

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事務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。お客様に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。

お客様各位

資料中の「日立製作所」、「日立XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って三菱電機株式会社及び株式会社日立製作所のマイコン、ロジック、アナログ、ディスクリット半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。従いまして、本資料中には「日立製作所」、「株式会社日立製作所」、「日立半導体」、「日立XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

ルネサステクノロジ ホームページ (<http://www.renesas.com>)

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

ご注意

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注意ください。

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すとは、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。

SH7144 シリーズ SH7144、SH7145

アプリケーションノート

ご注意

1. 本書に記載の製品及び技術のうち「外国為替及び外国貿易法」に基づき安全保障貿易管理関連貨物・技術に該当するものを輸出する場合、または国外に持ち出す場合は日本国政府の許可が必要です。
2. 本書に記載された情報の使用に際して、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等の権利に対する保証または実施権の許諾を行うものではありません。また本書に記載された情報を使用した事により第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
3. 製品及び製品仕様は予告無く変更する場合がありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書をお求めになりご確認ください。
4. 弊社は品質・信頼性の向上に努めておりますが、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途にご使用をお考えのお客様は、事前に弊社営業担当迄ご相談をお願い致します。
5. 設計に際しては、特に最大定格、動作電源電圧範囲、放熱特性、実装条件及びその他諸条件につきましては、弊社保証範囲内でご使用いただきますようお願い致します。
保証値を超えてご使用された場合の故障及び事故につきましては、弊社はその責を負いません。また保証値内のご使用であっても半導体製品について通常予測される故障発生率、故障モードをご考慮の上、弊社製品の動作が原因でご使用機器が人身事故、火災事故、その他の拡大損害を生じないようにフェールセーフ等のシステム上の対策を講じて頂きますようお願い致します。
6. 本製品は耐放射線設計をしておりません。
7. 本書の一部または全部を弊社の文書による承認なしに転載または複製することを堅くお断り致します。
8. 本書をはじめ弊社半導体についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

製品に関する一般的注意事項

1. NC 端子の処理

【注意】NC端子には、何も接続しないようにしてください。

NC(Non-Connection)端子は、内部回路に接続しない場合の他、テスト用端子やノイズ軽減などの目的で使用します。このため、NC端子には、何も接続しないようにしてください。

2. 未使用入力端子の処理

【注意】未使用の入力端子は、ハイまたはローレベルに固定してください。

CMOS製品の入力端子は、一般にハイインピーダンス入力となっています。未使用端子を開放状態で動作させると、周辺ノイズの誘導により中間レベルが発生し、内部で貫通電流が流れて誤動作を起こす恐れがあります。未使用の入力端子は、入力をプルアップかプルダウンによって、ハイまたはローレベルに固定してください。

3. 初期化前の処置

【注意】電源投入時は、製品の状態は不定です。

すべての電源に電圧が印加され、リセット端子にローレベルが入力されるまでの間、内部回路は不確定であり、レジスタの設定や各端子の出力状態は不定となります。この不定状態によってシステムが誤動作を起こさないようにシステム設計を行ってください。リセット機能を持つ製品は、電源投入後は、まずリセット動作を実行してください。

4. 未定義・リザーブアドレスのアクセス禁止

【注意】未定義・リザーブアドレスのアクセスを禁止します。

未定義・リザーブアドレスは、将来の機能拡張用の他、テスト用レジスタなどが割り付けられています。これらのレジスタをアクセスしたときの動作および継続する動作については、保証できませんので、アクセスしないようにしてください。

目次

第 1 章	SH7144 シリーズ—SH7144、SH7145 アプリケーションノート—応用編—使用手引	
1.1	応用編構成.....	1-2
第 2 章	応用編	
2.1	パルスの High および Low 幅測定	2-2
2.1.1	仕様.....	2-2
2.1.2	使用機能説明	2-3
2.1.3	動作説明.....	2-4
2.1.4	ソフトウェア説明	2-5
2.1.5	フローチャート	2-6
2.1.6	プログラムリスト	2-8
2.2	パルス出力.....	2-10
2.2.1	仕様.....	2-10
2.2.2	使用機能説明	2-11
2.2.3	動作説明.....	2-12
2.2.4	ソフトウェア説明	2-13
2.2.5	フローチャート	2-14
2.2.6	プログラムリスト	2-15
2.3	PWM4 相出力.....	2-16
2.3.1	仕様.....	2-16
2.3.2	使用機能説明	2-17
2.3.3	動作説明.....	2-19
2.3.4	ソフトウェア説明	2-20
2.3.5	フローチャート	2-22
2.3.6	プログラムリスト	2-23
2.4	PWM7 相出力.....	2-25
2.4.1	仕様.....	2-25
2.4.2	使用機能説明	2-26
2.4.3	動作説明.....	2-28
2.4.4	ソフトウェア説明	2-29
2.4.5	フローチャート	2-31
2.4.6	プログラムリスト	2-32
2.5	正相・逆相 PWM3 相出力.....	2-34
2.5.1	仕様.....	2-34
2.5.2	使用機能説明	2-35
2.5.3	動作説明.....	2-36
2.5.4	ソフトウェア説明	2-37
2.5.5	フローチャート	2-38
2.5.6	プログラムリスト	2-39

2.6	相補 PWM3 相出力.....	2-40
2.6.1	仕様.....	2-40
2.6.2	使用機能説明.....	2-41
2.6.3	動作説明.....	2-43
2.6.4	ソフトウェア説明.....	2-48
2.6.5	フローチャート.....	2-50
2.6.6	プログラムリスト.....	2-51
2.7	2相エンコーダカウント.....	2-53
2.7.1	仕様.....	2-53
2.7.2	使用機能説明.....	2-54
2.7.3	動作説明.....	2-57
2.7.4	ソフトウェア説明.....	2-59
2.7.5	フローチャート.....	2-61
2.7.6	プログラムリスト.....	2-65
2.8	外部トリガによるタイマ波形の遮断.....	2-68
2.8.1	仕様.....	2-68
2.8.2	使用機能説明.....	2-69
2.8.3	動作説明.....	2-71
2.8.4	ソフトウェア説明.....	2-72
2.8.5	フローチャート.....	2-73
2.8.6	プログラムリスト.....	2-74
2.9	DC モータ制御用信号出力.....	2-76
2.9.1	仕様.....	2-76
2.9.2	使用機能説明.....	2-77
2.9.3	動作説明.....	2-79
2.9.4	ソフトウェア説明.....	2-80
2.9.5	フローチャート.....	2-81
2.9.6	プログラムリスト.....	2-82
2.10	MTU による A/D 変換の開始.....	2-84
2.10.1	仕様.....	2-84
2.10.2	使用機能説明.....	2-85
2.10.3	動作説明.....	2-88
2.10.4	ソフトウェア説明.....	2-89
2.10.5	フローチャート.....	2-90
2.10.6	プログラムリスト.....	2-91
2.11	DMAC を使用した RAM モニタ.....	2-92
2.11.1	仕様.....	2-92
2.11.2	使用機能説明.....	2-94
2.11.3	動作説明.....	2-98
2.11.4	ソフトウェア説明.....	2-99
2.11.5	フローチャート.....	2-100
2.11.6	プログラムリスト.....	2-103

1. SH7144 シリーズ—SH7144、SH7145 アプリケーションノート—応用編—使用手引

本アプリケーションノートは、図 1.1 に示すように 2 部構成になっています。

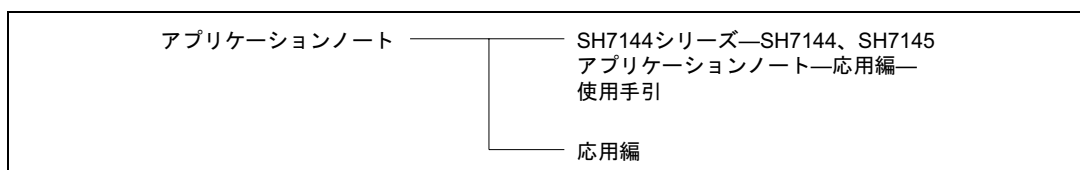


図 1.1 アプリケーションノート構成

(1) SH7144 シリーズ—SH7144、SH7145 アプリケーションノート—応用編—使用手引

SH7144 シリーズ—SH7144、SH7145 アプリケーションノート—応用編—の使用法について説明しています。

(2) 応用編

SH7144、SH7145 の内蔵周辺機能（タイマ、シリアルコミュニケーションインタフェース、A/D コンバータ、PWM、I/O ポート、割込み、低消費電力モード等）を組み合わせ使用した場合の使用法を簡単なタスク例をもとに説明しています。

1.1 応用編構成

応用編は図 1.2 に示す構成で内蔵周辺機能を組み合わせて使用した場合の使用法について説明しています。

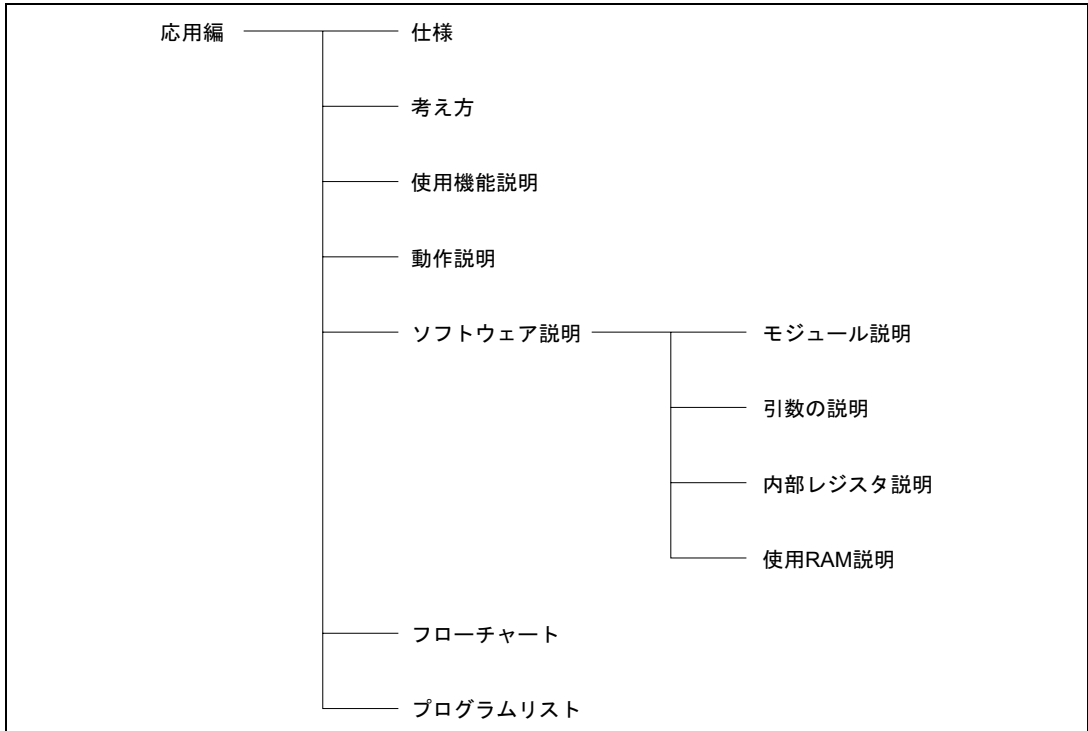


図 1.2 応用編構成

(1) 仕様

タスク例のシステム仕様について説明しています。

(2) 考え方

タスク例のシステムを実現するための方法を説明しています。

(3) 使用機能説明

タスク例で使用する周辺機能の特徴および周辺機能の割り付けについて説明しています。

(4) 動作説明

タスク例の動作をタイミングチャートを使用し説明しています。

(5) ソフトウェア説明

(a) モジュール説明

タスク例を動作させるソフトウェアのモジュールについて説明しています。

(b) 引数の説明

モジュールを実行する際に必要な入力引数と、実行後の出力引数について説明しています。

(c) 内部レジスタ説明

モジュールで設定する周辺機能の内部レジスタ（タイマコントロールレジスタ、シリアルモードレ

ジスタ等) について説明しています。

(d) 使用 RAM 説明

モジュールで使用する RAM のラベル名および機能について説明しています。

(6) フローチャート

タスク例を実行するソフトウェアについてフローチャートを使用し説明しています。

(7) プログラムリスト

タスク例を実行するソフトウェアのプログラムリストを示しています。

2. 応用編

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプット キャプチャ)
仕様				

- (1) 図2.1に示すように、パルスのHigh幅およびLow幅の時間を測定し、結果をRAMに格納します。
- (2) 内蔵周辺クロック $P_{\phi}=40.0\text{MHz}$ 動作時、パルスのHigh幅およびLow幅は25.0nsから1.63msまで25.0ns単位で測定可能です。

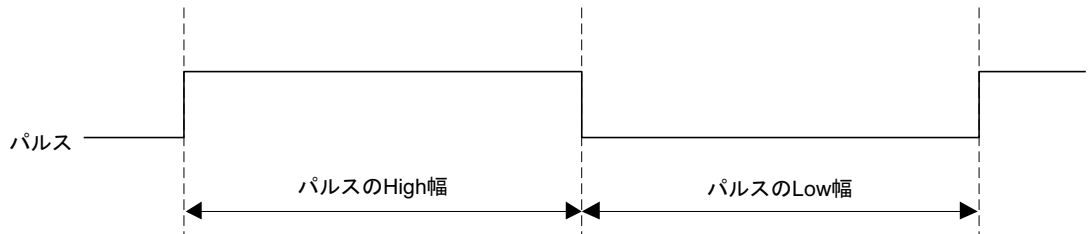


図 2.1 パルス幅測定タイミング

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプットキャプチャ)
使用機能説明				

- (1) 本タスク例では、ch0を使用してパルスのHigh幅およびLow幅を測定します。
- (a) 図2.2にch0のブロック図を示します。本タスクでは、以下の機能を使用します。
- パルスの立ち上がりエッジおよび立ち下がりエッジの検出を行い、そのときのタイマ値を内部レジスタに設定する機能。(インプットキャプチャ)
 - インプットキャプチャ発生時、タイマカウンタをクリアする機能。(カウンタクリア)
 - パルスの立ち上がりエッジおよび立ち下がりエッジ検出時、割り込み処理を起動する機能。

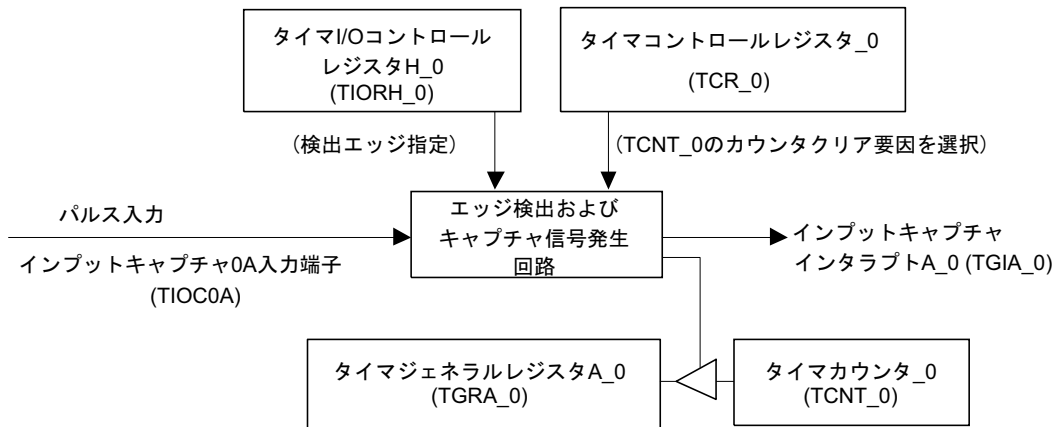


図 2.2 MTU/ch0 ブロック図

- (2) 表2.1に本タスク例の機能割り付けを示します。表に示すようにMTUの機能を割り付け、パルスのHigh幅およびLow幅を測定しています。

表 2.1 機能割り付け

端子、レジスタ名	機能割り付け
TCR_0	カウンタクリア要因の選択
TIORH_0	インプットキャプチャ信号の入力エッジを選択
TIOC0A	測定するパルスを入力
TGRA_0	パルスの立ち上がりおよび立ち下がり時のカウンタ値を検出
TGIA_0	パルスの立ち上がりおよび立ち下がり時、パルスの High および Low 幅測定を起動

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプットキャプチャ)
-----------------------	-----	-----------	------	------------------

動作説明

(1) 図2.3に動作原理を示します。図に示すようにSH7145のハードウェア処理およびソフトウェア処理によりパルスのHigh幅およびLow幅を測定します。

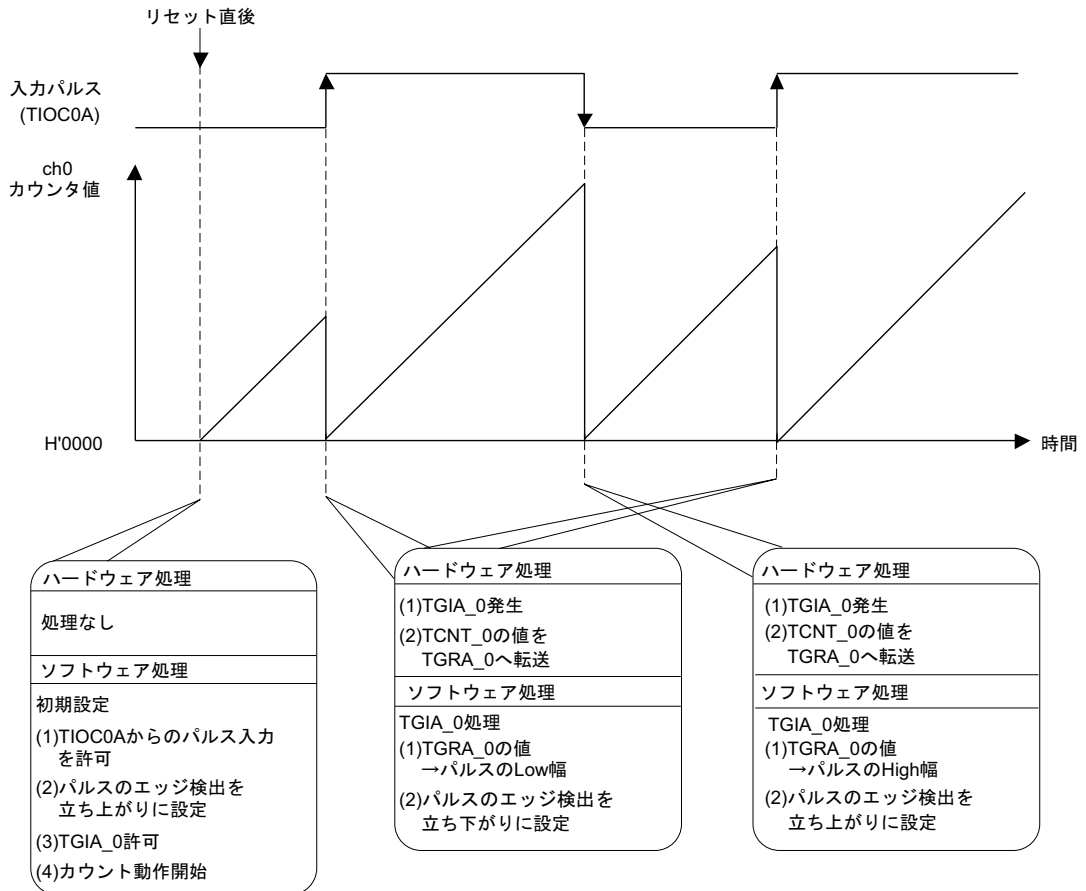


図 2.3 パルス幅測定動作原理

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプットキャプチャ)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	pwhlmn	MTU の初期設定
パルスの High 幅および Low 幅測定	pwhl1	TGIA_0 により起動し、TGRA_0 の値からパルスの High 幅および Low 幅を測定し、RAM に格納

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
pwh_hdata	パルスの High 幅に相当するタイマ値を設定 パルスの High 幅は以下の式にて算出 パルスの High 幅 (ns) = タイマ値 × φ 周期 (40.0MHz 動作時 25.0ns)	1 ワード	パルスの High 幅および Low 幅測定	出力
pwh_ldata	パルスの Low 幅に相当するタイマ値を設定 パルスの Low 幅は以下の式にて算出 パルスの Low 幅 (ns) = タイマ値 × φ 周期 (40.0MHz 動作時 25.0ns)	1 ワード		

(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
P_PORTE.PECRL2	PE0 を TIOC0A 入力端子に設定	H'FFFF83BA	H'0001
P_MTU0.TCR_0	TCNT_0 のカウンタクロックの選択、およびカウンタクリア要因を、TGRA_0 のインプットキャプチャで TCNT_0 をクリアに設定	H'FFFF8260	H'20
P_MTU0.TIORH_0	パルスの立ち上がりエッジおよび立ち下がりエッジ検出により、TCNT_0 の値を TGRA_0 へ転送を行うように設定	H'FFFF8262	H'08
P_MTU0.TIER_0	TGIA_0 による割り込み要求を許可	H'FFFF8264	H'41
P_MTU0.TGRA_0	パルスの立ち上がりおよび立ち下がり時の TCNT_0 の値が格納され、この値からパルスの周期を算出	H'FFFF8268	pwh_ldata pwh_hdata
P_INTC.IPRD	TGIA_0 の割り込み優先レベルを 15 に設定	H'FFFF834E	H'f000
P_STBY.MSTCR2	MTU モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd

(4) 使用 RAM 説明

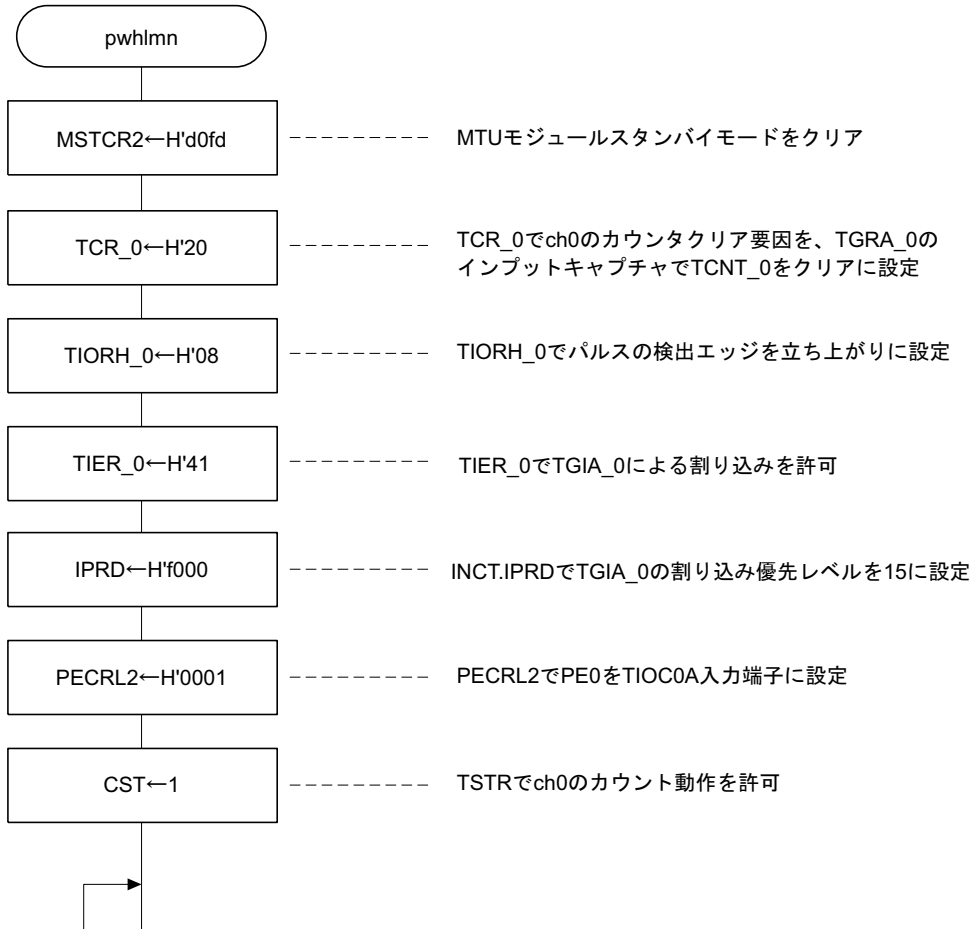
本タスク例では引数以外の RAM は使用していません。

【注】 レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプットキャプチャ)
-----------------------	-----	-----------	------	------------------

フローチャート

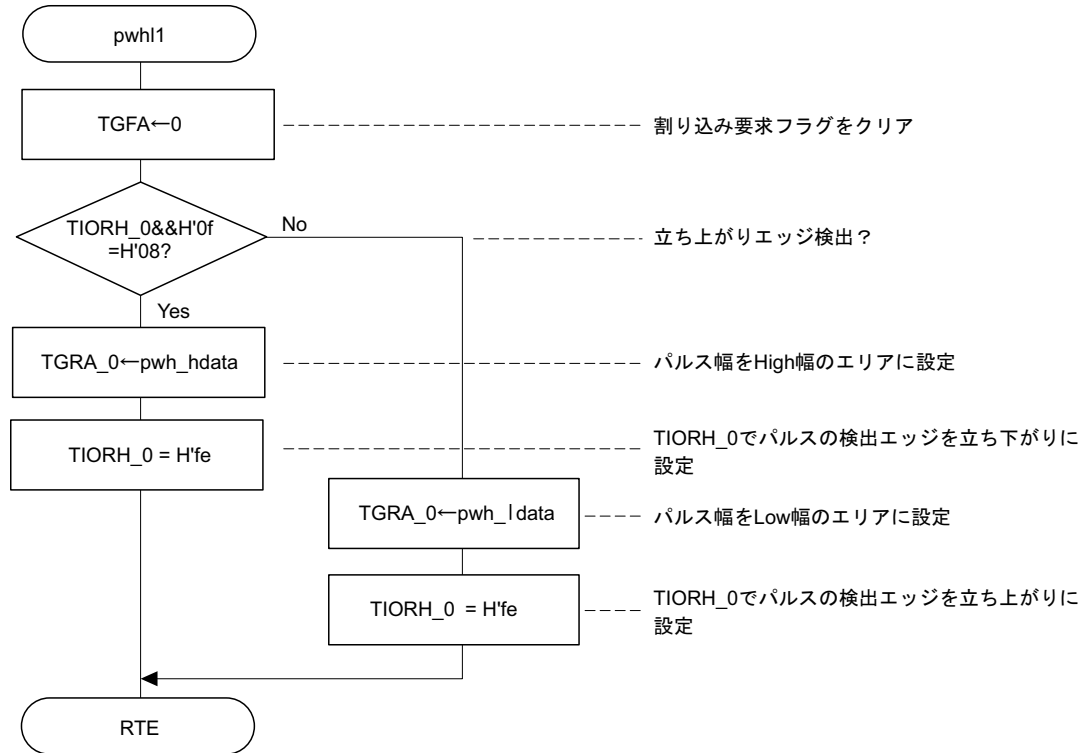
(1) メインルーチン



パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプットキャプチャ)
-----------------------	-----	-----------	------	------------------

フローチャート

(2) パルスの High および Low 幅測定



パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプット キャプチャ)
プログラムリスト				
<pre> /***** /* INCLUDE FILE */ /***** #include <machine.h> #include "iodefine_7145F.h" /***** /* PROTOTYPE */ /***** void pwhlmn (void) ; #pragma interrupt (pwhl1) /***** /* RAM ALLOCATION */ /***** #define pwh_hdata (* (unsigned short *) 0xffffe000) #define pwh_ldata (* (unsigned short *) 0xffffe002) /***** /* MAIN PROGRAM */ /***** void pwhlmn (void) { set_imask (0xf) ; P_STBY.MSTCR2.WORD = 0xd0fd; P_MTU0.TCR_0.BYTE = 0x20; /* timer clear input capture with TGRA_0 */ /* counter clock = Pφ/1 */ P_MTU0.TIORH_0.BYTE = 0x08; /* input capture by TIOC0A rising edge */ P_MTU0.TIER_0.BYTE = 0x41; /* enable TGIA interrupt */ P_INTC.IPRD.WORD = 0xf000; /* set initialize level = 15 */ P_PORTE.PECRL2.WORD = 0x0001; P_MTU34.TSTR.BIT.CST = 1; /* start TCNT_0 */ set_imask (0x0) ; while (1) ; } void pwhl1 () { P_MTU0.TSR_0.BIT.TGFA = 0; /* cleara interrupt flag */ if ((P_MTU0.TIORH_0.BYTE & 0x0f) == 0x08) { pwh_hdata = P_MTU0.TGRA_0.BYTE; /* set pwh */ </pre>				

パルスの High および Low 幅測定	MCU	SH7144/45	使用機能	MTU (インプット キャプチャ)
プログラムリスト				

```

        P_MTU0.TIORH_0 |= 0x01;      /* input capture falling edge TIOC0A */
    }
    else
    {
        pwh_ldata = P_MTU0.TGRA_0.BYTE; /* set pw1 */
        P_MTU0.TIORH_0 |= 0xfe;      /* input capture rising edge TIOC0A */
    }
}

```

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
仕様				

- (1) MTUのch0を使用して、図2.4に示すようにRAMに設定された周期のデューティ50%のパルスを出します。
- (2) 内蔵周辺クロック $P\phi=40.0\text{MHz}$ で動作時、出力するパルスの周期は50.0nsから1.63msの間で任意に設定できます。

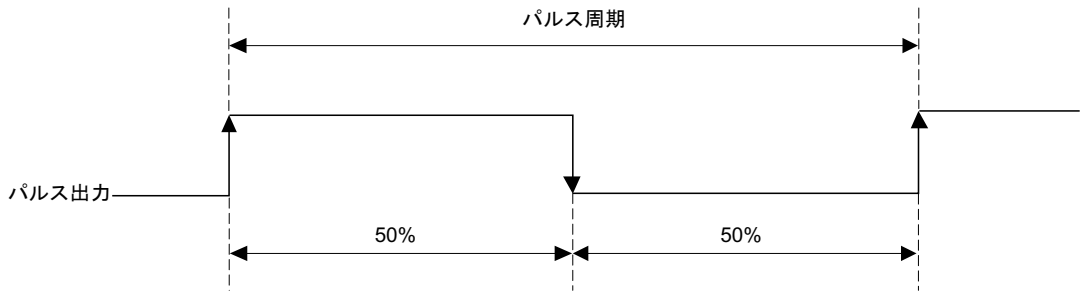


図 2.4 パルス出力

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
使用機能説明				

- (1) 本タスク例ではMTUのch0を使用し、デューティ50%のパルスを出力します。
- (a) 図2.5に本タスクで使用するMTU/ch0のブロック図を示します。
ch0では以下の機能を使用します。
- ソフトウェアを介さずハードウェアで自動的にパルスを出力する機能。(アウトプットコンペア)
 - コンペアマッチ時、カウンタをクリアする機能(カウンタクリア)
 - コンペアマッチが起きる毎に出力が反転する機能(トグル出力)

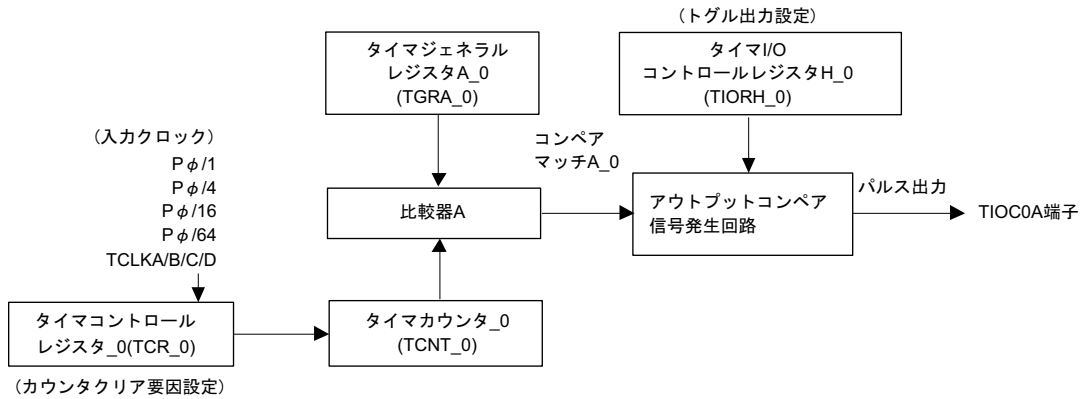


図 2.5 MTU/ch0 ブロック図

- (2) 表2.2に本タスク例の機能割り付けを示します。表に示すようにMTUの機能を割り付け、パルスを出力します。

表 2.2 機能割り付け

端子、レジスタ名	機能割り付け
TIOC0A	パルス出力端子
TCR_0	カウンタクリア要因および入カクロックの選択
TIORH_0	パルスの出力レベルを設定
TGRA_0	パルスの 1/2 周期を設定

