

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。

H8/300L Super Low Power シリーズ

タイマ F による時計動作

要旨

38.4kHz のサブクロック (ϕ_w) を使用して、タイマ F による時計動作を行います。タイマ F 割り込みを 1s ごとに発生させ、RAM に設定した時計用のカウンタをカウントアップさせます。

59 分 59 秒までカウントアップすると、次のカウントで初期化します。

動作確認デバイス

H8/38024

目次

1. 仕様	2
2. 使用機能説明	2
3. 動作説明	10
4. ソフトウェア説明	12
5. フローチャート	17
6. プログラムリスト	19

1. 仕様

1. 32.768kHz のサブクロック (ϕ_w) を使用して、タイマ F による時計動作を行います。
2. タイマ F 割り込みを 1s ごとに発生させ、RAM に設定した時計用のカウンタをカウントアップさせます。
3. RAM に設定する時計用のカウンタは、秒カウント用 8 ビット、分カウント用 8 ビットとし、00 分 00 秒からカウントアップを開始し、59 分 59 秒までカウントアップすると、次のカウントで 00 分 00 秒に初期化して、再びカウントアップを継続します。
4. 初期設定終了後アクティブ (高速) モードからウォッチモードへ遷移し、タイマ F 割り込み要求によりサブアクティブモードへ遷移し、RAM に設定したカウンタをインクリメントして、再びウォッチモードへ遷移します。
5. 図 1 に本タスク例におけるモード遷移図を示します。

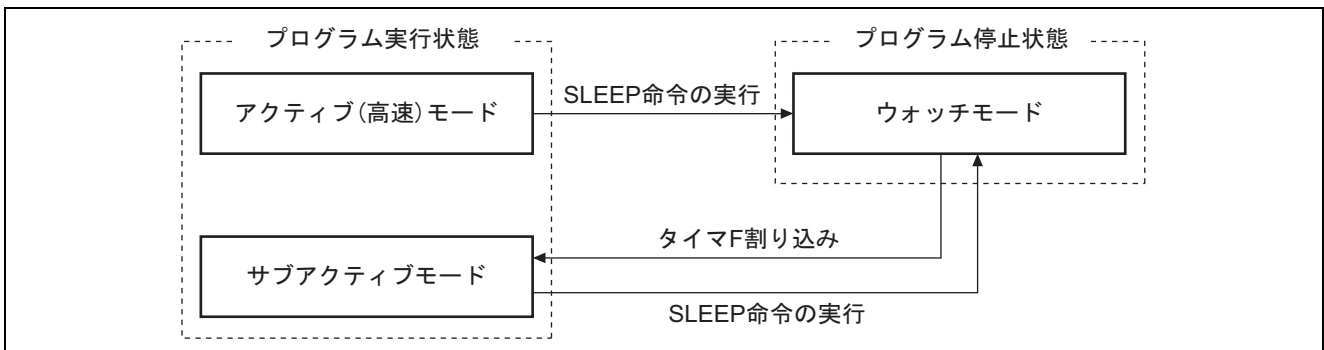


図 1 本タスク例におけるモード遷移図

2. 使用機能説明

1. 本タスク例では、タイマ F を使用して 1s ごとに RAM に設定したカウンタをインクリメントする時計動作を行います。以下にタイマ F の特長を示します。
 - システムクロック (ϕ) は、5MHz のクロックで、CPU および周辺機能を動作させるための基準クロックです。
 - プリスケアラ S (PSS) は、 ϕ を入力とする 13 ビットのカウンタで、1 サイクルごとにカウントアップします。
 - ϕ_w は、サブクロック発振器の出力クロック (32.768kHz) です。
 - プリスケアラ W (PSW) は、 ϕ_w を 4 分周したクロック ($\phi_w/4$) を入力とする 5 ビットのカウンタです。
 - 4 種類の内部クロック ($\phi/32$, $\phi/16$, $\phi/4$, $\phi_w/4$) と、外部クロックからの選択が可能です (外部イベントのカウントが可能)。
 - 1 本のコンペアマッチ信号により、TMOFH 端子にトグル出力します (トグル出力の初期値を設定可能)。
 - コンペアマッチ信号によるカウンタリセットが可能です。
 - 割り込み要因はコンペアマッチ $\times 1$ 要因、オーバフロー $\times 1$ 要因の計 2 要因です。
 - 2 本の独立した 8 ビットタイマ (タイマ FH, タイマ FL) としても動作が可能です (8 ビットモード時)。
 - 内部クロックとして $\phi_w/4$ を選択した場合、ウォッチモード、サブアクティブモード、スリープモードで動作可能です。
 - モジュールスタンバイモードにより、未使用時はモジュール単体でスタンバイモードに設定が可能です。

