

CNT-3208M-PE Board Support Package Installation on RedHawk

Release Notes Revision B

September 9, 2022



1. はじめに

本書は、Concurrent Real Time Inc(CCRT)の RedHawk 上で動作する、コンテック社製 CNT3208M PCI Express ボードサポートパッケージ 用リリースノートです。

2. インストールのための条件

CNT3208M BSP をインストールするためには、以下の製品がインストールされている事が必要です。

- CNT3208M ボード
- RedHawk 6.x 以上
- Extmem version 8.3 以上

CNT3208Mは、外部装置からのパルス信号を入力してカウントする、PCI Expressバス準拠のカウンタ入力ボードです。32ビットアップダウンカウンタを8ch搭載しており、バスマスタ転送機能により最大10MHzまでの高速パルス入力(非絶縁TTLレベル、差動)で高速データ転送ができます。また、ロータリエンコーダ、リニアスケールなどの2相信号および单相信号をカウントできます。

3. インストール方法

CNT3208M BSP は、IRQ 共有するように設計されています。もしこのデバイスの IRQ が、別のデバイスによって共有されている場合に、このドライバの性能は損なわれる場合があります。そのため、可能な限り、このボードはその IRQ が他の装置と共有されていないPCIスロットの中に実装する事が奨励されます。“lspci -v”コマンドをシステムで種々の装置の IRQ を確認するために使用することができます。

CNT3208M BSP は、CDROM/DVD 上の RPM/DEB フォーマットで供給され、別途 extmem デバイスドライバがインストールされている必要があります。

以下に、インストールの手順を示します。:

x86_64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
# mount /dev/cdrom /mnt あるいは mount /dev/dvd /mnt
# cd /mnt
もし、extmem を同時にインストールする場合には、以下のコマンドを入力してください
# rpm -ivh bin-extmem-X.Y_RHx.y-z.x86_64.rpm
CNT3208M BSP 実行パッケージのインストール
# rpm -ivh bin-cnt3208m-X.Y_RHx.y-z.x86_64.rpm
もし必要であれば、続けて開発パッケージのインストールを行ってください
# rpm -ivh dev-cnt3208m-X.Y_RHx.y-z.x86_64.rpm
# umount /mnt
```

amd64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
# mount /dev/cdrom /mnt あるいは mount /dev/dvd /mnt
# cd /mnt
もし、extmem を同時にインストールする場合には、以下のコマンドを入力してください
# apt install ./bin-extmem-rhx.y_X.Y_amd64.deb
```

CNT3208M BSP 実行パッケージのインストール

```
# apt install ./bin-cnt3208m-rhx.y_X.Y_amd64.deb
```

もし必要であれば、続けて開発パッケージのインストールを行ってください

```
# apt install ./dev-cnt3208m-rhx.y_X.Y_amd64.deb
# umount /mnt
```

arm64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
# mount /dev/cdrom /mnt あるいは mount /dev/dvd /mnt
# cd /mnt
もし、extmem を同時にインストールする場合には、以下のコマンドを入力してください
# apt install ./bin-extmem-rhx.y_X.Y_arm64.deb
```

CNT3208M BSP 実行パッケージのインストール

```
# apt install ./bin-cnt3208m-rhx.y_X.Y_arm64.deb
```

もし必要であれば、続けて開発パッケージのインストールを行ってください

```
# apt install ./dev-cnt3208m-rhx.y_X.Y_arm64.deb
# umount /mnt
```

(x.y は RedHawk のバージョン番号であり、6.x,7.x または 8.x で、X.Y は、BSP のバージョン、z は、BSP のリリース番号を示し、予告なく変更することがあります。)

CNT3208M BSP パッケージは */usr/local/CNC/drivers/extmem/interface/cnt3208m* ディレクトリにインストールされ、必要な場所に展開されます。

4. アンインストール方法

CNT3208M BSP パッケージは、以下のコマンドでアンインストールします。この作業により */usr/local/CNC/drivers/extmem/interface/cnt3208m* ディレクトリは削除されます。

x86_64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
開発パッケージをインストールしていた場合には、
# rpm -e dev-cnt3208m-X.Y_RHx.y-z.x86_64 (開発パッケージの削除)
# rpm -e bin-cnt3208m-X.Y_RHx.y-z.x86_64 (実行パッケージの削除)
実行パッケージのみをインストールしていた場合には、
# rpm -e bin-cnt3208m-X.Y_RHx.y-z.x86_64 (実行パッケージの削除)
```

amd64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
```

```
開発パッケージをインストールしていた場合には、
# apt purge dev-cnt3208m-rhx.y (開発パッケージの削除)
# apt purge bin-cnt3208m-rhx.y (実行パッケージの削除)
実行パッケージのみをインストールしていた場合には、
# apt purge bin-cnt3208m-rhx.y (実行パッケージの削除)
```

arm64 アーキテクチャの場合

```
==== root ユーザで実行してください=====
```

```
開発パッケージをインストールしていた場合には、
# apt purge dev-cnt3208m-rhx.y (開発パッケージの削除)
# apt purge bin-cnt3208m-rhx.y (実行パッケージの削除)
実行パッケージのみをインストールしていた場合には、
# apt purge bin-cnt3208m-rhx.y (実行パッケージの削除)
```

5. ライブラリマニュアル

ライブラリマニュアルは、オンラインで提供されます。

man cnt3208m
cnt3208m(3)

cnt3208m(3)

NAME
cnt3208m - external memory device access library

SYNOPSIS
[ボードの詳細は、各メーカーマニュアルを見てください]

DESCRIPTION
cnt3208m は、external memory ドライバを利用した cnt3208m ボードサポートライブラリ。

