



ROBO CYLINDER シリーズ
PCON、ACON、SCON、ERC2

シリアル通信【Modbus 版】

取扱説明書 第3版

**R ROBO
C CYLINDER**

はじめに

本書は、シリアル通信の手順に限った説明書となっておりますので、他の制御、設置・接続等に関しましてはロボシリンダ用コントローラ(以下 RC コントローラ)に付属している取扱説明書をご覧ください。

注意

- (1) 本仕様書に掲載されていないアドレスまたは機能を RC コントローラに送信した場合には、RC コントローラが正常動作できなくなる、または意図しない動きをする可能性があります。指定外の機能及びアドレスへの送信は行わないでください。
- (2) RC コントローラは、SIO ポートより 150[msec]以上のブレイク(スペース)信号を検出するとボーレートが 9600[bps]に切り替わる仕様となっております。
一部のパソコンでは通信ポート非オープン時、送信ラインがブレイク(スペース)状態となっているものがあります。このようなパソコンを上位機器として使用する場合、意図せずに RC コントローラのボーレートが 9600[bps]となっている場合がありますのでご注意ください。
- (3) 通信速度等のパラメータ設定は弊社のティーチングツール(ティーチングボックスまたはパソコン対応ソフト)を用いて、予め設定を行ってから通信を行ってください。
- (4) 下記の場所で使用する際は、遮蔽対策を十分行ってください。措置しない場合は、誤作動を起こす可能性があります。
 - ① 大電流や高磁界が発生している場所
 - ② 溶接作業などアーク放電の生じる場所
 - ③ 静電気などによるノイズが発生する場所
 - ④ 放射能に被爆する可能性がある場所
- (5) 配線を行ったり、各コネクタの抜き差しの際には、上位側、各 RC コントローラの電源を OFF して行ってください。電源を ON したまま行くと感電、部品の破損を招く恐れがあります。



- (6) ノイズによる誤動作を防止する為、通信の配線は動力線や他の制御用配線と分離して配線してください。
- (7) ノイズによる誤動作を防止する為、同一電源路 あるいは同一装置内の電気機器にはノイズ防止対策を施してください。

本文中における会社名、商品名は各社の商標 または 登録商標です。

目次

1	概要	1
2	仕様	2
2.1	通信方式	3
3	通信までの準備	4
3.1	上位が RS232C インタフェースの場合	4
(1)	システム構成	4
(2)	配線	5
(3)	SIO 変換器(垂直仕様:RCB-TU-SIO-A、水平仕様:RCB-TU-SIO-B)	7
3.2	上位が RS485 インタフェースの場合	8
(1)	システム構成	8
(2)	配線	9
3.3	PLC 及び パソコンの通信コネクタ ピンアサイン(参考)	10
3.4	通信を行うまでの各種設定	11
3.5	軸番号の設定	12
3.6	コントローラの通信速度を設定	13
4	通信	15
4.1	メッセージ送信タイミング	15
4.2	タイムアウトとリトライ	16
4.3	RC コントローラの内部アドレス 及び データ構造	17
4.3.1	データレジスタの構造	17
4.3.2	データレジスタ詳細	18
(1)	デバイス制御レジスタ 1 内容 (DRG1)	19
(2)	デバイス制御レジスタ 2 内容 (DRG2)	20
(3)	ポジション番号指定レジスタ内容(POSR)	21
(4)	デバイスステータスレジスタ 1 内容(DSS1)	22
(5)	デバイスステータスレジスタ 2 内容(DSS2)	23
(6)	拡張デバイスステータスレジスタ内容(DSSE)	24
(7)	システムステータスレジスタ内容(STAT)	25
(8)	特殊ポートモニタレジスタ内容(SIPM)	26
(9)	ゾーンステータスレジスタ内容(ZONS)	27



(10) ポジション番号ステータスレジスタ内容(POSS).....	28
4.3.3 ステータスレジスタの構造.....	29
4.3.4 ステータスレジスタ詳細.....	30
5 Modbus RTU.....	33
5.1 メッセージフレーム(クエリ、レスポンス).....	34
5.2 RTUモード クエリ一覧.....	37
5.3 データ、ステータス読み取り(コード 03 使用クエリ).....	41
5.3.1 レジスタの連続複数読み取り.....	41
5.3.2 現在位置の読み取り《PNOW》(0.01mm 単位).....	44
5.3.3 現在発生アラームコード読み取り《ALMC》.....	46
5.3.4 I/Oポート入力信号状態読み取り《DIPM》.....	48
5.3.5 I/Oポート出力信号状態の読み取り《DOPM》.....	52
5.3.6 コントラ状態信号の読み取り1《DSS1》.....	56
5.3.7 コントラ状態信号の読み取り2《DSS2》.....	58
5.3.8 コントラ状態信号の読み取り3《DSSE》.....	60
5.3.9 コントラ状態信号の読み取り4《STAT》.....	62
5.3.10 現在速度の読み取り《VNOW》.....	64
5.3.11 電流値の読み取り《CNOW》.....	66
5.3.12 偏差の読み取り《DEVI》.....	68
5.3.13 電源投入後の積算時間の読み取り《STIM》.....	70
5.3.14 特殊入力ポートの入力信号状態読み取り《SIPM》.....	72
5.3.15 ゾーン出力信号の状態読み取り《ZONS》.....	74
5.3.16 位置決め完了ポジションNo.の読み取り《POSS》.....	76
5.4 動作指令 及び、データ書き換え(コード 05 使用クエリ).....	78
5.4.1 コイルへの書込み.....	78
5.4.2 セーフティ速度有効/無効切替《SFTY》.....	79
5.4.3 サーボON/OFF《SON》.....	81
5.4.4 アラームリセット《ALRS》.....	83
5.4.5 ブレーキ強制解除《BKRL》.....	85
5.4.6 一時停止《STP》.....	87
5.4.7 原点復帰《HOME》.....	89
5.4.8 位置決め動作起動指令《CSTR》.....	91
5.4.9 ジョグ/インチング切替《JISL》.....	93
5.4.10 ティーチモード指令《MOD》.....	95
5.4.11 ポジションデータ取込み指令《TEAC》.....	97

5. 4. 12	ジヨグ+指令 《JOG+》	99
5. 4. 13	ジヨグ-指令 《JOG-》	101
5. 4. 14	スタートポジション 0~7《ST0~ST7》移動指令 (PIO パターン 4, 5 限定)	103
5. 4. 15	PIO/Modbus 切替設定 《PMSL》	105
5. 4. 16	減速停止 《STOP》	107
5. 5	制御情報の直接書き込み(コード 06 使用ケリ)	109
5. 5. 1	レジスタへの書き込み	109
5. 6	位置決めデータの直接書き込み(コード 10 使用ケリ)	112
5. 6. 1	直値移動指令	112
5. 6. 2	ポジションテーブルデータ書き込み	123
6	Modbus ASCII	131
6. 1	メッセージフレーム(ケリ、レスポンス)	132
6. 2	ASCII コード表	135
6. 3	ASCII モード ケリ一覧	136
6. 4	データ、ステータス読み取り(コード 03 使用ケリ)	140
6. 4. 1	レジスタの連続複数読み取り	140
6. 4. 2	現在位置の読み取り(0.01mm 単位) 《PNOW》	143
6. 4. 3	現在発生アラームコードの読み取り 《ALMC》	145
6. 4. 4	I/O ポート入力信号の状態読み取り 《DIPM》	147
6. 4. 5	I/O ポート出力信号の状態読み取り 《DOPM》	151
6. 4. 6	コントローラ状態信号の読み取り 1 《DSS1》	155
6. 4. 7	コントローラ状態信号の読み取り 2 《DSS2》	157
6. 4. 8	コントローラ状態信号の読み取り 3 《DSSE》	159
6. 4. 9	コントローラ状態信号の読み取り 4 《STAT》	161
6. 4. 10	現在速度の読み取り 《VNOW》	163
6. 4. 11	電流値の読み取り 《CNOW》	165
6. 4. 12	偏差の読み取り 《DEVI》	167
6. 4. 13	電源投入後の積算時間の読み取り 《STIM》	169
6. 4. 14	特殊入力ポートの入力信号状態の読み取り 《SIPM》	171
6. 4. 15	ゾーン出力信号の状態の読み取り 《ZONS》	173
6. 4. 16	位置決め完了ポジションNo.の読み取り 《POSS》	175
6. 5	動作指令 及び、データ書き換え(コード 05 使用ケリ)	177
6. 5. 1	コイルへの書き込み	177
6. 5. 2	セーフティ速度有効/無効切替 《SFTY》	178
6. 5. 3	サーボ ON/OFF 《SON》	180

R ROBO C CYLINDER

6.5.4	アラームリセット《ALRS》.....	182
6.5.5	ブレーキ強制解除《BKRL》.....	184
6.5.6	一時停止《STP》.....	186
6.5.7	原点復帰《HOME》.....	188
6.5.8	位置決め動作起動指令《CSTR》.....	190
6.5.9	ジヨグ／インチング切替《JISL》.....	192
6.5.10	ティーチモード指令《MOD》.....	194
6.5.11	ポジションデータ取込み指令《TEAC》.....	196
6.5.12	ジヨグ+指令《JOG+》.....	198
6.5.13	ジヨグ-指令《JOG-》.....	200
6.5.14	スタートポジション0～7《ST0～ST7》(PIOパターン4、5 限定).....	202
6.5.15	PIO/Modbus 切替設定《PMSL》.....	204
6.5.16	減速停止《STOP》.....	206
6.6	制御情報の直接書き込み(コード06 使用ケリ).....	208
6.6.1	レジスタへの書き込み.....	208
6.7	位置決めデータ直接書き込み(コード10 使用ケリ).....	211
6.7.1	直値移動指令.....	211
6.7.2	ポジションテーブルデータ書き込み.....	222
7	トラブルシューティング	229
7.1	異常時の返信(例外レスポンス)について.....	230
7.2	注意事項.....	233
7.3	通信がうまくいかない時は.....	234
8	参考資料	241
8.1	CRC チェック計算.....	242
8.2	SIO と PIO の併用システム構成.....	244

1 概要

ロボシリンダ用コントローラ(以下 RC コントローラ)はホスト(上位コントローラ)とのインタフェースに EIA RS485 に準拠した調歩同期式シリアルバスインタフェースを装備しています。このインタフェースによって、最大 16 軸までのスレーブ(RC コントローラ)を接続(※1)し、制御できる SIO リンクシステムを構築することができます。

各軸毎に指令を出すことはもちろん、全てのスレーブに同時に同じ指令を出すことも可能です。

通信プロトコルには Modbus Protocol を採用し、ホストから指令を出したり、内部情報の参照を行えます。

Modbus Protocol は、仕様が全世界に公開されていますので、ソフトウェア開発が手軽に行えます。

(※1) 同一回線上に旧 RC シリーズ(プロトコル T)や RC シリーズ以外の機器を接続することはできませんのでご了承ください。

シリアル伝送モードには ASCII モード(1 バイト(8 ビット)データを ASCII コード(2 文字)に変換して伝送)と、RTU モード(1 バイト(8 ビット)データをそのまま伝送)の 2 種類がありますが、RC コントローラ(ROBO NET コントローラ除く)においては 1 パケットごとに伝送モードを判定しており、どちらのモードでも受信可能(※2)となっています。ROBONET では RTU モードに限り受信可能となっており、ASCII モードには対応していませんのでご注意ください。

(※2)1 つのネットワーク上では、全てのデバイスで同一モードで御使用ください。混在はできません。

☆ 制御可能なコントローラ

- ERC2
- PCON-C / CG / CF / CY / SE / PL / PO
- ACON-C / CG / CY / SE / PL / PO
- SCON

2 仕様

項目	方式・条件
インターフェイス	EIA RS485 準拠
通信方式	半二重通信
最大総延長距離	100m
同期方式	調歩同期式
接続形態	1:N 不平衡バス接続 (1 ≤ N ≤ 16)
伝送モード	RTU/ASCII(自動判別) ※
通信速度(bps)	パラメータ設定により下記速度から選択可能 9600,14400,19200,28800,38400 57600,76800,115200,230400
ビット長	8ビット
ストップビット	1ビット
パリティ	なし

※ ROBONET は ASCII モードには対応していません。

2.1 通信方式

Modbus Protocol の通信方式は、シングルマスタ/マルチスレーブ方式です。マスタ(上位:以下の例では PLC)が指定したスレーブ(以下の例では C 軸に接続された RC コントローラ)にクエリを発行し、指定されたスレーブは、このクエリを受けて、指定された機能を実行し、レスポンスメッセージを返します。(これで 1 回の通信サイクルが終了します。)

クエリの伝送フォーマットは、スレーブのアドレス、要求内容を定義するファンクションコード、データおよびエラーチェックから構成されています。

また、レスポンスメッセージの伝送フォーマットは、要求内容の確認ファンクションコード、データおよびエラーチェックから構成されています。クエリとレスポンスメッセージの伝送構造を下図に示します。

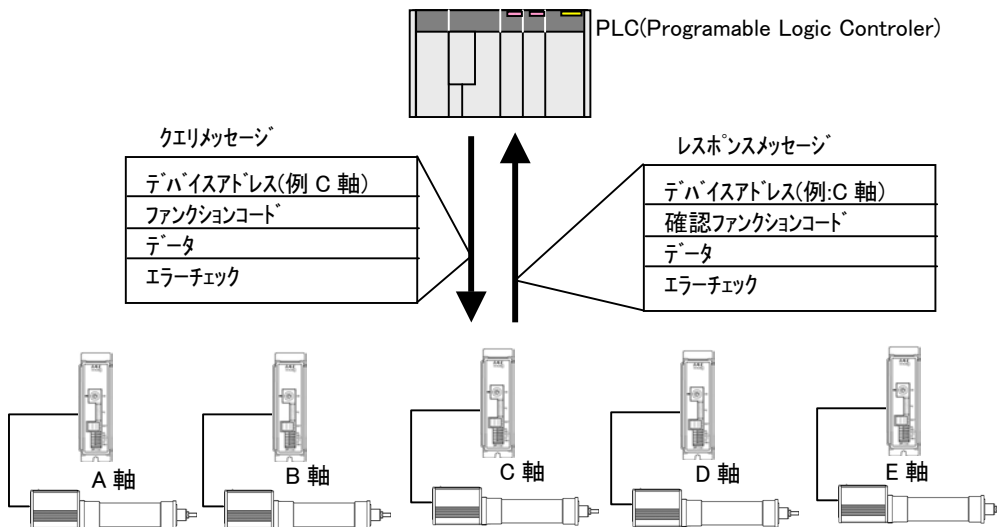


図 2.1

3 通信までの準備

3.1 上位が RS232C インタフェースの場合

(1) システム構成

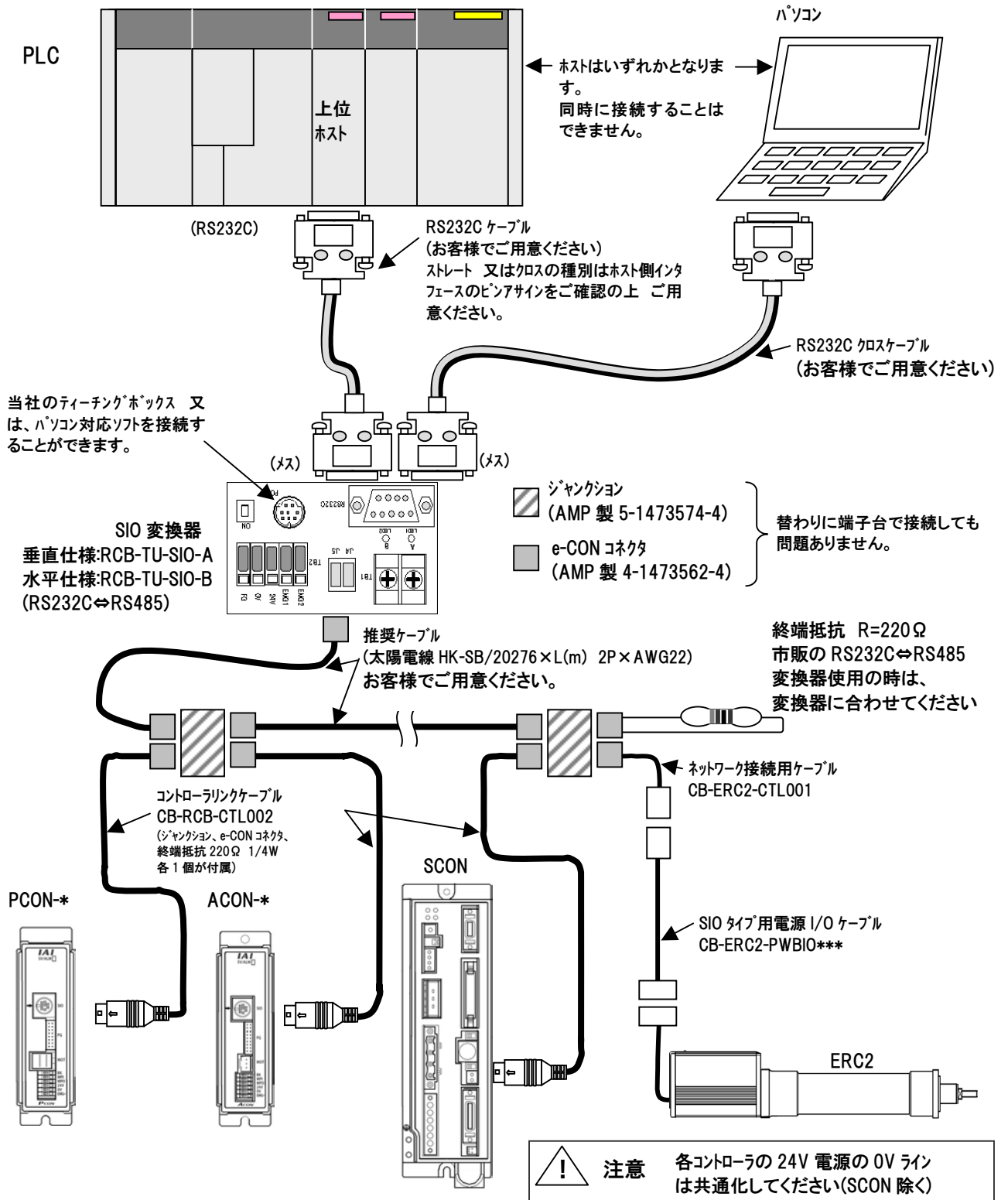
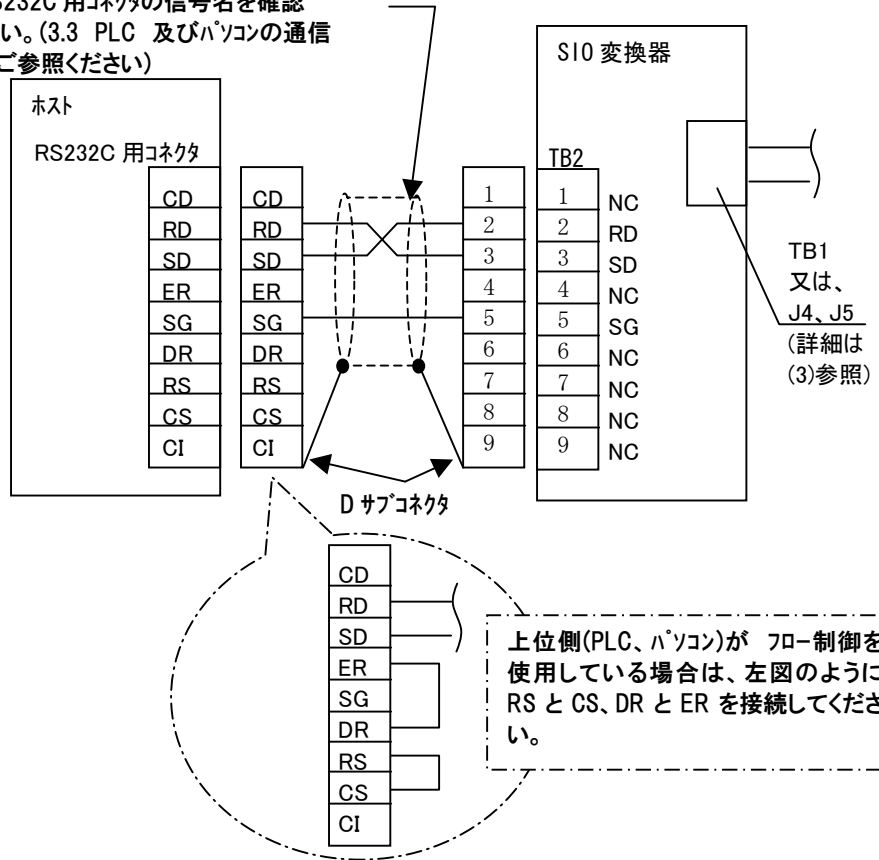


図 3.1

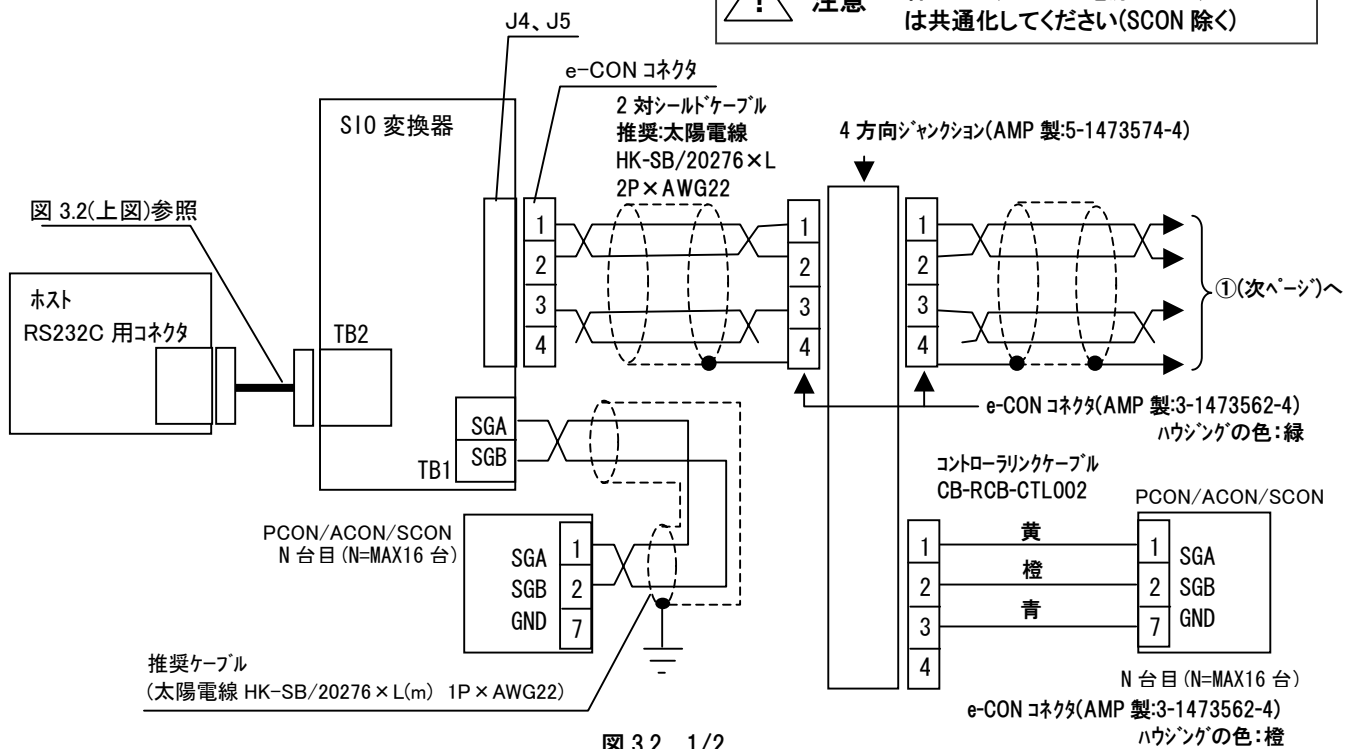
(2) 配線

RS232C ケーブル(市販品等)

必ず、ホスト側の RS232C 用コネクタの信号名を確認して接続してください。(3.3 PLC 及びパソコンの通信コネクタ ピンサインをご参照ください)



注意 各コントローラの 24V 電源の 0V ラインは共通化してください(SCON 除く)



ROBO CYLINDER

注意 各コントローラの 24V 電源の 0V ラインは共通化してください(SCON 除く)

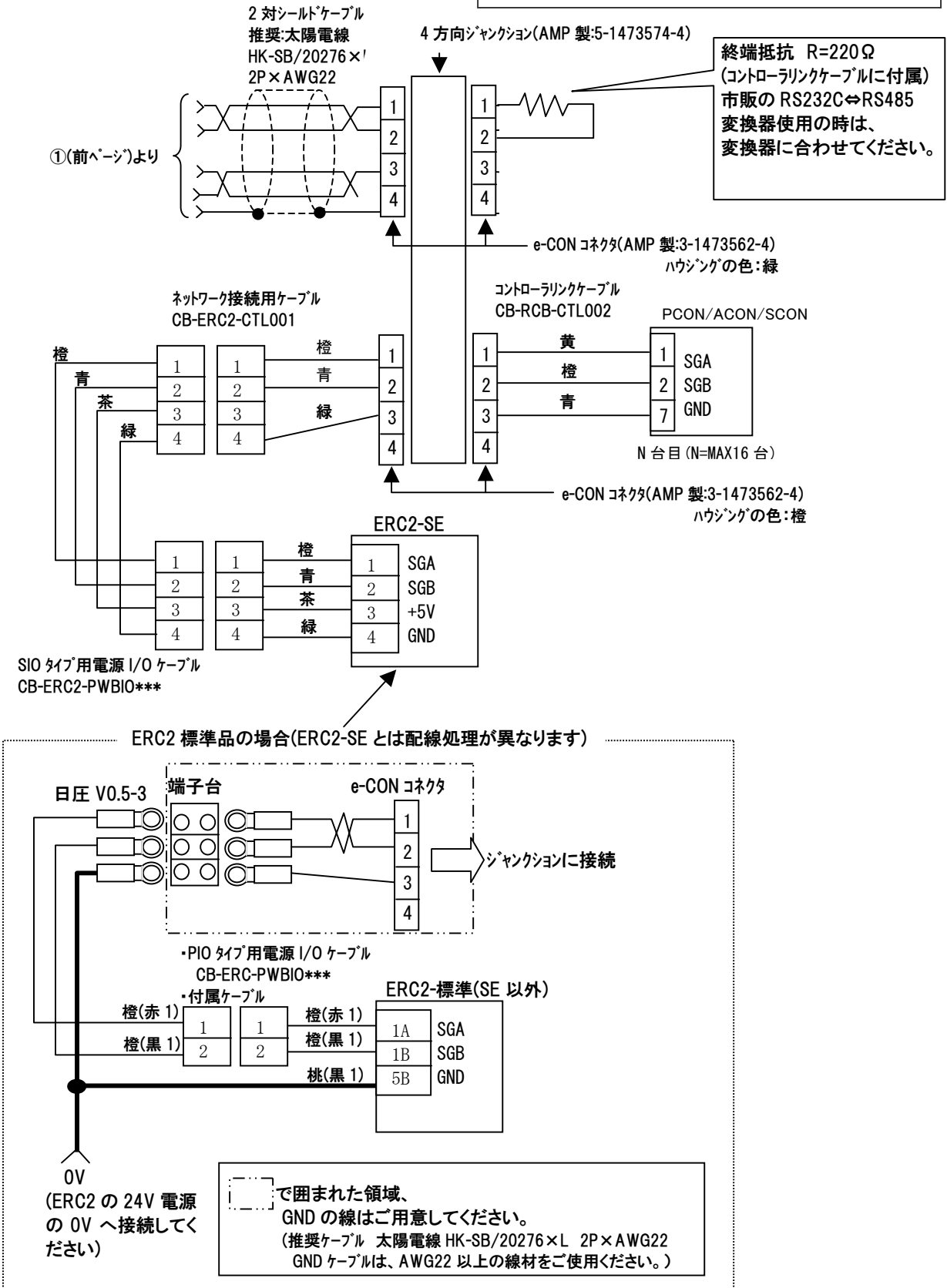
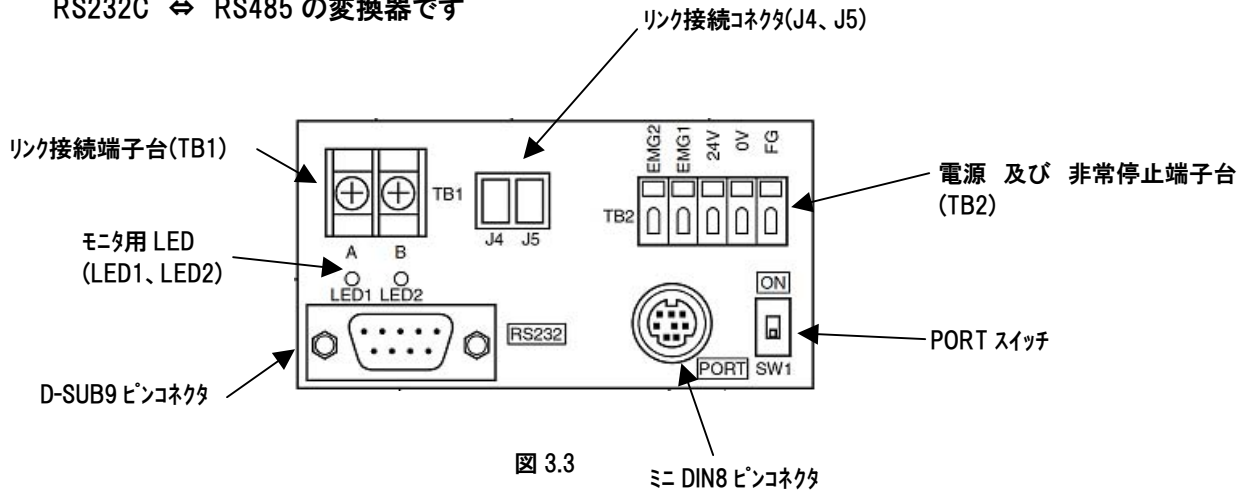


図 3.2_2/2

(3) SIO 変換器(垂直仕様:RCB-TU-SIO-A、水平仕様:RCB-TU-SIO-B)
RS232C ⇔ RS485 の変換器です



◎電源 及び 非常停止端子台(TB2)

- ・EMG1、EMG2: テーチングボックスの非常停止スイッチの接点出力
PORT スイッチが ON 側でテーチングボックスの非常停止スイッチに接続され、
OFF 側では EMG1 と EMG2 は 短絡されます。
 - ・24V: +24V 電源を供給してください。(消費電流 0.1A 以下)
 - ・0V : 0V 電源を供給してください。(DC24V 仕様のコントローラの 0V と共通にしてください)
 - ・FG : FG を接続する端子です。
- ※適合電線 単線 Φ0.8~1.2mm、
撚り線 AWG18~20(ストリップ長 10mm、先端部半田処理)

◎リンク接続端子台(TB1)

- RC コントローラとリンク接続するための接続口です。
- ・A: RC コントローラの通信コネクタの 1 ピン(SGA)に接続します。
 - ・B: RC コントローラの通信コネクタの 2 ピン(SGB)に接続します。

◎D-sub9 ピンコネクタ

上位(ホスト)側との接続口です。

◎ミニ DIN8 ピンコネクタ

テーチングボックス、パソコン対応ソフトとの接続口です。

◎PORT スイッチ

- ・ON: テーチングボックス使用
- ・OFF: テーチングボックス未使用

◎モータ用 LED(LED1、LED2)

- ・LED1: RC コントローラが送信中のときに点灯/点滅します。
- ・LED2: 上位(ホスト)側が送信中のときに点灯/点滅します。

◎リンク接続コネクタ(J4、J5)

RC コントローラとリンク接続するための接続口です。
オプションのリンクケーブル(CB-RCB-CTL002)をそのまま接続できます。

3.2 上位が RS485 インタフェースの場合

(1) システム構成

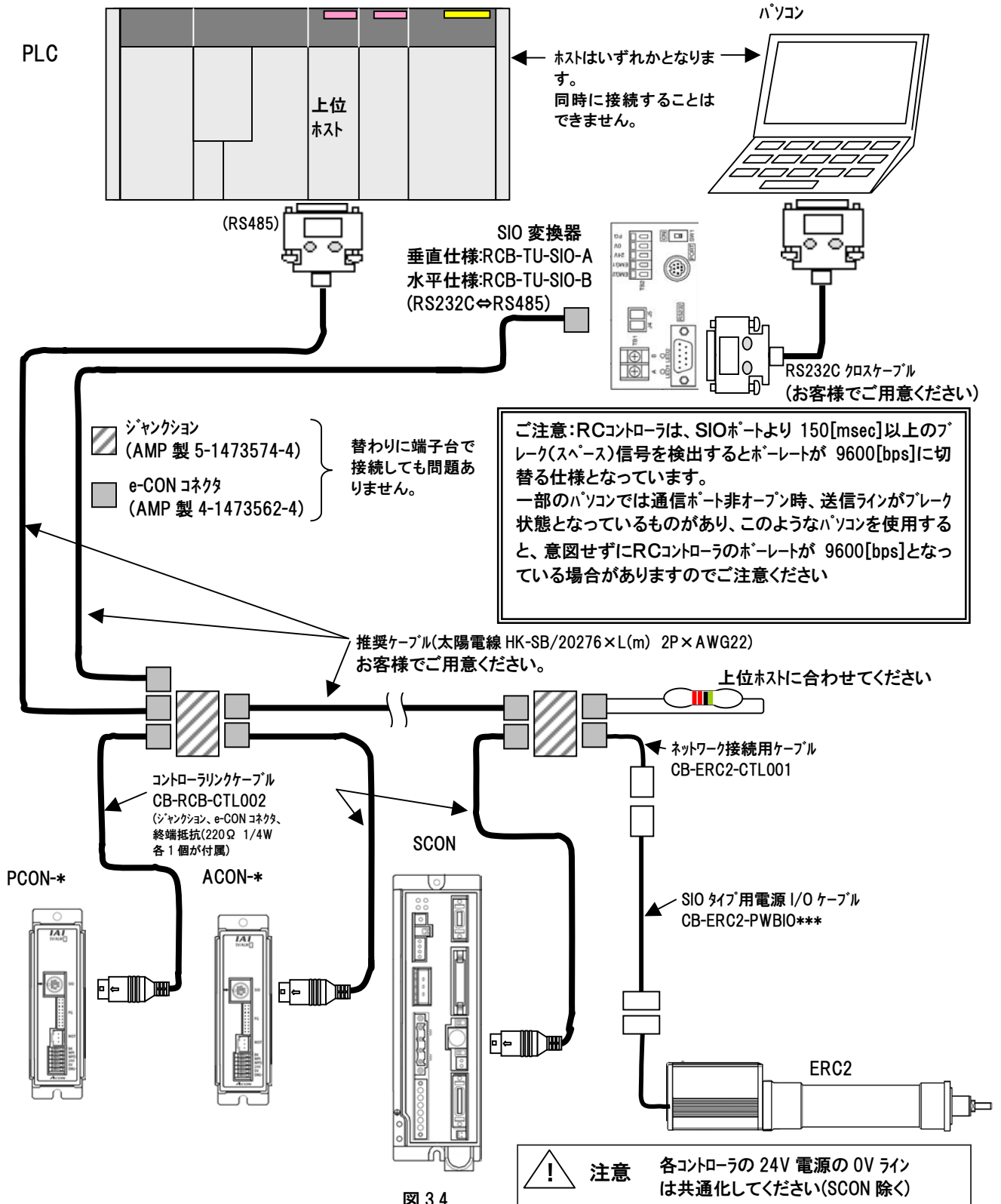


図 3.4

(2) 配線

注意 各コントローラの 24V 電源の 0V ラインは共通化してください(SCON 除く)

