

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

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

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

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、ディスクリート半導体、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留意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したものです。万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海底中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気付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。

アプリケーションノート

DC ブラシレスモータ制御 (H8/3664)

要旨

H8/3664 を使用した DC ブラシレスモータを制御します。

動作確認デバイス

H8/300H Tiny シリーズ－H8/3664－

目次

ご注意.....	2
1. 仕様.....	3
2. 考え方.....	4
3. 使用機能説明.....	5
4. 動作説明.....	14
5. ソフトウェア説明.....	18
5.1 モジュール説明.....	18
5.2 引数の説明.....	18
5.3 内部使用レジスタ説明.....	19
5.4 使用 RAM 説明.....	21
5.5 データテーブル説明.....	22
6. フローチャート.....	23
7. プログラムリスト.....	29

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。保証値を越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。

Copyright©Hitachi, Ltd., 2003. All rights reserved.

1. 仕様

1. 図 1.1 に示すように、H8/3664 を使用し DC ブラシレスモータを制御します。
2. DC ブラシレスモータはロータの磁極位置信号を検出し、各位置信号にあった回転磁界をポートから出力することで回転します。
3. H8/3664 内蔵タイマにより PWM 波形を生成し、チョッピング制御します。
4. DC ブラシレスモータの磁極位置を検出して I/O ポートで 6 相のレベルを切替え、ベースドライバへ出力します。

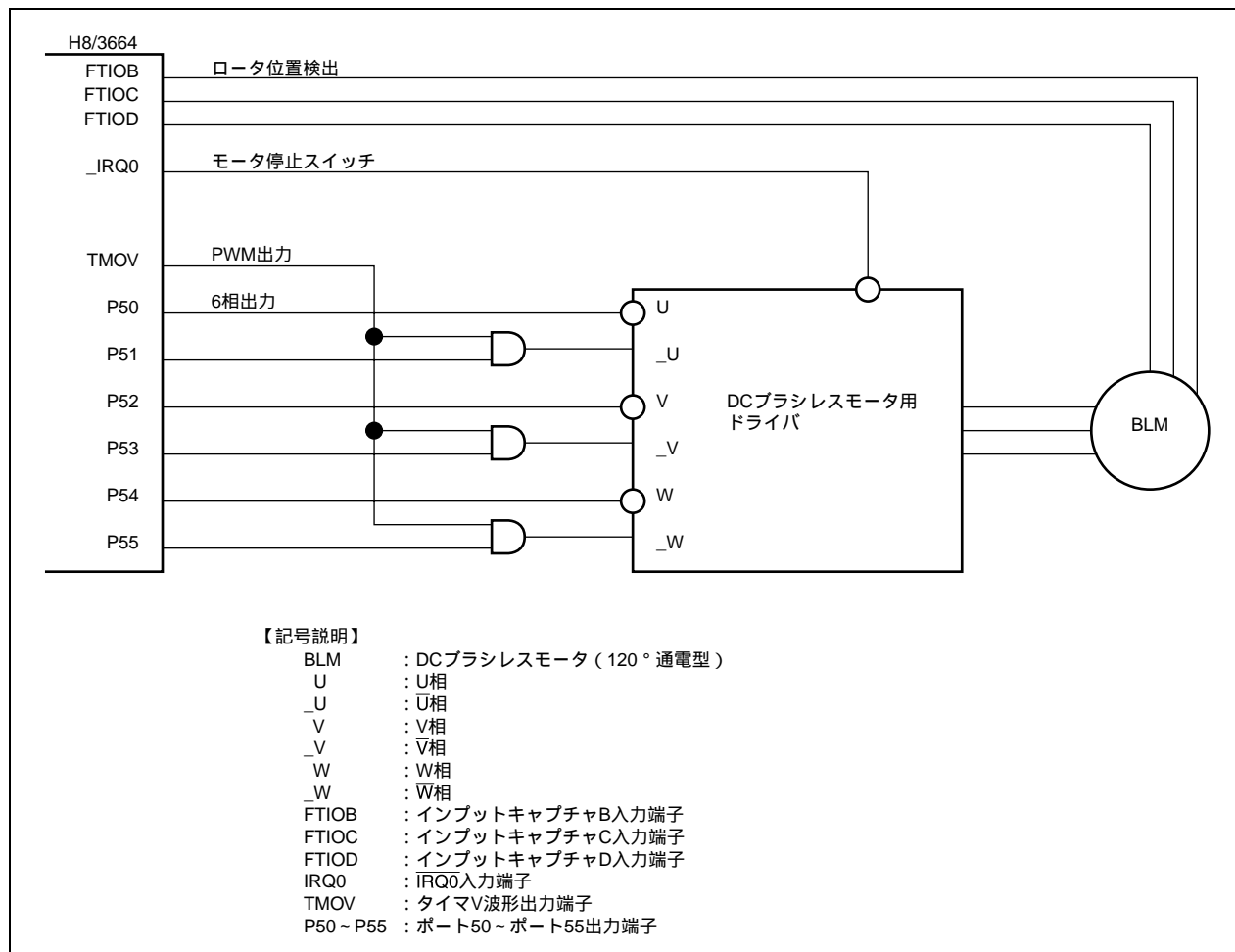


図 1.1 DC ブラシレスモータ制御

2. 考え方

1. DC ブラシレスモータを制御するための PWM 波形をタイマで生成し、I/O ポートより出力します。
2. モータの回り始めは、励磁する相を一定周期ごとに順に切換え、ポート出力します。
3. 励磁する相の切り換えを 6 回行なった後、モータより出力されるロータ位置信号を検出して制御する方法に遷移します。
4. モータより出力されるロータ位置検出信号は、タイマ W のインプットキャプチャ端子で取り込み、割り込みを発生させます。
5. 割り込みにより、回転磁界を切り換え、励磁する相をチョッピング制御します。

3. 使用機能説明

1. 図 3.1 に示すように H8/3664 のタイマ W (インプットキャプチャ、アウトプットコンペア機能)、タイマ V (コンペアマッチ機能)、I/O ポート (ポート 5 機能)、IRQ (外部割り込み機能) の各機能を割り付け、DC ブラシレスモータ制御を行います。

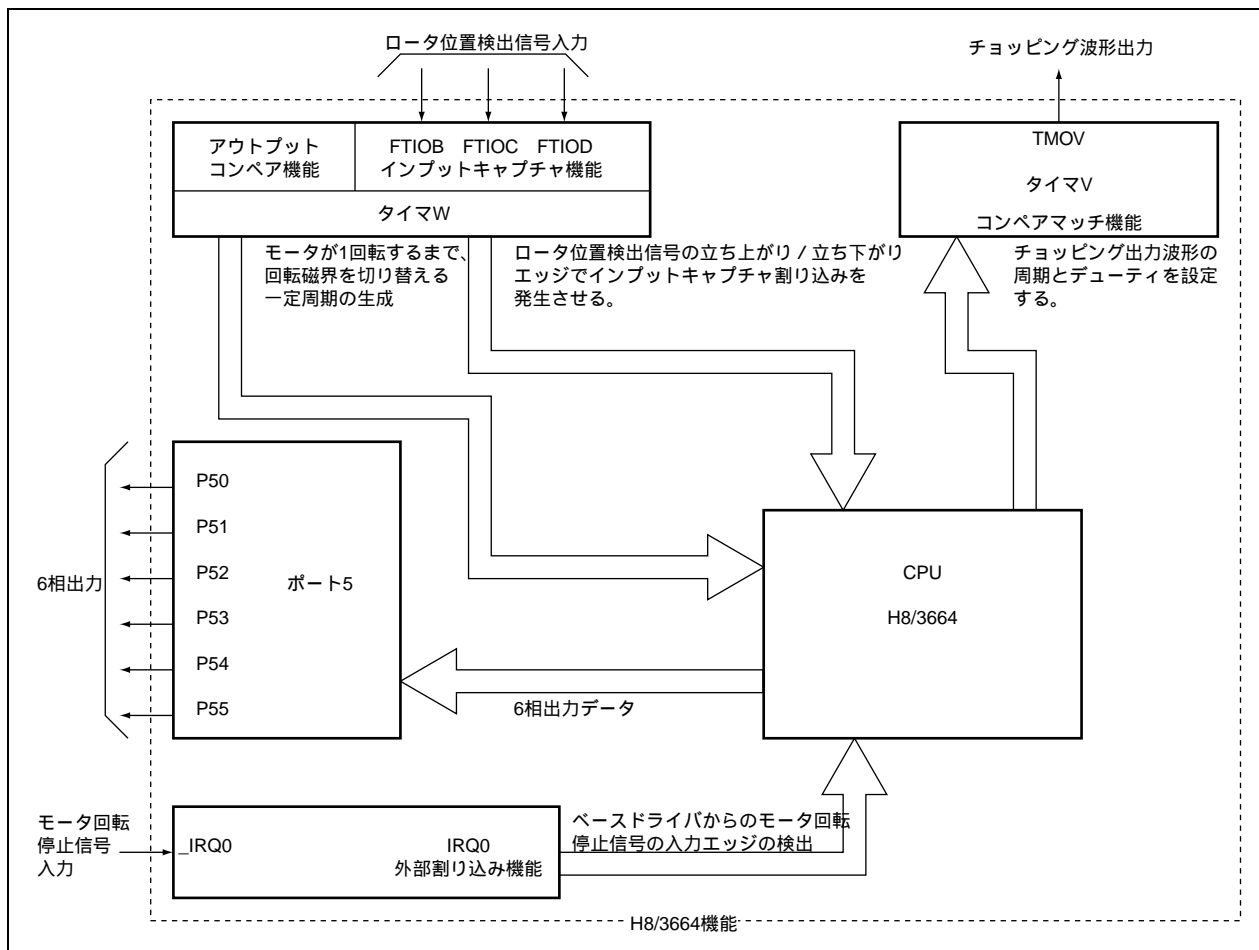


図 3.1 DC ブラシレスモータ制御ブロック図

以下に H8/3664 の機能割り付けについて説明します。

- タイマ W インプットキャプチャ機能：ロータ位置検出信号の立ち上がり／立ち下がりエッジを検出し、CPU に割り込みを要求します。
- タイマ W アウトプットコンペア機能：モータが 1 回転するまで、回転磁界切り換え周期（モータの回転周波数の 60° 分に相当する時間）ごとに CPU に割り込みを要求します。
- タイマ V コンペアマッチ機能：ドライバが ON する際にチョッピング制御を行なうためのチョッピング波形の生成を行い、TMOV 端子より出力します。
- ポート 5 機能：ドライバへ出力する 6 相のデータを出力します。
- IRQ 外部割り込み機能：外部からのモータ回転停止信号によりモータを停止させます。

2. 以下に各機能ごとに説明します。

- (a) タイマ **W** は、ロータ位置検出信号の立ち上がり／立ち下がりエッジを検出し、割り込みを発生させるインプットキャプチャ機能と、初期モータ制御時の回転磁界切換え周期を測定するためのアウトプットコンペア機能で使用します。

以下にタイマ **W** のインプットキャプチャ機能とアウトプットコンペア機能で共通に使用する機能について説明します。

- システムクロック (ϕ) は、CPU および周辺機能を動作させるための基準クロックです。システムクロックはプリスケアラ **S** によって $\phi/8192 \sim \phi/2$ に分周され、それぞれ各周辺モジュールに供給されます。
- プリスケアラ **S** (PSS) は、 ϕ を入力とする 13 ビットのカウンタで、1 サイクルごとにカウントアップします。
- タイマカウンタ (TCNT) は、16 ビットのリード／ライト可能なアップカウンタで、入力するクロックはシステムクロックの 2 分周、4 分周、8 分周および外部クロックの計 4 種のロックより選択可能です。TCNT の入力クロックは TCRW により設定します。本タスク例では TCNT の入力クロックにシステムクロックの 8 分周のクロックを選択しています。
- タイマコントロールレジスタ **W** (TCRW) は、8 ビットのリード／ライト可能なレジスタで、TCNT の入力クロックの選択を行ないます。
- タイマステータスレジスタ **W** (TSRW) は、8 ビットのレジスタで、各割り込み要求信号の制御を行ないます。
- タイマインタラプトイネーブルレジスタ **W** (TIERW) は、8 ビットのリード／ライト可能なレジスタで各割り込み要求の許可／禁止を制御します。本タスク例では、IMFA による割り込み要求 (IMIA)、IMFB による割り込み要求 (IMIB)、IMFC による割り込み要求 (IMIC)、IMFD による割り込み要求 (IMID) を許可しています。
- タイマ I/O コントロールレジスタ (TIOR0、TIOR1) は、GRA、GRB、GRC、GRD の機能の選択、インプットキャプチャ入力 **B**～**D** の入力エッジの選択を行います。