2. 図 2 に本タスク例で使用するタイマ F 16 ビットアウトプットコンペア機能のブロック図を示します。

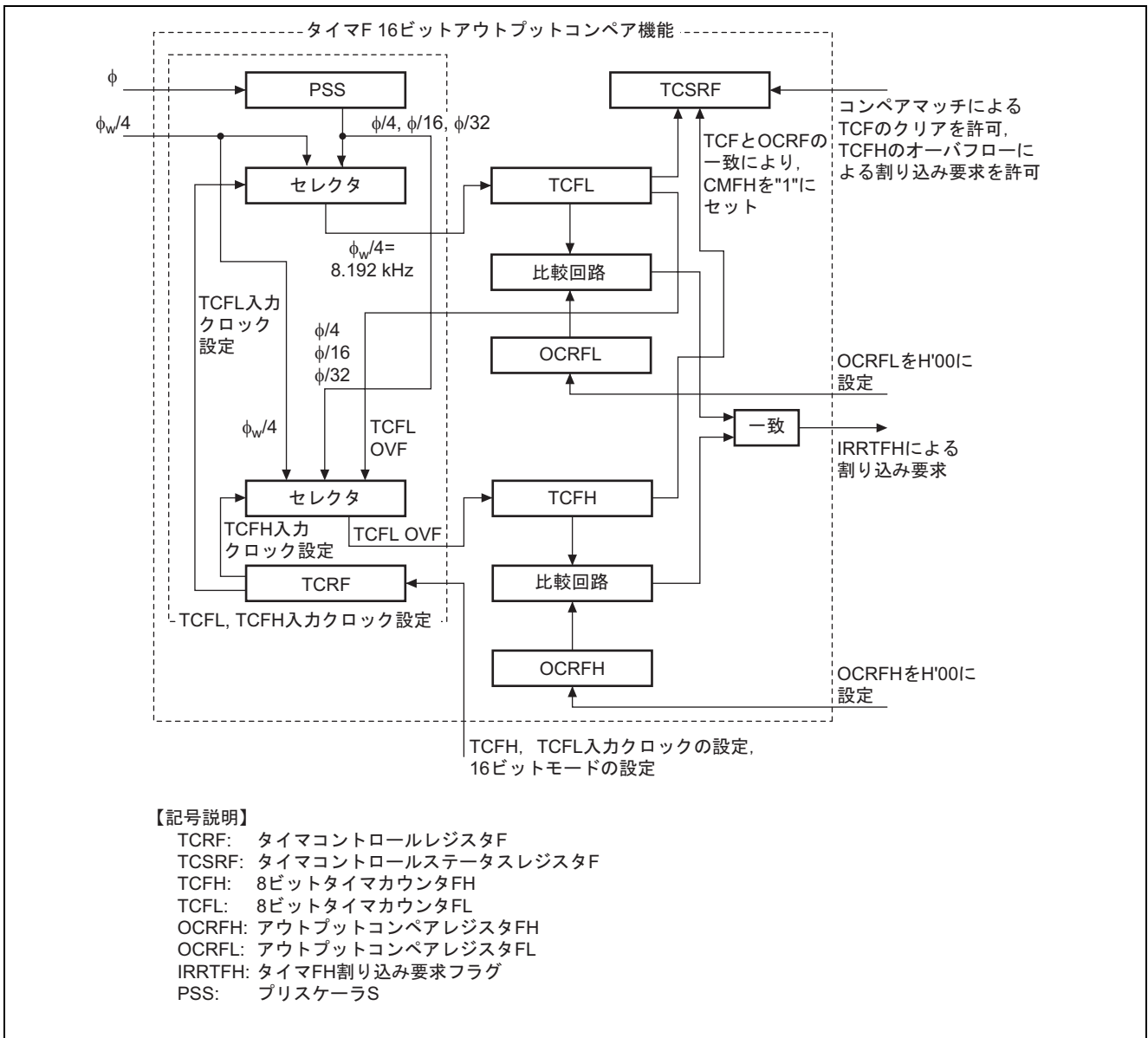


図 2 タイマ F 16 ビットアウトプットコンペア機能ブロック図

3. 表 1 にタイマ F の各機能について説明します。

表 1 タイマ F 機能

	説明
タイマコントロールレジスタ F (TCRF)	TCRF は 8 ビットのライト専用のレジスタです。16 ビットモード, 8 ビットモードの切り替え, 4 種類の内部クロックおよび外部イベントの選択, TMOFH, TMOFL 端子の出力レベルの設定を行います。リセット時, TCRF は H'00 に初期化されます。
タイマコントロールステータスレジスタ F (TCSRf)	TCSRf は 8 ビットのリード/ライト可能なレジスタです。カウンタクリアの選択, オーバフローフラグのセット, コンペアマッチフラグのセット, オーバフローによる割り込み要求の許可の制御を行います。
16 ビットタイマカウンタ F (TCF)	TCF は 16 ビットのリード/ライト可能なアップカウンタで, 8 ビットのタイマカウンタ (TCFH, TCFL) のカスケード接続により構成されています。上位 8 ビットを TCFH, 下位 8 ビットを TCFL とする 16 ビットカウンタとして使用できるほか, TCFH, TCFL を独立した 8 ビットカウンタとして使用することができます。TCFH, TCFL は, CPU からリード/ライト可能ですが, 16 ビットモードで使用する場合, CPU とのデータ転送はテンポラリレジスタ (TEMP) を介して行われます。リセット時, TCFH, TCFL はおののおの H'00 に初期化されます。TCRF の CKSH2 を "0" に設定すると, TCF は 16 ビットカウンタとして動作します。TCF の入力クロックは, TCRF の CKSL2 ~ CKSL0 により選択します。TCSRf の CCLRfH により, コンペアマッチ時に TCF をクリアすることができます。TCF がオーバフローすると, TCSRf の OVfH が "1" にセットされ, さらに IENR2 の IENTfH が "1" ならば CPU に割り込みを要求します。
8 ビットタイマカウンタ FH (TCFH)	TCFH は TCRF の CKSH2 を "1" に設定することにより独立した 8 ビットカウンタとして動作します。TCFH の入力クロックは, TCRF の CKSH2 ~ CKSH0 により選択します。TCSRf の CCLRfH により, コンペアマッチ時に TCFH をクリアすることができます。TCFH がオーバフローすると, TCSRf の OVfH が "1" にセットされます。このとき, TCSRf の OVIEH が "1" の場合, IRR2 の IRRTfH が "1" にセットされ, さらに IENR2 の IENTfH が "1" ならば CPU に割り込みを要求します。
8 ビットタイマカウンタ FL (TCFL)	TCFL は TCRF の CKSH2 を "1" に設定することにより独立した 8 ビットカウンタとして動作します。TCFL の入力クロックは, TCRF の CKSL2 ~ CKSL0 により選択します。TCSRf の CCLRfL により, コンペアマッチ時に TCFL をクリアすることができます。TCFL がオーバフローすると, TCSRf の OVfL が "1" にセットされます。このとき, TCSRf の OVIEL が "1" の場合, IRR2 の IRRTfL が "1" にセットされ, さらに IENR2 の IENTfL が "1" ならば CPU に割り込みを要求します。
16 ビットアウトプットコンペアレジスタ F (OCRf)	OCRf は 16 ビットのリード/ライト可能なレジスタ (OCRfH, OCRfL) により構成されています。上位 8 ビットを OCRfH, 下位 8 ビットを OCRfL とする 16 ビットレジスタとして使用できるほか, OCRfH, OCRfL を独立した 8 ビットレジスタとして使用することもできます。OCRfH, OCRfL は CPU からリード/ライト可能ですが, 16 ビットモードで使用する場合, CPU とのデータ転送は TEMP を介して行われます。リセット時, OCRfH, OCRfL はおののおの H'FF に初期化されます。TCRF の CKSH2 を "0" に設定すると, OCRf は 16 ビットレジスタとして動作します。OCRf の内容は, 常に TCF と比較されており, 両者の値が一致すると, TCSRf の CMfH が "1" にセットされます。また, 同時に IRR2 の IRRTfH も "1" にセットされます。このとき, IENR2 の IENTfH が "1" ならば CPU に割り込みを要求します。コンペアマッチによるトグル出力を TMOFH 端子より出力することができます。また, TCRF の TOLH により, 出力レベル ("High" / "Low") の設定が可能です。

	説明
8 ビットアウトプット コンペアレジスタ FH (OCRFB)	TCRF の CKSH2 を "1" に設定すると、OCRFB は 2 本の 8 ビットレジスタとして動作します。OCRFBH の内容は TCFB と、また OCRFL の内容は TCFL とそれぞれ個別に比較されます。OCRFBH と TCFB の値が一致すると、TCSRFB の CMFBH が "1" にセットされます。また同時に IRR2 の IRRFBH も "1" にセットされます。このとき、IENR2 の IENFBH が "1" ならば CPU に割り込みを要求します。コンペアマッチによるトグル出力を TMOFBH 端子より出力することができます。また、TCRF の TOLH により出力レベル ("High" / "Low") の設定が可能です。
8 ビットアウトプット コンペアレジスタ FL (OCRFL)	TCRF の CKSH2 を "1" に設定すると、OCRFB は 2 本の 8 ビットレジスタとして動作します。OCRFBH の内容は TCFB と、また OCRFL の内容は TCFL とそれぞれ個別に比較されます。OCRFL と TCFL の値が一致すると、TCSRFB の CMFL が "1" にセットされます。また同時に IRR2 の IRRFL も "1" にセットされます。このとき、IENR2 の IENFL が "1" ならば CPU に割り込みを要求します。コンペアマッチによるトグル出力を TMOFL 端子より出力することができます。また、TCRF の TOLL により出力レベル ("High" / "Low") の設定が可能です。
タイマ FH 割り込み要求フラグ (IRRFH)	16 ビットモードで TCF と OCRFB が一致したとき、8 ビットモードで TCFB と OCRFBH が一致したとき、また、IENFBH が "1" の状態で TCF および TCFB がオーバーフローしたときに IRRFBH は "1" にセットされます。IRRFH が "1" の状態で IRRFBH に "1" をライトしたとき、IRRFH は "0" にクリアされます。
タイマ FL 割り込み要求フラグ (IRRTFL)	8 ビットモードで TCFL と OCRFL が一致したとき、また、IENFL が "1" の状態で TCFL がオーバーフローしたときに IRRFL は "1" にセットされます。IRRTFL が "1" の状態で IRRFL に "1" をライトしたとき、IRRTFL は "0" にクリアされます。
タイマ FH 割り込みイネーブル (IENFBH)	タイマ FH のコンペアマッチまたはオーバーフロー割り込み要求の許可 / 禁止を制御します。
タイマ FL 割り込みイネーブル (IENFL)	タイマ FL のコンペアマッチまたはオーバーフロー割り込み要求の許可 / 禁止を制御します。
タイマ F イベント入力 (TMIF)	TCFL に入力するイベント入力端子です。
タイマ FH 出力 (TMOFBH)	タイマ FH のトグル出力端子です。
タイマ FL 出力 (TMOFL)	タイマ FL のトグル出力端子です。

