

PCL61x4 シリーズ から PCL61x5 シリーズ への移行について

日本パルスモーター株式会社

拝啓 時下益々ご清栄のこととお慶び申し上げます。
 平素は格別のご高配に預かり厚くお礼申し上げます。
 さて、PCL61x4 シリーズ (PCL6114/6124/6144) と PCL61x5 シリーズ (PCL6115/6125/6145) の相違点と、
 PCL61x5 シリーズへ移行される場合にご確認いただきたい事項を下記の通りご案内いたします。

つきましては、誠にお手数をお掛けしますが、内容のご確認のほど、よろしくお願い申し上げます。

敬具

記

1. 概要

PCL61x5 シリーズは、PCL61x4 シリーズを機能アップした製品ですが、完全な上位互換製品ではなく、一部の機能が異なりますのでご注意ください。

ソフト的には、ほとんどの場合は、PCL61x4 用に開発したソフトにより PCL61x5 を制御することができます。

ハード的には、PCL6115/6125/6145 はパッケージと端子配置が PCL6114/6124/6144 と同一ですが、PCL61x5 を PCL61x4 用ボードに実装される場合には、「2. 移行時の注意点」のご確認をお願いいたします。

2. 移行時の注意点

移行検討時には、下記の相違点が問題にならないことをご確認ください。

ご使用される制御システムで、相違点が問題になる場合にはソフト・ハードの変更が必要になります。

2-1. 消費電流の増加

PCL61x4 とは消費電流が異なります。

機種	条件	PCL61x4	PCL61x5
PCL6114, PCL6115	1 軸 15Mpps 動作、無負荷	28 mA Max	37 mA Max
PCL6124, PCL6125	2 軸 15Mpps 動作、無負荷	65 mA Max	75 mA Max
PCL6144, PCL6145	4 軸 15Mpps 動作、無負荷	129 mA Max	139 mA Max

2-2. 原点復帰時の ORG サンプリングタイミングの変更

PCL61x5 では、PCL61x4 使用時と、原点位置が 1 パルス分だけずれる場合があります。

PCL61x4 では ORG 入力フィルタとして時間フィルタと距離フィルタの併用を行っていました。

距離フィルタでは出力パルスの ON タイミングで ORG 入力をサンプリングして、1 パルス区間以上の間、ORG センサが ON し続けていないとノイズとして無効としていました。

しかし、ON 距離範囲が短い近接センサでは使いにくい場合があるため、PCL61x5 では距離フィルタを廃止し、時間フィルタだけにしました。

2-3. PCS 入力による位置管理制御方法の変更

RENV1.PCSM=1 にすると、PCS 入力端子を自軸だけの CSTA 入力として使用できました。

PCL61x4 では、PCSM=1 でも本来の PCS 機能が動作しており、自軸だけの CSTA との認識と同時に PCS 入力としても認識して目標位置オーバーライド 2 の開始信号としても動作していました。

PCL61x5 では、PCSM=1 の場合には目標位置オーバーライドとしては認識しないようにしました。

2-4. ERC 出力条件の変更

モータドライバ内の偏差カウンタのクリア信号(ERC)は、原点復帰動作時、エラー停止時、ERC 出力コマンドにより出力されますが、エラー停止時の出力条件を変更しました。

PCL61x4 では、RENV1.EROE=1 で、停止方法が即停止の時にだけ出力されていました。

PCL61x5 では、RENV1.EROE=1 で、停止時の速度が FL 速度より高かった場合だけ出力され、停止方法には影響されません。

2-5. 次動作スタートコマンドの過書き込み

プリレジスタは 1 段であるため、動作中に次動作のスタートコマンドを書き込むことはできますが、次々動作のスタートコマンドは書き込めません。

PCL61x4 ではプリレジスタが確定状態(次動作スタートコマンド書き込み済み)の時に、さらにスタートコマンドを書き込むと、次動作スタートコマンドが上書きされていました。

PCL61x5 では、プリレジスタ確定状態でのスタートコマンド書き込みは無視します。

2-6. CMEMG コマンドの実行軸の変更

CEMG 端子から非常停止信号を入力すると全軸が即停止します。

CEMG 入力と同等なコマンドとして CMEMG コマンドがあります。

PCL61x4 では、CMEMG コマンドを書き込んだ軸しか停止しないので、CMEMG コマンド書き込み時に COMW(11~8)で全軸有効に指定する必要性がありました。また、停止中の CEMG 入力でも発生するエラー割り込み(REST. ESEM)が、停止中の CMEMG コマンドでは発生しませんでした。

