

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、誤りがな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。

アプリケーション・ノート

V850ES/Jx3-L サンプル・プログラム (16ビット・タイマ/イベント・カウンタ(TMP, TMQ)) パルス幅測定モード編

この資料は、サンプル・プログラムの動作概要や使用方法、および16ビット・タイマ/イベント・カウンタ(TMP, TMQ)のパルス幅測定機能の設定方法や活用方法を説明したものです。サンプル・プログラムでは、16ビット・タイマ/イベント・カウンタ(TMP)のパルス幅測定機能を使用して、SW1の押下時間分をパルス幅として測定し、SW1の押下時間に合わせてLED1を点灯させます。

対象デバイス

V850ES/JF3-L マイクロコントローラ
V850ES/JG3-L マイクロコントローラ

目次

第1章 概要	...	3
1.1 初期化	...	5
1.2 SW1入力による外部割り込み処理(INTP0)	...	6
1.3 16ビット・タイマ/イベント・カウンタ(TMP0)を使用したキャプチャ割り込み処理	...	6
第2章 回路図	...	7
2.1 回路図	...	7
2.2 周辺ハードウェア	...	8
第3章 ソフトウェアについて	...	9
3.1 ファイル構成	...	9
3.2 使用する内蔵周辺機能	...	10
3.3 初期設定と動作概要	...	10
3.4 フロー・チャート	...	12
3.5 V850ES/JG3-LとV850ES/JF3-Lの違い	...	16
3.6 TMPとTMQの違い	...	16
3.7 セキュリティIDについて	...	16
第4章 レジスタ設定について	...	17
4.1 16ビット・タイマ/イベント・カウンタ(TMP)の設定	...	18
4.2 SW1入力用割り込み端子設定	...	29
4.3 パルス幅測定モード用キャプチャ割り込み端子設定	...	33
4.4 LED出力用端子設定	...	36
第5章 関連資料	...	38
付録A プログラム・リスト	...	39

資料番号 U20013JJ1V0AN00
発行年月 September 2009 NS

- 本資料に記載されている内容は2009年09月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
- 当社は、本資料に記載された当社製品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、一切その責を負いません。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。
- 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、一切その責を負いません。
- 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。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。当社製品をお客様の機器にご使用の際には、当社製品の不具合の結果として、生命、身体および財産に対する損害や社会的損害を生じさせないように、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を行ってください。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「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。

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

特別水準：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

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、生命維持のための装置またはシステム等

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。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

(注)

- (1) 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNEC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う。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

(M8E0909J)

第1章 概 要

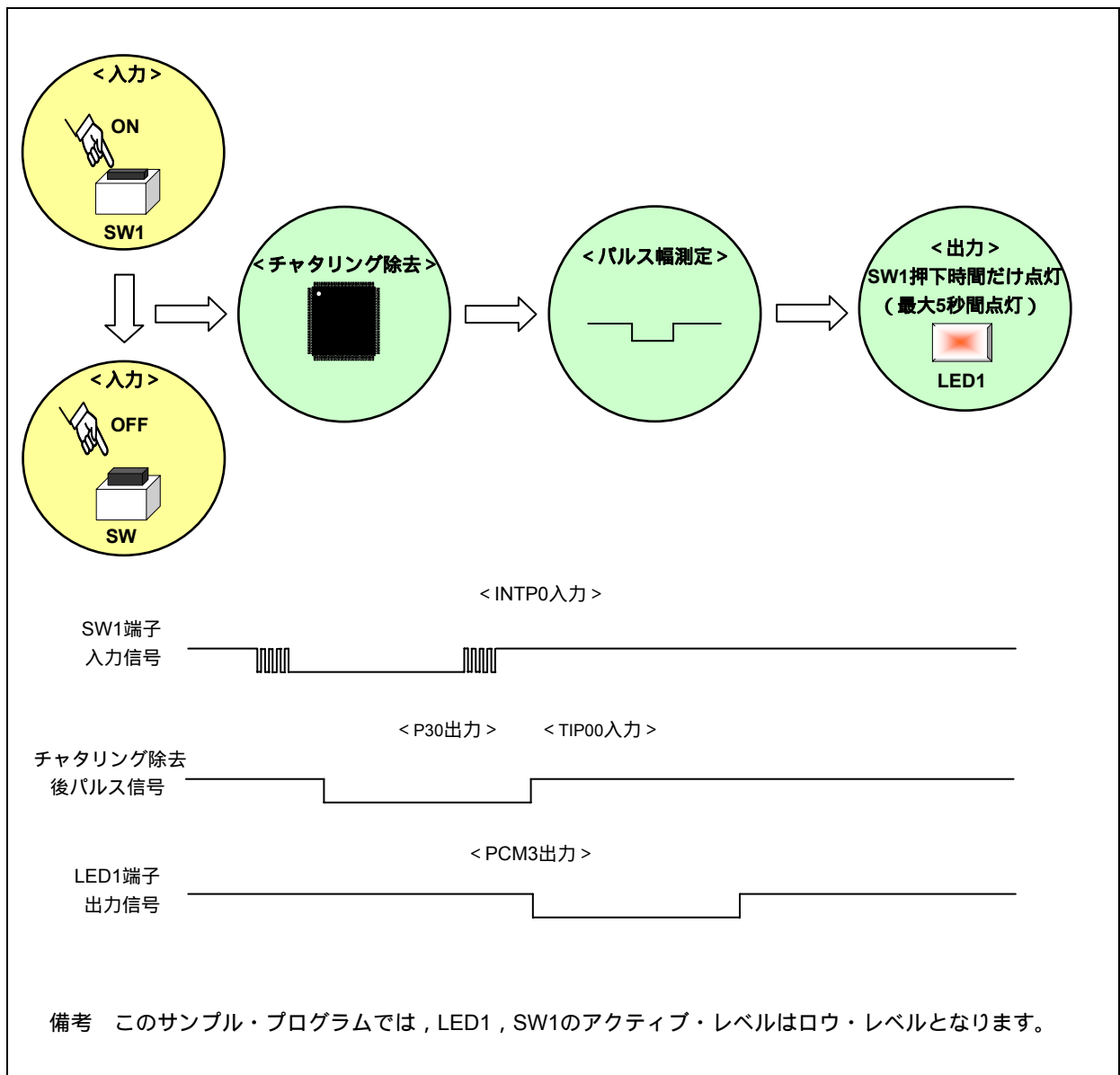
このサンプル・プログラムでは、16ビット・タイマ/イベント・カウンタ (TMP) のパルス幅測定機能の使用例を示しています。SW1を押下していた時間をパルス幅として測定し^{注1}、押下されていた時間分だけ、LED1を点灯させます。

パルス幅の測定では、設定したカウント・クロック (625 kHz) を16ビット・カウンタでカウントすることで、104.856 msまで測定できます。本サンプル・プログラムでは、16ビット・カウンタのオーバーフローする回数をソフトウェアにてカウントすることで、それ以上の時間の計測も可能にしています^{注2}。

リセット解除後に動作停止状態の周辺機能について、このサンプル・プログラムで使用しない周辺機能は、設定していません。

- 注1. SW1入力のチャタリングの影響をなくすため、ソフトウェアにてチャタリングを除去したパルスを生成し、そのパルス幅を測定します。
2. 押下時間は最大5秒で、それ以上押下しても5秒とみなします。またLED1を点灯している間はSW1の押下入力を受け付けません。

図1 - 1 動作概要



1.1 初期化

主な初期設定について次に説明します。

(1) オプション・バイトの参照

- ・リセット解除後の発振安定時間を参照

(2) 内蔵周辺設定

- ・内蔵周辺I/Oレジスタへのバス・アクセスのウエイト<ウエイト：1>設定
- ・オンチップ・デバッグ・モード・レジスタ（OCDM）を通常動作モードに設定
- ・内蔵発振器を停止，ウォッチドッグ・タイマ2停止
- ・CPUクロックを分周なしに設定
- ・PLLモードに設定し5 MHz×4通倍 = 20 MHz動作に設定

(3) 端子設定

- ・未使用端子の設定
- ・SW1入力による外部割り込み端子の設定（INTP0機能）
- ・LED1出力端子の設定（PCM3端子）
- ・SW1入力チャタリング除去後出力端子の設定（P30端子）
- ・パルス幅測定用入力端子の設定（TIP00機能）

(4) 外部割り込み（INTP0）の設定

- ・INTP0入力信号の有効エッジを両エッジに設定
- ・INTP0の割り込みの優先度7，割り込みマスク解除

(5) タイマP0（TMP0）の設定

- ・TP0CTL0のカウント・クロック選択を $f_{xx}/32$ （625 kHz）に設定
- ・TP0CTL1のタイマ・モード設定をパルス幅測定モードに設定
- ・TP0IOC1：キャプチャ・トリガ入力信号の有効エッジを両エッジに設定
- ・INTTP0CC0割り込みの優先度7，割り込みマスク解除
- ・TP0OVMK：16ビット・カウンタのオーバフロー割り込みをマスク
- ・TP0CE：TMP0動作許可

(6) 変数の設定

- ・オーバフロー・カウンタ初期化

1.2 SW1入力による外部割り込み処理 (INTP0)

SW1入力によるINTP0端子の立ち上がり、立ち下がりエッジの両エッジを検出し、割り込み処理を行います。

SW1による外部割り込み処理では、INTP0端子のエッジを検出してから約10 ms経過後にSW1の押下または解放の確認をし、P30端子の出力レベルを変化させます。

なお、INTP0端子のエッジ検出後、約10 ms経過後にSW1の入力レベルが変化していた場合は、チャタリング・ノイズであると判定し、P30端子の出力レベルを変化させません。

備考 デバイス使用上の注意事項については、各製品のユーザーズ・マニュアル (V850ES/Jx3-L) を参照してください。

1.3 16ビット・タイマ/イベント・カウンタ (TMP0) を使用したキャプチャ割り込み処理

P30端子出力によるTIP00端子入力信号の立ち上がり、立ち下がりエッジの両エッジを検出し、割り込み処理を行います。

16ビット・タイマ/イベント・カウンタ (TMP0) を使用したキャプチャ割り込み処理では、立ち下がりエッジ検出から、立ち上がりエッジ検出までのパルスのロウ・レベル幅を測定することにより、SW1の押下から解放までの時間を算出します。算出したSW1の押下受付時間分だけ (最大5秒間) LED1を点灯します。