(b) タイマ W インพุットキャプチャ機能によりロータ位置検出信号の立ち上がり／立ち下がりエッジを検出し、割り込みを発生させます。図 3.2 にインพุットキャプチャ機能によるロータ位置検出信号の入力エッジによる割り込み要求のブロック図を示します。以下にブロックについて説明します。

- インพุットキャプチャ入力端子 B、C、D (FTIOB、FTIOC、FTIOD) は、ロータ位置検出信号の入力端子として機能します。
- インพุットキャプチャレジスタ B、C、D (GRB、GRC、GRD) は、16 ビットのリード／ライト可能なレジスタで、インพุットキャプチャ入力信号の入力エッジが検出されると、そのときの TCNT の値が GRB、GRC、GRD に転送され、TSRW の IMFB、IMFC、IMFD が 1 にセットされます。このとき、TIERW の IMIEB、IMIEC、IMIED が 1 ならば CPU に割り込みを要求します。

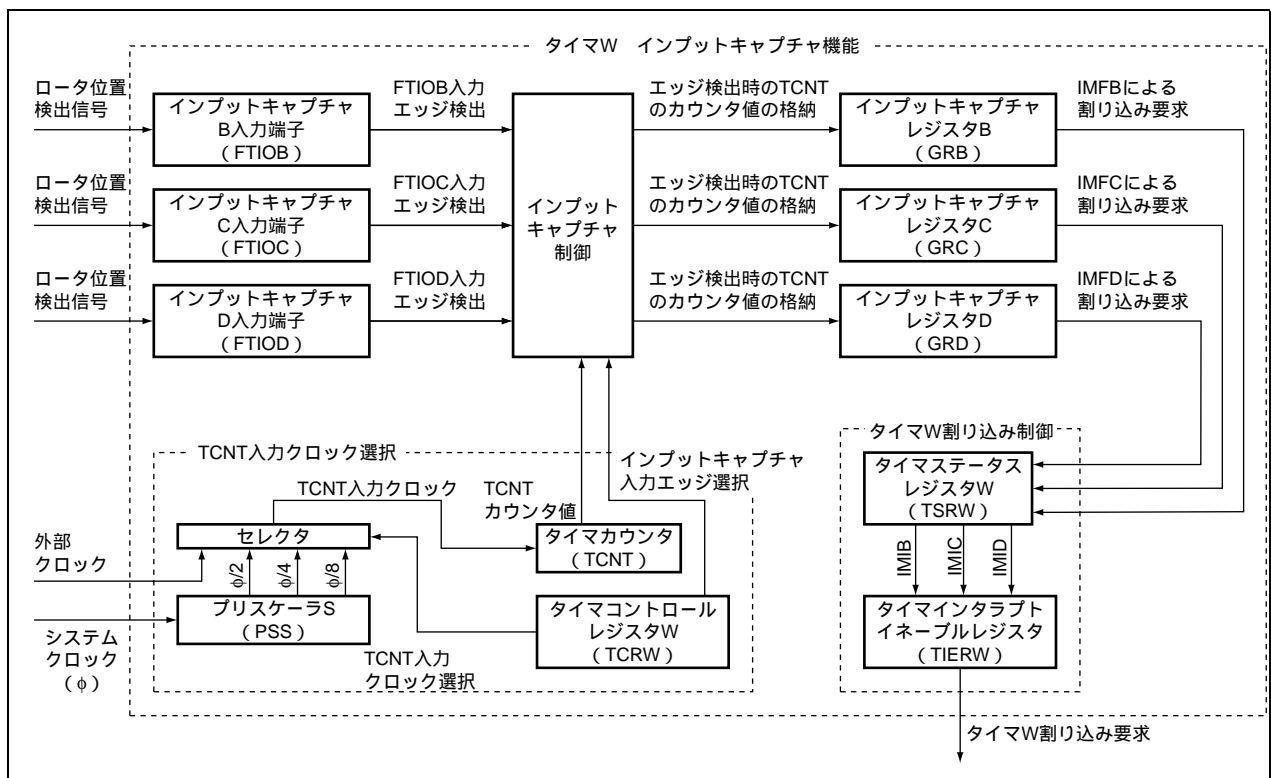


図 3.2 インพุットキャプチャ機能を使用したロータ位置検出信号の入力エッジ検出による割り込み要求ブロック図

(c) タイマ W アウトプットコンペア機能によりモータが 1 回転するまで、回転磁界切り換え周期（モータの回転周波数の 60° 分に相当する時間）ごとに CPU に割り込みを要求します。図 3.3 にタイマ W アウトプットコンペア機能を使用した回転磁界切り換え周期ごとの割り込み要求のブロック図を示します。

アウトプットコンペアレジスタ A (GRA) は、16 ビットのリード/ライト可能なレジスタです。GRA の内容は TCNT と常に比較されており、両者の値が一致すると TSRW の IMFA が 1 にセットされます。このとき、TIERW の IMIEA が 1 ならば CPU に割り込みを要求します。

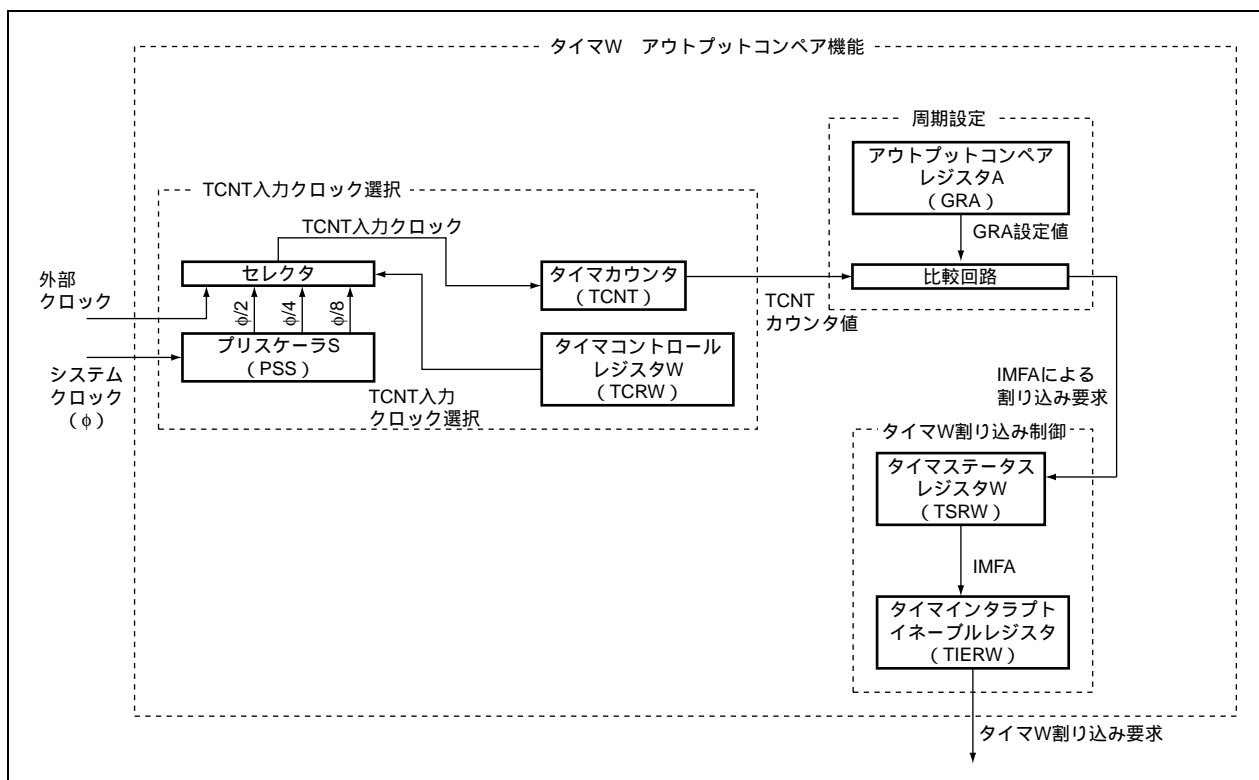


図 3.3 タイマ W アウトプットコンペア機能を使用した回転磁界切り換え周期ごとの割り込み要求ブロック図

- (d) タイマ V コンペアマッチ機能によりチョッピング波形を生成し、TMOV 端子より出力します。図 3.4 にタイマ V コンペアマッチ機能によるチョッピング波形出力のブロック図を示します。以下にブロック図について説明します。
- タイマカウンタ V (TCNTV) は、8 ビットのリード/ライト可能な 8 ビットカウンタで、入力する内部クロック/外部クロックによりカウントアップされます。入力するクロックは、TCRV0 の CKS2~CKS0 により選択します。
TCNTV の値は、CPU から常にリード/ライトできます。TCNTV の値は、外部リセット入力信号またはコンペアマッチ信号 A、B により、クリアすることができます。いずれの信号でクリアするかは、TCRV0 の CCLR1、CCLR0 により選択します。
 - タイムコンスタントレジスタ A、B (TCORA、B) は、8 ビットのリード/ライト可能なレジスタです。TCORA、B の内容は TCNTV と常に比較されており、両者の値が一致すると TCSR V の CMFA、B が 1 にセットされます。
このとき TCRV0 の CMIEA、B が 1 ならば CPU に割り込みを要求します。ただし、TCORA、B へのライトサイクルの T₃ ステートでの比較は禁止されています。
 - タイムコントロールレジスタ V0 (TCRV0) は、8 ビットのリード/ライト可能なレジスタで、TCNTV の入力クロックの選択、TCNTV のクリア指定、および割り込み要求の許可を行いません。本タスクでは、TCNTV のクリア指定はコンペアマッチ A によるクリアを選択しています。TCNTV の入力クロックは、TCRV1 の ICKS0 との組み合わせにより選択しています。
 - タイムコントロール/ステータスレジスタ V (TCSR V) は、8 ビットのレジスタで、コンペアマッチフラグのセット、およびコンペアマッチ出力の制御を行いません。本タスク例では、コンペアマッチ B により 0 出力、コンペアマッチ A により 1 出力を選択しています。
 - タイムコントロールレジスタ V1 (TCRV1) は、8 ビットのリード/ライト可能なレジスタで、TCNTV の入力クロックの選択を行いません。
 - タイマ V 出力端子 (TMOV) によりチョッピング波形を出力します。

