

# 永久磁石同期モータの 120 度通電制御

## Evaluation System for BLDC Motor 用

### 要旨

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を使用し、永久磁石同期モータを 120 度通電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 :
- RX24U (R5F524UEADFB)
- RX24T (R5F524TAADFP)
- RX23T (R5F523T5ADFM)
- RX13T (R5F513T5ADFL)

### 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。

- RXxxx\_ESB\_SPM\_120\_CONDUCTION\_CSP\_V100 (IDE : CS+版) \*1
- RXxxx\_ESB\_SPM\_120\_CONDUCTION\_E2S\_V100 (IDE : e<sup>2</sup>studio 版) \*1  
Evaluation System For BLDC Motor & RXxxx CPU カード向けエンコーダベクトル制御ソフトウェア\*1

【注】 1. “xxx”は使用する MCU 名称です。

## 目次

1. 概要 .....	3
2. 開発環境 .....	4
2.1 動作確認環境 .....	4
2.2 ハードウェア仕様 .....	5
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 .....	16
4. ソフトウェア .....	17
4.1 ソフトウェアスペック .....	17
4.2 ソフトウェア構成 .....	18
4.3 ファイル/フォルダ構成 .....	20
5. 機能 .....	23
5.1 アプリケーション層 .....	23
5.2 マネージャモジュール .....	31
5.3 速度制御モジュール .....	51
5.4 センサモジュール（ホールセンサ） .....	59
5.5 ドライバモジュール .....	68
5.6 スマートコンフィグレータ設定 .....	74
6. 120 度通電制御アルゴリズム .....	78
6.1 120 度通電制御 .....	78
6.2 120 度通電制御における位置・速度算出 .....	79
7. 試験結果 .....	87
7.1 プログラムサイズ .....	87
7.2 CPU 負荷率 .....	88
7.3 動作波形 .....	89
8. 参考資料 .....	91
9. 改訂記録 .....	92

## 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、永久磁石同期モータを 120 度通電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(Evaluation System for BLDC Motor)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。

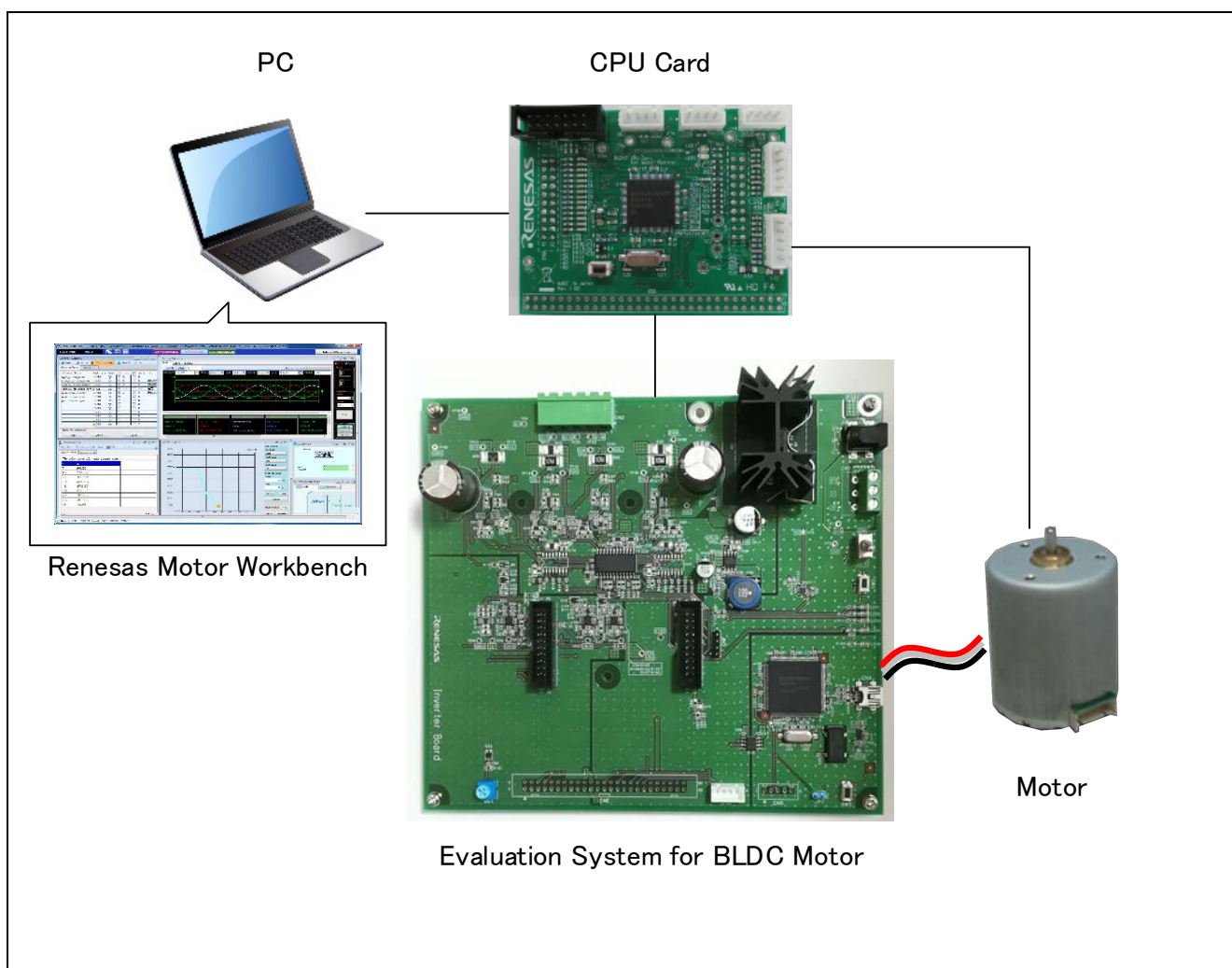


図 1-1 サンプルプログラムを使用した動作環境

## 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 カード型名	RX24U (R5F524UEADFB) / RTK0EMX590C02000BJ RX24T (R5F524TAADFP) / RTK0EM0009C03402BJ RX23T (R5F523T5ADFM) / RTK0EM0003C01202BJ RX13T (R5F513T5ADFL) / RTK0EMXA10C00000BJ
インバータボード	Evaluation System for BLDC Motor (RTK0EM0000B10020BJ) 同梱 48V 5A BLDC 用インバータボード
モータ	TG55L-KA (ツカサ電工株式会社製)
センサ	ホール : TG55L-KA (ツカサ電工株式会社製) 付属 センサレス : なし

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+ : V8.07.00	バージョン 2.12.0	CC-RX : V3.04.00
e <sup>2</sup> studio : 2022-01	e <sup>2</sup> 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

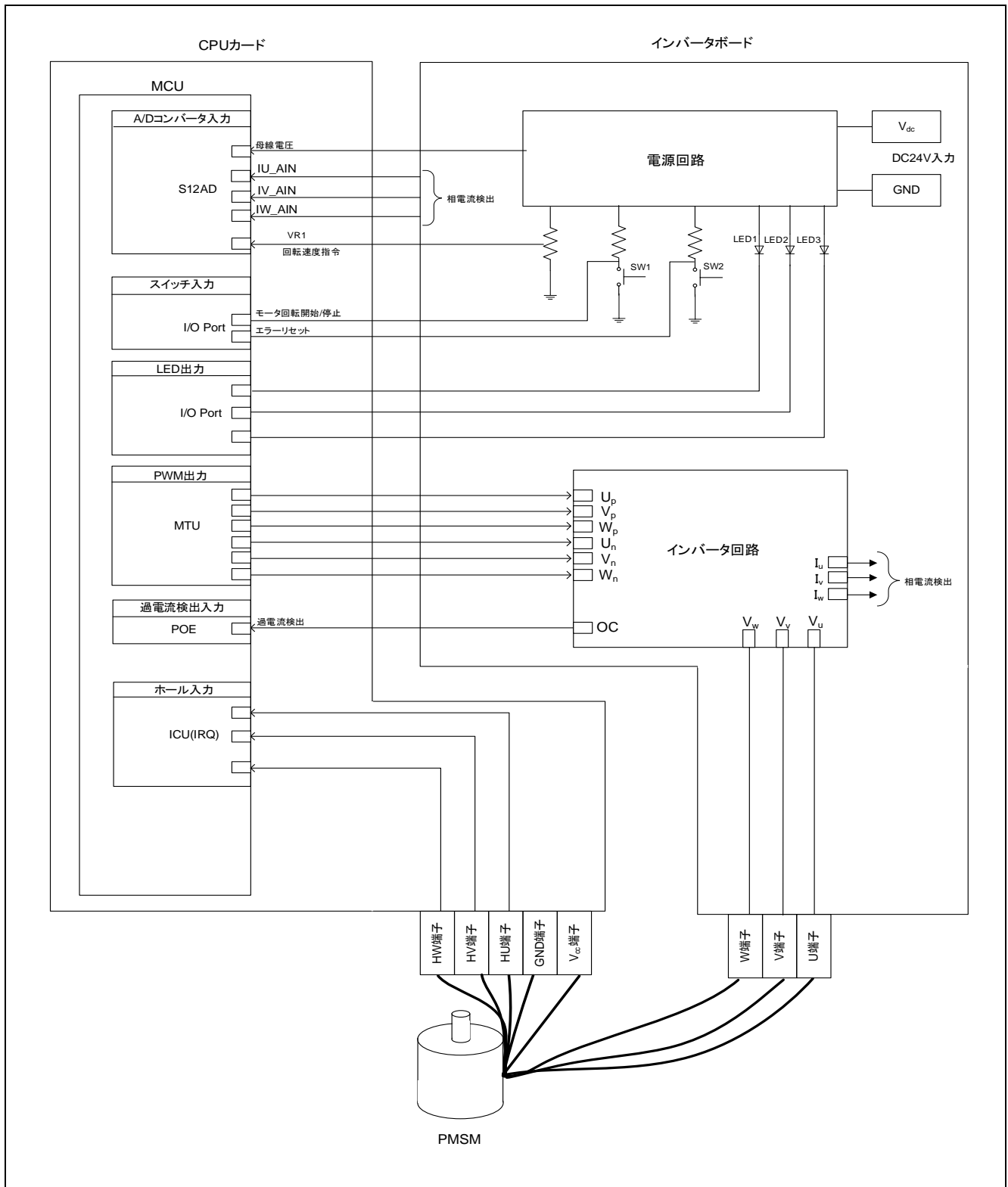


図 2-1 ハードウェア構成図 (ホールセンサ使用)

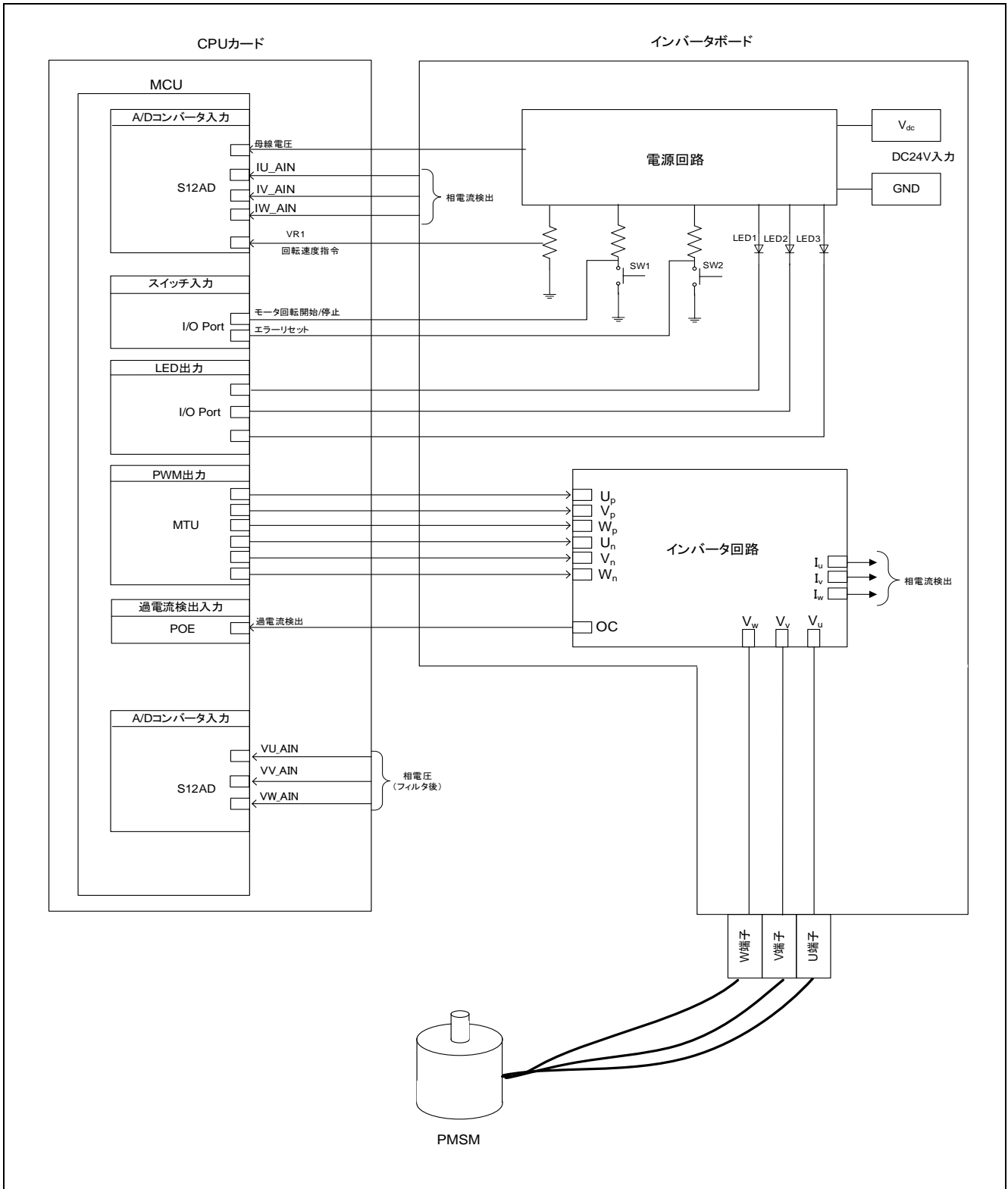


図 2-2 ハードウェア構成図 (センサレス)

## 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	<ul style="list-style-type: none"><li>モータ回転時：点灯</li><li>停止時：消灯</li></ul>
LED2	オレンジ色 LED	<ul style="list-style-type: none"><li>エラー検出時：点灯</li><li>通常動作時：消灯</li></ul>
LED3	オレンジ色 LED	本システムでは未使用
RESET	プッシュスイッチ(RESET1)	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。使用する CPU カードによって端子割り当てが異なるため、MCU 周辺機能の割り当ても異なります。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、5.6 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
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)
LED3 点灯/消灯制御	I/O Port (output)
U 相電流測定	S12AD
V 相電流測定	S12AD
W 相電流測定	S12AD
U 相電圧測定	S12AD
V 相電圧測定	S12AD
W 相電圧測定	S12AD
PWM 出力 (U <sub>p</sub> ) / “Low” アクティブ	MTU
PWM 出力 (V <sub>p</sub> ) / “Low” アクティブ	MTU
PWM 出力 (W <sub>p</sub> ) / “Low” アクティブ	MTU
PWM 出力 (U <sub>n</sub> ) / “High” アクティブ	MTU
PWM 出力 (V <sub>n</sub> ) / “High” アクティブ	MTU
PWM 出力 (W <sub>n</sub> ) / “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)
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 Evaluation System for BLDC Motor 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。Evaluation System for BLDC Motor のボード設定、接続に関しては Evaluation System for BLDC Motor ユーザーズマニュアル (R12UZ0062) 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 カードの 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。E2 on board 非搭載の CPU カードではプログラムを書き込むために専用エミュレータが必要です。

#### 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。

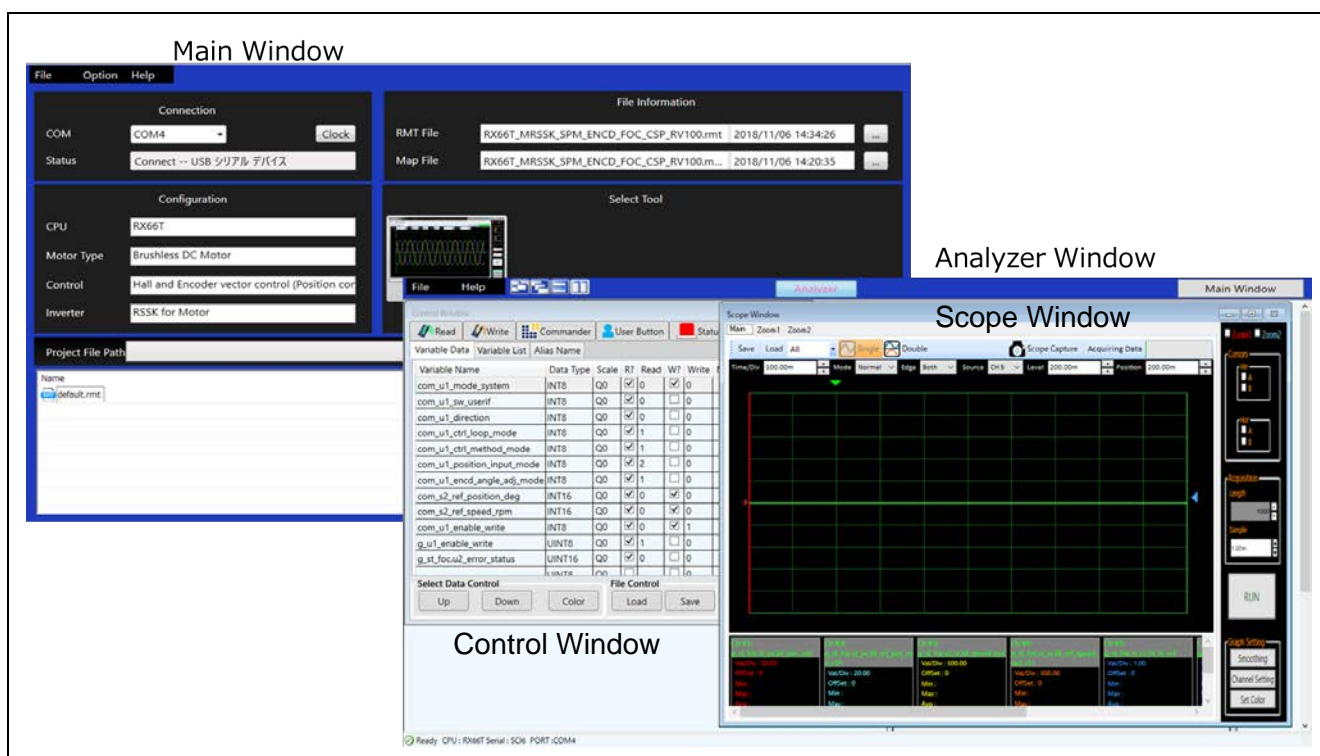



図 3-1 Renesas Motor Workbench 外観

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_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 120 度通電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 5.1.4 を参照してください。

表 3-2 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_120_conduction.u2_error_status	uint16_t	エラーステータス
g_st_speed.f4_ref_speed_rad	float	速度指令値 (機械角) [rad/s]
g_st_speed.f4_speed_rad	float	速度検出値 (機械角) [rad/s]
g_st_speed.f4_v_ref	float	指令電圧 [V]

### 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。

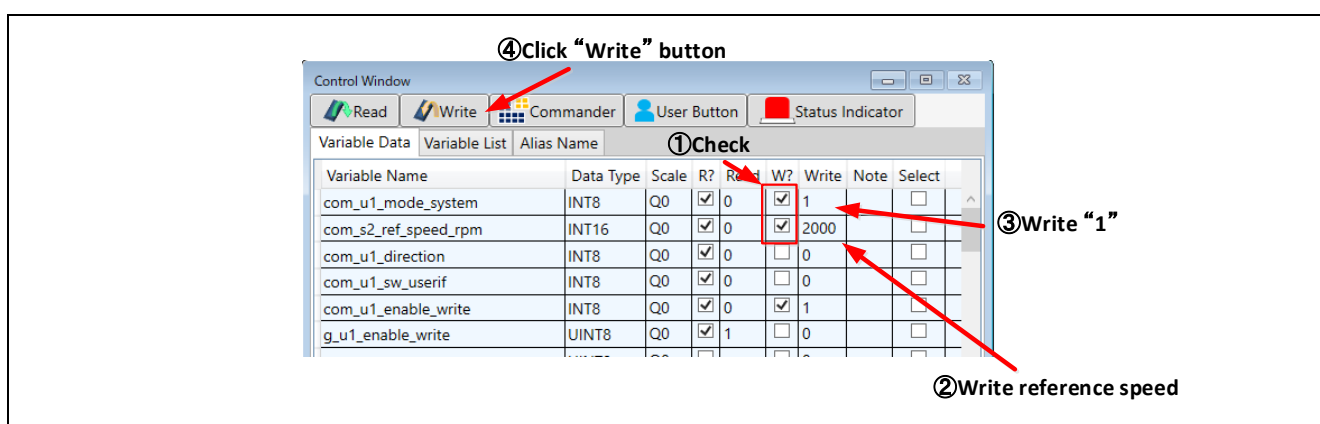


図 3-2 モータ回転の手順

##### (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。

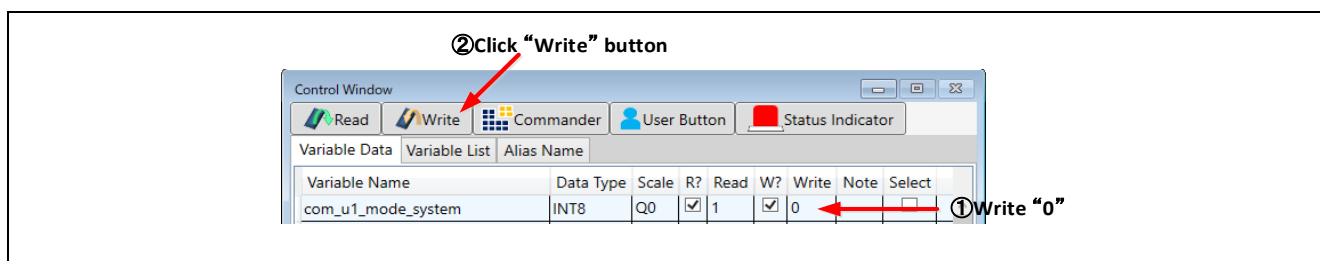


図 3-3 モータ停止の手順

## (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。

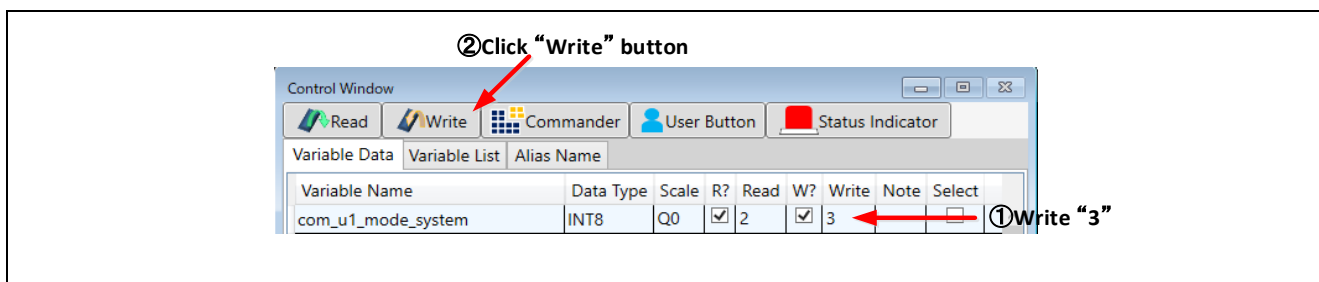


図 3-4 エラー解除の手順

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。

- モータを速度制御で駆動する／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。

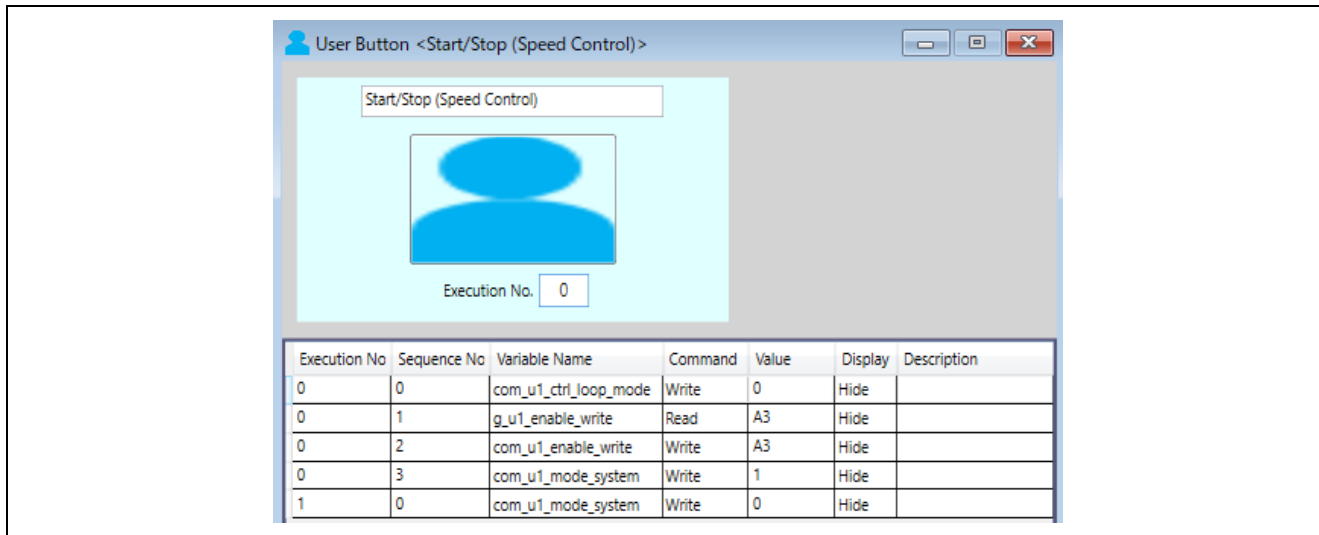


図 3-5 モータの駆動／停止

- 速度指令を変更する  
 図 3-6 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。

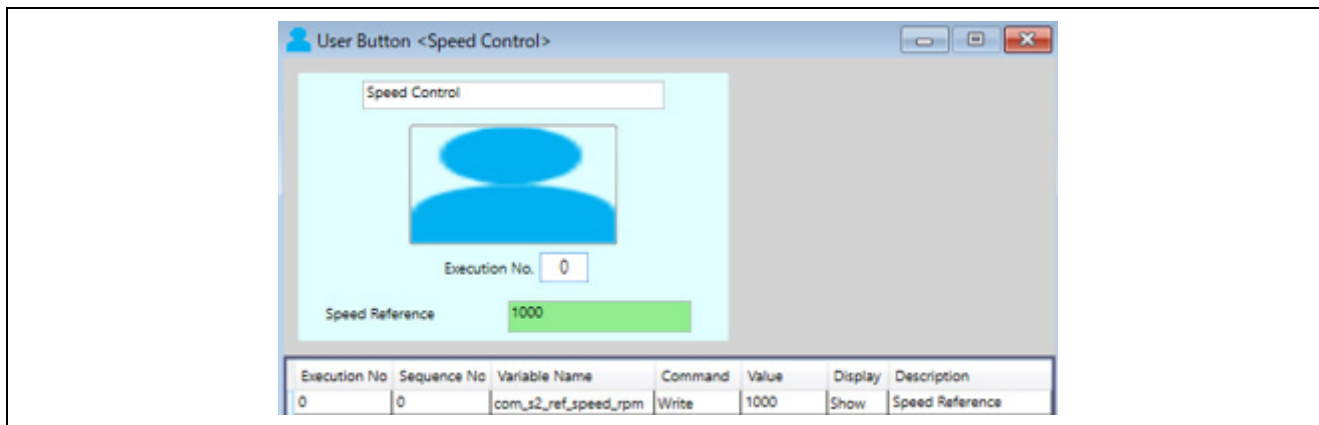


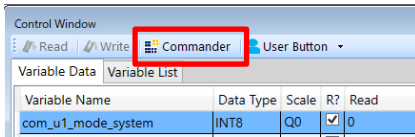
図 3-6 速度指令の変更

### 3.4.3 コマンダー機能操作例

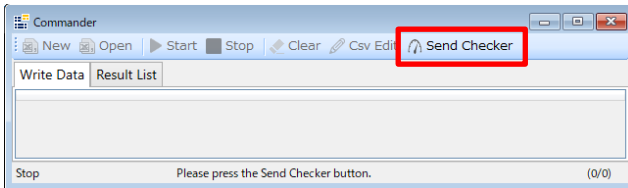
コマンダー機能を使用した位置制御：

(Commander の起動)

