

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用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社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。

32192/32195/32196 グループ

タイマ TOU の応用（三相モータ制御）

1. 要約

この資料は 32192/32195/32196 グループのタイマ TOU を使用した参考プログラム例を掲載しています。

2. はじめに

この資料で説明する応用例は次のマイコン、条件での利用に適用されます。

- ・マイコン : 32192 グループ (M32192F8VFP、M32192F8UFP、M32192F8TFP)
32195 グループ (M32195F8VFP、M32195F8UFP、M32195F8TFP)
32196 グループ (M32196F8VFP、M32196F8UFP、M32196F8TFP)
- ・動作周波数 : 128～160MHz (参考プログラムは 160MHz を想定して作成しています)
- ・動作ボード : 32192 グループ用スタータキット

3. 応用技術の説明

3.1 マルチジャンクションタイマの概要

マルチジャンクションタイマ（以下 MJT と略）には入力イベントバスおよび出力イベントバスが備えられており、タイマ単独での使用に加えてタイマ相互の内部接続が可能です。この機能によりフレキシビリティに富んだタイマが構成でき、多様なアプリケーションに対応できます。タイマが内部のイベントバスとの多数の接続点を持つことからマルチジャンクションの名前が付けられています。

MJT 機能の詳細は、32192/32195/32196 グループのハードウェアマニュアルを参照してください。

4. 相補 PWM 波形出力参考プログラム

4.1 参考プログラムの概要

本参考プログラム例では、TOU0_0～TOU0_5（PWM 波形出力）、TOU0_6（次周期データの計算と設定）、TOU0_7（キャリア周期生成）を使用し、三相分の相補 PWM 波形の出力を行います。波形は外部端子 TO21～TO26（端子グループ A）へ出力します。また、相補 PWM 波形の正逆相にて同時”L”レベル検出時に外部端子への出力をカットする PWM 出力禁止機能を使用しています。

本参考プログラム例では 256 分割した正弦波テーブルを使用し、キャリア周波数 20kHz ごとに一回づつインクリメントし、78.125Hz の三角波制御による PWM 波形を出力します。

図 4.1.1 に三相モータ制御で使用される三相の相補 PWM 波形を示します。

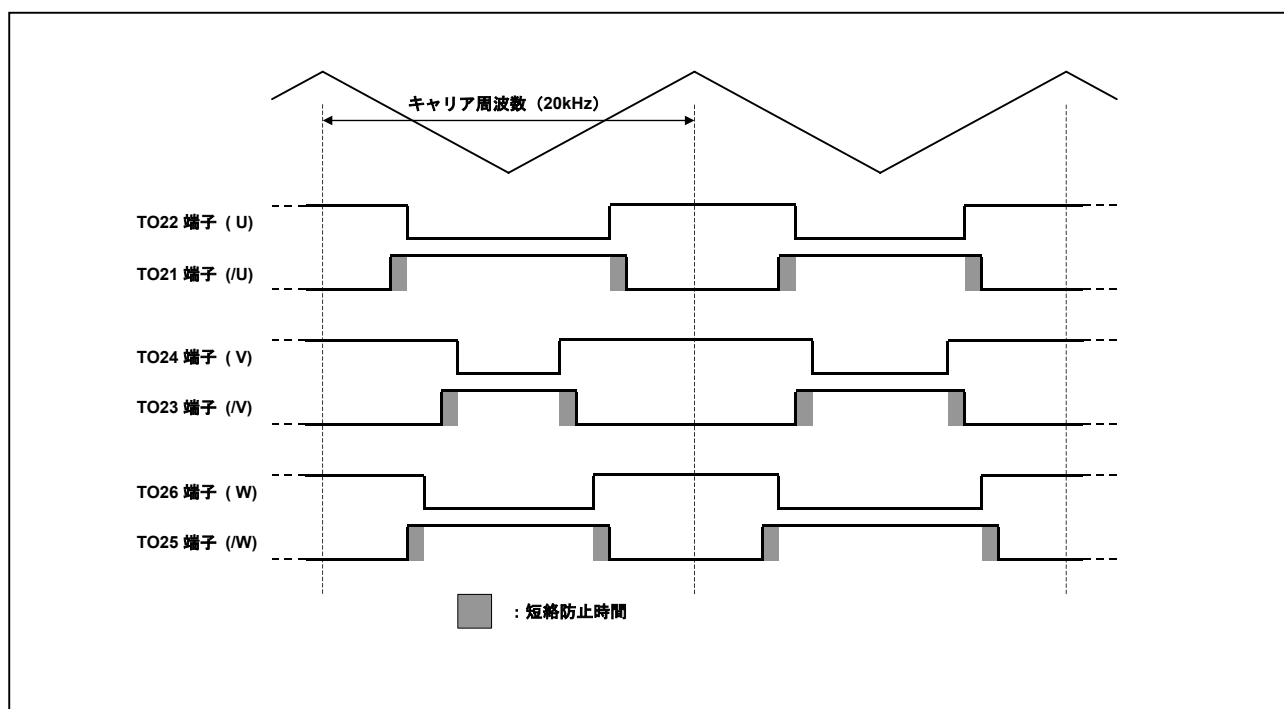


図 4.1.1 出力波形

4.2 タイマ構成図

タイマ TOU のカウントソースは、PWM の分解能を最大とするため、BCLK（40MHz）を選択し、プリスケーラ 3 で 1 分周して使用します。

TOU0_7 を連続出力モードに設定し、20kHz のキャリア周期を生成します。TOU0_0、TOU0_2、TOU0_4 をワンショット PWM 出力モードに、TOU0_6 をワンショット出力モードに設定し、TOU0_7 のアンダフローで TOU0_0、TOU0_2、TOU0_4、TOU0_6 を起動します。

