

2.22 スリープモードへの遷移

スリープモードへの遷移	使用機能	低消費電力モード：スリープモード
-------------	------	------------------

仕様

- (1) スリープモードへの遷移を行ないます。
- (2) アクティブモードでプログラム実行中に、IRQ₀端子に接続したスイッチ入力によりIRQ₀割込みを発生させ、IRQ₀割込み処理終了後にSLEEP命令の実行によりスリープモードへ遷移させます。
- (3) スリープモードに遷移してから、1sec後にタイマA割込みによりスリープモードを解除し、アクティブモードへ遷移させます。
- (4) プログラム起動と同時にLEDを点灯し、スリープモードへ遷移するとLEDを消灯します。そして、アクティブモードへ遷移すると再びLEDを点灯します。
- (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。
- (6) 図1にIRQ₀端子に接続するスイッチの接続例を示します。

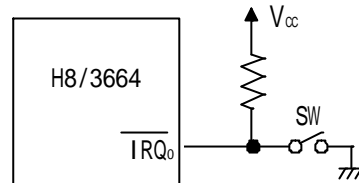


図1 スリープモードへの遷移におけるスイッチ接続例

使用機能説明

- (1) 本タスク例では、低消費電力モードのスリープモードへの遷移を行ないます。図2にスリープモードへのモード遷移図を示します。以下にスリープモードの機能の説明を示します。
 - ・アクティブモードで、SYSCR1のSSBYが"0"、SYSCR2のSMSELが"0"、LSONが"0"、DTONが"0"のときSLEEP命令を実行すると、スリープモードに遷移します。
 - ・スリープモードではCPUの動作は停止しますが、内蔵周辺モジュールはSYSCR2のMA2、MA1、MA0で設定した周波数のクロックで動作します。なお、CPUのレジスタの内容は保持されます。
 - ・スリープモードの解除は、すべての割込み（タイマA、タイマV、タイマW、IRQ₃～IRQ₀、WKP₅～WKP₀、ウォッチドックタイマ、SCI3、IIC、A/D変換器）、RES端子入力によって行われます。
 - ・割込みによる解除において、割込み要求が発生すると、スリープモードは解除され、割込み例外処理を開始します。スリープモードからはアクティブモードに遷移します。
 - ・CCRのIビットが"1"のとき、あるい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スリープモードは解除されません。
 - ・RES端子による解除において、RES端子を"Low"レベルにするとリセット状態に遷移し、スリープモードは解除されます。
 - ・本タスク例では、スリープモードの解除にタイマA割込みを使用します。



図2 スリープモードへの遷移におけるモード遷移図

使用機能説明

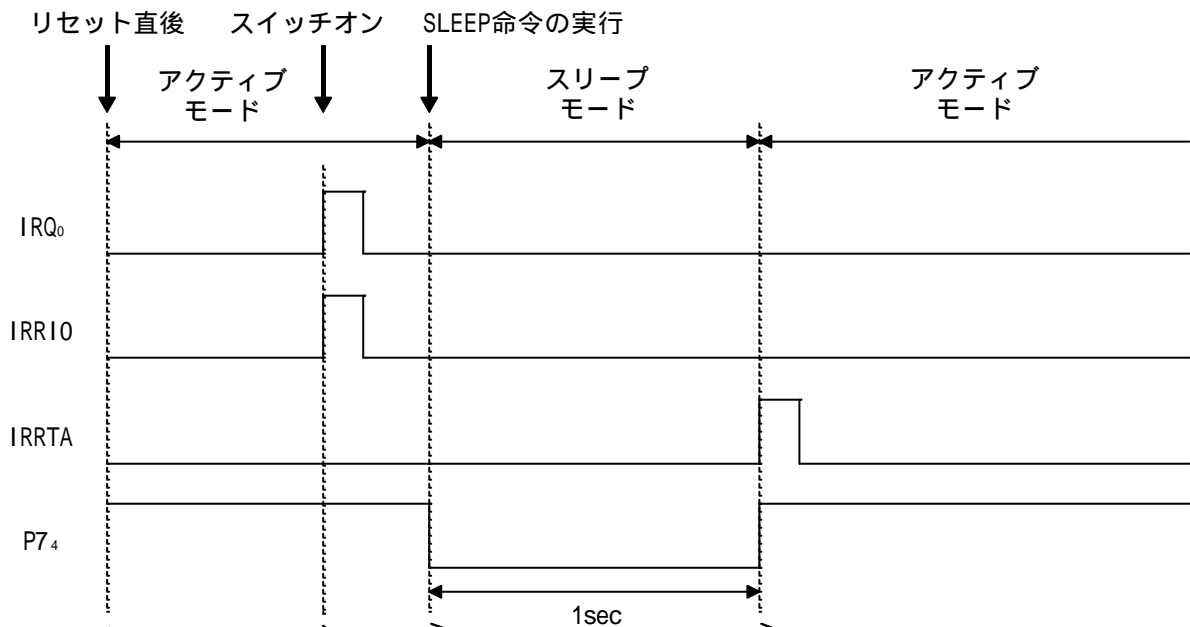
(2) 表1に本タスク例の機能割付けを示します。表1に示すように機能を割付け、スリープモードへの遷移を行ないます。

