

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

H8/300H Tiny シリーズ

内部割り込みによる多重割り込み動作

要旨

タイマ A, およびタイマ V 割り込みを使用して, 多重割り込み処理を行いません。

動作確認デバイス

H8/300H Tiny シリーズ –H8/3664–

目次

1. 仕様	2
2. 使用機能説明	3
3. 動作原理	5
4. ソフトウェア説明	6
5. フローチャート	8
6. プログラムリスト	10

1. 仕様

1. タイマ A, およびタイマ V 割り込みを使用して, 多重割り込み処理を行いません。
2. タイマ A 割り込み処理の中で, タイマ V 割り込み要求を受け付けることにより, ソフトウェアでタイマ V 割り込み要求の優先順位をタイマ A 割り込み要求の優先順位より高くなるように設定します。
3. タイマ A 割り込み要求は, インターバル機能により 32.768ms ごとに発生するように設定します。
4. タイマ V 割り込み要求は, インターバル機能により 2.048ms ごとに発生するように設定します。
5. LED はポート 7 の P74 出力端子に接続されているものとします。

2. 使用機能説明

1. 本タスク例では、内部割り込みを使用してタイマ A およびタイマ V 割り込みの多重割り込み動作を行います。以下に内部割り込みについて説明します。
 - 各内蔵周辺モジュールには割り込み要求ステータスフラグとこれらの割り込みイネーブルビットがあります。タイマ A 割り込み要求と SLEEP 命令実行によって発生する直接遷移割り込みについてはこの機能は IRR1, IENR1 に含まれています。内蔵周辺モジュールからの割り込み要求が発生すると対応する割り込み要求ステータスフラグが "1" にセットされ CPU に対して割り込みを要求します。これらの割り込み要求は対応するイネーブルビットを "0" にクリアすることにより禁止できます。
 - CCR の I ビットを "1" にセットすることにより、すべての割り込みをマスクできます。
 - 以下に割り込みの動作を示します。
 - (1) 割り込みイネーブルレジスタ 1 の対応するビットが "1" にセットされている状態で、割り込み要因が発生したとき、割り込みコントローラに対して割り込み要求信号が送られます。
 - (2) 割り込みコントローラに割り込み要求信号が送られると、割り込み要求フラグが "1" にセットされます。
 - (3) 割り込み要求フラグが "1" にセットされている割り込みの中で、優先順位に従って最高位の割り込み要求が選択され、その他は保留となります。
 - (4) CCR の I ビットを参照し、I ビットが "0" にクリアされている場合は、割り込み要求は受け付けられませんが、I ビットが "1" にセットされている場合は割り込み要求は保留となります。
 - (5) 割り込みが受け付けられると、そのとき実行中の命令の処理が終了した後、プログラムカウンタ (PC) と CCR がスタック領域に退避されます。スタックされる PC は、リターン後に実行する最初のアドレスを示しています。
 - (6) CCR の I ビットが "1" にセットされます。これにより、すべての割り込みはマスクされます。
 - (7) 受け付けた割り込みに対応するベクタアドレスを生成し、そのアドレスの内容によって示されるアドレスから、割り込み処理ルーチンの実行を開始します。
 - 割り込みイネーブルレジスタ 1 を "0" にクリアすることにより割り込みを禁止にする場合、または割り込みフラグレジスタ 1 をクリアする場合は、必ず割り込みをマスクした状態 (I = "1") で行ないます。I = "0" の状態でこのような操作を行なうと、当該操作命令の実行と当該割り込みの発生が競合した場合に、当該操作命令の実行終了時に発生した割り込みに対応する例外処理を実行します。

- 以下に本例におけるタイマ A 割り込み周期とタイマ V 割り込み周期の計算方法を示します。

$$\text{タイマA割り込み周期} = \frac{1}{\text{システムクロック}/2048\text{分周}} \times 256$$

$$32.768\text{ms} = \frac{1}{16\text{MHz}/2048\text{分周}} \times 256$$

$$\text{タイマV割り込み周期} = 2 \times \frac{1}{\text{システムクロック}/64\text{分周}} \times 256$$

$$2.048\text{ms} = 2 \times \frac{1}{16\text{MHz}/64\text{分周}} \times 256$$

2. 表 1 に本タスク例の機能割り付けを示します。表 1 に示すように機能を割り付け、内部割り込みによる多重割り込み動作を行ないます。

表 1 機能割り付け

機能	機能割り付け
OVIE	TCSR.V の OVF による割り込み要求の許可を行なう
IEN.A	タイマ A 割り込み要求の許可を行なう
OVF	TCNT.V のオーバフロー割り込み要求の有無を反映
IRR.TA	タイマ A 割り込み要求の有無を反映
CCR.Iビット	すべての割り込み要求の許可/禁止を行なう
P74	LED 出力

