는 X, Y, Z 및 U 를 나타냅니다.
- ↑: 신호가 Low level 에서 High level 로 변화할 때의 상승 에지
- ↓: 신호가 High level 에서 Low level 로 변화할 때의 하강 에지
- nPP, nPM: nPP 는 nP+P, nP+N 을 nPM 은 nP-P, nP-N 을 나타냅니다.
- nECA, nECB, nECZ: EC 는 엔코더를 나타내며 A, B, C 는 출력상을 나타냅니다.  
레지스터 설명 시 nIN2 는 nECZ 를 나타냅니다.
- 기능의 제한  
본 보드에서는 편차 카운터 클리어 기능, BUSYN, SCLK 은 지원하지 않습니다.

### (7) 제품의 구성

본 모션 컨트롤러 PMC-4B-PCI 는 아래와 같이 구성되어 있습니다.

- PMC-4B-PCI Board
- I/O 케이블
- 취급설명서

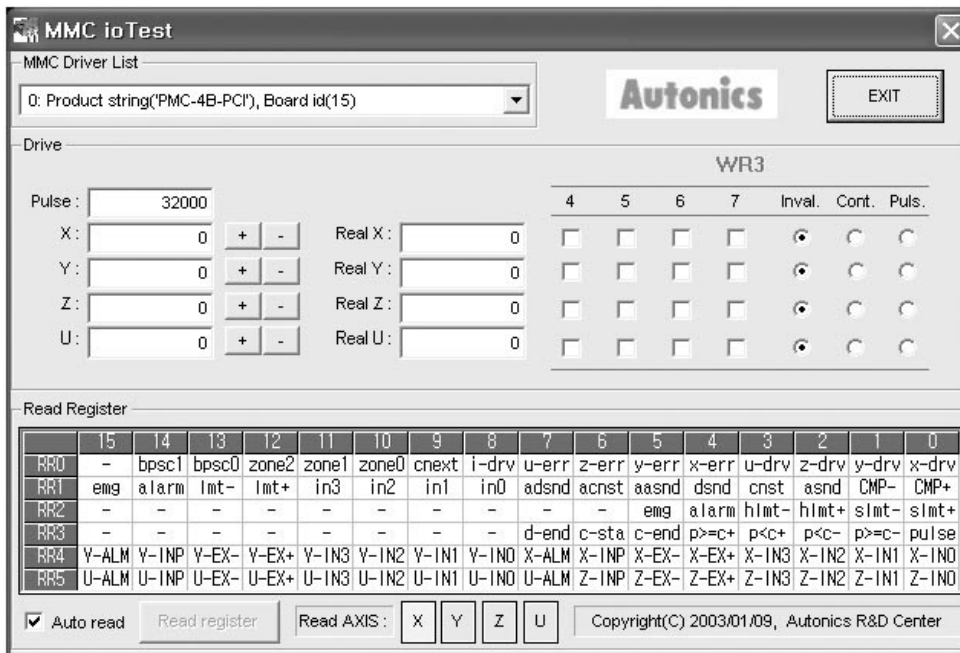
**(8) 홈페이지 지원 소프트웨어**

- 윈도우 드라이버(Windows7 32bit, 64bit 지원)
- Labview 라이브러리 및 상세도움말
- C 언어 라이브러리 및 예제(라이브러리 설명서 제공)

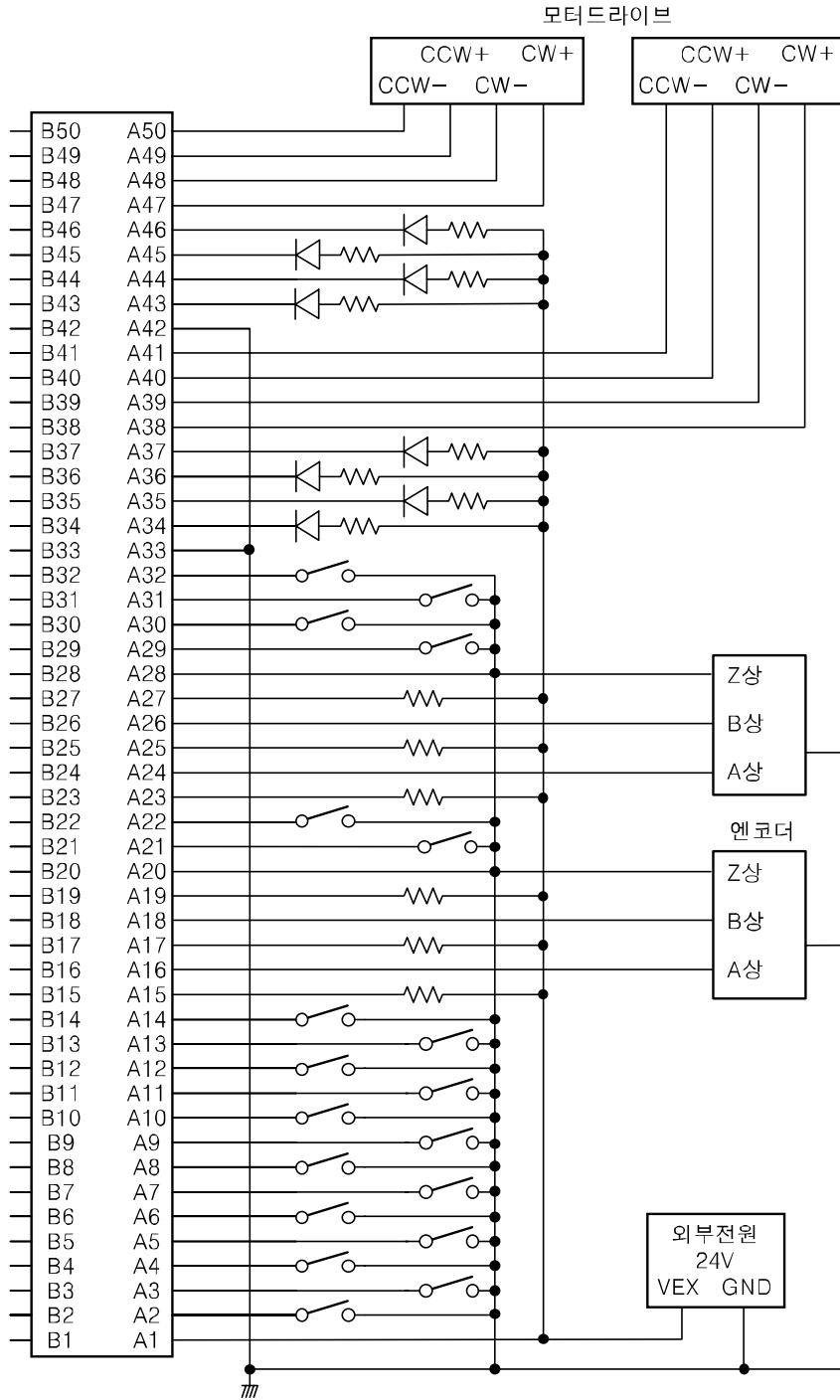
**(9) I/O test**

PMC-4B-PCI 보드의 입/출력 테스트를 위한 프로그램입니다

본 프로그램의 소스(C 언어 라이브러리 및 예제)는 당사 홈페이지에서 다운로드 할 수 있습니다. PC 에 카드를 장착하고, 윈 드라이브를 인스톨 한 후, ioTest 폴더 안의 ioTest.exe 를 실행하면 다음 화면이 표시됩니다.



각 문자는 입출력 및 상태 레지스트를 나타내는 것이며, 드라이브 펄스 및 각종 입/출력을 연결하여 신호를 가하면 해당 신호의 문자가 대문자는 소문자로, 소문자는 대문자로 변경되어 입/출력 상태를 확인할 수 있습니다. 아래와 같은 회로를 구성하여 입/출력 신호 및 출력 펄스를 확인할 수 있습니다.



- 15, 17, 19, 23, 25, 27 번에 연결된 저항은 1/2W 220Ω 저항이며 나머지 저항은 모두 1/2W 3.3kΩ저항입니다.
- 범용 출력핀에 연결되는 다이오드 사양은 50V/0.1A 이상을 사용하십시오.
- 엔코더는 오픈콜렉터 방식의 엔코더를 사용합니다.
- 그림에서 100 핀 커넥터에서 A 측 50 핀만 표시하였으며, 나머지 B 측 50 핀도 동일한 방식으로 결선하면 됩니다  
단, B 측의 2 번 단자는 사용하지 않습니다.
- 결선 방법은 '4 접속도' 를 참조하십시오.



# Table of Contents

	제품 구입 감사 안내문 .....	iii
	사용자 매뉴얼 안내 .....	iv
	사용자 매뉴얼의 공통 기호 .....	v
	안전을 위한 주의사항 .....	vi
	Table of Contents .....	xiii
<b>1</b>	<b>개요</b> .....	<b>17</b>
<b>2</b>	<b>정격/성능</b> .....	<b>25</b>
<b>3</b>	<b>외형치수도</b> .....	<b>27</b>
<b>4</b>	<b>접속도</b> .....	<b>29</b>
4.1	Drive 펄스 출력 신호의 접속(nP+P/N, nP-P/N) .....	29
4.2	범용 출력 신호의 접속(nOUT4~7) .....	29
4.3	입력 신호의 접속(nIN1~3, nINPOS, nALARM, nEXP+/-, EMG) .....	30
4.4	엔코더 입력 신호(nECAP/N, nECBP/N)와 nIN0+/- 신호의 접속 .....	30
4.5	Limit 입력 신호의 접속(nLMIT+/-) .....	31
<b>5</b>	<b>기능 설명</b> .....	<b>33</b>
5.1	정량 Pulse drive와 연속 Pulse drive .....	33
5.1.1	정량 Pulse drive .....	33
5.1.2	연속 Pulse drive .....	36
5.2	속도 Curve .....	37
5.2.1	정속 Drive .....	37
5.2.2	직선 가감속 Drive .....	38
5.2.3	비대칭 직선 가감속 Drive .....	40
5.2.4	S자형 가감속 Drive .....	42
5.2.5	비대칭 S자 가감속 Drive .....	46
5.2.6	Drive pulse 폭과 속도 정밀도 .....	48
5.3	포지션 (Position) 관리 .....	50
5.3.1	논리 위치 Counter와 실제위치 Counter .....	50
5.3.2	비교 Register와 소프트웨어 리미트 .....	51
5.3.3	위치 Counter의 가변 링 .....	52
5.3.4	외부 신호에 의한 실제위치 Counter의 Clear .....	53
5.4	보간 .....	55
5.4.1	2축/3축 직선보간 .....	56
5.4.2	원호보간 .....	58
5.4.3	2축/3축 Bit pattern 보간 .....	61
5.4.4	선속일정 .....	67
5.4.5	연속보간 .....	69
5.4.6	가감속 Drive에서의 보간 .....	73
5.4.7	보간 Step 전송 (Command) .....	76
5.5	자동 원점 복귀 출력 .....	77
5.5.1	각 Step 의 동작 .....	78
5.5.2	서치 속도와 Mode의 설정 .....	80

5.5.3	자동원점복귀의 실행과 상황(Status).....	81
5.5.4	자동 원점 출력시의 에러.....	83
5.5.5	자동 원점 출력 시의 주의점.....	84
5.5.6	자동 원점 복귀의 예.....	85
5.6	동기 동작.....	91
5.6.1	동기 동작의 예.....	93
5.6.2	동기 동작의 지연시간.....	98
5.6.3	동기 동작의 주의점.....	99
5.7	Interrupt.....	100
5.8	입력 신호 필터.....	102
5.9	그 밖의 기능.....	105
5.9.1	외부 신호에 의한 Drive 조작.....	105
5.9.2	Pulse 출력 방식의 선택.....	107
5.9.3	Pulse 입력 방식의 선택.....	108
5.9.4	하드웨어 Limit 신호.....	108
5.9.5	서보(Servo) 모터 Driver 대응의 신호.....	109
5.9.6	긴급정지.....	109
5.9.7	Drive 상태의 출력.....	110
5.9.8	범용 출력 신호.....	110
<b>6</b>	<b>각 신호의 설명.....</b>	<b>111</b>
<b>7</b>	<b>Read/Write register.....</b>	<b>115</b>
7.1	16 bit data bus의 Register address.....	115
7.2	8 bit data bus의 Register address.....	118
7.3	WR0 command register.....	119
7.4	WR1 mode register 1.....	120
7.5	WR2 mode register 2.....	121
7.6	WR3 mode register 3.....	123
7.7	WR5 보간 Mode register.....	125
7.8	WR6, 7 Write data register 1, 2.....	126
7.9	RR0 status register.....	127
7.10	RR1 status register 1.....	129
7.11	RR2 status register 2.....	131
7.12	RR3 status register 3.....	132
7.13	RR4, 5 Input register 1, 2.....	133
7.14	RR6, 7 read data register 1, 2.....	133
<b>8</b>	<b>명령 일람.....</b>	<b>135</b>
<b>9</b>	<b>Data 기록(Write) 명령.....</b>	<b>137</b>
9.1	Range 설정.....	137
9.2	가속도 증가율 설정.....	138
9.3	가속도 설정.....	138
9.4	감속도 설정.....	139
9.5	기동 속도 설정.....	139
9.6	Drive 속도 설정.....	140
9.7	출력 Pulse 수/보간 종점 설정.....	140

9.8	매뉴얼(Manual) 감속점 설정 .....	141
9.9	원호 중심점 설정 .....	141
9.10	논리위치 Counter 설정 .....	141
9.11	실제위치 Counter 설정 .....	141
9.12	COMP+ register 설정 .....	141
9.13	COMP- register 설정 .....	142
9.14	가속 Counter offset 설정 .....	142
9.15	감속도 증가율 설정 .....	142
9.16	확장 Mode 설정 .....	143
9.17	원점 검출 속도 설정 .....	145
9.18	동기 동작 Mode 설정 .....	145
<b>10</b>	<b>Data 읽기 (Read) 명령 .....</b>	<b>147</b>
10.1	논리위치 Counter 읽기 .....	147
10.2	실제위치 Counter 읽기 .....	147
10.3	현재 Drive 속도 읽기 .....	147
10.4	현재 가/감속도 읽기 .....	147
10.5	동기 동작 버퍼 Register 읽기 .....	148
<b>11</b>	<b>Drive 명령 .....</b>	<b>149</b>
11.1	+ 방향 정량 Pulse drive .....	149
11.2	- 방향 정량 Pulse drive .....	150
11.3	+방향 연속 Pulse drive .....	150
11.4	- 방향 연속 Pulse drive .....	150
11.5	Drive 시작 홀드(Hold).....	150
11.6	Drive 시작 (Free)/종료 Status clear.....	151
11.7	Drive 감속 정지 .....	151
11.8	Drive 즉시 정지 .....	151
<b>12</b>	<b>보간 명령 .....</b>	<b>153</b>
12.1	2축 직선 보간 Drive .....	153
12.2	3축 직선 보간 Drive .....	153
12.3	CW 원호보간 Drive .....	153
12.4	CCW 원호 보간 Drive .....	154
12.5	2축 bit pattern 보간 Drive .....	154
12.6	3축 bit pattern 보간 Drive .....	154
12.7	BP register 기입 가능 .....	154
12.8	BP register 기입 불가.....	154
12.9	BP data stack.....	155
12.10	BP data clear .....	155
12.11	보간 Single step .....	155
12.12	감속 유효.....	155
12.13	감속 무효.....	155
12.14	보간 Interrupt clear.....	156
<b>13</b>	<b>그 밖의 명령 .....</b>	<b>157</b>

13.1	자동 원점 복귀 실행.....	157
13.2	동기 동작 기동.....	157
13.3	NOP .....	157
<b>14</b>	<b>제어 프로그램의 예.....</b>	<b>159</b>
<b>15</b>	<b>입/출력 신호 타이밍.....</b>	<b>173</b>
15.1	파워-온 타이밍 .....	173
15.2	Drive 시작/종료 시.....	173
15.3	보간 Drive 시 .....	174
15.4	Drive 시작 프리 .....	174
15.5	Drive 즉시 정지 .....	174
15.6	Drive 감속 정지 .....	175
<b>16</b>	<b>사양 정리.....</b>	<b>177</b>
<b>17</b>	<b>부록: 가감 속도 Drive의 속도 프로파일.....</b>	<b>181</b>



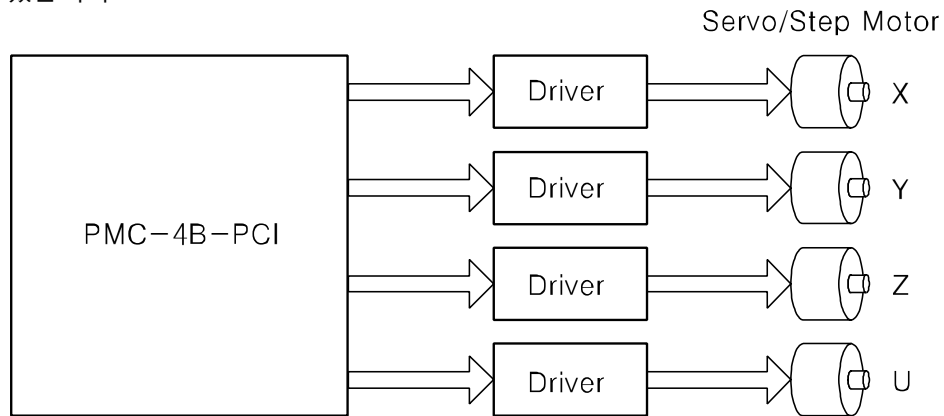
# 1 개요

PMC-4B-PCI 는, 1Board 로 4 축의 pulse 입력의 서보 모터, Step 모터의 위치 결정 제어 (Positioning control), 보간 Drive (Interpolation drive), 또는 속도 제어 (Speed control)가 가능한 PC/AT 호환기 PCI bus 대응의 회로 기판입니다. 다음의 기능을 갖추고 있습니다.

## (1) 독립 4 축 Drive

4 축을 독립적으로 제어할 수 있습니다. 4 축의 기능은 모두 동일합니다.

정속 Drive, 직선 가감속 Drive, S 자 가감속 Drive 등 4 축 모두 동일하게 제어할 수 있습니다.



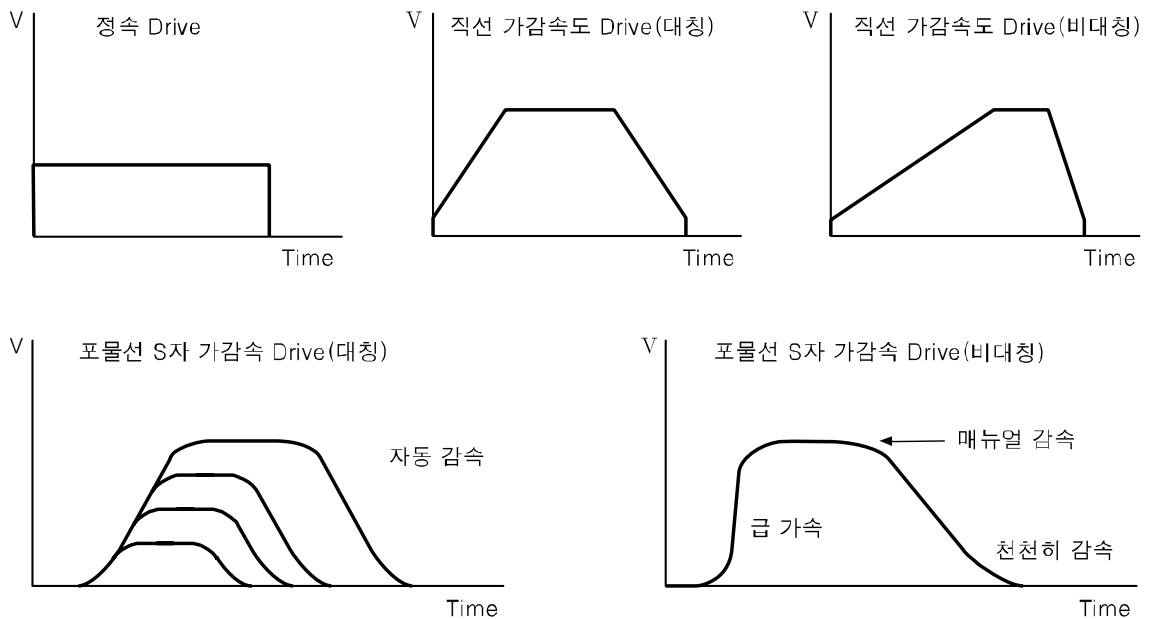
## (2) 속도 제어

Drive 속도는 1pps 에서 최고 4Mpps 까지 출력할 수 있으며 정속 Drive, 직선 가감속 Drive, S 자 가감속 Drive 가 가능합니다. 출력되는 Drive pulse 의 속도 정밀도는 설정치에 대하여  $\pm 0.1\%$  이하입니다. (CLK=16MHz 표준 시). 또한, Drive 중에 Drive 속도를 자유롭게 변경 가능합니다.

**(3) 가감속 Drive**

각 축 가감속 Drive에는 정속 Drive, 직선 가감속 Drive (대칭/비대칭), S자 가감속 Drive (대칭/비대칭)를 실행할 수 있습니다. 직선 가감속 정량 Pulse drive에서는 대칭형 직선 가감속 정량 Pulse drive와 비대칭형 직선 가감속 정량 Pulse drive 모두, 자동 감속이 가능합니다.

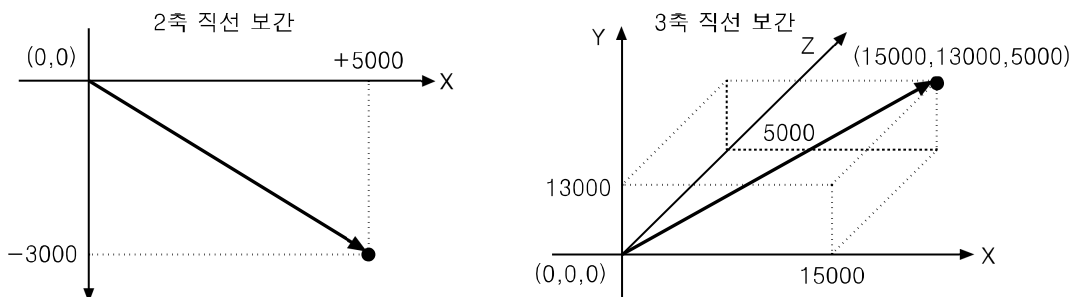
S자 가감속은 1차 직선으로 증가/감소하는 방식을 사용하므로 속도 커브는 2차 포물선 가속/감속으로 나타납니다. 또한, S자 가감속 정량 Pulse drive에 있어서는 대칭 S자에 한하여 자동 감속이 가능하고 S자 가감속 driver 중 삼각파형 방지 기능도 있습니다.



**(4) 2축/3축 직선보간**

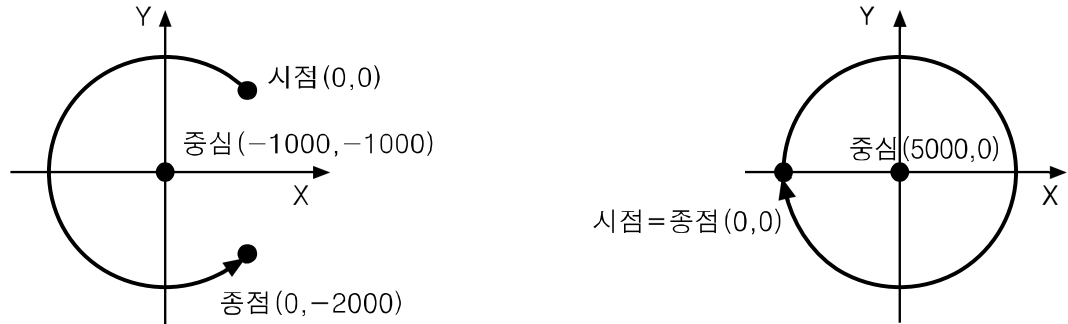
4축의 Driver 중에서 임의의 2축, 또는 3축을 선택하고 2축/3축 직선 보간 Drive를 실행할 수 있습니다. 보간 좌표 범위는 현재 위치에서 -2,147,483,646 ~ +2,147,483,646입니다.

지정된 직선에 대한 위치 오차는 전 보간 범위 내에서 ±0.5 LSB입니다. 보간 속도는 1pps ~ 4Mpps입니다.



**(5) 원호 보간**

4 축의 Driver 중에서 임의의 2 축을 선택하고 원호보간 Drive 를 실행할 수 있습니다. 보간 좌표 범위는 현재 위치에서 -2,147,483,646 ~ +2,147,483,646 입니다. 지정된 원호 곡선에 대한 위치 오차는 전 보간 범위내에서 ±1 LSB 입니다. 보간 속도는 1pps ~ 4Mpps 입니다.

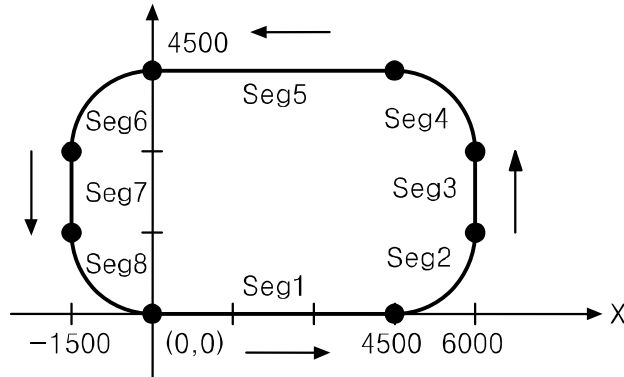


**(6) 2 축/3 축 Bit pattern 보간**

상위 CPU 로 연산된 Bit pattern화된 보간 Data 를 각 축 16bit 단위로 받아들여, 지정된 Drive 속도로 보간 Pulse 를 연속적으로 출력하는 보간 Drive 입니다. 이 기능을 이용하여 상위 CPU 에서 만들어진 다양한 궤적을 그릴 수 있습니다.

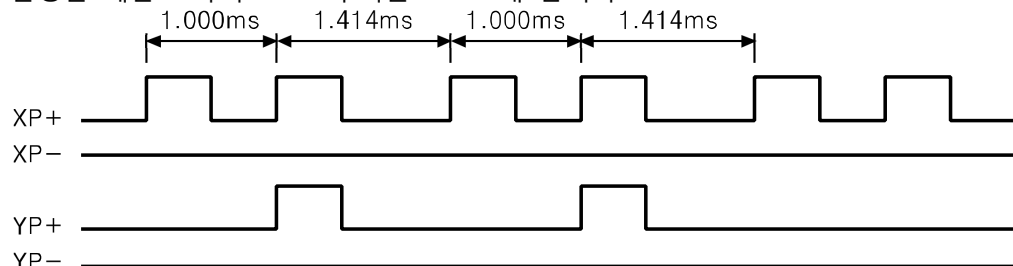
**(7) 연속 보간 동작**

직선 보간→원호 보간→직선 보간→... 처럼 보간 drive 를 각각의 보간 명령마다 정지하지 않고, 연속적으로 실행할 수 있습니다. 연속 보간시의 Drive 속도는 최고 2MHz 입니다.



**(8) 선속 일정 제어선속 일정 제어**

보간을 실행하고 있는 축의 합성 속도를 항상 일정하게 하는 기능입니다. 2 축 동시 Drive pulse 를 발생할 때 2 축의 Pulse 주기를 1.414 배, 3 축 동시에 Drive pulse 를 발생할 때는 3 축의 Pulse 주기를 1.732 배 합니다.



**(9) 포지션 관리 기능**

4 축 모두, 모션 컨트롤 IC 내부에서 Drive pulse 출력을 제어하는 논리위치 Counter와 외부 Encoder로부터 Pulse를 제어하는 실제 위치 Counter 2개 (32bit)를 내장하고 있습니다.

**(10) 비교 Register와 소프트 Limit 기능**

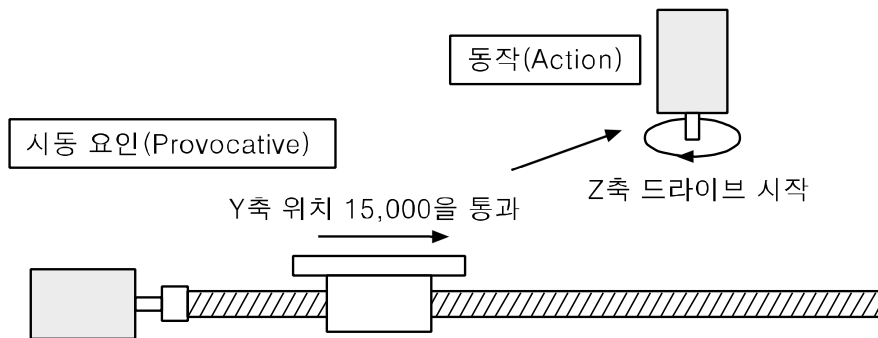
논리 위치 Counter와 실제 위치 Counter와의 위치를 비교하기 위한 32bit 비교 Register를 각 축 2개씩 가지고 있습니다. Drive 중에 이러한 비교 Register와 논리/실제 위치 Counter와의 대소 관계를 실시간적으로 읽을 수 있으며, 대소 관계의 변화를 이용하여 Interrupt 발생도 가능합니다. 또한, 2개의 비교 Register를 소프트 Limit로 동작시키는 것도 가능합니다.

**(11) 자동 원점 출력**

PMC-4B-PCI는 특별한 명령 없이 고속 원점 근접 서치→저속 원점 서치→Encoder Z상 서치→Offset 이동 등의 일련의 원점 출력 시퀀스를 자동으로 실행하는 기능을 가지고 있습니다. 다축 제어에 있어서 CPU의 부담을 줄입니다.

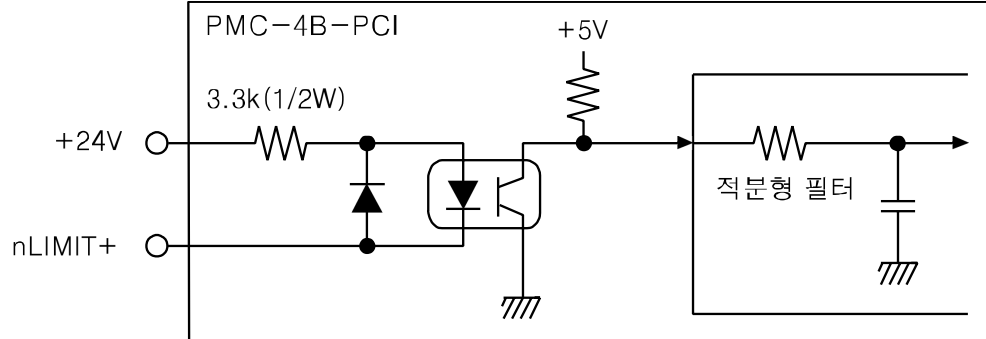
**(12) 동기 동작**

동기 동작은 각 축 사이 및 모션 컨트롤러 이외의 디바이스 사이에 있어 설정된 동작 신호가 발생하면 Drive 시작이나 정지 등의 지정된 동기 동작을 실행하는 기능입니다. 동기 동작의 시동 요인으로는 지정 위치 통과, Drive 시작/종료, 외부 입력 신호의 상승/하강 등 10 종류가 준비되어 있으며 동기 동작 발생 후, 동작으로서는 Drive 시작/정지, 위치 Counter 값 저장, Drive 속도 기록 등 14 종류가 준비되어 있습니다.



**(13) 입력 신호 필터**

각 입력 신호의 입력단에는 적분필터를 갖추고 있습니다. 입력신호 마다 필터 기능을 유효하게 할 지, 무효하게 할 지를 설정할 수 있습니다. 또한, 필터의 통과 시간은 8 종류 중에서 하나를 선택할 수 있습니다.

**(14) 외부 조작 신호**

각 축은 외부 신호에 의하여 +/- 방향의 정량 Pulse drive, 연속 Pulse drive 를 실시할 수 있습니다. 이 기능이 수동 Pulse 출력 (MANUAL JOG DRIVE)으로 가능하여 상위 CPU 의 부담을 줄일 수 있습니다.

**(15) 서보 모터용 각종 신호**

2 상 Encoder 신호, 인포지션 (Inposition), Alarm 등의 서보모터 Driver 출력 신호를 입력할 수 있습니다.

**(16) Interrupt 발생 기능**

각 축 모두, 가감속 Drive 중의 정속 시작 시, 정속 종료 시, Drive 종료 시, 위치 Counter 와 비교 Register 의 대소 관계가 변경되었을 경우 등 다양한 요인으로 Interrupt 를 발생시킬 수 있습니다. 또한, 연속 보간, Bit pattern 보간에서는 다음 Data 요구에 대한 Interrupt 발생도 가능합니다.

**(17) 실시간 처리 모니터링 기능**

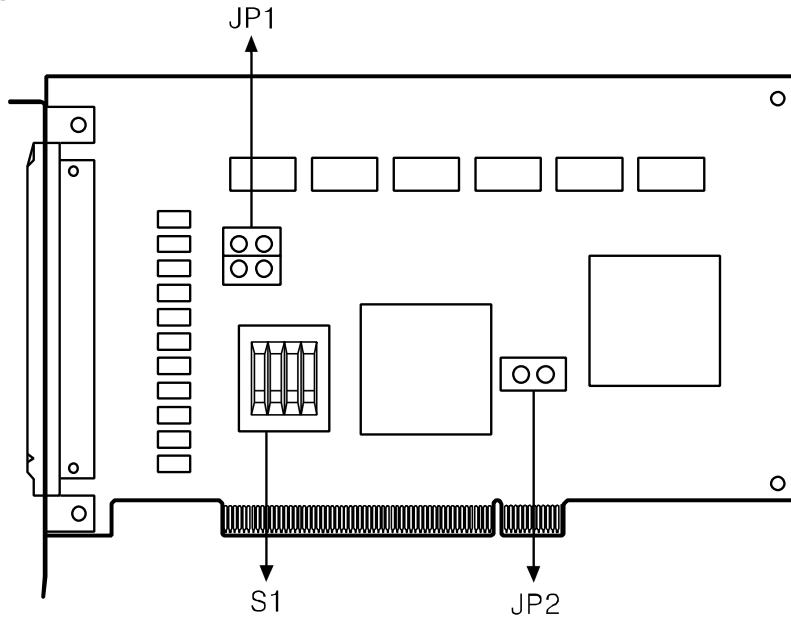
Drive 중에 현재의 논리 위치, 실제 위치, Drive 속도, 가속도, 가감속 상태 (가속 중, 정속 중, 감속 중) 등을 실시간으로 읽을 수 있습니다.

**(18) 16bit bus 대응**

상위 CPU 와의 data bus 는 16bit 로 접속이 가능합니다.

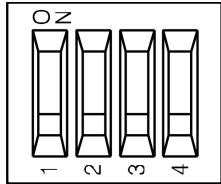
X, Y, Z, U 4 축의 기능은 완전히 동일하며 보간 Drive 시에는 주축(ax1)으로 지정된 축의 기본 Pulse 발진의 타이밍에 보간 연산을 합니다. 정속 Drive 에서도 가감속 Drive 에서도 실시할 수가 있습니다.

**(19) 스위치 및 점퍼핀의 용도**



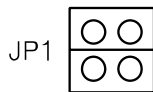
■ S1 스위치의 용도 및 설정

S1 스위치는 PMC-4B-PCI 보드를 여러장 사용할 때, 보드의 ID를 구분하기 위한 스위치로 특별한 기준 없이 다른 보드와 스위치 설정을 다르게 하여 사용하시면 됩니다. 총 16개의 보드를 동시 사용 가능합니다.



■ JP1 점퍼핀의 용도 및 설정

긴급 정지(EMG)의 Active 레벨을 High로 할지 Low로 할지를 선택합니다.



긴급 정지의 Active 레벨 설정은 '2.9.6 긴급정지'를 참고하십시오.

■ JP2 점퍼핀의 용도 및 설정



EEPROM의 프로그램 쓰기 마스크로써 사용자는 JP2 점퍼핀을 임의로 제거할 수 없습니다.

## (20) 입/출력 사양

I/O connector 의 핀 배치


핀 번호	핀 이름	핀 설명	핀 번호	핀 이름	핀 설명
A1	VEX	12-24 VDC	B1	VEX	12-24 VDC
A2	EMG	비상정지(4 축 정지)	B2	-	-
A3	XLIMIT+	X 축 + 방향 리미트	B3	ZLIMIT+	Z 축 + 방향 리미트
A4	XLIMIT-	X 축 - 방향 리미트	B4	ZLIMIT-	Z 축 - 방향 리미트
A5	XIN1	X 축 입력신호(원점신호)	B5	ZIN1	Z 축 입력신호(원점신호)
A6	XIN0	X 축 입력신호(원점근접신호)	B6	ZIN0	Z 축 입력신호(원점근접신호)
A7	XIN3	X 축 입력신호(엔코더 Z 상 신호)	B7	ZIN3	Z 축 입력신호(엔코더 Z 상 신호)
A8	YLIMIT+	Y 축 +방향 리미트	B8	ULIMIT+	U 축 +방향 리미트
A9	YLIMIT-	Y 축 -방향 리미트	B9	ULIMIT-	U 축 -방향 리미트
A10	YIN1	Y 축 입력신호(원점신호)	B10	UIN1	U 축 입력신호(원점신호)
A11	YIN0	Y 축 입력신호(원점근접신호)	B11	UIN0	U 축 입력신호(원점근접신호)
A12	YIN3	Y 축 입력신호(엔코더 Z 상 신호)	B12	UIN3	U 축 입력신호(엔코더 Z 상 신호)
A13	XINPOS	X 축 위치결정완료 입력	B13	ZINPOS	Z 축 위치결정완료 입력
A14	XALRAM	X 축 알람 입력	B14	ZALRAM	Z 축 알람 입력
A15	XECAP	X 축 엔코더 A 상+	B15	ZECAP	Z 축 엔코더 A 상+
A16	XECAN	X 축 엔코더 A 상-	B16	ZECAN	Z 축 엔코더 A 상-
A17	XECBP	X 축 엔코더 B 상+	B17	ZECBP	Z 축 엔코더 B 상+
A18	XECBN	X 축 엔코더 B 상-	B18	ZECBN	Z 축 엔코더 B 상-
A19	XECZP	X 축 엔코더 Z 상+	B19	ZECZP	Z 축 엔코더 Z 상+
A20	XECZN	X 축 엔코더 Z 상-	B20	ZECZN	Z 축 엔코더 Z 상-
A21	YINPOS	Y 축 위치결정완료 입력	B21	UNIPOS	U 축 위치결정완료 입력
A22	YALARM	Y 축 알람입력	B22	UALARM	U 축 알람입력
A23	YECAP	Y 축 엔코더 A 상+	B23	UECAP	U 축 엔코더 A 상+
A24	YECAN	Y 축 엔코더 A 상-	B24	UECAN	U 축 엔코더 A 상-
A25	YECBP	Y 축 엔코더 B 상+	B25	UECBP	U 축 엔코더 B 상+
A26	YECBN	Y 축 엔코더 B 상-	B26	UECBN	U 축 엔코더 B 상-
A27	YECZP	Y 축 엔코더 Z 상+	B27	UECZP	U 축 엔코더 Z 상+
A28	YECZN	Y 축 엔코더 Z 상-	B28	UECZN	U 축 엔코더 Z 상-
A29	XEXP+	X 축 매뉴얼 + 드라이브	B29	ZEXP+	Z 축 매뉴얼 + 드라이브
A30	XEXP-	X 축 매뉴얼 - 드라이브	B30	ZEXP-	Z 축 매뉴얼 - 드라이브
A31	YEXP+	Y 축 매뉴얼 + 드라이브	B31	UEXP+	U 축 매뉴얼 + 드라이브
A32	YEXP-	Y 축 매뉴얼 - 드라이브	B32	UEXP-	U 축 매뉴얼 - 드라이브
A33	GND	GND	B33	GND	GND
A34	XOUT4/C MPP	X 축 범용출력	B34	ZOUT4/C MPP	Z 축 범용출력
A35	XOUT5/C MPM	X 축 범용출력	B35	ZOUT5/C MPM	Z 축 범용출력
A36	XOUT6/A SND	X 축 범용출력	B36	ZOUT6/A SND	Z 축 범용출력
A37	XOUT7/D SND	X 축 범용출력	B37	ZOUT7/S ND	Z 축 범용출력
A38	XP+P	X 축 +방향 +드라이브 신호출력	B38	ZP+P	Z 축 +방향 +드라이브 신호출력
A39	XP+N	X 축 +방향 -드라이브 신호출력	B39	ZP+N	Z 축 +방향 -드라이브 신호출력
A40	XP-P	X 축 -방향 +드라이브 신호출력	B40	ZP-P	Z 축 -방향 +드라이브 신호출력
A41	XP-N	X 축 -방향 -드라이브 신호출력	B41	ZP-N	Z 축 -방향 -드라이브 신호출력
A42	GND	GND	B42	GND	GND
A43	YOUT4/C MPP	Y 축 범용출력	B43	UOUT4/C MPP	U 축 범용출력
A44	YOUT5/C MPM	Y 축 범용출력	B44	UOUT5/C MPM	U 축 범용출력
A45	YOUT6/A SND	Y 축 범용출력	B45	UOUT6/A SND	U 축 범용출력
A46	YOUT7/D SND	Y 축 범용출력	B46	UOUT7/D SND	U 축 범용출력

핀 번호	핀 이름	핀 설명	핀 번호	핀 이름	핀 설명
A47	YP+P	Y 축 +방향 +드라이브 신호출력	B47	UP+P	U 축 +방향 +드라이브 신호출력
A48	YP+N	Y 축 +방향 -드라이브 신호출력	B48	UP+N	U 축 +방향 -드라이브 신호출력
A49	YP-P	Y 축 -방향 +드라이브 신호출력	B49	UP-P	U 축 -방향 +드라이브 신호출력
A50	YP-N	Y 축 -방향 -드라이브 신호출력	B50	UP-N	U 축 -방향 -드라이브 신호출력



## 2 정격/성능

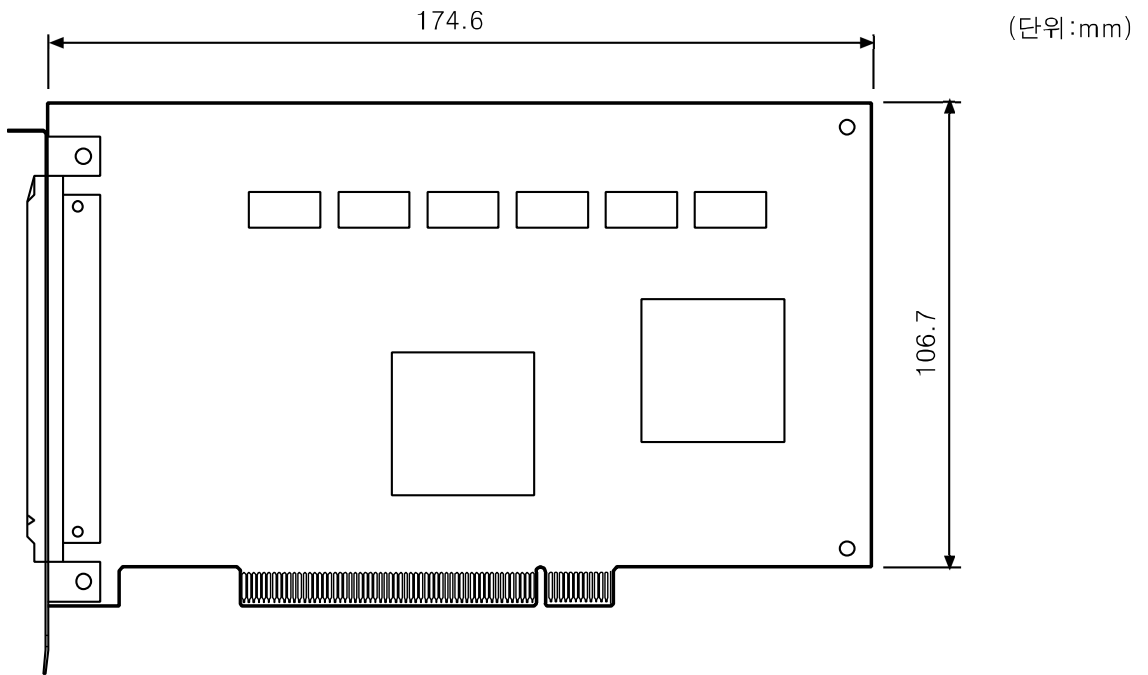
모델명	PMC-4B-PCI	
제어축	4 축	
전원전압	5VDC(PC 내부전원 사용)	
외부전원전압	12-24VDC	
허용전압변동범위	전원전압의 90~110%	
CPU 데이터 버스 길이	8/16 bit 선택	
2/3 축	보간범위	각 축 -2,147,483,648 ~ 2,147,483,647
	보간속도	1pps~4Mpps
직선 보간	위치정밀도	±0.5 LBS 이하(전 보간범위 내에서)
	원호 보간	보간범위
원호 보간	보간속도	1pps~4Mpps
	위치정밀도	±1 LBS 이하(전 보간범위 내에서)
2/3 축 bit pattern 보간속도	1pps~4Mpps(CPU 데이터 setup 시간에 의존)	
그외 보간기능	임의 축 선택가능, 선속일정, 연속보간, 보간 Step 전송(COMMAND, 외부신호)	
Drive 펄스 출력	출력회로 범위: 1pps~4Mpps	
(X, Y 축 공통사양)	출력속도 정밀도: ±0.1%이하(설정치에 대하여)	
	속도 배율: 1~500	
	S 자용 가감속도: 954~62.5×10 <sup>6</sup> pps/sec(배율=1 일때) (가/감속도의 증가율) 477×10 <sup>3</sup> ~31.25×10 <sup>9</sup> pps/sec(배율=500 일때)	
	가/감속도: 125~1×10 <sup>6</sup> pps/sec(배율=1 일때) 62.5×10 <sup>3</sup> ~500×10 <sup>6</sup> pps/sec(배율=500 일때)	
	초속도: 1~8,000pps(배율=1 일때)/500~4×10 <sup>6</sup> pps(배율=500 일때)	
	Drive 속도: 1~8,000pps(배율=1 일때)/500~4×10 <sup>6</sup> pps(배율=500 일때)	
	출력 펄스 수: 0~4,294,967,295(정량 펄스 drive)	
	속도 Curve: 정속/대칭, 비대칭 직선 가감속/포물선 S 자 가감속 Drive	
	정량 펄스 drive 의 감속 mode 자동감속(비대칭 직선 가감속 가능)/Manual 감속	
	Drive 중의 출력 펄스, Drive 속도의 변경 가능	
	독립 2 펄스/1 펄스 방향 방식 선택 가능	
	펄스 논리 Level 선택 가능, 출력 단자 변경 가능	
엔코더 입력 펄스	2 상 펄스/Up down 펄스 입력 가능, 2 상 펄스 1, 2, 4 체배 선택 가능	
위치 카운터	논리 위치 카운터(출력 펄스용) 카운터 범위: -2,147,483,648~2,147,483,647 실제 위치 카운터(입력 펄스용) 카운터 범위: -2,147,483,648~2,147,483,647	

Compare register	Comp. +register 위치 비교범위: -2,147,483,648~2,147,483,647	
	Comp. -register 위치 비교범위: -2,147,483,648~2,147,483,647	
	위치 카운터와의 대/소를 Status 출력 및 신호 출력	
	Software limit 로써 동작 가능	
자동원점복귀	Step1(고속원점근접서치) → Step2(저속원점근접서치)	
Interrupt 기능(보간제외)	1Drive 펄스 출력 위치카운터 $\geq$ Comp.- 변할때, 위치카운터 $\leq$ Comp.+ 변할때 위치카운터 < Comp.- 변할때, 위치카운터 > Comp.+ 변할때 가감속 Drive 중의 정속 시작 시, 가감속 Drive 중의 정속 완료 시 Drive 종료시 자동원점, 자동원점 복귀 종료, 동기 동작	
외부신호에 의한 Drive 조작	EXP+, EXP- 신호에 의해 +/- 방향의 정량/연속 펄스 Drive 가 가능 2 상 엔코더 신호 모드(엔코더 입력) Drive 가능	
외부감속정지/즉시정지신호	IN 0~3 각축 4 점	
	신호의 유효/무효 및 논리 Level 선택 가능, 범용 입력으로 사용 가능	
서보모터용 입력신호	알람(Alarm), INPOS(위치 결정 완료) 신호의 유효/무효 및 논리 Level 선택 가능	
범용 출력신호	OUT4~7 각 축 4 점(Drive 상태 출력신호와 단자 공용)	
Drive 상태 신호출력	ASND(가속중), DSND(감속중)	
Overrun limit 신호입력	+방향, -방향 각 1 점, 논리 Level 선택 가능	
	Active 시, 즉시 정지/감속 정지 선택 가능	
긴급정지 신호입력	EMG 1 점, Low leve 로 전축의 Drive 펄스를 즉시 정지	
적분형 필터내장	각 입력신호의 입력단에 적분필터 내장, 통과시간(8 종류) 선택 가능	
기타	임의의 축 선택가능, 선속일정, 연속보간, 보간 Step 전송(Command, 외부신호)	
내환경성	사용주위온도	0~45°C, 보존 시: -10~55°C
	사용주위습도	35~85%RH, 보존 시: 35~85%RH
획득 규격	CE 	
중량	약 98g	

※내환경 항목의 온, 습도는 결빙 또는 결로되지 않는 상태입니다.

※중량은 포장박스를 제외한 무게입니다.

### 3 외형치수도



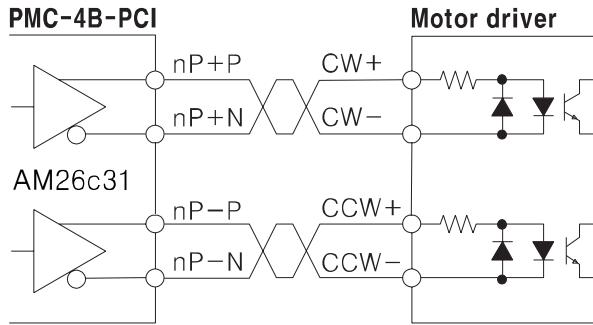


## 4 접속도

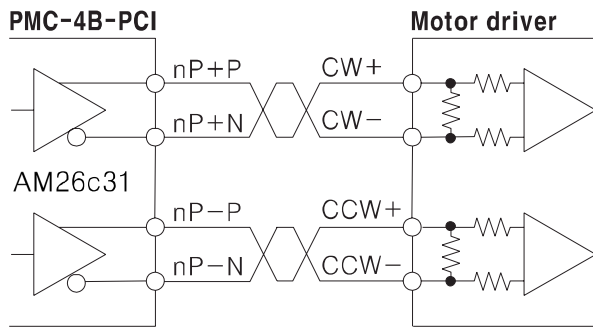
### 4.1 Drive 펄스 출력 신호의 접속(nP+P/N, nP-P/N)

Drive 펄스 출력은 +/-방향의 Drive 펄스 신호를 차동출력의 Line driver(AM26c31)를 이용하여 출력합니다. 포토커플러 및 Line driver 입력을 갖는 Motor driver 와의 접속예입니다.

#### (1) 포토커플러 입력의 Motor driver 와의 접속예

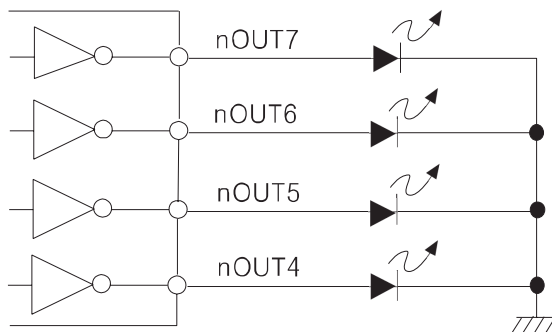


#### (2) Line driver 입력의 Motor driver 와의 접속예

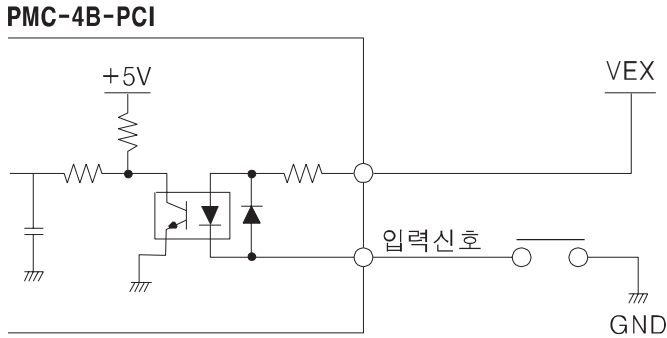


### 4.2 범용 출력 신호의 접속(nOUT4~7)

출력 신호는 버퍼(74LS06)로 출력되며, Reset 후에는 모든 출력이 OFF 됩니다.