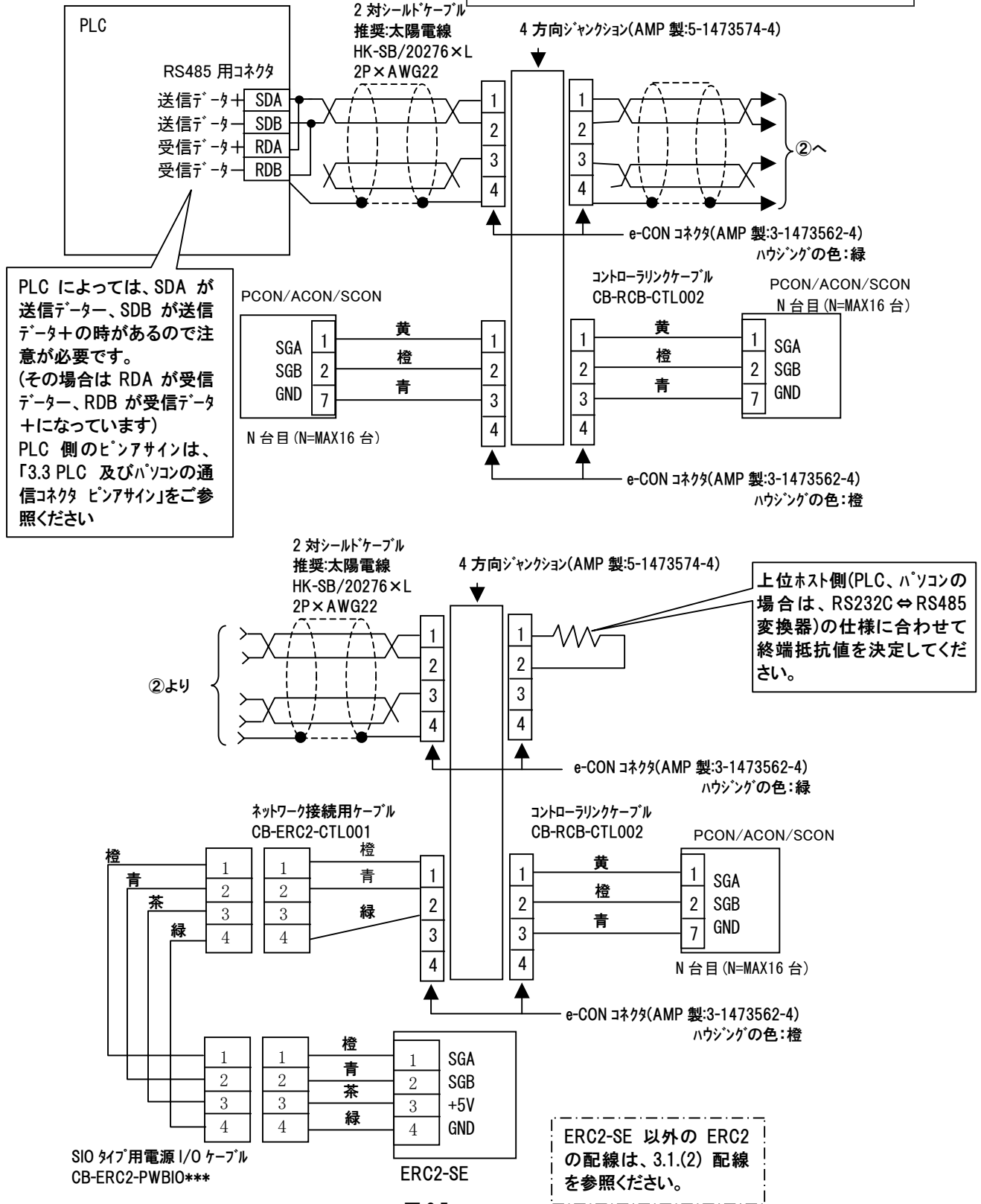


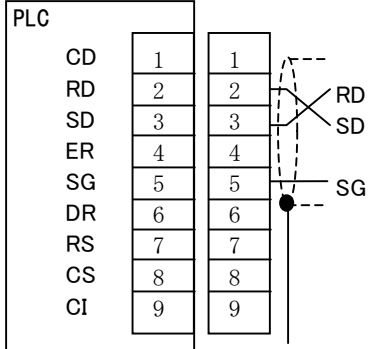
図 3.5

3.3 PLC 及び パソコンの通信コネクタ ピンサイン(参考)

三菱製 PLC:QJ71C24

RS232C の時

D サブ 9 ピンコネクタ(オス:ケーブル側)

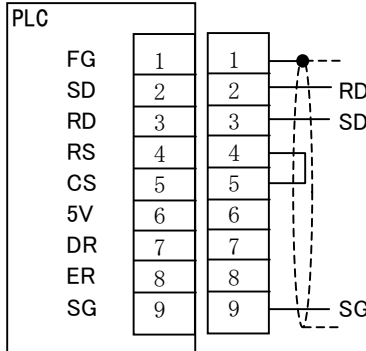


シールドは片側を
コネクタハウジングに接続
または接地

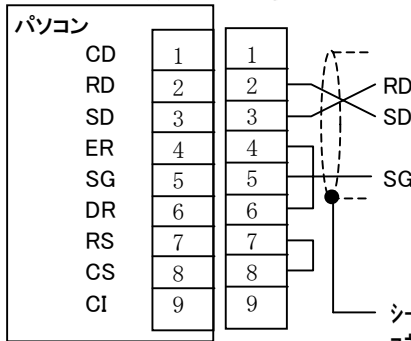
オムロン製 PLC:CJ1W-SCB、SCU

RS232C の時

D サブ 9 ピンコネクタ(オス:ケーブル側)



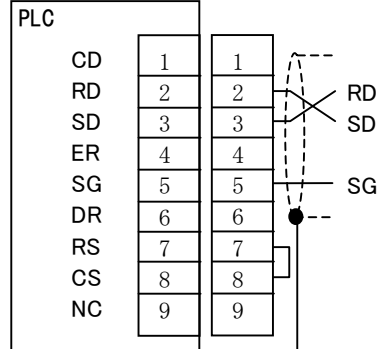
パソコン :RS232C
D サブ 9 ピンコネクタ(メス:ケーブル側)



キーエンス製 PLC:KV-L20R

RS232C の時

D サブ 9 ピンコネクタ(メス:ケーブル側)



シールドは片側を
コネクタハウジングに
接続または接地

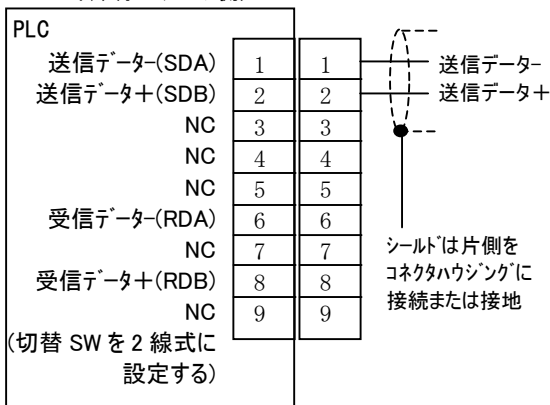
フロー制御を使用する場合は、
RS と CS、DR と ER を接続して
ください。

シールドは
コネクタハウジングに接続

オムロン製 PLC:CJ1W-SCB、SCU

RS485 の時

D サブ 9 ピンコネクタ(オス:ケーブル側)



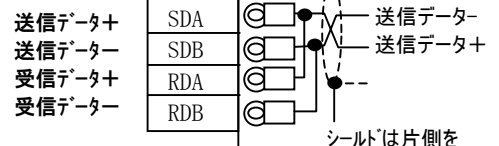
シールドは片側を
コネクタハウジングに
接続または接地

(切替 SW を 2 線式に
設定する)

三菱製 PLC:QJ71C24 RS485 の時

端子台

通信ユニットパネルに信号名が印字されてい
るのでそれに従って配線



シールドは片側を
コネクタハウジングに
接続または接地

キーエンス製 PLC:KV-L20R RS485 の時

端子台



(切替 SW を 485(2)側にする)

シールドは片側を
コネクタハウジングに
接続または接地

※ 詳細は、各メーカーの取扱説明書をご確認ください。

図 3.6

3.4 通信を行うまでの各種設定

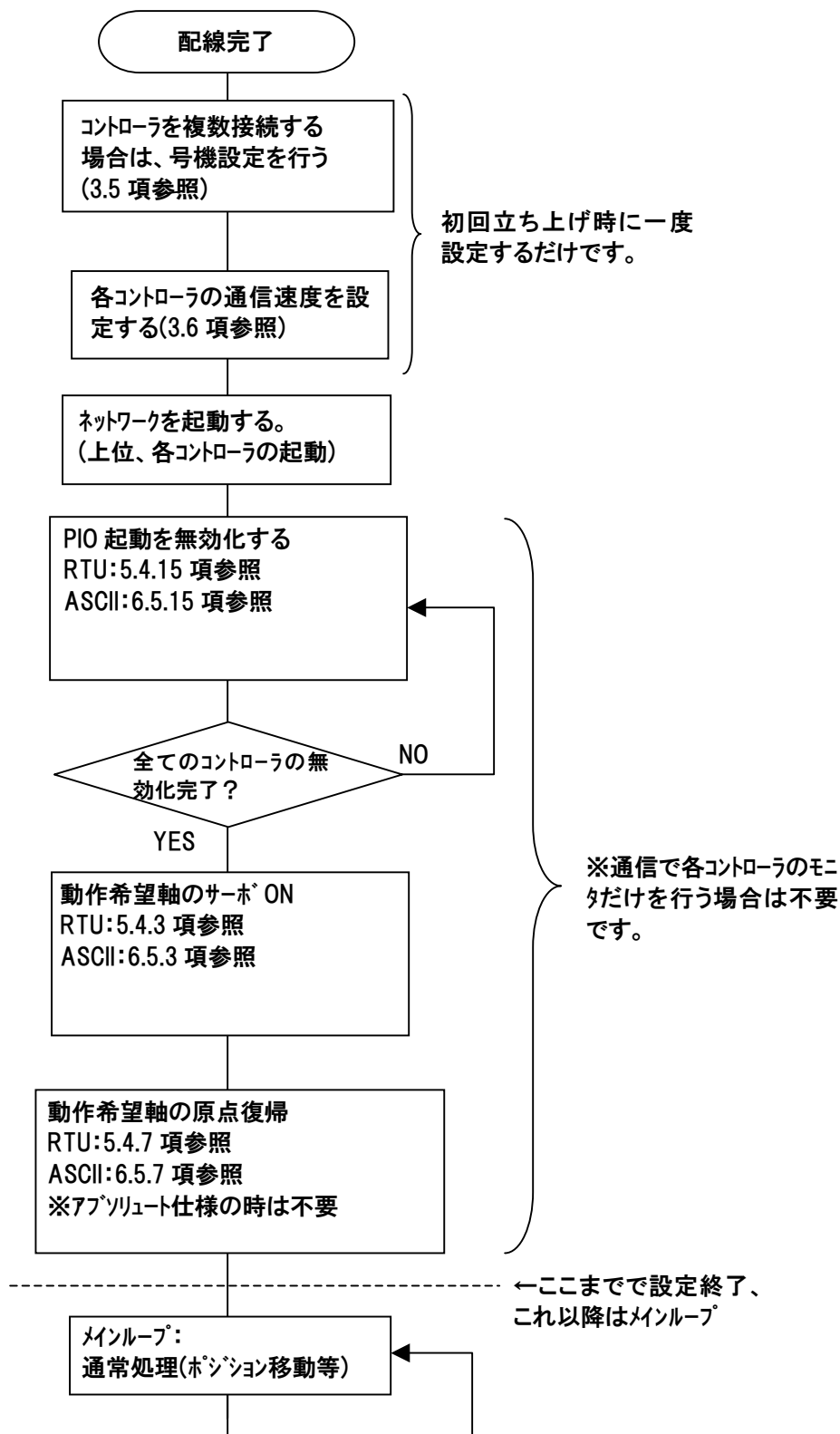


図 3.7

3.5 軸番号の設定

SIOリンク上の各RCコントローラに16軸目をF_Hとした0~F_Hの16進数で軸番号を設定します。

RCコントローラのパネル面に軸番号設定SW(ADRS)がある場合(PCON-C/CG、ACON-C/CG、SCON)は、マイナストライバで軸番号に矢印を合わせてください。(軸番号を重複させないでください。)



図 3.8

軸番号設定用SWが無いRCコントローラは、パソコンでティーチングボックスから設定します。ここでは、パソコンからの設定を紹介します。ティーチングボックスからの設定はティーチングボックスの取扱説明書(CON-T、RCM-E、RCM-T)をご覧ください。

パソコンを軸番号設定したいRCコントローラのSIOコネクタに接続します。

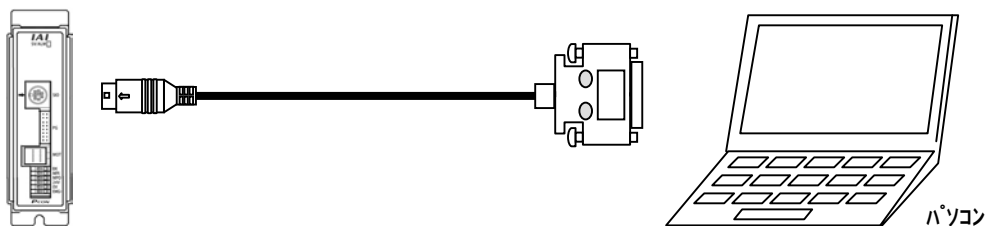


図 3.9

以下の手順で設定を行ってください。

①RC 接続用ソフト起動し、設定(s)をクリック→②コントローラ設定にカーソルを合わせます。

→③軸番号割付(N)にカーソルを合わせクリック→④軸番号テーブルに(※)軸番号(0~15)を入力します(軸番号が重ならないようにして下さい。)

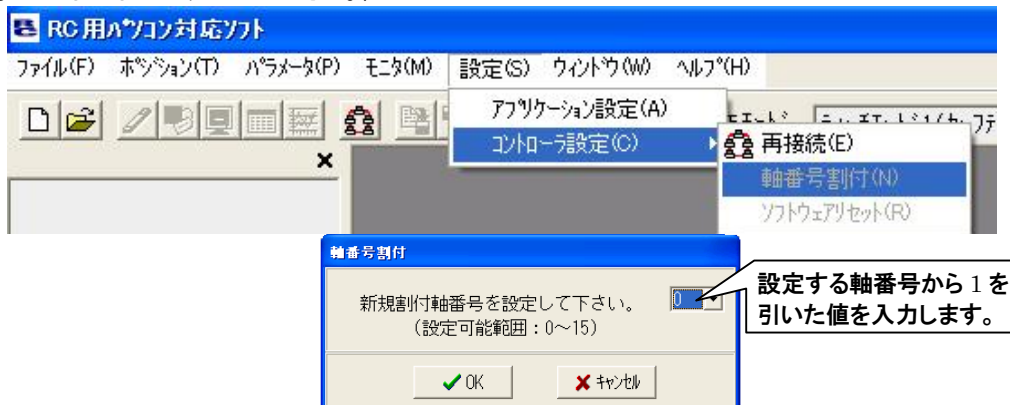


図 3.10

3.6 コントローラの通信速度を設定

通信を行う場合、PLC、各 RC コントローラの通信速度を合わせる必要があります。

通信速度は、3.6.1 項、3.6.2 項の順に設定してください。(PLC は、各 PLC の取扱説明書を参照ください。)

システム構成に応じて配線が異なりますので、ご注意ください。

3.6.1 各システム毎の配線およびハードウェアの設定

(1) 上位(ホスト)コントローラにパソコンを使用する場合

現在の接続のまま設定を行うことができます。モード切替 SW のある RC コントローラ(PCON-C/CG、ACON-C/CG、SCON)は、モード切替 SW を MANU 側で設定を行ってください。

(2) 上位(ホスト)コントローラに PLC を使用して RS232C で接続している場合

PLC の代わりに上位(ホスト)コントローラにパソコンを接続してください(図 3.1 参照)。その際、PLC と SIO 変換器の接続を外し、パソコン対応ソフトに付属のケーブルを使用して SIO 変換器のテスターポート(3.1(3) 項参照)に接続してください。モード切替 SW のある RC コントローラ(PCON-C/CG、ACON-C/CG、SCON)は、モード切替 SW は、MANU にしてください。

(3) 上位(ホスト)コントローラに PLC を使用して RS422 で接続している場合

軸番号の設定と同様に各 RC コントローラに直接パソコンを接続してください。モード切替 SW のある RC コントローラ(PCON-C/CG、ACON-C/CG、SCON)は、モード切替 SW は、MANU にしてください。

3.6.2 通信速度の設定

以下の手順で設定を行ってください。

① RC 接続用ソフト起動し、パラメータ(P)→編集(E)をクリックします

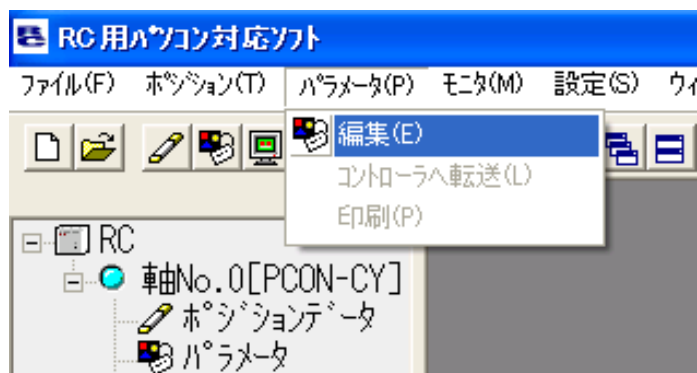


図 3.11

②変更を行うコントローラの軸番号を選択します。

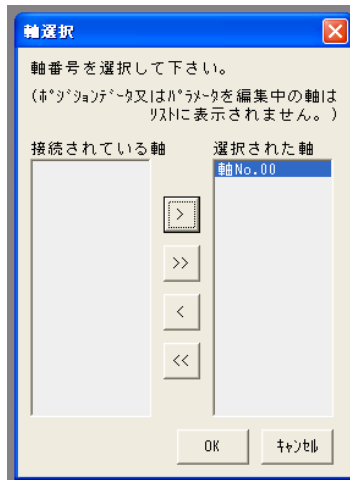


図 3.12

③パラメータNo.16 の SIO 通信速度を設定して下さい。

No	パラメータ名称	設定値
1	ツォン境界1+側[mm]	100.00
2	ツォン境界1-側[mm]	-0.30
3	ツォトリミット+側[mm]	150.30
4	ツォトリミット-側[mm]	-0.30
5	原点復帰方向[0:逆/1:正]	1
6	押付け停止判定時間[msec]	255
7	サーボゲイン番号	8
8	速度初期値[mm/sec]	600
9	加減速度初期値[G]	0.30
10	位置決め幅初期値[mm]	0.10
11	(将来の拡張のための予約)	0
12	位置決め停止時電流制限値[%]	35
13	原点復帰時電流制限値[%]	35
14	(将来の拡張のための予約)	0
15	(将来の拡張のための予約)	1
16	SIO通信速度[bps]	38400
17	従局トランスミッタ活性化最小遅延時間(RTIM)[msec]	5

図 3.13

4 通信

4.1 メッセージ送信タイミング

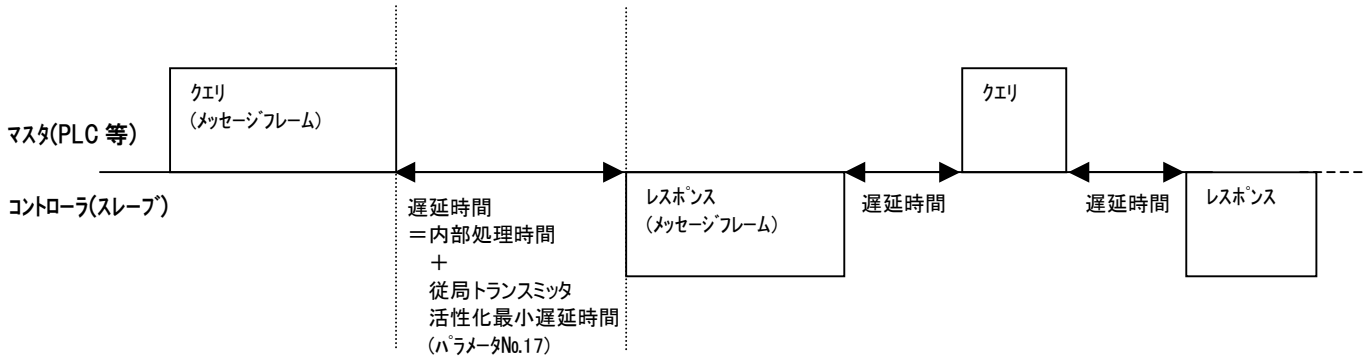


図 4.1

基本的な伝送制御手順は、マスターからのクエリの送信と、これを受信した RC コントローラからのレスポンスの返信を 1 単位とした伝送となります。

受信後、送信までの遅延時間は、運転時に有効になっているパラメータNo.17「従局トランスミッタ活性化最小遅延時間(初期値 5ms)」と内部処理時間(下表参照)の合計になります。

RC コントローラはクエリ・メッセージ受信完了後、この遅延時間以上経過してからレスポンス・メッセージの送信を開始します。マスターは、クエリ・メッセージ送信完了後、この遅延時間以内に自局を受信可能な状態にする必要があります。

RC コントローラはレスポンス・メッセージの送信完了後、直ちに次のクエリ受信に備えます。

内部処理時間(アクセスするカテゴリによって処理時間が異なります)

項目	時間
低速メモリ領域以外の読出し/書込み	最大 1 msec
ポジションデータ(1 ポジション)読出し	最大 4 msec
ポジションデータ(1 ポジション)書込み	最大 15 msec
ポジションデータ(1 ポジション)読出し/書込み	最大 18 msec
ポジションデータ(9 ポジション)読出し	最大 9 msec
ポジションデータ(9 ポジション)書込み	最大 90 msec
ポジションデータ(9 ポジション)読出し/書込み	最大 98 msec

4.2 タイムアウトとリトライ

上位ホストはクエリ送信完了後、コントローラからのレスポンスを待ちます。(ブロードキャストクエリの場合を除く)

コマンド送信後、レスポンス受信完了までの経過時間がタイムアウト値 Tout を超えた場合、上位ホストはコマンド再送による通信回復を行うように設定して下さい。リトライ回数が 3 回を超えた場合は、回復不可能な通信異常としてください。

以下にタイムアウト値 Tout 算出方法を示します。

1. タイムアウト値 Tout

$$Tout = To + \alpha + (10 \times Bprt / Kbr) \text{ [msec]}$$

To : 内部処理時間※ × 安全率 3

α : 従局トランスミッタ活性化最小遅延時間 [msec](パラメータNo.17 初期値 5 ms)

Kbr : 通信速度 [kbps]

Bprt: レスポンス・メッセージのバイト数+8



注意 内部処理時間は、アクセスするカテゴリによって処理時間が異なります。下記の表に処理時間を示します。

項目	時間
低速メモリ領域以外の読出し／書込み	最大 1 msec
ポジションデータ(1 ポジション)読出し	最大 4 msec
ポジションデータ(1 ポジション)書込み	最大 15 msec
ポジションデータ(1 ポジション)読出し／書込み	最大 18 msec
ポジションデータ(9 ポジション)読出し	最大 9 msec
ポジションデータ(9 ポジション)書込み	最大 90 msec
ポジションデータ(9 ポジション)読出し／書込み	最大 98 msec

2. リトライ回数

Nrt = 3(リトライは 必ず設定願います。)

4.3 RCコントローラの内部アドレス 及び データ構造

RCコントローラのレジスタは、データレジスタとステータスレジスタによって構成されています。

4.3.1 データレジスタの構造

データレジスタは次のように配置されています。

0000 _H	(システム用)※
0D00 _H ↕	I/O 制御情報カテゴリ
0D03 _H	(システム用)※
1000 _H ↕	ポジションテーブル情報 《低速メモリ領域》
3FFF _H	(システム用)※
9000 _H ↕	コントローラモニタ情報カテゴリ
9013 _H	(システム用)※
9800 _H	ポジション指令カテゴリ
	(システム用)※
9900 _H ↕	直値指令カテゴリ
9908 _H	
FFFF _H	(システム用)※

※システム用領域は通信に使用できません。

4.3.2 データレジスタ詳細

アドレス[HEX]	エリア名称	内容	記号	掲載頁						
				RTU		ASCII				
0000~0CFF	システム用									
0D00	I/O 制御情報カテゴリ	デバイス制御レジスタ 1	DRG1	109	19	208	19			
0D01		デバイス制御レジスタ 2	DRG2		20		20			
0D03		ポジション番号指定レジスタ	POSR		21		21			
0D04~0FFF	システム用									
1000~3FFF	ポジションテーブル情報(低速メモリ領域)	オフセット[HEX]		123	125	222	224			
		+0000 _H	目標位置					PCMD		
		+0002 _H	位置決め幅					INP		
		+0004 _H	速度指令					VCMD		
		+0006 _H	個別ゾーン境界+側					ZNMP		
		+0008 _H	個別ゾーン境界-側					ZNLP		
		+000A _H	加速度指令					ACMD		
		+000B _H	減速度指令					DCMD	126	225
		+000C _H	押付け時電流制限値					PPOW		
		+000D _H	負荷電流閾値					LPOW		
+000E _H	制御フラグ指定	CTLF								
※詳細アドレスの計算は右の通り →		※ アドレス = 1000 _H + (16 × ポジションNo.) + オフセット								
4000~8FFF	システム用									
9000	コントローラ モニタ情報 カテゴリ	現在位置モニタ	PNOW	41	44	140	143			
9002		現在発生アラームコード照会	ALMC		46		145			
9003		入力ポート照会	DIPM		48		147			
9004		出力ポートモニタ照会	DOPM		52		151			
9005		デバイスステータス 1 照会	DSS1		56		155			
9006		デバイスステータス 2 照会	DSS2		58		157			
9007		拡張デバイスステータス照会	DSSE		60		159			
9008		システムステータス照会	STAT		62		161			
900A		現在速度モニタ	VNOW		64		163			
900C		電流値モニタ	CNOW		66		165			
900E		偏差モニタ	DEVI		68		167			
9010		システムタイム照会	STIM		70		169			
9012		特殊入力ポート照会	SIPM		72		171			
9013		ゾーンステータス照会	ZONS		74		173			
9014		位置決め完了ポジションNo.ステータス照会	POSS		76		175			
9015~97FF	システム用									
9800	ポジション指令カテゴリ	ポジション移動指令レジスタ	POSR							
9801~98FF	システム用									
9900	直値指令 カテゴリ	目標位置指定レジスタ	PCMD	112	114	211	213			
9902		位置決め幅指定レジスタ	INP							
9904		速度指定レジスタ	VCMD							
9906		加減速度指定レジスタ	ACMD					115	214	
9907		押付け時電流制限指定レジスタ	PPOW							
9908		制御フラグ指定レジスタ	CTLF					116	225	
9909~FFFF		システム用								

(1) デバイス制御レジスタ1 内容(アドレス= 0D00_H)(DRG1)

ビット	記号	名称	機能
15	EMG	EMG 動作指定	0:非常停止解除状態 1:非常停止 本ビットを 1 にすると非常停止状態となります。ただし駆動源遮断は行ないませんのでご注意ください。 (コントローラ本体の ALM LED は点灯しません)
14	SFTY	セーフティ速度指令	0:セーフティ速度無効 1:セーフティ速度有効 本ビットを 1 にすると、No.35 パラメータの「セーフティ速度」で指定された速度により、全ての移動指令速度が制限されます。
13	—	使用できません	
12	SON	サーボ ON 指令	0:サーボ OFF 1:サーボ ON 本ビットを 1 にするとサーボ ON 状態となります。 ただし下記条件を満たしている必要があります。 ・デバイスステータスレジスタ 1(5.3.6 または 6.4.6):EMG ステータスビット=0 ・デバイスステータスレジスタ 1(5.3.6 または 6.4.6):重故障ステータスビット=0 ・デバイスステータスレジスタ 2(5.3.7 または 6.4.7):イネーブルステータスビット=1 ・システムステータスレジスタ(5.3.9 または 6.4.9):自動サーボ OFF 中ステータスビット=0
11~9	—	使用できません	
8	ALRS	アラームリセット指令	0:通常 1:アラームリセット実行 本ビットへのエッジ入力(本ビット:0→1)で、アラームがリセットされます。ただしアラーム要因が解消されていないと再びアラームとなります。また一時停止中に本ビットにエッジ入力(本ビット:0→1)されると残移動量のキャンセルが行われます。
7	BKRL	ブレーキ強制解除指令	0:通常 1:ブレーキ強制解除 ブレーキが ON の状態でも本ビットを 1 にすることにより、強制的に解除することができます。
6	—	使用できません	
5	STP	一時停止指令	0:通常 1:一時停止指令 本ビットが 1 の状態では、モータの移動は全て禁止されます。移動中に本ビットが 1 になった場合は減速停止します。再び 0 にセットされると残移動量の移動を行います。 原点復帰中に本ビットが立てられた場合は、押付け反転前は移動指令が保留されますが、押付け反転後では原点復帰を最初からやり直します。
4	HOME	原点復帰指令	0:通常 1:原点復帰指令 本ビットへのエッジ入力(本ビット:0→1)で、原点復帰動作を行います。原点復帰が完了すると HEND ビットが 1 になります。 原点復帰完了は、何度でも入力可能です。
3	CSTR	位置決め動作起動指令	0:通常 1:ポジションスタート指令 本ビットへのエッジ入力(本ビット:0→1)で、ポジション番号指定レジスタ (POSR)内のポジション番号の指定位置に移動します。本ビットが 1 のままですと位置決め幅内に入っても完了ポジションは出力されません。(0 を書き込んで通常状態に戻してください。) 電源投入後、一度も原点復帰動作を行っていない状態(HEND ビットが 0 の状態)では、原点復帰動作を実行した後に目標位置に移動を開始します。 ※目標位置及び速度等の動作パラメータは、全てコントローラ内部のポジションテーブル(不揮発性メモリ)に予め設定しておく必要があります。
2~0	—	使用できません	

(2) デバイス制御レジスタ 2 内容(アドレス= 0D01_H)(DRG2)

ビット	記号	名称	機能
15	—	使用できません	
14	JISL	ジヨグ/インチング切替え	0:ジヨグ 1:インチング 0 の時、ジヨグ動作が選択され、1 の時はインチング動作が選択されます。 また、ジヨグ動作中に本ビットが切替ると減速停止します。 インチング動作中に本ビットが切替ってもインチング動作は継続します。 本ビットでのジヨグ/インチング切替はツールからのジヨグ/インチング動作には反映されません。
13	—	使用できません	
12	—	使用できません	
11	MOD	ティーチモード指令	0:通常運転モード 1:教示モード 本ビットを 1 にすると教示モードになります。
10	TEAC	ポジションデータ取込み指令	0:通常 1:ポジションデータ取込み指令 11ビット目のティーチモード指令が 1(教示モード)の時に、本ビットに 1 を書き込むことで現在位置データの取込みを行います。 取込み場所は、ポジション番号指定レジスタで指定されているポジション番号の中です。取込みポジションが空のポジションの場合、目標位置以外のデータ(位置決め幅 INP,速度 VCMD,加減速度 ACMD,制御フラグ CTLF)はパラメータの初期値が一緒に書込まれます。 本ビットに 1 を書き込みましたら 20 msec 以上そのままの状態を保ってください。
9	JOG+	ジヨグ+指令	0:通常 1:ジヨグ+指令 ・14ビット目の JISL ビットが 0 の時、本ビットを 1 にしている間、反原点方向にジヨグ移動を行います。速度および加減速度はユーザパラメータ No.26 の PIO ジヨグ速度と定格加減速度が使用されます。 ジヨグ移動中に本ビットを 0 にするか、8ビット目のジヨグ-指令を 1 にすると減速停止します。 ・14ビット目の JISL ビットが 1 の時、ジヨグ+指令のエッジを立てる(本ビット:0→1)と、反原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIO ジヨグ速度)、ユーザパラメータ No.48(PIO インチング距離)、定格加減速度が使用されます。
8	JOG-	ジヨグ-指令	0:通常 1:ジヨグ-指令 ・14ビット目の JISL ビットが 0 の時、本ビットを 1 にしている間、原点方向にジヨグ移動を行います。速度および加減速度はユーザパラメータNo.26 の PIO ジヨグ速度と定格加減速度が使用されます。 ジヨグ移動中に本ビットを 0 にするか、9ビット目のジヨグ+指令を 1 にすると減速停止します。 ・14ビット目の JISL ビットが 1 の時、ジヨグ-指令のエッジを立てる(本ビット:0→1)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIO ジヨグ速度)、ユーザパラメータNo.48(PIO インチング距離)、定格加減速度が使用されます。
7	ST7	スタートポジション 7	指定されたポジションNo.位置に移動します。
6	ST6	スタートポジション 6	PIO パターン 4、5(電磁弁モード)が選択されている時に使用できます。
5	ST5	スタートポジション 5	ST0～ST7 のどれかを 1 にする(本ビット:0→1)と移動を行います。
4	ST4	スタートポジション 4	有効スタートポジション以外を選択するとアラーム「085 移動時ポジションNo.異常」が発生します。
3	ST3	スタートポジション 3	
2	ST2	スタートポジション 2	ユーザパラメータNo.27 移動指令種別によりレベル動作とエッジ動作が選択可能です。
1	ST1	スタートポジション 1	
0	ST0	スタートポジション 0	同時に複数のポジションを入力すると若い番号が優先されます。

(3) ポジション番号指定レジスタ内容(アドレス= 0D03_H)(POSR)

ビット	記号	名称	機能
15	—	使用できません	
14	—	使用できません	
13	—	使用できません	
12	—	使用できません	
11	—	使用できません	
10	—	使用できません	
9	—	使用できません	
8	PC256	ポジション指令ビット 256	移動したいポジションNo.をバイナリコードで指定します。 ポジションNo.指定の後、5.5.1または6.6.1デバイス制御レジスタ1 のCSTRを1にするとポジション移動します。 機種や、PIOパターンにより最大ポジション数は異なりますのでご 注意ください。
7	PC128	ポジション指令ビット 128	
6	PC64	ポジション指令ビット 64	
5	PC32	ポジション指令ビット 32	
4	PC16	ポジション指令ビット 16	
3	PC8	ポジション指令ビット 8	
2	PC4	ポジション指令ビット 4	
1	PC2	ポジション指令ビット 2	
0	PC1	ポジション指令ビット 1	

(4) デバイスステータスレジスタ1 内容(アドレス = 9005_H)(DSS1)

ビット	記号	名称	機能
15	EMGS	EMG ステータス	0:非常停止解除状態 1:非常停止状態 非常停止入力及び駆動源遮断等によりコントローラが非常停止状態にあることを示します。
14	SFTY	セーフティ速度有効ステータス	0:セーフティ速度無効 1:セーフティ速度有効 デバイス制御レジスタ1の「セーフティ速度指令ビット」によりコントローラのセーフティ速度状態を示します。
13	PWR	コントローラステータス	0:コントローラ BUSY 1:コントローラ READY コントローラが外部より制御可能かを示します。通常は BUSY になることはありません。
12	SV	サーボ ON ステータス	0:サーボ OFF 1:サーボ ON サーボ ON 状態を示します。サーボ ON 指令を行ってからパラメータのサーボ ON が実行されるまでは本ビットは0のままです。またサーボ ON 指令を行ってもサーボ ON できない場合も本ビットは0のままです。本ビットが0の状態では、RCコントローラは一切の移動動作指令を受け付けません。
11	PSFL	押付け空振り	0:通常 1:押付け空振り 押付け動作指令を行い、押付け幅まで移動した時(押付け空振り)に1となります。 押付け動作以外の動作指令では0のままです。
10	ALMH	重故障ステータス	0:通常 1:重故障アラーム発生中 コールドスタートレベル または動作解除レベルのアラームが発生した場合に1となります。 動作解除レベルのアラーム解除はアラームリセット指令により行えますが、コールドスタートレベルは電源再投入が必要です。
9	ALML	軽故障ステータス	0:通常 1:軽故障アラーム発生中 メッセージレベルのアラームが発生した場合に1となります。
8	ABER	アブソエラーステータス	0:通常 1:アブソエラー発生中 アブソリュート仕様の場合、アブソ関連のエラー発生で1になります。
7	BKRL	ブレーキ強制解除ステータス	0:ブレーキ動作中 1:ブレーキ解除中 ブレーキ動作の状態を示します。通常サーボ ON 中は1となります。サーボ OFF 中であってもデバイス制御レジスタ1の「ブレーキ強制解除指令ビット」を1にすることで1となります。
6	-	使用できません	
5	STP	一時停止中ステータス	0:通常 1:一時停止指令中 本ビットは移動中の一時停止中を示すものではなく、一時停止指令の入力中は1となります。 5.4.15 または 6.5.15PIO/Modbus 切替設定が PIO 有効の時は一時停止の PIO 信号のモニタを行います。(モータ切替 SW がある RC コントローラは、AUTO 側に設定しておきます。) Modbus が有効の時は 5.4.6 または 6.5.6 一時停止指令のモニタを行います。
4	HEND	原点復帰完了ステータス	0:原点復帰未完了 1:原点復帰完了 原点復帰完了で1となります。アブソリュート仕様の場合はアブソリュートが完了していれば起動時から1となります。 本ビットが0の状態での絶対位置指令はアラームとなります。
3	PEND	位置決め完了ステータス	0:位置決め未完了 1:位置決め完了 目標位置まで移動して、インポジション幅に入ると1となります。サーボ ON の時はその場に位置決めを行ったことになるので1となります。 また、押付け動作中 押付け完了で1となります。
2	-	使用できません	
1	-	使用できません	
0	-	使用できません	

(5) デバイスステータスレジスタ 2 内容(アドレス = 9006_H)(DSS2)

