RA6T1

永久磁石同期モータのセンサレスベクトル制御

要旨

本アプリケーションノートは RA6T1 マイクロコントローラで永久磁石同期モータを駆動するセンサレスベクトル制御ソフトウェア及びモータ制御開発支援ツール「Renesas Motor Workbench」の使用方法について説明することを目的としています。

本アプリケーションノート対象ソフトウェア(サンプルソフトウェア)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。サンプル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上で御使用ください。

動作確認デバイス

サンプル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。

・ RA6T1 (R7FA6T1AD3CFP)

目次

1. 概説 3
1.1 開発環境 3
2. システム概要 4
2.1 ハードウェア構成 4
2.2 ハードウェア仕様 5
2.2.1 ユーザインタフェース 5
2.2.2 周辺機能 7
2.3 ソフトウェア構成 12
2.3.1 ソフトウェア・ファイル構成 12
2.3.2 モジュール構成 13
2.4 ソフトウェア仕様 14
2.5 割込み優先順位 15
3. 制御ソフトウェア説明 16
3.1 制御内容 16
3.1.1 モータ起動/停止 16
3.1.2 A/D 変換 16
3.1.3 変調 17
3.1.4 状態遷移 19
3.1.5 始動方法 20
3.1.6 システム保護機能 21
3.1.7 AD トリガ 22
3.2 センサレスベクトル制御ソフトウェア関数仕様 23
3.3 Contents of control 30
3.3.1 Configuration Options 30

3.3.2 Configuration Options for included modules 30
3.4 制御フロー（フローチャート） 38
3.4.1 メイン処理 38
3.4.2 50 [µs]周期割り込み（キャリア同期割り込み）処理 39
3.4.3 500 [µs]周期割り込み処理 40
3.4.4 過電流検出割り込み処理 41

4. 評価環境説明 42
4.1 動作環境 42
4.2 プロジェクトのインポート 42
4.3 ビルドとデバッグ 43
4.4 サンプルソフトウェア操作概要 44
4.4.1 クイックスタート 44
4.5 モータ制御開発支援ツール「Renesas Motor Workbench」 45
4.5.1 概要 45
4.5.2 Easy 機能操作例 46
4.5.3 Analyzer 機能用変数一覧 48
4.5.4 Analyzer 機能操作例 49
4.5.5 Tuner 機能 50

5. 参考ドキュメント 51

1. 概説

本アプリケーションノートでは、RA6T1 マイクロコントローラで永久磁石同期モータ(PMSM)を駆動するセンサレスベクトル制御ソフトウェアの実装方法及びモータ開発支援ツール「Renesas Motor Workbench」の使用方法について説明します。

なお、このサンプルサンプルソフトウェアは「永久磁石同期モータのセンサレスベクトル制御 (アルゴリズム編)」(R01AN3786)のアルゴリズムを使用していますので、アルゴリズムの詳細についてはそちらを参照してください。

1.1 開発環境

サンプルソフトウェアの開発環境を表 1-1、表 1-2 に示します。

表 1-1 ハードウェア開発環境

マイコン	評価ボード 注1	モータ 注2
RA6T1(R7FA6T1AD3CFP)	48V 系インバータボード RA6T1 CPU カード	TG-55L-KA 24V

表 1-2 ソフトウェア開発環境

e²studio バージョン	FSP バージョン	ツールチェーン バージョン
V2022-10	V4.1.0 以降	GCC ARM Embedded : V10.3.1.20210824

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。

【注】 1. 48V 系インバータボード(RTK0EM0000B10020BJ) は、キット製品 RTK0EMA170S00020BJ、RTK0EMX270S00020BJ に同梱しており、ルネサスエレクトロニクス株式会社の製品です。
RA6T1 CPU カード(RTK0EMA170C00000BJ)は、ルネサスエレクトロニクス株式会社の製品です。
2. TG-55L-KA,24V は、ツカサ電工株式会社の製品です。
ツカサ電工株式会社(https://www.tsukasa-d.co.jp/)

2. システム概要

サンプルソフトウェアのシステム概要を以下に説明します。

2.1 ハードウェア構成

ハードウェア構成を次に示します。

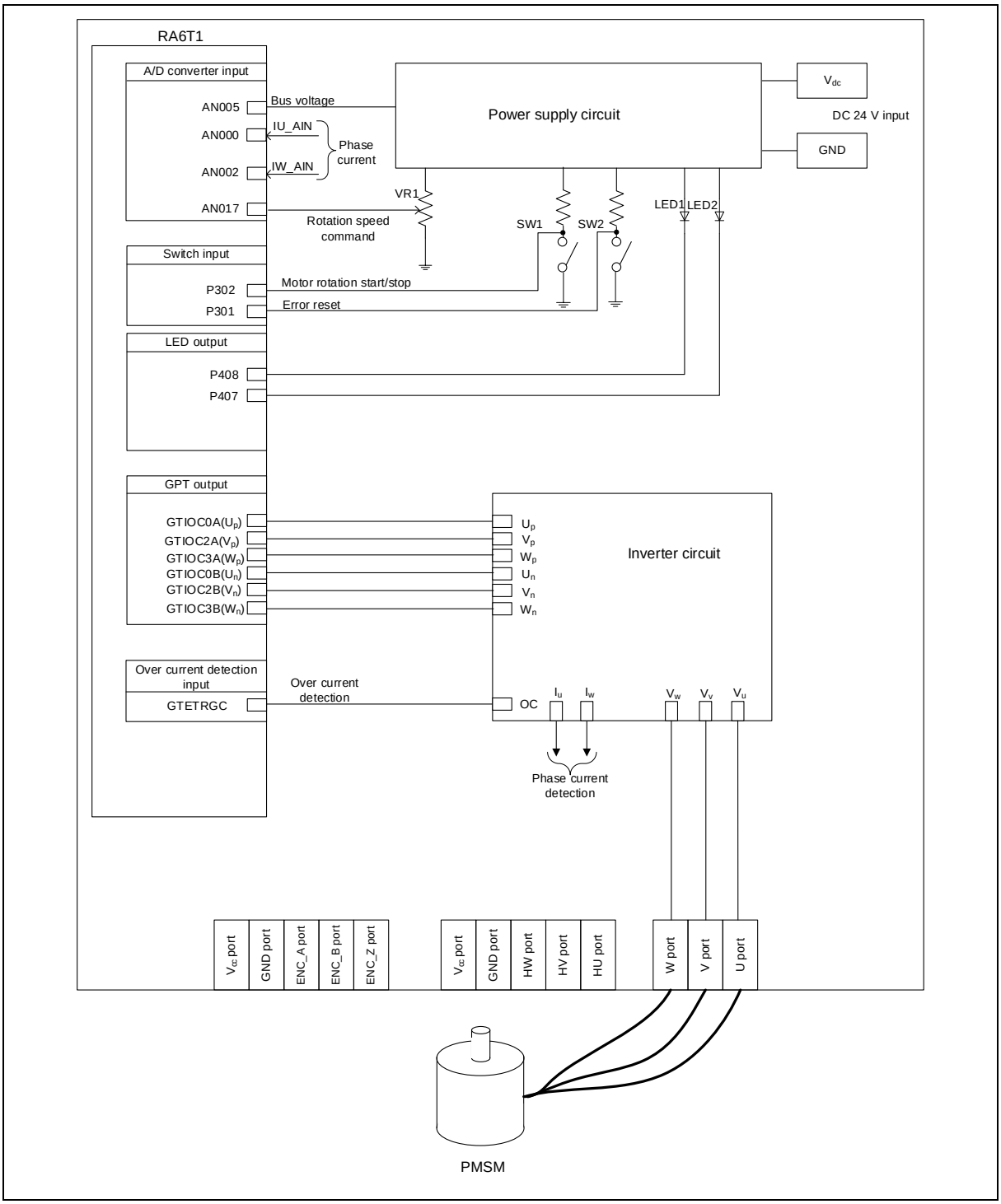


図 2-1 ハードウェア構成図

2.2 ハードウェア仕様

2.2.1 ユーザインタフェース

サンプルソフトウェアのユーザインタフェースを表 2-1 に示します。

表 2-1 インバータ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
回転速度指令	可変抵抗器	回転速度指令値入力
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	橙色 LED	・モータ駆動時　：点灯 ・モータ停止時　：消灯
LED2	橙色 LED	・エラー検出時　：点灯 ・通常動作時　：消灯
LED3	橙色 LED	未使用

表 2-2 CPU 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
LED1	緑色 LED	・モータ駆動時　：点灯 ・モータ停止時　：消灯
LED2	緑色 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(RESET)	システムリセット

サンプルソフトウェアの端子インタフェースを表 2-3 に示します。

表 2-3 端子インタフェース

R7FA6T1AD3CFP 端子名	機能
P014 / AN005	インバータ母線電圧測定
P502 / AN017	回転速度指令値入力用（アナログ値）
P302	START/STOP トグルスイッチ
P301	ERROR RESET プッシュスイッチ
P408	LED1 点灯／消灯制御
P407	LED2 点灯／消灯制御
P000 / AN000	U 相電流測定
P001 / AN001	U 相電流測定
P002 / AN002	W 相電流測定
P415 / GTIOC0A	PWM 出力 (Up)/“Low”アクティブ
P113 / GTIOC2A	PWM 出力 (Vp)/“Low”アクティブ
P111 / GTIOC3A	PWM 出力 (Wp)/“Low”アクティブ
P414 / GTIOC0B	PWM 出力 (Un)/“High”アクティブ
P114 / GTIOC2B	PWM 出力 (Vn)/“High”アクティブ
P112 / GTIOC3B	PWM 出力 (Wn)/“High”アクティブ
P503 / GTETRGC	過電流検出時の PWM 緊急停止入力

2.2.2 周辺機能

サンプルソフトウェアで使用する周辺機能一覧を表 2-4 に示します。

表 2-4 周辺機能対応表

周辺機能	リソース	用途
12 ビット A/D コンバータ	AN000, AN001, AN002, AN005, AN017	・回転速度指令値入力 ・各 U／V／W 相電流測定 ・インバータ母線電圧測定
AGT	AGT0	500 [µs]インターバルタイマ
GPT	CH0, CH2, CH3	相補 PWM 出力
POEG	Group C	PWM 出力端子をハイインピーダンス状態にし、PWM 出力を停止

(1). 12 ビット A/D コンバータ(ADC12)

U 相電流(Iu)、V 相電流(Iv)、W 相電流(Iw)、およびインバータ母線電圧(V_{dc})と回転速度指令値(VR)を、「シングルスキャンモード」で測定します(ハードウェアトリガを使用)。

A/D 変換は GPT のアンダーフロー(PWM の谷)と連動して動作させています。

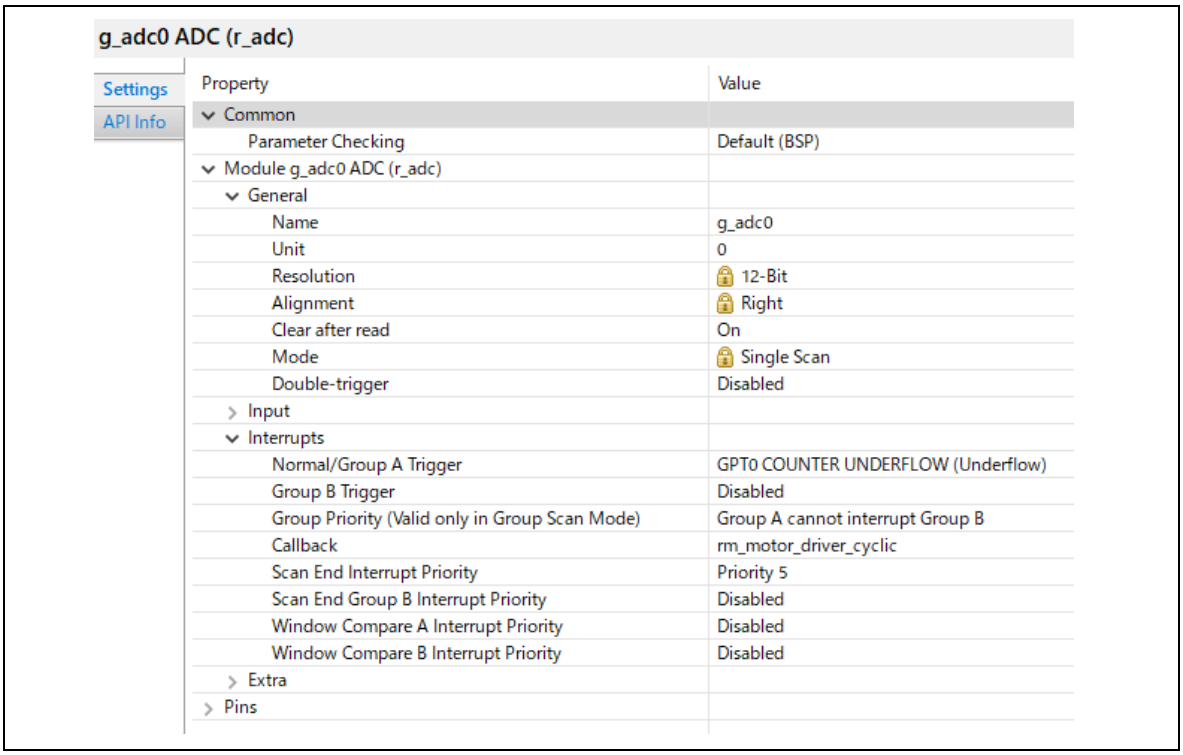


図 2-2 ADC ドライバの FSP コンフィグレーション (FSP3.5.0)

(2). 低消費電力非同期汎用タイマ（AGT）

500 [µs]インターバルタイマとして使用します。

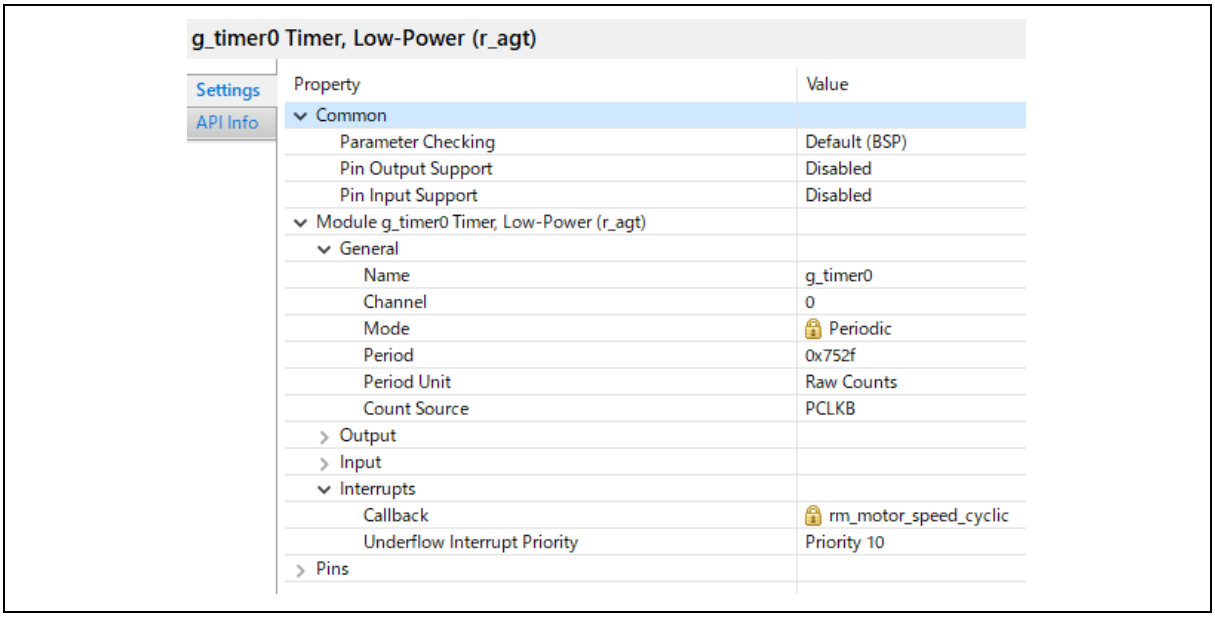


図 2-3 AGT ドライバの FSP コンフィグレーション (FSP3.5.0)

