

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い申し上げます。

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が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

サブアクティブモードへの遷移

要旨

サブアクティブモードへの遷移を行ないます。

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	9
6. プログラムリスト	12

1. 仕様

1. サブアクティブモードへの遷移を行ないます。
2. アクティブモードで、SLEEP 命令を実行することによりスリープモードに遷移します。
3. スリープモードで、 $\overline{\text{IRQ0}}$ 端子に接続したスイッチ入力が入オンになると IRQ0 割り込みが発生し、スリープモードは解除され、サブアクティブモードに遷移します。
4. サブアクティブモードでは、0.5s ごとにタイマ A 割り込み要求が発生し、タイマ A 割り込み処理の中で LED の制御を行ないます。LED は 0.5s ごとに点灯、消灯を繰り返します。
5. サブアクティブモードで、 $\overline{\text{IRQ1}}$ 端子に接続したスイッチ入力が入オンになると IRQ1 割り込みが発生し、IRQ1 割り込み終了後に、SLEEP 命令を実行しアクティブモードに直接遷移します。
6. 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。
7. 図 1 に $\overline{\text{IRQ0}}$ 、 $\overline{\text{IRQ1}}$ 端子に接続するスイッチの接続例を示します。

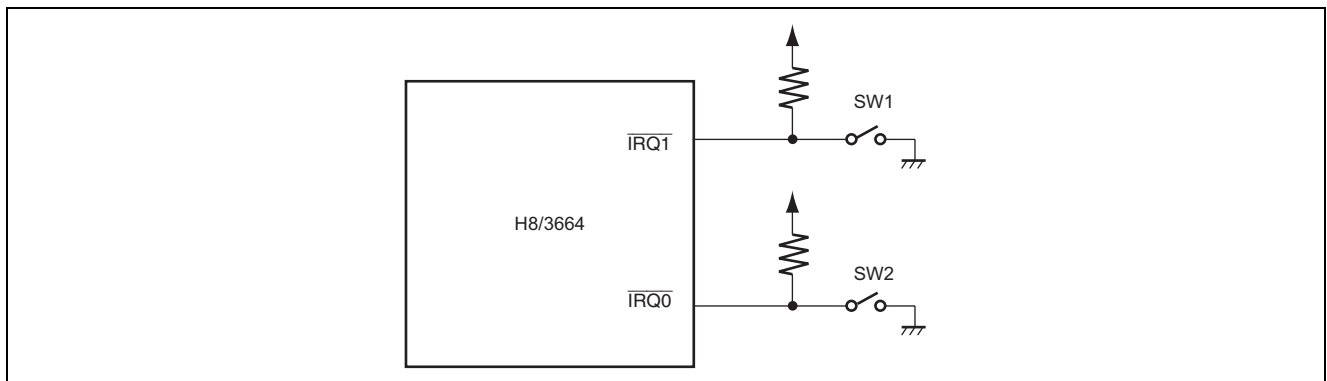


図 1 サブアクティブモードへの遷移におけるスイッチ接続例

2. 使用機能説明

- 本タスク例では、低消費電力モードのサブアクティブモードへの遷移を行ないます。図 2 にサブアクティブモードへのモード遷移図を示します。以下にサブアクティブモードの機能の説明を示します。
 - スリープモードで割り込み (タイマ A, タイマ V, タイマ W, IRQ3 ~ IRQ0, WKP5 ~ WKP0, ウォッチドッグタイマ, SCI3, IIC, A/D 変換器) が発生したとき, SYSCR2 の LSON が 1 ならば, サブアクティブモードに遷移します。また, サブスリープモードで割り込み (タイマ A, タイマ W, IRQ3 ~ IRQ0, WKP5 ~ WKP0, ウォッチドッグタイマ, IIC) が発生したとき, サブアクティブモードに遷移します。
 - CCR の I ビットが "1" の場合, または割り込みイネーブルレジスタ 1 により当該割り込みの受け付けが禁止されている場合は, サブアクティブモードに遷移しません。
 - SLEEP 命令による解除は, SYSCR1 の SSBY が X, SYSCR2 の DTON が 1, SMSEL が 0, LSON が 0 の状態で SLEEP 命令を実行することによってアクティブモードへ直接遷移します。
 - RES 端子による解除は, RES 端子を Low レベルにするとリセット状態に遷移し, サブアクティブモードは解除されます。発振安定時間経過後, RES 端子を High レベルにすると, CPU はリセット例外処理を開始します。なお, システムクロックの発振開始と同時に LSI 全体にシステムクロックが供給されます。
 - RES 端子は, 必ずシステムクロックの発振が安定するまで, Low レベルを保持してください。
 - サブアクティブモード解除後の発振安定の待機時間は, SYSCR1 の STS2 ~ STS0 により設定します。
 - 本タスク例では, 動作周波数に 16MHz を使用し, 待機ステート数を 131,072 ステートに設定します。(発振安定の待機時間: 8.2ms)
 - CPU がプログラムを実行している動作モードにはアクティブモード, サブアクティブモードの 2 つのモードがあります。この 2 つの動作モードの間で, プログラムを停止する事なく遷移することを直接遷移と呼びます。直接遷移は SYSCR2 の DTON を 1 にセットし, SLEEP 命令を実行することにより可能です。アクティブモード, サブアクティブモードで動作周波数を変更する場合にも有効です。遷移後は直接遷移割り込み例外処理を開始します。なお, 割り込みイネーブルレジスタ 1 により直接遷移割り込みが禁止されている場合は, スリープモードまたはサブスリープモードへ遷移します。また, CCR の I ビットを "1" の状態で直接遷移を行なうとスリープモードまたはサブスリープモードに遷移し, 遷移後のモードから割り込みによる解除は不可能となります。
 - サブアクティブモードからアクティブモードへの直接遷移は, サブアクティブモードで SYSCR1 の SSBY を X, SYSCR2 の LSON を 0, SMSEL を 0, DTON を 1 にセットした状態で SLEEP 命令を実行すると SYSCR1 の STS2 ~ STS0 により設定された待機時間を経過した後アクティブモードに直接遷移します。

