

Hardware Chip User Manual

# Motor Control Chip

**CAMC-FS<sub>2.1</sub>**

AJINEXTEK The logo for AJINEXTEK AXT, featuring the word "AJINEXTEK" in a blue sans-serif font followed by "AXT" in a larger, stylized blue font where the letters are interconnected.

### *Product Information*

Full information about other AJINEXTEK products is available by visiting our Web Site at:

[www.ajinextek.com](http://www.ajinextek.com)

### *Useful Contact Information*

#### **Customer Support Seoul**

Tel : 82-31-389-1580~2 Fax: 82-31-389-1583

E-mail : [marketing@ajinextek.com](mailto:marketing@ajinextek.com)

#### **Customer Support Daegu**

Tel : 82-53-593-3700 Fax: 82-53-593-3703

E-mail : [support@ajinextek.com](mailto:support@ajinextek.com)

#### **Customer Support Cheonan**

Tel : 82-41-555-9771~2 Fax: 82-41-555-9773

E-mail : [ljh001024@ajinextek.com](mailto:ljh001024@ajinextek.com)



AJINEXTEK's sales team is always available to assist you in making your decision the final choice of boards or systems is solely and wholly theresponsibility of the buyer. AJINEXTEK's entire liability in respect of the board or systems is as set out in AJINEXTEK's standard terms and conditions of sale

## Contents

1. 개요	1
2. 사양	5
2.1. 사양	5
2.2. Block Diagram	6
2.3. 입출력 핀 배치도	7
2.4. 핀의 기능 설명	8
3. 기능 개요	15
3.1. 드라이브 기능	15
3.1.1. PRESET PULSE DRIVE	15
3.1.2. CONTINUOUS DRIVE	16
3.1.3. SIGNAL SEARCH-1 DRIVE	16
3.1.4. SIGNAL SEARCH-2 DRIVE	17
3.1.5. HOME SEARCH DRIVE	17
3.1.6. SENSOR POSITIONING DRIVE	18
3.1.7. MPG DRIVE	20
3.2. 가감속 모드 설정 기능	22
3.2.1. 대칭 포물선 가감속 모드	22
3.2.2. 비대칭 포물선 가감속 모드	23
3.2.3. 대칭 직선 가감속 모드	24
3.2.4. 비대칭 직선 가감속 모드	25
3.3. 감속 개시 포인트 검출 방식 선택 기능	26
3.4. 서보 모터 인터페이스 기능	26
3.5. 다축 동기 맞춤 기능	26
3.6. 범용 입출력 기능	27
3.7. 기계계 외부 입력 제어 기능	27
3.8. 펄스출력 방식 전환 기능	27
3.9. 내부 위치 링 카운터(Internal Ring Counter) 관리 기능	29
3.10. 외부 위치 링 카운터(External Ring Counter) 관리 기능	30
3.11. 내부 위치 카운터 비교 기능	30
3.12. 외부 위치 카운터 비교 기능	31
3.13. 내부/외부 위치 Scale 카운트 기능	31
3.14. 외부 위치 카운터 Clear 기능	32
3.15. 외부 위치 카운터 방향 반전 기능	32

3.16. 편차량 산출 기능 .....	32
3.17. 속도 데이터/편차량 데이터 모니터 기능.....	33
3.18. 인터럽트 발생 기능 .....	33
3.19. 설정 데이터 에러 판정 기능.....	33
3.20. 탈조(Off-Range) 검출 기능.....	33
3.21. 실거리 이동 연산 기능.....	33
3.22. PWM 출력 기능.....	34
3.23. 외부 센서 필터 대역폭(Bandwidth) 설정 기능.....	34
3.24. 스크립트(Script) 기능.....	34
3.25. 갈무리(Caption)기능 .....	35
4. Address Map .....	36
5. Port <b>설명</b> .....	38
5.1. DATA1 WRITE PORT.....	38
5.2. DATA2 WRITE PORT.....	38
5.3. DATA3 WRITE PORT.....	39
5.4. DATA4 WRITE PORT.....	39
5.5. COMMAND WRITE PORT.....	39
5.6. DATA1 READ PORT .....	40
5.7. DATA2 READ PORT .....	40
5.8. DATA3 READ PORT .....	40
5.9. DATA4 READ PORT .....	41
6. COMMAND <b>설명</b> .....	42
6.1. COMMAND 일람표.....	42
6.2. COMMAND 기능 설명 .....	66
6.3. COMMAND 실행 방법 .....	110
7. TIMING .....	113
7.1. DATA BUS TIMING .....	113
7.2. 입력 신호 TIMING .....	115
7.3. 출력 신호 TIMING 1 .....	117
7.4. 출력 신호 TIMING 2 .....	121
8. <b>보충 사항</b> .....	126
8.1. ICLK 입력 주파수의 선택.....	126
8.2. 각종 PARAMETER의 변경.....	127

**8.3. DATA의 ERROR 판정..... 127**

**8.4. S字 가감속시 주의사항 ..... 129**

**8.5. UP, CONST, DOWN 신호 ..... 130**

**8.6. 가감속 PULSE의 산출..... 132**

**8.7. 미사용(未使用) 입력 단자의 처리..... 132**

**8.8. LIMIT 신호의 검출..... 132**

**8.9. 엔코더 2相 신호의 CHATTERING..... 133**

**8.10. 엔코더 2相신호 COUNT POINT..... 134**

**9. 외형 치수 ..... 135**

**10. 부록 ..... 136**

**10.1. Address Map ..... 136**

**10.2. PORT 설명..... 137**

**10.3. COMMAND 일람표(Compact, register map) ..... 140**

**10.4. COMMAND 기능 부연 설명 ..... 165**

*Revision History*

Manual	PCB	Comments
Rev. 4.0 issue 1.0		2003.09.01



# 1. 개요

CAMC-FS는 CAMC-5M의 성능 상위의 **Single Motor Control LSI**이다.

이 칩은 CPU 버스 인터페이스를 통하여 전달된 Command를 참조하여 Stepping 및 Servo Motor등을 정밀제어 할 수 있다. 따라서 이 칩을 이용하여 CPU의 부하를 감소시키면서 연속적인 모션제어를 구현할 수 있으며, 원점검출, 속도 오버라이드 및 이동량 오버라이드를 직선 또는 S-curve 가감속 프로파일로 구동할 수 있다.

CAMC-FS는 모터 구동 중 발생하는 특정 이벤트 시간에 실행할 Command를 미리 저장할 수 있는 Script/Caption 기능이 있다. 이 기능을 이용하여 모터 구동 중의 특정 동작을 software로 구현하기가 용이하다. 이에 사용되는 모든 파라미터들은 소프트웨어적으로 설정할 수 있으며, 펄스 출력 중에도 변경이 가능하다. CAMC-FS의 자세한 특징은 다음과 같다.

## ▶ S-Curve 가감속 제어

S-curve 가속/감속 제어가 가능하다.

가감속율(기울기)은 가속기울기와 감속기울기를 독립적으로 설정 가능하다.

## ▶ 최대 4 Mpps 펄스 출력

최대 펄스출력속도는 입력 클럭 16.384MHz(사용권장 주파수)를 기준으로 약 4 Mpps(pulse per second)이다.

## ▶ 8/16 bit CPU interface

8-bit 및 16-bit Data bus를 제공한다.

## ▶ 7가지 드라이브 모드 제공

연속 드라이브, 신호 및 원점 검출 드라이브, 펄스 수 설정 드라이브, 센서 위치 드라이브, MPG 드라이브를 지원한다.

## ▶ 엔코더 노이즈 제거용 디지털 필터 내장

## ▶ 외부 펄스 카운트 기능

단상 및 2상 1체배, 2체배, 4체배 엔코더에서 발생하는 신호를 칩 내부에서 카운트하고 저장되면 펄스 개수와 편차량 등을 읽어 볼 수 있다.

## ▶ 설정 Data Error 판정 기능

설정 레지스터에 대응하는 검사기능이 있어, 설정데이터에 에러가 있거나 이전 드라이브 실행 중인

경우는 급정지하고 드라이브를 시작하면 바로 종료한다.

▶ 인터럽트 발생기능

CAMC-FS에는 인터럽트 발생에 필요한 32개의 인터럽트 소스와 각각의 인터럽트를 가능 또는 불가능하게 만드는 32개의 인터럽트 마스크가 있다.

▶ 다축 동기 맞춤 기능

여러 개의 CAMC-FS 칩을 구동하는 경우를 위해 SYNC(Synchronization) 입력 핀이 있다. 이 입력 핀이 활성화 되기 전까지는 드라이브를 시작하더라도 펄스를 출력하지 않는다. 따라서, 이 핀을 이용하여 여러 개의 CAMC-FS 칩의 펄스 출력을 동시에 시작할 수 있다.

▶ 다양한 펄스 출력 기능

CAMC-FS 칩에는 접속한 모터 드라이브의 사양에 맞추어 펄스 출력방식을 선택할 수 있도록 8 종류의 출력 방식이 갖추어져 있다. 펄스 출력 형태는 Software적으로 설정 가능하다.

▶ Limit 정지 기능

CAMC-FS 칩에는 기계의 over-run limit 신호로서 정/역 방향 각각에 급정지용/감속정지용 limit 신호 입력 단자가 있으며 입력 레벨은 개별 지정 가능하다.

▶ 탈조 검출 기능

CAMC-FS에는 내부 카운터 값과 외부 카운터 값의 차이 값의 범위를 탈조 검출 데이터 레지스터에 설정할 수 있다.

내부 카운터 값과 외부 카운터 값의 차가 탈조 검출 데이터 값보다 커지면 급정지, 감속정지를 한다.

▶ 실거리 이동 연산 기능

지정 펄스 수 드라이브 실행 시 실제 이동거리로 연산을 수행하여 즉, 외부 입력 펄스로 연산을 수행하여 실행 오차를 감소시킬 수 있다.

▶ PWM 출력 기능

드라이브 구동 중 속도 값에 해당하는 PWM(Pulse Width Modulation) 신호를 출력하는 핀이 제공된다.

▶ 스크립트(Script) 처리 기능

CAMC-FS가 구동 중에 실행할 96개 명령 코드를 예약하여 실행할 수 있다.

▶ 상태 갈무리(Caption) 기능

구동 중인 모터 칩의 상태를 읽을 수 있는 96개 명령어를 예약하여 확인 가능하다.

▶ 센서 신호 및 제어 신호의 설정 기능

외부에서 입력되는 센서 신호 및 제어 신호를 Enable/Disable 할 수 있다. 센서 및 제어 신호의 기능의 Enable/Disable 설정은 Register화하여 사용자가 Software적으로 지정 가능하다.

▶ 감속 정지 모드 기능 강화

감속 정지 모드의 설정에 따라 드라이브 구동 중 감속 정지 명령을 실행할 경우 감속한 후 드라이브를 정지할 수 있고, 감속한 후 등속도로 드라이브 구동을 유지할 수 있다.

▶ 엔코드 입력 내장 (A, B, Z 상 입력 처리)

A, B 상 입력으로 외부 카운터가 동작하여 기계계열의 현재 위치를 알 수 있고 Z상을 이용하여 외부 위치 카운터를 timing clear하는 기능이 있다.

이 때 Z 상 입력 신호는 4개의 범용 입력 신호 중 선택 가능하다.

▶ 입력 필터 대역폭 설정

외부에서 입력되는 limit 센서 신호와 입력 처리 센서 신호에 입력 노이즈를 제거하는 필터 기능을 가지고 있다.

필터 주기는 사용자가  $Tclk \sim 255 * Tclk$  범위에서 지정 가능하다. (Tclk: 기준 clock의 주기)

▶ 내부 위치 링 카운트 기능

내부 위치 링 카운트 기능이 있으며, M\_DATA, P\_DATA를 설정할 수 있는 2개의 32-bit 레지스터가 있다. 위치 링 카운트 기능이란 설정한 범위 내에서 무한대로 카운트하는 기능이다.

▶ 외부 위치 링 카운트 기능

외부 위치 링 카운트 기능이 있으며, M\_DATA, P\_DATA를 설정할 수 있는 2개의 32-bit 레지스터가 있다. 위치 링 카운트 기능이란 설정한 범위 내에서 무한대로 카운트하는 기능이다.

▶ 위치 카운터의 Scale 기능

Scale 카운트 기능이란 펄스가 지정한 scale값의 배수 크기만큼 카운트 되면 내부/외부 카운터 값이 +1 증가하는 기능이다. 즉, 설정한 scale값의 배수 배로 카운트할 수 있는 기능이다. 이 때 Scale 데이터는 8-bit(0~255)이며 내부/외부 카운터용으로 각각 내장되어 있다.

▶ 소프트웨어 리미트 설정 기능

내부/외부 카운터 값과 설정된 비교 데이터에 따라 특정 위치를 소프트웨어적인 리미트로 지정 할 수 있다.

- ▶ 범용 입력 단자 (4개)
  
- ▶ 범용 출력 단자 (8개)
  
- ▶ 범용 신호 IO 입출력 선택 기능(1개)

## 2. 사양

### 2.1. 사양

전원	: +5V ± 5%, +3.3V ± 5%
(40,47,89,90번 핀은 반드시 +3.3V와 연결. 나머지 VDD핀은 +5V 또는 +3.3V와 연결)	
기준 clock	: 16.384 MHz(standard)
제어 축 수	: 1축
속도 설정 범위	: 16-bit (65,535)
가/감속도 설정 범위	: 16-bit (65,535)
속도 배율 설정범위	: 0.002 ~ 250(기준 clock 16.384MHz)
	0.002 배 - 0.00095 ~ 31.25 pps
	1 배 - 0.5 ~ 16,384 pps
	125 배 - 62.5 ~ 2,047,932 pps
	250 배 - 125 ~ 4,095,375 pps
Slow down point 설정 범위	: 32-bit (0 ~ 4,294,967,295)
주요 드라이브 모드	: 지정 펄스 수 드라이브, 연속 드라이브 신호 검출-1 드라이브, 신호 검출-2 드라이브 원점 검출 드라이브, 센서 위치 드라이브, MPG 드라이브
오버라이드 기능	: 구동 중 속도 오버라이드, 구동 중 이동량 오버라이드
가감속도 구동	: 대칭/비대칭 Full S 가감속 구동, 대칭/비대칭 Quasi-S 가감속 구동 대칭/비대칭 직선 가감속 구동, 등속 구동
Servo Interface	: 디지털 펄스 방식
출력 펄스 방식	: 1펄스/2펄스 방식 선택 가능(총 8 종류 지원)
편차량 산출 기능	: 15-bit (-32767 ~ 32,767)
Limit 정지 기능	: ±ELM(Emergency stop Limit), ±SLM(Slow down stop Limit), SSTOP(Slow down Stop), ESTOP(Emergency Stop)
정지 명령어	: SSC(Slow down Stop Command), ESC(Emergency Stop Command)
CPU 인터페이스	: 8/16 bit 선택
범용 입력 단자	: 4 채널 TTL Schmitt Trigger Level
범용 출력 단자	: 8 채널 CMOS Level, 8mA
범용 입출력 선택 단자	: 1 채널
Comparator 기능	: 내부/외부 카운터 각각 32-bit 1개씩 제공
Interrupt 발생 기능	: 인터럽트 source 32개 선택 가능
다축 동기 맞춤 기능	: 외부 sync 입력 신호에 의해 펄스 출력 시작 가능 ENABLE / DISABLE 선택 가능

- 내부,외부 카운터 기능 : 각각 32-bit
- 엔코더 입력 : 단상(UP/DOWN 입력), 2상(1, 2, 4 체배)
- 현재 속도 데이터 Read Port : 13-bit
- 소프트웨어 리미트 기능 : 내부, 외부 설정 기능, Slow Down Stop, Emergency Stop 기능

2.2.Block Diagram

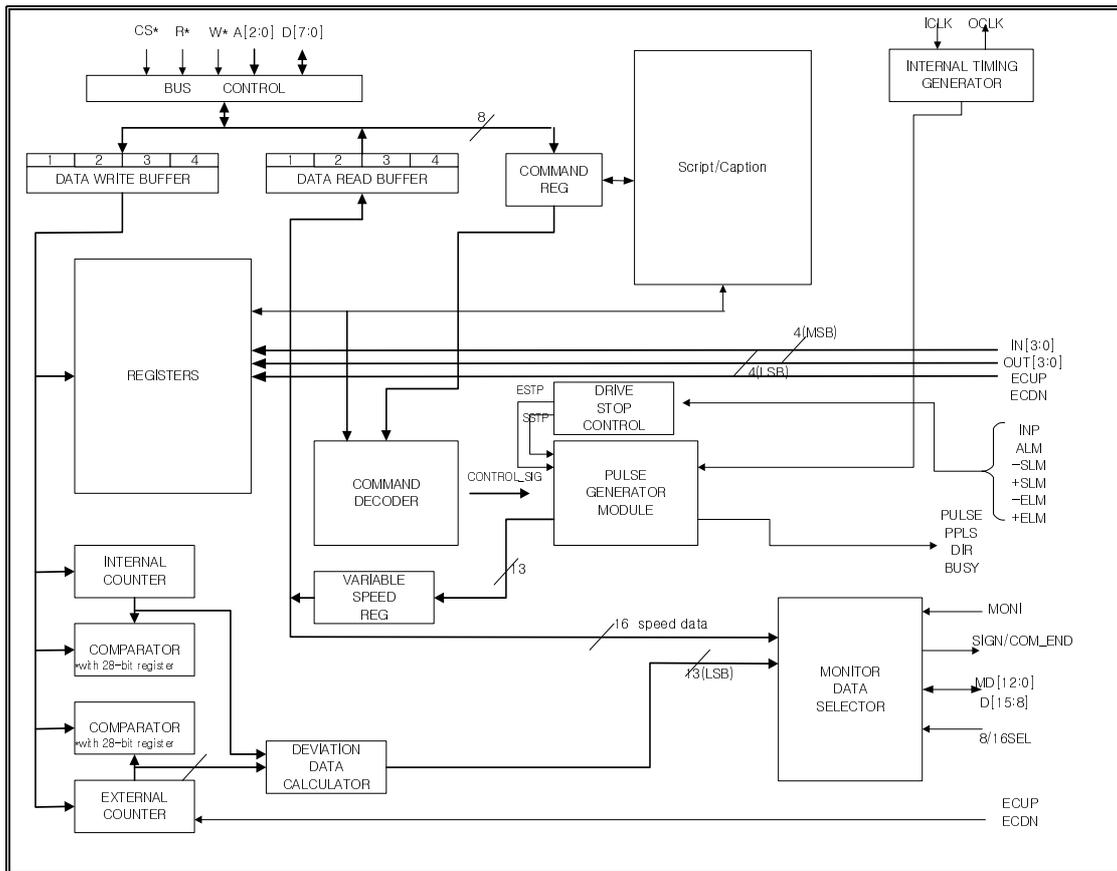


그림 2.1. CAMC - FS 전체 기능 블록도

### 2.3. 입출력 핀 배치도

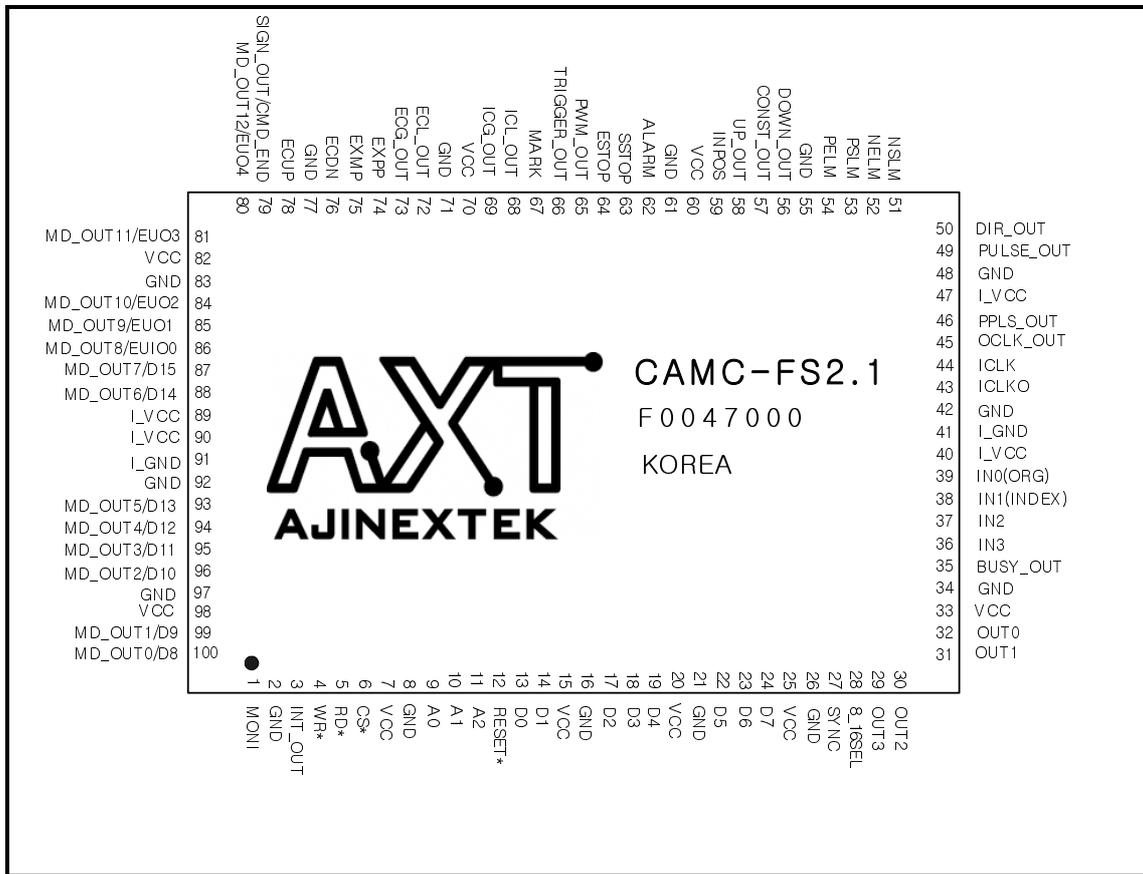


그림 2.2. 입출력 핀 배치도

## 2.4.핀의 기능 설명

▶ D7 ~ D0 (Data Bus) --- 3 STATE 입출력

8 Bit 3 State 양방향 Data Bus. CPU의 Data Bus에 접속.

▶ D15 ~ D8 (Data Bus) --- 3 STATE 입출력

\*(MD7~MD0을 공용으로 사용)

8 Bit 3 State 양방향 Data Bus. CPU의 Data Bus에 접속.

16-Bit Data Access 모드(8\_16SEL = High)에는 Data Bus로 사용, 8-Bit Data Access 모드에서는 MD7~MD0(모니터 데이터)를 출력하는 신호로 사용

▶ A2 ~ A0 (Address) --- 입력

\*(A0는 16bit access시 사용 안 함)

CAMC-FS 칩이 가지고있는 8 포트를 선택한다. CPU의 Address Bus에 접속.

▶ CS\* (Chip Select) --- 입력

CS\*=L로 함으로써 CAMC-FS 칩에 대해 Access가 가능해진다.

▶ RD\* (Read Strobe) --- 입력

내부 버퍼 읽어내기 동작 시 RD\*=L로 한다.

▶ WR\* (Write Strobe) --- 입력

내부 버퍼에 써넣기 동작 시 WR\*=L로 한다.

▶ RESET\* (Reset) --- 입력

RESET\*=L 에 의해 CAMC-FS 칩은 Reset되어 초기상태로 된다.

최소 Tclk\*8이상의 LOW가 유지될 때 유효하다.(Tclk = Main clock의 주기)

▶ ICLK (Input Clock) --- 입력

동작을 위해 기준 클럭을 입력한다. CAMC-FS 칩의 동작 속도는 ICLK입력에 의해 결정된다.

16.384MHz보다 낮은 주파수의 클럭을 사용하였을 경우 CAMC-FS 칩의 정상적인 동작을 보장할 수 있다.

▶ OCLK (Output Clock) --- 출력

기준 입력 클럭의 2분주 신호를 출력한다.

예) 16.384MHz를 ICLK로 사용하였다면 OCLK의 출력은 8.192MHz가 된다.

▶ DIR/PULSE (Direction / Pulse) --- 출력

모터 드라이브의 입력신호로 사용할 출력 신호이다. 드라이브 방향 신호 및 Duty 50%의 드라이브 펄스가 출력된다.

▶ PWM\_OUT(PWM output) --- 출력

속도 데이터 값에 해당하는 PWM 신호를 출력한다. 이 신호는 현재 속도 데이터 만큼의 PWM 신호이므로 DC 모터 제어에 직접 사용 가능하다.

▶ PPLS (P\_Pulse) --- 출력

기준 클럭 입력의 2주기폭(OCLK출력의 1주기폭)만큼 High가 되고 주기는 출력 펄스와 동일한 신호를 출력한다.

▶ BUSY (Busy) --- 출력

드라이브 중(PULSE출력 중) BUSY = H로 되어 현재 칩이 펄스 출력 동작 중임을 나타내는 상태 신호이다.. 동기 맞춤 드라이브 시작을 위해 SYNC신호 대기 중 및 PULSE출력 종료 후의 INP신호 대기 중인 동안에도 BUSY = H로 된다.

▶ UP (Up) --- 출력

가속 드라이브 중에 UP = H로 되어 가속 드라이브 중임을 나타내는 상태 신호이다.

▶ CONST (Constant) --- 출력

정속 드라이브 중에 CONST = H로 되어 현재 등속 드라이브 중임을 나타내는 상태 신호이다.

▶ DOWN (Down) --- 출력

감속 드라이브 중에 DOWN = H로 되어 현재 감속 드라이브 중임을 나타내는 상태 신호이다.

▶ 8\_16SEL (Bus Select) --- 입력

8-bit Data Bus(Low)와 16-bit Data Bus(High) Access를 결정하는 입력 신호이다. 이 신호를 조절함으로써 16-bit CPU와 8-bit CPU 모두 사용하기 간편해졌다. 이 신호가 Low 일때

▶ MONI (Monitor Select) --- 입력

MD12~MD0로 출력할 신호를 선택하는 신호이다. MD12~MD0는 현재 속도 데이터나 편차량 데이터를 출력하는 Monitor Data 신호로서 MONI 입력 값에 따라 출력하는 데이터가 변경된다.

MONI = Low인 경우, SIGN은 편차량 데이터의 부호를, MD12~MD0은 편차량 데이터를 출력한다. 실제 편차량 값은 16-bit크기이며 하위 13-bit만 출력하게 된다. MONI = High인 경우에 SIGN은 속도 데이터의 부호를, MD12~MD0은 속도 데이터를 출력한다. 속도 데이터는 실제 16-bit 크기이며 상위 13-bit만

외부로 출력된다. 16-bit Data Bus Access 모드(8\_16SEL = High)에서는 MD7~MD0까지 데이터 버스로 사용하므로 이와 같은 동작은 8-bit Data Bus Access 모드에서만 이루어진다.

▶ SIGN/ CMD\_END (Monitor Data SIGN/ COMMAND END) --- 출력

Monitor Data의 부호를 출력하거나, 칩 내부에서 명령 실행이 상태를 나타내는 신호이다.

8-bit Data Bus Access 모드(8\_16SEL = Low)에서는 Monitor Data(MD12~MD0)의 부호를 출력한다. Monitor Data가 속도 데이터인 경우는 부호가 양(Low)일 경우는 정방향 회전을 나타내고, 음(High)일 때는 역방향 회전을 나타낸다. Monitor Data가 편차량 데이터인 경우는 이 값의 부호를 나타낸다.(양수=Low, 음수=High) 16-bit Data Bus Access 모드(8\_16SEL = High)에서는 칩 내부 명령을 처리 중일 때는 High를 출력, 명령 실행 프로세스가 완료될 경우 Low를 출력한다.

▶ MD12 ~ MD00/DATA15~DATA8(Monitor Data/Msb Data Bus) --- 출력

모니터 데이터를 출력하거나 16-bit Data Bus를 사용할 경우 상위 8-bit의 Data Bus로 사용하는 신호이다. 8/16SEL = Low (8bit Data Bus Access 모드로 선택)일 경우 MONI 신호에 의해 선택된 데이터(현재 속도 데이터 또는 편차량 데이터)를 출력한다. 출력되는 데이터는 부호를 포함하지 않은 13-bit 이진수이다. MONI = High일 경우 현재 출력 중인 속도 데이터(16-bit)의 상위 13-bit를 출력한다. MONI = Low일 경우 편차량 데이터(16-bit)의 하위 13-bit를 출력한다.

8\_16SEL = High (16bit DATA BUS로 선택)일 경우 DATA15~DATA8(상위 8-bit로 Data Bus 기능을 한다. 이 때 나머지 5bit는 EUO/EUIO의 설정에 따른다.

▶ ESTOP (Emergency Stop) --- 입력

급정지 입력 신호. 드라이브 중에 ESTOP = High 가 입력되면 펄스 출력이 중지 된다.

또 ESTOP 신호에 의해 PULSE출력이 정지된 경우, END STATUS READ PORT(칩이 드라이브 정지하였을 경우 정지한 이유는 나타내는 상태 레지스터) 내의 ESSD BIT(칩이 급정지하였음을 나타내는 상태 bit)가 1로 된다.

▶ SSTOP (Slow Down Stop) --- 입력

감속정지 입력 신호. 드라이브 중에 SSTOP = High가 입력되면 시작속도까지 감속하고 펄스 출력이 정지된다.

또 SSTOP신호에 의해 펄스 출력이 정지된 경우, END STATUS READ PORT내의 SSSD BIT(칩이 감속 정지하였음을 나타내는 상태 bit)가 1로 된다. CAMC-FS는 시작 속도까지 감속한 이후 바로 정지하게 할 수도 있으며, 시작 속도까지 감속한 이후 바로 정지하지 않고 급정지 신호나 명령이 입력될 때까지 등속으로 펄스출력을 유지하게 할 수도 있다..

▶ +ELM (+Emergency Limit) --- 입력

정 회전 방향 LIMIT 입력 신호. 정 회전 방향 Drive중에 Active Level이 입력되면 현재 출력 중인

PULSE가 Not Active로 되고 이후의 PULSE출력을 정지한다. 단, +ELM신호의 Active Level은 User Program 의해 지정 가능하다. 또, +ELM신호에 의해 PULSE출력이 정지된 경우, END STATUS READ PORT내의 급정지 외부센서에 의해 종료 되었음을 나타내는 상태 BIT가 1로 된다.

▶ -ELM (-Emergency Limit) --- 입력

역 회전 방향 LIMIT 입력 신호. 역 회전 방향 Drive중에 Active Level이 입력되면 현재 출력 중인 PULSE가 Not Active로 되고 이후의 PULSE출력을 정지한다.

단, -ELM신호의 Active Level은 User Program에 의해 지정 가능하다.

또, -ELM신호에 의해 PULSE출력이 정지된 경우, END STATUS READ PORT내의 급정지 외부센서에 의해 종료 되었음을 나타내는 상태 BIT가 1로 된다.

▶ +SLM (+Slow Down Limit) --- 입력

정 회전 방향 감속 정지 LIMIT 입력 신호 정 회전 방향 Drive중에 Active Level이 입력되면 Start / Stop Speed까지 감속하고 출력 중인 PULSE가 Not Active로 되면서 이후의 PULSE출력을 정지한다.

단, +SLM신호의 Active Level은 User Program에 의해 지정 가능하다.

또, +SLM신호에 의해 PULSE출력이 정지된 경우, END STATUS READ PORT내의 감속정지 외부센서에 의해 종료 되었음을 나타내는 상태 BIT가 1로 된다.

▶ -SLM (-Slow Down Limit) --- 입력

역 회전 방향 감속 정지 LIMIT 입력 신호 역 회전 방향 Drive중에 Active Level이 입력되면 Start / Stop Speed까지 감속하고 출력 중인 PULSE가 Not Active로 되면서 이후의 PULSE출력을 정지한다.

단, -SLM신호의 Active Level은 User Program에 의해 지정 가능하다.

또, -SLM신호에 의해 PULSE출력이 정지된 경우, END STATUS READ PORT내의 감속정지 외부센서에 의해 종료 되었음을 나타내는 상태 BIT가 1로 된다.

▶ ALM (Alarm) --- 입력

Motor Driver로부터의 ALARM입력 신호. Drive중에 Active Level이 입력되면 현재 출력 중인 PULSE가 Not Active로 되고, 이후의 PULSE출력을 정지한다.

단, ALM신호의 Active Level및 ALM신호 감시 기능의 유효/무효는 User Program에서 지정 가능하다.

또, ALM신호에 의해 PULSE출력이 중단된 경우 END STATUS READ PORT내의 alarm에 의해 종료 되었음을 나타내는 상태 BIT가 1로 된다.

▶ INP(Inposition) --- 입력

Motor Driver로부터의 INPOSITION입력 신호. PULSE출력 종료 후 Active Level이 입력될 때까지 BUSY = H를 유지한다.

INP신호의 Active Level및 INP신호 감시 기능의 유효/무효는 User Program에서 지정 가능하다.

▶ SYNC (Synchronise) --- 입력

동기 Start 입력. Drive Command 실행 후라도 SYNC = L 가 입력되고 있는 동안에는 PULSE 출력을 개시하지 않는다. 그 후에 SYNC = H 가 입력됨으로써 PULSE 출력을 개시한다.

동기 맞춤 기능을 필요로 하지 않을 경우에는 SYNC = H로 고정시켜두거나 Sync 기능 사용 설정을 disable로 하여, 최단 시간에 PULSE 출력을 개시할 수가 있다.

▶ ECUP (External Counter Up) --- 입력

External Counter용 입력 신호. UP 신호 또는 2상 신호의 A상 신호를 입력한다.

입력 사양(UP/DOWN 신호, 2상 신호의 변환 및 2상 신호시의 체배(遞倍)지정)은 User Program에 의해 지정 가능하다.

▶ ECDN (External Counter Down) --- 입력

External Counter용 입력 신호. DOWN 신호 또는 2상 신호의 B상 신호를 입력한다.

입력 사양(UP/DOWN 신호, 2상 신호의 변환 및 2상 신호시의 체배(遞倍)지정)은 User Program에 의해 지정 가능하다.

▶ ICL (Internal Counter Low) --- 출력

내부 Address Compare 결과를 출력한다.

Internal Compare Data > Internal Counter 시에 ICL = H로 된다.

▶ ICG (Internal Counter Great) --- 출력

내부 Address Compare 결과를 출력한다.

Internal Compare Data < Internal Counter 시에 ICG = H로 된다.

▶ ECL (External Counter Low) --- 출력

외부 Address Compare 결과를 출력한다.

external Compare Data > external Counter 시에 ECL = H로 된다.

▶ ECG (External Counter Great) --- 출력

외부 Address Compare 결과를 출력한다.

external Compare Data < external Counter 시에 ECG = H로 된다.

▶ OUT3 ~ OUT0 (Universal Output) --- 출력

User Program에 의해 상시 출력 가능한 범용 출력.

▶ IN3 ~ IN0 (Universal Input) --- 입력

User Program 의해 항상 읽기가 가능한 범용 입력 신호. Signal Search-1 Drive, Signal Search-2 Drive에 있어서 검출 대상 신호로도 사용가능, 갈무리(caption)기능 및 스크립트(script)기능을 사용할 때 선택적으로 프로그램 가능하다. 기본 설정 값으로 IN1(Index)는 엔코더의 Z상 입력으로 사용할 수 있다.

▶ INT (Interrupt) --- 출력

인터럽트 출력신호. 인터럽터 Flag 레지스터를 Read하면 인터럽터 설정이 해제 된다. INT신호 출력 기능의 Resource, Mask 레지스터, Active Level은 User Program에 의해 지정 가능하다.

▶ EXPP\* (External Puls Pulse) --- 입력

(+) 방향 MPG 드라이브 입력 신호. 드라이브 정지 중(BUSY='L')에 Active가 되면 MPG 드라이브가 구동 (BUSY='H')되고 펄스를 출력한다.

▶ EXMP\* (External Minus Pulse) --- 입력

(-) 방향 MPG 드라이브 입력 신호. 드라이브 정지 중(BUSY='L')에 Active가 되면 MPG 드라이브가 구동 (BUSY='H')되고 펄스를 출력한다.

▶ MARK --- 입력

Sensor Positioning I /II/III 드라이브에서 위치 결정 모드 전환 신호이다. Sensor Positioning I /II/III드라이브에서는 MARK 신호가 Active Level로 변환되는 시점에서 미리 설정한 펄스 수 만큼 구동한다. MARK의 Active Level은 User Program에 의해 지정 가능하다.

▶ TRIGGER\_OUT --- 출력

사용자가 설정한 특정 위치에서 혹은 일정한 거리마다 출력되는 trigger 신호이다. 출력 Active Level, Active Level 출력 시간 설정 및 trigger 출력 주기 또는 특정위치는 사용자가 설정 가능하다.

▶ EUO 4~1, EUIO0 (MD12~MD8을 공용으로 사용) --- 출력, 입력/출력

사용자 설정에 의해 출력, 입/출력이 각각 가능하다. 사용자 설정에 따라 EUO4~1, EUIO0 이나 MD12~MD8 중에 선택하여 사용할 수 있다. EUO4~1은 범용 출력 핀이고, EUIO0은 범용 입출력 핀이다. EUIO0 핀은 내부 레지스터 설정에 따라 입력으로

▶ VDD

+5V 또는 +3.3V 전원 입력 단자. 단자 전부 접속한다.

▶ GND

5V 또는 3.3V GND 단자. 단자 전부 접속한다.

▶ I\_VDD

+3.3V 전원 입력 단자. 40, 47, 89, 90번 핀으로 반드시 +3.3V에 접속한다. 단자 전부 접속한다.

▶ L\_GND

3.3V GND 단자. 단자 전부 접속한다. 일반 GND단자와 같이 접속하여 사용해도 무방하다.

## 3. 기능 개요

CAMC-FS은 최고 출력주파수 4MPPS 및 속도 Override, 이동량 Override, 직선 가감속, 자동 S자(포물선) 가감속을 실현한 CAMC-5M 상위 Compatible Single Chip Motor Control LSI이다. 모든 Parameter는 User Program에 의해 항상 설정 변경할 수 있으며, 원하는 제어를 실현할 수 있다. 대상 Motor는 Stepping 및 Servo Motor이며 100 Pin QFP로 구성되어 있다.

### 3.1. 드라이브 기능

#### 3.1.1. Preset Pulse Drive

모터를 설정한 이동량(이동거리)만큼 이동시킬 때 사용하는 드라이브로 삼각 구동 방지 기능이 있다. 이동량을 미리 설정한 후 드라이브를 시작하면 사용자가 지정한 개수만큼의 펄스(이동량)를 출력한 후 정지하여, 원하는 거리만큼 모터를 이동시킬 수 있다. 이때 적용되는 속도 패턴은 사용자가 설정할 수 있다. 구동 중 이동량의 변경이 가능하다. 구동 중 감속 포인트(감속 시작점)는 칩에서 자동으로 설정한 값을 사용할 수 있고(기본), 사용자가 감속할 포인트의 펄스 수를 지정할 수도 있다. 감속 시작점의 펄스 수는 기본값으로는 감속 후 등속으로 구동할 펄스 수가 되며, 감속 포인트 사용자 설정인 경우는 현재 출력되고 남은 펄스 수가 이 값과 같은 시점에서 감속을 시작하는 기준 값으로 사용된다. 지정 가능 펄스 수는 0(0x00000000) ~ 4,294,967,295(0xFFFFFFFF)개이다.

단, 구동 중 PGM register change 사용은 불가능하다.

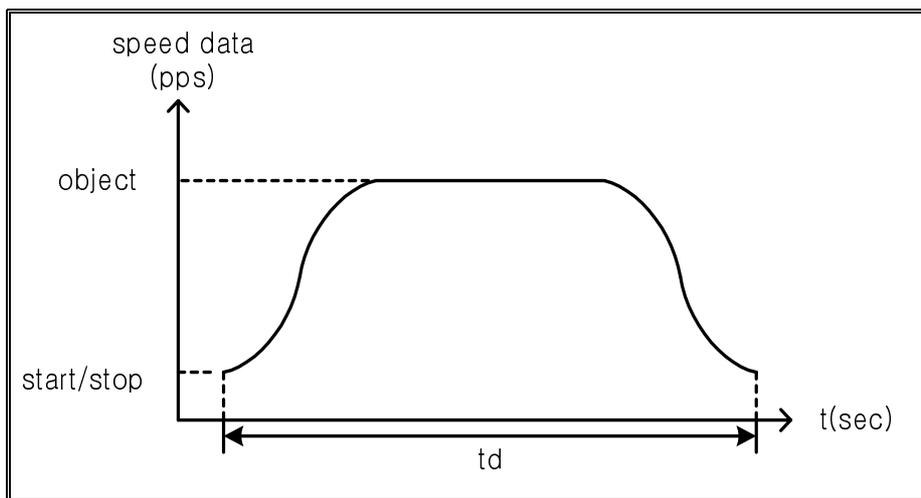


그림 3.1. Preset Pulse Drive 속도 Profile

\*\* CAMC-FS2.1에서는 1 펄스만큼 이동할 경우 예외로 다음과 같은 방법으로 구동해야 정상동작 한다.

<CW 방향 구동 시>

command	Data	Comment
0xC4	0x00000001	Script1 설정 데이터
0xC0	0x000002AA	Script1 설정 command, 드라이브 시작 시 거리 오버라이드 실행(1 펄스 오버라이드)
0xA0	0x00000002	Preset Pulse Drive(CW방향, 2 펄스 이동)

\*\* 위의 표와 같은 순서대로 실행하여 구동하여야 정상적으로 CW방향으로 1 펄스만큼 이동한다.

<CCW 방향 구동 시>

command	Data	Comment
0xC4	0x00000001	Script1 설정 데이터
0xC0	0x000002AA	Script1 설정 command, 드라이브 시작 시 거리 오버라이드 실행(1 펄스 오버라이드)
0xA5	0x00000002	Preset Pulse Drive(CCW 방향, 2 펄스 이동)

\*\* 위의 표와 같은 순서대로 실행하여 구동하여야 정상적으로 CCW 방향으로 1 펄스만큼 이동한다.

### 3.1.2. Continuous Drive

연속적인 모터 구동 동작이 필요할 때 사용하는 드라이브이다. Start command 실행 후 펄스 출력이 시작되고, 급정지 또는 감속정지 command 실행에 의해 드라이브 구동을 정지한다. 정지 command의 종류(급정지, 감속정지)에 따라 감속도 프로파일이 결정된다.

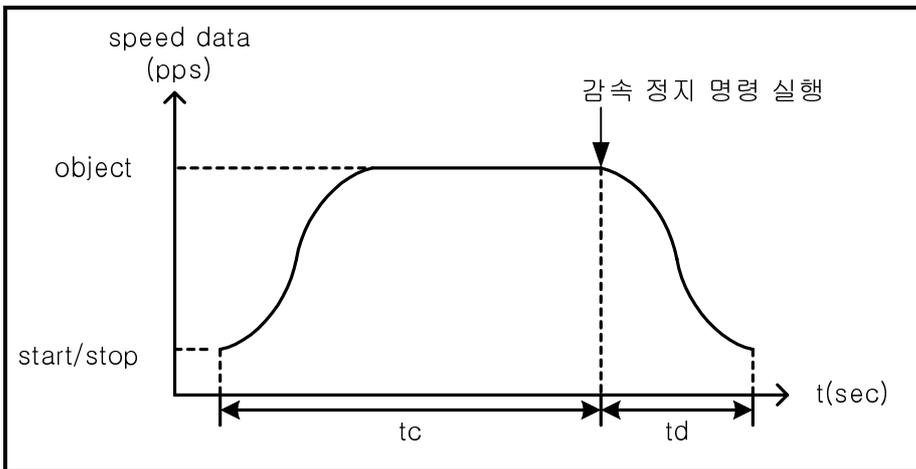


그림 3.2. Continuous Drive 속도 Profile

### 3.1.3. Signal search-1 Drive

지정한 센서 위치에서 정지 또는 감속하여야 할 경우 사용하는 드라이브이다. 드라이브 start command

실행 후 지정된 센서 입력신호의 에지(edge)를 검출하여 감속 정지한다.

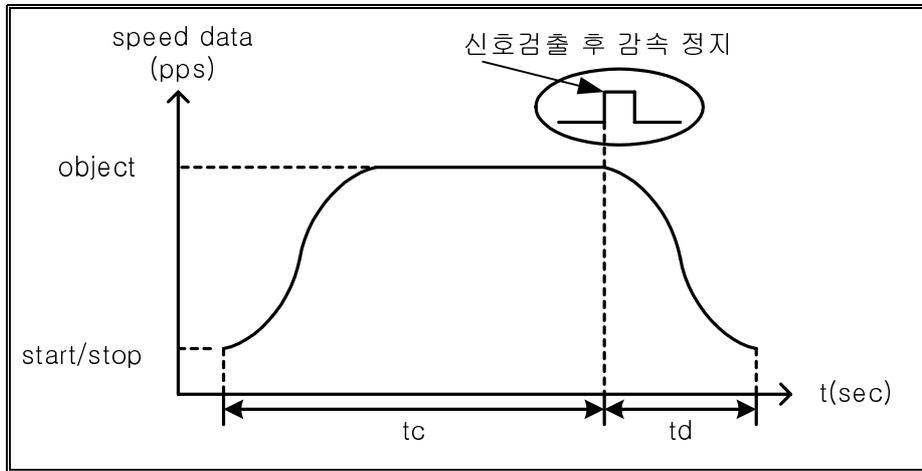


그림 3.3. Signal search - 1 Drive 속도 Profile

### 3.1.4. Signal search-2 Drive

모터를 적은 거리를 등속으로 움직여 센서 위치에 정확하게 정지시킬 때 사용하는 드라이브이다. Signal search-1 Drive와 지정된 센서 위치에 정지시킨다는 공통점을 가지고 있지만, 항상 등속도로 실행되는 드라이브이므로 이동량이 작을 때 사용할 수 있다.

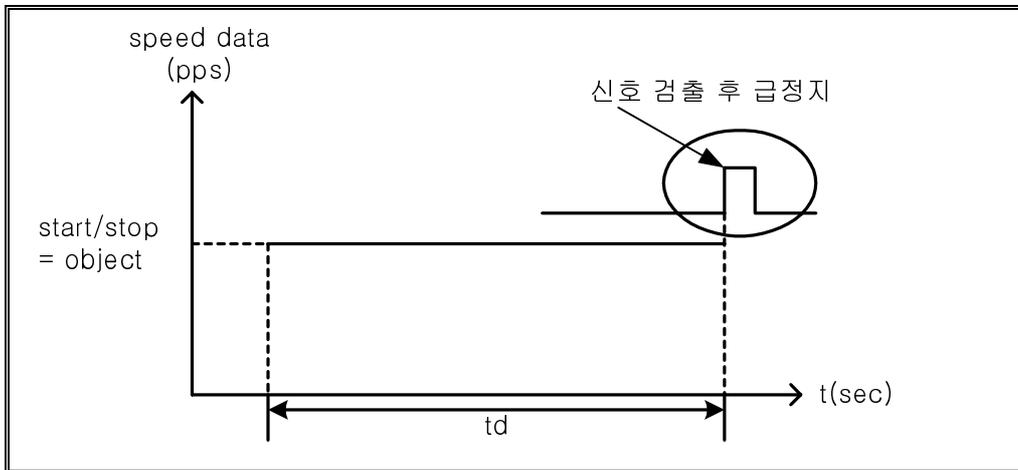


그림 3.4. Signal Search - 2 Drive 속도 Profile

### 3.1.5. Home Search Drive

Home Search Drive는 드라이브 시작 명령으로 가장 기본적인 원점검출 동작을 수행한다. CW(+), CCW(-) 방향으로 구동가능하며 검출할 원점 신호 입력(IN0/ORG)의 edge point(rising/falling)를 사용자가 설정할 수 있다. 그림 3.5.은 (+) 방향 원점 검출 동작을 설명한 그림이다.

▶ (+)방향 원점 Drive Start, Positive Edge에서 정지

ORG신호 Active High일 경우, Drive Start 후 Limit 신호를 먼저 만날 경우

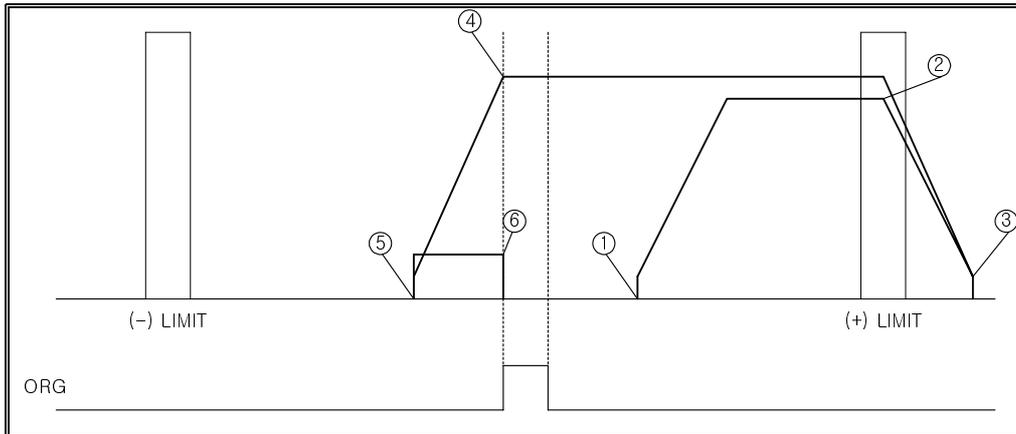


그림 3.5. Home Search Drive 속도 Profile

동작

- ① (+)방향 Signal Search-1 Drive Start
- ② (+)Limit에 의한 감속 정지
- ③ (-)방향 Signal Search-1 Drive Start
- ④ ORG Negative에 의한 감속정지
- ⑤ (+)방향 Signal Search-2 Drive Start
- ⑥ ORG Positive에 의한 급정지

리미트 입력에 의한 방향 전환 시 active level 신호가 길게 입력되면 방향이 다시 변경된다. Home Search Drive의 limit입력은 edge에 의해 동작한다.

### 3.1.6.Sensor Positioning Drive

Sensor Positioning Drive는 특정 센서 신호(MARK)의 입력부터 사용자가 지정한 펄스 수 만큼 구동하는 드라이브이다. 센서 신호는 MARK 입력핀으로 입력되며 이 신호의 Active Level 설정은 변경 가능하다. MARK 신호 입력 후 구동할 펄스수의 설정 범위는 2~4,294,967,295(0x00000002~0xFFFFFFFF)이다.

▶ Sensor Positioning Drive I , 0xb0(+방향), 0xb1(-방향)

Sensor Positioning Drive I은 MARK 신호의 edge가 검출되면, 사용자가 지정한 펄스 수 만큼 구동하는 드라이브이다. Sensor Positioning Drive I에서는 MARK 신호 입력 전에 가속 동작을 수행한다.

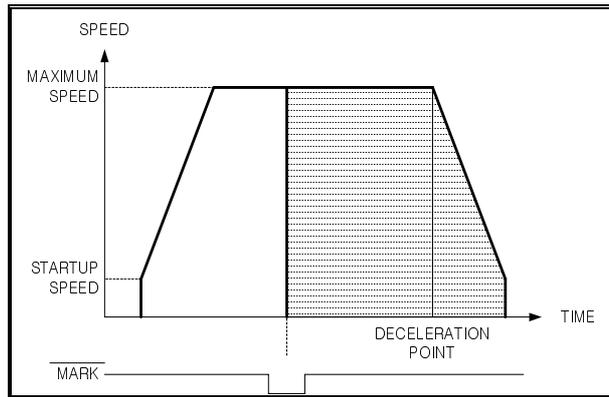


그림 3.6. Sensor Positioning Drive I (+, - 방향) 속도 Profile

▶ Sensor Positioning Drive II, 0xb2(+방향), 0xb3(-방향)

Sensor Positioning Drive II는 MARK 신호의 edge가 검출되면, 사용자가 지정한 펄스수 만큼 가감속하면서 구동하는 드라이브이다. Sensor Positioning Drive II에서는 MARK 신호 입력 전에 Start/Stop Speed data로 등속한다.

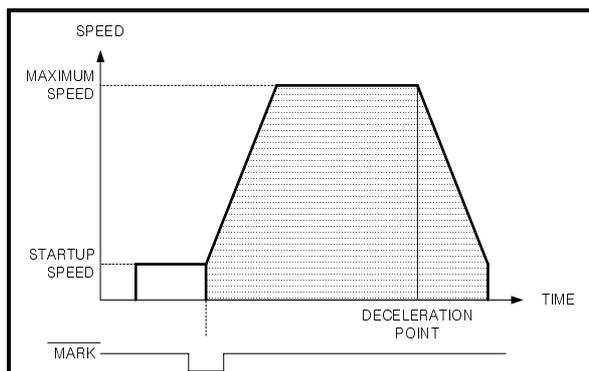


그림 3.7. Sensor Positioning Drive II (+, - 방향) 속도 Profile

▶ Sensor Positioning Drive III, 0xb4(+방향), 0xb5(-방향)

Sensor Positioning Drive III는 MARK 신호의 edge가 검출되면 사용자가 지정한 펄스 수만큼 등속하면서 구동하는 드라이브이다. Sensor Positioning Drive III에서는 전체적으로 Start/Stop Speed data로 등속 동작을 수행한다.

