

RX ファミリ

永久磁石同期モータのエンコーダベクトル制御 - MCK 用

要旨

本アプリケーションノートはルネサス製 RX26T マイクロコントローラを使用し、エンコーダ付き永久磁石同期モータをベクトル制御で駆動するソフトウェアサンプルプログラムについて説明することを目的としています。なお、本アプリケーションノート対象ソフトウェアにはスマート・コンフィグレータを使用しており、モータ制御に必要なコンポーネントを使用しています。

本アプリケーションノート対象ソフトウェアはあくまで参考用途であり、弊社がこの動作を保証するものではありません。本アプリケーションノート対象ソフトウェアを使用する場合、適切な環境で十分な評価をしたうえで御使用ください。

動作確認デバイス

本アプリケーションノートの対象ソフトウェアの動作確認は下記のデバイスで行っております。

- 使用 MCU :
- RX26T RAM64KB バージョン (R5F526TFCDFF)
- RX26T RAM48KB バージョン (R5F526TACDFM)

対象ソフトウェア

本アプリケーションノートの対象ソフトウェアを下記に示します。

(RX26T RAM64KB バージョン)

- RX26T_MCBA_MCILV1_SPM_ENCD_FOC_CSP_V110 (IDE: CS + version)
- RX26T_MCBA_MCILV1_SPM_ENCD_FOC_E2S_V100 (IDE: e²studio version)

(RX26T RAM48KB バージョン)

- RX26T_MCBC_MCILV1_SPM_ENCD_FOC_CSP_V100 (IDE: CS + version)
- RX26T_MCBC_MCILV1_SPM_ENCD_FOC_E2S_V100 (IDE: e²studio version)

Renesas Flexible Motor Control Kit & RX26T CPU ボード向けエンコーダベクトル制御ソフトウェア

目次

1. 概要	4
2. 開発環境	5
2.1 動作確認環境	5
2.2 ハードウェア仕様	6
3. クイックスタートガイド	9
3.1 サンプルプログラムのダウンロード・書き込み	9
3.2 Analyzer 起動と RMT ファイル	9
3.3 Analyzer 機能用変数一覧	11
3.4 RMW UI 操作	12
3.5 ボード UI 操作	17
4. ソフトウェア	18
4.1 ソフトウェアスペック	18
4.2 ソフトウェア構成	19
4.3 ファイル/フォルダ構成	22
5. 機能	26
5.1 アプリケーション層	26
5.2 マネージャモジュール	38
5.3 電流制御モジュール	58
5.4 変調 (電流制御モジュール)	67
5.5 電圧誤差補償 (電流制御モジュール)	69
5.6 速度制御モジュール	71
5.7 弱め磁束制御 (速度制御モジュール)	78
5.8 外乱トルク・速度推定オブザーバ (速度制御モジュール)	79
5.9 位置制御モジュール	80
5.10 位置プロファイル(位置制御モジュール)	90
5.11 IPD 制御モジュール	94
5.12 センサモジュール (エンコーダ)	97
5.13 ドライバモジュール	110
5.14 スマート・コンフィグレータ設定	115
6. ベクトル制御アルゴリズム	121
6.1 永久磁石同期モータの解析モデル	121
6.2 永久磁石同期モータの d q 軸モデル	122
6.3 ベクトル制御システムとコントローラ	124
7. 試験結果	130
7.1 プログラムサイズ	130
7.2 CPU 負荷率	130
7.3 動作波形	131
8. 参考資料	133

改訂記録 134

1. 概要

本アプリケーションノートはルネサス製マイクロコントローラ(MCU)を使用し、エンコーダ付き永久磁石同期モータをベクトル制御で駆動するサンプルプログラムの使用方法について説明することを目的としています。サンプルプログラムはモータ制御用のキット(Renesas Flexible Motor Control Kit)と組み合わせることで、モータ制御を行うことができます。また、モータ制御開発支援ツール「Renesas Motor Workbench」に対応しており MCU の内部データ確認や、モータ制御のユーザインタフェース(UI)として使用可能です。サンプルプログラムの MCU 機能割り当てや、制御の割り込み負荷状況などを参照頂くことで、使用する MCU の選定やソフトウェア開発の参考としてご活用ください。

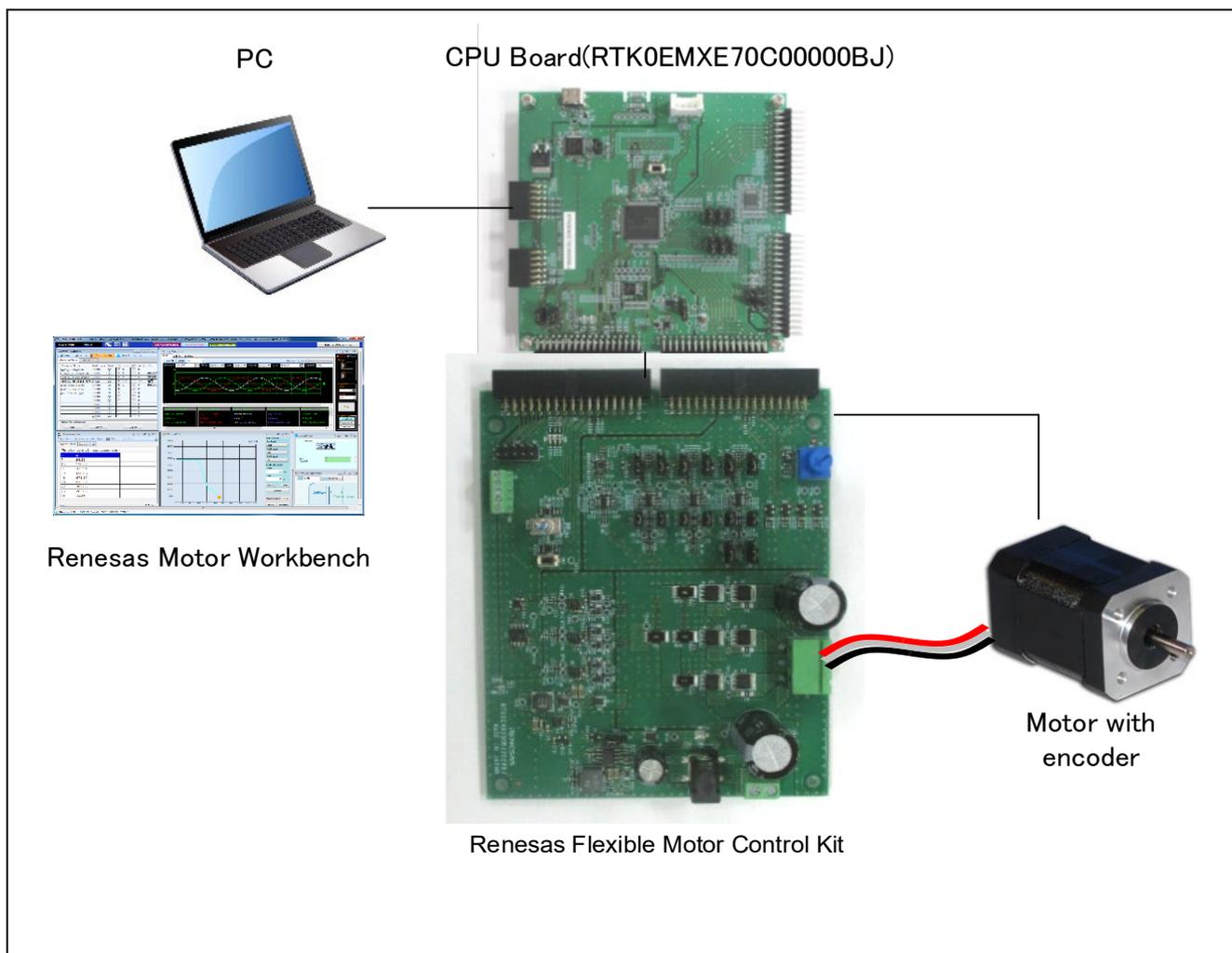


図 1-1 サンプルプログラムを使用した動作環境

2. 開発環境

2.1 動作確認環境

本アプリケーションノート対象ソフトウェアの開発環境を表 2-1、表 2-2 に示します。

表 2-1 ハードウェアの開発環境

分類	使用製品
マイコン / CPU ボード型名	RX26T RAM64KB バージョン (R5F526TFCDFP) / RTK0EMXE70C00000BJ RX26T RAM48KB バージョン (R5F526TACDFM) / RTK0EMXE30C00000BJ
インバータボード	Renesas Flexible Motor Control Kit 同梱 48V 10A BLDC 用インバータボード(RTK0EM0000B12020BJ)
モータ	BLY171D-24V-4000 (Anaheim Automation 社製)
センサ	エンコーダ : AMT102-V (CUI DEVICES 社製)

表 2-2 ソフトウェアの開発環境

IDE バージョン	RX スマート・コンフィグレータ	ツールチェーンバージョン
CS+ : V8.10.00	バージョン 2.18.0	CC-RX : V3.05.00
e ² studio : 2023-07	e ² studio プラグイン版	

ご購入、技術サポートにつきましては、弊社営業及び特約店にお問い合わせください。

2.2 ハードウェア仕様

2.2.1 ハードウェア構成図

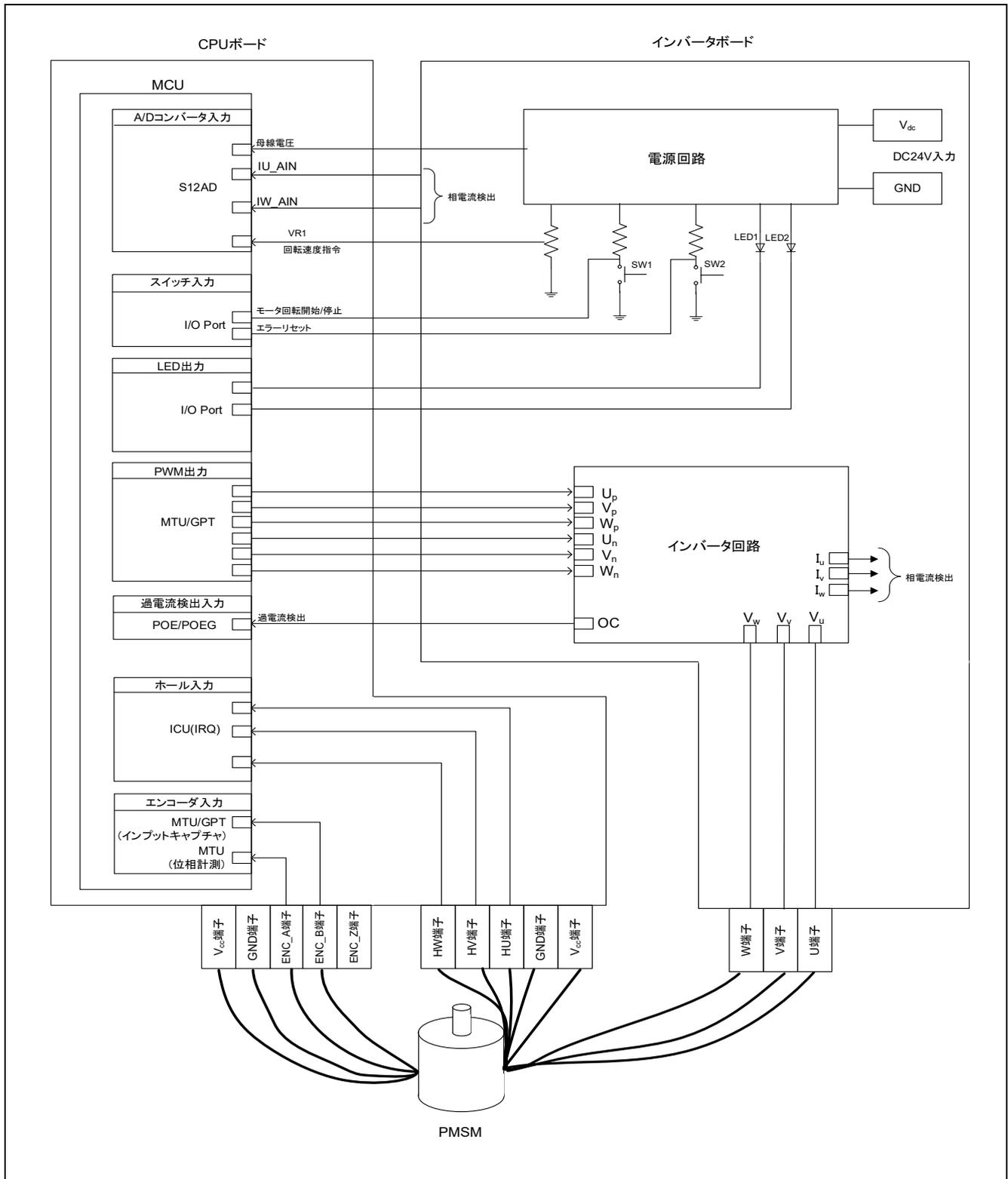


図 2-1 ハードウェア構成図

2.2.2 ボードユーザインタフェース

本システムのボードユーザインタフェース一覧を表 2-3 に示します。

表 2-3 ボードユーザインタフェース

項目	インタフェース部品	機能
回転位置/速度	ボリューム(VR1)	回転位置/速度指令値入力(アナログ値)
START/STOP	トグルスイッチ(SW1)	モータ回転開始/停止指令
ERROR RESET	プッシュスイッチ(SW2)	エラー状態からの復帰指令
LED1	オレンジ色 LED	・モータ回転時：点灯 ・停止時：消灯
LED2	オレンジ色 LED	・エラー検出時：点灯 ・通常動作時：消灯
RESET	プッシュスイッチ(SW1-CPU ボード上)	システムリセット

2.2.3 周辺機能

本システムで使用する入出力機能と周辺機能の割り当てを表 2-4 に示します。サンプルプログラムでは、スマート・コンフィグレータを使って周辺機能の設定を行っています。詳細については、5.14 を参照してください。

表 2-4 入出力機能と周辺機能

機能	周辺機能	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
インバータ母線電圧測定	S12AD	
回転位置/速度指令値入力用(アナログ値)	S12AD	
START/STOP トグルスイッチ	I/O Port (Input)	
LED1 点灯/消灯制御	I/O Port (output)	
LED2 点灯/消灯制御	I/O Port (output)	
U 相電流測定	S12AD	
W 相電流測定	S12AD	
PWM 出力 (U _p) / “High” アクティブ	MTU	
PWM 出力 (V _p) / “High” アクティブ	MTU	
PWM 出力 (W _p) / “High” アクティブ	MTU	
PWM 出力 (U _n) / “High” アクティブ	MTU	
PWM 出力 (V _n) / “High” アクティブ	MTU	
PWM 出力 (W _n) / “High” アクティブ	MTU	
ホール U 相入力	ICU (IRQ)	
ホール V 相入力	ICU (IRQ)	
ホール W 相入力	ICU (IRQ)	
エンコーダ A 相入力	MTU	GPT
エンコーダ B 相入力	MTU	GPT
過電流検出時の PWM 緊急停止入力	POE	

3. クイックスタートガイド

本章は Renesas Flexible Motor Control Kit とサンプルプログラムを使用してモータを駆動するためのクイックスタートガイドです。Renesas Flexible Motor Control Kit のボード設定、接続に関しては MCK-RX26T ユーザーズマニュアル (R12UZ0111) を参照ください。また、Renesas Motor Workbench (RMW) の使用方法詳細については、Renesas Motor Workbench のユーザーズマニュアル (R21UZ0004) を参照ください。

3.1 サンプルプログラムのダウンロード・書き込み

弊社 WEB サイトからダウンロードしたサンプルプログラムを、IDE や Renesas Flash Programmer を使用して CPU ボードの MCU に書き込んでください。プログラムの書き込み方法は IDE 及び Renesas Flash Programmer の取扱説明書を参照してください。

3.2 Analyzer 起動と RMT ファイル

モータ制御開発支援ツール「Renesas Motor Workbench」をユーザインタフェース（回転／停止指令、回転速度指令等）として使用します。モータ制御開発支援ツール「Renesas Motor Workbench」は弊社 WEB サイトより入手してください。

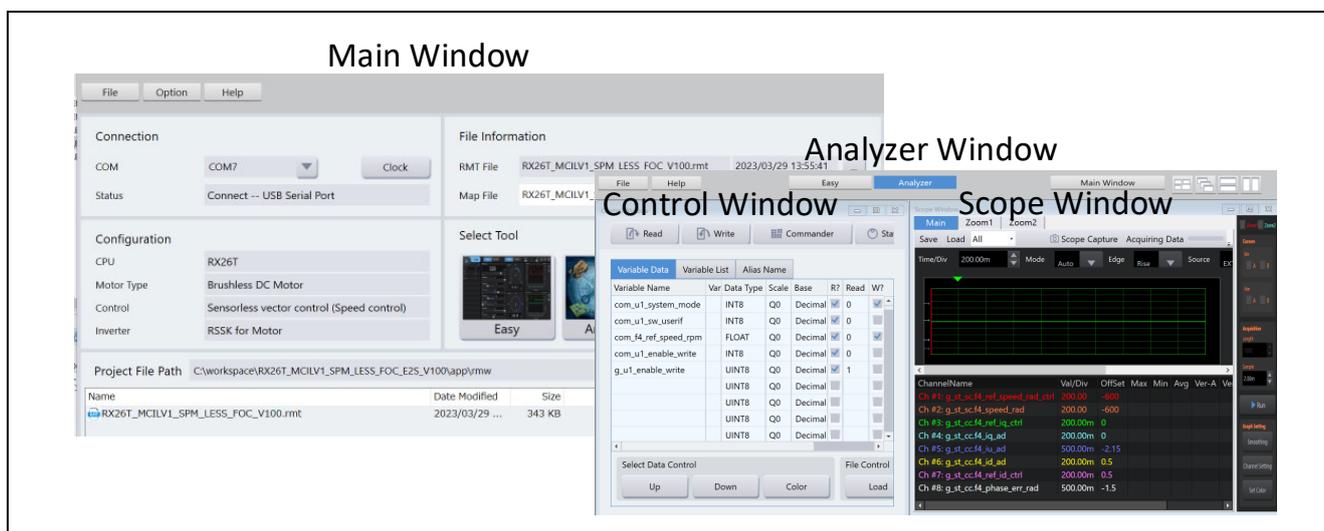


図 3-1 Renesas Motor Workbench 外観

モータ制御開発支援ツール「Renesas Motor Workbench」の使い方

- ツールアイコン  をクリックしツールを起動します。
- Main Panel の MENU バーから、[File] → [Open RMT File(O)]を選択します。
プロジェクトフォルダの“rmw”フォルダ内にある RMT ファイルを読み込みます。
- “Connection”の COM で接続されたキットの COM を選択します。
- “Select Tool”画面の“Analyzer”ボタンをクリックし、Analyzer 機能画面を表示します。
- “RMW UI 操作”を元にモータを駆動させます。（詳細は 3.4 を参照ください。）

RMT ファイルとは

- RMT ファイルとは、RMW にて操作/設定した環境情報を保存したファイルです。
- RMT ファイルに環境情報を保存することで、以降は RMT ファイルを呼び出して同じ環境を復元できます。
- プログラムのアドレス情報が変更された場合は、プログラムのビルドで生成された Map ファイルの読み込みを行い、RMT ファイルを再度保存してください。

3.3 Analyzer 機能用変数一覧

RMW UI 使用時の入力用変数一覧を表 3-1 に示します。なお、これらの変数への入力値は com_u1_enable_write に g_u1_enable_write と同じ値を書き込んだ場合にモータモジュール内の対応する変数へ反映され、モータ制御に使用されます。ただし、(*)が付けられた変数は com_u1_enable_write に依存しません。

表 3-1 Analyzer 機能主要入力用変数一覧

Analyzer 機能入力用変数名	型	内容
com_u1_sw_userif (*)	uint8_t	ユーザインタフェーススイッチ 0 : RMW UI 使用 (default) 1 : ボード UI 使用
com_u1_system_mode (*)	uint8_t	ステート管理 0 : ストップモード 1 : ランモード 3 : リセット
com_u1_ctrl_loop_mode	uint8_t	制御ループの切り換え 0 : 位置制御 1 : 速度制御 (default)
com_s2_ref_position_deg (*)	int16_t	位置指令値 (機械角) [度]
com_s2_ref_speed_rpm (*)	int16_t	速度指令値 (機械角) [rpm]
com_u1_enable_write	uint8_t	ユーザ入力用変数書き換え許可 g_u1_enable_write と変数一致で入力データ反映

次にエンコーダ位置/速度制御の駆動評価を行う際に観測することの多い主要な構造体変数の一覧を表 3-2 に示します。Analyzer 機能で波形表示する際や変数の値を読み込む際に参考にしてください。一覧にない変数の詳細については 5.1.5 を参照してください。

表 3-2 エンコーダ位置/速度制御主要変数一覧

エンコーダ位置/速度制御主要変数名	型	内容
g_st_encoder_vector.u2_error_status	uint16_t	エラーステータス
g_st_cc.f4_id_ref	float	d 軸電流指令値 [A]
g_st_cc.f4_id_ad	float	d 軸電流検出値 [A]
g_st_cc.f4_iq_ref	float	q 軸電流指令値 [A]
g_st_cc.f4_iq_ad	float	q 軸電流検出値 [A]
g_st_cc.f4_iu_ad	float	U 相電流検出値 [A]
g_st_cc.f4_iv_ad	float	V 相電流検出値 [A]
g_st_cc.f4_iw_ad	float	W 相電流検出値 [A]
g_st_cc.f4_vd_ref	float	d 軸電圧指令値 [V]
g_st_cc.f4_vq_ref	float	q 軸電圧指令値 [V]
g_st_cc.f4_refu	float	U 相電圧指令値 [V]
g_st_cc.f4_refv	float	V 相電圧指令値 [V]
g_st_cc.f4_refw	float	W 相電圧指令値 [V]
g_st_sc.f4_ref_speed_rad_ctrl	float	速度指令値 (機械角) [rad/s]
g_st_sc.f4_speed_rad	float	速度検出値 (機械角) [rad/s]
g_st_pc.f4_ref_pos_rad_ctrl	float	位置指令値 (機械角) [rad]
g_st_pc.f4_pos_rad	float	位置検出値 (機械角) [rad]

3.4 RMW UI 操作

3.4.1 Analyzer 操作例

Analyzer 機能を使用し、モータを操作する例を以下に示します。操作は、“Control Window”で行います。“Control Window”の詳細は、「Renesas Motor Workbench ユーザーズマニュアル」を参照してください。

初期状態では、制御ループは速度制御となっています。以下を参考に、操作を実施してください。

(a) モータを回転させる

- ① “com_u1_mode_system”、“com_s2_ref_speed_rpm”の[W?]欄に“チェック”が入っていることを確認する。
- ② 指令回転速度を“com_s2_ref_speed_rpm”の[Write]欄に入力する。
- ③ “com_u1_mode_system”の[Write]欄に“1”を入力する。
- ④ “Write”ボタンを押す。

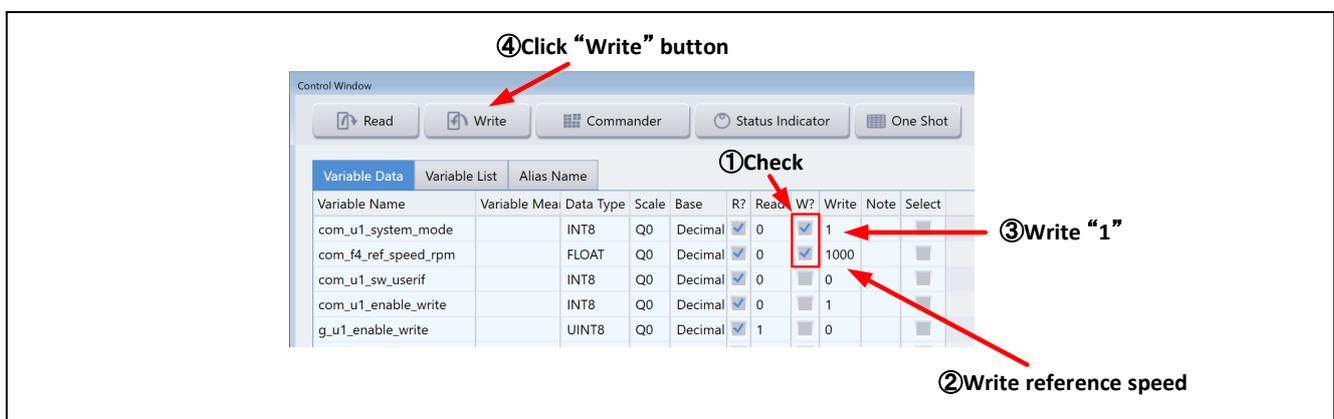


図 3-2 モータ回転の手順

(b) モータを停止させる

- ① “com_u1_mode_system”の[Write]欄に“0”を入力する。
- ② “Write”ボタンを押す。



図 3-3 モータ停止の手順

(c) 止まってしまった（エラー）場合の処理

- ① “com_u1_mode_system”の[Write]欄に“3”を入力する。
- ② “Write”ボタンを押す。



図 3-4 エラー解除の手順

3.4.2 User Button 機能操作例

User Button 機能を使用し、モータを操作する例を以下に示します。例として記載するユーザボタンはサンプルプログラムの RMT ファイルに含まれています。

- モータを位置制御で駆動する／停止する

図 3-5 のように設定することで、ボタンを押すごとに駆動と停止が切り替わります。

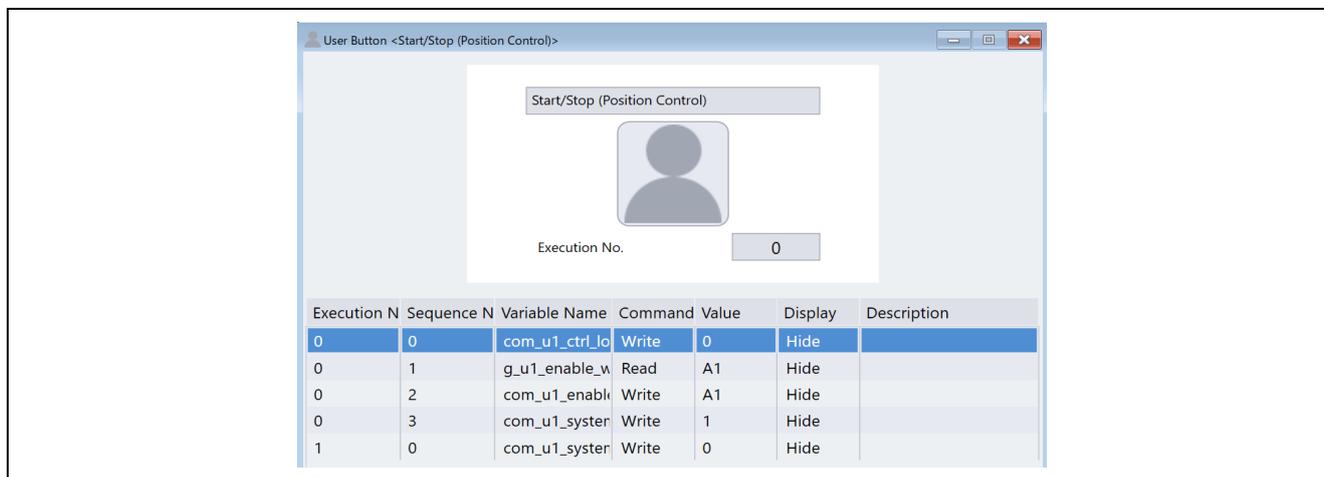


図 3-5 モータの駆動／停止

- 位置指令を変更する

図 3-6 のように設定することで、位置指令を入力し、ボタンを押すことで位置が変更できます。

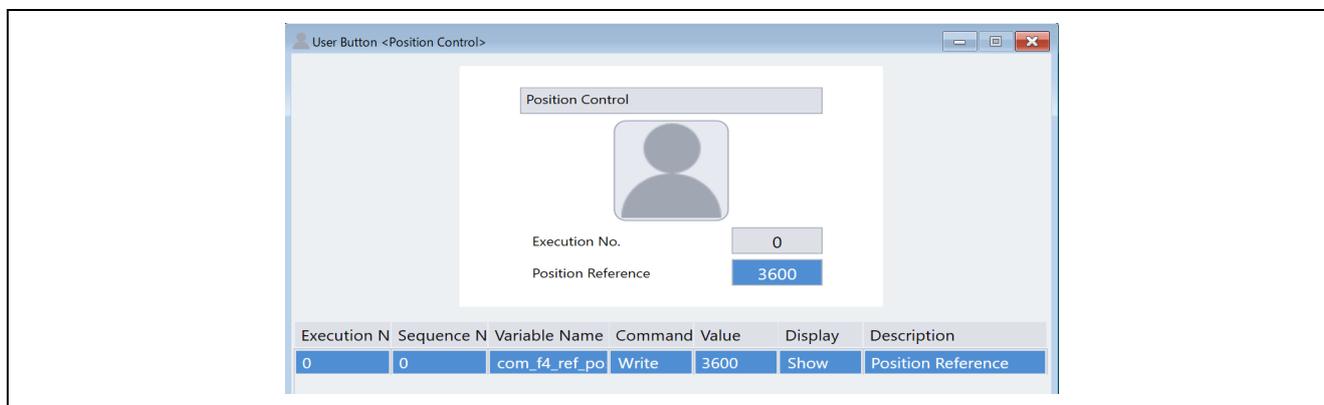


図 3-6 位置指令の変更

- モータを速度制御で駆動する／停止する
図 3-7 のように設定することで、ボタンを押すごとに駆動と停止が切り替わります。

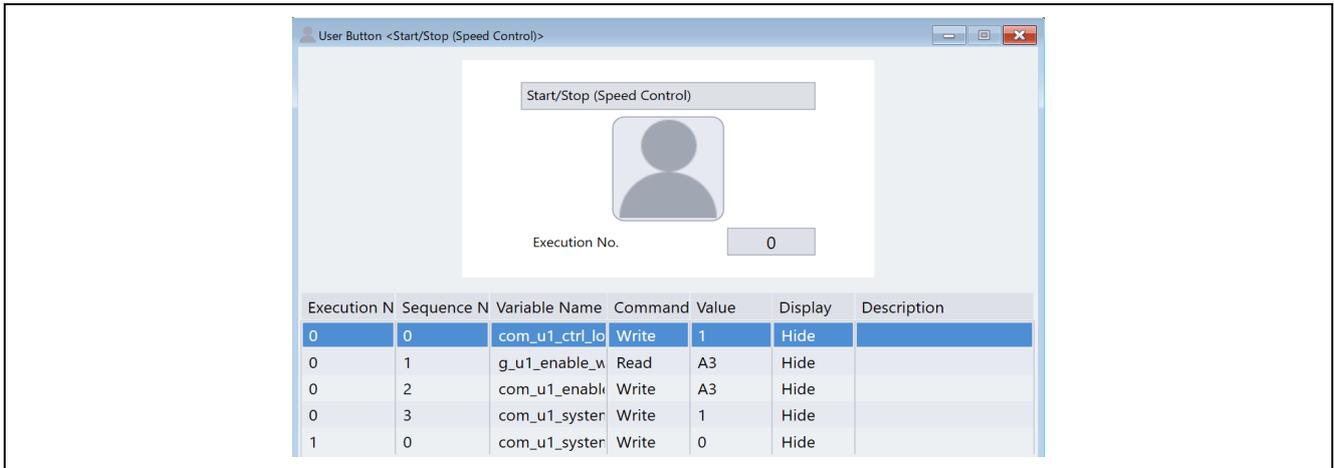


図 3-7 モータの駆動／停止

- 速度指令を変更する
図 3-8 のように設定することで、速度指令を入力し、ボタンを押すことで速度指令が変更できます。

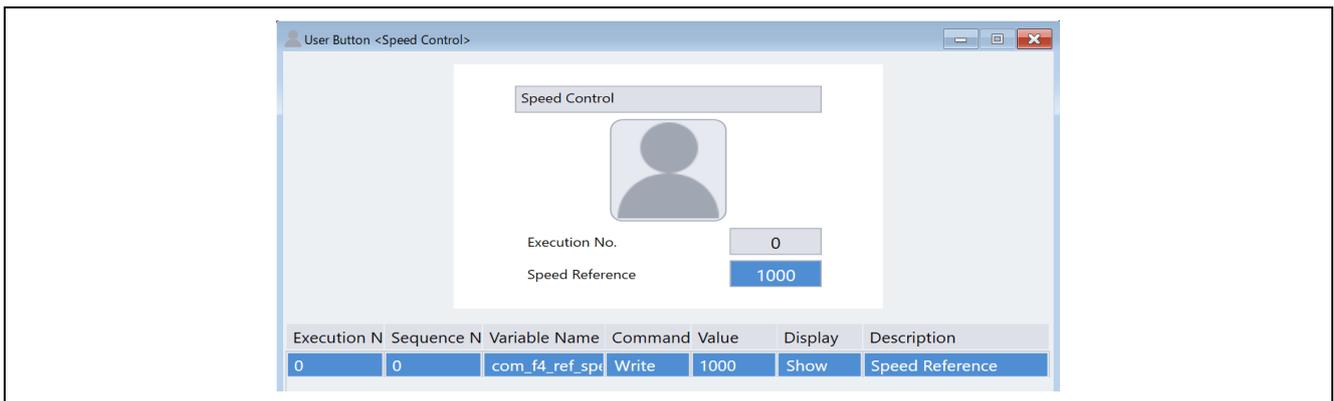


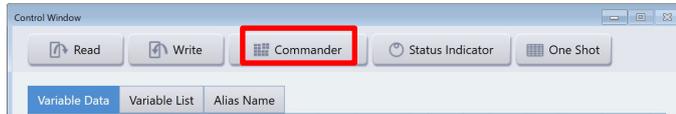
図 3-8 速度指令の変更

3.4.3 コマンダー機能操作例

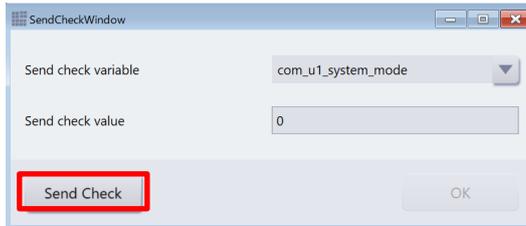
コマンダー機能を使用した位置制御：

(Commander の起動)

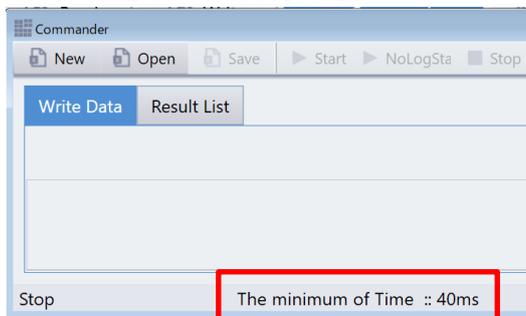
- ① Control Window の“Commander”ボタンを押す。



- ② Commander ウィンドウが立ち上がるので“Send Checker”ボタンを押し、データの送信速度を確認する。

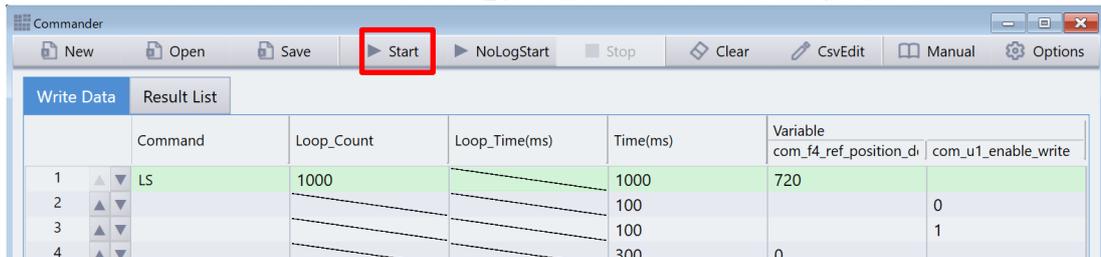


最小の転送時間が表示される。



- ③ Open ボタンを押して“Position_test.csv”を読み込む。位置制御モードに設定し、com_u1_mode_system に“1”を書き込み<Write>ボタンを押し、ランモードにする。モータが位置決め制御を開始する。

- ④ Commander ウィンドウの“Start”ボタンを押すと、シーケンス動作が開始する。



3.5 ボード UI 操作

3.5.1 ユーザインタフェースの切り替え

本サンプルプログラムは、RMW UI をユーザインタフェースとして設定しています。ボード UI へ変更する場合は、以下の手順に従って切り替えてください。

“com_u1_sw_userif” の[W?]欄に “チェック” が入っていることを確認し、[Write]欄に “1” を入力する。“Write” ボタンを押す。

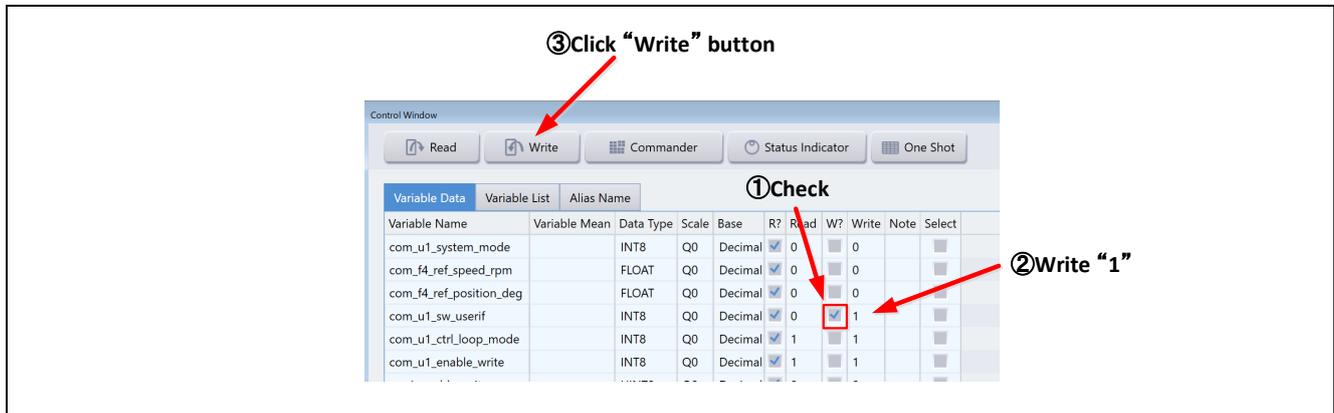


図 3-9 UI の切り替えの手順

3.5.2 モータ起動／停止

ボード UI 時はモータの起動と停止をインバータボードの SW1 からの入力(ボード UI)によって制御します。SW1 には汎用ポートが割り当てられ、メイン・ループ内で、端子を読み、“ON”レベルのときスタートスイッチが押されていると判断し、逆に“OFF”レベルのときはモータを停止すると判断します。

3.5.3 モータ回転位置／速度指令値

モータの回転位置／速度指令値はインバータボードの VR1 の出力値（アナログ値）を A/D 変換することによって決定します。A/D 変換された VR1 の値は、以下の表のように、回転位置／速度指令値として使用します。

表 3-3 回転位置/速度指令値の変換比

項目	変換比 (指令値 : A/D 変換値)	
	回転位置指令値	CW
	CCW	0 [度]~-180 [度] : 0800H~0FFFH
回転速度指令値	CW	0 [rpm]~4000 [rpm] : 07FFH~0000H
	CCW	0 [rpm]~-4000 [rpm] : 0800H~0FFFH

