

PCI Express, PCI, ISA, PC/104, CompactPCI
, USB, Ethernet, WiFi

CPD ボードシリーズ

ユーザーズマニュアル

〈運用編〉

NC ボード

多機能・高速 円弧・直線補間・位置決め



<http://www.hivertec.co.jp/>

この説明書は次のボードに適応しています.

PCI	HPCI-	CPD532
	HPCI-	CPD534
	HPCI-	CPD508
	HPCI-	CPD578
	HPCI-	CPD574N
	HPCI-	CPD578N
	HPCI-	CPD5212M
	HPCI-	CPD5016
	HPCI-	CPD553
PCI Express	HPCle-	CPD632
	HPCle-	CPD674N
	HPCle-	CPD678N
	HPCle-	NCB674N
	HPCle-	NCB674N(1)
ISA	HPC-	CPD234
	HPC-	CPD278
PC/104 PC104D	HPC104-	CPD132
	HP104D-	CPD364
USB	HUSB-	CPD434v2
	HUSB-	CPD434U
USB + Ethernet	HETN-	CPD834T
USB + WiFi	HWIF-	CPD834W
Compact PCI	HCPCI-	CPD734
	HCPCI-	CPD738

本マニュアル及びプログラムの全部又は一部の無断転載、コピーを禁止します.

本製品の内容に関しましては、改良等により将来予告なしに変更することがあります.

本製品の内容についてお気づきの点がございましたら、お手数ながら当社までご連絡ください.

Windows は Microsoft Corporation の米国及びその他の国における登録商標です.

その他、記載されている会社名、製品名は、各社の商標又は登録商標です.

株式会社 ハイパーテック
東京都江東区新大橋 1-8-11
三井生命新大橋ビル
TEL 03-3846-3801
FAX 03-3846-3773
sales@hivertec.co.jp

第 5.41 版 2018 年 3 月 2 日発行
不許複製・転載



本製品をご使用される前に各編の「注意事項」を必ずご一読の上
ご利用をお願い致します。

目 次

■	注意事項	1
■	保証範囲	1
■	免責事項	1
■	安全にお使い頂くために	1
■	対象ユーザー	2
■	ユーザープログラム	2
■	マニュアル構成	3
■	本マニュアルに記載される用語	4
1.	はじめに	5
1.1	マニュアルの記載内容	5
1.2	用語	6
1.2.1	レジスタ名	6
1.2.2	入力極性	7
1.2.3	関数名規約	7
2.	基本的な運用	8
2.1	プログラム手順	8
2.2	初期化と終了	9
2.2.1	ライブラリ関数を用いたボードアクセスの準備	9
2.2.2	ドライバ関数を用いたボードアクセスの準備	10
2.2.3	CPD の初期化項目	10
2.2.4	終了処理	10
2.2.5	速度と加速度	11
2.2.6	サーボ I/F	16
2.2.7	マシン I/F	17
2.2.8	マシン I/F 及びサーボ I/F の入力確認	20
2.2.9	原点復帰初期設定	22
2.2.10	原点復帰方式 (ORGmode)	23
2.3	基本の動作手順	27
2.3.1	サーボオン/サーボリセット	27
2.3.2	動作モードレジスタ(RMD)	28
2.3.3	スタートコマンド	30
2.3.4	停止コマンド	31
2.3.5	動作完了監視	32
2.3.6	INPOS 制御	37
2.3.7	指令速度モニタ	37
2.4	連続送り	38
2.4.1	動作手順	38
2.5	原点復帰	39
2.5.1	動作手順	39
2.5.2	原点サーチについて	40
2.5.3	座標系	41
2.6	位置決め	43
2.6.1	動作手順	43
2.7	タイマ動作	44

2.7.1	動作手順	44
2.8	直線補間	45
2.8.1	動作手順	45
2.9	円弧補間	47
2.9.1	動作手順	47
3.	特殊な運用	49
3.1	非対称加減速	49
3.1.1	位置決め動作時	49
3.1.2	直線補間時	49
3.2	次動作連続実行	50
3.2.1	動作用プリレジスタ	50
3.2.2	動作用プリレジスタ制御コマンド	50
3.2.3	プリレジスタの状態変化	51
3.2.4	手順	52
3.2.5	加速ブロック, 定速ブロック, 減速ブロック	53
3.2.6	次動作連続実行例	54
3.3	同時スタート(STA 出力)	56
3.3.1	コマンドによる同時スタート(STA 出力)	56
3.3.2	同時スタート(STA 入力)による軸間の同期	56
3.4	同時ストップ(STP 出力)	56
3.4.1	コマンドによる同時ストップ(STP 出力)	56
3.4.2	異常停止時自動同時ストップ(STP 入出力)	56
3.5	コンパレータ	57
3.5.1	コンパレータの種類と機能	57
3.5.2	ステッピングモータの脱調検出	57
3.5.3	PRCP5 及び RCMP5 の状態変化	58
3.6	外部機器へのトリガ信号	59
3.6.1	定ピッチ出力(CMP4 のみ)	59
3.6.2	CMP3 比較結果出力	60
3.6.3	CMP4 比較結果出力	60
3.6.4	CMP5 比較結果出力	60
3.6.5	CMP 端子出力設定	61
3.6.6	不等ピッチ出力例	63
3.7	速度オーバーライド	64
3.8	位置オーバーライド	65
3.9	複数軸の同期	66
3.9.1	コンパレータ条件一致及び加減速タイミングによる他軸スタート	66
3.9.2	CMP 条件成立による他軸スタートの簡単な例	67
3.9.3	コンパレータ条件一致 STA 出力による同期	68
3.9.4	コンパレータ条件成立での STA 出力による同時スタート設定例	69
3.9.5	軸停止による他軸スタート	70
3.9.6	軸停止トリガによるスタートの例	72
3.9.7	複数 PCL 間の直線補間	73
3.9.8	PCL 間の直線補間の例	74
3.10	PCS 機能	75
3.10.1	対応機種	75
3.10.2	設定・手順	75
3.10.3	PCS コード例	76
3.11	パルサ入力による動作	77
3.11.1	対応機種	77
3.11.2	接続	77
3.11.3	設定・手順	77

3.11.4	パルサ入力による動作のコード例	78
3.12	外部 JOG 起動入力による動作	79
3.12.1	対応機種	79
3.12.2	初期設定	79
3.12.3	JOG 送り.....	79
3.12.4	位置決め.....	80
3.12.5	ステータス	80
3.13	LTC 入力.....	81
3.13.1	対応機種	81
3.13.2	機能	81
3.13.3	設定箇所	81
3.13.4	ステータス	81
3.13.5	ラッチコマンド.....	81
3.13.6	Set/Read DLS/PCS 入力選択 (DLS/PCS).....	81
3.13.7	LTC 入力コード例	82
3.14	マスタ・スレーブ機能	83
3.14.1	対応機種	83
3.14.2	マスタ・スレーブ用語.....	83
3.14.3	マスタ・スレーブ機能のモード種類	83
3.14.4	マスタ・スレーブ機能設定ポート	83
3.14.5	マスタ追従.....	86
3.14.6	マスタ・スレーブエリア機能	90
3.14.7	並行 2 軸制御機能.....	93
3.14.8	スレーブ拡張モード	100
3.15	汎用入力/非常停止入力(HPCI-CPD508, HPCI-CPD5016)	103
3.15.1	HPCI-CPD508	103
3.15.2	HPCI-CPD5016	104
4.	PCL 資料	105
4.1	コマンド.....	105
4.1.1	コマンドデータのビット構成.....	105
4.1.2	スタートコマンド.....	105
4.1.3	速度変更コマンド.....	105
4.1.4	停止コマンド.....	105
4.1.5	コントロールコマンド	106
4.1.6	レジスタ制御コマンド	107
4.2	ステータス.....	108
4.2.1	メインステータス(MSTS)	108
4.2.2	サブステータス(SSTS).....	108
4.2.3	エラーステータス(REST)	109
4.2.4	イベントステータス(RIST).....	110
4.2.5	イベントマスクレジスタ(RIRQ).....	111
4.2.6	拡張ステータス(RSTS)	112
4.3	動作モードレジスタ(RMD)	113
4.3.1	動作モード(MOD).....	113
4.3.2	動作モードレジスタ内容	114
4.4	環境設定レジスタ	115
4.4.1	環境設定レジスタ 1(RENV1)	115
4.4.2	環境設定レジスタ 2(RENV2)	116
4.4.3	環境設定レジスタ 3(RENV3)	117
4.4.4	環境設定レジスタ 4(RENV4)	118
4.4.5	環境設定レジスタ 5(RENV5)	119
4.4.6	環境設定レジスタ 6(RENV6)	119

4.4.7	環境設定レジスタ 7(RENV7)	120
5.	更新履歴	121

図 表 目 次

表 2.2-1 DevOpen 初期値.....	9
図 2.2-1 速度・加速度とレジスタの関係	11
表 2.2-2 速度倍率設定例	11
表 2.2-3 速度関係のレジスタ設定	15
表 2.2-4 サブステータス(SSTS)	20
表 2.2-5 拡張ステータス(RSTS).....	21
表 2.2-6 センサ原点(OLS).....	23
表 2.2-7 センサ(OLS)+エンコーダ Z 相原点	24
表 2.2-8 ELS 兼用センサ原点	25
表 2.2-9 ELS 兼用センサ+エンコーダ Z 相原点	25
表 2.2-10 CTR2 参照方式.....	26
表 2.3-1 サーボオン/サーボリセット.....	27
表 2.3-2 動作モード(MOD)一覧	28
表 2.3-3 RMD の内容	29
表 2.3-4 スタートコマンド	30
表 2.3-5 停止コマンド.....	31
表 2.3-6 メインステータス(MSTS).....	33
表 2.3-7 エラーステータス(REST).....	34
表 2.3-8 イベントステータス(RIST)	35
図 2.5-1 特異点からの原点復帰動作	40
図 2.5-2 原点サーチ	40
表 2.5-1 カウンタ種別	41
図 2.5-3 ソフトリミット	42
図 3.11-1 パルス入力	77
表 3.14-1 マスタ・スレーブ機能用語	83
表 3.14-2 マスタ・スレーブ機能モード	83
図 3.14-1 マスタ追従モードブロック図.....	86
図 3.14-3 マスタ追従動作例.....	87
図 3.14-4 並行 2 軸制御機能モードブロック図	93
図 3.14-5 並行 2 軸制御機能モード接続構成	93
図 3.14-6 スレーブ拡張モード接続構成.....	100
図 3.15-1 汎用入力ポートの用途選択設定	103
図 3.15-2 汎用入力ポートの用途選択設定	104
表 4.1-1 レジスタコマンド一覧.....	107
表 4.2-1 メインステータス(MSTS).....	108
表 4.2-2 サブステータス(SSTS)	108
表 4.2-3 エラーステータス(REST).....	109
表 4.2-4 イベントステータス(RIST)	110
表 4.2-5 イベントマスクレジスタ(RIRQ).....	111
表 4.2-6 拡張ステータス(RSTS).....	112
表 4.3-1 動作モード(MOD)一覧	113
表 4.3-2 RMD の内容	114

■ 注意事項

■ 保証範囲



1. 本製品の保証期間は、お買い上げ頂いた日より 3 年間です。保証期間中に弊社の判断により欠陥が判明した場合には、本製品を弊社に引き取り、修理または交換を行います。
2. 保証期間内外に関わらず、弊社製品の使用、供給(納期)または故障に起因する、お客様及び第三者が被った、直接、間接、二次的な損害あるいは、遺失利益の損害に付いて、弊社は本製品の販売価格以上の責任を負わないものとしますので、予めご了承ください。

■ 免責事項



1. 本書に記載された内容に沿わない、製品の取付、接続、設定、運用により生じた損害に対しましては、一切の責任を負いかねますので、予めご了承ください。
2. 本製品は、一般電子機器用(工作機械・計測機器・FA/OA 機器・通信機器等)に製造された半導体製品を使用していますので、その誤作動や故障が直接、生命を脅かしたり、身体・財産等に危害を及ぼしたりする恐れのある装置(医療機器・交通機器・燃焼機器・安全装置等)に適用できるような設計、意図、または、承認、保証もされていません。
ゆえに本製品の安全性、品質および性能に関しては、本マニュアル(またはカタログ)に記載してあること以外は明示的にも黙示的にも一切保証するものではありませんので、予めご了承ください。
3. 保証期間内外に関わらず、お客様が行った弊社の承認しない製品の改造または、修理が原因で生じた損害に対しましては、一切の責任を負いかねますので、予めご了承ください。
4. 本書に記載された内容について、弊社もしくは、第三者の特許権、著作権、商標権、その他の知的所有権の権利に対する保証または実施権の許諾を行うものではありません。
また本マニュアルに記載された情報を使用したことにより第三者の知的所有権等の権利に関わる問題が生じた場合、弊社は、その責任を負いかねますので、予めご了承ください。

■ 安全にお使い頂くために



この度は、弊社 NC ボードシリーズをご採用頂きまして、誠に有り難う御座います。本マニュアルは、本製品をご使用して頂く場合の取扱い、留意点に付いて記入してありますので、必ずご一読の上ご利用をお願い致します。
尚、本マニュアルは、本マニュアルが添付されたNCボード常設箇所付近の分かりやすい場所に常時保管し、必要に応じて適宜参照・確認頂きますよう、お願い致します。

安全上の注意	
本製品のご使用前に、必ずこのユーザーズマニュアル及び付属書類を全て熟読し、内容を理解してから正しくご使用下さい。本製品の知識、安全の情報及び注意事項の全てに付いて習熟してからご使用下さい。 本ユーザーズマニュアルでは、安全注意事項のランクを「警告」、「注意」として区分してあります。	
 警告	この表示を無視して、誤った取扱いをすると、人が死亡または重傷を負う可能性が想定される内容を示しています。
 注意	この表示を無視して、誤った取扱いをすると、人が傷害を負う可能性または物的損害が想定される内容を示しています。

■ 対象ユーザー

 注 意	
	<p>本製品およびマニュアルは、以下の様な、ユーザーを対象としています。</p> <ul style="list-style-type: none">・拡張用ボードの増設および配線に付いて基本的な知識を有している方。・制御用電子機器およびパソコン等に付いて基本的な知識を有している方。

■ ユーザープログラム

 警 告	
	<p>本製品を使用し装置を動作させる時は、プログラムのデバッグを充分行ってから動作させて下さい。 プログラムに間違いがあると、思わぬ動きをすることがあり、人が死亡または重傷を負う可能性があります。</p>

■ マニュアル構成

CPD シリーズ製品のマニュアルは

- (1) CPD シリーズユーザーズマニュアル <導入編>
- (2) CPD シリーズユーザーズマニュアル <運用編>
- (3) 各製品ユーザーズマニュアル <ハードウェア編>
- (4) 各製品ユーザーズマニュアル <ソフトウェア編>

の4部構成です。

各マニュアルの内容は以下の通りです。

CPD シリーズユーザーズマニュアル <導入編>

ー全ての開発者向け

- CPD シリーズ概要
- インストール
- 動かしてみる
- 試運転
- 用語解説

別冊において以下 INtime HLS-CPD5212 導入版
の差異部分を説明

- インストール
- サンプルプログラム(動かしてみるの代替)

各製品ユーザーズマニュアル <ハードウェア編>

ー主として配線担当者向け

- 製品仕様, 購入時オプション
- ブロック図
- 接続構成
- ボード上の設定
- 外部との接続
- アクセサリ(中継コネクタボード, 接続ケーブルなど)
- 各社サーボアンプとの接続例

CPD シリーズユーザーズマニュアル <運用編>

ー主としてソフトウェア開発者向け

- 基本的な運用
- 特殊な運用

各製品ユーザーズマニュアル <ソフトウェア編>

ー主としてソフトウェア開発者向け

- ソフトウェア概要
- ライブラリ関数
- ドライバ関数
- サンプルプログラム
- ポート資料

別冊において以下 INtime HLS-CPD5212 導入版
の差異部分を説明

- ソフトウェア概要
- ライブラリ関数
- ドライバ関数
- サンプルプログラム

■ 本マニュアルに記載される用語

1. 軸は X、Y、Z、U、V、W、A、および B と命名されています。
2. 他の種類を含め、本マニュアルにおける軸を明確に指定するための入出力の記述は下記ようになります。
例: **XCWP** (CW : pulse output+ for X-axis)
3. 以下に、本マニュアルに記載の略語を記します。

ELS	ELS エンドリミットセンサ (XELS: X 軸のエンドリミットセンサ)
DLS	DLS 減速センサ (YDLS: Y 軸の減速センサ)
OLS	OLS 原点センサ (ZOLS: Z 軸の原点センサ)
CMP	CMP コンパレータ一致出力 (CMPX: X 軸のコンパレータ出力)
INPOS	インポジション INPOS
SVALM	SVALM サーボアラーム (XSVALM: X 軸のサーボアラーム)
SVCTRCL	SVCTRCL サーボエラーカウンタクリア (YSVCTRCL: Y 軸用サーボエラーカウンタクリア)
EXTPOW	EXTPOW 外部電源
EXTGND	EXTGND 外部接地
EMG	EMG マスタ停止要求 (最終的には各 4 軸の完全な停止を略し、それは X から U へと V から B にあります) (これは、ハードウェアデバイスとしての「緊急停止」ではありません。)
DSW	DSW のディップスイッチ
SYNCA	CW パルス用ボード間のマスタ・スレーブ接続用 SYNCA 同期接続
SYNCB	CCW パルス用ボード間のマスタ・スレーブ接続用 SYNCB 同期接続
AP	AP エンコーダ相入力+ (正) (XAP: X 軸用エンコーダ相入力+)
AN	AP エンコーダ相入力- (負) (XAP: X 軸用エンコーダ相入力-)
TTL	Transistor Transistor Logic

4. ステータスについて

以下のような略語については「ユーザズマニュアル(動作とソフトウェア)」を参照してください。

例: ERST、MSTS、RSTS

1. はじめに

このマニュアルは、制御 LSI に PCL6045(4 軸), PCL6025(2 軸)及び同等品(以降、これら LSI の総称として“PCL”を使用します)を使用した高速・高機能・円弧、直線補間、位置決めモータ制御ボード「CPD ボードシリーズ」の運用方法に関する共通取扱説明書です。

デバイスドライバのインストール、動作確認、用語については「CPD ボードシリーズ<導入編>」を併せて参照して下さい。

配線等については「各製品ユーザーズマニュアル<ハードウェア編>」を併せて参照して下さい。

API 関数などについては「各製品ユーザーズマニュアル<ソフトウェア編>」を併せて参照して下さい。

1.1 マニュアルの記載内容

第 1 章	はじめに	全ての開発者向け
1.1	マニュアルの記載内容	
1.2	用語	
第 2 章	基本的な運用	主としてソフトウェア開発者
2.1	プログラム手順	
2.2	初期化と終了	
2.3	動作手順	
2.4	連続送り	
2.5	原点復帰	
2.6	位置決め	
2.7	直線補間	
2.8	円弧補間	
第 3 章	特殊な機能	主としてソフトウェア開発者
3.1	非対称加減速	
3.2	次動作連続実行	
3.3	同時スタート(STA 出力)	
3.4	同時ストップ(STP 出力)	
3.5	コンパレータ	
3.6	外部機器へのトリガ信号	
3.7	速度オーバーライド	
3.8	位置オーバーライド	
3.9	複数軸の同期	
3.10	PCS 機能	
3.11	手動パルス	
3.12	外部 JOG 起動入力	
3.13	LTC 入力	
3.14	マスタ・スレーブ機能	

1.2 用語

1.2.1 レジスタ名

本マニュアルでは設定箇所の説明文中にレジスタ名が記載されています。

No	レジスタ名	プリレジスタ名	内 容
1	RMV	PRMV	移動量
2	RFL	PRFL	ベース速度設定 (ベース速度[pps]=RFL×速度倍率)
3	RFH	PRFH	動作速度設定 (動作速度[pps]=RFH×速度倍率)
4	RUR	PRUR	加速レートを決定するパラメータ
5	RDR	PRDR	減速レートを決定するパラメータ
6	RMG	PRMG	速度倍率を決定するパラメータ(速度倍率=300/(RMG+1))
7	RDP	PRDP	減速開始点
8	RMD	PRMD	動作モード
9	RIP	PRIP	円弧補間中心
10	RUS	PRUS	加速時 S 字区間設定(S 字加速区間速度[pps]=RUS×速度倍率)
11	RDS	PRDS	減速時 S 字区間設定(S 字減速区間速度[pps]=RDS×速度倍率)
12	RFA		補助速度設定 (補助速度[pps]=RFA×速度倍率)
13	RENV1		環境設定 1
14	RENV2		環境設定 2
15	RENV3		環境設定 3
16	RENV4		環境設定 4
17	RENV5		環境設定 5
18	RENV6		環境設定 6
19	RENV7		環境設定 7
20	RCTR1		カウンタ 1(指令位置用)
21	RCTR2		カウンタ 2(機械位置用)
22	RCTR3		カウンタ 3(脱調検出用)
23	RCTR4		カウンタ 4(汎用)
24	RCMP1		コンパレータ 1 用データ
25	RCMP2		コンパレータ 2 用データ
26	RCMP3		コンパレータ 3 用データ
27	RCMP4		コンパレータ 4 用データ
28	RCMP5	PRCP5	コンパレータ 5 用データ
29	RIRQ		イベントマスク設定
30	RLTC1		カウンタ 1 ラッチデータ
31	RLTC2		カウンタ 2 ラッチデータ
32	RLTC3		カウンタ 3 ラッチデータ
33	RLTC4		カウンタ 4 ラッチデータ
34	RSTS		拡張ステータス
35	REST		エラーステータス
36	RIST		イベントステータス
37	RPLS		移動残パルス 位置決カウンタ
38	RSPD		Z 相カウンタ(上位 16 ビット), 速度モニタ(下位 16 ビット, 速度[pps]=RSPD×速度倍率)
39	RSDC		減速開始点自動計算値
40	RCI	PRCI	円弧補間歩進数
41	RCIC		円弧補間歩進カウンタ
42	RIPS		補間ステータス

<表記例>

RENV1.b2-0 … 環境設定 1 レジスタのビット 2~0

RMD.b10 … 動作モードレジスタのビット 10



注 意

通常のレジスタへの書き込みはプリレジスタに行いますが
設定箇所を表記する場合はレジスタ名で表記します。

1.2.2 入力極性

B 接(カプラ電流 OFF で検出), A 接(カプラ電流 ON で検出)

1.2.3 関数名規約

ボード種別	HPCI HPCle	-CPD	HPCle-NCB	HPC-CPD	
ボード	534 532 632 508 553 574N 578N 5016 674N	5212M	674N 674N(1)	234	278
ライブラリ関数	hcp530_xxx()	hcp52c_xxx()	hcp670_xxx	hcp234_xxx()	hcp278_xxx()
ドライバ関数	cp530_xxx()	cp52c_xxx()	cp670_xxx	cp230_xxx()	cp230_xxx()

ボード種別	HUSB-CPD	HETN-CPD
ボード	434v2 434U	834T 834W
ライブラリ関数	hcp430_xxx()	het830_xxx()
ドライバ関数	cp430_xxx()	et830_xxx()

以降説明文中は xxx 部分の関数名称で説明します。

ボード種別	HCPCI-CPD	HPC104-CPD	HP104D-CPD
ボード	734 738	132	364
ライブラリ関数	hcp730_xxx()	hcp132xxx()	hcp364xxx()
ドライバ関数	cp730_xxx()	cp130_xxx()	cp360_xxx()

関数仕様については「各製品ユーザーズマニュアル<ソフトウェア編>」、使用方法については添付される「サンプルプログラム」を併せてご参照ください。

2. 基本的な運用

ここでは、CPD の基本的な運用方法を説明します。

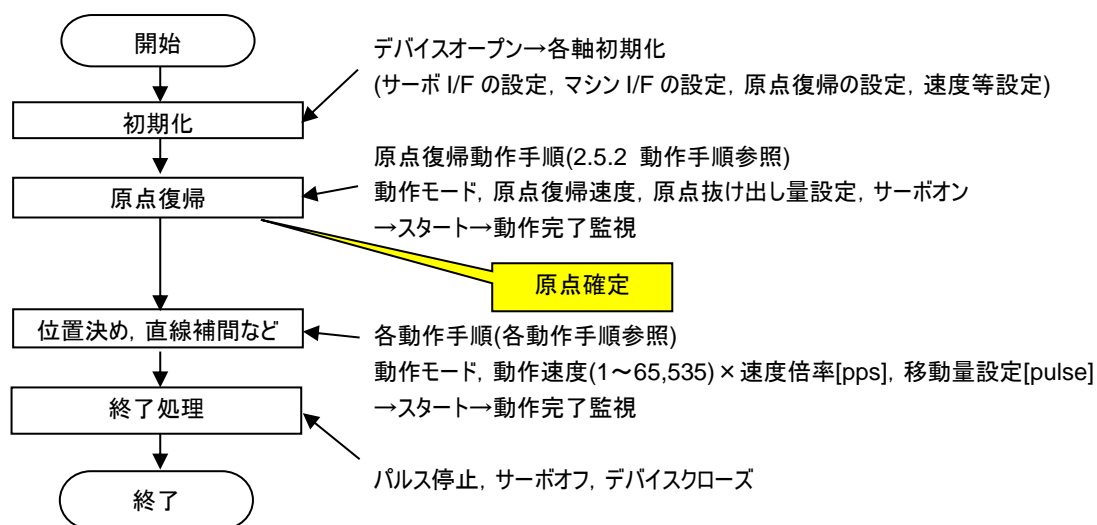
基本的運用とは次の範囲を想定しています。

- (1) デバイスオープン/クローズ
- (2) 動作手順
- (3) 連続送り
- (4) 原点復帰
- (5) 位置決め
- (6) 直線補間
- (7) 円弧補間

なお、上記以外の運用は次章を参照して下さい。

2.1 プログラム手順

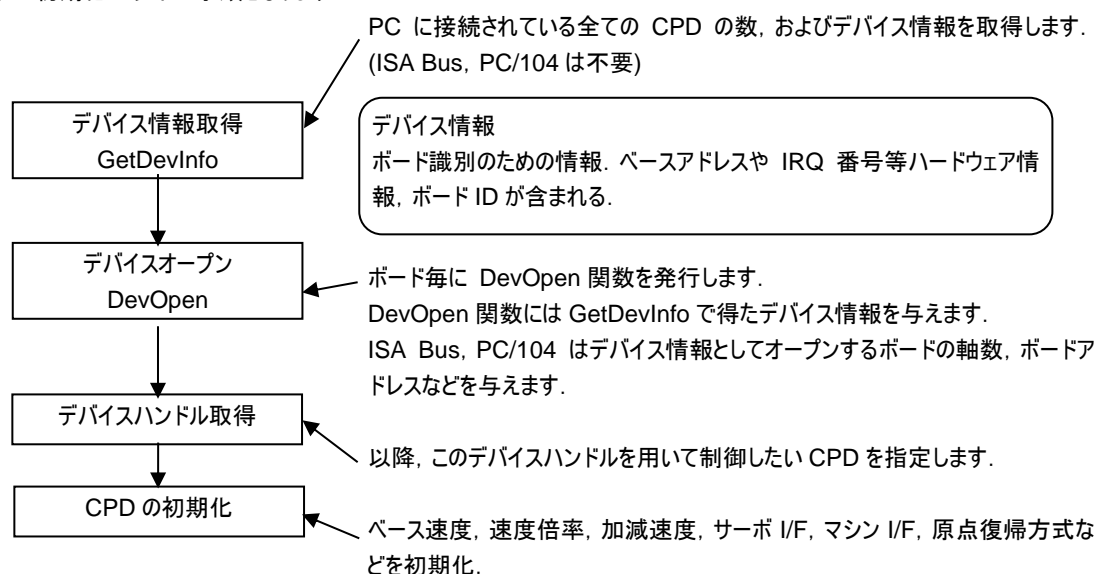
プログラム手順を下記フローに示します。



2.2 初期化と終了

2.2.1 ライブラリ関数を用いたボードアクセスの準備

デバイスの初期化は以下の手順となります。



DevOpen 関数は次の初期化を行っています。

レジスタ類の初期設定(各軸同一値で初期化を実施)、オプションポートの初期化。

レジスタ	内容	初期値	設定内容
RFL	ベース速度設定	200	$RFL \times 300 / (RMG+1) = 200[\text{pps}]$
RFH	動作速度設定	2000	$RFH \times 300 / (RMG+1) = 2000[\text{pps}]$
RUR	加速レート設定	1364	直線加減速時加速時間、減速時間 加速時 200 → 2000 pps まで約 0.5 秒 減速時 2000 → 200pps まで約 0.5 秒
RMG	速度倍率設定	299	速度倍率 = $300 / (RMG+1) = 1$ 倍
RFA	補助速度	200	$RFA \times 300 / (RMG+1) = 200[\text{pps}]$
RMD	動作モード	08008000h	
RENV1	環境設定 1	20434004h	個別パルス方式, SVALM, OLS は B 接, INPOS は A 接, SVCTRCL 自動出力なし, SVCTRCL 出力幅 13ms, SVCTRCL 出力後タイマ 104ms
RENV2	環境設定 2	0020fd65h	CPD364
		0020fd55h	CPD574N, CPD578, CPD578N, CPD278, CPD5212M CPD434v2, CPD834T, CPD834W
		0020f555h	上記以外
RENV3	環境設定 3	00f00002h	原点復帰モード 2(OLS+Z 相), 原点復帰完了時各カウンタクリア
RIRQ	イベントマスク	1	正常停止時
その他		0	PCL の初期値
オプション ポート	ELS 入力極性等		ELS 入力極性は B 接 DLS 端子 → PCS その他 POW ON 時初期値

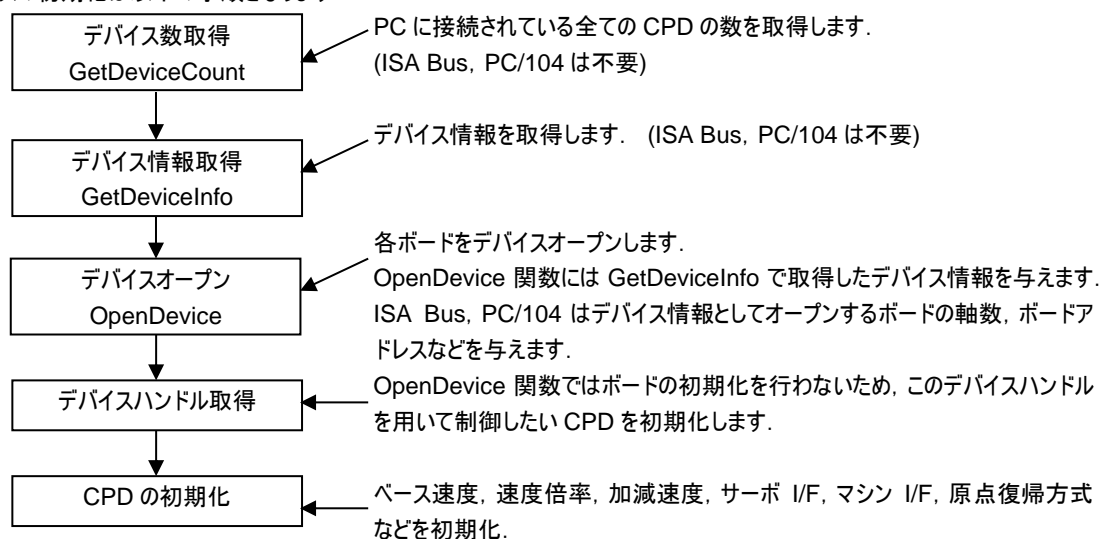
表 2.2-1 DevOpen 初期値

DevOpen 初期値から変更が必要な部分は各種 API 関数を用いて CPD の初期化を行います。

2.2.2 ドライバ関数を用いたボードアクセスの準備

ドライバ関数を用いた場合は手順が若干異なります。

デバイスの初期化は以下の手順となります。



2.2.3 CPDの初期化項目

通常, 以下の項目はアプリケーションの初期化で行います。

No.	項目	参照箇所
1	速度倍率	2.2.5 速度と加速度
2	ベース速度	
3	加減速度	
4	サーボ I/F	2.2.6 サーボ I/F
5	マシン I/F	2.2.7 マシン I/F
6	原点復帰方式	2.2.9 原点復帰初期設定 , 2.2.10 原点復帰方式

2.2.4 終了処理

アプリケーション上で必要な終了処理(パルス停止, サーボオフ等)を終えた後, DevClose 関数または CloseDevice 関数で, 使用していた資源を解放します。

2.2.5 速度と加速度

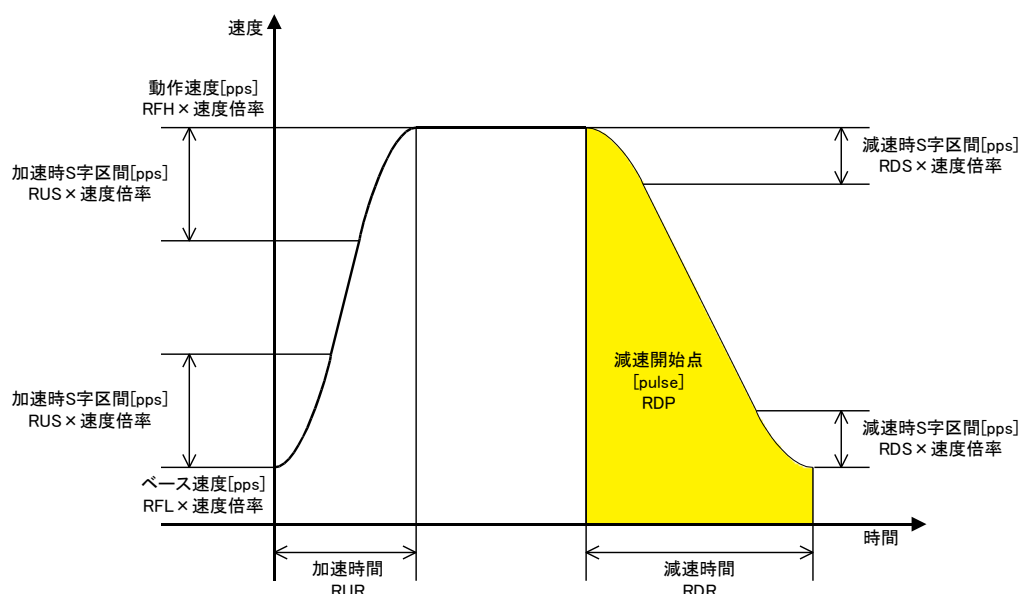


図 2.2-1 速度・加速度とレジスタの関係

速度倍率、ベース速度、補助速度、加速レート、減速レートは通常初期化時に設定します。

(1) 指令速度

指令速度は速度レジスタ(RFL,RFH,RFA)と速度倍率との積で決定されます。

高倍率になるほど設定できる速度間隔が粗くなりますので、通常はできるだけ小さな倍率を使用します。

補間動作の場合は補間代表軸(X 軸に最も近い軸)の速度倍率が使用されます。

RMG と速度倍率の関係は次式のようになります。

$$\text{速度倍率[倍]} = \frac{300}{\text{RMG} + 1} \quad \text{RMG の設定範囲: 2~4095(0FFFh)}$$

倍率	RMG の値	F(pps)	指令周波数 F の取りうる範囲
0.1	2999	0.1×RFx	0.1 pps ~ 6.5535 kpps
0.2	1499	0.2×RFx	0.2 pps ~ 13.107 kpps
0.5	599	0.5×RFx	0.5 pps ~ 32.7675 kpps
1	299	1×RFx	1 pps ~ 65.535 kpps
2	149	2×RFx	2 pps ~ 131.070 kpps
5	59	5×RFx	5 pps ~ 327.675 kpps
10	29	10×RFx	10 pps ~ 655.350 kpps
20	14	20×RFx	20 pps ~ 1,310.700 kpps
50	5	50×RFx	50 pps ~ 3,276.750 kpps
75	3	75×RFx	75 pps ~ 4,915.125 kpps
100	2	100×RFx	100 pps ~ 6,553.500 kpps

