

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 SLP 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。アクティブ(高速)モードで、SLEEP 命令を実行することによりウォッチモードに遷移します。ウォッチモードで、IRQ0 端子に割込みを発生させ、サブアクティブモードに遷移させます。SLEEP 命令を実行しアクティブ(高速)モードに直接遷移します。

### 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 .....	4
4. ソフトウェア説明 .....	5
5. フローチャート .....	9
6. プログラムリスト .....	11

### 1. 仕様

- (1) サブアクティブモードへの遷移を行いません。
- (2) アクティブ(高速)モードで、SLEEP 命令を実行することによりウォッチモードに遷移します。
- (3) ウォッチモードで、IRQ0 端子に接続したスイッチ 0(SW0)がオンになると IRQ0 割込みが発生し、ウォッチモードは解除され、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、0.5sec ごとにタイマ A 割込み要求が発生し、タイマ A 割込み処理の中で LED の制御を行いません。LED は 0.5sec 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、IRQ1 端子に接続したスイッチ 1(SW1)がオンになると IRQ1 割込みが発生し、IRQ1 割込み終了後に、SLEEP 命令を実行しアクティブ(高速)モードに直接遷移します。
- (6) LED はポート 9 の P92 出力端子に接続されているものとします。
- (7) P92 は、大電流ポートです。
- (8) 図 1 に IRQ1, IRQ0 端子に接続するスイッチ 1, スwitch 0 の接続例を示します。

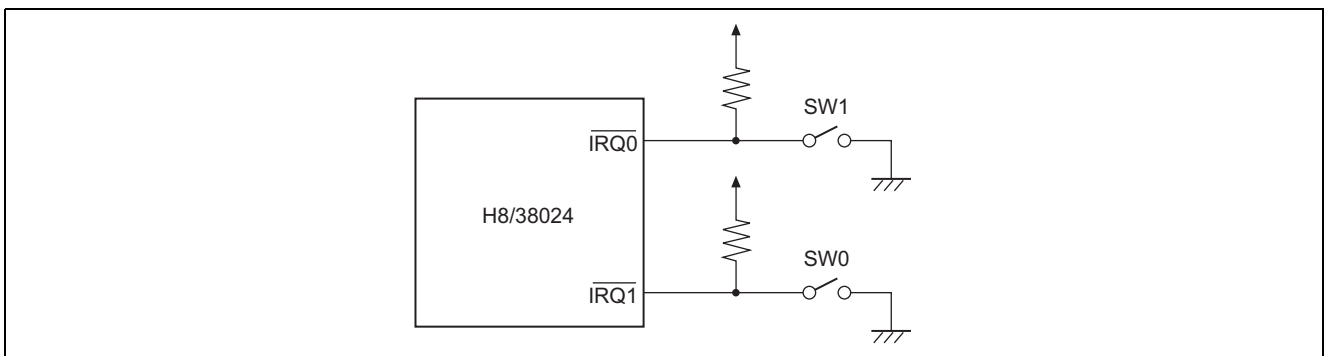


図 1 サブアクティブモードへの遷移におけるスイッチ接続例

### 2. 使用機能説明

- (1) 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いません。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。
  - ウォッチモードで割込み(タイマ A, タイマ F, タイマ G, IRQ0, WKP7 ~ WKP0)が発生したとき、SYSCR1 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込み(タイマ A, タイマ C, タイマ F, タイマ G, SCI3, IRQ4, IRQ3, IRQ1, IRQ0, IRQAEC, WKP7 ~ WKP0, 非同期イベントカウンタ)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、または割込み許可レジスタにより当該割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。
  - サブアクティブモードの解除は、SLEEP 命令または RES 端子入力により行われます。
  - SLEEP 命令による解除は、SYSCR1 の SSBY が"1", TMA の TMA3 が"1"の状態です。SLEEP 命令を実行すると、サブアクティブモードは解除され、ウォッチモードに遷移します。
  - 直接遷移によってアクティブ(高速)モードへ遷移します。
  - RES 端子による解除は、RES 端子を"Low"レベルにすると、システムクロックの発振が開始されます。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時間は、SYSCR1 の STS2 ~ STS0 により設定します。
  - 本タスク例では、発振安定時間を 1.638ms に設定しています。
  - 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(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 3 つのモードがあります。この 3 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。遷移後は直接遷移割込み例外処理を開始します。なお、割込み許可レジスタ 2 により直接遷移割込みが禁止されている場合は、スリープモードまたはウォッ

