

RA2T1 永久磁石同期モータのセンサレスベクトル制御

Renesas Flexible Motor Control シリーズ用

要旨

本アプリケーションノートは RA2T1 マイクロコントローラを使用し、永久磁石同期モータをセンサレスベクトル方式で駆動するサンプルプログラム及びモータ制御開発支援ツール「Renesas Motor Workbench (RMW)」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。サンプルプログラムを使用する場合、適切な環境で十分な評価をした上でご使用ください。本アプリケーションノート掲載開発環境での事故、損害等が発生した場合、一切の責任を負いません。

動作確認デバイス

サンプルプログラムの動作確認は以下のデバイスで行っております。

- RA2T1(R7FA2T1074CFL)

対象サンプルプログラム

本アプリケーションノートの対象サンプルプログラムを下記に示します。

<1 シャント電流検出方式>

- RA2T1_MCILV1_PM_LESS_FOC_1S_E2S_V101 (IDE : e²studio)

<3 シャント電流検出方式>

- RA2T1_MCILV1_PM_LESS_FOC_3S_E2S_V101 (IDE : e²studio)

参考資料

- RA2T1 グループ ユーザーズマニュアル ハードウェア編(R01UH1089)
- Renesas Motor Workbench ユーザーズマニュアル (R21UZ0004)
- RA2T1 Group MCK-RA2T1 ユーザーズマニュアル (R12UZ0152)
- RA2T1 Group MCB-RA2T1 ユーザーズマニュアル (R12UZ0145)
- MCI-LV-1 ユーザーズマニュアル (R12UZ0092)
- MC-COM ユーザーズマニュアル (R12UZ0093)

目次

1. 概説 3

2. システム概要 4

3. 制御プログラム 14

4. モータ制御開発支援ツール「Renesas Motor Workbench」の利用方法 78

1. 概説

本アプリケーションノートでは、RA2T1 マイクロコントローラを使用した永久磁石同期モータ(PMSM)のセンサレスベクトル制御サンプルプログラムの実装方法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します。

1.1 開発環境

本アプリケーションノート対象サンプルプログラムの開発環境を表 1-1、表 1-2 に示します。

表 1-1 サンプルプログラムの開発環境(H/W)

マイコン	評価ボード	モータ
RA2T1 (R7FA2T1074CFL)	RA2T1 CPU ボード(RTK0EMA810C00000BJ) ^{注1} インバータボード(RTK0EM0000B12020BJ) ^{注1} 通信ボード(RTK0EMXC90Z00000BJ) ^{注1}	MOONS'製 R42BLD30L3

表 1-2 サンプルプログラムの開発環境(S/W)

e²studio バージョン	ツールチェーン	FSP	デバッグ・ツール
2026-04.2	gcc 13.2.1.arm-13-7	v6.5.0	Renesas Motor Workbench 3.3.1

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下さい。

【注】

1. RA2T1 CPU ボード(RTK0EMA810C00000BJ)、並びにインバータボード(RTK0EM0000B12020BJ)、通信ボード(RTK0EMXC90Z00000BJ)は、ルネサスエレクトロニクス株式会社の製品です。

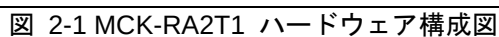
2. システム概要

本システムの概要を以下に説明します。

2.1 ハードウェア仕様

MCK-RA2T1 は CPU ボード、インバータボード、通信ボードで構成されています。キットとしてのスペックおよびそれぞれのボードのスペックに関しては、以下のドキュメントを参照してください。

- MCK-RA2T1
RA2T1 Group MCK-RA2T1 ユーザーズマニュアル (R12UZ0152)
- CPU ボード
RA2T1 Group MCB-RA2T1 ユーザーズマニュアル (R12UZ0145)
- インバータボード
MCI-LV-1 ユーザーズマニュアル (R12UZ0092)
- 通信ボード
MC-COM ユーザーズマニュアル (R12UZ0093)



2.1.2 ハードウェア設定

サンプルソフトウェアを使用するためには、ハードウェアの設定が必要になります。以下にその設定内容を示します。設定内容は MCK-RA2T1 の初期設定と同じです。

(1) インバータボード

ジャンパーピン(JP1~JP15)の設定と機能について以下に示します。

表 2-1 インバータボードのジャンパーピン設定

ジャンパーピン	設定	機能
JP1	2-3pin ショート	1-2pin ショート : 5V レギュレータ無効 2-3pin ショート : 5V レギュレータ有効
JP2, JP3, JP4, JP6, JP12, JP13	2-3pin ショート	1-2pin ショート : 電流検出アンプ無効 2-3pin ショート : 電流検出アンプ有効
JP5, JP7, JP9, JP10, JP14, JP15	1-2pin ショート	1-2pin ショート : 電流検出アンプゲイン 20 倍 1-2pin オープン : 電流検出アンプゲイン 10 倍
JP8, JP11	2-3pin ショート	1-2pin ショート : 2-/3-シャント電流検出 2-3pin ショート : 1-シャント電流検出

(2) CPU ボード

ジャンパーピン(JP3)の設定と機能について以下に示します。

表 2-2 CPU ボードのジャンパーピン設定

ジャンパーピン	設定	機能
JP3	1-2pin オープン	1-2pin ショート : オンボードデバッグ無効 1-2pin オープン : オンボードデバッグ有効

(3) 通信ボード

ジャンパーピン(JP1~JP3)の設定と機能について以下に示します。

表 2-3 通信ボードのジャンパーピン設定

ジャンパーピン	設定	機能
JP1	1-2pin オープン	1-2pin ショート : MD 端子プルアップ無効(設定禁止) 1-2pin オープン : MD 端子プルアップ有効
JP2	1-2pin ショート	1-2pin ショート : 汎用ポート(PC6)プルアップ無効 1-2pin オープン : 汎用ポート(PC6)プルアップ有効
JP3	1-2pin ショート	1-2pin ショート : 汎用ポート(PC5)プルアップ無効 1-2pin オープン : 汎用ポート(PC5)プルアップ有効

2.1.3 ユーザインタフェース

本システムのユーザインタフェース一覧を以下に示します。

(1) インバータボード

表 2-4 インバータボードのユーザインタフェース

項目	インタフェース部品	機能
VR1	可変抵抗器	未使用
SW1	トグルスイッチ (SW1)	未使用
SW2	プッシュスイッチ (SW2)	未使用
LED1	橙色 LED	・ 駆動時 : 点灯 ・ 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 2-5 インバータボードのコネクタインタフェース

項目	端子数	機能
CN1	2	電源入力コネクタ
CN2	3	モータコネクタ
CN3	34	CPU ボードコネクタ
CN4	34	CPU ボードコネクタ
CN5	10	ENC/IPS 信号入力用コネクタ
CN6	5	ホールセンサ信号入力用コネクタ

(2) CPU ボード

表 2-6 CPU ボードのユーザインタフェース

項目	インタフェース部品	機能
RESET	タクトスイッチ (SW1)	MCU の外部リセット
LED1	橙色 LED	・ 駆動時 : 点灯 ・ 停止時 : 消灯
LED2	橙色 LED	・ エラー検出時 : 点灯 ・ 通常動作時 : 消灯

表 2-7 CPU ボードのコネクタインタフェース

項目	端子数	機能
CN1	6	非搭載
CN2	34	インバータボードコネクタ
CN3	34	インバータボードコネクタ
CN4	4	通信ボード接続用 SCI コネクタ
CN5	(USB Type-C)	J-Link OB 用 USB コネクタ

(3) 通信ボード

表 2-8 通信ボードのユーザインタフェース

項目	インタフェース部品	機能
RESET	タクトスイッチ (SW1)	MCU の外部リセット

表 2-9 通信ボードのコネクタインタフェース

項目	端子数	機能
CN1	4	非搭載
CN2	(USB miniB)	USB コネクタ (ユーザは使用不可)
CN3	(USB Type-C)	PC 接続用 USB コネクタ
CN4	4	非搭載
CN5	4	CPU ボード接続用 SCI コネクタ

本システムの RA2T1 マイクロコントローラの端子機能一覧を表 2-10 に示します。

表 2-10 RA2T1 端子機能一覧

ピン番号	RA2T1 端子機能	信号機能
1	P400/GTIOC3A	PFCPWM1/ENCA
2	P401/GTIOC3B	PFCPWM2/ENCB
3	VCL1	System
4	P402	SW1
5	P403	INV CONNECTED
6	VSS1	GND
7	P213/XTAL	External clock
8	P212/EXTAL	External clock
9	VCC1	VCC
10	P409/IRQ6	HALL_V
11	P408/IRQ7	HALL_W
12	P407/GTIOC0B	UN
13	P915	LED3
14	P914	LED2
15	P913	LED1
16	P208/GTIOC0A	UP
17	P207	Relay
18	P206/IRQ0	HALL_U
19	RES#	Reset
20	P201/MD	ARM debugger
21	P200	SAFE LOCK
22	P302/GTETRGA	OC INV
23	P301/GTETRGB	OCPFC/ENCZ
24	P300/SWCLK	ARM debugger
25	P108/SWDIO	ARM debugger
26	P109/GTIOC2A	VP
27	P110/GTIOC2B	VN
28	P111/GTIOC1A	WP
29	P112/GTIOC1B	WN
30	VCC2	VCC
31	VSS2	GND
32	P104/IRQ1	SW2/Overtemp
33	P103/CTS0/AN019	VCOM/HVtemp
34	P102/AN020	VU
35	P101/TXD0	SCI
36	P100/RXD0	SCI
37	P500 VCOM	VCOM
38	P015/AN002	IW
39	P014/AN001	IV
40	P013/AN000	IU
41	P012/AN007	VV
42	AVCC0	AVCC
43	AVSS0	AGND
44	P011/AN006	VW
45	P010/AN005	IPFC2/VR
46	P002/AN010	IPFC1
47	P001/AN009	VPFC
48	P000/AN008	VPN