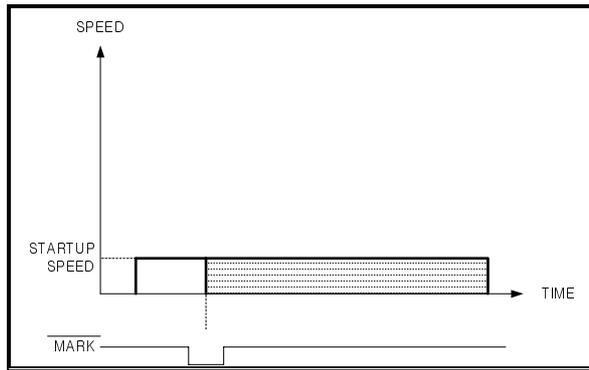


그림 3.8. Sensor Positioning Drive III (+, - 방향) 속도 Profile

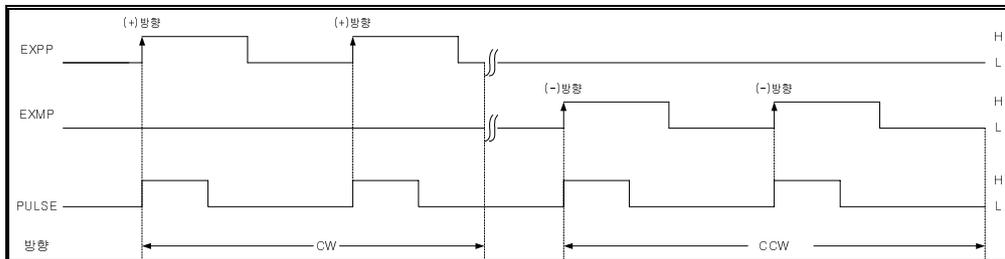
### 3.1.7. MPG Drive

▶ 외부 2상 펄스에 의한 구동

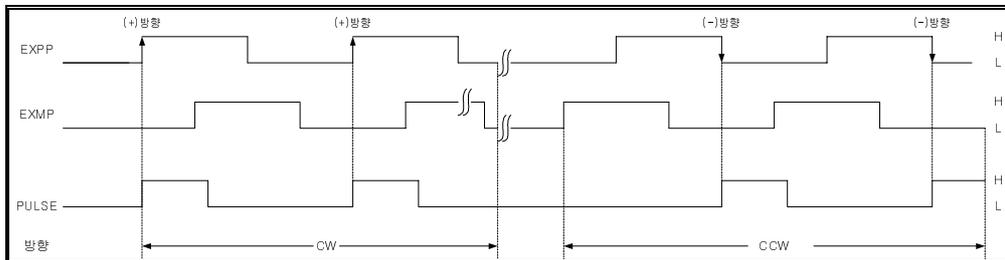
EXPP, EXMP 신호의 입력에 따라 방향 신호가 결정되고, 입력 사양에 따라 펄스 수가 결정된다. 그림은 펄스 수와 방향을 나타낸다.

출력되는 펄스의 속도는 설정된 Object Speed에 의해 출력된다.

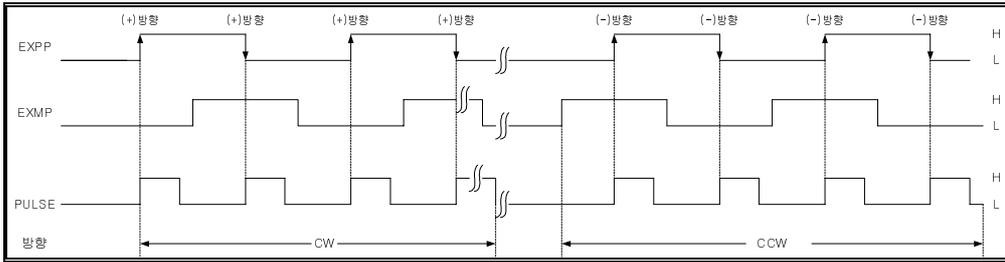
① 입력사양 - 3,4 BIT : 00 일 경우



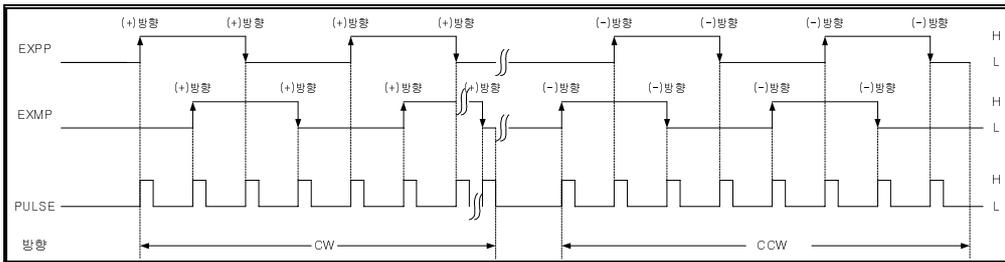
② 입력사양 - 3,4 BIT : 01 일 경우



③ 입력사양 - 3,4 BIT : 10 일 경우



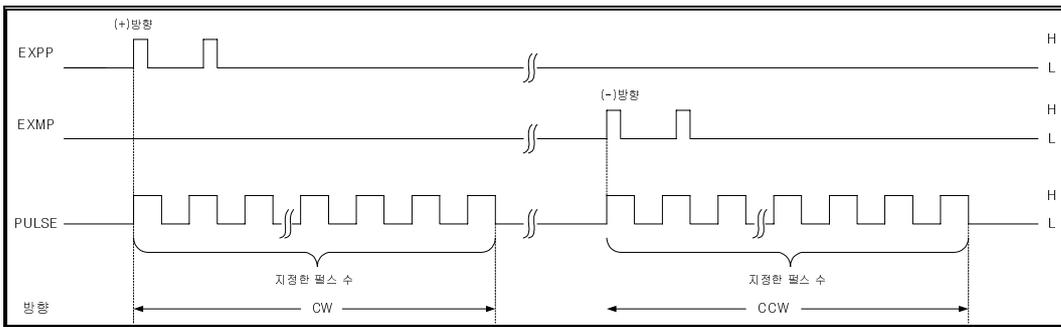
④ 입력사양 - 3, 4 BIT : 11 일 경우



단, 입력 사양을 체제로 구동할 경우 입력되는 Speed가 Object Speed보다 늦어야 한다.

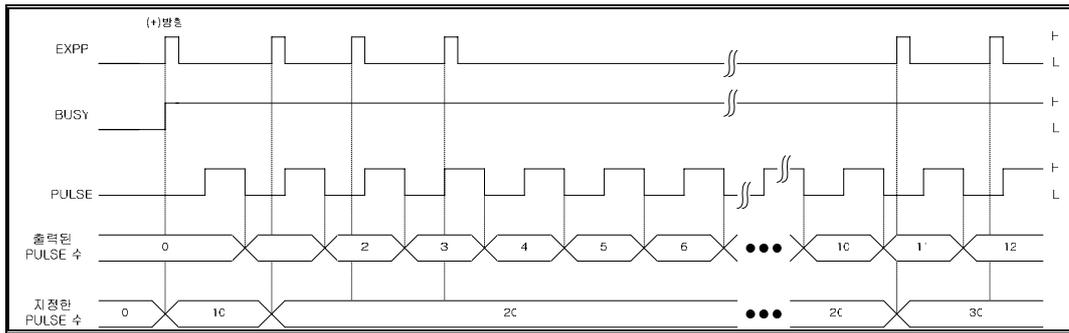
▶ MPG Preset Pulse Drive 구동

EXPP, EXMP 신호의 Edge 신호가 입력되면 설정되어 있던 펄스 수 만큼 출력하고 구동 형태는 Preset Pulse Drive와 같다. 구동 중 EXPP, EXMP 신호의 Edge 신호가 입력되면 설정되어 있는 펄스 수 만큼 Override를 한다. EXPP 신호가 Active Level이 되면 (+)방향, EXMP 신호가 Active Level이 되면 (-) 방향으로 구동한다.



단, 연속적으로 Edge 신호가 입력되면 출력된 펄스 수와 비교하여 입력된 신호에 의해 무시 혹은 Override를 결정한다.

예를 들어 지정 펄스 수를 10개 구동하였을 경우는 다음 그림과 같다.



▶ MPG Continuous Drive 구동

EXPP, EXMP 신호가 Active Level이 되는 동안에 펄스를 출력하고, 구동 형태는 Continuous Drive와 같다.

EXPP 신호가 Active Level이 되는 동안은 (+)방향, EXMP 신호가 Active Level이 되는 동안은 (-)방향으로 구동한다.

### 3.2. 가감속 모드 설정 기능

#### 3.2.1. 대칭 포물선 가감속 모드

대칭 포물선 가감속 모드는 가속 시간과 감속 시간이 같은 형태로 S-curve 속도 프로파일을 가지는 모드이다. SW1 데이터는 감속 시 S자 구간을 설정하는 데이터이다. SW1 데이터는 리셋 후 최고값으로 설정되어 있으며 이 경우 Full-S 패턴의 속도 프로파일을 생성할 수 있다. SW1 데이터를 (Object-Std)/2 보다 작은 값으로 설정하여 직선 가감속 구간을 삽입할 수 있다. SW1 데이터를 0x7FF로 정한 후 드라이브 동작 설정 값을 0x02로 정하면 대칭 포물선(S-curve)가감속 모드에서 드라이브를 구동할 수 있다. S-curve 가감속 시간은 시작/목표 속도와 Rate-1 데이터로 설정한다. S-curve 생성 시 Rate-1 값이 변화할 경우 동작의 정확성을 보증할 수 없다.

S-curve 구간의 가감속 시간 단위는 다음 식과 같다.

$$T_{unit} = \frac{Rate - 1 \ Data \times 16}{F_{clk}} : \text{가감속 시간 설정단위(sec)}$$

S-curve의 가감속 시간은 다음 식과 같다.

$$T_{up/down} = T_{unit} \times \left( \left| \frac{\text{목표속도} - \text{시작속도}}{\text{Rate-1}} \right| \right) : \text{가감속 시간(sec)}$$

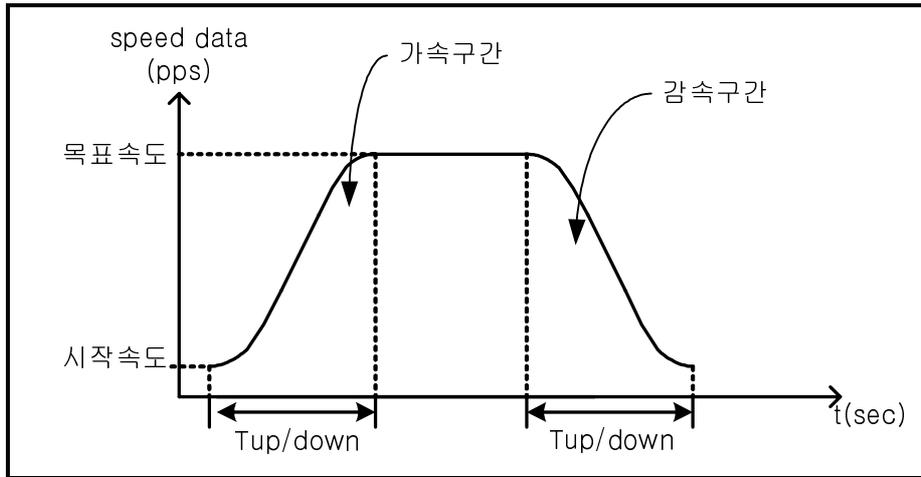


그림 3.9. 대칭 포물선 가감속 모드 속도 Profile

3.2.2. 비대칭 포물선 가감속 모드

비대칭 포물선 가감속 모드는 가속 시간과 감속 시간이 다른 형태로 S-curve 속도 프로파일을 가지는 모드이다. 드라이브 동작 설정 값을 0x03로 설정하면 비대칭 포물선(S-curve)가감속 모드에서 드라이브를 구동할 수 있다. 비대칭 S-curve 가속 시간은 시작/목표 속도와 Rate-1 데이터로 설정하고 직선 감속 시간은 시작/목표 속도와 Rate-2 데이터로 설정한다. SW1은 가속시 S자 구간을 SW2는 감속시 S자 구간을 설정하는 데이터이다. SW1, SW2 데이터는 리셋후 최고값으로 설정되어 있으며 이 경우 Full-S 패턴의 속도 프로파일을 생성할 수 있다. SW1, SW2 데이터를 (Obj-Std)/2 보다 작은 값으로 설정하여 직선 가감속 구간을 삽입할 수 있다.

드라이브 구동 시 Rate-1 값 혹은 Rate-2 값이 변화할 경우 동작의 정확성을 보증할 수 없다.

① 비대칭 S-curve 구간의 가속 시간 설정 단위와 가속 시간

$$T_{unit(up)} = \frac{Rate - 1 Data \times 16}{F_{clk}} : \text{가속 시간 설정단위(sec)}$$

$$T_{up} = T_{unit(up)} \times \left( \left| \frac{\text{목표속도} - \text{시작속도}}{\text{시작속도}} \right| \right) : \text{가속 시간(sec)}$$

② 비대칭 S-curve 구간의 감속 시간 설정 단위와 가속 시간

$$T_{unit(down)} = \frac{Rate - 2 Data \times 16}{F_{clk}} : \text{감속 시간 설정단위(sec)}$$

$$T_{down} = T_{unit(down)} \times \left( \left| \frac{\text{목표속도} - \text{시작속도}}{\text{시작속도}} \right| \right) : \text{감속 시간(sec)}$$

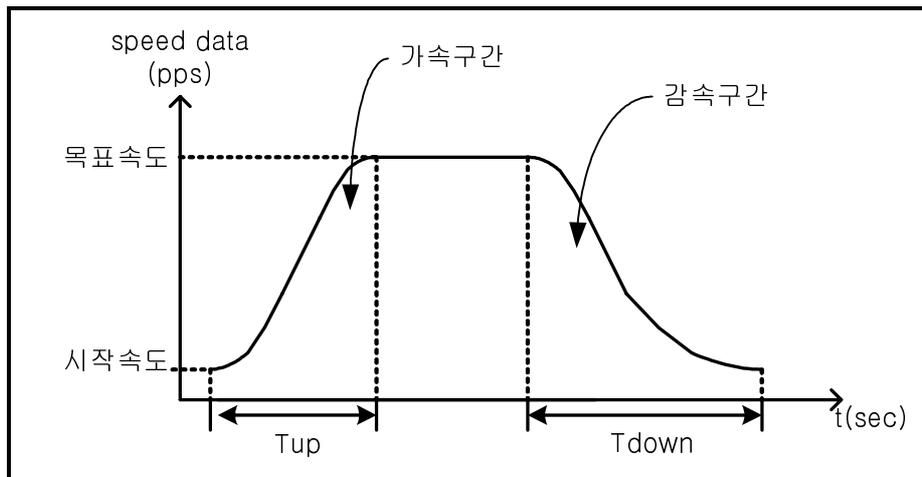


그림 3.10. 비대칭 포물선 가감속 모드 속도 Profile

3.2.3. 대칭 직선 가감속 모드

대칭 직선 가감속 모드는 가속 시간과 감속 시간이 같은 형태로 직선 가감속도 프로파일을 가지는 모드이다. 드라이브 동작 설정값을 0x00으로 설정하면 대칭 직선 가감속 모드에서 드라이브를 구동할 수 있다. 직선 가감속 시간은 시작/목표 속도와 Rate-1/ Rate-2/ Rate-3 데이터로 설정한다. 드라이브 구동 시 Rate-1/ Rate-2/ Rate-3 값이 변화할 경우 동작의 정확성을 보증할 수 없다.

직선 구간의 가감속 시간 설정단위는 다음 식과 같다.

$$T_{unit} = \frac{Rate - n \text{ Data} \times 8}{F_{clk}} : \text{가감속 시간 설정단위(sec)} \quad (n=1, 2, 3)$$

직선 가감속 시간은 다음 식과 같다.

$$T_{up/down} = T_{unit} \times (\text{목표속도} - \text{시작속도}) : \text{가감속 시간(sec)}$$

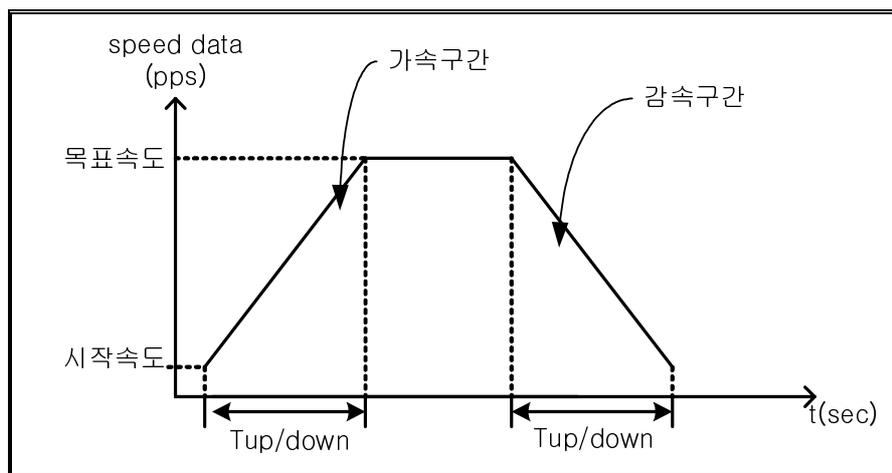


그림 3.11. 대칭 직선 가감속 모드 속도 Profile

3.2.4.비대칭 직선 가감속 모드

비대칭 직선 가감속 모드는 가속 시간과 감속 시간이 다른 형태로 직선 가감속도 프로파일을 가지는 모드이다. 드라이브 동작 설정값을 0x01로 설정하면 비대칭 직선 가감속 모드에서 드라이브를 구동할 수 있다. 직선 가속 시간은 시작/목표 속도와 Rate-1 데이터로 설정하고 직선 감속 시간은 시작/목표 속도와 Rate-2 데이터로 설정한다. 드라이브 구동 시 Rate-1 값 혹은 Rate-2 값이 변화할 경우 동작의 정확성을 보증할 수 없다.

① 비대칭 직선 구간의 가속 시간 설정 단위와 가속 시간

$$T_{unit(up)} = \frac{Rate-1 Data \times 8}{F_{clk}} : \text{가속 시간 설정단위(sec)}$$

$$T_{up} = T_{unit(up)} \times \left( \left| \frac{\text{목표속도} - \text{시작속도}}{\text{시작속도}} \right| \right) : \text{가속 시간(sec)}$$

② 비대칭 직선 구간의 감속 시간 설정 단위와 감속 시간

$$T_{unit(down)} = \frac{Rate-2 Data \times 8}{F_{clk}} : \text{감속 시간 설정단위(sec)}$$

$$T_{down} = T_{unit(down)} \times \left( \left| \frac{\text{목표속도} - \text{시작속도}}{\text{목표속도}} \right| \right) : \text{감속 시간(sec)}$$

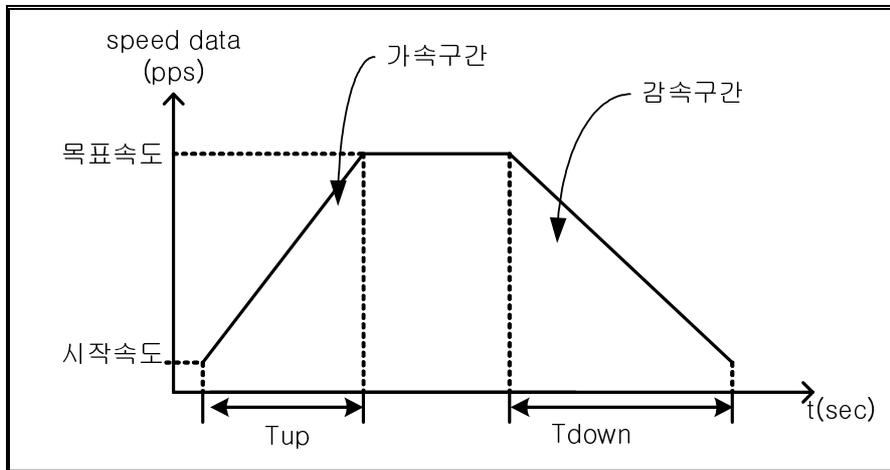


그림 3.12. 비대칭 직선 가감속 모드 속도 Profile

### 3.3. 감속 개시 포인트 검출 방식 선택 기능

Preset Pulse Drive 실행 시 감속 개시 포인트 검출 방식은 두 가지가 있다.

▶ 자동검출 방식(기본값)

- 지정 펄스 수에 따라 CAMC-FS가 자동적으로 계산하여 감속 포인트를 검출한 후 감속을 시작한다.

▶ 남은 펄스 수 방식

- 현재 출력된 펄스 수에서 지정 펄스 수까지의 남은 펄스 수와 사용자가 설정한 남은 펄스수가 일치한 시점부터 감속을 시작한다.

### 3.4. 서보 모터 인터페이스 기능

CAMC-FS에서는 다음과 같은 서보 모터 제어용 신호를 제공하여 서보 모터와의 인터페이스가 가능하다.

▶ INP(인포지션)

- 일반적으로 펄스 입력 타입의 서보 모터 드라이버 내부에는 명령 펄스 입력과 피드백 펄스 입력과의 차이를 카운트하는 편차 카운터가 있으며, 그 차이가 영(zero)이 되도록 모터를 제어한다. 따라서 원리적으로 모터의 움직임은 명령 펄스보다도 지연되어가 동작하고, 명령 펄스를 정지하더라도 편차 카운터에 쌓여 있는 누적량이 영(zero)이 될 때까지 모터는 정지하지 않는다. 이 때문에 서보 모터 드라이버는 모터가 정지하였음을 표시하는 Inposition(위치결정완료)신호를 출력한다. 이 신호를 INP입력 신호의 입력신호와 연결하여 위치결정완료신호로 사용할 수 있다.

▶ ALM(강제 정지)

- 서보 모터 드라이버의 alarm 신호를 입력한다. 이 신호가 On(active 신호가 입력)되면 펄스 구동 드라이브가 급정지 한다.

### 3.5. 다축 동기 맞춤 기능

CAMC-FS 칩을 여러 개 동시에 사용할 경우의 동기 맞춤 기능으로서 SYN 입력 신호가 준비되어 있다. 이 입력 신호가 Low level이 입력되어 있는 동안에는 드라이브 시작 command가 실행되어도 펄스 출력을 하지 않고, BUSY 신호만 High level로 출력된다. 따라서 여러 개의 칩의 펄스 출력을 동시에 할 경우, SYN 입력을 Low로 두고 각 칩에 드라이브 시작 command를 실행한 후 SYN 입력에 High 신호를 입력하면 된다.

CAMC-FS2.1 버전 칩에서는 SYN 입력 신호 사용 여부를 사용자가 software적으로 설정 가능하다. 따라서, 여러 축을 CAMC-FS로 구동 시 동기 구동을 하는 축과 동기 구동을 하지 않는 축들을 혼용하여 제어하는 시스템의 하드웨어 구현이 용이하다.

### 3.6. 범용 입출력 기능

사용자의 software에서 항상 read/write가 가능한 범용 입출력 신호가 각각 4개씩 제공됩니다. 범용 입력 신호는 주로 신호검출 드라이브에서 검출 대상 신호로 사용되고, 원점검출 드라이브에서는 원점신호 입력으로도 사용됩니다. 또한 외부카운터에 대한 외부 Clear 입력신호로도 사용 가능합니다.

### 3.7. 기계계 외부 입력 제어 기능

기계계로부터의 센서 입력 신호를 검출하여 그에 따른 동작을 할 수 있다. 이 신호들의 active level은 active high(0=low, 1=high)나 active low(0=high, 1=low)로 사용자가 설정가능하며, 센서 입력 신호 폭을 기준 클록의 1~255배( $T_{clk} \sim 255 \times T_{clk}$ ,  $T_{clk}$ :기준클록) 사이에 설정 가능하다.

▶ **±Emergency stop Limit 신호(±ELM)**

정(+)/역(-) 방향으로 limit 급정지신호(+ELM, -ELM)가 있어 기계계의 over run을 방지할 수 있다.

▶ **±Slowdown stop Limit 신호(±SLM)**

정(+)/역(-) 방향으로 limit 감속정지 신호(+SLM, -SLM)가 있어 기계계의 over run을 방지할 수 있다. 주로 이 입력 신호는 ±ELM 입력 신호로 급정지하기 전에 감속하여 기계계의 부하를 줄이기 위해 사용한다.

▶ **Emergency Stop(ESTOP)**

급정지를 위한 센서 입력 신호이다.

▶ **Slow down Stop(SSTOP)**

감속정지를 위한 센서 입력 신호이다.

### 3.8. 펄스출력 방식 전환 기능

접속한 모터 드라이버의 입력 사양에 맞추어 펄스 출력 방식을 선택할 수 있도록 8종류의 펄스 출력 방식이 제공됩니다. 제공되는 펄스의 종류는 다음과 같다.

▶ **1 펄스 방식**

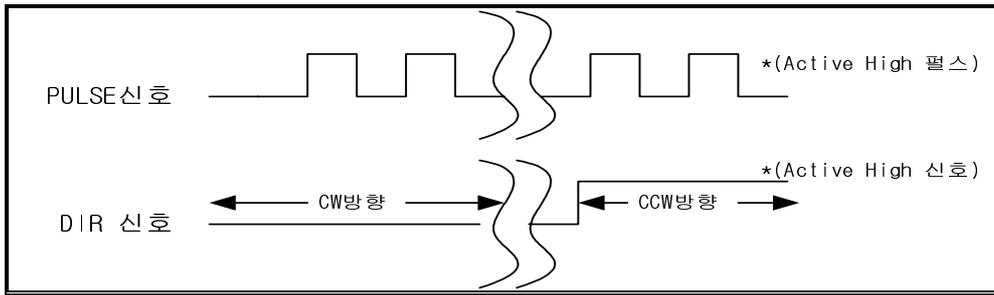


그림 3.13. 1 펄스 방식 - 1

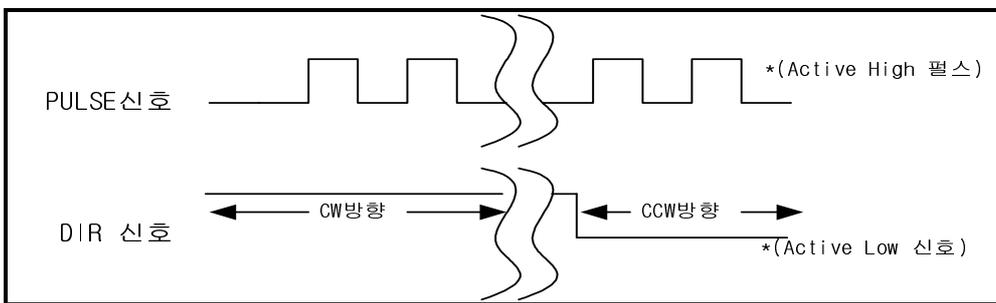


그림 3.14. 1 펄스 방식 - 2

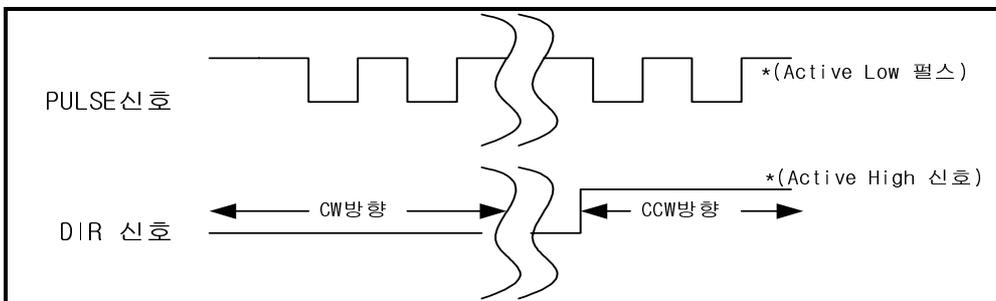


그림 3.15. 1 펄스 방식 - 3

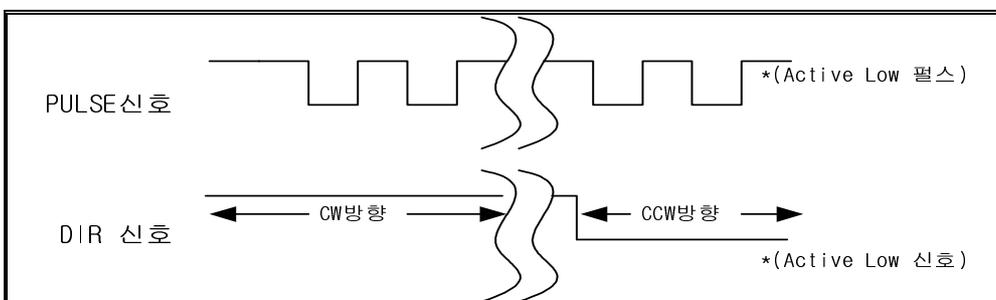


그림 3.16. 1 펄스 방식 - 4

▶ 2 펄스 방식

그림 3.17. 2펄스 방식 - 1

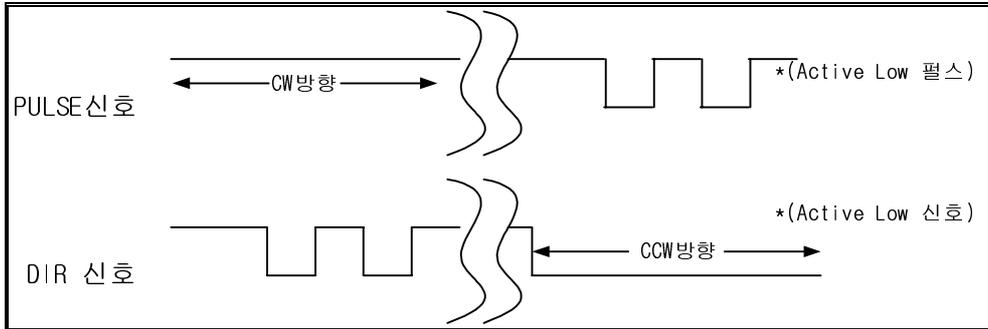


그림 3.18. 2펄스 방식 - 2

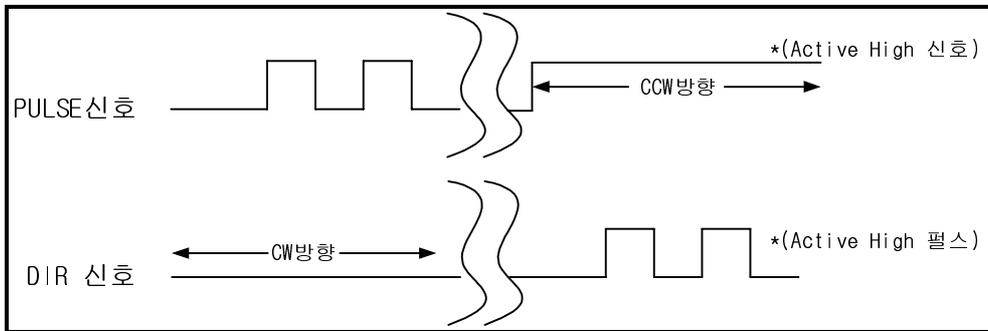


그림 3.19. 2펄스 방식 - 3

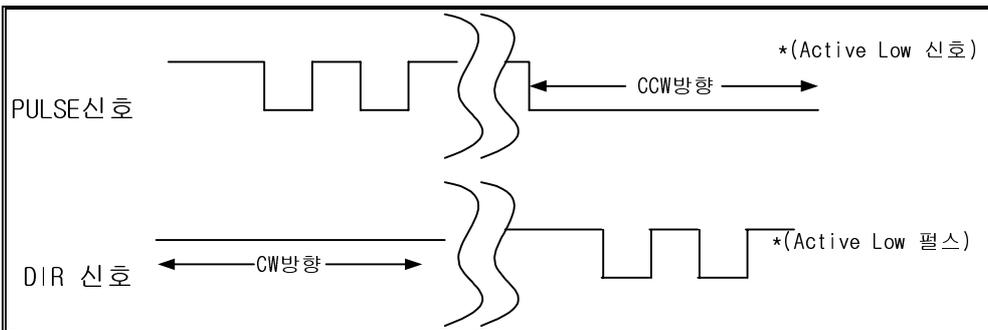


그림 3.20. 2펄스 방식 - 4

### 3.9. 내부 위치 링 카운터(Internal Ring Counter) 관리 기능

CAMC-FS는 칩에서 출력되는 펄스를 카운트하는 32-bit up/down counter가 내장되어 있어 이론상의 위치를 알 수 있다. 내부 카운터는 출력 펄스가 CW(+) 방향일 때 up-count(펄스 한 개 출력 때마다 카운터값 +1 증가)하고, CCW(-) 방향의 출력펄스에 대하여는 down-count(펄스 한 개 출력할 때마다 카운터값 -1 감소) 동작을 수행한다. 따라서 펄스 출력 중이거나 종료한 후에 이 카운터값을 읽어봄으

로써 이론상의 현재 위치를 알 수 있다.

내부 위치 링 카운트 기능이 있으며, M\_DATA와 P\_DATA를 설정할 수 있는 32-bit 레지스터가 있다.

단, M\_DATA < P\_DATA 조건이 성립하여야 한다.

내부 위치 링 카운트 기능이란 설정한 범위 내에서 무한대로 카운트하는 기능이다. INTERNAL COUNTER M\_DATA를 -10, INTERNAL COUNTER P\_DATA를 10으로 설정하였을 경우 내부 카운터의 값은 다음과 같이 카운트 된다.

(CW 방향인 경우) 0 → 1 → 2 → ... → 10 → -10 → -9 ... → -1 → 0

(CCW방향인 경우) 0 → -1 → ... → -9 → -10 → 10 ... → 2 → 1 → 0

이 기능은 모터를 원운동에 사용할 경우의 현재 위치 관리에 유용하게 사용할 수 있다. 원을 한바퀴 도는 펄스 수를 M\_DATA와 P\_DATA에 설정하면 한바퀴 움직인 후 내부 카운터 값이 자동으로 클리어 되므로 초기 시작점에서의 현재 위치 정보를 내부 카운터 값을 읽어봄으로써 알 수 있다.

### 3.10. 외부 위치 링 카운터(External Ring Counter) 관리 기능

CAMC-FS는 외부 펄스로부터 feedback받은 up/down신호,또는 2상 신호를 카운트하는 32-bit up/down counter가 내장되어 있다. 외부 카운터는 기계계에서 feedback된 펄스를 카운트하여 기계계의 현재 위치 정보를 알려주며, 외부 카운터 값과 비교도 가능하여 이론적인 위치와 기계계의 위치를 비교할 수도 있다. 외부 카운터의 입력 사양(2상 신호시의 체배(遞倍))은 소프트웨어적으로 변경 가능하다.

외부 위치 링 카운트 기능이 있으며, M\_DATA와 P\_DATA를 설정할 수 있는 32-bit 레지스터가 있다.

단, M\_DATA < P\_DATA 조건이 성립하여야 한다.

외부 위치 링 카운트 기능이란 설정한 범위 내에서 무한대로 카운트하는 기능이다. EXTERNAL COUNTER M\_DATA를 -10, EXTERNAL COUNTER P\_DATA를 10으로 설정하였을 경우 외부 카운터의 값은 다음과 같이 카운트 된다.

(CW 방향인 경우) 0 → 1 → 2 → ... → 10 → -10 → -9 ... → -1 → 0

(CCW방향인 경우) 0 → -1 → ... → -9 → -10 → 10 ... → 2 → 1 → 0

이 기능은 모터를 원운동에 사용할 경우의 현재 위치 관리에 유용하게 사용할 수 있다. 원을 한바퀴 도는 펄스 수를 M\_DATA와 P\_DATA에 설정하면 한바퀴 움직인 후 외부 카운터 값이 자동으로 클리어 되므로 초기 시작점에서의 현재 위치 정보를 외부 카운터 값을 읽어봄으로써 알 수 있다.

### 3.11. 내부 위치 카운터 비교 기능

내부 위치 링 카운터 값과 비교할 값을 저장하는 32-bit 레지스터가 있으며, 이 값과 내부카운터 값을 비교하는 기능이 있다. 내부 카운터와 비교할 데이터를 내부 비교 데이터라고 부르며 이 값은 사용자가 소프트웨어적으로 변경 가능하다. 두 값의 비교는 2진 보수 비교를 하며, 결과는 상태 읽기를 실행하여 항상 확인 가능하다.

\*32-bit 2진 보수의 범위를 10진수로 변환하여 나타내면 다음과 같다.

2진보수 : 0x80000000 ....., 0xFFFFFFFF, 0x00000000, 0x00000001, ....., 0x7FFFFFFF

10진수 : -2,147,483,647 ....., -1, 0, 1, ....., 2,147,483,647

### 3.12. 외부 위치 카운터 비교 기능

외부 위치 링 카운터 값과 비교할 값을 저장하는 32-bit 레지스터가 있으며, 이 값과 외부카운터 값을 비교하는 기능이 있다. 외부 카운터와 비교할 데이터를 외부 비교 데이터라고 부르며 이 값은 사용자가 software적으로 변경 가능하다. 두 값의 비교는 2진 보수 비교를 하며, 결과는 상태 읽기를 실행하여 항상 확인 가능하다.

### 3.13. 내부/외부 위치 Scale 카운트 기능

Scale 카운트 기능이란 내부/외부 위치 링 카운터에 펄스가 입력되면 지정된 scale값의 배수 크기만큼 카운트 되면 내부/외부 카운터 값이 '1' 증가하는 기능이다. 이 때 Scale 데이터는 8-bit이며 내부/외부 카운터용으로 각각 내장되어 있다. 내부 카운터 Scale 데이터를 8로 설정하고 CW 펄스 출력동작을 실행할 경우 카운터 값을 다음과 같이 증가한다.

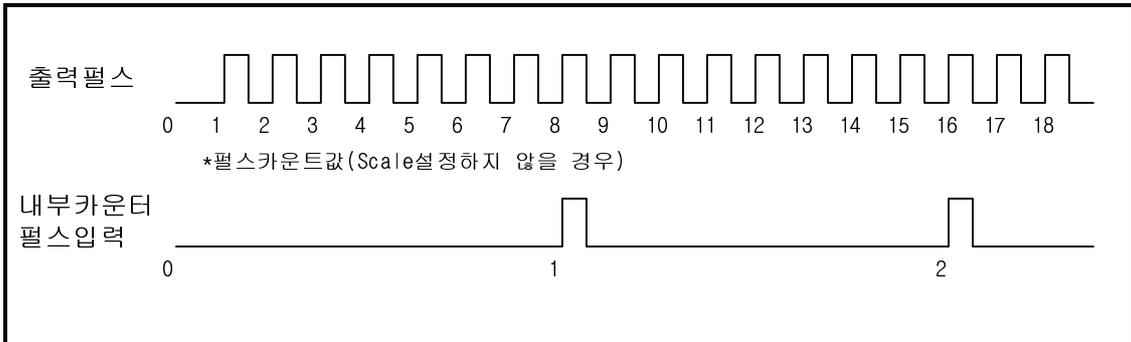
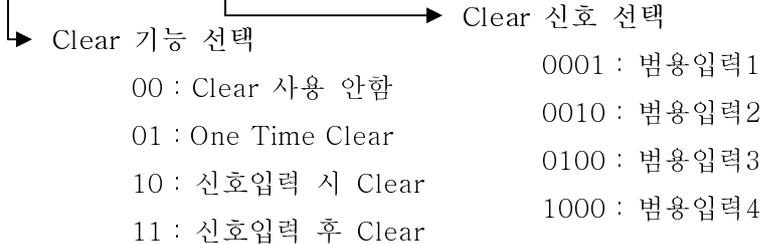
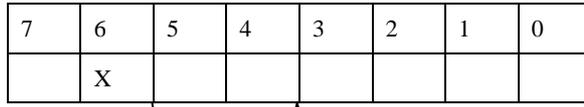


그림 3.21. 내부 위치 Scale 카운트 예제

### 3.14.외부 위치 카운터 Clear 기능

범용 입력 신호 중 한 개를 설정하여 외부 위치 카운터를 clear하는 신호로 사용할 수 있다. 외부 카운터 clear 기능은 선택한 범용 입력 신호에 High Level이 입력되면 외부 카운터를 Clear할 수 있다. 외부 카운터 Clear 신호 선택과 기능 선택은 외부 카운터 설정 명령으로 설정할 수 있다.

[외부 카운터 설정 데이터]



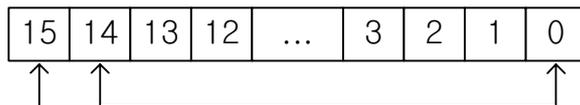
\* Clear 기능 선택에서 One Time Clear(01)는 Clear 신호에 High가 입력되면 외부 카운터는 Zero가 되고 기능 선택 값은 00로 변경된다. 다시 외부 카운터를 Clear하고자 할 경우 기능선택 값을 다시 01로 설정한 후 사용하여야 한다. 신호 입력 시 Clear(10)는 Clear 신호에 High가 입력되더라도 설정 값은 그대로 유지되므로 Clear 신호에 High가 입력될 때마다 외부카운터는 Zero가 된다. 신호 입력 후 Clear(11)는 Clear 신호에 High가 입력되면 외부카운터는 Zero값이 유지된다. 다시 외부 카운터를 동작하기 위해서는 Clear 기능 선택 값을 11이외의 값으로 설정하여야 한다.

### 3.15. 외부 위치 카운터 방향 반전 기능

엔코더 입력 신호를 카운트하는 외부 위치 카운터의 카운트 방향을 반전시킬 수 있다. 이 기능의 설정은 소프트웨어적으로 변경 가능하다.

### 3.16. 편차량 산출 기능

이론상의 위치(내부 위치 카운터)와 기계계의 현재 위치(외부 위치 카운터)와의 편차(deviation) 계산이 가능하다. 편차량은 [내부 위치 카운터값 - 외부 위치 카운터값]과 같으며 15-bit 데이터로 나타난다.



편차량 부호      편차량 절대값(내부 위치 카운터와 외부 위치 카운터의 차이값)

### 3.17. 속도 데이터/편차량 데이터 모니터 기능

서보 드라이브를 위한 보조 기능으로서, 속도 데이터나 편차량 데이터를 Sign bit(부호비트)와 모니터 데이터(13-bit) 출력 신호를 모니터 할 수 있는 기능이 있다.

MONI 입력 신호로써 모니터 데이터를 결정하고, 출력신호는 부호비트와 모니터 데이터로 구성된다.

MONI = Low (편차량 데이터 모니터)

Sign 신호 - Low

MD12~MD0 - 편차량 데이터(16-bit)의 하위 13-bit 데이터

MONI = High(속도 데이터 모니터)

Sign 신호 - 방향 (Low : 정방향, High : 역방향)

MD12~MD0 - 현재 속도 데이터(16-bit)의 상위 13-bit 데이터

### 3.18. 인터럽트 발생 기능

사용자 CPU에 인터럽트를 알리기 위해 인터럽트 발생에 필요한 32개의 인터럽트 소스와 각각의 인터럽트를 가능 또는 불가능하게 만드는 32개의 인터럽트 마스크가 있다. 또한 인터럽트 출력 Active Level 변경이 가능하다. 주의) Active Level의 초기값은 High이다.

### 3.19. 설정 데이터 에러 판정 기능

사용자가 설정한 데이터의 대소 관계를 항상 검사하는 기능이 있다. 설정한 데이터가 드라이브를 실행하기에 부적합한 경우 데이터 에러가 발생하며, 이 경우 드라이브가 진행 중이면 급정지하여 드라이브가 종료하고 데이터 에러가 발생한 상태에서 드라이브를 실행할 경우 바로 급정지하게 된다.

### 3.20. 탈조(Off-Range) 검출 기능

내부 카운터와 외부 카운터의 차이는 현재 모터의 이론적인 위치와 실제 위치의 차이와 같다. CAMC-FS는 이런 차이 값이 설정한 데이터보다 크게 되면 탈조로 간주하고 드라이브를 급정지/감속 정지하는 기능이 있다. 이때 설정할 수 있는 데이터를 탈조검출 데이터라고 한다. 이 데이터는 8-bit(1 ~ 255)값을 가질 수 있으며 값이 0 일 경우는 탈조 검출 기능을 사용하지 않는 경우이다.

### 3.21. 실거리 이동 연산 기능

지정 펄스 수 드라이브를 실행할 경우 이동거리를 출력펄스로 계산하지 않고 외부 엔코더 출력 신호를 카운트하여 사용할 수 있다. 이 기능을 실거리 이동 연산 기능이라 한다. 실거리 이동 연산 기능을 이용하여 지정 펄스 수 드라이브를 실행할 경우 구동 모터의 이동거리를 정확하게 제어할 수 있다. 그러나 출력 펄스카운트 값과 외부 엔코더 카운트 값 차이가 큰 경우 이동 거리에 따른 속도 가 감속 제어가 정확하지 않을 수 있으므로 사용 시 주의가 필요하다.

### 3.22. PWM 출력 기능

CAMC-FS는 현재 속도 데이터 값에 비례하는 PWM 신호를 출력하는 기능이 있다. 출력 파형의 주기는 PWM 레지스터값(0x00~0x06)에 따라 결정된다. PWM 레지스터값에 따른 PWM 파형의 모양은 다음과 같다.

$$F_{pwm} = \text{LSHIFT}(F_{clk}, 16 - R_{pwm}) \text{ [Hz]}$$

$$T_{high} = \text{LSHIFT}(R_v, R_{pwm}) * T_{clk} \text{ [ns]}$$

$F_{pwm}$ : PWM 신호 주파수

$T_{high}$ : PWM 신호가 High로 유지되는 시간

$R_{pwm}$ : PWM 레지스터 데이터

$F_{clk}$ : 기준 클럭 주파수

$R_v$ : 현재 속도 데이터

$\text{LSHIFT}(A,B)$ : A를 B만큼 left shift,  $A/2^B$

### 3.23. 외부 센서 필터 대역폭(Bandwidth) 설정 기능

외부 센서 신호들의 필터 대역폭을 변경 가능하다. 센서 필터 대역폭 설정 레지스터값(BW)에 따라 센서 신호의 유효신호의 시간조절이 가능하다. BW 레지스터 값은 0~255까지 설정 가능하며, 값이 '0' 일 경우는 센서 신호를 사용하지 않으며, 나머지 값을 가질 경우  $T_{clk} * BW$  시간 이상 유효한 센서 신호가 입력되면 그에 따른 동작을 수행한다. BW 레지스터의 RESET 후 기본값은 0x05가 된다.

### 3.24. 스크립트(Script) 기능

CAMC-FS는 칩에서 발생하는 특정한 동작이 실행될 때(특정 이벤트가 발생하였을 때) 실행할 명령어와 그에 필요한 데이터를 저장하는 32-bit 레지스터가 각각 16개씩 존재한다. 스크립트 레지스터에 저장된 이벤트가 발생하면 저장된 명령이 수행되며 이 기능을 스크립트 기능이라 한다. 이 때 동작 이벤트는 두개까지 지정 가능하며 두개의 동작 이벤트는 논리 AND(두 동작 이벤트가 동시에 검출할 경우 명령 실행), 논리 OR(두 동작 이벤트 중 한 개라도 검출할 경우 명령 실행), 논리 XOR(두 동작 이벤트 중 한 개만 검출될 때 명령 실행)에 의한 동작 이벤트 검출이 가능하다.

스크립트 설정은 16개 중 기본적으로 4개로 설정 가능하며 나머지 12개는 순차적인 검출만이 가능하다.

스크립트 기능을 사용하면 드라이브 실행을 연속적으로 실행할 경우 유용하게 사용할 수 있다. 센서 신호 검출 후 바로 역방향으로 연속 드라이브를 실행할 경우, 드라이브가 정지하는 동작 이벤트에 다시 연속 드라이브 실행 명령을 저장해 놓고 신호 검출-1 드라이브를 실행할 경우, 드라이브 정지 후 다시 다른 드라이브 명령을 입력하지 않아도 스크립트 기능에 의해 자동적으로 저장한 드라이브가 실행된다.

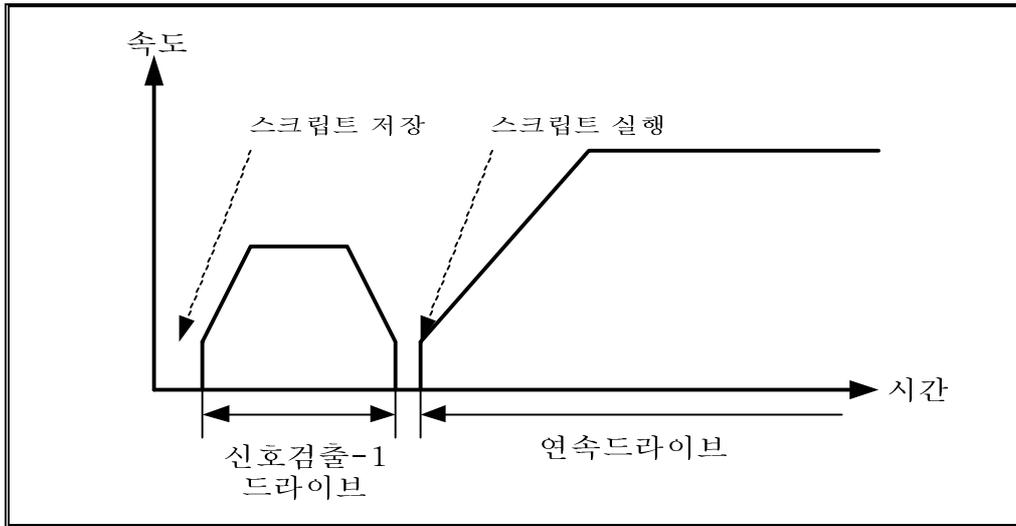


그림 3.22. 스크립트 기능 사용 예

### 3.25. 갈무리(Caption)기능

특정 동작 이벤트에 명령을 실행하는 스크립트 기능과 달리, 이때의 레지스터 상태를 저장하는 기능이 있다. 이 기능을 갈무리 기능이라 한다.

갈무리 기능을 위한 레지스터는 동작이벤트, 명령을 설정하는 32-bit 레지스터 12개와 이 때의 데이터를 저장하는 레지스터 12개가 존재한다. 갈무리 기능에 사용하는 명령들은 저장된 레지스터들을 읽는 명령들만 사용하며, 이 명령이 실행한 후 결과는 데이터 레지스터에 순차적으로 저장된다. 따라서 사용자가 CPU를 통하여 갈무리 데이터 레지스터를 읽어봄으로써 그 결과를 알 수 있다.

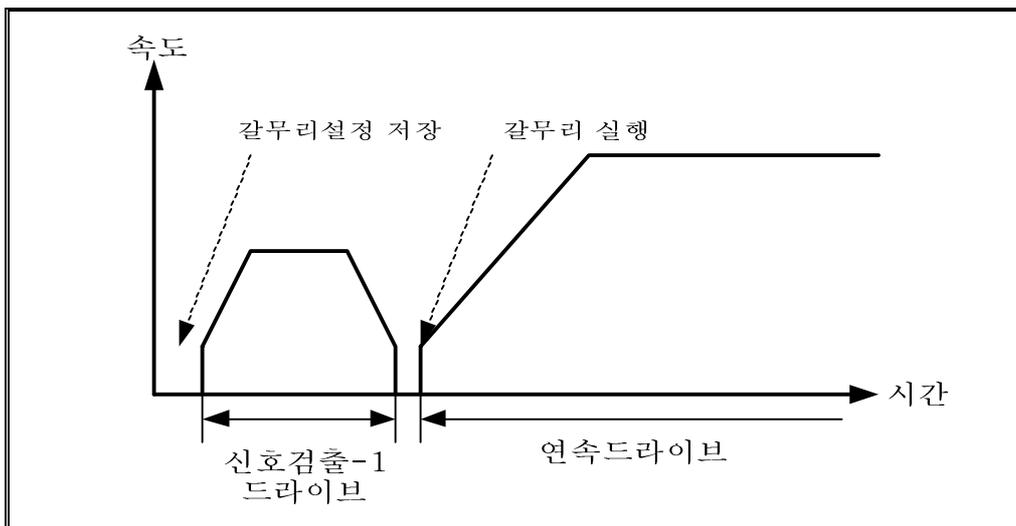


그림 3.23. 갈무리 기능 사용 예

## 4. Address Map

CAMC-FS 칩의 address map은 다음과 같다. 내부에는 8-bit 레지스터가 4개 준비되어 있어서, 이곳에 칩 구동이나 설정 명령 실행에 사용하는 데이터를 저장하여 사용할 수 있으며 내부 레지스터 값을 읽는 경우는 4개의 버퍼에 저장된 32-bit 데이터를 8-bit씩(16-bit data bus access 모드인 경우는 16-bit) 외부 데이터 핀을 통해 데이터를 읽어볼 수 있다.

A2	A1	A0	8/16SEL	CS*	RD*	WR*	Operation	비고
0	0	0	0	0	1	0	Data1 write (2 <sup>31</sup> ~2 <sup>24</sup> bit 데이터)	8bit access
0	0	1	0	0	1	0	Data2 write (2 <sup>23</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	0	0	0	1	0	Data3 write (2 <sup>15</sup> ~2 <sup>8</sup> bit 데이터)	"
0	1	1	0	0	1	0	Data4 write (2 <sup>7</sup> ~2 <sup>0</sup> bit 데이터)	"
1	0	0	0	0	1	0	Command write	"
1	0	1	0	0	1	0	사용하지 않음	"
1	1	0	0	0	1	0	사용하지 않음	"
1	1	1	0	0	1	0	사용하지 않음	"
0	0	0	0	0	0	1	Data1 read (2 <sup>31</sup> ~2 <sup>24</sup> bit 데이터)	"
0	0	1	0	0	0	1	Data2 read (2 <sup>23</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	0	0	0	0	1	Data3 read (2 <sup>15</sup> ~2 <sup>8</sup> bit 데이터)	"
0	1	1	0	0	0	1	Data4 read (2 <sup>7</sup> ~2 <sup>0</sup> bit 데이터)	"
1	0	0	0	0	0	1	사용하지 않음	"
1	0	1	0	0	0	1	사용하지 않음	"
1	1	0	0	0	0	1	사용하지 않음	"
1	1	1	0	0	0	1	사용하지 않음	"
0	0	0	1	0	1	0	Data1,Data2 write (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	16bit access
0	0	1	1	0	1	0	사용하지 않음	
0	1	0	1	0	1	0	Data1,Data2 write (2 <sup>15</sup> ~2 <sup>0</sup> bit 데이터)	"
0	1	1	1	0	1	0	사용하지 않음	
1	0	0	1	0	1	0	Command write	"
1	0	1	1	0	1	0	사용하지 않음	
1	1	0	1	0	1	0	사용하지 않음	
1	1	1	1	0	1	0	사용하지 않음	
0	0	0	1	0	0	1	Data1,Data2 read (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	"
0	0	1	1	0	0	1	사용하지 않음	

A2	A1	A0	8/16SEL	CS*	RD*	WR*	Operation	비고
0	1	0	1	0	0	1	Data1,Data2 read (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	1	1	0	0	1	사용하지 않음	
1	0	0	1	0	0	1	사용하지 않음	
1	0	1	1	0	0	1	사용하지 않음	
1	1	0	1	0	0	1	사용하지 않음	
1	1	1	1	0	0	1	사용하지 않음	

## 5. Port 설명

DATA1~DATA4 port에는 실행 명령에 필요한 데이터를 write한다. 최대 32-bit 데이터까지 저장 가능하며, 저장된 데이터는 명령어 실행할 때 자동으로 로드하여 사용한다. Read 명령을 실행한 후 DATA1~DATA4 port에 읽은 데이터가 저장된다. 사용자는 이 데이터를 8-bit 또는 16-bit씩 읽으면 결과를 알 수 있다.