(3). 汎用 PWM タイマ（GPT）

チャネル 0、2、3 の相補 PWM 出力動作モードを使用して、デッドタイム付きの出力を行います。

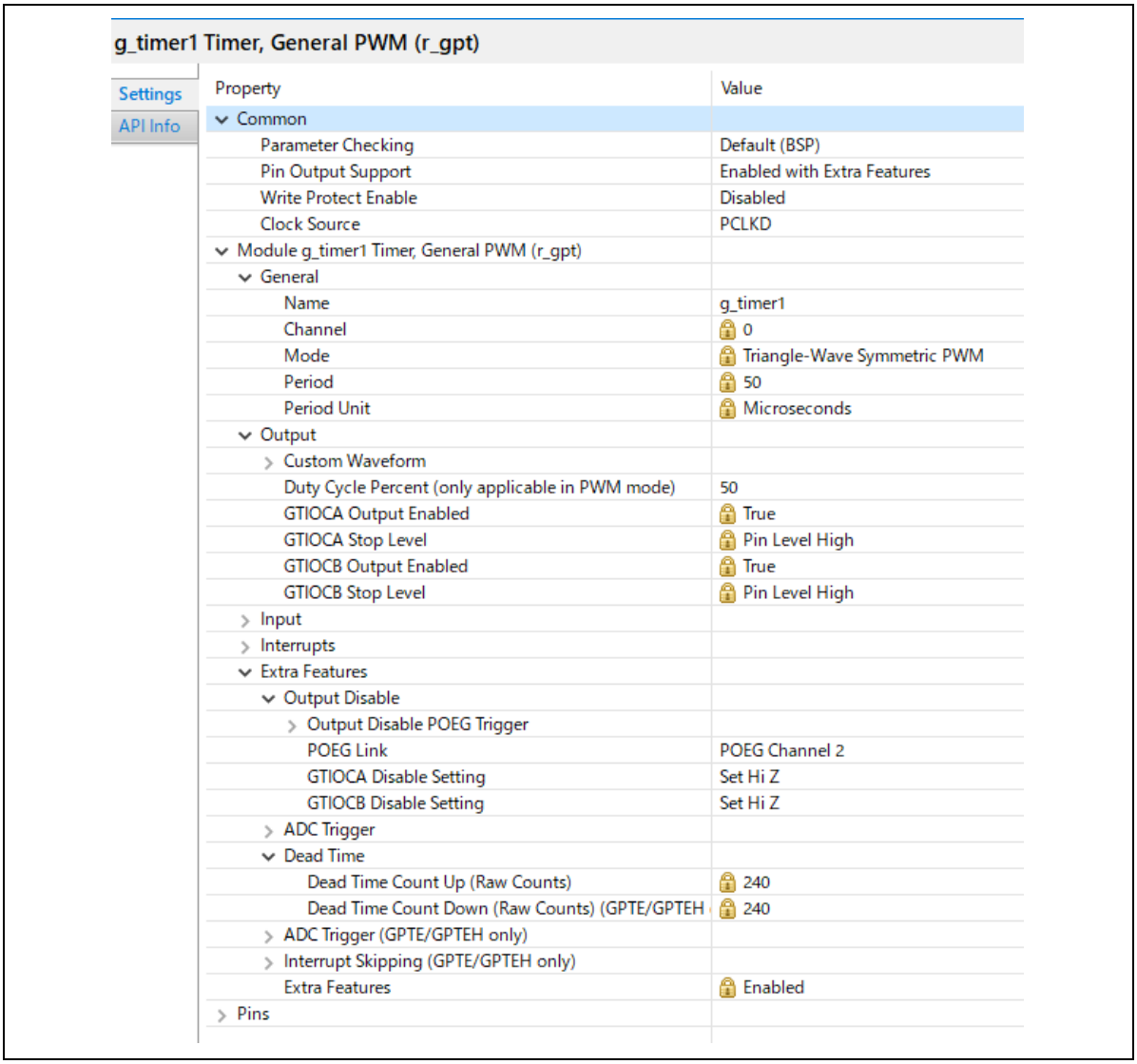


図 2-4 GPT ドライバの FSP コンフィグレーション(FSP3.5.0)

(4). GPT 用ポートアウトプットイネーブル（POEG）

過電流検出時（GTETRGC 端子の Low レベル検出時）は PWM 出力端子をハイインピーダンス状態にします。

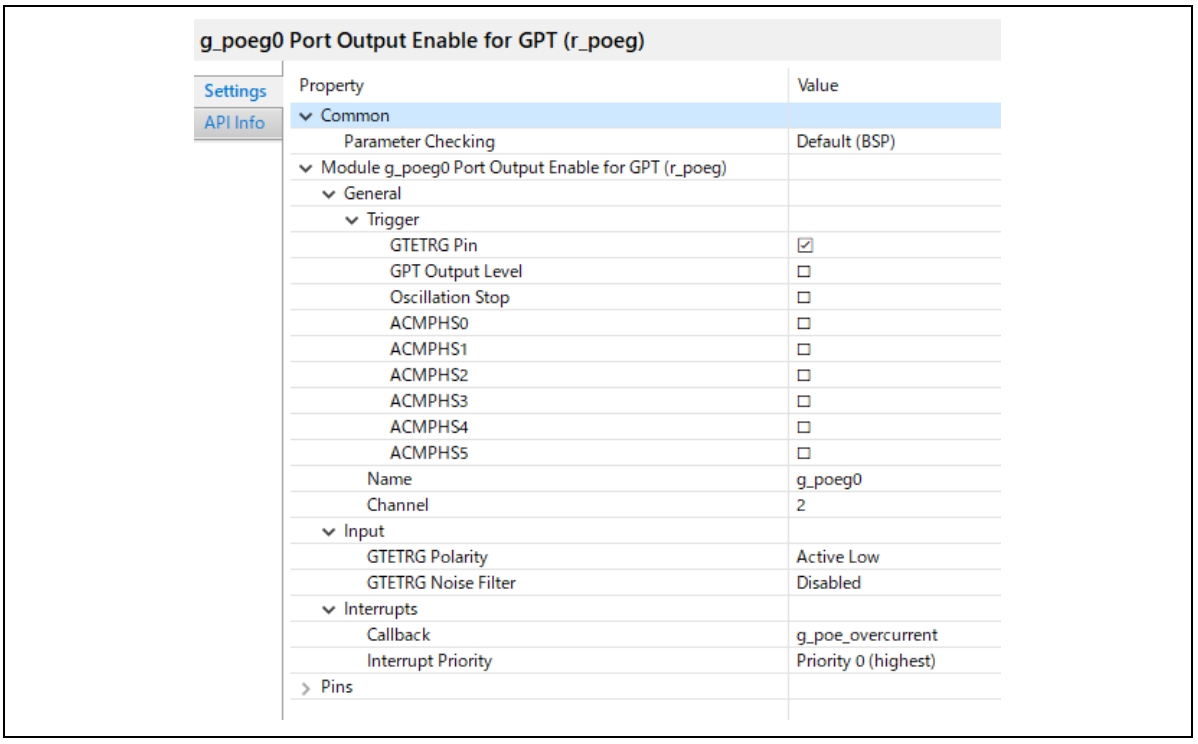


図 2-5 POEG ドライバの FSP コンフィグレーション(FSP3.5.0)

2.3 ソフトウェア構成

2.3.1 ソフトウェア・ファイル構成

サンプルソフトウェアのフォルダとファイル構成を下記に示します。

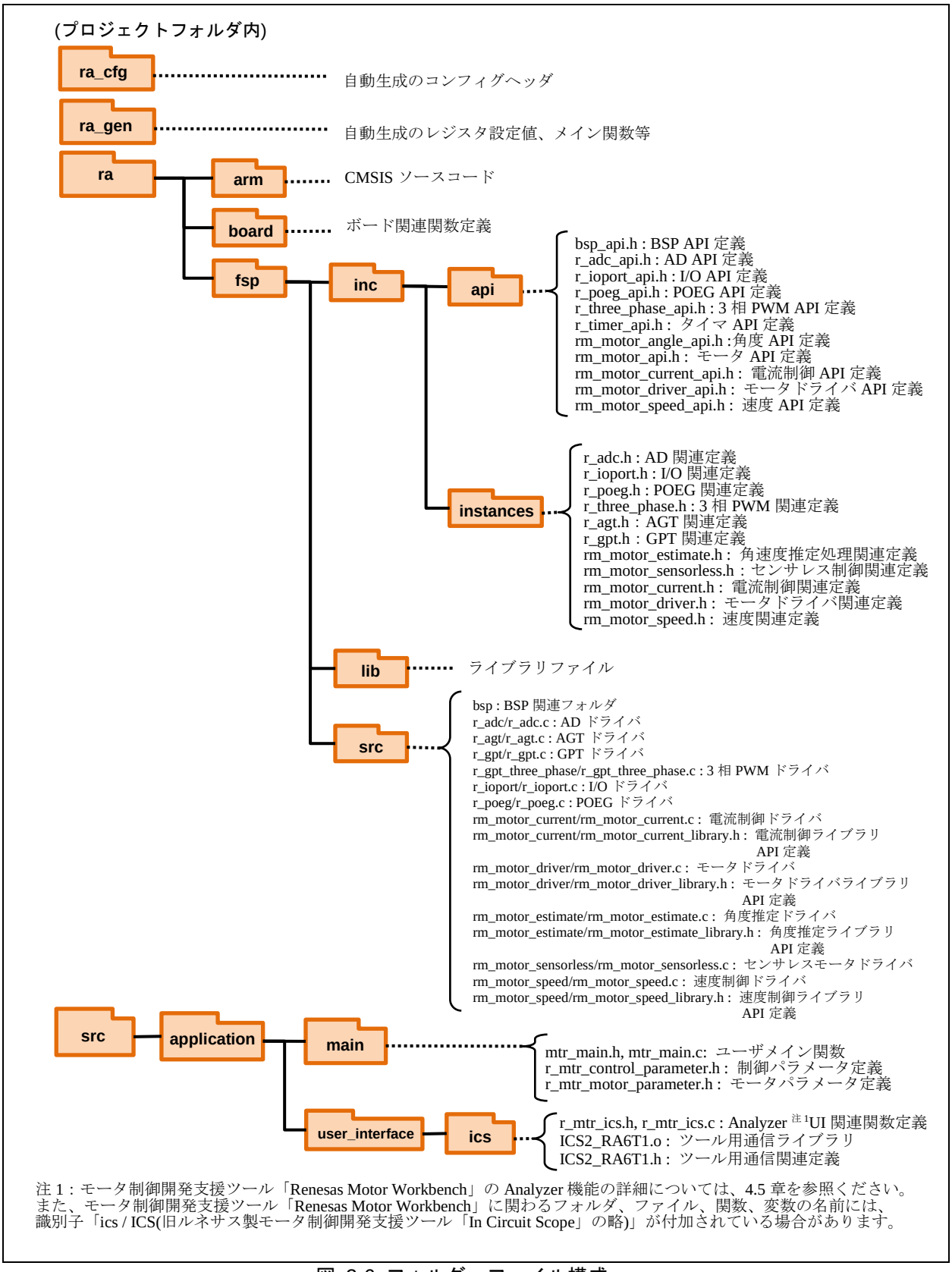


図 2-6 フォルダ・ファイル構成

2.3.2 モジュール構成

サンプルソフトウェアのモジュール構成を図 2-7 に示します。

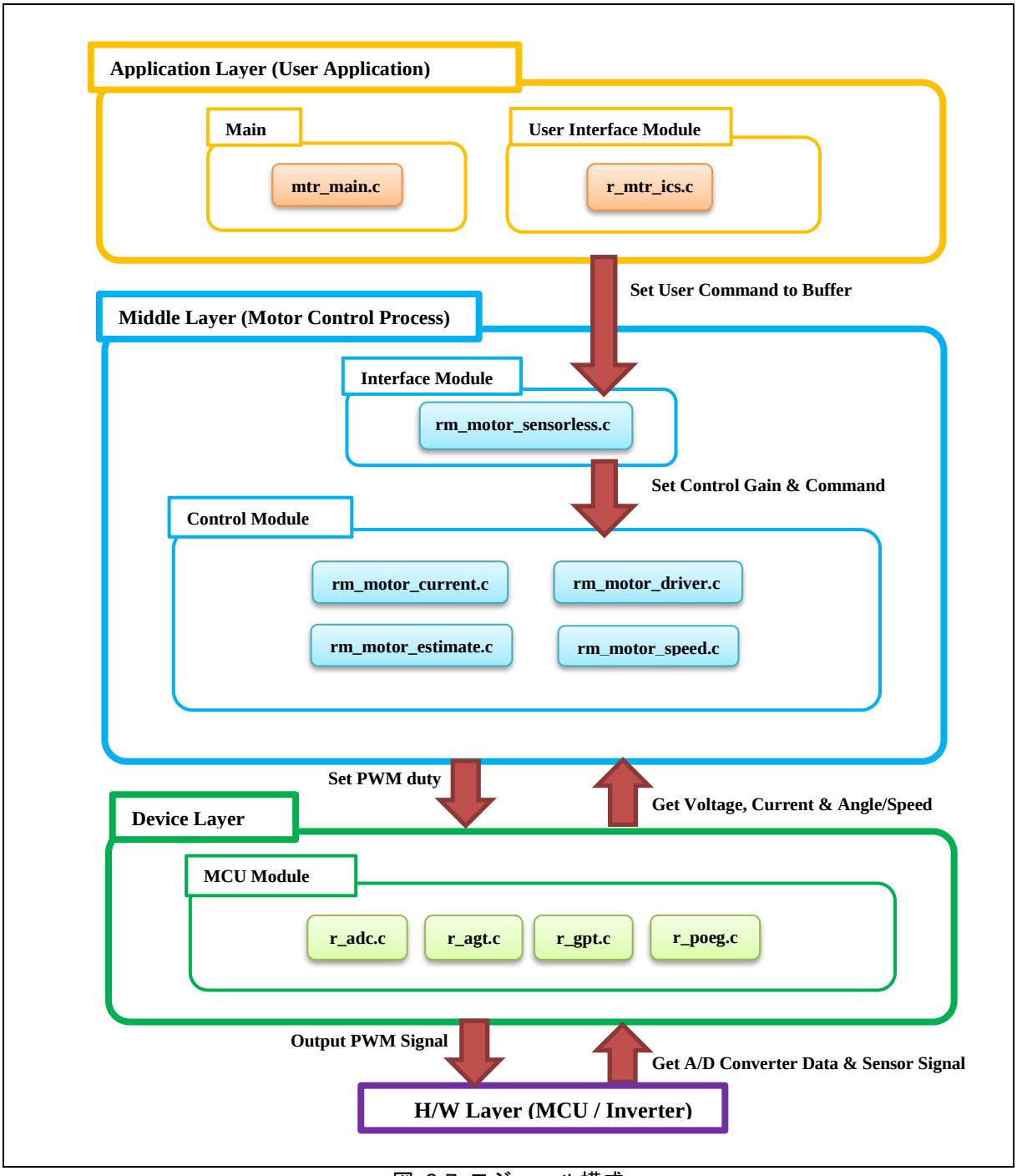


図 2-7 モジュール構成

2.4 ソフトウェア仕様

サンプルソフトウェアの基本仕様を下記に示します。センサレスベクトル制御の詳細に関しては「永久磁石同期モータのセンサレスベクトル制御 (アルゴリズム編)」(R01AN3786)を参照してください。

表 2-5 センサレス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	
回転子磁極位置検出	センサレス	
モータ回転開始/停止	SW1 のレベルにより判定 または Renesas Motor Workbench から入力	
入力電圧	DC 24V	
メインクロック周波数	120 [MHz]	
キャリア(PWM)周波数	20 [kHz](キャリア周期：50 [µs])	
デッドタイム	2 [µs]	
制御周期	電流制御 / 位置・速度推定：50 [µs] 速度制御：500 [µs]	
回転速度範囲	CW：0 [rpm] ～ 2650 [rpm] CCW：0 [rpm] ～ 2650 [rpm] ただし、600 [rpm]以下は速度オープンループで駆動	
各制御系固有周波数	電流制御系：300 [Hz] 速度制御系：5 [Hz] 誘起電圧推定系：1000 [Hz] 位置推定系：50 [Hz]	
コンパイラ最適化設定	最適化レベル	Optimize more(-O2) (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(=0.42*sqrt(2)*1.5) [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) 外部からの過電流検出信号(GTETRGC 端子に Low レベルを検出)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	

2.5 割込み優先順位

サンプルソフトウェアで使用している割り込みと優先順位を下記に示します。

表 2-6 割込み優先順位

割込みレベル	優先度	処理
15	Min	
14		
13		
12		
11		
10		500[µs]割り込み処理
9		
8		
7		
6		
5		A/D 変換完了割込み
4		
3		
2		
1		
0	Max	過電流異常割込み

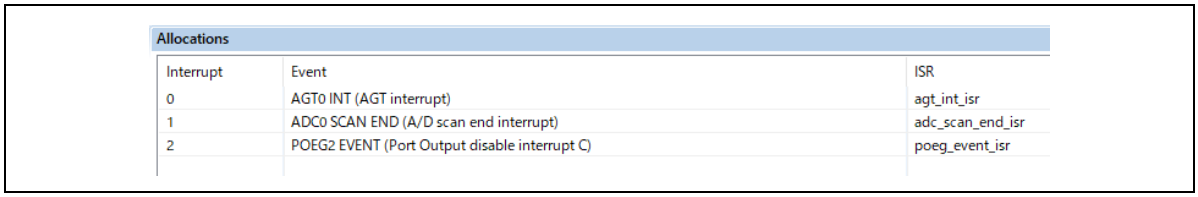


図 2-8 FSP 割り込みコンフィグレーション(FSP3.5.0)

3. 制御ソフトウェア説明

サンプルソフトウェアについて説明します。

3.1 制御内容

3.1.1 モータ起動/停止

モータの起動と停止は、Renesas Motor Workbench からの入力または SW1 からの入力によって制御します。

SW1 には汎用ポートが割り当てられ、”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、”High”レベルのときはモータを停止すると判断します。

3.1.2 A/D 変換

(1). モータ回転速度指令値

モータの回転速度指令値は Renesas Motor Workbench からの入力または 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 VR の値は、以下の表のように、回転速度指令値として使用します。