TOU0_1、TOU0_3、TOU0_5 をワンショット出力モードに設定し、短絡防止機能として使用します。

TOU0_0～TOU0_5 で生成される PWM 出力波形は外部端子 TO21～TO26 から出力します。

TOU0_6 は、キャリア周期の 3/4 位置でアンダフローする値をリロードレジスタに設定します。

また、外部端子からの入力と、ポート端子レベルによる PWM 出力禁止機能を使用します。

図 4.2.1 にタイマ構成図を、図 4.2.2 にタイマ動作タイミング図を示します。

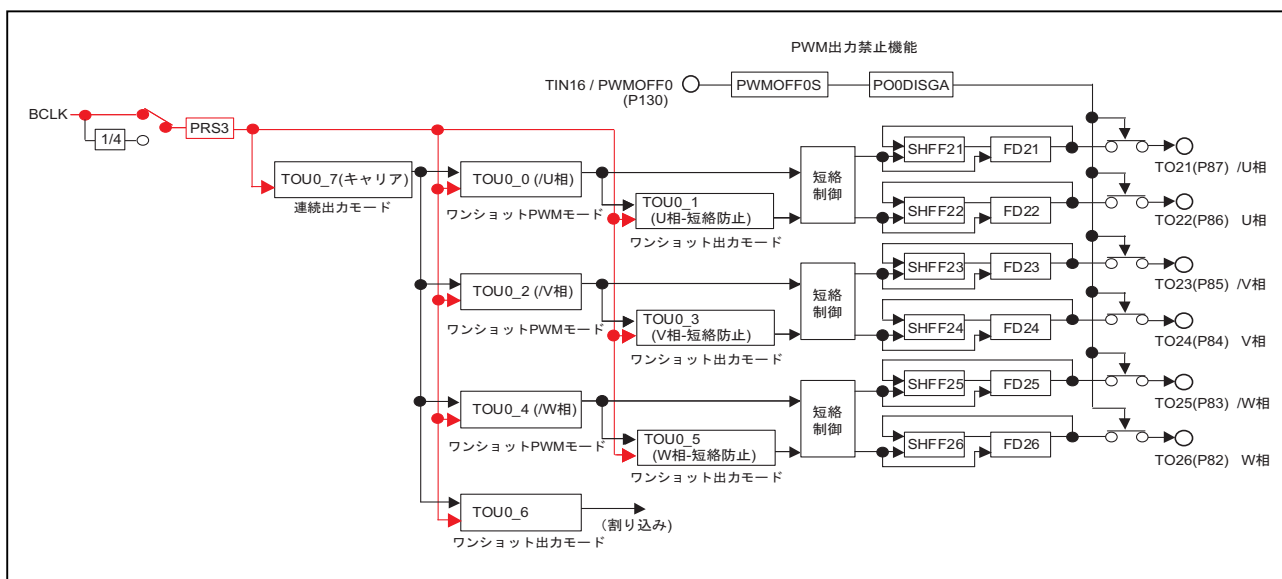


図 4.2.1 タイマ構成

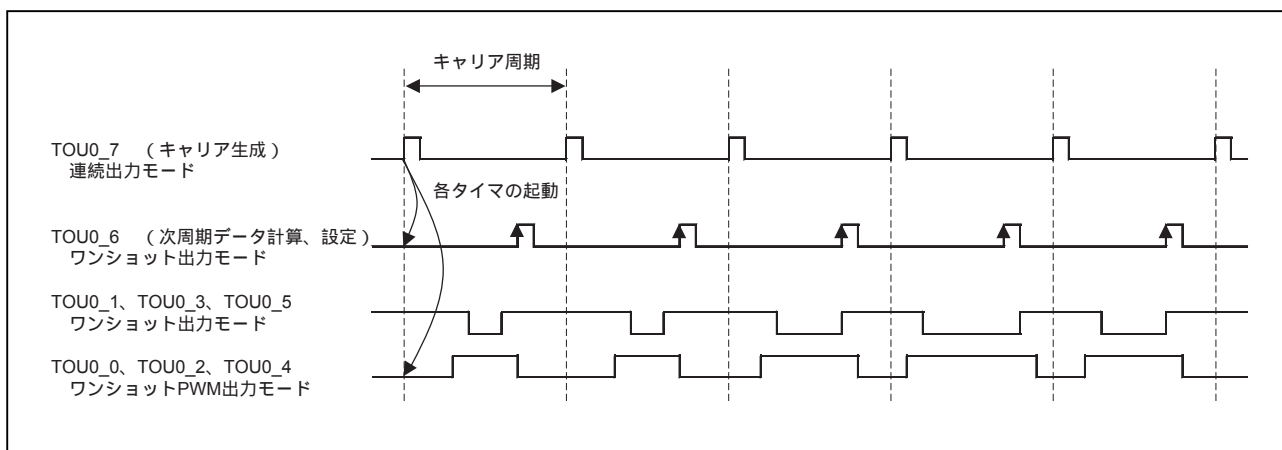


図 4.2.2 動作タイミング

4.3 タイマ動作

4.3.1 出力波形のデューティ計算方法

TOU0_6 の割り込み処理により PWM 波形のデューティ計算と TOU0_0、TOU0_2、TOU0_4 のリロードレジスタへ設定を行います。

短絡防止時間は、TOU0_1、TOU0_3、TOU0_5 のリロードレジスタへ設定を行います。

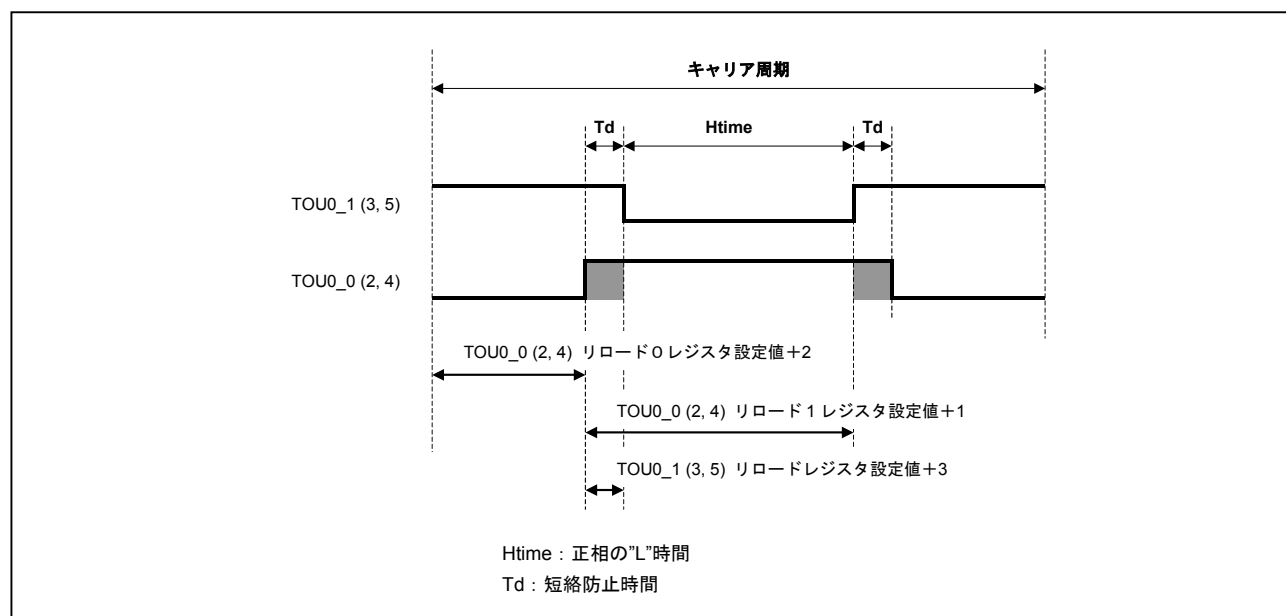


図 4.3.1 出力波形のデューティ計算方法

図 4.3.1 の波形を出力する場合、各タイマのリロードレジスタへの設定値は以下の計算式で求めます。

TOU0_1 (3, 5) :

$$\text{リロードレジスタ設定値} = Td - 3$$

TOU0_0 (2, 4) :

$$\text{リロード1 レジスタ設定値} = Htime + Td - 1$$

$$\text{リロード0 レジスタ設定値} = ((\text{キャリア周期} - Htime) / 2) - Td - 2$$

4.3.2 短絡防止機能の動作

短絡防止機能有効時は、TOU0_1 (3, 5) の TOU0 イネーブル要因選択ビットは無効になり、それぞれ TOU0_0 (2, 4) のアンダフローによって起動されます。

TOU0_0 (2, 4) は TOU0_7 のアンダフローにより起動され、カウントクロックに同期して”リロード 0 レジスタ設定値-1”をカウンタにロードしダウンカウントを行います。1 回目のカウンタアンダフローで”リロード 1 レジスタ設定値-1”をカウンタにロードしてダウンカウントを開始します。2 回目のカウンタアンダフローでカウントを停止します。1 回目と 2 回目のカウンタアンダフローの 3 カウントクロック後に TOU0_1 (3, 5) が起動されます。

図 4.3.2 に短絡防止機能有効時のタイマ動作と出力波形図を示します。

短絡防止機能を有効とする場合、F/F データレジスタと短絡防止機能用 F/F データレジスタに値を設定する必要があります。F/F 出力の反転時、F/F データレジスタと短絡防止機能用 F/F データレジスタの値が入れ替わり、入れ替わり後の F/F データレジスタの値が F/F の出力となります。

なお、F/F データレジスタと短絡防止機能用 F/F データレジスタに同じ値を設定した場合、一定出力となります。

- ・ 最初に”H”レベルを出力する場合

F/F データレジスタに”1”を、短絡防止機能用 F/F データレジスタに”0”を設定します。

- ・ 最初に”L”レベルを出力する場合

F/F データレジスタに”0”を、短絡防止機能用 F/F データレジスタに”1”を設定します。

- ・ 一定出力する場合

F/F データレジスタと短絡防止機能用 F/F データレジスタに同じ値を設定します。

各レジスタに”0”を設定した場合は”L”レベル、”1”を設定した場合は”H”レベルの出力となります。

注. ・ 短絡防止機能有効/無効ビットを設定する際は、すべてのタイマ TOU のカウンタをカウント停止状態で行ってください。

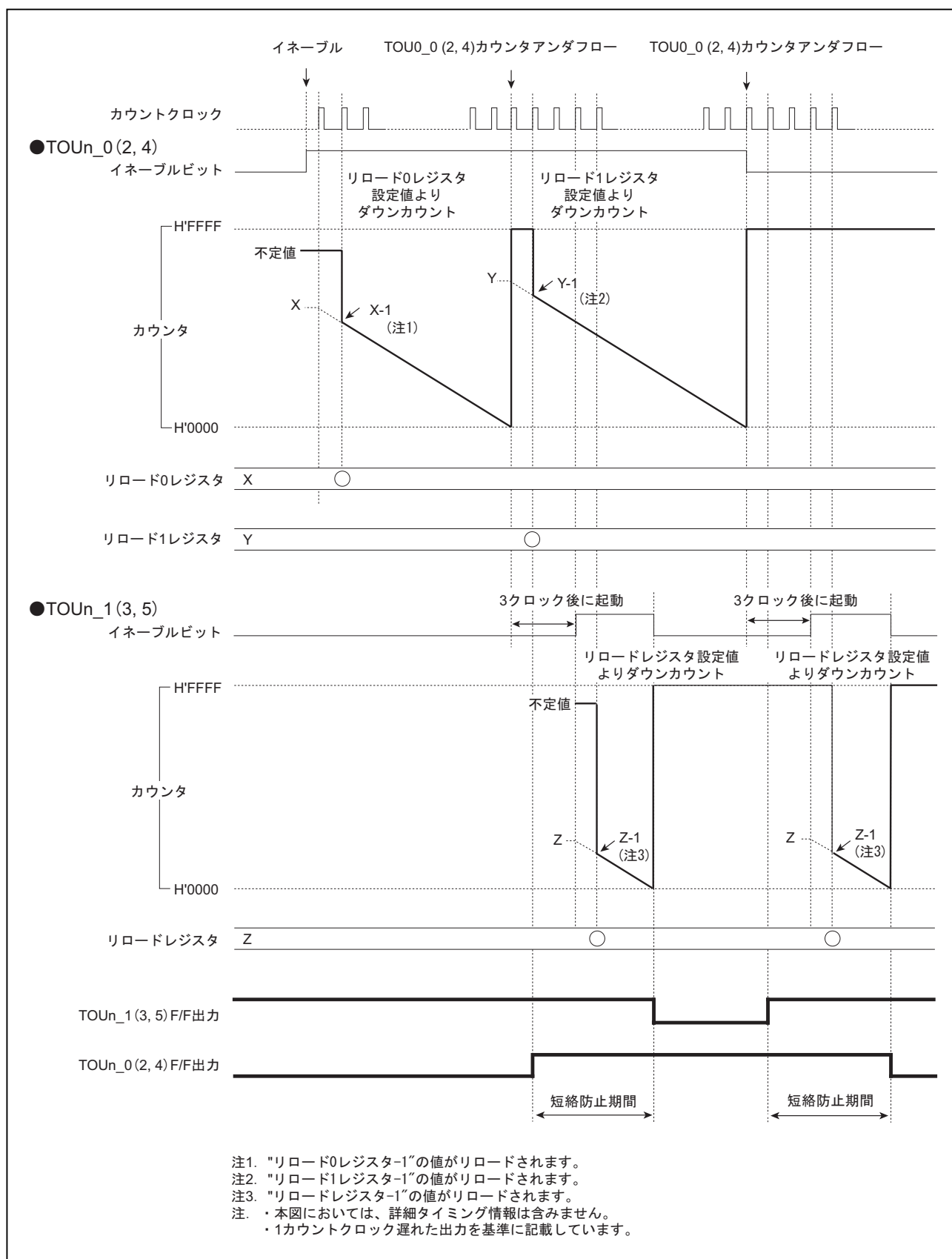


図 4.3.2 短絡防止機能有効時のタイマ動作

4.3.3 F/F データレジスタ、短絡防止機能用 F/F データレジスタの設定値と端子出力

F/F データレジスタ、短絡防止機能用 F/F データレジスタの設定値と端子出力の例を以下に示します。
なお、本参考プログラム例では、初期状態で正相側に”H”、逆相側に”L”を出力する設定を使用しています。