3. 動作原理

図1に動作原理を示します。図1に示すようなハードウェア処理、およびソフトウェア処理により内部割り込みによる多重割り込み動作を行ないます。

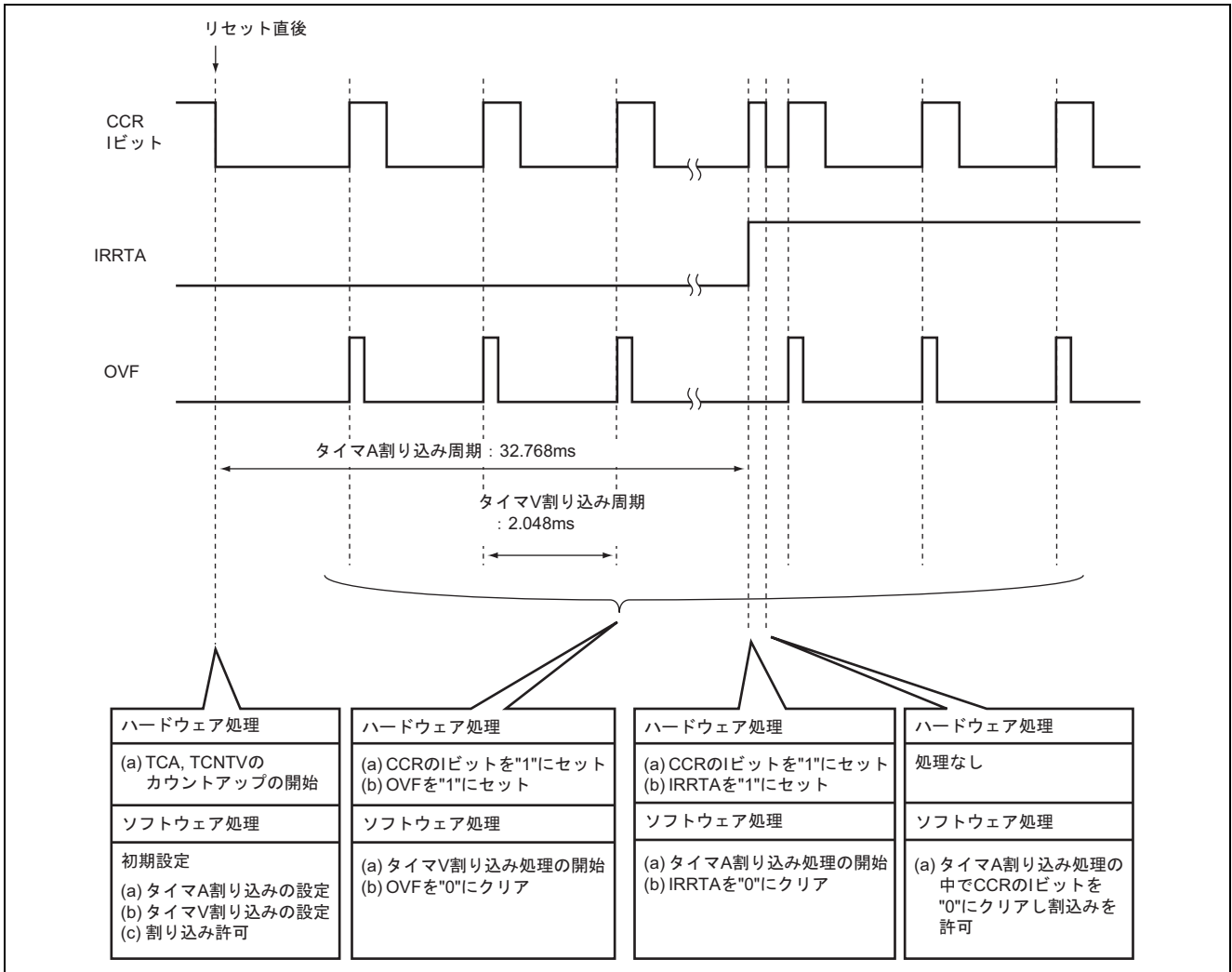


図1 内部割り込みによる多重割り込み動作の動作原理

4. ソフトウェア説明

4.1 モジュール説明

表 2 に本タスク例におけるモジュール説明を示します。

表 2 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	タイマ A 割り込みの設定, タイマ V 割り込みの設定, ポート 7 の設定, 割り込みの許可を行なう
カウント	taint	タイマ A 割り込み処理ルーチンで, 割り込みの許可, 16 ビットカウンタをインクリメントし, H'4000 になったら終了
LED 制御	tvint	タイマ V 割り込み処理ルーチンで, LED の点灯/消灯を行なう

4.2 引数の説明

本タスク例では, 引数は使用していません。

4.3 使用内部レジスタ説明

表 3 に本タスク例における使用内部レジスタ説明を示します。

表 3 使用内部レジスタ説明

レジスタ名		機能	アドレス	設定値
TCRV0	OVIE	タイマコントロールレジスタ V0 (タイマオーバーフローインタラプトイネーブル) : OVIE = 1 のとき TCNTV のオーバーフロー割り込みを許可	H'FFA0 ビット 5	0
	CKS2	タイマコントロールレジスタ V0 (クロックセレクト 2~0) : CKS2 = "0", CKS1 = "1", CKS0 = "1", ICKS0 = "0" のとき, TCNTV は システムクロックの 64 分周のクロックの立ち下がりエッジでカウント	H'FFA0 ビット 2 ビット 1 ビット 0	CKS2 = 0
	CKS1			CKS1 = 1
CKS0	CKS0 = 1			