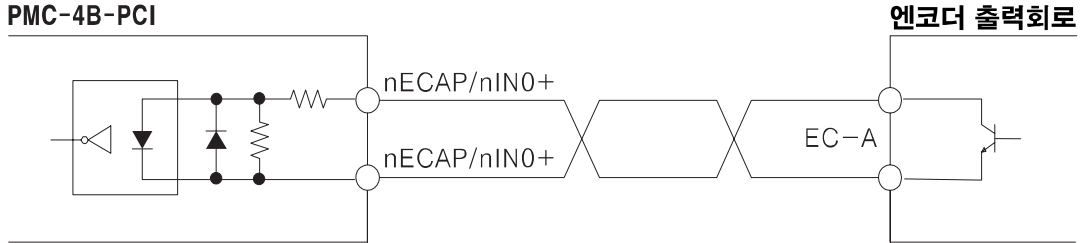


### 4.3 입력 신호의 접속(nIN1~3, nINPOS, nALARM, nEXP+/-, EMG)

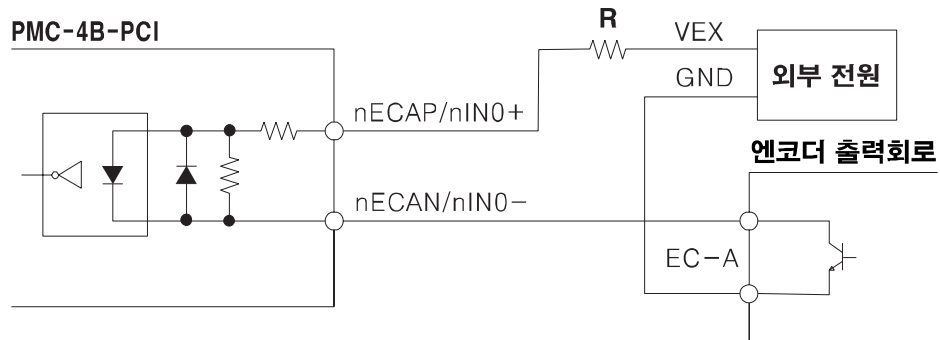


### 4.4 엔코더 입력 신호(nECAP/N, nECBP/N)와 nIN0+/- 신호의 접속

#### (1) 엔코더 차동출력 Line driver와의 접속 예



#### (2) 엔코더 NPN Open collector 출력 접속 예

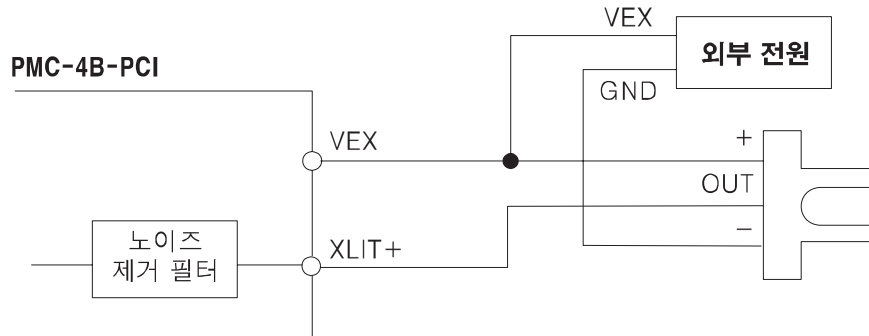


외부전원전압	저항(R)
5V	0
12V	820Ω 1/4W
24V	2kΩ 1W

※엔코더의 A, B, Z 상 모두 접속은 동일합니다.

## 4.5 Limit 입력 신호의 접속(nLMIT+/-)

Limit 신호는 일반적으로, 외부로 배선의 노출이 불가피하므로 노이즈에 취약합니다. 포토커플러만으로는 노이즈 제거가 불가능하여 PMC-4B-PCI 내부에 필터회로를 내장하였으므로 적당한 통과시간(FL=2, 3)을 설정하십시오.







## 5 기능 설명

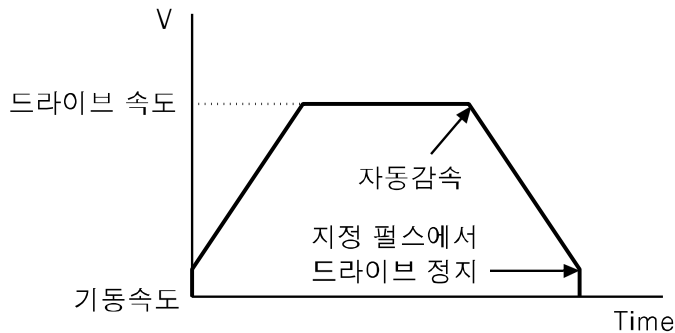
### 5.1 정량 Pulse drive와 연속 Pulse drive

각 축의 Drive pulse 출력은 기본적으로 + 방향/-방향의 정량 Pulse drive 명령, 또는 연속 Pulse drive 명령이 있습니다.

#### 5.1.1 정량 Pulse drive

정량 Pulse drive 는 지정된 출력 Pulse 수 만큼 정속 Drive 또는 가감속 Drive 를 합니다. 이동 대상물을 일정한 위치로 이동시킬 때, 정해진 펄스양 (정해진 이동 거리) 만큼 동작을 실시하고 싶을 때에 사용합니다.

가속도와 감속도가 동일한 가감속에서의 정량 Pulse drive 의 동작은 <그림 2.1>과 같이 출력 Pulse 의 나머지가 가속시에 소비된 Pulse 수 보다 작으면 자동 감속을 시작하고 지정된 출력 Pulse 를 출력하면 Drive 를 종료합니다.



<그림 2.1 정량 Pulse Drive>

정량 Drive 를 가감속 Drive 로 실시하기 위해서는 다음과 같은 Parameter 설정이 필요합니다.

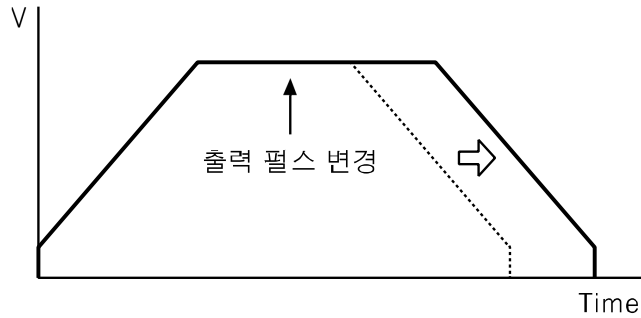
파라미터	설정값
Range	R
가/감속도	A/D
기동속도	SV
Drive 속도	V
출력 pulse 수	P

**(1) Drive 중, 출력 Pulse 수 변경**

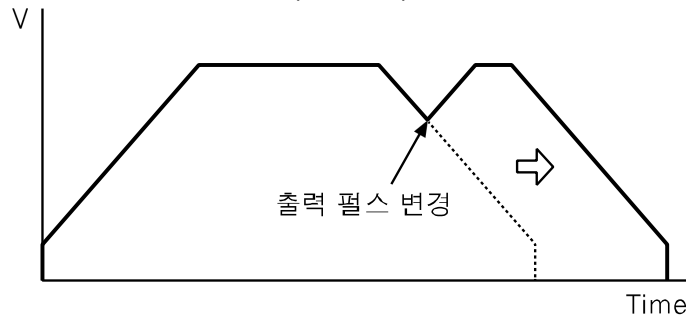
정량 Pulse drive 중에 출력 Pulse 수를 변경할 수 있습니다.

가감속 Drive 중, 출력 Pulse 의 나머지가 가속시 Pulse 수 보다 적게 되어, 감속으로 들어갈 때 출력 Pulse 수가 변경되는 경우는 다시 가속을 시작합니다. <그림 2.2>

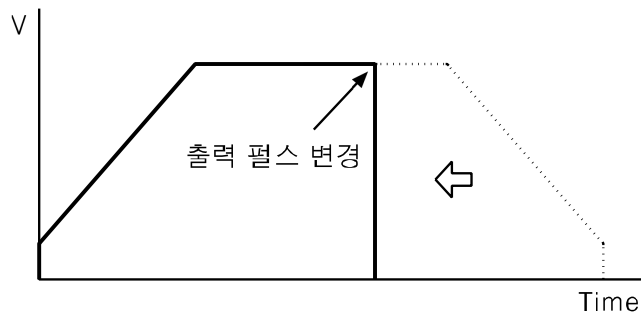
또한, 변경된 출력 Pulse 수가 이미 종료된 Pulse 수 보다 작은 경우는 즉시 정지 <그림 2.4> 합니다. S 자 가감속에서는 <그림 2.3>과 같이 감속시에 pulse 수가 변경되면 올바른 S 자 커브를 그릴 수가 없기 때문에 주의하십시오.



<그림 2.2 Drive 중 출력 Pulse 수 변경>



<그림 2.3 감속시 출력 Pulse 수 변경도>



<그림 2.4 Drive 중 출력 Pulse 수 변경>

## (2) 가감속 정량 Pulse drive 에 있어서의 매뉴얼 감속

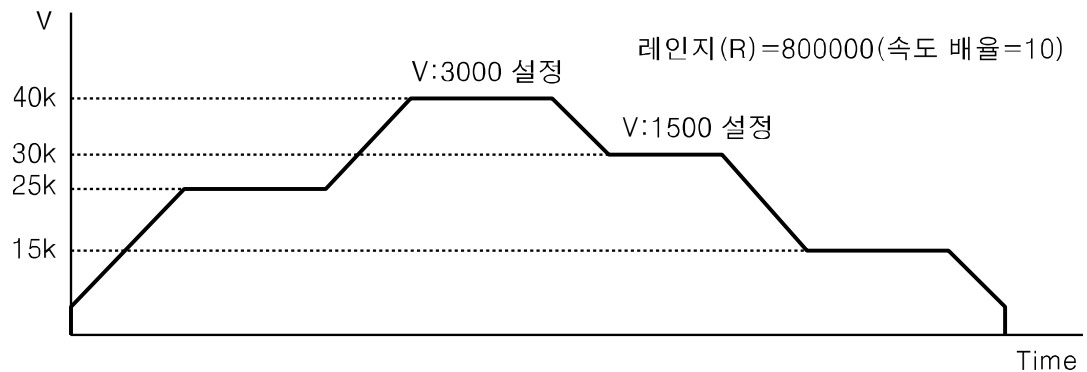
가감속 정량 Pulse drive 에서는 일반적으로 <그림 2.1>과 같이 계산된 감속점에서 자동 감속하지만 감속점을 매뉴얼로 지정할 수도 있습니다. 아래와 같은 경우, 자동 감속점이 지정되지 않거나 올바른 감속점을 산출할 수 없기 때문에 매뉴얼로 감속점을 지정해야 합니다.

- 직선 가감속 정량 Drive 에 있어서 Drive 도중에 속도 변경을 할 경우
- S 자 가감속 정량 Drive 에 있어서 가속도와 감속도, 가속도 증가율과 감속도 증가율을 개별로 설정할 경우
- 원호보간, bit pattern 보간, 연속보간을 가감속으로 실시할 경우  
매뉴얼 감속 Mode 로 하려면 WR3 register 의 D0 bit 를 1 로 하여 매뉴얼 감속점 설정 명령 (07h) 에 의해 감속점을 설정합니다. 그 외의 조작은 일반적인 정량 Drive 와 같습니다.

## (3) Drive 중 Drive 속도 변경

직선 가감속 및 정속의 정량 Drive 에 대해서는 Drive 도중에 Drive 속도 (V) 를 변경할 수가 있습니다.

다만, 직선 가감속 정량 Pulse drive 에서 Drive 속도를 변경하면 Drive 가 정지되는 경우가 발생하기 때문에 기동속도가 낮은 경우에는 주의해 주십시오. 또한, S 자 가감속 정량 Drive 에서는 Drive 도중에 Drive 속도(V)를 변경할 수 없습니다.

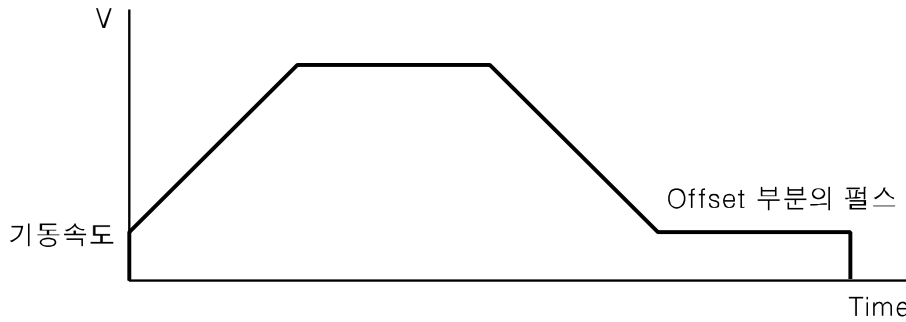


<그림 2.5 Drive 도중의 Drive 속도 변경의 예>

## (4) 가감속 정량 Pulse drive 의 가속 Counter offset

가감속 정량 drive 의 동작에서는 가속시에 가속으로 소비되는 Pulse 를 가속 Counter 로 Count 합니다. 설정되어 있는 출력 Pulse 수의 나머지가 가속 Counter 의 값보다 적게 되면 감속을 시작하고, 감속에는 가속과 같은 Pulse 수를 출력합니다. 가속 Counter offset 은 가속 Counter 로 지정된 Offset 값을 <그림 2.6>과 같이 추가합니다.

가속 Counter offset 은 Reset 시 8 로 설정됩니다. 일반적으로는 직선 가감속 Drive 를 실시하는 경우에는 본 Parameter 를 재 설정할 필요는 거의 없습니다. 비대칭 사다리꼴 가감속이나 S 자 가감속 정량 Drive 로 Drive 종료 시에 속도가 기동속도까지 떨어지지 않을 때 등에 가속 Counter offset 을 적당한 값에 설정하여 보정합니다.



<그림 2.6 가속 Counter offset>

### 5.1.2 연속 Pulse drive

연속 Drive 는 정지명령 또는 외부로부터 정지신호가 Active 될 때 까지 연속하여 Drive pulse 를 출력합니다.

원점 서치, 스캐닝 조그 이송 또는 속도 제어로 모터를 회전시킬 때 사용합니다.

연속 Pulse drive 에서는 Drive 도중에 Drive 속도를 자유롭게 변경할 수가 있습니다.

정지 명령에는 감속 정지 명령과 즉시 정지 명령이 있습니다.

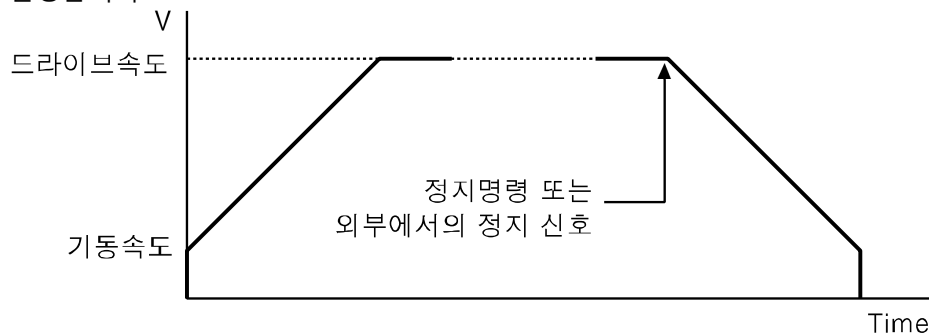
외부로부터 감속/즉시정지 신호는 각 축 IN3~IN0 (+/-) 의 4 점이 준비되어 있으며 각 신호는 유효/무효, Active level 을 설정할 수가 있습니다.

#### (1) 연속 Pulse drive 에 의한 원점 검출 동작

Encoder Z 상 신호, 원점 신호, 원점 근접신호를 nECZ, nIN1, nIN0 에 설정합니다. 각 축의 WR1 register 로 각 신호의 유효/무효, 논리 level 을 설정합니다. 고속 서치의 경우 가감속으로 Drive 하여 연속 Pulse drive 를 실시합니다.

이 때 설정된 신호가 Active level 이 되면 감속정지를 합니다.

저속 서치의 경우는 정속으로 연속 Pulse drive 를 실시합니다. 설정된 신호가 Active level 이 되면 즉시 정지합니다. 자동 원점 복귀 기능을 사용하는 경우에는 Z 상 신호는 nECZP/N, 원점 신호는 nIN1, 원점 근접 신호는 nIN0 으로 설정됩니다. 연속 Drive 를 가감속으로 실시하려면 출력 Pulse 수 이외는 정량 Drive 와 같은 Parameter 로 설정합니다.



<그림 2.7 연속 Pulse drive>

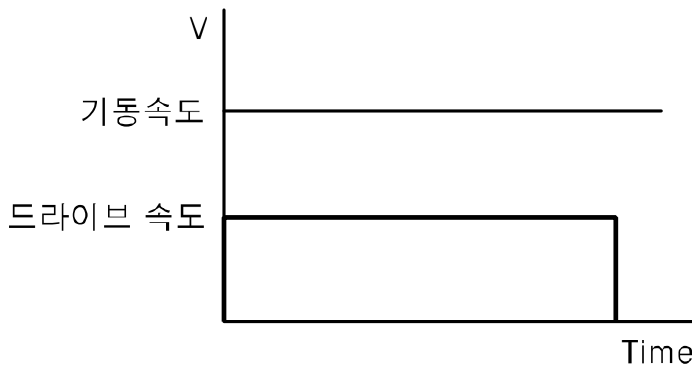
## 5.2 속도 Curve

각 축의 Drive pulse 출력은 기본적으로 +방향/-방향의 정량 Drive 명령 또는 연속 Drive 명령으로 실시하지만, 이러한 Drive 를 설정, 동작 Parameter 의 값에 의해 정속, 직선 가감속, 비대칭직선 가감속, S 자 가감속, 비대칭 S 자 가감속의 속도 커브로도 설정할 수가 있습니다.

### 5.2.1 정속 Drive

정속 Drive 는 항상 일정한 속도로 Drive pulse 를 출력합니다. 본 제품은 Drive 속도가 기동 속도보다 낮은 경우에는 가감속 Drive 는 실행되지 않고, 시작부터 일정속도 drive 가 됩니다.

원점 서치나 Encoder 의 Z 상 서치 등의 신호가 검출될 때 즉시 정지시키고 싶을 때는 가감속 Drive 를 실시하지 않고 처음부터 낮은 속도로 정속 Drive 를 실시합니다. 정속 Drive 를 실시하려면, 다음 Parameter 를 설정합니다.



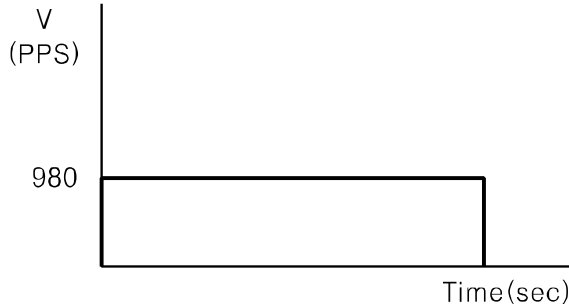
<그림 2.8 정속 Drive>

파라미터	기호	설명
Range	R	
기동속도	SV	드라이브 속도(V) 보다 높은 값을 설정
Drive 속도	V	
출력 Pulse 수	P	연속 펄스 드라이브의 때는 불필요

#### (1) Parameter 설정 예

아래와 같이 980pps 로 정속 Drive 합니다.

파라미터	기호	설정값	비고
Range	R	8,000,000	배율=1
기동속도	SV	980	기동속도 $\geq$ Drive 속도의 값을 설정
Drive 속도	V	980	-
출력 Pulse 수	P	2,450	-



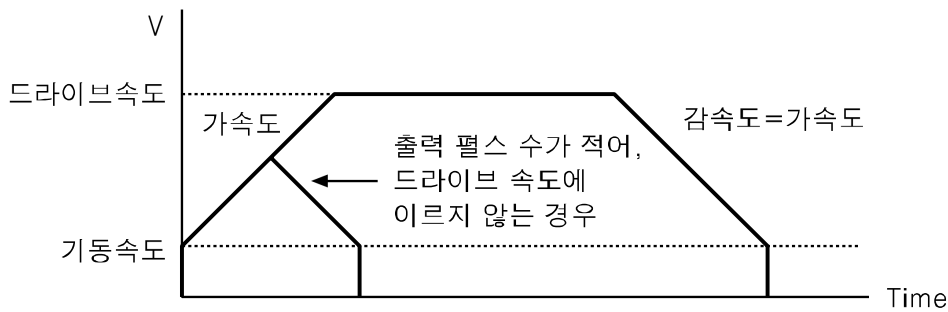
각 Parameter 에 대해서는 '6 Data 기록(Write) 명령'을 참조하십시오.

### 5.2.2 직선 가감속 Drive

직선 가감속 Drive 는 Drive 시작 시의 기동속도에서 지정된 가속도의 기울기를 가지는 일차 직선으로 Drive 속도까지 가속합니다. 정량 Drive 의 가속도와 감속도가 같은 값 (대칭 사다리꼴)인 경우는, 가속시에 소비하는 Pulse 가 출력 Pulse 의 나머지 보다 작으면 감속을 시작하여 가속도와 같은 기울기를 가지는 일차 직선으로 기동 속도까지 감속하여 모든 출력 Pulse 를 출력하면 정지합니다. (자동감속)

가속 중에 감속 정지가 발생했을 경우, 또는 정량 Drive 에서 출력 Pulse 수가 Drive 속도까지의 가속이 필요로 하는 Pulse 수에 못 미친 경우는 <그림 2.9> 와 같이 가속중에 감속합니다. 삼각방지 Mode 로 하면 출력 Pulse 수가 적어도 이러한 삼각파형을 사다리꼴 파형으로 발생 가능합니다. 가속도와 감속도는 일반적으로 같은 값으로 가속도의 값을 사용하지만 감속도를 개별로 설정할 수 도 있습니다.

감속도를 개별로 설정할 경우는 정량 Drive 의 자동 감속이 불가능해지므로 Manual 로 감속시켜 주십시오.



<그림 2.9 직선 가감속 Drive (대칭 사다리꼴)>

감속도를 개별로 설정하기 위해서는 WR3 register 의 D2~D0 bit 가 다음과 같이 설정되어야 합니다.

모드 설정 비트	기호	설정치
WR3/D0	MANLD	0
WR3/D1	DSNDE	0
WR3/D2	SACC	0

WR3 register 의 자세한 것은 '4.6 WR3 mode register 3'을 참조하십시오.

또한, 다음의 Parameter 를 설정할 필요가 있습니다.

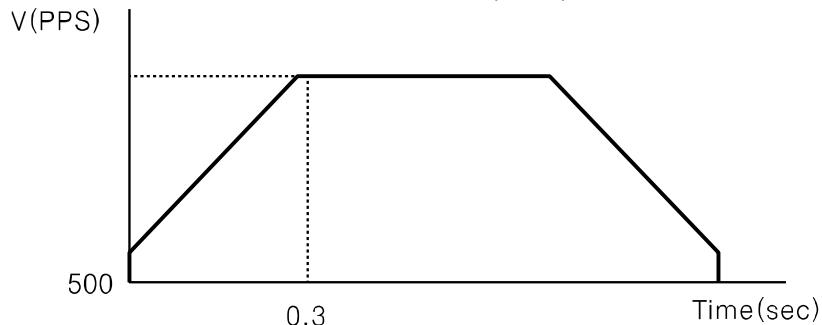
파라미터	기호	설명
Range	R	
가속도	A	감속시도 이 값으로 감속합니다.
기동속도	SV	
Drive 속도	V	
출력 pulse 수	P	연속 펄스 드라이브 시에는 불 필요

**(1) Parameter 설정 예**

아래 그림과 같이 기동속도: 500 pps, drive 속도: 15,000 pps 까지 0.3 초에 직선 가속/감속 합니다.

파라미터	기호	설정값	비고
Range	R	4,000,000	배율 = 2
가속도	A	193	$(15,000-500)/0.3 = 488,333 \text{ pps/sec}, (48333/125)/2 = 193$
기동속도	SV	250	$500/2 = 250$
Drive 속도	V	7500	$15000/2=7500$

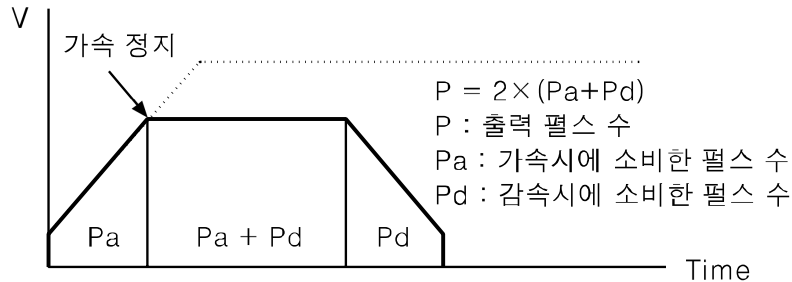
각 Parameter 에 대해서는 '6 Data 기록(Write) 명령'을 참조하십시오.



**(2) 정량 Pulse drive 의 삼각방지**

삼각방지 기능은 직선 가감속의 정량 Pulse drive 에 대해서 출력 Pulse 수가 적을 때 발생하는 삼각파형을 방지하는 기능입니다. 가속중에 소비하는 펄스가 가속시와 감속시에 소비하는 Pulse 의 출력 Pulse 수의 1/2 보다 크면 가속을 정지하고 정속으로 들어갑니다. 따라서 출력 Pulse 수가 아무리 적어도 출력 Pulse 수의 1/2 이 정량 Drive 됩니다. 삼각 파형 방지 기능은 Reset 후에는 적용되지 않습니다. 확장 Mode 설정 명령 (60h) 의 WR6/D3 (AVTRI) bit 를 1 로 설정하면 유효가 됩니다.

확장 Mode 설정 명령의 자세한 내용은 '6.16 확장 Mode 설정'을 참조하십시오.

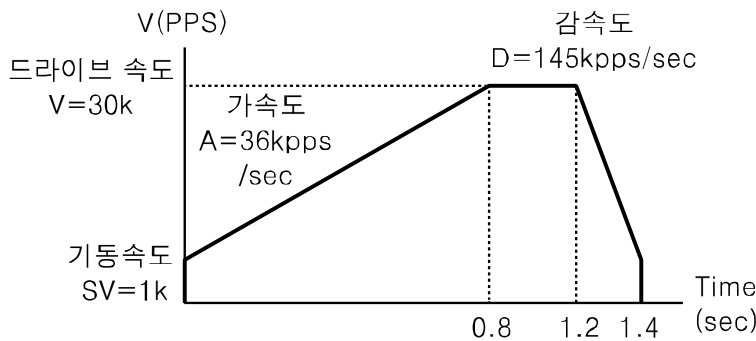


<그림 2.10 직선 가감속 Drive 의 삼각 파형 방지>

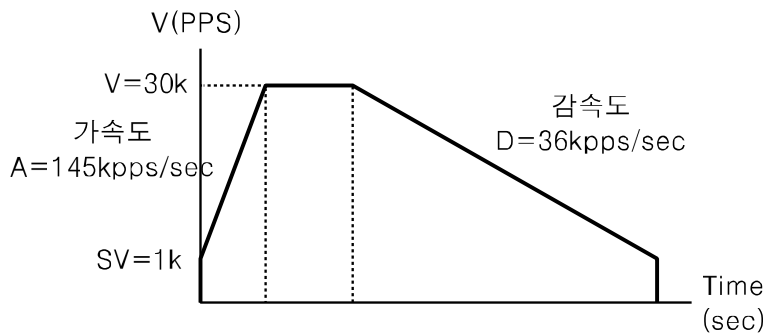
### 5.2.3 비대칭 직선 가감속 Drive

다양한 작업을 수행하는 디바이스 중에서 수직방향으로 대상물을 움직이는 경우, 대상물에 대해서 중력 가속도의 영향으로 상하 이동의 가속도와 감속도를 바꾸고 싶은 경우가 있습니다. 가속도와 감속도가 다른 비대칭 직선 가감속의 정량 Drive 에 대해서도 자동 감속이 가능합니다. 미리 계산된 매뉴얼 감속점을 설정하여 <그림 2.11>과 같이 가속도보다 감속도가 큰 예 <그림 2.12>는 감속도보다 가속도가 큰 예를 보입니다.

이러한 비대칭의 직선 가감속에 대해서 출력 Pulse 수 P 와 각 속도 Parameter 로부터 감속 시작점을 설정합니다.



<그림 2.11 비대칭 직선 가감속 Drive (가속도<감속도)>



<그림 2.12 비대칭 직선 가감속 Drive (가속도>감속도)>

비대칭 직선 가감속의 정량 Pulse drive 를 자동 감속 시키려면, WR3 register 의 D2~D0 bit 를 다음과 같이 설정합니다.

모드 설정 비트	기호	설정값	설명
WR3/D0	MANLD	0	자동 감속
WR3/D1	DSNDE	0	감속 시에 감속도 설정값을 사용합니다.
WR3/D2	SACC	0	직선 가감속



또한, 다음의 Parameter 를 설정할 필요가 있습니다.

파라미터	기호	설명
Range	R	
가속도	A	
감속도	D	
기동속도	SV	
Drive 속도	V	
출력 pulse 수	P	연속 펄스 드라이브 시에는 불 필요



**Note**

- 가속도>감속도  
<그림 2.12>의 경우, 가속도와 감속도의 비율에 대하여 다음과 같은 조건이 있습니다.

$$D > A \times \frac{V}{4 \times 10^6}$$

D: 감속도 (pps/sec.), A: 가속도 (pps/sec.), V: 드라이브 속도 (pps), CLK=16MHz



**Ex.**

drive 속도 V=100 kpps 로 하면 감속도 D 는 가속도 A 의 값의 1/40 보다 큰 값으로 하지 않으면 안됩니다.

가속도>감속도:

<그림 2.12>의 경우, 가속도 A 와 감속도 D 의 비율이 커지면 감속시 Pulse 가 많아져 Drive 속도에 도달하기 전에 감속 Pulse 가 발생할 수 있습니다. 이러한 경우에는 ①기동속도를 올리거나 ②가속 Counter offset 에 마이너스 값을 설정합니다.

- Parameter 설정 예

<그림 2.11>의 비대칭 직선 가감속 (가속도<감속도) 정량 Pulse drive 의 Parameter 설정은 다음과 같습니다.

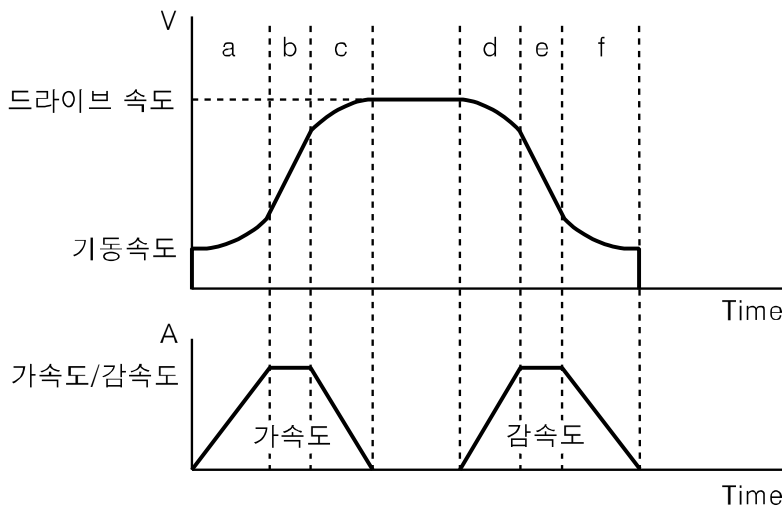
파라미터	기호	설정값	비고
WR3 ← 0002h			WR3 register 의 Mode 설정
Range	R	800,000	배율=10
가속도	A	29	(30,000-1,000)/0.8=36,250 pps/sec, (36250/125)/10=29
감속도	D	116	(30,000-1,000)/0.2=145,000pps/sec (145,000/125)/10=116
기동속도	SV	100	1,000/10=100
Drive 속도	V	3,000	30,000/10=3,000
출력 Pulse 수	P	275,000	

### 5.2.4 S자형 가감속 Drive

Drive 속도의 가속 및 감속 시 가속도/감속도를 일차 직선으로 증가/감소시키는 것으로 속도의 S자 커브를 만들 수 있습니다. 가속과 감속이 대칭형 S자 가감속 Drive 는 <그림 2.13>과 같은 동작을 실시합니다.

가속 시, 가속도가 0에서 지정된 가가속도(K)로 직선 증가합니다. 따라서 이 때의 속도 커브는 2차의 포물선 곡선이 됩니다. (a 구간) 가속도가 (A)로 되면 가속도는 그대로 유지합니다. 이 때의 속도 커브는 직선으로 가속됩니다. (b 구간) Drive 의 속도(V)와 현재 속도와의 차가 가속도를 증가했을 때에 소비하던 Pulse 수보다 적어지면 가속도는 0으로 감소합니다. 감소의 비율은 증가할 때와 같으며 지정된 가가속도(K)에서 직선으로 감소합니다. 이 때의 속도 커브는 2차 포물선이 됩니다. (c 구간) 이와 같이 가속에 있어서 가속도가 일정하게 되는 부분을 가지는 경우 부분 S자 가속이라고 합니다. 한편 a 구간에서 가속도가 지정치(D)에 도달하기 전에 Drive 속도 (V)와 현재 속도와의 차가 가속을 증가하였을 때, 소비한 Pulse 수보다 적어지면 b 구간은 없어지고 a에서 c구간으로 연결됩니다.

이와 같이 가속도에 있어서 가속도가 일정하게 되는 부분 (b 구간)이 없는 가속을 완전 S자 가속이라 합니다.



<그림 2.13 S자 가감 속도 Drive>

S자 가감속 Drive 를 실시하려면 nWR3 register 의 D2, D1, D0 bit 를 아래 표와 같이 설정합니다.

모드 설정 비트	기호	설정치	설명
WR3/D0	MANLD	0	자동 감속
WR3/D1	DSNDE	0	감속 시에 가속도, 가속도 증가율 설정값을 사용합니다.
WR3/D2	SACC	1	S자 가감속

또한, 다음의 Parameter 를 설정할 필요가 있습니다.

파라미터	기호	설명
Range	R	
가속도 증가율	K	
가속도	A	반드시 최대값 8,000 으로 설정합니다.※1
기동속도	SV	
Drive 속도	V	
출력 pulse 수	P	연속 펄스 드라이브 시에는 불 필요

※1. 가속도를 낮게 설정하면 S 자 가속시의 가속도 증가 및 감속도 증가에 있어 가/감속도가 설정된 값(A)까지 상승하지 못합니다.

### (1) 정량 Pulse drive 에서의 삼각파형 방지기능

가속과 감속이 대칭형 S 자 가감속의 정량 Drive 에서는 출력 Pulse 가 drive 속도까지의 가속에 사용된 Pulse 수와 Drive 속도에서 기동 속도까지 감속하는데 사용된 Pulse 의 수가 같을 경우의 속도 커브는 삼각파형이 됩니다.

기동 속도를 0 으로 했을 때 가속도를 가가속도로 시간 t 까지 증가시킵니다.

이 때, 시간 t 에 있어서 속도는  $V(t)=at^2$  로 나타낼 수 있습니다. 따라서 0 에서부터 시간 t 까지 소비하는 Pulse 수는 0 에서부터 시간 t 까지 속도 v(t)를 적분한 값이기 때문에  $P(t)=1/3 \times at^2$  됩니다.

이 값은 가가속도의 값에 관계없이  $at^2 \times t$  의 1/3 인 것을 나타내고 있습니다.

정량 Drive 에 있어서 0 에서 시간 t 까지 가속도를 어느 가가속도로 증가시켜 시간 t 에서와 같은 가가속도로 감속시킵니다. 가속도가 0 이면 감속시에서도 같은 가가속도로 감속을 하여 전체 소비되는 Pulse 수는 다음과 같이  $1/3 + 2/3 + 1 + 2/3 + 1 + 1/3 = 4$  의 Pulse 수가 됩니다.



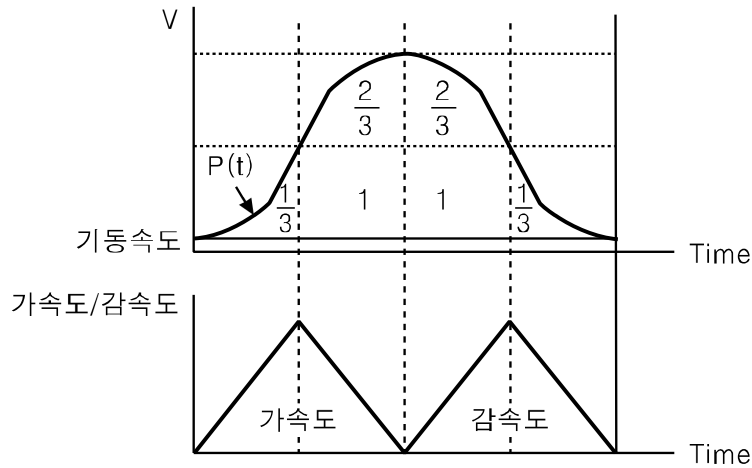
### Note

가속도를 낮게 설정하면 S 자 가속 시의 가속도 증가 및 감속도 증가에 대해서 가/감속도가 설정된 값(A)이상으로 상승하지 않고 속도 커브가 직선 부분으로 나타납니다. 이 값은 가속도 증가율의 값에 관계없이  $at^2 \times t$  (그림 중의 1 의 Pulse 수)의 1/3 인 것을 나타내고 있습니다.

정량 Pulse drive 에서 0 에서 시간 t 까지 가속도가 해당하는 가속도 증가율로 증가시켜 시간 t 로부터 같은 가속도 증가 비율로 가속도를 감속 시킵니다. 가속도가 0 되면 감속시에도 같은 가속도 증가율로 감소를 실시하고 전체 소비되는 Pulse 수는 <그림 2.14>와 같은 Pulse 수가 됩니다.

따라서 기동시간 0 에서부터 시간 t 까지의 Pulse 수 (1/3)는 전체 Pulse 수의 1/12 이상으로 S 자 가감속 정량 Pulse drive 는 가속도 증가 시 Pulse 가 총 출력 Pulse 의 1/12 보다 커지면 가속도를 감소하여 <그림 2.14>와 같은 속도 커브를 그립니다. [1/12 법칙]

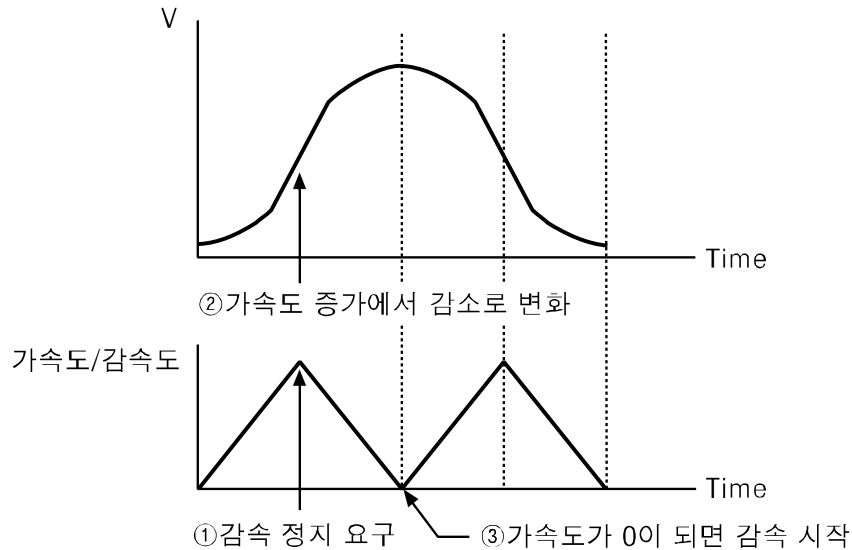
그러나 이 방식은 엄밀하게 기동속도=0 일 때 이상적인 커브가 됩니다. 기동속도는 실제로 0 에는 시작할 수 없기 때문에, 그림 중의 속도 0 에서부터 기동속도까지의 Pulse 수가 남게 되어 이 부분은 피크 속도 시에 출력되게 됩니다.



<그림 2.14 포물선 가감 속도의 1/12 법칙>

**(2) 감속 정지에서의 삼각파형 방지기능**

직선 가감속 Driver 에서는 가속시에 감속 정지를 시켰을 때는 속도 커브가 삼각파형이 됩니다. S 자 가감속 Drive 에서는 속도 커브의 부드러움이 중요하기 때문에 <그림 2.15>와 같이 가속시에 감속 정지가 걸린 경우, 바로 감속하지 않고 가속도를 일단 0 까지 감속시킨 다음 감속을 실행합니다.



<그림 2.15 포물선 가감 속도의 1/12 법칙>

**(3) S 자 가감속 Drive 시 주의사항**

- S 자 가감속의 정량 Drive 에 있어서 Drive 속도는 Drive 도중에 변경할 수 없습니다.
- S 자 가감속의 정량 Drive 에 있어서 감속 중에 출력 Pulse 수를 변경하면 올바른 S 자 커브를 그릴 수 없습니다.
- 원호보간, Bit pattern 보간, 연속 보간은 S 자 가감속에서는 Drive 할 수 없습니다.
- S 자 가감속의 정량 Drive 에서는 기동 속도를 너무 낮게 설정하면 감속 시에 기동 속도까지 떨어지기 전에 Drive pulse 를 종료하거나 기동 속도까지 도달해도 기동 속도로 나머지의 Drive pulse 출력하는 현상이 발생하는 경우가 있습니다.

**(4) Parameter 설정의 예 (대칭 S 자 가감속)**

그림과 같이 기동속도 100pps 로 Drive 속도 40kpps 까지 0.4 초 동안 S 자 가속을 하는 예입니다. 가속시에는 일정한 가가속도 (k)에 따라 가속도를 직선 증가시키면 이 때의 적분치 (사선의 면적)가 상승하는 속도 형태가 됩니다.

가속시간(t=0.4sec.)의 반(t/2)으로 정확히 속도가 기동속도(SV)로부터 드라이브 속도(V)의 반<(V-SV)/2>이 되는 가가속도(k)를 구하는 식은 좌변의 k 를 사용한 사선부의 면적이 우변과 동일한 가를 알아보면 됩니다. K 를 요구하는 식은 다음과 같습니다.

$$\frac{k}{2} \left(\frac{t}{2}\right)^2 = \frac{V-SV}{2}$$

$$k = \frac{4(V-SV)}{t^2}$$

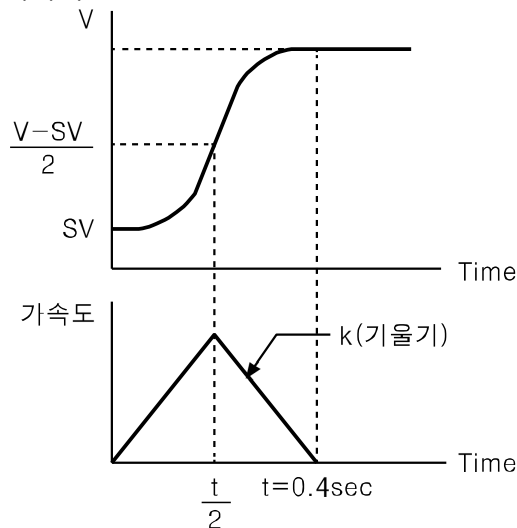
$$k = \frac{4(40,000-100)}{0.4^2} = 997,500 \text{ pps/sec}^2$$

가속도 증가율 k: pps/sec<sup>2</sup>

드라이브 속도 V: pps

기동속도 SV: pps

가속시간 t: sec

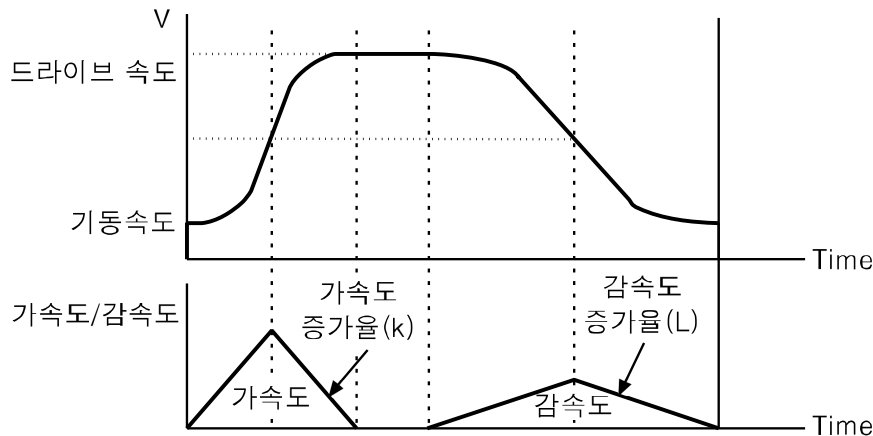


Parameter 설정은 다음과 같습니다.

파라미터	기호	설정값	비고
WR3 ← 0004h			WR3 register 의 Mode 설정
Range	R	800,000	배율=10
가가속도	K	627	$(62.5 \times 106/k) \times \text{배율} = (62.5 \times 106/997,500) \times 10$
가속도	A	8,000	최대치에 고정
기동속도	SV	10	$100/10=10$
Drive 속도	V	4,000	$40,000/10=4,000$
출력 Pulse 수	P	25,000	정량 Pulse drive 의 경우 설정합니다.
가속 Counter offset	A0	0	

### 5.2.5 비대칭 S 자 가감속 Drive

<그림 2.16>과 같이 S 자 가감속 Drive 에 대해서 가속시 가가속도와 감속시의 가가속도를 다르게 설정하여 비대칭의 S 자 커브를 만들어 낼 수가 있습니다. 정량 Drive 의 경우 대칭 S 자 가감속도 Drive 는 자동으로 감속할 수 없기 때문에 매뉴얼 조작(수동)으로 감속점을 지정할 필요가 있으며 가/감속시의 가가속도 출력 Pulse 수의 값에 대한 드라이브 속도를 설정할 필요가 있습니다.



<그림 2.16 비대칭형 S 자 가감속 Drive>

비대칭형 S 자 가감속 Drive 를 실행하려면 nWR3 register 의 D2, 1, 0 bit 를 아래와 같이 설정합니다.

모드 설정 비트	기호	설정치	설명
WR3/D0	MANLD	1	매뉴얼 감속
WR3/D1	DSNDE	1	감속시에 가속도, 가속도 증가율 설정치를 사용합니다.
WR3/D2	SACC	1	S 자 가감속

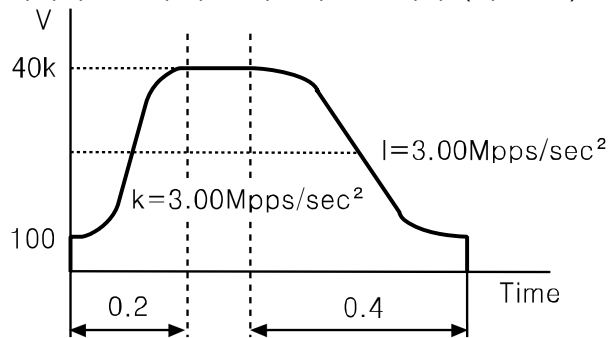
Parameter 설정은 다음과 같습니다.

파라미터	기호	설명
Range	R	
가속도 증가율	K	
감속도 증가율	L	
가속도	A	반드시 최대값 8,000 으로 설정합니다.
감속도	D	반드시 최대값 8,000 으로 설정합니다.
기동속도	SV	
Drive 속도	V	
출력 pulse 수	P	연속 펄스 드라이브 시에는 불 필요
매뉴얼 감속점	DP	출력펄스(P) 중 감속 시에 소비되는 펄스 수를 계산하여 설정합니다. 연속 펄스 드라이브 시에는 불필요

**(1) Parameter 설정의 예 (비대칭 S자 가감속)**

그림과 같이 가속 시에는 기동속도(SV) 100pps 로 Drive 속도(V) 40kpps 까지를 0.2 초에 가속시키고 감속시에는 Drive 속도 (V) 40kpps 에서 기동속도(SV) 100pps 까지 0.4 초에 감속시키는 비대칭 S자 가감속의 예입니다.

대칭 S자 가감속 파라미터 설정 예의 식을 사용하여 가속시의 가가속도, 감속시의 가가속도를 구하면 다음과 같습니다. (배율: 10)



$$\text{가속도의 증가율 } k = \frac{4(40,000-100)}{0.2^2} = 3.99\text{Mpps/sec}^2$$

$$\text{감속도의 증가율 } l = \frac{4(40,000-100)}{0.4^2} = 0.9975\text{Mpps/sec}^2$$

Parameter 설정은 다음과 같습니다.

$$\text{가속도의 증가율 } k = \frac{62.5 \times 10^6}{0.2^2} \times \text{배율} = \frac{62.5 \times 10^6}{3.99 \times 10^6} \times 10 = 157$$

$$\text{감속도의 증가율 } l = \frac{62.5 \times 10^6}{l} \times \text{배율} = \frac{62.5 \times 10^6}{0.9975 \times 10^6} \times 10 = 627$$

비대칭 S자 가감속에서는 자동 감속을 할 수 없기 때문에 매뉴얼 감속점(DP)을 설정합니다.

매뉴얼 감속점은 출력 Pulse 수(P)로부터 감속시 소비되는 Pulse(Pd)를 설정해야 하므로, 감속 시 소비 Pulse(Pd)를 산출합니다.

$$\text{감속시 소비 Pulse(Pd)} = (V + SV) \sqrt{\frac{V-SV}{I}} = (40000 + 100) \sqrt{\frac{40000-100}{0.9975 \times 10^6}} = 8020$$

출력 Pulse 수를 20,000 으로 하면 매뉴얼 감속점(DP)은 다음과 같습니다.

$$\text{매뉴얼 감속점 DP} = P - Pd = 20000 - 8020 = 11980$$

따라서 Parameter 설정은 다음과 같습니다.

파라미터	기호	설정값	비고
WR3 ← 0007h			WR3 register 의 Mode 설정
Range	R	800,000	배율=10
가속시 가가속도	K	157	$(62.5 \times 106/k) \times \text{배율} = (62.5 \times 106 / 3.99 \times 106) \times 10$
감속시 가가속도	L	627	$(62.5 \times 106/k) \times \text{배율} = (62.5 \times 106 / 0.9975 \times 106) \times 10$
가속도	A	8,000	최대치에 고정
감속도	D	8,000	최대치에 고정
기동속도	SV	10	$100/10=10$
Drive 속도	V	4,000	$40,000/10=4,000$
출력 Pulse 수	P	20,000	정량 Pulse drive 의 경우 설정합니다.
매뉴얼 감속점	DP	11,980	
가속 Counter offset	A0	0	

## 5.2.6 Drive pulse 폭과 속도 정밀도

### (1) Drive pulse 의 Pulse 비율

각 축의 +방향/-방향의 Drive pulse 에 있어서 Drive 속도에 의해서 결정되는 Pulse 주기의 시간은 연산상의 오차  $\pm 1\text{SCLK}$ (CLK=16MHz 일 때  $\pm 125\text{ nsec}$ ) 이지만, 기본적으로 High level 과 Low level 에 50%씩 나누어 집니다.



**Ex.**

아래 그림과 같이 R=8,000,000, V=1,000 (배율=1, Drive 속도=1,000pps)으로 설정하면 Drive pulse 는 High level 폭=500  $\mu\text{s}$ , 주기=1.00ms 의 Pulse 를 출력합니다.

R=8000000
SV=1000
V=1000



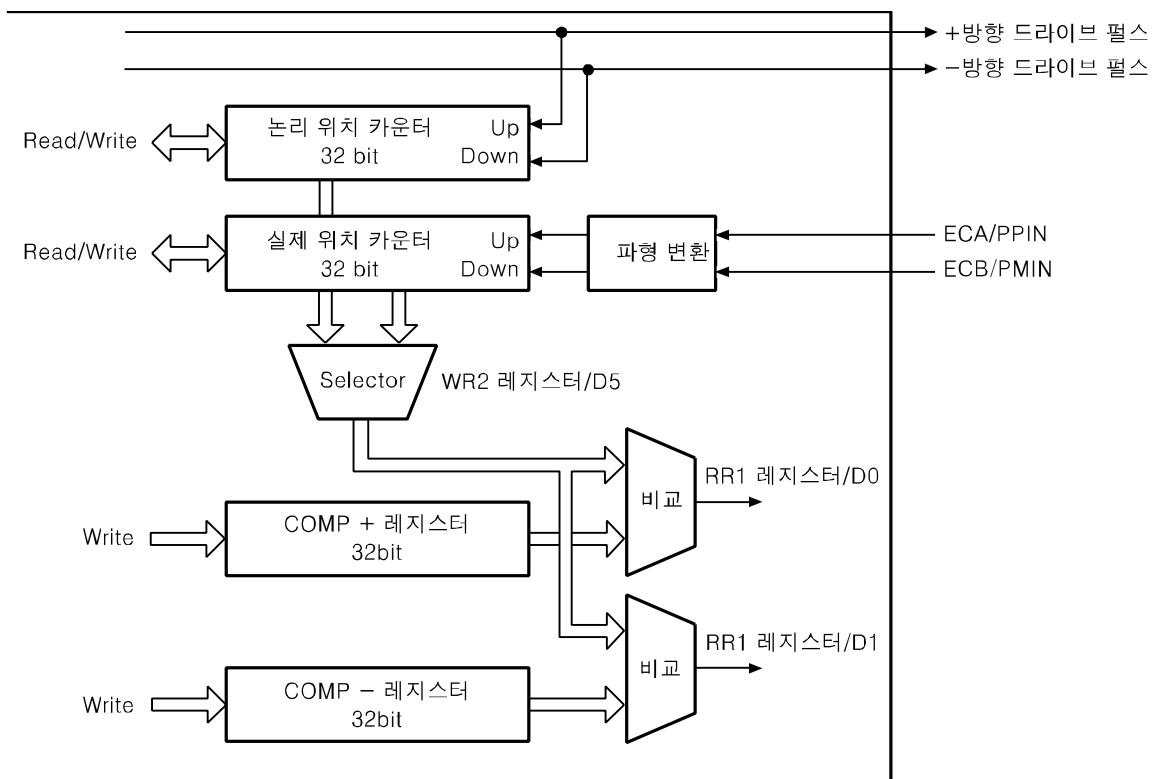


이 방식에 의해 지정된 속도의 Drive pulse 의 정도를 좋게 출력할 수가 있습니다. 속도 배율을 올려도, 지정한 속도에 대한 실제로 출력되는 Drive pulse 의 속도 정도는 ±0.1% 이하로 유지됩니다.

Drive pulse 를 오실로스코프 (Oscilloscope)로 관측하면 Drive pulse 의 주기가 SCLK 주기의 정수배가 아닐 때에는 <그림 2.19>와 같이 Pulse 주기에 1SCLK (125nsec)의 시간차이가 생기기 때문에 이것이 Jitter 와 같이 보이지만 1SCLK 의 시간차이는 모터를 돌리는 경우 부하의 관성이 흡수되어 대부분 문제가 되지 않습니다.

### 5.3 포지션 (Position) 관리

<그림 2.20>은 1 축에 해당하는 포지션 관리부의 블록도 입니다. 각 축 모두 현재 위치 관리를 위한 32bit Up/Down counter 2 개와 현재 위치의 대소비교를 위한 비교 Register 2 개를 가지고 있습니다.



<그림 2.20 포지션 관리부 블록도>

#### 5.3.1 논리 위치 Counter와 실제위치 Counter

논리위치 Counter 는 <그림 2.20>과 같이 +방향/-방향의 Drive 출력 Pulse 를 Count 합니다. +방향 1Pulse 로 1Count up, -방향 1Pulse 로 1Count down 합니다.