2.1.4 周辺機能

本システムに使用する周辺機能一覧を表 2-11 に示します。

表 2-11 周辺機能一覧

周辺機能	用途
12 ビット A/D コンバータ (ADC12)	インバータ母線電圧測定 3 相電流検出 [3 シャントモード時使用] DC リンク電流検出 [1 シャントモード時使用]
低消費電力非同期汎用タイマ (AGTW チャンネル 0)	1 [ms] インターバル・タイマ
汎用 PWM タイマ (GPT チャンネル 0, 1, 2)	相補 PWM モード使用の PWM 出力 50 [us] 制御周期タイマ
GPT 用のポートアウトプット イネーブル (POEG)	トリガ入力による PWM 出力強制遮断
ウォッチドッグタイマ(WDT)	プログラムの暴走検出

(1) 12bit A/D コンバータ(ADC12)

V 相電流(I_v)、W 相電流(I_w)、及びインバータ母線電圧(V_{dc})を測定します。VW 相電流については同時サンプリングにより測定します。[3 シャントモード時使用]

DC リンク電流及びインバータ母線電圧(V_{dc})を「12bit A/D コンバータ」を使用して測定します。[1 シャントモード時使用]

(2) 低消費電力非同期汎用タイマ(AGTW0)

1 [ms]インターバル・タイマとして使用します。

(3) 汎用 PWM タイマ(GPT チャンネル 0, 1, 2)

2 本の AD 変換トリガコンペアレジスタを使用して任意のタイミングで AD 変換を実行できます。相補 PWM モード 3 を使用して非対称波形を出力できます。

(4) GPT 用のポートアウトプットイネーブル (POEG)

インバータボードで検出した過電流信号(GTETRGA 入力)により PWM 出力を強制遮断します。遮断解除要因を検出後、ソフトウェアにより解除。

(5) ウォッチドッグタイマ

プログラムの暴走検出で使用します。

2.2 ソフトウェア仕様

2.2.1 基本仕様

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 2-12 ソフトウェア仕様

項目	内容	
制御方式	ベクトル制御	
電流検出方式	DC リンク電流を検出する方式(1 シャントモード)と 3 相電流を検出する方式(3 シャントモード)をコンパイルスイッチで切り替えて使用 1 シャントモード又は 3 シャントモードをコンパイルスイッチにて切替え(デフォルトは 1 シャントモード)	
モータ回転開始/停止	モータ制御開発支援ツールによる操作 ^注	
回転子磁極位置検出(センサレス)	誘起電圧を利用した角度推定 モータの突極性と磁気飽和特性を利用した初期位置検出	
入力電圧	DC24[V]	
メインクロック周波数	CPU クロック : ICLK 64[MHz] GPT クロック : PCLKB 32[MHz]	
キャリア周波数(PWM)	16, 20(デフォルト), 40 [kHz]	
デッドタイム	2 [μs]	
制御周期	電流制御/位置・速度推定(デフォルト) : 100 [μs] (1 シャントモード) 50 [μs] (3 シャントモード) 速度制御 : 1 [ms]	
回転速度範囲	CW : 0 [rpm] ~ 4000[rpm] CCW : 0 [rpm] ~ 4000 [rpm] ただし、800 [rpm]以下は速度オープンループで駆動	
最適化設定	Optimize more(-O2)	
ROM/RAM サイズ	ROM	1 シャントモード : 32476 B 3 シャントモード : 30476 B
	RAM	1 シャントモード : 3180 B 3 シャントモード : 3072 B
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. インバータ母線電圧が 48 [V]を超過 2. インバータ母線電圧が 8 [V]より下落 3. 回転速度が 8000 [rpm]を超過 4. 各相の電流が 2.26 [A]を超過 5. 各相の電流が 21.4 [A]を超える値を検出(POEG による遮断)	

【注】 詳細に関しては“モータ制御開発支援ツール「Renesas Motor Workbench」の利用方法”を参照してください。

2.2.2 ソフトウェア・ファイル構成

サンプルプログラムのフォルダとファイル構成を表 2-13 に記します。

表 2-13 サンプルプログラムのフォルダとファイル構成

フォルダ		ファイル	内容
src/config		r_mtr_config.h	コンフィギュレーション定義
		r_mtr_motor_parameter.h	モータパラメータ定義
		r_mtr_control_parameter.h	制御パラメータ定義
		r_mtr_inverter_parameter.h	インバータパラメータ定義
		r_mtr_scaling_parameter.h	スケーリングパラメータ定義
src/application	main	mtr_main.h	メイン関数
		mtr_main.c	
	board	r_mtr_board.h	ハードウェア UI 関数定義 ハードウェア UI 処理
		r_mtr_board.c	
	user_interface/ ics	r_mtr_ics.h	Analyzer ^{注1} UI 関連関数定義 Analyzer ^{注1} UI 処理
		r_mtr_ics.c	
		ICS2_RA2T1.h	RMW 通信用関数宣言 RMW 通信用ライブラリ
		ICS2_RA2T1.o	
		RA2T1_MCILV1_PM_LESS_FOC_1 S_E2S_conv.map	マップファイル (1 シャント時) RMW 初期設定ファイル (1 シャント時)
		RA2T1_MCILV1_PM_LESS_FOC_1 S_E2S.rmt	
		RA2T1_MCILV1_PM_LESS_FOC_3 S_E2S_conv.map	マップファイル (3 シャント時) RMW 初期設定ファイル (3 シャント時)
		RA2T1_MCILV1_PM_LESS_FOC_3 S_E2S.rmt	
src/driver		r_mtr_ra2t1.h	MCU 固有定義
		r_mtr_ra2t1.c	
		r_mtr_ADC.h	ADC ドライバ関数定義 ADC ドライバ処理
		r_mtr_ADC.c	
		r_mtr_GPT_THREE_PHASE.h	GPT ドライバ関数定義 GPT ドライバ処理
		r_mtr_GPT_THREE_PHASE.c	
		r_mtr_POEG.h	POEG ドライバ関数定義 POEG ドライバ処理
		r_mtr_POEG.c	
src/middle		r_mtr_pin_data.h	未使用端子の処理
		r_mtr_pin_data.c	
		r_mtr_common.h	ハードウェア共通定義
		R_DSP_RA2_GCC_32BIT.h	DSP 定義 モータ制御向け演算ライブラリ
		R_DSP_RA2_GCC_32BIT.o	
		rm_mtr_common.h	共通定義
		rm_mtr_parameter.h	モータ制御パラメータ定義
		rm_mtr_ctrl_gain.h	ゲイン設計関数定義 ゲイン設計
		rm_mtr_ctrl_gain.o	
		rm_mtr_driver.h	ドライバアクセス関数定義 ドライバアクセス処理
		rm_mtr_driver.c	
		rm_mtr_foc_less_speed.h	センサレスベクトル制御関連関数定義 センサレスベクトル制御関連処理
		rm_mtr_foc_less_speed.c	
		rm_mtr_speed.c	速度制御割り込み処理
		rm_mtr_est_phase_err.h	軸誤差の推定処理関数定義 軸誤差の推定処理
		rm_mtr_est_phase_err.o	
		rm_mtr_ol2cl_ctrl.h	センサレス切り替え制御関数定義 センサレス切り替え制御処理
		rm_mtr_ol2cl_ctrl.o	
		rm_mtr_carrier.c	キャリア周期割り込み処理
		rm_mtr_statemachine.h	ステートマシン関数定義

注 1: モータ制御開発支援ツール「Renesas Motor Workbench」の Analyzer 機能の詳細については、4 章を参照下さい。また、モータ制御開発支援ツール「Renesas Motor Workbench(RMW)」に関わるフォルダ、ファイル、関数、変数の名前には識別子「ics / ICS(旧ルネサス製モータ制御開発支援ツール「In Circuit Scope」の略)」が付加されている場合があります。

2.2.3 モジュール構成

サンプルプログラムのモジュール構成を図 2-2 に示します。

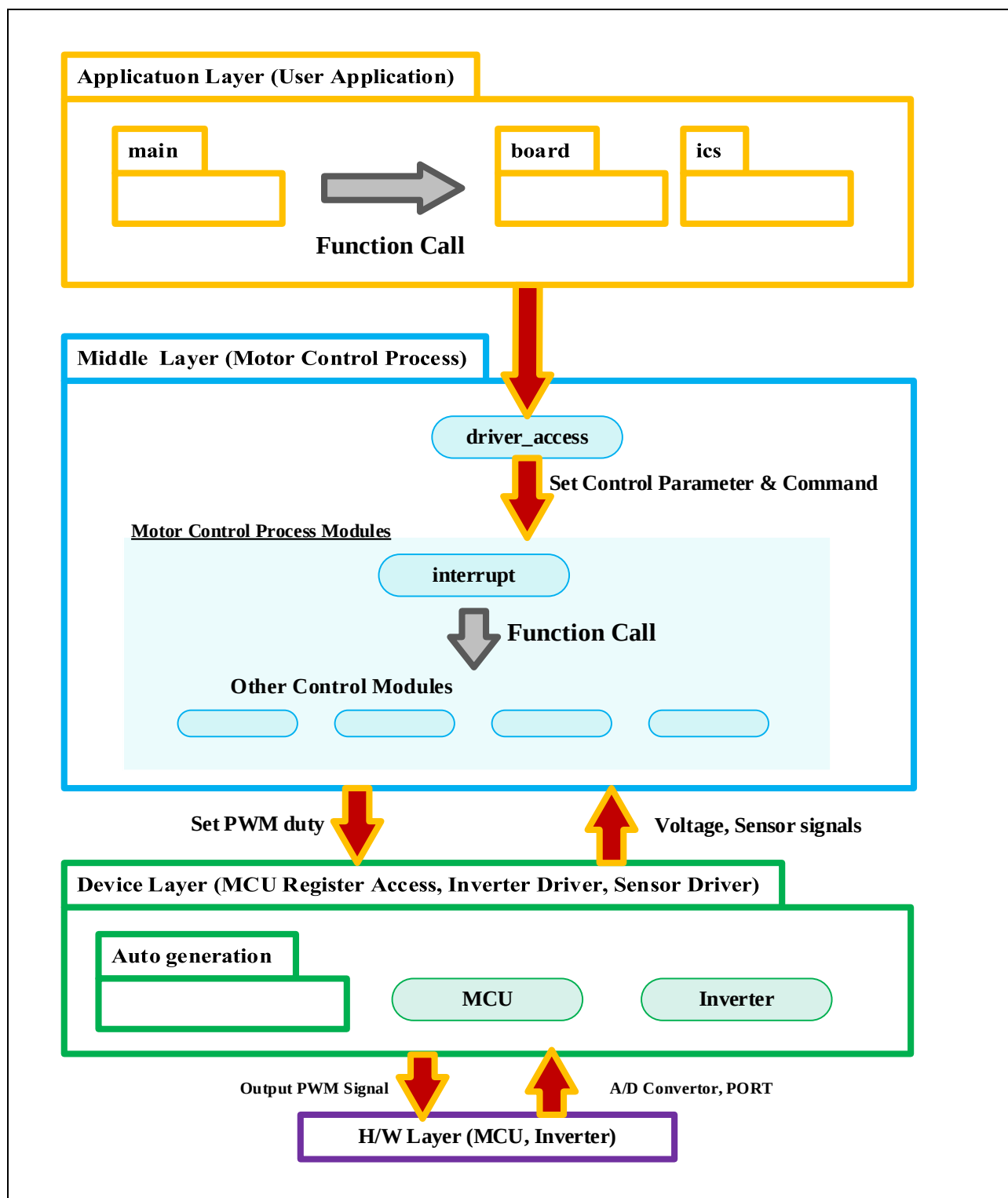


図 2-2 サンプルプログラムのモジュール構成

3. 制御プログラム

本アプリケーションノートの対象サンプルプログラム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、モータ制御開発支援ツールからの入力によって制御します。