### 5.1. DATA1 WRITE PORT

각종 설정 DATA의  $2^{31} \sim 2^{24}$  Bit Data를 써 넣는다.

D7	—	$2^{31}$ BIT
D6	—	$2^{30}$ BIT
D5	—	$2^{29}$ BIT
D4	—	$2^{28}$ BIT
D3	—	$2^{27}$ BIT
D2	—	$2^{26}$ BIT
D1	—	$2^{25}$ BIT
D0	—	$2^{24}$ BIT

### 5.2. DATA2 WRITE PORT

각종 설정 DATA의  $2^{23} \sim 2^{16}$  Bit Data를 써 넣는다.

D7	—	$2^{23}$ BIT
D6	—	$2^{22}$ BIT
D5	—	$2^{21}$ BIT
D4	—	$2^{20}$ BIT
D3	—	$2^{19}$ BIT
D2	—	$2^{18}$ BIT
D1	—	$2^{17}$ BIT
D0	—	$2^{16}$ BIT

### 5.3. DATA3 WRITE PORT

각종 설정 DATA의  $2^{15} \sim 2^8$  Bit Data를 써 넣는다.

D7	—	$2^{15}$ BIT
D6	—	$2^{14}$ BIT
D5	—	$2^{13}$ BIT
D4	—	$2^{12}$ BIT
D3	—	$2^{11}$ BIT
D2	—	$2^{10}$ BIT
D1	—	$2^9$ BIT
D0	—	$2^8$ BIT

### 5.4. DATA4 WRITE PORT

각종 설정 DATA의  $2^7 \sim 2^0$  Bit Data를 써 넣는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

### 5.5. COMMAND WRITE PORT

COMMAND를 써 넣는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

## 5.6. DATA1 READ PORT

각종 설정 DATA의  $2^{31} \sim 2^{24}$  Bit Data를 읽는다.

D7	—	$2^{31}$ BIT
D6	—	$2^{30}$ BIT
D5	—	$2^{29}$ BIT
D4	—	$2^{28}$ BIT
D3	—	$2^{27}$ BIT
D2	—	$2^{26}$ BIT
D1	—	$2^{25}$ BIT
D0	—	$2^{24}$ BIT

## 5.7. DATA2 READ PORT

각종 설정 DATA의  $2^{23} \sim 2^{16}$  Bit Data를 읽는다.

D7	—	$2^{23}$ BIT
D6	—	$2^{22}$ BIT
D5	—	$2^{21}$ BIT
D4	—	$2^{20}$ BIT
D3	—	$2^{19}$ BIT
D2	—	$2^{18}$ BIT
D1	—	$2^{17}$ BIT
D0	—	$2^{16}$ BIT

## 5.8. DATA3 READ PORT

각종 설정 DATA의  $2^{15} \sim 2^8$  Bit Data를 읽는다.

D7	—	$2^{15}$ BIT
D6	—	$2^{14}$ BIT
D5	—	$2^{13}$ BIT
D4	—	$2^{12}$ BIT
D3	—	$2^{11}$ BIT
D2	—	$2^{10}$ BIT
D1	—	$2^9$ BIT
D0	—	$2^8$ BIT

## 5.9. DATA4 READ PORT

각종 설정 DATA의  $2^7 \sim 2^0$  Bit Data를 읽는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

## 6. COMMAND 설명

### 6.1. COMMAND 일람표

CAMC-FS칩에는 레지스터 설정 및 +/- 방향 DRIVE 구동 등의 COMMAND가 제공된다. DRIVE COMMAND를 제외한 모든 COMMAND는 DRIVE 중(BUSY = H), 혹은 DRIVE 정지 중(BUSY = L)에 관계없이 상시 실행이 가능하나 DATA의 기입COMMAND에 따라서는 그 이후의 동작이 정상적으로 실행되지 않을 수도 있으므로 주의가 필요하다.

구동 펄스 출력 드라이브에 관련된 레지스터, 즉, PGM(Pulse Generator Module)에 관련된 레지스터는 2개의 बैं크로 구분되어 있다. 따라서 같은 종류의 데이터를 2가지 패턴으로 저장할 수 있다. PGM-1 그룹에 관련되는 레지스터들은 기본적으로 드라이브를 실행할 때 사용하는 레지스터들이고, PGM-1 UP-DATE 그룹에 관련된 레지스터는 PGM-1 그룹의 레지스터와 같은 방법으로 설정한 후 PGM Register Change Command를 실행하여 PGM-1 UP-DATE 그룹의 레지스터를 이용하여 펄스출력 드라이브를 구동할 수 있다. 이와 같은 레지스터 구조는 두 가지의 속도 패턴을 저장한 후, 드라이브 도중에 사용자가 계속적으로 레지스터 설정을 변경하지 않고도 드라이브 시작 명령만을 이용하여 두 가지 패턴의 속도로 모터를 구동할 수 있다.

CAMC-FS의 실행 명령은 PGM-1 그룹, PGM-1 UP-DATE 그룹, 확장 그룹, 스크립트/갈무리 그룹, BUS-1 그룹, BUS-2 그룹으로 구분된다.

PGM-1 그룹과 PGM-1 UP-DATE 그룹 COMMAND는 펄스 출력에 관련된 데이터 설정에 관련된 명령들이다.

스크립트/갈무리 그룹 COMMAND는 스크립트와 갈무리 기능을 사용하기 위해 저장할 조건, 명령, 데이터 등을 설정하는 명령들이다.

BUS-1 그룹 COMMAND는 내부/외부 카운터 관련 데이터 설정, 외부 센서 설정 및 PGM Register Change 명령들이 포함된다.

BUS-2 그룹 COMMAND는 칩 구동 이외의 부가적인 기능의 설정에 관련된 명령들이다.

다음의 표에서 전체 COMMAND의 일람표를 표시한다.

▶ **CAMC-FS READ COMMAND**

CODE(HEX)	이 름 (PGM-1 GROUP 레지스터 READ)	R	DEFAULT
00	PGM-1 RANGE READ	16	0xFFFF
01	PGM-1 START/STOP SPEED DATA READ	16	0x0001
02	PGM-1 OBJECT SPEED DATA READ	16	0x0001
03	PGM-1 RATE-1 DATA READ	16	0xFFFF
04	PGM-1 RATE-2 DATA READ	16	0xFFFF
05	PGM-1 RATE-3 DATA READ	16	0xFFFF
06	PGM-1 RATE CHANGE POINT 1-2 READ	16	0xFFFF
07	PGM-1 RATE CHANGE POINT 2-3 READ	16	0xFFFF
08	PGM-1 SW-1 DATA READ	15	0X7FFF
09	PGM-1 SW-2 DATA READ	15	0X7FFF
0A	PGM-1 PWM 출력 설정 DATA READ(0~6)	3	0X00
0B	PGM-1 SLOW DOWN / REAR PULSE READ	32	0X00000000
0C	PGM-1 현재 SPEED DATA READ	16	0x0000
0D	PGM-1 현재 SPEED 비교 DATA READ	16	0x0000
0E	PGM-1 DRIVE PULSE COUNTER READ	32	0X00000000
0F	PGM-1 PRESET PULSE DATA READ	32	0X00000000

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 READ)	R	DEFAULT
10	PGM-1 UP-DATE RANGE READ	16	0xFFFF
11	PGM-1 UP-DATE START/STOP SPEED DATA READ	16	0x0001
12	PGM-1 UP-DATE OBJECT SPEED DATA READ	16	0x0001
13	PGM-1 UP-DATE RATE-1 DATA READ	16	0xFFFF
14	PGM-1 UP-DATE RATE-2 DATA READ	16	0xFFFF
15	PGM-1 UP-DATE RATE-3 DATA READ	16	0xFFFF
16	PGM-1 UP-DATE RATE CHANGE POINT 1-2 READ	16	0xFFFF
17	PGM-1 UP-DATE RATE CHANGE POINT 2-3 READ	16	0xFFFF
18	PGM-1 UP-DATE SW-1 DATA READ	15	0X7FFF
19	PGM-1 UP-DATE SW-2 DATA READ	15	0X7FFF
1A	NO OPERATION		
1B	PGM-1 UP-DATE SLOW DOWN / REAR PULSE READ	32	0X00000000

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 READ)	R	DEFAULT
1C	PGM-1 현재 SPEED DATA READ	16	0x0000
1D	PGM-1 현재 SPEED 비교 DATA READ	16	0x0000
1E	PGM-1 DRIVE PULSE COUNTER READ	32	0X00000000
1F	PGM-1 PRESET PULSE DATA READ	32	0X00000000

CODE(HEX)	이 름 (PGM-2 GROUP 레지스터 READ)	R	DEFULT
20~2C	NO OPERATION		
2D	드라이브 동작 설정 DATA READ 2bit : 실거리 연산기능(0: 사용안함, 1:사용) 0~1bit : 00: 직선 가감속 모드 01: 비대칭 직선 가감속 모드 10: 대칭 S-곡선 가감속 모드 11: 비대칭 S-곡선 가감속 모드	3	0x0
2E	MPG OPERATION SETTING DATA WRITE 6bit : 외부 신호에 의한 Continuous drive 구동 (0: RESET, 1: SET) 5bit : 외부 신호에 의한 Preset pulse drive 구동 (0: RESET, 1: SET) 3~4bit : 00: 단상 01: 1체배 10: 2체배 11: 4체배 2bit : 0: 사용자 설정 방향 + 1: 사용자 설정 방향 - 1bit : 0: 외부 2상 펄스방향 출력 1: 사용자 설정 방향 출력 0bit : 외부 2상 펄스에 의한 구동(0: RESET, 1: SET)	7	0x00
2F	MPG PRESET PULSE DATA READ	32	0x00000000

CODE(HEX)	이 름 ( 확장 GROUP 레지스터 READ )	R	DEFULT
30~35	NO OPERATION		

CODE(HEX)	이름 ( 확장 GROUP 레지스터 READ )	R	DEFULT
36	Soft limit 설정 READ 2 bit : Limit 비교 데이터 설정 (0 : Internal count, 1 : External count) 1 bit : Soft limit 시 동작 드라이브(0: 급정지, 1: 감속정지) 0 bit : Soft limit Enable/Disable(0 : Enabel, 1 : Disable)	3	0x0
37	- Soft limit 비교 레지스터 설정 READ	32	0x80000000
38	+ Soft limit 비교 레지스터 설정 READ	32	0x7FFFFFFF
39	Trigger mode 설정 READ 31~16 bit : Trigger active level 출력시간 설정 (Trigger 기능 초기화 : 0x0000 로 설정) 시간 = main clock 주기 x (Decimal value of 31~16bit) 15~ 2 bit : Don't care 1 bit : Trigger 비교 count 지정(0 : Internal, 1 : External) 0 bit : Trigger mode(0 : 위치 모드, 1 : 주기 모드)	32	0x00010000
3A	Trigger비교 데이터 설정 READ	32	0x00000000
3B	INTERNAL COUNTER M_DATA READ	32	0x80000000
3C	EXTERNAL COUNTER M_DATA READ	32	0x80000000
3D~3F	NO OPERATION		

CODE(HEX)	이름 (스트립터 GROUP 레지스터 READ)	R	DEFULT
40	스크립트 동작 설정 레지스터-1 READ 31 bit : 0 : 한번 실행, 1 : 계속 실행 30~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND *검출 이벤트 설정 별첨	32	0x00000000
41	스크립트 동작 설정 레지스터-2 READ	32	0x00000000

CODE(HEX)	이름 (스크립터 GROUP 레지스터 READ)	R	DEFULT
	스크립트 동작 설정 레지스터-1과 bit 할당 동일		
42	스크립트 동작 설정 레지스터-3 READ 스크립트 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
43	스크립트 동작 설정 레지스터-Queue READ 31 bit : 0: 한번 실행, 1: 계속 실행 30 bit : 실행시 Interrupt 발생(0:Disable, 1:Enable) 29~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00: 논리연산 없음(1번 이벤트만 검출) 01: OR 연산 10: AND 연산 11: XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND	32	0x00000000
44	스크립트 동작 데이터 레지스터-1 READ	32	0x00000000
45	스크립트 동작 데이터 레지스터-2 READ	32	0x00000000
46	스크립트 동작 데이터 레지스터-3 READ	32	0x00000000
47	스크립트 동작 데이터 레지스터-Queue READ	32	0x00000000
48	NO OPERATION		
49	스크립트 동작 설정 Queue 인덱스 READ	4	0x0
4A	스크립트 동작 데이터 Queue 인덱스 READ	4	0x0
4B	스크립트 Queue Full/Empty Flag READ 3 bit : 스크립트 동작 설정 레지스터_4 Full Flag 2 bit : 스크립트 동작 설정 레지스터_4 Empty Flag 1 bit : 스크립트 동작 데이터 레지스터_4 Full Flag 0 bit : 스크립트 동작 데이터 레지스터_4 Empty Flag	4	0x5
4C	스크립트 Queue size 설정(0~13) READ 15~ 12 bit : 동작 설정 레지스터-Queue full size 11~ 8 bit : 동작 설정 레지스터-Queue empty size 7~ 4 bit : 동작 데이터 레지스터-Queue full size 3~ 0 bit : 동작 데이터 레지스터-Queue empty size	16	0xD0D0
4D	스크립트 Queue status READ 11 ~ 8 bit : 스크립트 동작 설정 Queue 인덱스	12	0x005

CODE(HEX)	이름 (스크립트 GROUP레지스터 READ)	R	DEFULT
	7~4 bit : 스크립트 동작 데이터 Queue 인덱스 3 bit : 스크립트 동작 설정 레지스터_4 Full Flag 2 bit : 스크립트 동작 설정 레지스터_4 Empty Flag 1 bit : 스크립트 동작 데이터 레지스터_4 Full Flag 0 bit : 스크립트 동작 데이터 레지스터_4 Empty Flag		
4E~4F	NO OPERATION		

CODE(HEX)	이름 (갈무리 GROUP레지스터 READ)	R	DEFAULT
50	갈무리 동작 설정 레지스터-1 READ 갈무리 동작 설정 레지스터-Queue READ 31 bit : 0 : 한번 실행, 1 : 계속 실행 30 bit : 실행시 Interrupt 발생(0:Disable, 1:Enable) 29~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND	32	0x00000000
51	갈무리 동작 설정 레지스터-2 READ 갈무리 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
52	갈무리 동작 설정 레지스터-3 READ 갈무리 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
53	갈무리 동작 설정 레지스터-Queue READ 31 bit : 0 : 한번 실행, 1 : 계속 실행 30 bit : 실행시 Interrupt 발생(0:Disable, 1:Enable) 29~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산	32	0x00000000

CODE(HEX)	이름 (갈무리 GROUP 레지스터 READ)	R	DEFAULT
	10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND		
54	갈무리 동작 데이터 레지스터-1 READ	32	0x00000000
55	갈무리 동작 데이터 레지스터-2 READ	32	0x00000000
56	갈무리 동작 데이터 레지스터-3 READ	32	0x00000000
57	갈무리 동작 데이터 레지스터-Queue READ	32	0x00000000
58	NO OPERATION		
59	갈무리 동작 설정 Queue 인덱스 READ	4	0x0
5A	갈무리 동작 데이터 Queue 인덱스 READ	4	0x0
5B	갈무리 Queue Full/Empty Flag READ 6 bit : 갈무리 동작 데이터 레지스터_3 data update Flag 5 bit : 갈무리 동작 데이터 레지스터_2 data update Flag 4 bit : 갈무리 동작 데이터 레지스터_1 data update Flag 3 bit : 갈무리 동작 설정 레지스터_4 Full Flag 2 bit : 갈무리 동작 설정 레지스터_4 Empty Flag 1 bit : 갈무리 동작 데이터 레지스터_4 Full Flag 0 bit : 갈무리 동작 데이터 레지스터_4 Empty Flag	7	0x05
5C	갈무리 Queue size 설정(0~13) READ 15~ 12 bit : 동작 설정 레지스터-Queue full size 11~ 8 bit : 동작 설정 레지스터-Queue empty size 7~ 4 bit : 동작 데이터 레지스터-Queue full size 3~ 0 bit : 동작 데이터 레지스터-Queue empty size	16	0xD0D0
5D	캡션Queue status READ 11 ~ 8 bit : 갈무리 동작 설정 Queue 인덱스 7 ~ 4 bit : 갈무리 동작 데이터 Queue 인덱스 3 bit : 갈무리 동작 설정 레지스터_4 Full Flag 2 bit : 갈무리 동작 설정 레지스터_4 Empty Flag 1 bit : 갈무리 동작 데이터 레지스터_4 Full Flag 0 bit : 갈무리 동작 데이터 레지스터_4 Empty Flag	12	0x005
5E~5F	NO OPERATION		

CODE(HEX)	이름 (BUS-1 GROUP 레지스터 READ)	R	DEFAULT
60	INTERNAL COUNTER DATA READ(Signed)	32	0x00000000
61	INTERNAL COUNTER COMPARATE DATA READ (Signed)	32	0x00000000
62	INTERNAL COUNTER PRE-SCALE DATA READ	8	0x00
63	INTERNAL COUNTER P-DATA READ	32	0x7FFFFFFF
64	EXTERNAL COUNTER DATA READ(Signed)	32	0x00000000
65	EXTERNAL COUNTER COMPARATE DATA READ (Signed)	32	0x00000000
66	EXTERNAL COUNTER PRE-SCALE DATA READ	8	0x00
67	EXTERNAL COUNTER P-DATA READ	32	0x7FFFFFFF
68	EXTERNAL SPEED DATA READ	32	0x00000000
69	EXTERNAL SPEED COMPARATE DATA READ	32	0x00000000
6A	외부 센서 필터 대역폭 설정 DATA READ	8	0x05
6B	OFF-RANGE DATA READ	8	0x00
6C	DEVIATION DATA READ 15 bit : 편차량의 부호 14 ~ 0 bit : 편차량의 절대치	16	0x0000
6D	PGM REGISTER CHANGE DATA READ	1	0x0
6E~6F	NO OPERATION		

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
70	칩 기능 설정 DATA READ 12bit: S profile 감속정지시 삼각구동 사용 여부 (0 : 안함, 1 : 사용) 11bit: Search drive source filter 사용 여부(0 : 안함, 1 : 사용) 10bit: Sync 사용 여부(0 : 기능 사용 안함, 1 : 사용) 9 bit : Limit 완전정지(0 : 기능 사용 안함, 1 : 사용) 8 bit : Inposition 대기 확장모드(0 : 기능 사용 안함, 1 : 사용) 7 bit : 탈조 급정지/감속정지 설정 (0 : 급정지, 1 : 감속정지) 6 bit : 감속정지설정(0 : 정지속도에서 정지, 1 : 등속) 5 bit : Limit 감속정지 모드(0 : 기능 사용 안함, 1 : 사용) 4 bit : Limit 급정지 모드(0 : 기능 사용 안함, 1 : 사용) 3 bit : estop, sstop enable/disable(0 : disable, 1 : enable)	13	0x0C3E

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	2 bit : Don't care 1 bit : Alarm Stop 모드(0 : 사용 안함, 1 : 사용) 0 bit : Inposition 대기(0 : 사용 안함, 1 : 사용)		
71	MODE1 DATA READ 7 bit : 감속 시작 포인트 검출 방식 (0 : 자동, 1 : 나머지) 6~4 bit : 펄스 출력 방식 설정(* 별첨 테이블 참조) 3~0 bit : 신호검출 선택 설정(* 별첨 테이블 참조)	8	0x00
72	MODE2 DATA READ 10 bit : Trigger 출력신호의 active level 설정 9 bit : Interrupt active level 설정 8 bit : MARK 신호 active level 설정 7~6 bit : Encoder signal count mode(ECUP, ECDN signal) 00 : 단상 신호(ECUP : up 입력, ECDN : down 입력) 01 : 90° difference phase, 1 times multiplied 10 : 90° difference phase, 2 times multiplied 11 : 90° difference phase, 4 times multiplied 5 bit : INP 신호 ACTIVE LEVEL 설정 4 bit : ALM 신호 ACTIVE LEVEL 설정 3 bit : -SLM 신호 ACTIVE LEVEL 설정 2 bit : +SLM 신호 ACTIVE LEVEL 설정 1 bit : -ELM 신호 ACTIVE LEVEL 설정 0 bit : +ELM 신호 ACTIVE LEVEL 설정	11	0x200
73	UNIVERSAL IN READ 10~8 bit : Universal output bit operation 0xx : 연산데이터 값을 UOUT으로 출력 100 : NOT (현재 UOUT의 반전) 101 : AND (연산데이터 값과 현재 UOUT의 and 연산 출력) 110 : OR (연산데이터 값과 현재 UOUT의 or 연산 출력) 111 : XOR (연산데이터 값과 현재 UOUT의 difference check) 7 ~ 4 bit : Universal Input 3 ~ 0 bit : Universal output	11	0X0000
74	END STATUS DATA READ 14 bit : Limit(PELM, NELM, PSLM, PELM, soft)에 의한 종료	15	0x0000

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	13 bit : Limit 완전정지에 의한 종료 12 bit : Sensor positioning drive 종료 11 bit : Preset pulse drive 종료 10 bit : 신호 검출에 의한 종료(signal search-1/2 drive 종료) 9 bit : 원점 검출에 의한 종료 8 bit : 탈조 에러에 의한 종료 7 bit : 데이터 설정 에러에 의한 종료 6 bit : ALARM 신호 입력에 의한 종료 5 bit : 급정지 명령에 의한 종료 4 bit : 감속 정지 명령에 의한 종료 3 bit : 급정지 신호 입력에 의한 종료 2 bit : 감속 정지 신호 입력에 의한 종료 1 bit : Limit(PELM, NELM, soft) 급정지에 의한 종료 0 bit : Limit(PSLM, NSLM, soft) 감속정지 의한 종료		
75	MECHANICAL SIGNAL DATA READ 12 bit : ESTOP 신호 입력 Level 11 bit : SSTOP 신호 입력 Level 10 bit : MARK# 신호 입력 Level 9 bit : EXPP 신호 입력 Level 8 bit : EXMP 신호 입력 Level 7 bit : 엔코더 UP 신호 입력 Level ( A 상 신호 ) 6 bit : 엔코더 DOWN 신호 입력 Level ( B 상 신호 ) 5 bit : INPOSITION 신호 active 상태 4 bit : ALARM 신호 active 상태 3 bit : -Limit 감속정지 신호 active 상태 2 bit : +Limit 감속정지 신호 active 상태 1 bit : -Limit 급정지 신호 active 상태 0 bit : +Limit 급정지 신호 active 상태	13	
76	DRIVE STATE DATA READ 8 bit : 드라이브 방향 신호(0:CW, 1:CCW) 7 bit : ECG(외부 위치 카운터 비교값 < 외부 위치 카운터값) 6 bit : ECL(외부 위치 카운터 비교값 > 외부 위치 카운터값) 5 bit : ICG(내부 위치 카운터 비교값 < 내부 위치 카운터값) 4 bit : ICL(내부 위치 카운터 비교값 > 내부 위치 카운터값)	9	

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	3 bit : UP ( 가속 중 ) 2 bit : CONST ( 등속 중 ) 1 bit : DOWN ( 감속 중 ) 0 bit : BUSY ( 드라이브 구동 중 )		
77	EXTERNAL COUNTER 설정 DATA READ 8 bit : External count clear flag(Read only) 7 bit : 0 : Not reverse count, 1: Reverse count 6 bit : Don't care 5 ~ 4 bit : 00 : CLEAR REQUEST RESET 01 : ONE TIME CLEAR REQUEST SET 10 : FULL TIME CLEAR REQUEST SET 11 : EXTERNAL COUNTER CLEAR SET 3 ~ 0 bit : CLEAR 입력신호 선택 0000 : IN0 0010 : IN1 0100 : IN2 1000 : IN3 Others : Don't care	9	0x00
79	Interrupt FLAG READ (0 : Interrupt inactivated, 1 : Interrupt activated) 31 bit : Selectable Interrupt source3("FE"의 31~24bit) 30 bit : Selectable Interrupt source2("FE"의 23~16bit) 29 bit : Selectable Interrupt source1("FE"의 15~8bit) 28 bit : Selectable Interrupt source0("FE"의 7~0bit) 27 bit : Emergency limit 신호 입력시 26 bit : Slow down limit 신호 입력시 25 bit : 캡션 queue 명령어 수행시(30 bit '1' 일때) 24 bit : 스크립트 queue 명령어 수행시(30 bit '1' 일때) 23 bit : M-data count clear 22 bit : 현재속도 데이터 = 현재 속도 비교데이터 21 bit : 현재속도 데이터 = RCP12 20 bit : 현재속도 데이터 = RCP23 19 bit : MODE1 register에 설정된 신호 검출시 18 bit : 가속	32	0x00000000

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	17 bit : 등속 16 bit : 감속 15 bit : Interrupt command에 의한 생성 14 bit : 캡션-3에 저장된 command가 실행될 때 13 bit : 캡션-2에 저장된 command가 실행될 때 12 bit : 캡션-1에 저장된 command가 실행될 때 11 bit : 스크립트-3에 저장된 command가 실행될 때 10 bit : 스크립트-2에 저장된 command가 실행될 때 9 bit : 스크립트-1에 저장된 command가 실행될 때 8 bit : 캡션 동작 설정 queue가 full될 때 7 bit : 스크립트 data queue가 empty될 때 6 bit : External counter < External comparator가 될 때 5 bit : External counter = External comparator가 될 때 4 bit : External counter > External comparator가 될 때 3 bit : Internal counter < Internal comparator가 될 때 2 bit : Internal counter = Internal comparator가 될 때 1 bit : Internal counter > Internal comparator가 될 때 0 bit : 드라이브 종료		
7A	Interrupt MASK READ (0 : Interrupt Disable, 1 : Enable) 31 bit : Selectable Interrupt source3("FE"의 31~24bit) 30 bit : Selectable Interrupt source2("FE"의 23~16bit) 29 bit : Selectable Interrupt source1("FE"의 15~8bit) 28 bit : Selectable Interrupt source0("FE"의 7~0bit) 27 bit : Emergency limit 신호 입력시 26 bit : Slow down limit 신호 입력시 25 bit : 캡션 queue 명령어 수행시(30 bit '1' 일때) 24 bit : 스크립트 queue 명령어 수행시(30 bit '1' 일때) 23 bit : P-data count clear 22 bit : 현재속도 데이터 = 현재 속도 비교데이터 21 bit : 현재속도 데이터 = RCP12 20 bit : 현재속도 데이터 = RCP23 19 bit : MODE1 register에 설정된 신호 검출시 18 bit : 가속	32	0x00000001

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	17 bit : 등속 16 bit : 감속 15 bit : Don't care 14 bit : 캡션-3에 저장된 command가 실행될 때 13 bit : 캡션-2에 저장된 command가 실행될 때 12 bit : 캡션-1에 저장된 command가 실행될 때 11 bit : 스크립트-3에 저장된 command가 실행될 때 10 bit : 스크립트-2에 저장된 command가 실행될 때 9 bit : 스크립트-1에 저장된 command가 실행될 때 8 bit : 캡션 동작 설정 queue가 full될 때 7 bit : 스크립트 data queue가 empty될 때 6 bit : External counter < External comparator가 될 때 5 bit : External counter = External comparator가 될 때 4 bit : External counter > External comparator가 될 때 3 bit : Internal counter < Internal comparator가 될 때 2 bit : Internal counter = Internal comparator가 될 때 1 bit : Internal counter > Internal comparator가 될 때 0 bit : 드라이브 종료		
7B	EMODE1 DATA READ 7 bit : Extension mode enable(0 : Disable, 1 : Enable) 0 : md_out[12:8] => md_out[12:8] 1 : md_out[12:8] => EUIO 6~ 5 bit : Don't care 4 bit : EUO4 enable/disable(0 : disable, 1 : enable) 3 bit : EUO3 enable/disable(0 : disable, 1 : enable) 2 bit : EUO2 enable/disable(0 : disable, 1 : enable) 1 bit : EUO1 enable/disable(0 : disable, 1 : enable) 0 bit : EUIO0 direction select(0 : input, 1 : output)	8	0x00
7C	Extension UNIVERSAL OUT READ 12~8 bit : Extension Universal output 연산 데이터(not Visible) 7 ~ 5 bit : Extension Universal output bit operation 0xx : 연산데이터 값을 OUT으로 출력 100 : NOT(현재 OUT의 반전) 101 : AND(연산데이터 값과 현재 OUT의 and 연산 출력)	8	0x00

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	R	DEFAULT
	110 : OR(연산데이터 값과 현재 OUT의 or 연산 출력) 111 : XOR(연산데이터 값과 현재 OUT의 difference check) 4 ~ 0 bit : Extension Universal in/output		
78,7D,7F	NO OPERATION		
7E	USER Interrupt source selection register READ 31~24 bit : User selectable interrupt 3 source selection 스크립트, 캡션 검출이벤트 리스트와 동일 23~16 bit : User selectable interrupt 2 source selection 스크립트, 캡션 검출이벤트 리스트와 동일 15~ 8 bit : User selectable interrupt 1 source selection 스크립트, 캡션 검출이벤트 리스트와 동일 7 ~ 0 bit : User selectable interrupt 0 source selection 스크립트, 캡션 검출이벤트 리스트와 동일	32	0x00000000

## ▶ CAMC-FS WRITE COMMAND

CODE(HEX)	이름 (PGM-1 GROUP 레지스터 WRITE)	W	DEFAULT
80	PGM-1 RANGE WRITE	16	0xFFFF
81	PGM-1 START/STOP SPEED DATA WRITE	16	0X0001
82	PGM-1 OBJECT SPEED DATA WRITE	16	0X0001
83	PGM-1 RATE-1 DATA WRITE	16	0xFFFF
84	PGM-1 RATE-2 DATA WRITE	16	0xFFFF
85	PGM-1 RATE-3 DATA WRITE	16	0xFFFF
86	PGM-1 RATE CHANGE POINT 1-2 WRITE	16	0xFFFF
87	PGM-1 RATE CHANGE POINT 2-3 WRITE	16	0xFFFF
88	PGM-1 SW-1 DATA WRITE	15	0X7FFF
89	PGM-1 SW-2 DATA WRITE1	15	0X7FFF
8A	PGM-1 PWM 출력 설정 DATA WRITE(0~6)	3	0x00
8B	PGM-1 SLOW DOWN / REAR PULSE WRITE	32	0X00000000
8C	NO OPERATION		
8D	PGM-1 현재 SPEED 비교 DATA WRITE	16	0x0000
8E	NO OPERATION		
8F	NO OPERATION		

CODE(HEX)	이름 (PGM-1 UP-DATE GROUP 레지스터 WRITE)	W	DEFAULT
90	PGM-1 UP-DATE RANGE WRITE	16	0xFFFF
91	PGM-1 UP-DATE START/STOP SPEED DATA WRITE	16	0X0001
92	PGM-1 UP-DATE OBJECT SPEED DATA WRITE	16	0X0001
93	PGM-1 UP-DATE RATE-1 DATA WRITE	16	0xFFFF
94	PGM-1 UP-DATE RATE-2 DATA WRITE	16	0xFFFF
95	PGM-1 UP-DATE RATE-3 DATA WRITE	16	0xFFFF
96	PGM-1 UP-DATE RATE CHANGE POINT 1-2 WRITE	16	0xFFFF
97	PGM-1 UP-DATE RATE CHANGE POINT 2-3 WRITE	16	0xFFFF
98	PGM-1 UP-DATE SW-1 DATA WRITE	15	0X7FFF
99	PGM-1 UP-DATE SW-2 DATA WRITE	15	0X7FFF
9A	NO OPERATION		
9B	PGM-1 UP-DATE SLOW DOWN / REAR PULSE WRITE	32	0X00000000

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 WRITE)	W	DEFAULT
9C	NO OPERATION		
9D	PGM-1 현재 SPEED 비교 DATA WRITE	16	0x0000
9E	NO OPERATION		
9F	NO OPERATION		

CODE(HEX)	이 름 (PGM-2 GROUP 레지스터 WRITE)	W	DEFAULT
A0	+ PRESET PULSE DRIVE	32	
A1	+ CONTINUOUS DRIVE		
A2	+ SIGNAL SEARCH - 1 DRIVE		
A3	+ SIGNAL SEARCH - 2 DRIVE		
A4	+ ORIGIN (원 점) SEARCH DRIVE	1	
A5	- PRESET PULSE DRIVE	32	
A6	- CONTINUOUS DRIVE		
A7	- SIGNAL SEARCH - 1 DRIVE		
A8	- SIGNAL SEARCH - 2 DRIVE		
A9	- ORIGIN (원 점) SEARCH DRIVE	1	
AA	PRESET PULSE DATA OVERRIDE(ON_BUSY)	32	
AB	SLOW DOWN STOP		
AC	EMERGENCY STOP		
AD	DRIVE MODE SETTING DATA WRITE 2bit : 실거리 연산기능(0: 사용안함, 1:사용) 1~0bit : 00 : 직선 가감속 모드 01 : 비대칭 직선 가감속 모드 10 : 대칭 S-곡선 가감속 모드 11 : 비대칭 S-곡선 가감속 모드	3	0x0
AE	MPG OPERATION SETTING DATA WRITE 6bit : 외부 신호에 의한 Continuous drive 구동 (0 : RESET, 1 : SET) 5bit : 외부 신호에 의한 Preset pulse drive 구동 (0 : RESET, 1 : SET) 4~3bit : 외부 입력 신호 체배 모드 (00 : 단상, 01 : 1체배, 10 : 2체배, 11 : 4체배) 2bit : 0 : 사용자 설정 방향 +	7	0X00

CODE(HEX)	이름 (PGM-2 GROUP 레지스터 WRITE)	W	DEFAULT
	1 : 사용자 설정 방향 - 1bit : 0 : 외부 2상 펄스방향 출력 1 : 사용자 설정 방향 출력 0bit : 외부 2상 펄스에 의한 구동(0 : RESET, 1 : SET)		
AF	MPG PRESET PULSE DATA WRITE	32	0X0000000

CODE(HEX)	이름 (확장 GROUP 레지스터 WRITE)	W	DEFAULT
B0	+ Sensor positioning drive I	32	
B1	- Sensor positioning drive I	32	
B2	+ Sensor positioning drive II	32	
B3	- Sensor positioning drive II	32	
B4	+ Sensor positioning drive III	32	
B5	- Sensor positioning drive III	32	
B6	Soft limit 설정 2 bit : Limit 비교 데이터 설정 (0 : Internal count, 1 : External count) 1 bit : Soft limit 시 동작 드라이브 (0 : 급정지, 1 : 감속정지) 0 bit : Soft limit Enable/Disable (0 : Enabel, 1 : Disable)	3	0x0
B7	- Soft limit 비교 레지스터 설정	32	0x80000000
B8	+ Soft limit 상한 비교 레지스터 설정	32	0x7FFFFFFF
B9	Trigger mode 설정 31~16 bit : Trigger active level 설정 1 bit : Trigger 비교 count 설정 (0:Internal, 1 : External) 0 bit : Trigger mode (0 : 위치 모드, 1 : 주기 모드)	32	0x00010000
BA	Trigger비교 데이터 설정	32	0x00000000
BB	Internal M-data 설정	32	0x80000000
BC	External M-data 설정	32	0x80000000
BD ~ BF	NO OPERATION		

CODE(HEX)	이름 (스트립터 GROUP 레지스터 WRITE)	W	DEFAULT
-----------	----------------------------	---	---------

CODE(HEX)	이름 (스크립트 GROUP 레지스터 WRITE)	W	DEFAULT
C0	스크립트 동작 설정 레지스터-1 WRITE 31 bit : 0: 한번 실행, 1: 계속 실행 30~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND *검출 이벤트 설정 별첨	32	0x00000000
C1	스크립트 동작 설정 레지스터-2 WRITE 스크립트 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
C2	스크립트 동작 설정 레지스터-3 WRITE 스크립트 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
C3	스크립트 동작 설정 레지스터-Queue WRITE 31 bit : 0: 한번 실행, 1: 계속 실행 30 bit : 실행시 Interrupt 발생(0:Disable, 1:Enable) 29~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND	32	0x00000000
C4	스크립트 동작 데이터 레지스터-1 WRITE	32	0x00000000
C5	스크립트 동작 데이터 레지스터-2 WRITE	32	0x00000000
C6	스크립트 동작 데이터 레지스터-3 WRITE	32	0x00000000
C7	스크립트 동작 데이터 레지스터-Queue WRITE	32	0x00000000
C8	스크립트 Queue clear		
CC	스크립트 Queue size 설정(0~13)	16	0xD0D0

CODE(HEX)	이름 (스트리퍼 GROUP 레지스터 WRITE)	W	DEFAULT
	15~ 12bit : 동작 설정 레지스터-Queue full size 11~ 8 bit : 동작 설정 레지스터-Queue empty size 7~ 4 bit : 동작 데이터 레지스터-Queue full size 3~ 0 bit : 동작 데이터 레지스터-Queue empty size		
C9~CB CD~CF	NO OPERATION		

CODE(HEX)	이름 (갈무리 GROUP 레지스터 WRITE)	W	DEFAULT
D0	갈무리 동작 설정 레지스터-1 WRITE 갈무리 동작 설정 레지스터-Queue WRITE 31 bit : 0 : 한번 실행, 1 : 계속 실행 30~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산 23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND	32	0x00000000
D1	갈무리 동작 설정 레지스터-2 WRITE 갈무리 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
D2	갈무리 동작 설정 레지스터-3 WRITE 갈무리 동작 설정 레지스터-1과 bit 할당 동일	32	0x00000000
D3	갈무리 동작 설정 레지스터-Queue WRITE 31 bit : 0 : 한번 실행, 1 : 계속 실행 30 bit : 실행시 Interrupt 발생(0:Disable, 1:Enable) 29~26bit : Don't care 25~24bit : 검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산	32	0x00000000

CODE(HEX)	이름 (갈무리 GROUP 레지스터 WRITE)	W	DEFAULT
	23~16bit : 2번 검출 이벤트 설정 15~ 8 bit : 1 번 검출 이벤트 설정 7~ 0 bit : 이벤트 검출 후 실행할 COMMAND		
D8	갈무리 Queue clear		
DC	갈무리 Queue size 설정(0~13) 15~ 12bit : 동작 설정 레지스터-Queue full size 11~ 8 bit : 동작 설정 레지스터-Queue empty size 7~ 4 bit : 동작 데이터 레지스터-Queue full size 3~ 0 bit : 동작 데이터 레지스터-Queue empty size	16	0xD0D0
D4~D7 D9~DB DD~DF	NO OPERATION		

CODE(HEX)	이름 (BUS-1 GROUP 레지스터 WRITE)	W	DEFAULT
E0	INTERNAL COUNTER DATA WRITE	32	0x00000000
E1	INTERNAL COUNTER COMPARATE DATA WRITE (Signed)	32	0x00000000
E2	INTERNAL COUNTER PRE-SCALE DATA WRITE	8	0x00
E3	INTERNAL COUNTER P- DATA WRITE	32	0x7FFFFFFF
E4	EXTERNAL COUNTER DATA WRITE	32	0x00000000
E5	EXTERNAL COUNTER COMPARATE DATA WRITE (Signed)	32	0x00000000
E6	EXTERNAL COUNTER PRE-SCALE DATA WRITE	8	0x00
E7	EXTERNAL COUNTER P- DATA WRITE	32	0x7FFFFFFF
E8	EXTERNAL SPEED DATA WRITE	32	0x00000000
E9	EXTERNAL SPEED COMPARATE DATA WRITE	32	0x00000000
EA	외부 센서 필터 대역폭 설정 DATA WRITE	8	0x05
EB	OFF-RANGE DATA WRITE	8	0x00
EC	NO OPERATION		
ED	PGM REGISTER CHANGE DATA WRITE	1	0x0
EE	COMPARE REGISTER INPUT CHANGE	2	0X00

CODE(HEX)	이 름 (BUS-1 GROUP 레지스터 WRITE)	W	DEFAULT
	1~0 bit : 00 : 1'st : Internal count, 2'nd : External count 01 : 1'st : Internal count, 2'nd : Internal count 10 : 1'st : External count, 2'nd : External count		
EF	NO OPERATION		

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
F0	칩 기능 설정 DATA WRITE 12bit : S profile 감속정지시 삼각구동 사용 여부 (0 : 안함, 1 : 사용) 11bit : Search source filter 사용 여부(0 : 안함, 1 : 사용) 10bit : Sync 사용 여부(0 : 기능 사용 안함, 1 : 사용) 9 bit : Limit 완전정지(0 : 기능 사용 안함, 사용) 8 bit : Inposition 대기 확장모드(0 : 기능 사용 안함, 1 : 사용) 7 bit : 탈조 급정지/감속정지 설정 (0 : 급정지, 1 : 감속정지) 6 bit : 감속정지설정(0 : 정지속도에서 정지, 1 : 등속) 5 bit : Limit 감속정지 모드(0 : 기능 사용 안함, 1 : 사용) 4 bit : Limit 급정지 모드(0 : 기능 사용 안함, 1 : 사용) 3 bit : estop, sstop enable/disable(0 : disable, 1 : enable) 2 bit : Don't care 1 bit : Alarm Stop 모드(0 : 사용 안함, 1 : 사용) 0 bit : Inposition 대기(0 : 사용 안함, 1 : 사용)	13	0x0C3E
F1	MODE1 DATA WRITE 7 bit : 감속 시작 포인트 검출 방식 (0 : 자동, 1 : 나머지) 6~4 bit : 펄스 출력 방식 설정(* 별첨 테이블 참조) 3~0 bit : 신호검출 선택 설정(* 별첨 테이블 참조)	8	0x00
F2	MODE2 DATA WRITE 10 bit : Trigger 신호의 active level 설정 9 bit : Interrupt active level 설정 8 bit : MARK 신호 active level 설정 7~6 bit : Encoder signal count mode(ECUP, ECDN signal)	11	0x200

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
	00 : 단상 신호(ECUP : up 입력, ECDN : down 입력) 01 : 90° difference phase, 1 times multiplied 10 : 90° difference phase, 2 times multiplied 11 : 90° difference phase, 4 times multiplied 5 bit : INP 신호 ACTIVE LEVEL 설정 4 bit : ALM 신호 ACTIVE LEVEL 설정 3 bit : -SLM 신호 ACTIVE LEVEL 설정 2 bit : +SLM 신호 ACTIVE LEVEL 설정 1 bit : -ELM 신호 ACTIVE LEVEL 설정 0 bit : +ELM 신호 ACTIVE LEVEL 설정		
F3	UNIVERSAL OUT WRITE 10~8 bit : Universal output bit operation 0xx : 연산데이터 값을 UOUT으로 출력 100 : NOT(현재 UOUT의 반전) 101 : AND(연산데이터 값과 현재 UOUT의 and 연산 출력) 110 : OR(연산데이터 값과 현재 UOUT의 or 연산 출력) 111 : XOR(연산데이터 값과 현재 UOUT의 difference check) 7 ~ 4 bit : Don't care 3 ~ 0 bit : 각 Universal output(uout3~0) 연산 데이터	11	0x000
F4~F6	NO OPERATION		
F7	EXTERNAL COUNTER 설정 DATA WRITE 7 bit : 0 : Not reverse count, 1: Reverse count 6 bit : Don't care 5 ~ 4 bit : 00 : CLEAR REQUEST RESET 01 : ONE TIME CLEAR REQUEST SET 10 : FULL TIME CLEAR REQUEST SET 11 : EXTERNAL COUNTER CLEAR SET 3 ~ 0 bit : CLEAR 입력신호 선택 0001 : IN0 0010 : IN1 0100 : IN2 1000 : IN3 Others : Don't care	9	0x00
F8	REGISTER CLEAR (INITIALIZATION)		

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
F9	Interrupt 발생 Command		
FA	Interrupt MASK(0 : Interrupt Disable, 1 : Enable) 31 bit : Selectable Interrupt source3("FE"의 31~24bit) 30 bit : Selectable Interrupt source2("FE"의 23~16bit) 29 bit : Selectable Interrupt source1("FE"의 15~8bit) 28 bit : Selectable Interrupt source0("FE"의 7~0bit) 27 bit : Emergency limit 신호 입력시 26 bit : Slow down limit 신호 입력시 25 bit : 캡션 queue 명령어 수행시(30 bit '1' 일때) 24 bit : 스크립트 queue 명령어 수행시(30 bit '1' 일때) 23 bit : M-data count clear 22 bit : 현재속도 데이터 = 현재 속도 비교데이터 21 bit : 현재속도 데이터 = RCP12 20 bit : 현재속도 데이터 = RCP23 19 bit : MODE1 register에 설정된 신호 검출시 18 bit : 가속 17 bit : 등속 16 bit : 감속 15 bit : Don't care 14 bit : 캡션-3에 저장된 command가 실행될 때 13 bit : 캡션-2에 저장된 command가 실행될 때 12 bit : 캡션-1에 저장된 command가 실행될 때 11 bit : 스크립트-3에 저장된 command가 실행될 때 10 bit : 스크립트-2에 저장된 command가 실행될 때 9 bit : 스크립트-1에 저장된 command가 실행될 때 8 bit : 캡션 동작 설정 queue가 full될 때 7 bit : 스크립트 data queue가 empty될 때 6 bit : External counter < External comparator가 될 때 5 bit : External counter = External comparator가 될 때 4 bit : External counter > External comparator가 될 때 3 bit : Internal counter < Internal comparator가 될 때 2 bit : Internal counter = Internal comparator가 될 때 1 bit : Internal counter > Internal comparator가 될 때 0 bit : 드라이브 종료	32	0x0001

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
FB	<p>EMODE1 DATA WRITE</p> <p>7 bit : Extension mode enable(0 : Disable, 1 : Enable)                      0 : md_out[12:8] =&gt; md_out[12:8]                      1 : md_out[12:8] =&gt; EUIO</p> <p>6~ 5 bit : Don't care</p> <p>4 bit : EUO4 enable/disable(0 : disable, 1 : enable)</p> <p>3 bit : EUO3 enable/disable(0 : disable, 1 : enable)</p> <p>2 bit : EUO2 enable/disable(0 : disable, 1 : enable)</p> <p>1 bit : EUO1 enable/disable(0 : disable, 1 : enable)</p> <p>0 bit : EUIO0 direction select(0 : input, 1 : output)</p>	8	0X00
FC	<p>Extension UNIVERSAL OUT WRITE</p> <p>7 ~ 5 bit : Extension Universal output bit operation</p> <p>0xx : 연산데이터 값을 OUT으로 출력</p> <p>100 : NOT (현재 OUT의 반전)</p> <p>101 : AND (연산데이터 값과 현재 UOUT의 and 연산 출력)</p> <p>110 : OR (연산데이터 값과 현재 UOUT의 or 연산 출력)</p> <p>111 : XOR (연산데이터 값과 현재 UOUT의 difference check)</p> <p>4 ~ 0 bit : Extension Universal output 연산 데이터(input의 경우 무시)</p>	8	0x00
FD	Limit 완전정지 해제		
FE	<p>USER Interrupt source selection register WRITE</p> <p>31~24 bit : User selectable interrupt 3 source selection                      스크립트, 캡션 검출이벤트 리스트와 동일</p> <p>23~16 bit : User selectable interrupt 2 source selection                      스크립트, 캡션 검출이벤트 리스트와 동일</p> <p>15~ 8 bit : User selectable interrupt 1 source selection                      스크립트, 캡션 검출이벤트 리스트와 동일</p> <p>7 ~ 0 bit : User selectable interrupt 0 source selection                      스크립트, 캡션 검출이벤트 리스트와 동일</p>	32	0x00000000
FF	No operation		

## 6.2. COMMAND 기능 설명

펄스 출력에 관계되는 레지스터들은 2개의 그룹으로 나누어져 있다. 이에 관련된 명령들은 기본적으로 “PGM-1” 레지스터 WRITE/READ로 되어있다. 이 레지스터들과 같은 종류의 데이터를 저장하거나 읽을 수 있는 “PGM-1 UPDATE” 레지스터 WRITE/READ가 있다. 이 명령들을 이용하여 두 종류의 속도 패턴을 저장할 수 있으므로 두 종류 속도 패턴으로 드라이브 구동을 간편하게 전환할 수 있다.

“PGM-2” 그룹의 명령들은 드라이브 시작, 거리 오버라이드 등의 명령들로 구성되어 있다.

“BUS-1” 그룹의 명령들은 내부/외부 카운터에 관련된 명령들이다.

“BUS-2” 그룹의 명령들은 칩 동작에 필요한 설정에 관련된 명령들이다.

“스크립트”/“갈무리” 그룹의 명령들은 스크립트 기능과 갈무리 기능의 설정에 관련된 명령들이다.

이들 명령들의 자세한 설명은 다음과 같다.

### ▶ PGM-1 RANGE DATA READ COMMAND ( CODE = 0x00 )

현재 설정되어 있는 PGM-1 RANGE DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

### ▶ PGM-1 RANGE DATA WRITE COMMAND ( CODE = 0x80 )

PGM-1의 RANGE DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 RANGE 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)에 실행할 경우는 출력 중인 펄스 속도가 갑자기 변경되기 때문에 모터 구동이 정지할 수도 있으므로 주의가 필요하다.

### ▶ PGM-1 START/STOP SPEED DATA READ COMMAND ( CODE = 0x01 )

현재 설정되어 있는 PGM-1 START/STOP SPEED DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

### ▶ PGM-1 START/STOP SPEED DATA WRITE COMMAND ( CODE = 0x81 )

PGM-1의 START/STOP SPEED DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 START/STOP SPEED 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 1(0x0001)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)에 실행할 경우는 출발/정지 속도 데이터가 변하므로 주의가 필요하다.

### ▶ PGM-1 OBJECT SPEED DATA READ COMMAND ( CODE = 0x02 )

현재 설정되어 있는 PGM-1 OBJECT SPEED DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3,

DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 OBJECT SPEED DATA WRITE COMMAND ( CODE = 0x82 )

PGM-1의 OBJECT SPEED DATA를 설정하는 COMMAND이다.

DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 OBJECT SPEED 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 1(0x0001)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)에 실행할 경우는 목표 속도 데이터가 변경되므로 주의가 필요하다.

▶ PGM-1 RATE-n (n=1, 2, 3) DATA READ COMMAND ( CODE = 0x03, 0x04, 0x05 )

현재 설정되어 있는 PGM-1 RATE-n (n=1, 2, 3) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 RATE-n (n=1, 2, 3) DATA WRITE COMMAND ( CODE = 0x83, 0x84, 0x85 )

PGM-1의 RATE-n (n=1, 2, 3) DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 RATE-n (n=1, 2, 3) 데이터로 저장한다.

설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)에 실행할 경우는 출력 중인 펄스의 가감속율이 변화된다. 그러므로 펄스 수 설정 드라이브 구동 중 RATE-n (n=1, 2, 3) 데이터를 변경할 경우 비정상적인 정지 동작이 일어날 수 있다.

▶ PGM-1 RATE CHANGE POINT n(n=1-2, 2-3) READ COMMAND ( CODE = 0x06, 0x07 )

현재 설정되어 있는 PGM-1 Rate Change Point n(n=1-2, 2-3) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다. 이 Command는 항상 실행 할 수 있다.

▶ PGM-1 RATE CHANGE POINT n(n=1-2, 2-3) WRITE COMMAND ( CODE = 0x86, 0x87 )

Data 3, 4 Port를 사용하여 PGM-1 Rate Change Point n(n=1-2, 2-3) Data를 설정하는 Command이다.

설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다.

이 Command는 상시 실행 가능하지만 Drive 중(BUSY = High)에 실행할 경우에는 새로이 설정된 PGM-1 Rate Change Point n(n=1-2, 2-3) Data에 따라 변곡점(變曲点)이 바뀐다. 따라서 지정 펄스 수 구동 중에 실행할 경우에는 정상적인 감속 정지가 실행되지 않는 경우도 있으므로 주의가 필요하다.

▶ PGM-1 SW-n (n=1, 2) DATA READ COMMAND ( CODE = 0x08, 0x09 )

현재 설정되어 있는 PGM-1 SW-n (n=1, 2) (S-curve Width) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{14} \sim 2^0$  bit 순으로 15-bit 크기로 저장된다. 저장된 15-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 SW-n (n=1, 2) DATA WRITE COMMAND ( CODE = 0x88, 0x89 )

PGM-1의 SW-n (n=1, 2) (S-curve Width) DATA를 설정하는 COMMAND이다. 설정한 데이터로 직선 또는 S 곡선 가감속을 설정하는 명령이다. PGM-1 SW-n (n=1, 2) 값을 0x7FFF로 설정하면 S 곡선 가감속 동작을 하고, 이 값을 0x0000로 설정하면 직선 가감속 동작을 한다. RESET 후 기본값은 0x7FFF이다. 드라이브 구동 중 이 값을 변경하는 경우 비정상적인 가감속 동작이 발생할 수 있다.

▶ PGM-1 PWM 출력 설정 DATA READ COMMAND ( CODE = 0x0A )

현재 설정된 PGM-1 PWM 출력 설정 DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA4 port에 저장된다. 저장된 데이터는 DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 PWM 출력 설정 DATA WRITE COMMAND ( CODE = 0x8A )

PGM-1 PWM 출력 설정 DATA를 설정하는 COMMAND이다. 이 COMMAND에 의해 설정된 DATA로 PWM 출력 파형을 설정할 수 있다. 초기값은 0x0이며 0~6(0x00~0x06) 사이의 값을 설정할 수 있다.

▶ PGM-1 SLOW DOWN/REAR PULSE READ COMMAND ( CODE = 0x0B )

현재 설정되어 있는 PGM-1 SLOW DOWN/REAR PULSE DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장된다. 저장된 32-bit 데이터는 DATA1, DATA2, DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 SLOW DOWN/REAR PULSE WRITE COMMAND ( CODE = 0x8B )

PGM-1 SLOW DOWN/REAR PULSE DATA를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit의 32-bit 데이터를 PGM-1 SLOW DOWN/REAR PULSE 데이터로 설정한다.

RESET 후 기본값은 0 (0x00000000)이며 설정 범위는 0 ~ 4,294,967,295(0x00000000 ~ 0xFFFFFFFF)이다. SLOW DOWN / REAR PULSE DATA는 지정 펄스 수 드라이브에서 감속 시작 타이밍을 제어하는 데이터이므로 이 데이터 설정에 따라 드라이브 속도의 가감속 형태가 달라진다.

1) 자동 검출 방식의 경우

감속 후 시작/정지 속도로 출력되는 펄스수(그림의 빗금 부분)를 지정한다.