また、LED1の点灯処理中に発生したINTP0端子のエッジ検出による外部割り込みを無効化します。

備考 デバイス使用上の注意事項については、各製品のユーザーズ・マニュアル (V850ES/Jx3-L) を参照してください。

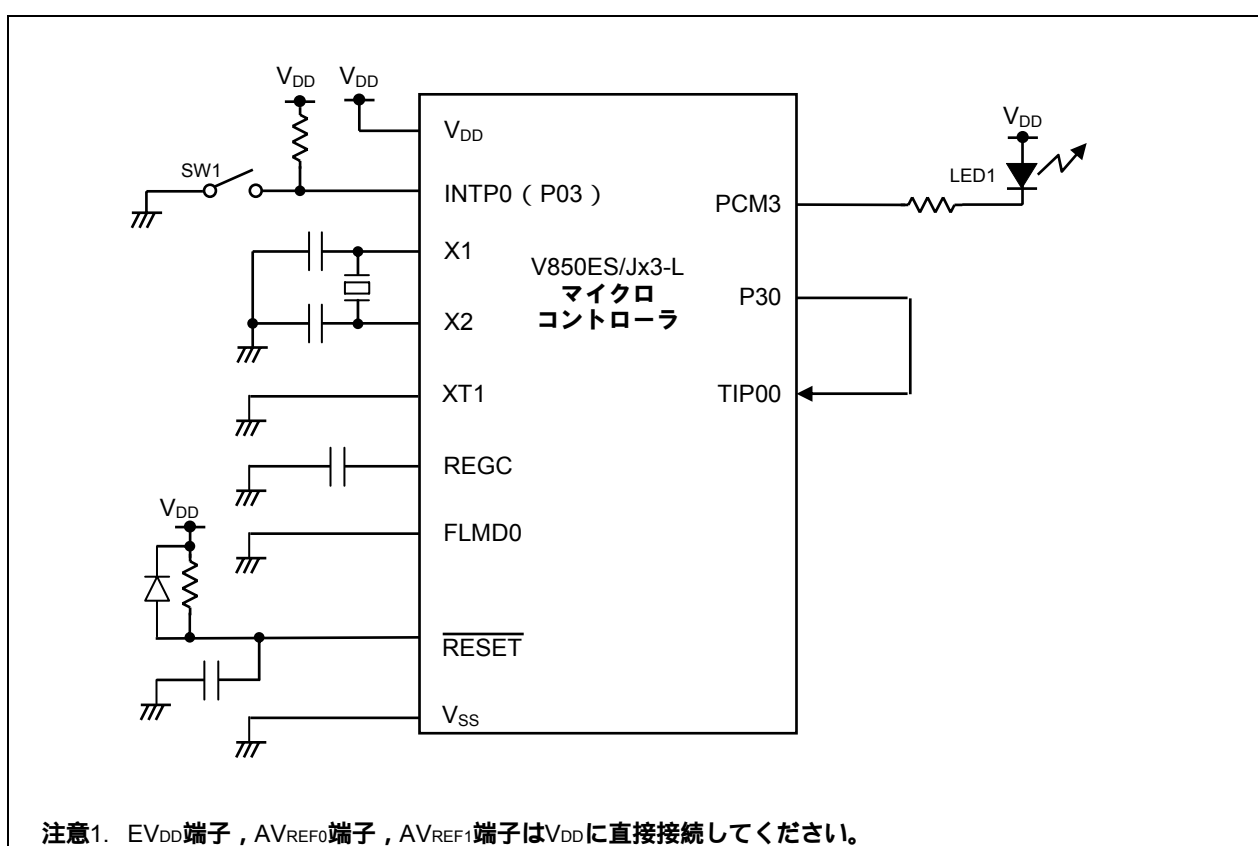
第2章 回路図

この章では、このサンプル・プログラムで使用する場合の回路図および周辺ハードウェアを説明します。

2.1 回路図

回路図を次に示します。

図2 - 1 回路図



- 注意1. EV_{DD}端子, AV_{REF0}端子, AV_{REF1}端子はV_{DD}に直接接続してください。
- EV_{SS}端子, AV_{SS}端子はGNDに直接接続してください。
 - FLMD0端子は、通常動作モード時はGNDに接続してください。
 - REGCはコンデンサ（推奨値：4.7 μ F）を介し、GNDに接続してください。
 - 未使用ポートについては、出力ポートとして処理するため、すべてオープンとしてください。

2.2 周辺ハードウェア

使用する周辺ハードウェアを次に示します。

(1) スイッチ (SW1)

パルス出力制御用の割り込み入力として、スイッチを使用します。

(2) LED (LED1)


スイッチ入力時間に対応した出力として、LEDを使用します。

第3章 ソフトウェアについて

この章では、ダウンロードする圧縮ファイルのファイル構成、使用するマイコンの内蔵周辺機能、サンプル・プログラムの初期設定と動作概要、およびフロー・チャートを説明します。

3.1 ファイル構成

ダウンロードする圧縮ファイルのファイル構成は、次のようになっています。

ファイル名 (ツリー構造)	説明	同封圧縮 (*.zip) ファイル	
			
<ul style="list-style-type: none"> ・ conf <ul style="list-style-type: none"> — crtE.s — AppNote_Pulse.dir — AppNote_Pulse.prj — AppNote_Pulse.prw 	スタート・アップ・ルーチン・ファイル ^{注1}	-	
	リンク・ディレクティブ・ファイル ^{注2}		
	統合開発環境PM+用プロジェクト・ファイル	-	
	統合開発環境PM+用ワーク・スペース・ファイル	-	
<ul style="list-style-type: none"> ・ src <ul style="list-style-type: none"> — main.c — minicube2.s — opt_b.s 	マイコンのハードウェア初期化処理とメイン処理を記述したC言語ソース・ファイル		
	MINICUBE2用の領域予約を行う ソース・ファイル		
	オプション・バイト設定を行う ソース・ファイル		

注1. ワーク・スペース新規作成時の「スタート・アップ・ファイルの指定」時に、「サンプルをコピーして使用する(C)」を選択した際にコピーされるスタート・アップ・ファイル。


(デフォルト・インストール・パスであれば、


C:\Program Files\NEC Electronics Tools\CA850**使用バージョン**\lib850\r32\crtE.sのコピーとなります。)

2. ワーク・スペース新規作成時の「リンク・ディレクティブ・ファイルの指定」時に、「サンプルを作成して使用する(C)」を選択し、「メモリの使用方法：内蔵メモリのみ(C)」をチェックした際に、自動生成されるリンク・ディレクティブ・ファイルに、**MINICUBE2用のセグメントを追加**したもの。

(デフォルト・インストール・パスであれば、

C:\Program Files\NEC Electronics Tools\PM+**使用バージョン**\bin\w_data\850_i.datが基準となります。)

備考  : ソース・ファイルのみ同封

 : 統合開発環境 PM+で使用するファイルを同封

3.2 使用する内蔵周辺機能

このサンプル・プログラムでは、マイコンに内蔵する次の周辺機能を使用します。

- ・パルス幅測定機能 : 16ビット・タイマ/イベント・カウンタ (TMP0)
- ・外部割り込み入力 (スイッチ入力用) : INTP0 (SW1)
- ・キャプチャ・トリガ入力ポート : TIP00^注
- ・出力ポート (LED1) : PCM3
- ・出力ポート (TIP00端子へのパルス出力用) : P30

注 V850ES/JG3-Lマイコンでは、この他にASCKA0/ $\overline{\text{SCKB4}}$ /TOP00/P32と兼用端子になっています。
V850ES/JF3-Lマイコンでは、この他にASCKA0/TOP00/P32と兼用端子になっています。

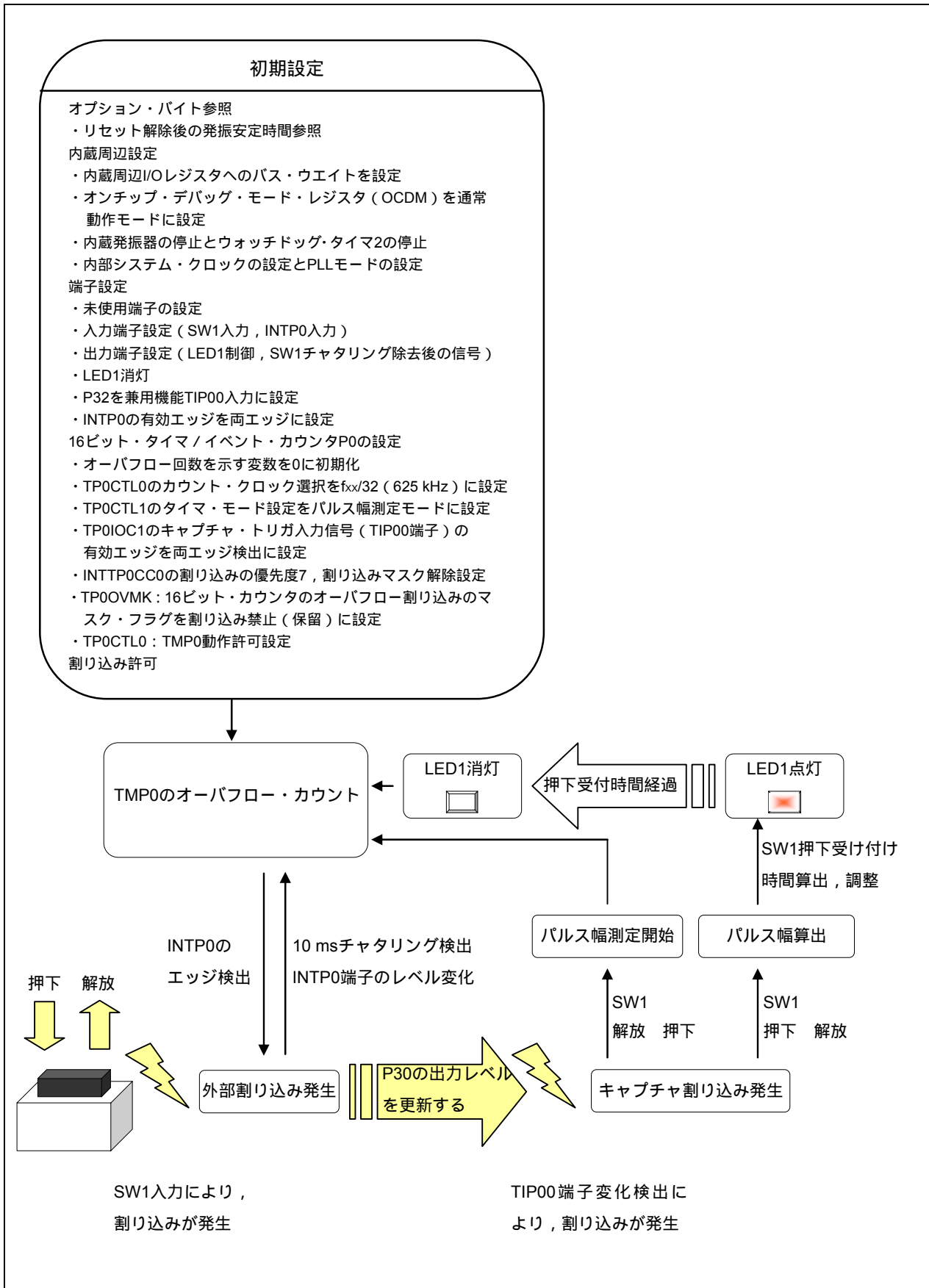
3.3 初期設定と動作概要

このサンプル・プログラムでは、初期設定にて、クロック周波数の選択や、ウォッチドッグ・タイマ2の停止設定、入出力ポート設定、16ビット・タイマ/イベント・カウンタ (TMP0) のパルス幅測定モード設定、割り込みの設定などを行います。

初期設定完了後は、外部割り込み入力によりSW1の押下状態を確認し、P30端子への出力値を切り替えます。また、TIP00の16ビット・タイマ/イベント・カウンタ (TMP0) のパルス幅測定モードを利用して、P30端子からのTIP00端子への入力パルス幅を測定し、SW1の押下から解放までの時間を算出したあと、LED1を点灯させます。LED1の点灯時間はSW1の押下時間に合わせて、最大5秒間点灯し、LED1が点灯中はSW1の押下は受け付けません。

初期設定と動作概要の詳細について、次に示します。

図3 - 1 状態遷移図 (ステート・チャート)



3.4 フロー・チャート

このサンプル・プログラムのフロー・チャートを次に示します。

図3-2 フロー・チャート (1/3)