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
動作説明				

(1) 図2.6に動作原理を示します。図に示すようにSH7145のハードウェア処理およびソフトウェア処理により、パルスを出力します。

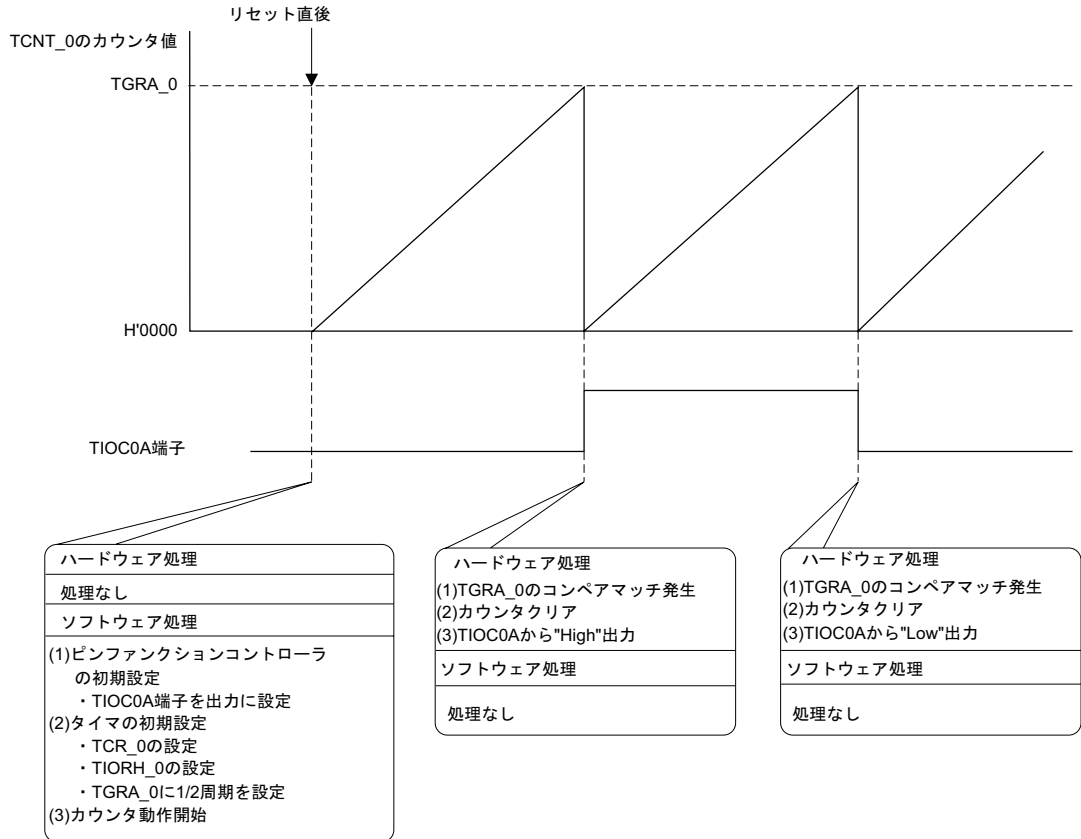


図 2.6 パルス出力動作原理

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	puls_out	PFC およびパルス出力設定

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
pul_cyc	パルスの 1/2 周期に相当するタイマ値を設定。 パルスの周期は以下の式にて算出 パルス周期 (ns) = タイマ値 × φ 周期 (40MHz 時 25ns)	1 ワード	メインルーチン	入力

(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
P_PORTE.PECRL2	PE0 を TIOC0A 出力に設定	H'FFFF83BA	H'0001
P_MTU0.TCR_0	カウンタクリア要因を TGRA0 のコンペアマッチに設定 入力ロックは Pφ/1 を設定	H'FFFF8260	H'20
P_MTU0.TIORH_0	TIOC0A は初期出力 0、アウトプットコンペアでトグル出力	H'FFFF8262	H'03
P_MTU0.TGRA_0	出力パルスの 1/2 周期を設定	H'FFFF8268	pul_cyc
P_MTU0.TMDR_0	ch0 は通常モードに設定	H'FFFF8261	H'c0
P_STBY.MSTCR2	MTU モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd

(4) 使用 RAM 説明

本アプリケーション例では引数以外の RAM は使用していません。

注) レジスタのラベル名は、SH7145 ヘッダファイルの名前を使用しています

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
フローチャート				
(1) メインルーチン				
<pre> graph TD Start([puls_out]) --> MSTCR2[MSTCR2 ← H'd0fd] MSTCR2 --> PEIORL[PEIORL ← H'0001 PECRL2 ← H'0001] PEIORL --> TCR_0[TCR_0 ← H'20] TCR_0 --> TIORH_0[TIORH_0 ← H'03] TIORH_0 --> TGRA_0[TGRA_0 ← pul_cyc] TGRA_0 --> TMDR_0[TMDR_0 ← H'c0] TMDR_0 --> CST[CST ← 1] CST --> Loop[] Loop --> Start </pre>				
				MTUモジュールスタンバイモードのクリア
				PE0をTIOC0A出力に設定
				P_MTU0.TCR_0でカウンタクリア要因を設定
				P_MTU0.TIORH_0によりコンペアマッチでトグル出力に設定
				P_MTU0.TGRA_0にパルス周期の1/2を設定
				P_MTU0.TMDR_0でモードを通常動作に設定
				P_MTU34.TSTR.CSTでMTU/ch0のカウンタ動作を許可

パルス出力	MCU	SH7144/45	使用機能	MTU (アウトプットコンペア)
プログラムリスト	<pre> /***** / * INCLUDE FILE * /***** #include<machine.h> #include"iodefine_7145F.h" /***** / * PROTOTYPE * /***** void puls_out (void) ; /***** / * RAM ALLOCATION * /***** #define pul_cyc (* (unsigned short *) 0xffffe000) /***** / * MAIN PROGRAM * /***** void puls_out (void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* MTU module standby mode clear */ P_PORTE.PEIORL.WORD = 0x0001; /* TIIOC0A = output */ P_PORTE.PECRL2.WORD = 0x0001; /* PE0 function = TIIOC0A */ P_MTU0.TCR_0.BYTE = 0x20; /* Counter clear by TGRA_0 */ P_MTU0.TIORH_0.BYTE = 0x03; /* toggle output */ P_MTU0.TGRA_0 = pul_cyc; /* set 1/2 period */ P_MTU0.TCNT_0 = 0x0000; /* Clear timer counter */ P_MTU0.TMDR_0.BYTE = 0xc0; /* Set operation mode */ P_MTU34.TSTR.CST.BIT = 1; /* Start timer counter */ while (1) ; } </pre>			

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
仕様				

- (1) MTUのPWMモード1を使用して、設定されたデューティおよび周期を基に4相のPWM出力をします。
- (2) PWMモード1は各チャンネルで任意の周期が設定できます。ch0、ch3、ch4は各2本ずつ、ch1、ch2は各1本ずつの出力が可能です。よってch0、ch3、ch4では同一周期の中でHigh幅の異なる波形の生成が可能です。
- (3) デューティは0%~100%まで1/65535の分解能で設定できます。

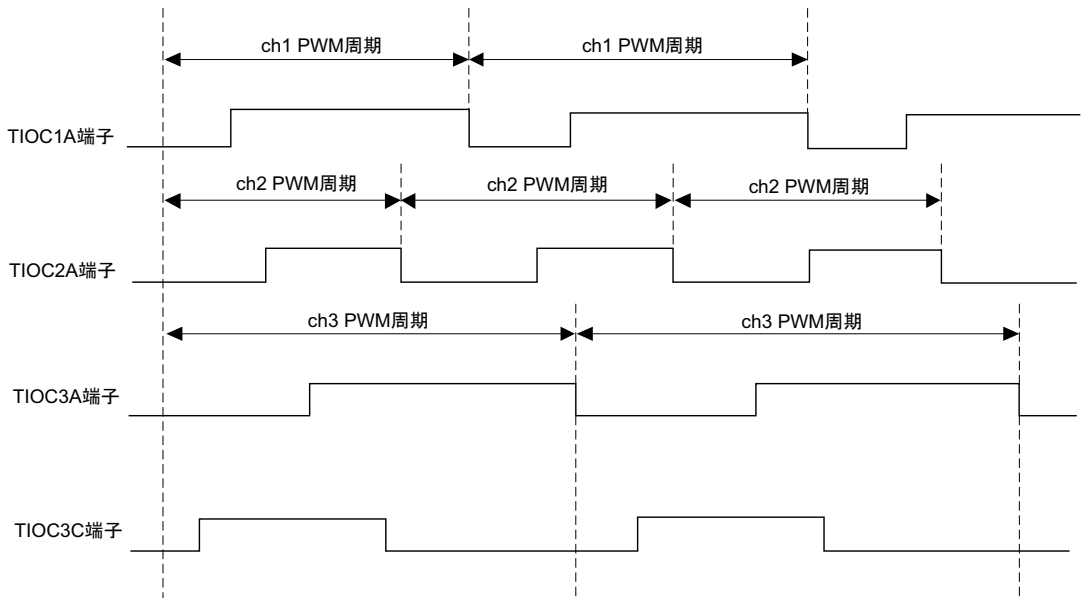
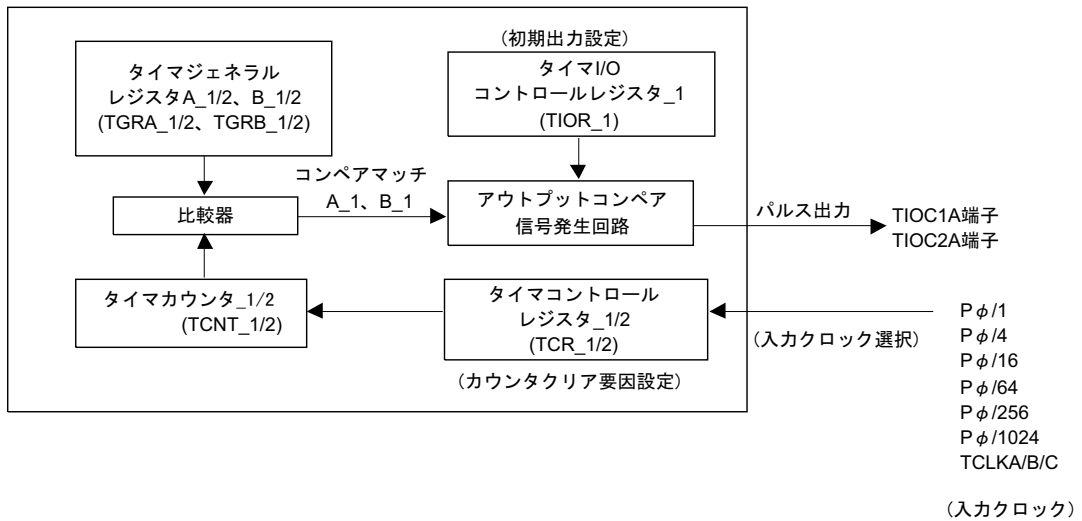


図 2.7 PWM 出力例

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
使用機能説明				

- (1) 本タスク例ではMTUのch1~ch3を使用し、4相のPWM出力を行います。
- PWM モード 1 では、TGRA と TGRB、TGRC と TGRD をそれぞれペアで使用して、PWM 出力を生成します。
また、ch0~ch4まで使用することで、最大8相のPWM出力が可能です。
- (a) 図2.8に本タスク例で使用するMTUのブロック図を示します。

ch1、2



ch3

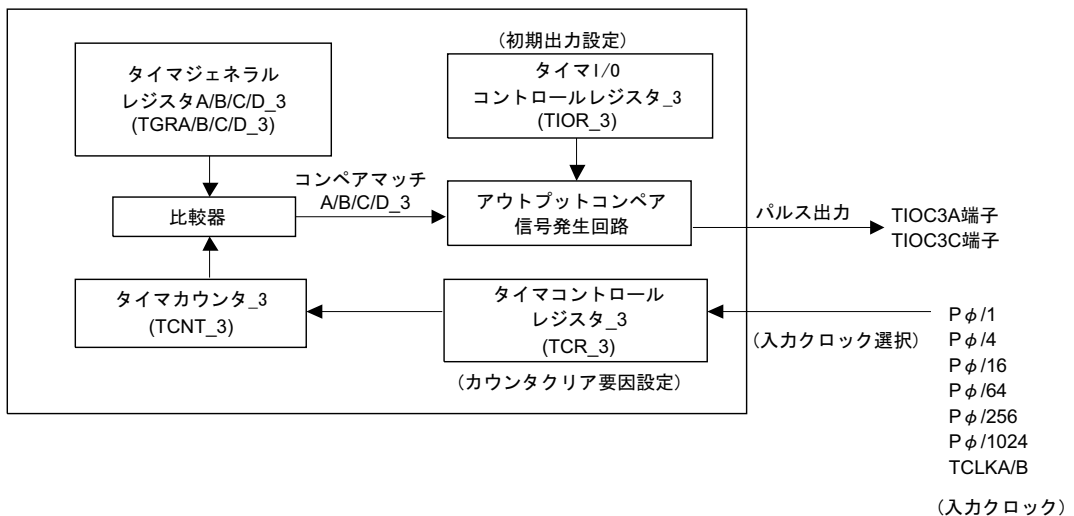


図 2.8 MTU/ch1、2、3 ブロック図

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
使用機能説明				

- (2) 表2.3に本タスクの機能割り付けを示します。表に示すようにMTUの機能を割り付け、PWMパルスを出力します。

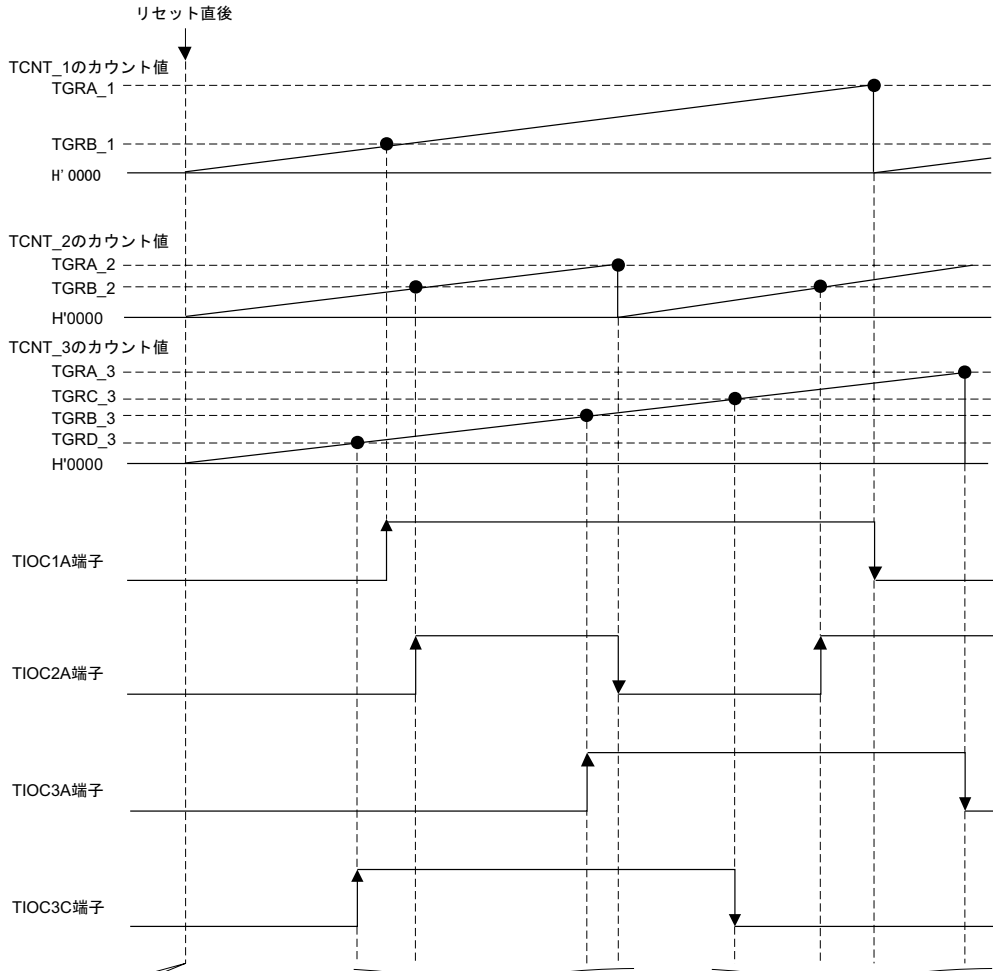
表 2.3 機能割り付け

端子、レジスタ名	機能割り付け
TIOC1A TIOC2A TIOC3A TIOC3C	PWM パルス出力端子
TCR_1 TCR_2 TCR_3	ch1～ch3 のタイマカウンタのクリア要因と入カクロックを選択
TMDR_1 TMDR_2 TMDR_3	ch1～ch3 を PWM モード 1 として動作
TGRA_1 TGRA_2 TGRA_3	PWM 周期の設定
TGRB_1 TGRB_2 TGRB_3 TGRC_3 TGRD_3	デューティの設定

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
動作説明				

動作説明

- (1) 図2.9に動作原理を示します。図に示すようにSH7145のハードウェア処理およびソフトウェア処理により、ch1～ch3の各PWM出力端子 (TIOC1A、TIOC2A、TIOC3A/C) から4相のPWM出力を行います。



ハードウェア処理
なし
ソフトウェア処理
(1)ピンファンクションコントローラの初期設定
・ TIOC3C、TIOC3A端子を出力に設定
・ TIOC2A、TIOC1A端子を出力に設定
(2)タイマの初期設定
・ TCR_1/2/3の設定
・ TGRA/B_1、TGRA/B_2、TGRA/B/C/D_3の設定
・ TIORH_1/2/3、TIORL_1/2/3の設定
(3)カウント動作開始

ハードウェア処理
コンペアマッチの発生で、PWM出力端子の出力レベルを"High"に設定
ソフトウェア処理
なし

ハードウェア処理
コンペアマッチAで、PWM出力端子の出力レベルを"Low"に変更
ソフトウェア処理
なし

図 2.9 PWM 波形の動作原理

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	pwm_1	PFC および PWM 出力の設定

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
pul_cyc1 pul_cyc2 pul_cyc3	パルスの周期に相当するタイム値を設定 パルスの周期は以下の式にて算出 パルス周期 (ns) = タイマ値 × φ 周期 (40MHz 動作時 25ns)	1 ワード	メインルーチン	入力
pul_duty1b pul_duty2b pul_duty3b pul_duty3c pul_duty3d	TIOC 端子から出力される波形変化タイミングを設定			

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
ソフトウェア説明				

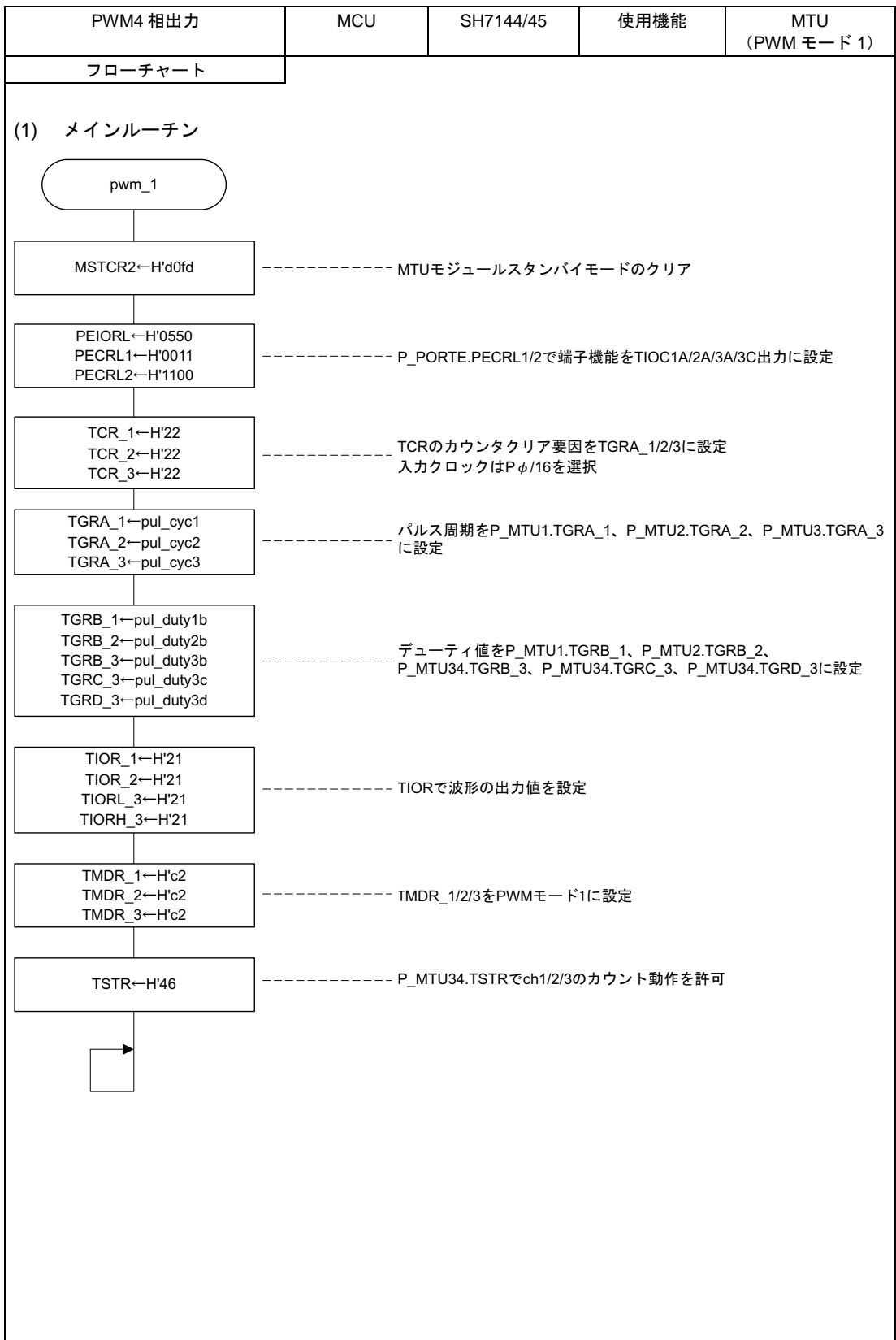
(3) 使用内部レジスタ説明

レジスタ名	機能割り付け	アドレス	設定値
P_STBY.MSTCR2	MTU モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd
P_MTU1.TCR_1 P_MTU2.TCR_2 P_MTU3.TCR_3	タイマカウンタのクリア要因を TGRA_1、TGRA_2、TGRA_3 のコンペアマッチでクリア 入カロックは Pφ/16 を選択	H'FFFF8280 H'FFFF82A0 H'FFFF8200	H'22 H'22 H'22
P_MTU1.TGRA_1	チャンネル 1 の PWM 周期を設定	H'FFFF8288	pul_cyc1
P_MTU1.TGRB_1	TIOC1A から "High" 出力させるタイマカウンタ値を設定	H'FFFF828A	pul_duty1b
P_MTU2.TGRA_2	チャンネル 2 の PWM 周期を設定	H'FFFF82A8	pul_cyc2
P_MTU2.TGRB_2	TIOC2A から "High" 出力させるタイマカウンタ値を設定	H'FFFF82AA	pul_duty2b
P_MTU34.TGRA_3	チャンネル 3 の PWM 周期を設定	H'FFFF8218	pul_cyc3
P_MTU34.TGRB_3	TIOC3A から "High" 出力させるタイマカウンタ値を設定	H'FFFF821A	pul_duty3b
P_MTU34.TGRC_3	TIOC3C から "Low" 出力させるタイマカウンタ値を設定	H'FFFF8224	pul_duty3c
P_MTU34.TGRD_3	TIOC3C から "High" 出力させるタイマカウンタ値を設定	H'FFFF8226	pul_duty3d
P_MTU1.TIOR_1	TGRA_1 は初期出力 0 でアウトプットコンペアで 0 出力 TGRB_1 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8282	H'21
P_MTU2.TIOR_2	TGRA_2 は初期出力 0 でアウトプットコンペアで 0 出力 TGRB_2 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF82A2	H'21
P_MTU34.TIORH_3	TGRA_3 は初期出力 0 でアウトプットコンペアで 0 出力 TGRB_3 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8204	H'21
P_MTU34.TIORL_3	TGRC_3 は初期出力 0 でアウトプットコンペアで 0 出力 TGRD_3 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8205	H'21
P_MTU1.TMDR_1 P_MTU2.TMDR_2 P_MTU34.TMDR_3	動作モードを PWM モード 1 に設定	H'FFFF8281 H'FFFF82A1 H'FFFF8202	H'c2 H'c2 H'c2

(4) 使用 RAM 説明

本アプリケーション例では引数以外の RAM は使用していません

注) レジスタのラベル名は、SH7145 ヘッダファイルの名前を使用しています。



PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
プログラムリスト				
<pre> /***** / / /***** #include<machine.h> #include"iodefine_7145F.h" /***** / /***** PROTOTYPE /***** void pwm_1 (void) ; /***** / /***** RAM ALLOCATION /***** #define pul_cyc1 (* (unsigned short *) 0xffffe000) #define pul_duty1b (* (unsigned short *) 0xffffe002) #define pul_cyc2 (* (unsigned short *) 0xffffe004) #define pul_duty2b (* (unsigned short *) 0xffffe006) #define pul_cyc3 (* (unsigned short *) 0xffffe008) #define pul_duty3b (* (unsigned short *) 0xffffe00a) #define pul_duty3c (* (unsigned short *) 0xffffe00c) #define pul_duty3d (* (unsigned short *) 0xffffe00e) /***** / /***** MAIN PROGRAM /***** / void pwm_1 (void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* Clear module standby mode */ P_PORTE.PEIORL.WORD = 0x0550; /* TIOC1A/2A/3A/3C = output */ P_PORTE.PECRL1.WORD = 0x0011; P_PORTE.PECRL2.WORD = 0x1100; P_MTU1.TCR_1.BYTE = 0x22; /* Counter1 clear by TGRA_1 */ P_MTU2.TCR_2.BYTE = 0x22; /* Counter2 clear by TGRA_2 */ P_MTU34.TCR_3.BYTE = 0x22; /* Counter3 clear by TGRA_3 */ P_MTU1.TGRA_1 = pul_cyc1; /* set TIOC1A period */ P_MTU2.TGRA_2 = pul_cyc2; /* set TIOC2A period */ P_MTU34.TGRA_3 = pul_cyc3; /* set TIOC3A period */ </pre>				

PWM4 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 1)
プログラムリスト	<pre> P_MTU1.TGRB_1 = pul_duty1b; /* set TIOC1A duty */ P_MTU2.TGRB_2 = pul_duty2b; /* set TIOC2A duty */ P_MTU34.TGRB_3 = pul_duty3b; /* set TIOC3A duty */ P_MTU34.TGRC_3 = pul_duty3c; /* set TIOC3C duty */ P_MTU34.TGRD_3 = pul_duty3d; /* set TIOC3C duty */ P_MTU1.TIOR_1.BYTE = 0x21; /* start"0",compare match"1"output*/ P_MTU2.TIOR_2.BYTE = 0x21; /* start"0",compare match"1"output*/ P_MTU34.TIORL_3.BYTE = 0x21; /* start"0",compare match"1"output*/ P_MTU34.TIORH_3.BYTE = 0x21; /* start"0",compare match"1"output*/ P_MTU1.TMDR_1.BYTE = 0xc2; /* PWM model */ P_MTU2.TMDR_2.BYTE = 0xc2; /* PWM model */ P_MTU34.TMDR_3.BYTE = 0xc2; /* PWM model */ P_MTU34.TSTR.BYTE = 0x46; /* Timer counter start */ while (1) ; } </pre>			

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
仕様				

- (1) 図2.10に示すようにパルスのHigh幅を変化させ、デューティを変化することができる7相のPWM出力をします。
- (2) 内蔵周辺クロック $P\phi=40.0\text{MHz}$ 動作時、出力するPWM周期は50nsから1.63msの間で任意に設定できます。

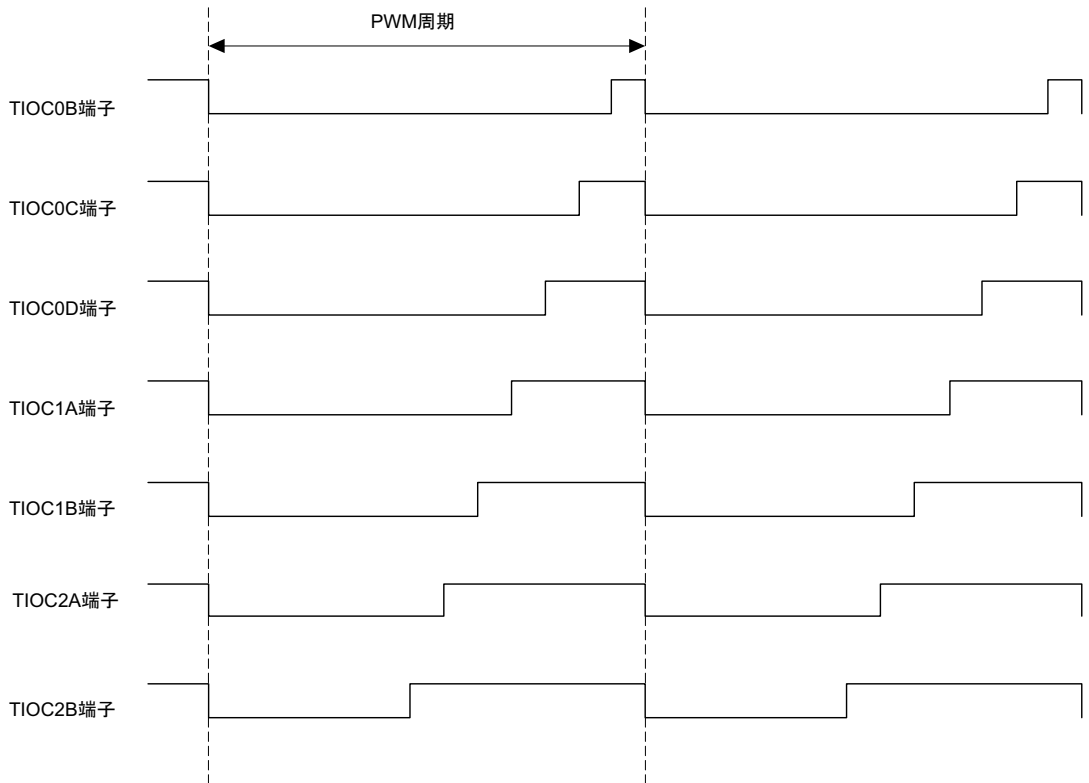


図 2.10 7相 PWM 波形出力例

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWMモード2)
使用機能説明				

- (1) 本タスク例ではMTUのch0～ch2までを同期動作させ、7相のPWM出力をします。
- (a) 図2.11に本タスク例で使用するMTUのブロック図を示します。
本タスク例では、MTUの以下の機能を使用しています。
- ソフトウェアを介さず、ハードウェアで自動的にパルスを出力する機能。(アウトプットコンペア)
 - コンペアマッチ時カウンタをクリアする機能。(カウンタクリア)
 - コンペアマッチ発生毎に出力が反転する機能。(トグル出力)