表 2.2-2 速度倍率設定例

したがってベース速度、動作速度、補助速度は以下のようになります。

$$\text{ベース速度[PPS]} = \frac{\text{RFL} \times 300}{\text{RMG} + 1}$$

$$\text{動作速度[PPS]} = \frac{\text{RFH} \times 300}{\text{RMG} + 1}$$

$$\text{補助速度[PPS]} = \frac{\text{RFA} \times 300}{\text{RMG} + 1}$$

RFL, RFH, RFA の設定範囲: 1~65535(FFFFh)

(2) 加減速

加速時間(減速時間)は RFL, RFH, RUR(RDR)との関係で決定されます。
RDR=0 の時は RDR=RUR とした減速時間となります(加速時間 = 減速時間)
RFL, RFH, RUR(RDR)と加速時間(減速時間)の関係は以下のようになります。

- 加速時間

- 直線加速時 (RMD.b10=0)

$$\text{加速時間[sec]} = \frac{(\text{RFH} - \text{RFL}) \times (\text{RUR} + 1) \times 4}{19,660,800}$$

- 直線部分無 S 字加速時 (RMD.b10=1 かつ RUS=0)

$$\text{加速時間} = \frac{(\text{RFH} - \text{RFL}) \times (\text{RUR} + 1) \times 8}{19,660,800}$$

- 直線部分有 S 字加速時 (RMD.b10=1 かつ RUS>0)

$$\text{加速時間[sec]} = \frac{(\text{RFH} - \text{RFL} + 2 \times \text{RUS}) \times (\text{RUR} + 1) \times 4}{19,660,800}$$

- 減速時間

- 直線減速時 (RMD.b10=0)

$$\text{減速時間[sec]} = \frac{(\text{RFH} - \text{RFL}) \times (\text{RDR} + 1) \times 4}{19,660,800}$$

- 直線部分無 S 字減速時 (RMD.b10=1 かつ RDS=0)

$$\text{加速時間} = \frac{(\text{RFH} - \text{RFL}) \times (\text{RDR} + 1) \times 8}{19,660,800}$$

- 直線部分有 S 字減速時 (RMD.b10=1 かつ RDS>0)

$$\text{減速時間[sec]} = \frac{(\text{RFH} - \text{RFL} + 2 \times \text{RDS}) \times (\text{RDR} + 1) \times 4}{19,660,800}$$

- S 字区間

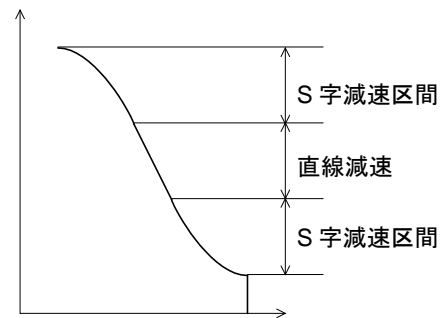
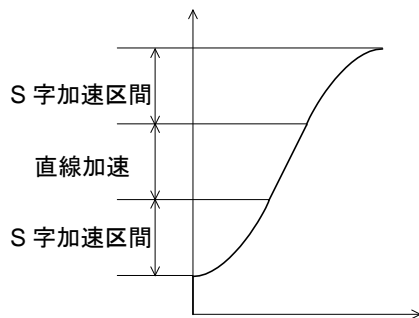
S 字加減速動作の S 字区間を 1~32,767(7FFFh)の範囲で設定します。

S 字加速区間の範囲 Ssu は RMG の設定値との計算値になります。

S 字減速区間の範囲 Ssd も同様になります。

$$\text{S 字加速区間 Ssu [pps]} = \frac{\text{RUS} \times 300}{\text{RMG} + 1}$$

$$\text{S 字減速区間 Ssd [pps]} = \frac{\text{RDS} \times 300}{\text{RMG} + 1}$$



ただし RUS(RDS)=0 を設定した場合、内部演算により $\frac{\text{RFH} - \text{RFL}}{2}$ が代用され、直線加速(減速)部分のない S 字加速(減速)動作となります。



注意

減速開始点自動計算(RMD.b13=0)の場合は減速時間 \leq (加速時間 $\times 2$)となるように RDR の値を設定してください。

減速時間 $>$ (加速時間 $\times 2$)の設定をする場合は減速開始点手動計算(RMD.b13=1)にし、かつ減速開始点(RDP)を計算し設定します。また FH 補正機能は OFF(RMD.b26=1)にします。

減速開始点自動設定時、減速時間 \leq (加速時間 $\times 2$)の設定をした場合・・・不適切な設定

速度
動作速度
ベース速度

時間

加速時間

減速時間

ベース速度まで減速しないで停止してしまう。

(3) 減速開始点

加減速・位置決め動作の場合の減速開始点(終点からのパルス数で指定。残移動量が減速開始点以下になると減速を開始します)を設定します。RMD.b13 の状態により、RDP に設定する値の意味は異なります。

- マニュアル計算時(RMD.b13=1)の場合

減速開始点のパルス数を 0～16,777,215(0FFFFFFh)の範囲で設定します。

(位置決めカウンタ値) \leq (RDP 設定値)のタイミングで減速を開始します。

減速開始点の最適値は次式の様になります。

- 直線減速時 (RMD.b10=0)

$$\text{最適値[パルス]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1)}{(RMG + 1) \times 32,768}$$

- 直線部分無 S 字減速時 (RMD.b10=0 かつ RDS=0)

$$\text{最適値[パルス]} = \frac{(RFH^2 - RFL^2) \times (RDR + 1) \times 2}{(RMG + 1) \times 32,768}$$

- 直線部分有 S 字減速時 (RMD.b10=0 かつ RDS>0)

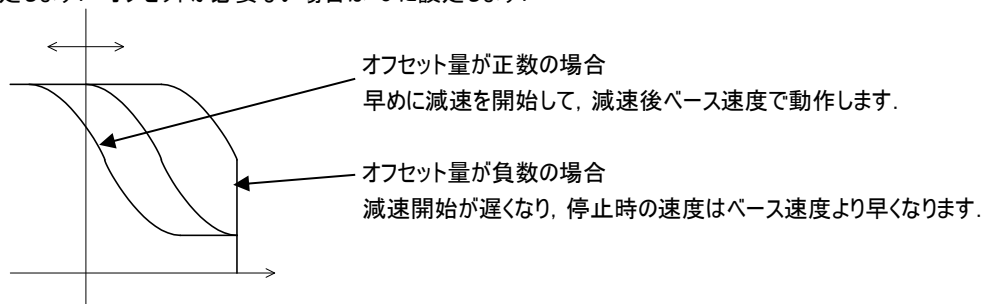
$$\text{最適値[パルス]} = \frac{(RFH + RFL) \times (RFH - RFL + 2 \times RDS) \times (RDR + 1)}{(RMG + 1) \times 32,768}$$

減速開始点が最適値より小さい場合、停止時の速度はベース速度より早くなります。

逆に、最適値より大きい場合、減速終了後にベース速度で定速動作をします。

- 自動計算時 (RMD.b10=0) の場合

自動で設定される減速開始点に対してオフセットとなり、-8,388,608 (800000h)～ 8,388,607 (7FFFFFFh) の範囲で設定します。オフセットが必要ない場合は '0' に設定します。



- 直線補間時の減速開始点の考え方

<合成速度一定 ON の場合>

動作モードレジスタ(PRMD)の bit15=1 の場合、合成速度一定制御が ON になります。

この時、 $\sqrt{2}$ 制御、 $\sqrt{3}$ 制御との差の関係で、移動量が全て同じ場合や 1 軸のみ動作する場合を除き、減速開始点の計算は困難です。

- ・2 軸で移動量が同じ場合

前述の減速開始点の式で計算した $\sqrt{2}$ で割った値を減速開始点とし 2 軸ともに設定します。

- ・3 軸で移動量が同じ場合

前述の減速開始点の式で計算した $\sqrt{3}$ で割った値を減速開始点とし 3 軸ともに設定します。

- ・1 軸のみ動作する場合

前述の減速開始点の式で計算した値を動作する軸に設定します。

<合成速度一定 OFF の場合>

動作モードレジスタ(PRMD)の bit15=0 の場合、合成速度一定制御が OFF になります。

この場合、補間代表軸に設定された速度データが長軸速度となります。

前述の減速開始点の計算式の計算した値を長軸に設定します。

(4) FH 補正機能(三角駆動自動回避)

位置決め動作時に加減速動作をさせた場合、十分加減速できない移動量の場合(移動量が少ない場合)に自動的に動作速度を低下させて三角駆動を回避する機能です。

(5) レジスタ設定箇所及び対応関数

No.	項目	設定箇所	ビット長	設定範囲	使用する関数
1	ベース速度	RFL, (RMG)	16	1～65,535	SetFLSpd, (SetMult)
2	動作速度	RFH, (RMG)	16	1～65,535	WritFHSpd, (SetMult)
3	補助速度	RFA, (RMG)	16	1～65,535	SetAuxSpd, (SetMult)
4	加減速方式	RMD.b10	1	直線:0, S 字:1	SetAccProfile
5	加速レート	RUR	16	1～65,535	SetAccRate
6	減速レート	RDR	16	0～65,535	SetDecRate
7	加速時 S 字区間	RUS	15	0～32,767	wReg
8	減速時 S 字区間	RDS	15	0～32,767	wReg
9	減速開始点の計算方法	RMD.b13	1	自動:0, 手動:1	SetAutoDec
10	減速開始点	RDP	24	0～16,777,215 または -8,388,608～8,388,607	SetDecPoint
11	FH 補正機能	RMD.b26	1	0:有効, 1:無効	wReg

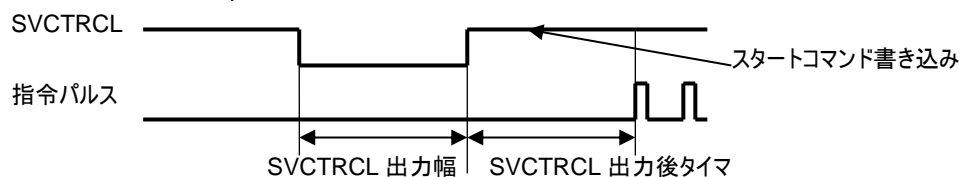
表 2.2-3 速度関係のレジスタ設定

2.2.6 サーボ I/F

モータアンプの設定及び仕様に応じて設定します。使用しない入力端子はオープンにし、入力極性は A 接にします。

No.	項目	設定箇所	使用する関数	備 考
1	指令パルス出力形式	RENV1.b2-0 個別: 100 , 共通: 010, 位相差: 110	SetCmdPulse	
2	SVALM 入力極性	RENV1.b9 A 接: 1, B 接: 0	SetSvAlm	CPD508, CPD5016 使用時に SVALM を使用する場合は PCL 単位で各軸の極性を同じにする必要があります。
3	SVALM 入力時動作	RENV1.b8 即停止: 0 , 減速停止: 1	SetSvAlm	
4	INPOS 制御有効/無効	RMD.b9 有効: 1, 無効: 0	SetInpos	
5	INPOS 入力極性	RENV1.b22 A 接: 1 , B 接: 0	SetInpos	
6	SVCTRCL 自動出力	RENV1.b10 異常停止時自動出力: 1 RENV1.b11 原点復帰完了時自動出力: 1	SetSvCtrCl	異常停止の停止方法が減速停止の時、または DLS 有効かつ DLS ON の時は自動出力されません。
7	SVCTRCL 出力幅	RENV1.b14-12 000: 12 μ s, 001: 102 μ s, 010: 409 μ s, 011: 1.6ms, 100: 13ms , 101: 52ms, 110: 104ms, 111: レベル	wReg	
8	SVCTRCL 出力後タイマ	RENV1.b17,16 00: なし, 01: 12 μ s, 10: 1.6ms, 11: 104ms	wReg	

太字はライブラリ DevOpen 関数初期値。



2.2.7 マシン I/F

接続されるセンサ仕様、及びアプリケーションの仕様に応じて設定します。

使用しない入力端子はオープンにし、入力極性は A 接にします。

No.	項目	設定箇所	使用する関数	備考
1	±ELS 入力極性	各ボード ELS 入力極性 切替ポート(別表)	SetEls	各機種により設定箇所が異なります。
2	±ELS 入力時動作	RENV1.b3 即停止:0 , 減速停止:1	SetEls	減速停止時は減速距離に注意 (減速距離÷減速開始点パルス数)
3	OLS 入力極性	RENV1.b7 A 接:1, B 接:0	SetOls	原点復帰時のみ使用
4	DLS 入力極性	RENV1.b6 A 接:1, B 接:0	SetDisSel	
5	DLS 入力時動作	RENV1.b4 減速のみ:0 , 減速停止:1	SetDisSel	RMD.b8=1 の時のみ有効
6	DLS 入力ラッチ	RENV1.b5 ラッチしない:0 , ラッチする:1	SetDisSel	DLS 信号がパルスの場合には入力ラッチが 使用できます

太字はライブラリ DevOpen 関数初期値。

HPCI-CPD508, 5016, 5212M には DLS がありません。

(1) ELS 入力極性切替ポート

各ボードにより設定箇所が異なります。

	ボード種別	オプションポート 設定箇所	bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	HPCI-CPD532 HPCIe-CPD632	BAR+80h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
2	HPCI-CPD534 HPCI-CPD553 HPCI-CPD574N HPCIe-CPD674N	BAR+80h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
3	HPCI-CPD508 HPCI-CPD578N	BAR+80h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
4	HPCI-CPD5016	BAR+80h		U4	Z4	Y4	X4	U3	Z3	Y3	X3	U2	Z2	Y2	X2	U1	Z1	Y1	X1
5	HPCI-CPD5212M	BAR3+00h		-	-	-	-	U3	Z3	Y3	X3	U2	Z2	Y2	X2	U1	Z1	Y1	X1
6	HPCIe-NCB674N HPCIe-NCB674N(1)	BAR+80h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
7	HPC-CPD234	BAR+20h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
8	HPC-CPD278	BAR+40h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
9	HPC104-CPD132	BAR+10h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
10	HP104D-CPD364	BAR+20h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
11	HUSB-CPD434v2 HETN-CPD834T HWIF-CPD834W	コマンド 80h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
12	HCPCI-CPD734	BAR+80h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
13	HCPCI-CPD738	BAR+80h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X

BAR:ベースアドレス

各ビット“0”で対象軸の±ELS 入力極性は B 接。

各ビット“1”で対象軸の±ELS 入力極性は A 接。

※ HPCI-CPD553 は、3軸仕様の製品となるため、U 軸がありません。

(2) DLS/PCS 入力切替ポート

各ボードにより設定箇所が異なります。

	ボード種別	オプションポート 設定箇所	bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	HPCI-CPD532 HPCIe-CPD632	BAR+82h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
2	HPCI-CPD534 HPCI-CPD553 HPCI-CPD574N HPCIe-CPD674N	BAR+82h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
3	HPCI-CPD578N	BAR+82h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
4	HPCI-CPD508	BAR+88h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
5	HPCI-CPD5016	BAR+88h		U4	Z4	Y4	X4	U3	Z3	Y3	X3	U2	Z2	Y2	X2	U1	Z1	Y1	X1
6	HPCIe-NCB674N HPCIe-NCB674N(1)	BAR+82h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
7	HPC-CPD234	BAR+22h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
8	HPC-CPD278	BAR+42h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
9	HPC104-CPD132	BAR+12h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
10	HP104D-CPD364	BAR+22h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
11	HUSB-CPD434v2 HETN-CPD834T HWIF-CPD834W	コマンド 82h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
12	HCPCI-CPD734	BAR+82h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
13	HCPCI-CPD738	BAR+82h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X

BAR: ベースアドレス

各ビット“0”で対象軸の DLS 端子を DLS 入力. (HPCI-CPD508, HPCI-CPD5016 を除く)

各ビット“1”で対象軸の DLS 端子を PCS 入力.

(HPCI-CPD508 は BOLS 端子, HPCI-CPD5016 の場合, X1~U2 は U2OLS 端子, X3~U4 は U4OLS 端子)

※ HPCI-CPD553 は、3軸仕様の製品となるため、U 軸がありません。

(3) DLS 入力時の動作

No.	設定内容	定速動作	加減速動作
1	減速のみ, ラッチしない RENV1.b4=0 RENV1.b5=0	DLS を無視します.	<p>動作速度</p> <p>ベース速度</p> <p>DLS</p> <p>OFF ON OFF</p> <p>時間</p> <p>DLS ON で減速</p> <p>DLS OFF で加速</p>
2	減速のみ, ラッチする RENV1.b4=0 RENV1.b5=1	DLS を無視します.	<p>動作速度</p> <p>ベース速度</p> <p>DLS</p> <p>OFF ON OFF</p> <p>時間</p> <p>DLS ON で減速</p>
3	減速停止, ラッチしない RENV1.b4=1 RENV1.b5=0	即停止します.	<p>動作速度</p> <p>ベース速度</p> <p>DLS</p> <p>OFF ON OFF</p> <p>時間</p> <p>DLS ON で減速</p> <p>減速中に DLS OFF で再加速</p>
4	減速停止, ラッチする RENV1.b4=1 RENV1.b5=1	即停止します.	<p>動作速度</p> <p>ベース速度</p> <p>DLS</p> <p>OFF ON OFF</p> <p>時間</p> <p>DLS ON で減速</p> <p>減速中に DLS OFF でも無視され減速停止</p>

2.2.8 マシン I/F 及びサーボ I/F の入力確認

実機とソフトウェアの接合テスト時や装置起動時にはサーボ I/F 入力状態やマシン I/F 入力状態を確認する必要があります。サブステータス(SSTS)や拡張ステータス(RSTS)によりサーボ I/F やマシン I/F の状態を確認します。

またソフトウェアデバッグ時に、現在の動作状態を知る必要がある場合(停止中にならない、スタートしない場合など)は RSTS.b3-0 を確認します。

(1) サブステータス(SSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SDLS	SOLS	SMEL	SPEL	SALM	SFC	SFD	SFU	不	使	用	(予	約)	SVRST	SVON

bit	名称	説 明	備 考
0	SVON	'1' = SVON ON	指令状態のモニタ
1	SVRST	'1' = SVRST ON	
7-2	---	不使用(予約)	
8	SFU	'1' = 加速中	レベル検出 } 入力極性設定が反映される。 A 接設定時はカプラ電流 ON で 1 B 接設定時はカプラ電流 OFF で 1
9	SFD	'1' = 減速中	
10	SFC	'1' = 定速動作中	
11	SALM	'1' = SVALM 検出中	
12	SPEL	'1' = +ELS 検出中	
13	SMEL	'1' = -ELS 検出中	
14	SOLS	'1' = OLS 検出中	
15	SDLS	'1' = DLS 検出中	

表 2.2-4 サブステータス(SSTS)

(2) 拡張ステータス(RSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDIN	SLTC	SCLR	SDRM	SDRP	SEZ	SERC	SPCS	SEMG	SSTP	SSTA	SDIR	CND3	CND2	CND1	CND0
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	PFM1	PFM0	PFC1	PFC0	0	SINP

bit	名称	内 容
0	CND0	動作状態を表す
1	CND1	0000: 停止中
2	CND2	0001: DR 入力待ち
3	CND3	0010: STA 入力待ち 0011: スタートコマンドが発行されていて、軸スタート条件成立待ち状態. 0100: 指定軸の停止によるスタート待ち 0101: サーボ偏差カウンタ・クリアタイム on 中 0110: 指令方向信号切替中(共通パルス指令時方向切替時間 0.2ms) 0111: バックラッシュ補正中(バックラッシュ補正中の時間は補正パルス量と補助速度による) 1000: パルス入力待ち 1001: FA(補助速度)定速で動作中 1010: FL 定速で動作中 1011: 加速中 1100: FH 定速で動作中 1101: 減速中 1110: INPOS 信号待ち 1111: その他
4	SDIR	動作方向(0: +方向, 1: -方向)
5	SSTA	'1'= 同時スタート信号 on 状態
6	SSTP	'1'= 同時停止信号 on 状態
7	SEMG	'1'= EMG 信号入力中
8	SPCS	'1'= PCS 位置決めスタート信号入力中
9	SERC	'1'= サーボ偏差カウンタクリア信号出力中
10	SEZ	'1'= エンコーダ Z 相信号入力中
11	SDRP	'1'= +DR 信号入力中
12	SDRM	'1'= -DR 信号入力中
13	SCLR	'1'= CLR 信号入力中
14	SLTC	'1'= LTC 信号入力中
15	SDIN	'1'= DLS 信号入力中
16	SINP	'1'= INPOS 信号入力中
17	未定義	常に'0'
19,18	PFC1,0	RCMP5 用プリレジスタの使用状態 00:未確定, 01:レジスタ確定, 10:1 st プリレジスタ確定, 11:2 nd プリレジスタ確定(プリレジスタフル)
21,20	PFM1,0	動作用プリレジスタ(RCMP5 用以外)の使用状態 00:未確定, 01:レジスタ確定, 10:1 st プリレジスタ確定, 11:2 nd プリレジスタ確定(プリレジスタフル)
31-22	未定義	常に'0'

表 2.2-5 拡張ステータス(RSTS)

(3) 対応関数

No.	内 容	使用する関数
1	サブステータス読み出し	ReadSubSts / rMstsW
2	拡張ステータス読み出し	ReadExSts / rReg

2.2.9 原点復帰初期設定

原点が必要な機械系において原点に関する初期設定しておきます。

(1) レジスタ設定箇所及び対応関数

No.	項目	選択事項	設定箇所	使用する関数	備考
1	原点復帰方式 (ORGmode)	センサ原点(OLS)/ センサ原点とZ相/ ELS 兼用原点/ ELS 兼用原点とZ相等	RENV3.b3-0 ORGmode を設定	SetOrgMode	DevOpen 初期値 ORGmode=2 (OLS+Z 相)
2	速度パターン	定速復帰/高速復帰	スタートコマンドによる.	CnstStartFH /AccStart	
3	カウンタのクリア	各カウンタクリアする/しない	RENV3.b23-20 CTR1 クリア:b20=1 CTR2 クリア:b21=1 CTR3 クリア:b22=1 CTR4 クリア:b23=1	wReg	DevOpen 初期値 全カウンタクリア
4	Z 相カウント 回数	1~15 回	RENV3.b7-4 設定値+1(回)	SetEz	DevOpen 初期値 1 回

2.2.10 原点復帰方式 (ORGmode)

原点復帰シーケンスを設定します。センサ構成などに応じて適当な原点復帰方式を選択します。
原点復帰を特異点(OLS 上または原点復帰方向 ELSと OLS の間)からスタートする場合は原点サーチを使用します。
 原点サーチについては「[2.5.2 原点サーチについて](#)」を参照してください。
 また原点復帰の動作手順については「[2.5.1 動作手順](#)」を参照してください。

(1) センサ原点(OLS)

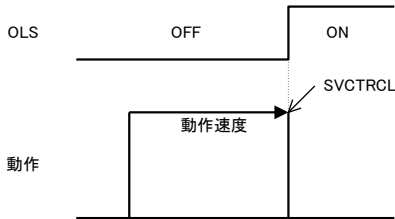
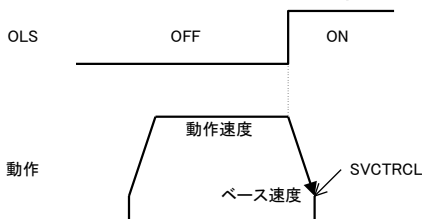
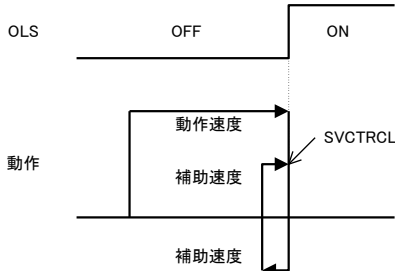
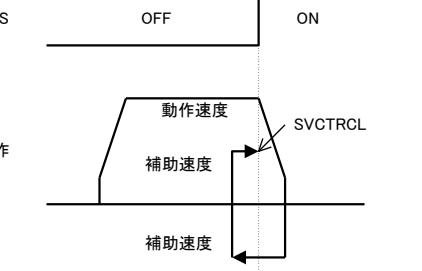
ORGmode	FH 定速復帰	高速復帰
0	<div>OLS ON で即停止し完了。 カウンタリセットタイミング: OLS OFF→ON 時</div> 	<div>OLS ON 検出で減速停止し完了 (DLS 減速, OLS 原点も可能) カウンタリセットタイミング: OLS OFF→ON 時 (注)</div> 
1	<div>OLS ON で即停止後、反転拔出し、再度 OLS 検出で完了 カウンタリセットタイミング: 完了時</div> 	<div>OLS ON で減速停止後、反転拔出し、再度 OLS 検出で完了 カウンタリセットタイミング: 完了時</div> 

表 2.2-6 センサ原点(OLS)

- 注1. No.0 の高速減速原点方式では原点精度がサーボゲインの影響をうけます。OLS の手前に DLS を設けて減速させる方法もあります。
- 注2. No.1 の高速原点復帰方式では減速停止時に OLS が OFF の場合、抜け出し完了とみなし反転せずに補助速度で原点復帰方向へ動作しますので、減速距離に注意してください。

(2) センサ(OLS)+エンコーダ Z 相原点

ORGmode	FH 定速復帰	高速復帰
2	<p>OLS on 検出後の Z 相で即停止し完了 カウンタリセットタイミング: Z 相カウントアップ時</p>	<p>OLS on で減速し, 検出後の Z 相で即停止し完了 カウンタリセットタイミング: Z 相カウントアップ時</p>
3	<p>ORGmode2 と同様</p>	<p>OLS OFF→ON 後の Z 相カウントで減速停止し完了. カウンタリセットタイミング: Z 相カウントアップ時(注)</p>
4	<p>OLS OFF→ON で即停止, 反転補助速度定速の Z 相検出で完了. カウンタリセットタイミング: Z 相カウントアップ時</p>	<p>OLS OFF→ON で減速停止, 反転補助速度定速の Z 相検出で完了. カウンタリセットタイミング: Z 相カウントアップ時</p>
5	<p>OLS on で即停止, 反転定速の Z 相検出で即停止し完了. カウンタリセットタイミング: Z 相カウントアップ時</p>	<p>OLS OFF→ON で減速停止, 反転高速の Z 相検出で減速停止し完了. カウンタリセットタイミング: Z 相カウントアップ時(注)</p>

注. No.3, 5 の高速減速原点方式では原点精度がサーボゲインの影響をうけます.

表 2.2-7 センサ(OLS)+エンコーダ Z 相原点

(3) ELS 兼用センサ原点

ORGmode	FH 定速復帰	高速復帰
6	<p>ELS on で即停止後,反転定速拔出し,ELS off 検出で完了 カウンタリセットタイミング: ELS ON→OFF 時</p>	<p>ELS on で停止(ELS での停止方法により即停止または減速停止)後, 反転定速拔出し,ELS off 検出で完了. カウンタリセットタイミング: ELS ON→OFF 時</p>

表 2.2-8 ELS 兼用センサ原点

(4) ELS 兼用センサ+エンコーダ Z 相原点

ORGmode	FH 定速復帰	高速復帰
7	<p>ELS ON で即停止後, 反転定速の Z 相検出で完了. カウンタリセットタイミング: Z 相カウントアップ時</p>	<p>ELS ON で停止(ELS での停止方法により即停止または減速停止)後, 反転定速の Z 相検出で完了. カウンタリセットタイミング: Z 相カウントアップ時</p>
8	<p>ELS ON で即停止後, 反転定速の Z 相検出で完了. カウンタリセットタイミング: Z 相カウントアップ時</p>	<p>ELS ON で停止(ELS での停止方法により即停止または減速停止)後, 反転高速の Z 相検出で減速停止し完了. カウンタリセットタイミング: Z 相カウントアップ時(注)</p>

注. No.8 の高速減速原点方式では原点精度がサーボゲインの影響をうけます.

表 2.2-9 ELS 兼用センサ+エンコーダ Z 相原点

(5) CTR2 参照方式

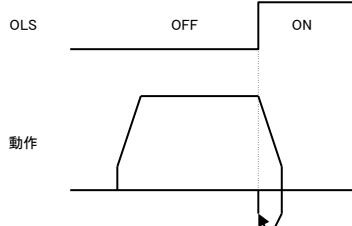
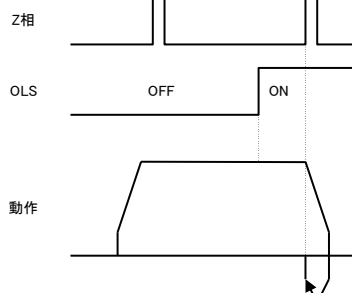
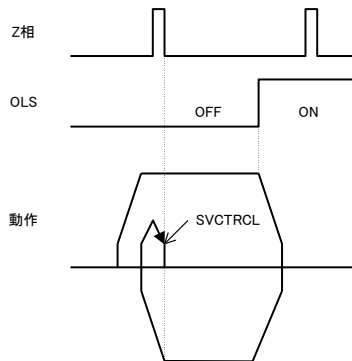
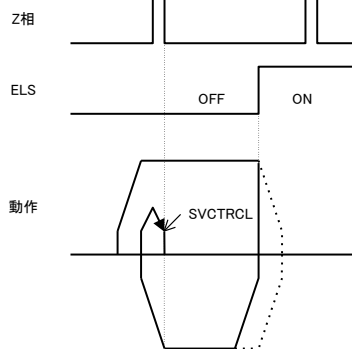
ORGmode	原点復帰内容	
9		ORGmode0 の動作後, 機械位置(CTR2)0 点復帰 カウンタリセットタイミング: OLS OFF→ON 時
10		ORGmode3 の動作後, 機械位置(CTR2)0 点復帰 カウンタリセットタイミング: Z 相カウントアップ時
11		ORGmode5 の動作後, 機械位置(CTR2)0 点復帰 カウンタリセットタイミング: Z 相カウントアップ時
12		ORGmode8 の動作後, 機械位置(CTR2)0 点復帰 カウンタリセットタイミング: Z 相カウントアップ時

表 2.2-10 CTR2 参照方式

注 1. No.9～12 の原点復帰方法は原点復帰途上の原点検出点で CTR2 をクリアし, 減速終了時の CTR2 の符号を反転した移動量の位置決めをする方式です. したがって原点復帰完了時の CTR2 は“0”になるとは限りません.

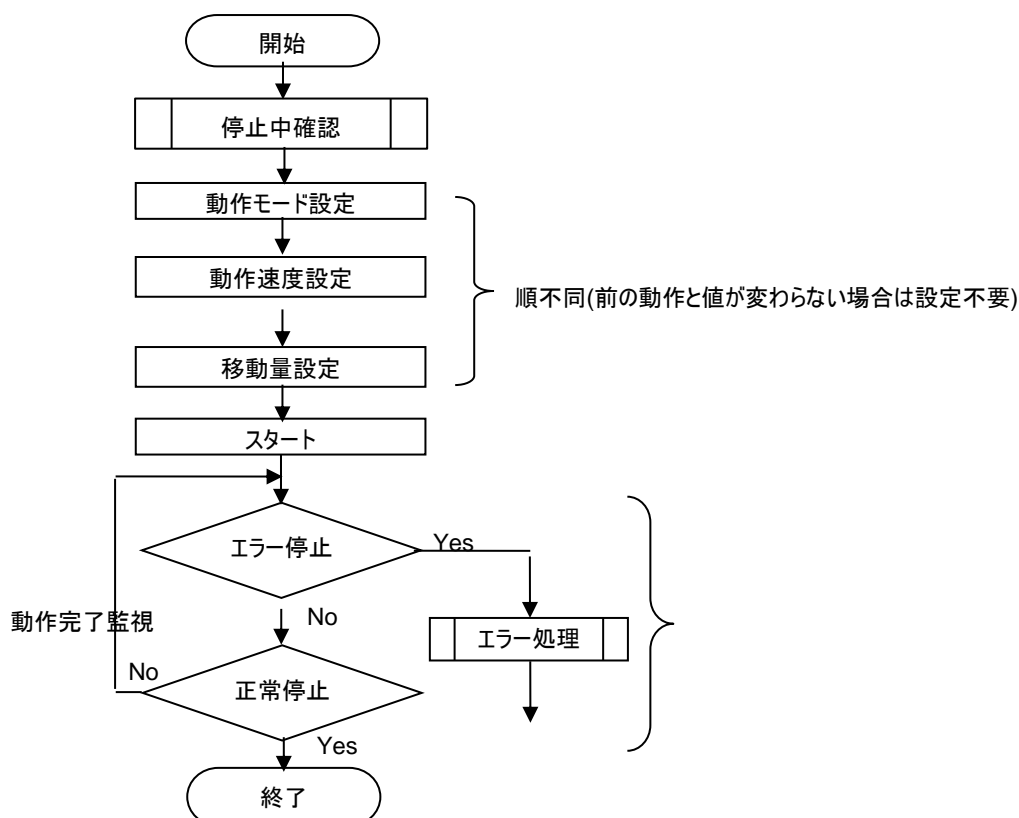
注 2. ライブラリ関数ではすべての CTR が自動的にクリアされる設定になっています.

