

罗普伺达机器人
N1 系列扩展
Robostar Protocol



- | Option Module
- Unihost Protocol

Robostar
www.robostar.co.kr

ROBOSTAR ROBOT
N1 Series Option
ROBOSTAR 通讯协议



- | Option Module
- Unihost Protocol

Robostar

www.robostar.co.kr

Copyright © ROBOSTAR Co., Ltd 2012

本使用说明书的版权由 ROBOSTAR 公司所有
未经许可，不得以任何方式传播或利用本说明书的部分或全部内容

配置升级，不另作通知

关于质量保证

本公司产品质量严格，全部产品保修期均为一年。保修期内出现的因机器本身故障或者在正常使用情况下因机器设计和制造上的问题发生的故障均免费维修。

以下几种情况，不在免费维修范围内：

- (1) 超过保修期。
- (2) 因用户或是第三方的不正当修理、改造、移动等造成的故障。
- (3) 因使用本公司以外的零配件及润滑脂引发的故障。
- (4) 因火灾、地震、台风、水灾等灾害导致的机器故障。
- (5) 因粪尿及进水等外部环境引起的非机器配置故障。
- (6) 耗材消耗引起的故障。
- (7) 未按照产品使用说明书的要求进行定期检查造成的故障。
- (8) 机器维修以外的费用和本公司无关。

ROBOSTAR 地址及联系方式

- 总公司及第一厂
京畿道安山市常绿区沙四洞 119-38
119-38, Sasa-dong, Sangnok-gu,
Ansan-City, Gyeonggi-do, Republic of
South Korea (426-220)
- 第二工厂
京畿道水原市劝善区古素洞 945
960, Gosaek-dong, Gwonseon-gu,
Suwon-City, Gyeonggi-do, Republic of
South Korea (441-813)

售后服务及产品资讯
-业务咨询
TEL. 031-400-3600
FAX. 031-419-4249
-客户中心
TEL. 1588-4428



www.robostar.co.kr

使用说明书的构成

本产品使用说明书如下。初次使用本产品请认真阅读产品使用说明书。

■ Robostar Protocol

关于RS-232C通讯在N1控制器上的连接方法及使用方法.

目录

第 1 章 概要	1-1
1.1 系统构成	1-1
第 2 章 功能	2-1
2.1 关于RS-232通讯	2-1
2.2 D-SUB 9PIN 连接头	2-1
第 3 章 安装及动作设定	3-1
3.1 HARDWARE 安装方法	3-1
3.2 CABLE 连接方法	3-1
3.3 CONTROLLER 设定	3-2
3.4 RS-232C 通讯设定	3-2
3.5 与PC进行RS-232C通讯规格	3-3
第 4 章 通讯协议	4-1
4.1 通讯规则	4-1
4.2 详细协议命令语	4-2
4.2.1 读取机器人状态信息(AA)	4-4
4.2.2 读取现在ERROR状态(AB)	4-5
4.2.3 读取机器人当前坐标值(AC)	4-7
4.2.4 读取控制器INFO(AD)	4-9
4.2.5 原点复位(BA)	4-11
4.2.6 机器人移动到指定的POINT编号坐标(BB)	4-12
4.2.7 动作运行到指定数据(BC)	4-14
4.2.8 从当前位置指定的DATA来增量移动(BD)	4-16
4.2.9 JOG START(BE)	4-18
4.2.10 JOG MOVE CONTINUE(BF)	4-20
4.2.11 JOG MOVE STOP(BG)	4-21
4.2.12 读取机器人速度(CA)	4-22
4.2.13 写入机器人速度(CB)	4-23
4.2.14 JOB START(CC)	4-24
4.2.15 JOB STOP(CD)	4-25
4.2.16 JOB RESET(CE)	4-26
4.2.17 ON LINE 紧急停止(CF)	4-27
4.2.18 ERROR RESET(CG)	4-28

4.2.19	ORIGIN STOP(CI)	4-29
4.2.20	控制器重新启动(CJ).....	4-30
4.2.21	补偿移动轴(CX)	4-31
4.2.22	读取补偿移动轴(CZ).....	4-32
4.2.23	SERVO ON/OFF(DB)	4-33
4.2.24	要运行的JOB设定(DC)	4-34
4.2.25	JOB的 AUTO/STEP MODE 设定(EA)	4-36
4.2.26	读取现在运行中的JOB STEP号码 (ED)	4-37
4.2.27	读取现运行的JOB名称 (EF)	4-38
4.2.28	控制器往PC传送文件 (FA)	4-39
4.2.29	保存文件(FB).....	4-43
4.2.30	搜索文件(FC).....	4-45
4.2.31	邀请文件信息(FD).....	4-46
4.2.32	删除文件(FE)	4-49
4.2.33	复制文件(FF)	4-50
4.2.34	变更文件名(FG).....	4-52
4.2.35	读取已保存的报警内容(FH).....	4-54
4.2.36	读取I/O卡IN PUT接点状态(GA)	4-56
4.2.37	读取I/O卡OUT PUT接点状态信息 (GB).....	4-58
4.2.38	I/O卡OUT PUT接点输出(GC).....	4-60
4.2.39	读取I/O卡信息 (GD).....	4-61
4.2.40	读取全局变数 DATA (GR)	4-62
4.2.40.1	读取全局点位	4-62
4.2.40.2	读取全局整数	4-64
4.2.40.3	读取全局实数	4-65
4.2.41	写全局变数 数据 (GW).....	4-66
4.2.41.1	保存全局点位	4-66
4.2.41.2	储存全局定数	4-68
4.2.41.3	储存全局实数	4-69
4.2.42	读取通讯错误原因(KD)	4-70
4.2.43	读取系统数据(KE)	4-71

第 1 章 概要

1.1 系统构成

N1控制器可以通过ROBOSTAR协议和PC等系统进行通讯
通过N1控制器Main Board的Host Port可进行Robostar协议通讯.

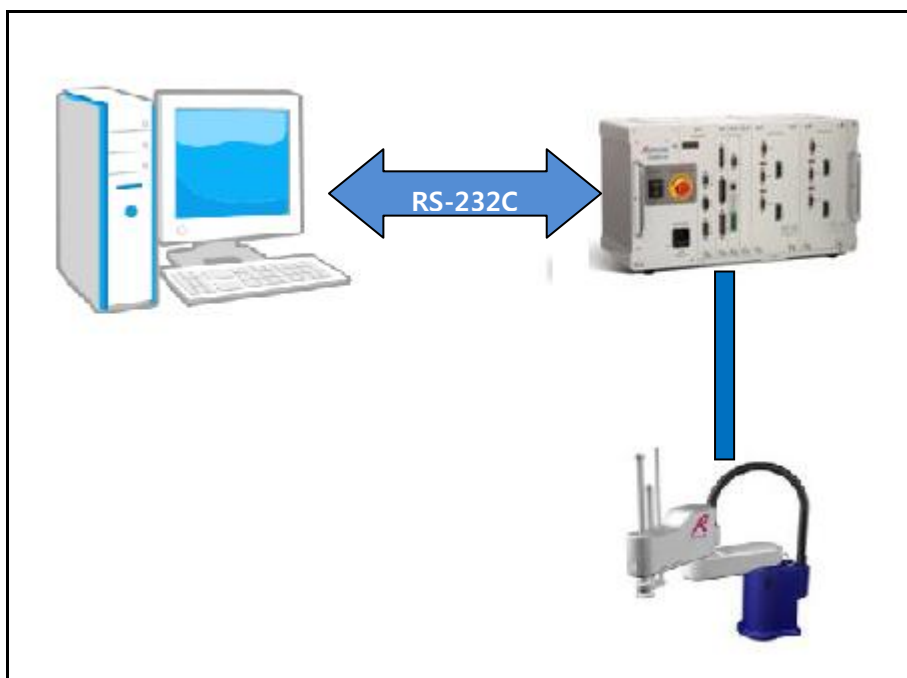


图 1.1 系统构成图

第 2 章 功能

2.1 关于RS-232通讯

RS232C 通讯距离一般是 15m,通讯速度越快距离越短.

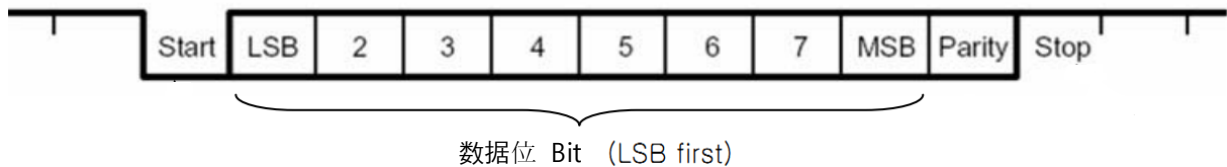
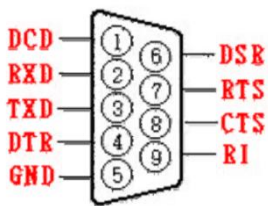


图 2.1 非同时通讯 Format

- Start BIT : 通讯 Format 开始
- DATA BIT : 可用 5/6/7/8 BIT 大小来设定, 用 LSB first 表示通讯数据值
- Parity BIT : 可以设定为 No/Even/Odd,因为干扰原因 1BIT 变化时能检测出来
- Stop BIT: 可以设定为 1/1.5/2 大小,表示数据的结束

2.2 D-SUB 9Pin 连接头



	说明
TXD	Transmit Data, 通讯数据输出信号
RXD	Receive Data, 通讯数据输入信号
RTS	用于 Ready To Send 调制解调器表示通讯准备状态, 可用于一般输出端口
CTS	用于 Clear To Send 调制解调器表示通讯准备状态, 可用于一般输出端口
DTR	Data Terminal Ready 调制解调器通讯准备信号,用于输出端口
DSR	Data Set Ready 调制解调器通讯准备信号用于输入端口
DCD	Data Carrier Detect, 输入端口
RI	Ring Indicator 输入端口
GND	接地

表 2.1 连接头说明

第 3 章 安装及动作设定

3.1 Hardware 安装方法

通过以下过程可使用控制器通讯协议.

- 1) 控制器 Main Board 通讯端口(HOST PORT)用 Serial 电缆连接.

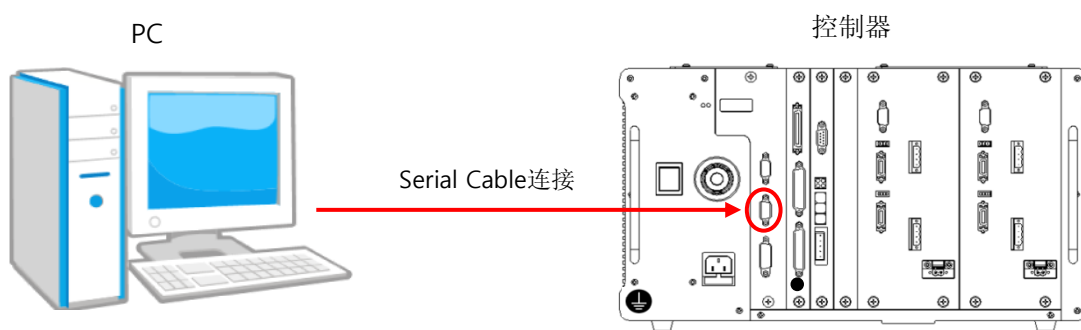


图 3.1 Serial 电缆连接方法

3.2 Cable 连接方法

电缆连接方法请参考“图 3.2 电缆连接图”.