실제 위치 Counter 는 Encoder 등의 외부입력 Pulse 를 Count 합니다. 입력 Pulse 를 2 상 펄스 입력 신호로 할지, 독립 2Pulse(Up/Down)펄스 입력 신호로 할지를 Command 로 선택할 수가 있습니다. ‘2.9.3 Pulse 입력 방식의 선택’을 참조하십시오.

Counter 는 Data 읽기/쓰기가 가능하며 Count 범위는 -2,147,483,648~+2,147,483,649 입니다.

### 5.3.2 비교 Register와 소프트웨어 리미트

각 축은 <그림 2.20>과 같이 논리위치 Counter 또는 실제위치 Counter 와 대소 비교를 할 수 있는 2 개의 32bit 레지스터 (COMP+, COMP-)를 가지고 있습니다. 2 개의 비교 Register 의 비교 대상을 논리위치 Counter 로 할지, 실제위치 Counter 로 할지는 WR2 register 의 D5(CMPSEL) bit 로 선택합니다.

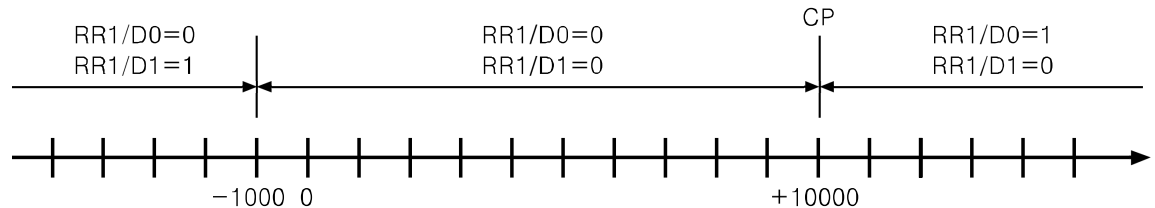
COMP+ register 는 논리/실제위치 Counter 에 대해서 어느 범위의 상한을 검출하기 위한 Register 로서 논리/실제위치 Counter 값이 COMP+ register 의 값보다 커지면 RR1 register 의 D0 (CMP+) bit 가 1 이 됩니다.

COMP- register 는 논리/실제위치 Counter 에 대해서 어느 범위의 하한을 검출하기 위한 Register 로서 논리/실제위치 Counter 가 COMP- register 의 값보다 작아지면 RR1 register 의 D1(CMP-) bit 가 1 이 됩니다.



**Ex.**

COMP + register=10,000, COMP- register= -1,000 을 설정한 예 입니다.



<그림 2.21 COMP+/- register 설정 예>

COMP+ register 와 COMP- register 를 각각 +방향/-방향의 소프트웨어 Limit 로서 사용할 수 있으며 WR2 register 의 D0, D1 (SLMT+, SLMT-) bit 를 1 로 소프트웨어 Limit 를 유효하게 하면 Drive 중에 논리/실제위치 Counter 가 COMP+ register 보다 커지면 감속 정지하여 RR2 register 의 D0 (SLMT+) bit 가 1 이 됩니다. 이 예러는 - 방향의 Drive 명령을 실행하여 논리/실제위치 Counter 가 COMP+ register 보다 작아지면 해제됩니다.

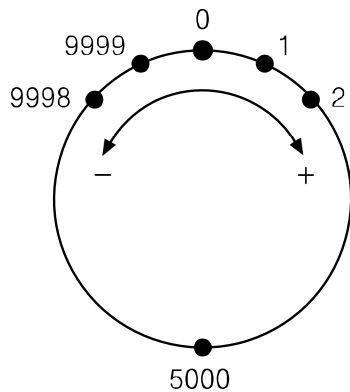
COMP- register 의 -방향에 대해서도 같습니다.

COMP+ register 와 COMP- register 는 언제든지 설정이 가능합니다. Reset 후 내용은 적용되지 않습니다.

### 5.3.3 위치 Counter의 가변 링

논리위치 Counter 및 실제위치 Counter 는 32bit 의 Up/Down counter 입니다.

따라서 일반적으로 32bit 의 최대치로 FFFFFFFFh 에서 +방향으로 count up 하면 0 으로 돌아옵니다. 또한 0 의 값에서 -방향으로 Count down 하면 FFFFFFFFh 로 돌아옵니다. 가변 링 기능은 이 링 Counter 의 최대치를 임의의 값으로 설정하는 기능입니다. 위치결정축이 직선운동은 아니고, 회전운동을 하는 축의 위치관리를 하는 디바이스에 편리한 기능입니다. 가변 링 기능을 유효하게 하려면 확장 Mode 설정 명령 (60h)의 WR6 register/D4 (VRING)bit 를 1 로 설정하여 논리위치 Counter 의 최대치를 COMP+ register 에 실제위치 Counter 의 최대치를 COMP- register 에 설정합니다.



<그림 2.22 위치 Counter 링 최대치 9999 의 동작>

10,000 pulse 로 1 회 회전하는 회전축의 경우, 다음과 같이 설정합니다.

- ① 가변 링 기능을 유효하게 하기 위해서 확장 mode 설정명령 (60h)의 WR6/D4 bit 에 1 을 설정합니다.
- ② 논리위치 Counter 의 최대값으로서 COMP+ register 에 9,999(270Fh)를 설정합니다.
- ③ 실제위치 Counter 도 사용하는 경우는 COMP- register 에 9,999(270Fh)를 설정합니다.

이 때의 Count 동작은 +방향에 Count up 시에는 ...→9998→9999→0→1→... 이 됩니다.

- 방향에 Count down 시에는 ...→1→0→9999→9998→... 이 됩니다.



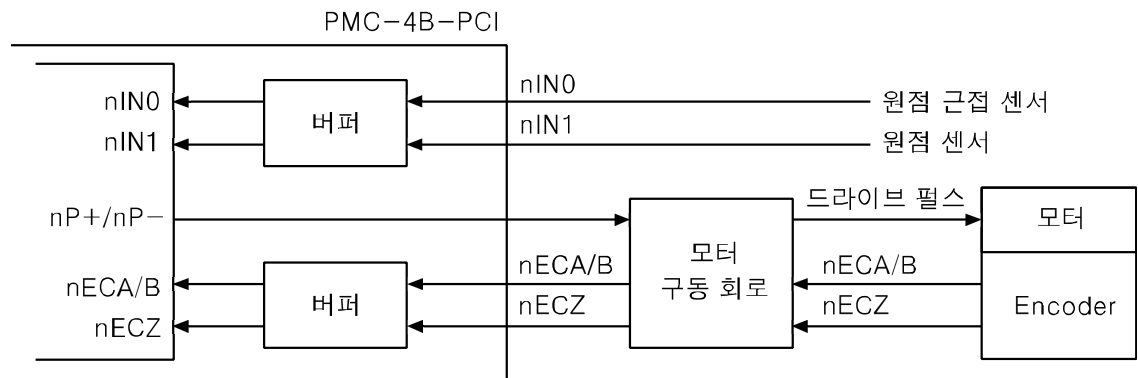
#### Note

- 가변 링 기능의 유효/무효는 각 축마다 설정가능합니다. 논리위치 Counter 와 실제위치 Counter 는 각각 개별적으로 유효/무효로 설정할 수 없습니다.
- 가변 링 기능을 유효하게 하면 소프트 리미트 기능은 사용할 수 없습니다.

### 5.3.4 외부 신호에 의한 실제위치 Counter의 Clear

원점 출력에 있어서 Z 상 서치를 실행하면 Z 상 신호가 Active 되는 Level 의 시작에서 실제위치 카운터를 Clear 시키는 기능입니다. 일반적으로 원점복귀는 원점근접신호, 원점신호, Encoder Z 상 신호 등을 nIN0, nIN1, nECZ 단자로 접속하여 연속 Drive 를 실행하는 것으로서 실시합니다. 지정된 신호가 Active 되면 Drive 가 정지하기 때문에 논리위치/실제위치 Counter 를 Clear 합니다. Z 상 서치는 저속의 Drive 속도로 서보계 혹은 기계계의 지연으로부터 발생하는 Z 상 검출의 위치 차이가 문제가 되는 경우에 본 기능을 사용하면 편리합니다.

Encoder Z 상 서치 시에 Z 상 신호에서 실제위치 Counter 를 Clear 하려면 <그림 2.23>과 같이 Z 상 신호를 nECZ 신호에 연결하고 실제위치 Counter clear 를 동반하는 Z 상 서치의 Mode 및 Command 를 다음과 같이 설정합니다.

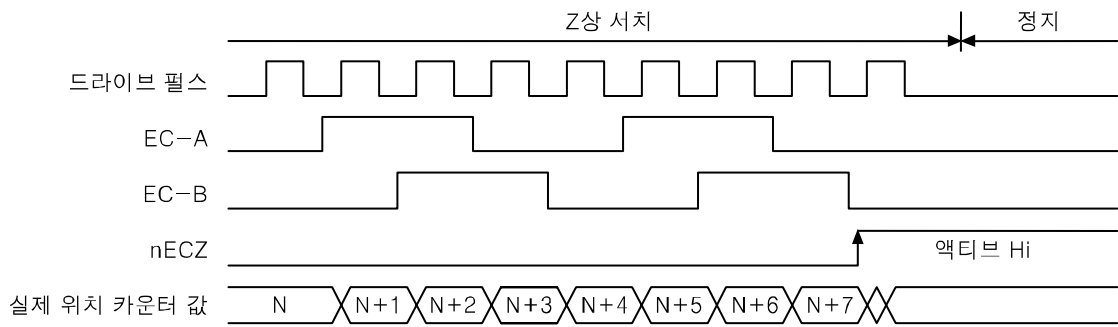


<그림 2.23 IECZ 신호에 의한 실제위치 Counter clear 신호의 접속 예>

- ① Range 와 기동속도를 설정합니다.
- ② Z 상 서치의 Drive 속도를 설정합니다. Drive 속도를 기동속도보다 낮은 값으로 설정하면 가감속 Drive 는 실행되지 않으므로 Z 상을 검출하면 Drive pulse 는 즉시 정지합니다.
- ③ nECZ 신호의 유효와 Active level 을 설정합니다.

WR1/D5 (IN2-E)	1
D4(IN2-L)	0(Low active), 1(High active)

- ④ nECZ 신호에 의한 실제위치 Counter clear 를 유효하게 설정합니다.  
WR6/D0 (EPCLR)를 1로 설정하여 확장 Mode 설정명령 (60h)을 발생합니다.  
확장 Mode 설정명령의 다른 bit 도 동시 설정됩니다.
- ⑤ +방향 또는 -방향 연속 Pulse drive 명령을 발행합니다.  
이상의 조작을 실시하면 <그림 2.24>와 같이 지정된 방향으로 Drive 를 시작하여 Z 상 신호가 Active level 이 되면 Drive pulse 가 정지됨과 동시에 실제위치 Counter 는 Z 상 신호 Active level 의 시작에서 Clear 됩니다.



<그림 2.24 IN2 신호에 의한 실제위치 Counter clear 의 동작 예>



## Note

- 실제위치 Counter 를 Clear 할 수 있는 신호는 nECZ 신호입니다. nIN3, nIN1, nIN0 신호에서는 Clear 할 수 없습니다.
- nECZ 신호의 Active level 폭은 입력 신호 필터가 무효인 경우, 4CLK 사이클 이상 필요합니다. 입력 신호 필터가 유효인 경우는 입력신호 지연의 배 이상의 시간이 필요합니다.
- Z상 서치는 위치 검출 정도를 올리기 위해서 반드시 한 방향으로 검출하는 것을 추천합니다.
- 실제 위치 Counter clear 기능을 유효하게 하기 위해서 WR6/D0 (EPCLR)를 1로 합니다. 그리고 이미 nECZ 신호가 Active level 이 되어 있는 경우는 확장 Mode 설정 명령을 발행한 시점에서 실제위치 Counter 가 Clear 됩니다.

## 5.4 보간

4 축 중 임의의 2 축 또는 3 축을 선택하여 직선보간, 원호보간, Bit pattern 보간 Drive 가 가능합니다.

보간을 실시하는 축 지정은 WR5 register 의 D0, D1(ax1), D2, D3(ax2), D4, D5(ax3)에 축 코드를 설정합니다.

보간 Drive 에서는 주축(ax1)으로 지정된 축의 기본 Pulse 타이밍에 보간 연산을 합니다. 따라서 보간명령을 발행하기 전에 주축(ax1)의 기동속도, Drive 속도 등의 Parameter 가 설정되어 있어야 합니다.

주축이란 ax1 로 지정된 축을 말하며 각각의 축으로 보간 명령에 필요한 Parameter 를 설정하여 보간 Drive 명령을 WR0 Command register 에 기입하면 보간 드라이브는 시작됩니다. 보간 Drive 중에 RR0 (주 Status register)의 D8(I-DRV) bit 가 1 이 되어 Drive 가 끝나면 0 으로 돌아옵니다.

또한 보간 Drive 중 보간을 실시하고 있는 축의 n-DRV bit 에도 1 이 설정됩니다. 보간 연산은 직선보간, 원호보간, Bit pattern 보간 모두 최고 4Mpps 까지 실시할 수 있습니다. 다만, 연속보간일 경우에는 최고 2Mpps 까지 입니다.

### (1) 보간 시의 Over run limit 등의 에러

보간 Drive 에 있어서 Drive 하는 각 축의 하드웨어 Limit, 소프트웨어 Limit 는 항상 작동합니다. 즉, 보간 Drive 중 해당 축의 Limit 가 Active 되면 보간 Drive 는 정지합니다. 에러로 정지했을 경우는 RR0 (주 Status register)의 보간이 지정되고 있는 축의 에러 Bit 를 확인하여 1 로 설정되면 그 축의 RR2 (에러 Register)를 읽을 수 있습니다.



### Note

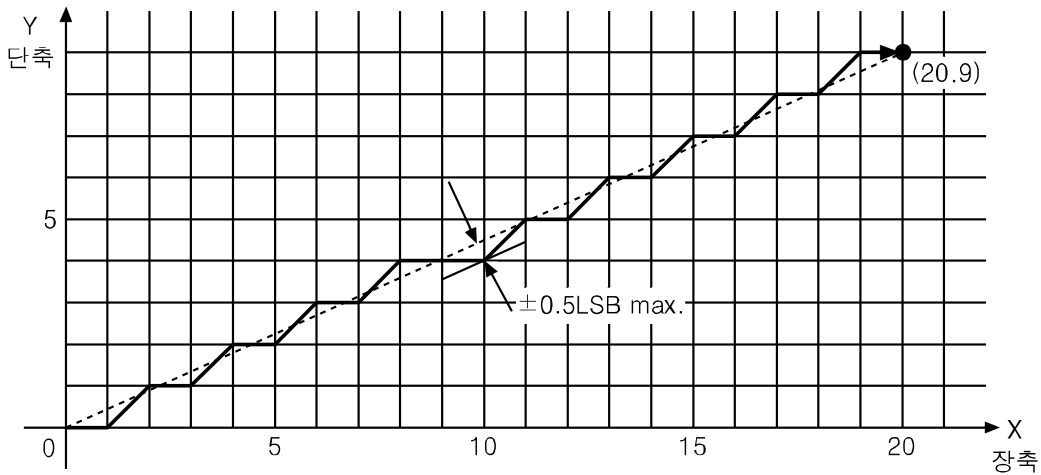
원호보간 및 Bit pattern 보간에서는 +방향/-방향 중 어느 방향의 하드웨어 Limit 및 소프트웨어 Limit 가 Active 되어도 보간이 정지하는 경우가 있습니다. 따라서 원호보간 및 Bit pattern 보간에서는 Limit 영역으로부터 탈출할 수 없기 때문에 주의하여 주십시오.

### (2) 서보 모터용 위치 결정 신호의 대응

보간 Drive 에 있어서 Drive 하는 각 축의 위치 결정신호(nINPOS)를 유효하게 하면 보간 Drive 종료 후, 모든 축의 nINPOS 신호가 Active level 이 되는 것을 확인하고 나서 RR0 register 의 D8(I-DRV) bit 가 0 으로 돌아옵니다.

### 5.4.1 2 축/3 축 직선보간

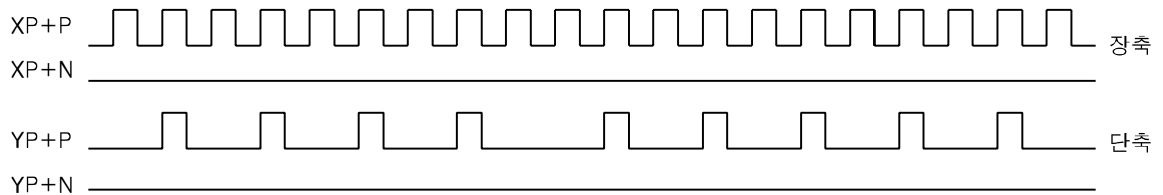
4 축 중 임의의 2 축 또는 3 축을 선택하여 직선보간 Drive 를 실시합니다. 직선보간은 현재 좌표에 대한 종점 좌표를 설정하여 2 축 또는 3 축 직선보간 명령을 발생하여 실행됩니다. <그림 2.25>는 2 축 보간의 예 입니다. 종점 좌표는 현재 위치에 대한 상대치로 각축의 출력 Pulse 수로 설정합니다. 출력 Pulse 수는 각 축 독립으로 움직일 때는 부호없는 값으로 설정하지만 보간 Drive 시에는 현재 위치에 대한 종점좌표를 상대치로 설정하기 때문에 주의하여 주십시오. 지정 직선에 대한 위치 정도는 <그림 2.25>와 같이 전 보간 범위내에서  $\pm 0.5\text{LSB}$  입니다.



<그림 2.25 직선 보간의 위치 정밀도>

<그림 2.26>은 직선 보간의 Drive pulse 출력 예입니다. 설정된 종점의 값중 절대치가 큰 축이 장축이 되고 보간 Drive 중에는 항상 Pulse 를 출력합니다. 다른 축은 단축되어 직선 보간연산 결과에 의해 Pulse 를 발생할 때 발생하지 않을 때가 있습니다. 직선 보간의 좌표 범위는 부호를 첨부하여 32bit 입니다.

각 축 모두 현재 위치로부터 -2,147,483,646~+2,147,483,646 의 범위에서 보간을 할 수 있습니다.



<그림 2.26 종점 (X: 20, Y:9) 의 Drive pulse 출력의 예>



**(1) 2 축 직선보간 Drive 의 예**

X, Y 축에 대하여 현재위치에서 종점좌표 (X: +300, Y:-200) 까지 직선보간 합니다. 보간 Drive 속도는 1,000pps 의 정속 Drive 입니다.

WR5←0004h write ax1: X 축, ax2: Y 축 지정  
 WR6←1200h write Range: 8,000,000 (배율:1)  
 WR7←007Ah write  
 WR0←0100h write  
 WR6←03E8h write 기동속도: 1,000pps  
 WR0←0104h write  
 WR6←03E8h write drive 속도: 1,000pps  
 WR0←0105h write  
 WR6←012Ch write 종점 X 축: 300  
 WR7←0000h write  
 WR0←0106h write  
 WR6←FF38h write 종점 Y 축: -200  
 WR7←FFFFh write  
 WR0←0206h write  
 WR0←0030h write 2 축 직선 보간 Drive

**(2) 3 축 직선보간 Drive 의 예**

X, Y, Z 축에 대하여 현재 위치에서 종점좌표 (X: 15000, Y: 16000, Z: 20000)까지 3 축 직선보간 합니다.

기동속도: 500pps, 가감속도: 40,000pps/sec, Drive 속도: 5,000pps 의 직선 가감속 Drive 입니다.

WR5←0024h write ax1: X 축, ax2: Y 축 지정, ax3: Z 축 지정  
 WR6←1200h write Range: 8,000,000 (배율:1)  
 WR7←007Ah write  
 WR0←0100h write  
 WR6←0140h write 가감속도: 40,000pps/sec  
 WR0←0102h write 40,000/125/1 =320  
 WR6←01F4h write 기동속도: 500pps  
 WR0←0104h write  
 WR6←1388h write drive 속도: 5000pps  
 WR0←0105h write  
 WR6←3A98h write 종점 X: 15,000  
 WR7←0000h write  
 WR0←0106h write  
 WR6←3E80h write 종점 Y: 16,000  
 WR7←0000h write  
 WR0←0206h write  
 WR6←4E20h write 종점 Z: 20,000

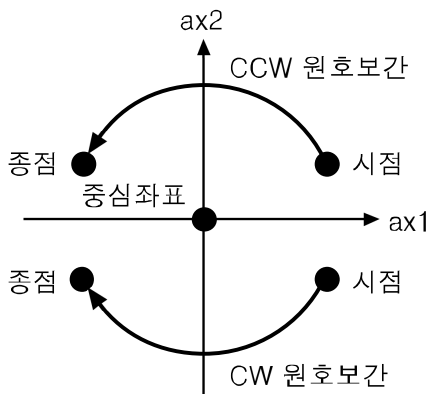
- WR7←0000h write
- WR0←0406h write
- WR0←003Bh write      감속 유효
- WR0←0031h write      3 축 직선보간 Drive

### 5.4.2 원호보간

4 축 중 임의의 2 축을 선택하여 원호보간 Drive 를 실시합니다. 원호보간은 현재 좌표에 대한 원호의 중심 좌표 및 종점 좌표를 설정하여 CW 원호보간 명령 또는 CCW 원호보간 명령을 기입하는 것으로 실행됩니다.

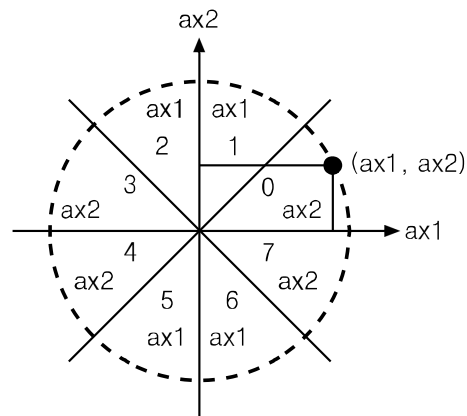
중심좌표 및 종점좌표의 지정은 현재좌표에 대한 상대치로 설정해야 합니다.

CW 원호보간은 현재 좌표에서 종점좌표까지 중심 좌표를 중심으로 시계방향으로 실시하고 CCW 원호보간은 반시계 방향으로 원호를 그립니다. 종점을 (0,0)으로 하면 진원을 그릴 수가 있습니다.

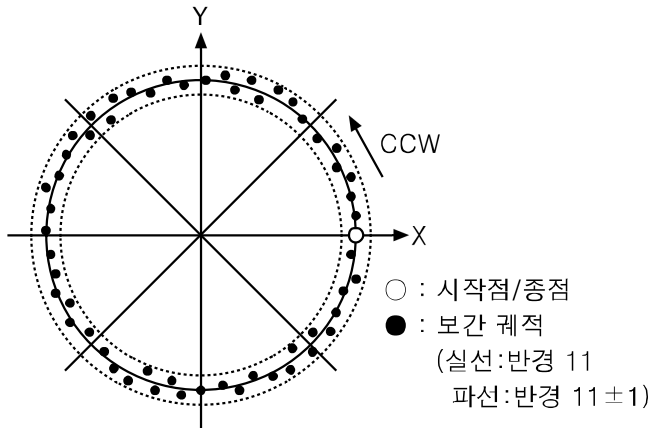


<그림 2.27 CW/CCW 원호보간>

PCM-4B-PCI 의 원호보간의 연산에서는 <그림 2.28>과 같이 제 1 축(ax1)과 제 2 축(ax2)에 의한 평면의 중심 좌표를 중심으로 0~7, 총 8 개의 상한으로 나누고 있습니다. 그림에서 0 상한에서는 원호상을 이동하는 보간 좌표(ax1, ax2)는 항상 ax2 의 절대치가 ax1 의 절대치보다 작습니다. 절대치값이 작은축을 단축으로 1, 2, 5, 6 상한에서는 제 1 축(ax1)이 단축이 되고, 0, 3, 4, 7 상한은 제 2 축 (ax2)이 단축이 됩니다.



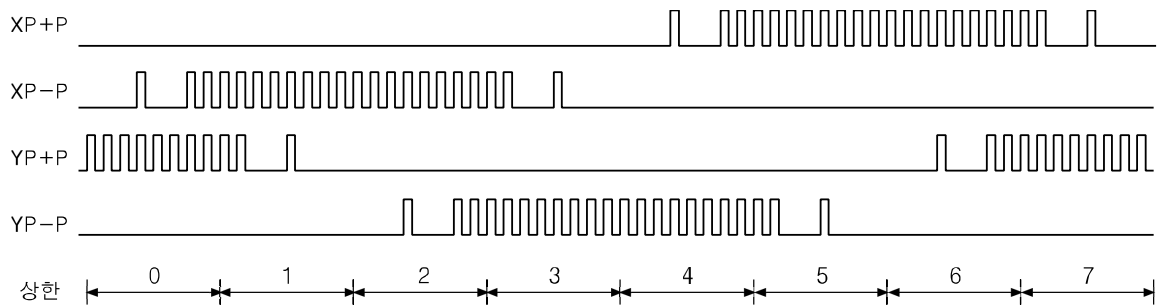
<그림 2.28 원호보간 연산의 0~7 상한과 단축>



<그림 2.29 원호보간>

<그림 2.29>는 현재 좌표 중심 (-11, 0), 종점 (0, 0)을 지정으로, 반경 11의 진원을 그린 예입니다. 또 <그림 2.30>은 이 때의 Drive pulse 출력을 나타냅니다.

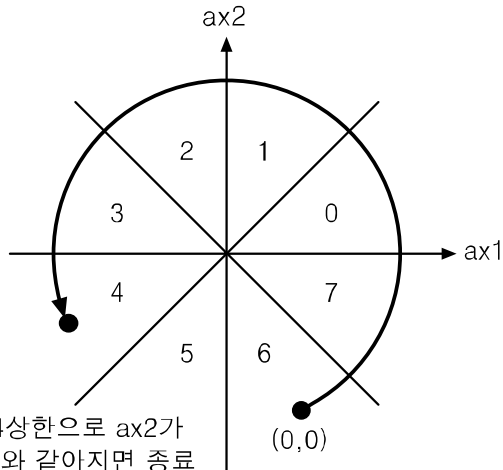
중심좌표 및 종점좌표의 지정범위는 현재 위치로부터 -2,147,483,646~+2,147,483,646입니다. 원호에 대한 위치 오차는 전 보간 범위 내에서 ±1LSB입니다. 보간속도는 1pps~4Mpps입니다.



<그림 2.30 원호보간 Drive pulse 출력의 예>

**(1) 종점 판정**

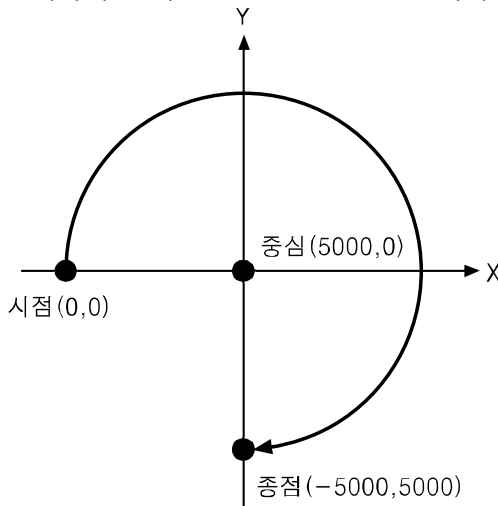
원호보간은 보간 Drive 를 시작하기 전 현재좌표와 중심좌표로 설정되어 반경이 결정되면 원호의 궤적을 그립니다. 원호 연산의 오차는 보간 좌표 범위 안에서 ±1LSB이기 때문에 지정된 종점이 반드시 원호의 궤적상에 있다고는 할 수 없습니다. 그래서 종점이 있는 상한에 대해서 종점의 단축의 값과 비등하게 되었을 때 원호보간 종료라고 판단하고 있습니다. <그림 2.31>은 현재위치(0, 0), 중심(-200, 500)으로 종점(-702, 299)이 결정되어 CCW 원호보간의 예입니다. 현재위치(0, 0)와 중심(-200, 500)으로부터 정해지는 반경에 의해 CCW 방향으로 보간을 실시하여 지정된 종점(-702, 299)은 4 상한으로 제 2 축(ax2)이 단축이 되고 종점값(-702, 299)에서 제 2 축의 299에 이르렀을 때에 보간 종료라고 판단합니다.



제 4상한으로 ax2가 299와 같아지면 종료  
<그림 2.31 원호보간 종료 판정의 예>

**(2) CW 원호보간 Drive 의 예**

X, Y 축에 대하여 현재위치 (시점: 0, 0)로부터 중심 (X: 5000, Y: 0), 종점(X: 5000, Y:-5000)에서 CW 원호보간을 합니다. 보간 드라이브 속도는 1000pps 의 정속 Drive 를 실시하며 선속일정 Mode 로 보간합니다.



- WR5←0104h write      ax1: X 축, ax2: Y 축 지정, 선속일정
- WR6←0900h write      Range: 4,000,000 (배율:2)
- WR7←003Dh write
- WR0←0100h write
- WR6←4DC0h write      2 축 선속일정을 위한 Range:  
4,000,000×1.414=5,656,000
- WR7←0056h write
- WR0←0200h write
- WR6←01F4h write      기동속도: 500×2=1000pps
- WR0←0104h write
- WR6←01F4h write      drive 속도: 500×2=1000pps
- WR0←0105h write
- WR6←1388h write      중심 X: 5000

WR7←0000h write  
 WR0←0108h write  
 WR6←0000h write      중심 Y: 0  
 WR7←0000h write  
 WR0←0208h write  
 WR6←1388h write      종점 X: 5000  
 WR7←0000h write  
 WR0←0106h write  
 WR6←EC78h write      종점 Y: -5000  
 WR7←FFFFh write  
 WR0←0206h write  
 WR0←0032h write      CW 원호보간 Drive

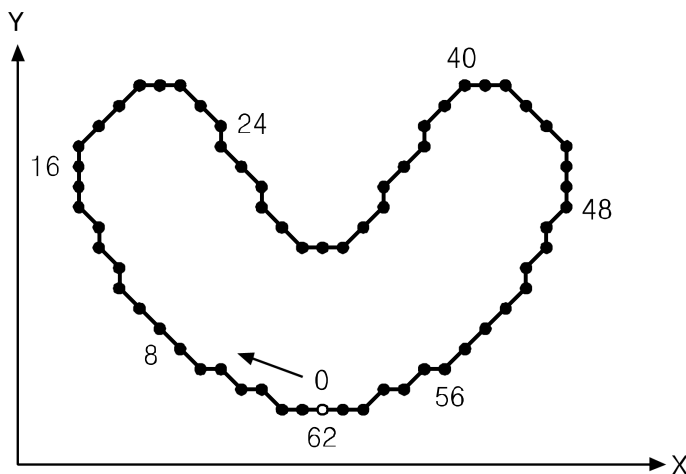
### 5.4.3 2 축/3 축 Bit pattern 보간

상위 CPU 에서 작성한 Bit pattern 화 된 보 간 Data 를 Packet(결정된 양의 Data 결정체)으로 받아, 지정된 드라이브 속도로 보 간 Pulse 를 연속적으로 출력하는 보 간 Drive 입니다. Bit pattern 보 간에서는 2 축 또는 3 축의 +방향, -방향의 드라이브 Pulse 를 1bit, 1pulse 로 각각의 Register 에 설정합니다. Drive pulse 를 출력할 때는 “1”, 출력하지 않을 때는 “0”으로 설정합니다.



Ex.

<그림 2.32> 와 같은 궤적을 그리는 경우 X+방향, X-방향, Y+방향, Y-방향 각각의 Drive pulse 를 발생할 때는 “1”, 발생하지 않을 때는 “0”으로 하면 Bit pattern data 는 아래와 같이 됩니다.

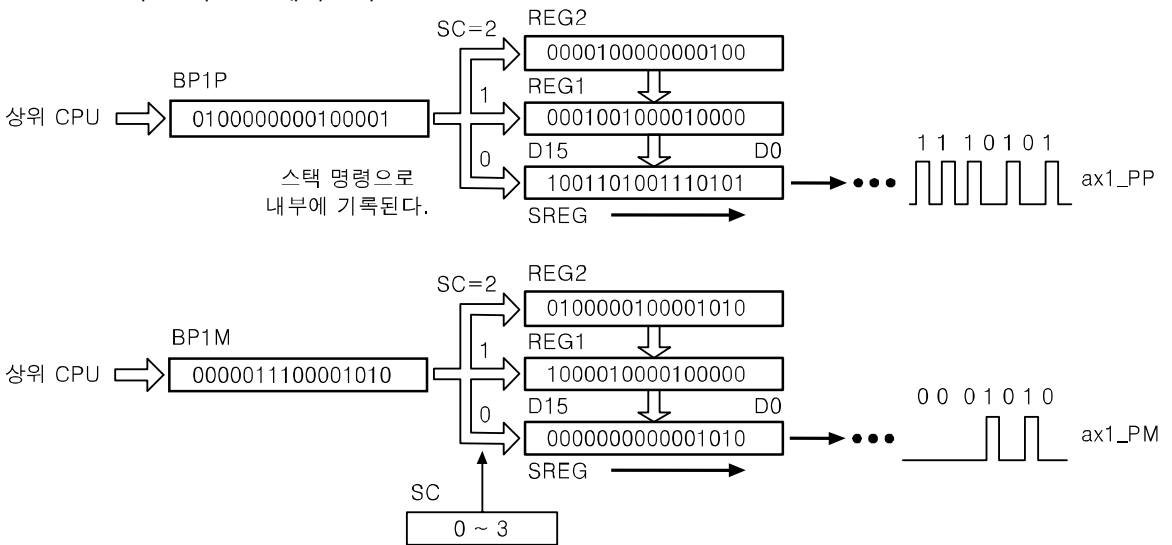


<그림 2.32 bit pattern 도형의 예>

	← 56	← 48	← 40	← 32	← 24	← 16	← 8	← 0	
01000000	00000000	00011111	11011011	11110110	11111110	00000000	00000000		XPP(X+방향)
01111111	11110101	00000000	00000000	00000000	00000000	00101011	11111111		XPP(X-방향)
00000000	00000000	00000000	11111111	00000000	00001111	11111111	11010100		YPP(Y+방향)
00001010	11111111	11111100	00000000	00111111	11000000	00000000	00000000		YPP(Y-방향)

<그림 2.33>은 Bit patter 보간의 제 1 축의 Register 구성과 bit data 의 움직임을 나타내고 있습니다. BP1P register, BP1M register 는 상위 CPU 로부터 Bit pattern data 를 기입하는 16Bit register 입니다. (8bit bus 의 경우는 L byte, H byte 로 나누어 기입합니다. ) +방향의 16bit 의 bit data 는 BP1P register 에, -방향의 data 는 BP1M register 에 기입합니다. Bit pattern 보간이 시작되면 D0 부터 차례대로 Drive pulse 를 출력합니다.

- SC: 스택 카운터 (RR0/D14, 13)
- BP1P: ax1 축+방향 데이터 레지스터
- BP1M: ax1 축-방향 데이터 레지스터
- SREG: 16 비트 시프트 레지스터
- REG1: 16 비트 시프트 레지스터
- REG2: 16 비트 시프트 레지스터



<그림 2.33> Bit pattern 보간의 Register 구성과 bit data 의 움직임 (ax1 축)>

Stack Counter (SC)는 Bit pattern data 를 Count 하는 Counter 로 0 에서 3 까지 변화합니다.

RR0 register 의 D14, D13 bit 가 stack counter 의 값을 나타내고 있습니다. BP1P, BP1M register 에 기입된 Data 는 BPdata stack 명령에 의해 내부 16bit shift register (SREG), 또는 2 개의 16bit register (REG1, REG2)의 어느쪽인가에 기입해집니다. 이 때, Stack counter SC=0 일 때는 SREG 에, SC=1 일 때는 REG1 에, SC=2 일 때는 REG2 에 기입해 집니다. Data 가 모두 기입되면 stack counter(SC)는 1 개 증가합니다.

2 축 또는 3 축 bit pattern 보간 명령에 의해, Bit pattern 보간이 시작되면 모든 축은 주축의 기본 펄스에 동기되어 16bit shift register (SREG)의 D0 bit 의 값에 의해 Drive pulse 를 출력합니다.

D0 의 값이 “1” 일 때는 Drive pulse 가 출력되고 “0” 일 때는 출력되지 않습니다. Shift register 의 16bit 가 모두 출력을 마치면 REG1 register 의 Data 가 shift register 에, REG2 register 의 Data 가 REG1 register 에 옮겨져 Stack Counter (SC)가 하나 감소합니다.

상위 CPU 는 Stack counter(SC)가 3 이 되면 더 이상 Bit pattern data 를 내부에 Stack 할 수 없습니다. 하지만 보간 Drive 가 시작되면 Drive pulse 의 출력에 따라 Stack counter (SC)의 값이 3→2→1 으로 감소하기 때문에 다시 Data 를 기입할 수가 있습니다.

Stack counter(SC)=0 는 보간 Drive 종료를 의미하기 때문에 연속하여 Bit pattern 을 보간하는 경우는 SC=2 또는 1 사이에 다음의 Data 를 설정하지 않으며 안되기 때문에 SC 의 값이 2 에서 1 로 바뀌었을 때 상위 CPU 에 대해서 Interrupt 를 발생하여 Data 기입을 요구합니다.

**(1) 보간 Drive 속도의 제한**

Bit pattern 보간의 Drive 속도는 최고 4MHz 까지 가능합니다. 그러나 bit 수가 48bit 를 넘는 경우에는 보간 Drive 중에 Data 를 보충해 나가지 않으면 안되기 때문에, 보간 Drive 속도는 CPU 의 Pattern data 의 Setup 에 필요로 하는 시간에 의존하게 됩니다.



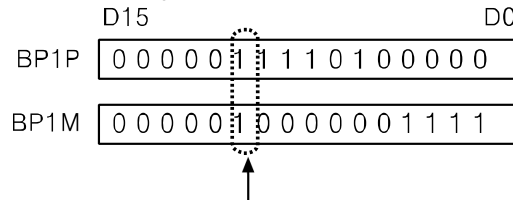
Ex.

2 축 Bit pattern 보간으로 CPU 축이 4×16 bit 의 연산과 Data set 및 BP data stack 명령의 발행으로 100 μs 걸린다고 하면, 보간 Drive 속도는 1/(100 μs/16)=160kpps 이하의 속도입니다.

**(2) Bit pattern 보간의 종료**

Bit pattern 보간에는 다음과 같이 2 가지 방법으로 종료합니다.

- ① 제 1 축 Data 에 종료 코드를 기입합니다.  
주축(ax1)의 +방향, -방향의 Bit data 를 동시에 “1”로 하면, Bit pattern 보간 종료라고 판단합니다. 종료 코드를 검출하면 Stack Counter(SC)는 강제적으로 0 이 되어, 이후에 Stack 된 Bit pattern data 는 모두 무효가 됩니다.



주축의 +방향, -방향 모두 1로 종료

- ② Data 기입을 중지합니다.  
BPdata stack 명령에 의한 내부 Register 의 Bit pattern data 의 기입을 중지하면 모든 Bit pattern data 를 Drive pulse 로서 출력하여 SC=0 이 되어 보간 Drive 를 종료합니다.

**(3) 정지 명령에 의한 보간 Drive 중단**

Bit pattern 보간 drive 를 실시하고 있는 주축(ax1)에 즉시정지명령 혹은 감속정지명령을 설정하면 보간중에 Drive 는 정지합니다. 여기서 다시 Bit pattern 보간명령을 설정하면 Bit pattern 보간을 계속할 수 있습니다. 정지 명령에 의해 Drive 를 정지하여 보간을 종료하는 경우는 반드시 BP data clear 명령에 의해 설정되어 있는 Data 를 모두 Clear 해야 합니다.

**(4) 하드웨어 Limit, 소프트웨어 Limit 에 의한 정지**

보간 Drive 중에 어느 축의 하드웨어 Limit 또는 소프트웨어 Limit 가 Active 되어도 보간 Drive 는 정지합니다. 그대로 보간을 종료하는 경우에는 반드시 BP data clear 명령에 의해 기입되어 있는 Data 를 모두 Clear 해야 합니다.

Bit pattern 보간에서는 +방향/-방향 중 어느 방향의 하드웨어 Limit 및 소프트웨어 Limit 가 Active 되어 보간이 정지하는 경우가 있습니다. Bit pattern 보간에서는 Limit over 영역으로부터 탈출할 수 없으므로 주의해 주십시오.

**(5) Bit pattern data 기입 register**

16 bit bus 및 8 bit bus 에 의한 ax1 축부터 ax3 축의 Bit pattern data 기입 register 의 address 를 각각 아래 표에 나타냅니다.

16 bit data bus 의 Bit pattern data 기입 Register address

Address			Register	내용	응답 Register
A2	A1	A0			
0	0	0			WR0
0	0	1			nWR1
0	0	0	BP1P	ax1 축 +방향 Data register	nWR2
0	0	1	BP1M	ax1 축 -방향 Data register	nWR3
1	1	0	BP2P	ax2 축 +방향 Data register	nWR4
1	1	1	BP2M	ax2 축 -방향 Data register	nWR5
1	1	0	BP3P <sup>※1</sup>	ax3 축 +방향 Data register	nWR6
1	1	1	BP3M <sup>※1</sup>	ax3 축 -방향 Data register	nWR7

※1. BP3P, BP3M 은 각각 WR6, 7 레지스터와 공용입니다.

8bit data bus 의 Bit pattern data 기입 Register address

Address				Register	Address				Register
A3	A2	A1	A0		A3	A2	A1	A0	
0	0	0	0		1	0	0	0	BP2PL
0	0	0	1		1	0	0	1	BP2PH
0	0	1	0		1	0	1	0	BP2ML
0	0	1	1		1	0	1	1	BP2MH
0	1	0	0	BP1PL	1	1	0	0	BP3PL
0	1	0	1	BP1PH	1	1	0	1	BP3PH
0	1	1	0	BP1ML	1	1	1	0	BP3ML
0	1	1	1	BP1PH	1	1	1	1	BP3MH



BPmPL, BPmPH, BPmML, BPmMH 는 각각 아래와 같은 byte 를 나타냅니다. (m 는 1~3)

BPmPL: BPmP 의 하위 byte (D7~D0)

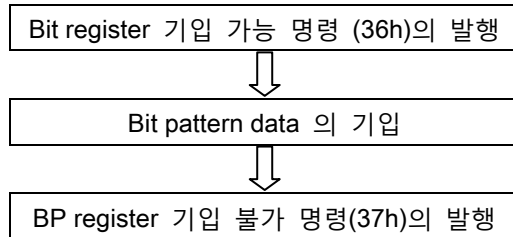
BPmPH: BPmP 의 상위 byte (D15~D8)

BPmML: BPmM 의 하위 byte (D7~D0)

BPmMH: BPmM 의 상위 byte (D15~D8)

Bit pattern data 기입 register 는 nWR2~WR7 register 와 같은 Address 입니다.

Reset 후에는 Bit pattern data register 에 data 를 기입할 수 없습니다. Data 기입은 다음의 순서로 실시합니다.



**Note**

Bit pattern data 의 기입 종료 후, BP register 기입 불가 명령 (37h)을 발행하지 않으면 해당 bit 의 블랭크가 교체된 상태가 되기 때문에, nWR2~WR5 register 의 기입을 할 수 없습니다.

**(6) Bit pattern 보간 Drive 의 예**

주축(ax1)=X 축, 제 2 축(ax2)=Y 축 으로 설정하고 <그림 2.32>의 Bit pattern 도형을 1000pps 의 정속 Drive, 선속 일정 Mode 로 보간합니다.

WR5←0104h write      ax1: X 축, ax2: Y 축 지정, 선속일정  
 WR6←0900h write      주축 속도 Parameter 설정  
 WR7←003Dh write      Range: 4,000,000 (배율:2)  
 WR0←0100h write

WR6←4DC0h write      2 축 선속일정을 위한 Range:  
 4,000,000×1.414=5,656,000

WR7←0056h write  
 WR0←0200h write  
 WR6←01F4h write      기동속도: 500×2=1000pps  
 WR0←0104h write

WR6←01F4h write      drive 속도: 500×2=1000pps  
 WR0←0105h write  
 WR0←0039h write      BPdata clear  
 WR0←0036h write      BP register 기입 가능

BP1P←0000h write      포인트 0~15 X 축 +방향  
 BP1M←2BFFh write      X 축 -방향  
 BP2P←FFD4h write      Y 축 +방향

	BP2M←0000h write	Y 축 -방향	
	WR0←0038h write	BP data stack	
	BP1P←F6FEh write	포인트 16~31 X 축 +방향	
	BP1M←0000h write	X 축 -방향	
	BP2P←000Fh write	Y 축 +방향	
	BP2M←3FC0h write	Y 축 -방향	
	WR0←0038h write	BP data stack	
	BP1P←1FDBh write	포인트 32~47 X 축 +방향	
	BP1M←0000h write	X 축 -방향	
	BP2P←00FFh write	Y 축 +방향	
	BP2M←FC00h write	Y 축 -방향	
	WR0←0038h write	BP data stack	
J1	WR0←0034h write RR0/D14, 13 read	2 축 Bit pattern 보간: Drive 시작 Stack counter 가 2 이하로 될 때 까지 만약, D14=D13=1 이라면 J1 으로 점프	} ※1
	BP1P←4000h write	포인트 48~61 X 축 +방향	
	BP1M←7FF5h write	X 축 -방향	
	BP2P←0000h write	Y 축 +방향	
	BP2M←0AFFh write	Y 축 -방향	
	WR0←0038h write	BP data stack	
J2	WR0←0037h write RR0/D8 read	BP register 기입 불가 보간 Drive 종료까지 기다립니다. 만약, D8=1 이면 J2 로 점프	

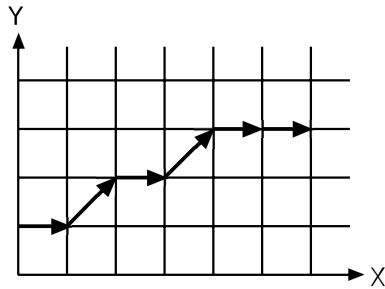
### (7) Interrupt 를 이용한 Bit pattern 보간 Drive

Bit pattern 보간 Drive 에서는 Drive 중에 Stack Counter(SC)의 값이 2 에서 1 로 바뀌었을 때 Interrupt 가 발생하여 Data 기입을 요구할 수가 있습니다. Interrupt 를 발생시키려면 WR5 register 의 D15 bit 를 1 으로 합니다.

이후 Bit pattern 보간 Drive 를 시작하면 Stack Counter(SC)의 값이 2 에서 1 로 바뀌었을 때 INTN 출력 신호가 Low level 로 떨어집니다. Bit pattern 보간 data 는 16bit 또는 32bit 의 Pattern data 이며 BP data stack 명령을 기입하면 Interrupt 는 해제됩니다. 보간 Drive 에서 발생시킨 Interrupt 는 보간 Interrupt clear 명령 (3Dh)으로 해제할 수 있습니다. 또한 INTN 출력 신호를 Low 상태로 두더라도, 보간 Drive 가 종료하면 해제되어 Hi-Z 로 돌아옵니다. 연속으로 BP data 가 계속되는 경우는 반복합니다.

### 5.4.4 선속일정

선속일정 제어는 보간을 실시하고 있는 축의 합성 속도를 항상 일정하게 하는 기능입니다. <그림 2.34>는 2축 보간의 궤적을 나타내고 있습니다. 주축의 기본 Pulse 에 따라서 각 축이 Drive pulse 를 출력하는 것으로 그림과 같이 X, Y 축 양쪽 모두 Drive pulse 가 출력 될 때에는 1축 만의 드라이브 Pulse 출력에 비해서 1.414 배 긴 거리를 이동하기 때문에 양축 모두 Drive pulse 가 출력될 때의 속도를 1축만의 Drive pulse 속도보다 1.414 배로 해야 합니다.



<그림 2.34 2축 보간의 예>

#### (1) 2축 선속일정

2축 선속일정을 설정하려면 WR5 register 의 D9, D8bit 를 각각 0, 1 로 합니다. 그리고 보간 제 2 축의 Range parameter 를 주축의 Range parameter 의 1.414 배의 값으로 설정해 두면 1축만의 Drive pulse 출력 시에는 주축의 Range parameter 가 사용되고 양축이 Drive pulse 를 출력하면 자동적으로 제 2 축의 Range parameter 가 사용되어 Pulse 주기가 1.414 배로 변경됩니다.

#### (2) 3축 선속일정

3축 선속일정도 마찬가지로 WR5 register 의 D9, D8 bit 를 1, 1 로 합니다. 그리고 제 2 축의 Range Parameter 는 주축의 Range 의 1.414 배로 제 3 축의 Range parameter 는 주축의 Range 의 1.732 배로 설정합니다.

보간 Drive 가 시작하면 3축 중 어느 축이든 1축만의 Drive pulse 출력 시에는 주축의 Range parameter 가 사용되고 2축의 Drive pulse 출력 시에는 제 2 축의 Range parameter 가 사용됩니다. 그리고 제 3 축의 Drive pulse 출력시에는 제 3 축의 Range parameter 가 사용됩니다. (<그림 2.36> 참조)

3축 보간에도 주축과 제 2 축만의 2축 선속일정이 가능합니다. 이 경우는 WR5 register 의 D9, D8 bit 를 0, 1 로 합니다.



#### Ex.

- 선속일정 보간 Drive 의 예

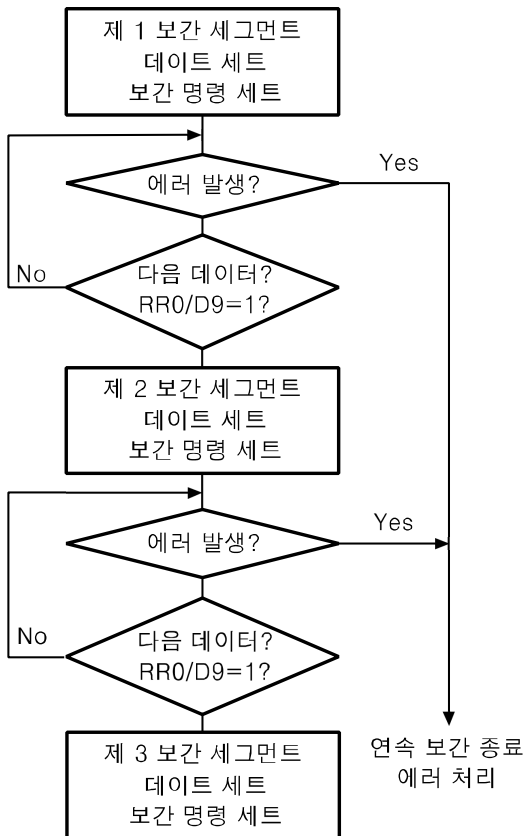
아래와 같이 주축(ax1)=X 축, 제 2 축(ax2)=Y 축 으로 설정하고, 1000pps 의 정속 Drive, 선속일정 Mode 로 직선보간을 실시하면 <그림 2.35>와 같이 Drive pulse 가 출력됩니다.

WR5←0104h write	ax1: X 축, ax2: Y 축 지정, 선속일정
WR6←0900h write	주축 속도 Parameter 설정
WR7←003Dh write	Range: 4,000,000 (배율:2)



### 5.4.5 연속보간

연속보간은 직선보간→원호보간→직선보간→... 과 같이 각각의 보간노드(node)를 Drive 도중에 정지하지 않고 연속해서 보간을 실시하는 동작입니다. 연속보간 Drive 는 현재 실행하고 있는 보간 Drive 와 다음 보간 Drive 사이에 Parameter data 및 보간 명령을 설정하여 연속적인 보간 Drive 를 실시합니다. 따라서 모든 보간노드 (node) 는 그 드라이브 시작부터 종료까지의 시간으로 다음의 보간노드(node)의 Data 명령을 설정할 시간 이상이 필요합니다. 아래의 그림은 연속 보간의 조작 순서를 나타내고 있습니다.



연속보간에서는 RR0 register 의 D9(CNEXT) Bit 를 사용합니다. 이 Bit 는 보간 Drive 중 다음의 보간노드(node)의 Data 및 보간 Drive 명령의 설정여부를 나타냅니다. 1 은 설정가능, 0 은 설정 불가능을 나타냅니다.

Drive 정지시에는 0 이 되고 보간 Drive 가 시작되면 즉시 1 이 되어 다음의 보간노드(node)의 Data 및 보간 Drive 명령이 설정됩니다. 다음의 보간노드(node)의 보간드라이브 명령이 설정되면 0(설정불가)으로 변경되고 다음의 보간세그먼트가 Drive 를 시작하면 다시 1 이 되어 다음의 보간 세그먼트의 Data 및 보간 Drive 명령이 설정 가능해 집니다.

**(3) Interrupt 를 이용한 연속보간**

WR5 register 의 D14 bit 는 연속보간 시, Interrupt 허가/금지를 설정하는 Bit 입니다. 이 Bit 를 1 로 하면, RR0 register 의 D9(CNEXT) bit 가 1(Interrupt 허가)이 되어 INTN 출력 신호가 Low level 로 떨어집니다. Interrupt 처리 루틴에서는 RR0 register 의 D9(CNEXT) bit 를 확인합니다. 1(기입가능)이면 다음 보간 노드(node)의 Data 및 보간 Drive 명령을 기입합니다. 연속보간 Interrupt 의 경우는 다음 보간 Drive 명령을 기입하면 INTN 신호는 Hi-Z 로 돌아옵니다. 다음 보간노드(node)의 Data 기입 전에 Clear 명령(3Dh)을 이용하여 Interrupt 를 해제하는 것도 가능합니다. 보간 Interrupt 는 보간 Drive 가 종료하면 강제적으로 해제되어 INTN 신호는 Hi-Z 로 돌아갑니다.

**(4) 연속보간 중의 에러 발생**

연속보간 Drive 도중에 Limit over run 등의 에러가 발생하면 현재 Drive 노드에서 정지합니다. 정지한 보간노드에서는 Drive 중에 다음의 노드(node)의 data 및 보간 명령을 설정하지만 이 보간명령은 무효가 됩니다.