注 3. ライブラリ関数を使用した場合の CTR2 入力

- No.9 を選択 …「指令パルス入力」
- No.10～12 を選択 …「ENC 入力」

2.3 基本の動作手順

動作の手順の基本は以下の通りです。サーボモータを制御対象としている場合はサーボオンされているものとします。



2.3.1 サーボオン/サーボリセット

サーボモータを制御対象としている場合はサーボオンしてから動作を開始します。

No.	内 容	コマンド	ドライバ関数	ライブラリ関数	備 考
1	SVON ON	18h	wCmdW	SvOn	xSVON が 0[V]
2	SVON OFF	10h	wCmdW	SvOff	xSVON が High(EXTPOW)レベル
3	SVRST ON	19h	wCmdW	SvResetOn	xSVRST が 0[V]
4	SVRST OFF	11h	wCmdW	SvResetOff	xSVRST が High(EXTPOW)レベル

表 2.3-1 サーボオン/サーボリセット

2.3.2 動作モードレジスタ(RMD)

(1) 動作モード(MOD)

ある動作をするためには動作モードを設定します。

No.	選択事項	設定箇所	使用する関数	備 考
1	動作モード(MOD)	RMD.b7-0	WritOpeMode / wReg	ライブラリ関数 WritOpeMode では MOD(RMD.b7-0)のみ設定します。

No.	動作モード分類	MOD	動作モード
1	連続送り	00h	＋方向連続送り
		08h	－方向連続送り
2	位置決め	41h	位置決め動作(移動量指定)
		42h	PCS 位置決め動作(ライブラリ関数固有)
		42h	位置決め動作(指令位置指定) ドライバ関数でのみ設定可能
		44h	指令位置 0 点復帰動作
		45h	機械位置 0 点復帰動作
		46h	＋方向 1 パルス動作
		4eh	－方向 1 パルス動作
		47h	タイマ動作
3	補間	60h	直線補間連続送り
		61h	直線補間
		62h	2 個の PCL 間の直線補間連続送り
		63h	2 個の PCL 間の直線補間
		64h	CW 方向円弧補間
		65h	CCW 方向円弧補間
4	原点復帰	10h	＋方向原点復帰動作
		18h	－方向原点復帰動作
		12h	＋方向原点拔出し
		1ah	－方向原点拔出し
		15h	＋方向原点サーチ
		1dh	－方向原点サーチ
5	ELS, SLS 動作	20h	＋ELS または＋SLS 位置まで動作
		28h	－ELS または－SLS 位置まで動作
		22h	＋ELS または＋SLS 拔出し動作
		2ah	－ELS または－SLS 拔出し動作
6	Z 相まで移動	24h	＋方向に Z 相カウント分動作
		2ch	－方向に Z 相カウント分動作
7	ハンドル送り	01h	パルス入力による連続動作
		51h	パルス入力による位置決め動作
		54h	パルス入力による指令位置 0 点復帰動作
		55h	パルス入力による機械位置 0 点復帰動作
8	JOG 送り	02h	±DR 入力による連続動作
		56h	±DR 入力による位置決め動作

表 2.3-2 動作モード(MOD)一覧

(2) 動作モードレジスタ内容

bit	名 称	説 明
7-0	MOD	動作モード(前ページ参照)
8	DLS 有効	0:DLS 無効, 1:DLS 有効
9	INPOS 有効	0:INPOS 無効, 1:INPOS 有効
10	加減速方式	0:直線加減速, 1:S 字加減速
11	CTR1 カウント禁止	1:指令位置カウンタ(CTR1) カウント禁止
12	動作完了タイミング	0:完了タイミングをパルスの周期完了とします.
13	減速開始点計算方法	0:減速開始点自動計算, 1:減速開始点手動計算
14	PCS 有効	0:PCS 無効, 1:PCS 有効
15	補間合成速度	0:補間時合成速度一定制御 OFF, 1:補間時合成速度一定制御 ON
17,16	SEQ No	MSTS.b7,6 に現在実行中の SEQ No が反映されます.
19-18	スタート条件	00: スタートコマンド書込み後即スタート. 01: STA 入力によるスタート. 10: 他軸の条件一致によるスタート. 11: 指定軸の停止によるスタート.
23-20	軸停止によるスタート時の軸指定	指定軸の停止によるスタート(RMD.b19,18=11)の場合, 軸を指定します. bit20: X(V,X1,X2,X3,X4)軸の停止 bit21: Y(W,Y1,Y2,Y3,Y4)軸の停止 bit22: Z(A,Z1,Z2,Z3,Z4)軸の停止 bit23: U(B,U1,U2,U3,U4)軸の停止 <例> 0001: X(V,X1,X2,X3,X4)軸の停止で次動作スタート 1001: X(V,X1,X2,X3,X4)軸の停止かつ U(B,U1,U2,U3,U4)軸の停止で次動作スタート
24	STP 有効	1:STP 入力により停止します.
25	異常時 STP 自動出力	1:異常停止時に STP(同時停止信号)を自動出力します.
26	FH 補正	0:ON, 1:OFF
27	円弧自動終点引込み	1:円弧自動終点引込み
28	予約	
29	MSDC	1:減速開始点自動計算時, 減速開始点=加速パルス数とする. 0:補間動作を合成速度一定制御 ON 時のみ減速開始点=加速パルス数とする. (位置決め動作の減速開始点計算は内部自動演算)
31,30	予約	

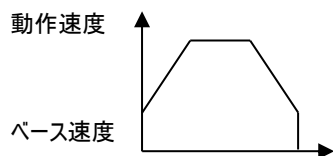
表 2.3-3 RMD の内容

2.3.3 スタートコマンド

動作をスタートさせるとともにボードの加減速パターンを決定します。

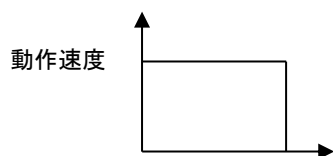
(1) 加速スタート

スタート時はベース速度で立ち上がり動作速度まで加速し、停止時は動作速度からベース速度まで減速して停止します。



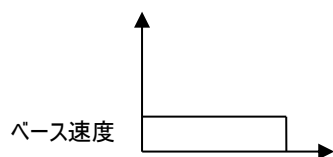
(2) FH 定速スタート

動作速度で一定速動作をします。



(3) FL 定速スタート

ベース速度で一定速動作をします。



No.	内 容	コマンド	ドライバ関数	ライブラリ関数
1	加速スタート	53h	wCmdW	AccStart
2	FH 定速スタート	51h	wCmdW	CnstStartFH
3	FL 定速スタート	50h	wCmdW	CnstStartFL

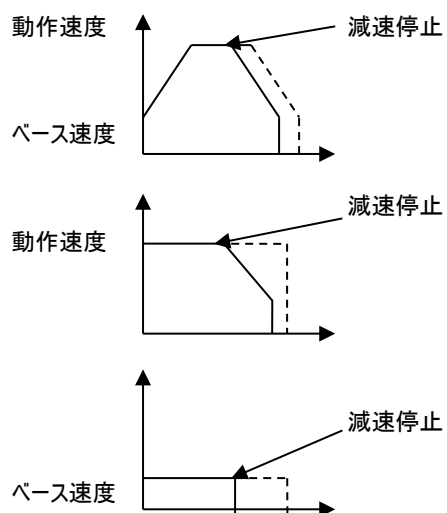
表 2.3-4 スタートコマンド

2.3.4 停止コマンド

動作を停止させます。

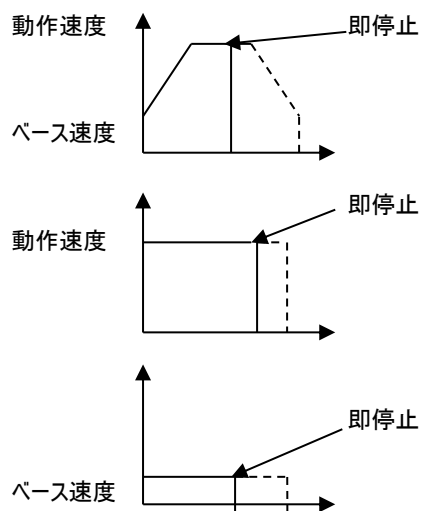
(1) 減速停止

加速スタート, FH 定速スタート時はベース速度まで減速して停止します
ベース速度定速スタート時は即停止します。



(2) 即停止

即停止します。

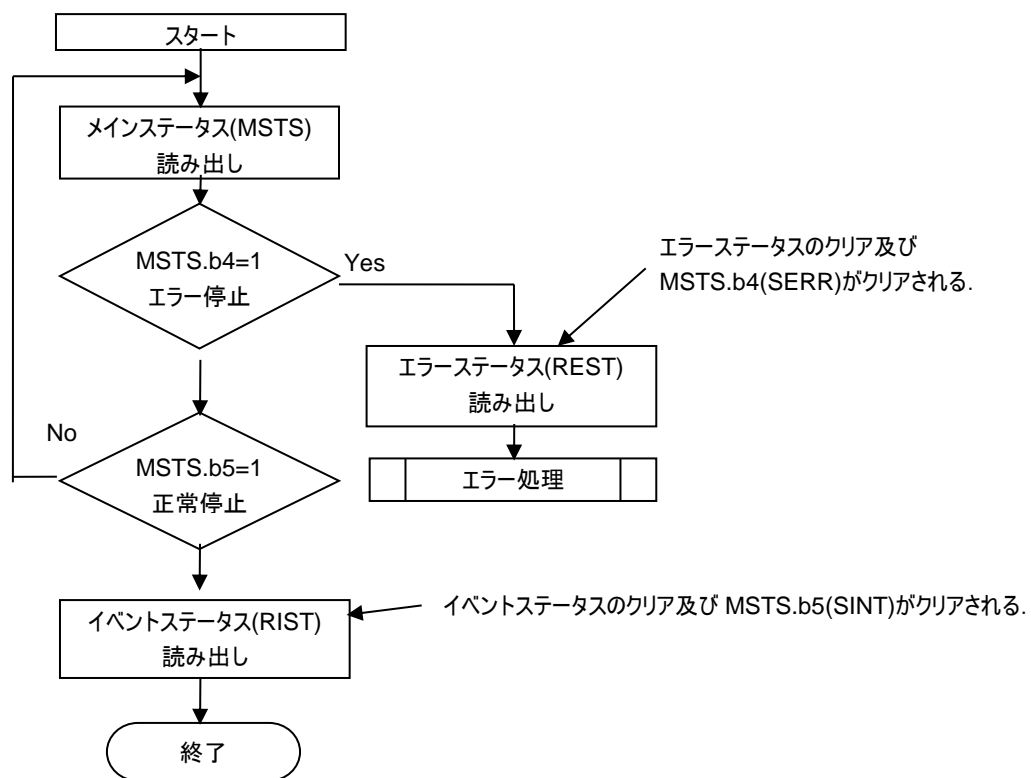


No.	内 容	コマンド	ドライバ関数	ライブラリ関数
1	減速停止	4Ah	wCmdW	DecStop
2	即停止	49h	wCmdW	QuickStop

表 2.3-5 停止コマンド

2.3.5 動作完了監視

動作の完了を確認するには動作している軸のメインステータスを読み出します。



(1) メインステータス(MSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPDF	SPRF	SEOR	SCMP5	SCMP4	SCMP3	SCMP2	SCMP1	SSC1	SSC0	SINT	SERR	SEND	0	SRUN	SSCM

bit	名 称	内 容	備 考
0	SSCM	1 = スタートコマンド書き込み済み	スタートコマンド書き込み後、即動作が始まらない場合があります。(条件付スタートのスタート保留状態など)
1	SRUN	1 = RUN 中	
2	---	予約	
3	SEND	1 = 停止中(電源投入直後は'0')	DevOpen 関数呼び出し後は SEND=1 になります。 本ビットの変化タイミングは、INPOS 制御設定により異なります。INPOS 制御の詳細は「 2.3.6 INPOS 制御 」を参照してください。
4	SERR	1 = エラーステータスに要因有	REST 読み出しで 1→0
5	SINT	1 = イベントステータスに要因有	RIST 読み出しで 1→0 イベントマスク(RIRQ)の設定が必要。 DevOpen 関数では RIRQ.b0=1(正常停止時)の設定をしています。
7,6	SSC1,0	実行中の RMD.b17,16	次動作連続実行中の動作確認などに使用
8	SCMP1	1 = CMP1 比較条件成立状態	比較条件成立中の間のみ'1'となります。 (既に通過してしまった場合は'0'となります) 条件が成立したことを監視する場合はイベントステータス(RIST)を使用します。
9	SCMP2	1 = CMP2 比較条件成立状態	
10	SCMP3	1 = CMP3 比較条件成立状態	
11	SCMP4	1 = CMP4 比較条件成立状態	
12	SCMP5	1 = CMP5 比較条件成立状態	
13	SEOR	1 = 位置オーバーライド失敗	本ステータスリードでクリア。停止中に RMV を書込んでも"1"。
14	SPRF	1 = 動作用プリレジスタフル	
15	SPDF	1 = コンパレータ用プリレジスタフル	

表 2.3-6 メインステータス(MSTS)

(2) エラーステータス(REST)

エラーステータスはマスクすることは出来ません。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ESAO	ESPO	ESIP	ESDT	0	ESSD	ESEM	ESSP	ESAL	ESML	ESPL	ESC5	ESC4	ESC3	ESC2	ESC1
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	ESPE	ESEE

bit	名 称	内 容	備 考
0	ESC1	1 = CMP1 条件成立で停止 (+SLS)	
1	ESC2	1 = CMP2 条件成立で停止 (-SLS)	
2	ESC3	1 = CMP3 条件成立で停止	
3	ESC4	1 = CMP4 条件成立で停止	
4	ESC5	1 = CMP5 条件成立で停止	
5	ESPL	1 = +ELS による停止	停止中に+ELS が ON しても 1 になりません。
6	ESML	1 = -ELS による停止	停止中に-ELS が ON しても 1 になりません。
7	ESAL	1 = サーボアラームによる停止	停止中に SVALM が ON しても 1 になりません。
8	ESSP	1 = STP 入力 ON による停止 (または同時ストップコマンド)による停止	次動作使用時はプリレジスタキャンセルコマンド (26h)を書き込んでください。
9	ESEM	1 = EMG 入力 ON による停止	非常停止コマンドによる停止または非常停止機能 を使用している場合
10	ESSD	1 = DLS 検出による減速停止	停止中に DLS が ON しても 1 になりません。
11	未定義	常に'0'	
12	ESDT	1 = 動作データが不正で停止	
13	ESIP	1 = 補間他軸の異常停止による停止	
14	ESPO	1 = パルス用バッファオーバーフローによる停止	(パルス入力×パルス通倍)>動作速度の状態が続 いた場合発生します。
15	ESAO	1 = 補間データのレンジオーバーによる停止	
17,16	ESPE ,ESEE	1 = エンコーダ A/B 相が同時に変化	停止しません。
31-18	未定義	常に'0'	

表 2.3-7 エラーステータス(REST)

(3) イベントステータス(RIST)

イベントマスク(RIRQ)の設定をすることでイベントステータスが有効となります。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISOL	ISLT	ISCL	ISC5	ISC4	ISC3	ISC2	ISC1	ISDE	ISDS	ISUE	ISUS	ISND	ISNM	ISN	ISEN
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	ISSA	ISMD	ISPD	ISSD

bit	名 称	内 容
0	ISEN	正常停止. 本ビットの変化タイミングは、INPOS 制御設定により異なります. INPOS 制御の詳細は「 2.3.6 INPOS 制御 」を参照してください.
1	ISN	次動作継続スタート.
2	ISNM	動作用プリレジスタフル→空き
3	ISND	CMP5 用プリレジスタフル→空き
4	ISUS	加速開始
5	ISUE	加速終了
6	ISDS	減速開始
7	ISDE	減速終了
12-8	ISCn	CMPn 比較条件成立
13	ISCL	CLR 入力によるカウンタクリア時
14	ISLT	LTC 入力によるカウンタラッチ時
15	ISOL	OLS 入力によるカウンタラッチ時
16	ISSD	DLS 信号 OFF→ON
17	ISPD	+DR 信号 OFF→ON
18	ISMD	−DR 信号 OFF→ON
19	ISSA	STA 信号 OFF→ON
31-20	未定義	常に'0'

表 2.3-8 イベントステータス(RIST)

(4) イベントマスクレジスタ(RIRQ)

イベントステータス(RIST)に対するイベントを設定します。各ビットを 1 にするとイベントステータス(RIST)に反映されます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IROL	IRLT	IRCL	IRC5	IRC4	IRC3	IRC2	IRC1	IRDE	IRDS	IRUE	IRUS	IRND	IRNM	IRN	IREN
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	IRSA	IRDR	IRSD

bit	名 称	内 容
0	IREN	動作完了報告
1	IRN	次動作継続スタート報告
2	IRNM	動作用プリレジスタフル→空き報告
3	IRND	CMP5 用プリレジ書込み可能報告
4	IRUS	加速開始報告
5	IRUE	加速終了報告
6	IRDS	減速開始報告
7	IRDE	減速終了報告
12-8	IRCn	CMPn 比較条件成立報告
13	IRCL	CLR 入力によるカウンタクリア報告
14	IRLT	LTC 入力によるカウンタラッチ報告
15	IROL	OLS 信号入力時にカウンタラッチ報告
16	IRSD	DLS 信号 OFF→ON
17	IRDR	±DR 信号 OFF→ON
18	IRSA	STA 信号 OFF→ON
19-31	未定義	常に'0'を設定してください

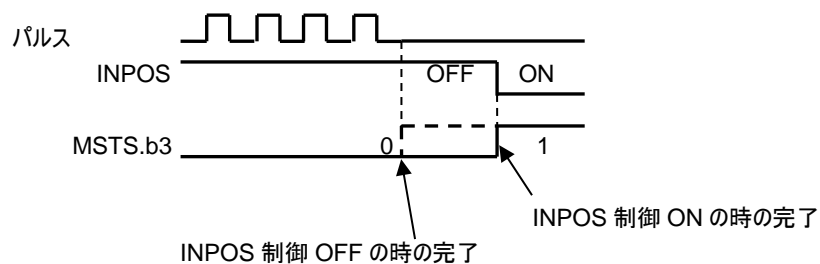
表 2.3-9 イベントマスクレジスタ

(5) 対応関数

No.	内 容	使用する関数
1	メインステータス読み出し	ReadMainSts / rMstsW
2	エラーステータス読み出し	ReadErrorSts / rReg
3	イベントステータス読み出し	ReadEventSts / rReg
4	イベントマスクレジスタ	SetEventMask / wReg

2.3.6 INPOS 制御

動作完了判断を指定したパルス数出力完了後のサーボドライバからの位置決め完了信号(INPOS)検出で行う制御です。
この設定を行うと下図のように最終パルスの出力完了後の INPOS 信号 ON で動作完了とみなします。



No	項目	設定箇所	使用する関数
1	INPOS 制御	RMD.b9 有効:1, 無効:0	SetInpos / wReg

2.3.7 指令速度モニタ

指令速度[pps]は RSPD.b15-0 に速度倍率を掛けた結果となります。

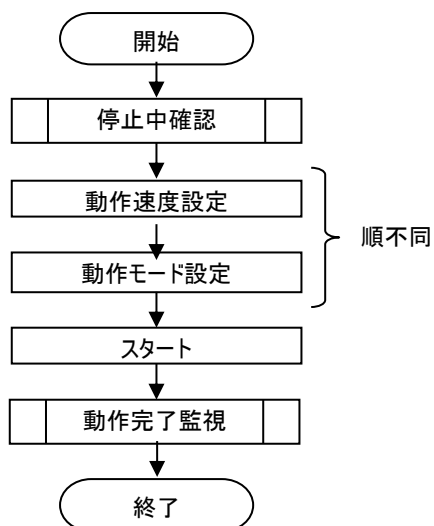
No.	内 容	使用する関数
1	指令速度読み出し	ReadSpd / rReg

2.4 連続送り

連続送り→指定した速さの動作。停止命令するまでまたは動作方向の ELS(SLS)まで動作します。

2.4.1 動作手順

ベース速度、速度倍率、加速レート、減速レートは初期化時に設定します。



No.	設定項目	選択事項	対応関数
1	動作モード	+連続送り:00h / -連続送り:08h	WritOpeMode
2	動作速度	RFH(1~65535) × 速度倍率 [pps]	WritFHSpd
3	スタートコマンド	加速スタート / FH 定速スタート / FL 定速スタート	AccStart / CnstStartFH / CnstStartFL
4	停止コマンド	減速停止 / 即停止	DecStop / QuickStop

2.5 原点復帰

装置上の位置とソフトウェア上の位置を一致させるための動作です。原点復帰完了で座標系が確定されます。

2.5.1 動作手順

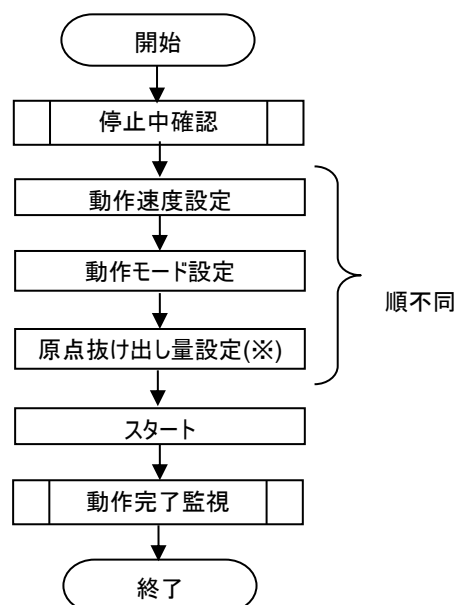
ベース速度、速度倍率、加速レート、減速レート、原点復帰方式(ORGmode)と原点センサ(OLS)の入力極性、Z 相カウント回数は初期化時に設定します。原点復帰のシーケンスは原点復帰方式(ORGmode)で設定します。

原点復帰方式の詳細は「[2.2.10 原点復帰方式\(ORGmode\)](#)」を参照してください。

原点復帰を特異点(OLS 上または原点復帰方向 ELSとOLSの間)からスタートする場合は原点サーチを使用します。

原点サーチについては「[2.5.2 原点サーチについて](#)」を参照してください。

但し ELS を使用する原点復帰方式(ORGmode=6,7,8)では動作モードは原点復帰に設定し、原点抜け出し量の設定は不要です。



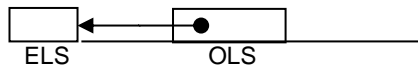
※. 原点サーチの場合は原点抜け出し量を設定します。

No.	設定項目	選択事項	対応関数
1	動作モード	+原点復帰:10h / -原点復帰:18h / +原点サーチ:15h / -原点サーチ:1Dh	WritOpeMode / wReg
2	動作速度	RFH(1~65535) × 速度倍率 [pps]	WritFHSpd / wReg
3	原点抜け出し量	RMV[pulse]	WritPos / wReg
4	スタートコマンド	加速スタート / FH 定速スタート / FL 定速スタート	AccStart / CnstStartFH / CnstStartFL / wCmdW
5	停止コマンド	減速停止 / 即停止	DecStop / QuickStop / wCmdW

2.5.2 原点サーチについて

原点復帰を特異点(OLS 上または原点復帰方向 ELS と OLS の間)からスタートする場合は原点復帰が正常に完了しません。

- OLS が ON
ELS 検出による停止



- OLS と ELS の中間
ELS 検出による停止

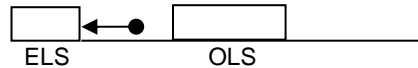
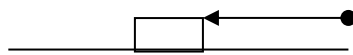


図 2.5-1 特異点からの原点復帰動作

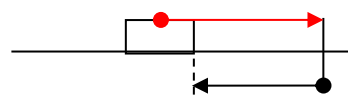
このような特異点から原点復帰を行う場合は原点サーチを使用します。

原点サーチは開始位置により、下図の動作となります。

- OLS が OFF
RENV3:ORGmode 設定に従った原点復帰動作



- OLS が ON
原点抜き出し動作→原点復帰動作



- OLS と ELS の中間
ELS 検出まで進む→ELS 検出で反転→OLS 検出で停止→原点抜き出し動作→原点復帰動作

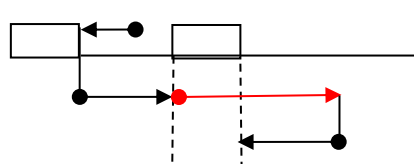


図 2.5-2 原点サーチ

この時の原点抜き出し動作(赤線の動作)は、OLS を検出しなくなるまで RMV に設定した移動量の位置決めを繰り返します。

したがって原点抜き出し量は OLS(原点センサ)の幅より若干大きい値を設定します。

2.5.3 座標系

装置の可動部の位置の管理は、通常原点復帰完了位置を原点とする座標系上の位置で管理します。

(1) カウンタ

現在座標はカウンタで管理します。各軸に 4 個のカウンタがあります。

	CTR1	CTR2	CTR3	CTR4
用 途	指令位置カウント	機械位置カウント	脱調検出	汎 用
機 能	UP/DOWN CTR	UP/DOWN CTR	偏差カウンタ	UP/DOWN CTR
ビット長	28	28	16	28
カウント入力の選択	指令パルス	指令パルス	指令パルス	指令パルス
	—	エンコーダ	エンコーダ	エンコーダ
	—	パルス	パルス	パルス
	—	—	—	カウント・CLK(注)
入力信号形式	—	位相差, UP/DOWN パルス	位相差, UP/DOWN パルス	位相差, UP/DOWN パルス

注. カウント入力に「カウント・CLK」を選択した場合は内部 CLK 9.8304MHz(約 0.102μsec)をカウントする。

表 2.5-1 カウンタ種別

(2) レジスタ設定箇所と対応関数

No.	項 目	設定箇所	使用する関数
1	カウンタ読み出し		ReadCtr / rReg
2	CTR2 入力の選択	RENV3.b9,8 エンコーダ:00, 指令パルス:01, パルス:10	wReg
3	CTR3 入力の選択	RENV3.b11,10 指令パルスとエンコーダ入力の偏差:00, 指令パルスとパルス入力の偏差:01	wReg
4	CTR4 入力の選択	RENV3.b13,12 指令パルス:00, エンコーダ:01, パルス:10, 9830400Hz クロックカウント:11	wReg
5	エンコーダ入力カウント形式	RENV2.b21,20 1 通倍:00, 2 通倍:01, 4 通倍:10, UP/DOWN:11	wReg
6	エンコーダ入力カウント方向	RENV2.b22 A 相進相:0, B 相進相:1	wReg
7	パルス入力カウント形式	RENV2.b25,24 1 通倍:00, 2 通倍:01, 4 通倍:10, UP/DOWN:11	wReg
8	パルス入力カウント方向	RENV2.b26 A 相進相:0, B 相進相:1	wReg

(3) ソフトリミット

直線軸ではコンパレータ 1(CMP1)とコンパレータ 2(CMP2)を利用して、ソフトリミット(SLS)機能が行えます。
 CMP1 と CMP2 の比較カウンタには、CTR1(指令位置)を選択します。
 CMP1 を(+)側リミット値、CMP2 を(-)側リミット値として使用し、比較結果と動作方向とから停止管理を行います。
 通常は原点復帰が完了し座標系確定後に設定します。
 SLS 検出時は即停止または減速停止します。
 また、動作スタート時に SLS ON の場合は、SLS から抜け出す方向にしかスタートできません。

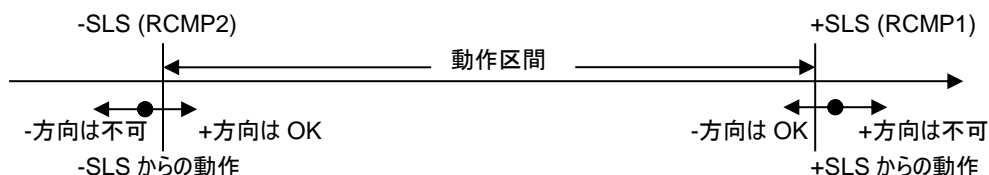


図 2.5-3 ソフトリミット

No.	設定事項	設定箇所	使用する関数
1	コンパレータ 1 の比較カウンタ	RENV4.b1,0=00(CTR1)	
2	コンパレータ 1 の比較方法	RENV4.b4-2=110(CMP1 を+SLS で使用)	SetSls
3	+SLS ON 時の停止方法	RENV4.b6,5 即停止:01, 減速停止:10	SetSls
4	+SLS 位置	RCMP1	SetSls
5	コンパレータ 2 の比較カウンタ	RENV4.b9,8=00(CTR1)	
6	コンパレータ 2 の比較方法	RENV4.b12-10=110(CMP2 を-SLS で使用)	SetSls
7	-SLS ON 時の停止方法	RENV4.b14,13 即停止:01, 減速停止:10	SetSls
8	-SLS 位置	RCMP2	SetSls

注. SetSls では±SLS での停止方法を個別に設定できません。wReg

(4) リングカウンタ

回転軸の座標管理用にリングカウンタが使用できます。
 環境設定 4 レジスタ(RENV4)、コンパレータ 1(CMP1)、コンパレータ 2(CMP2)の設定で CTR1・CTR2 が回転軸カウンタとして利用できます。

	設定事項	設定箇所	使用する関数
1	CTR1 をリングカウンタとする	RENV4.b7-0=80h(1000 0000)	wReg
2	CTR1 リングカウント数	RCMP1=(1 回転あたりのパルス数-1)	wReg
3	CTR2 をリングカウンタとする	RENV4.b15-8=81h(1000 0001)	wReg
4	CTR2 リングカウント数	RCMP2=(1 回転あたりのパルス数-1)	wReg

位置決め動作時の移動量は 0～RCMP の範囲外でも動作します。

例. 1 回転パルス数が 3,600 パルスの場合、RCMP には 3599 を設定し、カウンタ値は 0～3,599 で動作します。
 この時 RMV=7200 の位置決めを行うと回転軸は 2 回転し、カウンタ値はスタート前と同じ値になります。



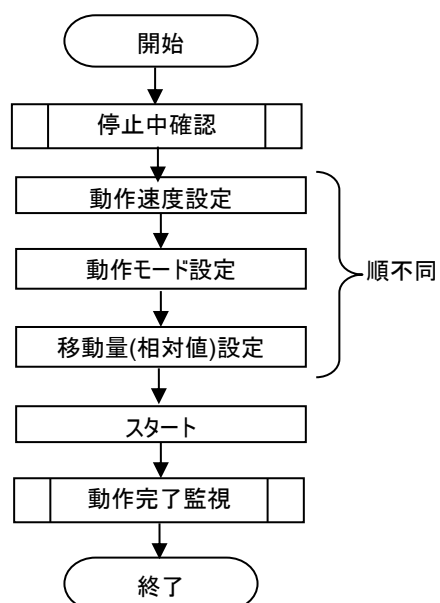
注 意

リングカウンタとして設定する前に、CTRn を 0～(RCMPn 設定値)の値にしてください。

2.6 位置決め

2.6.1 動作手順

ベース速度, 速度倍率, 加速レート, 減速レートは初期化時に設定します。



No.	設定項目	選択事項	対応関数
1	動作モード	41h	WritOpeMode
2	動作速度	RFH(1~65535)	WritFHSpd
3	移動量(相対値)	RMV[pulse]	WritPos
4	スタートコマンド	加速スタート / FH 定速スタート / FL 定速スタート	AccStart / CnstStartFH / CnstStartFL
5	停止コマンド	減速停止 / 即停止	DecStop / QuickStop

2.7 タイマ動作

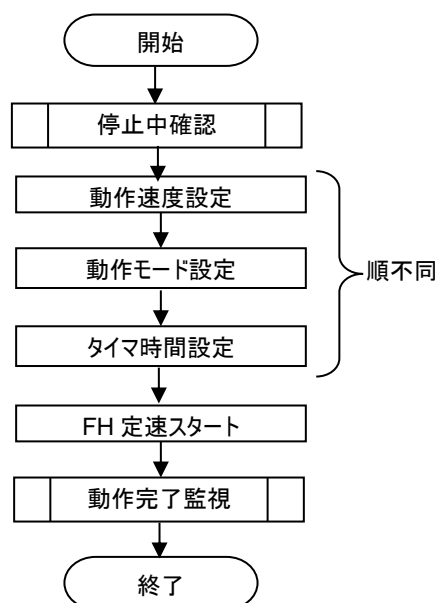
内部動作時間をタイマとして使用するモードです。動作中はパルス出力しません。また CTR1 も動作しません。SVALM, EMG 入力は有効です。STP 入力は RMD の設定により有効になります。INPOSITION は常に無効です。

内部動作時間[s]= RMV／動作速度[pps]となります。動作速度[pps] = RFH × { 300 / (RMG+1) }

例えば動作速度を 1000pps(RMG=299, RFH=1000)に設定した場合、

RMV=1 の場合 1msec, RMV=10 の場合 10msec となります。

2.7.1 動作手順



No.	設定項目	選択事項	対応関数
1	動作モード	47h	WritOpeMode
2	速度倍率	RMG(2~4095)	WritFHSpd
3	動作速度	RFH(1~65535)	WritFHSpd
4	タイマ時間	RMV[pulse]	WritPos
5	スタートコマンド	FH 定速スタート	CnstStartFH
6	停止コマンド	即停止	QuickStop

2.8 直線補間

ここでは同一 PCL(X-U, V-W, X1-U1, X2-U2, X3-U3, X4-U4 の中での組み合わせ)の直線補間について説明します。

2.8.1 動作手順

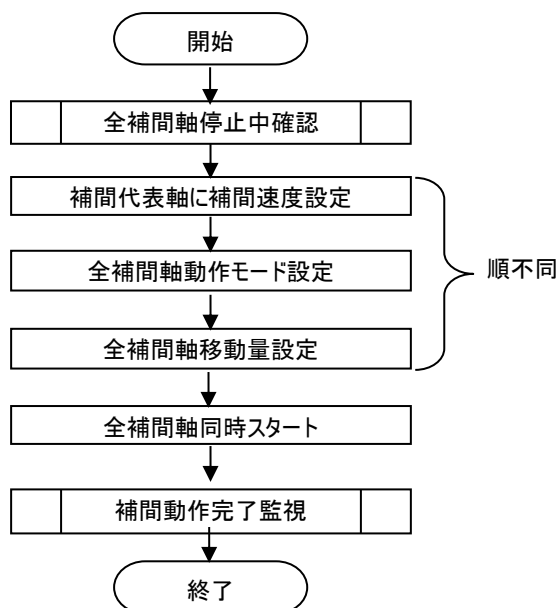
通常、ベース速度、速度倍率、加速レート、減速レートは初期化時に設定します。

但し、加速レート(RUR)と減速レート(RDR)が異なる場合は RDR=0 にします。(直線補間時は加速時間=減速時間のみ)

動作速度は補間代表軸(補間軸の内 X(V, X1, X2, X3, X4)軸に近い軸)に設定します。

移動量及び動作モードは全ての補間軸に設定します。

補間動作時に補間軸のいずれかが異常停止した場合、全ての補間軸が停止します。



補間動作の完了は補間代表軸のメインステータス(MSTS)を監視します。

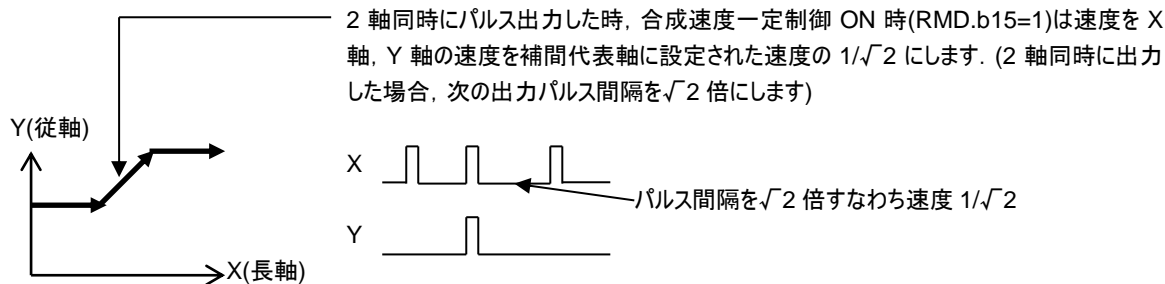
MSTS.b4(SERR)検出時には全ての補間軸のエラーステータス(REST), MSTS.b5(SINT)検出時には全ての補間軸のイベントステータス(RIST)を読み出します。

補間途中で停止する場合は必ず**全補間軸同時停止**を行います。

No.	設定項目	選択事項	対応関数
1	動作モード	61h	WritOpeMode
2	動作速度	RFH(1~65535) × 速度倍率 [pps]	WritFHSpd
3	移動量(相対値)	RMV[pulse]	WritLine
4	スタートコマンド	加速スタート/FH 定速スタート/FL 定速スタート	AccStart / CnstStartFH / CnstStartFL
5	停止コマンド	減速停止 / 即停止	DecStop / QuickStop
6	合成速度一定制御	RMD.b15=1	wReg

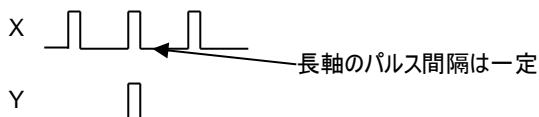
(1) 合成速度一定制御

2 軸同時にパルス出力した時に各軸のパルス速度を $1/\sqrt{2}$, 3 軸同時にパルス出力した時に各軸のパルス速度を $1/\sqrt{3}$ にすることで進行方向の速度を一定にする制御です。