PCL61x5 では、COMW(11~8)で全軸を指定しなくても、どれかの軸に CMEMG コマンドを書き込むと全軸が即停止します。また、停止中に書き込んでもエラー割り込み(REST. ESEM)が発生します。

2-7. PCS 端子状態のモニタ機能の変更

PCS 入力端子の状態は、RSTS.SPCS ビットで確認できます。

また、RENV1.PCSM=1 にすると、PCS 端子は自軸だけの CSTA 信号として動作します。

PCL61x4 では、RENV1.PCSM=1 の場合に、PCS 端子状態で RSTS.SSTA ビットが変化し、RSTS.SPCS ビットは変化しませんでした。

PCL61x5 では RENV1.PCSM 設定に関係なく、RSTS.SPCS ビットは PCS 端子状態の確認、RSTS.SSTA ビットは CSTA 端子状態の確認ができます。

機種	RENV1.PCSM=0		RENV1.PCSM=1	
	RSTS.SPCS	RSTS.SSTA	RSTS.SPCS	RSTS.SSTA
PCL61x4	PCS 端子モニタ	CSTA 端子モニタ	常に 0	PCS 端子モニタ
PCL61x5	PCS 端子モニタ	CSTA 端子モニタ	PCS 端子モニタ	CSTA 端子モニタ

2-8. 停止中の RPLS モニタ値

RPLS レジスタは、位置決め動作中に残パルス数を確認したい時に読み出します。

PCL61x4 では、停止中の読出しを想定していなかったため、停止中に RMV レジスタまたは RMD レジスタ書き込みを行うと、残パルス数以外の値に変化していました。

PCL61x5 では、停止中に RMV, RMD レジスタを変更しても、前回停止時の RPLS の値を保持します。

2-9. メインステータスの定義追加

ソフトリミット機能の追加により、未定義（読出し値は 0）ビット 10, 11 が SCP3, SCP4 となりました。
PCL61x4 では、初期状態（RENV3. M614=0）では“00”、M614=1 の時はソフトリミット状態に変化します。
PCL61x5 では、RENV3. M614 設定は無効になり、常にソフトリミット状態に変化します。

2-10. RSDC レジスタモニタ値の上位ビットの定義変更

RSDC はスローダウンポイント値の読出し専用レジスタです。

スローダウンポイントの範囲としては、0～16, 777, 215, の 24 ビットのため、PCL61x4 では未使用のビット 31～24 を全て 0 固定にしていました。

スローダウンポイント手動設定時（RMD. MSDP=1）には、RDP 設定値と等しくなります。

スローダウンポイント自動設定時（RMD. MSDP=0）には、RDP 設定値であるオフセット値（-8, 388, 608～+8, 388, 607）がスローダウンポイント値の初期値になり、スタート後に出力パルスに同期してカウントアップしてゆき、最大で +16, 777, 215 まで増加する可能性があります。

しかし、RSDC 読出し値として、-8, 388, 608 (00800000h) ～ -1 (00FFFFFFh) の値と、+8388, 608 (00800000h) ～ 16, 777, 215 (00FFFFFFh) との区別が出来ませんでした。なお、回路内部では符号付 25 ビット数値として制御して扱っているので問題ありません。

そこで、PCL61x5 では RSDC の上位ビットを 0 固定から符号拡張に変更して、符号付 32 ビットデータとして読み出せるように変更しました。数値範囲は、-8, 388, 608 (FF800000h) ～ +16, 777, 215 (00FFFFFFh) です。

2-11. PA/PB, +DR/-DR 動作の停止割り込み

RENV2. IEND=1 にすると、正常停止/異常停止に関係割り込み（MSTSW. SENI）が発生する仕様でした。

しかし、PCL61x4 では下記の 3 つの動作モードの時には、コマンド停止時とエラー停止時に INT 信号が出力されませんでした。

- ①パルサ（PA/PB）入力による連続動作（RMD. MOD=01h）
- ②外部信号（+DR/-DR）入力による連続動作（RMD. MOD=02h）
- ③外部信号（+DR/-DR）入力による位置決め動作（RMD. MOD=56h）

PCL61x5 では、上記のモードの時にも INT 信号が出力されます。

2-12. 停止中の停止割り込み

PCL61x4 では、RENV2. IEND=1 の設定で停止中でも STOP コマンドまたは DSTP コマンドを書き込むと停止割り込みが発生していました。

PCL61x5 では停止中には停止割り込みが発生しないようにしました。

3. 機能アップ内容

3-1. 機種確認用に ID モニタ機能を追加

PCL61x4 では、機種 (LSI 名) を確認する情報は有りませんでした。

PCL61x5 では、機種を判断できる ID コードの確認機能を追加しました。

ID モニタコマンド (IDMON:03h) に続けて RMG レジスタ読出しコマンド (RRMG:D5h) を書き込むと、読出しデータのビット 31~16 が ID コードになります。