チ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ウォッチモードに遷移し、遷移後のモードから割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(高速)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 "1", LSON を "0", SYSCR2 の MSON を "0", DTON を "1", TMA の TMA3 を "1" にセットした状態で SLEEP 命令を実行すると、ウォッチモードを経由し、SYSCR1 の STS2 ~ STS0 により設定された時間を経過した後、直接、アクティブ(高速)モードに遷移します。

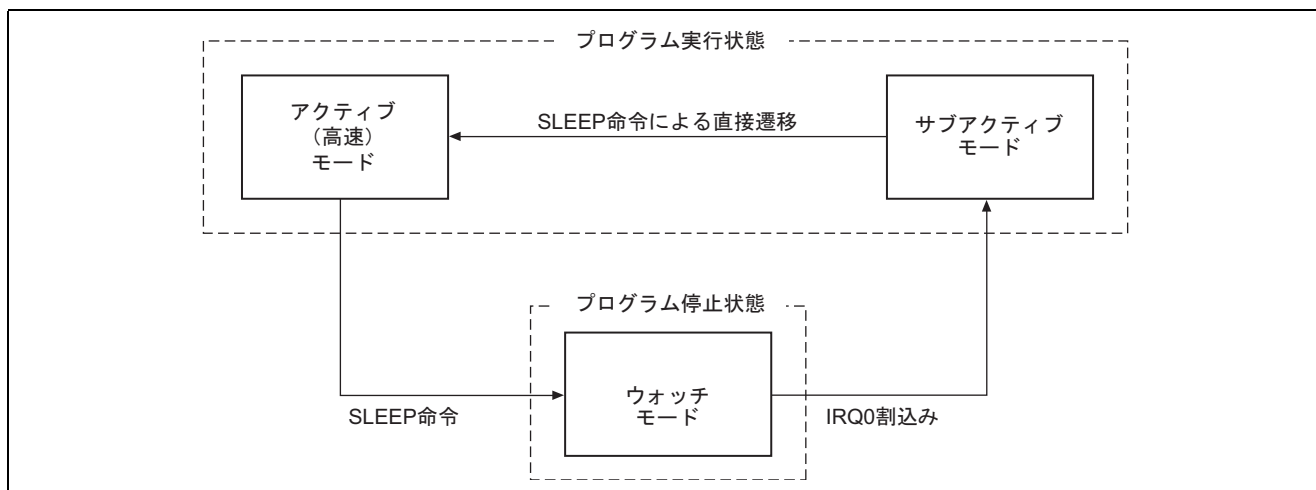


図2 サブアクティブモードへのモード遷移図

(2) 表 1 に本タスク例の機能割付けを示します。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 機能割付け

機能	機能割付け
PSW	サブクロック(32.768kHz)の4分周を入力とする5ビットのアップカウンタ
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
PDR9	P92 出力端子のデータの格納
P92	LED 出力
IEG1	IRQ1 端子の入力エッジの選択
IEG0	IRQ0 端子の入力エッジの選択
IENTA	タイマ A 割込み要求の許可、または禁止を設定
IENI1	IRQ1 割込み要求の許可、または禁止を設定
IENI0	IRQ0 割込み要求の許可、または禁止を設定
IENDT	直接遷移割込み要求の許可を行なう
IRRTA	タイマ A 割込み要求の有無を反映
IRRI1	IRQ1 割込み要求の有無を反映
IRRI0	IRQ0 割込み要求の有無を反映
IRRDT	直接遷移割込みの有無を反映
IRQ1	スイッチ 1 入力
IRQ0	スイッチ 0 入力