表1 機能割付け

機能	機能割付け
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	P7 ₄ 出力端子機能の設定
PDR7	P7 ₄ 出力端子のデータの格納
P7 ₄	LED出力
IRQ ₀	スイッチ入力

動作原理

(1) 図3に動作原理を示します。図2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スリープモードへの遷移を行ないます。



ハードウェア処理
処理なし
ソフトウェア処理
初期設定
(a)ポート7の設定
(b)P7 ₄ 出力端子を "High" に設定
(c)IRQ ₀ 割込みの設定
(d)割込みの許可

ハードウェア処理
(a) IRR10を "1" にセット
ソフトウェア処理
(a) IRR10を "0" にクリア

ハードウェア処理
処理なし
ソフトウェア処理
(a) P7 ₄ 出力端子を "Low" に設定
(b) SLEEP命令の実行

ハードウェア処理
(a) IRRTAを "1" にセット
ソフトウェア処理
(a) IRRTAを "0" にクリア
(b) P7 ₄ 出力端子を "High" に設定

図3 スリープモードへの遷移の動作原理

ソフトウェア説明

(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	IRQ ₀ 割込みの設定、ポート7の設定、割込みの許可、LEDの制御、タイマA割込みの設定、スリープモードへの遷移を行なう
スイッチオン	IRQ0	INT ₀ 割込み処理ルーチンで、SWONFのセット、INT ₀ 割り込みの禁止を行なう
スリープモード解除	taint	タイマA割込み処理ルーチンで、タイマA割り込みの禁止を行なう

(2) 引数の説明

本タスク例では、引数は使用していません。

(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'18のとき、タイマA機能を時計用タイムベース機能に、TCA入力クロックソースをPSWに、TCAオーバフロー周期を1secに設定	H'FFA6	H'18	
TCA	タイマカウンタA ：32.768kHzを32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	P7 ₄	ポートデータレジスタ7 (ポートデータレジスタ7 ₄) ：P7 ₄ =0のとき、P7 ₄ 端子の出力レベルは"Low" ：P7 ₄ =1のとき、P7 ₄ 端子の出力レベルは"High"	H'FFDA ビット4	0
PCR7	PCR7 ₄	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7 ₄) ：PCR7 ₄ =1のとき、P7 ₄ 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	H'FFF0 ビット7	0
SYSCR2	SMSEL	システムコントロールレジスタ2 (スリープモード選択) ：SMSEL="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="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、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	0
	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="1"、MA1="1"、MA0="0"、のとき、32.768kHzを32分周したクロックを設定	H'FFF1 ビット4 ビット3 ビット2	MA2="1" MA1="1" MA0="0"