合成速度一定制御 OFF 時(RMD.b15=0)補間代表軸に設定された速度が長軸の速度となります。

従って 2 軸同時にパルス出力した時, 進行方向の速度は設定した速度の $\sqrt{2}$ 倍になります。



注 意

減速開始点自動計算時(RMD.b13=0)の場合は RDR=0(加速時間=減速時間)にしてください。

速度モニタ(RSPD)は補間代表軸のみが有効となります。

合成速度一定制御 ON(RMD.b15=1)時は合成速度, 合成速度一定制御 OFF(RMD.b15=0)時は長軸速度を表します。

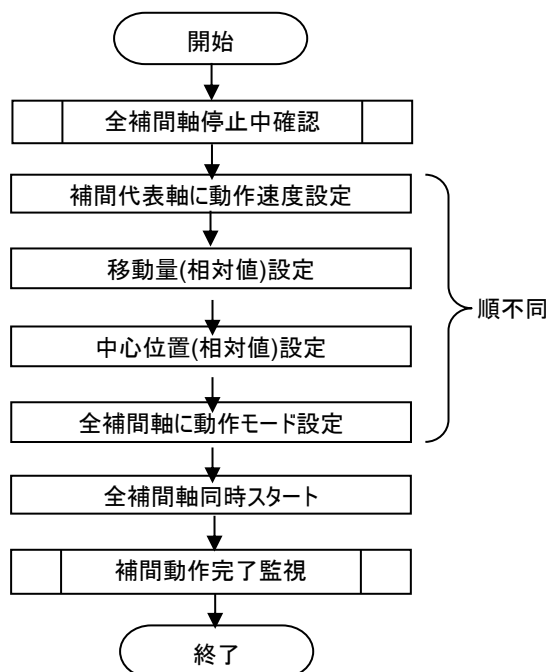
2.9 円弧補間

2.9.1 動作手順

速度倍率は初期化時に設定します。

動作速度は補間代表軸(補間軸の内 X(V, X1, X2, X3, X4)軸に近い軸)に設定します。

円弧補間は原則的に定速動作のみです。



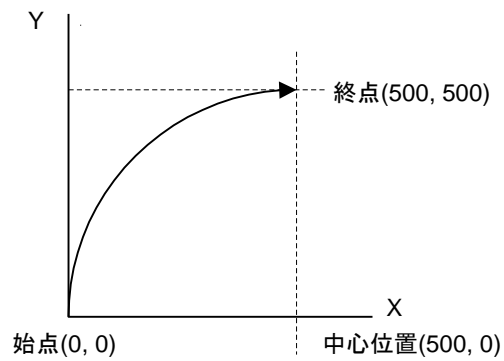
補間動作の完了は補間代表軸のメインステータス(MSTS)を監視します。

MSTS.b4(SERR)検出時には全ての補間軸のエラーステータス(REST), MSTS.b5(SINT)検出時は全ての補間軸のイベントステータス(RIST)を読み出します。

補間途中で停止する場合は必ず**全補間軸同時停止**を行います。

No.	設定項目	選択事項	対応関数
1	動作モード	CW 円弧補間:64h/CCW 円弧補間:65h	WritOpeMode
2	動作速度	RFH(1~65535) × 速度倍率 [pps]	WritFHSpd
3	移動量(相対値)	RMV[pulse]	WritCircl
4	中心位置(相対値)	RIP[pulse]	WritCircl
5	スタートコマンド	FH 定速スタート / FL 定速スタート	CnstStartFH / CnstStartFL
6	停止コマンド	即停止	QuickStop
7	合成速度一定制御	RMD.b15 1:有効, 0:無効	wReg
8	終点自動引き込み	RMD.b27 1:有効, 0:無効	wReg

<円弧補間設定例>



上記の円弧補間の場合

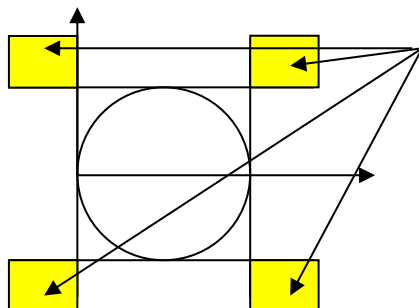
No		X 軸	Y 軸	備考
1	移動量(RMV)	500	500	
2	中心位置(RIP)	500	0	
3	動作モード(RMD.b7-0)	64h	64h	CW 方向円弧補間



注 意

終点象限で片方の軸が終点に到達した時点で円弧補間は完了します。

円弧補間動作完了時に指定した終点に到達していない場合に指定した終点まで移動させたい場合は、終点自動引き込みを有効(RMD.b27=1)にします。



黄色の部分を終点座標に設定した場合は停止コマンドを書き込むまで完了しません。

3. 特殊な運用

3.1 非対称加減速

3.1.1 位置決め動作時

(1) 減速時間 \leq (加速時間 $\times 2$)

減速開始点自動計算(RMD.b13=0)が可能です。

	設定事項	設定箇所	使用する関数
1	減速開始点自動計算	動作モード(RMD.b13=0)	SetAutoDec
2	減速時間	減速レート(RDR)	SetDecRate / CalAccRate

減速レート(RDR)の計算方法は「[2.2.3 速度と加速度 \(2\)加減速](#)」を参照。

(2) 減速時間 $>$ (加速時間 $\times 2$)

減速開始点を手動計算(RMD.b13=1)にします。

	設定事項	設定箇所	使用する関数
1	減速開始点手動計算	動作モード(RMD.b13=1)	SetAutoDec
2	減速時間	減速レート(RDR)	SetDecRate / CalAccRate
3	減速開始点	減速開始点(RDP)	SetDecPoint

減速レート(RDR)の計算方法は「[2.2.3 速度と加速度 \(2\)加減速](#)」を参照。

減速開始点(RDP)を計算方法は「[2.2.3 速度と加速度 \(3\)減速開始点](#)」を参照。

3.1.2 直線補間時

全ての補間軸の減速開始点を手動計算(RMD.b13=1)にします。

合成速度一定制御を OFF に設定し、長軸速度を指定するモードにします。

減速レートと減速開始点の計算は補間代表軸の速度パラメータ(長軸速度)を使用して計算します。

減速開始点を全ての補間軸の RDP に設定します。

	設定事項	設定箇所	使用する関数
1	減速開始点手動計算	動作モード(RMD.b13=1)	SetAutoDec
2	減速時間	減速レート(RDR)	SetDecRate / CalAccRate
3	合成速度一定制御 OFF		
4	減速開始点	減速開始点(RDP)	SetDecPoint

減速レート(RDR)の計算方法は「[2.2.5 速度と加速度 \(2\)加減速](#)」を参照。

減速開始点(RDP)を計算方法は「[2.2.5 速度と加速度 \(3\)減速開始点](#)」を参照。

3.2 次動作連続実行

動作用プリレジスタ(2nd プリレジスタ, 1st プリレジスタ)を切れ目なく使うことを「次動作連続実行」といいます。

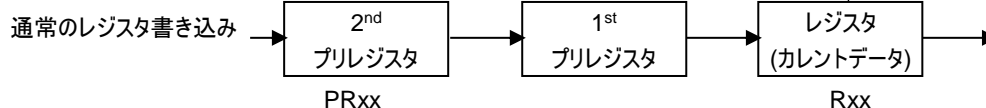
3.2.1 動作用プリレジスタ

プリレジスタとは動作中に次の動作用データを予約するレジスタです。

RMV, RFL, RFH, RUR, RDR, RMG, RDP, RMD, RIP, RUS, RDS, RCI とスタートコマンドにはプリレジスタがあります。

位置のオーバーライド

速度オーバーライド



停止中にプリレジスタに書き込まれたデータはシフトされカレントデータとして動作します。

動作中に書き込まれたデータは次の動作の予約となり、カレントデータの動作終了で自動的に動作が開始されます。

各レジスタはスタートコマンドの書き込みで確定します。

動作用プリレジスタのデータは動作用プリレジスタキャンセル, 減速停止, 即停止の各コマンドでキャンセルされます。

また, エラー停止でもキャンセルされます。

3.2.2 動作用プリレジスタ制御コマンド

No.	コマンド	略 称	CMD	機 能	使用する関数
1	動作用プリレジスタキャンセル	PREGCAN	26h	動作プリレジの内容をキャンセル	wCmdW
2	動作用プリレジスタシフト	PREGSFT	2Bh	動作プリレジの内容をシフト	wCmdW

3.2.3 プリレジスタの状態変化

動作用プリレジスタへの書き込み、スタートコマンドの書き込み、動作完了により記憶状態、レジスタの内容がどのように変化するかPRMVレジスタを例にして説明します。

(1) 停止状態でPRMVに"1000"を書き込む

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	1000	未確定	00	0
(1stプリレジスタ)	1000	未確定		
RMV(レジスタ)	1000	未確定		

(2) スタートコマンド書き込みにより動作開始

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	1000	未確定	01	0
(1stプリレジスタ)	1000	未確定		
RMV(レジスタ)	1000	確定		

(3) 動作中に次動作データPRMV=-5000を書き込む(前回と同じ内容の時は省略できます)

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	-5000	未確定	01	0
(1stプリレジスタ)	-5000	未確定		
RMV(レジスタ)	1000	確定		

(4) 次動作用スタートコマンドを書き込む(1stが確定)

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	-5000	未確定	10	0
(1stプリレジスタ)	-5000	確定		
RMV(レジスタ)	1000	確定		

(5) 動作中に次々動作データPRMV=3000を書き込む(前回と同じ内容の時は省略できます)

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	3000	未確定	10	0
(1stプリレジスタ)	-5000	確定		
RMV(レジスタ)	1000	確定		

(6) 次々動作用スタートコマンドを書き込む(2ndが確定) MSTS.b14=1 (プリレジスタ満杯状態)

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	3000	確定	11	1
(1stプリレジスタ)	-5000	確定		
RMV(レジスタ)	1000	確定		

(7) 最初の動作完了 MSTS.b14=0 (2ndが未確定), MSTS.b3=1, MSTS.b0=1

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	3000	未確定	10	0
(1stプリレジスタ)	3000	確定		
RMV(レジスタ)	-5000	確定		

(8) 次動作完了 MSTS.b14=0(1st,2ndが未確定), MSTS.b3=1, MSTS.b0=1

	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	3000	未確定	01	0
(1stプリレジスタ)	3000	未確定		
RMV(レジスタ)	3000	確定		

(9) 次々動作完了 MSTS.b14=0 (全レジスタが未確定), MSTS.b3=1, MSTS.b0=0

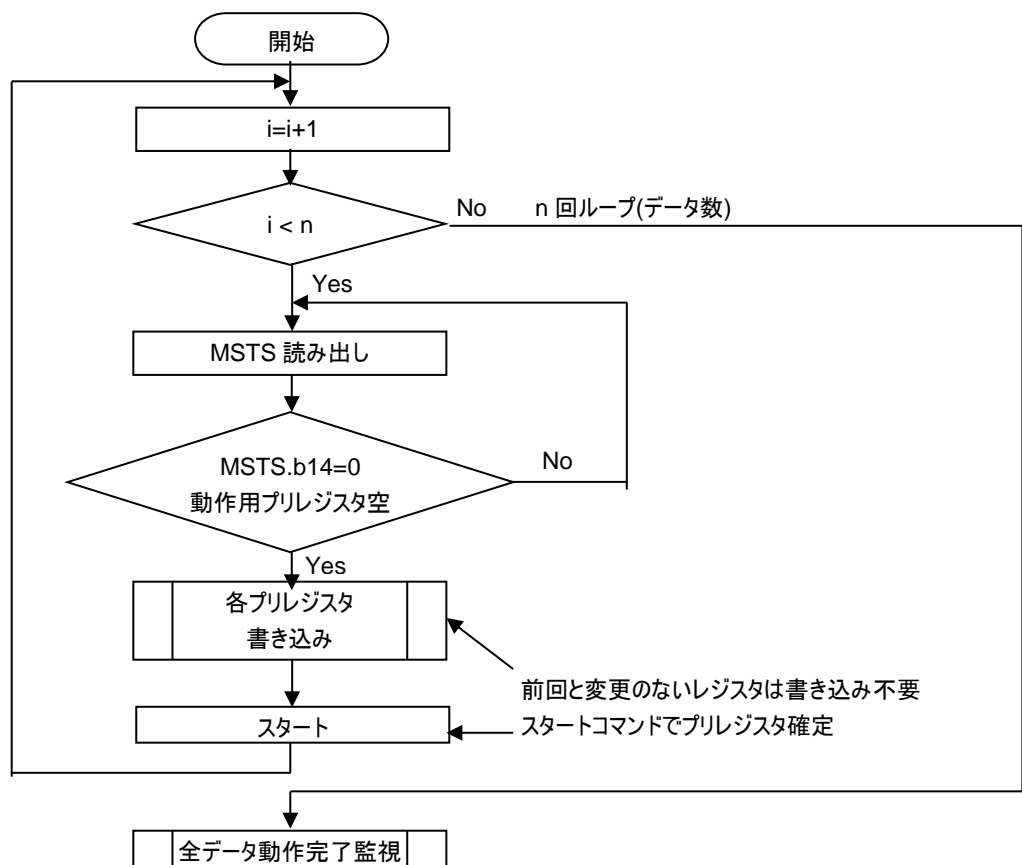
	内 容	記憶状態	RSTS.b21,20	MSTS.b14
PRMV(2ndプリレジスタ)	3000	未確定	00	0
(1stプリレジスタ)	3000	未確定		
RMV(レジスタ)	3000	未確定		

3.2.4 手順

MSTS をチェックしながら、以下のように運用します。

(1) プリレジスタへの書き込み

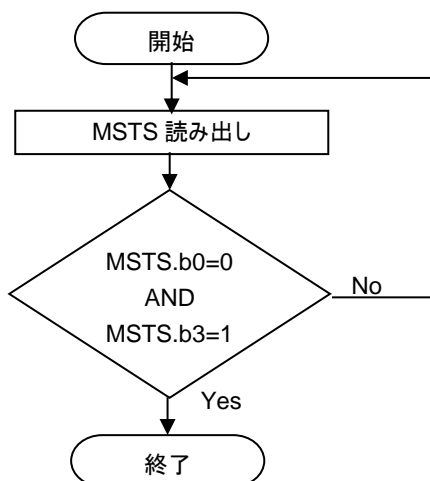
MSTS.b14=1(動作用プリレジスタフル)を検出したら、b14=0(動作用プリレジスタ空)になるまで待ち、b14=0 になったら、次のデータをプリレジスタに書き、スタートコマンドを書きます。以上の処理を繰り返します。



(2) 動作の完了監視

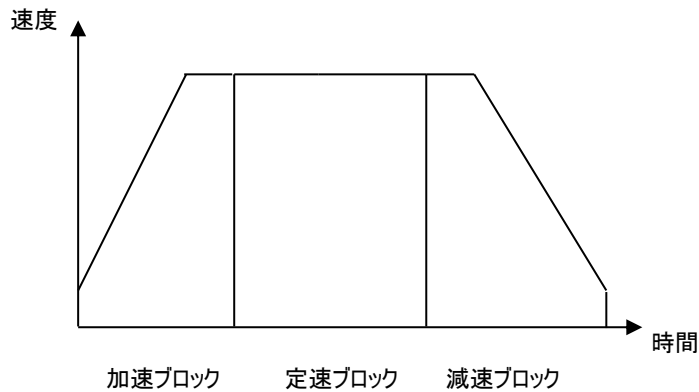
全てのデータの書き込みが終了したら、全ての動作が完了するのを待ちます。

MSTS.b0=0(スタートコマンドが書込まれていない), かつ MSTS.b3=1(停止中)の状態になるまで MSTS を監視します。



3.2.5 加速ブロック, 定速ブロック, 減速ブロック

次動作連続実行時に動作の初めに加速を追加する場合は「加速ブロック」、動作の最後に減速を追加する場合は「減速ブロック」を使用します。



(1) 加速ブロック作成方法

- 減速開始点手動計算(RMD.b13=1)
- 減速開始点(RDP=0)
- 加速スタート(53h), AccStart

(2) 定速ブロック作成方法

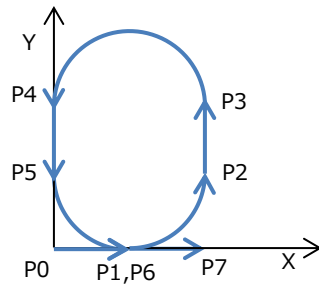
- FH 定速スタート(51h), CnstStartFH

(3) 減速ブロック作成方法

- 減速開始点手動計算(RMD.b13=1)
- RFL,RFH,RDR,RMG から計算し減速開始点(RDP)を設定します。(計算方法は「[2.2.5 速度と加速度](#)」参照)
- FH 定速スタート後減速停止(52h), CnstStartByDec

3.2.6 次動作連続実行例

下図の様な補間動作を、次動作連続実行で行います。(HPCI-CPD532, 534, 574N, 578N, 508 等の例)
速度等の設定は、ベース速度 500pps(RFL=500), 動作速度 5000pps(RFH=5000), 速度倍率 1 倍 (RMG=299), 加減速時間約 0.1sec(RUR=108, RDR=0), 直線加減速とします。



	位置(X,Y)
P0	(0,0)
P1	(1000,0)
P2	(2000,1000)
P3	(2000,2000)
P4	(0,2000)
P5	(0,1000)
P6	(1000,0)
P7	(2000,0)

	XYPRMD	XRMV	YRMV	XRIP	YRIP	XRDP	備考
P0→P1	0800A061h	1000	0	0	0	0	直線補間(加速ブロック)
P1→P2	0800A065h	1000	1000	0	1000	0	CCW 円弧補間(定速ブロック)
P2→P3	0800A061h	0	1000	0	0	0	直線補間(定速ブロック)
P3→P4	0800A065h	-2000	0	-1000	0	0	CCW 円弧補間(定速ブロック)
P4→P5	0800A061h	0	-1000	0	0	0	直線補間(定速ブロック)
P5→P6	0800A065h	1000	-1000	1000	0	0	CCW 円弧補間(定速ブロック)
P6→P7	0800A061h	1000	0	0	0	274(※)	直線補間(減速ブロック)

※.減速開始点の計算(RDR=0 の場合は RUR=RDR として計算します)

$$\text{減速開始点[パルス]} = \frac{(\text{RFH}^2 - \text{RFL}^2) \times (\text{RDR} + 1)}{(\text{RMG} + 1) \times 32768}$$

[C 言語記述例]

```
DWORD h; //デバイスハンドル
```

```
// 連続実行データの構造体(X 軸: 補間代表軸)
```

```
typedef struct {
```

```
    long    rmd; //X,Y 動作モード
```

```
    long    xmv; //X 移動量
```

```
    long    ymv; //Y 移動量
```

```
    long    xcn; //X 中心
```

```
    long    ycn; //Y 中心
```

```
    long    rdp; //X 減速開始点
```

```
    unsigned short cmd; //X スタートコマンド
```

```
} DATABLK;
```

```
// 設定データ
```

```
DATABLK dt_blk [7] = {
```

```
//動作モード, X 移動量, Y 移動量, X 中心, Y 中心, 減速開始点, コマンド
```

```
{0x800A061, 1000, 0, 0, 0, 0, 0x353} //dt_blk[0] 直線補間加速スタート
```

```
{0x800A065, 1000, 1000, 0, 1000, 0, 0x351} //dt_blk[1] CCW 円弧補間 FH 定速スタート
```

```
{0x800A061, 0, 1000, 0, 0, 0, 0x351} //dt_blk[2] 直線補間 FH 定速スタート
```

```
{0x800A065, -2000, 0, -1000, 0, 0, 0x351} //dt_blk[3] CCW 円弧補間 FH 定速スタート
```

```
{0x800A061, 0, -1000, 0, 0, 0, 0x351} //dt_blk[4] 直線補間 FH 定速スタート
```

```
{0x800A065, 1000, -1000, 1000, 0, 0, 0x351} //dt_blk[5] CCW 円弧補間 FH 定速スタート
```

```
{0x800A061, 1000, 0, 0, 0, 274, 0x352} //dt_blk[6] 直線補間加速スタート後減速停止
```

```
};
```

次ページに続く

```

short          ptr = 0;      //データポインタ
unsigned short sts[2];      //メインステータス, サブステータス
unsigned long  ersts;       //エラーステータス,
unsigned long  evsts;       //イベントステータス

// 動作用プリレジスタフル(メインステータスの bit14)を監視しながらデータ設定
while (ptr<7) {             // ptr が 7 未満の間ループ
    //動作用プリレジスタフル(メインステータスの bit14)を監視
    while (1) {
        cp530_rStsW(h, 0, sts);      //ステータス読み込み
        if(0x10==(sts[0] & 0x10)) {    //エラー停止
            //エラー停止時の処理はアプリケーションに合わせて適宜処理してください.
            //以下は一例です.
            cp530_rReg (h, 0, 0xf2, &ersts);    //X:エラーステータス読み込み
            cp530_rReg (h, 1, 0xf2, &ersts);    //Y:エラーステータス読み込み
            return;                          //処理キャンセル(終了)
        }
        if(0==(sts[0] & 0x4000)) {
            break;                          //動作用プリレジスタフルでなければループを抜ける
        }
    }
    // データ書き込み
    cp530_wReg (h, 0, 0x87, dt_blk[ptr].rmd);    //X:動作モード
    cp530_wReg (h, 1, 0x87, dt_blk [ptr].rmd);    //Y:
    cp530_wReg (h, 0, 0x80, dt_blk[ptr].xmv);    //X:移動量
    cp530_wReg (h, 1, 0x80, dt_blk [ptr].ymv);    //Y:
    cp530_wReg (h, 0, 0x88, dt_blk[ptr].xcn);    //X:中心位置
    cp530_wReg (h, 1, 0x88, dt_blk [ptr].ycn);    //Y:
    cp530_wReg (h, 0, 0x86, dt_blk[ptr].rdp);    //X:減速開始点
    cp530_wCmdW(h, 0, dt_blk[ptr].cmd);          //スタート(X,Y)
    ptr++;
}

//全書き込みデータの動作完了を待つ
while (1) {
    cp530_rStsW(h, 0, sts);      //ステータス読み込み
    if(0x10==(sts[0] & 0x10)) {    //エラー停止
        //エラー停止時の処理はアプリケーションに合わせて適宜処理してください.
        //以下は一例です.
        cp530_rReg (h, 0, 0xf2, &ersts);    //X:エラーステータス読み込み
        cp530_rReg (h, 1, 0xf2, &ersts);    //Y:エラーステータス読み込み
        return;                          //処理キャンセル(終了)
    }
    if(0x20==(sts[0] & 0x20)) {
        cp530_rReg (h, 0, 0xf3, &evsts);    //X:イベントステータス読み込み
        cp530_rReg (h, 1, 0xf3, &evsts);    //Y:イベントステータス読み込み
    }
    if(8==(sts[0] & 9)) {
        break;                          //メインステータス bit0=0 かつ bit3=1 で動作終了
    }
}

```

3.3 同時スタート(STA 出力)

3.3.1 コマンドによる同時スタート(STA 出力)

ボード内の複数軸をソフトウェアコマンドにより同時スタートできます。
手順は以下の通りです。

- (1) 同時スタートする軸の動作モードを設定。RMD.b19,18=01(STA によるスタート)
- (2) 速度, 移動量を各軸に設定
- (3) 各軸にスタートスタートコマンド書き込み
- (4) 任意の軸に同時スタートコマンド(06h)書き込み

3.3.2 同時スタート(STA 入力)による軸間の同期

ボード内の複数軸を外部からの STA 信号により同時スタートできます。
手順は以下の通りです。

- (1) 同時スタートする軸の動作モードを設定。RMD.b19,18=01(STA によるスタート)
- (2) 速度, 移動量を各軸に設定
- (3) 各軸にスタートスタートコマンド書き込み
- (4) STA 入力により各軸スタート

3.4 同時ストップ(STP 出力)

3.4.1 コマンドによる同時ストップ(STP 出力)

ボード内の複数軸をソフトウェアコマンドにより同時停止できます。
但し事前に動作モードの設定が必要です。

	設定事項	設定箇所	使用する関数
1	STP 入力による停止	動作モード(RMD.b24=1)	wReg
2	STP 入力時停止方法	環境設定レジスタ 1(RENV1.b19) 即停止:0, 減速停止:1	wReg

上記の設定がされている軸は任意の軸への同時停止コマンド(07h)書き込みにより同時停止します。

3.4.2 異常停止時自動同時ストップ(STP 入出力)

異常停止時, 自動的に STP 出力をすることができます。
この機能により異常停止時に他の軸も停止することができます。

	設定事項	設定箇所	使用する関数
1	異常停止時 STP 自動出力	動作モード(RMD.b25=1)	wReg

3.5 コンパレータ

3.5.1 コンパレータの種類と機能

各軸 5 式のコンパレータがあります。

No.	名 称	主な用途	比較データ CTR1~4	位置決め用 残パルス数 RPLS	現在 指令速度	プリレジスタ
1	コンパレータ 1(CMP1)	+SLS CTR1 のリングカウンタ	有	無	無	無
2	コンパレータ 2(CMP2)	-SLS CTR2 のリングカウンタ	有	無	無	無
3	コンパレータ 3(CMP3)	汎用	有	無	無	無
4	コンパレータ 4(CMP4)	等ピッチ出力 STA 信号	有	無	無	無
5	コンパレータ 5(CMP5)	外部トリガ STP 信号	有	有	有	有

コンパレータを利用し、以下のことができます。

- (1) ソフトリミット
- (2) リングカウント
- (3) ステッピングモータの脱調検出
- (4) 比較結果外部出力(定ピッチ, 不定ピッチ)
- (5) コンパレータ条件一致による他軸スタート

この節では(3)及びコンパレータ用プリレジスタの状態変移について説明します。

(1), (2)については「2.5.3 座標系」, (4)については「外部機器へのトリガ信号」, (5)については「3.9 複数軸の同期」で説明します。

3.5.2 ステッピングモータの脱調検出

エンコーダ付のステッピングモータ使用時に脱調検出ができます。

エンコーダの分解能とステッピングモータの分解能は同じものを使用します。

対象軸の CMP1~CMP5 が利用できますので使用していない CMP を使用します。

CMP3 を利用した場合の設定箇所は以下の通りです。

	設定事項	設定箇所	使用する関数
1	エンコーダカウント形式	RENV2.b21,20 1 通倍:00, 2 通倍:01, 4 通倍:10, UP/DOWN:11	wReg
2	エンコーダカウント方向	RENV2.b22 A 相進相:0, B 相進相:1	wReg
3	許容できる最大偏差(パルス数)	RCMP3=許容できる最大偏差(パルス数)	wReg
4	脱調検出時即停止	RENV4.b22-b16=011 0110	wReg

3.5.3 PRCP5 及び RCMP5 の状態変化

(1) ボード初期化時

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	0	未確定	00	0
(1stプリレジスタ)	0	未確定		
RCMP5(レジスタ)	0	未確定		

(2) ボード初期化時にPRCP5に0を書込む

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	0	未確定	01	0
(1stプリレジスタ)	0	未確定		
RCMP5(レジスタ)	0	確定		

(3) RCMP5プリレジスタキャンセルコマンド(27h)書込み

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	0	未確定	01	0
(1stプリレジスタ)	0	未確定		
RCMP5(レジスタ)	0	確定		

(4) RCMP5=1000を書き込む(1回目の比較データ)

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	0	未確定	01	0
(1stプリレジスタ)	0	未確定		
RCMP5(レジスタ)	1000	確定		

(5) PRCP5=3000を書き込む(2回目の比較データ)(1stが確定)

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	3000	未確定	10	0
(1stプリレジスタ)	3000	確定		
RCMP5(レジスタ)	1000	確定		

(6) PRCP5=6000を書き込む(3回目の比較データ)(2ndが確定) MSTS.b15=1 (プリレジスタ満杯状態)

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRCP5(2ndプリレジスタ)	6000	確定	11	1
(1stプリレジスタ)	3000	確定		
RCMP5(レジスタ)	1000	確定		

(7) 最初の比較データ一致 MSTSのSPDF(ビット15)='0' (2ndが未確定)

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRMV(2ndプリレジスタ)	6000	未確定	10	0
(1stプリレジスタ)	6000	確定		
RMV(レジスタ)	3000	確定		

(8) PRCP5=10000を書き込む(4回目の比較データ)(2ndが確定) MSTS.b15=1 (プリレジスタ満杯状態)

	内 容	記憶状態	RSTS.b19,18	MSTS.b15
PRMV(2ndプリレジスタ)	10000	確定	11	1
(1stプリレジスタ)	6000	確定		
RMV(レジスタ)	3000	確定		



注 意

途中でコンパレータ比較データをキャンセルする場合はプリレジスタキャンセルコマンド(27h)でキャンセルします。
 この時は RCMP5 のデータはキャンセルされないので注意が必要です。
 (次の動作時の 1 回目の比較データは RCMP5 にも書き込む)

3.6 外部機器へのトリガ信号

指定座標通過によるトリガ出力が可能です。

3.6.1 定ピッチ出力(CMP4のみ)

この機能はコンパレータ 4(CMP4)とカウンタ 4(CTR4)を使用して、一定間隔ごとに外部出力をする機能です。

	設定事項	設定箇所	使用する関数
1	CTR4 初期値設定	RCTR4	wReg
2	比較する入力ソース	RENV3.b13,12 指令パルス出力:00, エンコーダ入力:01	wReg
3	出力条件 比較カウンタの選択 比較方法 条件成立時の処理	RENV4.b31-b24 RENV4.b25,24 = 11(CTR4) RENV4.b29-26 カウント方向無関係:1000 カウントアップ :1001 カウントダウン :1010 RENV4.b31,30 = 00	wReg
4	ピッチ間パルス数	RCMP4	wReg
5	外部出力設定	オプションポート 「3.6.5 CMP 端子出力設定」を参照	wPortB / wPortW

(1) 補足説明

- 10 パルス毎に CMP 信号出力の場合は、RCMP4=9 にします。
- また、CTR4 の初期値は 9 にします。これにより、スタート前の時点で CMP 出力は ON 状態になります。
- アップカウントの時、CTR4 の値は、 $\underline{9} \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow 8 \rightarrow \underline{9} \rightarrow 0 \rightarrow \dots$ と、変化していきます。
値が 9 になる度に CMP 出力は ON になり、値が 0 になると OFF します。
- ダウンカウントの時、CTR4 の値は、 $\underline{9} \rightarrow 8 \rightarrow 7 \rightarrow 6 \rightarrow \dots \rightarrow 1 \rightarrow 0 \rightarrow \underline{9} \rightarrow 8 \dots$ と、変化します。
値が 9 になる度に CMP 出力は ON になり、値が 8 になると OFF します。
- 定速動作時の CMP 出力パルス幅は、軸の動作速度によって決まります。
- 最初に CTR4 の値を、RCMP4 と等しい値で初期化した時点で CMP 出力が ON となる点と、丁度 CTR4 が RCMP4 と等しい状態で停止すると、同じく CMP 出力は ON のままになる点に注意が必要です。
- 上記の条件を利用すると、例えば、アップカウント時に CTR4=8 でスタートすれば、同期出力はスタート直後からパルスで出力されます。(スタート前は OFF 状態です)
- また、アップカウント方向のみの移動時であれば、CTR4 の初期値を RCMP4 の値より大きな値(例:11)で初期化しておけば、スタート後 10 パルス目で始めの同期出力があり、その後、10 パルス毎に繰り返します。(同じくスタート前は OFF 状態です)
- 周期が 10 パルス以外の場合も上記と同様に、9 パルスの代りに(周期パルス数-1)での設定となります。

3.6.2 CMP3 比較結果出力

	設定事項	設定箇所	使用する関数
1	出力端子設定	RENV2.b11.10=11	wReg(ライブラリではDevOpen 時設定)
2	比較カウンタの選択 比較方法 条件成立時の処理	RENV4.b17,16 CTR1:00, CTR2:01, CTR3:10, CTR4:11 RENV4.b20-18 RCMP3 = 比較カウンタ(カウント方向無関係):001 RCMP3 = 比較カウンタ(カウントアップ) :010 RCMP3 = 比較カウンタ(カウントダウン) :011 RCMP3 > 比較カウンタ :100 RCMP3 < 比較カウンタ :101 RENV4.b22,21 = 00	wReg
3	比較データ	RCMP3	wReg
4	外部出力設定	オプションポート「3.6.5 CMP 端子出力設定」を参照	wPortB / wPortW

3.6.3 CMP4 比較結果出力

	設定事項	設定箇所	使用する関数
1	出力端子設定	RENV2.b13.12=11	wReg(ライブラリではDevOpen 時設定)
2	比較カウンタの選択 比較方法 条件成立時の処理	RENV4.b25,24 CTR1:00, CTR2:01, CTR3:10, CTR4:11 RENV4.b29-26 RCMP4 = 比較カウンタ(カウント方向無関係):0001 RCMP4 = 比較カウンタ(カウントアップ) :0010 RCMP4 = 比較カウンタ(カウントダウン) :0011 RCMP4 > 比較カウンタ :0100 RCMP4 < 比較カウンタ :0101 RENV4.b31,30 = 00	wReg
3	比較データ	RCMP4	wReg
4	外部出力設定	オプションポート「3.6.5 CMP 端子出力設定」を参照	wPortB / wPortW

3.6.4 CMP5 比較結果出力

RCMP5 はプリレジスタがあります。

MSTS.b15=0(PRCP5 空き)ならば PRCP5 に次々にデータを書込むことで、任意の点でのコンパレータ出力が可能です。

	設定事項	設定箇所	使用する関数
1	出力端子設定	RENV2.b15.14=11	wReg(ライブラリではDevOpen 時設定)
2	比較カウンタの選択 比較方法 条件成立時の処理	RENV5 b2-0 CTR1:000, CTR2:001, CTR3:010, CTR4:011, 移動残パルス(RPLS):100, 現在速度データ(RSPD):101 RENV5.b5-3 RCMP5 = 比較カウンタ(カウント方向無関係):001 RCMP5 = 比較カウンタ(カウントアップ) :010 RCMP5 = 比較カウンタ(カウントダウン) :011 RCMP5 > 比較カウンタ :100 RCMP5 < 比較カウンタ :101 RENV5.b7,6 = 00	wReg
3	比較データ	RCMP5	wReg
4	外部出力設定	オプションポート「3.6.5 CMP 端子出力設定」を参照	wPortB / wPortW

3.6.5 CMP 端子出力設定

CMP 端子は HPCI-CPD574N/578N/5212M, HPC-CPD278, HP104D-CPD364, HUSB-CPD434v2

, HUSB-CPD434U, HETN-CPD834T, HWIF-CPD834W に有ります。

端子のピン番号は各製品ユーザーズマニュアル<ハードウェア編>を参照ください。

設定は各ボードのオプションポートに行います。

	ボード種別	オプションポート 設定箇所	設定データ (bit)			
			7,6	5,4	3,2	1,0
1	HPCI-CPD574N HPCI-CPD578N HPCIe-CPD674N	BAR+8Ch	CMPU 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPZ 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPY 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPX 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
	HPCI-CPD578N	BAR+8Ah	CMPB 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPA 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPW 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPV 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
2	HPCIe-NCB674N HPCIe-NCB674N(1)	BAR+8Ch	CMPU 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPZ 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPY 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPX 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
3	HPCI-CPD5212M	BAR3+06h	CMPU1 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPZ1 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPY1 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPX1 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
4	HPC-CPD278	BAR+48h	CMPU 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPZ 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPY 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPX 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
		BAR+4Ah	CMPB 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPA 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPW 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止	CMPV 出力 00: CMP3 01: CMP4 10: CMP5 11: 出力禁止
5	HP104D-CPD364	BAR+2Ah	CMPU 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPZ 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPY 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPX 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND
6	HUSB-CPD434v2 HETN-CPD834T HWIF-CPD834W	コマンド A0h	CMPU 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPZ 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPY 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND	CMPX 出力 00: CMP3 01: CMP4 10: CMP5 11: CMP3-5AND

BAR: ベースアドレス

	ボード種別	オプションポート 設定箇所	設定データ (bit)			
			15-14	13-11	10-8	7-0
7	HPCI-CPD553	BAR+88h	00: XCMP 01: YCMP 10: ZCMP 11: UCMF	J1 CMP1 000: XCMP 001: YCMP 010: ZCMP 011: UCMF 1xx: input	J1 CMP0 000: XCMP 001: YCMP 010: ZCMP 011: UCMF 1xx: input	Same as HPCI-CPD574N

※HPCI-CPD553 は、X 軸、Y 軸、Z 軸として 3 つだけの外部入力を有しています。

しかしながら、内部レジスタとして U 軸を装備しており、アプリケーションで使用することができます。

	ボード種別	出力コネクタ
1	HPCI-CPD534	J2 コネクタ 及び 5 コンパレータ
2	HPCI-CPD553	J1 コネクタ, J2 コネクタ
3	HPCI-CPD574N, HPCle-CPD674N, HPCI-CPD574Nr1, HPCle-CPD674Nr1	J1 コネクタ, J3 コネクタ
4	HPCI-CPD578N, HPCle-CPD678N, HPCI-CPD578Nr1, HPCle-CPD678Nr1	J1 コネクタ, J2 コネクタ, J3 コネクタ
5	HPCI-CPD5212M	J1 コネクタ, J2 コネクタ, J3 コネクタ
6	HPCle-NCB674N and HPCle-NCB674N(1)	J1 コネクタ, J3 コネクタ
7	HPC-CPD278	J1 コネクタ, J2 コネクタ
8	HP104D-CPD364CB	J6 - J9 コネクタ
9	HP104D-CPD364KR and HP104D-CPD364MR	J4 コネクタ
10	HUSB-CPD434v2 HETN-CPD834T HWIF-CPD834W	J6 コネクタ

3.6.6 不等ピッチ出力例

以下は HPCI-CPD574N/578N, HPCIe-CPD674N/678N の場合のコード例です.