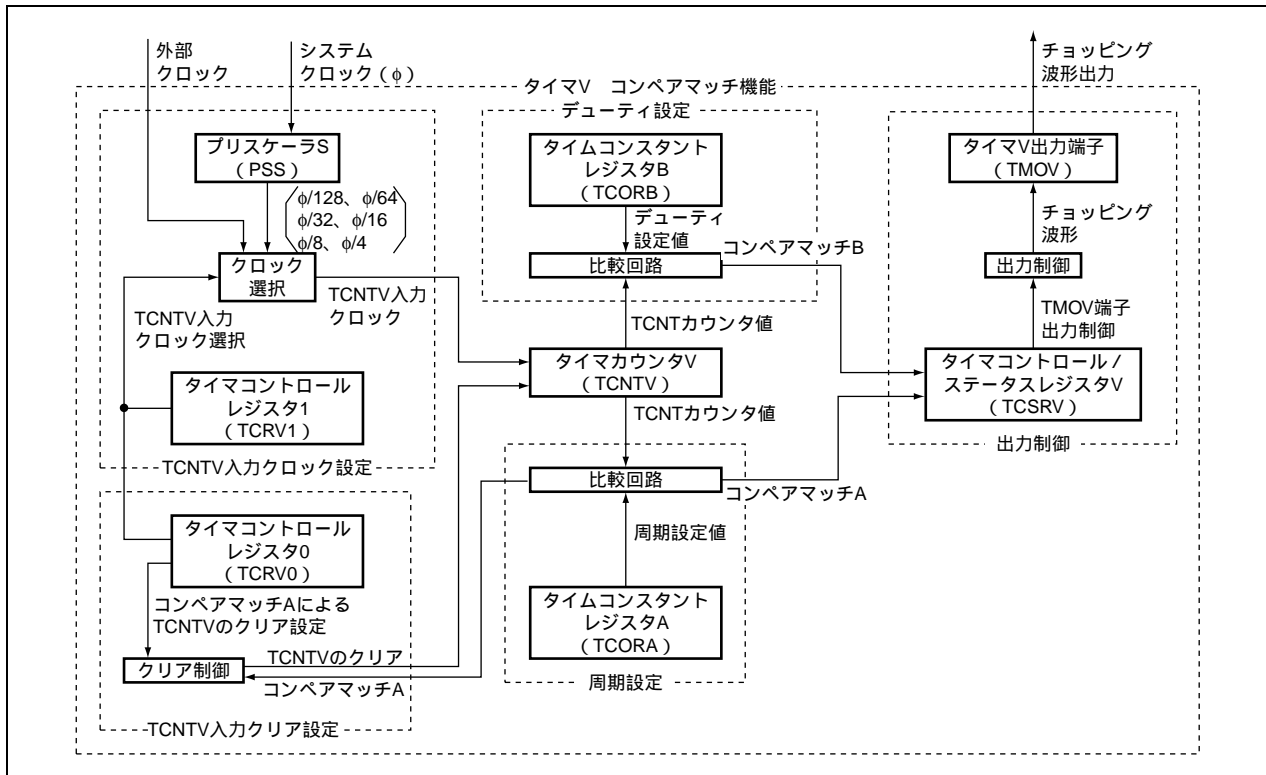


図 3.4 タイマVコンパアマッチ機能を使用したチョッピング波形出力ブロック図

(e) ポート 5 機能により 6 相出力を行ないます。図 3.5 にポート 5 機能による 6 相出力のブロック図を示します。以下にブロック図について説明します。

- ポート 50 端子 (P50) は、ベースドライバの U 相への出力端子として機能します。
- ポート 51 端子 (P51) は、ベースドライバの \bar{U} 相への出力端子として機能します。
- ポート 52 端子 (P52) は、ベースドライバの V 相への出力端子として機能します。
- ポート 53 端子 (P53) は、ベースドライバの \bar{V} 相への出力端子として機能します。
- ポート 54 端子 (P54) は、ベースドライバの W 相への出力端子として機能します。
- ポート 55 端子 (P55) は、ベースドライバの \bar{W} 相への出力端子として機能します。
- ポートデータレジスタ 5 (PDR5) は、ポート 5 の各端子 P50~P56 のデータを格納する 8 ビットのレジスタです。PCR5 が 1 のとき、ポート 5 のリードを行なうと、PDR5 の値を直接リードします。

そのため端子状態の影響を受けません。PCR5 が 0 のとき、ポート 5 のリードを行なうと、端子状態が読み出されます。

- ポートコントロールレジスタ 5 (PCR5) は、ポート 5 の各端子 P50~P56 の入出力をビットごとに制御します。本タスク例では、端子 P50~P55 を出力端子機能に設定します。

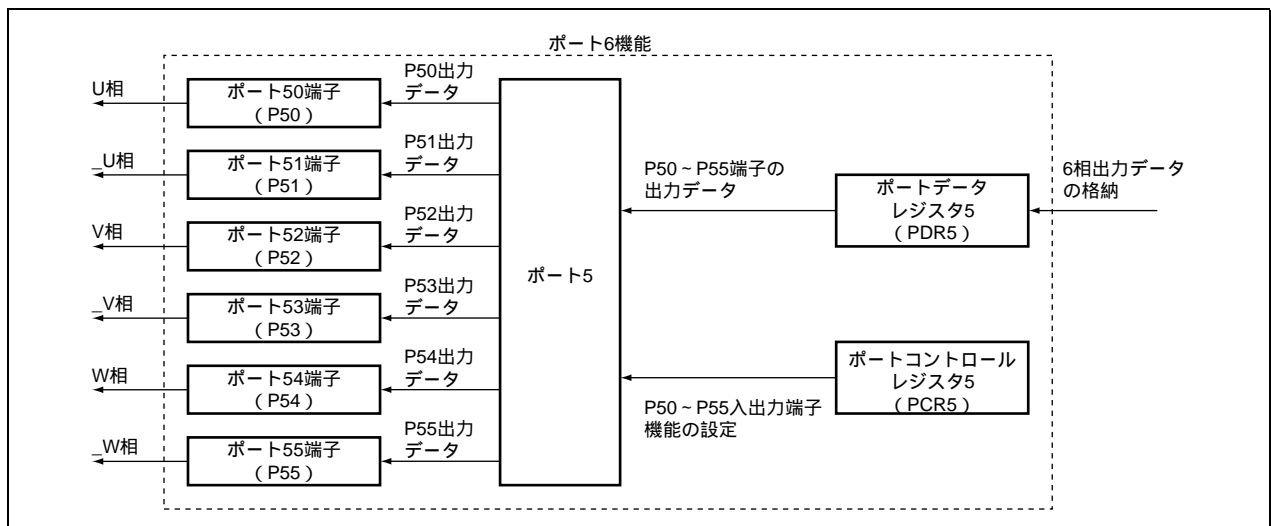


図 3.5 ポート 5 を使用した 6 相出力ブロック図

(f) IRQ 外部割り込み機能によりモータ回転停止信号の検出を行いません。図 3.6 に IRQ 外部割り込み機能を使用したモータ回転停止信号の検出のブロック図を示します。以下にブロック図について説明します。

- IRQ0 端子は外部からのモータ回転停止信号の入力端子として機能します。
- $_IRQ0$ 端子の入力エッジの検出により IRQ0 割り込みを要求します。 $_IRQ0$ 端子の入力エッジセンスは立ち上がり／立ち下がりより選択可能です。
- 割り込みエッジセレクトレジスタ 1 (IEGR1) は、8 ビットのリード／ライト可能なレジスタで、 $_IRQ0$ 端子の立ち上がり／立ち下がりエッジを指定します。
- 割り込み許可レジスタ 1 (IENR1) は、8 ビットのリード／ライト可能なレジスタで、割り込み要求の許可／禁止を制御します。本タスク例では、IRQ0 割り込み要求を許可し、タイマ A、IRQ1、IRQ2、IRQ3 割り込み要求は禁止します。
- 割り込み要求レジスタ 1 (IRR1) は、8 ビットのリード／ライト可能なレジスタで、タイマ A、IRQ0～IRQ3 割り込み要求が発生すると対応するフラグが 1 にセットされます。各割り込み要求フラグは割り込みが受け付けられてもオートクリアされません。各割り込み要求フラグをクリアする場合は 0 をライトしてください。

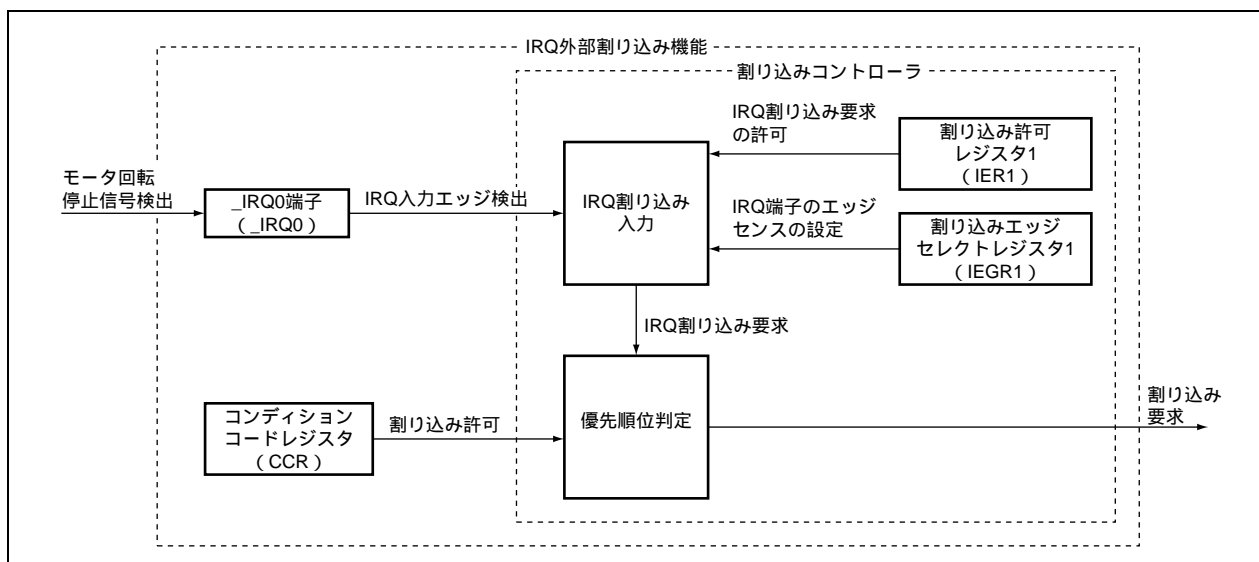


図 3.6 IRQ 外部割り込み機能を使用したモータ回転停止信号検出のブロック図

3. 表 3.1 に本タスク例の機能割付けを示します。表 3.1 に示すように H8/3664 の機能を割り付け、DC ブラシレスモータ制御を行ないます。

表 3.1 H8/3664 機能割り付け

機 能	機能割り付け
FTIOB、C、D	ロータ位置検出信号入力
TCNT	16 ビットアップカウンタ、入カクロックは TCRW により設定
TCRW	TCNT 入カクロックの設定。TCNT クリア要因の選択
GRA	初期モータ回転磁界切換え周期の設定
TSRW	タイマ W の割り込み要求 (IMIA、IMIB、IMIC、IMID) の有無を反映
TIERW	タイマ W の割り込み要求 (IMIA、IMIB、IMIC、IMID) 許可または禁止を設定
TMOV	チョッピング波形出力
TCNTV	8 ビットアップカウンタ
TCORA	チョッピング出力波形の周期を設定
TCORB	チョッピング出力波形のデューティを設定
TCRV0	TCNTV 入カクロックの設定、TCNTV のクリア条件の設定、タイマ V の割り込み要求 (CMIA、CMIB) の許可または禁止を設定
TCRV1	TCNTV 入カクロックの設定
TCSRv	タイマ V の割り込み要求 (CMIA、CMIB) の有無を反映、コンペアマッチ出力の設定
P50~P55	6 相波形出力
PCR5	P50~P55 出力端子機能の設定
PDR5	P50~P55 出力端子のデータの格納、端子レベルの読み出し
_IRQ0	モータ回転停止信号入力
IEGR1	_IRQ0 端子の入力エッジの設定
IER1	IRQ0 割り込み要求の有無を反映

4. 動作説明

1. 図 4.1 に初期モータ制御時（モータが 1 回転するまで、一定周期ごとに回転磁界を切り換えモータを回転させる）の動作原理を示します。図 4.1 に示すように H8/3664 のハードウェア処理、およびソフトウェア処理により初期モータ制御時の DC ブラシレスモータ制御を行ないます。