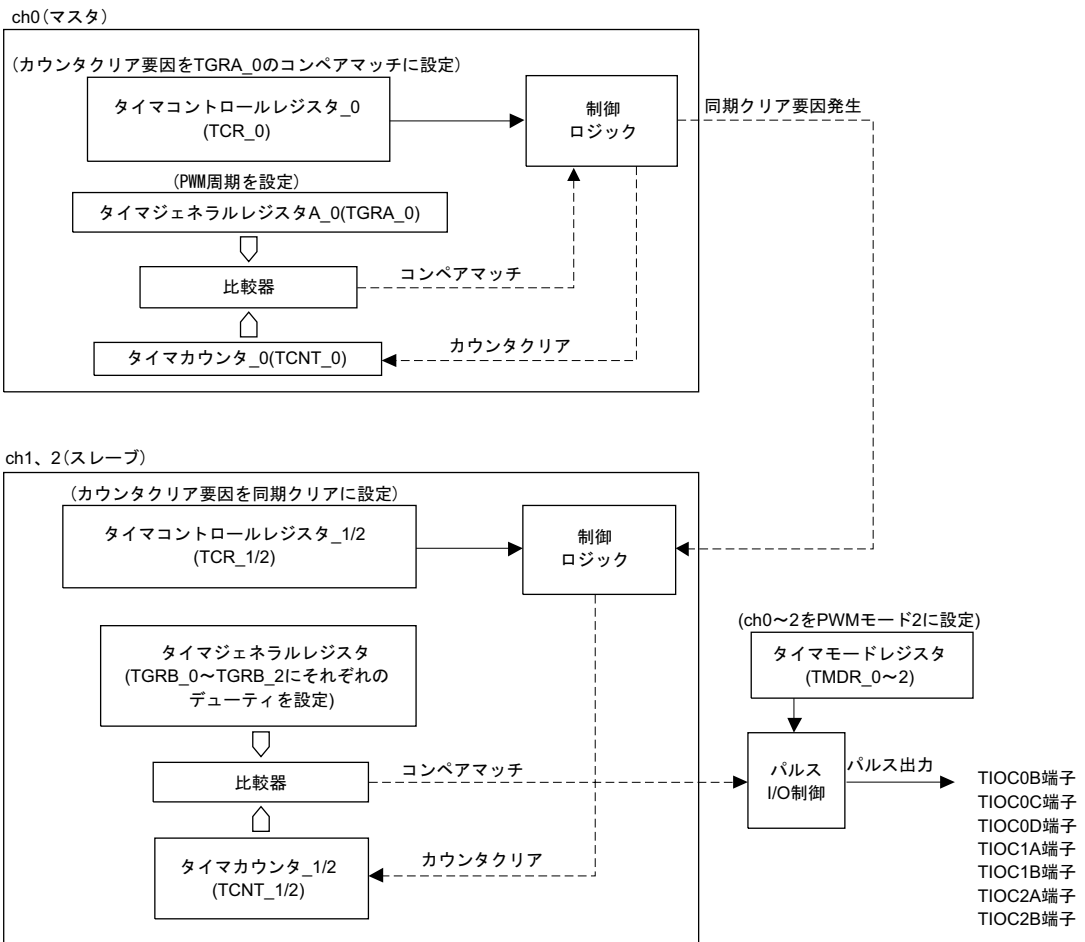


図 2.11 同期クリアブロック図

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
使用機能説明				

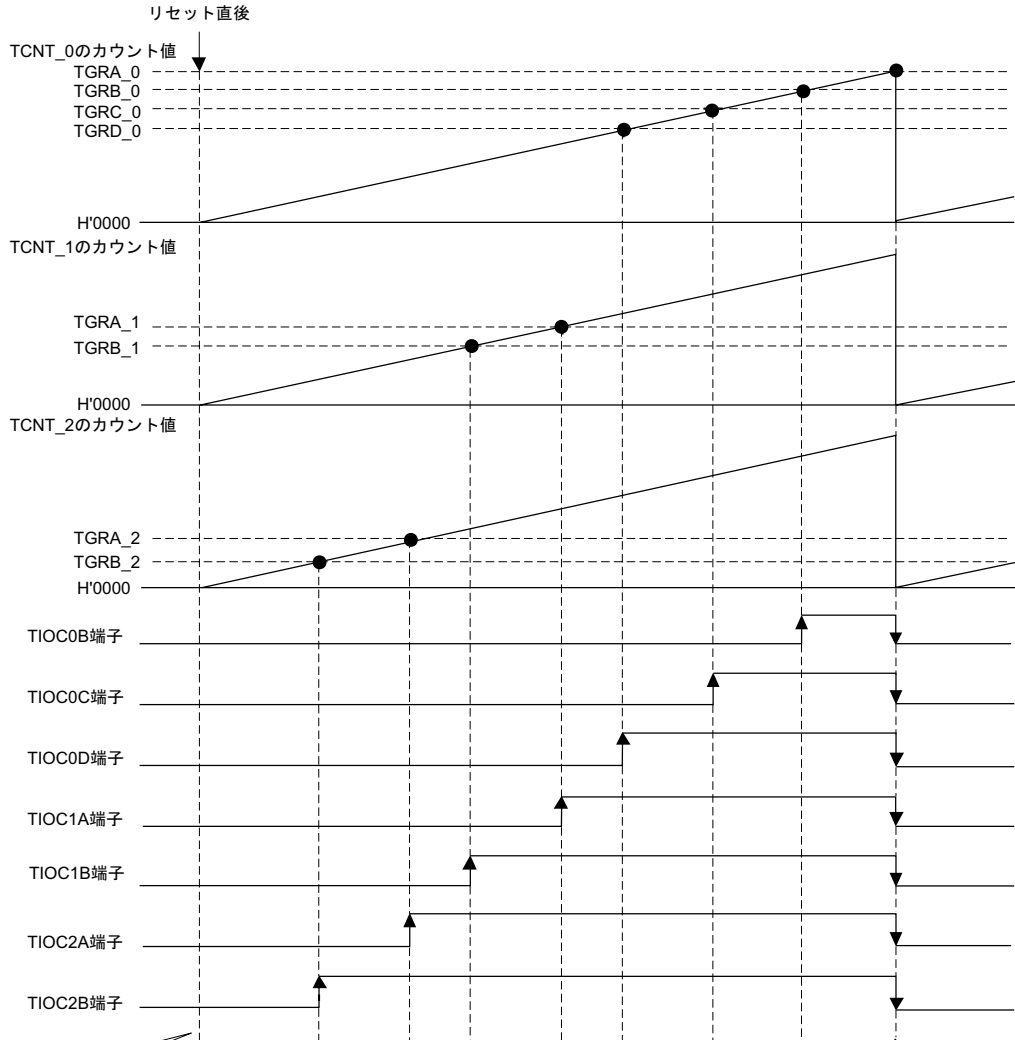
- (2) 表2.4に本タスクの機能割り付けを示します。表に示すようにMTUの機能を割り付け、PWMパルスを出力します。

表 2.4 MTU 機能割り付け

端子、レジスタ名	機能割り付け
TIOC0B TIOC0C TIOC0D TIOC1A TIOC1B TIOC2A TIOC2B	PWM パルス出力端子
TSYR	ch0/1/2 は同期動作
TCR_0/1/2	ch0/1/2 のタイマカウンタのクリア要因と入カクロックを選択
TGRA_0	PWM 周期の設定
TGRB_0 TGRC_0 TGRD_0 TGRA_1 TGRB_1 TGRA_2 TGRB_2	デューティの設定
TMDR_0/1/2	ch0/1/2 を PWM モード 2 として動作

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
動作説明				

(1) 図2.11に動作原理を示します。図に示すようにSH7145のハードウェア処理およびソフトウェア処理により、ch0/1/2の各PWM出力端子（TIOC0B/C/D、TIOC1A/B、TIOC2A/B）から7相のPWM出力をします。



ハードウェア処理
なし
ソフトウェア処理
(1)ピンファンクションコントローラの初期設定 ・端子機能をTIOC出力機能に設定
(2)タイマの初期設定 ・TCR_0/1/2の設定 ・TGRA/B/C/D_0、TGRA/B_1、TGRA/B_2の設定 ・TIORH/L_0、TIOR_1/2の設定 ・TSYRの設定
(3)カウンタ動作の開始

ハードウェア処理
コンペアマッチの発生で PWM出力端子の出力レベルを "High"に変更
ソフトウェア処理
なし

ハードウェア処理
(1)TGRA_0のコンペアマッチでPWM出力 端子の出力レベルを"Low"に変更 (2)ch0/1/2のタイマカウンタをクリア
ソフトウェア処理
なし

図 2.11 ノコギリ波形の生成に用いる PWM 出力 (7 相) の動作原理

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	pwm_2	PFC および PWM 出力の設定

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
pul_cyc0a	パルスの周期に相当するタイマ値を設定 パルスの周期は以下の式にて算出 パルス周期 (ns) = タイマ値 × φ 周期 (40.0MHz 動作時 25ns)	1 ワード	メインルーチン	入力
pul_duty0b pul_duty0c pul_duty0d pul_duty1a pul_duty1b pul_duty2a pul_duty2b	TIOC 端子から出力される波形変化タイミングを設定			

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
ソフトウェア説明				

(3) 使用内部レジスタ説明

レジスタ名	機能割り付け	アドレス	設定値
P_STBY.MSTCR2	モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd
P_PORTE.PECRL2	端子機能を TIOC0B/C/D、TIOC1A/B、TIOC2A/B 出力に設定	H'FFFF83BA	H'5554
P_MTU34.TSYR	タイマカウンタ 0/1/2 を同期動作に設定	H'FFFF8241	H'07
P_MTU0.TCR_0	タイマカウンタのクリア要因を TGRA_0 のコンペアマッチでクリアに設定	H'FFFF8260	H'22
P_MTU1.TCR_1	コンペアマッチでクリアに設定	H'FFFF8280	H'62
P_MTU2.TCR_2	入力クロックは Pφ/16 を選択	H'FFFF82A0	H'62
P_MTU0.TGRA_0	PWM 周期を設定	H'FFFF8268	pul_cyc0
P_MTU0.TGRB_0	TIOC0B から "High" を出力させるタイマカウンタ値を設定	H'FFFF826A	pul_duty0b
P_MTU0.TGRC_0	TIOC0C から "High" を出力させるタイマカウンタ値を設定	H'FFFF826C	pul_duty0c
P_MTU0.TGRD_0	TIOC0D から "High" を出力させるタイマカウンタ値を設定	H'FFFF826E	pul_duty0d
P_MTU1.TGRA_1	TIOC1A から "High" を出力させるタイマカウンタ値を設定	H'FFFF8288	pul_duty1a
P_MTU1.TGRB_1	TIOC1B から "High" を出力させるタイマカウンタ値を設定	H'FFFF828A	pul_duty1b
P_MTU2.TGRA_2	TIOC2A から "High" を出力させるタイマカウンタ値を設定	H'FFFF82A8	pul_duty2a
P_MTU2.TGRB_2	TIOC2B から "High" を出力させるタイマカウンタ値を設定	H'FFFF82AA	pul_duty2b
P_MTU0.TIORH_0	TGRA_0 は初期出力 0 でアウトプットコンペアで 0 出力、TGRB_0 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8262	H'21
P_MTU0.TIORL_0	TGRC_0 は初期出力 0 でアウトプットコンペアで 1 出力、TGRD_0 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8263	H'22
P_MTU1.TIOR_1	TGRA_1 は初期出力 0 でアウトプットコンペアで 1 出力、TGRB_1 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF8282	H'22
P_MTU1.TIOR_2	TGRA_2 は初期出力 0 でアウトプットコンペアで 1 出力、TGRB_2 は初期出力 0 でアウトプットコンペアで 1 出力に設定	H'FFFF82A2	H'22
P_MTU0.TMDR_0	各チャネルの動作モードを PWM モード 2 に設定	H'FFFF8261	H'c3
P_MTU1.TMDR_1		H'FFFF8281	H'c3
P_MTU2.TMDR_2		H'FFFF82A1	H'c3

(4) 使用 RAM 説明

本アプリケーションで例では引数以外の RAM は使用していません。

注) レジスタのラベル名は、SH7145 のヘッダファイルの名前を使用しています。

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWMモード2)
フローチャート				
(1) メインルーチン				
			MTUモジュールスタンバイモードのクリア	
			P_PORTE.PECRL2で端子機能をTIOC0B/C/D、TIOC1A/B、TIOC2A/B出力端子に設定	
			P_MTU0.TCR_0でカウンタクリア要因をTGRA_0に設定し、P_MTU1.TCR_1、P_MTU2.TCR_2で同期クリアに設定	
			P_MTU0.TGRA_0にパルス周期を設定	
			P_MTU0.TGRB/C/D_0、P_MTU1.TGRA/B_1、P_MTU2.TGRA/B_2にデューティを設定	
			TIORでTGRの機能を設定	
			TMDR_0/1/2でPWMモード2に設定	
			ch0/1/2を同期動作に設定	
			P_MTU34.TSTRでch0/1/2のカウンタ動作を許可	

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード2)
プログラムリスト	<pre> /***** / * INCLUDE FILE * / /***** #include<machine.h> #include"iodefine_7145F.h" /***** / * PROTOTYPE * / /***** void pwm_2 (void) ; /***** / * RAM ALLOCATION * / /***** #define pul_cyc0 (* (unsigned short *) 0xffffe00) #define pul_duty0b (* (unsigned short *) 0xffffe02) #define pul_duty0c (* (unsigned short *) 0xffffe04) #define pul_duty0d (* (unsigned short *) 0xffffe06) #define pul_duty1a (* (unsigned short *) 0xffffe08) #define pul_duty1b (* (unsigned short *) 0xffffe0a) #define pul_duty2a (* (unsigned short *) 0xffffe0c) #define pul_duty2b (* (unsigned short *) 0xffffe0e) /***** / * MAIN PROGRAM * / /***** void pwm_2 (void) { P_STBY.MSTCR2.WORD = 0xd0fd; /*Clear module standby mode*/ P_PORTE.PEIORL.WORD = 0x00fe; /*TIOC0B/C/D,TIOC1A/B,TIOC2A/B output*/ P_PORTE.PECRL2.WORD = 0x5554; P_MTU0.TCR_0.BYTE = 0x22; /*Counter clear by TGRA_0 */ P_MTU1.TCR_1.BYTE = 0x62; /*Counter clear by TGRA_0 */ P_MTU2.TCR_2.BYTE = 0x62; /*Counter clear by TGRA_0 */ P_MTU0.TGRA_0 = pul_cyc0; /*Set PWM period */ P_MTU0.TGRB_0 = pul_duty0b; /*Set PWM duty*/ P_MTU0.TGRC_0 = pul_duty0c; P_MTU0.TGRD_0 = pul_duty0d; P_MTU1.TGRA_1 = pul_duty1a; </pre>			

PWM7 相出力	MCU	SH7144/45	使用機能	MTU (PWM モード 2)
プログラムリスト	<pre> P_MTU1.TGRB_1 = pul_duty1b; P_MTU2.TGRA_2 = pul_duty2a; P_MTU2.TGRB_2 = pul_duty2b; P_MTU0.TIORH_0.BYTE = 0x21; /*TIOC0B=start"0",compare match"1"output*/ P_MTU0.TIORL_0.BYTE = 0x22; /*TIOC0B/0C=start"0",compare match"1"output*/ P_MTU1.TIOR_1.BYTE = 0x22; /*TIOC1A/1B=start"0",compare match"1"output*/ P_MTU2.TIOR_2.BYTE = 0x22; /*TIOC2A/2B=start"0",compare match"1"output*/ P_MTU0.TMDR_0.BYTE = 0xc3; /*Set PWM mode2*/ P_MTU1.TMDR_1.BYTE = 0xc3; /*Set PWM mode2*/ P_MTU2.TMDR_2.BYTE = 0xc3; /*Set PWM mode2*/ P_MTU34.TSYR.BYTE = 0x07; /*Synchronize ch0,ch1,ch2*/ P_MTU34.TSTR.BYTE = 0x07; /*Start timer counter */ while (1) ; } </pre>			

正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
仕様				

- (1) 図2.12に示すようにパルスのHigh幅を変化させ、デューティを変化できるパルス（デューティパルス）を正相と逆相で3相出力します。
- (2) 内蔵周辺クロックPφ=40.0MHz動作時、出力するパルスの周期は25.0nsから1.63msの間で任意に設定できます。

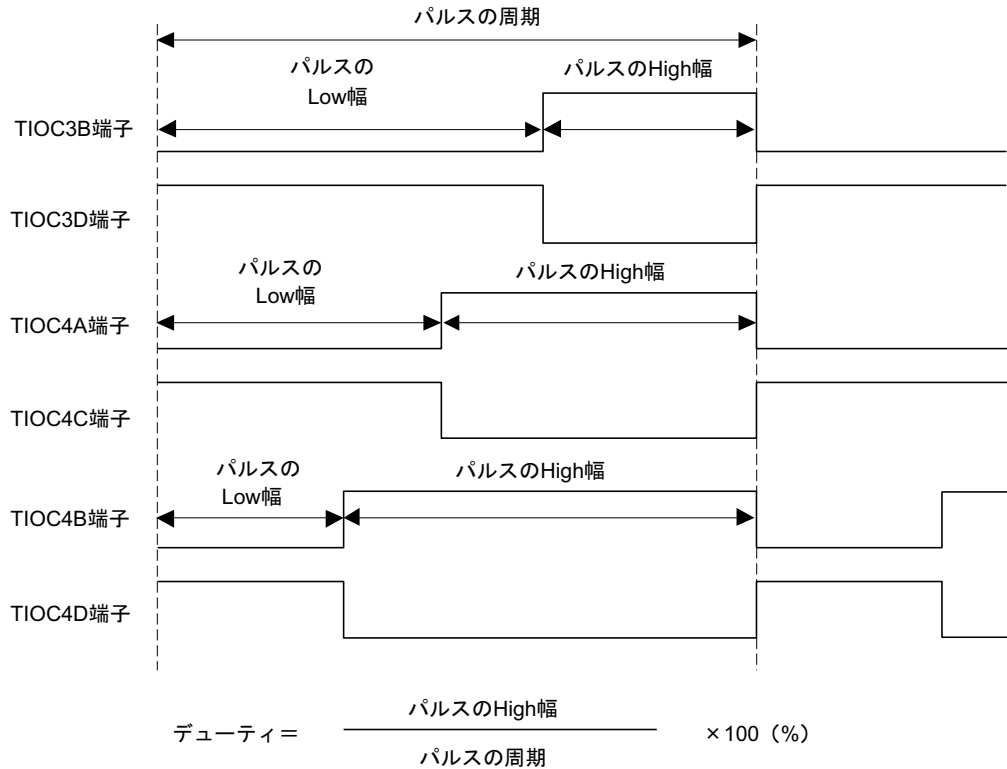


図 2.12 正相・逆相 PWM3 相出力波形

正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
使用機能説明				

- (1) 本タスク例ではMTUのch3、ch4を組み合わせて使用し、一方の変化点が共通で正相、逆相の関係にあるPWM波形を3相出力します。
- リセット同期 PWM モードでは、TGRA と TGRC、TGRB と TGRD をそれぞれペアでバッファ動作して PWM 波形を生成します。
- (a) 図2.13に本タスク例で使用するMTUのブロック図を示します。

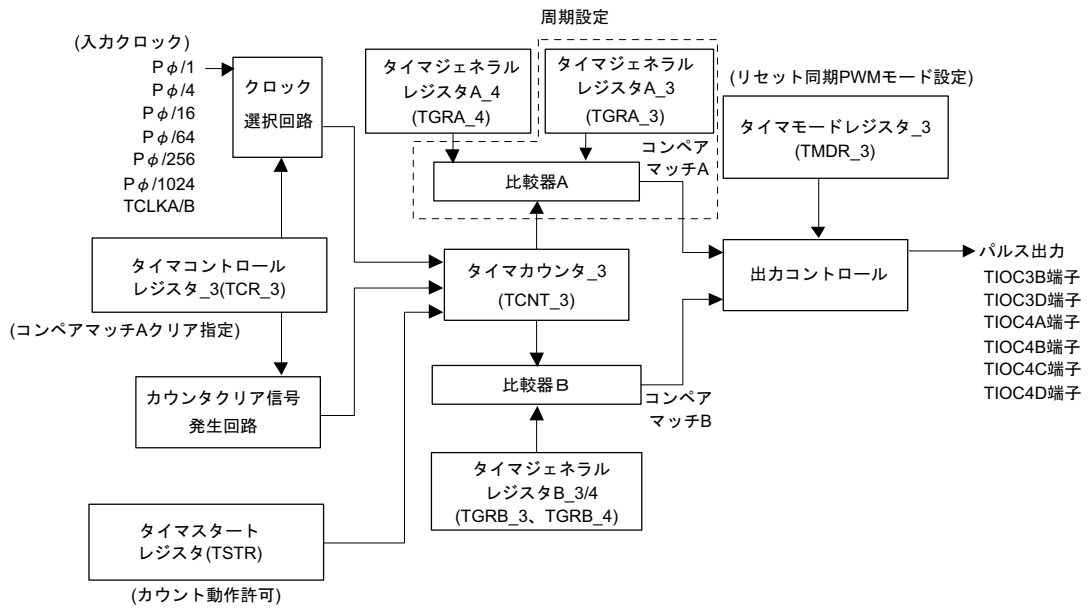


図 2.13 MTU/ch3、4 ブロック図

- (2) 表2.5に本タスクの機能割り付けを示します。表に示すようにMTUの機能を割り付け、PWMパルスを出力します。

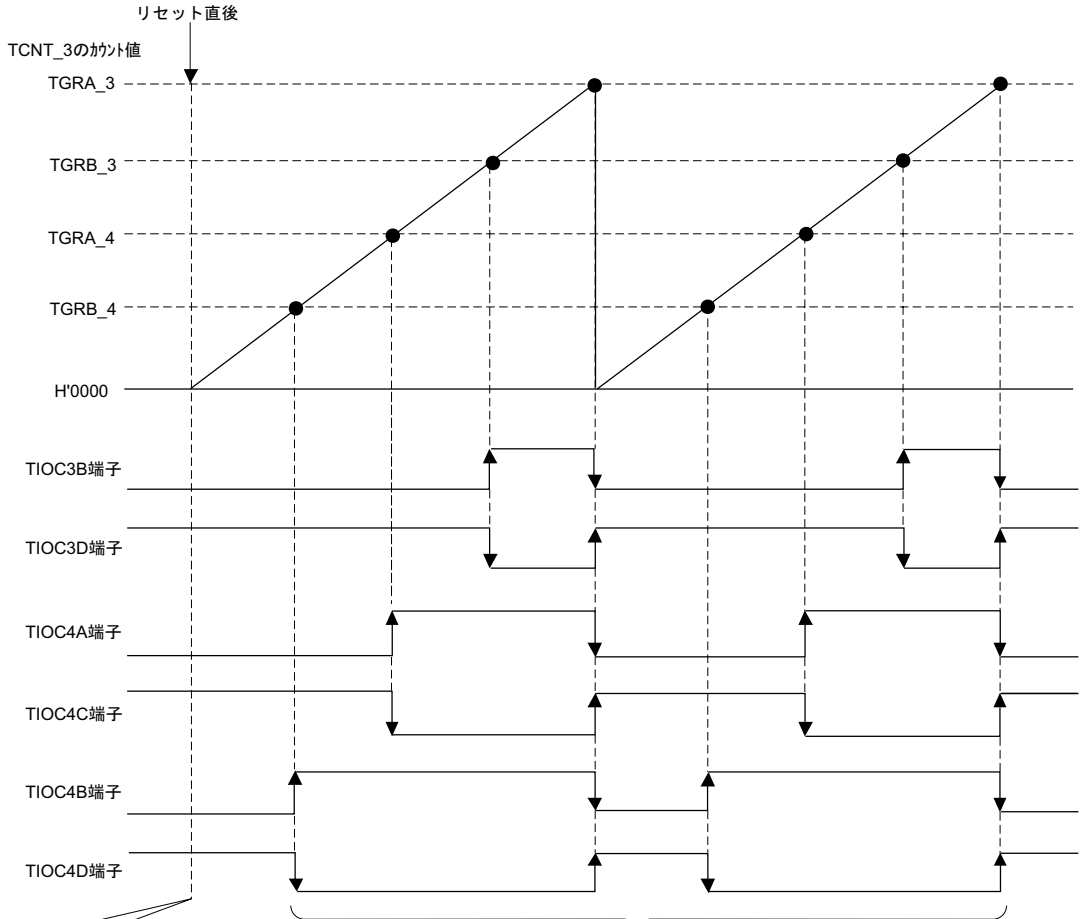
表 2.5 機能割り付け

端子、レジスタ名	機能割り付け
TIOC3B	PWM 出力 1
TIOC3D	PWM 出力 1 の逆相波形
TIOC4A	PWM 出力 2
TIOC4B	PWM 出力 3
TIOC4C	PWM 出力 2 の逆相波形
TIOC4D	PWM 出力 3 の逆相波形
TCR ₃	ch3 タイマカウンタのクリア要因と入力クロックを選択
TMDR ₃	ch3 をリセット同期 PWM モードとして動作
TGRA ₃	PWM 周期を設定
TGRB ₃ TGRA ₄ TGRB ₄	デューティを設定

正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
----------------	-----	-----------	------	----------------------

動作説明

(1) 図2.14に動作原理を示します。図2.14に示すようにSH7145のハードウェア処理およびソフトウェア処理により、各PWM出力端子 (TIOC3B/D、TIOC4A/B/C/D) から3相のPWM波形を出力します。



ハードウェア処理
無し
ソフトウェア処理
(1)ピンファンクションコントローラの初期設定 ・ TIOC3B/D、TIOC4A/B/C/D 出力端子に設定
(2)タイマの初期設定 ・ TCR_3の設定 ・ TOCRの設定 ・ TGRの設定 ・ TOERの設定 ・ TMDRの設定
(3)タイマカウンタ動作を開始

ハードウェア処理
TGRB_3、TGRA_4、TGRB_4のコンペアマッチおよびタイマカウンタクリアによりPWM波形を出力する
ソフトウェア処理
無し

図 2.14 リセット同期 PWM 波形の動作原理

正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	rst_pwm	PFC および PWM 出力の設定を行う。

(2) 引数の説明

ラベル名、レジスタ名	機能	データ長	使用モジュール名	入出力
pul_cyc1	パルス周期に相当するタイマ値を設定 パルス周期は以下の式にて算出 パルス周期 (ns) = タイマ値 × φ 周期 (40.0MHz 動作時 25.0ns)	1 ワード	メインルーチン	入力
pul_duty3b pul_duty4a pul_duty4b	TIOC 端子から出力される波形変化タイミングを設定			

(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
P_PORTE.PECRL1	TIOC3B/D、TIOC4A/B/C/D を出力端子に設定	H'FFFF83B8	H'5544
P_MTU34.TCR_3	タイマカウンタ 3 のカウンタクリア要因を TGRA_3 のコンペアマッチに設定 入力クロックは Pφ/16 を選択	H'FFFF8200	H'22
P_MTU34.TOCR	正相、逆相 PWM 出力の出力レベル反転の制御	H'FFFF820B	H'03
P_MTU34.TGRA_3	PWM 周期を設定	H'FFFF8218	pul_cyc1
P_MTU34.TGRB_3	TIOC3B/D 端子からの PWM 出力波形変化タイ ミングを設定	H'FFFF821A	pul_duty3b
P_MTU34.TGRA_4	TIOC4A/C 端子からの PWM 出力波形変化タイ ミングを設定	H'FFFF821C	pul_duty4a
P_MTU34.TGRB_4	TIOC4B/D 端子からの PWM 出力波形変化タイ ミングを設定	H'FFFF821E	pul_duty4b
P_MTU34.TOER	リセット同期 PWM 出力を許可に設定	H'FFFF821E	H'ff
P_MTU34.TMDR_3	ch3/4 をリセット同期 PWM モードに設定	H'FFFF8202	H'c8
P_STBY.MSTCR2	MTU モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd

(4) 使用 RAM 説明

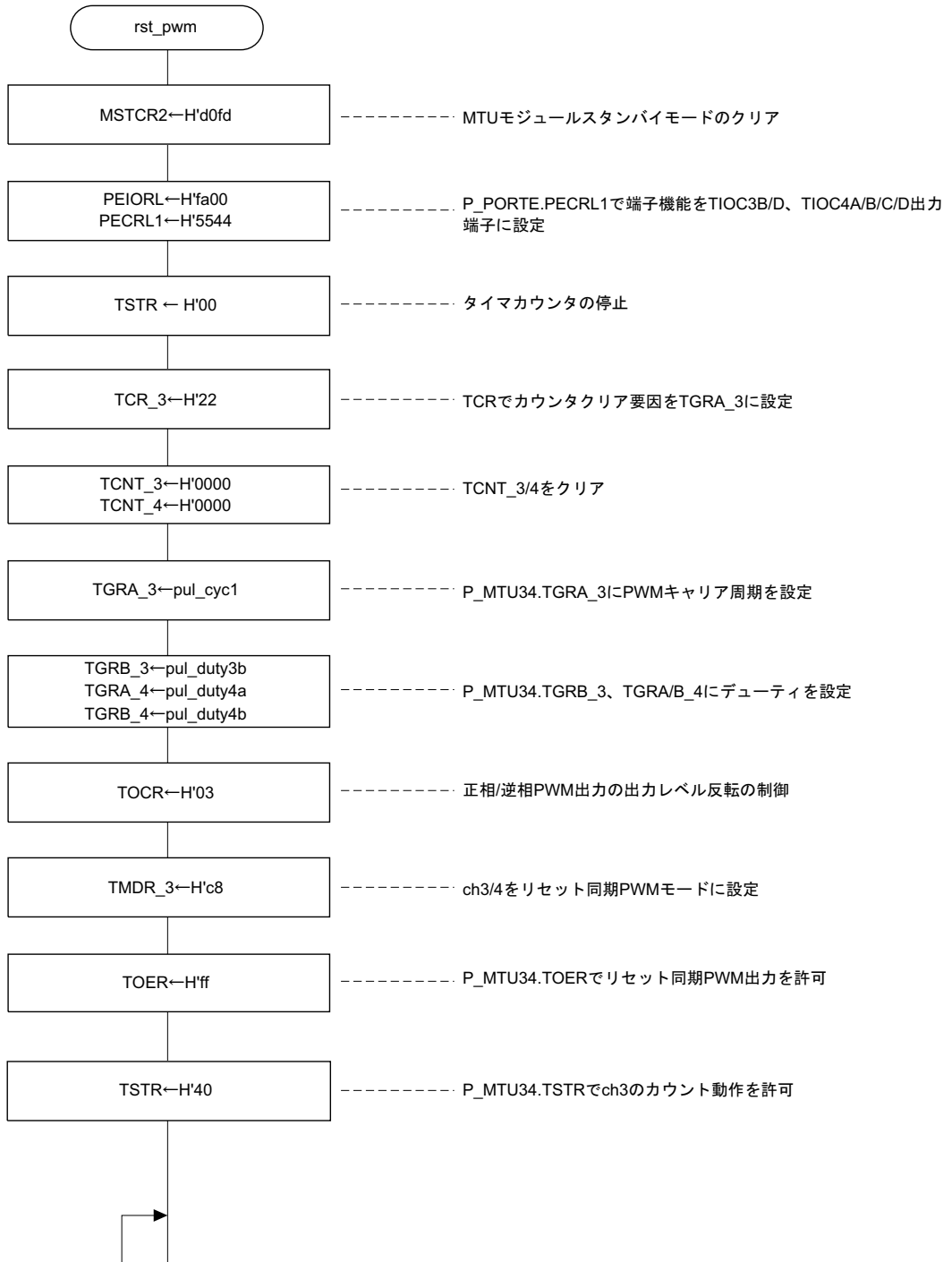
本アプリケーション例では引数以外の RAM は使用していません。

注) レジスタのラベル名は、SH7145 ヘッドファイルの名前を使用しています。

正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
----------------	-----	-----------	------	----------------------

フローチャート

(1) メインルーチン



正相・逆相 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (リセット同期 PWM モード)
プログラムリスト				
<pre> /***** / /***** #include<machine.h> #include"iodefine_7145F.h" / / /***** void rst_pwm (void) ; / / /***** #define pul_cycl (* (unsigned short *) 0xffffe00) #define pul_duty3b (* (unsigned short *) 0xffffe02) #define pul_duty4a (* (unsigned short *) 0xffffe04) #define pul_duty4b (* (unsigned short *) 0xffffe06) / / /***** / /***** MAIN PROGRAM / /***** void rst_pwm (void) { P_STBY.MSTCR2.WORD = 0xd0fd; /*Clear MTU module standby mode*/ P_PORTE.PEIORL.WORD = 0xfa00; /*TIOC3B/D,TIOC4A/B/C/D output*/ P_PORTE.PECRL1.WORD = 0x5544; P_MTU34.TSTR.BYTE = 0x00; P_MTU34.TCR_3.BYTE = 0x20; /*Counter clear by TGRA_3*/ P_MTU34.TCNT_3 = 0x0000; /*Clear timer counter*/ P_MTU34.TCNT_4 = 0x0000; P_MTU34.TGRA_3 = pul_cycl; /*Set PWM period*/ P_MTU34.TGRB_3 = pul_duty3b; /*Set duty*/ P_MTU34.TGRA_4 = pul_duty4a; P_MTU34.TGRB_4 = pul_duty4b; P_MTU34.TOCR.BYTE = 0x03; /*timer output control register*/ P_MTU34.TMDR_3.BYTE = 0xc8; /*Reset synchronized PWM mode*/ P_MTU34.TOER.BYTE = 0xff; /*Enable timer output */ P_MTU34.TSTR = 0x40; /*Start timer counter*/ while (1) ; } </pre>				