각 보간 노드(node)의 Data 및 보간 명령 설정 전에 에러 체크를 하지 않으면 에러로 정지한 후, 즉시 2 개의 보간노드(node)가 실행되기 때문에 각 보간노드(node)의 Data 및 보간 명령 설정 전에는 반드시 에러 체크를 실시하여 에러이면 연속보간 루프를 빠져나가도록 해야 합니다.

**(5) 연속보간의 주의사항**

- 각 보간노드(node)는 필요한 data 를 설정한 후에, 보간명령을 설정합니다. 설정을 반대로 하지마십시오.
- 연속보간의 Drive 속도는 최고 2MHz 까지 입니다.
- 모든 보간노드(node)를 Drive 할 시간은 보간축의 에러 체크 다음, 보간노드 (node)의 Data 및 명령을 설정할 시간 이상이 필요합니다. 만약 다음 보간노드 (node)의 Data 를 설정하고 있는 동안에 현재의 보간노드(node)의 Drive 가 종료 했을 경우에는 RR0 register 의 D9(CNEXT) bit 는 0 이 됩니다만 다음 보간노드 (node)의 Drive 명령이 기입해지면 일단 정지 후 계속해서 연속보간을 하게 됩니다.
- 연속보간 속에 원호보간이 있는 경우, 원호보간은 종점의 단축이 종점값보다  $\pm 1\text{LSB}$  어긋나는 경우가 있기 때문에 각 노드(node)의 오차가 누적하지 않게 미리 각각의 원호보간 종점을 확인하고 나서 연속보간을 실행합니다.
- 2 축 보간을 시작으로 3 축 보간을 실행 또는 3 축 보간을 시작으로 2 축 보간을 실행하는 연속 보간은 할 수 없습니다.
- 연속보간 도중에 보간 축지정 변경은 할 수 없습니다.



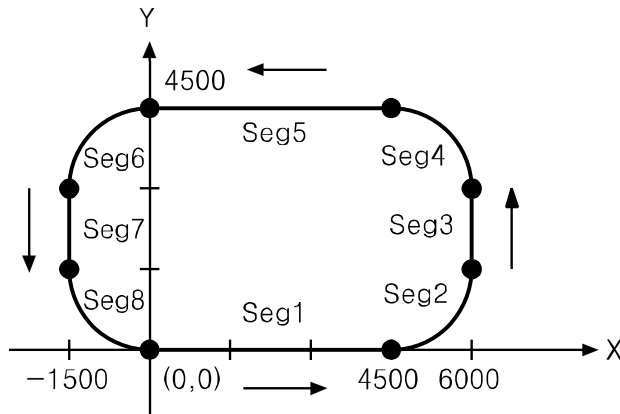
**Ex.**

- 연속보간의 예

<그림 2.37>은 (0, 0)를 시점으로 노드(node) 1 부터 2, 3 ... 노드(node) 8 까지를 연속보간 하는 예입니다.

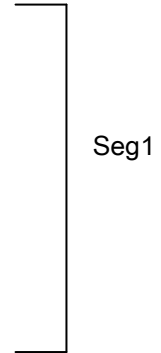
노드(node) 1, 3, 5, 7 은 직선보간이며, 노드(node) 2, 4, 6, 8 은 반경 1500 의 1/4 원 입니다.

보간 속도는 1000pps 의 정속 드라이브로 선속을 일정합니다.



<그림 2.37 연속보간 궤적의 예>

WR5←0104h write	ax1: X 축, ax2: Y 축 지정, 선속일정
WR6←0900h write	주축 속도 Parameter 설정
WR7←003Dh write	Range: 4,000,000 (배율:2)
WR0←0100h write	
WR6←4DC0h write	2 축 선속일정을 위한 Range: 4,000,000×1.414=5,656,000
WR7←0056h write	
WR0←0200h write	
WR6←01F4h write	기동속도: 500×2=1000pps
WR0←0104h write	
WR6←01F4h write	drive 속도: 500×2=1000pps
WR0←0105h write	
WR6←1194h write	종점 X 값: 4500
WR7←0000h write	
WR0←0106h write	
WR6←0000h write	종점 Y 값: 0
WR7←0000h write	
WR0←0206h write	
WR0←0030h write	2 축 직선보간



J1	RR0/ D5, 4	read; X, Y 축으로 에러가 있으면 D5 or D4=1 라면 Error 로 점프하여 에러 처리	A처리
	RR0/ D9	read; 다음의 노드(node) Data D9=0 이라면 J1 에 점프; 기입가능 대기	
	WR6←0000h write	중심 X 값: 0	Seg2
	WR7←0000h write		
	WR0←0108h write		
	WR6←05DCh write	중심 Y 값: 1500	
	WR7←0000h write		
	WR0←0208h write		
	WR6←05DCh write	중점 X 값: 1500	
	WR7←0000h write		
	WR0←0106h write		
	WR6←05DCh write	중점 Y 값: 1500	
	WR7←0000h write		
	WR0←0206h write		
	WR0←0033h write	CCW 원호보간 A 처리	
	WR6←0000h write	중점 X 값: 0	Seg3
	WR7←0000h write		
	WR0←0106h write		
	WR6←05DCh write	중점 Y 값: 1500	
	WR7←0000h write		
	WR0←0206h write		
	WR0←0030h write	2 축 직선 보간 A 처리	

이하 Seg 4~8 에 대해서도 이와 같이 실시합니다.



### 5.4.6 가감속 Drive에서의 보간

보간은 일반적으로 정속 Drive 로 실시합니다만 직선 가감속 Drive 또는 S 자 가감속 Drive (직선보간만)에서도 가능합니다. 보간 Drive 에서는 연속보간에 대해서 가감속 Drive 를 가능하게 하기 위해서 감속 유효명령 (3Bh), 감속 무효명령(3Ch)을 사용합니다. 감속 유효명령은 보간 Drive 에 대해서 자동감속 또는 매뉴얼 감속을 유효하게 하는 명령입니다. 감속 무효명령은 감속 유효명령을 무효로 하는 명령입니다. 가감속으로 단독 보간 Drive 를 한다면 Drive 시작 전에 반드시 감속 유효 상태로 해야 합니다. Drive 의 도중에 감속 유효 명령의 설정은 적용되지 않습니다.

#### (1) 2 축/3 축 직선 보간의 가감속 Drive

2 축/3 축 직선보간에서는 직선 가감속 Drive 및 S 자 가감속 Drive 가 가능합니다. 또한 감속에 대해서는 자동감속과 매뉴얼 감속 모두가 가능합니다. 매뉴얼 감속의 경우는 종점좌표의 각 축 중에서 절대치가 가장 큰 값을 주축의 매뉴얼 감속점으로 설정합니다.



Ex.

주축: X, 제 2 축: Y, 제 3 축: Z 축으로 두어, 종점 (X: -20000, Y: 30000, Z: -50000)까지의 3 축 직선 보간을 실시하는 경우, 감속에 필요로하는 Pulse 수를 만약 5000 으로 하면 Z 축의 종점의 절대치가 가장 크기 때문에,  $50000-5000=45000$  을 주축 X 축의 매뉴얼 감속점으로 설정합니다.

직선보간의 가감속 Drive 의 예는 '2.4.1 2 축/3 축 직선보간'을 참고하십시오.

#### (2) 원호보간, Bit pattern 보간의 가감속 Drive

원호보간, Bit pattern 보간에서는 매뉴얼 감속으로 직선 가감속 Drive 만이 가능합니다. S 자 가감속 Drive 와 자동 감속은 사용할 수 없습니다. 그림과 같이 반경 10,000 의 진원의 궤적을 직선 가감속 Drive 로 그린다면, 원호보간은 자동 감속할 수 없기 때문에 매뉴얼 감속점을 미리 설정해야 합니다.

반경 10,000 의 원은 0 에서 7 상한 모두를 통과합니다. 각 상한에 대해서 단축은 항상 Pulse 를 출력하기 때문에 단축은 1 상한 당  $10000/2=7071$  Pulse 출력하게 됩니다.

따라서 주축으로부터 출력되는 기본 Pulse 의 Pulse 수는 원 전체로  $7071 \times 8=56568$  이 됩니다. 또 기동속도를 500pps 로 하고, Drive 속도를 20,000pps 까지 0.3 초에 직선 가속시킨다면 가속도는  $(20,000-500)/0.3=65,000\text{pps/sec}$  가 되어, 가속시에는 소비되는 Pulse 수는 그림에서 사선부의 면적이 되기 때문에  $(500+20,000) \times 0.3/2=3075$  가 됩니다.

따라서, 매뉴얼 감속점은  $56568-3075=53493$  으로 설정합니다.



Note

선속일정 Mode 에서 이 계산식은 성립되지 않습니다.

WR3←0001h write	감속 시작점: 매뉴얼
WR5←0004h write	보간 ax1: X 축, ax2: Y 축 지정
WR6←8480h write	Range: 2,000,000 (배율:4)
WR7←001Eh write	

WR0←0100h write

WR6←0082h write

가속도:  $130 \times 125 \times 4 = 65,000 \text{ pps/sec}$

WR0←0102h write

WR6←007Dh write

기동속도:  $125 \times 4 = 500 \text{ pps}$

WR0←0104h write

WR6←1388h write

drive 속도:  $5000 \times 4 = 20,000 \text{ pps}$

WR0←0105h write

WR6←D8F0h write

중심 X 값: -10,000

WR7←FFFFh write

WR0←0108h write

WR6←0000h write

중심 Y 값: 0

WR7←0000h write

WR0←0208h write

WR6←0000h write

중점 X 값: 0

WR7←0000h write

WR0←0106h write

WR6←0000h write

중점 Y 값: 0

WR7←0000h write

WR0←0206h write

WR6←D0F5h write

매뉴얼 감속점: 53493

WR7←0000h write

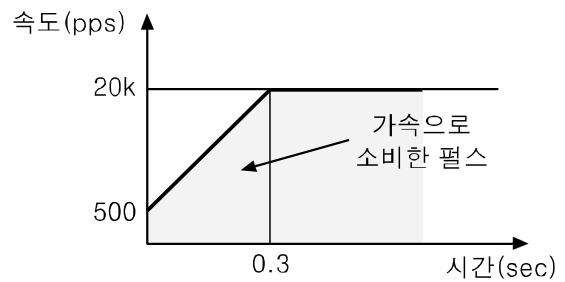
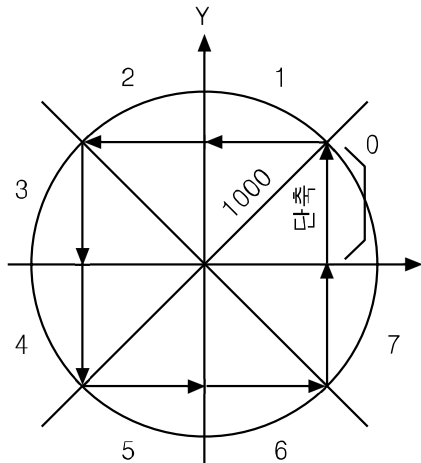
WR0←0107h write

WR0←003Bh write

감속 유효

WR0←0033h write

CCW 원호보간 Drive



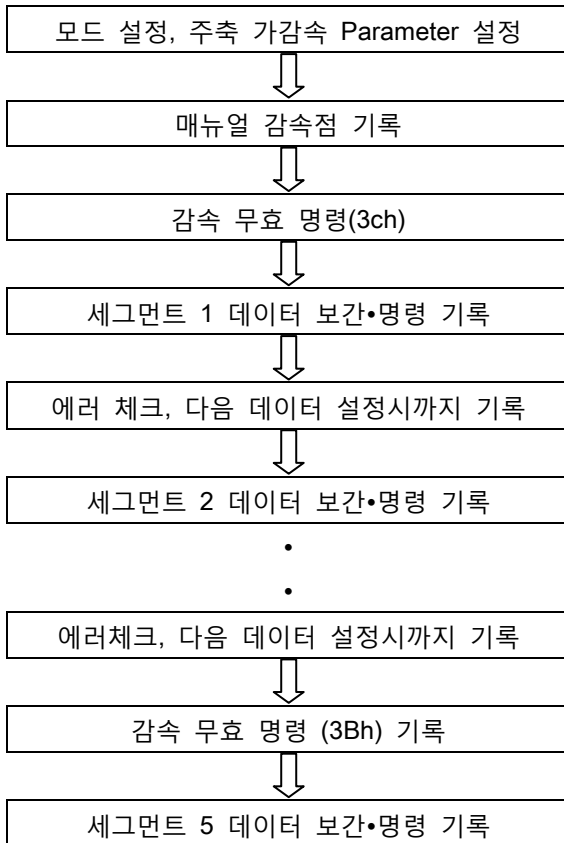
**(3) 연속보간의 가감속 Drive**

연속보간에서도 매뉴얼 감속의 직선 가감속 Drive 만이 가능합니다. S자 가감속 Drive 와 자동 감속은 사용할 수 없습니다. 연속 보간에서는 미리 매뉴얼 감속점을 설정해 두지 않으면 안됩니다. 매뉴얼 감속점은 감속을 실시하는 최종 노드(node)로 출력되는 주축으로부터 기본 Pulse 에 대한 값을 설정합니다. 연속 보간에서는 처음부터 감속을 무효로 하여 보간 Drive 를 시작합니다. 그리고 감속시키는 최종노드(node)의 보간 명령 설정 전에 감속유효명령을 설정합니다. 최종노드(node)의 Drive 에 들어가면 감속유효상태가 되고 최종노드(node)의 시작시점에서 주축의 기본 Pulse 수가 Count 되어 매뉴얼 감속점의 값을 넘었을 때에 감속이 시작됩니다.



**Ex.**

노드(node) 1 에서 5 까지의 연속보간에 대해서, 최종노드 (node) 5 로 매뉴얼 감속시키는 경우에는 다음과 같이 설정됩니다.



매뉴얼 감속점은 노드(node) 5 부터 주축의 기본 Pulse 수에 대한 값이므로 주의해 주십시오.

감속 Pulse 가 2000 소비된다면 노드(node) 5 로 출력되는 기본 Pulse 의 총 Pulse 수가 5000, 5000-2000=3000 을 매뉴얼 감속점으로 설정합니다.

감속을 시작하고 나서 정지할 때 까지는 반드시 하나의 노드(node) 안에서 설정해야 합니다. 즉, 감속 정지의 최종 노드(node)는 주축으로부터 출력되는 기본 Pulse 의 총 수가 감속에 소비하는 Pulse 수 이상이 필요합니다.

### 5.4.7 보간 Step 전송 (Command)

보간 Drive 를 1Pulse 마다의 Step 전송하는 동작입니다. Command 로 실시하는 방법으로 Step 전송 때에는 보간 주축은 정속 Drive 로 설정합니다. 각 축에서 출력되는 Drive pulse 의 High level 폭은 보간의 주축으로 설정되어 있는 Drive 속도에 의해 정해진 Pulse 주기의 1/2 값이 됩니다. Low level 폭은 다음의 Command 가 발생할 때 까지 연장됩니다.

#### (1) Command 에 의한 보간 Step 전송

보간 Drive 를 Step 전송하는 Command 로는 보간 Single step (3Ah) 명령이 있습니다. WR5 register 의 D12bit 를 1 로 하면 Command 에 의한 보간 Step 전송이 가능하게 됩니다. Command 에 의한 보간 Step 전송 방법은 아래와 같습니다.

- ① WR5 register 의 D12 bit 를 1 로 합니다.  
Command 에 의한 보간 step mode 가 됩니다.
- ② 보간 주축의 기동 속도와 Drive 속도를 같은 값으로 설정합니다.  
기동 속도와 Drive 속도를 같은 값으로 하면 정속 Drive 가 됩니다. 이 때의 속도치는 Single step 명령을 설정하는 사이클보다 빠르게 설정해야 합니다.



**Ex.**

Single step 명령을 최고 1msec 의 사이클로 설정한다면 기동속도와 Drive 속도는 1000pps 보다 빠른 값으로 설정합니다.

- ③ 보간 Data 를 설정합니다. (중점, 중심점 등)
- ④ 보간 명령을 기입합니다.  
보간 명령을 기입해도 Command 에 의한 보간 Step mode 로 되어 있기 때문에, 각 축의 Drive pulse 는 출력하지 않습니다.
- ⑤ 보간 Single step (3Ah) 명령을 기입합니다.  
보간 연산 결과로 Drive pulse 가 각 축에서 출력됩니다. 보간 Drive 가 종료할 때 까지 Single step (3Ah) 명령을 기입합니다. 보간 Step 전송을 도중에 중지하는 경우는 주축에 대해서 즉시정지명령(27h)을 설정하여 Drive 속도에서의 1 펄스 주기 이상의 시간지연을 주고 다시 보간 Single step 명령을 설정하면 Drive 가 정지합니다. 보간 Drive 종료 후에 기입해진 보간 Single step 명령은 적용되지 않습니다.

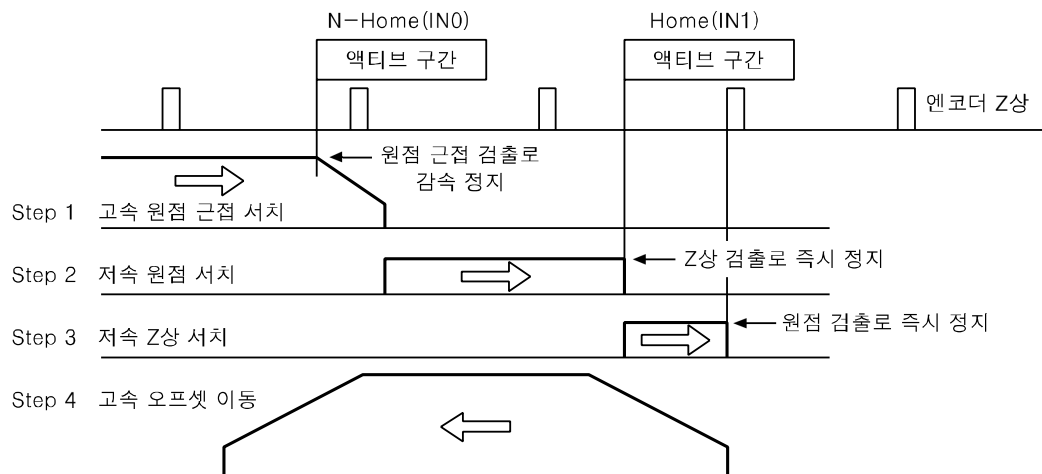
### 5.5 자동 원점 복귀 출력

프로그램 설정없이 고속 원점 근접 서치→저속 원점 서치→ Encoder Z 상 서치→Offset 이동 등의 원점복귀 순서를 자동으로 실행하는 기능을 가지고 있습니다. 자동 원점복귀는 아래 표와 같이 Step 1 부터 Step 4 를 설정한 순서에 따라 실행합니다. 각 Step 에 대해서 실행/비실행을 선택, 서치 방향을 설정합니다.

Step 1, 4 는 Drive 속도로 고속서치동작을 합니다. 또한, Step 2, 3 은 원점 검출 속도로 저속 서치동작을 합니다.

Step 번호	동작	서치 속도	검출 신호
Step 1	고속 원점 근접 서치	드라이브 속도 (V)	nIN0※1
Step 2	저속 원점 서치	원점 검출 속도(HV)	nIN1
Step 3	저속 Z 상 서치	원점 검출 속도(HV)	nECZ
Step 4	고속 오프셋 이동	드라이브 속도(V)	-

※1. 원점신호를 nIN0, nIN1 양쪽 모두에 입력하는 것으로서 원점 신호 1 점만으로도 고속 원점 서치가 가능합니다. '2.5.6 자동 원점 복귀의 예'를 참조하십시오.



<그림 2.39 자동 원점 복귀의 개략도>

### 5.5.1 각 Step 의 동작

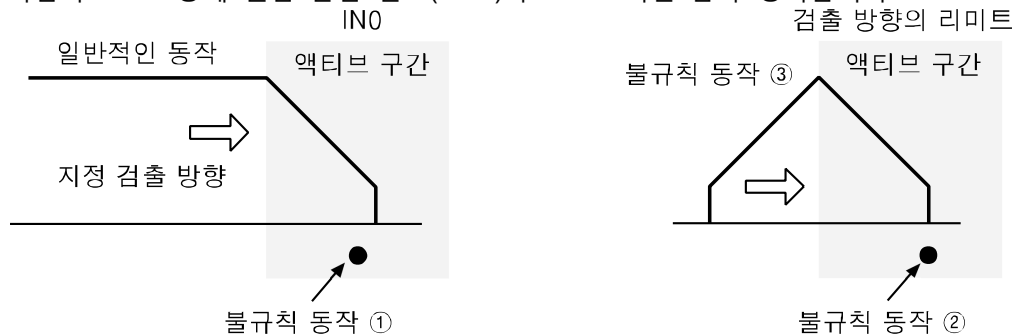
각 Step 은 실행/비실행, 검출방향(+방향/-방향)을 설정할 수 있습니다.

비 실행으로 설정하면 해당 Step 은 실행되지 않고 다음 Step 으로 진행됩니다.

#### (1) Step 1 고속 원점 근접 서치

고속 원점 근접 서치는 설정된 Drive 속도(V) 및 방향으로 Drive pulse 출력 중에 원점 근접 신호(nIN0)가 Active 되면 Step 2 로 진행합니다. 고속 서치 동작을 실행하기 위해서는 Drive 속도(V)를 기동속도(SV)보다 빠르게 설정합니다.

가감속 Drive 중에 원점 근접 신호(nIN0)가 Active 되면 감속 정지합니다.

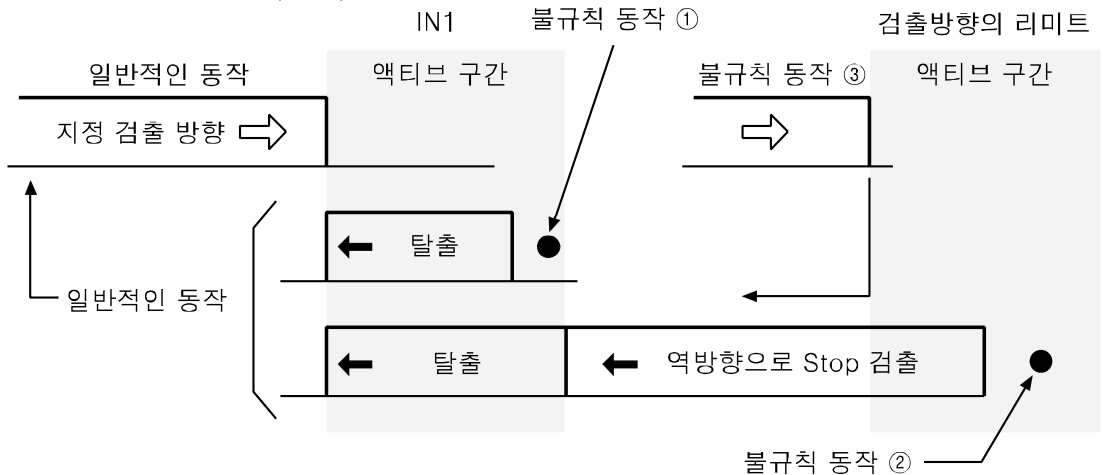


[불규칙 동작]

- ① Step 1 시작 전에 이미 원점 근접 신호(nIN0)가 Active 되어 있다. → Step 2 로 진행됩니다.
- ② Step 1 시작 전에 검출 방향의 Limit 신호가 Active 되어 있다. → Step 2 로 진행됩니다.
- ③ 실행 중에 검출 방향의 Limit 신호가 Active 되었다. → Drive 를 정지하고 Step 2 로 진행됩니다.

#### (2) Step 2 저속 원점 서치

저속 원점 서치는 Drive pulse 출력 중에 원점신호(nIN1)가 Active 되면 원점 검출 속도(HV), 지정된 방향으로 저속 서치 동작을 실행합니다. 저속 서치 동작을 실행하기 위해서는 원점 검출 속도(HV)를 기동속도(SV)보다 낮은 값으로 설정합니다. 정속 드라이브 시 원점신호(nIN1)가 Active 되면 즉시 정지합니다.



## [불규칙 동작]

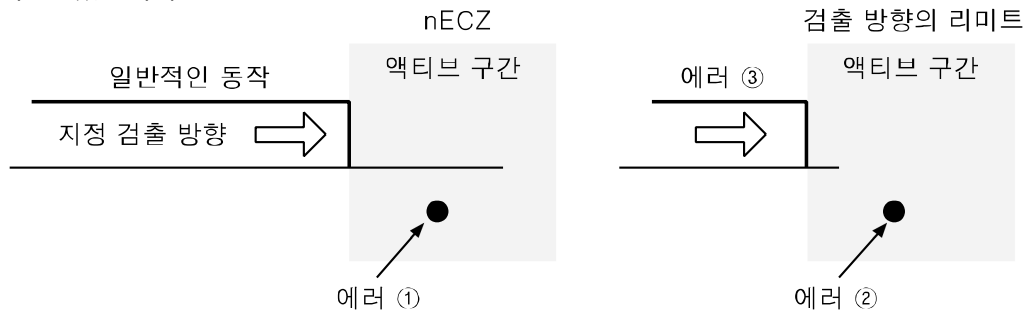
- ① Step 2 시작 전에 이미 원점신호(nIN1)가 Active 되어 있다.  
→ 원점 신호 (nIN1)가 Inactive 될 때까지, 지정의 검출 방향과 반대의 방향으로 원점검출속도(HV)로 이동하여 원점신호 (nIN1)가 Inactive 되면 Step 2 를 시작합니다.
- ② Step 2 시작 전에 검출 방향의 Limit 신호가 Active 되어 있다.  
→ 원점 신호(nIN1)가 Active 될 때 까지 지정된 검출 방향과 반대의 방향으로 원점 검출 속도(HV)로 이동합니다. 원점 신호(nIN1)가 Active 되면 원점 신호(nIN1)가 Inactive 될 때 까지 지정된 검출 방향과 반대 방향으로 원점 검출 속도(HV)로 이동합니다. 원점 신호(nIN1)가 Inactive 되면 Step 2 를 실행합니다.
- ③ 실행 중에 검출 방향의 Limit 신호가 Active 되었다.  
→ Drive 를 정지하고 → ②와 동일한 동작을 합니다.

**(3) Step 3 저속 Z 상 서치**

Encoder Z 상 신호(nECZ)가 Active 되면 원점 검출 속도(HV), 지정된 방향으로 Drive pulse 를 출력합니다. 저속 서치 동작을 실시하기 때문에 원점 검출 속도(HV)를 기동속도(SV)보다 낮은 값으로 설정합니다.

정속 Drive 실행 중, Encoder Z 상 신호(nECZ)가 Active 되면 즉시 정지합니다.

검출 조건으로서 Encoder Z 상 신호(nECZ)와 원점 신호(nIN1)의 AND 조건으로 정지시킬 수도 있습니다.



Encoder Z 상 신호(nECZ)가 Active 될 때 실제위치 Counter(EP)를 Clear 시킬 수도 있습니다. '2.3.4 외부 신호에 의한 실제위치 Counter의 Clear'을 참조하십시오.

**Note**

- ① Step 3 시작 전에 이미 Encoder Z 상 신호 (nECZ)가 Active 되어 있으면 에러가 되어 nRR2 register 의 D7bit 가 1 이 되어 자동 원점 복귀는 종료합니다. Step 3 은 반드시 Encoder Z 상 신호(nECZ)가 Inactive 상태(안정상태)에서 시작하도록 기계 시스템을 조정해야 합니다.
- ② Step 3 시작 전에 검출 방향의 Limit 신호가 Active 되어 있으면 에러가 되어 nRR2 register 의 검출 방향 Limit 에러 bit (D2 또는 D3)가 1 이 되어 자동 원점 복귀는 종료합니다.
- ③ 실행 중에 검출 방향 Limit 신호가 Active 되면 검출 동작은 중단되어 nRR2 register 의 검출 방향의 Limit 에러 bit (D2 또는 D3)가 1 이 되어 자동 원점 복귀는 종료합니다.

**(4) Step 4 고속 Offset 이동**

Step 4 는 Drive 속도(V), 지정된 방향으로 출력 Pulse 수(P)로 설정되어 있는 Pulse 수 만큼 Pulse 를 출력합니다. 기계적 원점에서 작업 원점으로 이동시키고 싶은 경우에 사용합니다. Mode 설정에 의해서 이동 종료 후, 논리위치 Counter 및 실제위치 Counter 를 Clear 시킬 수도 있습니다. Step 4 시작전 또는 실행중에 이동 방향의 Limit 신호가 Active 되면 에러가 되어 nRR2 register 의 검출방향 Limit 에러 bit (D2 또는 D3) 가 1 이 되어 자동 원점 복귀는 종료됩니다.

**5.5.2 서치 속도와 Mode의 설정**

자동 원점 복귀를 실행하기 위해서는 속도 Parameter 와 Mode 설정이 필요합니다.

**(1) 속도 Parameter 의 설정**

속도 Parameter	명령 코드	내용
드라이브 속도 (V)	05	Step 1, 4 의 고속 서치 속도가 됩니다. 가감속 드라이브를 시키기 위해서 레인지(R), 가속도(A), 기동속도(SV)를 적절한 값으로 설정할 필요가 있습니다. '2.2.2 직선 가감속 Drive'를 참조하십시오.
원점검출 속도(HV)	61	Step 2, 3 의 저속 서치 속도가 됩니다. 검출 신호가 Active 될 때, 즉시 정지시키기 위해서 기동속도(SV)보다 낮은 값으로 설정합니다. '2.2.2 직선 가감속 Drive'를 참조하십시오.

**(2) 자동 원점 복귀의 Mode 설정**

자동 원점 복귀의 Mode 설정은 확장 Mode 설정 명령(60h)에 의해 설정합니다. 아래와 같이 WR7 register 의 각 bit 를 설정합니다. 또한 자동원점복귀 종료 시에 Interrupt 를 발생시키는 경우는 WR6 register D5 ((HMINT)를 1 로 설정합니다. 확장 Mode 설정 명령 (60h)은 WR6 및 WR7 의 각 Bit data 가 동시에 내부 Register 에 설정되기 때문에 WR6 register 의 다른 bit 에 대해서도 적절히 설정할 필요가 있습니다.

WR7	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	DCCW2	DCCW1	DCCW0	DCC-L	DCC-E	LIMIT	STAND	PCLR	ST4-D	ST4-E	ST3-D	ST3-E	ST2-D	ST1-E	ST1-D	ST1-E
	편차 카운터 클리어 출력							Step 4		Step 3		Step 2		Step 1		
WR6	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	FL2	FL1	FL0	FE4	FE3	FE2	FE1	FE0	SMODE	0	HMINT	VRING	AVTRI	POINV	EPINV	EPCLR

WR7/D6, 4, 2, 0    STm-E    각 Step 의 동작의 실행/비실행을 설정합니다.  
 0: 비실행, 1: 실행  
 각 Step 에서 검출하는 입력 신호의 논리 설정은 WR1 register 로 설정합니다.  
 '4.4 WR1 mode register 1'을 참조하십시오.



WR7/D7, 5, 3, 1	STm-D	각 Step 의 검출/이동 방향을 설정합니다. 0: +방향, 1: -방향
WR7/D8	PCLR	1 로 설정하면 Step 4 종료 후, 논리위치 Counter 및 실제위치 Counter 가 Clear 됩니다.
WR7/D9	SAND	1 로 설정하면 Step 3 동작은 원점신호 (nIN1)가 Active 될 때, Encoder Z 상 신호(nIN2)가 Active 될 때 정지합니다.
WR7/D10	LIMIT	Over run limit 신호(nLMT+ 또는 nLMT-)를 사용하여 자동원점복귀를 실시할 때 1 로 설정합니다.
WR7/D11	DCC-E	0 으로 설정합니다. (본 보드에서는 편차카운터 클리어 기능은 지원하지 않습니다.)
WR7/D12	DCC-L	0 으로 설정합니다.
WR7/D15~13	DCCW2~0	0 으로 설정합니다.
WR5/D5	HMINT	자동원점복귀 종료 후 Interrupt 신호(INTN)를 발생시킵니다. 본 Bit 를 1 하면, 자동원점복귀 종료 후 Interrupt 신호 (INTN)가 Low active 되어 Interrupt 를 발생시킨 축의 RR3/D8 (HMEND) bit 를 1 로 설정합니다. Interrupt 를 발생시킨 축의 RR3 register 를 읽으면, RR3 register 의 Bit 는 0 으로 Clear 되어 Interrupt 출력 신호는 Hi-Z 으로 돌아옵니다.

Reset 시에는 각 축의 Mode 설정 Bit 는 모두 0 으로 설정됩니다.

### 5.5.3 자동원점복귀의 실행과 상황(Status)

#### (1) 자동원점복귀의 실행

자동원점복귀는 자동원점복귀 실행명령(62h)에 의해 설정됩니다.

각 축의 자동원점복귀 Mode 와 속도 Parameter 를 설정한 후, WR0 register 에 축 지정과 함께 명령코드 62h 를 기입하는 것으로 시작합니다.

각 축, 개별적으로 실행 가능하며 전체 축 동시 실행도 가능합니다.

#### (2) 자동원점복귀의 중단

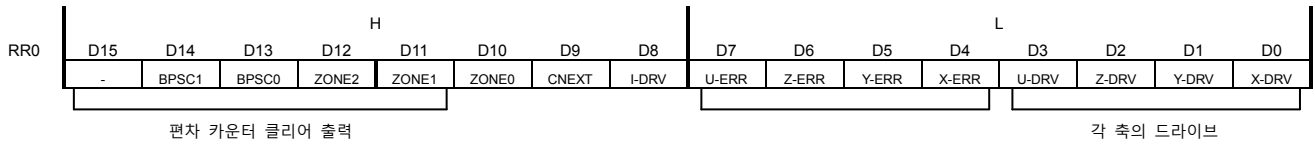
자동원점복귀를 도중에 중단시키고 싶을 때는 실행하고 있는 축에 대해 Drive 감속정지명령(26h), 또는 드라이브 즉시정지명령(27h)을 설정합니다.

현재 실행하고 있는 Step 은 중단되어 자동 원점 복귀를 종료합니다.

#### (3) 주 Status register

주 Status register RR0 의 D3~D0 은 각 축의 Drive 실행 상태를 나타내는 bit 입니다. 자동원점복귀 실행 시도 본 bit 로 상태 확인이 가능합니다.

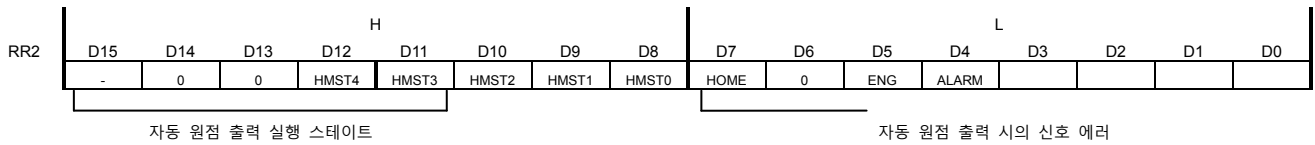
각 축의 자동원점복귀가 시작되면 이 Bit 가 1 이 되어 Step 1 동작 시작부터 Step 4 동작 종료까지 1 을 나타내고 있습니다. Step 4 를 종료하면 0 으로 돌아옵니다.



각 축의 에러를 나타내는 D7~D4(n-ERR) Bit 는 Step 1, 2 의 불규칙 동작으로 검출 방향의 Limit 신호 등에 있어서 정상 동작에도 불구하고 1 을 나타낼 때가 있기 때문에 주의하십시오. 이러한 에러 비트는 자동원점복귀 실행 시에는 확인하지 않고 자동원점복귀가 종료한 후에 확인하도록 하십시오.

**(4) Status register 2**

Status register 2(RR2)는 D7~D0 에는 에러 정보가 표시되고 D12~D8 에는 원점복귀 실행 Status 가 표시됩니다.



에러 정보 Bit 중 D7(HOME) bit 는 자동원점복귀 실행 중 Step 3 시작 시에 이미 Encoder Z 상 신호(nIN2)가 Active 되어 있으면 1 이 됩니다. 이 Bit 는 다음의 Drive 명령 또는 자동원점복귀 명령을 설정하면 Clear 됩니다.

그리고 종료 Status clear 명령 (25h)으로도 Clear 할 수가 있습니다. 자동원점복귀 실행 Status 는 자동원점복귀 실행 중에 현재 실행하고 있는 동작 상태를 나타냅니다.

실행 스테이트	실행 Step	동작 내용
0		자동원점출력 실행 명령 기다립니다.
3	Step 1	지정검출 방향으로 IN0 신호의 Active 기다립니다.
8		지정검출 반대 방향으로 IN1 신호의 Active 기다립니다. (불규칙적인 동작)
12	Step 2	지정검출 반대 방향으로 IN1 신호의 Inactive 기다립니다. (불규칙적인 동작)
15		지정검출 방향으로 IN1 신호의 Active 기다립니다.
20	Step 3	지정검출 방향으로 ECZ 신호의 Active 기다립니다.
25	Step 4	지정검출 방향으로 Offset 이동중입니다.

### 5.5.4 자동 원점 출력시의 에러

자동원점복귀 실행 중, 아래의 표와 같은 에러발생이 일어날 가능성이 있습니다.

에러 발생 요인	에러 발생 후의 동작	종료 표시
Step 1~4 에서 Alarm 신호 Active	검출 드라이브 즉시 정지, 이후 Step 은 실행하지 않고 종료	RR0-D7~4:1, nRR2-D4:1, nRR1-D14:1
Step 1~4 에서 EMG 신호 Active	검출 드라이브 즉시 정지, 이후 Step 은 실행하지 않고 종료	RR0-D7~4:1, nRR2-D5:1, nRR1-D15:1
Step 3 에서 진행방향의 Limit 신호 (LMT+/-) Active	검출 드라이브는 즉시/감속 정지, 이후 Step 은 실행하지 않고 종료	RR0-D7~4:1, nRR2-D3/2:1, nRR1-D13/12:1
Step 3 에서 시작 전에 ECZ 신호가 Active	이후 스텝은 실행하지 않고 종료	RR0-D7~4:1, nRR2-D7:1

자동원점복귀 종료 후, 반드시 각 축의 에러 Bit (RR0-D7~D4)를 확인해야 합니다.

1 로 설정된 에러 Bit 는 올바른 자동 원점 복귀를 실행하지 않습니다.

한편, 자동 원점 복귀 실행 도중에 각 축의 에러 Bit 를 확인하는 것은 올바르지 않습니다.

Step 1, 2 의 불규칙 동작으로 인하여 에러 Bit 가 1 로 설정될 수 있기 때문입니다.

#### (5) 센서 고장 시의 증상

원점 신호나 Limit 신호 등의 센서회로의 고장 시에 일어날 수 있는 증상에 대해서 기술합니다. 배선으로 인한 노이즈나 소자의 불안정 동작 등의 원인에 의한 고장에 대해서는 해석이 어렵고 여기에서는 시스템 개발 시, 신호 Level 의 논리 설정 및 신호 연결 시 일어날 수 있는 경우만 기술합니다.

고장 요인		증상
리미트 센서 및 배선 경로의 고장	항상 ON	해당 방향이 움직이지 않고, 종료시에 Limit 에러 비트 (nRR2-D3/2)가 1 이 되어 있습니다.
	항상 OFF	해당 방향의 기계적 종점에 부딪치고, 원점 출력 동작이 종료되지 않습니다.
원점근점(nIN0) 센서 및 배선 경로의 고장	항상 ON	Step1 을 유효로 설정하고 신호가 OFF 의 위치에서 자동 원점 출력을 시작하여도 Step1 (고속 원점 근점 서치)을 실행하지 않고, Step2 로 넘어갑니다.
	항상 OFF	Step1 (고속 원점 근점 서치)로 Limit 에서 정지하고 나서 Step2 의 불규칙 동작으로 진행합니다. 원점 출력의 결과는 올바르지만 일반적인 동작이 아닙니다.
원점근점(nIN1) 센서 및 배선 경로의 고장	항상 ON	Step2(저속 원점서치)로 역방향으로 움직이다 역방향의 리미트에서 정지합니다. 종료 시에 역방향 Limit 의 에러비트 (nRR2-D3/2)가 1 이 됩니다.
	항상 OFF	Step2 (저속 원점서치)로 지정된 방향의 Limit 에서 역방향으로 이동하고, 역 방향의 리미트에서 종료합니다. 종료 시에 역방향 리미트의 에러비트 (nRR2-D3/2)가 1 이 됩니다.

고장 요인		증상
Z 상 (nECZ) 센서 및 배선 경로의 고장	항상 ON	Step 3 (저속 Z 상 서치)에서 종료합니다. nRR2-D7 이 1 이 됩니다.
	항상 OFF	Step3 (저속 Z 상 서치)로, 지정된 방향의 리미트에서 정지합니다. 종료 시에 지정방향 Limit 의 에러 비트 (nRR2-D3/2)가 1 이 됩니다.

### 5.5.5 자동 원점 출력 시의 주의점

#### (1) 서치 속도

원점 검출 속도 (HV)는 원점 복귀 시, 위치 정밀도를 올리기 위해서 저속으로 설정할 필요가 있습니다. 입력 신호가 Active 되면 즉시정지 하도록 기동속도 이하의 값으로 설정합니다. Step3 의 Encoder Z 상 서치를 실시하는 경우에는 Z 상 신호의 지연과 원점 검출 속도(HV)의 관계가 중요하므로 다음과 같이 Z 상 신호의 Photo-coupler 의 지연시간과 적분필터의 지연시간이 최대 500  $\mu$ s 걸린다면, Encoder 의 Z 상 출력이 1msec 이상 ON 하도록 원점 검출 속도를 설정할 필요가 있습니다.

#### (2) Step3 (Z 상 서치) 시작 위치

Step3 의 Z 상 서치는 Z 상 (nECZ) 신호가 Inactive 상태에서 Active 로 변화할 때에 검출 Drive 를 정지시킵니다. 따라서 Step3 의 시작위치 (Step2 의 정지위치)가 이 변화지점에서 벗어나야 합니다. 일반적으로 Step3 의 시작위치가 Encoder Z 상 위치의 180° 반대 측이 되도록 기계적으로 조정합니다.

#### (3) 소프트웨어 Limit

자동원점복귀 실행 중 소프트웨어 Limit 는 적용되지 않습니다. 소프트웨어 Limit 를 적용하면 자동원점복귀는 올바르게 실행하지 않습니다. 자동원점복귀를 정상 완료 후 논리위치 Counter, 실제위치 Counter 를 올바르게 설정한 후에 소프트웨어 Limit 를 설정해야 합니다.

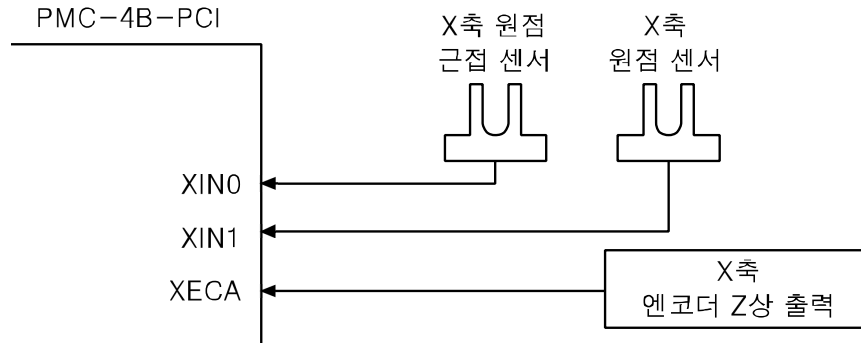
#### (4) 각 입력 신호의 논리설정

자동 원점 복귀로 사용하는 입력 신호의 설정은 WR1 register 의 Bit (WR1-D2, D4, D7)로 설정합니다.

### 5.5.6 자동 원점 복귀의 예

(1) 원점 근접, 원점, Z 상 신호에 의한 원점 복귀의 예

Step	입력 신호와 논리 레벨	검출 방향	검출 속도
Step1	원점 근접 (IN0)신호, Low Active	-	20,000pps
Step2	원점 (IN1)신호, Low Active	-	500pps
Step3	Z 상 (XECZ)신호, High Active	+	500pps
Step4	(+ 방향으로 3500 펄스 Offset 이동)	+	20,000pps



- Step1의 고속 서치 및 Step4의 Offset 이동은 가감속 Drive로 실시합니다.  
기동속도: 1,000pps로 20,000pps까지를 0.2초에  
(가감속도=19,000/0.2=95,000pps/sec)직선 가감속 시킵니다.
- Step4 완료 후 논리위치 Counter, 실제위치 Counter의 값을 Clear합니다.

[Parameter 및 Mode 설정]

- WR0←010Fh write X축 선택
- WR1←0010h write 입력 신호 논리 설정  
XIN0, XIN1: Low active,  
XIN2: High active  
'4.4 WR1 mode register 1'을 참조하십시오.
- WR6←5D00h write 확장 Mode 설정
- WR7←015Fh write WR6에 입력 신호 필터의 Mode 기록  
'2.8 입력 신호 필터'를 참조하십시오.  
D15~D13 010 필터 지연: 512 μs  
D9 0 XIN2 신호: 필터 무효  
D8 1 XIN1, 0 신호: 필터 유효  
WR7에 자동 원점 복귀의 Mode 기록  
D15~D13 000  
D12 0  
D11 0  
D10 0 Limit 신호를 원점 신호로서 사용:  
무효  
D9 0 Z 상 신호 AND 원점 신호: 무효

D8	1	논리/실제위치 Counter clear: 유효
D7	0	Step4 이동 방향: +방향
D6	1	Step4: 유효
D5	0	Step3 검출 방향: +방향
D4	1	Step3: 유효
D3	1	Step2 검출 방향: -방향
D2	1	Step2: 유효
D1	1	Step1 검출 방향: -방향
D0	1	Step1: 유효

WR0←0160h write

X 축으로 확장 Mode 설정 명령 기록

WR6←3500h write

Range: 8,000,000 (배율: 10)

WR7←000Ch write

WR0←0100h write

WR6←004Ch write

가감속도: 95,000pps/sec

WR0←0102h write

95,000/125/10=76

WR6←0064h write

기동속도: 1000pps

WR0←0104h write

WR6←07D0h write

Step 1, 4 의 속도: 20,000pps

WR0←0105h write

WR6←0032h write

Step 2, 3 의 속도: 500pps

WR0←0161h write

WR6←0DACH write

Offset 이동 Pulse 량: 3500

WR7←0000h write

WR0←0106h write

WR0←0162h write

자동 원점 복귀 실행 시작

실행 시작 후, RR0-D0 (X-DRV) bit 가 1 에서 0 으로 돌아오면 자동 원점 복귀는

종료합니다. 종료 후, RR0-D4 (X-ERR)에 Limit 가 1 이 되어 있으면 자동 원점 중에

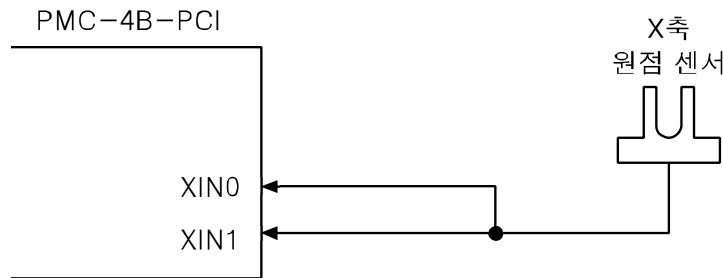
어떠한 에러가 발생하여 자동 원점 복귀가 정상적으로 종료되지 않는 것입니다. XRR2-

D7, D5~D0 bit, XRR1-D15~D12 bit 를 이용하여 해석을 실시합니다.

**(2) 원점 신호만의 원점 복귀 예**

원점 신호를 IN0 과 IN1 단자에 공동 입력하여 하나의 원점신호로 고속 원점 복귀를 실시하는 예입니다.

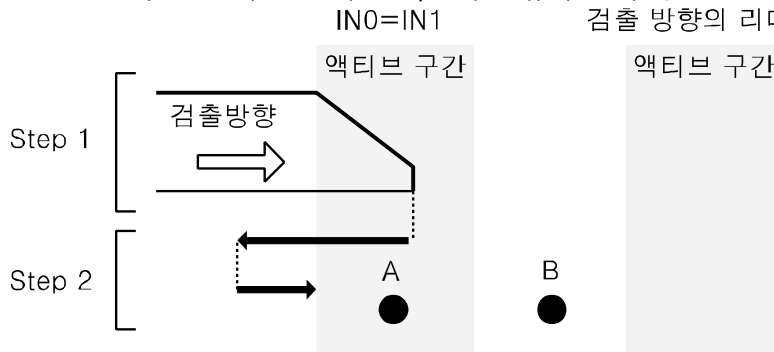
Step	입력 신호와 논리 레벨	검출 방향	검출 속도
Step1	원점 근접 (IN0)신호, Low Active	-	20,000pps
Step2	원점 (IN1)신호, Low Active	-	500pps
Step3	-		
Step4	(+ 방향으로 3500 펄스 Offset 이동)	+	20,000pps



위의 표와 같이, Step1 과 Step2 의 신호논리 Level 과 검출 방향은 같게 합니다. (논리 Level 을 반대로 설정하는 방법도 있습니다.) Step1 에서 고속으로 원점을 서치하여 원점 신호가 Active 되면 감속 정지합니다.

정지 위치가 원점 신호의 Active 구간이면, Step2 의 불규칙 동작 ①에 의해 역방향으로 탈출하고 나서, Step2 의 동작에 들어가 원점을 검출합니다. 만약 Step1 의 정지위치가 원점신호의 Active 구간을 넘겨 버렸을 경우에는 Step2 로 검출 방향의 Limit 에 걸리므로 불규칙 동작 ③의 동작이 됩니다.

자동 원점 복귀 시작 위치가 그림 A 점에 있는 경우에는, Step1 은 실행되지 않고 Step2 의 불규칙 동작 ①을 실행합니다. 그림에서 B 점에 있는 경우에는 Step1 에서 검출 방향의 Limit 에 걸린 후, Step2 의 불규칙 동작 ②를 실행합니다.



**[Parameter 및 Mode 설정]**

- WR0←010Fh write      X 축 선택
  - WR1←0000h write      입력 신호 논리 설정  
XIN0: Low active,  
XIN1: High active
  - WR6←5F00h write      확장 Mode 설정
- '4.4WR1 mode register 1'을 참조하십시오.

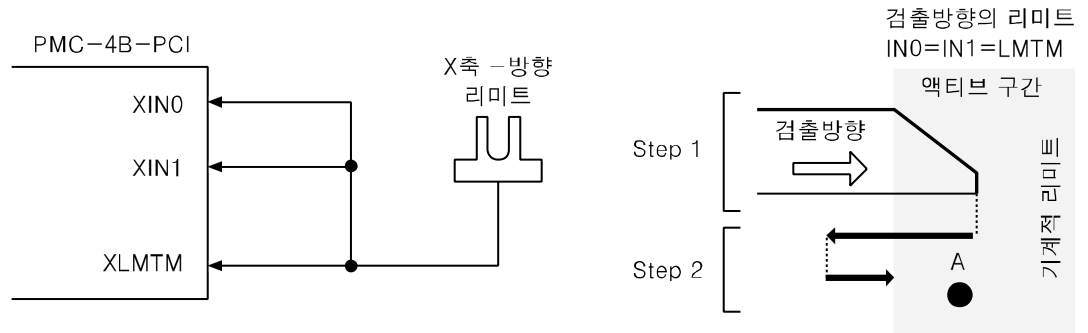




**(3) Limit 신호를 이용한 원점 복귀의 예**

한쪽의 Limit 신호를 원점 신호로 대응하는 방식입니다. 다만, 다음의 2 항의 조건이 있습니다.

- 고속 검출 동작을 실시하는 경우는 Limit 신호가 Active 되는 위치에서 기계적 Limit 까지의 거리 내에서 충분히 감속 정지할 수 있는 것
- 자동 원점 복귀를 시작하는 위치가 Limit 검출 영역 밖에 있지 않을 것  
여기에서는 -방향 Limit 신호를 원점 신호로 대응하는 예를 나타냅니다.
  - XLMT - 입력을 그림과 같이 XIN0 (XIN0+, XIN0-)와 XIN1 입력 단자에 접속합니다.
  - Step1 의 고속서치를 실시하기 때문에 Limit 정지 Mode 를 감속 정지로 설정합니다. (4.5 절 WR2/D2 bit)
  - XLMT-, XIN0, XIN1 신호의 논리 Level 을 모두 같게 설정합니다. (4.5 절 WR2/D4, 4.4 절 WR1/D0, 2 bit)
  - 확장 Mode 설정의 WR7/D10 (Limit 신호 사용) Bit 를 1 로 합니다.



[동작]

그림과 같이 Step1 은 -방향으로 고속으로 Limit 까지 이동합니다. -Limit 신호가 Active 되면 감속정지하고 Step2 로 진행됩니다. Step2 의 불규칙 동작②에 의해서 역방향으로 Limit 를 탈출하고 나서 저속으로 검출방향으로 이동 후, Limit 신호 Active 를 검출하여 정지합니다. 자동원점 복귀시작 위치가 Limit 내에 있을 때에는 (그림의 A 점) Step1 의 동작을 실행되지 않고, Step2 부터 시작됩니다.