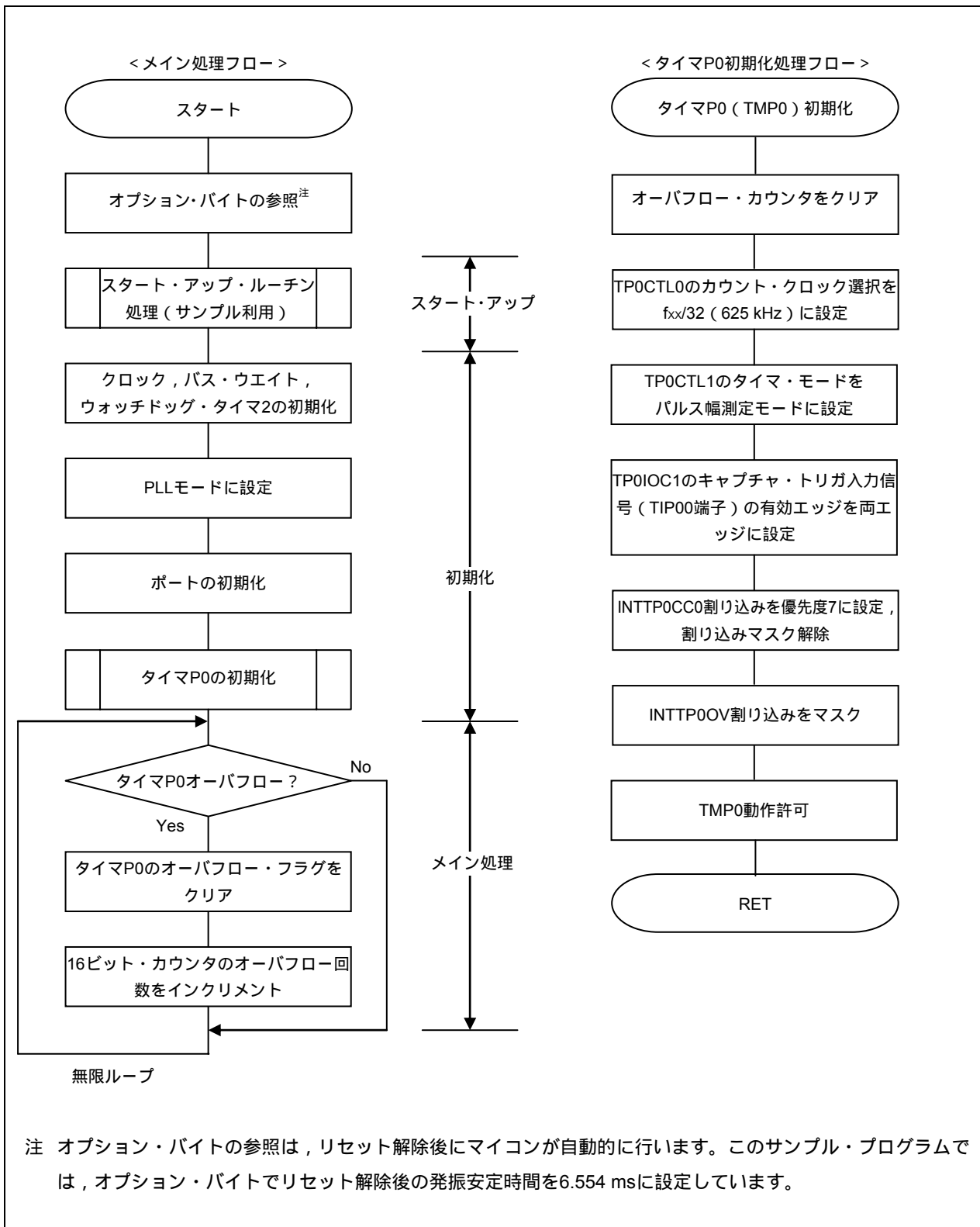


図3-2 フロー・チャート(2/3)

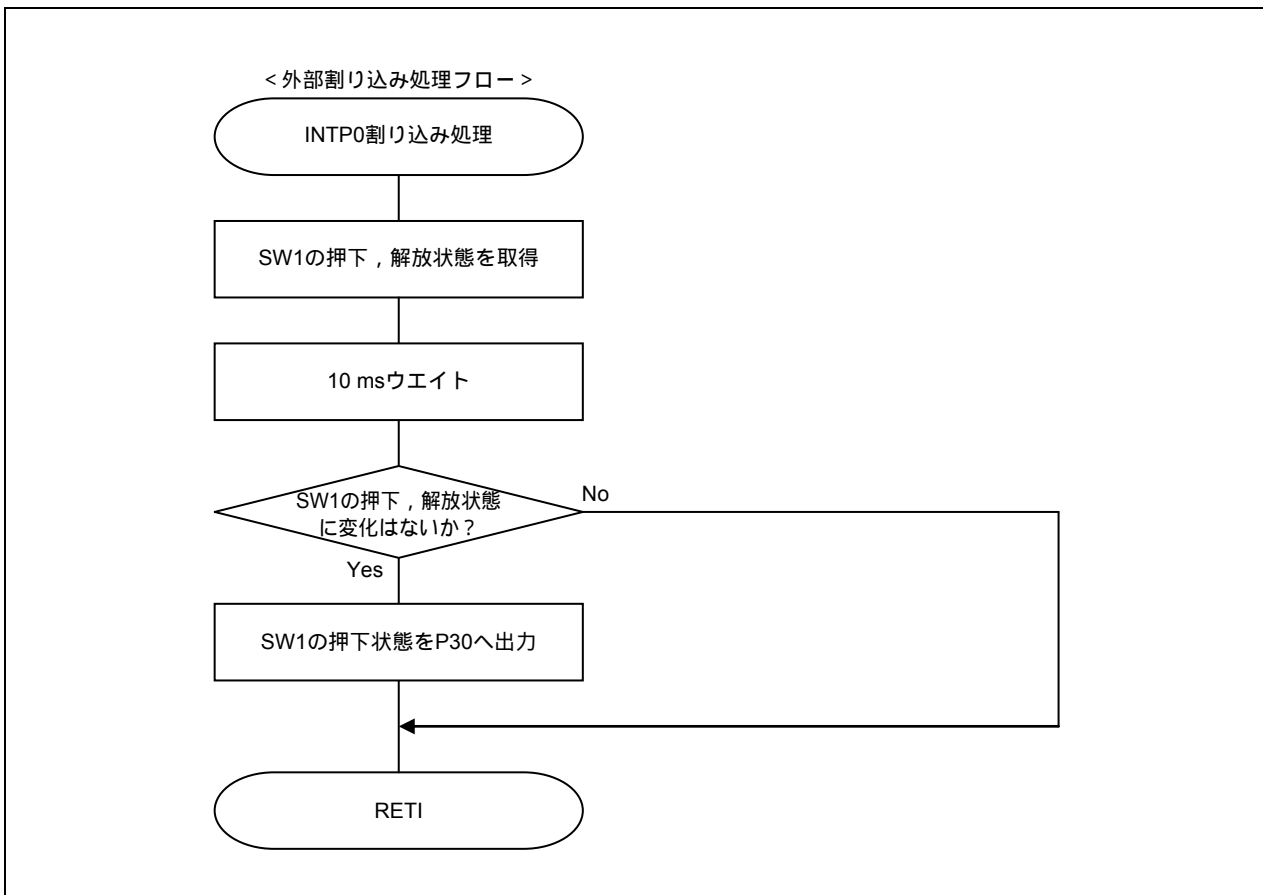
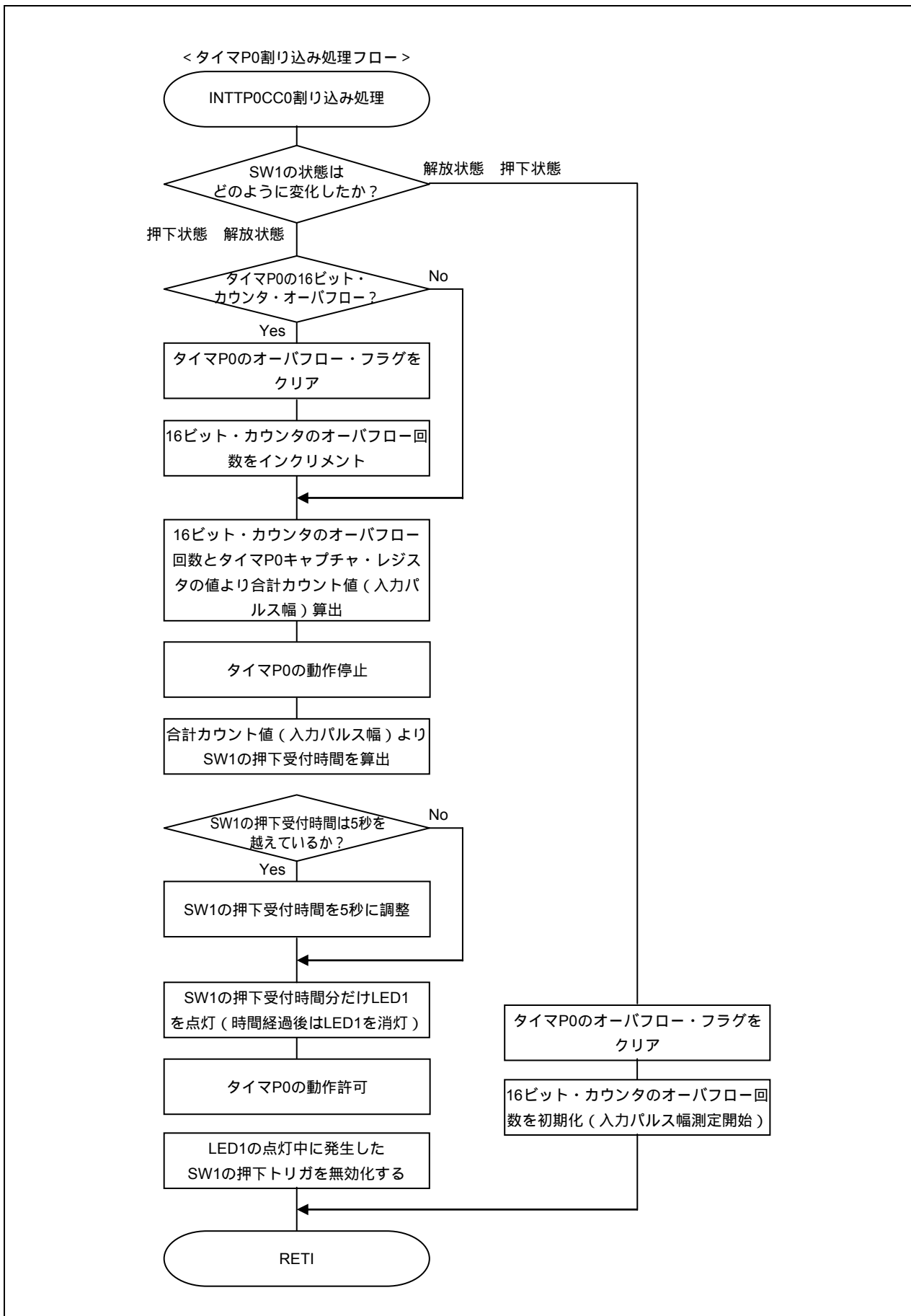


図3-2 フロー・チャート(3/3)



【コラム】 スタート・アップ・ルーチン処理の内容

スタート・アップ・ルーチンは、V850のリセットが解除されたあと、メイン関数を実行する前に、実行されるルーチンです。基本的にはC言語で記述されたプログラムが動作するための初期化処理を行います。

具体的には、次のことを行います。

- ・ main関数の引き数領域の確保
- ・ スタック領域の確保
- ・ リセットが入った時のRESETハンドラの設定
- ・ テキスト・ポインタ (tp) の設定
- ・ グローバル・ポインタ (gp) の設定
- ・ スタック・ポインタ (sp) の設定
- ・ エlement・ポインタ (ep) の設定
- ・ マスク・レジスタ (r20, r21) へマスク値を設定
- ・ sbss領域, bss領域のゼロ・クリア
- ・ 関数のプロローグ・エピローグ・ランタイム
- ・ ライブラリ用のCTBP値の設定
- ・ r6とr7をmain関数の引き数に設定
- ・ main関数へ分岐する

3.5 V850ES/JG3-LとV850ES/JF3-Lの違い

V850ES/JG3-Lは、V850ES/JF3-Lに対して、I/O、タイマ/カウンタ、シリアル・インターフェースなどの機能を拡張したものです。

このサンプル・プログラムにおいては、I/Oの初期化におけるP1, P3, P7, P9, PDHの初期化範囲が異なります。

サンプル・プログラムの詳細については、**付録A プログラム・リスト**を参照してください。

3.6 TMPとTMQの違い

16ビット・タイマ/イベント・カウンタP (TMP) と、16ビット・タイマ/イベント・カウンタQ (TMQ) は、キャプチャ・トリガ端子やタイマ出力端子、キャプチャ/コンペア・レジスタの本数などが異なります。