3.1.2 A/D 変換

(1) インバータ母線電圧

下記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 3-1 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]~73.5 [V] : 0000H~0FFFH	AN008

(2) DC リンクシャント抵抗電流 [1 シャントモード時]

以下の表のように DC リンクシャント抵抗電流を測定し、ベクトル制御に使用します。

表 3-2 1 シャントモード時の電流の変換比

項 目	変換比(DC リンクシャント抵抗電流 : A/D 変換値)	チャネル
DC リンクシャント抵抗電流	-8.25 [A]~8.25 [A] : 0000H~0FFFH	AN000

(3) U 相、V 相、W 相電流 [3 シャントモード時]

以下の表のように U 相、V 相、W 相電流を測定し、ベクトル制御に使用します。

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	-8.25 [A]~8.25 [A] : 0000H~0FFFH	Iu : AN000 Iv : AN001 Iw : AN002

A/D 変換特性の詳細に関しては「RA2T1 ユーザーズマニュアル ハードウェア編」を参照して下さい。

3.1.3 POEG

(1). 過電流検出

インバータ基板の過電流検出コンパレータ出力の L レベルを検出し、PWM 出力を強制遮断します。

3.1.4 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 3-1 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。

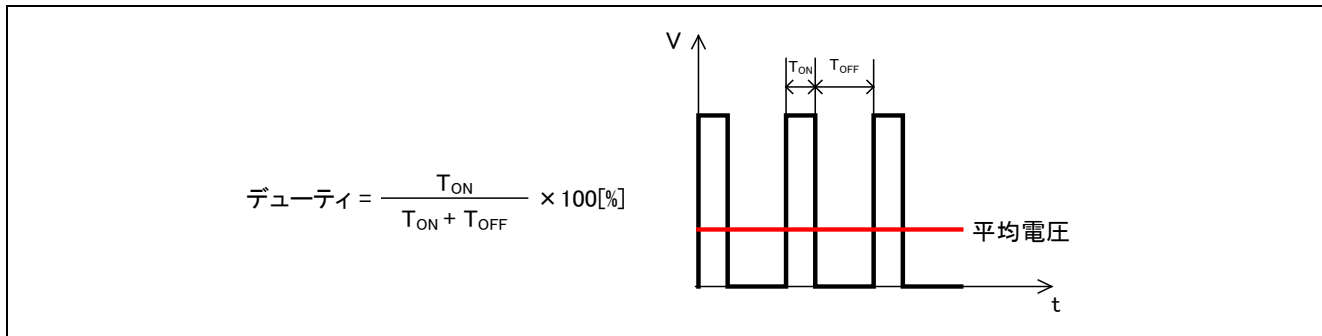


図 3-1 PWM 制御

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

3.1.5 変調

モータへの入力電圧はパルス幅変調（以降、PWM）によって信号生成し印加されます。PWM のパルス幅の生成方法を説明します。

(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。

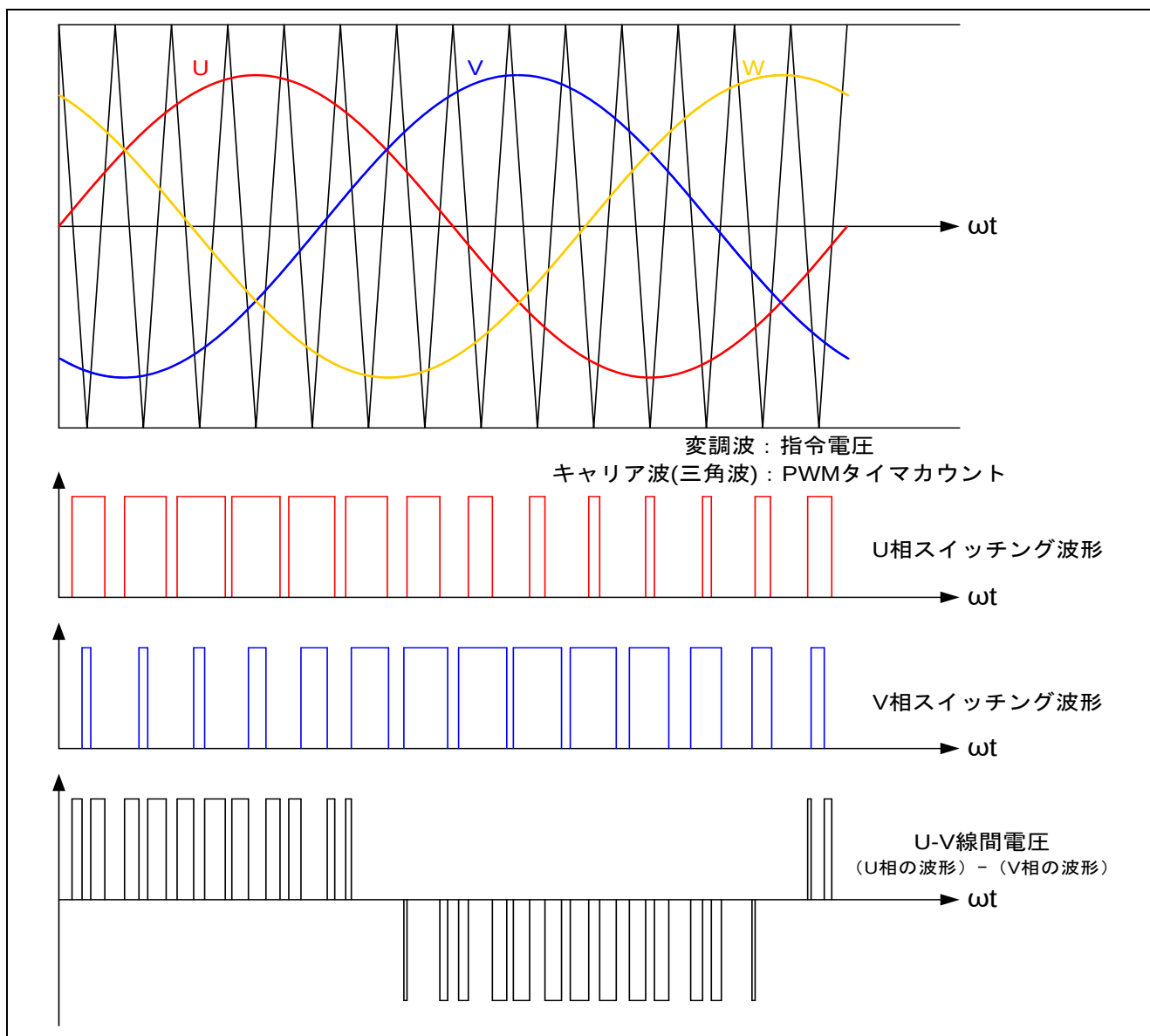


図 3-2 三角波比較法の概念図

(2). 3 次高調波重畳法

三角波比較法では線間電圧振幅が入力される直流電圧の約 86.6%までしか使用する事が出来ません。電圧利用効率を向上させるために数多の変調方法がありますが、本プログラムでは 3 次高調波重畳法を利用する事が出来ます。下記の様に指令電圧を計算する事で 3 次高調波を重畳したのと同じ指令電圧になります。

$$v_o = \frac{\max(v_u^*, v_v^*, v_w^*) + \min(v_u^*, v_v^*, v_w^*)}{2}$$

$$\overline{v_u^*} = v_u^* - v_o$$

$$\overline{v_v^*} = v_v^* - v_o$$

$$\overline{v_w^*} = v_w^* - v_o$$

v_u^*, v_v^*, v_w^* : 元の 3 相の指令電圧、

$\overline{v_u^*}, \overline{v_v^*}, \overline{v_w^*}$: 3 次高調波重畳法の 3 相の指令電圧

v_o : 最大指令電圧と最小指令電圧の平均値

(3). 二相変調

3 相のうち 2 相だけのスイッチングで制御する様に各相指令電圧にオフセット電圧を重畳する方法を二相変調と呼びます。本プログラムでは、最大指令電圧の相の指令電圧を DC リンク電圧と等しくなる様にオフセット電圧を重畳します。