表 3-1 回転速度指令値の変換比

項目	変換比 (指令値：A/D 変換値)		チャネル
回転速度指令値	CW	0 [rpm]～2650[rpm]：0800H～0FFFH	AN017
	CCW	0 [rpm]～2650[rpm]：07FFH～0000H	

(2). インバータ母線電圧

以下の表のようにインバータ母線電圧を測定します。変調率の算出と過電圧・低電圧検出(異常時は PWM 停止)に使用します。

表 3-2 インバータ母線電圧の変換比

項目	変換比 (インバータ母線電圧：A/D 変換値)	チャネル
インバータ母線電圧	0 [V]～111 [V]：0000H～0FFFH	AN005

(3). U 相、V 相、W 相電流

以下の表のように、U 相、V 相、W 相電流を測定します。

表 3-3 U、V、W 相電流の変換比

項目	変換比 (U 相、V 相、W 相電流：A/D 変換値)	チャネル
U 相、V 相、W 相電流	-12.5 [A]～12.5 [A]：0000H～0E8BH 注 電流値＝(5.0V－2.5V)÷(0.01Ohm×20)＝12.5A このシステムでは電流検出回路を 5V から 3V にレベルシフトしているため、0E8BH が A/D 変換の上限となります。	Iu：AN000 Iv：AN001 Iw：AN002

【注】 A/D 変換特性の詳細に関しては、「RA6T1 グループ ユーザーズマニュアル ハードウェア編」を参照してください。

3.1.3 変調

サンプルソフトウェアでは、モータへの入力電圧はパルス幅変調（以降、PWM）によって生成し、PWM波形は三角波比較法によって生成します。

(1). 三角波比較法

指令値電圧を実際に出力する方法の一つとして、キャリア波形（三角波）と指令値電圧波形を比較する事で出力電圧のパルス幅を決める三角波比較法があります。指令値電圧がキャリア波電圧より大きければスイッチをオン、小さければオフにする事で、正弦波状の指令値電圧を擬似的に出力する事が出来ます。

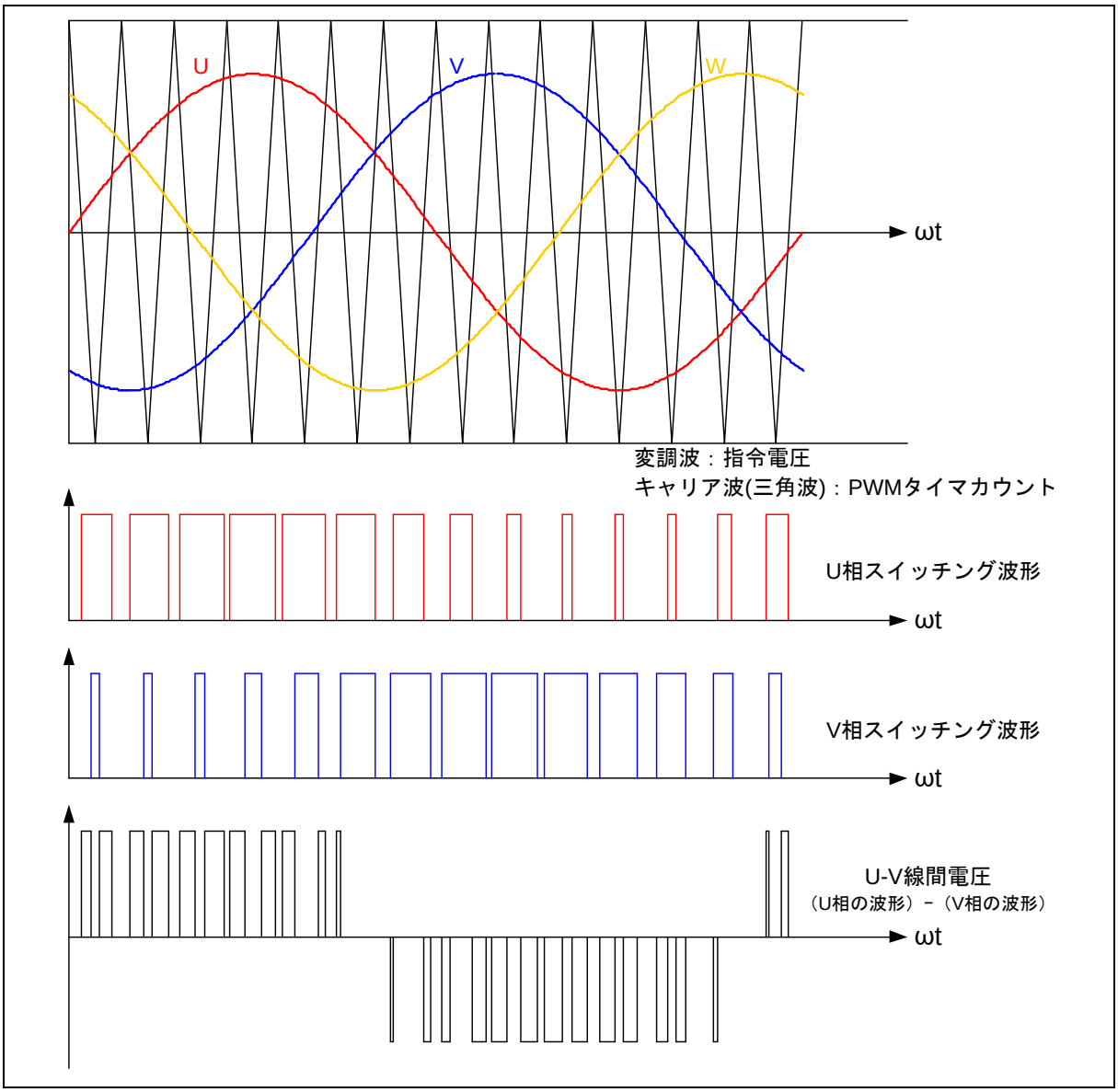


図 3-1 三角波比較法の概念図

図 3-2 のように、出力電圧パルスのキャリア波に対する割合をデューティと呼びます。

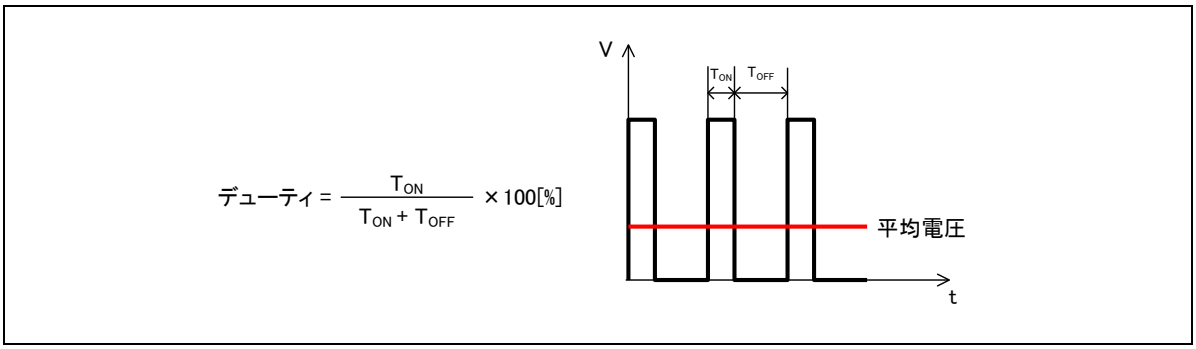


図 3-2 デューティの定義

また、変調率 m を以下のように定義します。

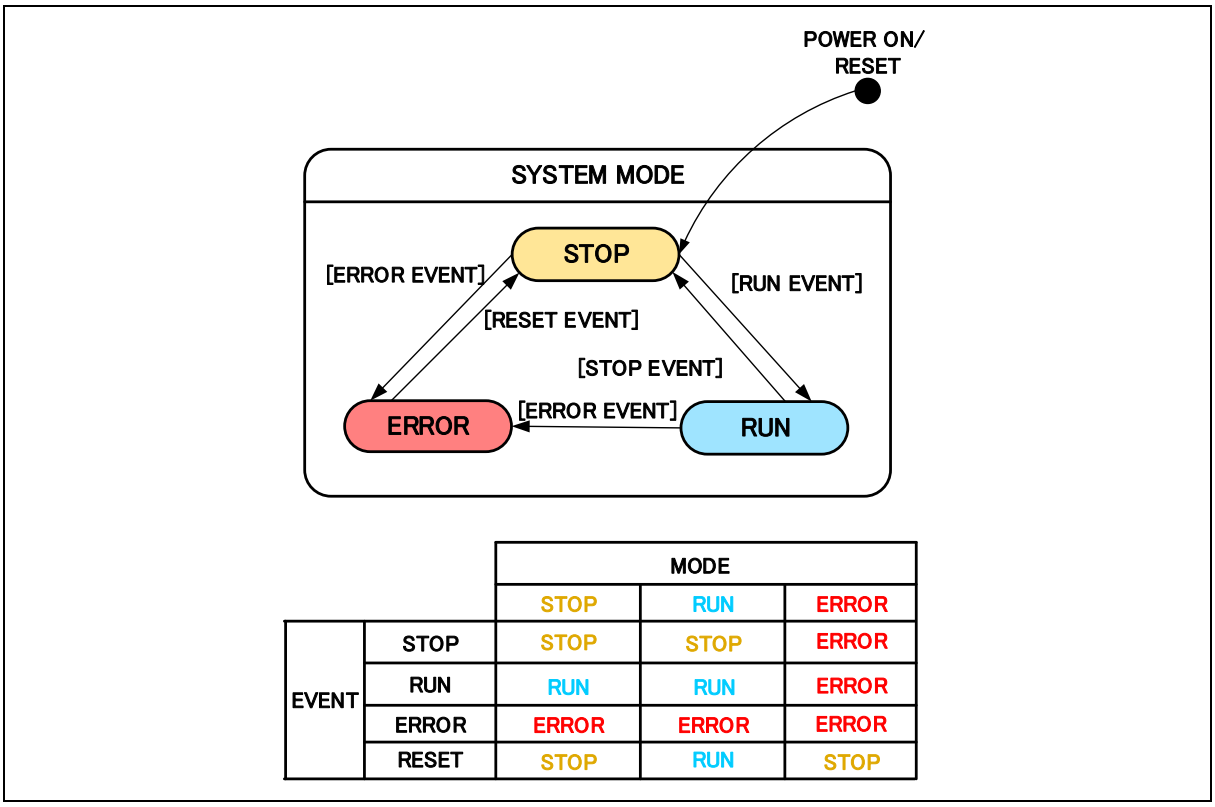
$$m = \frac{V}{E}$$

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に反映させることで所望の制御を行います。

3.1.4 状態遷移

図 3-3 にサンプルソフトウェアにおける状態遷移図を示します。サンプルソフトウェアでは、「SYSTEM MODE」により状態を管理します。



		MODE		
		STOP	RUN	ERROR
EVENT	STOP	STOP	STOP	ERROR
	RUN	RUN	RUN	ERROR
	ERROR	ERROR	ERROR	ERROR
	RESET	STOP	RUN	STOP

図 3-3 センサレス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) 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 3-3 中の表の様に遷移します。各 EVENT の発生要因は下記となります。

表 3-4 EVENT 一覧

イベント名	発生要因
STOP	ユーザー操作により発生します
RUN	ユーザー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	ユーザー操作により発生します

3.1.5 始動方法

サンプルソフトウェアの始動制御内容を図 3-4 に示します。d 軸電流、q 軸電流、速度それぞれの指令値を管理するフラグによってモードをコントロールしています。

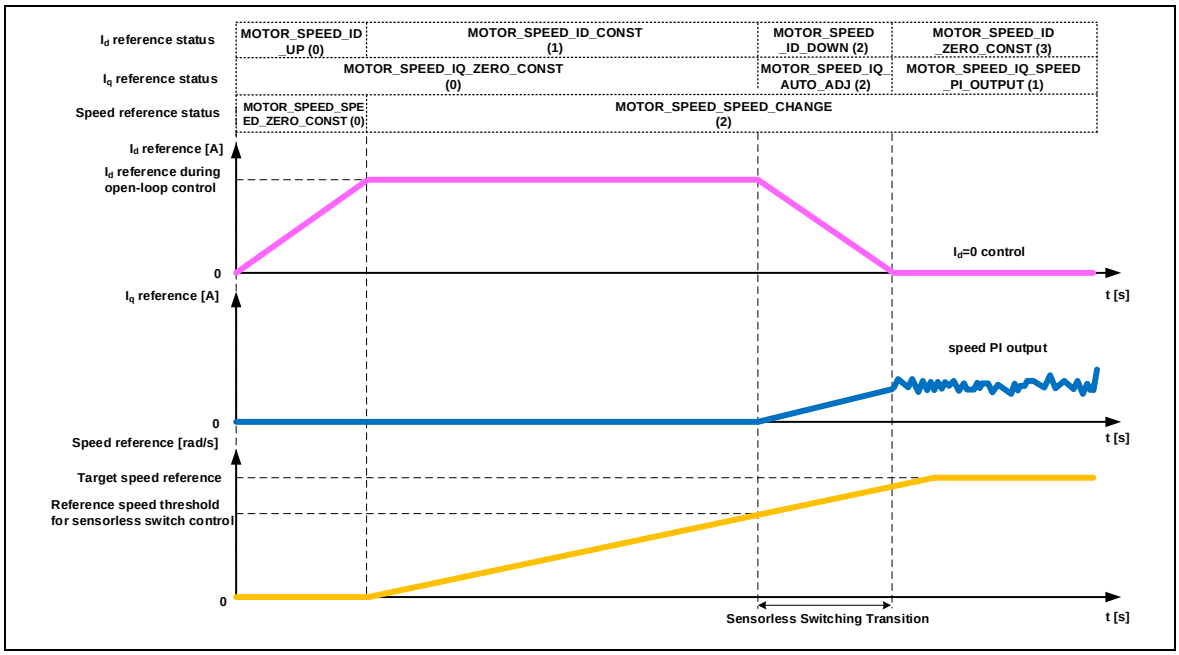


図 3-4 センサレスベクトル制御ソフトウェアの始動制御内容

3.1.6 システム保護機能

サンプルソフトウェアは、以下のエラー状態を持ち、それぞれの場合に緊急停止する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 3-5 を参照してください。

・過電流エラー

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 3-5 各システム保護機能設定値

エラー	閾値	
過電流エラー	過電流リミット値 [A]	0.89
	監視周期 [μs]	50
過電圧エラー	過電圧リミット値 [V]	28
	監視周期 [μs]	50
低電圧エラー	低電圧リミット値 [V]	14
	監視周期 [μs]	50
回転速度エラー	速度リミット値 [rpm]	3000
	監視周期 [μs]	50

3.1.7 AD トリガ

AD トリガとスキャングループのタイミングを示します。

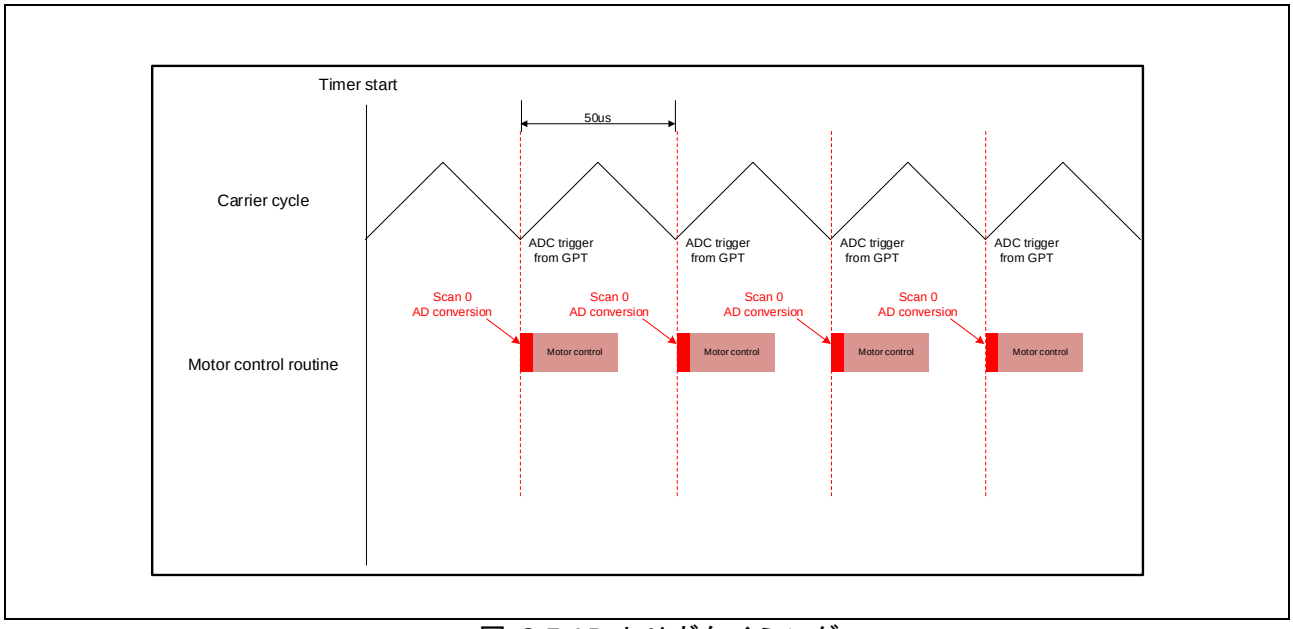


図 3-5 AD トリガタイミング

3.2 センサレスベクトル制御ソフトウェア関数仕様

サンプルソフトウェアにおける制御処理は、主に 50[μs]周期割り込みと、500[μs]周期割り込みの 2 つの割り込みにより構成されています。図 3-6、図 3-7 にあるように、赤破線部が 50[μs]周期毎に実行される処理で、青破線部が 500[μs]周期毎に実行される処理になります。