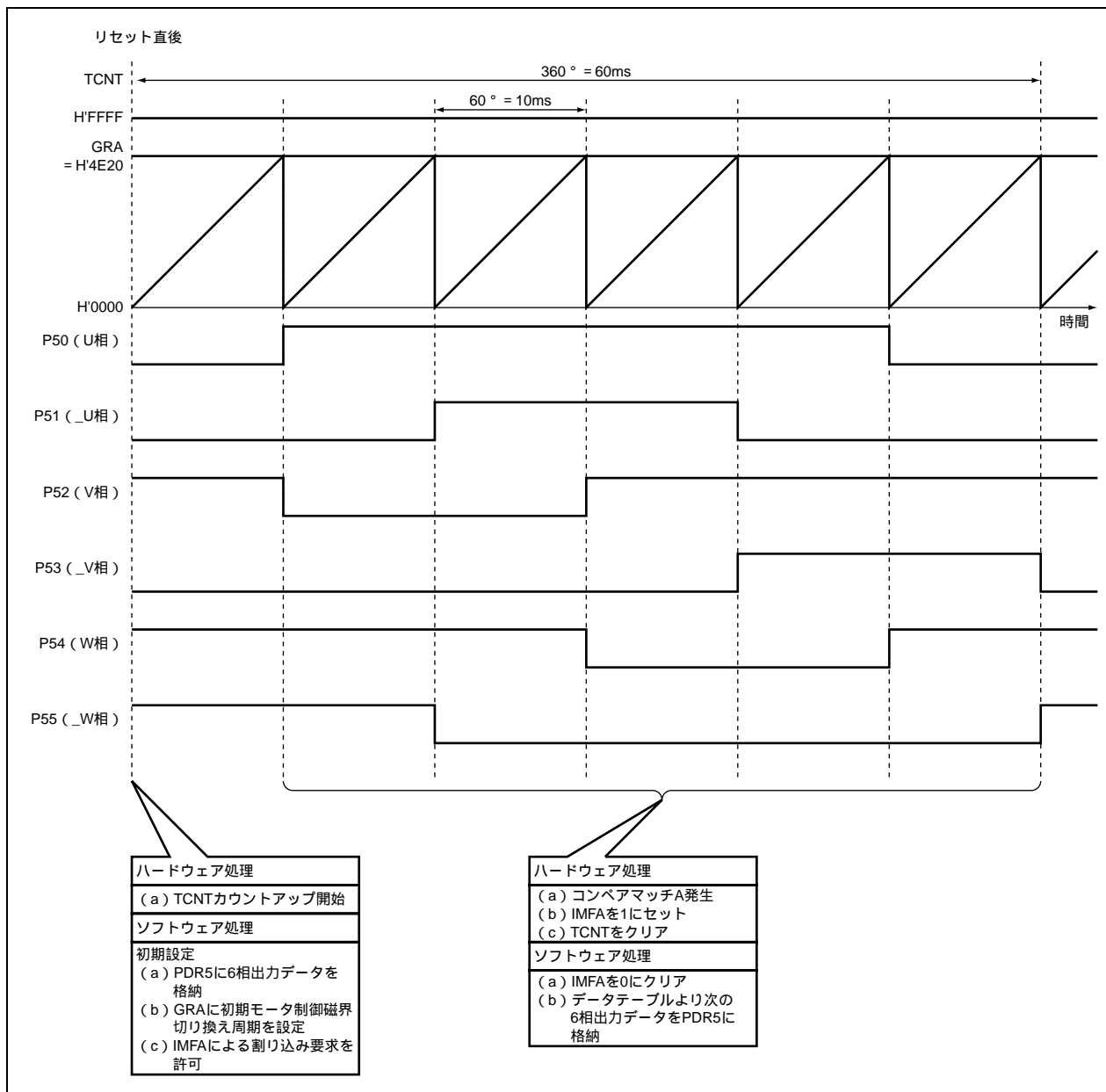


図 4.1 初期モータ制御時の DC ブラシレスモータ制御の動作原理

2. 図 4.2 にロータ位置信号の検出により回転磁界の切り換えを行なうことによる DC ブラシレスモータ制御の動作原理を示します。図 4.2 に示すように H8/3664 のハードウェア処理、およびソフトウェア処理によりロータ位置信号検出による DC ブラシレスモータ制御を行ないます。

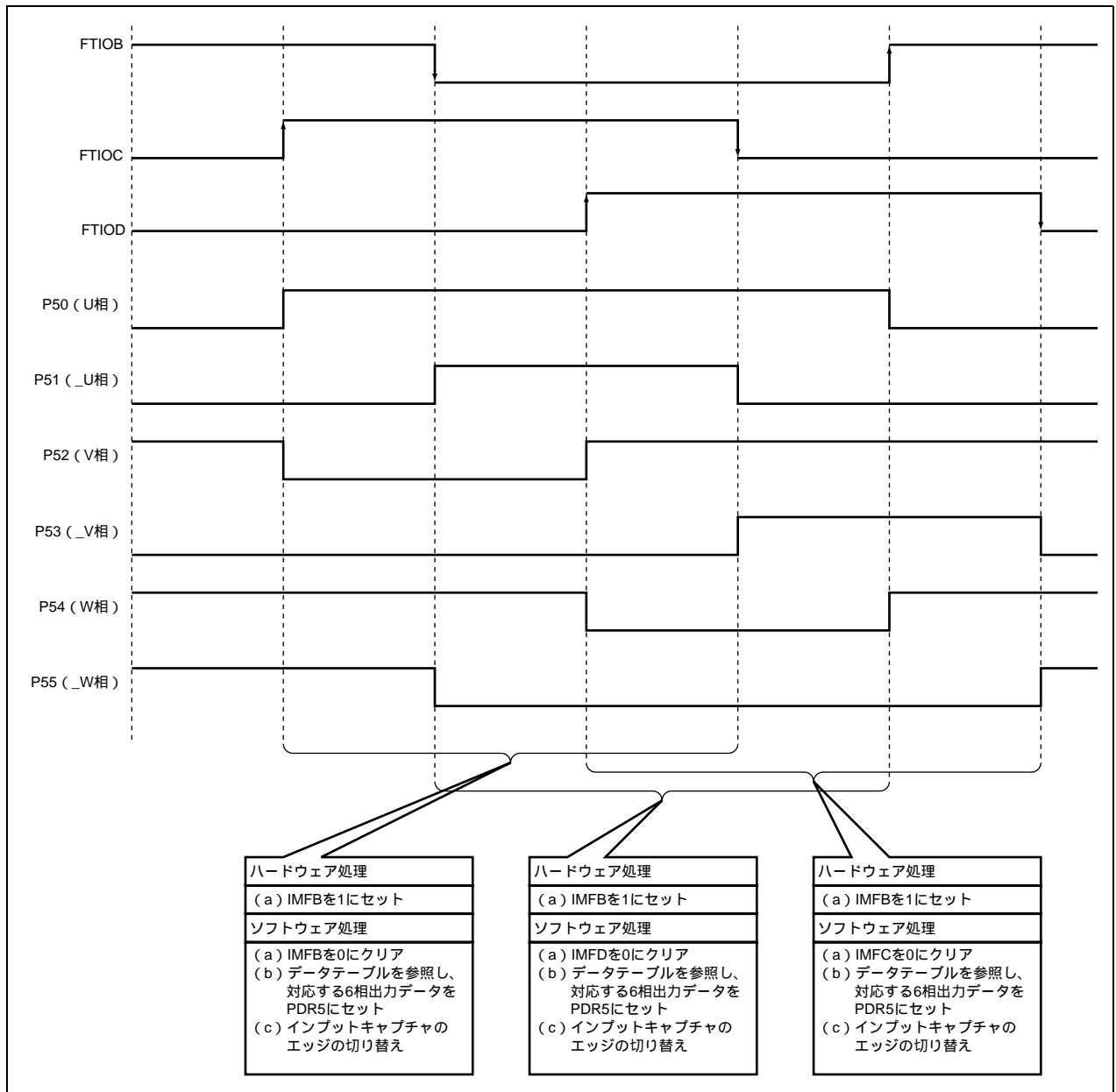


図 4.2 ロータ位置信号検出による DC ブラシレスモータ制御の動作原理

3. 本制御例では、逆相側ドライバが ON する際にチョッピング制御を行いません。チョッピング波形はタイマ V コンペアマッチ機能により生成し、TMOV 端子から出力します。図 4.3 にチョッピング波形出力の動作原理を示します。

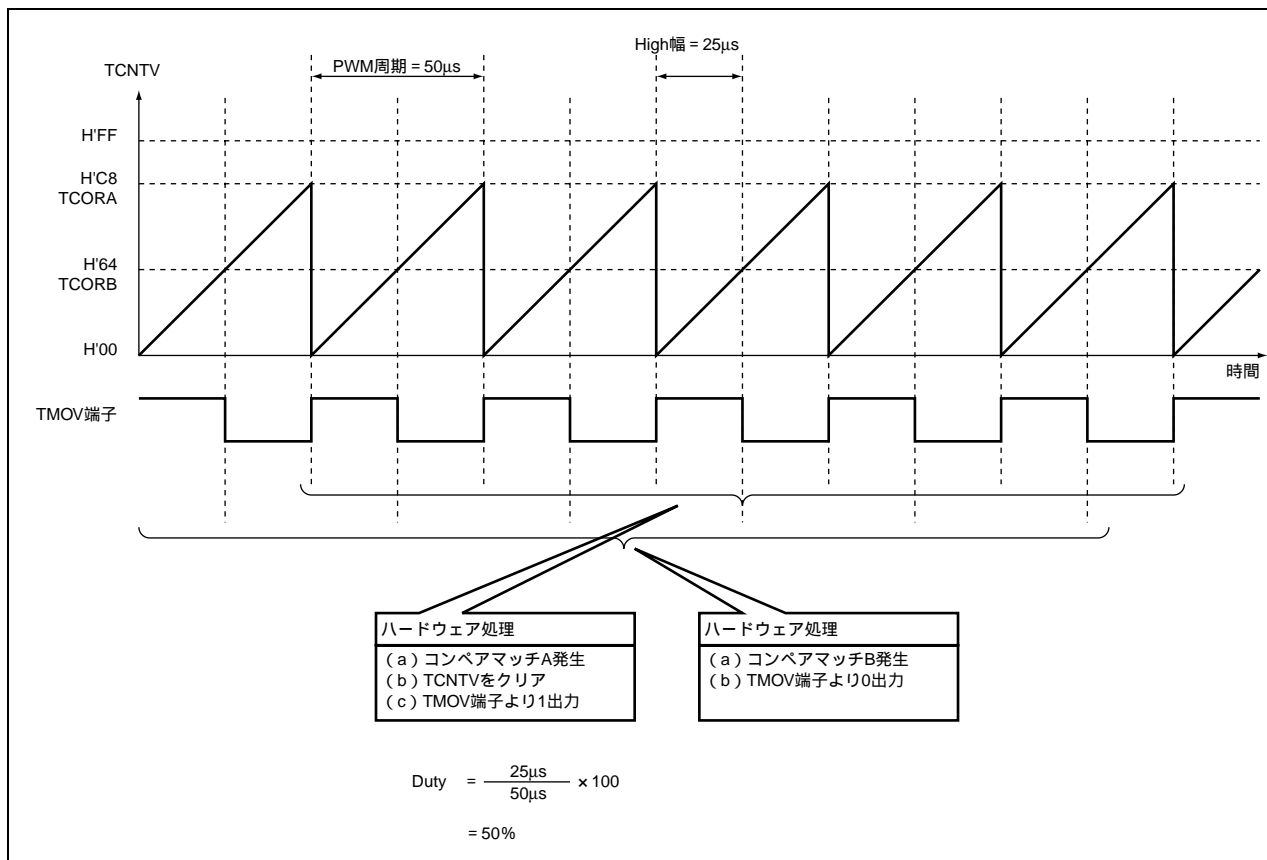


図 4.3 チョッピング波形出力の動作原理

本タスク例では、PWM 周期 (TCORA 設定値) を 50µs (H'C8)、および High 幅 (TCORB 設定値) を 25µs (H'64) の値に固定し、Duty=50%による制御を行なっています。

4. 6相出力の逆相側に外付けのAND回路を設け、チョッピング波形(TMOV端子)と回転磁界波形(ポート出力)のANDをとり、ドライバへ出力します。図4.4にチョッピング波形と回転磁界波形のANDで6相出力を行なう際の動作原理を示します。