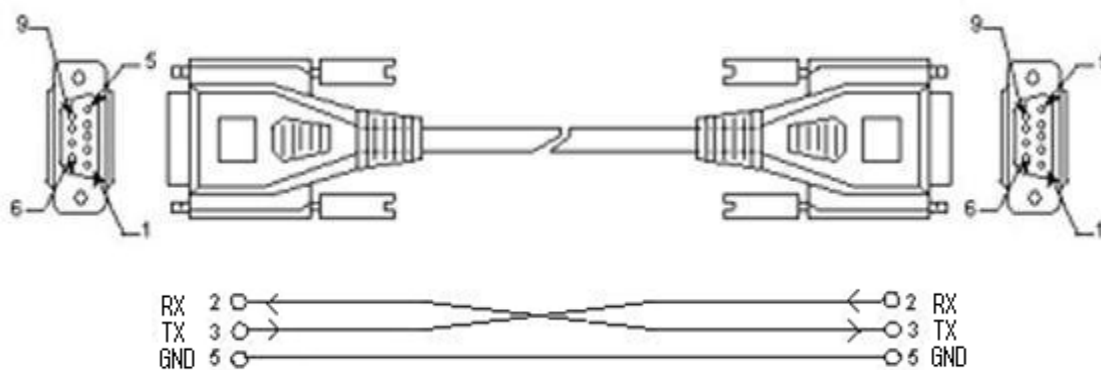


图 3.2 电缆连接图

PIN 编号	名称	内容	功能 说明
2	RxD	Received Data	数据接收信号
3	TxD	Transmitted Data	数据发送信号
5	GND	Ground	接地

表3.1 连接针编号说明

3.3 Controller 设定

控制器为了使用Robostar通讯协议应设置RS-232C通讯.
Baud Rate基本设定值是115,200bps.

3.4 RS-232C 通讯设定

1. 设定顺序

Step 1.

PARA 画面移动

```

<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 #
    
```

移动到初始画面OPEN

4: 选择PARA



```

<PARAMETER>
NO          TYPE
*CH1       XYZW
CH2        XY_TEST

SEL  INFO  PUB  EXIT
    
```

PUBLIC PARAMETER组画面OPEN



```

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

                group #
    
```

1:选择HW CONF



Step 2.

COMM 画面移动

```

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

                item #
    
```

2: 选择COMM



Step 3.

RS232 设定画面

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

group #
```

1: 选择RS232C



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

HOST RS485
```