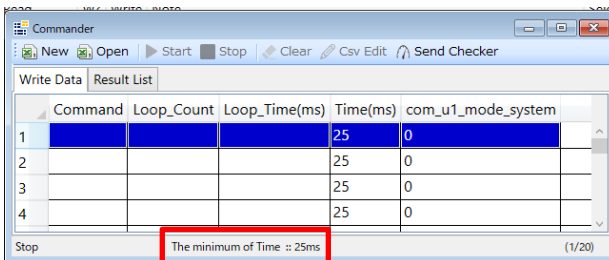
- ① Control Window の“Commander”ボタンを押す。



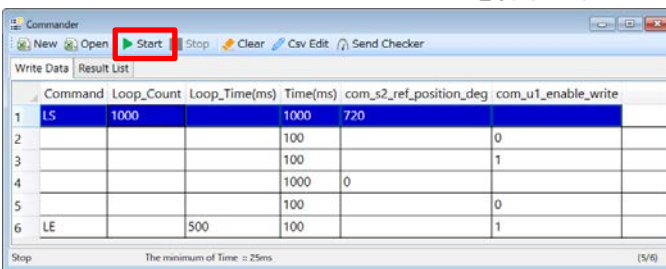
- ② Commander ウィンドウが立ち上がるので“Send Checker”ボタンを押し、データの送信速度を確認する。



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。速度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。モータが速度制御を開始する。
- 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。

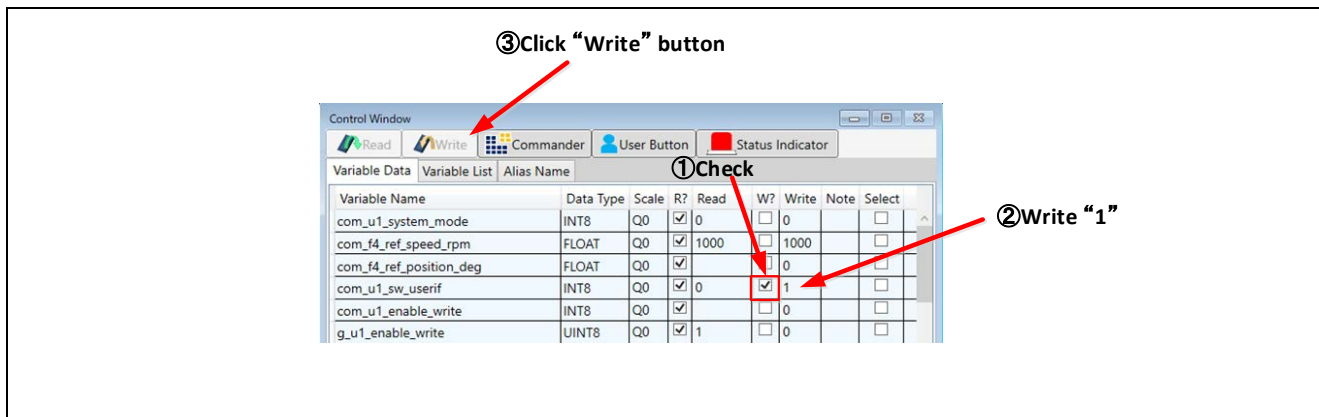


図 3-7 UI の切り替えの手順

#### 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、“Low”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“High”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、以下の表のように速度指令値として使用します。

表 3-3 指令値の変換比

項目	変換比 (指令値 : A/D 変換値)	
	回転速度指令値	CW
	CCW	0 [rpm]~-2650 [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 4-1 120 度通電制御ソフトウェア基本仕様

項目	内容	
制御方式	120 度通電方式（前半 60 度チョッピング）	
モータ制御開始／停止	SW1 のレベルにより判定（“Low”：制御開始 “High”：停止） または RMW から入力	
回転子磁極位置検出	ホールセンサ：ホールセンサによる位置検出（60 度毎） センサレス：誘起電圧による位置検出（60 度毎）	
入力電圧	DC 24V	
キャリア周波数 （PWM）	20 [kHz]、キャリア周期：50 [μs]	
デッドタイム	2 [μs]	
制御周期 （速度）	ホールセンサ： <ul style="list-style-type: none"> <li>ホールセンサ入力信号の両エッジにおける割り込み信号毎</li> </ul> センサレス： <ul style="list-style-type: none"> <li>キャリア周期毎に誘起電圧からゼロクロス判定を行う</li> <li>パターン切り替え時、PWM デューティ設定と通電パターンの決定を行う</li> </ul> 速度 PI 制御は 2 [ms]毎	
速度指令値管理	ホールセンサ：CW/CCW 共に 550 [rpm] ～ 2650 [rpm] センサレス：CW/CCW 共に 1000 [rpm] ～ 2650 [rpm]	
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 1. 各相の電流が 0.89 [A]を超過（50 [μs]毎に監視） 2. インバータ母線電圧が 28 [V]を超過（50 [μs]毎に監視） 3. インバータ母線電圧が 14 [V]未満（50 [μs]毎に監視） 4. 回転速度が 3000 [rpm]を超過（50 [μs]毎に監視） 5. ホールセンサ駆動時、ホール割り込みが 200[ms]間未発生 6. センサレス駆動時、ゼロクロス検出が 100 [ms]間未発生 7. ホールセンサパターンエラー 8. 仮想ホールセンサパターン（位置情報）の異常検出  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	

## 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。

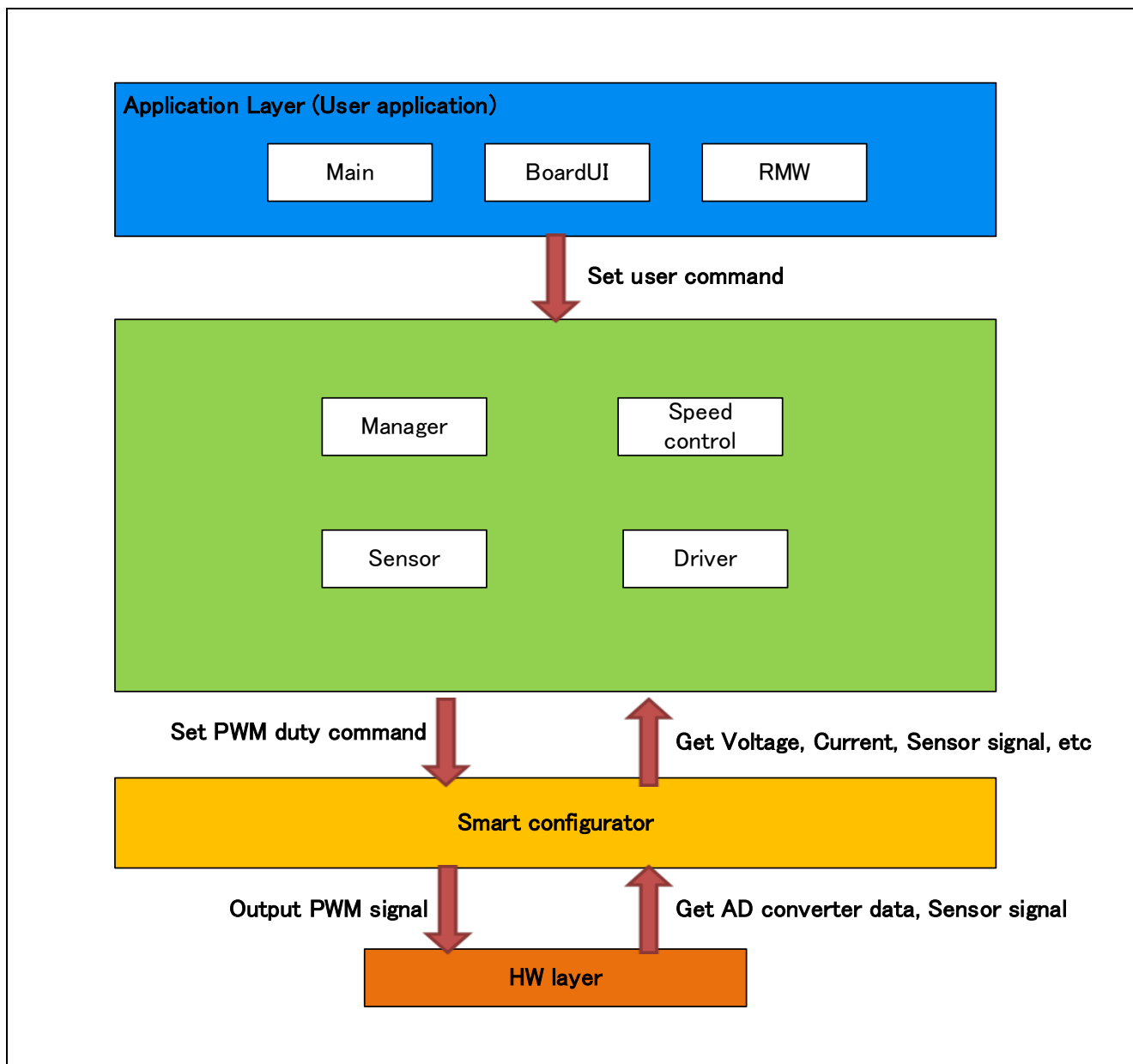


図 4-1 モータ制御ソフトウェアの全体構成

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。

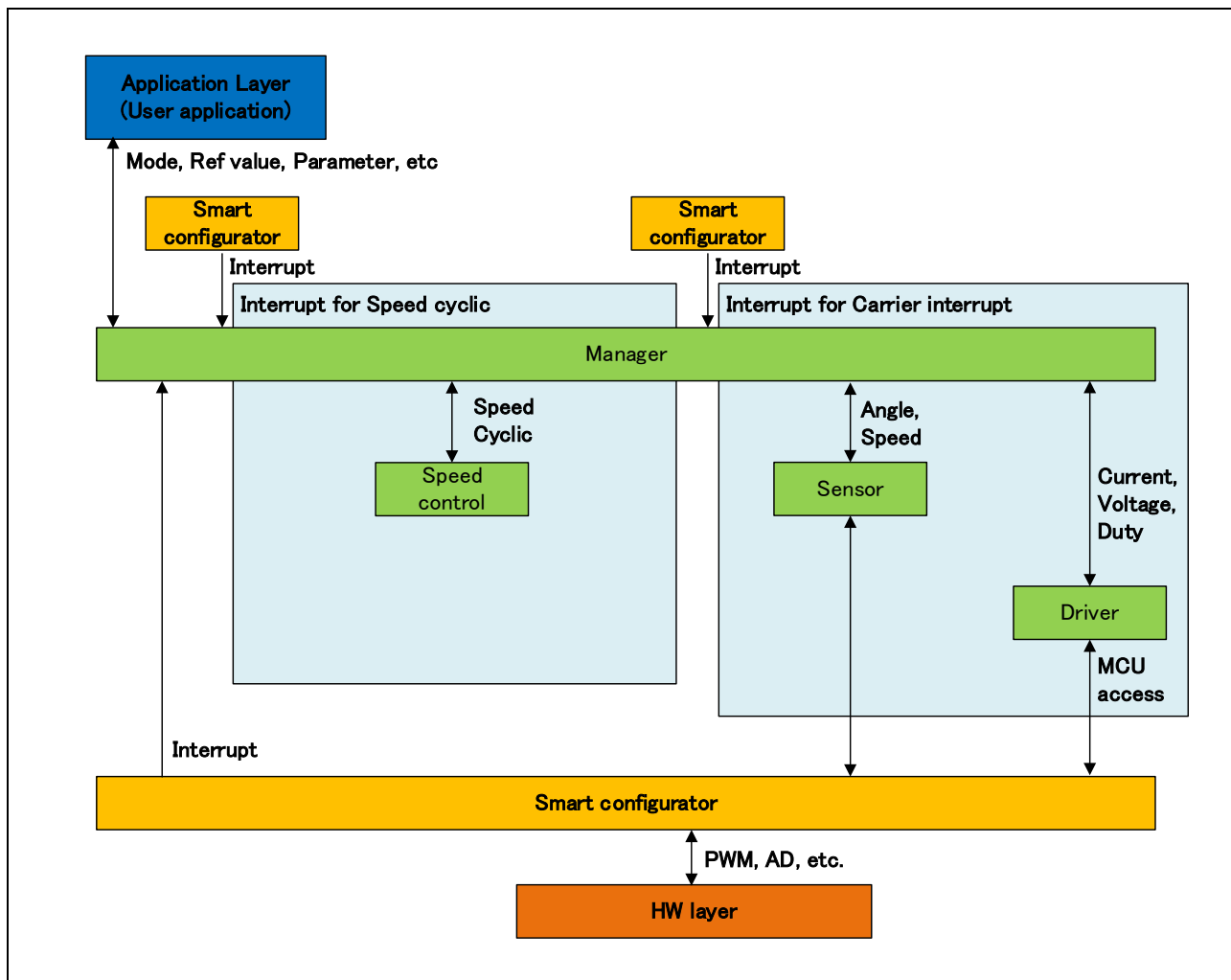


図 4-2 モータモジュールの構成

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
速度制御モジュール	速度制御に関するモジュール	5.3
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	5.4
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	5.5
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	5.6

## 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_RX"xxx".lib/h、 ICS_RX"xxx".obj/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_rx"xxx"_esb.c	MCU 依存のボード UI 関数定義
motor_module	120_conduction_rx	r_motor_120_conduction_action.c	アクション関数定義
		r_motor_120_conduction_api.c/h	マネージャモジュールの API 関数定義
		r_motor_120_conduction_manager.c/h	マネージャモジュールのローカル関数定義
		r_motor_120_conduction_protection.c/h	保護機能の関数定義
		r_motor_120_conduction_statemachine.c/h	状態遷移関連の関数定義
	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
	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.c/h	センサモジュールのローカル関数定義
	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
	src	smc_gen	別表

【注】 "xxx"は MCU の名称が入力される。ex. RX24T など

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。RX24T を参考例として示します。

“RX24T\_ESB\_SPM\_120\_xxx\_RVyyy.scfg”

(xxx : CSP は CS+版、E2S は e<sup>2</sup> studio 版を意味します。yyy : 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 (RX24T)

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_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
		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_CMT1	Config_CMT1.c/h	速度演算用 CMT 関連関数定義
			Config_CMT1_user.c	速度演算用 CMT 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_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

上記表の他に、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に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。

## スマートコンフィグレータ生成コードに関する注意点

スマートコンフィグレータのモータ・コンポーネントは、モータ駆動に必要なマルチファンクションタイマパルスユニット及び 12 ビット A/D コンバータという複数の周辺機能を 1 つのインタフェースで構成することができ、シンプルで理解しやすい GUI として提供しています。

ただし、12 ビット A/D コンバータ専用コンポーネントなど、同じ周辺機能に関するコードを生成する他のコンポーネントも同時に使用する場合、レジスタ設定が上書きされてしまう可能性があります。この場合、影響を受けるコンポーネントによって生成された“<コンフィグレーション名>\_user.c”ファイルを利用して、対策を施すことができます。

## &lt;参考：RX24T のサンプルプログラムの場合&gt;

12 ビット A/D コンバータ・コンポーネントの初期化関数“R\_Config\_S12AD2\_Create”で AN209 を A/D 変換対象として設定します。しかし、この初期化関数設定の後にモータ・コンポーネントの初期化関数“R\_Config\_MOTOR\_Create”がコールされる場合は、AN201～AN204 を A/D 変換対象として設定する際に、AN209 が A/D 変換対象から外れてしまいます。そのため、“Config\_MOTOR\_user.c”内で AN209 を A/D 変換対象として再設定しています。また、モータ・コンポーネントの初期化関数設定が 12 ビット A/D コンバータ・コンポーネントの初期化関数設定より前にある場合は、AN209 の設定時に AN201～AN204 が A/D 変換対象から外れてしまうため、“Config\_S12AD0\_user.c”内で AN201～AN204 を A/D 変換対象として再設定しています。

サンプルプログラムでは上記 2 つの再設定を行っているため、12 ビット A/D コンバータ・コンポーネントの初期化関数とモータ・コンポーネントの初期化関数の順番に関わらず、正しい設定が行われています。