このサンプル・プログラムにおいては、16ビット・タイマ/イベント・カウンタP (TMP) を使用しています。

16ビット・タイマ/イベント・カウンタQ (TMQ) を使用する場合は、**第4章 レジスタ設定について**、**付録A プログラム・リスト**の内容を参考に設定してください。

3.7 セキュリティIDについて

オンチップ・デバッグ・エミュレータによるオンチップ・デバッグ時、フラッシュ・メモリの内容を第三者に読み出される事を防ぐために、10バイトのIDコードによる認証を行います。

セキュリティIDについての詳細説明は、「V850ES/Jx3-L サンプル・プログラム (割り込み) スイッチ入力による外部割り込み編」を参照ください。

第4章 レジスタ設定について

この章では、16ビット・タイマ/イベント・カウンタ(TMP, TMQ)機能、および割り込み端子設定について説明します。

なお、リセット解除後に動作停止状態の周辺機能について、このサンプル・プログラムで使用しない周辺機能は、設定していません。レジスタ設定方法の詳細については、各製品のユーザーズ・マニュアルを参照してください。

- ・ V850ES/JG3-L 32ビット・シングルチップ・マイクロコントローラ ハードウェア編
- ・ V850ES/JF3-L 32ビット・シングルチップ・マイクロコントローラ ハードウェア編

C言語の拡張記述の詳細については、次のユーザーズ・マニュアルを参照してください。

- ・ CA850 Cコンパイラ・パッケージ C言語編 ユーザーズ・マニュアル

4.1 16ビット・タイマ/イベント・カウンタ (TMP) の設定

16ビット・タイマ/イベント・カウンタ (TMP) を使用する際に関係するレジスタには、次の9種類があります。

- ・TMPn制御レジスタ0 (TPnCTL0)
- ・TMPn制御レジスタ1 (TPnCTL1)
- ・TMPnI/O制御レジスタ0 (TPnIOC0)
- ・TMPnI/O制御レジスタ1 (TPnIOC1)
- ・TMPnI/O制御レジスタ2 (TPnIOC2)
- ・TMPnオプション・レジスタ0 (TPnOPT0)
- ・TMPnキャプチャ/コンペア・レジスタ0 (TPnCCR0)
- ・TMPnキャプチャ/コンペア・レジスタ1 (TPnCCR1)
- ・TMPnカウンタ・リード・バッファ・レジスタ (TPnCNT)

備考1. n = 0-5 (V850ES/JG3-L) / n = 0-2, 5 (V850ES/JF3-L)

2. このサンプル・プログラムでは、n = 0を使用しています。

3. 16ビット・タイマ/イベント・カウンタ (TMQ) を使用する際に関係するレジスタには、次の11種類があります。

次頁以降では、TMPの説明を行いますので、TMQを使用する場合は、以下の各レジスタと読み替えてください。また、各ビット名称の“TP”を“TQ”に読み替えて下さい。

- ・TMQ0制御レジスタ0 (TQ0CTL0)
- ・TMQ0制御レジスタ1 (TQ0CTL1)
- ・TMQ0I/O制御レジスタ0 (TQ0IOC0)
- ・TMQ0I/O制御レジスタ1 (TQ0IOC1)
- ・TMQ0I/O制御レジスタ2 (TQ0IOC2)
- ・TMQ0オプション・レジスタ0 (TQ0OPT0)
- ・TMQ0キャプチャ/コンペア・レジスタ0 (TQ0CCR0)
- ・TMQ0キャプチャ/コンペア・レジスタ1 (TQ0CCR1)
- ・TMQ0キャプチャ/コンペア・レジスタ2 (TQ0CCR2)
- ・TMQ0キャプチャ/コンペア・レジスタ3 (TQ0CCR3)
- ・TMQ0カウンタ・リード・バッファ・レジスタ (TQ0CNT)

4.1.1 16ビット・タイマ/イベント・カウンタ (TMP) の動作クロックの設定

TMPn制御レジスタ0 (TPnCTL0) で、16ビット・タイマ/イベント・カウンタ (TMP) のカウント・クロックの選択、および動作の制御を行います。

TPnCKS2-TPnCKS0ビットは、TPnCEビット = 0の状態を設定する必要があります。

このサンプル・プログラムでは、初期化時に00Hを設定することで、TPnCKS2-TPnCKS0ビットをfxx/32(625 kHz)に設定します。16ビット・タイマ/イベント・カウンタ (TMP) の各レジスタの設定が終了したあとに、TPnCEビット = 1に設定します。

図4 - 1 TPnCTL0のフォーマット

TMPn制御レジスタ0 (TPnCTL0)

アドレス: TP0CTL0 0xFFFFF590, TP1CTL0 0xFFFFF5A0

TP2CTL0 0xFFFFF5B0, TP3CTL0 0xFFFFF5C0

TP4CTL0 0xFFFFF5D0, TP5CTL0 0xFFFFF5E0

7	6	5	4	3	2	1	0
TPnCE	0	0	0	0	TPnCKS2	TPnCKS1	TPnCKS0

TPnCE	TMPnの動作の制御
0	TMPn動作禁止 (TMPnを非同期にリセット)
1	TMPn動作許可。TMPn動作開始

TPnCKS2	TPnCKS1	TPnCKS0	内部カウント・クロックの選択	
			n = 0, 2, 4	n = 1, 3, 5
0	0	0	fxx	
0	0	1	fxx/2	
0	1	0	fxx/4	
0	1	1	fxx/8	
1	0	0	fxx/16	
1	0	1	fxx/32	
1	1	0	fxx/64	fxx/256
1	1	1	fxx/128	fxx/512

備考 表の赤字部分がサンプル・プログラムでの設定値となります。

4.1.2 16ビット・タイマ/イベント・カウンタ (TMP) の動作モードの設定

TMPn制御レジスタ1 (TPnCTL1) で、16ビット・タイマ/イベント・カウンタ (TMP) の動作モードを設定します。

このサンプル・プログラムでは、TPnMD2-TPnMD0 = 110bに設定することにより、パルス幅測定モードとして設定しています。

図4-2 TPnCTL1のフォーマット

TMPn制御レジスタ1 (TPnCTL1)

アドレス: TP0CTL1 0xFFFFF591, TP1CTL1 0xFFFFF5A1

TP2CTL1 0xFFFFF5B1, TP3CTL1 0xFFFFF5C1

TP4CTL1 0xFFFFF5D1, TP5CTL1 0xFFFFF5E1

7	6	5	4	3	2	1	0
0	TPnEST	TPnEEE	0	0	TPnMD2	TPnMD1	TPnMD0

TPnEST	ソフトウェア・トリガ制御
0	-
1	外部トリガ入力の有効な信号を作成

TPnEEE	カウント・クロックの選択
0	外部イベント・カウント入力での動作禁止 ^注
1	外部イベント・カウント入力での動作許可

TPnMD2	TPnMD1	TPnMD0	タイマ・モードの選択
0	0	0	インターバル・タイマ・モード
0	0	1	外部イベント・カウント・モード
0	1	0	外部トリガ・パルス出力モード
0	1	1	ワンショット・パルス出力モード
1	0	0	PWM出力モード
1	0	1	フリー・ランニング・タイマ・モード
1	1	0	パルス幅測定モード
1	1	1	設定禁止

注 TPnCTL0.TPnCK0-TPnCK2ビットによって選択されたクロック

備考 表の赤字部分がサンプル・プログラムでの設定値となります。

4.1.3 タイマ出力の制御

TMPnI/O制御レジスタ0 (TPnIOC0) で、タイマ出力の制御を行います。

パルス幅測定モードとして動作する場合、TMPnI/O制御レジスタ0の制御は不要のため、このサンプル・プログラムでは、本レジスタの制御を行っていません。

図4 - 3 TPnIOC0のフォーマット

TMPnI/O制御レジスタ0 (TPnIOC0)

アドレス : TP0IOC0 0xFFFFF592, TP1IOC0 0xFFFFF5A2

TP2IOC0 0xFFFFF5B2, TP3IOC0 0xFFFFF5C2

TP4IOC0 0xFFFFF5D2, TP5IOC0 0xFFFFF5E2

7	6	5	4	3	2	1	0
0	0	0	0	TPnOL1	TPnOE1	TPnOL0	TPnOE0

TPnOL1	TOPn1端子出力レベルの設定
0	TOPn1端子ハイ・レベル・スタート
1	TOPn1端子ロウ・レベル・スタート

TPnOE1	TOPn1端子出力の設定
0	タイマ出力禁止
1	タイマ出力許可

TPnOL0	TOPn0端子出力レベルの設定
0	TOPn0端子ハイ・レベル・スタート
1	TOPn0端子ロウ・レベル・スタート

TPnOE0	TOPn0端子出力の設定
0	タイマ出力禁止
1	タイマ出力許可

備考1. 表の赤字部分がサンプル・プログラムでの設定値となります。

- 16ビット・タイマ/イベント・カウンタ (TMQ) では、“TP” を “TQ” に置き換える以外に、TQ0IOC0レジスタのビット7-4に、TQ0OL3, TQ0OE3, TQ0OL2, TQ0OE2ビットがそれぞれ割り当てられています。

4.1.4 キャプチャ・トリガ入力信号の有効エッジの制御

TMPnI/O制御レジスタ1 (TPnIOC1) で、キャプチャ・トリガ入力信号 (TIPn0, TIPn1端子) に対する有効エッジを制御します。

このサンプル・プログラムでは、TPnIS1-TPnIS0 = 11bに設定することにより、キャプチャ・トリガ入力信号 (TIPn0端子) を両エッジ検出として設定しています。

図4 - 4 TPnIOC1のフォーマット

TMPnI/O制御レジスタ1 (TPnIOC1)

アドレス : TP0IOC1 0xFFFFF593 , TP1IOC1 0xFFFFF5A3

TP2IOC1 0xFFFFF5B3 , TP3IOC1 0xFFFFF5C3

TP4IOC1 0xFFFFF5D3 , TP5IOC1 0xFFFFF5E3

7	6	5	4	3	2	1	0
0	0	0	0	TPnIS3	TPnIS2	TPnIS1	TPnIS0

TPnIS3	TPnIS2	キャプチャ・トリガ入力信号 (TIPn1端子) の有効エッジの設定
0	0	エッジ検出なし
0	1	立ち上がりエッジを検出
1	0	立ち下がりエッジを検出
1	1	両エッジを検出

TPnIS1	TPnIS0	キャプチャ・トリガ入力信号 (TIPn0端子) の有効エッジの設定
0	0	エッジ検出なし
0	1	立ち上がりエッジを検出
1	0	立ち下がりエッジを検出
1	1	両エッジを検出

備考1. このサンプル・プログラムでは使用していません。

2. 16ビット・タイマ/イベント・カウンタ (TMQ) では、“TP” を“TQ” に置き換える以外に、TQ0IOC1レジスタのビット7-4にTQ0IS7、TQ0IS6、TQ0IS5、TQ0IS4ビットがそれぞれ割り当てられています。

4.1.5 外部入力信号の制御

TMPnI/O制御レジスタ2 (TPnIOC2) で、外部イベント・カウント入力信号 (TIPn0端子)、外部トリガ入力信号 (TIPn0端子) を制御します。

パルス幅測定モードとして動作する場合、TMPnI/O制御レジスタ2の制御は不要のため、このサンプル・プログラムでは、本レジスタの制御を行っていません。

図4 - 5 TPnIOC2のフォーマット

TMPnI/O制御レジスタ2 (TPnIOC2)