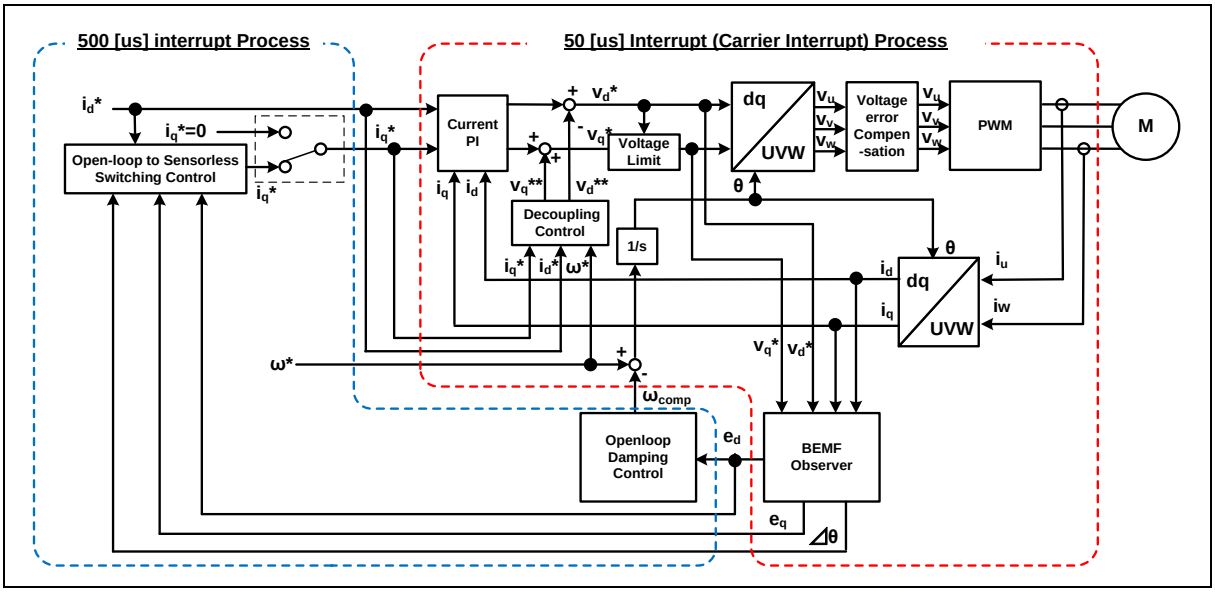


図 3-6 センサレスベクトル制御概略ブロック図(オープンループ制御時)

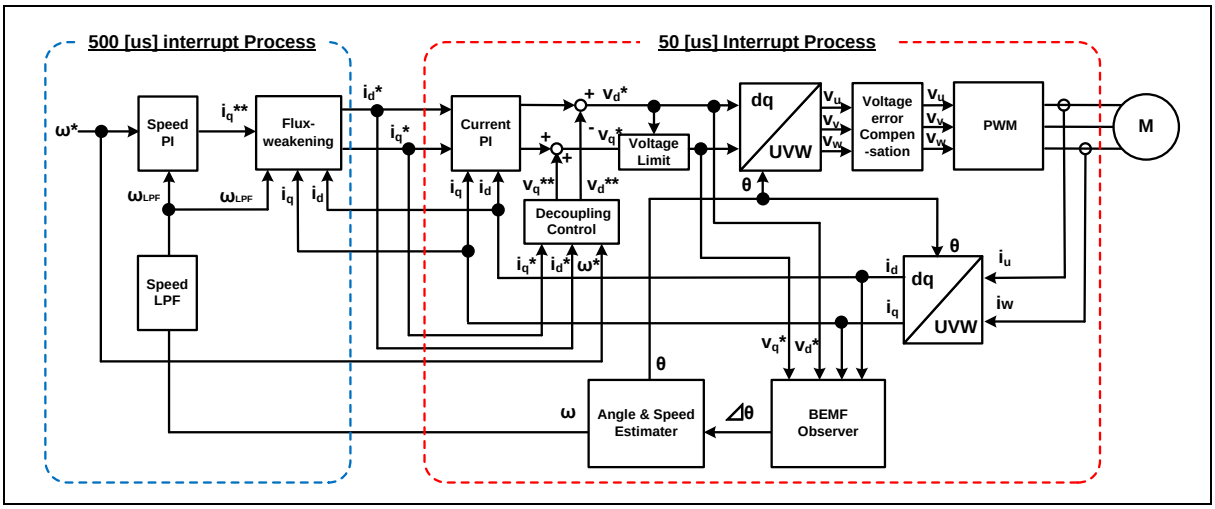


図 3-7 センサレスベクトル制御概略ブロック図(センサレス制御時)

ここでは、2 つの割り込み関数と、各割り込み周期毎に実行される関数についての仕様を表 3-6～表 3-11 にまとめます。また各表には、センサレスベクトル制御における主要な関数のみ記載しています。各表に記載のない関数の詳細については、ソースコードを参照ください。

表 3-6 50[μs]周期割り込み関数内実行関数一覧(1/4)

ファイル名	関数名	処理概要
mtr_main.c	mtr_callback_event 入力：(motor_sensorless_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	センサレス制御コールバック関数
rm_motor_sensorless.c	rm_motor_sensorless_current_callback 入力：(motor_current_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	速度制御の出力を電流制御の入力に設定
	RM_MOTOR_SENSORLESS_ErrorCheck 入力：(motor_ctrl_t * const) p_ctrl / モータ制御構造体ポインタ (uint16_t * const) p_error / モータエラー情報 出力：fsp_err_t / 関数実行結果	エラーチェック
	rm_motor_sensorless_copy_speed_current 入力：(motor_speed_output_t *) st_output / 速度制御出力データ (motor_current_input_t *) st_input / 電流制御入力データ 出力：なし	速度出力データを電流入力データにコピー
rm_motor_driver.c	rm_motor_driver_cyclic 入力：(adc_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	モータドライバコールバック関数
	rm_motor_driver_current_get 入力：(motor_driver_instance_ctrl_t *) p_ctrl / モータドライバインスタンス 出力：なし	A/D 変換データを取得（相電流と母線電圧）
	RM_MOTOR_DRIVER_FlagCurrentOffsetGet 入力：(motor_driver_ctrl_t * const) p_ctrl / モータドライバ制御構造体ポインタ (uint8_t * const) p_flag_offset / 電流オフセット検出完了フラグ 出力：fsp_err_t / 関数実行結果	電流オフセット値を測定
	RM_MOTOR_DRIVER_PhaseVoltageSet 入力：(motor_driver_ctrl_t * const) p_ctrl / モータドライバ制御構造体ポインタ (float const) u_voltage / U 相電圧 (float const) v_voltage / V 相電圧 (float const) w_voltage / W 相電圧 出力：fsp_err_t / 関数実行結果	PWM デューティ計算用の相電圧データを設定
	rm_motor_driver_modulation 入力：(motor_driver_instance_ctrl_t *) p_ctrl / モータドライバインスタンス 出力：なし	PWM 変調の実行

表 3-7 50[µs]周期割り込み関数内実行関数一覧(2/4)

ファイル名	関数名	処理概要
rm_motor_driver.c	rm_motor_driver_mod_run 入力：(motor_driver_modulation_t *) p_mod / 変調データ (const float *) p_f4_v_in / 入力三相電圧 (float *) p_f4_duty_out / 出力デューティ 出力：なし	入力三相電圧からデューティサイクルを計算
	rm_motor_driver_set_uvw_duty 入力：(motor_driver_instance_ctrl_t *) p_ctrl / モータドライバインスタンス (float) f_duty_u / U 相デューティ (float) f_duty_v / V 相デューティ (float) f_duty_w / W 相デューティ 出力：fsp_err_t / 関数実行結果	PWM デューティ設定
	RM_MOTOR_DRIVER_CurrentGet 入力：(motor_driver_ctrl_t * const) p_ctrl / モータドライバコントロールブロック (motor_driver_current_get_t * const) p_current_get / 取得データ 出力：fsp_err_t / 関数実行結果	相電流、Vdc、Va_max データを取得
rm_motor_current.c	rm_motor_current_cyclic 入力：(motor_driver_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	電流制御周期動作
	RM_MOTOR_CURRENT_ParameterSet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (motor_current_input_current_t const * const) p_st_input / インプット電流データ 出力：fsp_err_t / 関数実行結果	電流制御入力データを設定
	RM_MOTOR_CURRENT_CurrentSet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (motor_current_input_current_t const * const) p_st_current / インプット電流データ (motor_current_input_voltage_t const * const) p_st_voltage / インプット電圧データ 出力：fsp_err_t / 関数実行結果	d/q 軸の電流と電圧のデータを設定
	RM_MOTOR_CURRENT_CurrentGet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (float * const) p_id / 取得した d 軸電流へのポインタ (float * const) p_iq / 取得した q 軸電流へのポインタ 出力：fsp_err_t / 関数実行結果	d/q 軸電流を取得
	motor_current_transform_uvw_dq_abs 入力：(const float) f_angle / 回転角度 (const float *) f_uvw / UVW 相ポインタ (float *) f_dq / dq 軸ポインタ 出力：なし	UVW → dq 座標変換(絶対変換)

表 3-8 50[µs]周期割り込み関数内実行関数一覧(3/4)

ファイル名	関数名	処理概要
rm_motor_current.c	motor_current_angle_cyclic 入力：(motor_current_instance_t *) p_instance / 電流インスタンス 出力：なし	電流制御の周期動作における角度/速度プロセス
	RM_MOTOR_CURRENT_SpeedPhaseSet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (float const) speed_rad / 回転速度 (float const) phase_rad / 位相 出力：fsp_err_t / 関数実行結果	電流速度とロータ位相データを設定
	RM_MOTOR_CURRENT_CurrentReferenceSet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (float const) speed_rad / 回転速度 (float const) phase_rad / 位相 出力：fsp_err_t / 関数実行結果	電流指令値を設定
	RM_MOTOR_CURRENT_PhaseVoltageGet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流制御構造体ポインタ (motor_current_get_voltage_t * const) p_voltage / 取得電圧データ 出力：fsp_err_t / 関数実行結果	設定された相電圧を取得
	motor_current_pi_calculation 入力：(motor_current_instance_ctrl_t *) p_instance / 電流インスタンスコントロール 出力：なし	電流ベクトルコマンドと実際の電流ベクトルから出力電圧ベクトルを計算
	motor_current_pi_control 入力：(motor_current_pi_params_t *) pi_ctrl / PI 制御パラメータ 出力：float / PI 制御出力値	PI 制御
	motor_current_limit_abs 入力：(float) f4_value / ターゲットの値 (float) f4_limit_value / 制限値 出力：float / 制限した値	絶対値の制限
	motor_current_decoupling 入力：(motor_current_instance_ctrl_t *) p_ctrl / 電流制御インスタンス (float) f_speed_rad / 回転速度 (const motor_current_motor_parameter_t *) p_mtr / モータ電流パラメータ 出力：なし	非干渉制御
	motor_current_voltage_limit 入力：(motor_current_instance_ctrl_t *) p_ctrl / 電流制御インスタンス 出力：なし	電圧ベクトルの制限

表 3-9 50[μs]周期割り込み関数内実行関数一覧(4/4)

ファイル名	関数名	処理概要
rm_motor_current.c	motor_current_transform_dq_uvw_abs 入力：(const float) f_angle / 回転角度 (const float *) f_dq / dq 軸ポインタ (float *) f_uvw / UVW 相ポインタ 出力：なし	dq → UVW 座標変換(絶対変換)
librm_motor_current.a	rm_motor_voltage_error_compensation_main 入力：(motor_currnt_voltage_compensation_t *) st_volt_comp / 電圧誤差補償データ (float *) p_f4_v_array / リファレンス電圧 (float *) p_f4_i_array / リファレンス電流 (float) f4_vdc / 母線電圧 出力：なし	電圧誤差補償
rm_motor_estimate.c	RM_MOTOR_ESTIMATE_FlagPiCtrlSet 入力：(motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (uint32_t const) flag_pi / PI 制御実行中フラグ 出力：fsp_err_t / 関数実行結果	PI 制御実行のフラグを設定
	RM_MOTOR_ESTIMATE_SpeedSet 入力：(motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (float const) speed_ctrl / リファレンス回転速度 (float const) damp_speed / ダンピング回転速度 出力：fsp_err_t / 関数実行結果	速度情報を設定
	RM_MOTOR_ESTIMATE_CurrentSet 入力：(motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (motor_angle_current_t * const) p_st_current / 電流データ (motor_angle_voltage_reference_t * const) p_st_voltage / 電圧データ 出力：fsp_err_t / 関数実行結果	d/q 軸の電流データと電圧指令値を設定
	RM_MOTOR_ESTIMATE_AngleSpeedGet 入力：(motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (float * const) p_angle / 角度データ (float * const) p_speed / 速度データ (float * const) p_phase_err / 位相誤差データ 出力：fsp_err_t / 関数実行結果	ロータの角度と回転速度を取得
	RM_MOTOR_ESTIMATE_EstimatedComponentGet 入力：(motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (float * const) p_ed / 推定 d 軸成分 (float * const) p_eq / 推定 q 軸成分 出力：fsp_err_t / 関数実行結果	d/q 軸成分の推定値を取得
r_gpt_three_phase.c	R_GPT_THREE_PHASE_DutyCycleSet 入力：(three_phase_ctrl_t * const) p_ctrl / 三相タイマ制御構造体ポインタ (three_phase_duty_cycle_t * const) p_duty_cycle / 設定デューティサイクル 出力：fsp_err_t / 関数実行結果	各タイマのデューティサイクル設定

表 3-10 500[μs]周期割り込み関数内実行関数一覧(1/2)

ファイル名	関数名	処理概要
mtr_main.c	mtr_callback_event 入力：(motor_sensorless_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	センサレス制御コールバック関数
	get_vr1 入力：なし 出力：uint16_t / 変換値	VR1 の A/D 変換値取得
rm_motor_current.c	RM_MOTOR_CURRENT_ParameterGet 入力：(motor_current_ctrl_t * const) p_ctrl / 電流コントロールブロック (motor_current_input_current_t const * const) p_st_input / インプット電流データ 出力：fsp_err_t / 関数実行結果	電流制御から速度制御入力データを取得
rm_motor_sensorless.c	rm_motor_sensorless_speed_callback 入力：(motor_speed_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	速度制御コールバック関数
	rm_motor_sensorless_copy_current_speed 入力：(motor_current_output_t *) st_output / 電流出力データ (motor_speed_input_t *) st_input / 速度入力データ 出力：なし	電流出力データを速度入力データにコピー
rm_motor_speed.c	rm_motor_speed_cyclic 入力：(timer_callback_args_t *) p_args / コールバック関数パラメータデータ 出力：なし	速度制御の周期動作（タイマ割り込み時の呼び出し）
	RM_MOTOR_SPEED_ParameterSet 入力：(motor_speed_ctrl_t * const) p_ctrl / モータ速度コントロールブロック (motor_speed_input_t const * const) p_st_input / 速度入力パラメータ 出力：fsp_err_t / 関数実行結果	速度入力パラメータを設定
	RM_MOTOR_SPEED_SpeedControl 入力：(motor_speed_ctrl_t * const) p_ctrl / モータ速度コントロールブロック 出力：fsp_err_t / 関数実行結果	d/q 軸電流指令値を計算（速度制御のメインプロセス）
	rm_motor_speed_set_speed_ref 入力：(motor_speed_instance_ctrl_t *) p_ctrl / モータ速度インスタンス 出力：float / リファレンス速度	速度指令値を更新
	rm_motor_speed_set_iq_ref 入力：(motor_speed_instance_ctrl_t *) p_ctrl / モータ速度インスタンス 出力：float / q 軸リファレンス電流	q 軸の電流指令値を更新
	rm_motor_speed_set_id_ref 入力：(motor_speed_instance_ctrl_t *) p_ctrl / モータ速度インスタンス 出力：float / d 軸リファレンス電流	d 軸の電流指令値を更新
	RM_MOTOR_SPEED_ParameterGet 入力：(motor_speed_ctrl_t * const) p_ctrl / モータ速度コントロールブロック (motor_speed_output_t * const) p_st_output / 速度出力データ 出力：fsp_err_t / 関数実行結果	速度制御の出力パラメータを取得

表 3-11 500[μs]周期割り込み関数内実行関数一覧(2/2)

