

PCI Bus

# HPCI-CTR544F

(CCL3221搭載)

## ユーザーズマニュアル

45MHz・多機能 32ビット・カウンタ・ボード



株式会社ハイバーテック

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

## この説明書は

次のCCL3221搭載 **HPCI-CTR544F** のボードに適応しています。

---

本書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下さい。

Windows95, Windows98, WindowsNT 4.0, Windows2000, WindowsXP Home Edition, WindowsXP Professional,  
VisualC++, Visual Basic, Microsoft C/C++ 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 ハイバーテック  
東京都江東区新大橋 1-8-11  
三井生命新大橋ビル 6F  
TEL 03-3846-3801  
FAX 03-3846-3773  
sales@hivertec.co.jp

第1.20版 2010年10月 6日発行  
不許複製・転載

## 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、2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、予めご了承下さい。

## 免責事項

1. 本マニュアル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、予めご了承下さい。
2. 本製品は、一般電子機器用（工作機械・計測機器・F A / O A 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、予めご了承下さい。
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、予めご了承下さい。
4. 本マニュアル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、予めご了承下さい。

## 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。

本書は、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、本書が添付された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。

## 1. 対象ユーザー



**注意**

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。



- ・ 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。

## 2. 運搬・取り付け



### 警告



本製品にふれる前に、金属に触り身体の静電気を取り除いて下さい。  
静電気は、本ボードの故障の原因になります。



本製品を静電気の帯びやすい梱包材（エアークャップなど）でくるまないで下さい。  
静電気は、本ボードの故障の原因になります。



本製品のエッジコネクタ部分に触らないで下さい。  
エッジコネクタ部分が汚れますと、誤動作の原因になります。



本製品の上に重いものを載せないで下さい。重いものを乗せると、部品が損傷し  
故障の原因になります。



本製品のジャンパ設定は、パソコン等に取り付ける前に行ってください。電源がONの  
状態で設定しますと、設定を正しく認識しないで誤動作の原因になります。



本製品のジャンパ設定は、正しく行って下さい。  
設定を間違えますと誤動作の原因になります。



本製品をパソコン等に取り付ける時は、必ずパソコン等の電源をOFFにし、電源  
コードを抜いてから作業を行ってください。電源コードを抜かないで作業を行った場合、  
故障の原因になります。また、装置が思わぬ動作をすることがあります。



### 注意



本製品を落としたり乱暴に扱ったりしないで下さい。  
衝撃や振動が故障の原因となります。



本製品の半田面を手で直接触らないで下さい。  
部品の突起などにより怪我をする恐れがあります。

### 3. 配線

#### 警告



外線用コネクタへの配線作業や外線用コネクタの着脱は、パソコン等の電源をOFFし、電源コードを抜いてから行って下さい。

電源コードを抜かないで作業を行った場合、故障の原因になります。また、装置が思わぬ動作をすることがあります。



外線用コネクタへの配線は、コネクタ信号表などをよく確認し、正しく配線して下さい。間違った配線をしますと、故障・焼損の原因になります。



外部から供給する電源は、必ず定格以内でご使用下さい。定格以外で使用されますと、故障・焼損・誤動作の原因となります。



入出力回路に接続する回路は、必ず定格電流・電圧以内でご使用下さい。定格以外で使用されますと、故障・焼損・誤動作の原因となります。



外部配線用コネクタは、推奨のコネクタをご使用下さい。推奨以外のコネクタを使用されますと、接触不良などにより誤動作の原因となります。



外部配線用コネクタは、必ずロックしてご使用下さい。ロックしないで使用されますと、コネクタが外れる、または接触不良などにより誤動作の原因となります。



外部配線用ケーブルは、引っ張る、または重い荷重を掛けないで下さい。コネクタが外れる、または接触不良などにより誤動作の原因となります。



外部配線用ケーブルは、モーターの配線やAC電源ケーブルなど、ノイズの多い配線とは出来るだけ離して下さい。配線が近いとノイズが誤動作の原因となります。

### 4. 試運転・調整

#### 警告



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りますと、思わぬ動きをすることがあります。



本製品に添付してあるプログラムを使用し装置を動作させる時、機械系に合った設定を行って動作を確認して下さい。機械系に合わない設定で動作を行うと思わぬ動きをすることがあります。

## 5. 廃 棄



警 告



本製品を廃棄する時は、関連する法律・規則に従って処理して下さい。

# 目 次

1. はじめに.....	1
1. 1 このマニュアルの表記について.....	1
2. CTRボードの機能・概要.....	2
2. 1 CTRボードの機能・概要.....	2
3. 仕 様.....	3
4. ボード上の設定.....	4
5. ボード構成とポートアドレス.....	5
5. 1 ブロック図.....	5
5. 2 ポートアドレス.....	6
6. コマンド・ステータス・レジスタ.....	7
6. 1 ポートおよびレジスタの書込み、読出し.....	7
6. 2 レジスタ制御コマンド.....	7
6. 3 単独コマンド.....	7
6. 4 ステータス.....	8
6. 4. 1 ステータス (STS) .....	8
6. 4. 2 割込みステータスレジスタ (RIST) .....	8
6. 4. 3 割込み要因設定レジスタ (RIRQ) .....	9
6. 5 レジスタ.....	10
6. 5. 1 カウンタ (CTR1, CTR2) .....	10
6. 5. 2 比較データレジスタ (RCMP1~RCMP4) .....	10
6. 5. 3 イベントタイマレジスタ (ETMR) .....	10
6. 5. 4 ラッチレジスタ (LTCH1, LTCH2) .....	10
6. 5. 5 最大値レジスタ (MAX1, 2), 最小値レジスタ (MIN1, 2) .....	10
6. 5. 6 環境レジスタ1 (RENV1) .....	11
6. 5. 7 環境レジスタ2 (RENV2) .....	13
6. 6 オプションポート.....	15
7. 基本的な設定と運用.....	22
7. 1 操作手順.....	22
7. 2 エンコーダ等パルス計数 (カウント) .....	22
7. 2. 1 カウントパルスの入力信号形式.....	22
8. コネクタ信号表.....	24
9. 外部との接続.....	26
9. 1 カウンタ入力及Z相入力信号.....	26
9. 2 汎用入力.....	26
9. 3 汎用出力及び一致出力.....	26
10. 基板寸法図.....	27
11. ソフトウェア編.....	28
11. 1 ソフトウェアの概要.....	29
11. 2 Windows版のインストールとアンインストール.....	30
11. 3 準 備.....	31
11. 4 ドライバ関数の戻り値.....	33
11. 5 ドライバ関数詳細.....	34
11. 5. 1 Windows版ドライバ関数.....	35
11. 5. 2 Windows版サンプル (動かしてみる) .....	44
11. 5. 3 「動かしてみる」の操作.....	44

## 図 表 目 次

表 1.	略称呼称	1
表 2.	HPCI-CTR544F仕様	3
図 3.	ボード設定箇所	4
図 4.	ボードID設定ジャンパ	4
図 5.	エンコーダ入力回路終端ジャンパ	4
図 6.	ブロックダイア	5
表 7.	アドレスマップ	6
表 8.	レジスタ制御コマンド	7
表 9.	単独コマンド	7
表 10.	ステータス内容	8
表 11.	割込みステータスの内容	8
表 12.	割込み要因設定レジスタの内容	9
表 13.	環境レジスタ1 (RENV1)	12
表 14.	環境レジスタ2 (RENV2)	14
図 15.	入力状況確認レジスタ	15
表 16.	入力状況確認レジスタ	15
図 17.	汎用入力状況確認レジスタ (di_reg)	15
表 18.	汎用入力状況確認レジスタの詳細	15
図 19.	汎用出力レジスタ	15
図 20.	CMP出力レジスタ	16
表 21.	CMP出力レジスタの詳細	16
図 22.	ITMトグル出力レジスタ	16
表 23.	ITMトグル出力レジスタの詳細	16
図 24.	出力パルス幅レジスタ	16
図 25.	Z相カウンタクリアレジスタ	17
表 26.	Z相カウンタクリアレジスタの詳細	17
図 27.	カウンタラッチ条件レジスタ	18
表 28.	カウンタラッチ条件レジスタの詳細	18
図 29.	割込み要因 (ソース) レジスタ	18
表 30.	割込み要因 (ソース) レジスタの詳細	18
図 31.	割込みマスク レジスタ	19
表 32.	割込みマスク レジスタの詳細	19
図 33.	割込みステータス レジスタ	19
表 34.	割込みステータス レジスタの詳細	19
図 35.	ボードIDレジスタ	19
図 36.	入力フィルターレジスタ	20
表 37.	入力フィルターレジスタの詳細	20
図 38.	入力割込みマスクレジスタ	20
表 39.	入力割込み許可レジスタの詳細	20
図 40.	入力割込みクリア・ステータスレジスタ	20
表 41.	入力割込み許可レジスタの詳細	20
図 42.	配線状況確認レジスタ	21
表 43.	配線状況確認レジスタの詳細	21
図 44.	計測モード設定レジスタ	21
表 45.	計測モード設定レジスタの詳細	21
図 46.	基本的な操作手順	22
表 47.	カウントパルスの入力信号形式	22
図 48.	各入力信号形式におけるカウント動作のタイミング	23
表 49.	J1コネクタ信号表	24
表 50.	J2コネクタ信号表	24
表 51.	カウンタ入力及びZ相入力信号	26
表 52.	汎用入力	26
表 53.	汎用出力及び一致出力	26

図 54.	基板寸法図	27
表 55.	ドライバ関数一覧	28
図 56.	ソフトウェアの関連	29
表 57.	ドライバ関数の戻り値	33
表 58.	ドライバ関数のデータ型	34
表 59.	各言語の数値表記	34

# 1. はじめに

このマニュアルは、高速カウンタ LSI CCL3221 (2ch) を搭載した高速・多機能32ビット・アップ/ダウン・カウンタボード「HPCI-CTR544F」の取扱説明書です。

製品には、通常次の説明資料が付属します。

- ① HPCI-CTR544F ユーザーズマニュアル (このマニュアル)
- ② 添付ソフトウェア (FD)

## 1. 1 このマニュアルの表記について

カウンタ LSI CCL3221 は「2チャンネルのアップ/ダウンカウンタ と4組のコンパレータ および 1組のイベントタイマ」で構成されています。

4chのボードは XYchの機能のCCL3221を#1CCLと称し、もう一方のそれを#2CCLと呼び、このch3をZch, ch4をUchと称します。主な呼称を次に掲げます。

名 称	呼 称		代表呼称	備 考
1, 2ch/3, 4ch チャンネル	#1 CCL	#2 CCL	CCL	
	Xch Ych	Zch Uch	ch1 ch2	
カウンタ	XCTR YCTR	ZCTR UCTR	CTR1 CTR2	32ビット アップ/ダウン カウンタ
コンパレータ	XYCMP1 XYCMP2 XYCMP3 XYCMP4	ZUCMP1 ZUCMP2 ZUCMP3 ZUCMP4	CMP1 CMP2 CMP3 CMP4	1個のCCLに4組
コンパレータ一致信号	XYCMPOUT1 XYCMPOUT2 XYCMPOUT3 XYCMPOUT4	ZUCMPOUT1 ZUCMPOUT2 ZUCMPOUT3 ZUCMPOUT4	CMPOUT1 CMPOUT2 CMPOUT3 CMPOUT4	1個のCCLに4組
比較データレジスタ	XYRCMP1 XYRCMP2 XYRCMP3 XYRCMP4	ZURCMP1 ZURCMP2 ZURCMP3 ZURCMP4	RCMP1 RCMP2 RCMP3 RCMP4	比較データレジスタも 4組あるが、 CMP1~CMP4の いずれにも対応する
イベントタイマ	XYETMR	ZUETMR	ETMR	周期割込みタイマ
イベントタイマ信号	XYETMROUT	ZUETMROUT	ETMROUT	周期割込みタイマ信号

表 1. 略称呼称

なお、本書の説明文中ではHPCI-CTR544Fのボードの名称をCTRボードと呼称します。

## 2. CTRボードの機能・概要

### 2. 1 CTRボードの機能・概要

CTRボードは概略次の機能項目があります。

#### (1) エンコーダ等パルス計数(カウント)

カウント開始から終了までの間に到来するパルスカウントを行います。計数パルスはエンコーダなど「位相差(2相)パルス」、単に「アップ/ダウンパルス」、「カウント方向とパルス列」です。これらは信号形式と言います。

#### 【応用】

##### ●時間当たりカウント

- ① PC(パソコン)から時間単位でCTRラッチ。
- ② イベントタイマを利用して周期ごとにCTRラッチ。

##### ●外部の信号によりカウント値をラッチ

- ① XZ, YZ, などZ相信号入力により全CTR同時ラッチ。
- ② 汎用入力 IN1 信号入力により全CTR同時ラッチ。

##### ●最大値-最小値計測

- ① アップ/ダウンカウント計測期間中の最大値及び最小値を計測。
- ② 信号幅, パルス周期, 位相差計測期間中の最大値及び最小値を計測。

#### (2) コンパレータ

#1CCL (#2CCL)には4組のコンパレータ(以下CMP)があります。これらのCMPを利用して主として次の機能が実現できます。

- ① CMP条件成立によりPCへ割込み。(カウント周期的にPCに通知できる)
- ② CMP条件成立によりCTRクリア。(イコール比較による)
- ③ CMP条件成立により外部へパルス出力(A/Dコンバータサンプリング・トリガなど)
- ④ 2次元エリアのウィンドウ・コンパレート(例:XYエリアにあるとき,XYOUT1信号 ON)

#### (3) イベントタイマ

#1CCL (#2CCL)には1組のTIMER(EVENT TIMER)があります。このTIMER機能は次の通りです。

- ① TIMER周期でCTRラッチ,ラッチ後CTRクリア(同一CCLのChのみ),全CTR同時ラッチ。
- ② TIMER周期で一致出力。
- ③ TIMER周期でPCへ割込み(DOS版ソフトのみ対応)

### 3. 仕様

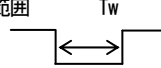
区分	項目	仕様	備考		
【パルス入力部】	■ カウンタ数	4 CH			
	■ カウンタ長	32ビットバイナリ $-2^{31} \sim +2^{31} - 1$ (-2,147,483,648 ~ 2,147,483,647)			
	■ カウンタ入力信号形式	3形式をソフトで選択 (1) 位相差(2相)パルス (2) アップ/ダウン個別パルス (3) 方向/パルス列			
	■ カウント周波数	カウント入力信号	最大カウント周波数 (CPS)	指定ケーブル以外 使用時の周波数は 4 通倍時 35MHz に なります。	
			フィルタ ON		フィルタ OFF
		90° 位相差信号 (1 通倍)	7 M		1 2 M
		90° 位相差信号 (2 通倍)	1 5 M		2 2 M
90° 位相差信号 (4 通倍)		3 0 M	4 5 M		
	アップ/ダウン個別パルス	1 5 M	2 2 M		
	方向/パルス列	1 5 M	2 2 M		
■ カウント信号入力回路 デジタルフィルタ付	差動レシーバ受け (26C32 相当)				
【カウント機能部】	■ アップダウンパルス・カウント	32ビットバイナリ・アップダウン・カウント	2相,UP/DOWN 方向//パルス列		
	■ パルス幅, 周期, 位相差計測 (エッジ間の時間測定) 1 Shotモード有り (Z相または IN1 入力でトリガ)	計測範囲  $40\text{ ns} \leq T_w \leq 43\text{ sec}$ 分解能 20 ns	YCTR, UCTR のみ		
	■ カウンタ読出後カウンタクリア	カウンタ読出後、カウンタ自動クリア			
【機能仕様】	■ MAX-MIN測定	アップダウン・カウント時の最大値-最小値測定			
	■ コンパレータ機能 ● セット数 ● 比較結果  ● 条件成立時の機能  ● ウィンドウ・コンパレート機能	4式の32ビット長コンパレータ(CMP1~4)			
		1. ステータスおよび割込みステータスで確認可 2. 条件成立を XYOUT1, XYOUT2(ZUOUT1, ZUOUT2)により外部出力可			
		1. イコール比較成立時比較カウンタクリア可 2. 条件成立で割込み出力可			
	■ Z相 (外部信号) 入力	Z相入力 (各ch当り1信号) でカウンタクリア, ラッチ, ラッチ後クリア, 選択Z相同時ラッチ			
	■ 同時ラッチ機能	コンパレータ一致, Z相入力, IN1入力, イベントタイマにより同時ラッチ			
	■ 個別ラッチ機能	各ch Z相入力で各chラッチ			
	■ イベントタイマ機能 ● 周期毎の機能	周期割込みタイマでサイクル毎に IRQ 割込み出力可 割込み周期 16 μs ~ 134, 21772 sec サイクル毎に XCTR, ~, UCTR の内容をラッチ可 またはラッチ後カウンタクリア可. XYETM(ZUETM)より出力			
【外部入出力機能】	■ カプラ絶縁汎用入出力 ● 入力ポート	4 IN / 4 OUT DIOポート			
		入力論理 : 入力フォトカプラON時 '1' 定格入力電圧 : DC5V 使用入力電圧範囲: DC5V±5% 定格入力電流 : 8mA/1点			
	● 出力ポート	出力論理 : '1' 書込時 出力ON 定格負荷電圧 : DC5V 負荷電流 : 5mA以下/1点			
		■ 一致出力 (XYOUT1, XYOUT2, ZUOUT1, ZUOUT2)	出力形式 : フォトカプラ絶縁オープンコレクタ出力 出力定格 : DC5V 負荷電流 : 5mA以下/1点 一致信号出力幅 : 100ns~5ms	ソフト設定	
■ イベントタイマ出力(XYETM, ZUETM)	イベントタイマ周期毎に外部出力 出力幅 : 100ns~5ms				
【周囲条件】	■ 供給電源	+5V±5%			
	■ 温度条件	0°C ~ 50°C 但し結露せぬこと			
	■ ボード形寸	横175mm × 縦107mm (シヨートサイズ)			

