

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。

SH7145 グループ

ソフトウェアスタンバイモードへの遷移および復帰

要旨

ソフトウェアスタンバイモードへの遷移および復帰を NMI 割り込み処理を用いて行います。また、遷移および復帰の状態を 7セグメント LED に表示します。

動作確認デバイス

SH7145F

目次

1. 仕様	2
2. 使用機能説明	4
3. 動作	10
4. ソフトウェア説明	11
5. フローチャート	15
6. プログラムリスト	19

1. 仕様

ソフトウェアスタンバイモードへの遷移および復帰を行います。

通常のプログラム実行中に、NMI 端子に接続したスイッチ入力の立ち上がりエッジを検出することにより、NMI 割り込みを発生させ、NMI 割り込み処理終了後にスイッチ入力のエッジ検出を立ち下がりに変更し、SLEEP 命令の実行を行いソフトウェアスタンバイモードへ遷移させます。通常のプログラム実行中では、7セグメント LED の表示は「A」となり、ソフトウェアスタンバイモードへ遷移すると7セグメント LED の表示は「S」となります。

ソフトウェアスタンバイモードで、NMI 端子に接続したスイッチ入力の立ち下がりエッジを検出することにより、NMI 割り込みが発生し、ウォッチドックタイマ (WDT) を起動させます。このタイマのオーバーフローにより、ソフトウェアスタンバイモードから通常のプログラム実行へ遷移させます。NMI 割り込み処理終了後にスイッチ入力のエッジ検出を立ち上がりに変更します。ソフトウェアスタンバイモードから通常のプログラム実行へと遷移すると、7セグメント LED の表示は「S」から「A」となります。

本タスク例では、7セグメント LED を表示させるためにバスステートコントローラ (BSC) を使用し、外部メモリとして7セグメント LED の表示を制御します。外部メモリとしての接続例と NMI 端子に接続するスイッチの接続例を図 1 に示し、7セグメント LED 表示回路を図 2 に示します。

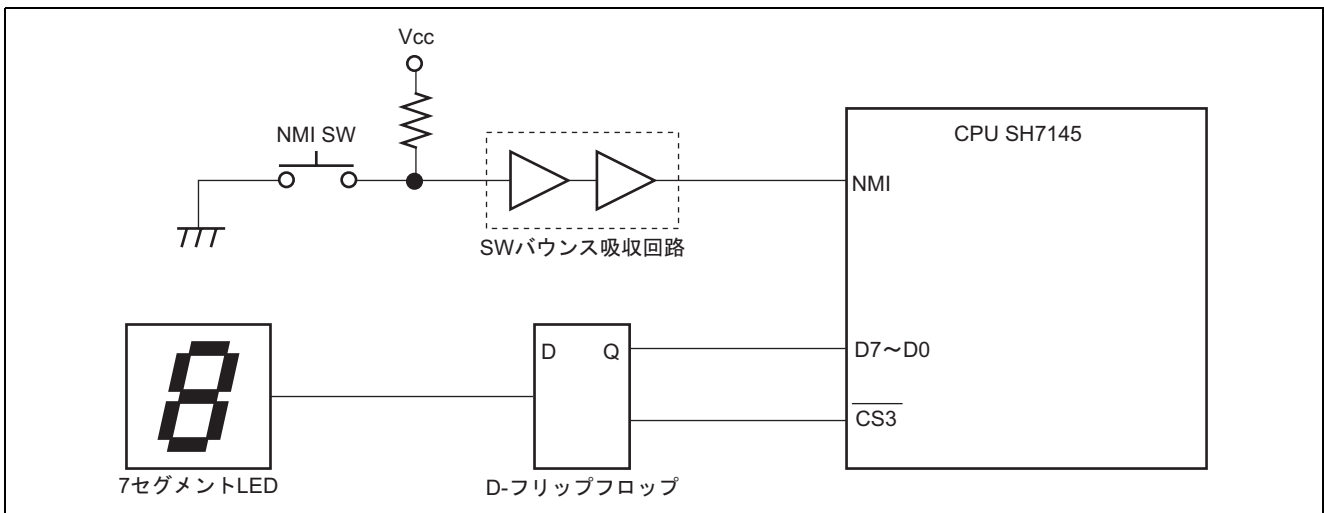


図 1 構成図

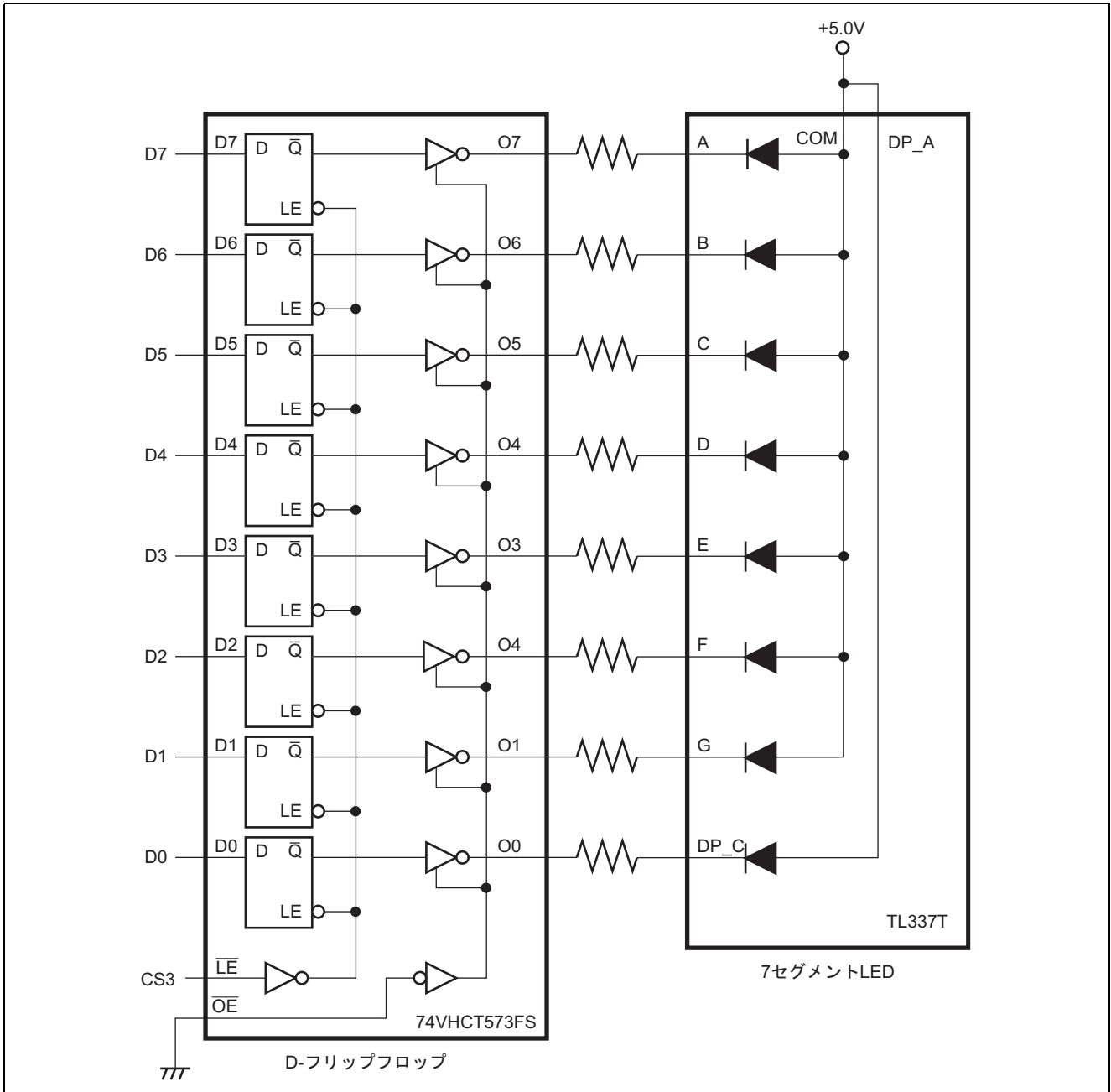


図2 7セグメントLED等価回路図

2. 使用機能説明

本タスク例では、ソフトウェアスタンバイモードへの遷移および復帰を行います。図3に、その遷移および復帰を示します。以下にソフトウェアスタンバイモードへの遷移および復帰に必要な機能説明を示します。