ファイル名	関数名	処理概要
librm_motor_speed.a	rm_motor_speed_first_order_lpf 入力：(motor_speed_lpf_t *) p_lpf / 一次 LPF 構造体ポインタ (float) f_input / 入力データ 出力：float / フィルター後のデータ	一次 LPF 処理
	rm_motor_speed_fluxwkn_set_vamax 入力：(motor_speed_flux_weakening_t *) p_fluxwkn / 弱め磁束構造体ポインタ (float) f4_va_max / 最大電圧ベクトル 出力：なし	最大電圧ベクトルを設定
	rm_motor_speed_fluxwkn_run 入力：(motor_speed_flux_weakening_t *) p_fluxwkn / 弱め磁束構造体ポインタ (float) f4_speed_rad / モータ電気速度 (const float *) p_f4_idq / d/q 軸電流 (float *) p_f4_idq_ref / d/q 軸リファレンス電流 出力：なし	弱め磁束制御実行

3.3 Contents of control

3.3.1 Configuration Options

モータ用センサレスベクトル制御モジュールの構成オプションは、RA Configurator を使用して構成できます。 変更されたオプションは、コードの生成時に common_data.c/h、及び hal_data.c/h に自動的に反映されます。 オプション名と設定値は、次の表 3-12 構成オプションに記載しています。

表 3-12 構成オプション

Configuration Options (rm_motor_sensorless.h)	
オプション名	内容
Limit of over current (A) Initial: 0.42F	相電流がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over voltage (V) Initial: 28.0F	母線電圧がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over speed (rpm) Initial: 3000.0F	回転速度がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over speed (rpm) Initial: 14.0F	母線電圧がこの値を下回ると、PWM 出力ポートがオフに設定されます。

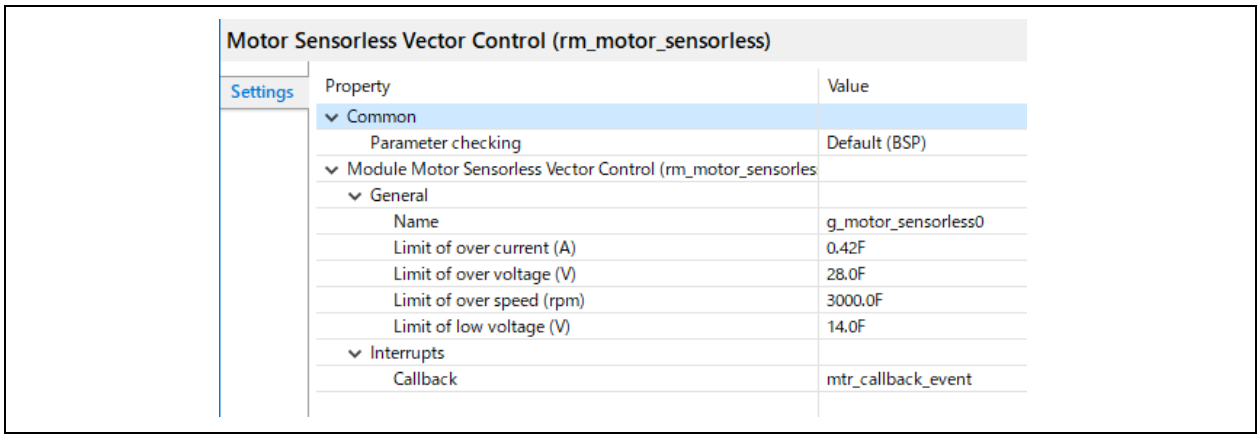


図 3-8 センサレスベクトル制御の FSP コンフィグレーション(FSP3.5.0)

3.3.2 Configuration Options for included modules

モータ用センサレスベクトル制御モジュールには、以下のモジュールが含まれます。

- Current Module
- Speed Module
- Angle Module
- Driver Module

また、これらのモジュールには、センサレスベクトル制御モジュールと同じ各構成パラメータがあります。オプション名と設定値を以下の表に示します。

表 3-13 構成オプション

Configuration Options (rm_motor_current.h)	
オプション名	内容
Shunt type Initial: 2shunt	シャント抵抗をいくつ用いて電流検出を行うかを選択します。
Input voltage (V) Initial: 24.0F	入力電圧 [V]
Voltage error compensation Initial: Enable	電圧誤差補償の有効／無効を選択します。
Design Parameter | Current PI loop omega Initial: 300.0F	電流制御系固有周波数 [Hz]
Design Parameter | Current PI loop zeta Initial: 1.0F	電流制御系減衰係
Motor Parameter | Pole pairs Initial: 2	極対数
Motor Parameter | Resistance (ohm) Initial: 8.5F	抵抗 [ohm].
Motor Parameter | Inductance of d-axis (H) Initial: 0.0045F	d 軸インダクタンス [H].
Motor Parameter | Inductance of d-axis (H) Initial: 0.0045F	q 軸インダクタンス [H].
Motor Parameter | Permanent magnetic flux (Wb) Initial: 0.02159F	磁束 [Wb].
Motor Parameter | Rotor inertia (kgm^2) Initial: 0.0000028F	イナーシャ [kgm^2].

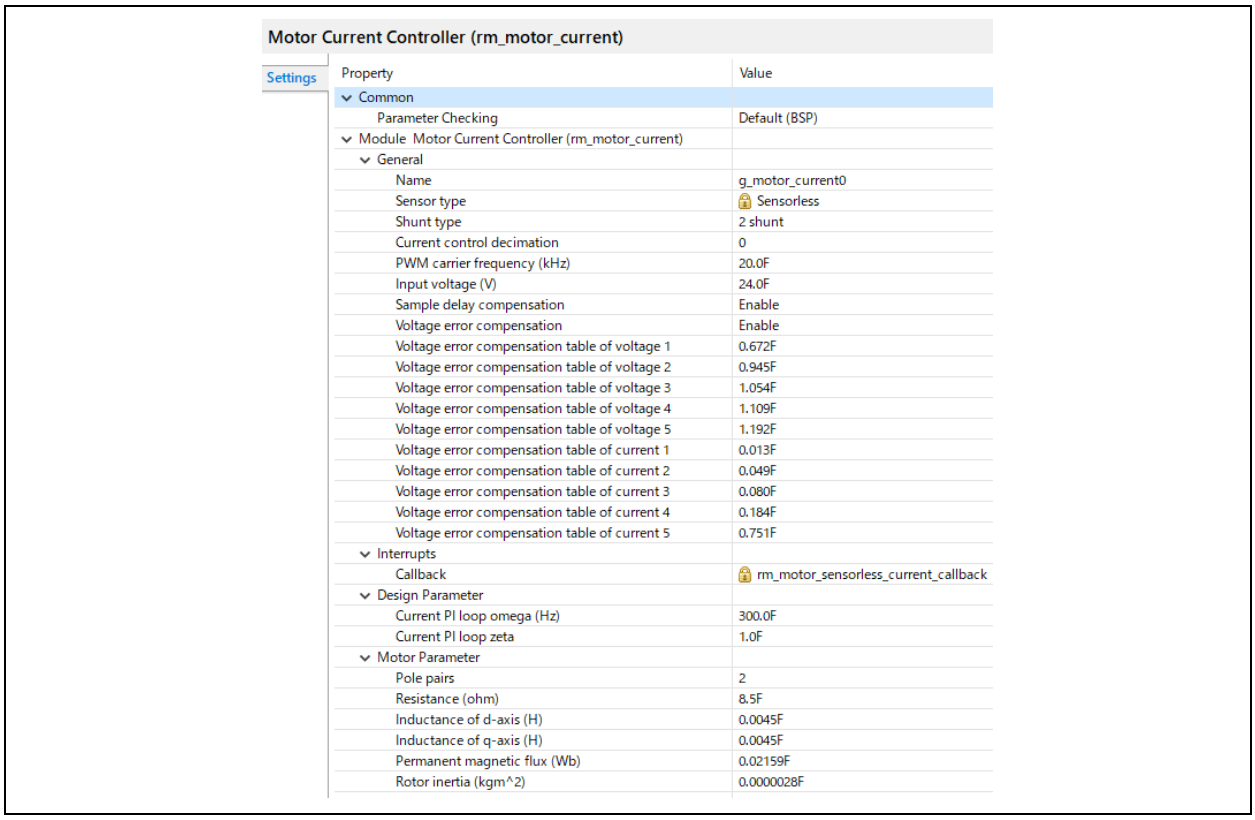


図 3-9 モータ電流コントローラの FSP コンフィグレーション(FSP3.5.0)

表 3-14 構成オプション

Configuration Options (rm_motor_speed.h)	
オプション名	内容
Speed control period (sec) Initial: 0.0005F	速度制御周期[sec]
Step of speed climbing (rpm) Initial: 0.5F	速度変動のステップ値[rpm]。加速と減速で この値によって速度を制御します。
Maximum rotational speed (rpm) Initial: 2650	最大速度 [rpm]
Speed LPF omega Initial: 10.0F	速度 LPF 固有周波数 [Hz]
Speed at Id climbing (rpm) Initial: 500.0F	d 軸電流の増加を制御する閾値[rad / s]。速度がこの値に 達するまで、起動時に d 軸電流を増加させます。
Limit of q-axis current (A) Initial: 0.42F	q 軸電流リミット [A]
Step of speed feedback at open-loop Initial: 0.20F	オープンループ時指令速度ステップ
Open-loop damping Initial: Enable	オープンループダンピング制御の選択
Flux weakening Initial: Disable	弱め磁束制御の選択
Torque compensation for sensorless transition Initial: Enable	センサレス切り替え制御の選択

表 3-15 構成オプション

Configuration Options (rm_motor_speed.h)	
オプション名	内容
Open-Loop | Step of d-axis current climbing Initial: 0.3F	d 軸電流指令値加算ステップ [A/msec]
Open-Loop | Step of d-axis current descending Initial: 0.3F	d 軸電流指令値減算ステップ [A/msec]
Open-Loop | Step of q-axis current descending ratio Initial: 1.0F	q 軸電流指令値減算ステップ [A/msec]
Open-Loop | Reference of d-axis current Initial: 0.3F	オープンループ制御時 d 軸電流指令値 [A]
Open-Loop | Threshold of speed control descending Initial: 600	d 軸電流指令値減算開始速度[rpm]
Open-Loop | Threshold of speed control climbing Initial: 500	d 軸電流指令値加算開始速度 [rpm]
Open-Loop | Period between open-loop to BEMF (sec) Initial: 0.025F	センサレス切り替え処理時間 [s]
Open-Loop | Phase error(degree) to decide sensor-less switch timing Initial: 10.0F	センサレス制御切り替え可能位相誤差(電気角) [deg]
Design parameter | Speed PI loop omega Initial: 5.0F	速度制御系固有周波数 [Hz]
Design parameter | Speed PI loop zeta Initial: 1.0F	速度制御系減衰係数
Design parameter | Estimated d-axis HPF omega Initial: 2.5F	d 軸誘起電圧 HPF カットオフ周波数 [Hz]
Design parameter | Open-loop damping zeta Initial: 1.0F	オープンループダンピング制御減衰係数
Design parameter | Cutoff frequency of phase error LPF Initial: 10.0F	位相誤差 LPF カットオフ周波数 [Hz]
Design parameter | Speed observer omega Initial: 200.0F	速度オブザーバカットオフ周波数 [Hz]
Design parameter | Speed observer zeta Initial: 1.0F	速度オブザーバ減衰係数
Motor Parameter | Pole pairs Initial: 2	極対数
Motor Parameter | Resistance (ohm) Initial: 8.5F	抵抗 [ohm]
Motor Parameter | Inductance of d-axis (H) Initial: 0.0045F	d 軸インダクタンス [H]
Motor Parameter | Inductance of q-axis (H) Initial: 0.0045F	q 軸インダクタンス [H]
Motor Parameter | Permanent magnetic flux (Wb) Initial: 0.02159F	磁束 [Wb]
Motor Parameter | Rotor inertia (kgm^2) Initial: 0.0000028F	イナーシャ [kgm^2]

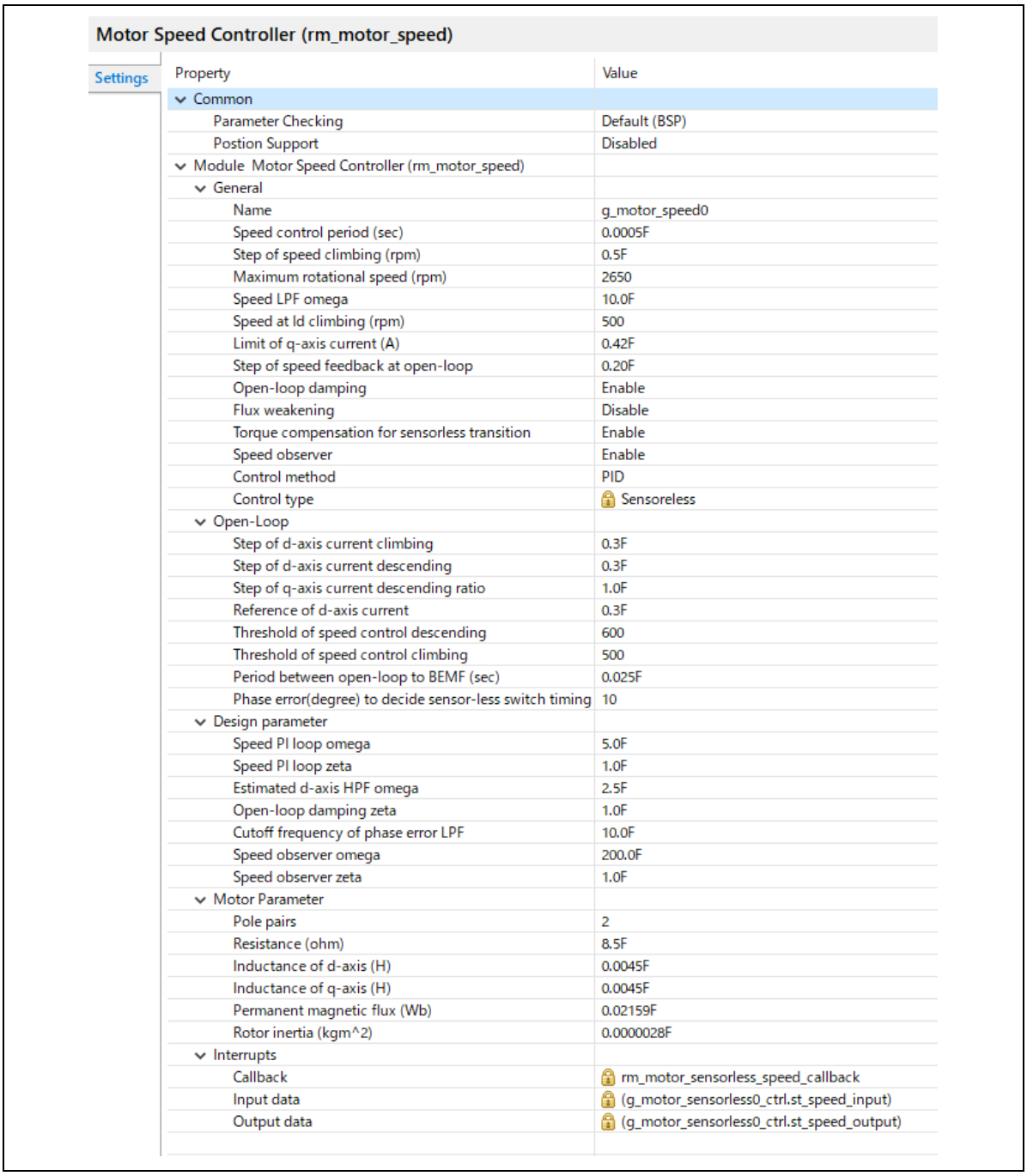


図 3-10 モータスピードコントローラの FSP コンフィグレーション(FSP3.5.0)

表 3-16 構成オプション

Configuration Options (rm_motor_ estimate.h)	
オプション名	内容
Motor Parameter | Pole pairs Initial: 2	極対数
Motor Parameter | Resistance (ohm) Initial: 8.5F	抵抗 [ohm]
Motor Parameter | Inductance of d-axis (H) Initial: 0.0045F	d 軸インダクタンス [H]
Motor Parameter | Inductance of q-axis (H) Initial: 0.0045F	q 軸インダクタンス [H]
Motor Parameter | Permanent magnetic flux (Wb) Initial: 0.02159F	磁束 [Wb]
Motor Parameter | Rotor inertia (kgm^2) Initial: 0.0000028F	イナーシャ [kgm^2]
Open-loop damping Initial: Enable	オープンループダンピング制御の選択
Natural frequency of BEMF observer Initial: 1000.0F	誘起電圧推定系固有周波数[Hz]
Damping ratio of BEMF observer Initial: 1.0F	誘起電圧推定系減衰係数
Natural frequency of PLL Speed estimate loop Initial: 20.0F	位置推定系固有周波数[Hz]
Damping ratio of PLL Speed estimate loop Initial: 1.0F	位置推定系減衰係数
Control period Initial: 0.0005F	速度制御周期 [sec]