ソフトウェア説明

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
IEGR1	IEGO	割込みエッジセレクトレジスタ1 (IRQ ₀ エッジセレクト) : IEGO="1"のとき、IRQ ₀ 端子入力の立上りエッジを検出	H'FFF2 ビット0	1
IENR1	IENTA	割込みイネーブルレジスタ1 (タイマA割込みイネーブル) : IENTA="0"のとき、タイマA割込み要求を禁止 : IENTA="1"のとき、タイマA割込み要求を許可	H'FFF4 ビット6	1
	IENO	割込みイネーブルレジスタ1 (IRQ ₀ 割込みイネーブル) : IENO="0"のとき、IRQ ₀ 割込み要求を禁止 : IENO="1"のとき、IRQ ₀ 割込み要求を許可	H'FFF4 ビット0	1
IRR1	IRRTA	割込みフラグレジスタ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6	0
	IRRI0	割込みフラグレジスタ1 (IRQ ₀ 割込み要求フラグ) : IRRI0="0"のとき、IRQ ₀ 割込みが要求されていない : IRRI0="1"のとき、IRQ ₀ 割込みが要求されている	H'FFF6 ビット0	0

(4) 使用RAM説明

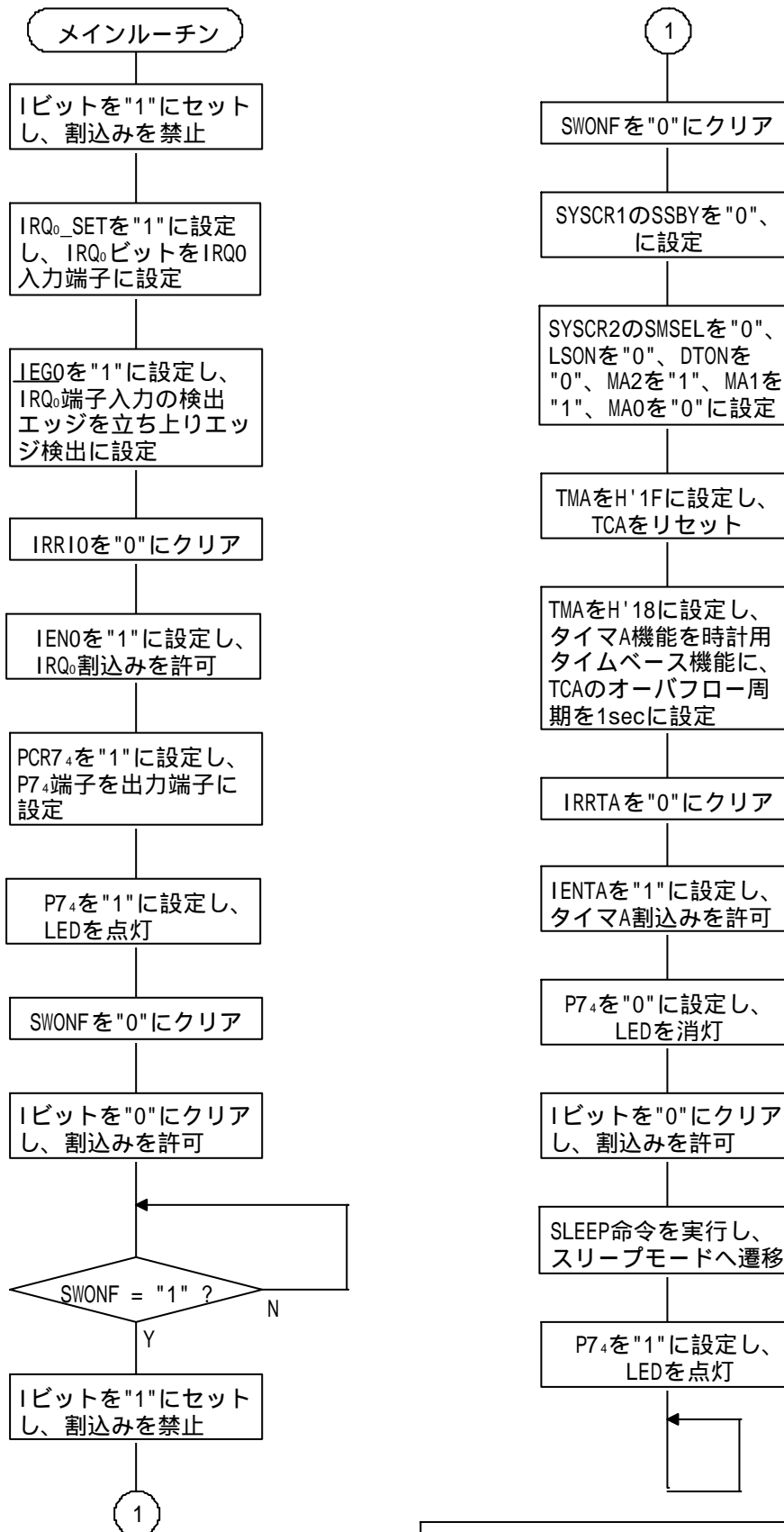
表4に本タスク例における使用RAM説明を示します。

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
USRFB	SWONF	スイッチ入力のON/OFFを判定するフラグ	H'FB80 ビット0	メインルーチン スイッチオン

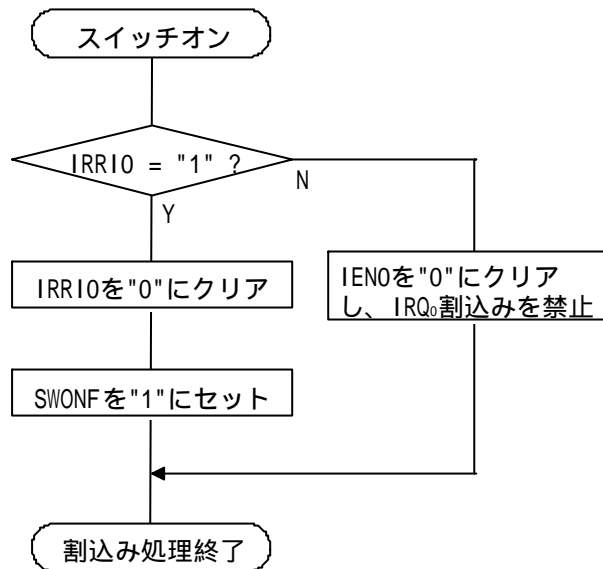
フローチャート

(a) メインルーチン



本例ではスタックポインタはINIT.SRC（アセンブリ言語）で設定してあります。

フローチャート

(b) IRQ₀割込み処理ルーチン

(c) タイマA割込み処理ルーチン



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 Sleep (High-Speed) Mode' */
/*                                     */
/* Function                             */
/* : Power-Down Mode                   */