【注】 X は 1, 0 どちらでも良い

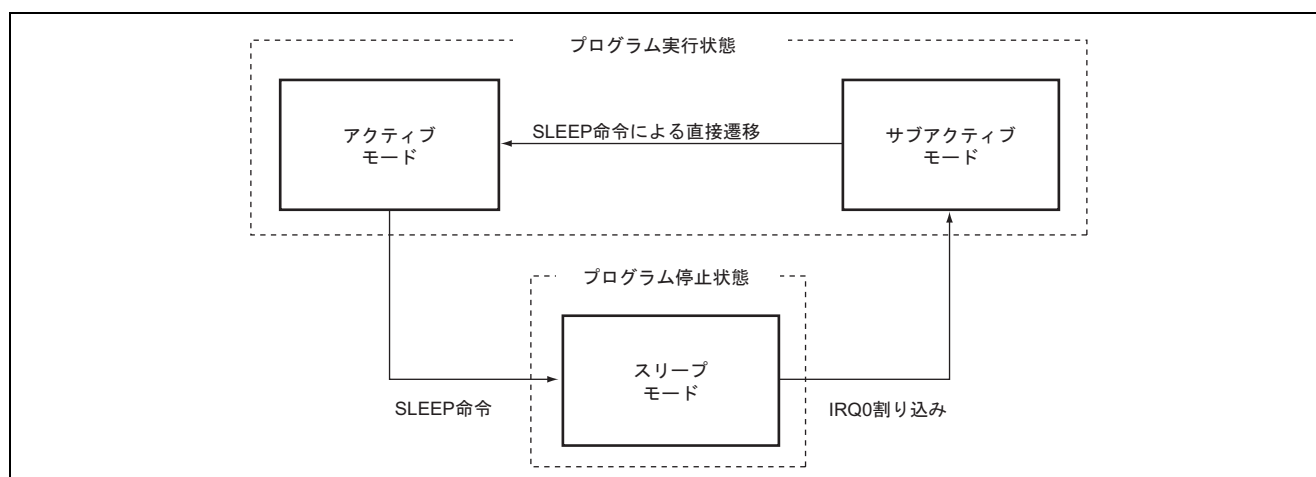


図 2 サブアクティブモードへの遷移におけるモード遷移図

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、サブアクティブモードへの遷移を行ないます。

表 1 機能割り付け

機能	機能割り付け
SYSCR1	低消費電力モードの制御を行なう
SYSCR2	低消費電力モードの制御を行なう
PCR7	P74 出力端子機能の設定
PDR7	P74 出力端子のデータの格納
P74	LED 出力
IRQ0	スイッチ 1 入力
IRRI0	タイマ A 割り込み要求の有無を反映
IENI0	タイマ A 割り込み要求の許可、または禁止を設定
IRQ1	スイッチ 2 入力

3. 動作原理

図3に動作原理を示します。図3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サブアクティブモードへの遷移を行ないます。