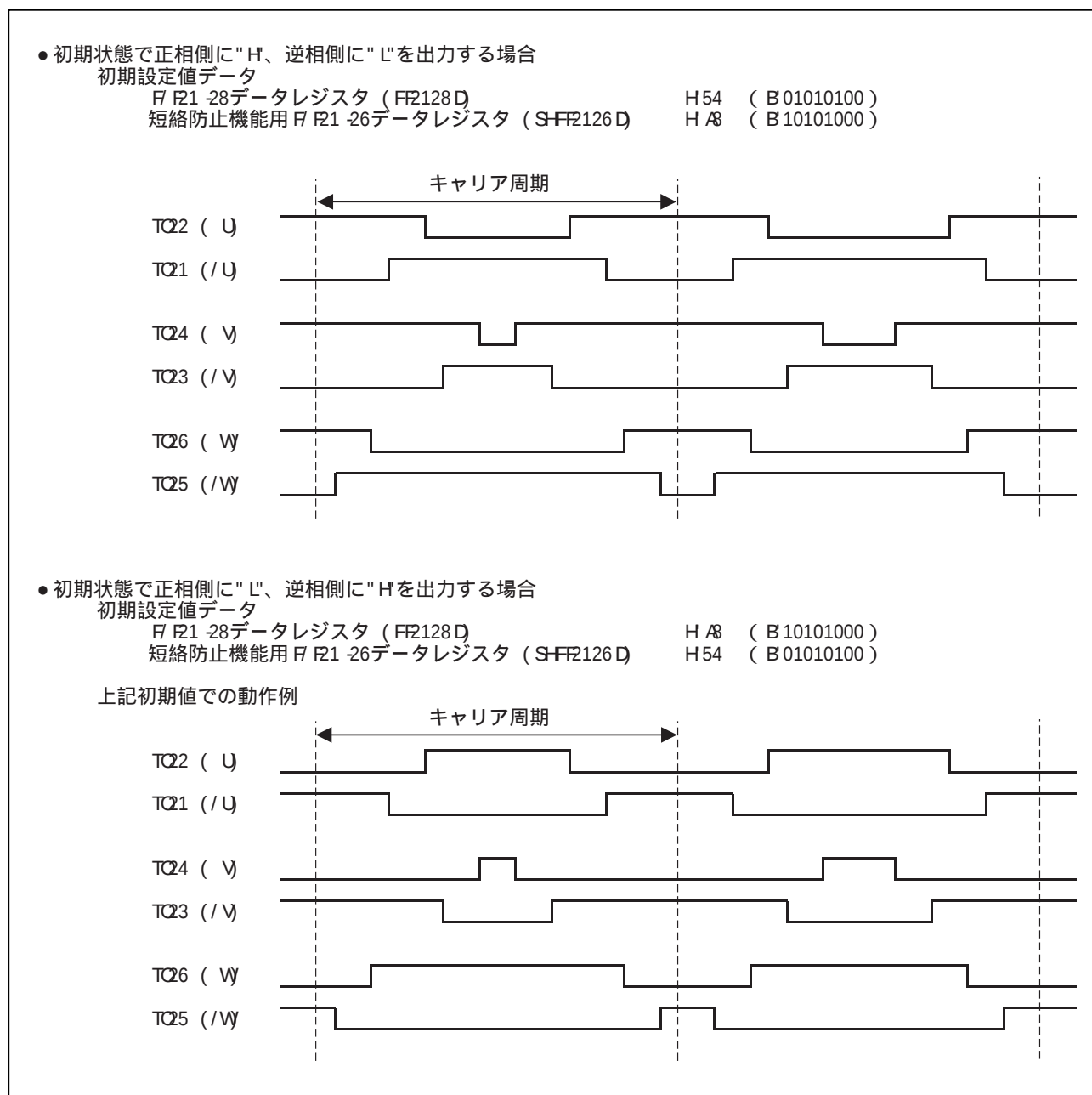


図 4.3.3 F/F データレジスタ設定値と端子出力

4.3.4 デューティ 0%、デューティ 100%の設定

本参考プログラム例では、デューティ 0%、またはデューティ 100%出力は、次周期のデータ設定タイミング（TOU0_6 割り込み処理）時に、短絡防止機能用 F/F データレジスタ（SHFF2126D）の値を操作することで実現しています。短絡防止機能用 F/F データレジスタに F/F データレジスタと同じ値を設定することで、“L”レベル、または“H”レベルを出力します。

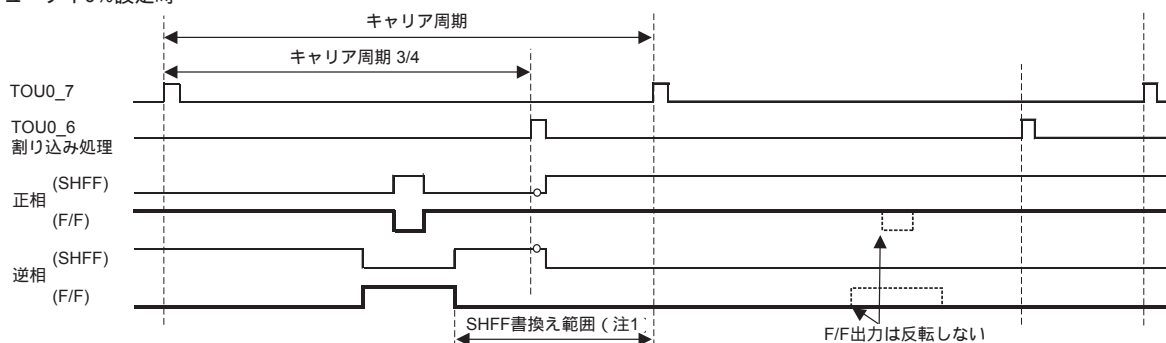
SHFF 書換え時間を確保するため、デューティ 0%設定時のキャリア周期では、0%に近い PWM 波形を出力し、デューティ 100%設定時のキャリア周期では、100%に近い PWM 波形を出力します。

デューティ 0%中は最小幅の PWM 波形を出力する値、デューティ 100%中は最大幅の PWM 波形を出力する値を各リロードレジスタに設定しています。

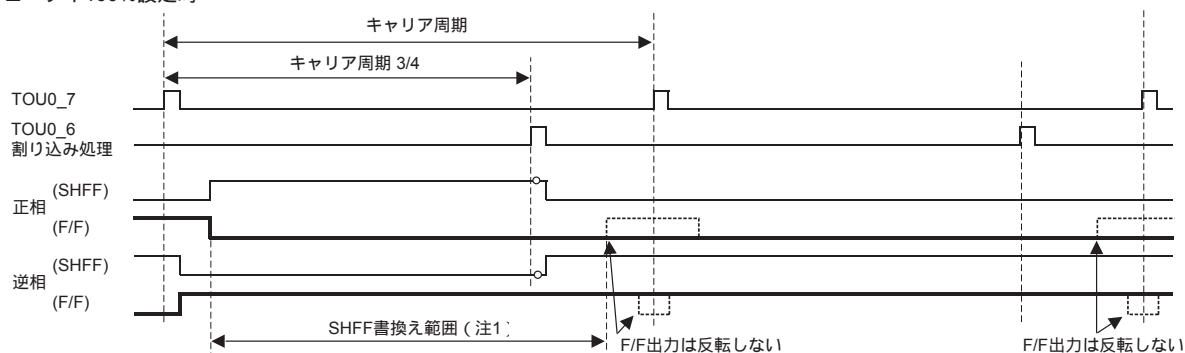
本参考プログラム例では、短絡防止機能用 F/F データレジスタ値の変更は、図 4.3.4 の“SHFF 書換え範囲”でおこなう必要があるため、TOU0_6 の割り込みをキャリア周期の 3/4 位置で発生するように設定しています。TOU0_6 割り込み発生タイミングは、他の割り込み処理などの影響による遅延が発生することを考慮する必要があります。

