

로보스타 로봇  
N1 시리즈 옵션  
로보스타 프로토콜



- |  Option Module
  - Unihost
  - Protocol

ROBOSTAR ROBOT  
N1 Series Option  
ROBOSTAR Protocol



- |  Option Module
  - Unihost
  - Protocol

---

Copyright © ROBOSTAR Co., Ltd 2012

이 사용 설명서의 저작권은 주식회사 로보스타에 있습니다.  
어떠한 부분도 로보스타의 허락 없이 다른 형식이나 수단으로 사용할 수 없습니다.

사양은 예고 없이 변경 될 수 있습니다.

---

## 제품 보증에 관하여

(주) 로보스타의 제품은 엄격한 품질 관리로 제조되고 있으며, 로보스타의 전 제품의 보증 기간은 제조일로부터 1년간입니다. 이 기간 내에 로보스타 측의 과실로 인한 기계의 고장 또는 정상적인 사용 중의 설계 및 제조상의 문제로 발생하는 고장에 한해서만, 무상으로 서비스를 합니다.

다음과 같은 경우에는 무상 서비스가 불가능합니다.

- (1) 보증 기간이 만료된 이후
- (2) 귀사 또는 제 3 자의 지시에 따른 부적당한 수리, 개조, 이동, 기타 취급 부주의로 인한 고장
- (3) 부품 및 그리스 등 당사의 지정품 이외의 것의 사용으로 인한 고장
- (4) 화재, 재해, 지진, 풍수해 기타 천재지변에 의한 사고로 발생하는 고장
- (5) 분료 및 침수 등 당사의 제품 사양 외의 환경에서 사용함으로 인한 고장
- (6) 소모 부품의 소모로 인한 고장
- (7) 사용설명서 및 취급 설명서에 기재된 보수 점검 작업 내용대로 실시하지 않음으로 인해 발생하는 고장
- (8) 로봇 수리에 드는 비용 이외의 손해

### (주) 로보스타 주소 및 연락처

- 본사 및 공장  
경기도 안산시 상록구 수인로 700  
700, Suin-ro, Sangnok-gu,  
Ansan-City, Gyeonggi-do, Republic of  
South Korea (426-220)
- 제 2공장  
경기도 수원시 권선구 산업로 108  
108, Saneop-ro, Gwonseon-gu,  
Suwon-City, Gyeonggi-do, Republic of  
South Korea (441-813)
- 서비스요청 및 제품문의  
- 영업문의  
TEL. 031-400-3600  
FAX. 031-419-4249  
- 고객문의  
TEL. 1588-4428



[www.robostar.co.kr](http://www.robostar.co.kr)

# 사용 설명서의 구성

본 제품에 관한 사용 설명서는 다음과 같이 구성되어 있습니다. 본 제품을 처음 사용하는 경우 모든 설명서를 충분히 숙지하신 후 사용하시기 바랍니다.

- **ROBOSTAR Protocol**

RS-232C통신을 사용하여 N1 제어기의 접속 방법 및 사용법에 대하여 설명합니다.

# 목차

<b>제1장</b>	<b>개요</b> .....	<b>1-1</b>
1.1	시스템 구성 .....	1-1
<b>제2장</b>	<b>기능</b> .....	<b>2-1</b>
2.1	RS-232 통신 .....	2-1
2.1.1	RS-232 통신에 대하여 .....	2-1
2.1.2	D-SUB 9Pin 커넥터 .....	2-1
2.2	ETHERNET(TCP/IP) 통신 .....	2-2
2.2.1	Ethernet 통신에 대하여 .....	2-2
2.2.2	전송 제어 프로토콜(Transmission Control Protocol).....	2-2
<b>제3장</b>	<b>설치 및 동작 설정</b> .....	<b>3-3</b>
3.1	RS-232 통신 .....	3-3
3.1.1	Hardware 설치 방법 .....	3-3
3.1.2	Cable 연결 방법 .....	3-3
3.1.3	Controller 설정 .....	3-4
3.2	ETHERNET 통신 .....	3-6
3.2.1	Hardware 설치 방법 .....	3-6
3.2.2	Cable 연결 방법 .....	3-6
3.2.3	Controller 설정 .....	3-8
<b>제4장</b>	<b>통신 프로토콜</b> .....	<b>4-1</b>
4.1	통신 규칙 .....	4-1
4.2	세부 프로토콜 명령어 .....	4-3
4.3	로봇 상태 정보 읽기(AA) .....	4-5
4.4	현재 ERROR 상태 읽기(AB) .....	4-6
4.5	로봇 현재 위치 좌표 읽기(AC) .....	4-8
4.6	제어기 INFO 읽기(AD).....	4-10
4.7	원점 수행(BA).....	4-12
4.8	지정한 POINT번호 좌표로 로봇 이동(BB).....	4-13
4.9	지정한 데이터로 모션 수행(BC).....	4-15
4.10	현재 위치에서 지정한 DATA만큼 증분이동(BD).....	4-17
4.11	JOG START(BE) .....	4-19
4.12	JOG MOVE CONTINUE(BF) .....	4-21
4.13	JOG MOVE STOP(BG).....	4-22

4.14	로봇 속도 읽기(CA)	4-23
4.15	로봇 속도 쓰기(CB)	4-24
4.16	JOB START(CC)	4-25
4.17	JOB STOP(CD)	4-26
4.18	JOB RESET(CE)	4-27
4.19	온라인 비상정지(CF)	4-28
4.20	ERROR RESET(CG)	4-29
4.21	ORIGIN STOP(CI)	4-30
4.22	제어기 재부팅(CJ)	4-31
4.23	보간 이동 기준 축 변경(CX)	4-32
4.24	보간 이동 기준 축 읽기(CZ)	4-33
4.25	SERVO ON/OFF(DB)	4-34
4.26	실행 할 JOB 설정(DC)	4-35
4.27	JOB의 AUTO/STEP MODE 설정(EA)	4-37
4.28	현재 실행 중인 JOB STEP번호 읽기(ED)	4-38
4.29	현재 실행 중인 JOB이름 읽기(EF)	4-39
4.30	제어기에서 PC로 파일 전송(FA)	4-40
4.31	파일 저장(FB)	4-44
4.32	파일 검색(FC)	4-46
4.33	파일 정보 요청(FD)	4-47
4.34	파일 삭제(FE)	4-50
4.35	파일 복사(FF)	4-51
4.36	파일 이름 변경(FG)	4-53
4.37	저장 되어 있는 알람 내용 읽기(FH)	4-55
4.38	I/O 카드 INPUT 접점 상태 정보 읽기(GA)	4-57
4.39	I/O 카드 OUT PUT 접점 상태 정보 읽기(GB)	4-59
4.40	I/O 카드 OUT PUT 접점 출력(GC)	4-61
4.41	I/O 카드 정보 읽기 (GD)	4-62
4.42	글로벌 변수 DATA 읽기(GR)	4-63
4.42.1	글로벌 포인트 읽기	4-63
4.42.2	글로벌 정수 읽기	4-65
4.42.3	글로벌 실수 읽기	4-66
4.43	글로벌 변수 DATA 쓰기(GW)	4-67
4.43.1	글로벌 포인트 저장	4-67
4.43.2	글로벌 정수 저장	4-69
4.43.3	글로벌 실수 저장	4-70
4.44	통신 에러 원인 읽기(KD)	4-71

4.45 시스템 DATA 읽기(KE) .....4-72

4.46 VISION POINT 저장(VA) .....4-75

4.47 대상물의 양/불 판정값 전송(VB) .....4-77

4.48 VISION POINT DATA 연속 저장(VM) .....4-78

4.49 VISION POINT 증분 DATA 저장(VN) .....4-80

4.50 MOTOR BRAKE ON/OFF(DD).....4-82

4.51 통신 프로토콜 에러 처리 .....4-84

    4.51.1 정상적인 통신 절차 ..... 4-84

    4.51.2 PC to Controller 통신 에러 ..... 4-84

    4.51.3 Controller to PC 통신 에러 ..... 4-88

## 제1장 개요

### 1.1 시스템 구성

N1 제어기는 Robostar 프로토콜을 사용하여 PC와 같은 시스템과 통신을 수행할 수 있습니다. N1 제어기 Main Board의 Host Port와 Ethernet Port 이용하여 Robostar 프로토콜 통신이 가능합니다.

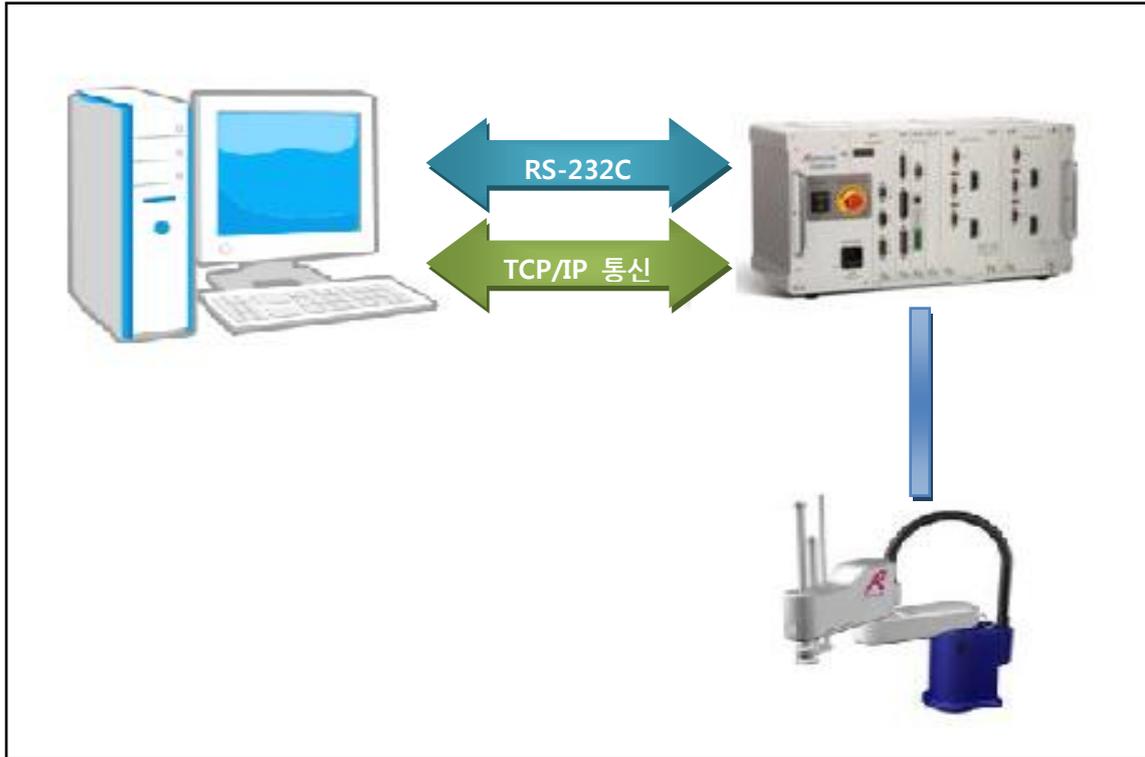


그림 1.1 시스템 구성도



#### CAUTION

- ▶ Ethernet 통신의 경우 MAIN Board V3.0A 부터 지원 합니다.

## 제2장 기능

### 2.1 RS-232 통신

#### 2.1.1 RS-232 통신에 대하여

RS232C 통신 거리는 일반적으로 15m 이지만, 통신 속도가 높을수록 거리가 짧아 집니다.

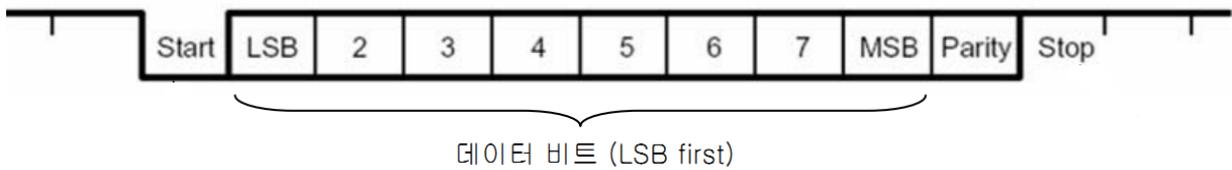
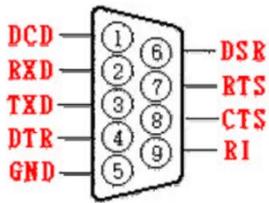


그림 2.1 비동기 통신 Format

- Start 비트 : 통신 Format 시작을 나타냄
- DATA 비트 : 5/6/7/8 비트 크기로 설정이 가능하며, LSB first 로 통신 데이터 값을 나타냄
- Parity 비트 : No/Even/Odd 로 설정이 가능하고, 노이즈로 1 비트가 변경되었을 때 검출이 가능
- Stop 비트: 1/1.5/2 크기로 설정이 가능하고, 데이터의 끝을 나타냄

#### 2.1.2 D-SUB 9Pin 커넥터



	설명
TXD	Transmit Data, 통신 데이터 출력 신호
RXD	Receive Data, 통신 데이터 입력 신호
RTS	Ready To Send 모뎀 통신 등에 사용하며 통신 준비 상태를 표시하며, 범용 출력 포트 사용 가능
CTS	Clear To Send 모뎀 통신 등에 사용하며 통신 준비 상태를 표시하며, 범용 입력 포트 사용가능
DTR	Data Terminal Ready 모뎀 통신 준비 신호로, 출력 포트 사용가능
DSR	Data Set Ready 모뎀 통신 준비 신호로 입력 포트 사용 가능
DCD	Data Carrier Detect, 입력 포트
RI	Ring Indicator 입력 포트
GND	그라운드

표 2.1 커넥터 설명

## 2.2 Ethernet(TCP/IP) 통신 (MAIN B/D V3.0A 부터 지원)

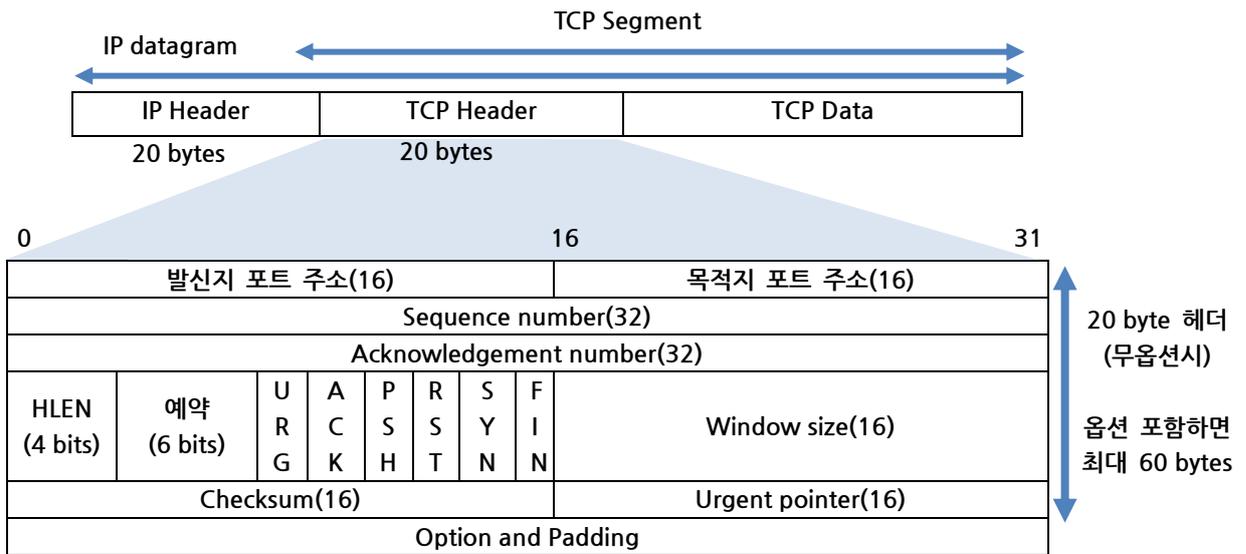
### 2.2.1 Ethernet 통신에 대하여

이더넷(영어: Ethernet)은 컴퓨터 네트워크 기술의 하나로, 전세계의 사무실이나 가정에서 일반적으로 사용되는 LAN에서 가장 많이 활용되는 기술 규격입니다.

이더넷은 OSI 모델의 물리 계층에서 신호와 배선, 데이터 링크 계층에서 MAC(media access control) 패킷과 프로토콜의 형식을 정의하며, 대부분 IEEE 802.3 규약으로 표준화되었습니다.

### 2.2.2 전송 제어 프로토콜(Transmission Control Protocol)

인터넷 프로토콜 스위트(IP)의 핵심 프로토콜 중 하나로, IP와 함께 TCP/IP라는 명칭으로도 널리 불립니다. TCP는 근거리 통신망이나 인트라넷, 인터넷에 연결된 컴퓨터에서 실행되는 프로그램 간에 일련의 옥텟을 안정적으로, 순서대로, 어려없이 교환할 수 있게 합니다. TCP 프로토콜은 전송 계층에 해당합니다.



**CAUTION**

▶ Ethernet 통신의 경우 MAIN Board V3.0A 부터 지원 합니다.

## 제3장 설치 및 동작 설정

### 3.1 RS-232 통신

#### 3.1.1 Hardware 설치 방법

다음과 같은 과정을 거쳐 제어기의 프로토콜을 사용할 수 있습니다.

- 1) 제어기 Main Board 통신포트(HOST PORT)에 Serial 케이블을 연결합니다.

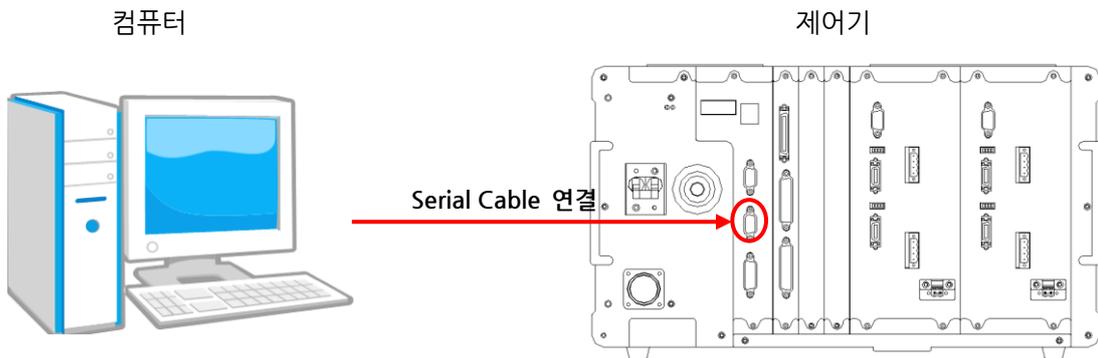


그림 3.1 Serial 케이블 연결방법

#### 3.1.2 Cable 연결 방법

케이블 연결 방법으로 “그림 3.2 케이블 결선도”를 참조하시기 바랍니다.

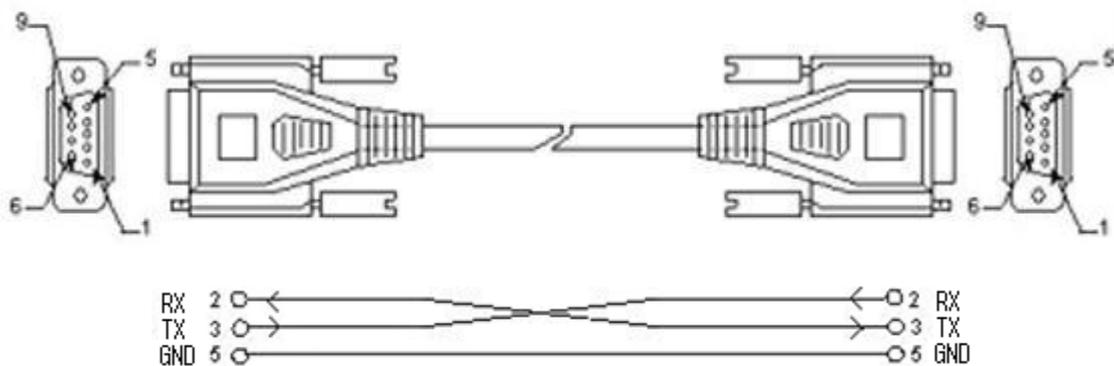


그림 3.2 케이블 결선도

핀 번호	이름	내용	기능 설명
2	RxD	Received Data	수신 데이터 신호
3	TxD	Transmitted Data	송신 데이터 신호
5	GND	Ground	그라운드

표3.1 결선핀 번호 설명

### 3.1.3 Controller 설정

제어기에서 Robostar 프로토콜을 사용하기 위해서는 RS-232C 통신 설정을 하셔야 합니다.  
Baud Rate 기본 설정값은 115,200bps로 되어있습니다.