$$v_o = -\frac{V_{dc}}{2} + \max(v_u^*, v_v^*, v_w^*)$$

$$\overline{v_u^*} = v_u^* - v_o$$

$$\overline{v_v^*} = v_v^* - v_o$$

$$\overline{v_w^*} = v_w^* - v_o$$

v_u^*, v_v^*, v_w^* : 元の UVW 相の指令電圧、

$\overline{v_u^*}, \overline{v_v^*}, \overline{v_w^*}$: 二相変調の 3 相の指令電圧

v_o : オフセット電圧

上記変調方式は r_mtr_config.h にある MOD_METHOD に下記の値を設定し、コンパイルする事で変更可能です。

MOD_3PH_SPWM	三角波比較法(未実装)	0
MOD_3PH_TOW	3 次高調波重畳法	1 : デフォルト設定
MOD_2PH_BOT	二相変調	2

3.1.6 状態遷移

図 3-3 に本プログラムにおける状態遷移図を示します。

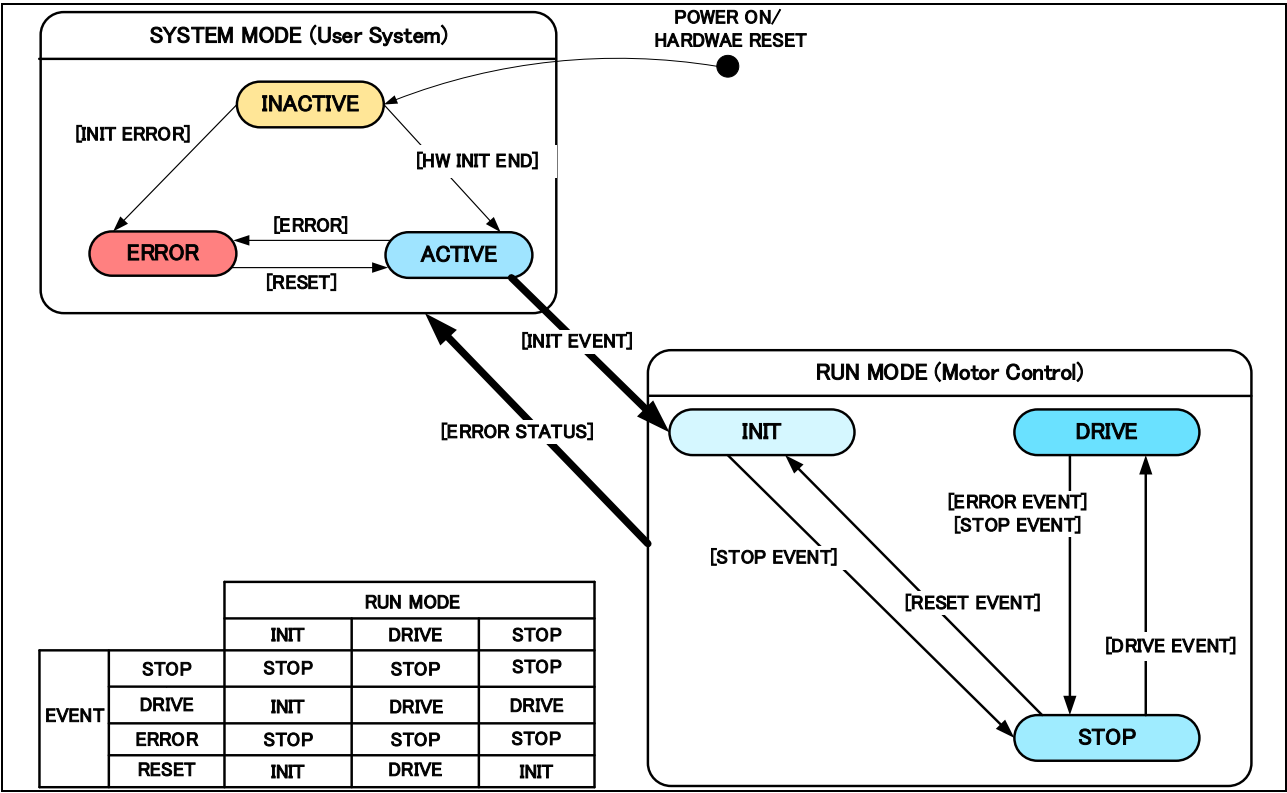


図 3-3 状態遷移図

- (1). SYSTEM MODE
システム動作状態を表します。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
モータの駆動状態を表します。各イベント(EVENT)の発生により状態が遷移します。
- (3). EVENT
モータの駆動状態の遷移を表します。EVENTが発生すると、RUN MODEが図 3-3 中の表の様に遷移します。各EVENTの発生要因を表 3-4 に示します。

表 3-4 EVENT 一覧

イベント名	発生条件
STOP	ユーザ操作により発生します
DRIVE	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

RUN MODE の DRIVE イベント内では DRIVE ステータスが表 3-5 の表の中からモータの駆動状態に合わせて遷移します。

表 3-5 DRIVE ステータス一覧

ステータス名	内容
MTR_STOP	モータ駆動停止
MTR_DRIVE	モータ駆動
MTR_DRIVE_INIT	モータ駆動初期化
MTR_DRIVE_START	モータ駆動開始
MTR_OFFSET_CALC_EXE	電流オフセットの検出処理の実行
MTR_DRIVE_END	駆動の停止

3.1.7 始動方法

センサレスベクトル制御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ステータスによってモードをコントロールしています。

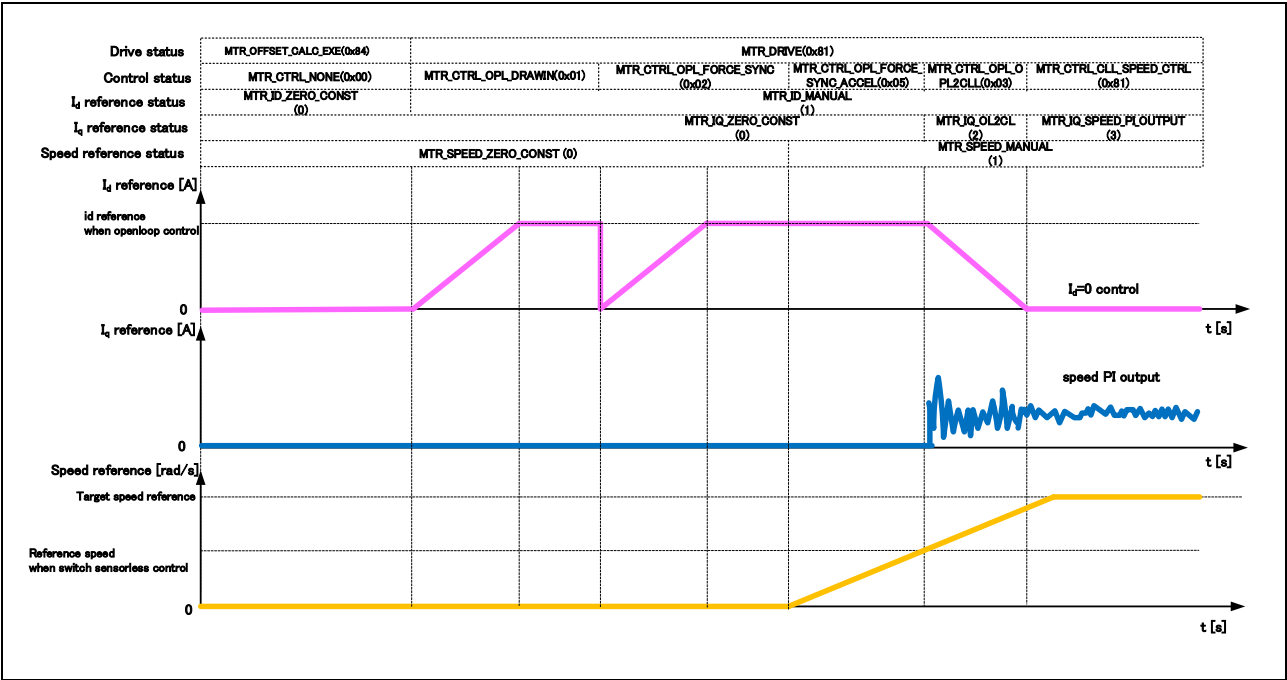


図 3-4 センサレス速度制御ソフトの始動制御内容

3.1.8 制御方法

制御システム全体のブロック図を図 3-5 に示します。制御システムは、座標変換器と非干渉制御器、軸誤差推定器、PLL 制御器、速度制御器（ASR）と電流制御器（ACR）から構成されます。また、推定速度に 1 次の LPF を設けています。

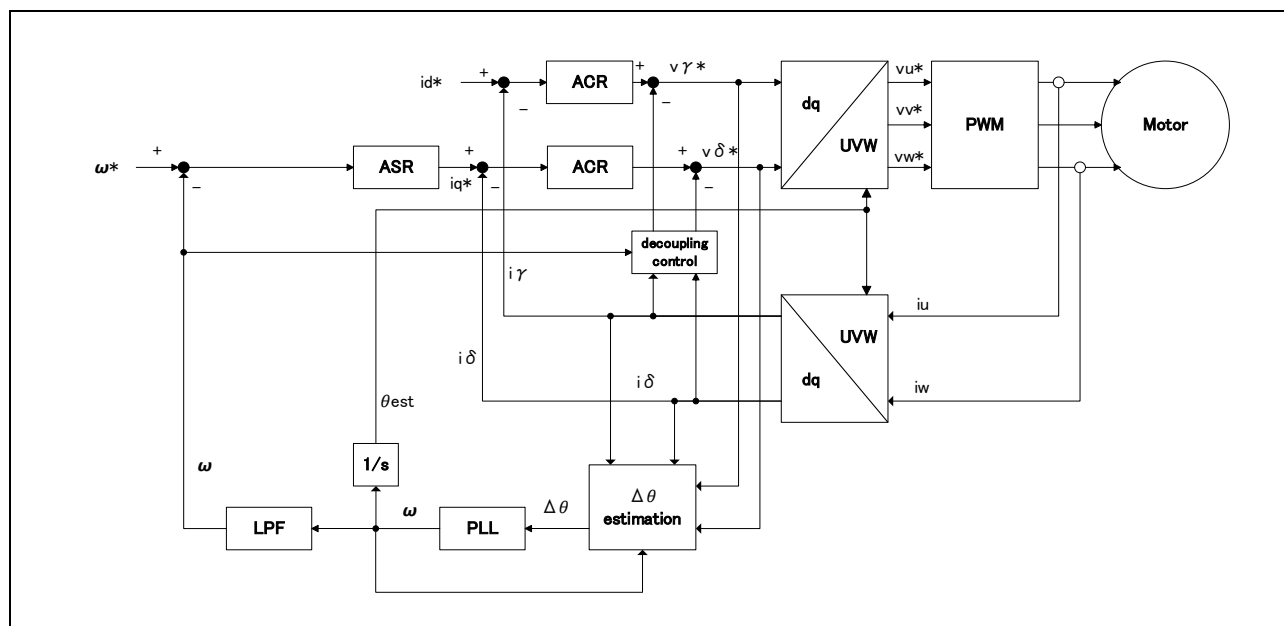


図 3-5 制御システムブロック図

軸誤差推定器は、実軸である dq 軸と推定軸である $\gamma\delta$ 軸との軸誤差 $\Delta\theta$ を推定します。 $\gamma\delta$ 軸における電圧方程式は、次式の dq 軸における電圧方程式の両辺に