2.1 低消費電力状態

SH7145には、通常のプログラム実行状態のほかに、CPUや発振器の動作を停止し、消費電力を低くする低消費電力状態があります。CPU、内蔵周辺機能などを個別に制御して、低消費電力化を実現できます。SH7145の動作状態には、通常動作モードの他、スリープモード、ソフトウェアスタンバイモード、モジュールスタンバイモードの低消費電力状態があります。

ソフトウェアスタンバイモードへの遷移は、SBYCRのSSBYビットを1にセットした状態でSLEEP命令を実行することで遷移します。このモードでは、CPU、内蔵周辺機能、および発振器のすべての機能が停止します。ただし、CPUのレジスタ内容は、規定の電圧が与えられている限り保持されます。本モードでは、発振器が停止するため、消費電力は著しく低減します。

ソフトウェアスタンバイモードからの復帰には、NMI端子の立ち下がりエッジまたは立ち上がりエッジを検出し、ウォッチドッグタイマ(WDT)のみに供給されるクロック発振が開始され、WDTのオーバーフロー発生によってクロックが安定したと判断され、本LSI全体にクロックが供給されることによって、ソフトウェアスタンバイモードを解除し、通常のプログラム実行に復帰します。その後、NMI例外処理が開始されます。

スタンバイコントロールレジスタ(SBYCR)は8ビットのリード/ライト可能なレジスタで、ソフトウェアスタンバイモードの制御を行います。

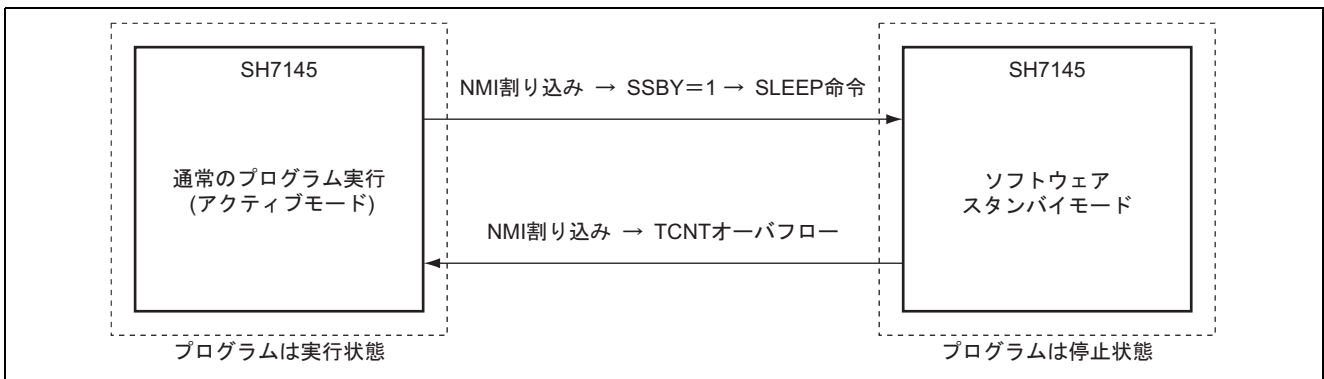


図3 ソフトウェアスタンバイモードへの遷移および復帰図

2.2 ウォッチドッグタイマ (WDT)

ウォッチドッグタイマ (WDT) は 8 ビットのタイマで、ソフトウェアスタンバイモードが NMI 割り込みで解除されるときに使用されます。WDT のブロック図を図 4 に示します。

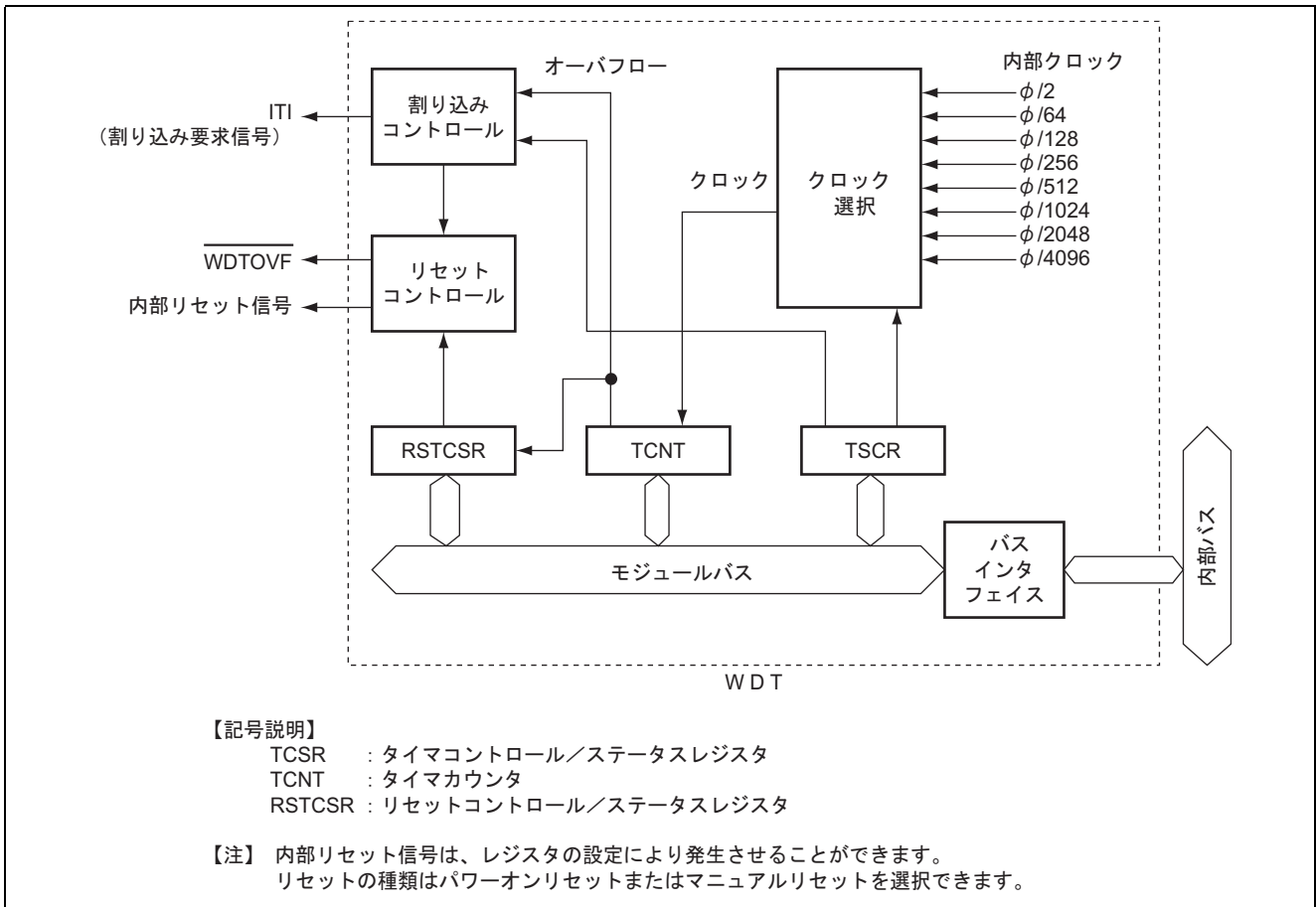


図 4 WDT ブロック図

ソフトウェアスタンバイモード遷移前の設定として、タイマコントロール/ステータスレジスタ (TCSR) の TME ビットを 0 にして、WDT を停止させてください。TME ビットが 1 になっていると、ソフトウェアスタンバイモードに遷移できません。また、タイマカウンタ (TCNT) のオーバーフロー周期が発振安定時間以上になるように、TCSR の CKS2~CKS0 ビットを設定してください。発振安定時間については、表 1 に示します。