#### ■ RS-232C 통신 설정

##### 1. 설정순서

##### Step 1.

##### PARA 화면 이동

```
<MAIN MENU>
1. JOB          2. RUN
3. HOST         4. PARA
5. ORIGIN      6. I/O
7. SYSTEM      8. GPNT
9. INT/FLT     A. ALARM

SELECT #
```

초기 메뉴 화면 OPEN  
4: PARA 선택

4  
L

```
<PARAMETER>
NO          TYPE
*CH1       XYZW
CH2        XY_TEST

SEL  INFO  PUB  EXIT
```

PUBLIC PARAMETER 그룹화면 OPEN

F3

```
<PUBLIC PARAMETER>
1: HW CONF  2: PALLET
3: PLC       4: ETC

group #
```

1:HW CONF 선택

1  
Q

##### Step 2.

##### COMM 화면 이동

```
<PUBLIC-HW CONF(0)>
1: TMR          2: COMM
3: I/O         4: D-MAN
5: SVON        6: A I/O

item #
```

2: COMM 선택

2  
R

```
<HW CONF - COM>
COMMUNICATION SET
1: MAIN COMM
2: FIELD BUS

group #
```

1: MAIN COMM 선택

1  
Q

```

<HW CONF - COMM>
  MAIN COMM SET
  1: SERIAL
  2: ETHERNET

group #
    
```

1:SERIAL 선택



Step 3.

RS232 설정 화면

```

<COM - PORT SET 1/2>
  PORT:HOST
  PROTOCOL:  N1
  BAUD RATE: 230400
  SEPARATOR: CR+LF
  ID:         0

  COM1  COM2  RS485
    
```

RS232C 설정 화면



```

<COM - PORT SET 2/2>
  PORT:HOST
  LENGTH:    8 bits
  STOP BIT:  1 bits
  PARITY:    disable
  FLOW:      NONE

  COM1  COM2  RS485
    
```

RS232C 설정 화면

■ 컴퓨터와 RS-232C 통신을 위한 규격

항 목	설정값	초기값
PROTOCOL	N1	N1
BAUD RATE	115,200	115,200
LENGTH	8	8
STOP BIT	1	1
PARITY	disable	disable
FLOW	NONE	NONE
ID	0	0

※ ID는 현재 기능 지원이 안됩니다.

### 3.2 Ethernet 통신 (MAIN B/D V3.0A 부터 지원)

#### 3.2.1 Hardware 설치 방법

다음과 같은 과정을 거쳐 제어기의 TCP/IP 프로토콜을 사용할 수 있습니다.

- 1) 제어기 Main Board 통신포트(Ethernet PORT)에 Ethernet 케이블을 연결합니다.

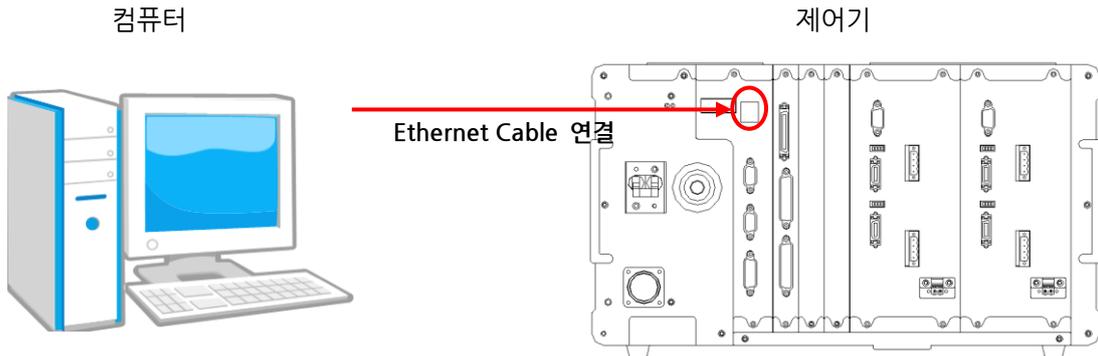
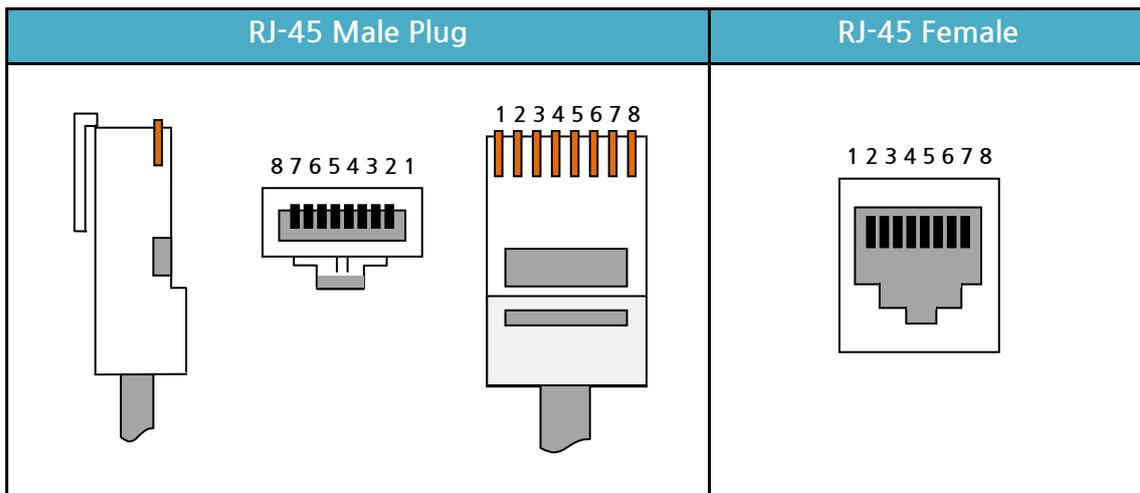


그림 3.3 Ethernet 케이블 연결방법

#### 3.2.2 Cable 연결 방법

Atuto-MDIX(Automatic medium dependent interface crossover)지원으로 다이렉트 케이블과 크로스 케이블 구분없이 사용 하실 수 있습니다.

케이블 연결 방법은 “그림 3.4 케이블 결선도”를 참조하시기 바랍니다.



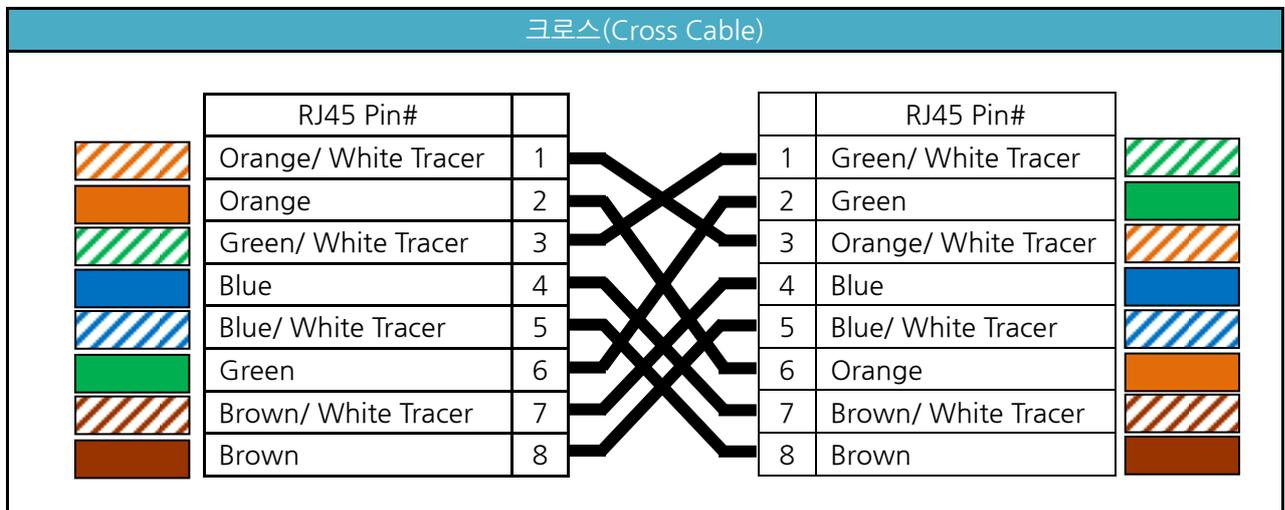
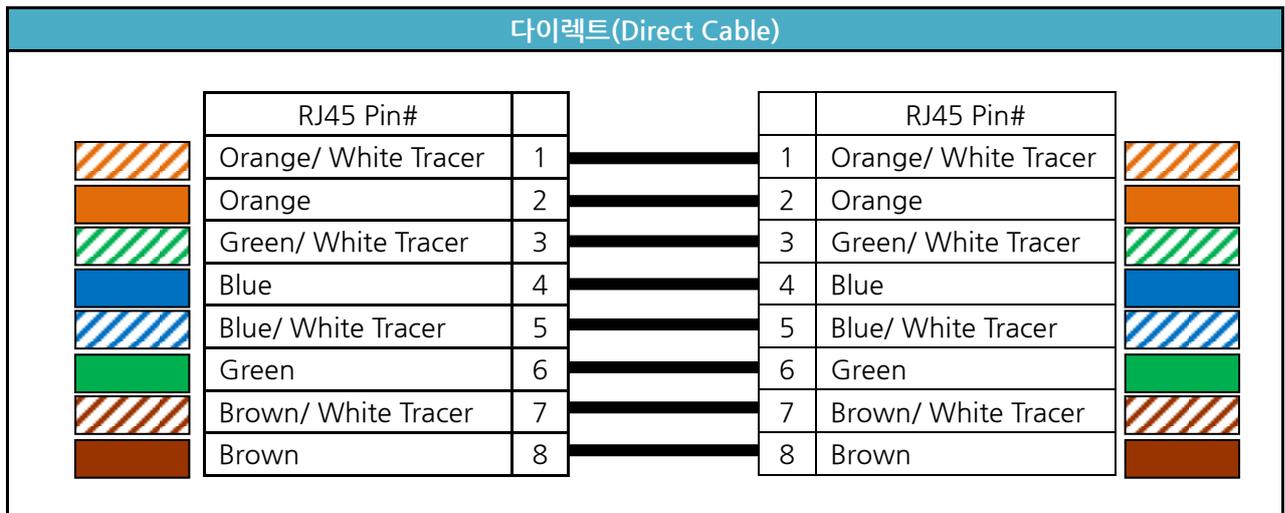


그림 3.4 케이블 결선도

### 3.2.3 Controller 설정

제어기에서 Robostar 프로토콜을 사용하기 위해서는 Ethernet 통신 설정을 하셔야 합니다.  
 Unihost 프로토콜의 경우 TCP 프로토콜을 사용하며, 포트번호는 20000 번지로 고정되어있습니다.

#### ■ Ethernet 통신 설정(TCP/IP)

##### 1. 설정순서

##### Step 1.

##### PARA 화면 이동

```

<MAIN MENU>
1. JOB          2. RUN
3. HOST         4. PARA
5. ORIGIN      6. I/O
7. SYSTEM      8. GPNT
9. INT/FLT     A. ALARM

                SELECT #
    
```

초기 메뉴 화면 OPEN  
 4: PARA 선택



```

<PARAMETER>
NO          TYPE
*CH1       XYZW
CH2        XY_TEST

SEL  INFO  PUB  EXIT
    
```

PUBLIC PARAMETER 그룹화면 OPEN



```

<PUBLIC PARAMETER>
1: HW CONF    2: PALLET
3: PLC        4: ETC

                group #
    
```

1:HW CONF 선택



##### Step 2.

##### COMM 화면 이동

```

<PUBLIC-HW CONF(0)>
1: TMR          2: COMM
3: I/O         4: D-MAN
5: SVON        6: A I/O

                item #
    
```

2: COMM 선택



```

<HW CONF - COM>
COMMUNICATION SET
1: MAIN COMM
2: FIELD BUS

                group #
    
```

1: MAIN COMM 선택



```

<HW CONF - COMM>
  MAIN COMM SET
  1: SERIAL
  2: ETHERNET
  
```

group #

2:ETHERNET 선택

2  
R

Step 3-1.

IP Address 설정 화면

```

<MAIN - ETHER>
  1: IP ADDR
  2: MAC ADDR
  
```

item #

IP ADDR 선택

1  
Q

```

<ETHER- IP ADDR>
IP  :192.168. 1.203
GW :192.168. 1. 1
  
```

VIEW                      SAVE

Internet Protocol 주소와 Gateway 주소의 경우  
사용 네트워크 환경에 맞게 설정하시기 바랍니다.

Step 3-2.

MAC Address 확인

```

<MAIN - ETHER>
  1: IP ADDR
  2: MAC ADDR
  
```

item #

MAC ADDR 선택

2  
R

```

<ETHER- MAC ADDR>
  READ ONLY MAC ADDR
  C4 - 7C - 8D - 40 - 01 - 00
  
```

ESC : EXIT

MAC 주소의 경우 공장 출하 시 설정되는 고유  
주소값이며, 임의로 변경 하실 수 없습니다.

■ 컴퓨터와 Ethernet 통신을 위한 규격

IP(Internet Protocol)와 GW(Gateway) 주소의 경우 사용하시는 네트워크 환경에 맞게 설정 하시기 바랍니다.

항 목	초기값
IP	192.168. 1.203
GW	192.168. 1. 1

※ TCP 통신 포트번호는 "20000"로 설정되어 있습니다.



**CAUTION**

- ▶ Ethernet 통신의 경우 MAIN Board V3.0A 부터 지원 합니다.
- ▶ N1 제어기의 경우 Ethernet 통신은 TCP/IP 프로토콜을 사용하고 있습니다.
- ▶ Unihost Protocol의 통신 포트번호는 "20000"로 설정 되어있습니다.
- ▶ MAC 주소의 경우 공장 출하 시 설정되는 고유 주소값으로 임의로 변경 하실 수 없습니다.

## 제4장 통신 프로토콜

### 4.1 통신 규칙

- 최대 Packet 길이는 STX, ETX, LRC를 포함하여 250Bytes까지 송/수신이 가능 합니다.
- Packet에 포함되어 있는 LRC를 확인 하여 LRC가 다른 경우 NAK를 보내고 일정 횟수 이상이면 RST를 보냅니다. NAK를 받으면 다시 이전 Packet을 보내야 합니다.
- RST는 통신 종료를 의미하며 RST를 받으면 즉시 통신을 종료하고 통신 대기점으로 복귀합니다.
- ACK 는 한 개의 통신 Packet을 잘 받았다는 Acknowledgement의 의미로 사용됩니다.  
한 개의 통신 Packet이란 다음과 같이 STX, DATA, ETX, LRC 로 이루어진 구조를 나타 냅니다.

STX	DATA	ETX	LRC
-----	------	-----	-----

STX: 0x02

ETX: 0x03

NAK: 0x15

RST: 0x12

ACK: 0x06

- LRC 생성 및 확인 방법입니다. (만약 LRC 계산값이 0이면 ETX로 합니다.)  
1) PC -> N1 의 경우 (ETX 제외)  
- LRC 계산법 :STX, ETX, LRC를 제외한 **Exclusive-OR** 입니다  
-아래 적색으로 표시된 데이터까지 각각의 바이트를 XOR 해야 합니다.  
2) N1 -> PC의 경우 (ETX 포함)  
- LRC 계산법 :STX, LRC를 제외한 **Exclusive -OR** 입니다  
-아래 청색으로 표시된 데이터까지 각각의 바이트를 XOR 해야 합니다.

예1) 글로벌 변수 Data 쓰기(GW)

PC	STX	Dummy(0xFF)	'G'	'W'	타입	채널	좌표계	축 번호	INDEX	DATA	ETX	LRC
----	-----	-------------	-----	-----	----	----	-----	------	-------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

예2) 글로벌 변수 Data 읽기(GR)

PC	STX	Dummy(0xFF)	'G'	'R'	타입	채널	좌표계	축 번호	INDEX	ETX	LRC
----	-----	-------------	-----	-----	----	----	-----	------	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	'0'	Global Point Data	ETX	LRC
-----	-----	-------------	------	-----	-------------------	-----	-----

- FLAG
  - 0x30: 기능 실행 OK  
요구한 프로토콜의 기능에 대하여 실행이 가능 하거나 완료 되었음을 나타냅니다.
  - 0x31: 프로토콜 ERROR  
PC 에서 보내는 Packet 내용이 제어기가 판단 가능한 DATA가 아닌경우 반환 됩니다.  
EX) “지정한 데이터로 모션 수행(BC)”의 경우 좌표계 입력 값이 ‘0’ or ‘1’인데 그 이상의 값이 입력 된 경우 0x31를 반환 합니다.
  - 0x32: 기능 실행 FAIL  
PC에서 요청한 기능을 실행 하지 못하는 경우 반환됩니다. 기능 실행 FAIL에 대한 자세한 내용은 “통신 에러 원인 읽기(KD)”를 이용하여 확인 가능 합니다.
  - 0x33: 현재 제어기가 지원하지 않는 기능  
프로토콜은 존재하지만 제어기 종류에 따라 기능 구현이 어려운 경우에 0x33을 반환 합니다.
  - 0x34: 연속된 패킷의 종료  
Packet이 한번으로 끝나지 않고 반복 될때 Packet의 종료를 나타냅니다.  
제어기로부터 파일을 읽어올 때에 파일의 크기에 따라 한 패킷을 넘는 경우가 발생 될 경우 제어기는 파일의 내용을 보내주다가 파일 내용을 다 보내게 되면 FLAG에 0x34를 보내어 Packet 종료를 나타냅니다.
  - 0x35: 통신 패킷 오버플로우  
수신가능한 최대 Packet 길이를 초과한 경우에 반환되거나 통신 프로토콜 명령어에 지원하지 않는 명령어 전송시 발생합니다. 통신 패킷 오버플로우에 대한 자세한 내용은 “통신 에러 원인 읽기(KD)”를 이용하여 확인 가능 합니다.

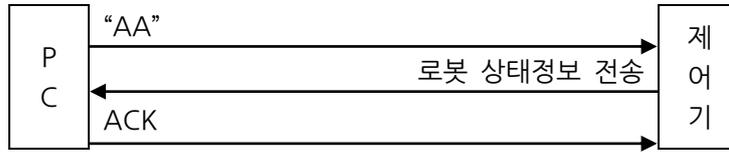
4.2 세부 프로토콜 명령어

GROUP 1	GROUP 2	기능 설명
A	A	로봇 상태 정보 읽기
	B	현재 Error 상태 읽기
	C	로봇 현재 위치 좌표 읽기
	D	제어기 INFO 읽기
B	A	원점 수행
	B	지정된 Point 번호 좌표로 로봇 이동
	C	지정한 데이터로 모션 수행
	D	현재 위치에서 지정한 DATA 만큼 증분 이동
	E	JOG Start
	F	JOG MOVE Continue
	G	JOG MOVE Stop
C	A	로봇 속도 읽기
	B	로봇 속도 쓰기
	C	JOB Start
	D	JOB Stop
	E	JOB Reset
	F	온라인 비상 정지
	G	Error Reset
	I	Origin Stop
	J	제어기 재부팅
	X	보간 이동 기준 축 변경(Desk TOP 전용 명령어)
	Z	보간 이동 기준 축 읽기(Desk TOP 전용 명령어)
	D	B
C		실행 할 JOB 설정
E	A	JOB의 Auto/Step Mode 설정
	D	현재 실행중인 JOB Step 번호 읽기
	F	현재 실행중인 JOB 이름 읽기
F	A	제어기에서 PC로 파일 전송
	B	PC에서 제어기로 파일 전송
	C	파일 검색
	D	파일 정보 요청

GROUP 1	GROUP 2	기능 설명
F	E	파일 삭제
	F	파일 복사
	G	파일 이름 변경
	H	저장되어 있는 알람 내용 읽기
G	A	I/O 카드 INPUT 접점 상태 정보 읽기
	B	I/O 카드 OUTPUT 접점 상태 정보 읽기
	C	I/O 카드 OUTPUT 접점 출력
	D	I/O 카드 정보 읽기
	R	Global Point, Global Integer, Global Float Read
	W	Global Point, Global Integer, Global Float Write
K	D	통신 에러 원인 읽기
	E	시스템 DATA 읽기
V	A	Vision Point 저장
	B	대상물의 양/불 판정값 전송
	M	Vision Point DATA 연속 저장
	N	Vision Point 증분 DATA 저장

### 4.3 로봇 상태 정보 읽기(AA)

로봇의 상태(채널별 로봇 상태)정보 읽기 명령어입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'A'	'A'	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG(0x30)	Channel 1	Channel 2	Channel 3	ETX	LRC
-----	-------------	------------	-----------	-----------	-----------	-----	-----

PC

ACK(0x06)
-----------

- Channel 1 : 로봇 Channel 1 상태정보를 반환 합니다.  
반환값이 0xB5 이면 Channel 1번의 상태정보는 Servo On, Origin, Ready, Run 상태를 의미합니다.