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

v_d, v_q : d軸電圧, q軸電圧

ω : 角速度

i_d, i_q : d軸電流, q軸電流

L_d, L_q : dq軸インダクタンス

R_a : 抵抗

ψ_a : 永久磁石による電機子鎖交磁束の実効値

次式の回転行列を乗じて得る事が出来ます。

$$\begin{bmatrix} \cos\Delta\theta & -\sin\Delta\theta \\ \sin\Delta\theta & \cos\Delta\theta \end{bmatrix}$$

γ 軸と δ 軸の誘起電圧成分 e_γ と e_δ を計算し、次式により軸誤差 $\Delta\theta$ を得ます。誘起電圧の計算の際、 γ 軸電圧と δ 軸電圧をそれぞれ電圧指令値 v_γ^* と v_δ^* で近似して使います。この軸誤差が 0 になる様にフィードバックループを構成する（PLL 制御器）ことで推定速度 ω を求めます。

$$\Delta\theta = \text{atan}\left(\frac{e_\gamma}{e_\delta}\right)$$

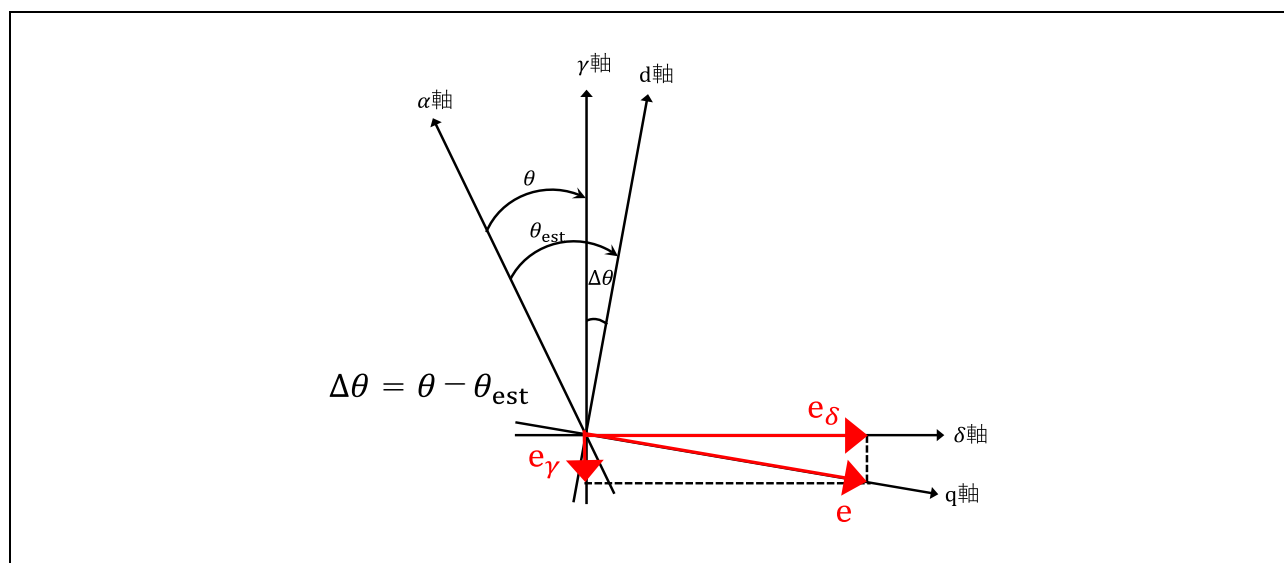


図 3-6 軸誤差とγ軸とδ軸の誘起電圧成分

速度制御器、電流制御器と PLL 制御器は PI 制御器を用いて実現されており、そのゲインは所望の制御に合わせて適宜調整する必要があります。電流 PI 制御ゲイン K_{pACR} 、 K_{iACR} と速度 PI 制御ゲイン K_{pASR} 、 K_{iASR} と PLL 制御ゲイン K_{pPLL} 、 K_{iPLL} はそれぞれ次式のようにになります。

$$K_{pACR} = \omega_{CG} L, \quad K_{iACR} = \omega_{CG} R$$

ω_{CG} : 電流 PI 制御固有周波数

L : インダクタンス (d 軸では L_d 、q 軸では L_q)

R : 抵抗

$$K_{pASR} = \frac{\omega_{SG} J}{P_n^2 \psi_a}, \quad K_{iASR} = \frac{\omega_{SG} D_1}{a P_n^2 \psi_a}$$

ω_{SG} : 速度 PI 制御固有周波数

J : イナーシャ

D_1 : 動摩擦係数

ψ_a : 誘起電圧定数

P_n : 極対数

a : 調整パラメータ

$$K_{pPLL} = \omega_{\Delta\theta}, \quad K_{iPLL} = \frac{\omega_{\Delta\theta}^2}{5}$$

$\omega_{\Delta\theta}$: PLL 制御固有周波数

3.1.9 システム保護機能

本プログラムは、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 3-6 を参照して下さい。

- ハードウェア過電流エラー
ハードウェアからの緊急停止信号（過電流検出）を検出した時に電圧出力を停止します。
- 過電流エラー
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。
- 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 3-6 各システム保護機能設定値

エラー	閾値	
ハードウェア過電流エラー	過電流リミット値 [A]	21.4
過電流エラー	過電流リミット値 [A]	2.26
	監視周期	電流制御/位置・速度推定周期
過電圧エラー	過電圧リミット値 [V]	48
	監視周期	速度制御周期
低電圧エラー	低電圧リミット値 [V]	8
	監視周期	速度制御周期
回転速度エラー	速度リミット値 [rpm]	8000
	監視周期	速度制御周期

3.1.10 単位法（PU）

固定小数点演算によるモータ制御はダイナミックレンジがコンパイル時に決まります。実際のモータの特性と設計時に想定したモータの特性との違いが大きければ、ダイナミックレンジの差の違いにより、桁あふれ、丸め誤差などの問題が起こり易くなります。本プログラムでは、演算のダイナミックレンジのモータ特性依存度を低減するために単位法（PU: per-unit）を使用します。任意の物理量の PU 値は基準となる物理量に対する相対的な値であり、以下のように計算できます。

$$PU \text{ 値} = \frac{\text{物理量}}{\text{基準値}}$$

制御に使用される物理量やゲインなどの PU 単位は全て電流、電圧、角周波数、角度の基準値から計算することが出来ます。例えば、抵抗基準値は以下のように電圧基準値と電流基準値で計算します。

$$\text{抵抗基準値} = \frac{\text{電圧基準値}}{\text{電流基準値}} = 1[\text{抵抗PU}]$$

演算ダイナミックレンジがモータの特性から受ける影響を低減するため、電流、電圧、角周波数の基準値はモータ特性を元に設定する必要があります（基準値の取り方は一意ではありません）。本プログラムでは、定格電流、インバータへの入力電圧、最大速度を電流、電圧、角周波数の基準値（PU 単位）に設定します。各物理量の基準値を表 3-7 に示します。これらの値は `r_mtr_scaling_parameter.h` の中で定義されています。

表 3-7 PU 単位系基準値

カテゴリ	項目	定義	単位
PU 基本物理量	電流	定格電流	[A]
	電圧	入力電圧（インバータ入力）	[V]
	角周波数	$2\pi \times \text{最大速度}[\text{rpm}] \times \text{極対数} / 60$	[Hz]
	角度	1	[rad]
物理量	時間	角度／角周波数	[s]
	抵抗	電圧／電流	[Ω]
	インダクタンス	抵抗／角周波数	[H]
	誘起電圧定数	電圧／角周波数	[V・s/rad]
	イナーシャ	誘起電圧定数×電流×(極対数／角周波数) ²	[kgm ² /(rad ²)]
電流制御	Kp	抵抗	[Ω]
	Kidt	抵抗	[Ω]
速度制御	Kp	電流／角周波数	[A/(rad/s)]
	Kidt	電流／角周波数	[A/(rad/s)]
PLL 制御	Kp	角周波数／角度	[Hz]
	Kidt	角周波数／角度	[Hz]

3.1.11 1 シャント抵抗電流測定方法

本プログラムで用いる1つのシャント抵抗による電流測定方法について説明します。

3.1.11.1 1 シャント抵抗電流測定タイミング

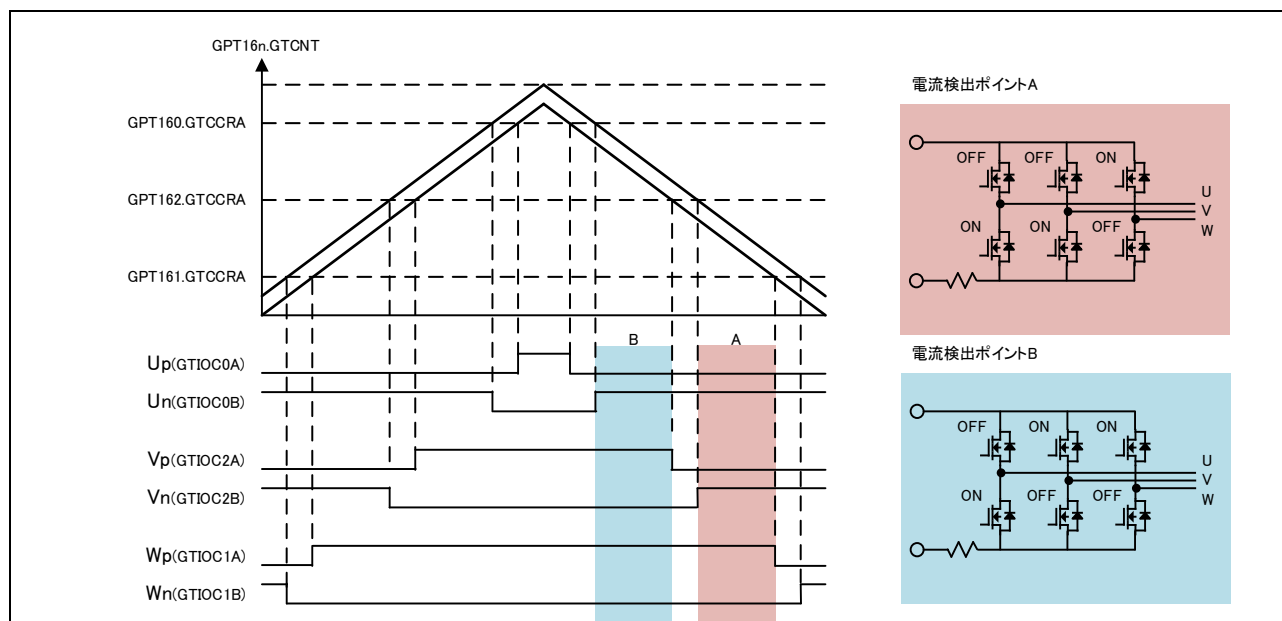


図 3-7 相補 PWM 波形図(例：デューティ大小関係 $W > V > U$)