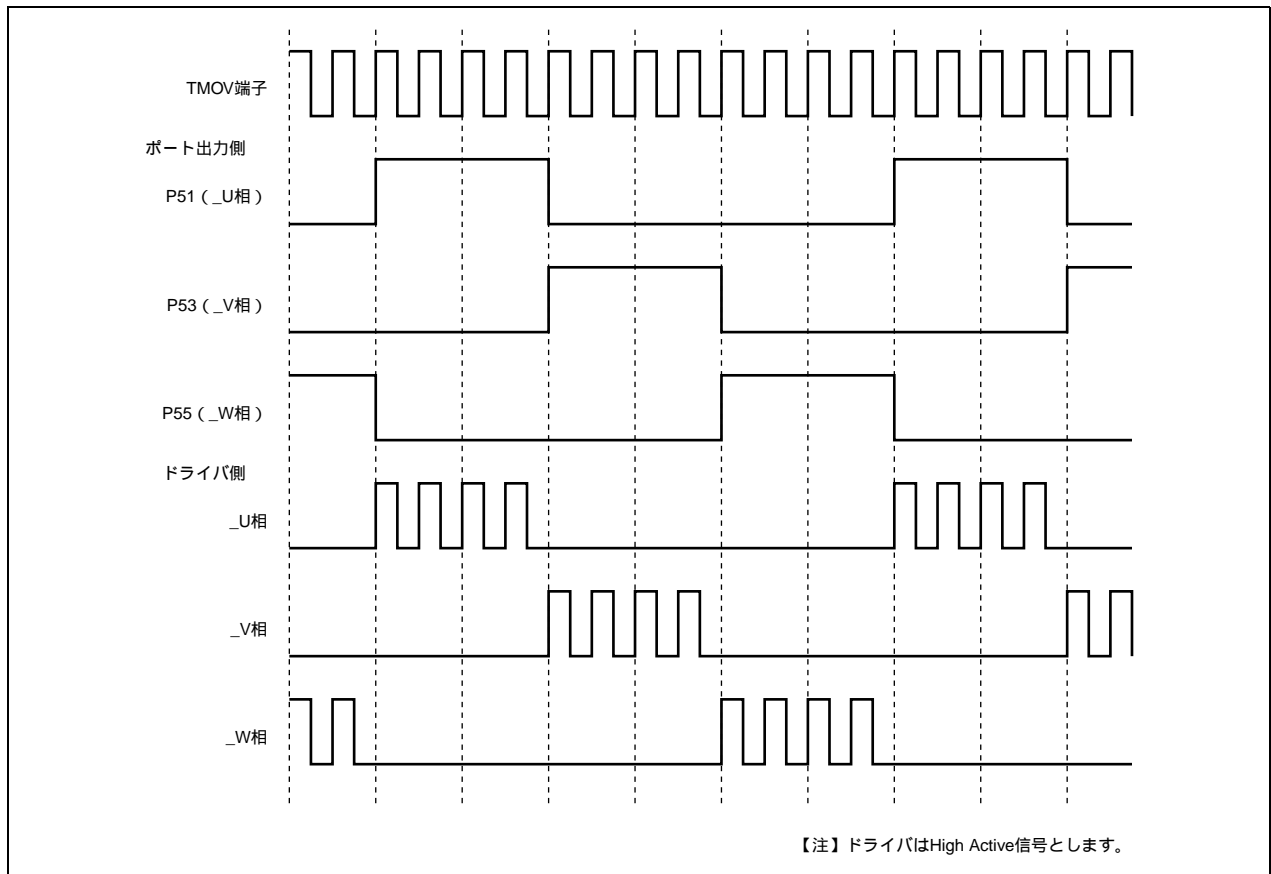


図 4.4 TMOV 端子とポートの AND による逆相側出力の動作原理

5. ソフトウェア説明

5.1 モジュール説明

表 5.1 に本タスク例におけるモジュール説明を示します。

表 5.1 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	スタックポインタのイニシャライズ、IRQ0 割り込みの設定、タイマカウンタのスタート、および各割り込みの設定を行なう。
イニシャライズ	IO_INIT	各レジスタのイニシャライズ、タイマ W アウトプットコンペアの設定、タイマ V アウトプットコンペアの設定、使用 RAM の設定を行う。
タイマ X 割り込み処理ルーチン	intpt_tmw	タイマ W 割り込み処理ルーチンで、割り込み要求フラグのクリアおよび 6 相出力の制御を行なう。
初期モータ制御ルーチン	tmw_3	モータが 1 回転するまで、一定周期毎に回転磁界を切り替えモータを回転させる
フィードバック信号 (FTIOB) による回転磁界切り換えルーチン	SA_1	モータから FTIOB 端子に出力されるロータ位置検出信号により、励磁する相の切り換えを行う。
フィードバック信号 (FTIOC) による回転磁界切り換えルーチン	SB_1	モータから FTIOC 端子に出力されるロータ位置検出信号により、励磁する相の切り換えを行う。
フィードバック信号 (FTIOD) による回転磁界切り換えルーチン	SC_1	モータから FTIOD 端子に出力されるロータ位置検出信号により、励磁する相の切り換えを行う。
IRQ0 割り込み処理ルーチン	r_stop	IRQ0 割り込み処理ルーチンで、割り込み要求フラグのクリアおよびエラーフラグをたて、モータ停止処理を行なう。

5.2 引数の説明

本タスクでは引数は使用していません。

5.3 内部使用レジスタ説明

表 5.2 に本タスク例における内部使用レジスタ説明を示します。

表 5.2 内部使用レジスタ説明

レジスタ名		機 能	アドレス	設定値
PDR5		ポートデータレジスタ 5 : P5n が 0 のとき、P5n 端子レベルは Low (n=0~7) : P5n が 1 のとき、P5n 端子レベルは High (n=0~7)	H'FFD8	H'00
PCR5		ポートコントロールレジスタ 5 : PCR5n が 0 のとき、P5n 端子は入力端子機能 (n=0~7) : PCR5n が 1 のとき、P5n 端子は出力端子機能 (n=0~7)	H'FFE8	H'FF
TSRW	IMFA	タイマコントロール/ステータスレジスタ (コンペアマッチフラグ A) : IMFA=0 のとき、コンペアマッチ A が発生していないことを示す : IMFA=1 のとき、コンペアマッチ A が発生したことを示す	H'FF83 ビット 0	0
	IMFB	タイマコントロール/ステータスレジスタ (インプットキャプチャフラグ B) : IMFB=0 のとき、インプットキャプチャ B が発生していないことを示す : IMFB=1 のとき、インプットキャプチャ B 発生したことを示す	H'FF83 ビット 1	0
	IMFC	タイマコントロール/ステータスレジスタ (インプットキャプチャフラグ C) : IMFC=0 のとき、インプットキャプチャ C が発生していないことを示す : IMFC=1 のとき、インプットキャプチャ C 発生したことを示す	H'FF83 ビット 2	0
	IMFD	タイマコントロール/ステータスレジスタ (インプットキャプチャフラグ D) : IMFD=0 のとき、インプットキャプチャ D が発生していないことを示す : IMFD=1 のとき、インプットキャプチャ D 発生したことを示す	H'FF83 ビット 3	0
TIERW	IMIEA	タイマインタラプトイネーブルレジスタ (コンペアマッチ割り込みイネーブル A) : IMIEA=0 のとき、IMFA による割り込み要求を禁止 : IMIEA=1 のとき、IMFA による割り込み要求を許可	H'FF82 ビット 0	1
	IMIEB	タイマインタラプトイネーブルレジスタ (インプットキャプチャ割り込みイネーブル B) : IMIEB=0 のとき、IMFB による割り込み要求を禁止 : IMIEB=1 のとき、IMFB による割り込み要求を許可	H'FF82 ビット 1	0
	IMIEC	タイマインタラプトイネーブルレジスタ (インプットキャプチャ割り込みイネーブル C) : IMIEC=0 のとき、IMFC による割り込み要求を禁止 : IMIEC=1 のとき、IMFC による割り込み要求を許可	H'FF82 ビット 2	0
	IMIED	タイマインタラプトイネーブルレジスタ (インプットキャプチャ割り込みイネーブル D) : IMIED=0 のとき、IMFD による割り込み要求を禁止 : IMIED=1 のとき、IMFD による割り込み要求を許可	H'FF82 ビット 3	0

表 5.2 使用内部レジスタ説明 (つづき)

レジスタ名		機 能	アドレス	設定値
TIOR0	IOA2	I/O コントロール A2 : 0 のとき、GRA はアウトプットコンペアレジスタとして機能 : 1 のとき、GRA はインプットキャプチャレジスタとして機能	H'FF84 ビット 2	0
	IOB0 IOB1	I/O コントロール B1~0 IOB2=1 の場合 : 00 のとき、FTIOB 端子の立上がりエッジで GRB ヘインプットキャプチャ : 01 のとき、FTIOB 端子の立下がりエッジで GRB ヘインプットキャプチャ	H'FF84 ビット 4 ビット 5	IOB0=1 IOB1=0
	IOB2	I/O コントロール B2 : 0 のとき、GRB はアウトプットコンペアレジスタとして機能 : 1 のとき、GRB はインプットキャプチャレジスタとして機能	H'FF84 ビット 6	1*1
TIOR1	IOC0 IOC1	I/O コントロール C1~0 IOC2=1 の場合 : 00 のとき、FTIOC 端子の立上がりエッジで GRC ヘインプットキャプチャ : 01 のとき、FTIOC 端子の立下がりエッジで GRC ヘインプットキャプチャ	H'FF85 ビット 0 ビット 1	IOC0=0 IOC1=0
	IOC2	I/O コントロール C2 : 0 のとき、GRC はアウトプットコンペアレジスタとして機能 : 1 のとき、GRC はインプットキャプチャレジスタとして機能	H'FF85 ビット 2	1*1
	IOD0 IOD1	I/O コントロール D1~0 IOD2=1 の場合 : 00 のとき、FTIOD 端子の立上がりエッジで GRD ヘインプットキャプチャ : 01 のとき、FTIOD 端子の立下がりエッジで GRD ヘインプットキャプチャ	H'FF85 ビット 4 ビット 5	IOD0=1 IOD1=0
	IOD2	I/O コントロール D2 : 0 のとき、GRD はアウトプットコンペアレジスタとして機能 : 1 のとき、GRD はインプットキャプチャレジスタとして機能	H'FF85 ビット 6	1
GRA		ジェネラルレジスタ A : アウトプットコンペアレジスタとして使用。TCNT の値と常に比較され、両者が一致 (コンペアマッチ) すると TSRW の IMFA フラグが 1 にセットされる。	H'FF88	H'4E20
GRB		ジェネラルレジスタ B : インプットキャプチャレジスタとして使用。FTIOB からのインプットキャプチャ信号を検出すると TCNT の値が格納され、TSRW の IMFB フラグが 1 にセットされる。	H'FF8A	H'0000
GRC		ジェネラルレジスタ C : インプットキャプチャレジスタとして使用。FTIOC からのインプットキャプチャ信号を検出すると TCNT の値が格納され、TSRW の IMFC フラグが 1 にセットされる。	H'FF8C	H'0000
GRD		ジェネラルレジスタ D : インプットキャプチャレジスタとして使用。FTIOD からのインプットキャプチャ信号を検出すると TCNT の値が格納され、TSRW の IMFD フラグが 1 にセットされる。	H'FF8E	H'0000
TCSR	OS0 OS1	タイマコントロール/ステータスレジスタ V (アウトプットセレクト 1~0) : 01 のとき、TCORA と TCNTV のコンペアマッチにより TMOV 端子より 0 を出力。 : 01 のとき、TCORA と TCNTV のコンペアマッチにより TMOV 端子より 1 を出力。	H'FFA1 ビット 0 ビット 1	OS1=1 OS0=0
	OS2 OS3	タイマコントロール/ステータスレジスタ V (アウトプットセレクト 3~2) : 01 のとき、TCORB と TCNTV のコンペアマッチにより TMOV 端子より 0 を出力。 : 01 のとき、TCORB と TCNTV のコンペアマッチにより TMOV 端子より 1 を出力。	H'FFA1 ビット 2 ビット 3	OS2=1 OS3=0

表 5.2 使用内部レジスタ説明 (つづき)