ビット	記号	名称	機能
15	-	使用できません	
14	-	使用できません	
13	LOAD	負荷出力判定ステータス	0:通常 1:負荷出力判定 移動指令時に負荷電流閾値及び検定範囲(個別ゾーン境界値:PCON-CF 以外には対応していません)が設定されている場合、検定範囲内でモータ電流が閾値に達したかを示します。 次の位置指令を受けるまでは現在の値を保持します。
12	TRQS	トルクレベルステータス	0:通常 1:トルクレベル達成 押付け動作により移動中、押付けトルクに電流値が達した時に 1 となります。 本ビットはレベルを示しているため電流が変化すれば本ビットの状態も変化します。
11	MODS	ティーチモードステータス	0:通常運転モード 1:教示モード デバイス制御レジスタ 2 の「ティーチモード指令ビット」により教示モードが選択されたときに 1 となります。
10	TEAC	ポジションデータ取り込み指令ステータス	0:通常 1:ポジションデータ取り込み完了 デバイス制御レジスタ 2 の「ポジションデータ取り込み指令ビット」を 1 にすると本ビットは 0 となり、ポジションデータが EEPROM に正常に書き込まれると本ビットは 1 となります。
9	JOG+	ジョグ+ステータス	0:通常 1:「ジョグ+」指令中 デバイス制御レジスタ 2 の「ジョグ+指令ビット」が選択されている間、1 となります。
8	JOG-	ジョグ-ステータス	0:通常 1:「ジョグ-」指令中 デバイス制御レジスタ 2 の「ジョグ-指令ビット」が選択されている間、1 となります。
7	PE7	完了ポジション 7	PIO パターン 4、5(電磁弁モード)時に完了ポジションNo.をハイナリで出力します。 ポジション移動指令により、ポジション移動が行われ、目標位置まで移動して、インポジション幅に入ると 1 となります。 サーボ OFF では 0 となりますが、再びサーボ ON した場合、指令ポジションデータのインポジション幅内であれば本ビットは 1 となります。また、押付け動作で押付け完了した時 あるいは、空振りした場合に 1 となります。
6	PE6	完了ポジション 6	
5	PE5	完了ポジション 5	
4	PE4	完了ポジション 4	
3	PE3	完了ポジション 3	
2	PE2	完了ポジション 2	
1	PE1	完了ポジション 1	
0	PE0	完了ポジション 0	

(6) 拡張デバイスステータスレジスタ内容(アドレス = 9007_H)(DSSE)

ビット	記号	名称	機能
15	EMGP	非常停止ステータス	0:非常停止入力 OFF 1:非常停止入力 ON 非常停止入力ポートの状態を示します。
14	MPUV	モータ電圧低下ステータス	0:通常 1:モータ駆動源遮断中 モータ駆動電源の入力がない場合に本ビットは 1 となります。
13	RMDS	運転モードステータス	0:AUTO モード 1:MANU モード RC コントローラが MANU モードの時に 1 となります。 但し、運転モード SW 非搭載機種(ERC2、PCON-SE、ACON-SE、PCON-CY、ACON-CY)では常に MANU モードとなります。
12	-	使用できません	
11	GHMS	原点復帰中ステータス	0:通常 1:原点復帰中 原点復帰動作中に 1 となります。それ以外では 0 となります。
10	PUSH	押付け動作中	0:通常 1:押付け動作中 押付け動作中の押付け範囲中(位置決め設定範囲)だけ 1 となります。但し、次の条件により 0 となります。 1.押付け空振り 2.一時停止 3.次の移動指令 4.サーボ OFF
9	PSNS	励磁検出ステータス	0:励磁検出未完了 1:励磁検出完了 PCON・ERC2 シリーズコントローラでは起動後最初のサーボ ON 指令の際、励磁検出動作を行います。励磁検出が完了すると 1 となります。 励磁検出動作を失敗すると 0 のまま、検出後もソフトウェアリセットにより再び 0 となります。 ACON シリーズコントローラでは起動後最初のサーボ ON 指令にてホールセンサ動作を行い、完了すると 1 となります。 SCON シリーズコントローラでは常に 0 です。
8	PMSS	PIO/Modbus 切替えステータス	0:PIO 指令有効 1:PIO 指令無効 拡張デバイス制御レジスタ(DRGE)の「PIO/Modbus 切替設定」により切替えします。
7	-	使用できません	
6	-	使用できません	
5	MOVE	移動中信号	0:停止中 1:移動中 移動中(原点復帰、押付け動作中含む)を示します。一時停止中は 0 となります。
4	-	使用できません	
3	-	使用できません	
2	-	使用できません	
1	-	使用できません	
0	-	使用できません	

(7) システムステータスレジスタ内容(アドレス = 9008_H)(STAT)

ビット	記号	名称	機能
31 ~ 18	-	使用できません	
17	ASOF	自動サーボ OFF 中	0:通常 1:自動サーボ OFF 中 RCコントローラのパラメータで「自動サーボ OFF 遅延時間」が設定されていて、位置決め完了後、該当時間が経過し自動サーボ OFF となった場合にこのビットが 1 となります。
16	AEEP	EEP アクセス中	0:通常 1:EEPROM アクセス中 RCコントローラのパラメータ・ポジションテーブル等の読書き中、EEPROM アクセスが開始された時 1 となります。 アクセスが完了するかタイムアウトエラーになった時 0 となります。
15 ~ 5	-	使用できません	
4	RMDS	運転モードステータス	0:AUTO モード 1:MANU モード RCコントローラが MANU モードの時に 1 となります。 但し、運転モード SW 非搭載機種(ERC2、PCON-SE、ACON-SE)では常に MANU モードとなります。
3	HEND	原点復帰完了状態	0:原点復帰未完了 1:原点復帰完了 原点復帰完了で 1 となります。777ソリット仕様の場合は777リセットが完了していれば起動時から 1 となります。 本ビットが 0 の状態での絶対位置指令はアラームとなります。
2	SV	サーボ状態	0:サーボ OFF 1:サーボ ON サーボ ON 状態を示します。サーボ ON 指令を行ってからパラメータのサーボ ON が実行されるまでは本ビットは 0 のままです。 またサーボ ON 指令を行ってもサーボ ON できない場合も本ビットは 0 のままです 本ビットが 0 の状態では、RCコントローラは一切の移動動作指令を受け付けません。
1	SON	サーボ指令状態	0:サーボ OFF 1:サーボ ON サーボ ON/OFF の指令状態を示す。 本ビットは以下の条件を満たしている場合 1 となる。 ・デバイスステータスレジスタ 1 の EMG ステータスビットが 0 (5.3.6 または 6.4.6 ビット 15) ・デバイスステータスレジスタ 1 の重故障ステータスビットが 0 (5.3.6 または 6.4.6 ビット 10) ・デバイスステータスレジスタ 2 のイネーブルステータスビットが 1 (5.3.7 または 6.4.7 ビット 15) ・システムステータスレジスタの自動サーボ OFF 中ステータスビットが 0 (5.3.9 または 6.4.9 ビット 17)
0	MPOW	駆動源 ON	0:駆動源遮断中 1:通常 モータ駆動電源の入力がない場合に本ビットは 0 となります。

(8) 特殊ポートモニタレジスタ内容(アドレス= 9012_H)(SIPM)

ビット	記号	名称	機能
15	-	使用できません	
14	NP	指令パルス NP 信号状態	指令パルス NP 信号の状態を示します。
13	-	使用できません	
12	PP	指令パルス PP 信号状態	指令パルス PP 信号の状態を示します。
11	-	使用できません	
10	-	使用できません	
9	-	使用できません	
8	MDSW	モードスイッチ状態	0:AUTO モード 1:MANU モード RC コントローラが MANU モードの時に 1 となります。 但し、運転モード SW 非搭載機種(ERC2、PCON-SE、 ACON-SE)では常に MANU モードとなります。
7	-	使用できません	
6	-	使用できません	
5	-	使用できません	
4	BLCT	ヘルト切断センサ (SCON のみ)	0:ヘルト切断 1:通常
3	HMCK	原点確認センサモニタ	0:センサ OFF 1:センサ ON 原点確認センサ機能搭載機種では、センサ入力の状態を示します。 非搭載機種では常に 0 です。
2	OT	オーバートラベルセンサ	0:センサ OFF 1:センサ ON エンコーダコネクタ内のオーバートラベルセンサ信号の状態を示します。 非搭載機種では常に 0 です。
1	CREP	クリープセンサ	0:センサ OFF 1:センサ ON エンコーダコネクタ内のクリープセンサ信号の状態を示します。 非搭載機種では常に 0 です。
0	LS	リミットセンサ	0:センサ OFF 1:センサ ON エンコーダコネクタ内の原点センサ信号の状態を示します。 非搭載機種では常に 0 です。

(9) ゾーンステータスレジスタ内容(アドレス= 9013_H)(ZONS)

ビット	記号	名称	機能
15	-	使用できません	
14	LS2	リミットセンサ出力モニタ 2 (PCON-C/CG、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジションNo.2の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。 範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
13	LS1	リミットセンサ出力モニタ 1 (PCON-C/CG、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジションNo.1の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。 範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
12	LS0	リミットセンサ出力モニタ 0 (PCON-C/CG、 ACON-C/CG、 SCON PIO パターン=5 の時)	0:範囲外 1:範囲内 ポジションNo.0の目標位置から位置決め幅分をマイナスした位置が境界値の-側となり、位置決め幅分をプラスした位置が境界値の+側となります。 現在位置が その範囲内にあるときに 1 になります。 範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
11	-	使用できません	
10	-	使用できません	
9	-	使用できません	
8	ZP	ポジションゾーン出力モニタ	0:範囲外 1:範囲内 各ポジション毎に設定されたゾーンの範囲内に現在位置があるときに 1 となります。範囲外では 0 となります。 本ビットは原点復帰完了後に有効となります。 サーボオフ中も有効です。
7	-	使用できません	
6	-	使用できません	
5	-	使用できません	
4	-	使用できません	
3	-	使用できません	
2	-	使用できません	
1	Z2	ゾーン出力モニタ 2	0:範囲外 1:範囲内 パラメータのゾーン境界 2 を設定された範囲内に現在位置があるときに 1 となります。範囲外では 0 となります。 本出力は原点復帰完了後に有効となります。 サーボオフ中も有効です。
0	Z1	ゾーン出力モニタ 1	0:範囲外 1:範囲内 パラメータのゾーン境界 1 を設定された範囲内に現在位置があるときに 1 となります。範囲外では 0 となります。 本出力は原点復帰完了後に有効となります。 サーボオフ中も有効です。

(10) ホジション番号ステータスレジスタ内容(アドレス = 9014_H)(POSS)

ビット	記号	名称	機能
15	-	使用できません	
14	-	使用できません	
13	-	使用できません	
12	-	使用できません	
11	-	使用できません	
10	-	使用できません	
9	-	使用できません	
8	PM256	完了ホジション番号ステータスビット 256	<p>位置決めが完了したホジション番号を示します。 (PIO パターン 4、5 電磁弁モード以外の時に有効)</p> <p>完了ホジションは、ハイナリコードで読み出されます。 目標位置の周囲(正方向も負方向も位置決め幅分の範囲)に現在位置が入ると完了ホジション番号を読み出すことができます。それ以外では、全て 0 が読み出されます。</p> <p>サーボ OFF では全て 0 となりますが、再びサーボ ON した時に目標位置の周囲内に現在位置が入っていれば、完了ホジション番号も再び有効となります。</p> <p>押付け時は、押付け完了 及び、押付け空振りのどちらでも完了ホジションを読み出すことができます。</p>
7	PM128	完了ホジション番号ステータスビット 128	
6	PM64	完了ホジション番号ステータスビット 64	
5	PM32	完了ホジション番号ステータスビット 32	
4	PM16	完了ホジション番号ステータスビット 16	
3	PM8	完了ホジション番号ステータスビット 8	
2	PM4	完了ホジション番号ステータスビット 4	
1	PM2	完了ホジション番号ステータスビット 2	
0	PM1	完了ホジション番号ステータスビット 1	

4.3.3 ステータスレジスタの構造

ステータスレジスタは次のように配置されています。

0000 _H	(システム用)※
0100 _H ? 010F _H	デバイスステータスレジスタ 1 [DSS1]
0110 _H ? 011F _H	デバイスステータスレジスタ 2 [DSS2]
0120 _H ? 012F _H	拡張デバイスステータスレジスタ [DSSE]
0130 _H ? 013F _H	ポジション番号ステータスレジスタ [POSS]
0140 _H ? 014F _H	ゾーンステータスレジスタ [ZONS]
0150 _H ? 015F _H	入力ポートモニタレジスタ [DIPM]
0160 _H ? 016F _H	出力ポートモニタレジスタ [DOPM]
0170 _H ? 017F _H	特殊入力ポートモニタレジスタ [SIPM]
	(システム用)※
0400 _H ? 040F _H	デバイス制御レジスタ 1 [DRG1]
0410 _H ? 041F _H	デバイス制御レジスタ 2 [DRG2]
0420 _H ? 042F _H	拡張デバイス制御レジスタ [DRGE]
0430 _H ? 043F _H	ポジション番号指定レジスタ [POSR]
FFFF _H	(システム用)※

※システム用領域は通信に使用できません。

4.3.4 ステータスレジスタ詳細

アドレス[HEX]	エリア名称	内容	記号	掲載頁			
				RTU		ASCII	
0000~00FF	システム用						
0100	デバイス	EMG ステータス	EMGS	(56)	22	(155)	22
0101	ステータスレジスタ 1	セーフティ速度有効ステータス	SFTY				
0102	(DSS1)	コントローラレディステータス	PWR				
0103		サーボ ON ステータス	SV				
0104		押付け空振り	PSFL				
0105		重故障ステータス	ALMH				
0106		軽故障ステータス	ALML				
0107		アブソエラーステータス	ABER				
0108		ブレーキ強制解除ステータス	BKRL				
0109		使用できません					
010A		一時停止ステータス	STP		22		22
010B		原点復帰ステータス	HEND				
010C		位置決め完了ステータス	PEND				
010D~010F		使用できません					
0110	デバイス	使用できません		(58)		(157)	
0111	ステータス	使用できません					
0112	レジスタ 2	負荷出力判定ステータス	LOAD		23		23
0113	(DSS2)	トルクレベルステータス	TRQS				
0114		ティーチモードステータス	MODS				
0115		ポジションデータ取込み指令ステータス	TEAC				
0116		ジョグ+ステータス	JOG+				
0117		ジョグ-ステータス	JOG-				
0118		完了ポジション 7	PE7				
0119		完了ポジション 6	PE6				
011A		完了ポジション 5	PE5				
011B		完了ポジション 4	PE4				
011C		完了ポジション 3	PE3				
011D		完了ポジション 2	PE2				
011E		完了ポジション 1	PE1				
011F		完了ポジション 0	PE0				
0120	拡張	非常停止ステータス	EMGP	(60)	24	(159)	24
0121	デバイス	モータ電圧低下ステータス	MPUV				
0122	ステータスレジスタ	運転モードステータス	RMDS				
0123	(DSSE)	使用できません					
0124		原点復帰中ステータス	GHMS		24		24
0125		押付け動作中	PUSH				
0126		励磁検出ステータス	PSNS				
0127		PIO/Modbus 切替ステータス	PMSS				
0128		使用できません					
0129		使用できません					
012A		移動中信号	MOVE		24		24
012B~012F		使用できません					

アドレス[HEX]	エリア名称	内容	記号	掲載頁			
				RTU		ASCII	
0130~0136	ポジション番号 ステータスレジスタ (POSS)	使用できません		(76)		(175)	
0137		完了ポジション番号ステータスビット 256	PM256		28		28
0138		完了ポジション番号ステータスビット 128	PM128				
0139		完了ポジション番号ステータスビット 64	PM64				
013A		完了ポジション番号ステータスビット 32	PM32				
013B		完了ポジション番号ステータスビット 16	PM16				
013C		完了ポジション番号ステータスビット 8	PM8				
013D		完了ポジション番号ステータスビット 4	PM4				
013E		完了ポジション番号ステータスビット 2	PM2				
013F		完了ポジション番号ステータスビット 1	PM1				
0140	ゾーン ステータス レジスタ (ZONS)	使用できません		(74)		(173)	
0141		リミットセンサ出力モータ 2	LS2		27		27
0142		リミットセンサ出力モータ 1	LS1				
0143		リミットセンサ出力モータ 0	LS0				
0144~0146		使用できません					
0147		ポジションゾーン出力モータ	ZP		27		27
0148~014D		使用できません					
014E		ゾーン出力モータ 2	Z2		27		27
014F		ゾーン出力モータ 1	Z1				
0150 ~ 015F	入力ポートモ ニタ レジスタ (DIPM)	PIO コネクタピン番号 20A(IN15)~ PIO コネクタピン番号 5A(IN0)		48		147	
0160~ 016F	出力 ポートモニタレ ジスタ (DOPM)	PIO コネクタピン番号 16B(OUT15)~ PIO コネクタピン番号 1B(OUT0)		52		151	
0170	特殊入力 ポートモニタレ ジスタ (SIPM)	使用できません		(72)		(171)	
0171		指令パルス NP 信号状態	NP		26		26
0172		使用できません					
0173		指令パルス PP 信号状態	PP		26		26
0174~0175		使用できません					
0176		使用できません					
0177		モードスイッチ状態	MDSW		26		26
0178		使用できません					
0179~017B		使用できません					
017C		原点確認センサモ ニタ	HMCK		26		26
017D		オーハトラベルセン サ	OT				
017E		クリーブセンサ	CREP				
017F		リミットセンサ	LS				
0180~03FF		システム用					

アドレス[HEX]	エリア名称	内容	記号	掲載頁								
				RTU		ASCII						
0400	デバイス 制御 レジスタ 1 (DRG1)	EMG 動作指定	EMG	(109)	19	(208)	19					
0401		セーフティ速度指令	SFTY									
0402		使用できません										
0403		サーボ ON 指令	SON					19	19			
0404~0406		使用できません										
0407		アラームリセット指令	ALRS					19	19			
0408		ブレーキ強制解除指令	BKRL									
0409		使用できません										
040A		一時停止指令	STP					19	19			
040B		原点復帰指令	HOME									
040C		位置決め動作起動指令	CSTR									
040D~040F		使用できません										
0410		デバイス 制御 レジスタ 2 (DRG2)	使用できません						(109)		(208)	
0411			ジョグ/インテック切替					JISL				
0412~0413	使用できません											
0414	ティーチモード指令		MOD	20	20							
0415	ポジションデータ取込み指令		TEAC									
0416	ジョグ+指令		JOG+									
0417	ジョグ指令		JOG-									
0418	スタートポジション 7		ST7									
0419	スタートポジション 6		ST6									
041A	スタートポジション 5		ST5									
041B	スタートポジション 4		ST4									
041C	スタートポジション 3		ST3									
041D	スタートポジション 2		ST2									
041E	スタートポジション 1		ST1									
041F	スタートポジション 0	ST0										
0420~0426	拡張	使用できません										
0427	デバイス	PIO/Modbus 切換え指定	PMSL					105	204			
0428~042B	制御	使用できません										
042C	レジスタ	減速停止	STOP					107	206			
042D~042F	(DRGE)	使用できません										
0430~0436	ポジション番 号指定 レジスタ (POSR)	使用できません		(109)	21	(208)	21					
0437		ポジション指令ビット 256	PC256									
0438		ポジション指令ビット 128	PC128									
0439		ポジション指令ビット 64	PC64									
043A		ポジション指令ビット 32	PC32									
043B		ポジション指令ビット 16	PC16									
043C		ポジション指令ビット 8	PC8									
043D		ポジション指令ビット 4	PC4									
043E		ポジション指令ビット 2	PC2									
043F	ポジション指令ビット 1	PC1										
0440~FFFF	システム用											

5 Modbus RTU



5.1 メッセージフレーム(クエリ、レスポンス)

スタート	アドレス	ファンクションコード	データ	CRC チェック	エンド
サイレント インターバル	1 byte	1 byte	n byte	2 byte	サイレント インターバル

(1) スタート

3.5 文字(キャラクタ)以上のサイレント・インターバル(無通信時間)です。

(1 文字(キャラクタ) = 10 bit)

例: 9600bps の時 $(10 \times 3.5) \text{bit} \times 1 / 9600 \text{bps} = 3.65 \text{ms}$

※レスポンスタイムアウトエラーが発生してしまう場合、パラメータの「No.45 サイレントインターバル倍率」または、「No.17 従局トランスミッタ活性化最小時間」を弊社ティーチングツールを使用して適宜変更してください。

(2) アドレス

接続されている RC コントローラのアドレス(01_H~10_H)を指定します。

アドレス = 軸番号 + 1

で設定します。

注意！ アドレスは軸番号と同じでないため 設定にご注意ください。

(3) ファンクション

RC コントローラで使用可能なファンクションコード及び機能を示します。

コード (Hex)	名称	機能
01H	Read Coil Status	コイル、DO の読出し
02H	Read Input Status	入力ステータス、DI の読出し
03H	Read Holding Registers	保持レジスタの読出し
04H	Read Input Registers	入力レジスタの読出し
05H	Force Single Coil	コイル、DO への 1 点書込み
06H	Preset Single Register	保持レジスタへの書込み
07H	Read Exception Status	例外ステータス読出し
0FH	Force Multiple Coils	複数コイル、DO への一括書込み
10H	Preset Multiple Registers	複数保持レジスタへの一括書込み
11H	Report Slave ID	スレーブ ID 問い合わせ
17H	Read / Write Registers	レジスタへの読出し、書込み

※本書では□マークのファンクションコードを使用しています。

※ROBONET ゲートウェイでは 3 種のファンクションコード(03_H、06_H、10_H)をサポートしています。
(別冊の ROBONET 取扱説明書をご参照ください)

(4) データ
 ファンクションコードで指示されたデータを付加する場合に用います。ファンクションコードでデータ付加の指示が無い場合は、データ無しも許されます。

(5) CRC チェック
 RTU モードでは、メッセージには CRC 方式に基づいてメッセージ全体の内容がチェックできるようにエラーチェックが自動的(※)に付加されています。また、チェックはメッセージ中の個別の文字(キャラクタ)のパリティチェック方式と関連せず行われます。
 CRC チェックは、16 ビットのバイナリ値で構成されています。CRC 値は、CRC をメッセージに付加する送信側が計算します。受信側は、メッセージ受信中に CRC を再計算して、その計算結果と送られてきた値と比較します。もし、この 2 つの値が一致しなければ、結果はエラーとなります。
 (※)パソコン及び、Modbus に対応していない PLC をホストとして使用する場合、CRC の計算を行う関数を作成する必要があります。
 「10.CRC チェック計算」に C 言語によるプログラムを掲載しています。
 生成多項式: $x^{16} + x^{15} + x^2 + 1$ (CRC-16 方式)

参考) オムロン製 PLC CJ1 シリーズの Modbus RTU 通信をサポートしている FINS コマンドでは、CRC の計算が自動的に行われます。

(6) エント
 3.5 文字(キャラクタ)以上のサイレント・インターバル(無通信時間)です。
 ※レスポンスタイムアウトエラーが発生してしまう場合、パラメータの「No.45 サイレントインターバル倍率」または、「No.17 従局トランスミッタ活性化最小時間」を弊社ティーチングツールを使用して適宜変更してください。

(7) ブロードキャスト
 アドレスを 00_H で指定すると接続されている全ての軸に 同一内容のクエリを送信することができます。この場合 RC コントローラからレスポンスは返信されません。
 また、本機能は使用できるファンクションコード等に制限がありますので、十分注意してご利用ください。使用できるファンクションコードは、「5.2 RTU モードクエリ一覧」でご確認ください。

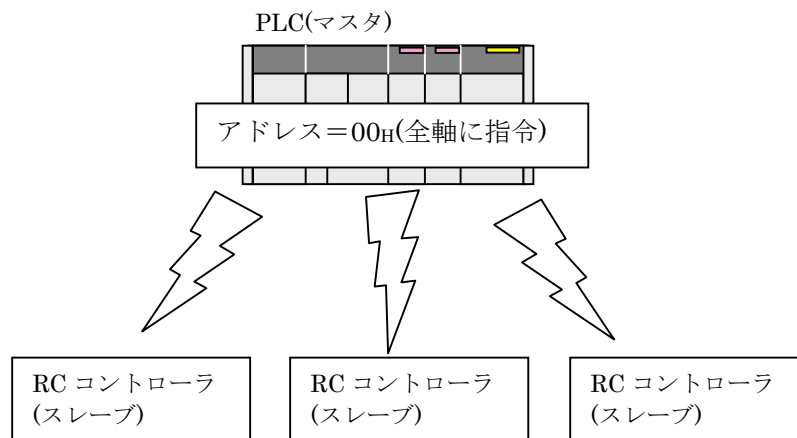


図 5.1

R ROBO C CYLINDER



注意

RC コントローラの送受信バッファサイズは、それぞれ 256 バイトとなっています。ホスト側から送信する伝文は受信バッファを、データをリクエストする場合は送信バッファを、それぞれオーバーしないように計算してください。

5.2 RTUモード クエリー一覧

RTUモードで使用できるクエリー一覧です。

FC	機能	記号	機能概要	PIOとの併用	ブロードキャスト	頁
03	ファンクションコード 03 複数レジスタ読み込み	無し	ファンクション 03 を使用するレジスタを連続的に複数読出すこと可能です。	○		41
03	現在位置モニタ	PNOW	アクチュエータの現在位置を 0.01mm 単位で読出します	○		44
03	現在発生アラームコード 照会	ALMC	現在発生中のアラームコードを読出します。	○		46
03	入力ポート照会	DIPM	PIO 入力ポート ON/OFF 状態を読出します。	○		48
03	出力ポート照会	DOPM	PIO 出力ポート ON/OFF 状態を読出します。	○		52
03	デバイスステータス 1 照会 (運転準備ステータス)	DSS1	次の 12 項目の状態(ステータス)を読出します。 ① 非常停止 ② セーフティー速度有効/無効 ③ コントローラレディ ④ サーボ ON/OFF ⑤ 押付け空振り ⑥ 重故障 ⑦ 軽故障 ⑧ アブソエラー ⑨ ブレーキ ⑩ 一時停止 ⑪ 原点復帰完了 ⑫ 位置決め完了	○		56
03	デバイスステータス 2 照会 (運転情報 1 ステータス)	DSS2	次の 15 項目の状態(ステータス)を読出します。 ① イネーブル ② 負荷出力判定(検定範囲負荷電流閾値) ③ トルクレベル(負荷電流閾値) ④ ティーチモード(通常/ティーチ) ⑤ ポジションデータ取込(通常/完了) ⑥ ジョグ+(通常/指令中) ⑦ ジョグ-(通常/指令中) ⑧ 完了ポジション 7 ⑨ 完了ポジション 6 ⑩ 完了ポジション 5 ⑪ 完了ポジション 4 ⑫ 完了ポジション 3 ⑬ 完了ポジション 2 ⑭ 完了ポジション 1 ⑮ 完了ポジション 0	○		58

FC:ファンクションコード

PIO:パラレル I/O(I/O コネクタの入出力)

PIO との併用欄、ブロードキャスト欄の○印は、PIO との併用、ブロードキャストが有効なクエリーを表しています

FC	機能	記号	機能概要	PIOとの併用	プロードキャスト	頁
03	拡張デバイスステータス照会 (運転情報2ステータス)	DSSE	次の9項目の状態(ステータス)を読出します。 ① 非常停止(非常停止入力ポート) ② モータ電圧低下 ③ 運転モード(AUTO/MANU) ④ 原点復帰中 ⑤ 押付け動作中 ⑥ 励磁検出 ⑦ PIO/Modbus 切替 ⑧ ポジションデータ書込み完了ステータス ⑨ 移動中	○		60
03	システムステータス照会 (コントローラステータス)	STAT	次の7項目の状態(ステータス)を読出します。 ① 自動サーボOFF中 ② EEPROMアクセス中 ③ 運転モード(AUTO/MANU) ④ 原点復帰完了 ⑤ サーボON/OFF ⑥ サーボ指令 ⑦ 駆動源ON(通常/遮断中)	○		62
03	現在速度モニタ	VNOW	アクチュエータの現在速度を 0.01mm/sec 単位で読出します	○		64
03	電流値モニタ	CNOW	アクチュエータモータトルク電流指令値を 1mA 単位で読出します	○		66
03	偏差モニタ	DEVI	1ms 周期毎の偏差量を 1pulse 単位で読出します	○		68
03	システムタイムモニタ	STIM	コントローラ電源投入時からの積算時間を 1msec 単位で読出します。	○		70
03	特殊入力ポート照会 (センサ入力ステータス)	SIPM	次の9項目の状態(ステータス)を読出します。 ① 指令パルス NP ② 指令パルス PP ③ ポートスイッチ ④ モードスイッチ ⑤ イネーブルスイッチ ⑥ 原点確認センサ ⑦ オーバートラベルセンサ ⑧ クリープセンサ ⑨ リミットセンサ	○		72

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
03	ゾーンステータス照会	ZONS	次の 6 項目の状態(ステータス)を読出します。 ① LS2(PIO パターン電磁弁モード[3 点タイプ]) ② LS1(PIO パターン電磁弁モード[3 点タイプ]) ③ LS0(PIO パターン電磁弁モード[3 点タイプ]) ④ ポジションゾーン ⑤ ゾーン 2 ⑥ ゾーン 1	○		74
03	位置決め完了ポジションNo.照会	POSS	次の 9 項目の状態(ステータス)を読出します。 ① 完了ポジション番号ビット 256 ② 完了ポジション番号ビット 128 ③ 完了ポジション番号ビット 64 ④ 完了ポジション番号ビット 32 ⑤ 完了ポジション番号ビット 16 ⑥ 完了ポジション番号ビット 8 ⑦ 完了ポジション番号ビット 4 ⑧ 完了ポジション番号ビット 2 ⑨ 完了ポジション番号ビット 1	○		76
05	セーフティ速度モード切替	SFTY	有効/無効モード切替を指令します。		○	79
05	サーボ ON/OFF	SON	サーボ ON/OFF を指令します。		○	81
05	アラームリセット	ALRS	アラームリセット/残移動量キャンセルを指令します。		○	83
05	ブレーキ解除	BKRL	ブレーキ強制解除を指令します。		○	85
05	一時停止	STP	一時停止を指令します。		○	87
05	原点復帰	HOME	原点復帰動作を指令します。		○	89
05	位置決め動作起動	CSTR	ポジションNo.指定移動時スタート信号 ON/OFF (本書では使用しません。)		○	91
05	ジヨグ/インチング切替	JISL	ジヨグ/インチングモード切替を行います。		○	93
05	ティーチモード指令	MOD	通常/教示モード切替を行います。		○	95
05	ポジションデータ取込	TEAC	教示モード時、現在位置取込指令を行います		○	97
05	ジヨグ+	JOG+	反原点方向にジヨグ/インチング動作を指令します。		○	99
05	ジヨグ-	JOG-	原点方向にジヨグ/インチング動作を指令します。		○	101

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
05	ポジションNo.0~7 指定	ST0~ST7	電磁弁モード時に有効なポジションNo.指定 この指令だけでアクチュエータが動作可能です。※		○	103
05	PIO/Modbus 切替設定	PMSL	PIO 外部指令信号の有効/無効切替指令		○	105
05	減速停止	STOP	移動中のアクチュエータを減速停止させる事が出来ます。		○	107
06	ポジションNo.指定移動	POSR	この指令だけでポジションNo.指定移動で通常移動、 相対移動、押付け動作が可能です。※		○	109
10	直値移動指令	無し	目標位置、位置決め幅、速度、加減速度、押付け、 制御設定を 1 つの伝文で送信し、動作させる事が 可能です。通常移動、相対移動、押付け動作が 可能です。		○	112
10	ポジションデータテーブル 書込	無し	指定された軸、ポジションNo.のデータを全て変更する ことが可能です。		○	123
不定	例外レスポンス	無し	伝文内容が不正データであった場合のレスポンスで す。			230

※あらかじめポジションテーブルへの位置決めデータの設定が必要です。

- ① ティーチングツールを使用する
- ② ポジションデータテーブル書込み機能を使用する。

5.3 データ、ステータス読み取り(コード 03 使用ケリ)

5.3.1 レジスタの連続複数読み取り

(1) 機能

スレーブのレジスタの内容を読み取ります。

プロトキャストではサポートされていません。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、RTUモード使用時256バイト中4バイト(スレーブアドレス+ファンクションコード+エラーチェック)を除く252バイト分最大126レジスタ分(1レジスタ2バイト使用)のデータ照会が可能です。次の一覧表のデータ(合計21レジスタ)を一度の送受信で照会することも可能です。

アドレス(H)	記号	名称	符号	レジスタサイズ*	バイト
9000,9001	PNOW	現在位置モニタ	○	2	4
9002	ALMC	現在発生アラームコード照会		1	2
9003	DIPM	入力ポート照会		1	2
9004	DOPM	出力ポートモニタ照会		1	2
9005	DSS1	デバイスステータス 1 照会		1	2
9006	DSS2	デバイスステータス 2 照会		1	2
9007	DSSE	拡張デバイスステータス照会		1	2
9008,9009	STAT	システムステータス照会		2	4
900A,900B	VNOW	現在速度モニタ	○	2	4
900C,900D	CNOW	電流値モニタ	○	2	4
900E,900F	DEVI	偏差モニタ	○	2	4
9010,9011	STIM	システムタイマ照会		2	4
9012	SIPM	特殊入力ポート照会		1	2
9013	ZONS	ゾーンステータス照会		1	2
9014	POSS	位置決め完了ポジションNo.ステータス照会		1	2



(3) クエリフォーマット

クエリメッセージでは、読み取りを開始するレジスタのアドレスとレジスタのバイト数を指定します。

1レジスタ(1アドレス)=2バイト=16ビットデータ

フィールド名称	RTUモード 8ビットデータ	データ数 (バイト数)	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	任意	1	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	03	1	レジスタ読出しコード
開始アドレス(H)	任意	2	5.3.1(2)開始アドレス 一覧参照
レジスタの数(H)	任意	2	開始アドレス一覧参照
エラーチェック(H)	CRC(16ビット)	2	
エンド	なし		サイレントインターバル
合計バイト数		8	

(4) レスポンスフォーマット

フィールド名称	RTUモード 8ビットデータ	データ数 (バイト数)	備考
スタート			サイレントインターバル
スレーブアドレス(H)	任意	1	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	03	1	レジスタ読出しコード
データバイト数(H)		1	クエリ指定レジスタの バイト数の合計
データ1(H)		クエリ指定レジスタの バイト数	
データ2(H)		クエリ指定レジスタの バイト数	
データ3(H)		クエリ指定レジスタの バイト数	
データ4(H)		クエリ指定レジスタの バイト数	
:		:	
:		:	
エラーチェック(H)	CRC(16ビット)	2	サイレントインターバル
エンド			
合計バイト数		最大256	

5.3.2 現在位置の読み取り《PNOW》(0.01mm 単位)

(1) 機能

現在位置を 0.01mm 単位で読み取ります。符号は有効です。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9000	現在位置モニタ
レジスタの数(H)	2	0002	アドレス9000 _H ~9001 _H 呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ1(H)	2	現在値による	現在値データ(Hex)(上位)
データ2(H)	2	現在値による	現在値データ(Hex)(下位)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの現在位置(アドレス9000_H~9001_H)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 00 00 02 E9 0B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9000
レジスタの数(H)	0002
エラーチェック(H)	E90B (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 0B FE 7C 83

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 00
データ2(H)	0B FE
エラーチェック(H)	7C83 (CRC計算による)
エンド	サイレントインターバル

現在位置は『00000BFE_H』→10進数に変換→3070(×0.01mm)→

現在位置は 30.7mm

例 2) 現在位置が『FFFFFFF5_H』と読み取れた時(負の位置)→

FFFFFFF_H - FFFFFFF5_H + 1(必ず1を加算)→

10進数に変換→11(×0.01mm)→ 現在位置は -0.11mm

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.3 現在発生アラームコードの読み取り《ALMC》

(1) 機能

コントローラの正常状態、またはアラーム状態を示すコードを読み取ります。

正常状態では 00_H が格納されています。

アラームコードの詳細内容は各コントローラの取扱説明書をご参照ください。

(2) クエリフォーマット

フィールド名	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9002	現在発生アラームコード
レジスタの数(H)	2	0001	アドレス9002 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータ列は、1レジスタ当たり16ビットのデータです。

フィールド名	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ1(H)	2	アラームコード	アラームコード(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのアラームコード(アドレス9002_H)を読み取りした使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 03 90 02 00 01 08 CA

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9002
レジスタの数(H)	0001
エラーチェック(H)	08CA (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下のようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 E8 B8 0A

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	00 E8
エラーチェック(H)	B80A (CRC計算による)
エンド	サイレントインターバル

このとき発生している最重要アラームは 『0E8』_H

A,B 相断線検出アラームです。

コードの詳細内容は各コントローラの取扱説明書を参照してください。

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.4 I/Oポート入力信号状態読み取り《DIPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート入力値を読み取ります。

データはRCコントローラが入力として認識しているポートの状態です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9003	入力ポートモニタレジスタ
レジスタの数(H)	2	0001	アドレス9003 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、17アドレス当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ1(H)	2	ポート入力値	ポート入力値(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		



(5) ポート割付け (詳細は、各 RC コントローラの取扱説明書をご参照ください)

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)			
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3
IN0	PC1	ST0	PC1	PC1
IN1	PC2	ST1	PC2	PC2
IN2	PC4	ST2	PC4	PC4
IN3	HOME	0	PC8	PC8
IN4	CSTR	RES	CSTR	CSTR
IN5	*STP	*STP	*STP	*STP
IN6	0	0	0	0
IN7	0	0	0	0
IN8	0	0	0	0
IN9	0	0	0	0
IN10	0	0	0	0
IN11	0	0	0	0
IN12	0	0	0	0
IN13	0	0	0	0
IN14	0	0	0	0
IN15	0	0	0	0

ポート	PCON-C/CG						PCON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

ポート	SCON						(ハルス列モード)
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	RES
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	TL
IN4	PC16	PC16	PC16	PC16	ST4	0	CSTR
IN5	PC32	PC32	PC32	PC32	ST5	0	DCLR
IN6	0	MODE	PC64	PC64	ST6	0	BKRL
IN7	0	JISL	PC128	PC128	0	0	RMOD
IN8	0	JOG+	0	PC256	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0
IN15	SON	SON	SON	SON	SON	SON	0

5.3.5 I/Oポート出力信号状態の読み取り《DOPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート出力値を読み取ります。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9004	出力ポートモニタレジスタ
レジスタの数(H)	2	0001	アドレス9004 _H のみの呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ1(H)	2	DO出力値	ポート出力値(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		



