

お客様各位

カタログ等資料中の旧社名の扱いについて

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

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

2010年4月1日
ルネサスエレクトロニクス株式会社

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

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

ご注意書き

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

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

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

V850マイクロコントローラによる モータ制御

ホールICによる120度通電方式制御編

V850ES/IE2

V850ES/IK1

V850E/IA3

V850E/IA4

V850E/IF3

V850E/IG3

〔メモ〕

目次要約

第1章 概 説 ... 13

第2章 BLDCモータ制御の原理 ... 14

第3章 システム概要 ... 19

第4章 制御プログラム ... 26

CMOSデバイスの一般的注意事項

入力端子の印加波形

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

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

未使用入力の処理

CMOSデバイスの未使用端子の入力レベルは固定してください。

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

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

静電気対策

MOSデバイス取り扱いの際は静電気防止を心がけてください。

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

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

初期化以前の状態

電源投入時、MOSデバイスの初期状態は不定です。

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

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

電源投入切断順序

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

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

電源OFF時における入力信号

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

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

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

非該当品 : μ PD70F3184, 70F3186, 70F3329, 70F3713, 70F3714, 70F3451, 70F3452, 70F3453,
70F3454

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

- 本資料に記載されている内容は2008年9月現在のもので、今後、予告なく変更することがあります。量産設計の際には最新の個別データ・シート等をご参照ください。
 - 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。当社は、本資料の誤りに関し、一切その責を負いません。
 - 当社は、本資料に記載された当社製品の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、一切その責を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
 - 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責を負いません。
 - 当社は、当社製品の品質、信頼性の向上に努めておりますが、当社製品の不具合が完全に発生しないことを保証するものではありません。また、当社製品は耐放射線設計については行っていません。当社製品をお客様の機器にご使用の際には、当社製品の不具合の結果として、生命、身体および財産に対する損害や社会的損害を生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計を行ってください。
 - 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定していただく「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。
標準水準：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット
特別水準：輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器
特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等
- 当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。意図されていない用途で当社製品の使用をお客様が希望する場合には、事前に当社販売窓口までお問い合わせください。

(注)

- (1) 本事項において使用されている「当社」とは、NECエレクトロニクス株式会社およびNECエレクトロニクス株式会社がその総株主の議決権の過半数を直接または間接に保有する会社をいう。
- (2) 本事項において使用されている「当社製品」とは、(1)において定義された当社の開発、製造製品をいう。

M8E0710J

はじめに

- 注 意** 1. このアプリケーション・ノートでは、 μ PD70F3714 (V850ES/IE2) を代表マイコンとして用いた場合で記述しています。なお、V850ES/IE2, V850ES/IK1 は動作確認済みです。
- V850E/IA3, V850E/IA4, V850E/IF3, V850E/IG3 を使用する場合は、このアプリケーション・ノートを参考にしてください。
2. このアプリケーション・ノートで使用するプログラムは、NEC エレクトロニクスのホーム・ページのサンプル・プログラムのページ (<http://www.necel.com/micro/ja/designsupports/sampleprogram/index.html>) よりダウンロードしてください。
3. サンプル・プログラムはあくまで参考用のものであり、当社がこの動作を保証するものではありません。サンプル・プログラムを使用する場合、ユーザのセット上で十分な評価をしたうえで使用してください。
4. サンプル・プログラムを使用する場合は、次のスタートアップ・ファイル、リンク・ディレクティブ・ファイルを参照し、必要に応じて修正してください。
- ・スタートアップ・ファイル : IE2crt.s
 - ・リンク・ディレクティブ・ファイル : IE2motor.s

対象者 このアプリケーション・ノートは、V850ES/IE2, V850ES/IK1, V850E/IA3, V850E/IA4, V850E/IF3, V850E/IG3の機能を理解し、それらを使用した応用システムを設計するユーザを対象とします。対象製品を次に示します。

- ・ V850ES/IE2
 μ PD70F3713, 70F3714
- ・ V850ES/IK1
 μ PD703327, 703329, 70F3329
- ・ V850E/IA3
 μ PD703183, 70F3184
- ・ V850E/IA4
 μ PD703185, 703186, 70F3186
- ・ V850E/IF3
 μ PD70F3451, 70F3452
- ・ V850E/IG3
 μ PD70F3453, 70F3454

目的 このアプリケーション・ノートでは、 μ PD70F3714 (V850ES/IE2) を利用し、ホールIC付きブラシレスDCモータ (Brush Less DC Motor) を120度通電方式で駆動するシステムについてユーザに理解していただくことを目的としています。

構 成 このアプリケーション・ノートは大きく分けて次の内容で構成しています。

- ・概 説
- ・システム概要
- ・BLDCモータ制御の原理
- ・制御プログラム

読 み 方 このマニュアルの読者には、電気、論理回路、およびマイクロコントローラに関する一般知識を必要とします。

ハードウェア機能の詳細（特にレジスタ機能とその設定方法など）、および電気的特性を知りたいとき別冊のV850ES/IE2 **ユーザーズ・マニュアル ハードウェア編**、V850ES/IK1 **ユーザーズ・マニュアル ハードウェア編**、V850E/IA3、V850E/IA4 **ユーザーズ・マニュアル ハードウェア編**、V850E/IF3、V850E/IG3 **ユーザーズ・マニュアル ハードウェア編**を参照してください。

命令機能の詳細を理解しようとするとき

別冊のV850E1 **ユーザーズ・マニュアル アーキテクチャ編**、V850ES **ユーザーズ・マニュアル アーキテクチャ編**を参照してください。

凡 例 データ表記の重み：左が上位桁，右が下位桁
アクティブ・ロウの表記：xxx（端子，信号名称に上線）
メモリ・マップのアドレス：上部 - 上位，下部 - 下位
注：本文中に付けた注の説明
注意：気を付けて読んでいただきたい内容
備考：本文の補足説明
数の表記：2進数 ... xxxxまたはxxxxB
10進数 ... xxxx
16進数 ... xxxxH
2のべき数を示す接頭語（アドレス空間，メモリ容量）：
K（キロ）... $2^{10} = 1024$
M（メガ）... $2^{20} = 1024^2$
G（ギガ）... $2^{30} = 1024^3$
データ・タイプ：ワード ... 32ビット
ハーフワード ... 16ビット
バイト ... 8ビット

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

V850ES/IE2, V850ES/IK1, V850E/IA3, V850E/IA4, V850E/IF3, V850E/IG3に関する資料

(1/2)

資料名	資料番号
V850ES ユーザーズ・マニュアル アーキテクチャ編	U15943J
V850ES/IK1 ユーザーズ・マニュアル ハードウェア編	U16910J
V850ES/IE2 ユーザーズ・マニュアル ハードウェア編	U17716J
V850E1 ユーザーズ・マニュアル アーキテクチャ編	U14559J
V850E/IA3, V850E/IA4 ユーザーズ・マニュアル ハードウェア編	U16543J
V850E/IF3, V850E/IG3 ユーザーズ・マニュアル ハードウェア編	U18279J
V850シリーズによるインバータ制御 アプリケーション・ノート ホール・センサによるベクトル制御編	U17338J
V850シリーズによるインバータ制御 アプリケーション・ノート エンコーダによるベクトル制御編	U17324J
V850シリーズによるインバータ制御 アプリケーション・ノート ゼロクロス検出による120度通電方式制御編	U17209J
V850マイクロコントローラによるインバータ制御 アプリケーション・ノート タイマQ, タイマQオプション, タイマP, A/Dコンバータ0, 1による6相PWM出力制御編	U18600J
V850E/IF3, V850E/IG3 アプリケーション・ノート タイマAB, タイマQオプション, タイマAA, A/Dコンバータ0, 1による6相PWM出力制御編	U18717J
V850マイクロコントローラによるモータ制御 アプリケーション・ノート ホールICによる120度通電方式制御編	このマニュアル
V850マイクロコントローラによるモータ制御 アプリケーション・ノート ホールICによる180度通電方式制御編	U19422J
V850マイクロコントローラによるモータ制御 アプリケーション・ノート センサレス (BEMFのA/D変換)による120度通電方式制御編	U19423J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート シリアル通信 (UARTA) 編	U18233J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート シリアル通信 (CSIB) 編	U18234J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート DMA機能編	U18235J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマM編	U18236J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート ウォッチドッグ・タイマ編	U18237J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマP編	U18238J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマQ編	U18239J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート タイマENC編	U18240J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート ポート機能編	U18241J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート クロック・ジェネレータ編	U18242J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート スタンバイ機能編	U18243J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート 割り込み機能編	U18244J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート A/Dコンバータ0, 1編	U18245J
サンプル・プログラム関数使用法説明書 アプリケーション・ノート A/Dコンバータ2編	U18246J

資料名	資料番号
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム シリアル通信 (UARTA) 編	U18723J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム シリアル通信 (UARTB) 編	U18724J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム シリアル通信 (CSIB) 編	U18725J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム シリアル通信 (I ² C) 編	U18726J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム DMA機能編	U18727J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム タイマM編	U18728J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム ウォッチドッグ・タイマ編	U18729J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム タイマAA編	U18730J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム タイマAB編	U18731J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム タイマT編	U18732J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム ポート機能編	U18733J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム クロック・ジェネレータ編	U18734J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム スタンバイ機能編	U18735J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム 割り込み機能編	U18736J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム A/Dコンバータ0, 1編	U18737J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム A/Dコンバータ2編	U18738J
V850E/IF3, V850E/IG3 アプリケーション・ノート サンプル・プログラム 低電圧検出回路 (LVI) 機能編	U18739J

開発ツールに関する資料(ユーザーズ・マニュアル)

資料名		資料番号
QB-V850ESIX2 (V850ES/IE2用インサーキット・エミュレータ)		U17909J
QB-V850EIA4 (V850E/IA3, V850E/IA4, V850ES/IK1用インサーキット・エミュレータ)		U17167J
QB-V850EIX3 インサーキット・エミュレータ		U18651J
QB-V850MINI (V850E/IA4, V850E/IF3, V850E/IG3用オンチップ・デバッグ・エミュレータ)		U17638J
QB-MINI2 プログラミング機能付きオンチップ・デバッグ・エミュレータ		U18371J
CA850 Ver.3.20 Cコンパイラ・パッケージ	操作編	U18512J
	C言語編	U18513J
	アセンブリ言語編	U18514J
	リンク・ディレクティブ編	U18515J
PM+ Ver.6.30 プロジェクト・マネージャ		U18416J
ID850QB Ver.3.40 統合デバッガ	操作編	U18604J
TW850 Ver.2.00 性能解析チューニング・ツール		U17241J
SM+ システム・シミュレータ	操作編	U18601J
	ユーザ・オープン・インタフェース編	U18212J
RX850 Ver.3.20 リアルタイムOS	基礎編	U13430J
	インストール編	U17419J
	テクニカル編	U13431J
	タスク・デバッガ編	U17420J
RX850 Pro Ver.3.21 リアルタイムOS	基礎編	U18165J
	インストール編	U17421J
	内部構造編	U18164J
	タスク・デバッガ編	U17422J
RX-NET TCP/IPライブラリ		U15083J
AZ850 Ver.3.30 システム・パフォーマンス・アナライザ		U17423J
PG-FP4 フラッシュ・メモリ・プログラマ		U15260J
PG-FP5 フラッシュ・メモリ・プログラマ		U18865J

目 次

第1章 概 説 ...	13
1.1 動作環境 ...	13
1.2 関連マニュアル ...	13
第2章 BLDCモータ制御の原理 ...	14
2.1 回転方向の定義 ...	14
2.2 回転原理 ...	14
2.3 通電パターン ...	15
2.4 インバータ ...	16
2.5 120度通電方式 ...	16
2.6 位置情報 ...	17
2.7 通電パターン切り替え ...	17
2.8 制 御 ...	18
2.8.1 電圧制御 ...	18
2.8.2 速度制御 ...	18
2.8.3 PID制御 ...	18
2.8.4 電流マイナ・ループ制御 ...	18
第3章 システム概要 ...	19
3.1 構 成 ...	19
3.2 インタフェース ...	21
3.3 機 能 ...	23
3.4 周辺I/O ...	24
3.5 割り込み ...	25
第4章 制御プログラム ...	26
4.1 通電パターン ...	26
4.2 速度検出 ...	27
4.3 電圧制御 ...	27
4.4 速度制御 ...	27
4.4.1 PID制御 ...	28
4.5 電流マイナ・ループ制御 ...	28
4.6 ノイズ・フィルタ処理 ...	29
4.7 モジュール構成 ...	29
4.7.1 システム状態遷移 ...	29
4.7.2 ライブラリ ...	30
4.8 関数一覧 ...	30
4.8.1 外部インタフェース用関数一覧 ...	30
4.8.2 内部関数一覧 ...	32
4.9 フロー・チャート ...	34

4.9.1	外部インタフェース用関数のフロー・チャート	...	34
4.9.2	内部用関数のフロー・チャート	...	67
4.10	定数一覧	...	96
4.10.1	外部インタフェース用定数（設定可能定数）	...	96
4.10.2	内部定数	...	98
4.11	変数一覧	...	102
4.11.1	外部インタフェース用変数	...	102
4.11.2	内部変数	...	102
4.12	ソース・プログラム	...	105

第1章 概 説

このシステムはホールIC付きブラシレスDCモータ（Brush Less DC Motor：以降BLDCモータ）を120度通電方式で駆動します。

- ・このシステム（サンプル・プログラム）は NEC エレクトロニクス のモータ・スタータ・キット（ μ PD70F3714（V850ES/IE2））を利用し、ホール IC 付き BLDC モータを 120 度通電方式で駆動します。
- ・制御ゲインは動作環境の特定のモータに合わせて調整しています。

1.1 動作環境

このシステムは以下の環境で使用することを前提に作成しています。それ以外の組み合わせでの動作は、保証できないので注意してください。

- ・モータ・スタータ・キット（ μ PD70F3714（V850ES/IE2））ボード一式
- ・低電圧インバータ・セット
BLDCモータ AMETEK
駆動コイル：3相（Y結線）
磁極ロータ：4極（2極対）
ステータ：6スロット
位置センサ：ホールIC
- ・PM+ 環境プラットフォームV6.30
- ・CA850コンパイラV3.20
- ・DF70F3714デバイス・ファイルV1.01

1.2 関連マニュアル

開発環境およびボードについては次に示すマニュアルを参照してください。

- ・低電圧モータ・スタータ・キット マニュアル
- ・PM+ Ver.6.30 ユーザーズ・マニュアル
- ・CA850 Ver.3.20 Cコンパイラ 各ユーザーズ・マニュアル

第2章 BLDCモータ制御の原理

BLDCモータは固定子部分（ステータ）のコイルが発生する磁界の作用により、永久磁石でできた回転部分（ロータ）が回転します。

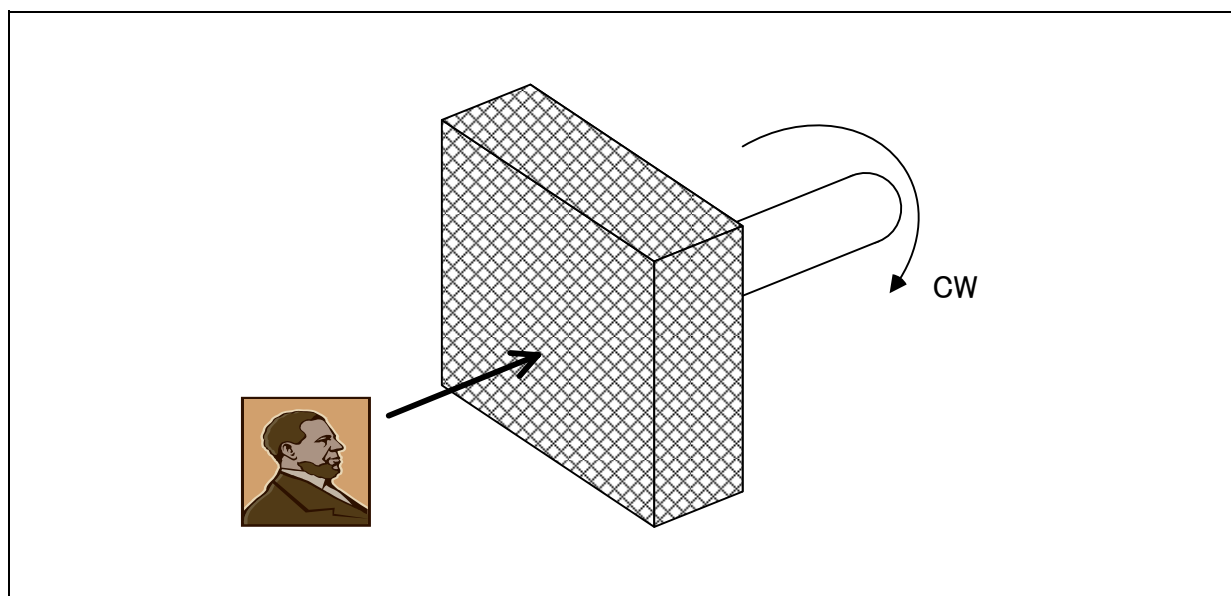
2.1 回転方向の定義

まず、モータの回転方向についての定義をします。

モータの回転方向は、CW（時計回り）/CCW（反時計回り）があります。

モータが回す対象物の回転方向を基準にしてCW/CCWが決定します。モータの軸がある面を対象物側に向けたときの回転方向を基準にします。したがって、次のようになります。

図2 - 1 モータの回転方向



2.2 回転原理

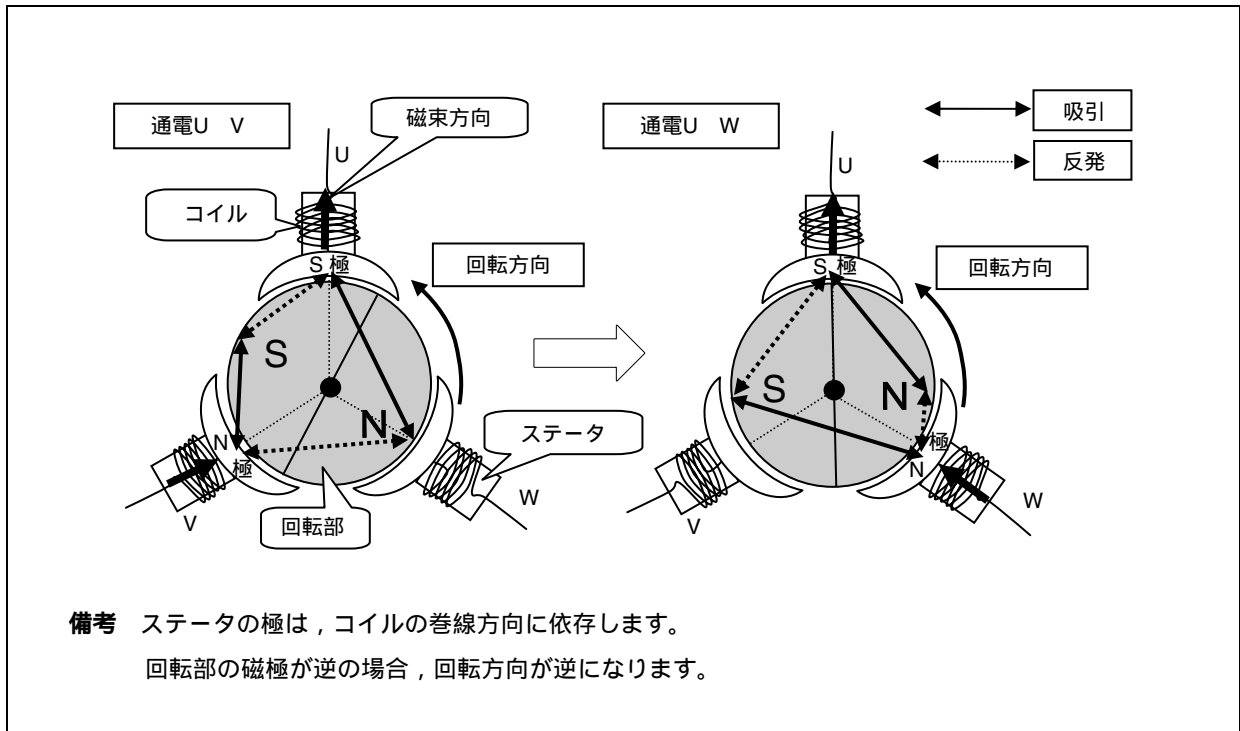
BLDCモータの回転原理を示します。

次に示したBLDCモータは3相2極3スロットY結線でインナ・ロータ型のSPM（Surface Permanent Magnet：永久磁石を表面に配置した表面磁石構造）です。

ステータの極と回転部永久磁石の磁極との吸引および反発によって発生するマグネット・トルクで回転します（リラクタンス・トルクは発生しません）。

備考	3相	: U相, V相, W相
	2極	: N極とS極
	3スロット	: ステータ数が3個
	Y結線	: U相, V相, W相を中性点で一括して接続する結線

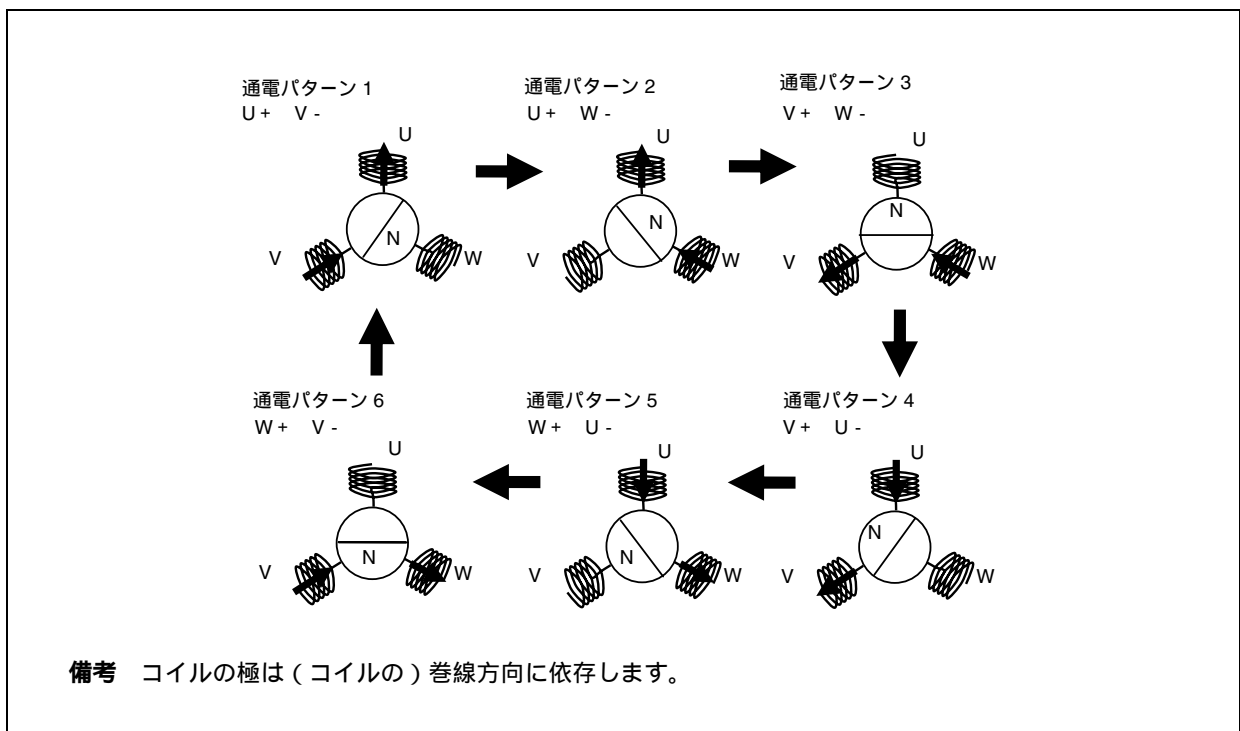
図2 - 2 BLDCモータの回転原理



2.3 通電パターン

次に通電パターンとコイルで発生する電流磁束方向（ステータの極）と回転部の磁極の関係を示します。

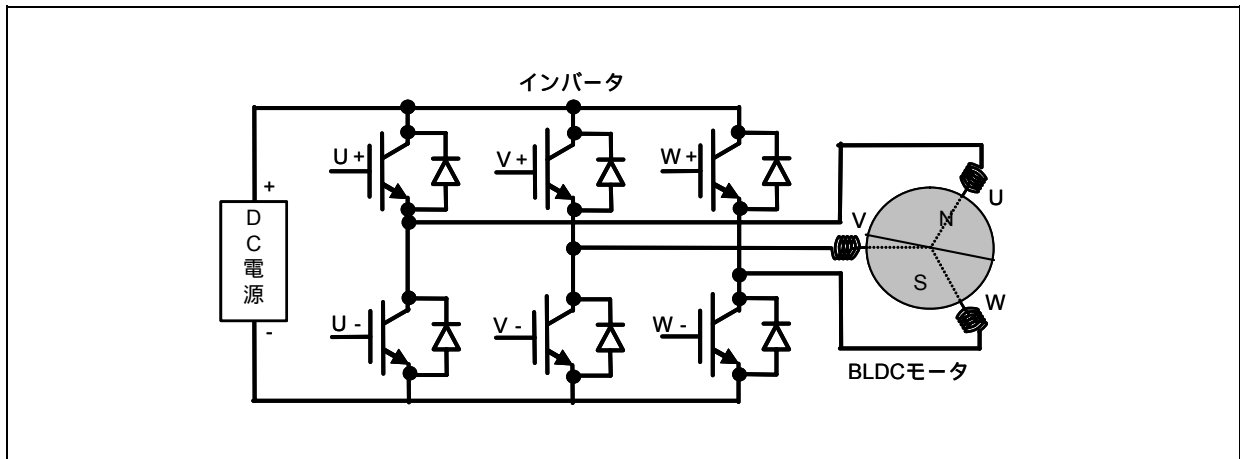
図2 - 3 通電パターンとコイルの磁束方向



2.4 インバータ

ブラシと整流子のないBLDCモータは、コイルと電流の向きをインバータで切り替えます。
次にインバータと3相Y結線BLDCモータとの結線図を示します。

図2-4 インバータと3相Y結線BLDCモータとの結線

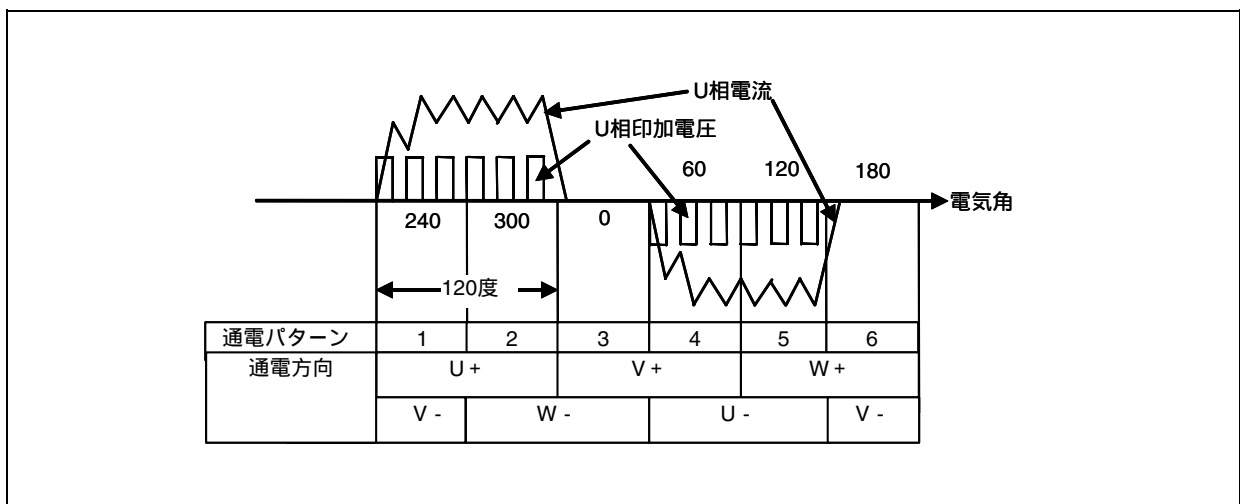


6個のスイッチング素子で通電時間や通電方向を制御します。

2.5 120度通電方式

次の図は2.3 通電パターンで示した通電パターンで各相に矩形波の電圧を印加したときのU相の状態（電圧，電流）を示したものです。U相には120度の通電期間と60度の無通電期間が繰り返し発生します。
各相の通電期間を120度で行うこの方式を120度通電方式と言います。

図2-5 120度通電と電流



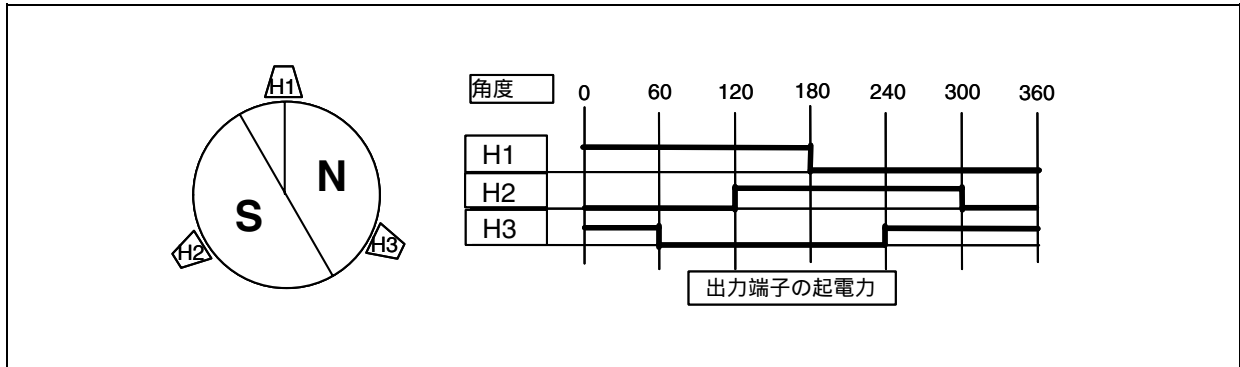
このシステムは、BLDCモータを120度通電方式で駆動します。

2.6 位置情報

120度通電方式ではインバータの切り換えを行うための位置情報が必要です。

次にホールICを利用した磁極位置センサの出力例を示します。

図2-6 ホールIC出力



180度ごとに变化するホールICを、120度ごとに3個配置することで、60度ごとの位置検出が可能です。

ホールICの値と通電パターン(2.3 通電パターン参照)と位置情報(図2-6 ホールIC出力参照)を組み合わせる例を次に示します。

表2-1 ホールICと通電パターンの関係

ホールIC \ 通電パターン	1	2	3	4	5	6
H1	L	H	H	H	L	L
H2	L	L	L	H	H	H
H3	H	H	L	L	L	H

備考 H: ハイ・レベル出力

L: ロウ・レベル出力

回転部の磁極が4極の場合、1つのホールICは90度ごとに値が変化し、通電パターンの切り替えは30度ごとに行います。

通常、1サイクル(6つの通電パターン)を電気角の360度、モータ軸の1回転を機械角の360度と定義します(このマニュアルではすべて電気角で記述しています)。

$$\text{機械角} = \text{電気角} / \text{極対数}$$

$$\text{極対数} = \text{極数} / 2$$

2.7 通電パターン切り替え

ホールICによる回転部の磁極位置情報から回転方向を考慮して、通電パターンを設定します。

2.8 制 御

2.8.1 電圧制御

モータのコイルに加える電圧は、120度通电の矩形波をスイッチング素子のいずれかの導通期間を高い周波数でチョッパ動作させて通流率（平均電圧）を調整するPWM（Pulse Width Modulation）で制御します。

2.8.2 速度制御

モータのコイルに加える電圧（PWMの通流率）をPID制御で変更することで速度を制御します。

2.8.3 PID制御

指定された速度と検出した回転速度の偏差でPID制御を行い、PWM電圧制御法の通流率（以降、デューティ比と記述）を変更します。

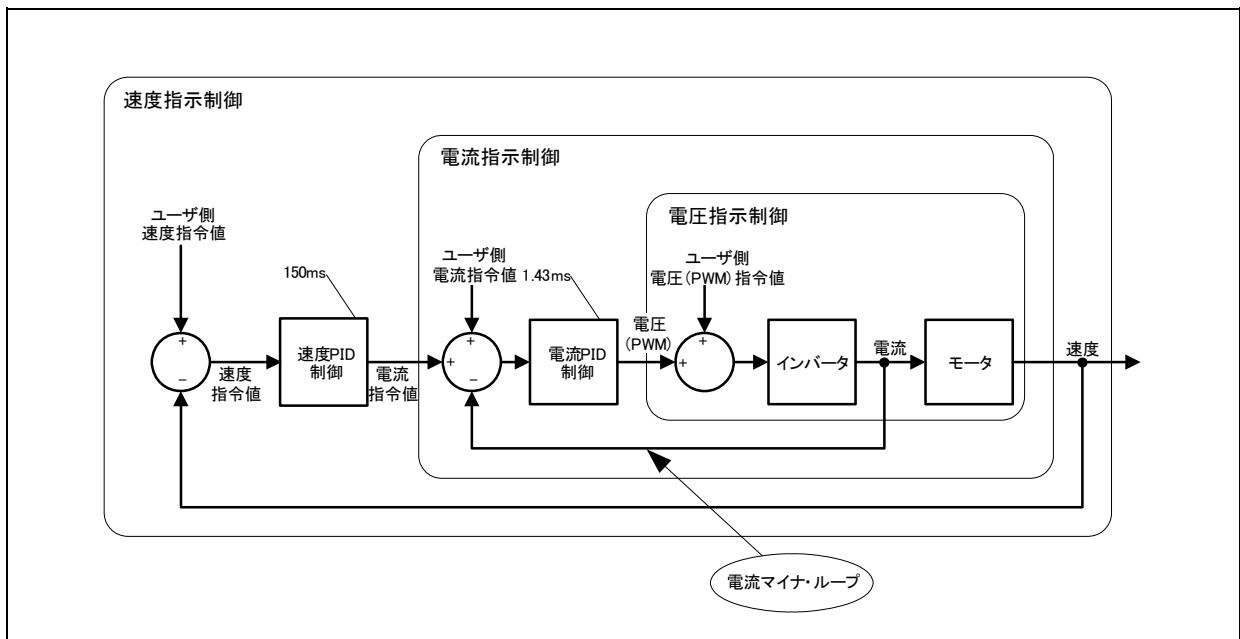
PID制御は、偏差に比例する出力を出す比例動作（Proportional action：P動作）、偏差の積分に比例する出力を出す積分動作（Integral action：I動作）、偏差の微分に比例する出力を出す微分動作（Derivative action：D動作）からPWMのデューティ比操作量を求めます。

2.8.4 電流マイナ・ループ制御

速度制御によるクローズド・ループの内側に、速度制御の制御間隔よりも短い間隔で微調整を行う制御をマイナ・ループ制御と言います。このマイナ・ループ制御の中で、電流を対象としたものを電流マイナ・ループと言います。

電流マイナ・ループでは、基準となる電流と検出した電流の偏差でPID制御を行い、PWMのデューティ比を変化させます。速度制御と同様にPIDパラメータを別に用意します。

図2-7 制御ループ



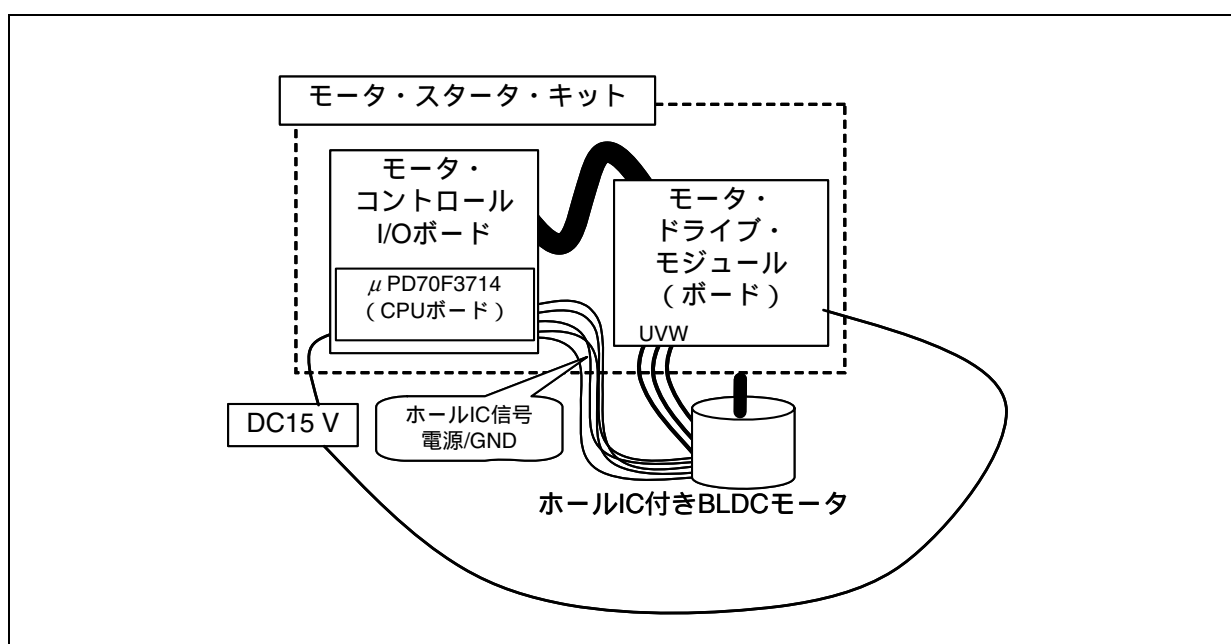
第3章 システム概要

このシステムの概要について示します。

3.1 構成

このシステムの構成を次に示します。

図3-1 システム構成

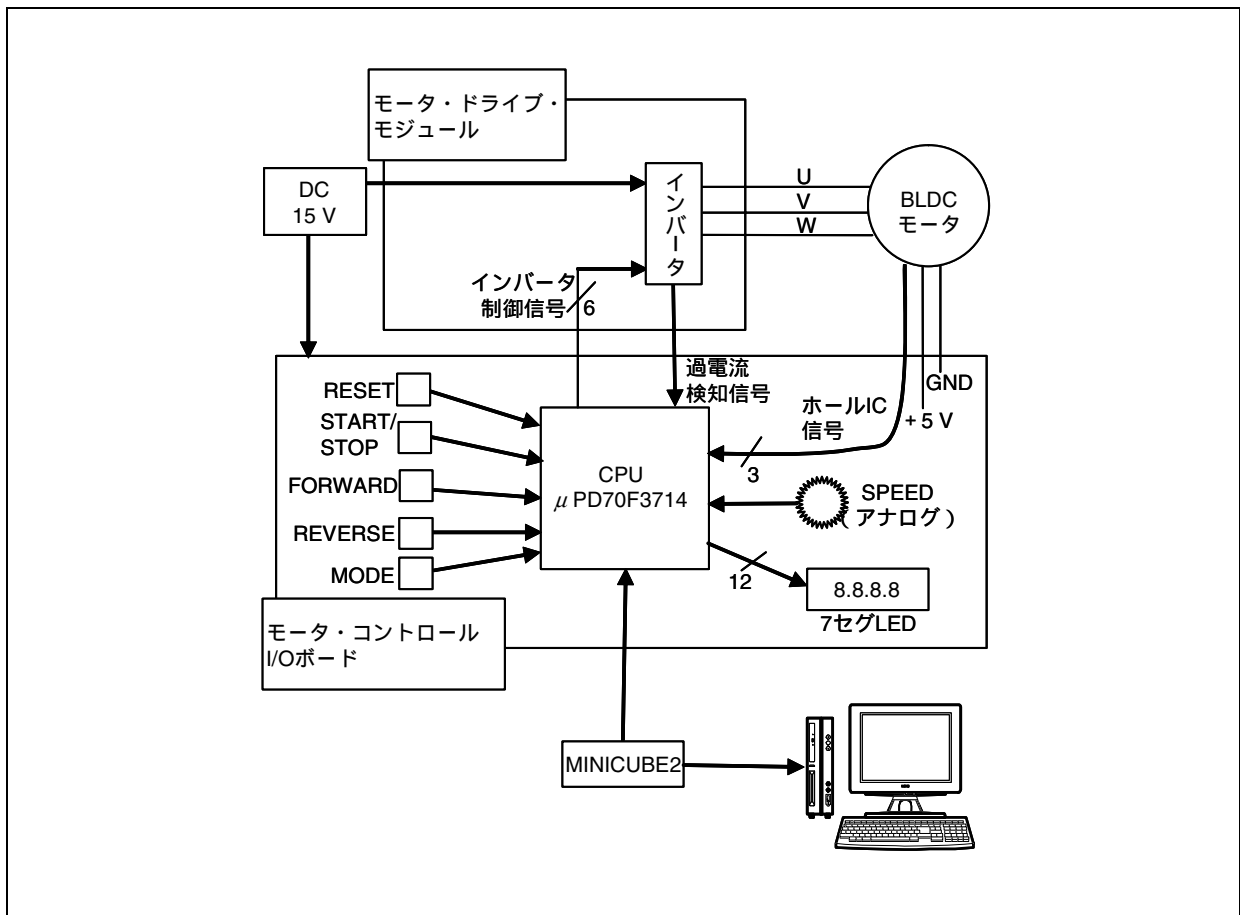


モータを駆動するモータ・ドライブ・モジュール，モータ制御用スイッチを搭載したモータ・コントロール I/O ボード，CPUを搭載した μ PD70F3714で構成されます。

BLDCモータは3相4極（2極対）のホールIC付きモータです。

モータ・スタータ・キットのブロック構成を次に示します。

図3-2 ブロック構成



モータ・コントロールI/Oボード上のスイッチでモータの制御を行います。

3.2 インタフェース

表3 - 1にユーザ・インタフェース一覧を示します。

表3 - 1 ユーザ・インタフェース

機能名	部品番号	機能
RESET	SW1	リセット
START/STOP	SW2	スタート/ストップ
FORWARD	SW3	回転方向（右回転，時計回り，CW）
REVERSE	SW4	回転方向（左回転，反時計回り，CCW）
MODE	SW5	速度指定切り替え 速度表示切り替え
SPEED	R52	指定速度変更
7セグLED	DISP1 DISP2 DISP3 DISP4	速度の表示（rpm） ^注

注 モータ停止中は常に指定速度を表示します。
 指定速度固定時には右下に“.”を表示します。
 モータ回転中は回転速度を表示します。
 モータ回転中はMODEスイッチを押している間，指定速度を表示します。
 RESETを押している間はリセット継続，離すとリセット解除です。
 リセット解除直後“SELF”と2秒表示します。