図 4.3.4 にデューティ 0%、デューティ 100%設定時のタイミング図を示します。

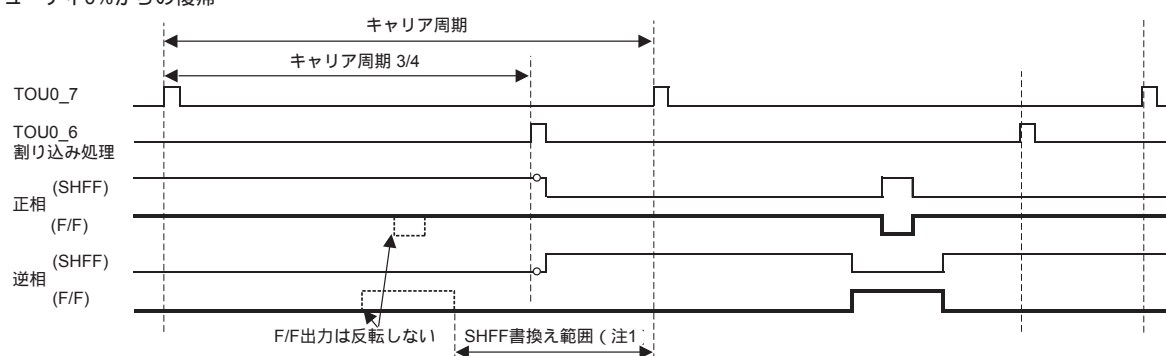
• デューティ0%設定時



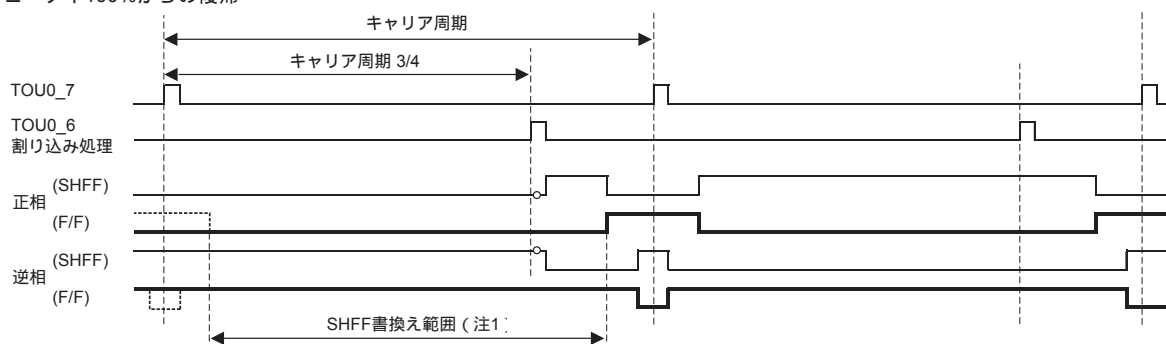
• デューティ100%設定時



• デューティ0%からの復帰



• デューティ100%からの復帰



注1. この範囲内でSHFFを書換ええない場合、想定したPWM波形が出力されません。

注. ・○: SHFF書換えタイミング。

図 4.3.4 デューティ 0%、デューティ 100%設定タイミング

4.3.5 PWM 出力禁止機能

32192/32196 グループでは、TOU0_0~TOU0_5 タイマ、および TOU1_0~TOU1_5 タイマの出力端子である P87 (P00) /TO21~P82 (P05) /TO26、および P110 (P10) /TO29~P115 (P15) /TO34 からの出力を強制的に禁止する機能を備えています。出力を禁止するには次の 3 方法があります。

- ・ 外部端子 (TIN16/PWMOFF0, TIN17/PWMOFF1) から入力された信号による PWM 出力禁止
- ・ PWM 出力禁止制御レジスタによる PWM 出力禁止
- ・ ポート P87 (P00) /TO21~P82 (P05) /TO26、P110 (P10) /TO29~P115 (P15) /TO34 の端子レベルによる PWM 出力禁止