アドレス : TP0IOC2 0xFFFFF594, TP1IOC2 0xFFFFF5A4

TP2IOC2 0xFFFFF5B4, TP3IOC2 0xFFFFF5C4

TP4IOC2 0xFFFFF5D4, TP5IOC2 0xFFFFF5E4

7	6	5	4	3	2	1	0
0	0	0	0	TPnEES1	TPnEES0	TPnETS1	TPnETS0

TPnEES1	TPnEES0	外部イベント・カウント入力信号 (TIPn0端子) の有効エッジの指定
0	0	エッジ検出なし
0	1	立ち上がりエッジを検出
1	0	立ち下がりエッジを検出
1	1	両エッジを検出

TPnETS1	TPnETS0	外部トリガ入力信号 (TIPn0端子) の有効エッジの指定
0	0	エッジ検出なし
0	1	立ち上がりエッジを検出
1	0	立ち下がりエッジを検出
1	1	両エッジを検出

備考 このサンプル・プログラムでは使用していません。

4.1.6 キャプチャ/コンペア動作の制御

TMPnオプション・レジスタ0 (TPnOPT0) で、キャプチャ/コンペア動作の設定、オーバーフローの検出を制御します。

このサンプル・プログラムでは、SW1が押下された時間を測定するためにTPnOVFフラグがセットされる回数をカウントします。

図4 - 6 TPnOPT0のフォーマット

TMPnオプション・レジスタ0 (TPnOPT0)

アドレス: TP0OPT0 0xFFFFF595, TP1OPT0 0xFFFFF5A5

TP2OPT0 0xFFFFF5B5, TP3OPT0 0xFFFFF5C5

TP4OPT0 0xFFFFF5D5, TP5OPT0 0xFFFFF5E5

7	6	5	4	3	2	1	0
0	0	TPnCCS1	TPnCCS0	0	0	0	TPnOVF

TPnCCS1	TPnCCR1レジスタのキャプチャ/コンペア選択
0	コンペア・レジスタに選択
1	キャプチャ・レジスタに選択

TPnCCS0	TPnCCR0レジスタのキャプチャ/コンペア選択
0	コンペア・レジスタに選択
1	キャプチャ・レジスタに選択

TPnOVF	TMPnのオーバーフロー検出フラグ
セット(1)	オーバーフロー発生
リセット(0)	TPnOVFビットへの0ライトまたはTPnCTL0.TPnCEビット = 0

備考1. 表の赤字部分がサンプル・プログラムで参照する値となり、セットされていた場合にはソフトウェアにてリセットします。

- 16ビット・タイマ/イベント・カウンタ (TMQ) では、“TP” を“TQ” に置き換える以外に、TQ0OPT0レジスタのビット7-4にTQ0CCS3, TQ0CCS2, TQ0CCS1, TQ0CCS0ビットがそれぞれ割り当てられています。

4.1.7 キャプチャ時のカウント値の参照

TMPnキャプチャ/コンペア・レジスタ0 (TPnCCR0) および, TMPnキャプチャ/コンペア・レジスタ1 (TPnCCR1) は, フリー・ランニング・タイマ・モードの場合のみ, キャプチャ機能とコンペア機能を切り替えて機能の設定をします。

パルス幅測定モードとして動作する場合, キャプチャ・トリガ入力 (TIPn0端子および, TIPn1端子) の有効エッジを検出すると, 16ビット・カウンタのカウント値をTMPnキャプチャ/コンペア・レジスタ0 (TPnCCR0) および, TMPnキャプチャ/コンペア・レジスタ1 (TPnCCR1) に格納し16ビット・カウンタをクリアします。

備考 サンプル・プログラムではパルス幅測定の入力端子としてTIP00を使用しているため, カウント値はTP0CCR0レジスタに反映されます。

図4 - 7 TPnCCR0のフォーマット

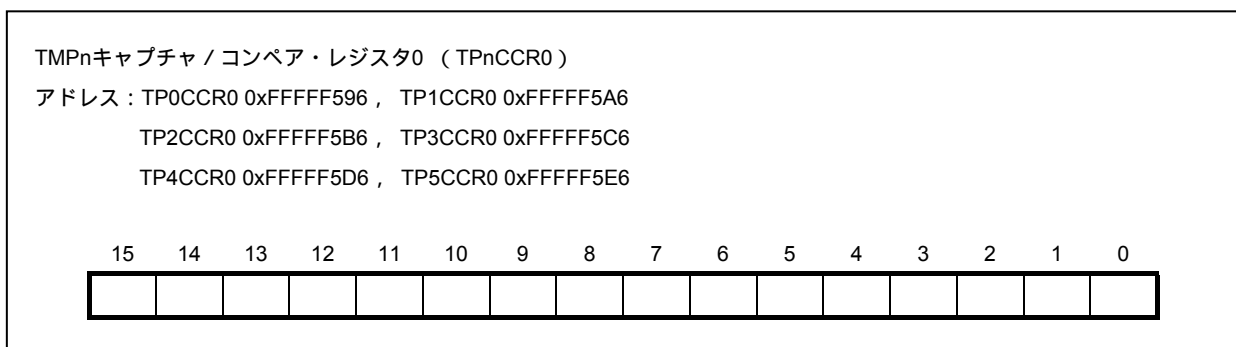
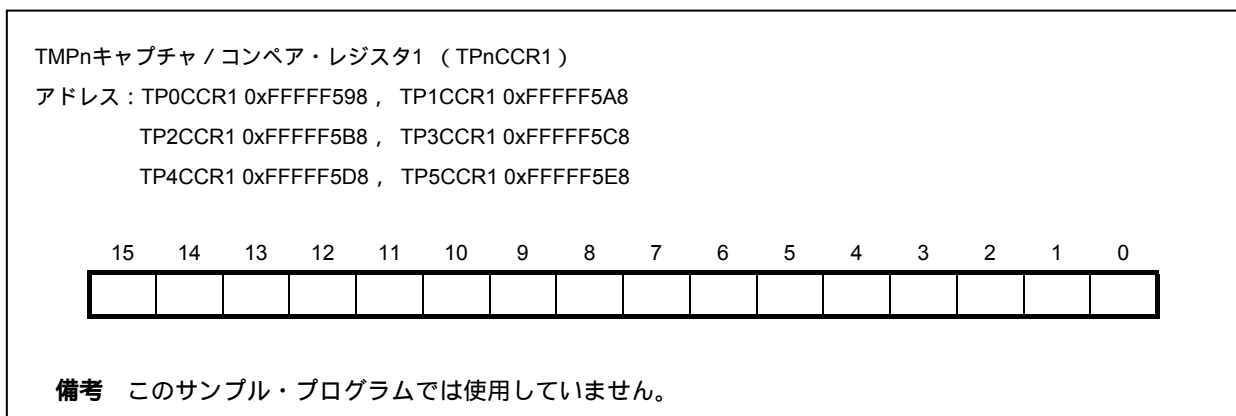


図4 - 8 TPnCCR1のフォーマット



16ビット・タイマ/イベント・カウンタ (TMQ) では, TQ0CCR0-3まで4つのキャプチャ/コンペア・レジスタが存在します。使用方法については, TPnCCR0, TPnCCR1レジスタと同様です。

4.1.8 タイマ・カウント値の参照

TPnCNTレジスタは、16ビットのカウント値をリードできるリード・バッファ・レジスタです。

タイマ動作中 (TPnCTL0.TPnCEビット = 1) に、TPnCNTレジスタの値をリードすることにより、現在のカウント値がリード可能です。

タイマ停止中 (TPnCTL0.TPnCEビット = 0) に、TPnCNTレジスタの値をリードすると0000Hとなります。このサンプル・プログラムでは、カウント値を参照する必要がないため使用していません。

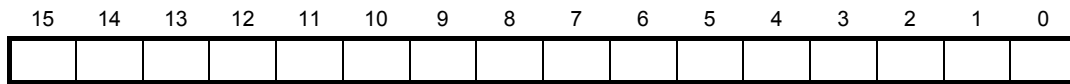
図4-9 TPnCNTのフォーマット

TMPnカウンタ・リード・バッファ・レジスタ (TPnCNT)

アドレス: TP0CNT 0xFFFFF59A, TP1CNT 0xFFFFF5AA

TP2CNT 0xFFFFF5BA, TP3CNT 0xFFFFF5CA

TP4CNT 0xFFFFF5DA, TP5CNT 0xFFFFF5EA



備考 このサンプル・プログラムでは使用していません。

4.1.9 タイマの設定例

【例 1】16ビット・タイマ/イベント・カウンタ (TMP) をパルス幅測定モードとして使用し、カウント・クロック、キャプチャ・トリガ入力信号の検出エッジを設定し、タイマ動作を開始する場合 (サンプル・プログラムと同内容)

・設定手順

カウント・クロックを $f_{xx}/32$ (625 kHz) に設定します。

パルス幅測定モードに設定します。

キャプチャ・トリガ入力信号を両エッジに設定します。

割り込みのマスクを解除します。

タイマの動作を開始します。

・プログラム例 (サンプル・プログラムと同内容)

```

/* タイマP0機能設定 */
TP0CTL0 = 0x05;          /* カウント・クロックを $f_{xx}/32$ に設定          */ }
                        /* FailSafe: TP0CE=0にしてTMP0停止          */ }
TP0CTL1 = 0x06;          /* パルス幅測定モード指定                    */ }
TP0IOC1 = 0x03;          /* キャプチャ・トリガ入力信号両エッジを検出 */ }

/* 割り込み制御レジスタ設定 */
TP0CCIC0 = 0x07;         /* INTTP0CC0 優先度7, マスク解除            */ }

TP0CE = 1;              /* TMP0動作開始                              */ }

```

【例 2】16ビット・タイマ/イベント・カウンタ (TMQ) をパルス幅測定モードとして使用し、カウント・クロック、キャプチャ・トリガ入力信号の検出エッジを設定し、タイマ動作を開始する場合

・設定手順

カウント・クロックを $f_{xx}/32$ (625 kHz) に設定します。

パルス幅測定モードに設定します。

キャプチャ・トリガ入力信号を両エッジに設定します。

割り込みのマスクを解除します。

タイマの動作を開始します。

・プログラム例

```

/* タイマQ0機能設定 */
TQ0CTL0 = 0x05;          /* カウント・クロックをfxx/32に設定          */ }
                        /* FailSafe : TQ0CE=0にしてTMP0停止          */ }
TQ0CTL1 = 0x06;          /* パルス幅測定モード指定                    */ }
TQ0IOC1 = 0x03;          /* キャプチャ・トリガ入力信号両エッジを検出 */ }

/* 割り込み制御レジスタ設定 */
TQ0CCIC0 = 0x07;         /* INTTQ0CC0 優先度7, マスク解除            */ }

TQ0CE = 1;              /* TMQ0動作開始                              */ }

```

4.2 SW1入力用割り込み端子設定

このサンプル・プログラムでは、P03端子をSW1入力の外部割り込み端子として使用し、割り込みは立ち上がり、立ち下がりの両エッジを検出するように設定します。

4.2.1 ポート0モード・レジスタ (PM0) の設定

PM0レジスタは、P02-P06端子の入出力モードの制御ができます。

このサンプル・プログラムでは、チャタリングの確認で端子状態を読み込みますので、P03端子の入出力モードを「入力モード」に設定します。

8/1ビット単位でリード/ライト可能です。

リセットにより、FFHになります。

図4 - 10 PM0レジスタのフォーマット

ポート0モード・レジスタ (PM0)							
アドレス : 0xFFFFF420							
7	6	5	4	3	2	1	0
1	PM06	PM05	PM04	PM03	PM02	1	1
PM03	P03端子の入出力モードの制御						
0	出力モード						
1	入力モード						