(5) ポート割付け (詳細は、各 RC コントローラの取扱説明書をご参照ください)

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)			
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3
OUT0	PEND	PE0	PEND	PEND
OUT1	HEND	PE1	HEND	HEND
OUT2	ZONE	PE2	ZONE	ZONE
OUT3	*ALM	*ALM	*ALM	*ALM
OUT4	0	0	0	0
OUT5	0	0	0	0
OUT6	0	0	0	0
OUT7	0	0	0	0
OUT8	0	0	0	0
OUT9	0	0	0	0
OUT10	0	0	0	0
OUT11	0	0	0	0
OUT12	0	0	0	0
OUT13	0	0	0	0
OUT14	0	0	0	0
OUT15	0	0	0	0

ポート	PCON-C/CG						PCON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/ TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	0	0	0	0	0	0	0	0

ポート	ACON-C/CG						ACON-PL/PO	
	PIO ハターン 0	PIO ハターン 1	PIO ハターン 2	PIO ハターン 3	PIO ハターン 4	PIO ハターン 5	PIO ハターン 0	PIO ハターン 1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/ TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	TRQS	0	TRQS	TRQS	TRQS	0	0	0

ポート	SCON						(ハルス列モード)
	PIO ハターン 0	PIO ハターン 1	PIO ハターン 2	PIO ハターン 3	PIO ハターン 4	PIO ハターン 5	PIO ハターン 0
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PWR
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	SV
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	INP
OUT3	PM8	PM8	PM8	PM8	PE3	0	HEND
OUT4	PM16	PM16	PM16	PM16	PE4	0	TLR
OUT5	PM32	PM32	PM32	PM32	PE5	0	*ALM
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	*EMGS
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	RMDS
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	ALM1
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	ALM4
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	ALM8
OUT12	SV	SV	SV	SV	SV	SV	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0
OUT15	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	0

5.3.6 コントローラ状態信号の読み取り1《DSS1》

(1) 機能

コントローラ内のステータスを読み取ります。

「4.3.2 (4) デバイスステータスレジスタ1 内容」をご参照してください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9005	デバイスステータス レジスタ1
レジスタの数(H)	2	0001	アドレス9005 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1アドレス当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	ステータス1	ステータス1(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9005_H)を読み取りした使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 03 90 05 00 01 B9 0B

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9005
レジスタの数(H)	0001
エラーチェック(H)	B90B (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 70 98 9C 2E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	70 98
エラーチェック(H)	9C2E (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.7 コントローラ状態信号の読み取り2《DSS2》

(1) 機能

コントローラ内のステータスを読み取ります。

「4.3.2 (5) デバイスステータスレジスタ2 内容」をご参照ください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9006	デバイスステータス レジスタ2
レジスタの数(H)	2	0001	アドレス9006 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	コントローラ内のステータス
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	ステータス2	ステータス2(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9006_H)を読み取りした使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 03 90 06 00 01 49 0B

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9006
レジスタの数(H)	0001
エラーチェック(H)	490B (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 80 00 D9 84

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	80 00
エラーチェック(H)	D984 (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.8 コントローラ状態信号の読み取り3《DSSE》

(1) 機能

コントローラ内のステータス(拡張デバイス)を読み取ります。

「4.3.2(6) 拡張デバイスステータスレジスタ内容」をご参照ください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9007	拡張デバイスステータス レジスタ
レジスタの数(H)	2	0001	アドレス9007 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	拡張ステータス	拡張ステータス(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの拡張デバイスステータス(アドレス9007_H)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 07 00 01 18 CB

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9007
レジスタの数(H)	0001
エラーチェック(H)	18CB (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 33 C7 ED 26

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	33 C7
エラーチェック(H)	ED26 (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.9 コントローラ状態信号の読み取り4《STAT》

(1) 機能

コントローラの内部動作状態を読み取ります。

「4.3.2 (7) システムステータスレジスタ内容」をご参照ください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9008	システムステータス レジスタ
レジスタの数(H)	4	0002	アドレス9008 _H ~9009 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	コントローラ内のステータス
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ(H)	4	システムステータス	システムステータス(Hex)
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラのシステムステータス(アドレス9008_H~)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 08 00 02 68 C9

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9008
レジスタの数(H)	0002
エラーチェック(H)	68C9 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 0C 00 17 7A 3E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 0C 00 17
エラーチェック(H)	7A3E (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.10 現在速度の読み取り《VNOW》

(1) 機能

モータ実速度のモニタデータを読み取ります。移動方向により±に振れます。

単位は0.01mm/secです。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	900A	現在速度モニタ
レジスタの数(H)	4	0002	アドレス900A _H ~900B _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ(H)	4	現在速度	現在速度(Hex) 単位は0.01mm/sec。
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの現在速度モニタ(アドレス900A_H~)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 0A 00 02 C9 09

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	900A
レジスタの数(H)	0002
エラーチェック(H)	C909 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 03 E4 FA 88

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 00 03 E4
エラーチェック(H)	FA88 (CRC計算による)
エンド	サイレントインターバル

現在速度は『000003E4』→10進数に変換→996(×0.01mm/sec)→
 現在速度モニタは 9.96mm/sec

例 2)現在速度が『FFFFFF35』と読み取れた時(上の例反対方向に動作)→
 FFFFFFFF_H - FFFFFFF35_H + 1(必ず1を加算)→
 10進数に変換→203(×0.01mm/sec)→
 現在速度モニタは 2.03mm/sec

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.11 電流値の読み取り《CNOW》

(1) 機能

モータ電流(トルク電流指令値)のモニタデータを読み取ります。

単位は mA です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	900C	電流値モニタ
レジスタの数(H)	4	0002	アドレス900C _H ~900D _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ(H)	4	モータ電流の モニタ	モータ電流のモニタ(Hex) 単位はmA
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの電流値モニタ(アドレス900C_H~)を読み取りした使用例を示します
クエリ(前後にサイレントインターバルが入ります)

01 03 90 0C 00 02 29 08

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	900C
レジスタの数(H)	0002
エラーチェック(H)	2908 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 01 C8 FA 35

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 00 01 C8
エラーチェック(H)	FA35 (CRC計算による)
エンド	サイレントインターバル

モータ電流値は『000001C8』→10進数に変換→456(mA)

電流モニタ値は 456mA

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.12 偏差の読み取り《DEVI》

(1) 機能

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量を読み取ります。

単位はパルスです。

モータ機械角 1 回転あたりのパルス数は使用エンコーダにより異なります。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	900E	偏差モニタ
レジスタの数(H)	4	0002	アドレス900E _H ~900F _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ(H)	4	偏差モニタ	偏差モニタ(Hex) 単位はpulse
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラの偏差モニタ(アドレス900E_H~)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 0E 00 02 88 C8

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	900E
レジスタの数(H)	0002
エラーチェック(H)	88C8 (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 00 00 0B BB F4

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 00 00 0B
エラーチェック(H)	BBF4 (CRC計算による)
エンド	サイレントインターバル

偏差モニタは『0000000B』→10進数に変換→11(pulse)

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量は 11pulse

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.13 電源投入後の積算時間の読み取り《STIM》

(1) 機能

コントローラ電源投入時からの積算時間を読み取ります。

単位は msec です。

ソフトウェアリセットではタイマ値はクリアされません。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9010	システムタイマ
レジスタの数(H)	4	0002	アドレス9010 _H ~9011 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	04	2レジスタ呼出=4バイト
データ(H)	4	システムタイマ	システムタイマ(Hex) 単位はmsec
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	9		

(4) 使用例

軸No.0コントローラのシステムタイマ値(アドレス9010_H~)を読み取りした使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 03 90 10 00 02 E8 CE

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9010
レジスタの数(H)	0002
エラーチェック(H)	E8CE (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 04 00 02 7A 72 F8 B6

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	04(4バイト= 2レジスタ)
データ1(H)	00 02 7A 72
エラーチェック(H)	F8B6 (CRC計算による)
エンド	サイレントインターバル

システムタイマは『00027A72』→10進数に変換→162418(msec)

コントローラ電源投入時からの積算時間は 162.418 秒

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.14 特殊入力ポートの入力信号状態の読み取り《SIPM》

(1) 機能

通常の入力ポート以外の入力ポートの状態を読み取ります。

特殊入力ポートの内容は、「4.3.2(8) 特殊入力ポートモニタレジスタ内容」をご参照ください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9012	特殊入力ポートモニタ
レジスタの数(H)	2	0001	アドレス9012 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	特殊ポートモニタ	4.3.2(8) 一覧表参照
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの特異入力ポート(アドレス9012_H)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 12 00 01 09 0F

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9012
レジスタの数(H)	0001
エラーチェック(H)	090F (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 43 00 89 74

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	43 00
エラーチェック(H)	8974 (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.15 ゾーン出力信号の状態読み取り《ZONS》

(1) 機能

ゾーン出力の状態を読み取ります。

「4.3.2 (9) ゾーンステータスレジスタ内容」をご参照ください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9013	ゾーンステータス照会
レジスタの数(H)	2	0001	アドレス9013 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド			サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	ゾーンステータス	4.3.2(9) 一覧表参照
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラのゾーン出力の状態(アドレス9013_H)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 13 00 01 58 CF

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9013
レジスタの数(H)	0001
エラーチェック(H)	58CF (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 00 B8 44

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	00 00
エラーチェック(H)	B844 (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.3.16 位置決め完了ポジションNo.の読み取り《POSS》

(1) 機能

完了ポジション番号を読み取ります。

「4.3.2 (10)」ポジション番号ステータスレジスタ内容をご参照してください。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
開始アドレス(H)	2	9014	ポジション番号ステータス
レジスタの数(H)	2	0001	アドレス9014 _H の呼出
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	1	03	レジスタの読み出し
データバイト数(H)	1	02	1レジスタ呼出=2バイト
データ(H)	2	ポジション番号 ステータス	4.3.2(10) 一覧表参照
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	7		

(4) 使用例

軸No.0コントローラの位置決め完了ポジション(アドレス9014_H)を読み取りした使用例を示します
 クエリ(前後にサイレントインターバルが入ります)

01 03 90 14 00 01 E9 0E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
開始アドレス(H)	9014
レジスタの数(H)	0001
エラーチェック(H)	E90E (CRC計算による)
エンド	サイレントインターバル

レスポンスは以下ようになります。

レスポンス(前後にサイレントインターバルが入ります)

01 03 02 00 00 B8 44

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	03
データバイト数(H)	02(2バイト= 1レジスタ)
データ1(H)	00 00
エラーチェック(H)	B844 (CRC計算による)
エンド	サイレントインターバル

※レスポンス例のデータは一例であり、各種状態により異なります。

5.4 動作指令 及び、データ書き換え(コード05 使用ケリ)

5.4.1 コイルへの書込み

(1) 機能

スレーブのDO(Discrete Output)の状態をON/OFFのいずれかに変更(書込み)します。

ブロードキャストの場合には、全スレーブの同じアドレスのコイルを書換えます。

(2) 開始アドレス一覧

アドレス(H)	記号	機能名称
0401	SFTY	セーフティ速度指令
0403	SON	サーボ ON 指令
0407	ALRS	アラームリセット指令
0408	BKRL	ブレーキ強制解除指令
040A	STP	一時停止指令
040B	HOME	原点復帰指令
040C	CSTR	位置決め動作起動指令
0411	JISL	ジョグ/インテグ切替え
0414	MOD	ティーチモード指令
0415	TEAC	ポジションデータ取込み指令
0416	JOG+	ジョグ+指令
0417	JOG-	ジョグ-指令
0418	ST7	スタートポジション 7(電磁弁モード)
0419	ST6	スタートポジション 6(電磁弁モード)
041A	ST5	スタートポジション 5(電磁弁モード)
041B	ST4	スタートポジション 4(電磁弁モード)
041C	ST3	スタートポジション 3(電磁弁モード)
041D	ST2	スタートポジション 2(電磁弁モード)
041E	ST1	スタートポジション 1(電磁弁モード)
041F	ST0	スタートポジション 0(電磁弁モード)
0427	PMSL	PIO/Modbus 切替え指定
042C	STOP	減速停止

5.4.2 セーフティ速度有効/無効切替《SFTY》

(1) 機能

ユーザパラメータNo.35の「セーフティ速度」で指定された速度の有効/無効切替を行います。
MANUモード時に有効にしますと、全ての移動指令速度が制限されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0401	セーフティ速度指令
変更データ(H)	2	任意	セーフティ速度有効: FF00 _H セーフティ速度無効: 0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのセーフティ速度を有効にする使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 01 FF 00 DC CA

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0401
変更データ(H)	FF00
エラーチェック(H)	DCCA (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.3 サーボ ON/OFF 《SON》

(1) 機能

サーボON/OFFの制御を行います。

変更データ部をサーボON状態にするとメーカーパラメータの「サーボオン遅延時間」経過後にサーボON状態へと遷移します。ただし下記条件を満たしている必要があります。

- ・デバイスステータスレジスタ1のEMGステータスビットが0
- ・デバイスステータスレジスタ1の重故障ステータスビットが0
- ・デバイスステータスレジスタ2のインーブルステータスビットが1
- ・システムステータスレジスタの自動サーボOFF中ステータスビットが0

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0403	サーボON/OFF指令
変更データ(H)	2	任意	サーボ ON: FF00 _H サーボOFF: 0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	17		

※上位との通信前に、ティーチングボックス、パソコンソフトを接続し、サーボOFF動作をした後に、非接続にした場合、上位との通信でサーボON/OFFが出来なくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをサーボONにする使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 03 FF 00 7D 0A

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0403
変更データ(H)	FF00
エラーチェック(H)	7D0A (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.4 アラームリセット《ALRS》

(1) 機能

アラームリセットのエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、アラームリセットを行います。

ただしアラーム要因が解消されていないと再びアラームとなります。

また一時停止中にアラームリセットのエッジを立てると残移動量のキャンセルが行われます。

アラームリセットを行ったら 必ず変更データを0000_Hにして書き込みを行い、通常の状態に戻してください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0407	アラームリセット指令
変更データ(H)	2	任意	アラームリセット実行:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのアラームリセットを行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 07 FF 00 3C CB ...アラームリセット実行

2回目 01 05 04 07 00 00 7D 3B ...通常状態に戻す

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0407
変更データ(H)	1回目:FF00 2回目:0000 (アラームリセット終了後に0000 _H を書き込んで通常状態に戻してください)
エラーチェック(H)	1回目:3CCB (CRC計算による) 2回目:7D3B (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.5 ブレーキ強制解除《BKRL》

(1) 機能

ブレーキの制御はサーボON/OFFと連動して行われますが、ブレーキがONの状態でも強制的に解除することができます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0408	ブレーキ強制解除指令
変更データ(H)	2	任意	ブレーキ強制解除:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※上位との通信前に、ティーチングボックス、パソコンソフトを接続し、サーボOFF動作をした後に、非接続にした場合、上位との通信でサーボON/OFFが出来なくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのブレーキを強制解除する使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 08 FF 00 0C C8

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0408
変更データ(H)	FF00
エラーチェック(H)	0CC8 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.6 一時停止《STP》

(1) 機能

移動中に一時停止指令を行うと減速停止を行い、再び通常状態にセットされると残移動量の移動を再開します。

一時停止指令の状態では、モータの移動は全て禁止されます。

一時停止指令中にアラムリセット指令ビットが立てられた場合は残移動量がキャンセルされます。

一時停止指令の入力が原点復帰動作中で、押付け反転前ならば移動指令が保留され、押付け反転後では原点復帰を最初からやり直します。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	040A	一時停止指令
変更データ(H)	2	任意	一時停止指令:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを一時停止する使用例を示します

クエリ(前後にサイレントインターバルが入ります)

01 05 04 0A FF 00 AD 08

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	040A
変更データ(H)	FF00
エラーチェック(H)	AD08 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.7 原点復帰《HOME》

(1) 機能

原点復帰指令のフラグを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、原点復帰動作を行います。原点復帰が完了するとHENDビットが1になります。

原点復帰指令は、原点復帰が完了していても入力可能です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	040B	原点復帰指令
変更データ(H)	2	任意	原点復帰実行:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※原点復帰は、サーボ ON した状態で行ってください。

上位との通信前に、ティーチングボックス、パソコンソフトを接続し、サーボ OFF 動作をした後に、非接続にした場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、RCコントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボ ON の状態にしてください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラの原点復帰を行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 0B 00 00 BD 38 ...通常状態に設定

2回目 01 05 04 0B FF 00 FC C8 ...原点復帰実行

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	040B
変更データ(H)	1回目:0000 2回目:FF00 (エッジを立てる為に2回データを送信してください。)
エラーチェック(H)	1回目:3CCB (CRC計算による) 2回目:7D3B (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.8 位置決め動作起動指令《CSTR》

(1) 機能

位置決め動作起動指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、ポジション番号指定レジスタ(POSR)内のポジション番号の指定位置に移動します。ポジションスタート指令状態のまま(FF00_Hを書き込んだまま)ですと位置決め幅内に入っても完了ポジションは出力されません。(0000_Hを書き込んで通常状態に戻してください。)

電源投入後、一度も原点復帰動作を行っていない状態(HENDビットが0の状態)では、原点復帰動作を実行した後に目標位置に移動を開始します。

※目標位置 及び速度等の動作パラメータは、全てコントローラ内部のポジションテーブル(不揮発性メモリ)に予め設定しておく必要があります。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	040C	位置決め動作起動指令
変更データ(H)	2	任意	ポジションスタート指令:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。



(4) 使用例

軸No.0コントローラのポジション番号指定レジスタ(POSR)内のポジション番号の指定位置に移動を行う使用例を示します

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 0C FF 00 4D 09 ...指定位置に移動

2回目 01 05 04 0C 00 00 0C F9 ...通常状態に戻す

フィールド ^① 名称	RTUモード ^② 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	040C
変更データ(H)	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック(H)	1回目:4D09 (CRC計算による) 2回目:0CF9 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.9 ジョグ/インチング切替《JISL》

(1) 機能

ジョグとインチングの切替を行います。

ジョグ動作中に本ビットが切替わると減速停止します。

インチング動作中に本ビットが切替わってもインチング動作は継続されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0411	ジョグ/インチング切替
変更データ(H)	2	任意	インチング動作状態:FF00 _H ジョグ動作状態:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをインチング動作に切替えます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 11 FF 00 AD 08

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0411
変更データ(H)	FF00
エラーチェック(H)	AD08 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.10 ティーチモード指令《MOD》

(1) 機能

通常運転モードと教示モードを切替えます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0414	通常モード⇔教示モード切替
変更データ(H)	2	任意	教示モード:FF00 _H 通常運転モード:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを教示モードに切替えます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 14 FF 00 CD 0E

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0414
変更データ(H)	FF00
エラーチェック(H)	CD0E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.11 ポジションデータ取込み指令《TEAC》

(1) 機能

5.4.10 ティーチモード指令が FF00_H(教示モード)の時に、本指令(FF00_Hを書き込み)で現在位置データの取込みを行います。

取込み場所は、ポジション番号指定レジスタで指定されているポジション番号の中です。取込みポジションが空のポジションの場合、目標位置以外のデータ(位置決め幅 INP,速度 VCMD,加減速度 ACMD,制御フラグ CTLF)はパラメータの初期値と一緒に書込まれます。

本指令(FF00_Hを書き込み)を行って 20msec 以上そのままの状態を保ってください。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0415	ポジションデータ取込み指令
変更データ(H)	2	任意	ポジションデータ取込み指令:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。



(4) 使用例

軸No.0コントローラが教示モード時に現在位置を取込みます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 15 FF 00 9C CE

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0415
変更データ(H)	FF00
エラーチェック(H)	9CCE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.12 ジョグ+指令 《JOG+》

(1) 機能

ジョグ またはインテグ動作を行います。

・5.4.9 ジョグ/インテグ切替指令が 0000_H(ジョグ設定)の時、ジョグ+指令(変更データ FF00_H)を送信すると、反原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26のPIOジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ+指令(変更データ 0000_H)を送信するか、5.4.13 ジョグ-指令(変更データ FF00_H)を送信すると減速停止します。

・5.4.9 ジョグ/インテグ切替指令が FF00_H(インテグ設定)の時、ジョグ+指令のエイジを立てる(変更データが 0000_Hの状態に FF00_Hを書き込む)と、反原点方向にインテグ移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインテグ距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,DOへの1点書き込み
開始アドレス(H)	2	0416	ジョグ+指令
変更データ(H)	2	任意	ジョグ+指令:FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 16 FF 00 6C CE

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0416
変更データ(H)	FF00
エラーチェック(H)	6CCE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 16 FF 00 6C CE ...インチング移動

2回目 01 05 04 16 00 00 2D 3E ...通常状態に戻す

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0416
変更データ(H)	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック(H)	1回目:6CCE (CRC計算による) 2回目:2D3E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.13 ジョグ指令《JOG-》

(1) 機能

ジョグ またはインチング動作を行います。

・5.4.9 ジョグ/インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ指令(変更データ FF00_H)を送信すると、原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 のPIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ指令(変更データ 0000_H)を送信するか、5.4.12 ジョグ+指令(変更データ FF00_H)を送信すると減速停止します。

・5.4.9 ジョグ/インチング切替指令が FF00_H(インチング設定)の時、ジョグ指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIO ジョグ速度)、ユーザパラメータNo.48(PIO インチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	0417	ジョグ指令
変更データ(H)	2	任意	ジョグ指令: FF00 _H 通常:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 17 FF 00 3D 0E

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0417
変更データ(H)	FF00
エラーチェック(H)	3D0E (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 17 FF 00 3D 0E ...インチング移動

2回目 01 05 04 17 00 00 7C FE ...通常状態に戻す

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0417
変更データ(H)	1回目:FF00 2回目:0000 (通常状態に戻してください。)
エラーチェック(H)	1回目:3D0E (CRC計算による) 2回目:7CFE (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.14 スタートポジション0～7《ST0～ST7》移動指令（PIOパターン4, 5 限定）

(1) 機能

指定されたポジションNo.位置に移動します。

スタートポジション0～7 移動指令は、PIOパターン4, 5(電磁弁モード)が選択されている時に使用できます。

移動指令は、5.4.14 (5)開始アドレス内のST0～ST7のどれかを有効にする(0000_Hの状態
でFF00_Hを書き込む)ことを行います。

有効スタートポジション以外を選択するとアラーム「085 移動時ポジションNo.異常」が発生します。

ユーザーパラメータNo.27 移動指令種別によりレベル動作とエッジ動作が選択可能です。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ～10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	任意	5.4.14(5)開始アドレス参照
変更データ(H)	2	任意	※1 動作指令 ON:FF00 _H 動作指令OFF:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※1 ユーザパラメータNo.27 移動指令種別を『レベル動作』設定した場合

FF00_H→0000_H書き込みで減速停止します。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをスタートポジション2へ移動します。

スタートポジション設定例

No	位置 [mm]	速度 [mm/s]	加速度 [G]	減速度 [G]
0	0.00	533.00	0.30	0.30
1	25.00	533.00	0.30	0.30
2	50.00	533.00	0.30	0.30

図5.2

クエリ(前後にサイレントインターバルが入ります)

1回目 01 05 04 1D 00 00 5C FC ...エッジを立てる為、0000_H書込み

2回目 01 05 04 1D FF 00 1D 0C ...移動指令

フィールド ^① 名称	RTUモード ^② 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	041D
変更データ(H)	1回目:0000 2回目:FF00
エラーチェック(H)	1回目:5CFC (CRC計算による) 2回目:1D0C (CRC計算による)
エンド ^③	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

(5) 開始アドレス

アドレス	記号	名称	機能
0418	ST7	スタートポジション 7	ポジション 7 へ移動します
0419	ST6	スタートポジション 6	ポジション 6 へ移動します
041A	ST5	スタートポジション 5	ポジション 5 へ移動します
041B	ST4	スタートポジション 4	ポジション 4 へ移動します
041C	ST3	スタートポジション 3	ポジション 3 へ移動します
041D	ST2	スタートポジション 2	ポジション 2 へ移動します
041E	ST1	スタートポジション 1	ポジション 1 へ移動します
041F	ST0	スタートポジション 0	ポジション 0 へ移動します

5. 4. 15 PIO/Modbus 切替設定 《PMSL》

(1) 機能

PIO 外部指令信号の有効/無効の切替を行うことができます。

(2) ケリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,DOへの1点書き込み
開始アドレス(H)	2	0427	PIO/Modbus切替設定
変更データ(H)	2	任意	※1 Modbus指令有効:FF00 _H Modbus指令無効:0000 _H
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

※1 ・Modbus 指令有効(ON)(PIO 指令無効): FF00_H

PIO 信号による運転はできません。

・Modbus 指令無効(OFF)(PIO 指令有効): 0000_H

外部からの PIO 信号による運転が可能です。

補足 Modbus 指令を有効に変更した場合、変更時の PIO 状態が保持されています。

Modbus 指令を無効に切替えた場合、現在の PIO 状態により運転状態が変化します。但し、その時にエッジ検出で動作をする信号の状態が変化していてもエッジを検出したことにしないようにしています。

(3) 注意事項

■運転モードスイッチ搭載機種では AUTO モードに変更されたら『PIO 指令有効』に

MANU モードに変更されたら『PIO 指令無効』になります。

■PIO 未搭載機種ではデフォルト設定が『PIO 指令無効』となります。

■弊社ツール接続時(ティーチングペンダント、パソコン対応ソフト)は、ツール内モードには『ティーチモード 1, 2』、『モニターモード 1, 2』が存在します。この場合

『モニターモード 1, 2』 → 『PIO 指令有効』

『ティーチモード 1, 2』 → 『PIO 指令無効』 となります。



(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをModbus指令有効にします。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 27 FF 00 3D 01

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0427
変更データ(H)	FF00
エラーチェック(H)	3D01 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.4.16 減速停止《STOP》

(1) 機能

減速停止指令のエッジを立てる(FF00_Hを書込む)と、減速停止します。

(2) クエリフォーマット

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	05	コイル,D0への1点書き込み
開始アドレス(H)	2	042C	減速停止設定
変更データ(H)	2	任意	減速停止指令(ON):FF00 _H ※コントローラが自動的に0000 _H に リセットします。
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	8		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。



(4) 使用例

軸No.0コントローラを減速停止させます。

クエリ(前後にサイレントインターバルが入ります)

01 05 04 2C FF 00 4C C3

フィールド名称	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	05
開始アドレス(H)	0427
変更データ(H)	FF00
エラーチェック(H)	4CC3 (CRC計算による)
エンド	サイレントインターバル

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.5 制御情報の直接書き込み(コード 06 使用ケリ)

5.5.1 レジスタへの書き込み

(1) 機能

スレーブのレジスタの内容を変更(書き込み)します。

ブロードキャストの場合には、全スレーブの同じアドレスのレジスタの内容が変更されます。

「4.3.2 (1) デバイス制御レジスタ 1 内容」をご参照ください。

または「4.3.2 (2) デバイス制御レジスタ 2 内容」をご参照ください。

または「4.3.2 (3) ポジション番号指定レジスタ内容」をご参照ください。

(2) 開始アドレス一覧

アドレス	記号	名称	バイト
0D00	DRG1	デバイス制御レジスタ 1	2
0D01	DRG2	デバイス制御レジスタ 2	2
0D03	POSR	ポジション番号指定レジスタ	2

上記は制御指令のレジスタです。本レジスタのヒットは、「PIO/Modbus 切替えステータス (PMSS)(5.3.8参照)」がModbus指令無効(PIO指令有効)の時、PIOパターンにより入力ポートに割り当てられます。本レジスタはModbus指令有効(PIO指令無効)の時、書換えが可能です。

(3) クエリフォーマット

クエリメッセージでは、変更するレジスタのアドレスとデータを指定します。

変更したいデータは、クエリの変更データエリアで16ビットのデータとして指定します。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	06	レジスタへの書込み
開始アドレス(H)	2	任意	5.5.1(2)開始アドレス一覧参照
変更データ(H)	2	任意	4.3.2(1)~4.3.2(3) 変更データ一覧参照
エラーチェック(H)	2	計算結果に基づく	
エンド	なし		サイレントインターバル
合計バイト数	10		

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをサーボON→原点復帰させます。

クエリ(前後にサイレントインターバルが入ります)

1回目 01 06 0D 00 10 00 86 A6 ...サーボON

2回目 01 06 0D 00 10 10 87 6A ...原点復帰

フィールド名	RTUモード 8ビットデータ
スタート	サイレントインターバル
スレーブアドレス(H)	01
ファンクションコード(H)	06
開始アドレス(H)	0D00
変更データ(H)	1回目:1000 2回目:1010 (サーボONのビットはサーボOFF時以外は1にしたままにする)
エラーチェック(H)	1回目:86A6 (CRC計算による) 2回目:876A (CRC計算による)
エンド	サイレントインターバル

- ※ ①サーボOFFの状態から 変更データを1010_Hとして送信しても原点復帰は行われません。(各RCコントローラの起動時のタイミングチャート参照ください。)
- ②前の状態を維持したい場合は、変更がなくても前の状態を送信してください。
使用例のようにサーボONのビットは 原点復帰時も1のままにしておきます。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

5.6 位置決めデータの直接書き込み(コード 10 使用ケリ)

5.6.1 直値移動指令

(1) 機能

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。アドレス 9900_H~9908_H のレジスタ群を書換えるとアクチュエータに対し直値移動指令が行えます。(一伝文で送信が可能です。)

制御フラグ指定レジスタ(アドレス:9908_H)以外のレジスタは電源投入後、1度送信すればその後は有効となりますので、目標位置、位置決め幅、速度、加減速度、押し付け電流制限値、制御指定の変更の必要がない場合、その後は単独変更による実移動指令可能なレジスタ(開始アドレス一覧参照)の書き込みだけで直値移動指令が可能となります。

(2) 開始アドレス一覧

目標位置座標、位置決め幅、速度、加減速度、押し付け電流制限値制御指定フラグ等を数値指定して移動を行うためのレジスタ群です。

開始アドレス一覧のデータ(合計 6 レジスタ)は、一度の送信で変更することが可能です。

アドレス (H)	記号	名称	符号	単独変更による 実移動指令 可能	レジスタ サイズ*	バイト サイズ*	単位
9900	PCMD	目標位置指定レジスタ	○	○	2	4	0.01mm
9902	INP	位置決め幅指定レジスタ		×	2	4	0.01mm
9904	VCMD	速度指定レジスタ		○	2	4	0.01mm/sec
9906	ACMD	加減速度指定レジスタ		○	1	2	0.01G
9907	PPOW	押し付け時電流制限 指定レジスタ		○	1	2	%
9908	CTLF	制御フラグ指定レジスタ		×	1	2	—

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート		なし	サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	10	直値指令
開始アドレス(H)	2	任意	5.6.1(2)開始アドレス一覧参照
レジスタの数(H)	2	任意	5.6.1(2)開始アドレス一覧参照
バイト数(H)	1	上記レジスタ数による	上記指定レジスタ数の2倍の数値を 入力
変更データ1(H)	2		5.6.1(2)開始アドレス一覧参照
変更データ2(H)	2		5.6.1(2)開始アドレス一覧参照
変更データ3(H)	2		5.6.1(2)開始アドレス一覧参照
:	:		:
エラーチェック(H)	2	CRC(16ビット)	
エンド		なし	サイレントインターバル
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート		なし	サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	10	直値指令
開始アドレス(H)	2	任意	5.6.1(2)開始アドレス一覧参照
レジスタの数(H)	2	任意	5.6.1(2)開始アドレス一覧参照
エラーチェック(H)	2	CRC(16ビット)	
エンド		なし	サイレントインターバル
合計バイト数	最大256	最大256	

(5) レジスタ詳細説明

■目標位置指定レジスタ(PCMD)

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999 です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると移動開始指令時にアラームが発生します。目標位置座標指定レジスタ(記号:PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

■位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって 2 種類の意味を持ちます。1 つ目の意味は、通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2 つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、0～999999 です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

本レジスタのみを変更しただけでは移動開始は行われません。

■速度指定レジスタ(VCMD)

移動速度を指定します。単位は 0.01mm/sec で設定範囲は 0～999999 です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタの下位ワードが書換えられると移動開始します。つまり移動中の速度可変を行う場合、本レジスタを書換えることにより実現できます。

■加減速度指定レジスタ(ACMD)

加速度・減速度を指定します。単位は 0.01G で設定範囲は 0~300 です。ただし、パラメータの最大加速度および最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタが書換えられると移動開始します。つまり移動中の加減速度可変を行う場合、本レジスタを書換えることにより実現できます。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。これらの値は、 FF_H を最大電流とする 256 段階で設定します。押付け電流はアクチュエータの制約により設定可能範囲を 20%~70%※($33_H \sim B3_H$)としてください。範囲はアクチュエータにより異なりますので詳しくは弊社カタログまたは取扱説明書参照してください。

本レジスタが書換えられると移動開始します。つまり押付け動作中の電流制限値可変を行う場合、本レジスタを書換えることにより実現できます。

押付け電流値設定例

※20%設定例

$256 \times 0.2 = 51.2 \rightarrow 33_H$ (16 進数変換)

■制御フラグ指定レジスタ(CTLF)

このパラメータは以下の動作設定をするビットパターンです。

本レジスタは、移動開始後、パラメータ初期値にて上書きされます。

ビット 0=0:固定

ビット 1=0:通常動作

1:押付け動作

ビット 2=0:アプローチ動作完了後の押付け動作の方向を正転とします。

1:アプローチ動作完了後の押付け動作の方向を逆転とします。

このビットにより、PCMDからの最終停止位置の方向を算出しますので方向を間違えると下記図 5.3 のように(2×INP)の幅だけずれた動作になりますので、注意してください。

また、ビット 1 の設定値が 0 の場合はこのビットの設定値は無効です。

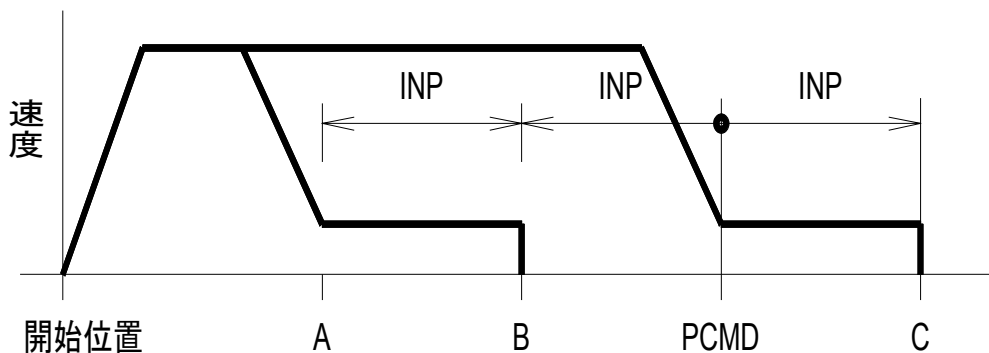


図 5.3

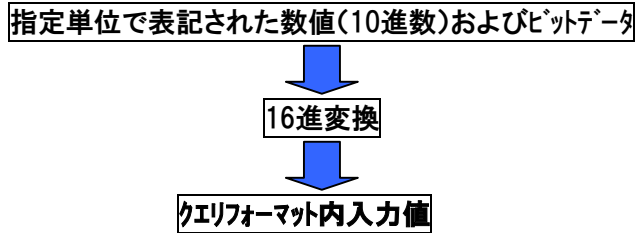
ビット 3=0:通常動作

1:インクリメンタル動作

このビットを 1 に設定することにより、相対位置移動を行う事が可能となります。この動作は、通常動作時と押付け動作時(CTLF のビット 1)で挙動が異なります。通常動作時は目標位置(PCMD)に対しての移動量を生成するのに対し、押付け動作時(ビット 1=1 の時)には現在位置に対しての移動量を生成します。

なお、相対座標の計算は mm 単位加算後パルス変換を行うので、パルス変換後の加算方法の場合に発生する『相対移動を繰り返した場合、リード設定による割り切れないパルスが累積誤差として位置ずれを起こす現象』は発生しません。

(6) 入力値(変更データ)について



(7) レジスタ入力フロー例

a) 通常動作 1 フロー(目標位置のデータ変更)

条件: コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値で動作条件はOK。目標位置だけを変更しアクチュエータを動作させたい。

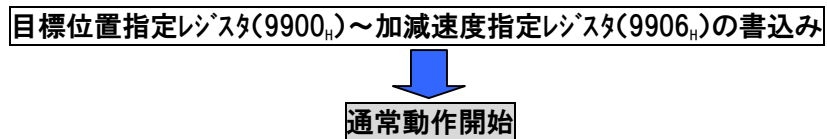
補足: コントローラユーザパラメータについて

- ・速度初期値(パラメータNo.8)→弊社カタログ内表記アクチュエータ依存最高速度
- ・加減速度初期値(パラメータNo.9)→弊社カタログ内表記アクチュエータ依存
定格加速度
- ・位置決め幅初期値(パラメータNo.10)→デフォルト値 0.1mm