DATA 1(1Byte)							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	0	Servo On	Origin	Alarm	Ready	In Position	Run

- Channel 2 : 로봇 Channel 2 또는 BGT(Back Ground Task) 상태정보를 반환 합니다.  
반환 값이 0x84 이면 Channel 2번의 상태정보는 Ready 상태를 의미합니다.

DATA 2(1Byte)							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	0	Servo On	Origin	Alarm	Ready	In Position	Run

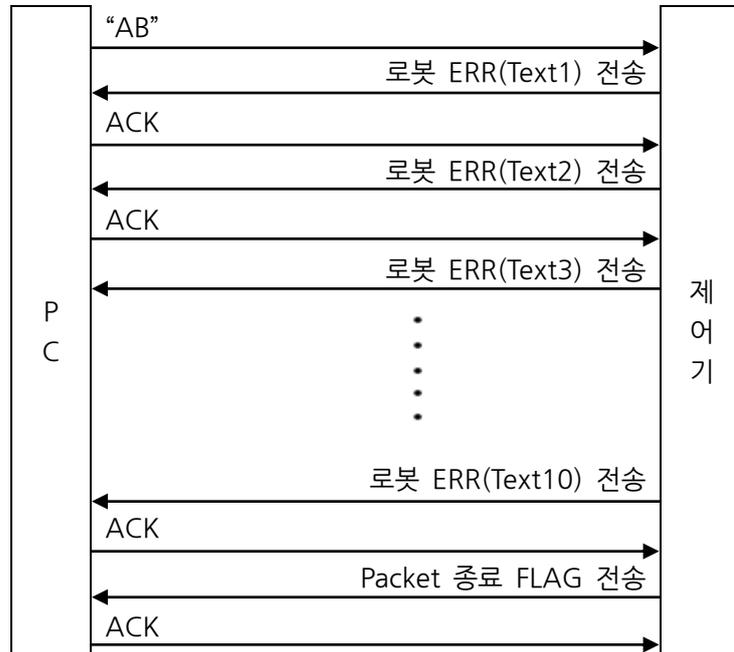
- Channel 3 : BGT(Back Ground Task) 상태정보를 반환 합니다.  
반환 값이 0x88 이면 Channel 3번의 상태정보는 Alarm 상태를 의미합니다.

DATA 3(1Byte)							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	0	Servo On	Origin	Alarm	Ready	In Position	Run

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

현재 ERROR 상태 읽기(AB)  
제어기 알람 정보 읽기 명령어 입니다.



ERR Text는 최대 10개까지 순차적으로 전송 할수 있습니다.

- Protocol DATA

PC	STX	Dummy(0xFF)	'A'	'B'	ETX	LRC
----	-----	-------------	-----	-----	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	'E'	ERR Text 1 EX) "1153 : T/P Emergency	ETX	LRC
-----	-----	-------------	------------	-----	---	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	'E'	ERR Text 2 EX) "1104 : Servo Not Redy"	ETX	LRC
-----	-----	-------------	------------	-----	---	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC	ACK
----	-----

- ERR Text: 알람코드와 알람 내용을 포함하여 총 27Byte로 이루어져 있습니다.

ERR Text(27Byte)		
Err Code(4Bytes)	(3Bytes)	Err 내용(20Bytes)
"1153"	“+ ‘:’ + “	“T/P Emergency”

- FLAG : 연속 되는 ERR Text 내용에 대하여 종료 지점을 알려 줍니다.  
 0x30: 요구한 프로토콜의 실행이 완료되었음을 의미합니다.  
 0x34: 연속된 Packet의 종료를 알립니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

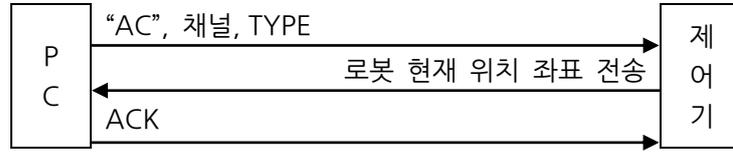


**CAUTION**

- ▶ 제어기가 알람 상태에 있는 경우에만 알람 내용에 대하여 반환됩니다.  
(현재 발생된 알람 내용을 알고 싶을 경우 사용합니다.)
- ▶ 제어기의 과거 알람을 알고 싶으신 경우 “저장되어 있는 알람 내용 읽기(FH)”를 이용하여 알 수 있습니다.

#### 4.4 로봇 현재 위치 좌표 읽기(AC)

지정한 채널의 로봇 현재 위치 좌표 읽기 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'A'	'C'	채널	TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	FLAG	ETX	LRC
-----	-----	-------------	------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 현재 위치 좌표 읽어 올 로봇 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE(0~2): 요청 DATA 타입(1Byte)  
 0: 로봇 현재 위치 정보를 Pulse DATA로 요청  
 1: 로봇 현재 위치 정보를 Angle DATA로 요청  
 2: 로봇 현재 위치 정보를 XY좌표 DATA로 요청
- DATA

STX	FLAG	DATA	ETX	LRC		
	1축 (10Byte)	2축 (10Byte)	3축 (10Byte)	4축 (10Byte)	.....	ARM (1Byte)

TYPE 0: Pulse DATA로 정수형입니다.(10Byte\*로봇 채널 축수+ARM)

1축(10Bytes)					.....	최대 축(10Bytes)					ARM(1Byte)
1Byte	1Byte	.....	1Byte	1Byte	.....	1Byte	1Byte	.....	1Byte	1Byte	1Byte
"123456789"				“	.....	"123456789"				“	'2'

TYPE 1: Angle DATA로 실수형입니다.(소수점 3자리까지 표시 합니다.)

1축(10Bytes)										.....
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	.....
"12345"					‘.’	"123"			‘‘	.....

최대 축(10Bytes)										ARM(1Byte)
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					‘.’	"123"			‘‘	‘2’

TYPE 2: XY좌표 값으로 실수형입니다.(소수점 3자리까지 표시 합니다.)

1축(10Bytes)										.....
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	.....
"12345"					‘.’	"123"			‘‘	.....

최대 축(10Bytes)										ARM(1Byte)
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					‘.’	"123"			‘‘	‘0’or’1’

- 최대축: 로봇 채널에서 사용되고 있는 최대 축을 의미 합니다. 최대 축 확인은 “제어기 INFO 읽기(AD)” 프로토콜로 확인 가능합니다.

- ARM(0~2): 스카라 로봇의 경우 로봇 자세를 뜻합니다.  
0: LEFT\_form  
1: RIGHT\_form  
2: NO\_form
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

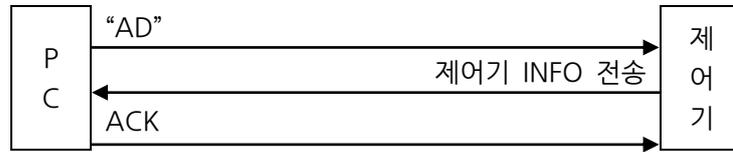
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

 CAUTION

- ▶ XY좌표 Data 값을 요청할 경우에만 ARM\_Form에 대한 값을 반환 합니다.
- ▶ Angle Data의 경우 Arm\_Form 정보값은 No\_Form값을 반환 합니다.

### 4.5 제어기 INFO 읽기(AD)

제어기 INFO 읽기 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'A'	'D'	ETX	LRC
----	-----	-------------	-----	-----	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	MAX CH	NAME	VER	MODEL	채널 정보	ETX	LRC
-----	-----	-------------	------	--------	------	-----	-------	-------	-----	-----

PC	ACK
----	-----

- MAX CH: 제어기 파라미터에 설정되어 있는 최대 운영 채널수를 반환 합니다.( 채널은 최대 3개 까지 운영 가능 합니다, 설정 방법은 “조작운용설명서 1.3.1.8 MAX CH” 참조하시기 바랍니다.
- NAME: 제어기에 설정되어 있는 이름을 반환 합니다. 만약 설정이 되어 있지 않을 경우 ‘ ’으로 15Bytes를 반환 합니다. (설정방법은 “조작운용 설명서 1.3.4.4 NAME” 참조 하시기 바랍니다.)

NAME(15Bytes)
EX) "N1-TESTNAME"

- VER: 펌웨어 버전입니다.

VER(20Bytes)
EX) "N1RO 03.02.05-SB"

- MODEL: 각 채널의 로봇 타입을 반환 합니다.

채널 1 MODEL(10Bytes)	채널 2 MODEL(10Bytes)	채널 3 MODEL(10Bytes)
"RSA60A"	"XY"	"BGT"

- 채널정보: 채널 설정에 대한 정보를 반환 합니다.

MAX AXIS(3Bytes)			채널 TYPE(3Bytes)			Using AXIS(3Bytes)		
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
CH1	CH2	CH3	CH1	CH2	CH3	CH1	CH2	CH3

- MAX AXIS(3Byte): 각 채널에 대하여 최대 사용할수 있는 축에 대한 정보를 반환 합니다.

CH1MAX AXIS	CH2 MAX AXIS	CH3 MAX AXIS
1 Byte	1 Byte	1 Byte

각각1~6까지 반환 합니다.

- 채널 TYPE(3Byte): 현재 설정되어 있는 채널 정보를 반환 합니다.

채널1 TYPE	채널2 TYPE	채널3 TYPE
1 Byte	1 Byte	1 Byte

- 0: XY\_ROBOT
- 1: SCARA\_ROBOT
- 2: TRANSFER\_ROBOT
- 3: CYLINDER\_ROBOT
- 4: BACKGROUND\_TASK
- 5: NOT\_DEFINE\_ROBOT

- Using AXIS(3Byte): 각 채널에 현재 사용중인 축 정보를 반환 합니다.

CH 1 Using Axis				CH 2 Using Axis				CH 3 Using Axis							
1Byte								1Byte							
8bit	7 bit	6 bit	5 bit	4 bit	3 bit	2 bit	1 bit	8bit	7 bit	6 bit	5 bit	4 bit	3 bit	2 bit	1 bit
'0'	'1'	6AXIS	5AXIS	4AXIS	3AXIS	2AXIS	1AXIS	'0'	'1'	6AXIS	5AXIS	4AXIS	3AXIS	2AXIS	1AXIS

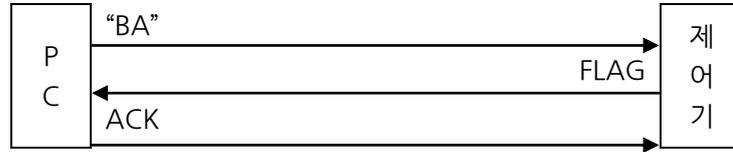
- 0: AXIS 사용 안함.
- 1: AXIS 사용 함.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

#### 4.6 원점 수행(BA)

로봇 원점(Origin)수행 명령어입니다.



- Protocol DATA

PC      

STX	Dummy(0xFF)	'B'	'A'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기      

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC      

ACK
-----

- 채널(0~2): 원점 수행 할 로봇의 채널(1Byte)  
 0: 채널 1번 로봇 원점수행.  
 1: 채널 2번 로봇 원점수행.  
 2: 채널 3번 로봇 원점수행.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

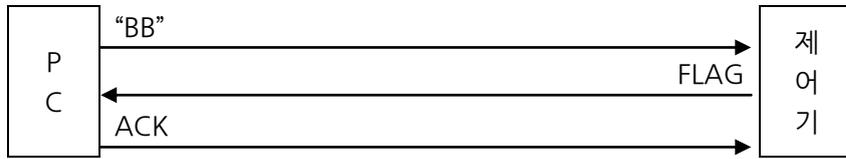
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

**CAUTION**

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “원점수행(BA)”시 자동으로 Servo ON이 됩니다
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “원점수행(BA)”전 “Servo ON/OFF(DB)’를 사용하여 Servo ON 후 원점수행을 하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

### 4.7 지정한 Point번호 좌표로 로봇 이동(BB)

지정한 Point 번호로 로봇 이동시 모션 수행 명령어 입니다.



● Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'B'	채널	파일 이름	모션 TYPE	Point Num 1	Point Num 2	ETX	LRC
----	-----	-------------	-----	-----	----	-------	---------	-------------	-------------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 파일 이름: Point파일의 파일명과 확장자(12Bytes)  
 - 파일명+ '.' 그리고 확장자를 포함하여 12글자입니다.  
 ※ 파일 이름이 5글자를 초과한 경우 "0x32 기능 실행 FAIL"의 FLAG가 발생합니다.  
 ※ 파일명은 대문자, 확장자명은 대문자(PNT) 혹은 소문자(pnt)만 가능합니다.  
 ※ 대, 소문자를 혼용하거나 확장자명이 다른 경우 "0x31 프로토콜 ERROR"의 FLAG가 발생합니다.

파일 이름(12Bytes)											
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
파일명 + '.' + 확장자											

- 모션 TYPE(0~3): 동작 모션 타입(1Byte)
  - 0: Joint(JMOV : PTP이동) 목표 지점의 Local Point번호 1개 .
  - 1: Linear(LMOV: 직선보간 이동) 목표 지점의 Local Point번호 1개.
  - 2: Arc(AMOV: 원호 보간이동) 경유 할 지점의 Local Point번호 1개 와 목표 지점의 Local Point 번호 1개.
  - 3: Circle(CMOV: 원형 보간 이동) 경유 할 지점의 Local Point번호 2개.
- ※ 로봇은 현재 위치 좌표부터 목표 좌표(Local Point번호)까지 이동 경로를 계산 합니다.
- Point Num 1, Point Num2: 경유 좌표 저장 Point 번호 및 목표 좌표 저장 Point 번호
  - 모션 Type 0, 1경우.
  - Local Point 5를 목표 좌표로 이동하는 경우.

Point Number 1 (4 Bytes)				Point Number 2 (4 Bytes)			
'0'	'0'	'0'	'5'	'0'	'0'	'0'	'0'

※ Point Number 2는 이동경로에 대하여 반영을 안합니다.

모션 Type 2, 3경우.

- Local point 5와 Local Point 15를 지나는 원과 원호 보간 이동하는 경우.

Point Number 1 (4 Bytes)				Point Number 2 (4 Bytes)			
'0'	'0'	'0'	'5'	'0'	'0'	'1'	'5'

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다..

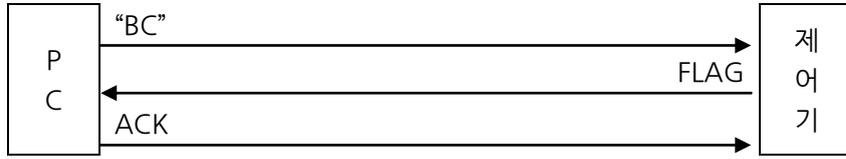
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

 **CAUTION**

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “BB”명령 수행시 자동으로 Servo ON 상태로 변경됩니다..
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “BB”명령 수행전에 “DB”명령을 통하여 Servo ON 상태로 변경하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

#### 4.8 지정한 데이터로 모션 수행(BC)

로봇의 경유 좌표 및 목표 좌표를 입력하여 모션을 수행하는 명령어입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'C'	채널	모션 TYPE	좌표계	위치 DATA	ETX	LRC
----	-----	-------------	-----	-----	----	---------	-----	---------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 모션 수행 할 로봇의 채널(1Byte)
  - 0: 로봇 채널1에 해당 됩니다.
  - 1: 로봇 채널2에 해당 됩니다.
  - 2: 로봇 채널3에 해당 됩니다.
- 모션 TYPE(0~3): 동작 모션 설정(1Byte)
  - 0: JMOV: 현재 위치에서 목표점으로 축보간 이동.
  - 1: LMOV: 현재 위치에서 목표점으로 직선보간 이동.
  - 2: AMOV: 현재 위치에서 경유점 1개와 목표점 1개를 잇는 원호를 그리며 이동.
  - 3: CMOV: 현재 위치에서 경유점 2개를 잇는 원을 그리며 이동.
  - ※ 자세한 동작 모션은 “프로그래밍 설명서 3.9~3.12”까지 참조 하시기 바랍니다.
  - ※ 이 외의 값을 입력하면 “0x33 현재 제어기에 구현되지 않은 기능”의 FLAG가 발생합니다.
- 좌표계(0,1): 좌표계 설정(1Byte)
  - 0: Angle 좌표: 도(Degree)단위의 좌표.(직각로봇일 경우 mm단위로 표현.)
  - 1: XY 좌표: mm단위의 좌표.
  - ※ 이 외의 값을 입력하면 “0x31 프로토콜 ERROR”의 FLAG가 발생합니다.



**CAUTION**

▶ SCARA 로봇의 경우 XYZ 좌표계 Data는 로봇의 현재 자세를 가지고 로봇의 이동 경로를 계산하게 됩니다.

- 위치 DATA : 이동 할 목표 좌표 DATA
  - 좌표계 설정값이 Joint Angle TYPE일 경우(로봇 타입이 직각로봇일 경우 mm 단위로 동작 됩니다.)

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- ※ MAX AXIS 보다 위치 DATA가 적을 경우 원하는 모션 동작이 안될수 있습니다.
- ※ 현재 로봇의 MAX AXIS를 모르는 경우 “제어기 INFO 읽기(AD)”를 이용하여 MAX AXIS 확인후 위치 DATA 값을 입력하십시오.

EX) 4축 로봇의 경우: 4축 × 10Bytes = 40Bytes)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	4축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- 좌표계 설정값이 XYZ TYPE일 경우(로봇 타입이 SCARA일 경우 XY 좌표계를 지원합니다.)

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

X 위치 DATA(10bytes)				Y 위치 DATA(10bytes)				...	W 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- ※ AMOV : 경유점 1개의 위치 DATA와 목표점 1개의 위치 DATA
- ※ CMOV : 경유점 2개의 위치 DATA

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

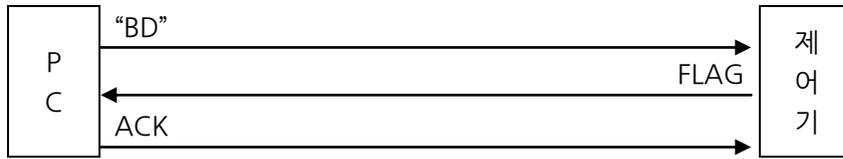
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

 **CAUTION**

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “BC”명령 수행시 자동으로 Servo ON 상태로 변경됩니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “BB”명령 수행전에 “DB”명령을 통하여 Servo ON 상태로 변경하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

#### 4.9 현재 위치에서 지정한 DATA만큼 증분이동(BD)

현재 위치에서 입력한 DATA 만큼 로봇을 증분 이동 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'D'	채널	모션 TPYE	좌표계	위치 DATA	ETX	LRC
제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC					
PC	ACK									

- 채널(0~2): 증분 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 모션 TYPE(0, 1): 동작 모션 설정(1Byte)  
 0: JMOV : 현 위치에서 위치 DATA 만큼 증분 이동  
 1: LMOV : 현 위치에서 위치 DATA 만큼 증분 이동  
 ※ 이 외의 수를 입력하면 "0x33 현재 제어기에 구현되지 않은 기능"의 FLAG가 발생합니다.
- 좌표계(0, 1) : 좌표계 설정(1Byte)  
 0: Angle 좌표 : 도(Degree)단위의 좌표.(직각 로봇일 경우 mm단위로 동작)  
 1: XY 좌표 : mm단위의 좌표.  
 ※ 이 외의 값을 입력하면 "0x31 프로토콜 ERROR"의 FLAG가 발생합니다.



#### CAUTION

- ▶ SCARA 로봇의 경우 XYZ 좌표계 Data는 로봇의 현재 자세를 가지고 로봇의 이동 경로를 계산하게 됩니다.