//Trigger source X-axis CMP5 condition CTR2 = PRCP5 (don't care count direction)

cp530_wReg (hDev, 0, 0xA0, 0x00000009); //XRENV5

cp530_wPortB (hDev, 0x8C, 0x02); //XCMP5->XCMP out

while (1) {

cp530_rMstsW (hDev, 0, &msts); //Read Main status

if(0==(msts & 0x8000)) { //Can write to PRCP5

cp530_wReg (hDev, 0, 0x8B, cmpdata[i]); //cmpdata[] is comparison data array

i++; //write count increment

}

If(DATA_NUM == i) { //DATA_NUM:number of comparison data

Break; //Exit loop

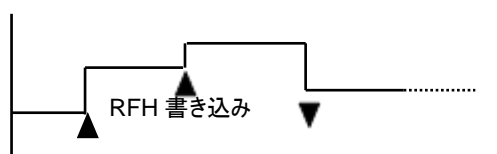
}

}

3.7 速度オーバーライド

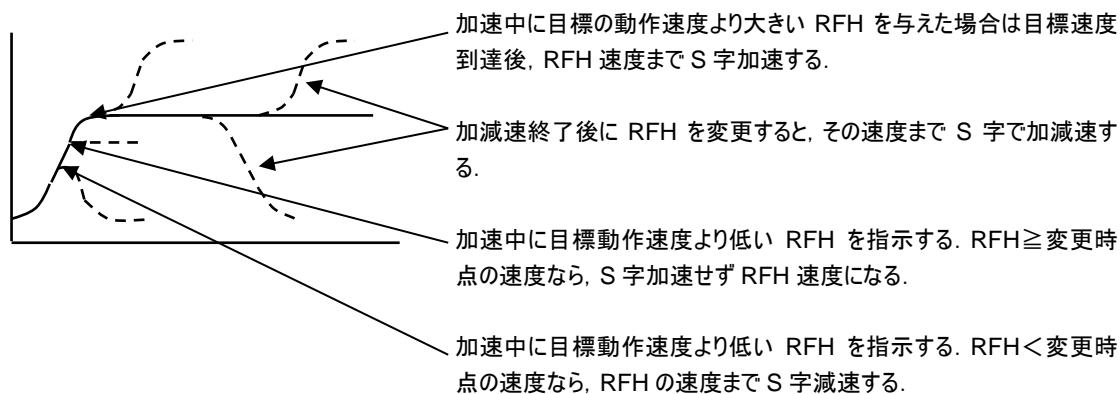
RFHへ直接書き込むことで動作中に速度変更が出来ます。

(1) FH 定速動作の場合

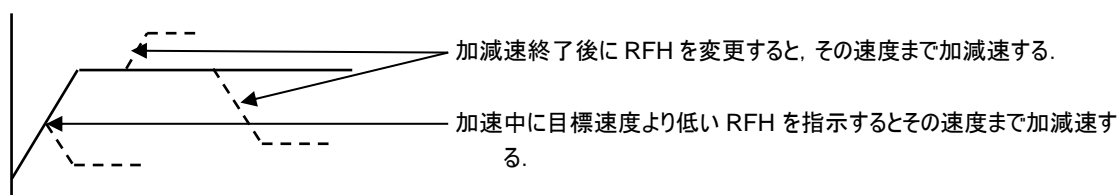


RFH書き込みで瞬時に速度が変更されます。

(2) S 字加減速動作の場合



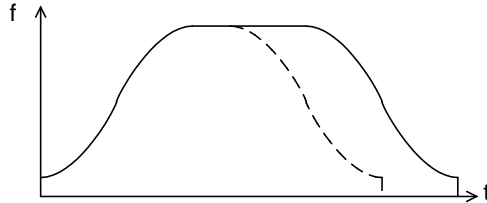
(3) 直線加減速動作の場合



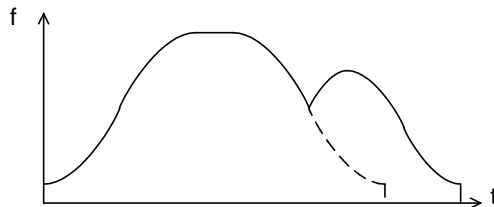
3.8 位置オーバーライド

位置決め動作中に、自由に目標位置のオーバーライド(目標位置の書き換え)ができます。
変更はスタート位置を基準とした新しい目標位置データ(RMV レジスタ値)を書き換えます。

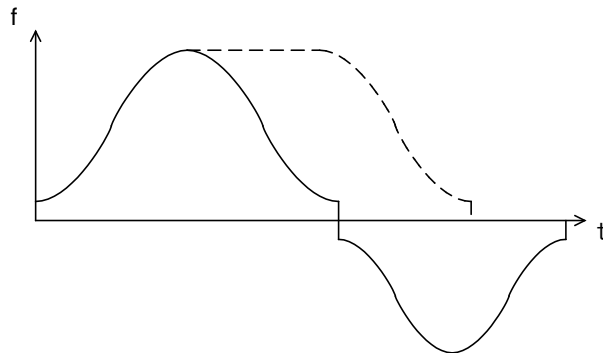
- (1) 加速または定速動作中に新データを最初の目標位置より遠方に変更する場合
そのままの速度パターンで動作し、新データ位置(新 RMV 値)で位置決め完了になります。



- (2) 減速中に新データを最初の目標値より遠方に変更する場合
オーバーライドした位置から FH まで再加速後、新データ位置(新 RMV 値)で位置決め完了になります。



- (3) 新データ位置をすでに通過している時、または減速中で最初の目標位置より手前に変更した場合
減速停止後に逆転して新データ位置(新 RMV 値)で位置決め完了になります。



- (4) 停止直前のオーバーライド指令を行う場合

RMV への書き込みタイミングによっては無視される可能性があります。

成功不成功を確認するには動作完了後に位置を読み出すか、または同じく動作完了確認時に MSTs.b13(SEOR)を確認します。MSTs.b13(SEOR)=1 の場合は位置のオーバーライドが無視されたことを示します。

*1. MSTs.b13(SEOR)は MSTs 読み出しでクリアされますので注意が必要です

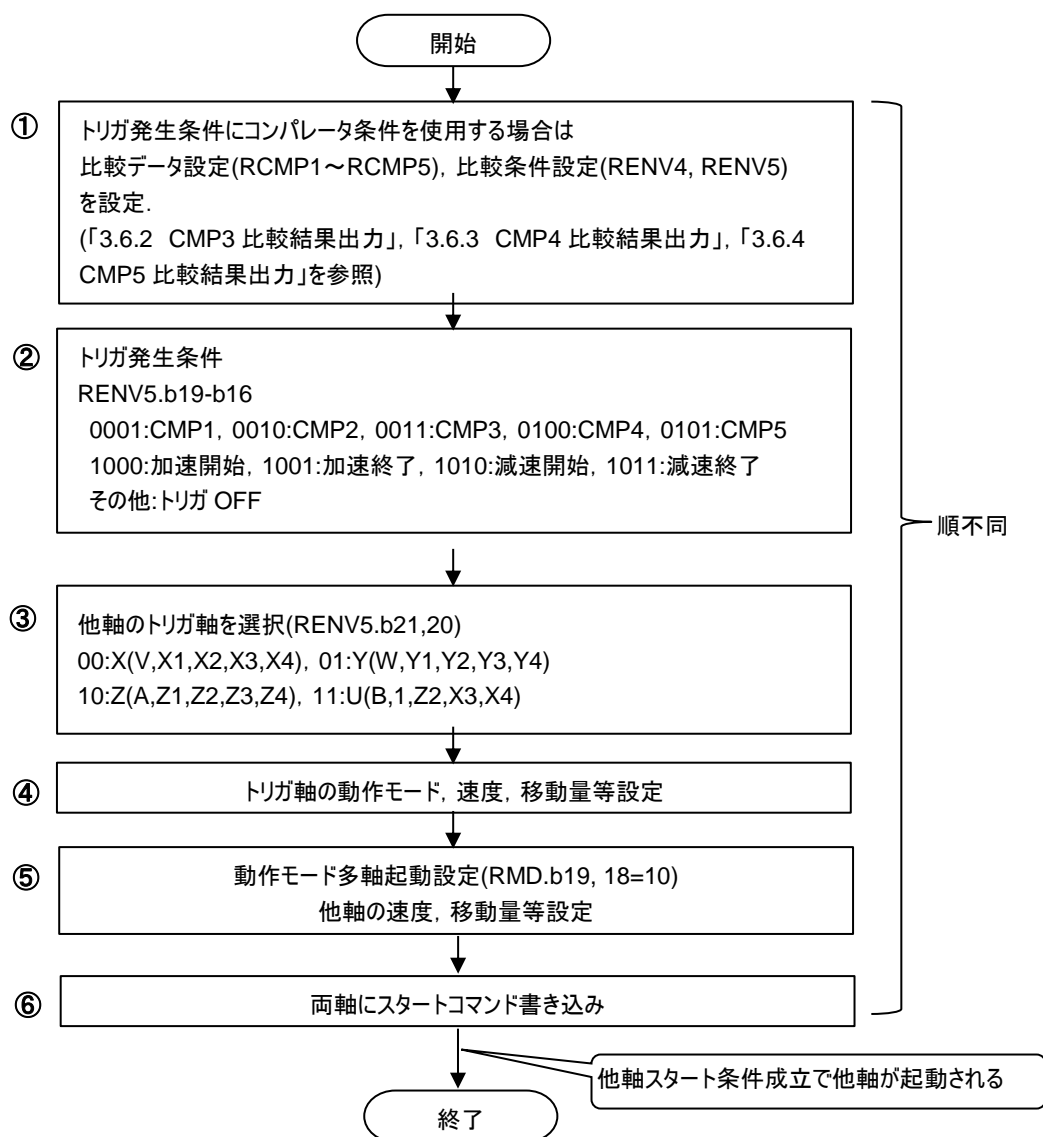
*2. MSTs.b13(SEOR)は HPCI-CPD574N, HPCI-CPD578N, HPCIe-CPD674N のみ対応しています。

3.9 複数軸の同期

3.9.1 コンパレータ条件一致及び加減速タイミングによる他軸スタート

同一 PCL 内の組み合わせ(X-U, V-W, X1-U1, X2-U2, X3-U3, X4-U4)でコンパレータ条件または加減速のタイミングにより他軸のスタートを行う場合について説明します。

コンパレータ比較を行う(または加減速のタイミングを作る)軸をトリガ軸, コンパレータ条件によりスタートされる軸を他軸とします。以下の手順で動作させます。



3.9.2 CMP 条件成立による他軸スタートの簡単な例

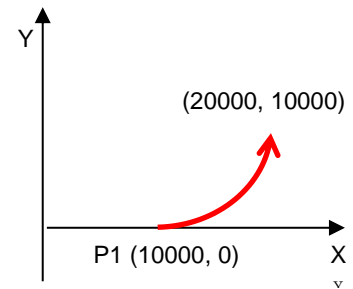
これは、X 軸が点 P1 (10000) を通過したことによる Y 軸スタートの例です。

【ドライバ関数による記述の例】

以下の例では、各軸の初期設定が既に終了しているものとします。

Y 軸は、X 軸のコンパレータ条件成立で開始します。

以下の例では、Y 軸は X 軸の指令位置が 10000 になった時から開始します



```
// X-axis is trigger axis, Y-axis is other axis
//X-axis compare data = 10000
cp530_wReg(hDev, X_AX, //device handle and x-axis
            0x8B,      //write PRCP5
            10000 );   //compare data
//X-axis environment 5 (set compare condition)
cp530_wReg(hDev, X_AX, //device handle and x-axis
            0xA0,      //write environment 5 (RENV5)
            0x50008 ); //RCTR1=RCMP5(b5-3=001,b2-0=000),
                      //start trigger by CMP5 coincidence (b19-16=0101)
```

前ページ ①, ②

```
//X-axis environment 5 (starting other axis trigger)
cp530_wReg(hDev, Y_AX, //device handle
            0xA0,      //write environment 5 (RENV5)
            0 );        //select the trigger axis for X-axis(b20,21=00)
```

前ページ ③

```
//X-axis distance
cp530_wReg(hDev, X_AX, //device handle and X-axis
            0x80,      //write PRMV
            20000 );   //distance = 20000pulse
//X-axis operation mode is PTP(Relative)
cp530_wReg(hDev, X_AX, //device handle and X-axis
            0x87,      //write PRMD
            0x41 );    //PTP(Relative)
```

前ページ ④

```
//Y-axis distance
cp530_wReg(hDev, Y_AX, //device handle and Y-axis
            0x80,      //write PRMV
            10000 );   //distance = 10000pulse
//Y-axis operation mode is PTP(Relative) and start
cp530_wReg(hDev, Y_AX, //device handle and Y-axis
            0x87,      //write PRMD
            0x80041 ); //PTP(Relative) and start by other axis trigger
```

前ページ ⑤

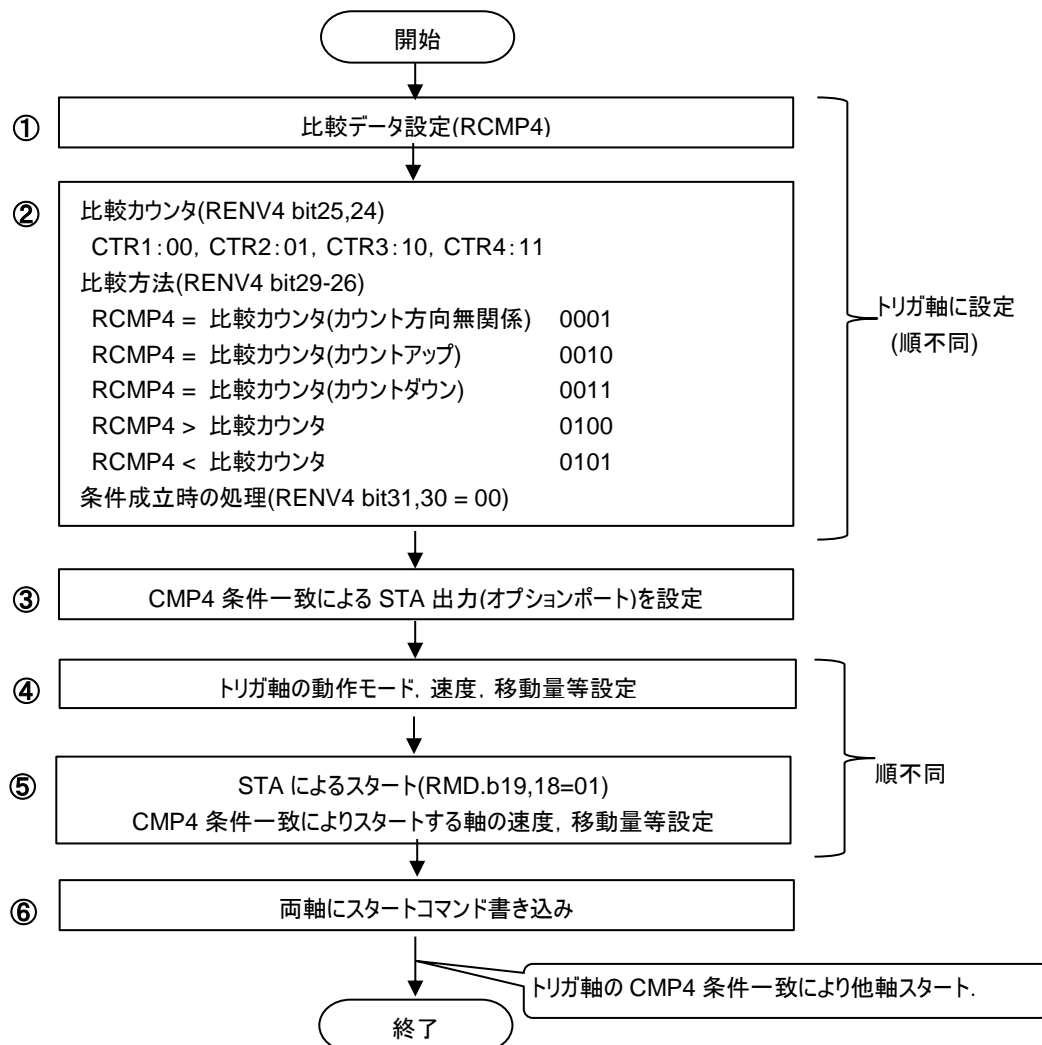
```
cp530_wCmdW(hDev, X_AX, 0x0353 ); //X and Y Acceleration start
```

前ページ ⑥

3.9.3 コンパレータ条件一致 STA 出力による同期

トリガ軸の CMP4 の条件一致により STA 出力し他軸を起動します。

以下の手順で動作させます。



3.9.4 コンパレータ条件成立での STA 出力による同時スタート設定例

```
// Y-axis start by using the simultaneous start signal of X-axis
// in case that output is executed at XCTR2 = 10000 (Count direction unrelated)
// Other data settings necessary for the operation of the X-axis and the Y-axis are assumed to be already
// configured
// Y-axis starts positioning at XCTR2 = 10000
cp530_wReg (hDev, X_AX, 0xaa, 10000); // comparison data. Previous Page①
cp530_wReg (hDev, X_AX, 0x9f, 0x05000000); // environmental settings 4 Previous Page②
cp530_wPortB(hDev, 0x84, 0x01); // option port settings Previous Page③
cp530_wReg (hDev, X_AX, 0x87, 0x00041); // X-axis (trigger axis) operation mode positioning Previous Page④

cp530_wReg (hDev, X_AX, 0x80, 20000); // X-axis (trigger axis) sets travel distance Previous Page④

cp530_wReg (hDev, Y_AX, 0x87, 0x40041); // Y-axis (other axes) starts positioning by STA input Previous Page⑤

cp530_wReg (hDev, Y_AX, 0x80, 10000); // Y-axis (other axes) sets travel distance Previous Page⑤

cp530_wCmdW (hDev, X_AX, 0x0353); // start command of X- and Y-axis Previous Page⑥
```

(1) CMP4 条件一致による STA 出力設定

各ボードにより設定箇所が異なります。

	ボード種別	オプションポート 設定箇所	bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	HPCI-CPD532 HPCIe-CPD632	BAR+84h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
2	HPCI-CPD534 HPCI-CPD553 HPCI-CPD574N HPCIe-CPD674N	BAR+84h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
3	HPCI-CPD508 HPCI-CPD578N	BAR+84h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
4	HPCI-CPD5016	BAR+84h		U4	Z4	Y4	X4	U3	Z3	Y3	X3	U2	Z2	Y2	X2	U1	Z1	Y1	X1
5	HPCI-CPD5212M	BAR3+02h		-	-	-	-	U3	Z3	Y3	X3	U2	Z2	Y2	X2	U1	Z1	Y1	X1
6	HPCIe-NCB674N HPCIe-NCB674N(1)	BAR+84h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
7	HPC-CPD234	BAR+24h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
8	HPC-CPD278	BAR+44h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X
9	HPC104-CPD132	BAR+14h		-	-	-	-	-	-	-	-	-	-	-	-	-	-	Y	X
10	HP104D-CPD364	BAR+24h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
11	HUSB-CPD434v2 HETN-CPD834T HWIF-CPD834W	コマンド 84h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
12	HCPCI-CPD734	BAR+84h		-	-	-	-	-	-	-	-	-	-	-	-	U	Z	Y	X
13	HCPCI-CPD738	BAR+84h		-	-	-	-	-	-	-	-	B	A	W	V	U	Z	Y	X

BAR: ベースアドレス

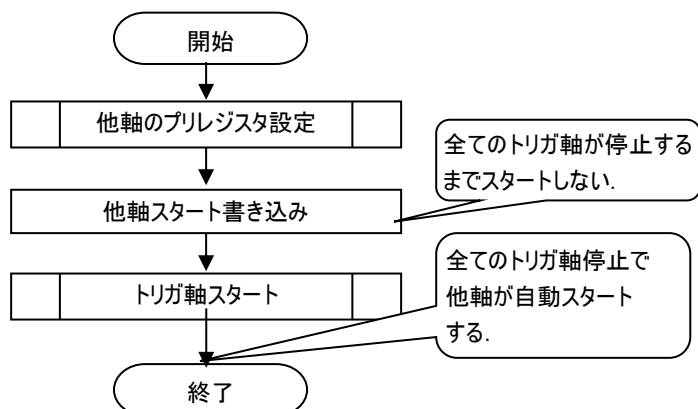
各ビット“1”で対象軸の CMP4 条件一致により STA 出力されます。

※ HPCI-CPD553 は、3軸仕様の製品となるため、U 軸がありません。

3.9.5 軸停止による他軸スタート

停止する軸をトリガ軸、停止により起動される軸を他軸とします。

以下の手順で動作させます。



	設定事項	設定箇所	使用する関数
1	自軸を含めた指定軸停止によるスタート	RENV2.b29 1:有効, 0:無効(自軸をトリガ軸にできない)	wReg
2	トリガ軸の指定	RMD.b23-20(下図参照)	wPortB / wPortW
3	軸停止によるスタート	RMD.b19,18=11	wPortB / wPortW

トリガ軸の指定 RMD.b23-20

bit	23	22	21	20
	トリガ軸の指定(1で指定)			
	U(B,U1,U2,U3,U4)	Z(A,Z1,Z2,Z3,Z4)	Y(W,Y1,Y2,Y3,Y4)	X(V,X1,X2,X3,X4)

3.9.6 軸停止トリガによるスタートの例

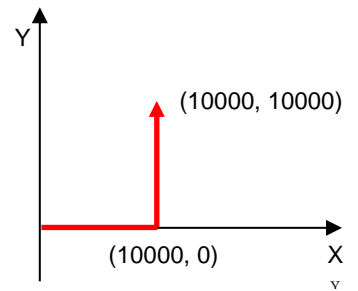
これは、X 軸が目標点に到達して停止したことによる Y 軸スタートの例です。

【ドライバ関数による記述の例】

以下の例では、各軸の初期設定が既に終了しているものとします。

Y 軸は、X 軸のコンパレータ条件成立で開始します。

以下の例では、Y 軸は X 軸の指令位置が 10000 になった時から開始します。

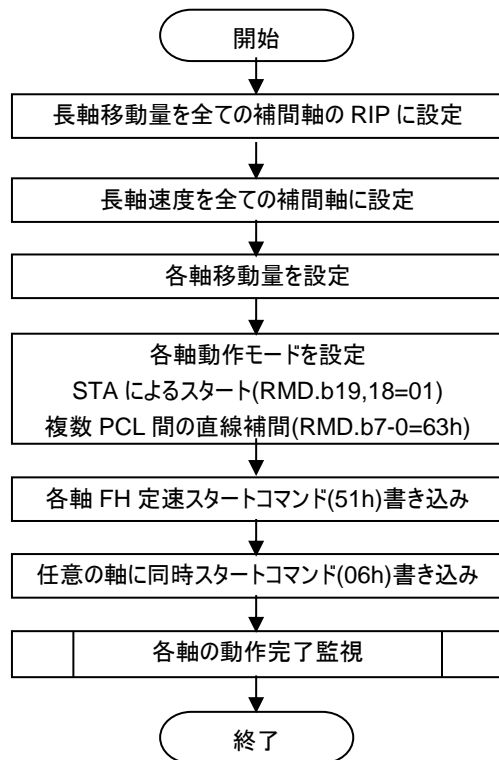


```
// X-axis travel distance setting (travel distance 10000)
cp530_wReg( hDev,          //Device handle
            X_AX,          //specifies the X-axis
            0x80,          //specifies the travel distance (PRMV)
            10000 );       // travel distance 10000
// Y-axis travel distance setting (travel distance 10000)
cp530_wReg( hDev,          //Device handle
            Y_AX,          // specifies the Y-axis
            0x80,          // specifies the travel distance (PRMV)
            10000 );       // travel distance 10000
// sets the operation mode of X-axis (positioning operation)
cp530_wReg( hDev,          //Device handle
            X_AX,          // specifies the X-axis
            0x87,          // specifies the operation mode (PRMD)
            0x41 );        // positioning operation
// sets the operation mode of Y-axis (start continuous feed to positive direction at the condition of X-axis
stop)
cp530_wReg( hDev,          //Device handle
            Y_AX,          // specifies the Y-axis
            0x87,          // specifies the operation mode (PRMD)
            0x1c0041 );    //start positioning at the condition of X-axis stop
// executes the start command
cp530_wCmdW( hDev,        //Device handle
            X_AX,          // specifies the X-axis
            0x0353 );      // X- and Y-axis acceleration start
```

上記の設定で、X 軸は直ちに開始され、Y 軸は X 軸の停止状態で開始されます。

3.9.7 複数 PCL 間の直線補間

基本的に次動作連続実行はできません。また定速動作のみです。



注:

INPOS 制御が有効な状態では、各補間軸の完了タイミングが一致しません。
PCL 間の各補間軸の完了タイミングは一致しません。
各 PC の最長軸の動作完了は、全体の動作の完了となります。
このため、連続的な実行は出来ません。

3.9.8 PCL 間の直線補間の例

以下の例では、同じ値を各補間軸速度レジスタに設定されているものとします(PRFH, PRMG).
そして、各軸の動作モードレジスタ (PRMD) は00040063hであるとしています。
(2 PCL間の直線補間:0x63 スタートはSTA input).

以下の例は、XZWA軸への直線補間の設定例で、
各々の移動量の値は X=10000, Z=5000, W=10000, A=2000です。

```
//Sets travel distance for X axis. (Travel distance: 10,000)
cp530_wReg ( hDev, X_AX,          //Specifies X axis.
             0x80,                //Specifies travel distance (PRMV).
             10000 ) ;            //Travel distance: 10,000
//Sets travel distance for long axis. (Travel distance: 10,000)
cp530_wReg ( hDev, X_AX,          //Specifies X axis.
             0x88,                //Specifies travel distance of long axis (PRIP).
             10000 ) ;            //Travel distance: 10,000
//Sets travel distance for Z axis. (Travel distance: 5,000)
cp530_wReg ( hDev, Z_AX,          //Specifies Z axis.
             0x80,                //Specifies travel distance (PRMV).
             5000 ) ;             //Travel distance: 5,000
//Sets travel distance for long axis. (Travel distance: 10,000)
cp530_wReg ( hDev, Z_AX,          //Specifies Z axis
             0x88,                //Specifies travel distance of long axis (PRIP).
             10000 ) ;            //Travel distance: 10,000
//Sets travel distance for W axis. (Travel distance: 1,000)
cp530_wReg ( hDev, W_AX,          //Specifies W axis.
             0x80,                //Specifies travel distance (PRMV).
             1000 ) ;             //Travel distance: 1,000
//Sets travel distance for long axis. (Travel distance: 10,000)
cp530_wReg ( hDev, W_AX,          //Specifies W axis.
             0x88,                //Specifies travel distance of long axis (PRIP).
             10000 ) ;            //Travel distance: 10,000
//Sets travel distance for A axis. (Travel distance: 2,000)
cp530_wReg ( hDev, A_XX,          //Specifies A axis.
             0x80,                //Specifies travel distance (PRMV).
             2000 ) ;             //Travel distance: 2,000
//Sets travel distance for long axis. (Travel distance: 10,000)
cp530_wReg ( hDev, A_AX,          //Specifies A axis.
             0x88,                //Specifies travel distance of long axis (PRIP).
             10000 ) ;            //Travel distance: 10,000

//Executes start command (X axis and Z axis).
cp530_wCmdW ( hDev, X_AX,        //Specifies group of axes U to X.
             0x0551 ) ;           //FH constant-speed start for X axis and Z axis.
//Executes start command (W axis and A axis).
cp530_wCmdW ( hDev, V_AX,        //Specifies group of axes V to B.
             0x0651 ) ;           //FH constant-speed start for W axis and A axis.

//Executes simultaneous start command.(STA output)
cp530_wCmdW ( hDevID, X_AX, 0x06 ) ; //Simultaneous start
```

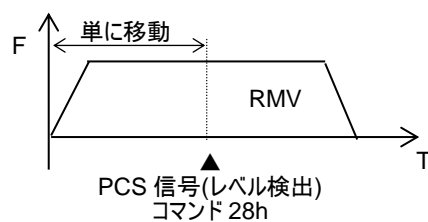
3.10 PCS 機能

3.10.1 対応機種

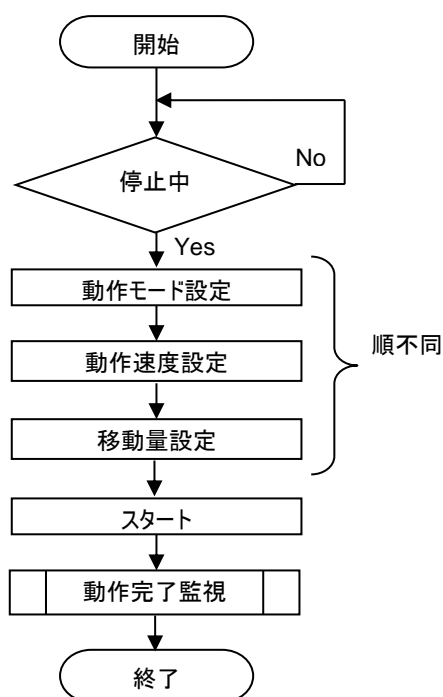
HPCI-CPD5212M 以外の機種。

3.10.2 設定・手順

動作スタート後, PCS 検出, またはコマンド(28h)書込みで RMV(移動量)に設定した移動量の位置決め動作を行います。
動作方向は RMV に設定された値の符号で決定されます。



速度倍率, ベース速度, 加速レート, 減速レート, PCS/DLS の切り替え, PCS 入力極性は初期化時に設定しておきます。
手順は以下の通りです。

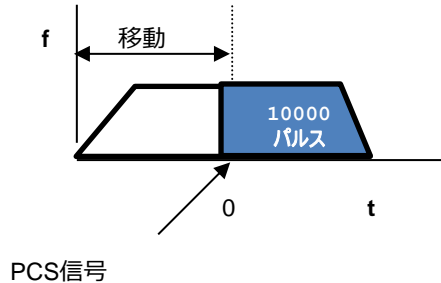


	設定事項	設定箇所	使用する関数
1	PCS/DLS 切り替え	オプションポート	SetDlsSel
2	PCS 入力極性	RENV1.b24	SetDlsSel
3	動作モード設定	RMD.b14 =1, RMD.b7-0=41h	WritOpeMode

3.10.3 PCS コード例

この例では、レジスタとオプションポートが既に初期化されていること、PCS 入力有効、そして PCS の極性が設定済みであることを前提としています。

位置決め（10000 の移動）は、PCS 信号が X 軸に対して入力された点から開始されます。



例1 (ドライバ関数)

```
//Sets travel distance for X axis. (Travel distance: 10,000)
cp530_wReg ( hDev, X_AX,                                //Specifies X axis.
             0x80,                                         //Specifies travel distance (PRMV).
             10000 );                                     //Travel distance: 10,000
//Set operation mode register (PRMD) for X axis (PCS positioning operation).
cp530_wReg ( hDev, X_AX,                                //Specifies X axis.
             0x87,                                         //Specifies operation mode (PRMD).
             0x4041 );                                     //PCS positioning operation
//Write start command.
cp530_wCmdW ( hDev, X_AX, 0x53);                          //Specifies X axis. Acceleration start
```

例2 (ライブラリ関数)

```
//Sets travel distance for X axis. (Travel distance: 10,000)
hcp530_WritPos ( hDev, X_AX, 10000 );                     //Specifies X axis. Travel distance: 10,000
//Set operation mode register (PRMD) for X axis (PCS positioning operation).
hcp530_WritOpeMode ( hDev, X_AX, 0x42 );                 //Specifies X axis.
                                                         //PCS positioning operation(0x42)
hcp530_AccStart( hDev, 0x0001);                          //X axis acceleration start
```

3.11 パルス入力による動作

ハンドルを＋または一方向に操作するとそのパルスの速さでモータが送られる動作です。
操作中に ELS を検出するとそれ以上モータは送られません。 ELS を拔出す方向には動作します。
パルスの送りパルスレートの上限値を RFH, RMG に設定します。
もし、パルスの送り速度がこの値を超えて動かし続けた場合、指令パルスは停止します。



注 意

- エンコーダ入力は各軸に 1 組ですのでパルス入力に使用した場合はフィードバックのエンコーダ信号は入力出来ません。
- イベントステータス RIST.b0 は使用しません。(ハンドル方向が反転する度に RIST.b0=1 になるため)
- ELS 検出時は停止しますが、エラーステータス REST.b6,5 はセットされません。

3.11.1 対応機種

HPCI-CPD508, HPCI-CPD5016 以外の機種。

3.11.2 接続

各軸のエンコーダ入力端子にパルスを接続します。＋/－(UP/DOWN)パルス方式も可能です。
パルスと各軸出力の対応は 1 対 1 です。(下图)

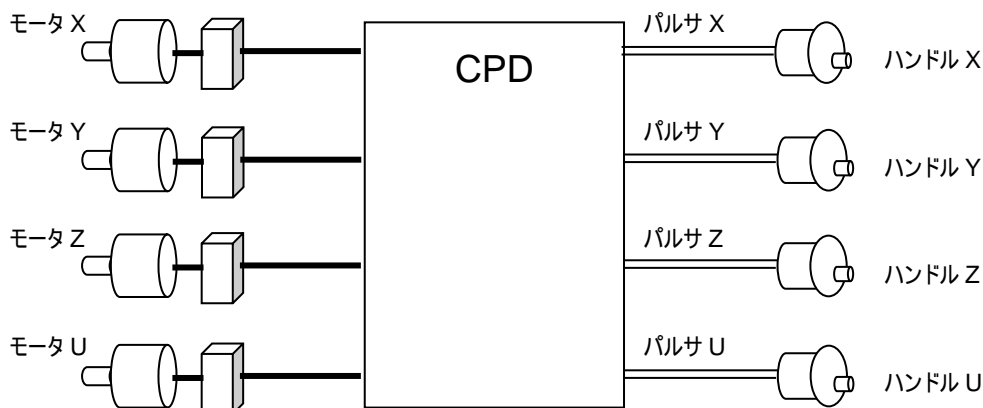


図 3.11-1 パルス入力

3.11.3 設定・手順

- (1) 対象軸の動作モードレジスタに「パルスによる連続動作(01h)」を設定します。
- (2) 対象軸の最高速度に合わせて RFH 及び RMG を適切な値に設定します。
- (3) 対象軸のパルス入力カウンタ形式を設定します。(RENV2.b26～24)
- (4) 対象軸を FH 定速スタートします。以後パルスの回転に同期して軸動作します。
- (5) 即停止コマンドでパルス連続動作モードが解除されます。

3.11.4 パルス入力による動作のコード例

この例では、レジスタとオプションのポートがすでに初期化されていること、PA/ PB 入力（RENV2）と速度倍率（PRMG）が設定されていることを前提としています。

例 1 (ドライバ関数)

```
//Starts manual pulsar.  
//Sets operation mode (continuous operation by pulsar).  
cp530_wReg (hDev, X_AX, 0x87, 1);  
cp530_wReg (hDev, X_AX, 0x82, 5000);           //Sets operation speed. PRFH = 5,000.  
cp530_wCmdW (hDev, X_AX, 0x51);               //FH constant-speed start  
  
//Manual pulsar mode ends.  
cp530_wCmdW (hDev, X_AX, 0x49);               //Immediate stop
```

例 2 (ライブラリ関数)

```
//Starts manual pulsar.  
//Sets operation mode (continuous operation by pulsar).  
hcp530_WritOpeMode (hDev, X_AX, 1);  
hcp530_WritFHSpd (hDev, X_AX, 5000);           //Sets operation speed. PRFH = 5,000.  
hcp530_CnstStartFH (hDev, 1);                 //FH constant-speed start  
  
//Manual pulsar mode ends.  
hcp530_QuickStop (hDev, 1);                   //Immediate stop
```

3.12 外部 JOG 起動入力による動作

3.12.1 対応機種

HP104D-CPD364, HPCI-CPD574N/JOG, HPCle-CPD674N /JOG, HPCle-CPD678N /JOG ,
HPCI-CPD578N/JOG, HPCle-NCB674N/JOG, HPCle-NCB674N(1)/JOG

3.12.2 初期設定

	設定事項	設定箇所	使用する関数	備 考
1	DR 入力極性	RENV1.b25 0:B 接, 1:A 接	wReg	
2	JOG 送り時 ELS による停止時のエラーステータス	RENV5.b11 0:エラーステータス変化なし 1:エラーステータス変化あり	wReg	再スタート時にスタートコマンド必要. HP104D-CPD364 は設定不可.