3. 動作説明

(1) 図3に動作説明を示します。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。

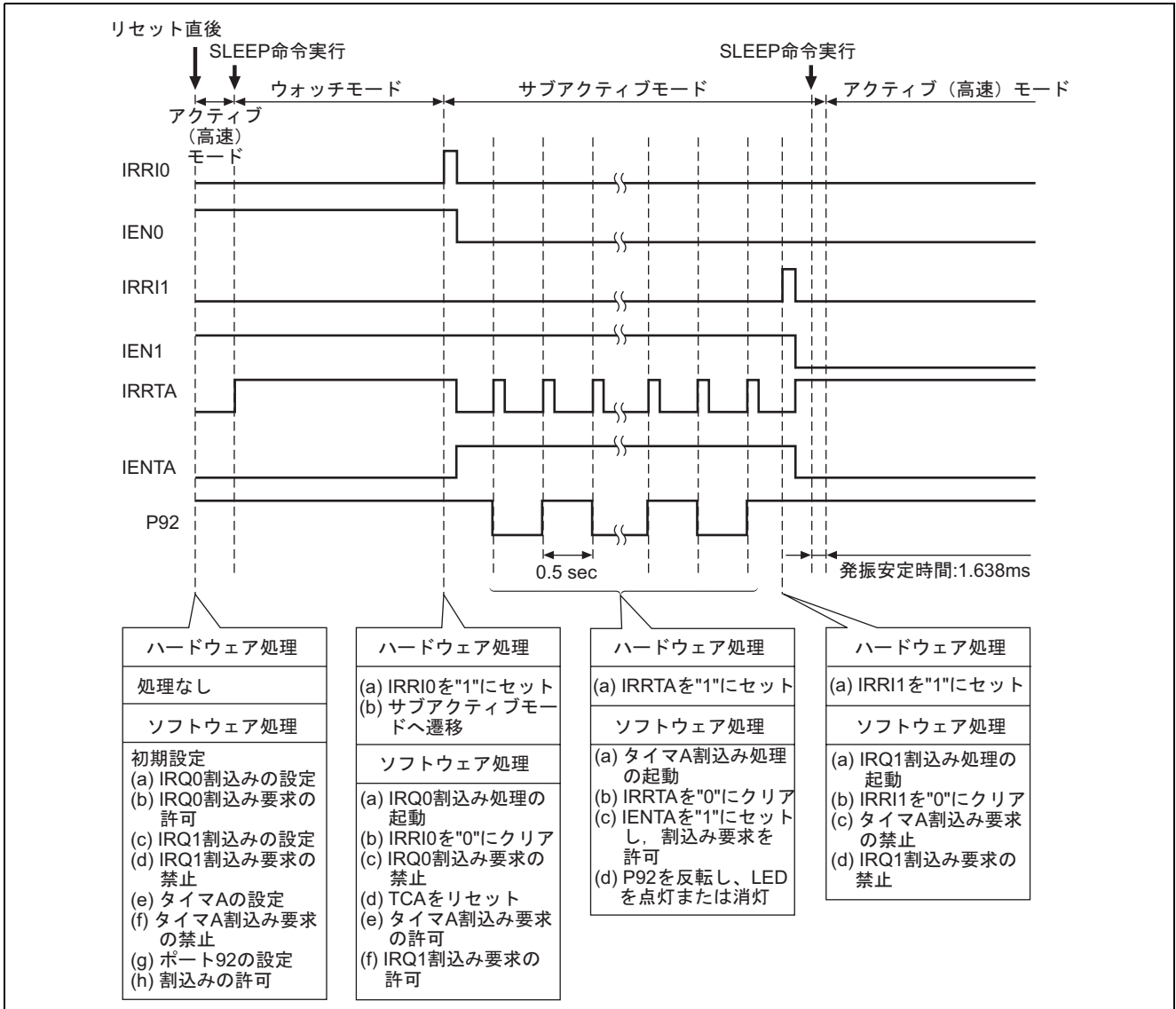


図3 サブアクティブモードへの遷移の動作説明

#### 4. ソフトウェア説明

##### (1) モジュール説明

本タスク例のモジュールを表 2 に示します。

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	IRQ1 割込みの設定, IRQ0 割込みの設定, タイマ A 割込みの設定, ポート 9 の設定, 割込みの許可, ウォッチモード, アクティブモードへの遷移を行なう
スイッチ 1 オン	irq1int	IRQ1 割込み処理ルーチンで, SWONF のセット, タイマ A 割込み要求の禁止, IRQ1 割込み要求の禁止を行なう
スイッチ 0 オン	irq0int	IRQ0 割込み処理ルーチンで, IRQ0 割込み要求の禁止を行なう
LED 制御	taint	タイマ A 割込み処理ルーチンで, 割込み要求の許可, LED の制御を行なう
直接遷移	dtint	直接遷移割込み処理ルーチンで, 直接遷移割込み要求フラグのクリアを行なう

##### (2) 引数の説明

本タスク例では, 引数を使用しません。

## (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 3 に示します。

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 0.5sec に設定	H'FFB0	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 0.5sec でオーバフローする 8 ビットのアップカウンタ	H'FFB1	H'00
PDR9	P92	ポートデータレジスタ 9(ポートデータレジスタ 92) : P92=0 のとき, P92 端子の出力レベルは"Low" : P92=1 のとき, P92 端子の出力レベルは"High"	H'FFDC ビット 2	1
PMR2	IRQ0	ポートモードレジスタ 2(IRQ0I/O ポートセレクト) : IRQ0="0" のとき, IRQ0 端子を汎用入出力ポートに設定 : IRQ0="1" のとき, IRQ0 端子を IRQ0 入力端子に設定	H'FFC9 ビット 0	1
PMRB	IRQ1	ポートモードレジスタ B(IRQ1I/O ポートセレクト) : IRQ1="0" のとき, IRQ1 端子を汎用入出力ポートに設定 : IRQ1="1" のとき, IRQ1 端子を IRQ1 入力端子に設定	H'FFEE ビット 3	1
IEGR	IEG1	IRQ エッジセレクトレジスタ(IRQ1 エッジセレクト) : IEG1="0" のとき, IRQ1 端子入力の立ち下がりエッジを検出 : IEG1="1" のとき, IRQ1 端子入力の立ち上がりエッジを検出	H'FFF2 ビット 1	0
	IEG0	IRQ エッジセレクトレジスタ(IRQ0 エッジセレクト) : IEG0="0" のとき, IRQ0 端子入力の立ち下がりエッジを検出 : IEG0="1" のとき, IRQ0 端子入力の立ち上がりエッジを検出	H'FFF2 ビット 0	0
IENR1	IEN1A	割込み許可レジスタ 1(タイマ A 割込みイネーブル) : IEN1A="0" のとき, タイマ A 割込み要求を禁止 : IEN1A="1" のとき, タイマ A 割込み要求を許可	H'FFF3 ビット 7	1