本プログラムでは GPT を用いて、相補 PWM モードでのデッドタイム付き三相 PWM 出力による制御を行っています。図 3-7 に相補 PWM の波形(例：デューティ大小関係 $W > V > U$)を示します。

図のポイント A では右上の赤い枠に示したように上アームの W 相のみが ON している状態となります。この時、1 シャント抵抗に流れる電流は W 相に流れている電流ということになります。

$$I_A = I_w$$

図のポイント B では右下の青の枠に示したように下アームの U 相のみが ON している状態となります。この時、1 シャント抵抗に流れる電流は U 相に流れている電流ということになります。但し、制御に使用する場合、モータへ流入する方向を電流の正方向としているので、ここでは正負反転が必要となります。

$$I_B = -I_u$$

残りの 1 相となるポイント C は V 相に流れる電流となるので、キルヒホッフの第一法則より U 相と W 相との合成電流として求めることができます。

$$I_C = -(I_A + I_B) = I_v$$

従って、ポイント A, B における 1 シャント抵抗に流れている電流値を取得することが出来れば、三相の電流を復元することが出来ます。

これはデューティの大小関係が $W > V > U$ となる場合のため、PWM の変化に伴いこの大小関係は 6 パターンの組み合わせでポイント A, B で検出できる電流は切り替わります。デューティを設定する時点でこの大小関係はわかっているため、それに合わせて、検出した電流の相を判断します。

3.1.11.2 1 ショント抵抗電流測定方法

3.1.11.1 に示した様に 1 ショント抵抗による電流測定を行う場合、A/D コンバータの変換タイミングを PWM のデューティ設定に従って調整する必要があります。本プログラムでは相補 PWM モードの機能を用いることでこれを実現しています。

- ・ 汎用 PWM タイマ GPT :
相補 PWM モード 3(非対称三相波形を出力)、A/D 変換トリガ A,B
- ・ A/D コンバータ :
シングルスキャンダブルトリガーモード

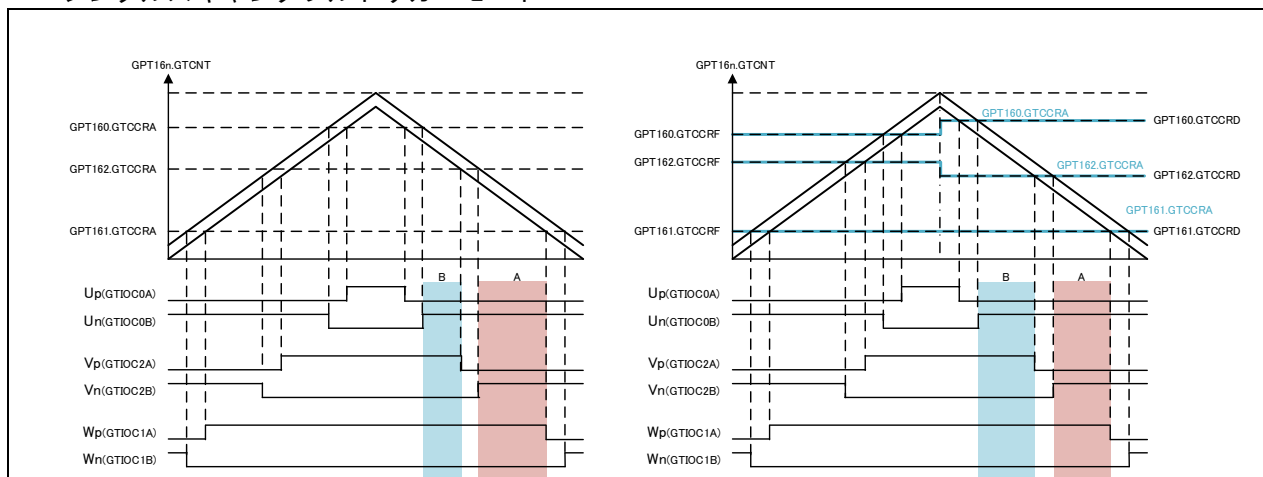


図 3-8 A/D 変換タイミング

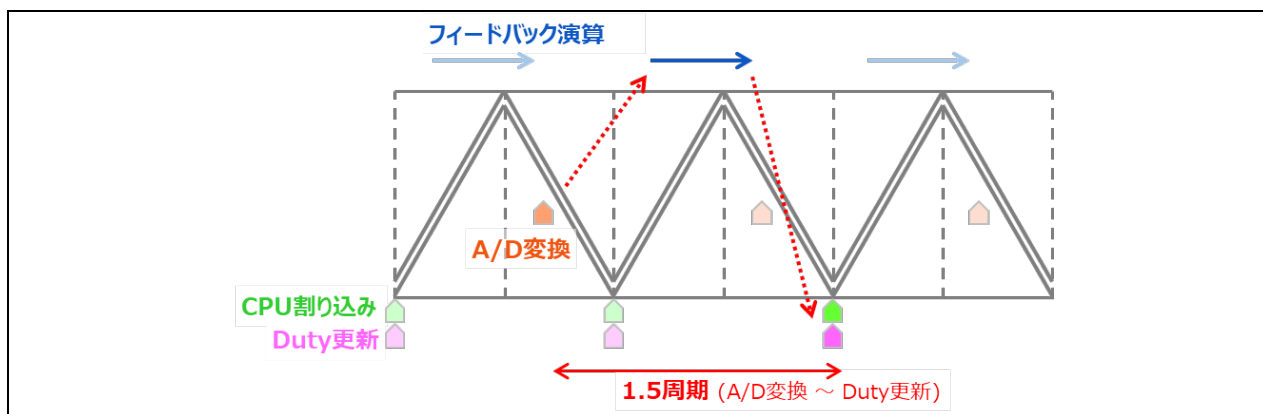


図 3-9 割り込みタイミング

図 3-8、図 3-9 を用いて動作を説明します。

GPT の設定によりキャリアの谷で割り込みを発生させ次のデューティを算出します。A/D 変換タイミングはポイント B の左エッジ、ポイント A の左エッジをそれぞれ GPT の A/D 変換開始要求 B、A/D 変換開始要求 A に設定します。

A/D コンバータは GPT の A/D 変換開始要求 A,B をハードウェア・トリガとして A/D 変換を行います。A/D 変換結果は次の周期の演算で使用され、デューティへの反映は次周期の谷(1.5 周期後)の割り込み時に一斉書き換えで行われます。

A/D データレジスタと A/D データ 2 重化レジスタ(ADDBLDRn)に格納された 2 点の A/D 変換結果から三相電流を復元することが出来ます。2 点の AD 変換時間の間、PWM 波形を一定に保てない場合、相補 PWM モードの非対称三相波形を出力することで AD 変換時間を確保します。

3.1.11.3 デューティ調整

各相のデューティ値の差分が小さい場合、A/D変換が間に合わない場合があります。そのため、A/D変換に必要な時間を確保すべく、非対称波形を出力することで、デューティを下図のように調整します。

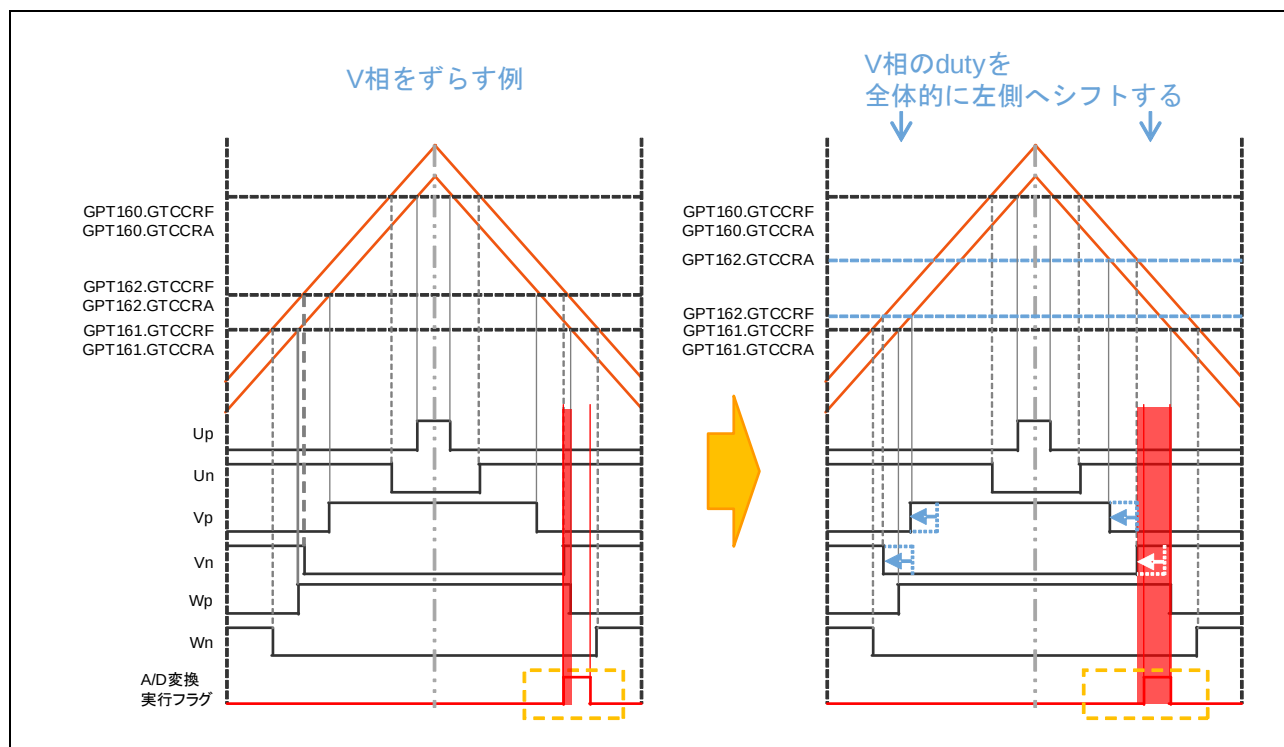


図 3-10 デューティ調整(例)