表 1 クロックタイミング

条件 : VCC=PLL_{Vcc}=3.3V±0.3V、VSS=PLL_{VSS}=0V、Ta=-20~+75° C (標準品)

項目	最小時間	単位
スタンバイ復帰発振安定時間	10	ms

ソフトウェアスタンバイモード解除時の動作は、ソフトウェアスタンバイモードで NMI 信号が入力されると、発振器が動作を開始し、TCNT はソフトウェアスタンバイモード遷移前に CKS2~CKS0 ビットで選択しておいたクロックにより、カウントアップを開始します。TCNT がオーバーフロー (H'FF→H'00) すると、クロックが安定し使用可能であると判断され、本 LSI 全体にクロックが供給されます。これによって、ソフトウェアスタンバイモードが解除されます。

タイマカウンタ (TCNT) は、読み出し/書き込み可能な 8 ビットのアップカウンタです。タイマコントロール/ステータスレジスタ (TCSR) のタイマイネーブルビット (TME) を 1 にすると、TCSR の CKS2~CKS0 ビットで選択した内部クロックにより、TCNT はカウントアップを開始します。TCNT の初期値は H'00 です。

タイマコントロール/ステータスレジスタ (TCSR) は、リード/ライト可能な 8 ビットのレジスタで、TCNT に入力するクロック、モードの選択などを行います。

2.3 割り込み要因

NMI 割り込みは、レベル 16 の割り込みで、常に受け付けられます。NMI 端子からの入力はエッジで検出され、エッジ種別の選択は、割り込みコントローラ (INTC) の割り込みコントロールレジスタ 1 (ICR1) の NMI エッジセレクトビット (NMIE) の設定によって立ち上がりエッジまたは立ち下がりエッジを選択できます。

割り込みコントロールレジスタ 1 (ICR1) は、16 ビットのレジスタで、外部割り込み入力端子 NMI の NMI エッジセレクトを設定し、NMI 端子への入力レベルを示します。表 2 に本タスク例におけるソフトウェアスタンバイモードへの遷移および復帰に対する割り込み動作を示します。

表 2 NMI 割り込み動作

状態遷移	割り込み要因	割り込み検出	クロック供給状態	遷移条件
ソフトウェアスタンバイモードへの遷移	MNI 割り込み	NMI 端子の立ち下がりエッジ	遷移後、本 LSI 全体のクロック供給停止	WDT が停止していること
通常のプログラム実行状態への遷移	MNI 割り込み	NMI 端子の立ち上がりエッジ	WDT のみクロックが供給され、遷移後、本 LSI 全体となる	WDT の TCNT がオーバーフローすること

2.4 バスステートコントローラ (BSC)

バスステートコントローラ (BSC) はアドレス空間の分割、各種メモリに応じた制御信号の出力などを行います。これにより、外付け回路なしに SRAM、ROM などを本 LSI に直結することができます。BSC のブロック図を図 5 に示します。本タスク例では、外部メモリ (CS3) としてアクセスし、7セグメント LED の表示制御を行います。

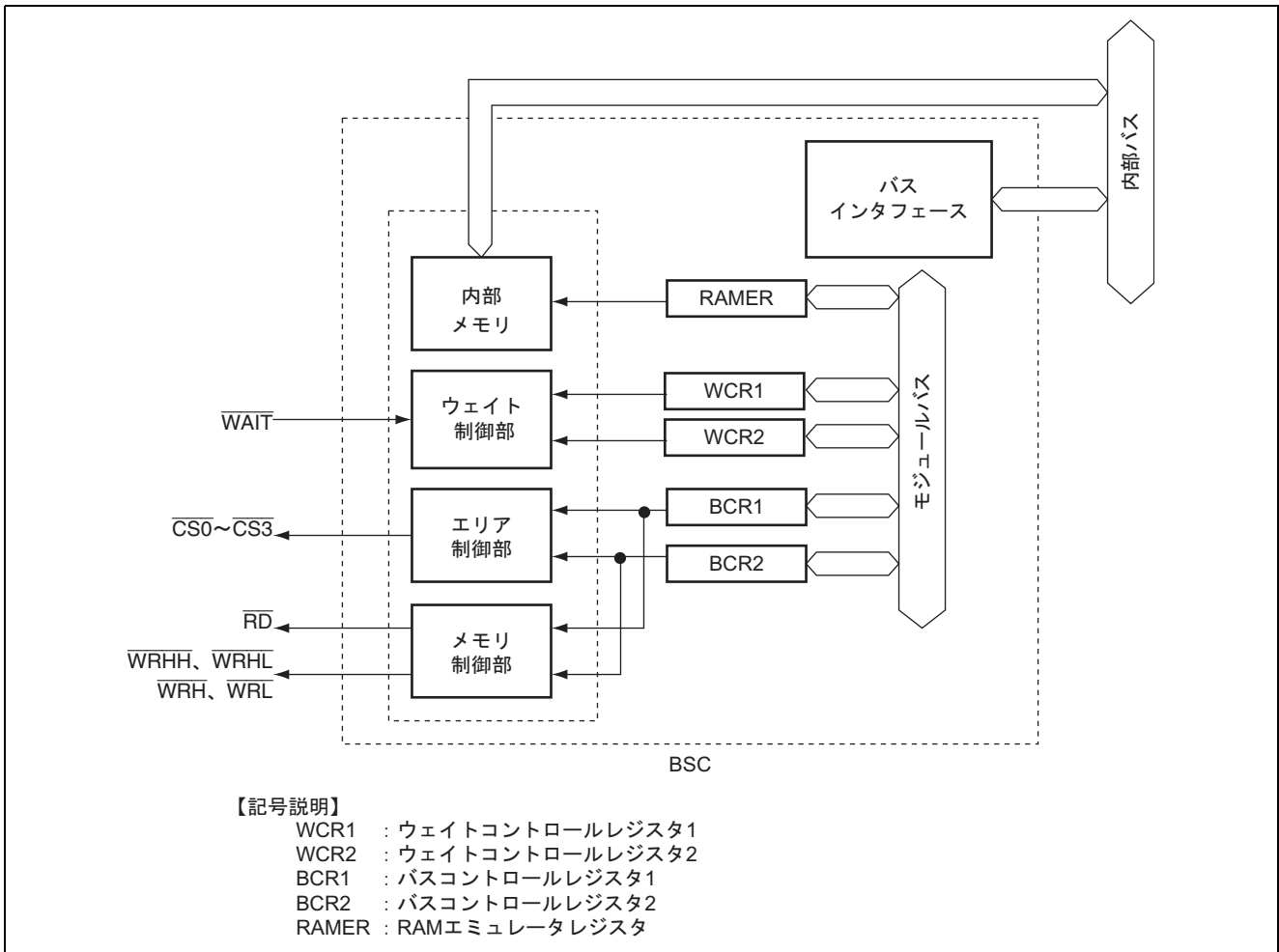


図 5 BSC ブロック図

外部空間アクセスのバスサイクルは、2 ステートで行われます。図 6 に外部空間アクセスの基本タイミングを示します。読み出し時は、オペランドサイズにかかわらず、RD 信号によりアクセスする空間（アドレス）のデータバス幅全ビットを LSI に取り込み、内部で必要なバイトを選択して使用します。書き込み時は、実際に書き込むバイト位置を WRHH（ビット 31～24）、WRHL（ビット 23～16）、WRH（ビット 15～8）、WRL（ビット 7～0）の各信号で示します。