	IEN1	割込み許可レジスタ 1(IRQ1 割込みイネーブル) : IEN1="0" のとき, IRQ1 端子による割込み要求を禁止 : IEN1="1" のとき, IRQ1 端子による割込み要求を許可	H'FFF3 ビット 1	1
	IEN0	割込み許可レジスタ 1(IRQ0 割込みイネーブル) : IEN0="0" のとき, IRQ0 端子による割込み要求を禁止 : IEN0="1" のとき, IRQ0 端子による割込み要求を許可	H'FFF3 ビット 0	1
IENR2	IENDT	割込み許可レジスタ 2(直接遷移割込みイネーブル) : IENDT="0" のとき, 直接遷移による割込み要求を禁止 : IENDT="1" のとき, 直接遷移による割込み要求を許可	H'FFF4 ビット 7	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
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="1"のとき, アクティブモードで SLEEP 命令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 命令実行後サブスリープモードに遷移	H'FFF0 ビット 7	1
	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="0", STS1="0", STS0="0"のとき, スタンバイモード, ウォッチモード解除後の発振安定時間を 1.638ms に設定	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) : LSON="0"のとき, ウォッチモードを解除時, CPU の動作クロックをシステムクロックに設定 : LSON="1"のとき, ウォッチモードを解除時, CPU の動作クロックをサブクロックに設定	H'FFF0 ビット 3	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, 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。サブアクティブモードで SLEEP 命令を実行したとき, ウォッチモード, またはサブスリープモードに遷移 : DTON="1"のとき, サブアクティブモードで SLEEP 命令を実行したとき, アクティブ(高速)モード(SSBY="1", TMA3="1", LSON="0", MSON="0"のとき), またはアクティブ(中速)モード(SSBY="1", TMA="1", LSON="0", MSON="1"のとき)に直接遷移	H'FFF1 ビット 3	1
	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) : MSON="0"のとき, 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。アクティブモードで SLEEP 命令を実行したとき, スリープ(高速)モードで動作 : MSON="1"のとき, 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(中速)モードで動作。アクティブモードで SLEEP 命令を実行したとき, スリープ(中速)モードで動作	H'FFF1 ビット 2	0
	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 $\omega/8$ に設定	H'FFF1 ビット 1 ビット 0	SA1="0" SA0="0"