3.12.3 JOG 送り

DR on の間だけ動作します。

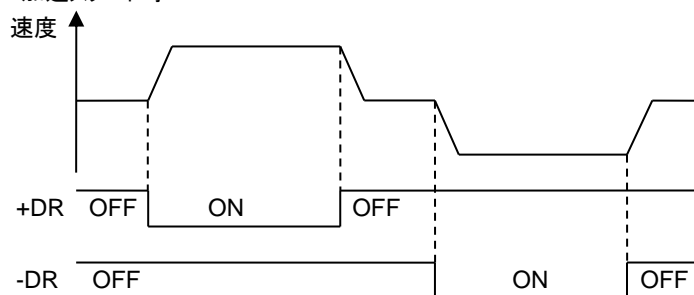
スタートコマンドを書き込み後, +DR on で+方向, -DR on で-方向へ, 設定された速度パターンで動作します。

動作方向の ELS で停止します。この時エラーステータスは変化しません。

(逆方向動作可能。スタートコマンドを再書き込み不必要)

即停止コマンド(49h)書き込みで動作解除します。(以降 DR 無視)

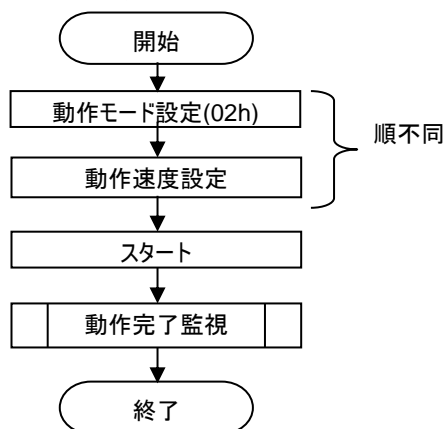
● 加速スタート時



加速スタート(53h)時は, DR off で減速停止します。

減速中に逆方向の DR が ON した時は, 減速停止後に新方向へスタートします。定速スタート時は加減速しません。

手順は以下の通りです。



モードを解除する時は即停止コマンド(49h)を書き込みます。

3.12.4 位置決め

DR off→on のタイミングで設定した速度パターン, 移動量の位置決め動作を行います。

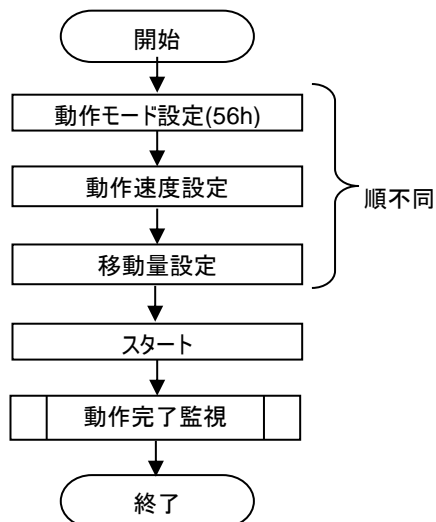
動作中の DR 入力は動作に影響しません。

RMV へ設定する値は正数を設定します。(RMV=0 の場合は即動作完了)

+DR on で+方向, -DR on で-方向へ動作します。

動作方向の ELS on により停止し, エラーステータスが変化します。(再動作時はスタートコマンド書き込み)

手順は以下の通りです。



3.12.5 ステータス

動作モード設定後のスタートコマンド書き込みで,

拡張ステータスレジスタ RSTS.b3-0=0001(DR 入力待ち) になります。

この状態で+DR/-DR on で所定の動作を開始します。

また DR 入力状態は拡張ステータスレジスタ RSTS.b11(+DR), RSTS.b12(-DR)で確認できます。



注 意

+DR と-DR が ON した状態でスタートコマンドを書き込んだ場合は動作しません。

3.13 LTC 入力

3.13.1 対応機種

HPCI-CPD574N, HPCI-CPD578N, HPCIe-CPD674N, HPCIe-CPD678N, HPCI-CPD553, HP104D-CPD364,
HUSB-CPD434CBv2, HUSB-CPD434MSv2, HUSB-CPD434UCB, HUSB-CPD434UMS,
HETN-CPD834TCB, HETN-CPD834TMS, HWIF-CPD834WCB, HWIF-CPD834WMS

3.13.2 機能

外部信号によりカウンタをラッチすることが可能です。
ラッチした値は RLTC1～RLTC4 から読み出せます。
この機能は、ティーチング動作やタッチプローブ動作に使用します。

3.13.3 設定箇所

	設定事項	設定箇所	使用する関数	備 考
1	LTC 入力極性	RENV1.b23 0:B 接, 1:A 接	wReg	エッジでラッチ
2	ラッチ方法選択	RENV5.b13,12=00	wReg	
3	指令速度ラッチ	RENV5.b14=1	wReg	RLTC3 にラッチ
4	ラッチ後カウンタリセット	RENV5.b24=1 CTR1 リセット RENV5.b25=1 CTR2 リセット RENV5.b26=1 CTR3 リセット RENV5.b27=1 CTR4 リセット	wReg	
5	ラッチ信号無効	RENV5.b15=1	wReg	

RENV1 リードコマンド:DCh, ライトコマンド:9Ch, RENV5 リードコマンド:E0h, ライトコマンド:A0h

3.13.4 ステータス

	項 目	確認箇所	使用する関数	備 考
1	LTC 入力状態	RSTS.b14	rReg	
2	カウンタラッチ時イベント	RIST.b14	rReg	RIRQ.b14=1 に設定

RSTS リードコマンド:F1h, RIST ライトコマンド:F3h

3.13.5 ラッチコマンド

	内容	設定箇所	コマンド(HEX)	備 考
1	CTR1 ラッチデータ	RLTC1	ED	
2	CTR2 ラッチデータ	RLTC2	EE	
3	CTR3 ラッチデータ	RLTC3	EF	
4	CTR4 ラッチデータ	RLTC4	F0	

3.13.6 Set/Read DLS/PCS 入力選択 (DLS/PCS)

リード/ライトコマンド: 82h

各軸の DLS 信号入力を、PCS 信号入力として選択します。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	BPCS	APCS	WPCS	VPCS	UPCS	ZPCS	YPCS	XPCS

ビット	ビット名	内容
7-0	nPCS	0: nDLS (POW ON 時), 1: nPCS (nLatch)

3.13.7 LTC 入力コード例

```
WORD      msts;
DWORD     rirq, rist;
long      ltc1, ltc2, ltc3, ltc4;

// XDLS->XLatch
cp530_wPortW(h, 0x82, 0x0001);

// Set latch input event
cp530_rReg(h, X_AX, 0xEC, &rirq);
rirq |= 0x4000;      //latch input event(bit14=1)
cp530_wReg(h, X_AX, 0xAC, &rirq);

while(1) {
    cp530_rMstsW(h, X_AX, &msts);
    if(msts & 0x0020) {
        // Read Event Status
        cp530_rReg(h, X_AX, 0xF3, &rist);
        if(rist & 0x4000) {
            // Read latch register
            cp530_rReg(h, X_AX, 0xED, &ltc1);
            cp530_rReg(h, X_AX, 0xEE, &ltc2);
            cp530_rReg(h, X_AX, 0xEF, &ltc3);
            cp530_rReg(h, X_AX, 0xF0, &ltc4);
        }
    }
}
```

3.14 マスタ・スレーブ機能

3.14.1 対応機種

HPCI-CPD574N, HPCI-CPD578N, HPCle-CPD674N, HPCle-CPD678N, HPCle-NCB674N,
HPCle-NCB674N(1), HPCI-CPD553

3.14.2 マスタ・スレーブ用語

項	用語	内容
1	マスタ・スレーブ機能	並行 2 軸制御モード, マスタ追従モード, スレーブ拡張モードの総称
2	並行 2 軸制御機能	マスタ軸の指令パルスに同期してスレーブ軸の指令パルスを出力.
3	マスタ追従	マスタ軸のエンコーダ入力に同期してスレーブ軸の指令パルスを出力.
4	スレーブ拡張	サブマスタからの指令パルスをスレーブ軸のパルス入力に入力.
5	サブマスタ	スレーブ拡張モード時に使用. この軸からの指令パルスをスレーブ軸のパルス入力に入力.
6	マスタ・スレーブエリア機能	予め設定したコンパレータによりマスタ・スレーブ機能の有効・無効を切り替える.
7	サブマスタ出力パルス スレーブ入力パルス	(J3 コネクタを使用した)ボード間のマスタ・スレーブ機能に使用する入出力信号. ボード間マスタ・スレーブ接続に使用される.
8	マスタエンコーダ	マスタ・スレーブ機能(マスタ追従モード, スレーブ拡張モード)に使用するエンコーダ.

表 3.14-1 マスタ・スレーブ機能用語

3.14.3 マスタ・スレーブ機能のモード種類

マスタ・スレーブ機能は以下の 3 種類のモードがあります.

モード	仕様
マスタ追従	X 軸のエンコーダ入力をスレーブ軸(Y~U)の指令パルス出力として出力可能. V 軸のエンコーダ入力をスレーブ軸(W~B)の指令パルス出力として出力可能.
並行 2 軸 制御機能	X 軸をマスタ軸とし, X 軸の指令パルスをスレーブ軸(Y, Z, U)の指令パルス出力として出力可能. V 軸をマスタ軸とし, V 軸の指令パルスをスレーブ軸(W, A, B)の指令パルス出力として出力可能.
スレーブ拡張	Z 軸または U 軸をサブマスタ軸とし, サブマスタ軸の指令パルスをスレーブ軸(U~B, 他ボードの X~B)のパルス入力へ入力可能. この機能により, スレーブ軸の動作モードを手動パルスモードに設定してサブマスタ軸とスレーブ軸の同期が可能となる.

表 3.14-2 マスタ・スレーブ機能モード

3.14.4 マスタ・スレーブ機能設定ポート

(1) マスタ・スレーブエリア機能有効(SYNC_C_EN)

Read/Write コマンド: 94h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	V_EN	*	*	*	X_EN

ビット	名称	内容
0	X_EN	0: X-U 軸マスタ・スレーブエリア機能無効 (POW ON 時), 1: X-U 軸マスタ・スレーブエリア機能有効
4	V_EN	0: V-B 軸マスタ・スレーブエリア機能無効 (POW ON 時), 1: V-B 軸マスタ・スレーブエリア機能有効

V-B 軸は 4 軸製品にありません.

(2) X-U マスタ・スレーブエリア用コンパレータ選択(XSYNC_C)

Read/Write コマンド: 96h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSC1	BSC0	ASC1	ASC0	WSC1	WSC0	VSC1	VSC0	USC1	USC0	ZSC1	ZSC0	YSC1	YSC0	XSC1	XSC0

ビット	名 称	内 容
1-0	XSC1-0	00: 選択なし (POW ON 時), 01: XCMP4, 10: XCMP5, 11: XCMP4 AND XCMP5
7-2	nSC1-0	00: 選択なし (POW ON 時), 01:nCMP3 AND nCMP4, 10:nCMP3 AND nCMP5, 11:nCMP4 AND nCMP5 (n は Y-U)
9-8	VSC1-0	00: 選択なし (POW ON 時), 01: VCMP4, 10: VCMP5, 11: VCMP4 AND VCMP5
15-10	nSC1-0	00: 選択なし (POW ON 時), 01:nCMP3 AND nCMP4, 10:nCMP3 AND nCMP5, 11:nCMP4 AND nCMP5 (n は W-B)

(3) V-B マスタ・スレーブエリア用コンパレータ選択(VSYNC_C)

Read/Write コマンド: 98h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSC1	BSC0	ASC1	ASC0	WSC1	WSC0	VSC1	VSC0	USC1	USC0	ZSC1	ZSC0	YSC1	YSC0	XSC1	XSC0

ビット	名 称	内 容
1-0	XSC1-0	00: 選択なし (POW ON 時), 01: XCMP4, 10: XCMP5, 11: XCMP4 AND XCMP5
7-2	nSC1-0	00: 選択なし (POW ON 時), 01:nCMP3 AND nCMP4, 10:nCMP3 AND nCMP5, 11:nCMP4 AND nCMP5 (n は Y-U)
9-8	VSC1-0	00: 選択なし (POW ON 時), 01: VCMP4, 10: VCMP5, 11: VCMP4 AND VCMP5
15-10	nSC1-0	00: 選択なし (POW ON 時), 01:nCMP3 AND nCMP4, 10:nCMP3 AND nCMP5, 11:nCMP4 AND nCMP5 (n は W-B)

V-B 軸は 4 軸製品にありません。

(4) マスタインコード設定(J3_SEL)

Read/Write コマンド: a4h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	J3 SYNC	J3V1	J3V0	J3X1	J3X0

ビット	ビット名	内 容
1-0	J3X1,0	00:X 軸インコードが X-U マスタインコードとして使用されます。(POW ON 時) 10:B 軸インコードが X-U マスタインコードとして使用されます。 J3SYNC=0 の時 01: J3 コネクタ SYNCA/B 入力がマスタインコードとして使用されます。 (J3SYNC=1 の時は設定禁止) J3SYNC=1 の時 11: 予約。
3-2	J3V1,0	00:V 軸インコードが V-B マスタインコードとして使用されます。(POW ON 時) 10:B 軸インコードが V-B マスタインコードとして使用されます。 J3SYNC=0 の時 01: J3 コネクタ SYNCA/B 入力が V-B マスタインコードとして使用されます。 (J3SYNC=1 の時は設定禁止) 11: Z 軸指令パルスが V-B マスタインコードとして使用さればます。
4	J3SYNC	0: J3 コネクタ SYNCA/B から Z 軸指令パルス出力を禁止します。(POW ON 時) 1: J3 コネクタ SYNCA/B から Z 軸指令パルスを出します。

V-B 軸は 4 軸製品にありません。

(5) X-U マスタ・スレーブ機能設定(SYNC_SET1)

Read/Write コマンド: f0h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPW3	SPW2	SPW1	SPW0	C5XS	SPDR	SPM1	SPM0	SYEU	SYEZ	SYEY	SYEX	SBMS	SBME	SMD1	SMD0

ビット	名 称	内 容
1-0	SMD1-0	マスタ・スレーブ機能のモード設定を行います。 00:マスタ・スレーブ機能無効。 01:マスタ追従モードです。 10:並行 2 軸制御機能モードです。 11:スレーブ拡張モードです。
2	SBME	スレーブ拡張時のサブマスタ有効/無効 0:無効, 1:有効
3	SBMS	スレーブ拡張時のサブマスタ軸を選択します。 0:Z軸, 1:U軸
4	SYEX	X 軸をスレーブにします。 0:無効, 1:スレーブ(スレーブ拡張時使用可能)
5	SYEY	Y 軸をスレーブにします。 0:無効, 1:スレーブ
6	SYEZ	Z 軸をスレーブにします。 0:無効, 1:スレーブ
7	SYEU	U 軸をスレーブにします。 0:無効, 1:スレーブ
9-8	SPM1-0	マスタ追従モードでのマスタエンコーダ(X 軸)のA相/B相入力仕様を設定します。 00:90°位相差 1 通倍 (A 相入力位相が進んでいるときにカウントアップ) 01:90°位相差 2 通倍 (") 10:90°位相差 4 通倍 (") 11:A 相の立ち上がりでカウントアップ, B 相の立ち上がりでカウントダウン
10	SPDR	1:マスタ追従モードでのマスタエンコーダのカウント方向を逆にします。
11	C5XS	マスタ・スレーブエリア機能の有効/無効 1:マスタ・スレーブエリア用コンパレータ条件成立時マスタ・スレーブ機能有効。
15-12	SPW3-0	マスタ追従モードでのスレーブ軸指令パルスの出力幅を設定します。 0000:0.25us, 0001:0.5 us, 0010:0.75us, 0011:1.0 us, 0100:1.25us, 0101:2.5 us, 0110:5.0 us, 0111:7.5 us, 1000:25 us, その他:0.05us

(6) V-B マスタ・スレーブ機能設定(SYNC SET2)

Read/Write コマンド:f2h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPW3	SPW2	SPW1	SPW0	C5XS	SPDR	SPM1	SPM0	SYEB	SYEA	SYEW	SYEV	SBMS	SBME	SMD1	SMD0

ビット	名 称	内 容
1-0	SMD1-0	マスタ・スレーブ機能のモード設定を行います。 00:マスタ・スレーブ機能無効. 01:マスタ追従モード. 10:並行 2 軸制御機能モード. 11:スレーブ拡張モード.
2	SBME	スレーブ拡張時のサブマスタ有効/無効 0:無効, 1:有効
3	SBMS	スレーブ拡張時のサブマスタ軸を選択します. 0:A軸, 1:B軸
4	SYEV	V 軸をスレーブにします. 0:無効, 1:スレーブ(スレーブ拡張時使用可能)
5	SYEW	W 軸をスレーブにします. 0:無効, 1:スレーブ
6	SYEA	A 軸をスレーブにします. 0:無効, 1:スレーブ
7	SYEB	B 軸をスレーブにします. 0:無効, 1:スレーブ
9-8	SPM1-0	マスタ追従モードでのマスタエンコーダ(V 軸)のA相/B相入力仕様を設定します。 00:90° 位相差 1 通倍 (A 相入力位相が進んでいるときにカウントアップ) 01:90° 位相差 2 通倍 (") 10:90° 位相差 4 通倍 (") 11:A 相の立ち上がりでカウントアップ, B 相の立ち上がりでカウントダウン
10	SPDR	1:マスタ追従モードでのマスタエンコーダのカウント方向を逆にします.
11	C5XS	マスタ・スレーブエリア機能の有効/無効 1:マスタ・スレーブエリア用コンパレータ条件成立時マスタ・スレーブ機能有効.
15-12	SPW3-0	マスタ追従モードでのスレーブ軸指令パルスの出力幅を設定します。 0000:0.25us, 0001:0.5 us, 0010:0.75us, 0011:1.0 us, 0100:1.25us, 0101:2.5 us, 0110:5.0 us, 0111:7.5 us, 1000:25 us, その他:0.05us

V-B 軸は 4 軸製品にありません。

3.14.5 マスタ追従

マスタ軸(Master-axis) のエンコーダ入力(Encoder feedback input) に、スレーブ軸(Slave-axis)が同期動作します。

(1) ブロック図

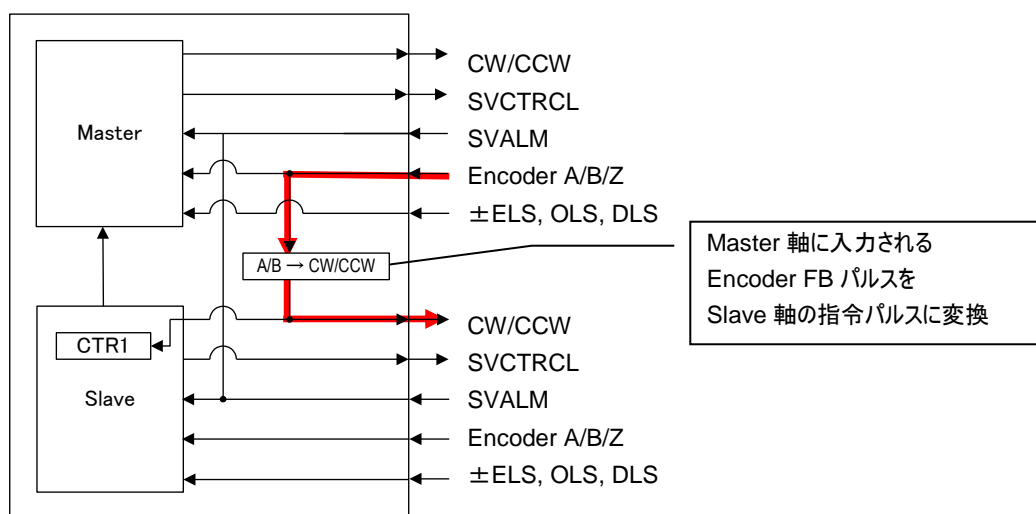


図 3.14-1 マスタ追従モードブロック図

(2) 接続構成

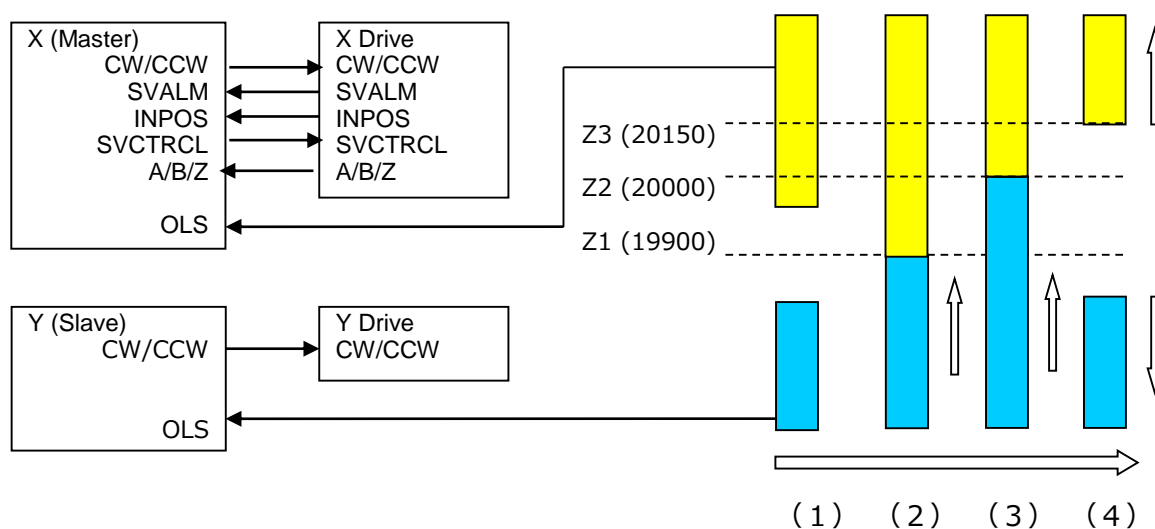


図 3.14-2 マスタ追従接続構成

上の(1)～(4)は各段階に於ける各軸の動作モードを示す。

	(1) ～ (2)	(2) ～ (3)	(3) ～ (4)
X 動作モード	(PTP)	PTP	PTP
Y 動作モード	PTP	MPG	PTP

※MPG：手動パルス発生器によつての移動。手動送り。

注：(ELS、SVAL などによる)エラー停止となった場合、軸が停止します。

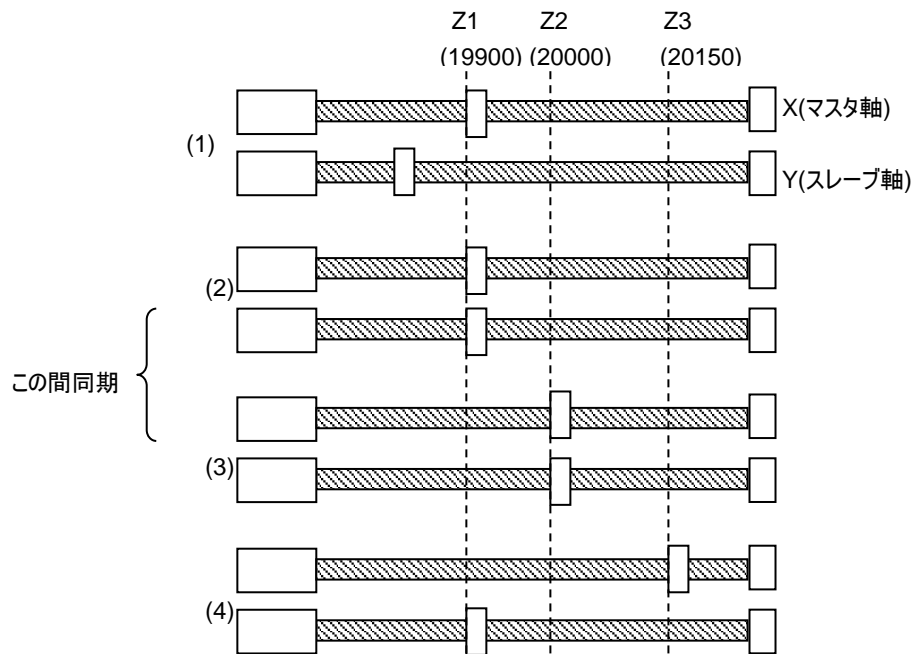
RMD.bit24= 1 (STP 入力による停止)と RMD.bit25=1 (エラー停止時の STP 出力)を設定する必要があります



注 意

- マスタ軸とスレーブ軸の SVALM 極性は同じである必要があります。
- Error-stop(Stop by ELS, SVALM)時に軸を停止するために RMD の bit24=1(Stop by STP),bit25=1(STP Output by error-stop)の設定が必要です。
- スレーブ軸の RMG の設定は 2, RFH の設定は 65535 を設定してください。
- スレーブ軸の CTR1 の更新はマスタ軸に対し、約 $1\mu\text{s}$ の遅れがあります。

(3) 動作例



上記(1)～(4)各フェーズでの各軸動作モード(RMD)

	(1)～(2)	(2)～(3)	(3)～(4)
X 軸動作モード	(位置決め動作)(41h)	位置決め動作(41h)	位置決め動作(41h)
Y 軸動作モード	位置決め動作(41h)	パルス入力による連続動作(01h)	位置決め動作(41h)

(2)～(3)の間がマスタ追従

(4) 設定例

レジスタ	X-軸	Y-軸	オプション ポート	備考
環境設定				
RENV1	104248C4h	104248C4h		方向変化タイマ OFF
RENV2	0320fd45h	0320fd55h		EA/EB:4 通倍, PA/PB: UP/DOWN PA/PB 入力 は CW/CCW これは指令パルスカウント用
RENV3	00f00002h	00f00002h		
RENV5	00000000h	00000800h		Y 軸 ELS 入力 でエラーステータス反映. この設定と RMD のビット 25=1 の設定で STP 自動出力が可能
FPGA				
SYNC_SET1 (f0h)			7221h	マスタ追従モード, Y が Slave 軸, マスタエンコーダ 90°位相差 4 通倍 Y 軸指令パルス幅 7.5us(66.6kpps 相当) 通常動作時は 0 に設定
速度設定				
PRFL	5d	5d		5pps
PRFH	10000d	10000d		10kpps
PRUR	28d	28d		加速時間: 59msec
PRMG	299d	299d		速度倍率 x1.
RFA	5d	5d		5pps

初回のみ、Y 軸を、PRMD.bit6-0 (動作モード)=1 (手動モード)、最大速度 (PRFH=65535, PRMG=2) にセットし、Y 軸スタートする。(これは、指令パルスカウント用)

(5) ステータスの運用

原点復帰時、通常動作時は各々通常のステータス運用。
マスタ追従動作時は X 軸のステータス運用。

(6) 原点復帰

各々通常通り。

(7) 手 順

1. Z1 の位置まで Y 軸位置決め動作
2. Y 軸 Z1 までの位置決め完了でマスタ追従モードに切り替え。
3. X 軸を Z2 まで位置決め。
4. X 軸 Z2 までの位置決め完了で通常モードに切り替え。
5. その後、各々位置決め。

(8) 同期精度について

同期中に、X 軸エンコーダの入力パルスよりも、Y 軸出力パルスは、約 1 μ sec の遅延を有します。

(9) コード例

この例では、レジスタとオプションのポートがすでに初期化されていることを前提としています。
以下は、前ページ (7) 4 (7) 2 の手順です。

```
//Master tracking
//(7) 2 in the previous page.
    //Y-axis RENV2(PA/PB input:up/down)
    cp530_wReg( hDev, Y_AX, 0x9D, 0x0320FD55 );
    //SYNC_SET1(Master tracking mode, Y-axis is slave,
        //Master encoder:x4, Y-axis command pulse width:7.5us)
        // cp530_wPortW( hDev, 0xF0, 0);           // DLL version 4.0.0.0 previously
    cp530_wMasterSlavePort( hDev, 0, 0x7221 );      // DLL version 4.0.0.0 or later
    // Y-axis pulsar mode
    cp530_wReg( hDev, Y_AX, 0x87, 1 );              //PRMD
    // Y-axis maximum velocity
    cp530_wReg( hDev, Y_AX, 0x82, 0xFFFF );         //PRFH
    cp530_wReg( hDev, Y_AX, 0x85, 2 );              //PRMG
    cp530_wCmdW( hDev, Y_AX, 0x51 );

//(7) 3 in the previous page.
    // X-axis Move Relative
    hcp530_WritPos( hDev, X_AX, 10000 );             // X-axis distance
    hcp530_WritFHSpd( hDev, X_AX, 10000 );          // X-axis velocity
    hcp530_WritOpeMode( hDev, X_AX, 0x41 );         // X-axis operational mode
    hcp530_AccStart( hDev, 1 );                     // X-axis start

//(7) 4 in the previous page.
    //SYNC_SET1(Disable master-slave)
    // cp530_wPortW( hDev, 0xF0, 0);           // DLL version 4.0.0.0 previously
    cp530_wMasterSlavePort( hDev, 0, 0 );          // DLL version 4.0.0.0 or later
    cp530_wCmdW( hDev, Y_AX, 0x49 );               //Y-axis stop
```

3.14.6 マスタ・スレーブエリア機能

「3.14.5 マスタ追従」の動作をコンパレータにより範囲設定します。

次の3種類のモードはマスタ・スレーブ機能用に利用可能:

モード	仕様
マスタ追従	X 軸エンコーダ(マスタエンコーダ)入力を、スレーブ軸(Y または U 軸)に対する同期のとれた指令パルスとして出力することが可能です。 V 軸エンコーダ入力をスレーブ軸(W または B 軸)の指令パルスとして出力可能です。
並行 2 軸制御機能	X 軸を、その指令パルスをスレーブ軸(Y,Z 及び U)の指令パルス出力として出力するためのマスタ軸として定義可能です。 V 軸を、その指令パルスをスレーブ軸(W,A 及び B)の指令パルス出力として出力するためのマスタ軸として定義可能です。
スレーブ拡張	Z または U 軸を、その指令パルスをスレーブ軸(B 軸に対する U 軸,及び他ボードの B 軸に対する X 軸)のパルス入力として入力するためのサブマスタ軸として定義可能です。 この機能は、手動パルスモードにスレーブ軸の動作モードを設定することにより、サブマスタ軸およびスレーブ軸の間の同期を可能にします。

この機能を用いることで、その範囲(マスタ・スレーブエリア)の事前同期を指定することが出来ます。

(1) 初期設定例

設定箇所	X 軸	Y 軸	オプション ポート	備 考
RENV2	0320f145h	0320fd55h		EA/EB:4 通倍, PA/PB:UP/DOWN, X はマスタ・スレーブエリア機能有効(b11,10=00),
RENV4	11000000h	00000000h		XCMP4 条件 CMP4>CTR2
RENV5	00000000h	00000800h		Y 軸 ELS 入力でエラーステータス反映. この設定と RMD のビット 25=1 の設定で STP 自動出力が可能
CMP4	20000	——		XCMP4=20000 XCTR2=20000 で同期終了
SYNC_C(94h)			0001h	X 軸マスタ・スレーブエリア機能有効
XSYNC_C(96h)			0001h	マスタ・スレーブエリア機能用コンパレータを XCMP4
SYNC_SET1 (f0h)			7A21h	マスタ追従モード, Y がスレーブ マスタエンコーダ入力 90°位相差 4 通倍 マスタ・スレーブエリア機能有効 Y 軸指令パルス幅 7.5us(66.6kpps 相当)

(2) ステータスの運用

原点復帰時, 通常動作時は各々通常のステータス運用。

マスタ追従動作時は X 軸のステータス運用。

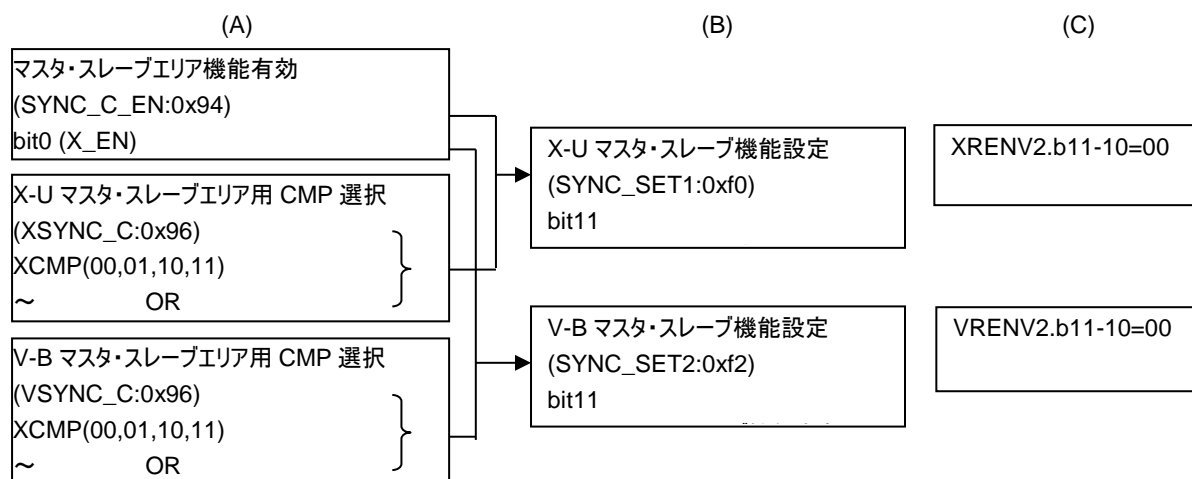
(3) 原点復帰

各々通常通り。

(4) 手 順

1. Z1 の位置まで Y 軸位置決め動作
2. Y 軸 Z1 までの位置決め完了でマスタ追従モードに切り替え。
3. X 軸を Z3 まで位置決め. この間.X 軸 Z2 到達で Y 軸同期動作自動終了して停止.
4. Y 軸を通常モードにして位置決め動作(戻し)。