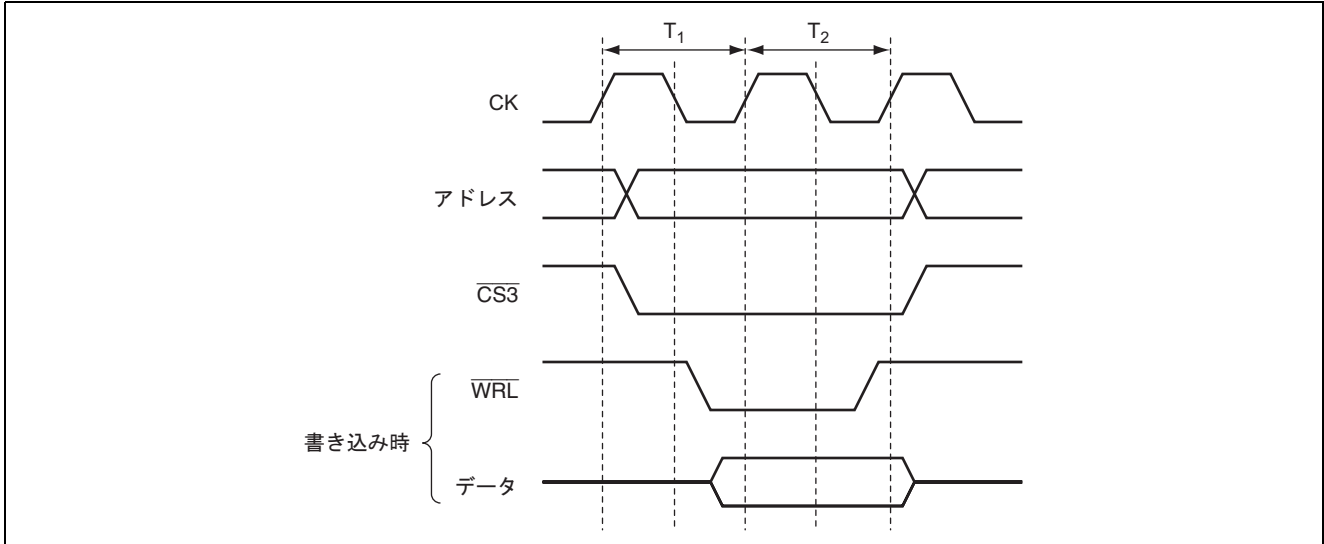


図 6 外部空間アクセスの基本タイミング

表 3 に 7 セグメント LED のメモリ割り付けを示します。

表 3 メモリ割り付け表

エリア	割付けアドレス	デバイスの種類	バスサイズ	アクセスウェイト
CS3	H'00C00000～	7 セグメント LED	8 (Bit)	なし

図 7 に 7 セグメント LED の表示対応図と、表 4 にセグメント対応表を示します。また、各セグメントは負論理点灯となります。

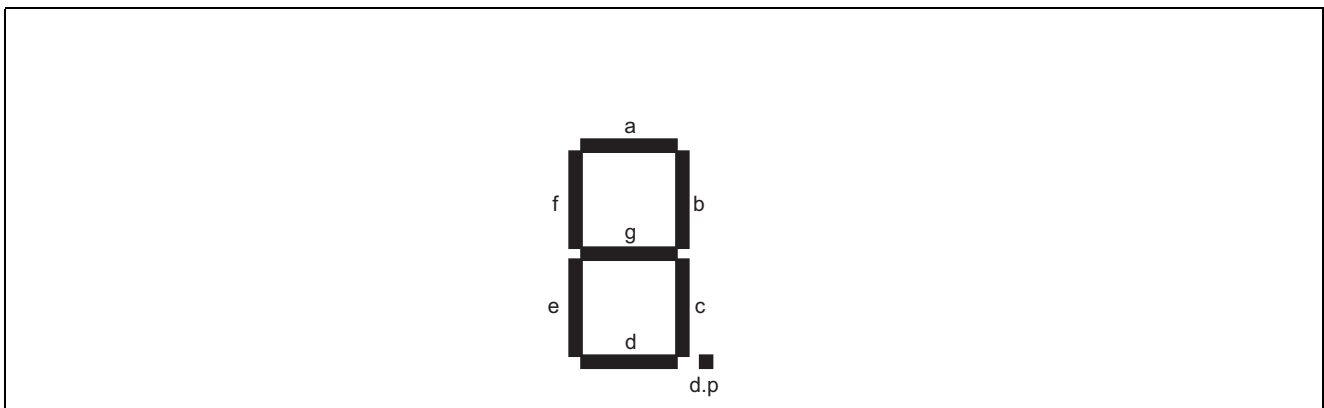


図 7 7 セグメント LED の表示対応図

表 4 セグメント対応表

※負論理点灯(0=点灯、1=消灯)

CS3 データバス名	D7	D6	D5	D4	D3	D2	D1	D0
セグメント	dp	g	f	e	d	c	b	a
「A」文字表示の場合	1	0	0	0	1	0	0	0
「S」文字表示の場合	1	0	0	1	0	0	1	0
「.」文字表示の場合	0	1	1	1	1	1	1	1

バスコントロールレジスタ 1 (BCR1) は、読み出し/書き込み可能な 16 ビットのレジスタで、MTU の制御レジスタの書き込み許可指定と各 CS 空間のバスサイズ指定を行います。BCR1 のビット 7~0 はパワーオンリセット後の初期設定時に書き込みを行い、以後は値を変更しないでください。内蔵 ROM 有効モードの場合、レジスタの初期設定が終了するまで各 CS 空間はアクセスしないでください。内蔵 ROM 無効モードの場合、レジスタの初期設定が終了するまで CS0 空間以外の CS 空間はアクセスしないでください。

※本タスク例では、内蔵 ROM 有効モードで動作しています。

ウェイトコントロールレジスタ 1 (WCR1) は、読み出し/書き込み可能な 16 ビットのレジスタで、各 CS 空間のウェイトサイクル数 (0~15) を指定します。

2.5 ピンファンクションコントローラ (PFC)

ピンファンクションコントローラ (PFC) は、マルチプレクス端子の機能とその入出力の方向を選ぶためのレジスタで構成されています。

ポート D コントロールレジスタ L1、L2 (PDCRL1、L2) は、それぞれ 16 ビットの読み出し/書き込み可能なレジスタで、ポート D にあるマルチプレクス端子の機能を選びます。本タスク例では、データバスを選択し (D0~D7)、7セグメント LED の表示制御を行います。

2.6 機能割り付け

表 5 に本タスク例の機能割り付けを示します。

表 5 機能割り付け

機能	機能割り付け
SBYCR	SLEEP 命令実行後、ソフトウェアスタンバイモードへの遷移するためのレジスタ
TCSR	WDT のタイマカウンタを制御するためのレジスタ
ICR1	外部割り込み入力端子 (NMI) の入力信号検出モードを設定するためのレジスタ
BCR1	CS 空間のバスサイズを指定するためのレジスタ
WCR1	CS 空間のウェイトサイクル数を指定するためのレジスタ
PDCRL1	ポート D にあるマルチプレクス端子 (ポート D またはデータバス) の機能を選ぶためのレジスタ

3. 動作

図8に動作を示します。図8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ソフトウェアスタンバイモードへの遷移および復帰動作を行います。