表3 - 2にエラー一覧を示します。

表3 - 2 エラー一覧

エラー	LEDの表示	状況
ハードウェア過電流	O.C.	モータの電流が異常
ソフトウェア過電流	S.O.C.	
ホールIC	HALL	ホールICの値が異常
装置異常	FAIL	モータが回転していない

表3 - 3に μ PD70F3714端子のインタフェース一覧を示します。

表3 - 3 端子のインタフェース

端子番号	端子名	機能
5	RESET	RESET (SW1)
49-54	TOQ1T1-TOQ1T13, TOQ1B1- TOQ1B3	3相PWMインバータ切り替え
15	P02/INTP2/TOP2OFF	ホールIC信号 (HALL1)
14	P03/INTP3/TOP3OFF	ホールIC信号 (HALL2)
13	P04/INTP4/ADTRG0	ホールIC信号 (HALL3)
37-40	P40-P43	7セグLED選択 (LD_LED0-LD_LED3)
12	P05/INTP5/ADTRG1	START/STOP (SW2)
18	P17/TOP21/TIP21	FORWARD (SW3)
19	P16/TOQ00/TIP20	REVERSE (SW4)
11	P06/INTP6	MODE (SW5)
28-35	PDL0-PDL7	7セグLEDへの出力データ
16	P01/INTP1/TOQ1OFF	過電流検知 (+5V 0V) (ITRIP)
58	ANI10	速度変更 (R52)
4	ANI03	ISHUNT電流
22	P12/TOQH02/TIQ03/TOQ03	タイマ・キャプチャ・トリガ
20	P14/TOQH03/EVTQ0	モータ・ドライブ・モジュール出力制御

3.3 機能

次にこのシステムの機能と動作概要を示します。

表3-4 システムの機能と動作概要 (1/2)

機能	概要
起動 (電源供給)	<ul style="list-style-type: none"> ・LEDに “ SELF ” を2秒表示する ・SPEEDボリュームの指定速度 (rpm) をLEDに表示する
RESETスイッチ	<ul style="list-style-type: none"> ・モータの制御状態に関係なく、システムの再起動を行う
START/STOP スイッチ	モータ制御停止中：モータ制御を開始する
	<ul style="list-style-type: none"> ・LEDに “ 0 ” を表示する ・モータはCWで回転を開始する ・モータの回転速度 (rpm) をLEDに表示する
	モータ制御中：モータ制御を停止する
	<ul style="list-style-type: none"> ・モータの回転速度 (rpm) が0になるまでLEDに表示する ・SPEEDボリュームの指定速度 (rpm) をLEDに表示する
	押し続けてもSTARTとSTOPはトグルしない
FORWARDスイッチ	モータ制御停止中：機能しない
	<ul style="list-style-type: none"> ・変化なし
	モータ制御中：回転方向を変える
	<ul style="list-style-type: none"> ・回転方向がCCWの場合、一度停止してからCWになる ・回転方向がCWの場合、変化しない
REVERSEスイッチ	モータ制御停止中：機能しない
	<ul style="list-style-type: none"> ・変化なし
	モータ制御中：回転方向を変える
	<ul style="list-style-type: none"> ・回転方向がCCWの場合、変化しない ・回転方向がCWの場合、一度停止してからCCWになる
MODEスイッチ	モータ制御停止中：速度指定を切り替える
	<ul style="list-style-type: none"> ・SPEEDボリュームによる指定速度変更を無効 / 有効にする ・無効時はLEDの数値右下に “ . ” が表示される
	モータ制御中：LEDの表示を変える
	<ul style="list-style-type: none"> ・押ししている間、SPEEDボリュームの指定速度 (rpm) をLEDに表示する
SPEEDボリューム	モータ制御停止中：指定速度の変更
	<ul style="list-style-type: none"> ・LEDの表示速度 (rpm) が300 ~ 3000に変化する ・MODEスイッチで無効に設定されている場合は変化しない
	モータ制御中：指定速度の変更
	<ul style="list-style-type: none"> ・モータの回転速度 (平均) が指定した速度になる
ハードウェア過電流	<ul style="list-style-type: none"> ・モータ・ドライブ・モジュールの許容電流超過で発生する ・モータ制御を停止する ・LEDに “ 0.C. ” を表示する ・RESETスイッチ以外の機能を無効にする
ソフトウェア過電流	<ul style="list-style-type: none"> ・モータの許容電流超過で発生する ・モータ制御を停止する ・LEDに “ S.0.C. ” を表示する ・RESETスイッチ以外の機能を無効にする

注意 複数のスイッチが押された場合の動作は保証していません。

表3 - 4 システムの機能と動作概要 (2/2)

機 能	概 要
無回転	<ul style="list-style-type: none"> ・ STARTから約1.5秒以上モータが回転しない場合に発生する ・ モータ制御を停止する ・ LEDに “ FAIL ” を表示する ・ RESETスイッチ以外の機能を無効にする
ホールIC異常	<ul style="list-style-type: none"> ・ ホールICからの値が不正の場合発生する ・ モータ制御を停止する ・ LEDに “ HALL ” を表示する ・ RESETスイッチ以外の機能を無効にする

注意 複数のスイッチが押された場合の動作は保証していません。

3.4 周辺I/O

このシステムでは次のような周辺I/Oを使用しています。

表3 - 5 使用周辺I/O一覧

機 能	周辺I/O機能名 (μ PD70F3714)
インバータ・タイマ	<ul style="list-style-type: none"> ・ PWM出力用 ・ 16ビット・インバータ制御用タイマ (TMQ1など) ・ キャリア (変調) 周波数は20 kHz対称三角波 ・ キャリア (変調) 同期割り込みが50 μs間隔で発生する (メインのPID制御でデューティ比が計算され, キャリア同期割り込みで更新される)
ISHUNT電流値取得用タイマ	<ul style="list-style-type: none"> ・ 電流制御用16ビット・インバータ制御用タイマ (TMQ1) ・ キャリア (変調) 同期 (山) 割り込み = 50 μs間隔発生 ・ 割り込み関数内で1.5 ms間隔ISHUNT電流測定関数を実施する。
ISHUNT電流値取得	モータ動作電流を電圧変換している端子 (ANI03) から電圧値を取得
キャプチャ割り込み	<ul style="list-style-type: none"> ・ 速度計測用 ・ 16ビット入力パルス間隔測定機能 (TIQ03)
ハードウェア過電流割り込み	<ul style="list-style-type: none"> ・ 割り込み機能 (INTP1) ・ モータ・ドライブ・モジュールの過電流発生 (ロウ・レベル・アクティブ)
フェイル・セーフ	・ ウォッチドッグ・タイマ

3.5 割り込み

このシステムで使用する割り込みの一覧を次に示します。

表3 - 6 使用割り込み一覧

名 称	機 能	発生条件
INTP1	過電流発生の検知	外部端子
INTTQ1OV	キャリア同期割り込み発生	インバータ・タイマ・カウンタの アンダフロー
INTTQ1CC0	ISHUNT電流の取得	16ビット・タイマ・カウンタのオーバフロー
INTTQ0CC3	HALL1の変化検知（速度計測）	外部端子
RESET	リセット発生	RESET端子
INTWDT	内部リセット発生	プログラム暴走による ウォッチドッグ・タイマ・オーバフロー

第4章 制御プログラム

このシステムでは基本的な制御プログラムを用いて実際にモータの速度制御を行います。

4.1 通電パターン

BLDCモータのスペックから端子情報と通電パターンとホールICの値を記述します。

表4 - 1 BLDCモータの端子スペック

カラー	機能	備考
RED	MOTOR ϕ A	U相
WHITE	MOTOR ϕ B	V相
BLACK	MOTOR ϕ C	W相
BLUE	SENSOR ϕ 1	HALL1
GREEN	SENSOR ϕ 2	HALL2
YELLOW	SENSOR ϕ 3	HALL3
PURPLE	SENSOR Vcc	-
ORANGE	SENSOR GND	-

表4 - 2 通電パターンとホールICの値

通電 (正回転)	+A	+A	+B	+B	+C	+C
	-B	-C	-C	-A	-A	-B
SENSOR ϕ 1	L	H	H	H	L	L
SENSOR ϕ 2	H	H	L	L	L	H
SENSOR ϕ 3	L	L	L	H	H	H
通電 (逆回転)	+B	+C	+C	+A	+A	+B
	-A	-A	-B	-B	-C	-C

備考 L : ロウ・レベル出力

H : ハイ・レベル出力

A : U相

B : V相

C : W相

4.2 速度検出

タイマ・キャプチャ機能を用いてホールICの変化時点でのタイマのカウンタ値を瞬時に保存することで、通常の処理でのカウンタ値の保存の際に発生する遅延の影響がない速度計算により精度の高い速度検出を行います。

このシステムのBLDCモータ（3相4極）の場合、1つのホールICは1回転で4回変化するので、1/4回転に変化したタイマ値から速度を計算します。

$$N = \frac{60}{s \times n \times 4} = \frac{1875000}{n}$$

N：1分間の回転数（rpm）

s：タイマの分解能（1.6 μs）

n：発生回数

4：変化回数

このシステムでは300 rpm～3000 rpmの回転速度範囲をサポートしています。

4.3 電圧制御

モータに印加する電圧は、インバータのプラス側を20 kHzのキャリア周波数でチョップ動作（PWM）させた平均電圧で制御します。

マイナス側はPWMによる相補動作を行わないため、減速時のブレーキ・トルクは発生します。

4.4 速度制御

150 ms周期にPWMのデューティ比を調整することでモータの回転速度を制御します。

デューティ比の調整量は指定速度とモータの回転速度の差をPID制御で求めています。

電流マイナ・ループ制御のために直接PWMを操作するのではなく、電流マイナ・ループ用の基準電流として値を生成しています。

4.4.1 PID制御

回転速度と指定速度の差をフィードバックしてPWMのデューティ比（平均電圧）にPID制御を行うことで速度調整を行っています。PWMのデューティ比操作量は、サンプリング方式（離散値）に適した次に示す速度形PIDアルゴリズムを使用しています。

$$MV_n = MV_{n-1} + \Delta MV_n$$

$$\Delta MV_n = Kp(e_n - e_{n-1}) + Ki \times e_n + Kd((e_n - e_{n-1}) - (e_{n-1} - e_{n-2}))$$

MV_n : 今回操作量

MV_{n-1} : 前回操作量

ΔMV_n : 今回操作量差分

e_n : 今回の偏差（指定速度と実速度の差分）

e_{n-1} : 前回の偏差

e_{n-2} : 前々回の偏差

Kp : フィードバック・ゲイン（比例要素）

Ki : フィードバック・ゲイン（積分要素）

Kd : フィードバック・ゲイン（微分要素）

フィードバック・ゲインの最適値はモータ特性や負荷の有無で変化します。このシステムでは、動作確認時にカットアンドトライで求めた値を初期値として設定しています。

PID制御の周期は最低速度（300 rpm）での速度更新間隔（150 ms）に設定しています。

4.5 電流マイナ・ループ制御

1.5 ms周期にPWMのデューティ比を調整することでモータの回転を制御します。デューティ比の調整量は基準電流とモータの動作電流の差をPID制御で求めています。

速度調整で生成した基準電流に対して、測定したモータの動作電流の差を速度制御と同様にPID制御を行うことで、PWMのデューティ比を操作します。

測定するモータの動作電流は、インバータの切り替わり時に発生するノイズの影響を受けるため、ノイズ・フィルタ処理を行い、安定したモータの動作電流を得ています。

4.6 ノイズ・フィルタ処理

測定したモータの動作電流の値の平滑化を行うことで、ノイズ成分をカットし安定した測定値を得られるようにしています。

この制御プログラムでは、次に示すフィルタ演算式を採用しています。

$$u[n] = u[n - 1] + x[n] + y[n - 1]$$

$$y[n] = \frac{u[n]}{\alpha}$$

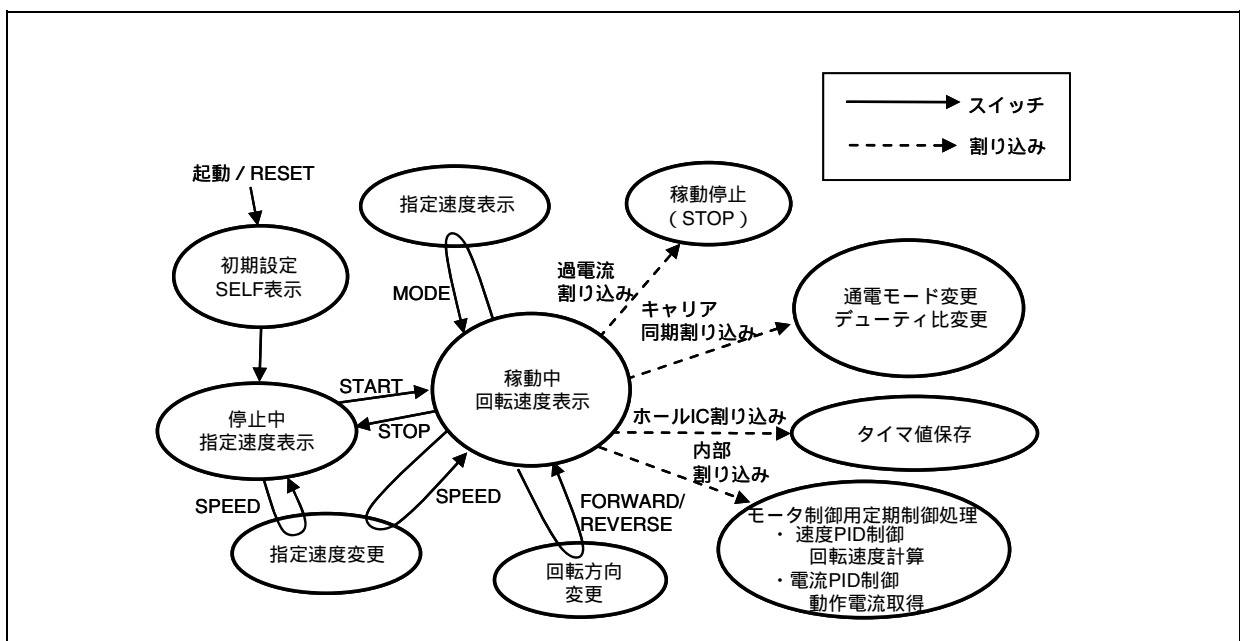
- u : 電流の積分量
- x : 今回測定分の電流値
- y : 平滑後の電流値
- α : 平滑係数

4.7 モジュール構成

4.7.1 システム状態遷移

システムの状態遷移を次に示します。

図4-1 システムの状態遷移



キャリア同期割り込みは50 μsごとに発生します。

内部割り込みは、速度PID制御は150 msごとに、電流PID制御は1.5 msごとに発生します。

4.7.2 ライブラリ

この制御プログラムでは、モータの動作指示およびモータ制御で使用する各種パラメータの設定についての外部インタフェース用関数を用意しています。この関数群を用いることにより、ユーザはモータ制御のパラメータについてのチップのレジスタ設定を変更しなくてもモータ制御できます。

4.8 関数一覧

制御プログラムは複数の関数で構成されています。次に各関数の機能概要を示します。より詳細な処理については4.9 フロー・チャートを参照してください。

4.8.1 外部インタフェース用関数一覧

表4-3 外部インタフェース用関数一覧 (1/2)

関数名	機能概要
Motor_UseGUI()	GUIによるモータの操作の使用を指示します。
Motor_SetupFunc()	モータ制御に使用する各機能の初期設定を行います。
Motor_StartMotor()	モータの回転開始を指示します。
Motor_StopMotor()	モータの回転停止を指示します。
Motor_GetStatusSys()	現在のシステムの動作状態を取得します。
Motor_GetStatusErr()	現在のシステムのエラー状態を取得する。
Motor_SetCW()	モータの回転方向をCWに設定します。
Motor_SetCCW()	モータの回転方向をCCWに設定します。
Motor_GetDirection()	現在のモータの回転方向の設定値を取得します。
Motor_SetSpeedRef()	目標回転数を設定します。
Motor_GetSpeedRef()	現在の目標回転数を取得します。
Motor_GetSpeedNow()	現在の回転数を取得します。
Motor_GetSpeedRefInner()	現在の処理中の目標回転数を取得します。
Motor_SetCurrentRef()	目標電流値を設定します。
Motor_GetCurrentRef()	現在の目標電流値を取得します。
Motor_GetCurrentNow()	現在の電流値を取得します。
Motor_GetCurrentRefInner()	現在の処理中の目標電流値を取得します。
Motor_SetPwmRef()	目標PWM値を設定します。
Motor_GetPwmRef()	現在の目標PWM値を取得します。
Motor_GetPwmNow()	現在のPWM値を取得します。
Motor_SetSpeedKp()	速度制御用PIDパラメータ (Kp) を設定します。
Motor_GetSpeedKp()	現在の速度制御用PIDパラメータ (Kp) を取得します。
Motor_SetSpeedKi()	速度制御用PIDパラメータ (Ki) を設定します。
Motor_GetSpeedKi()	現在の速度制御用PIDパラメータ (Ki) を取得します。
Motor_SetSpeedKd()	速度制御用PIDパラメータ (Kd) を設定します。
Motor_GetSpeedKd()	現在の速度制御用PIDパラメータ (Kd) を取得します。
Motor_SetCurrentKp()	電流制御用PIDパラメータ (Kp) を設定します。
Motor_GetCurrentKp()	現在の電流制御用PIDパラメータ (Kp) を取得します。
Motor_SetCurrentKi()	電流制御用PIDパラメータ (Ki) を設定します。
Motor_GetCurrentKi()	現在の電流制御用PIDパラメータ (Ki) を取得します。
Motor_SetCurrentKd()	電流制御用PIDパラメータ (Kd) を設定します。

表4-3 外部インタフェース用関数一覧(2/2)

関数名	機能概要
Motor_GetCurrentKd()	現在の電流制御用PIDパラメータ (Kd) を取得します。
Motor_SetMode()	モータの制御方式を速度指示制御 / 電流指示制御 / 電圧指示制御のいずれかに指定した指示対象に設定します。
Motor_GetMode()	現在の指示対象を取得します。
Motor_SetIntervalRpm()	速度制御のPID演算間隔を設定します。
Motor_GetIntervalRpm()	現在の速度制御のPID演算間隔を取得します。
Motor_SetIntervalCurrent()	電流制御 (電流マイナ・ループ) のPID演算間隔を設定します。
Motor_GetIntervalCurrent()	現在の電流制御 (電流マイナ・ループ) のPID演算間隔を取得します。
Motor_SetLimitCurrentSoft()	ソフトウェア過電流検知のリミットの電流値を設定します。
Motor_GetLimitCurrentSoft()	現在のソフトウェア過電流検知のリミットの電流値を取得します。
Motor_SetCarrier()	キャリア周波数を設定します。
Motor_GetCarrier()	現在のキャリア周波数を取得します。
Motor_SetDeadtime()	デッド・タイムを設定します。
Motor_GetDeadtime()	現在のデッド・タイムを取得します。
Motor_SetPole()	極対数を設定します。
Motor_GetPole()	現在の極対数を取得します。
Motor_SetMaxSpeed()	最大回転数を設定します。
Motor_GetMaxSpeed()	現在の最大回転数を取得します。
Motor_SetHighSpeed()	目標回転数の上限値を設定します。
Motor_GetHighSpeed()	現在の目標回転数の上限値を取得します。
Motor_SetLowSpeed()	目標回転数の下限値を設定します。
Motor_GetLowSpeed()	現在の目標回転数の下限値を取得します。
Motor_SetHighCurrent()	目標電流値の上限値を設定します。
Motor_GetHighCurrent()	現在の目標電流値の上限値を取得します。
Motor_SetLowCurrent()	目標電流値の下限値を設定します。
Motor_GetLowCurrent()	現在の目標電流値の下限値を取得します。
Motor_SetHighPwm()	目標PWM値の上限値を設定します。
Motor_GetHighPwm()	現在の目標PWM値の上限値を取得します。
Motor_SetLowPwm()	目標PWM値の下限値を設定します。
Motor_GetLowPwm()	現在の目標PWM値の下限値を取得します。
Motor_SetSpeedPer()	単位時間での回転数の変化量を設定します。
Motor_GetSpeedPer()	現在の単位時間での回転数の変化量を取得します。
Motor_SetCurrentPer()	単位時間での電流値の変化量を設定します。
Motor_GetCurrentPer()	現在の単位時間での電流値の変化量を取得します。
Motor_SetPwmPer()	単位時間でのPWM値の変化量を設定します。
Motor_GetPwmPer()	現在の単位時間でのPWM値の変化量を取得します。
Motor_SetGainAd()	A/D変換値を電流値へ換算する際の係数を設定します。
Motor_GetGainAd()	現在のA/D変換用ゲインを取得します。
Motor_SetOffsetAd()	A/D変換値を電流値へ換算する際のオフセットを設定します。
Motor_GetOffsetAd()	現在のA/D変換用オフセットを取得します。
Motor_SetFreqCutoff()	カットオフ周波数を設定します。
Motor_GetFreqCutoff()	現在のカットオフ周波数を取得します。

4.8.2 内部関数一覧

表4-4 内部関数一覧(1/2)

(a) motor.cの内部関数一覧(1/2)

関数名	機能	
	処理名	概要
MotorSetup()	初期設定処理	システムの初期設定処理を行います。
MotorStart()	起動処理	モータの回転開始を行います。
MotorRestart()	再起動処理	反転停止後の再起動を行います。
MotorStop()	停止処理	モータの回転停止を行います。
Init_LoadVariable()	変数初期設定	システムで使用する変数の初期設定処理を行います。
Init_Osc()	クロック設定処理	CPU動作速度などのクロック設定を行います。
Init_Port()	ポート初期設定処理	モータ制御で使用するポートの設定を行います。
Init_InverterTimer()	インバータ制御用タイマ初期設定処理	インバータ・タイマの設定を行います。
Init_TimerForRPM()	TMQ0初期設定処理	16ビット・タイマの設定を行います。
Init_AD()	A/D変換初期設定処理	A/D変換の設定を行います。
Init_IntervalTimer()	制御間隔管理用インターバル・タイマ初期設定処理	インターバル・タイマの初期設定を行います。
Start_AD()	A/D変換起動処理	A/D変換の起動を行います。
Start_TimerForRPM()	TMQ0起動処理	16ビット・タイマの起動を行います。
Start_IntervalTimer()	TMP1起動処理	16ビット・タイマの起動を行います。
Start_InverterTimer()	インバータ制御タイマ起動処理	インバータ・タイマの起動を行います。
Stop_TimerForRPM()	TMQ0停止処理	16ビット・タイマの停止を行います。
Stop_InverterTimer()	インバータ制御タイマ停止処理	インバータ・タイマの停止を行う。
EI_ForRPM()	INTTQ0CC3許可処理	速度情報取得用割り込みの許可を行います。
EI_ForOC()	INTP1許可処理	過電流割り込みの許可を行います。
EI_ForInverterTimer()	INTTQ1OV許可処理	キャリア同期割り込みの許可を行います。
EI_ForInverterBoard()	インバータ・ボード動作許可処理	インバータ・ボードを動作させます。
DI_ForRPM()	INTTQ0CC3禁止処理	速度情報取得用割り込みの禁止を行います。
DI_ForInverterTimer()	INTTQ1OV禁止処理	キャリア同期割り込みの禁止を行います。
DI_ForInverterBoard()	インバータ・ボード動作禁止処理	インバータ・ボードを停止させます。
Exec_PIDforSpeed()	速度指示制御PID演算	速度指示制御のPID演算を指示します。
Exec_PIDforCurrent()	電流指示制御PID演算	電流指示制御のPID演算を指示します。
Read_HallIC()	ホールICの情報を取得	ホールIC情報をポートから取得します。
Update_Parameter()	設定値更新処理	外部インタフェースで指定された値の更新を行います。
Change_Carrier()	キャリア周波数更新処理	指示されたキャリア周波数に従い、分解能の算出をします。
Change_Deadtime()	デッド・タイム更新処理	指示されたデッド・タイム時間をタイマ・コンペア値に変換します。
Change_Pole()	極対数更新処理	指示された極対数から測定回転数の基数を算出します。
Change_InvTimRpm()	速度指示制御間隔更新処理	指示された速度指示制御間隔時間を内部カウンタのコンペア値に変換します。

表4 - 4 内部関数一覧 (2/2)

(a) motor.cの内部関数一覧 (2/2)

関数名	機 能	
	処理名	概 要
Change_InvTimCmnt()	電流指示制御間隔更新処理	指示された電流指示制御間隔時間を内部カウンタのコンペア値に変換します。
Change_DeltaSpeed()	速度の変化量更新処理	指示された速度の変化量を内部の処理値に変換します。
Change_DeltaCurrent()	電流の変化量更新処理	指示された電流の変化量を内部の処理値に変換します。
Change_DeltaPwm()	PWMの変化量更新処理	指示されたPWMの変化量を内部の処理値に変換します。
Change_FreqCutoff()	カットオフ周波数更新処理	指示されたカットオフ周波数から内部の処理値に変換します。
Change_Pattern()	通電パターン切り替え処理	通電パターンを指示されたパターンに設定します。
Change_Duty()	デューティ比変更処理	指示されたPWM値を設定します。
Get_Current()	電流値取得処理	動作電流のフィルタ処理を行います。
int_fault()	ハードウェア過電流検知割り込み処理	モータ処理の停止を行います。
int_speed()	速度情報取得用割り込み処理	速度情報の更新を促すフラグの設定を行います。
int_carrier()	キャリア同期割り込み(谷)処理	通電パターン切り替え, デューティ比の更新, モータ動作状況診断を行います。
int_PID_Current()	電流マイナ・ループ用タイマ割り込み処理	電流マイナ・ループ用演算処理を行います。

(b) sub_mcio.cの内部関数一覧

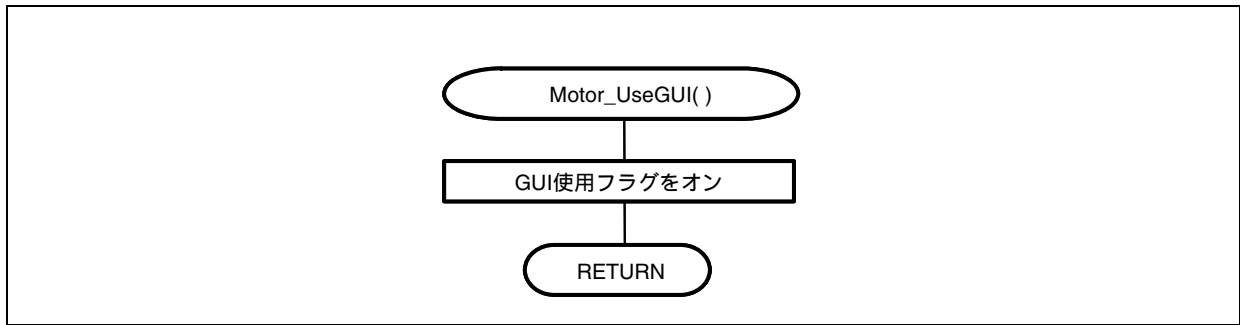
関数名	機 能	
	処理名	概 要
print_error	エラー表示	エラーが起きたらLEDに表示します。
get_sw	スイッチの読み出し	制御スイッチの状態を返します。
speed_print	LED速度表示	モータ制御停止中は指定速度を表示します。 モータ制御中でMODEスイッチが押されている場合は、モータの回転速度を表示します。
led_print	引数を4桁に変換処理	LEDに1-4桁の値を表示します。
led_set	LEDデータ表示	指定されたLEDに値を出力します。
vol2speed	指定速度読み出し	可変抵抗の電圧をA/D変換した値から指定速度を計算します。
get_vol	A/D変換値取得	可変抵抗の電圧をA/D変換値から読み出します。
wait	時間調整	ウェイト時間を設定します。
startup_disp	スタートアップ表示	LEDに“SELF”と表示します。
init_PORT	ポート初期設定	スイッチとLED用のポートを設定します。
init_TM	タイマ初期設定	ウェイト用のタイマを設定します。
start_TM	タイマ・スタート	ウェイトを開始します。
wait_TM	タイマ・ストップ	割り込み信号を確認するとタイマを停止させます。
clear_WDTM	ウォッチドッグ・タイマ・カウンタ・クリア	ウォッチドッグ・タイマのカウンタをクリアします。