/*   Sleep (High-Speed) 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 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 IEG0        IEGR1_BIT.b0 /* IRQ0 Edge Select */
#define IENR1_BIT   (*(struct BIT *)0xFFF4) /* Interrupt Enable Register 1 */
#define IENTA       IENR1_BIT.b6 /* Timer A Interrupt Enable */
#define IENO        IENR1_BIT.b0 /* IRQ0 Interrupt Enable */

```


プログラムリスト

```

SWONF = 0; /* Clear SWONF */

SYSCR1 = 0x00; /* Initialize Fubction of Sleep Mode 1 */
SYSCR2 = 0x0C; /* Initialize Fubction of Sleep Mode 2 */

TMA = 0x1F; /* Initialize TCA */
TMA = 0x18; /* Initialize TCA Overflow Period */
IRRTA = 0; /* Clear IRRTA */
IENTA = 1; /* Timer A Interrupt Enable */

P74 = 0; /* Turn Off LED */

set_imask_ccr(0); /* Interrupt Enable */

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

P74 = 1; /* Turn On LED */

while(1) {
;
}

/*****/
/* IRQO Interrupt */
/*****/
void IRQO ( void )
{
if(IRRIO == 1){ /* IRRIO = "1" ? */
IRRIO = 0; /* Clear IRRIO */
SWONF = 1; /* Set SWONF */
}
else{
IENO = 0; /* IENO Interrupt Disable */
}
}

/*****/
/* Timer A Interrupt */
/*****/
void taint ( void )
{
IRRTA = 0; /* Clear IRRTA */

IENTA = 0; /* TimerA Interrupt Disable */
}

```

リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'001C
CV3	H'0026
P	H'0100
B	H'FB80