ID コードは、PCL6115 が 03E0h、PCL6125 が 03F0h、PCL6145 が 0400h です。

なお、PCL61x4 では、読出しデータのビット 31~16 は 3 機種とも 0000h になります。

3-2. CPU 接続用シリアル I/F の追加

PCL61x4 では SPI-I/F を追加しましたが、書き込みを無視する場合があります、使用禁止にしました。

PCL61x5 では SPI-I/F 回路のバグを修正し、SPI-I/F (4 線式シリアル I/F) も使用できます。

なお、シリアル I/F 用通信クロック (SCK) 周波数は、基準クロック周波数の 1/1.5 以下までです。

【注意】

既存の平行 I/F とシリアル I/F との切り替えは、 \overline{RD} 、 \overline{WR} 入力端子で行います。

リセット解除時に共にローレベルの時にシリアル I/F、その他の時は平行 I/F になります。

CPU の機種によっては、 \overline{RD} 、 \overline{WR} 出力端子の初期状態がフローティング (汎用入力ポート) になっているものがありますが、PCL61x5 の \overline{CS} 、 \overline{RD} 、 \overline{WR} 入力端子はプルアップ抵抗を内蔵していますのでハイレベルとして認識します。

3-3. シリアル I/F 時のみに使用できる共用ポートの追加

PCL61x5 でシリアル I/F で使用する時には、データバス (D0~D15) を使用しません。

そこで、16 ビットの汎用ポートとして使用できるようにしました。軸ごとに割り付けられた汎用ポートではないため、多軸製品 (PCL6125、PCL6145) の場合は各軸が共通で使用します。このためこの汎用ポートは「共用ポート」と表記し、軸ごとに割り当てられている汎用ポートと区別します。

なお、入出力はビットごとに指定できます。

3-4. MSTSW.SENI、MSTSW.SEOR ビットのリセットについて

メインステータスの SENI、SEOR ビットのリセット方法は、RENV2.MRST の設定によって、読出し処理による自動リセットと、コマンドによるリセットのどちらかを選択ができます。

しかしシリアルバス I/F の場合、RENV2.MRST ビットは 1 に固定されるため、自動リセット機能は使用できません。

3-5. REST、RIST レジスタのリセットについて

エラー割り込みステータス (REST レジスタ) のリセット方法は、RENV2.MRST の設定によって、REST レジスタの読出し処理による自動リセットと、REST レジスタへの書込み処理によって指定したビットだけをリセットする手順のどちらかを選択ができます。

しかしシリアルバス I/F の場合、RENV2.MRST ビットは 1 に固定されるため、自動リセット機能は使用できません。

3-6. EZ 入力を使用した原点復帰での割り込みの追加

RENV2.ORM=1 にすると、ORG 入力と EZ 入力を使用した原点復帰仕様になり、高速スタート時には ORG=ON で減速を開始し、指定回数の EZ 入力により即停止しますが停止時の速度は確認していません。

そのため、必ず減速が完了するような EZ 回数を指定する必要があります。

PCL61x4 では、EZ 設定回数が少なく減速途中で停止しても検出できませんでした。

PCL61x5 では、RENV2.ORM=1 の高速原点復帰で、減速しきれずに停止した時にイベント割り込みを発生させる機能を追加しました。

3-7. スタート割り込みの追加

スタートコマンドを書き込んでから実際にスタートするまでには、方向変化タイマと ERC 制御タイマの完了を待ってからスタート処理を行うための遅延時間があります。

PCL61x4 で実際の動作開始タイミングを知るには、動作状態モニタ (RSTS. CND3~0) の確認を繰り返す必要がありました。

PCL61x5 には、実際の動作開始時にイベント割り込みを発生させる機能を追加しました。

3-8. 現在位置カウンタのラッチ回路を追加

現在位置カウンタのラッチ回路を 2 回路追加しました。(RLTC3, RLTC4)

4. 追加コマンド

4-1. ID モニタコマンド

COMBO	記号	内 容
03h	IDMON	RRMG コマンドで ID コードを読み出せる状態にします

本コマンド書き込み直後に RMG レジスタを読み出すと、ビット(31~16)で機種 ID コードが確認できます。(PCL61x4 では ID コードは 0000h)