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
IRR1	IRRTA	割込み要求レジスタ 1(タイマ A 割込み要求フラグ) : IRRTA="0"のとき, タイマ A 割込みが要求されていない : IRRTA="1"のとき, タイマ A 割込みが要求されている	H'FFF6 ビット 7	0
	IRRI1	割込み要求レジスタ 1(IRQ1 割込み要求フラグ) : IRR11="0"のとき, IRQ1 端子入力による割込みが要求されていない : IRR11="1"のとき, IRQ1 端子入力による割込みが要求されている	H'FFF6 ビット 1	0
	IRRI0	割込み要求レジスタ 1(IRQ0 割込み要求フラグ) : IRR10="0"のとき, IRQ0 端子入力による割込みが要求されていない : IRR10="1"のとき, IRQ0 端子入力による割込みが要求されている	H'FFF6 ビット 0	0
IRR2	IRRDT	割込み要求レジスタ 2(直接遷移割込み要求フラグ) : IRRDT="0"のとき, 直接遷移による割込みが要求されていない : IRRDT="1"のとき, 直接遷移による割込みが要求されている	H'FFF7 ビット 7	0

## (4) 使用 RAM 説明

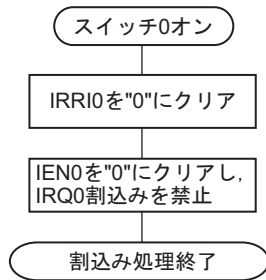
本タスク例の使用 RAM を表 4 に示します。

表 4 使用 RAM 説明

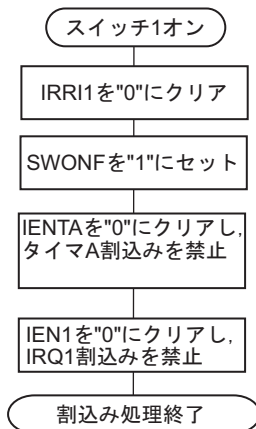
ラベル名		機能	アドレス	使用モジュール名
USRF	SWONF	スイッチ 1 入力の ON/OFF を判定するフラグ	H'FB80 ビット 1	メインルーチン スイッチ 1 オン
	LDONF	LED の ON/OFF を判定するフラグ	H'FB80 ビット 0	LED 制御