マスタ・スレーブエリア機能設定手順は下記手順で行います。



マスタ・スレーブエリア機能無効→有効設定時は(C)→(A)→(B)の手順で設定。

マスタ・スレーブエリア機能有効→無効設定時は(B)→(A)→(C)の手順で設定。

(5) マスタ・スレーブエリア機能のコード例

この例は前ページの手順 3 に対応します。

```

//Mster-slave function setting
//cp530_wPortW( hDev, 0xF0, 0x7A21);
cp530_wMasterSlavePort( hDev, 0, 0x7A21 );
// DLL version 4.0.0.0 previously
// DLL version 4.0.0.0 or later
} (B)

//enable master-slave-area
cp530_wPortW( hDev, 0x94, 0x0001 );
//XCMP4
cp530_wPortW( hDev, 0x96, 0x0001 );
//Set X-axis RCMP4
cp530_wReg( hDev, X_AX, 0xAA, 20000 );
//Set X-axis RENV4(CMP4>CTR2)
cp530_wReg( hDev, X_AX, 0x9F, 0x11000000 );
//Set X-axis RENV2(bit11-10=00)
cp530_wReg( hDev, X_AX, 0x9D, 0x0320F155 );
} (A)

//Y-axis RENV2(PA/PB input:up/down)
cp530_wReg( hDev, Y_AX, 0x9D, 0x0320FD55 );
// Y-axis pulsar mode
cp530_wReg( hDev, Y_AX, 0x87, 1 );
// Y-axis maximum velocity
cp530_wReg( hDev, Y_AX, 0x82, 0xFFFF );
cp530_wReg( hDev, Y_AX, 0x85, 2 );
cp530_wCmdW( hDev, Y_AX, 0x51 );
//PRMD
//PRFH
//PRMG
//Y-axis Start (for CTR1 count)

// X-axis Move Relative
hcp530_WritPos( hDev, X_AX, 20150 );
hcp530_WritFHSpd( hDev, X_AX, 10000 );
hcp530_WritOpeMode( hDev, X_AX, 0x41 );
hcp530_AccStart( hDev, 1 );
// X-axis distance
// X-axis velocity
// X-axis operational mode
// X-axis start
  
```



注 意

仮に追従スタート時のスレーブ軸の座標を 10001, マスタ・スレーブエリアを $10000 < XCTR2$ とした場合(X 軸=マスタ軸), $XCTR2=10001$ になった時から追従を開始します。

したがって $XCTR2$ が 10001 から仮に 20000 まで動作した場合, スレーブ軸も 10001 から 20000 まで動作します。

次にマスタ軸(X 軸)を 20000 から 10001 まで動作させるとスレーブ軸は 20000 から 10001 まで動作し, マスタ軸(X 軸)が 10000 になった時点でマスタ・スレーブ機能は OFF になりますが, スレーブ軸は 10000 まで動作します。

この動作を繰り返すとスレーブ軸の座標はマスタ軸(X 軸)が, 10001 から 10000 に動作するたびに 1 パルスずつ減っていきます。

ここで説明している例ではマスタ・スレーブエリア機能によるマスタ・スレーブ機能 OFF 時にスレーブ軸を通常の位置決め動作で行っているため, 位置のずれが積算していくことはありません。

3.14.7 並行 2 軸制御機能

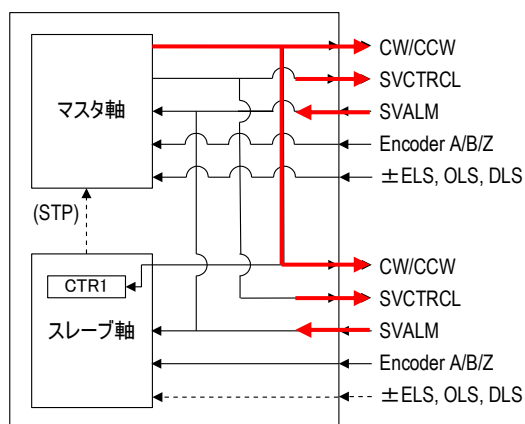
マスタ軸(Master-axis) の指令出力とスレーブ軸(Slave-axis)の指令出力が同期します。
例えば、HPCI-CPD578N では、ガントリシステムのための並行軸をドライブします。



注 意

- マスタ軸とスレーブ軸の SVALM 極性は同じである必要があります。
- スレーブ軸の RMG の設定は 2, RFH の設定は 65535 を設定してください。
- スレーブ軸の CTR1 の更新はマスタ軸に対し、約 $1\mu\text{s}$ の遅れがあります。

(1)ブロック図



赤のラインが同期します。

図 3.14-4 並行 2 軸制御機能モードブロック図

(2)接続構成

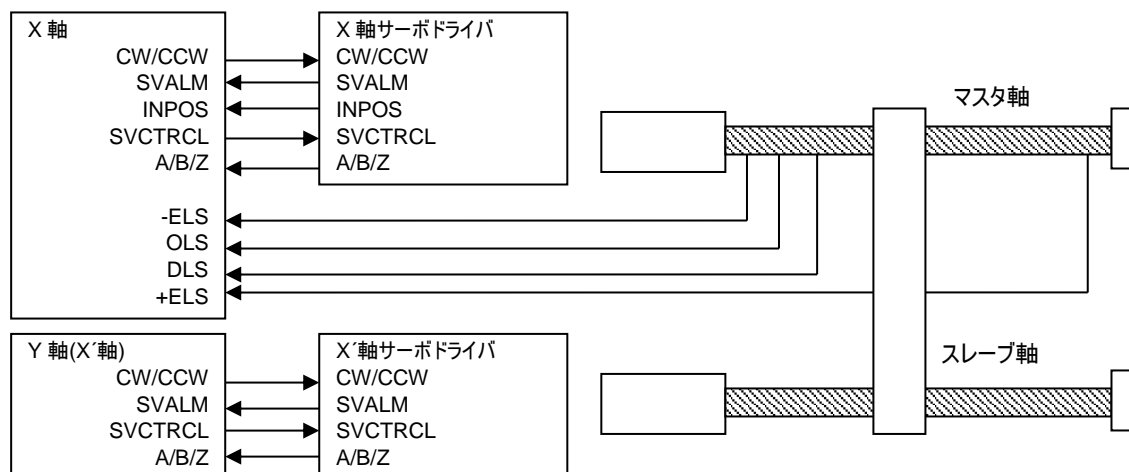


図 3.14-5 並行 2 軸制御機能モード接続構成

(3)初期設定例

設定箇所	X 軸	Y 軸	オプションポート	備 考
RENV1	xxxxxxx4h	xxxxxxx4h		指令パルス形式は個別パルス形式にする
RENV2	0320fd55h	0320fd55h		PA/PB:UP/DOWN, その他はデフォルト値
RENV3	00f00002h	00f00002h		
SYNC_SET1			0022h	並行 2 軸制御モード,Xと Y で同期

初回のみ、Y 軸を、PRMD.bit6-0 (動作モード)=1 (手動モード)、最大速度(PRFH=65535, PRMG=2)にセットし、Y 軸スタートする。(これは、指令パルスカウント用)

(4)ステータスの運用

通常は X 軸のステータスのみ監視すれば良い。

SVALM 停止時は X,Y のエラーステータスを確認。

X,Y が SVALM ならば Y の SVALM, X のみ SVALM ならば X の SVALM.

エラー要因を取り除いた後の再スタート時はスレーブ軸 FH 定速スタート後、マスタ軸スタート。

(5)原点復帰

X 軸のセンサと Z 相で行う。CTRCL 出力は原点復帰完了時に X 軸と Y 軸から同時出力。

原点復帰完了時は YCTR はリセットされないため、ソフトウェアによるリセットが必要。

(6)並行 2 軸原点復帰コード例

この例は、レジスタ及びオプションポートの初期化が完了していることを前提としています。

この例は、並行 2 軸原点復帰です。

原点復帰完了時点での SVCTRCL の自動出力を設定する場合、は事前設定が必要です。

SVCTRCL は、X 軸(マスタ軸)の原点復帰完了時点で、X 及び Y の両軸に対し自動的に出力されます。

```
//Parallel 2-axis homing
//Master-slave function setting
// cp530_wPortW( hDev, 0xF0, 0x0022);           // DLL version 4.0.0.0 previously
cp530_wMasterSlavePort( hDev, 0, 0x0022 );       // DLL version 4.0.0.0 or later
// X-axis(Master) setting
hcp530_WritPos( hDev, X_AX, 10000 );             // X-axis distance
hcp530_WritFHSpd( hDev, X_AX, 10000 );          // X-axis velocity
hcp530_WritOpeMode( hDev, X_AX, 0x1D );         // X-axis operational mode
hcp530_AccStart( hDev, 1 );                     // X-axis start
```

Each counters of slave axis set to the X-axis when the homing completion of X-axis.

If necessary, after homing completion, release synchronization by using following code, and correct the position of each axes by positioning operation, and reset each counters.

```
// cp530_wPortW( hDev, 0xF0, 0);                // DLL version 4.0.0.0 previously
cp530_wMasterSlavePort( hDev, 0, 0 );           // DLL version 4.0.0.0 or later
```

(7)通常動作

X 軸の RMV, RMD を設定.

Y 軸は RMG=2, RFH=65535(動作速度最大値を設定), RMD.b7-0(動作モード)を 01h(連続手パ)にする.

(Y 軸指令カウンタ動作の為の設定)

初回またはエラー発生後の再スタート時のみ Y 軸をスタートする.

その他は X 軸のみスタート.

(8)並行2軸位置決め動作コード例

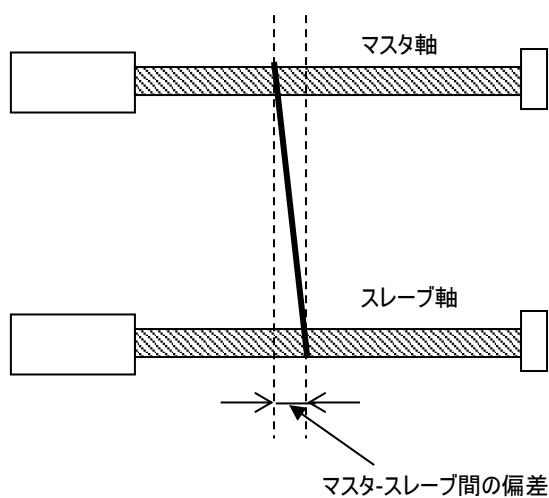
```
//Master-slave function setting
cp530_wMasterSlavePort (hDev, 0, 0x0022);

// for CTR1 count
//Y-axis RENV2 (PA/PB input: up/down) for CTR1 count
cp530_wReg (hDev, Y_AX, 0x9D, 0x0320FD55);

// Y-axis pulsar mode
cp530_wReg (hDev, Y_AX, 0x87, 1);           //PRMD
// Y-axis maximum velocity
cp530_wReg (hDev, Y_AX, 0x82, 0xFFFF);      //PRFH
cp530_wReg (hDev, Y_AX, 0x85, 2);           //PRMG
cp530_wCmdW (hDev, Y_AX, 0x51);             //Y-axis start

// X-axis (Master) setting
hcp530_WritPos (hDev, X_AX, 10000);           // X-axis distance
hcp530_WritFHSpd (hDev, X_AX, 10000);        // X-axis velocity
hcp530_WritOpeMode (hDev, X_AX, 0x41);       // X-axis operational mode
hcp530_AccStart (hDev, 1);                   // X-axis start
```

(9) 偏差エラーの説明



マスタ-スレーブ間の偏差が設定値を超えると、両軸は自動的に停止する。

(10) 偏差エラー設定

STP 信号用偏差エラー出力

(BAR3+A6H : STP_DEV) (Write/Read) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	x	x	x	x	x	x	x	OREC	x	x	x	x	x	x	x

ビット	記号	説明	備考
6-0	Reserved		
7	OREC	STP 信号用偏差エラー出力 0: STP 信号用偏差エラー出力しない, 1: STP 信号用偏差エラー出力	
15-8	Reserved		

カウンタ入力選択設定

(BAR3+AEH : CTR_SEL) (Write/Read) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	C2E2	C2E1	C2E0	0	C2C2	C2C1	C2C0	0	C1E2	C1E1	C1E0	0	C1C2	C1C1	C1C0

ビット	記号	説明	備考
3-0	C1Cx	CTR1 の CH(入力元)設定 000 : CH1 (X) 001 : CH2 (Y) 010 : CH3 (Z) 011 : CH4 (U) 100 : CH5 (V) 101 : CH6 (W) 110 : CH7 (A) 111 : CH8 (B)	
6-4	C1Ex	CTR1 のカウント方法 000 : A / B 4 通倍 001 : Up / Down 010 : Puls / Dir 011 : Dir / Puls 101 : A / B 1 通倍 110 : A / B 2 通倍	
11-8	C2Cx	CTR2 の入力元設定	
15-12	C2E2	CTR2 のカウント方法	

偏差エラーステータス

(BAR3+AAh : DEV_ERR) (Read) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ERR

ビット	記号	説明	備考
0	ERR	1: 偏差エラー	

ExCounter1, ExCounter2

(BAR3+B0H : CTR_1, BAR3+B4H : CTR_2) (Read/Write) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

ビット	記号	説明	備考
31-0		32 ビットエンコードのフィードバックカウント	

偏差許容範囲

(BAR3+B8H : EC_CMP) (Write) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C15	C14	C13	C12	C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0

ビット	記号	説明	備考
15-0	Cx	偏差エラー設定許容範囲(0~65535)	

設定値と FPGA カウンタの間の偏差は比較され、その違いが設定値と同じか大きい時、停止が可能になります。

許容範囲を超えたときの DLS

(BAR3+BAH : EC_DLS) (Write) Initial value : 0000 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DB1	DB0	DA1	DA0	DW1	DW0	DV1	DV0	DU1	DU0	DZ1	DZ0	DY1	DY0	DX1	DX0

ビット	記号	説明	備考
1,0	DXx	対 X 軸の DLS 00 : 偏差エラー時に何も出力しない 10 : 偏差エラー時に負論理を出力する 11 : 偏差エラー時に正論理を出力する s	
3,2	DYx	対 Y 軸の DLS	
5,4	DZx	対 Z 軸の DLS	
7,6	DUx	対 U 軸の DLS	
9,8	DVx	対 V 軸の DLS	
11,10	DWx	対 W 軸の DLS	
13,12	DAx	対 A 軸の DLS	
15,14	DBx	対 B 軸の DLS	

偏差エラー発生時、内部 DLS 信号を出力することが可能です。

(11) 偏差エラー設定コード例

軸を初期化する際、STP を有効にして下さい。これは重要です。以下の設定はマスタ軸の設定が必須です。

```
//Set PRMD
// bit24=1: Enable STP
// bit25=1: When stopped by STP, STP output to the other axis
#define WPRMD 0x87 //Write PRMD
hcp530_wReg (hDev, X_AX, WPRMD, 0x0B008041);
```

```
-----

#define EXCTR1 0 // FPGA Counter1
#define EXCTR2 1 // FPGA Counter2
#define AB4 0 // A/B (x4)
#define CWCCW 1 // CW/CCW
#define PLSDIR 2 // pulse + direction
```

```
// Master axis is X-axis, Slave axis is Y-axis.
hcp530_SetExCtrInput (hDev, EXCTR1, X_AX, AB4);
hcp530_SetExCtrInput (hDev, EXCTR2, Y_AX, AB4);
```

```
//The following is setting for STP by deviation error.
//Enable STP by deviation error
hcp530_SetSTPbyDevError (hDev, 1);
```

```
//set deviation width (critical value of deviation = 10pulse)
hcp530_SetDevErrorThreshold (hDev, 10);
```

```
-----

//Disable STP by deviation error
hcp530_SetSTPbyDevError (hDev, 0);
```

3.14.8 スレーブ拡張モード

サブマスタ軸(Sub-Master-axis) の指令出力がスレーブ軸(Slave-axis)の手動パルス入力に入力されます。
スレーブ軸を手動パルス連続モードに設定しサブマスタ軸と同期させます。

(1) 接続構成

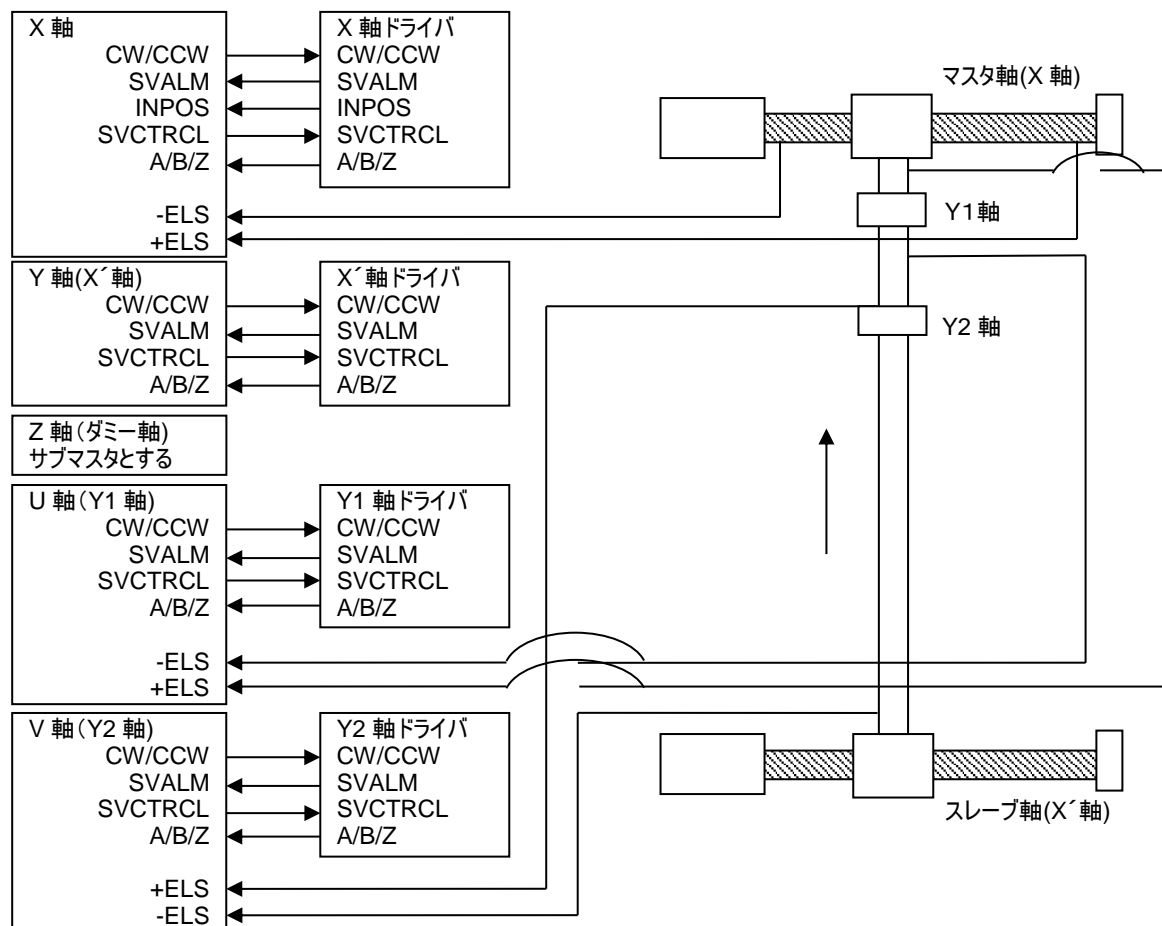


図 3.14-6 スレーブ拡張モード接続構成

(2) 初期設定例

設定箇所	X 軸	Y 軸	Z 軸	U 軸	V 軸	オプション ポート	備 考
RENV2	0320fd55h	0320fd55h	0320fd55h	0320fd55h	0320fd55h		PA/PB:UP/DOWN
RENV5	---	---	---	00000800h	00000800h		U,V 軸は動作方向 の ELS で停止
SYNC_SET1 (f0h)						30a6h	並行 2 軸制御モード XとY が並行 2 軸 サブマスタ:Z 軸 U 軸指令パルス幅 1us(500kpps)
SYNC_SET2 (f2h)						3013h	スレーブ拡張モード V がスレーブ V 軸指令パルス幅 1us(500kpps)
J3_SEL(a4h)						000ch	Z 軸指令パルスが V-B マスタエンコーダ と同じになる。

(3) ステータスの運用

XY 軸の動作は「並行 2 軸制御機能モード」と同じ。XY 軸と Z 軸の動作は X と Z の補間動作時のステータス運用。

(4) 原点復帰

XY 軸の動作は「並行 2 軸制御機能モード」と同じ。
Z, U, V は通常通り。

(5) XY 軸と Z(U, V)の補間設定

1. X 軸の PRMV, PRMD を設定。動作モード PRMD.MOD(bit7-0)は補間。
2. Y 軸は動作モード PRMD.MOD(bit7-0)を 01h(連続手入)にする。(CTR1 を動作させるため)
3. Z 軸は PRMV, PRMD を設定。動作モード PRMD.MOD(bit7-0)は補間。サブマスタ軸とする。
4. サブマスタ軸のスレーブ軸(U,V のいずれか)は PRMD.MOD(bit7-0)を 01h(連続手入)にする。
また PRMD.bit24=1(STP 入力で停止), PRMD.bit25=1(異常停止時に STP 出力)
スレーブ軸の RMG, RFH はサブマスタ軸と同じ設定にする。
5. SYNC_SET1, SYNC_SET2 でサブマスタ軸のスレーブ選択を行う。
6. サブマスタ軸のスレーブをスタートさせる。
7. X 軸, Y 軸, Z 軸をスタートする。(コマンド 0x07xxh)

(6) スレーブ拡張モードコード例

この例は、レジスタ及びオプションポートの初期化が完了していることを前提としています。

```
// Z command pulse -> V-B synchronize pulse
cp530_wPortW ( hDev, 0xA4, 0x000C );

//U, V axis RENV5(Enable ELS detection error(when pulsar mode))
cp530_wReg ( hDev, U_AX, 0xA0, 0x00000800 );
cp530_wReg ( hDev, V_AX, 0xA0, 0x00000800 );

//Master-slave function setting
cp530_wMasterSlavePort ( hDev, 0, 0x30A6 );           //X-U
cp530_wMasterSlavePort ( hDev, 1, 0x3013 );           //V-B

// for CTR1 count
//Y, U, V axis RENV2 (PA/PB input: up/down)
cp530_wReg( hDev, Y_AX, 0x9D, 0x0320FD55 );
cp530_wReg( hDev, U_AX, 0x9D, 0x0320FD55 );
cp530_wReg( hDev, V_AX, 0x9D, 0x0320FD55 );

// Y-axis pulsar mode
cp530_wReg( hDev, Y_AX, 0x87, 1 );                     //PRMD
// Y-axis maximum velocity
cp530_wReg( hDev, Y_AX, 0x82, 0xFFFF );               //PRFH
cp530_wReg( hDev, Y_AX, 0x85, 2 );                     //PRMG
cp530_wCmdW( hDev, Y_AX, 0x51 );

// Set to U-axis (the same as the Y-axis)
--- to leave out ---
// Set to V-axis (the same as the Y-axis)
--- to leave out ---

// Linear interpolation(X-Z)
hcp530_WritPos( hDev, X_AX, 10000 );                   // X-axis distance
hcp530_WritPos( hDev, Z_AX, 5000 );                     // Z-axis distance
hcp530_WritFHSpd( hDev, X_AX, 10000 );                 // Set Velocity
hcp530_WritOpeMode( hDev, X_AX, 0x61 );                // X-axis read operational mode
hcp530_WritOpeMode( hDev, Z_AX, 0x61 );                // Z-axis read operational mode
hcp530_AccStart( hDev, 5 );                             // X and Z start
```

3.15 汎用入力/非常停止入力(HPCI-CPD508, HPCI-CPD5016)

3.15.1 HPCI-CPD508

(1) 汎用入力ポートの用途選択設定と設定状態読込(INP_SEL)

使用関数: rPortB/wPortB

Read/Write コマンド: 8Ah

2ビットの組合せで、コネクタ入力信号 IN0(ピン番号:35), IN1(ピン番号:85)を次の3通りのひとつに設定します。

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SEL1	SEL0

ビット	ビット名	内 容
1-0	SEL0 SEL1	00: IN1:V-B 共通の SVALM, IN0:X-U 共通の SVALM (POWON 時)
		01: IN1: 汎用入力 1, IN0: 汎用入力 0
		10(11): IN1:V-B の EMG, IN0:X-U の EMG

下記「図 3.15-1 汎用入力ポートの用途選択設定」を参照

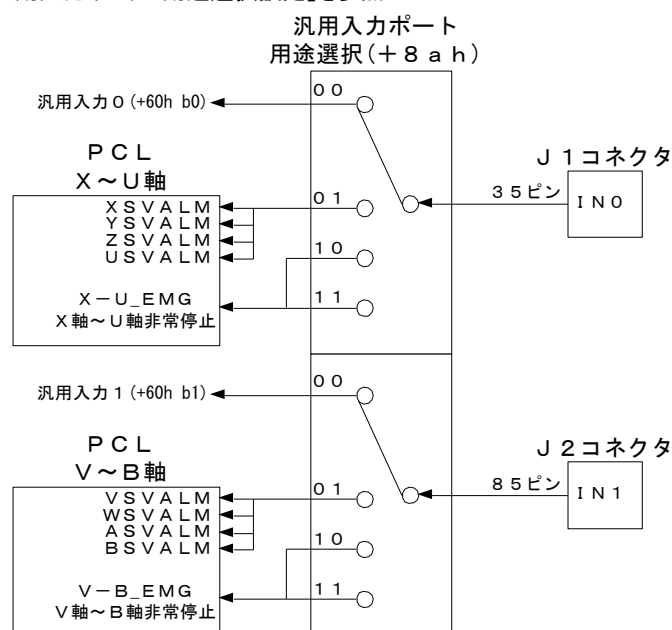


図 3.15-1 汎用入力ポートの用途選択設定

(2) 汎用入力ポートの読込(INPORT)

Read/Write コマンド: 60h

汎用入力ポート(INPORT)の内容は下図の通りです。

7	6	5	4	3	2	1	0
0	0	0	0	0	0	IN1	IN0

ビット	ビット名	内 容
0	IN0	0:IN0 OFF, 1:IN0 ON
1	IN1	0:IN1 OFF, 1:IN1 ON

3.15.2 HPCI-CPD5016

(1) 汎用入力ポートの用途選択設定と設定状態読込(INP_SEL)

使用関数: rPortB/wPortB, rPortW/wPortW

Read/Write コマンド: 8Ah

2ビット(b1,b0)の組合せで、コネクタ入力信号 IN1(ピン番号:35), IN2(ピン番号:85)を次の3通りのひとつに設定します。
2ビット(b3,b2)の組合せで、コネクタ入力信号 IN3(ピン番号:35), IN4(ピン番号:85)を次の3通りのひとつに設定します。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	SEL4	SEL3	SEL2	SEL1

ビット	ビット名	内 容
3-0	SEL1 SEL2	00: IN2:U2-X2 共通の SVALM IN1:U1-X1 共通の SVALM,
		01: IN2: 汎用入力 1, IN1: 汎用入力 0
		10 (11): IN2:U2-X2 の EMG, IN1:X1-U1 の EMG
	SEL3 SEL4	00: IN4:U4-X4 共通の SVALM IN3:U3-X3 共通の SVALM
		01: IN4: 汎用入力 3, IN3: 汎用入力 2
		10 (11): IN4:U4-X4 の EMG, IN3:X3-U3 の EMG

下記「図 3.15-2 汎用入力ポートの用途選択設定」を参照

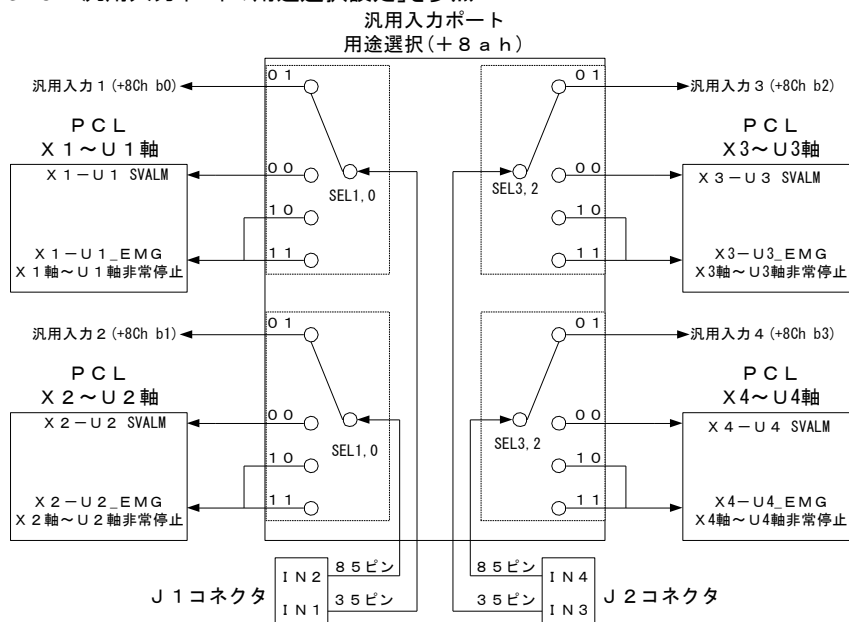


図 3.15-2 汎用入力ポートの用途選択設定

(2) 汎用入力ポートの読込(INPORT)

Read/Write コマンド: 8Ch

汎用入力ポート(INPORT)の内容は下図の通りです。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
*	*	*	*	*	*	*	*	*	*	*	*	IN4	IN3	IN2	IN1

ビット	ビット名	内 容
0	IN1	0: IN1 OFF, 1: IN1 ON
1	IN2	0: IN2 OFF, 1: IN2 ON
2	IN3	0: IN3 OFF, 1: IN3 ON
3	IN4	0: IN4 OFF, 1: IN4 ON

4. PCL 資料

4.1 コマンド

4.1.1 コマンドデータのビット構成

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	実行 U B	軸の Z A Z1 Z2 Z3 Z4	指定 Y W Y1 Y2 Y3 Y4	(SELx) X V X1 X2 X3 X4	コ マ ン ド							

- 実行軸の指定 (SELx)

個々の軸にコマンドを書込む場合はこの 4 ビットは全て"0"にします。

2~4 軸に同一コマンドを書込む場合(同時にスタート, 停止など)は, SELx のビットで書込む軸を指定し, 軸指定した PCL の任意の軸に書き込みます。

4.1.2 スタートコマンド

No.	名称	コマンド(HEX)	備考
1	加速スタート	53	加速→FH 定速→減速
2	FH 定速スタート	51	
3	FH 定速スタート後減速停止	52	FH 定速→減速
4	FL 定速スタート	50	
5	残量 FL 定速スタート	54	
6	残量 FH 定速スタート	55	
7	残量加速スタート	57	
8	残量 FH 定速スタート後減速停止	56	
9	同時スタート信号(STA)出力	06	約 407nsec のワンショット出力

4.1.3 速度変更コマンド

No.	名称	コマンド(HEX)	備考
1	FL 定速瞬時速度変更	40	
2	FH 定速瞬時速度変更	41	
3	FL 速度まで減速	42	
4	FH 速度まで加速	43	

4.1.4 停止コマンド

No.	名称	コマンド(HEX)	備考
1	即停止	49	
2	減速停止	4A	ベース速度動作中は即停止. その他動作中は減速停止.
3	非常停止	05	エラー停止(EMG 入力による停止と同等)
4	同時停止信号(STP)出力	07	約 407nsec のワンショット出力

4.1.5 コントロールコマンド

No.	名称	コマンド(HEX)	備考
1	ソフトウェアリセット	04	任意の軸への書き込みで 4 軸全てのレジスタがリセット (POWON 状態)になります。 オプションポートはリセットされません。
2	CTR1 リセット	20	
3	CTR2 リセット	21	
4	CTR3 リセット	22	
5	CTR4 リセット	23	
6	SVCTRCL 信号出力	24	
7	SVCTRCL 信号リセット	25	SVCTRCL 出力幅レベル時の SVCTRCL 出力リセット
8	動作用プリレジスタキャンセル	26	
9	CMP5 用プリレジスタキャンセル	27	
10	動作用プリレジスタシフト	2B	
11	CMP5 用プリレジスタシフト	2C	
12	カウンタラッチ	29	CTR1～CTR4 を一斉にラッチ
13	PCS コマンド	28	PCS 信号の代わりに PCS を実行
14	SVON 信号 ON	18	
15	SVON 信号 OFF	10	
16	SVRST 信号 ON	19	
17	SVRST 信号 OFF	11	

4.1.6 レジスタ制御コマンド

No	内容	レジスタ			プリレジスタ		
		名称	コマンド(HEX)		名称	コマンド(HEX)	
			読出	書込		読出	書込
1	移動量	RMV	d0	90	PRMV	c0	80
2	ベース速度設定	RFL	d1	91	PRFL	c1	81
3	動作速度設定	RFH	d2	92	PRFH	c2	82
4	加速レートを決定するパラメータ	RUR	d3	93	PRUR	c3	83
5	減速レートを決定するパラメータ	RDR	d4	94	PRDR	c4	84
6	速度倍率設定	RMG	d5	95	PRMG	c5	85
7	減速開始点	RDP	d6	96	PRDP	c6	86
8	動作モード	RMD	d7	97	PRMD	c7	87
9	円弧補間中心位置	RIP	d8	98	PRIP	c8	88
10	加速時 S 字区間設定	RUS	d9	99	PRUS	c9	89
11	減速時 S 字区間設定	RDS	da	9a	PRDS	ca	8a
12	補助速度設定	RFA	db	9b			
13	環境設定 1	RENV1	dc	9c			
14	環境設定 2	RENV2	dd	9d			
15	環境設定 3	RENV3	de	9e			
16	環境設定 4	RENV4	df	9f			
17	環境設定 5	RENV5	e0	a0			
18	環境設定 6	RENV6	e1	a1			
19	環境設定 7	RENV7	e2	a2			
20	カウンタ 1(指令パルス出力)	RCTR1	e3	a3			
21	カウンタ 2(エンコーダ入力)	RCTR2	e4	a4			
22	カウンタ 3(偏差カウンタ)	RCTR3	e5	a5			
23	カウンタ 4(汎用カウンタ)	RCTR4	e6	a6			
24	コンパレータ 1 用データ	RCMP1	e7	a7			
25	コンパレータ 2 用データ	RCMP2	e8	a8			
26	コンパレータ 3 用データ	RCMP3	e9	a9			
27	コンパレータ 4 用データ	RCMP4	ea	aa			
28	コンパレータ 5 用データ	RCMP5	eb	ab	PRCP5	cb	8b
29	イベントマスク設定	RIRQ	ec	ac			
30	カウンタ 1 ラッチデータ	RLTC1	ed				
31	カウンタ 2 ラッチデータ	RLTC2	ee				
32	カウンタ 3 ラッチデータ	RLTC3	ef				
33	カウンタ 4 ラッチデータ	RLTC4	f0				
34	拡張ステータス	RSTS	f1				
35	エラーステータス	REST	f2				
36	イベントステータス	RIST	f3				
37	位置決めカウンタ	RPLS	f4				
38	Z 相カウンタ, 指令速度モニタ	RSPD	f5				
39	減速開始点計算値	RSDC	f6				
40	円弧補間歩進数	RCI	fc	bc	PRCI	cc	8c
41	円弧補間歩進カウンタ	RCIC	fd				
42	補間ステータス	RIPS	ff				