- 위치 DATA : 증분 이동 할 좌표 DATA
  - 좌표계 설정값이 Angle 좌표일 경우  
각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

EX) 4축 로봇의 경우 : 4축 × 10Bytes = 40Bytes

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	4축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- 좌표계 설정값이 XYZ TYPE일 경우  
각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

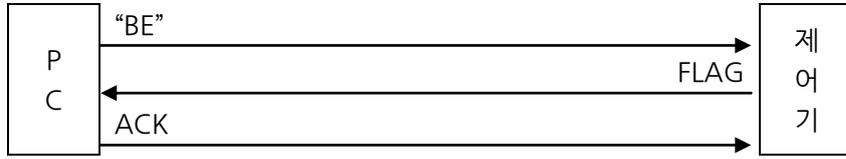


**CAUTION**

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우  
“BD”명령 수행시 자동으로 Servo ON 상태로 변경됩니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우  
“BD”명령 수행전에 “DB”명령을 통하여 Servo ON 상태로 변경하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

#### 4.10 JOG START(BE)

지정한 축에 대하여 JOG동작 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'E'	채널	AXIS	방향	모션 TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	----	---------	-----	-----

=

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK	s
----	-----	---

- 채널(0~2): 모션 수행 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- AXIS(0~5): 로봇 채널에 속한 로봇을 움직이고자 하는 축 번호를 입력 합니다.  
 0: 1(X)축 구동  
 1: 2(Y)축 구동  
 2: 3(Z)축 구동  
 3: 4(W)축 구동  
 4: 5(EX1)축 구동  
 5: 6(EX2)축 구동  
 ※ 이 외의 값을 입력하면 "0x31 프로토콜 ERROR"의 FLAG가 발생합니다.
- JOG 동작 방향(0, 1) : 로봇 축의 이동 방향을 선택합니다.  
 0: - 방향  
 1: + 방향  
 ※ 이 외의 값을 입력하면 "0x31 프로토콜 ERROR"의 FLAG가 발생합니다.

- 모션 TYPE(0, 1): 동작 모션 설정(1Byte)  
0(JMOV): 현 위치에서 설정한 방향으로 로봇축 이동  
1(LMOV): 현 위치에서 설정한 방향으로 직선 보간 이동  
※ 이 외의 값을 입력하면 “0x31 프로토콜 ERROR”의 FLAG가 발생합니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

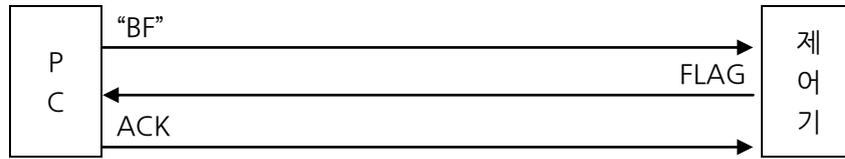


#### CAUTION

- ▶ 연속적인 Jog동작을 유지하기 위해서는 500ms 이내로 BF(Jog Move Continue)프로토콜을 전송을 하여야 합니다.
- ▶ SCARA 로봇일 경우 LMOV로 이동시 XY좌표계로 동작합니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “BE”명령 수행시 자동으로 Servo ON 상태로 변경됩니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “BE”명령 수행전에 “DB”명령을 통하여 Servo ON 상태로 변경하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

### 4.11 JOG MOVE Continue(BF)

JOG 동작이 계속 유지 되도록 제어기에 요구하는 명령어 입니다.



- Protocol DATA

PC 

STX	Dummy(0xFF)	'B'	'F'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC 

ACK
-----

- 채널(0~2): 모션 수행을 유지 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

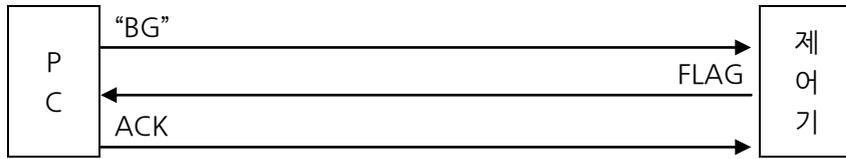


#### CAUTION

- ▶ JOG 동작을 유지 하려면 500ms 마다 “4.1.12 JOG MOVE Continue(BF)” 프로토콜을 전송 하여야 합니다.
- ▶ 500ms 초과 시 로봇은 정지 합니다.

### 4.12 JOG MOVE STOP(BG)

JOG 동작 정지 명령어 입니다.



- Protocol DATA

PC 

STX	Dummy(0xFF)	'B'	'G'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC 

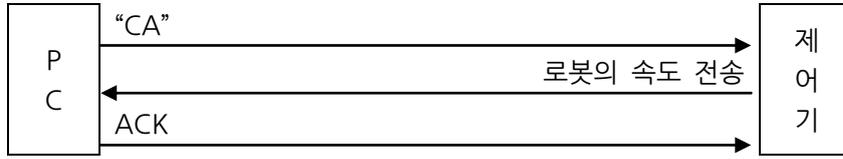
ACK
-----

- 채널(0~2): 정지 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.13 로봇 속도 읽기(CA)

로봇 채널의 로봇 현재 속도 읽기 명령어 입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'C'	'A'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	로봇 이동 속도	ETX	LRC
-----	-------------	------	----------	-----	-----

PC

ACK
-----

- 채널(0~2): 현재 로봇의 속도를 읽어 올 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 로봇 이동 속도: 현재 로봇 속도 DATA(4Bytes)
- 로봇 이동 속도 = Mv(각 축의 최대속도) \* 0.001 \* 로봇 속도

로봇 이동 속도(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'3'	'0'	'0'

- JOINT MOTION에 대한 자세한 내용이나 설정방법은 “조작운용 설명서 1.2.2.2 JONT”를 참고하시기 바랍니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

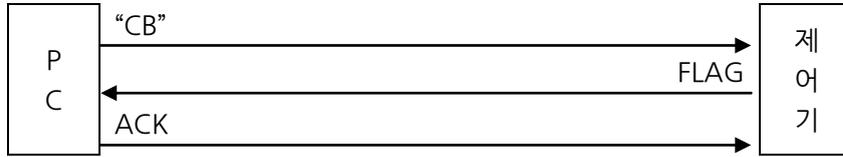
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

CAUTION

- ▶ 로봇 이동 속도의 최대값은 1000 입니다.  
 로봇 이동 속도의 출력값이 1000인 경우 Mv설정값의 100%의 속도로 이동 합니다.

#### 4.14 로봇 속도 쓰기(CB)

로봇 채널의 로봇 구동 속도 값을 입력 하는 명령어 입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'C'	'B'	채널	로봇 속도	ETX	LRC
-----	-------------	-----	-----	----	-------	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC

ACK
-----

- 채널(0~2): 지정된 속도값을 변경할 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.

- 로봇 속도: 로봇 속도 값(4Bytes)  
 - 입력 범위 : 0~1000 입니다.

로봇 이동 속도 = Mv(각 축의 최대속도) \* 0.001 \* 로봇 속도

DATA(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'1'	'0'	'0'

- JOINT MOTION에 대한 자세한 내용이나 설정방법은 “조작운용 설명서 1.2.2.2 JONT”를 참고 하시기 바랍니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

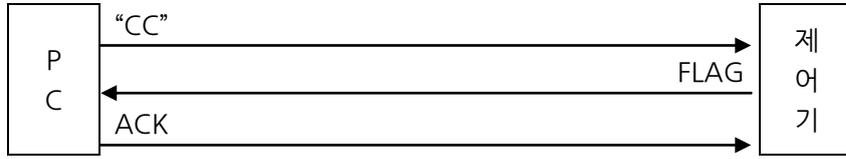
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

CAUTION

- ▶ 설정할 수 있는 최고 속도값은 1000입니다.  
 속도값 1000을 입력 시 100%의 속도로 이동 합니다. (속도값 1000 => 100%)
- ▶ 기구부의 허용 최대 RPM을 초과하여 사용 시 소음 및 파손의 위험이 있습니다.  
 반드시 기구부에 부착된 라벨을 확인 후 사용 하시길 바랍니다.

4.15 JOB START(CC)

JOB을 실행하는 명령어 입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'C'	'C'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC

ACK
-----

- 채널(0~2): JOB을 실행 할 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.

※ 원점 실행이 완료되지 않은 상태에서 Job 실행 시 Run Fail 알람이 발생 합니다.

※ 프로토콜 통신중 FLAG에서 0x32이 반환되면 이에 대한 원인을 “통신 에러 원인 읽기(KD)” 프로토콜을 이용하여 읽기를 권장한다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

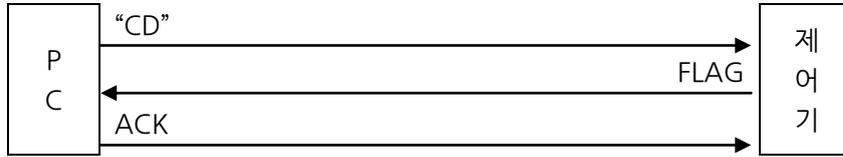
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

 CAUTION

- ▶ 반드시 “작업 실행(CC)” 하기 전에 “실행 할 JOB 설정(DC)”으로 실행 JOB을 설정한 상태에서 실행을 하여야 합니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “CC”명령 수행시 자동으로 Servo ON 상태로 변경됩니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “CC”명령 수행시 JOB은 Loading 되지만 실행은 안됩니다. “CC”명령 수행전에 “DB”명령을 통하여 Servo ON 상태로 변경하시기 바랍니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

#### 4.16 JOB STOP(CD)

로봇의 JOB 실행을 정지하는 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'C'	'D'	채널	ETX	LRC
----	-----	-------------	-----	-----	----	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC	ACK
----	-----

- 채널(0~2): JOB을 실행을 멈출 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ FLAG에 정상적인 패킷이 반환된다면 0x30이고, 잘못된 패킷이 반환되면 0x32입니다. 이에 대한 원인은 “통신 에러 원인 읽기(KD)”프로토콜을 이용하여 알 수 있습니다.

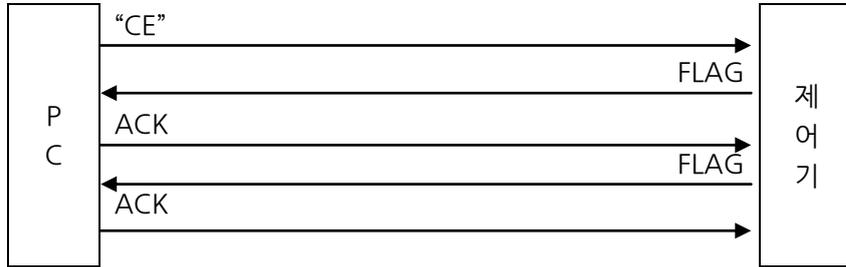


#### CAUTION

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우 “CD”명령 수행시 로봇 정지후 자동으로 Servo OFF 상태로 변합니다.
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우 “CD”명령 수행시 로봇 정지후 Servo ON 상태를 유지 합니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

### 4.17 JOB Reset(CE)

지정한 로봇 채널 JOB 실행 STEP을 처음(MAIN)으로 설정하는 명령어 입니다.



- Protocol DATA

PC     

STX	Dummy(0xFF)	‘C’	‘E’	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

- 채널(0~2): JOB STEP을 처음(MAIN)으로 설정할 로봇 채널을 설정(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

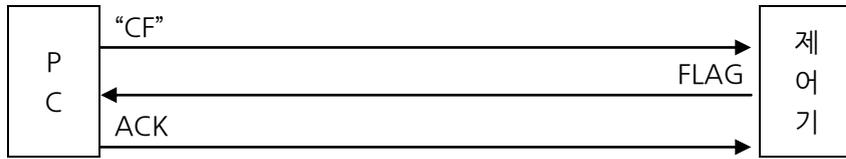


**CAUTION**

- ▶ 해당 로봇 채널이 Servo ON일 경우 FLAG는 0x32를 반환 합니다.
- ▶ “온라인 JOB Reset(CE)”은 Servo OFF 상태에서에서만 동작합니다.

#### 4.18 온라인 비상정지(CF)

로봇 온라인 비상정지 명령어 입니다.



- Protocol DATA

PC     

STX	Dummy(0xFF)	‘C’	‘F’	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

- 티치 펜던트에 “Host Emergency”를 띄우며 알람이 발생합니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

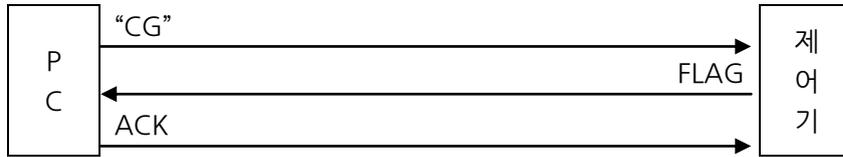


#### CAUTION

- ▶ 로봇 운전 중 긴급 정지 시에만 사용하기 바랍니다.
- ▶ CG명령을 통해 비상 정지를 해제 하실 수 있습니다.

### 4.19 Error Reset(CG)

제어기에 발생된 알람을 초기화 하는 명령어 입니다.



- Protocol DATA

PC     

STX	Dummy(0xFF)	'C'	'G'	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

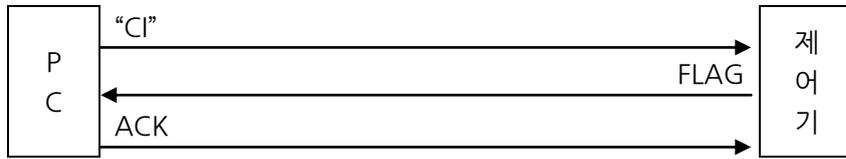
ACK
-----

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.20 Origin Stop(CI)

해당 채널의 원점수행 정지 명령어 입니다.



- Protocol DATA

PC 

STX	Dummy(0xFF)	'C'	'I'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC 

ACK
-----

- 채널(0~2): 원점 수행 정지 로봇 채널(1Byte)  
0: 로봇 채널1에 해당 됩니다.  
1: 로봇 채널2에 해당 됩니다.  
2: 로봇 채널3에 해당 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

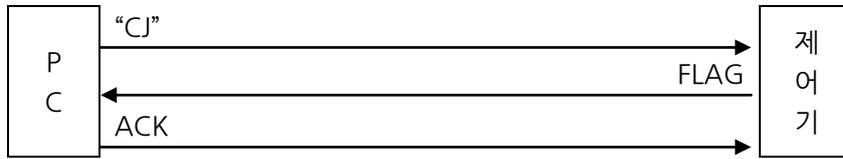


#### CAUTION

- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “ENB”일 경우  
-“Origin Stop(CI)”시 자동으로 Servo OFF가 됩니다
- ▶ N1 제어기의 Parameter에서 AUTO SERVO ON 설정이 “DIS”일 경우  
-“Origin Stop(CI)”시 원점 수행을 정지하고 Servo ON 상태를 유지 합니다.
- ▶ AUTO SERVO ON 설정은 “조작 운용 설명서 SVON”을 참고하시기 바랍니다.

### 4.21 제어기 재부팅(CJ)

제어기를 재부팅 명령어 입니다.



- Protocol DATA

PC     

STX	Dummy(0xFF)	‘C’	‘J’	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

- 제어기에 치명적인 알람이 발생되어 “Error Reset(CG)” 을 이용하여 시스템이 정상화 되지 않는 경우 사용합니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

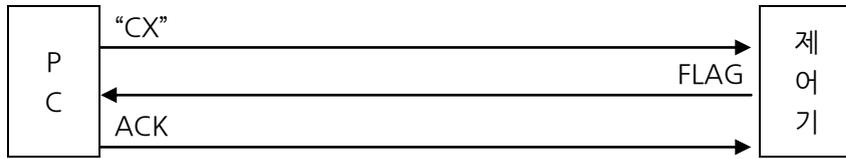
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

**CAUTION**

- ▶ 제어기 재부팅 명령어는 Servo OFF 상태에서만 가능합니다.  
Servo ON 상태에서 재부팅 할 경우 “기능 실행 FAIL(0x32)”를 반환 합니다.

### 4.22 보간 이동 기준 축 변경(CX)

XY 보간 이동시 기준이 되는 축을 변경하는 명령어 입니다.



- Protocol DATA

PC 

STX	Dummy(0xFF)	‘C’	‘X’	채널	AXIS	ETX	LRC
-----	-------------	-----	-----	----	------	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC 

ACK
-----

- 채널(0~2): 축 변경 로봇 채널(1Byte)  
0: 로봇 채널1에 해당 됩니다.  
1: 로봇 채널2에 해당 됩니다.  
2: 로봇 채널3에 해당 됩니다.
- AXIS(1또는4): 기준이 되는 축 번호  
1: X축 기준으로 JMOV, LMO, CMOV, AMOV 보간 이동.  
4: W축 기준으로 JMOV, LMO, CMOV, AMOV 보간 이동.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

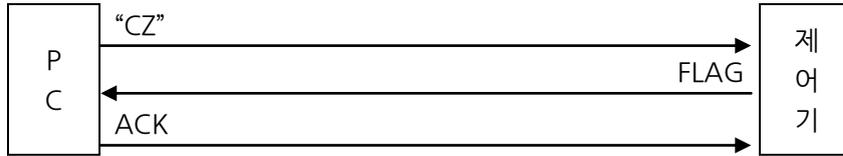
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

**CAUTION**

- ▶ DESK TOP 로봇 전용 명령어 입니다.
- ▶ Servo OFF 상태에서에서만 축 변경이 가능합니다.
- ▶ MAIN Menu 화면으로 전환시 축 변경 값은 초기화 됩니다.

### 4.23 보간 이동 기준 축 읽기(CZ)

XY 보간 이동시 현재 기준이 되는 축 정보를 읽기 명령어 입니다.



- Protocol DATA

PC     

STX	Dummy(0xFF)	'C'	'X'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

- 채널(0~2): 로봇 기준 축 정보 읽을 채널(1Byte)  
0: 로봇 채널1에 해당 됩니다.  
1: 로봇 채널2에 해당 됩니다.  
2: 로봇 채널3에 해당 됩니다.
- DATA(1또는4): 보간 이동 기준이 되는 축 번호를 반환 합니다.  
1: X축 기준으로 JMOV, LMO, CMOV, AMOV 보간 이동.  
4: W축 기준으로 JMOV, LMO, CMOV, AMOV 보간 이동.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

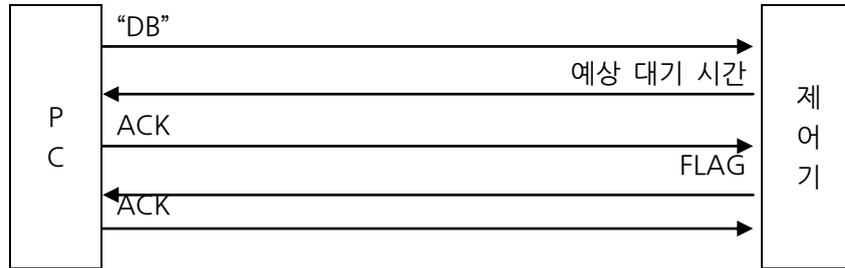


#### CAUTION

▶ DESK TOP 로봇 전용 명령어 입니다.

### 4.24 SERVO ON/OFF(DB)

해당 로봇 채널의 SERVO를 ON/OFF시키는 명령어 입니다.



● Protocol DATA

PC

STX	Dummy(0xFF)	'D'	'B'	채널	DATA	ETX	LRC
-----	-------------	-----	-----	----	------	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	예상 대기 시간	ETX	LRC
-----	-------------	------	----------	-----	-----

PC

ACK
-----

제어기

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC

ACK
-----

- 채널(0~2): SERVO를 ON/OFF 할 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- DATA(0, 1) : SERVO의 ON/OFF설정(1Byte)  
 0 : SERVO OFF  
 1 : SERVO ON  
 ※ 이외의 값 입력시 "0x31 프로토콜ERROR"의 FLAG가 발생합니다.

- 예상 대기 시간:실행에 걸리는 예측 시간.(2Bytes)  
 - 예상 시간 2초

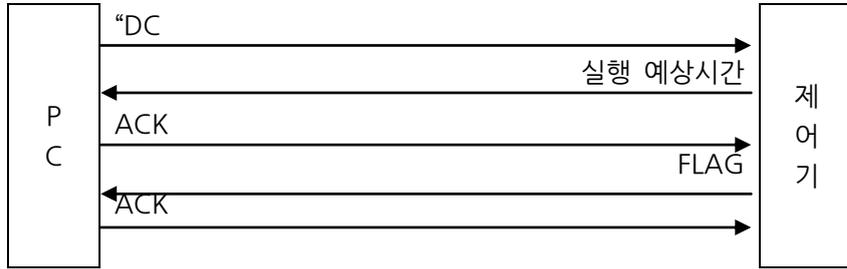
DATA2 (2Bytes)	
'0'	'2'

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

4.25 실행 할 JOB설정(DC)

실행할 JOB파일을 설정하는 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'D'	'C'	채널	파일 이름	ETX	LRC
----	-----	-------------	-----	-----	----	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	예상 대기 시간	ETX	LRC
-----	-----	-------------	------	----------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 실행 할 JOB파일을 설정 할 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 파일 이름: 실행 할 JOB파일의 파일명과 확장자명(12Bytes)  
 - 파일명, '.', 그리고 확장자를 포함하여 12글자입니다.  
 - 확장자명은 대문자(JOB) 혹은 소문자(job)만 가능합니다.