[Parameter 및 Mode 설정]

- |                 |   |
|-----------------|---|
| WR0←010Fh write | X 축 선택  |
| WR1←0000h write | 입력 신호 논리 설정   |
|                 | XIN0: Low active,   |
|                 | XIN1: Low active  |
|                 | '4.4WR1 mode register 1'을 참조하십시오.                         |
| WR2←0004h write | D4            0    -Limit 신호 논리: Low active<br>(4.5 절 참조) |
|                 | D2            1    Limit 정지 mode: 감속 정지                   |
|                 | 확장 Mode 설정  |
| WR6←5F00h write | WR6 에 입력 신호 필터의 Mode 를 기록 (2.8 절 참조)                      |

	D15~13	010	필터 지연: 512 $\mu$ s
	D8	1	XLMTM, XIN1, 0 신호: 필터 유효
WR7←054Fh write	WR7 에 자동 원점 복귀의 Mode 를 기록		
	D15~D13	000	
	D12	0	
	D11	0	
	D10	1	Limit 신호를 원점 신호로 사용: 유효
	D9	0	Z 상 신호 AND 원점 신호: 무효
	D8	1	논리/실제위치 Counter clear: 유효
	D7	0	Step4 이동 방향: +방향
WR0←0160h write	D6	1	Step4: 유효
WR6←3500h write	D5	0	Step3 검출 방향:
WR7←000Ch write	D4	0	Step3: 무효
WR0←0100h write	D3	1	Step2 검출 방향: -방향
WR6←004Ch write	D2	1	Step2: 유효
WR0←0102h write	D1	1	Step1 검출 방향: -방향
WR6←0064h write	D0	1	Step1: 유효
WR0←0104h write	X 축으로 확장 Mode 설정 명령 기록		
WR6←07D0h write	Range: 8,000,000 (배율: 10)		
WR0←0105h write	가감속도도: 95,000pps/sec		
WR6←0032h write	95,000/125/10=76		
WR0←0161h write	기동속도: 1000pps		
WR6←0DACH write	Step 1, 4 의 속도: 20,000pps		
WR7←0000h write	Step 2 의 속도: 500pps		
WR0←0106h write	Offset 이동 Pulse 량: 3500		
WR0←0162h write	자동원점복귀 실행시작		

[Limit 신호 사용사의 주의]

- Step1, 2 의 검출방향은 반드시 같은 방향으로 합니다. 또 Step3 (Z 상서치) 동작을 실시하는 경우는 Step1, 2 의 방향과는 반대 방향으로 합니다. Step4 (Offset 이동)는 Step1, 2 의 방향과 반대로 설정합니다.  
반드시 Limit active 구간 밖에서 자동 원점 복귀를 종료시키도록 하십시오.
- Step3 동작의 경우, Z 상 신호와 원점신호(IN1)의 AND 는 적용되지 않습니다. 확장 Mode 설정의 WR7/D9 (SAND) bit 는 반드시 0 으로 하여야 합니다.

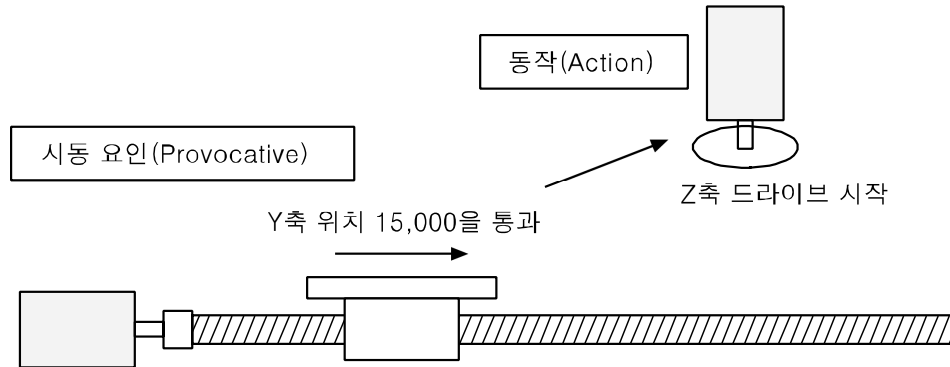
## 5.6 동기 동작

동기 동작은 각 축 사이와 다른 디바이스 사이에 있어 Drive 시작/정지 등의 동작(Action)를 연계시키는 기능입니다.



Ex.

- 예 1: Y 축이 15,000 을 통과하면 Z 축의 Drive 를 시작합니다.



- 예 2: X 축이 -320,000 을 통과하면 Y, Z 축의 Drive 를 정지시킵니다.
- 예 3: 입력 신호가 들어오면 X, Y, Z 축의 위치 Data 를 저장합니다.

일반적으로 이러한 동기 동작은 프로그램을 작성하여 실행할 수도 있지만, 소프트웨어 실행의 지연시간이 허락되지 않을 때 이 기능을 사용하면 편리합니다. 동기 동작은 지정된 기동 요인이 발생하면 즉시 지정된 동작을 실행합니다. 이 동작은 모션 컨트롤 IC의 간섭없이 실행되기 때문에 정밀한 작업의 동기가 필요한 경우 유익합니다.

동기 동작은 동기 동작 Mode register 로 기동 요인과 동작을 다음과 같이 설정합니다.

WR6 register 에 기동 요인 (Provocative)과 다른 축 기동을 지정하여 WR7 register 에 동작(Action)을 지정하고나서 WR0 register 에 축 지정과 함께 동기 동작 Mode 설정 명령 64h 를 설정합니다.

기동 요인은 아래와 같은 WR6 register 로 지정하는 10 종류, 동작은 WR7 로 지정하는 14 종류가 있습니다.

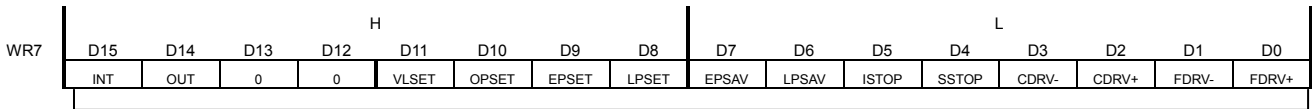
WR6	H								L								
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
	AXIS3	AXIS2	AXIS1	0	0	0	CMD	LPRD	IN3 ↓	IN3 ↑	D-END	D-STA	P≥C-	P<C-	P<C+	P≥C+	
	축 지정			기동 요인 (Provocative)													

기동 요인, 다른 축 기동의 Bit 를 1 로 설정하면 유효, 0 을 설정하면 무효가 됩니다.

- |    |      |  |
|----|------|--|
| D0 | P≥C+ | 논리/실제위치 Counter 가 COMP+ register 보다 크다.<br>(논리/실제위치 Counter 의 선택은 WR2/D5(CMP SL) bit 로 설정합니다.) |
| D1 | P<C+ | 논리/실제위치 Counter 가 COMP+ register 보다 작다.  |
| D2 | P<C- | 논리/실제위치 Counter 가 COMP- register 보다 작다.  |
| D3 | P≥C- | 논리/실제위치 Counter 가 COMP- register 보다 크다.  |

- D4 D-STA Drive 시작
- D5 D-END Drive 종료
- D6 IN3 ↑ nIN3 신호가 Low 에서 High level 로 변경
- D7 IN3 ↓ nIN3 신호가 High 에서 Low level 로 변경
- D8 LPRD 논리위치 Counter 명령 (10h) 설정  
(자/외 축의 동작(Action)에 의한 논리 위치 Counter(LP)저장, 실제위치 Counter(EP)저장 등을 설정하여 동시 읽기 가능)
- D9 CMD 동기 동작 기동 명령 (65h) 설정
- D15~13 AXIS3~1 설정 축의 기동 요인에 의해 동작시키는 것 외 축을 지정  
1: 유효

현재 지정된 축 (자축)	D15 (AXIS 3)	D14 (AXIS 2)	D15 (AXIS 3)
X	U 축 기동	Z 축 기동	Y 축 기동
Y	X 축 기동	U 축 기동	Z 축 기동
Z	Y 축 기동	X 축 기동	U 축 기동
U	Z 축 기동	Y 축 기동	X 축 기동



동작 (Action)

동작(Action) 지정의 각 bit 를 1 로 설정하면 유효, 0 으로 설정하면 무효가 됩니다.

- D0 FDRV+ +방향 정량 drive
- D1 FDRV- -방향 정량 drive
- D2 CDRV+ +방향 연속 drive
- D3 CDRV- -방향 연속 drive
- D4 SSTOP drive 감속 정지
- D5 ISTOP drive 즉시 정지
- D6 LPSAV 현재의 논리 위치 Counter(LP)를 동기 버퍼 Register (BP)에 저장  
LP→BR
- D7 EPSAV 현재의 실제 위치 Counter(EP)를 동기 버퍼 Register(BP)에 저장  
EP→BR
- D8 LPSET WR6, WR7 register 의 값을 논리위치 Counter(LP)에 저장  
LP←WR6, 7  
'2.6.3 동기 동작의 주의점'을 참조하십시오.
- D9 EPSET WR6, WR7 register 의 값을 실제 위치 Counter (EP)에 저장  
EP←WR6, 7  
'2.6.3 동기 동작의 주의점'을 참조하십시오.
- D10 OPSET WR6, WR7 register 의 값을 출력 Pulse 수(P)에 설정  
P←WR6, 7  
'2.6.3 동기 동작의 주의점'을 참조하십시오.

- D11 VLSET WR6 register 의 값을 Drive 속도(V)에 설정  
 $V \leftarrow WR6$   
 '2.6.3 동기 동작의 주의점'을 참조하십시오.
- D14 OUT 사용안함 (0 으로 설정)
- D15 INT Interrupt 신호(INTN) 발생  
 Interrupt 신호(INTN)가 Low active 되면 Interrupt 를 발생시킨 축의 RR3/D9 (SYNC) 비트가 1 을 나타냅니다. Interrupt 를 발생시킨 축의 RR3 register 가 모션 컨트롤 IC 에 의해 판독되면 RR3 register 의 비트는 0 으로 Clear 되어 Interrupt 출력 신호는 Hi-Z 로 돌아옵니다.  
 Reset 시에는 모든 기동 요인, 동작이 무효가 됩니다.

## 5.6.1 동기 동작의 예

### (1) 예 1. Y 축이 15,000 통과 시 → Z 축 + 방향 정량 drive 시작

Parameter, Command 를 아래와 같이 설정합니다.

Y 축 Drive 시작 후 Y 축이 15,000 pulse 를 통과하면, Z 축의 +방향 정량 Drive 가 시작합니다.

WR6←3500h

WR7←000Ch                    Y, Z 축 Range: 800,000 (배율: 10)

WR0←0600h

WR6←0190h

WR7←0000h                    Y, Z 축 가속도:  $400 \times 125 \times 10 = 500 \text{kpps/sec}$

WR0←0602h

WR6←0032h

WR7←0000h                    Y, Z 축 기동속도:  $50 \times 10 = 500 \text{pps}$

WR0←0604h

WR6←0BB8h

WR7←0000h                    Y, Z 축 Drive 속도:  $3000 \times 10 = 30 \text{kpps}$

WR0←0605h

WR6←C350h

WR7←0000h                    Y 축 출력 Pulse 수: 50,000

WR0←0206h

WR6←2710h

WR7←0000h                    Z 축 출력 Pulse 수: 10,000

WR0←0406h

WR6←3A98h

WR7←0

Y 축 COMP+에 15,000 을 Set

WR0←020Bh

WR6←0

WR7←0

Y, Z 축 논리 Counter (LP) clear

WR0←0609h

WR6←2001h

기동 요인: P≥C+, 다른 축 기동: Z

Y 축 동기동작 Mode 설정

WR7←0000h

자축동작: 없음

WR0←0264h

WR6←0000h

WR7←0001h

자축 동작: +방향 정량 Drive

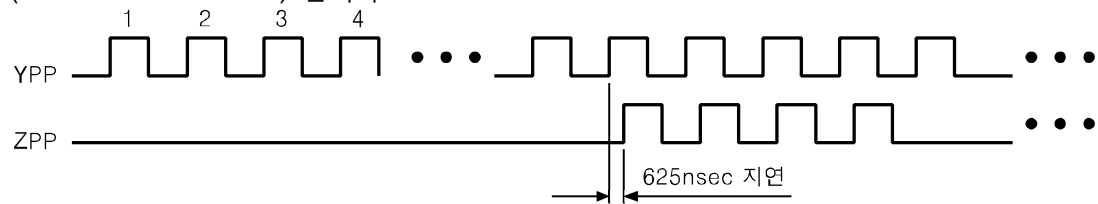
Z 축 동기동작 Mode 설정

WR0←0464h

WR0←0220h

Y 축 +방향 정량 Drive 시작

Y 축 15,000 번째 Pulse 시작에서 Z 축의 제 1pulse 의 시작까지의 지연 시간은 5SCLK (625nsec CLK=16MHz) 입니다.



**(2) 예 2. X 축이 -320,000 통과 → Y, Z 축 Drive 정지**

WR6←3500h

WR7←000Ch

X, Y, Z 축 Range: 800,000 (배율: 10)

WR0←0700h

WR6←0190h

WR7←0000h

X, Y, Z 축 가속도:  $400 \times 125 \times 10 = 500 \text{ kpps/sec}$

WR0←0702h

WR6←0032h

WR7←0000h

X, Y, Z 축 기동속도:  $50 \times 10 = 500 \text{ pps}$

WR0←0704h

WR6←0BB8h

WR7←0000h

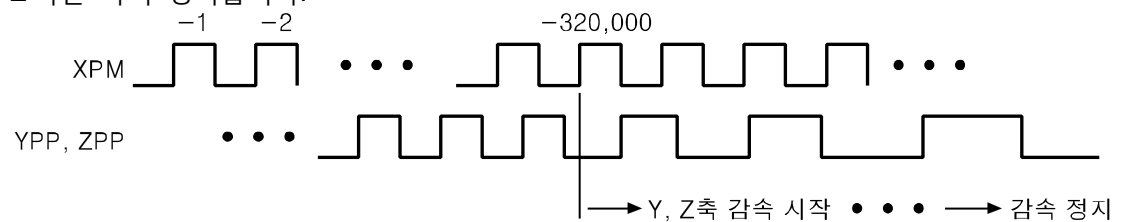
X, Y, Z 축 Drive 속도:  $3000 \times 10 = 30 \text{ kpps}$

WR0←0705h

WR6←A120h		
WR7←0007h	X 축 출력 Pulse 수: 50,000	
WR0←0106h		
WR6←1E00h		
WR7←FFFBh	X 축 COMP-에 -320,000	
WR0←010Ch		
WR6←0		
WR7←0	X 축 논리 counter (LP) clear	
WR0←0109h		
WR6←6004h	기동 요인: P<C-, 다른 축 기동: Y, Z	X 축 동기동작 Mode 설정
WR7←0000h	자속동작: 없음	
WR0←0164h		
WR6←0000h		
WR7←0010h	자속동작: 감속 정지	Y, Z 축 동기동작 Mode 설정
WR0←0664h		
WR0←0622h	Y, Z 축 + 방향 연속 Drive 시작	
WR0←0121h	X 축 -방향 정량 Drive 시작	

예 2 에서는 Y, Z 축을 연속 Drive 시작 후, X 축을 -방향의 정량 Drive 로 시작합니다.  
X 축이 -320,000 pulse 를 통과하면 Y, Z 축은 감속 정지합니다.

Y, Z 축의 동기 동작 지정을 즉시 정지로 지정하면 X 축이 -320,000 pulse 를 통과하면 Y, Z 축은 즉시 정지합니다.



**(3) 예 3. 입력 신호(XIN3) 입력 → X, Y, Z 축의 위치 Data 를 저장**

X, Y, Z 의 3 축 Drive 시작 후, XIN3 신호가 입력되면 3 축의 논리 위치 Counter 값을 각 축의 버퍼 Register (BR)에 저장합니다. 동작 절차는 X 축에 Interrupt 출력 신호(INTN)를 Low active 시켜 각 축의 버퍼를 읽어냅니다.

WR6←3500h

WR7←000Ch            X, Y, Z 축 Range: 800,000 (배율: 10)

WR0←0700h

WR6←0190h

WR7←0000h            X, Y, Z 축 가속도:  $400 \times 125 \times 10 = 500 \text{kpps/sec}$

WR0←0702h

WR6←0032h

WR7←0000h            X, Y, Z 축 기동속도:  $50 \times 10 = 500 \text{pps}$  WR0←0704h

WR0←0BB8h

WR7←0000h            X, Y, Z 축 Drive 속도:  $3000 \times 10 = 30 \text{kpps}$

WR0←0705h

WR6←0

WR7←0            X, Y, Z 축 논리 Counter(LP) clear

WR0←0709h

WR6←6080h            기동 요인: XIN3 ↓, 다른 축 기동: Y, Z    X 축 동기동작 Mode 설정

WR7←8040h            자축동작: LP 동작, Interrupt 발생

WR0←0164h

WR6←0000h

WR7←0040h            자축동작: LP 저장

Y, Z 축 동기동작 Node 설정

WR0←0664h

WR0←0722h            X, Y, Z 축 +방향 연속 Drive 시작

↓

XIN3 입력 신호 입력, Interrupt 발생

↓

동기 동작에 의한 Interrupt 확인  
(XRR3 register 를 읽어 내 D9(SYNC)=1 을 확인)

WR0←0114h

RR6→read            X 축 버퍼 Read

RR7→read

WR0←0214h





D9 (SYNC)=1 을 확인

WR6←0000h

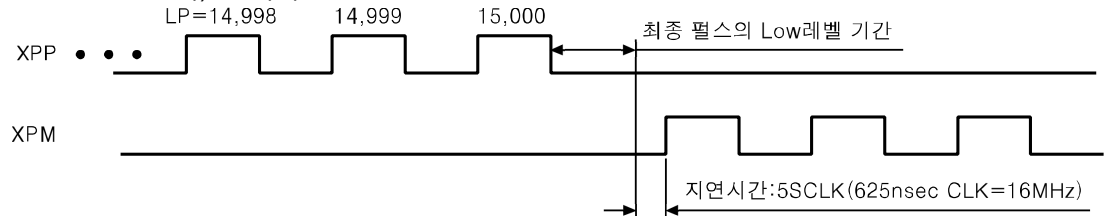
WR7←0000h

동기동작 Mode 의 해제

Interrupt 루틴 내의 처리

WR0←0164h

+15,000 이동 종료부터 -5,000 의 이동 시작까지의 지연시간은 5SCLK (625nsec CLK=16MHz 시) 입니다.



예 4 에서 -5,000 의 Drive 를 시작하는 것과 동시에 Interrupt 를 발생시켜 Interrupt 루틴 내에서 동기동작 Mode 를 해제하고 있습니다. 만약 해제하지 않으면 -방향 정량 Drive 만 실행합니다.

또한, +15,000 으로 이동 중에 +방향 Limit(LMT+)나 긴급정지(EMG)등으로 Drive 가 중단했을 경우에도 다음의 -5,000 의 Drive 는 실행되어 버립니다. 이러한 현상이 시스템상 문제가 되는 경우에는 동기 동작의 사용은 주의하십시오.

### 5.6.2 동기 동작의 지연시간

동기 동작의 지연시간은 아래 표와 같이 기동 요인 발생으로부터 지연과 동작(Action)까지 지연의 합입니다.

#### (1) 기동 요인 발생으로부터의 지연

1SCLK=125nsec (CLK=16MHz 일 때)

시동 요인	지연 시작의 정의		지연시간 (SCLK)		
			최소	표준	최대
P≥C+	P=LP	LP 값이 CMP+/- register 와 비교 조건에 일치할 때 드라이브 펄스의 ↑까지		1	
P<C+	P=EP (A/B 상 입력)	EP 값이 CMP+/- register 와 비교 조건에 일치할 때의 nECA/B 입력 신호의 ↑↓까지	3		4
P<C-					
P≥C-					
D-STA	드라이브 명령 기록 시 WRN 신호의 ↓까지		1		2
D-END	최종 드라이브 펄스의 Low 레벨 종료까지			1	
IN3 ↑	nIN3 신호의 ↑까지 (내장 필터 무효시)		0		1
IN3 ↓	nIN3 신호의 ↓까지 (내장 필터 무효시)		0		1
LPRD	LP 판독 명령 (10h)을 기록하고 WRN 신호의 ↓까지		0		1
CMD	동기 동작 시동 명령 (65h)을 기록하고 WRN 신호의 ↓까지		0		1

**(2) 동작 (Action) 까지의 지연**

1SCLK=125nsec (CLK=16MHz 일 때)

동작	지연 시작의 정의	지연시간 (SCLK)
FDRV+ FDRV- CDRV+ CDRV-	제 1 드라이브 펄스의 ↑까지	4
SSTOP	감속을 시작할 때까지	※1
ISTOP	드라이브를 정지할 때까지	※1
LPSAV EPSAV	LP, EP 의 값이 BR(버퍼)에 세이브될 때 까지	1
LPSET EPSET OPSET VLSET	WR6, 7 의 값이 LP, EP, P, V 에(로) 세트될 때까지	1
INT		1

※1. 현재 출력중인 1Drive pulse 가 종료할 때까지의 시간



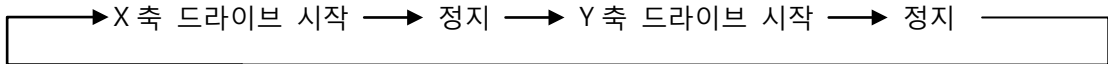
**Ex.**

IN3 입력신호의 ↑부터 논리위치 Counter(LP)를 동기 버퍼 Register(BR)에 저장할 때까지의 지연시간은 IN3 ↑ 지연시간 (0~1 SCLK)과 LP 저장 지연시간(1SCLK)을 합산한, 최소 1SCLK 부터 최대 2SCLK 가 됩니다.

CLK=16MHz 일 때는, 최소 125nsec 부터 최대 250nsec 가 됩니다.

**5.6.3 동기 동작의 주의점**

- 동기 동작은 동작(Action)에 Interrupt 를 동기 지정하는 등 희망하는 동기 동작이 시작된 후에는 동기 동작 Mode 설정 명령 64h 을 다시 설정하여 동기 동작 지정을 해제해야 합니다. 해제하지 않으면 생각지도 않은 곳에서 동작하는 경우가 있습니다.
- 동기 동작의 기능을 이용하면 다음과 같이 끝이 없는 Drive 가 가능하게 됩니다.



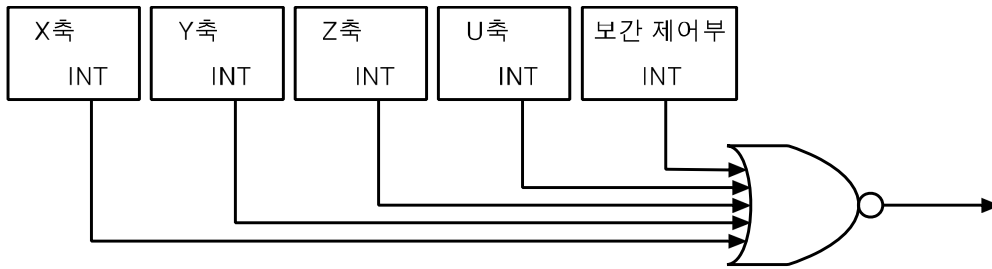
이러한 루프를 정지시키려면, 동기 동작 Mode 설정 명령 64h 를 다시 설정하여 기동 요인 및 동작의 각 Bit 를 무효로 해야 합니다. Drive 하고 있는 축에 대해 즉시정지 명령이나 감속정지 명령은 루프의 해제에는 적용되지 않습니다.

- 동작 (Action) 지정 D8 (LPSET), D9 (EPSET), D10(OPSET), D11(VLSET)은 동기 동작이 기동되기 전에 WR6, WR7 에 data 를 기입해야 합니다. 그런데 동기 동작을 연속적으로 실행할 경우에는 WR6, WR7 의 기록 타이밍이 늦어 동기 동작의 기동과 겹치면 부정 data 가 받아들여지는 일이 있습니다.  
WR6, WR7 의 기록은 동기 동작이 기동하지 않는 기간에 설정합니다.

- 현재 Drive 중 Drive 기동의 동작이 발생했을 경우에는 그 동작은 무시됩니다. 또한 현재 정지 중일 때에도 감속정지, 즉시정지의 동작은 무시됩니다.

## 5.7 Interrupt

Interrupt 의 발생은 X, Y, Z, U 각 축으로 발생시키는 Interrupt 와 보간 Drive 중 Bit pattern 보간 및 연속 보간시에 발생하는 Interrupt 가 있습니다. Interrupt 신호는 INTN 신호 1 개 입니다. 따라서 아래 그림과 같이 각 축의 Interrupt 신호 및 Bit pattern 보간, 연속 보간의 Interrupt 신호는 모두 OR 게이트를 거쳐 Interrupt 를 발생합니다.



<그림 2.42 모션 컨트롤 IC 내의 Interrupt 신호 경로>

각 축의 Interrupt 요인 및 보간 Drive 시 Interrupt 요인은 모두 Interrupt 허가/금지를 설정합니다. Reset 시에는 모두 금지 상태가 됩니다.

### (1) X, Y, Z, U 축의 Interrupt

아래 표는 X, Y, Z, U 축의 Interrupt 발생 요인입니다.

	발생의 확인 nRR3 register	Interrupt 발생 요인
D8 (PULSE)	D0 (PULSE)	1 개의 드라이브 펄스 출력 (정논리 펄스의 경우 펄스의 ↑에서 발생)
D9 (P≥C-)	D1 (P≥C-)	논리/실제위치 Counter 가 COMP- register(CM)보다 크다.
D10 (P<C-)	D2 (P<C-)	논리/실제위치 Counter 가 COMP- register(CM)보다 작다.
D11 (P<C+)	D3 (P<C+)	논리/실제위치 Counter 가 COMP+ register(CM)보다 작다.
D12 (P≥C+)	D4 (P≥C+)	논리/실제위치 Counter 가 COMP+ register(CM)보다 크다.
D13 (C-END)	D5 (C-END)	가감속 드라이브에서 정속영역에서 펄스 출력을 종료
D14 (C-STA)	D6 (C-STA)	가감속 드라이브에서 정속영역에서 펄스 출력을 시작
D15 (D-END)	D7 (D-END)	드라이브 종료

각각의 Interrupt 발생 요인은 nWR1 register 로 Interrupt 발생의 허가(1)/금지(0)를 설정합니다. Drive 를 시작하여 Interrupt 가 발생되면 nRR3 register 의 bit 가 1 이 되어, Interrupt 출력 신호(INTN)가 Low level 로 됩니다. Interrupt 를 발생시킨 축의 RR3 register 를 Read 하면 RR3 register 의 1 로 되어 있던 bit 는 0 으로 Clear 되어 Interrupt 출력 신호(INTN)는 Hi-Z 으로 돌아옵니다.



**Note**

8bit data bus 의 경우는 RR3L register 를 읽으면 모두 Clear 되기 때문에, 자동 원점 복귀 종료 D8 (HMEND), 동기 동작 기동 D9 (SYNC)를 사용하는 경우에는 반드시 RR3H 를 먼저 읽고 난 후, RR3L register 를 설정합니다. 다음의 자동 원점 복귀 종료, 동기 동작 기동의 Interrupt 를 나타냅니다.

허가/금지의 설정	발생의 확인 nRR3 register	Interrupt 발생 요인
확장 모드 설정 명령 (60h) WR6/D5 (HMINT)	D8 (H-END)	자동 원점 출력 종료
동기 동작 지정 명령 (64h) WR7/D15 (INT)	D9 (SYNC)	지정된 기동 요인에 의하여 동기 동작 기동

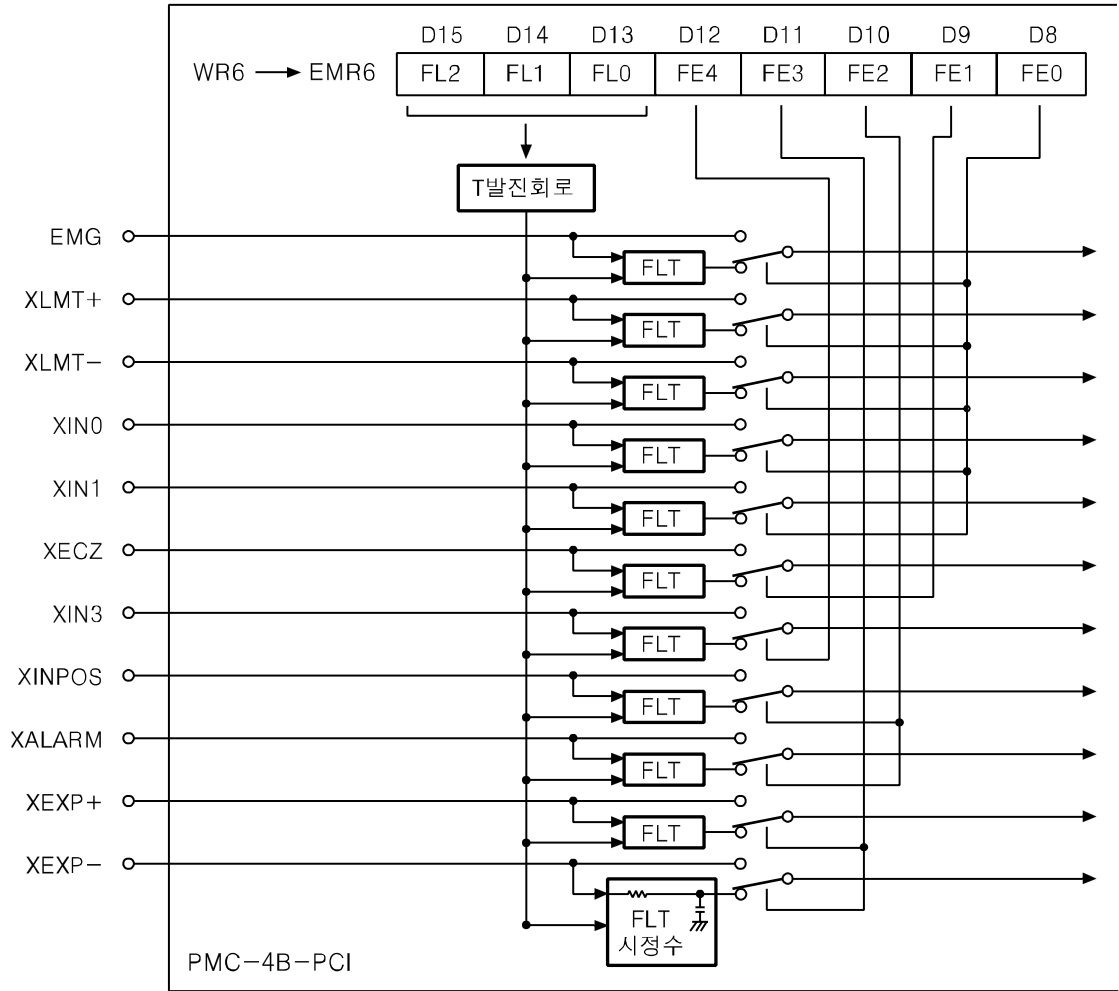
**(2) 보간 Drive 의 Interrupt**

( )안은 Interrupt clear 방법

허가/금지의 설정 WR5 register	발생의 확인 RR0 register	Interrupt 발생 요인
D14 (CIINT)	D9 (CENEXT)	연속 보간 Drive 로 다음 보간 세그먼트의 데이터와 보간 Drive 명령이 기록 가능하게 되었습니다. (다음 보간 Drive 명령을 기록하면 Interrupt 는 Clear 됩니다.)
D15 (BPINT)	D14, 13 (BPS1, 0)	Bit pattern 보간에서 스택 Counter (SC)가 2 에서 1 로 변하고, 다음 BP 데이터의 스택이 가능합니다. (BP 데이터를 스택한다면 Interrupt 는 Clear 됩니다.)

보간 Drive 중 발생하는 Interrupt 는 보간 Interrupt clear 명령(3Dh)으로도 해제할 수 있습니다. 또한, INTN 출력 신호가 Low 이면 보간 Drive 종료로 해제되어 Hi-Z 로 돌아옵니다. 보간 Drive 의 Interrupt 사용방법에 대해서는 Bit pattern 보간, 연속 보간을 참조하십시오.





<그림 2.43 입력 신호 필터 회로 개념도>

노이즈 Duty 비 (신호대 노이즈 발생 시간의 비율)가 어떠한 경우에도 1/4 이하인 것이 조건입니다. 각 입력 신호의 필터 기능을 유효하게 하든지, 입력 신호를 필터를 거치지 않고 통과시킬 것인지는 아래 표와 같이 확장 Mode 설정 명령(60h)의 WR6 register D12~8 (FE4~0) bit 로 설정합니다. 각 Bit 에 1 을 설정하면 그 신호의 필터 기능이 유효가 됩니다.

지정 Bit	유효의 신호
WR6/D8 (FE0)	EMG※1, nLMT+, nLMT-, nIN0, nIN1
WR6/D9 (FE1)	nECZ
WR6/D10 (FE2)	INPOS, nALARM
WR6/D11 (FE3)	nEXPM
WR6/D12 (FE4)	nIN3

※1. EMG 신호는 X 축의 WR6 register D8 bit 로 설정합니다.

**(1) 입력 신호 필터의 설정 예**

EMG 와 X, Y 축의 LMT+, LMT-, IN0, IN1, EXP+, EXP- 입력 신호에 512  $\mu$ s 시간 지연 필터를 설정하고 X, Y 축의 다른 입력 신호는 필터기능을 설정하지 않습니다.

Z, U 축의 LMT+, LMT-, IN0, IN1, EXP+, EXP- 입력 신호에 2 ms 시간 지연 필터를 설정하고 Z, U 축의 다른 입력 신호는 필터 기능을 설정하지 않습니다.

WR6←4900h write X, Y 축 확장 Mode 설정

WR6 에 입력신호 필터의 Mode 를 기록

D15~13	010	필터지연: 512 $\mu$ s
D12	0	IN3 신호: 필터 무효
D11	1	EXP+, EXP-: 필터 유효
D10	0	INPOS, ALARM 신호: 필터 무효
D9	0	ECZ 신호: 필터 무효
D8	1	EMG, LMT+, LMT-, IN1, IN0 신호: 필터 유효

D7~D0 내장 필터 기능 이외의 Mode (적정값을 설정하십시오.)

'6.16 확장 Mode 설정'을 참조하십시오.

WR7←0000h write 자동 원점 복귀를 실시하는 경우는 적정값을 설정하십시오.

'2.5 자동 원점 복귀 출력'을 참조하십시오.

WR0←0360h write X, Y 축으로 확장 Mode 설정 명령 기록

Z, U 축 확장 Mode 설정

WR6←8900h write WR6 에 입력 신호 필터의 Mode 를 기록

D15~D13	100	필터 지연: 2 ms
D12	0	IN3 신호: 필터 무효
D11	1	EXP+, EXP- 신호: 필터 유효
D10	0	
D9	0	
D8	1	

D7~D0 내장 필터 기능 이외의 Mode (적정값을 설정하십시오.)

'6.16 확장 Mode 설정'을 참조하십시오.

WR7←0000h write 자동 원점 복귀를 실시하는 경우는 적정값을 설정하십시오.

'2.5 자동 원점 복귀 출력'을 참조하십시오.

WR0←0C60h write Z, U 축으로 확장 Mode 설정 명령 기록



## 5.9 그 밖의 기능

### 5.9.1 외부 신호에 의한 Drive 조작

정량 Drive 나 연속 Drive 를 Command 로 설정하지 않고 신호 입력을 이용하여 시작시키는 기능입니다. 시스템 그리고 제어하는 모터의 축이 많아지면 각 축의 단순 조그 이동과 매뉴얼 조작을 하나의 모션 컨트롤 IC 가 모두 실시하려고 한다면 CPU 의 부담이 커져, 충분한 응답을 할 수 없게 될 가능성이 있습니다. 외부 신호에 의한 Drive 조작 기능으로 이러한 CPU 의 부담을 경감할 수가 있습니다. 또한 2 상 Encoder 신호를 입력해 각 축의 조그 이동을 실시할 수가 있습니다. 각 축은 nEXP+와 nEXP-의 조작 신호 입력을 가지고 있습니다.

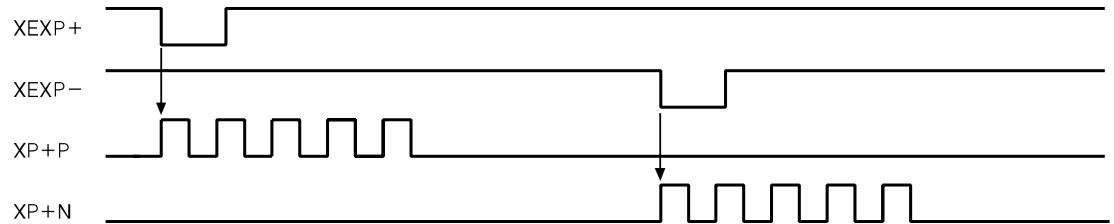
정량 Drive mode, 연속 Drive mode 에서는 nEXP+ 신호는 +방향, nEXP- 신호는 -방향의 Drive 조작을 합니다. WR3 register 의 D4, D3 bit 로 정량 Drive 또는 연속 Drive 를 설정합니다. 또, 정량 Drive 혹은 연속 Drive 에 필요한 Parameter 는 Command 에 의한 기동과 같이 미리 설정합니다. nEXP+와 nEXP- 신호는 일반적으로 High level 로 설정합니다. 2 상 Encoder 신호 Mode(2 상 펄스 입력 신호)에서는 nEXP+ 입력에 A 상 신호를 nEXP- 입력에 B 상 신호를 접속합니다.

#### (1) 정량 Drive mode

WR3 register 의 D4, 3 bit 를 1, 0 으로 설정하여 Drive 에 필요한 속도 Parameter, 출력 Pulse 수를 설정합니다.

nEXP+ 신호를 High level 에서 Low level 로 변경하면 +방향의 정량 Drive 가 시작합니다. nEXP- 신호의 경우 High level 에서 Low level 로 변경하면 -방향의 정량 Drive 가 시작합니다.

각 입력 조작 신호의 Low level 폭은 최소 4CLK 사이클 이상 필요합니다.

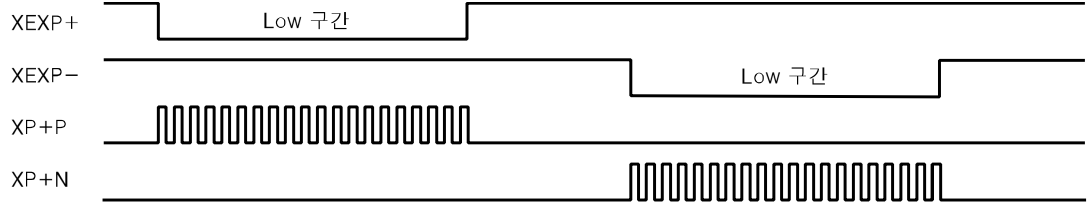


<그림 2.44 외부 조작 신호에 의한 출력 Pulse 5 의 정량 Drive 의 예>

**(2) 연속 Pulse drive mode**

WR3 register 의 D4, D3 bit 를 0, 1 으로 설정하여 Drive 에 필요한 속도 Parameter 를 설정합니다. nEXP+ 신호를 High level 에서 Low level 로 변경하면 설정된 Low level 의 폭으로 연속적인 +방향의 Drive pulse 를 출력합니다. nEXP+ 신호를 Low 에서 High level 로 변경하면 가감속 Drive 는 감속정지, 정속 Drive 는 즉시 정지합니다.

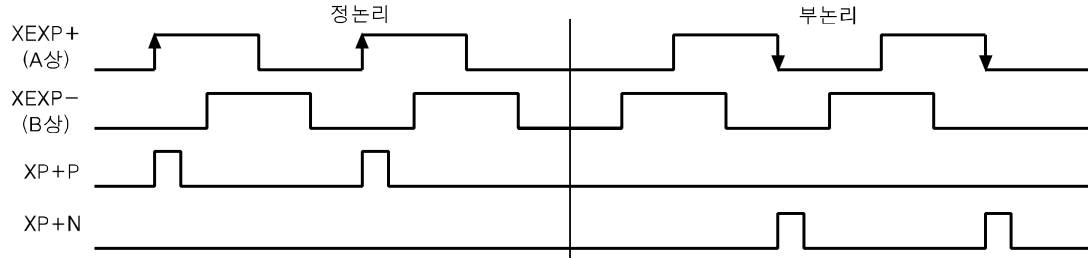
nEXP- 신호의 경우도 마찬가지로 -방향의 Drive pulse 를 연속해서 출력합니다.



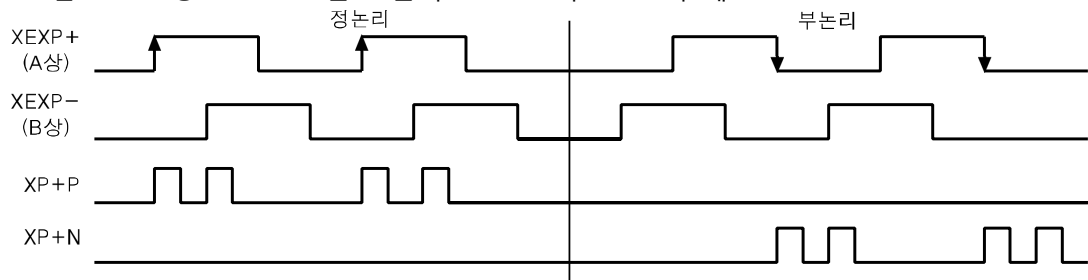
<그림 2.45 외부 조작 신호에 의한 연속 Pulse drive 의 예>

**(3) 2 상 Encoder 신호 Mode**

WR3 register 의 D4, D3 bit 를 1,1 로 설정하여 Drive 에 필요한 속도 Parameter, 출력 Pulse 수를 설정합니다. A 상 신호를 nEXP+ 입력에, B 상 신호를 nEXP- 입력에 접속합니다. nEXP- 신호가 Low level 일 때, nEXP+ 신호가 High level 로 변경될 경우(상승에지)는 +정량 Drive, nEXP- 신호가 Low level 일 때 nEXP+ 신호가 Low level 로 변경되는 경우(하강 에지)는 -정량 Drive 가 기동합니다. 출력 Pulse 수의 설정이 1 이면 nEXP+ 신호의 각 상승, 하강 에지에서 1 개의 Drive pulse 를 출력합니다. 출력 Pulse 수의 설정이 P 이면 P 개의 Drive pulse 를 출력합니다.



<그림 2.46 2 상 Encoder 신호 출력 Pulse 1 의 Drive 의 예>



<그림 2.47 2 상 Encoder 신호 출력 Pulse 2 의 Drive 의 예>

nEXP+ 신호의 Hi, Low 에지에서 다음 신호의 High, Low 에지 사이에 P 개의 Drive pulse 를 출력하기 위해서 속도 Parameter 는 다음과 같은 조건으로 설정합니다.

$$V \geq F \times P \times 2$$

V: Drive 속도 (pps)

P: 출력 Pulse

F: 2 상 Encoder 신호의 최고속시의 주파수(Hz)



**Ex.**

2 상 Encoder 신호의 최고 주파수를  $F=500\text{Hz}$ , 출력 Pulse 는  $P=1$  로 하면, Drive 속도  $V=1000\text{pps}$  이상의 값으로 설정합니다. 또 가감속 Drive 는 실시하지 않으므로 기동속도 SV 는 Drive 속도 V 와 같은 값으로 설정합니다. 다만, 구동 모터가 Step 모터인 경우는 모터의 자기동 주파수를 넘지 않는 범위 내에서 Drive 속도를 설정합니다.

### 5.9.2 Pulse 출력 방식의 선택

Drive 출력 방식은 아래 표와 같이 2 개의 Pulse 출력 방식을 선택할 수가 있습니다. 독립 2Pulse 방식은 +방향 Drive 시에는  $nP+P/nP+N$  에, -방향 Drive 시에는  $nP-P/nP-N$  에 Drive pulse 를 출력합니다.

또 1Pulse 방식은  $nP+P/nP+N$  로 Drive pulse 를 출력,  $nP-P/nP-N$  은 방향을 결정합니다.

(Pulse/방향은 정논리 설정 시)

Pulse 출력 방식	드라이브 방향	출력 신호 파형	
		$nP+P/nP+N$ 신호	$nP-P/nP-N$ 신호
독립 2 Pulse 방식	+ 방향 드라이브 출력 시		Low level
	- 방향 드라이브 출력 시	Low level	
1 Pulse 방식	+ 방향 드라이브 출력 시		Low level
	- 방향 드라이브 출력 시		High level

Pulse 출력 방식의 선택은 WR2 register 의 D6(PLSMD) bit 로 설정합니다. 또한 Pulse 출력, 방향 설정 및 논리 Level 도 선택할 수가 있습니다.



**Note**

1 pulse 방식의 경우는 Pulse 신호( $nPLS$ )와 방향 신호( $nDIR$ )가 출력되는 타이밍을 '14.2 Drive 시작/종료 시', '14.3 보간 Drive 시'를 참조하십시오.

### 5.9.3 Pulse 입력 방식의 선택

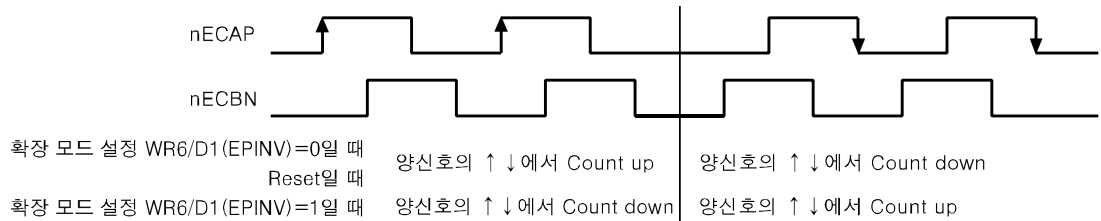
실제위치 Counter 의 Up/Down count 입력이 되는 Encoder pulse 입력은 2 상 Pulse 입력과 Up/Down pulse 입력 중에서 선택할 수가 있습니다.

#### (1) 2 상 Pulse 입력 Mode

WR2 register 의 D9(PINMD) bit 를 0 으로 설정하면 2 상 Pulse 입력 Mode 가 됩니다. 이 Mode 에서는 정논리 Pulse 로 A 상이 진행되고 있을 때는 Count up 하고 B 상이 진행되고 있을 때는 Count down 합니다.

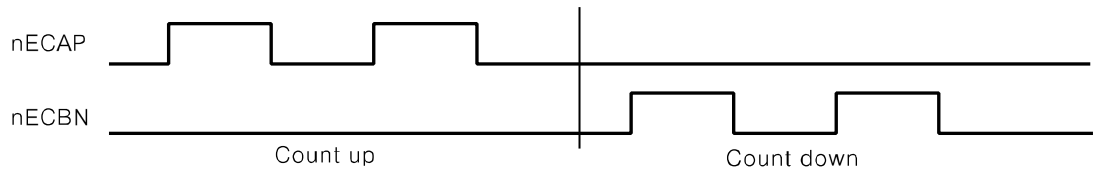
양 신호의 상승에지, 하강에지에서 Count up, down 합니다. 확장 Mode 설정으로 실제 위치 Counter 증감 반전 bit (WR6/D1)를 1 로 하면 실제위치 Counter 의 Up/Down 동작이 반대로 됩니다. '6.16 확장 Mode 설정'을 참조하십시오.

또한 2 상 Pulse 입력 Mode 에서는 입력 Pulse 를 1/2, 1/4 로 분주시킬 수도 있습니다.



#### (2) Up/Down pulse 입력 Mode

WR2 register 의 D9(PINMD) bit 를 1 로 설정하면 Up/Down pulse 입력 mode 가 됩니다. nECAP/nECAN 가 Count up 입력, nECBP/nECBN 가 Count down 입력이 됩니다. 각각 Pulse 의 ↑에서 Count 합니다.



Pulse 입력 방식의 선택은 WR2 register 의 D9(PINMD) bit 로 Encoder 2 상 Pulse 입력의 분주비는 D11, D10(PIND1, 0) bit 로 설정합니다.



#### Note

입력 Pulse 의 Pulse 폭, Pulse 주기 등에는 시간 규정이 있습니다.

### 5.9.4 하드웨어 Limit 신호

하드웨어 Limit 신호 (nLMT+, nLMT-)는 +방향/-방향의 drive pulse 를 정지시키는 신호 입력입니다. Limit 신호의 논리 Level 과 Limit 신호가 Active 되었을 때 감속 정지 또는 즉시 정지는 Command 로 선택할 수가 있습니다. WR2 register 의 D3, D4(HLMT+, HLMT-), D2(LMTMD) bit 로 설정합니다.

### 5.9.5 서보(Servo) 모터 Driver 대응의 신호

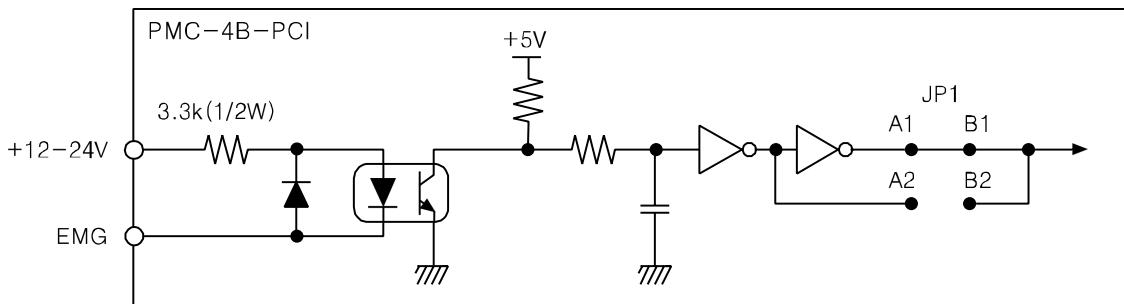
서보 모터 Driver 와의 접속을 위한 입력 신호로서 위치 결정 완료 신호를 입력하는 nINPOS 와 Alarm 신호를 입력하는 nALARM 이 있습니다. 각각의 신호의 유효/무효 및 논리 Level 을 설정할 수가 있습니다.

설정은 WR2 register 의 D15~D12 bit 로 실시합니다. nINPOS 입력 신호는 서보모터 Driver 의 위치 결정 완료 신호에 대응합니다. 유효로 설정하면 Drive 종료 후, nINPOS 입력 신호가 Active 되면, RR0 주 status register 의 n-DRV bit 가 0 으로 돌아옵니다. nALARM 입력 신호는 서보모터 Driver 로부터 Alarm 신호를 수신합니다. 유효로 설정하면 nALARM 입력 신호를 항상 감시하며, Active 상태의 경우는 RR2 register 의 D4(ALARM) bit 가 1 이 됩니다. Drive 중이면 Drive 를 즉시 정지합니다.

이러한 서보모터 Driver 용 입력 신호는 RR5, 6 register 로 그 상태를 상시 읽어낼 수가 있습니다.

### 5.9.6 긴급정지

4 축 모든 Drive 를 긴급정지시키기 위한 입력 신호로서 ENG 신호가 있습니다. ENG 신호는 일반적으로 High level 입니다. Low level 이 되면 Drive 중의 전축이 즉시정지하고 전축의 RR2 register 의 D5(EMG) Bit 가 1 이 됩니다. ENG 신호는 논리 Level 를 선택할 수가 없기 때문에 PMC-4B-PCI Board 의 JP1 점퍼핀을 이용하여 논리레벨을 설정합니다.

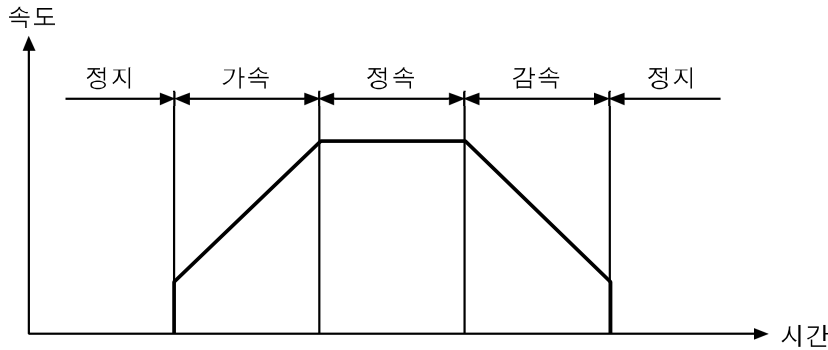


4 축에 대해서 긴급정지를 설정하기 위해서는 다음과 같은 방법이 있습니다.

- 4 축에 대해 동시에 즉시 정지 명령을 발행합니다.  
WR0 register 에 4 축 모두를 지정하여 즉시 정지 명령(27h)을 기입합니다.
- 소프트웨어 Reset 을 설정합니다.  
WR0 register 에 8000h 를 기입하면 소프트웨어 Reset 가 걸립니다.

### 5.9.7 Drive 상태의 출력

각 축의 Drive 중/정지 상태는 RR0 register D3~0(n-DRV) bit 에 출력됩니다.



각 축 Drive 중 Drive 속도의 가속/정속/감속 상태는 각 축의 RR1 register 의 D2(ASND), D3(CNST), D4(DSND) bit 와 nOUT6/ASND, nOUT7/DSND 신호에 출력됩니다. 다만 신호출력은 범용 출력 신호와 단자를 겸용하고 있기 때문에 Drive 상태를 출력하려면 WR3 register 의 D7(OUTSL) bit 를 1 으로 합니다.

PMC-4B-PCI 에서는 범용 출력 신호 단자로는 사용할 수 없습니다.

각 축 Drive 중 Drive 속도의 가속/정속/감속 상태는 각 축의 RR1 register 의 D2(ASND), D3(CNST), D4(DSND) bit 와 nOUT6/ASND, nOUT7/DSND 신호에 출력됩니다. 다만 신호출력은 범용 출력 신호와 단자를 겸용하고 있기 때문에 Drive 상태를 출력하려면 WR3 register 의 D7(OUTSL) bit 를 1 으로 합니다.

PMC-4B-PCI 에서는 범용 출력 신호 단자로는 사용할 수 없습니다.

드라이브 상태	레지스터				출력신호	
	RR0/n-DRV	nRR1/ASND	nRR1/CNST	nRR1/DSND	nOUT6/ASND	OUT7/DSND
정지	0	0	0	0	Low	Low
가속	1	1	0	0	High	Low
정속	1	0	1	0	Low	Low
감속	1	0	0	1	Low	High

또한 S 자 가감속 Drive 에 있어서 가속도, 감속도의 증가/일정/감소 상태도 RR1 register 의 D5(AASND), D6(ACNST), D7(ADSND) bit 로 출력됩니다.

### 5.9.8 범용 출력 신호

각 축 모두 nOUT7~4 의 4 개의 범용 출력 신호를 가지고 있지만, 위치 비교출력, Drive 상태 출력과 단자를 공용하고 있기 때문에 출력으로 사용하는 경우는 사용할 수 없습니다. nOUT7~4 신호는 WR3 register 의 D7(OUTSL)로 범용 출력을 사용하는 Mode 로 설정합니다. 그리고 W3 register 의 D11~8 (OUT7~4) 의 각 Bit 에 출력 Level 의 값을 설정하면 출력됩니다. 범용 출력 신호는 모터 Driver 의 여자 OFF, Alarm reset 등에 사용할 수가 있습니다.