RS232C 设定画面



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

HOST RS485
```

RS232C 设定画面

3.5 与PC进行RS-232C通讯规格

项目	设定值	初始值
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

※ 现在功能不支持ID.

第 4 章 通讯协议

4.1 通讯规则

- 最大Packet长度是包括**STX, ETX LRC** 收/发送可达到250Bytes.
- 确认包含在Packet的LRC如果LRC不同时发送NAK超过一定次数时发送RST. 如果收到NAK时需要再次发送之前Packet.
- RST意味着通讯结束收到RST会立即终止通讯回复到通讯待机点.
- ACK用于接收到了一个通讯Packet的Acknowledge来使用.
一个通讯Packet如下由 **STX, DATA, ETX, LRC**来构成.

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

STX: 0x02

ETX: 0x03

NAK: 0x15

RST: 0x12

ACK: 0x06

- LRC 计算方法 :除STX, ETX, LRC 的**exclusive-OR**.
 $LRC = DATA[0] \oplus DATA[1] \oplus DATA[2] \oplus \dots \oplus DATA[n]$ (如果LRC值是0时ETX.)
- FLAG
 - 0x30: 功能 运行 OK
表示对于要求的协议功能可运行或结束.
 - 0x31: 协议 ERROR
PC输出的Packet内容不是控制器能判断的DATA时变化.
EX) “执行指定的数据模式(BC)”时 如果输入值比坐标系输入值 '0' or '1'大时变化0x31.
 - 0x32: 功能 运行 FAIL
不能运行PC邀请的功能时变换.对功能运行FAIL的详细内容可通过“通讯报警读取(KD)”来确认.
 - 0x33: 现在控制器不支持的功能
有协议但根据控制器种类功能难实现时变换0x33.
 - 0x34: 连续的Packet的种类
Packet不能一次性结束而反复时表示Packet的种类.
从控制器读取文件时根据文件大小如果超过1个Packe时控制器发送结束文件内容后向FLAG发送0x34表示Packet结束.

4.2 详细协议命令语

GROUP 1	GROUP 2	功能 说明
A	A	读取机器人状态信息
	B	读取现在Error状态
	C	读取机器人当前位置坐标系
	D	读取控制器INFO
B	A	原点复位
	B	机器人移动到指定的Point
	C	运行到指定的数据
	D	从当前位置按指定数据增量移动
	E	JOG Start
	F	JOG MOVE Continue
	G	JOG MOVE Stop
C	A	读取机器人速度
	B	写入机器人速度
	C	JOB Start
	D	JOB Stop
	E	JOB Reset
	F	On line 紧急停止
	G	Error Reset
	I	Origin Stop
	J	控制器重新启动
	X	补偿移动轴(Desk TOP 专用命令语)
	Z	读取补偿移动轴(Desk TOP 专用命令语)
	D	B
C		运行 할 JOB 设定
E	A	JOB의 Auto/Step Mode 设定
	D	读取当前运行中的JOB Step编号
	F	读取当前运行中的JOB 名称
F	A	从控制器向PC传送文件
	B	从PC向控制器传送文件
	C	搜索文件
	D	邀请文件信息

GROUP 1	GROUP 2	功能 说明
F	E	删除文件
	F	复制文件
	G	变更文件名
	H	读取中鼎的报警内容
G	A	读取I/O卡INPUT接点状态
	B	读取I/O卡OUTPUT接点状态
	C	输出I/O卡OUTPUT接点
	D	读取I/O卡信息
	R	Global Point,Global Integer, Global Float Read
	W	Global Point,Global Integer, Global Float Write
K	D	读取通讯故障原因
	E	读取系统数据

4.2.1 读取机器人状态信息(AA)

读取机器人状态(各频道的机器人状态)信息的命令语.



- Protocol DATA

PC	STX(0x02)	Dummy(0xFF)	'A'	'A'	ETX(0x03)	LRC
----	-----------	-------------	-----	-----	-----------	-----

控制器	STX(0x02)	FLAG(0x30)	Channel 1	Channel 2	Channel 3	ETX(0x03)	LRC
-----	-----------	------------	-----------	-----------	-----------	-----------	-----

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

- Channel 1 : 变换机器人Channel 1状态信息.

如果变换值是0xB5时Channel 1的状态信息表示Servo On, Origin, Ready, Run.

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

- Channel 2 : 变换机器人Channel 2或BGT(Back Ground Task)状态信息.

变换值是0x84时表示Channel 2的信息是Ready状态.

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

- Channel 3 : 变换BGT(Back Ground Task)状态信息.

变换值是0x88时表示Channel 3的Alarm状态.

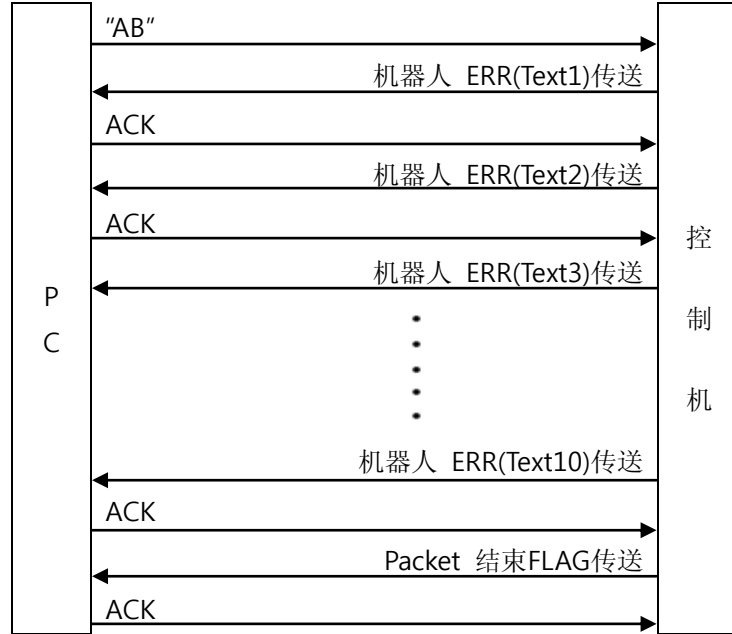
DATA 3(1Byte)							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	0	Servo On	Origin	Alarm	Ready	In Position	Run

- STX, ETX, LRC, FLAG: 请参考 "4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.2 读取现在ERROR状态(AB)

读取控制器报警信息的命令语.



ERR Text最大能顺序传送10个.

- Protocol DATA

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

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

PC	ACK
----	-----

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

PC	ACK
----	-----

控制器	STX	FLAG(0x34)	ETX	LRC
-----	-----	------------	-----	-----

PC	ACK
----	-----

- ERR Text: 包括报警编号和报警内容共27Byte组成.

ERR Text(27Byte)		
Err Code(4Bytes)	(3Bytes)	Err 内容(20Bytes)
"1153"	' ' + ':' + ' '	"T/P Emergency"

- FLAG : 连续的ERR Text内容结束.
0x30: 邀请的协议运行结束.
0x34: 连续的Packet结束.
- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

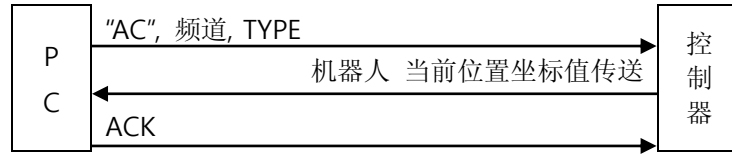
※Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 仅控制器在报警状态下变换报警内容.
(想知道当前报警内容时使用.)
- ▶ 查询以往报警信息时使用“读取以保存报警(FH)”.

4.2.3 读取机器人当前坐标值(AC)

读取指定频道机器人当前坐标值的命令语.



- Protocol DATA

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

控制器	STX	FLAG(0x30)	DATA	ETX	LRC
-----	-----	------------	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 读取当前位置坐标值机器人频道(1Byte)

- 0: 机器人频道1.
- 1: 机器人频道2.
- 2: 机器人频道3.

- TYPE(0~2): 邀请DATA Type(1Byte)

- 0: 机器人当前位置信息用Pulse DATA邀请
- 1: 机器人当前位置信息用Angle DATA邀请
- 2: 机器人当前位置信息用XY坐标DATA邀请

- DATA

STX	FLAG	DATA	ETX	LRC						
		<table border="1"> <tr> <td>1轴 (10Byte)</td> <td>2轴 (10Byte)</td> <td>3轴 (10Byte)</td> <td>4轴 (10Byte)</td> <td>.....</td> <td>ARM (1Byte)</td> </tr> </table>	1轴 (10Byte)	2轴 (10Byte)	3轴 (10Byte)	4轴 (10Byte)	ARM (1Byte)		
1轴 (10Byte)	2轴 (10Byte)	3轴 (10Byte)	4轴 (10Byte)	ARM (1Byte)					

TYPE 0: Pulse DATA整数型.(10Byte*机器人频道轴数+ARM)

1轴(10Bytes)				最大轴(10Bytes)				ARM(1Byte)		
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"123456789"				' '	"123456789"				' '	'2'

TYPE 1: Angle DATA实数型.(表示小数点3位.)

1轴(10Bytes)									
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					':'	"123"			''

最大轴(10Bytes)										ARM(1Byte)
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					':'	"123"			''	'2'

TYPE 2: XY坐标值实数型.(表示小数点3位.)

1轴(10Bytes)									
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					':'	"123"			''

最大轴(10Bytes)										ARM(1Byte)
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
"12345"					':'	"123"			''	'0'or'1'

- 最大轴: 机器人频道里使用的最大轴. 最大轴可用“控制器INFO读取(AD)”协议来确认.

- ARM(0~2): SCARA时表示机器人姿态.
 - 0: LEFT_form
 - 1: RIGHT_form
 - 2: NO_form
- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

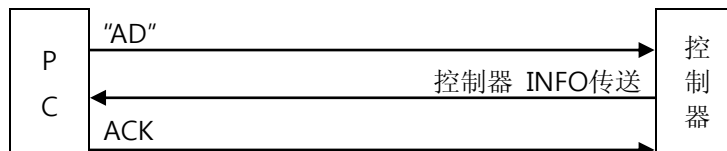
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 **CAUTION**

- ▶ 仅邀请XY坐标Data值时为ARM_Form值.
- ▶ Angle Data时Arm_Form信息值为No_Form值.

4.2.4 读取控制器INFO(AD)

读取控制器INFO命令语.



- Protocol DATA

PC

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

控制器

STX	Dummy(0xFF)	FLAG	MAX CH	NAME	VER	MODEL	频道 信息	ETX	LRC
-----	-------------	------	--------	------	-----	-------	-------	-----	-----

PC

ACK

- MAX CH: 控制器参数里已设定的最大运行频道数.(频道最多可运行3个,设定方法请参考“操作说明书1.3.1.8 MAX CH”.
- NAME: 控制器里设定的名称.如果没有设定则会以' ' 15Bytes来表示. (设定方法请参考“操作说明书 1.3.4.4 NAME”.)

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

- VER: 软件版本.

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

- MODEL: 各频道的机器人规格.

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

- 频道信息: 设定的频道信息.

MAX AXIS(3Bytes)			频道 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): 各轴频道可使用的最大使用轴信息.

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

各可设置1~6.

- 频道 TYPE(3Byte): 当前设定的频道信息.

频道1 TYPE	频道2 TYPE	频道3 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): 各频道当前使用的轴信息.

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

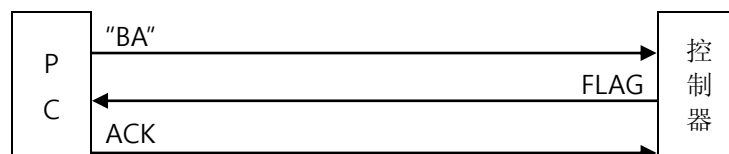
- 0: AXIS不使用.
- 1: AXIS使用.

- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.5 原点复位(BA)

机器人原点复位(Origin)命令语.



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): 需要原点复位的机器人频道(1Byte)
 - 0: 频道 1
 - 1: 频道 2
 - 2: 频道 3
- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

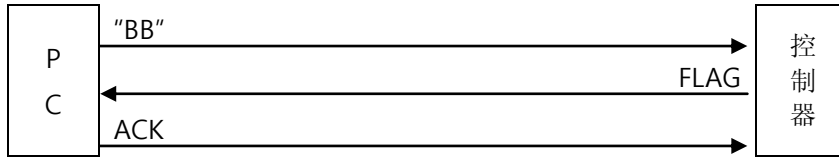
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时
-“原点复位(BA)”时自动Servo ON
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时
-“原点复位(BA)”前用“Servo ON/OFF(DB)”请Servo ON后回复原点.
- ▶ AUTO SERVO ON设定请参考“操作说明书SVON”.

4.2.6 机器人移动到指定的Point编号坐标(BB)

机器人移动到指定Point编号的命令语.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'B'	频道	文件名称	动作 TYPE	Point Num 1	Point Num 2	ETX	LRC