図 4.3.5 に PWM 出力禁止機能の回路構成図を示します。

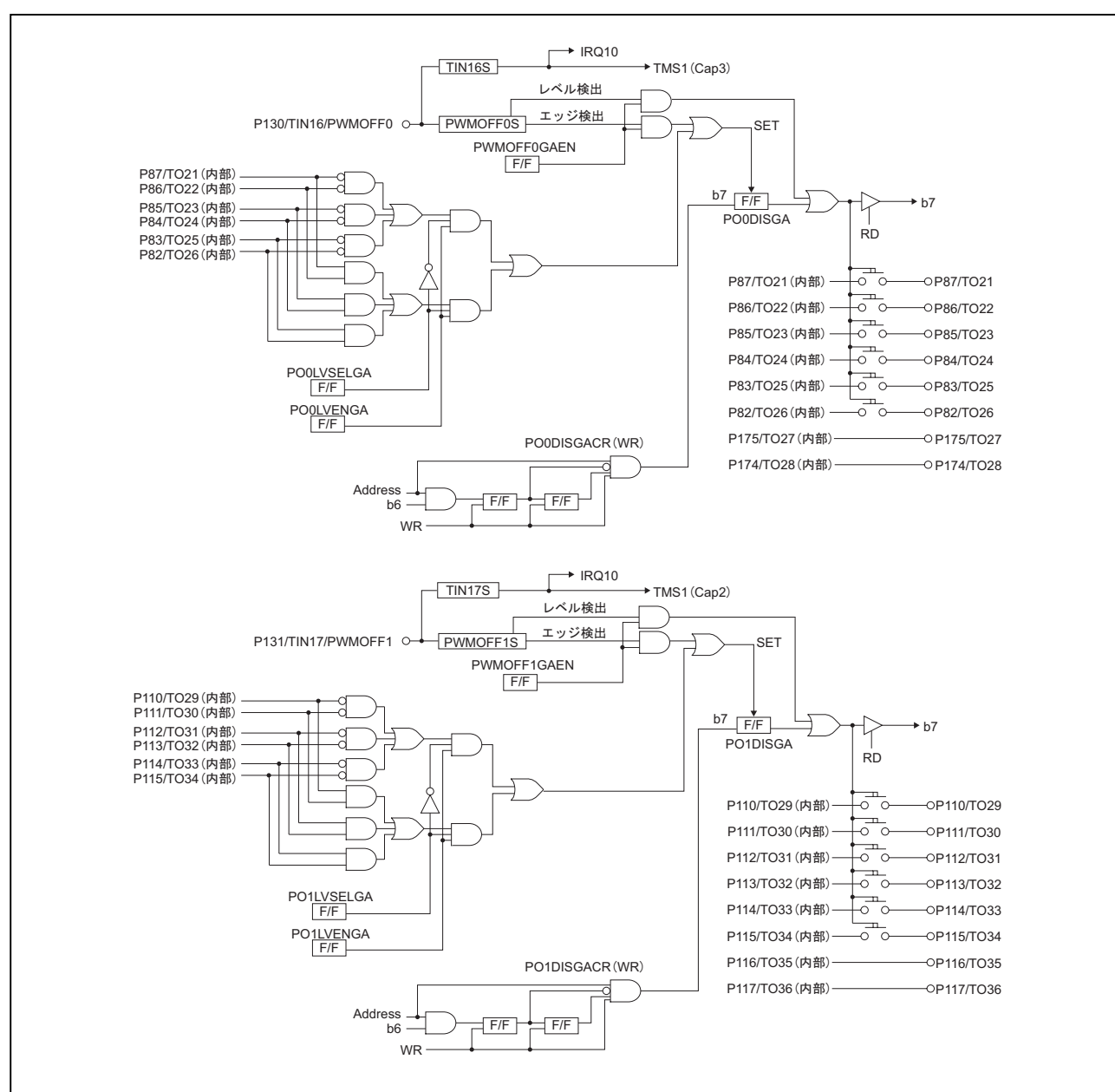


図 4.3.5 PWM 出力禁止機能の回路構成図（端子グループ A）

本参考プログラム例では、以下 2 種類の PWM 出力禁止機能を使用しています。

(1) 外部端子から入力された信号による PWM 出力禁止

外部信号（PWMOFF0）の立ち下がりエッジ検出時に外部端子 TO21~TO26 への PWM 出力が禁止されます。

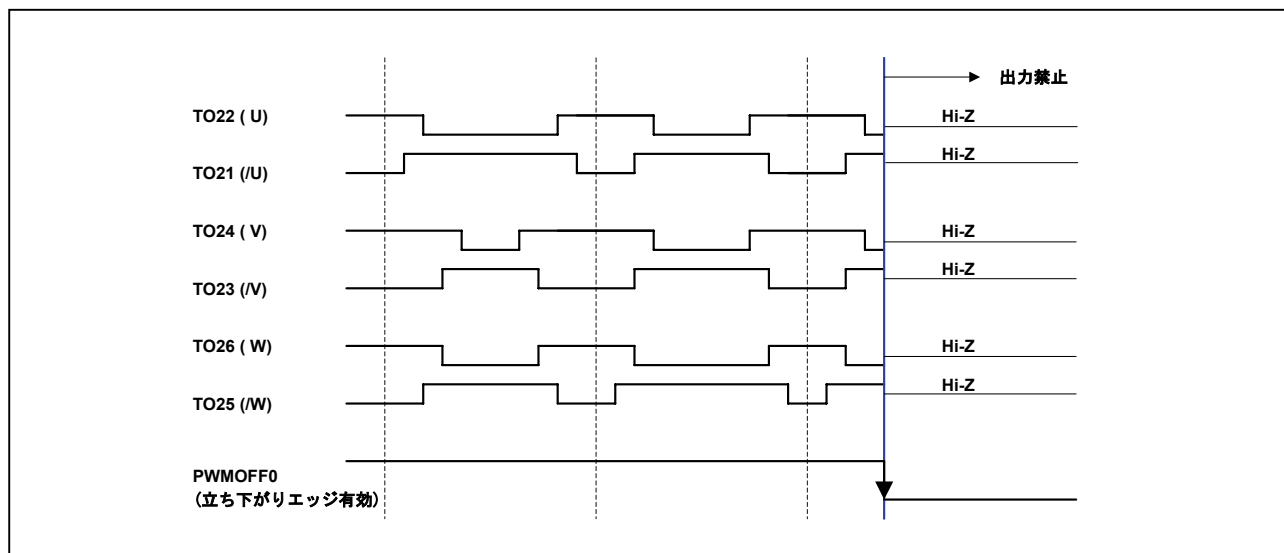


図 4.3.6 外部端子からの入力信号による PWM 出力禁止時の動作図

(2) ポート端子レベルによる PWM 出力禁止

PWM 波形の正逆相で同時”L”レベルを検出時に外部端子 TO21~TO26 への PWM 出力が禁止されます。

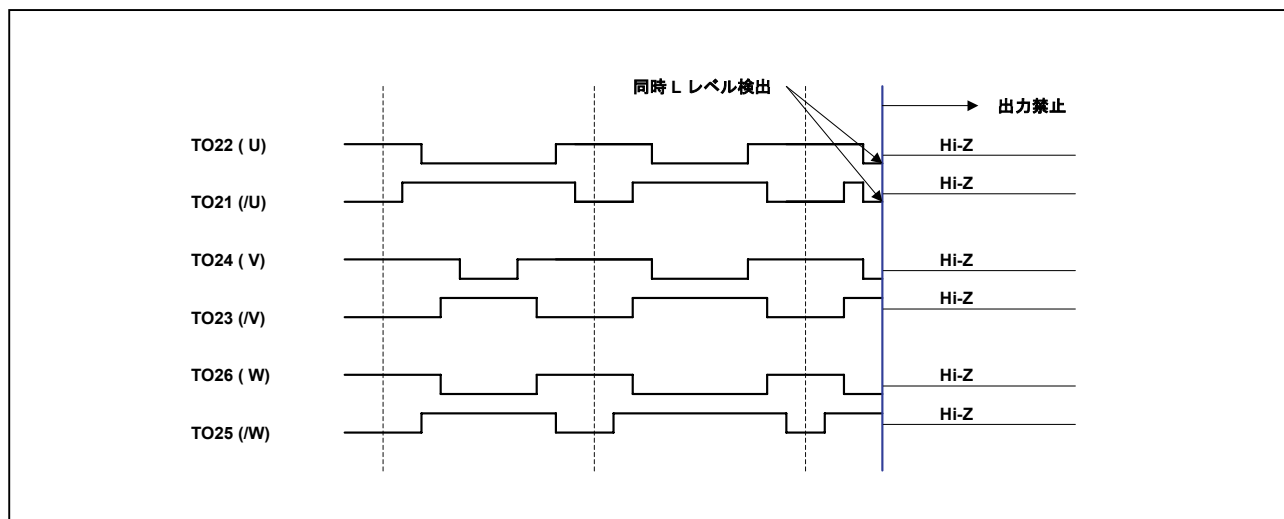


図 4.3.7 ポート端子レベルによる PWM 出力禁止時の動作図

4.4 処理手順

タイマ設定の基本処理フローを図 4.4.1 に示します。

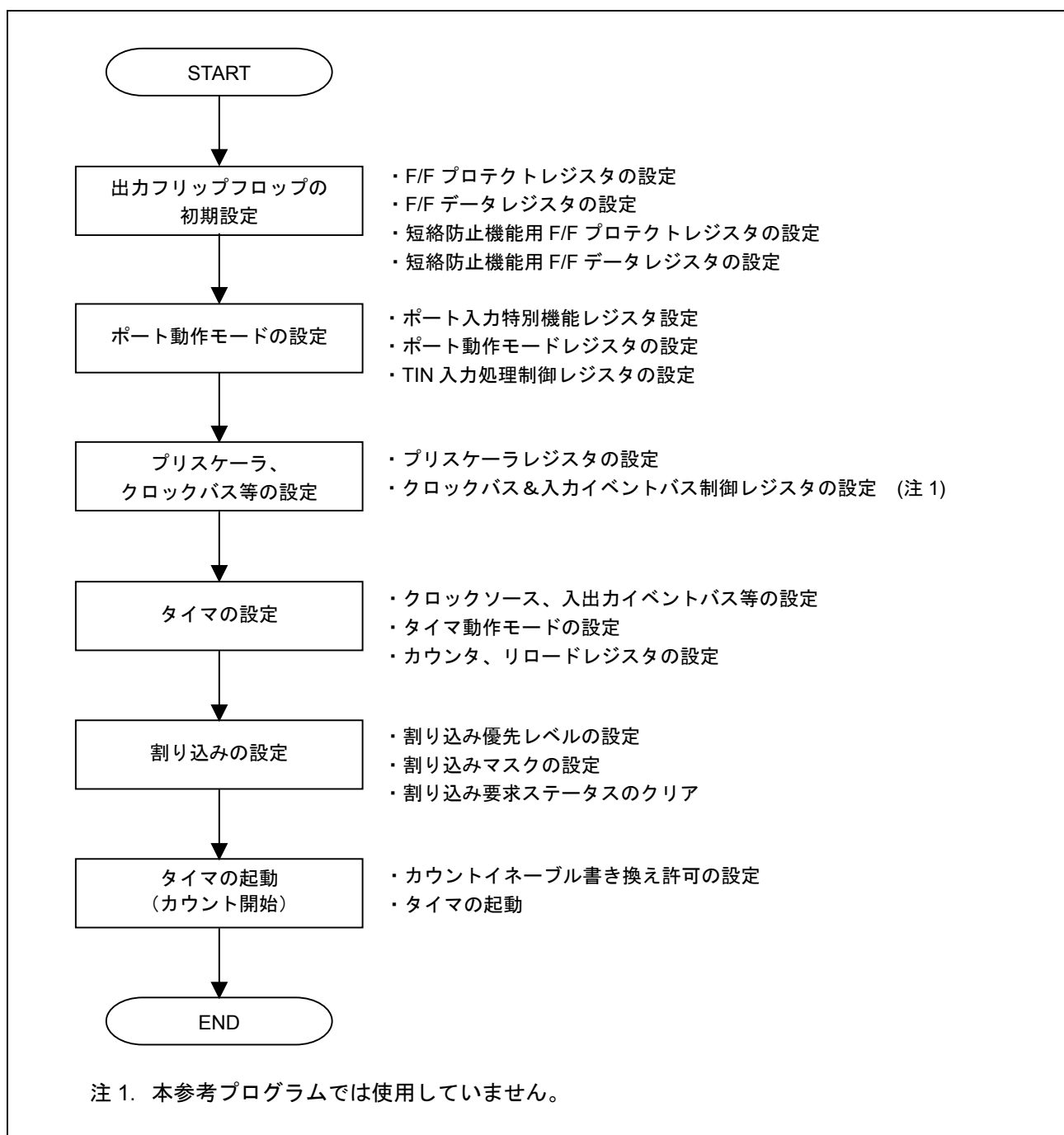


図 4.4.1 タイマ設定基本処理フロー

4.5 参考プログラムの解説

注. 使用しているレジスタを（レジスタ名：ビット名）と表記しています。

4.5.1 ポート初期化関数（port_init()）

- (1) 出力フリップフロップ制御レジスタの設定
 - ・ FF21～FF26 までの初期出力データを設定（FF2128P、FF2128D）
 - ・ 短絡防止機能用 FF21～FF26 までの初期出力データを設定（SHFF2126P、SHFF2126D）
- (2) ポートの設定
 - ・ ポート入力特別機能制御レジスタのポート入力許可ビットを入力許可に設定（PICNT：PIEN0）
 - ・ P11 を汎用ポートとして出力モードに設定（P11DATA、P11DIR、P11MOD）
 - ・ P8 を TO21～TO26 に設定（P8DATA、P8DIR、P8MOD、P8SMOD）
 - ・ P13 を TIN16 に設定（P13DIR、P13MOD、P13SMOD）
 - ・ TIN16 を立ち下がりエッジに設定（TINCR3）

4.5.2 各種初期化関数（init_func()）

- (1) ポート初期化関数の呼び出し

4.5.3 TOU0 初期化関数（TOU0_init()）

- (1) TOU0 制御レジスタ 0 を次のように設定（TOU0CR0）
 - ・ TOU0_0、TOU0_2、TOU0_4 をワンショット PWM 出力モード
 - ・ TOU0_1、TOU0_3、TOU0_5 をワンショット出力モード
 - ・ TOU0_6 をワンショット出力モード
 - ・ TOU0_7 を連続出力モード
- (2) TOU0 制御レジスタ 1 を次のように設定（TOU0CR1）
 - ・ プリスケアラ 3 使用
 - ・ プリスケアラ 3 供給クロックは BCLK
 - ・ 短絡防止機能を有効
 - ・ TOU0_0、TOU0_2、TOU0_4 をワンショット PWM 出力モード
- (3) TOU0_7 の動作周期をカウンタ、リロードレジスタに設定（TOU07CTW、TOU07RLW）
- (4) TOU0_6 のアンダフローがキャリア周期の 3/4 位置になる値をリロードレジスタに設定（TOU06RLW）
- (5) プリスケアラ分周値の設定
 - ・ プリスケアラレジスタ 3 に“プリスケアラ分周値-1”を設定（PRS3）
- (6) TID 制御&プリスケアライネーブルレジスタの設定（TID0PRS3EN）
 - ・ プリスケアラ 3 をカウント開始に設定
 - ・ TOU0 のイネーブル要因を TOU0_7 アンダフローに設定
- (7) TOU0 PWMOFF 入力処理制御レジスタを立ち下がりエッジに設定（PWMOFF0CR）
- (8) TOU0 PWM 出力 0 禁止レベル制御レジスタ GA の設定（PO0LVGACR）
 - ・ 出力禁止レベル”L”、出力禁止レベル選択有効に設定
- (9) TOU0 PWMOFF0 機能許可レジスタを PWMOFF0 機能有効に設定（PWMOFF0EN）
- (10) TIN 割り込み制御レジスタを TIN16 割り込み許可に設定（TINIR4、TINIR5）