## 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	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。

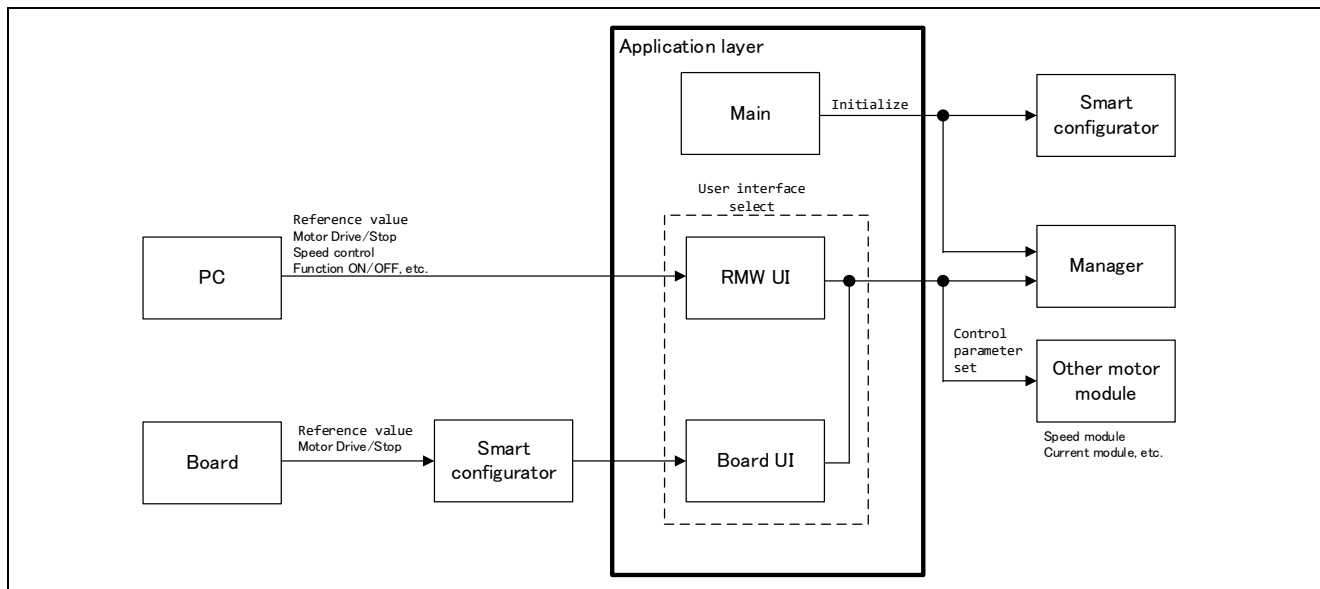


図 5-1 アプリケーション層の構成図

5.1.3 フローチャート

(a) メイン処理

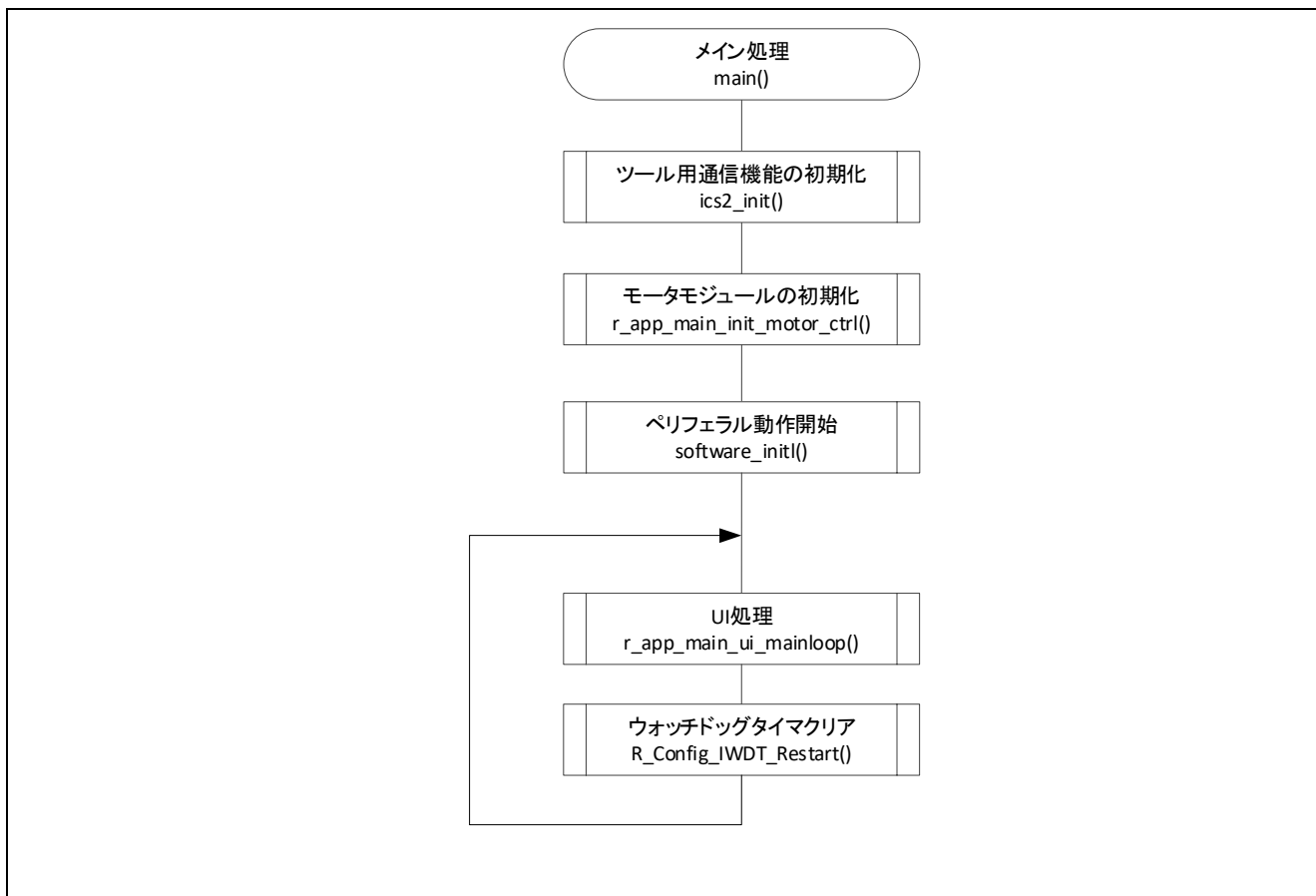


図 5-2 メイン処理フローチャート



(b) UI 処理

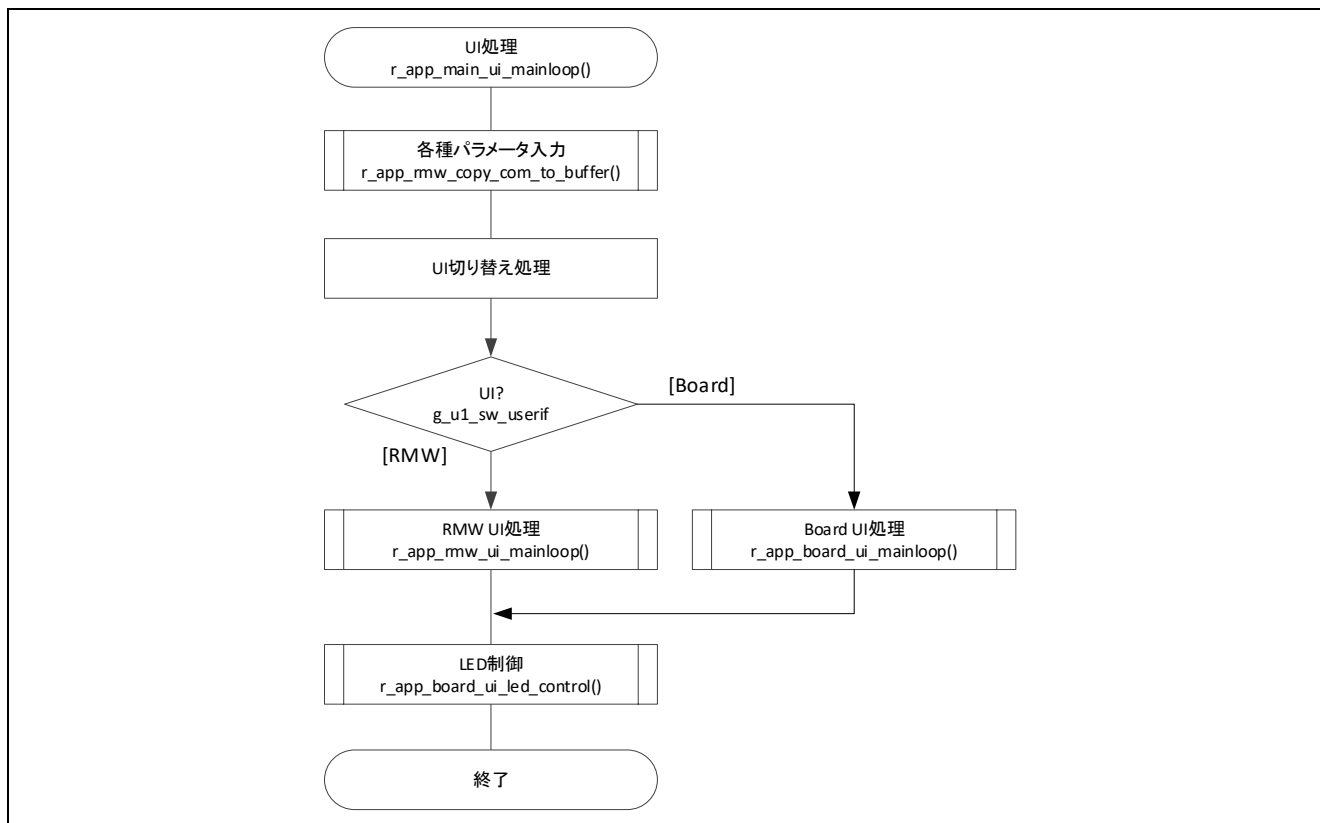


図 5-3 UI 処理フローチャート

(c) ボード UI 処理

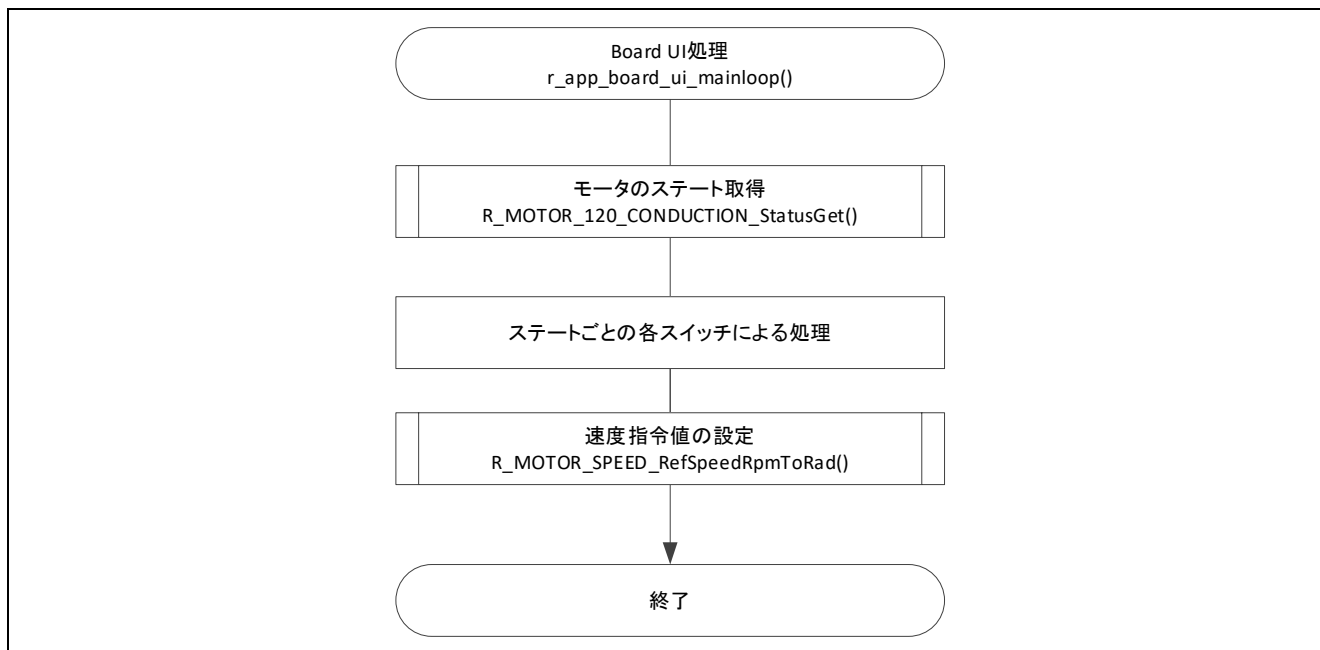


図 5-4 ボード UI 処理フローチャート

(d) RMW UI 処理

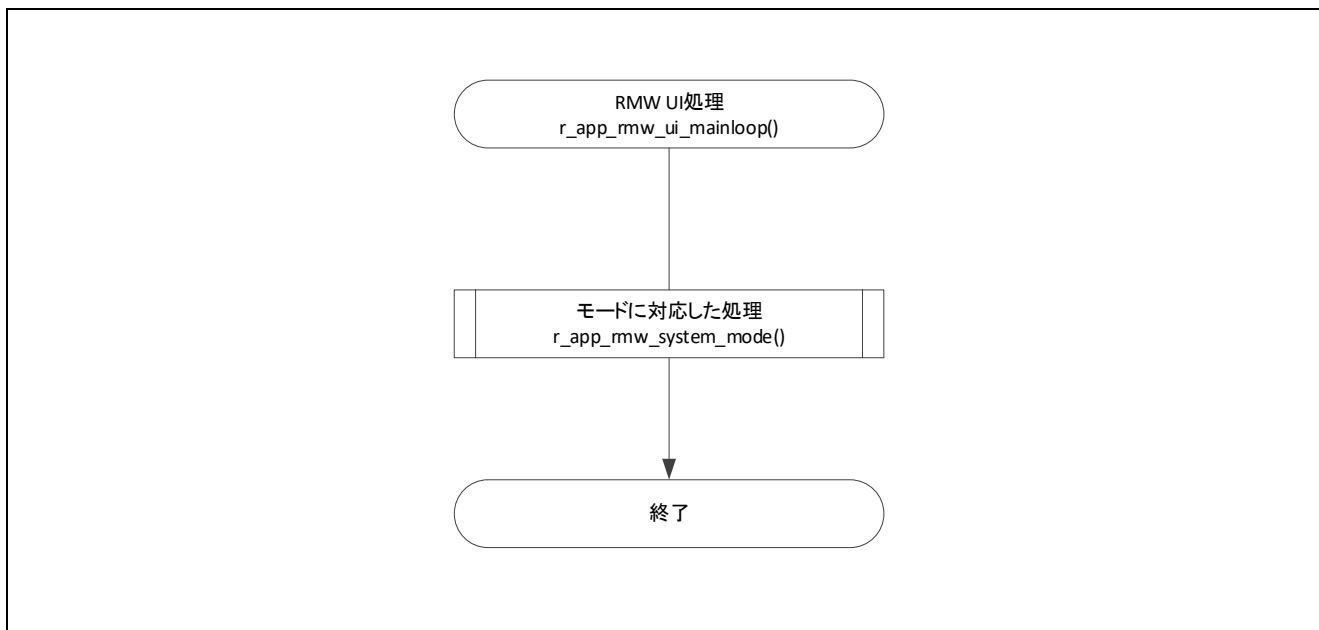


図 5-5 RMW UI 処理フローチャート

## 5.1.4 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 5-2 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 5-3 に示します。

表 5-2 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_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_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_s2_ref_speed_rpm	速度指令値切り替え変数 [rpm]
g_s2_ref_speed	速度指令値 [rpm]
com_f4_kp_speed	速度制御の制御ゲイン比例項。
com_f4_ki_speed	速度制御の制御ゲイン微分項。
com_f4_speed_lpf_k	速度制御用 LPF のゲイン
com_u2_mtr_p	駆動するモータの極対数
com_f4_limit_speed_change	加速度の設定
com_u2_offset_calc_time	電流オフセット測定時間設定 [s]
com_f4_start_refv	始動時の指令電圧設定 [V]
com_u1_hall_wait_cnt	ホール割り込み時間カウンタ設定
com_s2_ol_start_rpm	オープンループ開始時の速度設定 [rpm]
com_s2_ol_mode1_change_rpm	オープンループモード 1*1 への移行速度設定 [rpm] (センサレスのみ)
com_s2_ol_mode2_change_rpm	オープンループモード 2*1 への移行速度設定 [rpm] (センサレスのみ)
com_f4_ol_start_refv	オープンループ始動時の指令電圧 [V] (センサレスのみ)
com_f4_ol_mode1_rate_rpm	オープンループモード 1*1 の速度変化量設定 [rpm] (センサレスのみ)
com_f4_ol_mode2_rate_refv	オープンループモード 2*1 の電圧変化量設定 [V] (センサレスのみ)

変数	説明
com_f4_ol_mode2_rate_rpm	オープンループモード 2*1 の速度変化量設定 [rpm] (センサレスのみ)
com_f4_ol_mode3_rate_refv	オープンループモード 3*1 の電圧変化量設定 [V] (センサレスのみ)
com_f4_ol_mode3_max_refv	オープンループモード 3*1 の最大電圧設定 [V] (センサレスのみ)
com_f4_boot_ref_v	ブートモード*1 の電圧設定 [V] (センサレスのみ)
com_u2_v_up_time	ブートモード時の電圧上昇時間 [s] (センサレスのみ)
com_u2_v_const_time	ブートモード時の電圧一定時間 [s] (センサレスのみ)

【注】 1. 詳細は 5.2.11 始動シーケンス管理を参照ください。

表 5-3 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  RMW 変数更新用 構造体	s2_ref_speed	指令速度 [rpm]
	f4_kp_speed	速度制御の制御ゲイン比例項。
	f4_ki_speed	速度制御の制御ゲイン微分項。
	f4_speed_lpf_k	速度制御用 LPF のゲイン
	u2_mtr_p	駆動するモータの極対数
	f4_limit_speed_change	加速度の設定
	u2_offset_calc_time	電流オフセット測定時間設定 [s]
	f4_start_refv	始動時の指令電圧設定 [V]
	u1_hall_wait_cnt	ホール割り込み時間カウンタ設定
	s2_ol_start_rpm	オープンループ開始時の速度設定 [rpm] (センサレスのみ)
	s2_ol_mode1_change_rpm	オープンループモード 1*1 への移行速度設定 [rpm] (センサレスのみ)
	s2_ol_mode2_change_rpm	オープンループモード 2*1 への移行速度設定 [rpm] (センサレスのみ)
	f4_ol_start_refv	オープンループ開始時の指令電圧 [V] (センサレスのみ)
	s2_ol_start_rad	オープンループ開始時の速度設定 [rad/s] (センサレスのみ)
	f4_ol_mode1_rate_rpm	オープンループモード 1*1 の速度変化量設定 [rpm] (センサレスのみ)
	f4_ol_mode2_rate_refv	オープンループモード 2*1 の電圧変化量設定 [V] (センサレスのみ)
	f4_ol_mode2_rate_rpm	オープンループモード 2*1 の速度変化量設定 [rpm] (センサレスのみ)
	f4_ol_mode3_rate_refv	オープンループモード 3*1 の電圧変化量設定 [V] (センサレスのみ)
	f4_ol_mode3_max_refv	オープンループモード 3*1 の最大電圧設定 [V] (センサレスのみ)
	f4_boot_ref_v	ブートモード*1 の電圧設定 [V] (センサレスのみ)
u2_v_up_time	ブートモード時の電圧上昇時間 [s] (センサレスのみ)	
u2_v_const_time	ブートモード時の電圧一定時間 [s] (センサレスのみ)	

【注】 1. 詳細は 5.2.11 始動シーケンス管理を参照ください。

## 5.1.5 マクロ定義

マクロ一覧を表 5-4 に示します。

表 5-4 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_RMW	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_SW_ON	0	スイッチ ON
	BOARD_SW_OFF	1	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_SPEED_MARGIN	50	VR1 用速度マージン [rpm]
	BOARD_VR1_SCALING_SPEED	$(\text{MOTOR_CFG_MAX\_SPEED\_RPM} + 50.0f) / 2048$	VR1 用速度スケーリング係数
r_app_rmw.h	ICS_DECIMATION	3	RMW watchpoint のスキップ回数
	ICS_INT_LEVEL	6	RMW 割り込みの優先度
	ICS_BRR	4	RMW の通信ボーレート
	ICS_INT_MODE	1	RMW の通信モード選択
	ICS_SCI_CH_SELECT	0x60	使用 SCI チャンネル

【注】 RMW で通信を行うためのチャンネルを定義したマクロが MCU 毎に ICS2\_RXxxx.h に用意されています。“xxx”は MCU 名称です。

表 5-5 マクロ定義一覧

マクロ名	RX13T	RX23T	RX24T, RX24U
ICS_DECIMATION	3	3	3
ICS_INT_LEVEL	6	6	6
ICS_BRR	3	4	4
ICS_INT_MODE	1	1	1

## 5.1.6 パラメータ調整・設定

表 5-2 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 V.2.00 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 5-6 に示します。

表 5-6 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	速度制御の状態の取得・設定を行います。
速度情報の取得	速度情報の取得を行います。
制御モジュールの指令値設定	速度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。

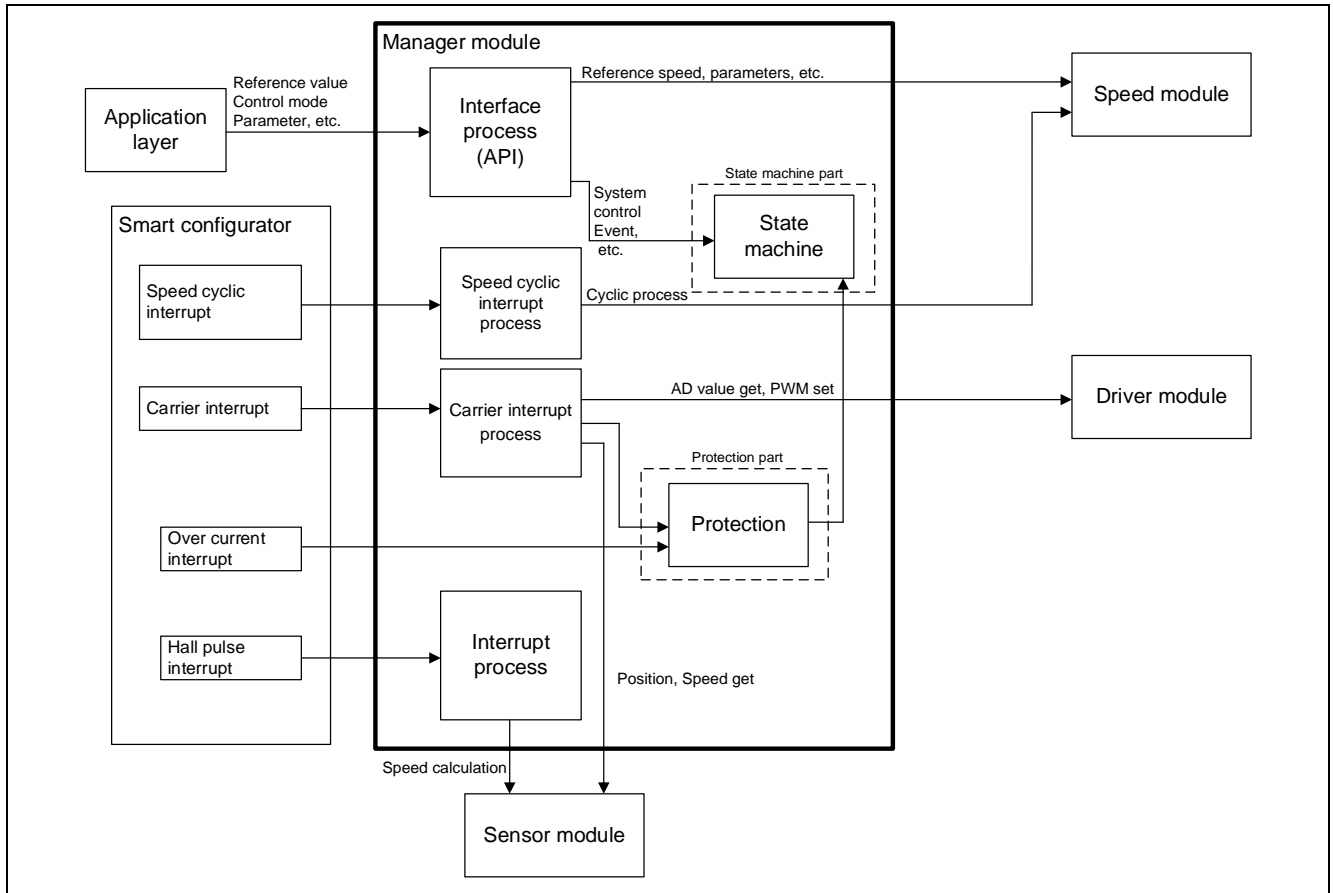


図 5-6 マネージャモジュール構成図



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。

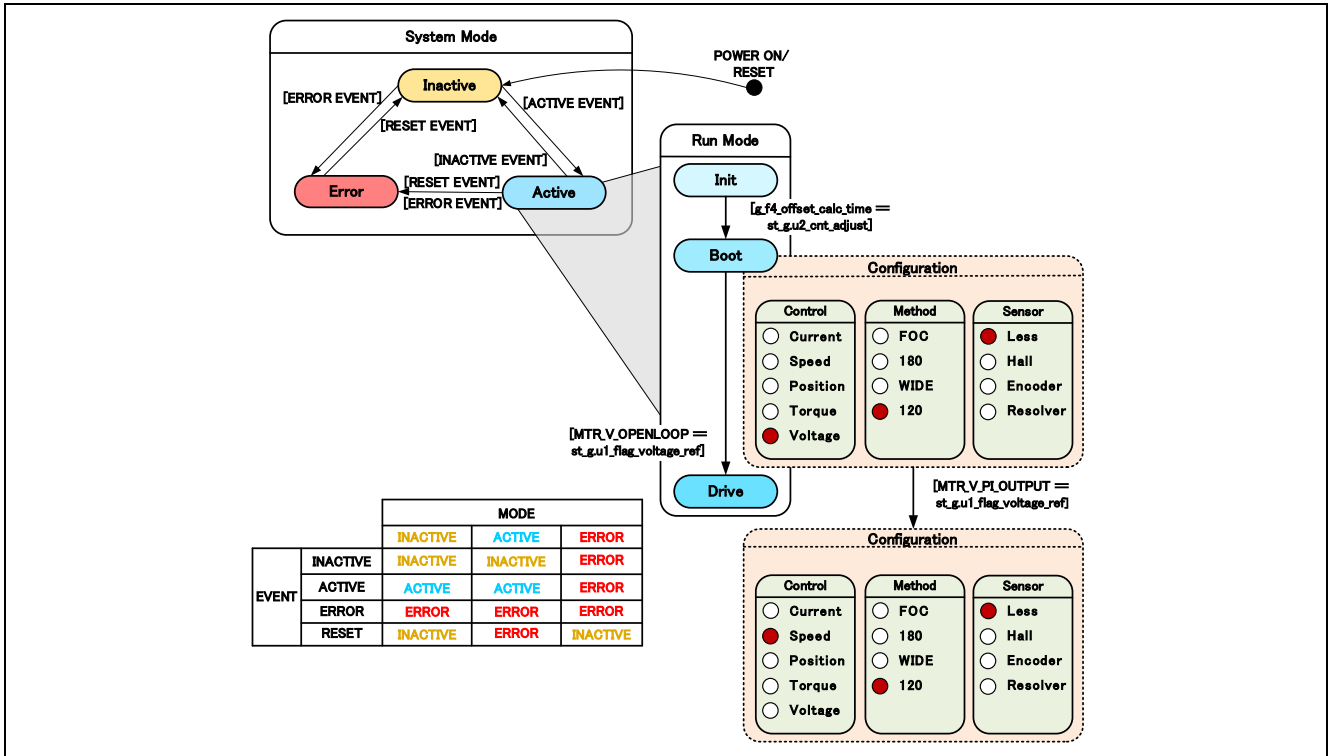


図 5-7 エンコーダベクトル制御ソフトウェアの状態遷移図

(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 5-7 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 5-8 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 (MP\_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 5-8 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]	0.89
	監視周期 [ $\mu$ s]	電流制御周期* <sup>1</sup>
過電圧エラー	過電圧リミット値 [V]	28
	監視周期 [ $\mu$ s]	電流制御周期* <sup>1</sup>
低電圧エラー	低電圧リミット値 [V]	14
	監視周期 [ $\mu$ s]	電流制御周期* <sup>1</sup>
回転速度エラー	速度リミット値 [rpm]	3000
	監視周期 [ $\mu$ s]	電流制御周期* <sup>1</sup>

【注】 1. 表 4-1 120 度通電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。

(a) キャリア割り込み処理

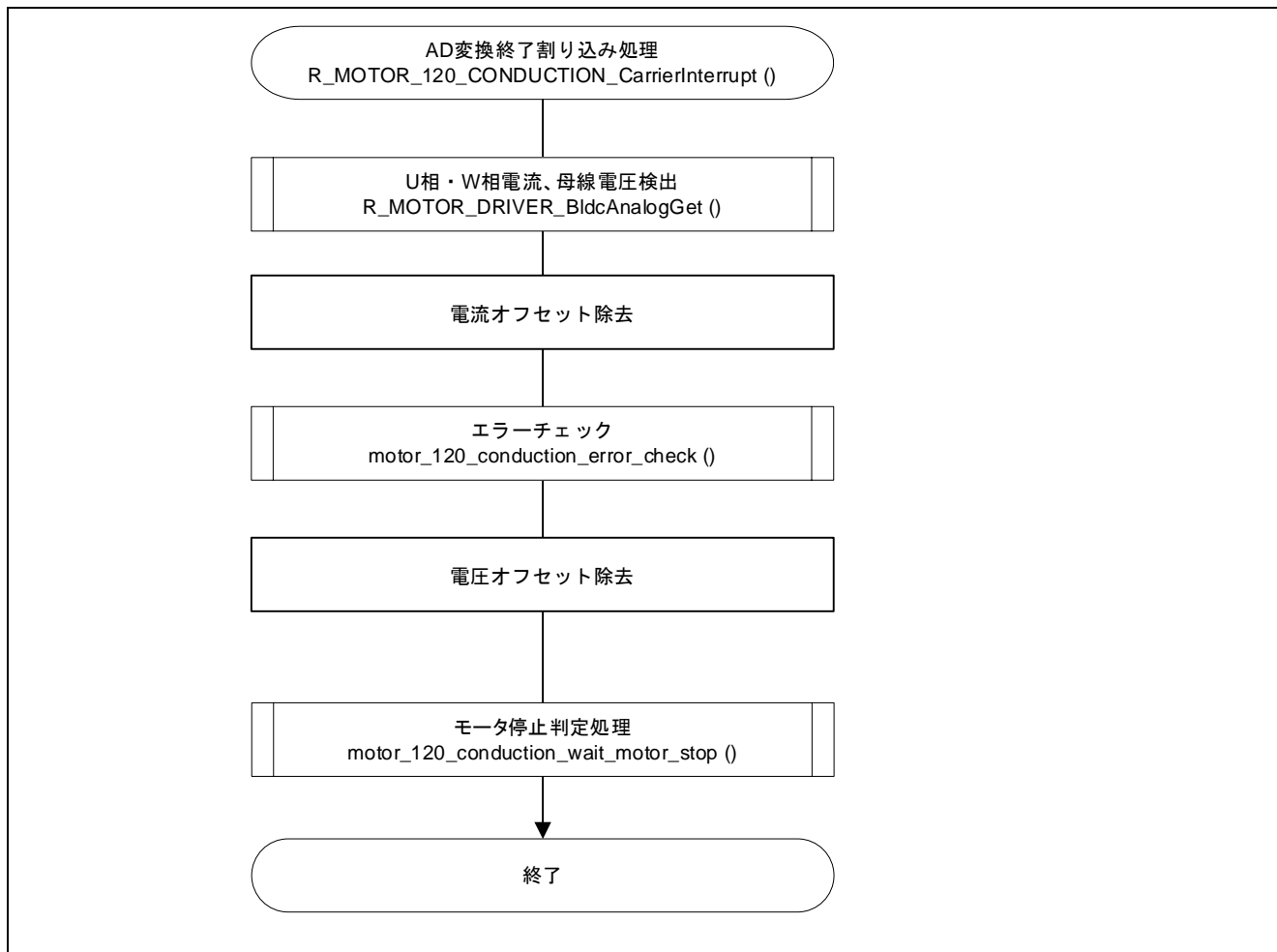


図 5-8 キャリア割り込み処理フローチャート（ホールセンサ使用時）

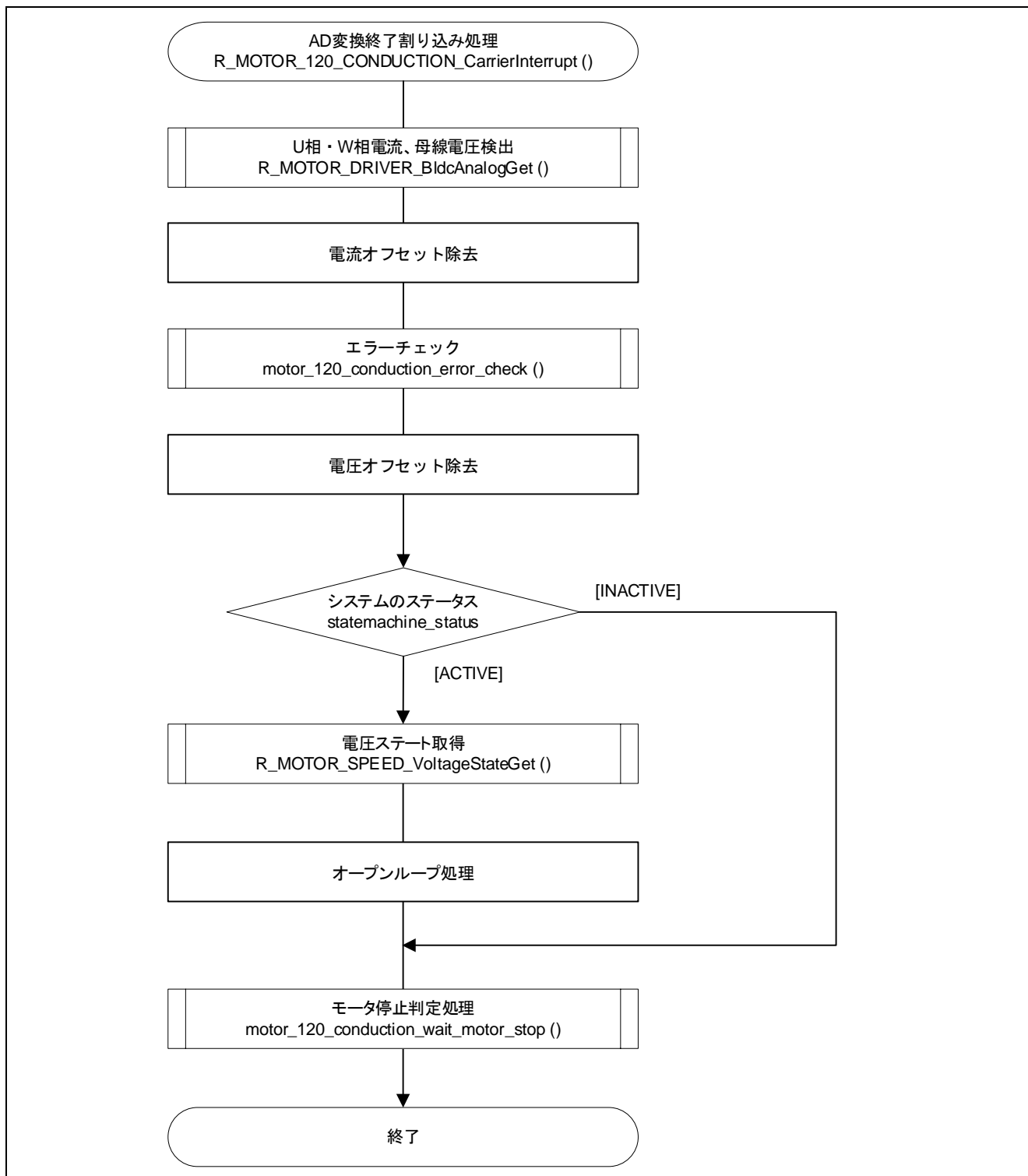


図 5-9 キャリア割り込み処理フローチャート（センサレス時）

(b) 速度制御用割り込み処理

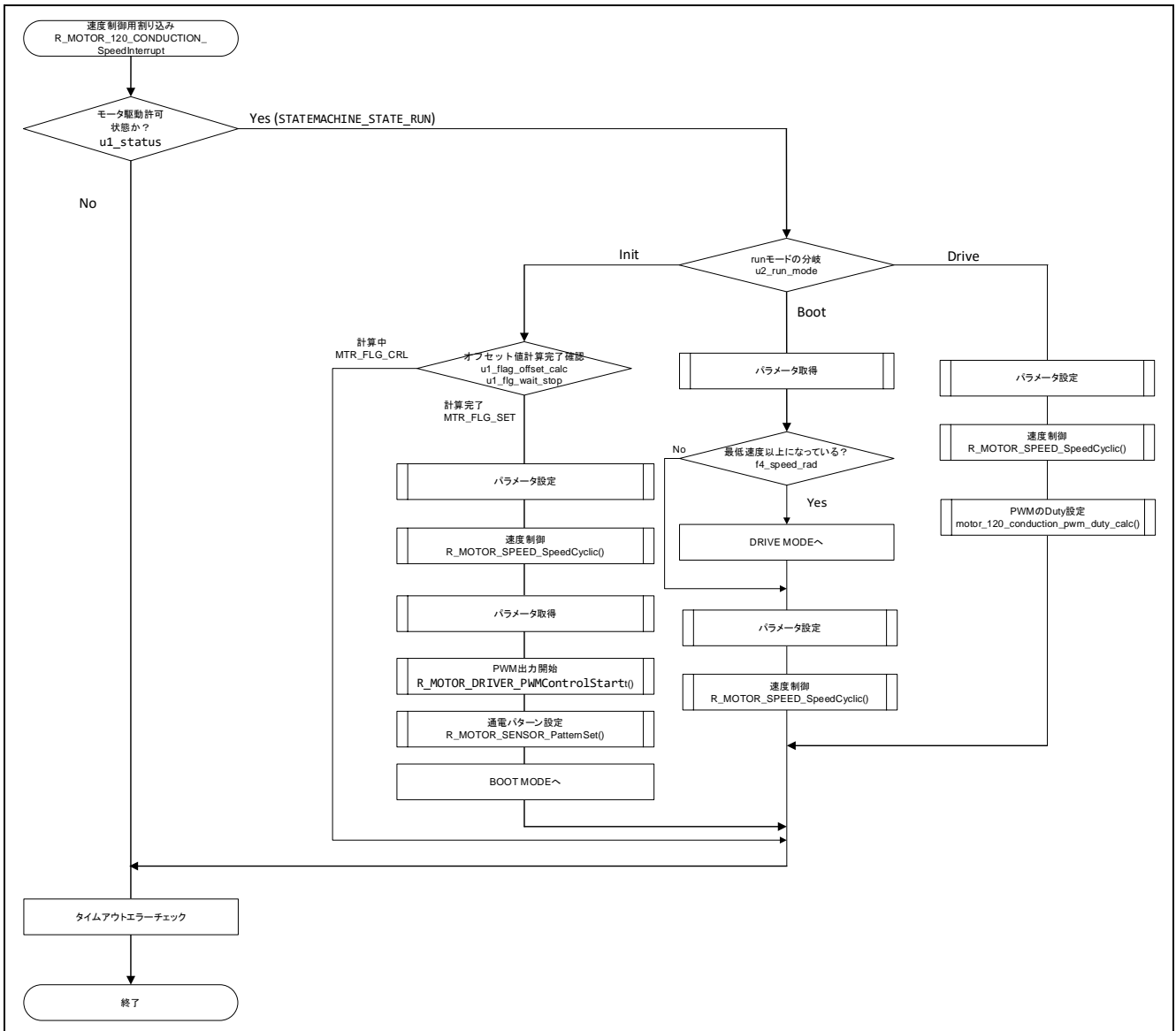


図 5-10 速度制御用割り込みフローチャート(ホールセンサ使用時)