表 2. HPCI-CTR544F仕様

## 4. ボード上の設定

CTRボードの設定箇所は、ボードIDとエンコーダ入力回路終端抵抗有/無選択です。

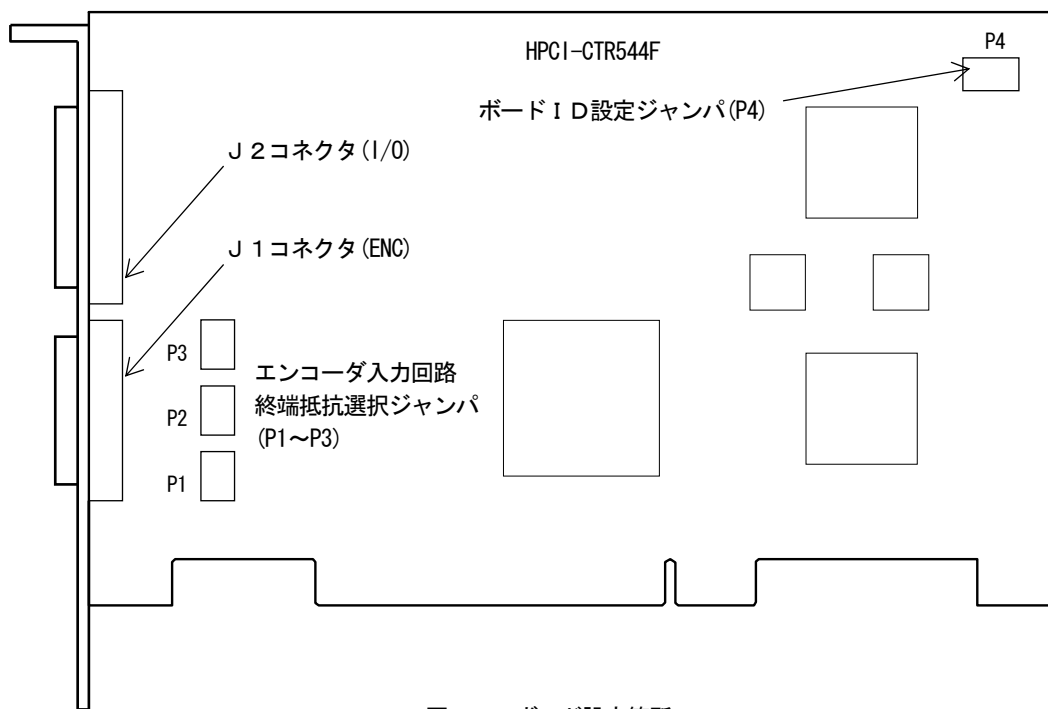


図 3. ボード設定箇所

### (1) ボードID設定 ジャンパ

CTRボードで次のボードはボード上のジャンパで設定したボードID (0~15) が使用できます。ボードIDの設定値とジャンパ状態は次表のようになります。(出荷時 ID=0)

ボードID設定値	0	5	7	10	15
ジャンパ状態					
(2進表記)	0 0 0 0	0 1 0 1	0 1 1 1	1 0 1 0	1 1 1 1

図 4. ボードID設定ジャンパ

### (2) エンコーダ入力回路終端抵抗有/無選択

エンコーダとカウンタ入力が1対1の場合、あるいは、このボードがケーブルの最後に接続される場合は抵抗で終端します。

1	8 1-8 ジャンパ XA100Ω終端	1	8 1-8 ジャンパ XB100Ω終端	1	8 1-8 ジャンパ XZ120Ω終端
	2-7 ジャンパ YA100Ω終端		2-7 ジャンパ YB100Ω終端		2-7 ジャンパ YZ120Ω終端
	3-6 ジャンパ ZA100Ω終端		3-6 ジャンパ ZB100Ω終端		3-6 ジャンパ ZZ120Ω終端
4	5 4-5 ジャンパ UA100Ω終端	4	5 4-5 ジャンパ UB100Ω終端	4	5 4-5 ジャンパ UZ120Ω終端
P1ジャンパ: XA, ~, UA		P2ジャンパ: XB, ~, UB		P3ジャンパ: XZ, ~, UZ	

出荷時: 全て終端あり

図 5. エンコーダ入力回路終端ジャンパ

# 5. ボード構成とポートアドレス

## 5.1 ブロック図

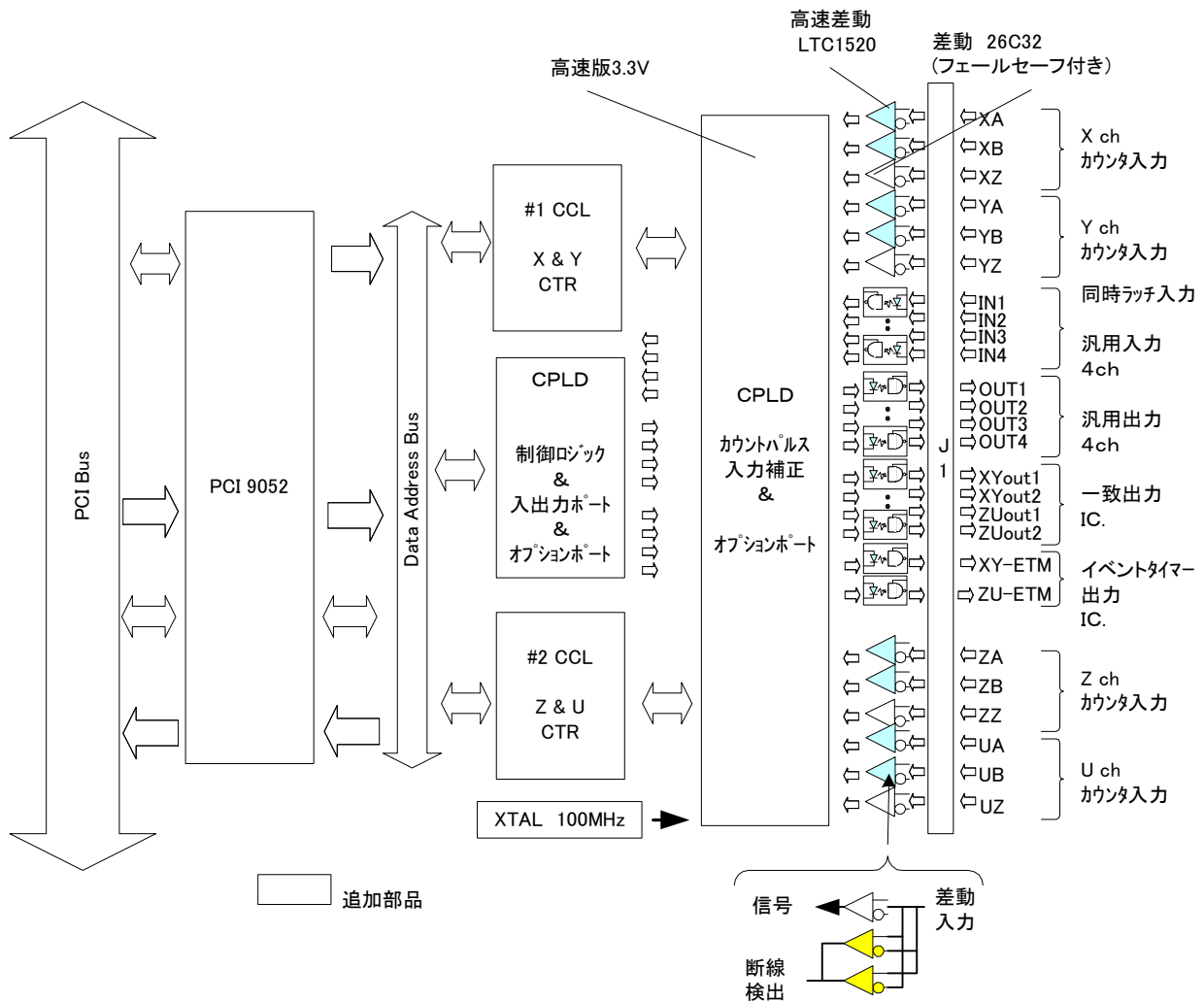


図 6. ブロックダイア

## 5. 2 ポートアドレス

ポートはすべてI/Oマップです。

表7 にHPCI-CTR544Fポート表を示します。

(このボードでは、80h(128)バイトを占有しています。)

I/O Map	対応デバイス	読み込み (Read)		書込み (Write)		Reg.bits	初期値
		ラベル	内容	ラベル	内容		
+00	#1 CCL	STS1	ステータス REG	CMD	コマンド REG		
+02		STS2	ステータス REG	—	未使用 (予約)		
+04		BUF0	下位データ読み出し	BUF0	下位書込みデータ		
+06		BUF1	上位データ読み出し	BUF1	上位書込みデータ		
+08	#2 CCL	STS1	ステータス REG	CMD	コマンド REG		
+0a		STS2	ステータス REG	—	未使用 (予約)		
+0c		BUF0	下位データ読み出し	BUF0	下位書込みデータ		
+0e		BUF1	上位データ読み出し	BUF1	上位書込みデータ		
+20	#1 CPLD CPL069	z_sts	入力状況モニタ	—		—	
+24							
+28		di_reg	汎用入力4点	—		—	
+2c		do_reg	汎用出力状況	do_reg	汎用出力	3.0	0000H
+30		sw_and	Xyout 条件状況	sw_and	Xyout 条件	15.0	0000H
+32		sw_tgl	ETM 出カトグル	sw_tgl	ETM 出カトグル	15.0	0000H
+34		pwm_rg	XYOUT パルス幅	pwm_rg	XYOUT パルス幅	15.0	0000H
+36		sw_clr	Z相カウンタクリア	sw_clr	Z相カウンタクリア	7.0	0000H
+38		sw_ltc	ラッチ条件	sw_ltc	ラッチ条件	15.0	0000H
+3a		f_infl	入力フィルタ	f_infl	入力フィルタ	9.0	0000H
+3c	f_inte	入力割り込み許可	f_inte	入力割り込み許可	9.0	0000H	
+3e	f_ints	入力割り込みステータス	f_intc	入力割り込みクリア	9.0		
+40	f_cabs	ケーブルステータス	—		—		
+4a	f_1sht	計測モード	f_1sht	計測モード	15.0	0000H	
+50	#1 CPLD CPL069	i_src	INT ソース・ステータス	i_clr	割り込みクリア	—	
+52		i_enab	割り込み許可	i_enab	割り込み許可	15.0	0000H
+54		i_sts	割り込みステータス			15.0	
+60		bd_id	ボードID	—		3.0	

表 7. アドレスマップ

## 6. コマンド・ステータス・レジスタ

### 6. 1 ポートおよびレジスタの書込み, 読出し

「表7. アドレスマップ」に示すように#xCCLにCMD, BUF0, BUF1, STS ポートがあります。全体に共通に1組のオプションポート（割込みおよび外部出力ルート設定, 4bit 汎用ポート）があります。CCLに対するコマンド, データは, CMD, BUF x ポート経由で指定するレジスタ, カウンタに書くことによりセットします。データを伴わないような 測定開始, CTR LTCH, CTR CLR などのコマンドのみの場合はCMD ポートに書くだけで行います。レジスタを読み出す場合はCMD ポートに読出しコマンドを書きます。その結果, BUF0, BUF1 に読み出されます。

### 6. 2 レジスタ制御コマンド

次表にレジスタ制御コマンド一覧を掲げます。

項	レジスタ名	内 容	書込みコマンド	読出しコマンド	ビット長	データ範囲	参照頁
1	CTR1 (XCTR, ZCTR)	カウンタ 1	0080h	00c0h	32	-2,147,483,648   2,147,483,647	8
2	CTR2 (YCTR, UCTR)	カウンタ 2	0081h	00c1h	32		8
3	RCMP1	比較データレジスタ 1	0082h	00c2h	32		8
4	RCMP2	比較データレジスタ 2	0083h	00c3h	32		8
5	RCMP3	比較データレジスタ 3	0084h	00c4h	32		8
6	RCMP4	比較データレジスタ 4	0085h	00c5h	32		8
7	ETMR	イベントタイマレジスタ	0086h	00c6h	24	2~16,777,215	8
8	RENV1	環境レジスタ 1 (入力仕様設定)	0087h	00c7h	32	logical data	9
9	RENV2	環境レジスタ 2 (コンパレータ条件設	0088h	00c8h	30	logical data	9
10	RIRQ	割込み要因設定レジスタ	0089h	00c9h	27	logical data	7
11	LTCH1 (XLTCH, ZLTCH)	カウンタ 1 のラッチレジスタ	/	00cah	32	-2,147,483,648   2,147,483,647	8
12	LTCH2 (YLTCH, ULTCH)	カウンタ 2 のラッチレジスタ		00cbh	32		8
13	MAX1 (XMAX, ZMAX)	カウンタ 1 の最大値レジスタ		00cch	32		8
14	MIN1 (XMIN, ZMIN)	カウンタ 1 の最小値レジスタ		00cdh	32		8
15	MAX2 (YMAX, UMAX)	カウンタ 2 の最大値レジスタ		00ceh	32		8
16	MIN2 (YMIN, UMIN)	カウンタ 2 の最小値レジスタ		00cfh	32		8
17	RIST	割込みステータス・レジスタ		00d0h	27	logical data	6

表 8. レジスタ制御コマンド

### 6. 3 単独コマンド

次表に単独コマンド一覧を掲げます。これらのコマンドはデータを伴わず、単独にCMD ポートに書込み、動作します。

項	コマンド名	内 容	CMDコード	備 考
1	CLR CTR1	CTR1 を 0 クリア	0001h	カウント動作中のCTR読出しは, LTCHコマンドを出し, ラッチレジスタ (LTCH) から読む。 全レジスタ, BUF0, 1 は 0 クリアされる。  MMSTART で CTR, MAX, MIN は 0 クリアされる。 注意: CTR 1, CTR 2 の通常のカウンタレジスタは RENV 1 の各々 b12, b13 で行う。
2	CLR CTR2	CTR2 を 0 クリア	0002h	
3	INIT MM1	CTR1 の MAX1, MIN2 を初期化	0004h	
4	INIT MM2	CTR1 の MAX2, MIN2 を初期化	0008h	
5	LTCH CTR1	CTR1 を LTCH1 にラッチ読出し	0010h	
6	LTCH CTR2	CTR2 を LTCH2 にラッチ読出し	0020h	
7	SRST	ソフトリセット	0040h	
8	SLTCH	全 ch 同時ラッチ	0041h	
9	MM1 START	CTR1 の 最大値最小値の測定開始	0048h	
10	MM1 STOP	CTR1 の 最大値最小値の測定終了	0049h	
11	MM2 START	CTR2 の 最大値最小値の測定開始	004ah	
12	MM2 STOP	CTR2 の 最大値最小値の測定終了	004bh	

注: カウンタスタートは RENV 1 b 1 2, b 1 3 で行います。

表 9. 単独コマンド

## 6. 4 ステータス

ステータスは直接ポートから読出す「STS」とレジスタとして読出す割込ステータス「RIST」があります。

### 6. 4. 1 ステータス (STS)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTS	*	*	*	*	*	*	*	CTR2MM	CTR1MM	CTR2S	CTR1S	CMP4	CMP3	CMP2	CMP1

bit	名称	説明	備考
0	CMP1	1:CMPOUT1 出力中	これらは（イコール比較など）カウンタが動作中の場合は正しい状態を反映しません。このような場合は INTS(bit15)を使用します。
1	CMP2	1:CMPOUT2 出力中	
2	CMP3	1:CMPOUT3 出力中	
3	CMP4	1:CMPOUT4 出力中	
4	CTR1S	1:CTR1 停止状態, 0:CTR1 動作状態	
5	CTR2S	1:CTR1 停止状態, 0:CTR2 動作状態	
6	CTR1MM	1:CTR1 MM 測定状態, 0:CTR1 MM 測定停止状態	
7	CTR2MM	1:CTR2 MM 測定状態, 0:CTR2 MM 測定停止状態	
8	*	DON'T CARE (通常保守用途)	
~	~		
14	*		
15	INTS	1:割込み要求中: 割込み内容は「6.4.2 割込みステータス」による	RIST 読出し後, ビットはクリアされる。