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
仕様				

- (1) 図2.15に示すように正相・逆相がノンオーバーラップの関係にあるPWM波形を3相出力します。
- (2) デューティは0%~100%まで任意の値をRAMに設定することにより変更ができます。

$$\text{デューティ} = \frac{\text{パルスのHigh幅}}{\text{パルスの周期}} \times 100 (\%)$$

- (3) 周期に同期したトグル波形出力を行います。
- (4) 内蔵周辺クロック $P\phi = 40.0\text{MHz}$ 動作時、出力するパルスの周期は50.0nsから1.63msの間任意に設定できます。

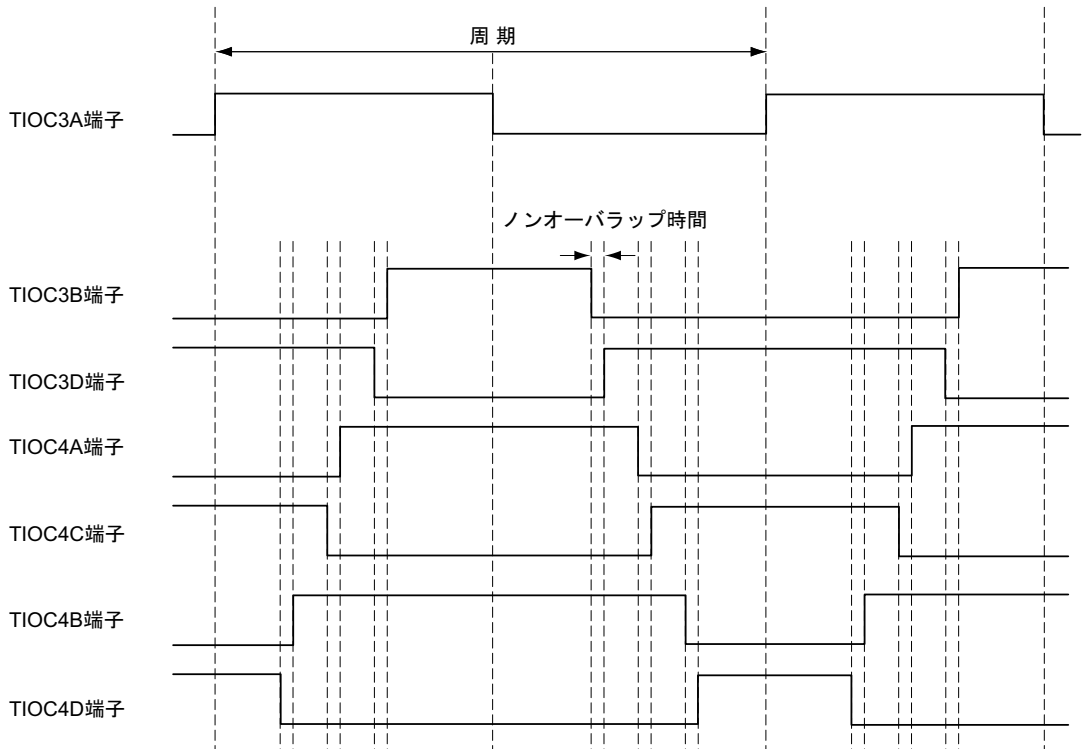


図 2.15 相補 PWM3 相出力波形

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
使用機能説明				

- (1) 本タスク例ではMTUのチャンネル3、4を使用して正相と逆相がノンオーバーラップの関係にあるPWM波形を3相出力します。
- (a) 図2.16に本タスク例で使用するMTU/ch3、4のブロック図を示します。
また、本タスクでは以下の機能を使用します。
- 正相と逆相がノンオーバーラップの関係にある PWM 波形を 3 相出力する機能。(相補 PWM モード)
 - コンペアマッチ発生時、バッファレジスタ (TGRC/D_3、TGRC/D_4) の内容をコンペアレジスタ (TGRA/B_3、TGRA/B_4) に転送する機能。
 - PWM 波形の周期に同期したトグル波形を出力する機能。

(入カロック)

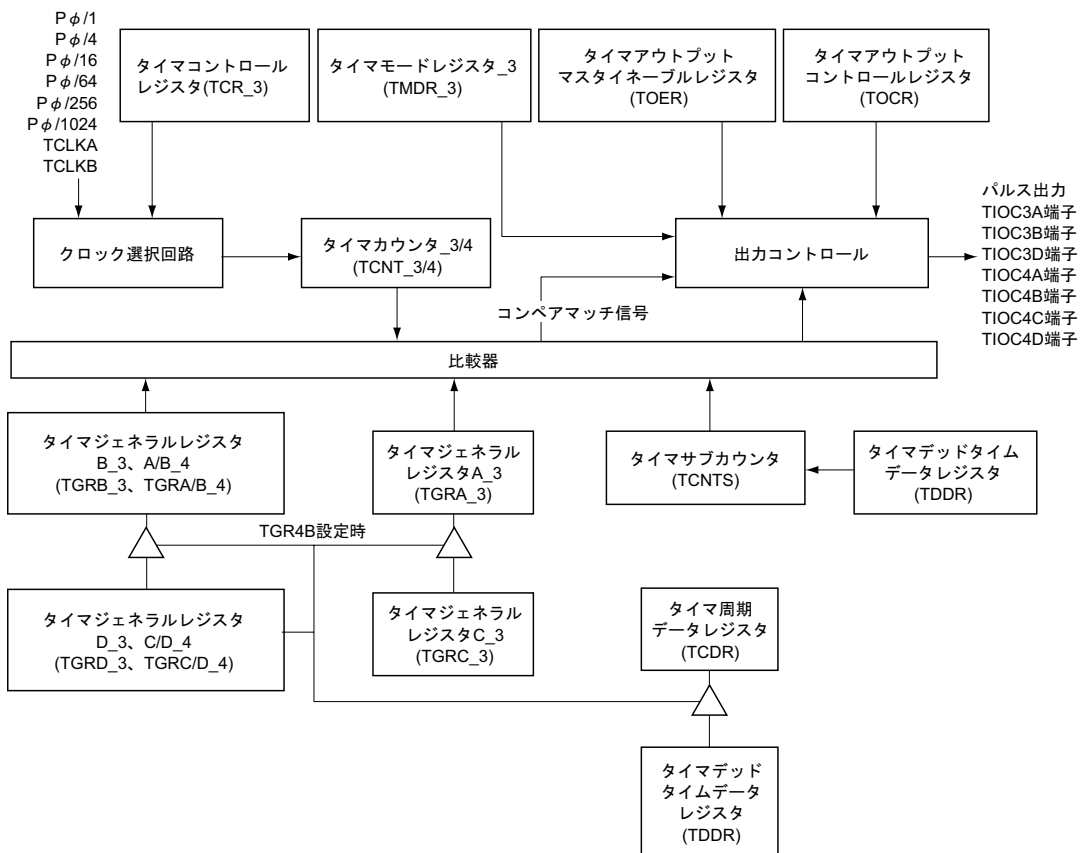


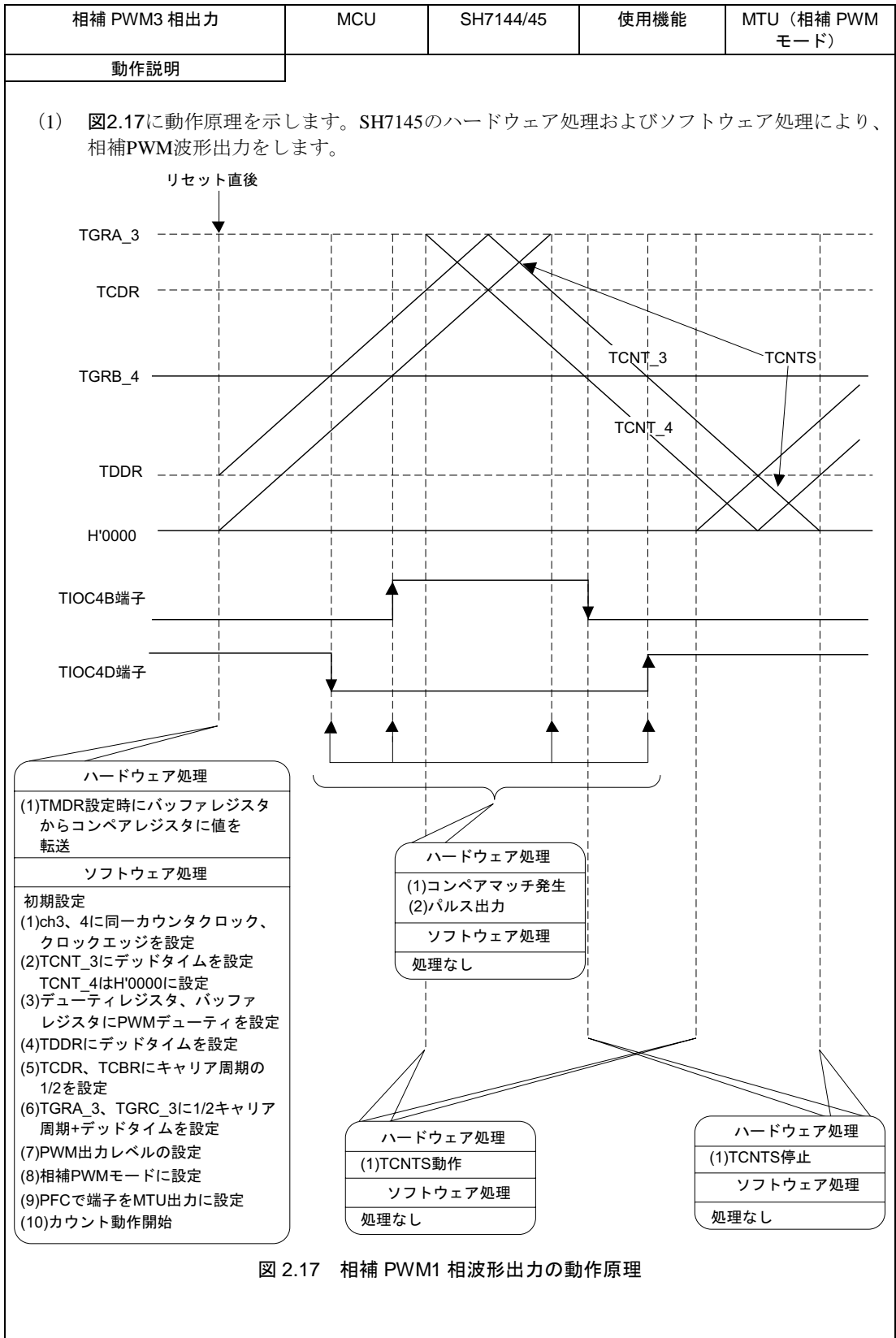
図 2.16 MTU/ch3、4 ブロック図

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
使用機能説明				

- (1) 表2.6に本タスクの機能割り付けを示します。表に示すようにMTUの機能を割り付け、PWMパルスを出力します。

表 2.6 機能割り付け

端子、レジスタ名	機能割り付け
TIOC3A	PWM 周期に同期したトグル出力
TIOC3C	PWM 出力 1
TIOC3D	PWM 出力 1 とノンオーバーラップ関係にある逆相波形
TIOC4A	PWM 出力 2
TIOC4B	PWM 出力 3
TIOC4C	PWM 出力 2 とノンオーバーラップ関係にある逆相波形
TIOC4D	PWM 出力 3 とノンオーバーラップ関係にある逆相波形
TOCR	PWM 周期に同期したトグル出力の許可/禁止
TOER	MTU 出力端子の出力の許可/禁止
TCR_3	ch3のタイマカウンタのクリア要因と入力クロックを選択
TMDR_3	ch3、4を相補 PWM モードに設定
TGRA_3	TCNT_3の上限値の設定 (キャリア周期の 1/2+デッドタイム)
TGRC_3	TGRA_3のバッファレジスタ
TGRB_3	出力パルスの変化点の設定 (コンペアレジスタ)
TGRA_4	
TGRB_4	
TGRC_4	TGRA_4のバッファレジスタ
TGRD_4	TGRB_4のバッファレジスタ
TDDR	デッドタイムの設定
TCDR	TCNT_4の上限値の設定 (キャリア周期の 1/2)
TGBR	TCDRのバッファレジスタ



相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
動作説明				

(2) 図2.18にPWM波形出力方式を示します。相補PWMモード設定時、データ転送およびコンペアに関しては以下の法則にしたがっています。

データ転送

- Ta 区間では、バッファレジスタにライトされたデータは、常時テンポラリレジスタに転送されます。
- Tb1 区間では、転送モードが山で転送に設定時、バッファレジスタからテンポラリレジスタにデータは転送されません。Tb2 区間では Ta 区間と同様の動作をします。同様に谷に設定時 Tb2 区間では転送されません。
- バッファレジスタへのデータ転送は任意に行なうことができます。

コンペアマッチ

- Tb 区間では、テンポラリレジスタ、コンペアレジスタ 2 本のレジスタと TCNT_3/4 および TCNTS の 3 本のカウンタが比較され、PWM 波形を制御します。
 - (a) 領域では変更前のデータと (3)、(4) のコンペアマッチが優先されます。
 - (b) 領域では変更後のデータと (1)、(2) のコンペアマッチが優先されます。
- ただし、出力波形がアクティブレベルとなるコンペアマッチ (1)、(3) のコンペアマッチの発生はそれぞれ出力波形がポジティブレベルとなるコンペアマッチ (4)、(2) のコンペアマッチ 発生後のみおこります。

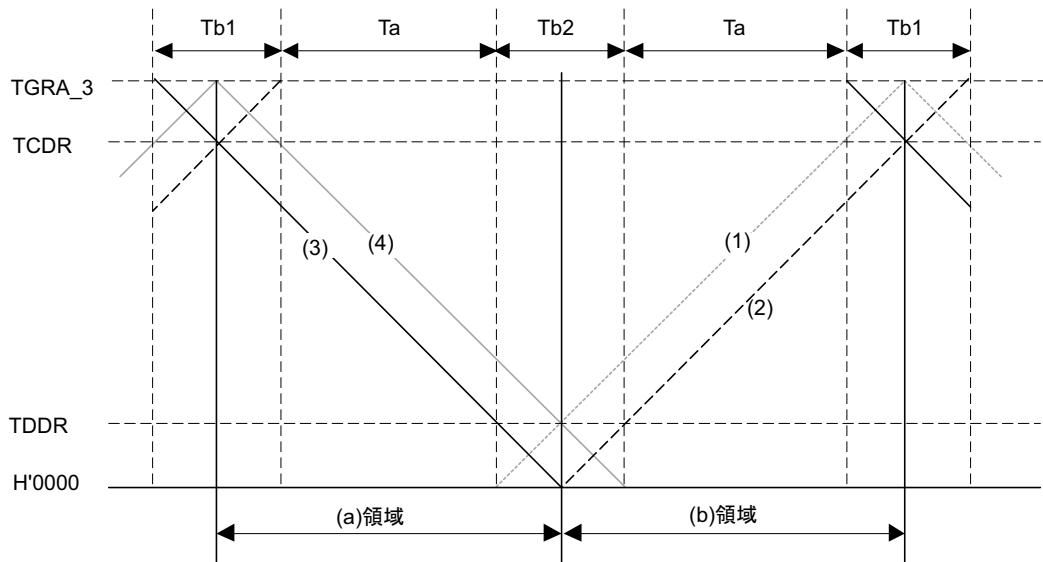
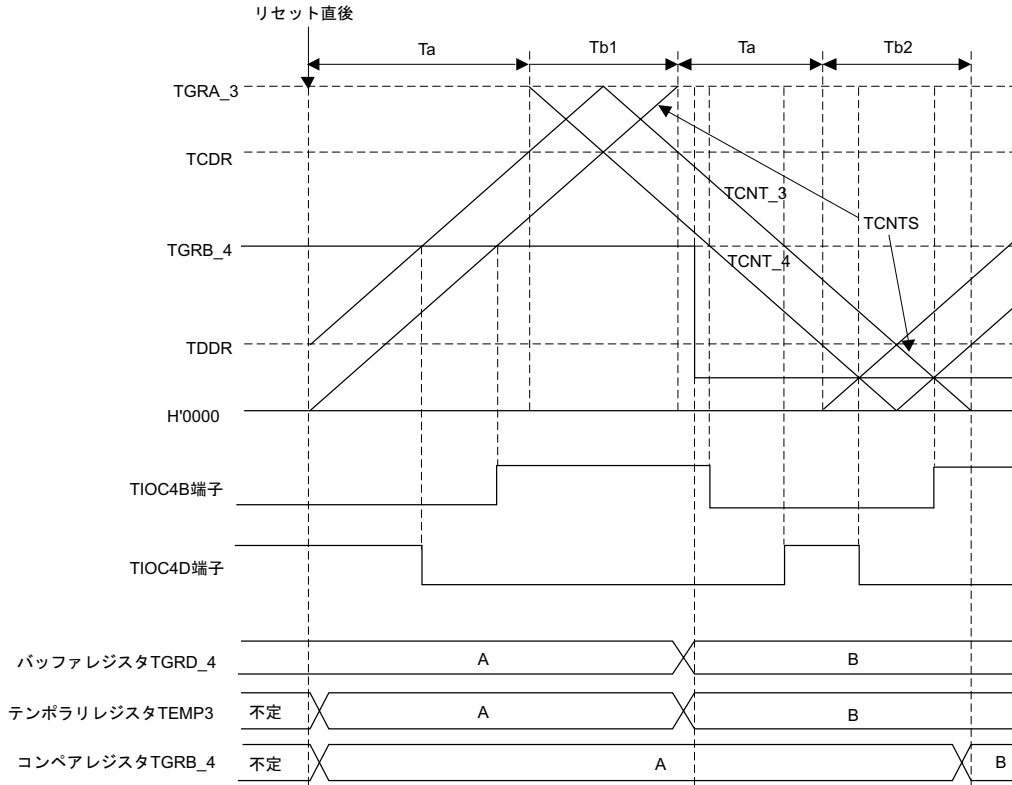


図 2.18 PWM 波形出力方式動作原理

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
動作説明				

- (3) 図2.19に動作原理を示します。SH7145のハードウェア処理およびソフトウェア処理により、相補PWM波形出力をします。
ただし、転送モードは、谷でデータを変更するモードに選択してあります。



ハードウェア処理
(1)TMDR設定時にバッファレジスタからコンペアレジスタに値を転送
ソフトウェア処理
初期設定
(1)ch3、4に同一カウンタクロック、クロックエッジを設定
(2)TCNT_3にデッドタイムを設定 TCNT_4はH'0000に設定
(3)デューティレジスタ、バッファレジスタにPWMデューティを設定
(4)TDDRにデッドタイムを設定
(5)TCDR、TCBRIにキャリア周期の1/2を設定
(6)TGRA_3、TGRC_3に1/2キャリア周期+デッドタイムを設定
(7)PWM出力レベルの設定
(8)相補PWMモードに設定
(9)PFCで端子をMTU出力に設定
(10)カウント動作開始

※1

ハードウェア処理
(1)データをバッファレジスタからテンポラリレジスタに転送
ソフトウェア処理
(1)TGRD_4にデータを設定

ハードウェア処理
(1)データをテンポラリレジスタからコンペアレジスタに転送
ソフトウェア処理
処理なし

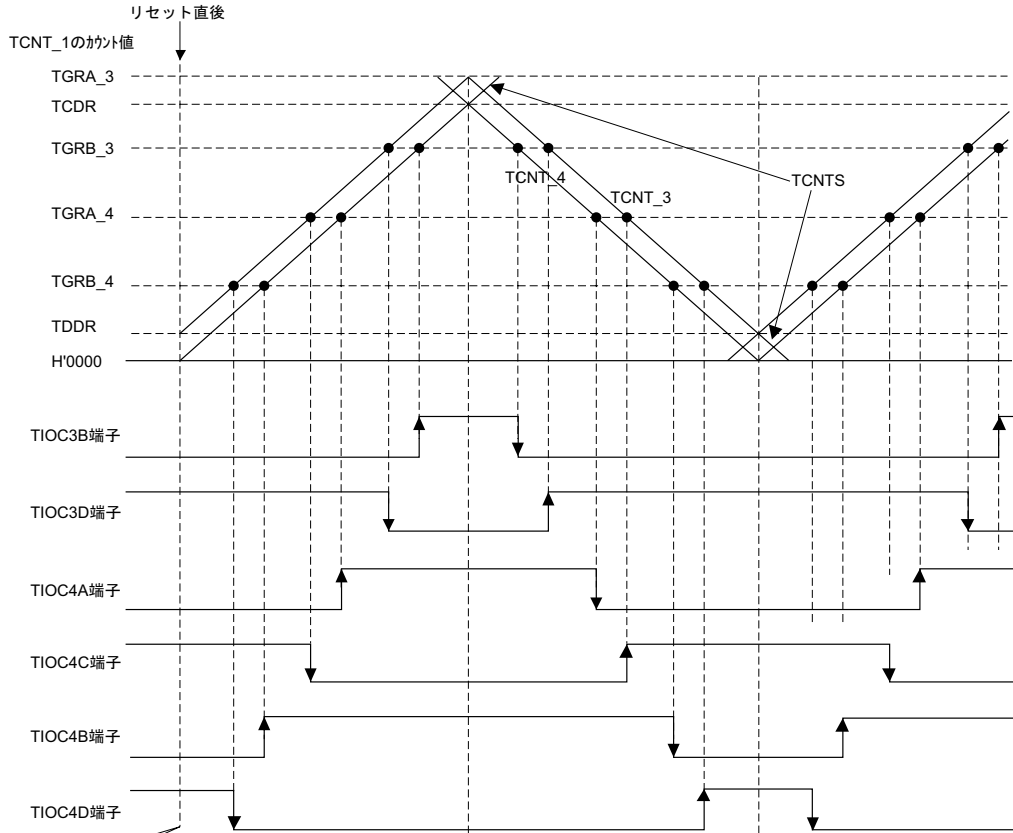
※1：本処理は3相出力時もTGRD_4にデータを設定した時点で発生

図 2.19 相補 PWM1 相波形出力の動作原理

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
-------------	-----	-----------	------	------------------

動作説明

(4) 図2.20に動作原理を示します。図に示すようにSH7145のハードウェア処理およびソフトウェア処理により、ch3、ch4の各PWM出力端子 (TIOC3B/D、TIOC4A/B/C/D) から3相のPWM出力をします。



ハードウェア処理
無し
ソフトウェア処理
初期設定
(1)ch3、4に同一カウンタクロック、クロックエッジを設定
(2)TCNT_3にデッドタイムを設定 TCNT_4はH'0000に設定
(3)デューティレジスタ、バッファレジスタにPWMデューティを設定
(4)TDDRにデッドタイムを設定
(5)TCDR、TCBRにキャリア周期の1/2を設定
(6)TGRA_3、TGRB_3に1/2キャリア周期+デッドタイムを設定
(7)PWM出力レベルの設定
(8)相補PWMモードに設定
(9)PFCで端子をMTU出力に設定
(10)カウント動作開始

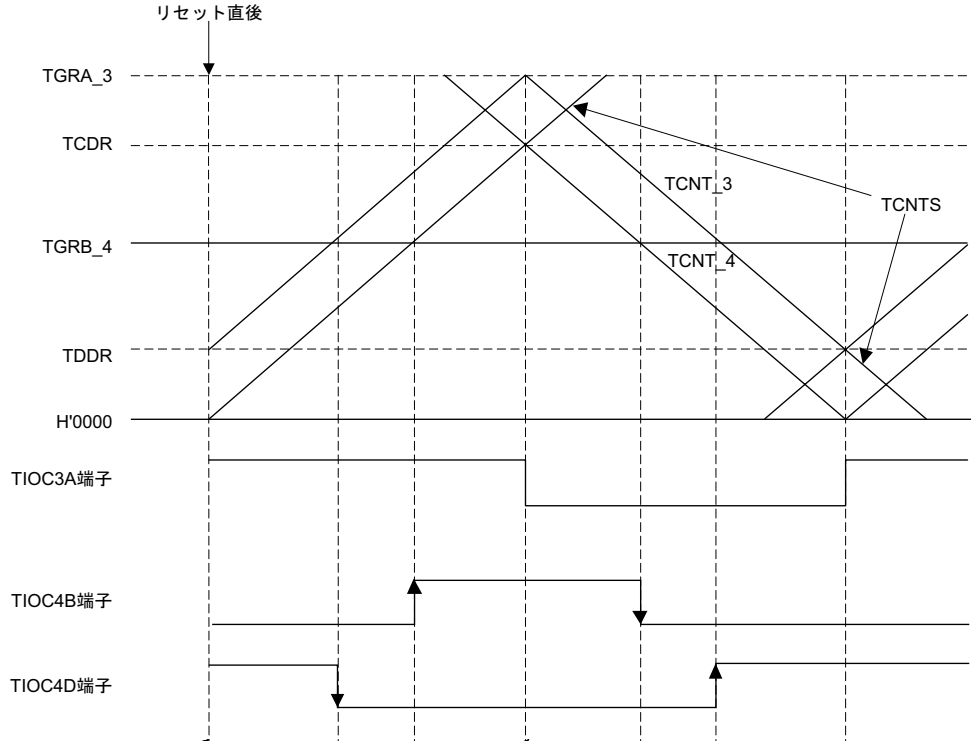
ハードウェア処理
TGRA_3のコンペアマッチ割り込み発生
ソフトウェア処理
(1)RAMの内容をバッファレジスタに転送

図 2.20 PWM 波形の動作原理

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
-------------	-----	-----------	------	------------------

動作説明

- (5) 図2.21に動作原理を示します。SH7145のハードウェア処理およびソフトウェア処理により、TIOC3A端子よりPWM周期に同期したトグル出力を行ないます。



ハードウェア処理
(1)TMDR設定時にバッファレジスタからコンペアレジスタに値を転送
ソフトウェア処理
初期設定
(1)ch3、4に同一カウンタクロック、クロックエッジを設定
(2)TCNT_3にデッドタイムを設定 TCNT_4はH'0000に設定
(3)デューティレジスタ、バッファレジスタにPWMデューティを設定
(4)TDDRにデッドタイムを設定
(5)TCDR、TCBRにキャリア周期の1/2を設定
(6)TGRA_3、TGRC_3に1/2キャリア周期+デッドタイムを設定
(7)PWM出力レベルの設定
(8)相補PWMモードに設定
(9)PFCで端子をタイマ出力に設定
(10)カウント動作開始

ハードウェア処理
TGRA_3のコンペアマッチ発生 (1)PWM周期に同期したトグル出力をする
ソフトウェア処理
処理なし

ハードウェア処理
ch4のアンダフロー発生 (1)PWM周期に同期したトグル出力をする
ソフトウェア処理
処理なし

図 2.21 PWM 周期に同期したトグル波形出力の動作原理

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	comple	相補 PWM 出力の設定
データ設定	setdata	バッファレジスタに波形変化タイミングを設定

(2) 引数の説明

ラベル名、 レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
pul_cyc1	パルスの 1/2 周期+デッドタイム値を設定 パルスの周期は以下の式にて算出 パルス周期 (ns) = タイマ値 × φ 周期 (40.0MHz 動作時 25.0ns)	1 ワード	メインルーチン	入力
pul_duty3d pul_duty4c pul_duty4d	TIOC 端子から出力される波形変化タイミング を設定			
c_cyc	PWM キャリア周期レジスタの値を設定			
dead_time	ノンオーバーラップ期間を設定			
			メインルーチン データ設定	

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
ソフトウェア説明				

(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
P_STBY.MSTCR2	MTU モジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd
P_PORTE.PEIORL	TIOC3B/D、TIOC4A/B/C/D 端子を出力に設定	H'FFFF83B4	H'fb00
P_PORTE.PECRL1	端子機能を TIOC3B/D、TIOC4A/B/C/D 端子に設定	H'FFFF83B8	H'5545
P_MTU34.TCR_3	MTU/ch3 のカウンタクロックとクロックエッジを設定	H'FFFF8200	H'01
P_MTU34.TCR_4	MTU/ch4 のカウンタクロックとクロックエッジを設定	H'FFFF8201	H'01
P_MTU34.TIER_3	TGIA_3 割り込みを許可	H'FFFF8208	H'41
P_MTU34.TGRA_3	キャリア周期の 1/2+デッドタイムを設定	H'FFFF8218	pul_cyc1
P_MTU34.TGRC_3	P_MTU34.TGRA_3 のバッファレジスタ (P_MTU34.TGRA_3 と同じ値を設定)	H'FFFF8224	pul_cyc1
P_MTU34.TCNT_3	デッドタイムを設定	H'FFFF8210	dead_time
P_MTU34.TCNT_4	H'0000 を設定	H'FFFF8212	H'0000
P_MTU34.TGRB_3	TIOC3B、TIOC3D から出力させる PWM デューティを設定	H'FFFF821A	pul_duty3d
P_MTU34.TGRA_4	TIOC4A、TIOC4C から出力させる PWM デューティを設定	H'FFFF821C	pul_duty4c
P_MTU34.TGRB_4	TIOC4B、TIOC4D から出力させる PWM デューティを設定	H'FFFF821E	pul_duty4d
P_MTU34.TGRD_3	P_MTU34.TGRB_3 のバッファレジスタ (P_MTU34.TGRB_3 と同じ値を設定)	H'FFFF8226	pul_duty3d
P_MTU34.TGRC_4	P_MTU34.TGRA_4 のバッファレジスタ (P_MTU34.TGRA_4 と同じ値を設定)	H'FFFF8228	pul_duty4c
P_MTU34.TGRD_4	P_MTU34.TGRB_4 のバッファレジスタ (P_MTU34.TGRB_4 と同じ値を設定)	H'FFFF822A	pul_duty4d
P_MTU34.TDDR	デッドタイムを設定	H'FFFF8216	dead_time
P_MTU34.TCDR	キャリア周期の 1/2 を設定	H'FFFF8214	c_cyc
P_MTU34.TCBR	P_MTU34.TCDR のバッファレジスタ (P_MTU34.TCDR と同じ値を設定)	H'FFFF8222	c_cyc
P_MTU34.TOCR	PWM 周期に同期したトグル出力の許可と正相、逆相の出力レベルの設定	H'FFFF820B	H'43
P_MTU34.TMDR_3	相補 PWM モードに設定	H'FFFF8202	H'ff
P_MTU34.TOER	PWM 波形出力端子を出力許可に設定	H'FFFF820A	H'ff
P_INTC.IPRE	MTU チャネル 3 の割り込み優先レベルを 15 に設定	H'FFFF8350	H'00f0