レジスタ名		機 能	アドレス	設定値
TCRV0	CCLR0 CCLR1	カウンタクリア 1~0 : 01 のとき、コンペアマッチ A で TCNTV がクリアされる。 : 10 のとき、コンペアマッチ B で TCNTV がクリアされる。	H'FFA0 ビット 0 ビット 1	CCLR0=0 CCLR1=1
TCORA		タイムコンスタントレジスタ A : TCORA=H'68 のとき、TCNTV が H'68 までカウントアップすると コンペアマッチ A が発生します。	H'FFA2	H'68
TCORB		タイムコンスタントレジスタ B : TCORB=H'64 のとき、TCNTV が H'64 までカウントアップすると コンペアマッチ B が発生します。	H'FFA3	H'64
IEGR1	IEG0	インタラプトエッジセレクトレジスタ 1 (IRQ0 エッジセレクト) : IEG0=0 のとき、IRQ0 端子入力の立ち下がりエッジを検出 : IEG0=1 のとき、IRQ0 端子入力の立ち上がりエッジを検出	H'FFF2 ビット 0	0
IENR1	IEN0	インタラプトイネーブルレジスタ 1 (IRQ0 インタラプトイネーブル) : IEN0=0 のとき、IRQ0 割り込み要求を禁止 : IEN0=1 のとき、IRQ0 割り込み要求を許可	H'FFF4 ビット 0	0
IRR1	IRRI0	インタラプトリクエストレジスタ 1 (IRQ0 インタラプトリクエストフラグ) : IRRI0=0 のとき、IRQ0 割り込みが要求されていないことを示す : IRRI0=1 のとき、IRQ0 割り込みが要求されていることを示す	H'FFF6 ビット 0	0

【注】 *1 初期起動時は“0”

5.4 使用 RAM 説明

表 5.3 に本タスク例における使用 RAM 説明を示します。

表 5.3 使用 RAM 説明

ラベル名	機 能	アドレス	使用モジュールラベル名
x	初期モータ制御時データテーブル参照用カウンタ	H'FB80	IO_INIT
y	タイマ機能切り換え用カウンタ	H'FB81	IO_INIT

5.5 データテーブル説明

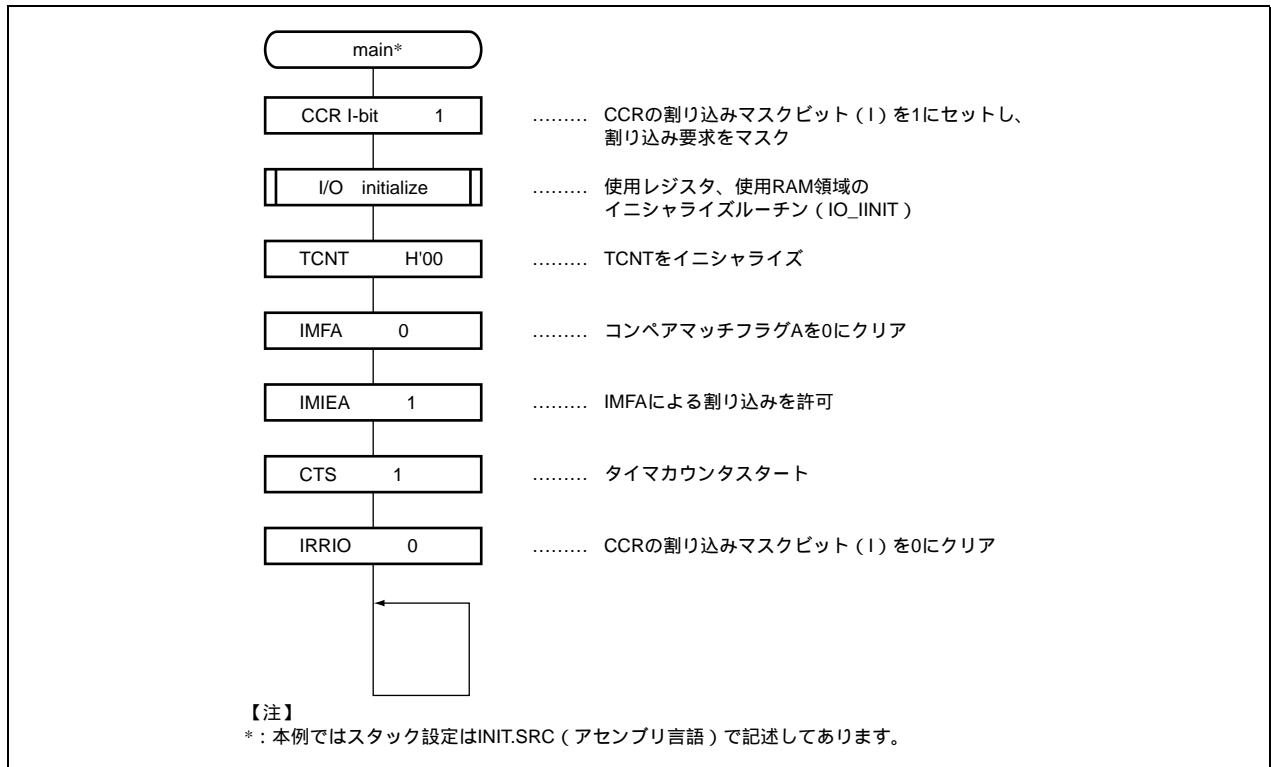
本タスクでは、初期モータ制御時にデータテーブル上の6パターンのデータを使用しポート5に出力します。表5.4に使用データテーブルの説明を示します。

表 5.4 データテーブル(data_0)説明

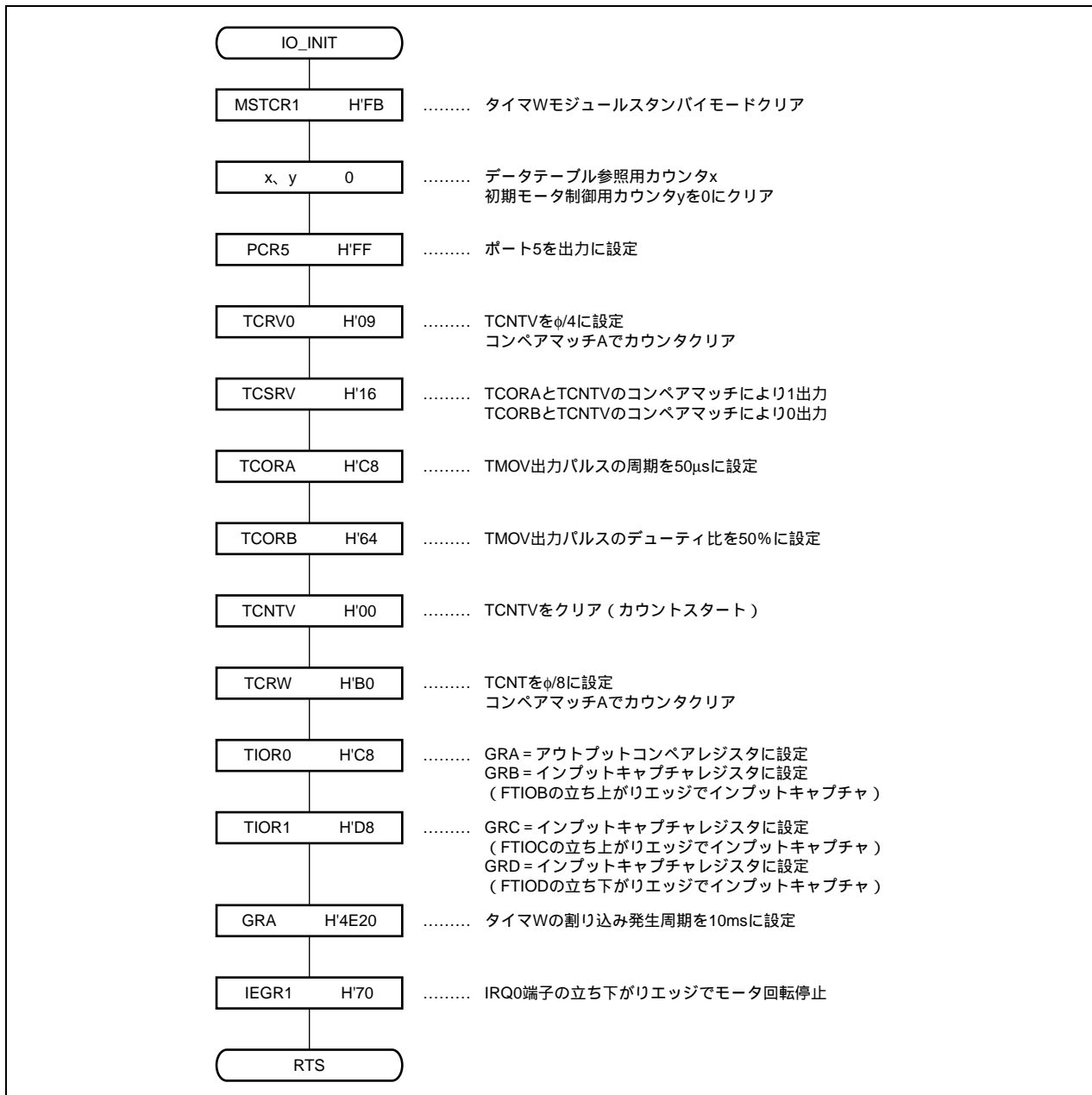
データ名	データ	出力パターン	サイズ	アドレス
data_0[0]	H'34	U=L、_U=L、V=H、_V=L、W=H、_W=H	1Byte	H'02D6
data_0[1]	H'31	U=H、_U=L、V=L、_V=L、W=H、_W=H	1Byte	H'02D7
data_0[2]	H'13	U=H、_U=H、V=L、_V=L、W=H、_W=L	1Byte	H'02D8
data_0[3]	H'07	U=H、_U=H、V=H、_V=L、W=L、_W=L	1Byte	H'02D9
data_0[4]	H'0D	U=H、_U=L、V=H、_V=H、W=L、_W=L	1Byte	H'02DA
data_0[5]	H'1C	U=L、_U=L、V=H、_V=H、W=H、_W=L	1Byte	H'02DB