表 10. ステータス内容

### 6. 4. 2 割込みステータスレジスタ (RIST)

RISTの各ビットは, RIST読出し後リセットされます。さらに, STS bit 15 (INTS) がリセットされます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
← 不使用 →	LTCH2	LTCH1	CLR2	CLR1	CMP4	CMP4	CMP3	CMP3	CMP2	CMP2	CMP1	CMP1	CMP1	CMP1	
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	ETMRINT	MEND	SLTCHI	C2OVFN	C2OVFP	C1OVFN	C1OVFP	ENC2ER	ENC1ER	CTR2=0	CTR1=0

bit	名称	説明	備考
0	CMP1	1: CMP 1が条件成立し, 割込	
1	CMP1	1: CMP 1の条件成立が解けて, 割込	
2	CMP2	1: CMP 2が条件成立し, 割込	
3	CMP2	1: CMP 2の条件成立が解けて, 割込	
4	CMP3	1: CMP 3が条件成立し, 割込	
5	CMP3	1: CMP 3の条件成立が解けて, 割込	
6	CMP4	1: CMP 4が条件成立し, 割込	
7	CMP4	1: CMP 4の条件成立が解けて, 割込	
8	CLR1	1: CTR 1にZ相入力によるクリアがかかり, 割込	Z相端子からのCTRクリア
9	CLR2	1: CTR 2にZ相入力によるクリアがかかり, 割込	Z相端子からのCTRクリア
10	LTCH1	1: CTR 1 XZ (ZZ) 信号入力によるラッチがかかり, 割込	Z相個別ラッチを
11	LTCH2	1: CTR 2 YZ (UZ) 信号入力によるラッチがかかり, 割込	選択した場合 X Z (ZZ) 個別ラッチ Y Z (UZ) 個別ラッチ
12	~	不使用	
15			
16	CTR1=0	1: CTR1 が0になり, 割込	
17	CTR2=0	1: CTR2 が0になり, 割込	
18	ENC1ER	1: Xch (Zch) の+/-カット信号が同時に変化した割込	
19	ENC2ER	1: Ych (Uch) の+/-カット信号が同時に変化した割込	
20	C1 OVFP	1: CTR1 の+側のオーバーフロー発生, 割込	
21	C1 OVFN	1: CTR1 の-側のオーバーフロー発生, 割込	
22	C2 OVFP	1: CTR2 の+側のオーバーフロー発生, 割込	
23	C2 OVFN	1: CTR2 の-側のオーバーフロー発生, 割込	
24	SLTCHI	1: 全 ch 同時ラッチ信号入力での割込	外部信号による全 ch 同時ラッチ, 同時ラッチ CMD (41h)
25	MEND	1: 幅計測終了エッジでの割込	
26	ETMRINT	1: イベントタイム周期割込み	※タイムスタート時にも割込み
27~31		不使用 a l l '0'	

表 11. 割込みステータスの内容

### 6. 4. 3 割り込み要因設定レジスタ (RIRQ)

RIRQの割り込みイネーブルに設定したビットが割り込み対象となります。電源投入直後 RIRQは全て割り込みディセーブル状態です。単独コマンドの「ソフトウェアリセット」もRIRQを全て'0'にします。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	LTCH2M	LTCH1M	CLR2M	CLR1M	CMP4M1	CMP4M0	CMP3M1	CMP3M0	CMP2M1	CMP2M0	CMP1M1	CMP1M0
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0	0	0	0	0	ETMRM	MENDM	SLTCHM	0	C2OVFM	0	C1OVFM	ENC2M	ENC1M	C2ZRO	C1ZRO

bit	名称	説明	備考
0	CMP1 MO	1: CMP 1が条件成立 割り込みイネーブル.	
1	CMP1 M1	1: CMP 1の条件成立が解けて 割り込みイネーブル	
2	CMP2 MO	1: CMP 2が条件成立 割り込みイネーブル	
3	CMP2 M1	1: CMP 2の条件成立が解けて 割り込みイネーブル.	
4	CMP3 MO	1: CMP 3が条件成立 割り込みイネーブル	
5	CMP3 M1	1: CMP 3の条件成立が解けて 割り込みイネーブル	
6	CMP4 MO	1: CMP 4が条件成立 割り込みイネーブル.	
7	CMP4 M1	1: CMP 4の条件成立が解けて 割り込みイネーブル	
8	CLR1M	1: CTR 1にZ相入力によるクリア 割り込みイネーブル.	
9	CLR2M	1: CTR 2にZ相入力によるクリア 割り込みイネーブル	
10	LTCH1M	1: CTR 1 XZ (ZZ) 信号入力による 割り込みイネーブル	Z相入力による個別ラッチ使用時のラッチ c h 判別用
11	LTCH2M	1: CTR 2 YZ (UZ) 信号入力による 割り込みイネーブル.	
12	不使用	0:	
13		0:	
14		0:	
15		0:	
16	C1ZRO	1: CTR1 の内容が0になった時, 割り込みイネーブル	
17	C2ZRO	1: CTR2 の内容が0になった時, 割り込みイネーブル.	
18	ENC1M	1: Xch (Zch) の+/-カット信号が同時に変化 エラー割り込みイネーブル	
19	ENC2M	1: Ych (Uch) の+/-カット信号が同時に変化 エラー割り込みイネーブル	
20	C1 OVFM	1: CTR1 のオーバーフロー発生, 割り込みイネーブル	
21	不使用	0:	
22	C2 OVFM	1: CTR2 のオーバーフロー発生, 割り込みイネーブル	
23	不使用	0:	
24	SLTCHM	1: 全 c h 同時ラッチが発生, 割り込みイネーブル	同時ラッチ
25	MENDM	1: 幅計測終了エッジで割り込みイネーブル	
26	ETMRM	1: イベントタイマ周期割り込みイネーブル	
27~31		不使用 a l l '0'	

表12. 割り込み要因設定レジスタの内容

## 6. 5 レジスタ

### 6. 5. 1 カウンタ (CTR1, CTR2)

#1CCL・・・XCTR: CTR1, YCTR: CTR2

#2CCL・・・ZCTR: CTR1, UCTR: CTR2

31	-----	24	23	-----	16	15	-----	8	7	-----	0
-2, 147, 483, 648 ~ 2, 147, 483, 647											

### 6. 5. 2 比較データレジスタ (RCMP1~RCMP4)

#1CCL, #2CCL共に4種類の比較データレジスタ (RCMP1~RCMP4) があります。

31	-----	24	23	-----	16	15	-----	8	7	-----	0
-2, 147, 483, 648 ~ 2, 147, 483, 647											

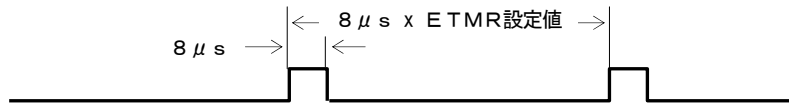
### 6. 5. 3 イベントタイマレジスタ (ETMR)

#1CCL, #2CCL共に1個のイベントタイマ・レジスタがあります。

イベントタイマの周期をこのレジスタで設定します。

設定単位は8μsであり, 設定値80で1msとなり, 信号幅は8μsです。

31	-----	24	23	-----	16	15	-----	8	7	-----	0
0			2 ~ 16, 777, 215 (16μsec ~ 134.216秒)								



### 6. 5. 4 ラッチレジスタ (LTCH1, LTCH2)

#1CCL, #2CCL共に1組のラッチレジスタがあり, 各カウンタと一対となります。

#1CCL・・・XCTR: XLTCH(LTCH1), YCTR: YLTCH(LTCH2)

#2CCL・・・ZCTR: ZLTCH(LTCH1), UCTR: ULTCH(LTCH2)

31	-----	24	23	-----	16	15	-----	8	7	-----	0
-2, 147, 483, 648 ~ 2, 147, 483, 647											

### 6. 5. 5 最大値レジスタ (MAX1, 2), 最小値レジスタ (MIN1, 2)

#1CCL, #2CCL共に2組の最大値・最小値レジスタがあり, 各カウンタと一対となります。

#1CCL・・・XCTR: RXMAX(MAX1), RXMIN(MIN1)

YCTR: RYMAX(MAX2), RYMIN(MIN2)

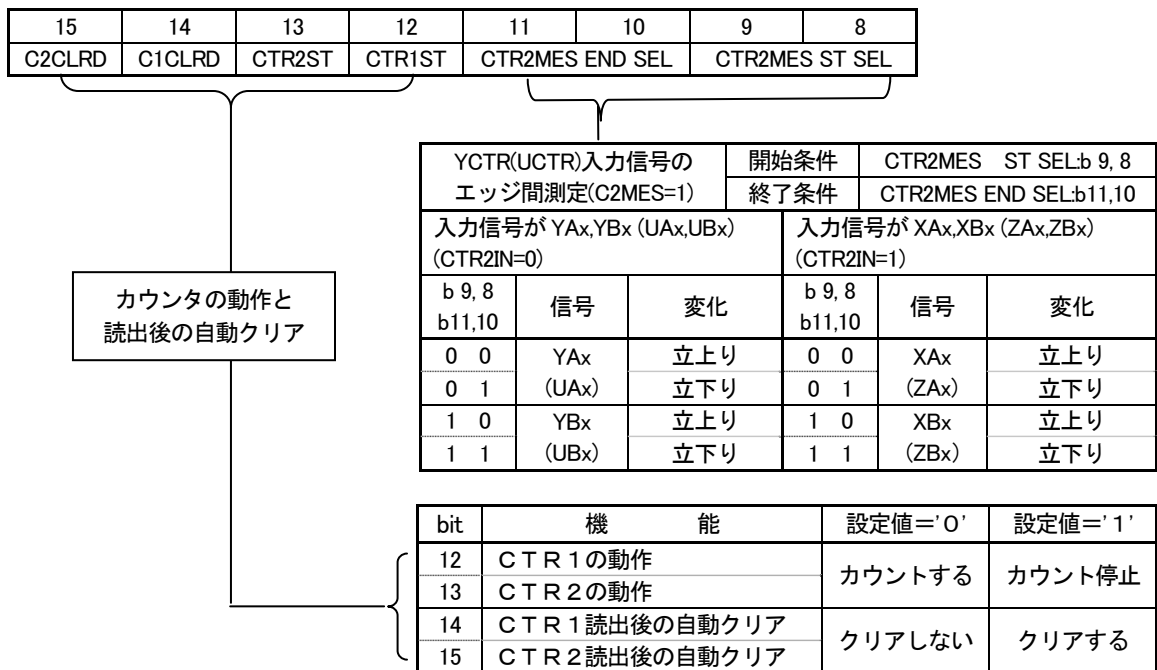
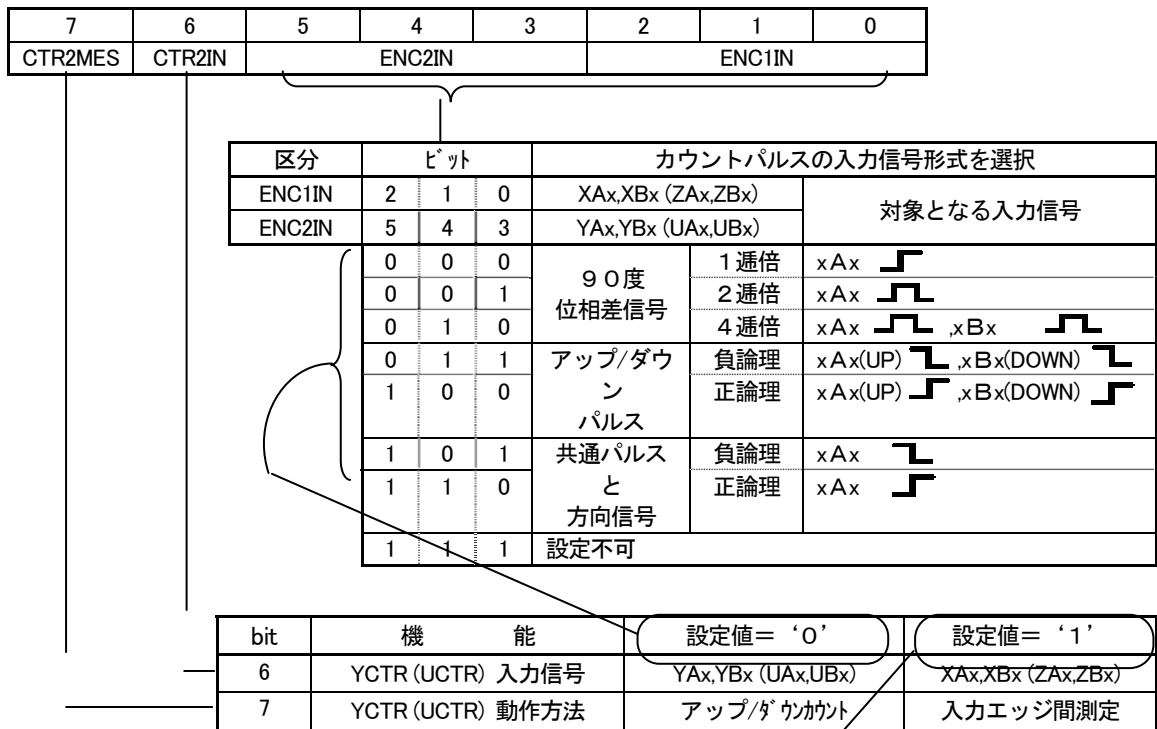
#2CCL・・・ZCTR: RZMAX(MAX1), RZMIN(MIN1)

UCTR: RUMAX(MAX2), RUMIN(MIN2)

31	-----	24	23	-----	16	15	-----	8	7	-----	0
-2, 147, 483, 648 ~ 2, 147, 483, 647											

### 6. 5. 6 環境レジスタ1 (RENV1)

#1 CCL, #2 CCL共に1組の環境レジスタがあり, 環境レジスタ1ではカウンタの動作条件を設定し, カウンタの起動・停止を行います.



REN V 1

23	22	21	20	19	18	17	16
1	1	0	ZLEDG	C2CLIN	C1CLIN	CLR-COND	

bit	機 能	設定値='0'	設定値='1'
18	XZ(ZZ)信号入力 カウンタクリア	有効	無効
19	YZ(UZ)信号入力 カウンタクリア		
上記2ビットはオプションポートと関連 不使用時は“無効(1)”とし、使用時は下記手順 ①REN V1:無効, ②オプションポート設定, ③REN V1:有効			
20	Z相(XZ, YZ, ZZ, UZ)信号入力 によるラッチのエッジ選択	設定値='0'	設定値='1'
	オプション ポート選択	同時ラッチ	任意(Don't Care)
		立上がり	立下がり

XZ,YZ(ZZ,UZ)クリア条件		
bit		内 容
17	16	
0	0	立上りエッジ
0	1	立下りエッジ
1	0	High レベル
1	1	Low レベル

(00:立上りエッジを推奨)

REN V 1

31	30	29	28	27	26	25	24
0	EVENT-ON CYCLE TIMER			CMP4-C	CMP3-C	CMP2-C	CMP1-C

比較カウンタ選択				
bit	機 能	設定値='0'	設定値='1'	
24	CMP 1	比較 カウンタ 選択	XCTR (ZCTR)	YCTR (UCTR)
25	CMP 2			
26	CMP 3			
27	CMP 4			

イベントタイマ出力時の処理									
bit			処理内容	bit			処理内容		
30	29	28		30	29	28			
0	0	1	CTR 1	ラッチ	0	1	0	CTR 1	ラッチ & クリア
0	1	1	CTR 2		1	0	0	CTR 2	
1	0	1	CTR1,CTR2		1	1	0	CTR1,CTR2	
0	0	0	処理なし		1	1	1		

表13. 環境レジスタ1 (REN V 1)

### 6. 5. 7 環境レジスタ2 (RENV2)

#1 CCL, #2 CCL共に1組の環境レジスタがあり、環境レジスタ2では各種コンパレータの比較条件の設定と、比較結果出力の設定等を行います。

7	6	5	4	3	2	1	0
CMPOUT2 COND SEL				CMPOUT1 COND SEL			

区分	bit				※CMP 1 と CMP 2 に設定する値は共通 (下表)	
CMP 1	3	2	1	0	CMP1 の比較方法	
CMP 2	7	6	5	4	CMP2 の比較方法	
	0	0	0	0	RCMP1 < [ CTR1/CTR2 ]	
	1	0	0	0	RCMP1 > [ CTR1/CTR2 ]	
	2	0	0	1	RCMP1 = [ CTR1/CTR2 ]	カウント方向に無関係
	3	0	0	1		カウントアップ時
	4	0	1	0		カウントダウン時
	5	0	1	0	RCMP2 < [ CTR1/CTR2 ]	
	6	0	1	1	RCMP2 > [ CTR1/CTR2 ]	
	7	0	1	1	RCMP2 = [ CTR1/CTR2 ]	カウント方向に無関係
	8	1	0	0		カウントアップ時
	9	1	0	0		カウントダウン時
	10	1	0	1	RCMP1 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP2	
	11	1	0	1	RCMP1 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP2	
	12	1	1	0	( RCMP1 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP2 ) AND ( RCMP3 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP4 )	
	13	1	1	0	( RCMP1 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP2 ) OR ( RCMP3 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP4 )	
	14	1	1	1	( RCMP1 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP2 ) AND ( RCMP3 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP4 )	
	15	1	1	1	( RCMP1 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP2 ) OR ( RCMP3 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP4 )	
※[ CTR1/CTR2 ]の表記は RENV1 CMP1・C(bit24)~CMP4・C(bit27) で決定されるカウンタの意味						