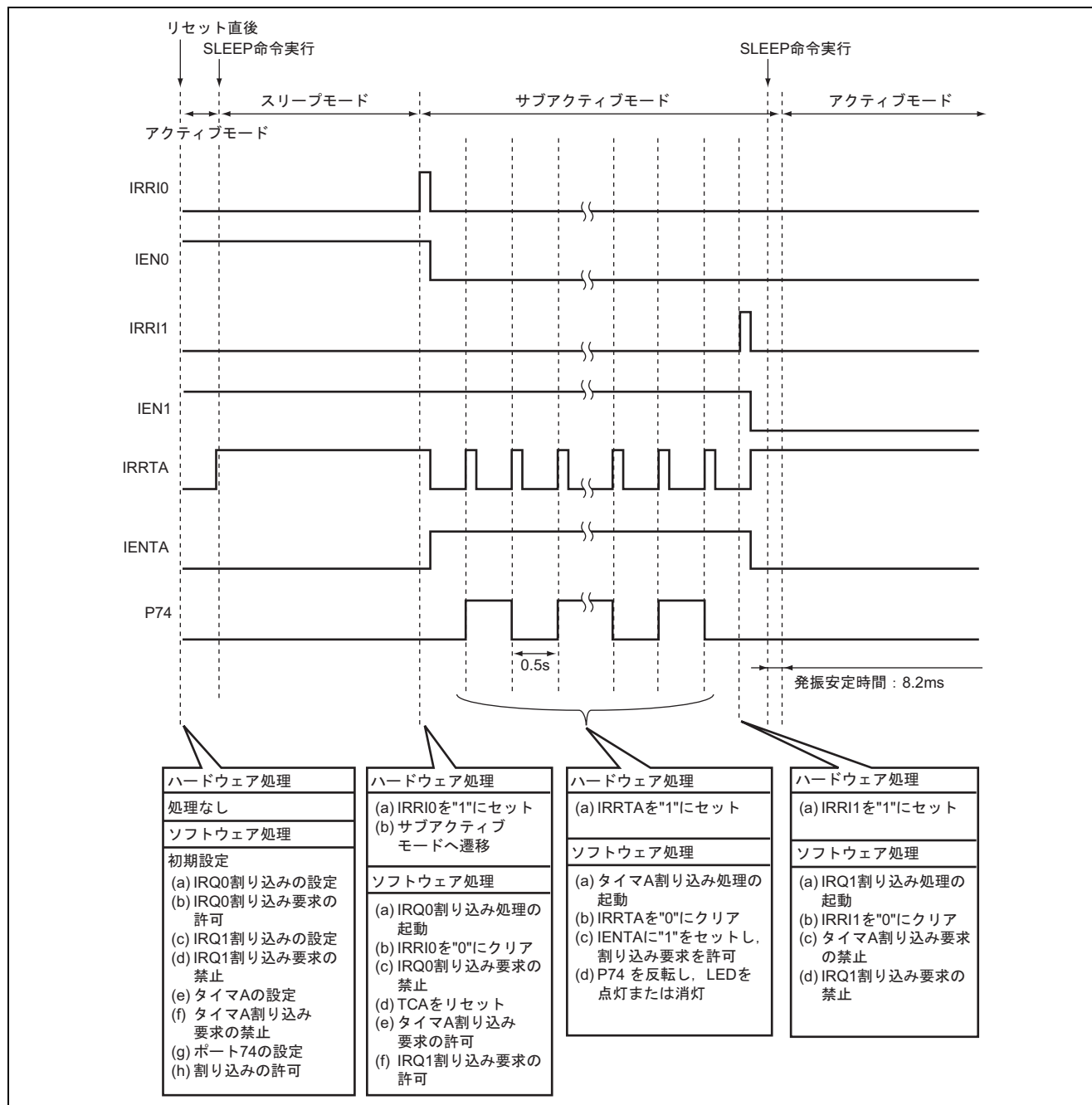


図3 サブアクティブモードへの遷移の動作原理

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	IRQ0 割り込みの設定 ,IRQ1 割り込みの設定 ,タイマ A 割り込みの設定 ,ポート 7 の設定 ,割り込みの許可 ,スリープモード ,アクティブモードへの遷移を行なう
スイッチ 1 オン	IRQ0	IRQ0 割り込み処理ルーチンで ,IRQ0 割り込み要求の禁止を行なう
スイッチ 2 オン	IRQ1	IRQ1 割り込み処理ルーチンで ,SWONF のセット ,タイマ A 割り込み要求の禁止 ,IRQ1 割り込み要求の禁止を行なう
LED 制御	taint	タイマ A 割り込み処理ルーチンで ,割り込み要求の許可 ,LED の制御を行なう
直接遷移	dtint	直接遷移割り込み処理ルーチンで ,直接遷移割り込み要求フラグのクリアを行なう

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
TMA		タイマモードレジスタ A : TMA = H'19 のとき, タイマ A 機能を時計用タイムベース機能に, TCA のオーバフロー周期を 0.5s に設定	H'FFA6	H'19
TCA		タイマカウンタ A : PSW 出力クロックを入力とし, 時計用タイムベース機能により 0.5s でオーバフローする 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	1
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 端子を出力端子に設定	H'FFEA ビット 4	1
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) : SSBY = "0" のとき, アクティブモードで SLEEP 命令実行後スリープモード, あるいはサブスリープモードに遷移。サブアクティブモードで SLEEP 命令実行後アクティブモードに直接遷移	H'FFF0 ビット 7	0
	STS2 STS1 STS0	システムコントロールレジスタ 1 (スタンバイタイムセレクト 2, 1, 0) : STS2 = "1", STS1 = "0", STS0 = ""0 のとき, 待機ステート数 131.072 ステートに設定	H'FFF0 ビット 6 ビット 5 ビット 4	STS2 = 1 STS1 = 0 STS0 = 0
	SMSSEL	システムコントロールレジスタ 2 (スリープモード選択) : SMSSEL = "0" のとき, スリープ命令実行後, スリープモードを設定	H'FFF1 ビット 7	0
	LSON	システムコントロールレジスタ 2 (ロースピードオンフラグ) : LSON = "1" のとき, スリープ命令実行後, スリープモード, サブスリープモードサブアクティブモード (直接遷移) の各 1 モードを設定	H'FFF1 ビット 6	1
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) : DTON = "0" のとき, スリープ命令実行後, スリープモード, サブスリープモード, スタンバイモードの各 1 モードを設定	H'FFF1 ビット 3	0
	MA2 MA1 MA0	システムコントロールレジスタ 2 (アクティブモードクロックセレクト 2~0) : MA2 = "0", MA1 = "1", MA0 = "1", のとき, ϕ OSC をクロックに設定	H'FFF1 ビット 4 ビット 3 ビット 2	MA2 = 0 MA1 = 1 MA0 = 1
	SA1 SA0	システムコントロールレジスタ 2 (サブアクティブモードクロックセレクト 1, 0) : SA1 = "0", SA0 = "0" のとき, サブアクティブモードの CPU の動作クロックを ϕ W/8 に設定	H'FFF1 ビット 1 ビット 0	SA1 = 0 SA0 = 0
PMR1	IRQ1	ポートモードレジスタ 1 (IRQ1I/O ポートセレクト) : IRQ1 = "0" のとき, $\overline{\text{IRQ1}}$ 端子を汎用入出力ポートに設定 : IRQ1 = "1" のとき, $\overline{\text{IRQ1}}$ 端子を $\overline{\text{IRQ1}}$ 入力端子に設定	H'FFE0 ビット 1	1
	IRQ0	ポートモードレジスタ 1 (IRQ0I/O ポートセレクト) : IRQ0 = "0" のとき, $\overline{\text{IRQ0}}$ 端子を汎用入出力ポートに設定 : IRQ0 = 1 のとき, $\overline{\text{IRQ0}}$ 端子を $\overline{\text{IRQ0}}$ 入力端子に設定	H'FFE0 ビット 0	1