```
#include <sys/cnt3208m.h>
gcc [options ...] file -lcnt3208m -lextmem ...
```

```
*****
CNT3208M
*****
```

デバイスの非初期化処理

```
int cnt3208m_reset(int fd);
int cnt3208m_uninit(int fd);
    戻り値
        エラーなら-1 成功なら 0
    引数
        fd ファイルディスクリプタ番号
        2つの関数は同じ処理、レジスタに値を設定する(下記)。
```

ソフトウェアストップ	CNT3208M_CNTRSEL_ENABLE_PORT_OFFSET	0
割り込み要求信号のマスクを割り込み無効設定する	CNT3208M_IRQ_MASK_PORT_OFFSET	CNT3208M_IRQ_ALL
割り込みフラグを読み出し	CNT3208M_IRQ_PORT_OFFSET	0
割り込みフラグをクリア	CNT3208M_IRQ_PORT_OFFSET	CNT3208M_CMD_TEST_PULSE_SETTING
テストパルスディセーブル	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_TEST_PULSE_INTERNAL CNT3208M_CMD_TEST_PULSE_DISABLE
	CNT3208M_CMD_DATA_PORT_OFFSET	
ソフトゼロクリア	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_SOFT_ZEROCLEAR
	CNT3208M_CMD_DATA_PORT_OFFSET	CNT3208M_CMD_SOFT_ZEROCLEAR_ALL
CHO 比較レジスタ 0 設定クリア	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_COMPARE_REG0_CHO
	CNT3208M_CMD_DATA_PORT_OFFSET	0
CHO 比較レジスタ 1 設定クリア	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_COMPARE_REG1_CHO
	CNT3208M_CMD_DATA_PORT_OFFSET	0
:	:	:
CH7 比較レジスタ 0 設定クリア	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_COMPARE_REG0_CH7
	CNT3208M_CMD_DATA_PORT_OFFSET	0
CH7 比較レジスタ 1 設定クリア	CNT3208M_CMD_PORT_OFFSET	CNT3208M_CMD_COMPARE_REG1_CH7
	CNT3208M_CMD_DATA_PORT_OFFSET	0

デバイスの初期化処理

```
int cnt3208m_init(int fd, int option);
    戻り値
        エラーなら-1 成功なら 0
    引数
        fd ファイルディスクリプタ番号
        option 1を指定すると以下の情報が表示される
        BAR0 I/O Region addr 0x0000ccaa offset 0x00000000 32 bytes
```

```
int cnt3208m_set_interval_timer(int fd, unsigned long int count, int flag);
    インターバルタイマーを設定する
    戻り値
```

エラーなら-1 成功なら 0

```
    引数
        fd ファイルディスクリプタ番号
        count インターバルタイマー値 0x0001(0.1ms) から 0xFFFF(6553.5ms)
        flag CNT3208M_CMD_DATA_TIMER_START または CNT3208M_CMD_DATA_TIMER_STOP
```

オンボードシステムタイマーの現在値を読み出す

アップカウンタは4MHz(250nsecステップ)のクロックで動作する32ビットバイナリカウンタで、
00000000hから順次カウントアップしていき、FFFFFFFFFFhになった後、再び00000000hになり、
この操作を繰り返します。

アップカウンタは電源投入後から常に動作します。

時間との対応は、カウントデータの1 LSB が 250nsecに対応します。

最大値 FFFFFFFFh は 1,073,741,823,750nsec(約 18 分)に相当します。

カウントデータはタイマの動作に影響を与えずに入力することができます。

```
int cnt3208m_get_system_timer(int fd, unsigned int *count);
```

戻り値

エラーなら-1 成功なら 0

引数

fd ファイルディスクリプタ番号
count 値を読み出す変数へのポインタ

汎用関数 オフセット値を指定してレジスタの値を読み出す

```
int cnt3208m_get_iport(int fd, int offset, unsigned int *value);
```

戻り値

エラーなら-1 成功なら 0

引数

fd ファイルディスクリプタ番号
offset レジスタオフセット
value 値を読み出す変数へのポインタ

汎用関数 オフセット値を指定してレジスタに値を書き出す

```
int cnt3208m_set_iport(int fd, int offset, unsigned int *value);
```

戻り値

エラーなら-1 成功なら 0

引数

fd ファイルディスクリプタ番号
offset レジスタオフセット
value 値を出す変数へのポインタ

機能設定コマンド

コマンドポート(0x10)に、commandを(16bit)出力し、データポート(0x0C)から、value(32bit)を入力する。

```
int cnt3208m_getcommand(int fd, unsigned long int command, unsigned long int *value);
```

[command]の定義は以下の通り

command	設定値	機能
CNT3208M_CMD_TIMER	0x0007	タイマ設定
CNT3208M_CMD_CNTRMODE_CH0	0x0008	CH0 カウンタモード
CNT3208M_CMD_CNTRMODE_CH1	0x0009	CH1 カウンタモード
CNT3208M_CMD_CNTRMODE_CH2	0x000A	CH2 カウンタモード
CNT3208M_CMD_CNTRMODE_CH3	0x000B	CH3 カウンタモード
CNT3208M_CMD_CNTRMODE_CH4	0x000C	CH4 カウンタモード
CNT3208M_CMD_CNTRMODE_CH5	0x000D	CH5 カウンタモード
CNT3208M_CMD_CNTRMODE_CH6	0x000E	CH6 カウンタモード
CNT3208M_CMD_CNTRMODE_CH7	0x000F	CH7 カウンタモード
CNT3208M_CMD_PRESETDATA_CH0	0x0010	CH0 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH1	0x0011	CH1 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH2	0x0012	CH2 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH3	0x0013	CH3 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH4	0x0014	CH4 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH5	0x0015	CH5 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH6	0x0016	CH6 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH7	0x0017	CH7 プリセットデータ
CNT3208M_CMD_COMPARE_REG0_CH0	0x0018	CH0 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH1	0x0019	CH1 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH2	0x001A	CH2 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH3	0x001B	CH3 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH4	0x001C	CH4 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH5	0x001D	CH5 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH6	0x001E	CH6 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH7	0x001F	CH7 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG1_CH0	0x0020	CH0 比較レジスタ 1

コマンドポート(0x10)に、commandを(16bit)出力し、パラメータをデータポート(0x0C)に、value(32bit)を出力する。