----	-----	-------------	-----	-----	----	------	------------	----------------	----------------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 要移动的机器人的频道(1Byte)

- 0: 机器人频道1
- 1: 机器人频道2
- 2: 机器人频道3

- 文件名称: Point文件名和扩展(12Bytes)

- 文件名+ '.' 包括扩展12字节.

- ※ 文件名超过5个字时发生“0x32功能运行FAIL”的FLAG.
- ※ 文件名是大写字母,扩展名仅是大写字母(PNT)或小写字母(pnt).
- ※ 大,小文字混用或扩展名不同时发生“0x31协议ERROR”的FLAG.

文件名称(12Bytes)											
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
文件名 + '.' + 扩展											

- 动作TYPE(0~3):动作运行TYPE(1Byte)
 - 0: Joint(JMOV : PTP移动)一个目标点的Local Point编号.
 - 1: Linear(LMOV: 直线插补移动) 一个目标点的Local Point编号.
 - 2: Arc(AMOV: 圆弧补偿移动)经由点Local Point一个和目标Local Point一个.
 - 3: Circle(CMOV:圆补偿移动)经由点Local Point两个.
- ※ 机器人计算从当前位置坐标值到目标坐标(Local Point)移动路径.

- Point Num 1, Point Num2: 保存经由点及目标点
 - 动作Type 0, 1情况.
 - Local Point 5为目标移动时.

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

※ Point Number 2是移动路径.

动作Type 2, 3时.

- 经过Local point 5和Local Point 15的圆和圆弧补偿移动.

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

- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”..

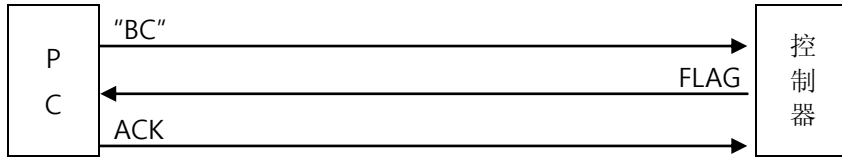
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时执行“BB”命令时自动变更为Servo ON状态..
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时请在执行“BB”命令前通过“DB”命令变更为Sevo ON状态.
- ▶ AUTO SERVO ON 设定请参考“操作说明书SVON”.

4.2.7 动作运行到指定数据(BC)

输入机器人经由坐标或目标坐标来执行动作的命令语.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'C'	频道	动作TYPE	坐标系	位置DATA	ETX	LRC
----	-----	-------------	-----	-----	----	--------	-----	--------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 动作执行中机器人的频道(1Byte)
 - 0: 机器人 频道1 .
 - 1: 机器人 频道2 .
 - 2: 机器人 频道3 .
- 动作TYPE(0~3): 动作设定(1Byte)
 - 0: JMOV : 从当前位置向目标点轴补偿移动.
 - 1: LMOV: 从当前位置向目标点支线插补移动.
 - 2: AMOV: 从当前位置向经由点一个和目标点一个连接来移动圆弧.
 - 3: CMOV: 从当前位置向经由点两个来画圆移动.
 - ※ 详细的动作请参考“程序说明书3.9~3.12”.
 - ※ 输入以外的值时发生“0x33当前控制器未具备的功能”的FLAG.
- 坐标系(0,1) : 坐标系设定(1Byte)
 - 0: Angle 坐标: 度(Degree)单位的坐标.(直角机器人时单位是mm.)
 - 1: XY 坐标: mm单位的坐标.
 - ※ 输入以外的值时发生“0x31协议ERROR”FLAG.



▶ SCARA机器人时XYZ坐标系用当前位置来计算移动的路径.

- 位置DATA：要移动的目标坐标DATA
 - 坐标系设定值为Joint Angle TYPE时（机器人类别为直角机器人时以mm单位来动作）

各轴各有10Bytes位置DATA(10Bytes×MAX AXIS)

1轴位置DATA(10bytes)				2轴位置DATA(10bytes)				...	6轴 位置DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

※ 位置数据比MAX AXIS少时有可能不能动作。

※ 不知道现机器人MAX AXIS时使用“控制器INFO读取(AD)来确认MAX AXIS，之后输入位置DATA值

EX) 4轴机器人时: 4轴 × 10Bytes = 40Bytes)

1轴位置DATA(10bytes)				2轴位置DATA(10bytes)				...	4轴位置DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- 坐标系设定值为XYZ TYPE时(机器人类型为SCARA时支持XY坐标系)

各轴各有10Bytes位置DATA(10Bytes×MAX AXIS)

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

- ※ AMOV：经由点1个，位置DATA和目标点1个的位置DATA
- ※ CMOV：经由点2个的位置DATA

- STX, ETX, LRC, FLAG: 请参考4.1通讯规则

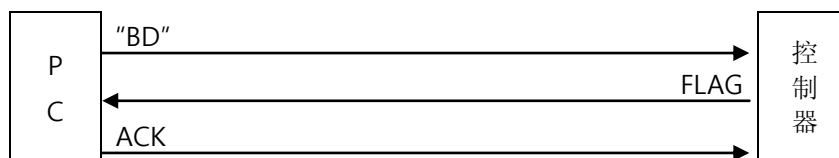
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因。

CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时执行“BC”命令时自动变更为Servo ON状态..
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时请在执行“BB”命令前通过“DB”命令变更为Sevo ON状态.
- ▶ AUTO SERVO ON 设定请参考“操作说明书SVON

4.2.8 从当前位置指定的DATA来增量移动(BD)

从当前位置输入的DATA来对机器人进行增量移动的命令语.



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'D'	频道	动作TYPE	坐标系	位置DATA	ETX	LRC
----	-----	-------------	-----	-----	----	--------	-----	--------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 将增量移动机器人的频道(1Byte)

0: 机器人频道1.

1: 机器人频道2.

2: 机器人频道3.

- 动作TYPE(0, 1): 动作设定(1Byte)

0: JMOV: 现位置上以位置DATA大小来增量移动

1: LMOV: 现位置上以位置DATA大小来增量移动

※ 但输入以外的数时, 0x33现控制器上不能表现的功能的FLAG将发生

- 坐标系(0, 1): 坐标系设定(1Byte)

0: Angle坐标: 度(Degree)单位的坐标.(直角机器人时以mm单位来动作)

1: XY坐标: mm单位的坐标.

※ 输入这之外的值时"0x31协议ERROR"的FLAG将发生

CAUTION

▶ SCARA机器人时XYZ坐标系Data根据机器人当前位置来计算机器人的移动路径

- 位置DATA：增量移动坐标DATA
 - 坐标系设定值是Angle坐标时
各轴各有10Bytes位置DATA.(10Bytes×MAX AXIS)

EX) 4轴机器人时:4轴 × 10Bytes = 40Bytes

1轴位置DATA(10bytes)				2轴位置DATA(10bytes)				...	4轴位置DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- 坐标系 设定值 XYZ TYPE时
各轴各有10Bytes位置DATA.(10Bytes×MAX AXIS)

1轴位置DATA(10bytes)				2轴位置DATA(10bytes)				...	6轴 位置DATA(10bytes)			
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...	1Byte	...	1Byte	

- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

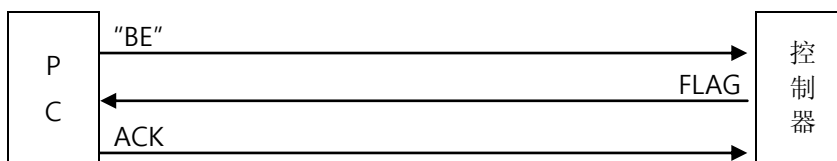
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时
执行“BD”命令时自动变更为Servo ON状态..
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时
请在执行“BD”命令前通过“DB”命令变更为Sevo ON状态.
- ▶ AUTO SERVO ON 设定请参考“操作说明书SVON

4.2.9 JOG START(BE)

对于指定的轴执行JOG动作的命令语



- Protocol DATA

PC	STX	Dummy(0xFF)	'B'	'E'	频道	AXIS	方向	动作TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	----	--------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 将执行动作机器人的频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- AXIS(0~5): 输入准备运行的机器人的轴号码
 - 0: 1(X)轴 驱动
 - 1: 2(Y)轴 驱动
 - 2: 3(Z)轴 驱动
 - 3: 4(W)轴 驱动
 - 4: 5(EX1)轴 驱动
 - 5: 6(EX2)轴 驱动
 - ※ 输入其他值时“0x31 协议 ERROR”의 FLAG将发生
- JOG 动作方向(0, 1) : 选择机器人轴的移动方向.
 - 0: - 方向
 - 1: + 方向
 - ※ 输入其他值时“0x31 协议 ERROR”의 FLAG将发生

- 动作TYPE(0, 1):动作设定(1Byte)
 - 0(JMOV): 从当前位置向已设定的方向来移动机器人轴
 - 1(LMOV): 从当前位置向已设定的方向来进行直线보간 이동
- ※ 输入其他值时发生“0x31 协议 ERROR”的FLAG
- STX, ETX, LRC, FLAG: 请参考4.1通讯规格

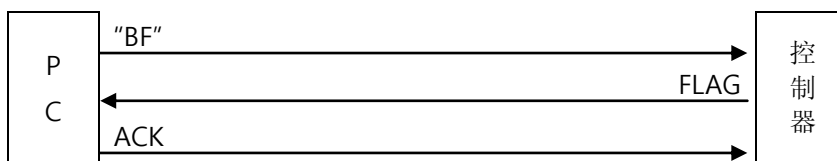
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 为了持续维持JOG动作，500ms以内必须传送BF(Jog Move Continue)协议
- ▶ SCARA机器人时以LMOV移动时以XY坐标系动作
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时执行“BE”命令时自动变更为Servo ON状态..
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时请在执行“BE”命令前通过“DB”命令变更为Sevo ON状态.
- ▶ AUTO SERVO ON 设定请参考“操作说明书SVON

4.2.10 JOG MOVE Continue(BF)

为了JOG动作持续维持，对控制器要求的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): 维持动作执行的机器人的频道 (1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

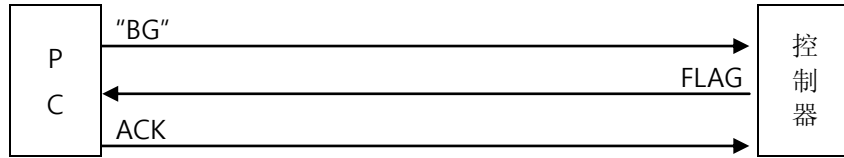
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

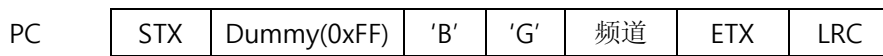
- ▶ 欲维持JOG动作每500MS，必须传送“4.1.12 JOG MOVE Continue(BF)”协议
- ▶ 超过500ms时机器人终止.

4.2.11 JOG MOVE STOP(BG)

JOG动作停止的命令语



- Protocol DATA

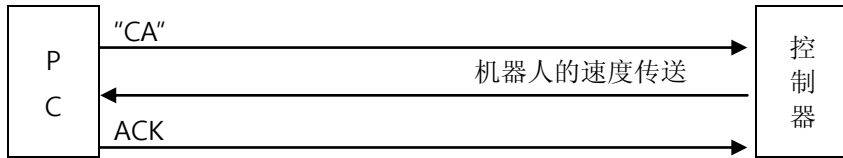


- 频道(0~2): 要终止的机器人频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.12 读取机器人速度(CA)

读取机器人频道的机器人当前速度的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	机器人移动速度	ETX	LRC
-----	------	---------	-----	-----

PC

ACK

- 频道(0~2): 读取当前机器人速度的频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- 机器人移动速度: 现在机器人速度 DATA(4Bytes)
- 机器人移动速度 = $Mv(\text{各轴的最大速度}) * 0.001 * \text{机器人速度}$

机器人移动速度(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'3'	'0'	'0'

- 关于JOINT MOTION的详细内容或者设定方法, 请参考“操作运用说明书1.2.2.2 JOINT

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

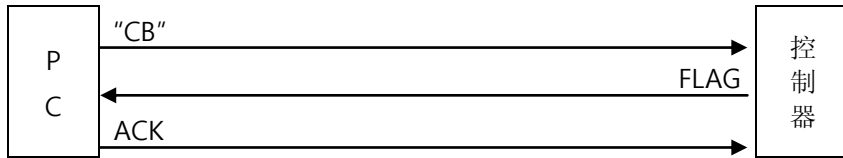
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

! CAUTION

- ▶ 机器人移动速度最大值为1000 机器人
- ▶ 机器人移动速度的输出值为1000时, 以Mv设定值的100%的速度移动

4.2.13 写入机器人速度(CB)

写入机器人频道的机器人的驱动速度值的命令语



- Protocol DATA

PC	STX	Dummy(0xFF)	'C'	'B'	频道	机器人速度	ETX	LRC
----	-----	-------------	-----	-----	----	-------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 变更指定速度值的频道 (1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .

- 机器人速度: 机器人速度值(4Bytes)

- 写入范围 : 0~1000

机器人移动速度 = $Mv(\text{各轴的最大速度}) \times 0.001 \times \text{机器人速度}$

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

- 关于JOINT MOTION的详细内容或者设定方法, 请参考“操作运用说明书1.2.2.2 JOINT

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

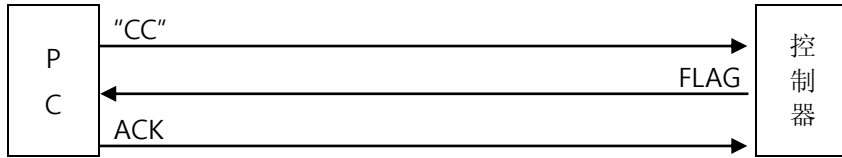