4. ソフトウェア

4.1 ソフトウェアスペック

本システムのソフトウェアの基本仕様を下記に示します。

表 4-1 エンコーダベクトル制御ソフトウェア基本仕様

項目	内容	
制御方式	ベクトル制御	
モータ制御開始/停止	SW1 のレベルにより判定 (“ON” : 制御開始 “OFF” : 停止) または RMW から入力	
回転子磁極位置検出	インクリメンタルエンコーダ (A 相、B 相)、ホールセンサ (UVW 相)	
入力電圧	DC 24V	
キャリア周波数 (PWM)	20 [kHz]、キャリア周期 : 50 [μ s]	
デッドタイム	2 [μ s]	
制御周期 (電流)	50 [μ s]	
制御周期 (速度・位置)	500 [μ s]	
位置指令値管理	ボード UI	位置指令値の作成 : VR1 による直接入力 (入力範囲) -180° ~ 180°
	RMW UI	位置指令値の作成 : 速度台形波方式による位置プロファイル (入力範囲) -32768° ~ 32767° (速度制限) CW / CCW : -4000 ~ 4000 [rpm]
速度指令値管理	CW : 0 [rpm] to 4000 [rpm] CCW : 0 [rpm] to -4000 [rpm]	
位置分解能	0.09° (エンコーダパルス : 1000 [p/r]、4 通倍時 4000 [cpr])	
位置の不感帯*1	エンコーダ \pm 1 カウント (\pm 0.09°)	
各制御系固有周波数	電流制御系 : 300 Hz 速度制御系 : 12 Hz 位置制御系 : 4 Hz	
コンパイラ最適化設定	最適化レベル	2 (-optimize = 2) (デフォルト設定)
	最適化方法	コード・サイズ重視の最適化 (-size) (デフォルト設定)
保護停止処理	以下のいずれかの条件の時、モータ制御信号出力 (6 本) を非アクティブにする <ol style="list-style-type: none"> 1. 各相の電流が 2.69 [A] を超過 (50 [μs] 毎に監視) 2. インバータ母線電圧が 60 [V] を超過 (50 [μs] 毎に監視) 3. インバータ母線電圧が 8 [V] 未満 (50 [μs] 毎に監視) 4. 回転速度が 4500 [rpm] を超過 (50 [μs] 毎に監視) 5. ホールセンサのパターンエラー(始動時) 外部からの過電流検出信号 (POE) 及び出力短絡を検出した場合、PWM 出力端子をハイインピーダンスにする	

【注】 1. 位置決め時のハンチング等を防ぐため、不感帯を設けています。

4.2 ソフトウェア構成

サンプルプログラムはアプリケーション層とモータモジュール、スマート・コンフィグレータで構成されています。ユーザが操作するアプリケーション層から指示を受け、モータモジュールが制御を行います。HW 層への出力はスマート・コンフィグレータを介して行っております。

4.2.1 全体構成

ソフトウェアの全体構成を図 4-1 に示します。

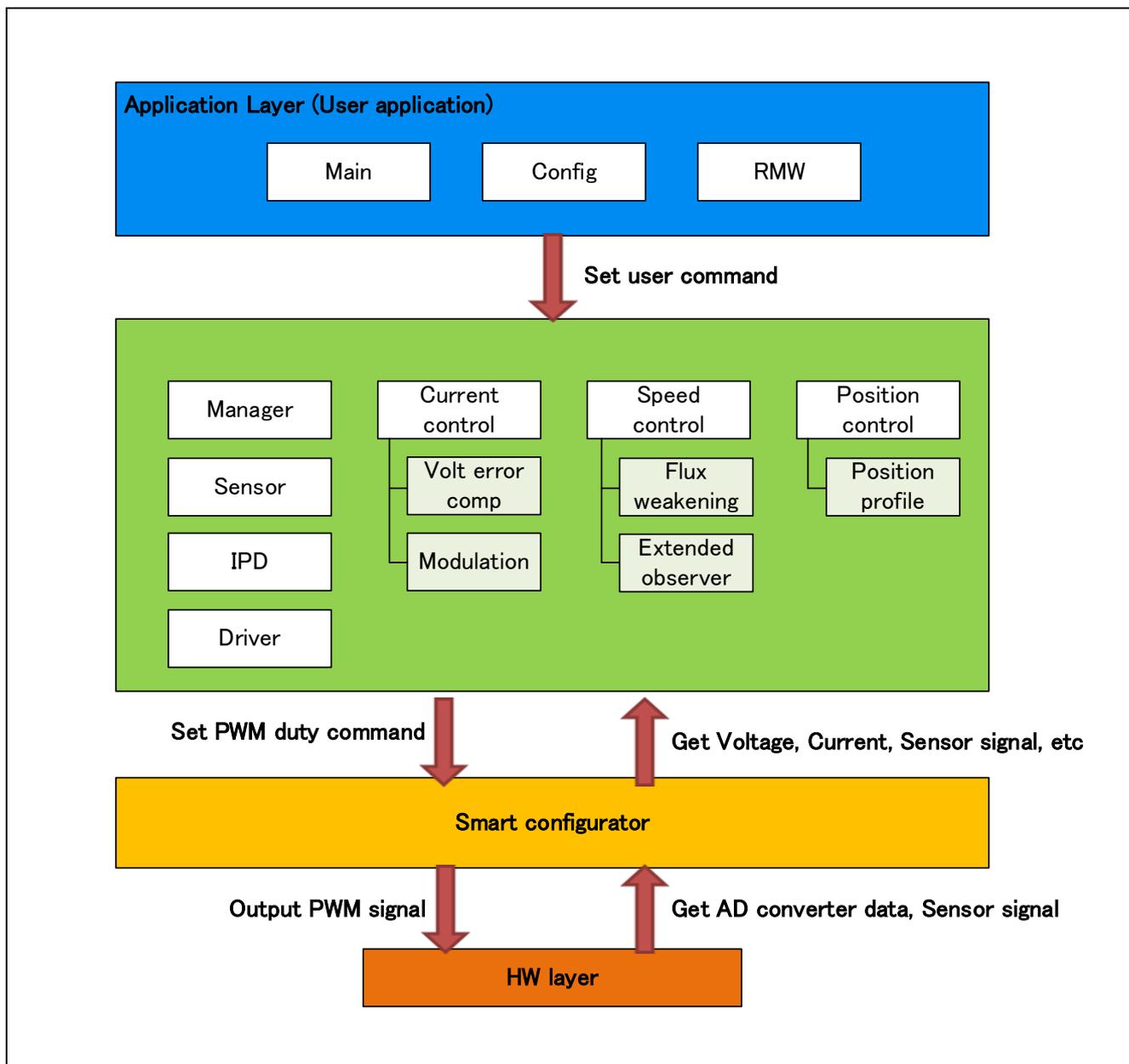


図 4-1 モータ制御ソフトウェアの全体構成

4.2.2 モータモジュールの構成

モータモジュールの構成を図 4-2 に示します。また、各モジュールの概要を表 4-2 に示します。マネージャモジュールが他のモジュールとのインタフェースになっており、適切なモジュールにデータの取得・設定を行います。

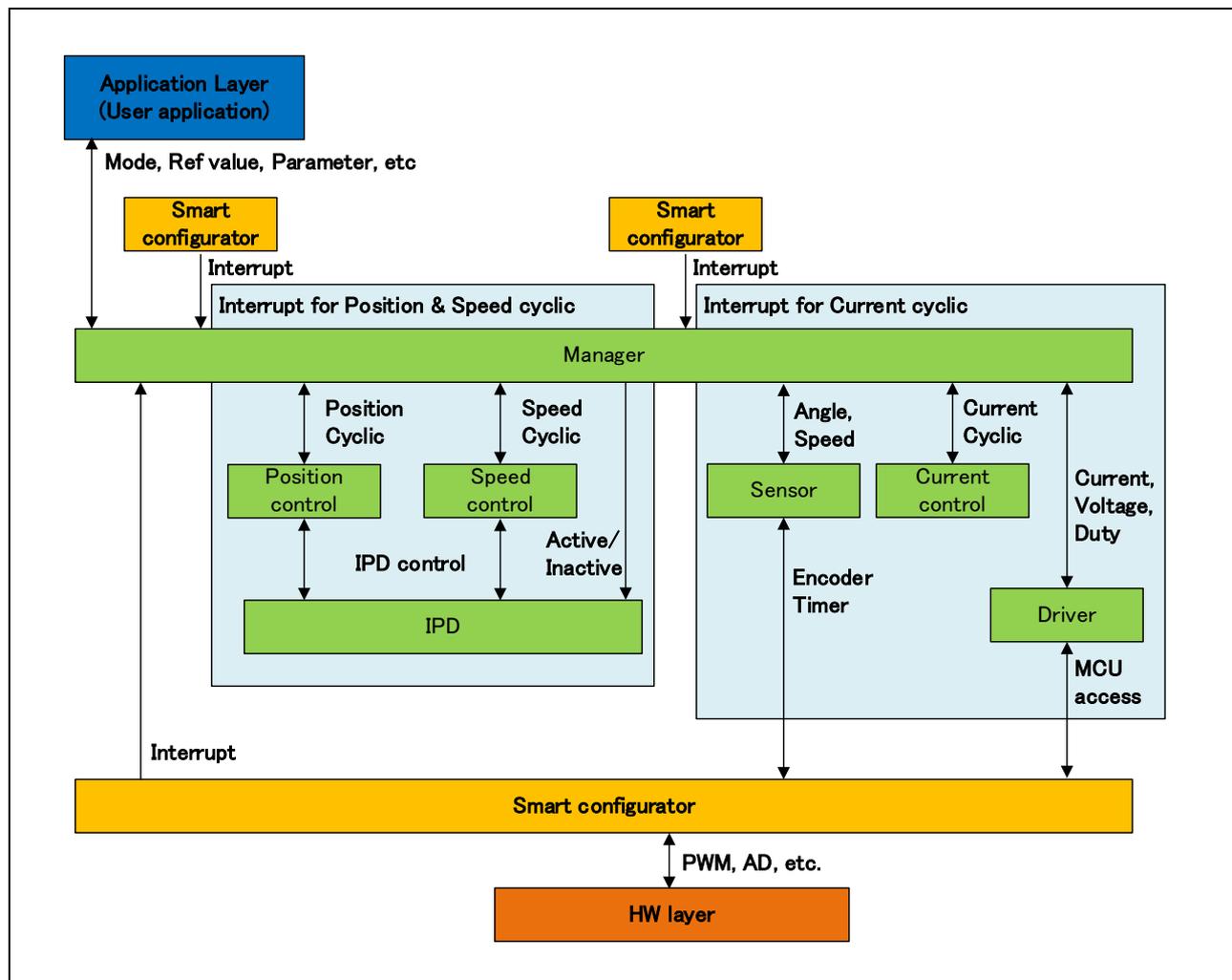


図 4-2 モータモジュールの構成

表 4-2 モジュール概要

モジュール	説明	章番号
アプリケーション層	メイン処理、ユーザの使用する領域	5.1
マネージャモジュール	サンプルプログラム全体の管理と各モジュールのインタフェース	5.2
電流制御モジュール	電流制御に関するモジュール	5.3
速度制御モジュール	速度制御に関するモジュール	5.6
位置制御モジュール	位置制御に関するモジュール	5.9
IPD 制御モジュール	IPD 制御に関するモジュール	5.11
センサモジュール	センサ信号から位置・速度情報を取得モジュール	5.12
ドライバモジュール	スマート・コンフィグレータとの接続に関するモジュール	5.13
スマート・コンフィグレータ層	HW レイヤとの接続に関するモジュール	5.14

4.3 ファイル/フォルダ構成

サンプルプログラムのフォルダとファイル構成を表 4-3 に示します。

表 4-3 ファイル・フォルダ構成

フォルダ	サブフォルダ	ファイル	備考
app	main	r_app_main.c/h	ユーザメイン関数
	rmw	r_app_rmw.c/h	RMW の Analyzer UI 関連関数定義
		r_app_rmw_interrupt.c	RMW の割り込み関数定義
		ICS2_RX26T.lib/h	RMW の通信用ライブラリ
	board_ui	r_app_board_ui.c/h	ボード UI 関連関数定義
		r_app_board_ui_ctrl.h	MCU 依存のボード UI 関数定義
		r_app_board_ui_ctrl_rx26t_mcilv1.c	MCU 依存のボード UI 関数定義
cfg	r_app_control_cfg.h	アプリ層のコンフィグレーション定義	
motor_module	encoder_vector_rx	r_motor_encoder_vector_action.c	アクション関数定義
		r_motor_encoder_vector_api.c/h	マネージャモジュールの API 関数定義
		r_motor_encoder_vector_manager.c/h	マネージャモジュールのローカル関数定義
		r_motor_encoder_vector_protection.c/h	保護機能の関数定義
		r_motor_encoder_vector_statemachine.c/h	状態遷移関連の関数定義
	current_rx	r_motor_current_api.c/h	電流制御モジュールの API 関数定義
		r_motor_current.c/h	電流制御モジュールのローカル関数定義
		r_motor_current_modulation.c/h	変調モジュールの関数定義
		r_motor_current_volt_err_comp.lib/h	電圧誤差補償モジュールの関数定義
		r_motor_current_pi_gain_calc.c	電流制御モジュールの制御ゲイン算出関数定義
	speed_rx	r_motor_speed_api.c/h	速度制御モジュールの API 関数定義
		r_motor_speed.c/h	速度制御モジュールのローカル関数定義
		r_motor_speed_fluxwkn.lib/h	弱め磁束モジュールの関数定義
		r_motor_speed_extobserver.lib/h	外乱オブザーバモジュールの関数定義
		r_motor_speed_pi_gain_calc.c	速度制御モジュールの制御ゲイン算出関数定義
	position_rx	r_motor_position_api.c/h	位置制御モジュールの API 関数定義
		r_motor_position.c/h	位置制御モジュールのローカル関数定義
		r_motor_position_profiling.c/h	位置制御指令値作成の関数定義
		r_motor_position_gain_calc.c	位置制御モジュールの制御ゲイン算出関数定義
	ipd_rx	r_motor_ipd_api.lib/h	IPD モジュールの API 関数定義
	driver_rx	r_motor_driver.c/h	ドライバモジュールの関数定義
	sensor_rx	r_motor_sensor_api.c/h	センサモジュールの API 関数定義
		r_motor_sensor_encoder.c/h	センサモジュールのエンコーダ処理関数定義
		r_motor_sensor_hall.c/h	センサモジュールのホールセンサ処理関数定義
	general	r_motor_filter.c/h	汎用フィルタ関数定義
		r_motor_pi_control.c/h	PI 制御関数定義
		r_motor_common.h	共通定義
	cfg	r_motor_inverter_cfg.h	インバータのコンフィグレーション定義
		r_motor_module_cfg.h	制御モジュールのコンフィグレーション定義
		r_motor_targetmotor_cfg.h	モータのコンフィグレーション定義

フォルダ	サブフォルダ	ファイル	備考
motor_module	cfg	r_mtr_control_parameter.h	RMW の Tuner 機能によるチューニング結果*1 (制御パラメータ定義)
		r_mtr_motor_parameter.h	RMW の Tuner 機能によるチューニング結果*1 (モータパラメータ定義)
QE_Motor			QE for Motor 生成ファイル
src	smc_gen	別表	スマート・コンフィグレータで生成されたドライバ及び API

【注】 1. QE for Motor から Tuning を実行した場合に更新されます。

スマート・コンフィグレータを使用することで、周辺機能ドライバを簡単に生成することができます。

スマート・コンフィグレータは、プロジェクトで使用するマイクロコントローラ、周辺機能、端子機能などの設定情報をプロジェクト・ファイル (*.scfg) に保存し、参照します。本ソフトウェアの周辺機能設定を確認する場合、以下のファイルを参照してください。

“RX26T_xxx_MCILV1_SPM_ENCD_FOC_yyy_Vzzz.scfg”

(xxx : MCBA は R RX26T RAM64KB バージョン向け、MCBC は RX26T RAM48KB バージョン向けを意味します。yyy : CSP は CS+版、E2S は e² studio 版を意味します。zzz : リビジョン番号)

スマート・コンフィグレータで生成したフォルダとファイル構成を下記に示します。

表 4-4 スマート・コンフィグレータのフォルダ・ファイル構成(RX26T RAM64KB バージョン)

フォルダ	サブフォルダ	サブフォルダ 2	ファイル	備考
src	smc_gen	Config_ICU	Config_ICU.c/h	ホール割り込みコントローラ関連関数定義
			Config_ICU_user.c	ホール割り込みコントローラ関連ユーザ関数定義
		Config_S12AD2	Config_S12AD2.c/h	12bitADC 関連関数定義
			Config_S12AD2_user.c	12bitADC 関連ユーザ関数定義
		Config_PORT	Config_PORT.c/h	ポート関連関数定義
			Config_PORT_user.c	ポート関連ユーザ関数定義
		Config_CMT0	Config_CMT0.c/h	制御周期用 CMT 関連関数定義
			Config_CMT0_user.c	制御周期用 CMT 関連ユーザ関数定義
		Config_GPT3	Config_GPT3.c/h	速度計測用 GPT3 関連関数定義
			Config_GPT3_user.c	速度計測用 GPT3 関連ユーザ関数定義
		Config_IWDT	Config_IWDT.c/h	IWDT 関連関数定義
			Config_IWDT_user.c	IWDT 関連ユーザ関数定義
		Config_MTU0	Config_MTU0.c/h	速度計測用 MTU 関連関数定義
			Config_MTU0_user.c	速度計測用 MTU 関連ユーザ関数定義
		Config_MTU1	Config_MTU1.c/h	位相係数用 MTU 関連関数定義
			Config_MTU1_user.c	位相係数用 MTU 関連ユーザ関数定義
		Config_POE	Config_POE.c/h	POE 関連関数定義
			Config_POE_user.c	POE 関連ユーザ関数定義
		Config_MOTOR	Config_MOTOR.c/h	モータコンポーネント関連関数定義
			Config_MOTOR_user.c	モータコンポーネント関連ユーザ関数定義

表 4-5 スマート・コンフィグレータのフォルダ・ファイル構成(RX26T RAM48KB バージョン)

フォルダ	サブフォルダ	サブフォルダ 2	ファイル	備考
src	smc_gen	Config_ICU	Config_ICU.c/h	ホール割り込みコントローラ関連関数定義
			Config_ICU_user.c	ホール割り込みコントローラ関連ユーザ関数定義
		Config_S12AD2	Config_S12AD2.c/h	12bitADC 関連関数定義
			Config_S12AD2_user.c	12bitADC 関連ユーザ関数定義
		Config_PORT	Config_PORT.c/h	ポート関連関数定義
			Config_PORT_user.c	ポート関連ユーザ関数定義
		Config_CMT0	Config_CMT0.c/h	制御周期用 CMT 関連関数定義
			Config_CMT0_user.c	制御周期用 CMT 関連ユーザ関数定義
		Config_CMTW0	Config_CMTW0.c/h	速度計測用 CMTW0 関連関数定義
			Config_CMTW0_user.c	速度計測用 CMTW0 関連ユーザ関数定義
		Config_IWDT	Config_IWDT.c/h	IWDT 関連関数定義
			Config_IWDT_user.c	IWDT 関連ユーザ関数定義
		Config_MTU0	Config_MTU0.c/h	速度計測用 MTU 関連関数定義
			Config_MTU0_user.c	速度計測用 MTU 関連ユーザ関数定義
		Config_ELC	Config_ELC.c/h	速度計測用 MTU へのイベントリンク関連関数定義
			Config_ELC_user.c	速度計測用 MTU へのイベントリンク関連ユーザ関数定義
		Config_GPT5	Config_GPT5.c/h	位相係数用 GPT 関連関数定義
			Config_GPT5_user.c	位相係数用 GPT 関連ユーザ関数定義
		Config_POE	Config_POE.c/h	POE 関連関数定義
			Config_POE_user.c	POE 関連ユーザ関数定義
Config_MOTOR	Config_MOTOR.c/h	モータコンポーネント関連関数定義		
	Config_MOTOR_user.c	モータコンポーネント関連ユーザ関数定義		

上記各表の他に、スマート・コンフィグレータ使用時に 4 つのフォルダが自動生成されます。

r_bsp : 様々な BSP (BSP : Board Support Package) ファイルを含みます。詳細は“r_bsp”フォルダ内の“readme.txt”ファイルを参照してください。

general : スマート・コンフィグレータ生成ドライバで共通で使用される様々なファイルを含みます。

r_config : MCU パッケージ、クロック、割り込み、R_xxx_Open の名前を持つドライバ初期化関数のコンフィグレーションヘッダファイルを含みます。

r_pincfg : ピン設定に関する様々なファイルを含みます。

5. 機能

5.1 アプリケーション層

アプリケーション層はユーザインタフェース(UI)の選択と RMW を使用したモータモジュールに対する制御の指令値設定や制御モジュールのパラメータ更新を行っています。サンプルプログラムでは、インバータボードのスイッチとボリュームを使用してモータを駆動する方式(ボード UI)と RMW を使用してモータを駆動する方式(RMW UI)があるため、それらの設定及び処理を行っています。また、これらの UI からモータの駆動/停止や、制御の指令値設定などを行っています。

5.1.1 機能

アプリケーション層で設定している機能一覧を表 5-1 に示します。

表 5-1 アプリケーション層の機能一覧

機能	説明
メイン処理	ユーザの指令に対してシステムを有効 / 無効に設定します。
UI 処理	ボード UI と RMW の UI の選択、切り替えを行います。
ボード UI 処理	位置制御や速度制御の指令値の取得・設定を行います。
RMW の UI 処理	速度・位置情報の指令値含むパラメータの取得・設定を行います。

5.1.2 モジュール構成図

モジュール構成図を図 5-1 に示します。

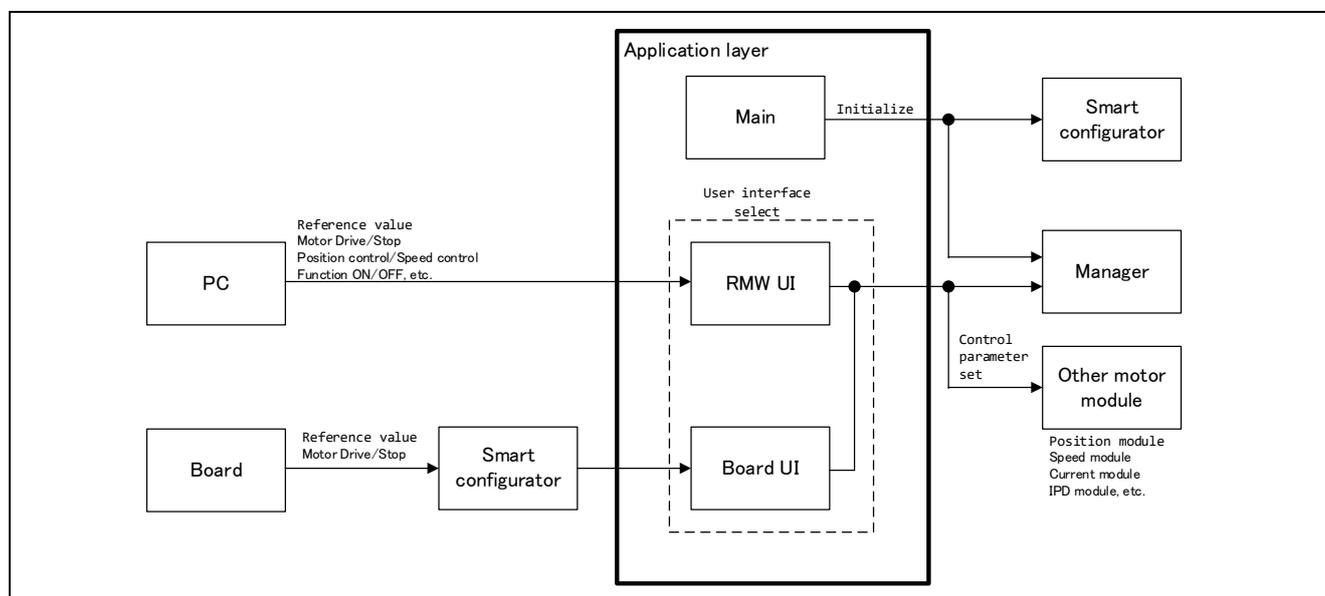


図 5-1 アプリケーション層の構成図

5.1.3 フローチャート

5.1.3.1 メイン処理

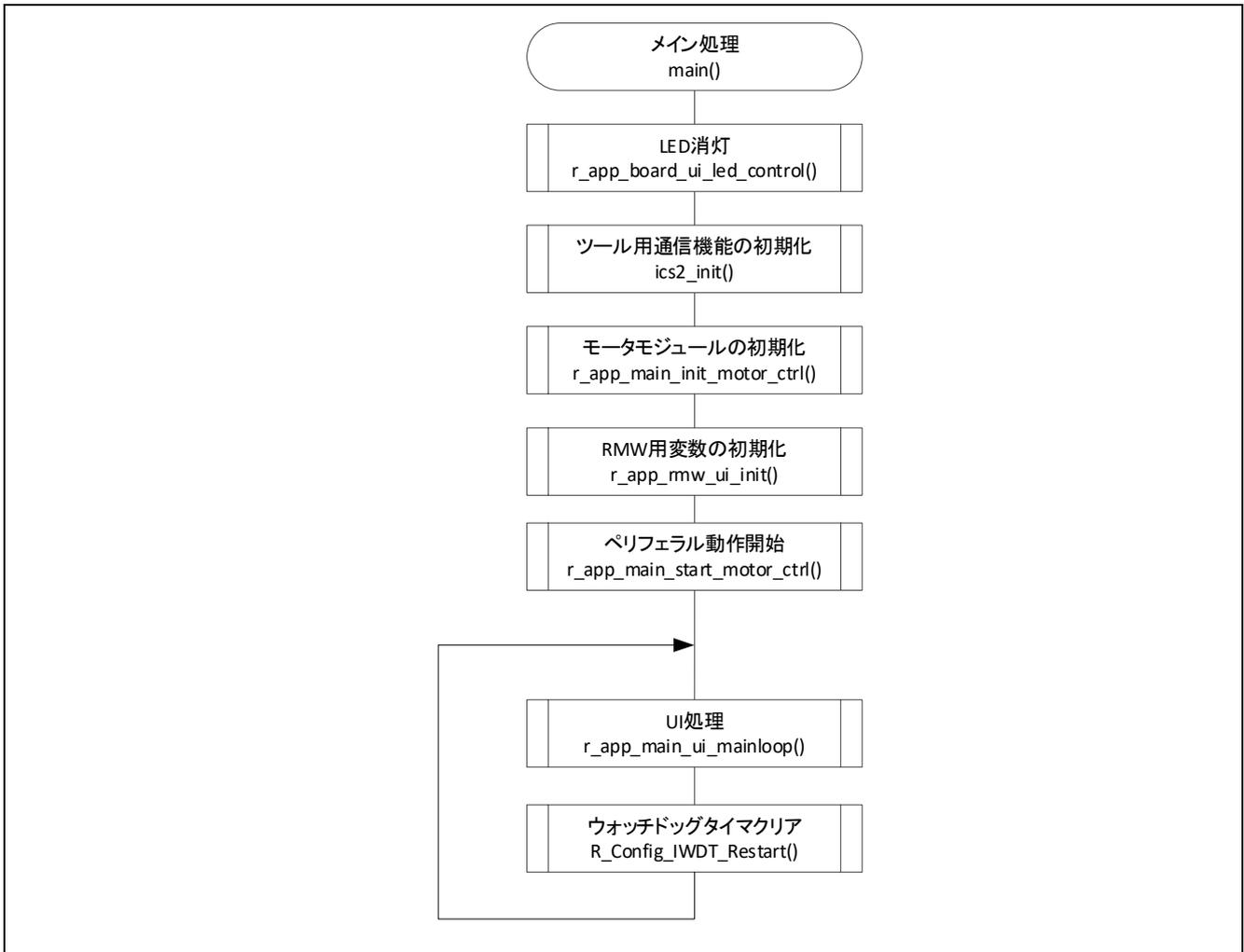


図 5-2 メイン処理フローチャート

5.1.3.2 UI 処理

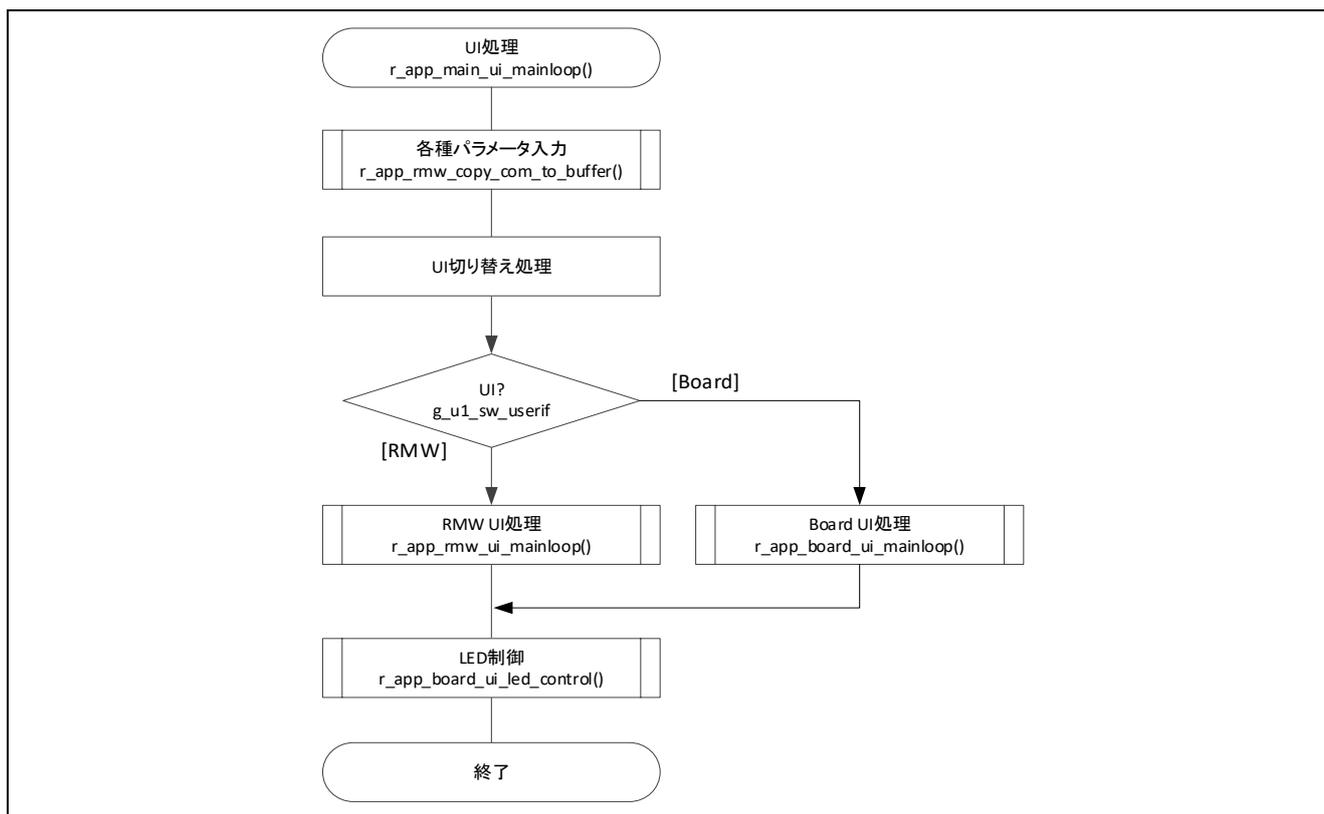


図 5-3 UI 処理フローチャート

5.1.3.3 ボード UI 処理



図 5-4 ボード UI 処理フローチャート

5.1.3.4 RMW UI 処理

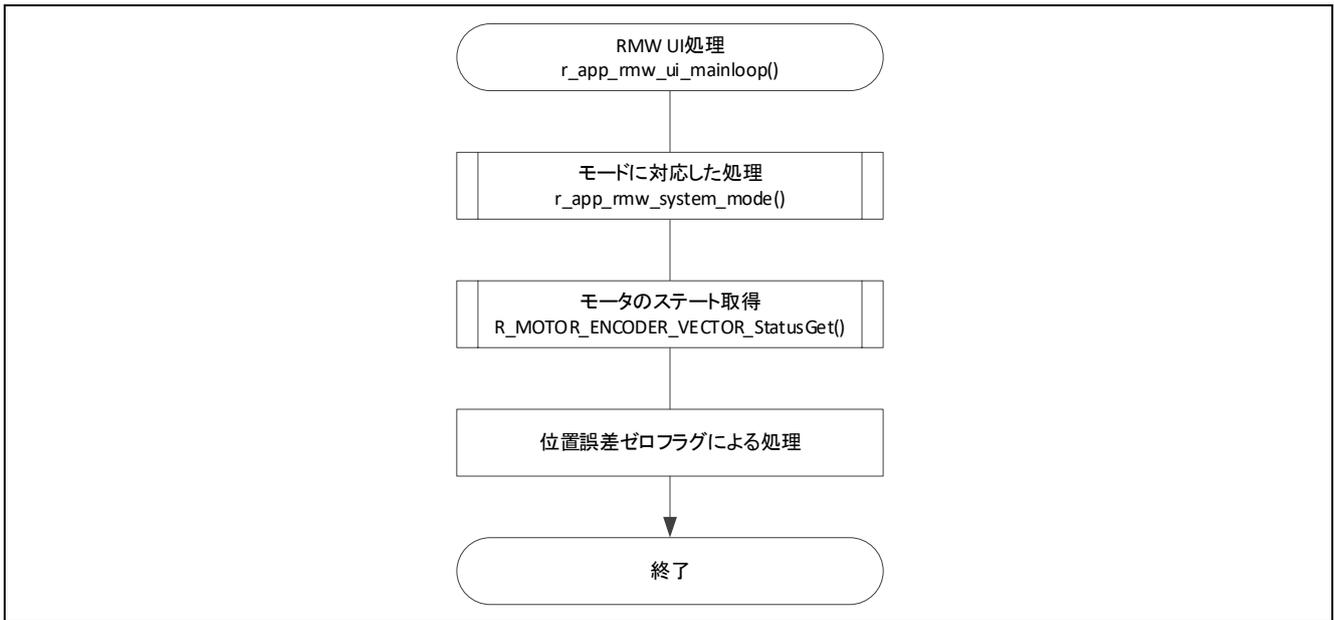


図 5-5 RMW UI 処理フローチャート

5.1.4 コンフィグレーション情報

アプリケーション層で使用するコンフィグレーション情報を表 5-2 に示します。

表 5-2 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_app_control_cfg.h	APP_CFG_USE_UI	UI の初期値設定 RMW : MAIN_UI_RMW BOARD : MAIN_UI_BOARD
	APP_CFG_FREQ_BAND_LIMIT	電流制御、速度制御、位置制御の固有周波数が近い値にならないための制限値。
	APP_CFG_MAX_CURRENT_OMEGA	電流制御系固有周波数の上限値[Hz]
	APP_CFG_MIN_OMEGA	固有周波数の下限値[Hz]
	APP_CFG_SCI_CH_SELECT	RMW 用 SCI のチャンネルセレクト

表 5-3 コンフィグレーション情報初期値一覧

マクロ名	設定	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
APP_CFG_USE_UI	MAIN_UI_RMW	
APP_CFG_FREQ_BAND_LIMIT	3.0f	
APP_CFG_MAX_CURRENT_OMEGA	1000.0f	
APP_CFG_MIN_OMEGA	1.0f	
APP_CFG_SCI_CH_SELECT	0x60 (SCI6)	0x10(SCI1)

5.1.5 構造体・変数情報

アプリケーション層でユーザが使用可能な変数一覧を表 5-4 に示します。また、RMW を使用してモータモジュールのパラメータを更新するための構造体を用意しており、その構造体メンバを表 5-5 に示します。

表 5-4 変数一覧

変数	説明
g_st_rmw_input_buffer	RMW 変数更新用構造体
g_u1_update_param_flag	バッファ転送完了フラグ
com_u1_system_mode	ユーザ入力用システムモード切り替え変数 0: モータ停止 1: モータ駆動 3: エラー解除
g_u1_system_mode	システムモード 0: モータ停止 1: モータ駆動 2: エラー
com_u1_enable_write	ユーザ入力用変数書き換え許可
g_u1_enable_write	変数書き換え許可
com_u1_sw_userif	ユーザ入力用 UI 切り替え変数 0: RMW UI 1: BOARD UI
g_u1_sw_userif	UI 切り替え用変数
com_u2_offset_calc_time	電流オフセット値計算時間設定
com_u2_mtr_pp	駆動するモータの極対数
com_f4_mtr_r	駆動するモータの抵抗 [Ω]
com_f4_mtr_ld	駆動するモータの d 軸インダクタンス [H]
com_f4_mtr_lq	駆動するモータの q 軸インダクタンス [H]
com_f4_mtr_m	駆動するモータの磁束 [Wb]
com_f4_mtr_j	駆動するモータのロータイナーシャ [kgm^2]
com_f4_nominal_current_rms	駆動するモータの定格電流 [Arms]
com_f4_max_speed_rpm	駆動するモータの速度最大値 (機械角) [rpm]
com_u1_ctrl_loop_mode	制御ループの切り換え 0: 位置制御 1: 速度制御
com_u1_encd_angle_adj_mode	磁極位置検出モード 0: 強制励磁による位置検出 1: ホールセンサを使用した位置検出
com_u2_encd_cpr	エンコーダパルス数 [p/r]
com_f4_hs_change_speed_rpm	切り換え速度 (高速時速度算出切り換え機能) [rpm]
com_f4_hs_change_margin_rpm	切り換え速度マージン (高速時速度算出切り換え機能) [rpm]
com_f4_ol_ref_id	d 軸電流指令値 [A]

変数	説明
com_f4_id_up_time	d 軸電流指令値の増加時間の設定
com_f4_current_omega_hz	電流制御系固有周波数 [Hz]
com_f4_current_zeta	電流制御系減衰係数
com_f4_speed_omega_hz	速度制御系固有周波数 [Hz]
com_f4_speed_zeta	速度制御系減衰係数
com_f4_speed_lpf_hz	速度 LPF カットオフ周波数[Hz]
com_f4_ref_speed_rpm	速度指令値（機械角）[rpm]
com_f4_speed_rate_limit_rpm	速度指令最大増減幅 [rpm/s]（速度制御時使用）
com_f4_overspeed_limit_rpm	速度制限値（機械角）[rpm]
com_u1_pos_cmd_mode	位置指令値の入力方式切り換え 0：位置指令 0 固定 1：ステップ応答 2：台形波応答
com_u2_pos_interval_time	位置応答定常待ち時間
com_u2_pos_dead_band	不感帯（エンコーダパルス数）[Pulse]
com_u2_pos_band_limit	位置誤差ゼロ範囲 [Pulse]
com_f4_pos_omega_hz	位置制御系固有周波数 [Hz]
com_f4_pos_ff_ratio	位置フィードフォワードゲイン
com_f4_ref_position_deg	位置指令値（機械角）[度]
com_u1_flag_extobserver_use	外乱トルク・速度推定オブザーバの設定 0：無効 1：有効
com_f4_extobs_omega	速度モジュールのオブザーバ固有周波数 [Hz]
com_f4_accel_time	加速時間 [s]（位置指令値作成用）
com_f4_posprof_max_speed_rpm	位置プロファイル用速度最大値（機械角）[rpm]
com_u1_flag_ipd_use	IPD 制御モジュールの設定 0：無効 1：有効
com_f4_ipd_speed_k_ratio	IPD 制御時速度ゲインの倍率
com_f4_ipd_pos_kp_ratio	IPD 制御時位置 P 制御量倍率
com_f4_ipd_omega_hz	IPD 制御固有周波数
com_f4_ipd_pos_ff_ratio	IPD 制御フィードフォワードゲイン
com_u1_flag_volt_err_comp_use	電圧誤差補償の設定 0：無効 1：有効
com_u1_flag_fluxwkn_use	弱め磁束制御の設定 0：無効 1：有効