```
int cnt3208m_setcommand(int fd, unsigned long int command, unsigned long int value);
```

[command]の定義は以下の通り

command	設定値	機能
CNT3208M_CMD_TIMER	0x0007	タイマ設定
CNT3208M_CMD_CNTRMODE_CH0	0x0008	CH0 カウンタモード
CNT3208M_CMD_CNTRMODE_CH1	0x0009	CH1 カウンタモード
CNT3208M_CMD_CNTRMODE_CH2	0x000A	CH2 カウンタモード
CNT3208M_CMD_CNTRMODE_CH3	0x000B	CH3 カウンタモード
CNT3208M_CMD_CNTRMODE_CH4	0x000C	CH4 カウンタモード
CNT3208M_CMD_CNTRMODE_CH5	0x000D	CH5 カウンタモード
CNT3208M_CMD_CNTRMODE_CH6	0x000E	CH6 カウンタモード
CNT3208M_CMD_CNTRMODE_CH7	0x000F	CH7 カウンタモード
CNT3208M_CMD_PRESETDATA_CH0	0x0010	CH0 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH1	0x0011	CH1 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH2	0x0012	CH2 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH3	0x0013	CH3 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH4	0x0014	CH4 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH5	0x0015	CH5 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH6	0x0016	CH6 プリセットデータ
CNT3208M_CMD_PRESETDATA_CH7	0x0017	CH7 プリセットデータ
CNT3208M_CMD_COMPARE_REG0_CH0	0x0018	CH0 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH1	0x0019	CH1 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH2	0x001A	CH2 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH3	0x001B	CH3 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH4	0x001C	CH4 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH5	0x001D	CH5 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH6	0x001E	CH6 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG0_CH7	0x001F	CH7 比較レジスタ 0
CNT3208M_CMD_COMPARE_REG1_CH0	0x0020	CH0 比較レジスタ 1

CNT3208M_CMD_COMPARE_REG1_CH1	0x0021	CH1 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH2	0x0022	CH2 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH3	0x0023	CH3 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH4	0x0024	CH4 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH5	0x0025	CH5 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH6	0x0026	CH6 比較レジスタ 1
CNT3208M_CMD_COMPARE_REG1_CH7	0x0027	CH7 比較レジスタ 1
CNT3208M_CMD_CTRLSIG_SETTING_CH0	0x0028	CH0 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH1	0x0029	CH1 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH2	0x002A	CH2 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH3	0x002B	CH3 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH4	0x002C	CH4 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH5	0x002D	CH5 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH6	0x002E	CH6 制御出力信号設定
CNT3208M_CMD_CTRLSIG_SETTING_CH7	0x002F	CH7 制御出力信号設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH0	0x0030	CH0 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH1	0x0031	CH1 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH2	0x0032	CH2 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH3	0x0033	CH3 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH4	0x0034	CH4 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH5	0x0035	CH5 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH6	0x0036	CH6 Z 相入力設定
CNT3208M_CMD_PHASEZ_INPUT_SETTING_CH7	0x0037	CH7 Z 相入力設定
CNT3208M_CMD_CONCORDANCE_STATUS	0x0038	カウント一致ステータス/クリア
CNT3208M_CMD_ERSTATUS	0x0039	エラーステータス/クリア
CNT3208M_CMD_CBSTATUS	0x003A	キャリーボローステータス/クリア
CNT3208M_CMD_GENERAL_OUTPUT_DATA	0x003B	汎用出力データ設定
CNT3208M_CMD_SOFT_PRELOAD	0x003C	ソフトブリロード
CNT3208M_CMD_SOFT_ZEROCLEAR	0x003D	ソフトゼロクリア
CNT3208M_CMD_TEST_PULSE_SETTING	0x003E	テストパルス設定

[value]の定義は以下の通り

(1) タイマ設定 (07h) 用定義

CNT3208M_CMD_DATA_TIMER_START	0x00010000
CNT3208M_CMD_DATA_TIMER_STOP	0x00000000
CNT3208M_CMD_DATA_TIMER_VALUE_MASK	0x0000FFFF
CNT3208M_CMD_DATA_TIMER_VALUE_UNIT	100 [0.1ms]

(2) カウンタモード (08h - 0Fh) 用定義

D31 - D30 カウンタスタート条件を設定する。

00: ソフトウェア

01: Reserved

10: 外部 TTL レベル入力立ち上がり [外部入力信号で外部カウンタスタート/ストップが選択された場合のみ有効。]
11: 外部 TTL レベル入力立ち下がり [外部入力信号で外部カウンタスタート/ストップが選択された場合のみ有効。]

CNT3208M_CMD_CNTRMODE_DATA_START_SOFTWARE	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_START_EXT_TTL_RISE	0x80000000
CNT3208M_CMD_CNTRMODE_DATA_START_EXT_TTL_FALL	0xC0000000

D29 - D28 カウンタストップ条件を設定する。

00: ソフトウェア

01: Reserved

10: 外部 TTL レベル入力立ち上がり [外部入力信号で外部カウンタスタート/ストップが選択された場合のみ有効。]
11: 外部 TTL レベル入力立ち下がり [外部入力信号で外部カウンタスタート/ストップが選択された場合のみ有効。]

CNT3208M_CMD_CNTRMODE_DATA_STOP_SOFTWARE	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_STOP_EXT_TTL_RISE	0x20000000
CNT3208M_CMD_CNTRMODE_DATA_STOP_EXT_TTL_FALL	0x30000000

D27 - D26 外部入力信号の選択を設定する。

00 : 汎用入力

01 : 外部カウンタスタート/ストップ

10 : プリセットロード

11 : カウンタゼロクリア

CNT3208M_CMD_CNTRMODE_DATA_GENERAL_INPUT	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_EXTERNAL_COUNT_ARM	0x04000000
CNT3208M_CMD_CNTRMODE_DATA_PRELOAD	0x08000000
CNT3208M_CMD_CNTRMODE_DATA_COUNTER_ZEROCLEAR	0x0C000000

D25 - D24 0 固定

D23 - D20 プリセットロード条件の選択を設定する。

該当ビットに “1” を設定するとプリセットロード条件となる。

だし、ソフトウェア設定はこの設定条件に関らずプリセットロードが可能。

D23 : 外部 TTL レベル入力立ち下がり [外部入力信号でプリセットロードが選択された場合のみ有効。]

D22 : 外部 TTL レベル入力立ち上がり [外部入力信号でプリセットロードが選択された場合のみ有効。]

D21 : カウント一致 1

D20 : カウント一致 0

CNT3208M_CMD_CNTRMODE_DATA_PRELOAD_EXT_TTL_FALL	0x00800000
CNT3208M_CMD_CNTRMODE_DATA_PRELOAD_EXT_TTL_RISE	0x00400000
CNT3208M_CMD_CNTRMODE_DATA_PRELOAD_CONCORDANCE_COUNT1	0x00200000
CNT3208M_CMD_CNTRMODE_DATA_PRELOAD_CONCORDANCE_COUNT0	0x00100000

D19 – D16 カウンタゼロクリア条件の選択を設定する。
 該当ビットに “1” を設定するとカウンタゼロクリア条件となる。