(11) TOU0 割り込み要求マスクレジスタを TOU0_6 割り込み要求許可に設定（TOU0IST、TOU0IMA）

(12) 割り込み許可と優先レベルの設定

- ・ TOU0 出力割り込み制御レジスタを割り込み優先レベル”2”に設定（ITOU0CR）
- ・ MJT 入力割り込み制御レジスタを割り込み優先レベル”0”（最優先）に設定（IMJTICR2）

(13) 短絡防止時間の設定

- ・ TOU0_1、TOU0_3、TOU0_5 のリロードレジスタに”短絡防止時間-3”を設定（TOU01RLW、TOU03RLW、TOU05RLW）

4.5.4 PWM 波形のデューティ計算と TOU0 リロードレジスタへ設定関数（SetDuty()）

(1) 次周期の正相側の”L”時間を計算

- ・ 正弦波テーブルより 3 相（U,V,W）分のデータを取得（120° 位相差のデータとなるよう計算）

(2) 次周期のデューティが 0%時、短絡防止機能用 FF データレジスタに 0%時の値を設定する（SHFF2126P、SHFF2126D）

- ・ 最小幅の PWM 波形を出力するため、正相側”L”時間データに最小値を設定する

(3) 次周期のデューティが 100%時、短絡防止機能用 FF データレジスタに 100%時の値を設定する（SHFF2126P、SHFF2126D）

- ・ 最大幅の PWM 波形を出力するため、正相側”L”時間データに最大値を設定する

(4) 次周期のデューティが 0%、100%でない場合で、前周期が 0%または 100%の場合、短絡防止機能用 FF データレジスタに通常動作時の値を設定する（SHFF2126P、SHFF2126D）

(5) TOU0 の各リロードレジスタに計算結果を設定（TOU00RL0、TOU00RL1、TOU02RL0、TOU02RL1、TOU04RL0、TOU04RL1）

4.5.5 正弦波テーブルのポインタ設定関数（SetPointer()）

(1) 正弦波テーブルへのポインタをインクリメントする

4.5.6 メイン関数（main()）

(1) 正弦波テーブルへのポインタ”gSinPtr”の初期化

(2) 正弦波振幅係数”gAmp”の初期化と”L”時間最大値”gLlevel100Percent”の計算

(3) デューティ 0%、デューティ 100%実行中フラグ（UVW 相）の初期化

(4) 割り込み禁止関数の呼び出し

(5) 各種初期化関数の呼び出し

(6) TOU0 初期化関数の呼び出し

(7) 割り込み許可関数の呼び出し

(8) PWM 波形のデューティ計算と TOU0 リロードレジスタへ設定関数の呼び出し

(9) TOU0_7 のカウント開始（TOU0PRO、TOU0CEN）

(10) 割り込み待ち無限ループ

4.5.7 TOU0 割り込み処理関数 (Int_TOU0_7())

- (1) TOU0_6 の割り込み要求が発生しているなら次の処理を実行
 - ・ TOU0_6 の割り込み要求をクリア
 - ・ 正弦波テーブルのポインタ設定関数の呼び出し
 - ・ PWM 波形のデューティ計算と TOU0 リロードレジスタへ設定関数の呼び出し

4.5.8 TIN 割り込み処理関数 (Int_TIN1619())

- (1) TIN16 の割り込み要求が発生しているなら次の処理を実行
 - ・ TIN16 割り込み要求をクリア (TINIR4)
 - ・ TOU0 タイマをすべて停止 (TOU0PRO、TOU0CEN)
 - ・ TOU0 割り込み要求マスクレジスタを割り込み禁止に設定 (TOU0IST、TOU0IMA)
 - ・ TOU0 出力割り込み制御レジスタを割り込み禁止に設定 (ITOU0CR)
 - ・ P11 データレジスタに H'03 と H'00 を交互に出力する無限ループ


```

58                                     /* +------ TOU0_0 Single-shot pwm mode */
59
60 #define TOU0_CR1          0x4100u    /* 0100 0001 0000 0000B */
61                                     /* |||| |||| ++++ ++++ Single-shot pwm mode */
62                                     /* |||| |||+----- enable shorting prevention function */
63                                     /* ||++ +++----- No operation */
64                                     /* |+----- select BCLK */
65                                     /* +----- select prescaler3 */
66
67 /*****
68 /*          Global variable          */
69 *****/
70 UCHAR   gSinPtr;                    /* pointer of sine wave table */
71 ULONG   gAmp;                      /* Amplitude coefficient */
72 ULONG   gLlevel100Percent;         /* L level max time(count clock) */
73
74 UCHAR   g0100FlgU;                 /* U phase duty 0% or 100% executing flag */
75 UCHAR   g0100FlgV;                 /* V phase duty 0% or 100% executing flag */
76 UCHAR   g0100FlgW;                 /* W phase duty 0% or 100% executing flag */
77                                     /* 0 : normal */
78                                     /* 1 : duty 0% */
79                                     /* 2 : duty 100% */
80
81 const USHORT gaSinTbl[ 256 ] = {    /* sine wave table */
82     0,      804,   1608,   2410,   3212,   4011,   4808,   5602,   6393,   7179,   7962,   8739,
83     9512,   10278,  11039,  11793,  12539,  13279,  14010,  14732,  15446,  16151,  16846,  17530,
84     18204,  18868,  19519,  20159,  20787,  21403,  22005,  22594,  23170,  23731,  24279,  24811,
85     25329,  25832,  26319,  26790,  27245,  27683,  28105,  28510,  28898,  29268,  29621,  29956,
86     30273,  30571,  30852,  31113,  31356,  31580,  31785,  31971,  32137,  32285,  32412,  32521,
87     32609,  32678,  32728,  32767,  32757,  32728,  32678,  32609,  32521,  32412,  32285,
88     32137,  31971,  31785,  31580,  31356,  31113,  30852,  30571,  30273,  29956,  29621,  29268,
89     28898,  28510,  28105,  27683,  27245,  26790,  26319,  25832,  25329,  24811,  24279,  23731,
90     23170,  22594,  22005,  21403,  20787,  20159,  19519,  18868,  18204,  17530,  16846,  16151,
91     15446,  14732,  14010,  13279,  12539,  11793,  11039,  10278,  9512,   8739,   7962,   7179,
92     6393,   5602,   4808,   4011,   3212,   2410,   1608,   804,    0,    -804,   -1608,  -2410,
93     -3212,  -4011,  -4808,  -5602,  -6393,  -7179,  -7962,  -8739,  -9512, -10278, -11039, -11793,
94     -12539, -13279, -14010, -14732, -15446, -16151, -16846, -17530, -18204, -18868, -19519, -20159,
95     -20787, -21403, -22005, -22594, -23170, -23731, -24279, -24811, -25329, -25832, -26319, -26790,
96     -27245, -27683, -28105, -28510, -28898, -29268, -29621, -29956, -30273, -30571, -30852, -31113,
97     -31356, -31580, -31785, -31971, -32137, -32285, -32412, -32521, -32609, -32678, -32728, -32757,
98     -32767, -32757, -32728, -32678, -32609, -32521, -32412, -32285, -32137, -31971, -31785, -31580,
99     -31356, -31113, -30852, -30571, -30273, -29956, -29621, -29268, -28898, -28510, -28105, -27683,
100    -27245, -26790, -26319, -25832, -25329, -24811, -24279, -23731, -23170, -22594, -22005, -21403,
101    -20787, -20159, -19519, -18868, -18204, -17530, -16846, -16151, -15446, -14732, -14010, -13279,
102    -12539, -11793, -11039, -10278, -9512,  -8739,  -7962,  -7179,  -6393,  -5602,  -4808,  -4011,
103    -3212,  -2410,  -1608,   -804
104 };
105
106 /*****FUNC COMMENT*****/
107 * Function name: port_init()
108 *-----
109 * Description   : Initialize port
110 *-----
111 * Argument      : -
112 *-----
113 * Returns       : -
114 *-----
115 * Notes         : -
116 *****/
117 void port_init(void)
118 {
119     /* Setting F/F */
120     FF2128P = 0x00;
121     FF2128D = 0x54; /* 01010100B */
122     SHFF2126P = 0x00;
123     SHFF2126D = 0xA8; /* 10101000B */
124
125     PICNT = PIEN0; /* Enable port input */
126
127     /*** Error Output (LED) ***/
128     P11DATA = 0x00; /* Output data (must be set prior to mode) */
129     P11DIR = 0xff; /* P110-P117 : Output mode */
130     P11MOD = 0x00; /* P110-P117 : Input/output port */
131
132     /*** Setting TO21-TO26 ***/
133     P8DATA = 0x00; /* Output data (must be set prior to mode) */
134     P8DIR = 0xff; /* P82-P87 : Output mode */
135     P8MOD = 0x00; /* P82-P87 : clear */