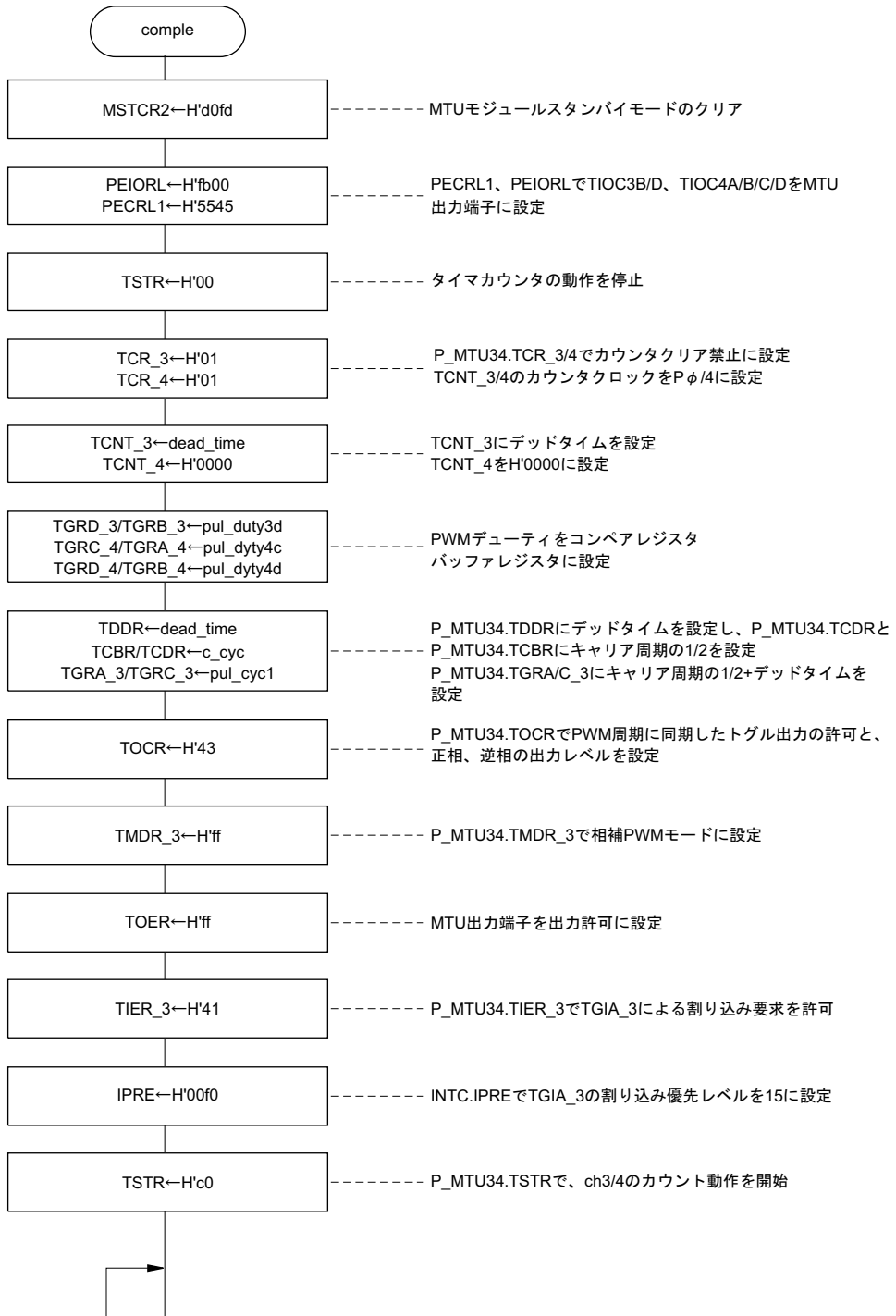
(4) 使用 RAM 説明

本アプリケーション例では引数以外の RAM は使用していません。

注) レジスタのラベル名は、SH7145 ヘッドファイルの名前を使用しています。

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
フローチャート				

(1) メインルーチン



相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
プログラムリスト				
<pre> /*-----*/ /* INCLUDE FILE */ /*-----*/ #include <machine.h> #include "iodefine_7145F.h" /*-----*/ /* PROTOTYPE */ /*-----*/ extern void comple (void) ; #pragma interrupt (setdata) /*-----*/ /* RAM ALLOCATION */ /*-----*/ #define pul_cycl (* (unsigned short *) 0xffffe00) #define pul_duty3d (* (unsigned short *) 0xffffe02) #define pul_duty4c (* (unsigned short *) 0xffffe04) #define pul_duty4d (* (unsigned short *) 0xffffe06) #define c_cyc (* (unsigned short *) 0xffffe08) #define dead_time (* (unsigned short *) 0xffffe0a) /*-----*/ /* MAIN PROGRAM */ /*-----*/ void comple (void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* MTU module standby mode clear */ P_PORTE.PEIORL.WORD = 0xfb00; /* TIOC3B/D,TIOC4A/B/C/D = output */ P_PORTE.PECRL1.WORD = 0x5545; P_MTU34.TSTR.BYTE = 0x00; /* Stop timer count */ P_MTU34.TCR_3.BYTE = 0x01; /* Don't clear TCNT_3 */ P_MTU34.TCR_4.BYTE = 0x01; /* Don't clear TCNT_4 */ P_MTU34.TCNT_3 = dead_time; /* Set dead time */ P_MTU34.TCNT_4 = 0x0000; P_MTU34.TGRD_3 = pul_duty3d; /* TGRB_3 buffer register */ P_MTU34.TGRB_3 = pul_duty3d; /* PWM output1 compare register */ P_MTU34.TGRC_4 = pul_duty4c; /* TGRA_4 buffer register */ P_MTU34.TGRA_4 = pul_duty4c; /* PWM output2 compare register */ P_MTU34.TGRD_4 = pul_duty4d; /* TGRB_4 buffer register */ </pre>				

相補 PWM3 相出力	MCU	SH7144/45	使用機能	MTU (相補 PWM モード)
プログラムリスト	<pre> P_MTU34.TGRB_4 = pul_duty4d; /* PWM output3 compare register */ P_MTU34.TDDR = dead_time; /* Dead time register */ P_MTU34.TCBR = c_cyc; /* 1/2 carrier period */ P_MTU34.TCDR = c_cyc; /* TCDR buffer register */ P_MTU34.TGRA_3 = pul_cycl1; /* 1/2 carrier period + dead time */ P_MTU34.TGRC_3 = pul_cycl1; /* TGRA_3 buffer register */ P_MTU34.TOCR.BYTE = 0x43; /* Timer output control register */ P_MTU34.TMDR_3.BYTE = 0xff; /* Set complementary-pwm mode */ P_MTU34.TOER.BYTE = 0xff; /* Timer output enable register */ P_MTU34.TIER_3.BYTE = 0x41; /* Timer interrupt enable register */ INTC.IPRE.WORD = 0x00f0; /* Set initialize level = 15 */ set_imask (0x0) ; /* Set imask level = 0 */ P_MTU34.TSTR.BYTE = 0xc0; /* Start timer counter3/4 */ while (1) ; /* Loop */ void setdata () { P_MTU34.TSR_3.BYTE &= 0xfe; /* interrupt flag clear */ P_MTU34.TCBR = c_cyc; P_MTU34.TGRC_3 = pul_cycl1; P_MTU34.TGRD_3 = pul_duty3d; P_MTU34.TGRC_4 = pul_duty4c; P_MTU34.TGRD_4 = pul_duty4d; } </pre>			

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
仕様				

- (1) 図2.22に示すようにch1に2本の外部クロックを入力し、そのパルスの位相差によりカウンタをカウントアップまたはカウントダウンします。またch0に設定した測定時間(測定時間1、2)に同期してch1のカウンタ数を測定し、結果をRAMに設定します。
- (2) タイマカウンタの初期値をH'0000として、ソフトウェアカウンタを用いて-2147483648～2147483647までカウントできます。

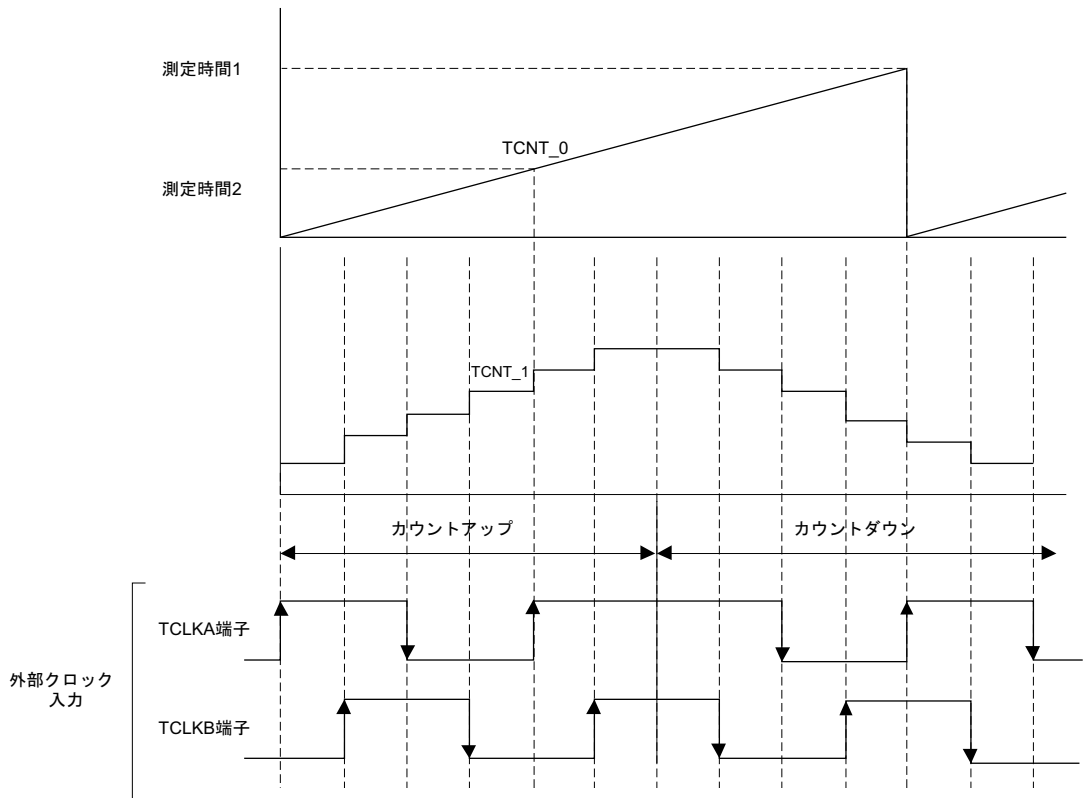


図 2.22 2相エンコーダカウンタの取り込み

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
使用機能説明				

- (1) 本タスク例では、MTUのch1をアップ/ダウンカウンタとして使用し、TGRA/B_0に測定時間を設定します。TGRA/B_0のアウトプットコンペアをトリガとしてch1のインプットキャプチャで制御周期時のTCNT_1の値を取り込みます。また、ch0のインプットキャプチャを用いてch1のカウント入力クロック幅を取り込みます。
- (a) 図2.23にch0のブロック図を示します。ch0では、以下の機能を使用して測定時間毎にch1のインプットキャプチャのトリガを出力します。ch1はインプットキャプチャ信号入力時にTCNT_1の値を測定します。
- ソフトウェアを介さずハードウェアで自動的にパルスを出力する機能。(アウトプットコンペア)
 - パルスの入力エッジの検出を行いタイマ値を内蔵レジスタに取り込む機能。(インプットキャプチャ)

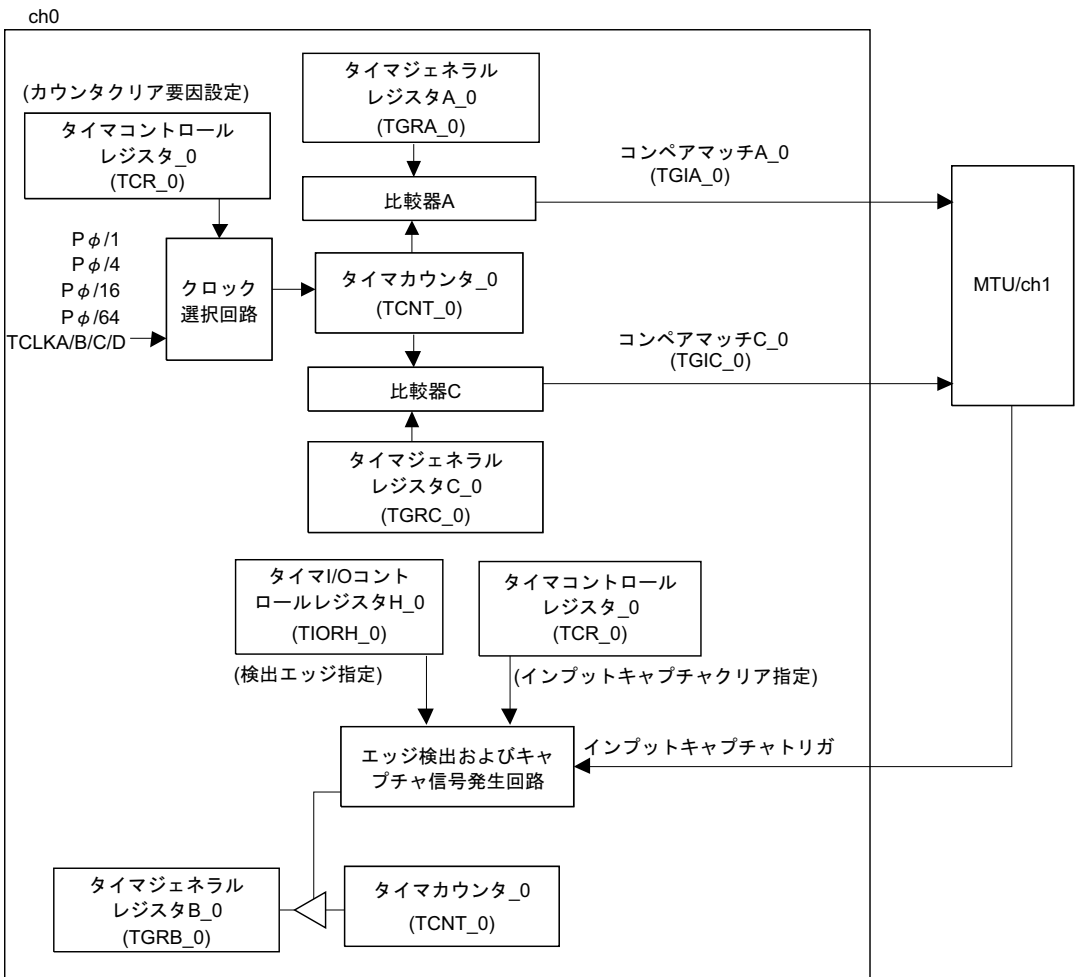


図 2.23 MTU/ch0 ブロック図

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
使用機能説明				

- (b) 図2.24にch1のブロック図を示します。ch1では以下の機能を使用してタイマカウンタをアップ/ダウンカウントします。入力キャプチャ信号立ち上がりエッジ検出時のカウンタ値を測定結果とします。
- 2本の外部クロックの位相差を検出し、タイマカウンタをアップ/ダウンカウントする機能。(位相計数モード)
 - パルスの入力エッジを検出を行い、そのときのタイマ値を内部レジスタに取り込む機能。(入力キャプチャ)
 - インพุットキャプチャ発生時、割り込み処理を起動する機能。
 - パルスの入力エッジ検出時、タイマカウンタをクリアする機能。
 - タイマカウンタのオーバーフローまたはアンダーフロー検出時、割り込み処理を起動する機能。

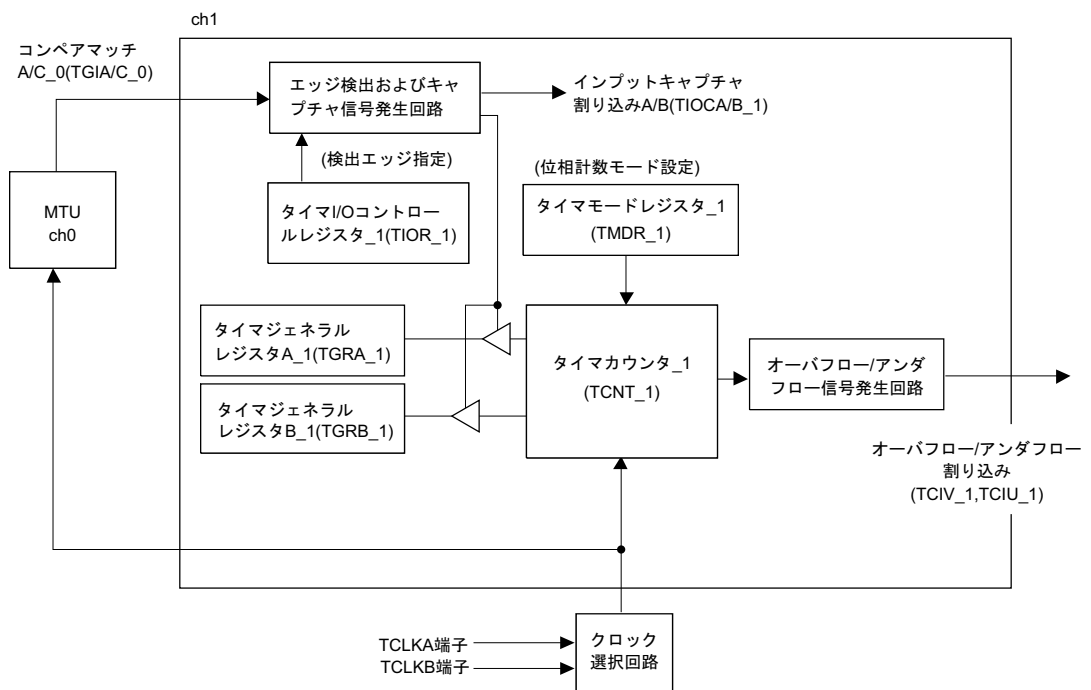


図 2.24 MTU/ch1 ブロック図

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
使用機能説明				

- (2) 表2.7に本タスク例の機能割り付けを示します。表に示すようにMTUの機能を割り付け、2相エンコーダパルスの2つの位相差を検出し、カウンタをアップ/ダウンカウントします。

表 2.7 機能割り付け

端子・レジスタ名	機能割り付け
TCLKA	外部クロック入力端子
TCLKB	
TSTR	MTU/ch0,1のタイマカウンタ動作を許可/禁止
TCR_0	カウンタクロック、カウンタクリア要因の選択
TIORH_0	TIOC0Aをアウトプットコンペアに設定。TIOC0BをTCNT_1のカウンタアップ/カウントダウンでインプットキャプチャに設定
TIORL_0	TIOC0Cをアウトプットコンペアに設定
TGRA_0	測定時間1の設定
TGRB_0	インプットキャプチャBによりカウント結果が格納
TGRC_0	測定時間2を設定
TMDR_1	MTU/ch1を位相計数モードに設定
TCR_1	カウンタクロック、カウンタクリア要因の選択
TIOR_1	TIOC1A/BをTGRA_0、TGRC_0のアウトプットコンペア発生でインプットキャプチャに設定
TIER_1	TIOC1A/B、TCIU_1、TCIV_1による割り込みを許可
TGRA_1	インプットキャプチャによるカウント結果の格納
TGRB_1	

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
動作説明				

(1) 図2.25に動作原理を示します。SH7145のハードウェア処理およびソフトウェア処理によりカウンタをカウントアップまたはカウントダウンします。

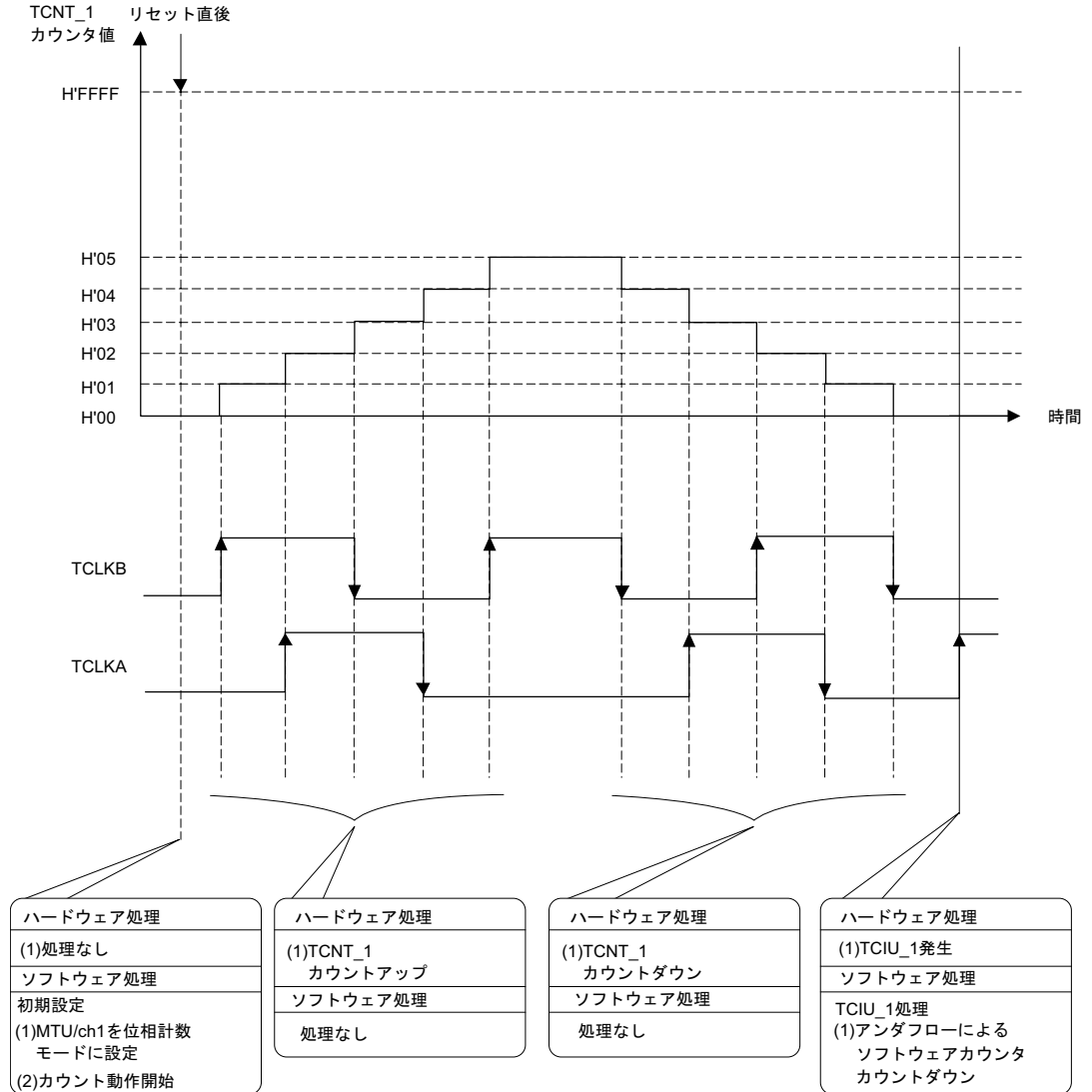
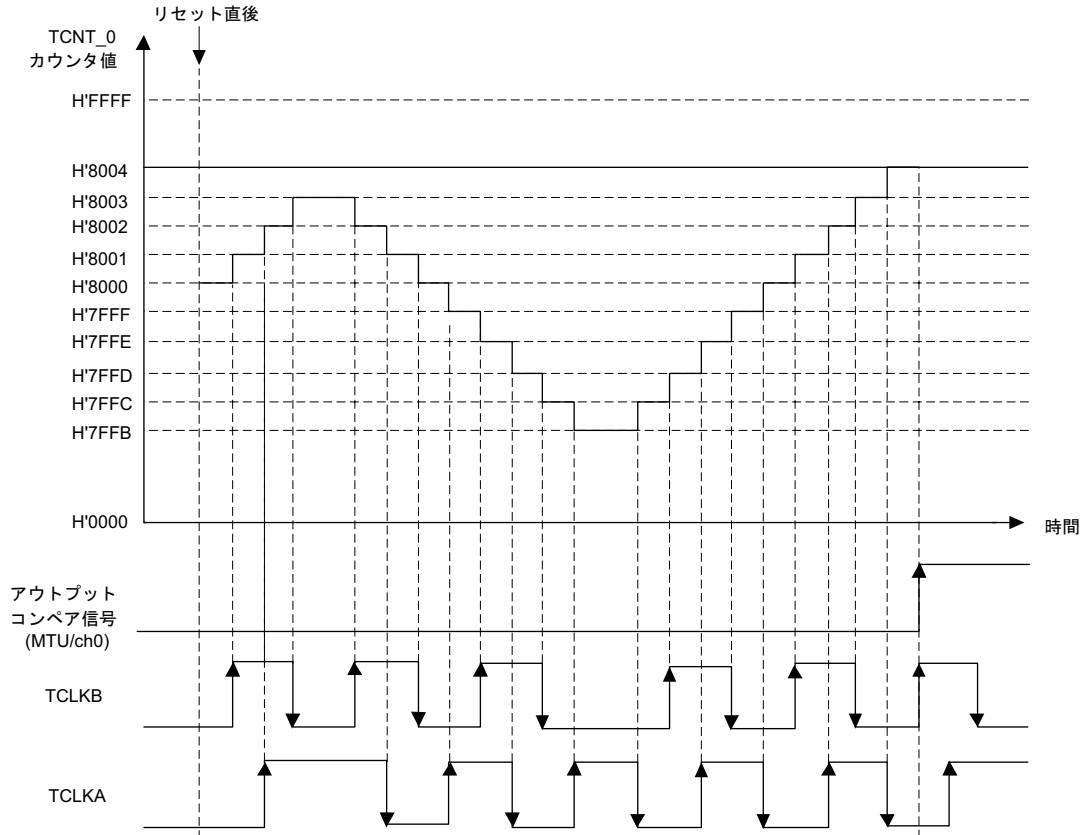


図 2.25 位相計数モードの動作原理 (1)

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
動作説明				

(2) 図2.26に示すようにSH7145のハードウェア処理およびソフトウェア処理により、MTU/ch0アウトプットコンペア発生で、TCNT_1のカウンタ数を測定します。



ハードウェア処理
処理なし
ソフトウェア処理
初期設定
(1)カウンタクロックをPφ/1に設定
(2)インプットキャプチャ信号をch0のアウトプットコンペアに設定
(3)測定時間1をTGRA_0に、測定時間2をTGRB_0に設定
(4)MTU/ch1を位相計数モードに設定
(5)インプットキャプチャ信号をch0のアウトプットコンペアに設定
(6)インプットキャプチャA、インプットキャプチャB、およびオーバーフロー/アンダフロー割り込み許可
(7)カウント動作許可

ハードウェア処理
(1)MTU/ch0アウトプットコンペア発生
(2)MTU/ch1インプットキャプチャ発生
ソフトウェア処理
(1)カウント結果をRAMIに設定

図 2.26 位相計数モードの動作原理 (2)

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	en2	MTU等の初期設定
カウンタ値測定1	phacnt1	TGIA_1により起動し、TGRA_1の値からアップ/ダウンカウント結果をRAMに設定。TGRB_0の値からカウンタ周期結果をRAMに設定
カウンタ値測定2	phacnt2	TGIB_1により起動し、TGRB_1の値からアップ/ダウンカウント結果をRAMに設定
オーバフロー	ovf1	TCIU_1により起動し、ソフトウェアカウンタのインクリメント
アンダフロー	unf1	TCIU_1により起動し、ソフトウェアカウンタのデクリメント

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
msr_tim1 msr_tim2	カウンタ測定時間に相当するタイム値を設定 測定時間は以下の式にて求める 測定時間(ns)=タイム値×φ周期(40.0MHz動作時25.0ns)	ワード	メインルーチン	入力
cnt_data1 cnt_data2	アップ/ダウンカウント結果を設定	ロングワード	カウンタ値測定1 カウンタ値測定2	出力
p_cycle	カウント周期結果を設定	ロングワード	カウンタ値測定2	

(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
P_STBY.MSTCR2	MTUのモジュールスタンバイモードのクリア	H'FFFF861E	H'd0fd
P_PORTA.PACRL2	PA6をTCLKA入力端子、PA7をTCLKB入力端子に設定	H'FFFF838E	H'5000
P_MTU0.TCR_0	カウンタクロック、カウンタクリア要因の選択	H'FFFF8260	H'20
P_MTU0.TIORH_0	TIOC0Aをアウトプットコンペアに設定する。 TIOBC0BをTCNT_1のカウントダウン/カウントアップでインプットキャプチャに設定	H'FFFF8262	H'f0
P_MTU0.TIORL_0	TIOC0Cをアウトプットコンペアに設定	H'FFFF8263	H'00
P_MTU0.TGRA_0	測定時間1を設定	H'FFFF8268	msr_tim1
P_MTU0.TGRC_0	測定時間2を設定	H'FFFF826C	msr_tim2
P_MTU1.TMDR_1	位相計数モードに設定	H'FFFF8281	H'04
P_MTU1.TIOR_1	TIOC1A/BをTGRA_0、TGRC_0のアウトプットコンペア発生でインプットキャプチャに設定	H'FFFF8282	H'ff
P_MTU1.TIER_1	TGIA/B_1、TCIU_1、TCIV_1による割り込みを許可	H'FFFF8284	H'73
P_INTC.IPRD	MTU0、MTU1割り込み優先レベルを15に設定	H'FFFF834E	H'00ff

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
ソフトウェア説明				

(4) 使用 RAM 説明

使用モジュール名	ラベル名	機能割り付け
カウンタ値測定 1、2	wrk	データ設定時のワークとして使用
全モジュール	cnt	ソフトウェアカウンタ

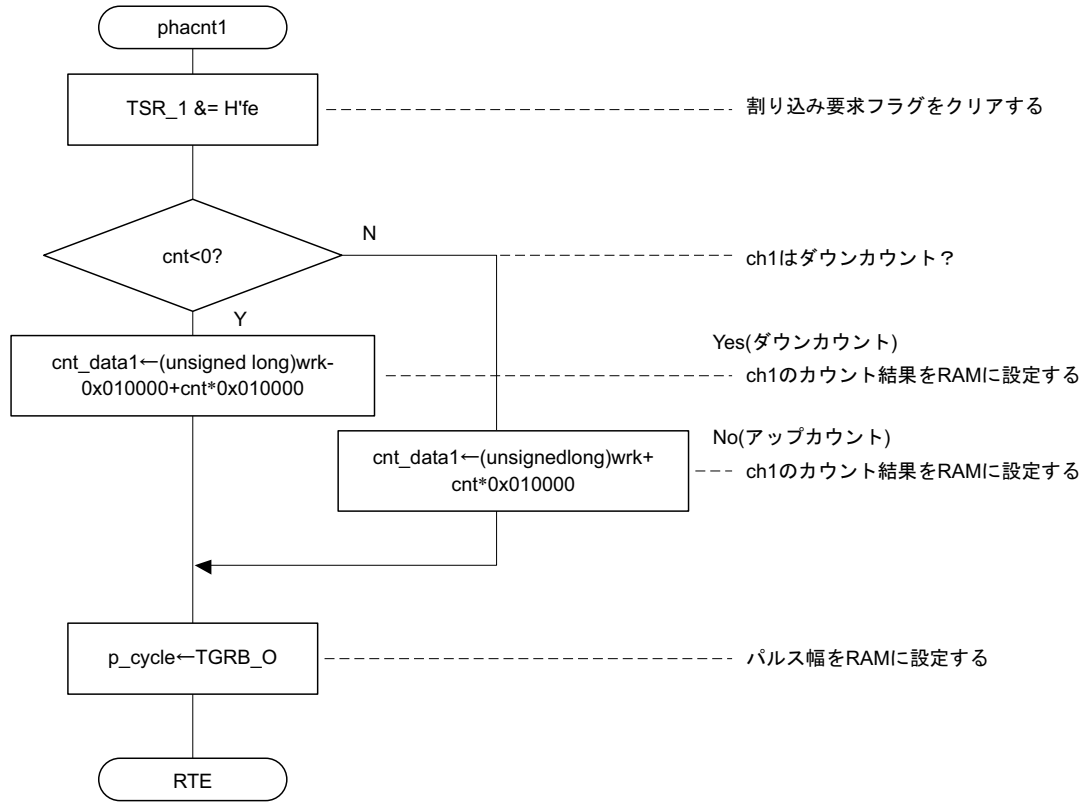
注) レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています。

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
フローチャート				
(1) メインルーチン				
<div style="text-align: center;"> <pre> graph TD Start([en2]) --> MSTCR2[MSTCR2 ← H'd0fd] MSTCR2 --> TCR_0[TCR_0 ← H'20] TCR_0 --> TIORH_0[TIORH_0 ← H'f0] TIORH_0 --> TIORL_0[TIORL_0 ← H'00] TIORL_0 --> TIOR_1[TIOR_1 ← H'ff] TIOR_1 --> TGRA_0[TGRA_0 ← msr_tim1 TGRC_0 ← msr_tim2] TGRA_0 --> TIER_1[TIER_1 ← H'73] TIER_1 --> IPRD[IPRD ← H'00ff] IPRD --> PACRL2[PACRL2 ← H'5000] PACRL2 --> TMDR_1[TMDR_1 ← H'04] TMDR_1 --> TSTR[TSTR ← H'03] TSTR --> End[] style End fill:none,stroke:none </pre> </div>				
				MTUモジュールスタンバイモードのクリア
				P_MTU0.TCR_0でTCNT_0の入クロックをPφ/1、TGRA_0とのコンペアマッチでTCNT_0をクリアに設定
				P_MTU0.TIORH_0でTIOC0Aをアウトプットコンペア動作、TIOC0BをTCNT_1のカウンタアップ/カウンタダウンでインプットキャプチャに設定
				P_MTU0.TIORL_0でTIOC0Cをアウトプットコンペアに設定
				TIOC1A/BをTGRA_0、TGRC_0のアウトプットコンペア発生でインプットキャプチャに設定
				P_MTU0.TGRA/C_0にカウンタ測定時間1、2を設定
				P_MTU1.TIER_1でTGIA/B_1、TCIU_1、TCIV_1割り込みを許可
				P_INTC.IPRDでMTU0、MTU1の割り込み優先レベルを15に設定
				PA6をTCLKA入力端子、PA7をTCLKB入力端子に設定
				MTU/ch1を位相計数モードに設定
				ch0/1のカウンタ動作を許可する