단, 빗금 부분의 펄스 수는 지정펄스 수 드라이브의 경우에 있어서 지정 펄스 수를 포함한다.

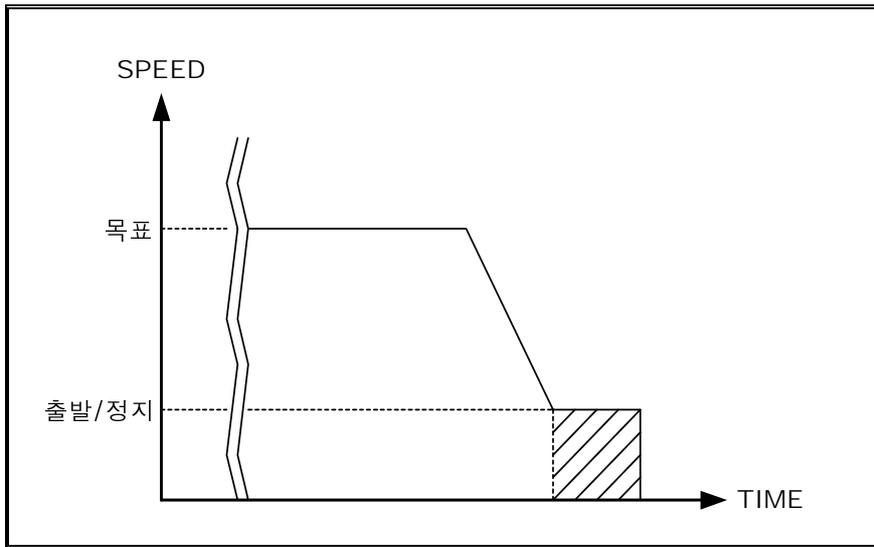


그림 6.1. 자동 감속방식

2) 남은 PULSE수 지정 방식의 경우

감속 개시 타이밍에 대해 남은 펄스 수(그림의 빗금 부분을)를 지정한다. 단, 남은 펄스 수의 지정에 의해서는 시작/정지 속도까지 감속하지 않는 경우나, 시작/정지 속도까지 감속하고 나서부터 실제로 정지할 때까지 시간을 요하는 경우가 있다.

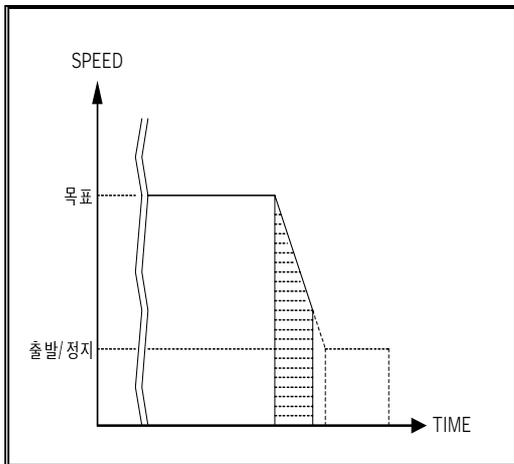


그림 6.2. 남은 PULSE수 지정방식 - 1

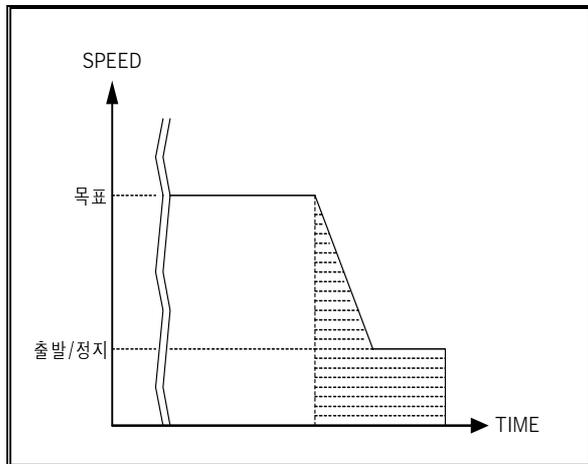


그림 6.3. 남은 PULSE수 지정방식 - 2

이 COMMAND는 상시 실행 가능하지만, DRIVE중(BUSY = H중)에 실행한 경우에는 현 DRIVE에는 반영 되지않고 다음 DRIVE할 때부터 반영된다.

▶ PGM-1 현재 SPEED DATA READ COMMAND ( CODE = 0x0C )

현재 출력되고 있는 속도 데이터를 출력하는 COMMAND이다. 읽은 DATA는 DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 16-bit 크기로 저장된다. 이 COMMAND는 항상 실행 가능하지만, 드라이브가 정지 중 (BUSY = Low)인 경우 이 COMMAND를 실행한 경우 0(0x0000)이 읽혀진다.

▶ PGM-1 현재 SPEED 비교 DATA READ COMMAND ( CODE = 0x0D )

현재 출력 중인 속도 데이터와 비교할 데이터인 현재 SPEED 비교 DATA를 출력하는 COMMAND이다. 읽은 DATA는 DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 16-bit 크기로 저장된다.

▶ PGM-1 현재 SPEED 비교 DATA WRITE COMMAND ( CODE = 0x8D )

펄스 출력 드라이브 중에 출력되는 현재 속도 데이터와 비교할 PGM-1 현재 SPEED 비교 DATA를 설정하는 COMMAND이다. RESET 후 현재 SPEED 비교 DATA의 기본값은 0(0x0000)이며, 데이터의 설정범위는 0~65,535(0x0000~0xFFFF)이며 DATA3, DATA4 port에 저장된 2<sup>15</sup>~2<sup>0</sup> bit인 16-bit 현재 SPEED 비교 DATA로 저장한다.

▶ PGM-1 DRIVE PULSE COUNTER READ COMMAND ( CODE = 0x0E )

드라이브 중 출력되는 펄스 카운터 값을 읽어내는 COMMAND이다. 카운트 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF)이며, 현재 구동 중인 드라이브에서 출력된 펄스 값을 출력한다. 따라서 드라이브 종료 후 값은 0(0x00000000)이 된다.

▶ PGM-1 PRESET PULSE DATA READ COMMAND ( CODE = 0x0F )

PGM-1의 설정 데이터로 PRESET PULSE 드라이브(지정펄스 수 드라이브) 구동에 출력할 총 펄스 수를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장된다.

▶ PGM-1 UP-DATE RANGE READ COMMAND ( CODE = 0x10 )

현재 설정되어 있는 PGM-1 UP-DATE RANGE DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE RANGE WRITE COMMAND ( CODE = 0x90 )

PGM-1 UP-DATE의 RANGE DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된 2<sup>15</sup>~2<sup>0</sup> bit 데이터를 RANGE 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)에 실행할 경우는 출력 중인 펄스 속도가 갑자기 변경되기 때문에 모터 구동이 정지할 수도 있으므로 주의가 필요하다.

▶ PGM-1 UP-DATE START/STOP SPEED DATA READ COMMAND ( CODE = 0x11 )

현재 설정되어 있는 PGM-1 UP-DATE START/STOP SPEED DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE START/STOP SPEED DATA WRITE COMMAND ( CODE = 0x91 )

PGM-1 UP-DATE의 START/STOP SPEED DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 START/STOP SPEED 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 1(0x0001)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중 (BUSY = High)에 실행할 경우는 출발/정지 속도 데이터가 변하므로 주의가 필요하다.

▶ PGM-1 UP-DATE OBJECT SPEED DATA READ COMMAND ( CODE = 0x12 )

현재 설정되어 있는 PGM-1 UP-DATE OBJECT SPEED DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE OBJECT SPEED DATA WRITE COMMAND ( CODE = 0x92 )

PGM-1 UP-DATE의 OBJECT SPEED DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 OBJECT SPEED 데이터로 저장한다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 1(0x0001)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중 (BUSY = High)에 실행할 경우는 목표 속도 데이터가 변경되므로 주의가 필요하다. S자 가감속 모드로 드라이브 구동 시 가감속 중 OBJECT SPEED DATA를 변경할 경우 S자 가감속을 위해 명령을 2번 실행해야 한다. PRESET PULSE DRIVE 구동 중 OBJECT SPEED DATA를 변경할 경우 자동 감속 시점 계산을 위해 SLOW DOWM/REAR PULSE WRITE COMMAND를 이용해야 한다. 자세한 수식은 부록에 첨가하였다.

▶ PGM-1 UP-DATE RATE-n (n=1, 2, 3) DATA READ COMMAND ( CODE = 0x13, 0x14, 0x15 )

현재 설정되어 있는 PGM-1 UP-DATE RATE-n (n=1, 2, 3) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE RATE-n (n=1, 2, 3) DATA WRITE COMMAND ( CODE = 0x93, 0x94, 0x95 )

PGM-1의 UP-DATE RATE-n (n=1, 2, 3) DATA를 설정하는 COMMAND이다. DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit 데이터를 PGM-1 UP-DATE RATE-n (n=1, 2, 3) 데이터로 저장한다.

설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다. 이 COMMAND는 항상 실행 가능하지만 드라이브 구동 중 (BUSY = High)에 실행할 경우는 출력 중인 펄스의 가감속율이 변화된다. 그러므로 펄스 수 설정 드라이브 구동 중 PGM-1 UP-DATE RATE-n (n=1, 2, 3) 데이터를 변경할 경우 비정상적인 정지 동작이 일어날 수 있다.

▶ PGM-1 UP-DATE RATE CHANGE POINT n(n=1-2, 2-3) READ COMMAND ( CODE = 0x16, 0x17 )

현재 설정되어 있는 PGM-1 UP-DATE Rate Change Point n(n=1-2, 2-3) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는

DATA3, DATA4 Address를 읽어 확인하면 된다. 이 Command는 항상 실행 할 수 있다.

▶ PGM-1 UP-DATA RATE CHANGE POINT n(n=1-2, 2-3) WRITE COMMAND ( CODE = 0x96, 0x97)

Data 3, 4 Port를 사용하여 PGM-1 UP-DATA Rate Change Point n(n=1-2, 2-3) Data를 설정하는 Command이다. 설정범위는 1~65,535(0x0001~0xFFFF)이며 RESET 후 기본값은 65,535(0xFFFF)이다.

이 Command는 상시 실행 가능하지만 Drive 중(BUSY = High)에 실행할 경우에는 새로이 설정된 PGM-1 Rata Change Point n(n=1-2, 2-3) Data에 따라 변곡점(變曲点)이 바뀐다. 따라서 지정 펄스 수 구동 중에 실행할 경우에는 정상적인 감속 정지가 실행되지 않는 경우도 있으므로 주의가 필요하다.

▶ PGM-1 UP-DATE SW-n (n=1, 2) DATA READ COMMAND ( CODE = 0x18, 0x19 )

현재 설정되어 있는 PGM-1 UP-DATE SW-n (n=1, 2) DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{14} \sim 2^0$  bit 순으로 15-bit 크기로 저장된다. 저장된 15-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE SW-n (n=1, 2) DATA WRITE COMMAND ( CODE = 0x98, 0x99 )

PGM-1의 UP-DATE SW-n (n=1, 2) (S-curve Width) DATA를 설정하는 COMMAND이다. 설정한 데이터로 직선 또는 S 곡선 가감속을 설정하는 명령이다. PGM-1 UP-DATE SW-n (n=1, 2) 값을 0x7FFF로 설정하면 S 곡선 가감속 동작을 하고, 이 값을 0x0000로 설정하면 직선 가감속 동작을 한다. RESET 후 기본값은 0x7FFF이다. 드라이브 구동 중 이 값을 변경하는 경우 비정상적인 가감속 동작이 발생할 수 있다.

▶ PGM-1 CURRENT SPEED CHANGE DATA READ COMMAND (0x1A)

현재 설정되어 있는 PGM-1 CURRENT SPEED CHANGE DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 저장된 16-bit 데이터는 DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 CURRENT SPEED CHANGE DATA WRITE COMMAND (0x9A)

현재 출력 SPEED DATA의 변경을 가능하게 하는 COMMAND이다. RESET 후 PGM-1 CURRENT SPEED CHANGE DATA의 기본값은 0(0x0000)이며, 데이터의 설정범위는 0~65,535(0x0000~0xFFFF)이며 DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit인 16-bit PGM-1 CURRENT SPEED CHANGE DATA로 저장한다.

단, Preset Drive와 S자 구동시에는 올바르게 작동하지 않을 수 있다.

▶ PGM-1 UP-DATE SLOW DOWN/REAR PULSE READ COMMAND ( CODE = 0x1B )

현재 설정되어 있는 PGM-1 UP-DATE SLOW DOWN/READ PULSE DATA를 읽어내는 COMMAND이다. 읽은 데이터는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장된다. 저장된 32-bit 데이터는 DATA1, DATA2, DATA3, DATA4 Address를 읽어 확인하면 된다.

▶ PGM-1 UP-DATE SLOW DOWN/REAR PULSE WRITE COMMAND ( CODE = 0x9B )

PGM-1 UP-DATE SLOW DOWN/REAR PULSE DATA를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit의 32-bit 데이터를 PGM-1 UP-DATE SLOW DOWN/READ PULSE 데이터로 설정한다. RESET 후 기본값은 0 (0x00000000)이며 설정 범위는 0 ~ 4,294,967,295(0x00000000 ~ 0xFFFFFFFF)이다. SLOW DOWN / REAR PULSE DATA는 지정 펄스 수 드라이브에서 감속 시작 타이밍을 제어하는 데이터이므로 이 데이터 설정에 따라 드라이브 속도의 가감속 형태가 달라진다.

1) 자동 검출 방식의 경우

감속 후 시작/정지 속도로 출력되는 펄스 수(그림의 빗금 부분)를 지정한다.

단, 빗금 부분의 펄스 수는 지정펄스 수 드라이브의 경우에 있어서 지정 펄스 수를 포함한다.

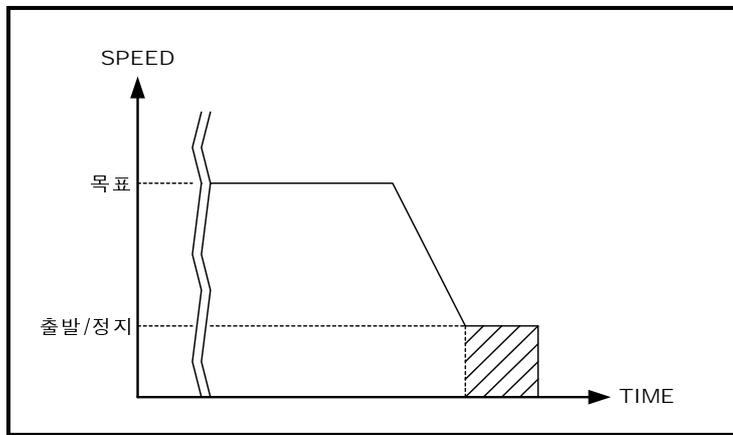


그림 6.4. 자동 검출 방식

2) 남은 PULSE수 지정 방식의 경우

감속 개시 타이밍에 대해 남은 펄스 수(그림의 빗금 부분)를 지정한다. 단, 남은 펄스 수의 지정에 의해서는 시작/정지 속도까지 감속하지 않는 경우나, 시작/정지 속도까지 감속하고 나서부터 실제로 정지할 때까지 시간을 요하는 경우가 있다.

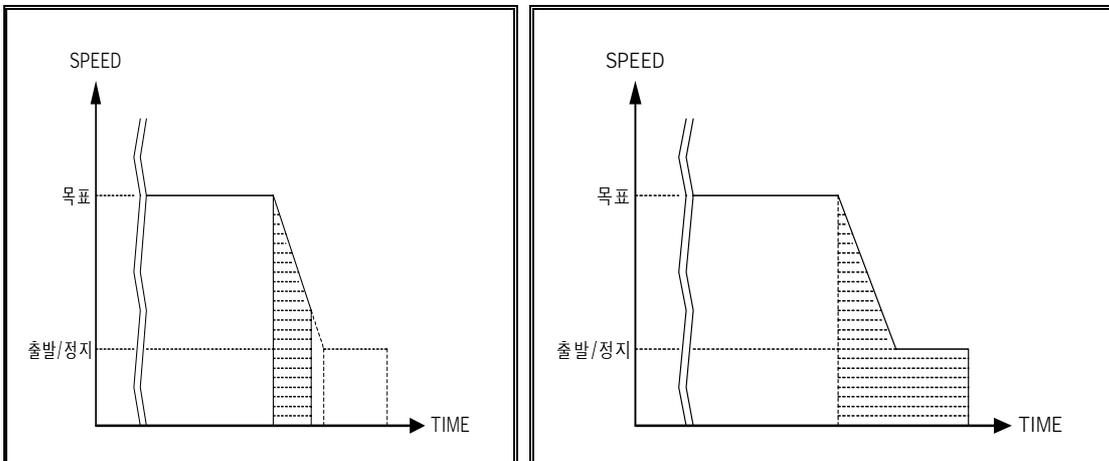


그림 6.5. 남은 PULSE수 지정 방식 -1      그림 6.6. 남은 PULSE수 지정 방식 -2

이 COMMAND는 상시 실행 가능하지만, DRIVE중(BUSY = H중)에 실행한 경우에는 현 DRIVE에는 반영 되지않고 다음 DRIVE할 때부터 반영된다.

▶ PGM-1 UP-DATE 현재 SPEED DATA READ COMMAND ( CODE = 0x1C )

PGM-1 UP-DATE의 현재 출력되고 있는 속도 데이터를 출력하는 COMMAND이다. 읽은 DATA는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다. 이 COMMAND는 항상 실행 가능하지만, 드라이브가 정지 중(BUSY = Low)인 경우 이 COMMAND를 실행한 경우 0(0x0000)이 읽혀진다.

▶ PGM-1 UP-DATE 현재 SPEED 비교 DATA READ COMMAND ( CODE = 0x1D )

PGM-1 UP-DATE의 현재 출력 중인 속도 데이터와 비교할 데이터인 현재 SPEED 비교 DATA를 출력하는 COMMAND이다. 읽은 DATA는 DATA3, DATA4 port에  $2^{15} \sim 2^0$  bit 순으로 16-bit 크기로 저장된다.

▶ PGM-1 UP-DATE 현재 SPEED 비교 DATA WRITE COMMAND ( CODE = 0x9D )

펄스 출력 드라이브 중에 출력되는 현재 속도 데이터와 비교할 PGM-1 UP-DATE 현재 SPEED 비교 DATA를 설정하는 COMMAND이다. RESET 후 현재 SPEED 비교 DATA의 기본값은 0(0x0000)이며, 데이터의 설정범위는 0~65,535(0x0000~0xFFFF)이며 DATA3, DATA4 port에 저장된  $2^{15} \sim 2^0$  bit인 16-bit 현재 SPEED 비교 DATA로 저장한다.

▶ PGM-1 UP-DATE DRIVE PULSE COUNTER READ COMMAND ( CODE = 0x1E )

PGM-1 UP-DATE의 드라이브 중 출력되는 펄스 카운터 값을 읽어내는 COMMAND이다. 카운트 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF) 이며, 현재 구동 중인 드라이브에서 출력된 펄스 값을 출력한다. 따라서 드라이브 종료 후 값은 0(0x00000000)이 된다.

▶ PGM-1 UP-DATE PRESET PULSE DATA READ COMMAND ( CODE = 0x1F )

PGM-1 UP-DATE의 설정 데이터로 PRESET PULSE 드라이브(지정펄스 수 드라이브) 구동에 출력할 총 펄스 수를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장된다.

▶ +/- PRESET PULSE DRIVE WRITE COMMAND( CODE = 0xA0, 0xA5 )

CW(+) 또는 CCW(-) 방향으로 지정펄스 수 드라이브를 실행하기 위한 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 지정펄스 수(드라이브 중 출력할 펄스 수)를 저장한 후 +/- PRESET PULSE DRIVE COMMAND를 실행한다.

지정 펄스 수 설정 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF)이고, DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 펄스 수를 설정하며 단위는 펄스 수이다. 드라이브 실행 후 지정펄스 수만큼의 펄스를 출력한 후 드라이브를 종료한다. 드라이브 동작 설정 명

령의 입력 데이터의 2bit의 setting에 따라서 출력 펄스 수는 칩에서 출력되는 펄스 수를 카운트한 값을 사용할 수 있고, 외부 펄스 수(외부 위치 카운터 값)를 카운트한 값을 사용할 수 있다.

PRESET PULSE DRIVE COMMAND 종료 후 END STATUS 레지스터값은 0x000가 된다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

▶ +/- CONTINUOUS DRIVE WRITE COMMAND ( CODE = 0xA1, 0xA6 )

CW(+) 또는 CCW(-) 방향으로 연속 드라이브를 실행하기 위한 COMMAND이다. 드라이브 시작 후 드라이브 종료 명령 또는 드라이브 종료 신호가 입력될 때까지 모터 구동 펄스를 출력한다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

▶ +/- SIGNAL SEARCH -1 DRIVE WRITE COMMAND ( CODE = 0xA2, 0xA7 )

CW(+) 또는 CCW(-) 방향으로 신호 검출 드라이브를 실행하기 위한 COMMAND이다.

SIGNAL SEARCH -1 DRIVE는 드라이브 시작 후 시작 속도에서 목표 속도까지 가속하면서 설정된 검출 신호에 의해 감속정지 동작을 한다. 검출 신호의 선택 및 상향, 하향 예지에 대한 설정은 MODE1(0xF1)에서 기술된다.

신호 검출에 의한 동작 정지 시 END STATUS 레지스터 값은 0x400이 된다.

검출 신호 입력을 +/- ELM/SLM (Emergency/Slow Down Limit) 신호를 사용할 경우 칩 기능 설정 명령에서 Emergency Limit Disable/Slow Down Limit Disable로 설정한 후 사용하여야 한다. Emergency/Slow Down Limit Enable로 설정한 후 +/- ELM/SLM를 신호 검출 신호로 사용한 경우 Limit 신호에 의한 급/감속 정지 기능이 우선 동작하여 신호 검출 기능이 올바르게 작동하지 않을 수 있다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

▶ +/- SIGNAL SEARCH -2 DRIVE WRITE COMMAND ( CODE = 0xA3, 0xA8 )

CW(+) 또는 CCW(-) 방향으로 신호 검출 드라이브를 실행하기 위한 COMMAND이다.

SIGNAL SEARCH -2 DRIVE는 시작 속도로 등속 드라이브를 수행하며 드라이브 중 신호 검출이 되면 펄스 출력을 중지한다. 검출 신호의 선택 및 상향, 하향 예지에 대한 설정은 MODE1(0xF1)에서 결정된다.

신호 검출에 의한 동작 정지 시 END STATUS 레지스터 값은 0x400이 된다.

검출 신호 입력을 +/- ELM/SLM (Emergency/Slow Down Limit) 신호를 사용할 경우 칩 기능 설정 명령에서 Emergency Limit Disable/Slow Down Limit Disable로 설정한 후 사용하여야 한다. Emergency/Slow Down Limit Enable로 설정한 후 +/- ELM/SLM를 신호 검출 신호로 사용한 경우 Limit 신호에 의한 급/감속 정지 기능이 우선 동작하여 신호 검출 기능이 올바르게 작동하지 않을 수 있다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

▶ +/- HOME (원 점) SEARCH DRIVE (CODE =0xA4, 0xA9)

Home Search Drive는 드라이브 시작 명령으로 가장 기본적인 원점 검출 동작을 수행한다.

CW(+) 또는 CCW(-) 방향으로 구동 가능하며 검출할 원점 신호 입력시 edge point (rising/falling)를 사용자가 설정할 수 있다.

원점 검출에 의한 동작 정지 시 END STATUS 레지스터 9bit가 HIGH로 표시된다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

▶ PRESET/MPG PULSE DATA OVERRIDE WRITE COMMAND ( CODE = 0xAA )

PRESET/MPG PULSE DATA를 변경하는 COMMAND이다.

펄스 수 설정 범위는 0~4,294,967,295(0X00000000~0XFFFFFFF)이고, DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 펄스 수를 설정하며 단위는 펄스 수이다.

지정펄스 수 드라이브 동작 중에 지정된 펄스 수를 변경하고자 할 경우 이 명령을 사용하면 된다. 이 명령은 지정펄스 수를 변경하는 명령이므로 지정펄스 수 드라이브 중 이외에 실행할 경우는 의미가 없다. 이 명령을 이용할 경우 변경하는 펄스 수에 따라 다음에서 설명하는 세 가지의 경우가 발생하므로 주의하여 사용해야 한다. 이외에 감속 중에 OVERRIDE 명령을 실행할 경우 속도 가감속에 이상 동작이 발생할 수 있으므로 사용에 주의가 필요하다. 다음 경우 이외에 감속 중에 OVERRIDE 명령을 실행할 경우 속도 가감속에 이상 동작이 발생할 수 있으므로 사용에 주의가 필요하다.

1) CASE 1

새로 설정된 지정펄스 수가 이미 출력을 마친 펄스 수와 비교하여 같거나 작을 경우 지정 펄스 수를 최우선으로 하기 때문에 이 COMMAND가 실행 되는대로 급정지 한다.

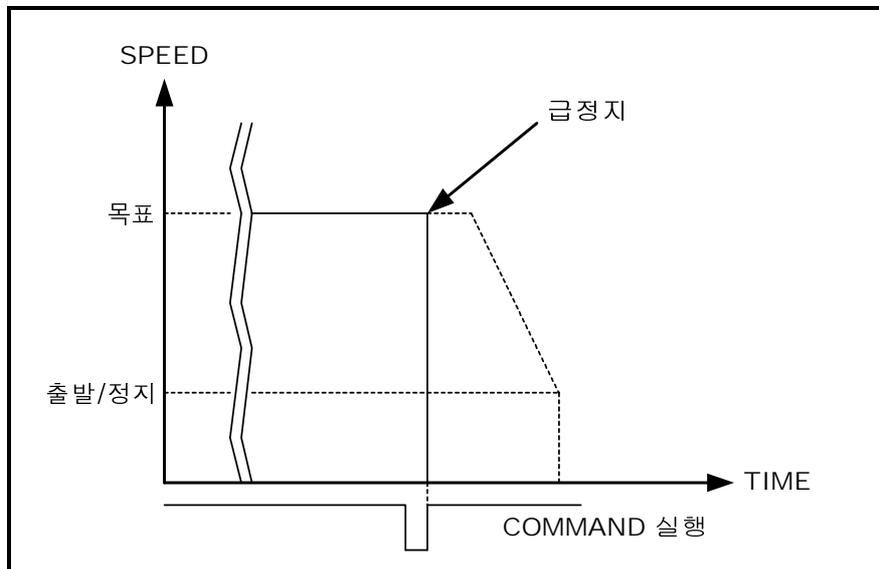


그림 6.7. 새로운 PULSE수 지정으로 인한 급정지동작

2) CASE 2

새로 설정된 지정펄스 수가 그 시점에서부터 감속 정지하기에는 부족할 경우, 이 명령이 실행되는 대로 감속을 시작하지만 지정펄스 수에 이르게 되면 정지한다.

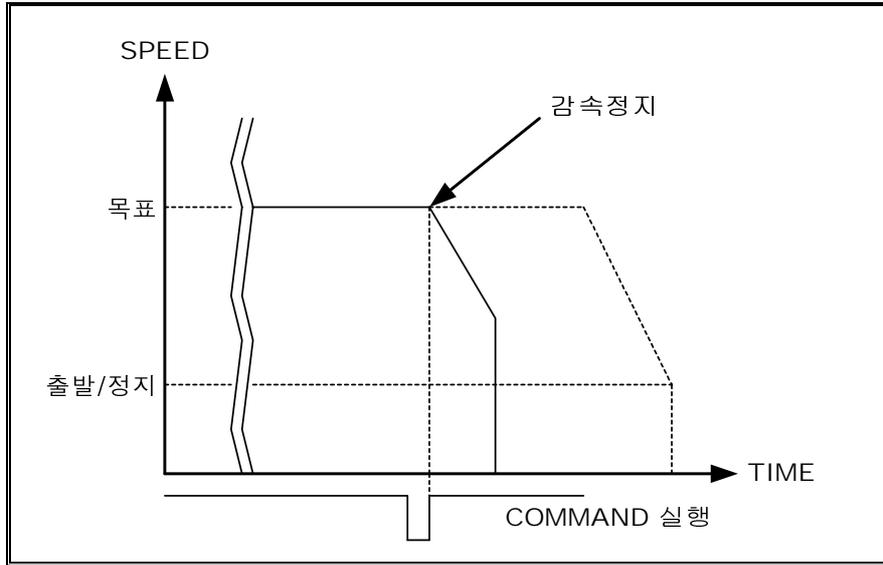


그림 6.8. 새로운 PULSE수 지정으로 인한 감속 정지 동작

▶ SLOW DOWN STOP WRITE COMMAND ( CODE = 0xAB )

드라이브를 감속 정지시키기 위한 COMMAND이다. 이 명령은 상시 실행 가능하지만 드라이브 정지 중(BUSY = Low)에 실행시킨 경우에는 아무런 의미가 없다.

또 이 명령에 의해 펄스 출력이 정지한 경우 END STATUS 레지스터의 SSCED Bit(Slowdown Stop Command END Bit, 4 bit)가 1(High)로 된다.

▶ EMERGENCY STOP WRITE COMMAND ( CODE = 0xAC )

드라이브를 급정지 시키기 위한 COMMAND이다. 이 명령은 상시 실행 가능하지만 드라이브 정지 중(BUSY = Low)에 실행시킨 경우에는 아무런 의미가 없다.

또 이 명령에 의해 펄스 출력이 정지한 경우 END STATUS 레지스터의 ESCED Bit(Emergency Stop Command END Bit, 5 bit)가 1(High)로 된다.

▶ DRIVE MODE SETTING DATA WRITE COMMAND ( CODE = 0xAD )

드라이브 동작 설정 데이터를 설정하는 COMMAND이다. RESET 후 초기값은 0x0을 가진다. DATA4 port에  $2^2 \sim 2^0$  bit 순으로 저장된 3-bit 크기 데이터를 드라이브 동작 설정 데이터로 사용된다. 이 COMMAND로 설정한 데이터를 이용하여 대칭/비대칭 가감속 동작 모드를 설정한다. 드라이브 동작 설정 데이터의 내용은 다음과 같다.

2bit	실거리 연산기능(0: 사용안함, 1 : 사용)
0~1bit	00: 대칭 직선 가감속 모드 01: 비대칭 직선 가감속 모드 10: 대칭 S-곡선 가감속 모드 11: 비대칭 S-곡선 가감속 모드

실거리 연산 기능(2 bit)는 지정펄스 수 드라이브에서 사용한다. 이 기능을 사용하는 경우는 출력 펄스 수를 외부 위치 카운터 값으로 사용한다. 따라서 칩에서 출력되는 펄스 수가 지정펄스 수가 넘어가더라도 외부 기계계의 모터가 회전한 수(외부 위치 카운터로 회전 정보를 읽음)가 지정펄스 수와 일치할 때까지 지정펄스 수 드라이브는 정지하지 않고 계속 동작한다.

1~0 bit는 구동 드라이브의 현재 속도 가감속 모드를 설정하는 Bit이다.

▶ MPG OPERATION SETTING DATA WRITE COMMAND (CODE =0xAE)

MPG OPERATION SETTING DATA를 설정하는 COMMAND이다. 7BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

6bit	외부 신호에 의한 Continuous drive 구동 (0 : RESET, 1 : SET)
5bit	외부 신호에 의한 Preset pulse drive 구동 (0 : RESET, 1 : SET)
3~4bit	00: 단상 01: 1채배 10: 2채배 11: 4채배
2bit	0: 사용자 설정 방향 +, 1: 사용자 설정 방향 -
1bit	0: 외부 2상 펄스방향 출력, 1: 사용자 설정 방향 출력
0bit	외부 2상 펄스에 의한 구동(0 : RESET, 1 : SET)

▶ MPG PRESET PULSE DATA WRITE (CODE=0xAF)

MPG Preset Pulse Drive에서 출력하고자 하는 펄스 수를 설정하는 COMMAND이다. . RESET 후 MPG PRESET PULSE DATA의 기본값은 0(0x00000000)이다.

▶ +/- Sensor Positioning Drive I (CODE=0xB0, 0xB1)

CW(+) 또는 CCW(-) 방향으로 MARK 신호의 edge가 검출되면, 사용자가 지정한 펄스 수 만큼 구동하는 드라이브를 실행하기 위한 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 지정펄스 수(드라이브 중 출력할 펄스 수)를 저장한 후 +/- Sensor Positioning Drive I COMMAND를 실행한다. 지정 펄스 수 설정 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF)이고, DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 저장된 32-bit 크기 데이터로 펄스 수를 설정하며 단위는 펄스 수이다.

드라이브 실행 후 지정펄스 수만큼의 펄스를 출력한 후 드라이브를 종료한다. 드라이브 동작 설정 명령(0xAD)의 입력 데이터의 2bit의 setting에 따라서 출력 펄스 수는 칩에서 출력되는 펄스 수를 카운트한 값을 사용할 수 있고, 외부 펄스 수(외부 위치 카운터 값)를 카운트한 값을 사용할 수 있다.

SENSOR POSITIONING DRIVE COMMAND 종료 후 END STATUS 레지스터값은 0x1000가 된다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다. Sensor Positioning Drive I에서는 MARK 신호 입력 전에 가속 동작을 수행한다.

▶ +/- Sensor Positioning Drive II (CODE=0xB2, 0xB3)

CW(+) 또는 CCW(-) 방향으로 MARK 신호의 edge가 검출되면, 사용자가 지정한 펄스 수 만큼 구동하는 드라이브를 실행하기 위한 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 지정펄스 수(드라이브 중 출력할 펄스 수)를 저장한 후 +/- Sensor Positioning Drive II COMMAND를 실행한다. 지정 펄스 수 설정 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF)이고, DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 펄스 수를 설정하며 단위는 펄스 수이다.

드라이브 실행 후 지정펄스 수만큼의 펄스를 출력한 후 드라이브를 종료한다. 드라이브 동작 설정 명령(0xAD)의 입력 데이터의 2bit의 setting에 따라서 출력 펄스 수는 칩에서 출력되는 펄스 수를 카운트한 값을 사용할 수 있고, 외부 펄스 수(외부 위치 카운터 값)를 카운트한 값을 사용할 수 있다.

SENSOR POSITIONING DRIVE COMMAND 종료 후 END STATUS 레지스터값은 0x1000가 된다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

Sensor Positioning Drive II에서는 MARK 신호 입력 전에 Start/Stop Speed data로 등속한다.

▶ +/- Sensor Positioning Drive III (CODE=0xB4, 0xB5)

CW(+) 또는 CCW(-) 방향으로 MARK 신호의 edge가 검출되면, 사용자가 지정한 펄스 수 만큼 구동하는 드라이브를 실행하기 위한 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 지정펄스 수(드라이브 중 출력할 펄스 수)를 저장한 후 +/- Sensor Positioning Drive III COMMAND를 실행한다. 지정 펄스 수 설정 범위는 0~4,294,967,295(0x00000000~0xFFFFFFFF)이고, DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 펄스 수를 설정하며 단위는 펄스 수이다.

드라이브 실행 후 지정펄스 수만큼의 펄스를 출력한 후 드라이브를 종료한다. 드라이브 동작 설정 명령의 입력 데이터의 2bit의 setting에 따라서 출력 펄스 수는 칩에서 출력되는 펄스 수를 카운트한 값을 사용할 수 있고, 외부 펄스 수(외부 위치 카운터 값)를 카운트한 값을 사용할 수 있다.

SENSOR POSITIONING DRIVE COMMAND 종료 후 END STATUS 레지스터값은 0x000가 된다.

이 명령은 항상 실행 가능하지만 드라이브 구동 중(BUSY = High)일 경우는 명령 실행이 무효가 된다.

Sensor Positioning Drive III에서는 전체적으로 Start/Stop Speed data로 등속 동작을 수행한다.

▶ Soft limit 설정 READ (CODE = 0x36)

soft limit의 사용 여부 및 soft limit이 일어났을 때 drive 동작 종류, 그리고 soft limit에 사용될 비교 데이터를 설정하는 Soft limit 설정 WRITE (CODE = 0xB6)를 읽어보는 명령어이다.

▶ Soft limit 설정 WRITE (CODE = 0xB6)

soft limit의 사용 여부 및 soft limit이 일어났을 때 drive 동작 종류, 그리고 soft limit에 사용될 비교 데이터를 설정하는 명령어이다.

DATA4 port에 2<sup>2</sup>~2<sup>0</sup> bit 순으로 저장된 3-bit 크기 데이터로 설정하며 각 bit의 설정 내용은 다음과 같다.

2bit	Limit 비교 데이터 설정(0 : Internal count, 1 : External count)
1bit	Soft limit 시 동작 드라이브(0: 급정지, 1: 감속정지)
0bit	Soft limit Enable/Disable(1 : Enabel, 0 : Disable)

▶ +/- Soft limit 비교 레지스터 설정 READ (CODE = 0x37, 0x38)

+/- soft limit 비교를 위한 비교 값을 읽는 명령어이다.

▶ +/- Soft limit 비교 레지스터 설정 WRITE (CODE = 0xB7, 0xB8)

+/- soft limit 비교를 위한 비교 값을 설정하는 명령어이다.

▶ Trigger mode 설정 READ (CODE = 0x39)

CAMC-FS는 사용자가 설정한 거리 및 주기에 따라 설정한 Active Level을 가지는 신호를 외부 trigger pin으로 출력한다. 사용자는 이 command를 통해 출력될 trigger 신호의 펄스 폭 및 trigger 신호를 발생시킬 시점(비교데이터), 그리고 출력 모드에 대하여 읽어 볼 수 있다. Trigger 출력 모드를 위치로 선택한 경우 trigger 비교 데이터와 설정된 비교 대상값(internal 혹은 external 중 택)과 같을 경우 trigger 신호가 출력되며, 주기 모드일 경우 trigger 비교 데이터(0xBA) 만큼 비교 대상값(internal 혹은 External 중 택)이 증가 혹은 감소할 경우 trigger 신호가 출력된다.(참고. trigger 신호의 active level은 mode2 레지스터의 10bit에서 설정한다.) 이러한 trigger 신호는 vision application 에서 area scan(위치모드) 및 line scan CCD(주기모드)에 모두 적용할 수 있다. 특히 line scan CCD의 경우 매우 조밀한 위치 주기마다 이미지를 CAPTURE해야 하므로 반드시 필요한 기능이며, 별도의 카운터 보드를 쓰지 않아도 된다.

각 bit의 설정 내용은 Trigger mode 설정 WRITE (CODE = 0xB9)와 같다.

▶ Trigger mode 설정 WRITE (CODE = 0xB9)

출력될 trigger 신호의 펄스 폭 및 trigger 신호를 발생시킬 시점(비교데이터), 그리고 출력 모드에 대한 설정을 하는 명령어이다. Trigger 기능 초기화 및 disable은 trigger active level 출력시간을 0x0000로 하면 된다.

각 bit의 설정 내용은 다음과 같다.

31~16bit	Trigger active level 출력시간 설정 시간 = main clock 주기 x (Decimal value of 31~16bit)
15~2bit	Don' t care
1bit	Trigger 비교 count 지정(0 : Internal, 1 : External)
0bit	Trigger mode(0 : 위치 모드, 1 : 주기 모드)

▶ Trigger비교 데이터 설정 READ (CODE = 0x3A)

Trigger 비교 값 설정을 읽어 보는 명령어이다.

▶ Trigger비교 데이터 설정 WRITE (CODE = 0xBA)

Trigger 비교 값을 설정하는 명령어이다.

▶ INTERNAL COUNTER M\_DATA READ (CODE = 0x3B)

현재 설정되어 있는 내부 카운터 하한데이터(M\_DATA)를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되어 있다.

▶ INTERNAL COUNTER M\_DATA WRITE (CODE = 0xBB)

내부 카운터 하한데이터(M\_DATA)를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은 0x80000000을 가진다.

▶ EXTERNAL COUNTER M\_DATA READ (CODE = 0x3C)

현재 설정되어 있는 외부 카운터 하한데이터(M\_DATA)를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되어 있다.

▶ EXTERNAL COUNTER M\_DATA WRITE (CODE = 0xBC)

외부 카운터 하한데이터(M\_DATA)를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은 0x80000000을 가진다.

▶ 스크립트 동작 설정 레지스터-n (n=1, 2, 3) READ COMMAND ( CODE = 0x40, 0x41, 0x42 )

현재 설정되어 있는 스크립트 동작 설정 레지스터-n (n=1, 2, 3) 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되며, 32-bit 데이터의 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

▶ 스크립트 동작 설정 레지스터-n (n=1, 2, 3) WRITE COMMAND ( CODE = 0xC0, 0xC1, 0xC2 )

스크립트 동작 설정 레지스터-n (n=1, 2, 3)을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000을 가진다. 이벤트 검출 후 실행할 COMMAND의 사용 데이터는 동작 데이터 레지스터-n (n=1, 2, 3) (CODE = 0xC4, 0xC5, 0xC6)에 설정한다.

DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 저장된 32-bit 크기 데이터로 설정하며 각 bit가 설정 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

주) 스크립트 동작 설정 레지스터는 스크립트 동작 데이터 레지스터 설정 이후 입력되어야 한다. 그렇지 않을 경우 이전에 사용하였던 스크립트 동작 데이터가 스크립트 명령 수행시 사용되는 오류가 발생 할 수 있다.

검출 이벤트 설정 별첨

▶ 스크립트 동작 설정 레지스터-Queue READ COMMAND ( CODE = 0x43 )

현재 설정되어 있는 스크립트 동작 설정 레지스터-Queue 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되며, 32-bit 데이터의 내용은 다음과 같다. 스크립트 동작 설정 레지스터-Queue는 Depth가 13인 Queue로 구성되어 있다. 이 COMMAND에서 READ되는 데이터는 인덱스 0에 위치한 데이터, 가장 먼저 검출할 이벤트에 대한 데이터이다.

31bit	0 : 한번 실행, 1 : 계속 실행
30bit	실행시 Interrupt 발생 (0: Disable, 1: Enable)
29~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

▶ 스크립트 동작 설정 레지스터-Queue WRITE COMMAND ( CODE = 0xC3 )

스크립트 동작 설정 레지스터-Queue 을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000로 설정된다. DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 저장된 32-bit 크기 데이터로 설정하며 각 bit가 설정 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30bit	실행시 Interrupt 발생 (0: Disable, 1: Enable)
29~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

주) 스크립트 동작 설정 레지스터는 스크립트 동작 데이터 레지스터 설정 이후 입력되어야 한다. 그렇지 않을 경우 이전에 사용하였던 스크립트 동작 데이터가 스크립트 명령 수행시 사용되는 오류가 발생할 수 있다.

검출 이벤트 설정 별첨

스크립트 동작 설정 레지스터-Queue는 Depth가 13인 Queue(FIFO)로 구성되어 있다. 즉, 최대 13개의 연속적인 COMMAND를 입력하는 순서대로 실행할 수 있다. 단, 먼저 입력한 DATA에 의한 이벤트를 검출하여 COMMAND를 실행하면 다음 입력한 이벤트에 의한 COMMAND를 실행할 수 있다. 이벤트 검출 후 실행 할 COMMAND의 사용 DATA는 스크립트 동작 데이터 레지스터- Queue (CODE = 0xC3)에서 설정할 수 있으며 동작 설정 레지스터와 동작 데이터 레지스터는 항상 같은 횟수를 실행하여 각각의 Queue의 인덱스 값은 같은 값을 가지도록 설정하여야 한다. 실행할 COMMAND에 데이터가 필요없는 경우는 0x00000000값을 동작 데이터 레지스터에 입력하여 동작 설정 레지스터 Queue와 동작 데이터 레지스터 Queue의 인덱스 값은 서로 같은 값을 가지도록 설정해야 올바른 동작을 수행할 수 있다. COMMAND 실행 후 설정 레지스터의 31bit가 0이면 사용된 데이터는 자동으로 Clear된다.

설정된 Queue의 인덱스 값은 동작 설정 Queue 인덱스 READ 명령 (CODE = 0x49)을 실행하여 알 수 있으며, 동작 설정 Queue 및 동작 데이터 Queue의 Full/Empty 상태는 Queue Full/Empty Flag 레지스터 값을 READ (CODE = 4B)하여 확인할 수 있다.

사용 예를 들면 두 번 목표속도 데이터를 등속일 때 변경하고, 감속 시 시작속도 데이터를 변경하는 동작을 할 경우 다음과 같이 설정한다. 이 때 스크립트 동작 데이터 레지스터-4에는 변경할 목표 속도 데이터-1, 목표 속도 데이터-2, 시작 속도 데이터를 각각의 설정 데이터에 앞서 입력해야 한다.

설정 데이터: 등속시 변경될 목표 속도데이터-1 입력(CODE=0xC7)

0x0000D82: 등속 시 목표 속도 데이터 변경-1 (CODE = 0xC3)

설정 데이터: 등속시 변경될 목표 속도데이터-2 입력(CODE=0xC7)

0x00000D82 : 등속 시 목표 속도 데이터 변경-2 (CODE = 0xC3)

설정 데이터: 감속시 변경될 시작 속도데이터 입력(CODE=0xC7)

0x00000E81 : 감속 시 시작 속도 데이터 변경 (CODE = 0xC3)

▶ 스크립트 동작 데이터 레지스터-n (n=1, 2, 3) READ COMMAND ( CODE = 0x44, 0x45, 0x46 )

현재 설정되어 있는 스크립트 동작 데이터 레지스터-n (n=1, 2, 3) 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31}\sim 2^0$  bit 순으로 32-bit 크기로 저장되며, 32-bit 데이터는 각각 스크립트 동작 설정 레지스터-n (n=1, 2, 3)에 설정된 명령을 실행하는데 필요한 데이터이다.

▶ 스크립트 동작 데이터 레지스터-n (n=1, 2, 3) WRITE COMMAND ( CODE = 0xC4, 0xC5, 0xC6 )

스크립트 동작 데이터 레지스터-n (n=1, 2, 3)을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정된다. DATA1, DATA2, DATA3, DATA4 port에  $2^{31}\sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 각각 스크립트 동작 설정 레지스터-n (n=1, 2, 3)에 설정된 명령을 실행하는데 사용하는 데이터이다.

▶ 스크립트 동작 데이터 레지스터-Queue READ COMMAND ( CODE = 0x47 )

현재 설정되어 있는 스크립트 동작 데이터 레지스터-Queue 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31}\sim 2^0$  bit 순으로 32-bit 크기로 저장되며, 스크립트 동작 설정 레지스터-Queue에 설정된 명령을 실행하는데 필요한 데이터이다. 스크립트 동작 데이터 레지스터-Queue는 스크립트 동작 설정 레지스터-Queue와 같이 Depth가 13인 Queue로 구성되어 있으며 인덱스가 0인 위치의 값이 READ된다.

▶ 스크립트 동작 데이터 레지스터-Queue WRITE COMMAND ( CODE = 0xC7 )

스크립트 동작 설정 레지스터-Queue 을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정된다. DATA1, DATA2, DATA3, DATA4 port에  $2^{31}\sim 2^0$  bit 순으로 저장된 32-bit 크기 데이터로 설정하며 스크립트 동작 설정 레지스터-4에 설정된 명령을 실행하는데 필요한 데이터를 입력할 때 사용하는 COMMAND이다.

▶ 스크립트 동작 설정/데이터 레지스터-Queue CLEAR COMMAND ( CODE = 0xC8 )

스크립트 동작 설정/데이터 레지스터-Queue의 모든 데이터를 초기화 하는 COMMAND이다. 실행 후 Queue에 저장된 데이터는 Clear되고 인덱스 값도 0으로 초기화된다.

▶ 스크립트 동작 설정 Queue 인덱스 READ COMMAND ( CODE = 0x49 )

현재 동작하고 있는 스크립터 동작 설정 레지스터-Queue의 인덱스(저장된 데이터 개수)값을 읽어내는 COMMAND이다. 읽는 DATA는 DATA4 port에  $2^3\sim 2^0$  bit로 저장된 4-bit 크기로 저장된다.

단, 스크립트 동작 설정을 하고자 하면 먼저 동작 데이터를 써 넣어야 한다.

▶ 스크립트 동작 데이터 Queue 인덱스 READ COMMAND ( CODE = 0x4A )

현재 동작하고 있는 스크립트 동작 데이터 레지스터-Queue의 인덱스(저장된 데이터 개수)값을 읽어내는 COMMAND이다. 읽는 DATA는 DATA4 port에 2<sup>3</sup>~2<sup>0</sup>bit로 저장된 4-bit 크기로 저장된다.

▶ 스크립트 Queue Full/Empty Flag READ COMMAND ( CODE = 0x4B )

스크립트 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty Flag 레지스터 값을 읽어내는 COMMAND이다. 4 BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

3bit	— 스크립트 동작 설정 레지스터-Queue Full Flag
2bit	— 스크립트 동작 설정 레지스터-Queue Empty Flag
1bit	— 스크립트 동작 데이터 레지스터-Queue Full Flag
0bit	— 스크립트 동작 데이터 레지스터-Queue Empty Flag

▶ 스크립트 Queue size 설정(1~13) READ COMMAND (0x4C)

스크립트 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty size 레지스터 값을 읽어내는 COMMAND이다. 16BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

15~12bit	— 스크립트 동작 설정 레지스터-Queue Full size
11~8bit	— 스크립트 동작 설정 레지스터-Queue Empty size
7~4bit	— 스크립트 동작 데이터 레지스터-Queue Full size
3~0bit	— 스크립트 동작 데이터 레지스터-Queue Empty size

▶ 스크립트 Queue size 설정(1~13) WRITE COMMAND (0xCC)

스크립트 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty size 레지스터 값을 설정하는 COMMAND이다. RESET 후 초기값은 0xD0D0으로 설정된다. DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 저장된 16-bit 크기 데이터로 설정된다.

15~12 bit에는 스크립트 동작 설정 Queue의 full size 레지스터 값을 설정하여 설정 인덱스 값보다 크거나 같으면 스크립트 동작 설정 레지스터-Queue Full Flag가 발생하게 한다.

11~8 bit에는 스크립트 동작 설정 Queue의 empty size 레지스터 값을 설정하여 설정 인덱스 값보다 작거나 같으면 스크립트 동작 설정 레지스터-Queue Empty Flag가 발생하게 한다.

7~4 bit에는 스크립트 동작 데이터 Queue의 full size 레지스터 값을 설정하여 설정 인덱스 값보다 크거나 같으면 스크립트 동작 데이터 레지스터-Queue Full Flag가 발생하게 한다.

3~0 bit에는 스크립트 동작 데이터 Queue의 empty size 레지스터 값을 설정하여 설정 인덱스 값보다 작거나 같으면 스크립트 동작 데이터 레지스터-Queue Empty Flag가 발생하게 한다.

▶ 갈무리 동작 설정 레지스터-n (n=1, 2, 3) READ COMMAND ( CODE = 0x50, 0x51, 0x52 )

현재 설정되어 있는 갈무리 동작 설정 레지스터-n (n=1, 2, 3) 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되며, 32-bit 데이터의 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30bit	실행시 Interrupt 발생 (0: Disable, 1: Enable)
29~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

▶ 갈무리 동작 설정 레지스터-n (n=1, 2, 3) WRITE COMMAND ( CODE = 0xD0, 0xD1, 0xD2 )

갈무리 동작 설정 레지스터-n (n=1, 2, 3)을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정된다.

갈무리 동작 설정에 사용하는 COMMAND는 READ COMMAND만 사용할 수 있다. 이벤트 검출 후 READ한 데이터는 갈무리 동작 데이터 레지스터-n (n=1, 2, 3)에 각각 저장되며, 이벤트 검출 후 31bit가 0이면 설정 값은 자동으로 Clear된다. 이벤트 검출 후 READ COMMAND 실행에 의해 저장된 갈무리 동작 데이터 레지스터-n (n=1, 2, 3) 값은 사용자가 갈무리 동작 데이터 레지스터-n (n=1, 2, 3) READ COMMAND를 실행하면 자동으로 Clear된다.

DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 저장된 32-bit 크기 데이터로 정하며 각 bit가 설정 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30bit	실행시 Interrupt 발생 (0: Disable, 1: Enable)
29~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

▶ 갈무리 동작 설정 레지스터-Queue READ COMMAND ( CODE = 0x53 )

현재 설정되어 있는 갈무리 동작 설정 레지스터-Queue 데이터를 읽어내는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정된다.

갈무리 동작 설정 레지스터-4는 Depth가 13인 Queue로 구성되어 있다. 갈무리 동작 설정 레지스터-Queue READ COMMAND 실행 시 인덱스 0에 해당하는(가장 먼저 입력한) 값을 읽을 수 있다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 32-bit 크기로 저장되며, 32-bit 데이터의 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30bit	실행시 Interrupt 발생 (0: Disable, 1: Enable)
29~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

▶ 갈무리 동작 설정 레지스터-Queue WRITE COMMAND ( CODE = 0xD3 )

갈무리 동작 설정 레지스터- Queue을 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정된다. DATA1, DATA2, DATA3, DATA4 port에 2<sup>31</sup>~2<sup>0</sup> bit 순으로 저장된 32-bit 크기 데이터로 설정하며 각 bit가 설정 내용은 다음과 같다.

31bit	0 : 한번 실행, 1 : 계속 실행
30~26bit	Don' t care
25~24bit	검출된 두 이벤트 결과의 연산 00 : 논리연산 없음(1번 이벤트만 검출) 01 : OR 연산 10 : AND 연산 11 : XOR 연산
23~16bit	2번 검출 이벤트 설정
15~8bit	1번 검출 이벤트 설정
7~0bit	이벤트 검출 후 실행할 COMMAND

검출 이벤트 설정 별첨

갈무리 동작 설정 레지스터-Queue는 Depth가 13인 Queue(FIFO)로 구성되어 있다. 즉 최대 13개의 순차적인 갈무리 동작을 실행할 수 있다. 이벤트 검출 후 READ COMMAND에 의해 읽은 데이터는 갈무리 동작 데이터 레지스터-Queue에 저장된다. 갈무리 동작 데이터 레지스터-Queue 값은 사용자가 READ할

때 한 개씩 순서대로 Queue 값이 Clear된다.

▶ 갈무리 동작 데이터 레지스터-n (n=1, 2, 3) READ COMMAND ( CODE = 0x54, 0x55, 0x56 )

갈무리 동작 데이터 레지스터-n (n=1, 2, 3) 데이터를 읽어내는 COMMAND이다.

