

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。

サンプル・プログラム関数使用法説明書【タイマ P 編】 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)

この説明書では、V850E/IA4 用の 16 ビット・タイマ/イベント・カウンタ P (TMP) のサンプル・プログラム関数について説明します。

なお、この説明書では V850E/IA4 を使用した場合で記述しています。V850E/IA3, V850ES/IK1, V850ES/IE2 を使用した場合は、この説明書を参考にしてください。

ご注意

このサンプル・プログラムはあくまで参考用のものであり、当社がこの動作を保証するものではありません。
このサンプル・プログラムを使用する場合、お客様のセット上で十分な評価をしたうえでご使用いただきますようお願いいたします。

CMOSデバイスの一般的注意事項

入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。

CMOSデバイスの入力がノイズなどに起因して、 V_{IL} (MAX.) から V_{IH} (MIN.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定な場合はもちろん、 V_{IL} (MAX.) から V_{IH} (MIN.) までの領域を通過する遷移期間中にチャタリングノイズ等が入らないようご使用ください。

未使用入力の処理

CMOSデバイスの未使用端子の入力レベルは固定してください。

未使用端子入力については、CMOSデバイスの入力に何も接続しない状態で動作させるのではなく、プルアップかプルダウンによって入力レベルを固定してください。また、未使用の入出力端子が出力となる可能性（タイミングは規定しません）を考慮すると、個別に抵抗を介して V_{DD} または GND に接続することが有効です。

資料中に「未使用端子の処理」について記載のある製品については、その内容を守ってください。

静電気対策

MOSデバイス取り扱いの際は静電気防止を心がけてください。

MOSデバイスは強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジン・ケース、または導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

初期化以前の状態

電源投入時、MOSデバイスの初期状態は不定です。

電源投入時の端子の出力状態や入出力設定、レジスタ内容などは保証しておりません。ただし、リセット動作やモード設定で定義している項目については、これらの動作ののちに保証の対象となります。

リセット機能を持つデバイスの電源投入後は、まずリセット動作を実行してください。

電源投入切断順序

内部動作および外部インタフェースで異なる電源を使用するデバイスの場合、原則として内部電源を投入した後に外部電源を投入してください。切断の際には、原則として外部電源を切断した後に内部電源を切断してください。逆の電源投入切断順により、内部素子に過電圧が印加され、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。

資料中に「電源投入切断シーケンス」についての記載のある製品については、その内容を守ってください。

電源OFF時における入力信号

当該デバイスの電源がOFF状態の時に、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。

資料中に「電源OFF時における入力信号」についての記載のある製品については、その内容を守ってください。

本製品のうち、外国為替及び外国貿易法の規定により規制貨物等（または役務）に該当するものについては、日本国外に輸出する際に、同法に基づき日本国政府の輸出許可が必要です。

非該当品 : μ PD70F3184, 70F3186, 70F3329, 70F3713, 70F3714

ユーザ判定品 : μ PD703183, 703185, 703186, 703327, 703329

- 本資料に記載されている内容は2006年7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。当社製品の不具合により生じた生命、身体および財産に対する損害の危険を最小限度にするために、冗長設計、延焼対策設計、誤動作防止設計等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。

はじめに

- 注 意 1.** この説明書で使用するプログラムは、NEC エレクトロニクスのホームページ (<http://www.necel.co.jp/>) よりダウンロードしてください。
- 2.** このサンプル・プログラムを使用する場合は、次のスタートアップ・ファイル、リンク・ディレクトティブ・ファイルを参照し、必要に応じて修正してください。
- ・スタートアップ・ファイル : IA4_start.s
 - ・リンク・ディレクトティブ・ファイル : IA4_link.dir

凡 例 関数一覧表は次のように構成されています。

テーマ（ハードウェア略号）

【機能】	テーマの説明
【関数名】	サンプル関数の名前
【引き数】	引き数の型と概要
【処理内容】	サンプル関数の処理内容
【起動方法】	関数の呼び出し条件
【使用SFR】	レジスタ名と設定内容
【call関数】	呼び出し関数の名前と機能
【変数】	サンプル関数での使用変数の型，名前，概要
【割り込み】	関数名
【割り込み要因】	名称
【ファイル名】	対応するサンプル・プログラム・ファイル名
【注意事項】	関数使用上の注意。使い方

割り込み関数

【関数名】	割り込み関数の名前
【概要】	処理の内容
【要因】	割り込み名と発生条件
【call関数】	なし
【変数】	変数名，機能
【ファイル名】	対応するサンプル・プログラム・ファイル名
【注意事項】	なし

製品間の違い V850E/IA4 と V850E/IA3, V850ES/IK1, V850ES/IE2 の 16 ビット・タイマ/イベント・カウンタ P (TMP) についての製品間の違いを次に示します。

項目	V850E/IA4	V850E/IA3	V850ES/IK1	V850ES/IE2
TOP31端子	あり	なし	あり	
カウント・クロック	fxx/2, fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128, fxx/256	fxx/2, fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128, fxx/256	fxx, fxx/2, fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128	

備考 fxx : 周辺クロック周波数

関連資料 関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。あらかじめご了承ください。

V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2 に関する資料

資料名	資料番号
V850E1 ユーザーズ・マニュアル アーキテクチャ編	U14559J
V850E/IA3, V850E/IA4 ユーザーズ・マニュアル ハードウェア編	U16543J
V850ES ユーザーズ・マニュアル アーキテクチャ編	U15943J
V850ES/IK1 ユーザーズ・マニュアル ハードウェア編	U16910J
V850ES/IE2 ユーザーズ・マニュアル ハードウェア編	U17716J
V850シリーズによるインバータ制御 アプリケーション・ノート ホール・センサによるベクトル制御編	U17338J
V850シリーズによるインバータ制御 アプリケーション・ノート エンコーダによるベクトル制御編	U17324J
V850シリーズによるインバータ制御 アプリケーション・ノート ゼロクロス検出による120度通電方式制御編	U17209J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート シリアル通信 (UARTA) 編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18233J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート シリアル通信 (CSIB) 編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18234J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート DMA機能編 (V850E/IA3, V850E/IA4)	U18235J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマM編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18236J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート ウォッチドッグ・タイマ編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18237J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマP編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	このマニュアル
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマQ編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18239J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマENC編 (V850E/IA3, V850E/IA4)	U18240J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート ポート機能編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18241J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート クロック・ジェネレータ編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18242J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート スタンバイ機能編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18243J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート 割り込み機能編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18244J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート A/Dコンバータ0, 1編 (V850E/IA3, V850E/IA4, V850ES/IK1, V850ES/IE2)	U18245J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート A/Dコンバータ2編 (V850E/IA3, V850E/IA4)	U18246J

目 次

- 16 ビット・タイマ/イベント・カウンタ P (TMPn) (n = 0-3)
 インターバル・タイマ・モード ... 8
- 16 ビット・タイマ/イベント・カウンタ P (TMPk) (k = 0, 2)
 外部イベント・カウント・モード ... 12
- 16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)
 外部トリガ・パルス出力モード ... 15
- 16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)
 ワンショット・パルス出力モード ... 19
- 16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)
 PWM出力モード ... 23
- 16 ビット・タイマ/イベント・カウンタ P (TMPn) (n = 0-3)
 フリー・ランニング・タイマ・モード ... 27
- 16 ビット・タイマ/イベント・カウンタ P (TMPk) (k = 0, 2)
 パルス幅測定モード ... 32

16ビット・タイマ/イベント・カウンタ P (TMPn) (n = 0-3)

インターバル・タイマ・モード

(1/2)