4.9 フロー・チャート

4.9.1 外部インタフェース用関数のフロー・チャート

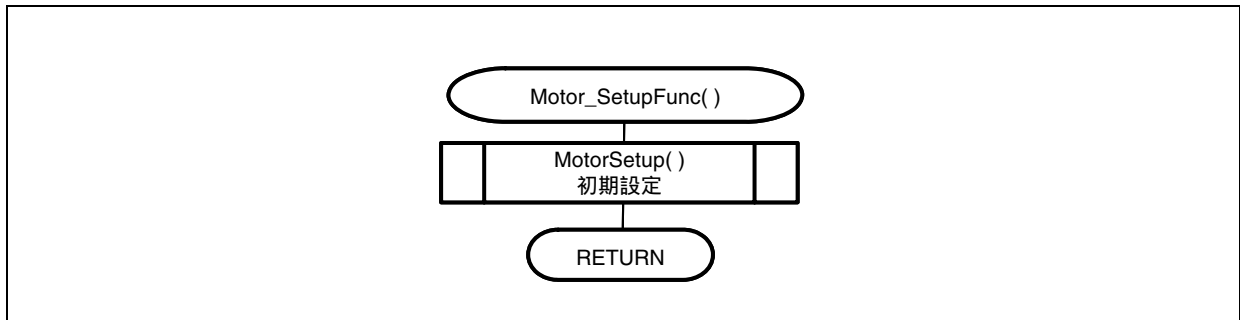
(1) GUI操作の設定処理

図4 - 2 GUI操作の設定処理



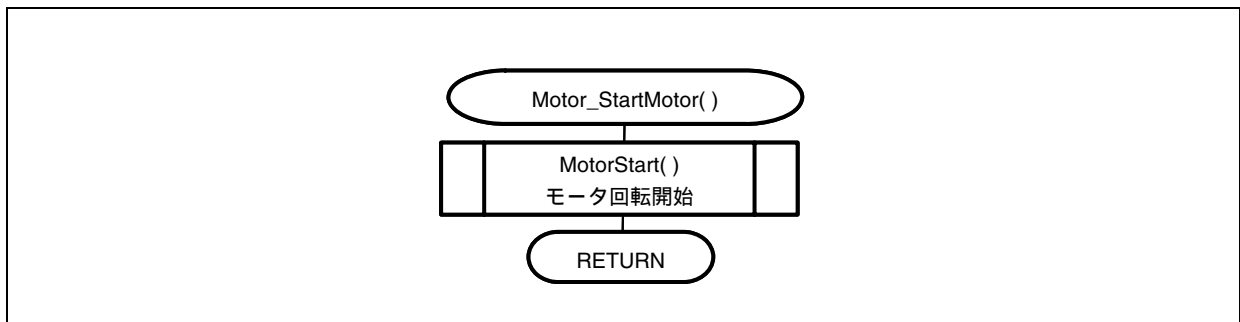
(2) モータ制御用各種機能の初期設定処理

図4 - 3 モータ制御用各種機能の初期設定処理



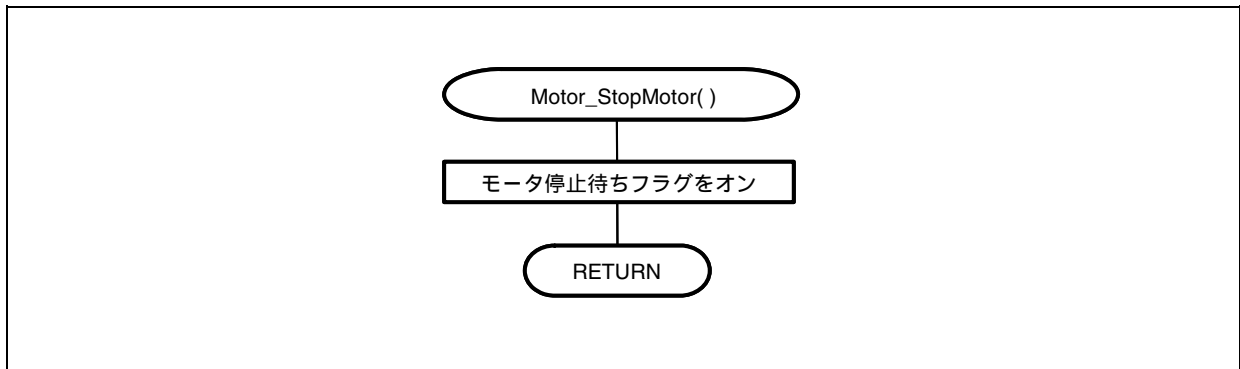
(3) モータ回転開始指示処理

図4 - 4 モータ回転開始指示処理



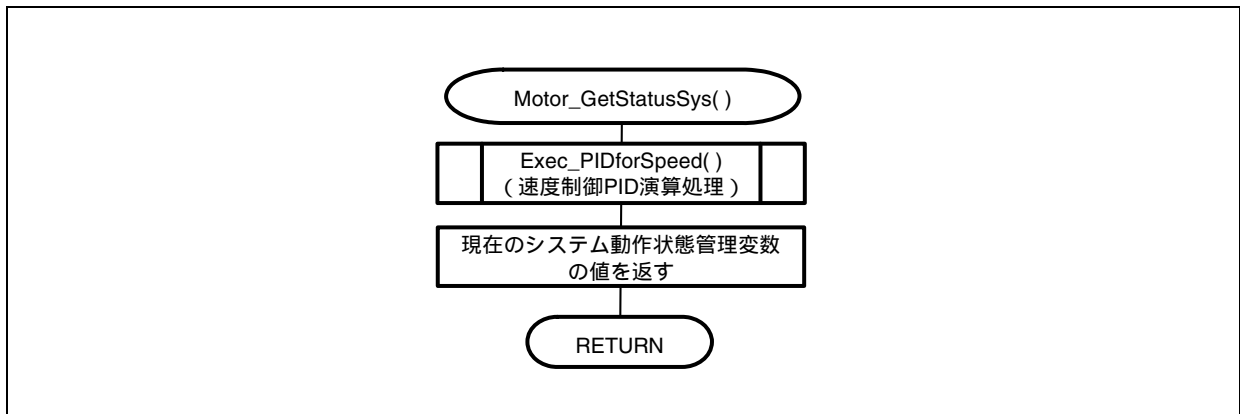
(4) モータ回転停止指示処理

図4 - 5 モータ回転停止指示処理



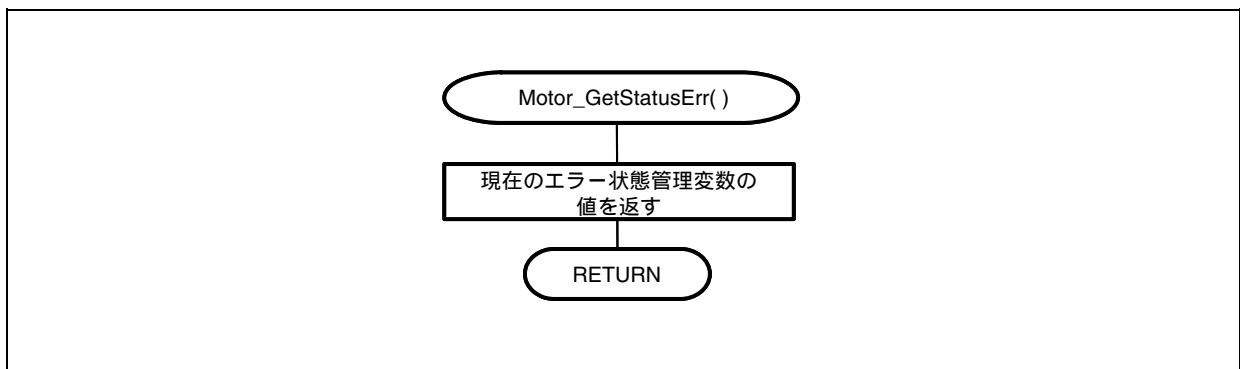
(5) 現在のシステム動作状態の取得処理

図4 - 6 現在のシステム動作状態の取得処理



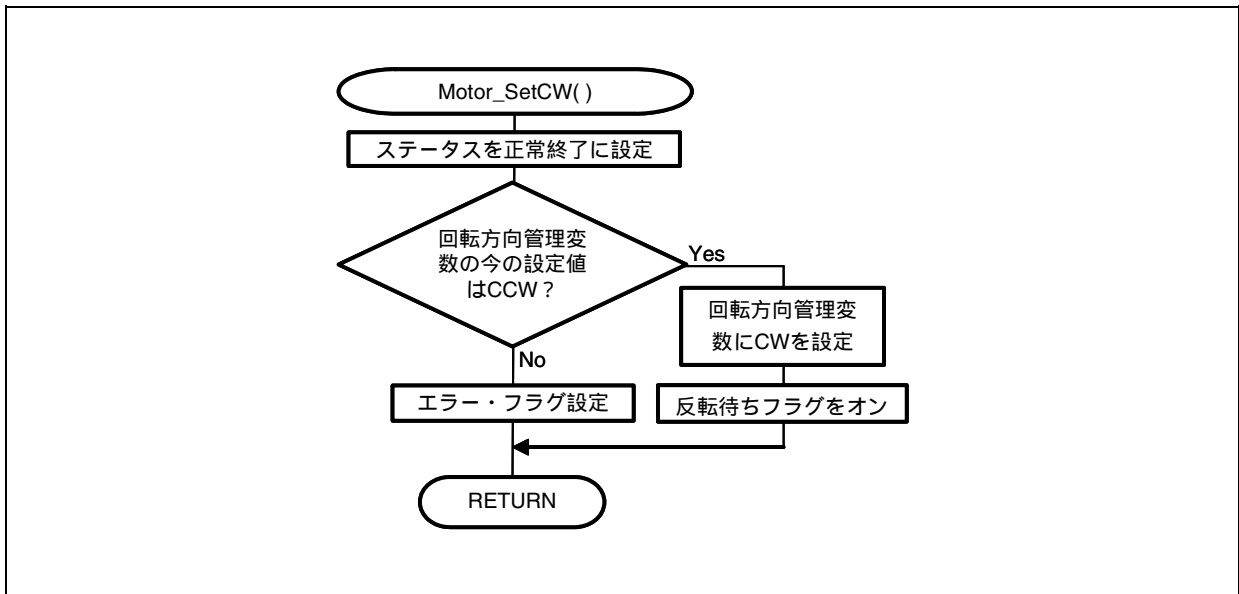
(6) 現在のエラー状態の取得処理

図4 - 7 現在のエラー状態の取得処理



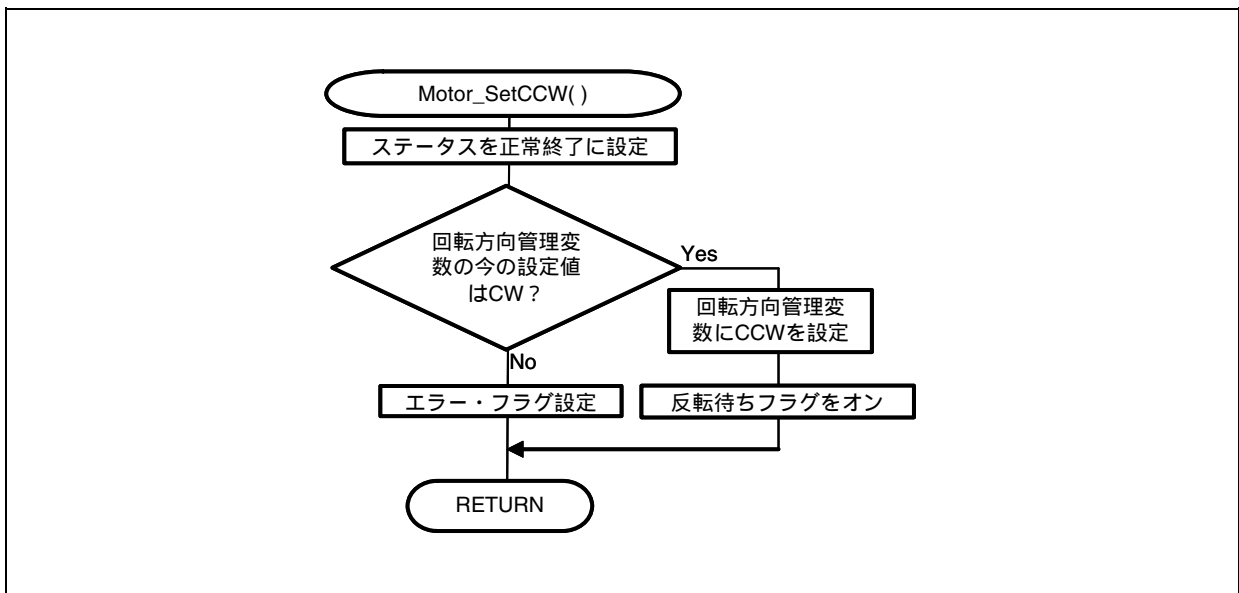
(7) モータ回転方向をCWに設定処理

図4 - 8 モータ回転方向をCWに設定処理



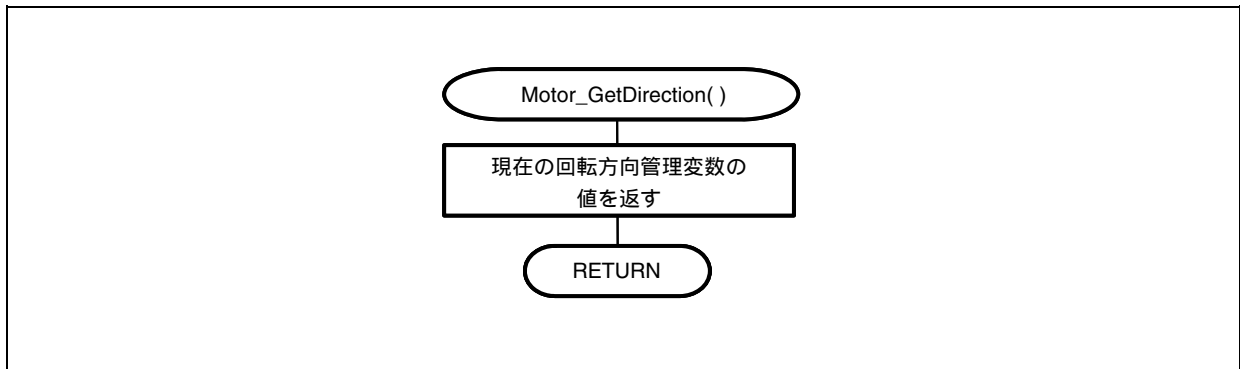
(8) モータ回転方向をCCWに設定処理

図4 - 9 モータ回転方向をCCWに設定処理



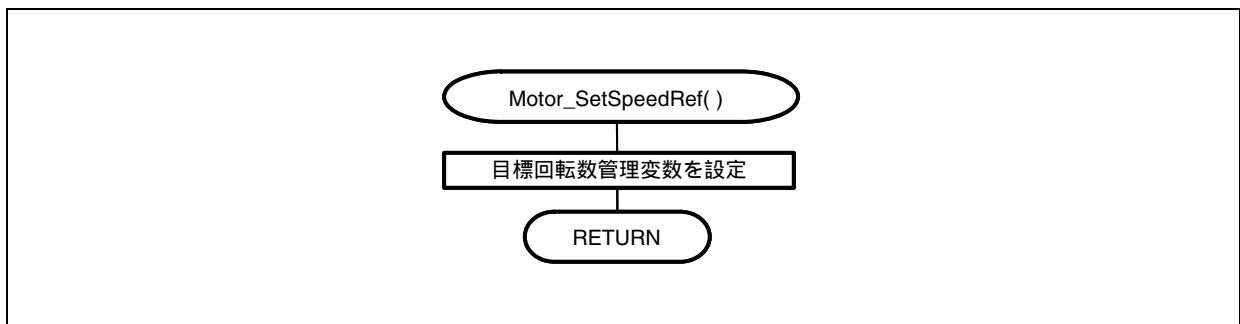
(9) 現在のモータ回転方向の設定値の取得処理

図4 - 10 現在のモータ回転方向の設定値の取得処理



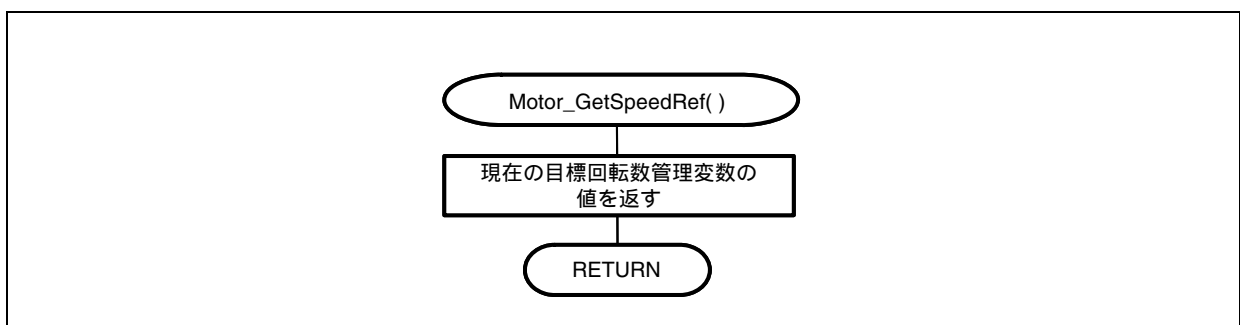
(10) 目標回転数の設定処理

図4 - 11 目標回転数の設定処理



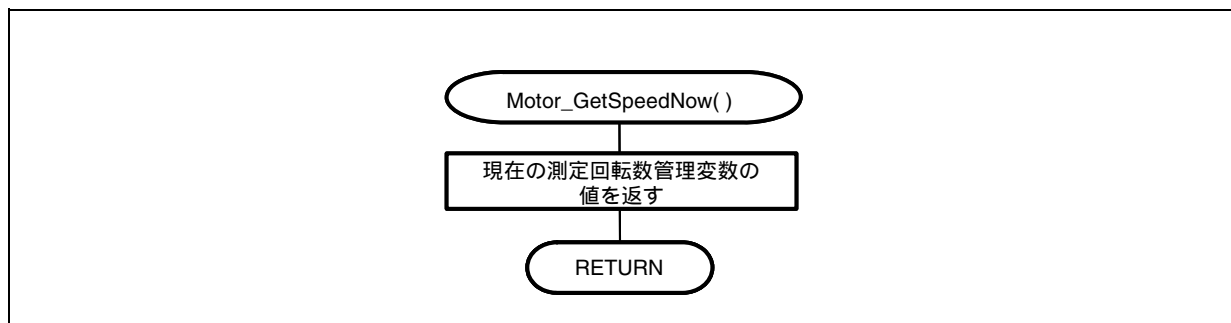
(11) 現在の目標回転数の取得処理

図4 - 12 現在の目標回転数の取得処理



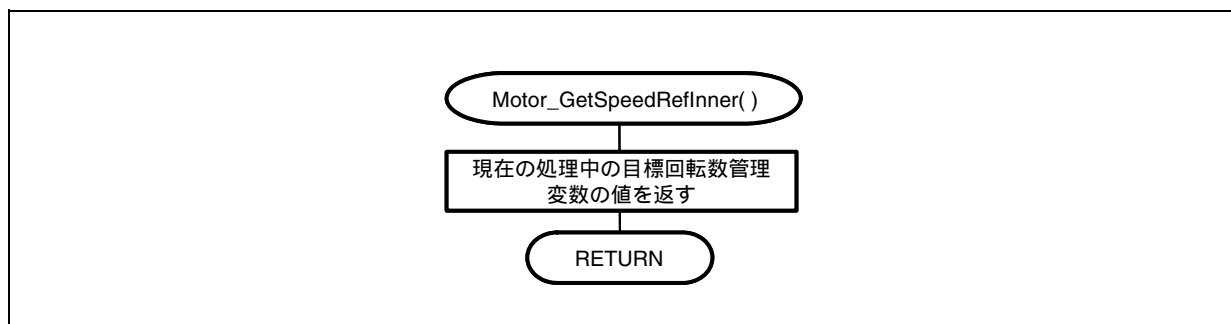
(12) 現在の回転数の取得処理

図4 - 13 現在の回転数の取得処理



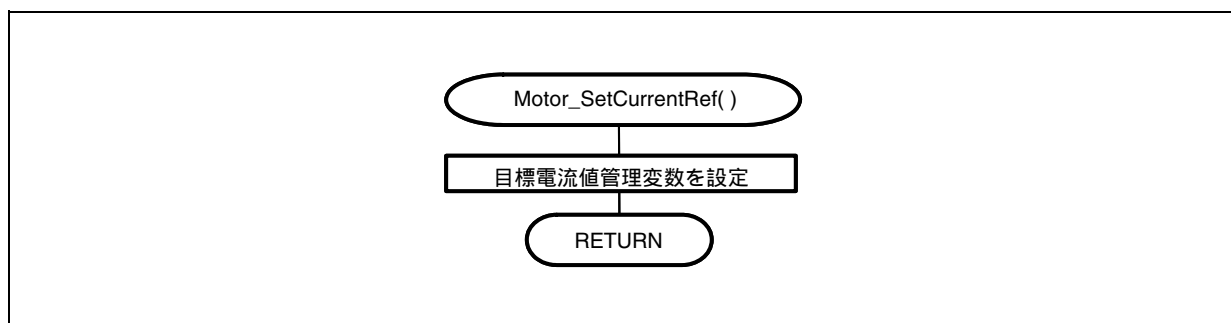
(13) 現在の処理中目標回転数の取得処理

図4 - 14 現在の処理中目標回転数の取得処理



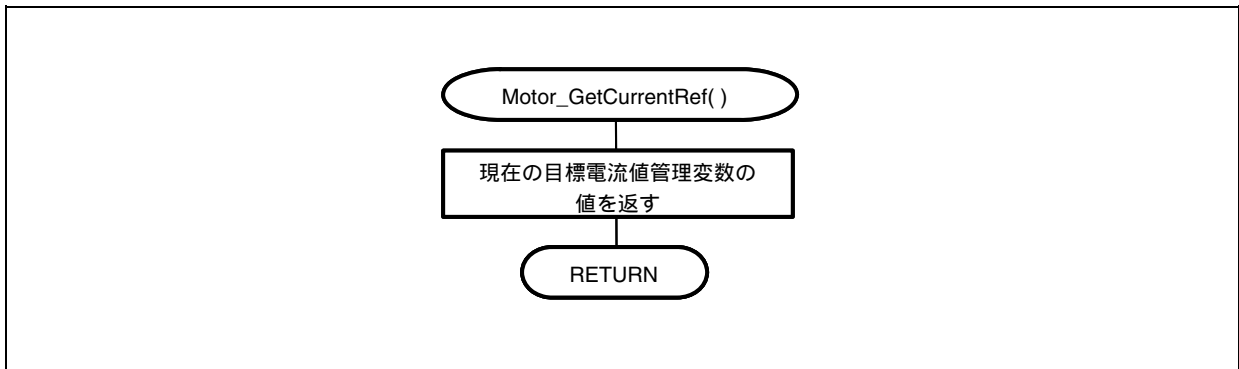
(14) 目標電流値の設定処理

図4 - 15 目標電流値の設定処理



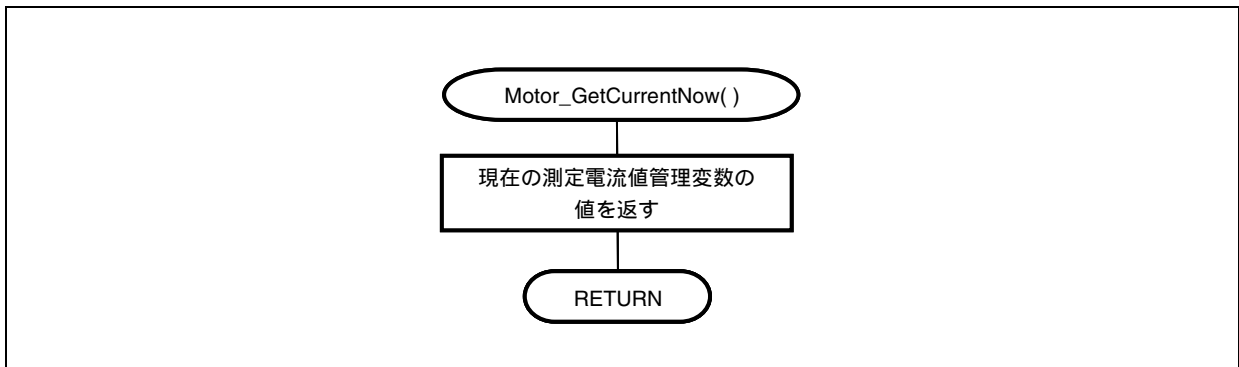
(15) 現在の目標電流値の取得処理

図4 - 16 現在の目標電流値の取得処理



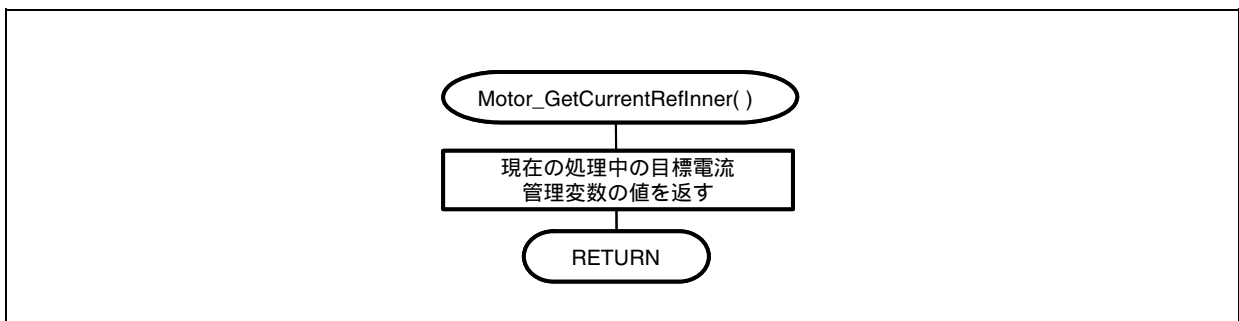
(16) 現在の測定電流値の取得処理

図4 - 17 現在の測定電流値の取得処理



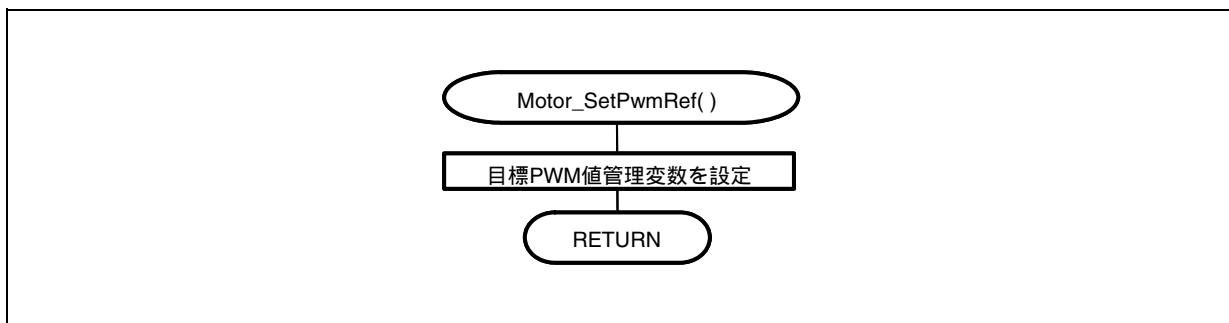
(17) 現在の処理中目標電流値の取得処理

図4 - 18 現在の処理中目標電流値の取得処理



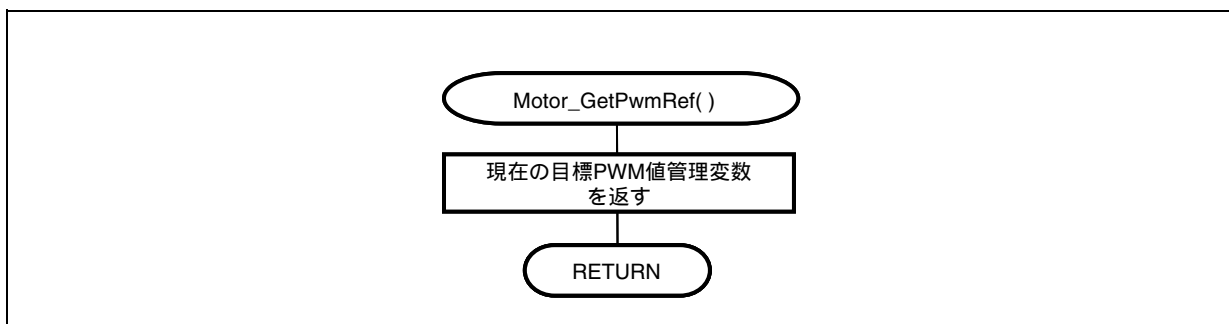
(18) 目標PWM値の設定処理

図4 - 19 目標PWM値の設定処理



(19) 現在の目標PWM値の取得処理

図4 - 20 現在の目標PWM値の取得処理



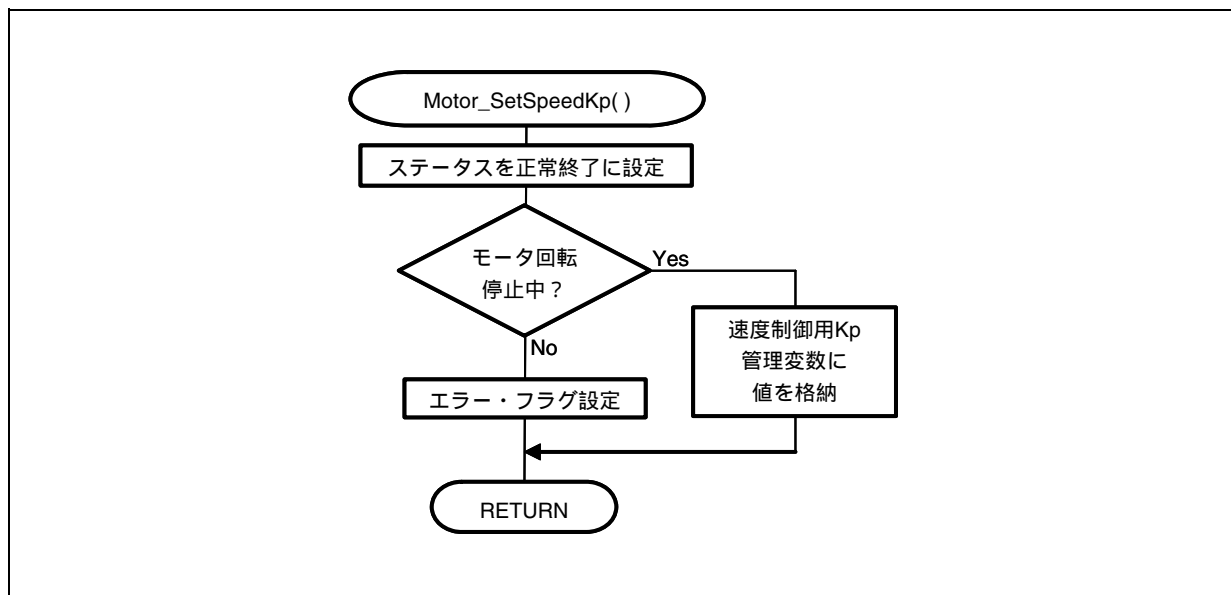
(20) 現在の動作PWM値の取得処理

図4 - 21 現在の動作PWM値の取得処理



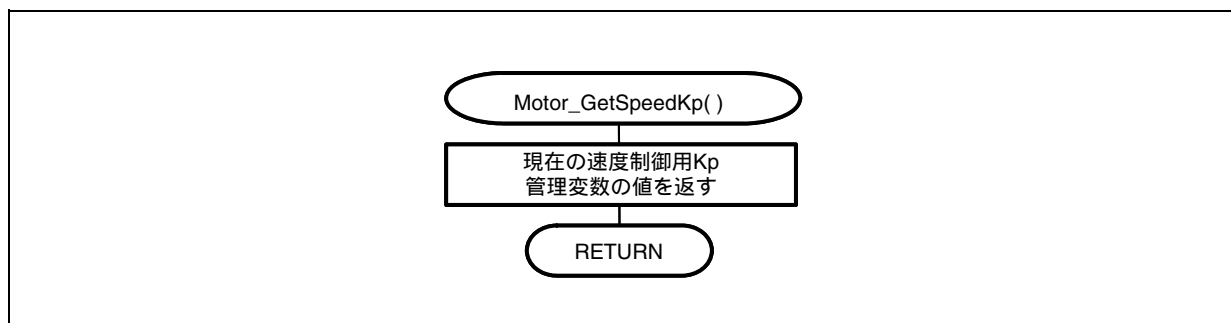
(21) 速度制御用Kpの設定処理

図4 - 22 速度制御用Kpの設定処理



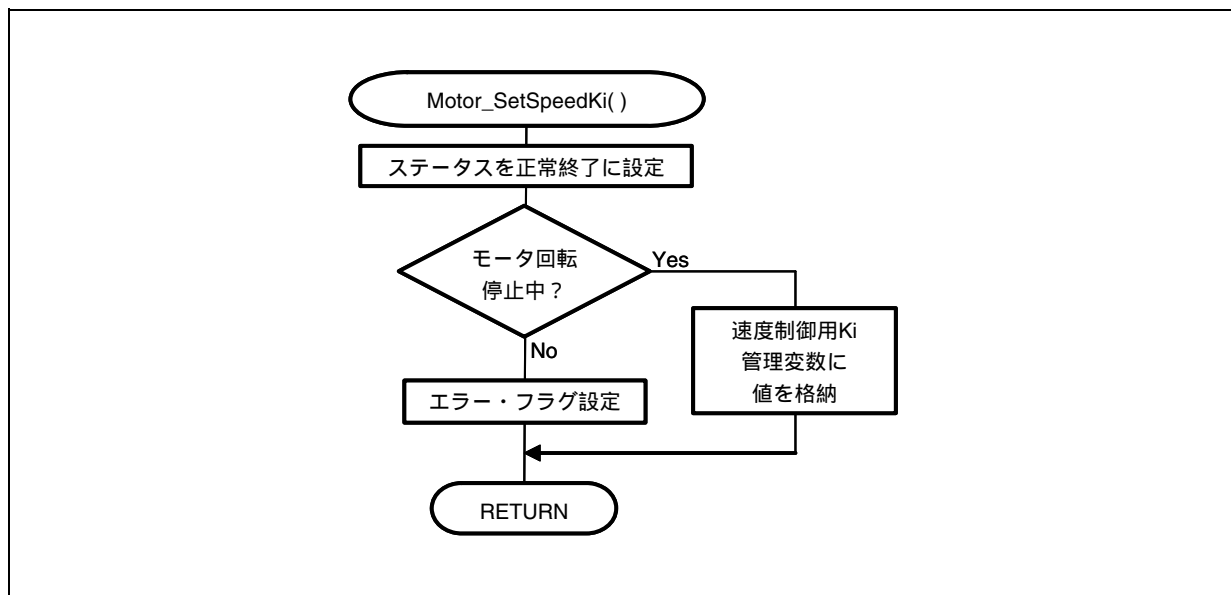
(22) 現在の速度制御用Kpの取得処理

図4 - 23 現在の速度制御用Kpの取得処理



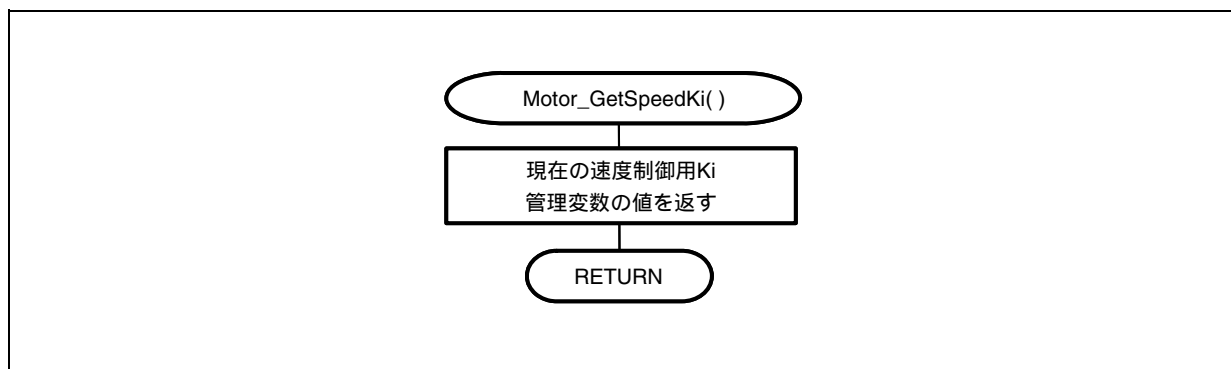
(23) 速度制御用Kiの設定処理

図4 - 24 速度制御用Kiの設定処理



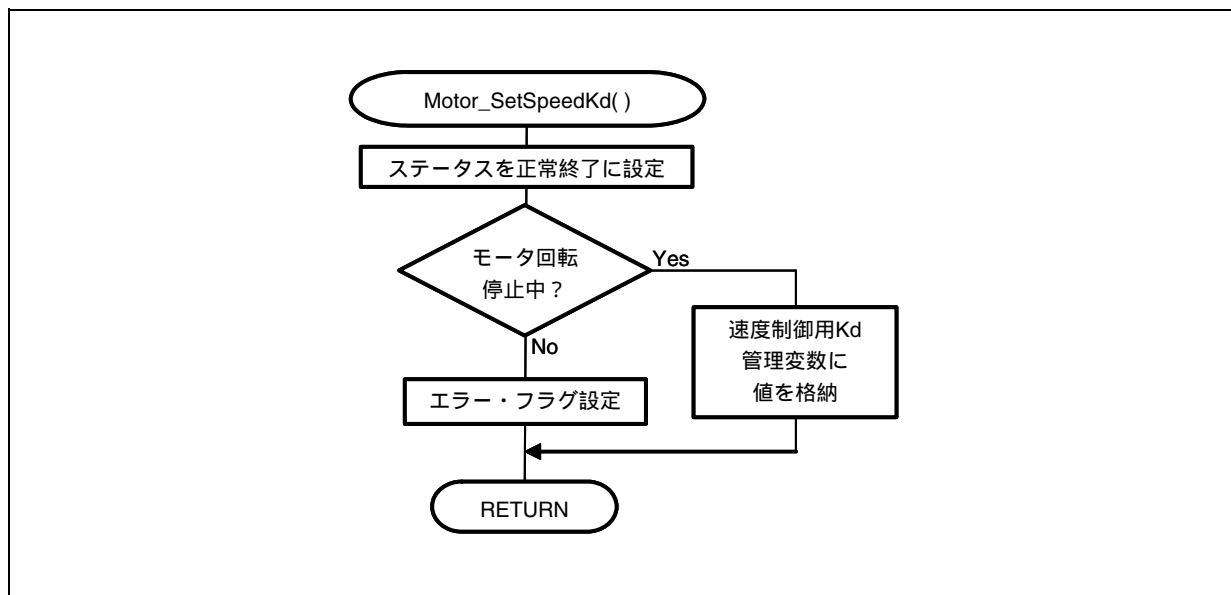
(24) 現在の速度制御用Kiの取得処理

図4 - 25 現在の速度制御用Kiの取得処理



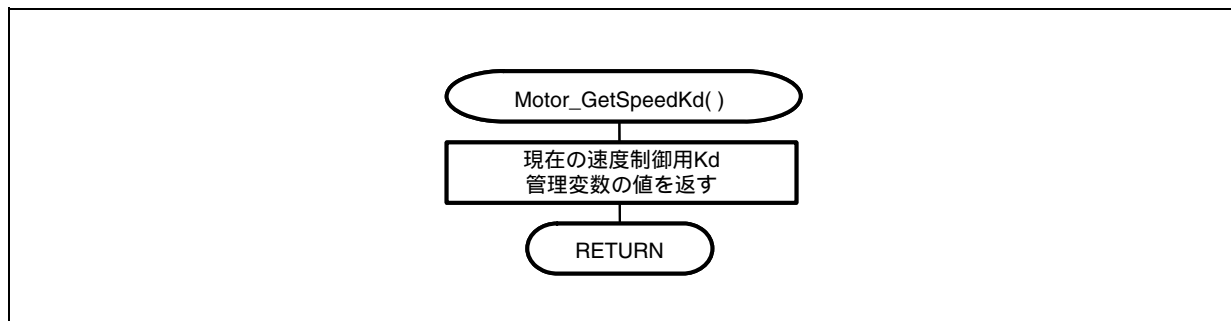
(25) 速度制御用Kdの設定処理

図4 - 26 速度制御用Kdの設定処理



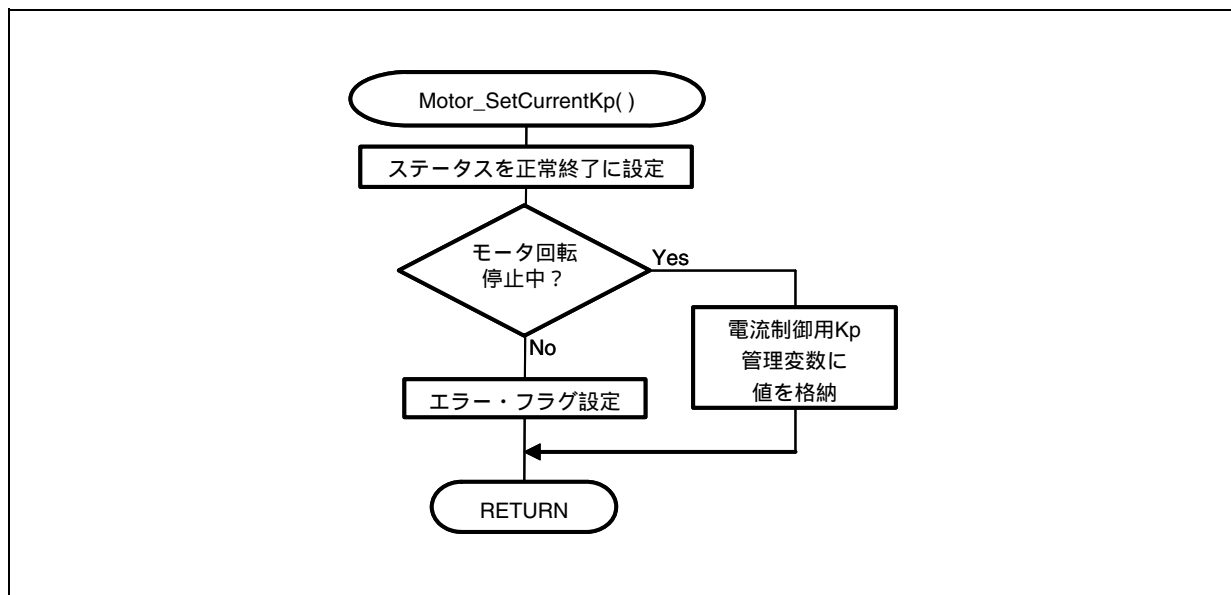
(26) 現在の速度制御用Kdの取得処理

図4 - 27 現在の速度制御用Kdの取得処理