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

! CAUTION

- ▶ 能够设定的最大速度为1000
写入速度值1000时, 以100%的速度移动 (速度值1000=>100%)
- ▶ 超过机械部容许的最大RPM使用时, 将会有噪音以及破损危险. 希望在确认好贴在机械部上的商标后使用机器人

4.2.14 JOB START(CC)

运行JOB的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): 需要运行的JOB频道(1Byte)
0: 机器人频道1 .
1: 机器人频道2 .
2: 机器人频道3 .

※ 在原点运行没完毕之前运行JOB时Run Fail报警将会发生

※ 协议通讯中FLAG里0x32返还时对于这的原因推荐使用“通讯错误原因读取”协议来读取

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

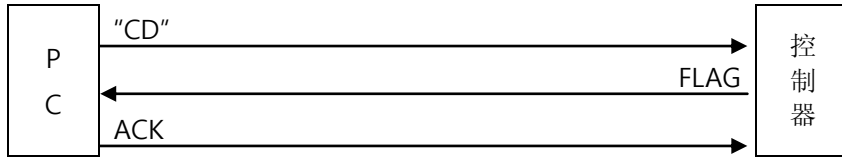
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ “作业运行(CC)”前必须用“要运行的JOB设定(DC)”来设定JOB状态下运行
- ▶ N1控制器Parameter上AUTO SERVO ON设定为ENB时，
- ▶ N1控制器Parameter上的AUTO SERVO ON设定为DIS时
- ▶ “CC命令执行时JOB虽然在读取状态但不能实行
- ▶ “CC命令执行前通过DB命令变更为Sevo ON状态
- ▶ AUTO SERVO ON 设定请参考“操作说明书 SVON”.

4.2.15 JOB STOP(CD)

停止机器人JOB运行的命令语



- Protocol DATA

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

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2):停止 JOB运行的频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ 向FLAG反应正确的数据包则是0x30,如果是不正确的数据包则是0x32.关联的原因可利用“读取通讯报警(KD)”协议来了解

CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时执行“CD”命令时自动变更为Servo ON状态..
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时请在执行“BD”命令前通过“DB”命令变更为Sevo ON状态.
- ▶ AUTO SERVO ON 设定请参考“操作说明书 SVON”

4.2.16 JOB Reset(CE)

指定的机器人频道JOB运行STEP设定为开始(MAIN)的命令语。



- Protocol DATA

PC	STX	Dummy(0xFF)	'C'	'E'	频道	ETX	LRC
----	-----	-------------	-----	-----	----	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): JOB STEP设定为开始(MAIN)的机器人频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

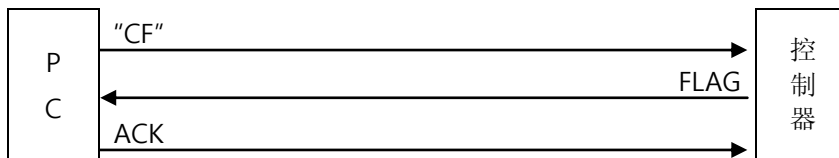
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 相关机器人频道Servo ON时FLAG返还0x32
- ▶ Online JOB Reset(CE)只有在Servo OFF的状态时才会动作

4.2.17 On line 紧急停止(CF)

机器人On line紧急停止命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 手柄上发信“Host Emergency”后报警将发生
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

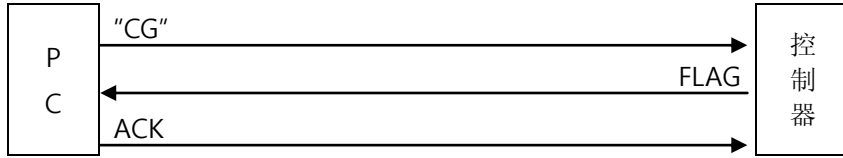
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 希望只有在机器人运转中紧急停止时使用
- ▶ 通过CG命令可以解除非常停止

4.2.18 Error Reset(CG)

初始化控制器上发生的报警的命令



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

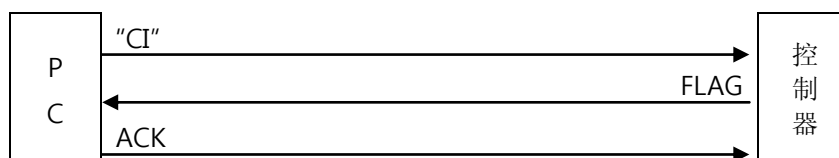
ACK

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.19 Origin Stop(CI)

相关频道的原点复位停止命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): 原点复位停止机器人频道(1Byte)
0: 机器人频道1 .
1: 机器人频道2 .
2: 机器人频道3 .
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

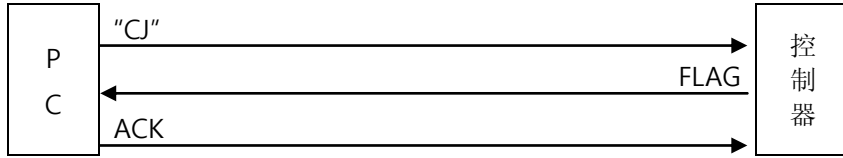
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ N1控制器Parameter上AUTO SERVO ON设定为“ENB”时
-“Origin Stop(CI)”时自动Servo OFF
- ▶ N1控制器Parameter上AUTO SERVO ON设定为“DIS”时
-“Origin Stop(CI)”时停止原点复位后维持 Servo ON
- ▶ AUTO SERVO ON 设定请参考“操作说明书SVON

4.2.20 控制器重新启动(CJ)

重新启动控制器的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 控制器上发生致命性的报警后使用“Error Reset(CG)”但依然不能正常化时使用此命令语
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

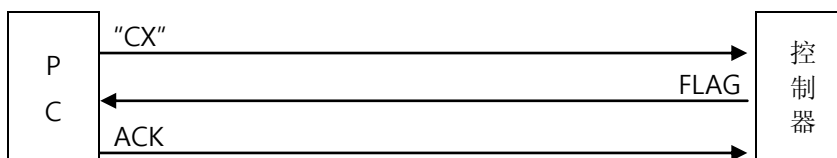
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 控制器重新启动命令语只能在Servo OFF的状态下使用，在Servo ON状态下使用重启时会返回“功能实行FAIL(0x32)”。

4.2.21 补偿移动轴(CX)

XY补偿移动时变更成为基准轴的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): 轴变更机器人频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- AXIS(1或者4): 成为基准的轴的号码
 - 1: X轴为基准JMOV, LMO, CMOV, AMOV补偿移动.
 - 4: W轴为基准JMOV, LMO, CMOV, AMOV补偿移动.
- STX, ETX, LRC, FLAG: 参考4.1通讯规格.

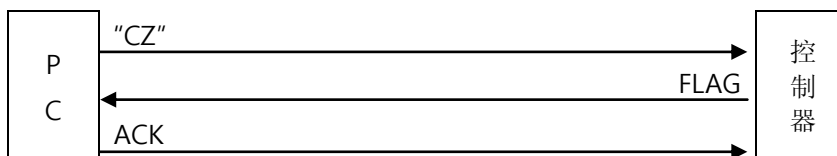
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 CAUTION

- ▶ DESK TOP是机器人专用命令语
- ▶ 只有在伺服关掉的状态下轴才可以变更
- ▶ 转换到MAIN Menu画面时轴变更值被初始化

4.2.22 读取补偿移动轴(CZ)

读取当XY补偿移动时成为现在基准轴的信息的命令语



- Protocol DATA

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

控制器	STX	FLAG	DATA	ETX	LRC
-----	-----	------	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 机器人基准轴信息读取频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- DATA(1或者4): 返还成为补偿移动的轴的号码
 - 1: X轴 以基准 JMOV, LMO, CMOV, AMOV 补偿移动.
 - 4: W轴 以基准 JMOV, LMO, CMOV, AMOV 补偿移动
- STX, ETX, LRC, FLAG: "参考4.1通讯规格.

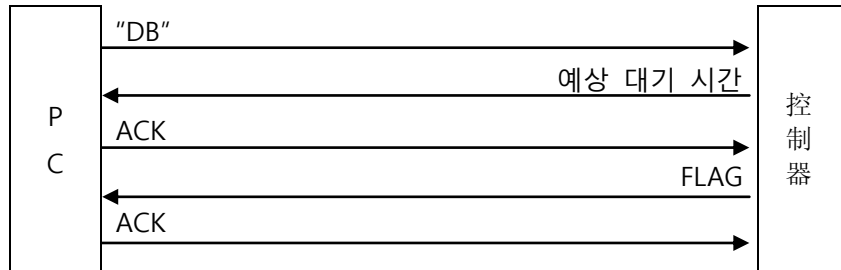
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

▶ DESK TOP是机器人专用命令语

4.2.23 SERVO ON/OFF(DB)

让相关机器人频道的伺服开或者关的命令语



- Protocol DATA

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

控制器	STX	FLAG	预计等待时间	ETX	LRC
-----	-----	------	--------	-----	-----

PC	ACK
----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 需要SERVO ON/OFF的频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- DATA(0, 1) : SERVO ON/OFF设定(1Byte)
 - 0 : SERVO OFF
 - 1 : SERVO ON
 - ※ 输入其他值时发生 "0x31 协议ERROR"的FLAG.
- 预想等待时间:运行时要花的预测时间(2Bytes)
 - 预想时间2秒

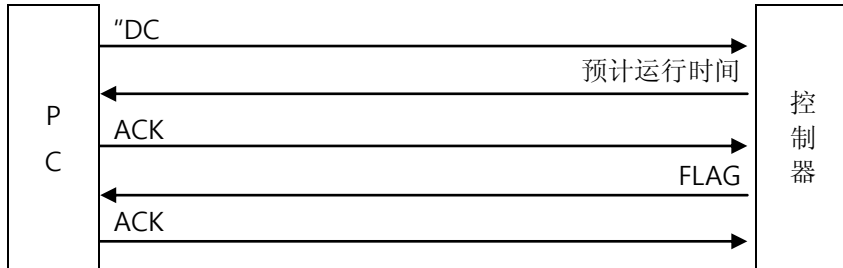
DATA2 (2Bytes)	
'0'	'2'

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.24 要运行的JOB设定(DC)

设定要运行的JOB文件的命令语



- Protocol DATA

PC	STX	Dummy(0xFF)	'D'	'C'	频道	文件名称	ETX	LRC
----	-----	-------------	-----	-----	----	------	-----	-----

控制器	STX	FLAG	预计等待时间	ETX	LRC
-----	-----	------	--------	-----	-----

PC	ACK
----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 频道(0~2):需要运行的 JOB文件设定频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- 文件名称: 要运行的JOB文件的文件名和扩展名(12Bytes)
 - 文件名, '.' 并且包括扩展在内在有12个字.
 - 扩展名用大写字母 (JOB) 或者小写字母 (job) 都行

文件名称(12Bytes)												
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	
文件名(5字以内)								''	扩展("JOB"or"job")			

※ 文件名和扩展除外其他部分上请写入 SPACE(0x20)

- 预计等待时间: 运行时要花的预测时间(2Bytes)
 - 预计时间20秒

DATA2 (2Bytes)	
'2'	'0'

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

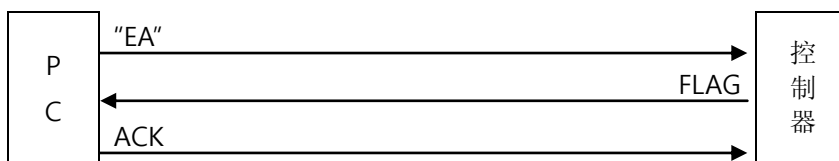
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 不能原点执行的状态下实行Job设定时Run fail报警将发生

4.2.25 JOB의 Auto/Step Mode 设定(EA)

设定JOB动作MODE的命令语



- Protocol DATA

PC

STX	Dummy(0xFF)	'E'	'A'	频道	MODE	ETX	LRC
-----	-------------	-----	-----	----	------	-----	-----

控制器

STX	FLAG	ETX	LRC
-----	------	-----	-----

PC

ACK

- 频道(0~2): JOB运行MODE变更频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- MODE(0, 1): Auto/Step Mode TYPE设定(1Byte)
 - 0: AUTO模式 – 程序全体步骤连续运行.
 - 1: STEP模式 – 程序运行第一步后暂停 (HOLD)

※ 输入其他值时发生“0x31 协议ERROR”的FLAG. 写入其他的值时发生“0x31 协议ERROR”的FLAG
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

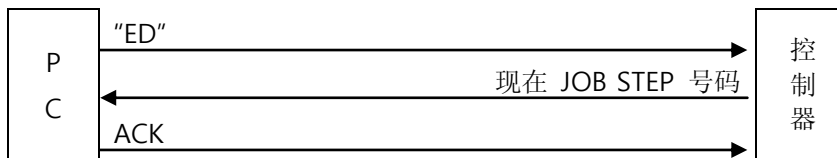
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

CAUTION

- ▶ 机器人驱动中模式转换时FLAG将会返还0x32。
- ▶ 模式转换要在相关频道的RUN Bit时OFF的情况下才能转换
(使用“机器人状态信息读取(AA)”能确定RUN (状态) Bit)

4.2.26 读取现在运行中的JOB STEP号码 (ED)

读取现在运行中的JOB STEP号码的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	STEP 号码	ETX	LRC
-----	------	---------	-----	-----

PC

ACK

- 频道(0~2): 读取运行中的STEP号码的频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .
- STEP号码: 运行中的STEP号码(4Bytes)

现在JOB的27步运行中的情况

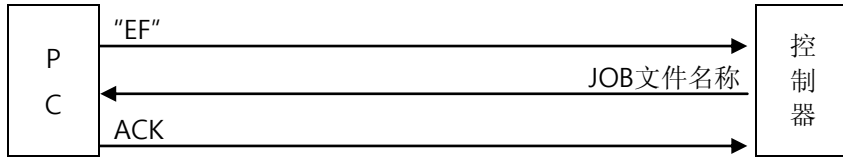
STEP 号码(4Bytes)			
1Byte	1Byte	1Byte	1Byte
'0'	'0'	'2'	'7'

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.27 读取现运行的JOB名称 (EF)

指定的机器人频道上读取现运行中的 JOB 名称的命令语



- Protocol DATA

PC

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

控制器

STX	FLAG	文件名称	ETX	LRC
-----	------	------	-----	-----

PC

ACK

- 频道(0~2): 读取运行中的JOB文件名称的频道(1Byte)

0: 机器人频道1 .

1: 机器人频道2 .

2: 机器人频道3 .

- 文件名称: 现在执行中的 JOB 文件名称.(12Bytes)

- 文件名 用+ ' ' + 'J' + 'O' + 'B'的形态传送.

- 写入文件名称以外的其他后缀部分输入' '.

EX) 叫做 RS 的 JOB 文件被返还时

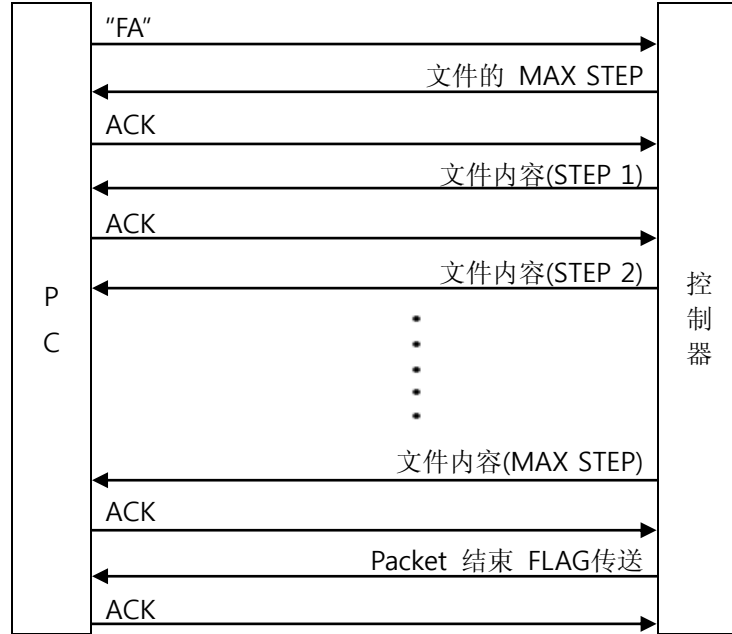