【機能】	TP0CCR0 レジスタで設定したインターバル間隔で TOP00 端子から 50%デューティの PWM 波形を出力します。 また、TP0CCR1 レジスタで設定した値と 16ビット・カウンタのカウンタ値の一致で TOP01 端子出力を反転します。 TMP0-TMP3 で実現可能です。
【関数名】	timerp_interval
【引き数】	なし
【処理内容】	fx/32 のカウンタ・クロックでカウンタ動作を行い、カウンタの値が TP0CCR0 レジスタの値と一致した次のカウンタのタイミングで、TOP00 端子出力を反転させて割り込みを発生し、カウンタをクリアします。また、カウンタの値が TP0CCR1 レジスタの値と一致した次のカウンタのタイミングで TOP01 端子出力を反転させて割り込みを発生します。 TOP00 端子、TOP01 端子ともにハイ・レベル・スタートです。
【起動方法】	timerp_interval_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC0 ・ TOP00, TOP01 端子出力の設定 ・ TOP00, TOP01 端子出力レベルの設定 TP0CCR0 16ビット・カウンタのコンペア・レジスタ TP0CCR1 16ビット・カウンタのコンペア・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int
【割り込み要因】	INTTP0CC0 INTTP0CC1
【ファイル名】	timerp_interval¥timerp_1.c, timerp_interval¥MAIN.C

【注 意 事 項】 レジスタの設定については、次の注意が必要です。

TPnCTL1 (n = 0-3)

ビット 7 : TMP0, TMP1 のみ設定可能です。TMP2, TMP3 には必ず 0 を設定してください。

ビット 6 : TMP0, TMP2, TMP3 のみ設定可能です。TMP1 には必ず 0 を設定してください。

ビット 5 : TMP0, TMP2 のみ設定可能です。TMP1, TMP3 には必ず 0 を設定してください。

TPmIOC0 (m = 0, 2, 3)

ビット 1, 0 : TMP0 のみ有効です。TMP2, TMP3 には必ず 0 を設定してください。

インターバル間隔は次に示す式で計算できます。

$$\text{インターバル間隔} = (\text{TP0CCR0 レジスタ設定値} + 1) \times \text{カウント} \cdot \text{クロック周期}$$

【関 数 名】 timerp_interval_st

【引 き 数】 なし

【処 理 内 容】 timerp_interval の起動関数です。

【起 動 方 法】 timerp_interval 関数のあとにコールしてください。

【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御

【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_interval¥timerp_1.c

【注 意 事 項】 なし

割り込み関数

【関 数 名】 timerp_TP0CC0_int

【概 要】 ユーザ定義

【要 因】 INTTP0CC0 16 ビット・カウンタのカウンタ値と TP0CCR0 の一致

【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_interval¥timerp_1.c

【注 意 事 項】 なし

【関 数 名】 timerp_TP0CC1_int

【概 要】 ユーザ定義

【要 因】 INTTP0CC1 16 ビット・カウンタのカウンタ値と TP0CCR1 の一致

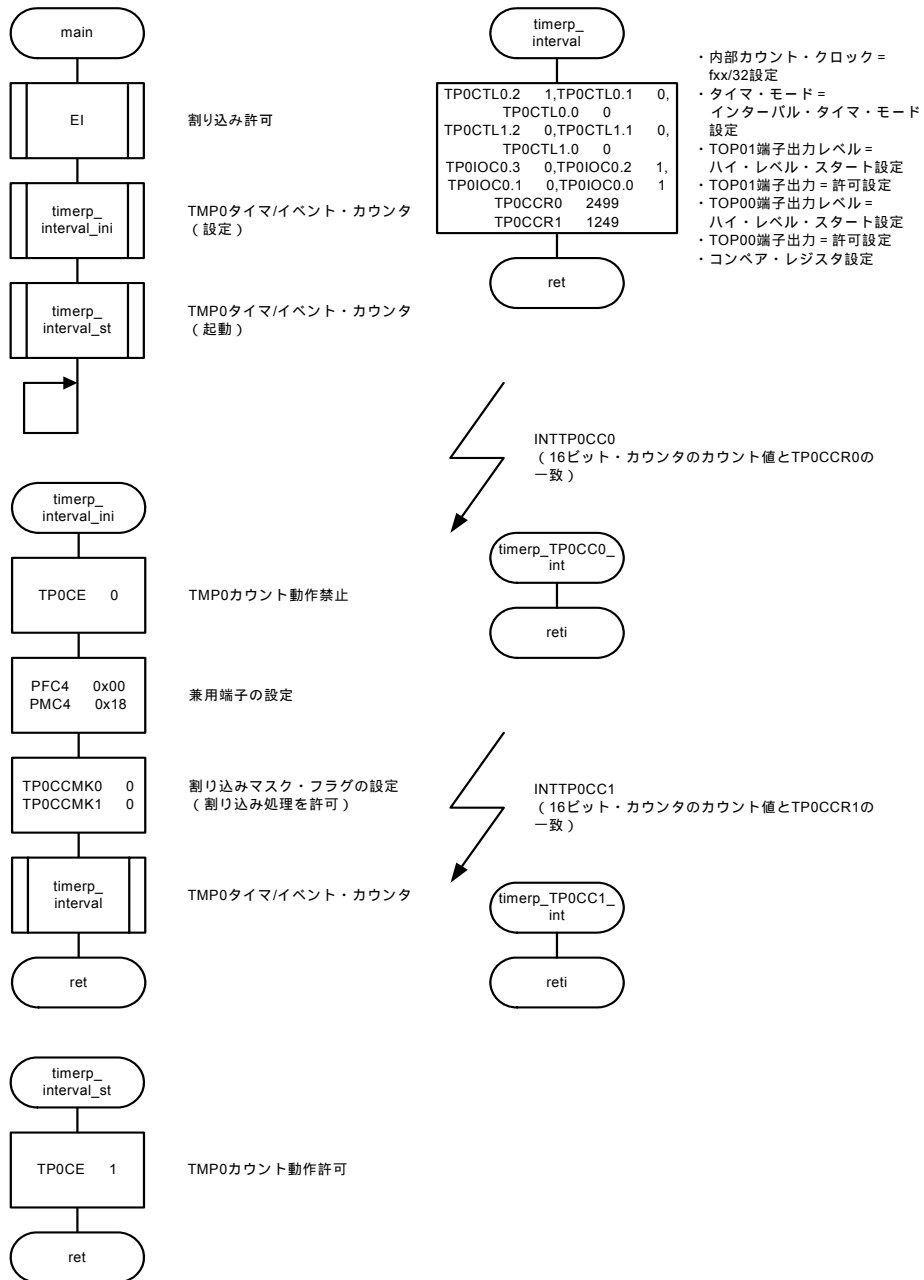
【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_interval¥timerp_1.c

【注 意 事 項】 なし

16ビット・タイマ/イベント・カウンタP (TMPn) インターバル・タイマ・モード



16 ビット・タイマ/イベント・カウンタ P (TMPk) (k = 0, 2)