파일 이름(12Bytes)												
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	
파일명(5자 이내)								'.'	확장자("JOB"or"job")			

※ 파일명과 확장자를 제외한 나머지 부분에는 SPACE(0x20)를 입력 하여 주십시오.

- 예상 대기 시간: 실행에 걸리는 예측 시간(2Bytes)  
- 예상 시간 20초

DATA2 (2Bytes)	
'2'	'0'

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

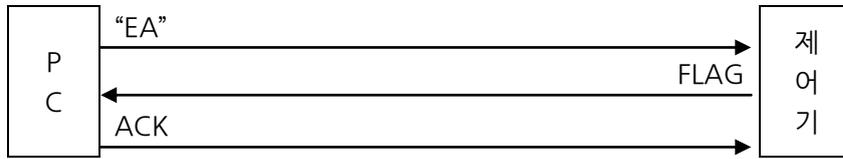


**CAUTION**

- ▶ 원점 수행이 되지 않은 상태에서 실행 Job(DC) 설정 시 Run fail 알람이 발생합니다.
- ▶ Servo ON 상태에서 “DC” 명령어 실행 시 Run fail 알람이 발생합니다.

### 4.26 JOB의 Auto/Step Mode 설정(EA)

JOB 동작 모드를 설정하는 명령어 입니다.



- Protocol DATA

PC 

STX	Dummy(0xFF)	‘E’	‘A’	채널	MODE	ETX	LRC
-----	-------------	-----	-----	----	------	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC 

ACK
-----

- 채널(0~2): JOB 실행 모드 변경 채널(1Byte)  
0: 로봇 채널1에 해당 됩니다.  
1: 로봇 채널2에 해당 됩니다.  
2: 로봇 채널3에 해당 됩니다.
- MODE(0, 1): Auto/Step Mode TYPE설정(1Byte)  
0: AUTOMODE - 프로그램 전체Step 연속실행.  
1: STEP모드 - 프로그램을 1 Step(1행)실행하고 HOLD  
※ 이외의 값 입력시 “0x31 프로토콜ERROR”의 FLAG가 발생합니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

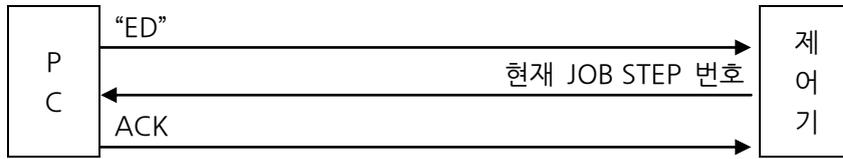
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

**CAUTION**

- ▶ 로봇 구동 중 모드 전환할 경우 FLAG는 0x32을 반환 합니다.
- ▶ 모드 전환은 해당 채널의 RUN Bit가 OFF인 경우에만 모드 전환이 가능 합니다.  
(“로봇 상태 정보 읽기(AA)”를 이용하여 RUN 상태 Bit를 확인 할 수 있습니다.)

### 4.27 현재 실행중인 JOB STEP번호 읽기(ED)

현재 실행중인 JOB STEP번호 읽는 명령어 입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'E'	'D'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	STEP 번호	ETX	LRC
-----	-------------	------	---------	-----	-----

PC

ACK
-----

- 채널(0~2): 실행중인 STEP번호를 읽을 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- STEP 번호: 실행중인 STEP 번호(4Bytes)  
 현재 JOB의 27번 STEP이 실행중인 경우

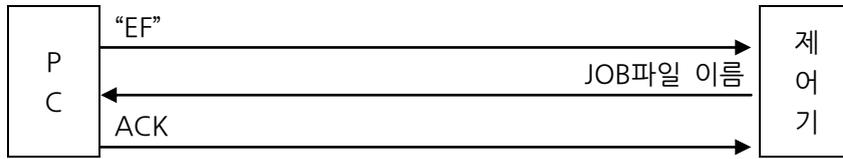
STEP 번호(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'0'	'2'	'7'

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.28 현재 실행 중인 JOB이름 읽기(EF)

지정한 로봇 채널에서 현재 실행 중인 JOB 이름 읽기 명령어 입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'E'	'F'	채널	ETX	LRC
-----	-------------	-----	-----	----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	파일 이름	ETX	LRC
-----	-------------	------	-------	-----	-----

PC

ACK
-----

- 채널(0~2): 실행중인 JOB 파일 이름을 읽을 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 파일 이름: 현재 수행중인 JOB 파일 이름.(12Bytes)  
 - 파일명 + '.' + 'J' + 'O' + 'B'의 형태로 전송합니다.  
 - 파일 이름이 들어간 나머지 뒷부분에는 ' '이 들어가게 됩니다.

EX) "RS"라는 JOB 파일이 반환이 된다면

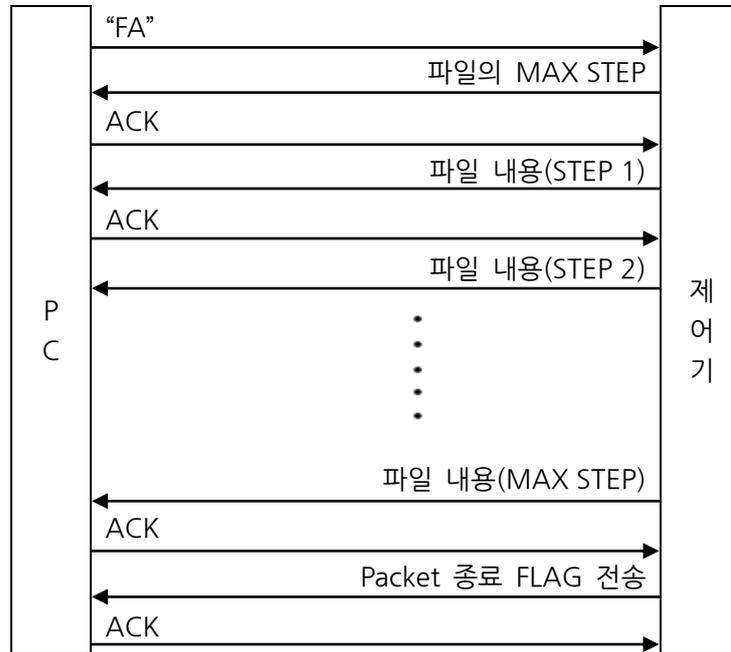
DATA(12Bytes)								
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	...	1Byte
'R'	'S'	'.'	'J'	'O'	'B'	' '	...	' '

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

### 4.29 제어기에서 PC로 파일 전송(FA)

제어기에서 PC로 JOB 파일 또는 포인트파일을 전송하는 명령어 입니다.



- Protocol DATA(JOB 파일을 요청한 경우)

PC	STX	Dummy(0xFF)	'F'	'A'	채널	TYPE	파일이름	PNT_TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	----------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	JOB MAX STEP("2000")	ETX	LRC
-----	-----	-------------	------------	----------------------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	STEP 1("MAIN")	ETX	LRC
-----	-----	-------------	------------	----------------	-----	-----

PC	ACK
----	-----

⋮

제어기	STX	Dummy(0xFF)	FLAG(0x30)	MAX STEP("EOP")	ETX	LRC
-----	-----	-------------	------------	-----------------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 전송 받을 파일을 가지고 있는 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE: 제어기의 저장위치(1Byte)  
 0: Backup RAM  
 ※ 이외의 값을 입력할 시에 “0x33 현재 제어기에 구성되어 있지 않은 기능”의 FLAG 를 받습니다.
- 파일 이름: JOB 파일 이름(12Bytes)  
 - 파일명은 최소 한 글자부터 최대 다섯 글자까지 가능합니다.  
 ※ 파일명은 대문자와 소문자를 혼용할 수 없습니다.  
 ※ 파일명이 다섯 글자를 초과하면 “0x32 기능실행 FAIL”의 FLAG 를 받습니다.  
 - 확장자명은 대문자(JOB, PNT) 혹은 소문자(job, pnt) 혼용이 가능합니다.

EX) “RS”라는 JOB 파일이 전송이 된다면

DATA(12Bytes)								
1Byte	· · ·	1Byte						
“	· · ·	“	‘R’	‘S’	‘.’	‘J’	‘O’	‘B’

※ 파일 이름과 확장자를 제외한 부분에는 SPACE(0x20)를 입력 하여 주십시오.

- PNT\_TYPE: 포인트 데이터의 경우 출력 타입 형태를 결정합니다.  
 0: Angle 타입의 위치 데이터  
 1: XY 타입의 위치 데이터

PNT_DATA
1Byte
‘0’or‘1’

※ JOB DATA 만 요청할 경우 PNT\_TYPE DATA 는 반영하지 않습니다.

- 파일 내용: JOB 파일에 저장되어 있는 내용에 따라 최대 100Byte 까지 반환 됩니다.  
 EX) IF I100==0 THEN

1 Byte · · · 16 Byte	17Byte
“IF I100==0 THEN”	‘\n’

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.
- Protocol DATA(PNT 파일을 요청한 경우)

PC	STX	Dummy(0xFF)	‘F’	‘A’	채널	TYPE	파일이름	PNT_TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	----------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	MAX Point 번호	ETX	LRC
-----	-----	-------------	------------	--------------	-----	-----

PC     ACK

제어기	STX	Dummy(0xFF)	FLAG(0x30)	Local Point 번호1	Point Data	ETX	LRC
-----	-----	-------------	------------	-----------------	------------	-----	-----

PC     ACK

⋮

제어기	STX	Dummy(0xFF)	FLAG(0x30)	Local Point 번호n	Point Data	ETX	LRC
-----	-----	-------------	------------	-----------------	------------	-----	-----

PC     ACK

제어기	STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC     ACK

- 채널(0~2): 전송 받을 파일을 가지고 있는 채널(1Byte)
  - 0: 로봇 채널1에 해당 됩니다.
  - 1: 로봇 채널2에 해당 됩니다.
  - 2: 로봇 채널3에 해당 됩니다.
  
- TYPE: 제어기의 저장위치(1Byte)
  - 0: Backup RAM
    - ※ 이외의 값을 입력할 시에 “0x33 현재 제어기에 구성되어 있지 않은 기능”의 FLAG 를 받습니다.
  
- 파일 이름: PNT 파일 이름(12Bytes)
  - 파일명은 최소 한 글자부터 최대 다섯 글자까지 가능합니다.
    - ※ 파일명은 대문자와 소문자를 혼용할 수 없습니다.
    - ※ 파일명이 다섯 글자를 초과하면 “0x32 기능실행 FAIL”의 FLAG 를 받습니다.

EX) “RS”라는 PNT 파일이 전송이 된다면

DATA(12Bytes)								
1Byte	· · ·	1Byte						
‘‘	· · ·	‘‘	‘R’	‘S’	‘.’	‘P’	‘N’	‘T’

※ 파일명과 확장자를 제외한 부분에는 SPACE(0x20)를 입력 하여 주십시오.

- PNT\_TYPE: 포인트 데이터의 경우 출력 타입 형태를 결정합니다.  
 0: Angle 타입의 좌표 데이터  
 1: XY 타입의 좌표 데이터

PNT_DATA
1Byte
‘0’or‘1’

- MAX Point: PNT 파일에 저장되어 있는 Point 번호중 가장 큰 번호를 반환 합니다.
- Local Point 번호(5Byte): Point Data 가 저장 되어 있는 번호를 반환 합니다.  
 EX) Point 번호 8 번을 반환 할 경우

Local Point(5Bytes)				
1Byte	1Byte	1Byte	1Byte	1Byte
‘P’	‘0’	‘0’	‘0’	‘8’

- Point Data: 저장되어 있는 위치 좌표를 반환 합니다.

EX) 6축 × 11Bytes = 66Bytes

1축 위치 DATA(11bytes)				2축 위치 DATA(11bytes)				· · ·
1Byte	1Byte	· · ·	1Byte	1Byte	1Byte	· · ·	1Byte	· · ·

6축 위치 DATA(11bytes)			ARM	USED	
1Byte	· · ·	1Byte	1Byte	1Byte	0x0A

- USED: Point Data 파일 사용 유/무를 나타냅니다.

0: Point Data 파일 사용 안함

1: Point Data 파일 사용 가능

- ARM(0~2):

0: LEFT\_form

1: RIGHT\_form

2: NO\_form

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용 하여 읽기를 권장합니다.



- 채널(0~2): 파일을 저장하고자 하는 로봇 채널(1Byte)
  - 0: 로봇 채널1에 해당 됩니다.
  - 1: 로봇 채널2에 해당 됩니다.
  - 2: 로봇 채널3에 해당 됩니다.
- TYPE(1): 제어기의 저장위치(1Byte)
  - 0: Backup RAM
    - ※ 이외의 값을 입력할 시에 “0x33 현재 제어기에 구성되어 있지 않은 기능”의 FLAG 를 받습니다.
- JOB 번호: JOB 파일의 Index 번호는 최대 200 까지 가능 합니다.

JOB번호(3Bytes)		
1Byte	1Byte	1Byte
'0'	'0'	;1;

- 파일 이름: JOB 파일 이름(12Bytes)
  - 파일명은 최소 한 글자부터 최대 다섯 글자까지 가능합니다.
  - ※ 파일명은 대문자와 소문자를 혼용할 수 없습니다.
  - ※ 파일명이 다섯 글자를 초과하면 “0x32 기능실행 FAIL”의 FLAG 를 받습니다.
  - 확장자명은 대문자(JOB, PNT) 혹은 소문자(job, pnt) 혼용이 가능합니다.
  - ※ 파일명과 확장자를 제외한 부분에서는 SPACE(0x20)을 입력하여 주십시오.

EX) “RS”라는 JOB 파일이 전송이 된다면

DATA(12Bytes)								
1Byte	· · ·	1Byte						
“	· · ·	“	'R'	'S'	'.'	'J'	'O'	'B'

- 파일내용: 최대 100Byte 까지 입력 가능합니다.

EX) IF I100==0 THEN

FLAG	1 Byte · · · 15 Byte	16Byte
0x30	“IF I100==0 THEN”	‘Wn’

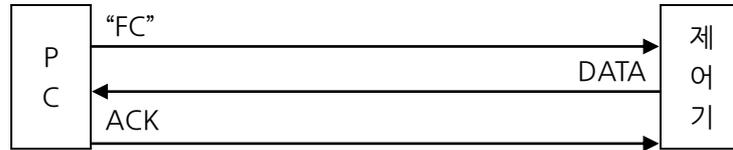
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

 **CAUTION**

- ▶ JOB 파일명이 동일한 경우 이전에 저장한 JOB 번호가 일치하여야 합니다. 만약 일치 하지 않을 경우에는 알람이 발생 합니다.
- ▶ 파일 내용 전송시 JOB MAX STEP을 제외 하고 보냅니다. 만약 JOB MAX STEP을 전송할 경우 제어기에서 JOB을 실행할 경우 “Syntax ERROR”가 발생합니다.

### 4.31 파일 검색(FC)

제어기내 파일을 검색하는 명령어입니다.



- Protocol DATA

PC

STX	Dummy(0xFF)	'F'	'C'	채널	TYPE	파일명	ETX	LRC
-----	-------------	-----	-----	----	------	-----	-----	-----

제어기

STX	Dummy(0xFF)	DATA	ETX	LRC
-----	-------------	------	-----	-----

PC

ACK
-----

- 채널(0~2): 전송 받을 파일을 가지고 있는 채널(1Byte)
  - 0: 로봇 채널1에 해당 됩니다.
  - 1: 로봇 채널2에 해당 됩니다.
  - 2: 로봇 채널3에 해당 됩니다.
- TYPE: 제어기의 저장위치(1Byte)
  - 0: Backup RAM
  - ※ 이외의 값을 입력할 시에 “0x33 현재 제어기에 구성되어 있지 않은 기능”의 FLAG 를 받습니다.
- 파일명: 검색 하고자 하는 JOB 파일 이름(12Bytes)
  - 파일명은 최소 한 글자부터 최대 다섯 글자까지 가능합니다.
  - EX) “RS”라는 JOB 파일을 검색할 경우

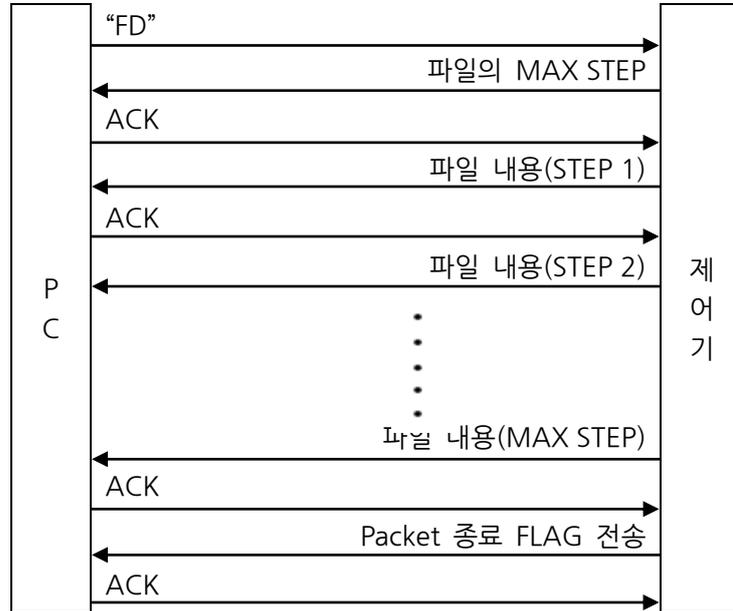
DATA(12Bytes)								
1Byte	...	1Byte						
“	...	“	'R'	'S'	'.'	'J'	'O'	'B'

- ※ 파일명이 다섯 글자 초과 시 “0x32 기능실행 FAIL”의 FLAG 를 받습니다.
- DATA(0, 1): 파일의 유/무를 판별합니다.
  - 0: 해당 파일이 존재하지 않습니다.
  - 1: 해당 파일이 존재합니다..
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

4.32 파일 정보 요청(FD)

제어기내 파일 정보를 요청하는 명령어입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'F'	'D'	채널	TYPE	파일이름	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	파일명	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	파일명	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

⋮

제어기	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	파일명	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC	ACK
----	-----

-

- 채널(0~2): 파일 정보를 요청할 로봇 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE(1Byte): 파일 저장위치  
 0: Backup RAM  
 ※ 상기 이외의 값을 입력하면 FLAG(0x33)를 반환 합니다.  
 ※ N1 제어기는 저장 공간이 '0' 만을 지원합니다.

- 파일 이름: 정보를 원하는 JOB 파일 이름.(12Bytes)  
 EX) "RS"라는 JOB 파일의 정보를 원할 경우

파일 이름 (12Bytes)								
1Byte	· · ·	1Byte						
“	· · ·	“	'R'	'S'	'.'	'J'	'O'	'B'

- 파일 이름은 5 글자 이내로 입력합니다.
- “\*.\*” 입력시 해당채널에 대한 JOB, PNT 정보를 모두 반환 합니다  
 ※ 5 글자 이상이면 “0x32 기능실행 FAIL”의 FLAG 를 반환합니다.  
 ※ 파일명은 대문자/소문자를 혼용할 수 없습니다.

- JOB NUM(3 Byte): 해당 JOB 에 번호를 반환 합니다.  
 EX) JOB 번호가 1 인 경우

JOB NUM(3Bytes)		
1Byte	1Byte	1Byte
“	“	'1'

- 파일명(10 Byte):  
 EX) RS.JOB 을 반환할 경우

파일명(10Bytes)								
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	· · ·	1Byte
'R'	'S'	'.'	'J'	'O'	'B'	“	· · ·	“

- SIZE:요청한 JOB 의 크기를 반환 합니다.(단위는 Kbyte 입니다.)  
 EX) 27Kbyte 크기를 반환할 경우

DATA(5Bytes)				
1Byte	1Byte	1Byte	1Byte	1Byte
“	“	“	'2'	'7'

- STEP(6Byte): JOB 파일의 전체 STEP 을 반환 합니다.  
 EX) JOB 파일이 200 STEP 을 저장하고 있는 JOB 을 반환할 경우

STEP (6Bytes)					
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
“	“	“	'2'	'0'	'0'

- DATA (8Bytes): 기능 추가 예정입니다. 반환값은 '0' 입니다.