備考1. 表の赤字部分がサンプル・プログラムでの設定値となります。

2. PM03を出力モードにしてしまうと、P03の読み込みで出力ラッチの値を読み出してしまいうため、注意が必要です。

4.2.2 ポート0 モード・コントロール・レジスタ (PMC0) の設定

PMC0レジスタは、P02-P06端子の動作モードの指定ができます。

このサンプル・プログラムでは、P03端子の動作モードをINTP0（外部割り込み）入力に設定します。

8/1ビット単位でリード/ライト可能です。

リセットにより、00Hになります。

図4 - 11 PMC0レジスタのフォーマット

ポート0 モード・コントロール・レジスタ (PMC0)

アドレス : 0xFFFFF440

7	6	5	4	3	2	1	0
0	PMC06	PMC05 ^注	PMC04	PMC03	PMC02	0	0

PMC03	P03端子の動作モードの指定
0	入出力ポート
1	INTP0入力 / ADTRG入力

注 P05/INTP2/ $\overline{\text{DRST}}$ 端子は、OCDM.OCDM0ビット = 1のときは、PMC05ビットの値に関係なく $\overline{\text{DRST}}$ 端子となります。

備考1. 表の赤字部分がサンプル・プログラムでの設定値となります。

2. P02-P06端子は、兼用機能の入力時にはヒステリシス特性を持ちますが、ポート・モード時にはヒステリシス特性を持ちません。

4.2.3 外部割り込み立ち下がり，立ち上がりエッジ指定レジスタ0 (INTF0, INTR0) の設定

ビット2でNMI端子，ビット3-6で外部割り込み端子 (INTP0-INTP3) の立ち下がり，立ち上がりエッジ検出を指定する8ビットのレジスタです。

このサンプル・プログラムでは，両エッジを検出する設定にします。

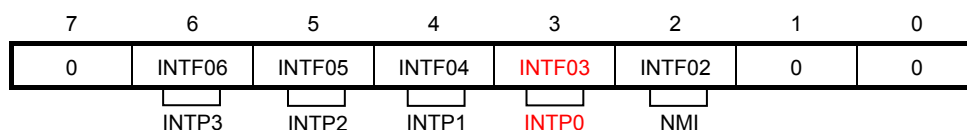
8/1ビット単位でリード/ライト可能です。

リセットにより，00Hになります。

図4 - 12 INTF0/INTR0レジスタのフォーマット

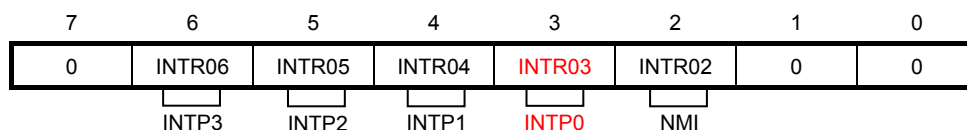
外部割り込み立ち下がりエッジ指定レジスタ0 (INTF0)

アドレス : 0xFFFFFC00



外部割り込み立ち上がりエッジ指定レジスタ0 (INTR0)

アドレス : 0xFFFFC20



備考 有効エッジの指定については以下の表を参照してください。

有効エッジの指定

INTF03	INTR03	有効エッジ指定
0	0	エッジ検出なし
0	1	立ち上がりエッジ検出
1	0	立ち下がりエッジ検出
1	1	両エッジ

備考 表の赤字部分がサンプル・プログラムでの設定値となります。

4.2.4 割り込み制御レジスタ (PIC0) の設定

割り込み要求信号 (マスカブル割り込み) ごとに割り当てられ, 各割り込みに対する制御条件を設定します。

このサンプル・プログラムでは, INTP0を最低優先順位で使用可能にします。

8/1ビット単位でリード/ライト可能です。

リセットにより, 47Hになります。

注意 PIC0.PIF0ビットを読み出す場合は, 割り込み禁止(DI)状態で行ってください。割り込み許可(EI)状態ではPIF0ビットを読み出すと, 割り込みの受け付けとビットの読み出しのタイミングが競合した場合に, 正常な値が読み出せないことがあります。

図4 - 13 PIC0レジスタのフォーマット

INTP0 割り込み制御・レジスタ (PIC0)
アドレス : 0xFFFFF112

7	6	5	4	3	2	1	0
PIF0	PMK0	0	0	0	PPR02	PPR01	PPR00

PIF0	割り込み要求フラグ
0	割り込み要求信号なし
1	割り込み要求信号あり

PMK0	割り込みマスク・フラグ
0	割り込み処理を許可
1	割り込み処理を禁止 (保留)

PPR02	PPR01	PPR00	割り込み優先順位指定ビット
0	0	0	レベル0 (最高位) を指定
1	0	1	レベル1 を指定
0	1	0	レベル2 を指定
0	1	1	レベル3 を指定
1	0	0	レベル4 を指定
1	0	1	レベル5 を指定
1	1	0	レベル6 を指定
1	1	1	レベル7 (最低位) を指定

備考 表の赤字部分がサンプル・プログラムでの設定値となります。



【コラム】 割り込み要求フラグ PIF0について

PIC0レジスタのPIF0「割り込み要求フラグ」は, 割り込み要因発生時に「1」にセットされ, 割り込み要求信号が受け付けられると, ハードウェアにより自動的にリセットされます。

4.3 パルス幅測定モード用キャプチャ割り込み端子設定

このサンプル・プログラムでは、P30端子をSW1入力確定用パルス出力端子として使用するよう設定し、P32端子をパルス幅測定モード用キャプチャ割り込み端子として使用するよう設定します。

4.3.1 ポート3モード・レジスタ (PM3) の設定

PM3レジスタは、P30-P39端子の入出力モードの制御ができます。

このサンプル・プログラムでは、P30端子をSW1入力確定用パルス出力端子として使用するため「出力モード」に設定します。

16/8/1ビット単位でリード/ライト可能です。

リセットにより、FFFFHになります。

図4 - 14 PM3レジスタのフォーマット

ポート3 モード・レジスタ (PM3)							
アドレス : 0xFFFFF426							
15	14	13	12	11	10	9	8
1	1	1	1	1	1	PM39	PM38
7	6	5	4	3	2	1	0
PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30
PM30	P30端子の入出力モードの制御						
0	出力モード						
1	入力モード						
備考 表の赤字部分がサンプル・プログラムでの設定値となります。							

4.3.2 ポート3モード・コントロール・レジスタ (PMC3) の設定

PMC3レジスタは、P30-P35端子および、P38-P39端子の動作モードの指定ができます。

このサンプル・プログラムでは、P32端子の動作モードをTIP00入力に設定します。

16/8/1ビット単位でリード/ライト可能です。

リセットにより、0000Hになります。

図4 - 15 PMC3レジスタのフォーマット

ポート3モード・コントロール・レジスタ (PMC3)
アドレス : 0xFFFFF446

15	14	13	12	11	10	9	8
0	0	0	0	0	0	PMC39	PMC38

7	6	5	4	3	2	1	0
0	0	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

PMC32	P32端子の動作モードの制御
0	入出力ポート
1	ASCKA0入力/SCKB4入出力/TIP00入力

備考 表の赤字部分がサンプル・プログラムでの設定値となります。

4.3.3 ポート3ファンクション・コントロール・レジスタ (PFC3) および、 ポート3ファンクション・コントロール拡張レジスタ (PFCE3L) の設定

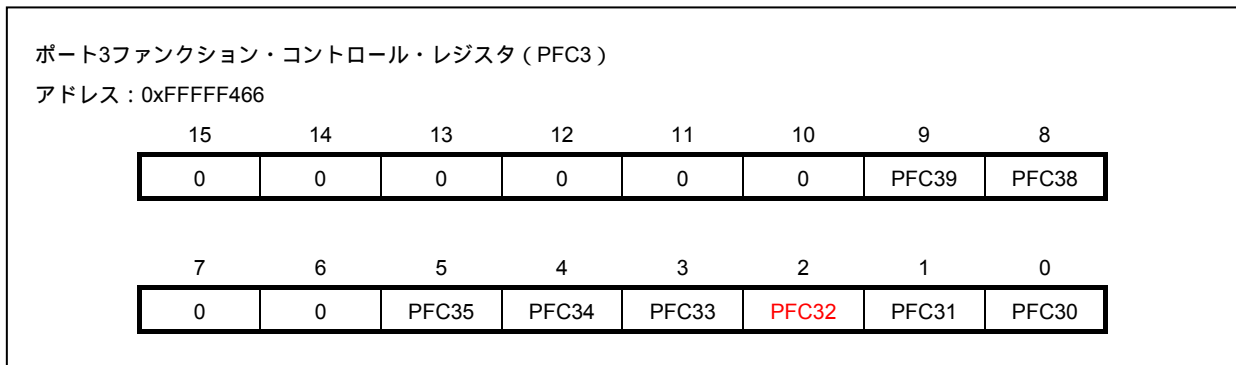
PFC3レジスタ、およびPFCE3Lレジスタは、P30-P35端子、およびP38-P39端子の兼用機能の指定ができます。

このサンプル・プログラムでは、P32端子をパルス幅測定モード用キャプチャ割り込み端子として使用するためTIP00入力に設定します。

16/8/1ビット単位でリード/ライト可能です。

リセットにより、0000Hになります。

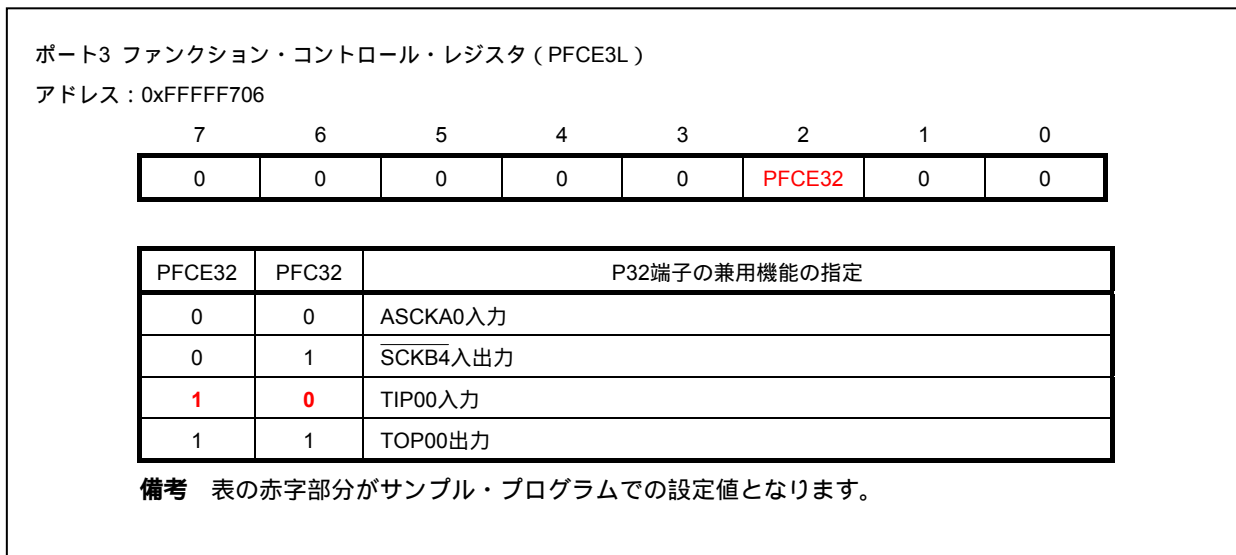
図4 - 16 PFC3レジスタのフォーマット



8/1ビット単位でリード/ライト可能です。

リセットにより、00Hになります。

図4 - 17 PFCE3Lレジスタのフォーマット



4.4 LED出力用端子設定

このサンプル・プログラムでは、LED1出力は初期状態で点灯させない出力に設定したあとに、PCM3端子を出力ポートとして動作させるように設定します。

4.4.1 ポートCMレジスタ (PCM) の設定

PCMレジスタは、PCM0-PCM3端子を1ビット単位で出力データの制御ができます。

このサンプル・プログラムでは、初期状態でPCM3端子をLED消灯としての出力をするため「1を出力」に設定します。

8/1ビット単位でリード/ライト可能です。

リセットにより、00Hになります。

図4 - 18 PCMレジスタのフォーマット

ポートCMレジスタ (PCM)							
アドレス : 0xFFFFF00C							
7	6	5	4	3	2	1	0
0	0	0	0	PCM3	PCM2	PCM1	PCM0
PCM3	PCM3端子の出力データの制御(出力モード時)						
0	0を出力						
1	1を出力						
備考 表の赤字部分がサンプル・プログラムでの設定値となります。							