Reset 시에는 nWR3 register 의 각 bit 는 clear 되어 모든 출력은 Low level 로 됩니다.

## 6 각 신호의 설명

### (1) 신호의 설명

신호명의 X○○○○, Y○○○○, Z○○○○, U○○○○은 각각 X 축, Y 축, Z 축, U 축의 입/출력 신호입니다.

또한 n○○○의 "n"은 X, Y, Z, U 를 표현하고 있습니다. "-F-"기호가 붙은 입력 신호는 모션 컨트롤 IC 내부의 입력단에 적분 필터 회로를 가지고 있습니다. 필터 기능에 대해서는 '2.8 입력 신호 필터'를 참조하십시오.

신호	단자번호	입/출력	신호의 설명
XP+P / XP+N YP+P / YP+N ZP+P / ZP+N UP+P / UP+N	A38/A39 A78/A48 B38/B39 B47/B48	출력	Pulse+/Pulse: +방향의 Drive pulse 를 출력합니다. Reset 시의 상태는 Low 레벨이며 Drive 동작으로 들어가면 듀티비 50%(정속시)의 정 Pulse 가 출력됩니다. 정 Pulse/부 Pulse 는 Mode 로 선택할 수 있습니다. 또한, Mode 선택으로 1Pulse 방식이 선택되었을 경우에는 본 단자의 Drive pulse 가 출력됩니다.
XP-P / XP-N YP-P / YP-N ZP-P / ZP-N UP-P / UP-N	A40/A41 A49/A50 B40/B41 B49/B50	출력	Pulse- / Direction: -방향의 Drive pulse 를 출력합니다. Reset 시의 상태는 Low level 이며 Drive 동작으로 들어가면 듀티비 50%(정속시)의 정 Pulse 가 출력됩니다. 정 Pulse/부 Pulse 는 Mode 로 선택할 수 있습니다. 또한, Mode 선택으로 1Pulse 방식이 선택되었을 경우에는 본 단자는 방향신호가 됩니다.
XECAP / XECAN YECAP / YECAN ZECAP / ZECAN UECAP / UECAN	A15/A16 A23/A24 B15/B16 B23/B24	입력	Encoder-A / Pulse +in : Encoder A 상 신호의 입력으로 모션 컨트롤 IC 내부에서 Up/Down pulse 로 변환되어 실제위치 Counter 의 Count 입력이 됩니다. 또한 Mode 선택을 Up/Down pulse 입력으로 선택하면 Up pulse 입력으로 설정되며 입력 Pulse 의 ↑에서 실제위치 Counter 가 Count up 됩니다.
XECBP / XECBN YECBP / YECBN ZECBP / ZECBN UECBP / UECBN	A17/A18 A25/A26 B17/B18 B25/B26	입력	Encoder-B / Pulse-in: Encoder B 상 신호의 입력으로 모션 컨트롤 IC 내부에서 Up/Down pulse 로 변환되어 실제위치 Counter 의 Count 입력이 됩니다. 또한 Mode 선택을 Up/Down pulse 입력으로 선택하면 Down pulse 입력으로 설정되며 입력 Pulse 의 ↑에서 실제위치 Counter 가 Count down 됩니다.
XOUT7 /	A37	출력 A	Universal Output 7/ Descend : 범용 출력 신호

신호	단자번호	입/출력	신호의 설명
DSND YOUT7 / DSND ZOUT7 / DSND UOUT7 / DSND	A46 B37 B46		입니다. nOut7~4 출력은 WR0 register 로 축 지정 후, WR3 register 의 D11~8 로 출력 level 을 결정합니다. Reset 시는 Low 가 됩니다. Mode 선택으로, Drive 상태 출력 Mode 로 하면 감속 Drive 출력 신호가 됩니다. Drive 명령 실행 중 감속상태가 되면 High level 이 됩니다.
XOUT6 / ASND YOUT6 / ASND ZOUT6 / ASND UOUT6 / ASND	A36 A45 B36 B45	출력 A	Universal Output 6 / Ascend: 범용 출력 신호 입니다. 설정은 nOUT7 과 같습니다. Mode 선택으로, Drive 상태 출력 Mode 로 하면 가속 Drive 출력 신호가 됩니다. Drive 명령 실행 중 가속 상태가 되면 High level 이 됩니다.
XOUT5 / CMPM YOUT5 / CMPM ZOUT5 / CMPM UOUT6 / CMPM	A35 A44 B35 B44	출력 A	Universal Output 5 / Compare -: 범용 출력 신호 입니다. 설정은 nOUT7 과 같습니다. Mode 선택으로, Drive 상태 출력 Mode 로 하면 논리/실제위치 Counter 가 COMP- register 보다 작을 때 High level 로, 클 때 Low level 이 됩니다.
XOUT4 / CMPP YOUT4 / CMPP ZOUT4 / CMPP UOUT4 / CMPP	A34 A43 B34 B43	출력	Universal Output 4 / Compare -: 범용 출력 신호 입니다. 설정은 nOUT7 과 같습니다. Mode 선택으로, Drive 상태 출력 Mode 로 하면 논리/실제위치 Counter 가 COMP+ register 보다 클 때 High level 로, 작을 때 Low level 이 됩니다.
XINPOS YINPOS ZINPOS UINPOS	A13 A21 B13 B21	입력 -F-	Inposition: 서보 모터 Driver 의 Inposition (위치 결정 완료)출력에 대응하는 입력 신호입니다. 유효/무효, 논리 Level 은 Command 로 설정할 수 있습니다. 유효로 설정하면 Drive 종료 후 이 신호가 Active 되는 것을 기다리고 나서 주 status register 의 n-DRV bit 가 0 으로 돌아옵니다.
XALARM YALARM ZALARM	A14 A22 B14	입력 -F-	Servo Alarm: 서보 모터 Drive 의 Alarm 출력에 대응하는 입력신호입니다. 유효/무효, 논리 Level 은



신호	단자번호	입/출력	신호의 설명
UALARM	B22		Mode 로 선택할 수가 있습니다. 유효로 설정하고 신호가 Active level 이 되면 RR2 register 의 ALARM bit 가 1 이 됩니다.
XLMT+ YLMT+ ZLMT+ ULMT+	A3 A8 B3 B8	입력 -F-	Over Run Limit +: +방향의 Over run limit 신호입니다. +방향의 Drive pulse 출력 중 Active 하면 Drive 는 감속정지 또는 즉시 정지합니다. 필터 기능이 무효인 경우, 2CLK 이상의 Active pulse 폭이 필요합니다. 감속정지/즉시정지, 논리 Level 을 Mode 로 선택할 수가 있습니다. 또한, 이 신호가 Active level 이 되면 RR2 register 의 HLMT+ bit 가 1 이 됩니다.
XLMT- YLMT- ZLMT- ULMT-	A4 A9 B4 B9	입력 -F-	Over Run Limit -: -방향의 Over run limit 신호입니다. -방향의 Drive pulse 출력 중 Active 하면 Drive 는 감속정지 또는 즉시 정지합니다. 필터 기능이 무효인 경우, 2CLK 이상의 Active pulse 폭이 필요합니다. 감속정지/즉시정지, 논리 Level 을 Mode 로 선택할 수가 있습니다. 또한, 이 신호가 Active level 이 되면 RR2 register 의 HLMT+ bit 가 1 이 됩니다.
XIN0 YIN0 ZIN0 UIN0	A6 A11 B6 B11	입력 -F-	원점 근접 서치 동작의 입력 신호로 Active level 을 선택하여 설정합니다. 필터 기능이 무효인 경우, 2CLK 이상의 Active pulse 폭이 필요합니다. RR4/RR5 register 로 신호의 상태를 읽어낼 수가 있습니다. 오픈콜렉터 입력입니다. '11.3 Encoder 입력 신호의 접속'을 참조하십시오.
XECAP/N YECAP/N ZECAP/N UECAP/N	A19~A20 A27~A28 B19~B20 B27~B28	입력 -F-	Encoder-C ECZ 는 encoder Z 상 신호에 할당됩니다.
XIN3, XIN1~0 YIN3, YIN1~0 ZIN3, ZIN1~0 UIN3, UIN1~0	A7, A5, A6, A12, A10, A11 B7, B5, B6, B12, B10, B11	입력 -F-	Input 3~0: Drive 도중에 감속정지 또는 즉시 정지시키기 위한 신호입니다. 서치 동작의 입력신호로 사용합니다. 필터 기능이 무효인 경우, 2CLK 이상의 Active pulse 폭이 필요합니다. IN3~IN1 각각에 대하여 유효/무효, 논리 Level 을 설정할 수가 있습니다. 자동 원점 복귀에서는 IN1 는 원점 신호에, IN0 은 원점 근접 신호에 할당됩니다. 또한 RR4/RR5 register 로 신호의 상태를 읽어낼 수가 있습니다.
XEXP+ YEXP+ ZEXP+	A29 A31 B29	입력 -F-	External Operation +: 외부에서 +방향의 Drive 를 기동하는 신호입니다. 외부 정량 Pulse drive mode 로 하면, 신호의 ↓에서 +정량 Pulse drive 가

신호	단자번호	입/출력	신호의 설명
UEXP+	B31		기동합니다. 외부 연속 Pulse drive mode 로 하면, Low level 에서 +연속 Pulse drive 를 기동합니다. 2 상 Encoder 신호 Mode 의 경우는 Encoder A 상 신호를 입력합니다.
XEXP- YEXP- ZEXP- UEXP-	A30 A32 B30 B32	입력 -F-	External Operation -: 외부에서 -방향의 Drive 를 기동하는 신호입니다. 외부 정량 Pulse drive mode 로 하면, 신호의 ↓에서 -정량 Pulse drive 가 기동합니다. 외부 연속 Pulse drive mode 로 하면, Low level 에서 -연속 Pulse drive 를 기동합니다. 2 상 Encoder 신호 Mode 의 경우는 Encoder B 상 신호를 입력합니다.
EMG	A2	입력 -F-	Emergency Stop: 전 축의 Drive 를 긴급 정지시키는 입력신호입니다. 이 신호를 Low level 로 하면 보간 Drive 를 포함하여 전 축의 Drive 가 즉시 정지하면 각 축의 RR2 register 의 EMG bit 가 1 이 됩니다. 필터 기능이 무효인 경우, 2CLK 이상의 Low level pulse 폭이 필요합니다. [주의] 이 신호는 JP1 으로 논리 Level 을 선택할 수 있습니다.

## 7 Read/Write register

이 장에서는 각 축을 제어하기 위한 Read/Write register 에 대해서 상세하게 기술합니다. Bit pattern 보간용 register (BP1P/M, BP2P/M, BP3P/M)에 대해서는 2.4.3 절의 Bit pattern 보간을 참조하십시오.

### 7.1 16 bit data bus의 Register address

아래 표와 같이 16 bit data bus 를 사용하는 경우에는 16 bit 의 Read/write register 를 액세스하는 Address 가 있습니다.

#### (1) 16 bit data bus 에 있어서 Write register

ADDRESS A2 A1 A0	RESISTER 기호	REGISTER	내용
0 0 0	WR0	Command register	축 지정, 명령 코드 설정
0 0 1	XWR1 YWR1 ZWR1 UWR1	X 축 Mode register 1 Y 축 Mode register 1 Z 축 Mode register 1 U 축 Mode register 1	각 축의 외부 감속 정지 신호의 논리 Level 의 유효/무효 설정 각 축의 Interrupt 허가/금지 설정
0 1 0	XWR2 YWR2 ZWR2 UWR2	X 축 Mode register 2 Y 축 Mode register 2 Z 축 Mode register 2 U 축 Mode register 2	각 축의 Limit 신호의 Mode 설정 Drive pulse 의 Mode 설정 Encoder 입력 신호의 Mode 설정 서보 모터용 신호의 논리 Level 의 유효/무효 설정
	BP1P	BP1P register	Bit pattern 보간 제 1 축 +방향 bit data 설정
0 1 1	XWR3 YWR3 ZWR3 UWR3	X 축 Mode register 3 Y 축 Mode register 3 Z 축 Mode register 3 U 축 Mode register 3	각 축의 매뉴얼 감속, 감속도, S 자 가감속 Mode 설정 외부조작 Mode 설정 범용 출력 OUT7~4 설정
	BP1M	BP1M register	Bit pattern 보간 제 1 축 -방향 bit data 설정
1 0 0	WR4	Output register	범용 출력 OUT3~0 설정 (사용안함)
	BP2P	BP2P register	Bit pattern 보간 제 2 축 +방향 bit data 설정
1 0 1	WR5	보간 Mode register	축 지정. 선속일정 Mode, Step 출력 Mode, Interrupt 설정
	BP2M	BP2M register	Bit pattern 보간 제 2 축 +방향 bit data 설정
1 1 0	WR6	Write data register 1	범용 출력 OUT3~0 설정 (사용안함)
	BP3P	BP3P register	Bit pattern 보간 제 2 축 +방향 bit data

ADDRESS A2 A1 A0			RESISTER 기호	REGISTER	내용
					설정
1	1	1	WR7	Write data register 2	Write data 상위 16 bit (D31~D16) 설정
			BP3M	BP3M register	Bit pattern 보간 제 3 축 -방향 bit data 설정

- 위의 표와 같이 각 축 모두 WR1, WR2, WR3 (Mode register 1, 2, 3) 를 가지고 있습니다. 이러한 Register 는 동일 Address 에 기입하여 실시하게 됩니다. 어느 축의 Mode register 에 기입할지는 직전에 설정한 축 지정에 의해 정해집니다.
- Bit pattern 보간용의 Bit data register BP1P~3P, BP1M~3M 은 Reset 직후에는 기입할 수가 없습니다. 이러한 Register 의 기입은 BP register 기입 가능 명령 (36h)으로 가능하게 됩니다.  
BP register 기입 가능명령 (36h)을 발행 후, nWR2~3 의 기입을 할 수 없게 되기 때문에 Bit pattern 보간으로 Bit data 의 기입이 끝나면 BP register 기입 불가 명령 (37h)을 발행해야 합니다.
- WR6 register 와 BP3P register, WR7 register 와 BP3M register 에 대해서는 하드웨어적으로 동일 Register 를 공유하고 있으므로 주의하여 주십시오.
- Reset 은 nWR1, nWR2, nWR3, WR4, WR5 register 의 모든 bit 를 0 으로 clear 시킵니다. 그 외의 Register 는 적용되지 않습니다.

**(2) 16 bit data bus 에 있어서의 Read register**

모든 Register 는 16 bit 입니다.

ADDRESS A2 A1 A0	RESISTER 기호	REGISTER	내용
0 0 0	RR0	주 Status register	각 축의 Drive, 에러상태를 표시 보간의 Drive, 연속 보간 시 다음 Data 설정 원호보간의 상한 설정 BP 보간 시 Stack counter 표시
0 0 1	XRR1 YRR1 ZRR1 URR1	X 축 Status register 1 Y 축 Status register 1 Z 축 Status register 1 U 축 Status register 1	위치: COMP register 비교 가/감속 상태, 가/감속 증가/감소 상태 표시 종료 Status 의 표시
0 1 0	XRR2 YRR2 ZRR2 URR2	X 축 Status register 2 Y 축 Status register 2 Z 축 Status register 2 U 축 Status register 2	에러 발생 요인 표시 자동 원점 복귀 실행 Status 표시
0 1 1	XRR3 YRR3 ZRR3 URR3	X 축 Status register 3 Y 축 Status register 3 Z 축 Status register 3 U 축 Status register 3	Interrupt 발생 요인 표시
1 0 0	RR4	Input register 1	X 축, Y 축 입력 신호 상태 표시
1 0 1	RR5	Input register 2	Z 축, U 축 입력 신호 상태 표시
1 1 0	RR6	Read data register 1	Read data 하위 16bit (D15~D0) 표시
1 1 1	RR7	Read data register 2	Read data 상위 16bit (D31~D16) 표시

Write register 와 같이 각 축 모두 RR1, RR2, RR3 (각 축 status register 1, 2, 3)를 가지고 있습니다. 이러한 Register 는 동일 Address 로 읽기를 실시하게 됩니다. 어느 축의 Status register 를 읽을 지는 직전에 설정한 축 지정에 의해 정해집니다.

## 7.2 8 bit data bus의 Register address

8 bit data bus 로 액세스 하는 경우는 16 bit register 를 상위 바이트, 하위 바이트로 나누어 액세스 합니다. 아래 표에서, \*\*\*\*L 는 16 bit register\*\*\*\*의 하위 바이트(D7~D0), \*\*\*\*H 는 16bit register\*\*\*\*의 상위 바이트(D15~D8)를 나타내고 있습니다. Command register (WR0L, WR0H) 만은 반드시 상위 바이트(WR0H)를 먼저, 하위 바이트 (WR0L)를 나중에 기입합니다.

### (1) 8 bit data bus 에 있어서의 Write register

ADDRESS				Write 하는 Register
A3	A2	A1	A0	
0	0	0	0	WR0L
0	0	0	1	WR0H
0	0	1	0	XWR1L, YWR1L, ZWR1L, UWR1L
0	0	1	1	XWR1H, YWR1H, ZWR1H, UWR1H
0	1	0	0	XWR2L, YWR2L, ZWR2L, UWR2L
0	1	0	1	XWR2H, YWR2H, ZWR2H, UWR2H
0	1	1	0	XWR3L, YWR3L, ZWR3L, UWR3L
0	1	1	1	XWR3H, YWR3H, ZWR3H, UWR3H
1	0	0	0	WR4L, BP2PL
1	0	0	1	WR4H, BP2PH
1	0	1	0	WR5L, BP2ML
1	0	1	1	WR5H, BP2MH
1	1	0	0	WR6L, BP3PL
1	1	0	1	WR6H, BP3PH
1	1	1	0	WR7L, BP3ML
1	1	1	1	WR7H, BP3MH

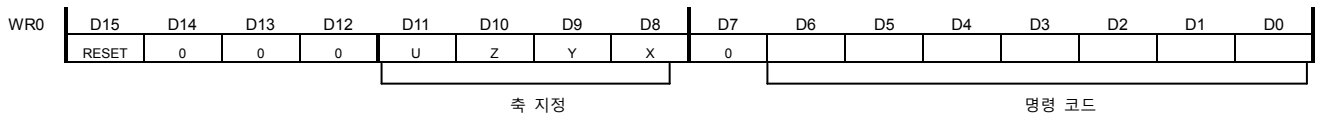
### (2) 8 bit data bus 에 있어서의 Read register

ADDRESS				Write 하는 Register
A3	A2	A1	A0	
0	0	0	0	RR0L
0	0	0	1	RR0H
0	0	1	0	XRR1L, YRR1L, ZRR1L, URR1L
0	0	1	1	XRR1H, YRR1H, ZRR1H, URR1H
0	1	0	0	XRR2L, YRR2L, ZRR2L, URR2L
0	1	0	1	XRR2H, YRR2H, ZRR2H, URR2H
0	1	1	0	XRR3L, YRR3L, ZRR3L, URR3L
0	1	1	1	XRR3H, YRR3H, ZRR3H, URR3H
1	0	0	0	RR4L
1	0	0	1	RR4H
1	0	1	0	RR5L
1	0	1	1	RR5H
1	1	0	0	RR6L
1	1	0	1	RR6H
1	1	1	0	RR7L
1	1	1	1	RR7H

## 7.3 WR0 command register

각 축에 대하여 축 지정용 bit 를 하여 명령을 기입하는 Register 입니다. Register 는 축을 지정하는 bit, 명령 코드를 설정하는 bit 및 Command reset bit 로 구성되어 있습니다.

본 Register 에 축 지정 및 명령 코드를 기입하면 그 명령은 즉시 실행됩니다. Drive 속도 설정 등의 Data 기입 명령은 WR6, 7 register 에 Data 를 먼저 기입해야 합니다. 또한 Data read 명령은 이 Command register 에 명령을 기입하면 내부 회로로부터 RR6, 7 register 에 Data 가 설정됩니다. 8bit data bus 는 반드시 상위 바이트(H)를 먼저, 하위 바이트(L)를 나중에 기입합니다. 모든 명령 코드의 명령 처리에 필요로 하는 시간은 최대 250nsec (CLK=16 MHz 의 경우) 입니다.



D6~0 명령 코드를 설정합니다. 명령 코드는 5 장 이후의 각 명령의 설명을 참조하십시오.

D11~8 실행할 축을 지정합니다. 각 축의 해당 bit 를 1 로 설정하면 그 축이 지정됩니다.

축의 지정은 동시에 여러 축을 지정할 수 있으며 같은 명령을 실행하거나 같은 Parameter 값을 입력할 수 있습니다. 다만 Data 읽기 명령의 경우는 1 축만 지정하십시오.

보간 관계의 명령에서는 축 지정 bit 는 모두 0 으로 하십시오.

D15 RESET 모션 컨트롤 IC 를 Reset 합니다. 이 bit 를 1 로 하면 모션 컨트롤 IC 는 reset 됩니다.

Command 기입 후, 최대 875nsec (CLK=16MHz 의 경우) 동안 모션 컨트롤 IC 의 Register 를 액세스할 수 없습니다. 8bit data bus 의 경우는 WR0H(80h)의 기입으로 Reset 됩니다.

RESET bit 는 일반적으로 명령 기입에서는 반드시 0 으로 둡니다.

1 로 하면 모션 컨트롤 IC 내부 회로의 테스트 명령이 기동하여 생각하지 않는 동작이 발생합니다.

## 7.4 WR1 mode register 1

Mode register1 은 4 축 모두 각각 개별적으로 가지고 있습니다. 어느 축의 Mode register 를 사용할 지는 축 지정에 의해서 정해집니다. Mode register1 은 Drive 도중에 감속정지/즉시정지 입력신호 IN3~IN0 의 유효/무효와 유효일 때의 논리 Level 을 설정하는 Bit 와 Interrupt 의 허가/금지를 설정하는 Bit 로 구성됩니다.

IN3~IN0 을 유효로 하여 정량 Pulse drive 또는 연속 Pulse drive 로 Drive 를 시작하여 지정된 IN 신호가 설정해 둔 논리 Level 이 되면 Drive 는 감속정지 또는 즉시정지 합니다. 가감속 Drive 이면 감속정지, 정속 Drive 이면 즉시 정지합니다.

WR1	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	D-END	C-STA	C-END	P≥C+	P<C+	P>C-	P≥C-	PULSE	IN3-E	IN3-L	IN2-E	IN2-L	IN1-E	IN1-L	IN0-E	IN0-L
	Interrupt 의 허가/금지								드라이브 정지 입력 신호의 유효/무효, 논리							

- D7, 5, 3, 1      INm-E      Drive 정지 입력 신호 INm(m: 0~3)의 유효/무효를 설정하는 bit 입니다.  
0: 무효, 1: 유효
- D6, 4, 2, 0      INm-L      입력 신호 INm 의 유효 시, 논리 Level 을 설정하는 bit 입니다.  
0: Low 에서 정지, 1: High 에서 정지  
자동 원점 복귀에서 사용하는 INm 신호의 논리 Level 을 설정하는 bit 입니다.  
유효/무효 bit (D5, D3, D1)는 무효로 합니다.

이하의 bit 는 Interrupt 허가/금지 bit 입니다. 1 로 하면 Interrupt 허가, 0 으로 하면 Interrupt 금지가 됩니다.

- D8              PULSE      Drive pulse 마다 Pulse 의 ↑로 Interrupt 가 발생합니다.  
(Drive pulse 정논리 설정 시)
- D9              P≥C-      논리/실제위치 Counter 의 값이 COMP- register 의 값보다 클 때  
Interrupt 가 발생합니다.
- D10             P<C-      논리/실제위치 Counter 의 값이 COMP- register 의 값보다 작을 때  
Interrupt 가 발생합니다.
- D11             P<C+      논리/실제위치 Counter 의 값이 COMP+ register 의 값보다 작을 때  
Interrupt 가 발생합니다.
- D12             P≥C+      논리/실제위치 Counter 의 값이 COMP+ register 의 값보다 클 때  
Interrupt 가 발생합니다.
- D13             C-END      가감속 Drive 시에 정속영역에서 Pulse 출력을 종료했을 때,  
Interrupt 가 발생합니다.
- D14             C-STA      가감속 Drive 시에 정속영역에서 Pulse 출력을 시작했을 때, Interrupt 가  
발생합니다.
- D15             D-END      Drive 가 종료했을 때, Interrupt 가 발생합니다.

Reset 시에는 D15~D0 는 모두 0 으로 설정됩니다.



## 7.5 WR2 mode register 2

WR2	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
		INP-E	INP-L	ALM-E	ALM-L	PIND1	PIND0	PINMD	DIR-L	PLS-L	PLSMD	CMPSL	HLMT-	HLMT+	LMTMD	CLMT-

Mode register 2 는 4 축 모두 각각 개별적으로 가지고 있습니다. 어느 축의 Mode register 를 사용할 지는 축 지정에 의해 정해집니다. Mode register 2 는 Limit 입력 신호의 Mode 설정, Drive pulse 의 Mode 설정, Encoder 입력 신호의 Mode 설정 및 서보 모터용 신호의 논리 Level, 유효/무효의 설정을 실시합니다.

**D0 SLMT+** COMP+ register 를 +방향으로 소프트웨어 Limit 로 설정합니다. 1 로 하면 유효, 0 으로 하면 무효가 됩니다. 유효로 하면 +방향의 drive 중, 논리/실제위치 Counter 가 COMP+ register 의 값 보다 크게되어 감속정지 합니다. 또한 RR2 register 의 D0(SLMT+) Bit 가 1 이 됩니다.

이 상태에서 +방향의 Drive 명령은 실행되지 않습니다.

[주의] 확장 Mode 의 위치 Counter 가변 링을 동작시키는 경우에는 소프트웨어 Limit 를 사용할 수가 없습니다.

**D1 SLMT-** COMP- register 를 -방향으로 소프트웨어 Limit 로 설정합니다. 1 로 하면 유효, 0 으로 하면 무효가 됩니다. 유효로 하면 -방향의 drive 중, 논리/실제위치 Counter 가 COMP- register 의 값 보다 작으면 감속정지합니다. 또한 RR2 register 의 D1(SLMT-) Bit 가 1 이 됩니다.

이 상태에서 -방향의 Drive 명령은 실행되지 않습니다.

**D2 LMTM D** 하드웨어 Limit (nLMTP, nLMTM 입력 신호)가 Active 되었을 때의 Drive 정지 방식을 설정합니다.

0 으로 하면 즉시 정지, 1 로 하면 감속정지합니다.

**D3 HLMT+** +방향 Limit 입력신호(nLMTP)의 논리 Level 을 설정합니다.

0: Low active, 1: High active

**D4 HLMT-** -방향 Limit 입력신호(nLMTM)의 논리 Level 을 설정합니다.

0: Low active, 1: High active

**D5 CMPSL** COMP+/- register 의 비교 대상을 논리위치/실제위치 Counter 로 설정합니다.

0: 논리위치 Counter, 1: 실제위치 Counter

**D6 PLSMD** Drive pulse 의 출력 방식을 설정합니다.

0: 독립 2pulse 방식, 1: 1 pulse 방식

독립 2pulse 방식으로 하면 출력신호 nP+P/nP+N 에 +방향 Pulse 가, 출력신호 nP-P/nP-N 에 -방향 Pulse 가 출력됩니다. 1Pulse 방식으로 하면 출력신호 nP+P/nP+N 에 +/- 양방향의 Drive pulse 가, 출력신호 nP-P/nP-N 에 Pulse 의 방향 신호가 출력됩니다.

[주의]

1 pulse 방식의 경우는 Pulse 신호와 방향 신호가 출력되는 타이밍 '14.2 Drive 시작/종료 시'을 참조하십시오.

**D7 PLS-L** Drive pulse 의 논리 Level 을 설정합니다.

0: 정논리 Pulse, 1: 부논리 Pulse

**D8 DIR-L** Drive pulse 의 방향 출력신호의 논리 Level 을 설정합니다.

이 bit 의 값에 의해 nP-P/nP-N 출력신호의 전압 Level 은 아래 표와 같이 출력됩니다.

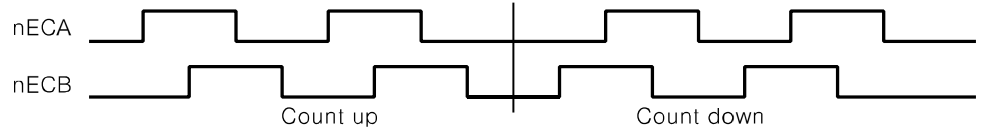
D8 (DIR-L)	+ 방향 Pulse 출력 시	- 방향 Pulse 출력 시
0	Low	High
1	High	Low

D9 PINMD Encoder 입력신호 (nECAP/nECAN, nECBP/nECBN)를 2 상 Pulse 입력 또는 Up/Down pulse 입력으로 선택합니다.

Encoder 입력 신호는 실제위치 Counter 를 Count up/down 합니다.

0: 2 상 Pulse 입력, 1: Up/Down Pulse 입력

이 Bit 를 2 상 Pulse 입력 Mode 로 설정하면, 정논리 Pulse 로 A 상이 진행되고 있을 때는 Count up, B 상이 진행되고 있을 때는 Count down 합니다. 양 신호의 ↑, ↓에서 Count up, Count down 합니다.



이 Bit 를 Up/Down Pulse 입력의 Mode 로 설정하면, nECAP/nECAN 가 Count up 입력에 nECBP/nECBN 가 Count down 입력이 됩니다. 각각 정 Pulse 의 ↑에서 Count 합니다.

D11 PIND1, Encoder 2 상 Pulse 입력의 분주비를 설정합니다.  
, 10 0

D11	D10	2 상 Pulse 입력의 분주비
0	0	1/1
0	1	1/2
1	0	1/4
1	1	무효

D12 ALM-L nALARM 입력 신호의 논리 Level 을 설정합니다.  
0: Low Active, 1: High Active

D13 ALM-E 서보모터 Alarm 용 입력 신호 nALARM 의 유효/무효를 설정합니다.  
0: 무효, 1: 유효

유효로 설정하면 nALARM 입력신호가 Active 상태로 되면 RR2 register 의 D14(ALARM) bit 가 1 이 됩니다. Drive 중에 Active level 이 되면 Drive 는 즉시 정지합니다.

D14 INP-L nINPOS 입력 신호의 논리 Level 을 설정합니다.  
0: Low Active, 1: High Active

D15 INP-E 서보 모터 위치 결정 완료용 입력 신호 nINPOS 의 유효/무효를 설정합니다.  
0: 무효, 1: 유효

유효로 설정하면 Drive 종료 후 nINPOS 신호가 Active 되면 RR0(주 status) register 의 n-DRV bit 가 0 으로 돌아옵니다.

Reset 시에는 D15~D0 는 모두 0 으로 설정됩니다.

## 7.6 WR3 mode register 3

Mode register 3 은 4 축, 각각 개별적으로 가지고 있습니다. 어느 축의 Mode register 를 사용할 지는 축 지정에 의해 정해집니다. Mode register3 은 매뉴얼 감속, 감속도 개별, S 자 가감속 Mode, 외부조작 Mode 의 설정과 범용출력 OUT7~4 를 설정합니다.

WR3	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	0	0	0	0	OUT7	OUT6	OUT5	OUT4	OUTSL	0	0	EXOP1	EXOP0	SACC	DSNDE	MANLD

**D0 MANLD** 가감속 정량 Pulse drive 에 있어서 감속을 자동 감속/매뉴얼 감속으로 설정합니다.

0: 자동 감속, 1: 매뉴얼 감속

매뉴얼 감속 Mode 로 설정할 경우는 매뉴얼 감속점이 설정되어 있어야 합니다.

**D1 DSNDE** 직선 가감속 Drive 감속 시의 감속도를 가속도의 값을 사용할 지, 개별적인 감속도의 값으로 사용할 지를 설정합니다. 또한 S 자 가감속 Drive 의 감속시의 감속도 증가율을 가속도 증가율의 값을 사용할 지, 개별의 감속도 증가율의 값으로 사용할 지를 설정합니다.

**D1(DSNDE)** 직선(사다리꼴) 가감속시의 감속도 S 자 가감속시의 감속도 증가율 가감속 커브의 형상

D1 (DSNDE)	직선(사다리꼴) 가감속시의 감속도	S 자 가감속시의 감속도 증가율	가감속 커브의 형상
0	가속도(A)의 값을 사용	가속도 증가율(K)의 값을 사용	대칭
1	감속도(D)의 값을 사용	감속도 증가율(L)의 값을 사용	비대칭

가속과 감속이 대칭형 가감속 Drive 로 실시할 때는 이 Bit 를 0 으로 비대칭형 가감속 Drive 를 실시할 때에는 1 로 합니다.

비대칭의 S 자 가감속/정량 Pulse drive 에서는 자동 감속할 수 없기 때문에, D0(MANLD)bit 를 1 로하여 매뉴얼 감속점(DP)을 설정해야 합니다.

**D2 SACC** 직선(사다리꼴) 가감속/S 자 가감속을 설정합니다.

0: 직선(사다리꼴) 가감속, 1: S 자 가감속

S 자 가감속의 경우는 가속도 증가율(K), (감속도 증가율(L))이 설정되어 있어야 합니다.

**D4, 3 EXOP1, 0** 외부입력신호(nEXP+, nEXP-)에 의한 Drive 조작을 설정합니다.

D4(EXOP1)	D3(EXOP0)	내용
0	0	외부 입력 신호에 의한 Drive 조작 무효
0	1	연속 Pulse drive mode
1	0	정량 Pulse drive mode
1	1	2 상 Encoder 신호 mode

연속 Pulse drive mode 에서는 nEXP+ 신호의 Low level 기간 동안 +방향의 Drive pulse 를 연속적으로 출력합니다. nEXP- 신호의 경우도 이와 같이 -방향의 Drive pulse 를 연속해서 출력합니다.

정량 Pulse drive mode 에서는 nEXP+ 신호를 High level 에서 Low level 로 변경될 때, ↓에서 +방향의 정량 pulse drive 가 기동합니다. nEXP- 신호의 경우도 -방향의 정량 Pulse drive 를 기동합니다. 2 상 Encoder 신호 Mode 에서는 nEXP- 신호가 Low level 로 nEXP+ 신호의 ↑에서 +방향의 정량 Pulse drive 가 기동합니다. 또한 nEXP- 신호가 Low level 로 nEXP+ 신호의 ↓에서 -방향의 정량 Pulse drive 가 기동합니다.

D7 OUTSL

출력신호 nOUT7~4 를 범용 출력/Drive 상태를 출력할 지를 선택합니다.

0: 범용 출력으로 사용합니다. D11~D8 의 내용이 nOUT7~4 단자로 출력됩니다.

1: nOUT7~4 는 아래 표와 같이 Drive 상태를 출력합니다.

신호명	출력 내용
nOUT4/CMPP	논리/실제위치 Counter 가 COMP+ register 보다 클 때 High level 이, 작을 때 Low level 이 됩니다.
nOUT5/CMPM	논리/실제위치 Counter 가 COMP- register 보다 작을 때 High level 이, 클 때 Low level 이 됩니다.
nOUT6/ASND	Drive 명령 실행 중 가속상태가 되면 High level 이 됩니다.
nOUT7/DSND	Drive 명령 실행 중 감속상태가 되면 High level 이 됩니다.

D11~8 OUTm

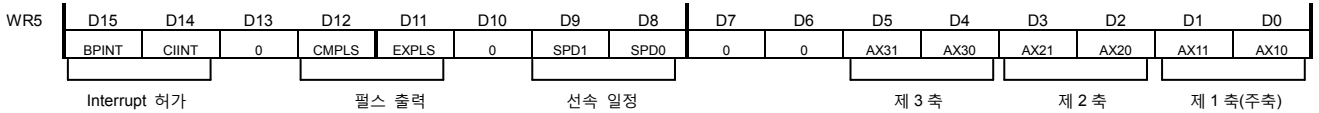
출력 신호 nOUT7~4 를 범용 출력으로 설정합니다.

0: Low level 출력, 1: High level 출력

Reset 시 D15~D0 는 모두 0 으로 설정됩니다. D15~12, 6, 5 bit 는 항상 0 으로 설정합니다.

## 7.7 WR5 보간 Mode register

보간 Drive 를 실행하기 위한 축 지정, 선속일정 Mode, 보간 Step 출력 Mode, 보간 시 Interrupt 설정을 실시합니다.



D1, 0      AX11, 10      보간 Drive 를 실시하는 제 1 축 (주축)을 지정합니다. 축 코드를 아래 표에 나타냅니다.

축	코드(2 진)
X	0 0
Y	0 1
Z	1 0
U	1 1

제 1 축(주축)으로 지정된 축은 보간연산을 기동하는 기본 Pulse 를 발생하기 때문에 정속/가감속 Drive 에 필요한 속도 Parameter 를 설정해야 합니다.

D3, 2      AX21, 20      보간 Drive 를 실시하는 제 2 축을 지정합니다.

D5, 4      AX31, 30      3 축 보간 Drive 를 실시하는 제 3 축을 지정합니다.  
2 축 보간 Drive 시에는 임의의 값을 지정합니다.

D9, 8      LSPD1, 0      보간 Drive 의 선속일정 Mode 를 설정합니다.

D9	D8	동작 Mode
0	0	선속일정 무효
0	1	2 축 선속일정
1	0	(설정 불가)
1	1	3 축 선속일정

2 축 선속일정 Mode 의 경우는 제 2 축의 Range(R)를 주축 R 의 1.414 배의 값으로 설정합니다. 3 축 선속일정 Mode 의 경우는 제 2 축의 Range(R)를 주축 R 의 1.414 배의 값으로 제 3 축 Range(R)를 주축 R 의 1.732 배의 값으로 설정합니다.

D11      EXPLS      0 으로 설정합니다. (사용 안함)

D12      CMPLS      1 로 하면 보간 Drive 를 Command 로 Step 출력하는 Mode 가 됩니다.

D14      CIINT      연속 보간 시 Interrupt 발생의 허가/금지를 설정합니다.  
0: 금지, 1: 허가

D15      BPINT      Bit pattern 보간 시 Interrupt 발생의 허가/금지를 설정합니다.  
0: 금지, 1: 허가

Reset 시 D15~D0 은 모두 0 으로 설정합니다.

## 7.8 WR6, 7 Write data register 1, 2

Data 기록 명령을 설정하는 Register 입니다. WR6 register 에는 Write data 하위 16 bit(WD15~WD0), WR7 register 에는 Write data 상위 16 bit(WD31~WD16)를 설정합니다.

WR6	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	WD15	WD14	WD13	WD12	WD11	WD10	WD9	WD8	WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0

WR7	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	WD31	WD30	WD29	WD28	WD27	WD26	WD25	WD24	WD23	WD22	WD21	WD20	WD19	WD18	WD17	WD16

Data 기록 명령은 우선 각각의 명령으로 지정되어 있는 Data 길이를 이러한 Write data register 에 기입합니다. Write data register WR6, 7(8bit data bus 의 경우는 WR6L, WR6H, WR7L, WR7H)을 Write 한 후 Command register 에 명령코드를 기입하면 Write data register 의 내용이 각각의 내부 Register 에 들어갑니다. 기입해지는 수치 Data 는 모두 바이너리(Binary, 2 진수)입니다. 또한 부의 값은 2 의 보수를 취급합니다.

각 명령의 Data 는 반드시 지정되는 Data 길이로 설정해야 합니다.

Reset 시에는 WR6, WR7 register 의 내용은 부정입니다.

## 7.9 RR0 status register

각 축의 Drive, 에러 상태를 표시합니다. 또한 보간 Drive, 연속 보간시 다음 Data 가능, 원호보간의 상한, BP 보간의 Stack counter 를 표시합니다.

R0	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	-	BPSC1	BPSC0	ZONE2	ZONE1	ZONE0	CNEXT	I-DRV	U-ERR	Z-ERR	Y-ERR	X-ERR	U-DRV	Z-DRV	Y-DRV	X-DRV

		각 축의 에러 표시				각 축의 드라이브 표시			
D3~0	n-DRV	<p>각 축의 Drive 상태를 나타냅니다. 이 Bit 가 1 이 되면 그 축이 Drive pulse 를 출력 중인 것을 나타내고 있습니다. 0 이면 그 축이 Drive 를 종료하고 있는 것을 나타냅니다.</p> <p>자동 원점 복귀 실행시에는 실행하고 있는 동안 이 bit 가 1 이 됩니다. 서보모터 위치 결정 완료용 입력신호의 nINPOS 를 유효로 설정하면 Drive pulse 를 출력 후 nINPOS 신호가 Active 되고 나서 0 으로 돌아옵니다.</p>							
D7~4	n-ERR	<p>각 축의 에러 발생 상태를 표시합니다. 각 축의 RR2 register 의 에러 Bit (D7~D0) 및 RR1 register 의 에러 종료 bit (D15~D12) 중 하나라도 1 이 되면 이 bit 가 1 이 됩니다.</p>							
D8	I-DRV	<p>보간 Drive 상태를 나타냅니다. 이 bit 가 1 이 되면, 보간 Drive pulse 를 출력중인 것을 나타내고 있습니다.</p>							
D9	CNEXT	<p>연속보간시 다음 Data 기입 가능을 나타냅니다. 연속보간 Drive 로 이 bit 가 1 이 되면 다음 세그먼트(Segment)를 위한 Parameter data 및 보간 명령을 기입하는 것이 가능하게 됩니다.</p>							
D3~0	n-DRV	<p>각 축의 Drive 상태를 나타냅니다. 이 bit 가 1 이 되면 그 축이 Drive pulse 를 출력중인 것을 나타내고 있습니다. 0 이면 그 축이 Drive 를 종료하고 있는 것을 나타냅니다.</p> <p>자동 원점 복귀 실행시에는 실행하고 있는 동안 이 bit 가 1 이 됩니다. 서보모터 위치 결정 완료용 입력 신호의 nINPOS 를 유효로 설정하면 Drive pulse 를 출력 후 nINPOS 신호가 Active 되고 나서 0 으로 돌아옵니다.</p>							
D7~4	n-ERR	<p>각 축의 에러 발생 상태를 표시합니다. 각 축의 RR2 register 의 에러 bit(D7~D0) 및 RR1 register 의 에러 종료 bit (D15~D12) 중 하나라도 1 이 되면 이 bit 가 1 이 됩니다.</p>							
D8	I-DRV	<p>보간 Drive 상태를 나타냅니다. 이 bit 가 1 이 되면 보간 Drive pulse 를 출력중인 것을 나타내고 있습니다.</p>							
D9	CNEXT	<p>연속보간시 다음 Data 기입 가능을 나타냅니다. 연속보간 Drive 로 이 Bit 가 1 이 되면 다음 세그먼트(Segment)를 위한 Parameter data 및 보간 명령을 기입하는 것이 가능하게 됩니다.</p>							

D1~D10 ZONEm 원호 보간 Drive 에 대해서 현재 Drive 중의 상한을 나타냅니다.

D12	D11	D10	현재 Drive 중의 상한
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

D14, 13 BPSC1, 0 Bit pattern 보간 Drive 에 대해서, Stack Counter(SC)의 값을 나타냅니다.

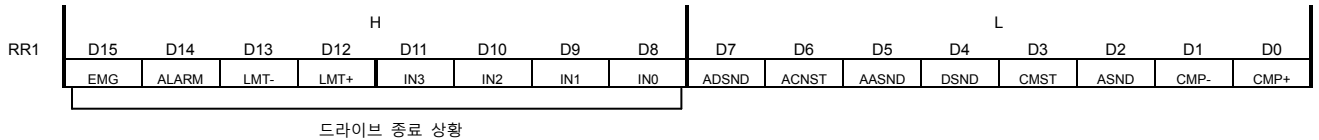
D14	D13	Stack Counter(SC)의 값
0	0	0
0	1	1
1	0	2
1	1	3

Bit pattern 보간의 Drive 중, SC=3 일 때는 Bit data stack 이 가득 차 있다는 것을 나타내고 있습니다. SC=2 일 때는 각 축당 16bit 가 비어 있으며, SC=1 일 때는, 각 축당 16bit×2 가 비어 있습니다. SC=0 는 Bit data 를 모두 출력하여 Drive 가 종료한 것을 나타냅니다.

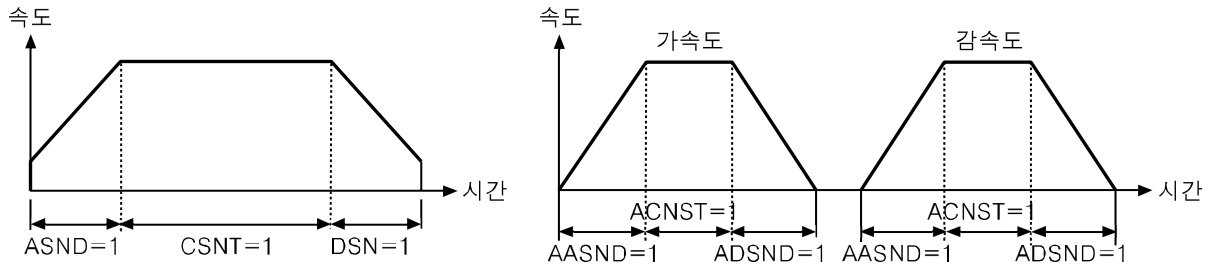


### 7.10 RR1 status register 1

Status register1 은 4 축, 각각 개별적으로 가지고 있습니다. 어느 축의 Status register 를 읽을 것인가는 축 지정에 의해 정해집니다. Status register1 은 논리/실제 위치 Counter 와 COMP± register 의 대소비교, 가감속 Drive 의 가속상태, S 자 가감의 가속 상태를 표시합니다. 또한 Drive 종료 Status 를 표시합니다.



- D0      CMP+      논리/실제위치 Counter 와 COMP+ register 의 대소관계를 나타냅니다.  
1: 논리/실제위치 Counter ≥ COMP+ register  
0: 논리/실제위치 Counter < COMP+ register
- D1      CMP-      논리/실제위치 Counter 와 COMP- register 의 대소관계를 나타냅니다.  
1: 논리/실제위치 Counter < COMP- register  
0: 논리/실제위치 Counter ≥ COMP- register



- D2      ASND      가감속 Drive 시, 가속 때 1 이 됩니다.
- D3      CNST      가감속 Drive 시, 정속 때 1 이 됩니다.
- D4      DSND      가감속 Drive 시, 감속 때 1 이 됩니다.
- D5      AASND      S 자 가감속 Drive 로 가속도/감속도가 증가할 때 1 이 됩니다.
- D6      ACNST      S 자 가감속 Drive 로 가속도/감속도가 일정할 때 1 이 됩니다.
- D7      ADSND      S 자 가감속 Drive 로 가속도/감속도가 감소할 때 1 이 됩니다.
- D11~8    IN3~0      Drive 가 외부 감속 정지 신호(nIN3~0)에 의해 정지했을 때 1 이 됩니다.
- D12      LMT+      Drive 가 +방향 Limit 신호 (nLMT+)에 의해 정지했을 때 1 이 됩니다.
- D13      LMT-      Drive 가 -방향 Limit 신호 (nLMT-)에 의해 정지했을 때 1 이 됩니다.
- D14      ALARM      Drive 가 서보 모터용 Alarm 신호(nALARM)에 의해 정지했을 때 1 이 됩니다.
- D15      EMG      Drive 가 긴급정지 신호 (EMG)에 의해 정지했을 때 1 이 됩니다.

**(3) Drive 종료 Status bit 에 대해**

Drive 종료 Status bit 는 Drive 를 종료시킨 요인의 정보를 보관, 유지하는 bit 입니다.

정량 Pulse drive, 연속 Pulse drive 는 다음과 같은 요인에 의해 종료합니다.

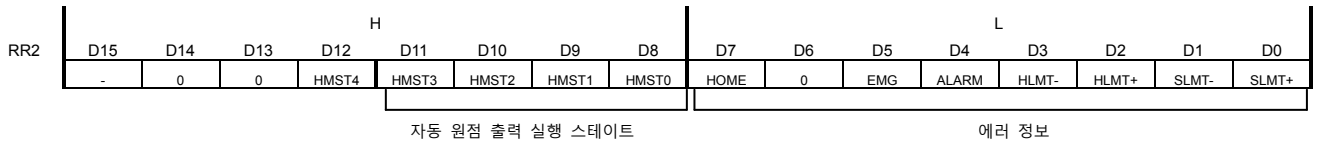
- ① 정량 Pulse drive 로 출력 Pulse 를 모두 출력했을 때
- ② 감속정지 또는 즉시 정지 명령이 기입되었을 때
- ③ 소프트웨어 Limit 가 유효로 설정되어 Active 되었을 때
- ④ 정량/연속 Pulse drive 로 감속정지시키는 외부신호(nIN3, 2, 1, 0)가 Active 되었을 때
- ⑤ Limit 입력 신호(nLIMIT+P, nLMT-)가 Active 되었을 때
- ⑥ nALARM 신호가 유효로 설정되어 Active 되었을 때
- ⑦ EMG 신호가 Low level 이 되었을 때

여기서 ①, ②의 요인에 대해서는 상위 CPU 에 의해 제어 가능한 요소이며, ③의 요인은 Drive 종료 후에도 RR2 register 로 확인할 수가 있습니다. 그러나 ④~⑦ 요인에 대해서 Drive 를 종료시킨 요인의 bit 가 1 이 된 후, 신호가 Non active 되어도 bit 정보를 유지합니다.

Drive 종료 status bit 가운데 에러 요인이 되는 D15~D12 의 bit 가 1 이 되면 RR0 주 status register 의 n-ERR bit 가 1 이 됩니다. Drive 종료 status bit 는 다음 Drive 의 명령의 기입으로 자동적으로 Clear 되지만, 종료 status clear 명령(25h)에 의해서도 Clear 할 수가 있습니다.

## 7.11 RR2 status register 2

Status register 2 는 4 축 각각 개별적으로 가지고 있습니다. 어느 축의 Status register 를 읽을 것인가는 축 지정에 의해 정해집니다. Status register 2 는 에러정보 및 자동 원점 복귀 실행시의 실행 status 를 표시하는 Register 입니다. 에러정보(D7~D0)의 각 bit 가 1 이 되면 그 bit 에 해당하는 에러가 발생한 것을 나타냅니다. 이 RR2 register 의 D7~D0 의 몇 개의 bit 가 1 이 되면 RR0 주 status register 의 n-ERR bit 가 1 이 됩니다.



D0	SLMT+	COMP+ register 를 소프트웨어 Limit 로서 유효하게 하고 +방향 Drive 시에 논리/실제위치 Count 가 COMP+ register 의 값 보다 클 때
D1	SLMT-	COMP- register 를 소프트웨어 Limit 로서 유효하게 하고 -방향 Drive 시에 논리/실제위치 Count 가 COMP- register 의 값 보다 클 때
D2	HLMT+	+방향 Limit 신호 (nLIMT+)가 Active level 일 때
D3	HLMT-	-방향 Limit 신호 (nLIMT-)가 Active level 일 때
D4	ALARM	서보모터용 Alarm 신호(nALARM)가 유효로 설정되고 Active level 일 때
D5	EMG	긴급정지 신호(EMG)가 Low level 일 때
D7	HOME	자동 원점 복귀 실행시의 에러입니다. Step3 시작시, Encoder Z 상 신호(nIN2)가 Active 되어 있으면 1 이 됩니다.
D12~8	HMST4~0	자동 원점 복귀 실행 Status 는 자동 원점 복귀 실행 중에 현재 실행하고 있는 동작 내용을 나타냅니다. '2.5.4 자동 원점 출력시의 에러'를 참조하십시오.

Drive 중에 진행 방향의 Hard/Soft limit 가 작동하면, Drive 는 감속정지 또는 즉시 정지합니다.

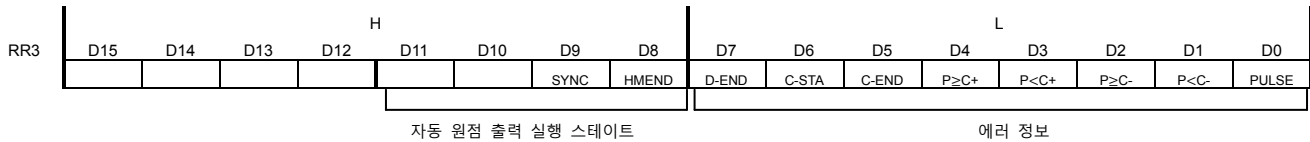
정지 후 정지 전과 동일한 방향으로의 Drive 명령은 실행되지 않습니다.

SLMT+/- bit 는 역방향 Drive 시 조건이 되어도 1 이 되지 않습니다.

## 7.12 RR3 status register 3

Status register3 은 4 축 각각 개별적으로 가지고 있습니다. 어느 축의 Status register 를 읽을 것인가는 축 지정에 의해 정해집니다. Status register3 은 Interrupt 발생 요인을 나타내는 register 입니다. Interrupt 가 발행하면 그 Interrupt 발생요인의 bit 가 1 이 됩니다.

D0 에서 D7 의 Interrupt 를 발생시키려면 WR1 register 의 각 발생 요인을 Interrupt 허가로 설정해야 합니다.



D0	PULSE	Drive pulse 가 발생 (Drive pulse 정논리 설정시)
D1	P≥C-	논리/실제위치 Counter 가 COMP- register 보다 크다.
D2	P<C-	논리/실제위치 Counter 가 COMP- register 보다 작다.
D3	P<C+	논리/실제위치 Counter 가 COMP+ register 보다 작다.
D4	P≥C+	논리/실제위치 Counter 가 COMP+ register 보다 크다.
D5	C-END	가감속 Drive 시, 정속영역 Pulse 출력을 종료
D6	C-STA	가감속 Drive 시, 정속영역 Pulse 출력을 시작
D7	D-END	Drive 가 종료
D8	HMEND	자동 원점 복귀가 종료 '2.5 자동 원점 복귀 출력'을 참조하십시오.
D9	SYNC	동기 동작이 기동 '2.6 동기 동작'을 참조하십시오.

어떤 Interrupt 요인의 Interrupt 가 발생하면 이 Register 의 bit 가 1 이 되어 Interrupt 출력 신호가 Low level 이 됩니다. CPU 가 Interrupt 를 발생시킨 축의 RR3 register 를 읽으면 RR3 register 의 bit 는 0 으로 복귀되어 Interrupt 출력신호는 Non-active level 로 돌아옵니다.