4. タイマ FH 割り込み周期の設定方法について説明します。

本タスク例では、 ϕ_w に 32.768kHz を使用して、タイマ F をクロックタイムベースとして動作させています。

TCRF の CKSL2 を "1", CKSL1 を "1", CKSL0 を "1" に設定することにより TCF の入力クロックは $\phi_w/4$ に設定されます。 $\phi_w/4$ は、

$$\phi_w/4 = 32.768\text{kHz} / 4 = 8.192\text{kHz}$$

となります。したがって、TCF の入力クロック周期は、

$$1 / 8.192\text{kHz} = 122.07\mu\text{s}$$

となります。

ここで、OCRF を H'2000 に設定すると、TCF と OCRF の値が一致するまでの時間は、

$$H'2000 \times (1 / 8.192\text{kHz}) = 8192 \times 122.07\mu\text{s} = 1\text{s}$$

と計算されます。したがって、タイマ FH の割り込み周期 TFH を設定するための OCRF の設定値は、次式により計算できます。

$$\text{OCRF 設定値} = \text{TFH} / (1 / 8.192\text{kHz}) = \text{TFH} \times 8.192\text{kHz}$$

表 2 にタイマ FH 割り込み周期 TFH と OCRF 設定値例について示します。

表 2 タイマ FH 割り込み周期と OCRF 設定値例

T_{FH} (s)	計算方法	OCRF 設定値
0.125	$0.125\text{s} \times 8.192\text{kHz} = 1024$	H'0400
0.250	$0.250\text{s} \times 8.192\text{kHz} = 2048$	H'0800
0.500	$0.500\text{s} \times 8.192\text{kHz} = 4096$	H'1000
1.000	$1.000\text{s} \times 8.192\text{kHz} = 8192$	H'2000
2.000	$2.000\text{s} \times 8.192\text{kHz} = 16384$	H'4000

5. CPU とのインタフェースについて説明します。

TCF, OCRF は 16 ビットのリード/ライト可能なレジスタで構成されています。一方, CPU と内蔵周辺モジュール間のデータバスは, 8 ビット幅となっています。したがって, CPU が TCF, OCRF をアクセスする場合, 8 ビットのテンポラリレジスタ (TEMP) を介して行います。

16 ビットモードで TCF のリード/ライト, OCRF のライトを行うときは, 上位バイトのみや下位バイトのみのアクセスでは, データは正しく転送されません。必ず 16 ビットレジスタ全体に対して (バイトサイズの MOV 命令を 2 回連続して行う) リード/ライトを行い, かつ上位バイト, 下位バイトの順序で行います。

なお, 8 ビットモードでは特にアクセスの順序に制限はありません。

a. ライト時の動作

上位バイトのライトにより, 上位バイトのデータが TEMP に転送されます。

次に下位バイトのライトで, TEMP にあるデータが上位バイトのレジスタへ, 下位バイトのデータは直接下位バイトのレジスタへライトされます。

TCF に H'AA55 をライトするときの TCF のライト動作を図 3 に示します。

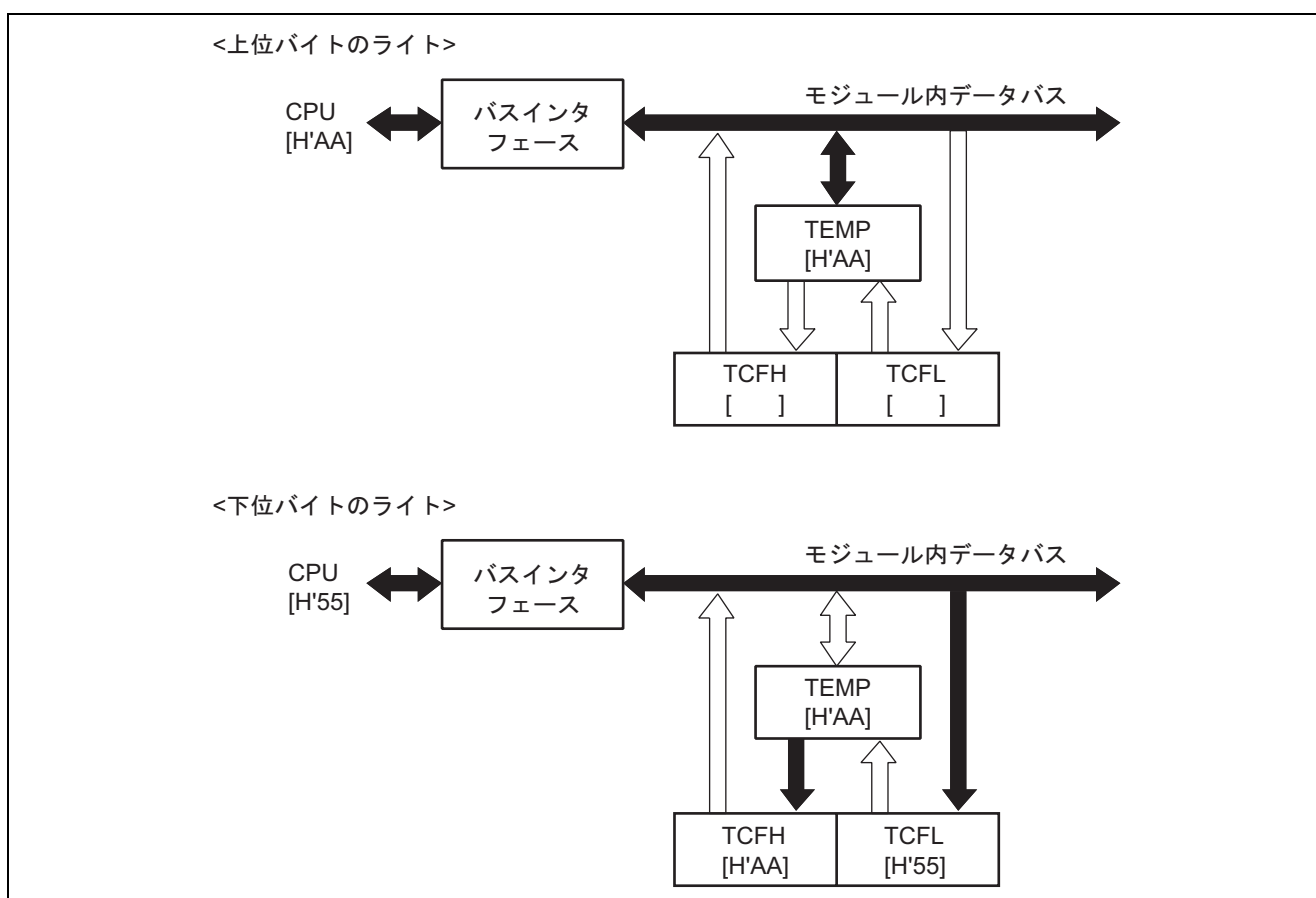


図 3 TCF のライト動作

b. リード時の動作

TCF の場合, 上位バイトのリードで, 上位バイトのデータは直接 CPU 転送され, 下位バイトのデータは TEMP に転送されます。

次に下位バイトのリードで, TEMP にある下位バイトのデータが CPU に転送されます。OCRF の場合, 上位バイトのリードで, 上位バイトのデータは直接 CPU に転送されます。下位バイトのリードで, 下位バイトのデータは直接 CPU に転送されます。