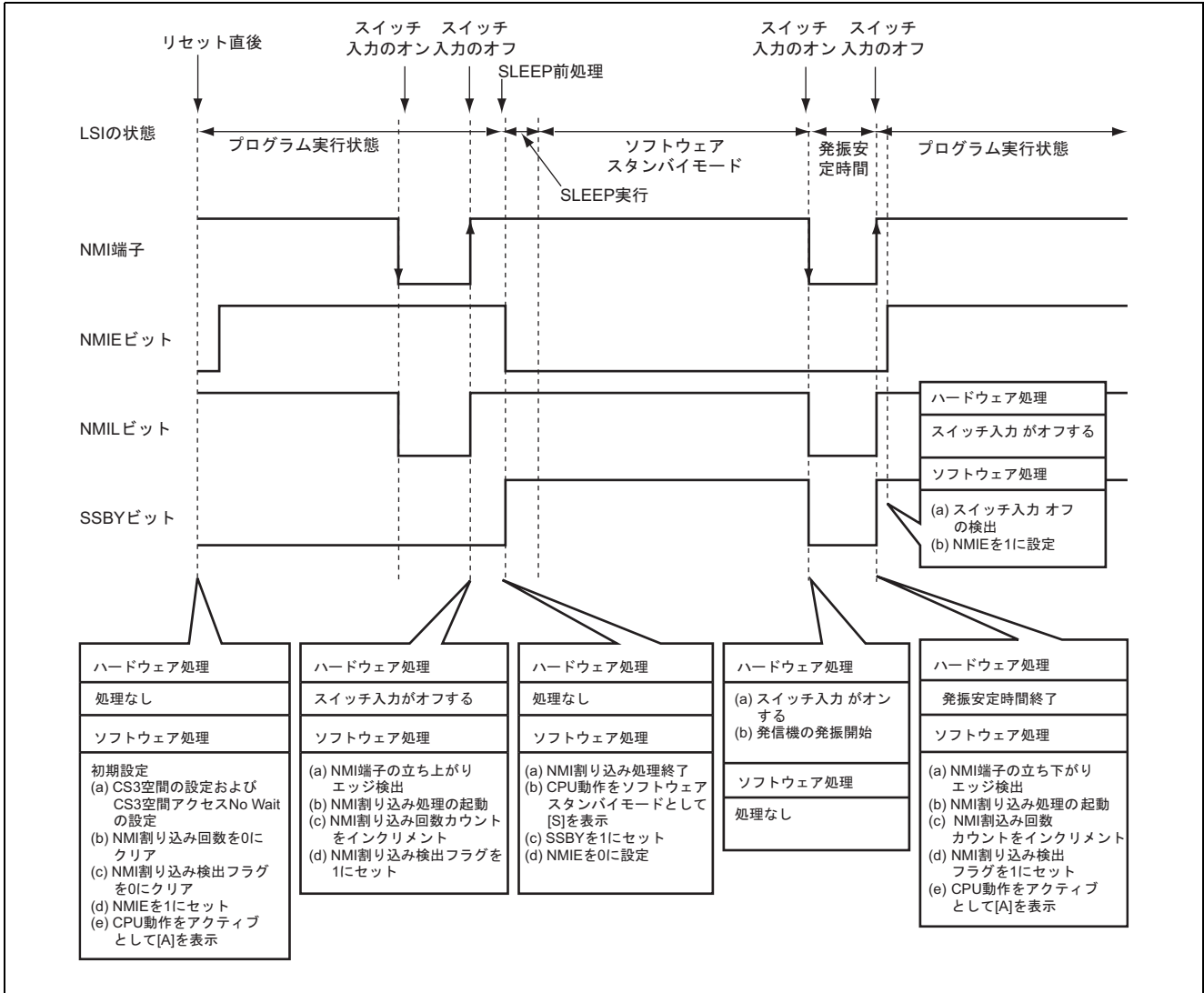


図8 ソフトウェアスタンバイモードへの遷移および復帰の動作

4. ソフトウェア説明

4.1 モジュール説明

表 6 に本タスク例におけるモジュール説明を示します。

表 6 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	初期設定の呼び出し、遷移状態を示す 7 セグメント LED 表示、WDT カウンタの停止、ソフトウェアスタンバイモードへのスタンバイコントロールレジスタの設定、MNI 割り込み検出エッジの設定、スリープ命令によるソフトウェアスタンバイモードへの遷移、通常のプログラム実行への遷移後の初期化処理
7 セグメント LED 表示処理	led7	7 セグメント LED 表示
7 セグメント LED 表示データテーブル設定処理	tblset	初期設定処理で 7 セグメント LED 表示データテーブルを設定
初期設定処理	inisub	BSC でのアクセス幅とウェイト設定、PFC でのデータバス指定、NMI 割り込み検出フラグの初期化、NMI 割り込み回数カウンタの初期化、NMI 割り込み検出エッジの設定、7 セグメント LED 表示の初期設定の呼び出し
書き込み処理	wri	任意のアドレスに対してロング型データの書き込み
ウェイト処理	wait	ウェイト動作
NMI 割り込み処理	nmisub	NMI 割り込み検出フラグの設定、NMI 割り込み回数のカウント

4.2 引数の説明

表 7 に本タスク例における引数の説明を示します。

表 7 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
data	7 セグメント LED 表示のデータを指定	7 セグメント LED 表示ルーチン	1 バイト	入力
dot	7 セグメント LED にドット表示をするか指定	7 セグメント LED 表示ルーチン	1 バイト	入力

4.3 使用内部レジスタ説明

表 8 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 8 使用内部レジスタ説明

レジスタ名		機能	アドレス	設定値
TCSR	WT/IT	タイマコントロール/ステータスレジスタ(タイマモードセレクト) : WT/IT="0"のとき、インターバルタイマモード	H'FFFF8610 ビット 6	0
	TME	タイマコントロール/ステータスレジスタ(タイマイネーブル) : TME="0"のとき、タイマディスエーブル	H'FFFF8610 ビット 5	0
	CKS2 CKS1 CKS0	タイマコントロール/ステータスレジスタ(クロックセレクト 2~0) : CKS2="1"、CKS1="1"、CKS0="0"のとき、クロック 1/4096(周期 26.2ms)を設定	H'FFFF8610 ビット 2、 ビット 1、ビット 0	CKS2 =1 CKS1 =1 CKS0 =0
ADCR_0	TRGE	A/D コントロールレジスタ_0(トリガイネーブル) : TRGE="0"のとき、トリガによる開始は無効	H'FFFF8488 ビット 7	0
SBYCR	SSBY	スタンバイコントロールレジスタ(ソフトウェアスタンバイ) : SSBY="1"のとき、SLEEP 命令実行後、ソフトウェアスタンバイモードに遷移	H'FFFF8614 ビット 7	1
	HIZ	スタンバイコントロールレジスタ(ポートハイインピーダンス) : HIZ="0"のとき、ソフトウェアスタンバイモード時に、端子状態を保持する	H'FFFF8614 ビット 6	0
	IRQEH	スタンバイコントロールレジスタ(IRQ7~4 イネーブル) : IRQEH="1"のとき、IRQ7~IRQ4 によるソフトウェアスタンバイモードの解除を無効にする	H'FFFF8614 ビット 1	1
	IRQEL	スタンバイコントロールレジスタ(IRQ3~0 イネーブル) : IRQEL="1"のとき、IRQ3~IRQ0 によるソフトウェアスタンバイモードの解除を無効にする	H'FFFF8614 ビット 0	1
ICR1	NMIL	割り込みコントロールレジスタ 1(NMI 入力レベル) : NMIL="0"のとき、NMI 端子にローレベルが入力されている : NMIL="1"のとき、NMI 端子にハイレベルが入力されている	H'FFF8358 ビット 15	—
	NMIE	割り込みコントロールレジスタ 1(NMI エッジセレクト) : NMIE="0"のとき、NMI 入力の立ち下がりエッジで割り込み要求を検出 : NMIE="1"のとき、NMI 入力の立ち上がりエッジで割り込み要求を検出	H'FFF8358 ビット 8	【立ち下がりエッジのとき】 0 【立ち上がりエッジのとき】 1