 ただし、ソフトウェア、Z_相入力はこの設定条件に関らずゼロクリアが可能。
 D19 : 外部 TTL レベル入力立ち下がり [外部入力信号でカウンタゼロクリアが選択された場合のみ有効。]
 D18 : 外部 TTL レベル入力立ち上がり [外部入力信号でカウンタゼロクリアが選択された場合のみ有効。]
 D17 : カウント一致 1
 D16 : カウント一致 0

CNT3208M_CMD_CNTRMODE_DATA_OCLR_PRELOAD_EXT_TTL_FALL	0x00080000
CNT3208M_CMD_CNTRMODE_DATA_OCLR_PRELOAD_EXT_TTL_RISE	0x00040000
CNT3208M_CMD_CNTRMODE_DATA_OCLR_CONCORDANCE_COUNT1	0x00020000
CNT3208M_CMD_CNTRMODE_DATA_OCLR_CONCORDANCE_COUNT0	0x00010000

D15 – D12 デジタルフィルタの選択ポート。
 設定データ範囲 : 0.1us – 1.628ms または未使用

0000	: 未使用
0001	: 0.1us
0010	: 0.2us
0011	: 0.4us
0100	: 0.8us
0101	: 1.6us
0110	: 3.2us
0111	: 6.4us
1000	: 12.8us EX. 0.1us 設定時
1001	: 25.6us – 75ns のパルス幅を持つ信号に対してフィルタエラーが起ち、カウントしない。
1010	: 51.2us 75ns – 100ns のパルス幅を持つ信号に対してフィルタ エラーが起つこともあり、カウントすることもある。
1011	: 102.4us 100ns – のパルス幅を持つ信号に対してフィルタエラーは起たず、正常にカウントする。
1100	: 204.8us
1101	: 409.6us
1110	: 819.2us
1111	: 1.6384ms

CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_NONE	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_100NS	0x00001000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_200NS	0x00002000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_400NS	0x00003000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_800NS	0x00004000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_1600NS	0x00005000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_3200NS	0x00006000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_6400NS	0x00007000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_12800NS	0x00008000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_25600NS	0x00009000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_51200NS	0x0000A000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_102400NS	0x0000B000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_204800NS	0x0000C000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_409600NS	0x0000D000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_819200NS	0x0000E000
CNT3208M_CMD_CNTRMODE_DATA_DIGITAL_FILTER_1638400NS	0x0000F000

D11 – D09 0 固定
 D08 Z 相入力の論理(正/負)を設定する。
 0 : 正論理(High アクティブ)
 1 : 負論理(Low アクティブ)

CNT3208M_CMD_CNTRMODE_DATA_PHASEZ_ACTIVE_HIGH	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_PHASEZ_ACTIVE_LOW	0x00000100

D07 0 固定
 D06 カウンタへのパルス入力を設定する。
 0 : 差動入力
 1 : TTL レベル入力

CNT3208M_CMD_CNTRMODE_DATA_DIFFERENTIAL_INPUT	0x00000000
CNT3208M_CMD_CNTRMODE_DATA_TTL_INPUT	0x00000040

D05 カウント方向の切り替えを設定する。
 0 : CW 方向 Down/CCW 方向 Up
 1 : CW 方向 Up/CCW 方向 Down

CNT3208M_CMD_CNTRMODE_DATA_CW_UP	0x00000020
CNT3208M_CMD_CNTRMODE_DATA_CW_DOWN	0x00000000

D04 – D00 カウンタ動作モードの選択
 ポート。

CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_SCLEAR1	0x00000000	[2 相入力、同期クリア、1 遅倍モード]
CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_SCLEAR2	0x00000001	[2 相入力、同期クリア、2 遅倍モード]
CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_SCLEAR4	0x00000002	[2 相入力、同期クリア、4 遅倍モード]
CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_ACLEAR1	0x00000004	[2 相入力、非同期クリア、1 遅倍モード]
CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_ACLEAR2	0x00000005	[2 相入力、非同期クリア、2 遅倍モード]

CNT3208M_CMD_CNTRMODE_DATA_N2PHASE_ACLEAR4 0x00000006 [2 相入力、非同期クリア、4 遅倍モード]
 CNT3208M_CMD_CNTRMODE_DATA_N1PHASE_ACLEAR1 0x0000000C [単相入力、非同期クリア、1 遅倍モード]
 CNT3208M_CMD_CNTRMODE_DATA_G1PHASE_ACLEAR1 0x00000014 [ゲートコントロール付き単相入力、非同期クリア、1 遅倍モード]

期クリア、1 遅倍モード]

CNT3208M_CMD_CNTRMODE_DATA_G1PHASE_ACLEAR2 0x00000015 [ゲートコントロール付き単相入力、非同期クリア、2 遅倍モード]

(3) 制御出力信号設定 (28h - 2Fh) 用定義
 D07 - D05 ワンショットパルス幅を設定する。

000 : 10us
 001 : 100us
 010 : 1ms
 011 : 10ms
 100 : 100ms
 101 - 111 : Reserved

CNT3208M_CMD_CTRLSIG_DATA_OUT_10US	0x00000000
CNT3208M_CMD_CTRLSIG_DATA_OUT_10MS	0x00000020
CNT3208M_CMD_CTRLSIG_DATA_OUT_100US	0x00000040
CNT3208M_CMD_CTRLSIG_DATA_OUT_100MS	0x00000060
CNT3208M_CMD_CTRLSIG_DATA_OUT_1MS	0x00000080

D04 - D00 外部出力信号の選択を設定する。該当ビットに “1” を設定するとその要因により
 外部へワンショットパルスを出力する。
 全てのビット“0”とした場合は汎用出力となり、レベル出力となる
 (正/負論理の切り替えはボード上スイッチにて行う)。

D04 : アラームエラー
 D03 : 異常入力エラー
 D02 : デジタルフィルタエラー
 D01 : カウント一致 1
 D00 : カウント一致 0

CNT3208M_CMD_CTRLSIG_DATA_OUT_LEVEL	0x00000000
CNT3208M_CMD_CTRLSIG_DATA_OUT_ALARM_ERROR	0x00000010
CNT3208M_CMD_CTRLSIG_DATA_OUT_ILLIGALIN_ERROR	0x00000008
CNT3208M_CMD_CTRLSIG_DATA_OUT_DFILTER_ERROR	0x00000004
CNT3208M_CMD_CTRLSIG_DATA_OUT_COUNT_CONCORDANCE1	0x00000002
CNT3208M_CMD_CTRLSIG_DATA_OUT_COUNT_CONCORDANCE0	0x00000001

(4) Z 相入力設定 (30h - 37h) 用定義

01 - D00 Z 相入力の有効/無効を設定する。

CNT3208M_CMD_PHASEZ_DATA_VALID	0x00000000 [00 : 前回設定有効]
CNT3208M_CMD_PHASEZ_DATA_INVALID	0x00000001 [01 : Z 相入力無効 Z 相がない場合、この値に設定する]
CNT3208M_CMD_PHASEZ_DATA_ONCE	0x00000002 [10 : 次に来る Z 相の入力 1 回だけ有効]
CNT3208M_CMD_PHASEZ_DATA_EVERY	0x00000003 [11 : Z 相毎回有効]

(5) カウント一致ステータス (38h) 用定義

CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH7	0x80000000 [カウント一致 1DOWN (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH6	0x40000000 [カウント一致 1DOWN (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH5	0x20000000 [カウント一致 1DOWN (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH4	0x10000000 [カウント一致 1DOWN (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH3	0x08000000 [カウント一致 1DOWN (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH2	0x04000000 [カウント一致 1DOWN (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH1	0x02000000 [カウント一致 1DOWN (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CHO	0x01000000 [カウント一致 1DOWN (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH7	0x00800000 [カウント一致 0DOWN (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH6	0x00400000 [カウント一致 0DOWN (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH5	0x00200000 [カウント一致 0DOWN (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH4	0x00100000 [カウント一致 0DOWN (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH3	0x00080000 [カウント一致 0DOWN (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH2	0x00040000 [カウント一致 0DOWN (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CH1	0x00020000 [カウント一致 0DOWN (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_0DOWN_CHO	0x00010000 [カウント一致 0DOWN (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH7	0x00008000 [カウント一致 1UP (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH6	0x00004000 [カウント一致 1UP (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH5	0x00002000 [カウント一致 1UP (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH4	0x00001000 [カウント一致 1UP (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH3	0x00000800 [カウント一致 1UP (Ch 3)]

CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH2	0x00000400	[カウント一致 1UP (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH1	0x00000200	[カウント一致 1UP (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH0	0x00000100	[カウント一致 1UP (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH7	0x00000080	[カウント一致 OUP (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH6	0x00000040	[カウント一致 OUP (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH5	0x00000020	[カウント一致 OUP (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH4	0x00000010	[カウント一致 OUP (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH3	0x00000008	[カウント一致 OUP (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH2	0x00000004	[カウント一致 OUP (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH1	0x00000002	[カウント一致 OUP (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH0	0x00000001	[カウント一致 OUP (Ch 0)]

(6) エラーステータス (39h) 用定義

CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH7	0x00800000	[アラームエラー (Channel 7)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH6	0x00400000	[アラームエラー (Channel 6)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH5	0x00200000	[アラームエラー (Channel 5)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH4	0x00100000	[アラームエラー (Channel 4)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH3	0x00080000	[アラームエラー (Channel 3)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH2	0x00040000	[アラームエラー (Channel 2)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CH1	0x00020000	[アラームエラー (Channel 1)]
CNT3208M_CMD_ERSTATUS_DATA_ALARM_CHO	0x00010000	[アラームエラー (Channel 0)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH7	0x00008000	[デジタルフィルタエラー (Channel 7)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH6	0x00004000	[デジタルフィルタエラー (Channel 6)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH5	0x00002000	[デジタルフィルタエラー (Channel 5)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH4	0x00001000	[デジタルフィルタエラー (Channel 4)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH3	0x00000800	[デジタルフィルタエラー (Channel 3)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH2	0x00000400	[デジタルフィルタエラー (Channel 2)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CH1	0x00000200	[デジタルフィルタエラー (Channel 1)]
CNT3208M_CMD_ERSTATUS_DATA_DFILTER_CHO	0x00000100	[デジタルフィルタエラー (Channel 0)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH7	0x00000080	[異常入力エラー (Channel 7)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH6	0x00000040	[異常入力エラー (Channel 6)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH5	0x00000020	[異常入力エラー (Channel 5)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH4	0x00000010	[異常入力エラー (Channel 4)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH3	0x00000008	[異常入力エラー (Channel 3)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH2	0x00000004	[異常入力エラー (Channel 2)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CH1	0x00000002	[異常入力エラー (Channel 1)]
CNT3208M_CMD_ERSTATUS_DATA_ILLIGALIN_CHO	0x00000001	[異常入力エラー (Channel 0)]

(7) キャリーボローステータス (3Ah) 用定義

CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH7	0x00008000	[D15 : Borrow (Channel 7)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH6	0x00004000	[D14 : Borrow (Channel 6)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH5	0x00002000	[D13 : Borrow (Channel 5)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH4	0x00001000	[D12 : Borrow (Channel 4)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH3	0x00000800	[D11 : Borrow (Channel 3)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH2	0x00000400	[D10 : Borrow (Channel 2)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CH1	0x00000200	[D09 : Borrow (Channel 1)]
CNT3208M_CMD_CBSTATUS_DATA_BORROW_CHO	0x00000100	[D08 : Borrow (Channel 0)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH7	0x00000080	[D07 : Carry (Channel 7)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH6	0x00000040	[D06 : Carry (Channel 6)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH5	0x00000020	[D05 : Carry (Channel 5)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH4	0x00000010	[D04 : Carry (Channel 4)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH3	0x00000008	[D03 : Carry (Channel 3)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH2	0x00000004	[D02 : Carry (Channel 2)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CH1	0x00000002	[D01 : Carry (Channel 1)]