表 4.1-1 レジスタコマンド一覧

4.2 ステータス

4.2.1 メインステータス(MSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPDF	SPRF	SEOR	SCMP5	SCMP4	SCMP3	SCMP2	SCMP1	SSC1	SSC0	SINT	SERR	SEND	0	SRUN	SSCM

bit	名 称	内 容	備 考
0	SSCM	1 = スタートコマンド書き込み済み	スタートコマンド書き込み後、即動作が始まらない場合があります。(条件付スタートのスタート保留状態など)
1	SRUN	1 = RUN 中	
2	---	予約	
3	SEND	1 = 停止中(電源投入直後は'0')	DevOpen 関数呼び出し後は SEND=1 になります。 本ビットの変化タイミングは、INPOS 制御設定により異なります。INPOS 制御の詳細は「 2.3.6 INPOS 制御 」を参照してください。
4	SERR	1 = エラーステータスに要因有	REST 読み出しで 1→0
5	SINT	1 = イベントステータスに要因有	RIST 読み出しで 1→0 イベントマスク(RIRQ)の設定が必要。 DevOpen 関数では RIRQ.b0=1(正常停止時)の設定をしています。
7,6	SSC1,0	実行中の RMD.b17,16	次動作連続実行中の動作確認などに使用
8	SCMP1	1 = CMP1 比較条件成立状態	比較条件成立中の間のみ'1'となります。 (既に通過してしまった場合は'0'となります) 条件が成立したことを監視する場合はイベントステータス(RIST)を使用します。
9	SCMP2	1 = CMP2 比較条件成立状態	
10	SCMP3	1 = CMP3 比較条件成立状態	
11	SCMP4	1 = CMP4 比較条件成立状態	
12	SCMP5	1 = CMP5 比較条件成立状態	
13	SEOR	1 = 位置オーバーライド失敗	本ステータスリードでクリア。停止中に RMV を書込んでも"1"。
14	SPRF	1 = 動作用プリレジスタフル	
15	SPDF	1 = コンパレータ用プリレジスタフル	

*1. MSTS.b13(SEOR)は HPCI-CPD574N, HPCI-CPD578N, HPCIe-CPD674N のみ対応しています。

表 4.2-1 メインステータス(MSTS)

4.2.2 サブステータス(SSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SDLS	SOLS	SMEL	SPEL	SALM	SFC	SFD	SFU	不	使	用	(予	約)	SVRST	SVON

bit	名 称	説 明	備 考
0	SVON	'1' = SVON ON	指令状態のモニタ
1	SVRST	'1' = SVRST ON	
7-2	---	不使用(予約)	
8	SFU	'1' = 加速中	レベル検出
9	SFD	'1' = 減速中	
10	SFC	'1' = 定速動作中	
11	SALM	'1' = SVALM 検出中	
12	SPEL	'1' = +ELS 検出中	
13	SMEL	'1' = -ELS 検出中	入力極性設定が反映される。 A 接設定時はカプラ電流 ON で 1 B 接設定時はカプラ電流 OFF で 1
14	SOLS	'1' = OLS 検出中	
15	SDLS	'1' = DLS 検出中	

表 4.2-2 サブステータス(SSTS)

4.2.3 エラーステータス(REST)

エラーステータスはマスクすることは出来ません。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ESAO	ESPO	ESIP	ESDT	0	ESSD	ESEM	ESSP	ESAL	ESML	ESPL	ESC5	ESC4	ESC3	ESC2	ESC1
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	ESPE	ESEE

bit	名 称	内 容	備 考
0	ESC1	1 = CMP1 条件成立で停止 (+SLS)	
1	ESC2	1 = CMP2 条件成立で停止 (-SLS)	
2	ESC3	1 = CMP3 条件成立で停止	
3	ESC4	1 = CMP4 条件成立で停止	
4	ESC5	1 = CMP5 条件成立で停止	
5	ESPL	1 = +ELS による停止	停止中に+ELS が ON しても 1 になりません。
6	ESML	1 = -ELS による停止	停止中に-ELS が ON しても 1 になりません。
7	ESAL	1 = サーボアラームによる停止	停止中に SVALM が ON しても 1 になりません。
8	ESSP	1 = STP 入力 ON による停止 (または同時ストップコマンド)による停止	次動作使用時はプリレジスタキャンセルコマンド (26h)を書き込んでください。
9	ESEM	1 = EMG 入力 ON による停止	非常停止コマンドによる停止または非常停止機能 を使用している場合
10	ESSD	1 = DLS 検出による減速停止	停止中に DLS が ON しても 1 になりません。
11	未定義	常に'0'	
12	ESDT	1 = 動作データが不正で停止	
13	ESIP	1 = 補間他軸の異常停止による停止	
14	ESPO	1 = パルス用バッファオーバーフローによる停止	(パルス入力×パルス通倍)>動作速度の状態が続 いた場合発生します。
15	ESAO	1 = 補間データのレンジオーバーによる停止	
17,16	ESPE ,ESEE	1 = エンコーダ A/B 相が同時に変化	停止しません。
31-18	未定義	常に'0'	

表 4.2-3 エラーステータス(REST)

4.2.4 イベントステータス(RIST)

イベントマスク(RIRQ)の設定をすることでイベントステータスが有効となります。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISOL	ISLT	ISCL	ISC5	ISC4	ISC3	ISC2	ISC1	ISDE	ISDS	ISUE	ISUS	ISND	ISNM	ISN	ISEN
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	ISSA	ISMD	ISPD	ISSD

bit	名 称	内 容
0	ISEN	正常停止. 本ビットの変化タイミングは、INPOS 制御設定により異なります。 INPOS 制御の詳細は「 2.3.6 INPOS 制御 」を参照してください。
1	ISN	次動作継続スタート。
2	ISNM	動作用プリレジスタフル→空き
3	ISND	CMP5 用プリレジスタフル→空き
4	ISUS	加速開始
5	ISUE	加速終了
6	ISDS	減速開始
7	ISDE	減速終了
12-8	ISCn	CMPn 比較条件成立
13	ISCL	CLR 入力によるカウンタクリア時
14	ISLT	LTC 入力によるカウンタラッチ時
15	ISOL	OLS 入力によるカウンタラッチ時
16	ISSD	DLS 信号 OFF→ON
17	ISPD	+DR 信号 OFF→ON
18	ISMD	−DR 信号 OFF→ON
19	ISSA	STA 信号 OFF→ON
31-20	未定義	常に'0'

表 4.2-4 イベントステータス(RIST)

4.2.5 イベントマスクレジスタ(RIRQ)

イベントステータス(RIST)に対するイベントを設定します。各ビットを 1 にするとイベントステータス(RIST)に反映されます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IROL	IRLT	IRCL	IRC5	IRC4	IRC3	IRC2	IRC1	IRDE	IRDS	IRUE	IRUS	IRND	IRNM	IRN	IREN
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	0	0	IRSA	IRDR	IRSD

bit	名 称	内 容
0	IREN	動作完了報告
1	IRN	次動作継続スタート報告
2	IRNM	動作用プリレジスタフル→空き報告
3	IRND	CMP5 用プリレジ書込み可能報告
4	IRUS	加速開始報告
5	IRUE	加速終了報告
6	IRDS	減速開始報告
7	IRDE	減速終了報告
12-8	IRCn	CMPn 比較条件成立報告
13	IRCL	CLR 入力によるカウンタクリア報告
14	IRLT	LTC 入力によるカウンタラッチ報告
15	IROL	OLS 信号入力時にカウンタラッチ報告
16	IRSD	DLS 信号 OFF→ON
17	IRDR	±DR 信号 OFF→ON
18	IRSA	STA 信号 OFF→ON
19-31	未定義	常に'0'を設定してください

表 4.2-5 イベントマスクレジスタ(RIRQ)

4.2.6 拡張ステータス(RSTS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDIN	SLTC	SCLR	SDRM	SDRP	SEZ	SERC	SPCS	SEMG	SSTP	SSTA	SDIR	CND3	CND2	CND1	CND0
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	0	0	0	0	PFM1	PFM0	PFC1	PFC0	0
															SINP

bit	名称	内 容
0	CND0	動作状態を表す
1	CND1	0000: 停止中
2	CND2	0001: DR 入力待ち
3	CND3	0010: STA 入力待ち 0011: スタートコマンドが発行されていて、軸スタート条件成立待ち状態. 0100: 指定軸の停止によるスタート待ち 0101: サーボ偏差カウンタ・クリアタイム on 中 0110: 指令方向信号切替中(共通パルス指令時方向切替時間 0.2ms) 0111: バックラッシュ補正中(バックラッシュ補正中の時間は補正パルス量と補助速度による) 1000: パルス入力待ち 1001: FA(補助速度)定速で動作中 1010: FL 定速で動作中 1011: 加速中 1100: FH 定速で動作中 1101: 減速中 1110: INPOS 信号待ち 1111: その他
4	SDIR	動作方向(0: +方向, 1: -方向)
5	SSTA	'1'= 同時スタート信号 on 状態
6	SSTP	'1'= 同時停止信号 on 状態
7	SEMG	'1'= EMG 信号入力中
8	SPCS	'1'= PCS 位置決めスタート信号入力中
9	SERC	'1'= サーボ偏差カウンタクリア信号出力中
10	SEZ	'1'= エンコーダ Z 相信号入力中
11	SDRP	'1'= +DR 信号入力中
12	SDRM	'1'= -DR 信号入力中
13	SCLR	'1'= CLR 信号入力中
14	SLTC	'1'= LTC 信号入力中
15	SDIN	'1'= DLS 信号入力中
16	SINP	'1'= INPOS 信号入力中
17	未定義	常に'0'
19,18	PFC1,0	RCMP5 用プリレジスタの使用状態 00:未確定, 01:レジスタ確定, 10:1 st プリレジスタ確定, 11:2 nd プリレジスタ確定(プリレジスタフル)
21,20	PFM1,0	動作用プリレジスタ(RCMP5 用以外)の使用状態 00:未確定, 01:レジスタ確定, 10:1 st プリレジスタ確定, 11:2 nd プリレジスタ確定(プリレジスタフル)
31-22	未定義	常に'0'

表 4.2-6 拡張ステータス(RSTS)

4.3 動作モードレジスタ(RMD)

4.3.1 動作モード(MOD)

ある動作をするためには動作モードを設定します。

No.	選択事項	設定箇所	使用する関数	備 考
1	動作モード(MOD)	RMD.b7-0	WritOpeMode / wReg	ライブラリ関数 WritOpeMode では MOD(RMD.b7-0)のみ設定します。

No.	動作モード分類	MOD	動作モード
1	連続送り	00h	＋方向連続送り
		08h	－方向連続送り
2	位置決め	41h	位置決め動作(移動量指定)
		42h	PCS 位置決め動作(ライブラリ関数固有)
		42h	位置決め動作(指令位置指定) ドライバ関数でのみ設定可能
		44h	指令位置 0 点復帰動作
		45h	機械位置 0 点復帰動作
		46h	＋方向 1 パルス動作
		4eh	－方向 1 パルス動作
		47h	タイマ動作
3	補間	60h	直線補間連続送り
		61h	直線補間
		62h	2 個の PCL 間の直線補間連続送り
		63h	2 個の PCL 間の直線補間
		64h	CW 方向円弧補間
		65h	CCW 方向円弧補間
4	原点復帰	10h	＋方向原点復帰動作
		18h	－方向原点復帰動作
		12h	＋方向原点拔出し
		1ah	－方向原点拔出し
		15h	＋方向原点サーチ
		1dh	－方向原点サーチ
5	ELS, SLS 動作	20h	＋ELS または＋SLS 位置まで動作
		28h	－ELS または－SLS 位置まで動作
		22h	＋ELS または＋SLS 拔出し動作
		2ah	－ELS または－SLS 拔出し動作
6	Z 相まで移動	24h	＋方向に Z 相カウント分動作
		2ch	－方向に Z 相カウント分動作
7	ハンドル送り	01h	パルス入力による連続動作
		51h	パルス入力による位置決め動作
		54h	パルス入力による指令位置 0 点復帰動作
		55h	パルス入力による機械位置 0 点復帰動作
8	JOG 送り	02h	±DR 入力による連続動作
		56h	±DR 入力による位置決め動作

表 4.3-1 動作モード(MOD)一覧

4.3.2 動作モードレジスタ内容

bit	名 称	説 明
7-0	MOD	動作モード(前ページ参照)
8	DLS 有効	0:DLS 無効, 1:DLS 有効
9	INPOS 有効	0:INPOS 無効, 1:INPOS 有効
10	加減速方式	0:直線加減速, 1:S 字加減速
11	CTR1 カウント禁止	1:指令位置カウンタ(CTR1) カウント禁止
12	動作完了タイミング	0:完了タイミングをパルスの周期完了とします.
13	減速開始点計算方法	0:減速開始点自動計算, 1:減速開始点手動計算
14	PCS 有効	0:PCS 無効, 1:PCS 有効
15	補間合成速度	0:補間時合成速度一定制御 OFF, 1:補間時合成速度一定制御 ON
17,16	SEQ No	MSTS.b7,6 に現在実行中の SEQ No が反映されます.
19-18	スタート条件	00: スタートコマンド書込み後即スタート. 01: STA 入力によるスタート. 10: 他軸の条件一致によるスタート. 11: 指定軸の停止によるスタート.
23-20	軸停止によるスタート時の軸指定	指定軸の停止によるスタート(RMD.b19,18=11)の場合, 軸を指定します. bit20: X(V,X1,X2,X3,X4)軸の停止 bit21: Y(W,Y1,Y2,Y3,Y4)軸の停止 bit22: Z(A,Z1,Z2,Z3,Z4)軸の停止 bit23: U(B,U1,U2,U3,U4)軸の停止 <例> 0001: X(V,X1,X2,X3,X4)軸の停止で次動作スタート 1001: X(V,X1,X2,X3,X4)軸の停止かつ U(B,U1,U2,U3,U4)軸の停止で次動作スタート
24	STP 有効	1:STP 入力により停止します.
25	異常時 STP 自動出力	1:異常停止時に STP(同時停止信号)を自動出力します.
26	FH 補正	0:ON, 1:OFF
27	円弧自動終点引込み	1:円弧自動終点引込み
28	予約	
29	MSDC	1:減速開始点自動計算時, 減速開始点=加速パルス数とする. 0:補間動作を合成速度一定制御 ON 時のみ減速開始点=加速パルス数とする. (位置決め動作の減速開始点計算は内部自動演算)
31,30	予約	

表 4.3-2 RMD の内容









4.4 環境設定レジスタ

4.4.1 環境設定レジスタ 1(RENV1)

読み出しコマンド:DCh, 書き込みコマンド:9Ch

指令出力, マシンインターフェイス信号, サーボインターフェイス信号など入出力極性, パルス幅, 動作などを設定します.

15				12		11		8		7		4		3		0																	
ERCTL				EPW2		EPW1		EPW0		ERCTO		ERCTE		ALML		ALMM		OLL		SDL		SDLT		SDM		ELM		PMD2		PMD1		PMD0	
31				28		27		24		23		20		19		16																	
0		0		INTM		DTMF		DRF		FLTR		DR		PCS		LTC		INPS		CLR1		CLR0		CSTP		CSTA		ETW1		ETW0			

bit	名 称	説 明
2-0	PMD2-0	信号端子名 100: 個別パルス方式 指令出力
		 CW 0v 0v CW パルス出力
		 CCW 0v 0v CCW パルス出力
		信号端子名 010: 共通パルス方式 指令出力
		 0v 0v パルス列
		 CW 方向 0v 0v CCW 方向 0v 方向出力
		信号端子名 101: 位相差方式(B 相進相) 指令出力
		 CW 方向 0v 0v CCW 方向 0v A 相出力
		 0v 0v B 相出力
		信号端子名 110: 位相差方式(A 相進相) 指令出力
		 CW 方向 0v 0v CCW 方向 0v A 相出力
		 0v 0v B 相出力
3	ELM	ELS 検出時の停止方法 0:即停止, 1:減速停止(減速距離に注意)
4	SDM	DLS 検出時の動作方法 0:減速後定速移動継続, 1:減速停止
5	SDLT	DLS のラッチ機能 0:使用しない, 1:使用する
6	SDL	DLS の入力極性 0:B 接, 1:A 接
7	OLL	OLS の入力極性 0:B 接, 1:A 接
8	SVALMM	SVALM 入力時停止方法 0:即停止, 1:減速停止
9	SVALML	SVALM 信号入力極性 0:B 接, 1:A 接
10	ERCTE	1:異常即停止時のサーボ偏差カウンタクリア信号(SVCTRCL)自動出力する
11	ERCTO	1:原点復帰完了時サーボ偏差カウンタクリア信号(SVCTRCL)自動出力する
14-12	EPW2-EPW0	サーボ偏差カウンタクリア信号幅(SVCTRCL)出力幅 000:12us, 001:0.1ms, 010:0.4ms, 011:1.6ms, 100:13ms, 101:52ms, 110:104ms, 111:レベル
15	SVCTRCL	1:サーボ偏差カウンタクリア信号の出力論理反転
17-16	ETW1-ETW0	サーボ偏差カウンタクリア信号出力後デレイ時間 00:0, 01:12us, 10:1.6ms, 11:104ms

15		12		11		8		7		4		3		0																	
ERCTL		EPW2		EPW1		EPW0		ERCTO		ERCTE		ALML		ALMM		OLL		SDL		SDLT		SDM		ELM		PMD2		PMD1		PMD0	
31		28		27		24		23		20		19		16																	
0		0		INTM		DTMF		DRF		FLTR		DR		PCS		LTC		INPS		CLR1		CLR0		CSTP		CSTA		ETW1		ETW0	

bit	名 称	説 明
18	CSTA	STA 入力仕様 0:レベル, 1:エッジ
19	CSTP	STP 入力時停止方法 0:即停止, 1:減速停止(注意 1)
22	INPOS	インポジション信号入力極性 0:B 接, 1:A 接
23	LTC	LTC 入力極性 0:B 接, 1:A 接 (エッジ検出)
24	PCS	PCS入力信号極性 0:B 接, 1:A 接
25	DR	DR入力信号極性 0:B 接, 1:A 接
28	DTMF	方向変化タイマ ON/OFF 0:ON, 1:OFF
29	INTM	0:この軸からの割込信号出力, 1:この軸からの割込信号出力禁止



注 意

次動作連続実行時に STP 入力時減速停止に設定した場合, STP を外部から入力する場合は停止するまで入力 ON 状態を保持してください。

同時ストップコマンド使用時はプリレジキャンセルコマンドを発行してから同時ストップコマンドを発行してください。

4.4.2 環境設定レジスタ 2(RENV2)

読み出しコマンド: DDh, 書き込みコマンド: 9Dh

エンコーダ A/B 相カウント方法/方向, Z 相極性, パルサ A/B 相カウント方法/方向を設定します。

15				12		11		8				7		4				3		0							
1	1	1	1	x	1	0	1	0	1	x	x	0	1	0	1												
31				28		27		24				23		20				19		16							
POFF		EOFF		SMAX		PMSK		0		PDIR		PULSR		EZL		EDIR		ENCM		0		0		0		0	

bit	名 称	説 明
19-0	ボード仕様 (固定)	汎用入出力ポートの仕様設定. b19~b0 は必ず次のように初期設定する. HPCI-CPD574N, HPCI-CPD578N, HPCle-CPD674N, HPCle-CPD678N, HPCle-NCB674N, HPCle-NCB674N(1), HUSB-CPD434v2, HUSB-CPD434U, HETN-CPD834T, HWIF-CPD834W .. 0x0fd55, その他 .. 0x0f555
21-20	ENCM	エンコーダ逡倍設定 00:1 逡倍, 01:2 逡倍, 10:4 逡倍, 11:up/down パルス
22	EDIR	1:エンコーダ A 相, B 相カウント方向逆転
23	EZL	1:エンコーダ Z 相入力極性反転
25-24	PULSR	パルサ逡倍設定 00:1 逡倍, 01:2 逡倍, 10:4 逡倍, 11:up/down パルス
26	PDIR	1:パルサ A 相, B 相カウント方向逆転
28	PMSK	1:指令パルス出力禁止(CTR1 は動作します)
29	SMAX	1:自軸を含めた指定軸停止によるスタート有効(RMD.b23-20)
30	EOFF	1:エンコーダ入力をマスク
31	POFF	1:パルサ入力をマスク

4.4.3 環境設定レジスタ 3(RENV3)

読み出しコマンド: DEh, 書き込みコマンド: 9Eh

原点復帰方法指定, 原点復帰 Z 相カウント設定, CTR 動作方法の設定などを行います.

15				12				11				8				7				4				3				0			
0				CTR4G				CTR4IP				CTR3IP				CTR2IP				Z 相CT				ORGmode							
31				28				27				24				23				20				19				16			
CTR4 DIS		CTR3 DIS		CTR2 DIS		0		CTR4 BS		CTR3 BS		CTR2 BS		CTR1 BS		CTR4 ORG		CTR3 ORG		CTR2 ORG		CTR1 ORG		0		0		0		0	

bit	名 称	説 明
3-0	ORG mode	0000:OLSon で完了 0001:OLSon 停止後 RFA 反転 OLSoff まで動作後 RFA 再反転 OLSon で完了 0010:OLSon 減速+Z 相カウントで完了 0011:OLSon 後の Z 相カウントで完了 0100:OLSon 停止後 RFA 反転+ Z 相カウントで完了 0101:OLSon 停止後加速反転+ Z 相カウントで完了 0110:ELSon 停止後 RFA 反転 ELSoff で完了 0111:ELSon 停止後 RFA 反転+ Z 相カウントで完了 1000:ELSon 停止後加速反転+ Z 相カウントで完了 1001:ORGmode0 の動作後 CTR2 零点復帰 1010:ORGmode3 の動作後 CTR2 零点復帰 1011:ORGmode5 の動作後 CTR2 零点復帰 1100:ORGmode8 の動作後 CTR2 零点復帰
7-4	Z 相 CT	原点復帰時のエンコーダZ相カウント値:0000(1 回)~1111(16 回)
9-8	CTR2IP	CTR2 入力元選択 00:ENC 入力, 01:指令パルス, 10:パルス入力
11-10	CTR3IP	CTR3 入力元選択 00:指令パルスと ENC 入力の偏差, 01:指令パルスとパルス入力の偏差
13-12	CTR4IP	CTR4 入力元選択 00:指令パルス, 01:ENC 入力, 10:パルス入力, 11: 9.8304MHz CLK カウント
14	CTR4G	1:動作中のみ CTR4 カウント
23-20	CTRxORG	1:原点復帰完了時 CTRx クリア
27-24	CTRxBs	1:バックラッシュ動作, スリップ動作中も CTRx カウント
31-29	CTRxDIS	1:CTRx カウントしない

4.4.4 環境設定レジスタ 4(RENV4)

読み出しコマンド: DFh, 書き込みコマンド: 9Fh

コンパレータ(CMP)は 5 組あります. CMP1~4 を設定します. (CMP5 は RENV5)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C2RM	CMP2M	CMP2D				CMP2C	C1RM	CMP1M	CMP1D				CMP1C		
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMP4M			CMP4D				CMP4C	0	CMP3M	CMP3D				CMP3C	

bit	名 称	説 明
1-0	CMP1C	CMP1 比較カウンタ 00:CTR1, 01:CTR2, 10:CTR3, 11:CTR4
4-2	CMP1D	CMP1 条件 001:RCMP1=比較 CTR(UP/DOWN), 010: RCMP1=比較 CTR(UP 時) 011:RCMP1=比較 CTR(DOWN 時), 100: RCMP1>比較 CTR 101: RCMP1<比較 CTR, 110:+SLS(RCMP1<CTR1), その他:比較せず
6-5	CMP1M	CMP1 条件成立時の処理 00:処理なし, 01:即停止, 10:減速停止, 11:動作プリレジスタのデータに変更
7	C1RM	1:RCMP1 と対で CTR1 を回転軸カウンタとして使用
9-8	CMP2C	CMP2 比較カウンタ 00:CTR1, 01:CTR2, 10:CTR3, 11:CTR4
12-10	CMP2D	CMP2 条件 001:RCMP2=比較 CTR(UP/DOWN), 010: RCMP2=比較 CTR(UP 時) 011:RCMP2=比較 CTR(DOWN 時), 100: RCMP2>比較 CTR 101: RCMP2<比較 CTR, 110:-SLS(RCMP2>CTR1), その他:比較せず
14-13	CMP2M	CMP2 条件成立時の処理 00:処理なし, 01:即停止, 10:減速停止, 11:動作プリレジスタのデータに変更
15	C2RM	1: RCMP2 と対で CTR2 を回転軸カウンタとして使用
17-16	CMP3C	CMP3 比較カウンタ 00:CTR1, 01:CTR2, 10:CTR3, 11:CTR4
20-18	CMP3D	CMP3 条件 001:RCMP3=比較 CTR(UP/DOWN), 010: RCMP3=比較 CTR(UP 時) 011:RCMP3=比較 CTR(DOWN 時), 100: RCMP3>比較 CTR 101: RCMP3<比較 CTR, 110:設定禁止, その他:比較せず
22-21	CMP3M	CMP3 条件成立時の処理 00:処理なし, 01:即停止, 10:減速停止, 11:動作プリレジスタのデータに変更
25-24	CMP4C	CMP4 比較カウンタ 00:CTR1, 01:CTR2, 10:CTR3, 11:CTR4
29-26	CMP4D	CMP4 条件 0001:RCMP4=比較 CTR(UP/DOWN) 1000:定ピッチ信号出力用として使用(UP/DOWN) 0010:RCMP4=比較 CTR(UP 時) 1001:定ピッチ信号出力用として使用(UP 時) 0011:RCMP4=比較 CTR(DOWN 時) 1010:定ピッチ信号出力用として使用(DOWN 時) 0100: RCMP4>比較 CTR その他:比較せず 0101: RCMP4<比較 CTR
31-30	CMP4M	CMP4 条件成立時の処理 00:処理なし, 01:即停止, 10:減速停止, 11:動作プリレジスタのデータに変更



注 意

比較対象カウンタとして CTR3 を選択した場合はカウント値の絶対値(0~32,767)と比較データとの比較となります。
+SLS, -SLS に設定する場合は比較対象カウンタに CTR1 を選択します。

4.4.5 環境設定レジスタ 5(RENv5)

読み出しコマンド: E0h, 書き込みコマンド: A0h

CMP5 の設定、アイドリングパルス数、カウンタのラッチ条件、他軸スタート条件などの設定を行います。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0		CTLCHF		CTLCH		PDSM		IDL P		CMP5M		CMP5D		CMP5C	
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0		0		0		0		CU4L CU3L CU2L CU1L		0		0		TRIGRINsel	
										TRIGROTsel					

bit	名 称	説 明
2-0	CMP5C	CMP5 比較対象 000:CTR1, 001:CTR2, 010:CTR3, 011:CTR4, 100:PCTR, 101:現在速度
5-3	CMP5D	CMP5 条件 001:RCMP5=比較 CTR(UP/DOWN), 010: RCMP5=比較 CTR(UP 時) 011:RCMP5=比較 CTR(DOWN 時), 100: RCMP5>比較 CTR 101: RCMP5<比較 CTR, その他:比較せず
7-6	CMP5M	CMP5 条件成立時の処理 00:処理なし, 01:即停止, 10:減速停止, 11:動作プリレジスタのデータに変更
10-8	IDLP	アイドルングパルス数(0～7)
11	PDSM	1: PA/PB 及び±DR による連続動作時, 動作方向の ELS によるエラー割り込みが発生します。 再スタートにはスタートコマンドが必要です。(CPDxxxN のみ)
13-12	CTLCH	CTR ラッチ条件 00:LTCoff→on, 01:OLSoFF→on, 10:CMP4 条件成立時, 11:CMP5 条件成立時
14	CTLCHF	1:CTR3 の代わりに速度レジスタをラッチ
19-16	TRIGROT sel	他軸をスタートさせるトリガ条件設定 0001:CMP1 条件成立時 1000:加速開始時 0010:CMP2 条件成立時 1001:加速終了時 0011:CMP3 条件成立時 1010:減速開始時 0100:CMP4 条件成立時 1011:減速終了時 0101:CMP5 条件成立時 その他:トリガ OFF
21-20	TRIGRIN sel	自軸がスタートするためのトリガ軸の選択 00:X(V)軸, 01:Y(W)軸, 10:Z(A)軸, 11:U(B)軸
27-24	CUxL	1:CTR _x のラッチ後 CTR _x クリア

4.4.6 環境設定レジスタ 6(RENv6)

読み出しコマンド:E1h, 書き込みコマンド:A1h

バックラッシュ/スリップ補正量、パルサ分周/逡倍比を設定します。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	BLSHsel	BLSH												
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PMG					PD										

bit	名 称	説 明
11-0	BLSH	バックラッシュ/スリップ補正量
13-12	BLSHsel	00:補正 off, 01:バックラッシュ補正, 10:スリップ補正
26-16	PD	パルサ分周比 (設定値) / 2,048 に分周 0:分周 OFF
31-27	PMG	パルサ逡倍比 (設定値+1) 倍に逡倍

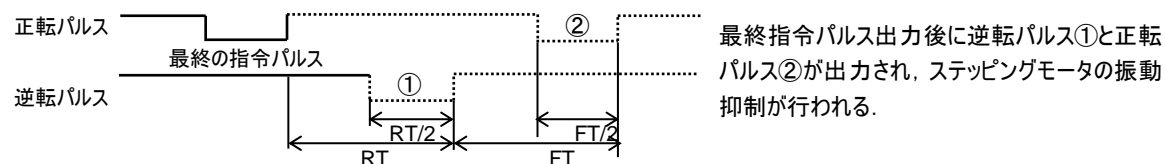
4.4.7 環境設定レジスタ 7(RENV7)

読み出しコマンド: E2h, 書き込みコマンド: A2h

ステッピングモータの停止時特性を改善する「制振機能」を設定します.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FT															

bit	名 称	説 明
15-0	RT	最終出力パルス後の逆転パルス出力タイミング RT×1.6us(1.6us～104ms)
31-16	FT	上記逆転パルス出力後の正転パルス出力タイミング FT×1.6us(1.6us～104ms)



5. 更新履歴

版数	日付	更新内容
5.00 版	2011/07/22	CPD ボードシリーズ運用編新規作成
5.01 版	2011/09/22	HPCI-CPD553 追加
5.02 版	2012/03/06	注意事項を追加
5.03 版	2012/06/15	「2.2.6 サーボ I/F」に HPCI-CPD508, HPCI-CPD5016 使用時の注意事項を追加. 「3.15-2 HPCI-CPD5016 (1) 汎用入力ポートの用途選択設定と設定状態読込」修正.
5.10 版	2013/01/09	HPCle-NCB674N, HPCle-NCB674N(1)追加. 「1.2 用語」レジスタ名表及び「4.1.6 レジスタ制御コマンド」の内容の説明を一部修正. 「2.2.5 速度と加減速」内の計算式レジスタ名の誤記修正. 「2.2.9 原点復帰方式」に原点サーチについての説明を追加. 「2.3.5 動作完了監視」に INPOS 制御についての説明を追加. 「2.5 原点復帰」の中の原点サーチ説明を詳細に記述. 「2.7.1 動作手順」誤記修正. 「3.10 PCS 機能」の PCS 信号入力仕様追記. 「4.PCL 資料」にステータス, 動作モードの記述を追加.
5.20 版	2013/09/06	HPCle-CPD674N 追加 「3.2.6 次動作連続実行例」追加 位置オーバーライド失敗ビット説明追加 「3.8 位置のオーバーライド」の説明を修正
5.21 版	2013/12/26	「3.2.6 次動作連続実行例」関数名修正 「3.14 マスタ・スレーブ機能」のスレーブ軸の速度設定を追記. 「4.4.2 環境設定レジスタ 2(RENV2)」の設定値修正. (HUSB-CPD434v2 のみ関係)
5.22 版	2014/03/17	HPCle-CPD678N 追加 INtime 版ソフトウェア追加によるマニュアル構成の記載変更
5.23 版	2015/09/16	「2.2.1 ライブラリ関数を用いたボードアクセスの準備」の環境設定 2 の設定内容修正 「2.7 タイマ動作」追加 「3.6.2 CMP3 比較結果出力」に出力端子設定追加 「3.6.3 CMP4 比較結果出力」に出力端子設定追加 「3.6.4 CMP5 比較結果出力」に出力端子設定追加 「3.6.5 CMP 端子出力設定」の HUSB-CPD434v2 の誤記修正 「3.6.6 不等ピッチ出力例」追加 「3.13.1 対応機種」に HUSB-CPD434CBv2, HUSB-CPD434MSv2 を追加 「3.14.4 マスタ・スレーブ機能設定ポート」追加 「3.14.5 マスタ追従」注意事項追加. 設定例に環境設定レジスタ1 追加. 「3.14.6 マスタ・スレーブエリア機能」機能設定手順追加. 「3.14.7 並行 2 軸制御機能」 注意事項追加. ステータスの運用, 原点復帰に追記. 通常動作の記述修正 「4.4.4 環境設定レジスタ 4(RENV4)」の誤記修正 「4.4.5 環境設定レジスタ 5(RENV5)」の CTR ラッチ条件追加 LATCH, LTCH, LTC の表記を LTC に統一
5.27 版	2016/01/12	適応ボードの追加 本マニュアルに記載される用語の追加 「1.2.1 レジスタ名」で一部誤字修正 「2.2.7 マシン I/F」に HPCI-CPD553 を追加 「3.5.1 コンパレータの種類と機能」で一部誤記修正 「3.6.2 CMP3 比較結果出力」に出力幅設定を追加 「3.6.3 CMP4 比較結果出力」に出力幅設定を追加 「3.6.4 CMP5 比較結果出力」に出力幅設定を追加 「3.6.5 CMP 端子出力設定」に HPCI-CPD553 を追加、及びボード種別と出力コネクタの対応表を追加 「3.6.6 不等ピッチ出力例」を追加 「3.9.1 コンパレータ条件一致及び減速タイミングによる他軸スタート」のフローチャートを一部見直し 「3.9.2」 CMP 条件成立による他軸スタートの簡単な例」を追加 「3.9.3 コンパレータ条件一致 STA 出力による同期」のフローチャートを一部見直し 「3.9.4 コンパレータ条件成立での STA 出力による同時スタート設定例」追加、及び HPCI-CPD553 を追加

		「3.9.6 軸停止トリガによるスタートの例」を追加 「3.9.7 複数 PCL 間の直線補間」に注を追加 「3.9.8 PCL 間の直線補間の例」を追加 「3.10.3 PCS コード例」を追加 「3.11.4 パルス入力による動作のコード例」を追加 「3.13.1 対応機種」にボード追加 「3.13.2 機能」に一部コメント追記 「3.13.3 設定箇所」に一部コメント追記 「3.13.4 ステータス」に一部コメント追記 「3.13.5 ラッチコマンド」追加 「3.13.6 Set/Read DLS/PCS 入力選択(DLS/PCS)」追加 「3.13.7 LTC 入力コード例」 「3.14.1 対応機種」に HPCI-CPD553 を追加 「3.14.5 マスタ追従」(2)接続構成 の図 3.14-2 マスタ追従接続構成に図を追加 「3.14.5 マスタ追従」(4)設定例 の一覧表に RENV3 他を追加 「3.14.5 マスタ追従」(8)同期精度について を追加 「3.14.5 マスタ追従」(9)コード例 を追加 「3.14.6 マスタ・スレーブエリア機能」にモードと仕様の情報を追加 「3.14.6 マスタ・スレーブエリア機能」(5)に マスタ・スレーブエリア機能のコード例を追加 「3.14.7 並行 2 軸制御機能」(3)初期設定例 に RENV3 及びコメントを追加 「3.14.7 並行 2 軸制御機能」(6)に 並行 2 軸原点復帰コード例を追加 「3.14.7 並行 2 軸制御機能」(8)に 並行 2 軸位置決め動作コード例を追加 「3.14.7 並行 2 軸制御機能」(9)に 偏差エラーを追加 「3.14.7 並行 2 軸制御機能」(10)に 偏差エラー設定を追加 「3.14.7 並行 2 軸制御機能」(11)に 偏差エラー設定コード例を追加 「3.14.8 スレーブ拡張モード」(2)初期設定例 に J3_SEL を追加 「3.14.8 スレーブ拡張モード」(6)に スレーブ拡張モードコード例を追加
5.30 版	2016/10/07	「2.2.5 速度と加速度 (2) 加減速」の誤記修正 「4.4.1 環境設定レジスタ 1(RENV1)」に bit23 LTC 入力極性追加
5.40 版	2017/2/15	HUSB-CPD434U/ HETN-CPD434T/ HWIF-CPD834W 製品ラインナップ追加により 関連事項追記
5.41 版	2018/3/02	HPCIe-CPD632 製品ラインナップ追加により関連事項追記及び誤記修正