レジスタ名		機能	アドレス	設定値
BCR1	A3LG	バスコントロールレジスタ 1 (CS3 空間ロングサイズ指定) : A3LG="0" のとき、本レジスタ中の A3SZ ビットで設定される値に従います	H' FFF8620 ビット 7	0
	A3SZ	バスコントロールレジスタ 1 (CS3 空間サイズ指定) : A3SZ="0" のとき、バイト (8 ビット) サイズ	H' FFF8620 ビット 3	0
WCR1	W33 W32 W31 W30	ウェイトコントロールレジスタ 1 (CS3 空間ウェイト指定) : W33="0"、W32="0"、W31="0"、W30="0" のとき、ノーウェイト (外部ウェイト入力禁止)	H' FFF8624 ビット 15 ビット 14 ビット 13 ビット 12	W33=0 W32=0 W31=0 W30=0
PDCRL1	PD7MD	ポート D コントロールレジスタ L1 (PD7 モードビット) : PD7MD="1" のとき、D7 入出力 (BSC)	H' FFFF83AD ビット 7	PD7MD=1
PDCRL1	PD6MD	ポート D コントロールレジスタ L1 (PD6 モードビット) : PD6MD="1" のとき、D6 入出力 (BSC)	H' FFFF83AD ビット 6	PD6MD=1
PDCRL1	PD5MD	ポート D コントロールレジスタ L1 (PD5 モードビット) : PD5MD="1" のとき、D5 入出力 (BSC)	H' FFFF83AD ビット 5	PD5MD=1
PDCRL1	PD4MD	ポート D コントロールレジスタ L1 (PD4 モードビット) : PD4MD="1" のとき、D4 入出力 (BSC)	H' FFFF83AD ビット 4	PD4MD=1
PDCRL1	PD3MD	ポート D コントロールレジスタ L1 (PD3 モードビット) : PD3MD="1" のとき、D3 入出力 (BSC)	H' FFFF83AD ビット 3	PD3MD=1
PDCRL1	PD2MD	ポート D コントロールレジスタ L1 (PD2 モードビット) : PD2MD="1" のとき、D2 入出力 (BSC)	H' FFFF83AD ビット 2	PD2MD=1
PDCRL1	PD1MD	ポート D コントロールレジスタ L1 (PD1 モードビット) : PD1MD="1" のとき、D1 入出力 (BSC)	H' FFFF83AD ビット 1	PD1MD=1
PDCRL1	PD0MD	ポート D コントロールレジスタ L1 (PD0 モードビット) : PD0MD="1" のとき、D0 入出力 (BSC)	H' FFFF83AD ビット 0	PD0MD=1