15	14	13	12	11	10	9	8
CMPOUT4 COND SEL				CMPOUT3 COND SEL			

区分	bit				※CMP 3 と CMP 4 に設定する値は共通 (下表)	
CMP 3	11	10	9	8	CMP3 の比較方法	
CMP 4	15	14	13	12	CMP4 の比較方法	
	0	0	0	0	RCMP3 < [ CTR1/CTR2 ]	
	1	0	0	0	RCMP3 > [ CTR1/CTR2 ]	
	2	0	0	1	RCMP3 = [ CTR1/CTR2 ]	カウント方向に無関係
	3	0	0	1		カウントアップ時
	4	0	1	0		カウントダウン時
	5	0	1	0	RCMP4 < [ CTR1/CTR2 ]	
	6	0	1	1	RCMP4 > [ CTR1/CTR2 ]	
	7	0	1	1	RCMP4 = [ CTR1/CTR2 ]	カウント方向に無関係
	8	1	0	0		カウントアップ時
	9	1	0	0		カウントダウン時
	10	1	0	1	RCMP3 < [ CTR1/CTR2 ] AND [ CTR1/CTR2 ] < RCMP4	
	11	1	0	1	RCMP3 > [ CTR1/CTR2 ] OR [ CTR1/CTR2 ] > RCMP4	
	12	1	1	0	CTR2 < CTR1	
	13	1	1	0	CTR2 > CTR1	
	14	1	1	1	CTR2 = CTR1	
	15	1	1	1	常に比較条件不成立	
※[ CTR1/CTR2 ]の表記は RENV1 CMP1・C(bit24)~CMP4・C(bit27) で決定されるカウンタの意味						

REN V 2

23	22	21	20	19	18	17	16
CMP3,4-P	CMP1,2 P	CMP2 EQ	CMP1 EQ	CMPOUT3,4-WIDTH		CMPOUT1,2-WIDTH	

bit	機 能	設定値=0	設定値=1
20	CMPOUT1	"="比較条件成立時	比較カウンタのクリア
21	CMPOUT2		
22	CMPOUT1,CMPOUT2	出力論理	負論理
23	CMPOUT3,CMPOUT4		

b17,16	CMPOUT1,2	パルス幅
b19,18	CMPOUT3,4	
0 0	レベル出力	
0 1	50 μs	ワンショット出力
1 0	1ms	
1 1	6.25ms	

※オプションポートも参照

REN V 2

31	30	29	28	27	26	25	24
0	0	FILTR ON	CMP1,2 MSK	0	0	0	0

bit	機 能	設定値='0'	設定値='1'
28	CMPOUT1,2 出力マスク	マスクしない	マスクする
29	xAx,xBx 入力信号フィルタ	フィルタON	フィルタOFF

表14. 環境レジスタ 2 (REN V 2)

## 6. 6 オプションポート

(1) 入力状況確認レジスタ・・z\_sts (Base+20) Read only

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	UCH Z	ZCH Z	ZUETM	0	YCH Z	XCH Z	XYETM

図15. 入力状況確認レジスタ

bit	記号	機 能	備 考
0	XYETM	XYイベントタイマ出力状態モニタ	R+32-4 でトグル設定
1	XCH Z	X c h Z相状態モニタ	
2	YCH Z	Y c h Z相状態モニタ	
4	ZUETM	ZUイベントタイマ出力状態モニタ	R+32-5 でトグル設定
5	ZCH Z	Z c h Z相状態モニタ	
6	UCH Z	U c h Z相状態モニタ	

表16. 入力状況確認レジスタ

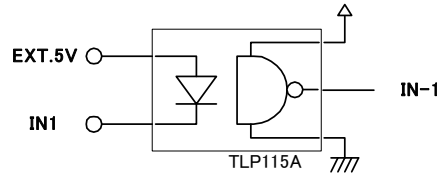
(2) 汎用入力状況確認レジスタ・・di\_reg (Base+28) Read only

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

図17. 汎用入力状況確認レジスタ (di\_reg)

bit	記号	機 能	備 考
0	IN-1	汎用 D-I/O 入力	ラッチ, 割り込みで使用
1~3	IN-2.4	汎用 D-I/O 入力	

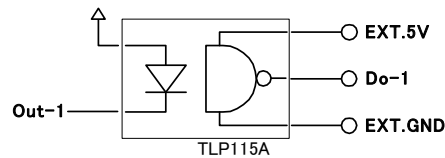
表18. 汎用入力状況確認レジスタの詳細



(3) 汎用出力レジスタ・・do\_reg (Base+2C) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												汎用 D-I/O 出力			
												Out-4	Out-3	Out-2	Out-1

図19. 汎用出力レジスタ



(4) CMP出力レジスタ・・sw\_and (Base+30) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		ZU ETM	ZU CP22	ZU CMP4	ZU CMP3	ZU CMP2	ZU CMP1			XY ETM	XY CP22	XY CMP4	XY CMP3	XY CMP2	XY CMP1

図20. CMP出力レジスタ

bit	記号	機 能	備 考
0	XY CMP1	XYout1 の出力条件 3種類 (CMP1~3) 選択 AND 条件	
1	XY CMP2		
2	XY CMP3		
3	XY CMP4	XYout2 の出力条件 3種類 (CMP2,4,イベントタイマ) 選択 AND 条件	
4	XY CMP2		
5	XY ETM		
8	ZU CMP1	ZUout1 の出力条件 3種類 (CMP1~3) 選択 AND 条件	
9	ZU CMP2		
10	ZU CMP3		
11	ZU CMP4	ZUout2 の出力条件 3種類 (CMP2,4,イベントタイマ) 選択 AND 条件	
12	ZU CP22		
13	ZU ETM		

表21. CMP出力レジスタの詳細

(5) ITMトグル出力レジスタ・・sw\_tgl (Base+32) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				ZO2	ZO1	XO2	Xo1			ZU ETM	XY ETM			IN-1 Not	Z Not

図22. ITMトグル出力レジスタ

bit	記号	機 能	備 考
0	Z Not	Z相入力 極性反転 (4ch)	
1	IN-1 Not	IN-1入力 極性反転	
4	XY ETM	XY イベントタイマ 出力切り替え 1:トグル (1/2周波数)	
5	ZU ETM	ZU イベントタイマ 出力切り替え 1:トグル (1/2周波数)	
8	XO1	XYOUT1 出力論理反転	
9	XO2	XYOUT2 出力論理反転	
10	ZO1	ZUOUT1 出力論理反転	
11	ZO2	ZUOUT2 出力論理反転	

表23. ITMトグル出力レジスタの詳細

(6) XYout2の出力パルス幅設定レジスタ・・pwm\_reg (Base+34) Read /Write 初期値 : 0000H

100nS~5mS (100nS ピッチ)

0000H 設定でスルー出力 (CCL の設定に依存)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
C15	C14	C13	C12	C11	C10	C09	C08	C07	C06	C05	C04	C03	C02	C01	C00

図24. 出力パルス幅レジスタ

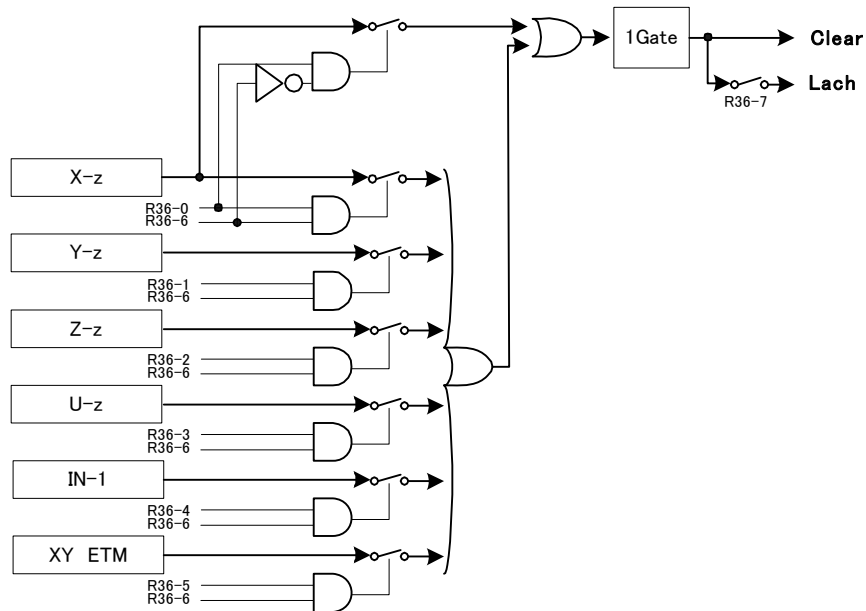
(7) Z相カウンタクリアレジスタ ・ sw\_clr( Base+36) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								LandC	ALL	Etm	IN-1	U-CH ZCLR	Z-CH ZCLR	Y-CH ZCLR	X-CH ZCLR

図25. Z相カウンタクリアレジスタ

bit	記号	機能	備考
0	X-CH ZCLR	X-CH Z相入力でカウンタクリア	CCL-REG でエッジ指定
1	Y-CH ZCLR	Y-CH Z相入力でカウンタクリア	CCL-REG でエッジ指定
2	Z-CH ZCLR	Z-CH Z相入力でカウンタクリア	CCL-REG でエッジ指定
3	U-CH ZCLR	U-CH Z相入力でカウンタクリア	CCL-REG でエッジ指定
4	IN-1	外部入力でクリア	B6=' 1' の時
5	etm	XY (CCL1) イベントタイマ入力でクリア	B6=' 1' の時
6	ALL	同時 クリア 0 : は各 CH 毎クリア	
7	LandC	ラッチ AND クリア	R38 (Lach と併用不可)

表26. Z相カウンタクリアレジスタの詳細



以下同様 (4CH分)

ラッチ AND クリア時のラッチは個別ラッチになります。  
クリアはカウンタ値±1の可能性あります。

(8) カウンタラッチ条件レジスタ ・ sw\_ltc( Base+38) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALL		U-CH z	Z-CH z	ZCMP 4	ZCMP 3	ZCMP 2	ZCMP 1	IN- 1	XY ETM	Y-CH z	X-CH z	XCMP 4	XCMP 3	XCMP 2	XCMP 1

図27. カウンタラッチ条件レジスタ

bit	記号	機 能	備 考
0~3	XCMP1..4	XY-CMP 条件でカウンタラッチ (他の条件と一致した場合)	
4	X-CH z	X-CH Z 相 入力でカウンタラッチ (他の条件と一致した場合)	
5	Y-CH z	Y-CH Z 相 入力でカウンタラッチ (他の条件と一致した場合)	
6	XY ETM	XY イベントタイマでカウンタラッチ (他の条件と一致した場合)	
7	IN-1	汎用入力 1 カウンタラッチ (他の条件と一致した場合)	
8~11	ZCMP1..4	ZU-CMP 条件でカウンタラッチ (他の条件と一致した場合)	
12	Z-CH z	Z-CH Z 相 入力でカウンタラッチ (他の条件と一致した場合)	
13	U-CH z	U-CH Z 相 入力でカウンタラッチ (他の条件と一致した場合)	
15	ALL	同一ラッチ/CH 個別ラッチ切り替え (1 : 個別)	

表28. カウンタラッチ条件レジスタの詳細

同一ラッチは60nsecのパルス幅

(9) 割り込み要因 (ソース) レジスタ ・ i\_src ( Base+50) Read /Write

I\_clr( Base+54) bits 対応 “1” 書込みでクリア

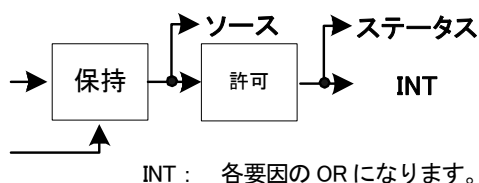
割り込みデータポーリング用のステータスレジスタ (内部ラッチします)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ZUout 2	ZUout 1	Xyout 2	Xyout 1	U-Z	Z-Z	Y-Z	X-Z					ICPLD	IN-1	ZU-CH	XY-CH

図29. 割り込み要因 (ソース) レジスタ

bit	記号	機 能	備 考
0	XY-CH	XY-CH (CCL #1) の割り込み	
1	ZU-CH	ZU-CH (CCL #2) の割り込み	
2	IN-1	IN-1 入力の割り込み	
3	ICPLD	入力 CPDL エラー (同相、断線)	+3e I-ints 参照
8~11	X~U - Z	Z 相立ち上がり	
12~15	XYZUout	OUT4 出力 出力	

表30. 割り込み要因 (ソース) レジスタの詳細



(10) 割込みマスク レジスタ・・I\_enab (Base+52) Read /Write 初期値：0000H  
 “1” で割り込み許可

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ZUout 2	ZUout 1	Xyout 2	Xyout 1	U-Z	Z-Z	Y-Z	X-Z					ICPLD	IN-1	ZU-CH	XY-CH

図31. 割込みマスク レジスタ

bit	記号	機 能	備 考
0	XY-CH	XY-CH (CCL #1) の割り込み	
1	ZU-CH	ZU-CH (CCL #2) の割り込み	
2	IN-1	IN-1 入力の割り込み	
3	ICPLD	入力 CPDL エラー (同相、断線)	+3e I-ints 参照
8~11	X~U - Z	Z 相立ち上がり	
12~15	XYZUout	OUT4 出力 出力	

表32. 割込みマスク レジスタの詳細

‘0’ でマスク 割り込み許可時は必ず、I\_clr (Base+54) の対応ビットに “1” 又は “FFFF” を書き込みしてください

(11) 割込みステータス レジスタ・・I\_sts/I\_clr (Base+54) Read Only

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ZUout 2	ZUout 1	Xyout 2	Xyout 1	U-Z	Z-Z	Y-Z	X-Z					ICPLD STS	IN-1 STS	ZU-ST S	XY-ST S

図33. 割込みステータス レジスタ

bit	記号	機 能	備 考
0	XY-CH	XY-CH (CCL #1) の割り込み状態	Read Only
1	ZU-CH	ZU-CH (CCL #2) の割り込み状態	Read Only
2	IN-1	IN-1 入力の割り込み状態	Read Only
3	ICPLD-STS	入力 CPDL エラー (同相、断線)	Read Only
8~11	X~U - Z	Z 相立ち上がり	Read Only
12~15	XYZUout	OUT4 出力 出力	Read Only

表34. 割込みステータス レジスタの詳細

(12) ボード ID レジスタ・・bd\_id (Base+60) Read Only

Board ID 読み出し

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												BID B3	BID B2	BID B1	BID B0

図35. ボード ID レジスタ

基板上のジャンパーポスト (ジャンパーショートで “0” が読めます)

(13) 入力フィルターレジスタ・・f\_infl (Base+3a) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						IN1 Inv	Z Inv	U-CH Ref	Z-CH Ref	Y-CH Ref	X-CH Ref	1shot U	1shot Y		

図36. 入力フィルターレジスタ

bit	記号	機 能	備 考
0			
2	1shot Y	波形計測モード 1 : 計測モード	R+4a 設定参照
3	1shot U	波形計測モード 1 : 計測モード	R+4a 設定参照
4	X-CH Ref	X-CH 波形再生 1 : OFF	
5	Y-CH Ref	Y-CH 波形再生 1 : OFF	
6	Z-CH Ref	Z-CH 波形再生 1 : OFF	
7	U-CH Ref	U-CH 波形再生 1 : OFF	
8	Z Inv	各 Z 相 インバート 1 : 入力反転	
9	IN1 Inv	汎用入力 1 インバート 1 : 入力反転	

表37. 入力フィルターレジスタの詳細

(14) 入力割込み許可レジスタ・・f\_inte (Base+3c) Read /Write 初期値 : 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						1Shot u	1Shot Y	U-CH ERR	Z-CH ERR	Y-CH ERR	X-CH ERR	U-CH FL	Z-CH FL	Y-CH FL	X-CH FL