機種	機種 ID コード
PCL6115	03E0h
PCL6125	03F0h
PCL6145	0400h

4-2. レジスタ制御コマンド

内 容	Bit 長	名称	読出しコマンド		書き込みコマンド	
			COMBO	記号	COMBO	記号
環境設定 4	32	RENV4	DFh	RRENV4	9Fh	WRENV4
ラッチデータ 3	32	RLTC3	EFh	RRLTC3		
ラッチデータ 4	32	RLTC4	F0h	RRLTC4		
共用ポート (GP0~15) 仕様設定	16	RGPM	FAh	RRGPM	BAh	WRGPM
共用ポート (GP0~15) データ	16	RGPD	FBh	RRGPD	BBh	WRGPD

4-3. ラッチ 3、4 制御コマンド

COMBO	記号	内 容
3Ch	LTC3E	RLTC3 のラッチ動作を有効にします
3Dh	LTC4E	RLTC4 のラッチ動作を有効にします
3Eh	LTC3D	RLTC3 のラッチ動作を無効にします
3Fh	LTC4D	RLTC4 のラッチ動作を無効にします

5. 追加・変更レジスタ

機能アップのために、PCL61x4 では未定義でしたビット(網掛け部分)の定義を追加しています。

PCL61x4 のマニュアルには '0' 固定と記載してありますので、PCL61x4 用ソフトでも問題は発生しないはずですが、動作がおかしい時は '0' になっているかご確認ください。

5-1. RMG レジスタ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	RMG データ											
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDDT (ID モニタ値)															

ビット	ビット名	内容
31-16	IDDT	ID モニタコマンド (IDMON:03h) の直後に RRMG コマンド (D5h) を書き込んだ時だけ、ID コードが読み出せます。その他の時の読出し値は 0000h です。

5-2. RENV2 レジスタ (シリアル I/F 時の機能変更)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POFF	EOFF	GSP0	P7M	P6M	P5M	P4M1	P4M0	P3M1	P3M0	P2M1	P2M0	P1M1	P1M0	POM1	POM0
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MRST	IEND	ORM	EZL	EZD3	EZD2	EZD1	EZD0	PD1R	PINF	P1M1	P1M0	ED1R	E1NF	E1M1	E1M0

ビット	ビット名	内容
31	MRST	MSTSW, REST, R1ST の読み出し直後自動リセット機能の制御 0 : 自動リセット有効 1 : 自動リセット無効 シリアルバス I/F の場合は、"1" に固定されます。

5-2. RENV3 レジスタ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C2S1	C2S0	C1S1	C1S0	C2RM	CU2R	LOF2	CU2L	C1RM	CU1R	LOF1	CU1L	CU2H	CU1H	CIS2	CIS1
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	0	0	SLCU	SLM1	SLM0	SYI1	SYI0	SYO3	SYO2	SYO1	SYO0

ビット	ビット名	内容
25	未定義	PCL61x4 では M614 設定ビットで、M614=0 の時は PCL61x3 互換モードでした。PCL61x5 では常に M614=1 と同じ動作になり、M614 ビットを削除しました。

5-3. RENV4 レジスタ (レジスタの追加)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
L4F1	L4F0	L4MD	L4DT	L4TL	L4T2	L4T1	L4T0	L3F1	L3F0	L3MD	L3DT	L3TL	L3T2	L3T1	L3T0
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	0	0

ビット	ビット名	内容
2~0	L3T2~L3T0	RLTC3 にラッチするトリガ信号の選択 000:無効 001:LTC 入力 010:ORG 入力 011:EZ 入力 100:P4 入力 101:P5 入力 110:P6 入力 111:P7 入力
3	L3TL	RLTC3 用トリガ信号の有効エッジ選択 注 0:立下りエッジ 1:立上りエッジ
4	L3DT	RLTC3 にラッチするカウンタを選択します 0:COUNTER1 (RCUN1) 1:COUNTER2 (RCUN2)
5	L3MD	RLTC3 用ラッチ動作モード選択 0: LTC3E コマンドから LTC3D コマンドの間に 1 回だけラッチ 1: LTC3E コマンドから LTC3D コマンドの間に毎回ラッチ
7~6	L3F1~L3F0	RLTC3 用トリガ入力 (LTC, ORG, EZ, P4~P7) のフィルタ選択 00: フィルタ無 (入力パルス幅 > CLK 周期) 01: 3.2us 以下のパルス幅の入力は無視します 10: 25us " 11: 200us " 注. RENV1. FLTR、RENV2. EINF の設定とは関係ありません。
ビット	ビット名	内容
10~8	L4T2~L4T0	RLTC4 にラッチするトリガ信号の選択 000:無効 001:LTC 入力 010:ORG 入力 011:EZ 入力 100:P4 入力 101:P5 入力 110:P6 入力 111:P7 入力
11	L4TL	RLTC4 用トリガ信号の有効エッジ選択 注 0:立下りエッジ 1:立上りエッジ
12	L4DT	RLTC4 にラッチするカウンタを選択します 0:COUNTER1 (RCUN1) 1:COUNTER2 (RCUN2)
13	L4MD	RLTC4 用ラッチ動作モード選択 0: LTC4E コマンドから LTC4D コマンドの間に 1 回だけラッチ 1: LTC4E コマンドから LTC4D コマンドの間に毎回ラッチ
15~14	L4F1~L4F0	RLTC4 用トリガ入力 (LTC, ORG, EZ, P4~P7) のフィルタ選択 00: フィルタ無 (入力パルス幅 > CLK 周期) 01: 3.2us 以下のパルス幅の入力は無視します 10: 25us " 11: 200us " 注. RENV1. FLTR、RENV2. EINF の設定とは関係ありません。
28~31	未定義	(常に 0 を設定)