4.4.2 ポートCMモード・レジスタ (PMCM) の設定

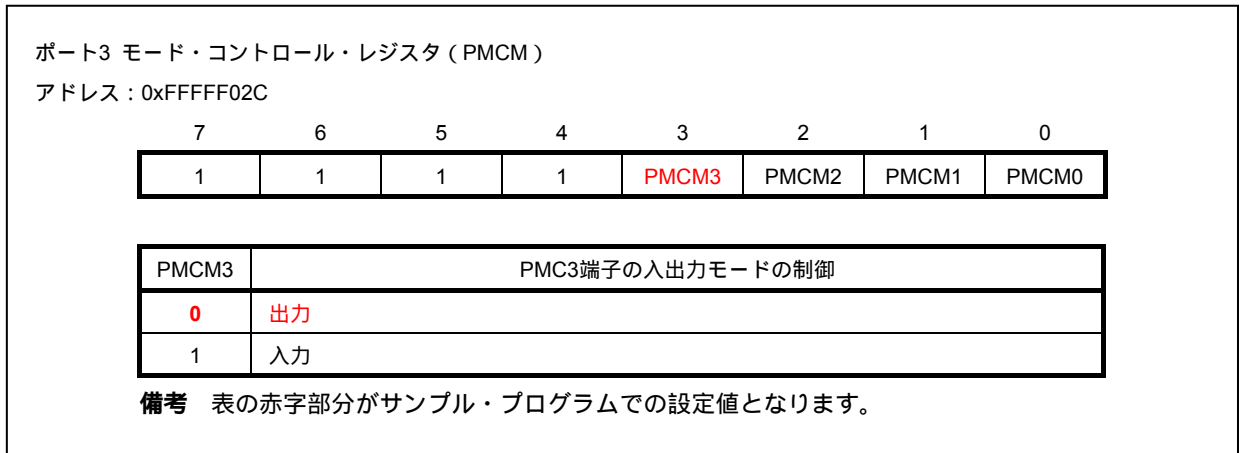
PMCMレジスタは、PCM0-PCM3端子の動作モードの指定ができます。

このサンプル・プログラムでは、PCM3端子の動作モードを出力に設定します。

8/1ビット単位でリード/ライト可能です。

リセットにより、FFHになります。

図4 - 19 PMCMレジスタのフォーマット



第5章 関連資料

資料名	資料番号
V850ES/JF3-L ユーザーズ・マニュアル ハードウェア編	U18952J
V850ES/JG3-L ユーザーズ・マニュアル ハードウェア編	U18953J
PM+ Ver.6.30 ユーザーズ・マニュアル	U18416J
CA850 Ver.3.20 Cコンパイラ・パッケージ 操作編	U18512J
CA850 Ver.3.20 Cコンパイラ・パッケージ C言語編	U18513J
CA850 Ver.3.20 Cコンパイラ・パッケージ アセンブリ言語編	U18514J
CA850 Ver.3.20 Cコンパイラ・パッケージ リンク・ディレクティブ編	U18515J
V850ES アーキテクチャ編	U15943J
QB-MINI2 プログラミング機能付きオンチップ・デバッグ・エミュレータ	U18371J
ID850QB Ver.3.40 統合デバッガ 操作編	U18604J

ドキュメント検索URL <http://www.necel.com/micro/ja/documentation.html>

付録A プログラム・リスト

プログラム・リスト例として、V850ES/JG3-Lマイクロコントローラのソース・プログラムを次に示します。

● opt_b.s

```
#-----  
#  
#     NEC Electronics     V850ES/Jx3-L マイクロコントローラ  
#  
#-----  
#     V850ES/JG3-L JF3-L サンプル・プログラム  
#-----  
#     パルス幅測定モード編  
#-----  
# 【履歴】  
#     2009.8.--   新規作成  
#-----  
# 【概要】  
#     本サンプル・プログラムは、オプション・バイトの設定を行う  
#-----  
  
    .section "OPTION_BYTES"  
    .byte 0b00000101 -- 0x7a (5MHz: 発振安定時間 6.554ms に設定)  
    .byte 0b00000000 -- 0x7b      ↑  
    .byte 0b00000000 -- 0x7c      ↑  
    .byte 0b00000000 -- 0x7d 0x7b-0x7f番地には0x00を設定する必要がある  
    .byte 0b00000000 -- 0x7e      ↓  
    .byte 0b00000000 -- 0x7f      ↓
```

● minicube2.s

```
#-----  
#  
# NEC Electronics      V850ES/Jx3-L マイクロコントローラ  
#  
#-----  
# V850ES/JG3-L JF3-L サンプル・プログラム  
#-----  
# パルス幅測定モード編  
#-----  
# 【履歴】  
# 2009.8.-- 新規作成  
#-----  
# 【概要】  
# 本サンプル・プログラムは、MINICUBE2使用時に必要なリソースの確保を行っている  
# (CSIB0を用いてMINICUBE2を使用する場合の例)  
#-----  
  
--monitorROMセクションとして2Kバイトの空間を確保  
.section "MonitorROM", const  
.space 0x800, 0xff  
  
--デバッグ用割り込みベクタの確保  
.section "DBG0"  
.space 4, 0xff  
  
--シリアル通信用受信割り込みベクタの確保  
.section "INTCB0R"  
.space 4, 0xff  
  
--MonitorRAMセクションとして16バイトの空間を確保  
.section "MonitorRAM", bss  
.lcomm monitorramsym, 16, 4
```

```

● AppNote_Pulse.dir

# Sample link directive file (not use RTOS/use internal memory only)
#
# Copyright (C) NEC Electronics Corporation 2002
# All rights reserved by NEC Electronics Corporation.
#
# This is a sample file.
# NEC Electronics assumes no responsibility for any losses incurred by customers or
# third parties arising from the use of this file.
#
# Generated      : PM+ V6.31 [ 9 Jul 2007]
# Sample Version : E1.00b [12 Jun 2002]
# Device         : uPD70F3738 (C:¥Program Files¥NEC Electronics Tools¥DEV¥DF3738.800)
# Internal RAM   : 0x3ffb000 - 0x3ffefff
#
# NOTICE:
#   Allocation of SCONST, CONST and TEXT depends on the user program.
#
#   If interrupt handler(s) are specified in the user program then
#   the interrupt handler(s) are allocated from address 0 and
#   SCONST, CONST and TEXT are allocated after the interrupt handler(s).

SCONST : !LOAD ?R {
    .sconst      = $PROGBITS    ?A .sconst;
};

CONST   : !LOAD ?R {
    .const       = $PROGBITS    ?A .const;
};

TEXT    : !LOAD ?RX {
    .pro_epi_runtime = $PROGBITS    ?AX .pro_epi_runtime;
    .text          = $PROGBITS    ?AX .text;
};

### MINICUBE2用###
MROMSEG : !LOAD ?R 0x03F800{
    MonitorROM = $PROGBITS ?A MonitorROM;
};

```

内蔵ROMサイズが128Kバイト
製品の場合は 0x01F800

デフォルトのリンク・ディレクティブ
ファイルとの差分（追加コード）

MINICUBE2用の予約領域を確保。

```

SIDATA : !LOAD ?RW V0x3ffb000 {
    .tidata.byte = $PROGBITS ?AW .tidata.byte;
    .tibss.byte = $NOBITS ?AW .tibss.byte;
    .tidata.word = $PROGBITS ?AW .tidata.word;
    .tibss.word = $NOBITS ?AW .tibss.word;
    .tidata = $PROGBITS ?AW .tidata;
    .tibss = $NOBITS ?AW .tibss;
    .sidata = $PROGBITS ?AW .sidata;
    .sibss = $NOBITS ?AW .sibss;
};

```

```

DATA : !LOAD ?RW V0x3ffb100 {
    .data = $PROGBITS ?AW .data;
    .sdata = $PROGBITS ?AWG .sdata;
    .sbss = $NOBITS ?AWG .sbss;
    .bss = $NOBITS ?AW .bss;
};

```

```

### MINICUBE2用###
MRAMSEG : !LOAD ?RW V0x03FFEFF0{
    MonitorRAM = $NOBITS ?AW MonitorRAM;
};

```

デフォルトのリンク・ディレクティブ
ファイルとの差分（追加コード）

MINICUBE2用の予約領域を確保。

```

__tp_TEXT @ %TP_SYMBOL;
__gp_DATA @ %GP_SYMBOL &__tp_TEXT{DATA};
__ep_DATA @ %EP_SYMBOL;

```

```
● main.c
/*-----*/
/*
/*   NEC Electronics   V850ES/Jx3-L マイクロコントローラ
/*
/*-----*/
/*   V850ES/JG3-L サンプル・プログラム
/*-----*/
/*   パルス幅測定モード編
/*-----*/
/* 【履歴】
/*   2009.8.--   新規作成
/*-----*/
/* 【概要】
/*   本サンプル・プログラムは、16ビット・タイマ/イベント・カウンタ (TMP0) の
/*   パルス幅測定モードの使用例を示すものです。
/*
/*   スイッチ入力の立ち下り・立ち上りエッジを検出して割り込みを発生させ、
/*   チャタリング除去後の信号をP30ポートから出力して、TIP00ポートに入力します。
/*
/*   TIP00ポートのパルス幅を16ビット・タイマ/イベント・カウンタ (TMP0) の
/*   パルス幅測定モードで測定し、5secを上限としてスイッチの押下時間だけLED1を点灯します。
/*
/*   LED1はスイッチを離した直後から点灯し、LED1点灯中のスイッチ押下は無効です。
/*
/*   リセット解除後に動作停止状態の周辺機能について、このサンプル・プログラムで
/*   使用しない周辺機能は、設定していません。
/*
/*   <主な設定内容>
/*   ・pragma指令にて、割り込みハンドラの設定、周辺IOレジスタ名を記述可能にする
/*   ・チャタリング用10msウエイトの調整値を定義
/*   ・スイッチ入力の最大の受付時間を5secとする調整値を定義
/*   ・タイマP0 (TMP0) の16ビット・カウンタのオーバフロー時のカウント値を定義
/*   ・プロトタイプ宣言を実施
/*   ・内蔵周辺I/Oへのバス・ウエイト設定、ウォッチドッグ・タイマ2の動作停止、クロック設定処理
/*   ・未使用ポートの初期化
/*   ・外部割り込みポート (INTP0) の初期化
/*   ・P30ポート、TIP00ポート、PCM3ポートの初期化
/*   ・タイマP0 (TMP0) の初期化
/*
/*   <タイマP0設定>
/*   ・TP0CTL0にカウント・クロックとしてfxx/32 (625KHz) を設定
```

```

/*      ・TP0CTL1に動作モードとしてパルス幅測定モードを設定
/*      ・TP0IOC1にキャプチャ・トリガ入力信号として両エッジ検出を設定
/*      ・INTTP0CC0 優先度7, 割り込みマスク解除
/*      ・INTTP0OV          割り込みをマスク
/*
/*      <メイン処理>
/*      ・TMP0の16ビット・カウンタのオーバフロー時に回数をカウントし, オーバフロー・フラグをクリア
/*
/*      <外部割り込みポート (INTP0) 割り込み処理>
/*      ・スイッチ入力のチャタリング除去
/*      ・チャタリング除去後の信号をP30ポートから出力し, TIP00ポートへ入力
/*
/*      <タイマP0 (INTTP0CC0) 割り込み処理>
/*      ・キャプチャ・トリガのエッジを検出し, LED1を点灯させる
/*
/* 【ポート入出力の設定】
/*
/* 入力ポート: P03, P32 (TIP00)
/* 出力ポート: P30, PCM3
/* 空きポート: P02, P04-P06, P10-P11, P31, P33-P39, P50-P55, P70-P711, P90-P915,
/*             PCM0-PCM2, PCT0, PCT1, PCT4, PCT6, PDH0-PDH5, PDL0-PDL15
/*             未使用端子は全て出力ポート (Low出力) に設定しておく
/*
/*-----*/

/*-----*/
/*      pragma指令      */
/*-----*/

#pragma ioreg                /* 周辺IOレジスタ名称を記述可にする      */
#pragma interrupt INTP0      f_int_intp0      /* 外部割り込み (INTP0) 割り込みハンドラ指定      */
#pragma interrupt INTTP0CC0 f_int_inttp0cc0 /* タイマP0 (INTTP0CC0) 割り込みハンドラ指定      */

/*-----*/
/*      定数定義      */
/*-----*/

#define LIMIT_10ms_WAIT      ( 0x6F9B )      /* 10ms ウエイトの調整用定数定義      */
#define LIMIT_5sec_VALUE     ( 20 * 1000 * 1000 * 5 ) /* 5secのカウンタ・クロック定数定義      */
#define OVER_FLOW_VALUE     ( 65536 )      /* 16ビット・カウンタ・オーバフロー値      */
#define TMP0_COUNT_CLK_DIV  ( 32 )      /* 16ビット・カウンタのカウント・クロック分周比      */
#define PORT_HIGH           ( 1 )      /* ポートの状態 (HIGH)      */
#define PORT_LOW            ( 0 )      /* ポートの状態 (LOW)      */