3.1.12 オープンループ制御

低速域では誘起電圧が小さい為、位置推定が出来ません。そこで d 軸電流を用いて回転磁界を発生させてモータを位置推定が可能な速度まで強制同期駆動させます(オープンループ制御)。その時モータの速度は電流やモータパラメータに依存する固有周波数で振動します。そこで図 3-11 のブロック線図にあるようなダンピング制御を行うことで低速域のオープンループ制御時におけるモータの振動を低減します。

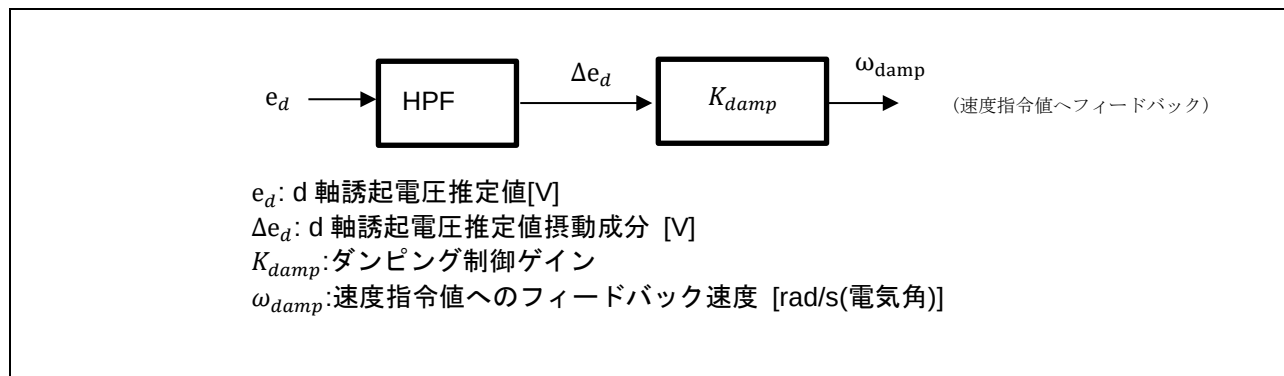


図 3-11 オープンループダンピング制御ブロック線図

3.1.13 センサレス切り替え処理

オープンループ制御からクローズドループ制御に切り替える際、オープンループ時は d 軸と γ 軸の軸誤差によりトルクが発生してモータが回転します。特に高負荷時などでは、軸誤差が大きくクローズドループ制御に移行時に電流や推定速度にハンチングが起き制御が不安定になる場合があります。この現象を低減するため、速度 PI 制御に移行する前にこの軸誤差を 0 にするよう q 軸電流を調整します。

3.2 センサレスベクトル制御ソフト関数仕様

本プログラムの関数一覧を以下に示します。

表 3-8 “mtr_main.c”関数一覧

ファイル	関数	処理概要
mtr_main.c	mtr_main 入力：なし 出力：なし	<ul style="list-style-type: none"> ・ハードウェア初期化関数呼び出し ・Renesas Motor Workbench 通信初期化 ・ユーザインタフェース初期化関数呼び出し ・メイン処理使用変数初期化関数呼び出し ・状態遷移及びイベント実行関数呼び出し ・母線電圧安定待ち処理呼び出し ・メイン処理 ⇒ユーザインタフェース処理呼び出し ⇒ウォッチドッグタイマクリア関数呼び出し
	ics_ui 入力：なし 出力：なし	Renesas Motor Workbench 使用 ・モータステータスの変更
	system_init 入力：なし 出力: int :: 実行結果	メイン処理にて使用するシステムの初期化
	software_term 入力：なし 出力：なし	システムの終了処理
	software_init 入力：なし 出力：なし	メイン処理にて使用する変数の初期化

表 3-9 “r_mtr_ics.c” 関数一覧

ファイル	関数	処理概要
r_mtr_ics.c	R_MTR_SetIcs 入力:なし 出力:なし	RMW への通信準備
	R_MTR_SetCOMVariables 入力:なし 出力:なし	制御変数への入力受け渡し前処理 <ul style="list-style-type: none"> ・制御変数の書き換え変数(com 変数)の値を制御受け渡しバッファ変数(ICS 変数)へ入力 ・ICS 変数の値を ICS バッファ変数へ入力
	R_MTR_ICSVVariablesInit 入力:なし 出力:なし	com 変数の初期化
	R_MTR_ICSIIntLevel 入力: uint8_t u1_level :: 割り込み優先順位 出力: なし	ICS の割り込み優先順位設定
	mtr_limit 【inline function】 入力: int32_t s4_value :: 対象の値 int32_t s4_max :: 最大値 int32_t s4_min :: 最小値 出力: int32_t :: リミット処理値	最大値、最小値のリミット処理

表 3-10 “ICS2_RA2T1.o” 関数一覧

ファイル	関数	処理概要
ICS2_RA2T1.o	ics2_watchpoint 入力: なし 出力: なし	転送関数の呼び出し 300us 以上の間隔で呼び出す必要があります。
	ics2_init 入力: uint16_t addr :: DTC ベクタテーブル先頭アドレス uint16_t pin :: SCI 使用ピン uint8_t level :: 割り込みレベル uint8_t num :: DTC の構造体の先頭番地 uint8_t brr :: 通信速度 uint8_t mode :: 通信モード 出力: なし	通信初期化
	ics2_version 入力: なし 出力: uint32_t :: バージョン番号	バージョン番号の取得

表 3-11 “r_mtr_board.c” 関数一覧

ファイル	関数	処理概要
r_mtr_board.c	RM_MTR_BoardLedContrl 入力: uint8_t u1_motor_status :: モータステータス uint8_t u1_system_status :: システムステータス 出力: なし	LED の制御

表 3-12 “r_mtr_ADC.c” 関数一覧

ファイル	関数	処理概要
r_mtr_ADC.c	R_MTR_ADC_Open 入力: adc_ctrl_t *p_ctrl :: ADC 情報 adc_cfg_t const *const p_cfg :: コンフィグ情報 出力: fsp_err_t :: エラー番号	ADC の初期化
	R_MTR_ADC_WaitConversion 入力: adc_ctrl_t *p_ctrl :: ADC 情報 uint16_t timeout_count :: タイムアウトカウンタ 出力: fsp_err_t :: エラー番号	AD 変換のウェイト
	R_MTR_ADC_SetChannels 入力: adc_ctrl_t *p_ctrl :: ADC 情報 uint32_t scan_mask_a :: AD チャネル選択レジスタ A uint32_t scan_mask_b :: AD チャネル選択レジスタ B 出力: fsp_err_t :: エラー番号	ADC チャネルの設定
	R_MTR_ADC_ReadDuplex 入力: adc_ctrl_t *p_ctrl :: ADC 情報 uint16_t *const p_data :: リードしたデータ 出力: fsp_err_t :: エラー番号	2 重化データのリード
	R_MTR_ADC_SetSampleStateCount 入力: adc_ctrl_t *p_ctrl :: ADC 情報 出力: fsp_err_t :: エラー番号	サンプリングステートレジスタの設定

表 3-13 “r_mtr_GPT_THREE_PHASE.c” 関数一覧

ファイル	関数	処理概要
r_mtr_GPT_THREE_PHASE.c	R_MTR_GPT_THREE_PHASE_CompMode2_Open 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 three_phase_cfg_t const * const p_cfg :: コンフィグ情報 出力: fsp_err_t :: エラー番号	3 相タイマモジュールを相補 PWM モード 2 で初期化
	R_GPT_THREE_PHASE_CompMode2_DutyCycleSet 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 three_phase_duty_cycle_t * const p_duty_cycle :: デューティサイクル情報 出力: fsp_err_t :: エラー番号	相補 PWM モード 2 の 3 相タイマモジュールのデューティサイクル設定
	R_MTR_GPT_THREE_PHASE_CompMode3_Open 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 three_phase_cfg_t const * const p_cfg :: コンフィグ情報 出力: fsp_err_t :: エラー番号	3 相タイマモジュールを相補 PWM モード 3 で初期化
	R_MTR_GPT_THREE_PHASE_CompMode3_DutyCycleSet 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 three_phase_duty_cycle_t * const p_duty_cycle :: デューティサイクル情報 出力: fsp_err_t :: エラー番号	相補 PWM モード 3 の 3 相タイマモジュールのデューティサイクル設定
	R_MTR_GPT_THREE_PHASE_SetDeadTime 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 uint16_t up_count :: アップカウント uint16_t down_count :: ダウンカウント 出力: fsp_err_t :: エラー番号	3 相タイマモジュールのデッドタイム値設定
	R_MTR_GPT_THREE_PHASE_SetADCTrigger 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 uint16_t a_down_count :: A/D 変換開始要求タイミング A uint16_t b_down_count :: A/D 変換開始要求タイミング B 出力: fsp_err_t :: エラー番号	相補 PWM モード 3 で 3 相タイマモジュールの ADC トリガ設定
	R_MTR_GPT_THREE_PHASE_EnableOutputAll 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 出力: fsp_err_t :: エラー番号	相補 PWM モード 3 で 3 相タイマモジュールの全出力を有効化
	R_MTR_GPT_THREE_PHASE_DisableOutputAll 入力: three_phase_ctrl_t * const p_ctrl :: 3 相制御情報 出力: fsp_err_t :: エラー番号	相補 PWM モード 3 で 3 相タイマモジュールの全出力を無効化