2相エンコーダカウンタ	MCU	SH7144/45	使用機能	MTU (位相計数モード)
-------------	-----	-----------	------	------------------

フローチャート

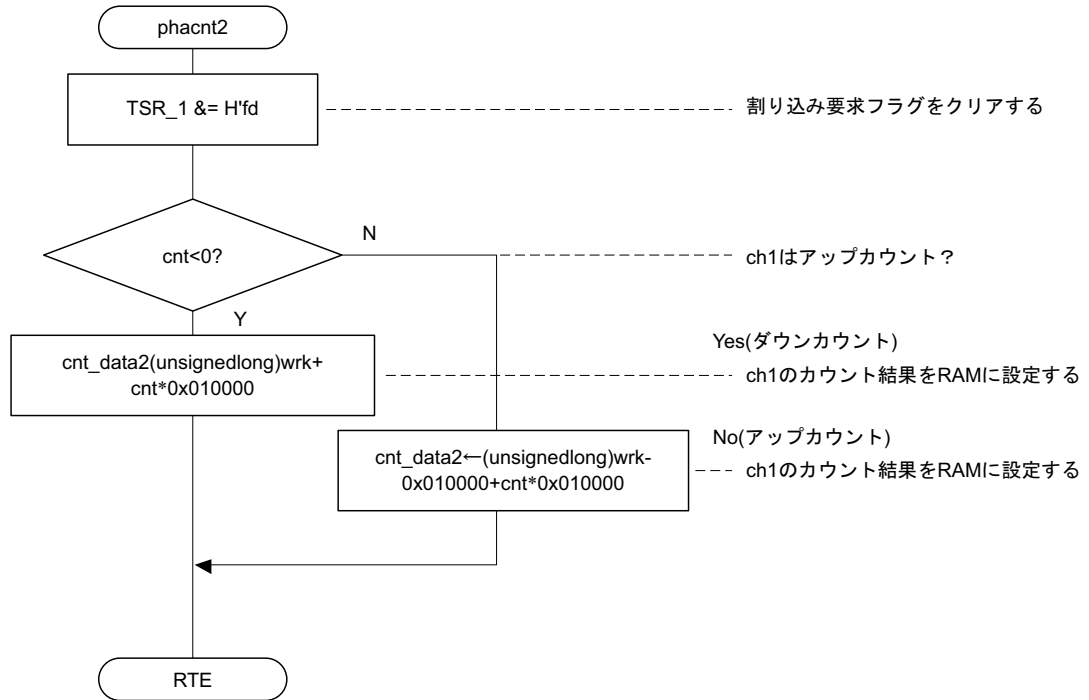
(2) カウンタ値測定 1



2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
-------------	-----	-----------	------	------------------

フローチャート

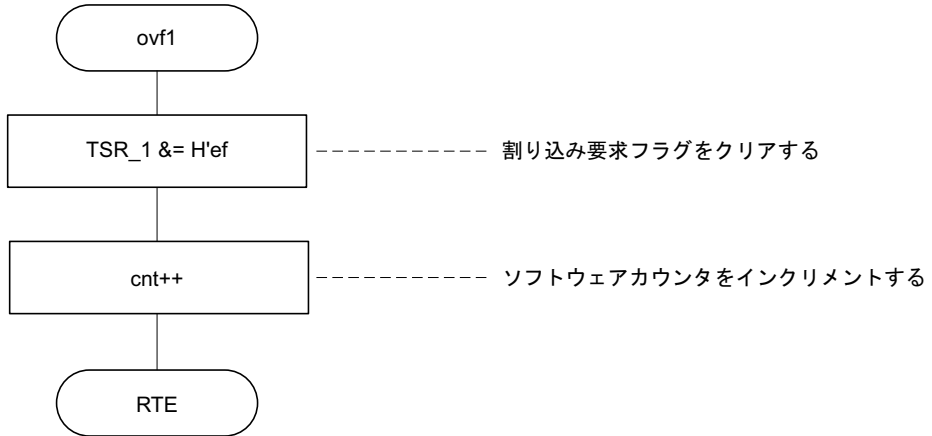
(3) カウンタ値測定 2



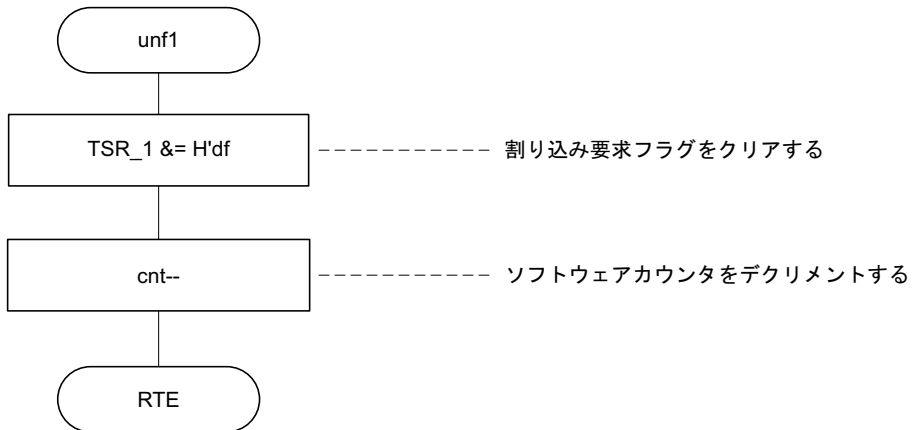
2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
-------------	-----	-----------	------	------------------

フローチャート

(4) オーバフロー



(5) アンダフロー



2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
プログラムリスト				
<pre> /*-----*/ /* INCLUDE FILE */ /*-----*/ #include <machine.h> #include "iodefine_7145F.h" /*-----*/ /* PROTOTYPE */ /*-----*/ void en2(void); #pragma interrupt(phacnt1,phacnt2,ovf1,unf1) /*-----*/ /* RAM ALLOCATION */ /*-----*/ #define msr_tim1 (*(unsigned short *)0xffffe000) #define msr_tim2 (*(unsigned short *)0xffffe002) #define cnt_data2 (*(signed long *)0xffffe004) #define cnt_data1 (*(signed long *)0xffffe008) #define p_cycle (*(unsigned long *)0xffffe00c) #define cnt (*(signed long *)0xffffe010) #define wrk (*(unsigned short *)0xffffe014) /*-----*/ /* MAIN PROGRAM */ /*-----*/ void en2(void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* MTU module stop mode clear*/ P_MTU_0.TCR_0.BYTE = 0x20; /* timer clear output compare TGRA_0*/ P_MTU_0.TIORH_0.BYTE = 0xf0; /* output compare TIOC0B */ P_MTU0.TIORL_0.BYTE = 0x00; /* output compare TIOC0C */ P_MTU1.TIOR_1.BYTE = 0xff; /* input capture TIOC1A,B */ P_MTU1.TIER_1.BYTE = 0x73; /* enable TGIA,TGIB,TCIU,TCIV */ P_MTU0.TGRC_0 = msr_tim2; /* set position cycle */ P_MTU0.TGRA_0 = msr_tim1; /* set speed cycle */ INTC.IPRD.WORD = 0x00ff; /* set interrupt level=15 */ P_PORTA.PACRL2.WORD = 0x5000; /* TCLKA,TCLKB select */ P_MTU1.TMDR_1.BYTE = 0x04; /* set phase counting model */ </pre>				

2相エンコーダカウント	MCU	SH7144/45	使用機能	MTU (位相計数モード)
プログラムリスト	<pre data-bbox="189 282 934 426"> P_MTU34.TSTR.BYTE = 0x03; /* start MTU/ch0,1 */ set_imask(0x0); /* set imask level=0 */ while(1); /* loop */ } </pre>			

2相エンコーダカウンタ	MCU	SH7144/45	使用機能	MTU (位相計数モード)
プログラムリスト				
<pre> void phacnt1(void) { P_MTU1.TSR_1.BYTE &= 0xfe; /* clear flag */ wrk = P_MTU1.TGRA_1; if(cnt < 0) /* down count? */ cnt_data1 = (unsigned long)wrk-0x010000+cnt*0x010000; /* set sp */ else cnt_data1 = (unsigned long)wrk+cnt*0x010000; /* set sp */ p_cycle = P_MTU0.TGRB_0; /* set width pulse */ } void phacnt2(void) { P_MTU1.TSR_1.BYTE &= 0xfd; /* clear flag */ wrk = P_MTU1.TGRB_1; if(cnt < 0) /* down count? */ cnt_data2 = (unsigned long)wrk+cnt*0x010000; /* set po */ else cnt_data2 = (unsigned long)wrk-0x010000+cnt*0x010000; /* set po */ } void ovf1(void) { P_MTU1.TSR_1.BYTE &= 0xef; /* clear flag */ cnt++; /* count up */ } void unfl(void) { P_MTU1.TSR_1.BYTE &= 0xdf; /* clear flag */ cnt--; /* count down */ } </pre>				

外部トリガによるタイマ波形の遮断仕様	MCU	SH7144/45	使用機能	MTU、POE
--------------------	-----	-----------	------	---------

- (1) 図2.27に示すように、外部信号の立ち下がりエッジに同期してタイマの出力波形をハイ・インピーダンス状態とすることで波形の遮断を行います。

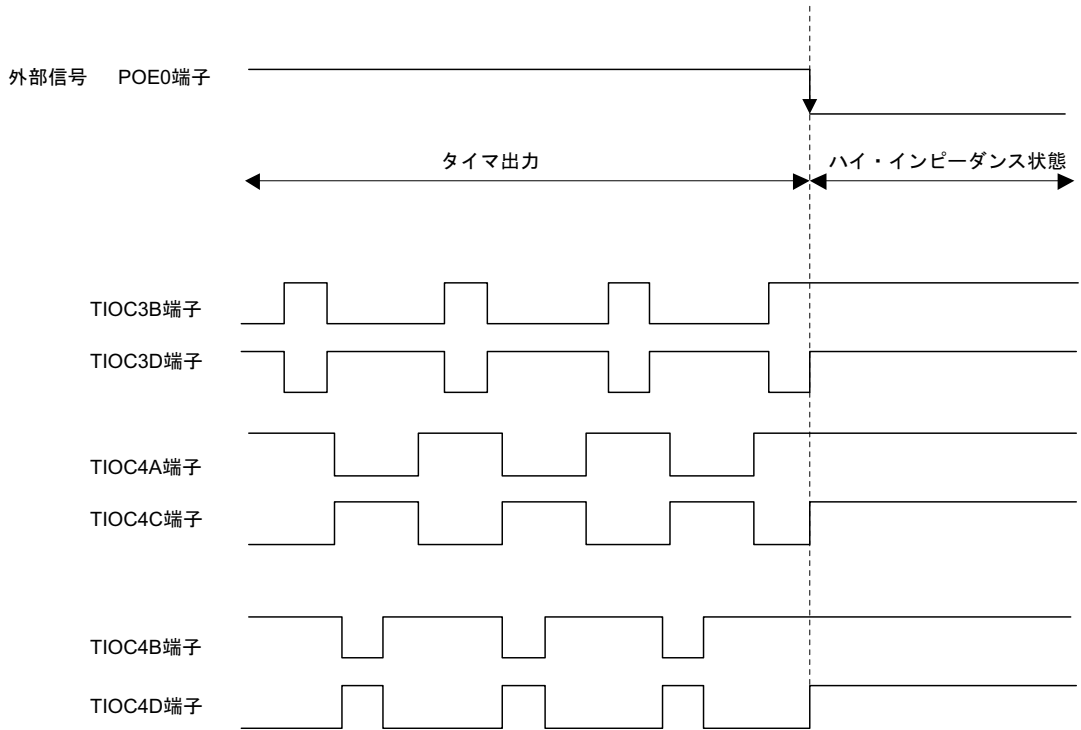


図 2.27 外部トリガによる波形の遮断例

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
使用機能説明				

- (1) 本タスク例ではMTUのch3/4 (リセット同期PWMモード) で出力している波形を外部信号立ち下がりエッジに同期して、ハイ・インピーダンス状態にすることで波形出力を遮断します。
- (a) 図2.28にMTU/ch3、4、POEブロック図を示します。

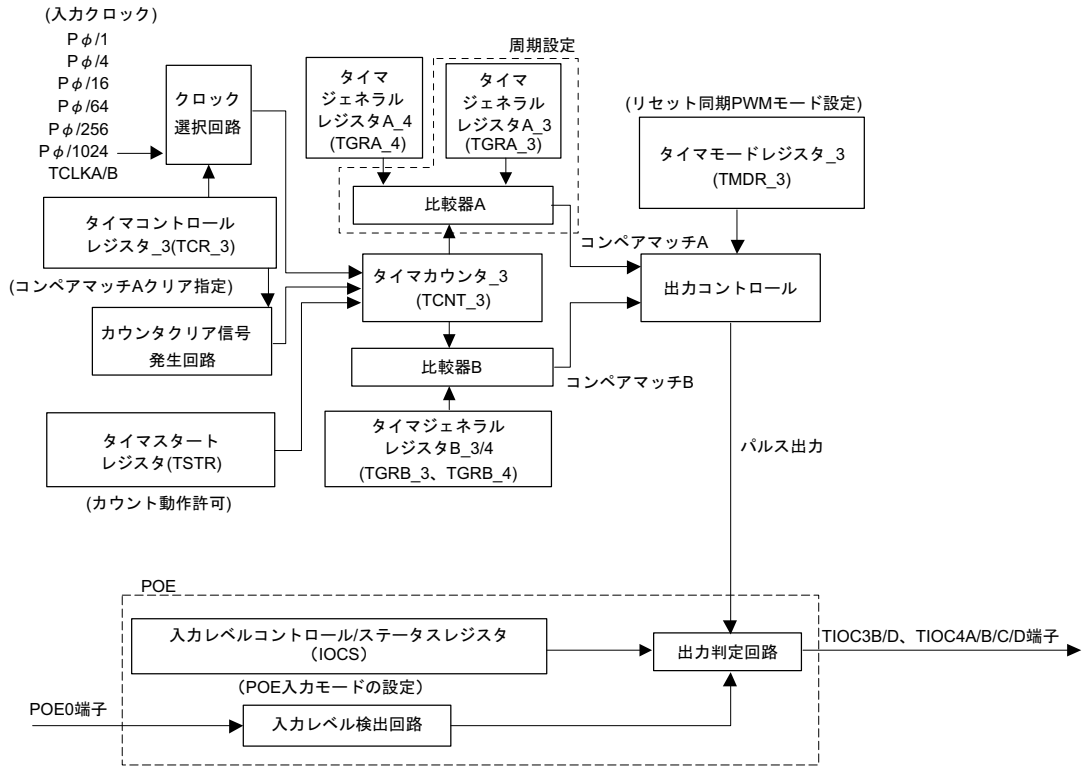


図 2.28 MTU/ch3、4、POE ブロック図

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
------------------	-----	-----------	------	---------

使用機能説明

- (2) 表2.8に本タスクの機能割り付けを示します。MTU、POEの機能を割り付け、波形の遮断を行います。

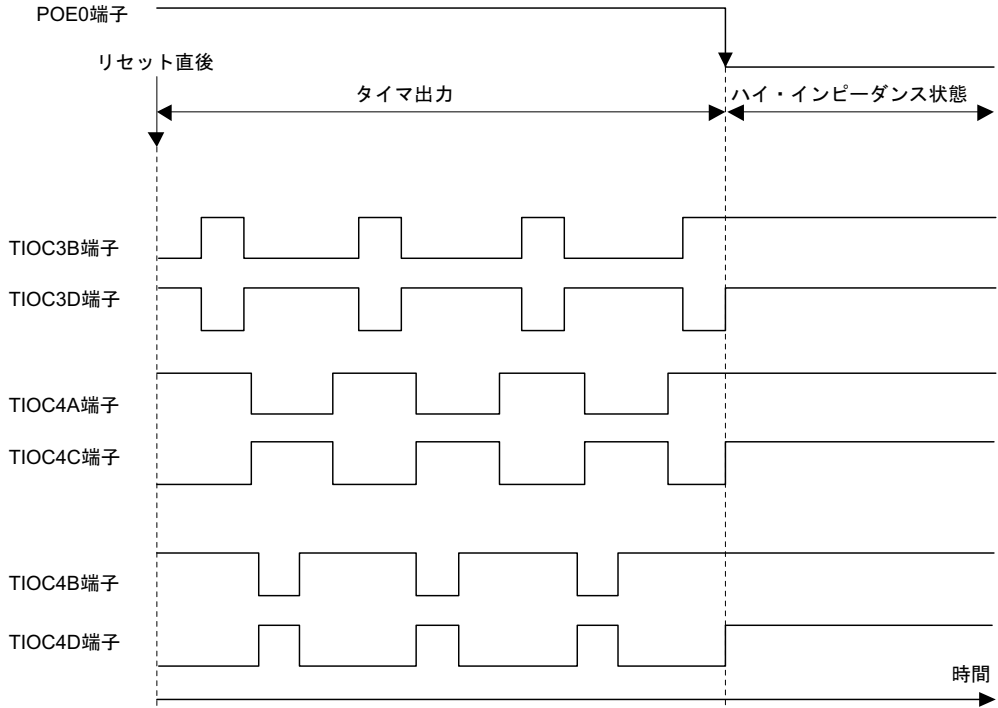
表 2.8 機能割り付け

端子、レジスタ名称	機能割り付け
TIOC3B	パルス出力端子
TIOC3D	
TIOC4A	
TIOC4B	
TIOC4C	
TIOC4D	
POE0	
TSTR_3	ch3 のタイマカウンタ動作の許可/禁止を設定
TCR_3	ch3 のタイマカウンタのクリア要因と入カロックの選択
TMDR_3	ch3、4 をリセット同期 PWM モードに設定
TGRA_3	PWM 周期の設定
TGRB_3	PWM デューティの設定
TGRA_4	
TGRB_4	
TOER	TIOC3B/D、TIOC4A/B/C/D 端子のタイマ出力の許可/禁止を設定
ICSR	POE 入力モードの選択

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
------------------	-----	-----------	------	---------

動作説明

- (1) 図2.29に動作原理を示します。波形の遮断をハードウェアで自動的にを行います。(リセット同期PWMの動作原理については本アプリケーションノート2章5項の正相、逆相PWM3相出力を参考にしてください。)



ハードウェア処理

処理なし

ソフトウェア処理

初期設定

- (1)カウンタクリア要因をTGRA_3のコンペアマッチに設定
- (2)リセット同期PWMモードに設定
- (3)TGRA_3にPWMキャリア周期、TGRB_3、TGRA_4、TGRB_4にデューティを設定
- (4)TIOC3B/D、TIOC4A/B/C/D端子の出力を許可
- (5)POE0の立ち下がり波形遮断に設定
- (6)カウント動作開始

ハードウェア処理

- (1)TIOC3B/D、TIOC4A/B/C/D端子からハイ・インピーダンス出力

ソフトウェア処理

無し

図 2.29 外部トリガによるタイマ波形の遮断動作原理

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	down	DC モータ制御用波形の生成

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
cycle	PWM 周期を設定	1ワード	メインルーチン	入力
duk1	TIOC3B/D から出力させる波形変化タイミングを設定			
duk2	TIOC4A/C から出力させる波形変化タイミングを設定			
duk3	TIOC4B/D から出力させる波形変化タイミングを設定			

(3) 使用内部レジスタ説明

レジスタ名	機能割り付け	アドレス	設定値
P_STBY.MSTCR2	MTU のモジュールスタンバイモードをクリア	H'FFFF861E	H'd0fd
P_PORTE.PEIORL	TIOC3B/D、TIOC4A/B/C/D 端子を出力に設定	H'FFFF83B4	H'fa00
P_PORTE.PECLR1	TIOC3B/D、TIOC4A/B/C/D 端子を MTU 出力に設定	H'FFFF83B8	H'5544
P_PORTB.PBCR2	PB2 を POE0 端子に設定	H'FFFF839A	H'0020
P_MTU34.TCR_3	タイマカウンタのクリア要因と入力クロックの選択	H'FFFF8200	H'22
P_MTU34.TOCR	PWM 周期に同期したトルク出力の許可と正相、逆相の出力レベルの設定	H'FFFF820B	H'43
P_MTU34.TGRA_3	PWM 周期を設定	H'FFFF8218	cycle
P_MTU34.TGRB_3	TIOC3B、TIOC3D から出力させる波形変化タイミングを設定	H'FFFF821A	duk1
P_MTU34.TGRA_4	TIOC4A、TIOC4C から出力させる波形変化タイミングを設定	H'FFFF821C	duk2
P_MTU34.TGRB_4	TIOC4B、TIOC4D から出力させる波形変化タイミングを設定	H'FFFF821E	duk3
P_MTU34.TOER	TIOC3B/D、TIOC4A/B/C/D 端子を MTU 出力に設定	H'FFFF820A	H'ff
P_MTU34.TMDR_3	リセット同期 PWM モードに設定	H'FFFF8202	H'c8
P_MTU.ICSR1	POE0 端子入力信号の立ち下がりがエッジに同期してハイ・インピーダンス出力に設定	H'FFFF83C0	H'0000

(4) 使用 RAM 説明

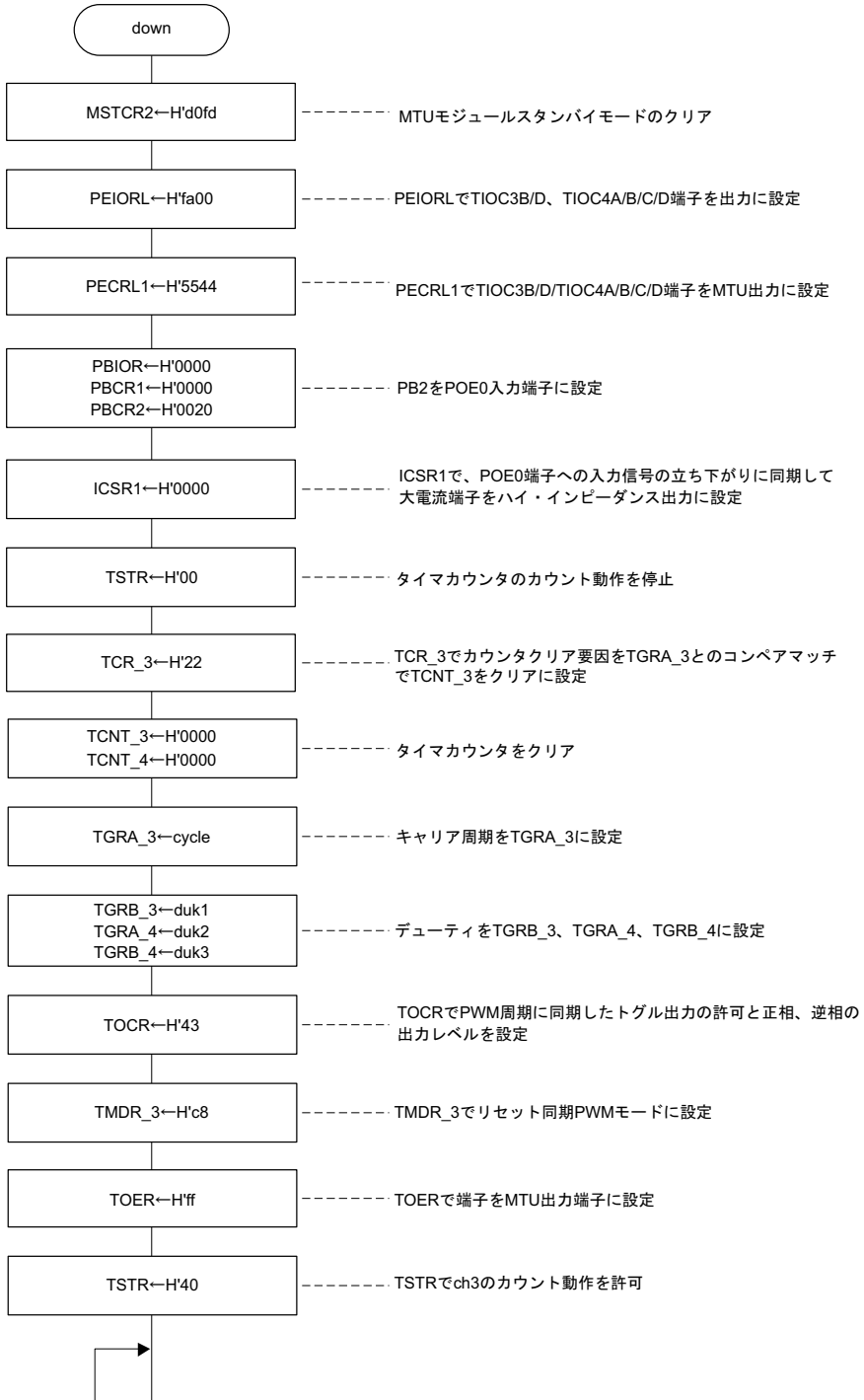
本タスク例では引数以外の RAM は使用していません。

注) レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています。

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
------------------	-----	-----------	------	---------

フローチャート

(1) メインルーチン



外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
プログラムリスト				
<pre> /*-----*/ /* INCLUDE FILE */ /*-----*/ #include <machine.h> #include "iodefine_7145F.h" /*-----*/ /* PROTOTYPE */ /*-----*/ void down(void); /*-----*/ /* RAM ALLOCATION */ /*-----*/ #define cycle (*(unsigned short *)0xffffe000) #define duk1 (*(unsigned short *)0xffffe002) #define duk2 (*(unsigned short *)0xffffe004) #define duk3 (*(unsigned short *)0xffffe006) /*-----*/ /* MAIN PROGRAM */ /*-----*/ void down(void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* MTU module standby mode clear*/ P_PORTE.PEIORL.WORD = 0xfa00; /* TIOC3B/D,TIOC4A/B/C/D=output*/ P_PORTE.PECRL1.WORD = 0x5544; /* TIOC3B/D,TIOC4A/B/C/D=output*/ P_PORTB.PBIOR.WORD = 0x0000; /* POE0 enable*/ P_PORTB.PBCR1.WORD = 0x0000; P_PORTB.PBCR2.WORD = 0x0020; P_MTU.ICSRL.WORD = 0x0000; /* stop timer POE0 falling edge*/ P_MTU.OCSR.WORD = 0x0000; P_MTU34.TSTR.BYTE = 0x00; P_MTU34.TCR_3.BYTE = 0x22; /* timer clear input capture TGRA_3*/ P_MTU34.TCNT_3 = 0x0000; /* set timer counter 0x0000*/ P_MTU34.TCNT_4 = 0x0000; P_MTU34.TGRA_3 = cycle; /* set PWM period */ P_MTU34.TGRB_3 = duk1; /* set duty */ P_MTU34.TGRA_4 = duk2; P_MTU34.TGRB_4 = duk3; </pre>				

外部トリガによるタイマ波形の遮断	MCU	SH7144/45	使用機能	MTU、POE
プログラムリスト	<pre data-bbox="193 282 1061 504"> P_MTU34.TOCR.BYTE = 0x43; /* set output level*/ P_MTU34.TMDR_3.BYTE = 0xc8; /* Reset-synchoronized PWM mode*/ P_MTU34.TOER.BYTE = 0xff; /* set timer3,4 output*/ P_MTU34.TSTR.BYTE = 0x40; /* start timer3*/ while(1); /* loop*/ } </pre>			

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
仕様				

- (1) 図2.30に示すように、DCブラシレスモータ制御に必要な波形を出力します。出力波形は、各端子のゲート信号とリセット同期PWM出力をチョッピングして出力します。

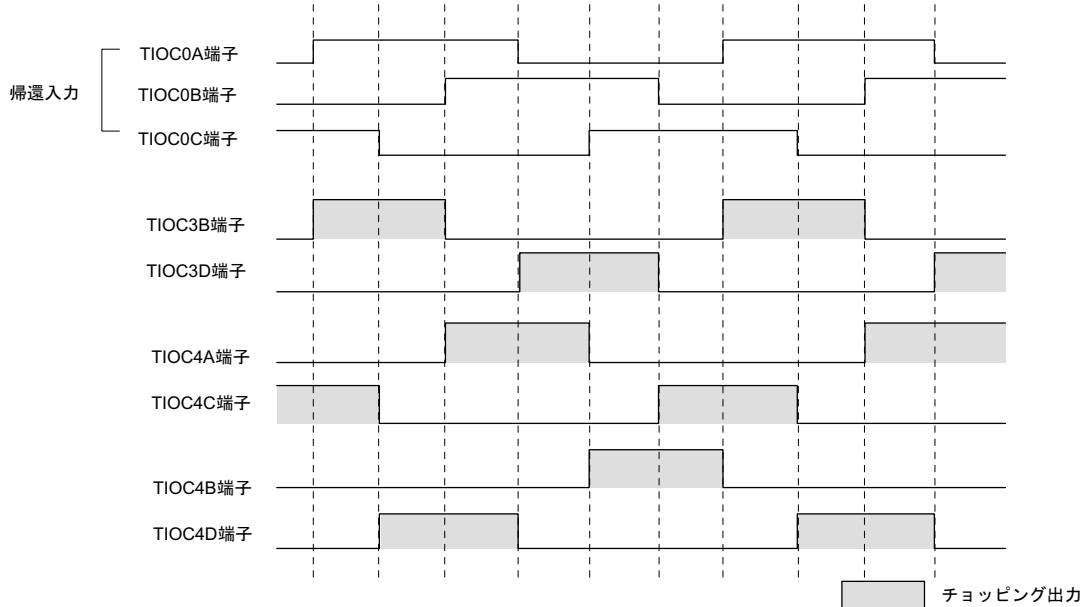


図 2.30 DC ブラシレスモータ制御用信号出力例

DC モーター制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
使用機能説明				

- (1) 本タスク例ではMTUのチャネル3、4を組み合わせ使用し、一方の変化点が共通で、正相、逆相の関係にあるPWM波形を3相出力します。生成した波形と帰還入力から生成したゲート信号をチョッピングして出力します。

(a) 図2.31に本タスク例で使用するMTUのブロック図を示します。

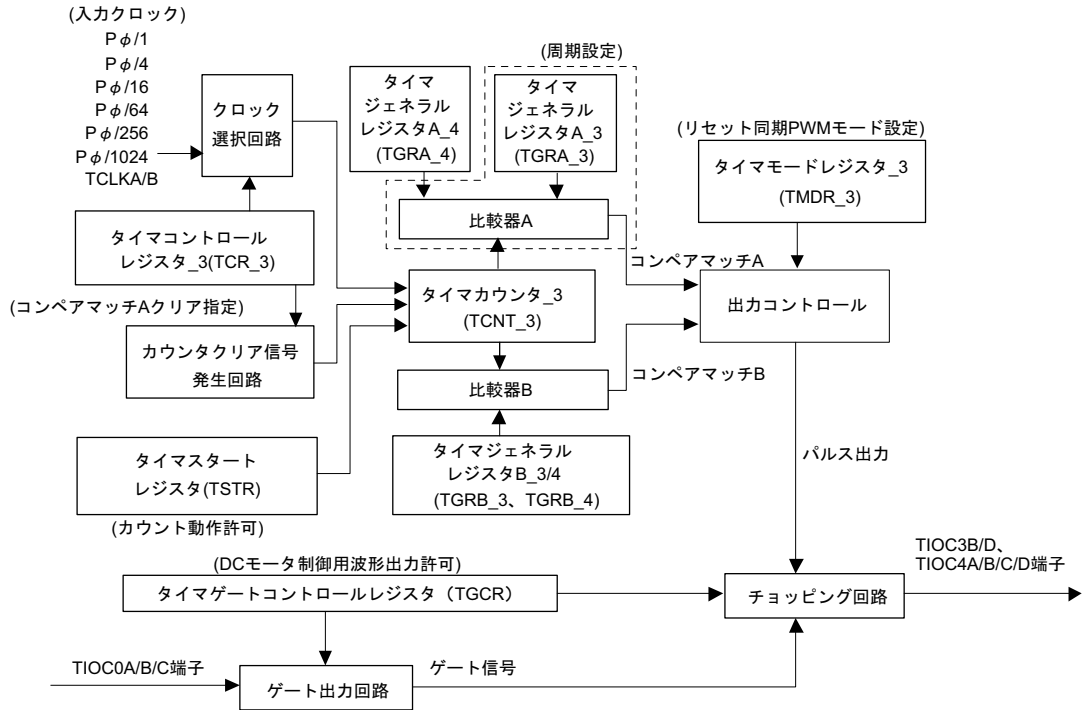


図 2.31 MTU/ch3、4 ブロック図

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
---------------	-----	-----------	------	---------------------