```

TMQを使用する場合は,
INTTQ0CC0を指定

```

/*-----*/
/*      グローバル変数定義      */
/*-----*/
static unsigned long ulOVFCnt;          /* オーバフロー・カウンタ */

/*-----*/
/*      プロトタイプ宣言      */
/*-----*/
static void f_init( void );           /* 初期化関数 */
static void f_init_clk_bus_wdt2( void ); /* クロック・バス・WDT2初期化関数 */
static void f_init_port_func( void ); /* ポート/兼用機能初期化関数 */
static void f_init_int_tmp0( void ); /* TMP0初期化関数 */

/*****
/*      メイン・モジュール      */
*****/
void main( void )
{

    f_init();                          /* 初期化実施 */

    __EI();                             /* 割り込み許可 */

    while( 1 )                          /* メイン・ループ(無限ループ) */
    {
        __DI();                          /* 割り込み禁止 */
        if ( TP0OVF == 1 )                /* TMP0の16ビット・カウンタがオーバーフロー */
        {
            TP0OVF = 0;                    /* TMP0のオーバーフロー・フラグをクリア */
            ulOVFCnt++;                    /* 16ビット・カウンタのオーバーフロー回数をインクリメント */
        }
        __EI();                             /* 割り込み許可 */
    }
}

```

TMQを使用する場合は、TQ0OVFフラグをチェック

TMQを使用する場合は、TQ0OVFフラグをクリア

```

/*-----*/
/*   初期化モジュール   */
/*-----*/
static void f_init( void )
{
    f_init_clk_bus_wdt2();          /* 内蔵周辺I/Oへのバス・ウエイト, WDT2, クロック設定処理 */

    f_init_port_func();            /* ポート/兼用機能の設定 */

    f_init_int_tmp0();            /* TMP0タイマ設定 */

    return;
}

```

```

/*-----*/
/* クロック, バス・ウエイト, WDT2初期化 */
/*-----*/
static void f_init_clk_bus_wdt2( void )
{
    VSWC = 0x01;                  /* 内蔵周辺へのバス・ウエイト設定*/

                                /* OCDMを「通常動作モード」に指定*/

```

```

#pragma asm
    st.b   r0, PRCMD
    st.b   r0, OCDM
#pragma endasm

```

特定レジスタへのアクセスは、アセンブラで記述する必要があります。注意が必要です。

```

RSTOP = 1;                      /* 内蔵発振器を停止 */
WDTM2 = 0x00;                   /* ウォッチドッグ・タイマ2を停止*/

```

```

/* クロック分周を「なし」に設定 */

```

```

#pragma asm
    push   r10
    mov    0x80, r10
    st.b   r10, PRCMD
    st.b   r10, PCC
    pop    r10
#pragma endasm

```

特定レジスタへのアクセスは、アセンブラで記述する必要があります。注意が必要です。

```

PLLCTL = 0x03;                  /* PLLモードに設定 */

```

```

return;
}

```



```

/*-----*/
/*   ポート/兼用機能の設定   */
/*-----*/
static void f_init_port_func( void )
{

    P0   = 0x00;                /* P02-P06を出力LOWに設定   */
    PM0  = 0x83;
    PMC0 = 0x00;

    P1   = 0x00;                /* P10, P11を出力LOWに設定 */
    PM1  = 0xFC;
    PMC1 = 0x00;

    /* P32を兼用機能TIP00入力に設定 */
    P3   = 0x0001;            /* P30をHIGH, P31-P39をLOWに設定 */
    PM3  = 0xFC04;           /* P30, P31, P33-P39を出力モードに設定 */
    PFC3 = 0x0000;           /* P32をTIP00入力に設定   */
    PFCE3L = 0x04;
    PMC3 = 0x0004;

    #if(0) /* MINICUBE2使用時はP4をCSIB0として使用する為, */
        /* P4は空き端子として初期化しない(QB-V850ESJG3L-TB) */
        P4   = 0x00;                /* P40-P42を出力LOWに設定   */
        PM4  = 0xF8;
        PMC4 = 0x00;
    #endif

    P5   = 0x00;                /* P50-P55を出力LOWに設定   */
    PM5  = 0xC0;
    PMC5 = 0x00;

    P7H  = 0x00;                /* P70-P711を出力LOWに設定 */
    P7L  = 0x00;
    PM7H = 0xF0;
    PM7L = 0x00;

    P9   = 0x0000;            /* P90-P915を出力LOWに設定 */
    PM9  = 0x0000;
    PMC9 = 0x0000;

    PCM  = 0x08;                /* PCM0-PCM2を出力LOWに, PCM3に消灯パターンを設定*/
    PMCM = 0xF0;
    PMCCM = 0x00;
}

```

V850ES/JF3-Lでの設定値は0xFE

V850ES/JF3-LではP10のみ

V850ES/JF3-LではP31-P35,P38,P39

V850ES/JF3-Lでの設定値は0xFCC4

V850ES/JF3-LではP33-P35,P38,P39

V850ES/JF3-Lでは, レジスタが存在しないため, 設定しない。

V850ES/JF3-LではP70-P77

V850ES/JF3-Lでの設定値は0x1C3C

V850ES/JF3-LではP90,P91,P96-P99,P913-P915

```

PCT = 0x00; /* PCT0,1,4,6を出力LOWに設定 */
PMCT = 0xAC;
PMCCT = 0x00;

PDH = 0x00; /* PDH0-PDH5を出力LOWに設定 */
PMDH = 0xC0; /* V850ES/JF3-Lでの設定値は0xFC */
PMCDH = 0x00; /* V850ES/JF3-LではPDH0,PDH1

PDL = 0x0000; /* PDL0-PDL15を出力LOWに設定 */
PMDL = 0x0000;
PMCDL = 0x0000;

/* INTP0両エッジ指定 */
INTF0 = 0x08; /* INTP0立ち下りエッジ指定 */
INTR0 = 0x08; /* INTP0立ち上りエッジ指定

PIC0 = 0x07; /* INTP0 優先度7, マスク解除

return;

}

/*-----*/
/* タイマP0 (TMP0) の設定 (パルス幅測定を指定) */
/*-----*/
static void f_init_int_tmp0( void )
{
/* 変数の初期化 */
uloVFCnt = 0; /* オーバフロー・カウンタをクリア */

/* タイマP0機能設定 */
TP0CTL0 = 0x05; /* カウント・クロックをfxx/32に設定 */
TP0CTL1 = 0x06; /* FailSafe: TP0CE=0にしてTMP0停止 */
TP0IOC1 = 0x03; /* パルス幅測定モード指定 */
/* キャプチャ・トリガ入力信号両エッジを検出

/* 注意1: パルス幅測定モードでは, 以下のレジスタは設定しません */
/* <未設定レジスタ>
/* - TP0IOC0レジスタ
/* - TP0IOC2レジスタ
/* - TP0OPT0レジスタ

```

```

/*          - TP0CCR0レジスタ          */
/*          - TP0CCR1レジスタ          */
/*          - TP0CNTレジスタ          */

/* 割り込み制御レジスタ設定 */
TP0CCIC0 = 0x07          /* INTTP0CC0 優先度7, マスク解除          */
TP0OVMK  = 1;          /* INTTP0OV 16ビット・カウンタのオーバーフロー割り込みをマスク */
TP0CE    = 1;          /* TMP0動作開始          */

return;

}

/*****
/* 割り込みモジュール (外部割り込み)
/*****
__interrupt
void f_int_intp0( void )
{
    unsigned long loop_wait;
    unsigned char start_sw;
    unsigned char end_sw;

    /* 10msウエイト前のスイッチ入力値を取得 */
    start_sw = ( unsigned char ) P0.3;

    /* チャタリング対策用 10msウエイト */
    for( loop_wait = 0 ; loop_wait < LIMIT_10ms_WAIT ; loop_wait++ )
    {
        __nop();
    }

    /* 10msウエイト後のスイッチ入力値を取得 */
    end_sw = ( unsigned char ) P0.3;

    if ( start_sw == end_sw )          /* 10msウエイトの前と後でスイッチ状態変化がない */
    {
        P3L.0 = start_sw;          /* P30ポート出力値更新          */
    }

    return;
}

```

TMQを使用する場合は, TQ0xxxレジスタに値を設定

```

/*****
/*   割り込みモジュール (TMP0)   */
/*****

__interrupt
void f_int_inttp0cc0( void )
{

    unsigned long ulTMP0CLKCnt;
    unsigned long ulCPUCLKCnt;
    unsigned long loop_wait;

    if ( P3L.0 == PORT_HIGH )
    {
        if ( TP0OVF == 1 )
        {
            TP0OVF = 0;
            ulOVFCnt++;
        }

        ulTMP0CLKCnt = OVER_FLOW_VALUE * ulOVFCnt + TP0CCR0;

        TP0CE = 0;

        ulCPUCLKCnt = ulTMP0CLKCnt * TMP0_COUNT_CLK_DIV;

        if ( ulCPUCLKCnt >= LIMIT_5sec_VALUE )
        {
            ulCPUCLKCnt = LIMIT_5sec_VALUE;
        }

        ulCPUCLKCnt /= 6;

        PCM.3 = PORT_LOW;

        /* スイッチの入力時間分ウエイト */
        for ( loop_wait = 0 ; loop_wait < ulCPUCLKCnt ; loop_wait++ )
        {
            __nop();
        }

        PCM.3 = PORT_HIGH;
    }
}

```

TMQを使用する場合は、TQ0OVFフラグをチェック

TMQを使用する場合は、TQ0CCR0レジスタのカウンタ値を取得

TMQを使用する場合は、TQ0CEフラグにてタイマ動作停止

```

TPOCE = 1; /* TMP0動作許可 */
PIFO = 0; /* LED点等処理中に発生した外部割り込み (INTTP0) を無視する */
}
else
{
  TPOOVF = 0; /* TMP0のオーバーフロー・フラグをクリア */
  ulOVFCnt = 0; /* 16ビット・カウンタのオーバーフロー回数を初期化 */
}

return;
}

```

TMQを使用する場合は、TQ0CEフラグにてタイマ動作開始