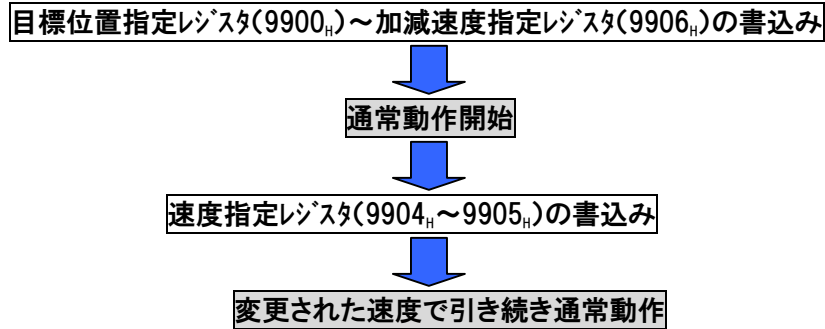
b) 通常動作 2 フロー(位置決めデータの変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させたい



c) 通常動作 3 フロー(移動中の速度変更)

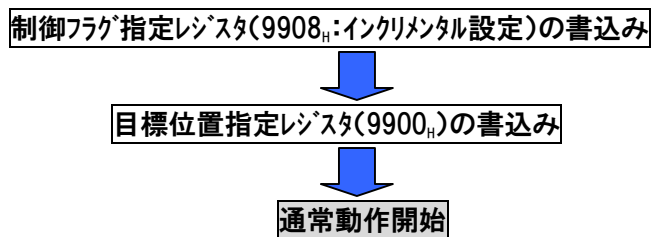
条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい



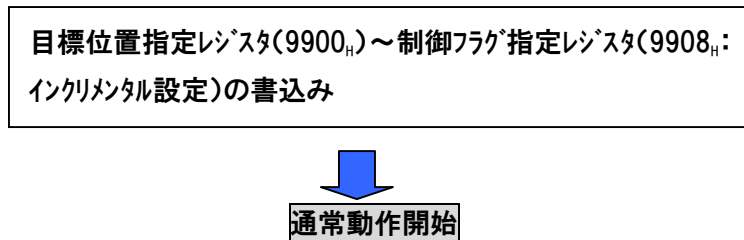
d) 相対位置決め動作 1(インクリメンタル移動)フロー(ピッチ幅の変更)

条件: コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値/で動作条件はOK。ピッチ幅だけを変更しアクチュエータを動作させたい。

手法 1



手法 2



補足: アドレス 9900_H と 9908_H だけを一度のデータ送信で変更することは出来ません。アドレスは連番となっているため、9900_H と 9908_H だけを変更したい場合は 2 度の伝文送信で実施し(手法 1)、1 度の伝文送信で済ませたい場合は、9900_H ~ 9908_H までを全て書き込み実行を行う(手法 2)必要があります。手法 2 を用いる場合は速度/加減速度/位置決め幅に対しても書き込みを実行する事になりますのでユーザパラメータNo.8,9,10 の値は無効となります。

e) 相対位置決め動作 2 (インクリメンタル移動) フロー (インクリメンタル移動中の速度変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを相対位置決め動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み



相対位置決め動作開始



速度指定レジスタ(9904_H)～制御フラグ指定レジスタ(9908_H:インクリメンタル設定)の書込み



変更された速度で引き続き相対位置決め動作

補足: 制御フラグ指定レジスタ(9908_H:インクリメンタル設定)は変更を行ったあと、一度アクチュエータが動作開始されずとリセットされます。(設定値 0_H:通常移動)このため、速度だけを変更する場合でも制御フラグ指定レジスタ(9908_H)をインクリメンタル設定にし、再度送信する必要があります。

f) 押付け動作フロー(押付け動作中の押し付け力の変更)

条件: 押付け動作をさせたい。但し、押し付け中には任意のタイミングで押し付け力を変化させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H: 押付け設定)の書込み



押付け動作開始



押付け電流制限指定レジスタ(9907_H)～制御フラグ指定レジスタ(9908_H: 押付け設定)の書込み



変更された押し付け力で引き続き押付け動作

g) 注意事項(移動中の位置決め幅の変更)

移動中の位置決め幅の変更はできません。

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングで位置決め幅を変更をしたい
(変更できません。書込みを行った場合、このデータは次の位置決めに反映されることとなります。)

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み



通常動作開始



位置決め幅指定レジスタ(9902_H～9903_H)の書込み



変更前の位置決め幅設定で引き続き通常動作

補足: 位置決め幅指定レジスタは単独書込みによる実移動指令は無効です。
このため位置決め幅指定レジスタ(9902_H～9903_H)の書込みによるデータは次回移動指令を実施した際に有効となります。

(8) 使用例

①軸No.0コントローラを通常移動させる使用例を示します

目標位置 (mm)	位置決め幅 (mm)	速度 (mm/sec)	加減速度 (G)	押付け (%)	移動制御
50	0.1	100	0.3	0	通常移動

■クエリ (前後にサイレントインターバルが入ります)

01 10 99 00 00 09 12 00 00 13 88 00 00 00 0A 00 00 27 10 00 1E 00 00 00 00 9F 82

■レスポンス 01 10 99 00 00 09 2E 93

・・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0009 _H	アドレス9900 _H ～9908 _H まで書き込み指定
バイト数	12 _H	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1388 _H	50(mm)×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1(mm)×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100(mm/sec)×100=10000→2710 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3(G)×100=30→001E _H
変更データ8(押付け) 入力単位(%)	0000 _H	0(%)→0 _H
変更データ9(制御フラグ)	0000 _H	通常動作のため何れのビットも0 0000 _b →0000 _H
エラーチェック	9F82 _H	CRCチェック計算結果→9F82 _H
エンド	なし	サイレントインターバル
合計バイト数	27	

ROBO CYLINDER

②軸No.0コントローラをピッチ送り(相対位置移動)させる使用例を示します

目標位置 (mm)	位置決め幅 (mm)	速度 (mm/sec)	加減速度 (G)	押付け (%)	移動制御
10	0.1	100	0.3	0	インクリメンタル

■クエリ (前後にサイレントインターバルが入ります)

01 10 99 00 00 09 12 00 00 03 E8 00 00 00 0A 00 00 27 10 00 1E 00 00 00 08 F3 A0

■レスポンス 01 10 99 00 00 09 2E 93

・クエリメッセージ中のバイト数と変更データを除いた部分のコピ-をレスポンスします。

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	9900 _H	開始アドレスは目標位置指定レジスタ9900 _H から
レジスタの数	0009 _H	アドレス9900 _H ~9908 _H まで書き込み指定
バイト数	12 _H	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	03E8 _H	10(mm)×100=1000→03E8 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1(mm)×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100(mm/sec)×100=10000→2710 _H
変更データ7(加減速度) 入力単位(0.01G)	001E _H	0.3(G)×100=30→001E _H
変更データ8(押付け) 入力単位(%)	0000 _H	0(%)→0 _H
変更データ9(制御フラグ)	0008 _H	相対移動のためビット3が1 (インクリメンタル設定) 1000b→0008 _H
エラーチェック	F3A0 _H	CRCチェック計算結果→F3A0 _H
エンド	なし	サイレントインターバル
合計バイト数	55	

5.6.2 ポジションテーブルデータ書込み

(1) 機能

このクエリを使用することによってポジションテーブル上のデータ変更が可能です。
 開始アドレス一覧(アドレス+0000_H~+000E_H)にアクセスがある毎に 1 ポジションデータ単位で EEPROM から読出され、書込み実施後、EEPROM に再び格納されます。
 ※EEPROMはデバイスの制約上、書込み回数が約10万回と制限されています。ポジションテーブルデータの書換えを頻繁に行くと短期間でEEPROMの書換え回数をオーバーし故障の原因となりますので、上位側のロジックは想定外のループ等が発生しないようご注意ください。

(2) 開始アドレス一覧

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_H + (16 \times \text{ポジション No.})_H + \text{アドレス(オフセット値)}_H$$

例 ポジションNo.200 の速度指令レジスタを変更したい場合

$$\begin{aligned} &1000_H + (16 \times 200 = 3200)_H + 4_H \\ &= 1000_H + C80_H + 4_H \\ &= 1C84_H \end{aligned}$$

『1C84』がクエリ開始アドレス部入力値になります。

※最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

■ポジションデータ変更レジスタ群

アドレス	記号	名称	符号	レジスタサイズ*	バイトサイズ*	入力単位
+0000	PCMD	目標位置	○	2	4	0.01mm
+0002	INP	位置決め幅		2	4	0.01mm
+0004	VCMD	速度指令		2	4	0.01mm/sec
+0006	ZNMP	個別ゾーン境界+側	○	2	4	0.01mm
+0008	ZNLP	個別ゾーン境界-側	○	2	4	0.01mm
+000A	ACMD	加速度指令		1	2	0.01G
+000B	DCMD	減速度指令		1	2	0.01G
+000C	PPOW	押付け時電流制限値		1	2	%
+000D	LPOW	負荷電流閾値		1	2	%
+000E	CTLF	制御フラグ指定		1	2	

※ '+' が付いているアドレスはオフセット値です。

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	10	直値指令
開始アドレス(H)	2	任意	5.6.2(2)開始アドレス一覧参照
レジスタの数(H)	2	任意	5.6.2(2)開始アドレス一覧参照
バイト数(H)	1	上記レジスタ数による	上記指定レジスタ数の2倍の数値 を入力
変更データ1(H)	2		5.6.2(2)開始アドレス一覧参照
変更データ2(H)	2		5.6.2(2)開始アドレス一覧参照
変更データ3(H)	2		5.6.2(2)開始アドレス一覧参照
:	:		:
エラーチェック(H)		CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	データ数 (バイト数)	RTUモード 8ビットデータ	備考
スタート	なし		サイレントインターバル
スレーブアドレス(H)	1	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	1	10	直値指令
開始アドレス(H)	2	任意	5.6.2(2)開始アドレス参照一覧
レジスタの数(H)	2	任意	5.6.2(2)開始アドレス参照一覧
エラーチェック(H)	2	CRC(16ビット)	
エンド	なし		サイレントインターバル
合計バイト数	最大256		

(5) レジスタ詳細説明

■目標位置(PCMD)

ポジション移動時の位置決め目標位置を絶対座標上の位置、または相対距離で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999 です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると移動開始指令時にアラームが発生します。絶対位置座標か相対距離の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって2種類の意味を持ちます。1つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、0～999999 です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■速度指定レジスタ(VCMD)

ポジション移動時の、移動速度を指定します。単位は 0.01mm/sec で設定範囲は 0～999999 です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■個別ゾーン境界±(ZNMP,ZNLP)

パラメータで設定されるゾーン境界とは別に、ポジション移動時だけ有効なゾーン信号を出力します。

絶対位置座標で表現した+側のゾーン信号出力境界値を ZNMP に、-側のゾーン信号出力境界値を ZNLP に設定します。現在位置がこの±境界値の内側にあるときは、ゾーンステータスレジスタの対応するビットが ON となります。設定単位は 0.01mm となります。設定可能範囲はともに -999999～999999 となりますが ZNMP > ZNLP の関係を満足している必要があります。

個別ゾーン出力を無効にする場合は、ZNMP と ZNLP を同じ値としてください。

■加速度指定レジスタ(ACMD)

ポジション移動時の、加速度を指定します。単位は 0.01G で設定範囲は 0～300 です。ただし、パラメータの最大加速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■減速度指定レジスタ(DCMD)

ポジション移動時の、減速度を指定します。単位は 0.01G で設定範囲は 0~300 です。ただし、パラメータの最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。この値は、 FF_H を最大電流とする 256 段階で設定します。押付け電流はアクチュエータの制約により設定可能範囲を 20%~70%($33_H \sim B3_H$)としてください。

■負荷出力電流閾値(LPOW)

負荷出力判定を行う場合、電流閾値を LPOW に設定します。この値は、 FF_H を最大電流とする 256 段階で設定します。判定を行わない場合は 0 を設定してください。

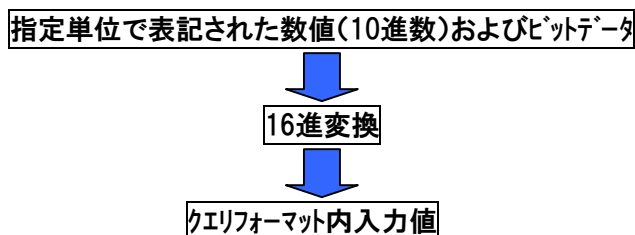
■制御フラグ指定レジスタ(CTLF)

このパラメータは以下の動作設定をするビットパターンです。

- ビット 1 = 0:通常動作
1:押付け動作
- ビット 2 = 0:アプローチ動作完了後の押付け動作の方向を正転とします。
1:アプローチ動作完了後の押付け動作の方向を逆転とします。
- ビット 3 = 0:通常動作
1:インクリメンタル動作

※未記載のビットは全て未使用となっています。

(6) 入力値(変更データ)について



(7) 使用例

軸No.0のポジションNo.12全データを下記のように書き換えます。

軸No.0

目標位置 (mm)	位置 決め幅 (mm)	速度 (mm/sec)	個別ゾーン 境界+側 (mm)	個別ゾーン 境界-側 (mm)	加速度 (G)	減速度 (G)	押付け (%)	閾値	移動制御
100	0.1	200	60	40	0.01	0.3	0	0	通常移動

■クエリ (前後にサイレントインターバルが入ります)

```
01 10 10 C0 00 0F 1E 00 00 27 10 00 00 00 0A 00 00 4E 20 00 00 17 70 00 00 0F A0 00 01
00 1E 00 00 00 00 00 00 70 1E
```

■受信レスポンス 01 10 10 C0 00 0F 84 F1

...クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします

■クエリメッセージ内訳

フィールド名称	RTUモード 8ビットデータ	備考
スタート	なし	サイレントインターバル
スレーブアドレス	01 _H	軸No.0+1
ファンクションコード	10 _H	
開始アドレス	10C0 _H	開始アドレスはポジションNo.12における目標位置指定レジスタ10C0 _H から※1
レジスタの数	000F _H	レジスタ記号PCMD~CTLFまで 合計15レジスタ書込み指定
バイト数	1E _H	15(レジスタ)×2=30(バイト)→1E _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	2710 _H	100(mm)×100=10000→2710 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	000A _H	0.1(mm)×100=10→000A _H
変更データ5、6(速度) 入力単位(0.01mm/sec)	0000 _H	32ビットデータ上位ビットは全て0
	4E20 _H	200(mm/sec)×100=20000→4E20 _H
	0FA0 _H	40(mm)×100=4000→0FA0 _H

次ページに続く

前ページからの続き

フィールド名称	RTUモード 8ビットデータ	備考
変更データ7、8 (個別ゾーン境界+) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	1770 _H	60(mm)×100=6000→1770 _H
変更データ9、10 (個別ゾーン境界-) 入力単位(0.01mm)	0000 _H	32ビットデータ上位ビットは全て0
	0FA0 _H	40(mm)×100=4000→0FA0 _H
変更データ11(加速度) 入力単位(0.01G)	0001 _H	0.01(G)×100=1→0001 _H
変更データ12(減速度) 入力単位(0.01G)	001E _H	0.3(G)×100=30→001E _H
変更データ13(押付け) 入力単位(%)	0000 _H	0(%)→0 _H
変更データ14(閾値) 入力単位(%)	0000 _H	0(%)→0 _H
変更データ15(制御フラグ)	0000 _H	通常動作のため何れのビットも0 0000 _b →0000 _H
エラーチェック	701E _H	CRCチェック計算結果→701E _H
エンド		サイレントインターバル
合計バイト数	39	

※1) 開始アドレス計算

例 ポジションNo.12 の全データを変更をかける為、本クエリの開始アドレス部には
ポジションNo.12 の目標位置アドレスとなる。

$$\begin{aligned}
 &1000_{\text{H}} + (16 \times 12 = 192)_{\text{H}} + 0_{\text{H}} \\
 &= 1000_{\text{H}} + \text{C0}_{\text{H}} + 0_{\text{H}} \\
 &= 10\text{C0}_{\text{H}}
 \end{aligned}$$

『10C0』が本クエリ開始アドレス部入力値になります。

下記は弊社 RC 用パソコンソフト上ポジションデータ内で、クエリメッセージ送信前と後の違いを表示したものです

■クエリ送信前

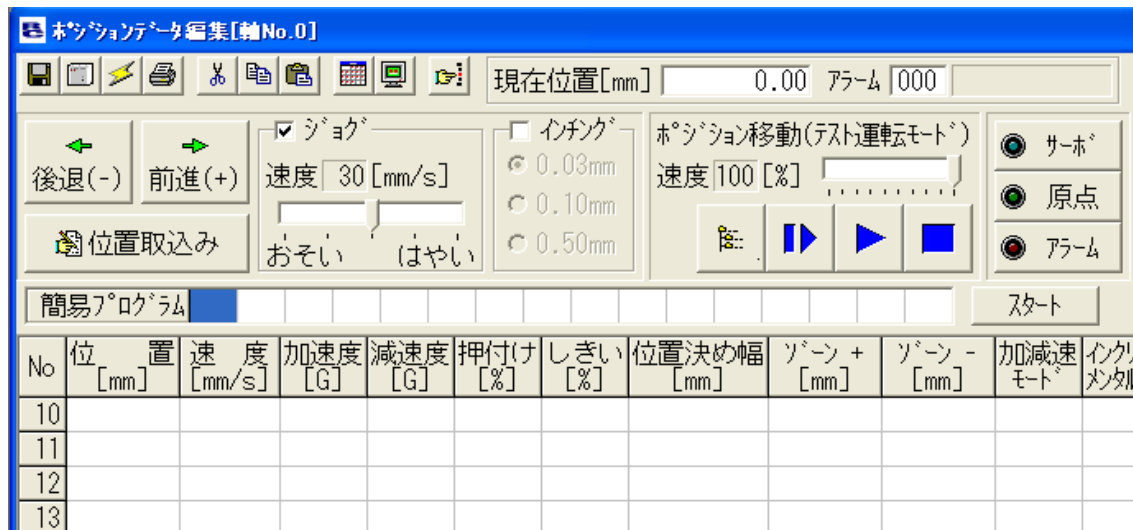


図 5.4

■送信後

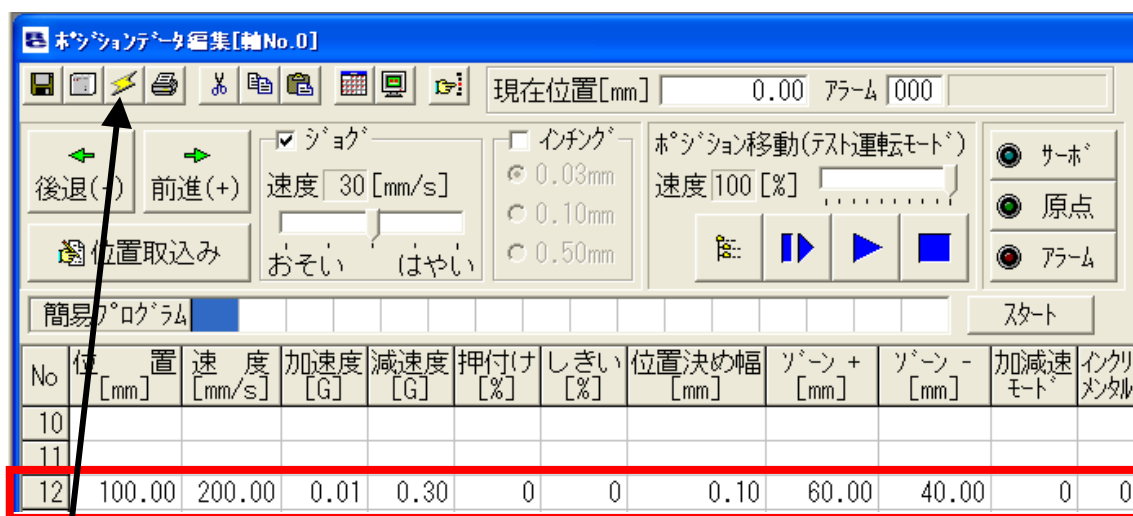



図 5.5

※  ボタンを押すか、ポジションデータ編集画面を開き直さないと書き換えた内容は表示されません。

R ROBO C CYLINDER

6 Modbus ASCII



6.1 メッセージフレーム(クエリ、レスポンス)

スタート	アドレス	ファンクションコード	データ	LRC チェック	エンド
1 文字	2 文字	2 文字	n 文字	2 文字	2 文字
1 byte	2 byte	2 byte	n×2 byte	2 byte	2 byte

※ASCIIコードでは、1文字は1バイト(2文字)で表されます。(6.2 ASCIIコード表を参照ください)

(1) スタート

ヘッダに相当し、ASCIIモードでは‘:’ (コロン)を使用します。
ASCIIコードでは、3A_Hと表されます。

(2) アドレス

接続されているRCコントローラのアドレス(01_H~10_H)を指定します。

$\boxed{\text{アドレス}} = \boxed{\text{軸番号}} + 1$

をASCIIコードで設定します。 例)軸番号1は、30_H32_Hとなります。



注意 アドレスは軸番号と同じでないため 設定にご注意ください。

(3) ファンクション

RCコントローラで使用可能なファンクションコード及び機能を示します。

コード		名 称	機 能
(Hex)	(ASCII)		
01 _H	30 _H 31 _H	Read Coil Status	コイル、DO の読出し
02 _H	30 _H 32 _H	Read Input Status	入力ステータス、DI の読出し
03 _H	30 _H 33 _H	Read Holding Registers	保持レジスタの読出し
04 _H	30 _H 34 _H	Read Input Registers	入力レジスタの読出し
05 _H	30 _H 35 _H	Force Single Coil	コイル、DO への1点書込み
06 _H	30 _H 36 _H	Preset Single Register	保持レジスタへの書込み
07 _H	30 _H 37 _H	Read Exception Status	例外ステータス読出し
0F _H	30 _H 46 _H	Force Multiple Coils	複数コイル、DO への一括書込み
10 _H	31 _H 30 _H	Preset Multiple Registers	複数保持レジスタへの一括書込み
11 _H	31 _H 31 _H	Report Slave ID	スレーブ ID 問い合わせ
17 _H	31 _H 37 _H	Read / Write Registers	レジスタへの読出し、書込み

※本書では マークのファンクションコードを使用しています。

※ROBONET ゲートウェイは、ASCIIモードをサポートしていません。

(4) データ

ファンクションコードで指示されたデータを付加する場合に用います。ファンクションコードでデータ付加の指示が無い場合は、データ無しも許されます。

(5) LRC チェック

ASCII モードでは、メッセージに LRC 方式に基づいて最初のコロンと CR/LF を除いたメッセージの中身をチェックするエラーチェックが自動的(※)に付加されます。また、チェックはメッセージ中の個別の文字(キャラクタ)のパリティチェック方式と関連せず行われます。

LRC チェックは 2 文字の ASCII コードで構成されています。LRC 値は、LRC をメッセージに付加する送信側が計算します。受信側は、メッセージ受信中に LRC を再計算して、その計算結果と送られてきた値とを比較します。もし、この 2 つの値が一致しなければ、結果はエラーとなります。

(※)ホスト側は、LRC の計算を行う関数を作成する必要があります。

●〈LRC チェックの計算例〉 領域がエラーチェック対象範囲

メッセージクエリが [':'] ["01"] ["05"] ["040B"] ["0000"] [LRC] [CR] [LF] の場合、

①最初に 1 バイト単位で数値を全て加算します。

$$\text{全加算値} = 01_{\text{H}} + 05_{\text{H}} + 04_{\text{H}} + 0B_{\text{H}} + 00_{\text{H}} + 00_{\text{H}} = 15_{\text{H}}$$

②次に 8 ビット長で この値に 2 の補数をとると FFFFFFFE_H となり

LRC は最下位の 1 バイトを除きます。従って LRC = "EB" となります。

(6) エンド

レーラ(デリミタ)に相当し、ASCII モードでは"CR/LF"を使用します。

ASCII コードでは、0D_H、0A_Hと表されます。

(7) ブロードキャスト

アドレスを 00_Hで指定すると接続されている全ての軸に 同一内容のクエリを送信することができます。この場合 RC コントローラからレスポンスは返信されません。

また、本機能は使用できるファンクションコード等に制限がありますので、十分注意してご利用ください。

使用できるファンクションコードは、「6.3 ASCII モード クエリー一覧」でご確認ください。

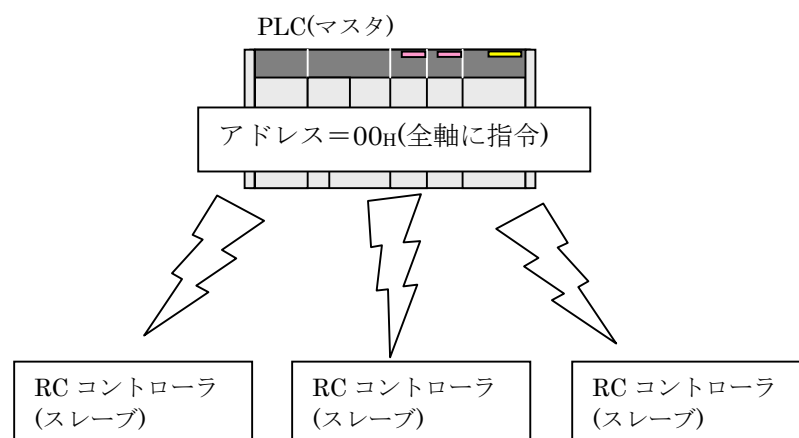


図 6.1



注意

・RCコントローラの送受信バッファサイズは、それぞれ 256 バイトとなっています。ホスト側から送信する伝文は受信バッファを、データをリクエストする場合は送信バッファを、それぞれオーバーしないように計算してください。

・データ数が奇数バイトになる場合には以下の理由から注意が必要です。

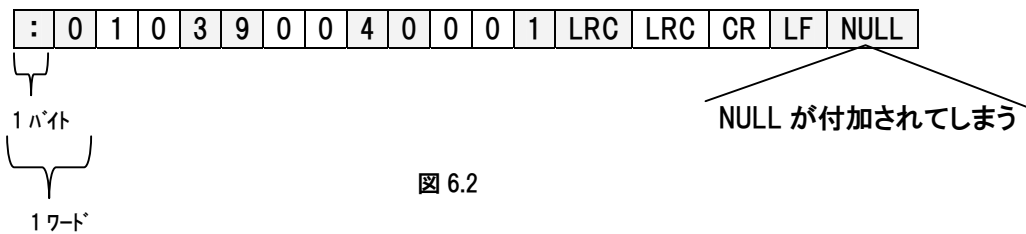
Modbus 通信の通信データはバイト単位です。

マスタ側では、データを 2 バイト単位で扱っている場合が多く、データ数が奇数になった場合には、自動的に 00_H<NULL>がパケットの最後に付加されてしまう場合があります。

RCコントローラは、マスタ側のインタフェースに Modbus RTU を使用して頂くことを基本にしています。通常 RTU モードで受信待機し、受信後 ASCII かどうかの判別を行っていますので、ヘッダ/データリミタの管理が行えません。

したがって、このような場合には、ASCII モードでの通信ができなくなります。

例)軸No.0 の出力ポート照会の場合



6.2 ASCIIコード表

ASCIIコード (□で囲まれた数字や文字(キャラクタ)を変換して伝送します)

上位 3bit 下位 4bit	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P		p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	“	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	‘	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	IS4	,	<	L	¥	l	
D	CR	IS4	—	=	M]	m	}
E	SO	IS4	.	>	N	^	n	
F	SI	IS4	/	?	O	—	o	DEL

- NUL:空文字
- ETX:テキスト終了
- ACK:肯定応答
- HT:水平タブ
- FF:改ページ
- SI:シフトイン
- NAC:否定応答
- CAN:取り消し
- ESC:拡張
- SOH:ヘディング開始
- EOT:伝送終了
- BEL:ベル
- LF:改行
- CR:復帰
- DLE:データリンクでの拡張
- SYN:同期文字
- EM:媒体終端
- SP:スペース
- STX:テキスト開始
- ENQ:問い合わせ
- BS:バックスペース
- VT:垂直タブ
- SO:シフトアウト
- DC*:制御装置*
- ETB:伝送ブロック終了
- DEL:削除

例) 「1」は、ASCIIコードで 31_H → 2進数表記では「00110001」

6.3 ASCIIモード クエリ一覧

FC	機能	記号	機能概要	PIOとの併用	ブロードキャスト	頁
03	ファンクションコード 03 複数レジスタ読み込み	無し	ファンクション 03 を使用するレジスタを連続的に複数読出すこと可能です。	○		140
03	現在位置モニタ	PNOW	アクチュエータの現在位置を 0.01mm 単位で読出します	○		143
03	現在発生アラームコード照会	ALMC	現在発生中のアラームコードを読出します。	○		145
03	入力ポート照会	DIPM	PIO 入力ポート ON/OFF 状態を読出します。	○		147
03	出力ポート照会	DOPM	PIO 出力ポート ON/OFF 状態を読出します。	○		151
03	デバイスステータス 1 照会 (運転準備ステータス)	DSS1	次の 12 項目の状態(ステータス)を読出します。 ① 非常停止 ② セーフティ速度有効/無効 ③ コントローラレディ ④ サーボ ON/OFF ⑤ 押付け空振り ⑥ 重故障 ⑦ 軽故障 ⑧ アブソエラー ⑨ ブレーキ ⑩ 一時停止 ⑪ 原点復帰完了 ⑫ 位置決め完了	○		155
03	デバイスステータス 2 照会 (運転情報 1 ステータス)	DSS2	次の 15 項目の状態(ステータス)を読出します。 ① イネーブル ② 負荷出力判定(検定範囲負荷電流閾値) ③ トルクレベル(負荷電流閾値) ④ ティーチモード(通常/ティーチ) ⑤ ポジションデータ取込(通常/完了) ⑥ ジョグ+(通常/指令中) ⑦ ジョグ-(通常/指令中) ⑧ 完了ポジション 7 ⑨ 完了ポジション 6 ⑩ 完了ポジション 5 ⑪ 完了ポジション 4 ⑫ 完了ポジション 3 ⑬ 完了ポジション 2 ⑭ 完了ポジション 1 ⑮ 完了ポジション 0	○		157

FC:ファンクションコード

PIO:パラレル I/O(I/O コネクタの入出力)

PIO との併用欄、ブロードキャスト欄の○印は、PIO との併用、ブロードキャストが有効なクエリを表しています

FC	機能	記号	機能概要	PIOとの併用	プロードキャスト	頁
03	拡張デバイスステータス照会 (運転情報2ステータス)	DSSE	次の9項目の状態(ステータス)を読出します。 ① 非常停止(非常停止入力ポート) ② モータ電圧低下 ③ 運転モード(AUTO/MANU) ④ 原点復帰中 ⑤ 押付け動作中 ⑥ 励磁検出 ⑦ PIO/Modbus 切替 ⑧ ポジションデータ書込み完了ステータス ⑨ 移動中	○		159
03	システムステータス照会 (コントローラステータス)	STAT	次の7項目の状態(ステータス)を読出します。 ① 自動サーボOFF中 ② EEPROMアクセス中 ③ 運転モード(AUTO/MANU) ④ 原点復帰完了 ⑤ サーボON/OFF ⑥ サーボ指令 ⑦ 駆動源ON(通常/遮断中)	○		161
03	現在速度モニタ	VNOW	アクチュエータの現在速度を 0.01mm/sec 単位で読出します	○		163
03	電流値モニタ	CNOW	アクチュエータモータトルク電流指令値を 1mA 単位で読出します	○		165
03	偏差モニタ	DEVI	1ms 周期毎の偏差量を 1pulse 単位で読出します	○		167
03	システムタイムモニタ	STIM	コントローラ電源投入時からの積算時間を 1msec 単位で読出します。	○		169
03	特殊入力ポート照会 (センサ入力ステータス)	SIPM	次の9項目の状態(ステータス)を読出します。 ① 指令パルス NP ② 指令パルス PP ③ ポートスイッチ ④ モードスイッチ ⑤ イネーブルスイッチ ⑥ 原点確認センサ ⑦ オーバートラベルセンサ ⑧ クリープセンサ ⑨ リミットセンサ	○		171

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
03	ゾーンステータス照会	ZONS	次の 6 項目の状態(ステータス)を読出します。 ① LS2(PIO パターン電磁弁モード[3 点タイプ]) ② LS1(PIO パターン電磁弁モード[3 点タイプ]) ③ LS0(PIO パターン電磁弁モード[3 点タイプ]) ④ ポジションゾーン ⑤ ゾーン 2 ⑥ ゾーン 1	○		173
03	位置決め完了ポジションNo.照会	POSS	次の 9 項目の状態(ステータス)を読出します。 ① 完了ポジション番号ビット 256 ② 完了ポジション番号ビット 128 ③ 完了ポジション番号ビット 64 ④ 完了ポジション番号ビット 32 ⑤ 完了ポジション番号ビット 16 ⑥ 完了ポジション番号ビット 8 ⑦ 完了ポジション番号ビット 4 ⑧ 完了ポジション番号ビット 2 ⑨ 完了ポジション番号ビット 1	○		175
05	セーフティ速度モード切替	SFTY	有効/無効モード切替を指令します。		○	178
05	サーボ ON/OFF	SON	サーボ ON/OFF を指令します。		○	180
05	アラームリセット	ALRS	アラームリセット/残移動量キャンセルを指令します。		○	182
05	ブレーキ解除	BKRL	ブレーキ強制解除を指令します。		○	184
05	一時停止	STP	一時停止を指令します。		○	186
05	原点復帰	HOME	原点復帰動作を指令します。		○	188
05	位置決め動作起動	CSTR	ポジションNo.指定移動時スタート信号 ON/OFF (本書では使用しません。)		○	190
05	ジヨグ/インチング切替	JISL	ジヨグ/インチングモード切替を行います。		○	192
05	ティーチモード指令	MOD	通常/教示モード切替を行います。		○	194
05	ポジションデータ取込	TEAC	教示モード時、現在位置取込指令を行います		○	196
05	ジヨグ+	JOG+	反原点方向にジヨグ/インチング動作を指令します。		○	198
05	ジヨグ-	JOG-	原点方向にジヨグ/インチング動作を指令します。		○	200

FC	機能	記号	機能概要	PIOとの併用	プロトタイプキャスト	頁
05	ポジションNo.0~7 指定	ST0~ST7	電磁弁モード時に有効なポジションNo.指定 この指令だけでアクチュエータが動作可能です。※		○	202
05	PIO/Modbus 切替設定	PMSL	PIO 外部指令信号の有効/無効切替指令		○	204
05	減速停止	STOP	移動中のアクチュエータを減速停止させる事が出来ます。		○	206
06	ポジションNo.指定移動	POSR	この指令だけでポジションNo.指定移動で通常移動、 相対移動、押付け動作が可能です。※		○	208
10	直値移動指令	無し	目標位置、位置決め幅、速度、加減速度、押付け、 制御設定を 1 つの伝文で送信し、動作させる事が 可能です。通常移動、相対移動、押付け動作が 可能です。		○	211
10	ポジションデータテーブル 書込	無し	指定された軸、ポジションNo.のデータを全て変更する ことが可能です。		○	222
不定	例外レスポンス	無し	伝文内容が不正データであった場合のレスポンスで す。			230

※あらかじめポジションテーブルへの位置決めデータの設定が必要です。

- ① ティーチングツールを使用する
- ② ポジションデータテーブル書込み機能を使用する。

6.4 データ、ステータス読み取り(コード 03 使用ケリ)

6.4.1 レジスタの連続複数読み取り

※)6.2 ASCIIコード表
を参照ください。

(1) 機能

スレーブの保持レジスタの内容を読み取ります。

ブロードキャストではサポートされていません。

(2) 開始アドレス一覧

RCシリーズコントローラでは送信・受信のバッファサイズは、それぞれ256バイトとなっています。このため、ASCIIモード使用時256バイト中9バイト(ヘッダ+スレーブアドレス+ファンクションコード+エラーチェック+トレーラ)を除く247バイト分最大123レジスタ分(1レジスタ2バイト使用)のデータ照会が可能ですので下記一覧のデータ(合計21レジスタ)を一度の送受信で照会することが可能となります。

アドレス(H)	記号	名称	符号	レジスタサイズ*	バイト
9000	PNOW	現在位置モニタ	○	2	4
9002	ALMC	現在発生アラームコード照会		1	2
9003	DIPM	入力ポート照会		1	2
9004	DOPM	出力ポートモニタ照会		1	2
9005	DSS1	デバイスステータス 1 照会		1	2
9006	DSS2	デバイスステータス 2 照会		1	2
9007	DSSE	拡張デバイスステータス照会		1	2
9008	STAT	システムステータス照会		2	4
900A	VNOW	現在速度モニタ	○	2	4
900C	CNOW	電流値モニタ	○	2	4
900E	DEVI	偏差モニタ	○	2	4
9010	STIM	システムタイム照会		2	4
9012	SIPM	特殊入力ポート照会		1	2
9013	ZONS	ゾーンステータス照会		1	2
9014	POSS	位置決め完了ポジションNo.ステータス照会		1	2

(3) クエリフォーマット

クエリメッセージでは、読取りを開始するレジスタのアドレスとレジスタの数を指定します。

1レジスタ=2バイト=16ビットデータ

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタ読出しコード
開始アドレス(H)	4	任意	6.4.1(2)開始アドレス 一覧参照
レジスタの数(H)	4	任意	開始アドレス一覧参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	CR / LF	
合計バイト数	17	17	

(4) レスポンスフォーマット

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	保持レジスタ読出しコード
データバイト数(H)	2		クエリフォーマット内 指定レジスタ数×2
データ1(H)	4		
データ2(H)	4		
データ3(H)	4		
データ4(H)	4		
:	:		
:	:		
エラーチェック(H)	2	LRC計算結果	
トレーラ		CR / LF	
合計バイト数			