TCSRv	OVF	タイマコントロール/ステータスレジスタ V (タイマオーバーフラグ) : OVF = 0 のとき TCNTV のオーバーフロー割り込みの要求されていない : OVF = 1 のとき TCNTV のオーバーフロー割り込みの要求されている	H'FFA1 ビット 5	0
TCNTV		タイマカウンタ V : システムクロックの 64 分周のクロックを入力とする 8 ビットのアップカウンタ	H'FFA4	H'00
TCRV1	ICKS0	タイマコントロールレジスタ V1 (インターナルクロックセレクト 0) : TCRV0 の CKS2 ~ CKS0 との組合わせて, TCNTV に入力するクロックを選択する。	H'FFA5 ビット 0	0
TMA		タイマモードレジスタ A : TMA = H'12 のとき, タイマ A 機能をインターバル機能に, TCA 入力クロックソース を PSS に, プリスケール分周比を 2048 分周に設定	H'FFA6	H'12
TCA		タイマカウンタ A : システムクロックを 2048 分周したクロックを入力とする 8 ビットのアップカウンタ	H'FFA7	H'00
PDR7	P74	ポートデータレジスタ 7 (ポートデータレジスタ 74) : P74 = 0 のとき, P74 端子の出力レベルは Low : P74 = 1 のとき, P74 端子の出力レベルは High	H'FFDA ビット 4	0
PCR7	PCR74	ポートコントロールレジスタ 7 (ポートコントロールレジスタ 74) : PCR74 = 1 のとき, P74 入出力端子は, P74 出力端子として機能	H'FFEA ビット 4	1
IENR1	IENTA	割り込みイネーブルレジスタ 1 (タイマ A 割り込み要求フラグ) : IENTA = 1 のとき, タイマ A 割り込み要求を許可	H'FFF4 ビット 6	1
IRR1	IRRTA	割り込みフラグレジスタ 1 (タイマ A 割り込み要求フラグ) : IRRTA = 0 のとき, タイマ A 割り込みが要求されていない : IRRTA = 1 のとき, タイマ A 割り込みが要求されている	H'FFF6 ビット 6	0