注: トリガー信号の入力仕様を変更した後にイネーブルコマンド (LTC3E、LTC4E) を書き込む場合は、必ずL3F (もしくはL4F) で設定されているフィルタ時間の経過を待った後としてください。経過前にイネーブルコマンドを書き込むと余分なラッチ動作が発生します。
待ち時間を短縮するには、フィルタ設定を"00"にした状態でトリガー信号の入力仕様を変更し、再度フィルタ値だけを変更してください (フィルタ設定が"00"の場合、フィルタ時間よりレジスタへの書き込み時間が長いため、時間待ち処理が不要となります)。

5-4. RIRQ レジスタ (ビット定義の追加)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	IRBY	IREZ	IRSA	IRDR	IRSD	IROL	IRLT	IRC2	IRC1	IRDE	IRDS	IRUE	IRUS	IRNM	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	0	IRL4	IRL3

ビット	ビット名	内容
13	IREZ	RENV2. ORM=1 での高速原点復帰で、減速途中で停止した時
14	IRBY	動作中になった時 (スタート時)
15	未定義	(常に 0 を設定)
16	IRL3	RLTC3 へのラッチ時
17	IRL4	RLTC4 へのラッチ時
31~18	未定義	(常に 0 を設定)

5-5. RLTC3 レジスタ (レジスタの追加)

ラッチ3の記憶データです。(読み出し専用)

31	28	24	20	16	12	8	4	0
[31-bit register]								

数値範囲は、-2, 147, 483, 648~+2, 147, 483, 647です。

5-6. RLTC4 レジスタ (レジスタの追加)

ラッチ4の記憶データです。(読み出し専用)

31	28	24	20	16	12	8	4	0
[31-bit register]								

数値範囲は、-2, 147, 483, 648~+2, 147, 483, 647です。

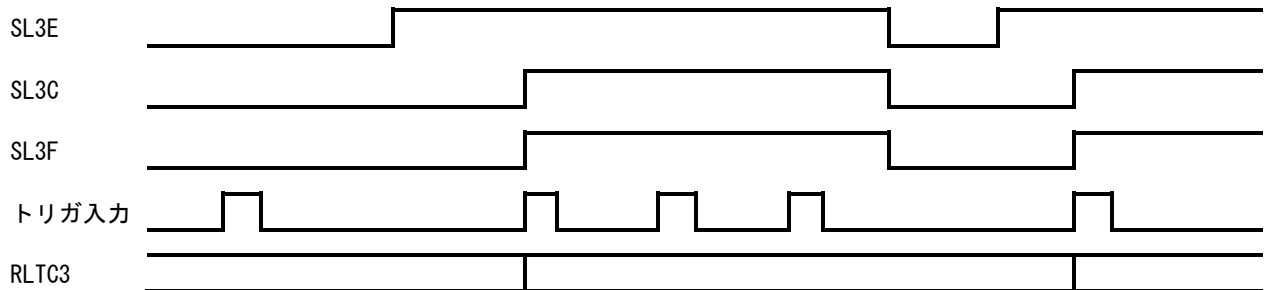
5-7. RSTS レジスタ (ビット定義の追加)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SINP	SDIN	SLTC	SDRM	SDRP	SEZ	SERC	SPCS	SEMG	SSTP	SSTA	SCD	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	SL4F	SL4C	SL4E	SL3F	SL3C	SL3E	SDIR