CNT3208M_CMD_CBSTATUS_DATA_CARRY_CHO	0x00000001	[D00 : Carry (Channel 0)]

(8) 汎用出力データ設定 (3Bh) 用定義

CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH7	0x00000080	[D07 : (Channel 7)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH6	0x00000040	[D06 : (Channel 6)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH5	0x00000020	[D05 : (Channel 5)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH4	0x00000010	[D04 : (Channel 4)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH3	0x00000008	[D03 : (Channel 3)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH2	0x00000004	[D02 : (Channel 2)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CH1	0x00000002	[D01 : (Channel 1)]
CNT3208M_CMD_GENERAL_OUTPUT_DATA_CHO	0x00000001	[D00 : (Channel 0)]

(9) ソフトプリロード (3Ch) 用定義

CNT3208M_CMD_SOFT_PRELOAD_CH7	0x00000080	[D07 : (Channel 7)]
CNT3208M_CMD_SOFT_PRELOAD_CH6	0x00000040	[D06 : (Channel 6)]
CNT3208M_CMD_SOFT_PRELOAD_CH5	0x00000020	[D05 : (Channel 5)]
CNT3208M_CMD_SOFT_PRELOAD_CH4	0x00000010	[D04 : (Channel 4)]
CNT3208M_CMD_SOFT_PRELOAD_CH3	0x00000008	[D03 : (Channel 3)]
CNT3208M_CMD_SOFT_PRELOAD_CH2	0x00000004	[D02 : (Channel 2)]
CNT3208M_CMD_SOFT_PRELOAD_CH1	0x00000002	[D01 : (Channel 1)]
CNT3208M_CMD_SOFT_PRELOAD_CHO	0x00000001	[D00 : (Channel 0)]

CNT3208M_CMD_SOFT_PRELOAD_ALL	0x000000FF
(10) ソフトゼロクリア (3Dh)用定義	
CNT3208M_CMD_SOFT_ZEROCLEAR_CH7	0x00000080 [D07 : (Channel 7)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH6	0x00000040 [D06 : (Channel 6)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH5	0x00000020 [D05 : (Channel 5)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH4	0x00000010 [D04 : (Channel 4)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH3	0x00000008 [D03 : (Channel 3)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH2	0x00000004 [D02 : (Channel 2)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CH1	0x00000002 [D01 : (Channel 1)]
CNT3208M_CMD_SOFT_ZEROCLEAR_CHO	0x00000001 [D00 : (Channel 0)]
CNT3208M_CMD_SOFT_ZEROCLEAR_ALL	0x000000FF

(11) テストパルス出力 (3Eh)用定義

- D01 出力パルスの有効/無効の設定をする。
 0 : テストパルスを出力しない
 1 : テストパルスを出力する
- D00 出力パルス信号の内部/外部の設定をする。
 0 : テストパルスを外部結線で使用
 1 : 内部で使用(CHO - CH7 全チャネルに 2 相パルスを出力する。)

CNT3208M_CMD_TEST_PULSE_ENABLE	0x00000000
CNT3208M_CMD_TEST_PULSE_DISABLE	0x00000002
CNT3208M_CMD_TEST_PULSE_EXTERNAL	0x00000000
CNT3208M_CMD_TEST_PULSE_INTERNAL	0x00000001

(12) カウンタステータス (3Fh)用定義

CNT3208M_CMD_COUNT_DATA_PHASEZ_CH7	0x80000000 [D31 : Z 相 (Channel 7)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH7	0x40000000 [D30 : A 相 (Channel 7)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH7	0x20000000 [D29 : B 相 (Channel 7)]
CNT3208M_CMD_COUNT_DATA_UD_CH7	0x10000000 [D28 : U/D (Channel 7)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH6	0x08000000 [D27 : Z 相 (Channel 6)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH6	0x04000000 [D26 : A 相 (Channel 6)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH6	0x02000000 [D25 : B 相 (Channel 6)]
CNT3208M_CMD_COUNT_DATA_UD_CH6	0x01000000 [D24 : U/D (Channel 6)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH5	0x00800000 [D23 : Z 相 (Channel 5)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH5	0x00400000 [D22 : A 相 (Channel 5)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH5	0x00200000 [D21 : B 相 (Channel 5)]
CNT3208M_CMD_COUNT_DATA_UD_CH5	0x00100000 [D20 : U/D (Channel 5)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH4	0x00080000 [D19 : Z 相 (Channel 4)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH4	0x00040000 [D18 : A 相 (Channel 4)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH4	0x00020000 [D17 : B 相 (Channel 4)]
CNT3208M_CMD_COUNT_DATA_UD_CH4	0x00010000 [D16 : U/D (Channel 4)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH3	0x00008000 [D15 : Z 相 (Channel 3)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH3	0x00004000 [D14 : A 相 (Channel 3)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH3	0x00002000 [D13 : B 相 (Channel 3)]
CNT3208M_CMD_COUNT_DATA_UD_CH3	0x00001000 [D12 : U/D (Channel 3)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH2	0x00000800 [D11 : Z 相 (Channel 2)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH2	0x00000400 [D10 : A 相 (Channel 2)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH2	0x00000200 [D09 : B 相 (Channel 2)]
CNT3208M_CMD_COUNT_DATA_UD_CH2	0x00000100 [D08 : U/D (Channel 2)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CH1	0x00000080 [D07 : Z 相 (Channel 1)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CH1	0x00000040 [D06 : A 相 (Channel 1)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CH1	0x00000020 [D05 : B 相 (Channel 1)]
CNT3208M_CMD_COUNT_DATA_UD_CH1	0x00000010 [D04 : U/D (Channel 1)]
CNT3208M_CMD_COUNT_DATA_PHASEZ_CHO	0x00000008 [D03 : Z 相 (Channel 0)]
CNT3208M_CMD_COUNT_DATA_PHASEA_CHO	0x00000004 [D02 : A 相 (Channel 0)]
CNT3208M_CMD_COUNT_DATA_PHASEB_CHO	0x00000002 [D01 : B 相 (Channel 0)]
CNT3208M_CMD_COUNT_DATA_UD_CHO	0x00000001 [D00 : U/D (Channel 0)]

LP 割り込みハンドラの登録と、割り込みの許可