4.4 使用 RAM 説明

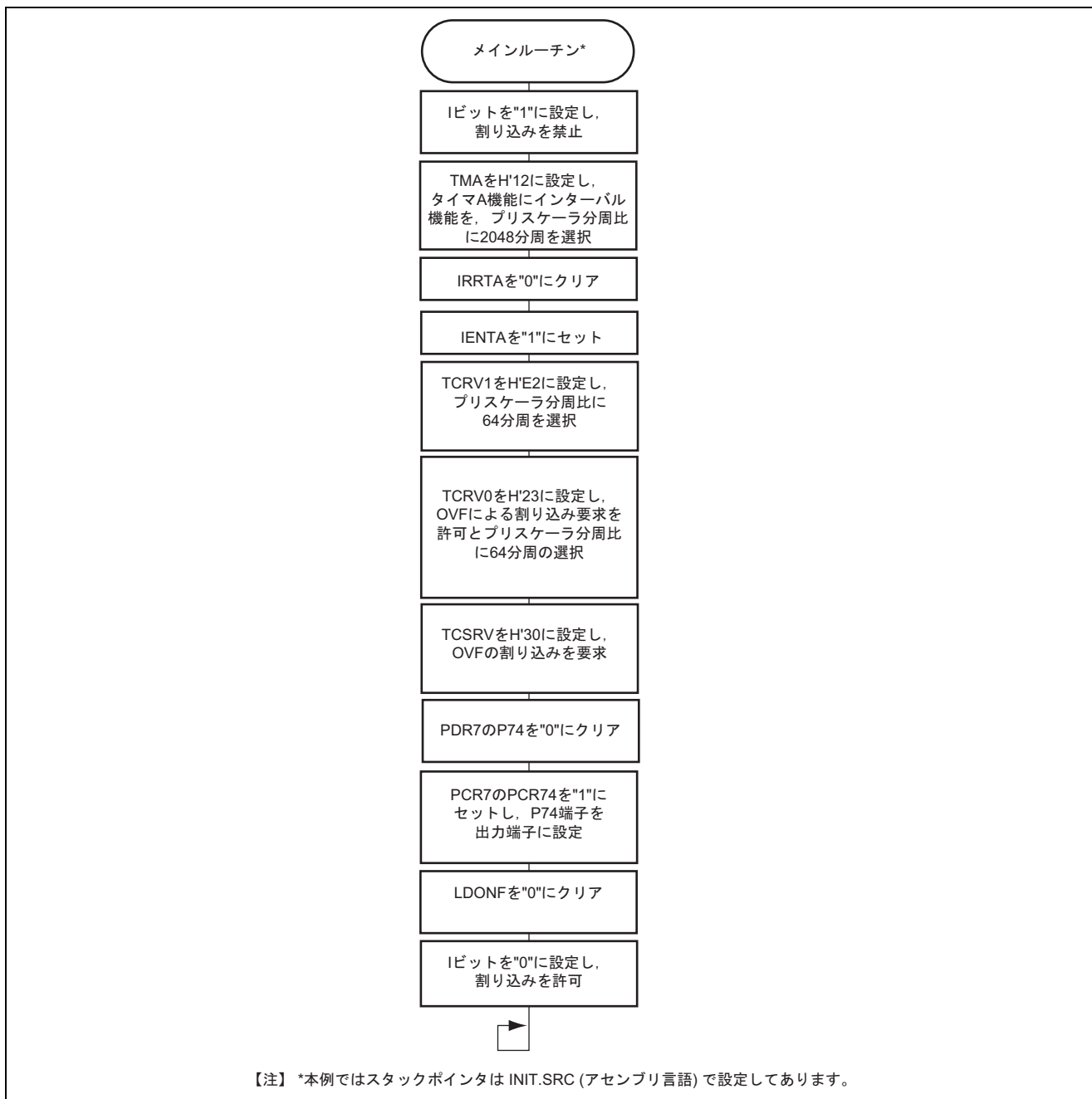
表 4 に本タスク例で使用する RAM の説明を示します。

表 4 使用 RAM 説明

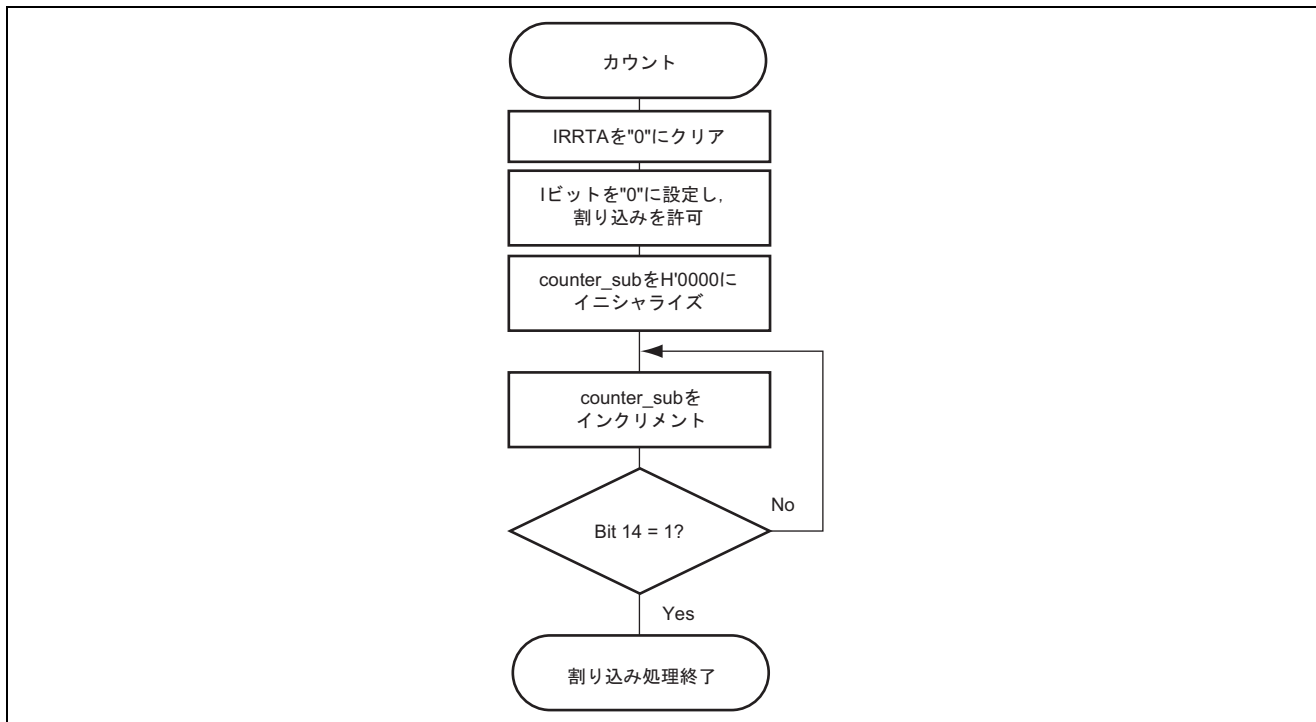
ラベル名		機能	アドレス	使用モジュール名
counter_sub		割り込みルーチン (タイマ A) の中で H'4000 まで カウントする 16 ビットアップカウンタ	H'FB80	割り込みルーチン タイマ A
USRF	LDONF	LED の ON/OFF を判定するフラグ	H'FB82 ビット 0	LED 制御