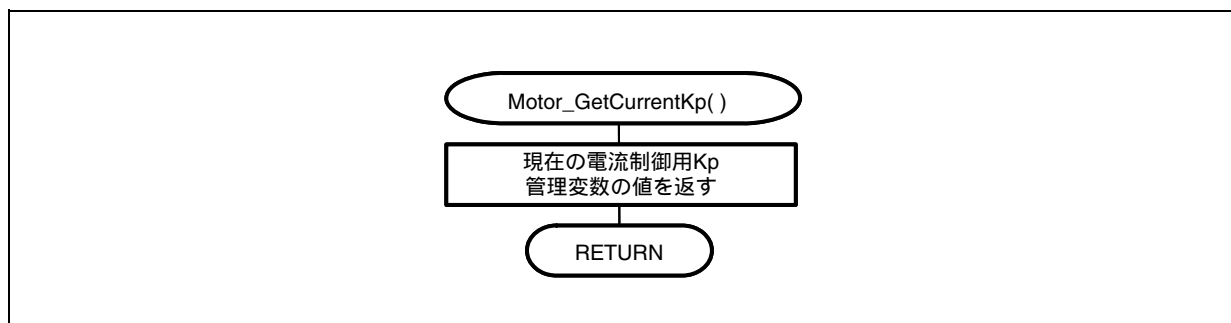
(27) 電流制御用Kpの設定処理

図4 - 28 電流制御用Kpの設定処理



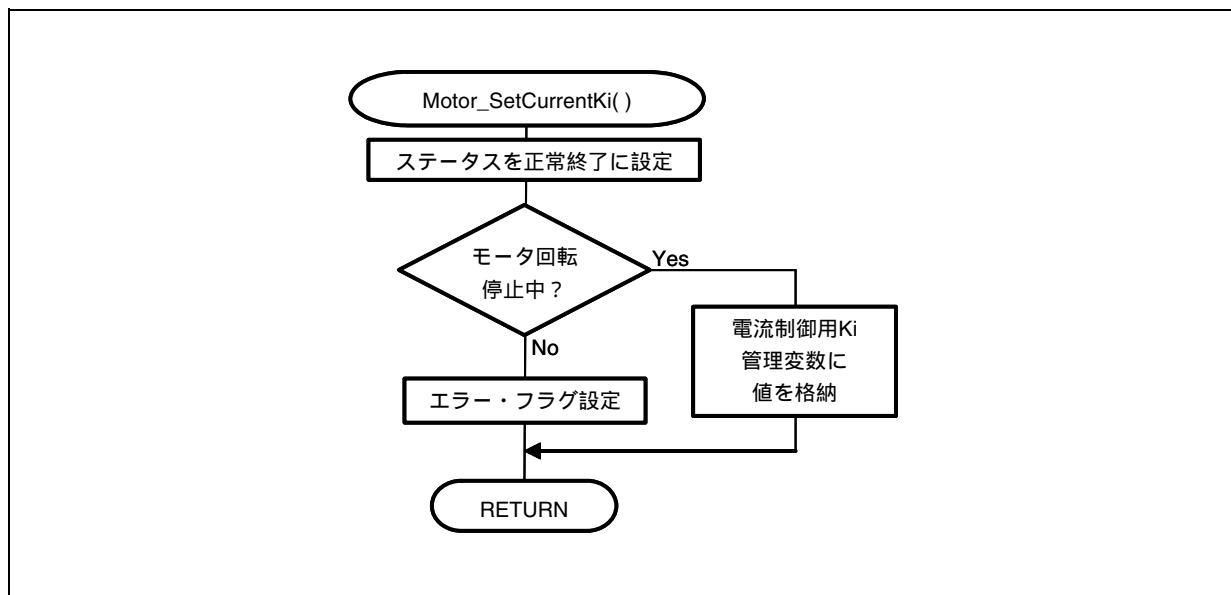
(28) 現在の電流制御用Kpの取得処理

図4 - 29 現在の電流制御用Kpの取得処理



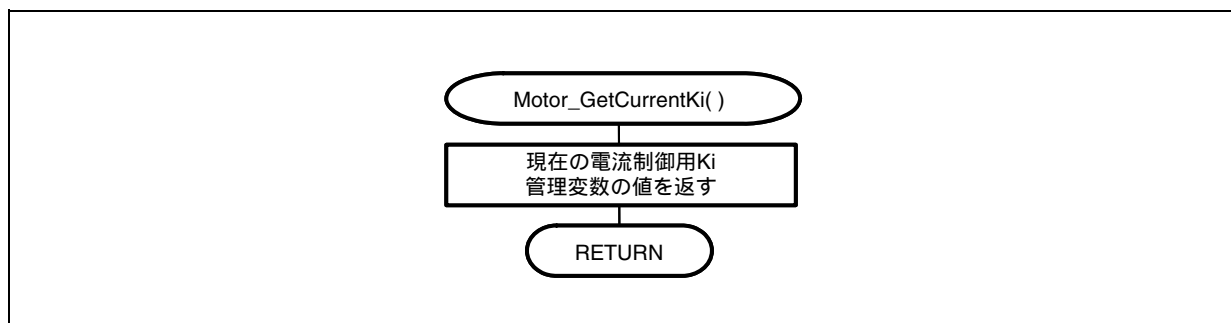
(29) 電流制御用Kiの設定処理

図4 - 30 電流制御用Kiの設定処理



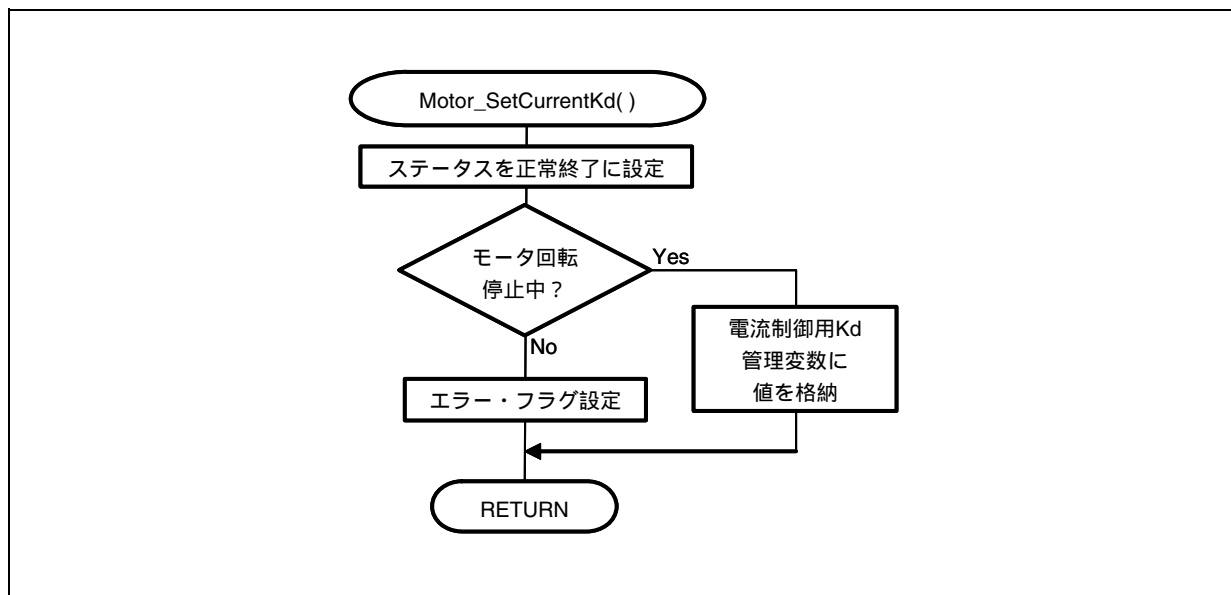
(30) 現在の電流制御用Kiの取得処理

図4 - 31 現在の電流制御用Kiの取得処理



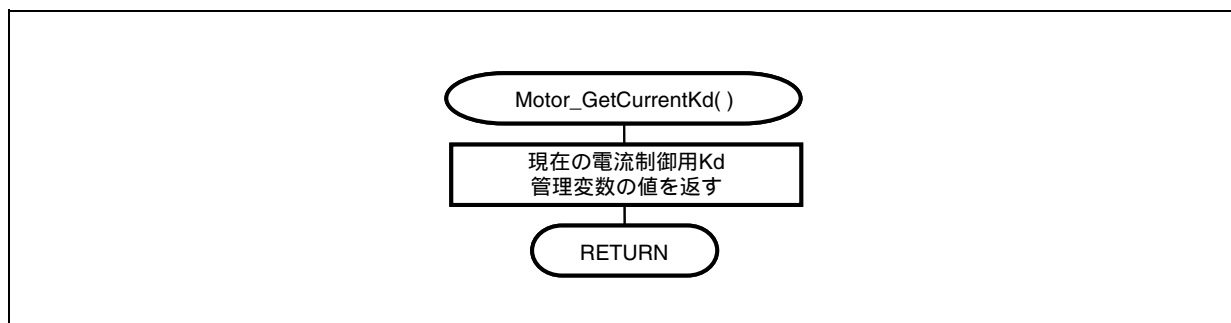
(31) 電流制御用Kdの設定処理

図4 - 32 電流制御用Kdの設定処理



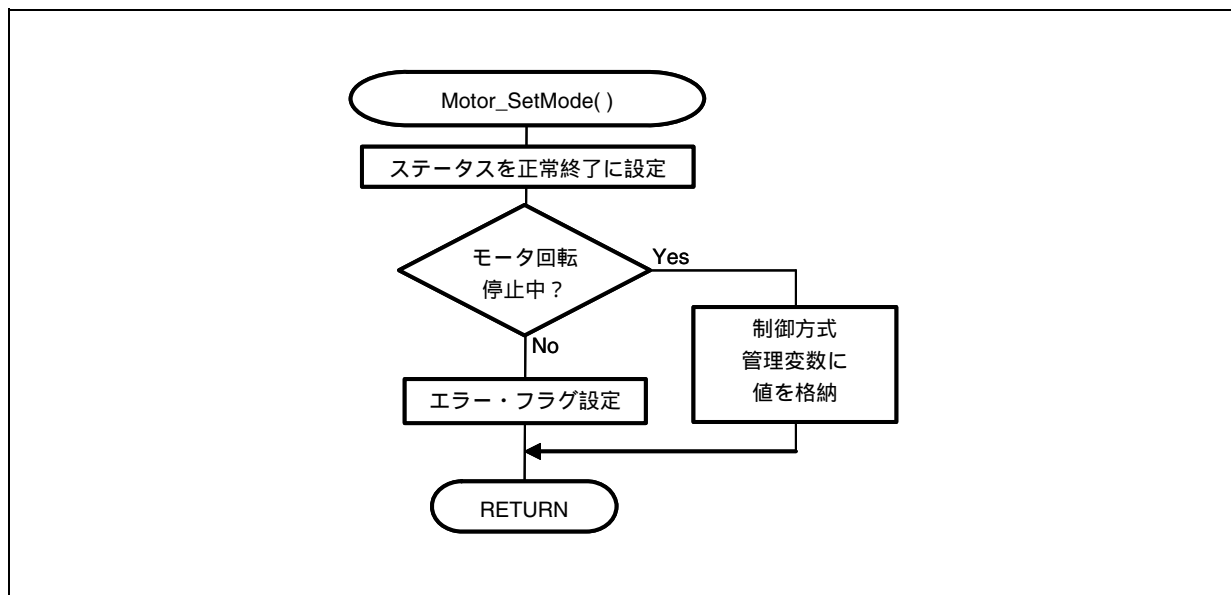
(32) 現在の電流制御用Kdの取得処理

図4 - 33 現在の電流制御用Kdの取得処理



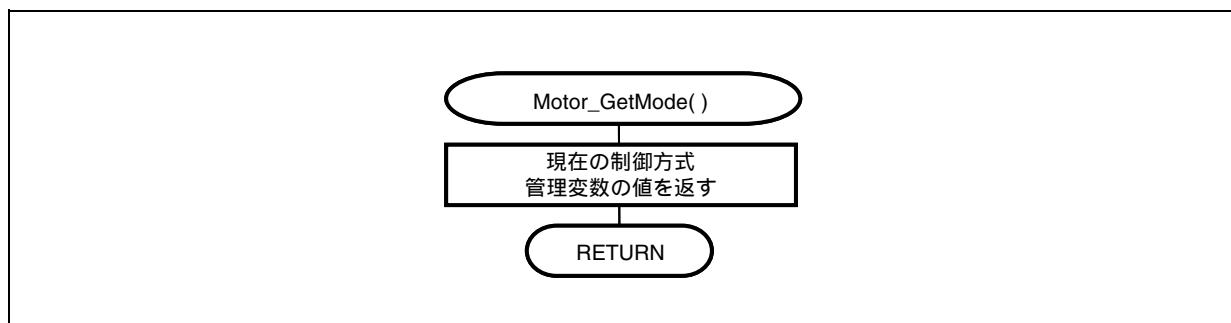
(33) 制御方式の設定処理

図4 - 34 制御方式の設定処理



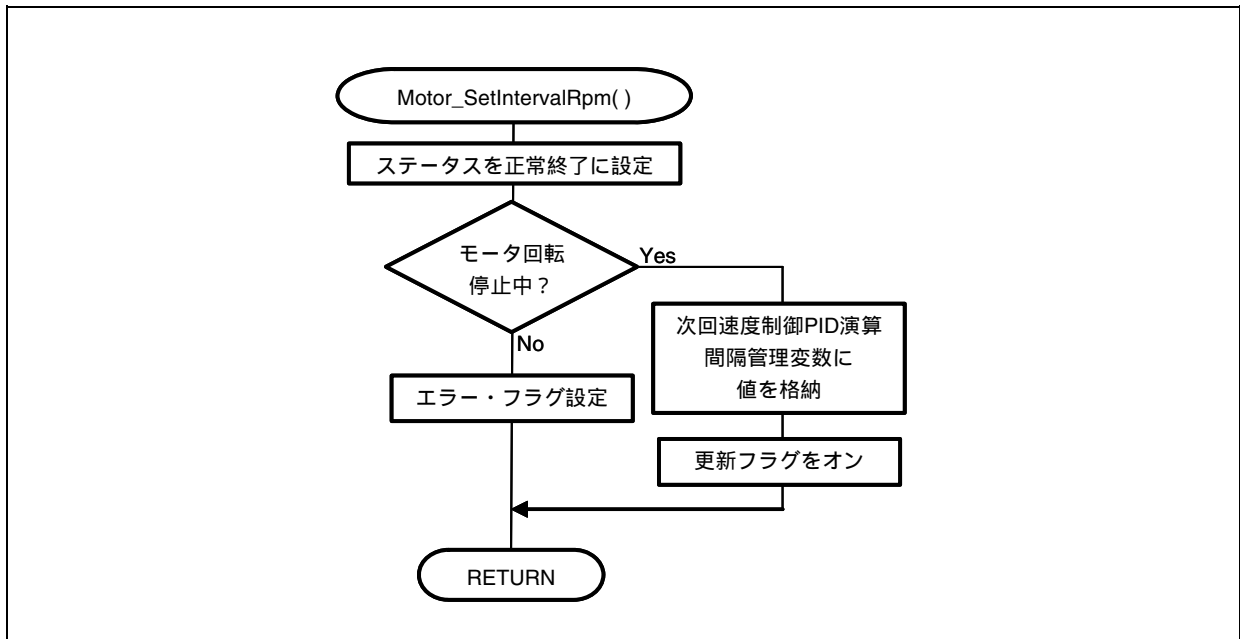
(34) 現在の制御方式の取得処理

図4 - 35 現在の制御方式の取得処理



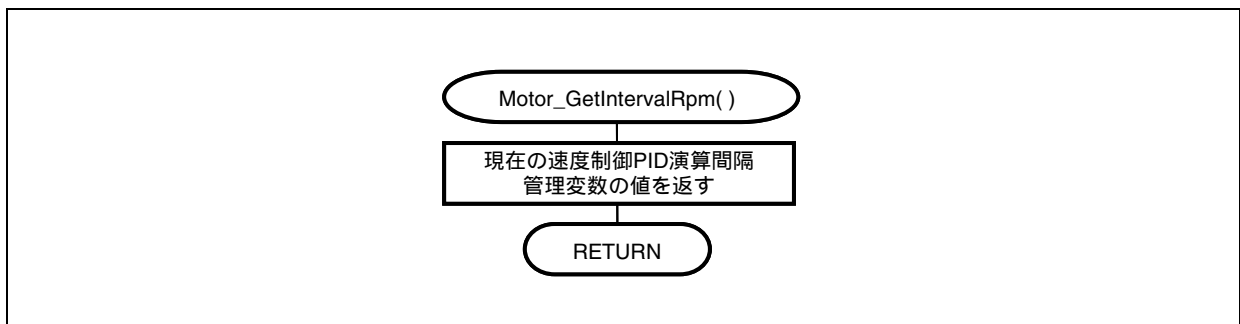
(35) 速度制御のPID演算間隔の設定処理

図4 - 36 速度制御のPID演算間隔の設定処理



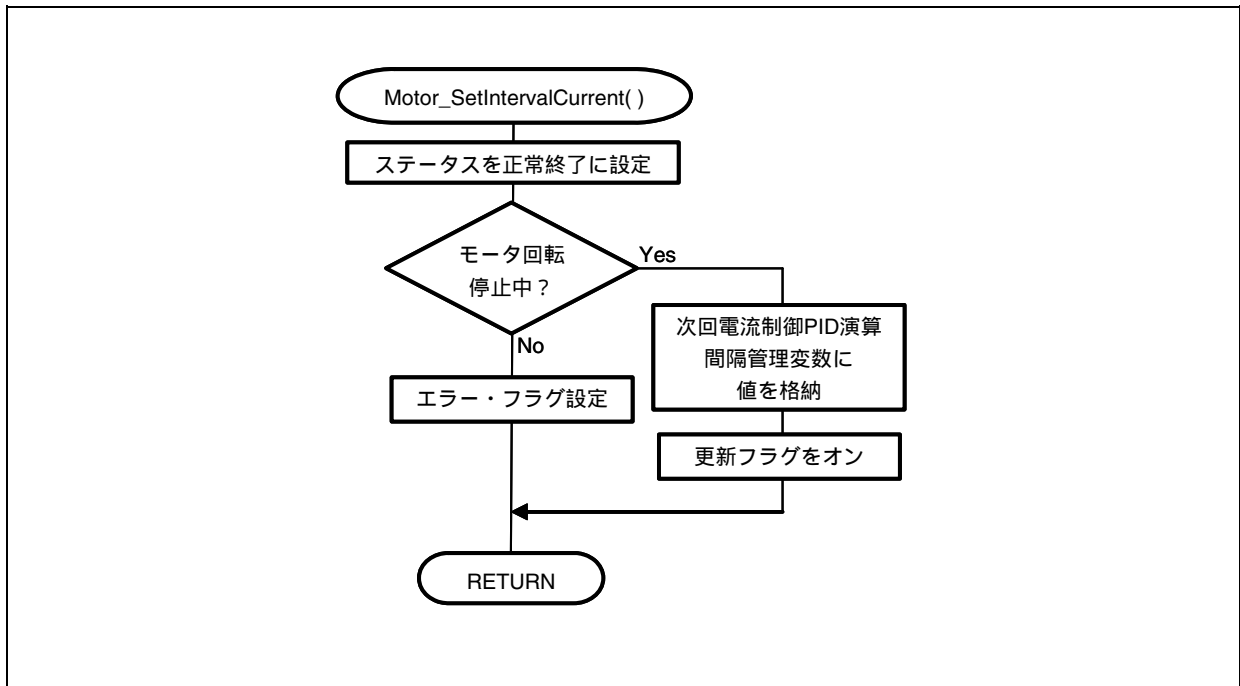
(36) 現在の速度制御のPID演算間隔の取得処理

図4 - 37 現在の速度制御のPID演算間隔の取得処理



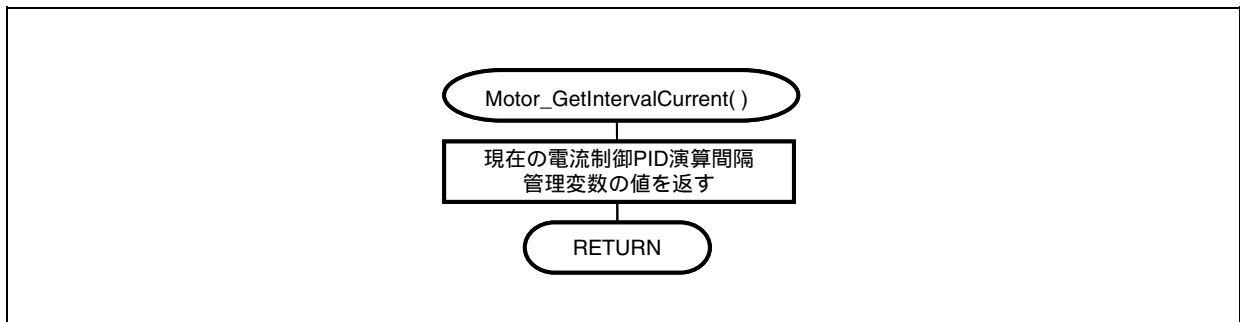
(37) 電流制御のPID演算間隔の設定処理

図4 - 38 電流制御のPID演算間隔の設定処理



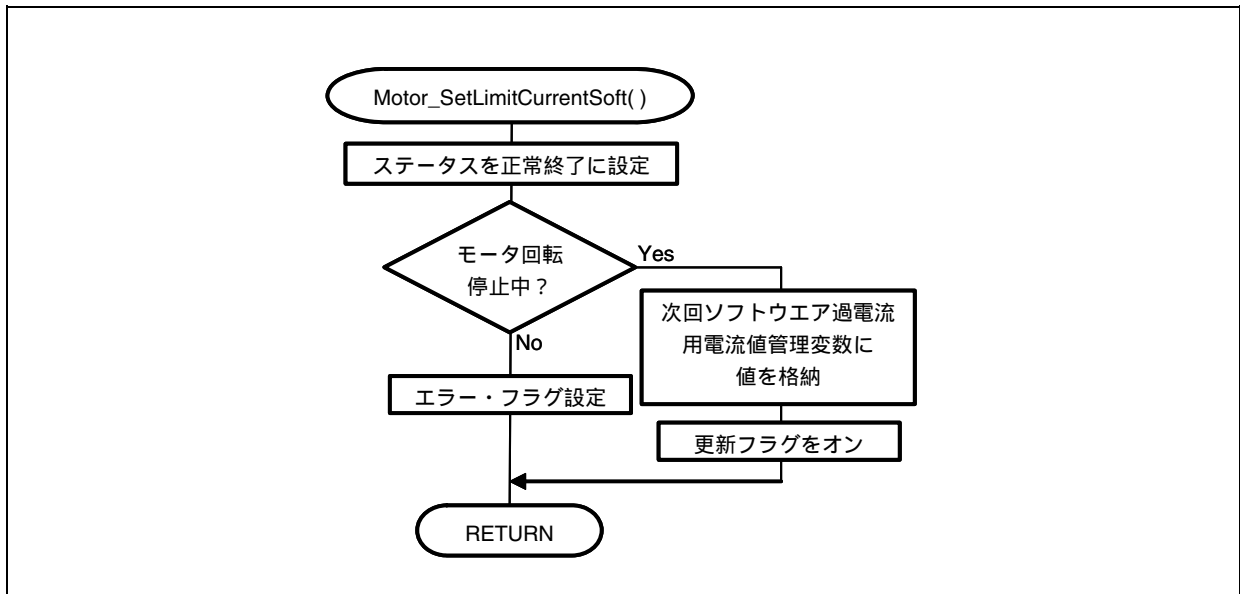
(38) 現在の電流制御のPID演算間隔の取得処理

図4 - 39 現在の電流制御のPID演算間隔の取得処理



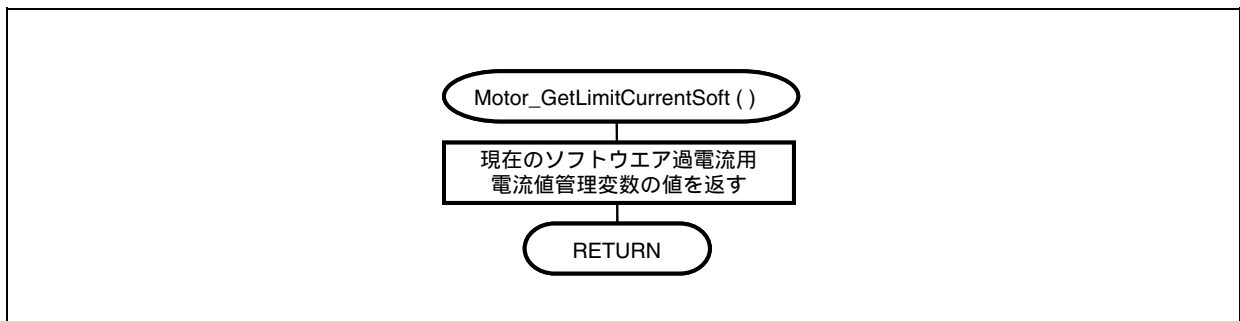
(39) ソフトウェア過電流検知のリミットの電流値設定処理

図4 - 40 ソフトウェア過電流検知のリミットの電流値設定処理



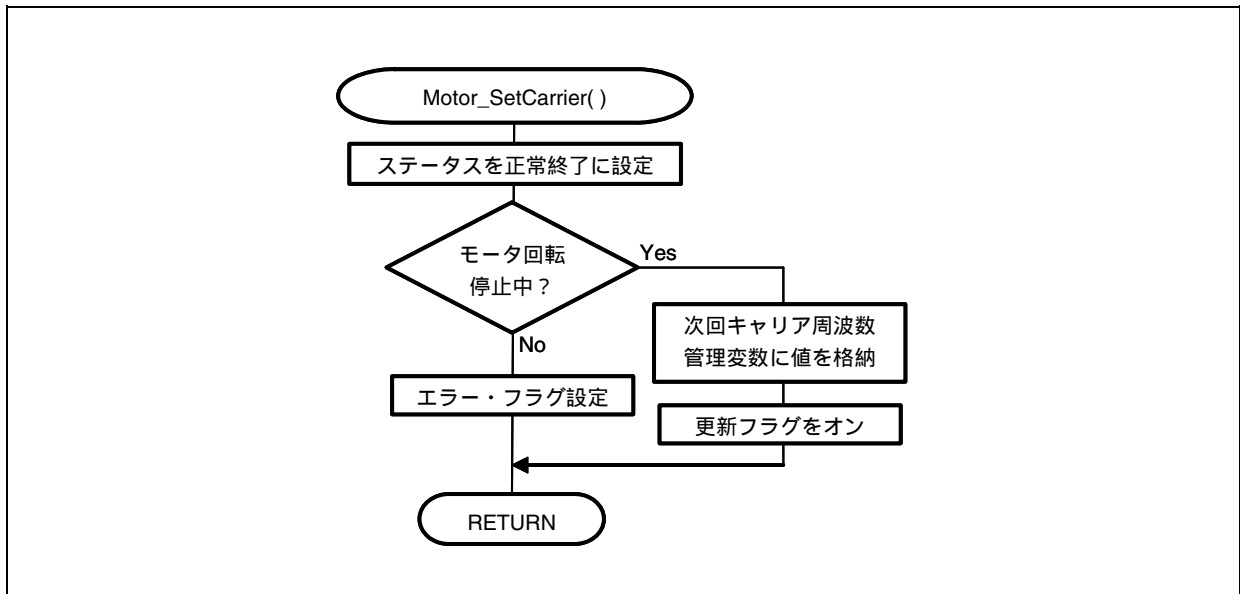
(40) 現在のソフトウェア過電流検知のリミットの電流値取得処理

図4 - 41 現在のソフトウェア過電流検知のリミットの電流値取得処理



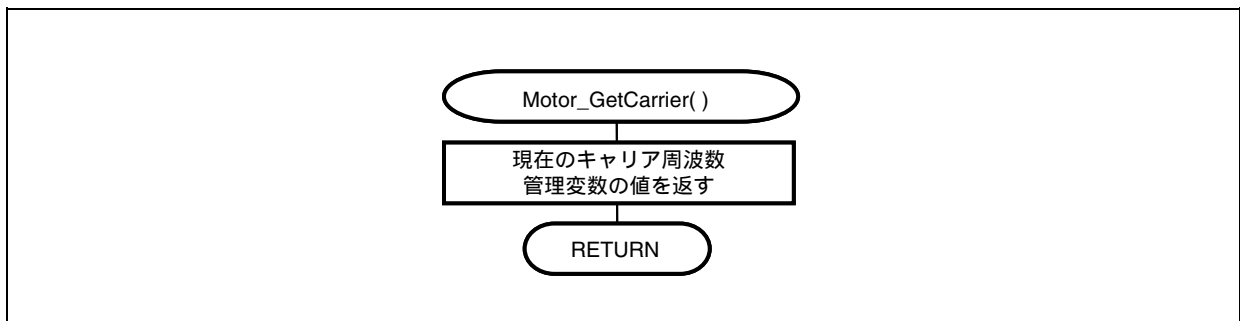
(41) キャリア周波数設定処理

図4 - 42 キャリア周波数設定処理



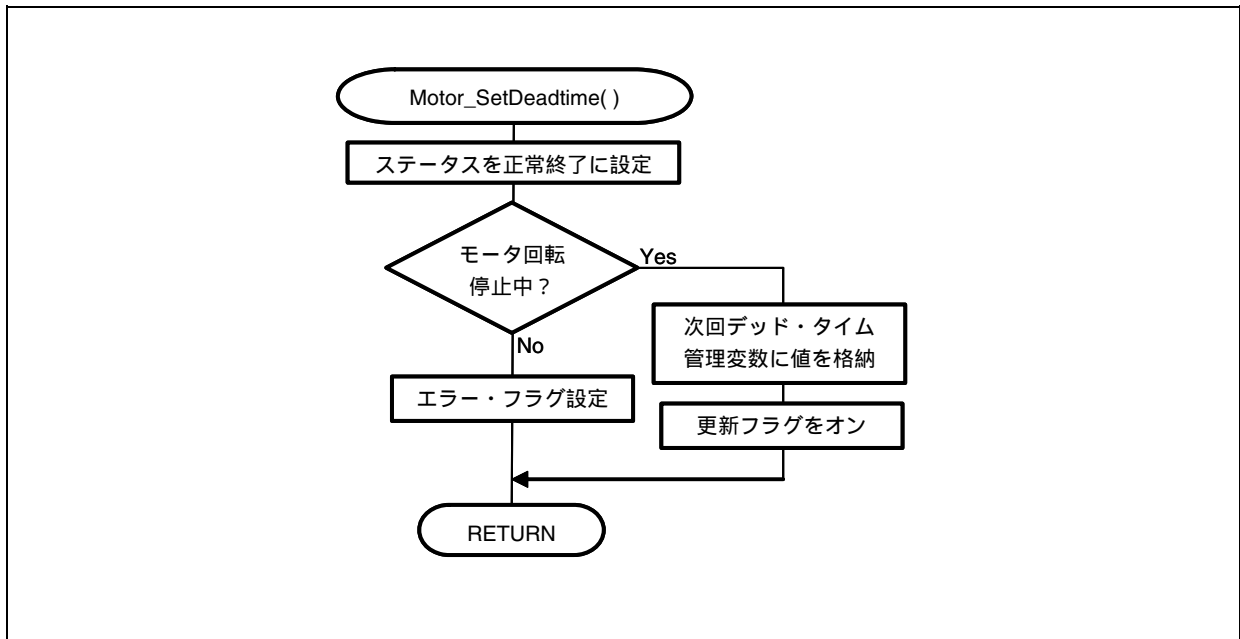
(42) 現在のキャリア周波数取得処理

図4 - 43 現在のキャリア周波数取得処理



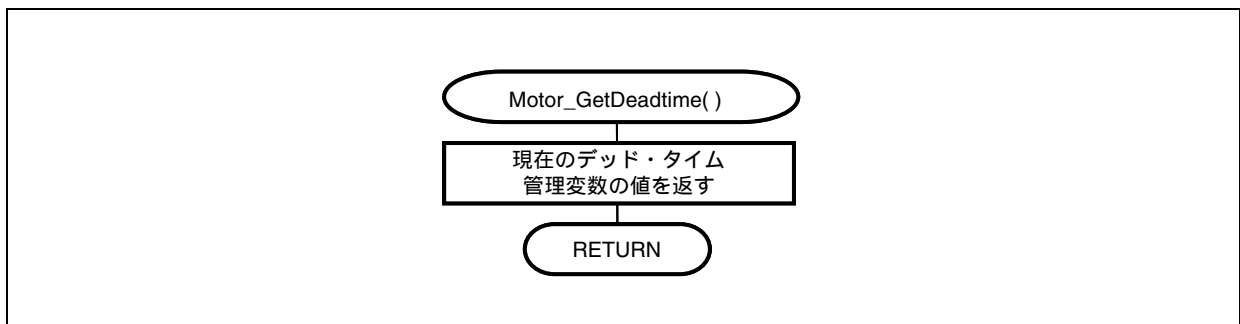
(43) デッド・タイム設定処理

図4 - 44 デッド・タイム設定処理



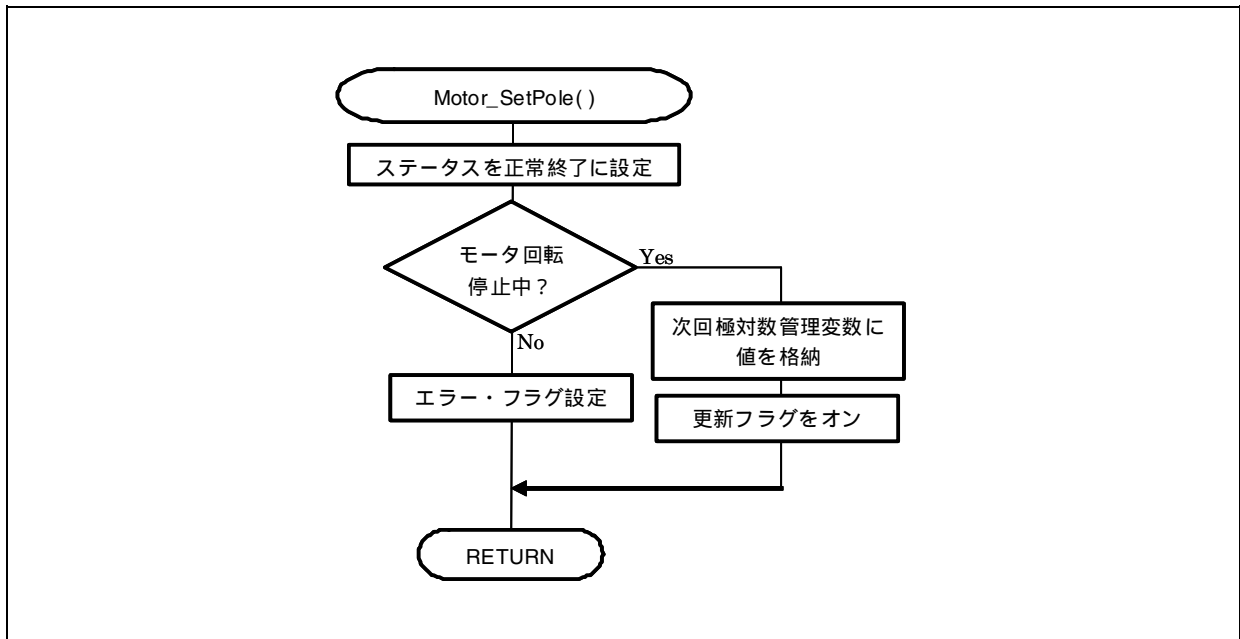
(44) 現在のデッド・タイム取得処理

図4 - 45 現在のデッド・タイム取得処理



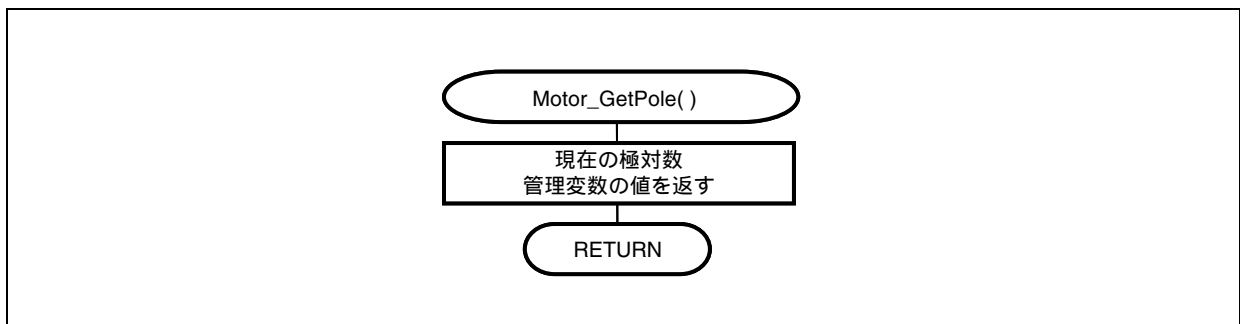
(45) 極対数設定処理

図4 - 46 極対数設定処理



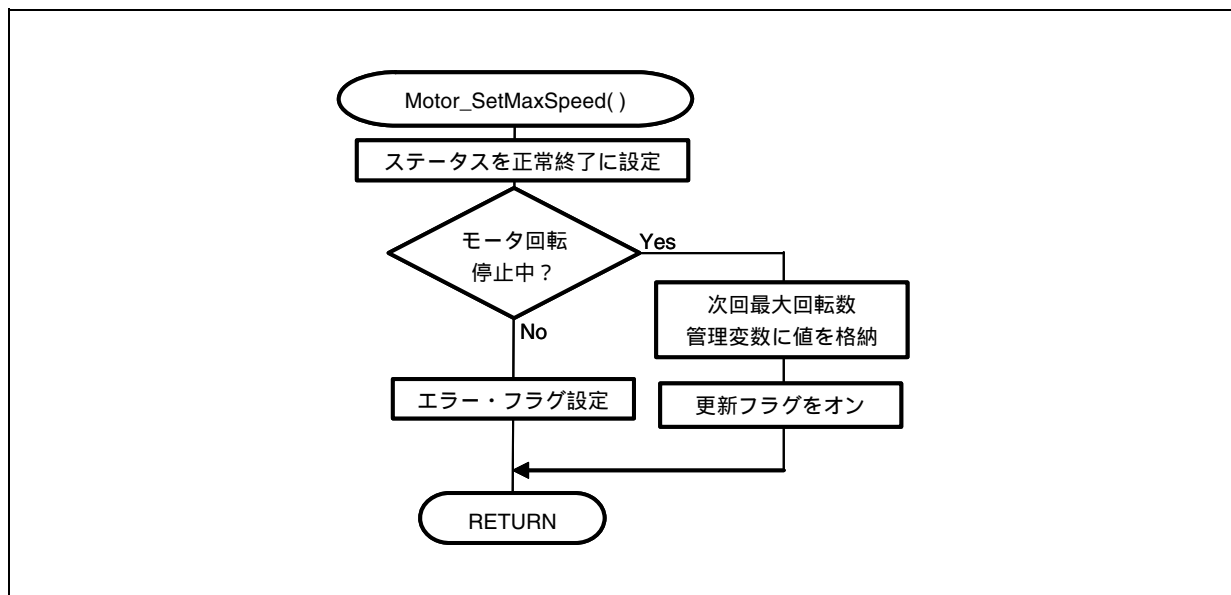
(46) 現在の極対数取得処理

図4 - 47 現在の極対数取得処理



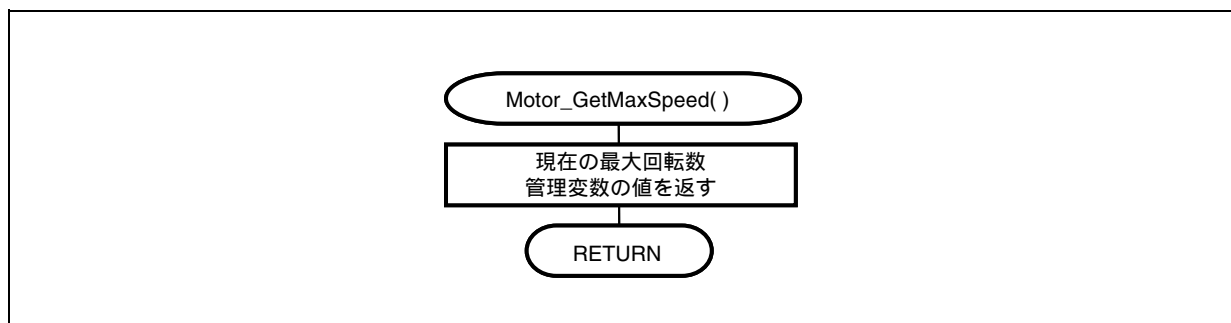
(47) 最大回転数設定処理

図4 - 48 最大回転数設定処理



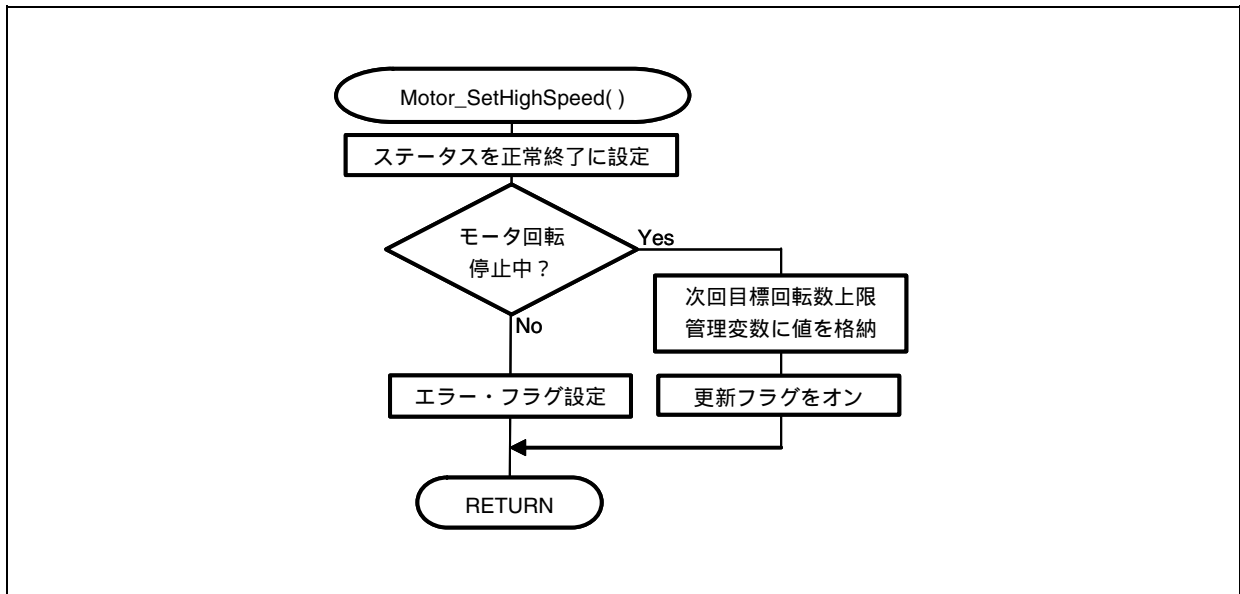
(48) 現在の最大回転数取得処理

図4 - 49 現在の最大回転数取得処理



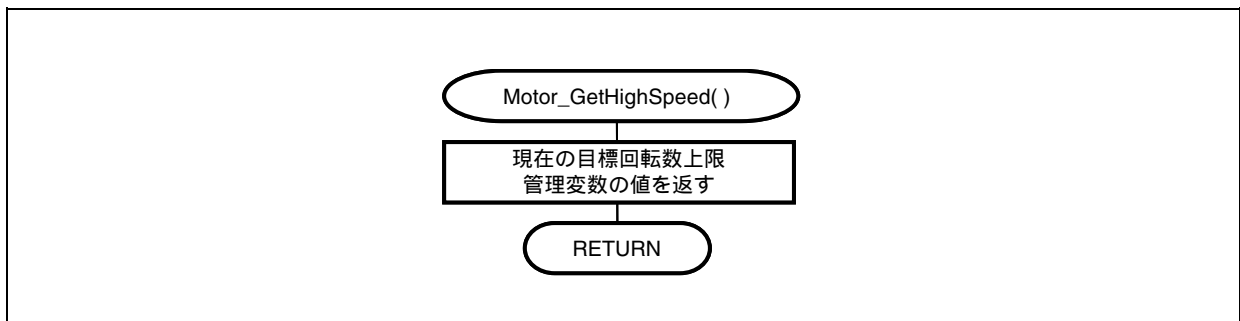
(49) 目標回転数の上限値設定処理

図4 - 50 目標回転数の上限値設定処理



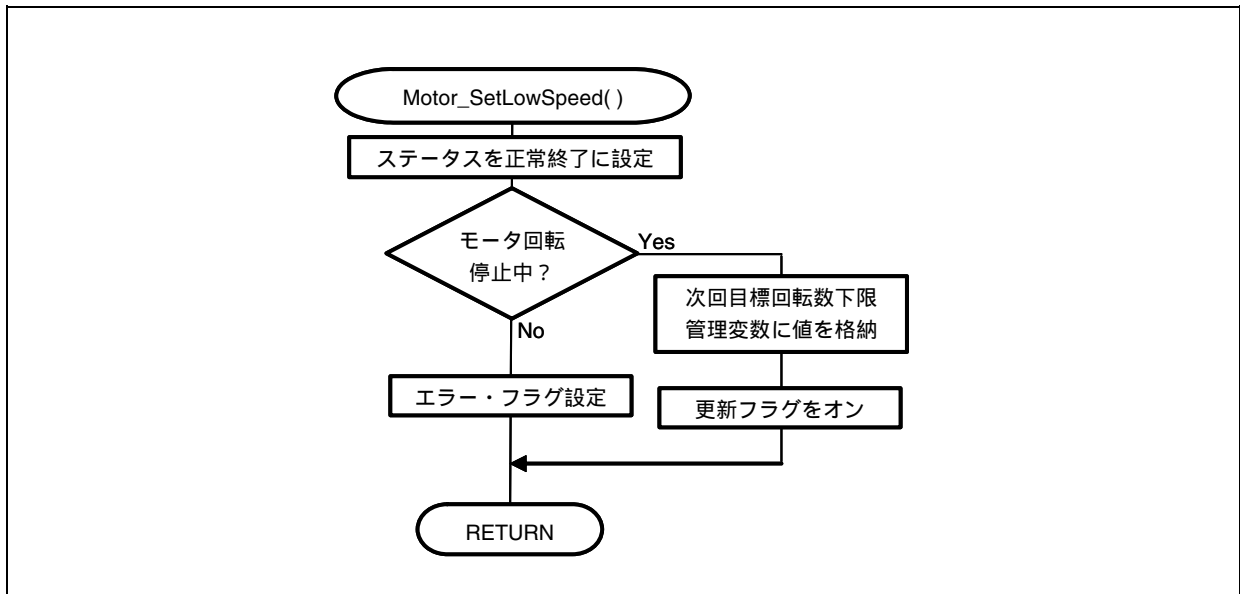
(50) 現在の目標回転数の上限値取得処理

図4 - 51 現在の目標回転数の上限値取得処理



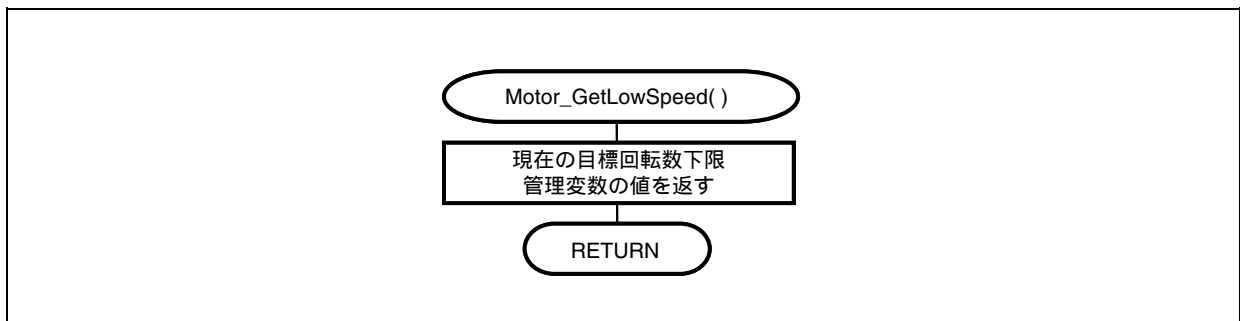
(51) 目標回転数の下限値設定処理

図4 - 52 目標回転数の下限値設定処理