```

```

136      P8SMOD = 0x3f;                                /* P82-P87 : TOn select */
137      P8MOD  = 0x3f;                                /* P82-P87 : TO26 - TO21 */
138
139      /**/ Setting TIN16 ***/
140      P13DIR  = 0x00;                                /* P130-P137 : Input mode */
141      P13MOD  = 0x00;
142      P13SMOD = 0x80;                                /* Select P130 for TIN16 */
143      P13MOD |= 0x80u;                                /* Select P130 for TIN16 */
144      TINCRC3 = 0x0200;                                /* TIN16 : falling edge */
145  }
146
147  /**/ FUNC COMMENT ***/
148  * Function name: init_func()
149  *-----
150  * Description : Call various initialization functions
151  *-----
152  * Argument : -
153  *-----
154  * Returns : -
155  *-----
156  * Notes : -
157  /**/ FUNC COMMENT END ***/
158  void init_func(void)
159  {
160      port_init();
161  }
162
163  /**/ FUNC COMMENT ***/
164  * Function name: TOU0_init()
165  *-----
166  * Description : -
167  *-----
168  * Argument : -
169  *-----
170  * Returns : -
171  *-----
172  * Notes : The prescaler, clock bus, etc. are set separately from the above
173  * : Must be executed while interrupts are disabled
174  /**/ FUNC COMMENT END ***/
175  void TOU0_init( void )
176  {
177      TOU0CR0 = TOU0_CR0;                                /* Set TOU0_n operation mode */
178      TOU0CR1 = TOU0_CR1;                                /* Set supplied clock select BCLK */
179
180      TOU07CTW = CARRIER_COUNT - 1u;
181      TOU07RLW = CARRIER_COUNT - 1u;                    /* TOU0_7 reload register */
182
183      TOU06RLW = (CARRIER_COUNT * 3u / 4u) - 80u;      /* TOU0_6 use for data setting */
184      /* 80 : Interrupt processing time */
185
186      PRS3 = ( 1 - 1 );                                /* Set prescaler3 (25ns@40MHz (BCLK)) */
187      TID0PRS3EN = TID0_PRS;
188
189      /* Setting PWM OFF */
190      PWM0FF0CR = PWM0FF0SP;                            /* PWM0FF0 writing control */
191      PWM0FF0CR = 0x02;                                /* PWM0FF0 falling edge */
192      POOLVGACR = 0x00;                                /* "L"level enable */
193      PWM0FF0EN = 0x40;                                /* P87 - P82 PWM OFF enable */
194
195      /* Setting interrupt */
196      TINIR4 = 0x00;                                    /* clear interrupt request */
197      TINIR5 = (~TINIM16) & 0xffu;                    /* enable TIN16 */
198      TOU0IST = 0x00;                                    /* clear interrupt request */
199      TOU0IMA = (~TOU0IM6) & 0xffu;                    /* enable TOU0_6 */
200      ITOU0CR = 0x02;                                    /* TOU0_n interrupt enable */
201      IMJTICR2 = 0x00;                                    /* TIN16 interrupt enable */
202
203      /* Setting shorting prevention time */
204      TOU01RLW = DEAD_TIME - 3u;
205      TOU03RLW = DEAD_TIME - 3u;
206      TOU05RLW = DEAD_TIME - 3u;
207  }
208
209  /**/ FUNC COMMENT ***/
210  * Function name: void SetDuty( void )
211  *-----
212  * Description : Next cycle duty calculation and setting of reload registers
213  *-----

```

```

214 * Argument      : none
215 *-----
216 * Returns       : none
217 *-----
218 * Notes         : The change from 0 to 100% and the change from 100 to 0% are the prohibitions.
219 *""FUNC COMMENT END""*****
220 void SetDuty( void )
221 {
222     UCHAR   SinPtrU, SinPtrV;
223     USHORT  DutyU_RL0, DutyU_RL1;
224     USHORT  DutyV_RL0, DutyV_RL1;
225     USHORT  DutyW_RL0, DutyW_RL1;
226     SSHORT  OrdVu, OrdVv, OrdVw;
227     SLONG   HtimeU, HtimeV, HtimeW;
228
229     /** Get sine wave table pointer **/
230     SinPtrU = gSinPtr;
231     SinPtrV = gSinPtr + 86;
232     /** U-phase increase and decrease amount calculation of "H" time **/
233     OrdVu = (gLlevel100Percent * gaSinTbl[ SinPtrU ]) >> 16;
234     HtimeU = OrdVu + H_MID;
235     /** V-phase increase and decrease amount calculation of "H" time **/
236     OrdVv = (gLlevel100Percent * gaSinTbl[ SinPtrV ]) >> 16;
237     HtimeV = OrdVv + H_MID;
238     /** W-phase increase and decrease amount calculation of "H" time **/
239     OrdVw = - OrdVu - OrdVv;
240     HtimeW = OrdVw + H_MID;
241
242     /** U-phase duty judgment and setting at the next cycle **/
243     if( HtimeU < H_MIN ){
244         HtimeU = H_MIN;
245         if( g0100FlgU == 0 ){
246             SHFF2126P = 0x3F;
247             g0100FlgU = 1;
248             SHFF2126D = 0x40u;
249         }
250     }
251     else if( H_MAX < HtimeU ){
252         HtimeU = H_MAX;
253         if( g0100FlgU == 0 ){
254             SHFF2126P = 0x3F;
255             g0100FlgU = 2;
256             SHFF2126D = 0x80u;
257         }
258     }
259     else{
260         if( g0100FlgU == 1 ){
261             SHFF2126P = 0x3F;
262             g0100FlgU = 0;
263             SHFF2126D = 0x80u;
264         }
265         else if( g0100FlgU == 2 ){
266             SHFF2126P = 0x3F;
267             g0100FlgU = 0;
268             SHFF2126D = 0x40u;
269         }
270     }
271     DutyU_RL1 = HtimeU + DEAD_TIME - 11;
272     DutyU_RL0 = ((CARRIER_COUNT - HtimeU) >> 1) - DEAD_TIME - 21;
273
274     /** V-phase duty judgment and setting at the next cycle **/
275     if( HtimeV < H_MIN ){
276         HtimeV = H_MIN;
277         if( g0100FlgV == 0 ){
278             SHFF2126P = 0xCF;
279             g0100FlgV = 1;
280             SHFF2126D = 0x10;
281         }
282     }
283     else if( H_MAX < HtimeV ){
284         HtimeV = H_MAX;
285         if( g0100FlgV == 0 ){
286             SHFF2126P = 0xCF;
287             g0100FlgV = 2;
288             SHFF2126D = 0x20;
289         }
290     }
291     else{

```

```

292         if( g0100FlgV == 1 ){                                /* It was duty 0% last time */
293             SHFF2126P = 0xCF;
294             g0100FlgV = 0;
295             SHFF2126D = 0x20;                                /* 'xx01 xxxx' write */
296         }
297         else if( g0100FlgV == 2 ){                            /* It was duty 100% last time */
298             SHFF2126P = 0xCF;
299             g0100FlgV = 0;
300             SHFF2126D = 0x10;                                /* 'xx10 xxxx' write */
301         }
302     }
303     DutyV_RL1 = HtimeV + DEAD_TIME - 11;
304     DutyV_RL0 = ((CARRIER_COUNT - HtimeV) >> 1) - DEAD_TIME - 21;
305
306     /** W-phase duty judgment and setting at the next cycle */
307     if( HtimeW < H_MIN ){                                     /* Duty 0% */
308         HtimeW = H_MIN;
309         if( g0100FlgW == 0 ){                                /* /W="H", W="L" fix */
310             SHFF2126P = 0xF3;
311             g0100FlgW = 1;                                    /* Duty 0% memory */
312             SHFF2126D = 0x04u;                                /* 'xxxx 10xx' write */
313         }
314     }
315     else if( H_MAX < HtimeW ){                                /* Duty 100% */
316         HtimeW = H_MAX;
317         if( g0100FlgW == 0 ){                                /* /W="L", W="H" fix */
318             SHFF2126P = 0xF3;
319             g0100FlgW = 2;                                    /* Duty 100% memory */
320             SHFF2126D = 0x08u;                                /* 'xxxx 01xx' write */
321         }
322     }
323     else{                                                     /* normal */
324         if( g0100FlgW == 1 ){                                /* It was duty 0% last time */
325             SHFF2126P = 0xF3;
326             g0100FlgW = 0;
327             SHFF2126D = 0x08u;                                /* 'xxxx 01xx' write */
328         }
329         else if( g0100FlgW == 2 ){                            /* It was duty 100% last time */
330             SHFF2126P = 0xF3;
331             g0100FlgW = 0;
332             SHFF2126D = 0x04u;                                /* 'xxxx 10xx' write */
333         }
334     }
335     DutyW_RL1 = HtimeW + DEAD_TIME - 11;
336     DutyW_RL0 = ((CARRIER_COUNT - HtimeW) >> 1) - DEAD_TIME - 21;
337
338     /** Data setting */
339     TOU00RL1 = DutyU_RL1;
340     TOU00RL0 = DutyU_RL0;
341     TOU02RL1 = DutyV_RL1;
342     TOU02RL0 = DutyV_RL0;
343     TOU04RL1 = DutyW_RL1;
344     TOU04RL0 = DutyW_RL0;
345 }
346
347 /*****FUNC COMMENT*****/
348 * Function name: void SetPointer( void )
349 *-----
350 * Description : setting pointer of sine wave table
351 *-----
352 * Argument    : none
353 *-----
354 * Returns     : none
355 *-----
356 * Notes       : Only the increment is done for the simplification.
357 *****/
358 void SetPointer( void )
359 {
360     gSinPtr++;
361 }
362
363 /*****FUNC COMMENT*****/
364 * Function name: main()
365 *-----
366 * Description : -
367 *-----
368 * Argument    : -
369 *-----