表 3 使用内部レジスタ説明 (つづき)

レジスタ名		機能	アドレス	設定値
IEGR1	IEG1	割り込みエッジセレクトレジスタ 1 (IRQ1 エッジセレクト) : IEG1 = "0" のとき, $\overline{\text{IRQ1}}$ 端子入力の立ち下がりエッジを検出 : IEG1 = "1" のとき, $\overline{\text{IRQ1}}$ 端子入力の立ち上がりエッジを検出	H'FFF2 ビット 1	1
	IEG0	割り込み許可レジスタ 1 (IRQ0 エッジセレクト) : IEN0 = "0" のとき, $\overline{\text{IRQ0}}$ 端子入力の立ち下がりエッジを検出 : IEN0 = "1" のとき, $\overline{\text{IRQ0}}$ 端子入力の立ち上がりエッジを検出	H'FFF2 ビット 0	1
IENR1	IENDT	割り込みイネーブルレジスタ 1 (直接遷移割り込みイネーブル) : IENDT = "0" のとき, 直接遷移による割り込み要求を禁止 : IENDT = "1" のとき, 直接遷移による割り込み要求を許可	H'FFF4 ビット 7	1
	IENTA	割り込みイネーブルレジスタ 1 (タイマ A 割り込みイネーブル) : IENTA = "0" のとき, タイマ A 割り込み要求を禁止 : 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
	IEN1	割り込みイネーブルレジスタ 1 (IRQ1 割り込み要求イネーブル) : IEN1 = "0" のとき, $\overline{\text{IRQ1}}$ 端子による割り込み要求を禁止 : IEN1 = "1" のとき, $\overline{\text{IRQ1}}$ 端子による割り込み要求を許可	H'FFF4 ビット 1	1
	IEN0	割り込みイネーブルレジスタ 1 (IRQ0 割り込み要求イネーブル) : IEN0 = "0" のとき, $\overline{\text{IRQ0}}$ 子による割り込み要求を禁止 : IEN0 = "1" のとき, $\overline{\text{IRQ0}}$ 端子による割り込み要求を許可	H'FFF4 ビット 0	1
IRR1	IRRDT	割り込みフラグレジスタ 1 (直接遷移割り込み要求フラグ) : IRRDT = "0" のとき, 直接遷移による割り込みが要求されていない : IRRDT = "1" のとき, 直接遷移による割り込みが要求されている	H'FFF6 ビット 7	0
	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
	IRRI1	割り込みフラグレジスタ 1 (IRQ1 割り込み要求フラグ) : IRR11 = "0" のとき, $\overline{\text{IRQ1}}$ 端子入力による割り込みが要求されていない : IRR11 = "1" のとき, $\overline{\text{IRQ1}}$ 端子入力による割り込みが要求されている	H'FFF6 ビット 1	0
	IRRI0	割り込みフラグレジスタ 1 (IRQ0 割り込み要求フラグ) : IRR10 = "0" のとき, $\overline{\text{IRQ0}}$ 端子入力による割り込みが要求されていない : IRR10 = "1" のとき, $\overline{\text{IRQ0}}$ 端子入力による割り込みが要求されている	H'FFF6 ビット 0	0