```
int cnt3208m_setup_signal(int fd, void (*interrupt_handler)( int, siginfo_t *, void *), unsigned long int imask);
マスク値は、1 である場合に許可、0 である場合に不許可である(実際の書き出しには、NOT の値を使用する)
この関数の呼び出しによって、ボードの IRQ 信号によって、(SIGIO) シグナルハンドラが起動されるようになる。
また、本関数は、cnt3208m_enable_interrupt() を呼び出す。
```

割り込みを許可する

```
int cnt3208m_enable_interrupt(int fd, unsigned long int imask);
各マスク値は、1 である場合に許可、0 である場合に不許可である(実際の書き出しには、NOT の値を使用する)
```

割り込みを禁止する

```
int cnt3208m_disable_interrupt(int fd, unsigned long int imask);
各マスク値は、1 である場合に不許可、0 である場合に許可である(実際の書き出にその値を使用する)
```

割り込みサービス関数 割り込んだ際の割り込み要因レジスタの値を戻す

```
int cnt3208m_intr_service(int fd, unsigned long int *iflag, unsigned long int *cflag, unsigned long int
*pending);
```

通常本関数は、cnt3208m_setup_signal() 関数で登録したシグナルハンドラ内で呼び出される。

pending は、ペンディングされていた割り込み回数である。

iflag は、割り込み時の Interrupt Request Sense Port (+14h) レジスタ値である。

cflag は、割り込み時のカウント一致ステータス値である。

iflag 定義を以下に示す

CNT3208M_IRQ_COMPO_CH0	0x00000100
CNT3208M_IRQ_COMPO_CH1	0x00000200
CNT3208M_IRQ_COMPO_CH2	0x00000400
CNT3208M_IRQ_COMPO_CH3	0x00000800
CNT3208M_IRQ_COMPO_CH4	0x00001000
CNT3208M_IRQ_COMPO_CH5	0x00002000
CNT3208M_IRQ_COMPO_CH6	0x00004000
CNT3208M_IRQ_COMPO_CH7	0x00008000
CNT3208M_IRQ_COMPO_ALL	0x0000FF00
CNT3208M_IRQ_COMP1_CH0	0x00010000
CNT3208M_IRQ_COMP1_CH1	0x00020000
CNT3208M_IRQ_COMP1_CH2	0x00040000
CNT3208M_IRQ_COMP1_CH3	0x00080000
CNT3208M_IRQ_COMP1_CH4	0x00100000
CNT3208M_IRQ_COMP1_CH5	0x00200000
CNT3208M_IRQ_COMP1_CH6	0x00400000
CNT3208M_IRQ_COMP1_CH7	0x00800000
CNT3208M_IRQ_COMP1_ALL	0x00FF0000
CNT3208M_IRQ_DIGITAL_FILTER_ERROR	0x01000000
CNT3208M_IRQ_TIMER	0x02000000
CNT3208M_IRQ_CARRY_BORROW	0x04000000
CNT3208M_IRQ_AI_ERROR	0x08000000
CNT3208M_IRQ_ALARM_ERROR	0x10000000
CNT3208M_IRQ_ALL	0x1FFFFFFF

cflag 定義を以下に示す

CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH7	0x80000000	[カウント一致 1DOWN (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH6	0x40000000	[カウント一致 1DOWN (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH5	0x20000000	[カウント一致 1DOWN (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH4	0x10000000	[カウント一致 1DOWN (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH3	0x08000000	[カウント一致 1DOWN (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH2	0x04000000	[カウント一致 1DOWN (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH1	0x02000000	[カウント一致 1DOWN (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_1DOWN_CH0	0x01000000	[カウント一致 1DOWN (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH7	0x00800000	[カウント一致 ODOWN (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH6	0x00400000	[カウント一致 ODOWN (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH5	0x00200000	[カウント一致 ODOWN (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH4	0x00100000	[カウント一致 ODOWN (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH3	0x00080000	[カウント一致 ODOWN (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH2	0x00040000	[カウント一致 ODOWN (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH1	0x00020000	[カウント一致 ODOWN (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_CH0	0x00010000	[カウント一致 ODOWN (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_ODOWN_ALL	0xFFFF0000	[カウント一致 ODOWN (全チャネル)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH7	0x00008000	[カウント一致 1UP (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH6	0x00004000	[カウント一致 1UP (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH5	0x00002000	[カウント一致 1UP (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH4	0x00001000	[カウント一致 1UP (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH3	0x00000800	[カウント一致 1UP (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH2	0x00000400	[カウント一致 1UP (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH1	0x00000200	[カウント一致 1UP (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_1UP_CH0	0x00000100	[カウント一致 1UP (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH7	0x00000080	[カウント一致 OUP (Ch 7)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH6	0x00000040	[カウント一致 OUP (Ch 6)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH5	0x00000020	[カウント一致 OUP (Ch 5)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH4	0x00000010	[カウント一致 OUP (Ch 4)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH3	0x00000008	[カウント一致 OUP (Ch 3)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH2	0x00000004	[カウント一致 OUP (Ch 2)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH1	0x00000002	[カウント一致 OUP (Ch 1)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_CH0	0x00000001	[カウント一致 OUP (Ch 0)]
CNT3208M_CMD_CONCORDANCE_DATA_OUP_ALL	0x0000FFFF	[カウント一致 OUP (全チャネル)]