外部イベント・カウント・モード

【機能】	外部イベント・カウント入力 (TIP00 端子) の有効エッジをカウントし, TP0CCR0 レジスタで設定したカウントごとに割り込み要求信号 (INTTP0CC0) を発生します (このとき, 16 ビット・カウンタもクリアします)。 また, 16 ビット・カウンタのカウント値と TP0CCR1 レジスタの値のコンペア一致で割り込み要求信号 (INTTP0CCR1) を発生します。 TMP0, TMP2 で実現可能です。
【関数名】	timerp_event_count
【引き数】	なし
【処理内容】	外部イベント・カウント入力の有効エッジをカウントし, カウンタの値が TP0CCR0 レジスタの値と一致した次のカウントのタイミングで, 割り込みを発生し, カウンタをクリアします。 また, カウンタの値が TP0CCR1 レジスタの値と一致した次のカウントのタイミングで, 割り込みを発生します。
【起動方法】	timerp_event_count_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC2 ・外部イベント・カウント入力信号 (TIP00 端子) の有効エッジの設定 ・外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0CCR0 16 ビット・カウンタのコンペア・レジスタ TP0CCR1 16 ビット・カウンタのコンペア・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int
【割り込み要因】	INTTP0CC0 INTTP0CC1
【ファイル名】	timerp_event_count¥timerp_2.c, timerp_event_count¥MAIN.C
【注意事項】	・ TP0CCR0, TP0CCR1 レジスタに 0x0000 を設定することは禁止します。 ・ レジスタの設定については, 次の注意が必要です。 TPkCTL1 ビット 7: TMP0 のみ設定可能です。TMP2 には必ず 0 を設定してください。 TPkIOC0 0x00 を設定してください。

【関 数 名】 timerp_event_count_st
【引 き 数】 なし
【処 理 内 容】 timerp_event_count の起動関数です。
【起 動 方 法】 timerp_event_count 関数のあとにコールしてください。
【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御
【 call 関数】 なし
【変 数】 なし
【フ ァ イ ル 名】 timerp_event_count¥timerp_2.c
【注 意 事 項】 なし

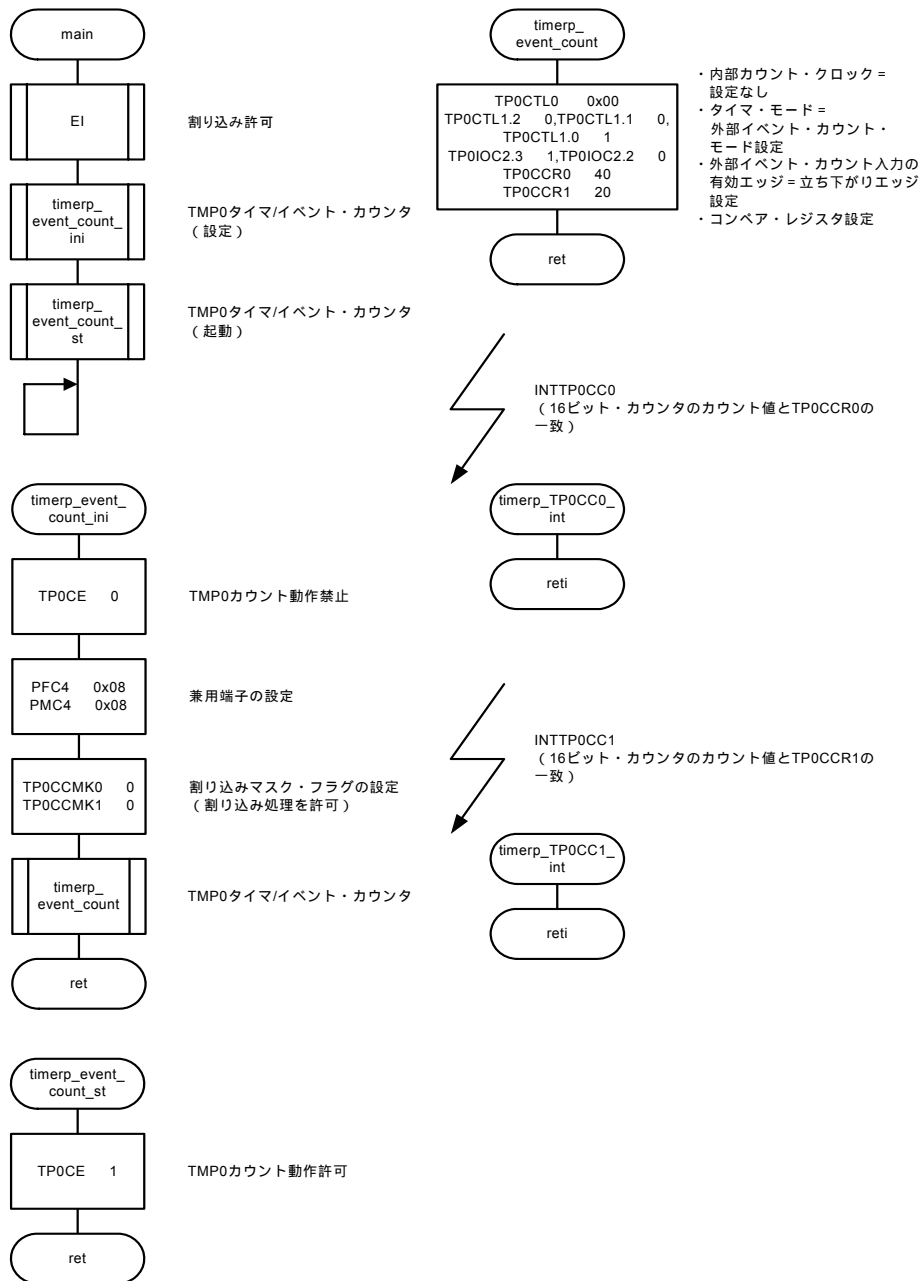
割り込み関数

【関 数 名】 timerp_TP0CC0_int
【概 要】 ユーザ定義
【要 因】 INTTP0CC0 16 ビット・カウンタのカウント値と TP0CCR0 の一致
【 call 関数】 なし
【変 数】 なし
【フ ァ イ ル 名】 timerp_event_count¥timerp_2.c
【注 意 事 項】 なし

【関 数 名】 timerp_TP0CC1_int
【概 要】 ユーザ定義
【要 因】 INTTP0CC1 16 ビット・カウンタのカウント値と TP0CCR1 の一致
【 call 関数】 なし
【変 数】 なし
【フ ァ イ ル 名】 timerp_event_count¥timerp_2.c
【注 意 事 項】 なし

16ビット・タイマ/イベント・カウンタP (TMPk)

外部イベント・カウント・モード



16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)

外部トリガ・パルス出力モード

(1/2)

【機能】	外部トリガ入力 (TIP00 端子) の有効エッジを検出すると 16 ビット・カウンタの動作を開始します。 TP0CCR0 レジスタとのコンペア一致で 16 ビット・カウンタをクリアします。 TP0CCR1 レジスタで設定した値と 16 ビット・カウンタのカウント値の一致で TOP01 端子を反転します。 TOP01 端子は 16 ビット・カウンタのクリア時に反転します。 TMP0, TMP2, TMP3 で実現可能です (TMP3 はソフトウェア・トリガのみ)。
【関数名】	timerp_trigger_pulse
【引き数】	なし
【処理内容】	外部トリガ入力の有効エッジ検出で, f _{clk} /32 のカウント・クロックのカウント動作を開始し, カウンタの値が TP0CCR0 レジスタの値と一致した次のカウントのタイミングで割り込みを発生し, カウンタをクリアします。 また, カウンタの値が TP0CCR1 レジスタの値と一致した次のカウントのタイミングで TOP01 端子出力を反転させて割り込みを発生します。 TOP01 端子はハイ・レベル・スタートです。
【起動方法】	timerp_trigger_pulse_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC0 ・ TOP00, TOP01 端子出力の設定 ・ TOP00, TOP01 端子出力レベルの設定 TP0IOC2 ・ 外部イベント・カウント入力信号 (TIP00 端子) の有効エッジの設定 ・ 外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0CCR0 16 ビット・カウンタのコンペア・レジスタ TP0CCR1 16 ビット・カウンタのコンペア・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int
【割り込み要因】	INTTP0CC0 INTTP0CC1
【ファイル名】	timerp_trigger_pulse¥timerp_3.c, timerp_trigger_pulse¥MAIN.C