図38. 入力割込みマスクレジスタ

bit	記号	機 能	備 考
0	X-CH FL	断線検出 1 : 断線 許可	
1	Y-CH FL	断線検出 1 : 断線 許可	
2	Z-CH FL	断線検出 1 : 断線 許可	
3	U-CH FL	断線検出 1 : 断線 許可	
4	X-CH ERR	X-CH 同相 ERR 1 : エラー 許可	
5	Y-CH ERR	Y-CH 同相 ERR 1 : エラー 許可	
6	Z-CH ERR	Z-CH 同相 ERR 1 : エラー 許可	
7	U-CH ERR	U-CH 同相 ERR 1 : エラー 許可	
8~9	1Shot Y, U	計測モード時、1 回取り込みの取り込み終了	R4a 設定参照

表39. 入力割込み許可レジスタの詳細

(15) 入力割込みクリア・ステータスレジスタ・・f\_ints/f\_intc (Base+3e) Read /Write

※読み込みでステータス、書き込みで要因クリア (ビット対応 : 1 : ビット書き込みでクリア)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						1Shot u	1Shot Y	U-CH ERR	Z-CH ERR	Y-CH ERR	X-CH ERR	U-CH FL	Z-CH FL	Y-CH FL	X-CH FL

図40. 入力割込みクリア・ステータスレジスタ

bit	記号	機 能	備 考
0~3	-CH FL	断線検出 1 : 断線	
4~7	-CH ERR	X-CH 同相 ERR 1 : エラー	
8~9	1Shot Y, U	計測モード時、1 回取り込みの取り込み終了	R4a 設定参照

表41. 入力割込み許可レジスタの詳細

(16) 配線状況確認レジスタ・・f\_cabs (Base+40) Read only

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								Z-CH B	Z-CH A	Z-CH B	Z-CH A	Y-CH B	Y-CH A	X-CH B	X-CH A

図42. 配線状況確認レジスタ

bit	記号	機 能	備 考
0	X-CH A	X-CH A相 結線状態 モニタ 1:断線	
1	X-CH B	X-CH B相 結線状態 モニタ 1:断線	
2	Y-CH A	Y-CH A相 結線状態 モニタ 1:断線	
3	Y-CH B	Y-CH B相 結線状態 モニタ 1:断線	
4	Z-CH A	Z-CH A相 結線状態 モニタ 1:断線	
5	Z-CH B	Z-CH B相 結線状態 モニタ 1:断線	
6	U-CH A	U-CH A相 結線状態 モニタ 1:断線	
7	U-CH B	U-CH B相 結線状態 モニタ 1:断線	

表43. 配線状況確認レジスタの詳細

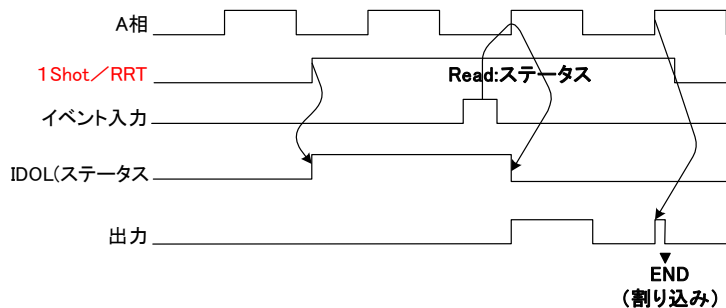
(17) 計測モード設定レジスタ・・f\_1sht (Base+4a) Read /Write 初期値: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RPT U	IN1	U-Z	Y-Z	X-Z	ETM ZU	IDOL U	Shot U	RPT Y	IN1	U-Z	Y-Z	X-Z	ETM XY	IDOL Y	Shot Y

図44. 計測モード設定レジスタ

bit	記号	機 能	備 考
0 / 8	Shot	1周期取り込み A相の立ち上がりから B02~06 がすべて '0' 時 Soft 同期 読み込み時 : IDOL+Busy ステータス	B02~06 で '1' が有場合 イベント同期
1 / 9	IDOL	イベント待ちステータス	Read Only
2 / 10	ETM	イベントタイマ立ち上がりでスタート	同期式
3-5 / 11-13	X,Y,U-Z	Z相立ち上がりでスタート	同期式
6 / 14	IN1	Zuout2 立ち上がりでスタート (Uも測定時は設定に 気をつけてください)	同期式
7 / 15	RPT	イベント同期時の連続測定モード	

表45. 計測モード設定レジスタの詳細



## 7. 基本的な設定と運用

### 7. 1 操作手順

操作手順は以下の流れで行います。

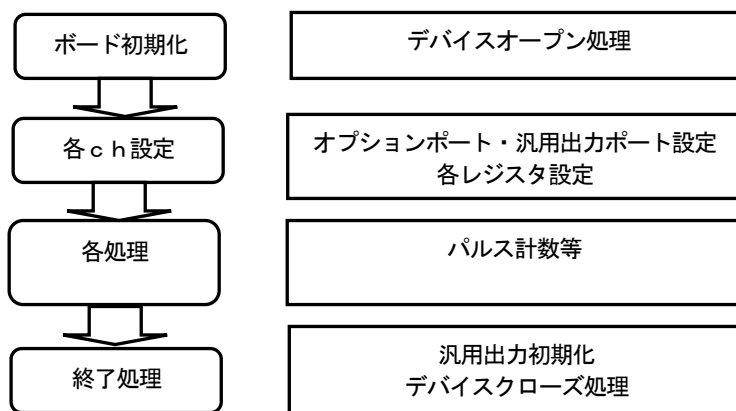


図46. 基本的な操作手順

### 7. 2 エンコーダ等パルス計数（カウント）

以下に基本的なカウントの手順を記します。

- ①使用するカウンタのカウントを停止  
環境レジスタ 1 (RENV1) の bit12=1 (CTR1), bit13=1 (CTR2) 設定.
- ②使用するカウンタのカウントが停止していることを確認  
ステータスの bit4=1 (CTR1 停止中), bit5=1 (CTR2 停止中) を確認.
- ③使用するカウンタの入力信号形式を選択  
環境レジスタ 1 (RENV1) の bit0~bit2 (CTR1), bit3~bit5 (CTR2) で設定.
- ④使用するカウンタをクリア  
単独コマンド CLRCTR1 (0001h), CLRCTR2 (0002h) 書込みで, 使用するカウンタを 0 にする.  
あらかじめオフセットするならば, 使用する CTR レジスタにオフセット値を書込む.
- ⑤使用するカウンタをスタート  
環境レジスタ 1 (RENV1) の bit12=0 (CTR1), bit13=0 (CTR2) 設定.
- ⑥使用するカウンタレジスタ読み

#### 7. 2. 1 カウントパルスの入力信号形式

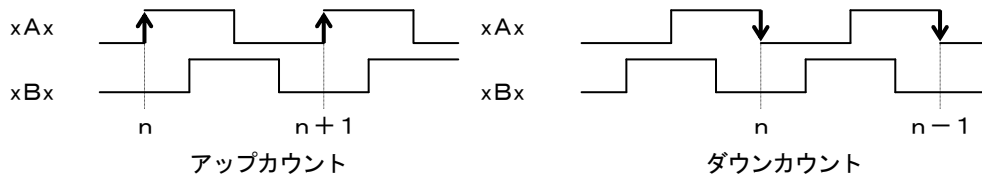
環境レジスタ 1 (RENV1) の bit 0~5 を設定します。

区分	ビット			カウントパルスの入力信号形式を選択	
ENC1IN	2	1	0	XAx, XBx (ZAx, ZBx)	
ENC2IN	5	4	3	YAx, YBx (UAx, UBx)	
	0	0	0	90度 位相差信号	1 逓倍
	0	0	1		2 逓倍
	0	1	0		4 逓倍
	0	1	1	アップ/ダウ ン パルス	負論理
	1	0	0		正論理
	1	0	1	共通パルス と方向信号	負論理
	1	1	0		正論理
	1	1	1	設定不可	

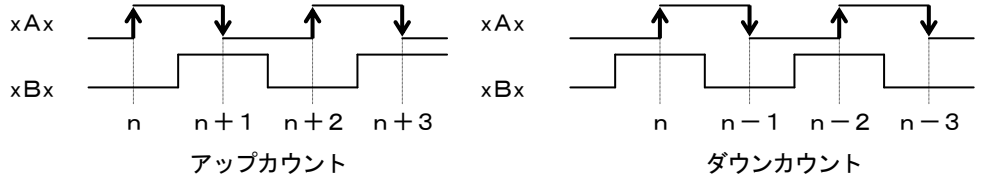
表47. カウントパルスの入力信号形式

各入力信号形式におけるカウント動作時のタイミングは、次頁の通りです。

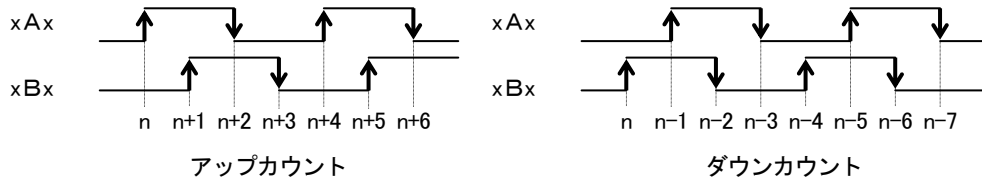
■ 90度位相差信号1通倍



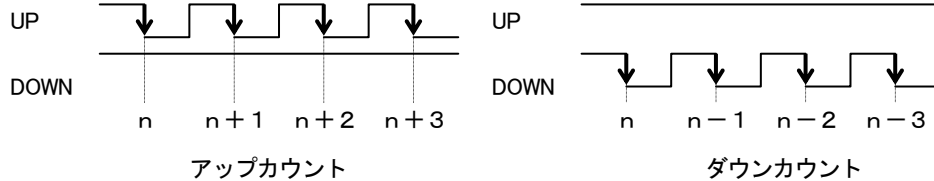
■ 90度位相差信号2通倍



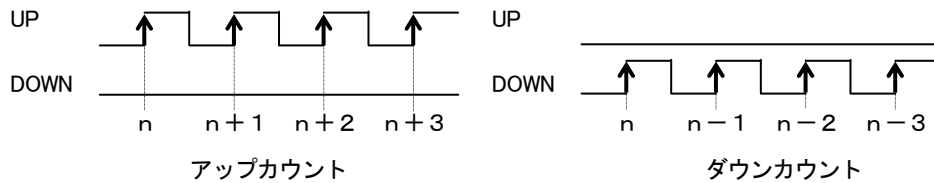
■ 90度位相差信号4通倍



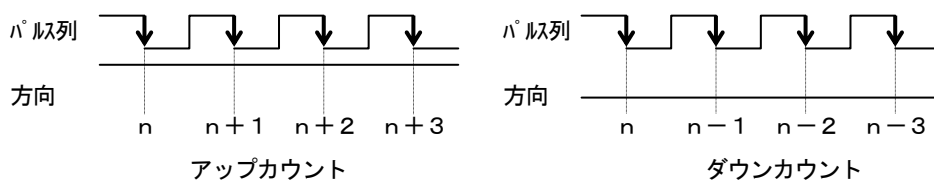
■ アップ/ダウンパルス負論理



■ アップ/ダウンパルス正論理



■ 共通パルスと方向信号負論理



■ 共通パルスと方向信号正論理

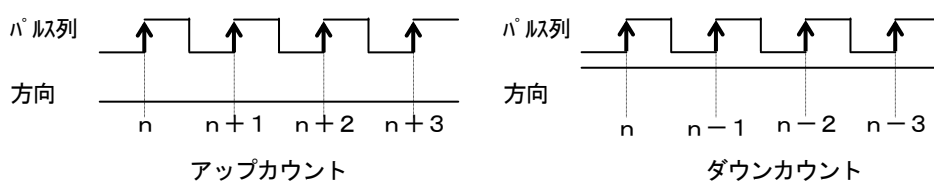


図48. 各入力信号形式におけるカウント動作のタイミング

## 8. コネクタ信号表

Pin No.	信号名	2相パル入力時	UP/DOWNパルス入力時	パルス列/方向入力時	Pin No.	信号名	2相パル入力時	UP/DOWNパルス入力時	パルス列/方向入力時
1	XAP	A相(差動+)	UP(差動+)	パルス(差動+)	14	XAN	A相(差動-)	UP(差動-)	パルス(差動-)
2	GND	GND			15	XBP	B相(差動+)	DOWN(差動+)	方向(差動+)
3	XBN	B相(差動-)	DOWN(差動-)	方向(差動-)	16	GND	GND		
4	YAP	A相(差動+)	UP(差動+)	パルス(差動+)	17	YAN	A相(差動-)	UP(差動-)	パルス(差動-)
5	GND	GND			18	YBP	B相(差動+)	DOWN(差動+)	方向(差動+)
6	YBN	B相(差動-)	DOWN(差動-)	方向(差動-)	19	GND	GND		
7	Vcc	補助(アイルレション無し+5V 100mA) 内部 SW			20	Vcc	補助(アイルレション無し+5V 100mA) 内部 SW		
8	ZAP	A相(差動+)	UP(差動+)	パルス(差動+)	21	ZAN	A相(差動-)	UP(差動-)	パルス(差動-)
9	GND	GND			22	ZBP	B相(差動+)	DOWN(差動+)	方向(差動+)
10	ZBN	B相(差動-)	DOWN(差動-)	方向(差動-)	23	GND	GND		
11	UAP	A相(差動+)	UP(差動+)	パルス(差動+)	24	UAN	A相(差動-)	UP(差動-)	パルス(差動-)
12	GND	GND			25	UBP	B相(差動+)	DOWN(差動+)	方向(差動+)
13	UBN	B相(差動-)	DOWN(差動-)	方向(差動-)	26	GND	GND		

使用コネクタ (MDR - 10226-52A2JL (住友3M))

表49. J1コネクタ信号表

Pin	信号名	内容	Pin	信号名	内容
1	Vcc	補助(アイルレション無し+5V 100mA) 内部 SW	19	Vcc	補助(アイルレション無し+5V 100mA) 内部 SW
2	XZP	X chのZ相入力(差動+)	20	ZZP	Z chのZ相入力(差動+)
3	XZN	X chのZ相入力(差動-)	21	ZZN	Z chのZ相入力(差動-)
4	GND	GND	22	GND	GND
5	YZP	Y chのZ相入力(差動+)	23	UZP	U chのZ相入力(差動+)
6	YZN	Y chのZ相入力(差動-)	24	UZN	U chのZ相入力(差動-)
7	GND	GND	25	GND	GND
8	Ext.5V	高速カブラ用5V入力	26	Ext.5V	高速カブラ用5V入力
9	IN1	汎用入力1	27	IN3	汎用入力3
10	IN2	汎用入力2	28	IN4	汎用入力4
11	Ext.5V	高速カブラ用5V入力	29	Ext.5V	高速カブラ用5V入力
12	XYO1	CMPOUT1(一致出力)高速カブラIC出力	30	ZUO1	CMPOUT1(一致出力)の高速カブラIC出力
13	XYO2	CMPOUT2(一致出力)高速カブラIC出力	31	ZUO2	CMPOUT2(一致出力)の高速カブラIC出力
14	Ext.GND	高速カブラ用GND入力	32	Ext.GND	高速カブラ用GND入力
15	Do-1	汎用出力1 高速カブラIC出力	33	Do-3	汎用出力3 高速カブラIC出力
16	Do-2	汎用出力2 高速カブラIC出力	34	Do-4	汎用出力4 高速カブラIC出力
17	XYETM	XYETM(イベントタイマ)高速カブラIC出力	35	ZUETM	ZUETM(イベントタイマ)の高速カブラIC出力
18	Ext.GND	高速カブラ用GND入力	36	Ext.GND	高速カブラ用GND入力

使用コネクタ (MDR - 10236-52A2JL (住友3M))

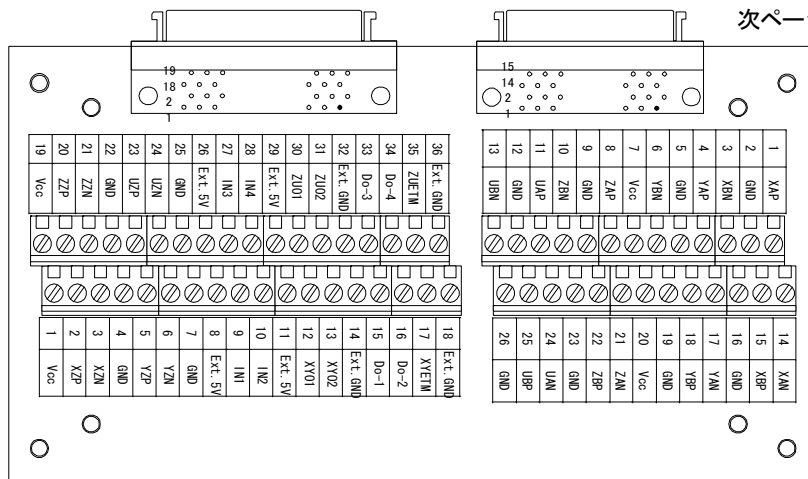
Vcc 補助(アイルレション無し+5V 100mA)は補助電源です。使用する場合、ノイズ環境では誤動作の原因になります。