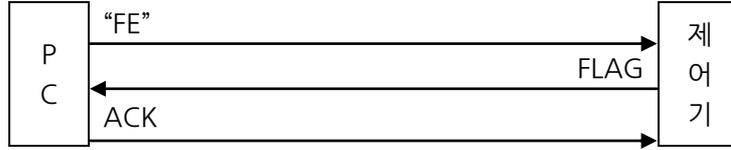
DATA(8 Bytes)							
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
“	“	“	“	“	“	“	‘0’

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.33 파일 삭제(FE)

제어기에 저장되어 있는 파일을 삭제하는 명령어 입니다.



● Protocol Data

PC	STX	Dummy(0xFF)	'F'	'E'	채널	TYPE	파일 이름	DATA	ETX	LRC
----	-----	-------------	-----	-----	----	------	-------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 파일 삭제를 원하는 로봇 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE(1Byte): 파일이 저장되어 있는 공간  
 0: Backup RAM  
 ※ 상기 이외의 값을 입력하면 FLAG(0x33)를 반환 합니다.  
 ※ N1 제어기는 저장 공간이 '0' 만을 지원합니다.
- 파일 이름: 삭제 할 파일 이름  
 EX) "RS"라는 JOB 파일을 삭제 할 경우

파일 이름 (12Bytes)								
1Byte	...	1Byte						
"	...	"	'R'	'S'	'.'	'J'	'O'	'B'

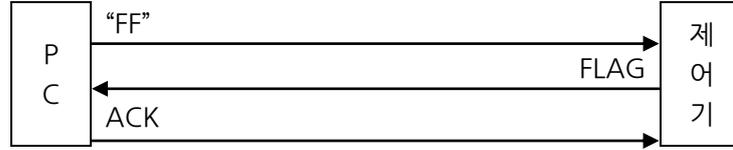
- ※ 현재 JOB파일만 삭제 가능 하며, PNT 파일 삭제 기능 추가 예정입니다.
- ※ 파일의 확장자가 JOB 또는 PNT가 아닌 경우 0x32 기능실행 FAIL"의 FLAG를 반환합니다.

- DATA(1Byte): 기능추가 예정 입니다.  
 '0'문자를 입력 하여 주십시오.
- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용 하여 읽기를 권장합니다.

### 4.34 파일 복사(FF)

동일 채널에서 JOB 파일을 복사하는 명령어 입니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'F'	'F'	채널 1	TYPE	파일명 1	채널 2	파일명 2	ETX	LRC
----	-----	-------------	-----	-----	------	------	-------	------	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널 1(0~2): 복사 대상 로봇채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE(1Byte): 파일 저장위치  
 0: Backup RAM  
 ※ 상기 이외의 값을 입력하면 FLAG(0x33)를 반환 합니다.  
 ※ N1 제어기는 저장 공간이 '0' 만을 지원합니다.

- 파일명 1: 복사 대상 파일명  
 EX) "RS"라는 JOB 파일을 복사 하는 경우

파일 이름 (12Bytes)								
1Byte	...	1Byte						
'	...	'	'R'	'S'	'.'	'J'	'O'	'B'

※ 파일 확장자가 JOB 혹은 PNT가 아닌 경우 0x32 기능실행 FAIL"의 FLAG를 반환합니다.

- 채널 2(0~2): 복사 하여 새로 만들 파일 로봇채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.

- 파일명 2: 복사한 내용을 붙여넣을 파일명  
EX) "TT.JOB" 으로 복사할 경우

파일 이름 (12Bytes)								
1Byte	· · ·	1Byte						
“	· · ·	“	‘T’	‘T’	‘.’	‘J’	‘O’	‘B’

※ 파일 확장자가 JOB 혹은 PNT가 아닌 경우 0x32 기능실행 FAIL”의 FLAG를 반환합니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

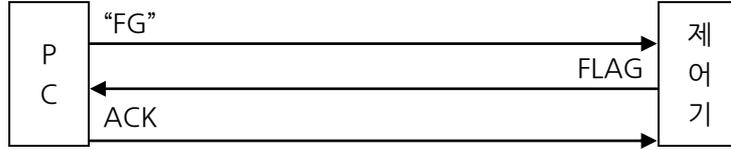


**CAUTION**

- ▶ JOB 파일 복사 기능은 **동일 로봇 채널에서 만 복사를 지원합니다.**
- ▶ 복사 대상 로봇 채널과 새로 만들 로봇 채널이 동일하지 않을 경우 RUNFAIL 알람이 발생 합니다

### 4.35 파일 이름 변경(FG)

JOB 파일 이름을 변경하는 명령어 입니다.



- Protocol Data

PC     

STX	Dummy(0xFF)	'F'	'G'	채널	TYPE	파일 이름 1	파일 이름 2	ETX	LRC
-----	-------------	-----	-----	----	------	---------	---------	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-------------	------	-----	-----

PC     

ACK
-----

- 채널 (0~2): 파일 이름을 변경 하고자 하는 로봇 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- TYPE(1Byte): 파일 저장위치  
 0: Backup RAM  
 ※ 상기 이외의 값을 입력하면 FLAG(0x33)를 반환 합니다.  
 ※ N1 제어기는 저장 공간이 '0' 만을 지원합니다.

- 파일명 1: 파일 이름을 변경할 파일명  
 EX) "RS"라는 JOB 파일 이름을 변경 할 경우

파일 이름 (13Bytes)									
1Byte	· · ·	1Byte							
“	· · ·	“	'R'	'S'	'·'	'J'	'O'	'B'	“(0x20)

- ※ 마지막 13번 Byte는 구분자로 활용 됩니다. 구분자 SPACE(0x20)가 없는 경우 되지 않습니다.
- ※ 파일 확장자가 JOB 또는 PNT가 아닌 경우 0x32 기능실행 FAIL”의 FLAG를 반환합니다.

- 파일명 2: 변경될 파일명  
EX) "RS"라는 JOB 파일을 T1 으로 이름을 변경할 경우

파일 이름 (12Bytes)								
1Byte	...	1Byte						
"	...	"	'T'	'1'	'.'	'J'	'O'	'B'

※ 파일 확장자가 JOB 또는 PNT가 아닌 경우 0x32 기능실행 FAIL"의 FLAG를 반환합니다.

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

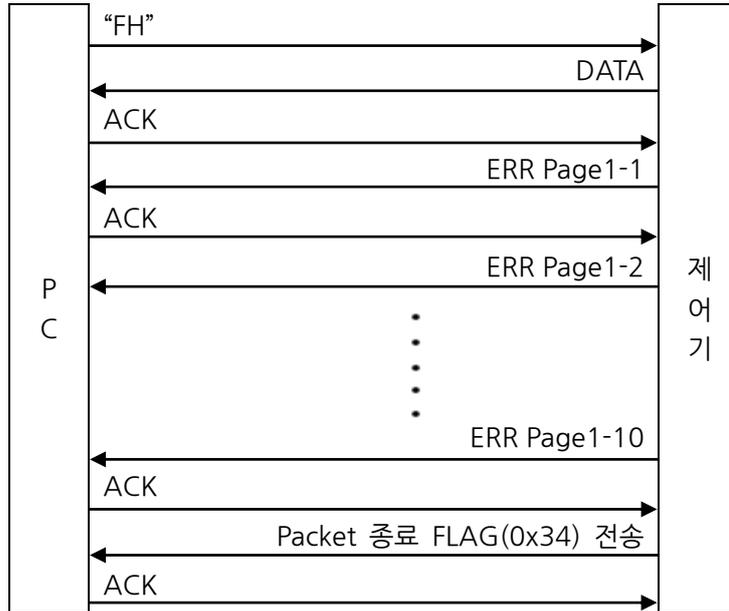


**CAUTION**

▶ 이름을 변경하고자하는 JOB파일이 해당 채널에 없을 경우 RUN FAIL 알람이 발생할 수 있습니다.

### 4.36 저장 되어 있는 알람 내용 읽기(FH)

N1 제어기에 저장되어 있는 알람 내용 읽기 명령어 입니다.



- Protocol Data

PC     

STX	Dummy(0xFF)	'F'	'H'	TYPE	파일 이름	ETX	LRC
-----	-------------	-----	-----	------	-------	-----	-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	DATA	ETX	LRC
-----	-------------	------------	------	-----	-----

PC     

ACK
-----

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ERR Page1-1	ETX	LRC
-----	-------------	------------	-------------	-----	-----

PC     

ACK
-----

⋮

제어기     

STX	Dummy(0xFF)	FLAG(0x30)	ERR Page10-10	ETX	LRC
-----	-------------	------------	---------------	-----	-----

PC     

ACK
-----

제어기     

STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-------------	------------	-----	-----

PC     

ACK
-----

- TYPE(1Byte): 파일 저장위치  
0: Backup RAM  
※ 상기 이외의 값을 입력하면 FLAG(0x33)를 반환 합니다.  
※ N1 제어기는 저장 공간이 '0' 만을 지원합니다.
- 파일이름(30Byte): ERR message 파일 요청

파일 이름 (30Bytes)
"alarm_history.txt"

※ 30Byte중 "alarm\_history.txt"부분을 제외한 빈(13Byte)공간은 SPACE(0x20) 문자를 삽입하여 주십시오.

- DATA(40Byte)

DATA (36Bytes)
"NO.₩tERROR TIME₩tCH ERROR MSG₩t(CODE) "

- ERR Page : 제어기에 저장되어 있는 ERR 내용을 Page당 최대 10개까지 반환 합니다. ERROR 내용에 따라 최대 76Byte까지 반환 합니다.

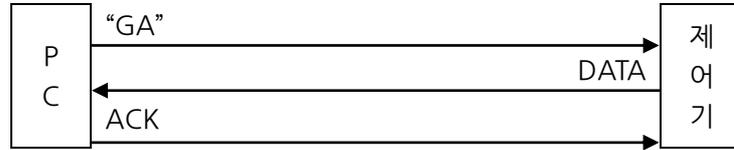
ERR Page(MAX 76Byte)									
"%02d₩t[%02d %02d:%02d:%02d]₩tCH%d - %s,%s₩t(%4d) "									
PAGE	INDEX	DAY	시	분	초	알람 채널	알람 내용	세부내용	알람 코드
"%02d	%02d	[%dD	%02d	:%02d	:%02d]	CH%d	- %s	,%s	(%4d) "

- PAGE: 알람 내용을 저장, 최근에 발생한 알람 순으로 저장
- INDEX: 알람 발생 시간이 동일한 알람 내용 저장
- 날짜, 시간, 분, 초: 제어기 WORK TIMER을 기준으로 저장됩니다.(WORK TIMER 에 대한 자세한 내용은 “조작 운용 설명서 1.3.4.1 W.TIMER” 참조 하시기 바랍니다.)
- 알람 채널: 알람이 발생한 채널을 표시합니다. 만약 '9'로 반환이 되면 전체 채널에 대한 알람이 발생 된것입니다.
- 알람 내용: 발생한 알람 내용 입니다.
- 세부내용: 알람 내용으로만 판단이 안되는 알람내용에 대하여 부가 설명을 나타냅니다.
- 알람 코드: 알람 내용에 대한 코드 번호 입니다.
- ※ ERR 내용 저장에 대한 자세한 내용은 “프로그램 설명서 3.41 RERROR” 명령어를 참고하시기 바랍니다.
- ※ ERR 원인이나 해결 방법은 “알람 코드 설명서”를 참고하시기 바랍니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.37 I/O 카드 IN PUT 접점 상태 정보 읽기(GA)

I/O카드 입력 접점 상태 정보 읽기 명령어 입니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'A'	입력 PORT	ETX	LRC
----	-----	-------------	-----	-----	---------	-----	-----

제어기	STX	Dumm(0xFF)	FLAG	PORT 접점 수	INPUT 상태 정보	ETX	LRC
-----	-----	------------	------	-----------	-------------	-----	-----

PC	ACK
----	-----

- 입력 PORT(0~4): 입력 PORT(2Byte)를 선택합니다.

입력 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- PORT 접점 수(2Byte): 선택한 입력 PORT 의 접점 수를 반환 합니다.

DATA1(2Bytes)	
1Byte	1Byte
'1'	'6'

- INPUT 상태 정보(4Byte): 선택한 입력 PORT 접점의 상태정보를 반환합니다.

INPUT 상태 정보(4Bytes)			
1Byte	1Byte	1Byte	1Byte

접점 상태 정보(0~7) (2Bytes)															
1Byte								1Byte							
0~3접점								4~7접점							
8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit
0	0	1	1	IN0	IN1	IN2	IN3	0	0	1	1	IN4	IN5	IN6	IN7

※ 접점 데이터는 ON:1 , OFF:0 입니다.

접점 상태 정보(8~15) (2Bytes)															
1Byte								1Byte							
8~11접점								12~15접점							
8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit
0	0	1	1	IN8	IN9	IN10	IN11	0	0	1	1	IN12	IN13	IN14	IN15

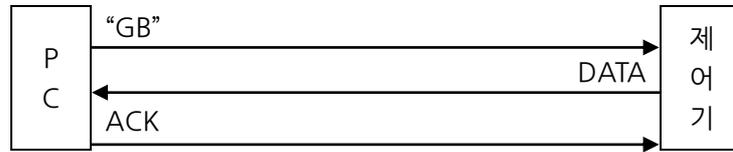
※ 접점 데이터는 ON:1 , OFF:0 입니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.38 I/O 카드 OUT PUT 접점 상태 정보 읽기(GB)

I/O카드 출력 접점 상태 정보 읽기 명령어 입니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'B'	출력 PORT	ETX	LRC
----	-----	-------------	-----	-----	---------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	PORT 접점 수	OUT 상태 정보	ETX	LRC
-----	-----	-------------	------	-----------	-----------	-----	-----

PC	ACK
----	-----

- 출력 PORT(0~4):출력 PORT를 선택합니다.

출력 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- PORT 접점수: 선택한 출력 PORT 의 접점 수를 반환 합니다.

DATA1(2Bytes)	
1Byte	1Byte
'1'	'6'

- OUT 상태 정보: 선택한 출력 PORT 접점의 상태정보를 반환합니다.

OUT 상태 정보(4Bytes)			
1Byte	1Byte	1Byte	1Byte

접점 상태 정보(0~7) (2Bytes)															
1Byte								1Byte							
0~3접점								4~7접점							
8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit
0	0	1	1	OUT0	OUT1	OUT2	OUT3	0	0	1	1	OUT 4	OUT 5	OUT 6	OUT 7

※ 접점 데이터는 ON:1 , OFF:0 입니다.

접점 상태 정보(8~15) (2Bytes)															
1Byte								1Byte							
8~11접점								12~15접점							
8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit	8bit	7bit	6bit	5bit	4bit	3bit	2bit	1bit
0	0	1	1	OUT8	OUT9	OUT10	OUT11	0	0	1	1	OUT12	OUT13	OUT14	OUT15

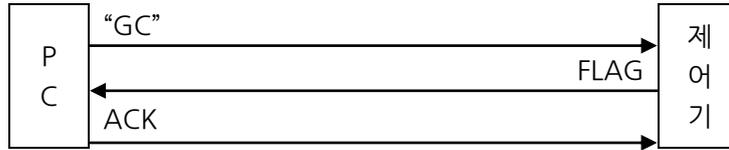
※ 접점 데이터는 ON:1 , OFF:0 입니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

4.39 I/O 카드 OUT PUT 접점 출력(GC)

I/O 접점 Data 쓰기 명령어 입니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'C'	출력 PORT	접점 번호	DATA	ETX	LRC
----	-----	-------------	-----	-----	---------	-------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 출력 PORT(0~4): 출력 PORT를 선택합니다.

출력 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- 접점 번호(0~15): 출력 접점을 선택합니다.

접점 번호(2Bytes)	
1Byte	1Byte
'0'	'1'

- DATA: 접점 출력 상태를 선택합니다.  
0: 접점 OFF  
1: 접점 ON

DATA
1 Byte
'0'or'1'

EX) 1번 PORT의 12번 출력 접점을 ON 상태로 만들 경우

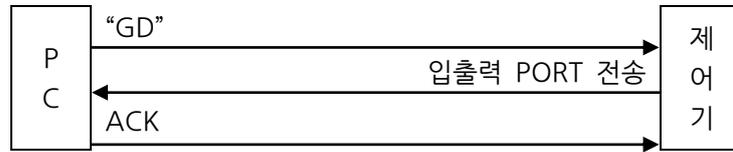
STX	Dummy	'G'	'C'	출력 PORT		접점 번호		DATA	ETX	LRC
				'0'	'1'	'1'	'2'	'1'		

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

#### 4.40 I/O 카드 정보 읽기 (GD)

제어기에 설정 되어 있는 입/출력 BOARD 설정 값을 읽는 명령어 입니다.



- Protocol Data

PC

STX	Dummy(0xFF)	'G'	'D'	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	입력 BOARD	출력 BOARD	ETX	LRC
-----	-------------	------	----------	----------	-----	-----

PC

ACK
-----

- 입력 BOARD: 제어기 I/O BOARD 설정 값(2Byte)

입력 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

제어기에 설정되어 있는 I/O BOARD 개수를 알려 줍니다.

- 출력 BOARD: 제어기 I/O BOARD 설정 값(2Byte)

출력 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

제어기에 설정되어 있는 I/O BOARD 개수를 알려 줍니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

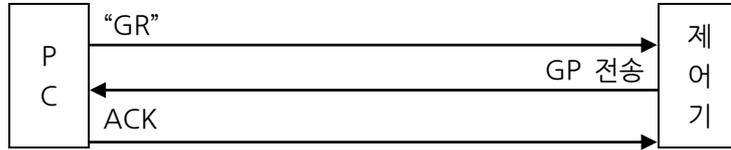
CAUTION

- ▶ N1 제어기 I/O Board의 개수는 0~2까지 설정 가능 합니다.
- ▶ “조작 및 운용 설명서 1.3.1.3 확장 I/O 보드 개수 설정” 참고 하시기 바랍니다.

### 4.41 글로벌 변수 Data 읽기(GR)

글로벌 변수(I, F, GP)에 Data읽기 명령어 입니다.

#### 4.41.1 글로벌 포인트 읽기



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'R'	타입	채널	좌표계	축 번호	INDEX	ETX	LRC
----	-----	-------------	-----	-----	----	----	-----	------	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	'0'	Global Point Data	ETX	LRC
-----	-----	-------------	------	-----	-------------------	-----	-----

PC	ACK
----	-----

- 타입(0~2): Global 변수(I, F, GP)를 선택합니다.  
**0: Global Point**  
 1: Global Integer  
 2: Global Float
- 채널: 동작중인 로봇채널을 선택합니다.  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 좌표계: 받고자 하는 Data 타입을 선택합니다.  
 0: Angle Data  
 1: XY좌표 Data
- 축 번호  
 0: 모든 축(1~6축)을 선택합니다.  
 1: 1축(X)을 선택 합니다.  
 2: 2축(Y)을 선택 합니다.  
 3: 3축(Z)을 선택 합니다.  
 4: 4축(W)을 선택 합니다.  
 5: 5축(E1)을 선택 합니다.  
 6: 6축(E2)을 선택 합니다.

- INDEX: Global Point 번호 입니다. 0~1023까지 입력 가능 합니다.

INDEX(5Bytes)				
1Byte	1Byte	1Byte	1Byte	1Byte
‘ ‘	‘ ‘	‘ ‘	‘4’	‘5’

EX) “45” 입력시 10000자리 1000자리와 100자리에는 SPACE(0x20) 문자를 삽입하여 주십시오.

- Global Point Data: 반환 되는 Data는 다음과 같습니다.

- 축 번호가 0인 경우

6축에 해당하는 Data(6 X 10Byte+USED+ARM=62Byte)를 반환 합니다

DATA(MAX 62Bytes)							
1축 (10Bytes)	2축 (10Bytes)	3축 (10Bytes)	4축 (10Bytes)	5축 (10Bytes)	6축 (10Bytes)	USED	ARM

- 축 번호가 1~6인 경우

선택 축에 해당되는 Data(1 X 10Byte+USED=11Byte)를 반환 합니다.

DATA(11Bytes)										
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
‘1’	‘2’	‘3’	‘4’	‘5’	‘.’	‘1’	‘2’	‘3’	0x20(‘ ’)	USED

- USED: Point Data 사용 유무를 결정합니다.