【注 意 事 項】 ・コンペア・レジスタは一斉書き込みです。
 タイマ動作中にコンペア・レジスタの値を変更する場合は、最後に TP0CCR1 レジスタの値を変更してください。

・レジスタの設定については、次の注意が必要です。

TPmCTL1
 ビット 7 : TMP0 のみ設定可能です。TMP2, TMP3 には必ず 0 を設定してください。
 ビット 5 : TMP0, TMP2 のみ設定可能です。TMP3 には必ず 0 を設定してください。

TPmIOC0
 ビット 1, 0 : TMP0 のみ有効です。TMP2, TMP3 には必ず 0 を設定してください。

PWM 波形のアクティブ・レベル幅，周期，およびデューティは次に示す式で計算できます。

アクティブ・レベル幅 = (TP0CCR1 レジスタの設定値) × カウント・クロック周期

周期 = (TP0CCR0 レジスタの設定値 + 1) × カウント・クロック周期

デューティ = (TP0CCR1 レジスタの設定値) / (TP0CCR0 レジスタの設定値 + 1)

【関 数 名】 timerp_trigger_pulse_st

【引 き 数】 なし

【処 理 内 容】 timerp_trigger_pulse の起動関数です。

【起 動 方 法】 timerp_trigger_pulse 関数のあとにコールしてください。

【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御

【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_trigger_pulse¥timerp_3.c

【注 意 事 項】 なし

割り込み関数

【関数名】 timerp_TP0CC0_int

【概要】 ユーザ定義

【要因】 INTTP0CC0 16ビット・カウンタのカウント値と TP0CCR0 の一致

【call 関数】 なし

【変数】 なし

【ファイル名】 timerp_trigger_pulse¥timerp_3.c

【注意事項】 なし

【関数名】 timerp_TP0CC1_int

【概要】 ユーザ定義

【要因】 INTTP0CC1 16ビット・カウンタのカウント値と TP0CCR1 の一致

【call 関数】 なし

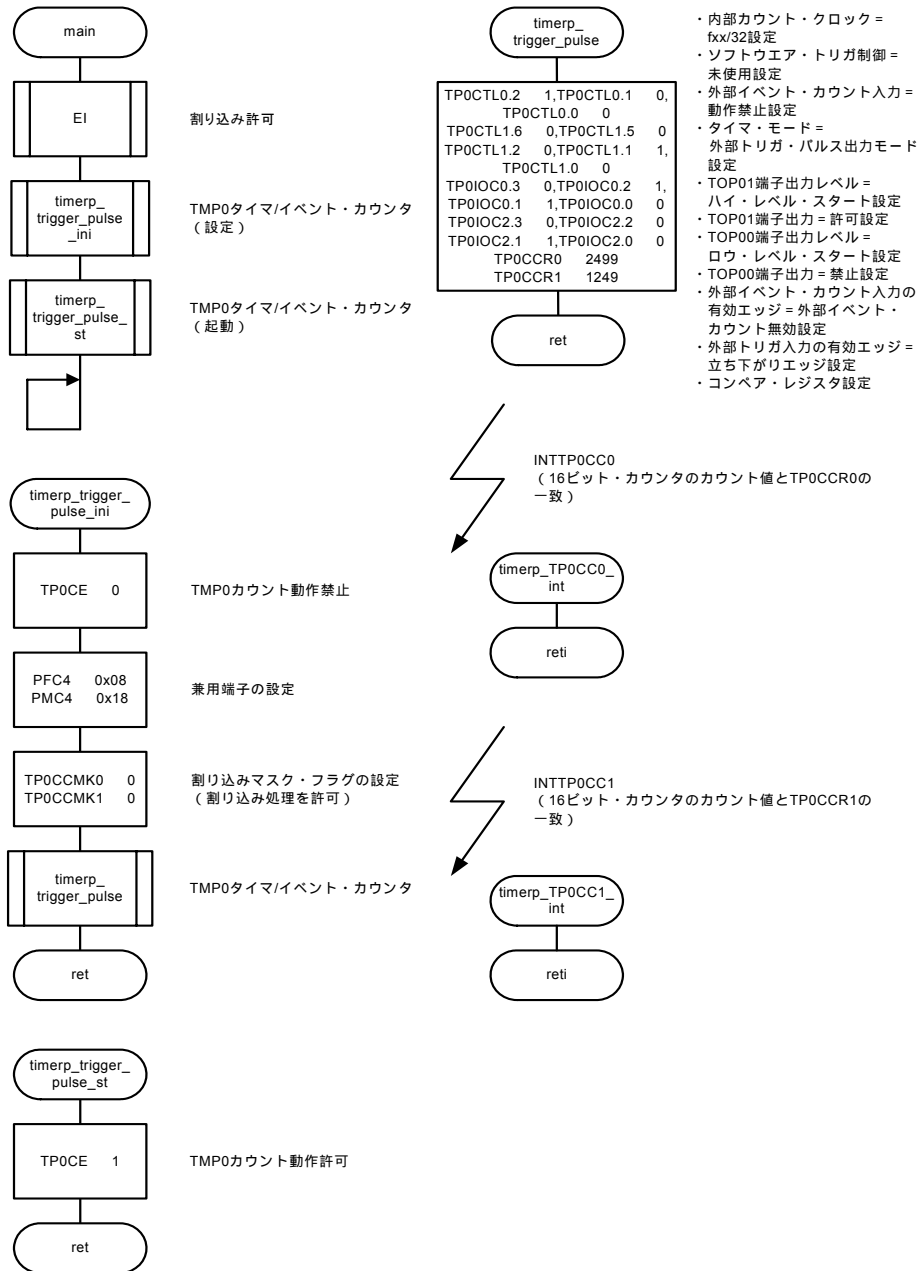
【変数】 なし

【ファイル名】 timerp_trigger_pulse¥timerp_3.c

【注意事項】 なし

16ビット・タイマ/イベント・カウンタP (TMPm)

外部トリガ・パルス出力モード



16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)

ワンショット・パルス出力モード

(1/2)

【機能】	外部トリガ入力 (TIP00 端子) の有効エッジを検出すると 16 ビット・カウンタの動作を開始します。 TP0CCR0 レジスタとのコンペア一致で 16 ビット・カウンタをクリアし、カウントを停止します。 TP0CCR1 レジスタで設定した値と 16 ビット・カウンタのカウント値の一致で TOP01 端子を反転します。 TOP01 端子は 16 ビット・カウンタのクリア時に反転します。 TMP0, TMP2, TMP3 で実現可能です (TMP3 はソフトウェア・トリガのみ)。
【関数名】	timerp_1shot_pulse
【引き数】	なし
【処理内容】	外部トリガ入力の有効エッジ検出で、 $f_{xx}/32$ のカウント・クロックのカウント動作を開始し、カウンタの値が TP0CCR0 レジスタの値と一致した次のカウントのタイミングで割り込みを発生し、TOP01 端子出力を反転させてカウンタをクリアし、カウント動作を停止します。 また、カウンタの値が TP0CCR1 レジスタの値と一致した次のカウントのタイミングで TOP01 端子出力を反転させて割り込みを発生します。 TOP01 端子はハイ・レベル・スタートです。
【起動方法】	timerp_1shot_pulse_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC0 ・ TOP00, TOP01 端子出力の設定 ・ TOP00, TOP01 端子出力レベルの設定 TP0IOC2 ・ 外部イベント・カウント入力信号 (TIP00 端子) の有効エッジの設定 ・ 外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0CCR0 16 ビット・カウンタのコンペア・レジスタ TP0CCR1 16 ビット・カウンタのコンペア・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int
【割り込み要因】	INTTP0CC0 INTTP0CC1
【ファイル名】	timerp_1shot_pulse¥timerp_4.c, timerp_1shot_pulse¥MAIN.C