表50. J2コネクタ信号表

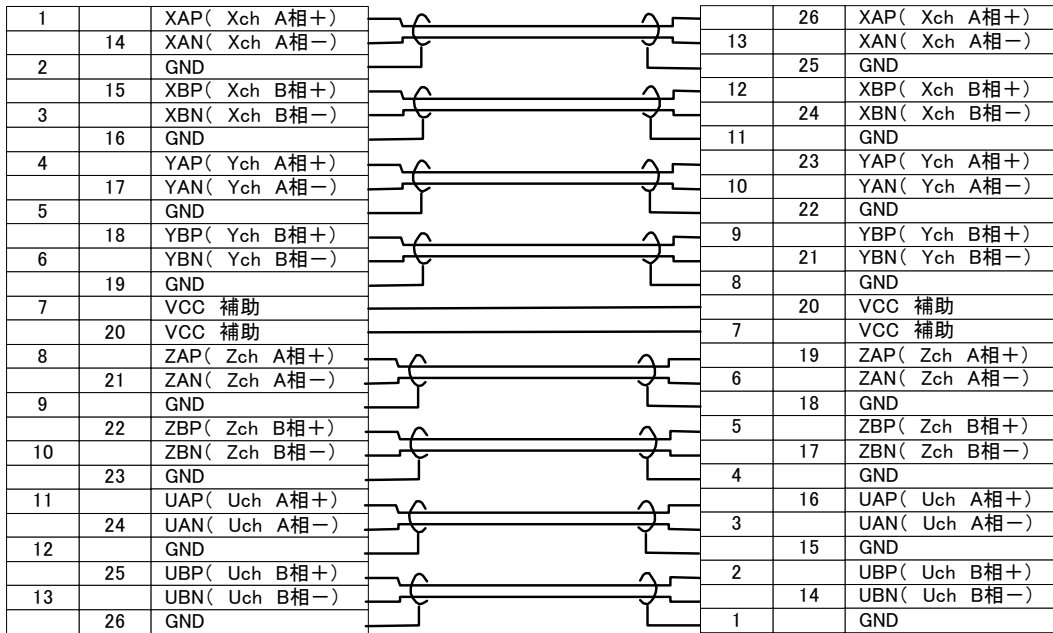
- 指定ケーブルを使用した場合にACBボード端子配列は下記になります。

(HCL-054W (HCL-053W)は標準ツインナックスケーブルを利用しているため、1-26が逆になります)

次ページに配線図を参照ください



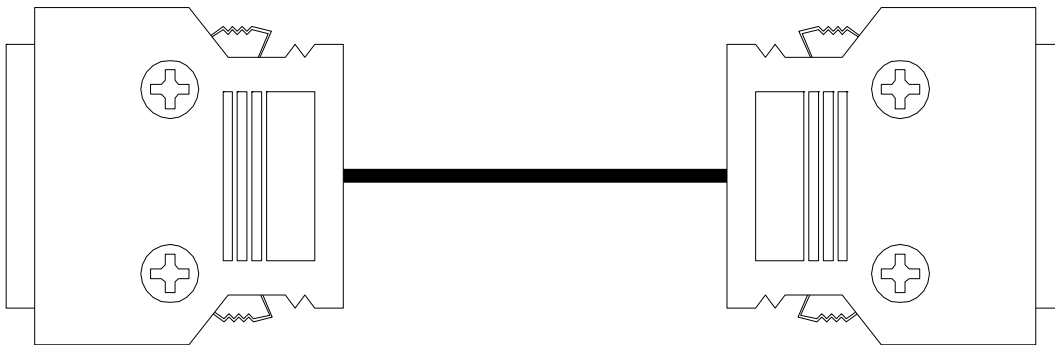
参考資料) HCL-054W 結線図



3M 10126-6000EL  
10326-A200-00

3M 14526EZ5B-500-02

3M 10126-6000EL  
10326-A200-00



■ 36芯側(J2側) HCL-052Wは1対1です。

## 9. 外部との接続

### 9. 1 カウンタ入力及Z相入力信号

項	項目	内容
1	XCTR,YCTR,ZCTR,UCTR XZ相入力,YZ相入力 ZZ相入力,UZ相入力 各入力回路	<p>終端インパ P1,P2,P3</p> <p>26LS32相当</p>
2	差動接続	

表51. カウンタ入力及びZ相入力信号

### 9. 2 汎用入力

項	項目	内容
1	IN1~IN4 入力回路	<p>TLP115A</p>
2	接点接続	<p>5V</p>

表52. 汎用入力

### 9. 3 汎用出力及び一致出力

項	項目	内容
1	OUT1~OUT4 XYOUT1,XYOUT2 ZUOUT1,ZUOUT2 XYETM,ZUETN 出力回路	<p>TLP115A</p>
2	接続回路	<p>5V</p> <p>負荷</p>

表53. 汎用出力及び一致出力

# 10. 基板寸法図

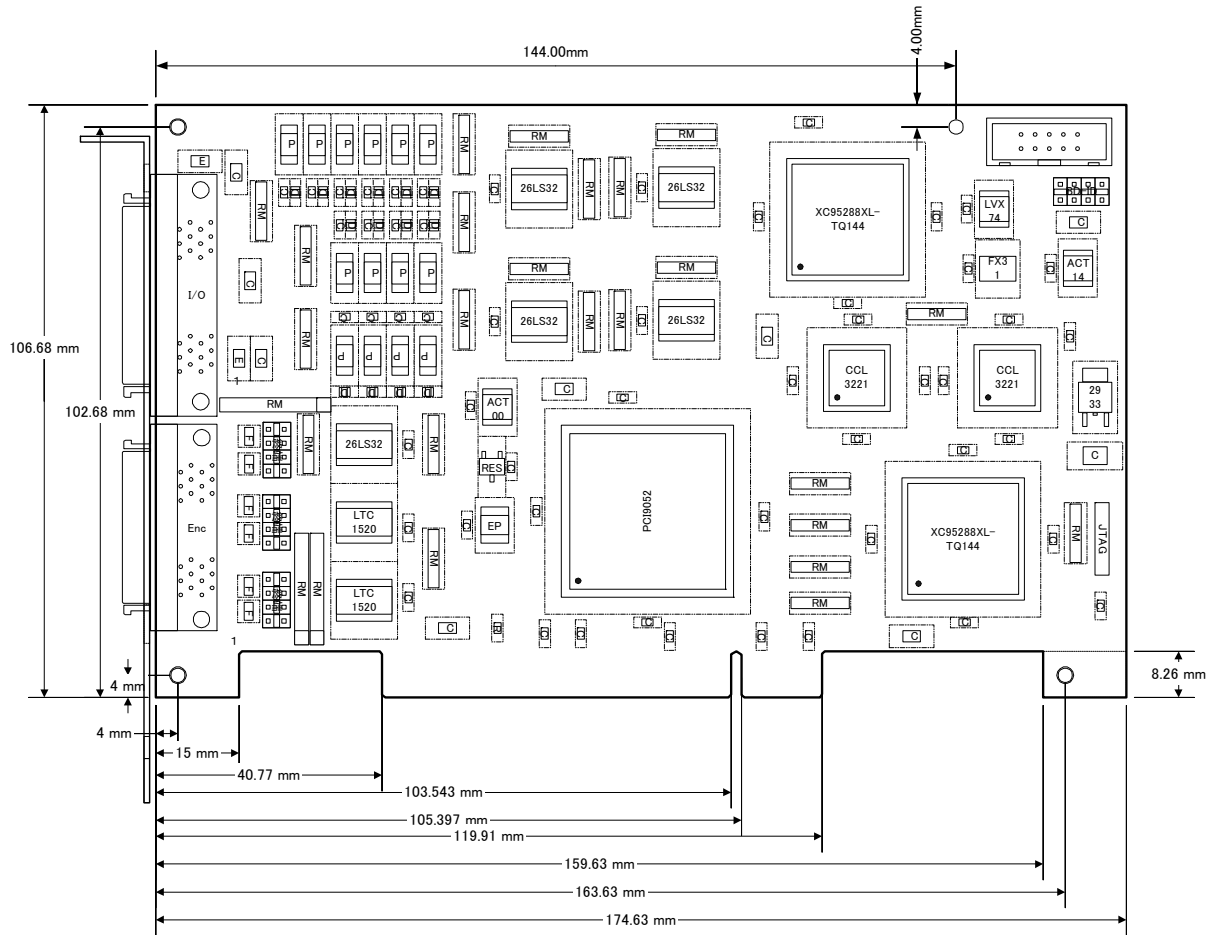


図54. 基板寸法図

差動入力部のケーブルは住友3M社製の「MDR ツインナックスケーブル： 14526-EZ5B-◇◇◇-02」が使用できます。(◇は長さ 050 (0.5m) ~A00 (10.0m) )

## 11. ソフトウェア編

この章ではソフトウェアについて説明をします。

対応するOSとして次の種類があります。

- WindowsXP Professional/Home Edition ・以降WinXPと記します
- Windows2000 . . . . .以降Win2Kと記します
- WindowsNT4.0 . . . . .以降WinNTと記します
- Windows98 . . . . .以降Win98と記します

デバイスドライバのI/F用ライブラリとしてデバイスドライバI/F用DLLが用意されています。  
このDLL関数をドライバ関数と称します。

デバイスドライバは、個々のOS毎にあります。

デバイスドライバ関数はCTRボードの各ポートアドレスへの入出力を制御します。

ファイル名を以下に示します。

- hictr540.dll . . . ドライバI/F用DLL
- hictr540.lib . . . C言語用 インポートライブラリ
- hictr540.h . . . C言語用関数結合用ヘッダー
- hictr540.bas . . . VB用関数定義標準モジュール

※ INTEL互換のCPUを搭載したマシン用です。その他のプラットフォームには対応していません。

No	関 数 名	機 能
1	ct540_GetDeviceCount()	ボード枚数の取得
2	ct540_GetDeviceInfo()	デバイス情報の取得
3	ct540_OpenDevice()	デバイスのオープン
4	ct540_CloseDevice()	デバイスのクローズ
5	ct540_rXYSts()	X Y c hステータスの読込
	ct540_rZUSts()	Z U c hステータスの読込
6	ct540_wXYCmd()	X Y c h制御コマンド書込
	ct540_wZUCmd()	Z U c h制御コマンド書込
7	ct540_rXYReg()	X Y c hレジスタの読込
	ct540_rZUReg()	Z U c hレジスタの読込
	ct540_wXYReg()	X Y c hレジスタの書込
	ct540_wZUReg()	Z U c hレジスタの書込
8	ct540_rPortB()	オプションポートのバイト読込
	ct540_rPortW()	オプションポートのワード読込
	ct540_wPortB()	オプションポートへバイト書込
	ct540_wPortW()	オプションポートへワード書込
9	ct540_rXYBuf()	X Y c h入出力バッファの読込
	ct540_rZUBuf()	Z U c h入出力バッファの読込
	ct540_wXYBuf()	X Y c h入出力バッファの書込
	ct540_wZUBuf()	Z U c h入出力バッファの書込

表55. ドライバ関数一覧

## 11.1 ソフトウェアの概要

弊社の提供するソフトウェアは、ドライバ関数、デバイスドライバです。  
アプリケーションプログラムと、これらのソフトウェアの関連は次図の通りです。

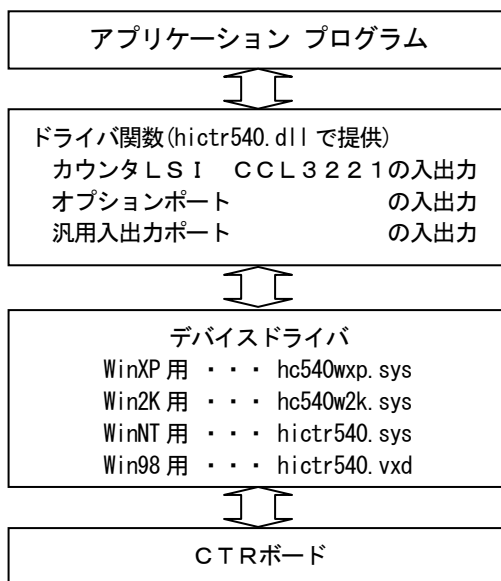


図56. ソフトウェアの関連

## 11.2 Windows版のインストールとアンインストール

### (1) Windows 7/Vista (32bit) へのインストール

- ① CTRボードをパソコンのPCIバススロットに装着する前に、パソコンの電源をONにしてWindowsを起動します。
- ② E:\¥win7\_x86¥dpinst.exe (CDドライブがEドライブである場合) を起動します。
- ③ “dpinst.exe” が起動されたら「次へ」をクリックして続行します。  
「ドライバーソフトウェアの発行元を検証できません」とのメッセージが出る場合がありますが、「このドライバーソフトウェアをインストールします」をクリックします。
- ④ インストーラー完了後、パソコンの電源をOFFし、CTRをパソコンのPCIバススロットに装着します。
- ⑤ パソコンの電源をONにしてWindowsを起動します。
- ⑥ デバイスのインストールが自動的に行われ、再起動を促されますので再起動してインストールが完了します。

### (2) Windows 7/Vista (64bit) へのインストール

- ① CTRボードをパソコンのPCIバススロットに装着する前に、パソコンの電源をONにしてWindowsを起動します。
- ② E:\¥win7\_x64¥dpinst.exe (CDドライブがEドライブである場合) を起動します。
- ③ “dpinst.exe” が起動されたら「次へ」をクリックして続行します。  
「ドライバーソフトウェアの発行元を検証できません」とのメッセージが出る場合がありますが、「このドライバーソフトウェアをインストールします」をクリックします。
- ④ インストーラー完了後、パソコンの電源をOFFし、CTRをパソコンのPCIバススロットに装着します。
- ⑤ パソコンの電源をONにしてWindowsを起動します。
- ⑥ デバイスのインストールが自動的に行われ、再起動を促されますので再起動してインストールが完了します。

### (3) Windows XPへのインストール

- ① パソコンの電源がOFFであることを確認した後、CTRボードをパソコンのPCIバススロットに装着し、パソコンの電源をONにしてWindowsを起動します。
- ② WinXPが起動すると、CTRボードがシステムにより検出され、自動的に必要なデバイスドライバのインストール画面が表示されます。添付ディスクをドライブに挿入します。
- ③ ソフトウェアを自動的にインストールする(推奨)をチェックします。
- ④ Hivertec HPCI-CTR544F(WinXP)を選択します。
- ⑤ 「Windows 007に合格していません」との警告が表示されますが、WinXPでの動作は弊社で確認していますので、**続行**を選択してインストールを続けてください。  
後はシステムの指示に従ってインストールを完了させます。

### (4) Windows 2000へのインストール

- ① パソコンの電源がOFFであることを確認した後、CTRボードをパソコンのPCIバススロットに装着します。パソコンの電源をONにしてWindowsを起動します。
- ② Win2Kが起動すると、CTRボードがシステムにより検出され、自動的に必要なデバイスドライバのインストール画面が表示されます。
- ③ システムがインストール元ディレクトリの指定を要求してきたら、添付ディスクをドライブに挿入します。
- ④ 「場所を指定」のチェックボックスをチェックします。
- ⑤ E:\¥WIN2K (Eドライブである場合) を指定して下さい。  
後はシステムの指示に従ってインストールを完了させます。

### (5) Windows NT4.0へのインストール

#### ■デバイスドライバのインストール

- ① 添付ディスクをドライブに挿入します。  
NTエクスペローラを起動し、E:\¥WinNT¥c540inst.inf (Eドライブである場合) を選択します。
- ② 次にマウスの右ボタンをクリックします。表示されるポップアップメニューから「インストール」を選択します。この操作によりデバイスドライバのインストールが開始されます。後はシステムの指示に従ってインストールを完了させます。  
コマンドプロンプトから、E:\¥WinNT¥c540inst.bat を実行させても同様にインストールが開始されます。

## (6) Windows 98へのインストール

- ① パソコンの電源がOFFであることを確認した後、CTRボードをパソコンのPCIバススロットに装着します。パソコンの電源をONにしてWindowsを起動します。
- ② Win98が起動すると、CTRボードがシステムにより検出され、自動的に必要なデバイスドライバのインストール画面が表示されます。
- ③ システムがインストール元ディレクトリの指定を要求してきたら、添付ディスクをドライブに挿入します。
- ④  検索場所の指定のチェックボックスをチェックします。
- ⑤ E:¥win9x (Eドライブである場合)を指定して下さい。  
後はシステムの指示に従ってインストールを完了させます。

## (7) Windows XP/2000/NT4.0/98 デバイスドライバのアンインストール

- ① 添付ディスクをドライブに挿入します。
- ② エクスプローラを起動し、E:¥ct540uin.exe (Eドライブである場合)を実行します。  
または、コマンドプロンプトから、E:¥ct540uin.exe (Eドライブである場合)を実行します。

## (8) Windows 7/Vista デバイスドライバのアンインストール

Windowsの「スタート」→「コントロールパネル」→「プログラムのアンインストール」  
→ [Windowsドライバ パッケージ Hivertec HPCI-CTR544F]を右クリックし  
アンインストールを行います。