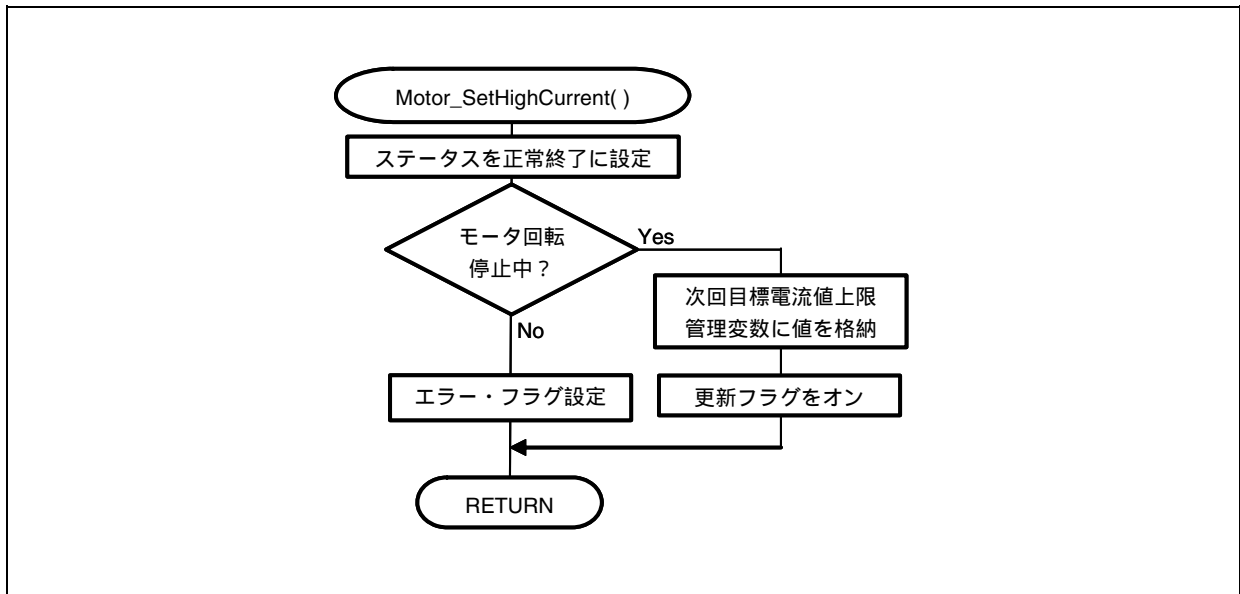
(52) 現在の目標回転数の下限値取得処理

図4 - 53 現在の目標回転数の下限値取得処理



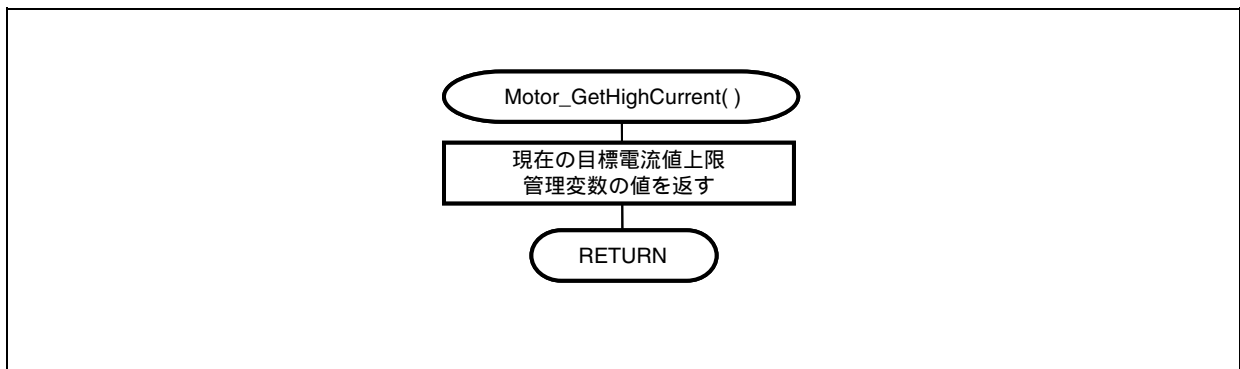
(53) 目標電流値の上限値設定処理

図4 - 54 目標電流値の上限値設定処理



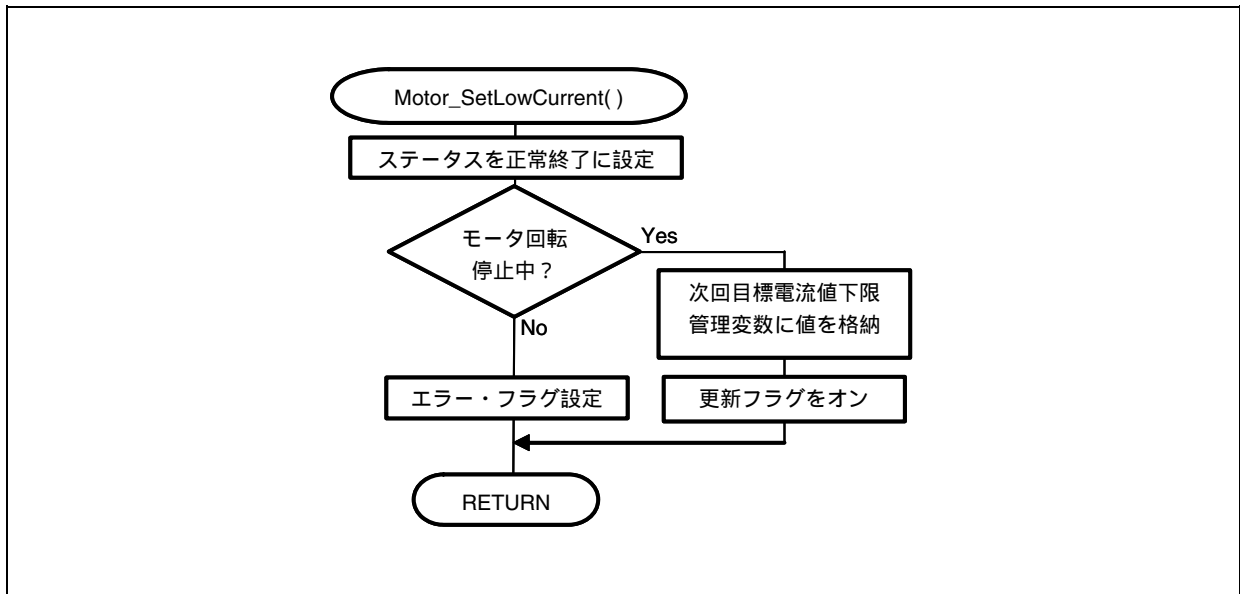
(54) 現在の目標電流値の上限値取得処理

図4 - 55 現在の目標電流値の上限値取得処理



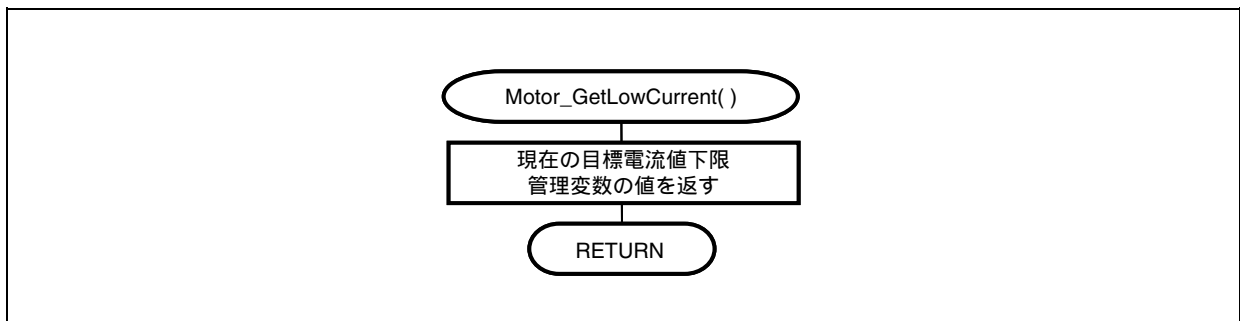
(55) 目標電流値の下限値設定処理

図4 - 56 目標電流値の下限値設定処理



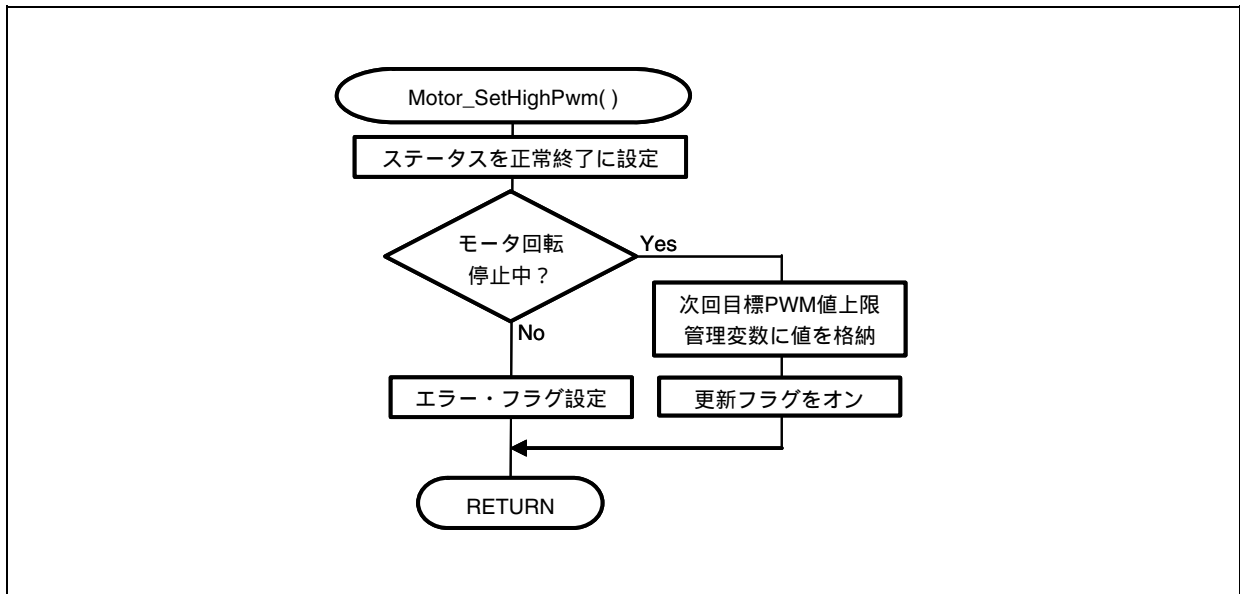
(56) 現在の目標電流値の下限値取得処理

図4 - 57 現在の目標電流値の下限値取得処理



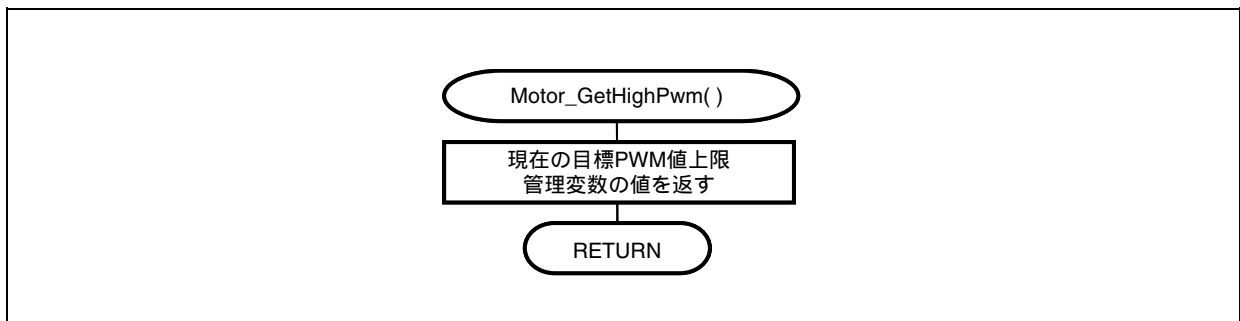
(57) 目標PWM値の上限値設定処理

図4 - 58 目標PWM値の上限値設定処理



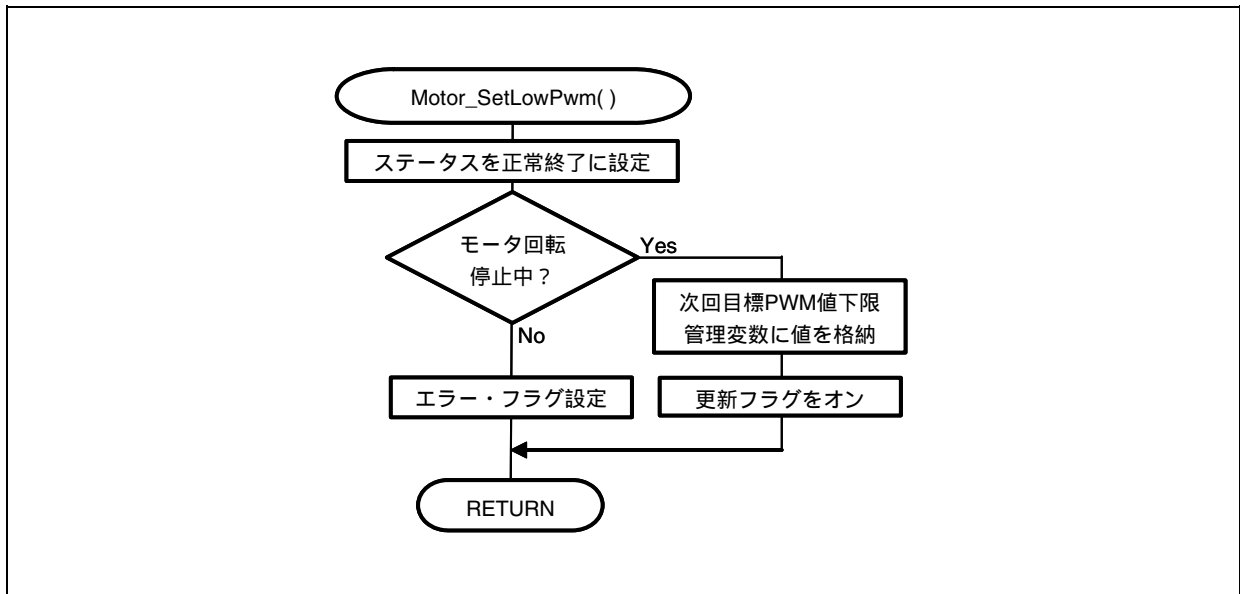
(58) 現在の目標PWM値の上限値取得処理

図4 - 59 現在の目標PWM値の上限値取得処理



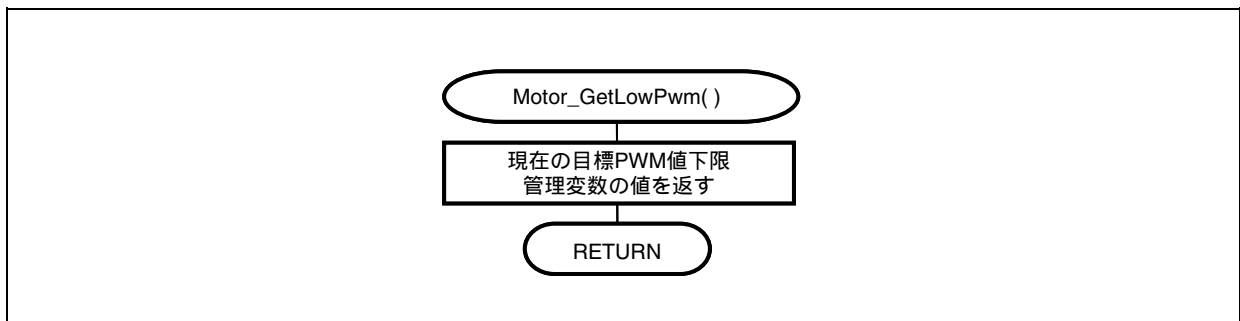
(59) 目標PWM値の下限値設定処理

図4 - 60 目標PWM値の下限値設定処理



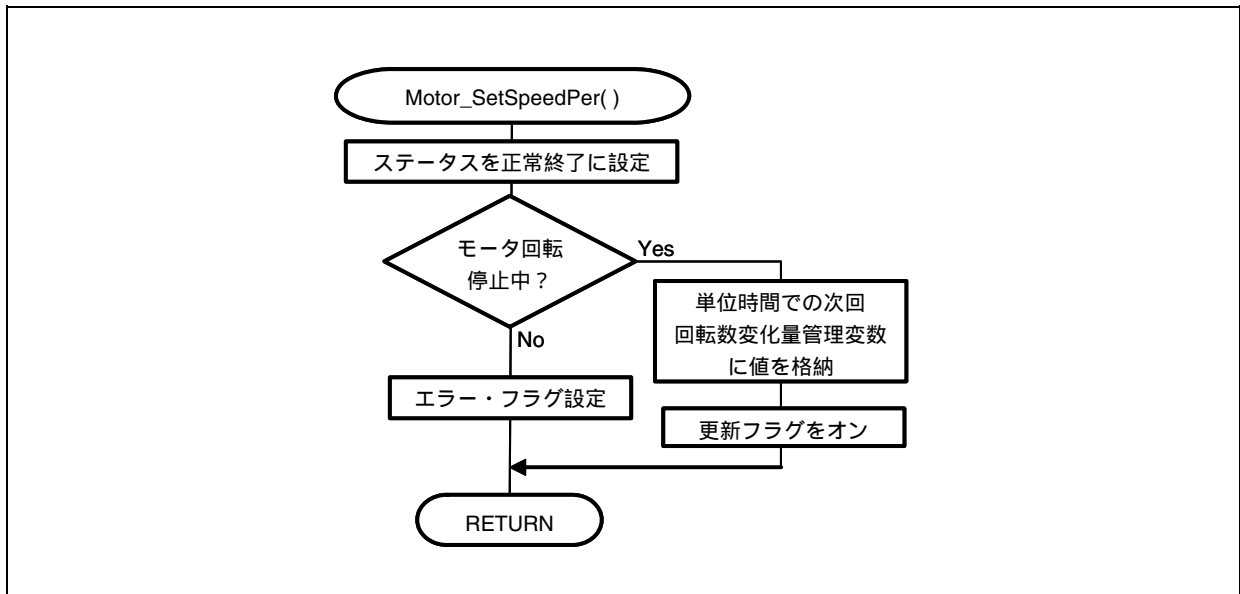
(60) 現在の目標PWM値の下限値取得処理

図4 - 61 現在の目標PWM値の下限値取得処理



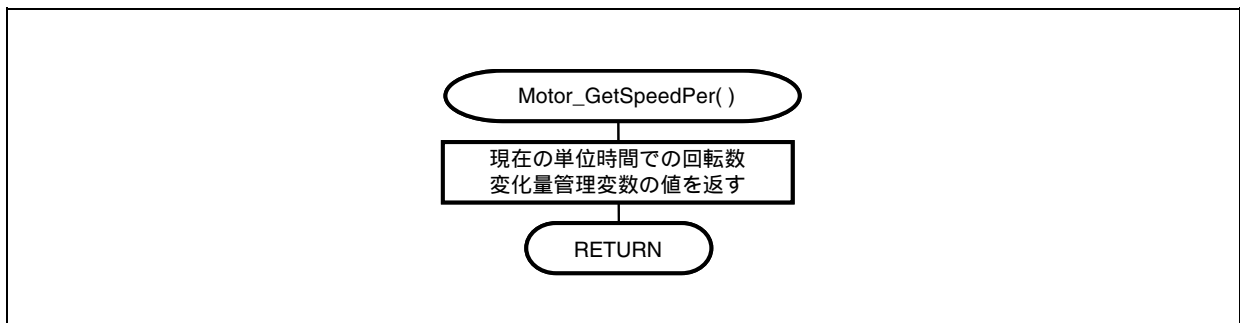
(61) 単位時間での回転数の変化量設定処理

図4 - 62 単位時間での回転数の変化量設定処理



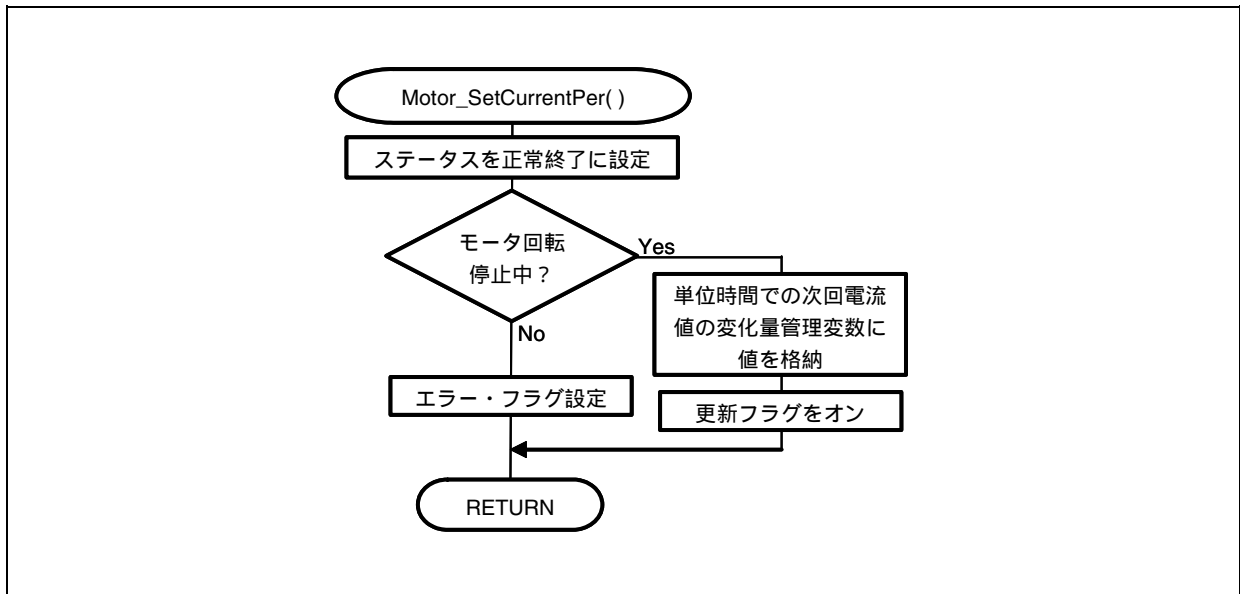
(62) 現在の単位時間での回転数の変化量取得処理

図4 - 63 現在の単位時間での回転数の変化量取得処理



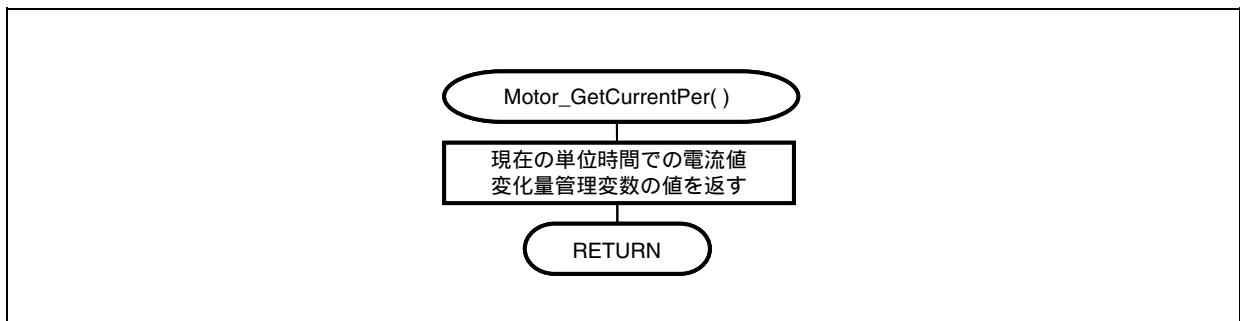
(63) 単位時間での電流値の変化量設定処理

図4 - 64 単位時間での電流値の変化量設定処理



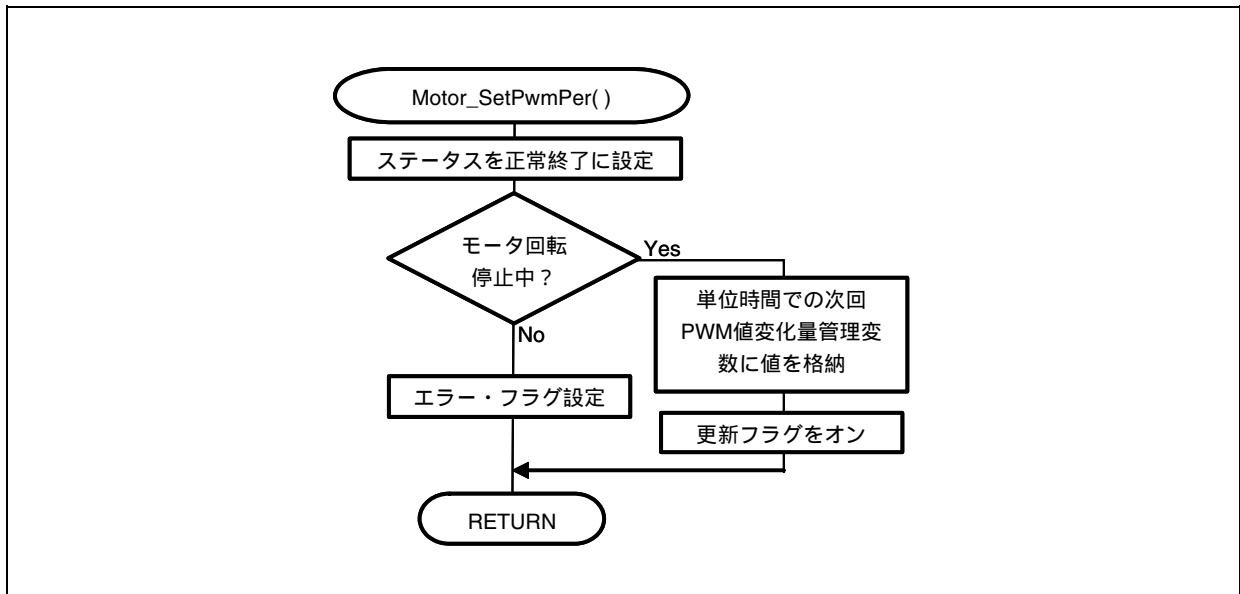
(64) 現在の単位時間での電流値の変化量取得処理

図4 - 65 現在の単位時間での電流値の変化量取得処理



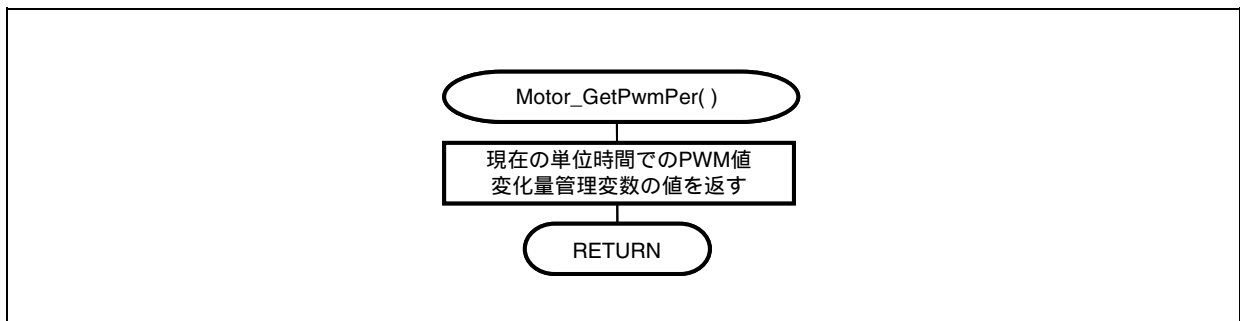
(65) 単位時間でのPWM値の変化量設定処理

図4 - 66 単位時間でのPWM値の変化量設定処理



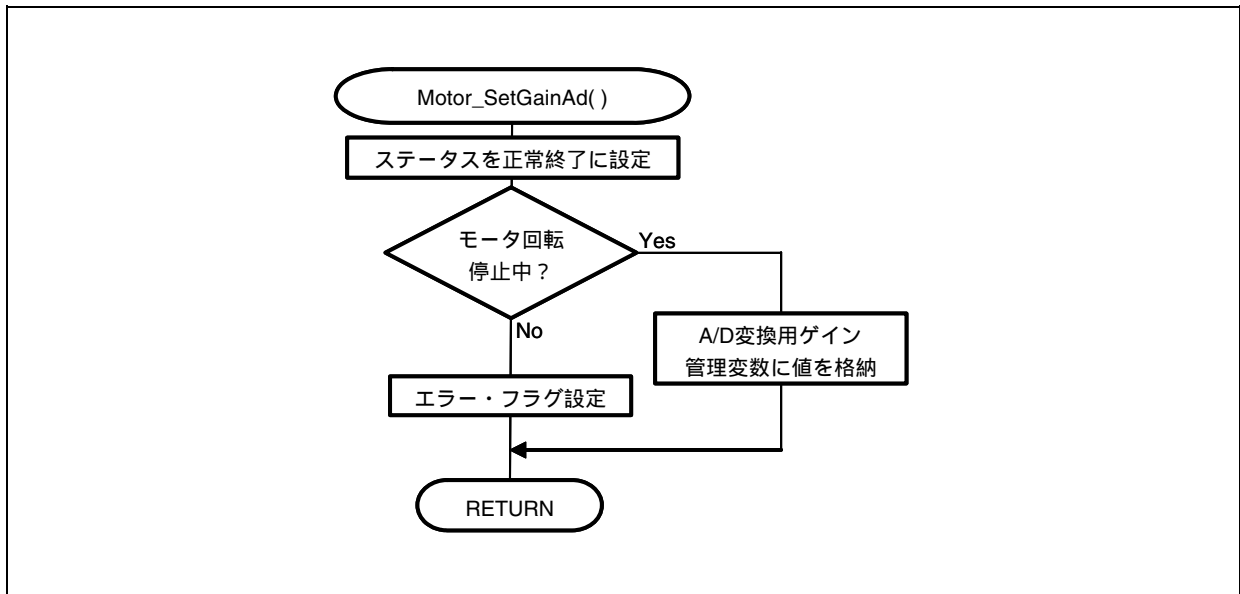
(66) 現在の単位時間でのPWM値の変化量取得処理

図4 - 67 現在の単位時間でのPWM値の変化量取得処理



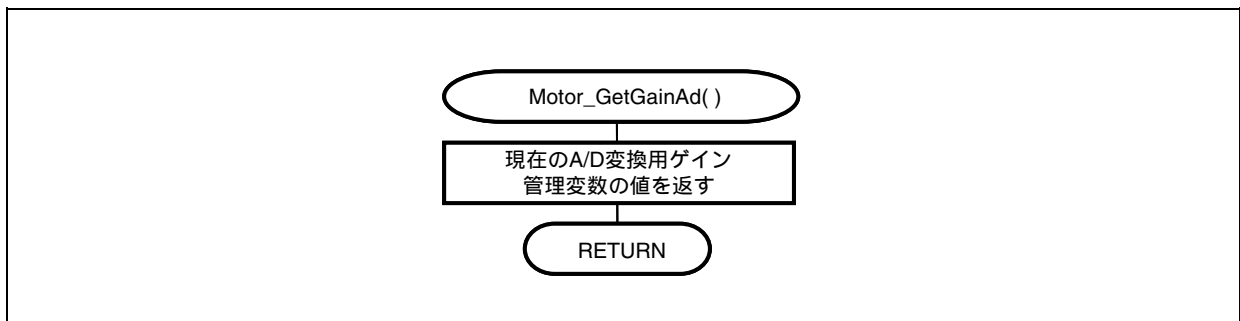
(67) A/D変換値を電流値へ換算する際のゲイン係数設定処理

図4 - 68 A/D変換値を電流値へ換算する際のゲイン係数設定処理



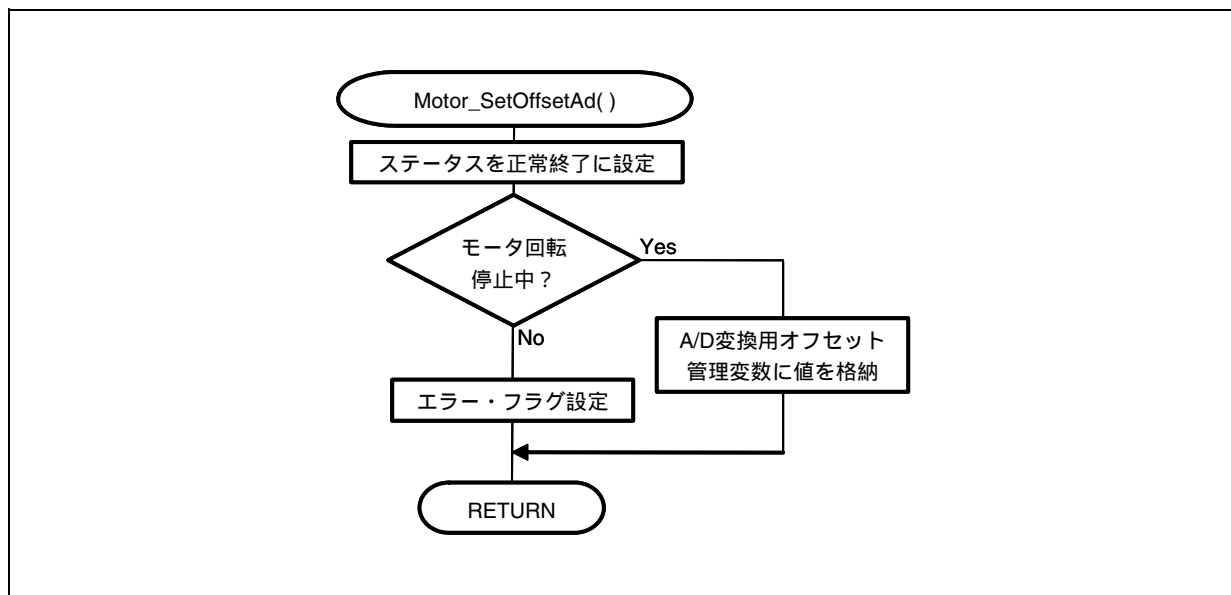
(68) 現在のA/D変換値を電流値へ換算する際のゲイン係数取得処理

図4 - 69 現在のA/D変換値を電流値へ換算する際のゲイン係数取得処理



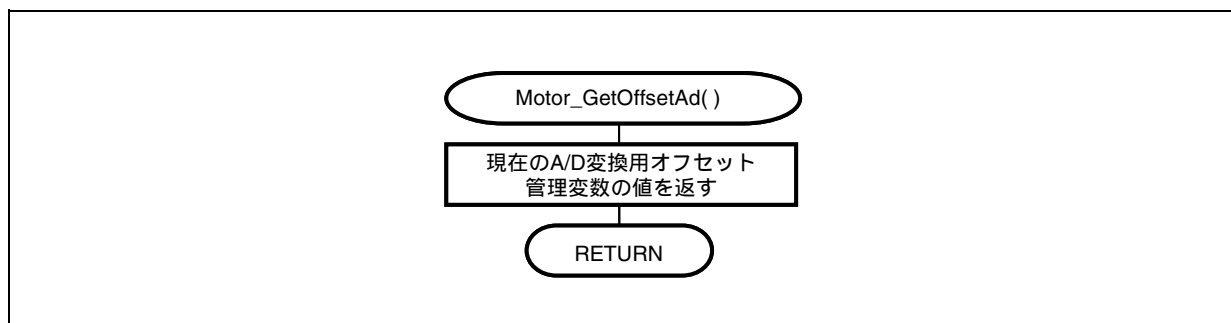
(69) A/D変換値を電流値へ換算する際のオフセット設定処理

図4 - 70 A/D変換値を電流値へ換算する際のオフセット設定処理



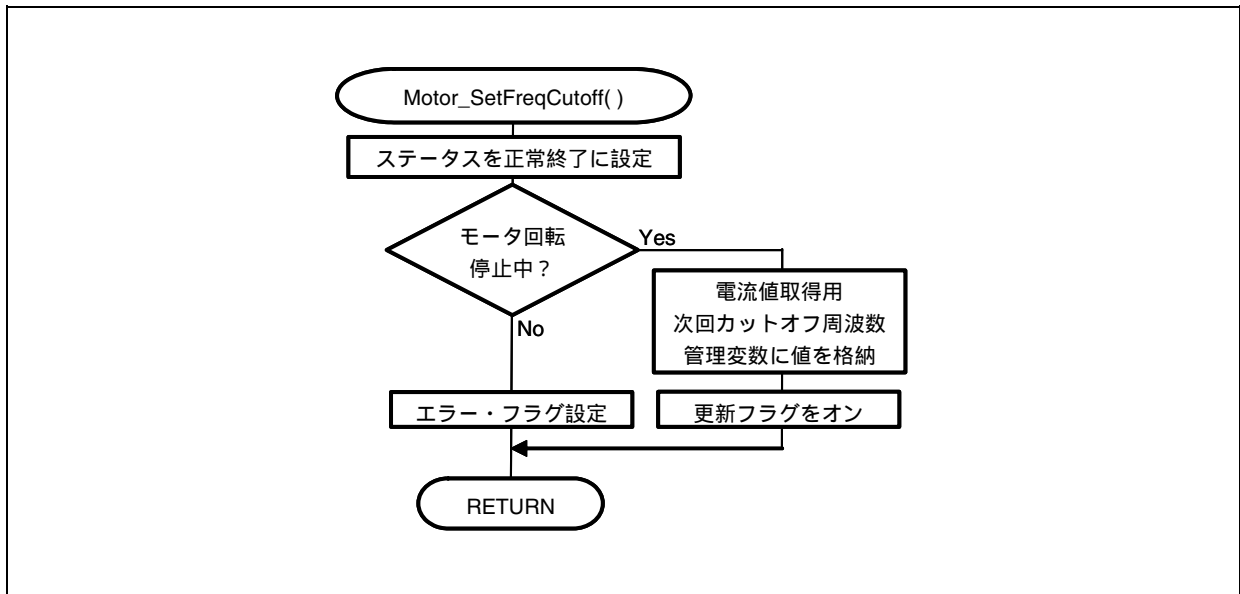
(70) 現在のA/D変換値を電流値へ換算する際のオフセット取得処理

図4 - 71 現在のA/D変換値を電流値へ換算する際のオフセット取得処理



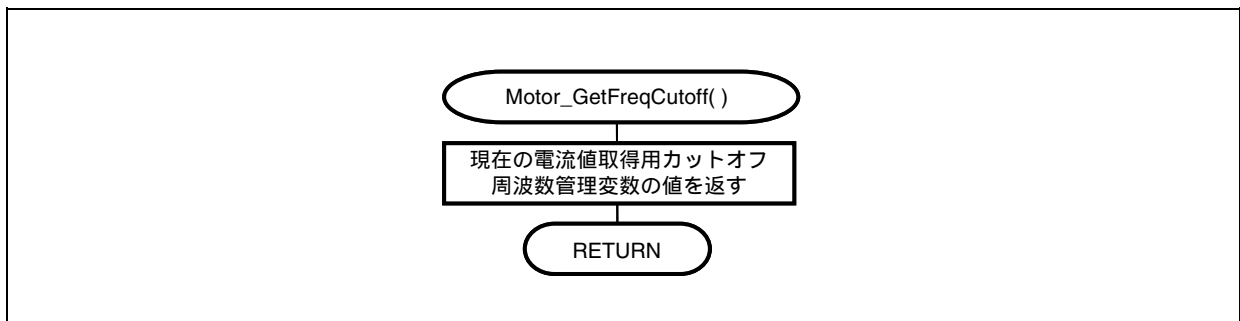
(71) 電流値取得用カットオフ周波数設定処理

図4 - 72 電流値取得用カットオフ周波数設定処理



(72) 現在の電流値取得用カットオフ周波数取得処理

図4 - 73 現在の電流値取得用カットオフ周波数取得処理



4.9.2 内部関数のフロー・チャート

【main_mcio.cファイルのフロー・チャート】

(1) メイン処理

図4-74 メイン処理 (1/2)

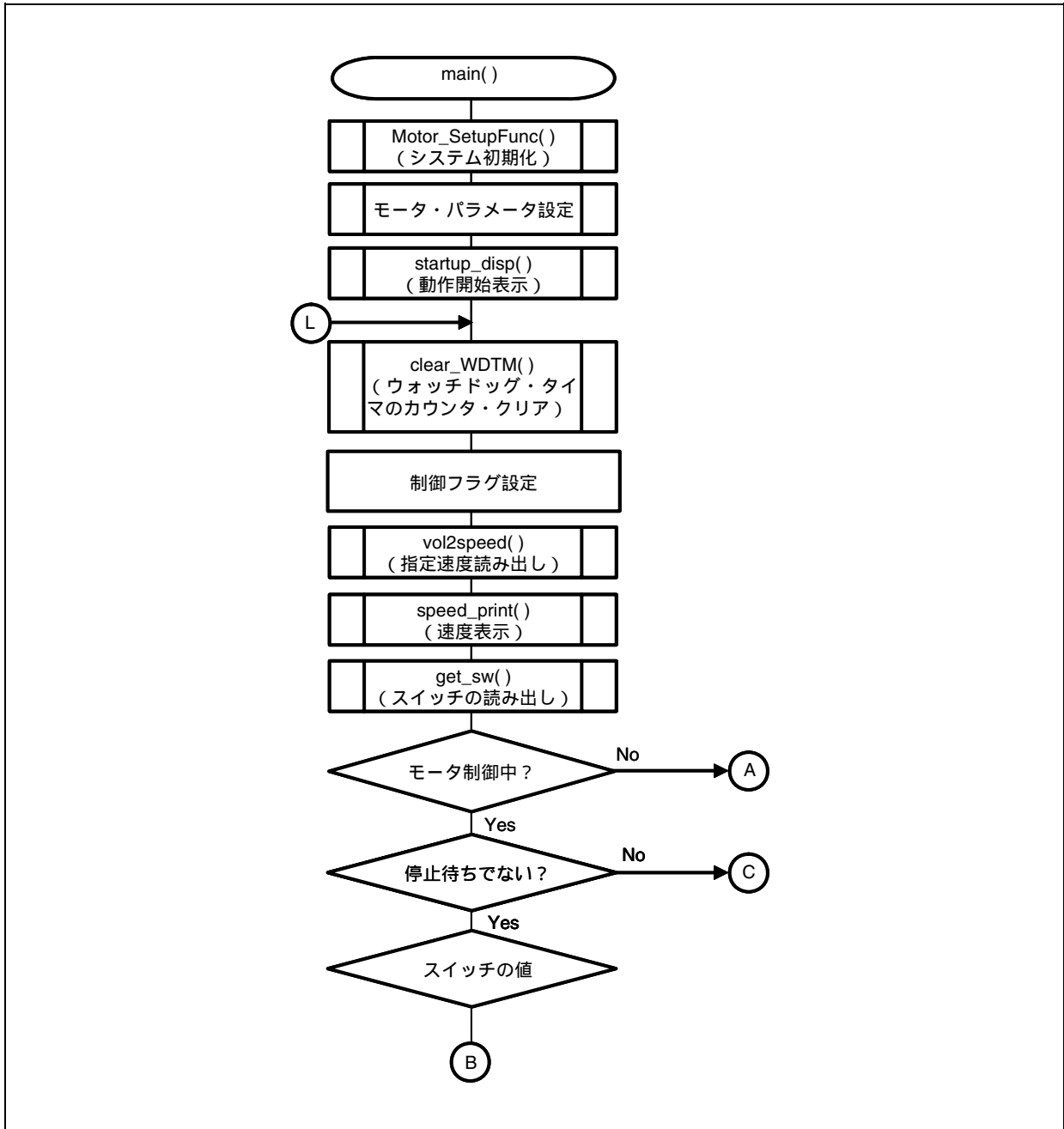
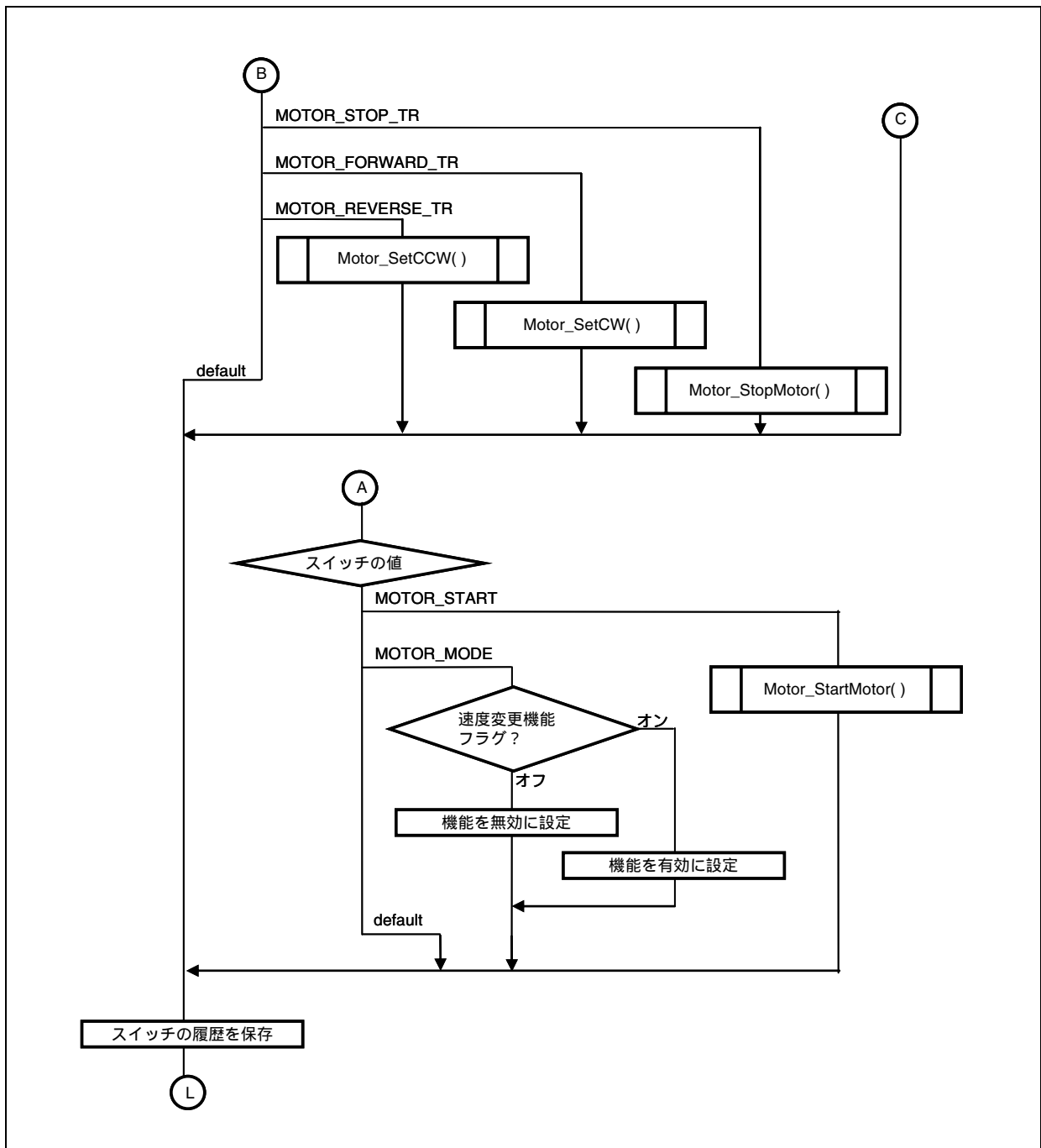


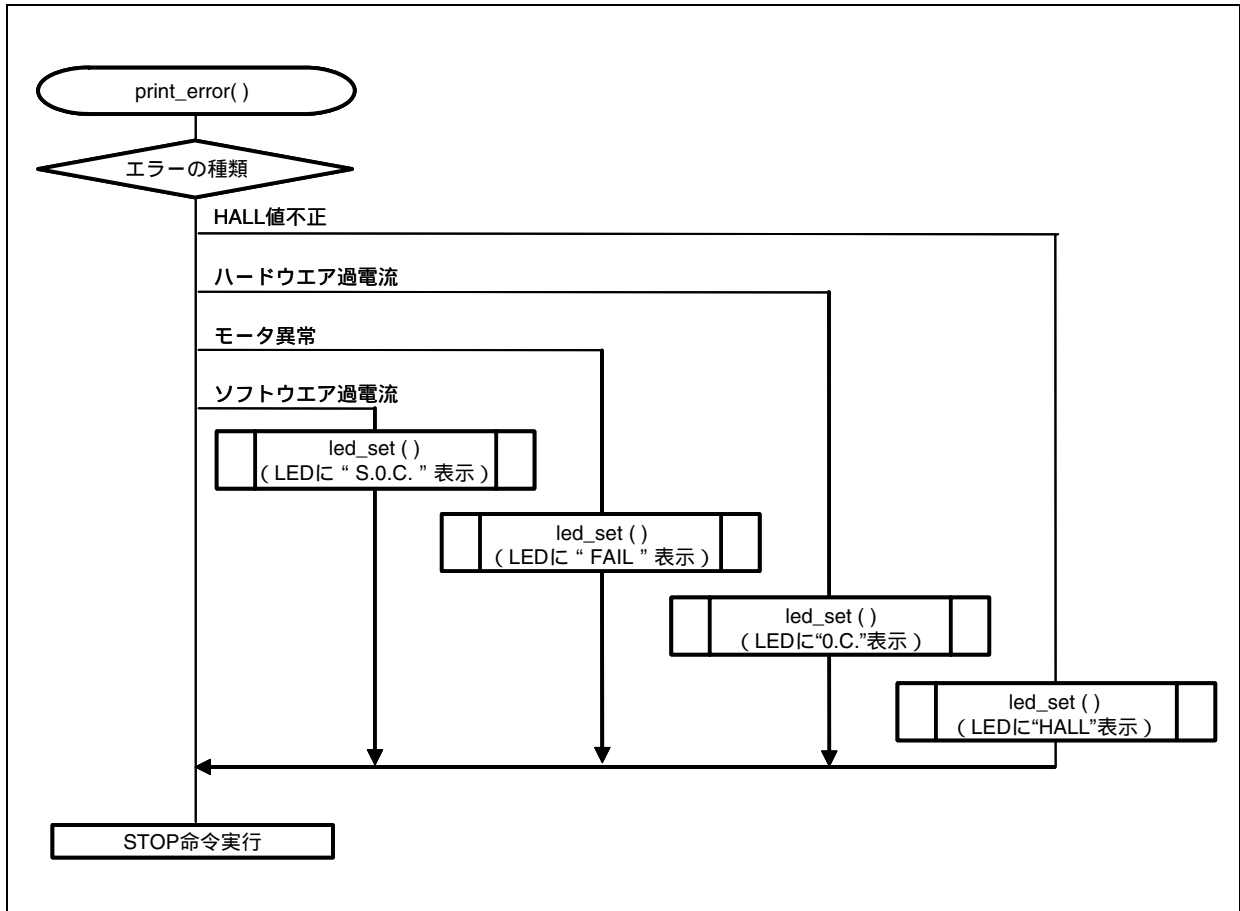
図4 - 74 メイン処理 (2/2)