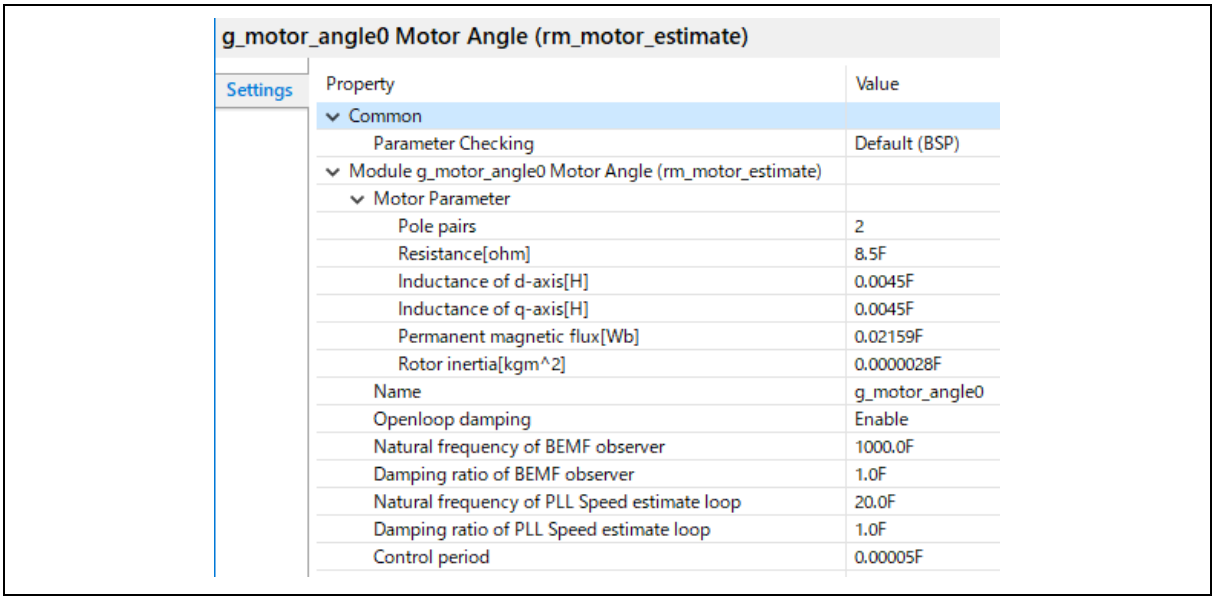


図 3-11 モータアナログドライバの FSP コンフィグレーション(FSP3.5.0)

表 3-17 構成オプション

Configuration Options (rm_motor_driver.h)	
オプション名	内容
Shunt type Initial: 2shunt	シャント抵抗をいくつ使って各相電流の検出を行うかを選択出来ます。
PWM timer frequency (MHz) Initial: 120U	PWM タイマ周波数 [MHz]
PWM carrier period (micro seconds) Initial: 50U	PWM キャリア周期 [Micro seconds]
Dead time (raw counts) Initial: 240	デッドタイムカウント値 [Raw Counts]
Current range (A) Initial: 27.5F	電流検出レンジ [A]
Voltage range (V) Initial: 111.0F	電圧検出レンジ [V]
Counts for current offset measurement Initial: 500	オフセット取得時積算回数
A/D conversion channel for U phase current Initial: 0	U 相電流検出 A/D チャンネル番号
A/D conversion channel for W phase current Initial: 2	W 相電流検出 A/D チャンネル番号
A/D conversion channel for main line voltage Initial: 5	母線電圧検出 A/D チャンネル番号
A/D conversion channel for V phase current Initial: 1	V 相電流検出 A/D チャンネル番号 2シャントを選択した場合は無効となります。
Input voltage Initial: 24.0F	母線電圧入力値
Resolution of A/D conversion Initial: 0xFFF	A/D コンバータ分解能
Offset of A/D conversion for current Initial: 0x745	A/D コンバータ入力オフセット
Conversion level of A/D conversion for voltage Initial: 0.66F	電圧変換レベル
GTIOCA stop level Initial: Pin level High	上アーム停止時レベル
GTIOCB stop level Initial: Pin level High	下アーム停止時レベル
Maximum duty Initial: 0.9375F	PWM 最大デューティ デッドタイムを除いた最大デューティ

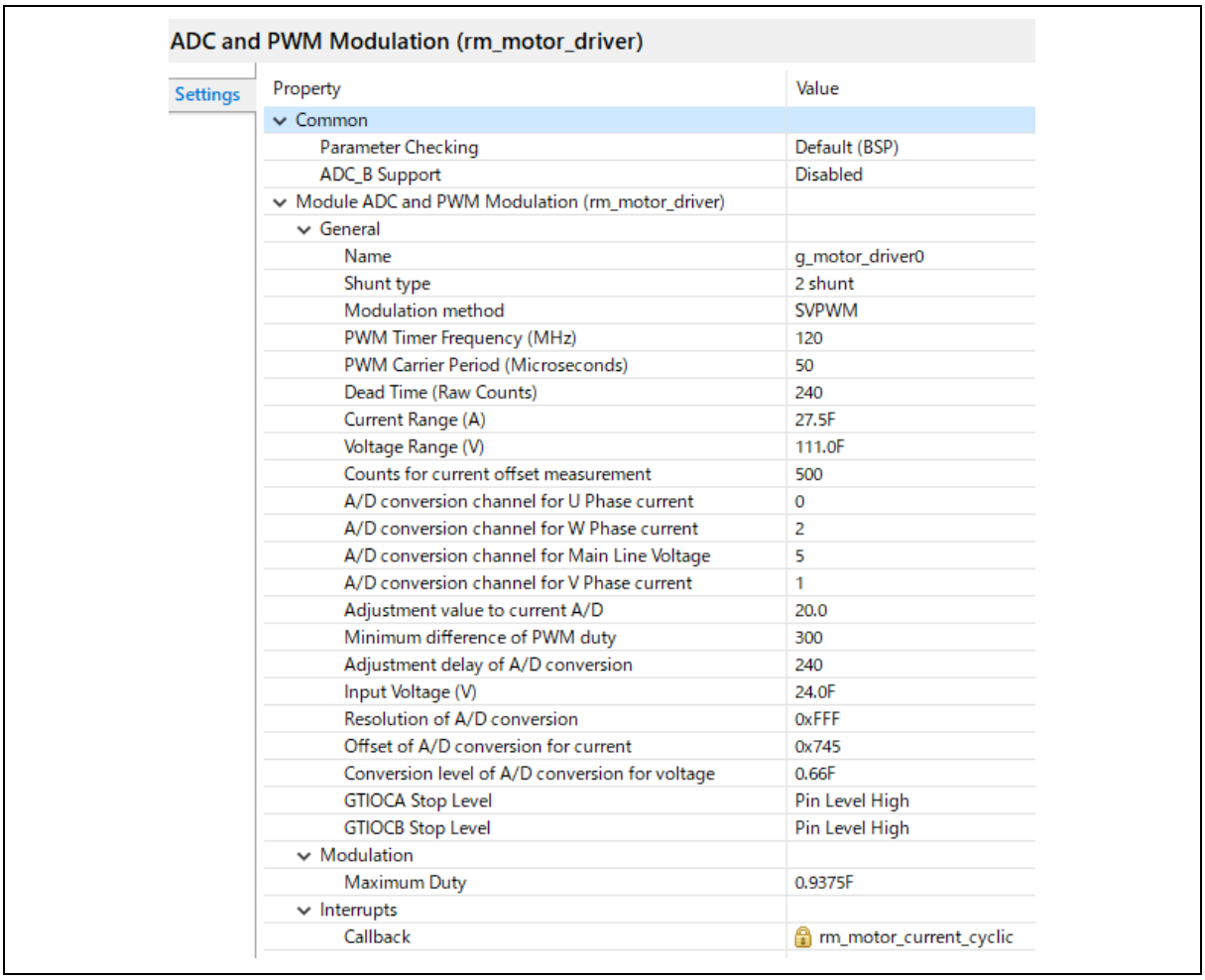


図 3-12 ADC と PWM モジュレーションドライバの FSP コンフィグレーション(FSP 3.5.0)

3.4 制御フロー（フローチャート）

3.4.1 メイン処理

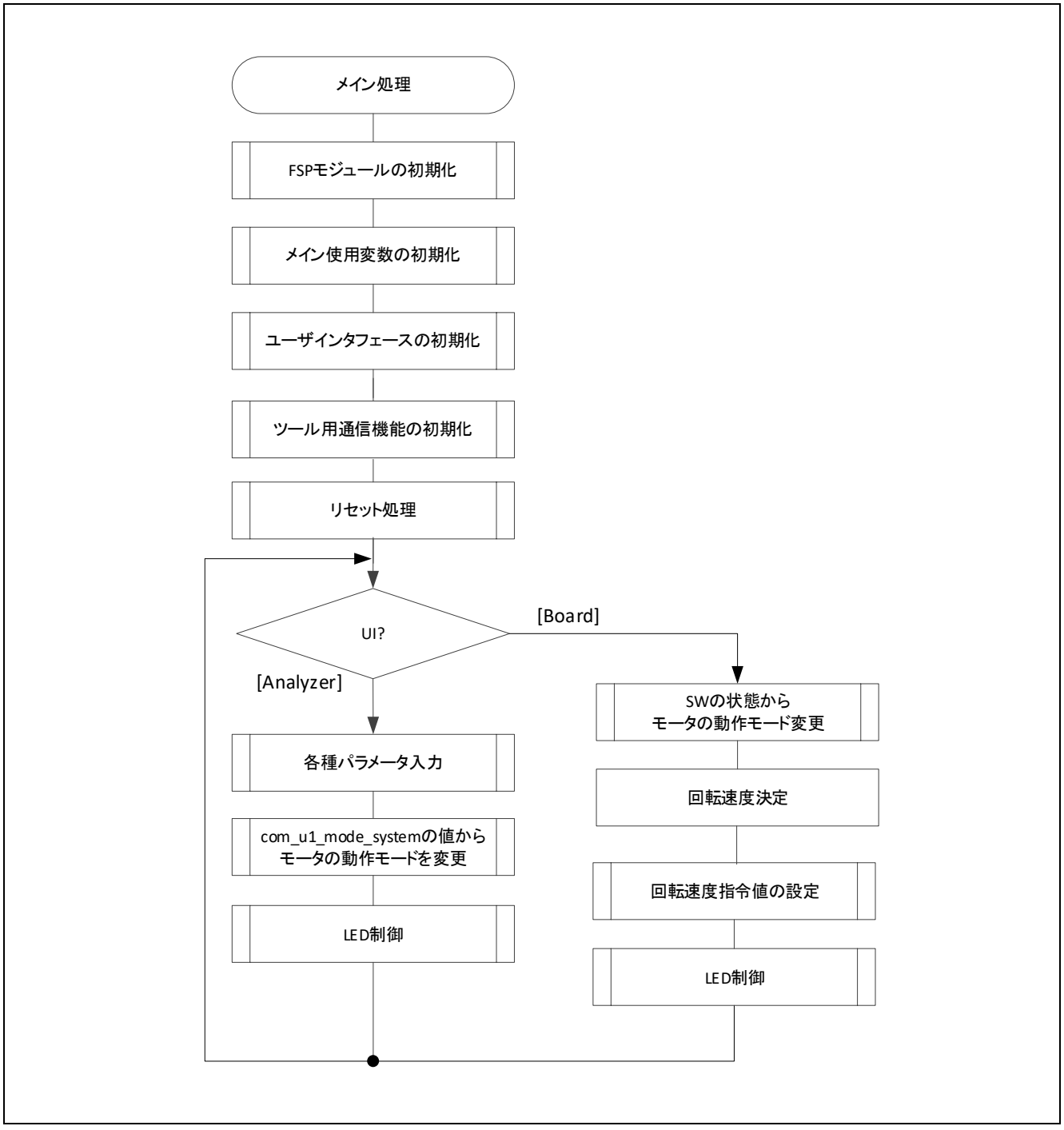


図 3-13 メイン処理フローチャート

3.4.2 50 [µs]周期割り込み（キャリア同期割り込み）処理

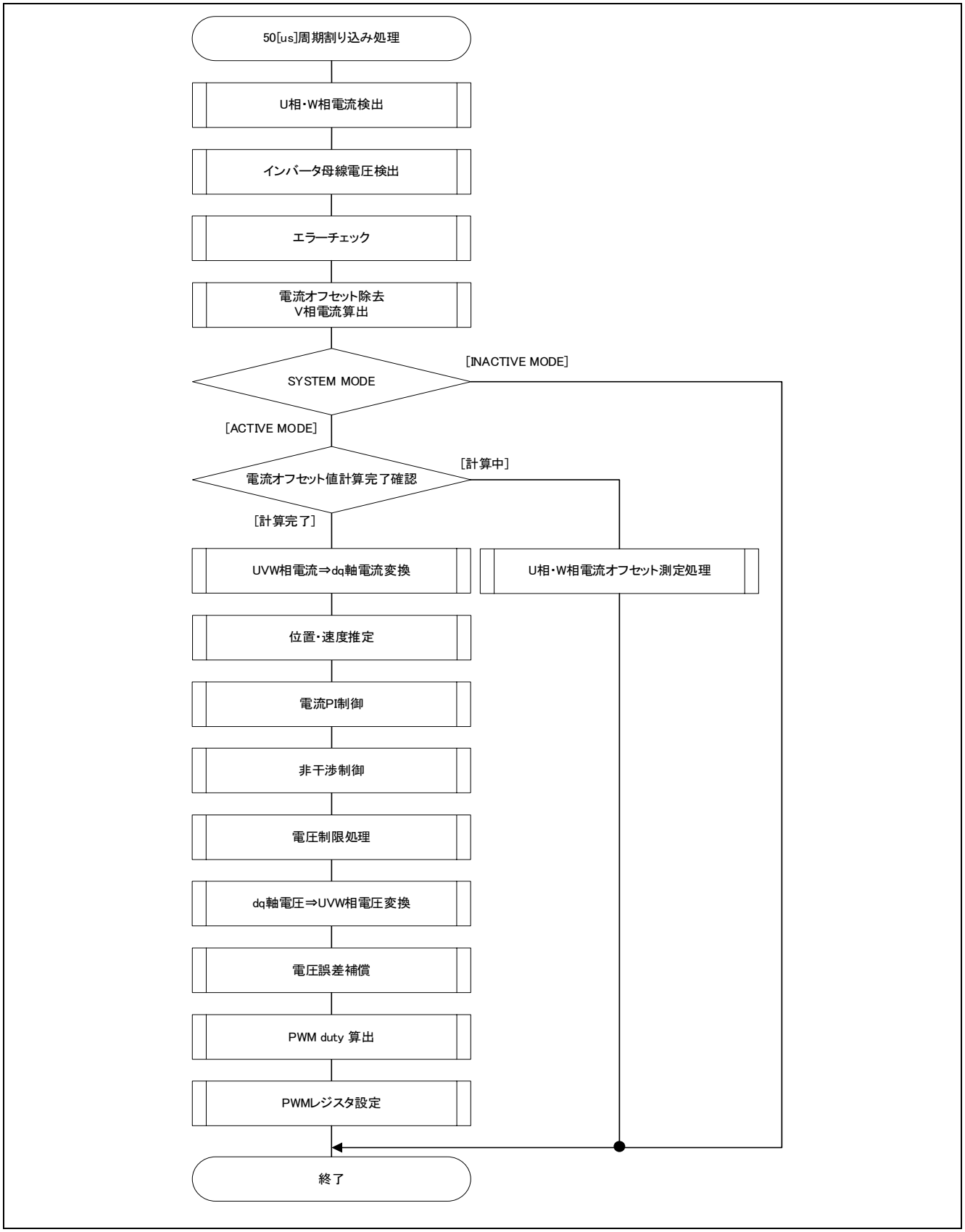


図 3-14 50[µs]周期割り込み処理フローチャート

3.4.3 500 [µs]周期割り込み処理

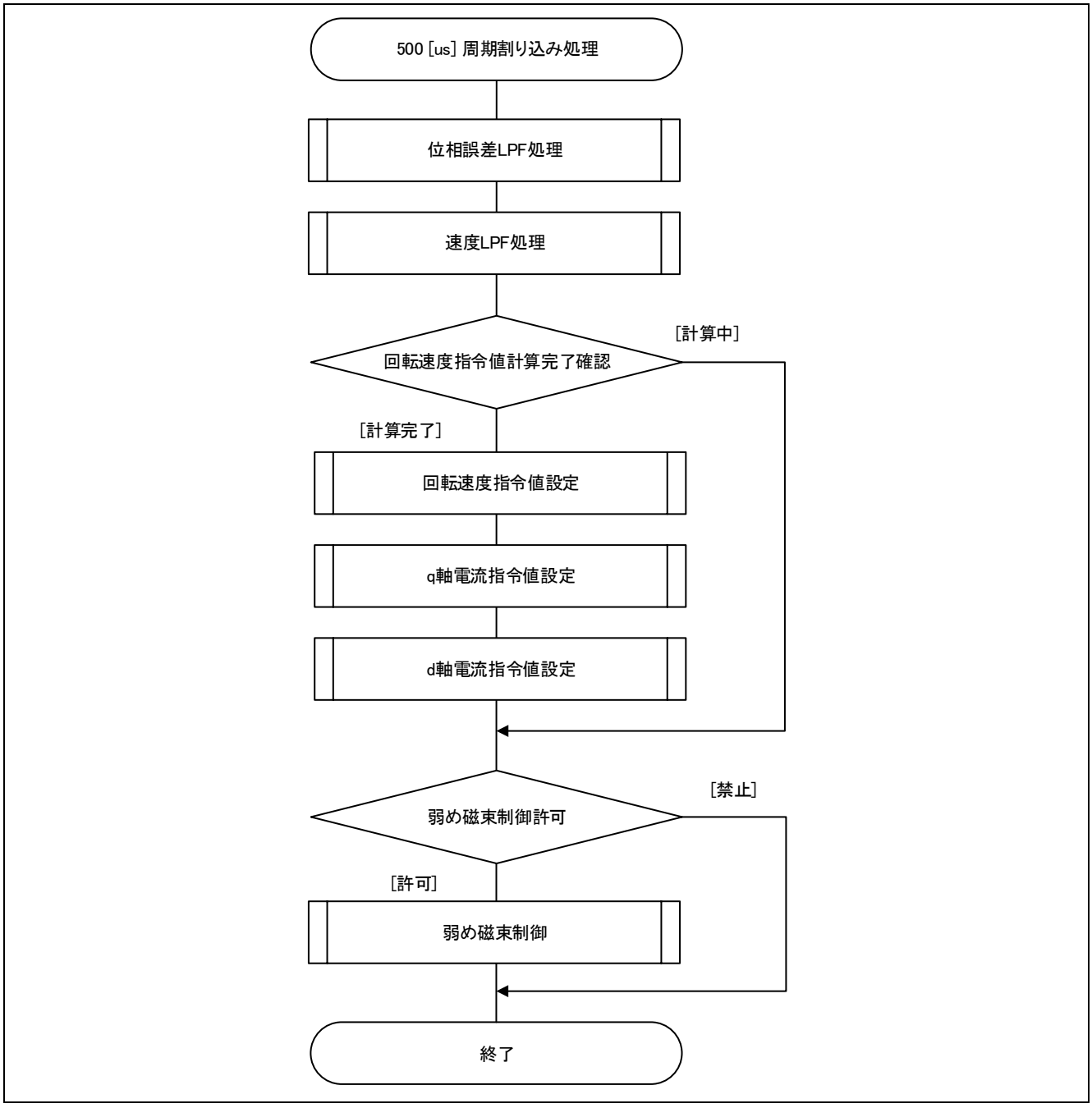


図 3-15 500 [µs]周期割り込み処理フローチャート

3.4.4 過電流検出割り込み処理

過電流検出割り込みは、サンプルソフトウェアにおける PWM 出力端子のハイインピーダンス制御条件である GTETRGC 出力レベル比較動作による出力短絡検出時に発生する割り込みです。そのため、本割り込み処理の実行開始時点では既に PWM 出力端子はハイインピーダンス状態になっており、モータへの出力は停止しています。