```

```

370  * Returns      : -
371  *-----
372  * Notes        : -
373  *""FUNC COMMENT END""*****
374  void main(void)
375  {
376      gSinPtr = 0;
377      gAmp = 40ul;                                /* Amplitude coefficient */
378
379      /* When the gAmp data is changed, it is necessary to calculate the "gLlevel100Percent" again */
380      if( gAmp > AMP_MAX ){
381          gAmp = AMP_MAX;
382      }
383      gLlevel100Percent = gAmp * H_MAX / 100;        /* L level max value(count clock) */
384
385      g0100FlgU = 0;                                /* duty 0% or 100% executing flags clear */
386      g0100FlgV = 0;
387      g0100FlgW = 0;
388
389      DisInt();                                       /* Disable interrupt */
390
391      init_func();
392
393      TOU0_init();                                   /* Initialize TOU0 */
394
395      EnInt();                                       /* Enable interrupt */
396
397      SetDuty();
398
399      TOU0PRO = 0xFE;                                /* TOU0_7 write enable */
400      TOU0CEN = 0x01;                                /* TOU0_7 start */
401
402      while( 1 ){
403          ;
404      }
405  }
406
407  /""FUNC COMMENT""*****
408  * Function name: void Int_TOU0_7( void )
409  *-----
410  * Description : TOU0 counter under flow interrupt
411  *-----
412  * Argument    : none
413  *-----
414  * Returns     : none
415  *-----
416  * Notes       : -
417  *""FUNC COMMENT END""*****
418  void Int_TOU0_7( void )
419  {
420      /* TOU0_6 interrupt */
421      if( (TOU0IST & TOU0IS6) != 0u ){
422          TOU0IST = ~TOU0IS6 & 0xffu;                /* clear TOU0_6 interrupt request */
423          SetPointer();
424          SetDuty();
425      }
426  }
427
428  /""FUNC COMMENT""*****
429  * Function name: void Int_TIN1619( void )
430  *-----
431  * Description : TIN16 - TIN19 interrupt request
432  *-----
433  * Argument    : none
434  *-----
435  * Returns     : none
436  *-----
437  * Notes       : FO signal input proc
438  *""FUNC COMMENT END""*****
439  void Int_TIN1619( void )
440  {
441      if ( (TINIR4 & TINIS16) != 0u ){
442          TINIR4 = ~TINIS16 & 0xffu;                /* clear TIN16 interrupt request */
443
444          TOU0PRO = 0x00;                            /* TOU0_0-7 write enable */
445          TOU0CEN = 0x00;                            /* TOU0_0-7 stop */
446
447          TOU0IST = 0x00;                            /* clear interrupt request */

```

```

448         TOU0IMA = 0xff;
449         ITOU0CR = 0x07;
450
451         /** Error output for LED **/
452         while(1){
453             SINT32 i;
454             for(i=0;i<0x100000;i++){
455                 ;
456             }
457             P11DATA^=0x03u;
458         }
459     }
460 }

```

4.6.2 startup.ms （一部を抜粋）

（省略）

```

72 ;*****
73 ; ICU Vector Table
74 ;*****
75 ;
76 .SECTION          ICUVECT, DATA, ALIGN=4
77 ;
78 .IMPORT            $Int_TIN1619
79 .IMPORT            $Int_TOU0_7
80 ;
81 vectbl:
82 .DATA.W            EIT_reset                ; H'0000 0094    MJT Input Interrupt 4:TIN3-TIN6
83 .DATA.W            EIT_reset                ; H'0000 0098    MJT Input Interrupt 3:TIN20-TIN27
84 .DATA.W            $Int_TIN1619            ; H'0000 009C    MJT Input Interrupt 2:TIN16-TIN19
85 .DATA.W            EIT_reset                ; H'0000 00A0    MJT Input Interrupt 1:TIN0
86 .DATA.W            EIT_reset                ; H'0000 00A4    MJT Input Interrupt 0:TIN7-TIN10
87 .DATA.W            EIT_reset                ; H'0000 00A8    MJT Output Interrupt 7:TMS0,TMS1
88 .DATA.W            EIT_reset                ; H'0000 00AC    MJT Output Interrupt 6:TOP8,TOP9
89 .DATA.W            EIT_reset                ; H'0000 00B0    MJT Output Interrupt 5:TOP10
90 .DATA.W            EIT_reset                ; H'0000 00B4    MJT Output Interrupt 4:TIO4-TIO7
91 .DATA.W            EIT_reset                ; H'0000 00B8    MJT Output Interrupt 3:TIO8,TIO9
92 .DATA.W            EIT_reset                ; H'0000 00BC    MJT Output Interrupt 2:TOP0-TOP5
93 .DATA.W            EIT_reset                ; H'0000 00C0    MJT Output Interrupt 1:TOP6,TOP7
94 .DATA.W            EIT_reset                ; H'0000 00C4    MJT Output Interrupt 0:TIO0-TIO3
95 .DATA.W            EIT_reset                ; H'0000 00C8    DMAC0-4 Interrupt:DMA0-DMA4
96 .DATA.W            EIT_reset                ; H'0000 00CC    SIO1 Receive Interrupt
97 .DATA.W            EIT_reset                ; H'0000 00D0    SIO1 Transmit Interrupt
98 .DATA.W            EIT_reset                ; H'0000 00D4    SIO0 Receive Interrupt
99 .DATA.W            EIT_reset                ; H'0000 00D8    SIO0 Transmit Interrupt
100 .DATA.W            EIT_reset                ; H'0000 00DC    A/D0 Conversion Interrupt
101 .DATA.W            EIT_reset                ; H'0000 00E0    TIO0 Output Interrupt
102 .DATA.W            $Int_TOU0_7            ; H'0000 00E4    TOD0 Output Interrupt
103 .DATA.W            EIT_reset                ; H'0000 00E8    DMAC5-9 Interrupt:DMA5-DMA9
104 .DATA.W            EIT_reset                ; H'0000 00EC    SIO2,3 Transmit/Receive Interrupt
105 .DATA.W            EIT_reset                ; H'0000 00F0    RTD Interrupt
106 .DATA.W            EIT_reset                ; H'0000 00F4    TID1 Output Interrupt
107 .DATA.W            EIT_reset                ; H'0000 00F8    TOU1 Output Interrupt:TOU1_0-TOU1_7
108 .DATA.W            EIT_reset                ; H'0000 00FC    SIO4,5 Transmit/Receive Interrupt
109 .DATA.W            EIT_reset                ; H'0000 0100    Reserved
110 .DATA.W            EIT_reset                ; H'0000 0104    Reserved
111 .DATA.W            EIT_reset                ; H'0000 0108    TML1 Input Interrupt:TIN30-TIN33
112 .DATA.W            EIT_reset                ; H'0000 010C    CAN0 Transmit/Receive & Error Interrupt
113 .DATA.W            EIT_reset                ; H'0000 0110    CAN1 Transmit/Receive & Error Interrupt
114 .DATA.W            EIT_reset                ; H'0000 0114    DRI Transfer Interrupt
115 .DATA.W            EIT_reset                ; H'0000 0118    DRI Counter Interrupt:DEC0-DEC4
116 .DATA.W            EIT_reset                ; H'0000 011C    DRI Event Detection Interrupt:DIN0-DIN5
117 .DATA.W            EIT_reset                ; H'0000 0120    CAN0 Transmit/Receive Completion Interrupt
118 .DATA.W            EIT_reset                ; H'0000 0124    CAN0 Single-Shot Interrupt
119 .DATA.W            EIT_reset                ; H'0000 0128    CAN0 Error Interrupt
120 .DATA.W            EIT_reset                ; H'0000 012C    CAN1 Transmit/Receive Completion Interrupt
121 .DATA.W            EIT_reset                ; H'0000 0130    CAN1 Single-Shot Interrupt
122 .DATA.W            EIT_reset                ; H'0000 0134    CAN1 Error Interrupt
123 .DATA.W            EIT_reset                ; H'0000 0138    RAM Write Monitor Interrupt
124 ;

```

（以下省略）

5. 参考ドキュメント

- 32192/32196 グループ ハードウェアマニュアル Rev.1.00
- 32195 グループ データシート Rev.1.00
- M3T-CC32R V.5.00 ユーザーズマニュアル（C コンパイラ編）
- M3T-AS32R V.5.00 ユーザーズマニュアル（アセンブラ編）
- M32R-FPU ソフトウェアマニュアル Rev.1.01
（最新版をルネサス テクノロジホームページから入手してください。）

6. ホームページとサポート窓口

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

ルネサス製品全般に関するお問合せと M32R ファミリに関する技術的なお問合せ先
カスタマサポートセンタ : csc@renesas.com

改訂記録	32192/32195/32196 グループ タイマ TOU の応用（三相モータ制御） アプリケーションノート
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2005.11.21	－	初版発行

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご注意ください。

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しては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気づきの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。