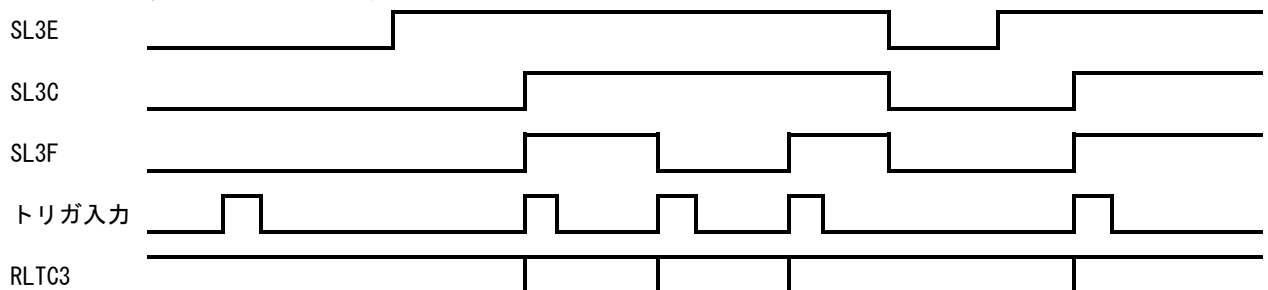
ビット	ビット名	内容
17	SL3E	1: RLTC3ラッチ可能状態 (LTC3Eコマンド書き込み~LTC3Dコマンド書き込み)
18	SL3C	1: RLTC3へ初回ラッチ完了 (LTC3Dコマンドでリセット)
19	SL3F	RLTC3の値変化時にトグル変化 (LTC3Dコマンドでリセット)
20	SL4E	1: RLTC4ラッチ可能状態 (LTC4Eコマンド書き込み~LTC4Dコマンド書き込み)
21	SL4C	1: RLTC4へ初回ラッチ完了 (LTC4Dコマンドでリセット)
22	SL4F	RLTC4の値変化時にトグル変化 (LTC4Dコマンドでリセット)
31~23	未定義	(常に"0"を設定)

ステータス変化タイミング

①RENV4.L3MD=0, RENV4.L3TL=1 の時



②RENV4.L3MD=1, RENV4.L3TL=1 の時



5-8. RIST レジスタ (ビット定義の追加)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISMS	ISPS	ISSA	ISMD	ISPD	ISSD	ISOL	ISLT	ISC2	ISC1	ISDE	ISDS	ISUE	ISUS	ISNM	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	ISL4	ISL3	ISBY	ISEZ

ビット	ビット名	内容
16	ISEZ	RENV2. ORM=1 での高速原点復帰で、減速途中で停止した時
17	ISBY	動作中になった時 (スタート時)
18	ISL3	RLTC3 にラッチした時
19	ISL4	RLTC4 にラッチした時
31~20	未定義	(常に"0"を設定)

5-9. RSDC レジスタ (ビット定義の変更)

31	28	24	20	16	12	8	4	0
#	#	#	#	#	#	#	#	#

スローダウンポイント自動設定 (RMD. MSDP=0) の時には、RDP レジスタにオフセット値として負数も設定でき、その時の RDP 設定範囲は、-8, 388, 608 (800000h) ~ +8, 388, 607 (7FFFFFFh) です。

高速スタート時には、オフセット値からカウントアップしてゆきますが、PCL61x4 ではビット 31~24 が常に 00h であったため、符号付 32 ビット数値としては認識できませんでした。

PCL61x5 では、負数の時にはビット 31~24 が FFh になり、符号付 32 ビット数値として認識できます。なお、RMD. MSDP=1 の時には、0~+16, 777, 215 (00FFFFFFh) の 32 ビット正数として認識できます。

6. シリアルI/F

一般的には、SPIバスと呼ばれている方式です。

シリアルI/Fを使用する場合には、 \overline{RD} 、 \overline{WR} 入力端子を共にローレベル (GND接続) にします。

\overline{CS} 、 \overline{RD} 、 \overline{WR} 端子は、PCL61x5内部でプルアップしています。

シリアルI/F用には下記の端子を使用します。

端子名	方向	シリアル信号名	内容
CS	CPU→PCL	SS	スレーブセレクト信号
IF0	CPU→PCL	SCK	シリアルクロック
IF1	CPU→PCL	MOSI	書き込みデータ
WRQ	CPU←PCL	MISO	読出しデータ
A0	→PCL	DS0	デバイス選択 0
A1	→PCL	DS1	デバイス選択 1

注意. DS0, DS1信号は、1組のシリアル通信線に複数のPCL61x5を接続する時に使用します。

1つのPCL61x5しか接続しない時は、両端子ともGNDに接続します。

6-1. 通信フォーマット

通信フォーマットは軸選択コードのタイプ選択により、下記の4種類があります。

①コマンド書き込みフォーマット

1) 動作コマンド

MOSI :