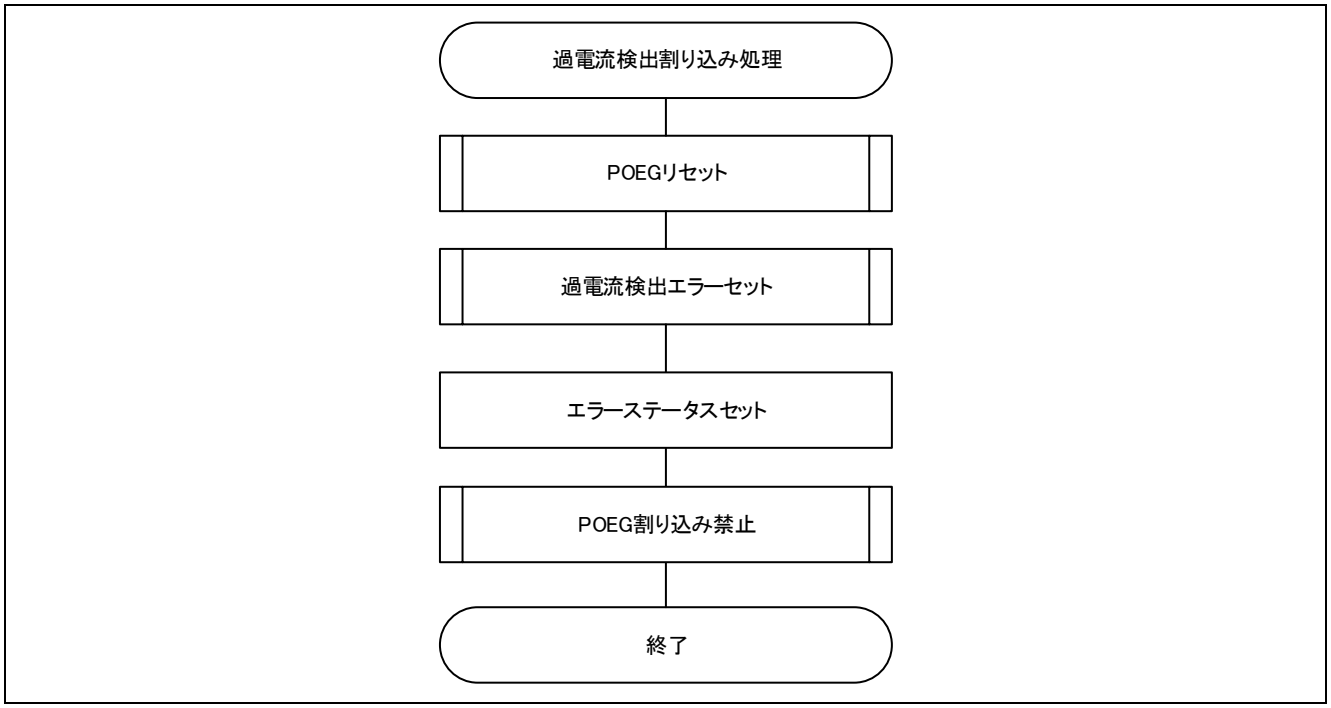


図 3-16 過電流検出割り込み処理フローチャート

4. 評価環境説明

サンプルソフトウェアの評価環境について説明します。

4.1 動作環境

表 4-1 に、サンプルソフトウェアをビルドおよびデバッグするためのハードウェア要件を示します。

表 4-1 センサレスベクトル制御ハードウェア基本仕様

項目	内容
インバータボード	RA6T1-RSSK [RTK0EM0000B10020BJ]
CPU ボード	RA6T1 CPU Card [RTK0EM0013C01201BJ]
モータ	ブラシレス DC モータ (TG-55L-KA 24V)
オンチップデバッグエミュレータ	RA6T1 CPU カードには、オンボードデバッガ（J-Link OB）が搭載されているため、エミュレータを用意する必要はありません。

表 4-2 に、サンプルソフトウェアをビルドおよびデバッグするためのソフトウェア要件を示します。

表 4-2 センサレスベクトル制御ソフトウェア基本仕様

項目		バージョン	内容
GCC	e2 studio	2021-10	Integrated development environment (IDE) for Renesas devices.
	GCC ARM Embedded	V10.3.1.20210824	C/C++ Compiler. (Download from e2 studio installer)
	Renesas Flexible Software Package (FSP)	V3.5.0 以降	Software package for writing applications for the RA microcontroller series.

4.2 プロジェクトのインポート

サンプルソフトウェアは、以下の手順で e2 studio にインポートできます。

1. File → Import

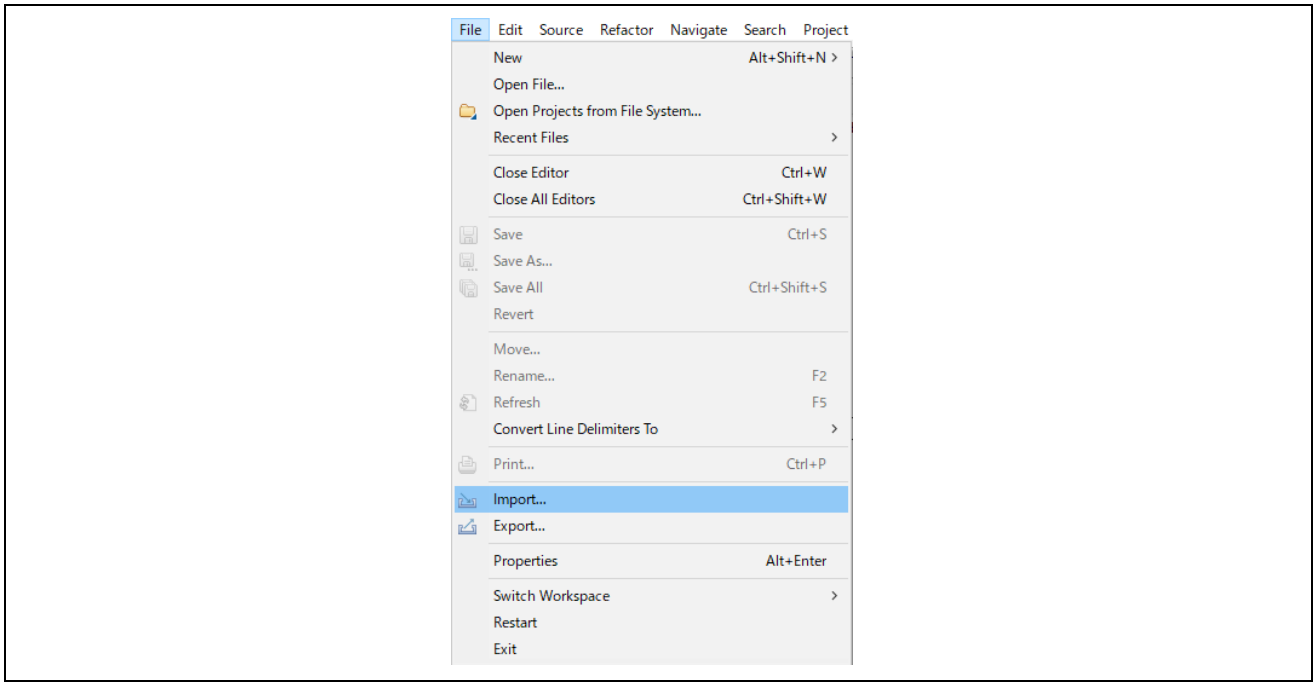


図 4-1 ファイルメニュー

2. 「Existing Projects into Workspace」を選択し、[次へ]ボタンをクリックします。

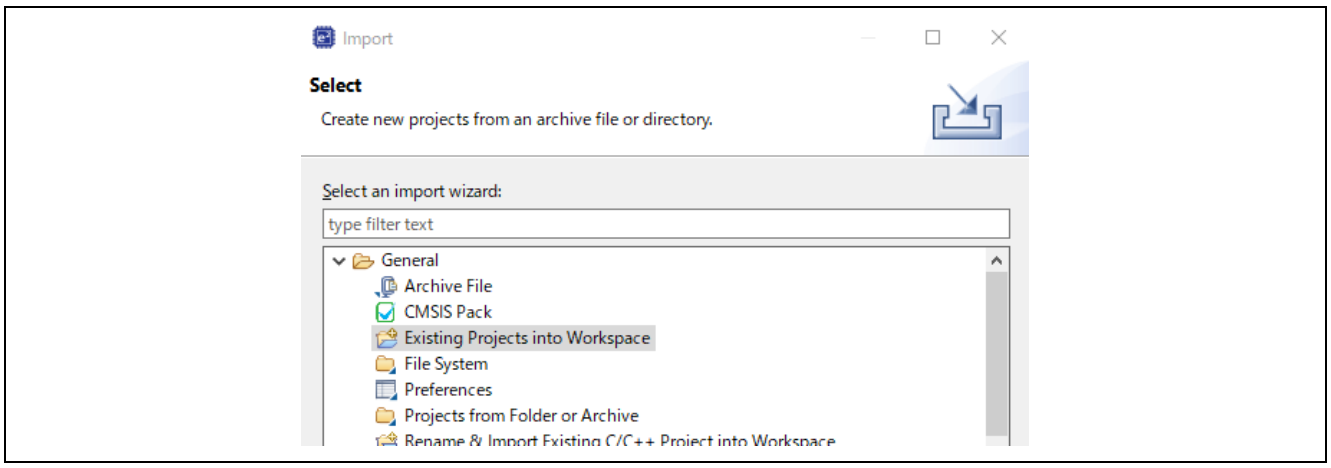


図 4-2　インポートメニュー

3. プロジェクトファイルを選択します。 Finish ボタンをクリックすると、プロジェクトがインポートされます。

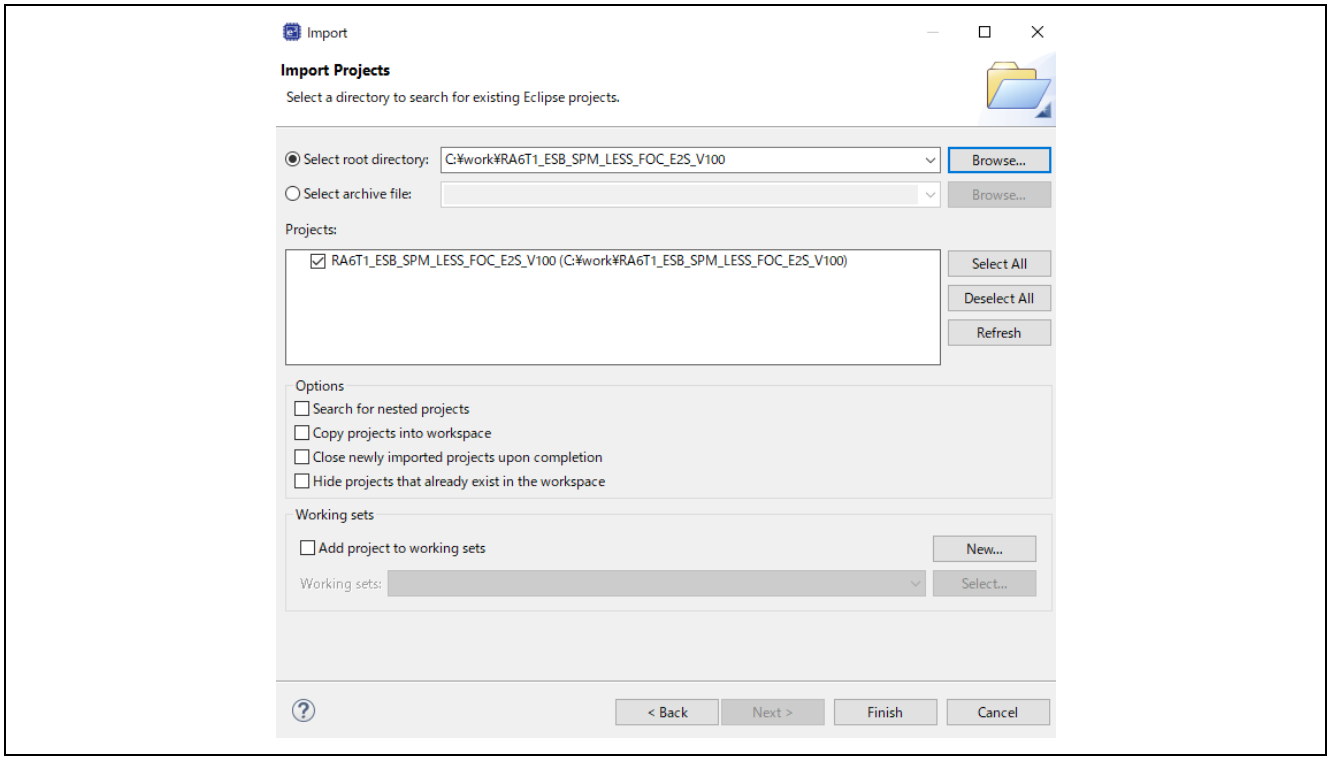


図 4-3　プロジェクトのインポート

4.3　ビルドとデバッグ

「e2 studio ユーザーズマニュアル入門ガイド（R20UT4204）」を参照してください。

4.4 サンプルソフトウェア操作概要

4.4.1 クイックスタート

モータ制御開発支援ツール「Renesas Motor Workbench」を使用せずにサンプルソフトウェアを動作させる場合、下記の手順でクイックスタートサンプルプロジェクトを実行します。

(1) 安定化電源投入後、またはリセット後はインバータボード上の LED1、LED2 は消灯状態で、モータは停止しています。

(2) インバータボード上のトグルスイッチ(SW1)を ON にするとモータが回転します。トグルスイッチ(SW1)を切り替えるごとにモータの回転開始/停止を繰り返します。モータが正常に回転している場合はインバータボード上の LED1 が点灯します。このとき、インバータボード上の LED2 が点灯している場合はエラーが発生しています。

(3) モータの回転方向を変更する場合は、インバータボード上のボリューム抵抗(VR)で調整します。

・ボリューム抵抗(VR)を右に回す：時計回りに回転

・ボリューム抵抗(VR)を左に回す：反時計回りに回転

(4) エラーが発生した場合、インバータボード上の LED2 が点灯し、回転が停止します。復帰するためにはインバータボードのトグルスイッチ(SW1)を OFF にした上でプッシュスイッチ(SW2)を押してください。

(5) 動作確認を終了する場合は、モータの回転が停止していることを確認し、安定化電源の出力を OFF にします。

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

4.5.1 概要

サンプルソフトウェアでは、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。使用方法などの詳細は「Renesas Motor Workbench ユーザーズマニュアル」を参照してください。