変数	説明
s_u1_cnt_ics	ICS watchpoint のスキップ回数カウンタ

表 5-5 RMW によるパラメータ更新用構造体の変数一覧

構造体	変数	説明
st_rmw_param_buf fer_t	u2_offset_calc_time	電流オフセットの検出時間設定
	st_motor	モータパラメータ用の構造体
	f4_max_speed_rpm	最大速度 [rpm]
RMW 変数更新用 構造体	u1_ctrl_loop_mode	制御ループのモード (位置制御, 速度制御)
	u1_encd_angle_adj_mode	初期位置検出モード選択
	u2_encd_cpr	エンコーダの 1 回転のパルス数 [p/r]
	f4_hs_change_speed_rpm	速度検出方式の切り替え速度 [rpm]
	f4_hs_change_margin_rpm	速度検出方式の切り替え速度マージン[rpm]
	f4_ol_ref_id	オープンループ時の d 軸電流指令値 [A]
	f4_id_up_time	Id の増加にかかる時間の設定
	f4_current_omega_hz	電流制御系固有周波数 [Hz]
	f4_current_zeta	電流制御系減衰係数
	f4_speed_omega_hz	速度制御系固有周波数 [Hz]
	f4_speed_zeta	速度制御系減衰係数
	f4_speed_lpf_hz	速度 LPF カットオフ周波数[Hz]
	f4_ref_speed_rpm	速度指令値 [rpm]
	f4_speed_rate_limit_rpm	速度の変化量制限 [rpm/s]
	f4_overspeed_limit_rpm	速度制限値 [rpm]
	u1_pos_cmd_mode	位置指令のステータス
	u2_pos_interval_time	位置制御のインターバル時間
	u2_pos_dead_band	位置のデッドバンド
	u2_pos_band_limit	デッドバンドの制限値
	f4_pos_omega_hz	位置制御系固有周波数 [Hz]
	f4_pos_ff_ratio	位置フィードフォワードゲイン
	f4_ref_position_deg	位置指令値 [deg.]
	u1_flag_extobserver_use	オブザーバの使用有無のフラグ
	f4_extobs_omega	速度モジュールのオブザーバ固有周波数 [Hz]
	f4_accel_time	加速時間 [s]
	f4_posprof_max_speed_rpm	位置プロファイル用速度最大値 (機械角) [rpm]
	u1_flag_ipd_use	IPD 制御使用有無のフラグ
	f4_ipd_speed_k_ratio	IPD 制御の速度定数
	f4_ipd_pos_kp_ratio	IPD 制御の位置 kp 定数

構造体	変数	説明
st_rmw_param_buf fer_t	f4_ipd_omega_hz	IPD 制御周波数 [Hz]
	f4_ipd_pos_ff_ratio	IPD 制御の位置フィードフォワードゲイン
RMW 変数更新用 構造体	u1_flag_volt_err_comp_use	電圧誤差補償使用有無のフラグ
	u1_flag_fluxwkn_use	弱め磁束制御の使用有無フラグ

5.1.6 マクロ定義

マクロ一覧を表 5-6 に示します。

表 5-6 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_app_main.h	MAIN_UI_RMWS	0	RMW UI 使用
	MAIN_UI_BOARD	1	ボード UI 使用
	MAIN_UI_SIZE	2	UI 選択可能数
r_app_board_ui.h	BOARD_SW1_ON	1	SW1 スイッチ ON
	BOARD_SW1_OFF	0	SW1 スイッチ OFF
	BOARD_SW2_ON	0	SW2 スイッチ ON
	BOARD_SW2_OFF	1	SW2 スイッチ OFF
	BOARD_CHATTERING_CNT	10	チャタリング除去用カウント数
	BOARD_AD12BIT_DATA	MOTOR_MCU_CFG_AD12BIT_DATA	12 ビット AD 値
	BOARD_VR1_POSITION_DEAD_BAND	2	VR1 用位置不感帯 [deg]
	BOARD_VR1_SPEED_DEAD_BAND	80	VR1 用速度不感帯 [rpm]
	BOARD_VR1_SPEED_MARGIN	300	VR1 用速度マージン [rpm]
	BOARD_VR1_SCALING_POS	$(180 + 18) / (\text{BOARD_AD12BIT_DATA} / 2 + 1)$	VR1 用位置スケール係数
	BOARD_VR1_SCALING_SPEED	$(\text{MOTOR_CFG_MAX_SPEED_RPM} + \text{BOARD_VR1_SPEED_MARGIN}) / (\text{BOARD_AD12BIT_DATA} / 2 + 1)$	VR1 用速度スケール係数
	BOARD_ADJUST_OFFSET	MOTOR_MCU_CFG_ADC_OFFSET	VR1 用オフセット値
r_app_control_cfg.h	APP_CFG_SCI_CHANNEL_SELECT	0x60	ユーザ用使用 SCI チャンネル選択
	APP_CFG_USE_UI	MAIN_UI_RMWS	UI 初期選択
	APP_CFG_FREQ_BAND_LIMIT	3.0f	制御系間の帯域制限 [倍率]
	APP_CFG_MAX_CURRENT_OMEGA	1000.0f	電流制御系固有周波数最大値 [Hz]
	APP_CFG_MIN_OMEGA	1.0f	固有周波数最小値 [Hz]
r_app_rmws.h	ICS_DECIMATION	5	RMW watchpoint のスキップ回数
	ICS_INT_LEVEL	6	RMW 割り込みの優先度
	ICS_BRR	251	RMW の通信ボーレート
	ICS_INT_MODE	1	RMW の通信モード選択

ファイル名	マクロ名	定義値	備考
r_app_rmw.h	ICS_SCI_CH_SELECT	APP_CFG_SCI_C H_SELECT	使用 SCI チャンネル

5.1.7 パラメータ調整・設定

アプリケーション層のコンフィグレーション情報は r_app_control_cfg.h で設定する必要があります。設定するパラメータは 5.1.4 を参照ください。

表 5-4 に示す変数の設定・更新は、RMW から行ってください。RMW の操作は 3 クイックスタートガイド及び Renesas Motor Workbench ユーザーズマニュアル (R21UZ0004) を参照ください。

5.2 マネージャモジュール

マネージャモジュールは各制御モジュールを適切に使用してモータ制御を行うモジュールです。各モジュールのインタフェースやモータ制御のシステム全体の管理、システム保護などを行っています。

5.2.1 機能

マネージャモジュールの機能一覧を表 5-7 に示します。

表 5-7 マネージャモジュールの機能一覧

機能	説明
モード管理	ユーザの指令に対してシステムを切り替えてモータを制御します。
保護機能	システム保護機能によりエラー処理を行います。
制御方式の管理	位置制御や電流制御の状態の取得・設定を行います。
速度・位置情報の取得	速度・位置情報の取得を行います。
制御モジュールの指令値設定	電流制御モジュール、速度制御モジュール、位置制御モジュールに対して入力する指令値を制御の状態から選択します。
割り込み処理	スマート・コンフィグレータで設定した割り込みを受けて処理を行い、適切なモジュールへ処理の割り振りを行います。

5.2.2 モジュール構成図

モジュール構成図を図 5-6 に示します。

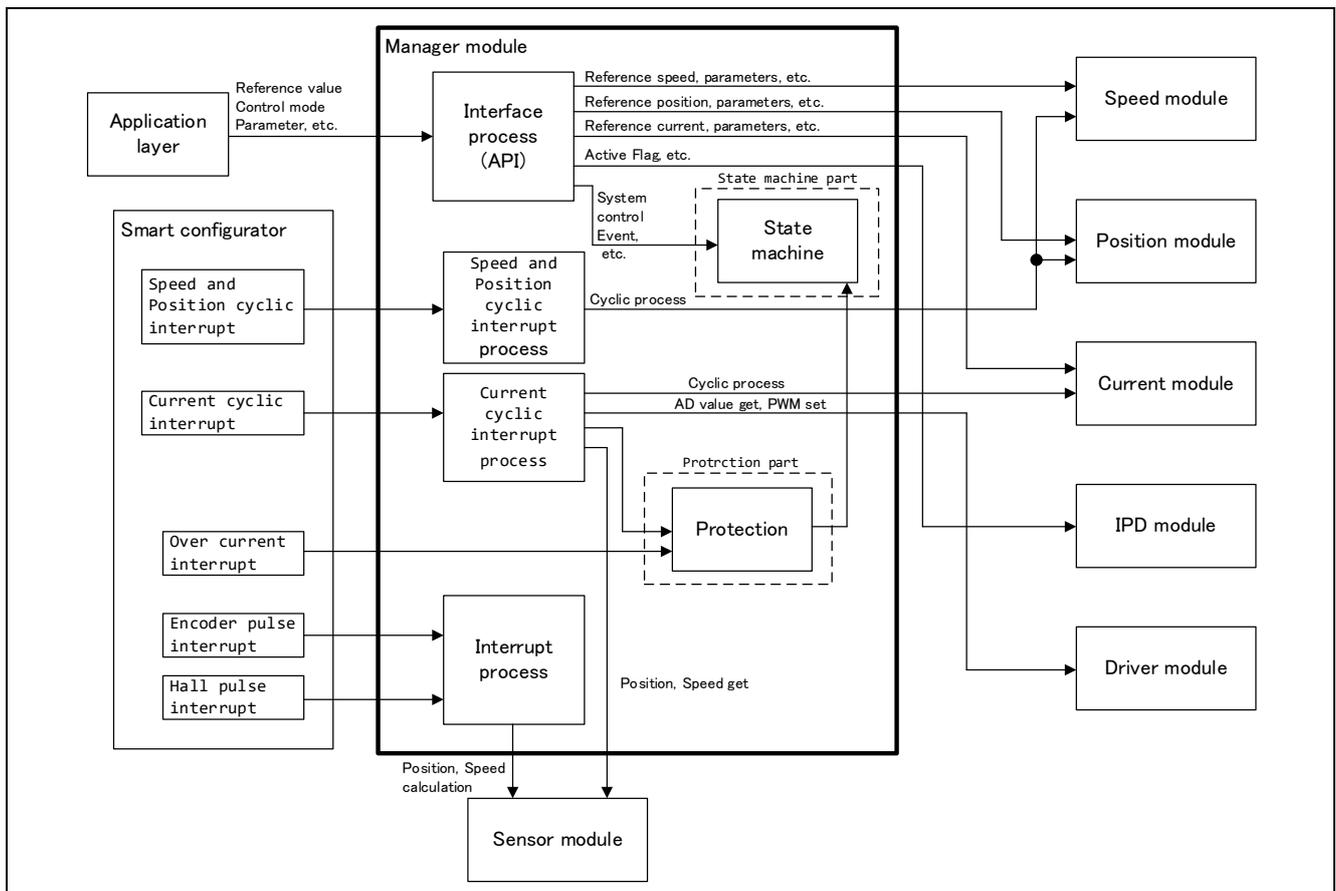


図 5-6 マネージャモジュール構成図

5.2.3 モード管理

図 5-7 に本アプリケーションノート対象ソフトウェアにおける状態遷移図を示します。本アプリケーションノート対象ソフトウェアでは、「SYSTEM MODE」と、「RUN MODE」により状態を管理し、「Control Config」は、ソフトウェア内でアクティブになっている制御系を表しています。

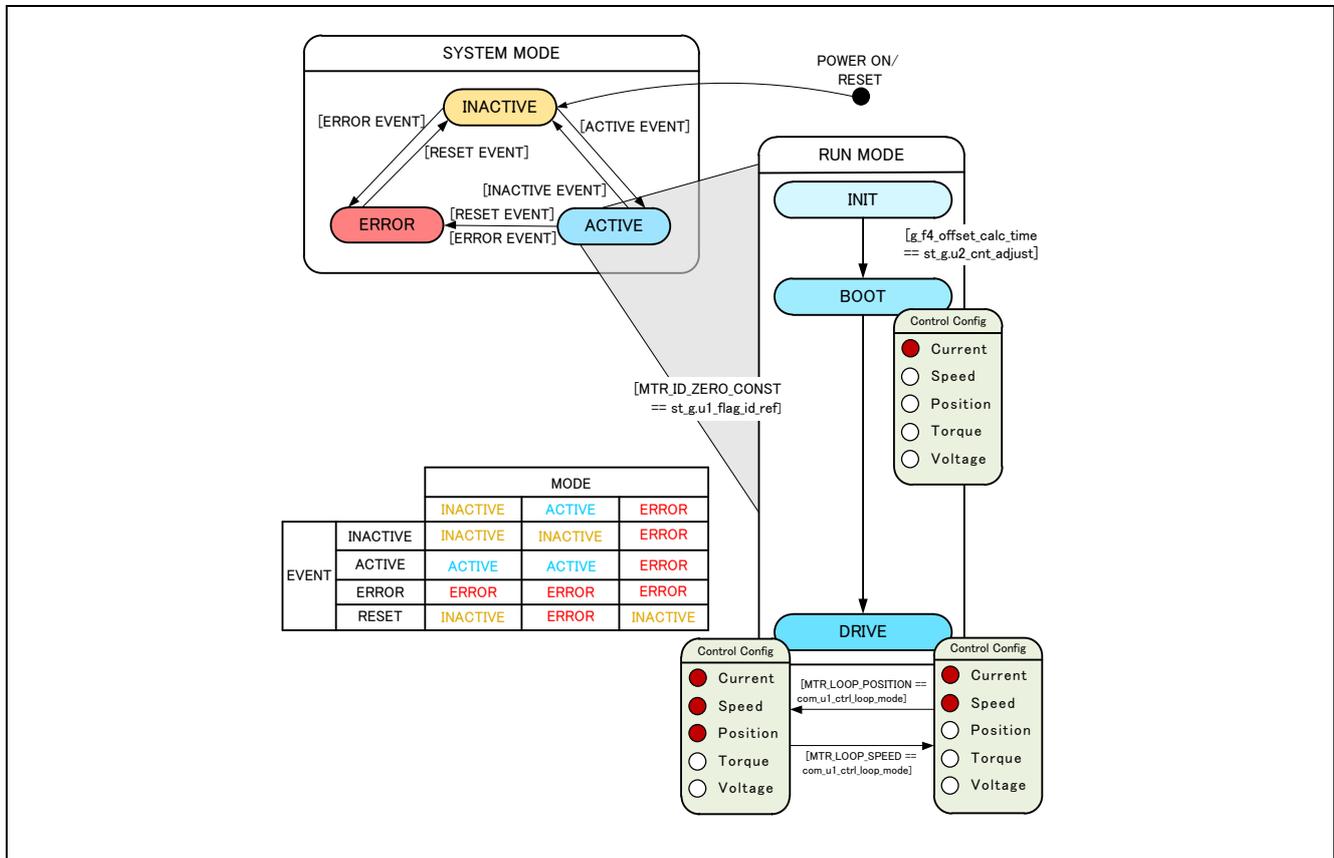


図 5-7 エンコーダベクトル制御ソフトウェアの状態遷移図

(1) SYSTEM MODE

システム動作状態を表します。各イベント (EVENT) の発生により、状態が遷移します。システムの動作状態は、モータ駆動停止 (INACTIVE)、モータ駆動 (ACTIVE)、異常状態 (ERROR) があります。

(2) RUN MODE

モータの制御状態を表します。システムの状態が ACTIVE になると、モータの駆動状態が図 5-7 のように遷移します。

(3) EVENT

各 SYSTEM MODE 中に EVENT が発生すると、その EVENT に従って、システム動作状態が図 5-7 中の表のように遷移します。各 EVENT の発生要因は下記となります。

表 5-8 EVENT 一覧

イベント名	発生要因
INACTIVE	ユーザ操作により発生します
ACTIVE	ユーザ操作により発生します
ERROR	システムが異常を検出したときに発生します
RESET	ユーザ操作により発生します

5.2.4 保護機能

本制御プログラムは、以下のエラー状態を持ち、それぞれの場合に緊急停止機能を実装しています。システム保護機能に関わる各設定値は表 5-9 を参照してください。

- 過電流エラー
過電流エラーはハードウェア及びソフトウェア両方で検出されます。
ハードウェアからの緊急停止信号（過電流検出）により、PWM 出力端子をハイインピーダンス状態にします。
また、過電流監視周期で U 相、V 相、W 相電流を監視し、過電流（過電流リミット値を超過）を検出した時に、緊急停止します（ソフトウェア検出）。
過電流リミット値はモータの定格電流 (MOTOR_CFG_NOMINAL_CURRENT_RMS) から自動で計算されます。
- 過電圧エラー
過電圧監視周期でインバータ母線電圧を監視し、過電圧（過電圧リミット値を超過）を検出した時に、緊急停止します。過電圧リミット値は検出回路の抵抗値の誤差等を考慮して設定した値です。
- 低電圧エラー
低電圧監視周期でインバータ母線電圧を監視し、低電圧（低電圧リミット値を下回った場合）を検出した時に、緊急停止します。低電圧リミット値は検出回路の抵抗値の誤差等を考慮して設定した値です。
- 回転速度エラー
回転速度監視周期で速度を監視し、速度リミット値を超過した場合、緊急停止します。

表 5-9 各システム保護機能設定値

過電流エラー	過電流リミット値 [A]	2.69
	監視周期 [μ s]	電流制御周期 ^{*1}
過電圧エラー	過電圧リミット値 [V]	60
	監視周期 [μ s]	電流制御周期 ^{*1}
低電圧エラー	低電圧リミット値 [V]	8
	監視周期 [μ s]	電流制御周期 ^{*1}
回転速度エラー	速度リミット値 [rpm]	4500
	監視周期 [μ s]	電流制御周期 ^{*1}

【注】 1. 表 4-1 エンコーダベクトル制御ソフトウェア基本仕様参照

5.2.5 フローチャート

マネージャモジュールはスマート・コンフィグレータにて設定された割り込みに対して様々なモジュールのAPIを使って処理を行い、モータ制御を行っています。各割り込み処理フローを示します。

5.2.5.1 電流制御用割り込み処理

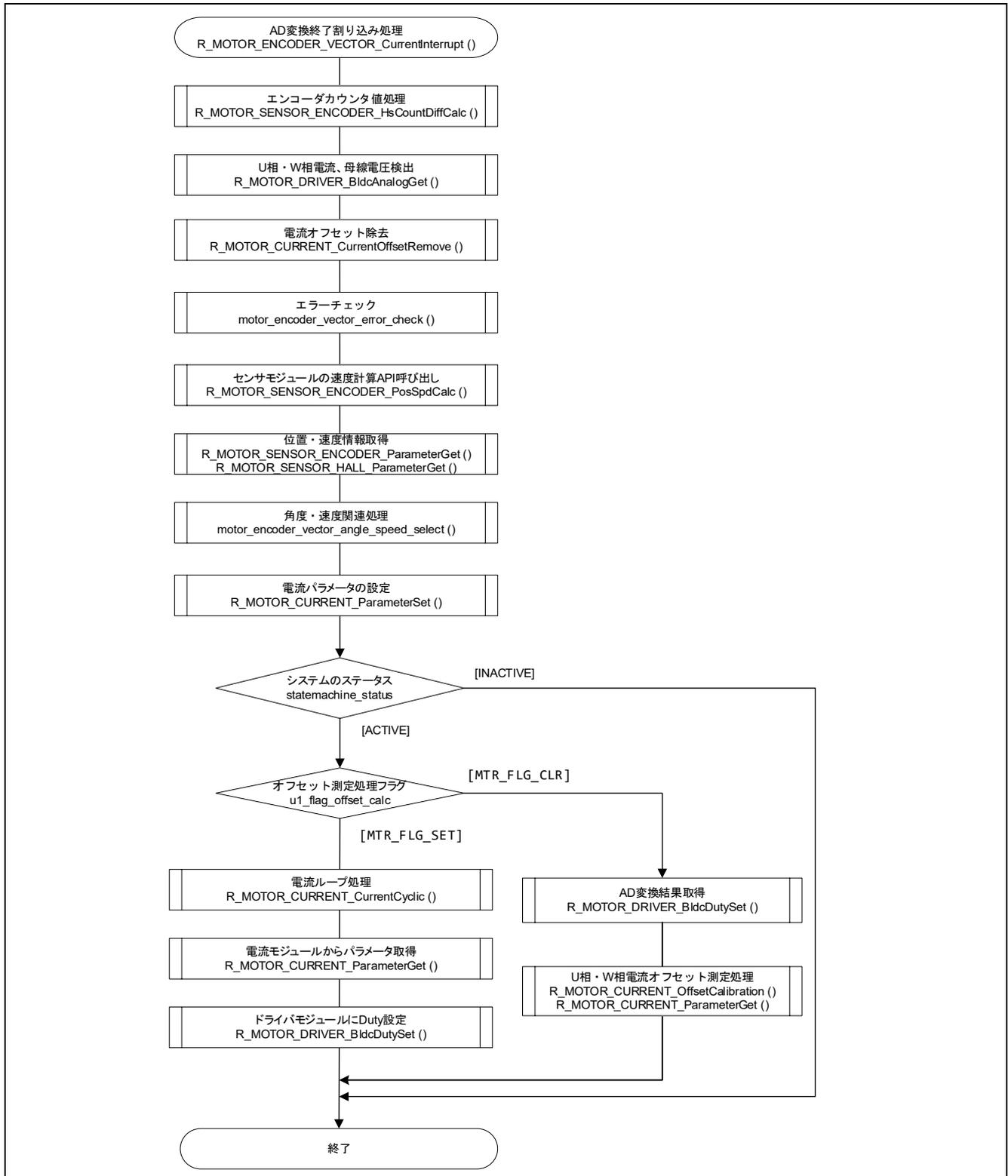


図 5-8 電流制御用割り込み処理フローチャート

5.2.5.2 位置・速度制御用割り込み処理

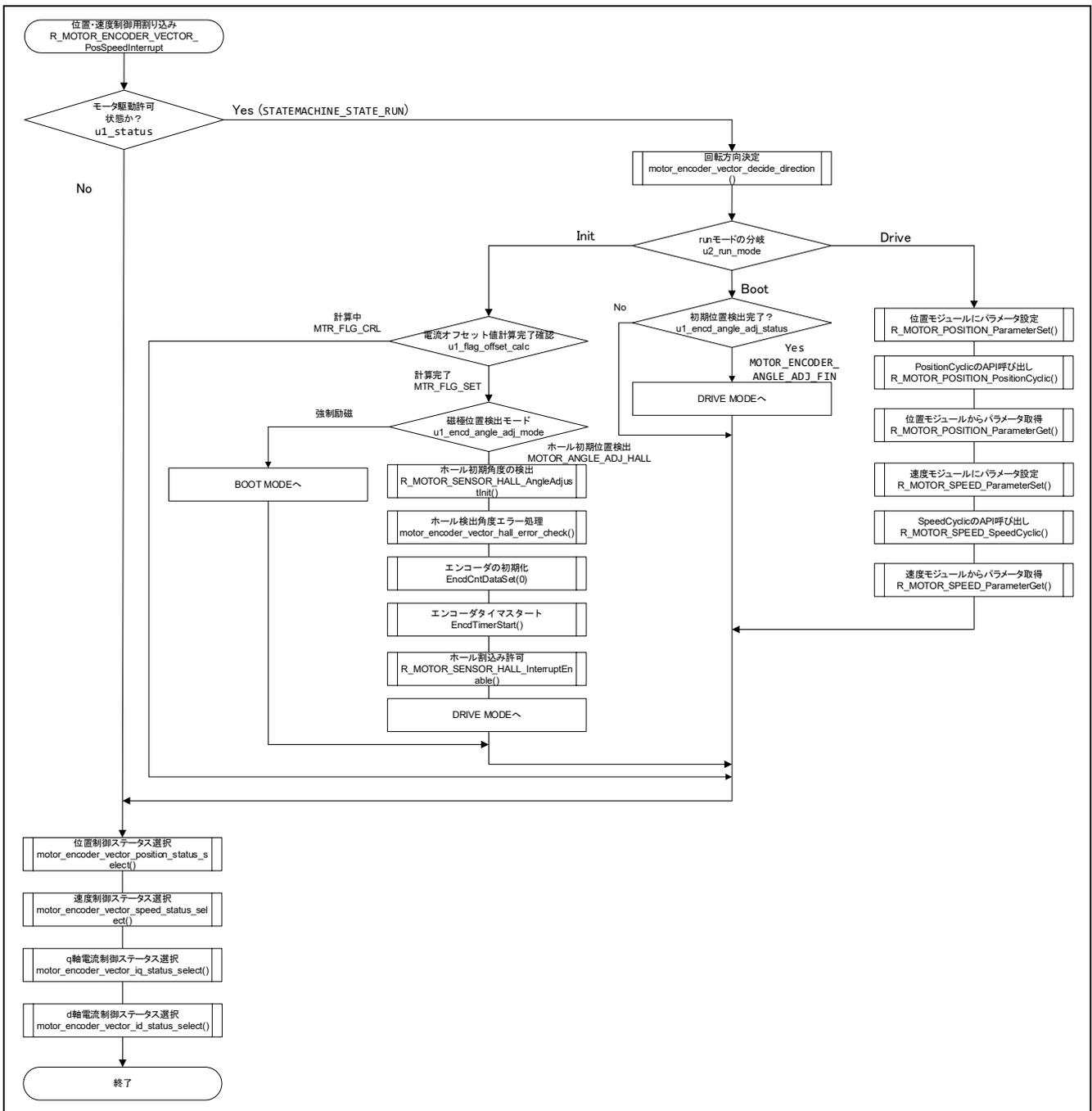


図 5-9 位置・速度制御用割り込みフローチャート

5.2.5.3 過電流検出割り込み処理

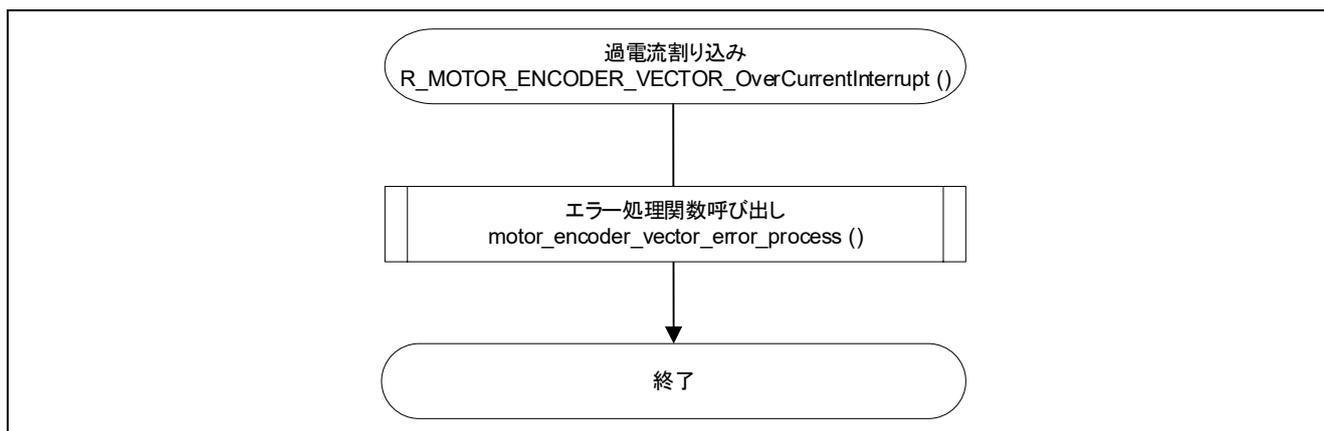


図 5-10 過電流検出割り込み処理フローチャート

5.2.5.4 エンコーダ入力信号割り込み処理

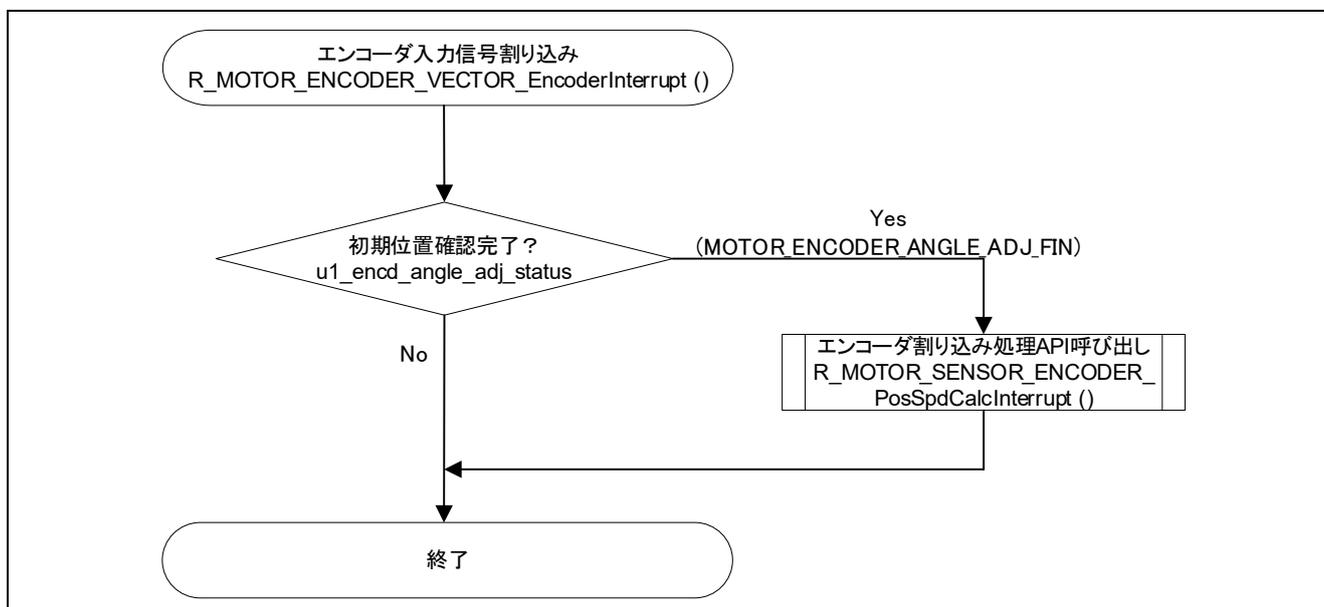


図 5-11 エンコーダカウントキャプチャ割り込み処理フローチャート

5.2.5.5 ホール信号割り込み処理

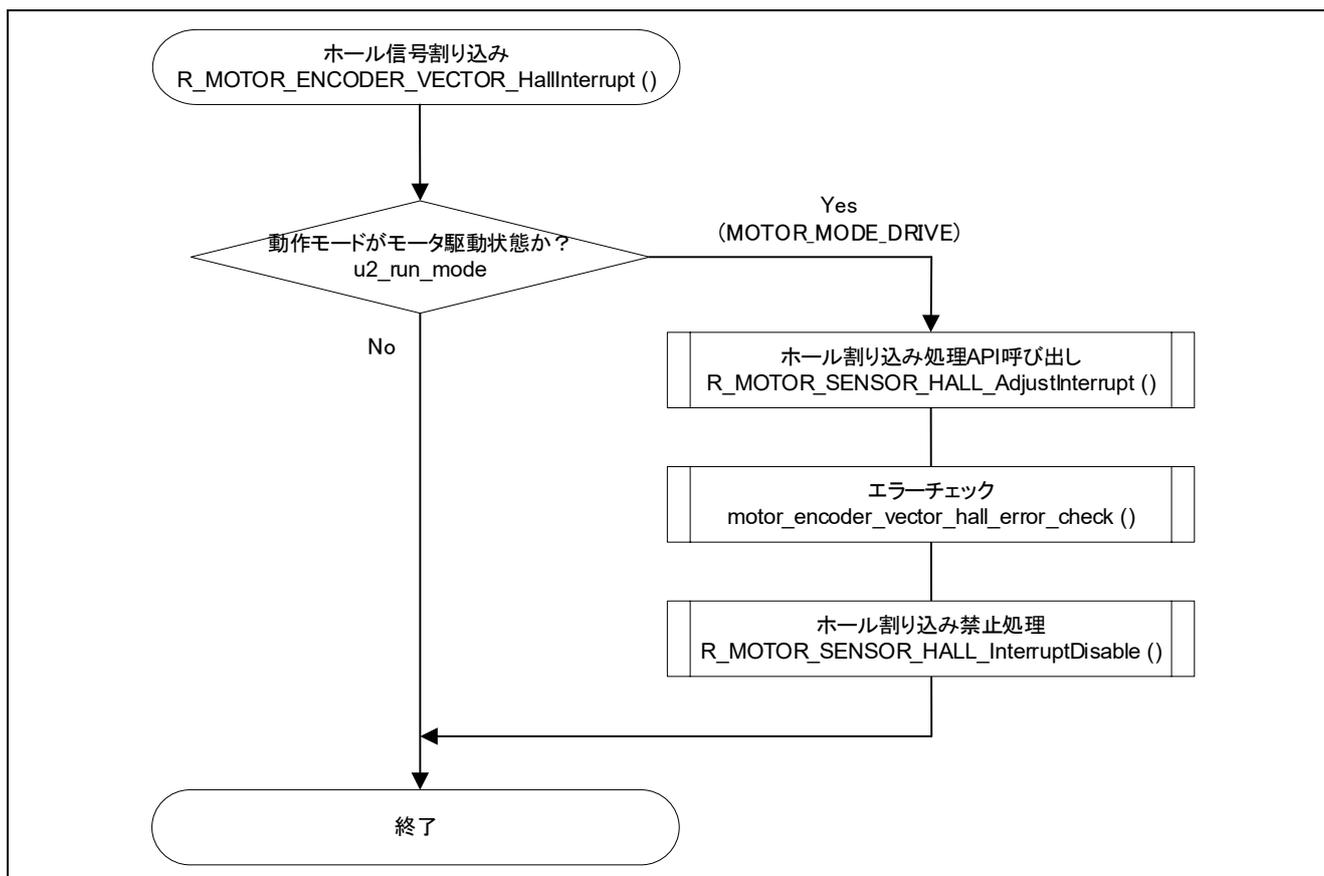


図 5-12 ホールエッジ割り込み処理フローチャート

5.2.6 API

マネージャモジュールの API 一覧を表 5-10 に示します。

表 5-10 API 一覧

API	説明
R_MOTOR_ENCODER_VECTOR_Open	マネージャモジュールのインスタンスを生成します。また、使用する制御モジュールの Open 関数を実行しインスタンスを生成させます。
R_MOTOR_ENCODER_VECTOR_Close	マネージャモジュールおよび、各モジュールをリセット状態にします。
R_MOTOR_ENCODER_VECTOR_Reset	マネージャモジュールおよび各モジュールを初期化します。
R_MOTOR_ENCODER_VECTOR_ParameterUpdate	本モジュールの制御パラメータを更新します。また、関連するモジュールの制御パラメータ更新を行います。
R_MOTOR_ENCODER_VECTOR_MotorStart	モータ駆動状態にします。
R_MOTOR_ENCODER_VECTOR_MotorStop	モータ停止状態にします。
R_MOTOR_ENCODER_VECTOR_MotorReset	システムのエラー状態を解除します。
R_MOTOR_ENCODER_VECTOR_ErrorSet	システムにエラー状態を設定します。
R_MOTOR_ENCODER_VECTOR_PositionSet	位置指令値を設定します。位置制御時に有効になります。
R_MOTOR_ENCODER_VECTOR_PositionGet	位置情報を取得します。
R_MOTOR_ENCODER_VECTOR_SpeedSet	速度指令値を設定します。速度制御時に有効になります。
R_MOTOR_ENCODER_VECTOR_SpeedGet	速度情報を取得します。
R_MOTOR_ENCODER_VECTOR_StatusGet	ステートマシンの状態を取得します。
R_MOTOR_ENCODER_VECTOR_ErrorStatusGet	エラー状態を取得します。
R_MOTOR_ENCODER_VECTOR_CtrlTypeSet	制御方式を設定します。制御方式を変更する場合は、モータを停止状態にしてください。 0：位置制御 1：速度制御
R_MOTOR_ENCODER_VECTOR_LoopModeStatusGet	制御方式を取得します。 0：位置制御 1：速度制御
R_MOTOR_ENCODER_VECTOR_PositionCommandModeSet	位置制御の位置指令生成のモードを選択します。位置制御を行う場合は、1か2に設定してください。 0：位置をゼロに固定 1：ステップ応答動作 2：台形駆動動作
R_MOTOR_ENCODER_VECTOR_InPositionFlagGet	位置制御完了状態を取得します。位置制御時に有効になります。 0：位置制御未完了状態 1：位置制御完了状態
R_MOTOR_ENCODER_VECTOR_PosSpeedInterrupt	位置制御・速度制御を行うための割り込み処理を行います。
R_MOTOR_ENCODER_VECTOR_CurrentInterrupt	電流制御を行うための割り込み処理を行います。

API	説明
R_MOTOR_ENCODER_VECTOR_OverCurrentInterrupt	過電流が発生した際の割り込み処理を行います。
R_MOTOR_ENCODER_VECTOR_HallInterrupt	ホールセンサ信号の割り込み処理を行います。
R_MOTOR_ENCODER_VECTOR_EncoderInterrupt	エンコーダ信号の割り込み処理を行います。
R_MOTOR_ENCODER_VECTOR_IPDEnableSet	IPD 制御モジュールの有効 / 無効を切り替えます。 0 : 無効 1 : 有効

5.2.7 コンフィグレーション情報

マネージャモジュールのコンフィグレーション情報一覧を表 5-11 に示します。

表 5-11 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	MOTOR_MCU_CFG_PWM_TIMER_FREQ	PWM のタイマ周波数 [MHz]
	MOTOR_MCU_CFG_CARRIER_FREQ	キャリア周波数 [kHz]
	MOTOR_MCU_CFG_INTR_DECIMATION	キャリア割り込みの間引き回数
	MOTOR_MCU_CFG_AD_FREQ	ADC の動作周波数 [MHz]
	MOTOR_MCU_CFG_AD_SAMPLING_CYCLE	ADC のサンプリングステート [cycle]
	MOTOR_MCU_CFG_AD12BIT_DATA	ADC の分解能
	MOTOR_MCU_CFG_ADC_OFFSET	ADC の中間データ
	MOTOR_TYPE_BLDC	使用モータ (BLDC)
	MOTOR_COMMON_CFG_LOOP_MODE	デフォルトの動作モード設定
	MOTOR_COMMON_CFG_IPD_CTRL	デフォルトの IPD 制御使用設定。
	MOTOR_COMMON_CFG_OVERCURRENT_MARGIN_MULT	過電流のリミット値 [A]
	MOTOR_COMMON_CFG_IA_MAX_CALC_MULT	過電流リミット値計算用係数。 BLDC: $\sqrt{3}$ STM: $\sqrt{2}$
	MOTOR_MCU_CFG_TFU_OPTIMIZE	TFU 専用関数処理の設定 MTR_ENABLE MTR_DISABLE

表 5-12 コンフィグレーション情報初期値一覧

マクロ名	設定値
MOTOR_MCU_CFG_PWM_TIMER_FREQ	CG_CONFIG_MOTOR_PWM_TIMER_FREQ
MOTOR_MCU_CFG_CARRIER_FREQ	CG_CONFIG_MOTOR_CARRIER_FREQ
MOTOR_MCU_CFG_INTR_DECIMATION	CG_CONFIG_MOTOR_INTR_DECIMATION
MOTOR_MCU_CFG_AD_FREQ	CG_MOTOR_MCU_CFG_AD_FREQ
MOTOR_MCU_CFG_AD_SAMPLING_CYCLE	45
MOTOR_MCU_CFG_AD12BIT_DATA	CG_MOTOR_CFG_MAX_AD_DATA
MOTOR_MCU_CFG_ADC_OFFSET	0x7FF
MOTOR_TYPE_BLDC	定義有り
MOTOR_COMMON_CFG_LOOP_MODE	MOTOR_LOOP_SPEED
MOTOR_COMMON_CFG_IPD_CTRL	MTR_DISABLE
MOTOR_COMMON_CFG_OVERCURRENT_MARGIN_MULT	1.5
MOTOR_COMMON_CFG_IA_MAX_CALC_MULT	MTR_SQRT_3
MOTOR_MCU_CFG_TFU_OPTIMIZE	MTR_ENABLE