```
int cnt3208m_get_data(int fd, int ch, unsigned long int *value)
```

読み出しを行うチャネル番号を(CNT3208M_READ_CHANNEL_SELECT_PORT_OFFSET

で)指定してラッチされているカウントデータを読む

value の有効な値は 32 ビットである。(注意: 読み出し前にデータをラッチしておく必要がある。)

ch 読み出しを行うチャネル番号(以下のいずれかの値)

CNT3208M_READ_CHANNEL_SELECT_CH0
CNT3208M_READ_CHANNEL_SELECT_CH1
CNT3208M_READ_CHANNEL_SELECT_CH2
CNT3208M_READ_CHANNEL_SELECT_CH3
CNT3208M_READ_CHANNEL_SELECT_CH4
CNT3208M_READ_CHANNEL_SELECT_CH5
CNT3208M_READ_CHANNEL_SELECT_CH6

CNT3208M_READ_CHANNEL_SELECT_CH7

使用例

```
unsigned long int value,data[8];
value = CNT3208M_DIRECT_COUNTER_LATCH_SELECT_ALL; /* 全チャネルラッチ */
cnt3208m_set_ioport(fd,CNT3208M_DIRECT_COUNTER_LATCH_SELECT_PORT_OFFSET,&value);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH0,&data[0]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH1,&data[1]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH2,&data[2]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH3,&data[3]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH4,&data[4]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH5,&data[5]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH6,&data[6]);
cnt3208m_get_data(fd,CNT3208M_READ_CHANNEL_SELECT_CH7,&data[7]);
```

SEE ALSO

/usr/local/CNC/drivers/extmem/contec/cnt3208m 下のプログラム

AUTHORS

Copyright (C) 1995-2016 Concurrent Real Time Inc.

28 Apr 2016

cnt3208m(3)