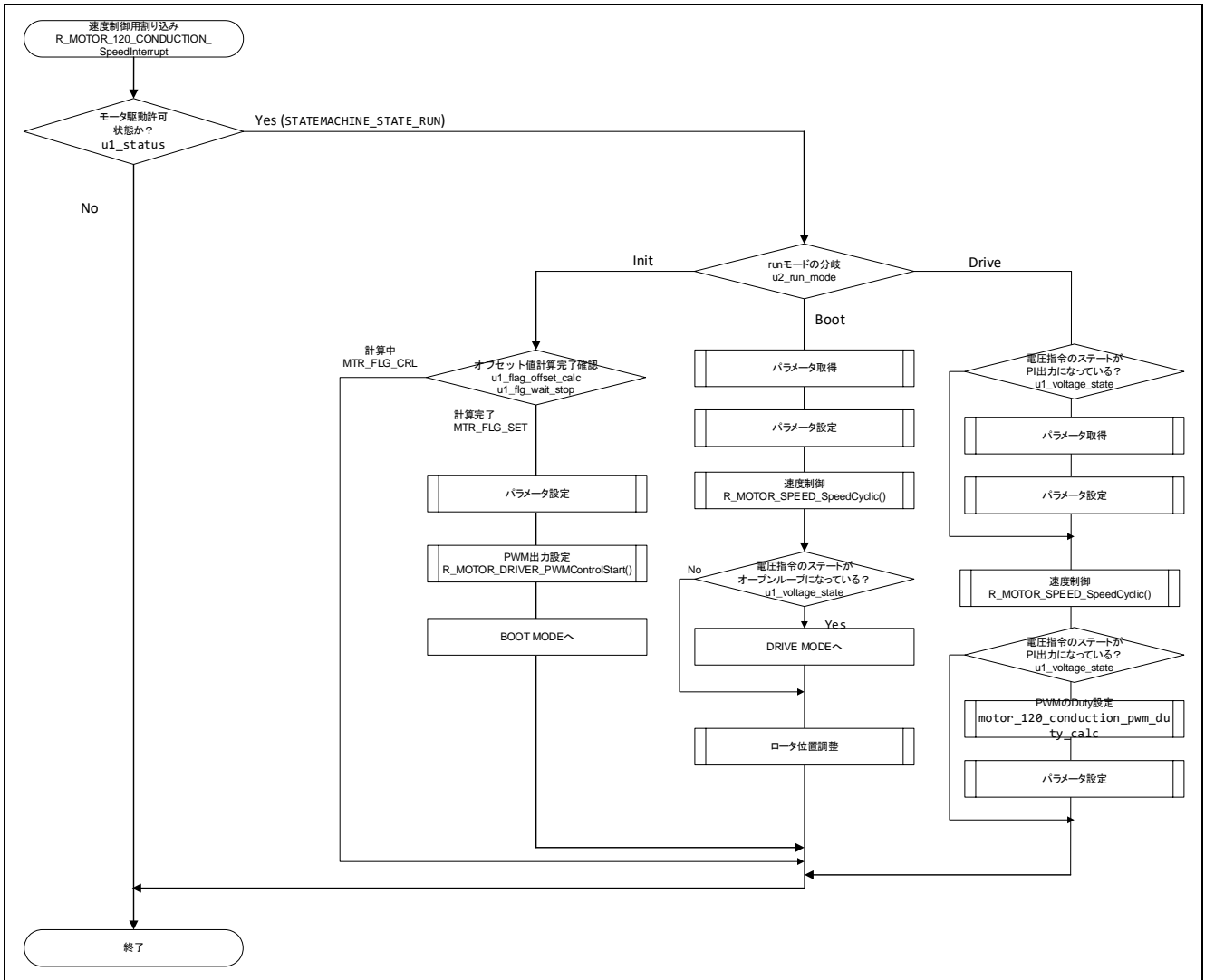


図 5-11 速度制御用割り込みフローチャート(センサレス時)

(c) 過電流検出割り込み処理

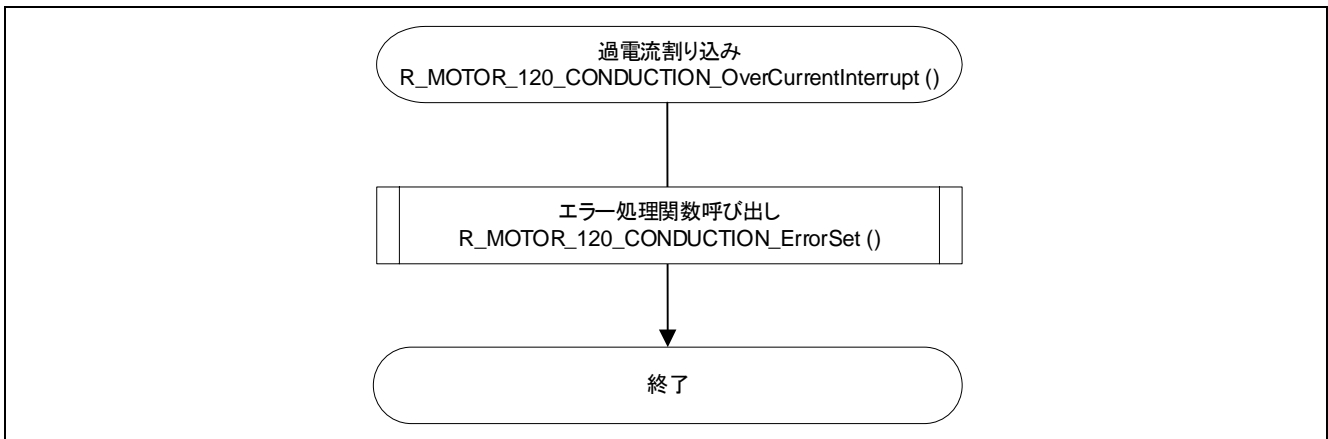


図 5-12 過電流検出割り込み処理フローチャート

(d) ホール信号割り込み処理

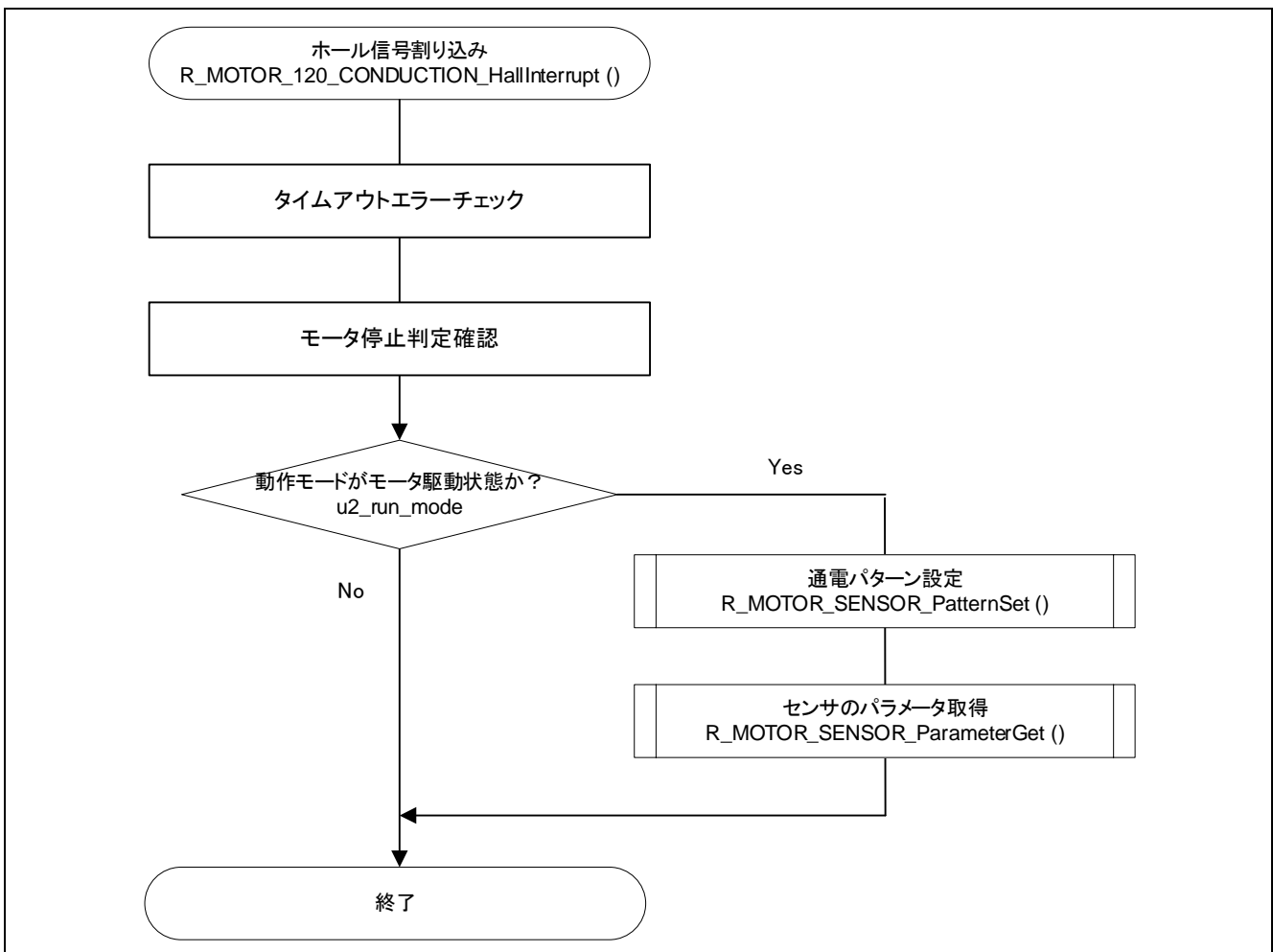


図 5-13 ホールエッジ割り込み処理フローチャート

## 5.2.6 API

マネージャモジュールの API 一覧を表 5-9 に示します。

表 5-9 API 一覧

API	説明
R_MOTOR_120_CONDUCTION_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_120_CONDUCTION_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_120_CONDUCTION_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_120_CONDUCTION_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_120_CONDUCTION_MotorStart	モータ駆動状態にします。
R_MOTOR_120_CONDUCTION_MotorStop	モータ停止状態にします。
R_MOTOR_120_CONDUCTION_MotorReset	システムのエラー状態を解除します。
R_MOTOR_120_CONDUCTION_ErrorSet	システムにエラー状態を設定します。
R_MOTOR_120_CONDUCTION_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_120_CONDUCTION_StatusGet	ステートマシンの状態を取得します。
R_MOTOR_120_CONDUCTION_ErrorStatusGet	エラー状態を取得します。
R_MOTOR_120_CONDUCTION_SpeedInterrupt	速度制御を行うための割り込み処理を行います。
R_MOTOR_120_CONDUCTION_CarrierInterrupt	キャリア割り込み処理を行うための割り込み処理を行います。
R_MOTOR_120_CONDUCTION_OverCurrentInterrupt	過電流が発生した際の割り込み処理を行います。
R_MOTOR_120_CONDUCTION_HallInterrupt	ホールセンサ信号の割り込み処理を行います。



## 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 5-10 に示します。

表 5-10 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_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_CFG_MAX_SPEED_RPM	最大速度 [rpm]
	MOTOR_CFG_MIN_SPEED_RPM	最低速度 [rpm]
	MOTOR_CFG_MARGIN_SPEED	センサレス切り替えのマージン速 度 [rpm]
	MOTOR_CFG_MARGIN_MIN_SPEED	マージンを考慮した最低速度 [rpm]

表 5-11 コンフィグレーション情報初期値一覧

マクロ名	RX13T	RX23T	RX24T, RX24U
MOTOR_MCU_CFG_PWM_TIMER_FREQ	32	40	80
MOTOR_MCU_CFG_CARRIER_FREQ	16	20	
MOTOR_MCU_CFG_AD_FREQ	32	40	40
MOTOR_MCU_CFG_AD_SAMPLING_CYCLE	47	47	47
MOTOR_MCU_CFG_AD12BIT_DATA	4095.0f		
MOTOR_MCU_CFG_ADC_OFFSET	0x7FF		
MOTOR_CFG_MAX_SPEED_RPM	2650		
MOTOR_CFG_MIN_SPEED_RPM	ホールセンサ使用時 : 550 センサレス時 : 1000		
MOTOR_CFG_MARGIN_SPEED	50		
MOTOR_CFG_MARGIN_MIN_SPEED	MOTOR_CFG_MIN_SPEED_RPM - MOTOR_CFG_MARGIN_SPEED		

## 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 5-12 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\_120\_conduction)を定義します。

表 5-12 構造体・変数一覧

構造体	変数	説明
st_120_conduction_control_t  マネージャモジュール用 構造体	u1_mode_system	システムモード
	u2_run_mode	動作モード 0：初期化 1：始動準備 2：モータ駆動
	u2_error_status	エラーステータス
	f4_vdc_ad	母線電圧 [V]
	f4_v_ref	電圧指令値 [V]
	f4_start_refv	始動時の電圧指令値 [V]
	f4_vu_ad	U 相電圧 [V]
	f4_vv_ad	V 相電圧 [V]
	f4_vw_ad	W 相電圧 [V]
	f4_vn_ad	中点電圧 [V]
	f4_offset_vu	U 相電圧オフセット [V]
	f4_offset_vv	V 相電圧オフセット [V]
	f4_offset_vw	W 相電圧オフセット [V]
	f4_offset_off_vu	インバータ停止時の U 相電圧 [V]
	f4_offset_off_vv	インバータ停止時の V 相電圧 [V]
	f4_offset_off_vw	インバータ停止時の W 相電圧 [V]
	f4_sum_vu_ad	U 相電圧合計値 [V]
	f4_sum_vv_ad	V 相電圧合計値 [V]
	f4_sum_vw_ad	W 相電圧合計値 [V]
	u1_flag_draw_in	初期位置設定用フラグ
	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_offset_iu	U 相電流オフセット [A]
	f4_offset_iv	V 相電流オフセット [A]
	f4_offset_iw	W 相電流オフセット [A]
	f4_sum_iu_ad	U 相電流合計値 [V]
	f4_sum_iv_ad	V 相電流合計値 [V]
	f4_sum_iw_ad	W 相電流合計値 [V]
	f4_inv_offset_calc	オフセット計算時間の逆数
	u2_offset_calc_time	オフセット計算時間 [s]
	u1_flag_offset_calc	オフセット計算フラグ
	u2_cnt_adjust	オフセット時間カウンタ
	u1_direction	回転方向
	u2_cnt_timeout	タイムアウトのカウント値
	u2_hall_timer_cnt	ホール用のタイマカウント値
	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]

構造体	変数	説明
st_120_conduction_control_t マネージャモジュール用 構造体	f4_overspeed_limit_rad	過速度制限値 [rad/s]
	u2_hall_timeout_cnt_limit	ホールのタイムアウトカウント制限値
	u2_pwm_duty	PWM の Duty 値
	f4_speed_rad	速度 [rad/s]
	f4_rpm_rad	単位換算用変数(rpm から rad/s)
	u1_flg_wait_stop	回転停止検知用フラグ
	u2_cnt_wait_stop	回転停止検知用カウント値
	u1_hall_wait_cnt	ホールの停止判定用カウント値
	st_speed_output	速度モジュールの出力用構造体
	st_sensor_output	センサモジュールの出力用構造体
	st_motor	モータパラメータ構造体
	*p_st_driver	ドライバモジュール用構造体へのポインタ
	*p_st_speed	速度モジュール構造体ポインタ
	*p_st_sensor	センサモジュール構造体ポインタ
st_120_conduction_cfg マネージャモジュール制御 パラメータ設定用構造体	u1_hall_wait_cnt	ホールの停止判定用カウント値
	u2_offset_calc_time	オフセット計算時間 [s]
	u2_mtr_p	モータの極対数
	u1_direction	回転方向
	f4_rpm_rad	単位換算用変数(rpm から rad/s)

## 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 5-13 に示します。

表 5-13 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_120_con duction_api.h	MOTOR_MODE_INIT	0x00	初期化モード。
	MOTOR_MODE_BOOT	0x01	始動準備モード。
	MOTOR_MODE_DRIVE	0x02	モータ駆動モード。
	MOTOR_MODE_ANALY SIS	0x03	解析モード。(使用不可)
	MOTOR_MODE_TUNE	0x04	自動調整モード。(使用不可)
	MOTOR_ERROR_NONE	0x0000	エラーの検知無し。
	MOTOR_ERROR_OVER _CURRENT_HW	0x0001	エラーステータス HW 過電流エラー状態。
	MOTOR_ERROR_OVER _VOLTAGE	0x0002	エラーステータス。過電圧エラー状態。
	MOTOR_ERROR_OVER _SPEED	0x0004	エラーステータス。過速度エラー状態。
	MOTOR_ERROR_HALL_ TIMEOUT	0x0008	エラーステータス。ホールの割り込み検知 タイムアウトエラー状態。
	MOTOR_ERROR_BEMF _TIMEOUT	0x0010	エラーステータス。逆起電力検知タイムア ウトエラー状態。
	MOTOR_ERROR_HALL_ PATTERN	0x0020	エラーステータス。ホールの通電パターン エラー状態。
	MOTOR_ERROR_BEMF _PATTERN	0x0040	エラーステータス。センサレスの通電パ ターンエラー状態。
	MOTOR_ERROR_LOW_ VOLTAGE	0x0080	エラーステータス。低電圧エラー状態。
	MOTOR_ERROR_OVER _CURRENT_SW	0x0100	エラーステータス。SW 過電流エラー状 態。
	MOTOR_ERROR_UNKN OWN	0xffff	エラーステータス。エラーコード不明のエ ラー状態。
	MOTOR_SENSOR_LESS	0x01	センサレス駆動を行います。
	MOTOR_SENSOR_HALL	0x02	ホールセンサを使用した駆動を行います。
	MOTOR_SENSOR_ENC D	0x04	エンコーダを使用した駆動を行います。 (使用不可)
	MOTOR_SENSOR_RES O	0x08	レゾルバセンサを使用した駆動を行いま す。(使用不可)
	MOTOR_METHOD_FOC	0x00	ベクトル制御を使用した駆動を行います。 (使用不可)
	MOTOR_METHOD_180	0x01	180 度通電制御を使用した駆動を行いま す。(使用不可)
	MOTOR_METHOD_WID E	0x02	広角制御を使用した駆動を行います。(使 用不可)
	MOTOR_METHOD_120	0x03	120 度通電制御を使用した駆動を行いま す。
	MOTOR_CONTROL_CU RRENT	0x01	電流制御を行います。(使用不可)
	MOTOR_CONTROL_SP EED	0x02	速度制御を行います。

ファイル名	マクロ名	定義値	備考
r_motor_120_con duction_api.h	MOTOR_CONTROL_PO SITION	0x04	位置制御を行います。(使用不可)
	MOTOR_CONTROL_TO RQUE	0x08	トルク制御を行います。(使用不可)
	MOTOR_CONTROL_VO LTAGE	0x10	電圧制御を行います。(使用不可)
r_motor_120_con duction_statemac hine.h	STATEMACHINE_STATE _STOP	0x00	停止状態
	STATEMACHINE_STATE _RUN	0x01	動作状態
	STATEMACHINE_STATE _ERROR	0x02	エラー状態
	STATEMACHINE_STATE _SIZE	3	ステートサイズ
	STATEMACHINE_EVEN T_STOP	0x00	停止イベント
	STATEMACHINE_EVEN T_RUN	0x01	動作イベント
	STATEMACHINE_EVEN T_ERROR	0x02	エラーイベント
	STATEMACHINE_EVEN T_RESET	0x03	リセットイベント
	STATEMACHINE_EVEN T_SIZE	4	イベントサイズ

## 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 5-14 に示します。

表 5-14 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_DEADTIME	2.0f	デットタイム [us]
	INVERTER_CFG_CURRENT_RANGE	25.0f	電流のスケールリング
	INVERTER_CFG_VDC_RANGE	111.0f	電圧のスケールリング
	INVERTER_CFG_INPUT_V	24.0f	入力電圧 [V]
	INVERTER_CFG_CURRENT_LIMIT	10.0f	インバータボードの過電流の制限値 [A]
	INVERTER_CFG_OVERVOLTAGE_LIMIT	28.0f	過電圧制限 [V]
	INVERTER_CFG_UNDERVOLTAGE_LIMIT	14.0f	低電圧制限 [V]
	INVERTER_CFG_ADC_REF_VOLTAGE	5.0f	MCU のアナログ電源電圧 [V]
r_motor_targetmotor_cfg.h	MOTOR_CFG_POLE_PAIRS	2	極対数
	MOTOR_CFG_MAGNETIC_FLUX	0.02159f	磁束 [wb]
	MOTOR_CFG_RESISTANCE	6.447f	抵抗 [ohm]
	MOTOR_CFG_D_INDUCTANCE	0.0045f	d 軸のインダクタンス [H]
	MOTOR_CFG_Q_INDUCTANCE	0.0045f	q 軸のインダクタンス [H]
	MOTOR_CFG_NOMINAL_CURRENT_RMS	0.42f	定格電流 [A]

5.2.11 始動シーケンス管理

(a) ホールセンサ

ホールセンサを使用した 120 度通電制御では、ホールセンサ信号により回転子位置が判別出来るため、始動時の通電パターンは一意に決まります。

但し、速度制御を行うためには 5.4.9 に示した様に最低でも最初の  $2\pi$  分の時間データを計測する必要があります。そのため、サンプルプログラムでは始動方法として一定電圧によるオープンループでの始動を行い時間データが取得出来る条件を待って速度制御へ遷移する方法を取っています。

図 5-14 ではサンプルプログラムでの始動方法を示しています。“MTR\_MODE\_BOOT”では、`g_st_speed.f4_v_ref` で与えられた一定電圧によるオープンループ始動を行っています。“MTR\_MODE\_DRIVE”への遷移条件はその時点での計測回転数が規定最低回転数（550rpm）に到達することです。

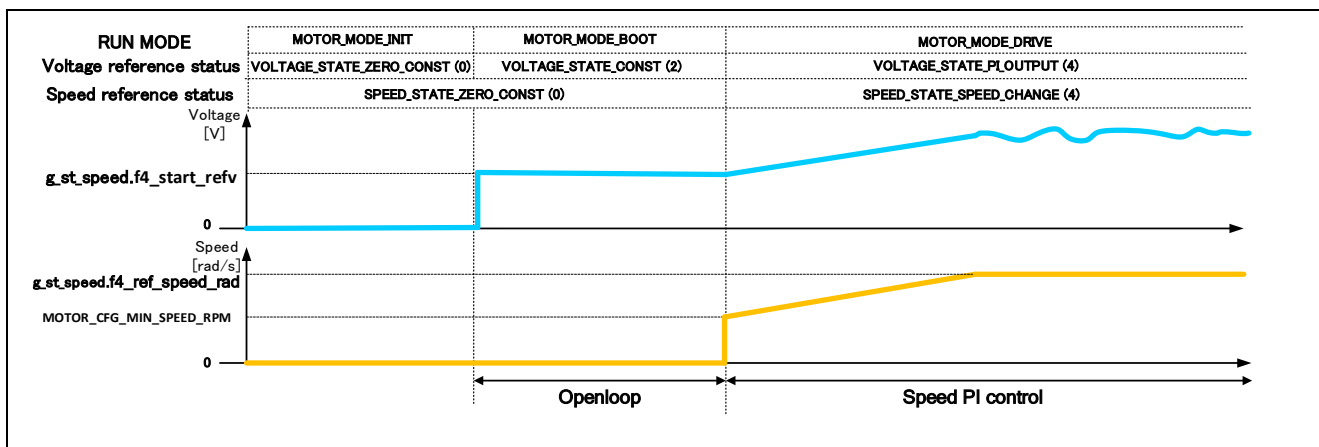


図 5-14 ホールセンサ使用時の 120 度通電制御の始動方法例

(b) センサレス

センサレス 120 度通電制御は、永久磁石（回転子）の磁束の変化による誘起電圧を利用し、60 度毎の磁極の位置を推定します。しかし、誘起電圧は回転することで発生するため、始動時は磁極の位置を推定することができず、磁極位置の推定に十分な回転速度が必要です。