5.2.8 構造体・変数情報

マネージャモジュールの構造体・変数一覧を表 5-13 に示します。マネージャモジュールは API のインスタンス確保にて、マネージャモジュール用構造体(g_st_encoder_vector)を定義します。

表 5-13 構造体・変数一覧

構造体	変数	説明
st_encoder_vector _control_t マネージャ モジュール用 構造体	u1_direction	回転方向 0 : CW 1 : CCW
	u1_ctrl_loop_mode	制御モード選択 0 : 位置制御 1 : 速度制御
	u1_encd_angle_adj_mode	初期位置検出モード選択 0 : 強制励磁 1 : ホールセンサ使用
	u1_encd_angle_adj_status	初期位置検出のステータス
	u2_encd_angle_adj_time	初期位置検出の待ち時間 [us]
	u2_encd_angle_adj_cnt	初期位置検出の待ち時間カウント
	u1_flag_ipd_use	IPD 制御使用有無のフラグ 0 : IPD 制御無効 1 : IPD 制御有効
	u2_error_status	エラーステータス
	u2_run_mode	動作モード 0 : 初期化 1 : 始動準備 2 : モータ駆動
	u1_state_id_ref	d 軸電流指令値のステータス
	u1_state_iq_ref	q 軸電流指令値のステータス
	u1_state_speed_ref	速度指令値のステータス
	u2_encd_cpr	エンコーダの 1 回転のパルス数
	f4_vdc_ad	母線電圧 [V]
	f4_iu_ad	u 相電流 [A]
	f4_iv_ad	v 相電流 [A]
	f4_iw_ad	w 相電流 [A]
	f4_overcurrent_limit	過電流制限値 [A]
	f4_overvoltage_limit	過電圧制限値 [V]
	f4_undervoltage_limit	低電圧制限値 [V]
	f4_overspeed_limit_rad	過速度制限値 [rad/s]
	f4_max_speed_rc_fil_rpm	最大速度(rc フィルタ使用)
	f4_rotor_angle_rad	ロータ角度[rad]

構造体	変数	説明
st_encoder_vector_control_t マネージャ モジュール用 構造体	st_current_output	電流モジュールの出力用構造体
	st_speed_output	速度モジュールの出力用構造体
	st_position_output	位置モジュールの出力用構造体
	st_sensor_encoder_output	センサモジュールのエンコーダ出力用構造体
	st_sensor_hall_output	センサモジュールのホール出力用構造体
	st_stm	ステートマシンの構造体
	st_motor	モータパラメータ構造体
	*p_st_cc	電流制御モジュール生成インスタンスポインタ
	*p_st_sc	速度制御モジュール生成インスタンスポインタ
	*p_st_pc	位置制御モジュール生成インスタンスポインタ
	*p_st_ipd	IPD 制御モジュール生成インスタンスポインタ
	*p_st_sensor	センサモジュール生成インスタンスポインタ
	*p_st_driver	ドライバモジュール生成インスタンスポインタ
st_encoder_vector_cfg_t マネージャ モジュール制御 パラメータ設定用 構造体	u1_encd_angle_adj_mode	初期位置検出モード選択
	u2_encd_cpr	エンコーダの 1 回転のパルス数
	f4_overspeed_limit_rpm	速度制限値 [rpm]
	st_motor	モータパラメータ構造体

5.2.9 マクロ定義

マネージャモジュールのマクロ一覧を表 5-14 に示します。

表 5-14 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_encoder_vector_api.h	MOTOR_LOOP_POSITION	0	位置制御モード。
	MOTOR_LOOP_SPEED	1	速度制御モード。
	MOTOR_ENCODER_VECTOR_ERROR_NONE	(0x0000)	エラーステータス。エラーなし状態。
	MOTOR_ENCODER_VECTOR_ERROR_OVERCURRENT_HW	(0x0001)	エラーステータス HW 過電流エラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_OVERVOLTAGE	(0x0002)	エラーステータス。過電圧エラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_OVERSPEED	(0x0004)	エラーステータス。過速度エラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_HALL_PATTERN	(0x0020)	エラーステータス。ホールのパターンエラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_LOWVOLTAGE	(0x0080)	エラーステータス。低電圧エラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_OVERCURRENT_SW	(0x0100)	エラーステータス。SW の過電流エラー状態。
	MOTOR_ENCODER_VECTOR_ERROR_UNKNOWN	(0xffff)	エラーステータス。エラーコード不明のエラー状態。
	MOTOR_ANGLE_ADJ_EXCIT	0	強制励磁による初期位置検出
	MOTOR_ANGLE_ADJ_HALL	1	ホールセンサによる初期位置検出
	MOTOR_CTRL_TYPE_POS	0	制御方式切り替え用マクロ。位置制御モード。
	MOTOR_CTRL_TYPE_SPEED	1	制御方式切り替え用マクロ。速度制御モード。
r_motor_encoder_vector_manager.h	MOTOR_MODE_INIT	(0x00)	初期化を行う動作モード。
	MOTOR_MODE_BOOT	(0x01)	駆動準備を行う動作モード。
	MOTOR_MODE_DRIVE	(0x02)	モータ駆動状態の動作モード。
	MOTOR_ENCODER_ANGLE_ADJ_90DEG	0	始動時のステータス。90 度位置状態。
	MOTOR_ENCODER_ANGLE_ADJ_0DEG	1	始動時のステータス。0 度位置状態。
	MOTOR_ENCODER_ANGLE_ADJ_FIN	2	始動時のステータス。初期位置判定完了状態。

5.2.10 パラメータ調整・設定

サンプルプログラムを使用する際に、インバータの情報と使用するモータの情報を正しく設定する必要があります。サンプルプログラムの設定値を表 5-15 に示します。

表 5-15 モータパラメータ、インバータパラメータ設定

ファイル名	マクロ名	設定	説明
r_motor_inverter_cfg.h	INVERTER_CFG_SHUNT_RESIST	0.010f	シャント抵抗値 [ohm]
	INVERTER_CFG_DEADTIME	CG_CONFIG_MOTOR_DEADTIME	デッドタイム [us]
	INVERTER_CFG_VOLTAGE_GAIN	22.2766f	電圧検出用係数
	INVERTER_CFG_CURRENT_AMP_GAIN	20.0f	電流検出用アンプのゲイン
	INVERTER_CFG_CURRENT_LIMIT	21.4f	インバータボードの過電流の制限値 [A]
	INVERTER_CFG_OVERVOLTAGE_LIMIT	60.0f	過電圧制限 [V]
	INVERTER_CFG_UNDERVOLTAGE_LIMIT	8.0f	低電圧制限 [V]
	INVERTER_CFG_INPUT_V	24.0f	入力電圧 [V]
	INVERTER_CFG_ADC_REF_VOLTAGE	5.0f	MCU のアナログ電源電圧 [V]
	INVERTER_CFG_COMP_V0	0.564f	電圧誤差補償用係数 [V] *1
	INVERTER_CFG_COMP_V1	0.782f	電圧誤差補償用係数 [V] *1
	INVERTER_CFG_COMP_V2	0.937f	電圧誤差補償用係数 [V] *1
	INVERTER_CFG_COMP_V3	1.027f	電圧誤差補償用係数 [V] *1
	INVERTER_CFG_COMP_V4	1.058f	電圧誤差補償用係数 [V] *1
	INVERTER_CFG_COMP_I0	0.022f	電圧誤差補償用係数 [A] *1
	INVERTER_CFG_COMP_I1	0.038f	電圧誤差補償用係数 [A] *1
	INVERTER_CFG_COMP_I2	0.088f	電圧誤差補償用係数 [A] *1
INVERTER_CFG_COMP_I3	0.248f	電圧誤差補償用係数 [A] *1	
INVERTER_CFG_COMP_I4	0.865f	電圧誤差補償用係数 [A] *1	
r_motor_targetmotor_cfg.h	MOTOR_CFG_POLE_PAIRS	4	極対数
	MOTOR_CFG_MAGNETIC_FLUX	0.006612919f	磁束 [wb]
	MOTOR_CFG_RESISTANCE	0.8933714f	抵抗 [ohm]
	MOTOR_CFG_D_INDUCTANCE	0.001091948f	d 軸のインダクタンス [H]
	MOTOR_CFG_Q_INDUCTANCE	0.001091948f	q 軸のインダクタンス [H]
	MOTOR_CFG_ROTOR_INERTIA	0.000002647f	ロータのイナーシャ [kg m ²]

ファイル名	マクロ名	設定	説明
	MOTOR_CFG_NOMINAL_CURRENT_RMS	1.27f	定格電流 [A]
r_motor_targetmotor_cfg.h	MOTOR_CFG_MAX_SPEED_RPM	4000.0f	最大速度 [rpm]

【注】 1. 詳細は 5.5 電圧誤差補償（電流制御モジュール）を参照してください。

5.2.11 始動シーケンス管理

マネージャモジュールでは RUN MODE のステータスに合わせて d 軸電流、q 軸電流、速度、位置それぞれの指令値を管理するフラグを変更してモータを制御しています。また、これらの指令値を適切に変更することで始動シーケンスを作成し、モータの始動を行います。エンコーダベクトル制御の始動を図 5-13、図 5-14 に示します。また、エンコーダのみを使用した強制励磁による始動と、ホールセンサを利用した始動が可能であるため、それぞれの始動方法について説明します。

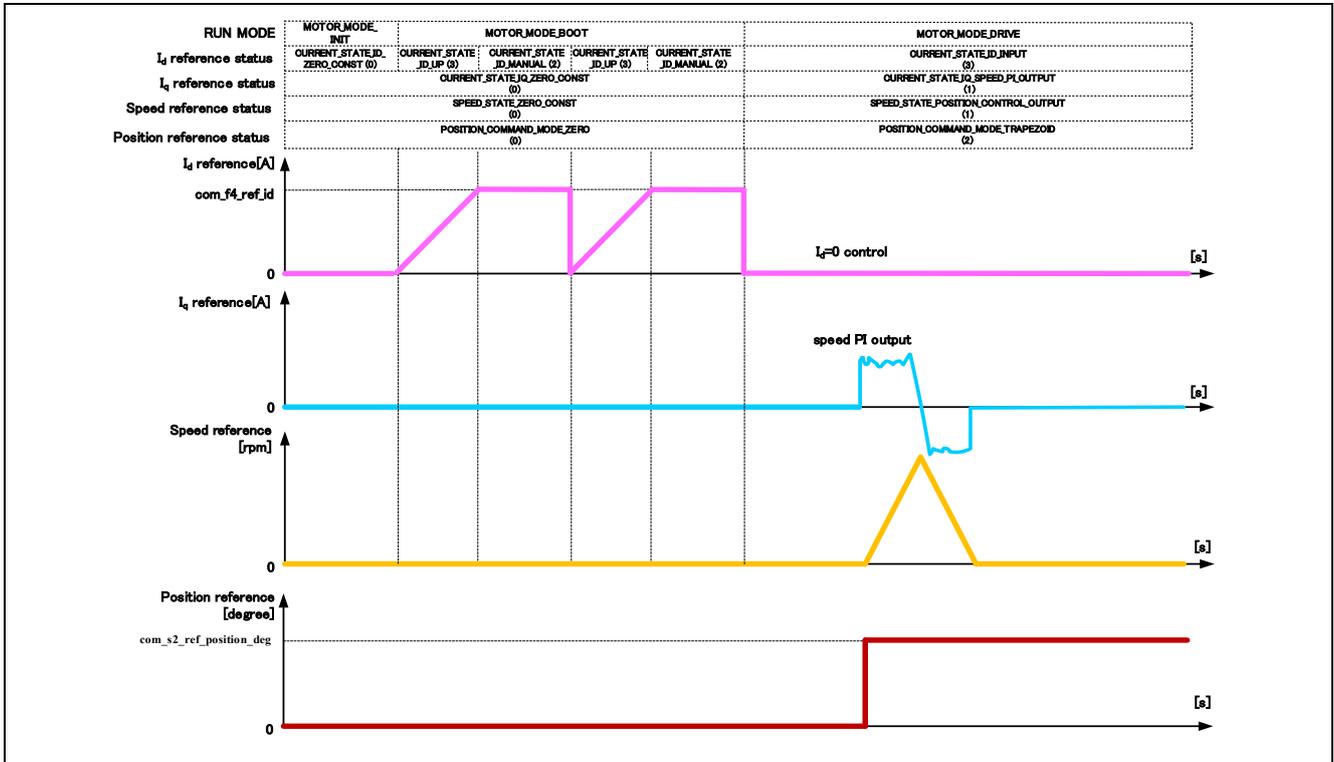


図 5-13 エンコーダ利用ベクトル制御の始動制御内容(位置制御時)

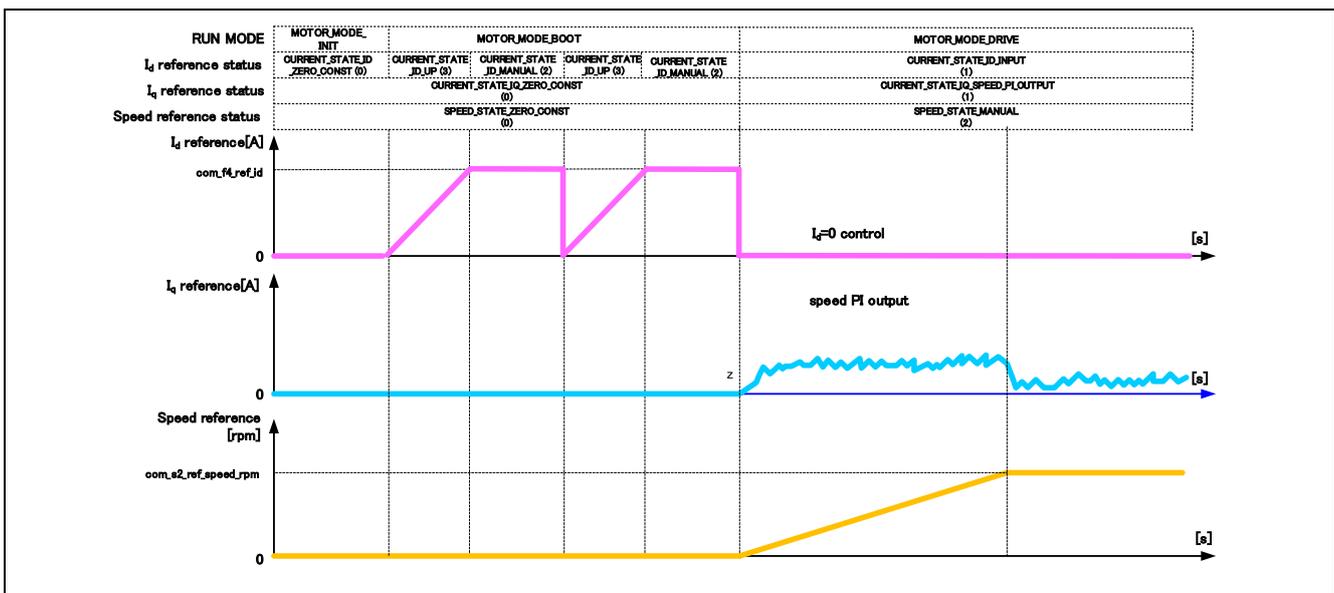


図 5-14 エンコーダ利用ベクトル制御の始動制御内容(速度制御時)

① エンコーダのみを用いた磁極位置の決定

位置センサとしてエンコーダを使用する場合、インクリメンタル式エンコーダでは絶対的な磁極位置情報が得られず、相対的な位置しかわかりません。そのため、始動時に初期の磁極位置が分かっている必要があります。そこで、図 5-15 のような手順で電流ベクトルを作ることによって磁石を引き込み、d 軸と電流ベクトルの向きを合わせることで、初期の磁極位置を定めます。また、この時の始動シーケンスを図 5-16 に記します。

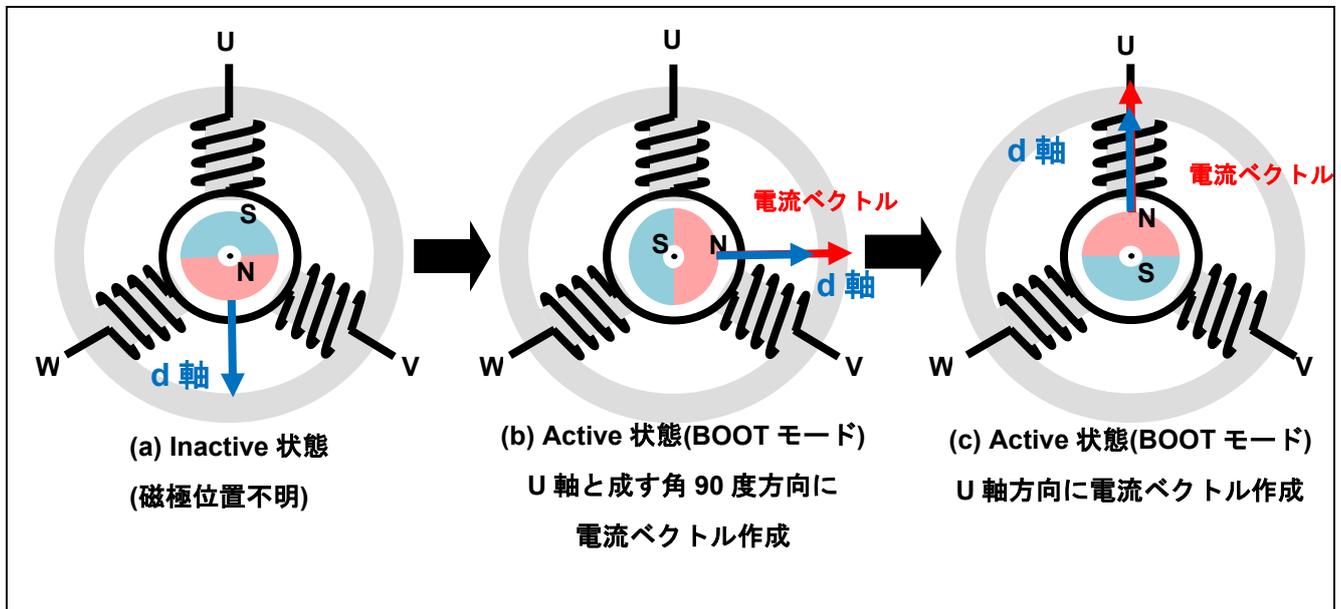


図 5-15 永久磁石位置の決定

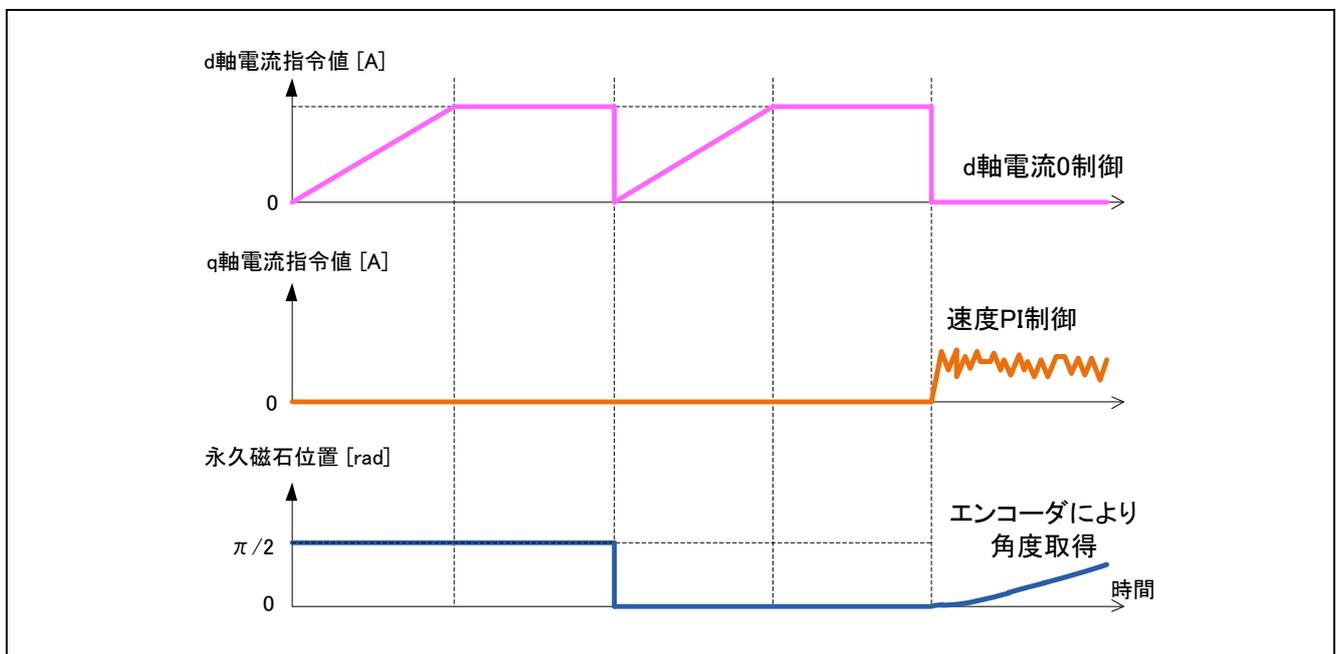


図 5-16 エンコーダ使用ベクトル制御における始動シーケンス(一例)

② ホールセンサを利用した磁極位置検出

エンコーダと合わせてホールセンサを使用した場合、磁極位置を検出することができます。ホールセンサを使用した場合の磁極位置の決定方法について図 5-17 に示します。

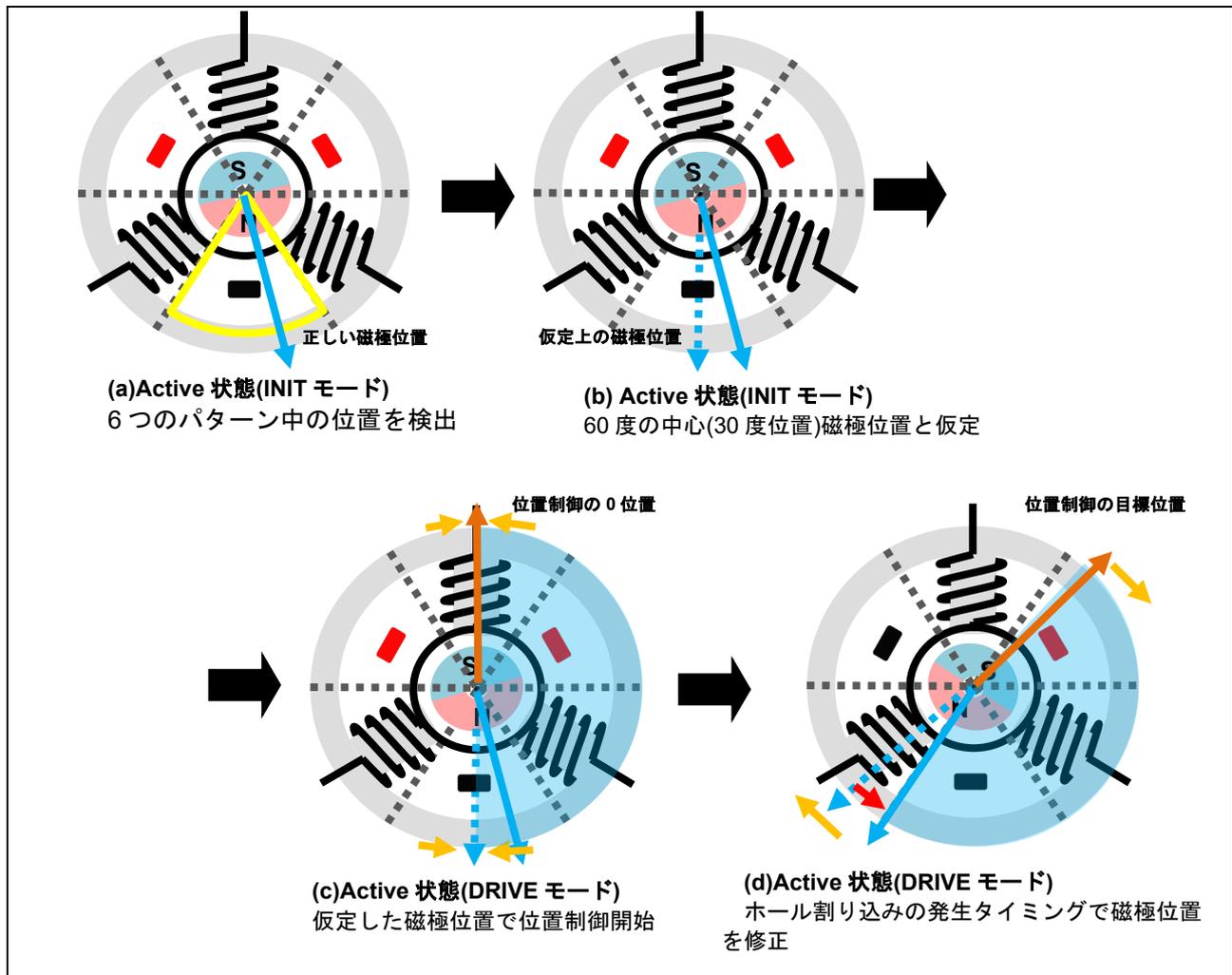


図 5-17 永久磁石の磁極位置の決定手順

- (a) ホールセンサから6つのパターンのどの60度の範囲にあるかを検出します。
- (b) 60度の中心30度位置を初期の磁極位置とします。
- (c) 位置制御を開始し、ホール割り込みが発生するまでは、正しい磁極位置から最大30度ずれた位置で制御を行います。
- (d) ホールエッジを検出した時点で、ベクトル制御での角度を正しい磁極位置に補正します。ホールセンサを併用することで、引き込み動作をせずに位置決め制御に移行できます。

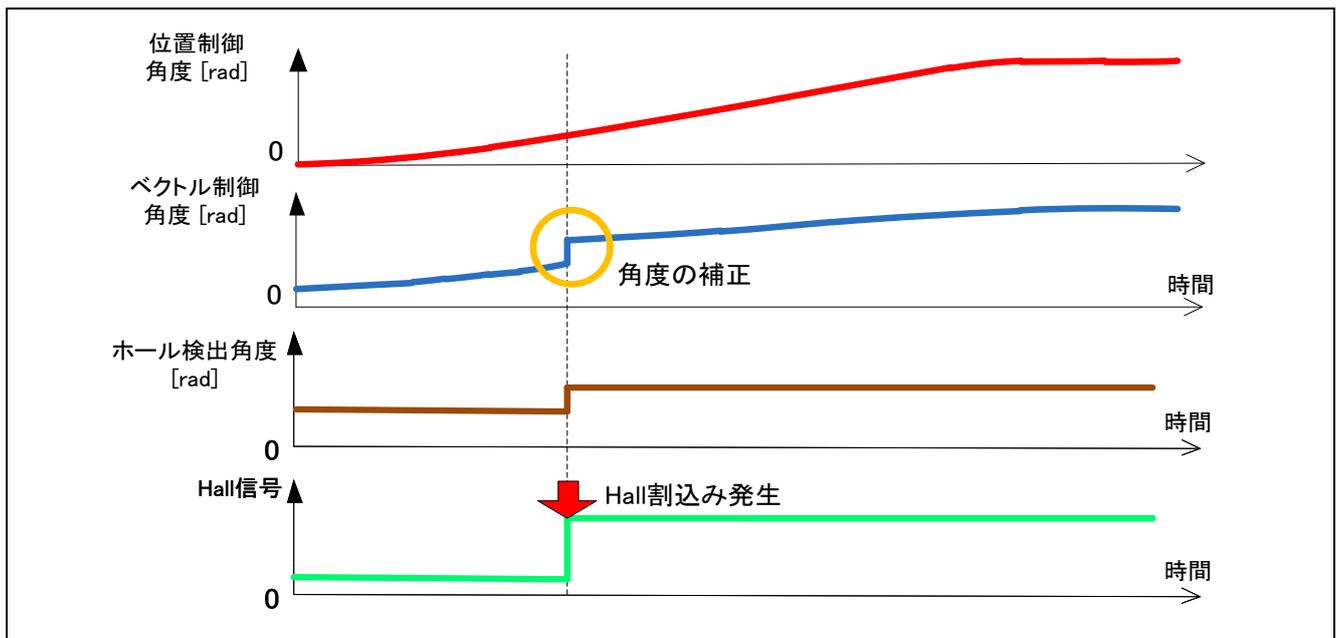


図 5-18 ホール使用初期位置検出における始動シーケンス

5.3 電流制御モジュール

電流制御モジュールは、入力された電流値からベクトル制御に必要な座標変換及びフィードバック制御を行い、PWM として出力する電圧を演算するモジュールです。また、サブモジュールとして変調と電圧誤差補償を本モジュールから制御します。

5.3.1 機能

電流制御モジュールの機能一覧を表 5-16 に示します。

表 5-16 電流制御モジュールの機能一覧

機能	説明
電流制御	電流指令値に追従するよう演算を行い、PWM 出力値を設定します。
電流オフセット調整	AD で検出した電流値のオフセット値を計算します。
電圧誤差補償	出力電圧のデッドタイムによる影響を補償します。
座標変換、逆変換	ベクトル制御を行うために検出した電流値に対して、座標変換を行います。演算結果に対して座標の逆変換を行い元の座標軸に戻します。
変調	PWM 信号を変調して電圧利用率を改善します。
非干渉制御	dq 軸の干渉を防ぐために干渉を打ち消す演算を行います。

5.3.2 モジュール構成図

モジュール構成図を図 5-19 に示します。

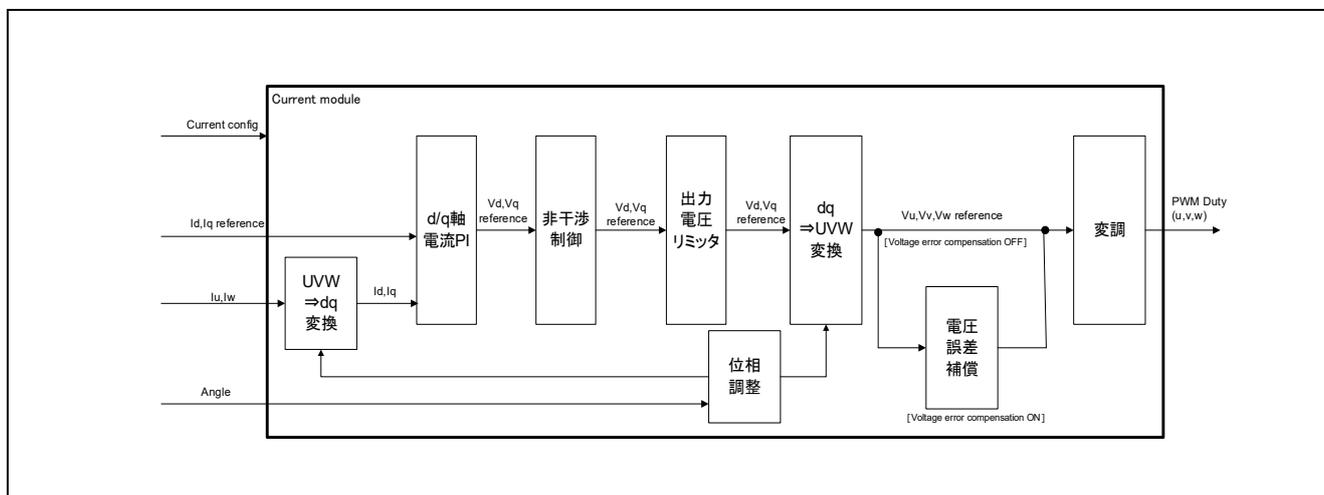


図 5-19 電流制御モジュール構成図

5.3.3 フローチャート

電流制御モジュールのループ処理フローチャートを図 5-20 に示します。

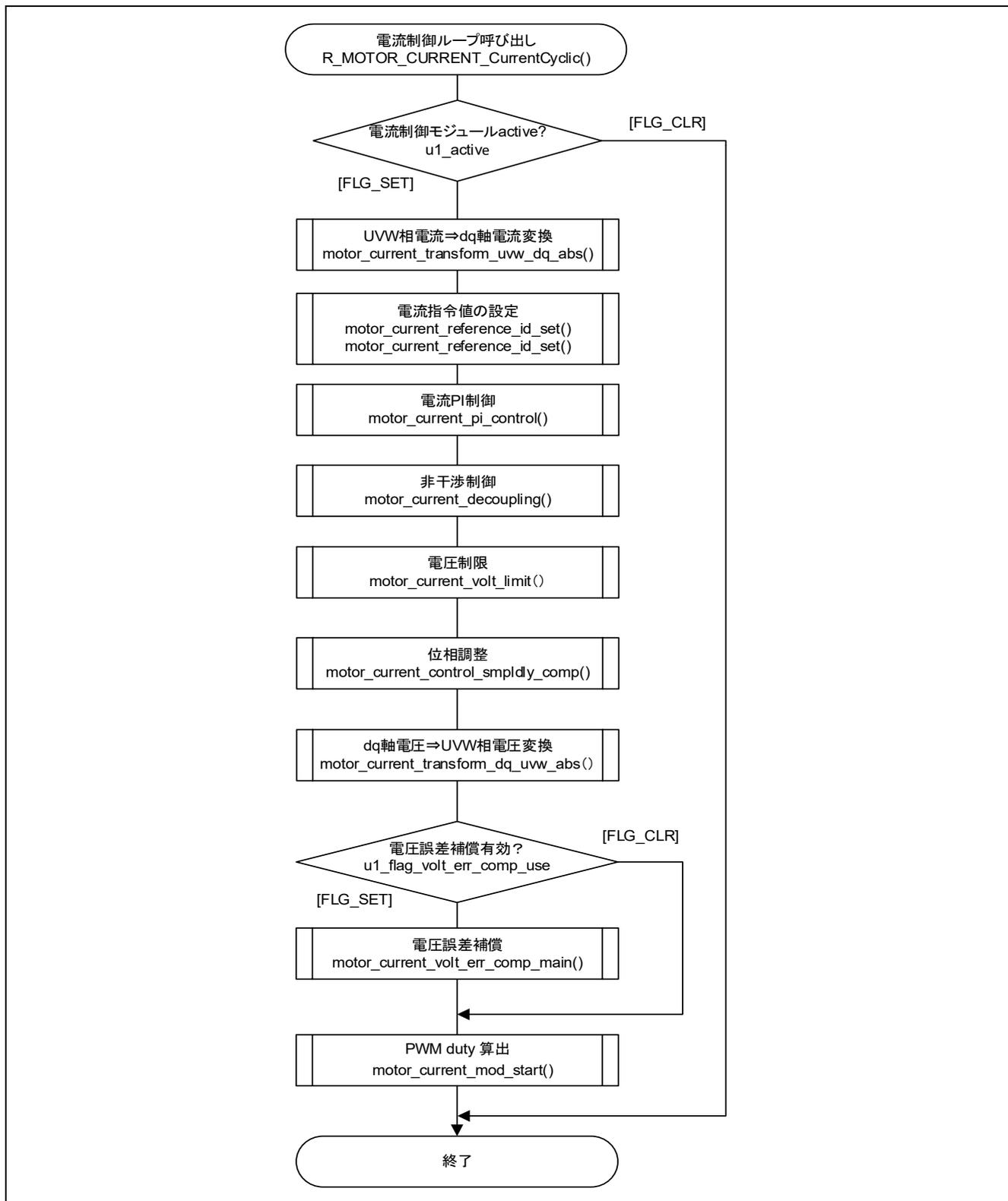


図 5-20 電流制御ループ処理フローチャート

5.3.4 API

電流制御モジュールの API 一覧を表 5-17 に示します。

表 5-17 API 一覧

API	説明
R_MOTOR_CURRENT_Open	電流制御モジュールのインスタンスを生成します。
R_MOTOR_CURRENT_Close	電流制御モジュールをリセット状態にします。
R_MOTOR_CURRENT_Reset	電流制御モジュールの初期化をします。
R_MOTOR_CURRENT_Run	電流制御モジュールをアクティブ状態にします。
R_MOTOR_CURRENT_ParameterSet	電流制御に使用する変数情報を入力します。
R_MOTOR_CURRENT_ParameterGet	電流制御結果の出力を取得します。
R_MOTOR_CURRENT_ParameterUpdate	電流制御モジュールの制御パラメータを更新します。
R_MOTOR_CURRENT_CurrentCyclic	電流制御を行います。
R_MOTOR_CURRENT_OffsetCalibration	電流検出のオフセット調整を行います。
R_MOTOR_CURRENT_CurrentOffsetRemove	電流検出オフセット値を除いた値を返します。
R_MOTOR_CURRENT_VoltErrCompParamSet	電圧誤差補償パラメータ設定を行います。

5.3.5 コンフィグレーション情報

電流制御モジュールで使用するコンフィグレーション情報を表 5-18 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-19 に示します。

表 5-18 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	CURRENT_CFG_VOLT_ERR_COMP	電圧誤差補償機能有効/無効 有効 : MTR_ENABLE 無効 : MTR_DISABLE
	CURRENT_CFG_MODULATION_METHOD	変調方式 MOD_METHOD_SPWM : 正弦波 PWM MOD_METHOD_SVPWM : 空間ベクトル PWM
	CURRENT_CFG_OFFSET_CALC_TIME	電流オフセットの測定時間設定
	CURRENT_CFG_PERIOD_MAG_VALUE	座標変換周期係数
	CURRENT_CFG_PI_INTEGRAL_LIMIT_VD	d 軸電流制限[V] INVERTER_CFG_INPUT_V : (最大入力電圧)は、r_motor_inverter_cfg.h で定義しています。
	CURRENT_CFG_PI_INTEGRAL_LIMIT_VQ	q 軸電流制限[V]
	CURRENT_CFG_OMEGA	電流制御系固有周波数[Hz]
	CURRENT_CFG_ZETA	電流制御系減衰係数
	CURRENT_CFG_REF_ID_OPENLOOP	オープンループ時の d 軸電流指令値[A]
	CURRENT_CFG_ID_UP_STEP_TIME	d 軸電流指令値加算時間設定

表 5-19 コンフィグレーション情報初期値一覧

マクロ名	設定
CURRENT_CFG_VOLT_ERR_COMP	MTR_ENABLE
CURRENT_CFG_MODULATION_METHOD	MOD_METHOD_SVPWM
CURRENT_CFG_OFFSET_CALC_TIME	512.0f
CURRENT_CFG_PERIOD_MAG_VALUE	1.0f
CURRENT_CFG_PI_INTEGRAL_LIMIT_VD	INVERTER_CFG_INPUT_V * 0.5f
CURRENT_CFG_PI_INTEGRAL_LIMIT_VQ	INVERTER_CFG_INPUT_V * 0.5f
CURRENT_CFG_OMEGA	300.0f
CURRENT_CFG_ZETA	1.0f
CURRENT_CFG_REF_ID_OPENLOOP	1.5f
CURRENT_CFG_ID_UP_STEP_TIME	2560.0f

5.3.6 構造体・変数情報

電流制御モジュールで使用する構造体・変数一覧を表 5-20 に示します。電流制御モジュールでは API のインスタンス確保にて、電流制御モジュール用構造体(g_st_cc)を定義します。

表 5-20 構造体・変数一覧

構造体	変数	説明
st_current_control_t 電流制御モジュール用 構造体	u1_active	電流制御モジュールのアクティブ状態
	u1_flag_volt_err_comp_use	電圧誤差補償機能の有効/無効
	u1_state_id_ref	始動時の d 軸ステータス
	u1_state iq_ref	始動時の q 軸ステータス
	u1_flag_offset_calc	電流オフセット計算のフラグ
	u2_offset_calc_time	電流オフセット調整時の測定時間設定
	u2_crnt_offset_cnt	電流オフセット調整時の測定回数
	f4_ctrl_period	電流制御周期(期間)[s]
	f4_refu	u 相指令電圧[V]
	f4_refv	v 相指令電圧[V]
	f4_refw	w 相指令電圧[V]
	f4_vd_ref	d 軸電圧指令値[V]
	f4_vq_ref	q 軸電圧指令値[V]
	f4_id_ref	d 軸電流指令値[A]
	f4 iq_ref	q 軸電流指令値[A]
	f4_id_ad	d 軸電流値[A]
	f4 iq_ad	q 軸電流値[A]
	f4_lim iq	q 軸電流制限値[A]
	f4_offset_iu	u 相オフセット電流値[A]
	f4_offset_iw	w 相オフセット電流値[A]
	f4_sum_iu_ad	u 相電流合計値[A]
	f4_sum_iw_ad	w 相電流合計値[A]
	f4_vdc_ad	母線電圧値[V]
	f4_iu_ad	u 相電流値[A]
	f4_iv_ad	v 相電流値[A]
	f4_iw_ad	w 相電流値[A]
	f4_modu	u 相デューティ比
	f4_modv	v 相デューティ比
	f4_modw	w 相デューティ比