갈무리 동작 데이터 레지스터-n (n=1, 2, 3)은 갈무리 동작 설정 레지스터-n (n=1, 2, 3)에서 설정한 COMMAND(READ COMMAND) 실행 시 READ된 값이 각각 저장되는 레지스터이다. 이때 새로운 데이터가 갱신되었다는 정보는 caption data flag(0x5B, 6~4bit)를 읽어 확인 할 수 있다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되며, 갈무리 동작 설정에 저장된 명령 실행 후 저장된 데이터이다. 읽기 전용 레지스터이며, 한번 Read하면 caption data flag가 자동으로 0가 된다.

▶ 갈무리 동작 데이터 레지스터-Queue READ COMMAND ( CODE = 0x57 )

현재 설정되어 있는 갈무리 동작 데이터 레지스터-Queue 데이터를 읽어내는 COMMAND이다.

갈무리 동작 데이터 레지스터-Queue는 갈무리 동작 설정 레지스터-Queue에서 설정한 COMMAND (READ COMMAND) 실행 시 READ된 값이 저장되는 레지스터이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되며, 갈무리 동작 설정에 저장된 명령 실행 후 저장된 데이터이다. 읽기 전용 레지스터이며, 한번 Read하면 queue의 인덱스 0에 있는 데이터는 자동으로 지원지며 인덱스가 1 감소한다.

▶ 갈무리 동작 설정/데이터 레지스터-Queue CLEAR COMMAND ( CODE = 0xD8 )

갈무리 동작 설정/데이터 레지스터-Queue의 모든 데이터를 초기화 하는 COMMAND이다. 실행 후 Queue에 저장된 데이터는 Clear되고 인덱스 값도 0으로 초기화된다.

▶ 갈무리 동작 설정 Queue 인덱스 READ COMMAND ( CODE = 0x59 )

현재 동작하고 있는 갈무리 동작 설정 레지스터-Queue의 저장된 인덱스(저장된 데이터 개수)값을 읽어내는 COMMAND이다. 읽는 DATA는 DATA4 port에  $2^3 \sim 2^0$  bit로 저장된 4-bit 크기로 저장된다.

▶ 갈무리 동작 데이터 Queue 인덱스 READ COMMAND ( CODE = 0x5A )

현재 동작하고 있는 갈무리 동작 데이터 레지스터-Queue의 저장된 인덱스(저장된 데이터 개수)값을 읽어내는 COMMAND이다. 읽는 DATA는 DATA4 port에  $2^3 \sim 2^0$  bit로 저장된 4-bit 크기로 저장된다.

▶ 갈무리 Queue Full/Empty Flag READ COMMAND ( CODE = 0x5B )

갈무리 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty Flag 레지스터 값 및 갈무리 설정 레지스터-1,2,3의 명령어가 수행되어 새로운 데이터가 갈무리 동작 데이터 레지스터에 입력될 때 발생하는 flag를 읽어내는 COMMAND이다. 7 BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

6bit	갈무리 동작 데이터 레지스터-3 DATA Flag
5bit	갈무리 동작 데이터 레지스터-2 DATA Flag
4bit	갈무리 동작 데이터 레지스터-1 DATA Flag
3bit	갈무리 동작 설정 레지스터-Queue Full Flag
2bit	갈무리 동작 설정 레지스터-Queue Empty Flag
1bit	갈무리 동작 데이터 레지스터-Queue Full Flag
0bit	갈무리 동작 데이터 레지스터-Queue Empty Flag

▶ 갈무리 Queue size 설정(1~13) READ COMMAND (0x5C)

갈무리 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty size 레지스터 값을 읽어내는 COMMAND 이다. 16BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

15~12bit	갈무리 동작 설정 레지스터-Queue Full size
11~8bit	갈무리 동작 설정 레지스터-Queue Empty size
7~4bit	갈무리 동작 데이터 레지스터-Queue Full size
3~0bit	갈무리 동작 데이터 레지스터-Queue Empty size

▶ 갈무리 Queue size 설정(1~13) WRITE COMMAND (0xDC)

갈무리 동작 설정/동작 데이터 레지스터-Queue의 Full/Empty size 레지스터 값을 설정하는 COMMAND 이다. RESET 후 초기값은 0xD0D0으로 설정된다. DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 저장된 16-bit 크기 데이터로 설정된다.

15~12 bit에는 갈무리 동작 설정 Queue의 full size 레지스터 값을 설정하여 설정 인덱스 값보다 크거나 같으면 갈무리 동작 설정 레지스터-Queue Full Flag가 발생하게 한다.

11~8 bit에는 갈무리 동작 설정 Queue의 empty size 레지스터 값을 설정하여 설정 인덱스 값보다 작거나 같으면 갈무리 동작 설정 레지스터-Queue Empty Flag가 발생하게 한다.

7~4 bit에는 갈무리 동작 데이터 Queue의 full size 레지스터 값을 설정하여 설정 인덱스 값보다 크거나 같으면 갈무리 동작 데이터 레지스터-Queue Full Flag가 발생하게 한다.

3~0 bit에는 갈무리 동작 데이터 Queue의 empty size 레지스터 값을 설정하여 설정 인덱스 값보다 작거나 같으면 갈무리 동작 데이터 레지스터-Queue Empty Flag가 발생하게 한다.

▶ \* 스크립트 및 갈무리 기능에 사용될 검출 이벤트

검출할 동작 이벤트	이벤트 동작 비트 설정값
검출사항 없음	0x00
드라이브 종료	0x01
지정펄스 수 드라이브 시작	0x02
지정펄스 수 드라이브 종료	0x03
연속 드라이브 시작	0x04

검출할 동작 이벤트	이벤트 동작 비트 설정값
연속 드라이브 종료	0x05
신호 검출-1 드라이브 시작	0x06
신호 검출-1 드라이브 종료	0x07
신호 검출-2 드라이브 시작	0x08
신호 검출-2 드라이브 종료	0x09
원점검출 드라이브 시작	0x0A
원점검출 드라이브 종료	0x0B
가속	0x0C
등속	0x0D
감속	0x0E
내부위치카운터 > 내부위치비교값	0x0F
내부위치카운터 = 내부위치비교값	0x10
내부위치카운터 < 내부위치비교값	0x11
외부위치카운터 > 외부위치비교값	0x12
외부위치카운터 = 외부위치비교값	0x13
외부위치카운터 < 외부위치비교값	0x14
외부펄스카운터 > 외부펄스카운터비교값	0x15
외부펄스카운터 = 외부펄스카운터비교값	0x16
외부펄스카운터 < 외부펄스카운터비교값	0x17
현재속도데이터 > 현재속도비교데이터	0x18
현재속도데이터 = 현재속도비교데이터	0x19
현재속도데이터 < 현재속도비교데이터	0x1A
현재속도데이터 > Rate Change Point 1-2	0x1B
현재속도데이터 = Rate Change Point 1-2	0x1C
현재속도데이터 < Rate Change Point 1-2	0x1D
현재속도데이터 > Rate Change Point 2-3	0x1E
현재속도데이터 = Rate Change Point 2-3	0x1F
현재속도데이터 < Rate Change Point 2-3	0x20
현재속도데이터 = 목표속도데이터	0x21
현재속도데이터 = 시작속도데이터	0x22
급속정지	0x23
감속정지	0x24
+ Emergency Limit 신호 입력	0x25
- Emergency Limit 신호 입력	0x26

검출할 동작 이벤트	이벤트 동작 비트 설정값
+ Slow Down Limit 신호 입력	0x27
- Slow Down Limit 신호 입력	0x28
탈조 에러 발생	0x29
데이터 설정 에러 발생	0x2A
Alarm 신호 입력	0x2B
급속 정지 명령 실행	0x2C
감속 정지 명령 실행	0x2D
급속 정지 신호 입력	0x2E
감속 정지 신호 입력	0x2F
Emergency Limit 신호 입력	0x30
Slow Down Limit 신호 입력	0x31
Inposition 신호 입력	0x32
IN0 High 신호 입력	0x33
IN0 Low 신호 입력	0x34
IN1 High 신호 입력	0x35
IN1 Low 신호 입력	0x36
IN2 High 신호 입력	0x37
IN2 Low 신호 입력	0x38
IN3 High 신호 입력	0x39
IN3 Low 신호 입력	0x3A
OUT0 High 신호 출력	0x3B
OUT0 Low 신호 출력	0x3C
OUT1 High 신호 출력	0x3D
OUT1 Low 신호 출력	0x3E
OUT2 High 신호 출력	0x3F
OUT2 Low 신호 출력	0x40
OUT3 High 신호 출력	0x41
OUT3 Low 신호 출력	0x42
Sensor Positioning drive I 시작	0x43
Sensor Positioning drive I 종료	0x44
Sensor Positioning drive II 시작	0x45
Sensor Positioning drive II 종료	0x46
Sensor Positioning drive III 시작	0x47
Sensor Positioning drive III 종료	0x48

검출할 동작 이벤트	이벤트 동작 비트 설정값
1'st counter N-data count clear	0x49
2'nd counter N-data count clear	0x4A
Mark# signal high	0x4B
Mark# signal low	0x4C
EUIO0 High 신호 입력/출력	0x4D
EUIO0 Low 신호 입력/출력	0x4E
EUO1 High 신호 출력	0x4F
EUO1 Low 신호 출력	0x50
EUO2 High 신호 출력	0x51
EUO2 Low 신호 출력	0x52
EUO3 High 신호 출력	0x53
EUO3 Low 신호 출력	0x54
EUO4 High 신호 출력	0x55
EUO4 Low 신호 출력	0x56
+ Software LIMIT	0X57
- Software LIMIT	0X58
Software LIMIT	0X59
TRIGGER ENABLE	0X5A
INTERRUPT GENERATED BY ANY SOURCE	0X5B
INTERRUPT GENERATED BY COMMAND "F9"	0X5C
PRESET 삼각구동 시작	0X5D
드라이브 busy High	0x5E
드라이브 busy Low	0x5F
무조건 수행(Queue COMMAND 한정)	0XFF

▶ INTERNAL COUNTER DATA READ COMMAND ( CODE = 0x60 )

현재 설정되어 있는 내부 위치 카운터의 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다. 이 데이터를 읽어 원점으로부터 현재 위치까지 이동한 총 펄스 수 또는 이동 거리를 산출할 수 있다. 읽은 데이터는 2진 보수 형식이다.

▶ INTERNAL COUNTER DATA WRITE COMMAND ( CODE = 0xE0 )

내부 위치 카운터의 값을 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 내부 위치 카운터 값을 설정한다. 설정된 위치값을 시점으로 내부 펄스의 출력 시 업/다운 카운트 동작을 수행한다. RESET 후 초기값은 0x00000000을 가진다.

기입하는 데이터는 2진 보수 형식으로서 설정 범위는  $-2,147,483,648 \sim 2,147,483,647$  (  $0x80000000 \sim 0x7FFFFFFF$  ) 이다. 모터를 정회전 시키는 펄스 출력 시 ( CW 방향으로 드라이브 실행 ) ( + ) 값으로 증가하고, 역회전 시키는 펄스 출력 시 ( CCW 방향으로 드라이브 실행 ) ( - ) 값으로 증가한다.

▶ INTERNAL COUNTER COMPARATE DATA READ COMMAND ( CODE = 0x61 )

현재 설정되어 있는 내부 위치 비교값을 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

이 데이터는 내부 위치 카운터 값과 비교할 때 사용한다. 비교 결과는 외부 핀 ( ICL, ICE, ICG ) 및 drive status 레지스터 read(0x76)를 통해 알 수 있다. 읽은 데이터는 2진 보수 형식이다.

▶ INTERNAL COUNTER COMPARATE DATA WRITE COMMAND ( CODE = 0xE1 )

내부 위치 카운터 값과 비교 연산을 실행할 데이터인 내부 위치 비교값을 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 비교값을 설정한다. RESET 후 초기값은  $0x00000000$ 을 가진다. 기입하는 데이터는 2진 보수 형식으로서 설정 범위는  $-2,147,483,648 \sim 2,147,483,647$  (  $0x80000000 \sim 0x7FFFFFFF$  ) 이다.

▶ INTERNAL COUNTER PRE-SCALE DATA READ COMMAND ( CODE = 0x62 )

현재 설정되어 있는 내부 위치 카운터의 Scale 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA4 port에  $2^7 \sim 2^0$  bit 순으로 8-bit 크기로 저장되어 있다.

▶ INTERNAL COUNTER PRE-SCALE DATA WRITE COMMAND ( CODE = 0xE2 )

내부 위치 카운터 값의 Scale을 설정하는 데이터를 기입하는 COMMAND이다. DATA4 port에 설정된 8-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은  $0x00$ 을 가진다.

CW 방향으로 설정한 데이터의 배수만큼의 펄스 출력 시에 내부 위치 카운터 값이 +1 증가한다. 반대 방향인 경우는 -1 감소하게 된다.

이 데이터의 설정은 사용하는 외부 엔코더의 체배 수가 다를 경우 이 데이터를 변경하여 내부 위치 카운터와 외부 위치 카운터가 같은 비율의 카운트 동작을 수행하도록 할 수 있다.

예를 들어 모터가 10,000펄스 당 1바퀴 움직이는 경우 엔코더 입력을 단상 신호로 받는 경우 내부 위치 카운터가 10,000개 증가할 때, 외부 위치 카운터 값은 2,500개 증가한다. 이 값을 제어 프로그램에서 사용할 경우 외부 위치 카운터 값에 \*4 연산을 하여 사용해야 한다. 이 경우 내부 Scale 데이터를 4로 설정하고 동작시키는 경우 내부 위치 카운터값이 10,000개 증가할 때 외부 위치 카운터 값도 같은 스케일로 10,000개 증가하기 때문에 프로그램에서 어떤 값을 곱할 필요 없이 그대로 사용하면 된다.

▶ INTERNAL COUNTER P-DATA READ COMMAND ( CODE = 0x63 )

현재 설정되어 있는 내부 카운터 상한데이터(P\_DATA)를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

▶ INTERNAL COUNTER P-DATA WRITE COMMAND ( CODE = 0xE3 )

내부 카운터 상한데이터(P\_DATA)를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은 0x7FFFFFFF을 가진다.

▶ EXTERNAL COUNTER DATA READ COMMAND ( CODE = 0x64 )

현재 설정되어 있는 외부 위치 카운터의 데이터를 읽어내는 COMMAND이다.

읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다. 이 데이터를 읽어 원점으로부터 현재 위치까지 이동한 총 펄스 수 또는 이동 거리를 산출할 수 있다. 읽은 데이터는 2진 보수 형식이다.

▶ EXTERNAL COUNTER DATA WRITE COMMAND ( CODE = 0xE4 )

외부 위치 카운터의 값을 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 외부 위치 카운터 값을 설정한다. RESET 후 초기값은 0x00000000을 가진다. MODE 2 레지스터의 외부 위치 카운터의 입력 사양 변경 시 외부 위치 카운터 값이 변경될 수 있으니 MODE 2 레지스터의 7, 6 BIT (외부 위치 카운터 입력 사양) 변경 시 외부 위치 카운터 값을 다시 설정할 필요가 있다.

설정된 위치 값을 시점으로 외부 펄스의 출력 시 업/다운 카운트 동작을 수행한다.

기입하는 데이터는 2진 보수 형식으로서 설정 범위는  $-2,147,483,648 \sim 2,147,483,647$  (  $0x80000000 \sim 0x7FFFFFFF$  ) 이다. 모터를 정회전 시키는 펄스 출력 시 ( CW 방향으로 드라이브 실행 ) (+) 값으로 증가하고, 역회전 시키는 펄스 출력 시 ( CCW 방향으로 드라이브 실행 ) (-) 값으로 증가한다.

▶ EXTERNAL COUNTER COMPARE DATA READ COMMAND ( CODE = 0x65 )

현재 설정되어 있는 외부 위치 비교값을 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

이 데이터는 외부 위치 카운터 값과 비교할 때 사용한다. 비교 결과는 외부 핀 ( ECL, ECE, ECG ) 및 drive status 레지스터 read(0x73)를 통해 알 수 있다. 읽은 데이터는 2진 보수 형식이다.

▶ EXTERNAL COUNTER COMPARE DATA WRITE COMMAND ( CODE = 0xE5 )

외부 위치 카운터 값과 비교 연산을 실행할 데이터인 외부 위치 비교값을 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 비교값을 설정한다. RESET 후 초기값은 0x00000000이며 기입하는 데이터는 2진 보수 형식으로서 설정 범위는  $-2,147,483,648 \sim 2,147,483,647$  (  $0x80000000 \sim 0x7FFFFFFF$  ) 이다.

▶ EXTERNAL COUNTER PRE-SCALE DATA READ COMMAND ( CODE = 0x66 )

현재 설정되어 있는 외부 위치 카운터의 Scale 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

▶ EXTERNAL COUNTER PRE-SCALE DATA WRITE COMMAND ( CODE = 0xE6 )

외부 위치 카운터 값의 Scale을 설정하는 데이터를 기입하는 COMMAND이다. DATA4 port에 설정된 8-bit 크기의 데이터로 설정할 수 있으며 RESET 후 초기값은 0x00을 가진다. CW 방향으로 설정한 데이터의 배수만큼의 펄스 출력 시에 내부 위치 카운터 값이 +1 증가한다. 반대 방향인 경우는 -1 감소하게 된다.

이 데이터 설정은 사용하는 외부 엔코더의 체배수가 다를 경우 이 데이터를 변경하여 내부 위치 카운터와 외부 위치 카운터가 같은 비율의 카운트 동작을 수행하도록 할 수 있다.

예를 들어 모터가 2,500펄스 당 1바퀴 움직이는 경우 엔코더 입력을 이상 4체배 신호로 받는 경우 내부 위치 카운터가 2,500개 증가할 때, 외부 위치 카운터 값은 10,000개 증가한다. 이 값을 제어 프로그램에서 사용할 경우 내부 위치 카운터 값에 \*4 연산을 하여 사용해야 한다. 이 경우 외부 Scale 데이터를 4로 설정하고 동작시키는 경우 내부 위치 카운터 값이 10,000개 증가할 때 외부 위치 카운터 값도 같은 스케일로 10,000개 증가하기 때문에 프로그램에서 어떤 값을 곱할 필요 없이 그대로 사용하면 된다.

▶ EXTERNAL COUNTER P-DATA READ COMMAND ( CODE = 0x67 )

현재 설정되어 있는 외부 카운터 상한데이터(P\_DATA)를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

▶ EXTERNAL COUNTER N-COUNT DATA WRITE COMMAND ( CODE = 0xE7 )

내부 카운터 상한데이터(P\_DATA)를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은 0x7FFFFFFF을 가진다.

▶ EXTERNAL SPEED DATA READ COMMAND ( CODE = 0x68 )

현재 출력 엔코더에서 출력되는 펄스 속도를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되며 읽은 값을 V라고 할 경우 엔코더 출력 펄스의 주기는( 50% duty의 일정한 펄스가 출력되었을 경우 )  $V * T_{clk}$  sec이다.  $T_{clk}$ 는 CAMC-FS 칩의 입력 클럭의 한 주기 시간이다. 속도[pps]는 주기값의 역수를 구하면 된다. 엔코더 신호의 속도 데이터는 드라이브 중 입력되는 신호의 주기값을 읽어낸다. 드라이브 종료 시 overflow가 발생한 최고 값을 다음 드라이브 시작 전까지 레치하게 된다. 설계시 단상, 체배모드에 따라서 카운트하는 시점을 주의하여야 한다.

▶ EXTERNAL SPEED COMPARATE DATA READ COMMAND ( CODE = 0x69 )

현재 설정된 외부 펄스 속도 데이터와 비교할 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA1, DATA2, DATA3, DATA4 port에  $2^{31} \sim 2^0$  bit 순으로 32-bit 크기로 저장되어 있다.

▶ EXTERNAL SPEED COMPARATE DATA WRITE COMMAND ( CODE = 0xE9 )

외부 엔코더 출력 펄스 속도와 비교할 데이터를 설정하는 COMMAND이다. DATA1, DATA2, DATA3, DATA4 port에 설정된 32-bit 크기의 데이터로 설정할 수 있다. RESET 후 초기값은 0x00000000을 가진다.

▶ 외부 센서 필터 대역폭 설정 DATA READ COMMAND ( CODE = 0x6A )

설정된 외부 센서 필터 대역폭 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA4 port에  $2^7 \sim 2^0$  bit 순으로 8-bit 크기로 저장되어 있다. 외부 센서 신호 신호의 Active Level 주기가 [설정된 데이터 \*  $T_{clk}$ (입력 클럭 한 주기)] 이상 일 때 칩 내부적으로 센서 신호의 변화를 인식하게 된다. 이 값이 0일 경우는 외부 센서신호는 칩 내부에서 delay 없이 즉시 반영되며,  $T_{clk}$ (입력 클럭 한 주기) 주기 이하의 노이즈에 대하여 영향을 받게 된다.

▶ 외부 센서 필터 대역폭 설정 DATA WRITE COMMAND ( CODE = 0xEA )

외부 센서 필터 대역폭을 설정하는 COMMAND이다. DATA4 port에  $2^7 \sim 2^0$  bit 순으로 8-bit 크기의 데이터로 설정한다. ( 설정한 데이터 \*  $T_{clk}$  )의 시간 이상의 Active Level을 가지는 센서 신호가 유효하게 된다. 따라서 사용하는 센서 신호의 감도에 따라서 입력 신호의 감도를 Software적으로 조정하여 사용할 수 있다. RESET 후 초기값은 0x05로 설정된다. 설정된 대역 폭에 영향을 받는 외부 신호는 다음과 같다. (ESTOP, SSTOP, PELM, NELM, PSLM, NSLM, ALM, INP, SYNC, MONI, MARK, 8\_16SEL, IN0, IN1, IN2, IN3, EUIO0)

주) Signal search drive I, II 및 Sensor positioning drive I, II, III 에 사용되는 검출 센서 신호들과 MARK 입력신호의 필터 적용 유무는 칩 기능 설정 레지스터 설정 명령(0xF0, 11 bit)으로 설정 된다.

▶ OFF-RANGE DATA READ COMMAND ( CODE = 0x6B )

탈조 검출 DATA를 읽어내는 COMMAND이다. 읽은 DATA는 DATA4 port에  $2^7 \sim 2^0$  bit 순으로 8-bit 크기로 저장된다.

▶ OFF-RANGE DATA WRITE COMMAND ( CODE = 0xEB )

탈조 검출 DATA를 설정하는 COMMAND이다. DATA4 port에  $2^7 \sim 2^0$  bit 순으로 8-bit 크기의 데이터로 설정하며, RESET 후 초기값은 0x00을 가진다.

탈조 검출 DATA는 내부 위치 카운터와 외부 위치 카운터의 차이 값을 의미하며 내부 위치 카운터와 외부 위치 카운터의 차이값이 설정한 탈조 검출 DATA보다 큰 경우 구동 드라이브는 급정지, 또는 감속정지를 하게 되고, 내부 위치 카운터와 외부 위치 카운터의 차이값을 설정한 탈조 검출 DATA보다

작도록 다시 설정하여야 다음 드라이브를 구동할 수 있다. 내부/외부 위치 카운터의 탈조에 의해 드라이브가 급정지한 경우 END STATUS 레지스터의 8bit가 1(high)이 된다. 이 설정값을 0x00으로 설정하면 탈조에 의한 정지기능은 사용하지 않게 된다.

주) 탈조에 의한 급정지/감속정지 선택 : 집기능 설정 레지스터 7bit(0: 급정지, 1: 감속 정지)

▶ DEVIATION DATA READ COMMAND ( CODE = 0x6C )

( 내부 위치 카운터값 - 외부 위치 카운터값 )에 의해 산출되는 편차량 절대값 데이터를 읽어내는 COMMAND이다. 읽은 DATA는 DATA3, DATA4 port에 2<sup>15</sup>~2<sup>0</sup> bit 순으로 16-bit 크기로 저장된다. 읽어낸 데이터는 ( 내부 위치 카운터값 - 외부 위치 카운터값 ) 의 하위 15-bit 크기의 편차량값( 14 bit ~ 0 bit )과 부호(15 bit)로 구성되어 있다. 단, 편차량의 크기가 32,767을 초과하는 경우의 데이터는 신뢰성이 없으므로 주의가 필요하다.

▶ PGM REGISTER CHANGE DATA READ COMMAND ( CODE = 0x6D )

현재 사용하고 있는 펄스 설정 파라미터 레지스터 뱅크 값을 읽어내는 COMMAND이다. 읽은 DATA는 DATA4 port에 2<sup>0</sup> bit로 1-bit 크기로 저장된다.

이 값이 0 일 경우 PGM-1 Register에 설정된 파라미터를 이용하여 펄스 출력을 실행하고, 1일 경우는 PGM-Update Register에 설정된 파라미터를 이용하여 펄스 출력을 실행하고 있다.

▶ PGM REGISTER CHANGE DATA WRITE COMMAND ( CODE = 0xED )

펄스 출력에 필요한 파라미터를 저장하고 있는 레지스터 뱅크 값을 설정하는 COMMAND이다. RESET 후 초기값은 0x0를 가진다. 레지스터 뱅크 값을 0으로 설정한 경우 PGM-1 Register 군에 설정된 파라미터를 이용하여 펄스 출력에 사용하며, 1로 설정한 경우 PGM-Update Register 군에 설정된 파라미터를 이용하여 펄스 출력에 사용한다. 드라이브 구동 중 사용 이 Register 설정 값을 변경하는 경우는 펄스 출력 속도의 갑작스런 변경이 발생할 수 있으므로 주의가 필요하다.

▶ COMPARE REGISTER INPUT CHANGE READ COMMAND ( CODE = 0x6E )

CAMC-FS에는 위치 정보의 비교기가 내부/외부위치에 대하여 2개가 있다. 내부 위치 비교기를 1'st, 외부 위치 비교기를 2'nd라고 하였을 때 그 입력들을 사용자는 변경 할 수 있다. 예를 들자면, COMPARE REGISTER INPUT CHANGE 값을 0x1로 설정하면 내부 위치 정보에 대하여 두개의 다른 비교값을 사용할 수 있다. 명령어 0x6E를 사용하여 Comparator register의 입력사양 기능을 설정한 데이터를 읽어낸다. 설정 데이터는 2-bit 크기이며, 각 비트가 설정하는 내용은 다음과 같다.

00 : 1'st : Internal count, 2'nd : External count(기본값)

01 : 1'st : Internal count, 2'nd : Internal count(내부 위치에 대하여 두개의 비교값을 설정)

10 : 1'st : External count, 2'nd : External count(외부 위치에 대하여 두개의 비교값을 설정)

▶ COMPARE REGISTER INPUT CHANGE WRITE COMMAND ( CODE = 0xEE )

Comparator register의 입력사양을 설정하고자 할 때 사용하는 COMMAND이다. 설정 데이터는 2-bit 크기이며, 각 비트가 설정하는 내용은 다음과 같다.

- 00 : 1'st : Internal count, 2'nd : External count
- 01 : 1'st : Internal count, 2'nd : Internal count
- 10 : 1'st : External count, 2'nd : External count

▶ 칩 기능 설정 DATA READ COMMAND ( CODE = 0x70 )

CAMC-FS 칩 구동 시 필요한 기능을 설정한 데이터를 읽어내는 COMMAND이다. 읽은 데이터는 13-bit 크기를 가지며 각 bit의 설정 내용은 칩 기능 설정 DATA WRITE COMMAND ( CODE = 0xF0 )와 동일하다.

▶ 칩 기능 설정 DATA WRITE COMMAND ( CODE = 0xF0 )

CAMC-FS 칩 구동 시 필요한 기능을 설정하는 COMMAND이다. 설정 데이터 크기는 13-bit이며 이 명령으로 Inposition 대기모드, Alarm 급정지 사용 여부, 펄스 출력 종료 시 인터럽트 신호 출력 여부, Limit 급/감속 정지 기능 사용 여부와 감속 정지 시의 동작 설정 등을 할 수 있다. RESET 후 초기값은 0x0C3E로 설정되며, 각 동작 설정과 해당 비트의 관계는 다음과 같다.

12bit	— S profile 삼각구동 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(0)
11bit	— Search source filter 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
10bit	— Sync 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
9bit	— Limit 완전정지(0: 사용 하지 않음, 1: 사용)	초기값(0)
8bit	— Inposition 대기 확장모드(0: 사용 하지 않음, 1: 사용)	초기값(0)
7bit	— 탈조 급정지/감속정지 설정(0: 급정지, 1: 감속정지)	초기값(0)
6bit	— 감속정지 설정(0: 정지속도에서 정지, 1:정지속도에서 등속)	초기값(0)
5bit	— +-Slowdown limit sensor 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
4bit	— +-Emergency limit sensor 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
3bit	— ESTOP, SSTOP signal 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
2bit	— Don't care	초기값(1)
1bit	— Alarm stop 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(1)
0bit	— Inpositoin 대기 사용 설정(0: 사용 하지 않음, 1: 사용)	초기값(0)

▶ MODE 1 DATA READ COMMAND ( CODE = 0x71 )

설정된 MODE 1 레지스터 값을 읽어내는 COMMAND이다. 읽은 데이터는 8-bit 크기이며, 각 비트가 설정하는 내용은 다음과 같다.

7 bit	— 감속 시작 포인트 검출 방식
-------	-------------------

	0: 자동 검출 방식
	1: 나머지 검출 방식
6 bit ~ 4 bit	— 펄스 출력 방식 지정 (표 6.2.참조 )
3 bit ~ 0 bit	— 검출 대상 신호 설정 (표 6.3.참조 )

▶ \* 펄스출력방식

MODE1 Register 설정에 따른 CAMC-FS의 펄스 출력 방식은 아래 표와 같다.

표 6.1. 펄스출력 방식

비트			방향 신호(DIR)		펄스 신호 ( PULSE )		방식
D6	D5	D4	CW	CCW			
0	0	0	L	H	Active H Pulse		1 Pulse 방식
0	0	1	H	L			
0	1	0	L	H	Active L Pulse		
0	1	1	H	L			
1	0	0	CW Pulse	Active H	CCW Pulse	Active H	2 Pulse 방식
1	0	1		Active L		Active L	
1	1	0	CCW Pulse	Active H	CW Pulse	Active H	
1	1	1		Active L		Active L	

▶ \* 검출 대상 신호 설정

MODE1 Register에서 설정되는 신호 검출 드라이브 시 검출대상 신호는 아래 표와 같다.

표 6.2. 검출 대상 신호 설정

D3	D2	D1	D0	검출 신호 및 검출 Edge
0	0	0	0	+ELM 신호의 Negative Edge 검출
0	0	0	1	-ELM 신호의 Negative Edge 검출
0	0	1	0	+SLM 신호의 Negative Edge 검출
0	0	1	1	-SLM 신호의 Negative Edge 검출
0	1	0	0	IN0 신호의 하강 Edge 검출
0	1	0	1	IN1 신호의 하강 Edge 검출
0	1	1	0	IN2 신호의 하강 Edge 검출
0	1	1	1	IN3 신호의 하강 Edge 검출
1	0	0	0	+ELM 신호의 Positive Edge 검출
1	0	0	1	-ELM 신호의 Positive Edge 검출
1	0	1	0	+SLM 신호의 Positive Edge 검출
1	0	1	1	-SLM 신호의 Positive Edge 검출
1	1	0	0	IN0 신호의 상승 Edge 검출

D3	D2	D1	D0	검출 신호 및 검출 Edge
1	1	0	1	IN1 신호의 상승 Edge 검출
1	1	1	0	IN2 신호의 상승 Edge 검출
1	1	1	1	IN3 신호의 상승 Edge 검출

주) 검출신호중 +ELM/SLM 신호의 edge는 active level을 기준으로 사용된다. 즉 +ELM신호의 active level이 0이고 검출 신호가 '+ELM 신호의 Positive Edge 검출'이면 신호 검색 드라이브는 외부 신호 PELM이 '1'에서 '0'로 천이 할 때를 검색하게 된다.

▶ MODE 1 DATA WRITE COMMAND ( CODE = 0xF1 )

MODE1 설정 데이터 크기는 8-bit이며 이 명령으로 지정펄스 수 드라이브에서 감속 시작 포인트 검출 방식, 펄스 출력 방식, 신호검출 드라이브 중 검출 대상 신호 설정을 할 수 있다. RESET 후 기본값은 0x00으로 설정되며, 각 동작 설정과 해당 비트의 관계는 다음과 같다.

7 bit	— 감속 시작 포인트 검출 방식
	0: 자동 검출 방식
	1: 나머지 검출 방식
6 bit ~ 4 bit	— 펄스 출력 방식 지정 ( 표 6.2. 참조 )
3 bit ~ 0 bit	— 검출 대상 신호 설정 ( 표 6.3. 참조 )

주) 검출신호중 +ELM/SLM 신호의 edge는 active level을 기준으로 사용된다. 즉 +ELM신호의 active level이 'LOW'이고 검출 신호 설정이 '+ELM 신호의 Positive Edge 검출'이면 신호 검색 드라이브는 외부 신호 PELM이 '1'에서 '0'로 천이 할 때를 완료된다.

▶ MODE 2 DATA READ COMMAND ( CODE = 0x72 )

설정된 MODE 2 레지스터 값을 읽어내는 COMMAND이다. 읽은 데이터는 11-bit 크기이며, 각 비트가 설정하는 내용은 MODE 2 DATA WRITE COMMAND ( CODE = 0xF2 )와 동일하다.

▶ MODE 2 DATA WRITE COMMAND ( CODE = 0xF2 )

설정된 MODE 2 레지스터 값을 설정하는 COMMAND이다. 설정 데이터는 11-bit 크기이며, RESET 후 초기값은 0x000으로 설정되며, 각 비트가 설정하는 내용은 다음과 같다.

10bit	Trigger 신호의 active level 설정
9bit	Interrupt 출력 active level 설정
8bit	MARK 신호 active level 설정
7~6bit	Encoder signal count mode(ECUP, ECDN signal) 00: 단상 신호(ECUP: up 입력, ECDN: down 입력) 01: 90도 difference phase, 1times multiplied 10: 90도 difference phase, 2times multiplied 11: 90도 difference phase, 4times multiplied
5bit	INP 신호 ACTIVE LEVEL 설정
4bit	ALM 신호 ACTIVE LEVEL 설정
3bit	-SLM 신호 ACTIVE LEVEL 설정
2bit	+SLM 신호 ACTIVE LEVEL 설정
1bit	-ELM 신호 ACTIVE LEVEL 설정
0bit	+ELM 신호 ACTIVE LEVEL 설정

0일 경우 Active Level High 신호 설정, 1일 경우 Active Level Low 신호 설정.

▶ 외부 위치 카운터 입력 사양 ( Mode2 레지스터 상위 비트에서 설정 )

7 bit	6 bit	입력 사양	ECUP 신호	ECDN 신호
0	0	UP/DOWN 신호 입력	UP 입력	DOWN 입력
0	1	2 상 신호 1 체배 입력	φ A 입력	φ B 입력
1	0	2 상 신호 2 체배 입력	φ A 입력	φ B 입력
1	1	2 상 신호 4 체배 입력	φ A 입력	φ B 입력

▶ UNIVERSAL SIGNAL READ COMMAND ( CODE = 0x73 )

현재 설정되어 있는 범용 입출력 신호의 Level을 읽어내는 COMMAND이다. 전체 11-bit 크기의 데이터이며, 4~7 bit는 범용 입력 Level을 나타내며, 나머지는 범용 출력 Level을 나타낸다. 각 비트가 설정하는 내용은 다음과 같다.

8~10bit	Universal output bit operation 0xx: 연산데이터 값을 UOUT으로 출력 100: NOT(현재 UOUT의 반전) 101: AND (연산데이터 값과 현재 UOUT의 and 연산 출력) 110: OR(연산데이터 값과 현재 UOUT의 or 연산 출력) 111: XOR(연산데이터 값과 현재 UOUT의 difference check)
4~7bit	Universal input
0~3bit	Universal output

▶ UNIVERSAL SIGNAL WRITE COMMAND ( CODE = 0xF3 )

범용 입출력 신호의 Level을 설정하는 COMMAND이다. 전체 11-bit 크기의 데이터이며 RESET 후 초기값은 0x000으로 설정되며 4~7 bit는 범용 입력 Level을 나타내며, 나머지는 범용 출력 Level을 나타낸다. 4~7 bit 데이터는 외부 입력 신호에 의존하므로 이 명령으로 쓴 값은 다시 외부 신호에 의한 값으로 변경된다.

8~10bit	— Universal output bit operation 0xx: 연산데이터 값을 UOUT으로 출력 100: NOT(현재 UOUT의 반전) 101: AND (연산데이터 값과 현재 UOUT의 and 연산 출력) 110: OR(연산데이터 값과 현재 UOUT의 or 연산 출력) 111: XOR(연산데이터 값과 현재 UOUT의 difference check)
4~7bit	— Don' t care
0~3bit	— Universal output

이 COMMAND에서 상위 3bit로 연산 모드를 설정하며 byte write인 경우는 하위 1byte(4bit)를 universal output register에 write 한다.

연산기능의 사용 예는 다음과 같다.

NOT 연산인 경우 저장된 universal output register의 값에서 '1'로 설정한 bit는 NOT 연산을 '0'으로 설정한 bit는 그대로 둔다. 예를 들어 0x6를 저장된 universal output register에 NOT 연산 모드로 0x2를 쓰게 되면 0x9가 저장된다.

AND 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 AND 연산을 수행한다. 예를 들어, 0x7이 저장된 universal output register에 NOT 연산 모드로 0x2를 쓰게 되면 0x4가 저장된다.

OR 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 OR 연산을 수행한다. 예를 들어, 0x2가 저장된 universal output register에 OR 연산 모드로 0x5를 쓰게 되면 0x7이 저장된다.

XOR 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 XOR 연산을 수행한다. 예를 들어, 0x02가 저장된 universal output register에 XOR 연산 모드로 0x7를 쓰게 되면 0x5가 저장된다.

▶ END STATUS READ COMMAND ( CODE = 0x74 )

END STATUS 레지스터는 드라이브 종료 시 종료 원인을 READ하기 위한 14-bit 데이터 레지스터이다. 이 명령을 실행하여 END STATUS 레지스터를 읽음으로써 종료 원인을 파악할 수 있다. 각 비트의 설정은 종료 후 (BUSY = Low) 다음 드라이브 시작 전까지 유지된다.

14bit	— Limit(PELM, NELM, PSLM, NSLM, Software limit)에 의한 종료
13bit	— Limit 완전정지에 의한 종료
12bit	— Sensor positioning 드라이브 종료
11bit	— Preset pulse 드라이브 종료
10bit	— 신호검출(Signal search-1/2) 드라이브 종료
9bit	— 원점검출(Original search) 드라이브 종료
8bit	— 탈조에 의한 드라이브 종료
7bit	— 데이터 설정 에러에 의한 드라이브 종료
6bit	— Alarm 신호에 의한 드라이브 종료
5bit	— 급정지 명령(0xAC)에 의한 드라이브 종료

4bit	—	감속정지 명령(0xAB)에 의한 드라이브 종료
3bit	—	ESTOP 신호에 의한 드라이브 종료
2bit	—	SSTOP 신호에 의한 드라이브 종료
1bit	—	Emergency limit(PELM, NELM, Software emergency limit)에 의한 드라이브 종료
0bit	—	Slowdown limit(PSLM, NSLM, Software slowdown limit)에 의한 드라이브 종료

▶ MECHANICAL SIGNAL READ COMMAND ( CODE = 0x75 )

현재 입력되는 외부 위치 카운터 입력 신호, 모터 드라이버의 상태 입력 신호, Limit 입력 신호의 Level을 READ하는 COMMAND이다. 13-bit 데이터이며 각 비트의 해당 내용은 다음 표와 같다.

12bit	—	ESTOP 신호 입력 level
11bit	—	SSTOP 신호 입력 level
10bit	—	MARK# 신호 입력 level
9bit	—	EXPP 신호 입력 level
8bit	—	EXMP 신호 입력 level
7bit	—	엔코더 UP 신호 입력 Level(A 상 신호)
6bit	—	엔코더 DOWN 신호 입력 Level(B 상 신호)
5bit	—	INPOSITION 신호 입력 Active 상태
4bit	—	ALARM 신호 입력 Active 상태
3bit	—	-Limit 감속정지 신호 입력 Active 상태
2bit	—	+Limit 감속정지 신호 입력 Active 상태
1bit	—	-Limit 급정지 신호 입력 Active 상태
0bit	—	+Limit 급정지 신호 입력 Active 상태

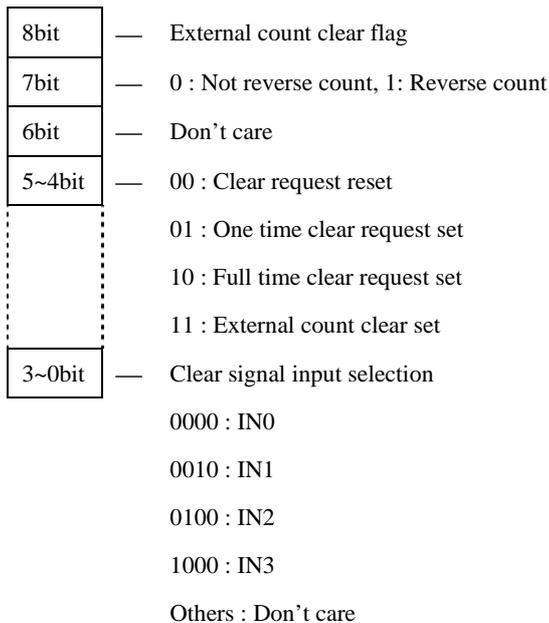
▶ DRIVE STATUS DATA READ COMMAND ( CODE = 0x76 )

현재 내부 위치 카운터와 외부 위치 카운터와의 비교 결과 및 가감속 상태를 알 수 있는 COMMAND이다. 9-bit 값으로 되어 있으며 각각 비트의 설정값은 아래의 표와 같다.

8bit	—	드라이브 방향 신호 (0: CW, 1: CCW)
7bit	—	ECG(외부 위치 카운터 비교값<외부 위치 카운터값)
6bit	—	ECL(외부 위치 카운터 비교값>외부 위치 카운터값)
5bit	—	ICG(내부 위치 카운터 비교값<내부 위치 카운터값)
4bit	—	ICL(내부 위치 카운터 비교값>내부 위치 카운터값)
3bit	—	UP(가속 중)
2bit	—	CONST(등속 중)
1bit	—	DOWN(감속 중)
0bit	—	BUSY(드라이브 구동 중)

▶ EXTERNAL COUNTER 설정 DATA READ COMMAND ( CODE = 0x77 )

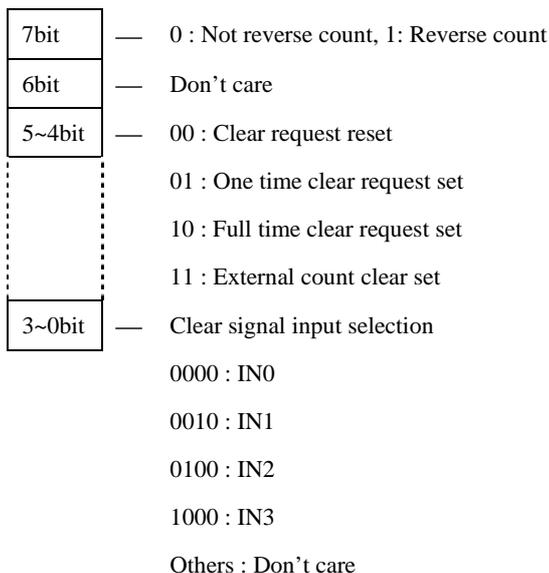
외부 위치 카운터의 설정 사항을 READ하는 COMMAND이다. 8-bit 값으로 되어 있으며 각 비트의 설정 사항은 다음과 같다. 특히 8bit은 read 전용 비트로 외부 위치 카운터가 설정한 입력에 의해 clear가 일어나면 set되고, 0xF7 command 를 다시 실행 하면 reset 된다.



▶ EXTERNAL COUNTER 설정 DATA WRITE COMMAND ( CODE = 0xF7 )

외부 위치 카운터의 설정 사항을 WRITE하는 COMMAND이다.

RESET 후 초기값은 0x00이며, 각 비트의 설정 사항은 다음 표와 같다.



7 bit가 1로 설정된 경우는 외부 위치 카운터가 역방향으로 카운트하게 된다.

5~4 bit에서 설정하는 항목은 One Time Clear Request Set으로 설정된 경우는 CLEAR 신호 입력이 있으면 한번 외부 카운터가 Clear되고 Clear Request Reset 상태로 변경된다. Full Time Clear Request Set으로 설정한 경우는 외부 Clear 입력 신호가 High가 입력될 때 마다 외부 카운터는 Clear된다. External Counter Clear Set 설정은 외부 카운터값을 0으로 계속 유지하게 하는 설정이다.

하위 4-bit에서는 표에서와 같이 Clear 신호 입력을 설정하는 항목이다.

▶ REGISTER CLEAR(INITIALIZATION) COMMAND ( CODE = 0xF8 )

내부의 모든 레지스터 값을 초기화할 수 있는 COMMAND이다. 즉 Software reset이면 외부 핀 RESET에 의한 초기화와 동일한 동작수행 한다. COMMAND 실행 후  $32 * T_{clk}$  시간 후에 내부 레지스터 값이 초기화 된다.

\*  $T_{clk}$  : 입력 주파수

▶ Interrupt FLAG READ COMMAND (CODE =0x79)

인터럽트 플래그를 READ하는 COMMAND이다. 32BIT 레지스터이며 각 BIT의 내용은 다음과 같으며 READ시에는 인터럽트 플래그가 클리어 된다.

INTERRUPT FLAG READ COMMAND

(0 : Interrupt inactivated, 1 : Interrupt activated)

31bit	—	Selectable Interrupt source3(“FE”의 31~24bit)
30bit	—	Selectable Interrupt source2(“FE”의 23~16bit)
29bit	—	Selectable Interrupt source1(“FE”의 15~8bit)
28bit	—	Selectable Interrupt source0(“FE”의 7~0bit)
27bit	—	Emergency limit 신호 입력시
26bit	—	Slow down limit 신호 입력시
25bit	—	캡션 queue 명령어 수행시(30 bit ‘1’ 일때)
24bit	—	스크립트 queue 명령어 수행시(30 bit ‘1’ 일때)
23bit	—	1’st counter 값이 M_DATA와 같을 때
22bit	—	현재속도 데이터 = 현재 속도 비교데이터
21bit	—	현재속도 데이터 = RCP12
20bit	—	현재속도 데이터 = RCP23
19bit	—	MODE1 register에 설정된 신호 검출시
18bit	—	가속
17bit	—	등속
16bit	—	감속

15bit	—	Interrupt command에 의해 생성
14bit	—	캡션-3에 저장된 command가 실행될 때
13bit	—	캡션-2에 저장된 command가 실행될 때
12bit	—	캡션-1에 저장된 command가 실행될 때
11bit	—	스크립트-3에 저장된 command가 실행될 때
10bit	—	스크립트-2에 저장된 command가 실행될 때
9bit	—	스크립트-1에 저장된 command가 실행될 때
8bit	—	캡션 동작 설정 queue가 full될 때
7bit	—	스크립트 data queue가 empty될 때
6bit	—	External counter < External comparator가 될 때
5bit	—	External counter = External comparator가 될 때
4bit	—	External counter > External comparator가 될 때
3bit	—	Internal counter < Internal comparator가 될 때
2bit	—	Internal counter = Internal comparator가 될 때
1bit	—	Internal counter > Internal comparator가 될 때
0bit	—	드라이브 종료

주) CAMC-FS의 interrupt flag는 32개중 최초 발생한 interrupt에 의해 갱신되고, 이후 flag read command가 수행되기 전까지는 유지된다. 즉 드라이브종료에 의해 interrupt flag가 0x00000001이 되었다면 이후 다른 interrupt 발생 조건이 만족하더라도 interrupt flag read command 가 수행되기 전까지는 0x00000001로 유지된다.

▶ Interrupt 발생 WRITE COMMAND (CODE =0xF9)

Interrupt 강제 발생 기능을 설정하는 COMMAND이다.

▶ Interrupt MASK READ COMMAND (CODE =0x7A)

인터럽트 마스크를 READ하는 COMMAND이다. RESET 후 기본값은 1(0x0000001)이다. 32BIT 레지스터이며 각 BIT의 내용은 Interrupt MASK WRITE COMMAND (CODE =0xFA)와 같다.

▶ Interrupt MASK WRITE COMMAND (CODE =0xFA)

인터럽트 마스크를 설정하는 COMMAND이다. 드라이브 종료에 대해서 인터럽트를 마스크 하지 않는다. 32 BIT 레지스터이며 각 BIT의 내용은 다음과 같다.

Interrupt MASK(0 : Interrupt Disable, 1 : Enable)

31bit	—	Selectable Interrupt source3(“FE”의 31~24bit)
30bit	—	Selectable Interrupt source2(“FE”의 23~16bit)
29bit	—	Selectable Interrupt source1(“FE”의 15~8bit)

28bit	—	Selectable Interrupt source0(“FE”의 7~0bit)
27bit	—	Emergency limit 신호 입력시
26bit	—	Slow down limit 신호 입력시
25bit	—	캡션 queue 명령어 수행시(30 bit ‘1’ 일때)
24bit	—	스크립트 queue 명령어 수행시(30 bit ‘1’ 일때)
23bit	—	1’st counter 값이 M_DATA와 같을 때
22bit	—	현재속도 데이터 = 현재 속도 비교데이터
21bit	—	현재속도 데이터 = RCP12
20bit	—	현재속도 데이터 = RCP23
19bit	—	MODE1 register에 설정된 신호 검출시
18bit	—	가속
17bit	—	등속
16bit	—	감속
15bit	—	Don’t care
14bit	—	캡션-3에 저장된 command가 실행될 때
13bit	—	캡션-2에 저장된 command가 실행될 때
12bit	—	캡션-1에 저장된 command가 실행될 때
11bit	—	스크립트-3에 저장된 command가 실행될 때
10bit	—	스크립트-2에 저장된 command가 실행될 때
9bit	—	스크립트-1에 저장된 command가 실행될 때
8bit	—	캡션 동작 설정 queue가 full될 때
7bit	—	스크립트 data queue가 empty될 때
6bit	—	External counter < External comparator가 될 때
5bit	—	External counter = External comparator가 될 때
4bit	—	External counter > External comparator가 될 때
3bit	—	Internal counter < Internal comparator가 될 때
2bit	—	Internal counter = Internal comparator가 될 때
1bit	—	Internal counter > Internal comparator가 될 때
0bit	—	드라이브 종료

▶ EMODE1 DATA READ COMMAND (CODE =0x7B)

확장 Universal I/O, Monitor신호의 사용 여부 및 입출력을 읽어내는 COMMAND이다. 읽은 DATA는 DATA4 port에 2<sup>7</sup>~2<sup>0</sup> bit 순으로 8-bit 크기로 저장되어 있다.

7bit	Extension mode enable (0: Disable, 1: Enable)
6~5bit	0: md_out[12:8]=>md_out[12:8]
	1: md_out[12:8]=>EU04~1,EUI00
6~5bit	Don't care
4bit	EU04 enable/disable(0:disable,1:enable)
3bit	EU03 enable/disable(0:disable,1:enable)
2bit	EU02 enable/disable(0:disable,1:enable)
1bit	EU01 enable/disable(0:disable,1:enable)
0bit	EUI00 direction (0:input, 1: output)

▶ EMODE1 DATA WRITE COMMAND (CODE =0xFB)

확장 Universal I/O, Monitor신호의 사용 여부 및 입출력 데이터 설정하는 COMMAND이다. RESET 후 초기값은 0x00을 가지며, DATA4 port에 설정된 8-bit 크기의 데이터로 설정할 수 있다.

7bit	Extension mode enable (0: Disable, 1: Enable)
6~5bit	0: md_out[12:8]=>md_out[12:8]
	1: md_out[12:8]=>EU04~1,EUI00
6~5bit	Don't care
4bit	EU04 enable/disable(0:disable,1:enable)
3bit	EU03 enable/disable(0:disable,1:enable)
2bit	EU02 enable/disable(0:disable,1:enable)
1bit	EU01 enable/disable(0:disable,1:enable)
0bit	EUI00 direction (0:input, 1: output)

▶ EXTENSION UNIVERSAL OUT READ COMMAND (CODE =0x7C)

현재 설정되어 있는 확장 출력 신호의 Level을 읽어내는 COMMAND이다. 전체 8-bit 크기의 데이터이며, 0~4 bit는 확장 출력 신호의 Level을 나타내며, 5~7 bit는 출력신호의 연산 기능을 나타낸다.

7~5bit	Extension Universal output bit operation
	0xx: 연산데이터 값을 UOUT으로 출력
	100: NOT(현재 UOUT의 반전)
	101: AND(연산데이터 값과 현재 UOUT의 and 연산 출력)
	110: OR(연산데이터 값과 현재 UOUT의 or 연산 출력)
111: XOR(연산데이터 값과 현재 UOUT의 difference check)	
4~0bit	Extension Universal output 연산 데이터(input의 경우 무시)

▶ EXTENSION UNIVERSAL OUT WRITE COMMAND (CODE =0xFC)

확장 출력 신호의 Level을 설정하는 COMMAND이다. EMODE1에서 direction을 input, 사용여부를 disable로 설정한 bit에 대해서는 의미가 없다. RESET 후 초기값은 0x000으로 설정되며, 전체 8-bit 크기의 데이터이며 0~4 bit는 확장 출력신호 Level을 나타내며, 5~7 bit는 연산 모드를 설정한다.

연산기능의 사용 예는 다음과 같다.

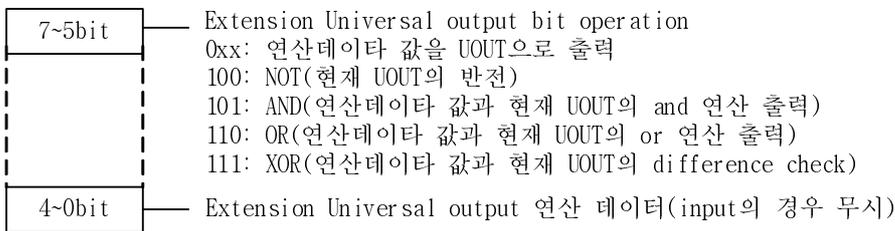
다음은 모든 EMODE1을 0x9F로 설정하였을 때의 예이다.

NOT 연산인 경우 현재 저장된 universal output register의 값이 반전된다. 예를 들어 0x06를 저장된 universal output register에 NOT 연산 모드를 쓰게 되면 0x19가 저장된다.