0: Point Data 사용 안함

1: Point Data 사용 가능

- ARM(0~2):

0: LEFT\_form

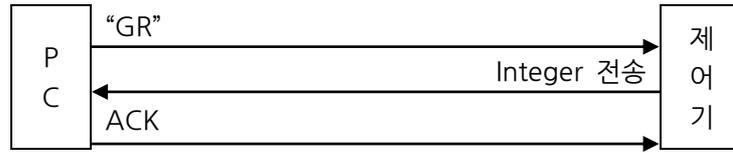
1: RIGHT\_form

2: No\_form

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

4.41.2 글로벌 정수 읽기



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'R'	타입	INDEX	ETX	LRC
----	-----	-------------	-----	-----	----	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	Integer Data	ETX	LRC
-----	-----	-------------	------	--------------	-----	-----

PC	ACK
----	-----

- 타입(0~2): Global 변수(I, F, GP)를 선택합니다.  
 0: Global Point  
1: Global Integer  
 2: Global Float
- INDEX: 0~499까지 입력 가능 합니다.  
 EX) "45" 입력시 1000자리와 100자리에는 SPACE(0x20) 문자를 삽입하여 주십시오.

INDEX(4Bytes)			
1Byte	1Byte	1Byte	1Byte
' '	' '	'4'	'5'

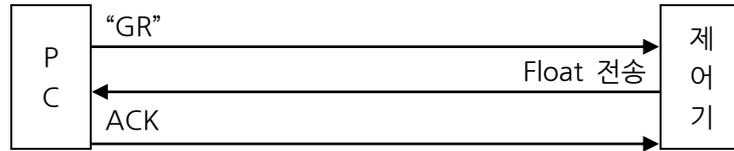
- DATA: Global Integer Data를 반환 합니다.

DATA(6Bytes)					
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'1'	'2'	'3'	'4'	'5'	'6'

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

4.41.3 글로벌 실수 읽기



- Protocol Data

PC

STX	Dummy(0xFF)	'G'	'R'	타입	INDEX	ETX	LRC
-----	-------------	-----	-----	----	-------	-----	-----

제어기

STX	Dummy(0xFF)	FLAG	Float Data	ETX	LRC
-----	-------------	------	------------	-----	-----

PC

ACK
-----

- 타입(0~2): Global 변수(I, F, GP)를 선택합니다.  
 0: Global Point  
 1: Global Integer  
**2: Global Float**
- INDEX: 0~499까지 입력 가능 합니다.  
 EX) "45" 입력시 1000자리와 100자리에는 SPACE(0x20) 문자를 삽입하여 주십시오.

INDEX(4Bytes)			
1Byte	1Byte	1Byte	1Byte
' '	' '	'4'	'5'

- DATA: Global Float Data를 반환 합니다.

DATA(9Bytes)								
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'1'	'2'	'3'	'4'	'5'	'.'	'1'	'2'	'3'

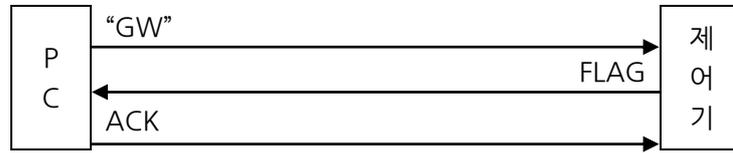
- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

### 4.42 글로벌 변수 Data 쓰기(GW)

글로벌 변수(I, F, GP)에 Data쓰기 명령어 입니다.

#### 4.42.1 글로벌 포인트 저장



- Protocol Data

PC     STX | Dummy(0xFF) | 'G' | 'W' | 타입 | 채널 | 좌표계 | 축 번호 | INDEX | DATA | ETX | LRC

제어기.     STX | Dummy(0xFF) | FLAG | ETX | LRC

PC     ACK

- 타입(0~2): Global 변수(I, F, GP)를 선택합니다.  
0: Global Point  
 1: Global Integer  
 2: Global Float
- 채널: 저장할 로봇채널을 선택합니다.  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- 좌표계: 제어기에 전송하고자 하는 Data 타입을 선택합니다.  
 0: Angle Data  
 1: XY좌표 Data  
 ※ 일반 직각 로봇은 Angle좌표계 선택하여도 XY좌표계로 저장됩니다.
- 축 번호: 저장하고자 하는 축을 선택합니다.  
 0: 모든 축(1~6축)을 선택합니다.  
 1: 1축(X)을 선택 합니다.  
 2: 2축(Y)을 선택 합니다.  
 3: 3축(Z)을 선택 합니다.  
 4: 4축(W)을 선택 합니다.  
 5: 5축(E1)을 선택 합니다.  
 6: 6축(E2)을 선택 합니다.

- INDEX: Global Point Index번호 입니다. 0~1023까지 입력 가능 합니다.  
EX) “45” 입력시 10000자리1000자리와 100자리에는 SPACE(0x20) 문자를 입력하여 주십시오.

INDEX(5Bytes)				
1Byte	1Byte	1Byte	1Byte	1Byte
‘ ‘	‘ ‘	‘ ‘	‘4’	‘5’

- DATA: 저장하고자 하는 Data를 입력합니다.  
- 축 번호가 0인 경우  
6축에 해당하는 Data(6 X 10Byte+USED+ARM=62Byte)를 전송합니다

DATA(MAX 62Bytes)							
1축(10 Byte)	2축(10 Byte)	3축(10 Byte)	4축(10 Byte)	5축(10 Byte)	6축(10 Byte)	USED	ARM

만약 Data 입력 하고자 하는 채널이 4축일 경우 5축,6축에 “0.0”을 입력 합니다.

- 축 번호가 1~6인 경우  
선택 축에 해당되는 Data(1 X 10Byte+USED+ARM=10Byte)를 전송합니다.

DATA(12Bytes)											
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
‘1’	‘2’	‘3’	‘4’	‘5’	‘.’	‘1’	‘2’	‘3’	0x20(‘ ’)	USED	ARM

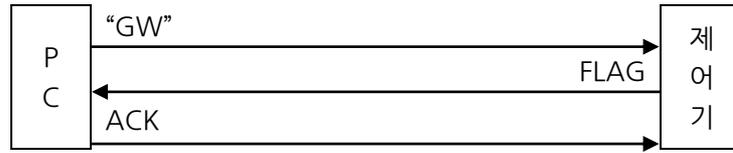
- USED: Point Data 사용 유/무를 결정합니다.  
0: Point Data 사용 안함  
1: Point Data 사용 가능

- ARM(0~2):  
0: LEFT\_form  
1: RIGHT\_form  
2: 로봇의 현재 자세를 저장합니다.

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

4.42.2 글로벌 정수 저장



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'W'	타입	INDEX	Data	ETX	LRC
----	-----	-------------	-----	-----	----	-------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 타입(0~2): Global 변수(I, F, GP)를 선택합니다.  
 0: Global Point  
**1: Global Integer**  
 2: Global Float
- INDEX: 0~499까지 입력 가능 합니다.  
 EX) "45" 입력시 1000자리와 100자리에는 SPACE(0x20) 문자를 삽입하여 주십시오.

INDEX(4Bytes)			
1Byte	1Byte	1Byte	1Byte
' '	' '	'4'	'5'

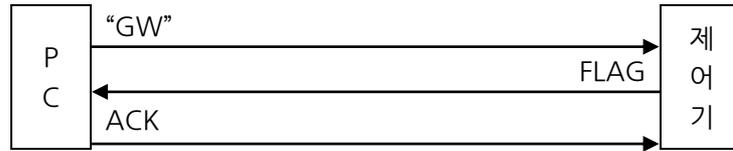
- DATA: 입력 가능한 정수 Data길이는 6Bytes 입니다.

DATA(6Byte)					
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'1'	'2'	'3'	'4'	'5'	'6'

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

4.42.3 글로벌 실수 저장



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'W'	TYPE	INDEX	Data	ETX	LRC
----	-----	-------------	-----	-----	------	-------	------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- TYPE(0~2): Global 변수(I, F, GP)를 선택합니다.  
 0: Global Point  
 1: Global Integer  
**2: Global Float**
- INDEX: 0~499까지 입력 가능 합니다.  
 EX) "45" 입력시 1000자리와 100자리에는 SPACE(0x20) 문자를 입력하여 주십시오.

INDEX(4Bytes)			
1Byte	1Byte	1Byte	1Byte
' '	' '	'4'	'5'

- DATA(10Byte): 실수형 Data로 소수점 셋째 자리까지 입력 가능합니다.

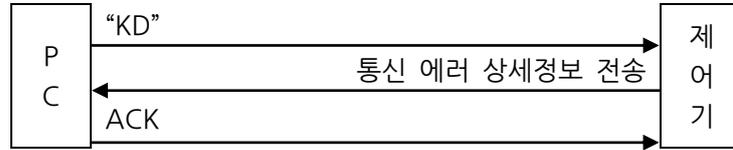
DATA(10Byte)									
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'1'	'2'	'3'	'4'	'5'	'6'	'.'	'1'	'2'	'3'

- STX, ETX, LRC, FLAG: "4.1통신 규칙"을 참조 하시기 바랍니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 "통신에러 원인읽기(KD)"프로토콜을 이용하여 읽기를 권장합니다.

### 4.43 통신 에러 원인 읽기(KD)

기능실패(0x32)에 대한 에러 내용을 확인 명령어 입니다.



- Protocol Data

PC 

STX	Dummy(0xFF)	'K'	'D'	ETX	LRC
-----	-------------	-----	-----	-----	-----

제어기 

STX	Dummy(0xFF)	FLAG	Text	ETX	LRC
-----	-------------	------	------	-----	-----

PC 

ACK
-----

- Text: 통신 에러 원인에 대한 상세 내용을 문자열로 전송 합니다.  
EX) LRC값이 다른 경우 다음과 같이 출력이 됩니다.

Text
"LRC is different with received data LRC"

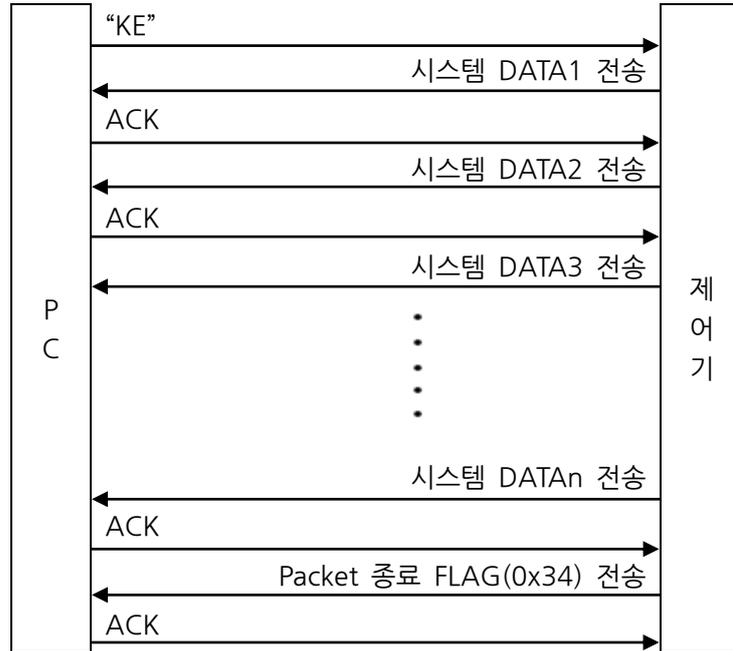
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.



**CAUTION**

▶ 통신에러 테이블은 다 채널간 공용으로 사용 하기 때문에 마지막 발생된 에러 메시지를 반환합니다.

4.44 시스템 DATA 읽기(KE)



● Protocol Data

PC	STX	Dummy(0xFF)	'K'	'E'	채널	INDEX	ETX	LRC
----	-----	-------------	-----	-----	----	-------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	DATA1	ETX	LRC
-----	-----	-------------	------	-------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG	DATA2	ETX	LRC
-----	-----	-------------	------	-------	-----	-----

⋮

제어기	STX	Dummy(0xFF)	FLAG	DATA <sub>n</sub>	ETX	LRC
-----	-----	-------------	------	-------------------	-----	-----

PC	ACK
----	-----

제어기	STX	Dummy(0xFF)	FLAG(0x34)	ETX	LRC
-----	-----	-------------	------------	-----	-----

PC	ACK
----	-----

- 채널:
  - 0: 채널 1번 PARA Data 요청
  - 1: 채널 2번 PARA Data 요청
  - 2: 채널 3번 PARA Data 요청
  - 4: PUB\_PARA Data 요청
- INDEX(2Byte): Parameter DATA 선택

INDEX(2Bytes)	
1Byte	1Byte
' '	'4'

※ 해당 채널이 BGT 타입으로 설정되어 있을 경우 INDEX 와 상관 없이 BGT PARA 내용만 반환 합니다.

- Parameter Data 개별 요청시 아래 상수 테이블을 참조하시기 바랍니다.

PARA				PUB_PARA	
인자	내용	인자	내용	인자	내용
-1	모든 Data 요청	52	FOW	-1	모든 Data 요청
1	CONF	53	OVS	1	TMR
2	LENG	54	OVT	2	HOST-S
3	RANG	55	OVA	3	FDBUS
4	OFFS	56	IPE	4	DIO
5	OFF IMPULSE	57	IPA	7	DEADMAN
6	JNT	58	TRQ	8	AUTO SVON
7	TOOL	59	OVL	9	PLC
8	MOTOR	61	OND	10	MAX_CH
9	ENCTY	62	TOL	12	A I/O
10	HALL SENSOR	63	DB	51	W.TIMER
11	MECH END	71	RDIS	52	CINF
12	MTYPE	72	USING AXIS	53	PASSWORD
21	ENCTY	73	XENB	54	NAME
22	JOINT	74	P/R		
23	LINR	75	IRNG		
24	DISP	76	IRNG		
25	MOVE	77	DUAL-SYNC		
26	SENS	91	S_VEL		
27	LIMIT	92	SEQ		
31	INIT_VEL	93	DIR		
32	DEC_ZONE	94	RULE		
33	STOP_TIME	95	SENSOR		
41	PVG	97	S_TYPE		
42	PVG2	98	HOME		
43	SMOOTHING FILTER				
51	SET				

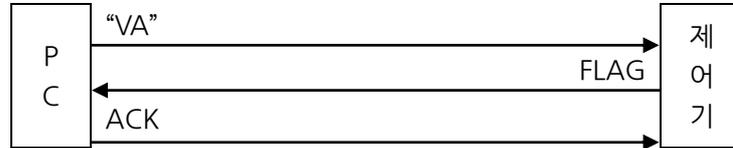
※ 위 PARA 와 PUB\_PARA의 자세한 내용은 “조작운영 설명서”를 참조하시기 바랍니다.

- DATA: 요청한 내용에 따라 DATA 길이는 변경 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.45 Vision Point 저장(VA)

Vision Point DATA를 제어기로 전송하기 위한 명령어 입니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'V' 'A' 채널	Reserved	좌표계	Vision Point	ARM	VAL	ETX	LRC
----	-----	-------------	------------	----------	-----	--------------	-----	-----	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- Reserved: '0'으로 고정 하여 사용
- 좌표계(0,1) : 좌표계 설정(1Byte)  
 0: Angle 좌표: 도(Degree)단위의 좌표.  
 1: XY 좌표: mm단위의 좌표.  
 ※ 이 외의 값 입력하면 "0x31 프로토콜 ERROR"의 FLAG가 발생합니다.
- Vision Point : Vision Point DATA

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- ※ MAX AXIS 보다 위치 DATA가 적을 경우 원하는 동작이 안될수 있습니다.
- ※ 현재 로봇의 MAX AXIS를 모르는 경우 "제어기 INFO 읽기(AD)"를 이용하여 MAX AXIS 확인후 위치 DATA 값을 입력하십시오.

EX) 4축 로봇의 경우: 4축 × 10Bytes = 40Bytes)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	4축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- 좌표계 설정값이 XYZ TYPE일 경우

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

X 위치 DATA(10bytes)				Y 위치 DATA(10bytes)				...	W 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- ARM(0~2):  
0: LEFT\_form  
1: RIGHT\_form  
2: 로봇의 현재 자세를 저장 합니다.
- VAL('0'~'9'): 대상물의 양/불 판정값 혹은 대상물의 종류로 사용  
- JOB 명령어 VVAL(시스템 변수)로 반환 합니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

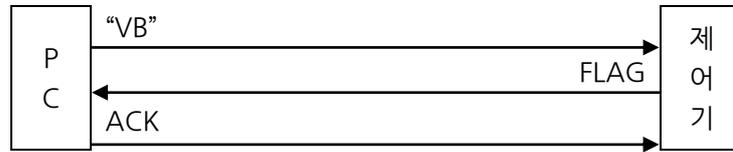
※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.



**CAUTION**

▶ 로봇 타입이 SCARA일 경우 XY 좌표계를 지원합니다.

4.46 대상물의 양/불 판정값 전송(VB)



- Protocol Data

PC	STX	Dummy(0xFF)	'V'	'B'	채널	VAL	ETX	LRC
----	-----	-------------	-----	-----	----	-----	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

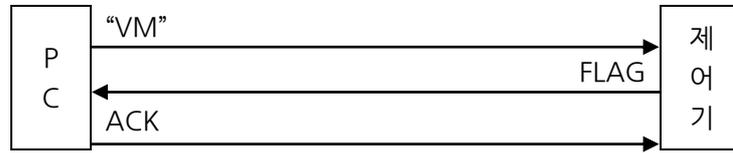
PC	ACK
----	-----

- 채널(0~2): 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.
- VAL('0'~'9'): 대상물의 양/불 판정값 혹은 대상물의 종류로 사용  
 - JOB 명령어 VVAL(시스템 변수)로 반환 합니다.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

#### 4.47 Vision Point DATA 연속 저장(VM)

Vision Point DATA값을 최대 3개까지 한번에 전송 합니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'V'	'M'	채널	ARM	Vision Point 1	...	Vision Point 3	ETX	LRC
----	-----	-------------	-----	-----	----	-----	----------------	-----	----------------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- ARM(0~2):  
 0: LEFT\_form  
 1: RIGHT\_form  
 2: 로봇의 현재 자세를 저장 합니다.
- Vision Point : Vision Point DATA는 최대 3개까지 한번에 전송 가능 합니다.  
 - Vision Point DATA는 XY 좌표계 DATA만 전송 가능 합니다.

Index	1축 위치 DATA(10bytes)			2축 위치 DATA(10bytes)			...	6축 위치 DATA(10bytes)			VAL
1Byte	1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	1Byte

- Index : Vision Point Data 저장 Point 번호 입니다.  
 Index 값으로 Hex코드로 0x30('0') ~ 0x62('b')까지 입력 가능 합니다.
- VAL('0'~'9'): 대상물의 양/불 판정값 혹은 대상물의 종류로 사용  
 JOB 명령어 VVAL(시스템 변수)로 반환 합니다.

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

※ MAX AXIS 보다 위치 DATA가 적을 경우 원하는 동작이 안될수 있습니다.

※ 현재 로봇의 MAX AXIS를 모르는 경우 “제어기 INFO 읽기(AD)”를 이용하여 MAX AXIS 확인후 위치 DATA 값을 입력하십시오.

EX) 4축 로봇의 경우: 4축 × 10Bytes = 40Bytes)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	4축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

X 위치 DATA(10bytes)				Y 위치 DATA(10bytes)				...	W 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

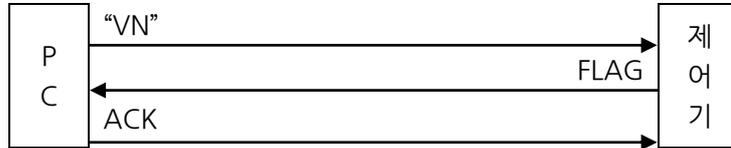


**CAUTION**

- ▶ Vision Point DATA는 XY좌표계(mm) 단위로 입력 하시기 바랍니다. Angle 좌표계로 입력시 원하는 동작이 안될수 있습니다.

#### 4.48 Vision Point 증분 DATA 저장(VN)

Vision Point DATA를 증분 값으로 전송 합니다.



- Protocol Data

PC	STX	Dummy(0xFF)	'V'	'N'	채널	ARM	Vision Point1	...	Vision Point 3	ETX	LRC
----	-----	-------------	-----	-----	----	-----	---------------	-----	----------------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): 이동 할 로봇의 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- ARM(0~2):  
 0: LEFT\_form  
 1: RIGHT\_form  
 2: 로봇의 현재 자세를 저장합니다.
- Vision Point : Vision Point DATA는 최대 3개까지 한번에 전송 가능 합니다.  
 - Vision Point DATA는 XY 좌표계 DATA만 전송 가능 합니다.