そのため、始動方法として、永久磁石の位置にかかわらず、強制的に通電パターンを変化させることで回転磁界を発生させ、同期速度に引き込む方法があります。

図 5-15 ではサンプルプログラムでの始動方法を示しています。“MOTOR\_MODE\_BOOT” では、磁極位置の引き込みを行い、始動時の過電流を防いでいます。

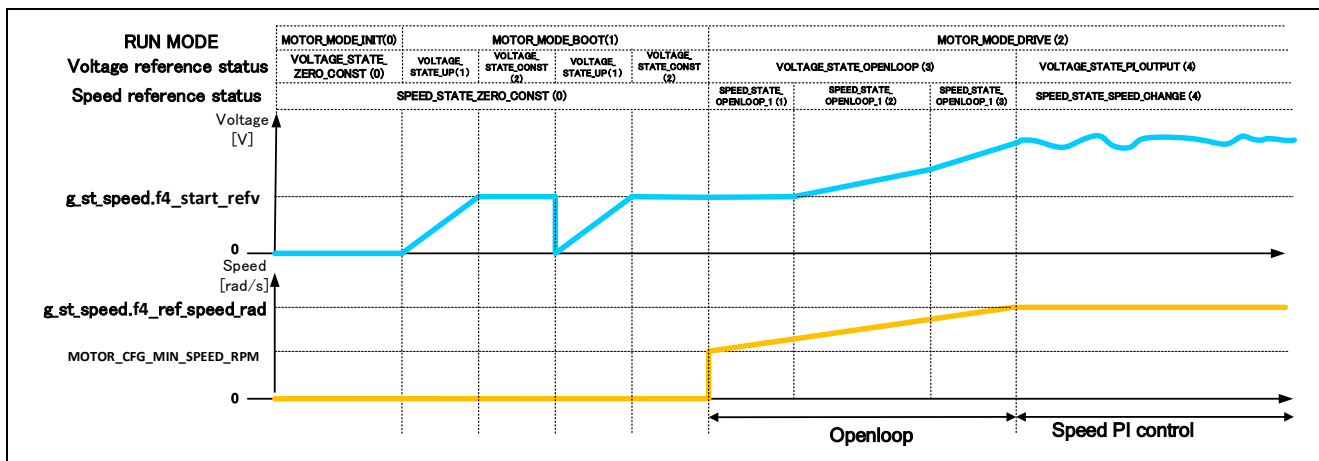


図 5-15 センサレス 120 度通電制御の始動方法例



5.2.12 PWM による電圧制御

出力電圧の制御には PWM 制御を使用しています。PWM 制御とは、図 5-16 のように、パルスのデューティを変化させることで平均電圧を調整していく制御方式です。

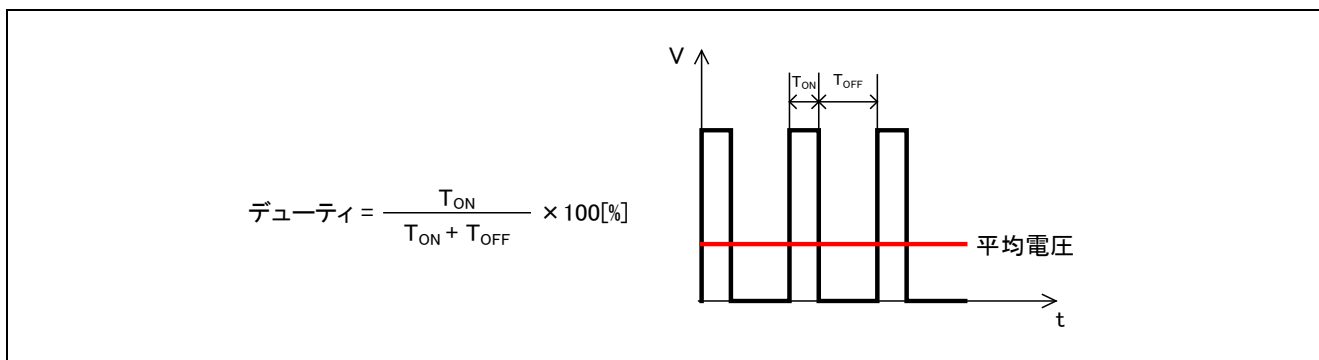


図 5-16 PWM 制御

ここで、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

この変調率を、PWM デューティを決めるレジスタの設定値に反映させます。

また、サンプルプログラムでは、前半 60 度チョッピングを採用し、出力電圧及び速度の制御を行っています。図 5-17 に、非相補前半 60 度チョッピング時のモータ制御信号出力波形例を示します。図 5-18 に、相補前半 60 度チョッピング時のモータ制御信号出力波形例を示します。

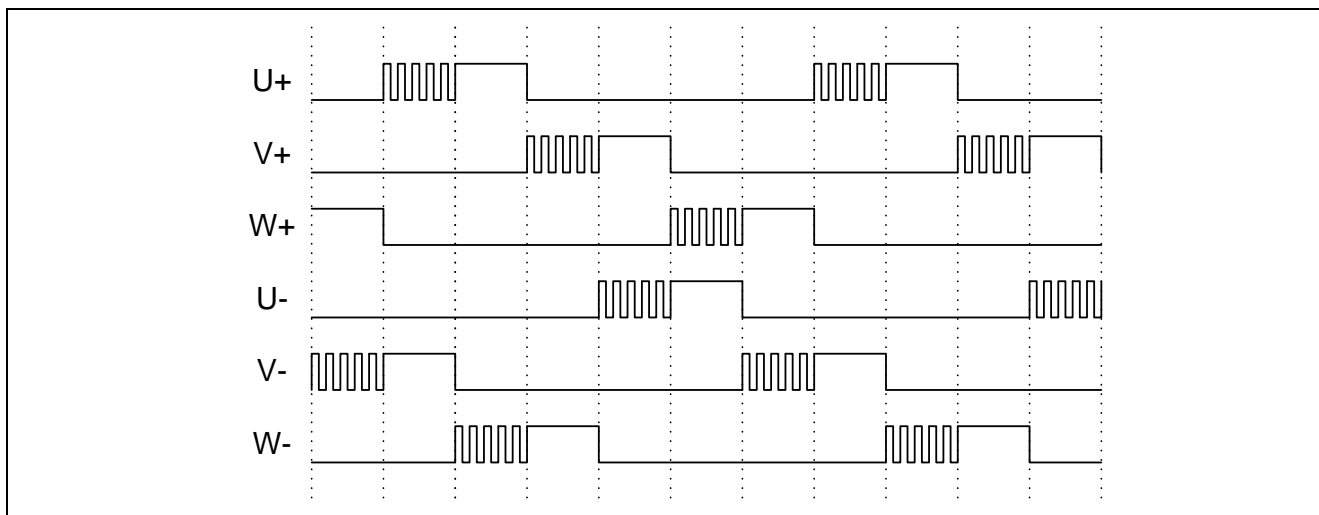


図 5-17 非相補前半 60 度チョッピング

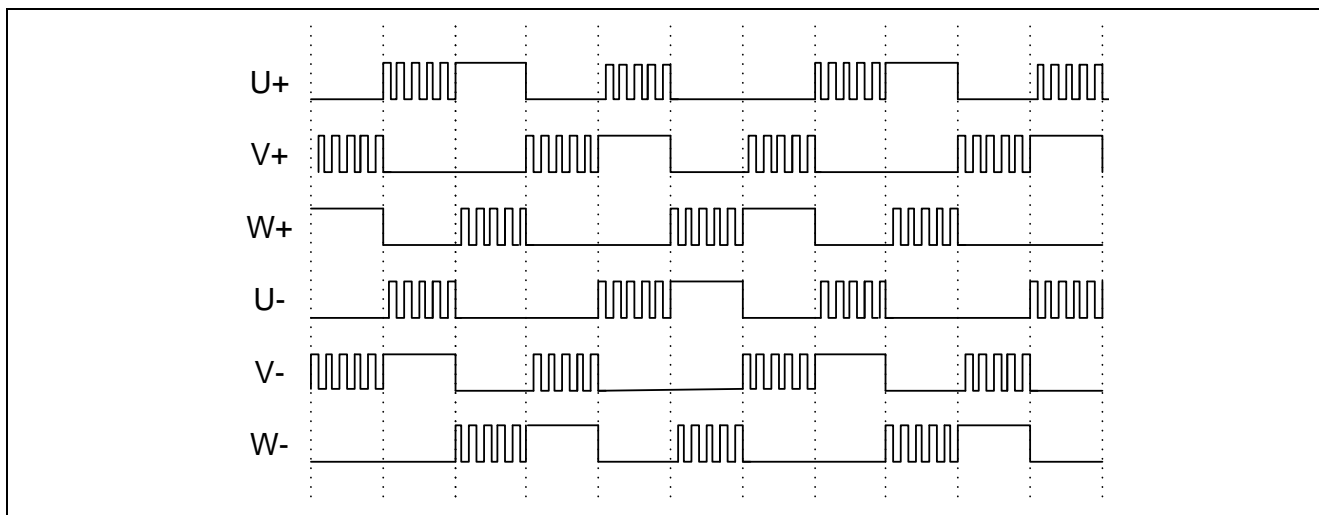


図 5-18 相補前半 60 度チョッピング

### 5.3 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、電圧指令値を出力します。

#### 5.3.1 機能

速度制御モジュールの機能一覧を表 5-15 に示します。

表 5-15 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、電圧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。

#### 5.3.2 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 5-19 に示します。

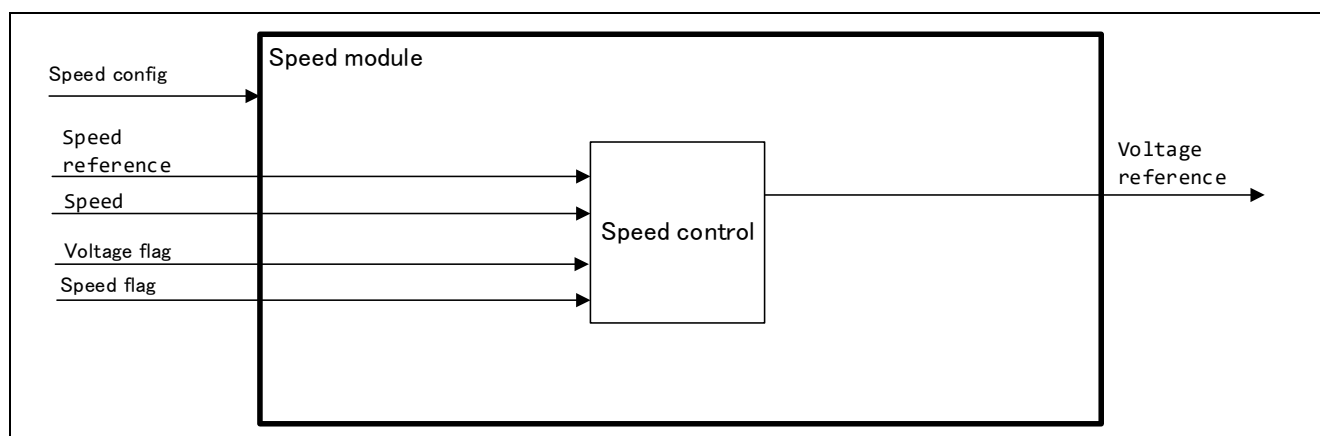


図 5-19 速度モジュール構成図

5.3.3 フローチャート

速度制御のフローチャートを図 5-20 に示します。

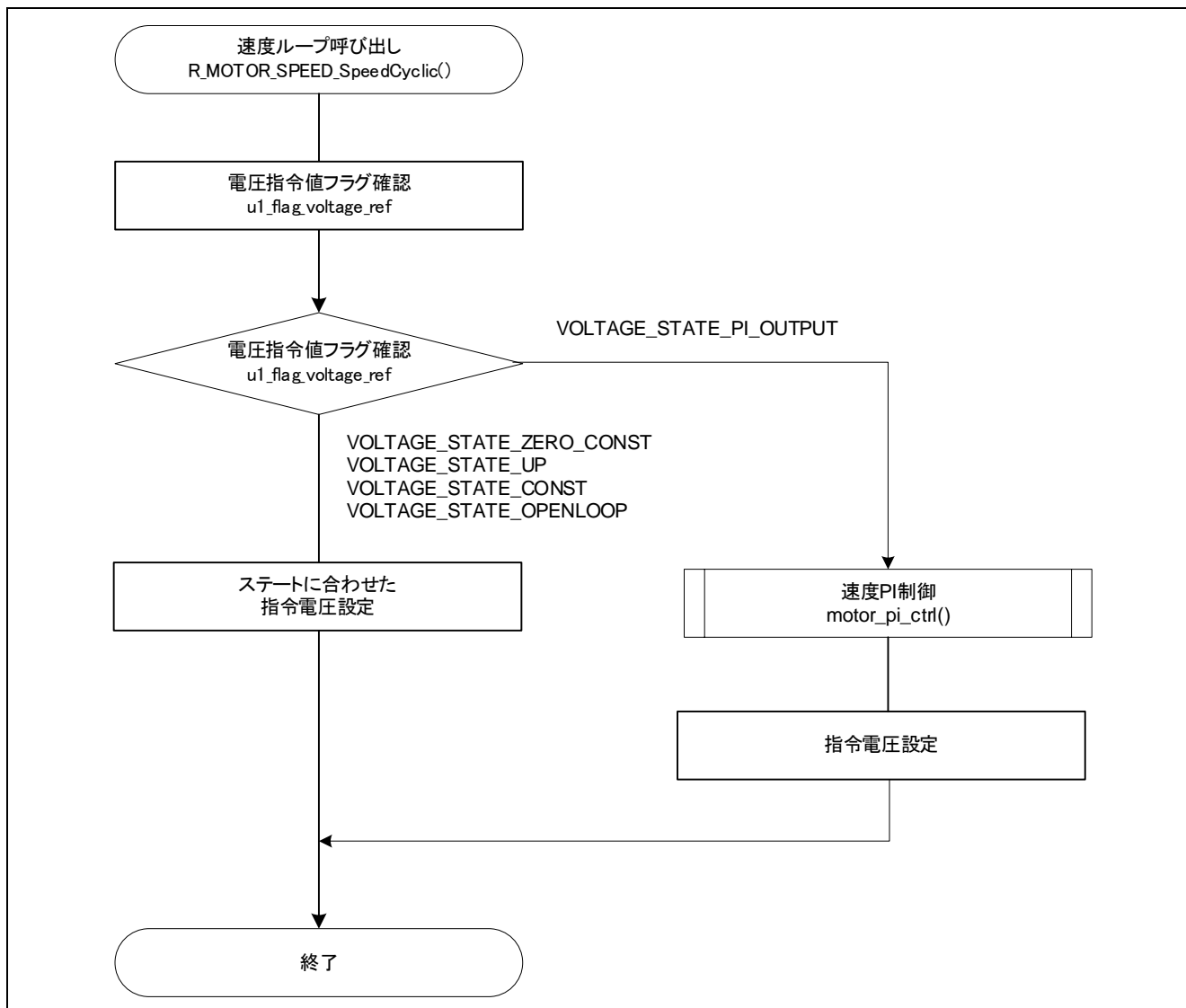


図 5-20 速度制御フローチャート

## 5.3.4 API

速度制御モジュールの API 一覧を表 5-16 に示します。

表 5-16 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_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_SpeedCyclic	速度制御を行います。
R_MOTOR_SPEED_RefSpeedSet	速度指令値を設定します。
R_MOTOR_SPEED_RefSpeedRpmToRad	速度指令値を RPM から RAD に変換します
R_MOTOR_SPEED_VoltageStateSet	出力電圧を決定するステータを設定します。
R_MOTOR_SPEED_VoltageStateGet	出力電圧を決定するステータを取得します。(センサレスのみ)
R_MOTOR_SPEED_DrawInFlagSet	始動時のモータ引き込みのフラグを設定します。(センサレスのみ)
R_MOTOR_SPEED_DrawInFlagGet	始動時のモータ引き込みのフラグを取得します。(センサレスのみ)

## 5.3.5 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 5-17 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。各 MCU の初期値は表 5-18 に示します。

表 5-17 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_SPEED_PI_DECIMATION	PI の間引き回数設定
	SPEED_CFG_SPEED_PI_KP	速度制御系の PI ゲイン比例項
	SPEED_CFG_SPEED_PI_KI	速度制御系の PI ゲイン微分項
	SPEED_CFG_SPEED_PI_I_LIMIT_V	速度制御系の PI ゲイン微分項制限値
	SPEED_CFG_SPEED_CALC_BASE	速度計算用タイマの周波数設定
	SPEED_CFG_SPEED_LPF_K	速度制御系の LPF ゲイン
	SPEED_CFG_SPEED_CHANGE_LIMIT	速度の変化量制限値
	MOTOR_CFG_MAX_OUTPUT_VOLTAGE	最大出力電圧 [V]
	MOTOR_CFG_MIN_OUTPUT_VOLTAGE	最小出力電圧 [V]
	MOTOR_CFG_START_REF_VOLTAGE	始動時の指令電圧 [V]
	MOTOR_CFG_MAX_BOOT_VOLTAGE	ブートモード時の最大電圧 [V] (センサレスのみ)
	MOTOR_CFG_SHIFT_ADJUST	位相の調整量 (センサレスのみ)

表 5-18 コンフィグレーション情報初期値一覧

マクロ名	RX13T, RX23T, RX24T, RX24U
SPEED_CFG_SPEED_PI_DECIMATION	ホール使用時 : 0 センサレス時 : 1
SPEED_CFG_SPEED_PI_KP	0.02f
SPEED_CFG_SPEED_PI_KI	ホール使用時 : 0.0005f センサレス時 : 0.004f
SPEED_CFG_SPEED_PI_I_LIMIT_V	24.0f
SPEED_CFG_SPEED_CALC_BASE	MTR_TWOPi × 5000000
SPEED_CFG_SPEED_LPF_K	1.0f
SPEED_CFG_SPEED_CHANGE_LIMIT	ホール使用時 : 0.2f センサレス時 : 0.2f * MTR_RPM2RAD
MOTOR_CFG_MAX_OUTPUT_VOLTAGE	20.0f
MOTOR_CFG_MIN_OUTPUT_VOLTAGE	ホール使用時 : 3.0f センサレス時 : 5.0f
MOTOR_CFG_START_REF_VOLTAGE	5.8f
MOTOR_CFG_MAX_BOOT_VOLTAGE	8.0f (センサレスのみ)
MOTOR_CFG_SHIFT_ADJUST	0 (センサレスのみ)

## 5.3.6 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 5-19 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 5-19 構造体・変数一覧 1

構造体	変数	説明
st_speed_control_t	u2_run_mode	動作モード
速度モジュール用 構造体	f4_v_ref	電圧指令値 [V]
	u1_cnt_speed_pi	速度 PI 用カウンタ値
	u1_flag_speed_ref	速度指令値用のフラグ
	u1_flag_voltage_ref	電圧指令値用のフラグ
	f4_ref_speed_rad	速度指令値 [rad/s]
	f4_ref_speed_rad_ctrl	制御用速度指令値 [rad/s]
	f4_speed_rad	速度 [rad/s]
	f4_kp_speed	PI 制御の比例項ゲイン
	f4_ki_speed	PI 制御の微分項ゲイン
	f4_limit_speed_change	速度の変化量制限
	f4_ilim_v	最大電圧値
	f4_start_refv	始動時の指令電圧 [V]
	f4_rpm_rad	単位換算用変数(rpm から rad/s)
	f4_boot_ref_v	ブートモード時の指令電圧 [V] (センサレスのみ)
	u2_v_up_time	電圧上昇時間 [s] (センサレスのみ)
	f4_v_up_step	電圧上昇幅 [V] (センサレスのみ)
	u2_v_const_time	電圧一定時間 [s] (センサレスのみ)
	u2_cnt_adj_v	電圧上昇時間カウンタ (センサレスのみ)
	u1_flag_draw_in	初期位置設定回数フラグ (センサレスのみ)
	s2_ol_start_rad	オープンループ始動時の速度 [rad/s] (センサレスのみ)
	s2_ol_mode1_change_rad	オープンループモード 1 への移行速度 [rad/s] (センサレスのみ)
	s2_ol_mode2_change_rad	オープンループモード 2 への移行速度 [rad/s] (センサレスのみ)
	f4_ol_start_refv	オープンループ始動時の指令電圧 [V] (センサレスのみ)
	f4_ol_mode1_rate_rad	オープンループモード 1*1 の速度変化量設定 [rad/s] (センサレスのみ)
	f4_ol_mode2_rate_refv	オープンループモード 2*1 の電圧変化量設定 [V] (センサレスのみ)
	f4_ol_mode2_rate_rad	オープンループモード 2*1 の速度変化量設定 [rad/s] (センサレスのみ)
	f4_ol_mode3_rate_refv	オープンループモード 3*1 の電圧変化量設定 [V] (センサレスのみ)
st_pi_ctrl_t	PI 制御用構造体	
st_motor_parameter_t	モータパラメータ用構造体	

【注】 1. 詳細は 5.2.11 始動シーケンス管理を参照ください。

表 5-20 構造体・変数一覧 2

構造体	変数	説明
st_speed_config_t 速度モジュール制御パラメータ設定用構造体	f4_ref_speed_rad	指令速度 [rad/s]
	f4_kp_speed	PI 制御の比例項ゲイン
	f4_ki_speed	PI 制御の微分項ゲイン
	f4_limit_speed_change	速度の変化量制限
	f4_start_refv	始動時の指令電圧 [V]
	f4_rpm_rad	単位換算用変数(rpm から rad/s)
	f4_boot_ref_v	ブートモード時の指令電圧 [V] (センサレスのみ)
	u2_v_up_time	電圧上昇時間 [s] (センサレスのみ)
	u2_v_const_time	電圧上昇幅 [V] (センサレスのみ)
	s2_ol_start_rad	オープンループ始動時の速度 [rad/s] (センサレスのみ)
	s2_ol_mode1_change_rad	オープンループモード 1 への移行速度 [rad/s] (センサレスのみ)
	s2_ol_mode2_change_rad	オープンループモード 2 への移行速度 [rad/s] (センサレスのみ)
	f4_ol_start_refv	オープンループ始動時の指令電圧 [V] (センサレスのみ)
	f4_ol_mode1_rate_rad	オープンループモード 1*1 の速度変化量設定 [rad/s] (センサレスのみ)
	f4_ol_mode2_rate_refv	オープンループモード 2*1 の電圧変化量設定 [V] (センサレスのみ)
f4_ol_mode2_rate_rad	オープンループモード 2*1 の速度変化量設定 [rad/s] (センサレスのみ)	
f4_ol_mode3_rate_refv	オープンループモード 3*1 の電圧変化量設定 [V] (センサレスのみ)	
f4_ol_mode3_max_refv	オープンループモード 3*1 の最大指令電圧 [V] (センサレスのみ)	
st_speed_input_t 速度モジュール入力用構造体	u2_run_mode	動作モード
	f4_speed_rad	速度 [rad/s]
st_speed_output_t 速度モジュール出力用構造体	f4_v_ref	指令電圧 [V]

【注】 1. 詳細は 5.2.11 始動シーケンス管理を参照ください。



## 5.3.7 マクロ定義

速度制御モジュールのマクロ一覧を表 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	備考
r_motor_speed_api.h	VOLTAGE_STATE_ZERO_CONST	0	電圧指令値をゼロ固定にします。
	VOLTAGE_STATE_UP	1	電圧指令値を上昇させます。
	VOLTAGE_STATE_CONST	2	電圧指令値を固定にします。
	VOLTAGE_STATE_OPENLOOP	3	電圧指令値をオープンループの固定値にします。
	VOLTAGE_STATE_PI_OUTPUT	4	電圧指令値を PI 制御の出力値にします。
	SPEED_STATE_ZERO_CONST	0	速度指令値をゼロ固定にします。
	SPEED_STATE_OPENLOOP_1	1	速度指令値をオープンループモード 1* <sup>1</sup> にします。
	SPEED_STATE_OPENLOOP_2	2	速度指令値をオープンループモード 2* <sup>1</sup> にします。
	SPEED_STATE_OPENLOOP_3	3	速度指令値をオープンループモード 3* <sup>1</sup> にします。
	SPEED_STATE_SPEED_CHANGE	4	速度指令値をユーザ入力指定値にします。

【注】 1. 詳細は 5.2.11 始動シーケンス管理を参照ください。

## 5.3.8 パラメータ調整・設定

## (a) 速度制御系のゲイン調整

本アプリケーションノート対象ソフトでの速度制御は、PI 制御によって行います。下記の速度 PI 制御によって電圧指令値を得ます。Kp 及び Ki の設定に関しては PI 制御の専門書を参照してください。

$$v^* = (K_{P\omega} + \frac{K_{I\omega}}{s})(\omega^* - \omega)$$

$v^*$ : 電圧指令値       $\omega^*$ : 速度指令値       $\omega$ : 回転速度

$K_{P\omega}$ : 速度PI比例ゲイン       $K_{I\omega}$ : 速度PI積分ゲイン       $s$ : ラプラス演算子

## (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\_config\_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 5.3.5 を参照ください。