DATA(12Bytes)								
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	...	1Byte
'R'	'S'	'.'	'J'	'O'	'B'	' '	...	' '

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.28 控制器往PC传送文件 (FA)

从控制器往PC传送JOB文件或者点位文件的命令语



- Protocol DATA(邀请JOB文件时)

PC	STX	Dummy(0xFF)	'F'	'A'	频道	TYPE	文件名称	PNT_TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	----------	-----	-----

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

PC	ACK
----	-----

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

PC	ACK
----	-----

⋮

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

PC	ACK
----	-----

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

PC	ACK
----	-----

- 频道(0~2): 有接受传送文件的频道(1Byte) 有接受传送文件的
 0: 机器人频道1 .
 1: 机器人频道2 .
 2: 机器人频道3 .
- TYPE: 控制器的储存位置(1Byte)
 0: Backup RAM
 ※ 写入其他值时会 0x33 对现在控制器不构成的功能的 FLAG
- 文件名称: JOB 文件名称(12Bytes)
 - 文件名最小一字开始最大五字截止可以
 ※ 文件名不能大小写字母混用.
 ※ 文件名超过五字时将会获得“0x32 功能运行 FAIL”的 FLAG.
 - 扩展名用大写字母(JOB, PNT)或者小写字母(job, pnt)可以混用.

EX) 叫做 RS 的 JOB 文件传送成功时

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

※ 文件名称和扩展除外的部分上请写入 SPACE(0x20).

- PNT_TYPE: 点位数据的情况时决定输出类型形态
 0: Angle类型的位置数据
 1: XY类型的位置数据

PNT_DATA
1Byte
'0'or'1'

※ 仅邀请 JOB DATA 时不反应 PNT_TYPE DATA.

- 文件内容: 根据在 JOB 文件上储存的内容最大可以返还 100Byte

EX) IF I100==0 THEN

1 Byte . . . 16 Byte	17Byte
"IF I100==0 THEN"	'\n'

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

- Protocol DATA(要求PNT文件时)

PC	STX	Dummy(0xFF)	'F'	'A'	频道	TYPE	文件名称	PNT_TYPE	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	----------	-----	-----

控制器	STX	Dummy(0xFF)	FLAG(0x30)	MAX Point 号码	ETX	LRC
-----	-----	-------------	------------	--------------	-----	-----

PC	ACK
----	-----

控制器	STX	Dummy(0xFF)	FLAG(0x30)	Local Point 号码1	Point Data	ETX	LRC
-----	-----	-------------	------------	-----------------	------------	-----	-----

PC	ACK
----	-----

⋮

控制器	STX	Dummy(0xFF)	FLAG(0x30)	Local Point 号码n	Point Data	ETX	LRC
-----	-----	-------------	------------	-----------------	------------	-----	-----

PC	ACK
----	-----

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

PC	ACK
----	-----

- 频道(0~2):持有接受传送文件的频道(1Byte)

0: 机器 频道1 .

1: 机器人频道2 .

2: 机器人频道3 .

- TYPE: 控制器储存位置(1Byte)

0: Backup RAM

※ 写入其他的值时“将会收到 0x33 现在控制器上不构成的功能的 FLAG

- 文件名称: PNT 文件名称(12Bytes)

- 文件名最小一字开始最大五字截止可以

※ 文件名不能大小写字母混用.

※ 文件名超过五字时将会获得“0x32 功能运行 FAIL”的 FLAG.

EX) 叫做 RS 的 PNT 文件传送成功时

DATA(12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
' '	...	' '	'R'	'S'	'.'	'P'	'N'	'T'

※ 文件名和扩展除外的部分上请写入 SPACE(0x20)

- PNT_TYPE: 点位数据时决定输出类型形态
0: Angle类型的坐标数据
1: XY类型的坐标数据

PNT_DATA
1Byte
'0'or'1'

- MAX Point: 返还储存在 PNT 文件上的最大 Point 号码
- Local Point 号码(5Byte): Point Data 返还储存已有的号码

EX) 返还 Point 号码 8 号时的情况

Local Point(5Bytes)				
1Byte	1Byte	1Byte	1Byte	1Byte
'P'	'0'	'0'	'0'	'8'

- Point Data: 返还储存已有的位置坐标

EX) 6轴 × 11Bytes = 66Bytes

1轴位置DATA(11bytes)				2轴位置DATA(11bytes)				...
1Byte	1Byte	...	1Byte	1Byte	1Byte	...	1Byte	...

6轴 位置DATA(11bytes)			ARM	USED	
1Byte	...	1Byte	1Byte	1Byte	0x0A

- USED: Point Data 表现出文件使用有无

0: Point Data 文件没有使用

1: Point Data 文件使用可以

- ARM(0~2):

0: LEFT_form

1: RIGHT_form

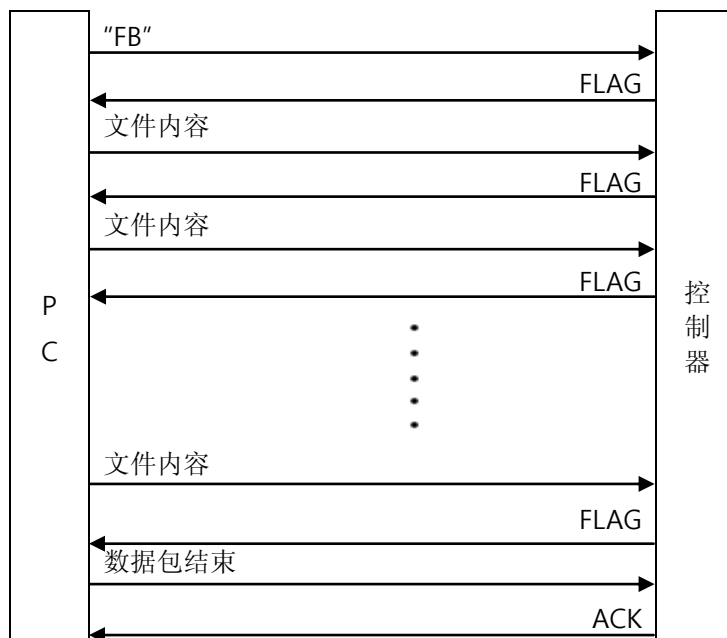
2: NO_form

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.29 保存文件(FB)

从PC到控制传送文件的命令语



● Protocol DATA

PC	STX	Dummy(0xFF)	'F'	'B'	频道	TYPE	JOB번호	文件名称	ETX	LRC
----	-----	-------------	-----	-----	----	------	-------	------	-----	-----

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

PC	STX	FLAG(0x30)	文件内容	ETX	LRC
----	-----	------------	------	-----	-----

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

⋮

PC	STX	FLAG(0x30)	文件内容	ETX	LRC
----	-----	------------	------	-----	-----

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

PC	STX	FLAG(0x34)	ETX	LRC
----	-----	------------	-----	-----

控制器	ACK
-----	-----

- 频道(0~2): 要储存文件的机器人频道(1Byte)
 - 0: 机器人 频道1 .
 - 1: 机器人 频道2 .
 - 2: 机器人 频道3 .
- TYPE(1): 控制器的储存位置(1Byte)
 - 0: Backup RAM
 - ※ 写入其他值时将会收到 0x33 现在控制器上没有构成的功能的 FLAG
- JOB 号码: JOB 文件 Index 号码是 最大 200 截止可以.

JOB号码(3Bytes)		
1Byte	1Byte	1Byte
'0'	'0'	;1;

- 文件名称: JOB 文件名称(12Bytes)
 - 文件名可以再最小一字开始带最大五字
 - ※ 文件名不能大小写字母混用.
 - ※ 文件名超过五字时将会获得“0x32 功能运行 FAIL”的 FLAG.
 - 扩展名可以大小字母混用.
- ※ 文件名和扩展除外的部分上请写入 SPACE(0x20)

EX) 叫做 RS 的 JOB 文件传送成功时

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

- 文件内容: 可以写入最大 100Byte

EX) IF I100==0 THEN

FLAG	1 Byte . . . 15 Byte	16Byte
0x30	"IF I100==0 THEN"	'Wn'

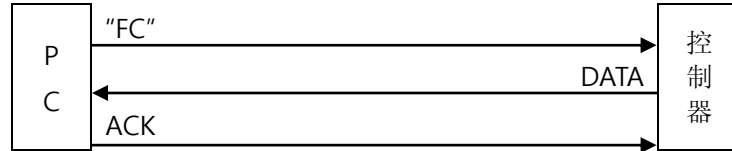
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 CAUTION

- ▶ JOB文件名相同时以前储存的JOB号码必须要一致，假如不一致的话会发生报警
- ▶ 传送文件内容时，JOB MAX STEP除外传送。假如JOB MAX STEP传送时控制器上实行了JOB的情况时“Syntax ERROR”将发生

4.2.30 搜索文件(FC)

在控制器内搜索文件的命令语



- Protocol DATA

PC	STX	Dummy(0xFF)	'F'	'C'	频道	TYPE	文件名	ETX	LRC
----	-----	-------------	-----	-----	----	------	-----	-----	-----

控制器	STX	FLAG(0x30)	DATA	ETX	LRC
-----	-----	------------	------	-----	-----

PC	ACK
----	-----

- 频道(0~2): 有接受传送文件的频道(1Byte)
 - 0: 机器人 频道1 .
 - 1: 机器人 频道2 .
 - 2: 机器人 频道3 .
- TYPE: 控制器的储存位置(1Byte)
 - 0: Backup RAM
 - ※ 写入其他值时将会收到 0x33 现在控制器上没有构成的功能的 FLAG
- 文件名: 要搜索 JOB 文件名称(12Bytes)
- - 文件名可以再最小一字开始带最大五字
EX) 搜索叫 RS 的 JOB 文件时

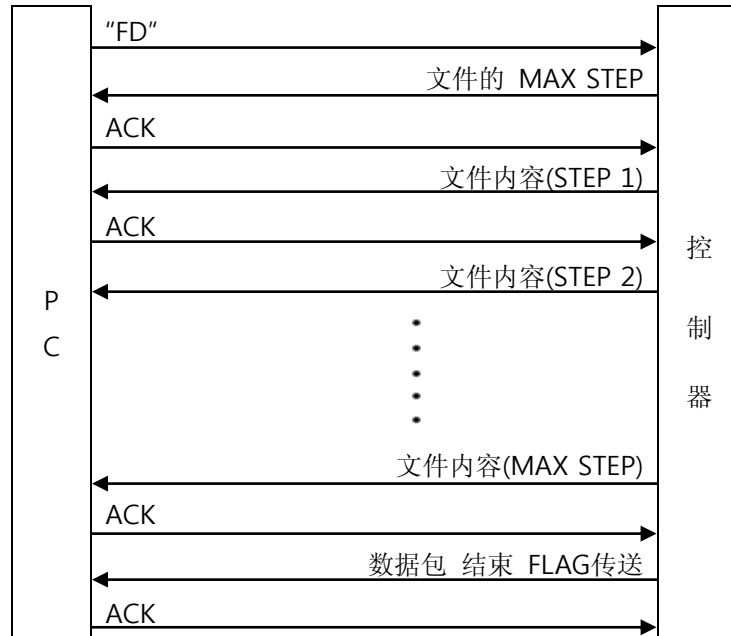
DATA(12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
' '	...	' '	'R'	'S'	'.'	'J'	'O'	'B'

- ※ 文件名超过 5 字时将收到“0x32 功能运行 FAIL”의 FLAG.
- DATA(0, 1): 判别文件的有无.
 - 0: 相关文件不存在
 - 1: 相关文件存在
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.31 邀请文件信息(FD)

控制器内邀请文件信息的命令语



● Protocol DATA

PC	STX	Dummy(0xFF)	'F'	'D'	频道	TYPE	文件名称	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	-----	-----

控制器	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	文件名	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

控制器	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	文件名	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

⋮

控制器	STX	Dummy(0xFF)	FLAG(0x30)	JOB NUM	文件名	SIZE	STEP	DATA	ETX	LRC
-----	-----	-------------	------------	---------	-----	------	------	------	-----	-----

PC	ACK
----	-----

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

PC	ACK
----	-----

- 频道(0~2): 邀请文件信息的机器人频道(1Byte)
 - 0: 机器人频道1 .
 - 1: 机器人频道2 .
 - 2: 机器人频道3 .

- TYPE(1Byte): 保存文件位置
 - 0: Backup RAM
 - ※ 写入上面以外的值时返还 FLAG (0x33)
 - ※ N1 控制器储存空间只支持 0

- 文件名称: 要信息的 JOB 文件名称.(12Bytes)
 - EX) 想要称为 RS 的 JOB 文件的信息时

文件名称 (12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'	...	'	'R'	'S'	'.	'J'	'O'	'B'

- 文件名称要写入 5 字以内.
- "*" 输入时关于相关频道的 JOB,PNT 信息全部返还
- ※ 5 字以上时 "0x32 功能运行 FAIL"的 FLAG 将返还.
- ※ 文件名大小字母不能混用.

- JOB NUM(3 Byte): 对相关 JOB 返还号码.
 - EX) JOB 号码为 1 时

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

- 文件名(10 Byte):
 - EX) 返还 RS.JOB 时

文件名(10Bytes)								
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	...	1Byte
'R'	'S'	'.	'J'	'O'	'B'	'	...	'

- SIZE:返还邀请的 JOB 的大小(单位为 Kbyte)
 - EX) 返还 27Kbyte 大小时

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

- STEP(6Byte): 返回 JOB 文件的所有步骤.

EX) JOB 文件返回储存 200 STEP 的 JOB 时

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

- DATA (8Bytes): 功能 增加预定. 返回值为 '0'.

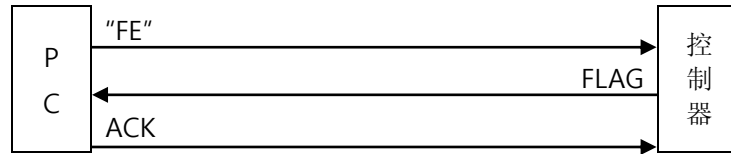
DATA(8 Bytes)							
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
' '	' '	' '	' '	' '	' '	' '	'0'