4.4 使用 RAM 説明

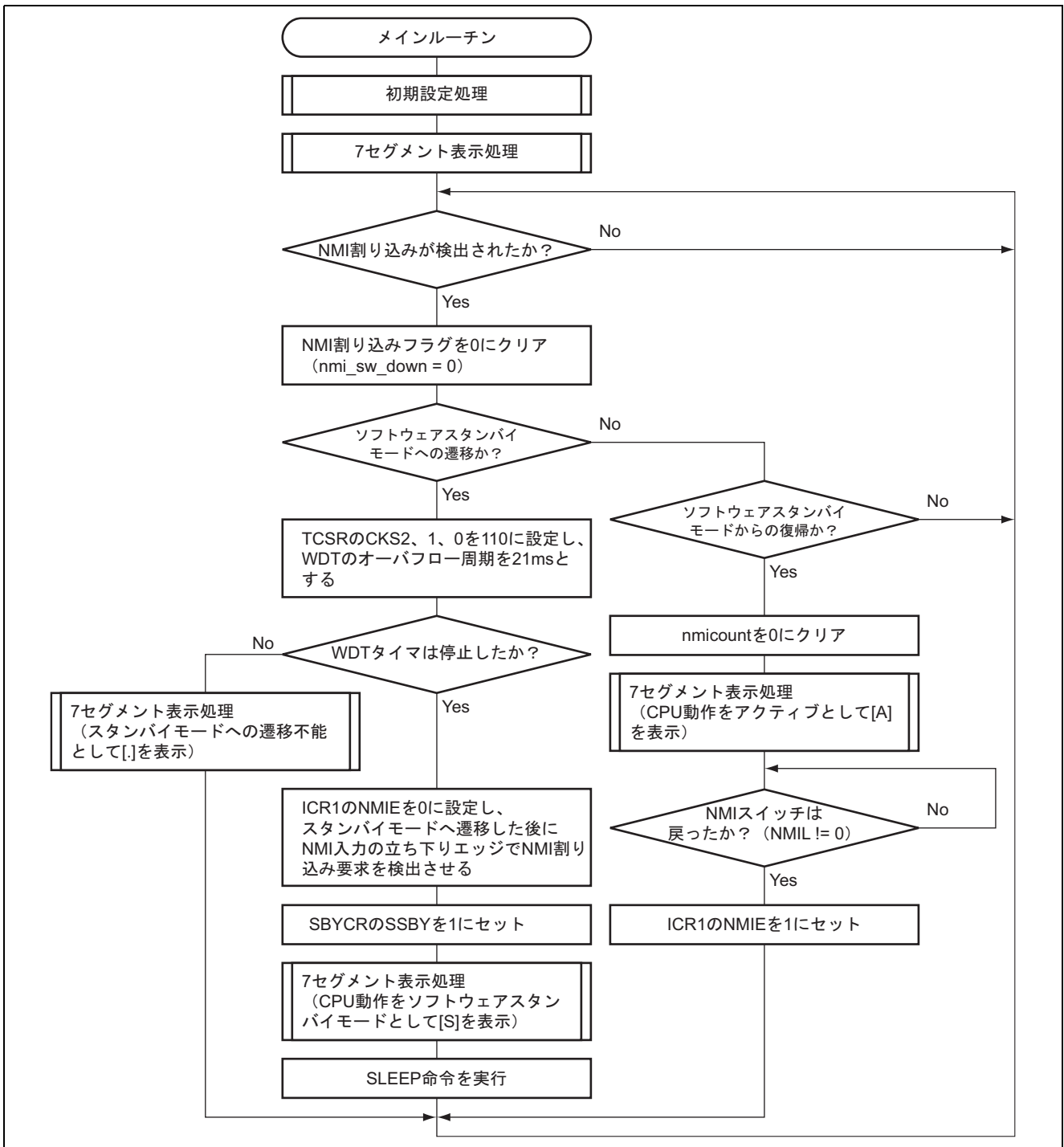
表 9 に本タスク例で使用する RAM の説明を示します。

表 9 使用 RAM 説明

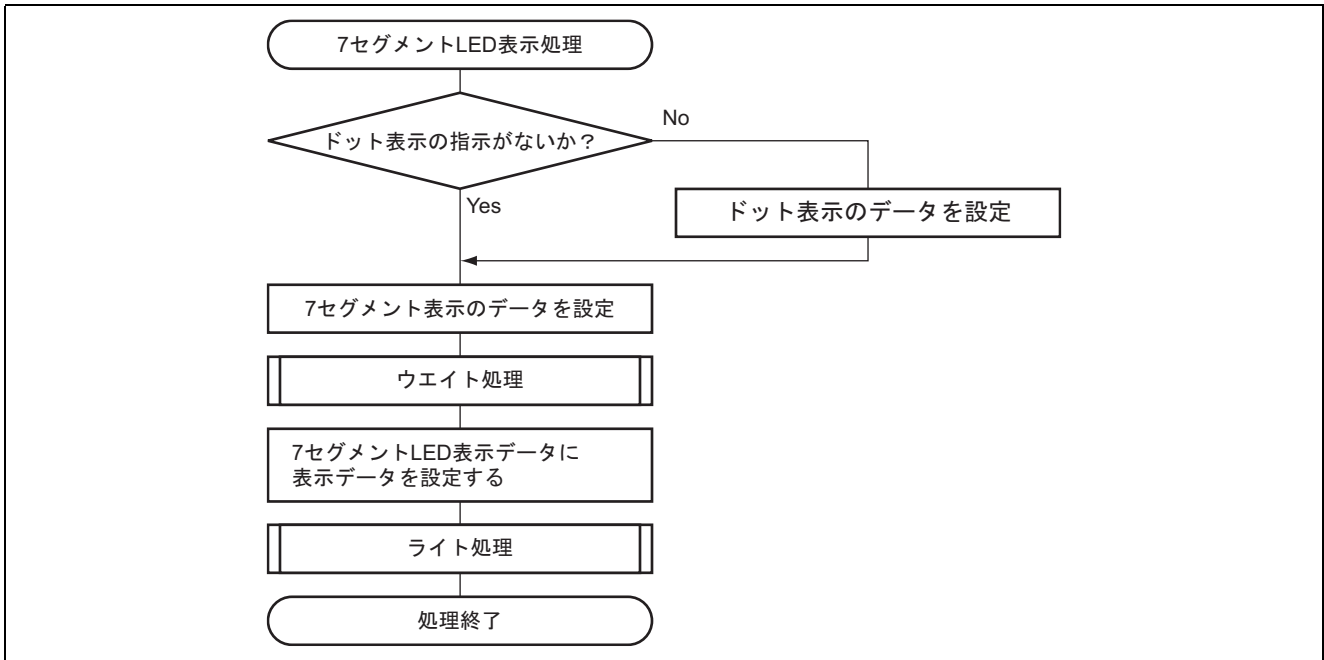
ラベル名	機能	アドレス	使用モジュール名
table[0]	文字“A”を表示するデータを格納	H' 00407005	7セグメントLED表示ルーチン、 7セグメントLED表示データテーブル設定ルーチン
table[1]	文字“S”を表示するデータを格納	H' 00407006	7セグメントLED表示ルーチン、 7セグメントLED表示データテーブル設定ルーチン
table[2]	文字“ ”を表示するデータを格納	H' 00407007	7セグメントLED表示ルーチン、 7セグメントLED表示データテーブル設定ルーチン
nmicount	NMI 割り込みの回数データを格納	H' 00407000	メインルーチン、初期設定ルーチン
nmi_sw_down	NMI 割り込み検出フラグデータを格納	H' 00407004	メインルーチン、初期設定ルーチン