構造体	変数	説明
st_current_control_t 電流制御モジュール用構造体	f4_speed_rad	速度[rad/s]
	f4_rotor_angle_input_rad	ロータ角度[rad]
	f4_id_up_step	id 設定時の変化量[A]
	f4_ref_id_ctrl	d 軸電流指令値 [A]
	f4_ref_iq_ctrl	q 軸電流指令値[A]
	f4_ol_ref_id	オープンループ時の d 軸電流指令値[A]
	f4_va_max	dq 軸上の最大電圧[V]
	f4_id_ref_buff	d 軸電流指令値のバッファ値[A]
	st_mod	変調用構造体
	st_volt_comp	電圧誤差補償用構造体
	st_pi_id	d 軸の pi 制御用構造体
	st_pi_iq	q 軸の pi 制御用構造体
	st_rotor_angle_t	ロータ情報の構造体
st_motor	モータパラメータの構造体	
st_current_cfg_t 電流制御モジュール制御パラメータ設定用構造体	u2_offset_calc_time	オフセット計算時間設定
	f4_ctrl_period	制御周期[s]
	f4_current_omega_hz	電流制御系固有周波数[Hz]
	f4_current_zeta	電流制御系減衰係数
	u1_flag_volt_err_comp_use	電圧誤差補償有効/無効
	f4_id_up_step	d 軸電流の増加量
	f4_ol_ref_id	オープンループ時の d 軸電流指令値[A]
	st_motor	モータパラメータの構造体
st_current_output_t 電流制御モジュール出力用構造体	u1_flag_offset_calc	電流オフセットフラグ
	f4_modu	u 相デューティ比
	f4_modv	v 相デューティ比
	f4_modw	w 相デューティ比
	f4_neutral_duty	オフセット測定時のデューティ比
	f4_va_max	dq 軸上の最大電圧[v]

構造体	変数	説明
st_current_input_t 電流制御モジュール入 力用構造体	u1_state_id_ref	d 軸ステータス
	u1_state_iq_ref	q 軸ステータス
	f4_rotor_angle_rad	ロータ角度[rad]
	f4_iu_ad	u 相電流値[A]
	f4_iv_ad	v 相電流値[A]
	f4_iw_ad	w 相電流値[A]
	f4_vdc_ad	母線電圧値[V]
	f4_speed_rad	速度[rad/s]
	f4_id_ref	d 軸電流指令値[A]
	f4_iq_ref	q 軸電流指令値[A]

5.3.7 マクロ定義

電流制御モジュールで使用するマクロ一覧を表 5-21 に示します。

表 5-21 マクロ一覧

ファイル名	マクロ名	定義値	説明
r_motor_current_api.h	CURRENT_STATE_ID_ZERO_CONST	0	d 軸用電流ステータス : d 軸電流 0 固定モード
	CURRENT_STATE_ID_INPUT	1	d 軸用電流ステータス : d 軸電流 指令入力モード
	CURRENT_STATE_ID_MANUAL	2	d 軸用電流ステータス : d 軸指令 固定モード
	CURRENT_STATE_ID_UP	3	d 軸用電流ステータス : d 軸電流 増加モード
	CURRENT_STATE_ID_DOWN	4	d 軸用電流ステータス : d 軸電流 減少モード
	CURRENT_STATE_IQ_ZERO_CONST	0	q 軸用電流ステータス : q 軸電流 0 固定モード
	CURRENT_STATE_IQ_SPEED_PI_OUTPUT	1	q 軸用電流ステータス : q 軸指令 PI 入力モード
	CURRENT_VERR_COMP_LIMIT	(MOTOR_MCU_CFG_CARRIER_FREQ * INVERTER_CFG_DEADTIME / 1000.0f)	電圧誤差補償期間リミッタ値 MOTOR_MCU_CFG_CARRIER は、r_motor_module_cfg.h 参照。 INVERTER_CFG_DEADTIME は、r_motor_inverter_cfg.h 参照。

5.3.8 パラメータ調整・設定

(a) 電流制御系固有周波数と減衰係数の調整

電流制御モジュールでは、電流制御系固有周波数と電流制御系減衰係数を調整して制御のゲインを調整します。電流制御系固有周波数は、電流制御を行う頻度に比例して設定してください。電流制御周波数の約 1/10 まで設定できますが、位置検出と電流検出のノイズなどを考慮し、低くする場合があります。

電流制御系減衰係数は、0.7~1.0 が常用範囲です。1.0 に近いほど安定で緩やかな応答になります。

電流制御系固有周波数と電流制御系減衰係数は、電流制御モジュール制御パラメータ設定用構造体 (st_current_cfg_t) の以下変数に値を設定し、電流制御モジュールの制御パラメータ更新用 API (R_MOTOR_CURRENT_ParameterUpdate) を使用して値の設定・更新をしてください。

電流制御系固有周波数と電流制御系減衰係数は、RMW から調整することが可能です。

電流制御系固有周波数 : f4_current_omega_hz (表 5-20 参照)

電流制御系減衰係数 : f4_current_zeta (表 5-20 参照)

から設定してください。

(b) 電流制御用パラメータの設定

電流制御モジュールでは、制御周期とモータのパラメータを使用するため、制御パラメータ設定 (R_MOTOR_CURRENT_ParameterUpdate) を使用して、各パラメータを更新することが可能です。設定項目は、電流制御モジュール制御パラメータ設定用構造体 (st_current_cfg_t) を参照ください。

(c) 電流制御用パラメータの初期値設定

電流制御モジュールのコンフィグレーション情報を r_motor_module_cfg.h で設定することができます。設定した値が初期値となり、システム起動時に適用されます。設定する項目は 5.3.7 マクロ定義を参照してください。

5.4 変調（電流制御モジュール）

サンプルプログラムでは、モータへの入力電圧はパルス幅変調（PWM）によって生成します。本モジュールでは、PWM Duty 比の算出を行います。また、電圧利用率を上げるために、変調を行った電圧を出力できます。電流制御モジュールの API を通して変調の動作を設定します。

5.4.1 機能説明

本モジュールでは、2 種類のパルス幅変調駆動方式から選択できます。

(a) 正弦波変調(MOD_METHOD_SPWM)

変調率 m を以下のように定義します。

$$m = \frac{V}{E}$$

m : 変調率 V : 指令値電圧 E : インバータ母線電圧

(b) 空間ベクトル変調(MOD_METHOD_SVPWM)*

永久磁石同期モータのベクトル制御において、一般的に所望の各相電圧指令値は正弦波状に生成します。ところが、そのまま PWM 生成のための変調波として使用すると、実際にモータに印加される電圧のインバータ母線電圧に対する電圧利用率は線間電圧換算で最大 86.7[%]となってしまいます。そこで、下記式にあるように各相電圧指令値の最大値と最小値の平均値を算出し、それらを各相電圧指令値から減算したものを変調波として使用します。その結果、変調波の最大振幅は $\sqrt{3}/2$ 倍となり、線間電圧はそのままに電圧利用率は 100[%]となります。

$$\begin{pmatrix} V'_u \\ V'_v \\ V'_w \end{pmatrix} = \begin{pmatrix} V_u \\ V_v \\ V_w \end{pmatrix} + \Delta V \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\therefore \Delta V = -\frac{V_{max} + V_{min}}{2}, V_{max} = \max\{V_u, V_v, V_w\}, V_{min} = \min\{V_u, V_v, V_w\}$$

V_u, V_v, V_w : U, V, W 相電圧指令値

V'_u, V'_v, V'_w : PWM 生成用 U, V, W 相電圧指令値(変調波)

変調率 m を以下のように定義します。

$$m = \frac{V'}{E}$$

m : 変調率 V' : PWM生成用相電圧指令 E : インバータ母線電圧

5.4.2 コンフィグレーション情報

変調機能のコンフィグ情報一覧を表 5-22 に示します。

表 5-22 コンフィグレーション情報一覧

ファイル名	マクロ名	設定	説明
r_motor_module_cfg.h	CURRENT_CFG_MODULATION_METHOD	MOD_METHOD_D_SVPWM	パルス幅変調駆動方式

5.4.3 構造体

変調機能で使用する構造体一覧を表 5-23 に示します。

表 5-23 変数一覧

構造体	変数	説明
st_mod_t	f4_vdc	母線電圧値[V]
	f4_1_div_vdc	1/f4_vdc
	f4_voltage_error_ratio	電圧誤差比率
	f4_max_duty	最大 PWM デューティ比
	f4_min_duty	最小 PWM デューティ比
	f4_neutral_duty	PWM デューティ比中間値

5.4.4 マクロ定義

変調機能で使用するマクロ一覧を表 5-24 に示します。

表 5-24 マクロ一覧

ファイル名	マクロ名	定義値	説明
r_motor_current_modulation.h	MOD_DEFAULT_MAX_DUTY	1.0f	最大 PWM デューティ比
	MOD_METHOD_SPWM	0	パルス幅変調駆動方式 : 正弦波 PWM
	MOD_METHOD_SVPWM	1	パルス幅変調駆動方式 : 空間ベクトル PWM
	MOD_VDC_TO_VAMAX_MULT	0.6124f	入力電圧から最大電圧への変換係数
	MOD_SVPWM_MULT	1.155f	空間ベクトル PWM 用係数

5.4.5 パラメータ調整・設定

変調機能でユーザが設定するパラメータはありません。

5.5 電圧誤差補償（電流制御モジュール）

電圧誤差補償機能は、デッドタイムによる出力電圧の影響を補修する機能です。電流制御モジュールの API を通して動作します。

5.5.1 機能説明

電圧形 PWM 変換器では、上下アームのスイッチング素子間の短絡を防止するために、上下アーム 2 つの素子が同時にオフとなるデッドタイムを設けています。そのため電圧指令値と実際にモータに印加される電圧には誤差が生じ、制御精度が悪化します。そこでその誤差を低減するため、電圧誤差補償を実装します。

電圧誤差の電流依存性は、電流(向きと大きさ)とデッドタイム、使用するパワー素子のスイッチング特性に依存し、下記のような特性を持ちます。電圧誤差補償では、下記電圧誤差と逆の電圧パターンを電流に応じて電圧指令値に補償します。

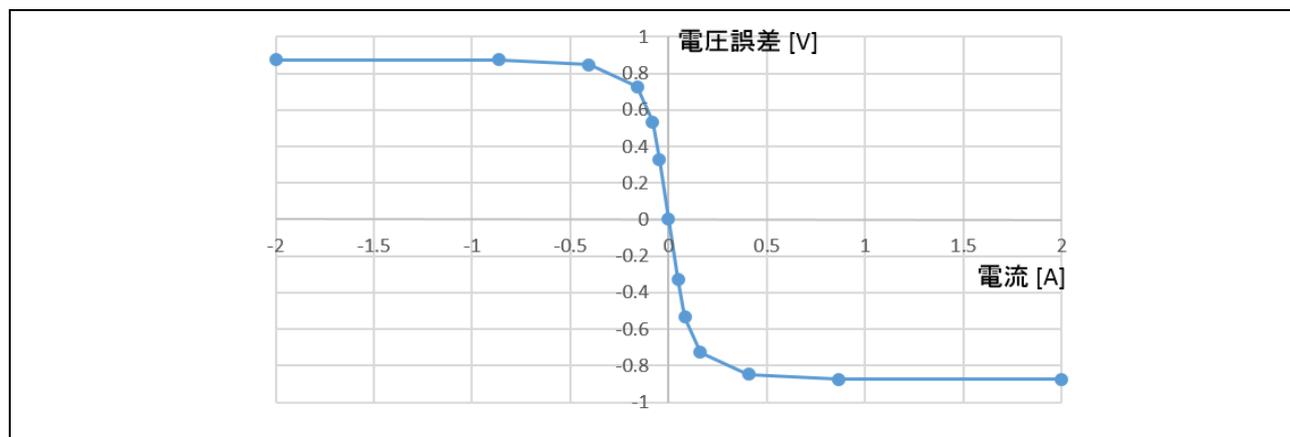


図 5-21 電圧誤差の電流依存性(一例)

5.5.2 コンフィグレーション情報

電圧誤差補償機能のコンフィグ情報一覧を表 5-25 に示します。

表 5-25 コンフィグレーション情報一覧

ファイル名	マクロ名	設定	説明
r_motor_inverter_cfg.h	INVERTER_CFG_COMP_V0	0.564f	電圧補償テーブル
	INVERTER_CFG_COMP_V1	0.782f	電圧補償テーブル
	INVERTER_CFG_COMP_V2	0.937f	電圧補償テーブル
	INVERTER_CFG_COMP_V3	1.027f	電圧補償テーブル
	INVERTER_CFG_COMP_V4	1.058f	電圧補償テーブル
	INVERTER_CFG_COMP_I0	0.022f	電圧補償テーブル
	INVERTER_CFG_COMP_I1	0.038f	電圧補償テーブル
	INVERTER_CFG_COMP_I2	0.088f	電圧補償テーブル
	INVERTER_CFG_COMP_I3	0.248f	電圧補償テーブル
	INVERTER_CFG_COMP_I4	0.865f	電圧補償テーブル

5.5.3 パラメータ調整・設定

(a) 電圧誤差機能有効フラグの設定

電流制御モジュールの制御パラメータ設定(R_MOTOR_CURRENT_ParameterUpdate)呼び出し時に、電圧誤差補償機能の有効/無効使用有無フラグ(u1_flag_volt_err_comp_use)を MTR_FLG_SET に設定することで機能が有効になります。無効にする場合は、上記フラグを MTR_FLG_CLR に設定してください。

5.6 速度制御モジュール

速度制御モジュールはモータが速度指令に追従するように制御するモジュールです。速度指令値の入力を受けて、電流指令値を出力します。また、サブモジュールの弱め磁束制御と外乱トルク・速度推定オブザーバ外を本モジュールから制御します。

5.6.1 機能

速度制御モジュールの機能一覧を表 5-26 に示します。

表 5-26 速度制御モジュールの機能一覧

機能	説明
速度制御	速度指令値に追従するよう演算を行い、電流指令値を出力します。
速度指令設定	速度モジュールに速度指令値を設定します。
弱め磁束制御設定	弱め磁束制御を用いて dq 軸電流指令値を演算・設定します。
外乱トルク・速度推定オブザーバ設定	モータモデルとトルク(q 軸)電流指令値及び検出した回転速度を用いて速度の推定を行います。
IPD 制御切り替え	PID 制御から IPD 制御モジュールを使用した IPD 制御へ切り替えを行います。

5.6.2 モジュール構成図

速度制御モジュールのモジュール構成図を図 5-22 に示します。

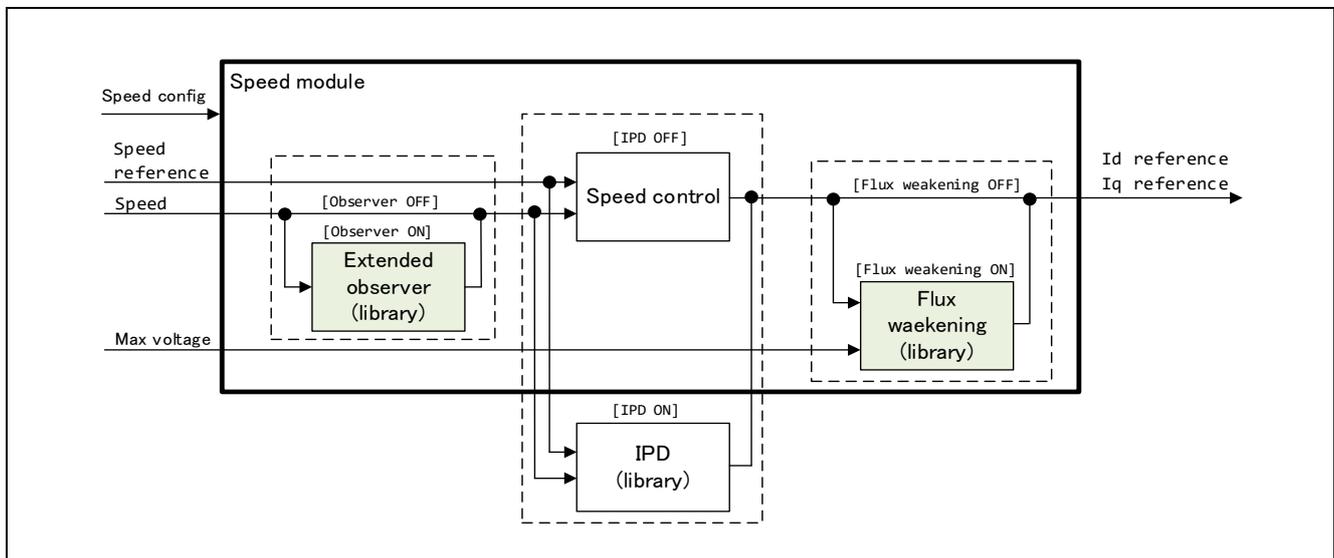


図 5-22 速度モジュール構成図

速度制御モジュールのサブモジュールである弱め磁束制御と外乱トルク・速度推定オブザーバの機能は 5.7 弱め磁束制御（速度制御モジュール） 5.8 外乱トルク・速度推定オブザーバ（速度制御モジュール）を参照ください。

5.6.3 フローチャート

速度制御のフローチャートを図 5-23 に示します。

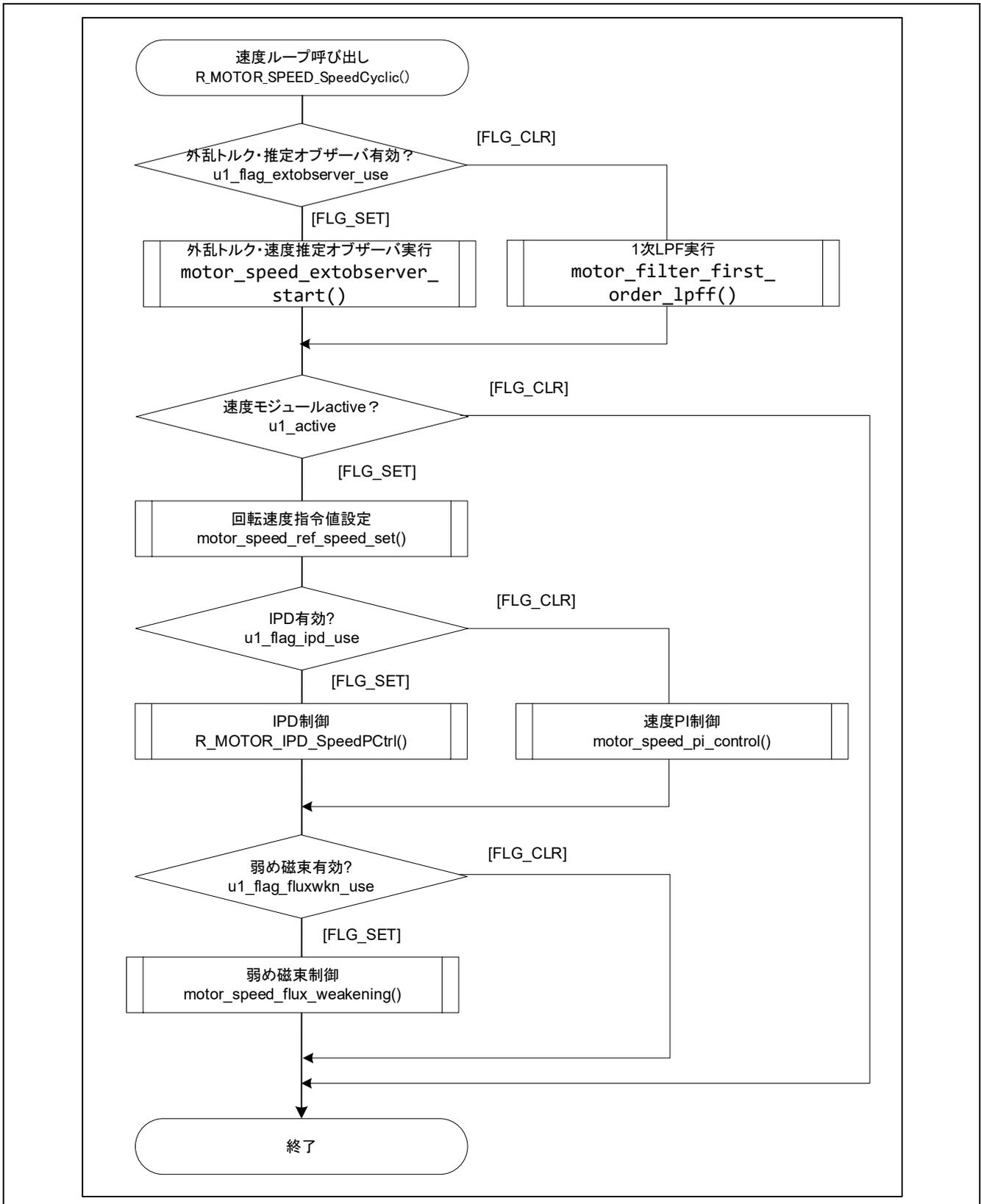


図 5-23 速度制御フローチャート

5.6.4 API

速度制御モジュールの API 一覧を表 5-27 に示します。

表 5-27 API 一覧

API	説明
R_MOTOR_SPEED_Open	速度モジュールのインスタンスを生成します。
R_MOTOR_SPEED_Close	モジュールをリセット状態にします。
R_MOTOR_SPEED_Reset	モジュールの初期化します。
R_MOTOR_SPEED_Run	モジュールをアクティブ状態にします。
R_MOTOR_SPEED_ParameterSet	速度制御に使用する変数情報を入力します。
R_MOTOR_SPEED_ParameterGet	速度制御結果の出力を取得します。
R_MOTOR_SPEED_ParameterUpdate	モジュールの制御パラメータを更新します。
R_MOTOR_SPEED_SpdRefSet	速度指令値を設定します。
R_MOTOR_SPEED_SpeedCyclic	速度制御を行います。
R_MOTOR_SPEED_IPDInstanceAddressSet	IPD 制御モジュールのアドレス設定を行います。
R_MOTOR_SPEED_IPDEnableSet	IPD 制御モジュールを有効にします。
R_MOTOR_SPEED_ExtObserverParameterUpdate	外乱トルク・速度推定オブザーバの制御パラメータを更新します。

5.6.5 コンフィグレーション情報

速度制御モジュールのコンフィグレーション情報一覧を表 5-28 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-29 に示します。

表 5-28 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	SPEED_CFG_FLUX_WEAKENING	弱め磁束制御の設定 有効：MTR_ENABLE 無効：MTR_DISABLE
	SPEED_CFG_OBSERVER	外乱トルク・速度推定オブザーバの設定 有効：MTR_ENABLE 無効：MTR_DISABLE
	SPEED_CFG_CTRL_PERIOD	制御周期設定 [s]
	SPEED_CFG_OMEGA	速度制御系固有周波数 [Hz]
	SPEED_CFG_ZETA	速度制御系減衰係数
	SPEED_CFG_LPF_OMEGA	速度制御系の LPF 帯域 [Hz]
	SPEED_CFG_SPEED_LIMIT_RPM	速度制限値 [rpm]
	SPEED_CFG_RATE_LIMIT_RPM	加速度制限 [rpm/s]
	SPEED_CFG_SOB_OMEGA	外乱トルク・速度推定オブザーバの固有周波数 [Hz]

表 5-29 コンフィグレーション情報初期値一覧

マクロ名	設定値
SPEED_CFG_FLUX_WEAKENING	MTR_DISABLE
SPEED_CFG_OBSERVER	MTR_DISABLE
SPEED_CFG_CTRL_PERIOD	0.0005f
SPEED_CFG_OMEGA	12.0f
SPEED_CFG_ZETA	1.0f
SPEED_CFG_LPF_OMEGA	250.0f
SPEED_CFG_SPEED_LIMIT_RPM	4500.0f
SPEED_CFG_RATE_LIMIT_RPM	1000.0f
SPEED_CFG_SOB_OMEGA	100.0f

5.6.6 構造体・変数情報

速度制御モジュールの構造体・変数一覧を表 5-30 に示します。速度モジュールは API のインスタンス確保にて、速度モジュール用構造体(g_st_sc)を定義します。

表 5-30 構造体・変数一覧 1

構造体	変数	説明
st_speed_control_t	u1_active	モジュールの有効/無効選択
速度モジュール用 構造体	u1_state_speed_ref	速度指令値を決定するステート管理。本節のマクロに記載するステートを管理する。
	u1_flag_extobserver_use	外乱トルク・速度推定オブザーバの使用有無のフラグ
	u1_flag_ipd_use	IPD 制御モジュールの使用有無のフラグ
	u1_flag_fluxwkn_use	弱め磁束制御の使用有無のフラグ
	f4_speed_ctrl_period	速度ループの周期 [s]
	f4_ref_speed_rad_ctrl	制御用の速度指令値 [rad/s]
	f4_ref_speed_rad	位置制御時の位置モジュール出力の速度指令値 [rad/s]
	f4_ref_speed_rad_manual	速度制御時のユーザの速度指令値設定値 [rad/s]
	f4_speed_rad_ctrl	速度制御モジュール内で演算する速度 [rad/s]
	f4_speed_rad	入力された速度 [rad/s]
	f4_max_speed_rad	最大速度 [rad/s]
	f4_speed_rate_limit_rad	速度の変化量の制限値 [rad/s]
	f4_speed_obsrv_rad	外乱トルク・速度推定オブザーバで演算された速度 [rad/s]
	f4_id_ref_output	d 軸電流指令値 [A]
	f4_iq_ref_output	q 軸電流指令値 [A]
	f4_va_max	dq 軸上の最大電圧 [V]
	f4_id_ad	d 軸電流値 [A]
	f4_iq_ad	q 軸電流値 [A]
	st_motor	モータ定数用構造体
	st_pi_speed	PI 制御用構造体
st_extobs	外乱トルク・速度推定オブザーバ用構造体	
st_fluxwkn	弱め磁束制御用構造体	
st_slpf	LPF 用構造体	
*p_st_ipd	IPD 制御用構造体	

表 5-31 構造体・変数一覧 2

構造体	変数	説明
st_speed_cfg_t 速度モジュール制御パラメータ設定用構造体	u1_flag_extobserver_use	外乱トルク・速度推定オブザーバの使用有無フラグ
	u1_flag_fluxwkn_use	弱め磁束制御の使用有無フラグ
	f4_max_speed_rpm	最大速度 [rpm]
	f4_speed_ctrl_period	速度制御の周期 [s]
	f4_speed_rate_limit_rpm	速度の変化量の制限値 [rpm]
	f4_speed_omega_hz	速度制御系固有周波数 [Hz]
	f4_speed_zeta	速度制御系減衰係数
	f4_speed_lpf_hz	速度制御用 LPF [Hz]
st_speed_input_t 速度モジュール入力用構造体	u1_state_speed_ref	速度指令ステータス
	f4_ref_speed_rad	速度指令値 [rad/s]
	f4_speed_rad	入力する速度 [rad/s]
	f4_va_max	dq 軸における最大電圧 [V]
st_speed_output_t 速度モジュール出力用構造体	f4_id_ref	d 軸電流指令値 [A]
	f4_iq_ref	q 軸電流指令値 [A]
	f4_ref_speed_rad_ctrl	PI 制御に使用する速度 [rad/s]
	f4_speed_rad_lpf	LPF 後の速度 [rad/s]
st_ext_observer_cfg_t 外乱オブザーバパラメータ設定用構造体	f4_extobs_omega	外乱トルク・速度推定オブザーバの固有周波数 [Hz]

5.6.7 マクロ定義

速度制御モジュールのマクロ一覧を表 5-32 に示します。

表 5-32 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_speed_api.h	SPEED_STATE_ZERO_CONST	0	速度モジュールのステート管理。速度指令値をゼロ固定になります。
	SPEED_STATE_POSITION_CONTROL_OUTPUT	1	速度モジュールのステート管理。速度指令値が位置制御モジュールの出力となります。
	SPEED_STATE_MANUAL	2	速度モジュールのステート管理。速度指令値がユーザ設定値になります。

5.6.8 パラメータ調整・設定

(a) 速度制御系固有周波数と減衰係数の調整

速度制御モジュールでは、速度制御系固有周波数と速度制御系減衰係数を調整して制御のゲインを調整します。速度制御系固有周波数を高くすると、応答性が向上し指令速度に対する速度の追従性が向上します。速度制御系固有周波数は電流制御との干渉を防ぐため、設定できる上限が電流制御系の固有周波数の 1/3 となっています。速度制御系減衰係数は 0.7~1.0 は常用範囲とし、値 1 に近いほど安定で緩やかな応答になります。速度の応答を確認しながら調整を行ってください。

速度制御系固有周波数と速度制御系減衰係数は、速度モジュール制御パラメータ設定用構造体 (st_speed_cfg_t) の以下変数に値を設定し、速度モジュールの制御パラメータを更新用 API (R_MOTOR_SPEED_ParameterUpdate) を使用して値の設定・更新をしてください。

- 速度制御系固有周波数 : f4_speed_omega_hz (表 5-30 参照)
- 速度制御系減衰係数 : f4_speed_zeta (表 5-30 参照)

(b) 速度制御用パラメータの設定

速度制御モジュールでは、制御周期とモータのパラメータを使用するため、制御パラメータの設定 (R_MOTOR_SPEED_ParameterUpdate) を使用して、各パラメータを更新することが可能です。設定項目は、速度モジュール制御パラメータ設定用構造体 (st_speed_cfg_t) を参照ください。

(c) 速度制御用パラメータの初期値設定

速度制御モジュールのコンフィグレーション情報を r_motor_module_cfg.h で設定することができます。設定した値が初期値となり、システム起動時に適用されます。設定する項目は 5.6.5 を参照ください。

5.7 弱め磁束制御（速度制御モジュール）

弱め磁束制御のモジュールは速度制御モジュールのサブモジュールです。回転子に磁石を持つモータを駆動すると、回転子の永久磁石磁束と回転速度に比例した誘起電圧が発生します。そして回転速度が上がり、誘起電圧が電源電圧と等しくなる、すなわち電圧が飽和すると、モータにそれより大きい電流を流せなくなり、回転速度が飽和します。この課題を解決する技術として弱め磁束制御があります。

5.7.1 機能説明

弱め磁束制御では、d 軸電流を負方向に印加することで、誘起電圧による電圧飽和の影響を抑え、高速回転化および高速回転域での出力向上を実現できます。

実際には図 5-24 に従って d 軸電流を決定し、制御を行います。

$$I_d = \frac{-\psi_a + \sqrt{\left(\frac{V_{om}}{\omega}\right)^2 - (L_q I_q)^2}}{L_d}$$

$$\therefore V_{om} = V_{amax} - I_a R$$

V_{om} : 誘起電圧制限値 [V]

V_{amax} : 電圧ベクトルの最大値 [V]

I_a : 電流ベクトルの大きさ [A]

図 5-24 弱め磁束制御における d 軸指令値の計算式

5.7.2 パラメータ調整・設定

本モジュールでユーザが設定するパラメータはありません。本モジュールを使用する場合は、速度モジュールの制御パラメータ更新用 API (R_MOTOR_SPEED_ParameterUpdate)にて、弱め磁束制御のフラグ(u1_flag_fluxwkn_use)を 1 に設定してください。

5.8 外乱トルク・速度推定オブザーバ（速度制御モジュール）

外乱トルク・速度推定オブザーバのモジュールは速度制御モジュールのサブモジュールです。オブザーバの機能を有効にすることで、速度指令に対する追従性の向上や速度リプルの低減に貢献します。

5.8.1 機能説明

ソフトウェアにより速度リプルを低減する手法として、オブザーバによる速度推定アルゴリズムを実装しています。オブザーバは q 軸の指令値 i_{q_ref} から計算したトルクと速度 ω を入力として、プラントモデルに基づいて推定速度 $\hat{\omega}$ を求めます。オブザーバにより速度リプルを低減させることが可能で、かつ通常のフィルタ処理に比べて制御系に影響を与えにくい特徴があります。センサの量子化誤差による影響や、ノイズの影響による速度リプルも低減することが可能です。

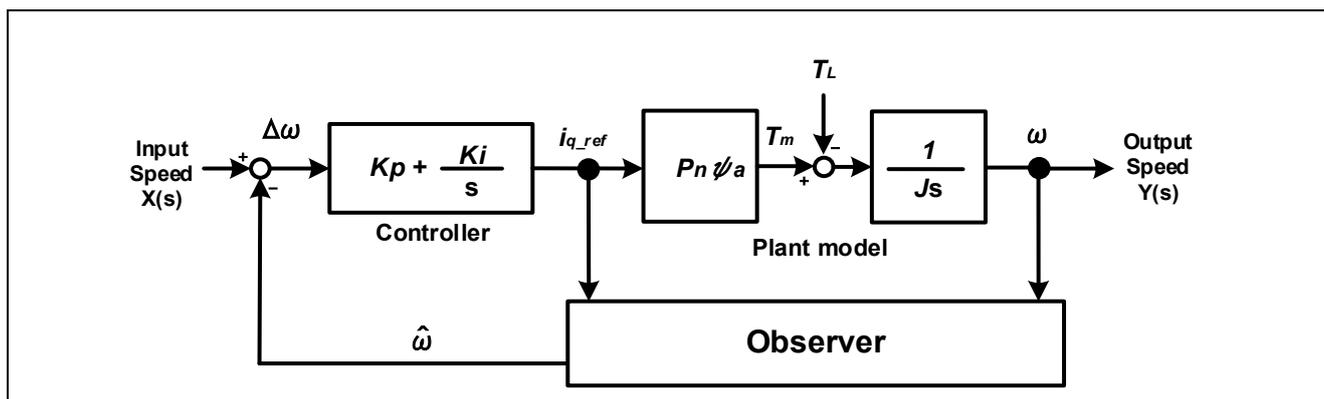


図 5-25 速度制御系のモデル

5.8.2 パラメータ調整・設定

本モジュールは、速度モジュールの API の外乱オブザーバの制御パラメータを更新用 API (R_MOTOR_SPEED_ExtObserverParameterUpdate) を使用してパラメータの設定を行います。設定するパラメータは以下の 3 種類です。

- モータのイナーシャ
- 外乱オブザーバの固有周波数
- オブザーバのサンプリング周期

イナーシャとオブザーバのサンプリング周期は、実際に制御で使用している正しい値を設定してください。外乱オブザーバの固有周波数は低くするほど速度リプルが低減されますが、速度指令の変化に対する応答が遅くなりますので、速度を確認しながら調整を行ってください。目安として速度制御系の固有周波数の 4~6 倍程度の周波数になります。

5.9 位置制御 モジュール

位置制御モジュールは、位置指令値と現在位置情報から速度指令値を求めます。位置制御には、P 制御と IPD 制御の 2 種類が選択できます。

サンプルプログラムでは位置指令の偏差から位置制御の指令値と駆動方式(三角・台形駆動)を決定する位置プロファイル処理を実装しています。また、速度指令へのフィードフォワード制御を実装し、速度制御の応答性を向上しています。

5.9.1 機能

位置制御モジュールの機能一覧を表 5-33 に示します。

表 5-33 位置制御モジュールの機能一覧

機能	説明
位置制御	位置指令値に追従するよう演算を行い、速度指令値を出力します。
速度フィードフォワード制御	速度指令へフィードフォワード(FF)制御を行います。
位置プロファイル	位置指令値の差分から位置指令値と駆動方式(三角・台形駆動)を制御します。
不感帯制御	ロータ位置が不感帯域に入ったかを検出して、位置指令値との差分を調整します。
IPD 制御切り替え	IPD 制御方式を用いた位置制御へ切り替えます。