【sub_mcio.cファイルのフロー・チャート】

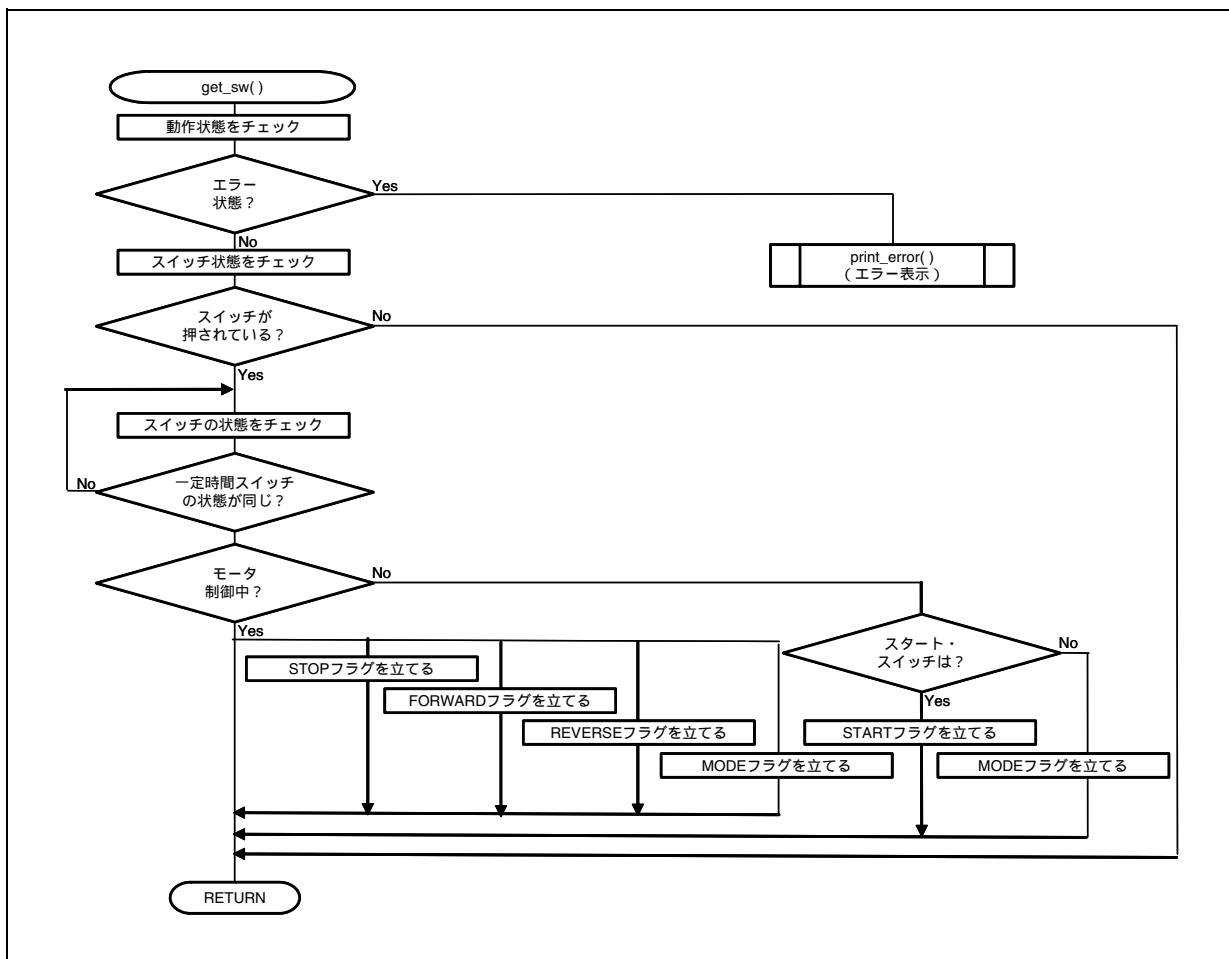
(1) エラー表示処理

図4 - 75 エラー表示処理



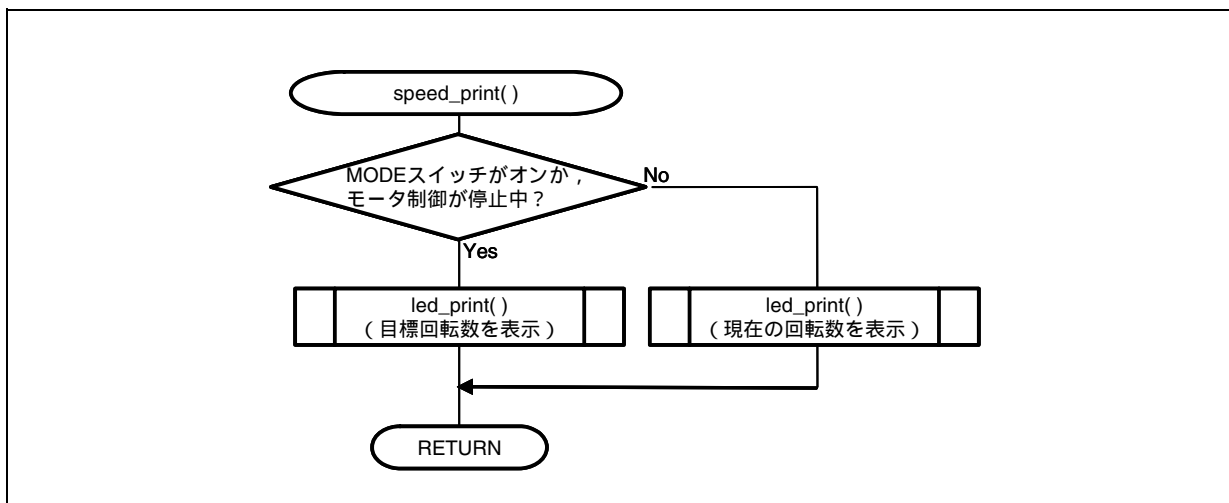
(2) スイッチ読み出し処理

図4 - 76 スイッチ読み出し処理



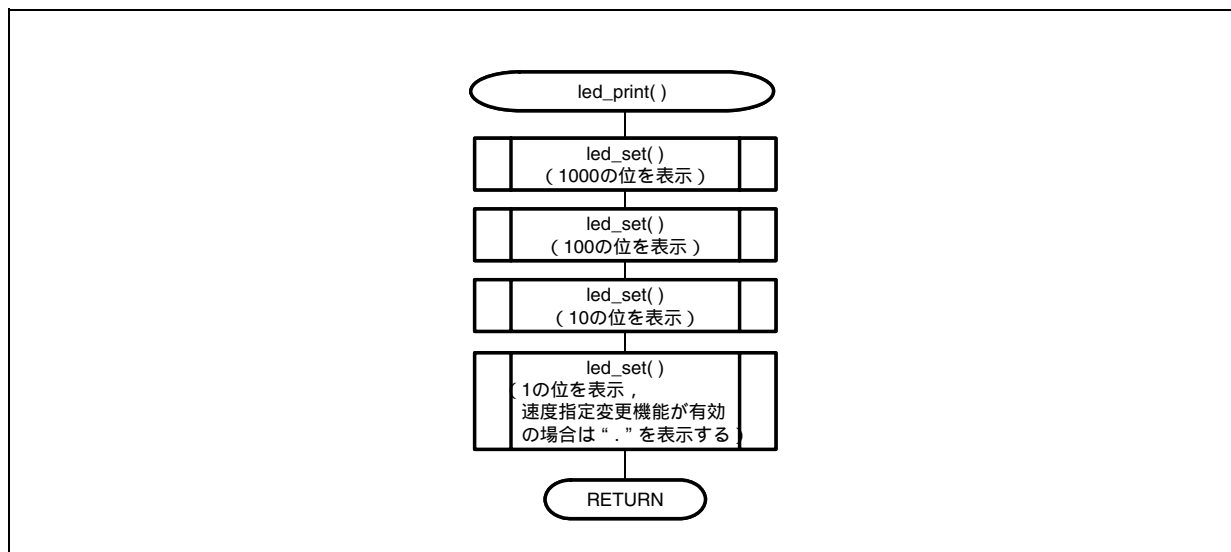
(3) LED速度表示処理

図4 - 77 LED速度表示処理



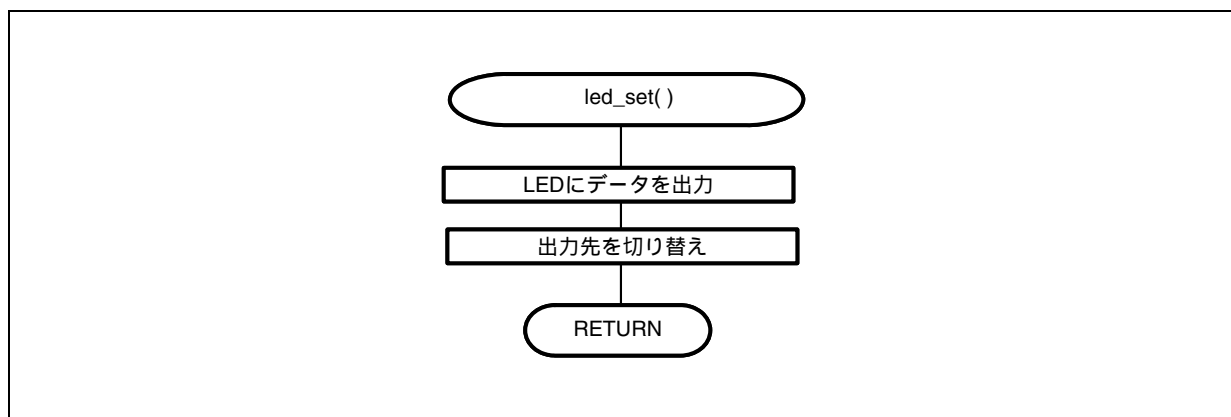
(4) LEDに1-4桁値表示処理

図4 - 78 LEDに1-4桁値表示処理



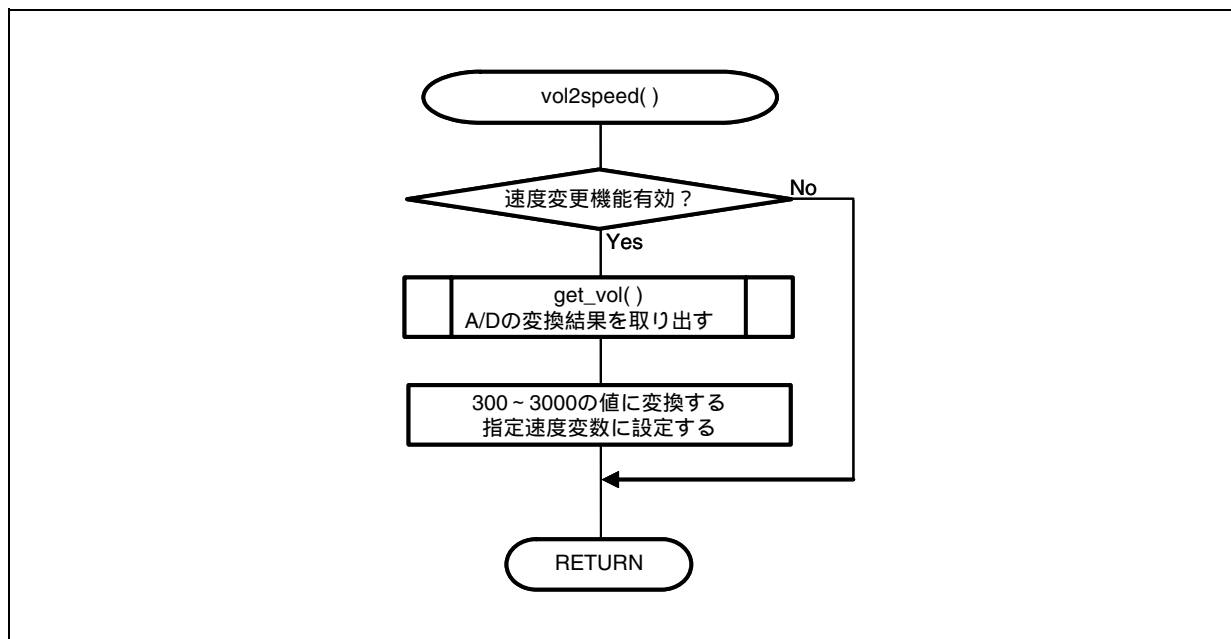
(5) 指定LEDへの値出力処理

図4 - 79 指定LEDへの値出力処理



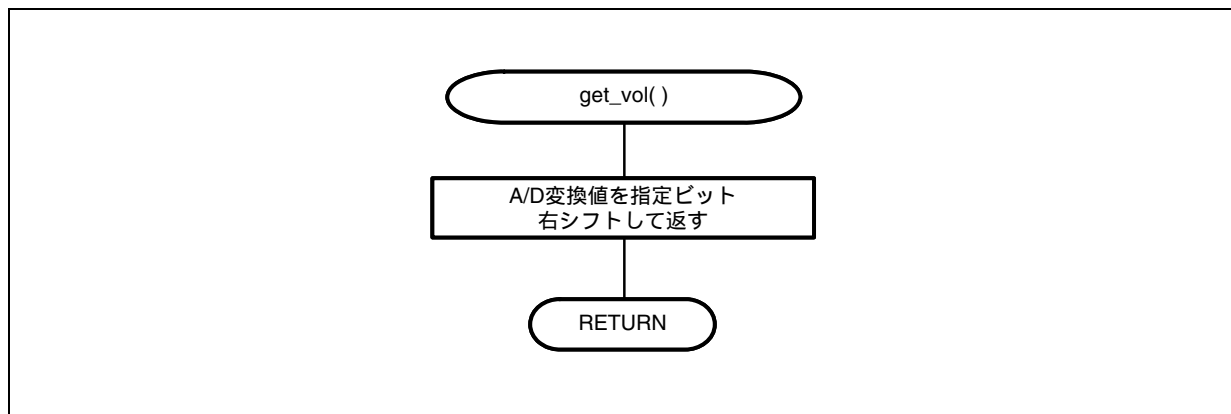
(6) 指定速度の読み出し処理

図4 - 80 指定速度の読み出し処理



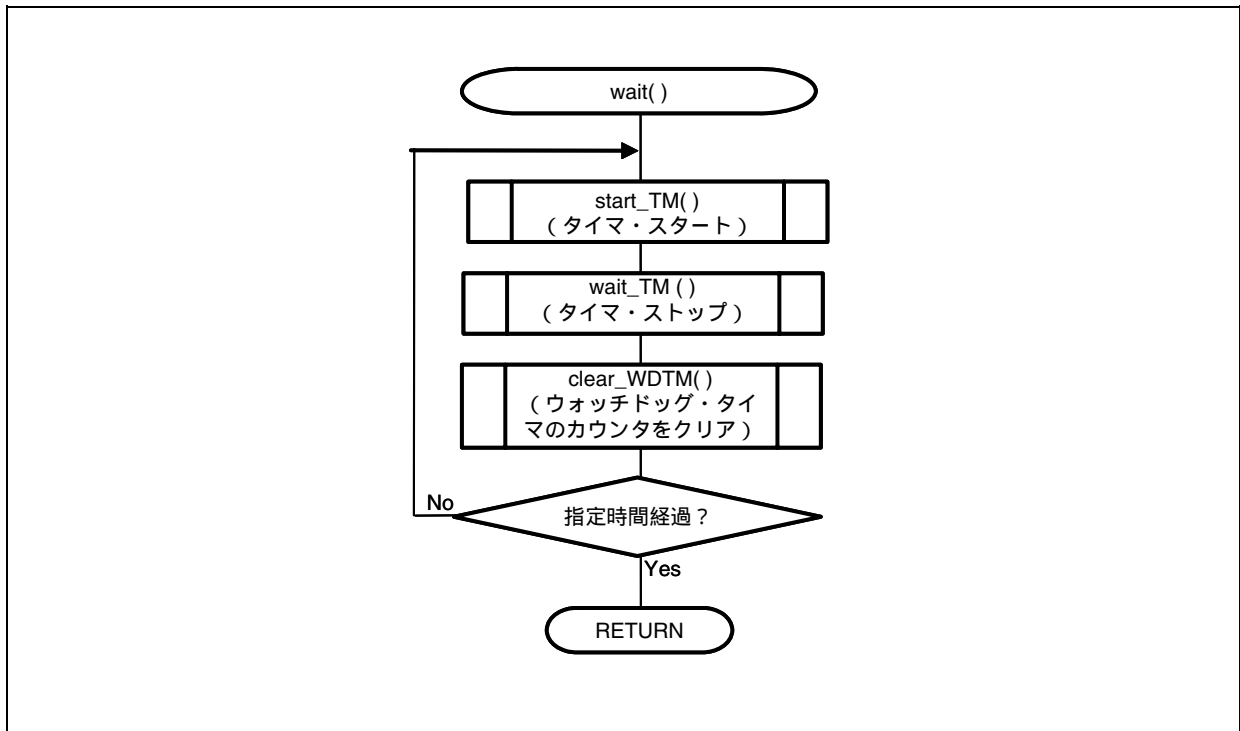
(7) A/D変換値の読み出し処理

図4 - 81 A/D変換値の読み出し処理



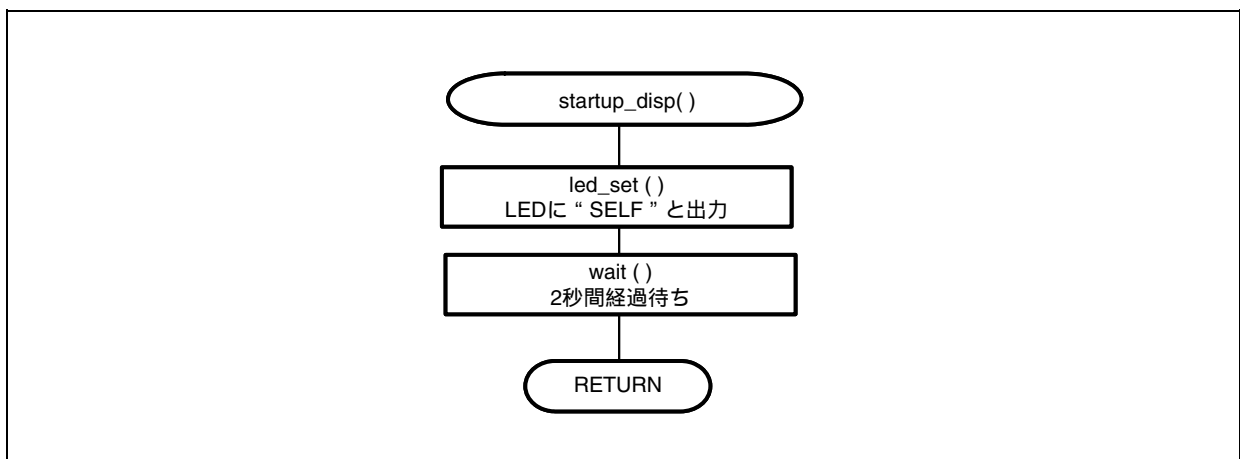
(8) 指定時間 (ms) 経過待ち処理

図4 - 82 指定時間 (ms) 経過待ち処理



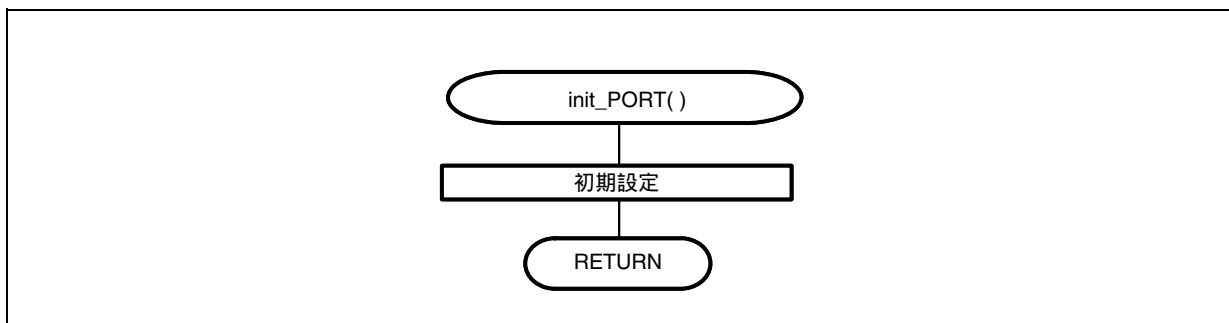
(9) 始動表示処理

図4 - 83 始動表示処理



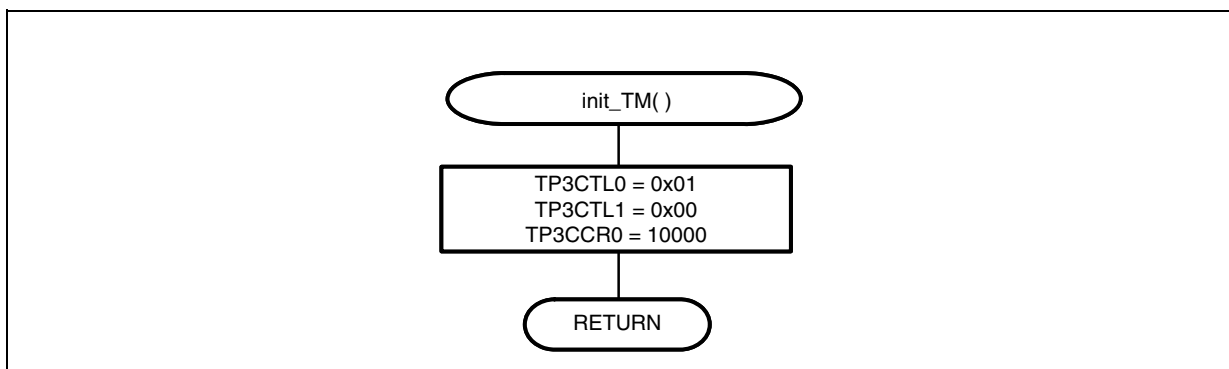
(10) ポート初期設定処理

図4 - 84 ポート初期設定処理



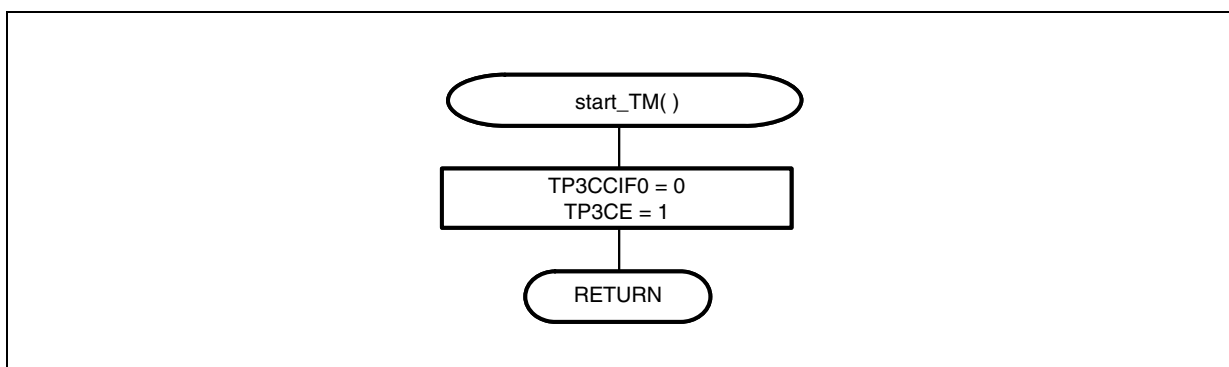
(11) タイマ初期設定処理

図4 - 85 タイマ初期設定処理



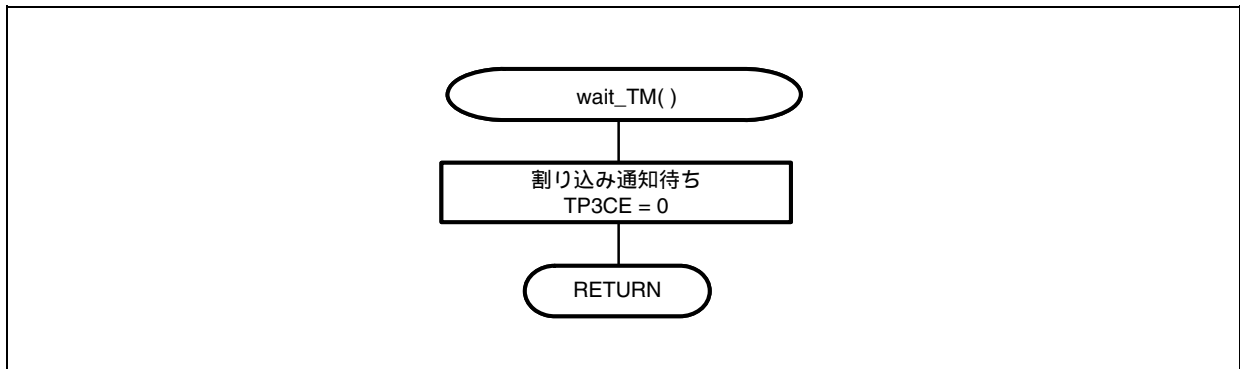
(12) タイマ開始処理

図4 - 86 タイマ開始処理



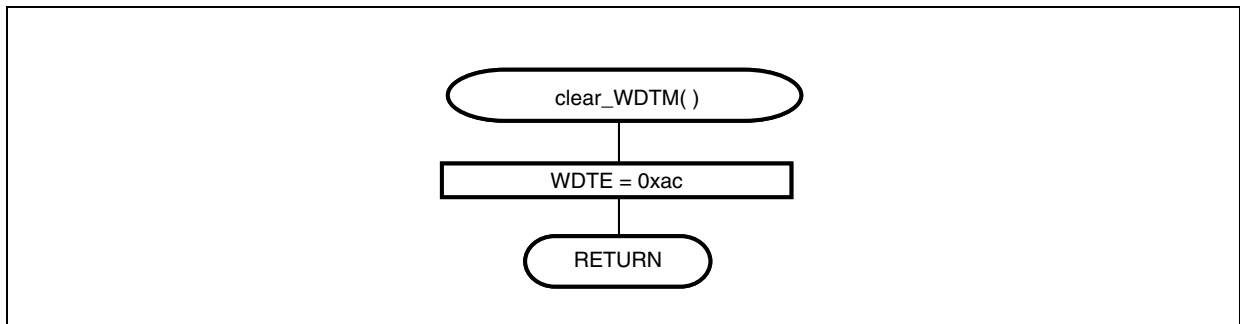
(13) タイマ停止処理

図4 - 87 タイマ停止処理



(14) ウォッチドッグ・タイマ・カウント・クリア処理

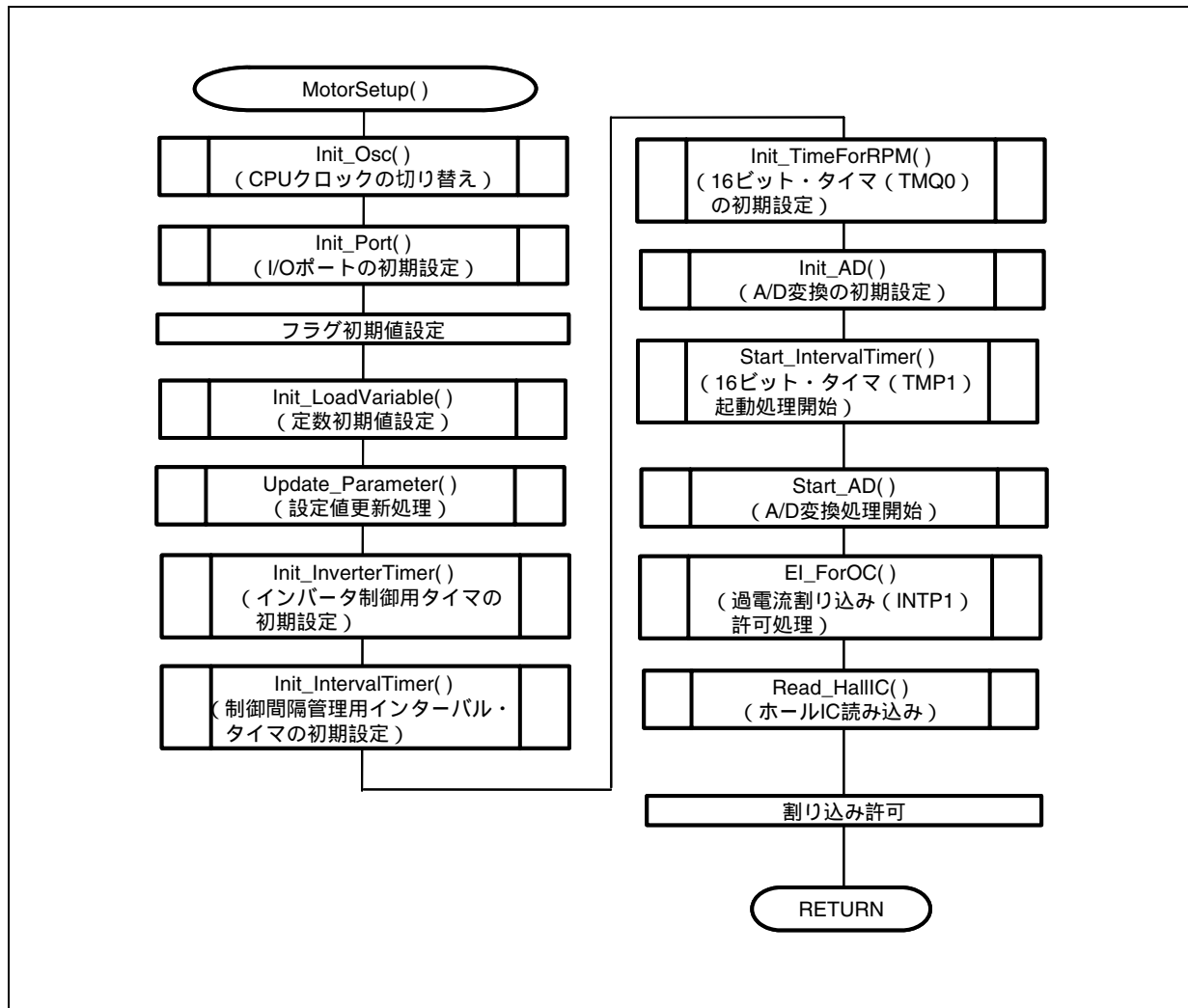
図4 - 88 ウォッチドッグ・タイマ・カウント・クリア処理



【motor.cファイルのフロー・チャート】

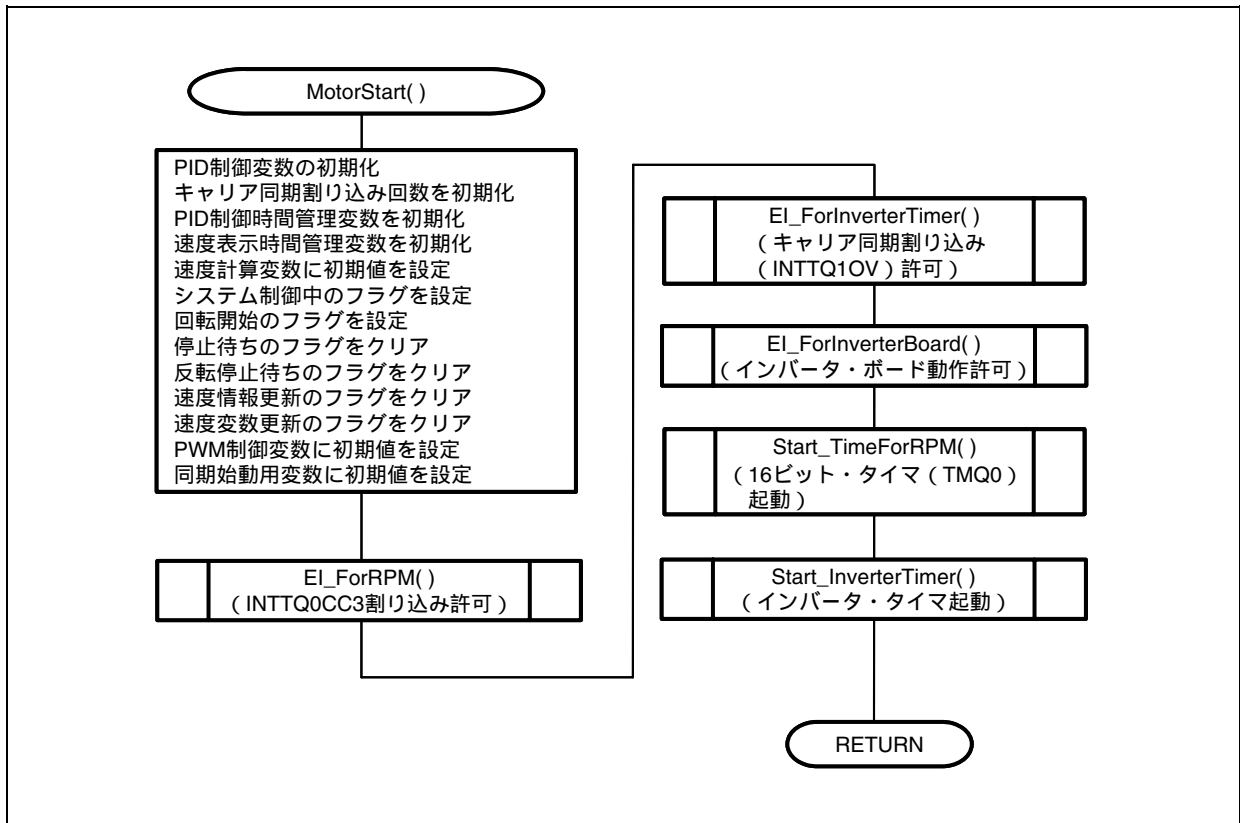
(1) 初期設定処理

図4 - 89 初期設定処理



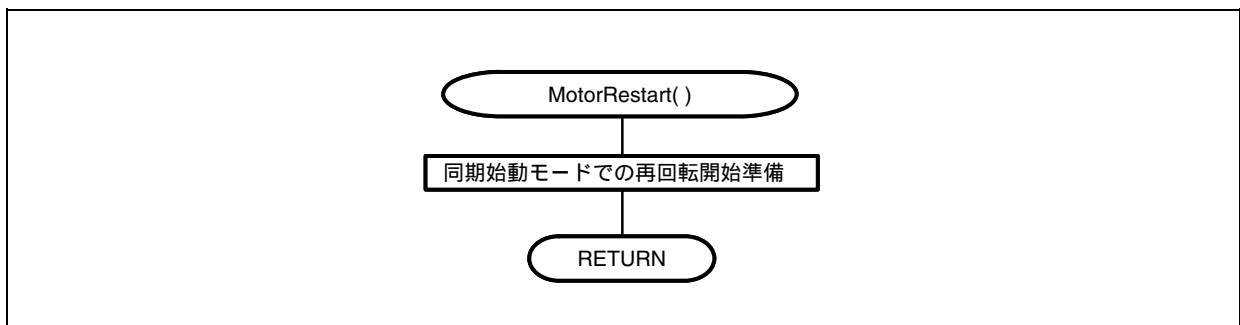
(2) モータ制御開始処理

図4 - 90 モータ制御開始処理



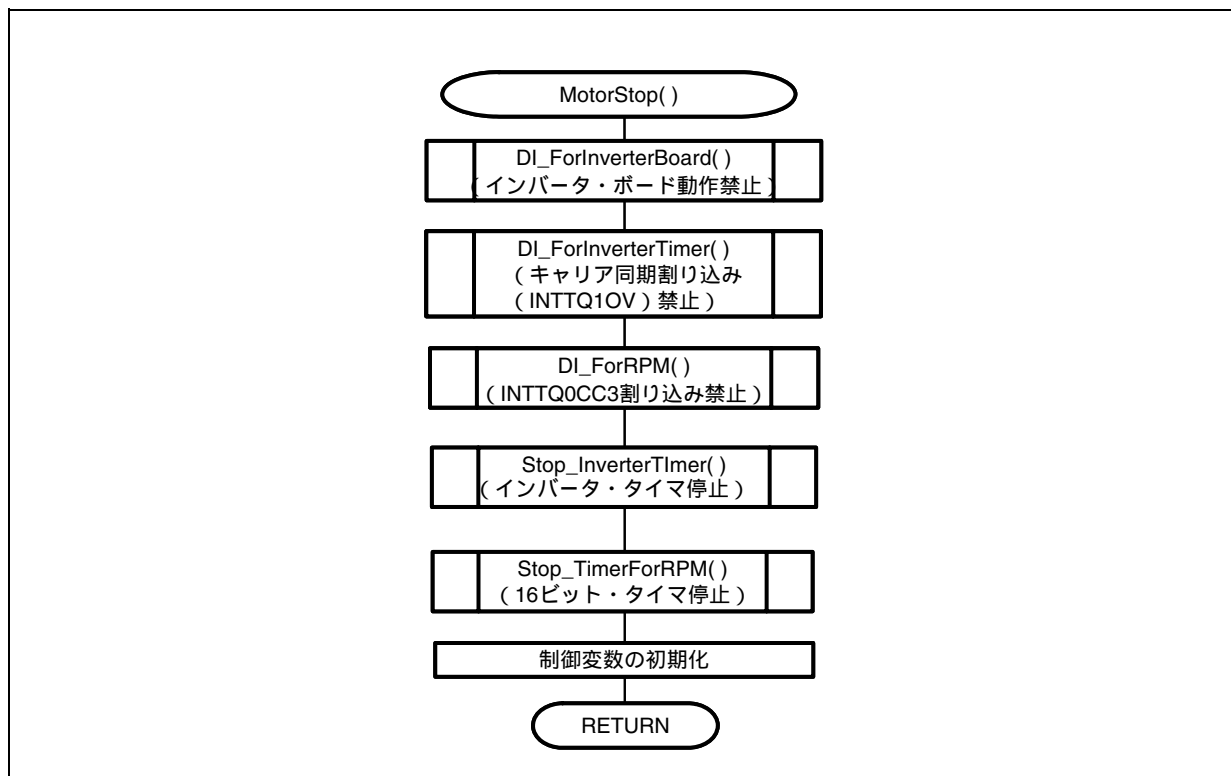
(3) 反転停止後のモータ制御再開処理

図4 - 91 反転停止後のモータ制御再開処理



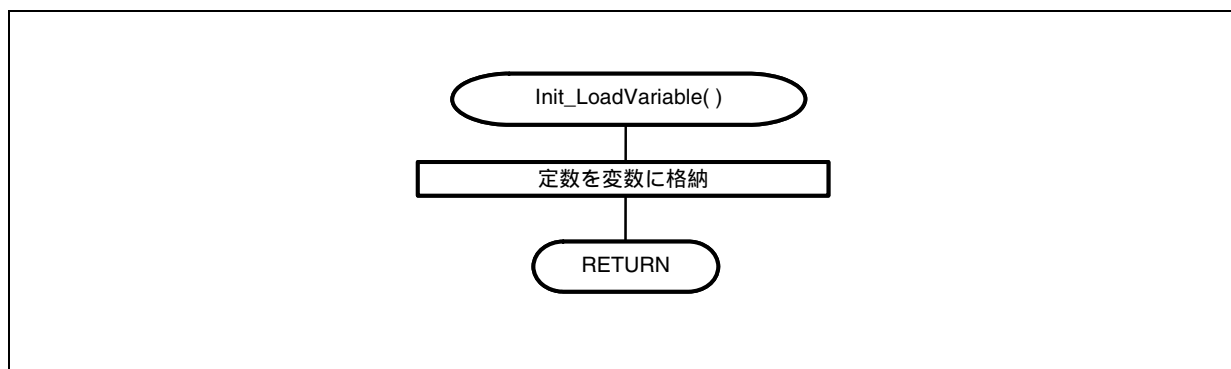
(4) モータ制御停止処理

図4 - 92 モータ制御停止処理



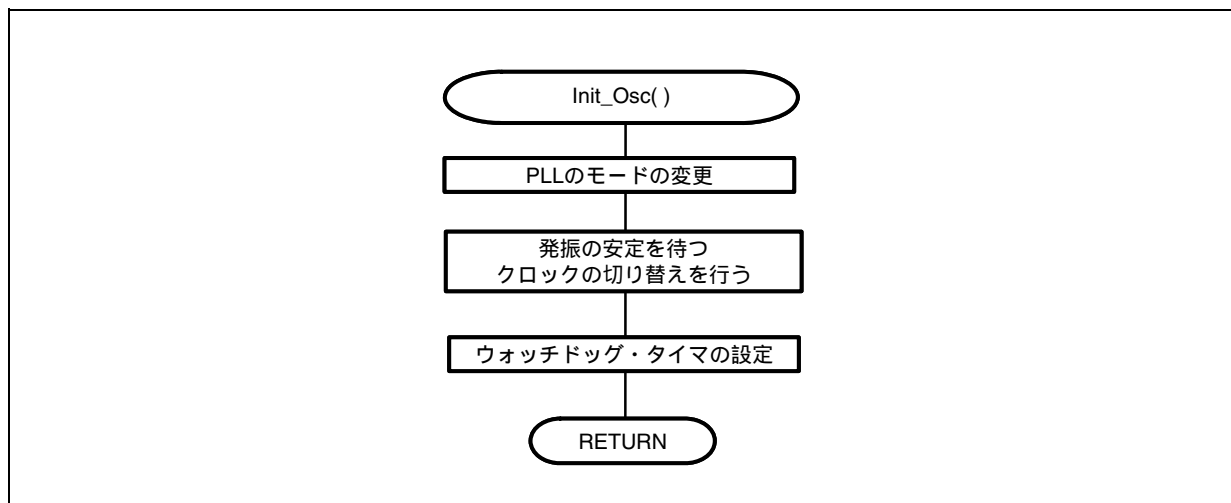
(5) 変数の初期設定処理

図4 - 93 変数の初期設定処理



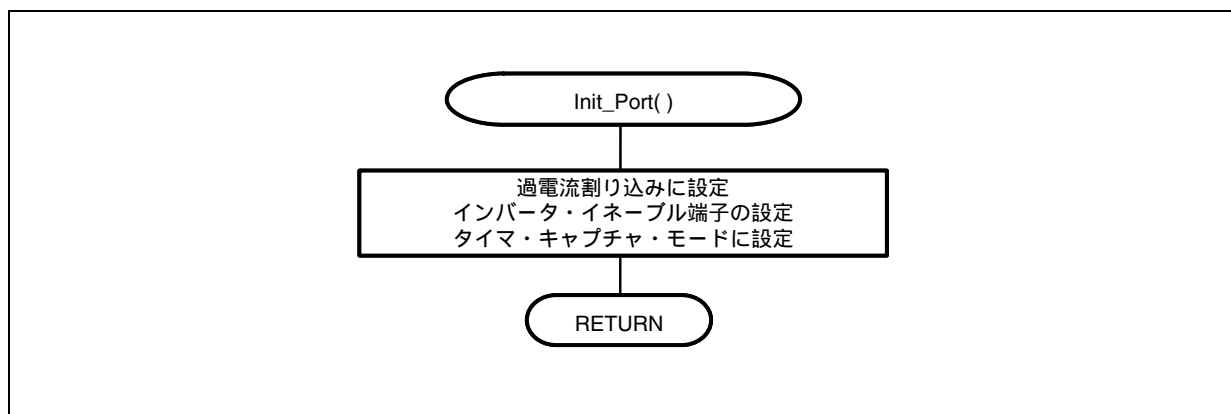
(6) CPUクロックの切り替え処理

図4 - 94 CPUクロックの切り替え処理



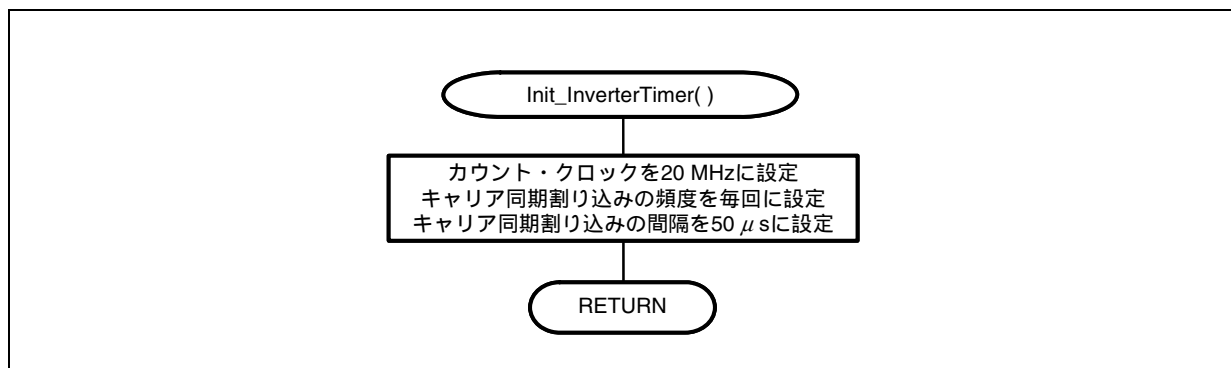
(7) I/Oポートの初期設定処理

図4 - 95 I/Oポートの初期設定処理



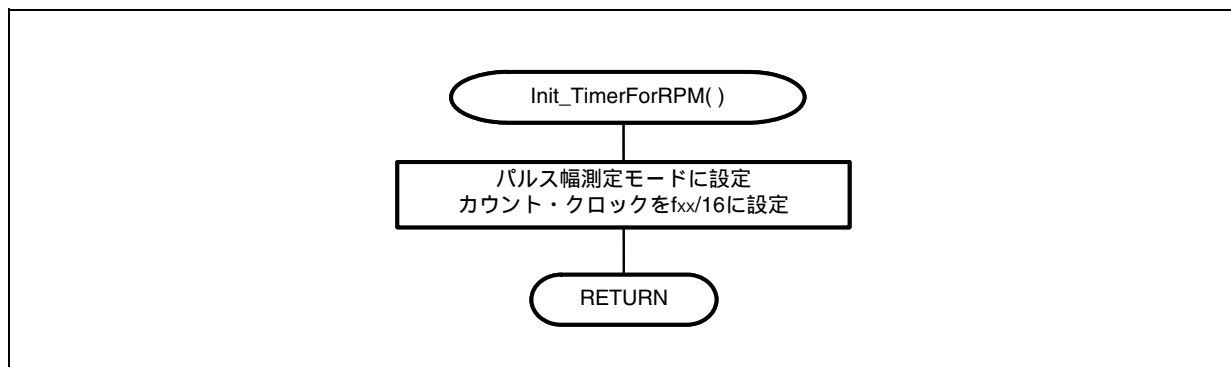
(8) 16ビット・インバータ制御用タイマの初期設定処理

図4 - 96 16ビット・インバータ制御用タイマの初期設定処理



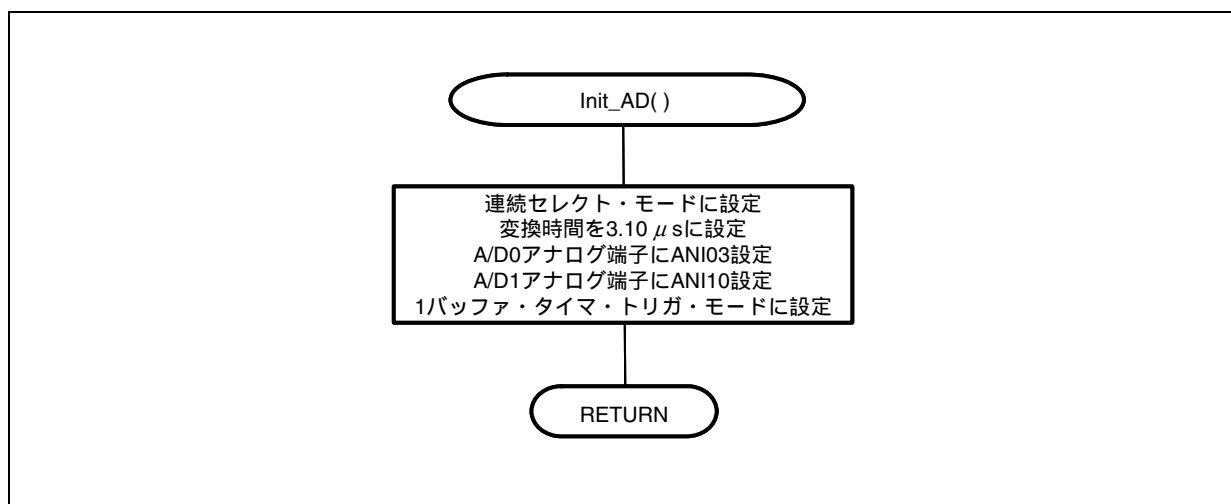
(9) 16ビット・タイマ (TMQ0) の初期設定処理

図4 - 97 16ビット・タイマ (TMQ0) の初期設定処理



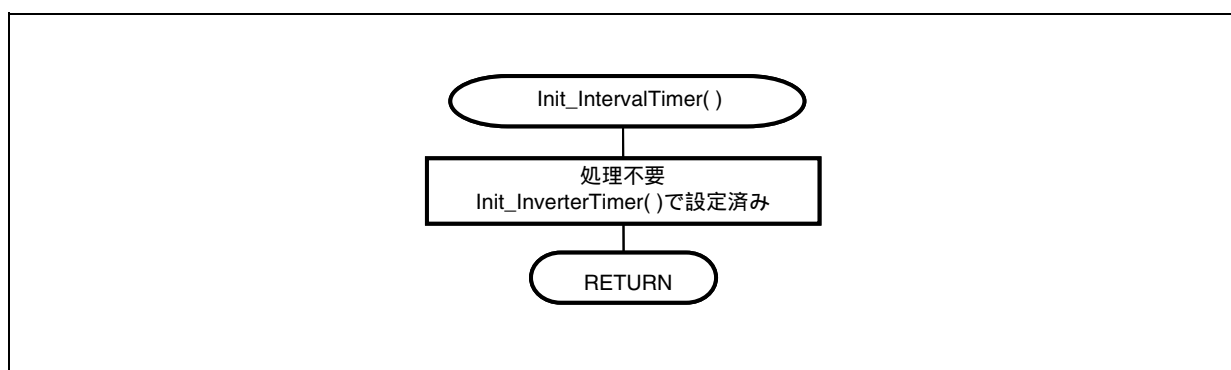
(10) A/D変換初期設定処理

図4 - 98 A/D変換初期設定処理



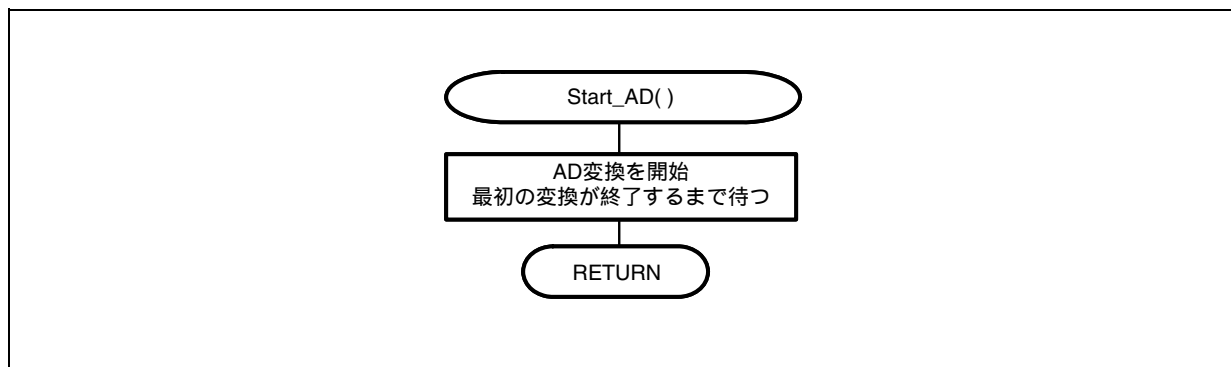
(11) 制御間隔管理用インターバル・タイマ初期設定処理

図4 - 99 制御間隔管理用インターバル・タイマ初期設定処理



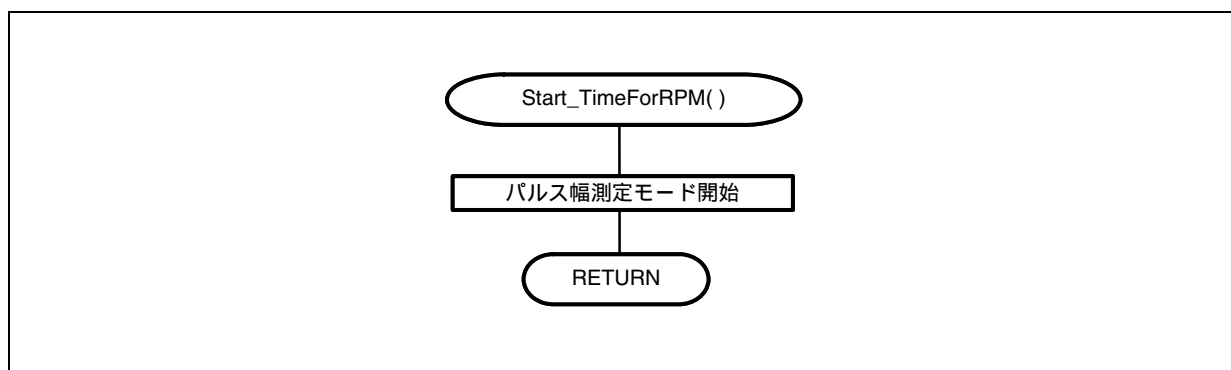
(12) A/D変換起動処理

図4 - 100 A/D変換起動処理



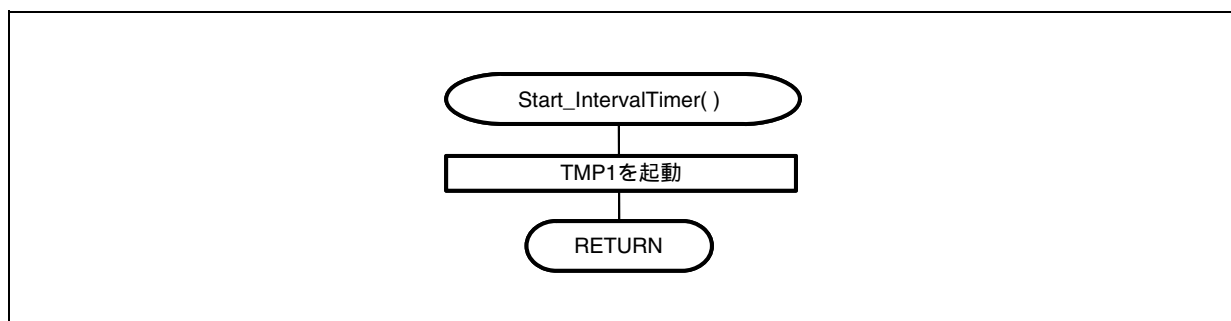
(13) 16ビット・タイマ (TMQ0) 起動処理

図4 - 101 16ビット・タイマ (TMQ0) 起動処理



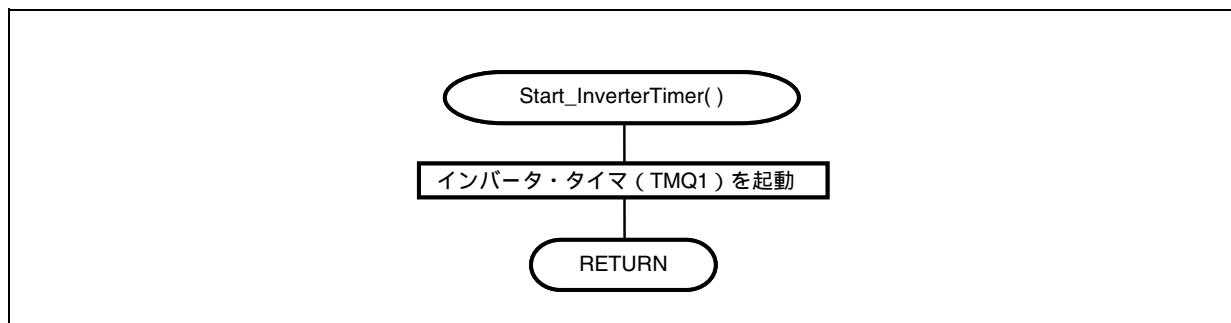
(14) 16ビット・タイマ (TMP1) 起動処理

図4 - 102 16ビット・タイマ (TMP1) 起動処理



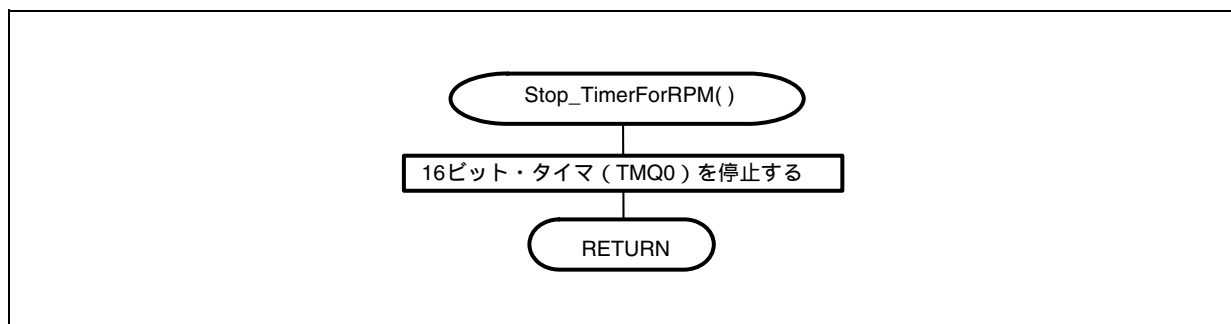
(15) インバータ制御タイマ起動 (PWMの出力開始) 処理

図4 - 103 インバータ制御タイマ起動 (PWMの出力開始) 処理



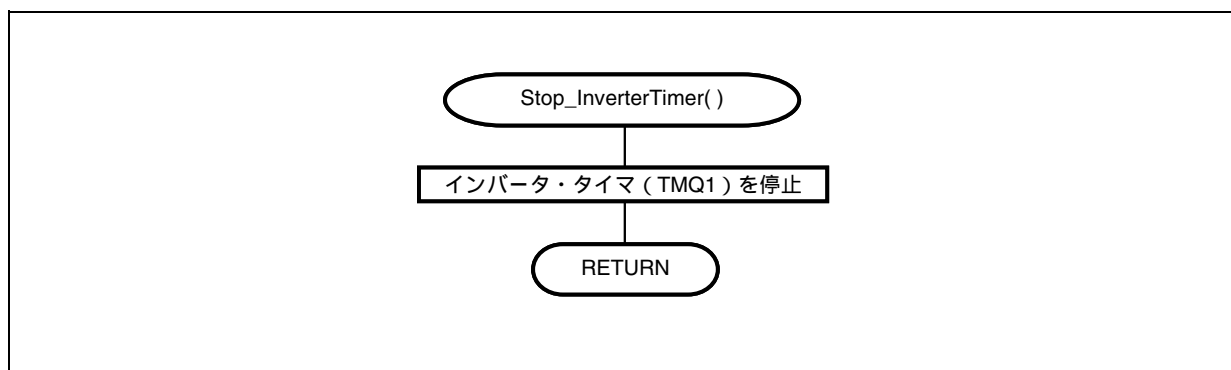
(16) 16ビット・タイマ (TMQ0) 停止処理

図4 - 104 16ビット・タイマ (TMQ0) 停止処理



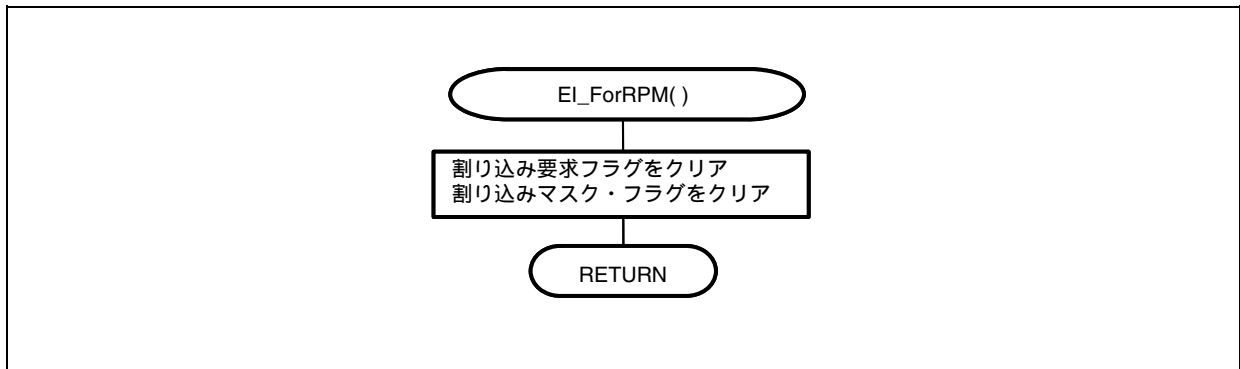
(17) インバータ制御タイマ停止 (PWMの出力停止) 処理

図4 - 105 インバータ制御タイマ停止 (PWMの出力停止) 処理



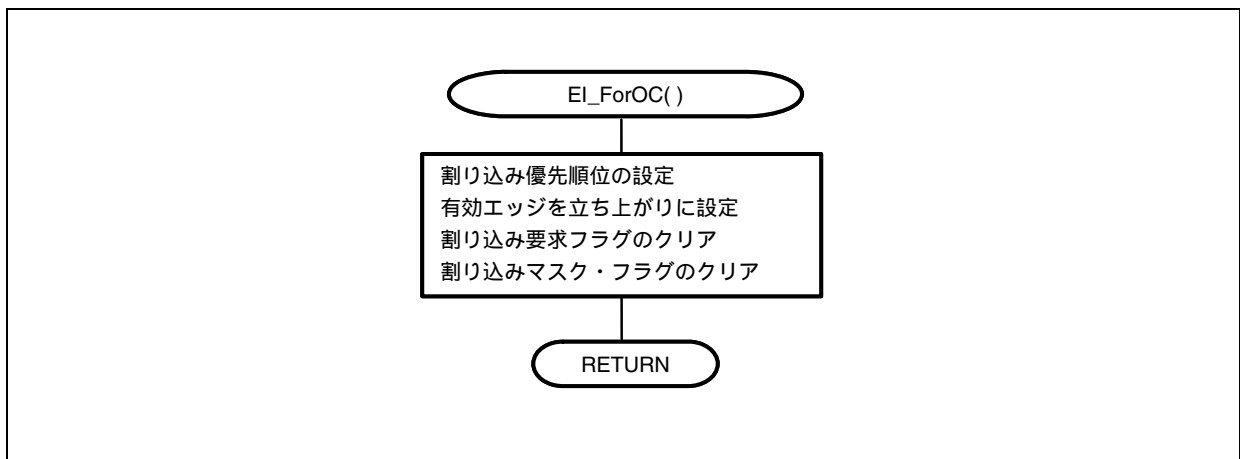
(18) 速度情報取得用割り込み (INTTQ0CC3) 許可処理

図4 - 106 速度情報取得用割り込み (INTTQ0CC3) 許可処理



(19) 過電流割り込み (INTP1) 許可処理

図4 - 107 過電流割り込み (INTP1) 許可処理



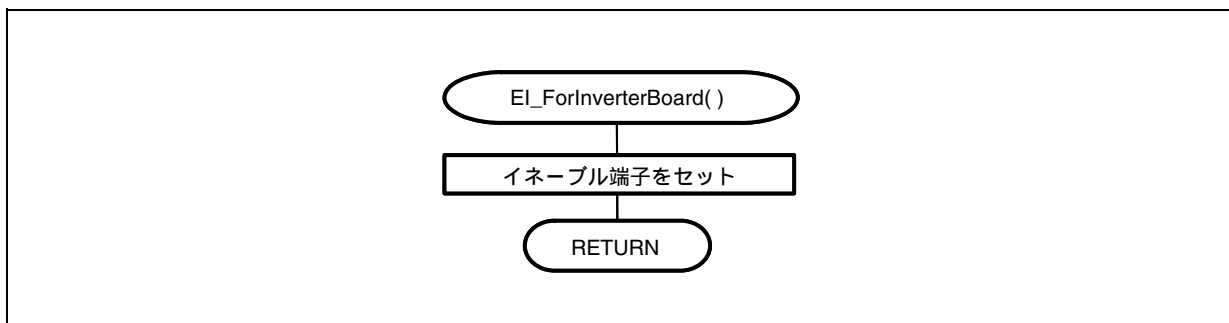
(20) キャリア同期割り込み (INTTQ1OV) 許可処理

図4 - 108 キャリア同期割り込み (INTTQ1OV) 許可処理



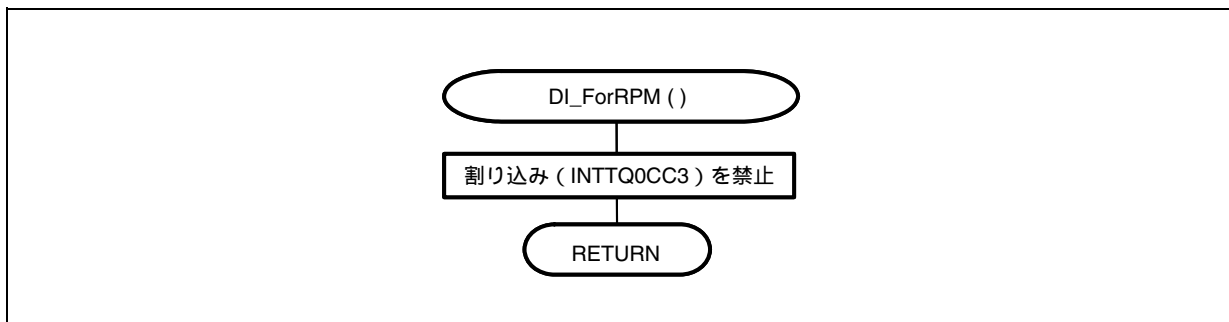
(21) インバータ・ボード動作許可処理

図4 - 109 インバータ・ボード動作許可処理



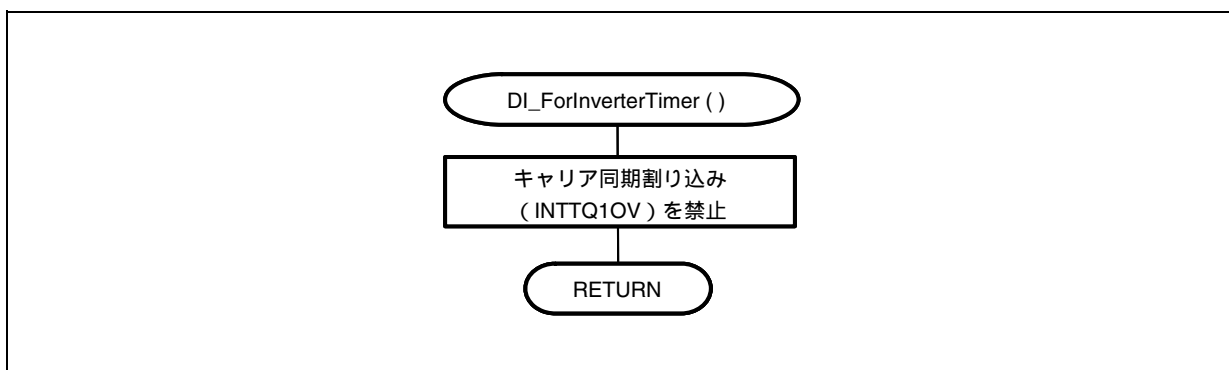
(22) 速度情報取得用割り込み (INTTQ0CC3) 禁止処理

図4 - 110 速度情報取得用割り込み (INTTQ0CC3) 禁止処理



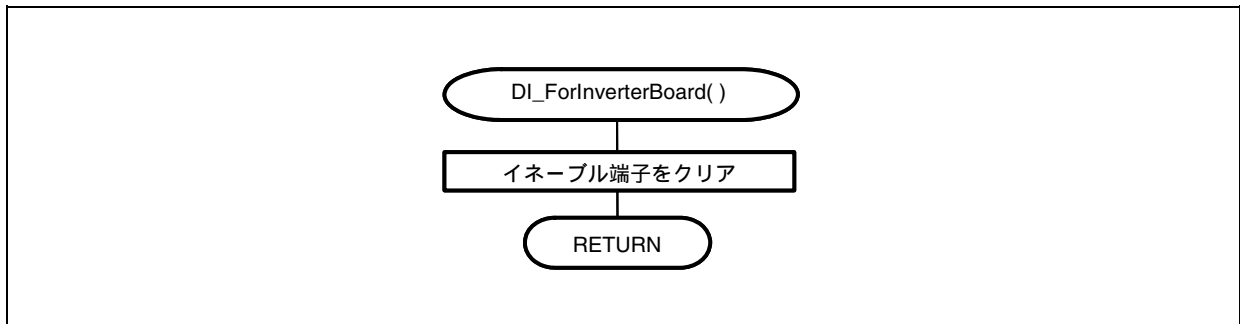
(23) キャリア同期割り込み (INTTQ1OV) 禁止処理

図4 - 111 キャリア同期割り込み (INTTQ1OV) 禁止処理



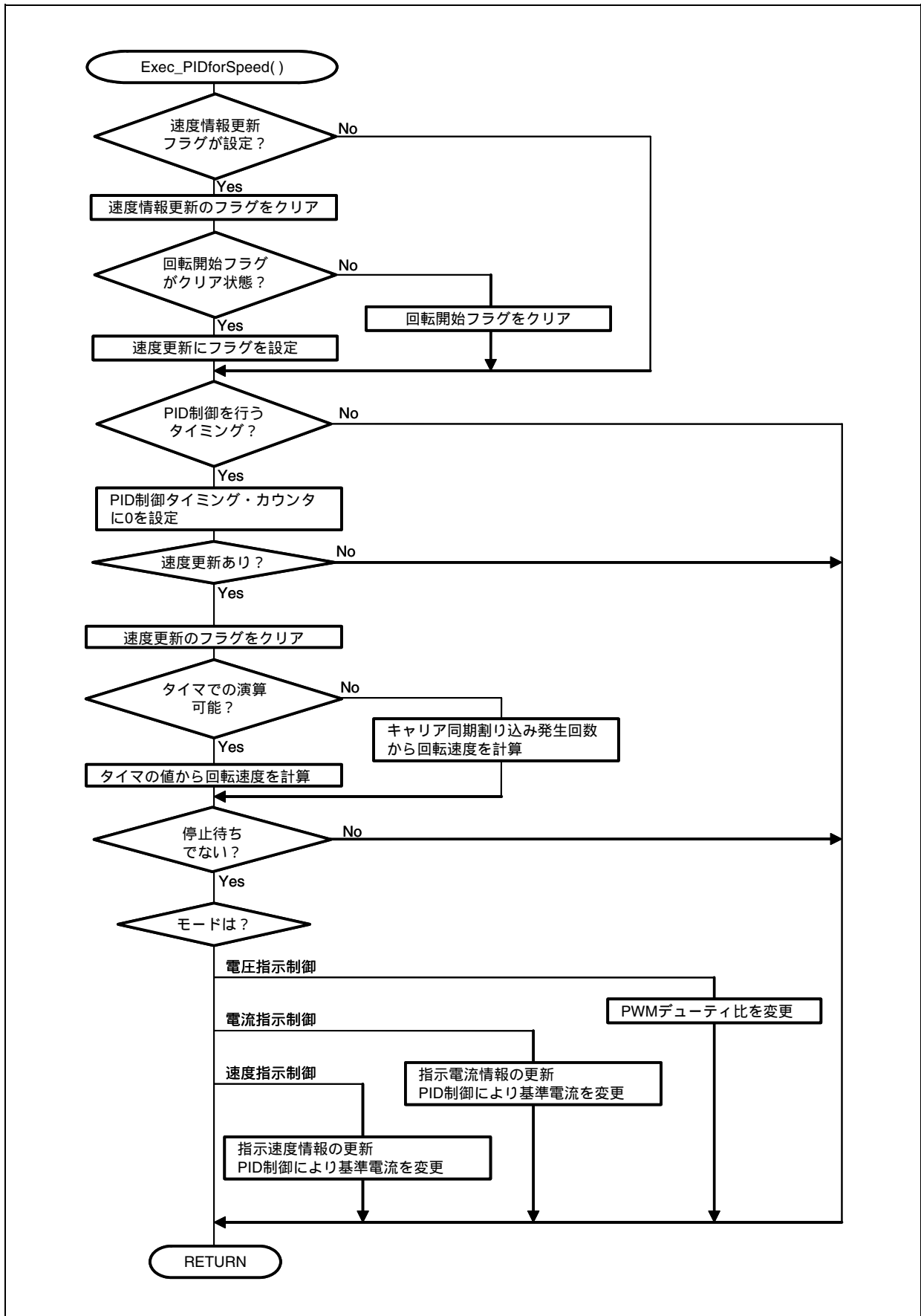
(24) インバータ・ボード動作禁止処理

図4 - 112 インバータ・ボード動作禁止処理



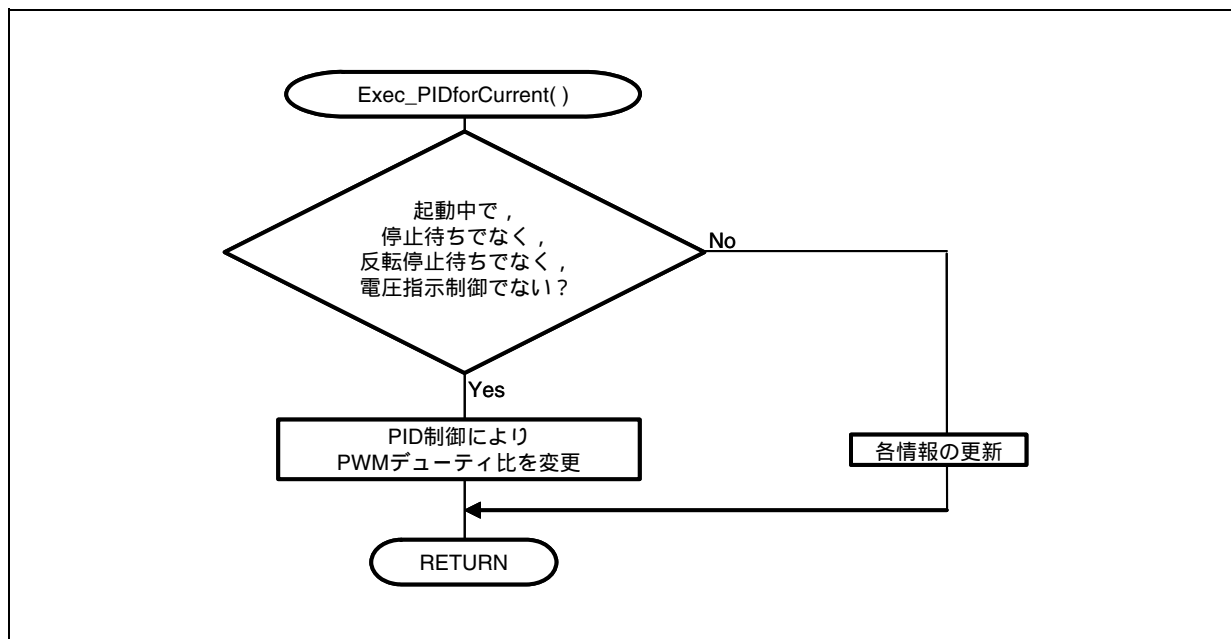
(25) 速度指示制御PID演算処理

図4 - 113 速度指示制御PID演算処理



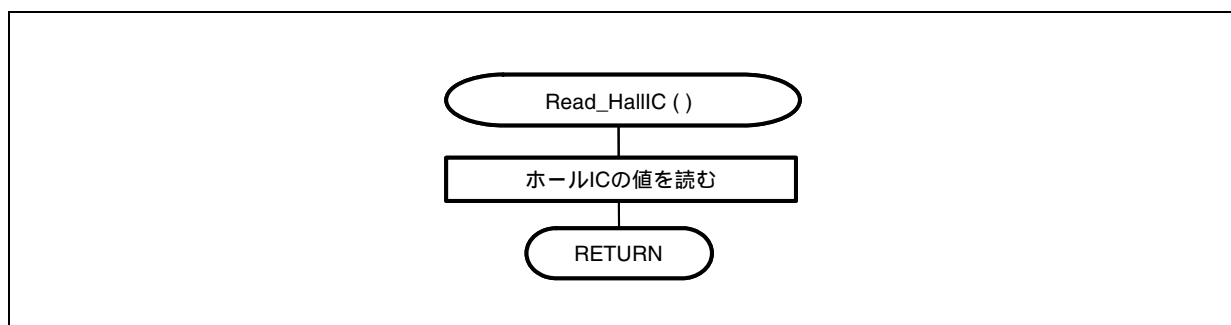
(26) 電流指示制御PID演算処理

図4 - 114 電流指示制御PID演算処理



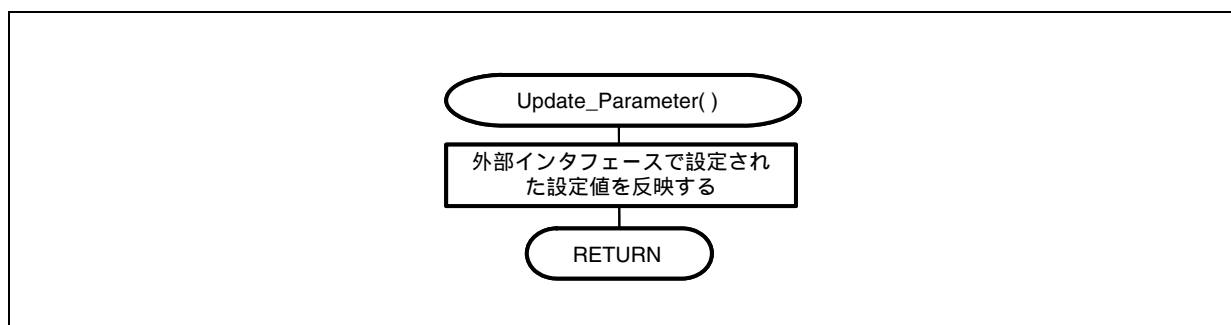
(27) ホールICの値読み出し処理

図4 - 115 ホールICの値読み出し処理



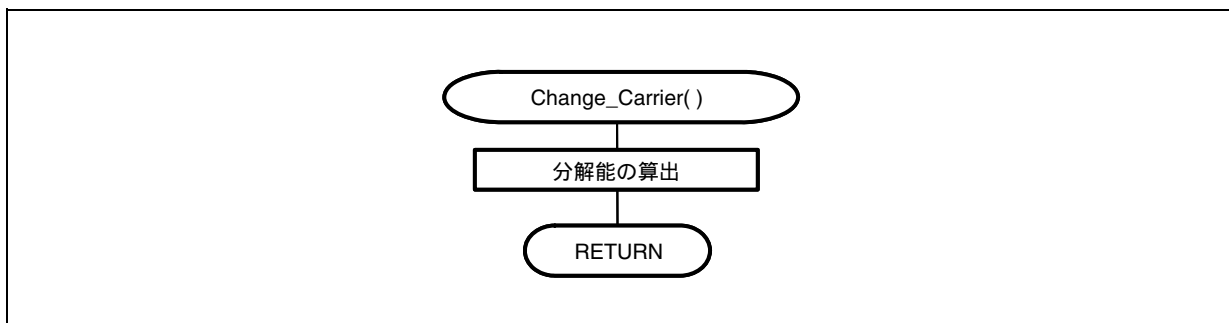
(28) 設定値更新処理

図4 - 116 設定値更新処理



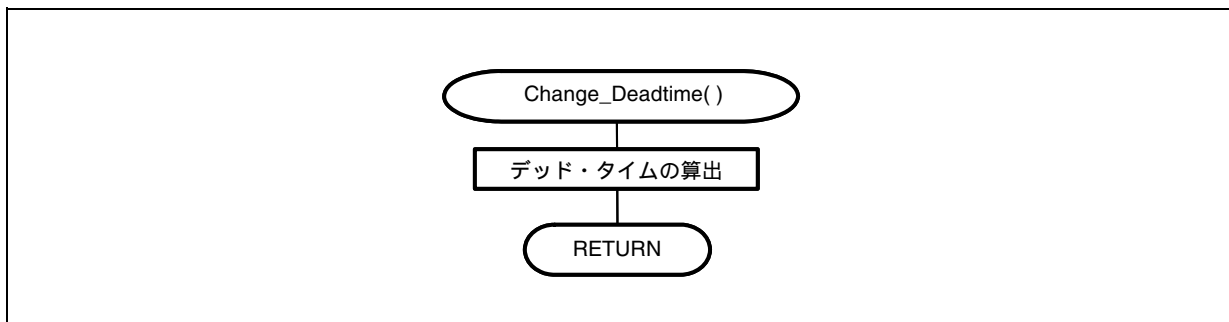
(29) キャリア周波数更新処理

図4 - 117 キャリア周波数更新処理



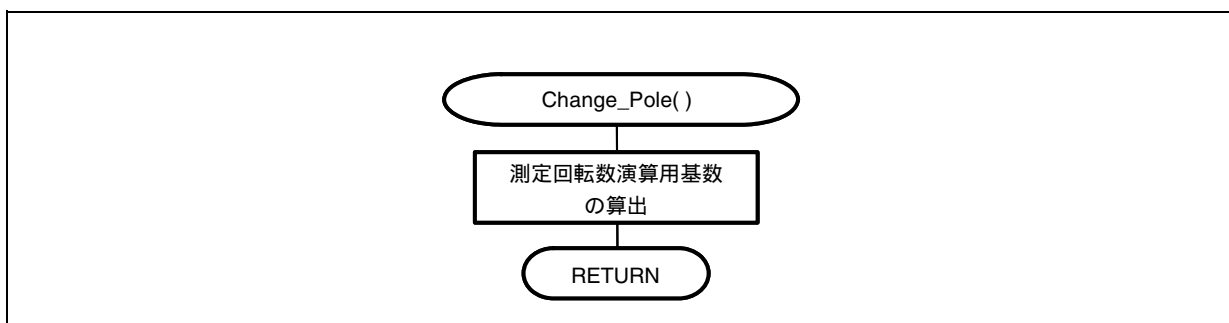
(30) デッド・タイム更新処理

図4 - 118 デッド・タイム更新処理



(31) 極対数更新処理

図4 - 119 極対数更新処理



(32) 速度指示制御間隔更新処理

図4 - 120 速度指示制御間隔更新処理



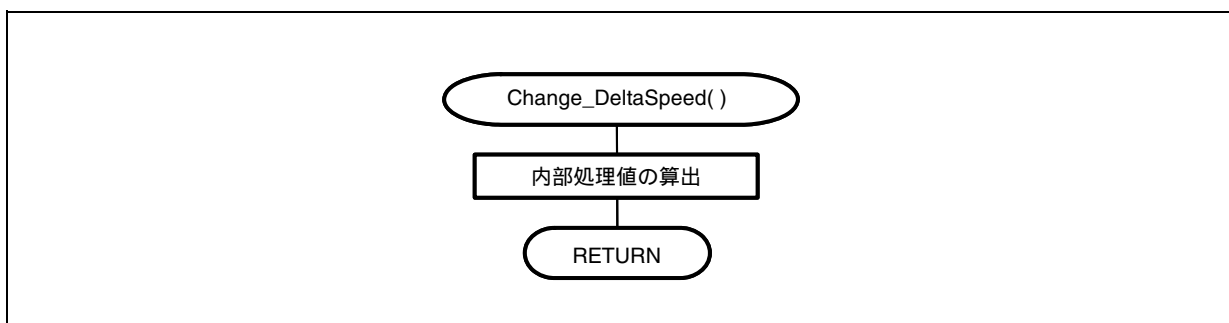
(33) 電流指示制御間隔更新処理

図4 - 121 電流指示制御間隔更新処理



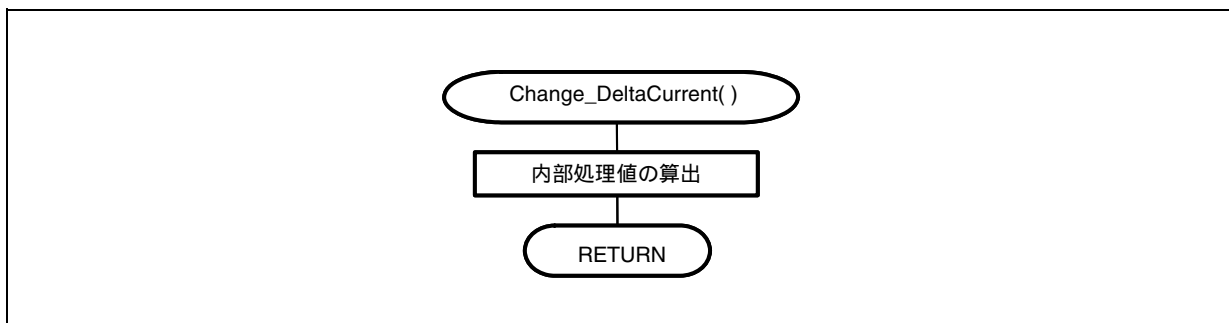
(34) 速度の変化量更新処理

図4 - 122 速度の変化量更新処理



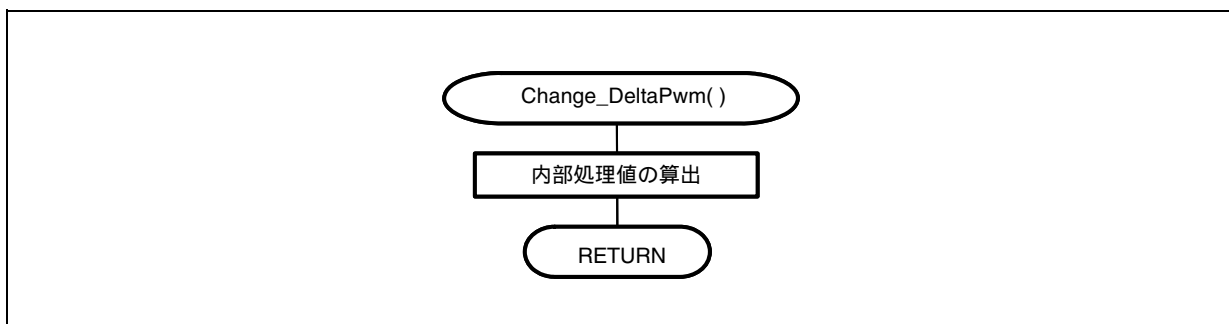
(35) 電流の変化量更新処理

図4 - 123 電流の変化量更新処理



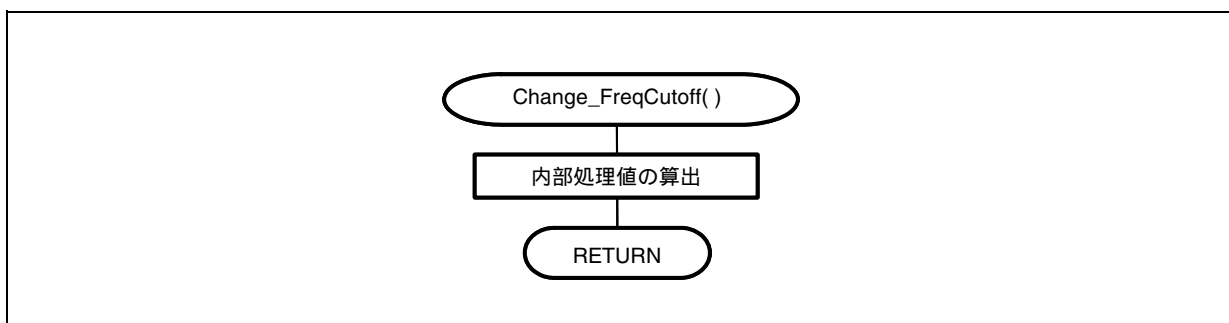
(36) PWMの変化量更新処理

図4 - 124 PWMの変化量更新処理



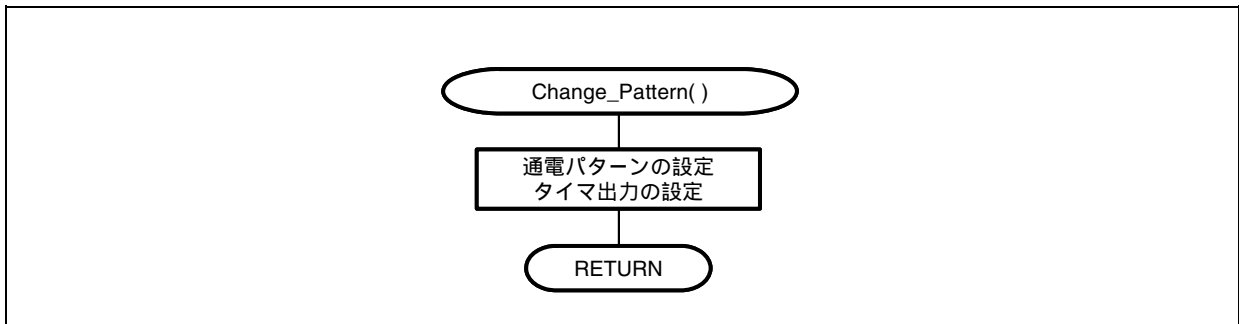
(37) カットオフ周波数更新処理

図4 - 125 カットオフ周波数更新処理



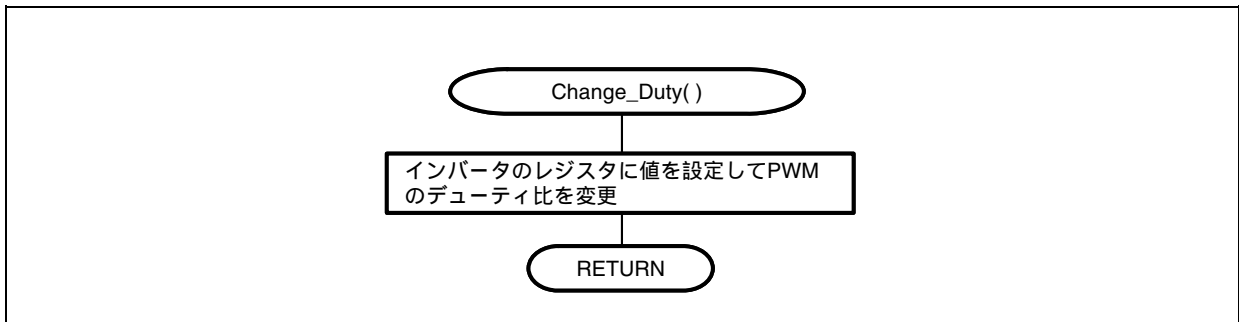
(38) 通電パターン切り替え処理

図4 - 126 通電パターン切り替え処理



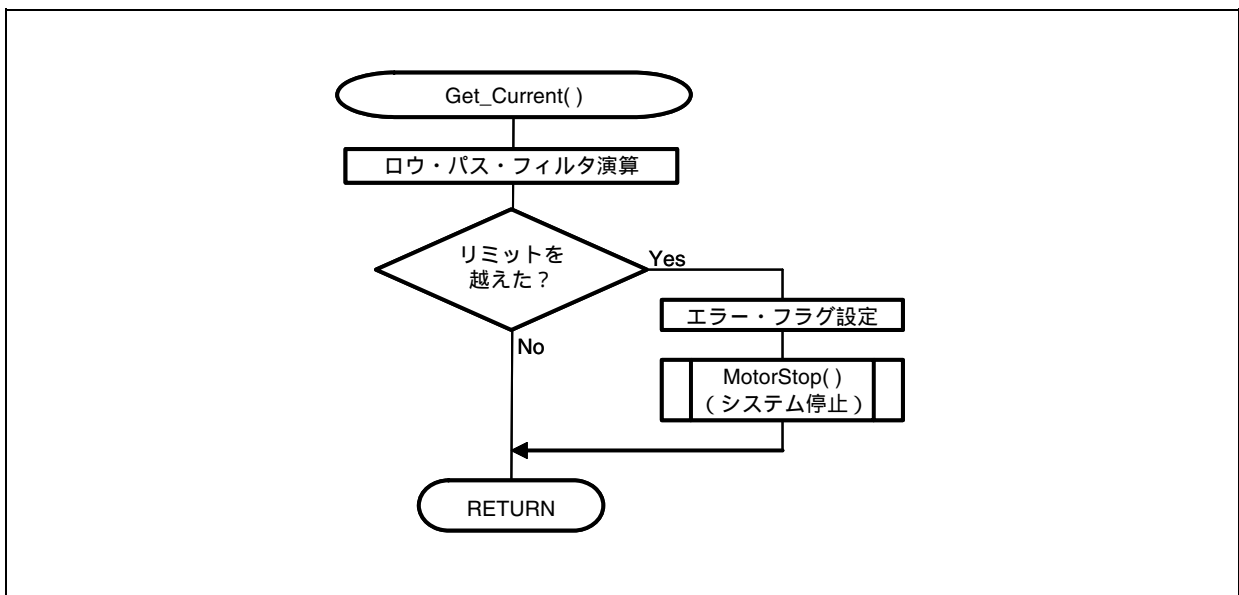
(39) PWMのデューティ比変更処理

図4 - 127 PWMのデューティ比変更処理



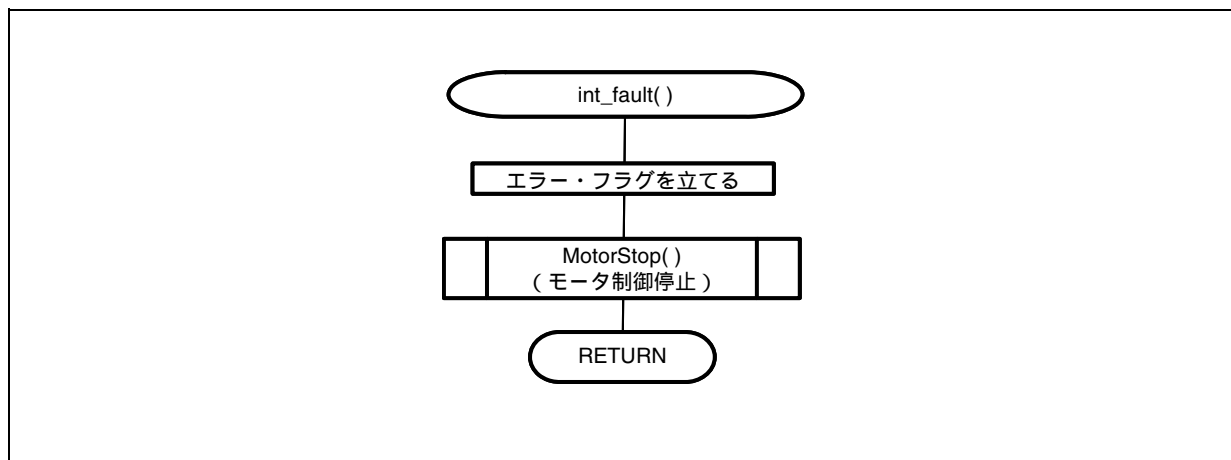
(40) 電流値取得処理

図4 - 128 電流値取得処理



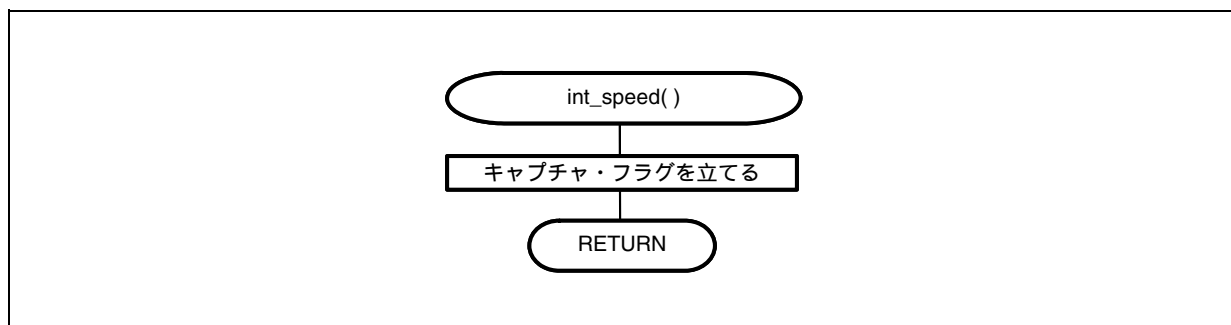
(41) ハードウェア過電流検知割り込み処理

図4 - 129 ハードウェア過電流検知割り込み処理



(42) 速度情報取得用割り込み (INTTQ0CC3) 処理

図4 - 130 速度情報取得用割り込み (INTTQ0CC3) 処理



(43) キャリア同期割り込み(谷)処理

図4 - 131 キャリア同期割り込み(谷)処理(1/2)

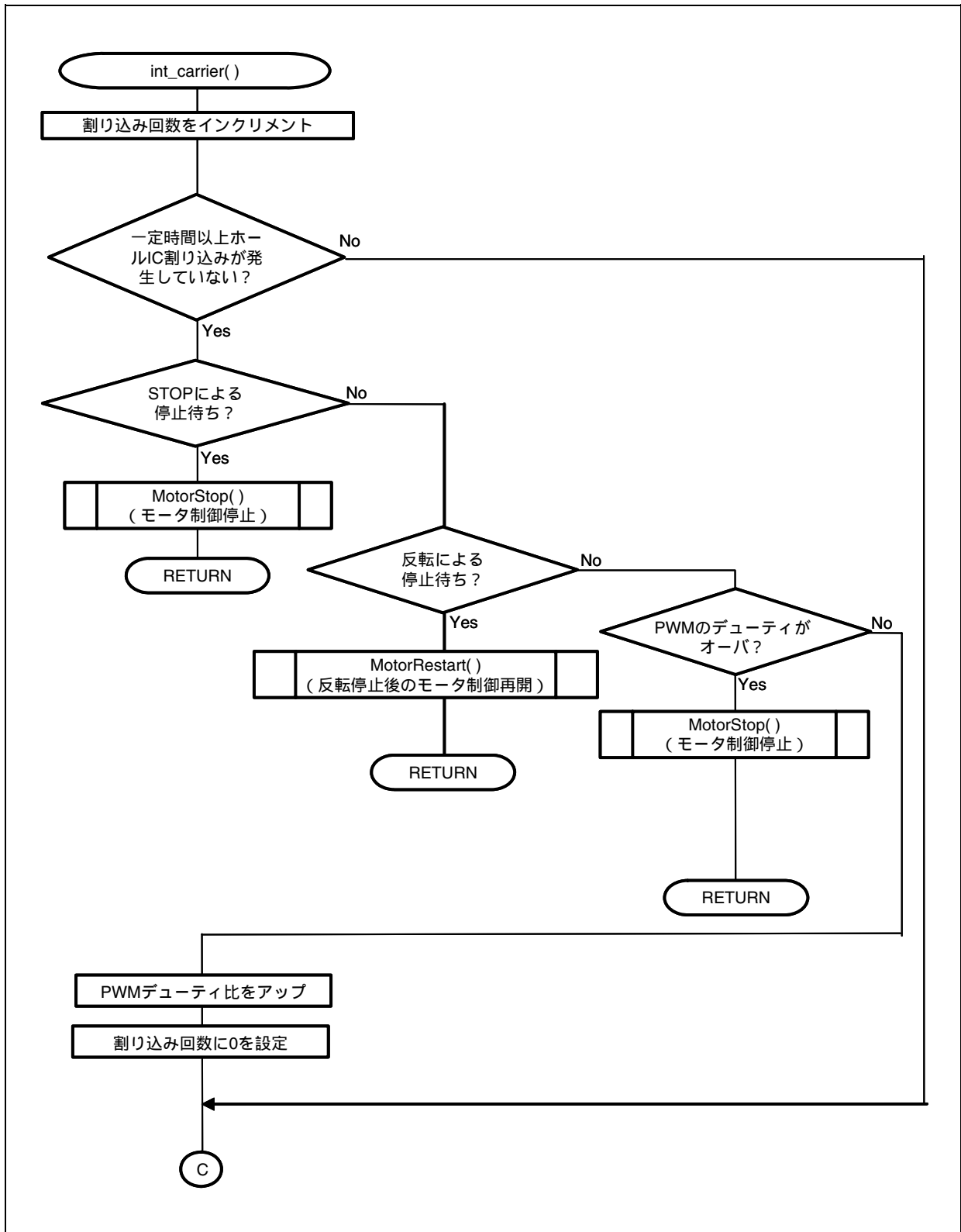