4.4 使用 RAM 説明

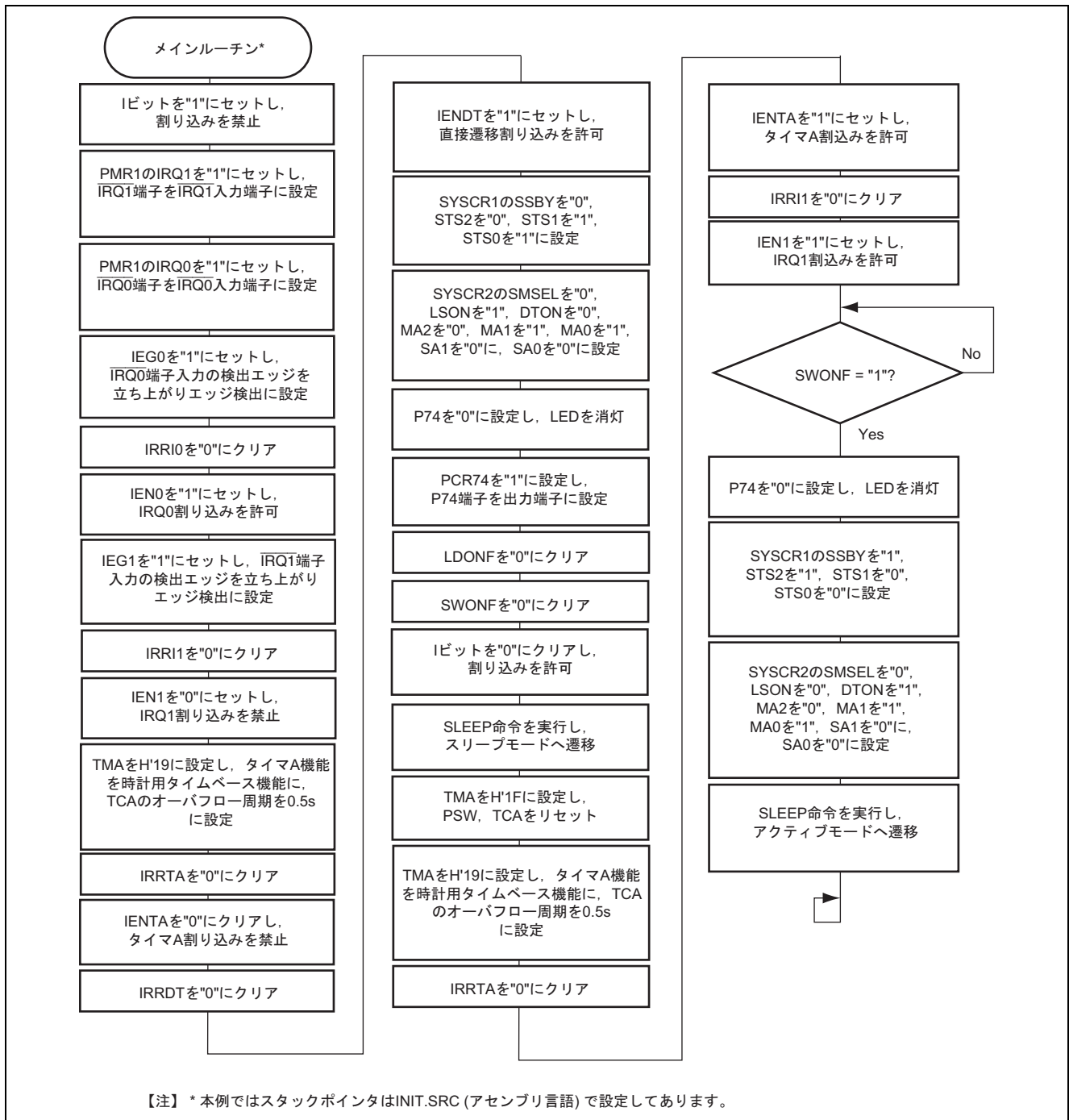
表 4 に本タスク例で使用する RAM の説明を示します。

表 4 使用 RAM 説明

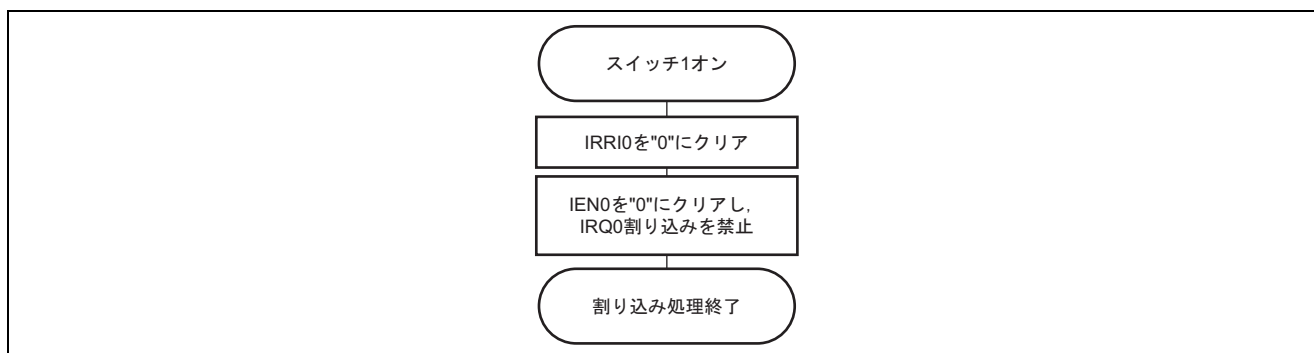
ラベル名		機能	アドレス	使用モジュール名
USRF	SWONF	スイッチ 2 入力の ON/OFF を判定するフラグ	H'FB80 ビット 1	メインルーチン スイッチ 2 オン
	LDONF	LED の ON/OFF を判定するフラグ	H'FB80 ビット 0	LED 制御