5.9.2 モジュール構成

位置制御モジュールのモジュール構成図を図 5-26 に示します。

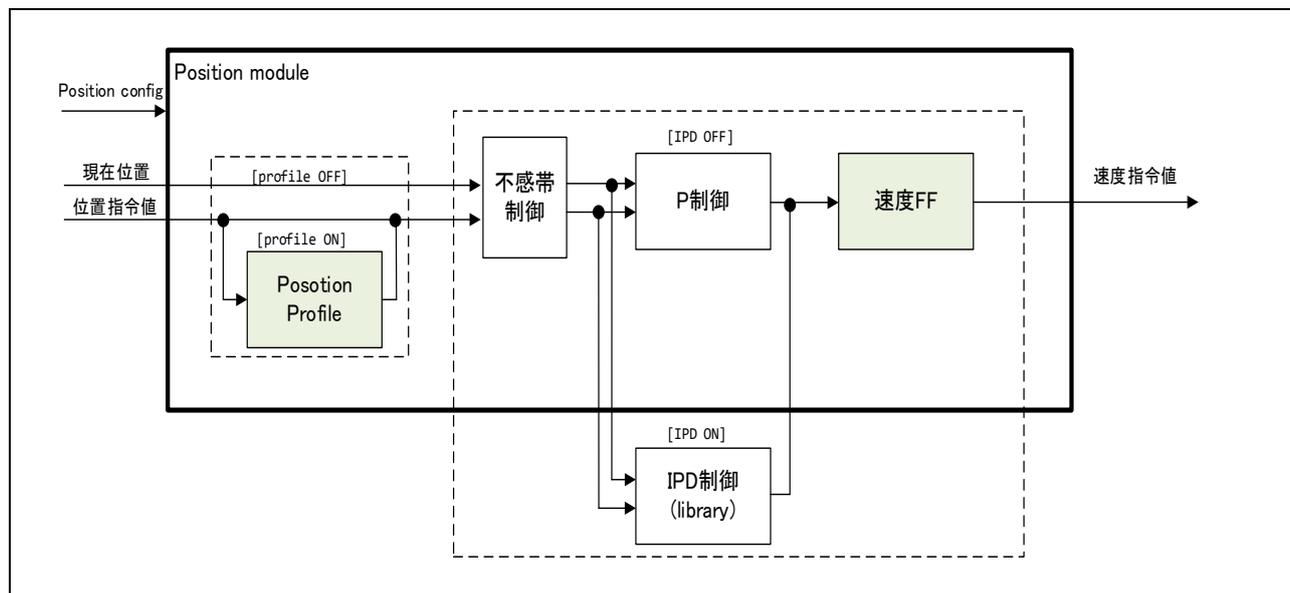


図 5-26 位置制御モジュール構成図

5.9.3 フローチャート

位置制御のフローチャートを図 5-27 に示します。

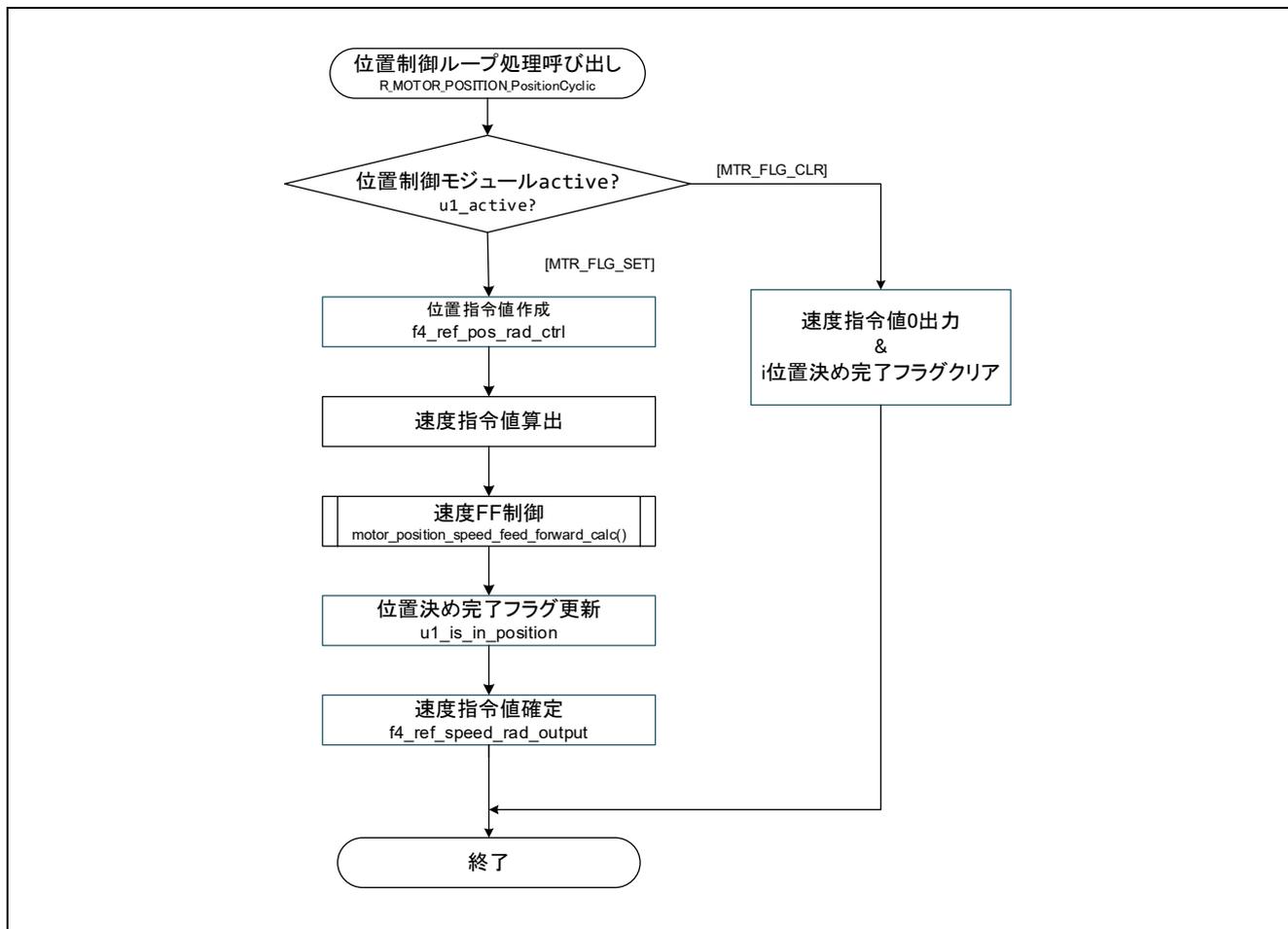


図 5-27 位置制御ループ処理フローチャート

位置指令作成処理の詳細フローは、図 5-28 を参照してください。

速度指令値算出処理の詳細フローは、図 5-29 を参照してください。

詳細フローに記載のモードについては、5.9.4 を参照してください。

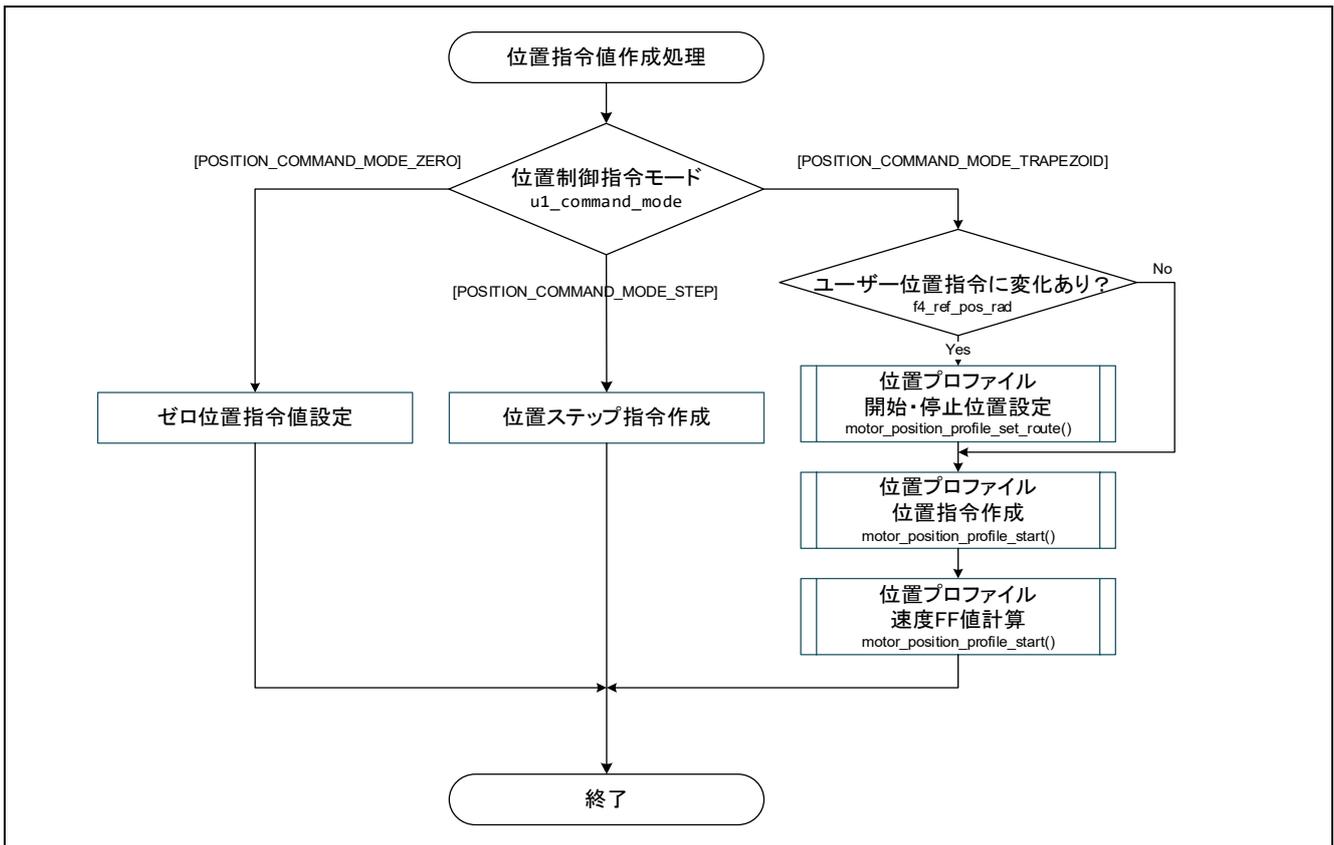


図 5-28 位置指令作成処理フローチャート

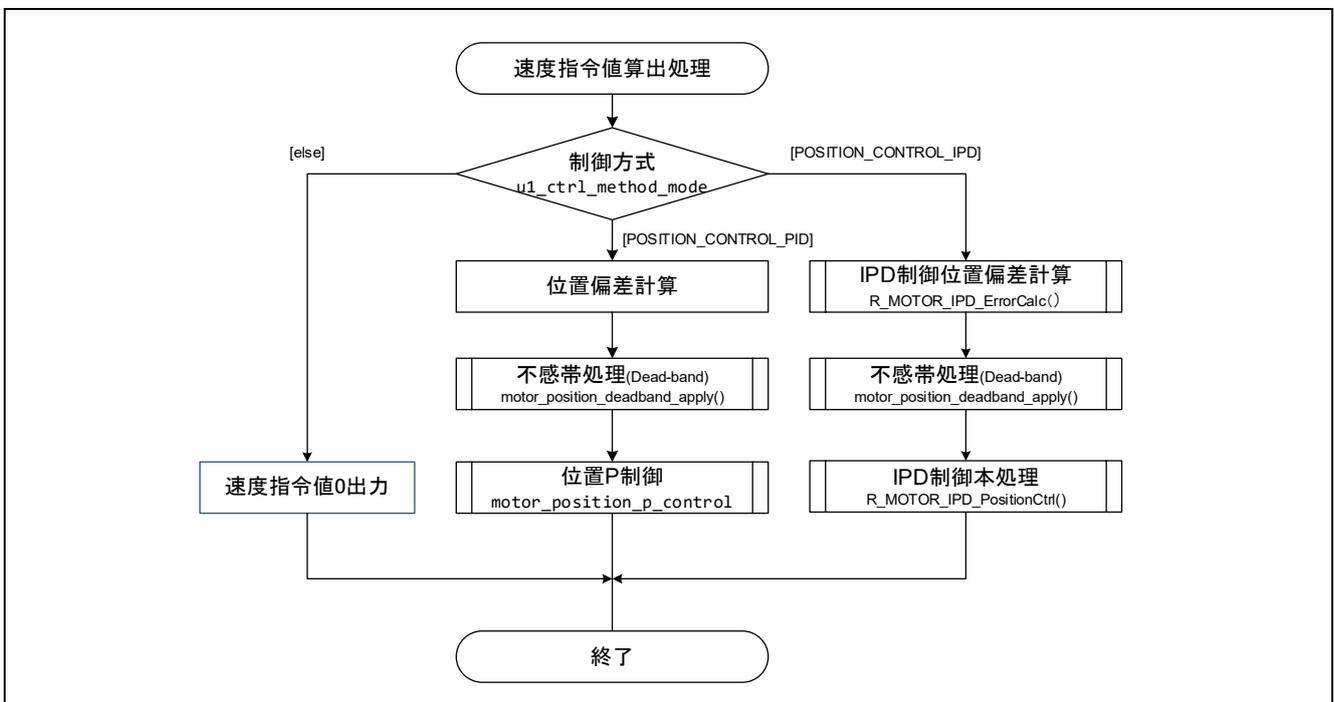


図 5-29 速度指令値算出処理

5.9.4 モード管理

(a) 位置指令用モード

モード一覧を表 5-34 に示します。

表 5-34 位置制御指令モード一覧

モード名	説明
POSITION_COMMAND_MODE_ZERO	ゼロ位置固定モード
POSITION_COMMAND_MODE_STEP	ステップモード
POSITION_COMMAND_MODE_TRAPEZOID	速度台形波方式

位置制御指令モード設定 API (R_MOTOR_POSITION_CommandModeSet) を使ってモードを切り替えます。

(b) 制御方式

制御方式一覧を表 5-35 に示します。

表 5-35 制御方式一覧

モード名	備考
POSITION_CONTROL_PID	PID 制御
POSITION_CONTROL_IPD	IPD 制御

IPD 制御モジュール有効設定 API (R_MOTOR_POSITION_IPDEnableSet) を使ってモードを切り変えます。

5.9.5 API

位置制御モジュールの API 一覧を表 5-36 に示します。

表 5-36 API 一覧

API	説明
R_MOTOR_POSITION_Open	位置制御モジュールのインスタンスを生成します。
R_MOTOR_POSITION_Close	位置制御モジュールをリセット状態にします。
R_MOTOR_POSITION_Reset	位置制御モジュールの初期化をします。
R_MOTOR_POSITION_Run	位置制御モジュールをアクティブ状態にします。
R_MOTOR_POSITION_PositionCyclic	位置制御ループ処理を行います。
R_MOTOR_POSITION_ParameterSet	位置制御ループで使用するパラメータを設定します。
R_MOTOR_POSITION_ParameterGet	位置制御モジュールの変数情報を取得します。
R_MOTOR_POSITION_ParameterUpdate	位置制御モジュールの制御パラメータを更新します。
R_MOTOR_POSITION_PosRefSet	位置指令を設定します。
R_MOTOR_POSITION_CommandModeSet	位置制御指令モードを設定します。
R_MOTOR_POSITION_IPDInstanceAddressSet	IPD 制御モジュールで生成したインスタンスのアドレスを登録します。
R_MOTOR_POSITION_IPDEnableSet	IPD 制御モジュールへの呼び出しを有効にします。
R_MOTOR_POSITION_Sync	位置情報を変更します。 前回停止位置から再開したい場合などで使用します。

5.9.6 コンフィグレーション情報

位置制御モジュールで使用するコンフィグレーション情報を表 5-37 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-38 に示します。

表 5-37 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	POSITION_CFG_CTRL_PERIOD	位置制御周期[s]
	POSITION_CFG_SPEED_FF_RATIO	速度フィードフォワード比例係数
	POSITION_CFG_DEAD_BAND	不感帯(位置センサパルス数)
	POSITION_CFG_INTERVAL_TIME	位置応答定常待ち時間(制御周期回数)
	POSITION_CFG_OMEGA	位置制御系固有周波数[Hz]
	POSITION_CFG_BAND_LIMIT	位置誤差ゼロ範囲(位置センサパルス数)

表 5-38 コンフィグ情報初期値一覧

マクロ名	設定
POSITION_CFG_CTRL_PERIOD	SPEED_CFG_CTRL_PERIOD
POSITION_CFG_SPEED_FF_RATIO	0.8f
POSITION_CFG_DEAD_BAND	1.0f
POSITION_CFG_INTERVAL_TIME	800.0f
POSITION_CFG_OMEGA	4.0f
POSITION_CFG_BAND_LIMIT	3.0f

5.9.7 構造体・変数情報

位置制御モジュールで使用する構造体一覧を表 5-39 に示します。位置制御モジュールは API のインスタンス確保にて、位置制御モジュール用構造体(g_st_pc)を定義します。

表 5-39 変数一覧

構造体	変数	説明
st_motor_position_t 位置制御モジュール用構造体	u1_is_in_position	位置決め完了フラグ
	u1_active	モジュールの Active 状態を示すフラグ
	u1_pos_command_mode	位置指令値作成モード
	u1_ctrl_method_mode	IPD/P 制御切り替え
	u2_pos_dead_band	デッドバンド(位置センサパルス数)
	u2_pos_band_limit	位置決め完了幅(位置センサパルス数)
	f4_pos_kp	位置 P 制御ゲイン係数
	f4_pos_err_rad	位置偏差[rad]
	f4_pos_rad	現在位置[rad]
	f4_ref_pos_rad	位置指令値 (上位階層からの指令) [rad]
	f4_ref_pos_pre_rad	位置指令値前回値[rad]
	f4_ref_pos_rad_ctrl	位置プロファイル処理後の位置指令[rad]
	f4_speed_ff_rad	速度フィードフォワード値[rad/s]
	f4_speed_ff_ratio	速度フィードフォワード比例係数
	f4_ref_speed_rad_output	速度指令値[rad/s]
	f4_max_speed_rad	最大速度[rad/s]
	f4_ctrl_period	制御周期[s]
	f4_mech_angle_per_sensor_cnt	位置センサの 1 カウントあたりの角度[rad]
	st_ppf	位置プロファイル用構造体
	st_motor	モータ定数用構造体
*p_st_ipd	IPD 制御モジュール生成インスタンスポインタ	

構造体	変数	説明
st_position_cfg_t	st_motor	モータ定数用構造体
位置制御モジュール制御パラメータ設定用構造体	u2_dead_band	デッドバンド(位置センサパルス数)
	u2_band_limit	位置決め完了幅(位置センサパルス数)
	u2_pos_interval_time	安定待ち時間(制御周期回数)
	f4_feedforward_ratio	速度フィードフォワード比例係数
	f4_position_omega_hz	位置制御周波数[Hz]
	f4_ctrl_period	制御周期[s]
	f4_mech_angle_per_sensor_cnt	位置センサの1パルスあたりの角度[rad]
	f4_max_speed_rad	最大速度[rad/s]
	f4_accel_time	加速時間[s]
	f4_posprof_max_speed_rad	位置プロファイル用速度最大値(機械角)[rad/s]
	st_position_input_t	f4_position_rad
位置制御モジュール入力用構造体		
st_position_output_t	f4_speed_ref	速度指令出力値 [rad/s]
	f4_position_err	位置偏差値[rad] 自動調整など外部で偏差判定したい場合に使用。
	u1_in_position	位置決め完了フラグ
位置制御モジュール出力用構造体		

5.9.8 マクロ定数

位置制御モジュールで使用するマクロ一覧を表 5-40 に示します。

表 5-40 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_position_api.h	POSITION_COMMAND_MODE_ZERO	0	位置指令用モード:ゼロ位置モード
	POSITION_COMMAND_MODE_STEP	1	位置指令用モード:ステップモード
	POSITION_COMMAND_MODE_Trapezoid	2	位置指令用モード:速度台形波方式
	POSITION_CONTROL_PID	0	制御モード:PID 制御
	POSITION_CONTROL_IPD	1	制御モード:IPD 制御

5.9.9 パラメータ調整・設定方法

(a) 位置制御系固有周波数の調整

位置制御モジュールでは、位置制御系固有周波数を調整して P 制御のゲインを調整します。設定できる上限が速度制御系の固有周波数の 1/3 となっています。

制御系固有周波数は、位置制御モジュール制御パラメータ設定用構造体(st_position_cfg_t)の以下変数に値を設定し、位置制御モジュールの制御パラメータ更新用 API (R_MOTOR_POSITION_ParameterUpdate) を使用して値の設定・更新をしてください。

- 位置制御系固有周波数 : f4_posprof_max_speed_rad (表 5-30 参照)

(b) 位置制御用パラメータの設定

位置制御モジュールでは、制御周期とモータのパラメータを使用するため、制御パラメータの設定 (R_MOTOR_POSITION_ParameterUpdate) を使用して、各パラメータを更新することが可能です。設定項目は、位置制御モジュール制御パラメータ設定用構造体(st_position_cfg_t)を参照してください。

(c) 位置制御用パラメータの初期値設定

位置制御モジュールのコンフィグ情報を r_motor_module_cfg.h で設定することができます。設定した値が初期値となり、システム起動時に適用されます。設定する項目は 5.9.6 コンフィグレーション情報を参照してください。

5.10 位置プロファイル(位置制御モジュール)

位置プロファイル機能は、位置制御モジュールの API を通して動作します。

5.10.1 機能説明

設定した位置指令値と加減速時間、最大速度を基に、制御周期ごとに位置指令値を再計算して速度指令値を制御する機能(移動平均型加減速方式)を実装しています。図 5-30 に概要を示します。位置偏差と加速時間から求めた速度が加速時の最大速度を超える場合、台形状の速度指令値となるよう位置指令値を作成します。図 5-30 に記載の変数名などについては、表 5-11 と表 5-12 を参照してください。

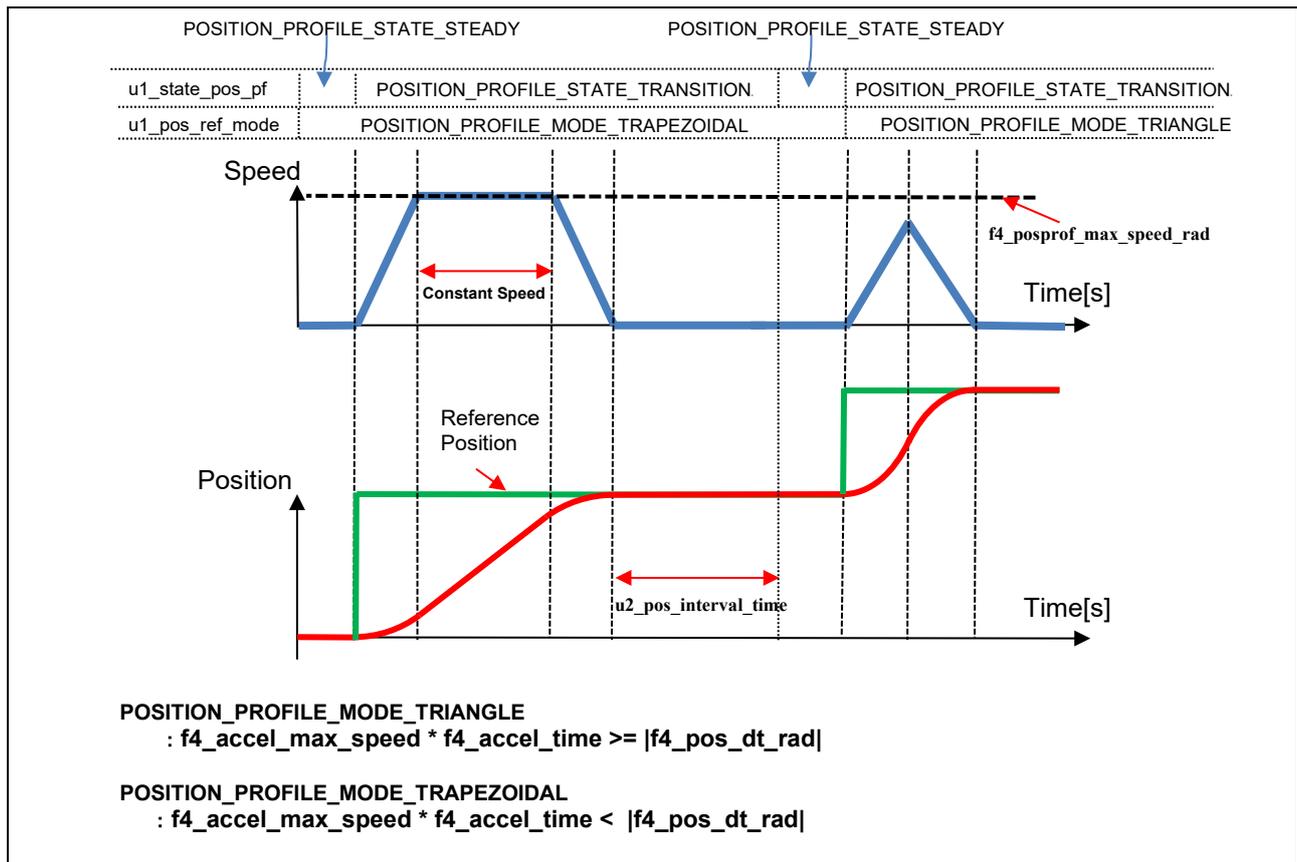


図 5-30 位置指令値の整形と三角形/台形型の速度指令値

5.10.2 コンフィグレーション情報

位置プロファイル機能で使用するコンフィグレーション情報を表 5-41 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-42 に示します。

表 5-41 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	POSITION_CFG_CTRL_PERIOD	位置制御周期[s]
	POSITION_CFG_INTERVAL_TIME	位置応答定常待ち時間(位置センサパルス数)
r_motor_targetmotor_cfg.h	MOTOR_CFG_MAX_SPEED_RPM	最大速度[rpm]

表 5-42 コンフィグ情報初期値一覧

マクロ名	設定
POSITION_CFG_CTRL_PERIOD	SPEED_CFG_CTRL_PERIOD
POSITION_CFG_INTERVAL_TIME	800.0f
MOTOR_CFG_MAX_SPEED_RPM	4000.0f

5.10.3 構造体

位置プロファイル機能で使用する構造体一覧を表 5-43 に示します。位置制御モジュール管理の変数で値を確認することができます。表 5-30 の st_ppf が該当します。

表 5-43 変数一覧

構造体	変数	説明
st_position_profiling_t	u1_state_pos_pf	位置プロファイルステータス
位置プロファイル用構造体	u1_pos_ref_mode	位置指令モード
	u2_interval_time	プロファイル周期(制御周期回数)
	u2_interval_time_buff	プロファイル周期バッファ(制御周期回数)
	u2_interval_time_cnt	プロファイル周期カウンタ
	f4_ctrl_period	制御周期[s]
	f4_accel_time	加速時間[s]
	f4_accel_time_buff	加速時間バッファ [s]
	f4_accel_time_inv	加速時間の逆数
	f4_max_accel_time	最大加速時間 [rad/s]
	f4_accel_max_speed	加速の最高速度 [rad/s]
	f4_accel_max_speed_buff	加速の最高速度バッファ[rad/s]
	f4_time_sec	プロファイリング用のタイマカウンタ
	f4_pos_st_rad	開始位置[rad]
	f4_pos_ed_rad	終了位置[rad]
	f4_pos_dt_rad	プロファイルの位置誤差 [rad]
	f4_pos_dt_time_sec	位置誤差/最高速度 [s]

5.10.4 マクロ定義

位置プロファイル機能で使用するマクロ一覧を表 5-44 に示します。

表 5-44 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_position_profiling.h	POSITION_PROFILE_ACCEL_TIME	0.3f	速度指令値の加速時間[s]
	POSITION_PROFILE_CTRL_TRIANGLE	0	三角波制御モード
	POSITION_PROFILE_CTRL_Trapezoidal	1	台形波制御モード
	POSITION_PROFILE_STEADY_STATE	0	待機状態
	POSITION_PROFILE_TRANSITION_STATE	1	遷移状態

5.10.5 パラメータ調整・設定

(a) 制御用パラメータの設定

位置制御モジュールの制御パラメータ更新(R_MOTOR_POSITION_ParameterUpdate)を使って、以下の変数を設定することで、加減速時間を考慮した位置指令値を作成することができます。図 5-30 で示した内容を調整するパラメータになります。

- 加速時間 : f4_accel_time
- 最大速度 : f4_posprof_max_speed_rad
- 整定待ち時間 : u2_pos_interval_time

5.11 IPD 制御モジュール

5.11.1 機能

位置制御系では、位置や速度の分解能が低い場合、位置決め時に振動が発生し続ける問題を生じます。これは、微小な位置偏差の変化に反応できないためです。位置決め時の振動を抑制するためには微小な変化を蓄積して偏差を零にするように働く積分要素が必要となります。

IPD 制御器は、偏差に対して積分のみが働き、比例と微分は操作量(制御器の出力)のみに働く制御方式となります。これによって、応答性を上げて位置決め時の振動を低減させることができます。

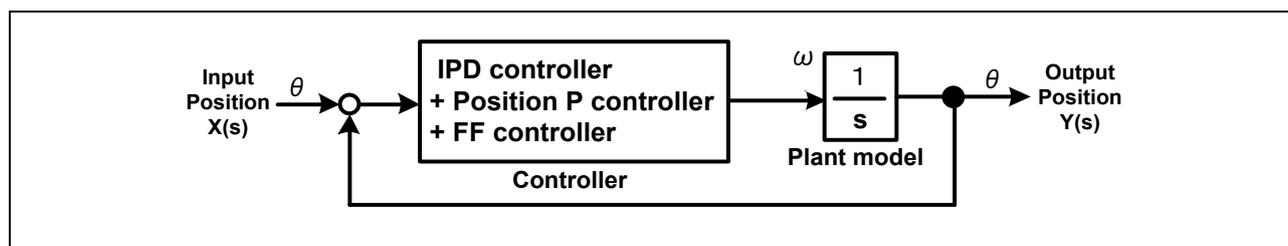


図 5-31 IPD 制御のモデル(位置)

実装している IPD 制御では、通常の比例制御、フィードフォワード制御と組み合わせています。

5.11.2 API

IPD 制御モジュールの API 一覧を表 5-45 に示します。

表 5-45 API 一覧

API	説明
R_MOTOR_IPD_Open	IPD 制御モジュールのインスタンスを生成します。
R_MOTOR_IPD_Close	モジュールをリセット状態にします。
R_MOTOR_IPD_Reset	モジュールの初期化します。
R_MOTOR_IPD_ParameterUpdate	モジュールの制御パラメータを更新します。
R_MOTOR_IPD_CtrlGainCalc	ゲインの計算を行います。
R_MOTOR_IPD_SpeedPCtrl	IPD 制御モジュールの速度制御を行います。
R_MOTOR_IPD_ErrorCalc	制御で使用する位置の偏差を計算します。
R_MOTOR_IPD_PositionCtrl	IPD 制御モジュールの位置制御を行います。
R_MOTOR_IPD_PositionSync	IPD 制御内部位置変数情報を更新します。

5.11.3 構造体・変数情報

IPD 制御モジュールの構造体・変数一覧を表 5-46 に示します。IPD 制御モジュールは API のインスタンス確保にて、IPD 制御モジュール用構造体(g_st_ipd)を定義します。

表 5-46 構造体・変数一覧

構造体	変数	説明
st_ipd_ctrl_t	u1_ipd_lpf_flag	LPF 有効フラグ
IPD 制御モジュール用構造体	f4_ref_pos_pre_rad_ctrl	位置指令前回値[rad]
	f4_ipd_pos_k	位置制御ゲイン係数
	f4_ipd_pos_1st_fb_rad	位置 feed-back (以下、FB)値[rad]
	f4_ipd_pos_1st_fb_pre_rad	位置 FB 前回値 [rad]
	f4_ipd_pos_2nd_fb_rad	位置 FB 値[rad] (2 段目)
	f4_ipd_ref_pos_rad	位置指令値[rad]
	f4_ipd_err_rad	位置 FB 偏差値[rad]
	f4_ipd_pos_fb_k	位置 FB ゲイン係数
	f4_ipd_pos_ff_rad	位置フィードフォワード値[rad]
	f4_ipd_pos_ff_k	位置フィードフォワードゲイン係数
	f4_ipd_pos_p_rad	位置 P 制御値[rad]
	f4_ipd_pos_kp	位置 P 制御ゲイン係数
	f4_ipd_pos_kp_ratio	位置 P 制御量倍率
	f4_ipd_pos_ff_ratio	位置フィードフォワードゲイン倍率
	f4_ipd_speed_k	速度ゲイン係数
	f4_ipd_speed_k_ratio	速度ゲイン倍率
	f4_ipd_ref_speed_rad	速度指令値[rad]
	f4_ipd_err_input_limit	位置偏差リミッタ[rad]
	f4_ipd_err_integrator_limit	位置誤差積分器リミッタ係数
	f4_ipd_lpf_omega	LPF 固有周波数[Hz]
f4_ipd_lpf_zeta	LPF 減衰係数	
st_ipd_2nd_lpf	2 次 LPF 構造体	

5.11.4 マクロ定義

IPD 制御モジュールのマクロ一覧を表 5-47 に示します。

表 5-47 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_ipd_api.h	IPD_LPF_OMEGA	200.0f	位置 LPF 用固有周波数[Hz]
	IPD_LPF_ZETA	1.0f	位置 LPF 用減衰係数
	IPD_SPEED_RATIO	2.5f	速度ゲイン係数
	IPD_POS_FF_RATIO	0.9f	位置フィードフォワード係数
	IPD_POS_KP_RATIO	0.3f	位置ゲイン係数
	IPD_POS_ERR_INPUT_LIMIT	10.0f	位置偏差リミッタ[rad]
	IPD_POS_ERR_INTEGRATOR_LIMIT_RATIO	1.0f	位置偏差積分器リミッタ係数
	IPD_LPF_FLAG	IPD_LPF_ON	LPF 有効フラグ

5.11.5 パラメータ調整・設定

(a) 演算係数の設定

制御パラメータ更新(R_MOTOR_IPD_ParameterUpdate)を使って行います。サンプルプログラムでの設定値を表 5-48 に示します。

表 5-48 パラメータ設定例

API 引数	説明	マネージャモジュール呼び出し時の設定値
f4_ipd_pos_kp_ratio	位置ゲイン係数	IPD_POS_KP_RATIO
f4_ipd_pos_ff_ratio	位置フィードフォワード係数	IPD_POS_FF_RATIO
f4_ipd_speed_k_ratio	速度ゲイン係数	IPD_SPEED_RATIO
f4_ipd_err_input_limit	位置偏差リミッタ	IPD_POS_ERR_INPUT_LIMIT
f4_ipd_err_integrator_limit	位置偏差積分器リミッタ係数	IPD_POS_ERR_INTEGRATOR_LIMIT_RATIO

5.12 センサモジュール（エンコーダ）

センサモジュールはモータの位置と速度を演算するモジュールです。サンプルプログラムでは、エンコーダのセンサモジュールとなっており、エンコーダ信号から位置と速度を演算して出力します。また、ホールセンサの入力を使用した始動にも対応しており、コンフィグレーションの設定にて切り替え可能です。

5.12.1 機能

センサモジュールの機能一覧を表 5-49 に示します。

表 5-49 センサモジュールの機能一覧

機能	説明
位置情報の取得	モータのロータ位置情報を取得します。
速度情報の取得	モータの回転速度を取得します。

5.12.2 モジュール構成図

センサモジュール構成図を図 5-32 に示します。

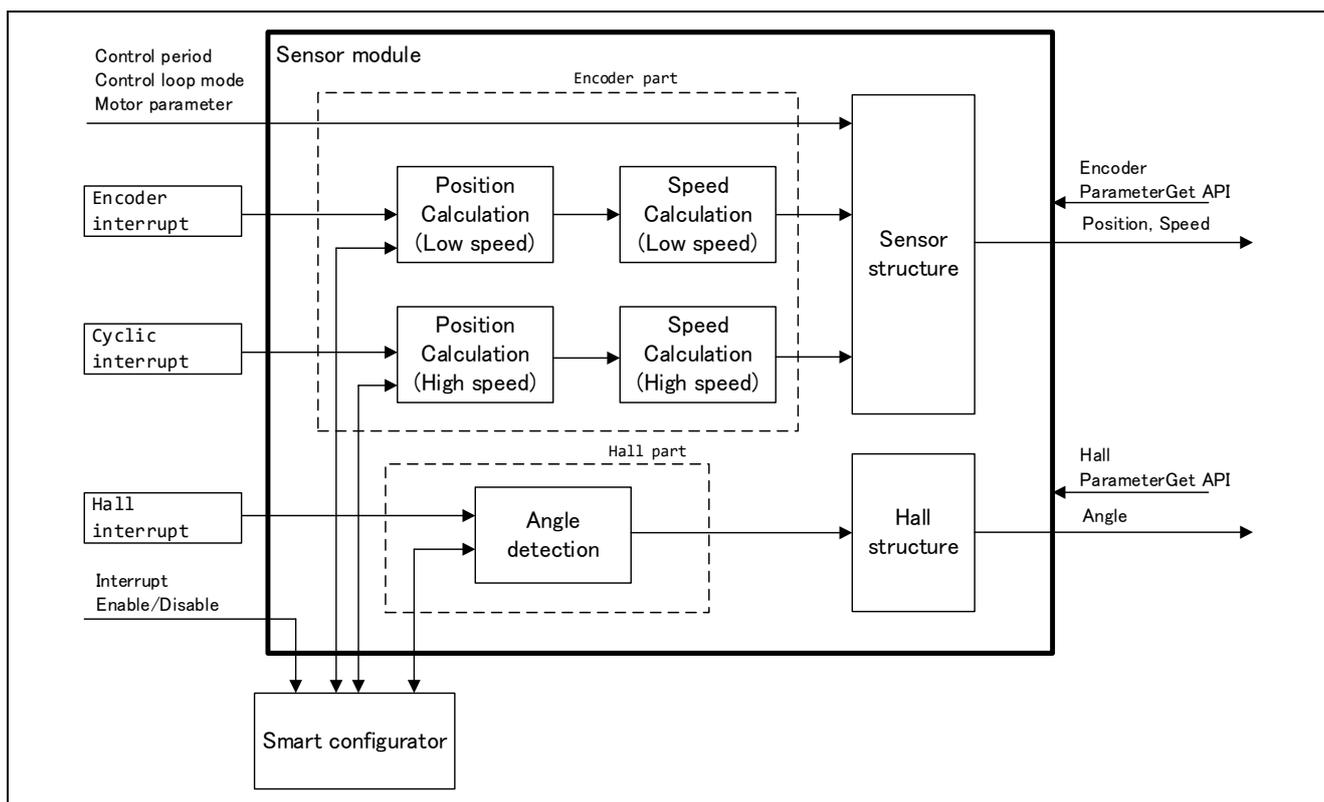


図 5-32 センサモジュール構成図

5.12.3 フローチャート

エンコーダのインプットキャプチャ割り込みから、位置・速度を演算するフローチャートを図 5-33 に示します。

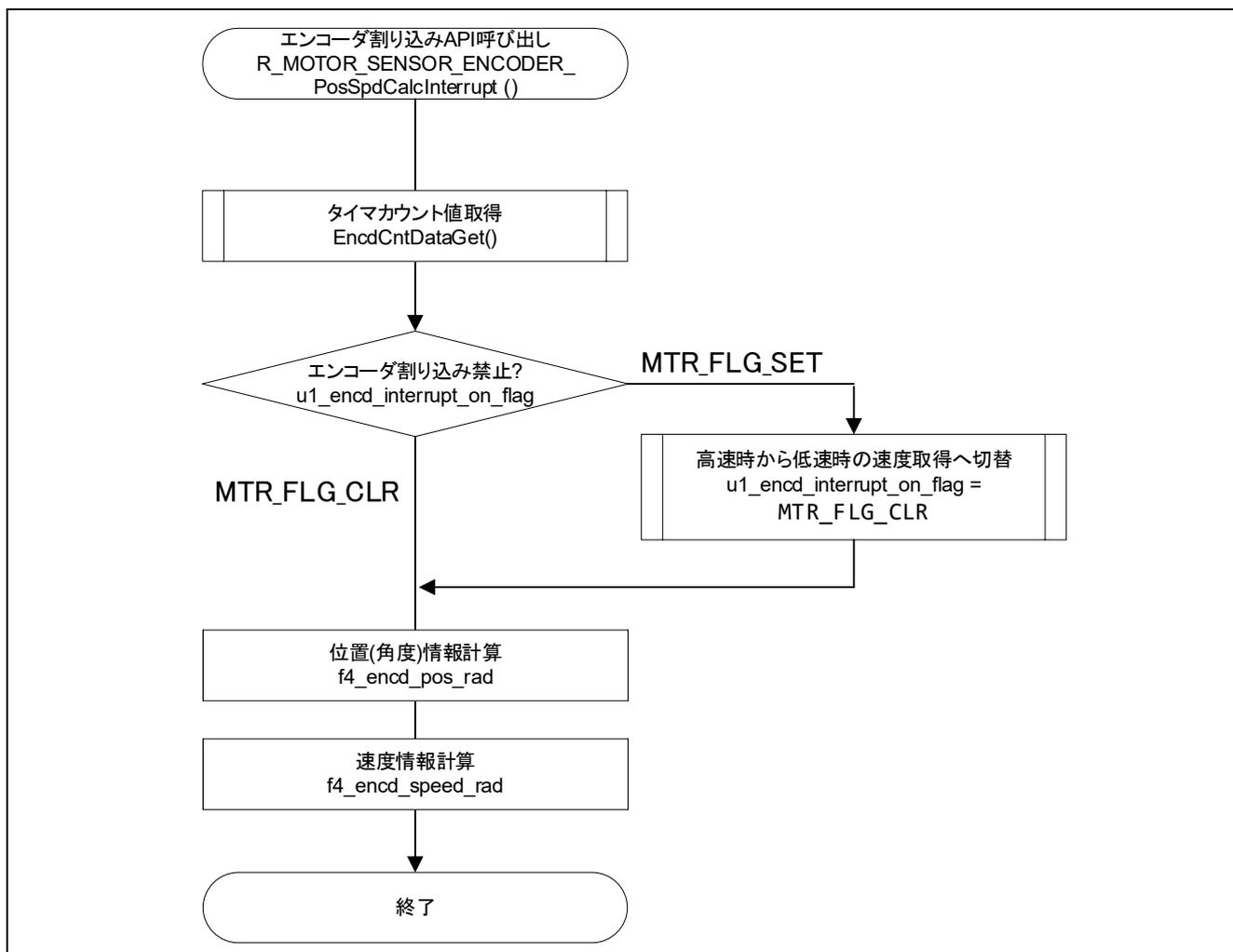


図 5-33 エンコーダ割り込み処理フローチャート

ホールセンサの割り込みを使用した始動時の割り込み処理に関するフローチャートを図 5-34 に示します。

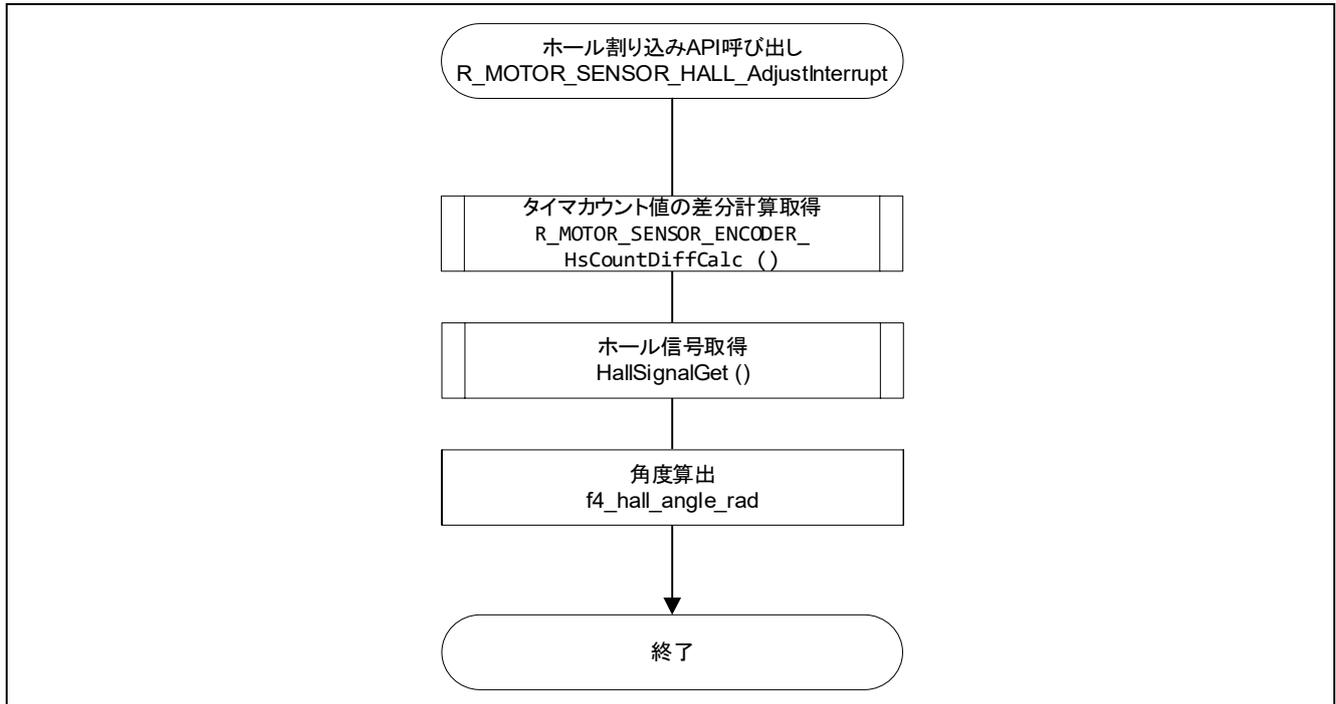


図 5-34 ホールセンサ割り込み処理フローチャート

5.12.4 API

センサモジュールの API 一覧を表 5-50 に示します。

表 5-50 API 一覧

API	説明
R_MOTOR_SENSOR_ENCODE R_Open	センサモジュールのインスタンスを生成します。本モジュールを使用する際に最初に実行してください。
R_MOTOR_SENSOR_ENCODE R_Close	センサモジュールをリセット状態にします。
R_MOTOR_SENSOR_ENCODE R_Reset	モジュールの初期化します。
R_MOTOR_SENSOR_ENCODE R_ParameterGet	エンコーダの位置・速度情報を取得します。
R_MOTOR_SENSOR_ENCODE R_ParameterUpdate	センサモジュールの制御パラメータを更新します。
R_MOTOR_SENSOR_ENCODE R_DriverParameterUpdate	センサモジュールにスマート・コンフィグレータの対応関数の設定を行います。
R_MOTOR_SENSOR_ENCODE R_PosSpdCalcInterrupt	エンコーダの信号割り込み間隔から位置・速度を演算します。
R_MOTOR_SENSOR_ENCODE R_PosSpdCalc	一定周期内に入力されたエンコーダパルス情報から位置・速度を演算します。
R_MOTOR_SENSOR_ENCODE R_HsCountDiffCalc	エンコーダのパルスをカウントするタイマの差分を取得します。
R_MOTOR_SENSOR_HALL_Par ameterGet	ホールセンサの角度情報を取得します。
R_MOTOR_SENSOR_HALL_Ang leAdjustInit	ホールセンサの初期信号を取得します。
R_MOTOR_SENSOR_HALL_Adj ustInterrupt	ホールの通電パターンを確認します。
R_MOTOR_SENSOR_HALL_Inte rruptEnable	ホール信号の割り込みを有効にします。
R_MOTOR_SENSOR_HALL_Inte rruptDisable	ホール信号の割り込みを無効にします。