AND 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 AND 연산을 수행한다. 예를 들어, 0x17이 저장된 universal output register에 AND 연산 모드로 0x2를 쓰게 되면 0x02가 저장된다.

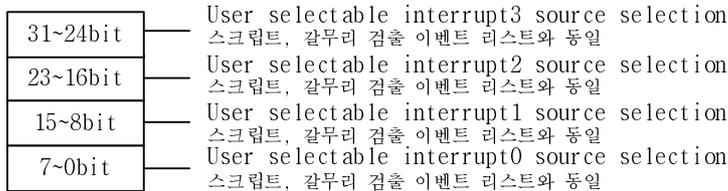
OR 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 OR 연산을 수행한다. 예를 들어, 0x02가 저장된 universal output register에 OR 연산 모드로 0x15를 쓰게 되면 0x17이 저장된다.

XOR 연산인 경우 저장된 universal output register와 입력하는 새로운 data와 bit별 XOR 연산을 수행한다. 예를 들어, 0x12가 저장된 universal output register에 XOR 연산 모드로 0x07를 쓰게 되면 0x15가 저장된다.



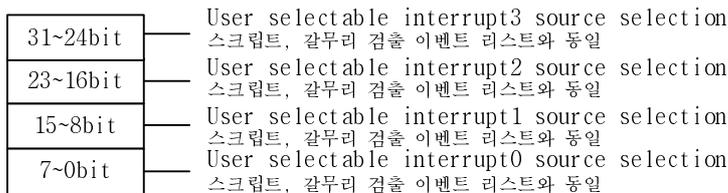
▶ USER INTERRUPT SOURCE SELECTION REGISTER READ COMMAND (CODE =0x7E)

Interrupt source를 사용자가 설정한 데이터를 읽어내는 COMMAND이다. 설정 가능한 Interrupt source는 스크립터/캡션의 검출이벤트 리스트와 동일하다.



▶ USER INTERRUPT SOURCE SELECTION REGISTER WRITE COMMAND (CODE =0xFE)

Interrupt source를 사용자가 설정하는 COMMAND이다. RESET 후 초기값은 0x00000000으로 설정되며, 설정 가능한 Interrupt source는 스크립터/캡션의 검출이벤트 리스트와 동일하다.

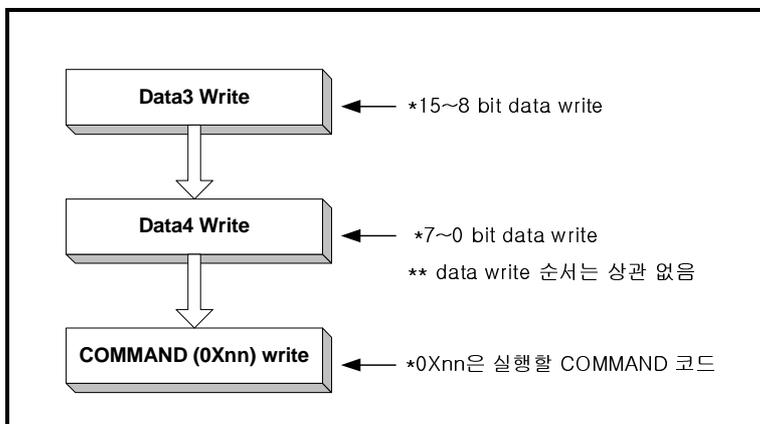


### 6.3. COMMAND 실행 방법

CAMC-FS 칩에 대한 COMMAND의 실행 예를 보여준다. COMMAND에는 DATA기입 COMMAND, DATA 읽기 COMMAND, DATA기입/읽기를 동반하는 COMMAND가 있고, DATA 기입/읽기를 동반하지않는 COMMAND에 있어서는 COMAND WRITE PORT에 대해 COMMAND CODE 기입만 가능하다.

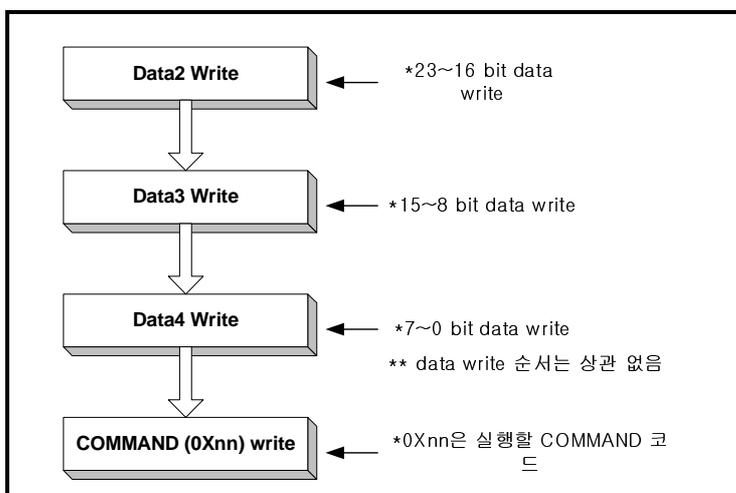
▶ 2-BYTE DATA 기입 COMMAND 실행 방법

다음에 표시하는 플로우 차트는 2 BYTE DATA 기입 COMMAND 실행 방법을 설명한다. 먼저 2-byte 데이터를 write한 후(이 데이터는 명령 실행에 사용됨) 실행한 명령을 write하여 2-byte 데이터 기입 명령을 실행한다.



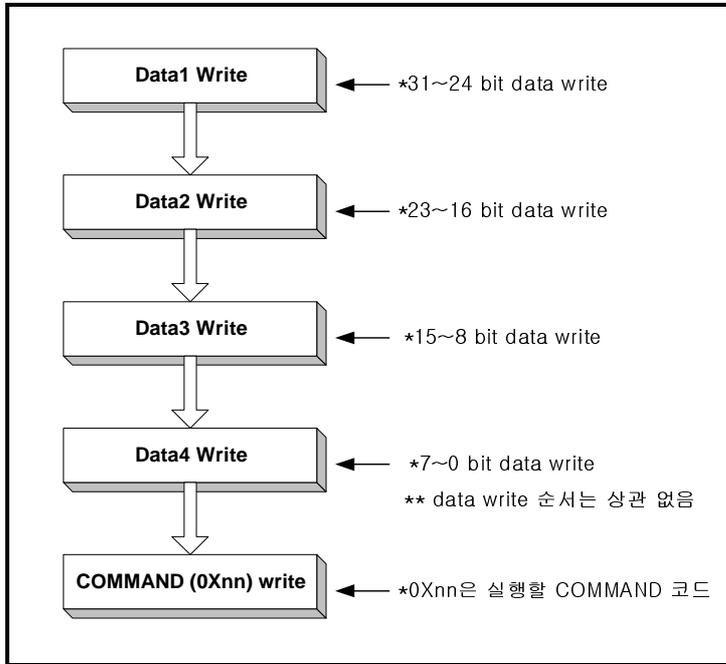
▶ 3-BYTE DATA 기입 COMMAND 실행 방법

다음에 표시하는 플로우 차트는 3 BYTE DATA 기입 COMMAND 실행 방법을 설명한다. 먼저 3-byte 데이터를 write한 후(이 데이터는 명령 실행에 사용됨) 실행한 명령을 write하여 3-byte 데이터 기입 명령을 실행한다.



▶ 4-BYTE DATA 기입 COMMAND 실행 방법

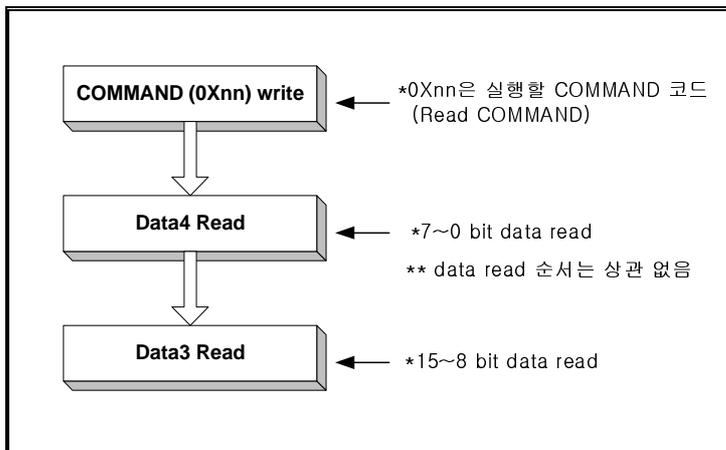
다음에 표시하는 플로우 차트는 4 BYTE DATA 기입 COMMAND 실행 방법을 설명한다. 먼저 4-byte 데이터를 write한 후(이 데이터는 명령 실행에 사용됨) 실행한 명령을 write하여 4-byte 데이터 기입 명령을 실행한다.



▶ 2-BYTE DATA 읽기 COMMAND 실행 방법

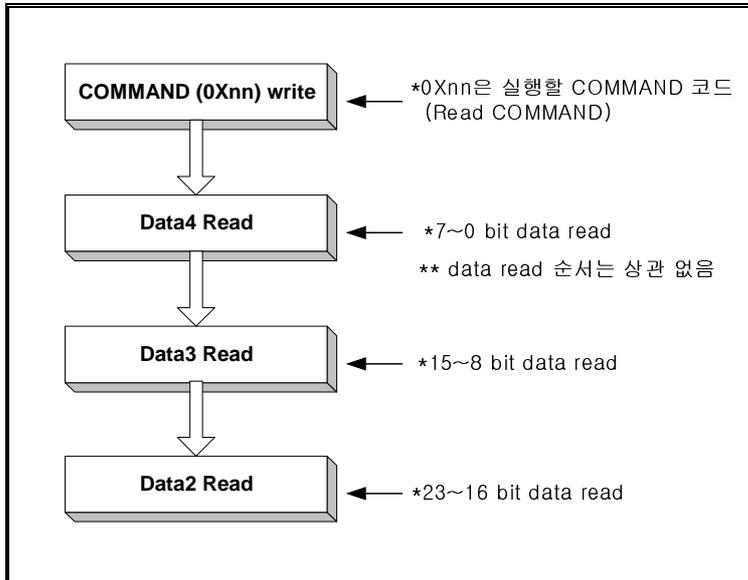
2-BYTE DATA 기입 COMAND 실행 방법을 표시한다.

Read command 실행 방법은 우선 Read할 데이터를 읽는 Read COMMAND를 실행한다. 그 후 Data port에 저장된 데이터를 8-bit씩 읽어낸다.



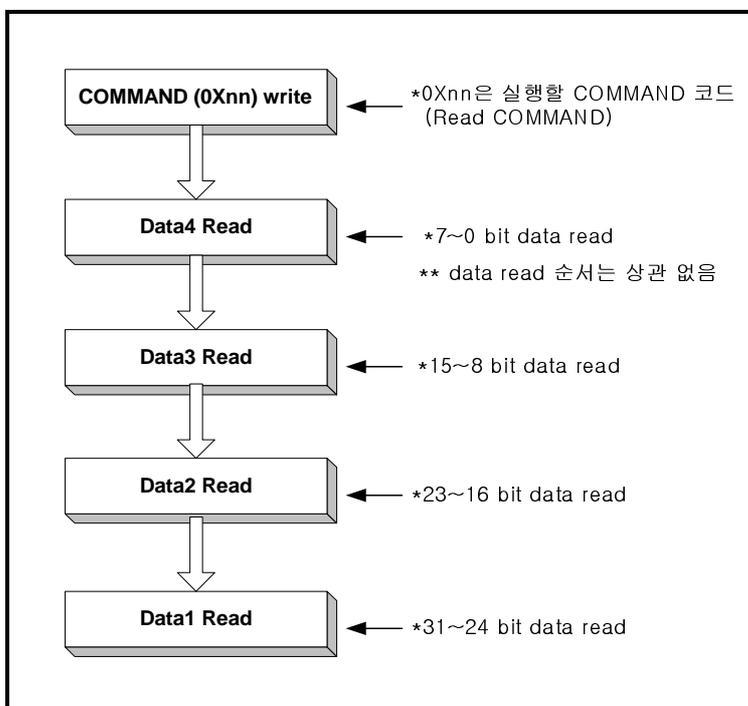
## ▶ 3-BYTE DATA 읽기 COMMAND 실행 방법

3-BYTE DATA 읽기 COMMAND 실행 방법을 표시한다.



## ▶ 4-BYTE DATA 읽기 COMMAND 실행 방법

4 BYTE DATA 읽기 COMMAND 실행 방법을 표시한다

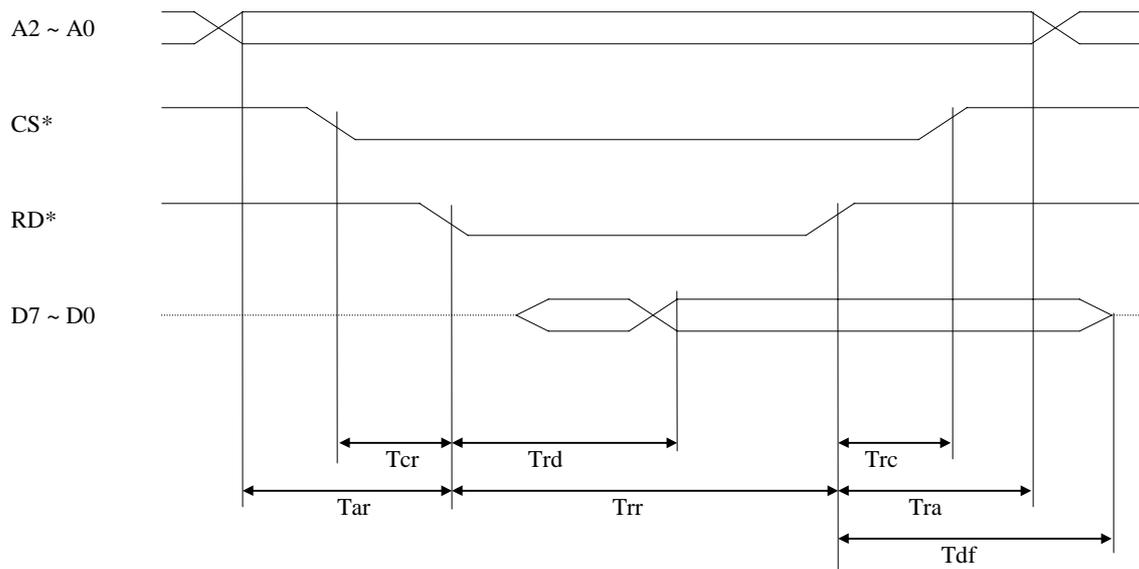


# 7. TIMING

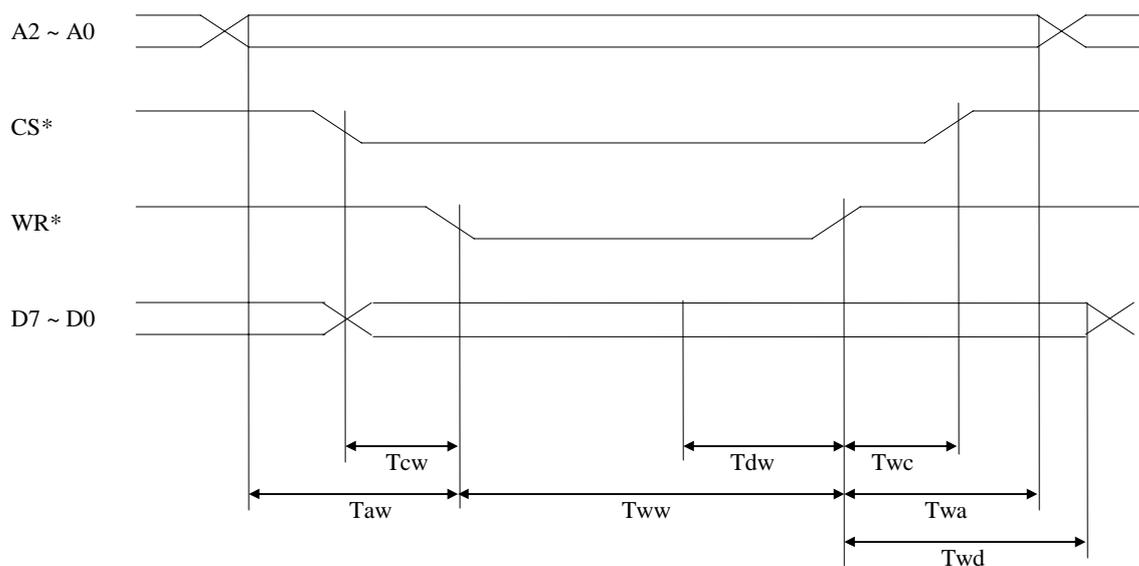
## 7.1. DATA BUS TIMING

여기에서는 DATA BUS에 의한 READ / WRITE TIMING을 나타내었다.

### ▶ READ TIMING



### ▶ WRITE TIMING



## ▶ DATA BUS TIMING 표

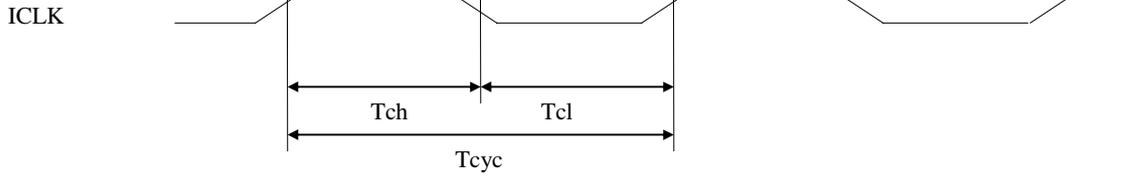
항 목	기 호	MIN	MAX
CS* 안정 시간 (R*에 대해 )	Tcr	0	
RD* → Data 출력 지연 시간	Trd		30
CS* 유지 시간 (R*에 대해 )	Trc	0	
Address 안정 시간 (R*에 대해 )	Tar	0	
RD* Pulse 폭	Trr	30	
Address 유지 시간 (R*에 대해 )	Tra	0	
RD* → Data Float 지연 시간	Tdf		30
CS* 안정 시간 (W*에 대해 )	Tcw	0	
Data 설정 시간 (W*에 대해 )	Tdw	20	
CS* 유지 시간 (W*에 대해 )	Twc	0	
Address 안정 시간 (W*에 대해 )	Taw	0	
WR* Pulse 폭	Tww	30	
Address 유지 시간 (W*에 대해 )	Twa	0	
Data 유지 시간 (W*에 대해 )	Twd	0	

단위 nSec

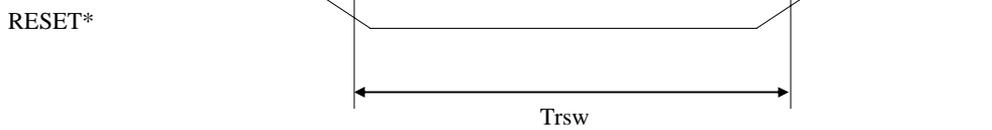
## 7.2. 입력 신호 TIMING

여기에서는 입력 신호에 있어서의 입력 폭 및 입력 주기를 나타내었다.

### ▶ ICLK입력 TIMING



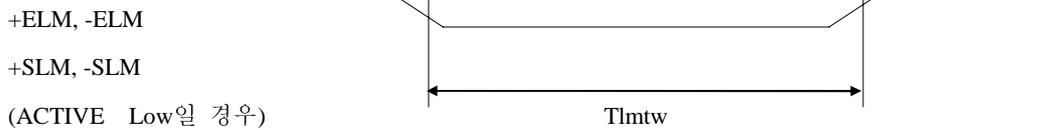
### ▶ RESET\* 입력 TIMING



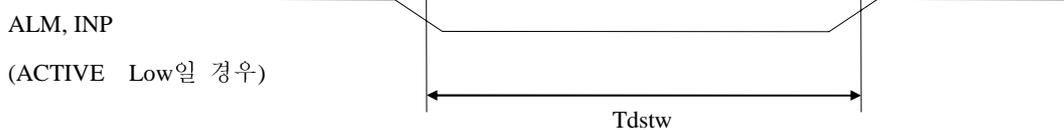
### ▶ ESTOP, SSTOP 입력 TIMING



### ▶ +ELM, -ELM, +SLM, -SLM 입력 TIMING



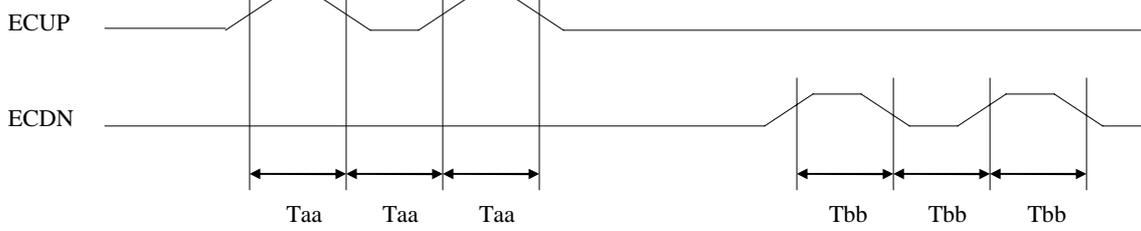
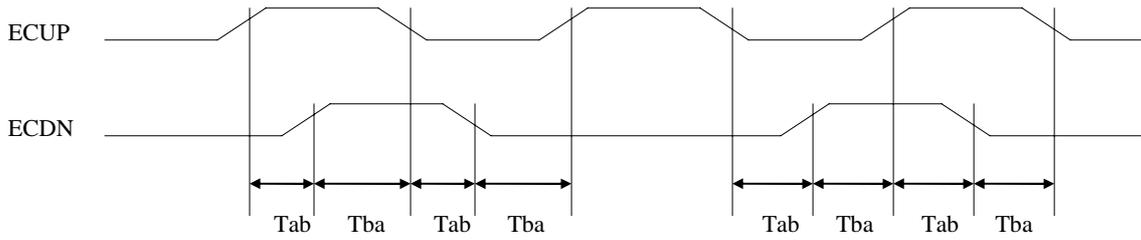
### ▶ ALM, INP 입력 TIMING



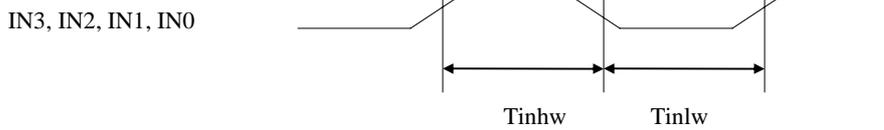
▶ SYNC 입력 TIMING



▶ ECUP, ECDN 입력 TIMING (2相 입력 시)



▶ IN3 ~ IN0 입력 TIMING (SIGNAL SEARCH - 1, - 2 DRIVE로 사용할 경우)



▶ 입력 신호 TIMING 표

항 목	기 호	MIN	MAX
ICLK 입력 High Level 폭	Tch	30	
ICLK 입력 Low Level 폭	Tcl	30	
ICLK 입력 주기	Tcyc	61	
Reset 입력 폭	Trsw	8* Tcyc	
ESTOP, SSTOP 입력 폭	Tstp	Tcyc * BW(*1)	
+ELM, -ELM, +SLM, -SLM 입력 폭	Tlmtw	Tcyc * BW(*1)	
ALM, INP 입력 폭	Tdstw	Tcyc * BW(*1)	
SYNC 입력 폭	Tsynw	Tcyc * 5	
ECUP --> ECDN HOLD 시간(*2)	Tab	Tcyc * 2 + 10	
ECDN --> ECUP HOLD 시간(*2)	Tba	Tcyc * 2 + 10	
ECUP --> ECUP HOLD 시간(*3)	Taa		Tcyc * 2 + 10
ECDN --> ECDN HOLD 시간 (*3)	Tbb		Tcyc * 2 + 10
IN3 ~ IN0 입력 H LEVEL 폭	Tinhw	30	
IN3 ~ IN0 입력 L LEVEL 폭	Tinlw	30	

단위 nSec

\* 1 : BW = 센서 신호 입력 폭 설정 데이터

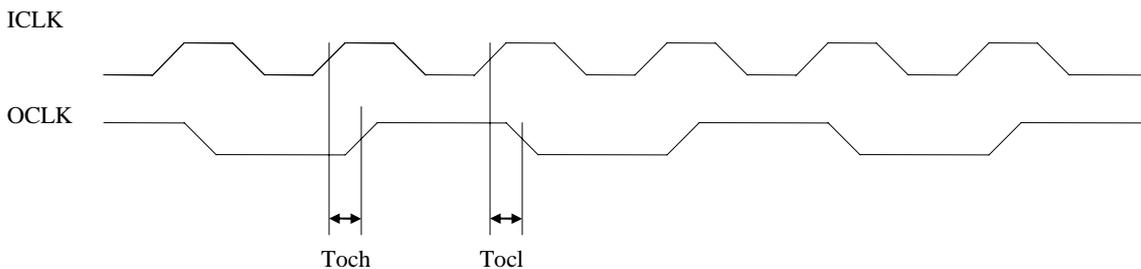
\* 2 : 2相신호 입력 시

\* 3 : 2相신호 입력 시. 2相신호에 CHATTERING이 발생할 경우의 규정에 대해서는 "2相신호의 CHATTERING"을 참고 할 것.

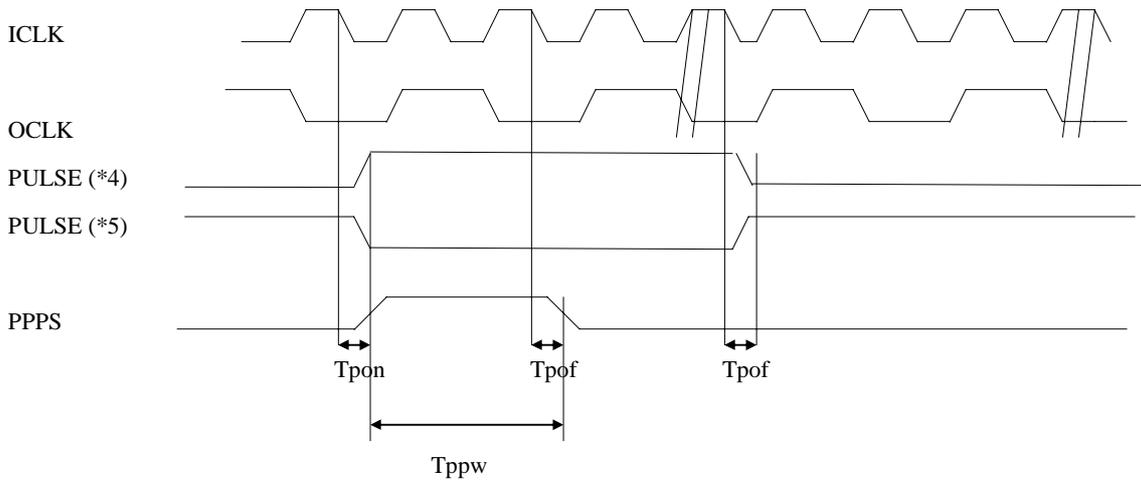
7.3. 출력 신호 TIMING 1

여기에서는 ICLK 입력에 동기하여 출력되는 신호에 있어서, ICLK에서부터의 출력 지연 시간을 나타내었다.

▶ OCLK 출력 TIMING



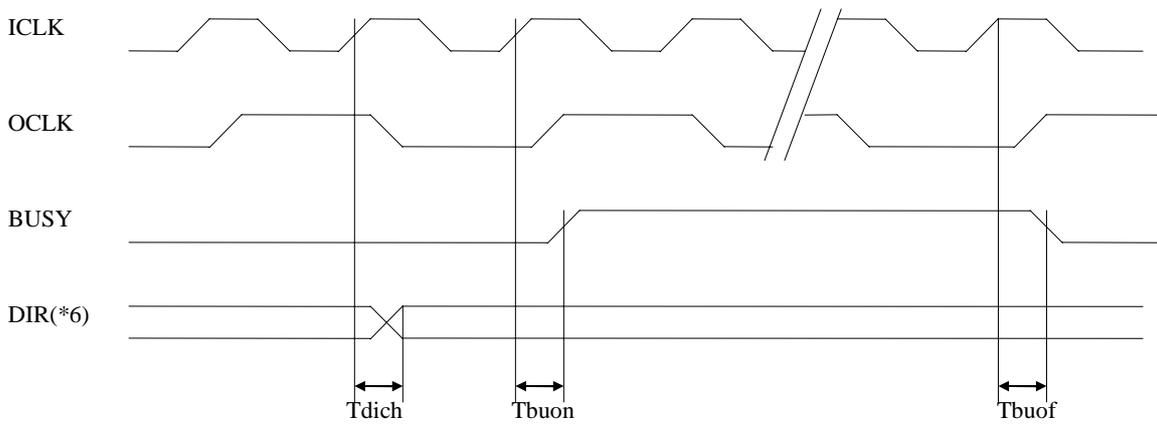
▶ PULSE, PPS 출력 TIMING



\*4. ACTIVE High인 경우

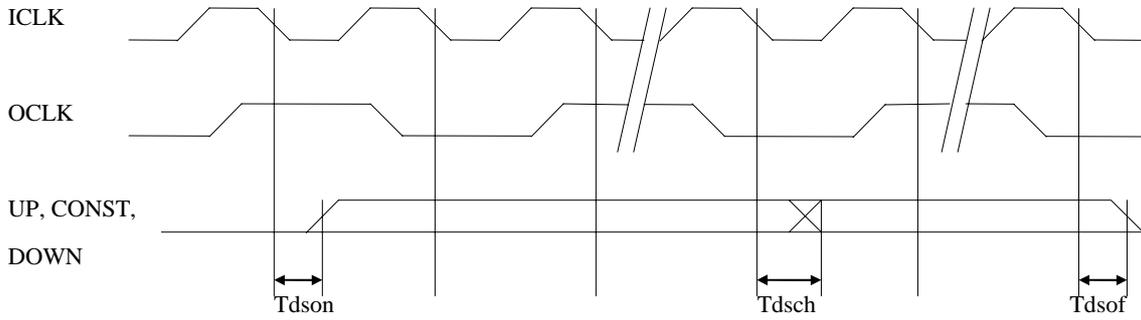
\*5. ACTIVE Low인 경우

▶ BUSY, DIR 출력 TIMING

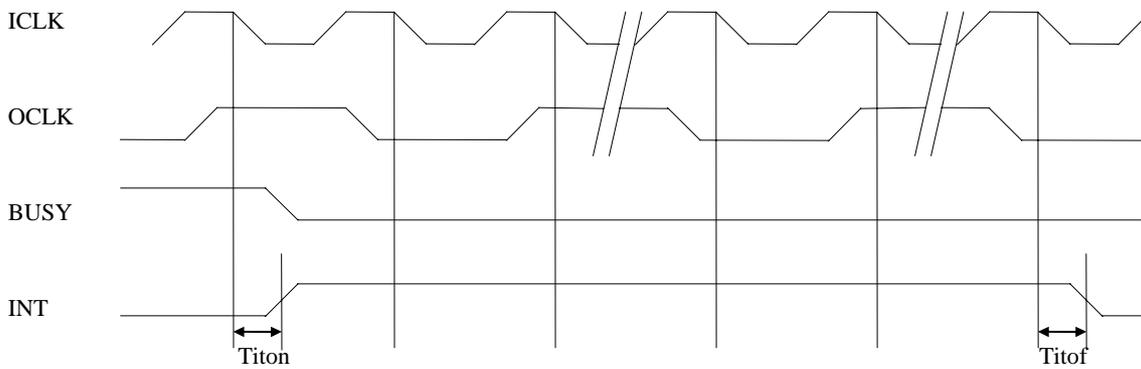


\*6. 2 PULSE 방식 지정시의 DIR 신호 TIMING에 대해서는 PULSE신호 TIMING을 참고할 것.

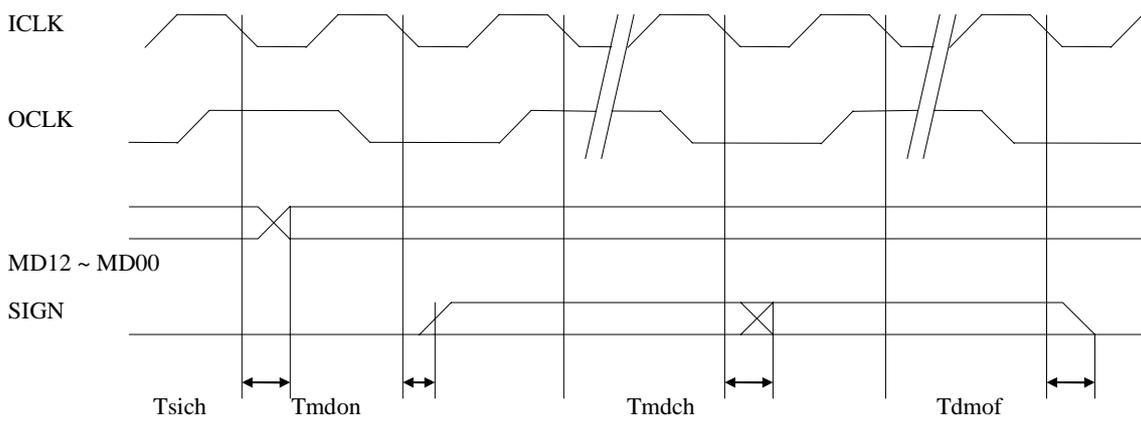
▶ UP, CONST, DOWN 출력 TIMING



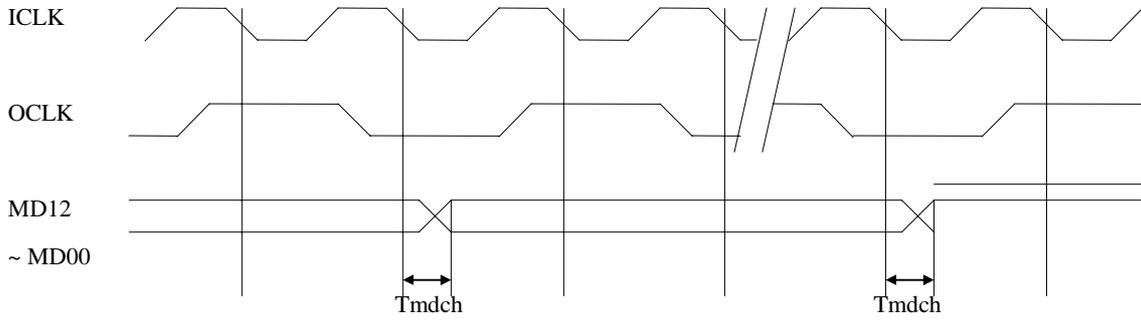
▶ INT 출력 TIMING



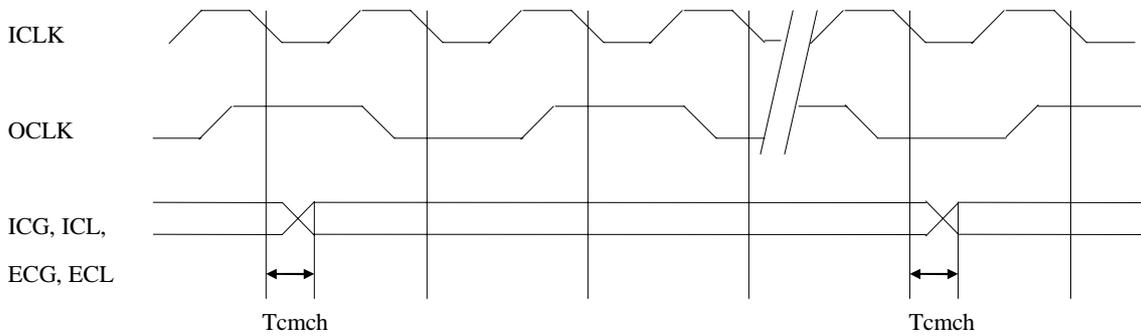
▶ SIGN, MD12 ~ MD00 출력 TIMING (MONI = H : SPEED DATA MONITOR 시)



▶ SIGN, MD12 ~ MD00 출력 TIMING (MONI = L : 편차량 DATA MONITOR 시)



▶ ICG, ICL, ECG, ECL 출력 TIMING



▶ ICLK 동기 출력 신호 TIMING 표

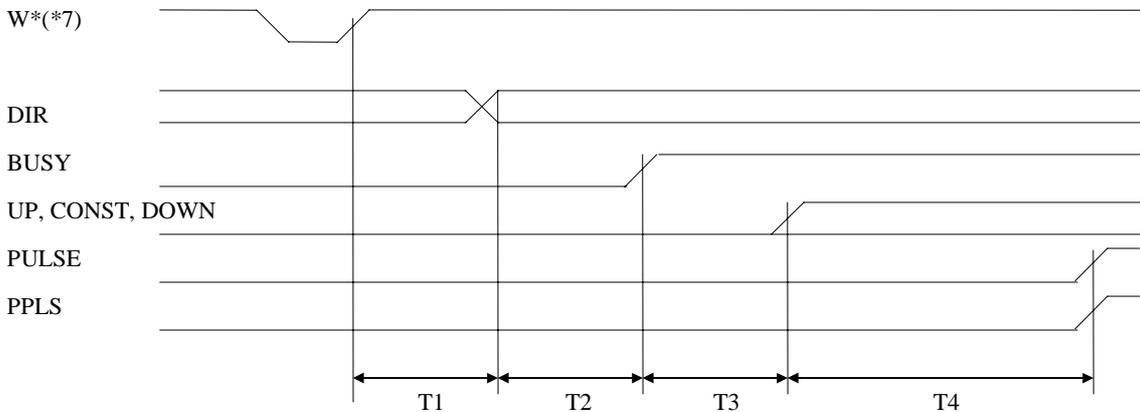
항 목	기 호	MIN	MAX
ICLK ~ OCLK H 출력 지연 시간	Toch	5	13
ICLK ~ OCLK L 출력 지연 시간	Tocl	5	13
ICLK ~ PULSE, PLS ON 출력 지연 시간	Tpon	6	26
ICLK ~ PULSE, PLS OFF 출력 지연 시간	Tpof	6	26
PPLS 출력 PULSE 폭	Tppw	Tcyc* 2-20	Tcyc*2+10
ICLK ~ DIR 전환 지연 시간	Tdich	15	32
ICLK ~ BUSY ON 출력 지연 시간	Tbuon	10	34
ICLK ~ BUSY OFF 출력 지연 시간	Tbuof	8	35
ICLK ~ UP, CONST, DOWN ON 출력 지연 시간	Tdson	9	20
ICLK ~ UP, CONST, DOWN OFF 전환 지연 시간	Tdsch	6	20
ICLK ~ UP, CONST, DOWN OFF 출력 지연 시간	Tdso	9	30
ICLK ~ INT ON 출력 지연 시간	Titon	10	30
ICLK ~ INT OFF 출력 지연 시간	Titof	7	30
ICLK ~ SIGN 전환 지연 시간	Tsich	15	33
ICLK ~ MD12 ~ MD00 ON 출력 지연 시간	Tmdon	10	30
ICLK ~ MD12 ~ MD00 전환 지연 시간	Tmdch	8	25
ICLK ~ MD12 ~ MD00 OFF 출력 지연 시간	Tmdof	10	30
ICLK ~ ICG, ICL, ECG, ECL 전환 지연 시간	Tcmch	5	20

단위 nSec

7.4. 출력 신호 TIMING 2

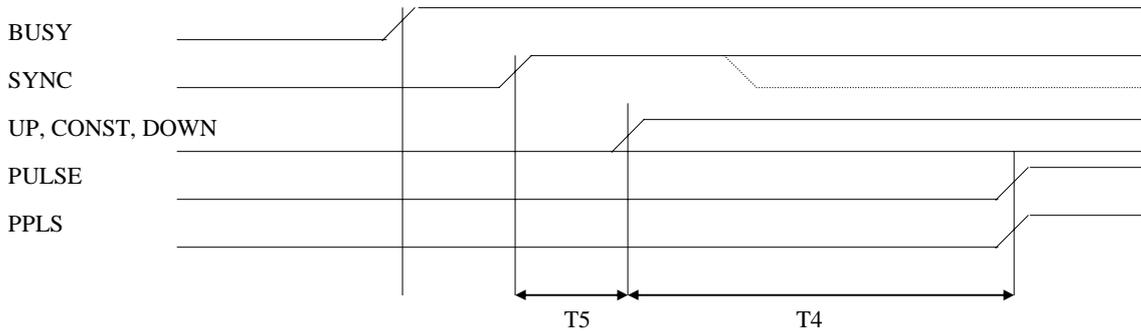
여기서는 실제 동작에 있어서의 입력 신호 및 출력 신호의 관계를 나타내었다.

▶ DRIVE 개시 TIMING (SYNC 입력에 의한 다축 동기 맞춤 하지 않을 시)

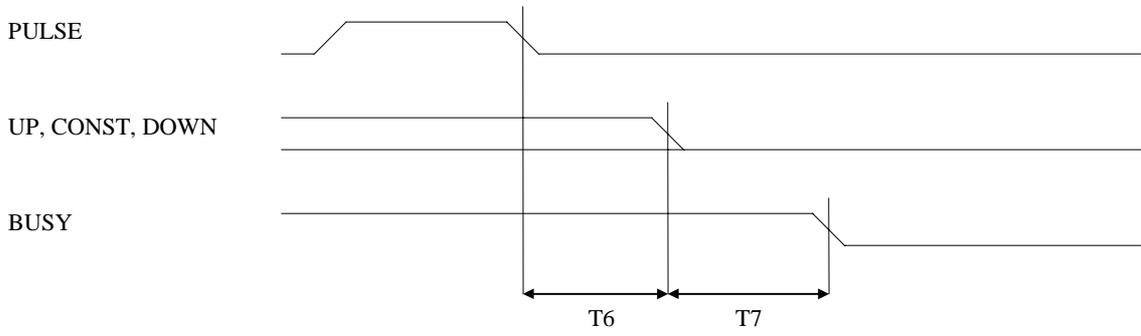


\*7. COMMAND WRITE PORT에 대한 기입을 표시 한다.

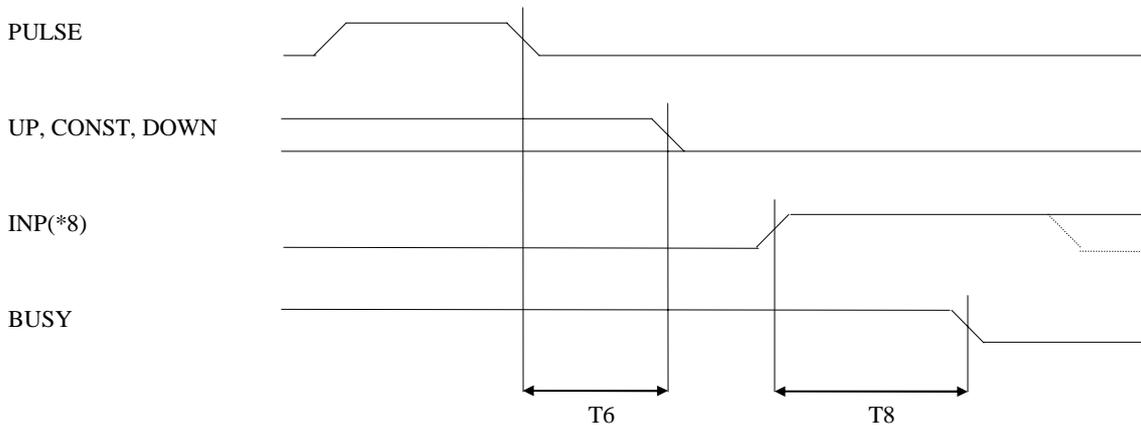
▶ SYNC 입력에 의한 다축 동기 맞춤시의 TIMING



▶ DRIVE 종료 TIMING (INP 신호 확인 하지 않을 경우)

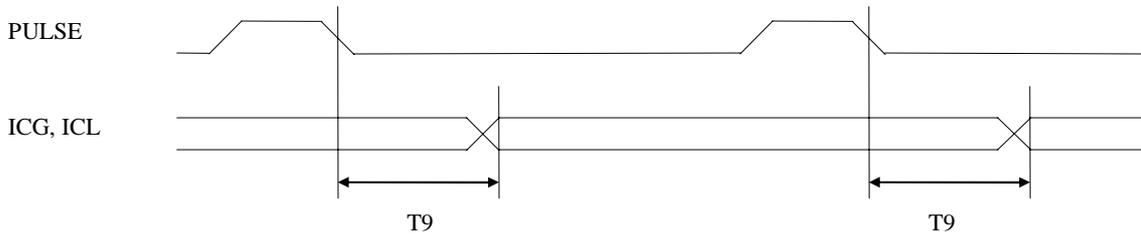


▶ DRIVE 종료 TIMING (INP 신호 확인 할 경우)

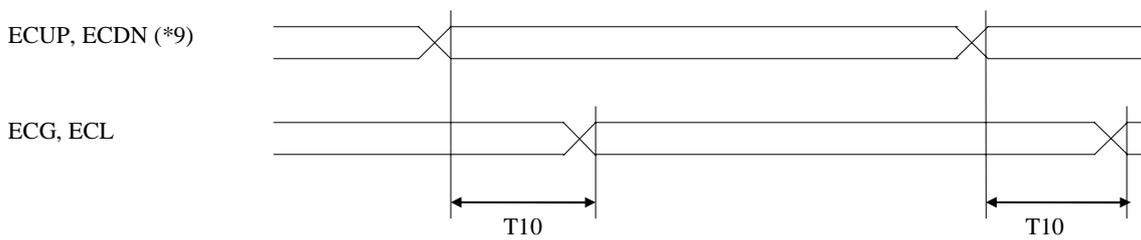


\*8. ACTIVE High일 경우

▶ ICG, ICL 출력 TIMING

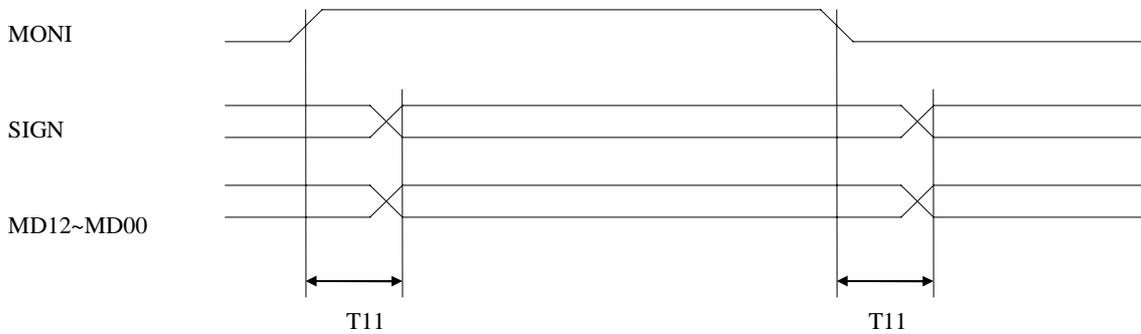


▶ ECG, ECL 출력 TIMING (2相 신호 입력 시)

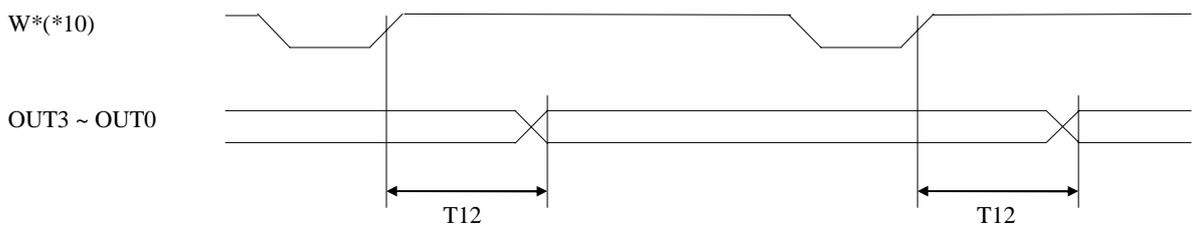


\*9. 2相 신호 입력 시를 나타냄.

▶ MONI 전환 TIMING

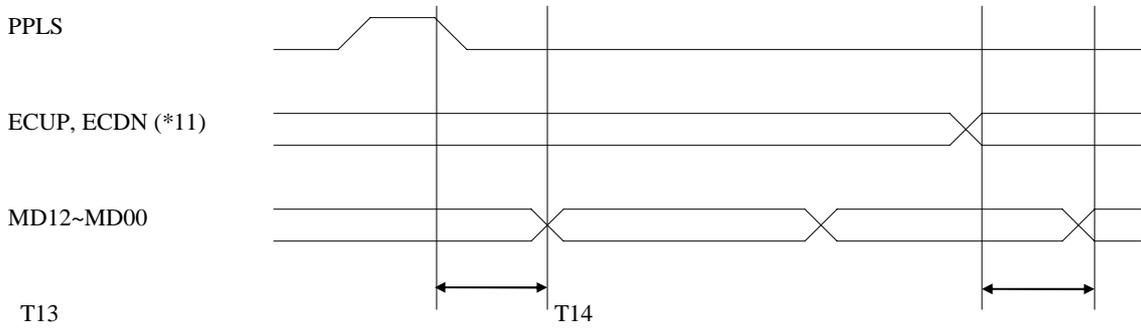


▶ OUT3 ~ OUT0 출력 TIMING



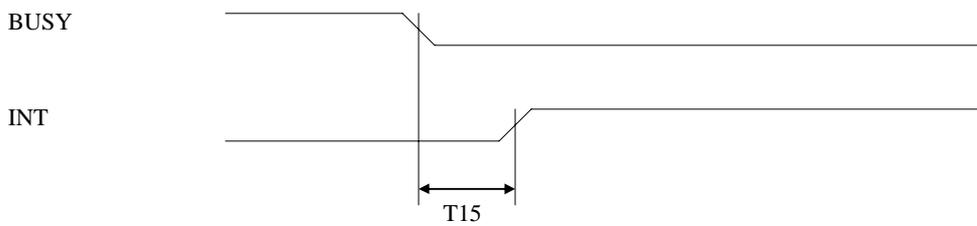
\*10. UNIVERSAL SIGNAL WRITE PORT에 대한 기입을 표시함.

▶ 편차량 DATA 반영 TIMING (MONI = L 시)

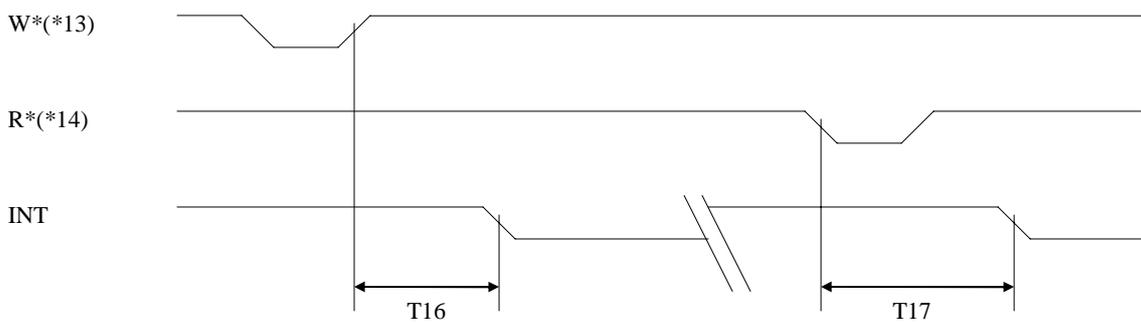


\*11. 2T相 신호 입력 시를 나타냄.

▶ INT 출력 ON TIMING



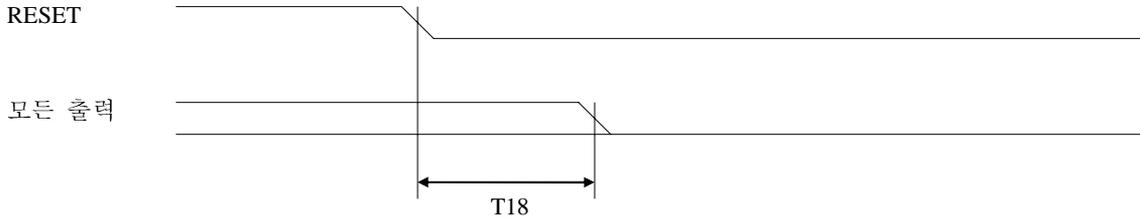
▶ INT 출력 OFF TIMING



\*13. COMMAND WRITE PORT에 대해서 새로운 DRIVE COMMAND를 기입함을 표시.

\*14. END STATUS READ PORT로부터 읽음을 표시.

▶ RESET TIMING



▶ 실제 동작 출력 신호의 TIMING 표

항 목	기 호	MIN	MAX
COMMAND WRITE PORT 기입 ~ DIR 출력 지연 시간	T1	Tcyc	Tcyc*8 + 70
DIR 출력 ~ BUSY ON 출력 지연 시간	T2	Tcyc - 30	Tcyc + 10
BUSY ON 출력 ~ UP, CONST, DOWN ON 출력 지연 시간	T3	Tcyc * 2 - 20	Tcyc*2 + 20
UP, CONST, DOWN ON출력 ~ PULSE, PPLS ON 출력지연 시간	T4	Tcyc * 4 - 20	
SYNC ON 입력 ~ UP, CONST, DOWN ON 출력 지연 시간	T5	Tcyc - 5	Tcyc*5 + 50
PULSE OFF 출력 ~ UP, CONST, DOWN OFF 출력 지연 시간	T6	Tcyc - 20	Tcyc + 30
UP, CONST, DOWN OFF 출력 ~ BUSY OFF 출력 지연 시간	T7	Tcyc - 5	Tcyc*5 + 50
INP ON 입력 ~ BUSY OFF 출력 지연 시간	T8	Tcyc - 10	Tcyc*5 + 45
PPLS OFF 출력 --> ICG, ICL 출력 지연 시간	T9	Tcyc - 25	Tcyc + 20
ECUP, ECDN 입력 --> ECG, ECL 출력 지연 시간 (*9)	T10	Tcyc*3 - 10	Tcyc*5 + 40
MONI 입력 전환 ~ SIGN, MD12 ~ MD00 출력 지연 시간	T11	7	40
UNIVERSAL SIGNAL WRITE PORT 기입 ~ OUT3 ~ OUT0 출력 지연 시간	T12	5	30
PPLS OFF 출력 ~ SIGN, MD12 ~ MD00 출력 지연 시간	T13	Tcyc * 3 - 25	Tcyc*5 + 40
ECUP, ECDN 입력 ~ SIGN, MD12 ~ MD00 출력 지연 시간 (*10)	T14	Tcyc * 5 - 10	Tcyc*9 + 55
BUSY OFF 출력 ~ INT ON 출력 지연 시간	T15	-15	25
COMMAND WRITE PORT 기입 ~ INT OFF 출력 지연 시간	T16	Tcyc - 5	Tcyc*5 + 65
END STATUS READ PORT 읽기 ~ INT OFF 출력 지연 시간	T17	Tcyc - 10	Tcyc*5 - 45
RESET* 입력 ~ 모든 출력 OFF 출력 지연 시간	T18		1,049

단위 nSec

## 8. 보충 사항

### 8.1. ICLK 입력 주파수의 선택

CAMC-FS 칩의 동작 SPEED는 ICLK입력 주파수에 의해 결정되고, 특히 아래의 4 항목에 대해서는 큰 영향을 미친다.