軸選択コード (S7-S0)	コマンド (C7-C0)
----------------	--------------

1回の通信に対して、軸選択コードとコマンドは1つです。

汎用出力ビット制御コマンドと制御コマンドも、動作コマンドと同様です。

2) レジスタ書き込みコマンド

MOSI :

軸選択コード (S7-S0)	コマンド (C7-C0)	データx	データy	データz	データu
----------------	--------------	------	------	------	------

1回の通信に対して、軸選択コードとコマンドは1つです。

データの数は軸選択コードで選択した軸数分を書き込みます。

データのサイズは32bitで、並び順は “データ [7-0]+データ [15-8]+データ [23-16]+データ [31-24]” です。

3) レジスタ読み出しコマンド

MOSI :

軸選択コード (S7-S0)	コマンド (C7-C0)
----------------	--------------

MISO :

データx	データy	データz	データu
------	------	------	------

1回の通信に対して、軸選択コードとコマンドは1つです。

データの数は軸選択コードで選択した軸数分を読み出します。

データサイズは32bitで、並び順は “データ [7-0]+データ [15-8]+データ [23-16]+データ [31-24]” です。

②メインステータス読み出しフォーマット

MOSI :

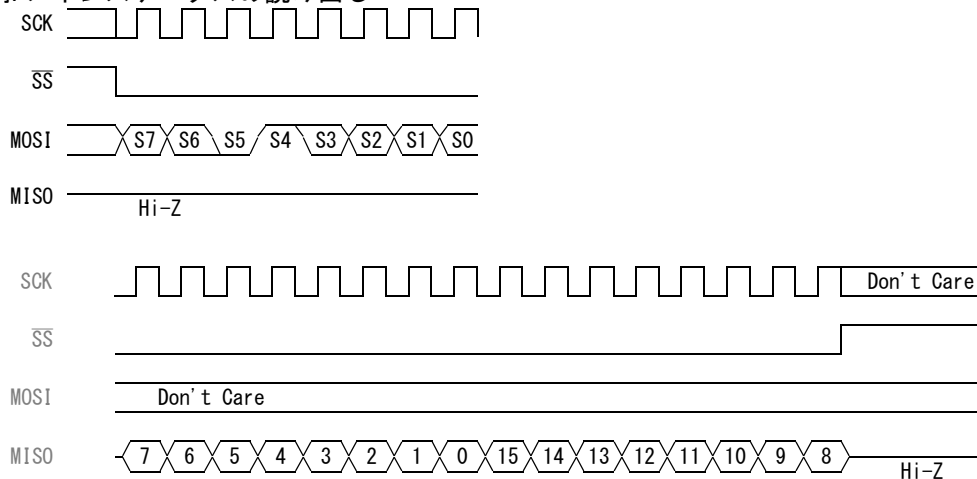
軸選択コード (S7-S0)

MISO :

データx	データy	データz	データu
------	------	------	------

1回の通信に対して、軸選択コードは1つです。
 データの数は軸選択コードで選択した軸数分を読み出します。
 データのサイズは16bitで、並び順は“データ [7-0]+データ [15-8]”です。

例. メインステータスの読み出し



S7-S0 : 軸選択コード(メインステータス読み出しフォーマットは、S5="0", S4="1")
 15-0 : メインステータス読み出しデータ

③汎用ポート書き込みフォーマット

MOSI :

軸選択コード (S7-S0)	データx	データy	データz	データu
----------------	------	------	------	------

1回の通信に対して、軸選択コードは1つです。
 データの数は軸選択コードで選択した軸数分を書き込みます。
 データのサイズは8bitで、並び順は“データ [7-0]”です。

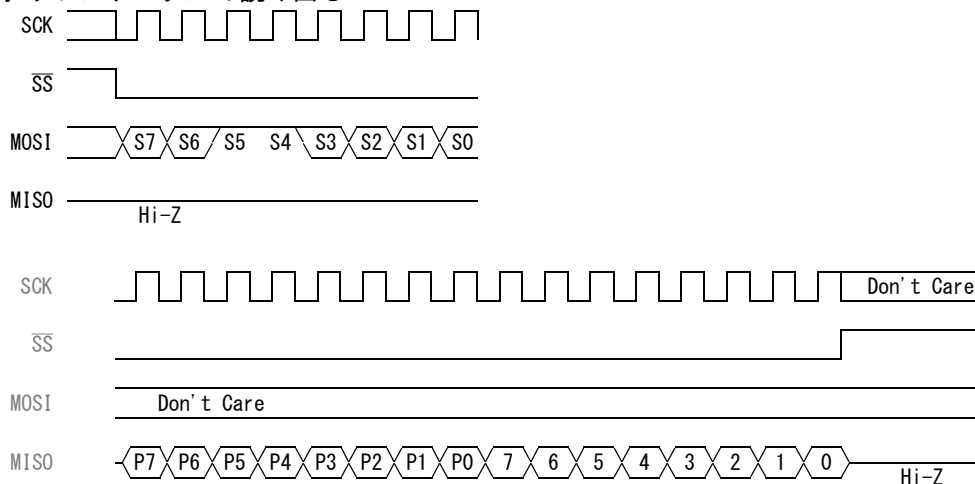
④サブステータス読み出しフォーマット

MOSI : 軸選択コード (S7-S0)