6. フローチャート

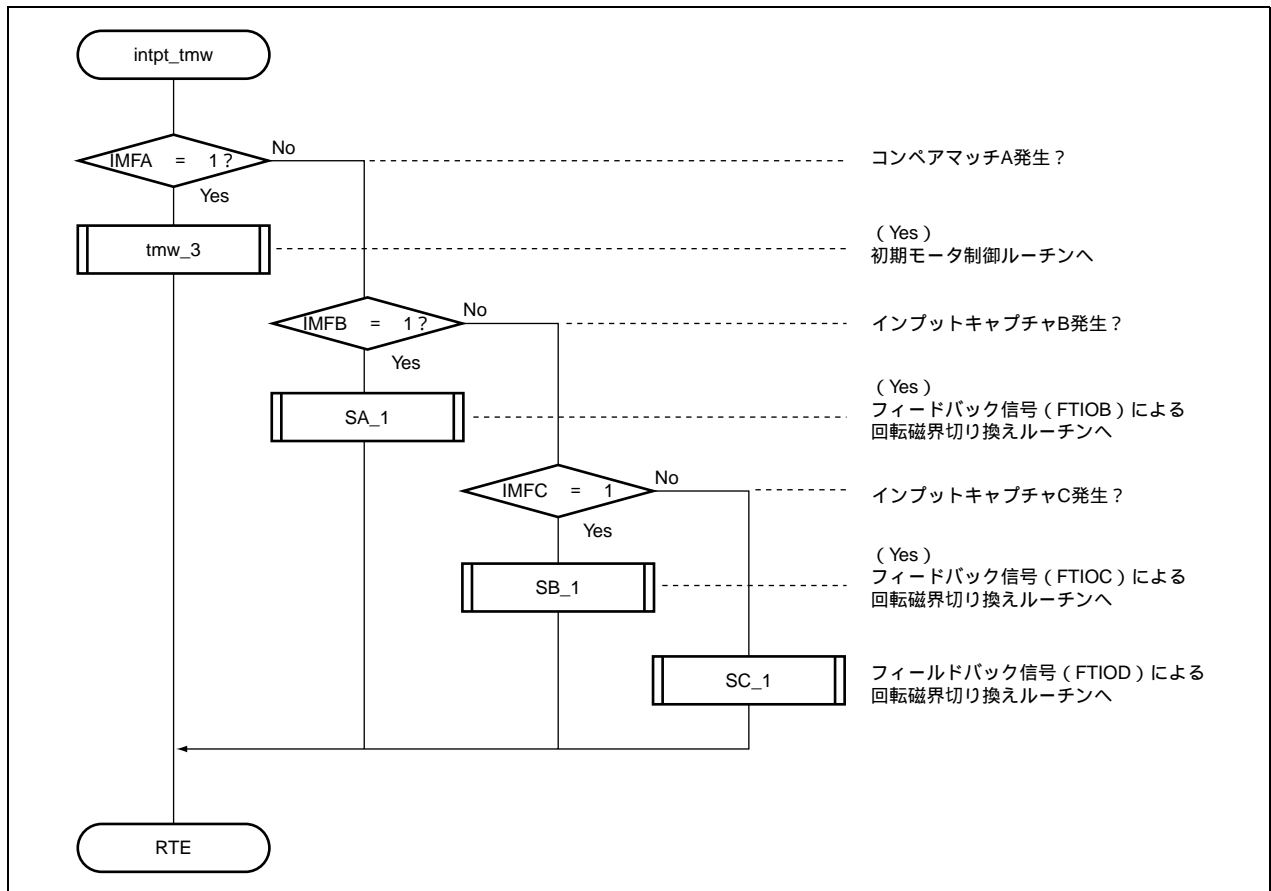
1. メインルーチン



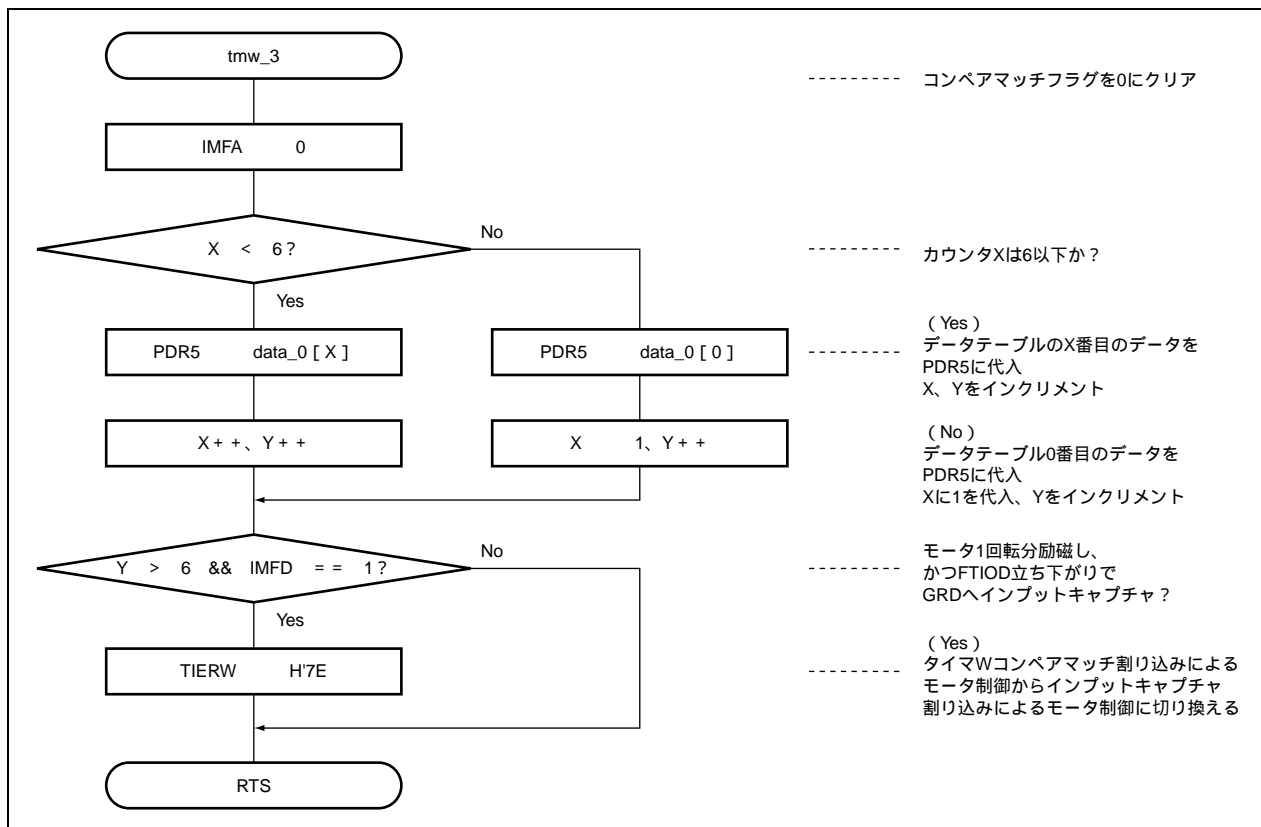
2. 使用レジスタイニシャライズルーチン



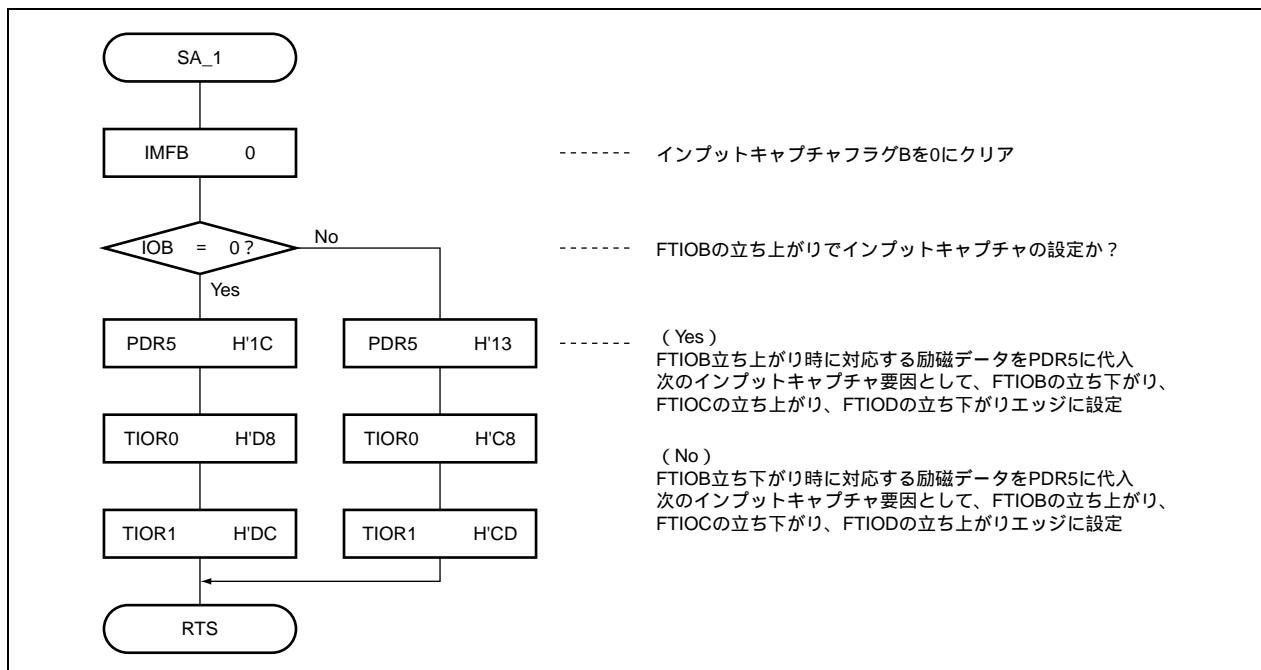
3. タイマ W 割り込み処理



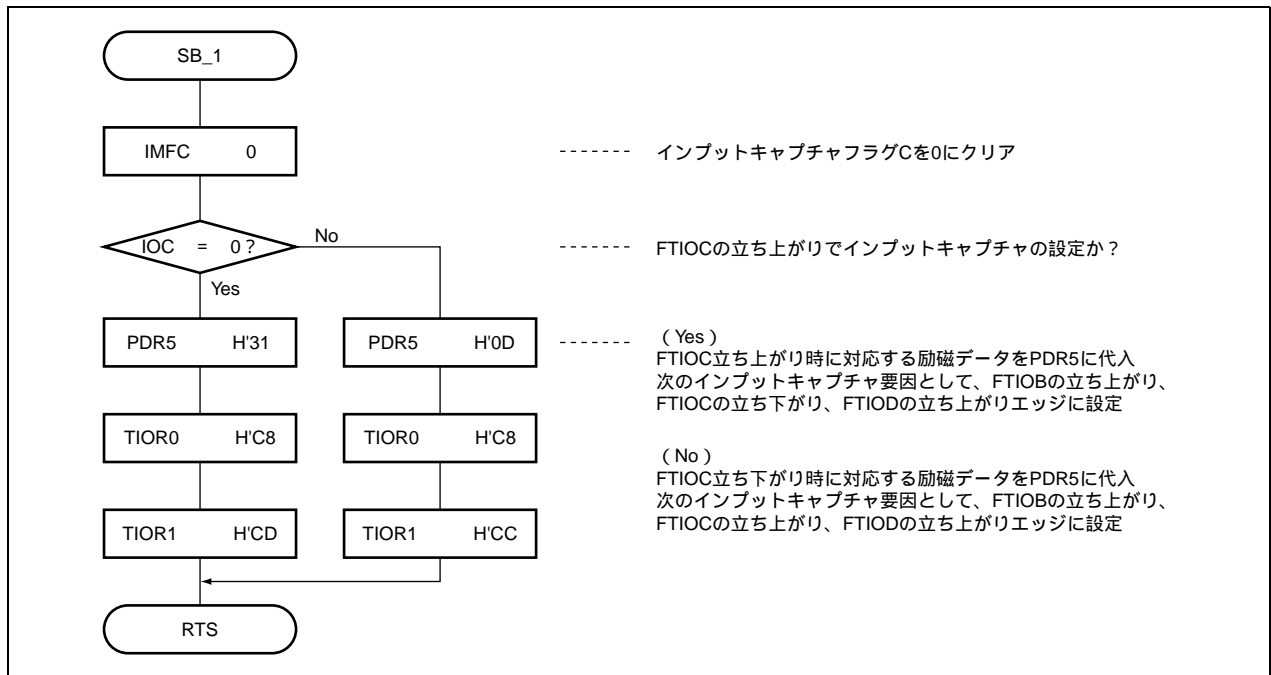
4. 初期モータ制御



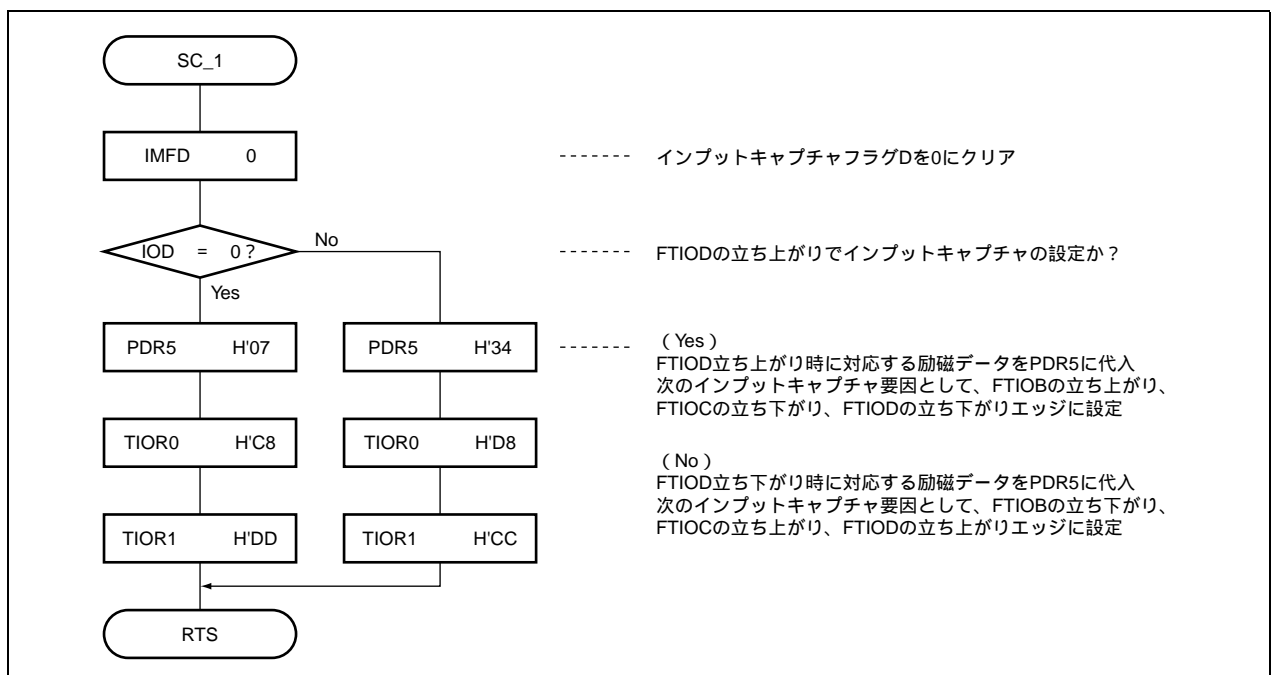
5. フィールドバック信号 (FTIOB) による回転磁界切り換えルーチン



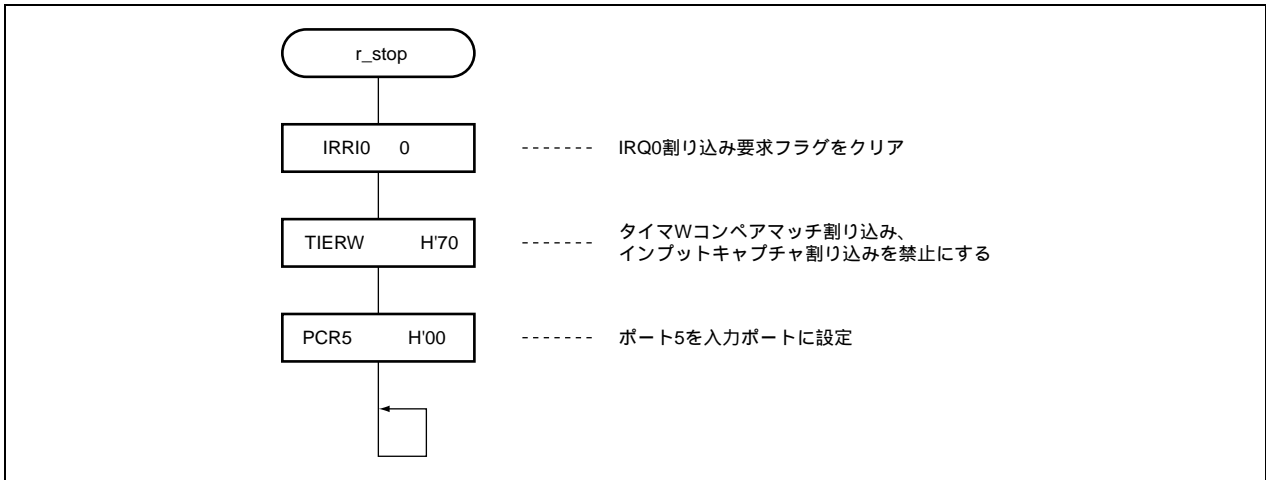
6. フィールドバック信号 (FTIOC) による回転磁界切り換えルーチン



7. フィールドバック信号 (FTIOD) による回転磁界切り換えルーチン



8. モータ回転停止信号検出



7. プログラムリスト

INIT.SRC (プログラムリスト)

```

        .EXPORT          _INIT
        .IMPORT          _main
;
        .SECTION        P, CODE
_INIT:
        MOV.W           #H'FF80,R7
        LDC.B           #B'10000000,CCR
        JMP             @_main
;
        .END

```

```

/* H8/300H TIny Series -H8/3664- Application Note */
/* 応用編-3 */
/* ~DC ブラシレスモータ制御例 */
#include <machine.h>

/* Symbol Definition */

struct BIT {
    unsigned char    b7:1;
    unsigned char    b6:1;
    unsigned char    b5:1;
    unsigned char    b4:1;
    unsigned char    b3:1;
    unsigned char    b2:1;
    unsigned char    b1:1;
    unsigned char    b0:1;
};

#define TMRW          *(volatile unsigned char *)0xFF80 /* Timer W */
#define TMRW_BIT      (*(struct BIT *)0xFF80)
#define CTS           TMRW_BIT.b7
#define TCRW          *(volatile unsigned char *)0xFF81
#define TIERW         *(volatile unsigned char *)0xFF82
#define TIERW_BIT     (*(struct BIT *)0xFF82)
#define IMIED         TIERW_BIT.b3
#define IMIEC         TIERW_BIT.b2
#define IMIEB         TIERW_BIT.b1
#define IMIEA         TIERW_BIT.b0

```

```

#define    TSRW        *(volatile unsigned char *)0xFF83
#define    TSRW_BIT    (*(struct BIT *)0xFF83)
#define    IMFD        TSRW_BIT.b3
#define    IMFC        TSRW_BIT.b2
#define    IMFB        TSRW_BIT.b1
#define    IMFA        TSRW_BIT.b0
#define    TIOR0       *(volatile unsigned char *)0xFF84
#define    TIOR0_BIT   (*(struct BIT *)0xFF84)
#define    IOB0        TIOR0_BIT.b4
#define    TIOR1       *(volatile unsigned char *)0xFF85
#define    TIOR1_BIT   (*(struct BIT *)0xFF85)
#define    IOD0        TIOR1_BIT.b4
#define    IOC0        TIOR1_BIT.b0
#define    TCNT        *(volatile unsigned int *)0xFF86
#define    GRA         *(volatile unsigned int *)0xFF88

#define    TCRV0       *(volatile unsigned char *)0xFFA0    /* Timer V */
#define    TCSR0       *(volatile unsigned char *)0xFFA1
#define    TCRV1       *(volatile unsigned char *)0xFFA2
#define    TCSR1       *(volatile unsigned char *)0xFFA3
#define    TCRV2       *(volatile unsigned char *)0xFFA4
#define    TCSR2       *(volatile unsigned char *)0xFFA5
#define    PUCR1       *(volatile unsigned char *)0xFFD0    /* I/O Port */
#define    PUCR5       *(volatile unsigned char *)0xFFD1
#define    PDR1        *(volatile unsigned char *)0xFFD4
#define    PDR2        *(volatile unsigned char *)0xFFD5
#define    PDR5        *(volatile unsigned char *)0xFFD8
#define    PDR7        *(volatile unsigned char *)0xFFDA
#define    PDR8        *(volatile unsigned char *)0xFFDB
#define    PMR1        *(volatile unsigned char *)0xFFE0
#define    PMR1_BIT    (*(struct BIT *)0xFFE0)
#define    IRQ0        PMR1_BIT.b4
#define    PMR5        *(volatile unsigned char *)0xFFE1
#define    PCR1        *(volatile unsigned char *)0xFFE4
#define    PCR2        *(volatile unsigned char *)0xFFE5
#define    PCR5        *(volatile unsigned char *)0xFFE8
#define    PCR7        *(volatile unsigned char *)0xFFEA
#define    PCR8        *(volatile unsigned char *)0xFFEB
#define    IEGR1       *(volatile unsigned char *)0xFFF2    /* IRQ0 */
#define    IENR1       *(volatile unsigned char *)0xFFF4
#define    IENR_BIT    (*(struct BIT *)0xFFF4)
#define    IEN0        IENR_BIT.b0
#define    IRR1        *(volatile unsigned char *)0xFFF6

```

```