H'AAFF である TCF をリードしたときの TCF のリード動作を図 4 に示します。

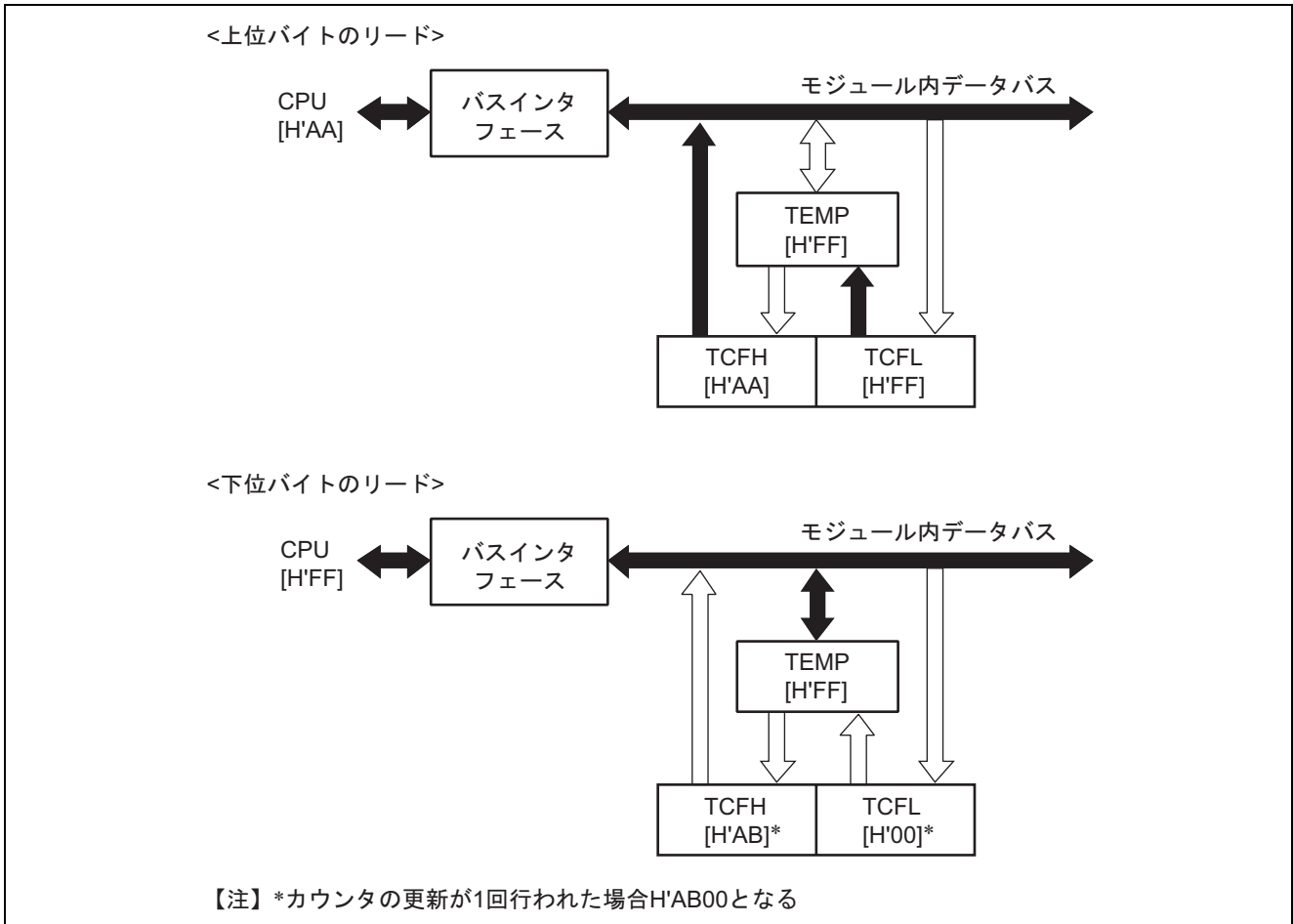


図 4 TCF のリード動作

6. タイマ F 使用上の注意事項

- タイマ F を 16 ビットタイマモードで動作中、次のような競合や動作が起こりますので注意してください。
- トグル出力は 16 ビットすべてが一致し、コンペアマッチ信号が発生したとき TMOFH 端子から出力されます。MOV 命令による TCRF のライトと、コンペアマッチ信号が同時に起こった場合、TCRF のライトによる TOLH のデータが TMOFH 端子に出力されます。TMOFL 端子は 16 ビットモード時には出力が不定となりますので使用しないで下さい。ポートとしてご使用下さい。
 - OCRFL のライトと、コンペアマッチ信号の発生が同時に起こった場合、コンペアマッチ信号は無効になります。ただし、ライトしたデータとカウンタ値がコンペアマッチする場合は、その時点でコンペアマッチ信号が発生します。コンペアマッチ信号は、TCFL のクロックに同期して出力されるので、クロックが停止している場合、コンペアマッチしていてもコンペアマッチ信号は発生しません。
 - コンペアマッチフラグは、16 ビットすべてが一致し、コンペアマッチ信号が発生したとき CMFH にセットされますが、CMFL についても下位 8 ビットについてのセット条件が満たされていればセットされます。
 - TCF がオーバーフローすると OVFH がセットされますが、OVFL についても下位 8 ビットがオーバーフローした時点で、セット条件が満たされていればセットされます。TCFL のライトと、オーバーフロー信号の出力が同時に起こった場合、オーバーフロー信号は出力されません。
 - アクティブモード、スリープモード時に TCF の内部クロックとして、 $\phi_w/4$ を選択して動作させる場合は、システムクロックと内部クロックとが非同期であるため、同期回路で同期をとっています。これにより、カウント周期は最大で $1/\phi$ (s) の誤差が生じます。誤差を生じさせないためには、サブアクティブモード、サブスリープモード、ウォッチモードにて動作させる必要があります。

7. 表 3 に本タスク例の機能割り付けを示します。

表 3 機能割り付け

機能	機能割り付け
TCRF	16 ビットモードの設定、TCFL の入力クロックの選択を行います。
TCSRFB	カウンタクリアの選択、オーバーフローフラグのセット、コンペアマッチフラグのセット、オーバーフロー割り込み要求の許可 / 禁止の制御を行います。
TCF	TCFH, TCFL から構成される 16 ビットのアップカウンタとして機能します。内部クロック ($\phi_w/4$) でカウントし、コンペアマッチすると IRTFH および CMFH を "1" にセットします。
OCRFB	OCRFB, OCRFL から構成される 16 ビットのレジスタで、OCRFB に設定した値と TCF の値が一致するとコンペアマッチ信号が発生します。
SYSCR1	低消費電力モードの制御を行う
SYSCR2	低消費電力モードの制御を行う
IRRTFH	タイマ FH 割り込み要求の有無を反映します。
IENFTFH	タイマ FH 割り込み要求の許可 / 禁止を制御します。
ϕ_w	サブクロック周波数で、32.768kHz を使用します。
TMA3	ウォッチモードへの移行を設定する。

3. 動作説明

1. タイマ F 16 ビットタイマモードの動作について説明します。

タイマ F は入力クロックが入るたびにカウントアップする 16 ビットのカウンタで、アウトプットコンペアレジスタ F に設定した値とタイマカウンタ F の値を常に比較しており、一致したタイミングでカウンタのクリア、割り込み要求、およびポートのトグル出力が可能です。また、2 本の独立した 8 ビットタイマとしても機能できます。

タイマコントロールレジスタ F (TCRF) の CKSH2 ビットを "0" に設定すると、タイマ F は 16 ビットのタイマとして動作します。

リセット直後、タイマカウンタ F (TCF) は H'0000 に、アウトプットコンペアレジスタ F (OCRF) は H'FFFF に、タイマコントロールレジスタ (TCRF)、タイマコントロールステータスレジスタ F (TCSR) は H'00 に初期化されます。カウンタは、外部イベント (TMIF) からの入力によりカウントアップを開始します。外部イベントのエッジ選択は、IRQ エッジセレクトレジスタ (IEGR) の IEG3 により設定します。

タイマ F の動作クロックは、TCRF の CKSL2 ~ CKSL0 ビットにより PSS の出力する 3 種類の内部クロック、 $\phi_w/4$ の内部クロック、または外部クロックを選択できます。

TCF と OCRF の内容は常に比較されており、両者の値が一致すると TCSR の CMFH が "1" にセットされます。このとき IENR2 の IENTFH が "1" ならば CPU に割り込みを要求し、同時に TMOFH 端子の出力をトグルします。また、TCSR の CCLR が "1" ならば TCF をクリアします。なお、TMOFH 端子の出力は、TCRF の TOLH により設定できます。