### 5.4 センサモジュール（ホールセンサ）

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。

#### 5.4.1 機能

センサモジュールの機能一覧を表 5-22 に示します。

表 5-22 センサモジュールの機能一覧

機能	説明
速度情報の取得	モータの回転速度を取得します。

#### 5.4.2 モジュール構成図

センサモジュール構成図を図 5-21 に示します。

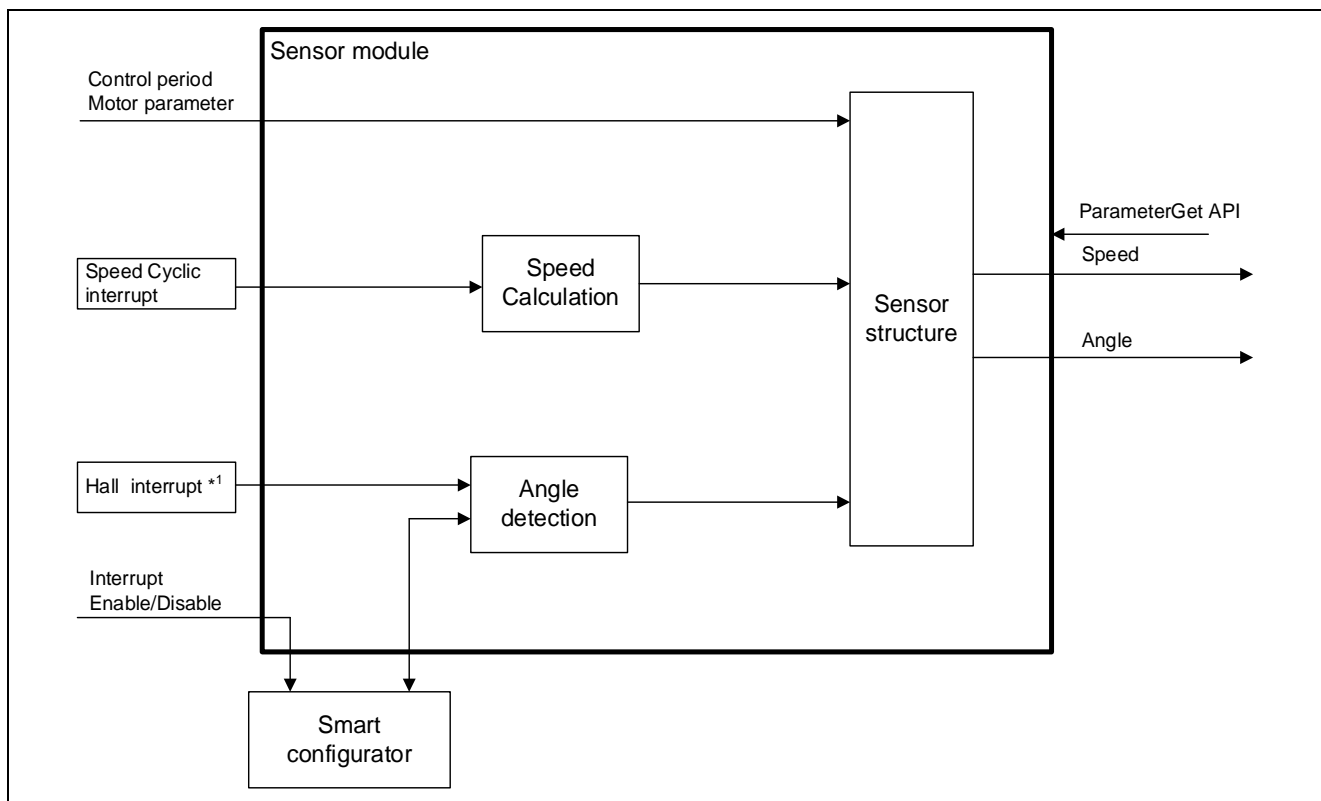


図 5-21 センサモジュール構成図

【注】 1. センサレス時は、Carrier Interrupt

#### 5.4.3 フローチャート

ホールセンサの割り込み処理に関するフローチャートは 5.2.5(d)を参照ください。

## 5.4.4 API

センサモジュールの API 一覧を表 5-23 に示します。

表 5-23 API 一覧

API	説明
R_MOTOR_SENSOR_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_Close	センサモジュールをリセット状態にします。
R_MOTOR_SENSOR_Reset	モジュールの初期化します。
R_MOTOR_SENSOR_ParameterSet	センサモジュールで使用するパラメータを設定します。
R_MOTOR_SENSOR_ParameterGet	センサモジュールのパラメータを取得します。
R_MOTOR_SENSOR_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_PatternSet	通電パターンの設定を行います。(ホール使用時のみ)
R_MOTOR_SENSOR_DirectionSet	モータの回転方向を設定します。
R_MOTOR_SENSOR_DirectionGet	モータの回転方向を取得します。
R_MOTOR_SENSOR_TimerCountGet	モータ駆動を確認するタイマカウンタの取得
R_MOTOR_SENSOR_TimerCountUp	モータ駆動を確認するタイマカウンタのカウンタアップ
R_MOTOR_SENSOR_TimerCountClear	モータ駆動を確認するタイマカウンタのクリア
R_MOTOR_SENSOR_StartOpenLoop	オープンループ駆動をスタートします。(センサレスのみ)
R_MOTOR_SENSOR_AngleShiftSet	通電パターンを切り替える位相シフト情報を設定します。(センサレスのみ)
R_MOTOR_SENSOR_CheckPattern	通電パターンのエラーチェックを行います。
R_MOTOR_SENSOR_ShiftAngle	通電パターンの設定を行い、パターン切り替えを行います。(センサレスのみ)
R_MOTOR_SENSOR_OutputPatternSet	通電パターンの設定を行い、パターン切り替えを行います。(センサレスのみ)

## 5.4.5 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 5-24 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。各 MCU の初期値は表 5-25 に示します。

表 5-24 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_TIMEOUT_CNT	ホールのタイムアウトカウント時間
	SENSOR_CFG_HALL_STOP_WAIT_CNT	停止判定カウント値
	SENSOR_CFG_HALL_WAIT_SPEED_CALC	ホール割り込みまでの速度計算待ち時間

表 5-25 コンフィグレーション情報初期値一覧

マクロ名	RX13T, RX23T, RX24T, RX24U
SENSOR_CFG_TIMEOUT_CNT	ホール使用時 : 4000 センサレス時 : 2000
SENSOR_CFG_HALL_STOP_WAIT_CNT	4000
SENSOR_CFG_HALL_WAIT_SPEED_CALC	12

5.4.6 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 5-26 に示します。

表 5-26 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_mode_system	システムモード
	u2_run_mode	動作モード
	u2_error_status	エラーステータス
	u1_v_pattern	通電パターン
	u2_pre_hall_timer_cnt	ホール用タイマの前回値
	u2_hall_timer_cnt	ホール用タイマのカウント値
	s4_timer_cnt_ave	速度計測タイマカウントの平均値
	u2_timer_cnt_buf[6]	速度計測用バッファ
	u2_timer_cnt_num	速度計測用カウント番号
	u1_hall_signal	ホール信号
	u1_hall_wait_cnt	ホールの待ち時間カウント値
	u1_direction	回転方向
	u2_pwm_duty	PWM の Duty 値
	f4_speed_rad	速度 [rad/s]
	f4_speed_lpf_k	速度用 LPF ゲイン
	u2_cnt_timeout	センサタイムアウト検出用タイマカウンタ
	u1_v_pattern_num	通電パターン番号 (センサレスのみ)
	u1_bemf_signal	逆起電力の信号値 (センサレスのみ)
	u1_pre_bemf_signal	逆起電力信号の前回値 (センサレスのみ)
	u1_flag_pattern_change	通電パターン切り替えフラグ (センサレスのみ)
	u1_v_pattern_ol[2][7]	オープンループ時の通電パターン (センサレスのみ)
	u1_ol_signal	オープンループ時の逆起電力信号値 (センサレスのみ)
	u2_ol_pattern_set	オープンループ時の通電パターン設定 (センサレスのみ)
	u2_cnt_ol_pattern_set	オープンループ時の通電パターン設定カウント値 (センサレスのみ)
	u2_bemf_timer_cnt	逆起電力のタイマカウント値 (センサレスのみ)
	u2_pre_bemf_timer_cnt	逆起電力のタイマ前回値 (センサレスのみ)
	u2_cnt_carrier	キャリア割り込みのカウント値 (センサレスのみ)
u2_pre_cnt_carrier	キャリア割り込みの前回値 (センサレスのみ)	
u2_angle_shift_cnt	位相シフト用カウント値 (センサレスのみ)	
s2_angle_shift_adjust	角度の位相調整 (センサレスのみ)	
st_sensor_output_t センサモジュール出力用構造体	f4_speed_rad	速度 [rad/s]
	u2_error_status	エラーステータス
st_sensor_input_t センサモジュール入力用構造体	u1_mode_system	システムモード
	u2_run_mode	動作モード
	u2_pwm_duty	PWM の Duty 値

構造体	変数	説明
st_sensor_cfg_t センサモジュールの パラメータ 設定用構造体	f4_speed_lpf_k	速度用 LPF ゲイン
	u1_hall_wait_cnt	ホールの待ち時間カウント値

## 5.4.7 マクロ定義

センサモジュールのマクロ一覧を表 5-27 に示します。

表 5-27 マクロ一覧

ファイル名	マクロ名	定義値	備考
r_motor_senso r.h	SENSOR_PATTERN_CW_V_U	(2)	CW 時の通電パターン (V 相から U 相)
	SENSOR_PATTERN_CW_W_U	(3)	CW 時の通電パターン (W 相から U 相)
	SENSOR_PATTERN_CW_W_V	(1)	CW 時の通電パターン (W 相から V 相)
	SENSOR_PATTERN_CW_U_V	(5)	CW 時の通電パターン (U 相から V 相)
	SENSOR_PATTERN_CW_U_W	(4)	CW 時の通電パターン (U 相から W 相)
	SENSOR_PATTERN_CW_V_W	(6)	CW 時の通電パターン (V 相から W 相)
	SENSOR_PATTERN_CCW_V_U	(5) (3)	CCW 時の通電パターン (V 相から U 相) ホール使用時 : (5) センサレス時 : (3)
	SENSOR_PATTERN_CCW_V_W	(1) (2)	CCW 時の通電パターン (V 相から W 相) ホール使用時 : (1) センサレス時 : (2)
	SENSOR_PATTERN_CCW_U_W	(3) (6)	CCW 時の通電パターン (U 相から W 相) ホール使用時 : (3) センサレス時 : (6)
	SENSOR_PATTERN_CCW_U_V	(2) (4)	CCW 時の通電パターン (U 相から V 相) ホール使用時 : (2) センサレス時 : (4)
	SENSOR_PATTERN_CCW_W_V	(6) (5)	CCW 時の通電パターン (W 相から V 相) ホール使用時 : (6) センサレス時 : (5)
	SENSOR_PATTERN_CCW_W_U	(4) (1)	CCW 時の通電パターン (W 相から U 相) ホール使用時 : (4) センサレス時 : (1)
	SENSOR_PATTERN_ERROR	(0)	通電パターンエラー用マクロ。
	SENSOR_UP_PWM_VN_ON	(1)	U 相上アームを PWM 出力し、V 相下アームに通電
	SENSOR_UP_PWM_WN_ON	(2)	U 相上アームを PWM 出力し、W 相下アームに通電
	SENSOR_VP_PWM_UN_ON	(3)	V 相上アームを PWM 出力し、U 相下アームに通電
	SENSOR_VP_PWM_WN_ON	(4)	V 相上アームを PWM 出力し、W 相下アームに通電
	SENSOR_WP_PWM_UN_ON	(5)	W 相上アームを PWM 出力し、U 相下アームに通電
	SENSOR_WP_PWM_VN_ON	(6)	W 相上アームを PWM 出力し、V 相下アームに通電
	SENSOR_UP_ON_VN_PWM	(7)	U 相上アームを通电し、V 相下アームを PWM 出力
	SENSOR_UP_ON_WN_PWM	(8)	U 相上アームを通电し、W 相下アームを PWM 出力
SENSOR_VP_ON_UN_PWM	(9)	V 相上アームを通电し、U 相下アームを PWM 出力	
SENSOR_VP_ON_WN_PWM	(10)	V 相上アームを通电し、W 相下アームを PWM 出力	



ファイル名	マクロ名	定義値	備考
r_motor_senso r.h	SENSOR_WP_ON_UN_PWM	(11)	W 相上アームを通电し、U 相下アームを PWM 出力
	SENSOR_WP_ON_VN_PWM	(12)	W 相上アームを通电し、V 相下アームを PWM 出力
	SENSOR_U_PWM_VN_ON	(13)	U 相を PWM 出力し、V 相下アームを通电
	SENSOR_U_PWM_WN_ON	(14)	U 相を PWM 出力し、W 相下アームを通电
	SENSOR_V_PWM_UN_ON	(15)	V 相を PWM 出力し、U 相下アームを通电
	SENSOR_V_PWM_WN_ON	(16)	V 相を PWM 出力し、W 相下アームを通电
	SENSOR_W_PWM_UN_ON	(17)	W 相を PWM 出力し、U 相下アームを通电
	SENSOR_W_PWM_VN_ON	(18)	W 相を PWM 出力し、V 相下アームを通电
	SENSOR_UP_ON_V_PWM	(19)	U 相上アームを通电し、V 相を PWM 出力
	SENSOR_UP_ON_W_PWM	(20)	U 相上アームを通电し、W 相を PWM 出力
	SENSOR_VP_ON_U_PWM	(21)	V 相上アームを通电し、U 相を PWM 出力
	SENSOR_VP_ON_W_PWM	(22)	V 相上アームを通电し、W 相を PWM 出力
	SENSOR_WP_ON_U_PWM	(23)	W 相上アームを通电し、U 相を PWM 出力
	SENSOR_WP_ON_V_PWM	(24)	W 相上アームを通电し、V 相を PWM 出力

#### 5.4.8 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 5.4.5 を参照ください。

5.4.9 速度算出方法

(a) ホールセンサ

ホールセンサを使用した場合のモータ回転速度は、コンペアマッチタイマのチャンネル 1 のタイマをフリーランニングさせ、ホールセンサ信号による外部割り込みルーチンでタイマ値を取り込み、 $2\pi$ [rad]前の取り込み値との差分から演算します。速度演算結果に対しては LPF（ローパスフィルタ）処理を行います。

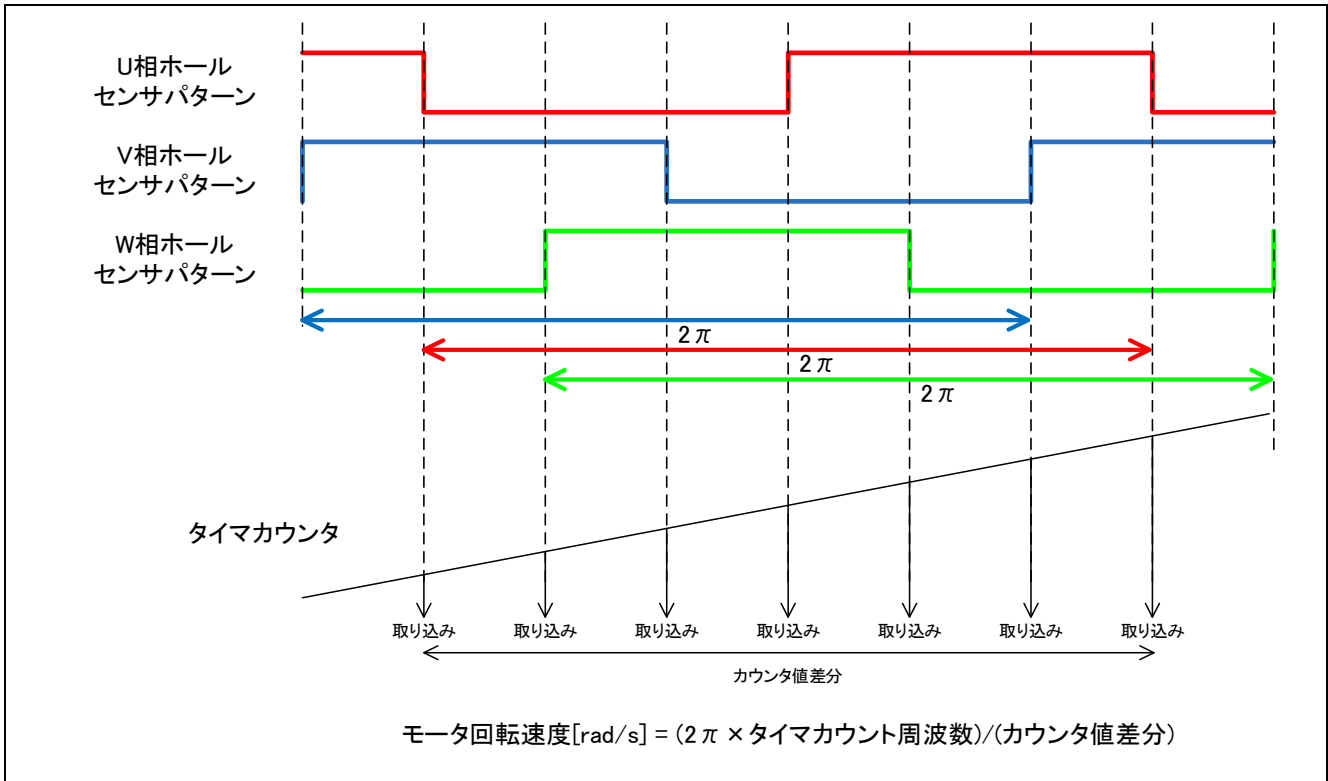


図 5-22 モータ回転速度の演算方法

(b) センサレス

センサレスでのモータ回転速度は、コンペアマッチタイマのチャンネル 1 のタイマをフリーランニングさせ、ゼロクロス発生時にタイマ値を取り込み、 $2\pi$  [rad] 前の取り込み値との差分から演算します。

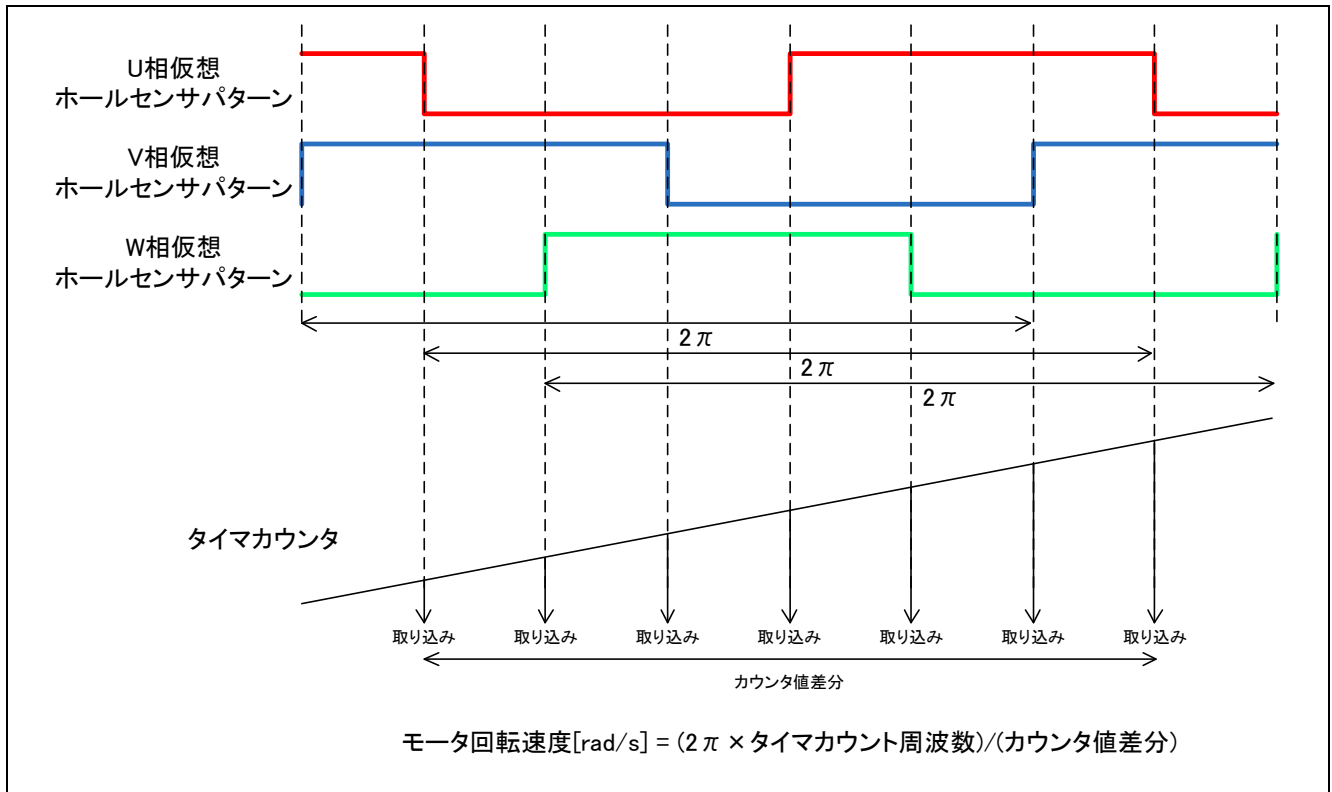


図 5-23 モータ回転速度の演算方法

### 5.5 ドライバモジュール

ドライバモジュールは、サンプルプログラム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。

#### 5.5.1 機能

ドライバモジュールの機能一覧を表 5-28 に示します。

表 5-28 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、相電圧*1 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。
速度計測用タイマの開始、停止	スマートコンフィグレータ関数経由で速度計測用タイマの開始、停止を設定します。
ホールセンサ割り込みの開始、停止	スマートコンフィグレータ関数経由でホールセンサの割り込み許可、禁止を設定します。

【注】 1. 相電圧はセンサレス時のみ取得する

#### 5.5.2 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 5-24 ドライバモジュール構成図に示します。

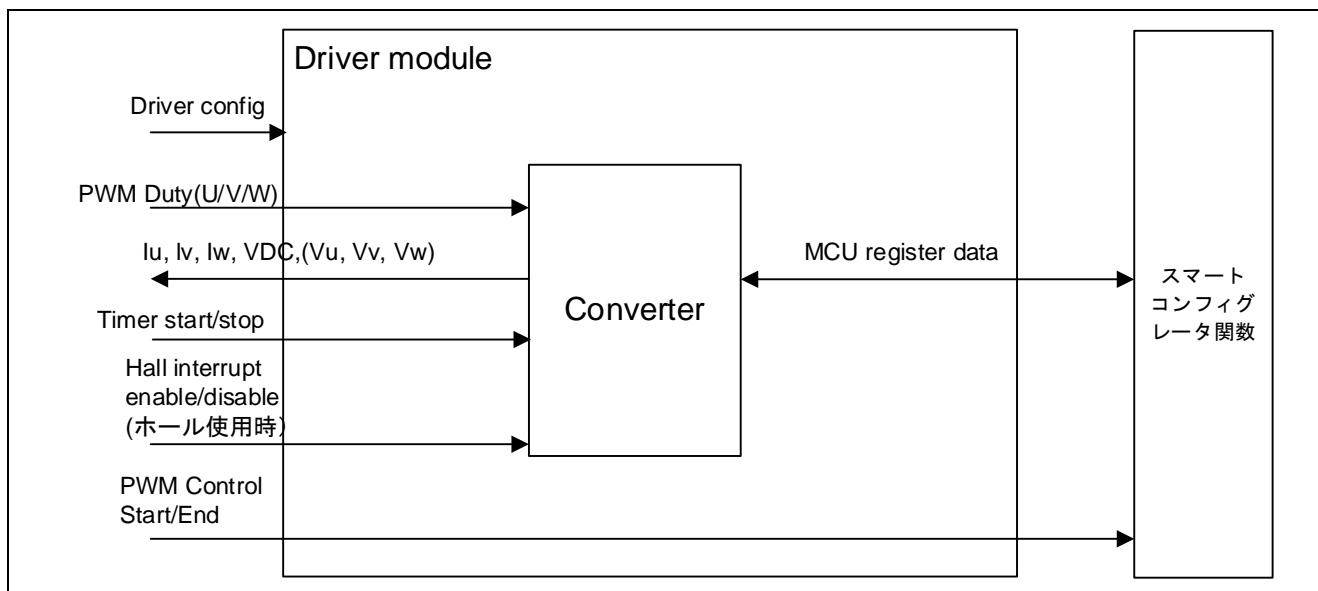


図 5-24 ドライバモジュール構成図

## 5.5.3 API

ドライバモジュールの API 一覧表と各 API の説明を表 5-29 に示します。

表 5-29 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_PWMControlStop	PWM 制御を停止します。
R_MOTOR_DRIVER_PWMControlStart	PWM 制御を開始します。
R_MOTOR_DRIVER_OutputPatternChange	120 度通電制御の通電パターンを切り替えます。

## 5.5.4 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 5-30 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。各 MCU の初期値は表 5-31 に示します。

表 5-30 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_CHANGE_PATTERN	通電パターンの変更を行う関数を設定
	DRIVER_CFG_FUNC_FREERUN_TIMER_START	速度計算用フリーランタイム開始の関数を設定
	DRIVER_CFG_FUNC_FREERUN_TIMER_STOP	速度計算用フリーランタイム停止の関数を設定
	DRIVER_CFG_FUNC_FREERUN_TIMERCNT_SET	速度計算用フリーランタイムのカウント値を設定する関数を設定
	DRIVER_CFG_FUNC_FREERUN_TIMERCNT_GET	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を設定
	DRIVER_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
	DRIVER_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
	DRIVER_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
	DRIVER_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
	DRIVER_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
	DRIVER_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 5-31 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_xxx_StartTimerCtrl (スマートコンフィグレータ関数) *1 *2
DRIVER_CFG_FUNC_PWM_OUTPUT_STOP	R_Config_xxx_StopTimerCtrl (スマートコンフィグレータ関数) *1 *2
DRIVER_CFG_FUNC_ADC_DATA_GET	R_Config_xxx_AdcGetConvVal (スマートコンフィグレータ関数) *1 *2
DRIVER_CFG_FUNC_CHANGE_PATTERN	R_Config_xxx_Chg_Pattern (スマートコンフィグレータ関数) *1 *2
DRIVER_CFG_FUNC_FREERUN_TIMER_START	R_Config_CMTx_Start (スマートコンフィグレータ関数) *1
DRIVER_CFG_FUNC_FREERUN_TIMER_STOP	R_Config_CMTx_Stop (スマートコンフィグレータ関数) *1
DRIVER_CFG_FUNC_FREERUN_TIMERCNT_SET	R_Config_CMTx_Set_Cmcnt (スマートコンフィグレータ関数) *1
DRIVER_CFG_FUNC_FREERUN_TIMERCNT_GET	R_Config_CMTx_Get_Cmcnt (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_U_START	R_Config_ICU_IRQx_Start (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_V_START	R_Config_ICU_IRQx_Start (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_W_START	R_Config_ICU_IRQx_Start (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_U_STOP	R_Config_ICU_IRQx_Stop (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_V_STOP	R_Config_ICU_IRQx_Stop (スマートコンフィグレータ関数) *1
DRIVER_CFG_HALL_FUNC_INT_W_STOP	R_Config_ICU_IRQx_Stop (スマートコンフィグレータ関数) *1

- 【注】 1. 設定値に記載した関数については、5.6 スマートコンフィグレータ設定を参照してください。  
 2. スマート・コンフィグレータのモータコンポーネントを使用する場合は、“xxx”は“MOTOR”と設定しています。モータコンポーネントを使用しない場合は、PWMに使用するモジュール名が入ります。

## 5.5.5 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 5-32 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 5-32 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	*ADCDDataGet	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)
	*ChangeOutputPattern	スマートコンフィグレータ関数へのポインタ (ホールパターン切り替え関数を設定)
	*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)
	*FreerunTimerStart	スマートコンフィグレータ関数へのポインタ (速度計測用タイマスタート関数を設定)
	*FreerunTimerStop	スマートコンフィグレータ関数へのポインタ (速度計測用タイマ停止関数を設定)
	*FreerunTimerSet	スマートコンフィグレータ関数へのポインタ (速度計測用タイマ値設定関数を設定)
	*FreerunTimerGet	スマートコンフィグレータ関数へのポインタ (速度計測用タイマ値取得関数を設定)
	*HallUEnable	スマートコンフィグレータ関数へのポインタ (U 相ホール割り込み許可設定関数を設定)
	*HallVEnable	スマートコンフィグレータ関数へのポインタ (V 相ホール割り込み許可設定関数を設定)
	*HallWEnable	スマートコンフィグレータ関数へのポインタ (W 相ホール割り込み許可設定関数を設定)
	*HallUDisable	スマートコンフィグレータ関数へのポインタ (U 相ホール割り込み禁止設定関数を設定)
	*HallVDisable	スマートコンフィグレータ関数へのポインタ (U 相ホール割り込み禁止設定関数を設定)
	*HallWDisable	スマートコンフィグレータ関数へのポインタ (U 相ホール割り込み禁止設定関数を設定)
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	st_motor_driver_t と同様	ドライバモジュールへ設定を行うための構造体。