Index	1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)				VAL
1Byte	1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	1Byte	1Byte

- Index : Vision Point Data 저장 Point 번호 입니다.  
 Index 값으로 Hex코드로 0x30('0') ~ 0x62('b')까지 입력 가능 합니다.
- VAL('0'~'9'): 대상물의 양/불 판정값 혹은 대상물의 종류로 사용  
 JOB 명령어 VVAL(시스템 변수)로 반환 합니다.

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	6축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

※ MAX AXIS 보다 위치 DATA가 적을 경우 원하는 동작이 안될수 있습니다.

※ 현재 로봇의 MAX AXIS를 모르는 경우 “제어기 INFO 읽기(AD)”를 이용하여 MAX AXIS 확인후 위치 DATA 값을 입력하십시오.

EX) 4축 로봇의 경우: 4축 × 10Bytes = 40Bytes)

1축 위치 DATA(10bytes)				2축 위치 DATA(10bytes)				...	4축 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

각 축마다 10Bytes씩 위치 DATA를 가집니다.(10Bytes×MAX AXIS)

X 위치 DATA(10bytes)				Y 위치 DATA(10bytes)				...	W 위치 DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

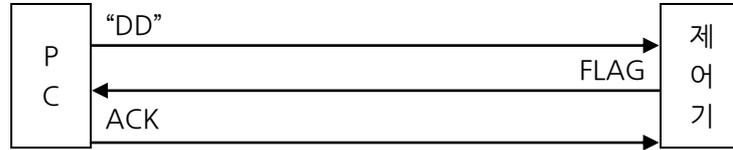


**CAUTION**

- ▶ Vision Point DATA는 XY좌표계(mm) 단위로 입력 하시기 바랍니다. Angle(도) 좌표계로 입력시 원하는 동작이 안될수 있습니다.
- ▶ 스카라 로봇의 경우 전송한 Vision Point DATA 값이 직각좌표계 값이므로, MOV 명령어 사용시 IMOV2를 이용해야 합니다.

### 4.49 MOTOR Brake ON/OFF(DD)

해당 로봇 채널의 Motor Brake를 ON/OFF시키는 명령어 입니다.



- Protocol DATA

PC	STX	Dummy(0xFF)	'D'	'D'	채널	Axis	Brake on/off	Password	ETC	LRC
----	-----	-------------	-----	-----	----	------	--------------	----------	-----	-----

제어기	STX	Dummy(0xFF)	FLAG	ETX	LRC
-----	-----	-------------	------	-----	-----

PC	ACK
----	-----

- 채널(0~2): Motor Brake ON/OFF 할 로봇 채널(1Byte)  
 0: 로봇 채널1에 해당 됩니다.  
 1: 로봇 채널2에 해당 됩니다.  
 2: 로봇 채널3에 해당 됩니다.
- AXIS(0~5): 로봇 채널에 속한 로봇의 Axis 번호를 입력 합니다.  
 0: 1(X)축 선택  
 1: 2(Y)축 선택  
 2: 3(Z)축 선택  
 3: 4(W)축 선택  
 4: 5(EX1)축 선택  
 5: 6(EX2)축 선택

※ 이 외의 값을 입력하면 “0x31 프로토콜 ERROR”의 FLAG가 발생합니다.

- Brake on/off: 선택한 축의 Motor Brake on/off를 선택합니다.

Brake ON/OFF
1Byte
'0'

- 0: 선택 Axis의 Motor Brake OFF
- 1: 선택 Axis의 Motor Brake ON

※ 이 외의 값을 입력하면 “0x31 프로토콜 ERROR”의 FLAG가 발생합니다.

- Password(4Byte): 제어기에 설정되어 있는 BRAKE Password와 동일한 값을 입력합니다.

Password(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'0'	'0'	'0'

제어기 초기 Password는 “0000” 입니다.

Password 변경 방법은 조작운용 설명서의 “PASSWORD 변경 모드”를 참조하시기 바랍니다.

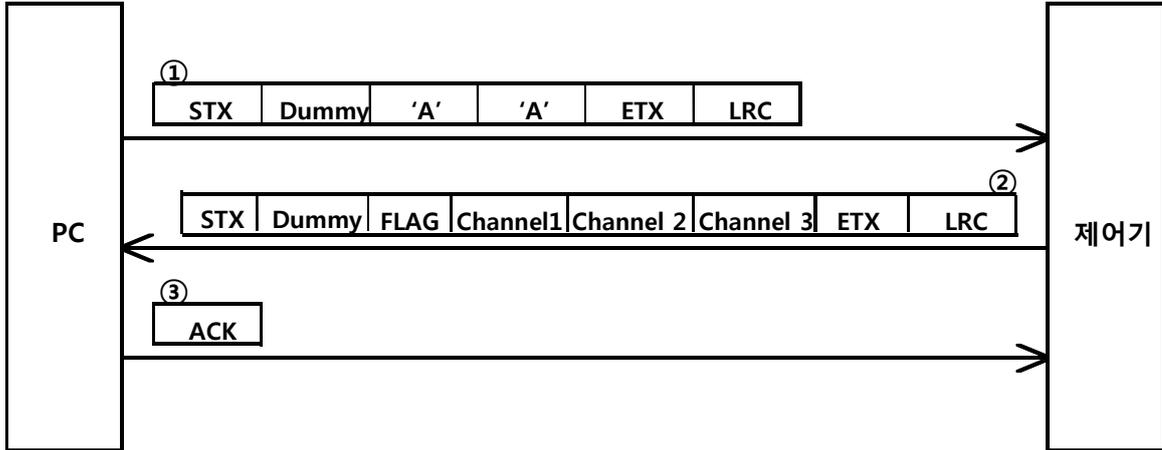
- STX, ETX, LRC, FLAG: “4.1통신 규칙”을 참조 하십시오.

※ Protocol 명령 후 통신에러 발생시 이에 대한 원인을 “통신에러 원인읽기(KD)”프로토콜을 이용하여 읽기를 권장합니다.

### 4.50 통신 프로토콜 에러 처리

#### 4.50.1 정상적인 통신 절차

다음은 정상적인 통신 절차 상황에 대한 설명입니다.

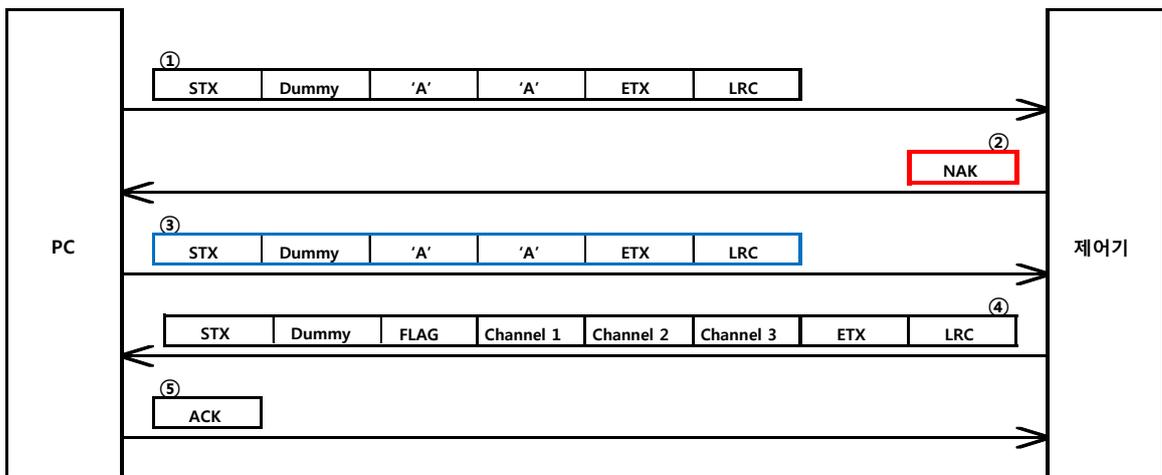


< 로봇 상태 정보 읽기(AA) >

- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② PC로 부터 요청 받은 로봇 상태 정보값을 전송한다.
- ③ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.

#### 4.50.2 PC to Controller 통신 에러

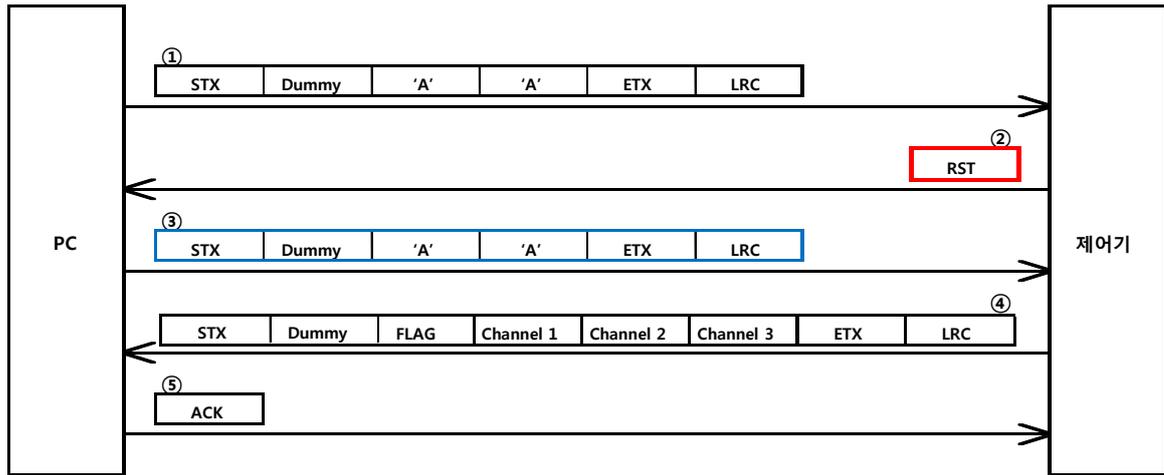
- 1) PC로부터 수신 받은 명령 프로토콜 LRC 이상 발생 시



< 로봇 상태 정보 읽기(AA) >

- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② 수신 받은 데이터 명령에 대한 LRC 이상으로 NAK(0x15) 응답을 전송한다.
- ③ NAK(0x15) 응답 수신 후 제어기에게 로봇 상태 정보를 재 요청한다.
- ④ 로봇 상태 정보값을 전송한다.
- ⑤ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.

2) PC로부터 수신 받은 명령 프로토콜 STX, ETX 데이터 이상 발생 시



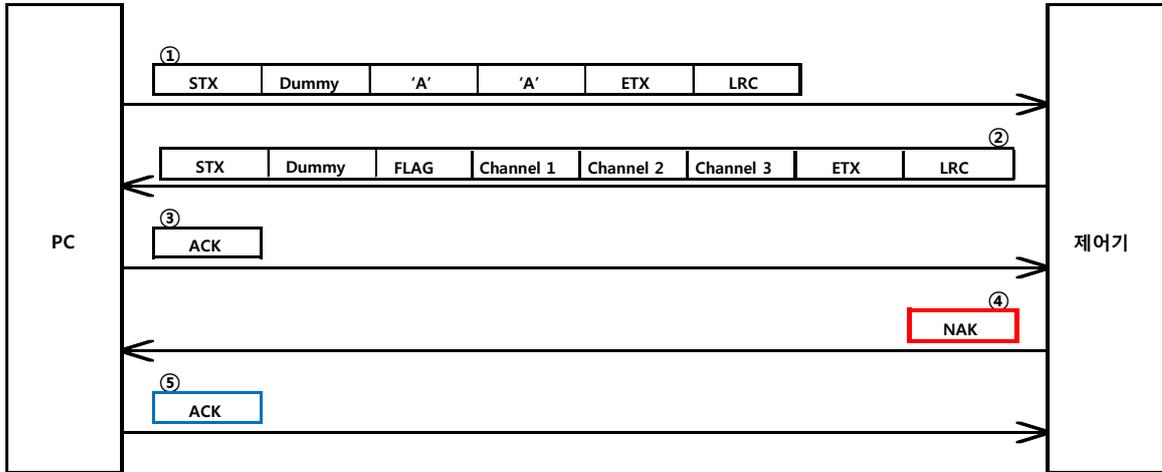
< 로봇 상태 정보 읽기(AA) >

- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② 수신 받은 데이터 중 STX 또는 ETX 패킷 손실 발생 시 일정 시간 대기 후 RST(0x12) 패킷을 전송한다.
- ③ RST(0x12) 응답 수신 후 제어기에게 로봇 상태 정보를 재 요청한다.
- ④ 로봇 상태 정보값을 전송한다.
- ⑤ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.

**CAUTION**

- ▶ NAK 및 RST 응답에 대한 자세한 사항은 “통신 에러 원인 읽기(KD)”를 통해 확인 하실 수 있습니다.
- ▶ RST 응답의 경우 수신 받은 패킷에 대한 처리 및 판단을 하기 위해 1초 이상 지연 대기 후 전송 됩니다.
- ▶ RST 명령은 통신 TIMEOUT 및 NAK 수행(3회) 초과 시 전송 됩니다.

3) ACK 응답 수신 이상 발생 1



< 로봇 상태 정보 읽기(AA) >

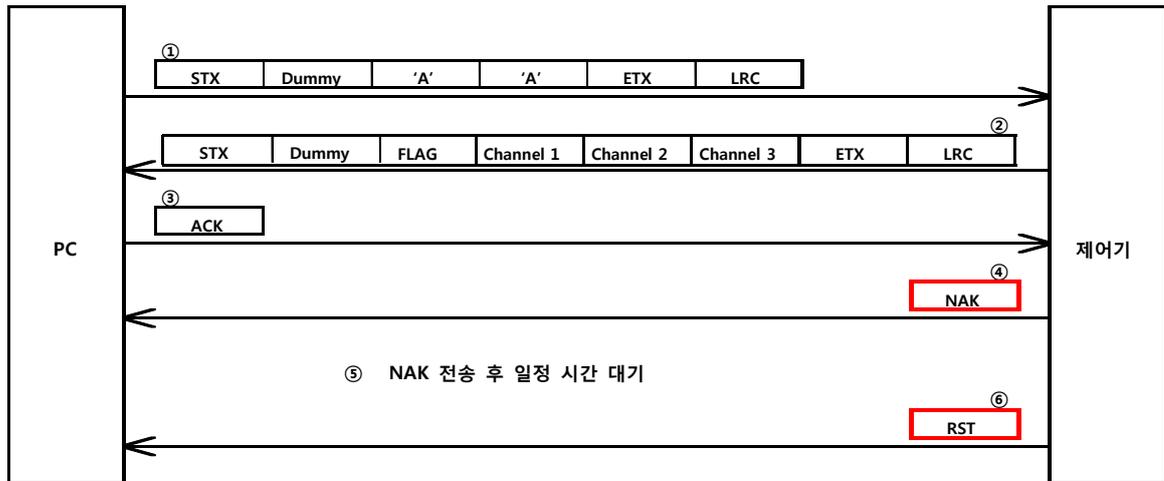
- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② PC로 부터 요청 받은 로봇 상태 정보값을 전송한다.
- ③ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.
- ④ 수신 받은 ACK(0x06) 패킷 손실 발생 시 NAK(0x15) 응답을 전송한다.
- ⑤ 제어기에게 ACK(0x06) 응답을 재 전송한다.



**CAUTION**

- ▶ 수신 완료 상태에 대한 ACK 대기 시간은 대략 5초 이상이며, ACK 응답 수신 시 즉시 대기 상태를 벗어 납니다.
- ▶ PC로부터 수신 받은 ACK 패킷 이상 발생 시 제어기는 NAK 응답을 전송하며, NAK 응답 전송 후 1초 이상 ACK 패킷에 대한 응답 대기를 수행합니다. 만약 응답 대기 시간동안 ACK 패킷 미수신시 RST 패킷을 전송 합니다.
- ▶ RST 명령은 통신 TIMEOUT 및 NAK 수행(3회) 초과 시 전송 됩니다.

4) ACK 응답 수신 이상 발생 2



< 로봇 상태 정보 읽기(AA) >

- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② PC로 부터 요청 받은 로봇 상태 정보값을 전송한다.
- ③ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.
- ④ 수신 받은 ACK(0x06) 패킷 손실 발생 시 NAK(0x15) 응답을 전송한다.
- ⑤ PC로부터 일정 시간 ACK(0x06) 응답을 대기한다.
- ⑥ 일정 시간 대기 후 ACK 응답 미 수신 시 RST(0x12) 패킷을 전송한다.

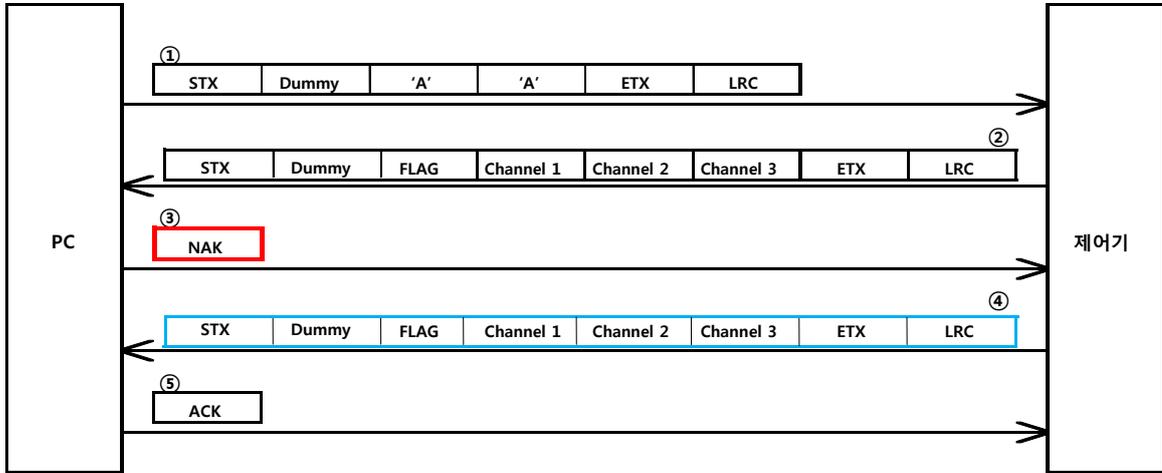


**CAUTION**

- ▶ 수신 완료 상태에 대한 ACK 대기 시간은 대략 5초 이상이며, ACK 응답 수신 시 즉시 대기 상태를 벗어 납니다.
- ▶ PC로부터 수신 받은 ACK 패킷 이상 발생 시 제어기는 NAK 응답을 전송하며, NAK 응답 전송 후 1초 이상 ACK 패킷에 대한 응답 대기를 수행합니다. 만약 응답 대기 시간동안 ACK 패킷 미 수신시 RST 패킷을 전송 합니다.
- ▶ RST 명령은 통신 TIMEOUT 및 NAK 수행(3회) 초과 시 전송 됩니다.

4.50.3 Controller to PC 통신 에러

1) 제어기로부터 수신 받은 명령 프로토콜 LRC 이상 발생 시



< 로봇 상태 정보 읽기(AA) >

- ① 제어기에게 로봇 상태 정보를 요청한다.
- ② PC로 부터 요청 받은 로봇 상태 정보값을 전송한다.
- ③ 제어기로부터 수신 받은 데이터 중 LRC 이상시 NAK(0x15)를 전송 한다.(최대 3회)
- ④ 로봇 상태 정보값을 재 전송한다.
- ⑤ 로봇 상태 정보값 수신 완료 후 ACK(0x06) 응답을 전송한다.

**CAUTION**

- ▶ 제어기로부터 수신 받은 데이터 이상 발생 시 PC는 NAK 명령을 통해 재전송 받을 수 있으며, 최대 3회까지 NAK 명령을 수행 할 수 있습니다. 만약 NAK 명령 3회 초과 시 제어기는 RST 명령을 전달 합니다.
- ▶ RST 명령은 통신 TIMEOUT 및 NAK 수행(3회) 초과 시 전송 됩니다.

Rev.	수정일자	내용	수정자	S/W Version
V.1	2012.12.30	초판 인쇄		
V.2	2013.05.30	CMD 추가	jsk	Ver 3.02.10
V.3	2016.04.05	ETHERNET 항목 추가	Bjlee	CE V3.0
V.4	2017.12.01	응답 패킷 추가(0xFF)	hjmin	KcsV3.03.04



N1 ROBOT CONTROLLER

---

# CONTROLLER MANUAL

FIRST EDITION DECEMBER 2012

ROBOSTAR CO, LTD

ROBOT R&D CENTER

---