TCF がオーバフロー (H'FFFF → H'0000) すると、TCSR の OVFH が "1" にセットされます。このとき、TCSR の OVIEH と IENR2 の IENTFH がともに "1" ならば CPU に割り込みを要求します。

2. タイマ F の動作モードについて表 4 に示します。

表 4 タイマ F の動作モード

動作モード	リセット	アクティブ	スリープ	ウォッチ	サブアクティブ	サブスリープ	スタンバイ	モジュールスタンバイ
TCF	リセット	動作	動作	動作/停止*	動作/停止*	動作/停止*	停止	停止
OCRF	リセット	動作	保持	保持	動作	保持	保持	保持
TCRF	リセット	動作	保持	保持	動作	保持	保持	保持
TCSR	リセット	動作	保持	保持	動作	保持	保持	保持

【注】 * アクティブモード、スリープモード時に TCF の内部クロックとして、 $\phi_w/4$ を選択した場合、システムクロックと内部クロックが非同期であるため、同期回路で同期をとっています。これにより、カウント周期は最大で $1/\phi$ (s) の誤差が生じます。

サブアクティブモード、ウォッチモード、サブスリープモード時にカウンタを動作する場合、外部クロック、内部クロックに $\phi_w/4$ を選択して下さい。その他の内部クロックでは、カウンタが動作しません。

3. 図 5 に本タスク例の動作説明を示します。

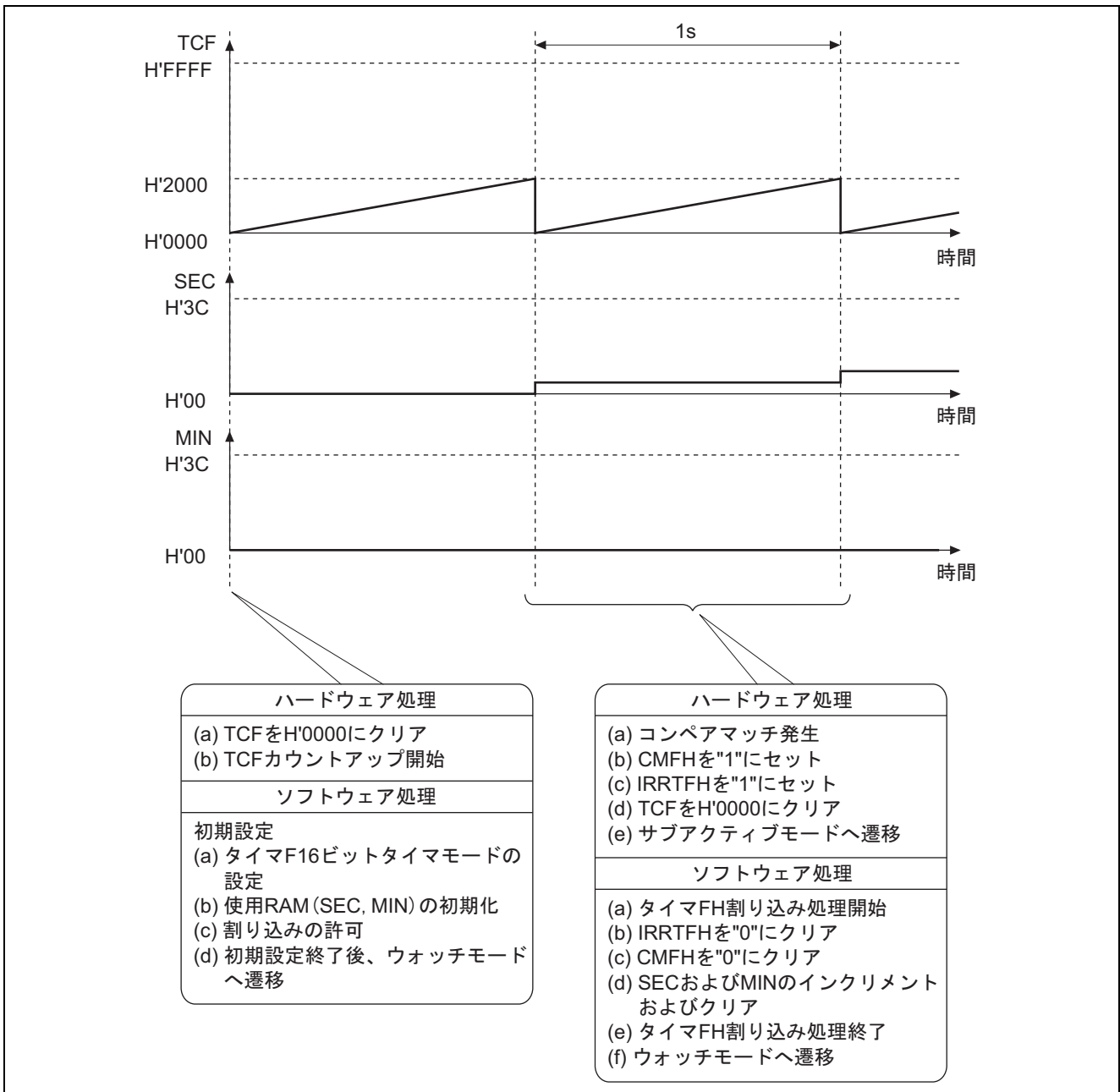


図 5 タイマ F による時計動作の動作説明

4. ソフトウェア説明

4.1 モジュール説明

本タスク例のモジュールを表 5 に示します。

表 5 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	使用 RAM, タイマ F の初期設定, 割り込みの許可, ウォッチモードへの遷移を行います。
タイマ F 割り込み処理ルーチン	tfint	IRRTFH, CMFH を"0"にクリア, レジスタの退避, RAM に設定した SEC および MIN のインクリメントまたはクリア, レジスタの復帰を行います。

4.2 引数の説明

本タスク例の引数を表 6 に示します。

表 6 引数の説明

引数名	機能	使用モジュール名	データ長	入出力
SEC	時計用のカウンタで, 秒をカウントします。	main, tfint	1 バイト	出力
MIN	時計用のカウンタで, 分をカウントします。	main, tfint	1 バイト	出力

4.3 使用内部レジスタ説明

本タスク例の使用内部レジスタを表 7 に示します。

表 7 使用内部レジスタ説明

レジスタ名	機能	アドレス	設定値
TCRF CKSH2 CKSH1 CKSH0	タイマコントロールレジスタ F (クロックセレクト H2~0) ~TCFH に入力するクロックを内部 4 種類, または TCFL の オーバフローから選択します。 : CKSH2 = 0, CKSH1 = 0, CKSH0 = 0 のとき, TCFL のオーバフローを選択 : CKSH2 = 0, CKSH1 = 0, CKSH0 = 1 のとき, TCFL のオーバフローを選択 : CKSH2 = 0, CKSH1 = 1, CKSH0 = 0 のとき, TCFL のオーバフローを選択 : CKSH2 = 0, CKSH1 = 1, CKSH0 = 1 は使用禁止 : CKSH2 = 1, CKSH1 = 0, CKSH0 = 0 のとき, 内部クロック $\phi/32$ を選択 : CKSH2 = 1, CKSH1 = 0, CKSH0 = 1 のとき, 内部クロック $\phi/16$ を選択 : CKSH2 = 1, CKSH1 = 1, CKSH0 = 0 のとき, 内部クロック $\phi/4$ を選択 : CKSH2=1, CKSH1=1, CKSH0=1 のとき, 内部クロック $\phi_w/4$ を選択	H'FFB6 ビット 6 ビット 5 ビット 4	CKSH2 = 0 CKSH1 = 0 CKSH0 = 0
CKSL2 CKSL1 CKSL0	タイマコントロールレジスタ F (クロックセレクト L2~0) ~TCFL に入力するクロックを内部 4 種類, または外部イベ ントから選択します。 : CKSL2 = 0, CKSL1 = 0, CKSL0 = 0 のとき, 外部イベントを選択 : CKSL2 = 0, CKSL1 = 0, CKSL0 = 1 のとき, 外部イベントを選択 : CKSL2 = 0, CKSL1 = 1, CKSL0 = 0 のとき, 外部イベントを選択 : CKSL2 = 0, CKSL1 = 1, CKSL0 = 1 は使用禁止 : CKSL2 = 1, CKSL1 = 0, CKSL0 = 0 のとき, 内部クロック $\phi/32$ を選択 : CKSL2 = 1, CKSL1 = 0, CKSL0 = 1 のとき, 内部クロック $\phi/16$ を選択 : CKSL2 = 1, CKSL1 = 1, CKSL0 = 0 のとき, 内部クロック $\phi/4$ を選択 : CKSL2 = 1, CKSL1 = 1, CKSL0 = 1 のとき, 内部クロック $\phi_w/4$ を選択	H'FFB6 ビット 2 ビット 1 ビット 0	CKSL2 = 1 CKSL1 = 1 CKSL0 = 1