5. フローチャート

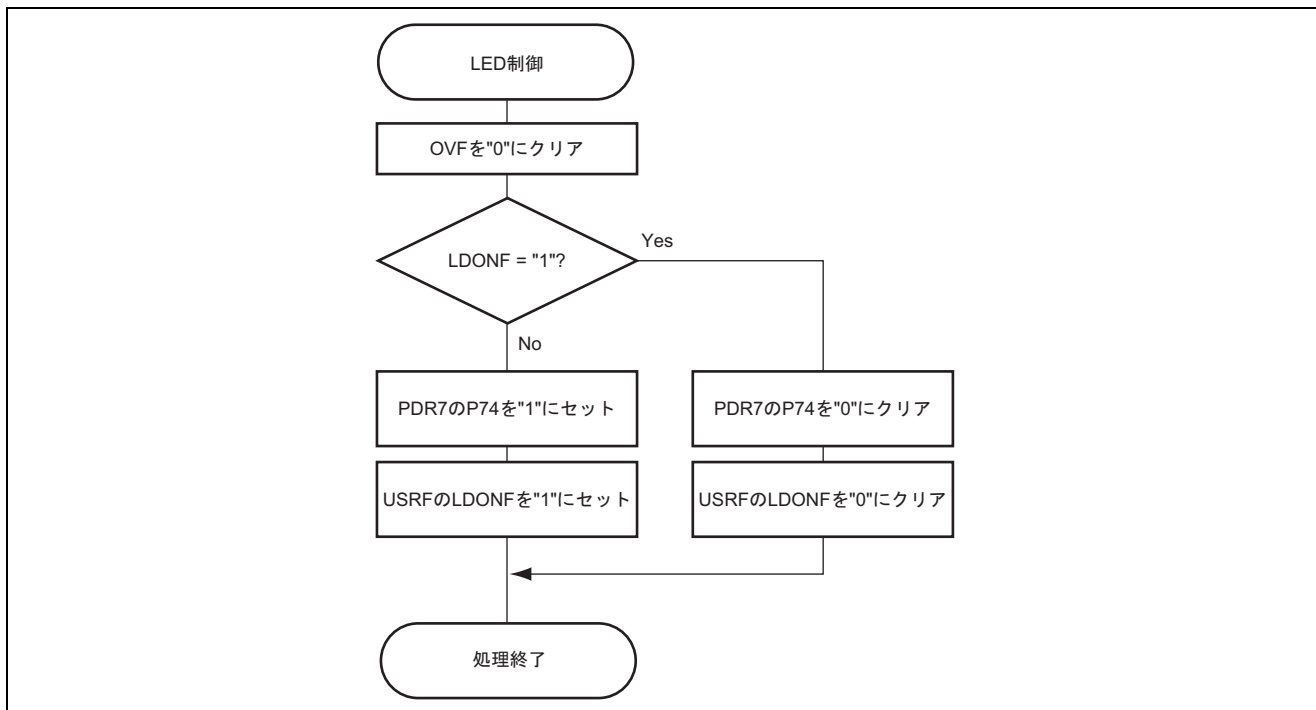
5.1 メインルーチン



5.2 タイマ A 割り込み処理ルーチン



5.3 タイマ V 割り込み処理ルーチン



6. プログラムリスト

INIT.SRC (プログラムリスト)

```

        .EXPORT          _INIT
        .IMPORT          _main
;

        .SECTION        P, CODE

_INIT:

        MOV.W           #H'FF80,R7
        LDC.B           #B'10000000,CCR
        JMP              @_main
;

        .END

/*****
/*
/* H8/300H Tiny Series -H8/3664-
/* Application Note
/*
/* 'Multiple Interrupt Operation by Internal
/* Interrupt'
/*
/* Function
/* : Internal Interrupt
/*
/* External Clock : 16MHz
/* Internal Clock : 16MHz
/* Sub Clock : 32.768kHz
/*
*****/

#include <machine.h>

/*****
/* Symbol Definition
*****/

struct BIT {
    unsigned char    b7:1;    /* bit7
    unsigned char    b6:1;    /* bit6
    unsigned char    b5:1;    /* bit5
    unsigned char    b4:1;    /* bit4
    unsigned char    b3:1;    /* bit3
    unsigned char    b2:1;    /* bit2
    unsigned char    b1:1;    /* bit1
    unsigned char    b0:1;    /* bit0
};

#define    TCRV0        *(volatile unsigned char *)0xFFA0    /* Timer Control Register V0
#define    TCSRVS        *(volatile unsigned char *)0xFFA1    /* Timer Control/Status Register V
#define    TCSRVS_BIT    (*(struct BIT *)0xFFA1)    /* Timer Control/Status Register V
#define    OVF        TCSRVS_BIT.b5    /* Timer Overflow Flag
#define    TCNTV        *(volatile unsigned char *)0xFFA4    /* Timer Counter V
#define    TCRV1        *(volatile unsigned char *)0xFFA5    /* Timer Control Register V1
#define    TMA        *(volatile unsigned char *)0xFFA6    /* Timer Mode Register A
#define    TCA        *(volatile unsigned char *)0xFFA7    /* Timer Counter A
#define    PDR7_BIT    (*(struct BIT *)0xFFDA)    /* Port Data Register 7
#define    P74        PDR7_BIT.b4    /* Port Data Register 7 bit4
#define    PCR7_BIT    (*(struct BIT *)0xFFEA)    /* Port Control Register 7
#define    PCR74        PCR7_BIT.b4    /* Port Control Register 7 bit4