使用機能説明

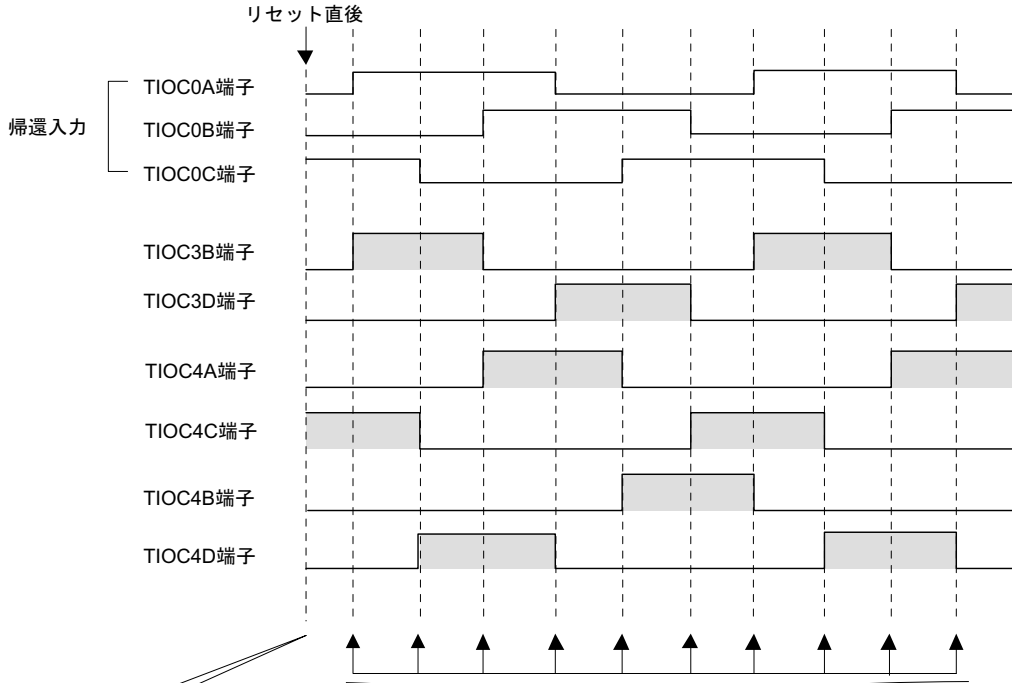
- (2) 表2.9に本タスクの機能割り付けを示します。MTUの機能を割り付け、DCモータ制御用波形の出力を行います。

表 2.9 機能割り付け

端子、レジスタ名称	機能割り付け
TIOC3B TIOC3D TIOC4A TIOC4B TIOC4C TIOC4D	PWM パルス出力端子
TIOC0A TIOC0B TIOC0C	帰還信号入力端子
TSTR	ch3、4のタイマカウンタ動作の許可/禁止
TCR_3	ch3のタイマカウンタのクリア要因と入カクロックを選択
TMDR_3	ch3、4をリセット同期PWMモードとして動作
TGRA_3	PWM周期の設定
TGRB_3 TGRC_3 TGRD_3 TGRA_4 TGRB_4	出力させる波形変化タイミングの設定
TOER	TIOC3B/D、TIOC4A/B/C/D端子のタイマ出力の許可/禁止
TGCR	DCモータ制御波形の出力の許可/禁止

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
動作説明				

- (1) 図2.32に動作原理を示します。DCモータ制御用波形の出力をハードウェアで自動的に行います。(リセット同期PWMの動作原理については本アプリケーションノート2章5項の正相、逆相PWM3相出力を参照してください。)



ハードウェア処理
処理なし
ソフトウェア処理
初期設定
(1)MTUch3/4をリセット同期PWMモードに設定
(2)TGRA_3にチョッピング周期、TGRB_3、TGRA_4、TGRB_4にPWMキャリア周期を設定
(3)TIOC3B/D、TIOC4A/B/C/D端子からの出力を許可
(4)カウント動作開始を許可

ハードウェア処理
(1)TIOC0A/B/C端子入力信号の立ち上がり/立ち下がりエッジでTIOC3B/D、TIOC4A/B/C/Dの各出力端子にゲート信号を出力する
(2)出力レベルがHigh時リセット同期PWM出力とゲート出力をチョッピング出力する
ソフトウェア処理
処理無し

図 2.32 DC モータ制御用信号出力動作原理

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
ソフトウェア説明				

(1) モジュール説明

モジュール名	ラベル名	機能割り付け
メインルーチン	dc_3	DC モータ制御用波形の生成

(2) 引数の説明

ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
cycle	PWM パルスの周期に相当するタイマ値を設定する	1ワード	メインルーチン	入力
duk1	TIOC3B/3D から出力させる波形変化タイミングを設定			
duk2	TIOC4A/4C から出力させる波形変化タイミングを設定			
duk3	TIOC4B/4D から出力させる波形変化タイミングを設定			

(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
P_STBY.MSTCR2	MTU モジュールスタンバイモードをクリア	H'FFFF861E	H'd0fd
P_PORTE.PEIORL	TIOC3B/3D、TIOC4A/4B/4C/4D 端子を出力に設定	H'FFFF83B4	H'fa00
P_PORTE.PECRL1	TIOC3B/3D、TIOC4A/4B/4C/4D 端子を MTU 入出力端子に設定	H'FFFF83B8	H'5544
P_PORTE.PECRL2	TIOC0A、TIOC0B、TIOC0C 端子を MTU 入力端子に設定	H'FFFF83BA	H'0015
P_MTU34.TCR_3	タイマカウンタのクリア要因と入カクロックを選択	H'FFFF8200	H'22
P_MTU34.TOCR	PWM 周期に同期したトグル出力の許可と正相、逆相の出力レベルの設定	H'FFFF820B	H'03
P_MTU34.TGRA_3	PWM キャリア周期の設定	H'FFFF8218	cycle
P_MTU34.TGRB_3	TIOC3B、TIOC3D から出力させる波形変化タイミングの設定	H'FFFF821A	duk1
P_MTU34.TGRA_4	TIOC4A、TIOC4C から出力させる波形変化タイミングの設定	H'FFFF821C	duk2
P_MTU34.TGRB_4	TIOC4B、TIOC4D から出力させる波形変化タイミングの設定	H'FFFF821E	duk3
P_MTU34.TOER	TIOC3B/3D、TIOC4A/4B/4C/4D 端子を MTU 出力に設定	H'FFFF820A	H'ff
P_MTU34.TMDR_3	リセット同期 PWM モードに設定	H'FFFF8202	H'c8
P_MTU34.TGCR	DC モータ制御用波形出力を許可	H'FFFF820D	H'f0

(4) 使用 RAM 説明

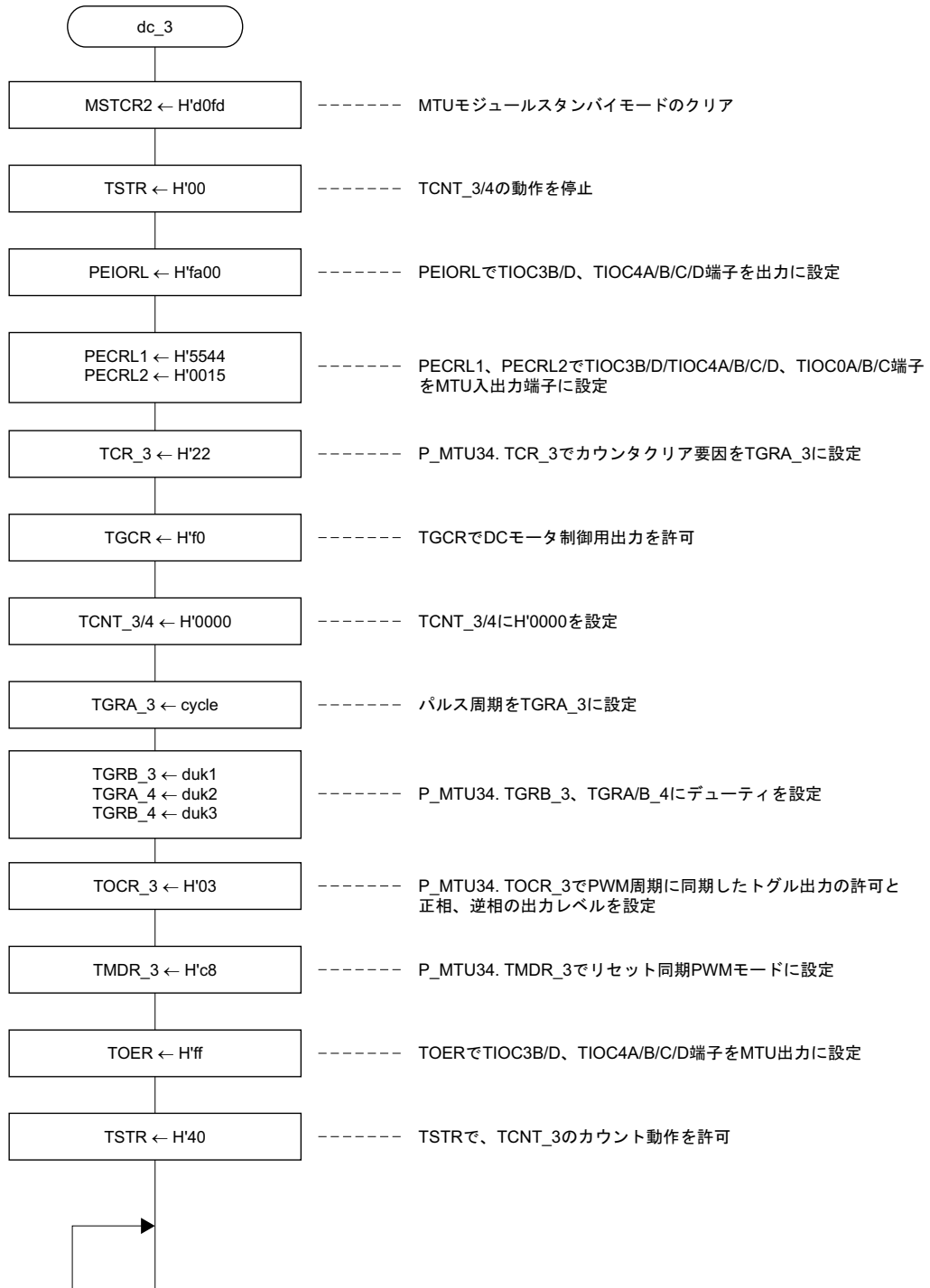
本タスク例では引数以外の RAM は使用していません。

注) レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています。

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
---------------	-----	-----------	------	---------------------

フローチャート

(1) メインルーチン



DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
プログラムリスト				
<pre> /*-----*/ /* INCLUDE FILE */ /*-----*/ #include <machine.h> #include "iodefine_7145F.h" /*-----*/ /* PROTOTYPE */ /*-----*/ void dc_3(void); /*-----*/ /* RAM ALLOCATION */ /*-----*/ #define cycle (*(unsigned short *)0xffffe000) #define duk1 (*(unsigned short *)0xffffe002) #define duk2 (*(unsigned short *)0xffffe004) #define duk3 (*(unsigned short *)0xffffe006) /*-----*/ /* MAIN PROGRAM */ /*-----*/ void dc_3(void) { P_STBY.MSTCR2.WORD = 0xd0fd; /* MTU module standby mode clear */ P_MTU34.TSTR.BYTE = 0x00; /* Stop timer counter */ P_PORTE.PEIORL.WORD = 0xfa00; /* TIOC3B/D,TIOC4A/B/C/D output */ P_PORTE.PECRL1.WORD = 0x5544; /* TIOC3B/D,TIOC4A/B/C/D output */ P_PORTE.PECRL2.WORD = 0x0015; /* TIOC0A/B/C = input */ P_MTU34.TCR_3.BYTE = 0x22; /* Counter clear by input capture TGRA_3 */ P_MTU34.TGCR.BYTE = 0xf0; /* for DC brushless motor control */ P_MTU34.TCNT_3 = 0x0000; /* Timer counter3 = 0 */ P_MTU34.TCNT_4 = 0x0000; /* Timer counter4 = 0 */ P_MTU34.TGRA_3 = cycle; /* Set carrier Period */ P_MTU34.TGRB_3 = duk1; /* Set duty */ P_MTU34.TGRA_4 = duk2; P_MTU34.TGRB_4 = duk3; P_MTU34.TOCR.BYTE = 0x03; /* Set output level*/ P_MTU34.TMDR_3.BYTE = 0xc8; /* Reset-synchronized pwm mode*/ P_MTU34.TOER.BYTE = 0xff; /* Enable timer3/4 output*/ P_MTU34.TSTR.BYTE = 0x40; /* Start timer3 */ </pre>				

DC モータ制御用信号出力	MCU	SH7144/45	使用機能	MTU (ゲートコントロールレジスタ)
プログラムリスト	<pre data-bbox="189 280 768 347"> while(1); /* Loop */ } </pre>			

MTU による A/D 変換の開始仕様	MCU	SH7144/45	使用機能	MTU、A/D 変換器
---------------------	-----	-----------	------	-------------

- (1) 図2.33に示すように、4chの電圧を入力し、A/D変換します。
- (2) A/D変換は1サイクルスキャンモードを使用し、0~3chの連続A/D変換を行います。
- (3) A/D変換器の起動はMTU/ch0のTGRA_0のコンペアマッチで行います。

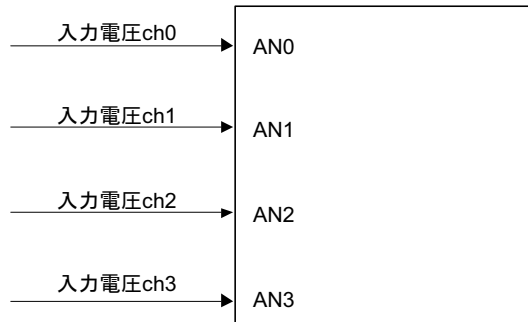
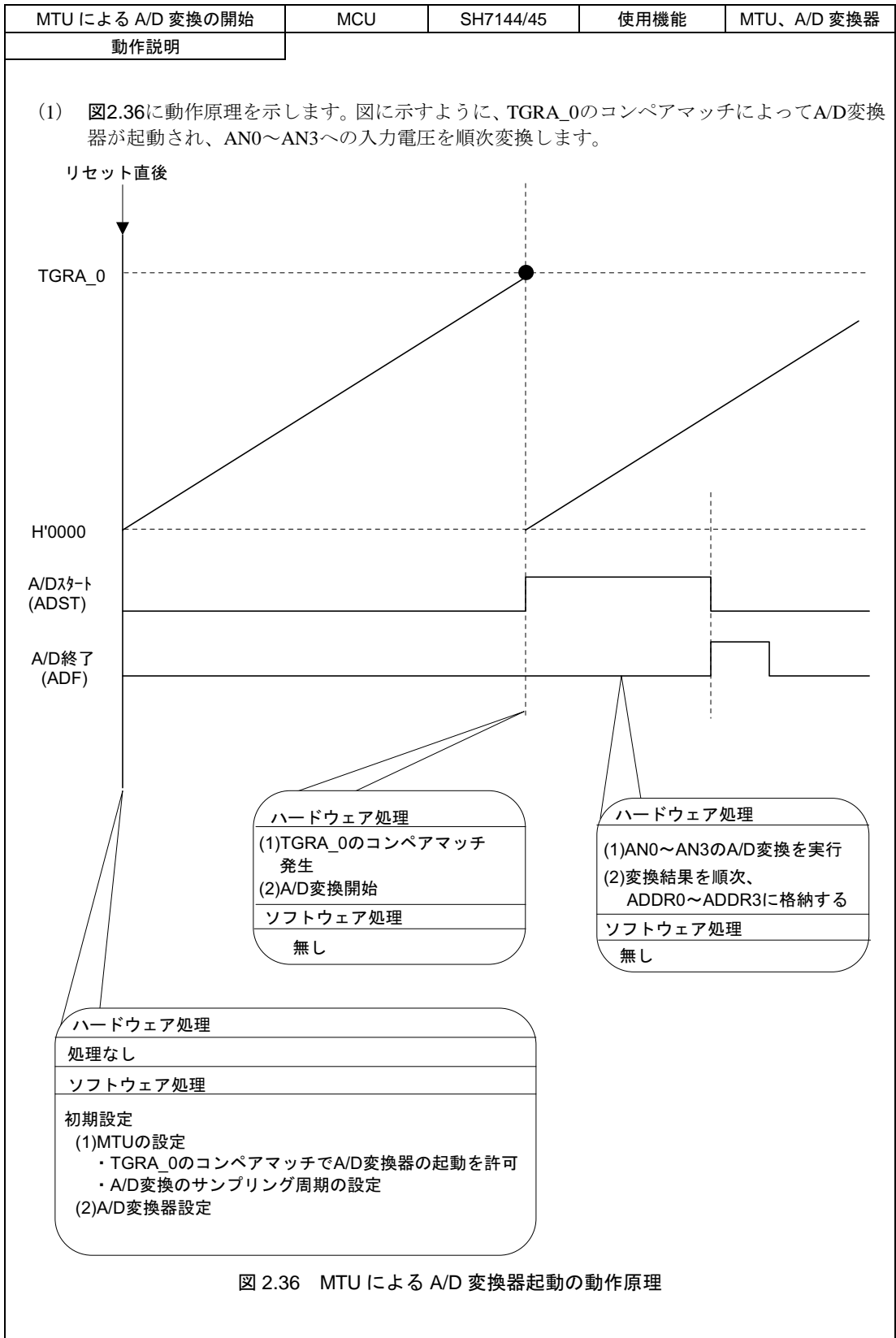


図 2.33 SH7145 による電圧測定ブロック図

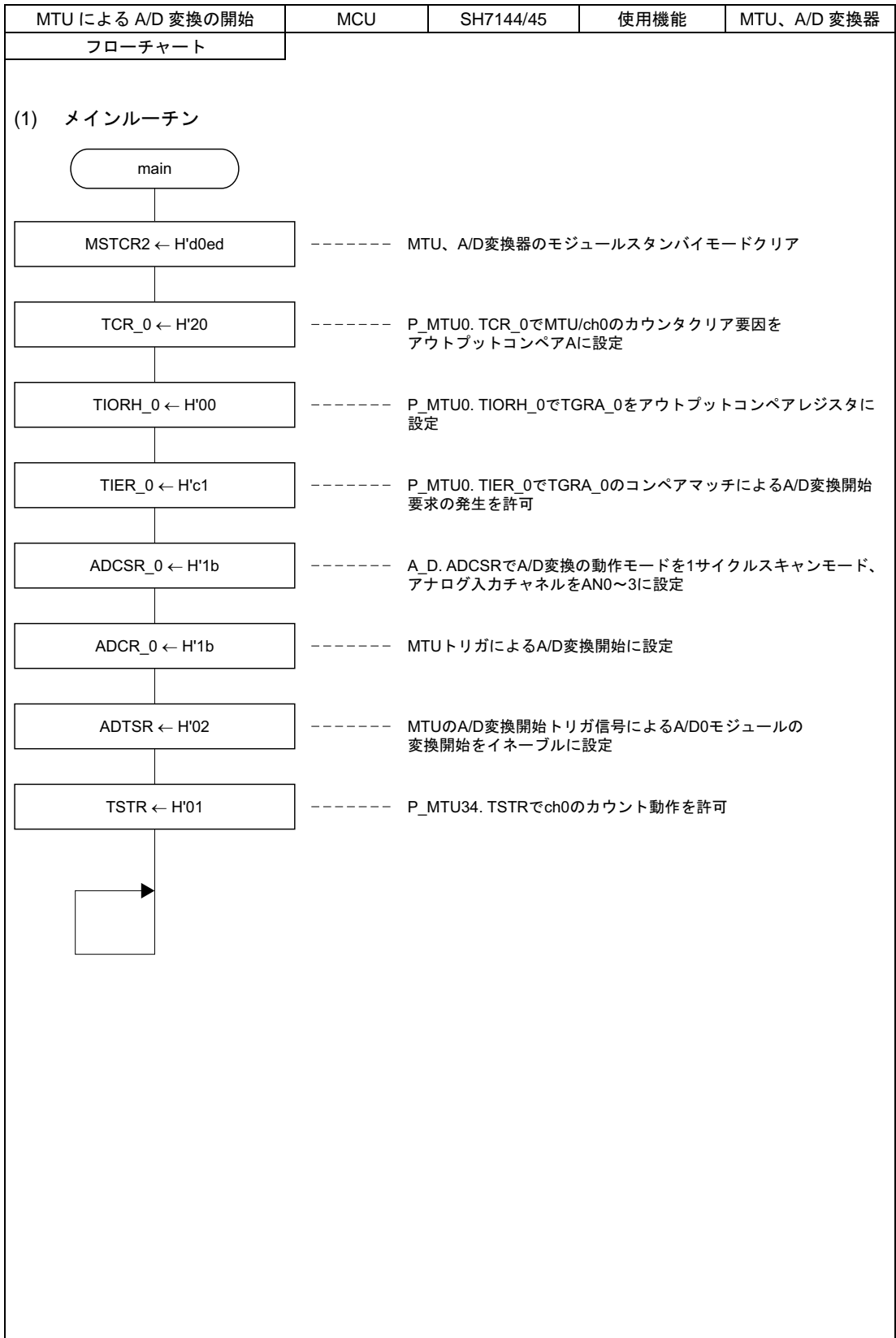
MTU による A/D 変換の開始	MCU	SH7144/45	使用機能	MTU、A/D 変換器
使用機能説明				
<p>(1) 本タスク例では、MTUからのA/D変換開始要求によりA/D変換器を起動します。</p> <p>(a) 図2.34にch0のブロック図を示します。ch0では、以下の機能を使用してA/D変換器を起動します。</p> <ul style="list-style-type: none"> ソフトウェアを介さずMTUのコンペアマッチでA/D変換を開始する機能。 ソフトウェアを介さずハードウェアで自動的にパルスを出力する機能。(アウトプットコンペア) <div data-bbox="148 465 1218 1058" style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;">MTU/ch0</p> <pre> graph TD subgraph MTU_ch0 [MTU/ch0] TIER_0[Tайминтабруйтинеэбулрегистр_0 (TIER_0)] TIORH_0[Tайми/OконтрорулрегистрH_0 (TIORH_0)] TGRA_0[TаймдженералулрегистрA_0 (TGRA_0)] TCNT_0[Tаймакаунта_0(TCNT_0)] Comp[比較器] OutComp[アウトプットコンペア信号発生回路] CtrlLogic[制御ロジック] TIER_0 --> CtrlLogic TIORH_0 --> CtrlLogic TGRA_0 --> Comp TCNT_0 --> Comp Comp --> OutComp OutComp --> CtrlLogic end A_D[A/D変換器] CtrlLogic -- "(A/D変換開始トリガ)" --> A_D </pre> </div>				
図 2.34 SH7145 による ch0 のブロック図				

MTU による A/D 変換の開始	MCU	SH7144/45	使用機能	MTU、A/D 変換器
使用機能説明				
<p>(b) 図2.35にA/D変換器のブロック図を示します。A/D変換器では、以下の機能を使用してアナログからデジタルへの変換を行います。</p> <ul style="list-style-type: none"> 指定されたチャンネル (ch0~ch3) の A/D 変換を 1 度行う機能。(1 サイクルスキャンモード) 				
*1 : サンプル&ホールド回路				
図 2.35 SH7145 による電圧測定ブロック図				

MTU による A/D 変換の開始	MCU	SH7144/45	使用機能	MTU、A/D 変換器																
使用機能説明																				
<p>(2) 表2.10に本タスク例の機能割り付けを示します。</p> <p style="text-align: center;">表 2.10 レジスタ機能割り付け</p> <table border="1"> <thead> <tr> <th>レジスタ名</th> <th>機能割り付け</th> </tr> </thead> <tbody> <tr> <td>AN0~AN3</td> <td>アナログ入力端子</td> </tr> <tr> <td>TCR_0</td> <td>カウンタクリア要因の選択</td> </tr> <tr> <td>TIER_0</td> <td>A/D 変換開始要求の発生を許可</td> </tr> <tr> <td>TGRA_0</td> <td>A/D 変換のサンプリング周期の設定</td> </tr> <tr> <td>ADCR</td> <td>A/D 変換のモードおよび測定端子の設定</td> </tr> <tr> <td>ADCSR</td> <td>変換時間および起動要因の選択</td> </tr> <tr> <td>ADDR0~ADDR3</td> <td>A/D 変換結果の格納</td> </tr> </tbody> </table>					レジスタ名	機能割り付け	AN0~AN3	アナログ入力端子	TCR_0	カウンタクリア要因の選択	TIER_0	A/D 変換開始要求の発生を許可	TGRA_0	A/D 変換のサンプリング周期の設定	ADCR	A/D 変換のモードおよび測定端子の設定	ADCSR	変換時間および起動要因の選択	ADDR0~ADDR3	A/D 変換結果の格納
レジスタ名	機能割り付け																			
AN0~AN3	アナログ入力端子																			
TCR_0	カウンタクリア要因の選択																			
TIER_0	A/D 変換開始要求の発生を許可																			
TGRA_0	A/D 変換のサンプリング周期の設定																			
ADCR	A/D 変換のモードおよび測定端子の設定																			
ADCSR	変換時間および起動要因の選択																			
ADDR0~ADDR3	A/D 変換結果の格納																			



MTU による A/D 変換の開始	MCU	SH7144/45	使用機能	MTU、A/D 変換器
ソフトウェア説明				
(1) モジュール説明				
モジュール名	ラベル名	機能割り付け		
メインルーチン	main	MTU による A/D 変換器の起動		
(2) 使用内部レジスタ説明				
レジスタ名	機能	アドレス	設定値	
P_STBY.MSTCR2	モジュールスタンバイモードのクリア(MTU、A/D 変換器)	H'FFFF861E	H'd0ed	
P_MTU0.TCR_0	TCNT_0 のカウンタクロックの選択、および TCNT_0 カウンタクリア要因をアウトプットコンペア A に設定	H'FFFF8260	H'20	
P_MTU0.TIORH_0	TGRA_0 をアウトプットコンペアに設定	H'FFFF8262	H'00	
P_MTU0.TIER_0	MTU からの A/D 変換開始要求の発生を許可	H'FFFF8264	H'c1	
P_MTU0.TGRA_0	A/D 変換サンプリング周期を 1ms に設定	H'FFFF8268	H'9c40	
P_AD.ADCR_0	A/D 変換モード(1 サイクルスキャンモード) 起動要因を MTU の変換開始トリガに設定	H'FFFF8488	H'87	
P_AD.ADCSR_0	A/D 変換チャンネル(AN0~AN3)、変換時間の設定	H'FFFF8480	H'1b	
P_AD.ADTSR	MTU の A/D 変換開始トリガ信号により、A/D0 モジュール変換開始を許可に設定	H'FFFF87F4	H'02	
(3) 使用 RAM 説明				
本タスク例では引数以外の RAM は使用していません。				
注) レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています。				



MTUによるA/D変換の開始	MCU	SH7144/45	使用機能	MTU、A/D変換器
プログラムリスト	<pre> /***** /* /* INCLUDE FILE /* /***** #include <machine.h> #include "iodefine_7145F.h" /***** /* /* PROTOTYPE /* /***** void main(void); /***** /* /* MAIN PROGRAM /* /***** void main(void) { P_STBY.MSTCR2.WORD = 0xd0ed; /* Clear Module standby mode */ P_MTU0.TCR_0.BYTE = 0x20; /* clock=Pφ/1,counter clear TGRA_0 */ P_MTU0.TIORH_0.BYTE = 0x00; P_MTU0.TIER_0.BYTE = 0xc1; /* enable TGIA */ P_MTU0.TCNT_0 = 0x0000; P_MTU0.TGRA_0 = 0x9c40; /* A/D sampling period = 1.0ms */ P_AD.ADCR_0.BYTE = 0x87; /* 1cycle scan mode */ P_AD.ADCSR_0.BYTE = 0x1b; /* 4channel scan mode */ P_AD.ADTSR.BYTE = 0x02; P_MTU34.TSTR.BYTE = 0x01; /* Start timer counter */ set_imask(0x0); while(1); } </pre>			

DMAC を使用した RAM モニタ 仕様	MCU	SH7144/45	使用機能	SCI, DMAC
--------------------------	-----	-----------	------	-----------

- (1) 図2.37に示すように、SH7145のSCIを調歩同期式モードで使用し、コンソールから送信されたRAM参照アドレス（4バイトデータ）を受信し、その内容をRAM上からとり出しSCIでコンソールに送信します。
- (2) 転送プロトコルは19200bps、8ビットデータ、1ストップビットおよびノンパリティとします。
- (3) RDRからRAMへのデータ転送は、図2.38に示すようにDMACの直接アドレスモードを使用し、RDRの受信データをRAM上に格納します。
- (4) RAMからTDRへのデータ転送は、図2.39に示すようにDMACの間接アドレスモードを使用し、以下のように行います。
 - (a) RAM上に格納されたデータをDMAC内のテンポラリバッファに格納し、それをアドレスとしてRAM上からデータを取り出します。
 - (b) 取り出したデータをTDRにバイト単位で順次転送します。
- (5) DMACの転送条件は表2.11、表2.12に示す通りです。