5. フローチャート

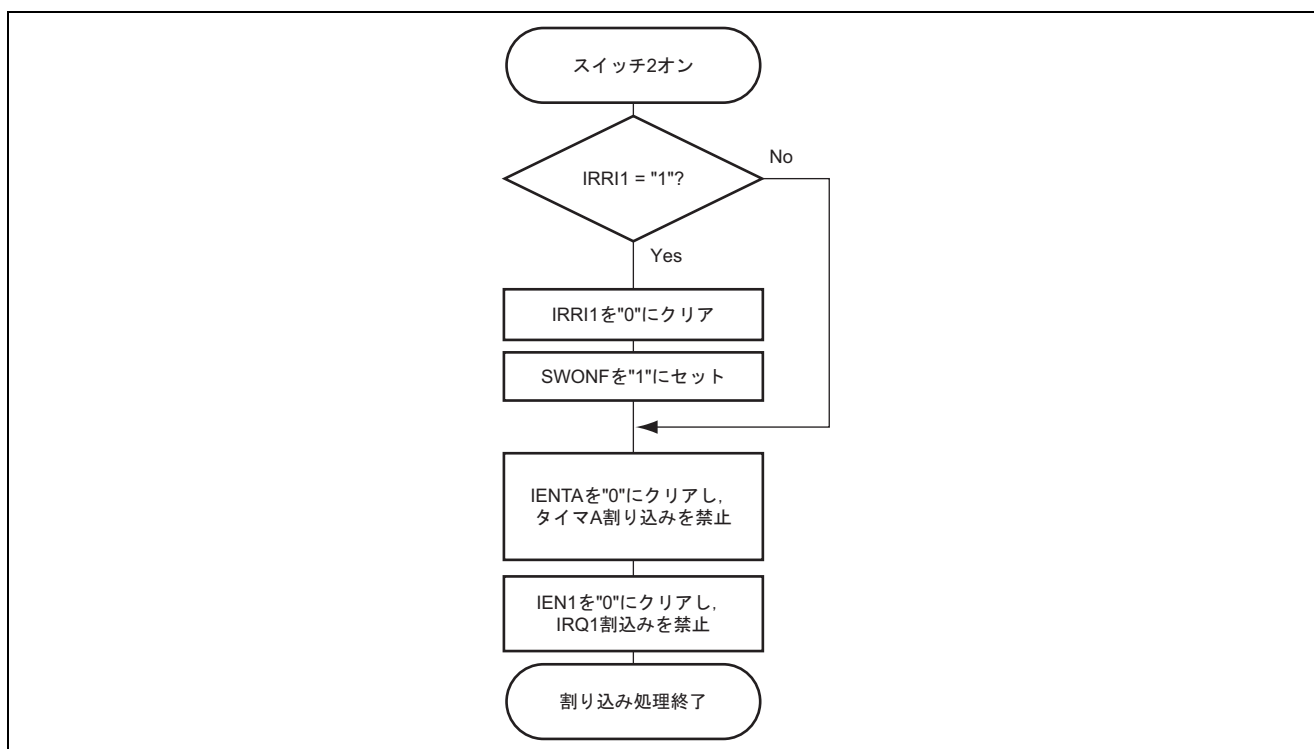
5.1 メインルーチン



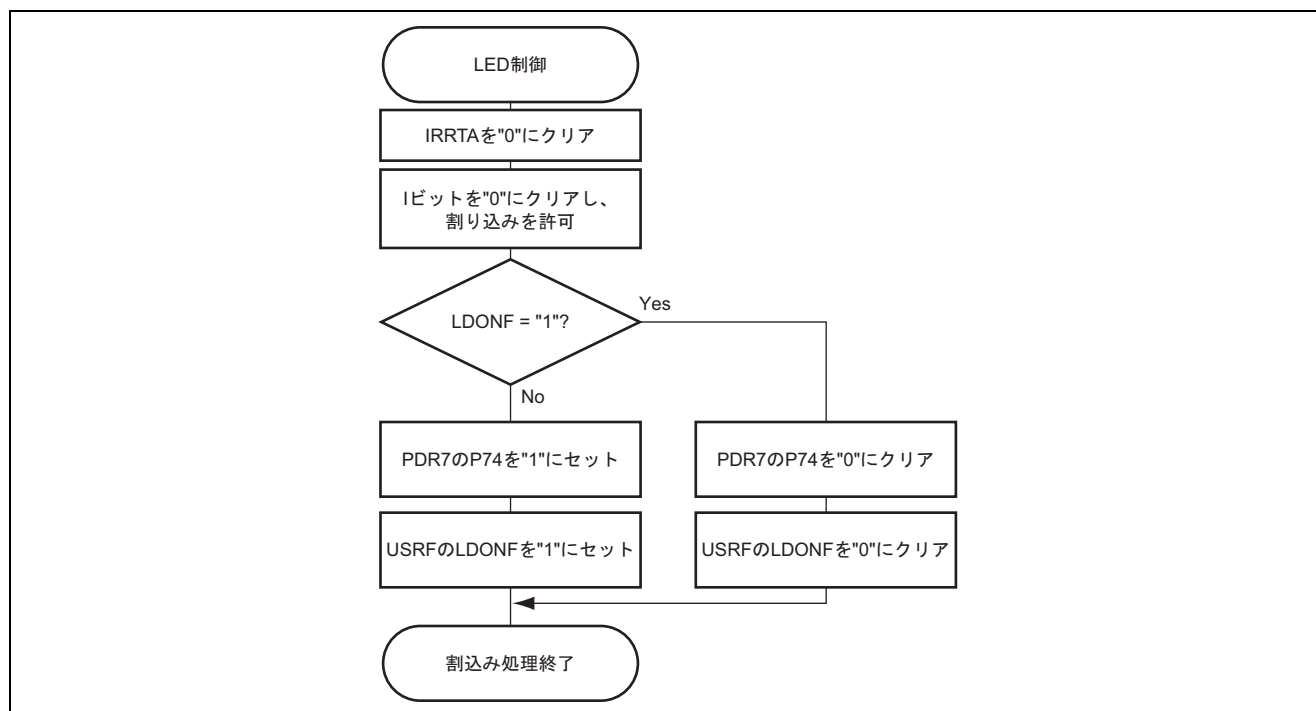
5.2 IRQ0 割り込み処理ルーチン



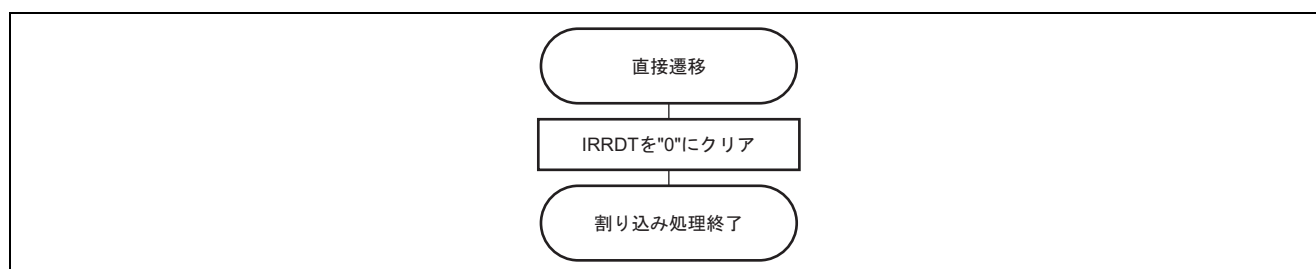
5.3 IRQ1 割り込み処理ルーチン



5.4 タイマ A 割り込み処理ルーチン



5.5 直接遷移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
/*
/* 'Transition to Subactive Mode'
/*
/*
/* Function
/* : Power-Down Mode
/* Subactive Mode
/*
/* 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 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 PMR1_BIT     (*(struct BIT *)0xFFE0) /* Port Mode Register 1
#define IRQ1_SET      PMR1_BIT.b5 /* Port Mode Register 1 bit5
#define IRQ0_SET      PMR1_BIT.b4 /* Port Mode Register 1 bit4
#define PCR7_BIT     (*(struct BIT *)0xFFEA) /* Port Control Register 7
#define PCR74         PCR7_BIT.b4 /* Port Control Register 7 bit4
#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 NESEL       SYSCR1_BIT.b3          /* Noise Elimination Sampling Frequency Select */
#define SYSCR2      *(volatile unsigned char *)0xFFF1 /* System Control Register 2      */
#define SYSCR2_BIT  (*(struct BIT *)0xFFF1) /* System Control Register 2      */
#define LSON        SYSCR2_BIT.b6          /* Low Speed On Flag              */
#define DTON        SYSCR2_BIT.b5          /* Direct Transfer On Flag        */