## 5.5.6 マクロ定義

ドライバで定義しているマクロはありません。

## 5.5.7 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\_motor\_driver\_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 5-33 に示します。

表 5-33 サンプルプログラム設定例

変数名	マクロ名	ファイル名
*ADCDDataGet	DRIVER_CFG_FUNC_ADC_DATA_GET	表 5-30 参照。
* ChangeOutputPattern	DRIVER_CFG_FUNC_CHANGE_PATTERN	
*PWMOutputStop	DRIVER_CFG_FUNC_PWM_OUTPUT_START	
*PWMOutputStart	DRIVER_CFG_FUNC_PWM_OUTPUT_STOP	
*FreerunTimerStart	DRIVER_CFG_FUNC_FREERUN_TIMER_START	
*FreerunTimerStop	DRIVER_CFG_FUNC_FREERUN_TIMER_STOP	
*FreerunTimerSet	DRIVER_CFG_FUNC_FREERUN_TIMERCNT_SET	
*FreerunTimerGet	DRIVER_CFG_FUNC_FREERUN_TIMERCNT_GET	
*HallUEnable	DRIVER_CFG_HALL_FUNC_INT_U_START	
*HallVEnable	DRIVER_CFG_HALL_FUNC_INT_V_START	
*HallWEnable	DRIVER_CFG_HALL_FUNC_INT_W_START	
*HallUDisable	DRIVER_CFG_HALL_FUNC_INT_U_STOP	
*HallVDisable	DRIVER_CFG_HALL_FUNC_INT_V_STOP	
*HallWDisable	DRIVER_CFG_HALL_FUNC_INT_W_STOP	

## 5.6 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。

### 5.6.1 クロック設定

クロック設定を表 5-34 に示します。

表 5-34 MCU クロック設定

クロック	周波数		
	RX13T	RX23T	RX24T, RX24U
メインクロック	32MHz * <sup>1</sup>	10MHz	10MHz
システムクロック (ICLK)	32MHz	40MHz	80MHz
周辺モジュールクロック (PCLKA)	-	40MHz	80MHz
周辺モジュールクロック (PCLKB)	32MHz	40MHz	40MHz
周辺モジュールクロック (PCLKC)	-	-	-
周辺モジュールクロック (PCLKD)	32MHz	40MHz	40MHz
外部バスクロック (BCLK)	-	-	-
FlashIF クロック (FCLK)	32MHz	20MHz	20MHz
IWDTCLK	15kHz		

【注】 1. HOCO を使用して 32MHz を設定

### 5.6.2 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 5-35 に示します。

表 5-35 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		
	RX13T	RX23T	RX24T, RX24U
ホール割り込み処理	Config_ICU		
3相 PWM 出力、 電流検出の AD 変換	Config_MOTOR		
A/D 変換処理(インバータ母線電圧検出, ボード UI 用指令電圧検出)	Config_S12AD0		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_CMT1		
独立ウォッチドックタイマ	Config_IWDT		
過電流検出	Config_POE		

## 5.6.3 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 5-36 に示します。

表 5-36 割り込み一覧

コンポーネント	割り込み関数	説明
Config_ICU	r_Config_ICU_irqxx_interrupt* <sup>1</sup> r_Config_ICU_irqxx_interrupt* <sup>1</sup> r_Config_ICU_irqxx_interrupt* <sup>1</sup>	ホールセンサの割り込み 割り込みレベル：13 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：14 多重割り込み：許可
Config_S12AD0/2	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：13 多重割り込み：許可
Config_CMT1	なし	なし
Config_IWDT	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

【注】 1. “xx”は各 MCU に割り当てられた設定です。詳細は表 5-35 を参照ください。

## 5.6.4 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 5-37 に示します。

表 5-37 ユーザ領域の関数一覧

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_Led3_on	LED3 点灯
	R_Config_PORT_Led1_off	LED1 消灯
	R_Config_PORT_Led2_off	LED2 消灯
	R_Config_PORT_Led3_off	LED3 消灯
Config_MOTOR	R_Config_MOTOR_Change_Pattern	通電パターン変更

## 5.6.5 端子設定

端子のインタフェース情報を表 5-38 に示します。

表 5-38 端子インタフェース

機能	RX13T	RX23T	RX24T	RX24U
インバータ 母線電圧測定	P46/AN006	P43/AN003	P64/AN204	P64/AN204
位置/速度指令値入力用 (アナログ値)	P47/AN007	P10/AN017	P53/AN209	P53/AN209
START/STOP トグルスイッチ	PB5	P91	P80	P80
ERROR RESET トグルスイッチ	PB4	P92	P81	P81
LED1 制御	PD6	P00	PA2	PA2
LED2 制御	PD4	P01	PA1	PA1
LED3 制御	PB3	P31	PD7	PD7
U 相電流測定	P40/AN000	P40/AN000	P44/AN100	P44/AN100
V 相電流測定	P41/AN001	P41/AN001	P45/AN101	P45/AN101
W 相電流測定	P42/AN002	P42/AN002	P46/AN102	P46/AN102
PWM 出力 (U <sub>p</sub> )	P71/MTIOC3B	P71/MTIOC3B	P71/MTIOC3B	P71/MTIOC3B
PWM 出力 (V <sub>p</sub> )	P72/MTIOC4A	P72/MTIOC4A	P72/MTIOC4A	P72/MTIOC4A
PWM 出力 (W <sub>p</sub> )	P73/MTIOC4B	P73/MTIOC4B	P73/MTIOC4B	P73/MTIOC4B
PWM 出力 (U <sub>n</sub> )	P74/MTIOC3D	P74/MTIOC3D	P74/MTIOC3D	P74/MTIOC3D
PWM 出力 (V <sub>n</sub> )	P75/MTIOC4C	P75/MTIOC4C	P75/MTIOC4C	P75/MTIOC4C
PWM 出力 (W <sub>n</sub> )	P76/MTIOC4D	P76/MTIOC4D	P76/MTIOC4D	P76/MTIOC4D
ホール U 相入力	P93/IRQ0	P30/IRQ0	P10/IRQ0	P10/IRQ0
ホール V 相入力	P94/IRQ1	P29/IRQ1	P11/IRQ1	P11/IRQ1
ホール W 相入力	PA2/IRQ4	P28/IRQ4	P96/IRQ4	P01/IRQ4
U 相電圧測定	P43/AN003	P44/AN004	P61/AN201	P61/AN201
V 相電圧測定	P44/AN004	P45/AN005	P62/AN202	P62/AN202
W 相電圧測定	P45/AN005	P46/AN006	P63/AN203	P63/AN203
過電流検出時の PWM 緊急停 止入力	PE2/POE10#	P70/POE0#	P70/POE0#	P70/POE0#

6. 120 度通電制御アルゴリズム

6.1 120 度通電制御

図 6-1 のように 60 度毎に通電パターンを変化させると、コイル磁束と回転子の永久磁石との間にトルクが生じ、コイル磁束に同期して回転子が回転します。このとき各スイッチング素子の通電期間が 120 度である事から、120 度通電制御と呼ばれています。

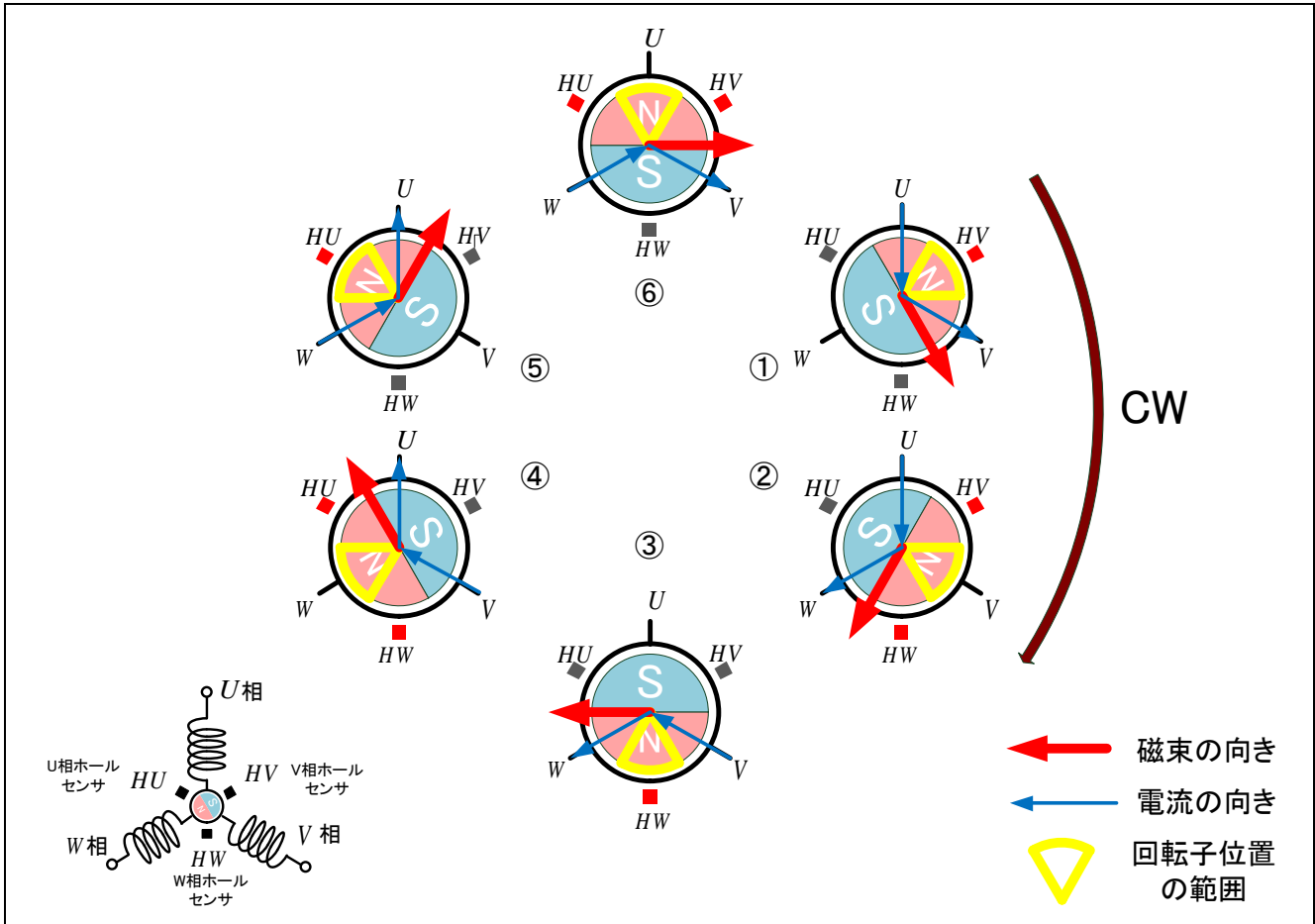


図 6-1 6 つの通電パターンと回転子位置範囲例

6.2 120 度通電制御における位置・速度算出

6.2.1 ホールセンサ利用 120 度通電制御

(a) 位置検出

ホールセンサを使用した永久磁石位置の位置検出では、ホールセンサ信号を位置情報としてマイコンに入力します。

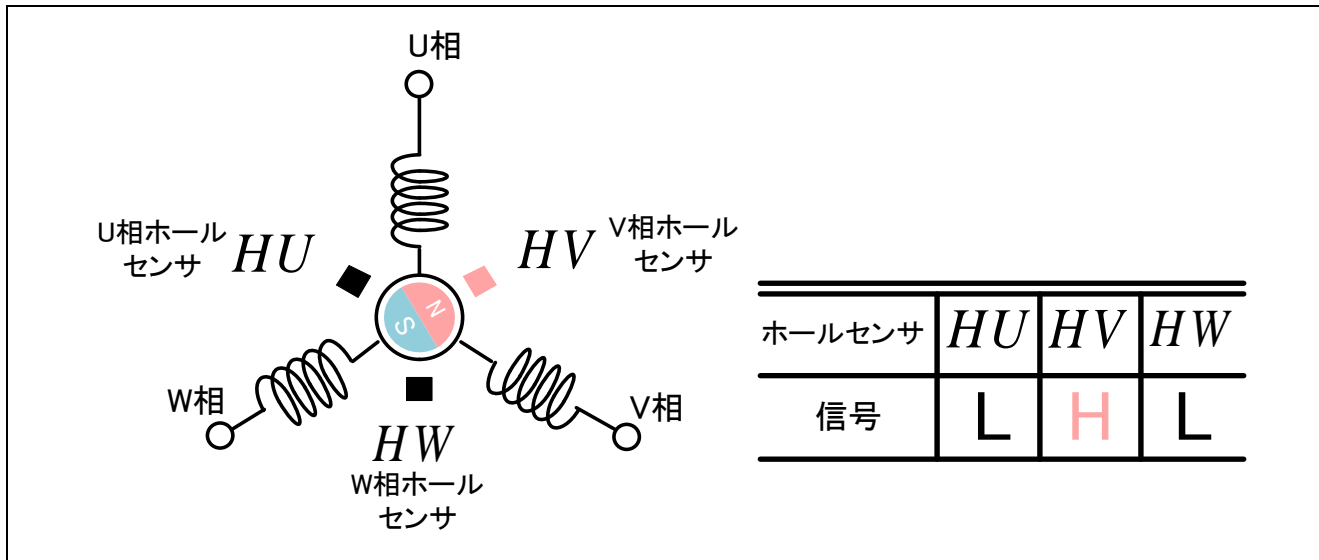


図 6-2 ホールセンサ(HU、HV、HW)の位置と位置信号の例

図 6-2 ではホールセンサは 120 度毎に配置され、それぞれのホールセンサ信号は永久磁石の磁極の変化により切り替わります。この 3 つのホールセンサ信号を組み合わせることで 60 度毎(1 周につき 6 パターン)に位置情報を得ることが出来ます。そのホールセンサ信号の切り替えタイミングで図 6-3 のように各相の通電パターンを変化させます。

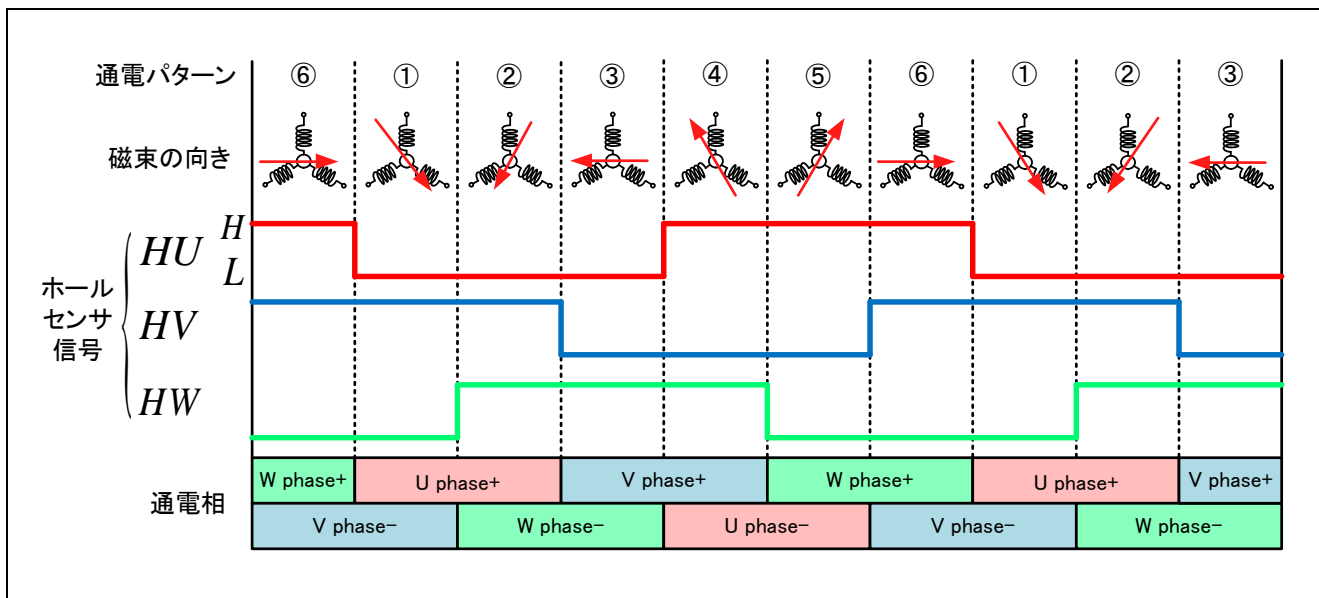


図 6-3 ホールセンサ信号と通電パターンの関係例(回転方向 : CW)

(b) 速度演算

モータ回転速度は、マイコン周辺機能のタイマをフリーランニングさせ、ホールセンサ信号による外部割り込みルーチンでタイマ値を取り込み、 $2\pi$  [rad]前の取り込み値との差分から演算することができます。この演算手法を用いると、ホールセンサ取り付け位置が不均一な場合でも回転速度を演算する事が出来ます。

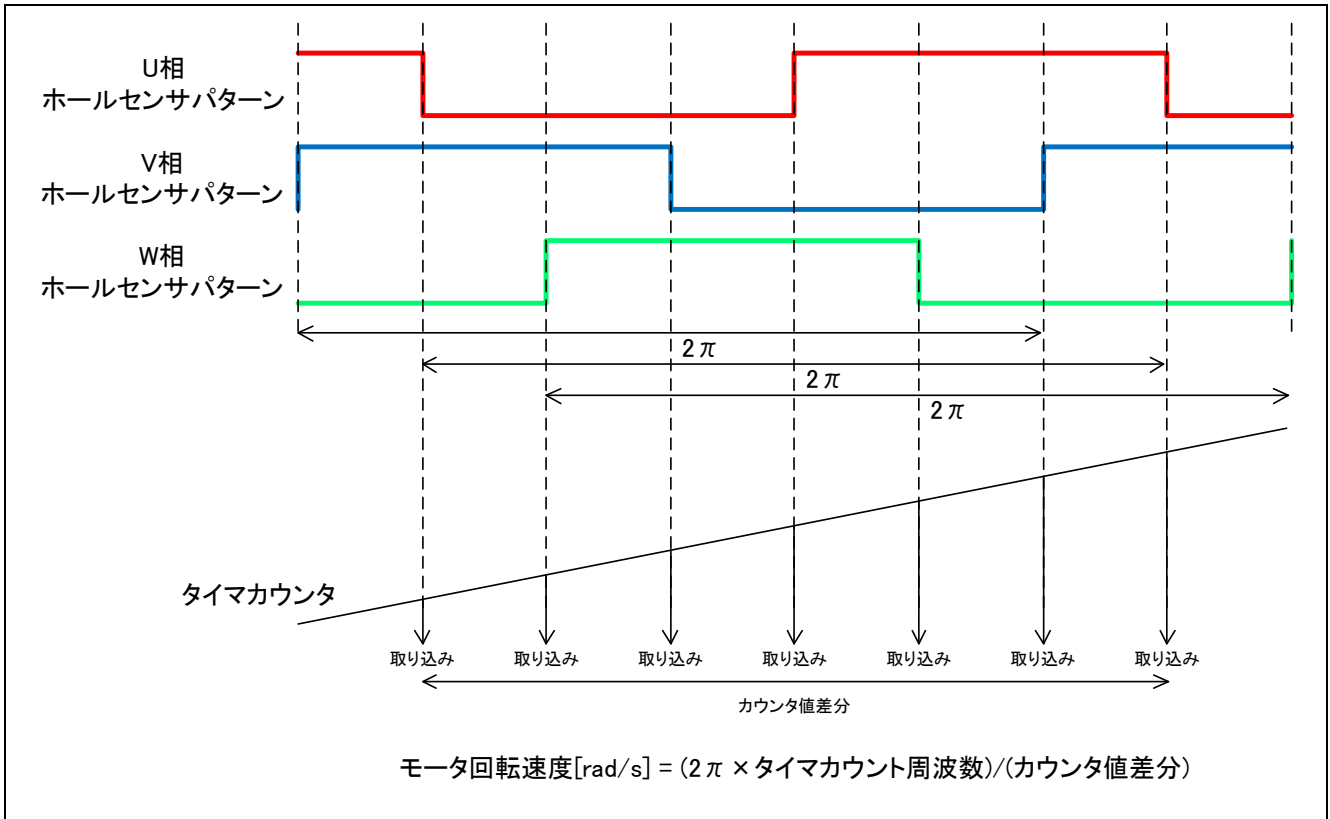


図 6-4 モータ回転速度の演算方法



## 6.2.2 センサレス 120 度通電制御

## (a) 位置推定

センサレス制御では永久磁石位置を知るためのセンサがありませんので、センサに変わって位置情報を得る手段が必要になります。永久磁石同期モータのセンサレス制御では、誘起電圧を検出することで位置を推定することが一般的です。

誘起電圧とは、コイル内を貫く磁束が変化する際、その変化を妨げるように、その割合に比例して発生するものです。

例えば、図 6-5 のようにコイルに磁石を近づけた場合を考えます。この場合、コイルを貫く磁束は増加するので、コイルは磁束の増加を妨げるために図の方向に電流を流す起電力を発生させます。（右ねじの法則により、磁石の磁束と逆向きの磁束が発生します）

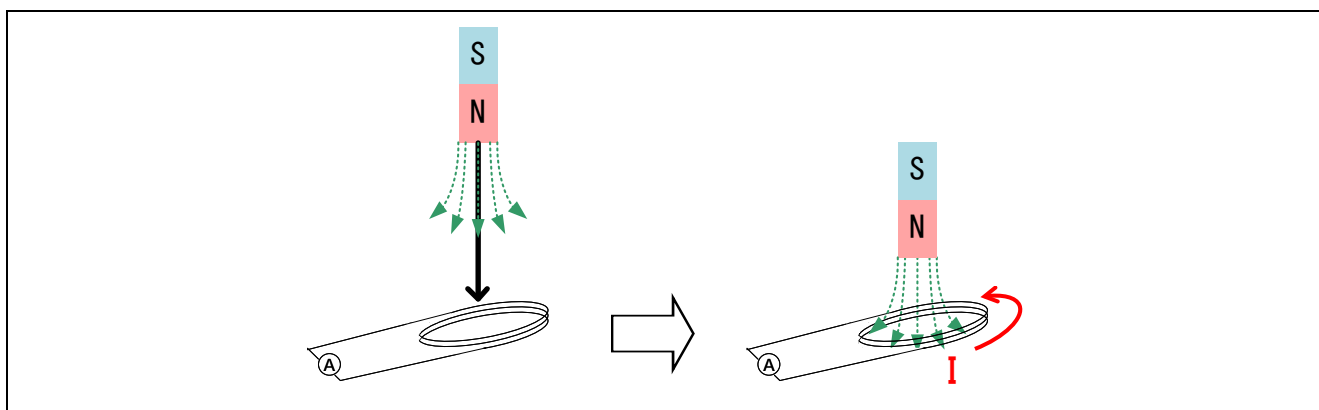


図 6-5 コイルと磁石による誘起電圧

この誘起電圧  $E_m$  は、磁石の磁束を  $\psi_m$  として以下の式で表すことができます。

$$E_m = -\frac{d}{dt} \psi_m \dots (1)$$

この現象は回転している永久磁石同期モータでも発生します。永久磁石が回転している場合、各相の鎖交磁束が常に変化することで、誘起電圧が発生しています。

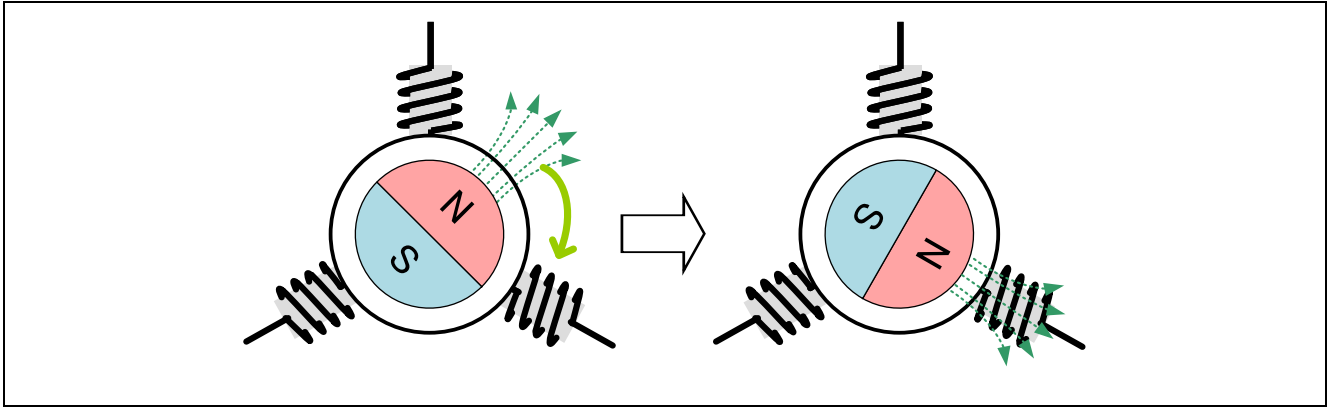


図 6-6 回転する永久磁石同期モータにおける誘起電圧

図 6-7 は、U 相における鎖交磁束の変化を表したもので、縦軸に鎖交磁束の大きさ、横軸に永久磁石の位相を表しています。また、永久磁石の N 極が U 相のコイルを向いている位置を  $\theta=0$  としています。