【注 意 事 項】 ・レジスタの設定については、次の注意が必要です。

TPmCTL1

ビット 7 : TMP0 のみ設定可能です。TMP2, TMP3 には必ず 0 を設定してください。

ビット 5 : TMP0, TMP2 のみ設定可能です。TMP3 には必ず 0 を設定してください。

TPmIOC0

ビット 1, 0 : TMP0 のみ有効です。TMP2, TMP3 には必ず 0 を設定してください。

ワンショット・パルスの出力ディレイ期間、およびアクティブ・レベル幅は次に示す式で計算できます。

出力ディレイ期間 = (TP0CCR1 レジスタの設定値) × カウント・クロック周期

アクティブ・レベル幅 = (TP0CCR0 レジスタの設定値 - TP0CCR1 レジスタの設定値 + 1) × カウント・クロック周期

【関 数 名】 timerp_1shot_pulse_st

【引 き 数】 なし

【処 理 内 容】 timerp_1shot_pulse の起動関数です。

【起 動 方 法】 timerp_1shot_pulse 関数のあとにコールしてください。

【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御

【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_1shot_pulse¥timerp_4.c

【注 意 事 項】 なし

割り込み関数

【関数名】 timerp_TP0CC0_int

【概要】 ユーザ定義

【要因】 INTTP0CC0 16ビット・カウンタのカウント値と TP0CCR0 の一致

【call 関数】 なし

【変数】 なし

【ファイル名】 timerp_1shot_pulse¥timerp_4.c

【注意事項】 なし

【関数名】 timerp_TP0CC1_int

【概要】 ユーザ定義

【要因】 INTTP0CC1 16ビット・カウンタのカウント値と TP0CCR1 の一致

【call 関数】 なし

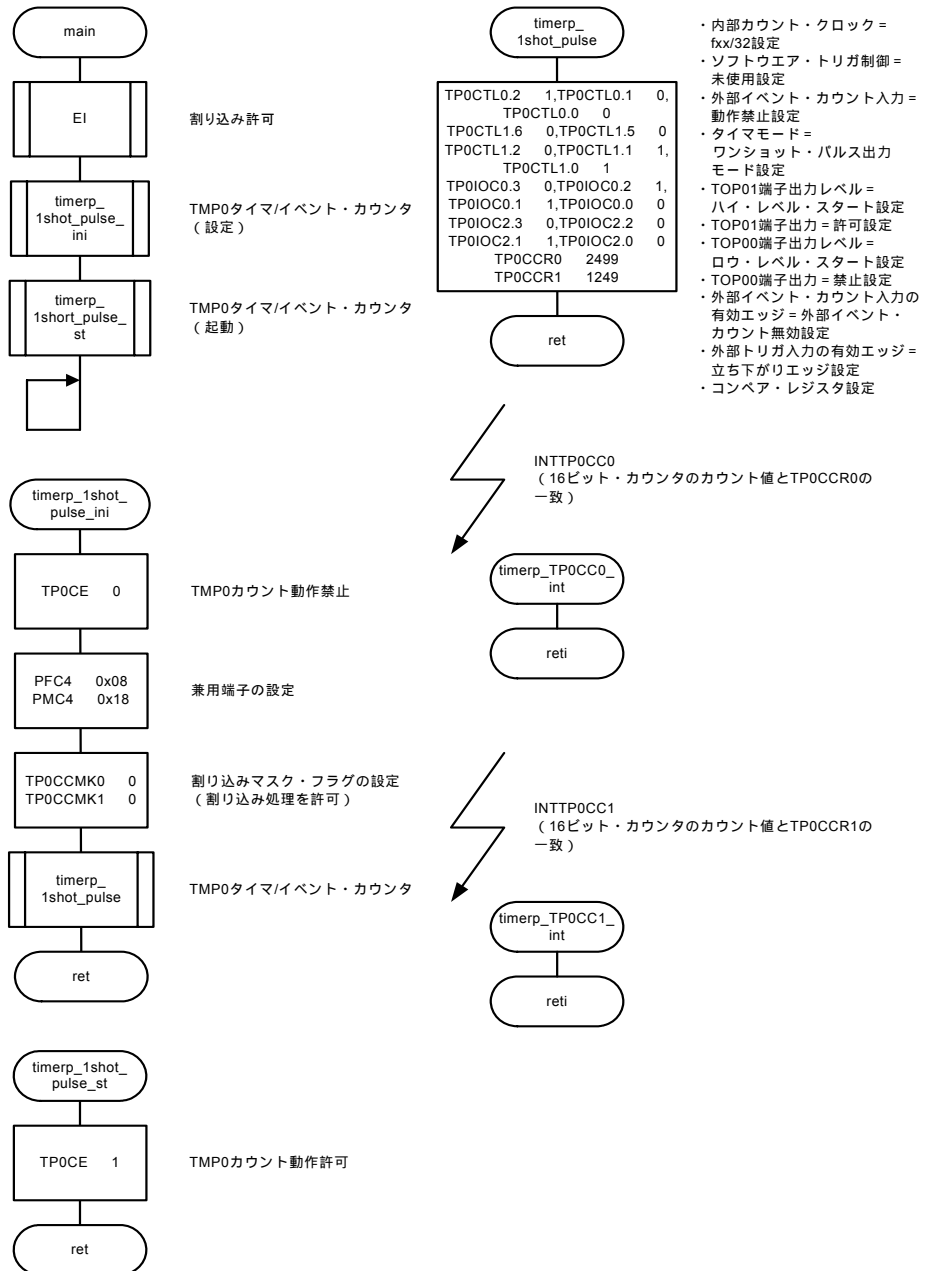
【変数】 なし

【ファイル名】 timerp_1shot_pulse¥timerp_4.c

【注意事項】 なし

16ビット・タイマ/イベント・カウンタP (TMPm)

ワンショット・パルス出力モード



16 ビット・タイマ/イベント・カウンタ P (TMPm) (m = 0, 2, 3)

PWM 出力モード

(1/2)

【機能】	TP0CE ビットをセットすることで 16 ビット・カウンタの動作を開始します。 TP0CCR0 レジスタとのコンペア一致で 16 ビット・カウンタをクリアし, TOP00 端子を反転し, TP0CCR0 レジスタの値 + 1 を半周期とする 50 %デューティの PWM 波形を出力します。 TP0CCR1 レジスタで設定した値と 16 ビット・カウンタのカウンタ値の一致で TOP01 端子を反転します。 TOP01 端子は 16 ビット・カウンタのクリア時に反転します。 TMP0, TMP2, TMP3 で実現可能です
【関数名】	timerp_pwm_output
【引き数】	なし
【処理内容】	fx/32 のカウンタ・クロックでカウンタ動作を行い, カウンタの値が TP0CCR0 レジスタの値と一致した次のカウンタのタイミングで TOP00, TOP01 端子出力を反転させて割り込みを発生し, カウンタをクリアします。 また, カウンタの値が TP0CCR1 レジスタの値と一致した次のカウンタのタイミングで TOP01 端子出力を反転させて割り込みを発生します。 TOP00, TOP01 端子ともにハイ・レベル・スタートです。
【起動方法】	timerp_pwm_output_st 関数のコールにより起動します。
【使用 SFR】	TP0CTL0 カウンタ・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC0 ・ TOP00, TOP01 端子出力の設定 ・ TOP00, TOP01 端子出力レベルの設定 TP0IOC2 ・ 外部イベント・カウンタ入力信号 (TIP00 端子) の有効エッジの設定 ・ 外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0CCR0 16 ビット・カウンタのコンペア・レジスタ TP0CCR1 16 ビット・カウンタのコンペア・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int
【割り込み要因】	INTTP0CC0 INTTP0CC1
【ファイル名】	timerp_pwm_output¥timerp_5.c, timerp_pwm_output¥MAIN.C