1) 출력 주파수의 설정 단위, 출력 주파수 및 가감속 시간 설정 단위, 가감속 시간은 다음식과 같다.  
( $F_{CLK}$  = ICLK 입력 주파수)

$$F_{UNIT} = \frac{F_{CLK}}{RANGE\ DATA \times 4 \times 65,536} \quad \text{--- 출력 주파수 설정 단(Pulse/Sec)}$$

$$F_{OUT} = F_{UNIT} \times SPEED\ DATA \quad \text{--- 출력 주파수 (Pulse/Sec)}$$

$$T_{UNIT} = \frac{RATE - n\ DATA \times 8}{F_{CLK}} \quad \text{--- 가감속 시간 설정 단위(직선) (Sec)}$$

$$T_{UNIT} = \frac{RATE - n\ DATA \times 16}{F_{CLK}} \quad \text{--- 가감속 시간 설정 단위(S-curve) (Sec)}$$

$$T_{UD} = T_{UNIT} \times (SPEED\ DATA\ A - SPEED\ DATA\ B) \quad \text{--- 가감속 시간 (Sec)}$$

2) 본 매뉴얼에 표시된 Tcyc 는 ICLK입력 주파수 이다.

3) 각종 DATA 기입 COMMAND에 있어서, COMMAND WRITE로부터 DATA가 내부로 LOAD되기 까지는 최대 Tcyc \* 8 의 시간이 필요하다. 따라서 COMMAND WRITE후 Tcyc \* 8 의 시간 동안 COMMAND PORT및 DATA1, 2, 3, 4 PORT로의 기입을 금지하여야 한다

4) 각종 DATA를 읽는 COMMAND에 있어서, COMMAND WRITE로부터 읽은 DATA가 출력 BUFFER에 LATCH될 때까지는 최대 Tcyc \* 8의 시간이 필요하다, 따라서 COMMAND WRITE후 DATA1, 2, 3, 4 PORT를 읽기까지는 Tcyc \* 8의 TIME DELAY가 필요하다.

이상의 4항목 중 SOFTWARE 작성에 영향을 미치는 것은 3), 4) 이고, 이 부담을 없애려면, CPU의 1 명령 실행 시간 > Tcyc \* 8 이 되는 사양의  $F_{CLK}$  을 선택한다. 이에 따라 3) 4)에 대한 고려는 불필요하고, PROGRAM SEQUENCE로 제어하는 것이 가능하다.

또, 보다 안정된 ( JITTER 현상이 적은) PULSE, PPLS파형을 얻기 위해서는 RANGE DATA에 가능한 한 큰 값을 설정하는 것이 필요한데, RANGE DATA에 가능한 한 큰 값을 설정하고, 덧붙여서 희망하

는 최고 주파수를 얻기 위해서는 FCLK을 최대한 크게 설정할 필요가 있다.

또한, 출력 펄스 주파수 계산치와 출력 디지털값을 정확하게 하기 위해서는 16.384 MHz, 8.192MHz 등 2의 지수승에 해당하는 주파수를 사용하는 것이 좋다.

## 8.2. 각종 PARAMETER 의 변경

CAMC-FS 칩은 DRIVE에 필요한 모든 PARAMETER를 USER PROGRAM에 의해 자유로이 설정할 수 있도록 설계되어 있다. 대부분의 PARAMETER는 상시 설정/변경이 가능하지만, DRIVE중의 PARAMETER 변경에 있어서는 그 결과를 십분 이해한 뒤에 실행할 필요가 있다.

DRIVE중에 변경할 경우의 주의점을 각 PARAMETER별로 나타내었다.

### ▶ RANGE DATA

주파수 설정 단위가 급격히 변화되면, PULSE MOTOR / SERVO MOTOR를 막론하고 정상적인 동작을 하지 못할 가능성이 있다. 특히, PRESET PULSE DRIVE중인 경우에는 정상적인 구동 동작이나 정상적인 가감속 정지를 기대할 수가 없다. (\* 참조)

### ▶ START/STOP SPEED DATA

감속 정지 시의 SPEED DATA가 변경되게 된다. PRESET PULSE DRIVE중에 실행된 경우에는 새로 지정된 START / STOP SPEED 까지 감속하지 않는 경우나, START / STOP SPEED 까지 감속한 후, 정지 시까지 시간이 걸리는 경우가 있다. (\* 참조)

### ▶ RATE DATA

가감속율이 변경될 수가 있다. PRESET PULSE DRIVE중에 실행한 경우에는 정상적인 감속 정지가 실행되지 않는 경우가 있다. (\* 참조)

\*PRESET PULSE DRIVE에 있어서, 감속 개시 POINT 검출 방식을 "자동 검출 방식"으로 한 경우, DRIVE형상은 "가속과 감속이 대칭을 이룬다"고 전제하고 감속 개시 POINT의 산출을 실행한다. 따라서 각종 PARAMETER의 변경에 의해 "가속과 감속의 대칭성"이 깨어지는 경우에는 정상적인 감속 정지를 기대할 수가 없게 된다.

## 8.3. DATA 의 ERROR 판정

CAMC-FS 칩에는 각종 설정 DATA에 대해서 CKECK기능이 있어서, 다음 중의 어느 하나라도 성립된 경우 DATA ERROR로 판정한다.

- RANGE DATA = 0
- START/STOP SPEED DATA = 0

- OBJECT SPEED DATA = 0
- START/STOP SPEED DATA > OBJECT SPEED DATA
- RATE DATA = 0

또, DATA ERROR로 판정된 경우의 동작은 다음과 같이 된다.

- 1) DATA ERROR 상태를 그대로 둔 채 DRIVE를 개시하는 경우  
 DRIVE COMMAND 실행 후 BUSY = H로 되지만, PULSE출력을 실행하지 않고 BUSY = L로 되며, DRIVE가 종료된다. 이때 END STATUS READ PORT내의 DTEED BIT가 1로 된다.  
 (단, 이런 경우를 만나도 SYNC신호, INP신호의 확인은 실행한다.)
  
- 2) DRIVE중에 DATA ERROR가 발생한 경우  
 DRIVE중(BUSY = H중)에 각종 DATA가 변경 되고, 그 결과 DATA ERROR가 발생한 경우, 현재 출력 중인 PULSE가 NOT ACTIVE로 되고, 이어서 BUSY = L로 되며, DRIVE가 종료된다. 이 때 END STATUS READ PORT내 DTEED BIT가 1로 된다.  
 (단, 이런 경우를 만나도 INP신호의 확인은 실행된다.)  
 주의) DRIVE중에 START/STOP SPEED DATA = 0, 또는 OBJECT SPEED DATA = 0로 된 경우, 현재 출력 중인 PULSE가 NOT ACTIVE로 되기 전에 발진이 정지되는 경우가 있다. 이럴 경우 출력 중인 PULSE가 ACTIVE상태에서 NOT ACTIVE상태로 이행할 수가 없고, BUSY BIT = 1인 채로 변화할 수가 없게 되므로 주의한다.

### 8.4.S字 가감속시 주의사항

S字 가감속 기능을 사용하고 바뀌어진 출력 PULSE 수가 적어서, 이른바 삼각 구동으로 된 경우에는 다음 그림 8.2과 같은 모양의 DRIVE로 되며, 기능이 제대로 발휘되지 않는 경우가 생기므로 주의한다.

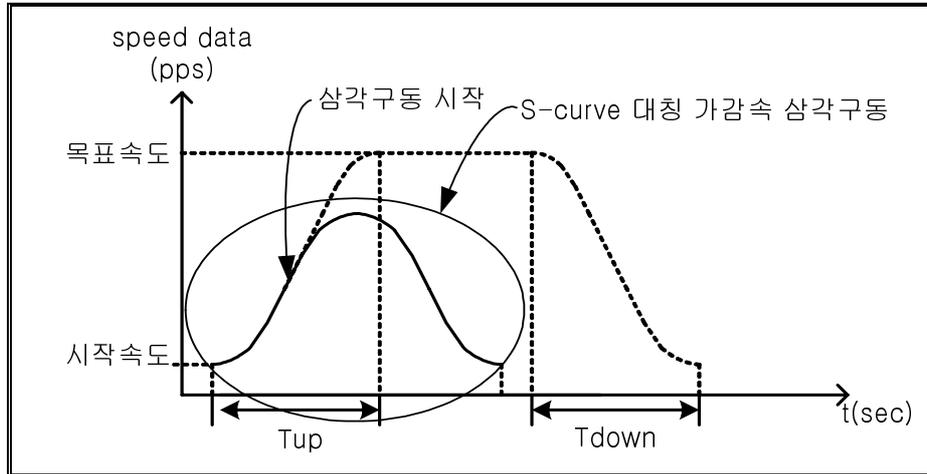


그림 8.1. 대칭 S자 Preset pulse drive 구동시 삼각구동 방지

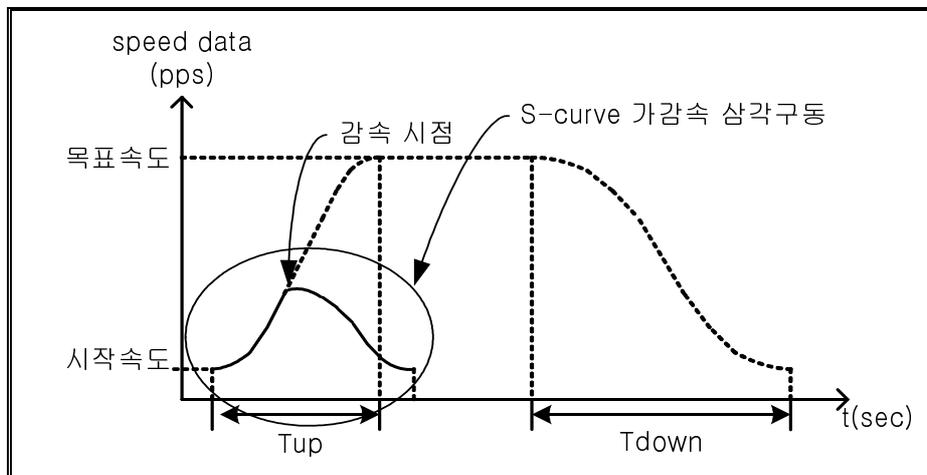


그림 8.2. 비대칭 S - curve 가감속 삼각구동

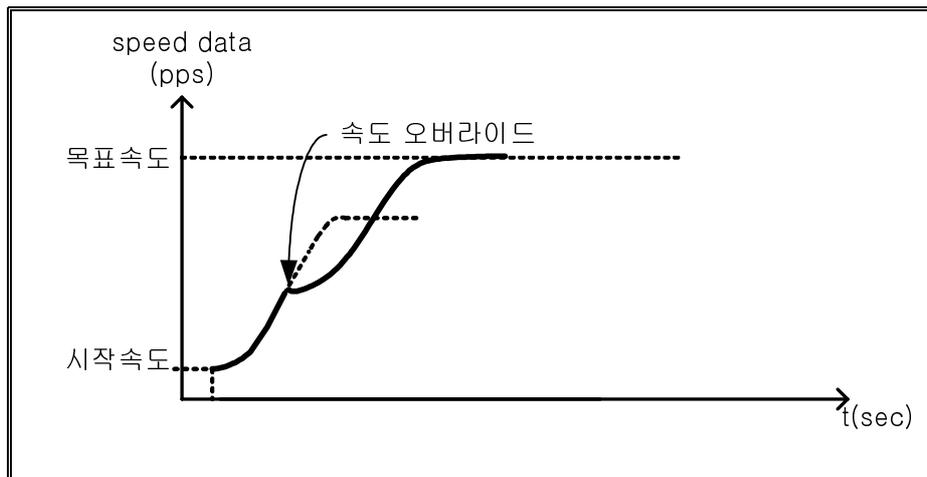


그림 8.3. 가속시 속도 오버라이드

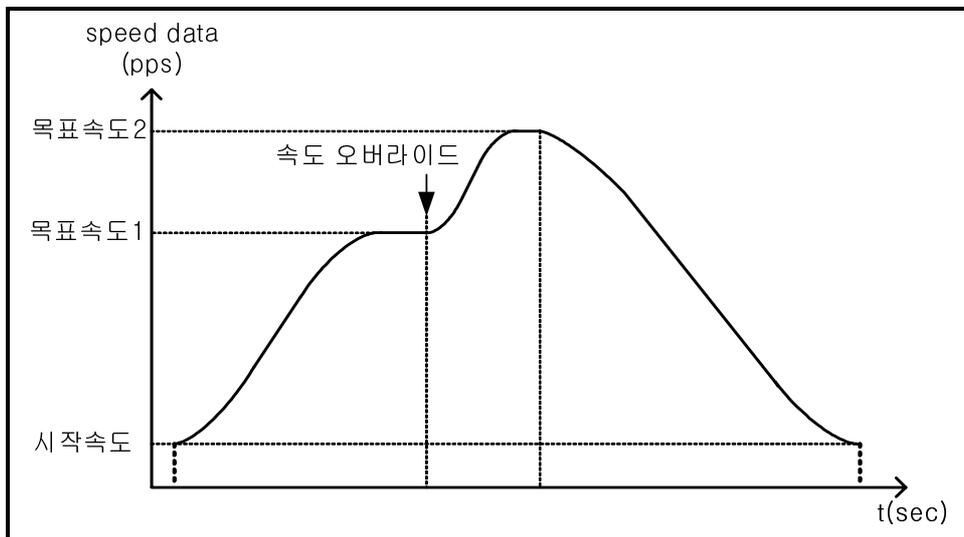


그림 8.4. S-curve 가감속시 속도 오버라이드

그림 8.4.는 드라이브 구동 중 속도 오버라이드는 OBJECT SPEED DATA WRITE COMMAND( CODE = 0X82, 0X92)를 사용하여 실행한다. 실행하는 OBJECT SPEED DATA WRITE COMMAND의 사용 데이터는 변경하고자 하는 속도 값을 사용하면 된다.

### 8.5.UP, CONST, DOWN 신호

CAMC-FS 칩에는 PULSE출력 상태를 판정하기 위해, UP, CONST, DOWN출력 신호가 갖추어져 있으며, 이들은 DRIVE STATUS READ PORT로부터도 상시 MONITORING할 수가 있다.

이들 신호에 의해 판정할 수 있는 것은 아래 표와 같다.

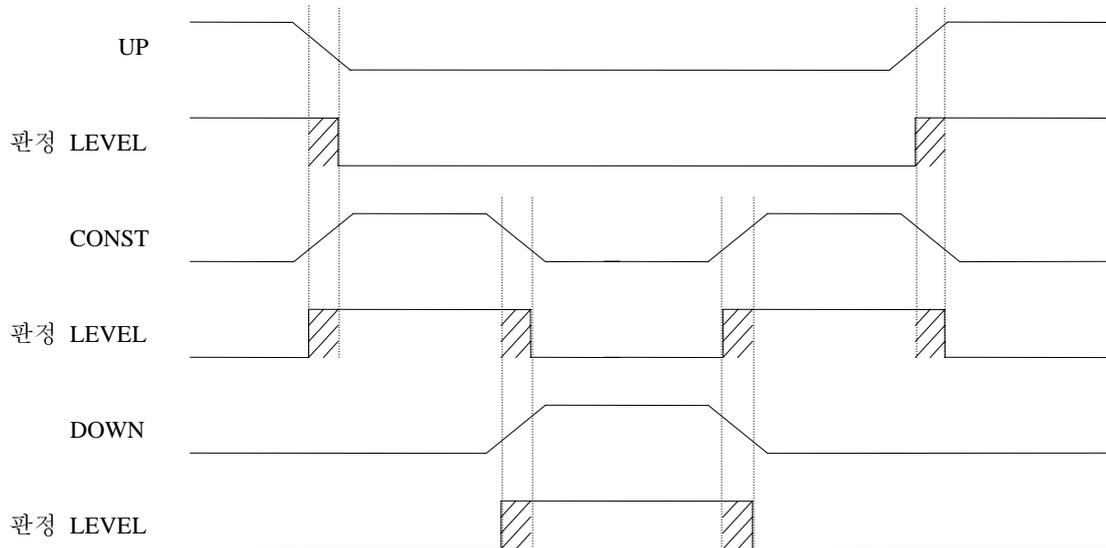
UP	CONST	DOWN	상 태
1	0	0	PULSE 출력 중이고 현재 가속 중에 있음

0	1	0	PULSE 출력 중이고 현재 정속 중에 있음
0	0	1	PULSE 출력 중이고 현재 감속 중에 있음
0	0	0	PULSE 출력을 하지 않고 있음
그 외			* 기능 없음

표에 설명된 대로 \* 로 표시된 상태는 논리적으로는 존재하지 않는 것으로 되어 있다.

그러나, 아래의 TIMING CHART에 표시된 대로, THRESHOLD LEVEL의 엇갈림과, 출력 지연 시간의 엇갈림에 의해 0 → 1의 변화 및 1 → 0의 변화가 동시에 이루어지기에는 한계가 있다.

따라서, 사선부에 표시된 대로 함께 1로 판정되거나, 또는 함께 0으로 판정되는 시간 (최대 20 nS정도) 가 존재할 가능성이 있다.

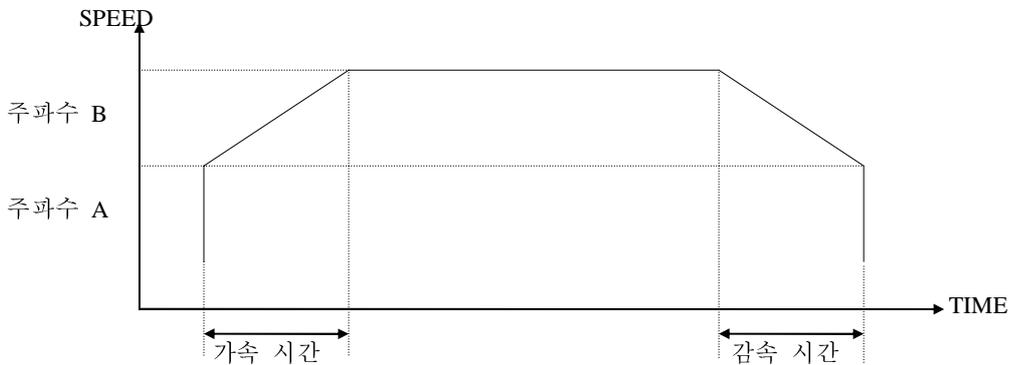


### 8.6.가감속 PULSE 의 산출

가속 및 감속부에 있어서 PULSE수는, 아래의 식으로 산출 할 수가 있다.

$$P_{UD} = (\text{주파수 A} + \text{주파수 B}) \times \text{가감속 시간} / 2$$

- P<sub>UD</sub> ----- 가감속 PULSE 수
- 주파수 A ----- 그림에 나타난 개시 / 정지시 주파수 (Pulse / Sec)
- 주파수 B ----- 그림에 나타난 최고 주파수 (Pulse / Sec)
- 가감속 시간 ----- 가속 또는 감속에 필요한 시간 (Sec)



이 식에서와 같이 가속 또는 감속 시 펄스 수는 가속/감속 주파수와 시간의 면적과 같다.

注意 ) CAMC-FS 칩은 DIGITAL LSI이다. 따라서 실제 동작에 있어서는 상기(上記)에 의해 산출된 값에 대해서 몇 PULSE분의 오차를 발생할 경우도 있다.

### 8.7. 미사용(未使用) 입력 단자의 처리

CAMC-FS 칩 입력 단자는 표준 값 50KΩ (최소값 18.5KΩ / 최대값 125KΩ)으로 내부 PULL\_UP되어 있다. 따라서, H LEVEL과 부조화를 일으키지 않는 미사용 단자에 대해서는 OPEN상태로 두어도 무방하다.

### 8.8. LIMIT 신호의 검출

CAMC-FS 칩에는 기계계(機械系)의 OVERRUN LIMIT 신호로서 정/역 방향 각각에 대해, 급정지용 / 감속 정지용 LIMIT 신호 입력 단자가 구비되어 있어서, DRIVE중에 입력된 경우에는 자동적으로 급정지 또는 감속 정지 한다.

더욱이 CAMC-FS 칩에는 신호 검출 DRIVE 기능 (+/- SIGNAL SEARCH - 1, - 2 DRIVE)이 있어서, 검출 대상 신호로서 앞서 말한 LIMIT 신호를 지정할 수가 있다. LIMIT 신호를 검출 대상 신호로 지정하고, +/- SIGNAL SEARCH - 1 DRIVE를 실행할 경우의 동작에서는 사용 시 주의가 필요하다. 이 경우 LIMIT 신호 검출에 의해 본래의 LIMIT 정지와 동시에 신호 검출에 동반하여 감속 정지가 일어난다. 입력된 LIMIT 신호가 감속 정지용인 경우에는 어떠한 정지 기능도 "감속"으로 되어 겹보기에 아무런

문제도 없지만, 입력된 LIMIT 신호가 급정지용 LIMIT 신호일 경우에는 LIMIT에 의해 "급정지"가 우선으로 된다. 따라서 신호 검출 후의 감속 정지를 희망하는 경우였다라도 결과는 급정지가 되고 희망하고 있던 동작과 다른 결과가 된다. 이와 같은 경우는 칩 기능 설정 COMMAND ( CODE = 0XF0 )에서 LIMIT 급정지/감속정지 설정 BIT를 0으로 설정하여 LIMIT 정지 기능 사용을 중지시킨 후 사용할 필요가 있다. 참고로 LIMIT 신호에 의한 급정지/감속정지 기능을 모두 사용하도록 설정한 경우 급정지와 감속 정지 신호가 동시에 입력된 경우에 있어서는 급정지가 우선이다.

### 8.9. 엔코더 2相 신호의 CHATTERING

CAMC-FS 칩에는 엔코더 입력 단자로 2상 신호를 입력하여 COUNT하는 EXTERNAL COUNTER를 내장하고 있다.

엔코더 2상 신호는 ROTARY ENCODER나 LINEAR SCALE등의 기계계로부터 전송되어 오는 경우가 많고, 미소한 진동에도 CHATTERING이 발생하는 경우가 많다. 엔코더 2상 신호를 COUNT하는 경우 CHATTERING을 어떻게 처리하는가가 중요한 POINT로 된다.

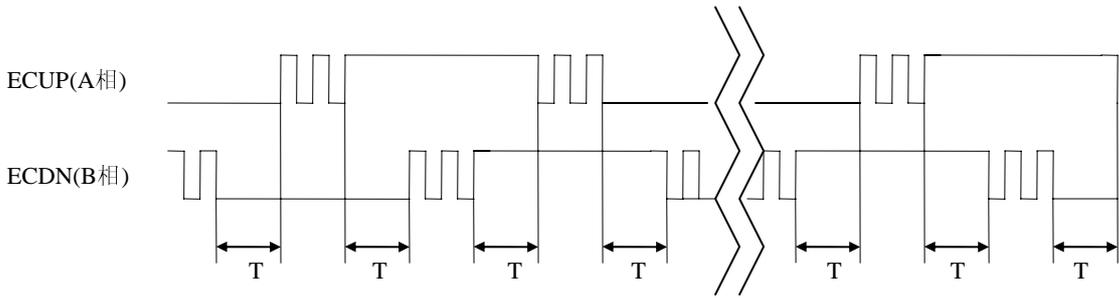
엔코더 2상 신호의 CHATTERING은 이론적으로는 초고속 UP / DOWN 동작의 조작을 반복하고, 이 원인으로 MISS COUNT를 일으키는 경우가 적지 않다.

그러나 간단한 적분 회로 등에 의해 CHATTERING을 흡수하는 경우 그런 폐해를 입히는 고속 COUNTER가 불가능하게 할 수 있다.

이상에 표시한 대로 2상 신호를 COUNT하는 경우에는 몇몇 문제가 있고 그것을 해결 하는데는 고도의 회로 기술이 필요하다.

그러나 CAMC-FS 칩에는 그러한 문제점의 해결을 위해 다음의 TIMING이 만족되는 한 COUNT값을 확실히 보증하는 DIGITAL FILTER가 내장되어 있다.

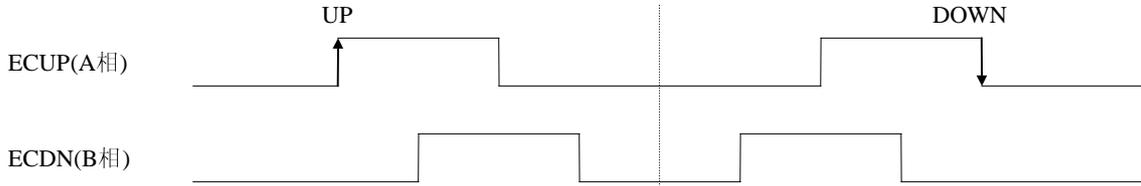
이것에 의해 CAMC-FS 칩에 2상 신호를 입력 시킬 경우 CHATTERING에 대한 대책은 따로 필요치 않고 전기적 NOISE에 대한 대책만 양호하면 입력 회로 등을 대폭 경감시킬 수가 있다.



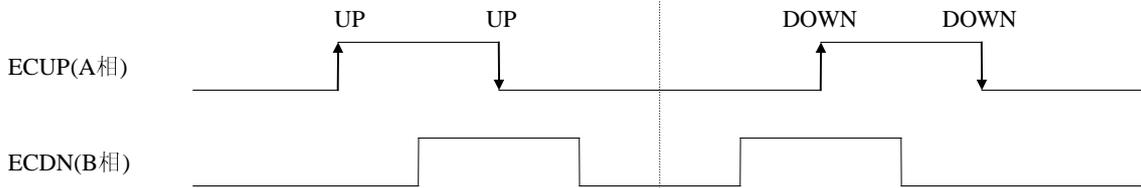
$T \geq T_{cyc} * 4 + 10 \text{ (nSec)}$  -----  $T_{cyc}$ 은 ICLK입력주기

### 8.10.엔코더 2 相 신호 COUNT POINT

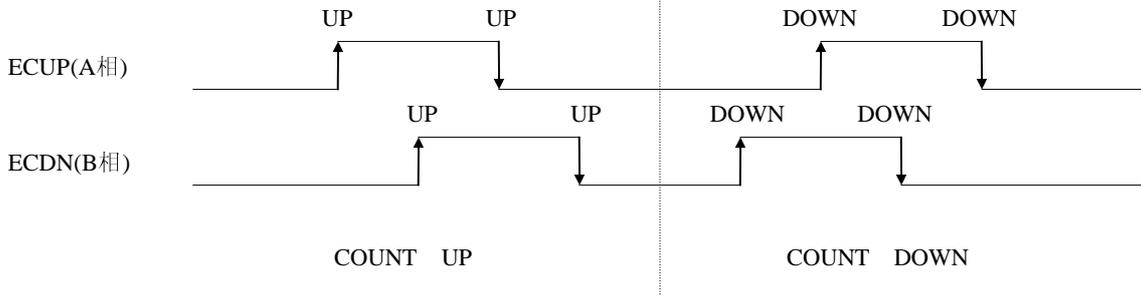
▶ 1배 시(遞倍時)



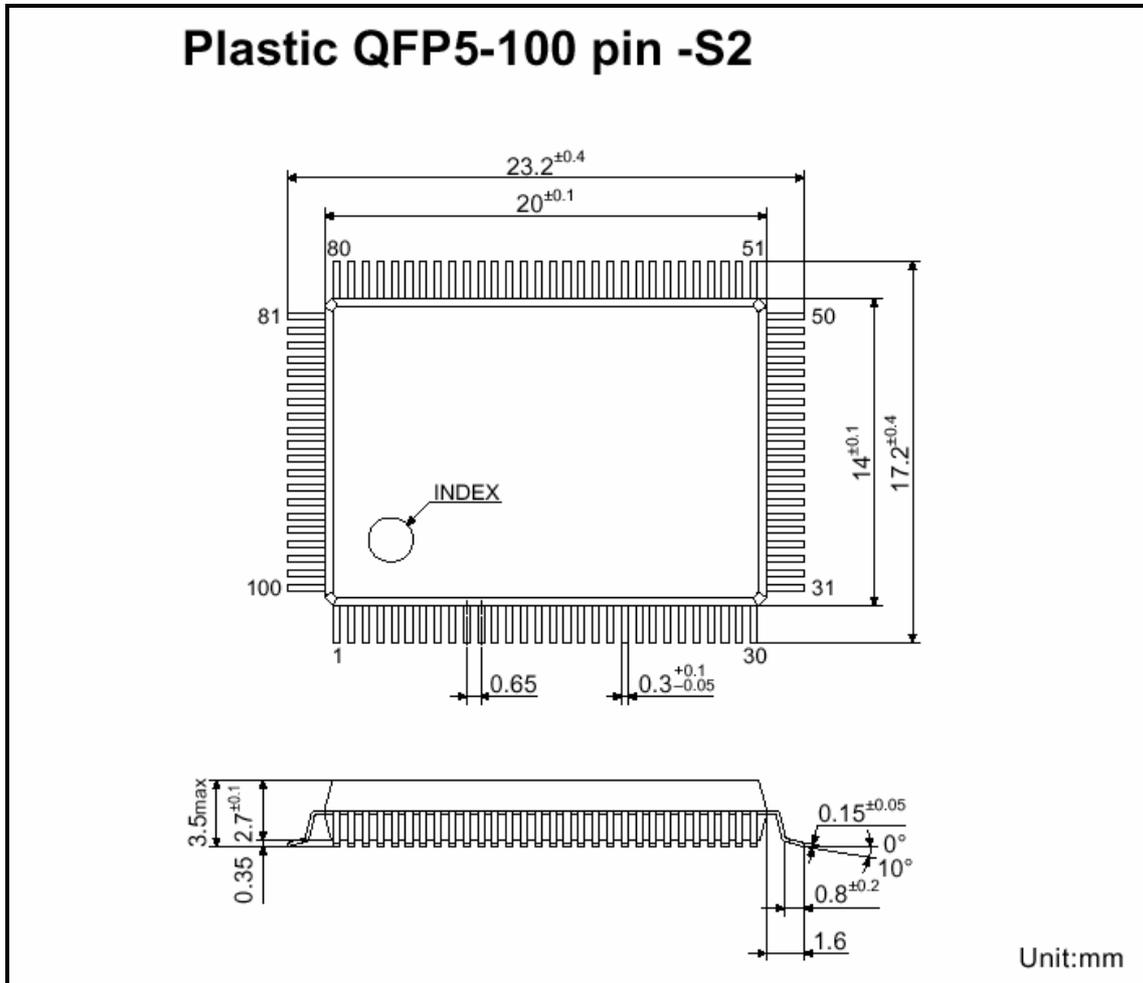
▶ 2배 시(遞倍時)



▶ 4배 시(遞倍時)



## 9. 외형 치수



## 10. 부록

### 10.1.Address Map

A2	A1	A0	8/16SEL	CS*	RD*	WR*	Operation	비고
0	0	0	0	0	1	0	Data1 write (2 <sup>31</sup> ~2 <sup>24</sup> bit 데이터)	8bit access
0	0	1	0	0	1	0	Data2 write (2 <sup>23</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	0	0	0	1	0	Data3 write (2 <sup>15</sup> ~2 <sup>8</sup> bit 데이터)	"
0	1	1	0	0	1	0	Data4 write (2 <sup>7</sup> ~2 <sup>0</sup> bit 데이터)	"
1	0	0	0	0	1	0	Command write	"
1	0	1	0	0	1	0	사용하지 않음	"
1	1	0	0	0	1	0	사용하지 않음	"
1	1	1	0	0	1	0	사용하지 않음	"
0	0	0	0	0	0	1	Data1 read (2 <sup>31</sup> ~2 <sup>24</sup> bit 데이터)	"
0	0	1	0	0	0	1	Data2 read (2 <sup>23</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	0	0	0	0	1	Data3 read (2 <sup>15</sup> ~2 <sup>8</sup> bit 데이터)	"
0	1	1	0	0	0	1	Data4 read (2 <sup>7</sup> ~2 <sup>0</sup> bit 데이터)	"
1	0	0	0	0	0	1	사용하지 않음	"
1	0	1	0	0	0	1	사용하지 않음	"
1	1	0	0	0	0	1	사용하지 않음	"
1	1	1	0	0	0	1	사용하지 않음	"
0	0	0	1	0	1	0	Data1,Data2 write (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	16bit access
0	0	1	1	0	1	0	사용하지 않음	
0	1	0	1	0	1	0	Data1,Data2 write (2 <sup>15</sup> ~2 <sup>0</sup> bit 데이터)	"
0	1	1	1	0	1	0	사용하지 않음	
1	0	0	1	0	1	0	Command write	"
1	0	1	1	0	1	0	사용하지 않음	
1	1	0	1	0	1	0	사용하지 않음	
1	1	1	1	0	1	0	사용하지 않음	
0	0	0	1	0	0	1	Data1,Data2 read (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	"
0	0	1	1	0	0	1	사용하지 않음	
0	1	0	1	0	0	1	Data1,Data2 read (2 <sup>31</sup> ~2 <sup>16</sup> bit 데이터)	"
0	1	1	1	0	0	1	사용하지 않음	
1	0	0	1	0	0	1	사용하지 않음	
1	0	1	1	0	0	1	사용하지 않음	
1	1	0	1	0	0	1	사용하지 않음	
1	1	1	1	0	0	1	사용하지 않음	

## 10.2.PORT 설명

### ▶ DATA1 WRITE PORT

각종 설정 DATA의  $2^{31} \sim 2^{24}$  Bit Data를 써 넣는다.

D7	—	$2^{31}$ BIT
D6	—	$2^{30}$ BIT
D5	—	$2^{29}$ BIT
D4	—	$2^{28}$ BIT
D3	—	$2^{27}$ BIT
D2	—	$2^{26}$ BIT
D1	—	$2^{25}$ BIT
D0	—	$2^{24}$ BIT

### ▶ DATA2 WRITE PORT

각종 설정 DATA의  $2^{23} \sim 2^{16}$  Bit Data를 써 넣는다.

D7	—	$2^{23}$ BIT
D6	—	$2^{22}$ BIT
D5	—	$2^{21}$ BIT
D4	—	$2^{20}$ BIT
D3	—	$2^{19}$ BIT
D2	—	$2^{18}$ BIT
D1	—	$2^{17}$ BIT
D0	—	$2^{16}$ BIT

### ▶ DATA3 WRITE PORT

각종 설정 DATA의  $2^{15} \sim 2^8$  Bit Data를 써 넣는다.

D7	—	$2^{15}$ BIT
D6	—	$2^{14}$ BIT
D5	—	$2^{13}$ BIT
D4	—	$2^{12}$ BIT
D3	—	$2^{11}$ BIT
D2	—	$2^{10}$ BIT
D1	—	$2^9$ BIT
D0	—	$2^8$ BIT

▶ DATA4 WRITE PORT

각종 설정 DATA의  $2^7 \sim 2^0$  Bit Data를 써 넣는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

▶ COMMAND WRITE PORT

COMMAND를 써 넣는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

▶ DATA1 READ PORT

각종 설정 DATA의  $2^{31} \sim 2^{24}$  Bit Data를 읽는다.

D7	—	$2^{31}$ BIT
D6	—	$2^{30}$ BIT
D5	—	$2^{29}$ BIT
D4	—	$2^{28}$ BIT
D3	—	$2^{27}$ BIT
D2	—	$2^{26}$ BIT
D1	—	$2^{25}$ BIT
D0	—	$2^{24}$ BIT

▶ DATA2 READ PORT

각종 설정 DATA의  $2^{23} \sim 2^{16}$  Bit Data를 읽는다.

D7	—	$2^{23}$ BIT
D6	—	$2^{22}$ BIT
D5	—	$2^{21}$ BIT
D4	—	$2^{20}$ BIT
D3	—	$2^{19}$ BIT
D2	—	$2^{18}$ BIT
D1	—	$2^{17}$ BIT
D0	—	$2^{16}$ BIT

▶ DATA3 READ PORT

각종 설정 DATA의  $2^{15} \sim 2^8$  Bit Data를 읽는다.

D7	—	$2^{15}$ BIT
D6	—	$2^{14}$ BIT
D5	—	$2^{13}$ BIT
D4	—	$2^{12}$ BIT
D3	—	$2^{11}$ BIT
D2	—	$2^{10}$ BIT
D1	—	$2^9$ BIT
D0	—	$2^8$ BIT

▶ DATA4 READ PORT

각종 설정 DATA의  $2^7 \sim 2^0$  Bit Data를 읽는다.

D7	—	$2^7$ BIT
D6	—	$2^6$ BIT
D5	—	$2^5$ BIT
D4	—	$2^4$ BIT
D3	—	$2^3$ BIT
D2	—	$2^2$ BIT
D1	—	$2^1$ BIT
D0	—	$2^0$ BIT

## 10.3. COMMAND 일람표(Compact, register map)

## ▶ CAMC-FS READ COMMAND

CODE(HEX)	이 름 (PGM-1 GROUP 레지스터 READ)	W	DEFAULT
00	PGM-1 Range data[RANGE]	16	0xFFFF
01	PGM-1 Start/Stop speed data[STD]	16	0x0001
02	PGM-1 Object speed data[OBJ]	16	0x0001
03	PGM-1 Rate-1 data[RATE1]	16	0xFFFF
04	PGM-1 Rate-2 data[RATE2]	16	0xFFFF
05	PGM-1 Rate-3 data[RATE3]	16	0xFFFF
06	PGM-1 Rate change point(RATE1 → RATE2)[RCP1]	16	0xFFFF
07	PGM-1 Rate change point(RATE2 → RATE3)[RCP2]	16	0xFFFF
08	PGM-1 S profile region width-1 data[SW1]	15	0x7FFF
09	PGM-1 S profile region width-2 data[SW2]	15	0x7FFF
0A	PGM-1 PWM output confiture data[PWM]	3	0x00
0B	PGM-1 Slowdown/rear pulse amount data[REAR]	32	0x00000000
0C	PGM-1 Current speed data[SPD]	16	0x0000
0D	PGM-1 Current speed compare data[SPDCMP]	16	0x0000
0E	PGM-1 Drive pulse amount data[DRVPULSE]	32	0x00000000
0F	PGM-1 Preset pulse amount data[PRESETPULSE]	32	0x00000000

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 READ)	W	DEFAULT
10	PGM-1 up-date Range data[URANGE]	16	0xFFFF
11	PGM-1 up-date Start/Stop speed data[USTD]	16	0x0001
12	PGM-1 up-date Object speed data[UOBJ]	16	0x0001
13	PGM-1 up-date Rate-1 data[URATE1]	16	0xFFFF
14	PGM-1 up-date Rate-2 data[URATE2]	16	0xFFFF
15	PGM-1 up-date Rate-3 data[URATE3]	16	0xFFFF
16	PGM-1 up-date Rate change point (URATE1 → URATE2)[URCP1]	16	0xFFFF
17	PGM-1 up-date Rate change point (URATE2 → URATE3)[URCP2]	16	0xFFFF
18	PGM-1 up-date S profile region width-1 data[USW1]	15	0x7FFF
19	PGM-1 up-date S profile region width-2 data[USW2]	15	0x7FFF
1A	No operation[NOP]		

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 READ)	W	DEFAULT
1B	PGM-1 up-date Slowdown/rear pulse amount data[UREAR]	32	0X00000000
1C	PGM-1 Current speed data(same with 0x0C)[SPD]	16	0x0000
1D	PGM-1 Current speed compare data(same with 0x0D) [SPDCMP]	16	0x0000
1E	PGM-1 Drive pulse amount data(same with 0x0E) [DRVPULSE]	32	0X00000000
1F	PGM-1 Preset pulse amount data (same with 0x0F) [PRESETPULSE]	32	0X00000000

CODE(HEX)	이 름 (PGM-2 GROUP 레지스터 READ)	W	DEFULT
20~2C	No operation[NOPI]		
2D	Drive mode configure data[DRIVEMODE] 2bit : Source selection for deceleration point search ‘0’(Use internal count), ‘1’(Use external count) 01~01 bit : Profile selection “00” : Symetric Trapezoid “01” : Unsymetric Trapezoid “10” : Symetric S curve “11” : Unsymetric S curve	3	0x0
2E	Mpg operation configure data[MPGCON] 6bit : Continuouse drive start by EXPP, EXMP; ‘0’(Reset), ‘1’(Set) 5bit : Preset drive start by EXPP, EXMP; ‘0’(Reset), ‘1’(Set) 03~04 bit : Quadrature EXPP, EXMP signal mode “00” : 1 phase pulse mode “01” : 2 phase pulse mode(One time) “10” : 2 phase pulse mode(Two times) “11” : 2 phase pulse mode(Four times) 2 bit : Mpg drive direction use configure ‘0’(Plus direction), ‘1’(Minus direction) 1 bit : Mpg drive direction source configure ‘0’(EXPP, EXMP signal dir.), ‘1’(User configure dir.) 0bit : Slave mpg drive start by EXPP, EXMP ‘0’(Reset), ‘1’(Set)	7	0x00
2F	Mpg preset drive pulse amount[PULSEMPG]	32	0x00000000

CODE(HEX)	이 름 ( 확장 GROUP 레지스터 READ )	W	DEFULT
30~35	No operation[NOPI]		

CODE(HEX)	이름 ( 확장 GROUP 레지스터 READ )	W	DEFULT
36	Software limit configure data[SWLMTCON] 2 bit : Software limit compare source selection '0'(Internal count), '1'(External count) 1 bit : Software limit execution mode '0'(Emergency stop), '1'(Slow down stop) 0 bit : Soft limit Enable/Disable '0'(Disable), '1'(Enable)	3	0x0
37	Minus software limit compare data[MSWLMTCOMP]	32	0x80000000
38	Plus software limit compare data[PSWLMTCOMP]	32	0x7FFFFFFF
39	Trigger configure data[TRGCON] 31~16 bit : Trigger active pulse width(unsigned) Width = $T_{clk} \times$ (Decimal value of trigger active pulse width) If Width = 0 then Initialize Trigger function. 15~ 02 bit : Don't care. 1 bit : Trigger compare source selection '0'(Internal), '1'(External) 0 bit : Trigger mode '0'(Absolute), '1'(Periodic)	32	0x00000000
3A	Trigger compare data[TRGCOMP]	32	0x00000000
3B	Internal count minus limit data[ICM]	32	0x80000000
3C	External count minus limit data[ECM]	32	0x80000000
3D~3F	No operation[NOP]		

CODE(HEX)	이름 ( 스트립터 GROUP 레지스터 READ )	W	DEFULT
40	Script control register-1[SCRCON1] 31 bit : '0'(One time execution, '1' : Always) 30~26bit : Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign 7~ 0 bit : Execution command (write command only)	32	0x00000000

CODE(HEX)	이름 (스트립터 GROUP 레지스터 READ)	W	DEFULT
	*See event list table for event assign		
41	Script control register-2[ <b>SCRCON2</b> ] (Same with script control register-1)	32	0x00000000
42	Script control register-3[ <b>SCRCON3</b> ] (Same with script control register-1)	32	0x00000000
43	Script control queue[ <b>SCRCONQ</b> ] (Queue bottom contents) 31 bit : '0'(One time execution, '1' : Always) 30 bit : Interrupt generation when executed '0'(Disable), '1'(Enable) 29~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~ 16 bit : Second event assign 15~ 8 bit : First event assign * When "0xFF", No check event, just execute command 7~ 0 bit : Execution command (write command only)	32	0x00000000
44	Script data register-1[ <b>SCRDATA1</b> ]	32	0x00000000
45	Script data register-2[ <b>SCRDATA1</b> ]	32	0x00000000
46	Script data register-3[ <b>SCRDATA1</b> ]	32	0x00000000
47	Script data queue (Queue bottom contents) [ <b>SCRDATAQ</b> ]	32	0x00000000
48	No operation[ <b>NOP</b> ]		
49	Script control queue index(max to 0xD)[ <b>SCRCQSIZE</b> ]	4	0x0
4A	Script data queue index(max to 0xD) [ <b>SCRDQSIZE</b> ]	4	0x0
4B	Script queue Full/Empty flag[ <b>SCRQSLAG</b> ] 3 bit : Script control queue full flag '0'(False), '1'(True) 2 bit : Script control queue empty flag '0'(False), '1'(True) 1 bit : Script data queue full flag '0'(False), '1'(True) 0 bit : Script data queue empty flag '0'(False), '1'(True)	4	0x5
4C	Script queue size configure data (1~13) [ <b>SCRQSIZECON</b> ] 15~ 12 bit : Script control queue full size 11~ 08 bit : Script control queue empty size	16	0xD0D0

CODE(HEX)	이름 (스트립터 GROUP 레지스터 READ)	W	DEFULT
	07~ 04 bit : Script data queue full size 03~ 00 bit : Script data queue empty size		
4D	Script queue status[SCRQSTATUS] 11~ 08 bit : Script control queue index(same with 0x49) 07~ 04 bit : Script data queue index(same with 0x4A) 3 bit : Script control queue full flag '0'(False), '1'(True) 2 bit : Script control queue empty flag '0'(False), '1'(True) 1 bit : Script data queue full flag '0'(False), '1'(True) 0 bit : Script data queue empty flag '0'(False), '1'(True)	12	0x005
4E~4F	No operation[NOP]		

CODE(HEX)	이름 (갈무리 GROUP 레지스터 READ)	R	DEFAULT
50	Caption control register-1[CAPCON1] 31 bit : '0'(One time execution, '1' : Always) 30~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign 7~ 0 bit : Execution command (read command only) *See event list table for event assign	32	0x00000000
51	Caption control register-2[CAPCON2] (Same with script control register-1)	32	0x00000000
52	Caption control register-3[CAPCON3] (Same with script control register-1)	32	0x00000000
53	Caption control queue[CAPCONQ] 31 bit : '0'(One time execution, '1' : Always) 30 bit : Interrupt generation when executed '0'(Disable), '1'(Enable) 29~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only	32	0x00000000

CODE(HEX)	이름 (갈무리 GROUP 레지스터 READ)	R	DEFAULT
	“01” : OR operation “10” : AND operation “11” : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign * When “0xFF”, No check event, just execute command 7~ 0 bit : Execution command (read command only) *See event list table for event assign		
54	Caption control register-1 execution result [CAPDATA1]	32	0x00000000
55	Caption control register-2 execution result [CAPDATA2]	32	0x00000000
56	Caption control register-3 execution result [CAPDATA3]	32	0x00000000
57	Caption control queue execution result data queue [CAPDATAQ]	32	0x00000000
58	No operation[NOP]		
59	Caption control queue index(max to 0xD) [CAPCQSIZE]	4	0x0
5A	Caption result data queue index(max to 0xD) [CAPDQSIZE]	4	0x0
5B	Caption queue Full/Empty flag[CAPQSLAG] 6 bit : Caption control resiger-1 result data updated ‘0’(False), ‘1’(True) 5 bit : Caption control resiger-2 result data updated ‘0’(False), ‘1’(True) 4 bit : Caption control resiger-3 result data updated ‘0’(False), ‘1’(True) 3 bit : Caption control queue full flag ‘0’(False), ‘1’(True) 2 bit : Caption control queue empty flag ‘0’(False), ‘1’(True) 1 bit : Caption result data queue full flag ‘0’(False), ‘1’(True) 0 bit : Caption result data queue empty flag ‘0’(False), ‘1’(True)	7	0x5
5C	Caption queue size configure data (0~13) [CAPQSIZECON] 15~ 12 bit : Caption control queue full size 11~ 08 bit : Caption control queue empty size 07~ 04 bit : Caption result data queue full size 03~ 00 bit : Caption result data queue empty size	16	0xD0D0
5D	Caption queue status[CAPQSTATUS] 11~ 08 bit : Caption control queue index(same with 0x59) 07~ 04 bit : Caption data queue index(same with 0x5A) 3 bit : Caption control queue full flag ‘0’(False), ‘1’(True)	12	0x005

CODE(HEX)	이름 (갈무리 GROUP 레지스터 READ)	R	DEFAULT
	2 bit : Caption control queue empty flag '0'(False), '1'(True) 1 bit : Caption result data queue full flag '0'(False), '1'(True) 0 bit : Caption result data queue empty flag '0'(False), '1'(True)		
5E~5F	No operation[ <b>NOP</b> ]		

CODE(HEX)	이름 (BUS-1 GROUP 레지스터 READ)	W	DEFAULT
60	Internal count data(Signed)[ <b>INCNT</b> ]	32	0x00000000
61	Internal count compare data(Signed)[ <b>INCNTCMP</b> ]	32	0x00000000
62	Internal count scale data(unsigned) [ <b>INCNTSCALE</b> ]	8	0x00
63	Internal count plus limit data[ <b>ICP</b> ](Use 0x3B for minus limit)	32	0x7FFFFFFF
64	External count data(Signed)[ <b>EXCNT</b> ]	32	0x00000000
65	External count compare data(Signed)[ <b>EXCNTCMP</b> ]	32	0x00000000
66	External count scale data(unsigned)[ <b>EXCNTSCALE</b> ]	8	0x00
67	External count plus limit data[ <b>ECP</b> ](Use 0x3C for minus limit)	32	0x7FFFFFFF
68	External speed data[ <b>EXSPD</b> ]	32	0x00000000
69	External speed compare data[ <b>EXSPDCMP</b> ]	32	0x00000000
6A	External signal filter depth data[ <b>EXFILTERD</b> ] 7~00 bit : Digital Filter for input signals	8	0x05
6B	Off region(over run or under run) decision data[ <b>OFFREGION</b> ]	8	0x00
6C	Deviation data(INCNT – EXCNT)[ <b>DEVIATION</b> ] 15 bit : Result sign '0'(Plus), '1'(Minus) 14~00 bit : Absolute value of deviation	16	0x0000
6D	PGM register bank selection configure data[ <b>PGMCH</b> ] '0'(PGM-1 bank use), '1'(PGM-1 update bank use)	1	0x0
6E	Comparator source selection configure[ <b>COMPCON</b> ] "00" : 1'st : Internal count, 2'nd : External count "01" : 1'st : Internal count, 2'nd : Internal count "10" : 1'st : External count, 2'nd : External count	2	0x0
6F	No operation[ <b>NOP</b> ]		

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
70	Function configure data [ <b>FUNCON</b> ] Each bit : '0'(Reset, Disable), '1'(Set, Enable) 12bit : S profile triangle drive configure 11bit : Search drive source filter configure	13	0x0C3E

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
	10bit : SYNC input signal function configure 9 bit : Limit complete stop configure 8 bit : INPOS input signal expanded function configure 7bit : Off region stop mode '0'(Emergency stop), '1'(Slowdown stop) 6 bit : Drive at start/stop speed '0'(Stop at <b>STD</b> speed), '1'(Continuous drive at <b>STD</b> speed) 5 bit : NSLM/PSLM input signal function configure 4 bit : NELM/PELM input signal function configure 3 bit : ESTOP/SSTOP signal function configure 2 bit : Don't care 1 bit : ALARM input signal function configure 0 bit : INPOS input signal basic function configure		
71	Mode1 data [ <b>MODE1</b> ] 7 bit : Preset drive slowdown point decision method '0'(Auto), '1'(Manual) 06~04 bit : PULSE/DIR output signal configure(* See table) 03~00 bit : Signal search drive source configure(* See table)	8	0x000
72	Mode2 data[ <b>MODE2</b> ] Active level : '0'(Active low), '1'(Active high) 10 bit : TRIG output signal active level 9 bit : INT output signal active level 8 bit : MARK input signal active level 07~06 bit : Encoder signal count mode(ECUP/ECDN input signal) "00" : Single phase, (ECUP : up count, ECDN : down count) "01" : Two phases, 1 times multiplied "10" : Two phases,, 2 times multiplied "11" : Two phases,, 4 times multiplied 5 bit : INPOS input signal active level 4 bit : ALARM input signal active level 3 bit : NSLM input signal active level 2 bit : PSLM input signal active level 1 bit : NELM input signal active level 0 bit : PELM input signal active level	11	0x200
73	Universal In/Out signal data[ <b>UIODATA</b> ]	11	0x000

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
	10~08 bit : Universal output bit operation “0xx” : Operand bypass “100” : NOT Current output “101” : Output operand AND Current output “110” : Output operand OR Current output “111” : Output operand XOR Current output 07~04 bit : Universal Input signal 03~00 bit : Universal Output signal		
74	End status[ENDSTATUS] 14 bit : Drive end by limit 13 bit : Drive end by limit complete stop 12 bit : Sensor positioning drive end 11 bit : Preset pulse drive end 10 bit : Signal search drive end 9 bit : Original search drive end 8 bit : Drive end by off region error 7 bit : Drive end by data error 6 bit : Drive end by ALARM signal function 5 bit : Drive end by emergency stop command(0xAC) 4 bit : Drive end by slowdown stop command(0xAB) 3 bit : Drive end by ESTOP signal function 2 bit : Drive end by SSTOP signal function 1 bit : Drive end by PELM/NELM, software emergency limit 0 bit : Drive end by PSLM/NSLM, software slowdown limit	15	0x0000
75	Mechanical data[MECH] 12 bit : ESTOP input signal level 11 bit : SSTOP input signal level 10 bit : MARK input signal is level 9 bit: EXPP input signal level 8 bit: EXMP input signal level 7 bit : ECUP input signal level 6 bit : ECDN input signal level 5 bit : INPOS input signal active status 4 bit : ALARM input signal active status 3 bit : NSLM input signal active status	13	