## 11.3 準備

ドライバ関数では複数のCTRボードを制御することができます。  
あるCTRボードを制御するために、まずこのデバイスをオープンして、デバイスハンドル値を取得します。  
デバイスをオープンするためにはオープンするデバイスのデバイス情報(ハードウェアリソース情報)が必要となります。(ハードウェアリソース → ボードアドレス、ボードID等)

### (1) HPCI-CTR544Fのボード認識用のデータ構造体

#### ■ Windows版ボード(デバイス)認識用のデータ構造体

ボード認識のために次に示す HCTRDEVINF 型構造体を、ボード枚数最大16枚として、使用枚数分用意します。

[ C言語 : Visual C++ ]

```
typedef struct _HPCDEVICEINFO {
    DWORD nBusNumber; /* バス番号 */
    DWORD nDeviceNumber; /* デバイス番号*/
    DWORD dwIoPortAddress; /* I/O ポートアドレス */
    DWORD dwIrqNo; /* IRQ 番号 */
    DWORD dwNumber; /* 管理番号 */
    DWORD dwBoardID; /* ボードID(0~15) */
} HPCDEVICEINFO, *PHPCDEVICEINFO
```

[ Visual Basic ]

```
Public Type HPCDEVICEINFO
    nBusNumber As Long 'バス番号
    nDeviceNumber As Long 'デバイス番号
    dwIoPortAddress As Long 'I/O ポートアドレス
    dwIrqNo As Long 'IRQ 番号
    dwNumber As Long '管理番号
    dwBoardID As Long 'ボードID(0~15)
End Type
```

(注) 1. 管理番号はWindows 98 では使用されません。  
常に「INVALID\_HPC\_NUMBER (-1/VB, 0xffffffff/C言語)」が格納されています。

## (2) アプリケーションの構築

### [ Visual C++ (5.0 以上) によるアプリケーションの構築 ]

次のファイルをプロジェクトへ追加して下さい。

#### ■プロジェクト追加ファイル

・hict540.lib   ・・・ドライバ I / F 用 DLL インポートライブラリ

#### ■インクルードファイル

・hict540.h   ・・・ドライバ I / F 用 DLL 関数結合用ヘッダーファイル

- (注) 1. ドライバ I / F 用 DLL 関数は C 言語で作成されています。  
2. ドライバ I / F 用 DLL 関数のプロトタイプ宣言は次のように記述されています。

```
-----  
//  
//      関数プロトタイプ宣言  
//-----  
#ifdef __cplusplus  
extern "C"  
{  
#endif  
#ifdef __cplusplus  
    !-----  
    !      関数のプロトタイプ宣言  
    !-----  
#endif  
#ifdef __cplusplus  
}  
#endif
```

これは、アプリケーションを C++ コーディング (ファイル拡張子 = cpp) で作成する場合に備えての処理です。

3. 「#ifdef \_\_cplusplus」の定義は「Visual C++」用です。  
この言語で使用する場合には、明示的な宣言に変更できます。

```
-----  
(例)  
#define CPLUS 1                   "1"でC++用 (ファイル拡張子: cpp)  
.....                   "0"でC用 (ファイル拡張子: c)  
#if CPLUS  
.....  
#endif
```

### [ Visual Basic (5.0/6.0) によるアプリケーションの構築 ]

次のファイルをプロジェクトへ追加して下さい。

#### ■ドライバ用

・hict540.bas   ・・・ドライバ I / F 用 DLL 関数定義標準モジュールファイル

このファイルに外部関数宣言 (Declare 宣言)、及びユーザー定義型宣言が記述されています。

- (注) 1. ドライバ I / F 用 LIB 関数は C 言語で作成されています。  
2. アプリケーション作成メモリモデルと同一のドライバ関数メモリモデルを使用します。  
3. 割込処理を行う場合は、ラージモデルとして下さい。

## 11.4 ドライバ関数の戻り値

ドライバの諸関数を使用する時、関数の戻り値が異常値（'0'以外）であった場合には、異常内容に対応した処理を行います。

No	戻り値			異常内容と確認項目
	記号表記	16進数表記		
		C言語	VB	
1	NO_ERROR	0x000	&H0	正 常 異常は発生していません
2	NOT_FOUND	0x001	&H1	デバイスが見つからない ◎デバイスドライバがインストールされていない ◎デバイスドライバが所定のフォルダに格納されていない ◎CTRボードがPCに挿入されていない
3	ALREADY_OPENED	0x002	&H2	既にオープン済みのデバイスをオープン ◎オープン済みデバイスに更にオープン指令 ◇オープンしたデバイスはクローズするまで使用 （多重のオープンは禁止） ◎ボード2枚以上を使用する場合、オープンするデバイス情報を確認します。
4	NOT_MEMORY	0x004	&H4	デバイス情報格納メモリが不足 ◎アプリケーション用のメモリ不足 ◇パソコン主記憶メモリの不足 ◎システムリソース（OS用メモリ）の不足 ◇多数のアプリケーション起動 ◇1度に多数のウィンドウを開いた
5	INVALID_HANDLE	0x008	&H8	無効なデバイスハンドルを指定 ◎デバイスオープンで得られた“デバイスハンドル”の不使用 ◎このデバイスは既にクローズされている
6	NOT_READY	0x010	&H10	デバイスの入出力ポートが使用できない ◎システムが不安定になっている可能性がありますので、 弊社サポートまでお問い合わせください
7	ILLEGAL_DEVICE	0x020	&H20	ボード固有情報が不正 ◎システムが不安定になっている可能性がありますので、 弊社サポートまでお問い合わせください
8	ILLEGAL_ADDRESS	0x040	&H40	不正なベースアドレス（HPCボード） ◎指定したベースアドレスの確認
9	ILLEGAL_ACCESS	0x080	&H80	読込/書込み中の軸への読込/書込指令（DOS版） ◎多重入出力処理の指令 ボード上LSIへの読込/書込み中に同一ポートへの読込/書込指令が行われた。 この結果としてLSIの動作が保証されなくなり、読込/書込指令を禁止した。 ◇マルチタスク処理では先行タスクの処理終了まで待つ。 ◇割込処理モジュール内では、この処理を“待処理”とし、 割込処理を終了させる。
10	ILLEGAL_PARAM	0x100	&H100	関数の引数の値が異常

表57. ドライバ関数の戻り値

## 11.5 ドライバ関数詳細

関数説明文中で引数のデータ型、及び16進数の表記はC言語記述で行っています。  
Visual Basicでご利用の場合には、データ型を次のようにして下さい。  
また、C言語16進数表記を次のように読み替えて下さい。

言語区分		C言語	Visual Basic
データ型	Windows版	BYTE	Byte
		WORD	Integer
		DWORD	Long

表58. ドライバ関数のデータ型

言語区分	C言語	Visual Basic
16進数表記	0x0000	&H0
	0x1000	&H1000
	0xffffffff	-1

表59. 各言語の数値表記

## 11. 5. 1 Windows 版ドライバ関数

## (1) ct540\_GetDeviceCount( ) ボード枚数の取得

## 《機能》

現在パソコンに装着されている C T R ボードの枚数を取得します。(HPCI-CTR524F/522F のみ)

## 《書式》

## [ C 言語 ]

```
DWORD WINAPI ct540_GetDeviceCount (DWORD*   HpcDevNumber);
```

## [ Visual Basic ]

```
Declare Function ct540_GetDeviceCount Lib "hict540.dll" (ByRef HpcDevNumber As Long) As Long
```

## 《引数》

◆ DWORD\* *HpcDevNumber* ・ ・ C T R ボードの枚数

## 《戻り値》 処理結果

0 : 成功

0 以外 : 失敗 ・ ・ 「10. 4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C 言語 ]

```
DWORD count; // C T R ボードの枚数
```

```
DWORD ret; // 関数の戻り値
```

```
ret = ct540_GetDeviceCount( &count );
```

## [ Visual Basic ]

```
Dim count As Long ' C T R ボードの枚数
```

```
Dim ret As Long ' 関数の戻り値
```

```
ret = ct540_GetDeviceCount ( count )
```

## (2) ct540\_GetDeviceInfo( ) デバイス情報の取得

### 《機能》

現在パソコンに装着されているCTRボードのデバイス情報(※1)を取得します。この結果、HPCDEVICEINFO型(※2)の配列にデバイス情報が格納されます。この値は、デバイスオープン時に利用します。(HPCI-CTR524F/522Fのみ)

### 《書式》

#### [ C言語 ]

```
DWORD WINAPI ct540_GetDeviceInfo( DWORD* HpcDevNumber, HPCDEVICEINFO* HpcDevInfo );
```

#### [ Visual Basic ]

```
Declare Function ct540_GetDeviceInfo Lib "hict540.dll" (ByRef HpcDevNumber As Long, _  
HpcDevInfo As HPCDEVICEINFO) As Long
```

### 《引数》

- ◆ DWORD\* *HpcDevNumber* ・ ・ 情報を取得するボードの最大枚数が格納された DWORD型エリアのアドレスを渡します。  
関数の呼び出し後、実際に情報を取得したボードの枚数が格納されます。
- ◆ HPCDEVICEINFO *HpcDevInfo* ・ ・ 各ボードのデバイス情報がセットされるべきエリアのアドレス、すなわちHPCDEVICEINFO型の配列の先頭アドレスを渡します。

### 《戻り値》 処理結果

0 : 成功  
0以外 : 失敗 ・ ・ 「10.4 ドライバ関数の戻り値」を参照して下さい。

### 《呼び出し例》 パソコンにCTRボードが2枚装着されていることを想定します。

#### [ C言語 ]

```
DWORD ret; //関数の戻り値  
DWORD count = 2; //最大枚数は2  
HPCDEVICEINFO HpcDevInfo[2]; //2枚のCTRボードのデバイス情報がセットされるべきエリア
```

```
ret = ct540_GetDeviceInfo( &count, //countのアドレスを渡す。  
&HpcDevInfo[0] ); //配列の先頭アドレスを渡す。
```

#### [ Visual Basic ]

```
Dim ret As Long ' 関数の戻り値  
Dim count As Long ' 枚数  
Dim HpcDevInfo(2) As HPCDEVICEINFO ' デバイス情報のエリア
```

```
count = 2 ' 最大枚数は2枚
```

```
ret = ct540_GetDeviceInfo( count, _ ' countのアドレスを渡す。  
HpcDevInfo(0) ) ' 配列の先頭アドレスを渡す。
```

※1, ※2. デバイス情報格納の構造体名及び構造体メンバは、ドライバの種類により異なる場合があります。

### (3) ct540\_OpenDevice( ) デバイスのオープン

#### 《機能》

渡したデバイス情報を持つCTRボードをオープンし、他のCTRボードと識別するためのデバイスハンドルを取得します。以降このデバイスハンドルは、このCTRボードにアクセスするためのハンドルとなります。

#### 《書式》

##### [ C言語 ]

```
DWORD WINAPI ct540_OpenDevice(DWORD * hDevID, HCTRDEVINF * HpcDevInfo);
```

##### [ Visual Basic ]

```
Declare Function ct540_OpenDevice Lib "hict540.dll" (Byref hDevID As Long, _  
HpcDevInfo As HCTRDEVINF) As Long
```

#### 《引数》

- ◆ DWORD *hDevID*      .. デバイスハンドル
- ◆ HCTRDEVINF\* *HpcDevInfo*    .. オープンするデバイスの情報がセットされたエリアのアドレス  
(※ 構造体のメンバーはボード種別により異なります。)

#### 《戻り値》      処理結果

- 0            : 成功
- 0以外      : 失敗      .. 「10.4 ドライバ関数の戻り値」を参照して下さい。

#### 《呼び出し例》

パソコンにCTRボードが2枚装着されていることを想定します。  
デバイス情報格納エリアとしてHCTRDEVINF型の配列HpcDevInfo[2]を準備します。

##### [ C言語 ]

```
DWORD ret;            //関数の戻り値  
DWORD hDevID[2];    //デバイスハンドル取得エリア  
  
ret = ct540_OpenDevice(&hDevID[0], &HpcDevInfo[0]);    //1番目のデバイス情報  
ret = ct540_OpenDevice(&hDevID[1], &HpcDevInfo[1]);    //2番目のデバイス情報
```

##### [ Visual Basic ]

```
Dim ret            As Long    '関数の戻り値  
Dim hDevID(2)    As Long    'デバイスハンドル取得エリア  
  
ret = ct540_OpenDevice(hDevID(0), HpcDevInfo(0))    '1番目のデバイス情報  
ret = ct540_OpenDevice(hDevID(1), HpcDevInfo(1))    '2番目のデバイス情報
```

#### 《デバイスハンドル》

デバイスハンドルは次のデータ内容となっています。

31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
ボードID設定値				デバイスオープンNo			
0	0	0	0~15	0	0	1~16	

## (4) ct540\_CloseDevice( ) デバイスのクローズ

### 《機能》

デバイスハンドルで指定されたCTRボードをクローズします。  
以降、このデバイスハンドルは無効となります。

### 《書式》

#### [ C言語 ]

```
DWORD WINAPI ct540_CloseDevice(DWORD hDevID);
```

#### [ Visual Basic ]

```
Declare Function ct540_CloseDevice Lib "hict540.dll" (ByVal hDevID As Long) As Long
```

### 《引数》

◆ DWORD hDevID . . . クローズするボードのデバイスハンドル

### 《戻り値》 処理結果

0 : 成功

0以外 : 失敗 . . . 「10.4 ドライバ関数の戻り値」を参照して下さい。

### 《呼び出し例》

既にデバイスハンドルとして hDevID が取得されているものとします。

#### [ C言語 ]

```
DWORD ret; //関数の戻り値
```

```
ret = ct540_CloseDevice( hDevID );
```

#### [ Visual Basic ]

```
Dim ret As Long '関数の戻り値
```

```
ret = ct540_CloseDevice( hDevID )
```

<p>(5) ct540_rXYSts( ) XYchステータスの読込 ct540_rZUSts( ) ZUchステータスの読込</p>
--

## 《機能》

デバイスハンドルで指定されたCTRボードの、XY(ZU)chのステータスを読込み、指定したエリアに格納します。

## 《書式》

## [ C言語 ]

```
DWORD WINAPI ct540_rXYSts1(DWORD hDevID, WORD * sts);
DWORD WINAPI ct540_rZUSts1(DWORD hDevID, WORD * sts);
```

## [ Visual Basic ]

```
Declare Function ct540_rXYSts Lib "hict540.dll"(ByVal hDevID As Long, ByRef sts As Integer) As Long
Declare Function ct540_rZUSts Lib "hict540.dll"(ByVal hDevID As Long, ByRef sts As Integer) As Long
```

## 《引数》

- ◆ DWORD hDevID    ・ ・ 対象デバイスのデバイスハンドル
- ◆ WORD \* sts        ・ ・ 読込んだデータが格納されるエリアのアドレス

## 《戻り値》    処理結果

0           : 成功  
0以外       : 失敗    ・ ・ 「10.4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C言語 ]

```
DWORD ret;    //関数の戻り値
WORD sts;     //ステータス
```

```
ret = ct540_rXYSts ( hDevID,        //デバイスハンドル
                    &sts );        //格納先のアドレス
```

## [ Visual Basic ]

```
Dim ret        As Long        '関数の戻り値
Dim sts        As Integer     'ステータス
```

```
ret = ct540_rXYSts ( hDevID, _     'デバイスハンドル
                    sts )        '格納先のアドレス
```

(6) ct540\_wXYCmd( ) X Y c h 制御コマンドの書込  
 ct540\_wZUCmd( ) Z U c h 制御コマンドの書込

## 《機能》

デバイスハンドルで指定されたCTRボードの、XY(ZU)c hのコマンドバッファへコマンドデータを書込みます。

## 《書式》

## [ C言語 ]

```
DWORD WINAPI ct540_wXYCmd( DWORD hDevID, WORD cmd);
DWORD WINAPI ct540_wZUCmd( DWORD hDevID, WORD cmd);
```

## [ Visual Basic ]

```
Declare Function ct540_wXYCmd Lib "hict540.dll" ( ByVal hDevID As Long, ByVal cmd As Integer) As Long
Declare Function ct540_wZUCmd Lib "hict540.dll" ( ByVal hDevID As Long, ByVal cmd As Integer) As Long
```

## 《引数》

- ◆ DWORD hDevID    ・ ・ 対象デバイスのデバイスハンドル
- ◆ WORD cmd        ・ ・ コマンドデータ

## 《戻り値》 処理結果

0           : 成功  
 0以外       : 失敗    ・ ・ 「10.4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C言語 ]

```
DWORD ret;    //関数の戻り値

ret = ct540_wXYCmd( hDevID,        //デバイスハンドル
                  0x03);        //X Y c hカウンタクリア
```

## [ Visual Basic ]

```
Dim ret As Long    '関数の戻り値

ret = ct540_wXYCmd( hDevID,        'デバイスハンドル
                  &H3)            'X Y c hカウンタクリア
```