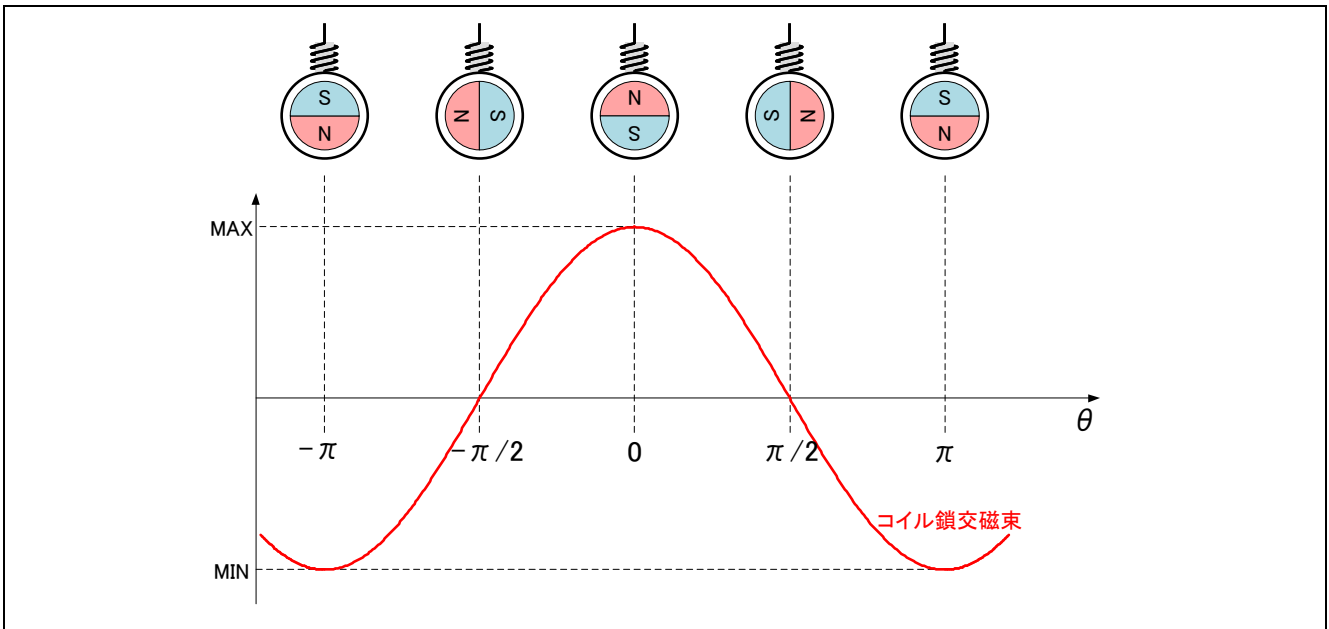


図 6-7 鎖交磁束の変化

U 相の鎖交磁束は余弦波状に変化します。

さらに、V 相、W 相についても同様に考えた場合、V 相、W 相は U 相からそれぞれ  $2\pi/3$ 、 $-2\pi/3$  位相がずれているので、3 相の鎖交磁束は以下の式で表すことができます。

$$\begin{aligned} \varphi_u &= \varphi_m \cos \theta \\ \varphi_v &= \varphi_m \cos\left(\theta - \frac{2}{3}\pi\right) \\ \varphi_w &= \varphi_m \cos\left(\theta + \frac{2}{3}\pi\right) \end{aligned}$$

また、3相の誘起電圧は角速度を $\omega$ とすると、式(1)を用いて以下の式で表すことができます。

$$E_u = \frac{d}{dt} \varphi_u = \frac{d}{dt} \varphi_m \cos \theta = -\omega \varphi_m \sin \theta = \omega \varphi_m \cos(\theta + \frac{\pi}{2})$$

$$E_v = \frac{d}{dt} \varphi_v = \frac{d}{dt} \varphi_m \cos(\theta - \frac{2}{3} \pi) = -\omega \varphi_m \sin(\theta - \frac{2}{3} \pi) = \omega \varphi_m \cos(\theta - \frac{\pi}{6})$$

$$E_w = \frac{d}{dt} \varphi_w = \frac{d}{dt} \varphi_m \cos(\theta + \frac{2}{3} \pi) = -\omega \varphi_m \sin(\theta + \frac{2}{3} \pi) = \omega \varphi_m \cos(\theta + \frac{\pi}{6})$$

本式より、各相に発生する誘起電圧は、永久磁石磁束に対して $\pi/2$ 位相が進んでいることが分かります。つまり、誘起電圧を検出することができれば、永久磁石の位置が推定できるということです。

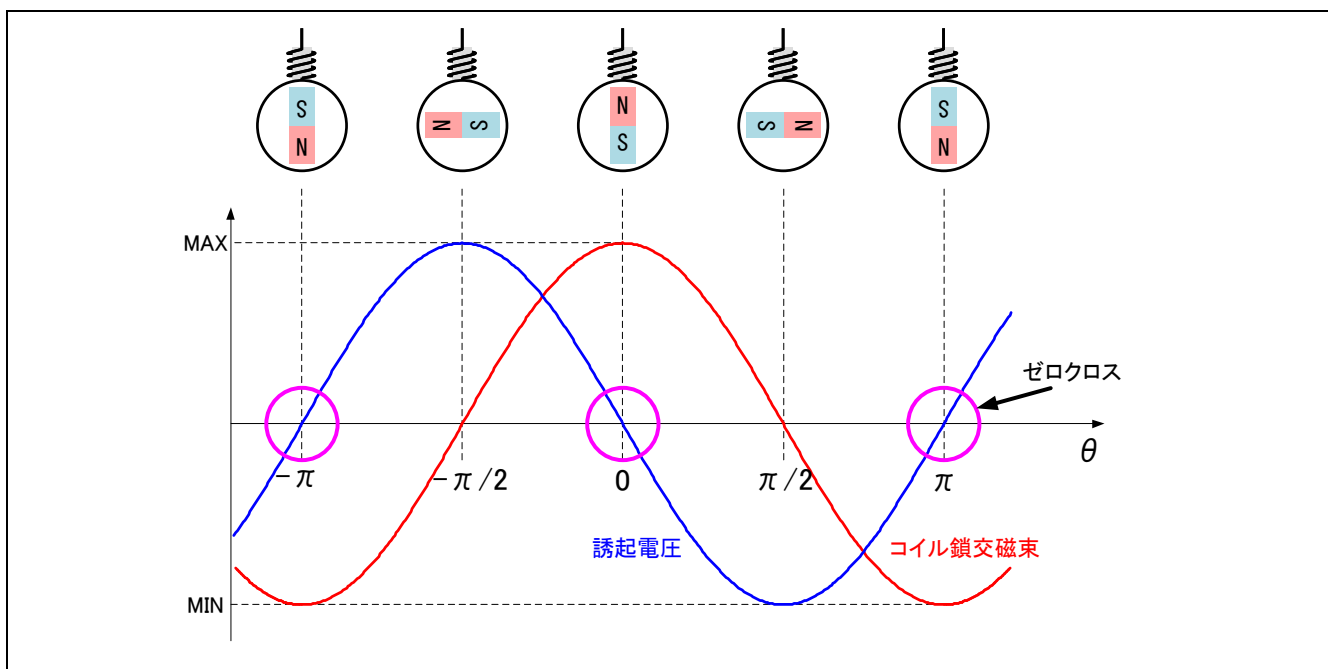


図 6-8 誘起電圧のゼロクロス

ただし、各相の誘起電圧は、モータが回転している間、常に検出できるわけではありません。

120度通電で駆動中は、3相のうち2相に通電していますので、残りの通電していない1相のみ誘起電圧を検出することができます。実際は、誘起電圧が検出できる無通電の相に発生する誘起電圧の符号が替わる点（ゼロクロス）を検出することで位置情報を得ています。

3相モータでは、このゼロクロスがモータ1回転（電気角）する間で各相2回、合計6回発生します。つまり、このゼロクロスを検出することで60度毎の位置検出ができ、ホールセンサと同等の分解能で位置情報を推定することができます。

本システムでは、PWM制御毎に各相電圧をA/D変換し、各相の和から仮想モータ中点電圧を求めています。その仮想モータ中点電圧と各相電圧を比較し、その位置関係によって「1」「0」のパターンを作成しています。

さらに、作成したパターンを  $\pi/6$  位相をずらし、仮想のホールセンサパターンを作成しています。  
 $\pi/6$  は、速度推定値から算出しています。

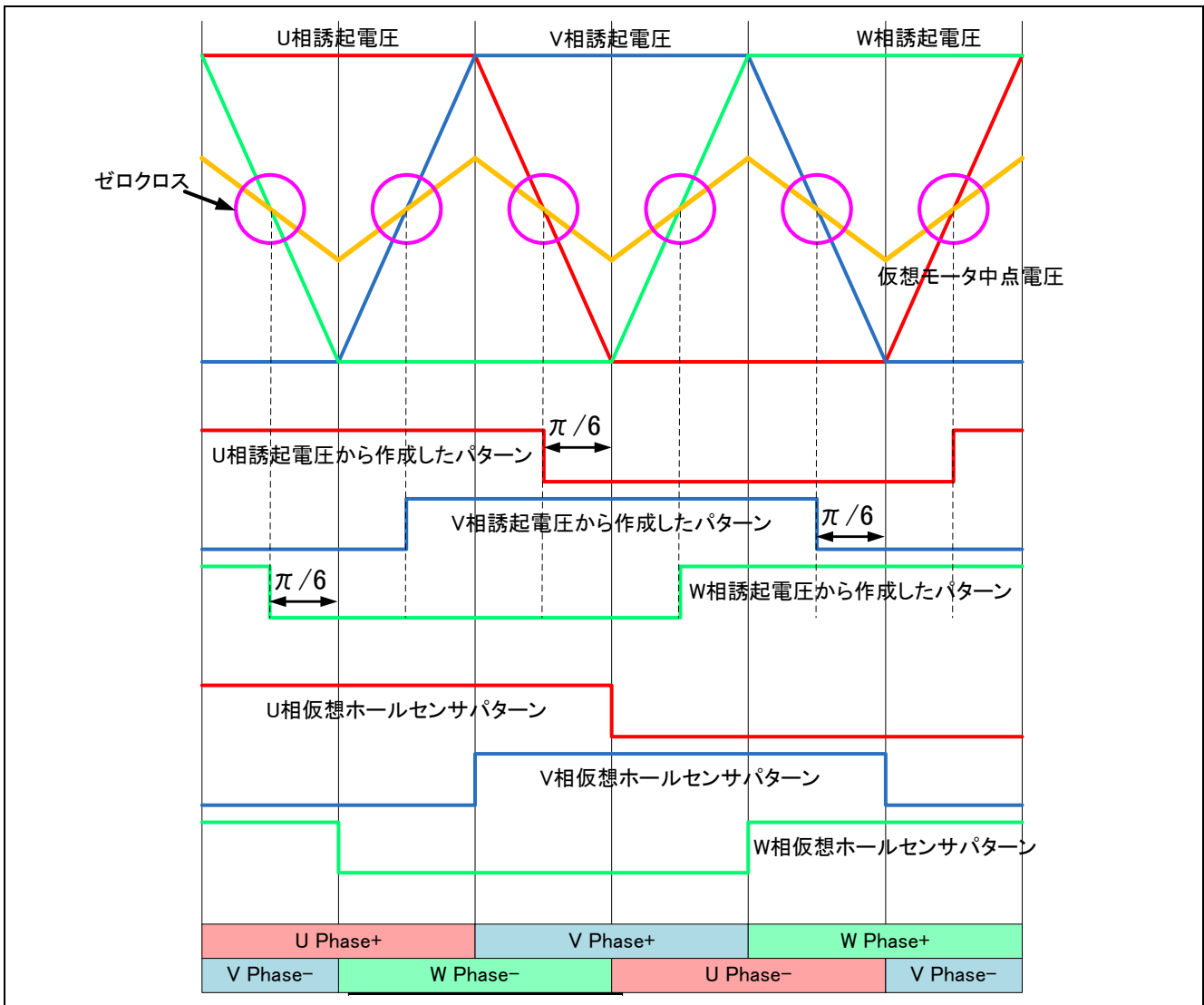


図 6-9 仮想ホールセンサパターン(上アームチョッピング時)

次にゼロクロス検出方法の一つとして、マイコンの A/D コンバータを用いた、誘起電圧と中点電圧の値をソフトウェアで比較してゼロクロスを検出する方式を紹介します。この方式は電圧の比較にコンパレータを使用しないため、コンパレータレス方式とも言われています。

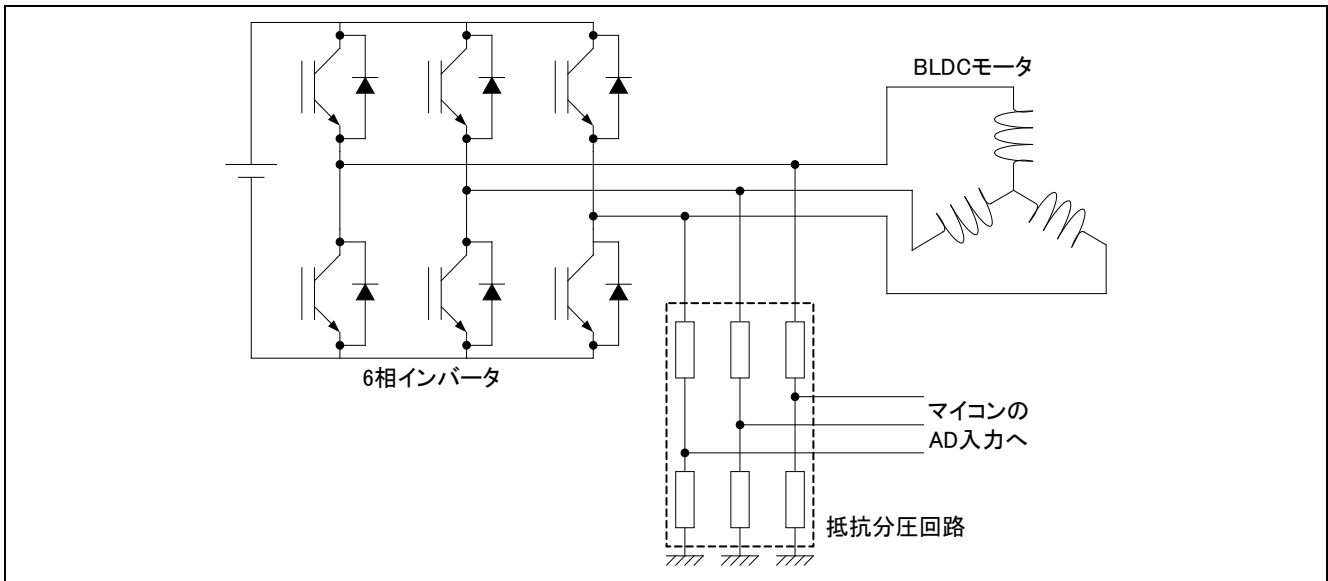


図 6-10 コンパレータレス方式

実際に検出される誘起電圧は、通電パターン切り替え時に発生する転流電圧と他相の PWM から受ける影響を考慮しなくてはなりません。その影響は図 6-11 に表したような形で現れます。この影響を軽減する方法として簡単なフィルタ回路の使用する方法やソフトウェアフィルタによる方法等があります。

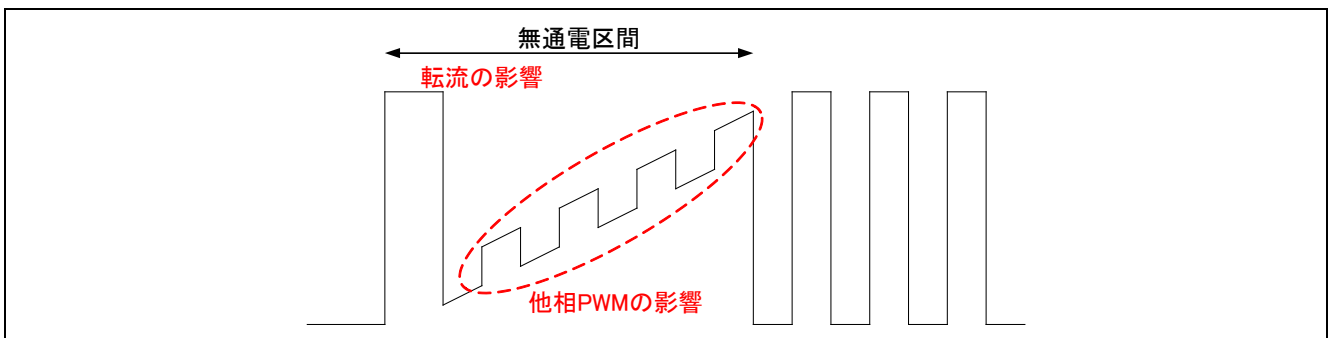


図 6-11 転流と他相 PWM の影響概念図

(b) 速度演算

モータ回転速度は、マイコン周辺機能のタイマをフリーランニングさせ、ゼロクロスを検出し、パターン切り替えを行う際にタイマ値を取り込み、 $2\pi$  [rad]前の取り込み値との差分から演算します。

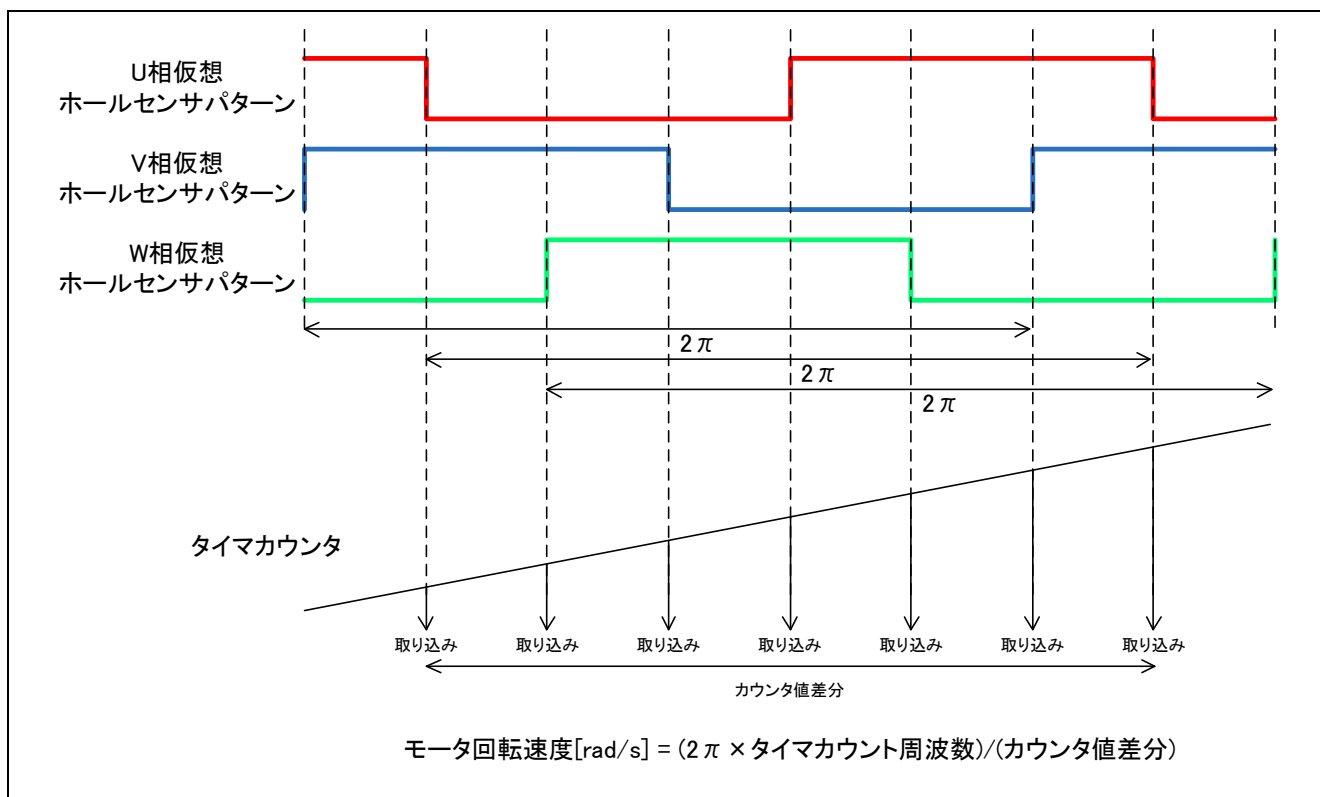


図 6-12 モータ回転速度の演算方法

## 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 (ホール使用時)

メモリ	RX13T	RX23T	RX24T	RX24U
ROM	16.3 KB	15.8 KB	16.2 KB	16.5 KB
RAM	4.9 KB	4.9 KB	5.0 KB	5.0 KB
スタック解析結果の最大値	164 B	164 B	164 B	164 B
スタックサイズ設定値	1536 B	1536 B	1536 B	1536 B

表 7-2 プログラムサイズ (センサレス時)

メモリ	RX13T	RX23T	RX24T	RX24U
ROM	17.7 KB	17.2 KB	17.8 KB	17.6 KB
RAM	5.1 KB	5.1 KB	5.2 KB	5.2 KB
スタック解析結果の最大値	204 B	204 B	204 B	204 B
スタックサイズ設定値	1536 B	1536 B	1536 B	1536 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 7-3 制御ループと CPU 負荷率（ホール使用時）

MCU	制御ループ種類	制御周期	処理時間	CPU 負荷率
RX24U	キャリア割り込み	50 $\mu$ s (間引き 0 回)	8.7 $\mu$ s	17.4 %
	速度制御ループ	1000 $\mu$ s	5.0 $\mu$ s	0.5 %
RX24T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	8.4 $\mu$ s	16.8 %
	速度制御ループ	1000 $\mu$ s	4.9 $\mu$ s	0.5 %
RX23T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	16.0 $\mu$ s	32.0 %
	速度制御ループ	1000 $\mu$ s	9.8 $\mu$ s	1.0 %
RX13T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	19.2 $\mu$ s	38.4 %
	速度制御ループ	1000 $\mu$ s	12.0 $\mu$ s	1.2 %

表 7-4 制御ループと CPU 負荷率（センサレス時）

MCU	制御ループ種類	制御周期	処理時間	CPU 負荷率
RX24U	キャリア割り込み	50 $\mu$ s (間引き 0 回)	15.3 $\mu$ s	30.6 %
	速度制御ループ	1000 $\mu$ s	4.1 $\mu$ s	0.4 %
RX24T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	15.7 $\mu$ s	31.4 %
	速度制御ループ	1000 $\mu$ s	4.1 $\mu$ s	0.4 %
RX23T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	27.6 $\mu$ s	55.2 %
	速度制御ループ	1000 $\mu$ s	8.8 $\mu$ s	0.9 %
RX13T	キャリア割り込み	50 $\mu$ s (間引き 0 回)	34.4 $\mu$ s	68.8 %
	速度制御ループ	1000 $\mu$ s	10.4 $\mu$ s	1.0 %



### 7.3 動作波形

試験結果として、参考に RX24T を使用した制御時の波形を示します。

表 7-5 測定条件

項目	値	備考
速度制御系制御パラメータ Kp	0.02f	
速度制御系制御パラメータ Ki	0.0005f 0.004f	ホール センサレス
負荷	—	無負荷で実施

ホールセンサ使用時の速度制御を行った結果を図 7-1 に示します。

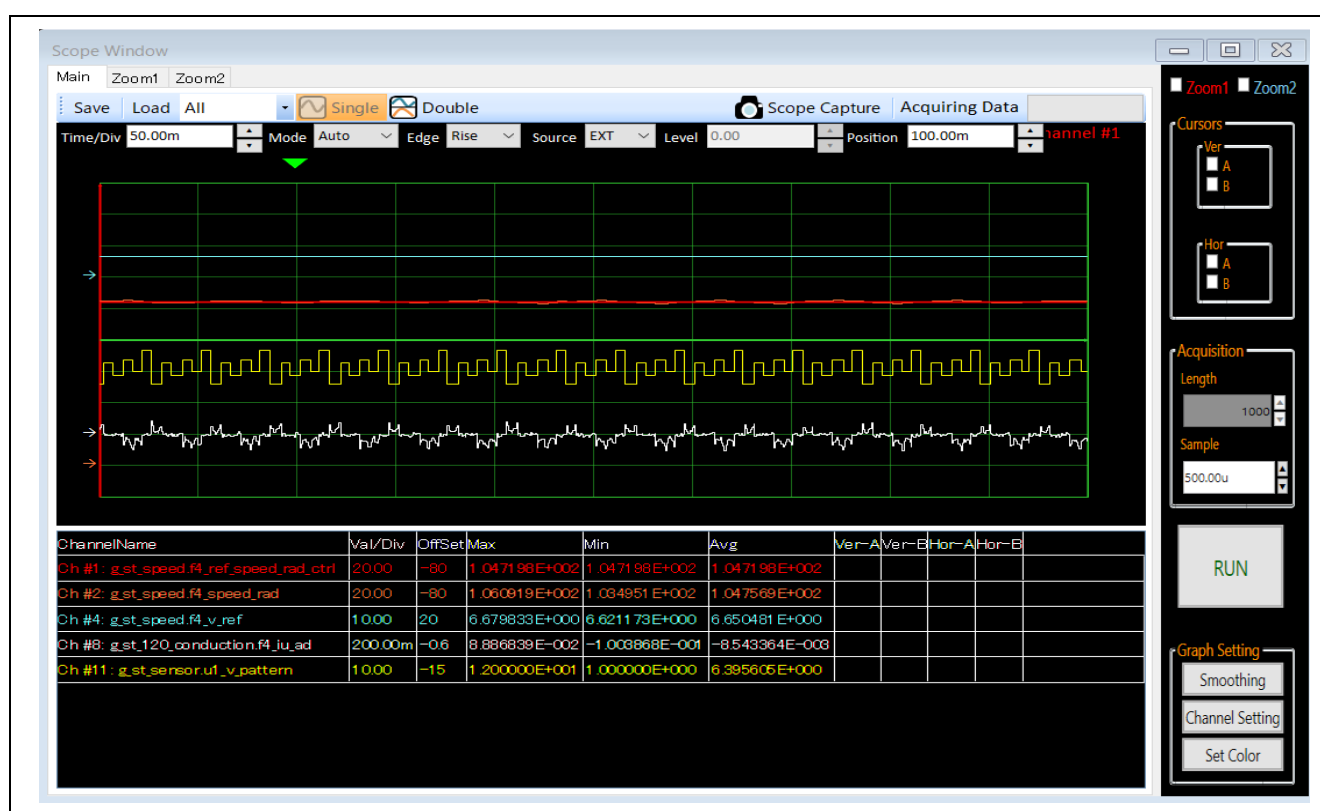


図 7-1 ホール 120 度通電制御を使用した速度制御

駆動条件：

回転速度：指令速度 1000 [rpm]

波形情報：

- 赤色：指令速度 [rad/s], (20rad/s / div)
- オレンジ色：検出速度 [rad/s], (20rad/s / div)
- 水色：指令電圧 [V], (10V / div)
- 黄色：通電パターン
- 白色：U 相電流値 [A] (200mA / div)
- 横軸：50ms / div.

センサレス駆動時の速度制御を行った結果を図 7-2 に示します。



図 7-2 センサレス 120 度通電制御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：

赤色：指令速度 [rad/s], (20rad/s / div)

オレンジ色：検出速度 [rad/s], (20rad/s / div)

水色：指令電圧 [V], (10V / div)

黄色：通電パターン

白色：U 相電流値 [A] (200mA / div)

横軸：50ms / div.

## 8. 参考資料

- Renesas Motor Workbench V.3.00 ユーザーズマニュアル (R21UZ0004)
- Evaluation System for BLDC Motor ユーザーズマニュアル (R12UZ0062)
- 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 : e<sup>2</sup> studio 編 (R20AN0451)
- RX13T CPU カード取扱説明書 (R12UZ0051)
- RX13T グループ ユーザーズマニュアル ハードウェア編 (R01UH0822)
- RX23T CPU カード取扱説明書 (R20UT3698)
- RX23T グループ ユーザーズマニュアル ハードウェア編 (R01UH0520)
- RX24T CPU カード取扱説明書 (R20UT3696)
- RX24T グループ ユーザーズマニュアル ハードウェア編 (R01UH0576)

## 9. 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	May 23.22	—	初版発行

## 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

- 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みます。以下同じです。）に関し、当社は、一切その責任を負いません。
  - 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。
  - 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。
  - 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。
  - 当社製品を、全部または一部を問わず、改造、変更、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、変更、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。
  - 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）、交通制御（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。
  - 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、これに限りません。）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。
  - 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。
  - 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。
  - 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。
  - 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。
  - 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いたします。
  - 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。
  - 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](http://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/](http://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。