レジスタ名		機能	アドレス	設定値
TCSR	OVFH	タイマコントロールステータスレジスタ F (タイマオーバーフローフラグ H) ~ TCF がオーバーフローしたか否かを示すステータスフラグ です。 : OVFH = 0 のとき, TCF がオーバーフローしていないこと を示す : OVFH = 1 のとき, TCF がオーバーフローしたことを示す	H'FFB7 ビット 7	0
	CMFH	タイマコントロールステータスレジスタ F (コンペアマッチフラグ H) ~ TCF と OCRF がコンペアマッチしたか否かを示すステ タスフラグです。 : CMFH = 0 のとき, TCF と OCRF がコンペアマッチ していないことを示す : CMFH = 1 のとき, TCF と OCRF がコンペアマッチ したことを示す	H'FFB7 ビット 6	0
	OVIEH	タイマコントロールステータスレジスタ F (タイマオーバーフローインタラプトイネーブル H) ~ TCF のオーバーフローが発生したとき, 割り込みの許可また は禁止を選択します。 : OVIEH = 0 のとき, TCF のオーバーフローによる 割り込み要求を禁止 : OVIEH = 1 のとき, TCF のオーバーフローによる 割り込み要求を許可	H'FFB7 ビット 5	1
TCSR	CCLR	タイマコントロールステータスレジスタ F (カウンタクリア H) ~ TCF と OCRF がコンペアマッチしたとき, TCF をクリア するかしないかを選択します。 : CCLR = "0" のとき, コンペアマッチによる TCF のクリアを禁止 : CCLR = "1" のとき, コンペアマッチによる TCF のクリアを許可	H'FFB7 ビット 4	1
TCFH		タイマカウンタ FH ~ 16 ビットタイマカウンタ F (TCF) の上位 8 ビットで, TCFL のオーバーフロー信号を入力クロックとする 8 ビットの アップカウンタです。	H'FFB8	H'00
TCFL		タイマカウンタ FL ~ 16 ビットタイマカウンタ F (TCF) の下位 8 ビットで, 内 部クロック $\phi_w/4$ を入力クロックとする 8 ビットのアップカウ ンタです。	H'FFB9	H'00
OCRFH		アウトプットコンペアレジスタ FH ~ 16 ビットアウトプットコンペアレジスタ (OCRF) の上位 8 ビットで, OCRF と TCF の値が一致するとコンペアマッチ 信号を発生。	H'FFBA	H'20
OCRFL		アウトプットコンペアレジスタ FL ~ 16 ビットアウトプットコンペアレジスタ (OCRF) の下位 8 ビットで, OCRF と TCF の値が一致するとコンペアマッチ 信号を発生。	H'FFBB	H'00

レジスタ名		機能	アドレス	設定値
IENR2	IENTFH	割り込み許可レジスタ 2 (タイマ FH 割り込みイネーブル) ~ タイマ FH 割り込み要求の許可 / 禁止を制御します。 : IENTFH = 0 のとき, タイマ FH 割り込み要求を禁止 : IENTFH = 1 のとき, タイマ FH 割り込み要求を許可	H'FFF4 ビット 3	1
IRR2	IRRTFH	割り込み要求レジスタ 2 (タイマ FH 割り込み要求フラグ) ~ タイマ FH 割り込み要求の有無を反映します。 : IRRTFH = 0 のとき, タイマ FH 割り込みが 要求されていないことを示す : IRRTFH = 1 のとき, タイマ FH 割り込みが 要求されていることを示す	H'FFF7 ビット 3	0
SYSCR1	SSBY	システムコントロールレジスタ 1 (ソフトウェアスタンバイ) ~ スタンバイモード, ウォッチモードへの遷移を指定します。 : SSBY = 0 のとき, アクティブモードで SLEEP 命令を 実行後, スリープモードに遷移。サブアクティブモード で SLEEP 命令を実行後, サブスリープモードに遷移。 : SSBY = 1 のとき, アクティブモードで SLEEP 命令を 実行後, スタンバイモードあるいはウォッチモードに 遷移。サブアクティブモードで SLEEP 命令を実行後, ウォッチモードに遷移。	H'FFF0 ビット 7	1
SYSCR1	STS2 STS1 STS0	システムコントロールレジスタ 1 (スタンバイタイマセレクト 2~0) ~ 特定の割り込みにより, スタンバイモード, ウォッチモード を解除し, アクティブモードに遷移する場合に, クロック が安定するまで CPU と周辺機能が待機する時間を指定しま す。動作周波数に応じて待機時間が発振安定時間以上となる ように指定してください。 : STS2 = 0, STS1 = 0, STS0 = 0 のとき, 待機時間は 8,192 ステート : STS2 = 0, STS1 = 0, STS0 = 1 のとき, 待機時間は 16,384 ステート : STS2 = 0, STS1 = 1, STS0 = 0 のとき, 待機時間は 1,024 ステート : STS2 = 0, STS1 = 1, STS0 = 1 のとき, 待機時間は 2,048 ステート : STS2 = 1, STS1 = 0, STS0 = 0 のとき, 待機時間は 4,096 ステート : STS2 = 1, STS1 = 0, STS0 = 1 のとき, 待機時間は 2 ステート : STS2 = 1, STS1 = 1, STS0 = 0 のとき, 待機時間は 8 ステート : STS2 = 1, STS1 = 1, STS0 = 1 のとき, 待機時間は 16 ステート	H'FFF0 ビット 6 ビット 5 ビット 4	STS2 = 0 STS1 = 0 STS0 = 0
	LSON	システムコントロールレジスタ 1 (ロースピードオンフラグ) ~ ウォッチモード解除時に, CPU の動作クロックをシステ ムクロックにするか, サブクロックにするかを選択します。 : LSON = 0 のとき, CPU の動作クロックはシステムク ロック : LSON = 1 のとき, CPU の動作クロックはサブクロック	H'FFF0 ビット 3	1

レジスタ名		機能	アドレス	設定値
SYSCR2	DTON	システムコントロールレジスタ 2 (ダイレクトトランスファオンフラグ) ~ アクティブ (高速) モード, アクティブ (中速) モード, サブアクティブモードの各モード間を, SLEEP 命令を実行 することにより直接遷移するか否かを指定します。 : DTON = 0 のとき, アクティブモードで SLEEP 命令を 実行したとき, スタンバイモード, ウォッチモード, またはスリープモードに遷移 : DTON = 1 のとき, アクティブ (高速) モードで SLEEP 命令を実行したとき, アクティブ (中速) モード (SSBY = 0, MSON = 1, LSON = 0 のとき), またはサブ アクティブモード (SSBY = 1, TMA3 = 1, LSON = 1 の とき) に直接遷移	H'FFF1 ビット 3	0
	MSON	システムコントロールレジスタ 2 (ミドルスピードオンフラグ) ~ スタンバイモード, ウォッチモード, スリープモード解除 後, アクティブ (高速) モードで動作させるか, アクティブ (中速) モードで動作させるかを選択します。 : MSON = 0 のとき, アクティブ (高速) モードで動作 : MSON = 1 のとき, アクティブ (中速) モードで動作	H'FFF1 ビット 2	0
TMA	TMA3	タイマモードレジスタ A (タイマモードレジスタ A3) ~ タイマカウンタ A に入力するクロックソースを選択しま す。 : TMA3 = 0 のとき, スタンバイモードに遷移可能 : TMA3 = 1 のとき, アクティブ (高速), アクティブ (中速), サブアクティブモード, ウォッチモード, サブスリープモードに遷移可能	H'FFB0 ビット 3	1