5.12.5 コンフィグレーション情報

センサモジュールのコンフィグレーション情報一覧を表 5-51 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-52 に示します。

表 5-51 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	SENSOR_CFG_ENCD_TIMER_FREQ	エンコーダの信号キャプチャ用タイマの周波数を設定。
	SENSOR_CFG_ENCD_PPR	使用するエンコーダのパルス数[p/r]を設定。
	SENSOR_CFG_ENCD_RESOLUTION_MULTIPL	エンコーダ信号の通倍率を設定。
	SENSOR_CFG_ENCD_FUNC_CNT_GET	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を取得する関数を設定。
	SENSOR_CFG_ENCD_FUNC_CNT_SET	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を設定する関数を設定。
	SENSOR_CFG_ENCD_FUNC_INT_ENABLE	エンコーダのインプットキャプチャ割り込みを許可にする関数を設定
	SENSOR_CFG_ENCD_FUNC_INT_DISABLE	エンコーダのインプットキャプチャ割り込みを禁止にする関数を設定。
	SENSOR_CFG_ENCD_FUNC_SPD_TIMER_START	速度計算用フリーランタイマのカウントをスタートする関数を設定。
	SENSOR_CFG_ENCD_FUNC_SPD_TIMER_CNT_GET	速度計算用フリーランタイマのカウント値を取得する関数を設定。
	SENSOR_CFG_HALL_FUNC_SIGNAL_GET	ホールセンサの信号を取得する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_U_START	ホールセンサの U 相の割り込みを許可する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_V_START	ホールセンサの V 相の割り込みを許可する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_W_START	ホールセンサの W 相の割り込みを許可する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_U_STOP	ホールセンサの U 相の割り込みを禁止する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_V_STOP	ホールセンサの V 相の割り込みを禁止する関数を設定。
	SENSOR_CFG_HALL_FUNC_INT_W_STOP	ホールセンサの W 相の割り込みを禁止する関数を設定。

表 5-52 コンフィグレーション情報初期値一覧

マクロ名	設定値	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
SENSOR_CFG_ENCD_TIMER_FREQ	120	60.0f/CMTW0_PCLK_COUNTER_DIVISION
SENSOR_CFG_ENCD_PPR	1000	
SENSOR_CFG_ENCD_RESOLUTION_MULTIPL	4	
SENSOR_CFG_ENCD_FUNC_CNT_GET	MTU1* ⁹	GPT5* ⁹
SENSOR_CFG_ENCD_FUNC_CNT_SET	MTU1* ¹⁰	GPT5* ¹⁰
SENSOR_CFG_ENCD_FUNC_INT_ENABLE	R_Config_MTU0_InterruptEnable	
SENSOR_CFG_ENCD_FUNC_INT_DISABLE	R_Config_MTU0_InterruptDisable	
SENSOR_CFG_ENCD_FUNC_SPD_TIMER_START	GPT3* ¹	CMTW0* ¹
SENSOR_CFG_ENCD_FUNC_SPD_TIMER_CNT_GET	GPT3* ²	CMTW0* ²
SENSOR_CFG_HALL_FUNC_SIGNAL_GET	R_Config_PORT_HallSignalGet	
SENSOR_CFG_HALL_FUNC_INT_U_START	IRQ7* ³	IRQ1* ³
SENSOR_CFG_HALL_FUNC_INT_V_START	IRQ15* ⁴	IRQ2* ⁴
SENSOR_CFG_HALL_FUNC_INT_W_START	IRQ4* ⁵	IRQ0* ⁵
SENSOR_CFG_HALL_FUNC_INT_U_STOP	IRQ7* ⁶	IRQ1* ⁶
SENSOR_CFG_HALL_FUNC_INT_V_STOP	IRQ15* ⁷	IRQ2* ⁷
SENSOR_CFG_HALL_FUNC_INT_W_STOP	IRQ4* ⁸	IRQ0* ⁸

表 5-52 の注釈番号を記したペリフェラル名称は、以下関数名の xxxx に該当します。

- 【注】
1. R_Config_xxxx_SpeedCalcTimerStart
 2. R_Config_xxxx_TcntGet
 3. R_Config_ICU_xxxx_Start
 4. R_Config_ICU_xxxx_Start
 5. R_Config_ICU_xxxx_Start
 6. R_Config_ICU_xxxx_Stop
 7. R_Config_ICU_xxxx_Stop
 8. R_Config_ICU_xxxx_Stop
 9. R_Config_xxxx_TcntGet
 10. R_Config_xxxx_TcntSet

5.12.6 構造体・変数情報

センサモジュールの構造体・変数一覧を表 5-53 に示します。

表 5-53 構造体・変数一覧

構造体	変数	説明
st_sensor_t センサモジュール 用構造体	u1_ctrl_loop_mode	制御モード情報
	f4_ctrl_period	制御周期情報 [s]
	st_ec	エンコーダ用構造体。
	st_ehc	高速回転時のエンコーダ用構造体。
	st_hc	ホールセンサ用構造体。
	st_motor	モータパラメータ用構造体。
	*EncdCntDataGet	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を取得する関数ポインタ。
	*EncdCntDataSet	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を設定する関数ポインタ。
	*EncdInterruptEnable	エンコーダのインプットキャプチャ割り込みを許可にする関数ポインタ。
	*EncdInterruptDisable	エンコーダのインプットキャプチャ割り込みを禁止にする関数ポインタ。
	*EncdTimerStart	速度計算用フリーランタイムのカウントをスタートする関数ポインタ。
	*EncdTimerGet	速度計算用フリーランタイムのカウント値を取得する関数ポインタ。
	*HallSignalGet	ホールセンサの信号を取得する関数ポインタ。
	*HallUEnable	ホールセンサの U 相の割り込みを許可する関数ポインタ。
	*HallVEnable	ホールセンサの V 相の割り込みを許可する関数ポインタ。
	*HallWEnable	ホールセンサの W 相の割り込みを許可する関数ポインタ。
*HallUDisable	ホールセンサの U 相の割り込みを禁止する関数ポインタ。	
*HallVDisable	ホールセンサの V 相の割り込みを禁止する関数ポインタ。	
*HallWDisable	ホールセンサの W 相の割り込みを禁止する関数ポインタ。	

構造体	変数	説明
st_encoder_t 低速回転時のエンコーダ用構造体	u2_encd_pre_phase_cnt;	エンコーダの位相カウント値（低速、高速切り替え時などに使用）
	u4_encd_timer_cap_tcnt;	エンコーダのインプットキャプチャのカウント値
	u4_encd_timer_cap_pre_tcnt;	エンコーダのインプットキャプチャの前のカウント値
	u2_encd_timer_cnt_num;	低速時の速度計算用バッファの番号
	u2_encd_cpr_mech;	1 回転当たりのパルス数 [p/r]
	u4_encd_timer_cnt_buff;	低速時の速度計算用バッファ
	u4_encd_pulse_width;	速度検出用のエンコーダのパルス間隔
	u4_encd_pulse_width_buff;	エンコーダのパルス間隔保管用バッファ
	u4_encd_pulse_width_sum;	エンコーダのパルス間隔の合計。速度 0 の検知用。
	s4_encd_angle_cnt;	位置用のエンコーダカウンタ値。周期毎のカウンタ値差分の積分。
	f4_encd_angle_diff;	エンコーダの 1 パルス幅の角度情報
	f4_encd_cpr_mech_inv;	モータ 1 回転のパルス数の逆数。
	f4_encd_speed_pre_rad;	前回の速度 [rad/s]
	f4_encd_speed_rad;	速度（低速回転時）[rad/s]
f4_encd_pos_rad;	位置（低速回転時）[rad]	
st_encoder_highspeed_t 高速回転時のエンコーダ用構造体	u1_encd_pos_speed_calc_mode;	エンコーダの位置・速度検出方式選択
	u1_encd_interrupt_on_flag;	エンコーダの割り込みフラグ
	u1_encd_pos_speed_calc_cnt;	エンコーダの位置速度計算用カウント値
	u2_encd_hs_pre_phase_cnt;	エンコーダの 1 周期前のカウント値
	s4_encd_hs_angle_cnt;	エンコーダの位置情報カウント値
	f4_encd_hs_pos_rad;	位置（高速回転時）[rad]
	f4_encd_hs_pos_pre_rad;	1 周期前のロータ位置情報 [rad]
	f4_encd_hs_speed_rad;	速度（高速回転時）[rad/s]
	f4_encd_hs_speed_pre_rad;	1 周期前の速度 [rad/s]
	f4_encd_hs_sw_speed_rad;	高低速のモード切り替え速度 [rad/s]
f4_encd_hs_sw_speed_margin_rad;	高低速のモード切り替えマージン [rad/s]	
st_hall_t ホールセンサ用構造体	u1_hall_signal;	ホール信号の入力値
	u1_hall_pre_signal;	前回のホール信号の入力値
	u1_hall_interrupt_flg;	ホール割り込みフラグ
	f4_hall_angle_rad;	ホールの角度情報 [rad]

構造体	変数	説明
st_sensor_encoder_cfg_t エンコーダモジュール制御パラメータ設定用構造体	u1_ctrl_loop_mode;	制御モード (位置、速度)
	u2_hs_change_speed_rpm;	高低速のモード切り替え速度 [rpm]
	u2_hs_change_margin_rpm;	高低速のモード切り替えマージン [rpm]
	u2_encd_cpr;	1 回転当たりのパルス数 [p/r]
	f4_ctrl_period;	制御周期 [s]
	st_motor_parameter_t	モータパラメータの構造体
st_sensor_encoder_output_t エンコーダ出力用構造体	f4_speed_rad;	速度 [rad/s]
	f4_pos_rad;	位置 [rad]
st_sensor_hall_output_t ホールセンサ出力用構造体	f4_hall_angle_rad;	ホールの角度 [rad]
st_encoder_driver_cfg_t センサモジュールの関数ポインタ設定用構造体	*EncdCntDataGet	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を取得する関数ポインタ。
	*EncdCntDataSet	エンコーダ信号をインプットキャプチャで取得するタイマのカウント値を設定する関数ポインタ。
	*EncdInterruptEnable	エンコーダのインプットキャプチャ割り込みを許可にする関数ポインタ。
	*EncdInterruptDisable	エンコーダのインプットキャプチャ割り込みを禁止にする関数ポインタ。
	*EncdTimerStart	速度計算用フリーランタイムのカウントをスタートする関数ポインタ。
	*EncdTimerGet	速度計算用フリーランタイムのカウント値を取得する関数ポインタ。
	*HallSignalGet	ホールセンサの信号を取得する関数ポインタ。
	*HallUEnable	ホールセンサの U 相の割り込みを許可する関数ポインタ。
	*HallVEnable	ホールセンサの V 相の割り込みを許可する関数ポインタ。
	*HallWEnable	ホールセンサの W 相の割り込みを許可する関数ポインタ。
	*HallUDisable	ホールセンサの U 相の割り込みを禁止する関数ポインタ。
	*HallVDisable	ホールセンサの V 相の割り込みを禁止する関数ポインタ。
	*HallWDisable	ホールセンサの W 相の割り込みを禁止する関数ポインタ。

5.12.7 マクロ定義

センサモジュールのマクロ一覧を表 5-54 に示します。

表 5-54 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_sensor_api.h	SENSOR_ENCD_ANGL E_ADJ_TIME	512	強制励磁による始動時のロータ ひきこみ時間調整値 (位置・速度割り込み周期×調 整値=引き込み時間)
	SENSOR_ENCD_CNTA VG	4	パルス間隔で速度を算出する平 均化回数
	SENSOR_HALL_EDGE_ ERROR	1	ホールの検出エラー
	SENSOR_ENCD_NUMB _OF_TIME	2	制御周期内に処理できるエン コーダ割り込みの回数
	SENSOR_ENCD_HS_C HANGE_RPM	$(\text{SENSOR_ENCD_NUMB_OF_TIME} * 60) /$ $(\text{MOTOR_COMMON_CTRL_PERIOD} *MOTOR_COMMON_SENSOR_ENCD_CPR)$	速度検出方法をパルス間隔の測 定から一定周期内のパルス数に 変更する速度 [rpm/s]
	SENSOR_ENCD_HS_C HANGE_RAD	$(\text{SENSOR_ENCD_HS_CHANGE_RPM} *MTR_RPM2RAD)$	速度検出方法をパルス間隔の測 定から一定周期内のパルス数に 変更する速度 [rad]
	SENSOR_ENCD_HS_C HANGE_MARGIN_RPM	150	速度検出方法をパルス間隔の測 定から一定周期内のパルス数に 変更する速度マージン [rpm]
	SENSOR_ENCD_HS_C HANGE_MARGIN_RAD	$\text{SENSOR_ENCD_HS_CHANGE_MARGIN_RPM} * \text{MTR_RPM2RAD}$	速度検出方法をパルス間隔の測 定から一定周期内のパルス数に 変更する速度マージン [rad/s]
	SENSOR_ENCD_LOOP _POSITION	0	位置制御に必要な演算を実施
	SENSOR_ENCD_LOOP _SPEED	1	速度制御に必要な演算を実施

5.12.8 パラメータ調整・設定

センサモジュールのパラメータ初期値は、コンフィグレーション情報(r_motor_module_cfg.h)で設定することができます。本設定はシステム起動時に適用されます。設定する項目は 5.12.5 を参照ください。

5.12.9 位置・速度算出方式切り替え

エンコーダ信号から位置・速度を算出する方法として、信号のエッジをカウントする方法が一般的です。しかし、分解能の低いエンコーダを使用した場合、エンコーダパルス間隔が制御周期に対して大きいため、低速での速度算出が正確に行えません。そのため、低速での算出にパルス間隔をフリーランタイムで計測する方法を実装しています。エンコーダ信号による割り込みを発生させ、位置と速度を算出します。

一方で、高速回転や、高分解能エンコーダを用いた場合に、エンコーダ信号による割り込みを発生させると制御周期内での割り込み発生回数が多くなり、処理の占有率が大きくなりすぎ、制御の破綻を引き起こしてしまいます。

これを防ぐため、エンコーダ信号割り込みでの算出方法から、一定速度以上ではキャリア割り込みでの速度算出方法に切り換えを行う方法を取っています。図 5-35 のように、高速時にエンコーダ割り込みによる速度算出から、電流制御割り込みでの速度算出への切り換えを行っています。

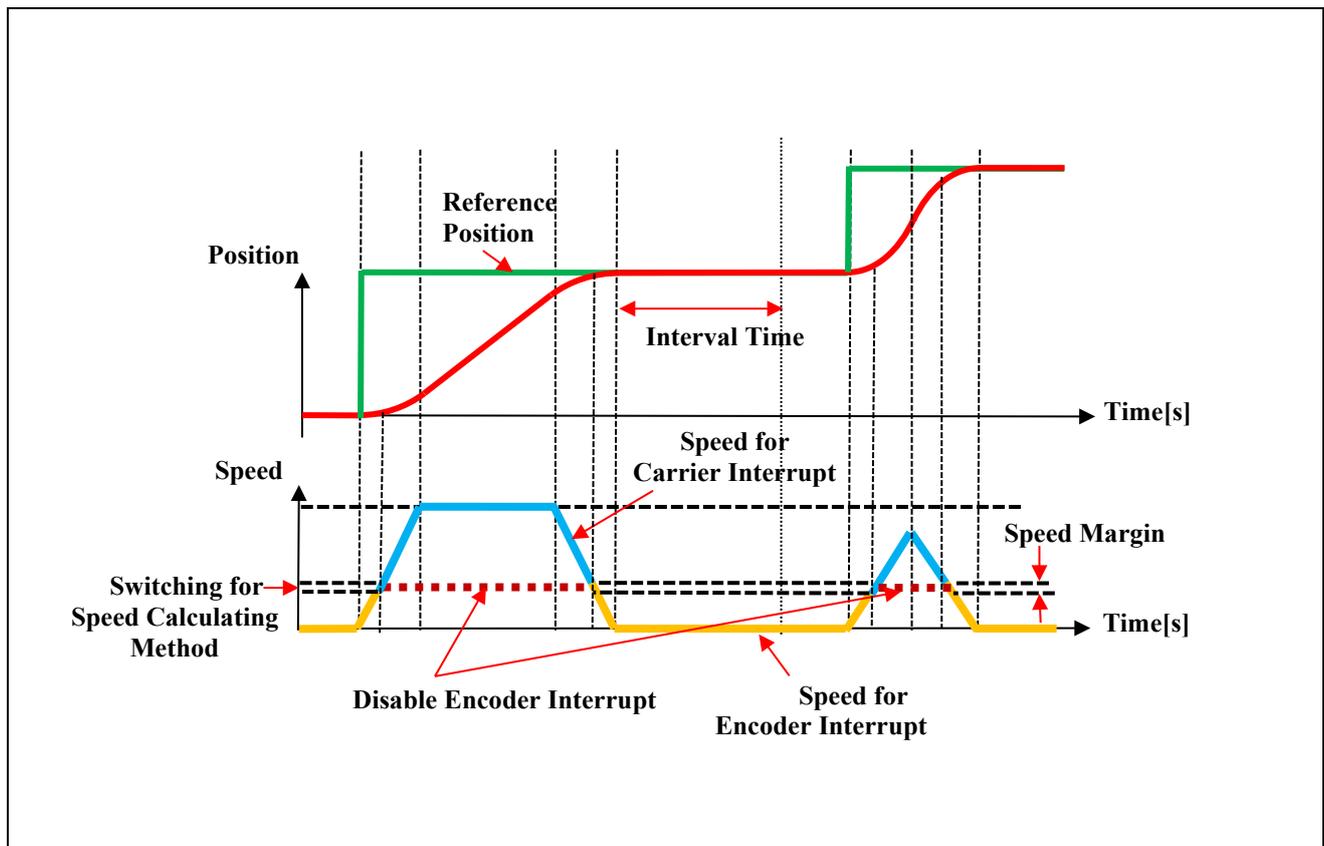


図 5-35 位置・速度の切り換え処理(一例)

5.12.10 速度センサを用いた位置・速度の算出方法

(a) 低速時のエンコーダを用いた速度算出

低速回転時にエンコーダを使用する場合の速度算出は、図 5-36 のように行います。

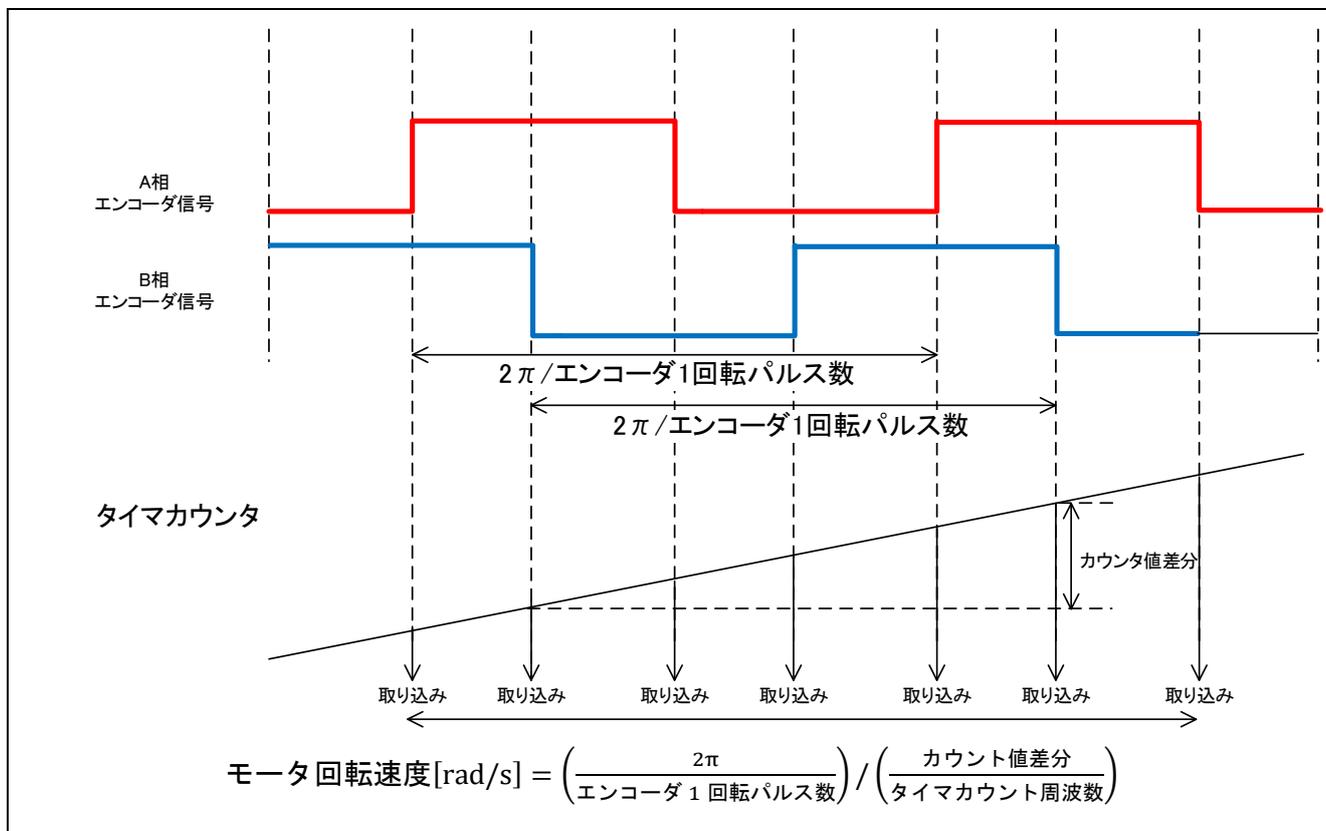


図 5-36 低速時のエンコーダによる速度計算

(b) 高速時のエンコーダを用いた速度算出

高速回転時にエンコーダを使用する場合の速度算出は、図 5-37 のように行います。

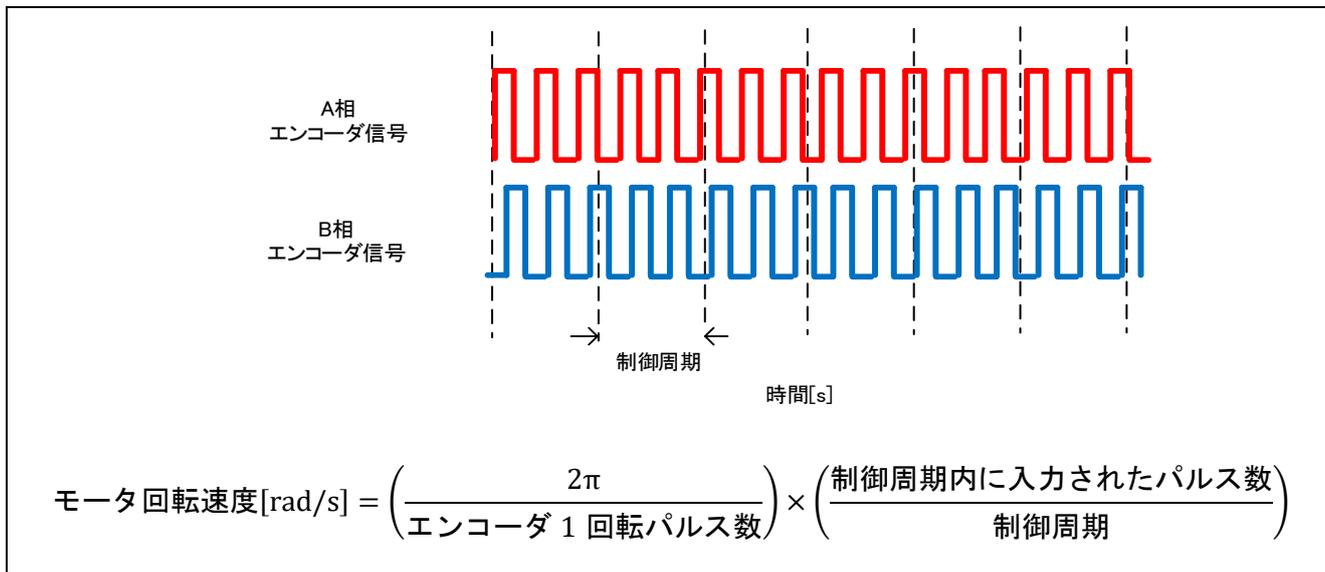


図 5-37 高速時のエンコーダによる速度計算

5.13 ドライバモジュール

ドライバモジュールは、サンプルソフトのミドルウェアに相当するマネージャモジュールと MCU のペリフェラルにアクセスするためのスマート・コンフィグレータを接続するインターフェースの役割を持つモジュールです。ドライバモジュールを適切に設定することで、MCU の機能割り当てや使用するボード仕様の差分をモータモジュールの変更無く使用することが可能になります。

5.13.1 機能

ドライバモジュールの機能一覧を表 5-55 に示します。

表 5-55 ドライバモジュールの機能一覧

機能	説明
A/D 変換値の取得	スマート・コンフィグレータ関数経由で相電流やインバータボードの母線電圧など AD 値を取得します。
PWM の duty 設定	スマート・コンフィグレータ関数経由で UVW 相へ出力する PWM Duty 値を設定します。
PWM の開始、停止	スマート・コンフィグレータ関数経由で PWM 出力の開始、停止を制御します。

5.13.2 モジュール構成図

ドライバモジュールのモジュール構成図を図 5-38 ドライバモジュール構成図に示します。

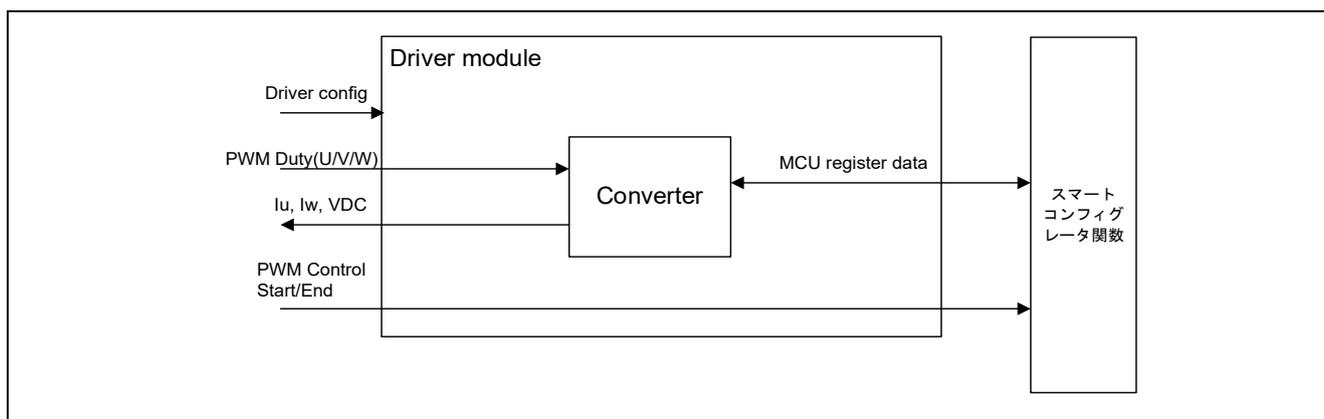


図 5-38 ドライバモジュール構成図

5.13.3 API

ドライバモジュールの API 一覧表と各 API の説明を表 5-56 に示します。

表 5-56 API 一覧

API	説明
R_MOTOR_DRIVER_Open	ドライバモジュールのインスタンスを生成します。
R_MOTOR_DRIVER_Close	モジュールをリセット状態にします。
R_MOTOR_DRIVER_ParameterUpdate	モジュール内部で使用する変数情報を入力します。
R_MOTOR_DRIVER_BldcAnalogGet	AD 変換結果を取得します。
R_MOTOR_DRIVER_BldcDutySet	PWM Duty の設定を行います。
R_MOTOR_DRIVER_PWMControlStop	PWM 制御を停止します。
R_MOTOR_DRIVER_PWMControlStart	PWM 制御を開始します。

5.13.4 コンフィグレーション情報

ドライバモジュールのコンフィグレーション情報一覧を表 5-57 に示します。使用する機能や各種パラメータを設定してください。初期値は表 5-58 に示します。

表 5-57 コンフィグレーション情報一覧

ファイル名	マクロ名	説明
r_motor_module_cfg.h	DRIVER_CFG_FUNC_PWM_OUTPUT_START	PWM 出力許可関数設定
	DRIVER_CFG_FUNC_PWM_OUTPUT_STOP	PWM 出力禁止関数設定
	DRIVER_CFG_FUNC_ADC_DATA_GET	AD 変換結果取得関数設定
	DRIVER_CFG_FUNC_DUTY_SET	Duty Cycle 設定関数設定
r_motor_inverter_cfg.h	INVERTER_CFG_ADC_REF_VOLTAGE	AD 変換基準電圧設定
r_motor_module_cfg.h	MOTOR_MCU_CFG_ADC_OFFSET	AD オフセット値設定

表 5-58 コンフィグレーション情報初期値一覧

マクロ名	設定
DRIVER_CFG_FUNC_PWM_OUTPUT_START	R_Config_MOTOR_StartTimerCtrl (スマート・コンフィグレータ関数) ^{*1}
DRIVER_CFG_FUNC_PWM_OUTPUT_STOP	R_Config_MOTOR_StopTimerCtrl (スマート・コンフィグレータ関数) ^{*1}
DRIVER_CFG_FUNC_ADC_DATA_GET	R_Config_MOTOR_AdcGetConvVal (スマート・コンフィグレータ関数) ^{*1}
DRIVER_CFG_FUNC_DUTY_SET	R_Config_MOTOR_UpdDuty (スマート・コンフィグレータ関数) ^{*1}
INVERTER_CFG_ADC_REF_VOLTAGE	5.0f
MOTOR_MCU_CFG_ADC_OFFSET	0x7FF

【注】 1. 設定値に記載した関数については、5.14 スマート・コンフィグレータ設定を参照してください。

5.13.5 構造体・変数情報

ドライバモジュールで使用する構造体一覧を表 5-59 に示します。ドライバモジュールは API のインスタンス確保にて、ドライバモジュール用構造体(g_st_driver)を定義します。

表 5-59 構造体・変数一覧

構造体	変数	説明
st_motor_driver_t ドライバモジュール用構造体	*ADCDataGet	スマート・コンフィグレータ関数へのポインタ (AD 変換結果取得関数を設定)
	*BLDCDutySet	スマート・コンフィグレータ関数へのポインタ (PWM 出力許可関数を設定)
	*PWMOutputStop	スマート・コンフィグレータ関数へのポインタ (PWM 出力禁止関数を設定)
	*PWMOutputStart	スマート・コンフィグレータ関数へのポインタ (Duty Cycle 設定関数を設定)
	f4_ad_crnt_per_digit	電流 AD 変換用スケール
	f4_ad_vdc_per_digit	電圧 AD 変換用スケール
	f4_pwm_period_cnt	PWM カウンタ周期のカウント数(Duty 設定用情報)
	f4_pwm_dead_time_cnt	デッドタイムのカウント数(Duty 設定用情報)
st_motor_driver_cfg_t ドライバモジュール制御 パラメータ設定用構造体	*ADCDataGet	スマート・コンフィグレータ関数へのポインタ
	*BLDCDutySet	スマート・コンフィグレータ関数へのポインタ
	*PWMOutputStop	スマート・コンフィグレータ関数へのポインタ
	*PWMOutputStart	スマート・コンフィグレータ関数へのポインタ
	f4_shunt_ohm	シャント抵抗値[ohm] (f4_ad_crnt_per_digit 計算用)
	f4_volt_gain	電圧変換ゲイン係数(f4_ad_vdc_per_digit 計算用)
	f4_crnt_amp_gain	電流変換ゲイン係数(f4_ad_crnt_per_digit 計算用)
	f4_pwm_period_cnt	PWM カウンタ周期のカウント数(Duty 設定用情報)
f4_pwm_dead_time_cnt	デッドタイムのカウント数(Duty 設定用情報)	

5.13.6 マクロ定義

ドライバモジュールのマクロ一覧を表 5-60 に示します。

表 5-60 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_driver.c	MOTOR_DRIVER_PRV_ADC_REF_VOLTAGE	INVERTER_CFG_ADC_REF_VOLTAGE (表 5-57 参照)	基準電圧[V]

5.13.7 パラメータ調整・設定

(a) ドライバモジュール制御パラメータの設定

ドライバモジュールでは、制御パラメータ設定(R_MOTOR_DRIVER_ParameterUpdate)から入力されたパラメータを使用して、モータモジュールとスマート・コンフィグレータとの関連付け、データ変換を行います。ドライバモジュール制御パラメータ設定用構造体(st_speed_cfg_t)を使って入力します。サンプルプログラムでは、コンフィグレーションとして定義されているものをパラメータ設定値として使用しています。設定内容を表 5-61 に示します。

表 5-61 サンプルプログラム設定例

変数名	マクロ名	ファイル名
*ADCDataGet	DRIVER_CFG_FUNC_ADC_DATA_GET	表 5-57 参照。
*BLDCDutySet	DRIVER_CFG_FUNC_DUTY_SET	
*PWMOutputStop	DRIVER_CFG_FUNC_PWM_OUTPUT_START	
*PWMOutputStart	DRIVER_CFG_FUNC_PWM_OUTPUT_STOP	
f4_shunt_ohm	INVERTER_CFG_SHUNT_RESIST	r_motor_inverter_cfg.h
f4_volt_gain	INVERTER_CFG_VOLTAGE_GAIN	
f4_crnt_amp_gain	INVERTER_CFG_CURRENT_AMP_GAIN	
f4_pwm_period_cnt	MOTOR_COMMON_CARRIER_SET_BASE	r_motor_module_cfg.h
f4_pwm_dead_time_cnt	MOTOR_COMMON_DEADTIME_SET	

5.14 スマート・コンフィグレータ設定

サンプルプログラムでは、スマート・コンフィグレータを使用してプロジェクトを作成しています。使用しているコンポーネントとユーザ領域に追加した関数を説明します。

5.14.1 クロック設定

クロック設定を表 5-62 に示します。

表 5-62 MCU クロック設定

クロック	周波数	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
メインクロック	10MHz	
システムクロック (ICLK)	120MHz	
周辺モジュールクロック (PCLKA)	120MHz	
周辺モジュールクロック (PCLKB/PCLKC/PCLKD)	60MHz/120MHz/60MHz	
FlashIF クロック (FCLK)	60MHz	
IWDTCLK	120kHz	

5.14.2 コンポーネント設定

使用するコンポーネント情報と機能割り当てを表 5-63 に示します。

表 5-63 スマート・コンフィグレータのコンポーネントと機能割り当て

機能	コンポーネント	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
ホール割り込み処理	Config_ICU	
3相 PWM 出力、A/D 変換処理(電流検出、インバータ母線電圧検出)	Config_MOTOR	
A/D 変換処理(ボード UI 用指令電圧検出)	Config_S12AD2	
使用ポートの設定	Config_PORT	
位置速度制御割り込みタイマ	Config_CMT0	
速度計測用フリーランタイマ	Config_GPT3	Config_CMTW0
独立ウォッチドックタイマ	Config_IWDT	
エンコーダの位相計数	Config_MTU0	
イベントリンク(位相計数用)	---	Config_ELC
エンコーダの位相計数	Config_MTU1	Config_GPT5
	Config_POE	

5.14.3 割り込み

モータコンポーネントを使用した MCU の割り込み情報を表 5-64、表 5-65 に示します。

表 5-64 割り込み一覧 (RX26T RAM64KB バージョン)