### Note

bit data bus 의 경우는 RR3L register 의 읽기로 모두 Clear 되기 때문에 D8(HMEND), D9(SYNC) bit 를 사용하는 경우에는 반드시 RR3H 를 먼저 읽어내고 나서 RR3L register 를 읽어냅니다.

## 7.13 RR4, 5 Input register 1, 2

Input register1, 2 는 각 축의 입력 신호 상태를 직접 표시합니다. 입력신호가 Low level 일 때는 0, High level 일 때는 1 을 나타냅니다. 이러한 입력 신호의 상태를 나타내지 않을 때에는 범용 입력 신호로 사용할 수 있습니다.

RR4	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Y-ALM	Y-INP	Y-EX-	Y-EX+	Y-IN3	Y-IN2	Y-IN1	Y-IN0	X-ALM	X-INP	X-EX-	X-EX+	X-IN3	X-IN2	X-IN1	X-IN0

RR5	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	U-ALM	U-INP	U-EX-	U-EX+	U-IN3	U-IN2	U-IN1	U-IN0	Z-ALM	Z-INP	Z-EX-	Z-EX+	Z-IN3	Z-IN2	Z-IN1	Z-IN0

Bit 명	입력 신호	Bit 명	입력 신호
n-IN0	n-IN0+ / n-IN-	n-EX+	nEXP+
n-IN1	n-IN1	n-EX-	nEXP-
n-IN2	n-IN2	n-INP	nINPOS
n-IN3	n-IN3	n-ALM	nALARM

## 7.14 RR6, 7 read data register 1, 2

Data 읽기 명령에 의해 내부 Register 의 Data 가 read data register 에 설정됩니다. RR6 register 에는 Read data 하위 16bit (RD15~RD0)가, RR7 register 에는 Read data 상위 16bit (RD31~RD16)가 설정됩니다.

RR4	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	Y-ALM	Y-INP	Y-EX-	Y-EX+	Y-IN3	Y-IN2	Y-IN1	Y-IN0	X-ALM	X-INP	X-EX-	X-EX+	X-IN3	X-IN2	X-IN1	X-IN0

RR5	H								L							
	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	U-ALM	U-INP	U-EX-	U-EX+	U-IN3	U-IN2	U-IN1	U-IN0	Z-ALM	Z-INP	Z-EX-	Z-EX+	Z-IN3	Z-IN2	Z-IN1	Z-IN0

Data 는 모두 2 진수(Binary)입니다. 또한 부의 값은 2 의 보수로 취급합니다.



# 8 명령 일람

## (1) Data 기입 명령

코드	명령	기호	Data 범위	Data 길이
00h	Range 설정	R	8,000,000(배율:1)~16,000(배율:500)	4 byte
01	가속도 증가율 설정	K	1~65,535	2
02	가속도 설정	A	1~8,000	2
03	감속도 설정	D	1~8,000	2
04	기동속도 설정	SV	1~8,000	2
05	Drive 속도 설정	V	1~8,000	2
06	출력 Pulse 수/보간 중점 설정	P	출력 Pulse 수: 0~4,294,967,295 보간 중점: -2,147,483,646~ +2,147,483,646	4
07	매뉴얼 감속점 설정	DP	0~4,294,967,295	4
08	원호 중심점 설정	C	-2,147,483,646~ +2,147,483,646	4
09	논리위치 Counter 설정	LP	-2,147,483,648~ +2,147,483,647	4
0A	실제위치 Counter 설정	EP	-2,147,483,648~ +2,147,483,647	4
0B	COMP+ register 설정	CP	-2,147,483,648~ +2,147,483,647	4
0C	COMP- register 설정	CM	-2,147,483,648~ +2,147,483,647	4
0D	가속 Counter offset 설정	AO	-32,768~+32,767	2
0E	감속도 증가율 설정	L	1~65,535	2
60	확장 Mode 설정	EM	(Bit data)	4
61	원점 검출 속도 설정	HV	1~8,000	2
64	동기동작 Mode 설정	SM	(Bit data)	4



### Note

Data 를 설정할 때는 반드시 지정된 Data 길이로 설정합니다.

[Parameter 계산식]

$$\text{배율} = \frac{8,000,000}{R}$$

$$\text{가감속 증가율 (pps/sec}^2\text{)} = \frac{6.25 \times 10^6}{K} \times \frac{8,000,000}{\text{배율}}$$

$$\text{감속도 증가율 (pps/sec}^2\text{)} = \frac{6.25 \times 10^6}{L} \times \frac{8,000,000}{\text{배율}}$$

$$\text{가감속 (pps/sec)} = A \times 125 \times \frac{8,000,000}{\text{배율}}$$

$$\text{감속도 (pps/sec)} = D \times 125 \times \frac{8,000,000}{\text{배율}}$$

$$\text{Drive 속도 (pps)} = V \times \frac{8,000,000}{\text{배율}}$$

$$\text{기동속도 (pps)} = SV \times \frac{8,000,000}{\text{배율}}$$

**(2) Data 읽기 명령**

코드	명령	기호	Data 범위	Data 길이
10h	논리위치 Counter 읽기	LP	-2,147,483,648~ +2,147,483,647	4 byte
11	실제위치 Counter 읽기	EP	2,147,483,648~ +2,147,483,647	4
12	현재 Drive 속도 읽기	CV	1~8,000	2
13	현재 가/감속도 읽기	CA	1~8,000	2
14	동기 버퍼 Register 읽기	SB	-2,147,483,648~ +2,147,483,647	4

**(3) Drive 명령**

코드	명령
20h	+방향 정량 Pulse drive
21	-방향 정량 Pulse drive
22	+방향 연속 Pulse drive
23	-방향 연속 Pulse drive
24	Drive 시작 Hold
25	Drive 시작 Free/종료 Status clear
26	Drive 감속 정지
27	Drive 즉시 정지

**(4) 보간 명령**

코드	명령
30h	2 축 직선보간 Drive
31	3 축 직선보간 Drive
32	CW 원호보간 Drive
33	CCW 원호보간 Drive
34	2 축 Bit pattern 보간 Drive
35	3 축 Bit pattern 보간 Drive
36	BP register 기입 가능 <sup>※1</sup>
37	BP register 기입 불가
38	BP data stack
39	BP data clear
3A	보간 Single step
3B	감속 유효
3C	감속 무효
3D	보간 Interrupt clear

※1. 이외의 명령 코드를 Command regist 에 기입하면 모션 컨트롤 IC 내부 회로의 테스트 명령이 기동하여 생각하지 않는 동작을 하는 경우가 있습니다.



## 9 Data 기록(Write) 명령

Data 기록 명령은 기록 Data 를 수반하는 명령입니다. Drive 를 위한 가속도, Drive 속도, 출력 Pulse 수 등의 동작 Parameter 를 설정합니다. 복수의 축을 지정하면 해당 Data 를 지정된 모든 축에 설정할 수 있습니다.

Data 기록 명령은 지정된 Data 길이가 2 바이트이면 WR6 register 에 Data 길이가 4 바이트이면 WR6, 7 register 에 설정됩니다. 그리고 WR0 register 에 축을 지정하고 명령 코드를 기입하면 실행됩니다. WR6, 7 write data register 에 설정하는 수치 Data 는 모두 2 진수(Binary)입니다. 또한 부의 값은 2의 보수로 취급합니다. 각각의 Data 는 반드시 Data 범위 내의 값을 설정해야 합니다. 범위외의 값을 설정하면 올바른 Drive 동작이 실행되지 않습니다.



### Note

- ① Data 기입 명령의 명령 처리에 필요한 시간은 최대로 250nsec (CLK=16MHz의 경우)입니다. 명령을 기입 후 명령을 처리하는 시간에는 다음 data 명령을 기입하지 마십시오.
- ② 가속 Counter offset(AO)를 제외한 모든 동작 Parameter 는 Reset 시 후에는 적용되지 않습니다. Drive 에 필요한 Parameter 에 대해서는 Drive 전에 반드시 적절한 값을 설정합니다.

### 9.1 Range 설정

명령 코드	명령	기호	Data 범위	Data 길이
00h	Range 설정	R	8,000,000(배율:1)~16,000(배율:500)	4 byte

Range 는 기동속도, Drive 속도, 가속도, 감속도, 가속도 증가율, 감속도 증가율의 배율을 결정하는 Parameter 입니다. Range 설정치를 R 배로 하면, 배율은 다음 식과 같습니다.

$$\text{배율} = \frac{8,000,000}{R}$$

Drive 속도, 기동속도, 가감속도 등의 Parameter 는 설정 범위가 1~8000 이므로 이 보다 크게 설정하는 경우에는 배율을 올리지 않으면 안됩니다. 배율을 크게 하면 고속으로 Drive 할 수가 있습니다만 속도 분해능은 영성해 집니다.

사용하시는 속도 범위를 커버할 수 있는 최소의 값으로 설정합니다.



### Ex.

40kpps 속도까지 사용한다면 속도 설정 범위가 1~8,000, 배율은 5 배 이므로 R 을 1,600,000 으로 설정합니다.

Range(R)를 Drive 중에 변경하면 속도가 불연속적으로 변화합니다.

## 9.2 가속도 증가율 설정

명령 코드	명령	기호	Data 범위	Data 길이
01h	가속도 증가율 설정	K	1~65,535	2 byte

가속도 증가율 설정값은 S 자 가감속에 있어서 가속도의 단위 시간 당 증가/감소율을 결정하는 Parameter 입니다. 가속과 감속이 대칭인 대칭형 S 자 가감속 Drive(WR3/D1=0)에서는, 감속시에도 이 가속도 증가율의 값이 사용됩니다. 가속도 증가율의 설정치를 K로 하면, 가속도 증가율은 다음 식과 같이 됩니다.

$$\text{가감속 증가율 (pps/sec}^2\text{)} = \frac{6.25 \times 10^6}{K} \times \frac{8,000,000}{R}$$

R  
 ↓  
 배율

가속도 증가율 설정치(K)의 설정 범위가 1~65,535 이기 때문에 가속도 증가율 범위는 다음과 같이 됩니다.

배율=1 일 때, 954pps/sec<sup>2</sup> ~ 62.5×10<sup>6</sup> pps/sec<sup>2</sup>

배율=500 일 때, 477×10<sup>3</sup> pps/sec<sup>2</sup>~31.25×10<sup>9</sup> pps/sec<sup>2</sup>

## 9.3 가속도 설정

명령 코드	명령	기호	Data 범위	Data 길이
02h	가속도 설정	A	1~8,000	2 byte

직선 가감속 Drive(사다리꼴)의 가속 시, 가속도를 결정하는 Parameter 입니다. 가속과 감속이 대칭형 직선 가감속 Drive(WR3/D1=0)에서는 감속시에도 이 가속도의 값이 사용됩니다. S 자 가감속 Drive에서는 이 Parameter 는 항상 최대값 8,000 을 설정합니다. 가속도 설정치를 A로 하면 가속도는 다음 식과 같이 됩니다.

$$\text{가감속(pps/sec)} = A \times 125 \times \frac{8,000,000}{R}$$

R  
 ↓  
 배율

가속도 설정치(A)의 설정 범위가 1~8,000 이기 때문에 실제의 가속도 범위는 다음과 같이 됩니다.

배율=1 일 때, 125pps/sec<sup>2</sup> ~ 1×10<sup>6</sup> pps/sec<sup>2</sup>

배율=500 일 때, 62.5×10<sup>3</sup> pps/sec<sup>2</sup>~500×10<sup>6</sup> pps/sec<sup>2</sup>

## 9.4 감속도 설정

명령 코드	명령	기호	Data 범위	Data 길이
03h	감속도 설정	D	1~8,000	2 byte

비대칭형 직선 가감속 Drive(WR3/D1=1)에서, 감속시의 감속도가 되는 Parameter 입니다.

비대칭형 S자 가감속 Drive 에서 이 Parameter 는 항상 최대치 8,000 으로 설정합니다.

감속도 설정치를 D 로 하면 감속도는 다음 식과 같이 됩니다.

$$\text{감속도(pps/sec)} = D \times 125 \times \frac{8,000,000}{R}$$

↓  
배율

## 9.5 기동 속도 설정

명령 코드	명령	기호	Data 범위	Data 길이
04h	기동속도 설정	SV	1~8,000	2 byte

가감속 Drive 의 가속 시작시의 속도와 감속 종료시의 속도입니다.

기동속도 설정치를 SV 로 하면 기동속도는 다음 식과 같습니다.

$$\text{기동속도(pps)} = SV \times \frac{8,000,000}{R}$$

↓  
배율

Step 모터의 경우는 자기동 주파수내의 값으로 설정합니다. 서보모터의 경우도 너무 낮게 값을 설정하면 정량 Pulse drive 의 감속 종료시에 기동 속도로 Drive 가 출력이 유지될 수 있습니다. 이러한 경우는 다음과 같이 실시합니다.

- 가속/감속 대칭형 직선 가감속 drive 의 경우
  - 가속 Counter offset(A0) 을 0 으로 설정
  - 삼각 방지 기능 유효(확장 명령 60h WR6/D3 (AVTRI)=1)
- 가속/감속 비대칭형 직선 가감속 drive 의 경우
  - 가속 Counter offset(A0) 을 0 으로 설정
  - 삼각 방지 기능 유효(확장 명령 60h WR6/D3 (AVTRI)=1)

그러나 가속도>감속도의 경우, 가속도 A 와 감속도 D 의 비율이 커지면 정속영역이 없어집니다.

이러한 경우에는 기동 속도를 올립니다.

## 9.6 Drive 속도 설정

명령 코드	명령	기호	Data 범위	Data 길이
05h	Drive 속도 설정	V	1~8,000	2 byte

가감속 Drive 에서 정속 영역에 이르렀을 때의 속도입니다. 정속 Drive 에서는 처음부터 이 속도가 됩니다.

Drive 속도 설정치를 V 로 하면 Drive 속도는 다음 식과 같이 됩니다.

$$\text{Drive 속도(pps)} = V \times \frac{8,000,000}{R}$$

↓  
배율

이 Drive 속도를 기동속도 이하로 설정하면 가감속 Drive 는 실행하지 않고 처음부터 정속 Drive 가 됩니다. Encoder 의 Z 상 서치를 저속으로 Drive 하여 즉시정지하고 싶을 때는 Drive 속도를 기동속도 이하로 설정합니다. Drive 속도는 Drive 도중에도 자유롭게 변경할 수가 있습니다. 가감속 Drive 의 정속 영역의 Drive 속도를 재설정하면, 가속 또는 감속을 시작하여 재설정된 속도에 이르면 정속 Drive 합니다. 자동 원점 복귀에서의 Drive 속도는 Step1, Step4 는 고속 검출 속도로 고속이동합니다.



### Note

- S 자 가감속 정량 Pulse drive 는 Drive 도중에 Drive 속도 변경을 할 수 없습니다. 또한 S 자 가감속 연속 Pulse drive 에서도 가속중이나 감속중에 속도를 변경하면 올바른 S 자 커브를 그릴 수가 없습니다. 정속영역에서 변경하십시오.
- 직선 가감속 정량 Pulse drive 에서 Drive 도중에 빈번하게 Drive 속도를 변경하면, 출력 Pulse 종료 시의 감속 영역에서 기동속도로 Drive 시킬 수 있습니다.

## 9.7 출력 Pulse 수/보간 종점 설정

명령 코드	명령	기호	Data 범위	Data 길이
06h	출력 Pulse 수/보간 종점 설정	P	출력 Pulse 수: 0~4,294,967,295 보간 종점: -2,147,483,646 ~ +2,147,483,646	4 byte

출력 Pulse 수는 Pulse drive 의 총 출력 Pulse 수 입니다. 부호없이 32bit 로 설정합니다.

직선보간, 원호보간 Drive 는 각 축의 종점을 설정합니다. 종점 좌표는 32bit 로 현재 위치에 대한 상대값을 부호 첨부하여 설정합니다. 출력 Pulse 수는 Drive 도중에 변경할 수가 있습니다.

## 9.8 매뉴얼(Manual) 감속점 설정

명령 코드	명령	기호	Data 범위	Data 길이
07h	매뉴얼 감속점 설정	DP	0~4,294,967,295	4 byte

매뉴얼 감속 Mode 의 가감속 정량 Pulse drive 의 감속점을 설정합니다.

매뉴얼 감속 Mode 는 WR3 register 의 D0 bit 를 1 로 하고 감속점은 다음과 같이 설정합니다.

매뉴얼 감속점 = 출력 Pulse 수 - 감속시에 소비하는 Pulse 수

## 9.9 원호 중심점 설정

명령 코드	명령	기호	Data 범위	Data 길이
08h	원호 중심점 설정	C	-2,147,483,646~ +2,147,483,646	4 byte

원호보간 Drive 의 중심점을 설정합니다. 중심 좌표는 현재 위치에 대한 상대값의 부호 첨부하여 설정합니다.

## 9.10 논리위치 Counter 설정

명령 코드	명령	기호	Data 범위	Data 길이
09h	논리위치 Counter 설정	LP	-2,147,483,648~ +2,147,483,647	4 byte

논리위치 Counter 의 값을 설정합니다. 논리위치 Counter 는 +방향/-방향의 Drive 출력 Pulse 를 Up/Down count 합니다. 논리위치 Counter 의 값은 항상 기록 가능하며 Data 읽기 명령으로 항상 읽어낼 수도 있습니다.

## 9.11 실제위치 Counter 설정

명령 코드	명령	기호	Data 범위	Data 길이
0Ah	실제위치 Counter 설정	EP	-2,147,483,648~ +2,147,483,647	4 byte

실제위치 Counter 의 값을 설정합니다. 실제위치 Counter 는 Encoder 입력 Pulse 를 Up/Down count 합니다. 실제위치 Counter 의 값은 항상 기록 가능하며 Data 읽기 명령으로 항상 읽어낼 수도 있습니다.

## 9.12 COMP+ register 설정

명령 코드	명령	기호	Data 범위	Data 길이
0Bh	COMP+ register 설정	CP	-2,147,483,648~ +2,147,483,647	4 byte

COMP+ register 의 값을 설정합니다. COMP+ register 는 논리/실제위치 Counter 와 대소 비교하는 Register 로 비교 결과는 RR1 register 의 D0 와 nOUT4/CMPP 신호로 출력됩니다. +방향의 소프트웨어 Limit 로도 사용합니다. COMP+ register 의 값은 항상 기록 가능합니다.

### 9.13 COMP- register 설정

명령 코드	명령	기호	Data 범위	Data 길이
0Ch	COMP- register 설정	CM	-2,147,483,648~ +2,147,483,647	4 byte

COMP- register 의 값을 설정합니다. COMP- register 는 논리/실제위치 Counter 와 대소 비교하는 Register 로 비교 결과는 RR1 register 의 D1 와 nOUT5/CMPM 신호로 출력됩니다. -방향의 소프트웨어 Limit 로도 사용합니다. COMP- register 의 값은 항상 기록 가능합니다.

### 9.14 가속 Counter offset 설정

명령 코드	명령	기호	Data 범위	Data 길이
0Dh	가속 Counter offset 설정	AO	-32,768~+32,767	2 byte


가속 Counter 의 Offset 값을 설정합니다. 가속 Counter 의 Offset 값은 Reset 시에 8 이 Set 됩니다. 기동 속도를 낮게 설정해 가감속의 정량 Pulse drive 를 실시하는 경우 이 Parameter 값은 0 으로 설정해 주십시오.

### 9.15 감속도 증가율 설정

명령 코드	명령	기호	Data 범위	Data 길이
0Eh	감속도 증가율 설정	L	1~65,535	2 byte

감속도 증가율 설정치는 가속과 감속이 비대칭형 S 자 가감속 Drive(WR3/D1=1)에 있어서 감속도의 단위 시간당의 증가/감소율을 결정하는 Parameter 입니다. 가속과 감속이 대칭인 대칭형 S 자 가감속 Drive 에서는 사용되지 않습니다. 감속도 증가율의 설정치를 L 로 하면 감속도 증가율은 아래의 식과 같습니다.

$$\text{감속도 증가율 (pps/sec}^2\text{)} = \frac{6.25 \times 10^6}{L} \times \frac{8,000,000}{R}$$



감속도 증가율 설정치(L)의 설정 범위가 1~65,535 이기 때문에 감속도 증가율 범위는 다음과 같이 됩니다.

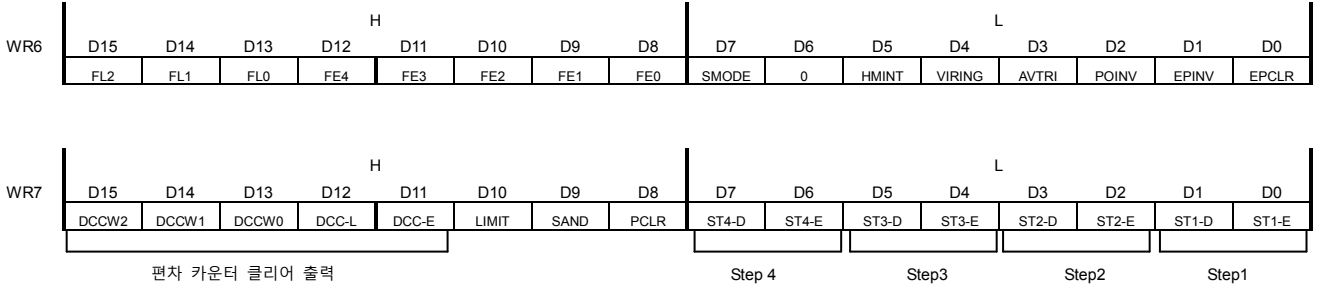
배율=1 일 때, 954pps/sec<sup>2</sup> ~ 62.5×10<sup>6</sup> pps/sec<sup>2</sup>

배율=500 일 때, 477×10<sup>3</sup> pps/sec<sup>2</sup>~31.25×10<sup>9</sup> pps/sec<sup>2</sup>

### 9.16 확장 Mode 설정

명령 코드	명령	기호	Data 범위	Data 길이
60h	확장 Mode 설정	EM	(Bit data)	4 byte

확장 Mode 의 설정은 아래와 같이 WR6 과 WR7 register 의 각 Bit 에 적정치를 설정한 후에 WR0 register 에 축 지정과 함께 명령 코드(60h)를 기입하면 WR6, 7 register 의 내용이 확장 Mode register(EM6, 7)에 설정되고, Reset 시에는 확장 Mode register(EM6, 7)의 모든 Bit 는 Clear 됩니다.



**WR6/D0 EPCLR** nIN2 신호에 의해 Drive 가 정지했을 때 실제 위치 Counter 를 Clear 합니다. 이 bit 를 1 로 하고 Drive 중에 nIN2 신호가 Active level 로 변화했을 때, Drive 가 정지됨과 함께 실제위치 Counter(EP)가 Clear 됩니다. WR1/D5 (IN2-E) bit 는 1 로, WR1/D4(IN2-L) bit 는 유효 Level 로 설정해야 합니다. '4.4 WR1 mode register 1'을 참조하십시오.

**WR6/D1 EPINV** 실제위치 Coujnter 의 증감을 반전시킵니다.

WR6/D1 (EPINV)	입력펄스 Mode	실제 위치의 Counter (EP) 증감
0	A/B 상 Mode	A 상 진행 시, Count up B 상 진행 시, Count down
	Up/Down 펄스 Mode	nECA+/- Pulse 입력 시, Count up nECB+/- Pulse 입력 시, Count down
1	A/B 상 Mode	B 상 진행 시, Count up A 상 진행 시, Count down
	Up/Down 펄스 Mode	nECB+/- Pulse 입력 시, Count up nECA+/- Pulse 입력 시, Count down

**WR6/D2 POINV** Drive pulse 출력, nP+P/nP+N (+방향의 Drive pulse)와 nP-P/nP-N (-방향의 Drive pulse)의 출력 신호를 바꿉니다. 이 Bit 를 1 로 하면, +방향의 Drive 에서는 nP-P/nP-N 신호의 Drive pulse 가 출력되고, -방향의 Drive 에서는 nP+P/nP+N 신호의 Drive pulse 가 출력됩니다.

**WR6/D3 AVTRI** 정량 Pulse drive 의 직선 가감속 (사다리꼴)에 있어서 삼각파형을 방지합니다.

0: 무효, 1: 유효

- WR6/D4 VRING 논리위치 Counter 및 실제위치 Counter 의 가변 링 기능을 유효하게 합니다.  
0: 무효, 1: 유효
- WR6/D5 HMINT 자동 원점 복귀 종료 후 Interrupt 신호를 발생시킵니다. 본 bit 를 1 하면 자동원점출력 종료 후 Interrupt 신호가 Low active 되어 Interrupt 를 발생시킨 축의 RR3/D8(HMEND) bit 가 1 을 나타냅니다. Interrupt 를 발생시킨 축의 RR3 register 를 읽고나면 RR3 register 의 bit 는 0 으로 clear 되어 Interrupt 출력 신호는 Hi-Z 으로 돌아옵니다.
- WR6/D7 SMODE S 자 가감속 Drive 시, 지정된 Drive 속도에 도달하는 것을 우선적으로 설정시키고 싶을 때에 1 으로 합니다.
- WR6/D12~8 FE4~0 입력 신호 시 모션 컨트롤 IC 내부의 필터 기능을 유효/무효로 할 지를 설정합니다.  
0: 무효, 1: 유효

지정 Bit	필터 유효의 신호
WR6/D8 (FE0)	EMG <sup>※1</sup> , nLMT+, nLMT-, nIN0, nIN1
WR6/D9 (FE1)	nIN2
WR6/D10 (FE2)	nINPOS, nALARM
WR6/D11 (FE3)	nEXP+, nEXP-
WR6/D12 (FE4)	nIN3

※1. EMG 신호는 X 축의 WR6 register D8 bit 로 설정합니다.

- WR6/D15~13 FE2~0 필터 통과시간을 설정합니다. 입력 신호 필터 기능의 자세한 내용은 '2.8 입력 신호 필터'를 참조하십시오.

WR6/D15~13 (FL2~0)	제거 가능한 최대 노이즈 폭	입력 신호 지연시간
0	1.75 $\mu$ s	2 $\mu$ s
1	224 $\mu$ s	256 $\mu$ s
2	448 $\mu$ s	512 $\mu$ s
3	896 $\mu$ s	1.024 ms
4	1.792 ms	2.048 ms
5	3.584 ms	4.096 ms
6	7.168 ms	8.192 ms
7	14.336 ms	16.384 ms

WR7 register 의 각 Bit 는 자동 원점 복귀 Mode 를 설정합니다.

각 Bit 의 상세한 내용에 대해서는 '2.5.2 서치 속도와 Mode 의 설정'을 참조하십시오.



**Note**

확장 Mode 의 설정 명령은 WR6 과 WR7 register 의 내용이 모두 모션 컨트롤 IC 내부의 확장 Mode register(EM6, 7)에 설정되기 때문에 반드시 WR6 와 WR7 register 의 양쪽 모두에 적정값을 설정해야 합니다.



## 9.17 원점 검출 속도 설정

명령 코드	명령	기호	Data 범위	Data 길이
61h	원점 검출 속도 설정	HV	1~8,000	2 byte

자동 원점 복귀의 Step2, 3 은 저속 서치 속도를 설정합니다.

원점 검출 속도 설정값을 HV 로 하면 원점 검출 속도는 다음 식과 같이 됩니다.

$$\text{Drive 속도(pps)} = V \times \frac{8,000,000}{R}$$

R  
배율

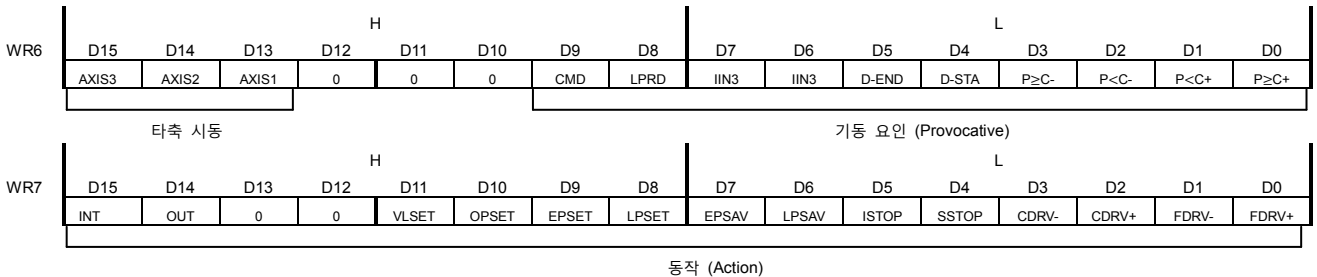
검출 신호가 Active 되었을 때 즉시 정지시키기 위해서 기동속도(SV)보다 낮은 값으로 설정합니다.

자동 원점 복귀에 대해서는 '2.5 자동 원점 복귀 출력'을 참고하십시오.

## 9.18 동기 동작 Mode 설정

명령 코드	명령	기호	Data 범위	Data 길이
64h	동기동작 Mode 설정	SM		4 byte

동기 동작 Mode 의 설정은 아래와 같이 WR6 과 WR7 register 의 각 Bit 에 적정값을 설정한 후에 WR0 register 에 축 지정과 함께 명령 코드(64h)를 기입하면 WR6, 7 register 의 내용이 모션 컨트롤 IC 내부의 동기동작 Mode register(SM6, 7)에 설정됩니다. Reset 시에는 모션 컨트롤 IC 내부의 동기 동작 Mode register(SM6, 7)의 모든 bit 는 0 으로 Clear 됩니다.



각 bit 의 상세한 설명은 '2.6 동기 동작'을 참조하십시오.



## 10 Data 읽기 (Read) 명령

Data 읽기 명령은 각 축의 Register 내용을 Read data register 로 읽어내는 명령입니다.

WR0 register 에 축 지정과 Data 읽기 명령 코드를 기입하면 지정된 Data 가 RR6, 7 register 에 설정됩니다.

CPU 는 RR6, 7 register 를 읽어내는 것에 의해 지정된 Data 를 얻을 수 있습니다.

읽기 Data 는 모두 2 진수 (Binary)입니다. 또한 부의 값은 2 의 보수로 취급합니다.



### Note

- Data 읽기 명령의 명령 처리에 필요로 하는 시간은 최대 250nsec (CLK=16MHz 의 경우) 입니다. 명령을 기입하고 나서, 이 시간 이후에 RR6, 7 register 를 읽어야 합니다.
- 축 지정은 반드시 1 축만 지정합니다. 2 축 이상 지정했을 경우는 X>Y>Z>U 의 우선 순위로 순위가 높은 축의 Data 가 읽어집니다.

### 10.1 논리위치 Counter 읽기

명령 코드	명령	기호	Data 범위	Data 길이
10h	논리위치 Counter 읽기	LP	-2,147,483,648~ +2,147,483,647	4 byte

논리위치 Counter 의 현재값이 RR6, 7 Read data register 에 설정됩니다.

### 10.2 실제위치 Counter 읽기

명령 코드	명령	기호	Data 범위	Data 길이
11 h	실제위치 Counter 읽기	EP	-2,147,483,648~ +2,147,483,647	4 byte

실제위치 Counter 의 현재값이 RR6, 7 Read data register 에 설정됩니다.

### 10.3 현재 Drive 속도 읽기

명령 코드	명령	기호	Data 범위	Data 길이
12 h	현재 Drive 속도 읽기	CV	1~8,000	2 byte

Drive 중 현재 Drive 속도의 값이 RR6, 7 Read data register 에 설정됩니다. Drive 정지 시에는 0 으로 설정되며, Data 의 단위는 Drive 속도 설정값(V)와 같습니다.

### 10.4 현재 가/감속도 읽기

명령 코드	명령	기호	Data 범위	Data 길이
13 h	현재 가/감속도 읽기	CA	1~8,000	2 byte

Drive 중 현재 가속도 또는 감속도의 값이 RR6, 7 Read data register 에 설정됩니다. Data 의 단위는 가속도 설정값(A)와 같습니다.

## 10.5 동기 동작 버퍼 Register 읽기

명령 코드	명령	기호	Data 범위	Data 길이
14 h	동기 동작 버퍼 Register 읽기	BR	-2,147,483,648~ +2,147,483,647	4 byte

동기 동작 버퍼 Register 의 값이 RR6, 7 data register 에 설정됩니다.

## 11 Drive 명령

Drive 명령은 각 축의 Drive pulse 를 출력하는 명령과 출력에 필요한 부수적인 명령입니다. 기록 Data 는 필요하지 않고 WR0 command register 에 축 지정과 명령 코드를 기입하면 즉시 실행됩니다. 복수의 축을 지정하여 같은 명령을 동시에 실행할 수도 있습니다.

Drive 중에는 RR0 주 status register 의 각 축의 n-DRV bit 가 1 이 되고 Drive 가 종료하면 n-DRV bit 는 0 으로 돌아옵니다.

서보모터 Drive 용의 nINPO 신호를 유효로 설정하면 nINPOS 입력 신호가 Active level 이 되고 RR0 주 Status register 의 n-DRV bit 는 0 으로 돌아옵니다.



### Note

Drive 명령의 명령 처리에 필요로 하는 시간은 최대 250nsec (CLK=16MHz 의 경우) 입니다. 다음 명령을 기입할 때에는 이 시간 이후에 설정합니다.

### 11.1 + 방향 정량 Pulse drive

명령 코드	명령
20h	+ 방향 정량 Pulse drive

설정되어 있는 출력 Pulse 수를 nP+P/nP+N 출력 신호에 Pulse 를 출력합니다.

Drive 중은 Drive pulse 를 1 pulse 출력할 때 마다 논리 위치 Counter 가 1 개 Count up 합니다. Drive 명령 기입 전에 출력시키고 싶은 속도 커브에 필요한 Parameter 와 출력 Pulse 수가 올바르게 설정되어 있어야 합니다.

(○: 설정이 필요)

Parameter	출력 시키고 싶은 속도 커브				
	정속	대칭 직선 가감속	비대칭 직선 가감속	대칭 S 자 가감속	비대칭 S 자 가감속
Range (R)	○	○	○	○	○
가속도 증가율 (K)				○	○
감속도 증가율(L)					○
가속도(A)		○	○	○ (8000)	○ (8000)
감속도(D)			○		○ (8000)
기동속도(SV)	○	○	○	○	○
Drive 속도(V)	○	○	○	○	○
출력 Pulse 수(P)	○	○	○	○	○
매뉴얼 감속점 (DP)					○

## 11.2 - 방향 정량 Pulse drive

명령 코드	명령
21h	- 방향 정량 Pulse drive

설정되어 있는 출력 Pulse 수를 수를 nP-P/nP-N 출력 신호에 Pulse 를 출력합니다.

Drive 중은 Drive pulse 를 1 pulse 출력할 때 마다 논리 위치 Counter 가 1 개 Count down 합니다. Drive 명령 기입 전에 출력시키고 싶은 속도 커브에 필요한 Parameter 와 출력 Pulse 수가 올바르게 설정되어 있어야 합니다.

## 11.3 +방향 연속 Pulse drive

명령 코드	명령
22h	+ 방향 연속 Pulse drive

정지 Command 또는 지정된 외부 신호가 Active 될 때 까지 연속해서 nP+P/nP+N 출력 신호에 Pulse 를 출력합니다. Drive 중에는 Drive pulse 를 1 pulse 출력할 때마다 논리 위치 Counter 가 1 개 Count up 합니다.

Drive 명령을 기입하기 전에 출력시키고 싶은 속도 커브에 필요한 Parameter 가 올바르게 설정되어 있어야 합니다.

## 11.4 - 방향 연속 Pulse drive

명령 코드	명령
23h	- 방향 연속 Pulse drive

정지 Command 또는 지정된 외부 신호가 Active 될 때 까지 연속해서 nP-P/nP-N 출력 신호에 Pulse 를 출력합니다. Drive 중에는 Drive pulse 를 1 pulse 출력할 때마다 논리 위치 Counter 가 1 개 Count down 합니다.

Drive 명령을 기입하기 전에 출력시키고 싶은 속도 커브에 필요한 Parameter 가 올바르게 설정되어 있어야 합니다.

## 11.5 Drive 시작 홀드(Hold)

명령 코드	명령
24h	Drive 시작 Hold

Drive 의 시작을 일시정지시켜 놓습니다. 복수의 축 Drive 를 동시 스타트 시킬 때 사용합니다. 동시 스타트 시키고 싶은 축에 본 명령을 기입한 후 Drive 시작명령(25h)으로 전 축 동시에 Drive 를 시작합니다. Drive 중에 본 명령을 기입해도 Drive 는 정지하지 않습니다.

## 11.6 Drive 시작 (Free)/종료 Status clear

명령 코드	명령
25h	Drive 시작/종료 Status clear

Drive 시작 Hold 명령 (24h)에 의해 Drive 시작이 Hold 되고 있는 상태를 해제합니다.

RR1 register 의 drive 종료 Status bit D15~8 을 Clear 합니다.

RR2 register 의 자동원점복귀 IN2 신호 에러 bit D7 (HOME)를 Clear 합니다.

## 11.7 Drive 감속 정지

명령 코드	명령
26h	Drive 감속정지

Drive pulse 출력 중에 감속정지 시킵니다. Drive 속도가 기동 속도보다 낮은 경우에는 즉시 정지합니다. 보간 Drive 중에 주축에 대해서 본 명령 또는 Drive 즉시 명령을 기입하면 보간 Drive 는 정지합니다. Drive 가 정지하고 있을 때의 기입은 아무런 동작을 하지 않습니다.

## 11.8 Drive 즉시 정지

명령 코드	명령
27h	Drive 즉시정지

Drive pulse 출력 중에 즉시정지시킵니다. 가감속 Drive 에 대해서도 즉시정지합니다.

Drive 가 정지하고 있을 때의 기입은 아무런 동작을 하지 않습니다.





## 12 보간 명령

보간 명령은 2 축/3 축 직선보간, CW/CCW 원호 보간, 2 축/3 축 Bit pattern 보간 및 보간 Drive 에 필요한 부수적인 명령으로 구성됩니다. 보간 명령은 WR0 command register 의 D11~8 bit 의 축 지정이 필요 없습니다.

0 으로 설정합니다.

어느 보간을 실시하는 경우도 보간 Drive 를 시작하기 전에 공통으로 필요한 것은 다음의 2 가지 입니다.

- ① 보간을 실시하는 축을 지정합니다. (WR5 register 의 D5~D0 set)
- ② 주축으로 지정한 축의 속도 Parameter 를 설정합니다.

보간 Drive 중에는 RR0 주 status register 의 D8 (I-DRV) bit 가 1 이 되고 Drive 가 종료하면 0 으로 돌아옵니다. 보간 Drive 중에는 보간을 실시하고 있는 축의 n-DRV bit 도 1 이 됩니다.



### Note

보간 명령의 명령 처리에 필요로 하는 시간은 최대 250nsec(CLK=16MHz 의 경우) 입니다. 다음 명령의 기입은 이 시간의 이후에 설정하십시오.

### 12.1 2 축 직선 보간 Drive

명령 코드	명령
30h	2 축 직선 보간 Drive

현재 좌표에서 종점 좌표까지 2 축 직선보간 합니다.

Drive 전에 보간을 실시하는 2 축의 각 종점을 상대값으로 출력 Pulse(P)에 설정합니다.

### 12.2 3 축 직선 보간 Drive

명령 코드	명령
31h	3 축 직선 보간 Drive

현재 좌표에서 종점 좌표까지 3 축 직선보간 합니다.

Drive 전에 보간을 실시하는 3 축의 각 종점을 상대값으로 출력 Pulse(P)에 설정합니다.

### 12.3 CW 원호보간 Drive

명령 코드	명령
32h	CW 원호 보간 Drive

지정된 중심 좌표를 중심으로 현재 좌표에서 종점 좌표까지 시계방향으로 원호보간 합니다.

Drive 전에 보간을 실시하는 2 축의 현재위치를 원호 중심점(C)에 설정하고 현재 위치에 대한 종점을 상대값으로 출력 Pulse(P)에 설정합니다. 종점좌표를 (0, 0)으로 설정하면 진원을 그립니다.

## 12.4 CCW 원호 보간 Drive

명령 코드	명령
33h	CCW 원호 보간 Drive

지정된 중심 좌표를 중심으로 현재 좌표에서 종점 좌표까지 반시계방향으로 원호보간 합니다. Drive 전에 보간을 실시하는 2 축의 현재위치를 원호 중심점(C)에 설정하고 현재 위치에 대한 종점을 상대값으로 출력 Pulse(P)에 설정합니다. 종점좌표를 (0, 0)으로 설정하면 진원을 그립니다.

## 12.5 2 축 bit pattern 보간 Drive

명령 코드	명령
34h	2 축 bit pattern 보간 Drive

2 축 bit pattern 보간을 실행합니다. Drive 전에 보간을 실시하는 2 축의 +방향/-방향의 bit data 를 설정합니다 Drive 전에 설정할 수 있는 bit data 의 크기는 각 축  $16 \times 3 = 48$  bit 입니다. 이것을 넘는 경우에는 Drive 중에 보충해 갑니다.

## 12.6 3 축 bit pattern 보간 Drive

명령 코드	명령
35h	3 축 bit pattern 보간 Drive

3 축 bit pattern 보간을 실행합니다. Drive 전에 보간을 실시하는 3 축의 +방향/-방향의 bit data 를 설정합니다 Drive 전에 설정할 수 있는 bit data 의 크기는 각 축  $16 \times 3 = 48$  bit 입니다. 이것을 넘는 경우에는 Drive 중에 보충해 갑니다.

## 12.7 BP register 기입 가능

명령 코드	명령
36h	BP register 기입 가능

Bit pattern 보간의 Bit pattern data 를 기입하는 Register (BP1P/M, BP2P/M, BP3P/M)의 기입을 가능하게 하는 명령으로 이 명령 발행에 의해서 nWR2~nWR5 register 의 기입은 할 수 없습니다.

Reset 시에는 Bit pattern data 기입은 할 수 없습니다.

## 12.8 BP register 기입 불가

명령 코드	명령
37h	BP register 기입 불가

Bit pattern 보간의 Bit pattern data 를 기입하는 Register (BP1P/M, BP2P/M, BP3P/M)에 기입을 불가능하게 설정합니다. 명령에 의해서 nWR2~nWR5 register 에의 기입은 가능하게 됩니다.

## 12.9 BP data stack

명령 코드	명령
38h	BP data stack

Bit pattern data 기입 Register (BP1P/M, BP2P/M, BP3P/M)에 기입해진 Bit pattern data 를 Registerfh 이동시켜 저장합니다. BP data stack 명령을 발행하면 Stack Counter(SC)가 1 개 증가합니다. Stack Counter(SC)가 3 이면 본 명령을 기입할 수 없습니다.

## 12.10 BP data clear

명령 코드	명령
39h	BP data clear

내부에 축적된 Bit pattern data 를 모두 Clear 하여 Stack Counter(SC)를 0 으로 합니다.

## 12.11 보간 Single step

명령 코드	명령
3Ah	보간 Single step

보간 Drive 를 1 pulse 마다 Step 출력합니다. WR5 register 의 D12 bit 를 1 로 하여 Command 에 의한 보간 Step mode 로 설정하고 보간 Drive 명령을 발행하고 나서 Singel step 을 실시합니다.

## 12.12 감속 유효

명령 코드	명령
3Bh	감속 유효

가감속으로 보간 Drive 를 실시할 때 자동 감속 또는 매뉴얼 감속을 유효로 합니다.

단독의 보간 Drive 를 가감속으로 실시할 때는 Drive 전에 반드시 본 명령을 실행할 필요가 있습니다. 연속 보간으로 감속을 무효로 하여 보간 Drive 를 시작합니다. 감속시키는 최종 보간 세그먼트(Segment)의 보간 명령 기입전에 감속 유효 명령을 기입합니다. Reset 시에는 감속 무효 상태가 됩니다. 본 명령에 의해 감속을 유효 상태로 하면 감속 무효 명령(3C)이 쓰기 전까지, Reset 할 때까지 유효상태가 됩니다.

감속 유효/무효는 보간 Drive 시에만 가능하며 각 축을 독립적으로 Drive 할 경우에는 자동 감속 또는 매뉴얼 감속은 항상 유효 상태입니다.

## 12.13 감속 무효

명령 코드	명령
3Ch	감속 무효

가감속으로 보간 Drive 를 실시할 때 자동 감속 또는 매뉴얼 감속을 무효 상태로 합니다.

## 12.14 보간 Interrupt clear

명령 코드	명령
3Dh	보간 Interrupt clear

Bit pattern 보간 또는 연속보간으로 발생한 Interrupt 를 Clear 합니다.

Bit pattern 보간에서는 WR5 register 의 D15 bit 를 1 로 하면 Stack Counter(SC)가 2 에서 1 로 바뀌어 Interrupt 가 발생합니다. 연속 보간에서는 WR5 register 의 D14 bit 를 1 로 하면 다음 보간 세그먼트(Segment)의 Data 및 보간 Drive 명령이 기입되면 Interrupt 가 발생합니다.

## 13 그 밖의 명령



### Note

명령의 명령 처리에 필요로 하는 시간은 최대 250nsec(CLK=16MHz의 경우)입니다. 다음 명령을 기입할 때에는 이 시간 이후에 기입하십시오.

### 13.1 자동 원점 복귀 실행

명령 코드	명령
62h	자동 원점 복귀 실행

자동 원점 복귀를 실행합니다. 실행전에 자동 원점 복귀 Mode 나 각 Parameter 를 올바르게 설정해 둘 필요가 있습니다. 자동 원점 복귀의 자세한 사항은 '2.5 자동 원점 복귀 출력'을 참조하십시오.

### 13.2 동기 동작 기동

명령 코드	명령
65h	동기 동작 기동

동기 동작을 본 명령에 의해서 기동시킵니다. 동기동작 Mode 설정 명령에 의해 기동 요인의 WR6/D9 (CMD) bit 를 1로 설정해야 합니다. 동기 동작의 자세한 사항은 '2.6 동기 동작'을 참조하십시오.

### 13.3 NOP

명령 코드	명령
0Fh	NOP

명령은 아무것도 실행되지 않습니다.

각 축의 WR1~3 register, RR1~3 register 를 선택하여 축 변경에 사용합니다.