#define MA1         SYSCR2_BIT.b3          /* Active Mode Clock Select 1     */
#define MA0         SYSCR2_BIT.b2          /* Active Mode Clock Select 0     */
#define SA1         SYSCR2_BIT.b1          /* Subactive Mode Clock Select 1  */
#define SA0         SYSCR2_BIT.b0          /* Subactive Mode Clock Select 0  */
#define IEGR1_BIT   (*(struct BIT *)0xFFF2) /* Interrupt Edge Select Register 1 */

#define IEG1        IEGR1_BIT.b1          /* IRQ1 Edge Select               */
#define IEG0        IEGR1_BIT.b0          /* IRQ0 Edge Select               */
#define IENR1_BIT   (*(struct BIT *)0xFFF4) /* Interrupt Enable Register 1    */
#define IENDT       IENR1_BIT.b7          /* Direct Transfer Interrupt Enable */
#define IENTA       IENR1_BIT.b6          /* Timer A Interrupt Enable       */
#define IEN1        IENR1_BIT.b1          /* IRQ1 Interrupt Request Enable  */
#define IEN0        IENR1_BIT.b0          /* IRQ0 Interrupt Request Enable  */
#define IRR1_BIT    (*(struct BIT *)0xFFF6) /* Interrupt Request Register 1   */
#define IRRDT       IRR1_BIT.b7          /* Direct Transfer Interrupt Request Flag */
#define IRRTA       IRR1_BIT.b6          /* Timer A Interrupt Request Flag  */
#define IRR11       IRR1_BIT.b1          /* IRQ1 Interrupt Request Flag    */
#define IRR10       IRR1_BIT.b0          /* IRQ0 Interrupt Request Flag    */