(7) ct540_rXYReg( )	X Y c h レジスタの読込
ct540_rZUReg( )	Z U c h レジスタの読込
ct540_wXYReg( )	X Y c h レジスタの書込
ct540_wZUReg( )	Z U c h レジスタの書込

## 《機能》

デバイスハンドルで指定されたCTRボードの、

レジスタの読込・・XY(ZU)chのレジスタ読込コマンドで指定したレジスタを読み込み、  
指定エリアに格納します。

レジスタの書込・・XY(ZU)chのレジスタ書込コマンドで指定したレジスタにデータを書込みます。

## 《書式》

## [ C言語 ]

```
DWORD WINAPI ct540_rXYReg(DWORD hDevID, WORD cmd, DWORD * reg);
DWORD WINAPI ct540_rZUReg(DWORD hDevID, WORD cmd, DWORD * reg);
DWORD WINAPI ct540_wXYReg(DWORD hDevID, WORD cmd, DWORD reg);
DWORD WINAPI ct540_wZUReg(DWORD hDevID, WORD cmd, DWORD reg);
```

## [ Visual Basic ]

```
Declare Function ct540_rXYReg Lib "hictr540.dll" (ByVal hDevID As Long, ByVal cmd As Integer, _
ByRef reg As Long) As Long
Declare Function ct540_rZUReg Lib "hictr540.dll" (ByVal hDevID As Long, ByVal cmd As Integer, _
ByRef reg As Long) As Long
Declare Function ct540_wXYReg Lib "hictr540.dll" (ByVal hDevID As Long, ByVal cmd As Integer, _
ByVal reg As Long) As Long
Declare Function ct540_wZUReg Lib "hictr540.dll" (ByVal hDevID As Long, ByVal cmd As Integer, _
ByVal reg As Long) As Long
```

## 《引数》

- ◆ DWORD hDevID    ・・ 対象デバイスのデバイスハンドル
- ◆ WORD cmd        ・・ レジスタ読込／書込コマンド
- ◆ DWORD \* reg     ・・ 読込んだデータが格納されるエリアのアドレス
- ◆ DWORD reg       ・・ レジスタ書込データ

## 《戻り値》 処理結果

0           : 成功  
0以外      : 失敗    ・・ 「10.4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C言語 ]

```
DWORD ret;    //関数の戻り値
DWORD reg;    //レジスタのデータ
```

```
ret = ct540_rXYReg( hDevID,        //デバイスハンドル
                  0x00c0,        //X c hのカウンタを読む
                  &reg);        //格納先のアドレス
ret = ct540_wXYReg( hDevID,        //デバイスハンドル
                  0x0080,        //X c hのカウンタを指定
                  10000);        //書込データ
```

## [ Visual Basic ]

```
Dim ret    As Long    '関数の戻り値
Dim reg    As Long    'レジスタのデータ

ret = ct540_rReg( hDevID, _        'デバイスハンドル
                  &H0c, _        'X c hのカウンタを読む
                  reg )        '格納先のアドレス
ret = ct540_wReg( hDevID, _        'デバイスハンドル
                  &H80, _        'X c hのカウンタを指定
                  10000 )        '書込データ
```

- (8) ct540\_rPortB( ) オプションポートのバイト読込  
 ct540\_rPortW( ) オプションポートのワード (2バイト) 読込  
 ct540\_wPortB( ) オプションポートへバイト書込  
 ct540\_wPortW( ) オプションポートへワード (2バイト) 書込

## 《機能》

デバイスハンドルで指定されたCTRボードの、  
 オプションポートの読込・・・オプションポートを読み、指定エリアに格納します。  
 オプションポートへ書込・・・オプションポートに指定データを書込みます。

## 《書式》

## [ C言語 ]

```
DWORD WINAPI ct540_rPortB(DWORD hDevID, BYTE OffsetAdrs, BYTE * byData);
DWORD WINAPI ct540_wPortB(DWORD hDevID, BYTE OffsetAdrs, BYTE byData);
DWORD WINAPI ct540_rPortW(DWORD hDevID, BYTE OffsetAdrs, WORD * wData);
DWORD WINAPI ct540_wPortW(DWORD hDevID, BYTE OffsetAdrs, WORD wData);
```

## [ Visual Basic ]

```
Declare Function ct540_rPortB Lib "hctr540.dll" _
    (ByVal hDevID As Long, ByVal OffsetAdrs As Byte, ByRef byData As Byte) As Long
Declare Function ct540_wPortB Lib "hctr540.dll" _
    (ByVal hDevID As Long, ByVal OffsetAdrs As Byte, ByVal byData As Byte) As Long
Declare Function ct540_rPortW Lib "hctr540.dll" _
    (ByVal hDevID As Long, ByVal OffsetAdrs As Byte, ByRef wData As Integer) As Long
Declare Function ct540_wPortW Lib "hctr540.dll" _
    (ByVal hDevID As Long, ByVal OffsetAdrs As Byte, ByVal wData As Integer) As Long
```

## 《引数》

- ◆ DWORD *hDevID*・・・対象デバイスのデバイスハンドル
- ◆ BYTE *OffsetAdrs*・・・オプションポートのオフセットアドレス
- ◆ BYTE \* *byData*・・・読込んだデータが格納される1バイトエリアのアドレス
- ◆ BYTE *byData*・・・オプションポートへの書込1バイトデータ
- ◆ WORD \* *wData*・・・読込んだデータが格納される2バイトエリアのアドレス
- ◆ WORD *wData*・・・オプションポートへの書込2バイトデータ

## 《戻り値》 処理結果

- 0 : 成功
- 0以外 : 失敗・・・「10.4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C言語 ]

```
DWORD ret; //関数の戻り値
BYTE byData;
```

```
ret = ct540_rPortB(hDevID, //デバイスハンドル
                  0x12, //オフセットアドレス(汎用出力ポート)
                  &byData); //格納先のアドレス
ret = ct540_wPortB(hDevID, //デバイスハンドル
                  0x12, //オフセットアドレス(汎用出力ポート)
                  0x01); //書込データ
```

## [ Visual Basic ]

```
Dim ret As Long '関数の戻り値
Dim byData As Byte
```

```
ret = ct540_rPortB(hDevID, _ 'デバイスハンドル
                  &H12, _ 'オフセットアドレス(汎用出力ポート)
                  byData) '格納先のアドレス
ret = ct540_wPortB(hDevID, _ 'デバイスハンドル
                  &H12, _ 'オフセットアドレス(汎用出力ポート)
                  &H1) '書込データ
```

(9) ct540_rXYBuf( )	X Y c h 入出力バッファの読込
ct540_rZUBuf( )	Z U c h 入出力バッファの読込
ct540_wXYBuf( )	X Y c h 入出力バッファの書込
ct540_wZUBuf( )	Z U c h 入出力バッファの書込

## 《機能》

デバイスハンドルで指定されたCTRボードの、

- 入出力バッファの読込・・・XY(ZU)chの入出力バッファを読み、指定エリアに格納します。
- 入出力バッファへ書込・・・XY(ZU)chの入出力バッファにデータを書込みます。

## 《書式》

## [ C言語 ]

```
DWORD WINAPI ct540_rXYBuf(DWORD hDevID, DWORD * data);
DWORD WINAPI ct540_rZUBuf(DWORD hDevID, DWORD * data);
DWORD WINAPI ct540_wXYBuf(DWORD hDevID, DWORD data);
DWORD WINAPI ct540_wZUBuf(DWORD hDevID, DWORD data);
```

## [ Visual Basic ]

```
Declare Function ct540_rXYBuf Lib "hict540.DLL" (ByVal hDevID As Long, ByRef data As Long) As Long
Declare Function ct540_rZUBuf Lib "hict540.DLL" (ByVal hDevID As Long, ByRef data As Long) As Long
Declare Function ct540_wXYBuf Lib "hict540.DLL" (ByVal hDevID As Long, ByRef data As Long) As Long
Declare Function ct540_wZUBuf Lib "hict540.DLL" (ByVal hDevID As Long, ByVal data As Long) As Long
```

## 《引数》

- ◆ DWORD *hDevID*・・・対象デバイスのデバイスハンドル
- ◆ DWORD \* *data*・・・読込んだデータが格納されるエリアのアドレス
- ◆ DWORD *data*・・・入出力バッファへの書込データ

## 《戻り値》 処理結果

- 0 : 成功
- 0以外 : 失敗・・・「10.4 ドライバ関数の戻り値」を参照して下さい。

## 《呼び出し例》

## [ C言語 ]

```
DWORD ret; //関数の戻り値
DWORD dwData; //入出力バッファデータ
```

```
ret = ct540_rXYBuf( hDevID, //デバイスハンドル
                  &dwData ); //格納先のアドレス
```

```
ret = ct540_wXYBuf( hDevID, //デバイスハンドル
                  10000 ); //入出力バッファデータ
```

## [ Visual Basic ]

```
Dim ret As Long '関数の戻り値
Dim dwData As Long '入出力バッファデータ
```

```
ret = ct540_rXYBuf( hDevID, _ 'デバイスハンドル
                  dwData ) '格納先のアドレス
```

```
ret = ct540_wXYBuf( hDevID, 'デバイスハンドル
                  10000 ) '入出力バッファデータ
```

### 11. 5. 2 Windows版サンプル（動かしてみる）

「動かしてみる」(tpct540.exe)プログラムは、ボードをパソコンへ装着するだけで、最小限の動作をディスプレイ上で確認できるソフトです。

添付ソフトウェアディスクの「(E:)¥test¥Release¥tpct540.exe」を実行して下さい。

<注 意>

デバイスドライバのインストールが完了するまでは動作できません。

### 11. 5. 3 「動かしてみる」の操作

「動かしてみる」プログラム実行で次の画面が表示されます。

ボードIDまたはデバイス番号でボードを選択します。

また、ボードの初期化は一部ソースプログラムで固定されています。

その為に、初期化の条件を変更して動作させたい場合には、ソースプログラム変更の必要があります。

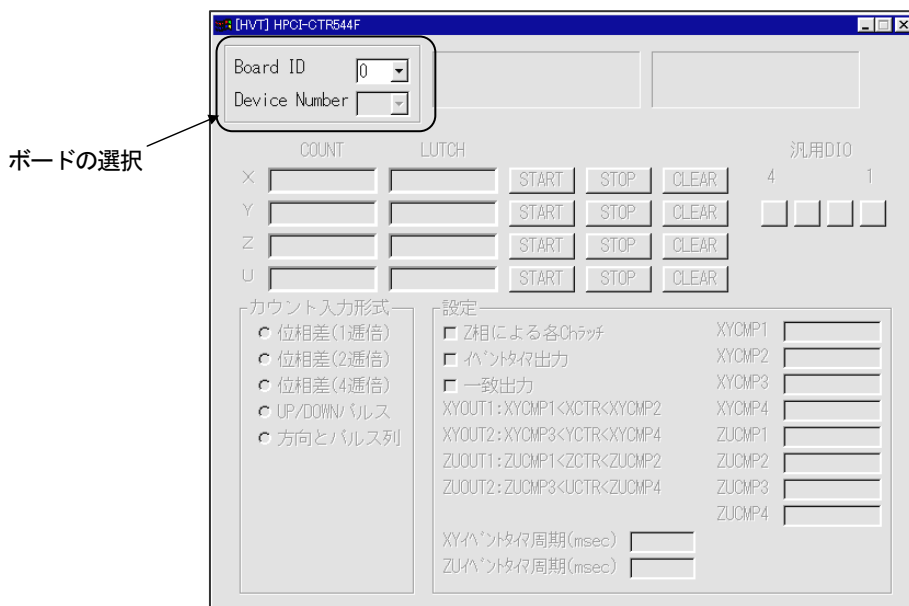


図. プログラム起動時画面



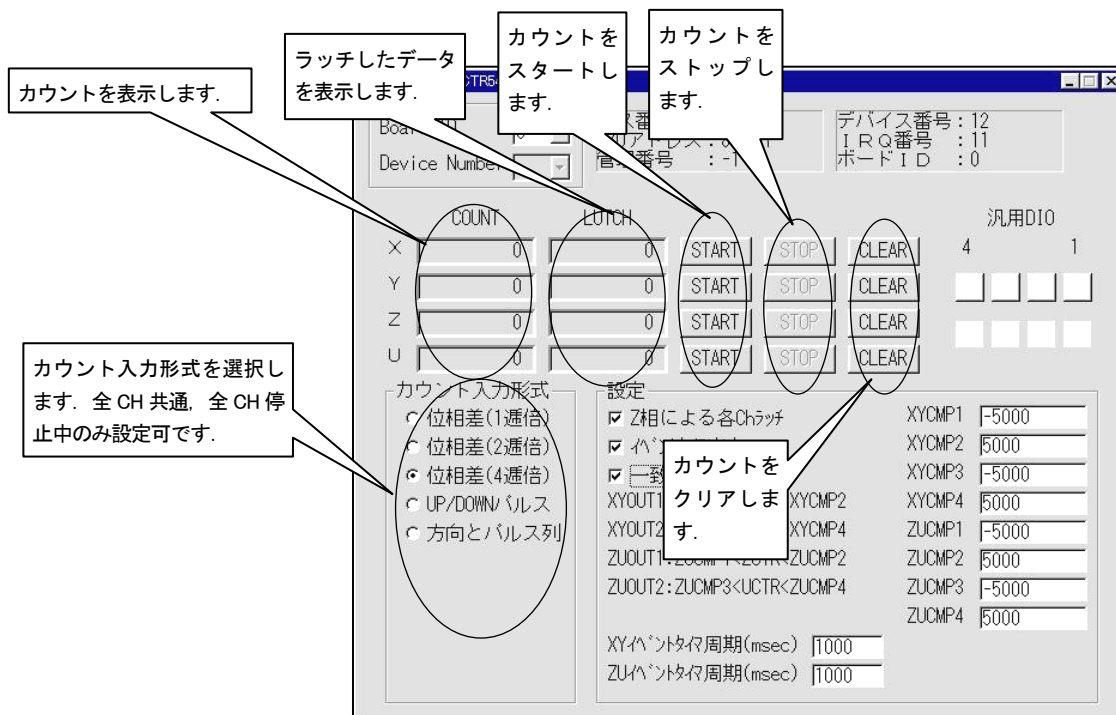
図. ボード選択後画面

(1) ボード選択とデバイス情報表示

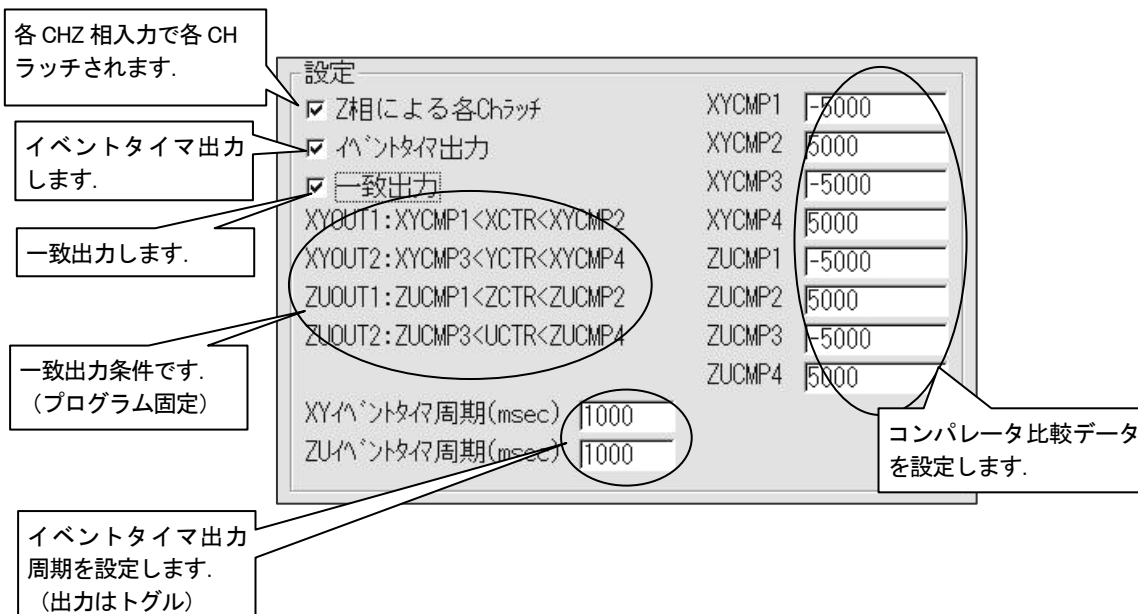


(2) 個々の表示と各設定

① カウントスタート/ストップ, カウントクリア, カウント入力形式設定



② Z相入力によるラッチ/イベントタイマ出力/一致出力



③汎用入出力



汎用入力1～4の状態を表示します。  
アクティブ状態・・・緑色表示  
ノンアクティブ状態・・・白色表示

これらのボタンをクリックすると汎用出力1～4を出力します。  
また、汎用出力1～4の状態を表示します。  
アクティブ状態・・・緑色表示  
ノンアクティブ状態・・・グレー表示