4.4 使用 RAM 説明

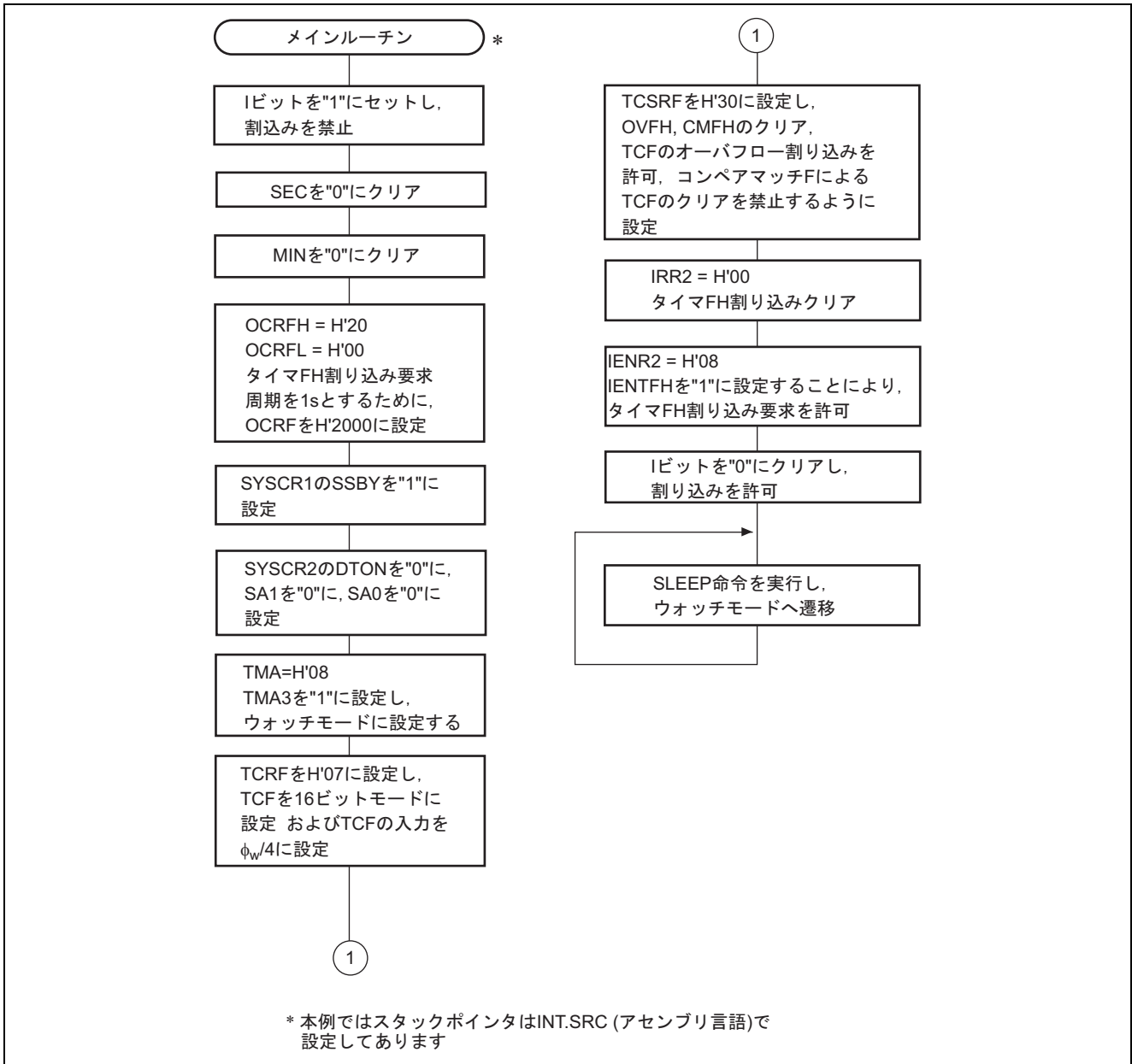
本タスク例の使用 RAM を表 8 に示します。

表 8 使用 RAM 説明

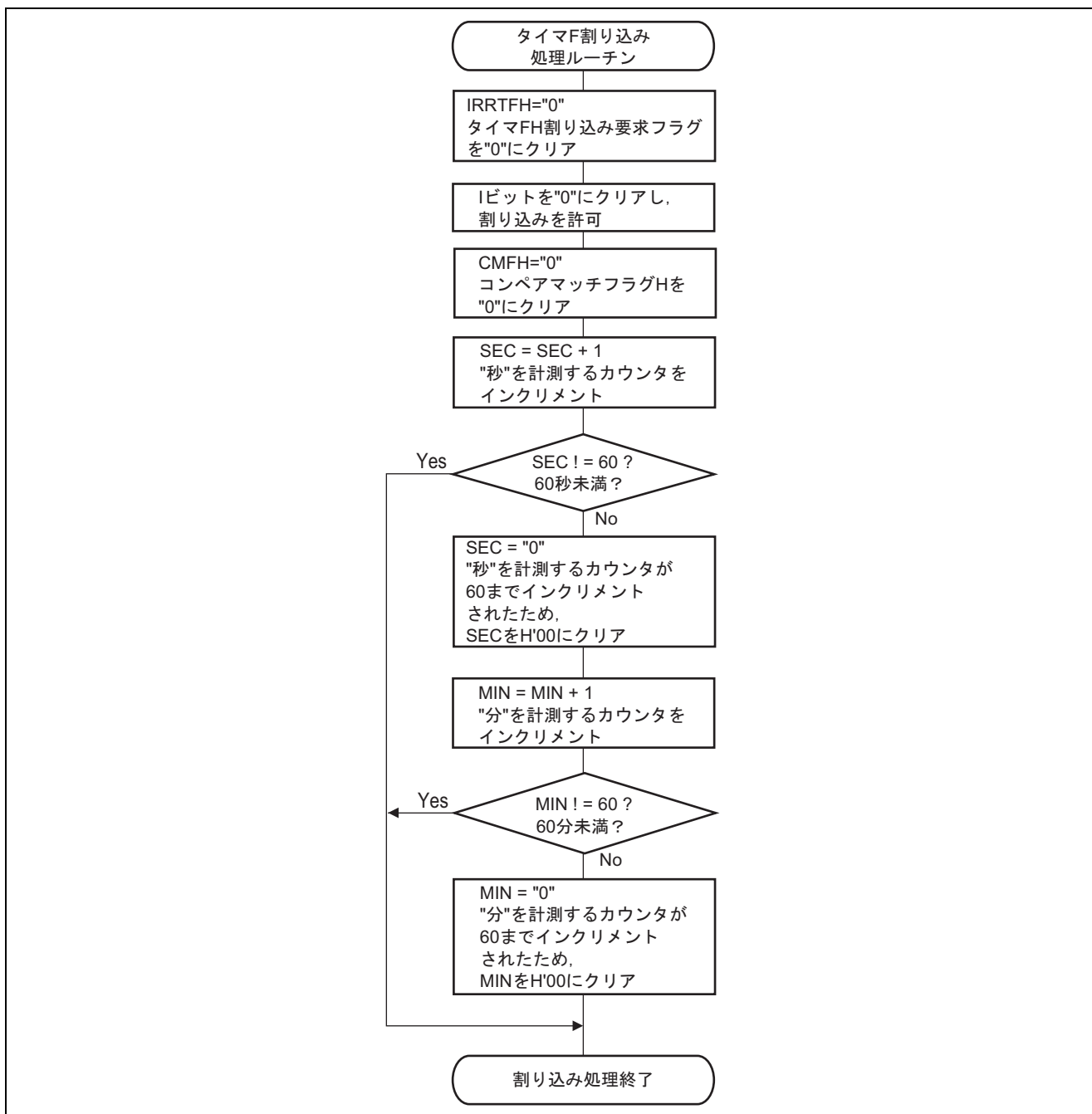
ラベル名	機能	アドレス	使用モジュール名
SEC	時計用のカウンタで, 秒をカウントします。	H'FB80	メインルーチン タイマ F 割り込み処理ルーチン
MIN	時計用のカウンタで, 分をカウントします。	H'FB81	メインルーチン タイマ F 割り込み処理ルーチン