(5) 使用例

下記に軸No.0 RCコントローラのアドレス9000_H～9009_Hまでを読み取るクエリ例を示します。

クエリ:010390000000A62[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
開始アドレス(H)	‘9’, ‘0’, ‘0’, ‘0’	39303030
レジスタの数(H)	‘0’, ‘0’, ‘0’, ‘A’	30303041
エラーチェック(H)	‘6’, ‘2’	3632
トレーラ	‘CR’, ‘LF’	0D0A

レスポンスは以下のようになります。

レスポンス:01031400000000000000B80162002000800031C7000800111C[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
データバイト数(H)	‘1’, ‘4’ (20バイト= 10レジスタ)	3134
データ1(H)	‘0’, ‘0’, ‘0’, ‘0’, ‘0’, ‘0’, ‘0’, ‘0’	3030303030303030
データ2(H)	‘0’, ‘0’, ‘0’, ‘0’	30303030
データ3(H)	‘B’, ‘8’, ‘0’, ‘1’	42383031
データ4(H)	‘6’, ‘2’, ‘0’, ‘0’	36323030
データ5(H)	‘2’, ‘0’, ‘0’, ‘0’	32303030
データ6(H)	‘8’, ‘0’, ‘0’, ‘0’	38303030
データ7(H)	‘3’, ‘1’, ‘C’, ‘7’	33314337
データ8(H)	‘0’, ‘0’, ‘0’, ‘8’, ‘0’, ‘0’, ‘1’, ‘1’	3030303830303131
エラーチェック(H)	‘1’, ‘C’ (LRC計算による)	3143
トレーラ	‘CR’, ‘LF’	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.2 現在位置の読み取り(0.01mm 単位)モニタ 《PNOW》

(1) 機能

現在位置を0.01mm単位で読み取ります。符号は有効です。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', '0'	現在位置モニタ
レジスタの数(H)	4	'0', '0', '0', '2'	アドレス9000 _H ~9001 _H 呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '4'	2レジスタ呼出=4バイト
データ1(H)	4	現在値による	現在値データ(Hex)
データ2(H)	4	現在値による	現在値データ(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例(軸No.が0の場合)

下記に軸No.0 RCコントローラのアドレス9000_Hを読み取るクエリ例を示します。

クエリ: 0103900000026A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	':'	3A
スレーブアドレス	'0', '1'	3031
ファンクションコード	'0', '3'	3033
開始アドレス	'9', '0', '0', '0'	39303030
レジスタの数	'0', '0', '0', '2'	30303032
エラーチェック	'6', 'A'	3641
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス: 010304000013885D[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '4' (4バイト= 2レジスタ)	3034
データ1(H)	'0', '0', '0', '0'	30303030
データ2(H)	'1', '3', '8', '8'	31333838
エラーチェック(H)	'5', 'D' (LRC計算による)	3544
トレーラ	'CR', 'LF'	0D0A

現在位置は『00001388』→10進数に変換→5000(×0.01mm)

現在位置は 50mm

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.3 現在発生アラームコードの読み取り《ALMC》

(1) 機能

コントローラの正常状態、アラーム状態をコードを読み取ります。

非アラーム状態では00_Hが格納されています。

コードの詳細内容は各コントローラの取扱説明書をご参照ください。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', '2'	現在発生アラームコード
レジスタの数(H)	4	'0', '0', '0', '1'	アドレス9002 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '2'	1レジスタ呼出=2バイト
データ1(H)	4	アラームコード	アラームコード(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

下記に軸No.0 RCコントローラのアドレス9002_Hまでを読み取りしたクエリ例を示します。

クエリ: 01039002000169[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	':'	3A
スレーブアドレス	'0', '1'	3031
ファンクションコード	'0', '3'	3033
開始アドレス	'9', '0', '0', '2'	39303032
レジスタの数	'0', '0', '0', '1'	30303031
エラーチェック	'6', '9'	3639
トレーラ	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス: 01030200E812[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
ヘッダ	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'0', '0', 'E', '8'	30304538
エラーチェック(H)	'1', '2' (LRC計算による)	3132
トレーラ	'CR', 'LF'	0D0A

このとき発生している最重要アラームは 『0E8』_H

A,B 相断線検出アラームです。

コードの詳細内容は各コントローラの取扱説明書を参照してください。

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.4 I/Oポート入力信号の状態読み取り《DIPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート入力値を読み取ります。

データは RCコントローラが入力として認識しているポートの状態です。

(2) クエリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', '3'	入力ポートモニタレジスタ
レジスタの数(H)	4	'0', '0', '0', '1'	アドレス9003 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、17アドレス当り16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '2'	1レジスタ呼出=2バイト
データ1(H)	4	DI入力値	DI入力値(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

軸No.0コントローラの入力ポート(アドレス9003_H)を読み取りした使用例を示します

クエリ:01 03 90 03 00 01 68 [CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', '3'	39303033
レジスタの数(H)	'0', '0', '0', '1'	30303031
エラーチェック(H)	'6', '8' (LRC計算による)	3638
エンド	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:01 03 02 B8 01 14 [CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'B', '8', '0', '1'	42383031
エラーチェック(H)	'1', '4' (LRC計算による)	3134
エンド	'CR', 'LF'	0D0A

入力ポートデータ部は『B801』_H →2進数変換『101110000000001』

※レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け (詳細は、各 RC コントローラの取扱説明書をご参照ください)

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)			
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3
IN0	PC1	ST0	PC1	PC1
IN1	PC2	ST1	PC2	PC2
IN2	PC4	ST2	PC4	PC4
IN3	HOME	0	PC8	PC8
IN4	CSTR	RES	CSTR	CSTR
IN5	*STP	*STP	*STP	*STP
IN6	0	0	0	0
IN7	0	0	0	0
IN8	0	0	0	0
IN9	0	0	0	0
IN10	0	0	0	0
IN11	0	0	0	0
IN12	0	0	0	0
IN13	0	0	0	0
IN14	0	0	0	0
IN15	0	0	0	0

ポート	PCON-C/CG						PCON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

ROBO CYLINDER

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	TL	TL
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	RES	RES/ DCLR
IN4	PC16	PC16	PC16	PC16	ST4	0	0	0
IN5	PC32	PC32	PC32	PC32	ST5	0	0	0
IN6	0	MODE	PC64	PC64	ST6	0	0	0
IN7	0	JISL	PC128	PC128	0	0	0	0
IN8	0	JOG+	0	PC256	0	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0	0
IN15	SON	SON	SON	SON	SON	SON	0	0

ポート	SCON						(ハルス列モード)
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0
IN0	PC1	PC1	PC1	PC1	ST0	ST0	SON
IN1	PC2	PC2	PC2	PC2	ST1	ST1	RES
IN2	PC4	PC4	PC4	PC4	ST2	ST2	HOME
IN3	PC8	PC8	PC8	PC8	ST3	0	TL
IN4	PC16	PC16	PC16	PC16	ST4	0	CSTR
IN5	PC32	PC32	PC32	PC32	ST5	0	DCLR
IN6	0	MODE	PC64	PC64	ST6	0	BKRL
IN7	0	JISL	PC128	PC128	0	0	RMOD
IN8	0	JOG+	0	PC256	0	0	0
IN9	BKRL	JOG-	BKRL	BKRL	BKRL	BKRL	0
IN10	RMOD	RMOD	RMOD	RMOD	RMOD	RMOD	0
IN11	HOME	HOME	HOME	HOME	HOME	0	0
IN12	*STP	*STP	*STP	*STP	*STP	0	0
IN13	CSTR	CSTR/ PWRT	CSTR	CSTR	0	0	0
IN14	RES	RES	RES	RES	RES	RES	0
IN15	SON	SON	SON	SON	SON	SON	0

6.4.5 I/Oポート出力信号の状態読み取り《DOPM》

(1) 機能

PIOパターンに関係なく、RCコントローラのポート出力値を読み取ります。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', '4'	出力ポートモニタレジスタ
レジスタの数(H)	4	'0', '0', '0', '1'	アドレス9004 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '2'	1レジスタ呼出=2バイト
データ1(H)	4	DO出力値	DI出力値(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	15		

(4) 使用例(軸No.が0の場合)

軸No.0コントローラの入力ポート(アドレス9004_H)を読み取りした使用例を示します

クエリ:01039004000167[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', '4'	39303034
レジスタの数(H)	'0', '0', '0', '1'	30303031
エラーチェック(H)	'6', '7' (LRC計算による)	3637
エンド	'CR', 'LF'	0D0A

レスポンスは以下のようになります。

レスポンス:010302740086[CR][LF]

フィールド名	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'7', '4', '0', '0'	37343030
エラーチェック(H)	'8', '6' (LRC計算による)	3836
エンド	'CR', 'LF'	0D0A

出力ポート部は『7400』_H →2進数変換『0111010000000000』

※レスポンス例のデータは一例であり、各種状態により異なります。

(5) ポート割付け (詳細は、各 RC コントローラの取扱説明書をご参照ください)

各 RC コントローラの PIO パターンに対するポート割付けを記載します。

0 は、レスポンスデータが常に 0 であることを示しています。

ポート	ERC2(PIOタイプ)			
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3
OUT0	PEND	PE0	PEND	PEND
OUT1	HEND	PE1	HEND	HEND
OUT2	ZONE	PE2	ZONE	ZONE
OUT3	*ALM	*ALM	*ALM	*ALM
OUT4	0	0	0	0
OUT5	0	0	0	0
OUT6	0	0	0	0
OUT7	0	0	0	0
OUT8	0	0	0	0
OUT9	0	0	0	0
OUT10	0	0	0	0
OUT11	0	0	0	0
OUT12	0	0	0	0
OUT13	0	0	0	0
OUT14	0	0	0	0
OUT15	0	0	0	0

ポート	PCON-C/CG						PCON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/ TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	0	0	0	0	0	0	0	0

ROBO CYLINDER

ポート	ACON-C/CG						ACON-PL/PO	
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0	PIO パターン 1
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	SV	SV
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	INP	INP/ TLR
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	HEND	HEND
OUT3	PM8	PM8	PM8	PM8	PE3	0	*ALM	*ALM
OUT4	PM16	PM16	PM16	PM16	PE4	0	0	0
OUT5	PM32	PM32	PM32	PM32	PE5	0	0	0
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	0	0
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	0	0
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	0	0
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	0	0
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	0	0
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	0	0
OUT12	SV	SV	SV	SV	SV	SV	0	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0	0
OUT15	TRQS	0	TRQS	TRQS	TRQS	0	0	0

ポート	SCON						(ハルス列モード)
	PIO パターン 0	PIO パターン 1	PIO パターン 2	PIO パターン 3	PIO パターン 4	PIO パターン 5	PIO パターン 0
OUT0	PM1	PM1	PM1	PM1	PE0	LS0	PWR
OUT1	PM2	PM2	PM2	PM2	PE1	LS1	SV
OUT2	PM4	PM4	PM4	PM4	PE2	LS2	INP
OUT3	PM8	PM8	PM8	PM8	PE3	0	HEND
OUT4	PM16	PM16	PM16	PM16	PE4	0	TLR
OUT5	PM32	PM32	PM32	PM32	PE5	0	*ALM
OUT6	MOVE	MOVE	PM64	PM64	PE6	0	*EMGS
OUT7	ZONE1	MODES	PM128	PM128	ZONE1	ZONE1	RMDS
OUT8	PZONE	PZONE	PZONE	PM256	PZONE	PZONE	ALM1
OUT9	RMDS	RMDS	RMDS	RMDS	RMDS	RMDS	ALM2
OUT10	HEND	HEND	HEND	HEND	HEND	HEND	ALM4
OUT11	PEND	PEND/ WEND	PEND	PEND	PEND	0	ALM8
OUT12	SV	SV	SV	SV	SV	SV	0
OUT13	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	*EMGS	0
OUT14	*ALM	*ALM	*ALM	*ALM	*ALM	*ALM	0
OUT15	*BALM	*BALM	*BALM	*BALM	*BALM	*BALM	0

6.4.6 コントローラ状態信号の読み取り1《DSS1》

(1) 機能

コントローラ内のステータスを読み取ります。

「4.3.2 (4) デバイスステータスレジスタ1 内容」をご参照ください。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘0’, ‘5’	デバイスステータス レジスタ1
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9005 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1アドレス当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	ステータス1	ステータス1(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのデバイスステータス(アドレス9005_H)を読み取る使用例を示します

クエリ:01 03 90 05 00 01 66[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', '5'	39303035
レジスタの数(H)	'0', '0', '0', '1'	30303031
エラーチェック(H)	'6', '6' (LRC計算による)	3636
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 30 88 42[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'3', '0', '8', '8'	33303838
エラーチェック(H)	'4', '2' (LRC計算による)	3432
エンド	'CR', 'LF'	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.7 コントローラ状態信号の読み取り2 《DSS2》

(1) 機能

コントローラ内のステータス2を読み取ります。

「4.3.2 (5) デバイスステータスレジスタ2 内容」をご参照ください。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘0’, ‘6’	デバイスステータス レジスタ2
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9006 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	コントローラ内のステータス
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	ステータス2	ステータス2(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのデバイスステータス2(アドレス9006_H)を読み取る使用例を示します

クエリ:01 03 90 06 00 01 65[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', '6'	39303036
レジスタの数(H)	'0', '0', '0', '1'	30303031
エラーチェック(H)	'6', '5' (LRC計算による)	3635
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 80 00 7A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'8', '0', '0', '0'	38303030
エラーチェック(H)	'7', 'A' (LRC計算による)	3741
エンド	'CR', 'LF'	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.8 コントラ状態信号の読み取り3 《DSSE》

(1) 機能

コントラ内のステータス(拡張デバイス)を読み取ります。

「4.3.2 (6) 拡張デバイスステータスレジスタ内容」をご参照ください。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘0’, ‘7’	拡張デバイスステータス レジスタ
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9007 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	拡張ステータス	拡張ステータス(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの拡張デバイスステータス(アドレス9007_H)を読み取る使用例を示します

クエリ:01 03 90 07 00 01 64[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
開始アドレス(H)	‘9’, ‘0’, ‘0’, ‘7’	39303037
レジスタの数(H)	‘0’, ‘0’, ‘0’, ‘1’	30303031
エラーチェック(H)	‘6’, ‘4’ (LRC計算による)	3634
エンド	‘CR’, ‘LF’	0D0A

レスポンスは以下のようになります。

レスポンス:01 03 02 33 C7 00[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
データバイト数(H)	‘0’, ‘2’ (2バイト= 1レジスタ)	3032
データ1(H)	‘3’, ‘3’, ‘C’, ‘7’	33334337
エラーチェック(H)	‘0’, ‘0’ (LRC計算による)	3030
エンド	‘CR’, ‘LF’	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.9 コントローラ状態信号の読み取り4《STAT》

(1) 機能

コントローラの内部動作状態を読み取ります。

「4.3.2 (7) システムステータスレジスタ内容」をご参照ください。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘0’, ‘8’	システムステータス レジスタ
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス9008 _H ~9009 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	コントローラ内のステータス
データバイト数(H)	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ(H)	8	システムステータス	システムステータス(Hex)
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラのシステムステータス(アドレス9008_H)を読み取る使用例を示します

クエリ:01 03 90 08 00 02 62[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', '8'	39303038
レジスタの数(H)	'0', '0', '0', '2'	30303032
エラーチェック(H)	'6', '2' (LRC計算による)	3632
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 0C 00 11 DB[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '4' (4バイト= 2レジスタ)	3034
データ1(H)	'0', '0', '0', 'C'	30303043
データ2(H)	'0', '0', '1', '1'	30303131
エラーチェック(H)	'D', 'B' (LRC計算による)	4442
エンド	'CR', 'LF'	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.10 現在速度の読み取り《VNOW》

(1) 機能

モータ実速度を読み取ります。移動方向により±に振れます。

単位は0.01mm/secです。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘0’, ‘A’	現在速度モニタ
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘2’	アドレス900A _H ~900B _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘4’	2レジスタ呼出=4バイト
データ(H)	8	現在速度	現在速度(Hex) 単位は0.01mm/sec。
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの速度(アドレス900A_H)を読み取る使用例を示します

ケリ:01 03 90 0A 00 02 60[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
開始アドレス(H)	‘9’, ‘0’, ‘0’, ‘A’	39303041
レジスタの数(H)	‘0’, ‘0’, ‘0’, ‘2’	30303032
エラーチェック(H)	‘6’, ‘0’ (LRC計算による)	3630
エンド	‘CR’, ‘LF’	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 26 FC D6[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
データバイト数(H)	‘0’, ‘4’ (4バイト= 2レジスタ)	3034
データ1(H)	‘0’, ‘0’, ‘0’, ‘0’	30303030
データ2(H)	‘2’, ‘6’, ‘F’, ‘C’	32364643
エラーチェック(H)	‘D’, ‘6’ (LRC計算による)	4436
エンド	‘CR’, ‘LF’	0D0A

現在速度は『000026FC』→10進数に変換→9980(×0.01mm/sec)

現在速度モニタは 99.8mm/sec

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.11 電流値の読み取り《CNOW》

(1) 機能

モータ電流のモニタデータです。

単位はmAです。

トルク電流指令値が格納されます。

(2) ケリフォーマット

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', 'C'	電流値モニタ
レジスタの数(H)	4	'0', '0', '0', '2'	アドレス900C _H ~900D _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '4'	2レジスタ呼出=4バイト
データ(H)	8	モータ電流の モニタ	モータ電流のモニタ(Hex) 単位はmA
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの電流値(アドレス900C_H)を読み取る使用例を示します

クエリ:01 03 90 0C 00 02 5E[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', 'C'	39303043
レジスタの数(H)	'0', '0', '0', '2'	30303032
エラーチェック(H)	'5', 'E' (LRC計算による)	3545
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 01 C8 2F[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '4' (4バイト= 2レジスタ)	3034
データ1(H)	'0', '0', '0', '0'	30303030
データ2(H)	'0', '1', 'C', '8'	30314338
エラーチェック(H)	'2', 'F' (LRC計算による)	3246
エンド	'CR', 'LF'	0D0A

モータ電流値は『000001C8』→10進数に変換→456(mA)

電流モニタ値は 456mA

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.12 偏差の読み取り《DEVI》

(1) 機能

1ms周期毎の位置指令値とフィードバック値(実位置)の偏差量を読み取ります。単位はパルスです。

モータ機械角1回転あたりのパルス数は使用エンコーダにより異なります。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '0', 'E'	偏差モニタ
レジスタの数(H)	4	'0', '0', '0', '2'	アドレス900E _H ~900F _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '4'	2レジスタ呼出=4バイト
データ(H)	8	偏差モニタ	偏差モニタ(Hex) 単位はpulse
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		

(4) 使用例

軸No.0コントローラの偏差量(アドレス90E_H)を読み取る使用例を示します

クエリ:01 03 90 0E 00 02 5C[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '0', 'E'	39303045
レジスタの数(H)	'0', '0', '0', '2'	30303032
エラーチェック(H)	'5', 'C' (LRC計算による)	3543
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 00 00 00 83 75[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '4' (4バイト= 2レジスタ)	3034
データ1(H)	'0', '0', '0', '0'	30303030
データ2(H)	'0', '0', '8', '3'	30303833
エラーチェック(H)	'7', '5' (LRC計算による)	3735
エンド	'CR', 'LF'	0D0A

偏差モニタは『00000083』→10進数に変換→131(pulse)

1ms 周期毎の位置指令値とフィードバック値(実位置)の偏差量は 131pulse

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.13 電源投入後の積算時間の読み取り《STIM》

(1) 機能

コントローラ電源投入時からの積算時間を読み取ります。

単位はmsecです。

ソフトウェアリセットではクリアされません。

(2) クエリフォーマット

フィールド名	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
開始アドレス(H)	4	'9', '0', '1', '0'	システムタイマ
レジスタの数(H)	4	'0', '0', '0', '2'	アドレス9010 _H ~9011 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	'0', '3'	レジスタの読み出し
データバイト数(H)	2	'0', '4'	2レジスタ呼出=4バイト
データ(H)	8	システムタイマ	システムタイマ(Hex) 単位はmsec
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	19		



(4) 使用例

軸No.0コントローラの起動時間(アドレス9010_H)を読み取り使用例を示します

クエリ:01 03 90 10 00 02 5A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '1', '0'	39303130
レジスタの数(H)	'0', '0', '0', '2'	30303032
エラーチェック(H)	'5', 'A' (LRC計算による)	3541
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 04 02 38 C0 94 6A[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '4' (4バイト= 2レジスタ)	3034
データ1(H)	'0', '2', '3', '8'	30323338
データ2(H)	'C', '0', '9', '4'	43303934
エラーチェック(H)	'6', 'A' (LRC計算による)	3641
エンド	'CR', 'LF'	0D0A

システムタイムは『0238C094』→10進数に変換→37273748(msec)

コントローラ電源投入時からの積算時間は 10. 3538 時間

※レスポンス例のデータは一例であり、各種状態により異なります。

6. 4. 14 特殊入力ポートの入力信号状態の読み取り《SIPM》

(1) 機能

通常の入力ポート以外の入力ポートの状態を読み取ります。

特殊入力ポートの内容は「4. 3. 2 (8) 特殊入力ポートレジスタ内容」
をご参照ください。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘1’, ‘2’	特殊入力ポートモニタ
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9012 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	特殊ポートモニタ	4.3.2(8) 一覧表参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの特異入力ポート(アドレス9012_H)を読み取る使用例を示します

クエリ:01 03 90 12 00 01 59[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
開始アドレス(H)	‘9’, ‘0’, ‘1’, ‘2’	39303132
レジスタの数(H)	‘0’, ‘0’, ‘0’, ‘1’	30303031
エラーチェック(H)	‘5’, ‘9’ (LRC計算による)	3539
エンド	‘CR’, ‘LF’	0D0A

レスポンスは以下のようになります。

レスポンス:01 03 02 03 00 F7[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
データバイト数(H)	‘0’, ‘2’ (2バイト= 1レジスタ)	3032
データ1(H)	‘0’, ‘3’, ‘0’, ‘0’	30333030
エラーチェック(H)	‘F’, ‘7’ (LRC計算による)	4637
エンド	‘CR’, ‘LF’	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6. 4. 15 ゾーン出力信号の状態の読み取り《ZONS》

(1) 機能機能

ゾーン出力の状態を読み取ります。

「4. 3. 2 (9) ゾーンステータスレジスタ内容」をご参照ください。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘1’, ‘3’	ゾーンステータス照会
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9013 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当り16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	ゾーンステータス	4.3.2(9)一覧表参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラのゾーンステータス(アドレス9013_H)を読み取る使用例を示します

クエリ:01 03 90 13 00 01 58[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
開始アドレス(H)	‘9’, ‘0’, ‘1’, ‘3’	39303133
レジスタの数(H)	‘0’, ‘0’, ‘0’, ‘1’	30303031
エラーチェック(H)	‘5’, ‘8’ (LRC計算による)	3538
エンド	‘CR’, ‘LF’	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 00 00 FA[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘3’	3033
データバイト数(H)	‘0’, ‘2’ (2バイト= 1レジスタ)	3032
データ1(H)	‘0’, ‘0’, ‘0’, ‘0’	30303030
エラーチェック(H)	‘F’, ‘A’ (LRC計算による)	4641
エンド	‘CR’, ‘LF’	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.4.16 位置決め完了ポジションNo.照会 《POSS》

(1) 機能

完了ポジション番号を読み取ります。

「4.3.2 (10) ポジション番号ステータスレジスタ内容」をご参照ください。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
開始アドレス(H)	4	‘9’, ‘0’, ‘1’, ‘4’	ポジション番号ステータス
レジスタの数(H)	4	‘0’, ‘0’, ‘0’, ‘1’	アドレス9014 _H の呼出
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンスフォーマット

レスポンスメッセージにおけるデータは、1レジスタ当たり16ビットのデータです。

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘3’	レジスタの読み出し
データバイト数(H)	2	‘0’, ‘2’	1レジスタ呼出=2バイト
データ(H)	4	ポジション番号 ステータス	4.3.2(10)一覧表参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	15		

(4) 使用例

軸No.0コントローラの位置決め完了ポジション(アドレス9014_H)を読み取る使用例を示します

クエリ:01 03 90 14 00 01 57[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
開始アドレス(H)	'9', '0', '1', '4'	39303134
レジスタの数(H)	'0', '0', '0', '1'	30303031
エラーチェック(H)	'5', '7' (LRC計算による)	3537
エンド	'CR', 'LF'	0D0A

レスポンスは以下ようになります。

レスポンス:01 03 02 00 00 FA[CR][LF]

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '3'	3033
データバイト数(H)	'0', '2' (2バイト= 1レジスタ)	3032
データ1(H)	'0', '0', '0', '0'	30303030
エラーチェック(H)	'F', 'A' (LRC計算による)	4641
エンド	'CR', 'LF'	0D0A

※レスポンス例のデータは一例であり、各種状態により異なります。

6.5 動作指令 及び、データ書き換え(コード 05 使用ケリ)

6.5.1 コイルへの書込み

※) [6.2 ASCIIコード表](#)
を参照ください。

(1) 機能

スレーブのDO(Discrete Output)の状態をON/OFFのいずれかに変更(書込み)します。プログラムキャストの場合には、全スレーブの同じアドレスのコイルを書換えます。

(2) 開始アドレス一覧

開始アドレス(H)	記号	機能名称
0401	SFTY	セーフティ速度指令
0403	SON	サーボ ON 指令
0407	ALRS	アラームリセット指令
0408	BKRL	ブレーキ強制解除指令
040A	STP	一時停止指令
040B	HOME	原点復帰指令
040C	CSTR	位置決め動作起動指令
0411	JISL	ジョグ/インテグ切替え
0414	MOD	ティーチモード指令
0415	TEAC	ポジションデータ取込み指令
0416	JOG+	ジョグ+指令
0417	JOG-	ジョグ-指令
0418	ST7	スタートポジション 7(電磁弁モード)
0419	ST6	スタートポジション 6(電磁弁モード)
041A	ST5	スタートポジション 5(電磁弁モード)
041B	ST4	スタートポジション 4(電磁弁モード)
041C	ST3	スタートポジション 3(電磁弁モード)
041D	ST2	スタートポジション 2(電磁弁モード)
041E	ST1	スタートポジション 1(電磁弁モード)
041F	ST0	スタートポジション 0(電磁弁モード)
0427	PMSL	PIO/Modbus 切替え指定
042C	STOP	減速停止

6.5.2 セーフティ速度有効/無効切替《SFTY》

(1) 機能

ユーザパラメータNo.35の「セーフティ速度」で指定された速度の有効/無効切替を行います。
MANUモード時に有効にしますと、全ての移動指令速度が制限されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘1’	セーフティ速度指令
変更データ(H)	4	任意	セーフティ速度有効: ‘F’, ‘F’, ‘0’, ‘0’ セーフティ速度無効: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。
不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのセーフティ速度を有効にする使用例を示します

クエリ:01 05 04 01 FF 00 DC CA

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '0', '1'	30343031
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'D', 'C' (LRC計算による)	4443
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.3 サーボ ON/OFF 《SON》

(1) 機能

サーボ ON/OFF の制御を行います。

変更データをサーボ ON 状態にするとメーカーパラメータの「サーボオン遅延時間」経過後にサーボ ON 状態へと遷移します。ただし下記条件を満たしている必要があります。

- ・デバイスステータスレジスタ 1 の EMG ステータスビットが 0
- ・デバイスステータスレジスタ 1 の重故障ステータスビットが 0
- ・デバイスステータスレジスタ 2 のイネーブルステータスビットが 1
- ・システムステータスレジスタの自動サーボ OFF 中ステータスビットが 0

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘3’	サーボON/OFF指令
変更データ(H)	4	任意	サーボON: ‘F’, ‘F’, ‘0’, ‘0’ サーボOFF: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※上位との通信前に、テーチングボックス、パソコンソフトを接続し、サーボ OFF 動作をした後に、非接続にした場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIO ポートの接続を外す際はサーボ ON の状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのサーボをONする使用例を示します

クエリ:01 05 04 03 FF 00 F4

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '0', '3'	30343033
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'F', '4' (LRC計算による)	4634
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.4 アラームリセット《ALRS》

(1) 機能

アラームリセットのエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、アラームリセットを行います。

ただしアラーム要因が解消されていないと再びアラームとなります。

また一時停止中にアラームリセットのエッジを立てると残移動量のキャンセルが行われます。

アラームリセットを行ったら 必ず変更データを0000_Hにして書き込みを行い、通常の状態に戻してください。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘7’	アラームリセット指令
変更データ(H)	4	任意	アラームリセット実行: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるかもしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのアラームをリセットする使用例を示します

- 1回目 01 05 04 07 FF 00 F0 ...アラームリセット実行
 2回目 01 05 04 07 00 00 EF ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘0’, ‘7’	30343037
変更データ(H)	1回目: ‘F’, ‘F’, ‘0’, ‘0’ 2回目: ‘0’, ‘0’, ‘0’, ‘0’ (アラームリセット終了後に0000 _H を書き込んで 通常状態に戻してください)	46463030 30303030
エラーチェック(H)	1回目: ‘F’, ‘0’ (LRC計算による) 2回目: ‘E’, ‘F’ (LRC計算による)	4630 4546
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.5 ブレーキ強制解除《BKRL》

(1) 機能

ブレーキの制御はサーボON/OFFと連動して行われますが、ブレーキがONの状態でも強制的に解除することができます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘8’	ブレーキ強制解除指令
変更データ(H)	4	任意	ブレーキ強制解除: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※上位との通信前に、テイチングボックス、パソコンソフトを接続し、サーボOFF動作をした後に、非接続にした場合、上位との通信でサーボON/OFFが出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのブレーキを解除する使用例を示します

クエリ:01 05 04 08 FF 00 F0

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '0', '8'	30343038
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'F', '0' (LRC計算による)	4630
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.6 一時停止《STP》

(1) 機能

移動中に一時停止指令を行うと減速停止を行い、再び通常状態にセットされると残移動量の移動を再開します。

一時停止指令の状態では、モータの移動は全て禁止されます。

一時停止指令中にアラームリセット指令ビットが立てられた場合は残移動量がキャンセルされます。

一時停止指令の入力が原点復帰動作中で、押付け反転前ならば移動指令が保留され、押付け反転後では原点復帰を最初からやり直します。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H)
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘A’	一次停止指令
変更データ(H)	4	任意	一時停止指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるかもしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを一時停止する使用例を示します

クエリ:01 05 04 0A FF 00 ED

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '0', 'A'	30343041
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'E', 'D' (LRC計算による)	4544
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.7 原点復帰《HOME》

(1) 機能

原点復帰指令のエッジを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、原点復帰動作を行います。原点復帰が完了するとHENDビットが1になります。

原点復帰指令は、原点復帰が完了していても入力可能です。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘B’	原点復帰指令
変更データ(H)	4	任意	原点復帰実行: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※上位との通信前に、テーチングボックス、パソコンソフトを接続し、サーボ OFF 動作をした後に、非接続にした場合、上位との通信でサーボ ON/OFF が出来なくなります。

このため、コントローラ電源を再投入するか、もしくは、SIOポートの接続を外す際はサーボONの状態にして外してください。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0 コントローラを原点復帰する使用例を示します

クエリ

1回目 01 05 04 0B 00 00 EC ...通常状態に設定

2回目 01 05 04 0B FF 00 EB ...原点復帰実行

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '0', 'B'	30343042
変更データ(H)	1回目: '0', '0', '0', '0' 2回目: 'F', 'F', '0', '0' (エッジを立てる為に2回データを送信してください。)	30303030 46463030
エラーチェック(H)	1回目: 'E', 'C' (LRC計算による) 2回目: 'E', 'B' (LRC計算による)	4543 4542
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.8 位置決め動作起動指令《CSTR》

(1) 機能

位置決め動作起動指令のレジスタを立てる(変更データが0000_Hの状態ではFF00_Hを書き込む)と、ポジション番号指定レジスタ(POSR)内のポジション番号の指定位置に移動します。ポジションスタート指令状態のまま(FF00_Hを書き込んだまま)ですと位置決め幅内に入っても完了ポジションは出力されません。(0000_Hを書き込んで通常状態に戻してください。)

電源投入後、一度も原点復帰動作を行っていない状態(HENDビットが0の状態)では、原点復帰動作を実行した後に目標位置に移動を開始します。

※目標位置 及び速度等の動作パラメータは、全てコントローラ内部のポジションテーブル(不揮発性メモリ)に予め設定しておく必要があります。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘0’, ‘C’	位置決め動作起動指令
変更データ(H)	4	任意	ポジションスタート指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラのポジション番号指定レジスタ(POSR)内のポジション番号の指定位置に移動を行う使用例を示します。

クエリ

1回目 01 05 04 0C FF 00 EB ...移動指令

2回目 01 05 04 0C 00 00 EA ...通常状態

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘0’, ‘C’	30343043
変更データ(H)	1回目: ‘F’, ‘F’, ‘0’, ‘0’ 2回目: ‘0’, ‘0’, ‘0’, ‘0’ (通常状態に戻してください。)	46463030 30303030
エラーチェック(H)	1回目: ‘E’, ‘B’ (LRC計算による) 2回目: ‘E’, ‘A’ (LRC計算による)	4542 4541
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.9 ジョグ/インチング切替《JISL》

(1) 機能

ジョグとインチングの切替を行います。

ジョグ動作中に本ビットが切替わると減速停止します。

インチング動作中に本ビットが切替わってもインチング動作は継続されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘1’, ‘1’	ジョグ/インチング切替
変更データ(H)	4	任意	インチング動作状態: ‘F’, ‘F’, ‘0’, ‘0’ ジョグ動作状態: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか

もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをインテグ動作に切替ます。

クエリ:01 05 04 11 FF 00 E6

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '1', '1'	30343131
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'E', '6' (LRC計算による)	4536
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.10 ティーチモード指令《MOD》

(1) 機能

通常運転モードと教示モードを切替ます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘1’, ‘4’	通常モード⇔教示モード切替
変更データ(H)	4	任意	教示モード: ‘F’, ‘F’, ‘0’, ‘0’ 通常運転モード: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラを教示モードに切替ます。

クエリ:01 05 04 14 FF 00 E3

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '1', '4'	30343134
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'E', '3' (LRC計算による)	4533
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.11 ポジションデータ取込み指令《TEAC》

(1)機能

6.5.10 ティーチモード指令が FF00_H(教示モード)の時に、本指令(FF00_Hを書き込み)で現在位置データの取込みを行います。

取込み場所は、ポジション番号指定レジスタで指定されているポジション番号の中です。取込みポジションが空のポジションの場合、目標位置以外のデータ(位置決め幅 INP,速度 VCMD,加減速度 ACMD,制御フラグ CTLF)はパラメータの初期値と一緒に書込まれます。

本指令(FF00_Hを書き込み)を行って20msec以上そのままの状態を保ってください。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘1’, ‘5’	ポジションデータ取込み指令
変更データ(H)	4	任意	ポジションデータ取込み指令 : ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラが教示モード時に現在位置を取込みます。

クエリ:01 05 04 15 FF 00 E2

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '1', '5'	30343135
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'E', '2' (LRC計算による)	4532
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.12 ジョグ+指令《JOG+》

(1) 機能

ジョグ またはインチング動作を行います。

・6.5.9 ジョグ／インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ+指令(変更データ FF00_H)を送信すると、反原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 のPIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ+指令(変更データ 0000_H)を送信するか、6.5.13 ジョグ-指令(変更データ FF00_H)を送信すると減速停止します。

・6.5.9 ジョグ／インチング切替指令が FF00_H(インチング設定)の時、ジョグ+指令のエッジを立てる(変更データが0000_Hの状態にFF00_Hを書き込む)と、反原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘1’, ‘6’	ジョグ+指令
変更データ(H)	4	任意	ジョグ+指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ:01 05 04 16 FF 00 E1

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレープアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘1’, ‘6’	30343136
変更データ(H)	‘F’, ‘F’, ‘0’, ‘0’	46463030
エラーチェック(H)	‘E’, ‘1’ (LRC計算による)	4531
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ

1回目 01 05 04 16 FF 00 E1 ...インチング移動

2回目 01 05 04 16 00 00 E0 ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレープアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘1’, ‘6’	30343046
変更データ(H)	1回目: ‘F’, ‘F’, ‘0’, ‘0’ 2回目: ‘0’, ‘0’, ‘0’, ‘0’ (通常状態に戻してください。)	46463030 30303030
エラーチェック(H)	1回目: ‘E’, ‘1’ (LRC計算による) 2回目: ‘E’, ‘0’ (LRC計算による)	4531 4530
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.13 ジョグ-指令《JOG-》

(1) 機能

ジョグ またはインチング動作を行います。

・6.5.9 ジョグ／インチング切替指令が 0000_H(ジョグ設定)の時、ジョグ-指令(変更データ FF00_H)を送信すると、原点方向にジョグ移動します。速度および加減速度はユーザパラメータNo.26 のPIO ジョグ速度と定格加減速度が使用されます。

ジョグ移動中にジョグ-指令(変更データ 0000_H)を送信するか、6.5.12 ジョグ+指令(変更データ FF00_H)を送信すると減速停止します。

・6.5.9 ジョグ／インチング切替指令が FF00_H(インチング設定)の時、ジョグ-指令のエッジを立てる(変更データが0000_Hの状態がFF00_Hを書き込む)と、原点方向にインチング移動します。速度・移動距離・加減速度はそれぞれユーザパラメータNo.26(PIOジョグ速度)、ユーザパラメータNo.48(PIOインチング距離)、定格加減速度が使用されます。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘1’, ‘7’	ジョグ指令
変更データ(H)	4	任意	ジョグ-指令: ‘F’, ‘F’, ‘0’, ‘0’ 通常: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか もしくは、レスポンスは返信されません。