【注 意 事 項】 ・コンペア・レジスタは一斉書き込みです。
 タイマ動作中にコンペア・レジスタの値を変更する場合は、最後に TP0CCR1 レジスタの値を変更してください。

・レジスタの設定については、次の注意が必要です。

TPmCTL1
 ビット 7 : TMP0 のみ設定可能です。TMP2, TMP3 には必ず 0 を設定してください。
 ビット 5 : TMP0, TMP2 のみ設定可能です。TMP3 には必ず 0 を設定してください。

TPmIOC0
 ビット 1, 0 : TMP0 のみ有効です。TMP2, TMP3 には必ず 0 を設定してください。

TOP01 端子から出力される PWM 波形のアクティブ・レベル幅，周期，およびデューティは，次に示す式で計算できます。

アクティブ・レベル幅 = (TP0CCR1 レジスタの設定値) × カウント・クロック周期

周期 = (TP0CCR0 レジスタの設定値 + 1) × カウント・クロック周期

デューティ = (TP0CCR1 レジスタの設定値) / (TP0CCR0 レジスタの設定値 + 1)

【関 数 名】 timerp_pwm_output_st

【引 き 数】 なし

【処 理 内 容】 timerp_pwm_output の起動関数です。

【起 動 方 法】 timerp_pwm_output 関数のあとにコールしてください。

【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御

【 call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_pwm_output*timerp_5.c

【注 意 事 項】 なし

割り込み関数

【関数名】 timerp_TP0CC0_int

【概要】 ユーザ定義

【要因】 INTTP0CC0 16ビット・カウンタのカウント値と TP0CCR0 の一致

【call 関数】 なし

【変数】 なし

【ファイル名】 timerp_pwm_output¥timerp_5.c

【注意事項】 なし

【関数名】 timerp_TP0CC1_int

【概要】 ユーザ定義

【要因】 INTTP0CC1 16ビット・カウンタのカウント値と TP0CCR1 の一致

【call 関数】 なし

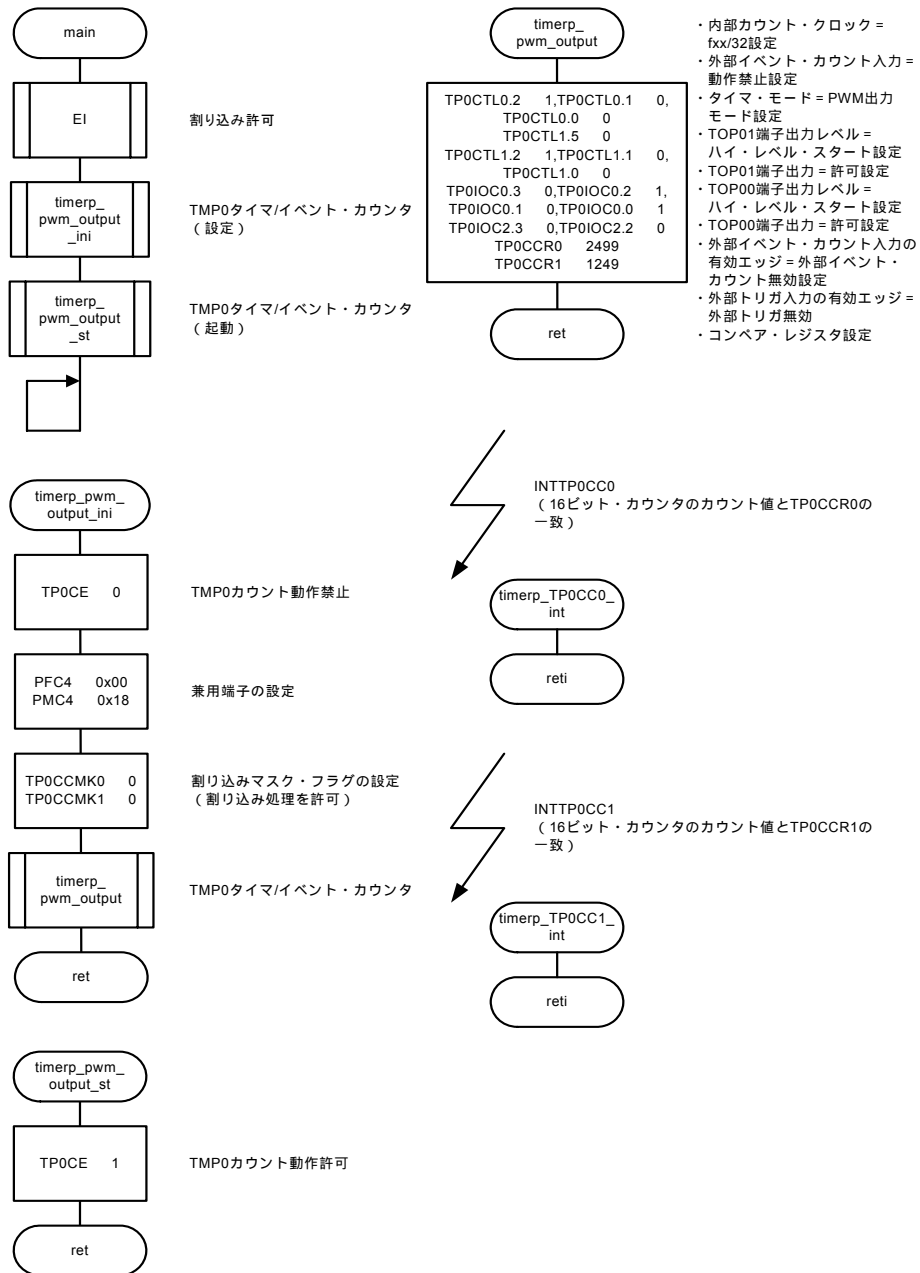
【変数】 なし

【ファイル名】 timerp_pwm_output¥timerp_5.c

【注意事項】 なし

16ビット・タイマ/イベント・カウンタP (TMPm)

PWM 出力モード



16 ビット・タイマ/イベント・カウンタ P (TMPn) (n = 0-3)

フリー・ランニング・タイマ・モード

(1/2)

【機能】	TP0CCR0 レジスタと 16 ビット・カウンタのカウンタ値とのコンペア一致で TOP00 端子の出力を反転します (コンペア機能)。 また、キャプチャ・トリガ入力 (TIP01 端子) の有効エッジ検出で 16 ビット・カウンタのカウンタ値を格納します (キャプチャ機能)。 コンペア機能は TMP0-TMP3 で実現可能です。 キャプチャ機能は TMP0, TMP2 で実現可能です。
【関数名】	timerp_free_running
【引き数】	なし
【処理内容】	fx/32 のカウンタ・クロックでカウンタ動作を行い、カウンタの値が TP0CCR0 レジスタの値と一致した次のカウンタのタイミングで TOP00 端子出力を反転させて割り込みを発生し、カウンタをクリアします。 また、TIP01 端子からの有効エッジ検出により、カウンタの値を TP0CCR1 レジスタにキャプチャし、割り込みを発生します。 カウンタのオーバフローが検出されると、INTTP0OV 割り込みを発生します。 TOP00 端子はハイ・レベル・スタートです。
【起動方法】	timerp_free_running_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC0 ・ TOP00, TOP01 端子出力の設定 ・ TOP00, TOP01 端子出力レベルの設定 TP0IOC1 キャプチャ・トリガ入力信号 (TIP00, TIP01 端子) の有効エッジの設定 TP0IOC2 ・ 外部イベント・カウンタ入力信号 (TIP00 端子) の有効エッジの設定 ・ 外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0OPT0 TP0CCR0, TP0CCR1 レジスタのキャプチャ / コンペア選択 TP0CCR0 16 ビット・カウンタのコンペア・レジスタ TP0CCR1 16 ビット・カウンタのキャプチャ・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0CC1_int timerp_TP0OV_int
【割り込み要因】	INTTP0CC0 INTTP0CC1 INTTP0OV