MISO : データx データy データz データu

1回の通信に対して、軸選択コードは1つです。
 データの数は軸選択コードで選択した軸数分を読み出します。
 データのサイズは16bitで、並び順は“データ [7-0]+データ [15-8]”です。
 “データ [7-0]”が汎用ポート入力データで、“データ [15-8]”がサブステータスです。

例. サブステータスの読み出し



S7-S0 : 軸選択コード(サブステータス読み出しフォーマットは、S5="1", S4="1")
 P7-P0 : 汎用ポート読み出しデータ
 7-0 : サブステータス読み出しデータ

注.

1. 軸選択コードで複数軸を選択すると、複数軸に対して1回のシリアル通信でアクセスが行えません。
 その場合、データ順序は選択した軸の中で、X > Y > Z > U の順になります。
2. 複数軸に対する、コマンド(レジスタ)書き込み、汎用ポート書き込みは、SS信号の立ち上がり時に行います。
3. 複数軸に対する、レジスタ読み出しは、“00”ビット書き込み時の状態をラッチして、結果を順次行います。
4. 複数軸に対する、メインステータス読み出し、サブステータス(汎用ポート)読み出しも、“S0”ビット書き込み時の状態をラッチして、結果を順次行います。
5. フォーマット通りのビット数を書き込まず、制御を中断(書き込み途中でSS信号を立ち上げ)した場合は、予期せぬデータが書き込まれてしまいますので、ご注意ください。
6. フォーマット通りのビット数を読み出さず、制御を中断(読み出し途中でSS信号を立ち上げ)した場合は、残りのデータが破棄されてしまいますので、ご注意ください。

6-2. 軸選択コード

軸選択コードは 1byte です。下記の 8bit から成ります。

Bit	PCL6114	PCL6124	PCL6144
0	X 軸選択	X 軸選択	X 軸選択
1	("0"固定)	Y 軸選択	Y 軸選択
2	("0"固定)	("0"固定)	Z 軸選択
3	("0"固定)	("0"固定)	U 軸選択
4	タイプ選択 A		
5	タイプ選択 B		
6	デバイス選択 0		
7	デバイス選択 1		

①軸選択 (Bit3-0)

書き込み、または読み出し対象の軸を選択します。

該当する軸選択ビットが"1"の場合に、選択している軸となります。

すべての軸選択ビットが"0"の場合は、X 軸のみ選択 ("0001"b と同じ) になります。

②タイプ選択 (Bit5, 4)

4 種類の通信フォーマットを選択します。

タイプ選択		通信フォーマット
B	A	
0	0	コマンド書き込み
0	1	メインステータス読み出し
1	0	汎用ポート書き込み
1	1	サブステータス、汎用ポート読み出し

③デバイス選択 (Bit7, 6)

通常は 1 つのスレーブセレクト信号 (\overline{SS}) で 1 つの LSI のみを接続しますが、本 LSI は最大 4 つの LSI まで拡張接続が行えます。

デバイス選択 1 とデバイス選択 0 のビットは、デバイス選択番号端子 (DS1, DS0) に対応しています。

デバイス選択番号端子の設定が異なる複数の LSI に、同時アクセスは行えません。

デバイス選択		デバイス選択 番号端子	
1	0	DS1	DS0
0	0	L	L
0	1	L	H
1	0	H	L
1	1	H	H

6-3. コマンド

コマンドコードは 1byte です。パラレル I/F 時のコマンドと同じです。

6-4. データ

データは 1byte の整数倍で、下位バイトから上位バイトの順に、各バイトは Bit7(MSB)から Bit0(LSB)の順に並びます。データは 8bit 未満の場合も、不足ビットに“0”を代用して 1byte 単位でご設定ください。

レジスタ書き込み時のデータが 4byte 未満の場合も、不足バイトに 00h を代用して 4byte 単位でご設定ください。複数軸レジスタ一括書き込みの場合も同様に、1 軸あたり 4byte 単位でご設定ください。

——— 以上 ———