5. フローチャート

5.1 メインルーチン



5.2 セグメント LED 表示処理



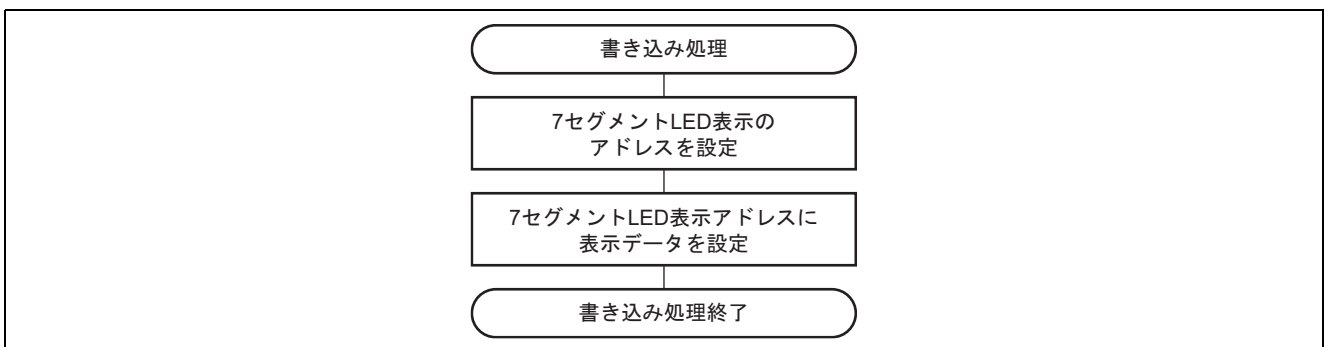
5.3 7セグメント LED 表示データテーブル設定処理



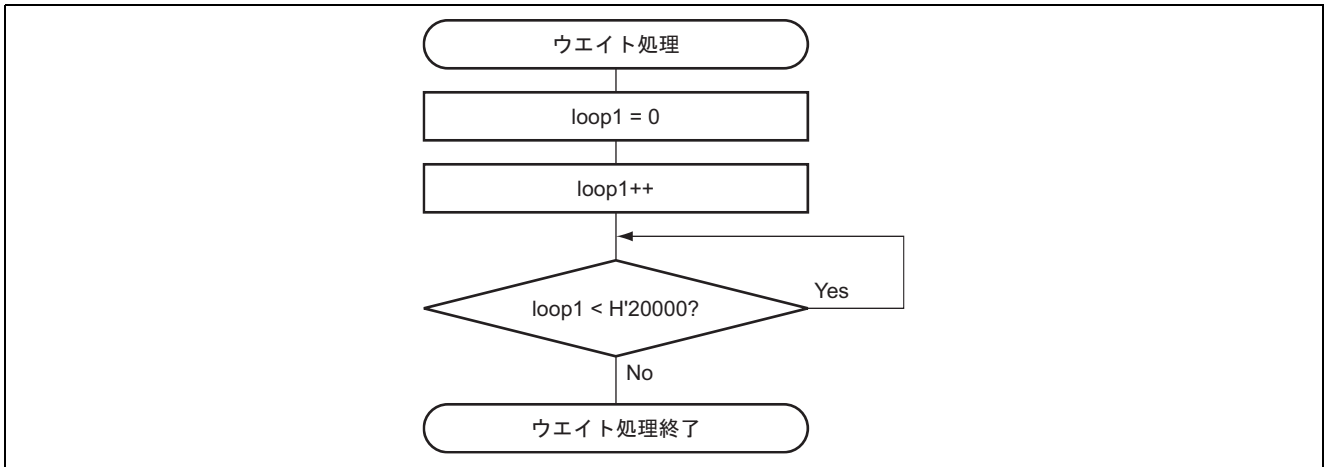
5.4 初期設定処理



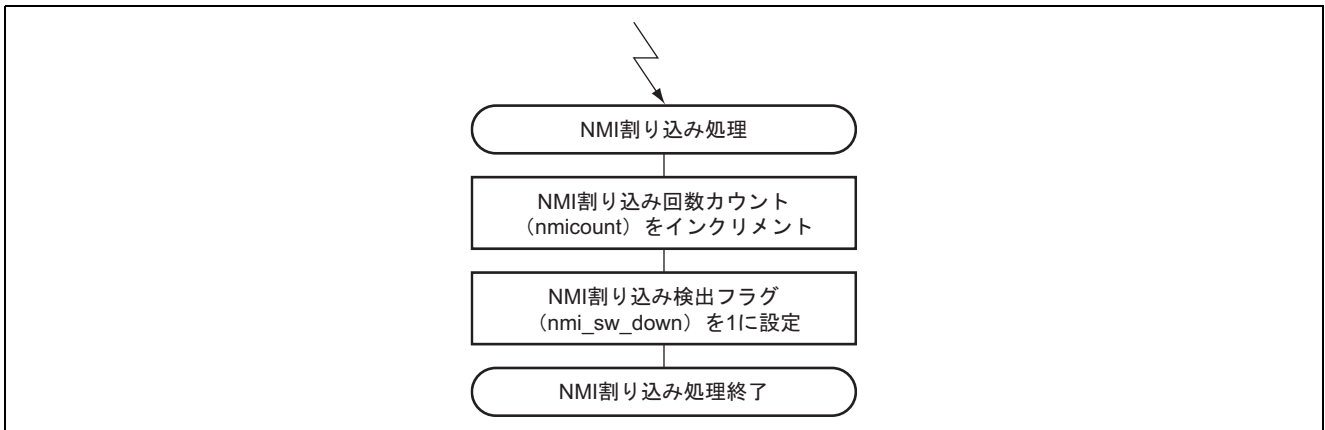
5.5 書き込み処理



5.6 ウェイト処理



5.7 NMI 割り込み処理



5.8 リンクアドレス指定

セクション名	アドレス
P,C	H'00002000
D,B	H'00407000
__STACK()	H'00407F00

※: ヘッダファイル"IODEFINE.H"は、下記の HP にて入手。ターゲットのヘッダ名は、"7144S.H"。
<http://www.renesas.com/jpn/products/mpumcu/tool/crosstool/iodef/index.html>

6. プログラムリスト

INIT.C (プログラムリスト)

```
extern  _INITSCT(void) ;
extern  void main(void) ;
```

```
void _INIT()
{
    _INITSCT() ;
    main() ;
    for(;;) ;
}
```

VEC_TBL.SRC (プログラムリスト)

```
.SECTION VECT,DATA,LOCATE=H'0000
```

```
.IMPORT  __INIT
.IMPORT  __STACK
.IMPORT  _nmisub
```

```
.ORG      H'00000000
.DATA.L   __INIT
    .DATA.L   __STACK
    .DATA.L   __INIT
    .DATA.L   __STACK
```

```
.ORG      H'0000002c
.DATA.L   _nmisub
.END
```

```

Standby.c (プログラムリスト)
/* SH7145 Series -SH7145- Application note */
/* Application Note */
/* Transition To Software Standby Mode And Return From Software */
/* Standby Mode By NMI Interrupt */
/*
/* Function : Transition To Software Standby Mode */
/*           And Return From Software Standby Mode By */
/*           NMI Interrupt */
/*
/* Peripheral Clock : 25MHz */
/* Internal Clock : 50MHz */
/*****/

/*****/
/* Include File */
/*****/
#include <machine.h>
#include "IODEFINE.H"

/*****/
/* Function Prototype */
/*****/
void inisub(void); /* Initialize Cofiguration */
void wr1(unsigned long addr, unsigned long data); /* Write Function (Long)
*/
void led7(unsigned char data,unsigned char dot); /* 7 Segment LED Display
Function */
void tblset(void); /* Set 7 Segment LED Display Table Function*/
void wait(void); /* Wait Function */
void main(void); /* Control Software Standby Mode Function */
#pragma interrupt(nmisub) /* NMI Interrupt Handler Function */

/*****/
/* RAM Allocation */
/*****/
unsigned long nmicount; /* NMI Interrupt Counter */
unsigned char nmi_sw_down; /* NMI Switch */
unsigned char table[4]; /* 7 Segment LED Display Data Table */
/*****/
/* I/O Definition */
/*****/
#define LED 0x00C00000 /* LED Port */

/*****/
/* Function Definition(Main Program) */
/*****/
void main(void)
{

    inisub(); /* Initialize Cofiguration */
    led7(0,0); /* Output "A" on 7 Segment LED Display*/

    while(1){ /* Check Software Standby Mode Status*/
        if( nmi_sw_down == 1 ){ /* NMI Switch is Down? */
            nmi_sw_down = 0; /* Clear NMI Switch Down Flag */
            if( nmicount == 1 ){ /* NMI Interrupt Counter is 1? */

```

```

/*****/
/* Set WDT Overflow Cycles by TCSR Register */
/* Cycles = ( CPU Clock / 4096 ) * 256(TCNT Clock) = 21[ms] */
/*****/
    WDT.WRITE.TCSR = 0x1e;
        /* WDT Timer Counter is Stopped,Set WDT Overflow Cycles */

    if( WDT.READ.TCSR.BIT.TME == 0 ){ /* WDT Timer Counter is Stopped? */
        INTC.ICR1.BIT.NMIE = 0x0; /* NMI Falling Edge Interrupt */
        SBYCR.BYTE = 0x9f; /* Set SSBY Bit */
        led7(1,0); /* Output "S" on 7 Segment LED
Display*/
        sleep(); /* Transition To Software Standby
Mode */
    }else { /* WDT Timer Counter is Working */
        led7(2,1); /* Output "." on 7 Segment LED Display*/
    }
}else if( nmicount == 2 ){ /* NMI Interrupt Counter is 2? */
    nmicount = 0; /* Clear NMI Interrupt Counter */
    led7(0,0); /* Output "A" on 7 Segment LED Display*/
    while( INTC.ICR1.BIT.NMIL == 0x0); /* Check NMI Switch Release */
    INTC.ICR1.BIT.NMIE = 0x1;
        /* NMI Rasing Edge Interrupt,Return From Software Standby Mode */
    }
}
}
}
}
/*****/
/* Initial Cofiguration */
/*****/
void inisub(void)
{
    BSC.BCR1.BIT.A3LG = 0x0; /* Clear A3LG Bit for A3SZ Bit */
    BSC.BCR1.BIT.A3SZ = 0x0; /* Byte Size CS3 Memory Area */
    BSC.WCR1.BIT.W3 = 0x0; /* No Wait CS3 Memory Area */

    PFC.PDCRL1.BYTE.L = 0xff; /* Use From D0 To D7 Input And Output(BSC) Port */
    PFC.PDCRL2.BYTE.L = 0x0; /* Use From D0 To D7 Input And Output(BSC) Port */

    nmicount = 0; /* Initialize NMI Interrupt Counter */
    nmi_sw_down = 0; /* Initialize NMI Switch */
    INTC.ICR1.BIT.NMIE = 0x1; /* Set NMI Rasing Edge Interruptt */
    tblset(); /* Set 7 Segment LED Display Table */
}
/*****/
/* Write Function */
/*****/
/* Write Function (Long) */
void wrl(unsigned long addr, unsigned long data)
{
    unsigned long *paddr = ((unsigned long *) addr); /* Set paddr as LED Port
*/
    *paddr = data; /* Save data */
}

```

```

/*****
/* 7 Segment LED Display
/*****
void led7(unsigned char data,unsigned char dot)
{
    unsigned char leddata;
    unsigned long wleddata;

    if( dot!=0 ){
        leddata=(table[data]|0x80); /* Output "." on 7 Segment LED Display? */
        /* Output "." */
    } else {
        leddata=(table[data]&0x7f); /* 7 Segment LED Display Output Value */
    }

    wait(); /* Wait */

    wleddata = leddata; /* Save 7 Segment LED Display Output Value */
    wr1(LED,~wleddata); /* Output 7 Seg LED Display */
}

/*****
/*Set 7 Segment LED Display Table
/*****
void tblset(void)
{
    table[0]=0x77; /* 7 Segment LED Display Data 'A' */
    table[1]=0x6d; /* 7 Segment LED Display Data 'S' */
    table[2]=0x00; /* 7 Segment LED Display Data ' ' */
}

/*****
/* NMI Interrupt Function
/*****
void nmisub(void)
{
    nmicount++; /* Increment NMI Interrupt Counter */
    nmi_sw_down = 1; /* Set NMI Switch Down */
}

/*****
/* Wait Function
/*****
void wait(void)
{
    unsigned long loop1;
    for(loop1=0;loop1<0x20000;loop1++){ /* wait */
    }
}

```


改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2004.09.15	—	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。