【ファイル名】 timerp_free_running¥timerp_6.c,
timerp_free_running¥MAIN.C

【注意事項】 ・レジスタの設定については、次の注意が必要です。

TPnCTL1

ビット7：TMP0, TMP1のみ設定可能です。TMP2, TMP3には必ず0を設定してください。

ビット6：TMP0, TMP2, TMP3のみ設定可能です。TMP1には必ず0を設定してください。

ビット5：TMP0, TMP2のみ設定可能です。TMP1, TMP3には必ず0を設定してください。

TPmIOC0 (m = 0, 2, 3)

ビット1, 0：TMP0のみ有効です。TMP2, TMP3には必ず0を設定してください。

TPnOPT0

ビット5, 4：TMP0, TMP2のみ有効です。TMP1, TMP3には必ず0を設定してください。

【関数名】 timerp_free_running_st

【引き数】 なし

【処理内容】 timerp_free_running の起動関数です。

【起動方法】 timerp_free_running 関数のあとにコールしてください。

【使用SFR】 TP0CTL0.TP0CE TMP0の動作制御

【call関数】 なし

【変数】 なし

【ファイル名】 timerp_free_running¥timerp_6.c

【注意事項】 なし

割り込み関数

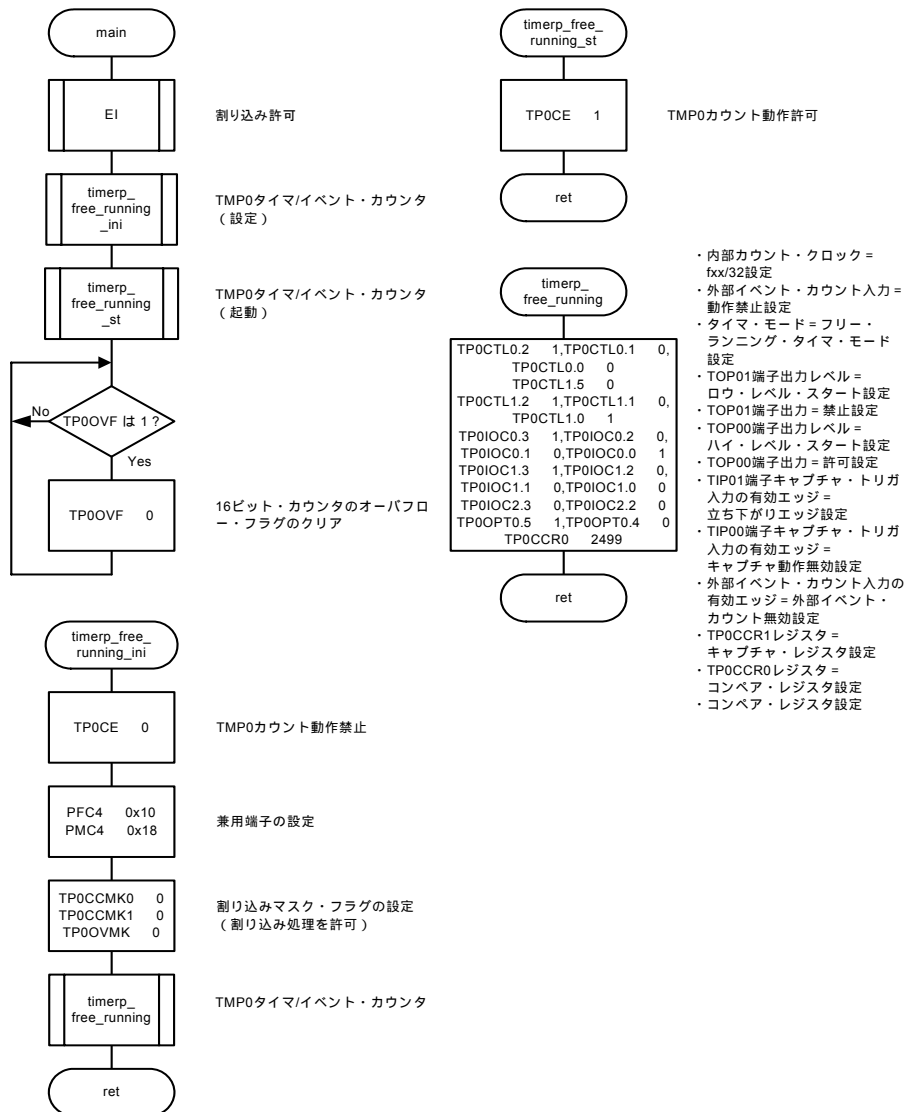
【関数名】 timerp_TP0CC0_int
【概要】 ユーザ定義
【要因】 INTTP0CC0 16ビット・カウンタのカウント値と TP0CCR0 の一致
【call 関数】 なし
【変数】 なし
【ファイル名】 timerp_free_running¥timerp_6.c
【注意事項】 なし

【関数名】 timerp_TP0CC1_int
【概要】 ユーザ定義
【要因】 INTTP0CC1 TIP01 端子入力の有効エッジ検出
【call 関数】 なし
【変数】 なし
【ファイル名】 timerp_free_running¥timerp_6.c
【注意事項】 なし

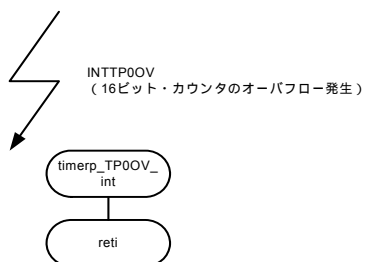
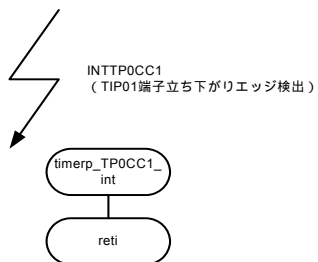
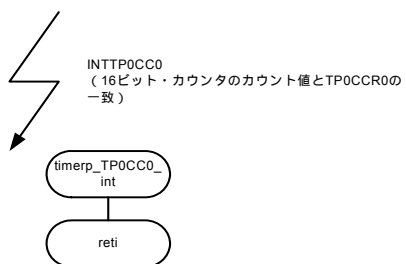
【関数名】 timerp_TP0OV_int
【概要】 ユーザ定義
【要因】 INTTP0OV 16ビット・カウンタのオーバフロー発生
【call 関数】 なし
【変数】 なし
【ファイル名】 timerp_free_running¥timerp_6.c
【注意事項】 なし

16ビット・タイマ/イベント・カウンタP (TMPn)

フリー・ランニング・タイマ・モード (1/2)



16ビット・タイマ/イベント・カウンタP (TMPn)
フリー・ランニング・タイマ・モード (2/2)



16 ビット・タイマ/イベント・カウンタ P (TMPk) (k = 0, 2)