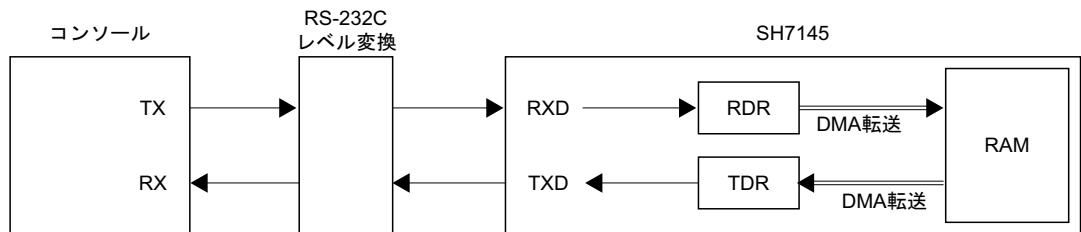


図 2.37 SH7145 による RAM 上データの SCI 転送ブロック図

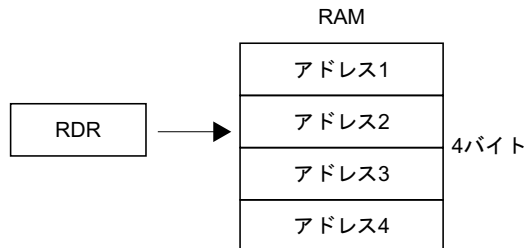


図 2.38 DMAC を用いたデータ転送（転送元直接アドレス）

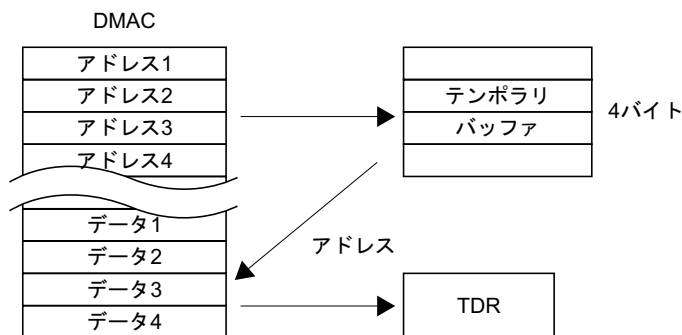


図 2.39 DMAC を用いたデータ転送（転送元間接アドレス）

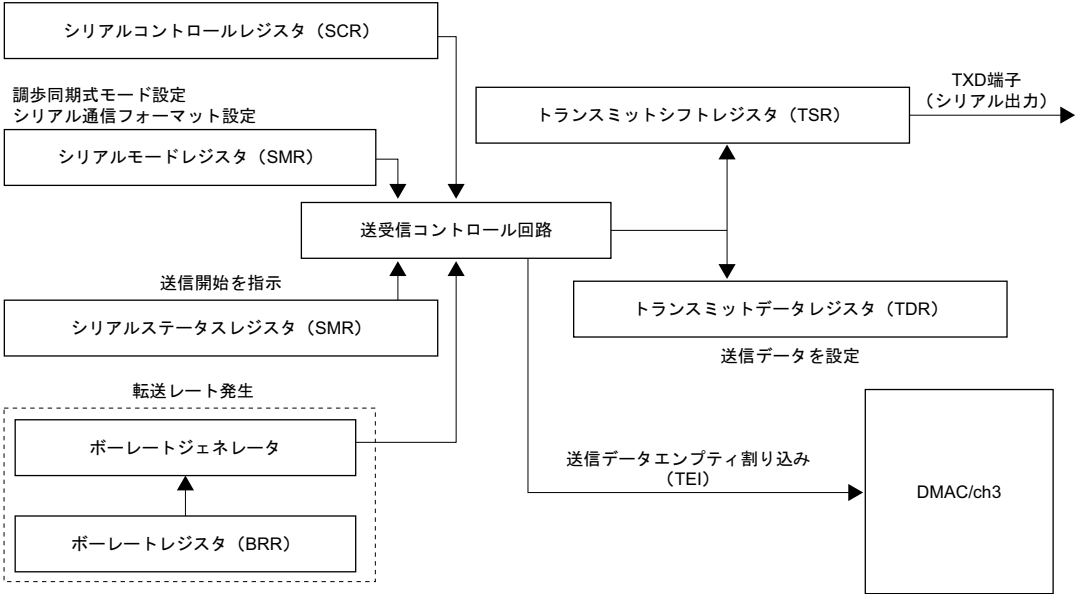
DMAC を使用した RAM モニタ 仕様	MCU	SH7144/45	使用機能	SCI、DMAC
--------------------------	-----	-----------	------	----------

表 2.11 SCI 受信時の DMAC 転送条件(RDR→RAM)

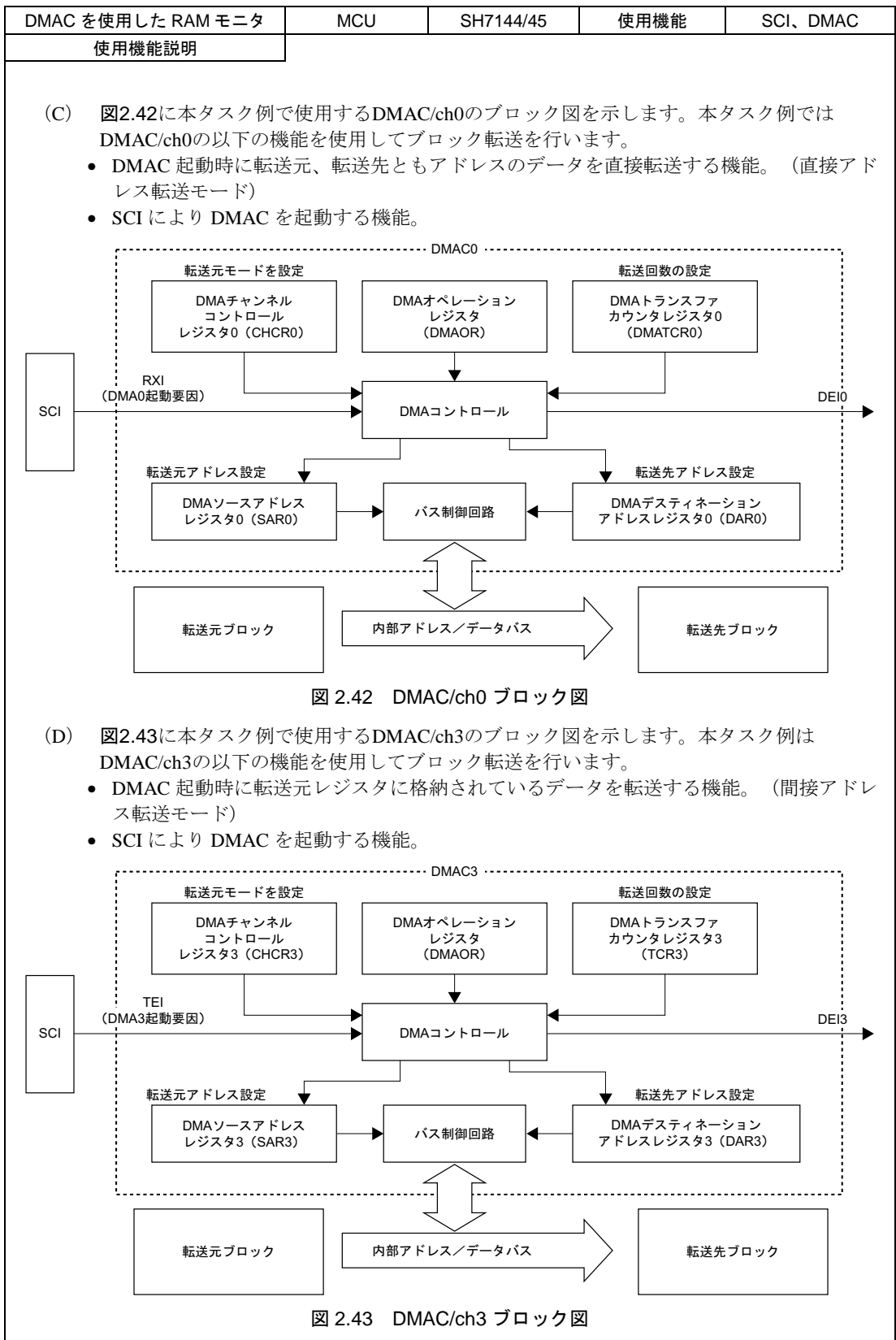
条件項目	内容
DMAC チャンネル	チャンネル 0
転送元	内蔵 SCI チャンネル 0
転送先	内蔵 RAM
転送回数	4 回
転送元アドレス	固定
転送先アドレス	増加
転送要求元	SCI チャンネル 0
バスモード	サイクルスチール
転送単位	バイト

表 2.12 SCI 受信時の DMAC 転送条件 (RAM→TDR)

条件項目	内容
DMAC チャンネル	チャンネル 3
転送元	内蔵 RAM
転送先	内蔵 SCI チャンネル 0
転送回数	1 回
転送元アドレス	増加
転送先アドレス	固定
転送要求元	SCI チャンネル 0
バスモード	サイクルスチール
転送単位	バイト

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
使用機能説明				
<p>本タスク例では、RAM モニタを SCI と DMAC を使用して行います。</p> <p>(a) 図2.40にSCIの送信ブロック図を示します。本タスクではSCIの以下の機能を使用してコンソールデータの送信を行います。</p> <ul style="list-style-type: none"> • キャラクタ単位で同期をとる調歩同期方式で、データの通信を行う機能。(調歩同期式モード) • 送信完了時に割り込みを発生する機能。(TEI 割り込み) <p>割り込みの設定 送信モード設定</p>  <p>図2.40は、SCI送信のハードウェアブロック図を示しています。左側のレジスタ群には、シリアルコントロールレジスタ (SCR)、シリアルモードレジスタ (SMR)、シリアルステータスレジスタ (SMR)、ポーレートジェネレータ、およびポーレートレジスタ (BRR) が含まれます。SCRとSMRは送受信コントロール回路に接続されています。SMRは送信開始を指示し、BRRはポーレートジェネレータを通じて転送レート発生を制御します。右側のレジスタ群には、トランスミットシフトレジスタ (TSR)、トランスミットデータレジスタ (TDR)、およびDMAC/ch3が含まれます。TSRはTXD端子 (シリアル出力) に接続されています。TDRは送信データを設定し、DMAC/ch3は送信データエンプティ割り込み (TEI) を発生させます。送受信コントロール回路はTSRとTDRの両方と接続されています。</p>				
図 2.40 SCI 送信ブロック図				

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
使用機能説明				
<p>(b) 図2.41に本タスク例で使用するSCIの受信ブロック図を示します。本タスク例では図2.41に示すSCIの以下の機能を使用してコンソールからのデータを受信します。</p> <ul style="list-style-type: none"> • キャラクタ単位で同期をとる調歩同期方式でデータ通信を行う機能。(調歩同期式モード) • 受信完了時に割り込みを発生する機能。(RXI 割り込み) 				
<p style="text-align: center;">(調歩同期式モード) 割り込みの設定 受信モード設定</p> <p style="text-align: center;">シリアルコントロールレジスタ (SCR)</p> <p style="text-align: center;">調歩同期式モード設定 シリアル通信フォーマット設定</p> <p style="text-align: center;">レシーブモードレジスタ (SMR)</p> <p style="text-align: center;">受信データを格納</p> <p style="text-align: center;">レシーブデータレジスタ (RDR)</p> <p style="text-align: center;">RXD端子 (シリアル入力)</p> <p style="text-align: center;">レシーブシフトレジスタ (RSR)</p> <p style="text-align: center;">送受信コントロール回路</p> <p style="text-align: center;">受信データフル割り込み (RXI)</p> <p style="text-align: center;">DMAC/ch0</p> <p style="text-align: center;">転送レート発生</p> <p style="text-align: center;">ボーレートジェネレータ</p> <p style="text-align: center;">ボーレートレジスタ (BRR)</p>				
<p>図 2.41 SCI 受信ブロック図</p>				



DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
使用機能説明				
<p>表 2.13 に本タスク例の機能割り付けを示します。DMAC および SCI の機能を割り付け、SCI によるデータ転送の送受信を行います。</p>				
表 2.13 機能割り付け				
端子、レジスタ名	機能割り付け			
SAR0	転送元アドレスの設定			
SAR3				
DAR0	転送先アドレスの設定			
DAR3				
TCR0	転送回数の設定			
TCR3				
CHCR0	DMAC の動作モード、転送方法等の設定			
CHCR3				
DMA0R	DMAC の実行するチャンネル優先順位の設定			
RXD	データ受信端子			
TXD	データ送信端子			
SMR	SCI の送信フォーマットの設定			
SCR	SCI 割り込みの許可／禁止の設定			
SSR	割り込みステータスの設定			
RDR	コンソールからの受信データを格納			
TDR	コンソールへ送信データを転送			
BRR	転送レートの設定			

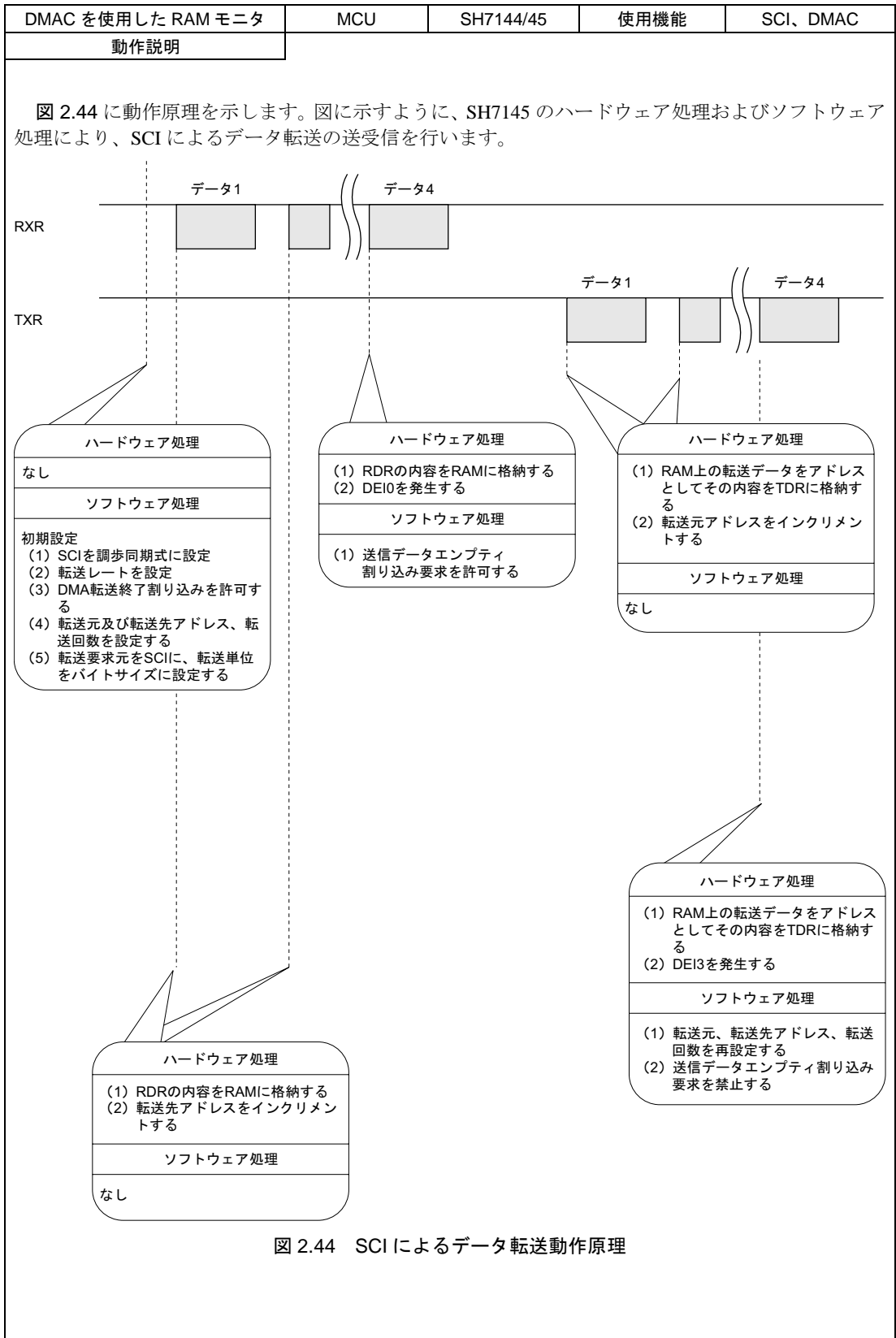
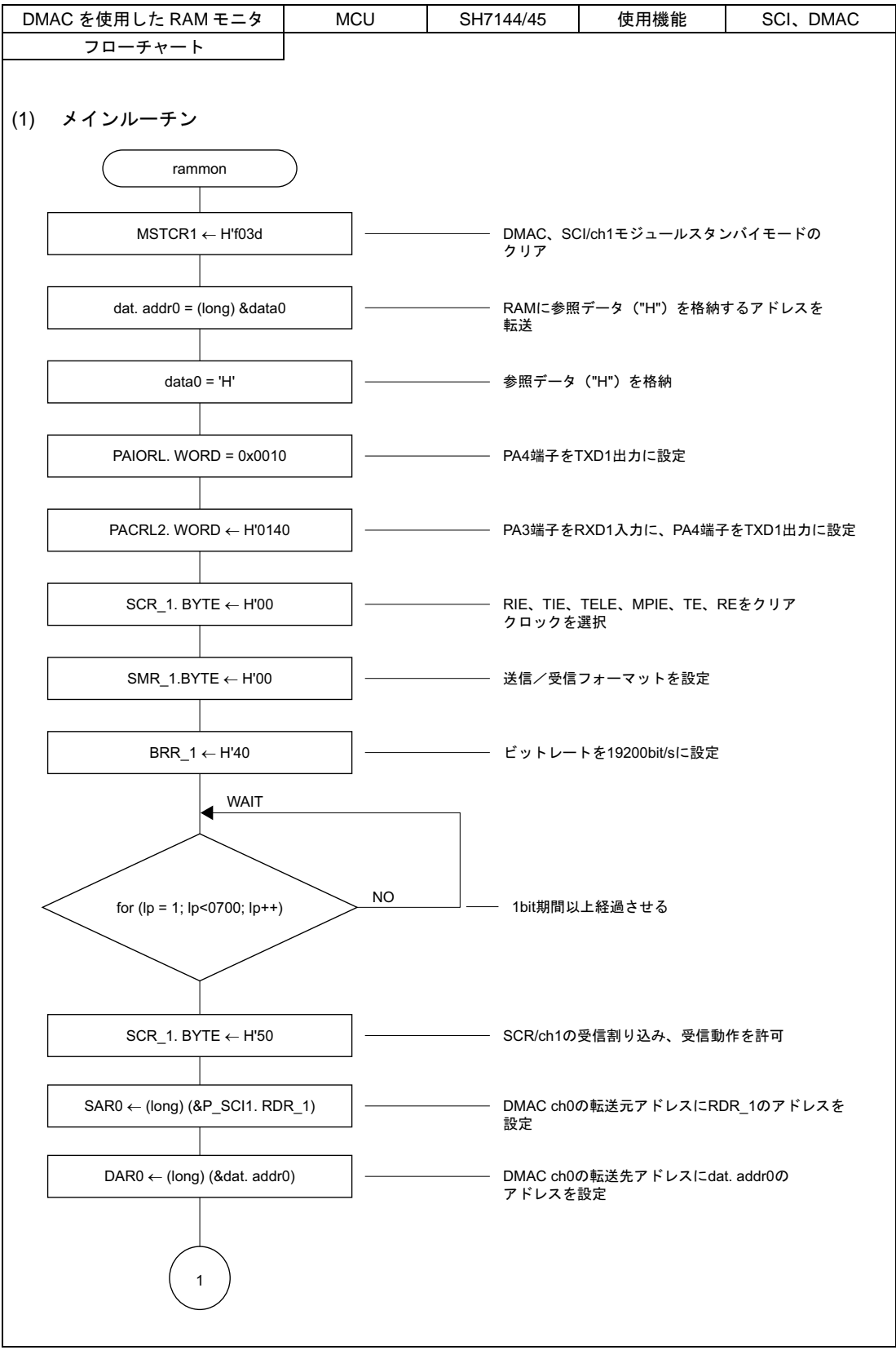
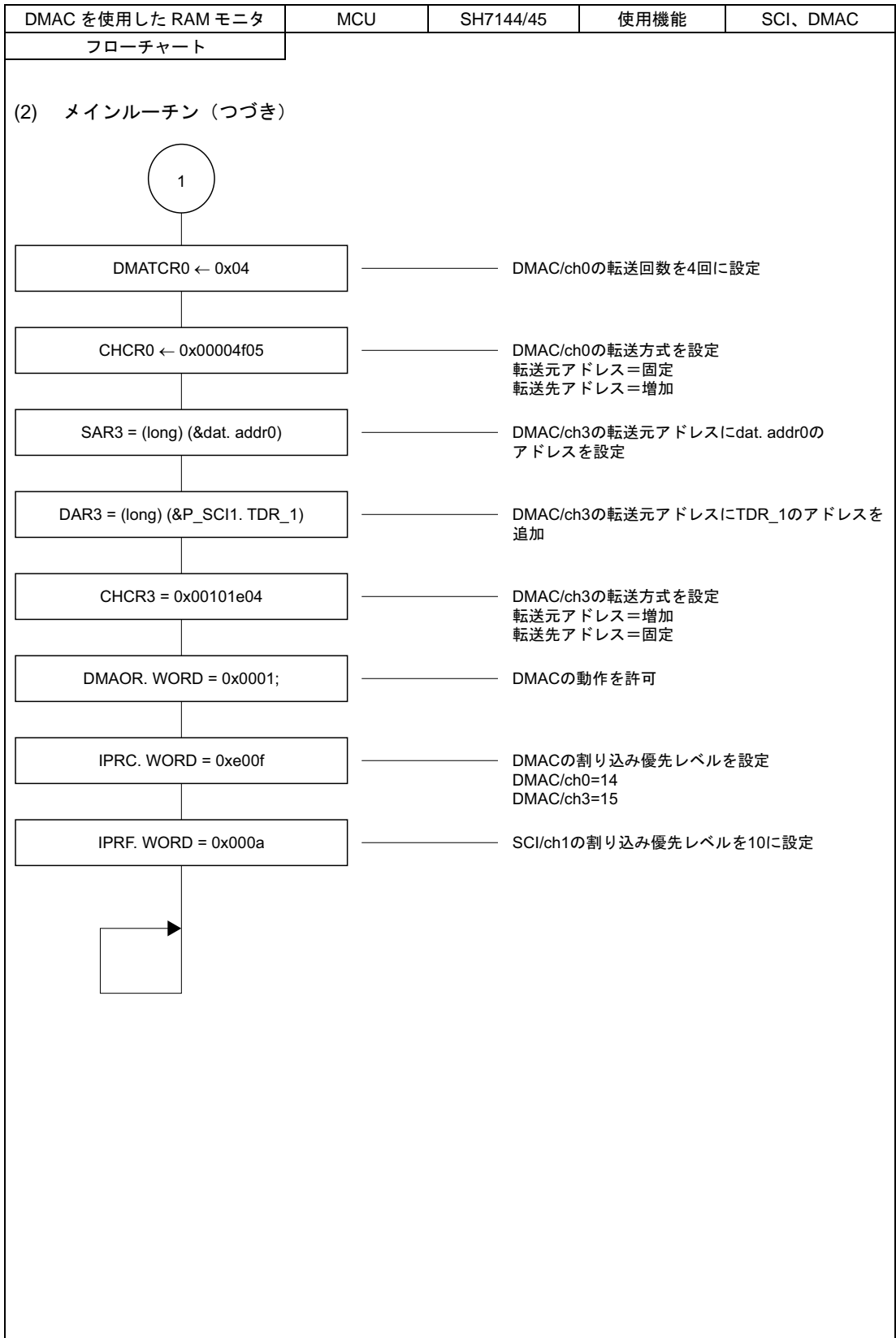
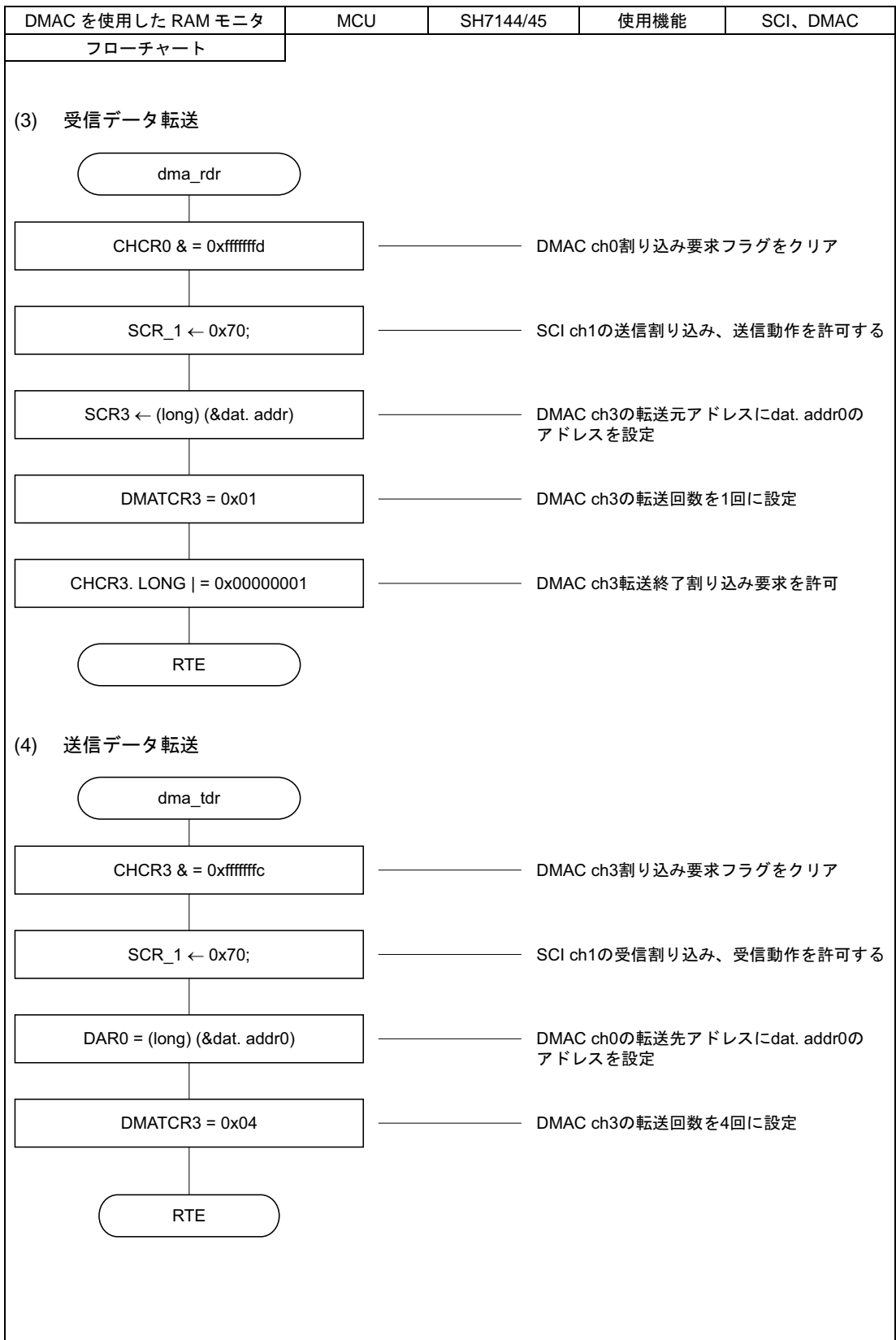


図 2.44 SCI によるデータ転送動作原理

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
ソフトウェア説明				
(1) モジュール説明				
モジュール名	ラベル名	機能割り付け		
メインルーチン	rammon	SCI および DMAC の初期設定を行う		
受信データ転送	dma_rdr	DEI0 で起動し、送信データエンプティ割り込み要求を許可する		
送信データ転送	dma_tdr	DEI3 で起動し、転送元、転送先アドレス、転送回数を再設定する。送信データエンプティ割り込み要求を禁止する		
(2) 引数の説明				
ラベル名、レジスタ名	機能割り付け	データ長	使用モジュール名	入出力
dat.addr0	RAM 参照アドレスを格納	ロングワード	メインルーチン	入力
data0	参照データ("H")を格納	バイト	メインルーチン	出力
(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	
P_STBY.MSTCR1	SCI、DMAC モジュールスタンバイモードのクリア	H'FFFF861C	H'f03d	
P_DMAC0.SAR0	RDR アドレスを設定	H'FFFF86C0	&P_SCI1.RDR_1	
P_DMAC0.DAR0	転送先 RAM 先頭アドレスを設定	H'FFFF86C4	&dat.addr0	
P_DMAC0.DMATCR0	転送回数を 4 回に設定	H'FFFF86C8	H'04	
P_DMAC0.CHCR0	DMAC の動作モード、転送方法等の設定	H'FFFF86CC	H'00004f05	
P_DMAC3.SAR3	転送元 RAM 先頭アドレスを設定	H'FFFF86F0	&dat.addr0	
P_DMAC3.DAR3	TDR のアドレスを設定	H'FFFF86F4	&P_SCI1.TDR_1	
P_DMAC3.DMATCR3	転送回数を 1 回に設定	H'FFFF86F8	H'01	
P_DMAC3.CHCR3	DMAC の動作モード、転送方法等の設定	H'FFFF86FC	H'00101e04	
P_DMAC.DMAOR	DMAC の実行するチャンネル優先順位の設定	H'FFFF86B0	H'0001	
P_PORTA.PAIORL	SCI ch1 の入出力端子に設定	H'FFFF8386	H'0010	
P_PORTA.PACRL2	端子マルチプレクスを SCI0 に設定	H'FFFF838E	H'0140	
P_INTC.IPRC	DMAC ch0/3 の割り込み優先レベルを 14/15 に設定	H'FFFF834C	H'e00f	
P_INTC.IPRF	SCI/ch1 の割り込み優先レベルを 10 に設定	H'FFFF8352	H'000a	
P_SCI1.SMR_1	SCI を調歩同期式モードに設定	H'FFFF81B0	H'00	
P_SCI1.SCR_1	送信/受信割り込み、送信/受信動作を許可	H'FFFF81B2	H'50	
P_SCI1.BRR_1	転送レートを設定(19200bit/s)	H'FFFF81B1	H'40	
(4) 使用 RAM				
本タスクでは引数以外の RAM は使用していません。				
注) レジスタのラベル名は SH7145 ヘッダファイルの名前を使用しています。				







DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
プログラムリスト	<pre> /***** /* INCLUDE FILE */ /***** #include<machine.h> #include "iodefine_7145F.h" /***** /* PROTOTYPE */ /***** void rammon(void); #pragma interrupt(dma_rdr,dma_tdr) /***** /* RAM ALLOCATION */ /***** #define data0 (*(volatile unsigned char *)0xffffe000) volatile struct addr { long addr0; }; #define dat (*(struct addr *)0xffffe010) /***** /* MAIN PROGRAM */ /***** void rammon(void) { signed int lp; P_STBY.MSTCR1.WORD = 0xf03d; /* DMAC、SCI ch1 module standby mode clear */ dat.addr0 = (long)&data0; data0 = 'H'; /* reference data = "H" */ P_PORTA.PAIORL.WORD = 0x0010; P_PORTA.PACRL2.WORD = 0X0140; P_SCI1.SCR_1.BYTE = 0x00; P_SCI1.SMR_1.BYTE = 0x00; /* Pφ/1 */ P_SCI1.BRR_1 = 0x40; /* 19200bps */ for(lp = 1; lp<0300; lp++); /* for loop = 1bit */ P_SCI1.SCR_1.BYTE = 0x50; P_DMAC0.SAR0 = (long>(&P_SCI1.RDR_1); /* DMAC ch0 source address */ </pre>			

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
プログラムリスト	<pre data-bbox="193 266 1229 761"> P_DMAC0.DAR0 = (long>(&dat.addr0); /* DMAC ch0 destination address */ P_DMAC0.DMATCR0 = 0x04; /* DMAC ch0 transfer count =4 */ P_DMAC0.CHCR0.LONG = 0x00004f05; P_DMAC3.SAR3 = (long>(&dat.addr0); /* DMAC ch3 source address */ P_DMAC3.DAR3 = (long>(&P_SCI1.TDR_1); /* DMAC ch0 destination address */ P_DMAC3.CHCR3.LONG = 0x00101e04; P_DMAC.DMAOR.WORD = 0x0001; /* Enable DMAC operation */ P_INTC.IPRC.WORD = 0xe00f; /* DMAC ch0 interrupt level =14 ch3 interrupt level =15 */ P_INTC.IPRF.WORD = 0x000a; /* SCI ch1 interrupt level =10 */ set_imask(0x0); while(1); /* Loop */ } </pre>			

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
プログラムリスト				
<pre> void dma_rdr(void) /* DMAC ch1 innterrupt routine */ { P_DMAC0.CHCR0.LONG &= 0xffffffff; /* Clear flag */ P_SCI1.SCR_1.BYTE = 0xa0; /* Enable SCI ch1 transfer & transfer data empty interrupt */ P_DMAC3.SAR3 = (long>(&dat.addr0); /* DMAC ch3 source address */ P_DMAC3.DMATCR3 = 0x01; /* DMAC ch3 transfer count =1 */ P_DMAC3.CHCR3.LONG = 0x00000001; /*DMAC ch3 enable*/ } void dma_tdr(void) /* DMAC ch3 interrupt routine*/ { P_DMAC3.CHCR3.LONG &= 0xffffffffc; /* Clear flag */ P_SCI1.SCR_1.BYTE = 0x70; /* Enable SCI ch1 receive data & receive data full interrupt */ P_DMAC0.DAR0 = (long>(&dat.addr0); /* DMAC ch0 destination address */ P_DMAC0.DMATCR0 = 0x04; /* DMAC ch3 transfer count =4 */ } </pre>				

DMAC を使用した RAM モニタ	MCU	SH7144/45	使用機能	SCI、DMAC
プログラムリスト				

SH7144シリーズ—SH7144、SH7145
アプリケーションノート

発行年月 2003年3月 第1版

発行 株式会社 日立製作所
半導体グループビジネスオペレーション本部

編集 株式会社 日立小平セミコン
技術ドキュメントグループ

©株式会社 日立製作所 2003

SH7144 シリーズ SH7144、SH7145 アプリケーションノート



ルネサスエレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668

ADJ-502-175