(4) 使用例

①軸No.0コントローラをジョグ移動させます。

クエリ:01 05 04 17 FF 00 E0

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘1’, ‘7’	30343137
変更データ(H)	‘F’, ‘F’, ‘0’, ‘0’	46463030
エラーチェック(H)	‘E’, ‘0’ (LRC計算による)	4530
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

②軸No.0コントローラをインチング移動させます。

クエリ

1回目 01 05 04 17 FF 00 E0 ...インチング移動

2回目 01 05 04 17 00 00 DF ...通常状態に戻す

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘1’, ‘7’	30343047
変更データ(H)	1回目: ‘F’, ‘F’, ‘0’, ‘0’ 2回目: ‘0’, ‘0’, ‘0’, ‘0’ (通常状態に戻してください。)	46463030 30303030
エラーチェック(H)	1回目: ‘E’, ‘0’ (LRC計算による) 2回目: ‘D’, ‘F’ (LRC計算による)	4530 4446
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.5.14 スタートポジション0～7《ST0～ST7》(PIOパターン4、5 限定)

(1) 機能

指定されたポジションNo.位置に移動します。

スタートポジション0～7 移動指令は、PIOパターン4、5(電磁弁モード)が選択されている時に使用できます。

移動指令は、6.5.14(5)開始アドレス内の ST0～ST7 のどれかを有効にする(0000_Hの状態 で FF00_Hを書き込む)ことで行います。

有効スタートポジション以外を選択するとアラーム「085 移動時ポジションNo.異常」が発生します。ユーザーパラメータNo.27移動指令種別によりレベル動作とエッジ動作が選択可能です。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ～10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,D0への1点書き込み
開始アドレス(H)	4	任意	6.5.14(5)開始アドレス参照
変更データ(H)	4	任意	※1動作指令: ‘F’, ‘F’, ‘0’, ‘0’ 動作指令: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※1 ユーザパラメータNo.27 移動指令種別を『レベル動作』設定した場合
FF00_H→0000_H書き込みで減速停止します。

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか、もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラをスタートポジション2へ移動します。

スタートポジション設定例

No	位置 [mm]	速度 [mm/s]	加速度 [G]	減速度 [G]
0	0.00	533.00	0.30	0.30
1	25.00	533.00	0.30	0.30
2	50.00	533.00	0.30	0.30

図6.2

クエリ

1回目 01 05 04 1D 00 00 D9...エッジを立てる為、0000_H書込み

2回目 01 05 04 1D FF 00 DA...移動指令

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘1’, ‘D’	30343044
変更データ(H)	1回目: ‘0’, ‘0’, ‘0’, ‘0’ 2回目: ‘F’, ‘F’, ‘0’, ‘0’	30303030 46463030
エラーチェック(H)	1回目: ‘D’, ‘9’ (LRC計算による) 2回目: ‘D’, ‘A’ (LRC計算による)	4439 4441
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

(5) 【開始アドレス】

アドレス	記号	名称	機能
0418	ST7	スタートポジション 7	ポジション 7 へ移動します
0419	ST6	スタートポジション 6	ポジション 6 へ移動します
041A	ST5	スタートポジション 5	ポジション 5 へ移動します
041B	ST4	スタートポジション 4	ポジション 4 へ移動します
041C	ST3	スタートポジション 3	ポジション 3 へ移動します
041D	ST2	スタートポジション 2	ポジション 2 へ移動します
041E	ST1	スタートポジション 1	ポジション 1 へ移動します
041F	ST0	スタートポジション 0	ポジション 0 へ移動します

6. 5. 15 PIO/Modbus 切替設定 《PMSL》

(1) 機能

PIO外部指令信号の有効/無効の切替を行うことができます。

(2) ケリフォーマット

フィールド名称	文字数	ASCIIモード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘2’, ‘7’	PIO/Modbus切替設定
変更データ(H)	4	任意	※1Modbus指令有効: ‘F’, ‘F’, ‘0’, ‘0’ Modbus指令無効: ‘0’, ‘0’, ‘0’, ‘0’
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

※1 ・Modbus 指令有効(ON)(PIO 指令無効): FF00_H

PIO 信号による運転はできません。

・Modbus 指令無効(OFF)(PIO 指令有効): 0000_H

外部からの PIO 信号による運転が可能です。

補足 Modbus 指令を有効に変更した場合、変更時の PIO 状態が保持されています。
Modbus指令を無効に切替えた場合、現在のPIO状態により運転状態が変化します。但し、その時にエッジ検出で動作をする信号の状態が変化していてもエッジを検出したことにしないようにしています。

(3) 注意事項

- 運転モードスイッチ搭載機種では AUTO モードに変更されたら『PIO 指令有効』に
MANU モードに変更されたら『PIO 指令無効』になります。
- PIO 未搭載機種ではデフォルト設定が『PIO 指令無効』となります。
- 弊社ツール接続時(ティーチングペンダント、パソコン対応ソフト)は、ツール内モードには『ティーチモード 1, 2』、『モニターモード 1, 2』が存在します。この場合
『モニターモード 1, 2』 → 『PIO 指令有効』
『ティーチモード 1, 2』 → 『PIO 指令無効』 となります。

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるかもしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをModbus指令有効にします。

クエリ:01 05 04 27 FF 00 D0

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘5’	3035
開始アドレス(H)	‘0’, ‘4’, ‘2’, ‘7’	30343237
変更データ(H)	‘F’, ‘F’, ‘0’, ‘0’	46463030
エラーチェック(H)	‘D’, ‘0’ (LRC計算による)	4430
エンド	‘CR’, ‘LF’	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6. 5. 16 減速停止《STOP》

(1) 機能

減速停止指令のエッジを立てる(FF00_Hを書込む)と、減速停止します。

(2) クエリフォーマット

フィールド名称	文字数	ASCIIコード 文字列(固定部)	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘0’, ‘5’	コイル,DOへの1点書き込み
開始アドレス(H)	4	‘0’, ‘4’, ‘2’, ‘C’	減速停止設定
変更データ(H)	4	任意	減速停止指令(ON): ‘F’, ‘F’, ‘0’, ‘0’ ※コントローラが自動的に0000 _H にリセット します。
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	17		

(3) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7 項参照)が返信されるか
もしくは、レスポンスは返信されません。

(4) 使用例

軸No.0コントローラに減速停止指令を出します。

クエリ:01 05 04 2C FF 00 CB

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	':'	3A
スレーブアドレス(H)	'0', '1'	3031
ファンクションコード(H)	'0', '5'	3035
開始アドレス(H)	'0', '4', '2', 'C'	30343243
変更データ(H)	'F', 'F', '0', '0'	46463030
エラーチェック(H)	'C', 'B' (LRC計算による)	4342
エンド	'CR', 'LF'	0D0A

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.6 制御情報の直接書き込み(コード 06 使用ケリ)

6.6.1 レジスタへの書き込み

※)6.2 [ASCIIコード表](#)
を参照ください。

(1) 機能

スレーブのレジスタの内容を変更(書き込み)します。

ブロードキャストの場合には、全スレーブの同じアドレスのレジスタの内容が変更されます。

「4.3.2 (1) デバイス制御レジスタ1 内容」をご参照ください。

または「4.3.2 (2) デバイス制御レジスタ2 内容」をご参照ください。

または「4.3.2 (3) ポジション番号指定レジスタ内容」をご参照ください。

(2) 開始アドレス一覧

アドレス	記号	名称	バイト
0D00	DRG1	デバイス制御レジスタ1	2
0D01	DRG2	デバイス制御レジスタ2	2
0D03	POSR	ポジション番号指定レジスタ	2

上記は制御指令のレジスタです。本レジスタのビットは、「PIO/Modbus 切替えステータス (PMSS)(6.4.8 参照)」が Modbus 指令無効(PIO 指令有効)の時、PIO パターンにより入力ポートに割り当てられます。本レジスタは Modbus 指令有効(PIO 指令無効)の時、書換えが可能です。

(3) クエリフォーマット

クエリメッセージでは、変更するレジスタのアドレスとデータを指定します。

変更したいデータは、クエリのデータエリアで16ビットのデータとして指定します。

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	':'	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	'0', '6'	レジスタへの書込み
開始アドレス(H)	4	任意	6.6.1(2)開始アドレス一覧参照
変更データ(H)	4		4.3.2(1)~4.3.2(3) 変更データ 一覧参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	'CR', 'LF'	
合計バイト数	17		

(4) レスポンス

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

不正なデータを送信した場合に関しては、例外レスポンス(7項参照)が返信されるか、もしくは、レスポンスは返信されません。

(5) 使用例

軸No.0コントローラをサーボON→原点復帰させます。

クエリ

1回目 01 06 0D 00 10 00 DC...サーボON

2回目 01 06 0D 00 10 10 CC...原点復帰

フィールド名称	ASCIIモード 8ビットデータ	ASCIIコード 変換データ(H)
スタート	‘:’	3A
スレーブアドレス(H)	‘0’, ‘1’	3031
ファンクションコード(H)	‘0’, ‘6’	3036
開始アドレス(H)	‘0’, ‘D’, ‘0’, ‘0’	30443030
変更データ(H)	1回目: ‘1’, ‘0’, ‘0’, ‘0’ 2回目: ‘1’, ‘0’, ‘1’, ‘0’	31303030 31303130
エラーチェック(H)	1回目: ‘D’, ‘C’ (LRC計算による) 2回目: ‘C’, ‘C’ (LRC計算による)	4443 4343
エンド	‘CR’, ‘LF’	0D0A

※ ①サーボOFFの状態から 変更データを1010_Hとして送信しても原点復帰は行われません。
(各RCコントローラの起動時のタイミングチャート参照ください。)

②前の状態を維持したい場合は、変更がなくても前の状態を送信してください。
使用例のようにサーボONのビットは 原点復帰時も1のままにしておきます。

正常に変更された場合のレスポンスメッセージは、クエリと同じになります。

6.7 位置決めデータ直接書き込み(コード10使用ケリ)

6.7.1 直値移動指令

※) [6.2 ASCIIコード表](#)を参照ください。

(1) 機能

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。アドレス 9900_H~9908_Hのレジスタ群を書換えるとアクチュエータに対し直値移動指令が行えます。(一伝文で送信が可能です。)

制御フラグ指定レジスタ(アドレス:9908_H)以外のレジスタは電源投入後、1度送信すればその後は有効となりますので、目標位置、位置決め幅、速度、加減速度、押し付け電流制限値、制御指定の変更の必要がない場合、その後は単独変更による実移動指令可能なレジスタ(開始アドレス一覧参照)の書込みだけで直値移動指令が可能となります。

(2) 開始アドレス一覧

目標位置座標、位置決め幅、速度、加減速度、押し付け電流制限値制御指定フラグ等を数値指定して移動を行うためのレジスタ群です。

開始アドレス一覧のデータ(合計6レジスタ)は、一度の送信で変更することが可能です。

アドレス(H)	記号	名称	符号	単独変更による実移動指令可能	レジスタサイズ	バイトサイズ	単位
9900	PCMD	目標位置指定レジスタ	○	○	2	4	0.01mm
9902	INP	位置決め幅指定レジスタ		×	2	4	0.01mm
9904	VCMD	速度指定レジスタ		○	2	4	0.01mm/sec
9906	ACMD	加減速度指定レジスタ		○	1	2	0.01G
9907	PPOW	押し付け時電流制限指定レジスタ		○	1	2	%
9908	CTLF	制御フラグ指定レジスタ		× 移動後都度初期化	1	2	—

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘1’, ‘0’	直値指令
開始アドレス(H)	4	任意	6.7.1(2)開始アドレス一覧参照
レジスタの数(H)	4	任意	6.7.1(2)開始アドレス一覧参照
バイト数(H)	2	上記レジスタ数による	上記指定レジスタ数の2倍の数値 を入力
変更データ1(H)	4		6.7.1(2)開始アドレス一覧参照
変更データ2(H)	4		6.7.1(2)開始アドレス一覧参照
変更データ3(H)	4		6.7.1(2)開始アドレス一覧参照
:	:		
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	最大256		

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名称	文字数 (バイト数)	ASCIIモード 固定文字列	備考
ヘッダ	1	‘:’	
スレーブアドレス(H)	2	任意	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	2	‘1’, ‘0’	直値指令
開始アドレス(H)	4	任意	6.7.1(2)開始アドレス一覧参照
レジスタの数(H)	4	任意	6.7.1(2)開始アドレス一覧参照
エラーチェック(H)	2	LRC計算結果	
トレーラ	2	‘CR’, ‘LF’	
合計バイト数	最大256		

(5) レジスタ詳細説明

■ 目標位置指定レジスタ(PCMD)

PTP 動作の位置決め目標位置を絶対座標上の位置で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999 です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると移動開始指令時にアラームが発生します。目標位置座標指定レジスタ(記号:PCMD、アドレス:9900_H)の下位ワードが書換えられると移動開始します。つまりこのレジスタに目標位置を書込むだけで直値移動指令が行えます。

■ 位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって2種類の意味を持ちます。1つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、0～999999 です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

本レジスタのみを変更しただけでは移動開始は行われません。

■ 速度指定レジスタ(VCMD)

移動速度を指定します。単位は 0.01mm/sec で設定範囲は 0～999999 です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタの下位ワードが書換えられると移動開始します。つまり移動中の速度可変を行う場合、本レジスタを書換えることにより実現できます。

■加減速度指定レジスタ(ACMD)

加速度・減速度を指定します。単位は 0.01G で設定範囲は 0~300 です。ただし、パラメータの最大加速度および最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

本レジスタが書換えられると移動開始します。つまり移動中の加減速度可変を行う場合、本レジスタを書換えることにより実現できます。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。これらの値は、FF_Hを最大電流とする 256 段階で設定します。押付け電流はアクチュエータの制約により設定可能範囲を 20%~70%※(33_H~B3_H)としてください。範囲はアクチュエータにより異なりますので詳しくは弊社カタログまたは取扱説明書参照してください。

本レジスタが書換えられると移動開始します。つまり押付け動作中の電流制限値可変を行う場合、本レジスタを書換えることにより実現できます。

押付け電流値設定例

※20%設定例

$256 \times 0.2 = 51.2 \rightarrow 33_{\text{H}}$ (16 進数変換)

■制御フラグ指定レジスタ(CTLF)

このパラメータは以下の動作設定をするビットパターンです。

本レジスタは、移動開始後、パラメータ初期値にて上書きされます。

ビット 0 = 0: 固定

ビット 1 = 0: 通常動作

1: 押付け動作

ビット 2 = 0: アプローチ動作完了後の押付け動作の方向を正転とします。

1: アプローチ動作完了後の押付け動作の方向を逆転とします。

このビットにより、PCMDからの最終停止位置の方向を算出しますので方向を間違えると下記図 6.3 のように(2×INP)の幅だけずれた動作になりますので、注意してください。

また、ビット 1 の設定値が 0 の場合はこのビットの設定値は無効です。

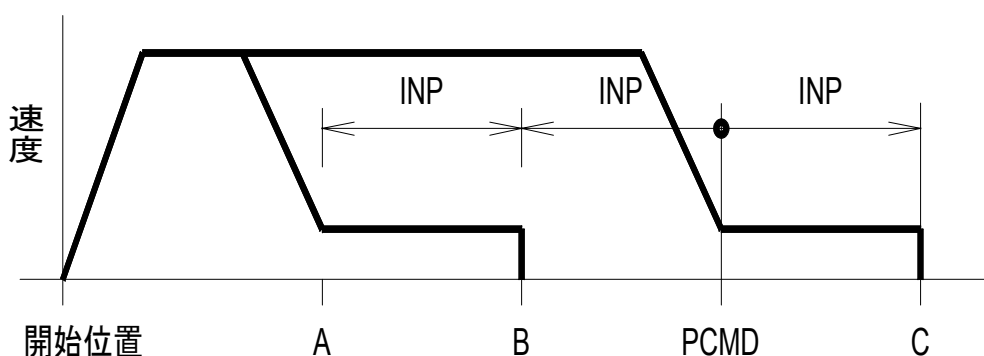


図 6.3 押付け時の動作方向

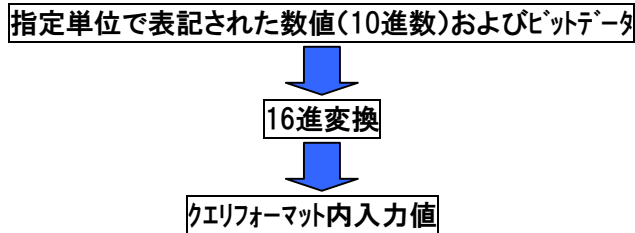
ビット 3 = 0: 通常動作

1: インクリメンタル動作

このビットを 1 に設定することにより、相対位置移動を行う事が可能となります。この動作は、通常動作時と押付け動作時(CTLF のビット 1)で挙動が異なります。通常動作時は目標位置(PCMD)に対しての移動量を生成するのに対し、押付け動作時(ビット 1=1 の時)には現在位置に対しての移動量を生成します。

なお、相対座標の計算は mm 単位加算後パルス変換を行うので、パルス変換後の加算方法の場合に発生する『相対移動を繰り返した場合は、リード設定による割り切れないパルスが累積誤差として位置ずれを起こす現象』は発生しません。

(6) 入力値(変更データ)について



(7) レジスタ入力フロー例

a) 通常動作 1 フロー(目標位置データの変更)

条件: コントローラユーザーパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値
で動作条件はOK。目標位置だけを変更しアクチュエータを動作させたい。

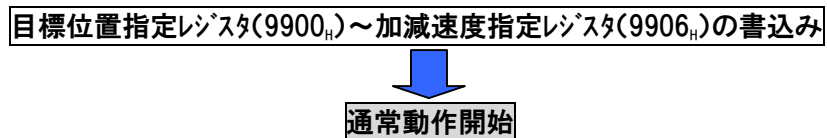
補足: コントローラユーザーパラメータについて

- ・速度初期値(パラメータNo.8)→弊社カタログ内表記アクチュエータ依存最高速度
- ・加減速度初期値(パラメータNo.9)→弊社カタログ内表記アクチュエータ依存
定格加減速度
- ・位置決め幅初期値(パラメータNo.10)→デフォルト値 0.1mm



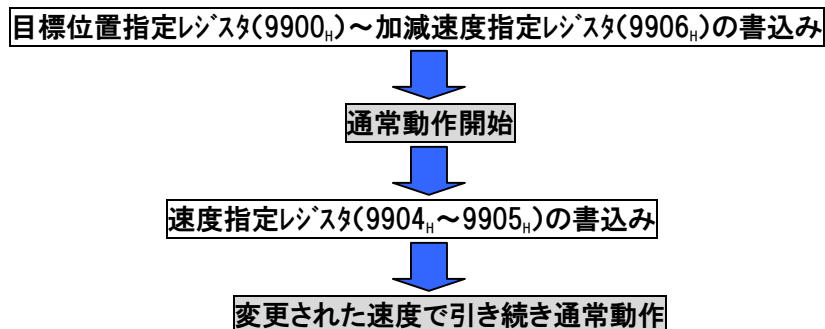
b) 通常動作 2 フロー(位置決めデータの変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させたい



c) 通常動作 3 フロー(移動中の速度変更)

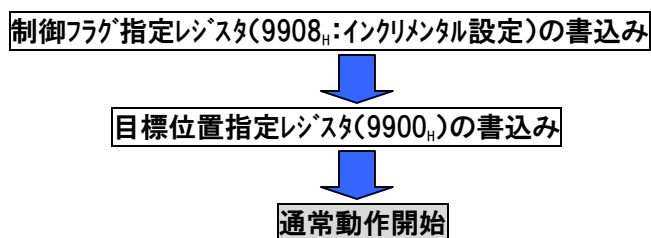
条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい



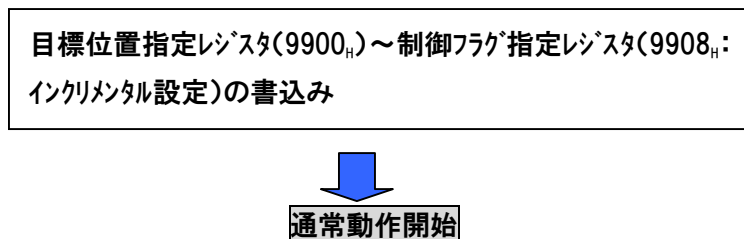
d) 相対位置決め動作 1 (インクリメンタル移動) フロー(ピッチ幅の変更)

条件: コントローラユーザパラメータ上の速度初期値/加減速度初期値/位置決め幅初期値/で動作条件はOK。ピッチ幅だけを変更しアクチュエータを動作させたい。

手法 1



手法 2



補足: アドレス 9900_H と 9908_H だけを一度のデータ送信で変更することは出来ません。アドレスは連番となっているため、9900_H と 9908_H だけを変更したい場合は 2 度の伝文送信で実施し(手法 1)、1 度の伝文送信で済ませたい場合は、9900_H～9908_H までを全て書き込み実行を行う(手法 2)必要があります。手法 2 を用いる場合は速度/加減速度/位置決め幅に対しても書き込みを実行する事になりますのでユーザパラメータNo.8,9,10 の値は無効となります。

e) 相対位置決め動作 2(インクリメンタル移動)フロー(インクリメンタル移動中の速度変更)

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを相対位置決め動作させ、動作中にあるタイミングでアクチュエータ動作速度を変更をしたい

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H:
インクリメンタル設定)の書込み



相対位置決め動作開始



速度指定レジスタ(9904_H)～制御フラグ指定レジスタ(9908_H:インク
リメンタル設定)の書込み



変更された速度で引き続き相対位置決め動作

補足: 制御フラグ指定レジスタ(9908_H:インクリメンタル設定)は変更を行ったあと、一度アクチュエータが動作開始されたとリセットされます。(設定値 0_H:通常移動)このため、速度だけを変更する場合でも制御フラグ指定レジスタ(9908_H)をインクリメンタル設定にし、再度送信する必要があります。

f) 押付け動作フロー(押付け動作中の押付け力の変更)

条件: 押付け動作をさせたい。但し、押し付け中には任意のタイミングで押付け力を変化させたい。

目標位置指定レジスタ(9900_H)～制御フラグ指定レジスタ(9908_H: 押付け設定)の書込み



押付け動作開始



押付け電流制限指定レジスタ(9907_H)～制御フラグ指定レジスタ(9908_H: 押付け設定)の書込み



変更された押付け力で引き続き押付け動作

g) 注意事項(移動中の位置決め幅の変更)

移動中の位置決め幅の変更はできません。

条件: 目標位置/速度/加減速度を都度変更しアクチュエータを動作させ、動作中にあるタイミングで位置決め幅を変更をしたい
(変更できません。書込みを行った場合、このデータは次の位置決めに反映されることとなります。)

目標位置指定レジスタ(9900_H)～加減速度指定レジスタ(9906_H)の書込み



通常動作開始



位置決め幅指定レジスタ(9902_H～9903_H)の書込み



変更前の位置決め幅設定で引き続き通常動作

補足: 位置決め幅指定レジスタは単独書込みによる実移動指令は無効です。このため位置決め幅指定レジスタ(9902_H～9903_H)の書込みによるデータは次回移動指令を実施した際に有効となります。

(8) 使用例

①軸No.0コントローラを通常移動させる使用例を示します

目標位置 (mm)	位置決め幅 (mm)	速度 (mm/sec)	加減速度 (G)	押付け (%)	移動制御
50	0.1	100	0.3	0	通常移動

■クエリ :01 10 9900 0009 12 0000 1388 0000 000A 0000 2710 001E 0000 0000 41[CR][LF]

■レスポンス :01 10 9900 0009 4D[CR][LF]

・・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘9’, ‘9’, ‘0’, ‘0’	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	‘0’, ‘0’, ‘0’, ‘9’	30303039	アドレス9900 _H ~9908 _H まで 書込み指定
バイト数	‘1’, ‘2’	3132	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘1’, ‘3’, ‘8’, ‘8’	31333838	50(mm)×100=5000→1388 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘0’, ‘0’, ‘0’, ‘A’	30303041	0.1(mm)×100=10→000A _H
変更データ5、6 (速度) 入力単位(0.01mm/sec)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘2’, ‘7’, ‘1’, ‘0’	32373130	100(mm/sec)×100=10000 →2710 _H
変更データ7(加減速度) 入力単位(0.01G)	‘0’, ‘0’, ‘1’, ‘E’	30303145	0.3(G)×100=30→001E _H
変更データ8(押付け) 入力単位(%)	‘0’, ‘0’, ‘0’, ‘0’	30303030	0(%)→0 _H
変更データ9(制御フラグ)	‘0’, ‘0’, ‘0’, ‘0’	30303030	通常動作のため全て 0 0000b→0000 _H
エラーチェック	‘4’, ‘1’	3431	LRCチェック計算結果→41 _H
トレーラ	‘CR’, ‘LF’	0D0A	
合計バイト数	55		

②軸No.0コントローラをピッチ送り(相対位置移動)させる使用例を示します

目標位置 (mm)	位置決め幅 (mm)	速度 (mm/sec)	加減速度 (G)	押付け (%)	移動制御
10	0.1	100	0.3	0	インクリメンタル

■クエリ :01 10 9900 0009 12 0000 03E8 0000 000A 0000 2710 001E 0000 0008 E9[CR][LF]

■レスポンス :01 10 9900 0009 4D[CR][LF]

・クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

■クエリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)	備考
ヘッダ	':'	3A	
スレーブアドレス	'0', '1'	3031	軸No.0+1
ファンクションコード	'1', '0'	3130	
開始アドレス	'9', '9', '0', '0'	39393030	開始アドレスは目標位置指定 レジスタ9900 _H から
レジスタの数	'0', '0', '0', '9'	30303039	アドレス9900 _H ~9908 _H まで 書き込み指定
バイト数	'1', '2'	3132	9(レジスタ)×2=18(バイト)→12 _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'0', '3', 'E', '8'	30334538	10(mm)×100=1000→03E8 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'0', '0', '0', 'A'	30303041	0.1(mm)×100=10→000A _H
変更データ5、6 (速度) 入力単位(0.01mm/sec)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'2', '7', '1', '0'	32373130	100(mm/sec)×100=10000 →2710 _H
変更データ7(加減速度) 入力単位(0.01G)	'0', '0', '1', 'E'	30303145	0.3(G)×100=30→001E _H
変更データ8(押付け) 入力単位(%)	'0', '0', '0', '0'	30303030	0(%)→0 _H
変更データ9(制御フラグ)	'0', '0', '0', '8'	30303038	(インクリメンタル設定) 1000b→0008 _H
エラーチェック	'E', '9'	4539	LRCチェック計算結果→E9 _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	55		

6.7.2 ポジションテーブルデータ書込み

(1) 機能

このクエリを使用することによってポジションテーブル上のデータ変更が可能です。

開始アドレス一覧(アドレス+0000_H~+000E_H)にアクセスがある毎に 1 ポジションデータ単位で EEPROM から読出され、書込み実施後、EEPROM に再び格納されます。

※EEPROMはデバイスの制約上、書込み回数が約10万回と制限されています。ポジションテーブルデータの書換えを頻繁に行くと短期間でEEPROMの書換え回数をオーバーし故障の原因となりますので、上位側のロジックは想定外のループ等が発生しないようご注意ください。

(2) 開始アドレス一覧

クエリ入力の際のアドレスは下記の式によって算出します。

$$1000_{\text{H}} + (16 \times \text{ポジション No.})_{\text{H}} + \text{アドレス(オフセット値)}_{\text{H}}$$

例 ポジションNo.200 の速度指令レジスタを変更したい場合

$$\begin{aligned} & 1000_{\text{H}} + (16 \times 200 = 3200)_{\text{H}} + 4_{\text{H}} \\ & = 1000_{\text{H}} + \text{C80}_{\text{H}} + 4_{\text{H}} \\ & = 1\text{C84}_{\text{H}} \end{aligned}$$

『1C84』がクエリ開始アドレス部入力値になります。

※最大ポジション番号は機種及び設定されているPIOパターンにより異なります。

■ポジションデータ変更レジスタ群

アドレス	記号	名称	符号	レジスタ サイズ	バイト サイズ	入力 単位
+0000	PCMD	目標位置	○	2	4	0.01mm
+0002	INP	位置決め幅		2	4	0.01mm
+0004	VCMD	速度指令		2	4	0.01mm/sec
+0006	ZNMP	個別ゾーン境界+側	○	2	4	0.01mm
+0008	ZNLP	個別ゾーン境界-側	○	2	4	0.01mm
+000A	ACMD	加速度指令		1	2	0.01G
+000B	DCMD	減速度指令		1	2	0.01G
+000C	PPOW	押付け時電流制限値		1	2	%
+000D	LPOW	負荷電流閾値		1	2	%
+000E	CTLF	制御フラグ指定		1	2	

※ '+' が付いているアドレスはオフセット値です。

(3) クエリフォーマット

1レジスタ=2バイト=16ビットデータ

フィールド名	ASCIIモード 固定文字列	文字数 (バイト数)	備考
ヘッダ	‘:’	1	
スレーブアドレス(H)	任意	2	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	‘1’, ‘0’	2	
開始アドレス(H)	任意	4	6.7.2(2)開始アドレス参照一覧
レジスタの数(H)	任意	4	6.7.2(2)開始アドレス参照一覧
バイト数(H)	上記レジスタ数に よる	2	上記指定レジスタ数の2倍の数値 を入力
変更データ1(H)		4	6.7.2(2)開始アドレス参照一覧
変更データ2(H)		4	6.7.2(2)開始アドレス参照一覧
変更データ3(H)		4	6.7.2(2)開始アドレス参照一覧
:		:	
エラーチェック(H)	LRC計算結果	2	
トレーラ	‘CR’, ‘LF’	2	
合計バイト数		最大256	

(4) レスポンスフォーマット

正常に変更された場合のレスポンスメッセージは、クエリメッセージ中のバイト数と変更データを除いた部分のコピーをレスポンスします。

フィールド名	ASCIIモード 固定文字列	文字数 (バイト数)	備考
ヘッダ	‘:’	1	
スレーブアドレス(H)	任意	2	軸No.+1(01 _H ~10 _H) ブロードキャスト指定時は00 _H
ファンクションコード(H)	‘1’, ‘0’	2	
開始アドレス(H)	任意	4	6.7.2(2)開始アドレス参照一覧
レジスタの数(H)	任意	4	6.7.2(2)開始アドレス参照一覧
エラーチェック(H)	LRC計算結果	2	
トレーラ	‘CR’, ‘LF’	2	
合計バイト数		最大256	

(5) レジスタ詳細説明

■目標位置(PCMD)

ポジション移動時の位置決め目標位置を絶対座標上の位置、または相対距離で指定します。単位は 0.01mm で、設定可能範囲は、-999999～999999 です。絶対座標指定時はパラメータのソフトリミットを超えた値が設定されていると移動開始指令時にアラームが発生します。絶対位置座標か相対距離の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■位置決め幅指定レジスタ(INP)

本レジスタは動作種別によって 2 種類の意味を持ちます。1 つ目の意味は通常位置決め動作の場合、位置決め時の動作完了検出に用いる目標位置と現在位置の差の許容値となります。2 つ目の意味は押付け動作時の押付け幅となります。単位は 0.01mm で、設定可能範囲は、0～999999 です。動作種別の指定は、後述する制御フラグ指定レジスタのビットで指定します。

■速度指定レジスタ(VCMD)

ポジション移動時の、移動速度を指定します。単位は 0.01mm/sec で設定範囲は 0～999999 です。ただし、パラメータの最大速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■個別ゾーン境界±(ZNMP,ZNLP)

パラメータで設定されるゾーン境界とは別に、ポジション移動時のみ有効なゾーン信号を出力します。

絶対位置座標で表現した+側のゾーン信号出力境界値を ZNMP に、-側のゾーン信号出力境界値を ZNLP に設定します。現在位置がこの±境界値の内側にあるときは、ゾーンステータスレジスタの対応するビットが ON となります。設定単位は 0.01mm となります。設定可能範囲はともに -999999～999999 となりますが ZNMP > ZNLP の関係が満足している必要があります。

個別ゾーン出力を無効にする場合は、ZNMP と ZNLP を同じ値としてください。

■加速度指定レジスタ(ACMD)

ポジション移動時の、加速度を指定します。単位は 0.01G で設定範囲は 0～300 です。ただし、パラメータの最大加速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■減速度指定レジスタ(DCMD)

ポジション移動時の、減速度を指定します。単位は 0.01G で設定範囲は 0~300 です。ただし、パラメータの最大減速度を超えた値が設定されていると移動開始指令時にアラームが発生します。

■押付け時電流制限値(PPOW)

押付け動作時電流制限を PPOW に設定します。この値は、 FF_H を最大電流とする 256 段階で設定します。押付け電流はアクチュエータの制約により設定可能範囲を 20%~70%($33_H \sim B3_H$)としてください。

■負荷出力電流閾値(LPOW)

負荷出力判定を行う場合、電流閾値を LPOW に設定します。この値は、 FF_H を最大電流とする 256 段階で設定します。判定を行わない場合は 0 を設定してください。

■制御フラグ指定レジスタ(CTLF)

このパラメータは以下の動作設定をするビットパターンです。

ビット 1 = 0:通常動作

1:押付け動作

ビット 2 = 0:アプローチ動作完了後の押付け動作の方向を正転とします。

1:アプローチ動作完了後の押付け動作の方向を逆転とします。

ビット 3 = 0:通常動作

1:インクリメンタル動作

※未記載のビットは全て未使用となっています。

(6) 入力値(データ変更)について

指定単位で表記された数値(10進数)およびビットデータ

↓
16進変換

↓
クエリフォーマット内入力値

次頁に上記レジスタの仕様を記述します。

(7) 使用例

軸No.0 のポジションNo.12 全データを下記のように書き換えます。

目標位置 (mm)	位置 決め幅 (mm)	速度 (mm/sec)	個別ゾーン 境界+側 (mm)	個別ゾーン 境界-側 (mm)	加速度 (G)	減速度 (G)	押付け (%)	閾値	移動制御
100	0.1	200	60	40	0.01	0.3	0	0	通常移動

■ケリ:01 10 10C0 000F 1E 0000 2710 0000 000A 0000 4E20 0000 1770

0000 0FA0 0001 001E 0000 0000 0000 EE[CR][LF]

■受信レスポンス :01 10 10C0 000F 10[CR][LF]

■ケリメッセージ内訳

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)	備考
ヘッダ	‘:’	3A	
スレーブアドレス	‘0’, ‘1’	3031	軸No.0+1
ファンクションコード	‘1’, ‘0’	3130	
開始アドレス	‘1’, ‘0’, ‘C’, ‘0’	31304330	開始アドレスはポジションNo.12における目標位置指定レジスタ10C0 _H から※1
レジスタの数	‘0’, ‘0’, ‘0’, ‘F’	30303046	レジスタ記号PCMD~CTLFまで 合計15レジスタ書込み指定
バイト数	‘1’, ‘E’	3145	15(レジスタ)×2=30(バイト)→1E _H
変更データ1、2 (目標位置) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘2’, ‘7’, ‘1’, ‘0’	32373130	100(mm)×100=10000→2710 _H
変更データ3、4 (位置決め幅) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘0’, ‘0’, ‘0’, ‘A’	30303041	0.1(mm)×100=10→000A _H
変更データ5、6 (速度) 入力単位(0.01mm/SEC)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘4’, ‘E’, ‘2’, ‘0’	34453230	200(mm/sec)×100=20000 →4E20 _H
変更データ7、8 (個別ゾーン境界+) 入力単位(0.01mm)	‘0’, ‘0’, ‘0’, ‘0’	30303030	32ビットデータ上位ビットは全て0
	‘1’, ‘7’, ‘7’, ‘0’	31373730	60(mm)×100=6000→1770 _H
	‘0’, ‘F’, ‘A’, ‘0’	30464130	40(mm)×100=4000→0FA0 _H

次ページに続く

前ページからの続き

フィールド名称	ASCIIモード 固定文字列	ASCIIコード 変換データ(H)	備考
変更データ9、10 (個別ゾーン境界ー) 入力単位(0.01mm)	'0', '0', '0', '0'	30303030	32ビットデータ上位ビットは全て0
	'0', 'F', 'A', '0'	30464130	40(mm)×100=4000→0FA0 _H
変更データ11(加速度) 入力単位(0.01G)	'0', '0', '0', '1'	30303031	0.01(G)×100=1→0001 _H
変更データ12(減速度) 入力単位(0.01G)	'0', '0', '1', 'E'	30303145	0.3(G)×100=30→001E _H
変更データ13(押付け) 入力単位(%)	'0', '0', '0', '0'	30303030	0(%)→0 _H
変更データ14(閾値) 入力単位(%)	'0', '0', '0', '0'	30303030	0(%)→0 _H
変更データ15(制御フラグ)	'0', '0', '0', '0'	30303030	通常動作のため全て 0 0000b→0000 _H
エラーチェック	'E', 'E'	4545	LRCチェック計算結果→EE _H
トレーラ	'CR', 'LF'	0D0A	
合計バイト数	79		