パルス幅測定モード

【機能】	キャプチャ・トリガ入力 (TIP00 端子) の有効エッジで 16 ビット・カウンタのカウンタ値を TP0CCR0 レジスタに格納し, 16 ビット・カウンタをクリアします。 また, TIP00 端子入力の有効エッジで割り込みを発生させ, TP0CCR0 レジスタの値を読み込むことにより TIP00 端子の有効エッジ間隔を測定します。 TMP0, TMP2 で実現可能です
【関数名】	timerp_pulse_measure
【引き数】	なし
【処理内容】	fx/32 のカウンタ・クロックでカウンタ動作を行い, TIP00 端子入力の有効エッジで 16 ビット・カウンタのカウンタ値を TP0CCR0 レジスタに格納して割り込みを発生し, カウンタをクリアします。 カウンタのオーバフローが検出されると, INTTP0OV 割り込みを発生します。
【起動方法】	timerp_pulse_measure_st 関数のコールにより起動します。
【使用 S F R】	TP0CTL0 カウント・クロックの選択 TP0CTL1 タイマ・モードの選択 TP0IOC1 キャプチャ・トリガ入力信号 (TIP00, TIP01 端子) の有効エッジの設定 TP0IOC2 ・外部イベント・カウンタ入力信号 (TIP00 端子) の有効エッジの設定 ・外部トリガ入力信号 (TIP00 端子) の有効エッジの設定 TP0OPT0 TP0CCR0, TP0CCR1 レジスタのキャプチャ / コンペア選択 TP0CCR0 16 ビット・カウンタのキャプチャ・レジスタ
【call 関数】	main メイン関数
【変数】	なし
【割り込み】	timerp_TP0CC0_int timerp_TP0OV_int
【割り込み要因】	INTTP0CC0 INTTP0OV
【ファイル名】	timerp_pulse_measure¥timerp_7.c, timerp_pulse_measure¥MAIN.C
【注意事項】	・カウンタ・クロックとして遅いクロックを選択した場合, TP0CTL0.TP0CE ビットをセットした直後にキャプチャ・トリガが入力されると, TP0CCR0 レジスタに 0x0000 ではなく 0xFFFF がキャプチャされる場合があります。 ・レジスタの設定については, 次の注意が必要です。 TPkCTL1 ビット 7 : TMP0 のみ設定可能です。TMP2 には必ず 0 を設定してください。 TPkIOC0 0x00 を設定してください。

パルス幅は次に示す式で計算できます。

$$\text{パルス幅} = (\text{TP0CCR0 レジスタの値} + 1) \times \text{カウント} \cdot \text{クロック周期}$$

16 ビット・カウンタのオーバーフローが検出された場合のパルス幅は次に示す式で計算できます。

$$\text{パルス幅} = (\text{TP0CCR0 レジスタの値} + 0x10001) \times \text{カウント} \cdot \text{クロック周期}$$

【関 数 名】 timerp_pulse_measure_st

【引 き 数】 なし

【処 理 内 容】 timerp_pulse_measure の起動関数です。

【起 動 方 法】 timerp_pulse_measure 関数のあとにコールしてください。

【使 用 S F R】 TP0CTL0.TP0CE TMP0 の動作制御

【call 関数】 なし

【変 数】 なし

【フ ァ イ ル 名】 timerp_pulse_measure¥timerp_7.c

【注 意 事 項】 なし

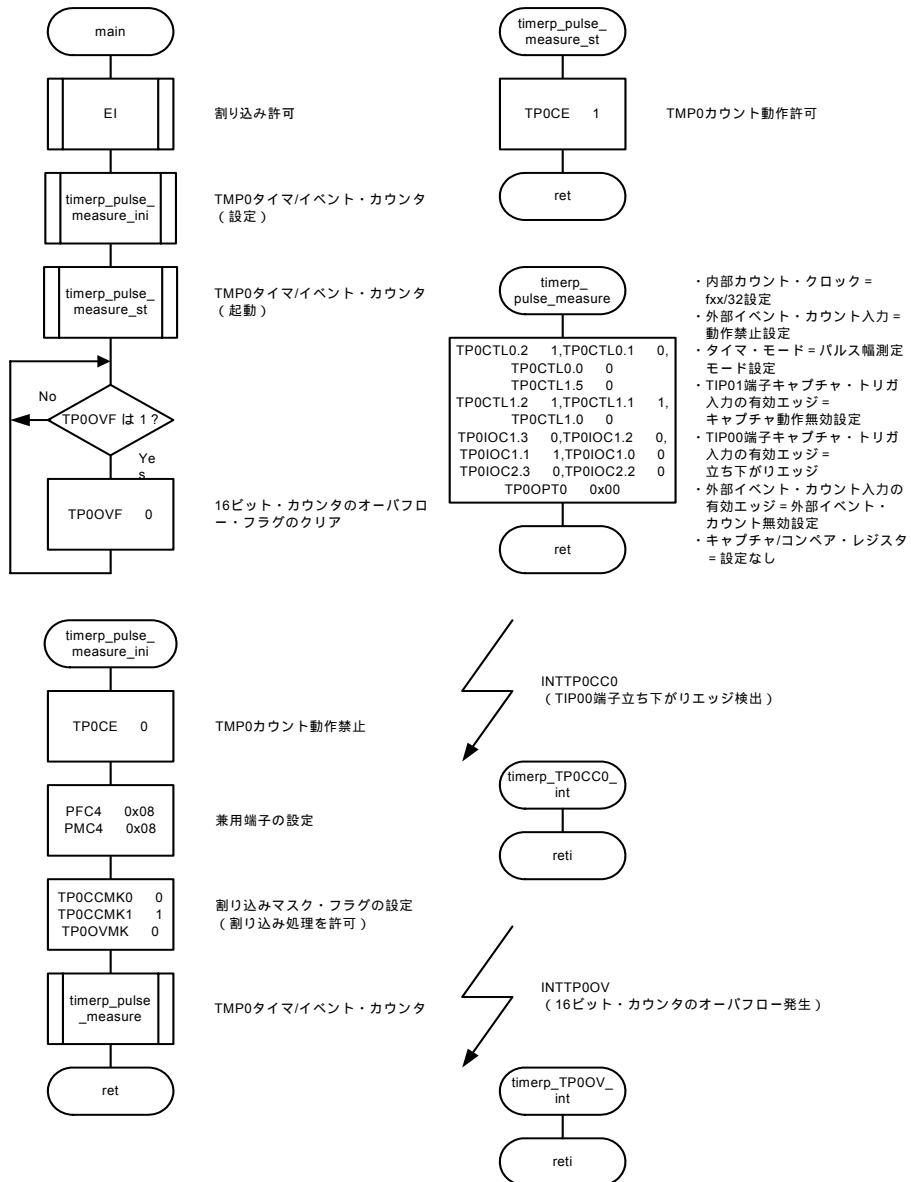
割り込み関数

【関 数 名】 timerp_TP0CC0_int
【概 要】 ユーザ定義
【要 因】 INTTP0CC0 TIP00 端子入力の有効エッジ検出
【call 関数】 なし
【変 数】 なし
【フ ァ イ ル 名】 timerp_pulse_measure¥timerp_7.c
【注 意 事 項】 なし

【関 数 名】 timerp_TP0OV_int
【概 要】 ユーザ定義
【要 因】 INTTP0OV 16 ビット・カウンタのオーバーフロー発生
【call 関数】 なし
【変 数】 なし
【フ ァ イ ル 名】 timerp_pulse_measure¥timerp_7.c
【注 意 事 項】 なし

16ビット・タイマ/イベント・カウンタP (TMPk)

パルス幅測定モード



【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

—— お問い合わせ先 ——

【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.necel.co.jp/>

【営業関係，技術関係お問い合わせ先】

半導体ホットライン

(電話：午前 9:00～12:00，午後 1:00～5:00)

電 話 : 044-435-9494

E-mail : info@necel.com

【資料請求先】

NECエレクトロニクスのホームページよりダウンロードいただくか，NECエレクトロニクスの販売特約店へお申し付けください。

C04.2T