```

```

#define IENR1_BIT    (*(struct BIT *)0xFFFF4)          /* Interrupt Enable Register 1          */
#define IENTA       IENR1_BIT.b6                     /* Timer A Interrupt Enable            */
#define IEN0        IENR1_BIT.b0                     /* IRQ0 Interrupt Enable              */
#define IRR1_BIT    (*(struct BIT *)0xFFFF6)          /* Interrupt Request Register 1        */
#define IRRTA       IRR1_BIT.b6                     /* Timer A Interrupt Request Flag      */
#define IRRIO       IRR1_BIT.b0                     /* IRQ0 Interrupt Request Flag        */
#define PMR1_BIT    (*(struct BIT *)0xFFE0)          /* Port Mode Register 1               */
#define IRQ0_SET    PMR1_BIT.b4                     /* Port Mode Register 1 bit4          */
#define TMOW        PMR1_BIT.b0                     /* P10/TMOW Terminal Function Change  */
#define IEGR1_BIT   (*(struct BIT *)0xFFFF2)          /* Interrupt Edge Select Register 1    */
#define IEG0        IEGR1_BIT.b0                     /* IRQ0 Edge Select                   */

#pragma interrupt    (taint)
#pragma interrupt    (tvint)
/*****
/*      関数定義
*****/
extern void INIT( void );                          /* SP Set
void main ( void );
void taint ( void );
void tvint ( void );

/*****
/*      RAM define
*****/
unsigned int counter_sub;
unsigned char USRF;                                /* User Flag Area

#define USRF_BIT    (*(struct BIT *)&USRF)
#define LDONF      USRF_BIT.b0                     /* LED On Flag

/*****
/*      Vector Address
*****/
#pragma section V1                                /* VECTOR SECTOIN SET
void (*const VEC_TBL1[])(void) = {
    INIT                                          /* 00 Reset
};
#pragma section V2                                /* VECTOR SECTOIN SET
void (*const VEC_TBL2[])(void) = {
    taint                                        /* 26 Timer A Interrupt
};
#pragma section V3                                /* VECTOR SECTOIN SET
void (*const VEC_TBL3[])(void) = {
    tvint                                        /* 2C Timer V Interrupt

#pragma section /* P
/*****
/*      Main Program
*****/
void main ( void )
{

    set_imask_ccr(1);                            /* Interrupt Disable
    TMA = 0x12;                                  /* Initialize TMA Function & TCA Input Clock Period

```

```

IRRRTA = 0; /* Clear IRRRTA */

IENTA = 1; /* Timer A Interrupt Enable */

TCRV1 = 0xE2; /* Initialize Time Control Register V1 */
TCRV0 = 0x23; /* Initialize TCRV0 Function & TCRV0 Input Clock Period */
TCSRv = 0x30; /* Initialize Timer Control/Status Register V */

P74 = 0; /* Clear P74 */
PCR74 = 1; /* Initialize P74 Output Terminal */

LDONF = 0; /* Clear LDONF */

set_imask_ccr(0); /* Interrupt Enable */

while(1){
    ;
}
}
/*****
/* Timer A Interrupt */
*****/
void taint ( void )
{

IRRRTA = 0; /* Clear IRRRTA */

set_imask_ccr(0); /* Interrupt Enable */

counter_sub = 0x0000; /* Initialize 16bit Counter */

do{
    counter_sub++; /* Increment 16bit Counter */
}while((counter_sub & 0x4000) == 0); /* 16bit Counter bit14 = 1 ? */

}

/*****
/* Timer V Interrupt */
*****/
void tvint ( void )
{

OVF = 0;

if(LDONF == 1){ /* LDONF = 1 ? */
    P74 = 0; /* Turn off LED */
    LDONF = 0; /* Clear LDONF */
}
else{
    P74 = 1; /* urn on LED */
    LDONF = 1; /* Set LDONF */
}
}
}

```

6.1 リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'0026
CV3	H'002C
P	H'0100
B	H'FB80

ホームページとサポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問合せ先

<http://japan.renesas.com/inquiry>

csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
2.00	2006.09.01	全頁	日立版からルネサス版へのフォーマット変更

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご留意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。