※1) 開始アドレス計算

例 ポジションNo.12 の全データを変更をかける為、本クエリの開始アドレス部には
ポジションNo.12 の目標位置アドレスとなる。

$$1000_{\text{H}} + (16 \times 12 = 192)_{\text{H}} + 0_{\text{H}}$$

$$= 1000_{\text{H}} + \text{C0}_{\text{H}} + 0_{\text{H}}$$

$$= 10\text{C0}_{\text{H}}$$

『10C0』が本クエリ開始アドレス部入力値になります。

下記は弊社 RC 用パソコン対応ソフト上ポジションデータ内で、クエリメッセージ送信前と後の違いを表示したものです

■クエリ送信前

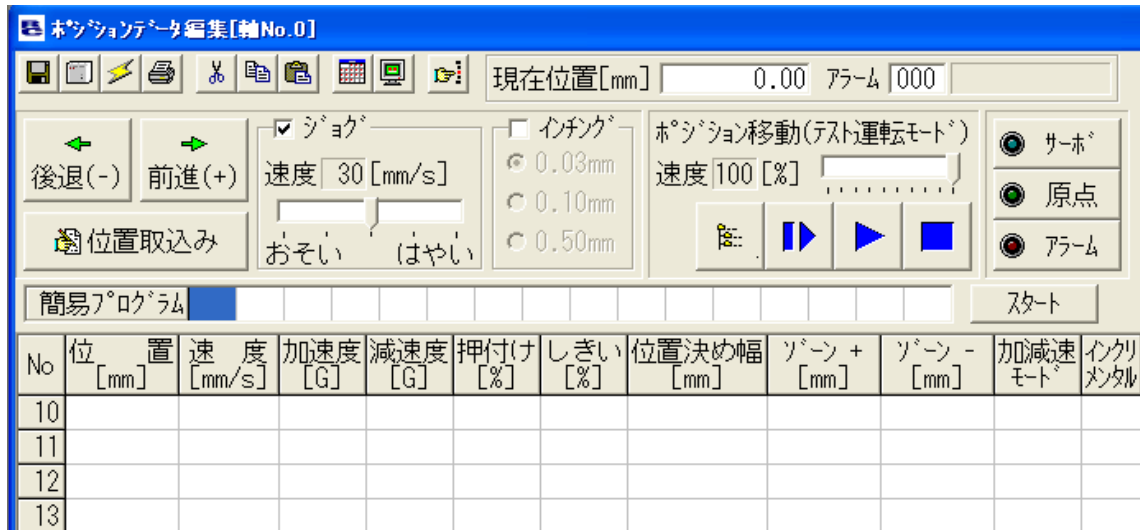


図 6.4

■送信後

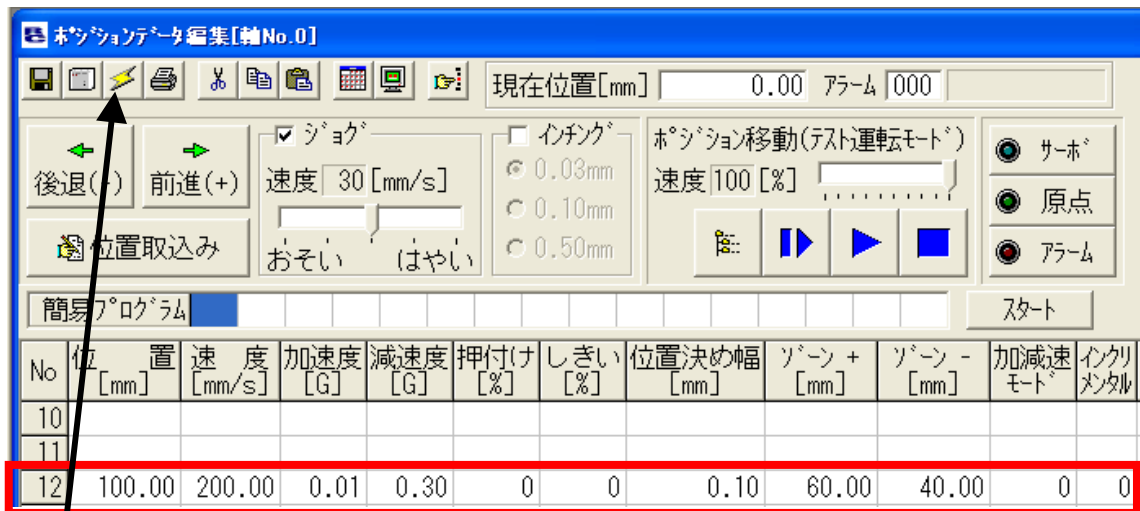



図 6.5

※  ボタンを押すか、ポジションデータ編集画面を開き直さないと書き換えた内容は表示されません。

7 トラブルシューティング



7. 1 異常時の返信(例外レスポンス)について

ブロードキャスト・メッセージ以外のクエリ(命令)の場合、マスターは正常なレスポンス(返信)を期待してクエリを発行します。それに対して、スレーブはレスポンスを返さなければなりません。正常な場合には正常レスポンスを返しますが、異常が発生した場合、例外レスポンスを返します。

クエリに対するスレーブの応答の方法には、以下の4種類があります。

- (1) 正常にクエリを受信し、正常に処理を行い、正常レスポンスを返します。
- (2) 通信エラー等により、スレーブはクエリを受信することができなく、無応答になります。マスターはタイムアウトによるエラー検出を行います。
- (3) スレーブはクエリを受信することができたが、LRC/CRC エラーを検出し、正しいクエリではない場合には、スレーブは無応答になります。従って、マスターはタイムアウトによるエラー検出を行います。
- (4) スレーブは、エラーもなく正しいクエリを受信できたが、何らかの理由(例えば、当該レジスタは存在しないなど)で処理できない場合には、例外の内容を示す例外コードをつけて例外レスポンスを返します。

例外レスポンスが発生する場合の例

(Read Input Status のクエリメッセージ例)

フィールド名称	数値例 (Hex)	ASCIIモード 文字列	RTUモード 8ビット(Hex)
ヘッダ		':'	なし
スレーブアドレス	03 _H	'0', '3'	03 _H
ファンクションコード	02 _H	'0', '2'	02 _H
開始アドレス(上位)	04 _H	'0', '4'	04 _H
開始アドレス(下位)	A1 _H	'A', '1'	A1 _H
DIの数(上位)	00 _H	'0', '0'	00 _H
DIの数(下位)	14 _H	'1', '4'	14 _H
エラーチェック		LRC(2文字)	CRC(16ビット)
トレーラ		CR / LF	なし
	合計バイト数	17	8

入力ステータス04A1_Hが存在しない場合、以下の例外レスポンスが返されます。

スレーブからの例外レスポンス例

フィールド名称	数値例 (Hex)	ASCIIモード 文字列	RTUモード 8ビット(Hex)
ヘッダ		':'	なし
スレーブアドレス	03 _H	'0', '3'	03 _H
ファンクションコード	82 _H	'8', '2'	82 _H
例外コード	02 _H	'0', '2'	02 _H
エラーチェック		LRC(2文字)	CRC(16ビット)
トレーラ		CR / LF	なし
	合計バイト数	11	5

例外レスポンスは、スレーブアドレス、ファンクションコードおよびデータフィールドより構成されています。スレーブアドレスフィールドには、正常レスポンスと同じように、スレーブのアドレスをセットします。ファンクションコードフィールドには、クエリのファンクションコードをセットし更に、そのMSB(ファンクションコードの最上位ビット)を1にします。これによりマスタは、正常レスポンスではなく例外レスポンスであることを検知できます。データフィールドには、例外の内容を示す例外コードがセットされます。

ROBO CYLINDER

例) クエリファンクションコード 『02_H』(00000010_b)
 →例外レスポンスファンクションコード 『82_H』(10000010_b)

■例外コード

RC シリーズコントローラで発生する、例外コード及び内容を示します。

コード (Hex)	例外コード名	機能	備考
01 _H	Illegal Function	不正ファンクション	ファンクションの誤りによりスレーブ側が重度のエラー発生し、クエリの実行ができないとき
02 _H	Illegal Data Address	不正データアドレス	データアドレスが許されない値の時
03 _H	Illegal Data Value	不正データ	データ値が許されない値の時
04 _H	Slave Device Failure	スレーブ回復不能エラー発生 の為、実行不可	スレーブ側の重度エラー発生により、クエリの実行ができないとき

7.2 注意事項

- ・ Modbus ファンクションによってレジスタを参照する場合、1 伝文で複数のカテゴリのレジスタを同時に読出すことは出来ません。従って、カテゴリをまたいで参照する場合は、カテゴリの区切りのアドレスで複数伝文に分けて読み出すようにしてください。
- ・ 本仕様書は RC コントローラシリーズ「プロトコル M」搭載機種共通の説明となっています。機種固有の仕様等は各 RC コントローラの取扱説明書を参照ください。
- ・ 本書は製品の機能向上に伴い、予告無く変更することがありますので、予めご了承ください。

7.3 通信がうまくいかない時は

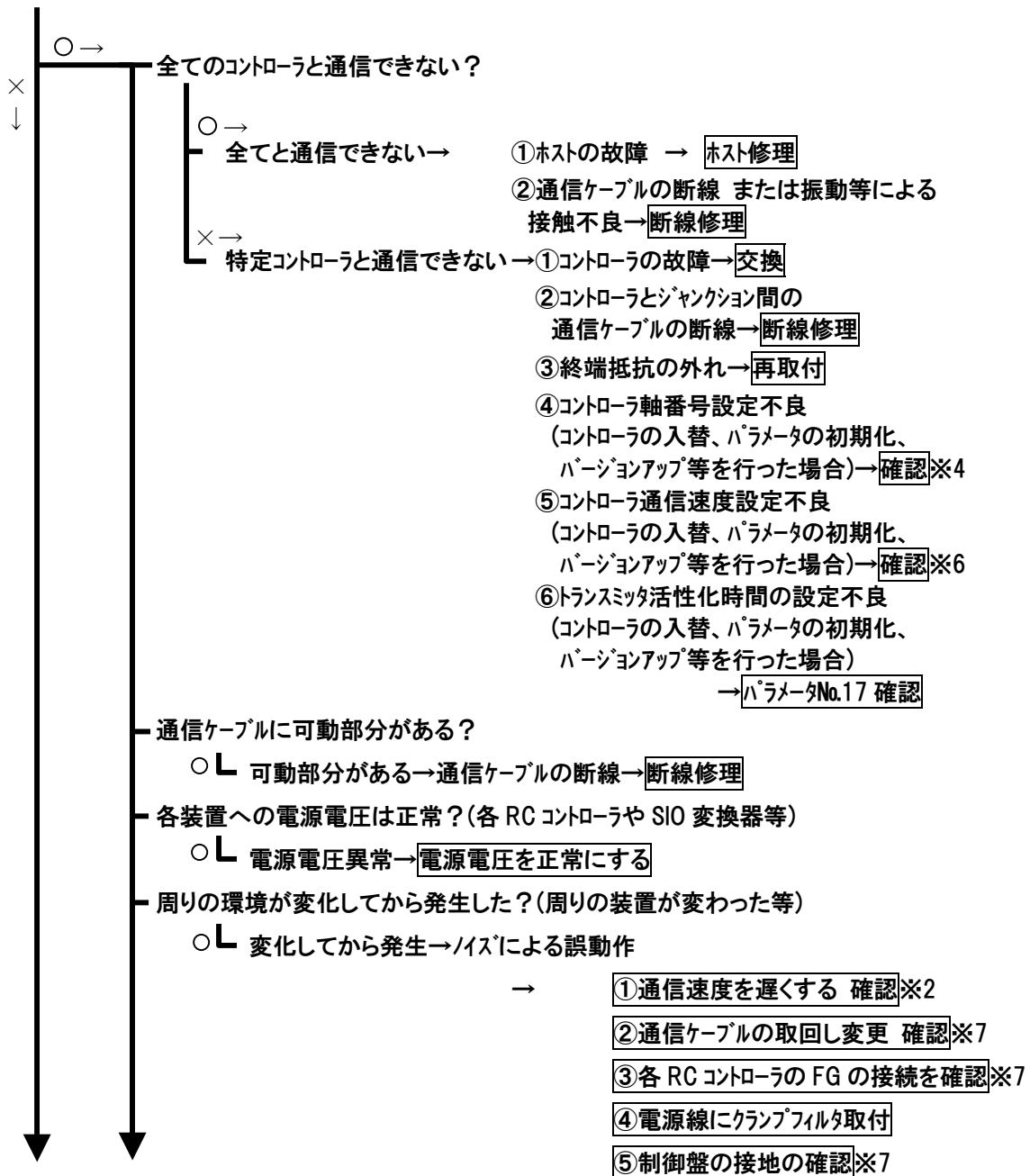
当てはまる項目を選択して □で囲まれた処理をおこなってください。

具体的な処理内容はフローチャートの後に記載していますので、※の示す内容をご確認ください。

○=YES、×=NO

・症状：正常に通信できない！

今まで通信できていた？



次ページに続く

前ページより

移設後に不具合が発生した？

- L 移設後に発生 →
 - ①通信ケーブルの断線 → **断線修理**
 - ②通信ケーブル用コネクタの緩み → **接続確認**
 - ③コントローラ通信速度設定不良 → **3.2.(1)項注意参照、確認※6**

コントローラの追加を行ってから発生した？

- L 追加してから発生 →
 - ①追加コントローラの軸番号設定不良 → **確認※4**
 - ②追加コントローラの通信速度設定不良 → **確認※6**
 - ③追加コントローラのトランスミッタ活性化時間の設定不良 → **パラメータNo.17 確認**
 - ④終端抵抗取付け位置不良 → **上位(ホスト)から最も離れたジャンクションに終端抵抗を取付けてください(3項参照)**

新規装置立上げ？(既存の設計で今回新たに組立した物も含む)

○ ↓ 全てのコントローラと通信できない？

- ↓
 - RC用パソコン対応ソフトを使用して接続の確認※1

- → 全てと通信できない →
 - ①ホスト⇄ジャンクション間の通信ケーブルの配線不良 → **配線確認※3**
 - ②装置の電源異常 → **電源電圧確認**
↓
0Vラインが共通か確認 → **0Vラインを共通化する**
 - ③コントローラの軸番号設定の重複 → **確認※4**
 - ④コントローラの通信速度設定不整合 → **確認※6**

- × → 特定のコントローラと通信できない →
 - ①コントローラとジャンクション間の通信ケーブルの配線不良 → **配線確認※3**
 - ②装置の電源異常 → **電源電圧確認**
 - ③コントローラの軸番号設定の重複 → **確認※4**
 - ④コントローラの通信速度設定不整合 → **確認※6**

- × → 通信できる →
 - ①ホストのプログラム不備 → **①プログラム再確認※5**
 - ②通信速度設定の再確認※2**

次のページへ

前ページより

- Y
- 特定のコントローラと通信できない？
 - L ①コントローラとジャンクション間の通信ケーブルの配線不良→配線確認※3
 - ②装置の電源異常→電源電圧確認→0Vラインが共通になっているか確認
→0Vラインを共通化する
 - 時々通信できない？
 - L ①ノイズによる誤動作→
 - ①通信速度を遅くする 確認※2
 - ②通信ケーブルの取回し変更 確認※7
 - ③各 RC コントローラの FG の接続を確認※7
 - ④電源線にクランプフィルタ取付
 - ⑤制御盤の接地の確認※7
 - ②ホストのプログラム不備→プログラム再確認(通信バッファオーバーフロー等発生?)

※1 3.1 項、3.2 項、3.3 項を参考にホストにパソコンを接続してください。

- ① パソコン対応ソフトを起動してください。
- ② 設定→アプリケーション設定を選択します。

通信設定画面のポートが、使用しているパソコンのポート番号になっているか、最終軸 No.が接続されている軸の数以上に設定されているかを確認してください。

(設定が違っていた場合は、設定してから RC 用パソコン対応ソフトを再起動してください。)

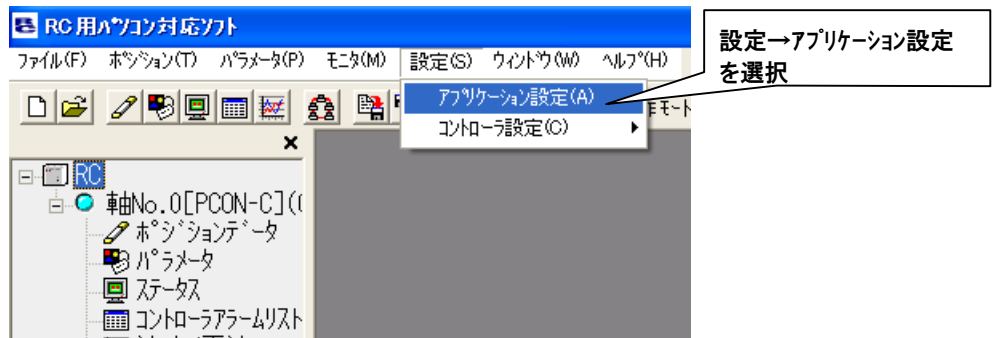


図 9.1

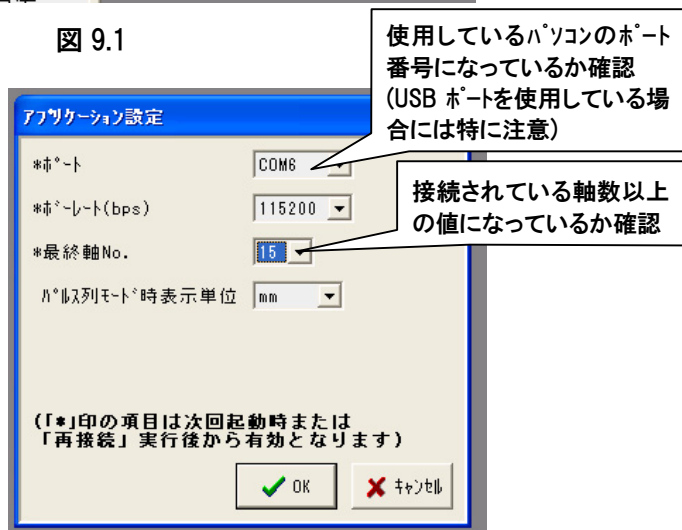


図 9.2

③ポジション→編集/ティーチを選択します。

ポジションデータ編集軸選択画面が表示され、その中に接続されている軸が表示されています。接続されている軸番号が表示されている軸は、通信ができています。

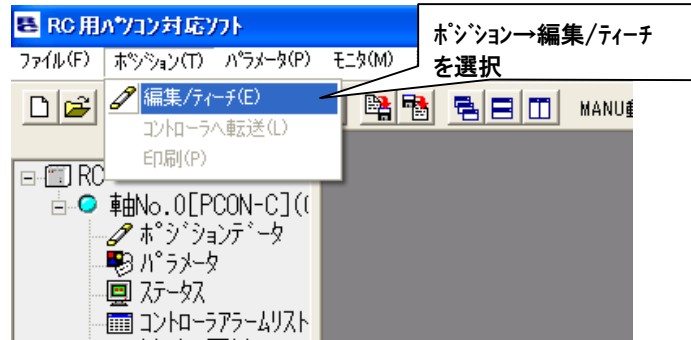


図 9.3

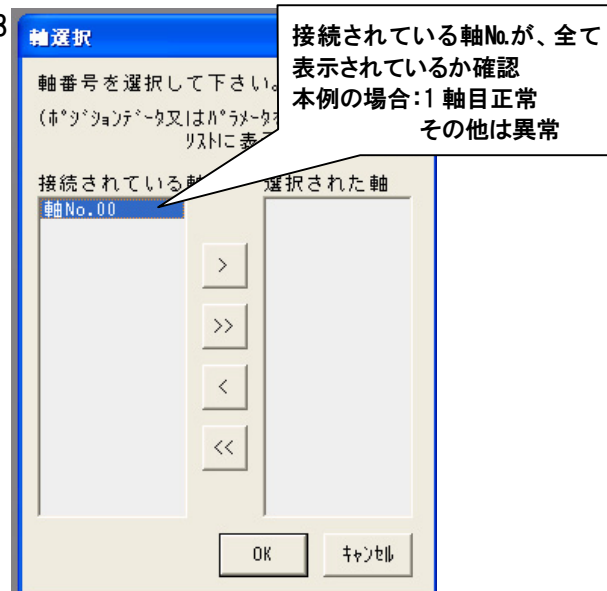


図 9.4

- ※2 3.6 項を参考に通信速度の設定を遅くなるように設定します。
- ※3 3.1 項、3.2 項、3.3 項を参考に配線の再確認をしてください。
- ※4 3.5 項を参考に軸番号の設定を再度確認してください。(重複しないように設定されているか確認)
- ※5 3.4 項の手順が守られているか再度確認してください。
 - ①ファンクションコード 03 のケリ以外を使用している場合は、5.4.15 項(RTU)、6.5.15 項(ASCII)のPIO/Modbus 切替を Modbus 側にしてあるか確認してください。
 - ②RC 用パソコン対応ソフトを使用して、RC コントローラの再起動を行わないと RC 用パソコン対応ソフトを接続した時の通信速度設定のままになっています。その場合には、RC コントローラを再起動してください。

- ※6 3.6 項を参考に通信速度の設定を再確認してください。
全ての RC コントローラ、ホストを同じ通信速度設定にしてください。
※5 の②を確認してください。
- ※7 動力線やパルス信号を送っているような配線と平行にならないように通信ケーブルを配線してください。
通信ケーブルのシールド処理は適切に行っているか確認してください。(推奨:1 点アース)
各 RC コントローラの取扱説明書にある設置環境 及びノイズ対策の内容が施されているかを確認してください。

以上のご確認で改善しない場合、弊社までご相談下さい。
尚、この際には発生している現象 およびフローチャートでの確認事項を併せて、ご連絡くださいますようお願い致します。

お問い合わせ先

アイエイアイお客様センター エイト

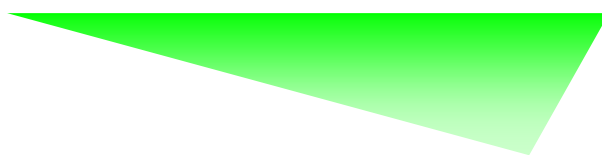
フリーコール 0800-888-0088

FAX(無料) 0800-888-0099

受付時間 月～金 8:00～20:00、土 9:00～17:00

R ROBO C CYLINDER

8 參考資料



8.1 CRC チェック計算

C言語による CRC 値計算用のサンプル関数を示します。

公開されている Modbus プロトコル仕様書(PI-MBUS-300 Rev.J)に記載されている CRC 値計算用の関数と同等のものです。

```

unsigned short CalcCRC16swap(
    unsigned char* puchMsg,                /* message to calculate */
    unsigned short usDataLen)             /* quantity of bytes in message */
{
    unsigned char uchCRCHi = 0xFF;        /* high byte of CRC initialized */
    unsigned char uchCRCLo = 0xFF;        /* low byte of CRC initialized */
    unsigned int  uIndex;                 /* will index into CRC lookup table */

    while(usDataLen-->0)                  /* pass through message buffer */
    {                                       /* calculate the CRC */
        uIndex = uchCRCHi ^ *puchMsg++;
        uchCRCHi = uchCRCLo ^ auchCRCHi[uIndex];
        uchCRCLo = auchCRCLo[uIndex];
    }
    return (uchCRCHi << 8 | uchCRCLo);
}

const unsigned char auchCRCHi[] =
/* Table of CRC values for high-order byte */
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,

```

R ROBO C CYLINDER

```
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,  
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,  
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,  
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,  
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,  
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,  
0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,  
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,  
};
```

```
const unsigned char auchCRCLo[] =
```

```
/* Table of CRC values for low-order byte */
```

```
0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7, 0x05, 0xC5, 0xC4, 0x04,  
0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8,  
0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,  
0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3, 0x11, 0xD1, 0xD0, 0x10,  
0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32, 0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4,  
0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,  
0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C,  
0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26, 0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0,  
0xA0, 0x60, 0x61, 0xA1, 0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,  
0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68,  
0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C,  
0xB4, 0x74, 0x75, 0xB5, 0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,  
0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54,  
0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98,  
0x88, 0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,  
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83, 0x41, 0x81, 0x80, 0x40,  
};
```

8.2 SIO と PIO の併用システム構成

RC コントローラを PIO で駆動しておき、SIO(通信)で現在の位置等をモニタすることができます。モニタが可能なのは、RTU/ASCII 共にファンクションコード 03 を使用するケリとなります。5.4.15 または 6.5.15 PIO/Modbus 切替を PIO 側に設定、及び モード SW がある RC コントローラに関しては AUTO 側に設定してご使用ください。

PIO と SIO を併用できる RC コントローラは、以下となります。

- ・PCON-C/CG/CF、PCON-CY、PCON-PL/PO、
- ・ACON-C/CG、ACON-CY、ACON-PL/PO、
- ・SCON、
- ・ERC2

SIO と PIO の併用システム構成例 1

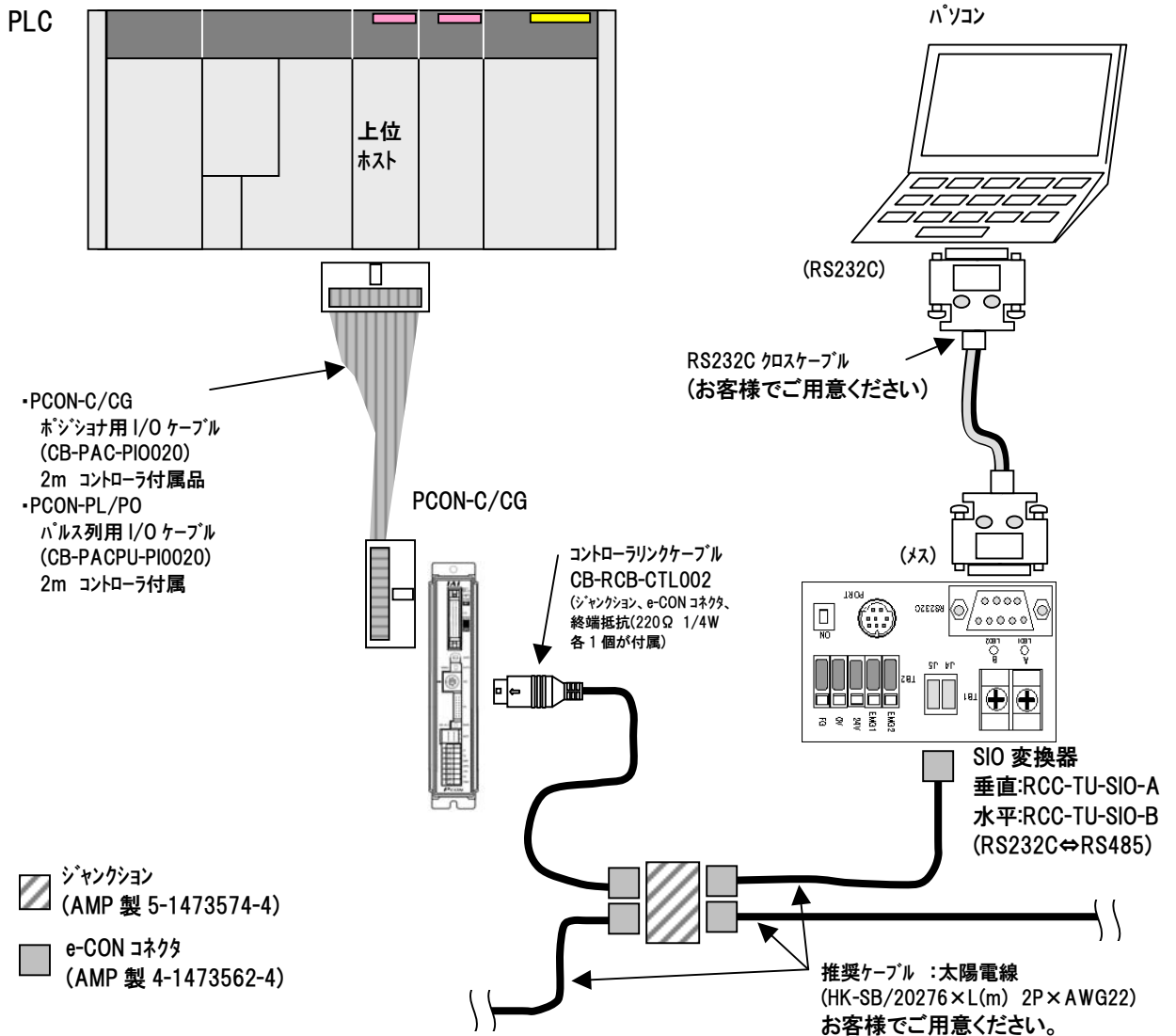


図 11.1

SIO と PIO の併用システム構成例 2

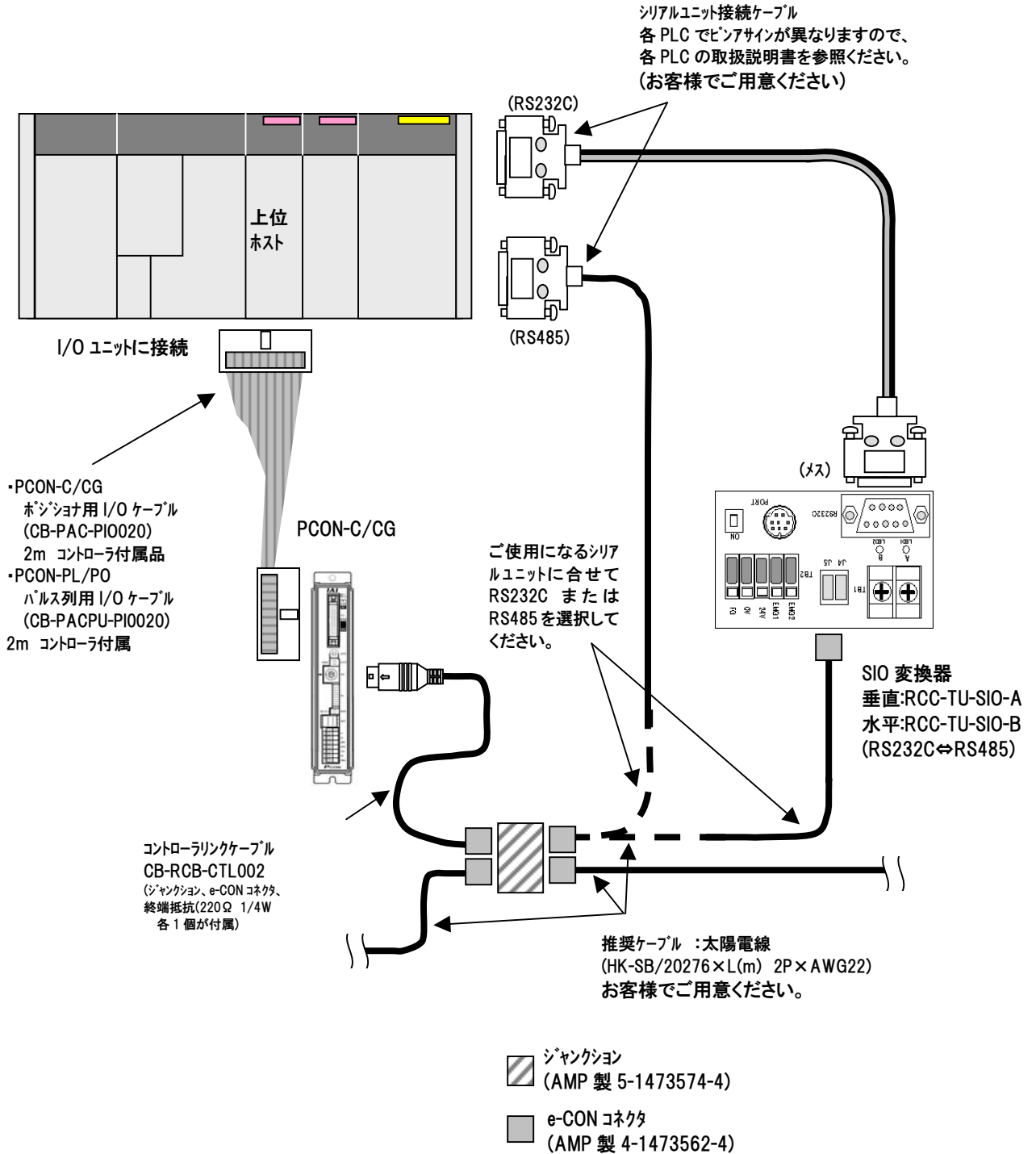


図 11. 2

R ROBO C CYLINDER



株式会社 **アイエイアイ**

本社・工場	〒424-0103	静岡県静岡市清水区尾羽577-1	TEL 054-364-5105 FAX 054-364-2589
東京営業所	〒105-0014	東京都港区芝3-24-7 芝エッセージビルディング4F	TEL 03-5419-1601 FAX 03-3455-5707
大阪営業所	〒530-0002	大阪市北区曽根崎新地2-5-3 堂島TSSビル4F	TEL 06-6457-1171 FAX 06-6457-1185
名古屋営業所	〒460-0008	名古屋市中区栄5-28-12 名古屋若宮ビル8F	TEL 052-269-2931 FAX 052-269-2933
盛岡営業所	〒020-0062	岩手県盛岡市長田町6-7 クリエ21ビル7F	TEL 019-623-9700 FAX 019-623-9701
仙台営業所	〒980-0802	宮城県仙台市青葉区二丁目14-15 アミ・グランデ2B町4F	TEL 022-723-2031 FAX 022-723-2032
新潟営業所	〒940-0082	新潟県長岡市千歳3-5-17 センザビル2F	TEL 0258-31-8320 FAX 0258-31-8321
宇都宮営業所	〒321-0953	栃木県宇都宮市東宿郷5-1-16 ルーセントビル3F	TEL 028-614-3651 FAX 028-614-3653
熊谷営業所	〒360-0847	埼玉県熊谷市龍原南1-3-12 あかりビル5F	TEL 048-530-6555 FAX 048-530-6556
茨城営業所	〒300-1207	茨城県牛久市ひたち野東5-3-2 ひたち野うしく池田ビル2F	TEL 029-830-8312 FAX 029-830-8313
多摩営業所	〒190-0023	東京都立川市柴崎町3-14-2 BOSENビル2F	TEL 042-522-9881 FAX 042-522-9882
厚木営業所	〒243-0014	神奈川県厚木市旭町1-10-6 シャンロック石井ビル3F	TEL 046-226-7131 FAX 046-226-7133
長野営業所	〒390-0877	長野県松本市沢村2-15-23 昭和開発ビル2F	TEL 0263-37-5160 FAX 0263-37-5161
甲府営業所	〒400-0031	山梨県甲府市丸の内2-12-1 ミサトビル3F	TEL 055-230-2626 FAX 055-230-2636
静岡営業所	〒424-0103	静岡県静岡市清水区尾羽577-1	TEL 054-364-6293 FAX 054-364-2589
浜松営業所	〒430-0936	静岡県浜松市中区大工町125 大発地所ビルディング7F	TEL 053-459-1780 FAX 053-458-1318
豊田営業所	〒446-0056	愛知県安城市三河安城町1-9-2 第二東洋ビル3F	TEL 0566-71-1888 FAX 0566-71-1877
金沢営業所	〒920-0024	石川県金沢市西念3-1-32 西清ビルA棟2F	TEL 076-234-3116 FAX 076-234-3107
京都営業所	〒612-8401	京都市伏見区深草下川原町22-11 市川ビル3F	TEL 075-646-0757 FAX 075-646-0758
兵庫営業所	〒673-0898	兵庫県明石市榑屋町8-34 大同生命明石ビル8F	TEL 078-913-6333 FAX 078-913-6339
岡山営業所	〒700-0973	岡山市北区中野311-114 OMOYO-ROOT BLD.101	TEL 086-805-2611 FAX 086-244-6767
広島営業所	〒730-0802	広島市中区本川町2-1-9 日宝本川町ビル5F	TEL 082-532-1750 FAX 082-532-1751
松山営業所	〒790-0905	愛媛県松山市榑味4-9-22 フォーレスト21 1F	TEL 089-986-8562 FAX 089-986-8563
福岡営業所	〒812-0013	福岡市博多区博多駅東3-13-21 エアビルWING 7F	TEL 092-415-4466 FAX 092-415-4467
大分出張所	〒870-0823	大分県大分市東大道1-11-1 タンネンバウムIII 2F	TEL 097-543-7745 FAX 097-543-7746
熊本営業所	〒862-0954	熊本県熊本市神水1-38-33 幸山ビル1F	TEL 096-386-5210 FAX 096-386-5112

お問い合わせ先

アイエイアイお客様センター エイト

(受付時間) 月～金 24時間 (月 7 : 00AM～金 翌朝 7 : 00AM)
土、日、祝日 9 : 00AM～5 : 00PM
(年末年始を除く)

フリー 0800-888-0088

FAX : 0800-888-0099 (通話料無料)

ホームページアドレス <http://www.iai-robot.co.jp>

IAI America, Inc.

Head Office : 2690 W. 237th Street Torrance, CA 90505
TEL (310) 891-6015 FAX (310) 891-0815

Chicago Office : 1261 Hamilton Parkway Itasca, IL 60143
TEL (630) 467-9900 FAX (630) 467-9912

Atlanta Office : 1220 Kennestone Circle Suite 108 Marietta, GA 30066
TEL (678) 354-9470 FAX (678) 354-9471

website: www.intelligentactuator.com

IAI Industrieroboter GmbH

Ober der Röth 4, D-65824 Schwalbach am Taunus, Germany
TEL 06196-88950 FAX 06196-889524

IAI (Shanghai) Co., Ltd.

SHANGHAI JIAHUA BUSINESS CENTER A8-303, 808, Hongqiao Rd. Shanghai 200030, China
TEL 021-6448-4753 FAX 021-6448-3992

website: www.iai-robot.com