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.32 删除文件(FE)

删除储存在控制上的文件的命令语



- Protocol Data

PC	STX	Dummy(0xFF)	'F'	'E'	频道	TYPE	文件名称	DATA	ETX	LRC
----	-----	-------------	-----	-----	----	------	------	------	-----	-----

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

PC	ACK
----	-----

- 频道(0~2): 欲删除文件的机器人频道 (1Byte)

- 0: 机器人频道1 .
- 1: 机器人频道2 .
- 2: 机器人频道3 .

- TYPE(1Byte): 储存文件的空间

- 0: Backup RAM
- ※ 写入上面以外的值时返回 FLAG((0x33)
- ※ N1 控制器的储存空间只支持 0

- 文件名称: 删除后的文件名称

EX) 删除称为 RS 的 JOB 文件的情形

文件名称 (12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'	...	'	'R'	'S'	'	'J'	'O'	'B'

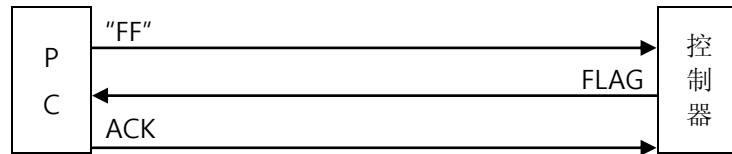
- ※ 现在只能删除JOB文件，删除文件PNT的功能追加预定.
- ※ 文件的扩展不是JOB活着PNT时将返回0x32功能运行FAIL的FLAG

- DATA(1Byte): 功能追加预定.
请输入0
- STX, ETX, LRC, FLAG: 参考4.1通讯规格

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.33 复制文件(FF)

同一频道里复制JOB文件的命令语



- Protocol Data

PC	STX	Dummy(0xFF)	'F'	'F'	频道 1	TYPE	文件名 1	频道 2	文件名 2	ETX	LRC
----	-----	-------------	-----	-----	------	------	-------	------	-------	-----	-----

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

PC	ACK
----	-----

- 频道 1(0~2): 复制对象的机器人频道(1Byte)

- 0: 机器人频道1 .
- 1: 机器人频道2 .
- 2: 机器人频道3 .

- TYPE(1Byte): 保存文件位置

- 0: Backup RAM
- ※ 写入上面以外的值时将返回 FLAG(0x33)
- ※ N1 控制器的储存空间只支持 0

- 文件名 1: 复制对象文件名

EX) 复制称作 RS 的 JOB 文件时的情形

文件名称 (12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'	...	'	'R'	'S'	'.	'J'	'O'	'B'

※ 文件扩展不是JOB或者PNT时将返回0x32功能运行FAIL”的FLAG.

- 频道 2(0~2): 复制后新建的文件机器人频道(1Byte)

- 0: 机器人频道1 .
- 1: 机器人频道2 .
- 2: 机器人频道3 .

- 文件名 2: 粘贴复制的内容的文件名
EX) 以“TT. JOB” 复制时

文件名称 (12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'	...	'	'T'	'T'	'	'J'	'O'	'B'

※ 文件扩展不是JOB或者PNT时返回0x32功能运行 FAIL“的 FLAG

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

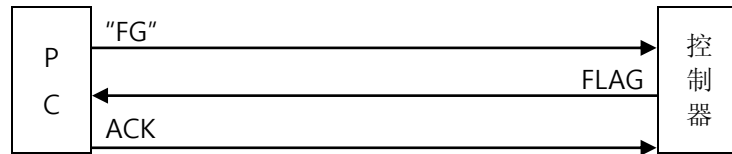
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 CAUTION

- ▶ JOB 复制文件 功能只支持在同一机器人频道里复制
- ▶ 复制对象机器人频道和新建机器人频道不相同RUNFAIL报警将发生

4.2.34 变更文件名(FG)

变更JOB文件名称的命令语



- Protocol Data

PC	STX	Dummy(0xFF)	'F'	'G'	频道	TYPE	文件名称 1	文件名称 2	ETX	LRC
----	-----	-------------	-----	-----	----	------	--------	--------	-----	-----

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

PC	ACK
----	-----

- 频道 (0~2): 要变更文件名称的机器人频道(1Byte)

0: 机器人频道1 .

1: 机器人频道2 .

2: 机器人频道3 .

- TYPE(1Byte): 保存文件位置

0: Backup RAM

※ 写入上面以外的值时将返回 FLAG(0x33)

※ N1 控制器的储存空间只支持 0

- 文件名 1: 将变更文件名称的文件名

EX) 变更称为 RS 的 JOB 文件名称时

文件名称 (13Bytes)									
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
' '	...	' '	'R'	'S'	'.'	'J'	'O'	'B'	' '(0x20)

※ 最后13号Byte以区分者来运用. 区分者没有SPACE(0x20)时不成立

※ 文件扩展不是JOB或者PNT时返回0x32功能运行 "FAIL"的 FLAG

- 文件名 2: 将被变更的文件名
EX) 称作为 RS 的 JOB 文件以 T1 的名称变更时

文件名称 (12Bytes)								
1Byte	...	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
' '	...	' '	'T'	'1'	'.'	'J'	'O'	'B'

※ 文件扩展不是JOB或者PNT时返回0x32功能运行“FAIL”的 FLAG

- STX, ETX, LRC, FLAG: 参考4.1通讯规格

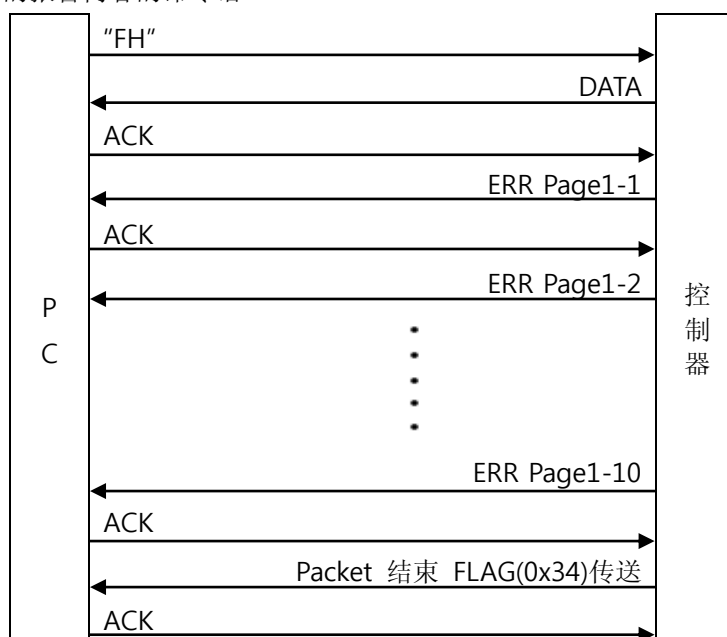
※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

 CAUTION

▶ 要变更名称的JOB文件没在相关频道时RUN FAIL报警会发生

4.2.35 读取已保存的报警内容(FH)

读取储存在N1控制器的报警内容的命令语



- Protocol Data

PC | STX | Dummy(0xFF) | 'F' | 'H' | TYPE | 文件名称 | ETX | LRC |

控制器 | STX | Dummy(0xFF) | FLAG(0x30) | DATA | ETX | LRC |

PC | ACK |

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

PC | ACK |

⋮

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

PC | ACK |

控制器 | STX | Dummy(0xFF) | FLAG(0x34) | ETX | LRC |

PC | ACK |

- TYPE(1Byte): 保存文件位置
0: Backup RAM
- ※ 写入上面以外的值时返还 FLAG((0x33)
- ※ N1 控制器的储存空间只支持 0
- 文件名称(30Byte): ERR message 文件邀请

文件名称 (30Bytes)
"alarm_history.txt"

※ 30Byte中 除去"alarm_history.txt"部分的空空间请插入SPACE(0x20)文字

- DATA(40Byte)

DATA (36Bytes)
"NO.WtERROR TIMEWtCH ERROR MSGWt(CODE) "

- ERR Page : 储存在控制器上的ERR内容每页最大返还10个。根据ERROR 内容最大可以返还76Byte

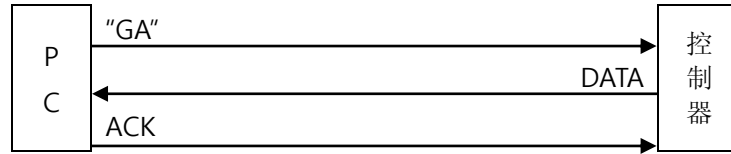
ERR Page(MAX 76Byte)									
%02dWt[%dD %02d:%02d:%02d]WtCH%d - %s ,%sWt(%4d) "									
PAGE	INDEX	DAY	时	分	秒	报警 频道	报警 内容	详细内容	报警线
"%02d	%02d	[%dD	%02d	:%02d	:%02d]	CH%d	- %s	,%s	(%4d) "

- PAGE: 储存报警内容, 最近发生的报警按照顺序储存
- INDEX:报警发生时储存同一报警内容
- 日期, 时间, 分, 秒: 控制器以基准储存WORK TIMER.(关于WORK TIMER的详细内容请参考“操作 运用 说明书 1.3.4.1 W.TIMER”)
- 报警 频道: 表示报警发生的频道을. 假如返还9时全体频道都会发生报警
- 报警内容: 发生的报警内容.
- 详细内容: 对仅报警内容无法判断的报警内容附加说明.
- 报警线: 关于报警内容的线号码
- ※ 关于ERR 内容 储存的详细内容请参考“程序 说明书 3.41 RERROR” 命令语
- ※ ERR原因或者解决方法 请参考“报警线 说明书”
- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”.

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.36 读取I/O卡IN PUT接点状态(GA)

读取I/O카드写入接点状态信息的命令语 .



- Protocol Data

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

控制器	STX	FLAG	PORT 接点数	INPUT 状态 信息	ETX	LRC
-----	-----	------	----------	-------------	-----	-----

PC	ACK
----	-----

- 写入 PORT(0~4): 选择写入 PORT(2Byte).

输入PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- PORT 接点数 (2Byte): 返还选择的写入PORT的接点数.

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

- INPUT 状态 信息(4Byte): 返还选择的写入PORT接点的状态信息.

INPUT 状态 信息(4Bytes)			
1Byte	1Byte	1Byte	1Byte

接点状态 信息(0~7) (2Bytes)															
1Byte								1Byte							
0~3接点								4~7接点							
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

※ 接点数据为 ON:1 , OFF:0

接点状态 信息(8~15) (2Bytes)															
1Byte								1Byte							
8~11接点								12~15接点							
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

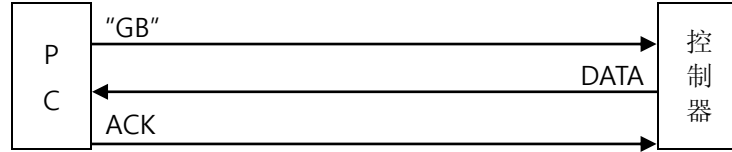
※ 接点数据为 ON:1 , OFF:0.

- STX, ETX, LRC, FLAG:请参考“4.1通讯规格”

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.37 读取I/O卡OUT PUT接点状态信息 (GB)

读取I/O卡输出接点状态信息的命令语.



- Protocol Data

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

控制器	STX	FLAG	PORT接点数	OUT状态 信息	ETX	LRC
-----	-----	------	---------	----------	-----	-----

PC	ACK
----	-----

- 输出 PORT(0~4):选择输出 PORT.

输出 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- PORT 接点数: 返还选择的输出PORT的接点数

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

- OUT 状态 信息: 返还选择的输出PORT接点的状态信息

OUT 状态 信息(4Bytes)			
1Byte	1Byte	1Byte	1Byte

接点状态 信息(0~7) (2Bytes)															
1Byte								1Byte							
0~3接点								4~7接点							
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

※ 节点数据为 ON:1, OFF:0.

接点状态信息(8~15) (2Bytes)															
1Byte								1Byte							
8~11接点								12~15接点							
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

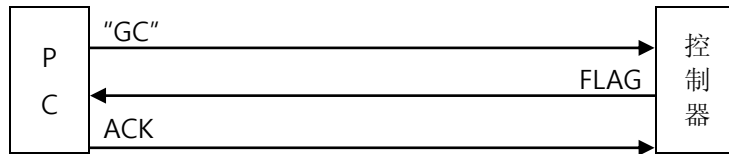
※ 接点数据为 ON:1 , OFF:0.

- STX, ETX, LRC, FLAG: 请参考 “4.1通讯 规格”.

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.38 I/O卡OUT PUT接点输出(GC)

写I/O接点数据的命令语



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'C'	输出 PORT	接点号码	DATA	ETX	LRC
----	-----	-------------	-----	-----	---------	------	------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 输出 PORT(0~4): 选择输出PORT.

输出 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

- 接点号码 (0~15): 选择输出接点.

接点号码(2Bytes)	
1Byte	1Byte
'0'	'1'

- DATA: 选择接点输出状态.

0: 接点 OFF

1: 接点 ON

DATA
1 Byte
'0'or'1'

EX) 1 PORT的12号输出节点做成 ON的状态时

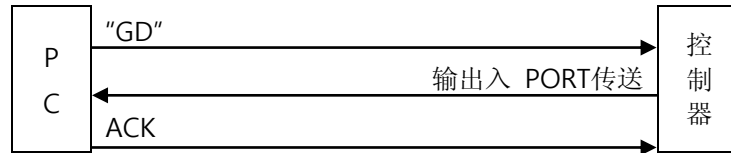
STX	Dummy	'G'	'C'	输出 PORT		接点号码		DATA	ETX	LRC
				'0'	'1'	'1'	'2'	'1'		

- STX, ETX, LRC, FLAG: 请参考“4.1通讯规格”

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

4.2.39 读取I/O卡信息 (GD)

读取设定在控制器上的输出输入BOARD设定的值的命令语



- Protocol Data

PC

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

控制器