5. フローチャート

1. メインルーチン



2. タイマ F 割り込み処理ルーチン



6. プログラムリスト

6.1 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

```

```

/*****
/*
/* H8/300L Super Low Power Series
/*   -H8/38024 Series-
/* Application Note
/*
/* 'Timer F -Clock Time Base-'
/*
/* Function
/* : Timer F
/*
/* External Clock :   10MHz
/* Internal Clock  :   5MHz
/* 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 TMA        *(volatile unsigned char *)0xFFB0    /* Timer Mode Register A
#define TCA        *(volatile unsigned char *)0xFFB1    /* Timer Counter A
#define TCRF       *(volatile unsigned char *)0xFFB6    /* Timer Control Register F
#define TCRF_BIT   (*(struct BIT *)0xFFB6)              /* Timer Control Register F
#define TOLH       TCRF_BIT.b7                          /* Toggle Output Level F
#define CKSH2      TCRF_BIT.b6                          /* Clock Select H2
#define CKSH1      TCRF_BIT.b5                          /* Clock Select H1

```

```

#define CKSH0      TCRF_BIT.b4                /* Clock Select H0                */
#define TCSRFB    *(volatile unsigned char *)0xFFB7 /* Timer Control Status Register F */
#define TCSRFB_BIT (*(struct BIT *)0xFFB7) /* Timer Control Status Register F */
#define OVFB      TCSRFB_BIT.b7              /* Timer Overflow Flag H          */
#define CMFB      TCSRFB_BIT.b6              /* Compare Match Flag H          */
#define OVIEH     TCSRFB_BIT.b5              /* Timer Overflow Interrupt Enable */
#define CCLR3     TCSRFB_BIT.b4              /* Output Select 3                */
#define OCRFB     *(volatile unsigned char *)0xFFBA /* Output Compare Register FH     */
#define OCRFL     *(volatile unsigned char *)0xFFBB /* Output Compare Register FH     */
#define TMG       *(volatile unsigned char *)0xFFBC /* Output Compare Register FH     */
#define SYSCR1    *(volatile unsigned char *)0xFFF0 /* System Control Register 1      */
#define SYSCR1_BIT (*(struct BIT *)0xFFF0) /* System Control Register 1      */
#define SSBY      SYSCR1_BIT.b7              /* Software Standby                */
#define STS2      SYSCR1_BIT.b6              /* Standby Timer Select 2          */
#define STS1      SYSCR1_BIT.b5              /* Standby Timer Select 1          */
#define STS0      SYSCR1_BIT.b4              /* Standby Timer Select 0          */
#define LSON      SYSCR1_BIT.b3              /* Low Speed On Flag                */
#define MA1       SYSCR1_BIT.b1              /* Active Mode Clock Select 1      */
#define MA0       SYSCR1_BIT.b0              /* Active Mode Clock Select 0      */
#define SYSCR2    *(volatile unsigned char *)0xFFF1 /* System Control Register 2      */
#define SYSCR2_BIT (*(struct BIT *)0xFFF1) /* System Control Register 2      */
#define NESEL     SYSCR2_BIT.b4              /* Noise Elimination Sampling Frequency
Select                               */
#define DTON      SYSCR2_BIT.b3              /* Direct Transfer On Flag          */
#define MSON      SYSCR2_BIT.b2              /* Middle Speed On Flag            */
#define SA1       SYSCR2_BIT.b1              /* Subactive Mode Clock Select 1   */
#define SA0       SYSCR2_BIT.b0              /* Subactive Mode Clock Select 0   */
#define IENR1_BIT (*(struct BIT *)0xFFF3) /* Interrupt Enable Register 1     */
#define IENTA     IENR1_BIT.b7              /* Timer A Interrupt Enable        */
#define IENR2     *(volatile unsigned char *)0xFFF4 /* Interrupt Enable Register 2     */
#define IENR2_BIT (*(struct BIT *)0xFFF4) /* Interrupt Enable Register 2     */
#define IENDT     IENR2_BIT.b7              /* Direct Transfer Interrupt Enable */
#define IENTFH    IENR2_BIT.b3              /* Timer FH Interrupt Enable       */
#define IENTG     IENR2_BIT.b4              /* Timer FH Interrupt Enable       */
#define IRR1_BIT  (*(struct BIT *)0xFFF6) /* Interrupt Request Register 1    */
#define IRR1A     IRR1_BIT.b7              /* Timer A Interrupt Request Flag   */
#define IRR2     *(volatile unsigned char *)0xFFF7 /* Interrupt Request Register 2    */
#define IRR2_BIT  (*(struct BIT *)0xFFF7) /* Interrupt Request Register 2    */
#define IRRDT     IRR2_BIT.b7              /* Direct Transfer Interrupt
Request Flag                          */
#define IRRTFH    IRR2_BIT.b3              /* Timer FH Interrupt Request Flag  */
#define IRRTG     IRR2_BIT.b4              /* Timer FH Interrupt Request Flag  */
#define CKSTPR1  *(volatile unsigned char *)0xFFFA /*                               */
#define CKSTPR2  *(volatile unsigned char *)0xFFFB /*                               */

#pragma interrupt      (tfint)
/*****
/*      Function define
*****/
extern void INIT ( void ); /* SP Set */
void      main ( void );
void      tfint ( void );

```

```

/*****
/* RAM define
/*****
unsigned char    SEC;
unsigned char    MIN;

/*****
/* 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) = {
    tfint                                                /* 1E Timer F Interrupt
};

#pragma section                                /* P
/*****
/* Main Program
/*****
void    main ( void )
{
    set_imask_ccr(1);                                /* Interrupt Disable

    SEC = 0;
    MIN = 0;

    OCRFH = 0x20;                                    /* Initialize Compare MatchFH Value
    OCRFL = 0x00;                                    /* Initialize Compare MatchFL Value

    SYSCR1 = 0x8F;                                    /* Set SYSCR 1
    SYSCR2 = 0xE0;                                    /* Set SYSCR 2

    TMA = 0x08;                                       /* Initialize TCA Overflow Period

    TCRF = 0x07;                                       /* TMOFH Highlevel Output
    TCSRFB = 0x30;                                       /* Initialize Overflow Interrupt

    IRR2 = 0x00;
    IENR2 = 0x08;                                       /* Timer A Interrupt Enable
    set_imask_ccr(0);                                    /* Interrupt Enable

    while(1){
        sleep();                                       /* Transition to Sleep Mode
    }
}

```

```

/*****
/* Timer F Interrupt
/*****
void tfint ( void )
{
    IRRTFH = 0;                /* Clear Timer F Interrupt Request */
    set_imask_ccr(0);         /* Interrupt Enable
                               */

    CMFH = 0;                /* Clear Compare Match Flag H
                               */
    SEC++;                    /* Increment Second Counter
                               */

    if ( SEC == 60 ){
        SEC = 0;             /* Initialize Second Counter
                               */
        MIN++;              /* Increment Minute Counter
                               */
        if ( MIN == 60 ){
            MIN = 0;         /* Initialize Minute Counter
                               */
        }
    }
}

```

リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'001E
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
1.00	2003.12.19	—	初版発行
2.00	2006.09.15	1 ~ 3, 7, 14, 17	内容変更

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、慎重に制作したのですが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。