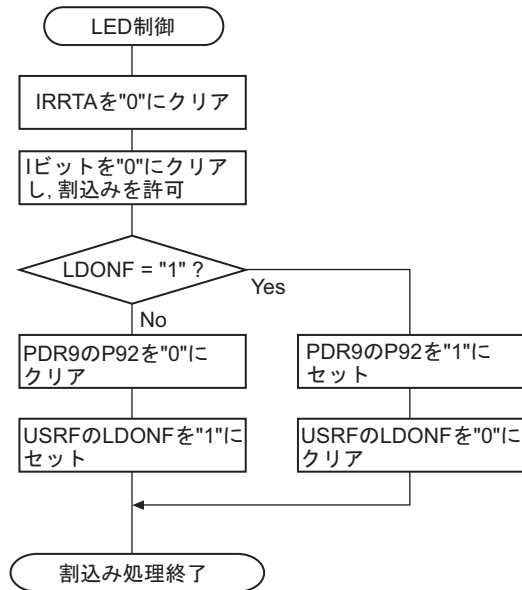
(2) IRQ0 割込み処理ルーチン



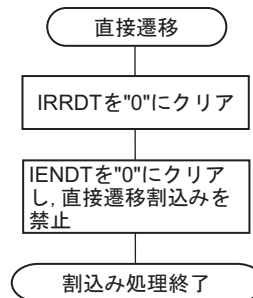
(3) IRQ1 割込み処理ルーチン



(4) タイマA 割込み処理ルーチン



(5) 直接遷移 割込み処理ルーチン



## 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   #'FF80,R7
        LDC.B   #'10000000,CCR
        JMP     @_main
;
        .END

```

```

/*****/
/*                                     */
/* H8/300L Super Low Power Series      */
/*   -H8/38024 Series-                 */
/* Application Note                     */
/*                                     */
/* 'Transition to Subactive Mode'      */
/*                                     */
/* Function                             */
/* : Power-Down Mode                   */
/*   Subactive Mode                     */
/*                                     */
/* 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 PMR2_BIT (*(struct BIT *)0xFFC9) /* Port Mode Register 2 */
#define IRQ0     PMR2_BIT.b0 /* Port Mode Register 2 bit0 */
#define PDR9_BIT (*(struct BIT *)0xFFDC) /* Port Data Register 9 */
#define P92      PDR9_BIT.b2 /* Port Data Register 92 */
#define PMRB_BIT (*(struct BIT *)0xFFEE) /* Port Mode Register B */
#define IRQ1     PMRB_BIT.b3 /* Port Mode Register B bit3 */
#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 IEGR_BIT (*(struct BIT *)0xFFF2) /* Interrupt Edge Select Register 1 */
#define IEG1 IEGR_BIT.b1 /* IRQ1 Edge Select */
#define IEG0 IEGR_BIT.b0 /* IRQ0 Edge Select */
#define IENR1_BIT (*(struct BIT *)0xFFF3) /* Interrupt Enable Register 1 */
#define IENTA IENR1_BIT.b7 /* Timer A Interrupt Enable */
#define IEN1 IENR1_BIT.b1 /* IRQ1 Interrupt Request Enable */
#define IEN0 IENR1_BIT.b0 /* IRQ0 Interrupt Request Enable */
#define IENR2_BIT (*(struct BIT *)0xFFF4) /* Interrupt Enable Register 1 */
#define IENDT IENR2_BIT.b7 /* Direct Transfer Interrupt Enable */
#define IRR1_BIT (*(struct BIT *)0xFFF6) /* Interrupt Request Register 1 */
#define IRRTA IRR1_BIT.b7 /* Timer A Interrupt Request Flag */
#define IRR11 IRR1_BIT.b1 /* IRQ1 Interrupt Request Flag */
#define IRR10 IRR1_BIT.b0 /* IRQ0 Interrupt Request Flag */
#define IRR2_BIT (*(struct BIT *)0xFFF7) /* Interrupt Request Register 1 */
#define IRRDT IRR2_BIT.b7 /* Direct Transfer Interrupt Request Flag */