## 14 제어 프로그램의 예

이 장에서는 C 언어에 의한 모션 컨트롤러의 제어 프로그램의 예를 나타냅니다. 16 bit bus 구성의 예입니다.

```
#include <studio.h>
#include <conio.h>
//---- 모션 컨트롤 IC register address 정의----
#define          adr          0x2a0    // 베이스 address
#define          wr0          0x0      // command register
#define          wr1          0x2      // mode register 1
#define          wr2          0x4      // mode register 2
#define          wr3          0x6      // mode register 3
#define          wr4          0x8      // output register
#define          wr5          0xa      // 보간 mode register
#define          wr6          0xc      // 하위 write data register
#define          wr7          0xe      // 상위 write data register
#define          rr0          0x0      // 주 status register
#define          rr1          0x2      // status register 1
#define          rr2          0x4      // status register 2
#define          rr3          0x6      // status register 3
#define          rr4          0x8      // input register 1
#define          rr5          0xa      // input register 2
#define          rr6          0xc      // 하위 read data register
#define          rr7          0xe      // 상위 read data register
#define          bp1p         0x4      // B P 제 1 축 +방향 data register
#define          bp1m         0x6      // B P 제 1 축 -방향 data register
#define          bp2p         0x8      // B P 제 2 축 +방향 data register
#define          bp2m         0xa      // B P 제 2 축 -방향 data register
#define          bp3p         0xc      // B P 제 3 축 +방향 data register
#define          bp3m         0xe      // B P 제 3 축 -방향 data register
//wreg1 (축지정, data) -----write register 1 설정
void wreg 1 (int axis, int wdata)
{
  outpw(adr+wr0, (axis << 8) + 0xf); // 축지정
  outpw (adr+wr1, wdata);
}
//wreg2 (축지정, data) -----write register 2 설정
void wreg 2 (int axis, int wdata)
{
  outpw(adr+wr0, (axis << 8) + 0xf); // 축지정
  outpw (adr+wr2, wdata);
}
```

```

}
//wreg3 (축 지정, data) -----write register 3 설정
void wreg 3 (int axis, int wdata)
{
  outpw(adr+wr0, (axis << 8) + 0xf); // 축 지정
  outpw (adr+wr3, wdata);
}
// command (축 지정, 명령코드) -----명령 기입
void command (int axis, int cmd)
{
  outpw(adr+wr0, (axis <<8) + cmd);
}
//range (축 지정, data) -----range (R) 설정
void range (int axis, long wdata)
{
  outpw(adr+wr7, (wdata >>16) & 0xffff);
  outpw(adr+wr6, wdata & 0xffff);
  outpw(adr+wr0, axis<<8) + 0x00;
}
//acac (축 지정, data) -----가속도 증가율(K) 설정
void acac (int axis, int wdata)
{
  outpw (adr+wr6, wdata);
  outpw (adr+wr0, (axis << 8) +0x01;
}
// dcac (축 지정, data) -----감속도 증가율(L) 설정
void dcac (int axis, int wdata)
{
  outpw (adr+wr6, wdata);
  outpw (adr+wr0, (axis <<8) + 0x0e
}
// acc (축 지정, data) -----가속도 (A) 설정
void acc (int axis, int wdata)
{
  outpw (adr+wr6, wdata);
  outpw (adr+wr0, (axis <<8) + 0x02;
}
//dec (축 지정, data) -----감속도 (D) 설정
void dec (int axis, int wdata)
{
  outpw (adr+wr6, wdata);

```



```

    outpw (adr+wr0, (axis <<8) +0x03);
}
// startv (축 지정, data) ----- 기동 속도 (SV) 설정
void startv (int axis, int wdata)
{
    outpw (adr+wr6, wdata);
    outpw (adr+wr0, (axis << 8) + 0x04);
}
// speed (축 지정, data) ----- drive 속도(V) 설정
void speed (int axis, int wdata)
{
    outpw (adr+wr6, wdata);
    outpw (adr+wr0, (axis << 8) + 0x05);
}
// pulse (축 지정, data) ----- 출력 pulse 수/중점(P) 설정
void pulse (int axis, long wdata)
{
    outpw (adr+wr7, (wdata >> 16) & 0xffff);
    outpw (adr+wr6, wdata & 0xffff);
    outpw (adr+wr0, (axis << 8) + 0x06);
}
//decp (축 지정, data) ----- 매뉴얼 감속점(DP) 설정
void decp (int axis, long wdata)
{
    outpw (adr+wr7, (wdata >> 16) & 0xffff);
    outpw (adr+wr6, wdata & 0xffff);
    outpw (adr+wr0, (axis << 8) + 0x07);
}
// center (축 지정, data) ----- 원호 중심점(C) 설정
void center (int axis, long wdata)
{
    outpw (adr+wr7, (wdata >> 16) & 0xffff);
    outpw (adr+wr6, wdata & 0xffff);
    outpw (adr+wr0, (axis <<8) + 0x08);
}
//lp (축 지정, data) ----- 논리위치 Counter (LP) 설정
void lp (int axis, long wdata)
{
    outpw (adr+wr7, (wdata >>16) & 0xffff);
    outpw (adr+wr6, wdata & 0xffff);
    outpw (adr+wr0, (axis <<8) + 0x09);
}

```

```

}
//ep (축 지정, data) ----- 실제위치 Counter (EP) 설정
void ep (int axis, long wdata)
{
  outpw (adr+wr7, (wdata >> 16) & 0xffff);
  outpw (adr+wr6, wdata & 0xffff);
  outpw (adr+wr0, (axis << 8) + 0x0a);
}
// compp (축 지정, data) ----- COMP+ (CP) 설정
void compp (int axis, long wdata)
{
  outpw (adr+wr7, (wdata >> 16) & 0xffff);
  outpw (adr+wr6, wdata & 0xffff);
  outpw (adr+wr0, (axis << 8) + 0x0b);
}
// compm (축 지정, data) ----- COMP- (CP) 설정
void compm (int axis, long wdata)
{
  outpw (adr+wr7, (wdata >> 16) & 0xffff);
  outpw (adr+wr6, wdata & 0xffff);
  outpw (adr+wr0, (axis << 8) + 0x0c);
}
//accfst (축 지정, data) ----- 가속 Counter Offset (AO) 설정
void accfst (int axis, long wdata)
{
  outpw (adr+wr7, (wdata >> 16) & 0xffff);
  outpw (adr+wr6, wdata & 0xffff);
  outpw (adr+wr0, (axis << 8) + 0x0d);
}
// hsspeed (축 지정, data) ----- 원점 검출 속도 (HV) 설정
void hsspeed (int axis, int wdata)
{
  outpw (adr+wr6, wdata);
  outpw (adr+wr0, (axis <<8) + 0x61);
}
// expmode (축 지정, data) ----- 확장 mode (EM) 설정
void expmode (int axis, int em6data, int em7data)
{
  outpw (adr+wr6, em6data);
  outpw (adr+wr7, em7data);
  outpw (adr+wr0, (axis << 8) + 0x60);
}

```

```

}
// syncmode (축 지정, data) ----- 동기 동작 mode (SM) 설정
void syncmode (int axis, int sm6data, int sm7data)
{
  outpw (adr+wr6, sm6data);
  outpw (adr+wr7, sm7data);
  outpw (adr+wr0, (axis << 8) + 0x64);
}
// readlp (축 지정) ----- 논리위치 Counter 값 (LP) 읽기
long readlp (int axis)
{
  long a; long d6; long d7;
  outpw (adr+wr0, (axis << 8) + 0x10);
  d6 = inpw (adr+rr6); d7 = inpw (adr+rr7);
  a = d6 + (d7 << 16);
  return (a);
}
// readep (축 지정) ----- 실제위치 Counter 값 (LP) 읽기
long readep (int axis)
{
  long a; long d6; long d7;
  outpw (adr+wr0, (axis << 8) + 0x11);
  d6 = inpw (adr+rr6); d7 = inpw (adr+rr7);
  a = d6 + (d7 << 16);
  return (a);
}
// wait (축 지정) ----- drive 종료 대기
void wait (int axis)
{
  while (inpw (adr+rr0) & axis);
}
// next_wait ( ) ----- 연속 보간 다음 data set 대기
void next_wait (void)
{
  while ((inpw (adr+rr0) & 0x0200) == 0x0);
}
// bp_wait ( ) ----- BP 보간 다음 data set 대기
void bp_wait (void)
{
  while ((inpw (adr+rr0) & 0x6000) == 0x6000);
}

```

```

// homesrch ( ) ----- 전축•원점 서치
//
// -----X 축 원점 서치 -----
// Step1 -방향에 20,000 pps 로 원점 근접(IN0) 신호 고속 서치
// Step2 -방향에 500 pps 로 원점(IN1) 신호 저속 서치
// Step3 -방향에 500 pps 로 Z 상(IN2) 신호 저속 서치
// Z 상 검출 시 편차 Counter clear 출력
// Step4 +방향에 20,000pps 로 3,500 pulse Offset 고속 이동
// -----Y 축 원점 서치 -----
// Step1 -방향에 20,000 pps 로 원점 근접(IN0) 신호 고속 서치
// Step2 -방향에 500 pps 로 원점(IN1) 신호 저속 서치
// Step3 -방향에 500 pps 로 Z 상(IN2) 신호 저속 서치
// Z 상 검출 시 편차 Counter clear 출력
// Step4 +방향에 20,000pps 로 700 pulse Offset 고속 이동
// -----Z 축 원점 서치 -----
// Step 1 고속 서치: 없음
// Step2 + 방향에 400pps 로 원점(IN1) 신호 저속 서치
// Step3 Z 상 서치: 없음
// Step4 -방향에 400pps 로 20 pulse Offset 이동
// -----U 축 원점 서치 -----
// Step 1 고속 서치: 없음
// Step2 -방향에 300pps 로 원점(IN1) 신호 저속 서치
// Step3 Z 상 서치: 없음
// Step4 Offset 이동: 없음
void homesrch (void)
{
// X, Y 축 원점 서치의 parameter 설정
// (mode 설정은 main 의 초기 설정을 참조)
speed (0x3, 2000);          // Step1, 4 고속 속도: 20,000pps
hsspeed (0x3, 50);         // Step2, 3 저속 속도: 500pps
pulse (0x1, 350 0);        // X 축 Offset: 3,500 pulse
pulse (0x2, 700 );         // Y 축 Offset: 700 pulse
// Z 축 원점 서치의 parameter 설정
speed (0x4, 40);           // Step4 이동 속도: 400pps
hsspeed (0x4, 40);         // Step2 서치 속도: 400pps
pulse (0x4, 20) ;          // Offset: 20 pulse
// U 축 원점 서치의 Parameter 설정
hsspeed (0x8, 30);         // Step2 서치 속도: 300pps
command (0xf, 0x62);       // 전축 자동 원점 복귀 실행
wait (0xf);                // 전축 종료 대기
if (inpw(adr+rr0) & 0x0010) // 에러 표시

```

```

{
printf("X-axis Home Search Error \n");
}
if (inpw(adr+rr0) & 0x0020)
{
printf("Y-axis Home Search Error \n");
}
if (inpw(adr+rr0) & 0x0040)
{
printf("Z-axis Home Search Error \n");
}
if (inpw(adr+rr0) & 0x0080)
{
printf("U-axis Home Search Error \n");
}
}
void main (void)
{
int count;
outpw (adr+wr0, 0x8000); // 소프트 reset
for (count = 0; count <2; ++ count);
command (0x3, 0 xf); // -----X, Y 축 mode 설정-----
outpw (adr+wr1, 0x0000); // mode register 1
//D15~8 : 0 interrupt 모두 금지
//D7: 0 IN3 신호: 무효
//D6: 0 IN3 신호 논리: Low active
//D5: 0 IN2 신호: 무효
//D4: 0 IN2 신호 논리: Low active
//D3: 0 IN1 신호: 무효
//D2: 0 IN1 신호 논리: Low active
//D1: 0 IN0 신호: 무효
//D0: 0 IN0 신호 논리: Low active
outpw (adr+wr2, 0xe0000); //mode register 2
//D15:1 INPOS 입력: 유효
//D14:1 INPOS 입력 논리: High active
//D13:1 ALARM 입력: 유효
//D12:0 ALARM 입력 논리: Low active
//D11:0
//D10:0 Encoder 입력 분주: 1/1
//D9:0 Encoder 입력 방식: 2 상 pulse
//D8:0 drive pulse 방향 논리

```

```

//D7:0 drive pulse 논리: 정논리
//D6:0 drive pulse 방식: 2 pulse
//D5:0 COMP 대상: 논리 위치 Counter
//D4:0 - limit 논리: Low active
//D3:0 + limit 논리: Low active
//D2:0 limit 정지 mode: 감속 정지
//D1:0 소프트 limit -: 무효
//D0:0 소프트 limit +: 무효
outpw (adr+wr3, 0x0000);          //mode register 3
//D15~12:0000
//11:0 범용 출력 OUT7: Low
//D10:0 범용 출력 OUT6: Low
//D9:0 범용 출력 OUT5: Low
//D8:0 범용 출력 OUT4: Low
//D7:0 drive 상태 출력: 무효
//D6: 0
//D5: 0
//D4: 0 외부 조작 신호 동작: 무효
//D3: 0
//D2: 0 가감 속도 커브 : 직선 가감 속도 (사다리꼴)
//D1: 0 가감 속도의 대칭/비대칭: 대칭
//D0: 0 정량 pulse drive 의 감속: 자동 감속
expmode (0x3, 0 x5d08, 0x497f);   //확장 mode
//[입력 신호 필터 그위]
//W6/D15~13:010 입력 신호 필터 지연: 512  $\mu$ s
//W6/D12:1 IN3 신호 필터 : 유효
//W6/D11:1 EXPP, EXPM, EXPLS 필터: 유효
//W6/D10:1 INPOS, ALARM 신호 필터: 유효
//W6/D9:0 IN2 신호 필터: 무효
//W6/D8:1 EMGN, LMTP/M, IN1, 0 필터: 유효
//W6/D7:0
//W6/D6:0
//W6/D5:0 자동 원점 복귀 종료 새치기: 금지
//W6/D4:0 LP/EP 가변 링 기능: 무효
//W6/D3:1 직선 가감 속도시의 삼각 방지: 유효
//W6/D2:0 pulse 출력의 교체: 무효
//W6/D1:0 EP 증감 반전: 무효
//W6/D0:0 IN2 신호에 의한 EP clear: 무효
//[자동 원점 복귀 mode]
//W7/D15~D13 010 편차 counter clear pulse 폭: 100  $\mu$ s
//W7/D12 0 편차 counter clear 출력의 논리 level: High

```

```

//W7/D11 1 편차 counter clear 출력: 유효
//W7/D10 0 Limit 신호를 원점 신호로서 사용: 무효
//W7/D9 0 Z 상 신호 AND 원점 신호: 무효
//W7/D8 1 논리/실제위치 Counter clear: 유효
//W7/D7 0 Step4 이동 방향: + 방향
//W7/D6 1 Step4: 유효
//W7/D5 1 Step3 검출 방향: - 방향
//W7/D4 1 Step3: 유효
//W7/D3 1 Step2 검출 방향: - 방향
//W7/D2 1 Step2: 유효
//W7/D1 1 Step1 검출 방향: - 방향
//W7/D0 1 Step1: 유효
//-----X, Y 축 동작 parameter 초기 설정 -----
accfst (0x3, 0);           //AO = 0
range (0x3, 800000);      // R = 800000 (배율= 10)
acac (0x3, 1010);        //K = 1010 (가/감속 증가율 = 619kpps/sec2)
dcac (0x3, 1010);        //L = 1010 (감속도 증가율 = 619kpps/sec2)
acc (0x3, 100);          // A = 100 (가/감속도 = 125kpps/sec)
dec (0x3, 100);          // D = 100 (감속도 = 125kpps/sec)
starv (0x3, 100);        //SV = 100 (기동속도= 1000pps)
speed (0x3, 4000);        // V = 4000 (drive 속도 = 40000pps)
pulse (0x3, 100000);     // P = 100000 (출력 pulse 수 = 100000)
lp (0x3, 0);             // LP= 0 (논리위치 counter= 0)
ep (0x3, 0);             // EP= 0 (실제위치 counter= 0)
command (0xc, 0 xf);     //-----Z, U 축 mode 설정 -----
outpw (adr+wr1, 0x0000); //mode register 1
//D15~8: 0 interrupt 모두 금지
//D7: 0 IN3 신호: 무효
//D6: 0 IN3 신호 논리: Low active
//D5: 0 IN2 신호: 무효
//D4: 0 IN2 신호 논리: Low active
//D3: 0 IN1 신호: 무효
//D2: 0 IN1 신호 논리: Low active
//D1: 0 IN0 신호: 무효
//D0: 0 IN0 신호 논리: Low active
outpw (adr+wr2, 0x0000); //mode register 2
//D15:0 INPOS 입력: 무효
//D14:0 INPOS 입력 논리: Low active
//D13:0 ALARM 입력: 무효
//D12:0 ALARM 입력 논리: Low active
//D11:0

```

```

//D10:0 Encoder 입력 분주: 1/1
//D9:0 Encoder 입력 방식: 2 상 pulse
//D8:0 Drive pulse 방향 논리:
//D7:0 Drive pulse 논리: 정논리
//D6:0 Drive pulse 방식: 2 pulse
//D5:0 COMP 대상: 논리위치 Counter
//D4:0 -Limit 논리: Low active
//D3:0 + Limit 논리: Low active
//D2:0 Limit 정지 mode: 감속 정지
//D1:0 소프트 Limit -: 무효
//D0:0 소프트 Limit+: 무효
outpw (adr+wr3, 0x0000); // mode register 3
//D15~12: 0000
//D11:0 범용 출력 OUT7: Low
//D10:0 범용 출력 OUT6: Low
//D9:0 범용 출력 OUT5: Low
//D8:0 범용 출력 OUT4: Low
//D7:0 drive 상태 출력: 무효
//D6:0
//D5:0
//D4:0 외부 조작 신호 동작: 무효
//D3:0
//D2:0 가감 속도 커브 : 직선 가감 속도 (사다리꼴)
//D1:0 가감 속도의 대칭/비대칭: 대칭
//D0:0 정량 pulse drive 의 감속: 자동 감속
// Z 축과 U 축의 자동 원점 복귀가 다르므로,
// 이하의 확장 mode 는 개별적으로 설정한다.
expmode (0x4, 0 x5d08, 0x01c4); //Z 축 확장 mode
// [입력 신호 필터 그 외]
//W6/D15~13: 010 입력 신호 필터 지연: 512  $\mu$ s
//W6/D12:1 IN3 신호 필터: 유효
//W6/D11:1 EXPP, EXPM, EXPLS 필터: 유효
//W6/D10:1 INPOS, ALARM 신호 필터: 유효
//W6/D9:0 IN2 신호 필터: 무효
//W6/D8:1 EMGN, LMTP/M, IN1, 0 필터: 유효
//W6/D7:0
//W6/D5: 0 자동 원점 복귀 종료 새치기: 금지
//W6/D4: 0 LP/EP 가변 링 기능: 무효
//W6/D3: 1 직선 가감 속도시의 삼각 방지: 유효
//W6/D2: 0 Pulse 출력의 교체: 무효
//W6/D1: 0 EP 증감 반전: 무효

```



```

//W6/D0: 0 IN2 신호에 의한 EP clear: 무효
// [자동 원점 복귀 mode]
//W7/D15~D13 000 편차 counter clear pulse 폭 :
//W7/D12 0 편차 Counter clear 출력의 논리 level:
//W7/D11 0 편차 Counter clear 출력: 무효
//W7/D10 0 Limit 신호를 원점 신호로서 사용: 무효
//W7/D9 0 Z 상 신호 AND 원점 신호: 무효
//W7/D8 1 논리/실제 위치 Counter clear: 유효
//W7/D7 1 Step4 이동 방향: -방향
//W7/D6 1 Step4 : 유효
//W7/D5 0 Step3 검출 방향:
//W7/D4 0 Step3: 무효
//W7/D3 0 Step2 검출 방향: + 방향
//W7/D2 1 Step2: 유효
//W7/D1 0 Step1 검출 방향:
//W7/D0 0 Step1: 무효
expmode (0x8, 0 x5d08x 0x010c); // U 축 확장 mode
// [입력 신호 필터 그 외]
//W6/D15~13: 010 입력 신호 필터 지연: 512  $\mu$ s
//W6/D12: 1 IN3 신호 필터: 유효
//W6/D11: 1 EXPP, EXPM, EXPLS 필터: 유효
//W6/D10: 1 INPOS, ALARM 신호 필터: 유효
//W6/D9: 0 IN2 신호 필터: 무효
//W6/D8: 1 EMGN, LMTP/M, IN1, 0 필터: 유효
//W6/D7: 0
//W6/D6: 0
//W6/D5: 0 자동 원점 복귀 종료 새치기: 금지
//W6/D4: 0 LP/EP 가변 링 기능: 무효
//W6/D3: 1 직선 가감 속도시의 삼각 방지: 유효
//W6/D2: 0 Pulse 출력의 교체: 무효
//W6/D1: 0 EP 증감 반전: 무효
//W6/D0: 0 IN2 신호에 의한 EP clear: 무효
// [자동 원점 복귀 mode]
//W7/D15~D13 0000 편차 counter clear pulse 폭:
//W7/D12 0 편차 counter clear 출력의 논리 level:
//W7/D11 0 편차 counter clear 출력: 무효
//W7/D10 0 limit 신호를 원점 신호로서 사용: 무효
//W7/D9 0 Z 상 신호 AND 원점 신호: 무효
//W7/D8 1 논리/실제 위치 counter clear: 유효
//W7/D7 0 Step4 이동 방향:
//W7/D6 0 Step4: 무효

```

```

//W7/D5 0 Step3 검출 방향:
//W7/D4 0 Step3: 무효
//W7/D3 1 Step2 검출 방향: -방향
//W7/D2 1 Step2: 유효
//W7/D1 0 Step1 검출 방향:
//W7/D0 0 Step1: 무효
//-----Z, U 축 동작 Parameter 초기 설정 -----
accfst (0xc, 0);                //AO = 0
range (0xc, 800000);           // R = 800000 (배율 = 10)
acac (0xc, 1010);              // K = 1010 (가/감속도 증가율=619kpps/sec2)
dcac (0xc, 1010);              // L = 1010 (감속도 증가율=619kpps/sec2)
acc (0xc, 100);                 // A = 100 (가/감속도 =125 kpps/sec)
dec (0xc, 100);                 // D = 100 (감속도 =125 kpps/sec)
startv (0xc, 50);              // SV = 50 (기동속도= 500pps)
speed (0xc, 40);                // V = 40 (drive 속도= 400pps)
pulse (0xc, 10);               // P = 10 (출력 pulse 수= 10)
lp (0xc, 0);                    // LP = 0 (논리위치 Counter= 0)
//-----범용 출력 register 초기 설정 -----
outpw(adr+wr4, 0x0000);        // 00000000 00000000
//-----보간 mode register 초기 설정 -----
outpw (adr+wr5, 0x0124);       // 00000001 00100100
ax1=x, ax2=y, ax3=z, 선속 일정
//----- drive 시작 -----
// homesrch ( ); //-----전축원점 서치 -----
//-----X, Y 축 직선 가감 속도 drive-----
acc (0x3, 200);                 //A = 200 (가/감속도 = 250 kpps/sec)
speed (0x3, 4000);              //V = 4000 (drive 속도 = 40000 pps)
pulse (0x1, 80000);             //xP = 80000
pulse (0x2, 40000);             //yP = 40000
command (0x3, 0 x20);           //+ 정량 pulse drive
wait (0x3);                      // drive 종료 대기
//-----X 축 비대칭 직선 가감 속도 drive -----
wreg3 (0x1, 0x0002);            // 가속*감속 개별 (비대칭) mode
acc (0x1, 200);                  // xA = 200 (가/감속도 = 250kpps/sec)
dec (0x1, 50);                  // xD = 50 (감속도 = 62.5 kpps/sec)
speed (0x1, 4000);              // xV = 4000 (drive 속도 = 40000pps)
pulse (0x1, 80000);             // xP = 80000
command (0x1, 0 x20);           // + 정량 pulse drive
wait (0x1);                      // drive 종료 대기
wreg3 (0x1, 0x0000);            // 가속*감속 개별 mode 해제
//-----X, Y 축 S 자 가감 속도 drive -----

```

```

wreg3 (0x3, 0x0004);           //S 자 mode
acac (0x3, 1010);             // K = 1010 (가속도 증가율= 619kpps/sec²)
acc (0x3, 200);               // A = 200 (가/감속도 = 250kpps/sec)
speed (0x3, 4000);            // V = 4000 (drive 속도 = 40000pps)
pulse (0x1, 50000);           // xP = 50000
pulse (0x2, 25000);           // yP = 25000
command (0x3, 0 x21);         // - 정량 pulse drive
wait (0x3) ;
wreg3 (0x3, 0x0000);           //S 자 가감 속도 mode 해제
//-----Z 축 정속 drive -----
startv (0x4, 40);             // SV= 40 (기동속도= 400pps)
speed (0x4, 40);              // V= 40 (drive 속도 = 400pps)
pulse (0x4, 700);             // P= 700
command (0x4, 0 x20);         // + 정량 pulse drive
wait (0x4);                   // (400pps 로 700 pulse +방향에 이동)
pulse (0x4, 350);            // P= 350
command (0x4, 0 x21);         // - 정량 pulse drive
wait (0x4);                   // (400pps 로 350 pulse - 방향에 이동)
//-----X, Y 축 직선 보간 drive -----
outpw (adr+wr5, 0x0124);       // ax1=x, ax2=y, ax3=z, 선속일정
range (0x1, 800000);           // ax1/R = 800000 (배율= 10)
range (0x2, 11311371);        // ax2/R = 800000×1.414
speed (0x1, 100);             // ax1/V = 100 (drive 속도 = 1000pps 정속)
pulse (0x1, 5000);            // xP = +5000 (종점 X= +5000)
pulse (0x2, -2000);           // yP = -2000 (종점 Y= -2000)
command (0x0, 0x30);          // 2 축 직선 보간
wait (0x3);
//-----X, Y 축 원호 보간 drive -----
outpw (adr+wr5, 0x0124);       // ax1=x, ax2=y, ax3=z, 선속일정
range (0x1, 800000);           // ax1/R = 800000 (배율= 10)
next_wait ( );
pulse (0x1, -45 00);           // Seg 5
pulse (0x2, 0);
command (0, 0x3 0);
next_wait ( );
center (0x1, 0);               // Seg 6
center (0x2, -1 500);
pulse (0x1, -15 00);
pulse (0x2, -15 00);
command (0, 0x3 3);
next_wait ( );

```

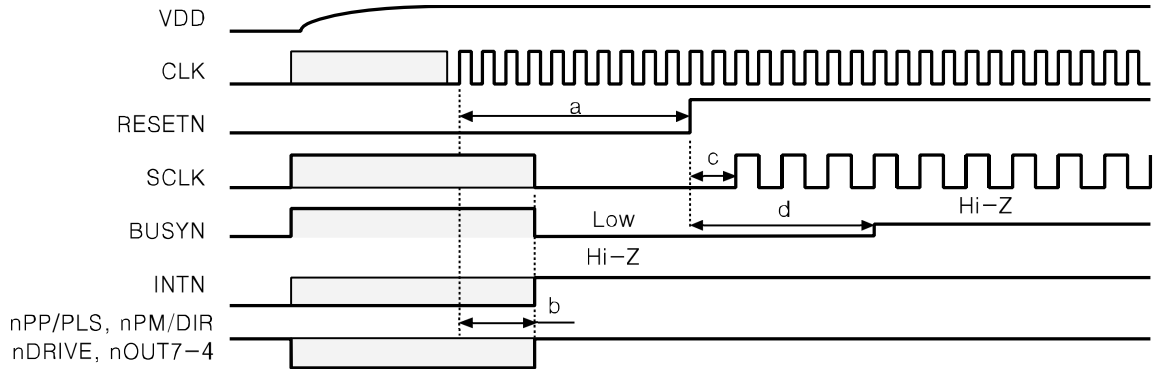
```

pulse (0x1, 0);           // Seg 7
pulse (0x2, -15 00);
command (0, 0x3 0);
next_wait ( );
center (0x1, 15 00);     // Seg 8
center (0x2, 0) ;
pulse (0x1, 150 0);
pulse (0x2, -15 00);
command (0, 0x3 3);
wait (0x3);
//-----동기 동작 (2.6.1 동기 동작의 예) -----
// Y 축이 위치 15000 을 통과하면 (자),
// Z 축의 +방향 정량 pulse drive 시작
range (0x6, 800000);     // R = 800000 (배율= 10)
acc (0x6, 400);         // A = 400 (가/감속도= 500kpps/sec)
startv(0x6, 50);        //SV= 50 (기동 속도=50pps)
speed (0x6, 3000);      // V= 3000 (drive 속도= 30kpps)
pulse (0x2, 50000);     // yP= 50000 (Y 축 출력 pulse 수)
pulse (0x4, 10000);     // zP= 10000 (Z 축 출력 pulse 수)
compp(0x2, 15000);      // yCP+ = 15000 (Y 축 CMP+)
lp (0x6, 0);            // LP= 0 (논리위치 Counter= 0)
syncmode (0x2, 0x2001, 0x0000); // Y 축 동기 동작 mode
// 기동 요인: P≥C+, 타축기동: Z
// 자축동작: 없음
syncmode (0x4, 0x0000, 0x0001); // Z 축 동기 동작 mode
// 자축동작: +방향 정량 pulse drive
command (0x2, 0 x20);   // Y 축 + 정량 pulse drive 시작
wait (0x6); // Y, Z 축 종료 대기
}

```

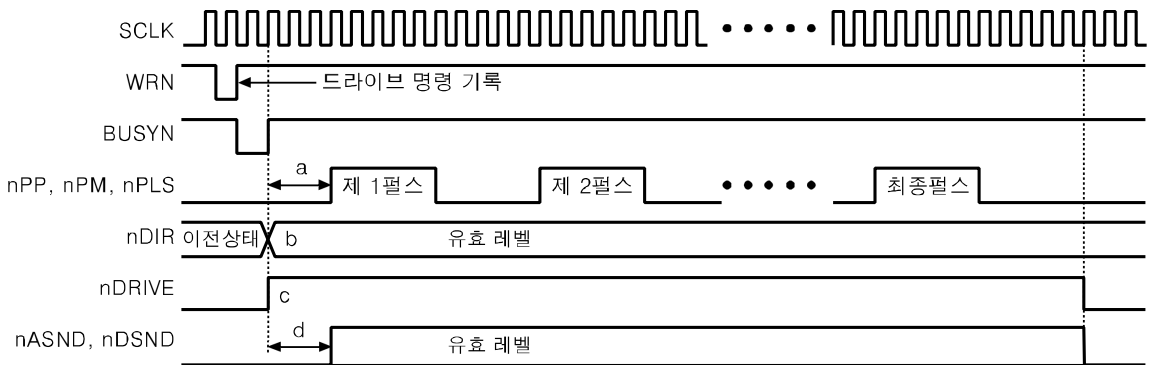
## 15 입/출력 신호 타이밍

### 15.1 파워-온 타이밍



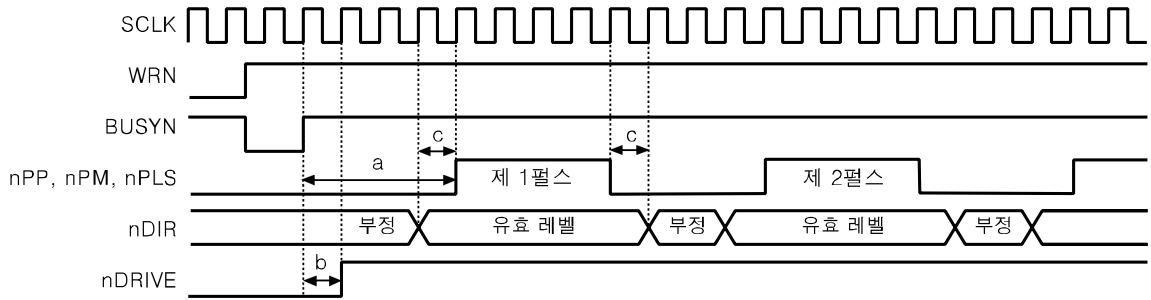
- a. Reset 입력 신호 RESETN 은 CLK 입력 후, CLK×4 사이클 이상 Low level 이 필요합니다.
- b. 전원 투입 시 출력 신호는 RESETN 이 Low level 이며, CLK 가 입력이 최대 CLK×4 사이클 후에 Level 이 확정됩니다.
- c. SCLK 는 RESETN 이 High level 후, 최대 CLK×2 사이클 후에 출력됩니다.
- d. BUSYN 은 RESETN 이 High level 후 최대 CLK×8 사이클의 후에 확정됩니다. 이사에는 모션 컨트롤 IC 를 Read/Write 는 할 수 없습니다.

### 15.2 Drive 시작/종료 시



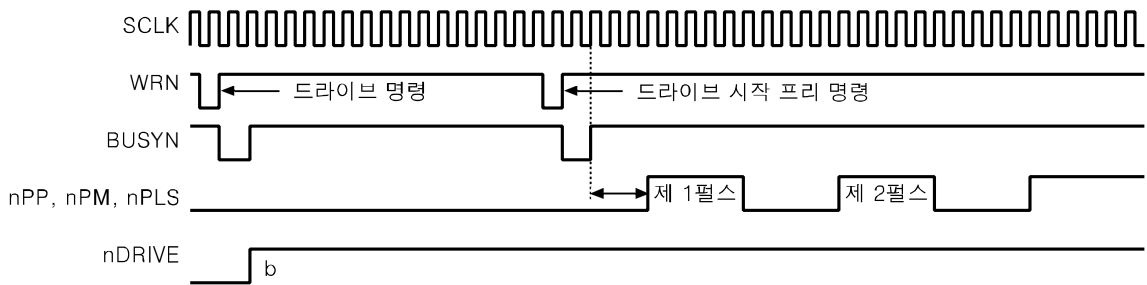
- a. 그림의 drive pulse (nPP, nPM, nPLS)는 정 pulse 의 경우를 나타내고 있습니다. BUSYN 의 ↑에서부터 SCLK3 사이클 후 제 1pulse 가 출력됩니다.
- b. Drive 출력 Pulse 방식을 1pulse 방식으로 설정했을 때의 nDIR(방향) 신호는 BUSYN 의 ↑에서 유효 Level 로 변화하여 Drive 종료 후에도 다음의 Drive 명령이 기입해질 때 까지 그 Level 을 유지합니다. 다만, 보간 Drive 시에는 적용되지 않습니다.
- c. nDRIVE 는 BUSYN 의 ↑에서 High level 이 되어 최종 Pulse 의 Low 기간 후에 Low level 로 돌아옵니다.
- d. nASND, nDSND 는 BUSYN 의 ↑에서 SCLK3 사이클 후에 유효 level 이 되어 nDRIVE 의 최종 Pulse 의 Low 시작과 같이 Low level 로 됩니다.

### 15.3 보간 Drive 시



- a. 보간 Drive 시의 Drive pulse (nPP, nPM, nPLS)는 BUSYN 의 ↑에서 SCLK4 사이클 후에 제 1pulse 가 출력됩니다.
- b. nDRIVE 는 BUSYN 의 ↑에서 SCLK1 사이클 후에 High level 이 됩니다.
- c. Drive 출력 Pulse 방식을 1pulse 방식으로 설정했을 때, nDIR(방향) 신호는 보간 Drive 시, Drive pulse 의 High level 폭과 그 전후 SCLK 1 사이클의 사이, 유효 Level 이 됩니다. (Drive pulse: 정논리 pulse 일 때)

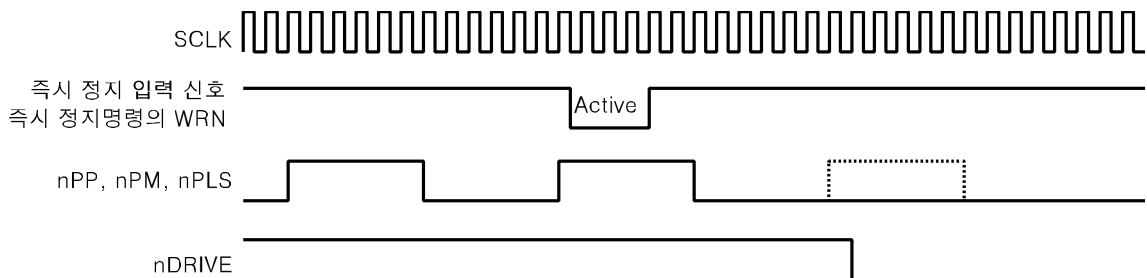
### 15.4 Drive 시작 프리



- a. 각 축의 Drive pulse(nPP, nPM, nPLS)는 Drive 시작 프리 명령 기입의 BUSYN 의 ↑에서부터 SCLK3 사이클 후 동시에 제 1Pulse 가 출력됩니다.
- b. nDRIVE 는 각 축의 Drive 명령 기입의 BUSYN 의 ↑에서 각각 High level 이 됩니다.

### 15.5 Drive 즉시 정지

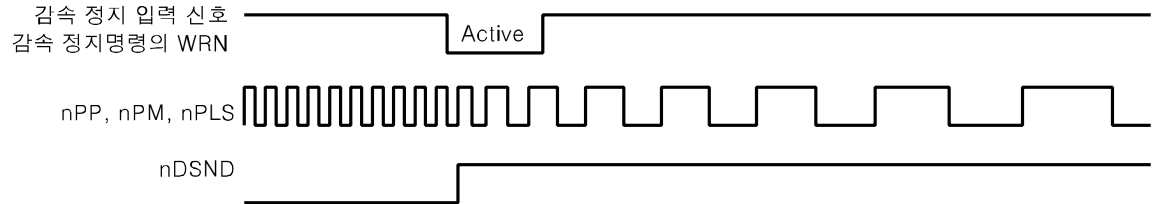
즉시 정지 입력 신호와 즉시 정지 명령의 동작 타이밍입니다. 즉시 정지 입력 신호는 EMG, nLMTP/M(즉시 정지 Mode 로 설정 시), nALARM 입니다. 즉시 정지 입력 신호가 Active level 또는 즉시 정지 명령이 기입되면, 현재 출력중의 Drive pulse 를 출력한 후에 Pulse 출력을 정지합니다.



즉시 정지 입력 신호는 입력 신호 필터가 무효인 경우에도 CLK2 사이클 이상의 Pulse 폭이 필요합니다. 입력 신호 필터를 유효로 하면 필터의 통과시간에 의해 입력 신호는 지연됩니다.

## 15.6 Drive 감속 정지

감속 정지 입력 신호와 감속 정지 명령의 동작 타이밍 입니다. 감속정지 입력신호는 nIN3~0, nLMT+/- (감속 정지 모드로 설정 시) 입니다. 감속 정지 입력 신호가 Active level 또는 감속 정지 명령이 기입되면 현재 출력중의 Drive pulse 를 출력한 후에 감속 정지합니다.



입력 신호 필터를 유효로 하면 필터의 통과 시간에 의해 입력 신호는 지연됩니다.





## 16 사양 정리

- 제어 축: 4 축
- CPU Data bus 길이: 16/8 bit 선택 가능

### (1) 보간 기능

- 2/3 축 직선 보간
  - 보간 범위: 각 축  $-2,147,483,646 \sim +2,147,483,646$
  - 보간 속도: 1pps ~ 4Mpps
  - 보간 위치 정밀도:  $\pm 0.5$  LSB 이하 (전 보간 범위 내에서)
- 원호 보간
  - 보간 범위: 각 축  $-2,147,483,646 \sim +2,147,483,646$
  - 보간 속도: 1pps ~ 4Mpps
  - 보간 위치 정밀도:  $\pm 1$  LSB 이하 (전 보간 범위 내에서)
- 2/3 축 Bit pattern 보간
  - 보간 속도: 1pps ~ 4Mpps (단지 CPU data setup 시간에 의존)
- 그 밖의 보간 기능
  - 임의의 축 선택 가능
  - 선속 일정
  - 연속 보간
  - 보간 Step 전송 (Command, 외부 신호)

### (2) 각 축 공통 사양

- Driver pulse 출력 (CLK=16MHz 시)
  - 출력 회로 범위: 1pps ~ 4Mpps
  - 출력 속도 정밀도:  $\pm 0.1\%$  이하 (설정값에 대하여)
  - 속도 배율: 1 ~ 500
  - S 자용가가속도:  $954 \sim 62.5 \times 10^6$  pps/sec<sup>2</sup> (배율=1 일 때)  
(가속도/감속도의 증가율):  $477 \times 10^3 \sim 31.25 \times 10^6$  pps/sec<sup>2</sup> (배율=500 일 때)
  - 가/감속도:  $125 \sim 1 \times 10^6$  pps/sec (배율=1 일 때)  
 $62.5 \times 10^3 \sim 500 \times 10^6$  pps/sec (배율=500 일 때)
  - 초속도: 1 ~ 8,000pps (배율=1 일 때)  
 $500 \sim 4 \times 10^6$  pps (배율=500 일 때)
  - Drive 속도: 1 ~ 8,000pps (배율=1 일 때)  
 $500 \sim 4 \times 10^6$  pps (배율=500 일 때)
  - 출력 Pulse 수: 0 ~ 4,294,967,295 (정량 Pulse drive)

- 속도 Curve 정속/대칭•비대칭 직선 가감속/대칭•비대칭 포물선 S자 가감속 Drive
- 정량 Pulse drive 의 감속 Mode 자동 감속 (비대칭 직선 가감속 가능) / Manual 감속
- Drive 중의 출력 Pulse, Drive 속도의 변경 가능
- 직선 가감속 정량 Pulse drive 의 삼각 방지, S자 가감 속도 정량 Pulse drive 의 삼각 방지 기능
- 독립 2 pulse / 1 pulse•방향 방식 선택 가능
- Drive pulse 의 논리 Level 선택 가능, 출력 단자 변경 가능
- Encoder 입력 Pulse
  - 2 상 Pulse/up down pulse 입력 선택 가능
  - 2 상 Pulse 1, 2, 4 체배 선택 가능
- 위치 Counter
  - 논리위치 Counter (출력 pulse 용) count 범위: -2,147,483,648 ~ +2,147,483,647
  - 실제위치 Counter (입력 pulse 용) count 범위: -2,147,483,648 ~ +2,147,483,647
  - 상시 write, read 가능
  - 가변 링 Counter 기능, 실제위치 Counter 의 증감 반전 기능, IN2 신호에 의한 실제위치 Counter clear 기능, 항상 쓰기 및 읽기 가능
- 비교 register
  - COMP+ register 위치 비교 범위: -2,147,483,648 ~ +2,147,483,647
  - COMP- register 위치 비교 범위: -2,147,483,648 ~ +2,147,483,647
  - 위치 Counter 와의 대소를 Status 출력 및 신호 출력
  - Software limit 로서 동작 가능
- 자동 원점 복귀
  - Step1(고속 원점 근접 서치) → Step2(저속 원점 서치) → Step3(저속 Encoder Z 상 서치) → Step4(고속 Offset 이동)를 차례대로 자동 실행  
각 Step 의 유효/무효, 검출 방향 선택 가능
- 동기 동작
  - 기동 요인  
위치 Counter ≥ COMP+ 변화, 위치 Counter < COMP+ 변화,  
위치 Counter < COMP- 변화, 위치 ≥ COMP- 변화, Drive 시작, Drive 종료, IN3 신호  
↑, IN3 신호 ↓, LP 읽기 명령, 기동 명령, 각 Step 의 유효/무효, 검출 방향 선택  
가능
  - 동작  
+/- 정량 Pulse drive 시작, +/- 연속 Pulse drive 시작, Drive 감속 정지, Drive 즉시

정지, 위치 Counter 값 저장, 위치 Counter set, 출력 Pulse 수 Set, Drive 속도 set, 인터럽트 발생 자축의 요인으로 임의의 타축 동작 기동 가능

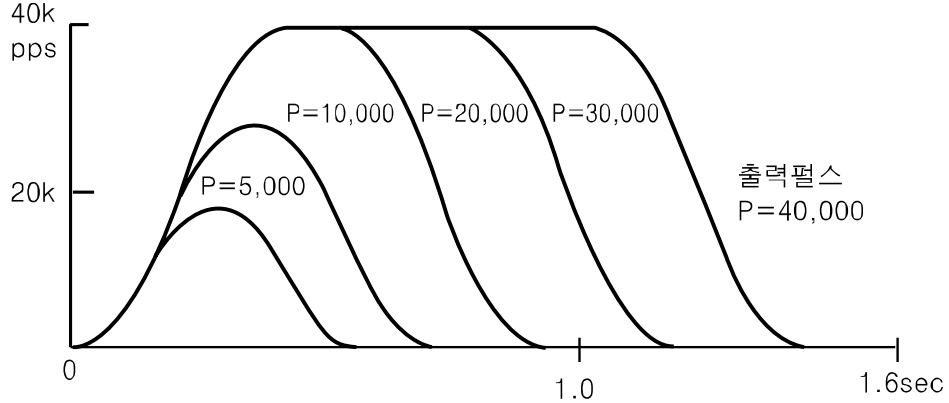
- Interrupt 기능 (보간 제외)
  - Interrupt 발생 요인
    - 1 drive pulse 출력
    - 위치 counter  $\geq$  COMP- 변할 때
    - 위치 counter  $\geq$  COMP+ 변할 때
    - 위치 counter  $<$  COMP- 변할 때
    - 위치 counter  $<$  COMP+ 변할 때
    - 가감속 Drive 중의 정속 시작 시
    - 가감속 Drive 중의 정속 완료 시
    - Drive 종료 때
    - 자동 원점 복귀 종료
    - 동기 동작
  - 어느 요인에 대해서도 유효/무효 선택 가능
- 외부 신호에 의한 Drive 조작
  - EXP+, EXP- 신호에 의한 +, -방향의 정량/연속 Pulse drive 가 가능
  - 2 상 Encoder 신호 Mode (Encoder 입력) drive 가능
- 외부 감속정지/즉시 정지신호
  - IN0~3 각 축 4 점
  - 신호의 유효/무효 및 논리 Level 선택 가능, 범용 입력으로 사용 가능
- 서보 모터용 입력 신호
  - ALARM (alarm), INPOS (위치 결정 완료) 신호의 유효/무효 및 논리 Level 선택 가능
- 범용 출력 신호
  - OUT4~7 각 축 4 점 (Drive 상태 출력 신호와 단자 공용)
- Drive 상태 신호 출력
  - ASND(가속중), DSND(감속중), CMPP(위치 $\geq$ COMP+), CMPM(위치 $<$ COMP-).
  - Drive 상태는 Status register 에서도 읽기 가능
- Overrun limit 신호 입력
  - +방향, -방향 각 1 점, 논리 Level 선택 가능. Active 시, 즉시 정지/감속 정지 선택 가능
- 긴급 정지 신호 입력
  - EMG 1 점, Low level 로 전축의 Drive pulse 를 즉시 정지

- 적분형 필터 내장
  - 각 입력 신호의 입력단에 적분 필터를 장비. 통과 시간(8 종류) 선택 가능
- 전기적 특성
  - 전원 전압: PC 내부 전원 사용 (5VDC)
  - 외부 전원전압: 12-24VDC
  - 동작 온도 범위: 0 ~ +45°C
  - 입력 클럭: 16.000 MHz (표준)
- 그 밖의 보간 기능
  - 임의의 축 선택 가능
  - 선속 일정
  - 연속 보간
  - 보간 Step 전송 (Command, 외부 신호)

## 17 부록: 가감 속도 Drive의 속도 프로파일

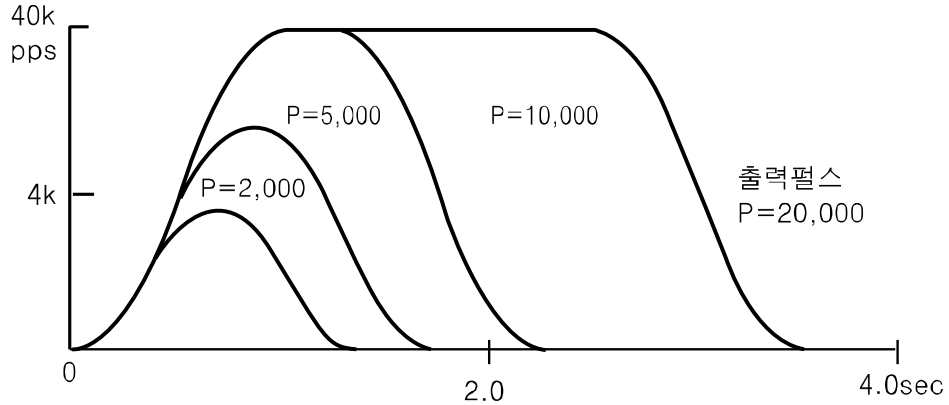
아래와 같은 Parameter 를 설정했을 때 출력되는 Drive pulse 의 속도 커브를 표시합니다.

### (1) 40kpps 대칭 S자 가감속



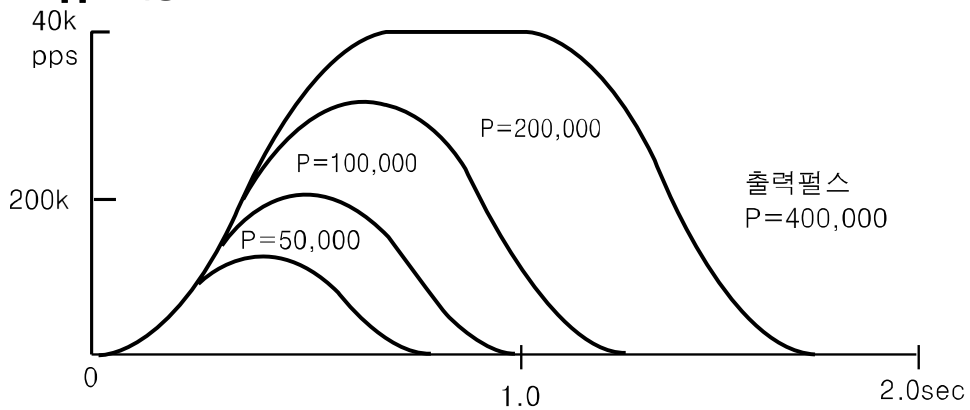
R=800000 (배율: 10)  
 K=700 (A=8000), SV=10, V=4000, A0=0  
 WR3/D2, 1, 0:1, 0, 0 자동 감속 Mode  
 가속도 증가율=893kpps/sec<sup>2</sup>  
 기동속도=100pps  
 Drive 속도=40kpps

### (2) 8,000pps 대칭 S자 가감속



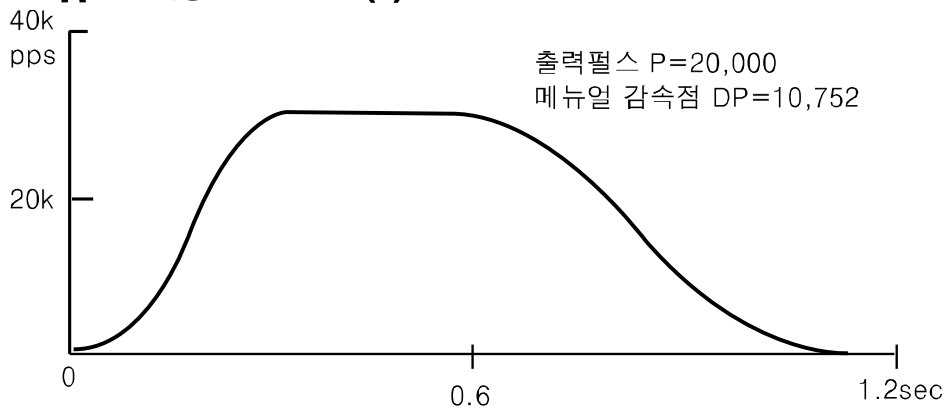
R=8000000 (배율: 1)  
 K=2000 (A=8000), SV=10,  
 V=8000, A0=0  
 WR3/D2, 1, 0:1, 0, 0 자동 감속 Mode  
 가속도 증가율=31kpps/sec<sup>2</sup>  
 기동속도=10pps  
 Drive 속도=8000kpps

**(3) 400pps 대칭 S 자 가감속**



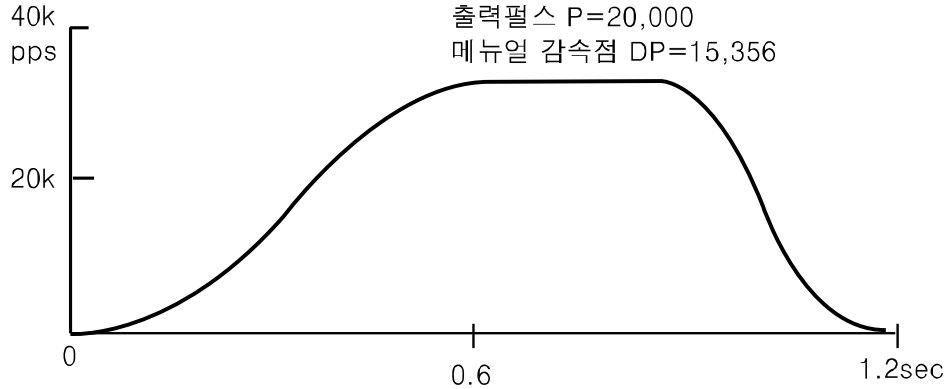
R=80000 (배율: 100)  
 K=2000 (A=8000), SV=10,  
 V=4000, A0=0  
 WR3/D2, 1, 0:1, 0, 0 자동 감속 Mode  
 가속도 증가율=3.13Mpps/sec<sup>2</sup>  
 기동속도=1000pps  
 Drive 속도=400kpps

**(4) 40kpps 비대칭 S 자 가감속(1)**



R=800000 (배율: 10)  
 K=500, L=2000 (A=D=8000),  
 SV=10, V=3000, A0=0  
 WR3/D2, 1, 0:1, 1, 1 매뉴얼 감속 Mode  
 가속도 증가율=1.25Mpps/sec<sup>2</sup>  
 감속도 증가율=0.31Mpps/sec<sup>2</sup>  
 기동속도=100pps  
 Drive 속도=30kpps

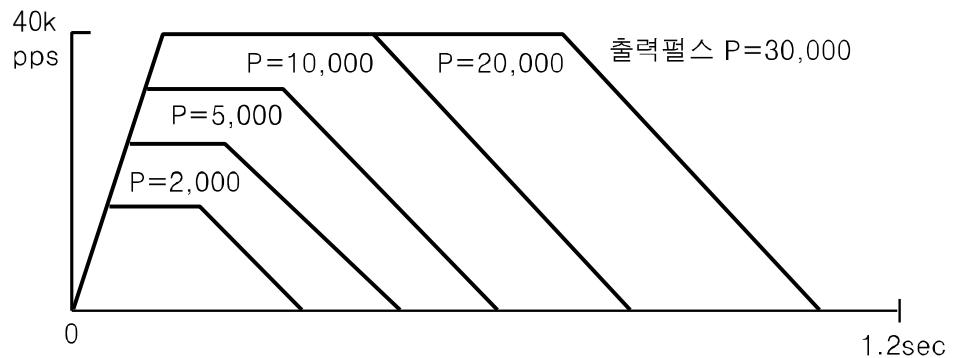
**(5) 40kpps 비대칭 S 자 가감속 (2)**



R=800000 (배율: 10)  
 K=2000, L=500 (A=D=8000),  
 SV=10, V=3000, A0=0  
 WR3/D2, 1, 0:1, 1, 1 매뉴얼 감속 Mode  
 가속도 증가율=0.31Mpps/sec<sup>2</sup>  
 감속도 증가율=1.25Mpps/sec<sup>2</sup>  
 기동속도=100pps  
 Drive 속도=30kpps

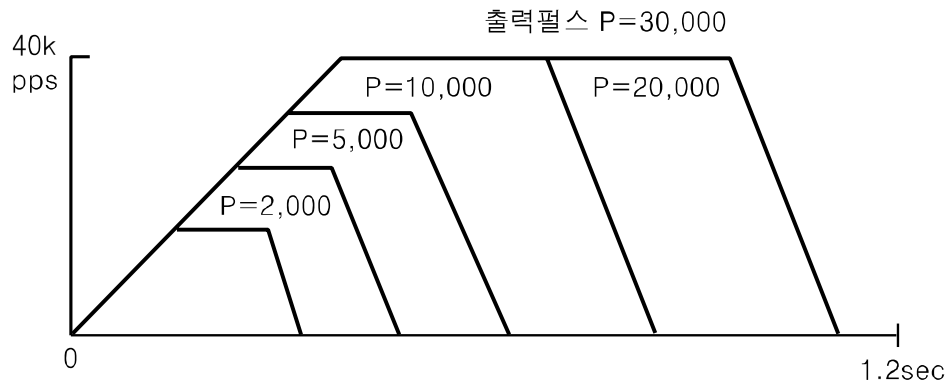
**(6) 40kpps 비대칭 사다리꼴 가감속**

1) 가감속도 비율 4:1



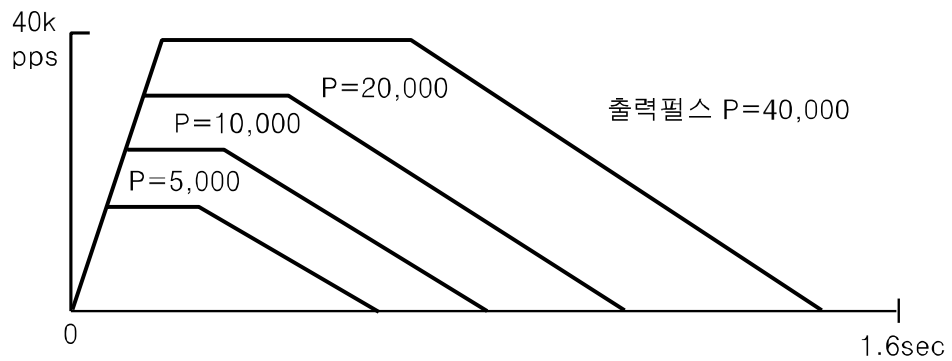
R=800000 (배율: 10), A=400  
 D=100, SV=40, V=4000, A0=0  
 WR3/D2, 1, 0:0, 1, 0 자동 감속 Mode  
 60H/WR6/D3:1 삼각 방지 ON  
 가속도=500kpps/sec  
 감속도 =125kpps/sec  
 기동속도=400pps  
 Drive 속도=40kpps

2) b. 가/감속도 비율 1:4



R=800000 (배율: 10), A=100  
 D=400, SV=40, V=4000, A0=0  
 WR3/D2, 1, 0:0, 1, 0 자동 감속 Mode  
 60H/WR6/D3:1 삼각 방지 ON  
 가속도=125kpps/sec  
 감속도 =500kpps/sec  
 기동속도=400pps  
 Drive 속도=40kpps

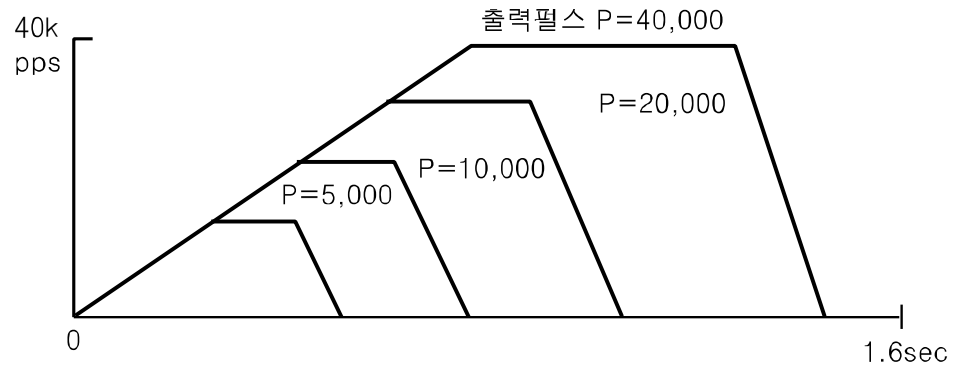
3) 가/감속도 비율 10:1



R=800000 (배율: 10), A=400, D=40,  
 SV=50, V=4000, A0=0  
 WR3/D2, 1, 0:0, 1, 0 자동 감속 Mode  
 60H/WR6/D3:1 삼각 방지 ON  
 가속도=500kpps/sec  
 감속도 =50kpps/sec  
 기동속도=500pps  
 Drive 속도=40kpps



4) d. 가감속도 비율 1:10



R=800000 (배율: 10), A=40, D=400,  
 SV=50, V=4000, A0=0  
 WR3/D2, 1, 0:0, 1, 0 자동 감속 Mode  
 60H/WR6/D3:1 삼각 방지 ON  
 가속도=50kpps/sec  
 감속도 =500kpps/sec  
 기동속도=500pps  
 Drive 속도=40kpps

# Autonics

Sensors & Controllers

 고객 서비스 센터  
1588-2333

[www.autonics.co.kr](http://www.autonics.co.kr)

- 본 사 : 부산광역시 해운대구 반송로 513번길 18(석대동)
- 서울사무소 : 경기도 부천시 원미구 평천로 655 부천테크노파크 402동 3/4층(약대동)
- 대구사무소 : 대구광역시 북구 유통단지로 8길 66 태영빌딩 3층(산격동)
- A/S 센터 : 부산/경남/대구/경북/광주/전라/제주 지역 : 본사  
서울/경기/인천/충청/강원 지역 : 서울사무소

Distributor

## ■ 주요생산품목

· 포토센서 · 광화이버센서 · 도어센서 · 도어사이드센서 · 에리어센서 · 근접센서 · 압력센서 · 로터리 엔코더 · 온도조절기 · 온/습도 센서 · SSR/전력조정기 · 카운터 · 타이머 · 판넬메타 · 타코/스피드/펄스메타 · 커넥티/소켓 · 디스플레이 유닛 · 센서 컨트롤러 · 스위칭 모드 파워 서플라이 · 제어용 스위치/램프/부저 · I/O 단자대/케이블 · 스테핑 모터&드라이버&컨트롤러 · 그래픽/로직 패널 · 필드 네트워크 기기 · 레이저 마킹 시스템(Fiber, CO<sub>2</sub>, Nd: YAG) · 레이저 웰딩/커팅 시스템