コンポーネント	割り込み関数	説明
Config_ICU	r_Config_ICU_irqxx_interrupt* ¹ r_Config_ICU_irqxx_interrupt* ¹ r_Config_ICU_irqxx_interrupt* ¹	ホールセンサの割り込み 割り込みレベル：14 多重割り込み：禁止
Config_MOTOR	r_Config_MOTOR_ad_interrupt	AD 変換終了割り込み 割り込みレベル：12 多重割り込み：許可
Config_S12AD2	なし	なし
Config_PORT	なし	なし
Config_CMT0	r_Config_CMT0_cmi0_interrupt	位置速度制御割り込み 割り込みレベル：11 多重割り込み：許可
Config_GPT3	なし	なし
Config_IWDT	なし	なし
Config_MTU0	r_Config_MTU0_tgib0_interrupt	エンコーダの割り込み 割り込みレベル：13 多重割り込み：禁止
Config_MTU1	なし	なし
Config_POE	r_Config_POE_oei1_interrupt	HW 過電流の割り込み/出力短絡 の割り込み 割り込みレベル：15 多重割り込み：禁止

【注】 1. “xx”は各 MCU に割り当てられた設定です。詳細は表 5-52 を参照ください。

表 5-65 割り込み一覧 (RX26T RAM48KB バージョン)

コンポーネント	割り込み関数	説明
Config_ICU	r_Config_ICU_irqxx_interrupt*1 r_Config_ICU_irqxx_interrupt*1 r_Config_ICU_irqxx_interrupt*1	ホールセンサの割り込み 割り込みレベル：14 多重割り込み：禁止
Config_MOTOR	r_Config_MOTOR_ad_interrupt	AD 変換終了割り込み 割り込みレベル：12 多重割り込み：許可
Config_S12AD2	なし	なし
Config_PORT	なし	なし
Config_CMT0	r_Config_CMT0_cmi0_interrupt	位置速度制御割り込み 割り込みレベル：11 多重割り込み：許可
Config_CMTW0	なし	なし
Config_IWDT	なし	なし
Config_MTU0	r_Config_MTU0_tgib0_interrupt	エンコーダの割り込み 割り込みレベル：13 多重割り込み：禁止
Config_GPT5	なし	なし
Config_ELC	なし	なし
Config_POE	r_Config_POE_oei1_interrupt	HW 過電流の割り込み 割り込みレベル：15 多重割り込み：禁止
	r_Config_POE_oei2_interrupt	出力短絡の割り込み 割り込みレベル：15 多重割り込み：禁止

【注】 1. “xx”は各 MCU に割り当てられた設定です。詳細は表 5-52 を参照ください。

5.14.4 ユーザコード詳細

ユーザコード領域に作成した関数一覧を表 5-66、表 5-67 に示します。

表 5-66 ユーザ領域の関数一覧(RX26T RAM64KB バージョン)

コンポーネント	関数	説明
Config_PORT	R_Config_PORT_GetSW1	SW1 状態の取得
	R_Config_PORT_GetSW2	SW2 状態の取得
	R_Config_PORT_Led1_on	LED1 点灯
	R_Config_PORT_Led2_on	LED2 点灯
	R_Config_PORT_Led1_off	LED1 消灯
	R_Config_PORT_Led2_off	LED2 消灯
	R_Config_PORT_HallSignalGet	ホール信号検出
Config_GPT3	R_Config_GPT3_TcntGet	タイマのカウンタ値取得
	R_Config_GPT3_SpeedCalcTimerStart	速度計測用フリーランタイムスタート
Config_MTU0	R_Config_MTU0_InterruptEnable	エンコーダ割り込み許可
	R_Config_MTU0_InterruptDisable	エンコーダ割り込み禁止
Config_MTU1	R_Config_MTU1_TcntSet	MTU1 タイマカウンタ設定
	R_Config_MTU1_TcntGet	MTU1 タイマカウンタ読み出し

表 5-67 ユーザ領域の関数一覧(RX26T RAM48KB バージョン)

コンポーネント	関数	説明
Config_PORT	R_Config_PORT_GetSW1	SW1 状態の取得
	R_Config_PORT_GetSW2	SW2 状態の取得
	R_Config_PORT_Led1_on	LED1 点灯
	R_Config_PORT_Led2_on	LED2 点灯
	R_Config_PORT_Led1_off	LED1 消灯
	R_Config_PORT_Led2_off	LED2 消灯
	R_Config_PORT_HallSignalGet	ホール信号検出
Config_CMTW0	R_Config_CMTW0_TcntGet	タイマのカウンタ値取得
	R_Config_CMTW0_SpeedCalcTimerStart	速度計測用フリーランタイムスタート
Config_MTU0	R_Config_MTU0_InterruptEnable	エンコーダ割り込み許可
	R_Config_MTU0_InterruptDisable	エンコーダ割り込み禁止
Config_GPT5	R_Config_GPT5_TcntSet	GPT5 タイマカウンタ設定
	R_Config_GPT5_TcntGet	GPT5 タイマカウンタ読み出し

5.14.5 端子設定

端子のインタフェース情報を表 5-68、表 5-69 に示します。

表 5-68 端子インタフェース(RX26T RAM64KB バージョン)

機 能	端子名
インバータ母線電圧測定	P43 / AN003
位置/速度指令値入力用 (アナログ値)	P50 / AN204
START/STOP トグルスイッチ	P23
ERROR RESET プッシュスイッチ	P22
LED1 制御	P21
LED2 制御	P20
U 相電流測定	P40 / AN000
W 相電流測定	P42 / AN002
PWM 出力 (U _p)	P73 / MTIOC4B
PWM 出力 (V _p)	P72 / MTIOC4A
PWM 出力 (W _p)	P71 / MTIOC3B
PWM 出力 (U _n)	P76 / MTIOC4D
PWM 出力 (V _n)	P75 / MTIOC4C
PWM 出力 (W _n)	P74 / MTIOC3D
ホール U 相入力	P30 / IRQ7
ホール V 相入力	P27 / IRQ15
ホール W 相入力	P24 / IRQ4
エンコーダ A 相入力	P33 / MTCLKA
エンコーダ B 相入力	P32 / MTCLKB
過電流検出時の PWM 緊急停止入力	P70 / POE0#

表 5-69 端子インタフェース(RX26T RAM48KB バージョン)

機能	端子名
インバータ母線電圧測定	P43 / AN003
位置/速度指令値入力用 (アナログ値)	P47 / AN206
START/STOP トグルスイッチ	P21
ERROR RESET プッシュスイッチ	P20
LED1 制御	P65
LED2 制御	PB5
U 相電流測定	P40 / AN000
W 相電流測定	P42 / AN002
PWM 出力 (U _p)	P71 / MTIOC3B
PWM 出力 (V _p)	P72 / MTIOC4A
PWM 出力 (W _p)	P73 / MTIOC4B
PWM 出力 (U _n)	P74 / MTIOC3D
PWM 出力 (V _n)	P75 / MTIOC4C
PWM 出力 (W _n)	P76 / MTIOC4D
ホール U 相入力	P11 / IRQ1
ホール V 相入力	P00 / IRQ2
ホール W 相入力	PE2 / IRQ0
エンコーダ A 相入力	P94 / MTCLKA
エンコーダ B 相入力	P91 / MTCLKB
過電流検出時の PWM 緊急停止入力	P96 / POE4#

5.14.6 マクロ定義

モータコンポーネントが出力するモータモジュール向けマクロ一覧を表 5-70 に示します。

表 5-70 マクロ一覧(RX26T RAM64KB バージョン/ RX26T RAM48KB バージョン)

ファイル名	マクロ名	定義値	備考
Config_MOTOR.h	CG_CONFIG_MOTOR_PWM_TIMER_FREQ	120.0f	PWM タイマカウンタクロック [MHz]
	CG_CONFIG_MOTOR_CARRIER_FREQ	20.000f	タイマ動作周波数[kHz]
	CG_CONFIG_MOTOR_DEADTIME	2.000f	デッドタイム[us]
	CG_CONFIG_MOTOR_INTR_DECIMATION	0.0	割り込み間引き回数
	CG_MOTOR_CFG_MAX_AD_DATA	4095.0f	A/D データ最大値
	CG_MOTOR_MCU_CFG_AD_FREQ	60.0f	A/D 変換クロック [MHz]

6. ベクトル制御アルゴリズム

6.1 永久磁石同期モータの解析モデル

図 6-1 のような正弦波状の磁束分布を持った永久磁石同期モータの電圧方程式は、下記のように表すことができます。

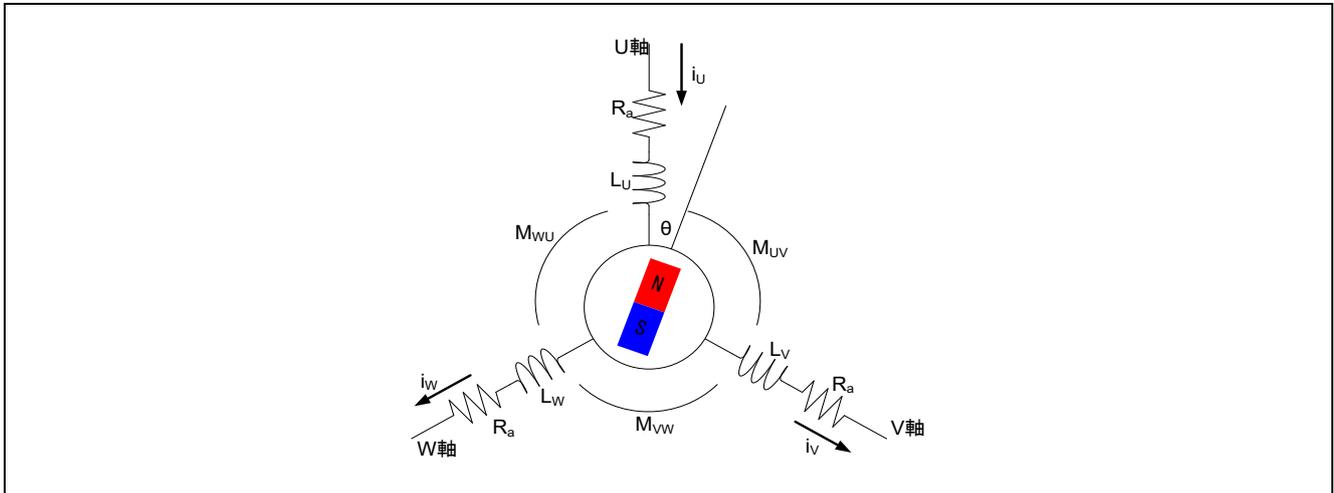


図 6-1 3相永久磁石同期モータの概念図

$$\begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix} = R_a \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + p \begin{bmatrix} \phi_u \\ \phi_v \\ \phi_w \end{bmatrix}$$

$$\begin{bmatrix} \phi_u \\ \phi_v \\ \phi_w \end{bmatrix} = \begin{bmatrix} L_u & M_{uv} & M_{wu} \\ M_{uv} & L_v & M_{vw} \\ M_{wu} & M_{vw} & L_w \end{bmatrix} \begin{bmatrix} i_u \\ i_v \\ i_w \end{bmatrix} + \psi \begin{bmatrix} \cos \theta \\ \cos (\theta - 2\pi/3) \\ \cos (\theta + 2\pi/3) \end{bmatrix}$$

v_u, v_v, v_w : 各相電機子電圧

L_u, L_v, L_w : 各相自己インダクタンス

i_u, i_v, i_w : 各相電機子電流

M_{uv}, M_{vw}, M_{wu} : 各相間相互インダクタンス

ϕ_u, ϕ_v, ϕ_w : 各相電機子鎖交磁束

ψ : 永久磁石による電機子鎖交磁束の最大値

R_a : 各相電機子抵抗

θ : U相からの永久磁石 (回転子) の進み角

p : 微分演算子

6.2 永久磁石同期モータの d q 軸モデル

ベクトル制御では、交流の3相(u, v, w)座標系を直流の2相(d, q)座標系で表します。固定子の3相巻線は、永久磁石のロータに同期して回転する2相巻線に変換されるため、相対的に静止し、電氣的に独立した2つの直流回路として扱うことができます。

2相(d, q)座標系は、回転子の永久磁石の磁束(N極)方向にd軸を定め、d軸から角度 θ の正方向に90度進んだ方向をq軸とします。dq座標系から見た永久磁石同期モータの電圧方程式を得るために以下の変換行列を用います。

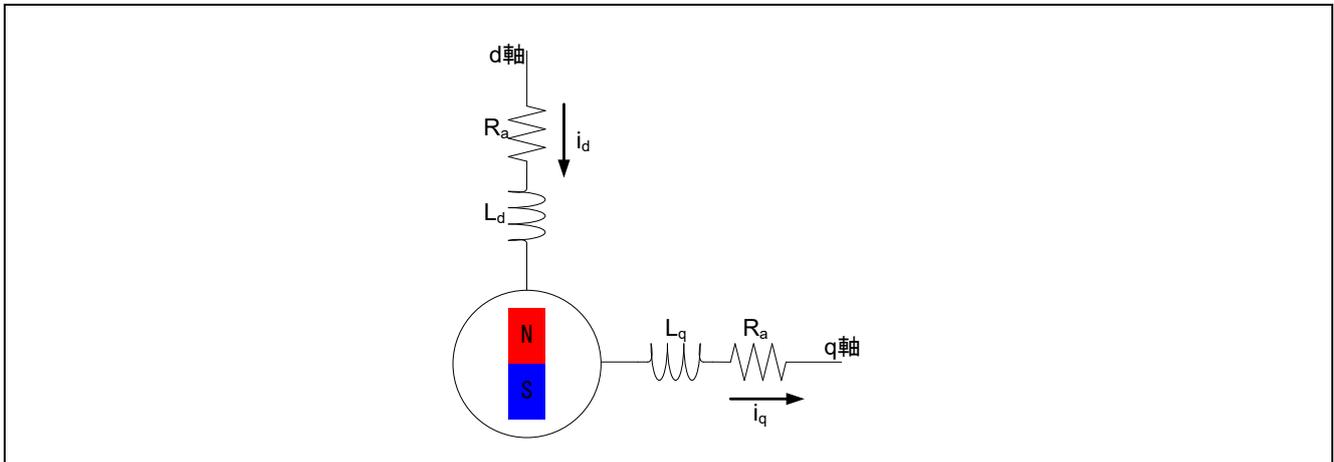


図 6-2 2相直流モータの概念図

$$C = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos\theta & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \\ -\sin\theta & -\sin(\theta - 2\pi/3) & -\sin(\theta + 2\pi/3) \end{bmatrix}$$

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = C \begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix}$$

上記の座標変換により dq 座標系での電圧方程式は以下のように表すことができます。

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} R_a + pL_d & -\omega L_q \\ \omega L_d & R_a + pL_q \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} 0 \\ \omega\psi_a \end{bmatrix}$$

v_d, v_q : dq軸電機子電圧

L_d, L_q : dq軸自己インダクタンス

i_d, i_q : dq軸電機子電流

$$L_d = l_a + \frac{3(L_a - L_{as})}{2}, L_q = l_a + \frac{3(L_a + L_{as})}{2}$$

R_a : 各相電機子抵抗

ψ_a : 永久磁石による電機子鎖交磁束の実効値

ω : 角速度

$$\psi_a = \sqrt{\frac{3}{2}}\psi$$

以上より、静止している 3 相固定子に流れていた交流は、回転子である永久磁石と同期して回転している 2 相の固定子に流れる直流に見なすことができます。

モータに生じるトルクの大きさは電流ベクトルと電機子鎖交磁束の外積より下記のように求めます。この式の右辺第一項をマグネットトルク、右辺第二項をリラクタンストルクと呼びます。

$$T = P_n \{ \psi_a i_q + (L_d - L_q) i_d i_q \}$$

T : モータトルク P_n : 極対数

d 軸と q 軸のインダクタンスの差が無いモータを突極性が無いモータと呼びます。この場合、リラクタンストルクは 0 になるので、q 軸電流に比例してトルクは大きくなります。このため、q 軸電流をトルク電流と呼ぶことがあります。一方、d 軸電流は、その大きさを変化させることであたかも永久磁石の磁束の大きさが変化しているかのように見なせる働きをするので励磁電流と呼ぶことがあります。

6.3 ベクトル制御システムとコントローラ

位置制御システム全体のブロック図を以下に示します。

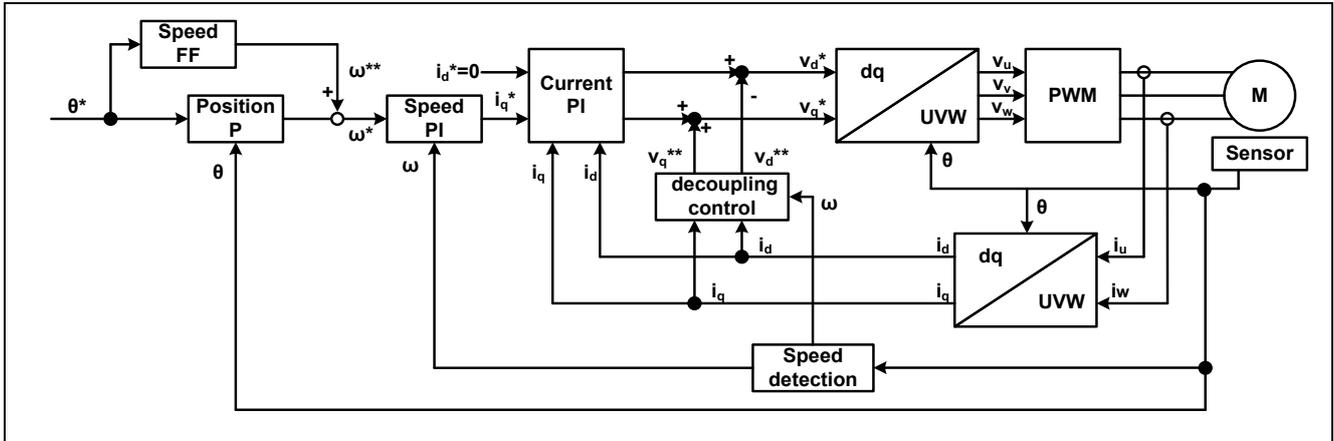


図 6-3 ベクトル制御のシステムブロック（位置制御）

図 6-3 のように位置制御システムは位置制御系、速度制御系と電流制御系によって構成されます。速度制御系と電流制御系は一般的な PI 制御コントローラを、位置制御系は P 制御と速度へのフィードフォワード制御を使用して実現され、それぞれのコントローラのゲインは所望の制御特性を実現するために適切に設計する必要があります。

また、システムブロック図中の非干渉制御（decoupling control）ブロックでは、モータが回転することによって発生する誘起電圧 v_d^{**} , v_q^{**} （下記式参照）を、各相の指令電圧にフィードフォワードします。それにより速度制御システムの高い応答性を実現するとともに、d 軸と q 軸を独立に制御することを可能とします。

$$v_d^{**} = -\omega L_d i_q$$

$$v_q^{**} = \omega(L_d i_d + \psi_a)$$

6.3.1 電流制御系の設計

モータの電気的な特性から電流制御系をモデル化します。固定子コイルは抵抗 R とインダクタンス L で表すことができるため、モータの固定子モデルは、一般的な RL 直列回路の伝達関数 $\frac{1}{R+Ls}$ で表せます。

コントローラは PI 制御を使用して、電流制御系は、図 6-4 のようなフィードバック制御系で表すことができます。

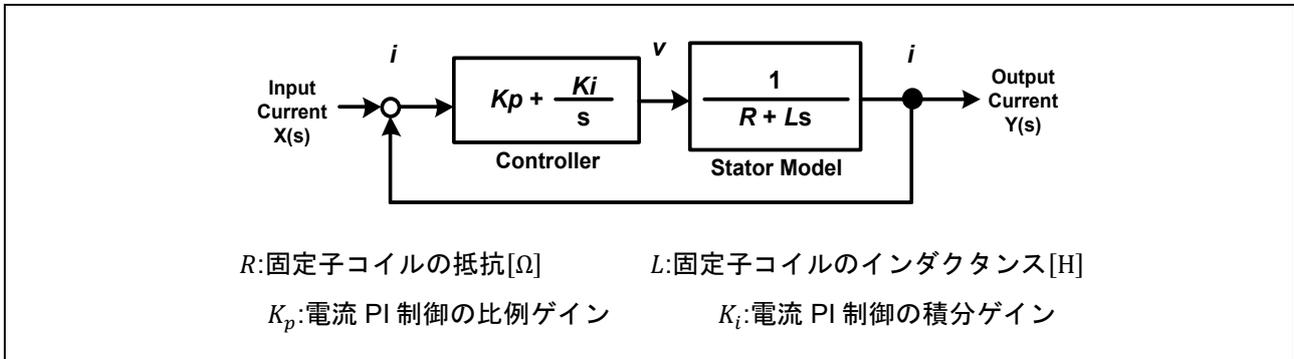


図 6-4 電流制御系のモデル

まず、モータ固定子の R と L を既知として電流制御系の PI 制御ゲインを設計します。
電流制御系の閉ループ伝達関数は以下のように求められます。

$$G(s) = \frac{Y(s)}{X(s)} = \frac{\frac{K_a}{K_b} \left(1 + \frac{s}{a}\right)}{s^2 + \frac{1}{K_b} \left(1 + \frac{K_a}{a}\right)s + \frac{K_a}{K_b}}$$

$$K_i = K_p a, \quad K_a = \frac{K_p a}{R}, \quad K_b = \frac{L}{R}$$

また、零点を持つ 2 次遅れ系の一般式は下記の通り書き表すことができます。

$$\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \left(1 + \frac{s}{\omega_z}\right)$$

そして電流制御系の伝達関数を、零点を持つ2次遅れ系の一般式と係数比較すると、次式のような関係が得られます。

$$\frac{\omega_n^2 \left(1 + \frac{s}{\omega_z}\right)}{s^2 + 2\zeta\omega_n s + \omega_n^2} \Leftrightarrow \frac{\frac{K_a}{K_b} \left(1 + \frac{s}{a}\right)}{s^2 + \frac{1}{K_b} \left(1 + \frac{K_a}{a}\right)s + \frac{K_a}{K_b}}$$

$$\omega_n^2 = \frac{K_a}{K_b}, \quad 2\zeta\omega_n = \frac{1}{K_b} \left(1 + \frac{K_a}{a}\right), \quad \omega_z = a$$

以上より、固有周波数 ω_n 、減衰係数 ζ 、零点周波数 ω_z は、下記の通り書き表すことができます。

$$\omega_n = \sqrt{\frac{K_a}{K_b}}, \quad \zeta = \frac{1}{2K_b\sqrt{\frac{K_a}{K_b}}} \left(1 + \frac{K_a}{a}\right), \quad \omega_z = a = \frac{\omega_n^2 L}{2\zeta\omega_n L - R}$$

このことから電流 PI 制御ゲイン $K_{p_current}$ 、 $K_{i_current}$ は、次式のようにになります。

$$K_{p_current} = 2\zeta_{CG}\omega_{CG}L - R, \quad K_{i_current} = K_{p_current}a = \omega_{CG}^2 L$$

ω_{CG} :電流制御系固有周波数

ζ_{CG} :電流制御系減衰係数

よって、電流制御系の PI 制御ゲインは、 ω_{CG} と ζ_{CG} により設計可能であることが分かります。

6.3.2 速度制御系の設計

モータの機械的な特性から速度制御系をモデル化します。回転系の運動方程式より、機械系のトルク式は次式のように書き表せます。

$$T = J\dot{\omega}_{mech}$$

J :回転子イナーシャ, ω_{mech} :機械角速度

これに対して、電気系のトルク式は、マグネットトルクのみを考慮すれば、次式のようになります。

$$T = P_n \psi_a i_q$$

力学系と電気系の2つのトルク式を使えば、機械角速度は次式のように書き表せます。

$$\omega_{mech} = \frac{P_n \psi_a}{sJ} i_q$$

ω_{mech} :機械角速度

よって、これが速度制御系におけるモータモデルとなります。また、コントローラはPI制御を使用し、速度制御系は、図6-5のようなフィードバック制御系で表すことができます。

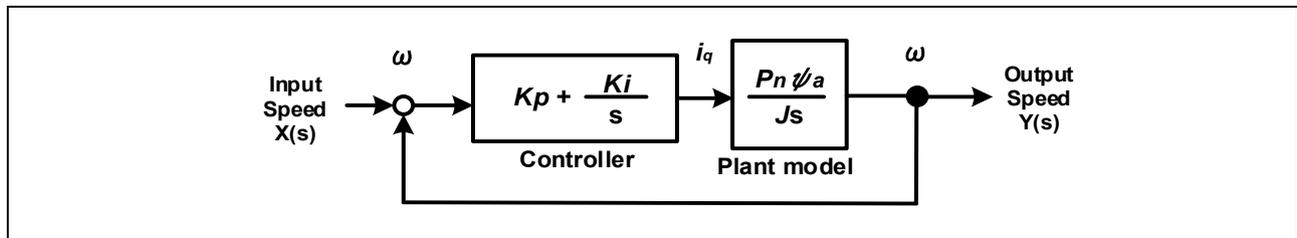


図 6-5 速度制御系のモデル

ここで、モータパラメータ P_n, ψ, J を既知として、速度制御系のPI制御ゲインを設定します。まずは、システムの伝達関数を求めます。

速度制御系の閉ループ伝達関数は以下のように求められます。

$$G(s) = \frac{Y(s)}{X(s)} = \frac{K_b a \left(1 + \frac{s}{a}\right)}{s^2 + K_b s + K_b a}$$

$$K_b = \frac{K_p P_n \psi}{J}, \quad K_i = K_p a$$

また、零点を持つ2次遅れ系の一般式は下記の通り書き表すことができます。

$$\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \left(1 + \frac{s}{\omega_z}\right)$$

電流制御系と同様に、速度制御系の伝達関数を、零点を持つ2次遅れ系の一般式と係数比較すると、次式のような関係式が得られます。

$$\frac{\omega_n^2 (1 + s/\omega_z)}{s^2 + 2\zeta\omega_n s + \omega_n^2} \Leftrightarrow \frac{aK_b \left(1 + \frac{s}{a}\right)}{s^2 + K_b s + aK_b}$$

$$\omega_n^2 = aK_b = \frac{K_p a P_n \psi a}{J}, \quad 2\zeta\omega_n = K_b = \frac{K_p P_n \psi a}{J}, \quad \omega_z = a$$

以上より、固有周波数 ω_n 、減衰係数 ζ 、零点周波数 ω_z は、下記の通り書き表すことができます。

$$\omega_n = \sqrt{\frac{K_p a P_n \psi a}{J}}, \quad \zeta = \frac{1}{2} \sqrt{\frac{K_p P_n \psi a}{aJ}}, \quad \omega_z = a = \frac{\omega_n}{2\zeta}$$

このことからPI制御ゲイン K_{p_speed}, K_{i_speed} は、次式のようになります。

$$K_{p_speed} = \frac{2\zeta_{SG}\omega_{SG}J}{P_n\psi a}, \quad K_{i_speed} = K_{p_speed} * a = \frac{\omega_{SG}^2 J}{P_n\psi a}$$

ω_{SG} :速度制御系固有周波数

ζ_{SG} :速度制御系減衰係数

よって、速度制御系のPI制御ゲインは、 ω_{SG} と ζ_{SG} により設計可能であることが分かります。

6.3.3 位置制御系の設計

位置制御系のコントローラは比例項のみを用いています。速度の指令値に比べて過大な入力に対して速く応答させるため、速度へのフィードフォワードを組み合わせて応答性を向上させています。位置制御系のブロックは以下ようになります。

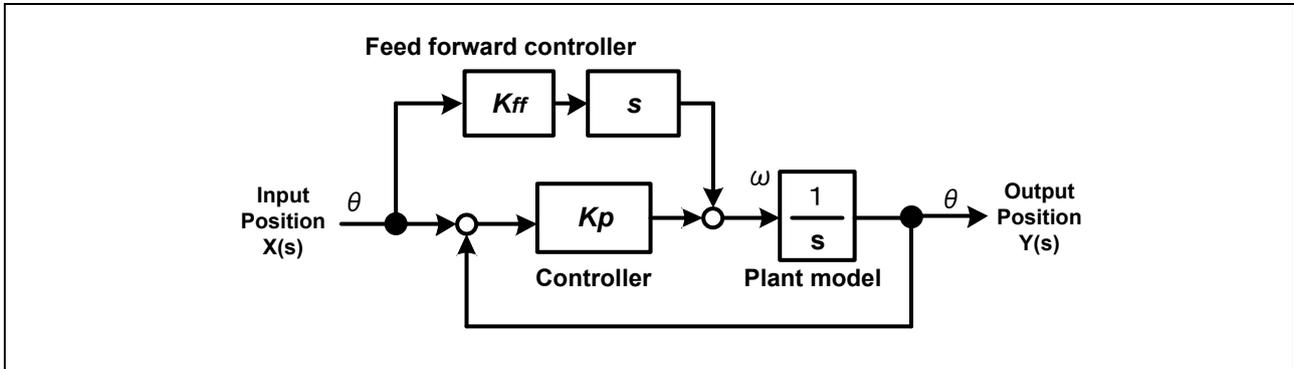


図 6-6 位置制御系のモデル

位置制御系は P 制御のみを用いており、 $K_{p_position}$ のゲイン設計には、位置制御系固有周波数 ω_{PG} のみで設計します。

$$\omega = K_{p_position}(\theta_{ref} - \theta)$$

$$K_{p_position} = \omega_{PG}$$

また速度応答性を向上させるため、速度へのフィードフォワード制御を実装しています。

$$\omega_{ff} = K_{speed_ff} \dot{\theta}_{ref}$$

よって、速度フィードフォワードは固定値、位置の P ゲインは固有周波数 ω_n により設計できます。

7. 試験結果

本章で示す試験結果は、2.1 動作確認環境において測定を行った参考値です。

7.1 プログラムサイズ

サンプルプログラムのプログラムサイズを表 7-1 に示します。コンパイラの最適化設定において、最適化レベル 2 (-optimize = 2) に設定し、最適化方法をコード・サイズ重視の最適化(-size)に設定しています。

表 7-1 プログラムサイズ

メモリ	サイズ	
	RX26T RAM64KB バージョン	RX26T RAM48KB バージョン
ROM	26.8 KB	26.9 KB
RAM	9.9 KB	9.9 KB
スタック解析結果の最大値	380 B	380 B
スタックサイズの設定値	5120 B	5120 B

7.2 CPU 負荷率

各制御周期の CPU 処理時間と負荷率を以下に示します。

表 7-2 制御ループと CPU 負荷率

CPU Board	制御ループ種類	制御周期	処理時間	CPU 負荷率
RX26T RAM64KB バージョン	電流制御ループ	50 us (間引き 0 回)	13.5 us	27.0 %
	速度・位置制御ループ	500 us	3.5 us	0.7 %
RX26T RAM48KB バージョン	電流制御ループ	50 us (間引き 0 回)	13.6 us	27.2 %
	速度・位置制御ループ	500 us	3.5 us	0.7 %

7.3 動作波形

試験結果として、サンプルプログラム(RX26T RAM64KB バージョン)で制御した時の波形を示します。

表 7-3 測定条件

項目	値	備考
電流制御系周波数	300 [Hz]	
電流制御系減衰係数	1	
速度制御系周波数	12 [Hz]	
速度制御系減衰係数	1	
位置制御系周波数	4 [Hz]	位置制御時のみ有効。
負荷	—	無負荷で実施

速度制御を行った結果を図 7-1 に示します。

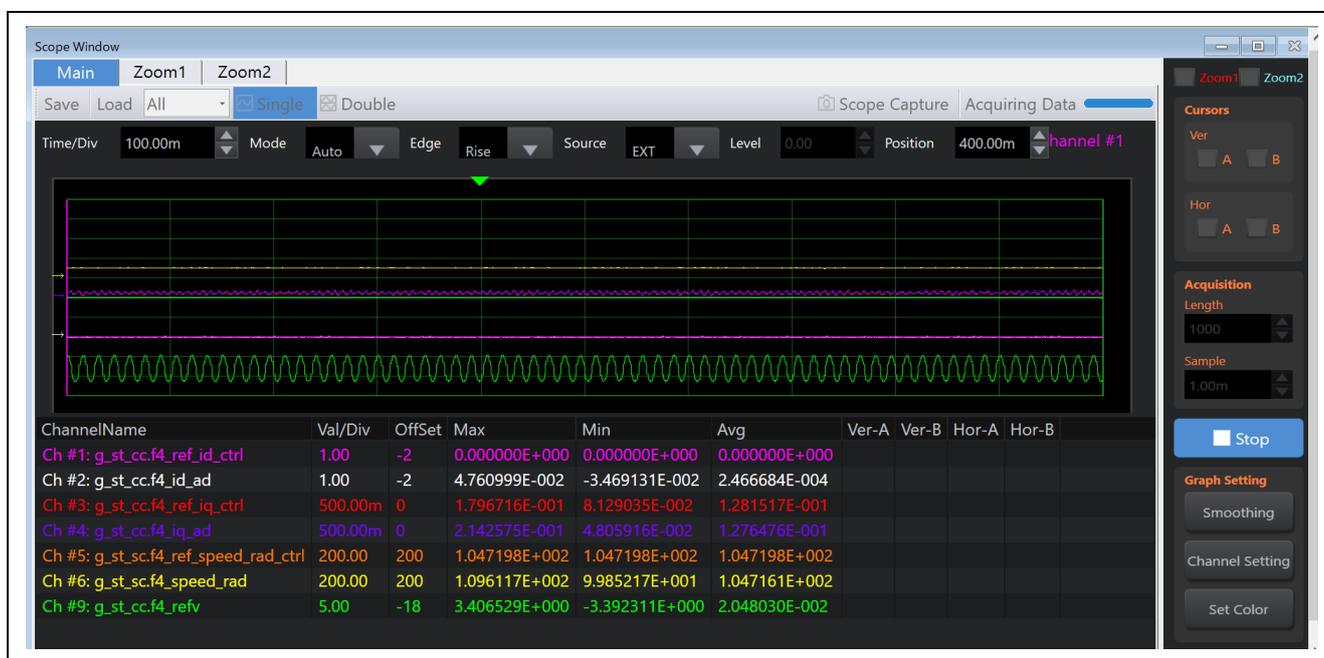


図 7-1 エンコーダを使用した速度制御

駆動条件：

回転速度：指令速度 1000 [rpm]

波形情報：

黄色：検出速度 [rad/s], (200rad/s / div.)、オレンジ：指令速度 [rad/s], (200rad/s / div.)

赤：q 軸電流指令値 [A], (500mA / div.)、紫：q 軸電流値 [A], (500mA / div.)、

ピンク：d 軸電流指令値 [A], (1A / div.)、白：d 軸電流値 [A], (1A / div.)

黄緑：U 相電圧 [V], (5V / div.)

横軸：100ms / div.

位置制御を行った結果を図 7-2 に示します。

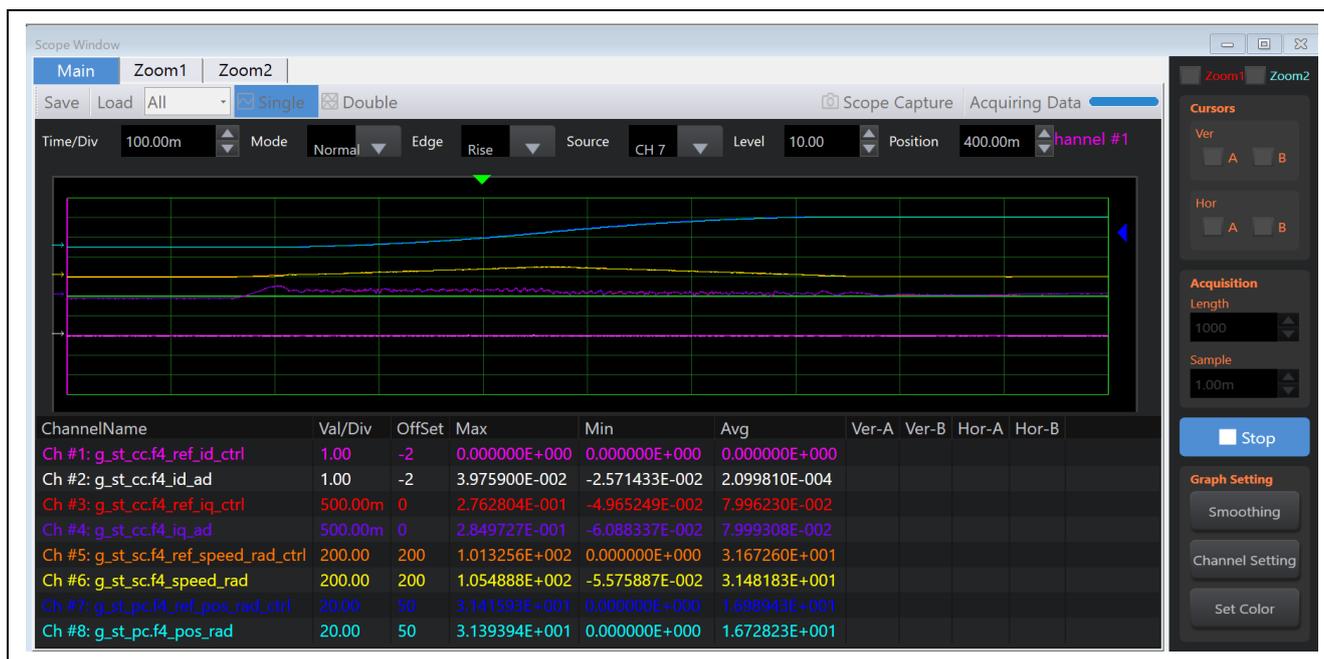


図 7-2 エンコーダを使用した位置制御

駆動条件：

- ・ 位置指令値：CW 方向に 5 回転（1800 度）
- ・ 位置プロファイルの最大速度：4000 [rpm]
- ・ 加減速時間：300 [ms]

波形情報：

黄色：検出速度 [rad/s], (200rad/s / div.)、オレンジ：指令速度 [rad/s], (200rad/s / div.)

赤：q 軸電流指令値 [A], (500mA / div.)、紫：q 軸電流値 [A], (500mA / div.)、

ピンク：d 軸電流指令値 [A], (1A / div.)、白：d 軸電流値 [A], (1A / div.)

水色：センサから算出した角度情報(機械角) [rad], (20rad/div.)、青：位置指令値[rad], (20rad/div.)

横軸：100ms / div.

8. 参考資料

- Renesas Motor Workbench ユーザーズマニュアル (R21UZ0004)
- MCK-RX26T ユーザーズマニュアル (RZ12UZ0111)
- スマート・コンフィグレータ ユーザーズマニュアル RX API リファレンス編 (R20UT4360)
- RX スマート・コンフィグレータ ユーザーガイド : CS+編 (R20AN0470)
- RX スマート・コンフィグレータ ユーザーガイド : e² studio 編 (R20AN0451)
- RX26T グループ ユーザーズマニュアル ハードウェア編 (R01UH0979)
- MCB-RX26T Type A ユーザーズマニュアル (R12UZ0112)
- MCB-RX26T Type C ユーザーズマニュアル (R12UZ0127)

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2023.05.30	—	初版発行
1.10	2023.08.29	—	使用 MCU 追加 (R5F526TACDFM)

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS製品の取り扱いの際は静電気防止を心がけてください。CMOS製品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、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、CMOS製品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。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れしないでください。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「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 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製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 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などが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられているリザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違えば、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 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。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、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
 2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
 3. 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。
 4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
 5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等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。
 6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
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、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等
当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
 7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア/ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限られません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア/ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
 8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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いますが、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、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等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、お客様の機器・システムとしての出荷保証を行ってください。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、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 10. 当社製品の環境適合性等の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。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 11. 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・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
 13. 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禁じます。
 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配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。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレストシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。