STX	FLAG	输入 BOARD	输出 BOARD	ETX	LRC
-----	------	----------	----------	-----	-----

PC

ACK

- 输入 BOARD: 控制器 I/O BOARD 设定 值(2Byte)

输入 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

告知设定在控制器上的 I/O BOARD个数

- 输出 BOARD: 控制器 I/O BOARD 设定 值(2Byte)

输出 PORT(2Bytes)	
1Byte	1Byte
'0'	'1'

告知设定在控制器上的I/O BOARD个数

- STX, ETX, LRC, FLAG: 参考“4.1通讯 规格”.

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

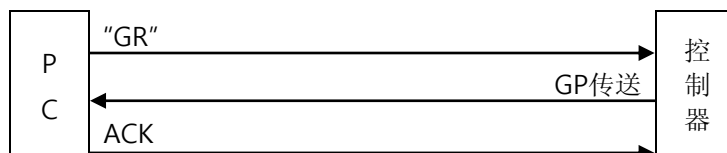


- ▶ N1 控制器I/O Board的个数只能设定在 0~2.
- ▶ 请参考“操作 及 运用 说明书 1.3.1.3 扩张 I/O BOARD 个数 设定”

4.2.40 读取全局变数 Data (GR)

读取全局变数(I, F, GP)上数据的命令语

4.2.40.1 读取全局点位



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'R'	Type	频道	坐标系	轴号码	INDEX	ETX	LRC
----	-----	-------------	-----	-----	------	----	-----	-----	-------	-----	-----

控制器	STX	FLAG	'0'	Global Point Data	ETX	LRC
-----	-----	------	-----	-------------------	-----	-----

PC	ACK
----	-----

- Type(0~2): 选择Global 变数(I, F, GP)

0: Global Point

1: Global Integer

2: Global Float

- 频道: 选择动作中的机器人频道.

0: 机器人频道1 .

1: 机器人频道2 .

2: 机器人频道3 .

- 坐标系: 选择要接受的数据类型

0: Angle Data

1: XY坐标 Data

- 轴号

0: 选择所有轴(1~6轴)

1: 1轴(X)选择.

2: 2轴(Y)选择.

3: 3轴(Z)选择.

4: 4轴(W)选择.

5: 5轴(E1)选择.

6: 6轴(E2)选择.

- INDEX: 为Global Point 号码. 可以写入在0~1023内.

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

EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

- Global Point Data: 返回 的Data为下.

- 轴号码为0时

返回和6轴相关的Data(6 X 10Byte+USED+ARM=62Byte)

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

- 轴号码为 1~6时

返回和选择 轴相关的Data(1 X 10Byte+USED=11Byte)

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决定使用有无

0: Point Data 不能使用

1: Point Data 使用可以

- ARM(0~2):

0: LEFT_form

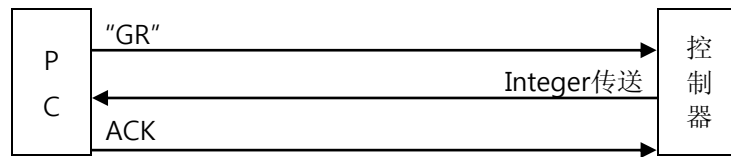
1: RIGHT_form

2: No_form

- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.40.2 读取全局整数



- Protocol Data

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

控制器	STX	FLAG	Integer Data	ETX	LRC
-----	-----	------	--------------	-----	-----

PC	ACK
----	-----

- 选择类型(0~2): Global 变数(I, F, GP)
0: Global Point
1: Global Integer
2: Global Float
- INDEX: .可以输入在0-499以内
- EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

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

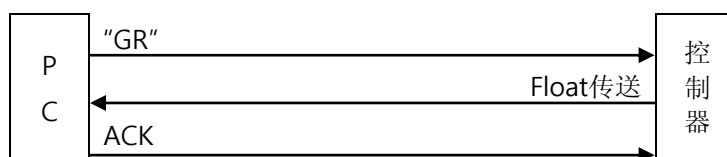
- DATA: 返还Global Integer Data.

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

- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.40.3 读取全局实数



- Protocol Data

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

控制器	STX	FLAG	Float Data	ETX	LRC
-----	-----	------	------------	-----	-----

PC	ACK
----	-----

- Type(0~2): 选择Global变数(I, F, GP).
 0: Global Point
 1: Global Integer
2: Global Float
- INDEX: 可以在0~499之内输入 .
- EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

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

- 返还DATA: Global Float Data.

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

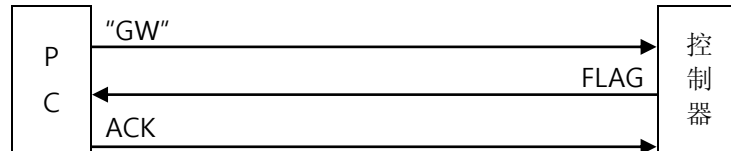
- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.41 写全局变数 数据 (GW)

全局变数(I, F, GP)上写数据的命令语 .

4.2.41.1 保存全局点位



- Protocol Data

PC	STX	Dummy(0xFF)	'G'	'W'	Type	频道	坐标系	轴号码	INDEX	DATA	ETX	LRC
----	-----	-------------	-----	-----	------	----	-----	-----	-------	------	-----	-----

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- Type(0~2): 选择Global 变数(I, F, GP).

0: Global Point

- 1: Global Integer
- 2: Global Float

- 频道: 选择将储存的机器人频道.

- 0: 机器人频道1 .
- 1: 机器人频道2 .
- 2: 机器人频道3 .

- 坐标系: 选择欲对控制器传送的数据类型.

- 0: Angle Data
- 1: XY坐标 Data

※ 一般直角机器人就算选择Angle坐标系也用XY坐标系来储存.

- 轴号码: 选择欲储存的轴.

- 0: 选择所有 轴(1~6轴).
- 1: 1轴(X)选择.
- 2: 2轴(Y)选择.
- 3: 3轴(Z)选择.
- 4: 4轴(W)选择.

5: 5轴(E1 选择).

6: 6轴(E2)选择.

- INDEX:为 Global Point Index号码. 可以再0~1023内输入.
 . EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

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

- DATA: 输入欲储存的数据.
 - 轴号码为0时
 传送和6轴相关的Data(6 X 10Byte+USED+ARM=62Byte)

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

假如欲输入数据的频道时4轴时在5轴,6轴输入 "0.0".

- 轴号码为 1~6时
 传送和选择 轴相关的Data(1 X 10Byte+USED+ARM=10Byte).

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决定使用有无.

0: Point Data 不使用

1: Point Data 可以使用

- ARM(0~2):

0: LEFT_form

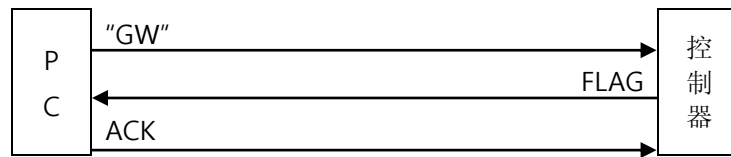
1: RIGHT_form

2: 储存机器人的现在姿势.

- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.41.2 储存全局定数



- Protocol Data

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

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- 类型(0~2): 选择Global 变数(I, F, GP).
0: Global Point
1: Global Integer
2: Global Float
- INDEX: 可以在0~499까지内输入.
EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

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

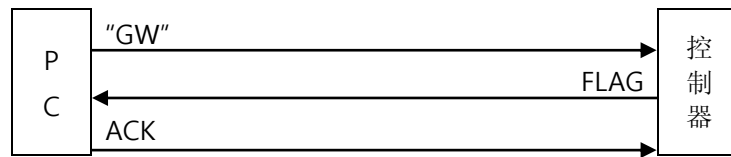
- DATA: 可以输入的定数数据长度为6Bytes.

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

- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.41.3 储存全局实数



- Protocol Data

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

控制器	STX	FLAG	ETX	LRC
-----	-----	------	-----	-----

PC	ACK
----	-----

- TYPE(0~2): 选择Global 变数(I, F, GP).

0: Global Point

1: Global Integer

2: Global Float

- INDEX: 可以再0~499까지之内输入.

. EX) "45" 写入时 10000的位置 1000的位置和 100的位置时请插入 SPACE(0x20) 字母

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

- DATA(10Byte): 用实数型数据 可以在小数点第三位以内任意输入.

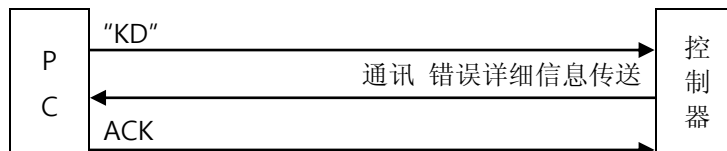
DATA(10Byte)									
1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte	1Byte
'1'	'2'	'3'	'4'	'5'	'6'	'.'	'1'	'2'	'3'

- STX, ETX, LRC, FLAG: 请参考"4.1通讯规格".

※ Protocol命令后发生通讯故障时利用"通讯故障原因读取(KD)"协议读取原因.

4.2.42 读取通讯错误原因(KD)

确认关于功能失败(0x32)的错误内容的命令语.



- Protocol Data

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

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

PC	ACK
----	-----

- Text: 关于通讯错误原因的详细内容以文字列的方式传送
EX) LRC值有问题时像如下一样输出.

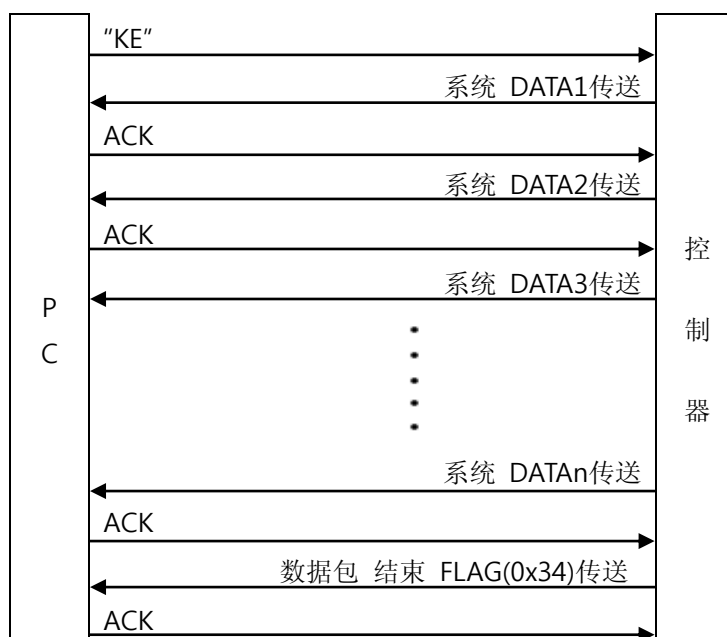
Text
"LRC is different with received data LRC"

- STX, ETX, LRC, FLAG: 参考 "4.1通讯 规格".

CAUTION

- ▶ 因为通讯ERRtable全频道间共同使用所以返还最后发生的ERR信息

4.2.43 读取系统数据(KE)



● Protocol Data

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

控制器	STX	Dummy(0xFF)	FLAG	DATA1	ETX	LRC
-----	-----	-------------	------	-------	-----	-----

PC	ACK
----	-----

控制器	STX	Dummy(0xFF)	FLAG	DATA2	ETX	LRC
-----	-----	-------------	------	-------	-----	-----

⋮

控制器	STX	Dummy(0xFF)	FLAG	DATA n	ETX	LRC
-----	-----	-------------	------	--------	-----	-----

PC	ACK
----	-----

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

PC	ACK
----	-----

- 频道:
 - 0: 频道 1号 PARA Data 邀请
 - 1: 频道 2号 PARA Data 邀请
 - 2: 频道 3号 PARA Data 邀请
 - 4: PUB_PARA Data 邀请
- INDEX(2Byte): Parameter DATA 选择

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

※ 相关频道设定为BGT类型时和INDEX无关只返还BGT PARA内容

- Parameter Data 个别邀请时参考下列常数TABLE.

PARA				PUB_PARA	
因子	内容	因子	内容	因子	内容
-1	所有 Data 邀请	52	FOW	-1	所有 Data 邀请
1	CONF	53	OVS	1	TMR
2	LENG	54	OVT	2	HOST-S
3	RANG	55	OVA	3	FDBUS
4	OFFS	56	IPPE	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				

※ 上面PARAM和PUB_PARAM的详细内容请参考“操作运营 说明书”

- DATA: 根据邀请的内容数据的长度也会被变更.
- STX, ETX, LRC, FLAG: 参考“4.1通讯 规格”

※ Protocol命令后发生通讯故障时利用“通讯故障原因读取(KD)”协议读取原因.

Rev.	修改日期	内容	修改人	S/W Version
V.1	2012.12.30	初版 印刷		

N1 ROBOT CONTROLLER

CONTROLLER MANUAL

FIRST EDITION OCTOBER 2012

ROBOSTAR CO, LTD

ROBOT R&D CENTER