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。

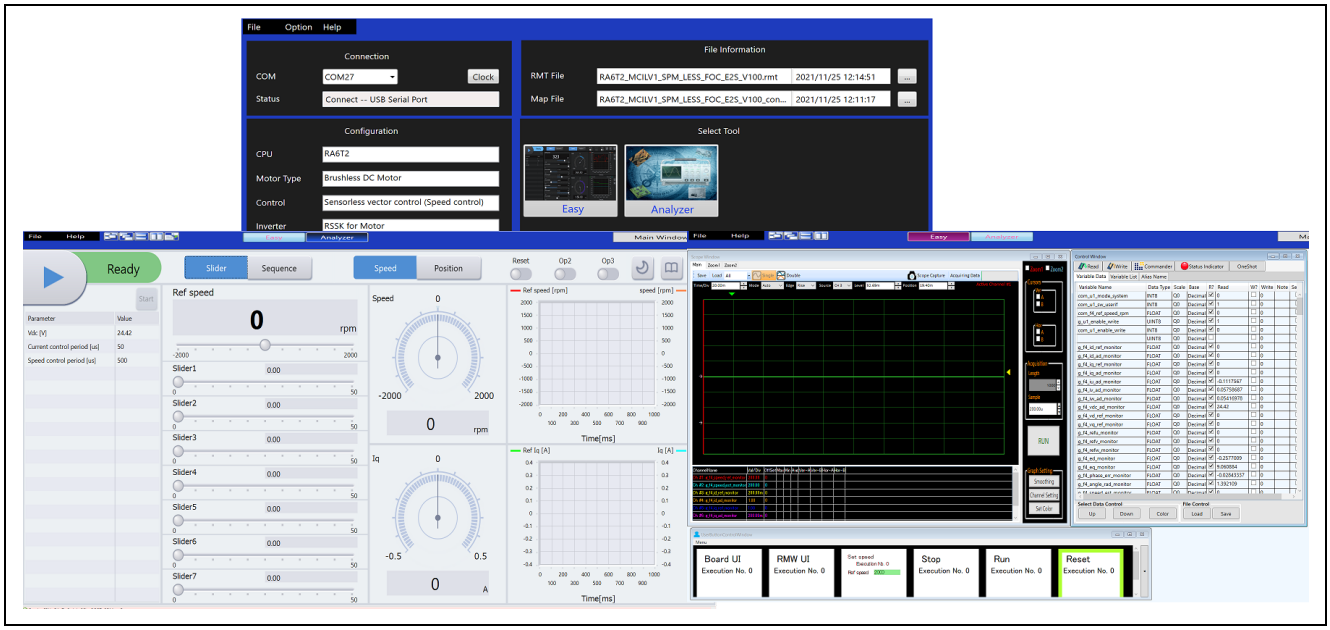
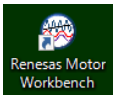


図 4-4 Renesas Motor Workbench 外観

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

① ツールアイコン 

をクリックしツールを起動する。

② Main Window の MENU バーから、[File] → [Open RMT File(O)]を選択。

プロジェクトフォルダの“src/application/user_interface/ics”フォルダ内にある RMT ファイルを読み込む。

③ ”Connection”の COM で接続されたキットの COM を選択する。

④ 右側の Select Tool の Easy または Analyzer ボタンをクリックする。

⑤ ”4.5.2 Easy 機能操作例”または”4.5.4 Analyzer 機能操作例”を元にモータを駆動させる。

4.5.2 Easy 機能操作例

Easy 機能を使用し、モータを操作する例を以下に示します。

- ユーザインタフェースを Renesas Motor Workbench 使用に変更する
  ① “RMW UI”を ON にする。

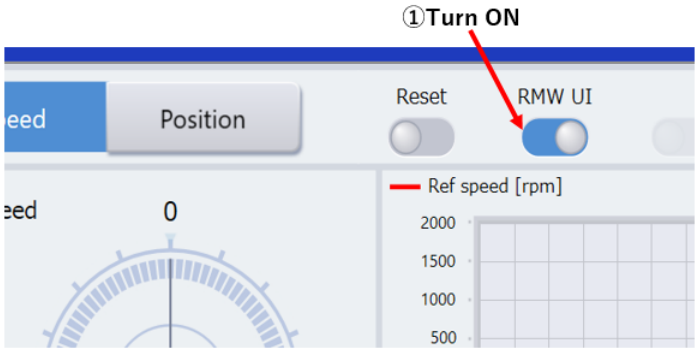


図 4-5 Renesas Motor Workbench 使用に変更する手順

- モータを回転させる
  ① “Run”ボタンを押す。
  ② 指令回転速度を“Ref speed”スライダで入力する。

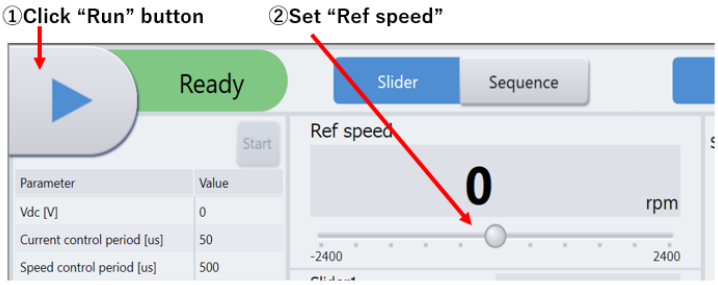


図 4-6 モータ回転の手順

- モータを停止させる
  ① “Stop”ボタンを押す。

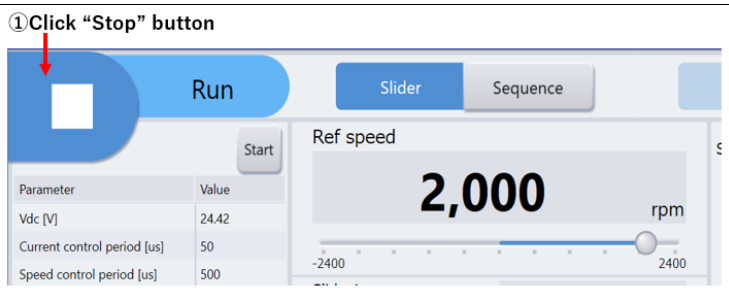


図 4-7 モータ停止の手順

- 止まってしまった（エラー）場合の処理
  ① “Reset”スイッチを ON にする。
  ② “Reset”スイッチを OFF にする。

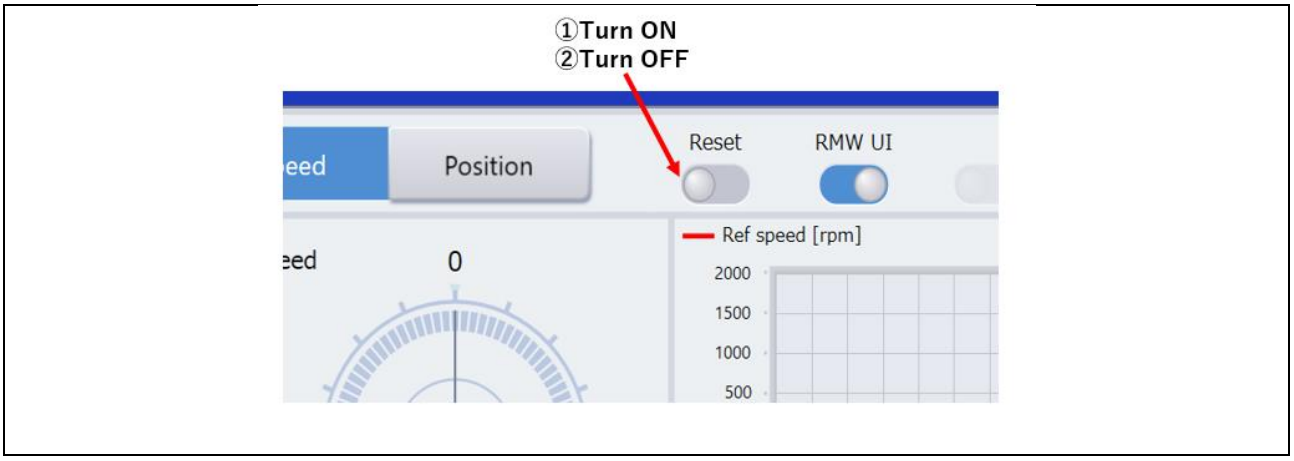


図 4-8 エラー解除の手順

4.5.3 Analyzer 機能用変数一覧

Analyzer ユーザインタフェース使用時の入力用変数一覧を表 4-3 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 と同じ値を書込んだ場合に対応する変数へ反映されます。ただし、(*)が付けられた変数は com_u1_enable_write に依存しません。

表 4-3 Analyzer 機能入力用変数一覧

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：Analyzer 使用　1：ボード使用（デフォルト）
com_u1_mode_system1(*)	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_f4_ref_speed_rpm	float	速度指令値（機械角）[rpm]
com_u2_mtr_pp	uint16_t	極対数
com_f4_mtr_r	float	抵抗 [Ω]
com_f4_mtr_ld	float	d 軸インダクタンス [H]
com_f4_mtr_lq	float	q 軸インダクタンス [H]
com_f4_mtr_m	float	磁束 [Wb]
com_f4_mtr_j	float	イナーシャ [kgm^2]
com_f4_current_omega	float	電流制御系固有周波数 [Hz]
com_f4_current_zeta	float	電流制御系減衰係数
com_f4_speed_omega	float	速度制御系固有周波数 [Hz]
com_f4_speed_zeta	float	速度制御系減衰係数
com_f4_e_obs_omega	float	誘起電圧推定系固有周波数 [Hz]
com_f4_e_obs_zeta	float	誘起電圧推定系減衰係数
com_f4_pll_est_omega	float	位置推定系固有周波数 [Hz]
com_f4_pll_est_zeta	float	位置推定系減衰係数
com_f4_ref_id	float	オープンループ制御時 d 軸電流指令値 [A]
com_f4_ol_id_up_step	float	d 軸電流指令値加算ステップ
com_f4_ol_id_down_step	float	d 軸電流指令値減算ステップ
com_f4_id_down_speed_rpm	float	d 軸電流指令値減算開始速度(機械角) [rpm]
com_f4_id_up_speed_rpm	float	d 軸電流指令値加算開始速度(機械角) [rpm]
com_f4_max_speed_rpm	float	速度最大値(機械角) [rpm]
com_f4_overspeed_limit_rpm	float	速度超過エラー閾値(機械角) [rpm]
com_f4_overcurrent_limit	float	電流超過エラー閾値[A]
com_f4_iq_limit	float	q 軸電流最大値[A]
com_f4_limit_speed_change	float	速度指令最大増減幅(電気角) [rad/s]
com_u1_enable_write	uint8_t	変数書き換え許可 (g_u1_enable_write と同じ値を書き込んだ場合に書き込み許可)

4.5.4 Analyzer 機能操作例

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、図 4-4 で示す”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。

- ユーザインタフェースを Renesas Motor Workbench 使用に変更する
  ① “com_u1_sw_userif”の[W?]欄に“チェック”が入っていることを確認する。
  ② [Write]欄に 0 を入力する。
  ③ “Write”ボタンを押す。

- モータを回転させる
  ① “com_u1_mode_system”,“com_f4_ref_speed_rpm”, “com_u1_enable_write”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_f4_ref_speed_rpm”の [Write] 欄に入力する。
  ③ “Write”ボタンを押す。
  ④ “Read”ボタンを押して現在の“com_f4_ref_speed_rpm”,”g_u1_enable_write”の [Read] 欄を確認する。
  ⑤ MCU 内の変数値へ反映させるため、“com_u1_enable_write” に④で確認した、”g_u1_enable_write”と同じ値を入力する。
  ⑥ “com_u1_mode_system”の [Write]欄に”1”を入力する。
  ⑦ “Write”ボタンを押す。

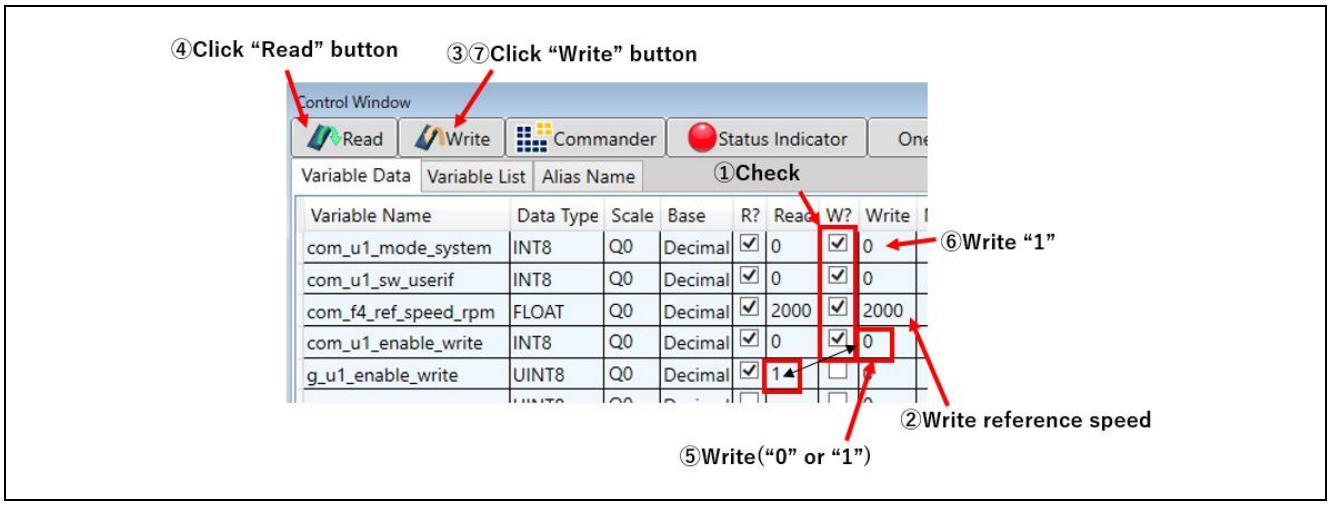


図 4-9 モータ回転の手順

- 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。

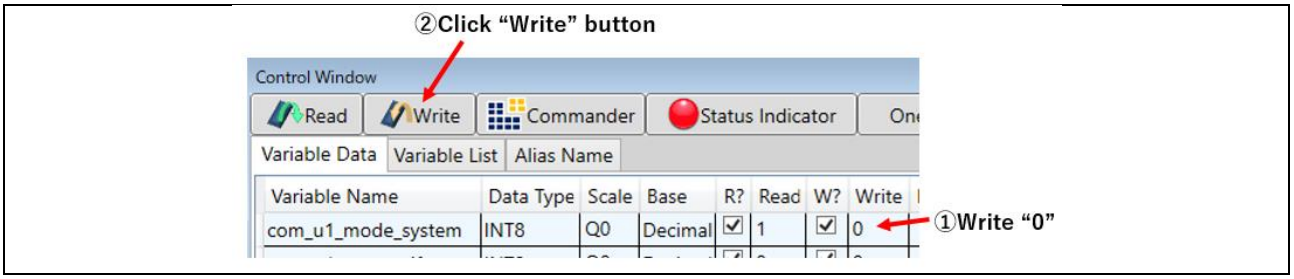


図 4-10 モータ停止の手順

- 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。

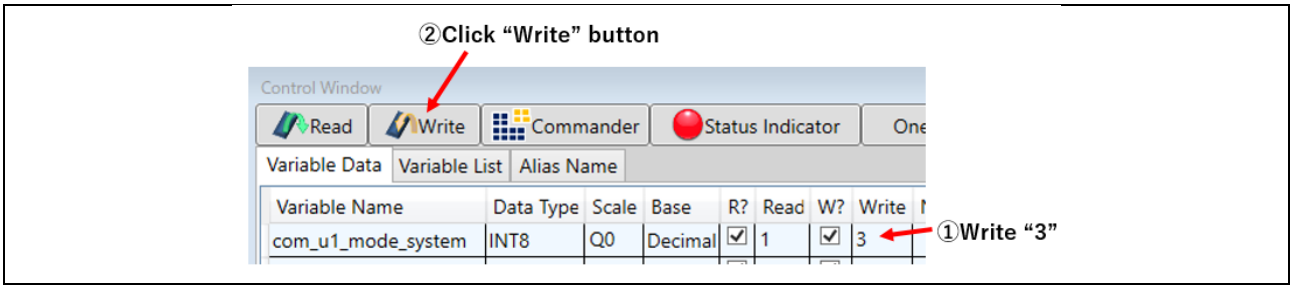


図 4-11 エラー解除の手順

4.5.5 Tuner 機能

Tuner 機能を使用するには、Renesas Motor Workbench が提供する実行ファイルを用います。

Tuner 機能の使用法については、Renesas Motor Workbench のダウンロードファイルに同梱されている Tuner 機能説明書(R20AN0528)を参照ください。

5. 参考ドキュメント

RA6T1 グループ　ユーザーズマニュアル　ハードウェア編（R01UH0897）

RA Flexible Software Package Documentation

永久磁石同期モータのセンサレスベクトル制御（アルゴリズム編）（R01AN3786）

Renesas Motor Workbench 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）

Motor Control Evaluation System for RA Family (R12UZ0078)

RA6T1 CPU カード ユーザーズマニュアル（R12UZ0077）

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2020.10.28	—	初版発行
1.10	2021.12.17	—	FSP V3.5.0 をサポート
1.11	2022.10.18	—	FSP V4.1.0 をサポート

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うと、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、ノイズ幅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。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（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。
2. 当社製品、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、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。
5. 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。
6. 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。
7. 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。
8. 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。
9. 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。
10. 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いたします。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
12. 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.4.0-1　2017.11)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

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。

© 2021 Renesas Electronics Corporation. All rights reserved.