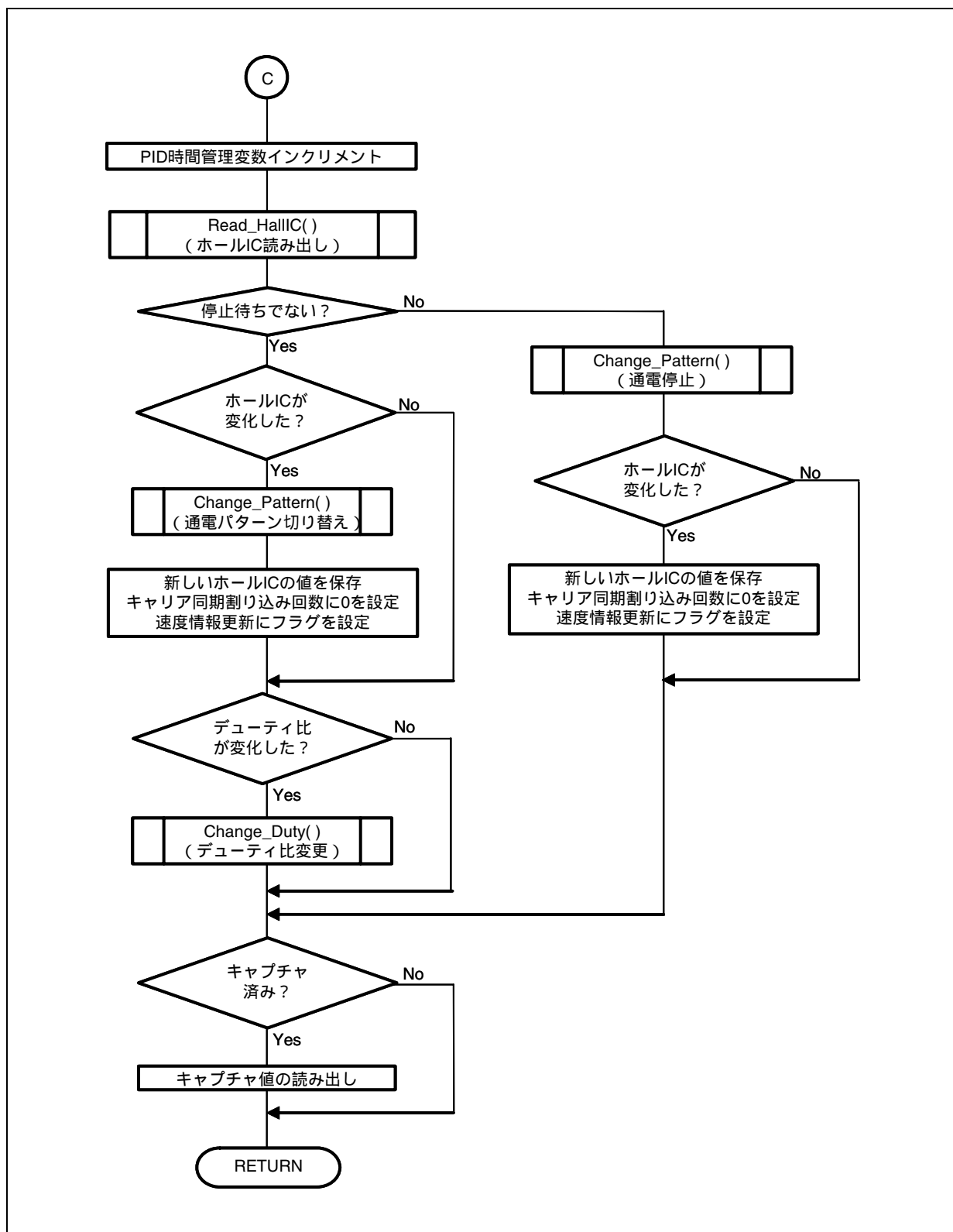
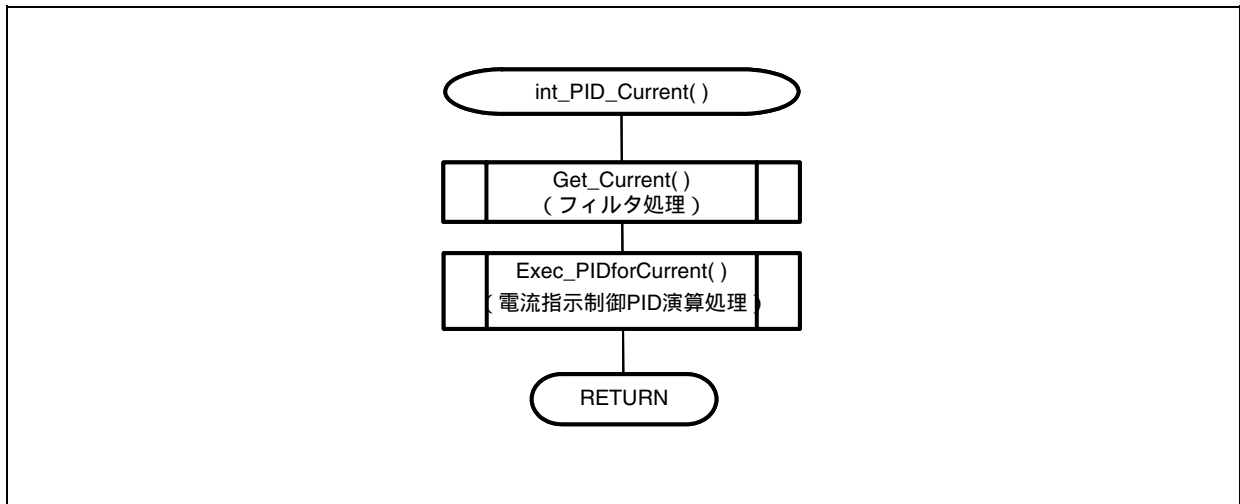


図4 - 131 キャリア同期割り込み (谷) 処理 (2/2)



(44) 電流マイナ・ループ用タイマ割り込み処理

図4 - 132 電流マイナ・ループ用タイマ割り込み処理



4.10 定数一覧

4.10.1 外部インタフェース用定数

(1) 設定変更可能定数

(a) motor.hファイル

(1/2)

名 称	意 味	設定値	備 考	
MOTOR_CARRIER_FREQ	キャリア周波数の指定	20	[kHz]	
MOTOR_DEADTIME	デッド・タイムの指定	2	[us]	
MOTOR_POLE	モータの極対数を指定	2		
MOTOR_INTERVALTIME_RPM	速度制御周期を指定	150	[ms]	
MOTOR_KRP_REF	速度制御用Kp値を指定	0.7		
MOTOR_KRI_REF	速度制御用Ki値を指定	0.014		
MOTOR_KRD_REF	速度制御用Kd値を指定	0.7		
MOTOR_INTERVALTIME_CURRENT	電流制御周期を指定	1.43	[ms]	
MOTOR_KIP_REF	電流制御用Kp値を指定	0.35		
MOTOR_KII_REF	電流制御用Ki値を指定	0.007		
MOTOR_KID_REF	電流制御用Kd値を指定	0.35		
MOTOR_MAX_SPEED	モータの最大回転数を指定	7000	[rpm]	
MOTOR_HIGH_SPEED	目標回転数の上限を指定	3000	[rpm]	
MOTOR_LOW_SPEED	目標回転数の下限を指定	300	[rpm]	
MOTOR_HIGH_CURRENT	目標電流値の上限を指定	1023		
MOTOR_LOW_CURRENT	目標電流値の下限を指定	10		
MOTOR_HIGH_PWM	目標PWM値の上限を指定	491		
MOTOR_LOW_PWM	目標PWM値の下限を指定	2		
MOTOR_RPM_RATE_MAX	回転数の変化量を指定	2000		
MOTOR_CURRENT_RATE_MAX	電流値の変化量を指定	900		
MOTOR_PWM_RATE_MAX	PWM値の変化量を指定	1000		
MOTOR_STARTUP_METHOD_REF	初期起動方法を指定	PWM_START	センサレス制御用初期起動プロセスのための設定です。 PWM2_REFのみ使用しています。	
MOTOR_TIME_KNEE_REF	次の起動制御までの時間を指定	0.75		
MOTOR_TIME_END_REF	初期起動終了時間を指定	1.5		
MOTOR_ST_RPM0_REF	初期起動時回転数を指定	60		
MOTOR_ST_RPM1_REF	次の制御時間時の回転数を指定	100		
MOTOR_ST_RPM2_REF	初期起動終了時回転数を指定	200		
MOTOR_ST_CRNT0_REF	初期起動時電流値を指定	140		
MOTOR_ST_CRNT1_REF	次の制御時間時の電流値を指定	160		
MOTOR_ST_CRNT2_REF	初期起動終了時電流値を指定	160		
MOTOR_ST_PWM0_REF	初期起動時PWM値を指定	100		
MOTOR_ST_PWM1_REF	次の制御時間時のPWM値を指定	100		
MOTOR_ST_PWM2_REF	初期起動終了時PWM値を指定	100		
MOTOR_AD_GAIN	A/D変換値の増幅量を指定	1		
MOTOR_AD_OFFSET	A/D変換値のオフセット量を指定	0		

名 称	意 味	設定値	備 考
MOTOR_LIMIT_CURRENT_SOFT	ソフトウェア過電流検知用リミットを指定	1	[A]
MOTOR_CUTOFF_FREQ	電流測定値ノイズ・カットオフ周波数を指定	13	[Hz]
MOTOR_CHANGE_PAT_OFFSET	通電パターン変更時の加算タイム値	3	

(2) 参照可能定数

(a) motor.hファイル

名 称	意 味	設定値	備 考
MOTOR_CW	回転方向	0x00	モータの回転方向 (CW) を示します。
MOTOR_CCW		0x01	モータの回転方向 (CCW) を示します。
MOTOR_OFF	モータ動作状態	0	モータ停止中を示します。
MOTOR_ON		1	モータ回転中を示します。
MOTOR_MODE_SPEED	動作モード	0x01	速度指示制御を示します。
MOTOR_MODE_CURRENT		0x02	電流指示制御を示します。
MOTOR_MODE_V		0x03	電圧指示制御を示します。
MOTOR_START_TR	モータ・コントロールI/Oボードからの受け付け命令	0x01	モータ回転開始指示を示します。
MOTOR_STOP_TR		0x02	モータ回転停止指示を示します。
MOTOR_FORWARD_TR		0x04	CW方向への回転指示を示します。
MOTOR_REVERSE_TR		0x08	CCW方向への回転指示を示します。
MOTOR_MODE_TR		0x10	モード切り替え指示を示します。
MOTOR_ERROR_TR		0x80	指示受け付けエラーを示します。
MOTOR_ERROR_NONE		エラー・フラグ・ビット	0x00
MOTOR_ERROR_HALL	0x01		ホールICのエラーを示します。
MOTOR_ERROR_OC	0x02		ハードウェア過電流によるエラーを示します。
MOTOR_ERROR_MOTOR	0x04		モータ異常のエラーを示します。
MOTOR_ERROR_S_OC	0x08		ソフトウェア過電流によるエラーを示します。
CURRENT_START	初期起動モード	0x01	電流指示による起動を示します。
PWM_START		0x02	電圧指示による起動を示します。
NOTABLE_START		0x03	
STATUS_NORMAL_END	外部インタフェース関数実行フラグ	1	正常終了を示します。
STATUS_ABNORMAL_END		- 1	異常終了を示します。

4.10.2 内部定数

(1) motor.cファイル

(1/3)

名 称	意 味	設定値	備 考
SW_AD_0	レジスタ置き換え	ADA0CE	
SW_AD_1		ADA1CE	
SW_INTERVAL_TIMER		TP1CE	
SW_INVERTER_TIMER		TQ1CE	
SW_TIMER_FOR_RPM		TQ0CE	
INVERTER_SW		P1.4	
INVERTER_SW_MODE		PM1.4	
PM_OVERCURRENT		PM0.1	
PMC_OVERCURRENT		PMC0.1	
IF_AD_0		AD0IF	
IF_AD_1		AD1IF	
IF_INVERTER_TIMER		TQ1OVIF	
IF_TIMER_FOR_RPM		TQ0CCIF3	
IF_INTP_FOR_OC		PIF1	
MK_INTERVAL_TIMER		AD0MK	
MK_INVERTER_TIMER		TQ1OVMK	
MK_TIMER_FOR_RPM		TQ0CCMK3	
MK_INTP_FOR_OC		PMK1	
PR0_INTP_FOR_OC		PIC1.0	
PR1_INTP_FOR_OC		PIC1.1	
PR2_INTP_FOR_OC		PIC1.2	
INTR_INTP_FOR_OC		INTR01	
INTF_INTP_FOR_OC		INTF01	
GET_HALLIC_VALUE	ホールIC読み込み用命令置き換え	(P0 & 0x1C)>>2	
CMD_SET_RESET	命令置き換え	WDTE = WDTE_RESET	
CMD_WDT_CLEAR		WDTE = WDTE_CLR	
WDTM_SET	WDT設定値	0x67	
WDTE_CLR		0xac	
WDTE_RESET		0x00	
IMS_DATA	メモリ・サイズ指定	0xc8	
REG_BASE	レジスタ置き換え	TQ1CCR0	
REG_PHASE_U		TQ1CCR1	
REG_PHASE_V		TQ1CCR2	
REG_PHASE_W		TQ1CCR3	
REG_DEADTIM		TQ1DTC	
REG_CMP_CAPTURE		TQ0CCR3	
REG_AD		ADA0CR3	
VALUE_SHIFT_AD	ADデータ・シフト量	6	
AD_ISHUNT	動作電流検知チャンネル	5	

名 称	意 味	設定値	備 考
P_OFF	タイム出力オフ	0	
P_STOP	タイマ出力停止	7	
P_V1	通電パターン1 U+ V-	1	
P_V2	通電パターン2 U+ W-	2	
P_V3	通電パターン3 V+ W-	3	
P_V4	通電パターン4 V+ U-	4	
P_V5	通電パターン5 W+ U-	5	
P_V6	通電パターン6 W+ V-	6	
MD_ERROR_HALL	ホールIC異常	0xF0	
MD_ERROR_OC	過電流	0xF1	
MD_ERROR_MOTOR	モータ異常	0xF2	
FLG_OFF	フラグ設定用定数	0	
FLG_ON		1	
IN	ポート・モード用定数	1	
OUT		0	
CLEAR	ポート用設定値定数	0	
SET		1	
INV_OFF	インバータ・ボード用イネーブ ル信号	1	
INV_ON		0	
BASE_TIMER_CLK_INV	タイマ・マスタ・クロック	20000000.0	[Hz]
BASE_TIMER_CLK_CUT	タイマ選択クロック	1250000.0	[Hz]
BASE_CALC_RPM	回転数演算用基数	18750000	
INT_CNT_MAX	無回転待ち時間	500	
START_PWM_F_MAX	無回転時のPWM限界値	250	
FLAG_01_USEGUI	制御フラグ用参照ビット	0x01	
FLAG_01_START		0x02	
FLAG_01_WAIT_STOP		0x04	
FLAG_01_WAIT_DIR		0x08	
FLAG_01_WAIT		0x0C	
FLAG_01_MEAS_SPEED		0x10	
FLAG_01_UPDT_SPEED		0x20	
FLAG_01_GET_TIM		0x40	
FLAG_01_CAPTURE		0x80	
FLAG_UPD_INVL_TIM_RPM	更新フラグ用参照ビット	0x01	
FLAG_UPD_INVL_TIM_CURRENT		0x02	
FLAG_UPD_LMT_CURRENT_SOFT		0x04	
FLAG_UPD_CARRIER_FREQ		0x08	
FLAG_UPD_DEADTIME		0x10	
FLAG_UPD_MAX_RPM		0x20	
FLAG_UPD_HIGH_RPM		0x40	
FLAG_UPD_LOW_RPM		0x80	
FLAG_UPD_HIGH_CRNT		0x01	
FLAG_UPD_LOW_CRNT		0x02	
FLAG_UPD_HIGH_PWM		0x04	

(3/3)

名 称	意 味	設定値	備 考
FLAG_UPD_LOW_PWM	更新フラグ用参照ビット	0x08	
FLAG_UPD_TIME_PER		0x10	
FLAG_UPD_RPM_PER		0x20	
FLAG_UPD_CURRENT_PER		0x40	