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
	2 bit : PSLM input signal active status 1 bit : NELM input signal active status 0 bit : PELM input signal active status		
76	Drive status[ <b>DRVSTATUS</b> ] 8 bit: Drive direction '0'(CW), '1'(CCW) 7 bit : ECG( <b>EXCNTCMP</b> less than <b>EXCNT</b> ) 6 bit : ECL( <b>EXCNTCMP</b> greater than <b>EXCNT</b> ) 5 bit : ICG( <b>INCNTCMP</b> less than <b>INCNT</b> ) 4 bit : ICL( <b>INCNTCMP</b> greater than <b>INCNT</b> ) 3 bit : UP (In acceleration drive) 2 bit : CONST (In constant speed drive) 1 bit : DOWN (In deceleration drive) 0 bit : BUSY (On driving)	9	
77	External count clear configure data[ <b>EXCNTCLR</b> ] 8 bit : External count clear flag(clear when write 0xF7) 7 bit : External count mode '0' : Normal count, '1': Reverse count 6 bit : Don't care 05~04 bit : Clear operation mode "00" : Clear request reset "01" : One time clear request set "10" : Full time clear request set "11" : Clear external count and LOCK to zero at clear input signal activated 03~00 bit : Clear input signal assign "0001" : IN0 "0010" : IN1 "0011" : IN2 "0100" : IN3 Others : Don't care(NO clear)	9	0x000
78	No operation[ <b>NOP</b> ]		
79	Interrupt flag data[ <b>INTFALG</b> ] Each bit : '0'(Interrupt inactivated), '1'( Interrupt activated) 31 bit : Selectable Interrupt source3(refer "0xFE" command) 30 bit : Selectable Interrupt source2(refer "0xFE" command)	32	0x00000000

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
	29 bit : Selectable Interrupt source1(refer "0xFE" command) 28 bit : Selectable Interrupt source0(refer "0xFE" command) 27 bit : NELM/PELM input signal activated 26 bit : NSLM/PSLM input signal activated 25 bit : Caption queue command executed when CAPCONQ's 30 bit is '1' 24 bit : Script queue command executed when SCRCONQ's 30 bit is '1' 23 bit : 1'st counter value is equal with minus limit data 22 bit : Current speed is equal with current speed compare data 21 bit : Current speed is equal with rate change point <b>RCP12</b> 20 bit : Current speed is equal with rate change point <b>RCP23</b> 19 bit : Search signal assigned in <b>MODE1</b> is detected 18 bit : Acceleration drive started 17 bit : Constant speed drive started 16 bit : Deceleration drive started 15 bit : Interrupt generated by command "0xFF" 14 bit : Caption control register-3 command is executed 13 bit : Caption control register-2 command is executed 12 bit : Caption control register-1 command is executed 11 bit : Script control register-3 command is executed 10 bit : Script control register-2 command is executed 9 bit : Script control register-1 command is executed 8 bit : Caption result queue is FULL 7 bit : Script control queue is EMPTY 6 bit : ECL( <b>EXCNTCMP</b> is greater than <b>EXCNT</b> ) 5 bit : ECE( <b>EXCNTCMP</b> is equal with <b>EXCNT</b> ) 4 bit : ECG( <b>EXCNTCMP</b> is less with <b>EXCNT</b> ) 3 bit : ICL( <b>INCNTCMP</b> is greater than <b>INCNT</b> ) 2 bit : ICE( <b>INCNTCMP</b> is equal with <b>INCNT</b> ) 1 bit : ICG( <b>INCNTCMP</b> is less than <b>INCNT</b> ) 0 bit : Drive end		
7A	Interrupt mask data[ <b>INTMASK</b> ] Each bit : '0'(Interrupt inactivated), '1'( Interrupt activated)	32	0x00000001
7B	EMODE1 data [ <b>EMODE1</b> ]	8	0x00

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 READ)	W	DEFAULT
	7 bit : Extension mode enable(0 : Disable, 1 : Enable) 0 : md_out[12:8] => md_out[12:8] 1 : md_out[12:8] => EUIO 6~ 5 bit : Don't care 4 bit : EUO4 enable/disable(0 : disable, 1 : enable) 3 bit : EUO3 enable/disable(0 : disable, 1 : enable) 2 bit : EUO2 enable/disable(0 : disable, 1 : enable) 1 bit : EUO1 enable/disable(0 : disable, 1 : enable) 0 bit : EUIO0 direction select(0 : input, 1 : output)		
7C	Extension universal in/out[EUIODATA] 07~05 bit : Extension Universal output bit operation "0xx" : Operand bypass "100" : NOT Current output "101" : Output operand AND Current output "110" : Output operand OR Current output "111" : Output operand XOR Current output 4 ~ 0 bit : Extension Universal in/output	8	0x00
7D	No operation[ <b>NOP</b> ]		
7E	User Interrupt source configure data[ <b>INTUSERCON</b> ] Same with Script/Caption event list(except "0xFF" event) 31~24 bit : User selectable interrupt 3 source selection 23~16 bit : User selectable interrupt 2 source selection 15~ 08 bit : User selectable interrupt 1 source selection 07 ~00 bit : User selectable interrupt 0 source selection	32	0x00000000
7F	No operation[ <b>NOP</b> ]		

## ▶ CAMC-FS WRITE COMMAND

CODE(HEX)	이름 (PGM-1 GROUP 레지스터 WRITE)	W	DEFAULT
80	PGM-1 Range data[RANGE]	16	0xFFFF
81	PGM-1 Start/Stop speed data[STD]	16	0x0001
82	PGM-1 Object speed data[OBJ]	16	0x0001
83	PGM-1 Rate-1 data[RATE1]	16	0xFFFF
84	PGM-1 Rate-2 data[RATE2]	16	0xFFFF
85	PGM-1 Rate-3 data[RATE3]	16	0xFFFF
86	PGM-1 Rate change point(RATE1 → RATE2)[RCP1]	16	0xFFFF
87	PGM-1 Rate change point(RATE2 → RATE3)[RCP2]	16	0xFFFF
88	PGM-1 S profile region width-1 data[SW1]	15	0x7FFF
89	PGM-1 S profile region width-2 data[SW2]	15	0x7FFF
8A	PGM-1 PWM output confiture data[PWM]	3	0x00
8B	PGM-1 Slowdown/rear pulse amount data[REAR]	32	0x00000000
8C	No operation[NOP]	16	0x0000
8D	PGM-1 Current speed compare data[SPDCMP]	16	0x0000
8E	No operation[NOP]	32	0x00000000
8F	No operation[NOP]	32	0x00000000

CODE(HEX)	이름 (PGM-1 UP-DATE GROUP 레지스터 WRITE)	W	DEFAULT
90	PGM-1 up-date Range data[URANGE]	16	0xFFFF
91	PGM-1 up-date Start/Stop speed data[USTD]	16	0x0001
92	PGM-1 up-date Object speed data[UOBJ]	16	0x0001
93	PGM-1 up-date Rate-1 data[URATE1]	16	0xFFFF
94	PGM-1 up-date Rate-2 data[URATE2]	16	0xFFFF
95	PGM-1 up-date Rate-3 data[URATE3]	16	0xFFFF
96	PGM-1 up-date Rate change point (URATE1 → URATE2)[URCP1]	16	0xFFFF
97	PGM-1 up-date Rate change point (URATE2 → URATE3)[URCP2]	16	0xFFFF
98	PGM-1 up-date S profile region width-1 data[USW1]	15	0x7FFF
99	PGM-1 up-date S profile region width-2 data[USW2]	15	0x7FFF
9A	No operation[NOP]		
9B	PGM-1 up-date Slowdown/rear pulse amount data[UREAR]	32	0X00000000

CODE(HEX)	이 름 (PGM-1 UP-DATE GROUP 레지스터 WRITE)	W	DEFAULT
9C	No operation[ <b>NOP</b> ]	16	0x0000
9D	PGM-1 Current speed compare data(same with 0x0D) [ <b>SPDCMP</b> ]	16	0x0000
9E	No operation[ <b>NOP</b> ]	32	0X00000000
9F	No operation[ <b>NOP</b> ]	32	0X00000000

CODE(HEX)	이 름 (PGM-2 GROUP 레지스터 WRITE)	W	DEFULT
A0	Plus preset pulse drive[ <b>PPRESETDRV</b> ]	32	
A1	Plus continuous drive[ <b>PCONTDRV</b> ]		
A2	Plus signal search-1 drive[ <b>PSCH1DRV</b> ]		
A3	Plus signal search-2 drive[ <b>PSCH2DRV</b> ]		
A4	Plus original search drive[ <b>PORGDRV</b> 0 bit : ORG signal detection edge configure '0'(Falling edge detect), '1'(Rising edge detect)	1	
A5	Minus preset pulse drive[ <b>MPRESETDRV</b> ]	32	
A6	Minus continuous drive[ <b>MCONTDRV</b> ]		
A7	Minus signal search-1 drive[ <b>MSCH1DRV</b> ]		
A8	Minus signal search-2 drive[ <b>MSCH2DRV</b> ]		
A9	Minus original search drive[ <b>MORGDRV</b> 0 bit : ORG signal detection edge configure '0'(Falling edge detect), '1'(Rising edge detect)	1	
AA	Preset/Mpg drive pulse amount override[ <b>PULSEOVER</b> ]	32	
AB	Slow down stop[ <b>SSTOPCMD</b> ]		
AC	Emergency stop[ <b>ESTOPCMD</b> ]		
AD	Drive mode configure data[ <b>DRIVEMODE</b> 2bit : Source selectioin for deceleration point search '0'(Use internal count), '1'(Use external count) 01~01 bit : Profile selection "00" : Symetric Trapezoid "01" : Unsymetric Trapezoid "10" : Symetric S curve "11" : Unsymetric S curve	3	0x0
AE	Mpg operation configure data[ <b>MPGCON</b> 6bit : Continuouse drive start by EXPP, EXMP; '0'(Reset), '1'(Set) 5bit : Preset drive start by EXPP, EXMP; '0'(Reset), '1'(Set) 03~04 bit : Quadrature EXPP, EXMP signal mode	7	0x00

CODE(HEX)	이 름 (PGM-2 GROUP 레지스터 WRITE)	W	DEFULT
	“00” : 1 phase pulse mode “01” : 2 phase pulse mode(One time) “10” : 2 phase pulse mode(Two times) “11” : 2 phase pulse mode(Four times) 2 bit : Mpg drive direction use configure ‘0’(Plus direction), ‘1’(Minus direction) 1 bit : Mpg drive direction source configure ‘0’(EXPP, EXMP signal dir.), ‘1’(User configure dir.) 0bit : Slave mpg drive start by EXPP, EXMP ‘0’(Reset), ‘1’(Set)		
AF	Mpg preset drive pulse amount[PULSEMPG]	32	0x00000000

CODE(HEX)	이 름 ( 확장 GROUP 레지스터 WHITE )	W	DEFULT
B0	Plus sensor positioning drive I [PSPO1DRV]	32	
B1	Minus sensor positioning drive I[MSPO1DRV]	32	
B2	Plus sensor positioning drive II [PSPO2DRV]	32	
B3	Minus sensor positioning drive II [MSPO2DRV]	32	
B4	Plus sensor positioning drive III [PSPO3DRV]	32	
B5	Minus sensor positioning drive III [MSPO3DRV]	32	
B6	Software limit configure data[SWLMTCON] 2 bit : Software limit compare source selection ‘0’(Internal count), ‘1’(External count) 1 bit : Software limit execution mode ‘0’(Emergency stop), ‘1’(Slow down stop) 0 bit : Soft limit Enable/Disable ‘0’(Disable), ‘1’(Enable)	3	0x0
B7	Minus software limit compare data[MSWLMTCOMP]	32	0x80000000
B8	Plus software limit compare data[PSWLMTCOMP]	32	0x7FFFFFFF
B9	Trigger configure data[TRGCON] 31~16 bit : Trigger active pulse width(unsigned) Width = $T_{clk} \times$ (Decimal value of trigger active pulse width) If Width = 0 then Initialize Trigger function. 15~ 02 bit : Don't care. 1 bit : Trigger compare source selection ‘0’(Internal), ‘1’(External)	32	0x00000000

CODE(HEX)	이 름 ( 확장 GROUP 레지스터 WHITE )	W	DEFULT
	0 bit : Trigger mode '0'(Absolute), '1'(Periodic)		
BA	Trigger compare data[TRGCOMP]	32	0x00000000
BB	Internal count minus limit data[ICM]	32	0x80000000
BC	External count minus limit data[ECM]	32	0x80000000
BD~BF	No operation[NOP]		

CODE(HEX)	이 름 ( 스트립터 GROUP 레지스터 WRITE )	W	DEFULT
C0	Script control register-1[SCRCON1] 31 bit : '0'(One time execution, '1' : Always) 30~26bit : Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign 7~ 0 bit : Execution command (write command only) *See event list table for event assign	32	0x00000000
C1	Script control register-2[SCRCON2] (Same with script control register-1)	32	0x00000000
C2	Script control register-3[SCRCON3] (Same with script control register-1)	32	0x00000000
C3	Script control queue[SCRCONQ] (Queue bottom contents) 31 bit : '0'(One time execution, '1' : Always) 30 bit : Interrupt generation when executed '0'(Disable), '1'(Enable) 29~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation	32	0x00000000

CODE(HEX)	이름 (스트립터 GROUP 레지스터 WRITE)	W	DEFULT
	23~ 16 bit : Second event assign 15~ 8 bit : First event assign * When "0xFF", No check event, just execute command 7~ 0 bit : Execution command (write command only)		
C4	Script data register-1[ <b>SCRDATA1</b> ]	32	0x00000000
C5	Script data register-2[ <b>SCRDATA1</b> ]	32	0x00000000
C6	Script data register-3[ <b>SCRDATA1</b> ]	32	0x00000000
C7	Script data queue (Queue bottom contents) [ <b>SCRDATAQ</b> ]	32	0x00000000
C8	Script control/data queue clear[ <b>SCRQCLR</b> ]		
C9~CB	No operation[ <b>NOP</b> ]		
CC	Script queue size configure data (1~13) [ <b>SCRQSIZECON</b> ] 15~ 12 bit : Script control queue full size 11~ 08 bit : Script control queue empty size 07~ 04 bit : Script data queue full size 03~ 00 bit : Script data queue empty size	16	0xD0D0
CD~CF	No operation[ <b>NOP</b> ]		

CODE(HEX)	이름 (갈무리 GROUP 레지스터 WRITE)	W	DEFAULT
D0	Caption control register-1[ <b>CAPCON1</b> ] 31 bit : '0'(One time execution, '1' : Always) 30~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign 7~ 0 bit : Execution command (read command only) *See event list table for event assign	32	0x00000000
D1	Caption control register-2[ <b>CAPCON2</b> ] (Same with script control register-1)	32	0x00000000
D2	Caption control register-3[ <b>CAPCON3</b> ] (Same with script control register-1)	32	0x00000000
D3	Caption control queue[ <b>CAPCONQ</b> ]	32	0x00000000

CODE(HEX)	이 름 (갈무리 GROUP 레지스터 WRITE)	W	DEFAULT
	31 bit : '0'(One time execution, '1' : Always) 30 bit : Interrupt generation when executed '0'(Disable), '1'(Enable) 29~26bit: Don't care 25~24bit : Event operation between first and second event "00" : None, Use first event source only "01" : OR operation "10" : AND operation "11" : XOR operation 23~16bit : Second event assign 15~ 8 bit : First event assign * When "0xFF", No check event, just execute command 7~ 0 bit : Execution command (read command only) *See event list table for event assign		
D4~D7	No operation[ <b>NOP</b> ]		
D8	Caption control queue clear[ <b>CAPQCLR</b> ]		
D9~DB	No operation[ <b>NOP</b> ]		
DC	Caption queue size configure data (0~13) [ <b>CAPQSIZECON</b> ] 15~ 12 bit : Caption control queue full size 11~ 08 bit : Caption control queue empty size 07~ 04 bit : Caption result data queue full size 03~ 00 bit : Caption result data queue empty size	16	0xD0D0
DD~DF	No operation[ <b>NOP</b> ]		

CODE(HEX)	이 름 (BUS-1 GROUP 레지스터 WRITE)	W	DEFAULT
E0	Internal count data(Signed)[ <b>INCNT</b> ]	32	0x00000000
E1	Internal count compare data(Signed)[ <b>INCNTCMP</b> ]	32	0x00000000
E2	Internal count scale data(unsigned) [ <b>INCNTSCALE</b> ]	8	0x00
E3	Internal count plus limit data[ <b>ICP</b> ](Use 0x3B for minus limit)	32	0x7FFFFFFF
E4	External count data(Signed)[ <b>EXCNT</b> ]	32	0x00000000
E5	External count compare data(Signed)[ <b>EXCNTCMP</b> ]	32	0x00000000
E6	External count scale data(unsigned)[ <b>EXCNTSCALE</b> ]	8	0x00
E7	External count plus limit data[ <b>ECP</b> ](Use 0x3C for minus limit)	32	0x7FFFFFFF
E8	External speed data[ <b>EXSPD</b> ]	32	0x00000000
E9	External speed compare data[ <b>EXSPDCMP</b> ]	32	0x00000000

CODE(HEX)	이름 (BUS-1 GROUP 레지스터 WRITE)	W	DEFAULT
EA	External signal filter depth data[ <b>EXFILTERD</b> ] 7~00 bit : Digital Filter for input signals	8	0x05
EB	Off region(over run or under run) decision data[ <b>OFFREGION</b> ]	8	0x00
EC	Deviation data(INCNT – EXCNT)[ <b>DEVIATION</b> ] 15 bit : Result sign '0'(Plus), '1'(Minus) 14~00 bit : Absolute value of deviation	16	0x0000
ED	PGM register bank selection configure data[ <b>PGMCH</b> ] '0'(PGM-1 bank use), '1'(PGM-1 update bank use)	1	0x0
EE	Comparator source selection configure[ <b>COMPCON</b> ] "00" : 1'st : Internal count, 2'nd : External count "01" : 1'st : Internal count, 2'nd : Internal count "10" : 1'st : External count, 2'nd : External count	2	0x0
EF	No operation[ <b>NOP</b> ]		

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
F0	Function configure data [ <b>FUNCON</b> ] Each bit : '0'(Reset, Disable), '1'(Set, Enable) 12bit : S profile triangle drive configure 11bit : Search drive source filter configure 10bit : SYNC input signal function configure 9 bit : Limit complete stop configure 8 bit : INPOS input signal expanded function configure 7bit : Off region stop mode '0'(Emergency stop), '1'(Slowdown stop) 6 bit : Drive at start/stop speed '0'(Stop at <b>STD</b> speed), '1'(Continuous drive at <b>STD</b> speed) 5 bit : NSLM/PSLM input signal function configure 4 bit : NELM/PELM input signal function configure 3 bit : ESTOP/SSTOP signal function configure 2 bit : Don't care 1 bit : ALARM input signal function configure 0 bit : INPOS input signal basic function configure	13	0x0C3E
F1	Mode1 data [ <b>MODE1</b> ] 7 bit : Preset drive slowdown point decision method '0'(Auto), '1'(Manual)	8	0x000

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
	06~04 bit : PULSE/DIR output signal configure(* See table) 03~00 bit : Signal search drive source configure(* See table)		
F2	Mode2 data[MODE2] Active level : '0'(Active low), '1'(Active high) 10 bit : TRIG output signal active level 9 bit : INT output signal active level 8 bit : MARK input signal active level 07~06 bit : Encoder signal count mode(ECUP/ECDN input signal) "00" : Single phase, (ECUP : up count, ECDN : down count) "01" : Two phases, 1 times multiplied "10" : Two phases,, 2 times multiplied "11" : Two phases,, 4 times multiplied 5 bit : INPOS input signal active level 4 bit : ALARM input signal active level 3 bit : NSLM input signal active level 2 bit : PSLM input signal active level 1 bit : NELM input signal active level 0 bit : PELM input signal active level	11	0x200
F3	Universal In/Out signal data[UIODATA] 10~08 bit : Universal output bit operation "0xx" : Operand bypass "100" : NOT Current output "101" : Output operand AND Current output "110" : Output operand OR Current output "111" : Output operand XOR Current output 07~04 bit : Don't care 03~00 bit : Universal output operand	11	0x000
F4~F6	No operation[NOOP]		
F7	External count clear configure data[EXCNTCLR] 8 bit : External count clear flag(clear when write 0xF7) 7 bit : External count mode '0' : Normal count, '1': Reverse count 6 bit : Don't care 05~04 bit : Clear operation mode "00" : Clear request reset	9	0x000

CODE(HEX)	이름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
	“01” : One time clear request set “10” : Full time clear request set “11” : Clear external count and LOCK to zero at clear input signal activated 03~00 bit : Clear input signal assign “0001” : IN0 “0010” : IN1 “0011” : IN2 “0100” : IN3 Others : Don't care(NO clear)		
F8	Software reset[ <b>SWRESET</b> ]		
F9	Interrupt generation command[ <b>INTGENCMD</b> ]		
FA	Interrupt mask data[ <b>INTMASK</b> ] Each bit : ‘0’(Interrupt inactivated), ‘1’( Interrupt activated)	32	0x00000001
FB	EMODE1 data [ <b>EMODE1</b> ] 7 bit : Extension mode enable(0 : Disable, 1 : Enable) 0 : md_out[12:8] => md_out[12:8] 1 : md_out[12:8] => EUIO 6~ 5 bit : Don't care 4 bit : EUO4 enable/disable(0 : disable, 1 : enable) 3 bit : EUO3 enable/disable(0 : disable, 1 : enable) 2 bit : EUO2 enable/disable(0 : disable, 1 : enable) 1 bit : EUO1 enable/disable(0 : disable, 1 : enable) 0 bit : EUIO0 direction select(0 : input, 1 : output)	8	0x00
FC	Extension universal in/out[ <b>EUIODATA</b> ] 07~05 bit : Extension Universal output bit operation “0xx” : Operand bypass “100” : NOT Current output “101” : Output operand AND Current output “110” : Output operand OR Current output “111” : Output operand XOR Current output 4 ~ 0 bit : Extension universal output operand	8	0x00
FD	Limit complete stop clear[ <b>CLIMCLR</b> ]		
FE	User Interrupt source configure data[ <b>INTUSERCON</b> ] Same with Script/Caption event list(except “0xFF” event)	32	0x00000000

CODE(HEX)	이 름 (BUS-2 GROUP 레지스터 WRITE)	W	DEFAULT
	31~24 bit : User selectable interrupt 3 source selection 23~16 bit : User selectable interrupt 2 source selection 15~ 08 bit : User selectable interrupt 1 source selection 07 ~00 bit : User selectable interrupt 0 source selection		
FF	No operation[ <b>NOP</b> ]		

## ► CAMC-FS REGISTER MAP(Alphabetical)

NAME	Description	W	READ	WRITE
CAPCON1	Caption control register-1	32	0x50	0xD0
CAPCON2	Caption control register-2	32	0x51	0xD1
CAPCON3	Caption control register-3	32	0x52	0xD2
CAPCONQ	Caption control queue	32	0x53	0xD3
CAPCQSIZE	Caption control queue index	4	0x59	-
CAPDATA1	Caption control register-1 execution result	32	0x54	-
CAPDATA2	Caption control register-2 execution result	32	0x55	-
CAPDATA3	Caption control register-3 execution result	32	0x56	-
CAPDATAQ	Caption control queue execution result data queue	32	0x57	-
CAPDQSIZE	Caption result data queue index	4	0x5A	-
CAQSIZECON	Caption queue size configure data (0~13)	16	0x5C	0xDC
CAQSLAG	Caption queue Full/Empty flag	7	0x5B	-
CAQSTATUS	Caption queue status	12	0x5D	-
COMPCON	Comparator source selection configure	2	0x6E	0xEE
DEVIATION	Deviation data(INCNT – EXCNT)	16	0x6C	0xEC
DRIVEMODE	Drive mode configure data	3	0x2D	0xAD
DRVPULSE	PGM-1 Drive pulse amount	32	0x0E 0x1E	-
DRVSTATUS	Drive status	9	0x76	-
ECM	External count minus limit data	32	0x3C	0xBC
ECP	External count plus limit data	32	0x67	0xE7
EMODE1	EMODE1 data	8	0x7B	0xFB
ENDSTATUS	End status	15	0x74	-
EUIODATA	Extension universal in/out	8	0x7C	0xFC
EXCNT	External count data(Signed)	32	0x64	0xE4
EXCNTCLR	External count clear configure data	9	0x77	0xF7
EXCNTCMP	External count compare data(Signed)	32	0x65	0xE5
EXCNTSCALE	External count scale data(unsigned)	8	0x66	0xE6
EXFILTERD	External signal filter depth data	8	0x6A	0xEA
EXSPD	External speed data	32	0x68	-
EXSPDCMP	External speed compare data	32	0x69	0xE9
FUNCON	Function configure data	13	0x70	0xF0
ICM	Internal count minus limit data	32	0x3B	0xBB
ICP	Internal count plus limit data	32	0x63	0xE3
INCNT	Internal count data(Signed)	32	0x60	0xE0
INCNTCMP	Internal count compare data(Signed)	32	0x61	0xE1
INCNTSCALE	Internal count scale data(unsigned)	8	0x62	0xE2
INTFALG	Interrupt flag data	32	0x79	-
INTMASK	Interrupt mask data	32	0x7A	0xFA
INTUSERCON	User Interrupt source configure data	32	0x7E	0xFE
MECH	Mechanical data	13	0x75	-
MODE1	Mode1 data	8	0x71	0xF1
MODE2	Mode2 data	11	0x72	0xF2
MPGCON	Mpg operation configure data	7	0x2E	0xAE
MSWLMTCOMP	Minus software limit compare data	32	0x37	0xB7
OBJ	PGM-1 Object speed	16	0x02	0x82
OFFREGION	Off region(over run or under run) decision data	8	0x6B	0xEB
PGMCH	PGM register bank selection configure data	1	0x6D	0xED
PRESETPULSE	PGM-1 Preset pulse amount	32	0x0F 0x1F	-
PSWLMTCOMP	Plus software limit compare data	32	0x38	0xB8
PULSEMPG	Mpg preset drive pulse amount	32	0x2F	0xAF
PWM	PGM-1 PWM output configure	3	0x0A	0x8A
RANGE	PGM-1 Range	16	0x00	0x80

NAME	Description	W	READ	WRITE
RATE1	PGM-1 Rate-1	16	0x03	0x83
RATE2	PGM-1 Rate-2	16	0x04	0x84
RATE3	PGM-1 Rate-3	16	0x05	0x85
RCP1	PGM-1 Rate change point(RATE1→RATE2)	16	0x06	0x86
RCP2	PGM-1 Rate change point(RATE2→RATE3)	16	0x07	0x87
REAR	PGM-1 Slowdown/rear pulse amount	32	0x0B	0x8B
SCRCON1	Script control register-1	32	0x40	0xC0
SCRCON2	Script control register-2	32	0x41	0xC1
SCRCON3	Script control register-3	32	0x42	0xC2
SCRCONQ	Script control queue	32	0x43	0xC3
SCRQSIZE	Script control queue index(max to 0xD)	4	0x49	-
SCRDATA1	Script data register-1	32	0x44	0xC4
SCRDATA1	Script data register-2	32	0x45	0xC5
SCRDATA1	Script data register-3	32	0x46	0xC6
SCRDATAQ	Script data queue (Queue bottom contents)	32	0x47	0xC7
SCRDQSIZE	Script data queue index(max to 0xD)	4	0x4A	-
SCRQSIZECON	Script queue size configure data (1~13)	16	0x4C	0xCC
SCRQFLAG	Script queue Full/Empty flag	4	0x4B	-
SCRQSTATUS	Script queue status	12	0x4D	-
SPD	PGM-1 Current speed	16	0x0C 0x1C	-
SPDCMP	PGM-1 Current speed compare	16	0x0D 0x1D	0x8D
STD	PGM-1 Start/Stop speed	16	0x01	0x81
SW1	PGM-1 S profile resion width-1	15	0x08	0x88
SW2	PGM-1 S profile resion width-2	15	0x09	0x89
SWLMTCON	Software limit configure data	3	0x36	0xB6
TRGCOMP	Trigger compare data	32	0x3A	0xBA
TRGCON	Trigger configure data	32	0x39	0xB9
UIODATA	Universal In/Out signal data	11	0x73	0xF3
UOBJ	PGM-1 UP-DATE Object speed data	16	0x12	0x92
URANGE	PGM-1 UP-DATE Range data	16	0x10	0x90
URATE1	PGM-1 UP-DATE Rate-1 data	16	0x13	0x93
URATE2	PGM-1 UP-DATE Rate-2 data	16	0x14	0x94
URATE3	PGM-1 UP-DATE Rate-3 data	16	0x15	0x95
URCP1	PGM-1 UP-DATE Rate change point(URATE1→URATE2)	16	0x16	0x96
URCP2	PGM-1 UP-DATE Rate change point(URATE2→URATE3)	16	0x17	0x97
UREAR	PGM-1 UP-DATE Slowdown/rear pulse amount data	32	0x1A	0x9A
USTD	PGM-1 UP-DATE Start/Stop speed data	16	0x11	0x91
USW1	PGM-1 UP-DATE S profile region width-1 data	15	0x18	0x98
USW2	PGM-1 UP-DATE S profile region width-2 data	15	0x19	0x99

► **CAMC-FS DRIVE COMMAND/SETTING COMMAND(write only, alphabetical)**

NAME	Description	W	READ	WRITE
<b>CAPQCLR</b>	Caption control queue clear			0xD8
<b>CLIMCLR</b>	Limit complete stop clear			0xFD
<b>ESTOPCMD</b>	Emergency stop		-	0xAC
<b>INTGENCMD</b>	Interrupt generation command			0xF9
<b>MCONTDRV</b>	Minus continuous drive		-	0xA6
<b>MORGDRV</b>	Minus original search drive	1	-	0xA9
<b>MPRESETDRV</b>	Minus preset pulse drive	32	-	0xA5
<b>MSCH1DRV</b>	Minus signal search-1 drive		-	0xA7
<b>MSCH2DRV</b>	Minus signal search-2 drive		-	0xA8
<b>MSPO1DRV</b>	Minus sensor positioning drive I	32		0xB1
<b>MSPO2DRV</b>	Minus sensor positioning drive II	32		0xB3
<b>MSPO3DRV</b>	Minus sensor positioning drive III	32		0xB5
<b>PCONTDRV</b>	Plus continuous drive		-	0xA1
<b>PORGDRV</b>	Plus original search drive	1	-	0xA4
<b>PPRESETDRV</b>	Plus preset pulse drive	32	-	0xA0
<b>PSCH1DRV</b>	Plus signal search-1 drive		-	0xA2
<b>PSCH2DRV</b>	Plus signal search-2 drive		-	0xA3
<b>PSPO1DRV</b>	Plus sensor positioning drive I	32		0xB0
<b>PSPO2DRV</b>	Plus sensor positioning drive II	32		0xB2
<b>PSPO3DRV</b>	Plus sensor positioning drive III	32		0xB4
<b>PULSEOVER</b>	Preset/Mpg drive pulse amount override	32	-	0xAA
<b>SCRQCLR</b>	Script control/data queue clear			0xC8
<b>SSTOPCMD</b>	Slow down stop		-	0xAB
<b>SWRESET</b>	Software reset			0xF8

### 10.4.COMMAND 기능 부연 설명

▶ 펄스출력방식

비트			방향 신호(DIR)		펄스 신호 ( PULSE )		방 식
D6	D5	D4	CW	CCW			
0	0	0	L	H	Active H Pulse		1 Pulse 방식
0	0	1	H	L			
0	1	0	L	H	Active L Pulse		
0	1	1	H	L			
1	0	0	CW Pulse	Active H	CCW Pulse	Active H	2 Pulse 방식
1	0	1		Active L		Active L	
1	1	0	CCW Pulse	Active H	CW Pulse	Active H	
1	1	1		Active L		Active L	

1 펄스 방식

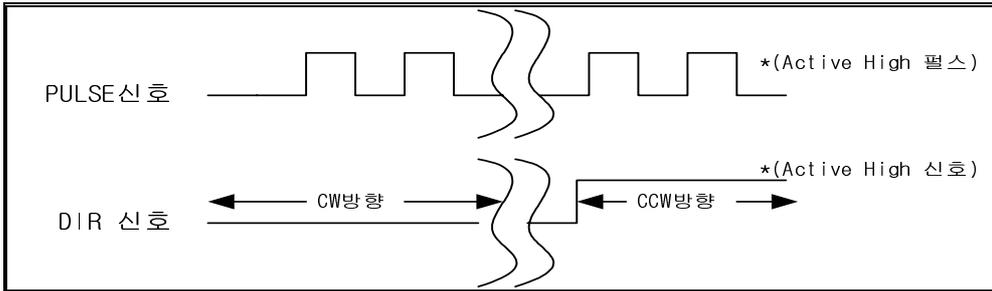


그림 10.1. 1펄스 방식 - 1(D6, D5, D4 → “000”)

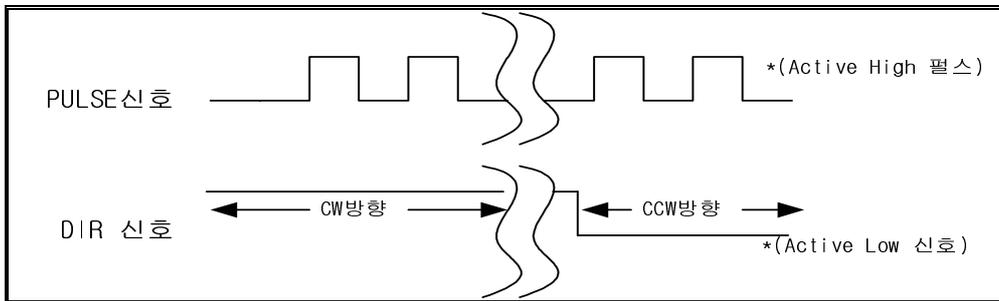


그림 10.2. 1펄스 방식 - 2(D6, D5, D4 → “001”)

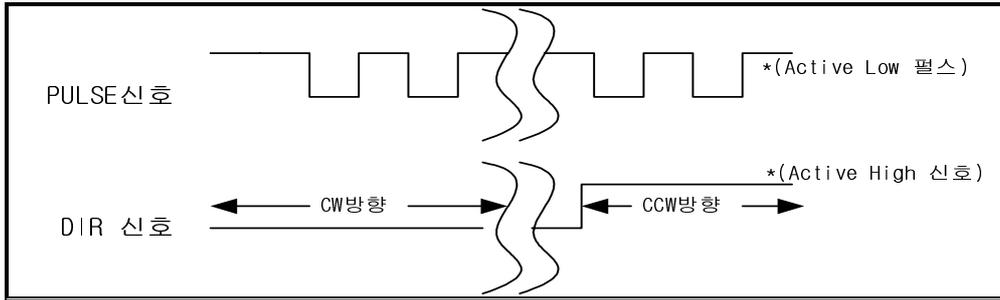


그림 10.3. 1펄스 방식 - 3(D6, D5, D4 → "010")

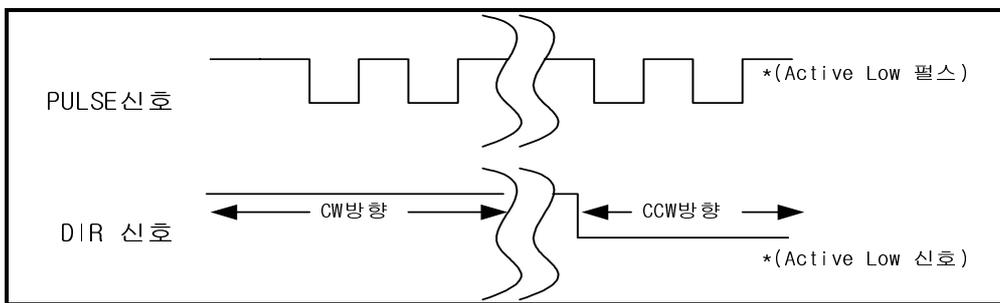


그림 10.4. 1펄스 방식 - 4(D6, D5, D4 → "011")

2 펄스 방식

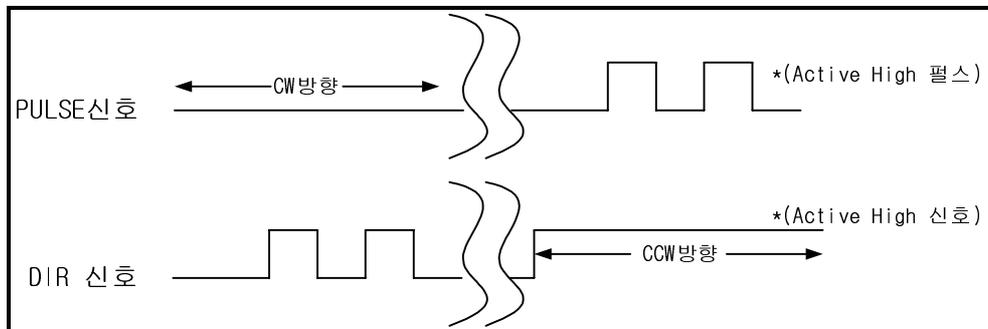


그림 10.5. 2펄스 방식 - 1(D6, D5, D4 → "100")

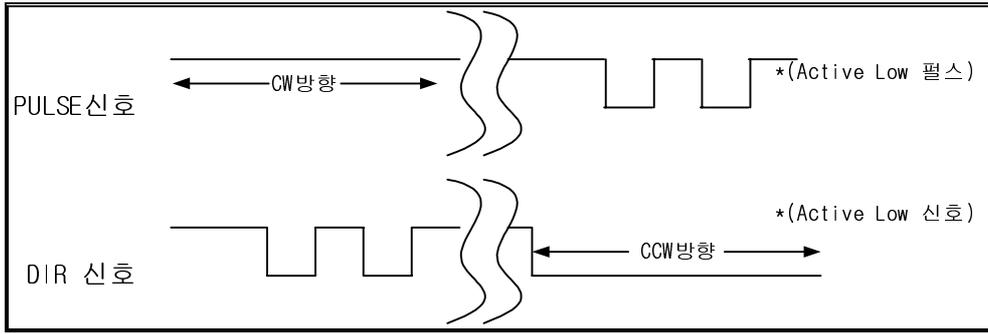


그림 10.6. 2펄스 방식 - 2(D6, D5, D4 → "101")

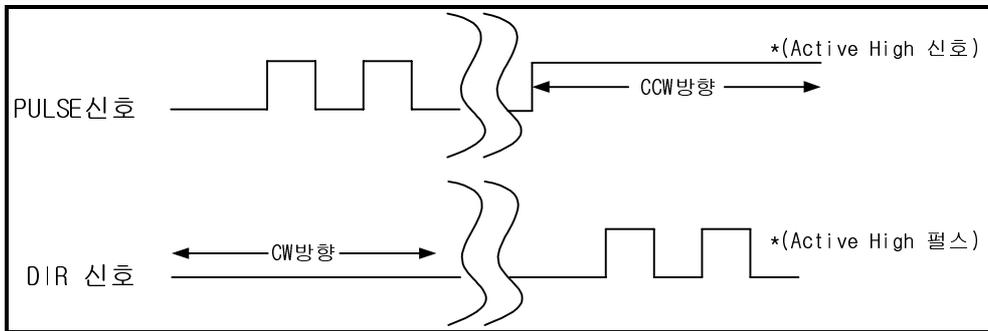


그림 10.7. 2펄스 방식 - 3(D6, D5, D4 → "110")

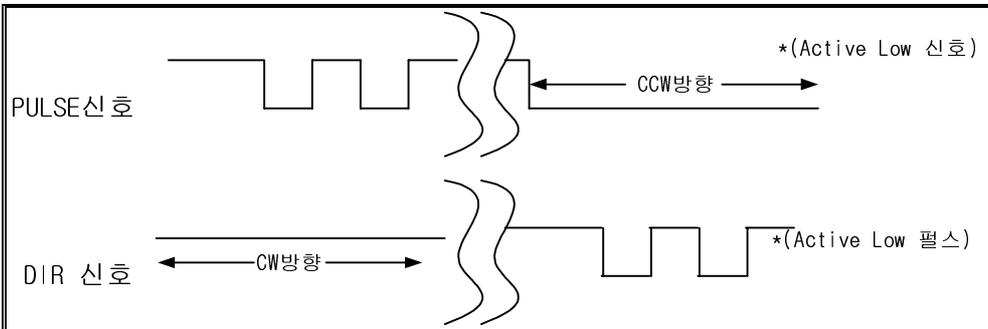


그림 10.8. 2펄스 방식 - 4(D6, D5, D4 → "111")

▶ 검출 대상 신호 설정

D3	D2	D1	D0	검출 신호 및 검출 Edge
0	0	0	0	+ELM 신호의 Negative Edge 검출
0	0	0	1	-ELM 신호의 Negative Edge 검출
0	0	1	0	+SLM 신호의 Negative Edge 검출
0	0	1	1	-SLM 신호의 Negative Edge 검출
0	1	0	0	IN0 신호의 하강 Edge 검출
0	1	0	1	IN1 신호의 하강 Edge 검출

0	1	1	0	IN2 신호의 하강 Edge 검출
0	1	1	1	IN3 신호의 하강 Edge 검출
1	0	0	0	+ELM 신호의 Positive Edge 검출
1	0	0	1	-ELM 신호의 Positive Edge 검출
1	0	1	0	+SLM 신호의 Positive Edge 검출
1	0	1	1	-SLM 신호의 Positive Edge 검출
1	1	0	0	IN0 신호의 상승 Edge 검출
1	1	0	1	IN1 신호의 상승 Edge 검출
1	1	1	0	IN2 신호의 상승 Edge 검출
1	1	1	1	IN3 신호의 상승 Edge 검출

▶ 외부 위치 카운터 입력 사양 ( Mode2 레지스터 상위 비트에서 설정 )

7 bit	6 bit	입력 사양	ECUP 신호	ECDN 신호
0	0	UP/DOWN 신호 입력	UP 입력	DOWN 입력
0	1	2 상 신호 1 체배 입력	$\phi A$ 입력	$\phi B$ 입력
1	0	2 상 신호 2 체배 입력	$\phi A$ 입력	$\phi B$ 입력
1	1	2 상 신호 4 체배 입력	$\phi A$ 입력	$\phi B$ 입력

\* Up/Down 신호 입력(D7, D6 →'00')

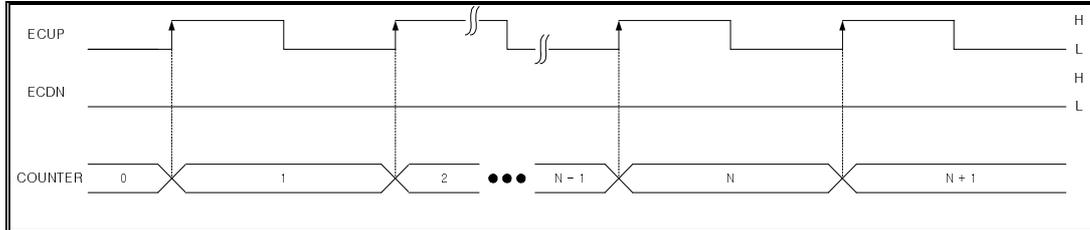


그림 10.9. Up/Down 신호 입력 시 Up Counter 동작

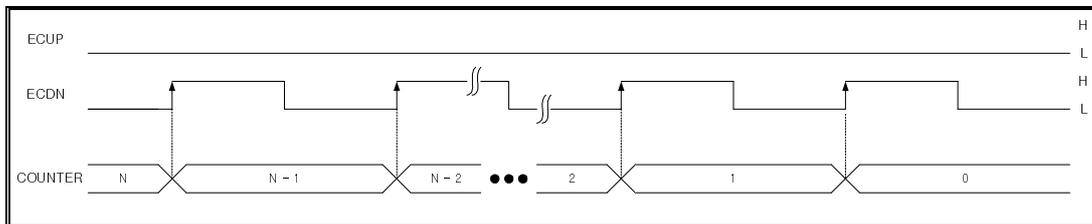


그림 10.10. Up/Down 신호 입력 시 Down Counter 동작

\* 2상 신호 입력 -1체배(D7, D6 →'01')

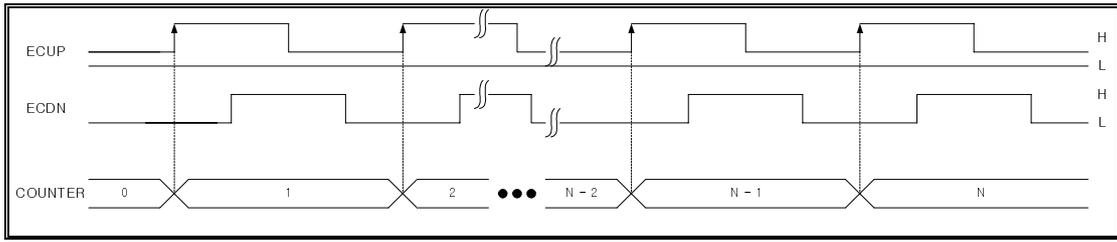


그림 10.11. 2상 신호 입력 시 1체배 Up Counter 동작

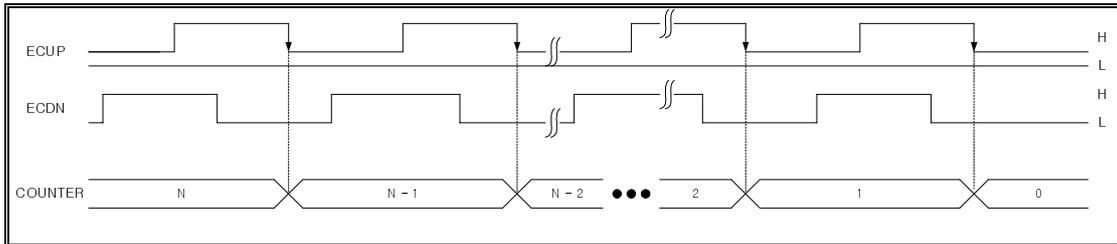


그림 10.12. 2상 신호 입력 시 1체배 Down Counter 동작

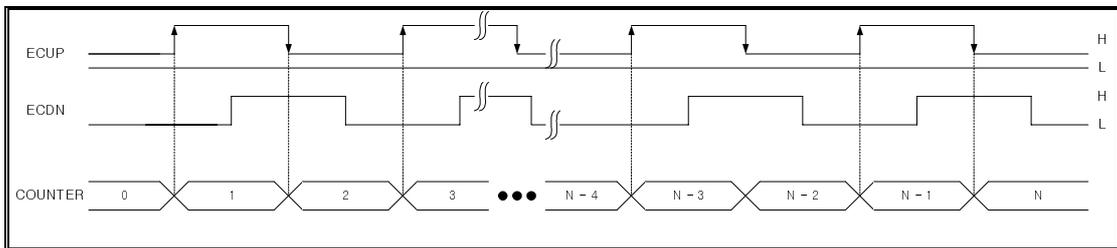


그림 10.13. 2상 신호 입력 시 2체배 Up Counter 동작

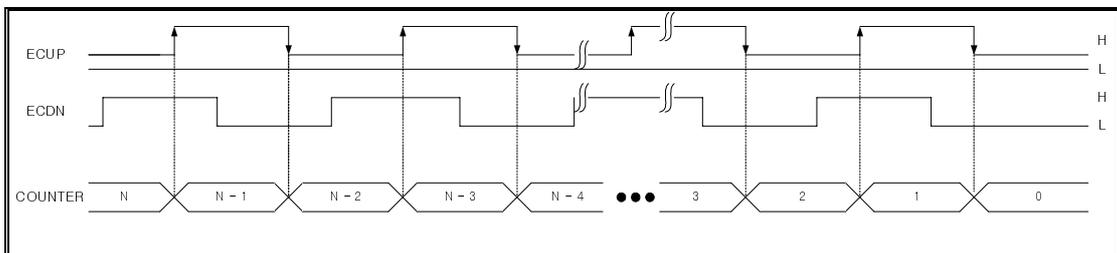


그림 10.14. 2상 신호 입력 시 2체배 Down Counter 동작

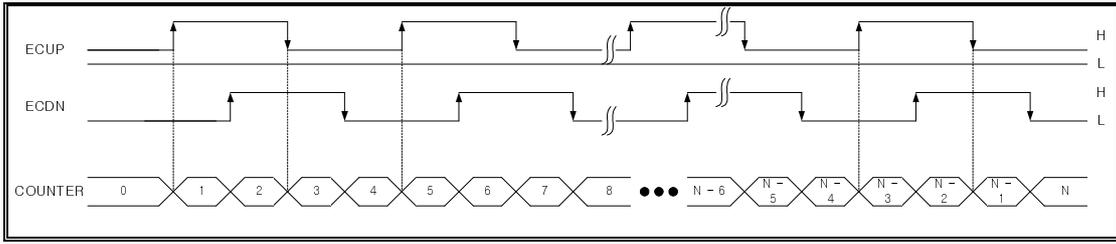


그림 10.15. 2상 신호 입력 시 4채배 Up Counter 동작

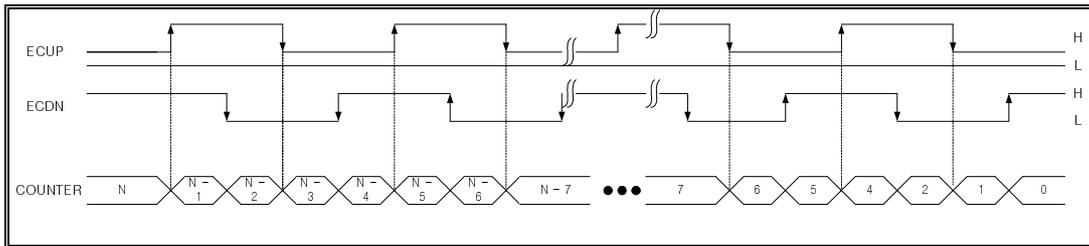


그림 10.16. 2상 신호 입력 시 4채배 Down Counter 동작

▶ \* 스크립트 및 갈무리 기능에 사용될 검출 이벤트

검출할 동작 이벤트	이벤트 동작 비트 설정값
No detect	0x00
Drive end	0x01
Preset dirve start	0x02
Preset dirve end	0x03
Continuous drive start	0x04
Continuous drive end	0x05
Signal search-1 drive start	0x06
Signal search-1 drive end	0x07
Signal search-2 drive start	0x08
Signal search-2 drive end	0x09
Original search drive start	0x0A
Original search drive end	0x0B
Acceleration	0x0C
Constant speed	0x0D
Deceleration	0x0E
ICG(internal count > internal count compare data)	0x0F
ICE(internal count = internal count compare data)	0x10
ICL(internal count < internal count compare data)	0x11
ECG(external count > external count compare data)	0x12
ECE(external count = external count compare data)	0x13
ECL(external count < external count compare data)	0x14
External speed > External speed compare data	0x15
External speed = External speed compare data	0x16
External speed < External speed compare data	0x17
Current speed > Current speed compare data	0x18
Current speed = Current speed compare data	0x19
Current speed < Current speed compare data	0x1A
Current speed > Rate Change Point 1-2	0x1B
Current speed = Rate Change Point 1-2	0x1C
Current speed < Rate Change Point 1-2	0x1D
Current speed > Rate Change Point 2-3	0x1E
Current speed = Rate Change Point 2-3	0x1F

검출할 동작 이벤트	이벤트 동작 비트 설정값
Current speed < Rate Change Point 2-3	0x20
Current speed = Object speed data	0x21
Current speed = Start/Stop speed data	0x22
Emergency stop	0x23
Slowdown stop	0x24
PELM signal in active level	0x25
NELM signal in active level	0x26
PSLM signal in active level	0x27
NSLM signal in active level	0x28
Off resion error occurred	0x29
Data configure error occurred	0x2A
ALARM signal in active level	0x2B
Emergency(0xAC) stop command executed	0x2C
Slowdown(0xAB) stop command executed	0x2D
ESTOP signal in HIGH level	0x2E
SSTOP signal in HIGH level	0x2F
PELM/NELM signal in active level	0x30
PSLM/NSLM signal in active level	0x31
INPOS signal in active level	0x32
IN0 input in HIGH level	0x33
IN0 input in LOW level	0x34
IN1 input in HIGH level	0x35
IN1 input in LOW level	0x36
IN2 input in HIGH level	0x37
IN2 input in LOW level	0x38
IN3 input in HIGH level	0x39
IN3 input in LOW level	0x3A
OUT0 outptu in HIGH level	0x3B
OUT0 outptu in LOW level	0x3C
OUT1 outptu in HIGH level	0x3D
OUT1 outptu in LOW level	0x3E
OUT2 outptu in HIGH level	0x3F
OUT2 outptu in LOW level	0x40
OUT3 outptu in HIGH level	0x41
OUT3 outptu in LOW level	0x42
Sensor Positioning drive I start	0x43
Sensor Positioning drive I end	0x44
Sensor Positioning drive II start	0x45
Sensor Positioning drive II end	0x46
Sensor Positioning drive III start	0x47
Sensor Positioning drive III end	0x48
1'st counter N-data count clear	0x49
2'nd counter N-data count clear	0x4A
MARK signal in HIGH level	0x4B
MARK signal in LOW level	0x4C
EUIO0 signal in HIGH level	0x4D
EUIO0 signal in LOW level	0x4E
EUO1 signal in HIGH level	0x4F
EUO1 signal in LOWlevel	0x50
EUO2 signal in HIGH level	0x51
EUO2 signal in LOWlevel	0x52
EUO3 signal in HIGH level	0x53
EUO3 signal in LOWlevel	0x54
EUO4 signal in HIGH level	0x55
EUO4 signal in LOWlevel	0x56
+ Software LIMIT	0x57
- Software LIMIT	0x58
Software LIMIT	0x59
Trigger enable	0x5A

---

검출할 동작 이벤트	이벤트 동작 비트 설정값
Interrupt generated by any source	0x5B
interrupt generated by command "F9"	0x5C
Preset triangular drive start	0x5D
Drive busy output in HIGH level	0x5E
Drive busy output in LOW level	0x5F
Just execution(in Queue command)	0xFF

이 설명서의 내용은 예고 없이 변경될 수 있습니다. 용례에 사용된 회사, 기관, 제품, 인물 및 사건 등은 실제 데이터가 아닙니다. 어떠한 실제 회사, 기관, 제품, 인물 또는 사건과도 연관시킬 의도가 없으며 그렇게 유추해서도 안됩니다. 해당 저작권법을 준수하는 것은 사용자의 책임입니다. 저작권에서의 권리와는 별도로, 이 설명서의 어떠한 부분도 (주)아진엑스텍의 명시적인 서면 승인 없이는 어떠한 형식이나 수단(전기적, 기계적, 복사기에 의한 복사, 디스크 복사 또는 다른 방법) 또는 다른 목적으로도 복제되거나, 검색 시스템에 저장 또는 도입되거나, 전송될 수 없습니다.

(주)아진엑스텍은 이 설명서 본안에 관련된 특허권, 상표권, 저작권 또는 기타 지적 소유권 등을 보유할 수 있습니다. 서면 사용권 계약에 따라 (주)아진엑스텍으로부터 귀하에게 명시적으로 제공된 권리 이외에, 이 설명서의 제공은 귀하에게 이러한 특허권, 저작권 또는 기타 지적 소유권 등에 대한 어떠한 사용권도 허용하지 않습니다.