#pragma interrupt (dtint)
#pragma interrupt (IRQ0)
#pragma interrupt (IRQ1)
#pragma interrupt (taint)
/*****
/* 関数定義
*****/

extern void INIT ( void );          /* SP Set */
void main ( void );
void dtint ( void );
void IRQ0 ( void );
void IRQ1 ( void );
void taint ( void );
void wait ( void );
void sleep ( void );

/*****
/* RAM define
*****/
unsigned char USRF;                /* User Flag Erea */

#define USRF_BIT  (*(struct BIT *)&USRF)
#define SWONF     USRF_BIT.b1      /* Switch On Flag */
#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 SECT0IN SET */
void (*const VEC_TBL2[])(void) = {
    dtint                                /* Sleep Interrupt */
};
#pragma section V3                      /* VECTOR SECT0IN SET */
void (*const VEC_TBL3[])(void) = {
    IRQ0                                 /* IRQ0 Interrupt */
};
#pragma section V4                      /* VECTOR SECT0IN SET */
void (*const VEC_TBL4[])(void) = {
    IRQ1                                 /* IRQ1 Interrupt */
};
#pragma section V5                      /* VECTOR SECT0IN SET */
void (*const VEC_TBL5[])(void) = {
    taint                                /* timer A Interrupt */
};
#pragma section                        /* P */
/*****
/* Main Program
*****/
void main ( void )
{

    set_imask_ccr(1);                    /* Interrupt Disable */

    IRQ1_SET = 1;                        /* Initialize IRQ1 Terminal Input */
    IRQ0_SET = 1;                        /* Initialize IRQ0 Terminal Input */

    IEG0 = 1;                            /* Set Rising Edge of IRQ0 Terminal Input */
    IRRIO = 0;                           /* Clear IRRIO */
    IEN0 = 1;                            /* IRQ0 Interrupt Enable */

    IEG1 = 1;                            /* Set Rising Edge of IRQ1 Terminal Input */
    IRRI1 = 0;                           /* Clear IRRI1 */
    IEN1 = 0;                            /* IRQ1 Interrupt Disable */

    TMA = 0x19;                          /* Set TMA3 */

    IRRTA = 0;                           /* Clear IRRTA */
    IENTA = 0;                           /* Timer A Interrupt Disable */

    IRRDT = 0;                           /* Clear IRRDT */
    IENDT = 1;                           /* Direct Transfer Interrupt Enable */

    SYSCR1 = 0x30;                       /* Initialize Function of Sleep Mode 1 */
    SYSCR2 = 0x4C;                       /* Initialize Function of Sleep Mode 2 */

    P74 = 0;                             /* Initialize P74 */
    CR74 = 1;                             /* Initialize P74 Output Port */

    LDONF = 0;                           /* Initialize LDONF */
    SWONF = 0;                           /* Initialize SWONF */

    set_imask_ccr(0);                    /* Interrupt Enable */

```

```

sleep();                                /* Transition to Sleep Mode */

TMA = 0x1F;                             /* Reset PSW & TCA */
TMA = 0x19;                             /* Initialize Timer A Function */

IRRTA = 0;                              /* Clear IRRTA */
IENTA = 1;                              /* Timer A Interrupt Enable */

IRRI1 = 0;                              /* Clear IRRI1 */
IEN1 = 1;                               /* IRQ1 Interrupt Enable */

while(SWONF != 1){                      /* SWONF = 1 ? */
    ;
}

P74 = 0;                                /* Turn off LED */

SYSCR1 = 0xC0;                          /* Initialize Fubction of Active Mode 1 */
SYSCR2 = 0x2C;                          /* Initialize Function of Active Mode 2 */

sleep();                                /* Transition to Active Mode */

while(1) {
    ;
}

/*****
/* Interrupt
*****/
void IRQ0 ( void )
{
    IRRI0 = 0;                          /* Clear IRRI0 */
    IEN0 = 0;                           /* IRQ0 Interrupt Disable */
}

/*****
/* IRQ1 Interrupt
*****/
void IRQ1 ( void )
{
    if(IRRI1 == 1){                      /* IRRI1 = 1 ? */
        IRRI1 = 0;                      /* Clear IRRI1 */
        SWONF = 1;                      /* Set SWONF */
        IENTA = 0;                      /* Timer A Interrupt Disable */
        IEN1 = 0;                       /* IEN1 Interrupt Disable */
    }
}

/*****
/* Timer A Interrupt
*****/
void taint ( void )

```

```

{

    IRRTA = 0;                                /* Clear IRRTA                */

    set_imask_ccr(0);                          /* Interrupt Enable          */

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

/*****
/* Direct Transfer Interrupt
*****/
void dtint ( void )
{

    IRRDT = 0;                                /* Clear IRRDT              */

}

```

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'001A
CV3	H'001C
CV4	H'001E
CV5	H'0026
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注意ください。

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。