FLAG_UPD_PWM_PER		0x80	
FLAG_UPD_POLE		0x01	
FLAG_UPD_FREQ_CUTOFF		0x02	

(2) sub_mcio.hファイル

(1/2)

名 称	意 味	設定値	備 考
LED_0	LED表示データ	0xc0	“ 0 ” を表示
LED_1		0xf9	“ 1 ” を表示
LED_2		0xa4	“ 2 ” を表示
LED_3		0xb0	“ 3 ” を表示
LED_4		0x99	“ 4 ” を表示
LED_5		0x92	“ 5 ” を表示
LED_6		0x82	“ 6 ” を表示
LED_7		0xf8	“ 7 ” を表示
LED_8		0x80	“ 8 ” を表示
LED_9		0x98	“ 9 ” を表示
LED_O		0xc0	“ O ” の代わりに “ 0 ” を表示
LED_C		0xc6	“ C ” を表示
LED_I		0xcf	“ I ” を表示
LED_H		0x89	“ H ” を表示
LED_A		0x88	“ A ” を表示
LED_L		0xc7	“ L ” を表示
LED_		0xff	“ ” を表示
LED_S		0x92	“ S ” を表示
LED_E		0x86	“ E ” を表示
LED_F		0x8e	“ F ” を表示
LED_P	0x8c	“ P ” を表示	
LED_dot	0x7f	“ . ” を表示	
START_SW	起動	0xe	スイッチの状態
STOP_SW	停止	0xe	
FORWARD_SW	CW	0x7	
REVERSE_SW	CCW	0xb	
MODE_SW	モード	0xd	
SW	スイッチ	$((P0 \& 0x60) \gg 1) (P1 \& 0xC0) \gg 4$	スイッチ接続ポート
SW2	レジスタ置き換え	PM0.5	START/STOP用ポート
SW3	レジスタ置き換え	PM1.7	FORWARD用ポート
SW4	レジスタ置き換え	PM1.6	REVERSE用ポート

名 称	意 味	設定値	備 考
SW5	レジスタ置き換え	PM0.6	MODE用ポート
KEY_WAIT	スイッチ観察時間	10	チャタリング除去で使用
LD_DATA	ポート制御レジスタ	PMDLL	LEDへのデータ出力ポート
LD_LED	ポート制御レジスタ	PM4	LED選択用ポート
AD_USER	A/Dデータ・シフト量	4	
MAX_SPEED	最高指令回転数 (rpm)	MOTOR_HIGH_SPEED	
MIN_SPEED	最低指令回転数 (rpm)	MOTOR_LOW_SPEED	

4.11 変数一覧

4.11.1 外部インタフェース用変数

関数で情報のやり取りを行うため、グローバル変数は用意していません。

4.11.2 内部変数

(1) motor.cファイル

(1/3)

変数名	型	意味
s_Flags_01	unsigned char	制御フラグ群
s_Flags_Upd_01	unsigned char	設定値更新フラグ1
s_Flags_Upd_02	unsigned char	設定値更新フラグ2
s_Flags_Upd_03	unsigned char	設定値更新フラグ3
s_StatusSys	unsigned char	動作状態管理変数
s_StatusErr	unsigned char	エラー状態管理変数
s_StatusDir	unsigned char	回転方向管理変数
s_StatusMode	unsigned char	制御方式管理変数
s_TimerClk_INV_K	float	演算用基数
s_TimerClk_INV_1	float	演算用基数
s_TimerClk_CUT_K	float	演算用基数
s_RefSped	int	目標回転数管理変数
s_MeasSped	int	測定回転数管理変数
s_RefSpedInner	int	処理中目標回転数管理変数
s_RefCrnt	float	目標電流値管理変数
s_MeasCrnt	float	測定電流値管理変数
s_RefCrntInner	float	処理中目標電流値管理変数
s_RefPwm	int	目標PWM値管理変数
s_ValuePwm	int	PWM値管理変数
s_ValuePwm_1	int	前回PWM値管理変数
s_ValueHIC_1	char	前回ホールIC値管理変数
s_RefKpSped	float	速度制御用Kp管理変数
s_RefKiSped	float	速度制御用Ki管理変数
s_RefKdSped	float	速度制御用Kd管理変数
s_RefKpCrnt	float	電流制御用Kp管理変数
s_RefKiCrnt	float	電流制御用Ki管理変数
s_RefKdCrnt	float	電流制御用Kd管理変数
s_ErrSped	float	回転数差分
s_ErrSped_1	float	前回回転数差分
s_ErrSped_2	float	前々回回転数差分
s_ErrCrnt	float	電流差分
s_ErrCrnt_1	float	前回電流差分
s_ErrCrnt_2	float	前々回電流差分

変数名	型	意味
s_IntvlTimRpm	int	速度制御PID演算間隔管理変数
s_IntvlTimRpm_0	int	次回速度制御PID演算間隔管理変数
s_IntvlTimCrnt	float	電流制御PID演算間隔管理変数
s_IntvlTimCrnt_0	float	次回電流制御PID演算間隔管理変数
s_LmtCrntSoftCMP	float	ソフトウェア過電流用電流A/D変換値
s_LmtCrntSoft	float	ソフトウェア過電流用電流値管理変数
s_LmtCrntSoft_0	float	次回ソフトウェア過電流用電流値管理変数
s_CarrierFreq	float	キャリア周波数管理変数
s_CarrierFreq_0	float	次回キャリア周波数管理変数
s_Deadtime	float	デッド・タイム管理変数
s_Deadtime_0	float	次回デッド・タイム管理変数
s_Pole	char	極対数管理変数
s_Pole_0	char	次回極対数管理変数
s_MaxSped	int	最大回転数管理変数
s_MaxSped_0	int	次回最大回転数管理変数
s_HighSped	int	目標回転数上限管理変数
s_HighSped_0	int	次回目標回転数上限管理変数
s_LowSped	int	目標回転数下限管理変数
s_LowSped_0	int	次回目標回転数下限管理変数
s_HighCrnt	float	目標電流値上限管理変数
s_HighCrnt_0	float	次回目標電流値上限管理変数
s_LowCrnt	float	目標電流値下限管理変数
s_LowCrnt_0	float	次回目標電流値下限管理変数
s_HighPwm	int	目標PWM値上限管理変数
s_HighPwm_0	int	次回目標PWM値上限管理変数
s_LowPwm	int	目標PWM値下限管理変数
s_LowPwm_0	int	次回目標PWM値下限管理変数
s_RpmPer	float	回転数変化量管理変数
s_RpmPer_0	float	次回回転数変化量管理変数
s_CrntPer	float	電流値変化量管理変数
s_CrntPer_0	float	次回電流値変化量管理変数
s_PwmPer	int	PWM値変化量管理変数
s_PwmPer_0	int	次回PWM値変化量管理変数
s_FreqCutoff	int	カット周波数管理変数
s_FreqCutoff_0	int	次回カット周波数管理変数
s_AdGain	float	A/D変換用ゲイン管理変数
s_AdOffset	int	A/D変換用オフセット管理変数
s_CntInt	unsigned int	割り込みカウンタ
s_CntPidSped	unsigned int	PID用割り込みカウンタ
s_CntIntER	unsigned int	電流マイナ・ループ制御間隔カウンタ
s_DeltaSpedMax	int	単位時間の変化量 速度
s_DeltaCrntMax	float	単位時間の変化量 電流
s_DeltaPwmMax	int	単位時間の変化量 PWM
s_BufCntNew	unsigned int	今回タイム値保存用変数

変数名	型	意 味	
s_PwmResolution	int	分解能管理変数	
s_DeadtimeCMP	unsigned int	デッド・タイム設定値管理変数	
s_BaseCalcRPM	long	回転数計算用基数管理変数	
s_TimeOffset	int	タイマ・オフセット管理変数	
s_FreqCutoffCMP	unsigned int	カットオフ周波数用コンペア値管理変数	
s_AlphaFilt	float	カットオフ周波数用アルファ値管理変数	
s_MaxedFlags	unsigned char	加速度範囲外認識フラグ	
s_StartupMethod	unsigned char	初期起動動作モード	
s_TimeKnee	float	起動用パラメータ格納変数	
s_TimeEnd	float		
s_StartRpm_0	float		
s_StartRpm_1	float		
s_StartRpm_2	float		
s_StartCmt_0	float		
s_StartCmt_1	float		
s_StartCmt_2	float		
s_StartPwm_0	float		
s_StartPwm_1	float		
s_StartPwm_2	float		
s_CycleTime	unsigned int		速度制御間隔用内部比較値
tr_sw[][]	unsigned char		通電パターン情報

(2) sub_mcio.cファイル

変数名	型	意 味	備 考
ad_flag	unsigned char	速度変更フラグ	指定速度変更機能を制限
led_data[]	unsigned char	LED出力データ	LEDに表示する数値

(3) sub_mcio.hファイル

変数名	型	意 味	備 考
sys_flag	unsigned char	制御フラグ	制御状態

4.12 ソース・プログラム

このアプリケーション・ノートで使用するプログラムは、NECエレクトロニクスのホーム・ページのサンプル・プログラムのページ（<http://www.necel.com/micro/ja/designsupports/sampleprogram/index.html>）よりダウンロードしてください。

【発 行】

NECエレクトロニクス株式会社

〒211-8668 神奈川県川崎市中原区下沼部1753

電話（代表）：044(435)5111

—— お問い合わせ先 ——

【ホームページ】

NECエレクトロニクスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.necel.co.jp/>

【営業関係，技術関係お問い合わせ先】

半導体ホットライン

(電話：午前 9:00～12:00，午後 1:00～5:00)

電 話 : 044-435-9494

E-mail : info@necel.com

【資料請求先】

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