#pragma interrupt (dtint)
#pragma interrupt (irq0int)
#pragma interrupt (irq1int)
#pragma interrupt (taint)
/*****/
/* Function define */
/*****/
extern void INIT ( void ); /* SP Set */
void main ( void );
void dtint ( void );
void irq0int ( void );
void irq1int ( void );
void taint ( void );

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

#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 SECTOIN SET          */
void (*const VEC_TBL2[])(void) = {
    irq0int                              /* 08 IRQ0 Interrupt          */
};
#pragma section      V3                                /* VECTOR SECTOIN SET          */
void (*const VEC_TBL3[])(void) = {
    irq1int                              /* 0A IRQ1 Interrupt          */
};
#pragma section      V4                                /* VECTOR SECTOIN SET          */
void (*const VEC_TBL4[])(void) = {
    taint                                /* 16 timer A Interrupt       */
};
#pragma section      V5                                /* VECTOR SECTOIN SET          */
void (*const VEC_TBL5[])(void) = {
    dtint                                /* 28 Sleep Interrupt         */
};

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

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

    P92 = 1;                                         /* Initialize P92             */

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

    TMA = 0x1F;                                       /* Reset PSW & TCA           */
    TMA = 0x19;                                       /* Set TMA3                   */
    SYSCR1 = 0x8F;                                    /* Set SYSCR 1                */
    SYSCR2 = 0xE0;                                    /* Set SYSCR 2                */

    IEG0 = 0;                                       /* Initialize IRQ0 Terminal Input Edge */
    IRRIO = 0;                                       /* Clear IRRIO                */
    IEN0 = 1;                                       /* IRQ0 Interruput Enable     */

    IEG1 = 0;                                       /* Initialize IRQ1 Terminal Input Edge */
    IRRI1 = 0;                                       /* Clear IRRI1                */
    IEN1 = 0;                                       /* IRQ1 Interruput Disable    */

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

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

    set_imask_ccr(0);                                /* Interrupt Enable           */
}

```

```

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

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

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

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

while(SWONF != 1){                       /* SWONF = "1" ?                   */
    ;
}
set_imask_ccr(1);                         /* Interrupt Disable               */

P92 = 1;                                  /* Turn off LED                     */

SYSCR1 = 0x87;                           /* Initialize Fubction of Active Mode 1 */
SYSCR2 = 0xE8;                           /* Initialize Function of Active Mode 2 */

set_imask_ccr(0);                         /* Interrupt Enable                 */
sleep();                                  /* Transition to Active Mode        */
set_imask_ccr(1);                         /* Interrupt Disable                */

while(1){
    ;
}

/*****/
/* IRQ0 Interrupt                          */
/*****/
void irq0int ( void )
{
    IRRIO = 0;                            /* Clear IRRIO                      */
    IENO = 0;                             /* IRQ0 Interrupt Disable           */
}

/*****/
/* IRQ1 Interrupt                          */
/*****/
void irq1int ( void )
{
    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 )
{
    IRRТА = 0;                            /* Clear IRRТА                      */

```



```

set_imask_ccr(0);                                /* Interrupt Enable */

if(LDONF == 1){                                  /* LDONF = "1" ? */
    P92 = 1;                                     /* Turn on LED */
    LDONF = 0;                                   /* Clear LDONF */
}
else{
    P92 = 0;                                     /* Turn off LED */
    LDONF = 1;                                   /* Set LDONF */
}
}

/*****/
/* Direct Transfer Interrupt */
/*****/
void dtint ( void )
{
    IRRDT = 0;                                   /* Clear IRRDT */
    IENDT = 0;                                   /* Direct Transfer Interrupt Enable */
}

```

リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'0008
CV3	H'000A
CV4	H'0016
CV5	H'0028
P	H'0100
B	H'FB80

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

### 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。