#define      IRR1_BIT      (*(struct BIT *)0xFFF6)
#define      IRR10        IRR1_BIT.b0
#define      MSTCR1       *(volatile unsigned char *)0xFFF9

#pragma interrupt (intpt_tmw)                /* Timer W interrupt */
#pragma interrupt (r_stop)                  /* IRQ0 */
/* 関数定義 */
extern void INIT (void);
void main(void);
void IO_INIT(void);
void intpt_tmw(void);
void r_stop(void);
void tmw_3(void);
void SA_1(void);
void SB_1(void);
void SC_1(void);

/* Data Table */
const unsigned char data_0[6]=
{
    0x34,
    0x31,
    0x13,
    0x07,
    0x0d,
    0x1c
};

unsigned char x;          /* データテーブル参照用カウンタ */
unsigned char y;          /* タイマ機能切り換え用カウンタ */
/* Vector Address */
#pragma section V1
void (*const VEC_TBL1[])(void) = {
    INIT                    /* H'0000 Reset vector */
};
#pragma section V2
void (*const VEC_TBL2[])(void) = {
    intpt_tmw
};
#pragma section V3
void (*const VEC_TBL3[])(void) = {
    r_stop
};

```

```
};

#pragma section /* P */
/* main routine */

void main(void)
{
    set_imask_ccr(1); /* CCR -Ibit = 1 */
    IO_INIT(); /* initialize */
    TCNT=0x00;
    IMFA = 0;
    IMIEA = 1; /* enable IMIA */
    CTS = 1; /* Start Timer count */
    IRRIO= 0;
    IENO=1;
    set_imask_ccr(0); /* CCR -Ibit = 0 */
    while(1) {
        ;
    }
}

void IO_INIT(void)
{
    MSTCR1 = 0xf9;
    x=0; /* clear counter */
    y=0;

    PCR2 = 0x03;
    PDR2 = 0x03;
    PMR5 = 0x00;
    PUCR5 = 0x00;
    PDR5 = 0x00;
    PCR5 = 0xff; /* Output Port5 */

    PDR7 = 0x70;
    PCR7 = 0x70;

    PDR8 = 0xff;
    PCR8 = 0xf1;
```

```

TCRV0 = 0x09;          /* Timer V initialize          */
TCRV1 = 0x02;          /* clock =  $\phi/4$                 */
TCSR0 = 0x16;
TCORA = 0xc8;          /* PWM period = 50  $\mu$ s          */
TCORB = 0x64;          /* PWM duty = 50%              */
TCNTV = 0x00;          /* TCNTV start                  */

TMRW = 0x48;          /* Timer W initialize          */
TCRW = 0xb0;          /* Clock =  $\phi/8$                 */
TIERW = 0x70;
TSRW = 0x70;
TIOR0 = 0xc8;
TIOR1 = 0xd8;
GRA = 0x4e20;          /* Port output period = 10.0ms */

IEGR1 = 0x70;
}
void r_stop(void)      /* IRQ0 interrupt routine      */
{
    IRRIO = 0;
    TIERW = 0x70;
    PCR5 = 0x00;
}
while(1){
    ;
}
}
void intpt_tmw(void)   /* Timer W interrupt routine    */
{
    if(IMFA==1){        /* Compare match A ?          */
        tmw_3();
    }
    if(IMFB==1){        /* FTIOB Input capture ?      */
        SA_1();
    }
    if(IMFC==1){        /* FTIOC Input capture ?      */
        SB_1();
    }
    if(IMFD==1){        /* FTIOD Input capture ?      */
        SC_1();
    }
}
void tmw_3(void)       /* Compare match A interrupt routine */

```

```

{

IMFA =0;
if(x < 6){
    PDR5 = data_0[x];
    x++;
    y++;
}
else{
    PDR5 = data_0[0];          /* If x > 6 = clear x          */
    x=1;
    y++;
}

if(y > 0x06 && IMFB == 1 ){ /* lrotated & input capture A ?          */
    TIERW = 0x7e;           /* disable IMIA          */
}
}

void SA_1(void)                /* FTIOB input capture routine          */
{
    IMFB = 0;
    if(IOB0 == 0){
        PDR5 = 0x1c;
        TIOR0 = 0xd8;
        TIOR1 = 0xdc;
    }
    else{
        PDR5 = 0x13;
        TIOR0 = 0xc8;
        TIOR1 = 0xcd;
    }
}

void SB_1(void)                /*FTIOC input capture routine          */
{
    IMFC = 0;
    if(IOC0 == 0){
        PDR5 = 0x31;
        TIOR0 = 0xc8;
        TIOR1 = 0xcd;
    }
    else{

```

```
PDR5 = 0x0d;
TIOR0 = 0xc8;
TIOR1 = 0xcc;
}
}
void SC_1(void) /* FTIOD input capture routine */
{
    IMFD = 0;
    if(IOD0 == 0){
        PDR5 = 0x07;
        TIOR0 = 0xc8;
        TIOR1 = 0xdd;
    }
    else{
        PDR5 = 0x34;
        TIOR0 = 0xd8;
        TIOR1 = 0xcc;
    }
}
```

リンクアドレス指定

セクション名	アドレス
CV1	H'0000
CV2	H'002A
CV3	H'001C
P	H'0100
B	H'FB80