表 3-14 “r_mtr_POEG.c” 関数一覧

ファイル	関数	処理概要
r_mtr_POEG.c	R_MTR_POEG_SoftwareRelease 入力: poeg_ctrl_t * const p_ctrl :: POEG 制御情報 uint16_t timeout_count :: タイムアウト値 出力: fsp_err_t :: エラー番号	POEG モジュールの初期化
	R_MTR_POEG_RecoverForcedShutdown 入力: poeg_ctrl_t * const p_ctrl :: POEG 制御情報 uint16_t timeout_count :: タイムアウト値 uint8_t * p_is_shutdown :: シャットダウン状態 出力: fsp_err_t :: エラー番号	POEG モジュールを強制シャットダウンの状態から回復

表 3-15 “r_mtr_ra2t1.c” 関数一覧

ファイル	関数	処理概要
r_mtr_ra2t1.c	R_MTR_InitUnusedPins 入力: なし 出力: なし	処理なし

表 3-16 “rm_mtr_ctrl_gain.o” 関数一覧

ファイル	関数	処理概要
rm_mtr_ctrl_gain.o	RM_MTR_CtrlGain 入力: st_mtr_ctrl_gain_t *st_gain_buf :: ゲイン設計構造体ポインタ const st_mtr_design_parameter_t *st_ctrl_param :: 設計パラメータ構造体ポインタ 出力: なし	ゲイン設計処理

表 3-17 “rm_mtr_driver.c” 関数一覧(1/2)

ファイル	関数	処理概要
rm_mtr_driver.c	RM_MTR_ControlInit 入力: なし 出力: なし	モータ制御システム初期化 ・ モータステータスの初期化 ・ 制御用変数の初期化
	RM_MTR_EventExec 入力: uint8_t u1_event :: イベント 出力: なし	モータステータスの変更とイベント処理
	RM_MTR_CapacitorCharge 入力: なし 出力: uint16_t :: タイムアウトエラー	母線電圧安定待ち処理
	RM_MTR_SpeedSet 入力: int32_t s4_ref_speed_rpm :: 目標回転速度 int32_t s4_ref_min_speed_rpm :: 目標回転速度の最小値 出力: uint8_t :: モータ停止フラグ	速度指令値の設定
	RM_MTR_SpeedGet 入力: なし 出力: int32_t :: 回転速度	速度の取得
	RM_MTR_DirSet 入力: int8_t s1_dir :: 回転方向 出力: なし	回転方向設定
	RM_MTR_DirGet 入力: なし 出力: int8_t :: 回転方向	回転方向取得
	RM_MTR_StatusGet 入力: なし 出力: uint8_t :: モータステータス	モータステータス取得
	RM_MTR_StatusErrorGet 入力: なし 出力: uint16_t :: エラーステータス	エラーステータス取得
	RM_MTR_IcsBufferSet 入力: なし 出力: なし	ICS 変数の値を ICS バッファ変数へ入力
	RM_MTR_IcsVariablesSet 入力: なし 出力: なし	ICS バッファ変数の値を制御変数へ入力
	RM_MTR_InputBuffParamReset 入力: なし 出力: なし	ICS バッファ変数のリセット
	RM_MTR_UpdatePollingSet 入力: なし 出力: なし	制御変数の設定

表 3-18 “rm_mtr_driver.c” 関数一覧(2/2)

ファイル	関数	処理概要
rm_mtr_driver.c	mtr_statemachine_init 入力:(st_mtr_statemachine_t) *p_state_machine :: モータステータス構造体 出力:なし	モータステータスの初期化
	mtr_statemachine_reset 入力:(st_mtr_statemachine_t) *p_state_machine :: モータステータス構造体 出力:なし	モータステータスのリセット
	mtr_state_machine_event 入力:(st_mtr_statemachine_t) *p_state_machine :: モータステータス構造体 (void) *p_object :: 制御変数用構造体 (uint8_t) u1_event :: イベント 出力:(uint8_t) p_state_machine->u2_error_status :: エラーステータス	イベントの実行
	mtr_statemachine_get_status 入力:(st_mtr_statemachine_t) *p_state_machine :: モータステータス構造体 出力:(uint8_t) p_state_machine->u1_status :: モータステータス	モータステータスの取得
	mtr_act_none 入力:(st_mtr_statemachine_t) *st_stm :: モータステータス構造体 (void) *p_param :: 制御変数用構造体 出力: (uint8_t) action_ret :: 実行結果(常に 0)	処理なし
	mtr_act_init 入力:(st_mtr_statemachine_t) *st_stm :: モータステータス構造体 (void) *p_param :: 制御変数用構造体 出力: (uint8_t) action_ret :: 実行結果(常に 0)	制御変数の初期化
	mtr_act_error 入力:(st_mtr_statemachine_t) *st_stm :: モータステータス構造体 (void) *p_param :: 制御変数用構造体 出力: (uint8_t) action_ret :: 実行結果(常に 0)	エラー検出時の後処理
	mtr_act_drive 入力:(st_mtr_statemachine_t) *st_stm :: モータステータス構造体 (void) *p_param :: 制御変数用構造体 出力: (uint8_t) action_ret :: 実行結果(常に 0)	モータの駆動開始
	mtr_act_stop 入力:(st_mtr_statemachine_t) *st_stm :: モータステータス構造体 (void) *p_param :: 制御変数用構造体 出力: (uint8_t) action_ret :: 実行結果(常に 0)	モータの停止

表 3-19 “rm_mtr_foc_less_speed.c”関数一覧

ファイル名	関数	処理概要
rm_mtr_foc_less_speed.c	RM_MTR_FOCMotorDefaultInit 入力: st_mtr_common_foc_t *p_st_foc_common :: センサレス FOC メインデータ構造体ポインタ 出力:なし	制御変数の初期化
	RM_MTR_FOCMotorReset 入力: st_mtr_common_foc_t *p_st_foc_common :: センサレス FOC メインデータ構造体ポインタ 出力:なし	制御変数のリセット

表 3-20 “rm_mtr_carrier.c”関数一覧(1/4)

ファイル	関数	処理概要
rm_mtr_carrier.c	mtr_carrier_cyclic 入力: timer_callback_args_t * p_args :: 割り込みハンドラ引数ポインタ[3 シャントモード時] adc_callback_args_t * p_args :: 割り込みハンドラ引数ポインタ[3 シャントモード時] 出力: なし	キャリア割り込みハンドラ 周期タイマ割り込み (GPT で呼び出し) 周期: 100[μs] [1 シャントモード時] (50[μs]の1回間引き) 50[μs] [3 シャントモード時] ・ベクトル制御関数の呼び出し ・インバータ母線電圧の取得 ・通信処理呼び出し ・ベクトル制御準備関数の呼び出し
	mtr_carrier_foc_sequence 【inline function】 入力: なし 出力: なし	ベクトル制御関数
	mtr_carrier_vdc_get 入力: int32_t * p_s4_vdc_ad :: VDC ポインタ st_mtr_mod_t * p_st_mod :: モジュール構造体ポインタ 出力: なし	インバータ母線電圧検出 ・A/D 変換値からインバータ母線電圧を計算
	mtr_carrier_current_ss_get 【inline function】 入力: st_mtr_sscs_t * p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ st_coordinate28 * p_st_ad_i :: UVW 相検出電流ポインタ st_coordinate28 * p_st_repro_i :: 再現電流座標系構造体ポインタ 出力: なし	電流検出 [1 シャントモード時] ・A/D 変換値から相電流を計算 ・2 相近接補償時に検出電流を再現電流座標で置換
	mtr_carrier_current_ts_get 【inline function】 入力: st_mtr_tscs_t * p_st_tscs :: 3 シャント電流検出構造体ポインタ st_coordinate28 * p_st_ad_i :: UVW 相検出電流ポインタ 出力: なし	電流検出 [3 シャントモード時] ・A/D 変換値から相電流を計算
	mtr_carrier_lim_sw_over_current 【inline function】 入力: st_coordinate28 * p_st_ad_i :: UVW 相検出電流ポインタ 出力: uint16_t :: エラーステータス	ソフトウェア過電流エラーチェック
	mtr_carrier_uvw2dq 【inline function】 入力: st_coordinate28 * p_st_ad_i :: ADC 電流座標系構造体ポインタ st_sincos28 * p_ad_i_angle :: 角度構造体ポインタ 出力: なし	検出電流情報の変換 (U,V,W 相→d,q 軸)
	mtr_carrier_est_phase_err 【inline function】 入力: st_coordinate28 * p_st_ad_i :: ADC 電流座標系構造体ポインタ st_mtr_parameter_t * p_st_motor :: モータパラメータ構造体ポインタ st_mtr_estimate_phe_t * p_st_phe :: 位相誤差推定構造体ポインタ int32_t * p_s4_vdq :: dq 軸指令電圧ポインタ st_mtr_asr_t * p_st_asr :: ASR 構造体ポインタ 出力: int32_t :: 軸誤差推定値	軸誤差推定処理
	mtr_carrier_est_speed_cli 【inline function】 入力: st_mtr_pll_t * p_st_pll :: PLL 制御構造体ポインタ st_mtr_asr_t * p_st_asr :: ASR 構造体ポインタ int32_t * p_ad_i_angle_rad :: 検出電流の角度情報 int32_t * p_ref_v_angle_rad :: 指令電圧の角度情報 st_mtr_common_foc_t * p_st_common :: センサレス FOC メインデータ構造体ポインタ 出力: なし	クローズドループ時速度推定

表 3-21 “rm_mtr_carrier.c”関数一覧(2/4)

ファイル	関数	処理概要
rm_mtr_carrier.c	mtr_carrier_est_speed_opl 【inline function】 入力: st_mtr_estimate_phe_t *p_st_phe :: 位相誤差推定構造体ポインタ st_mtr_asr_t *p_st_asr :: ASR 構造体ポインタ st_mtr_pll_t *p_st_pll :: PLL 制御構造体ポインタ int32_t *p_ad_i_angle_rad :: 検出電流の角度情報 int32_t *p_ref_v_angle_rad :: 指令電圧の角度情報 出力: なし	オープンループ時速度推定
	mtr_carrier_get_direction 【inline function】 入力: int32_t p_s4_direction :: 回転方向 出力: int8_t :: 回転方向	モータの回転方向取得 CW:1、CCW:-1
	mtr_carrier_ctrl_pi 【inline function】 入力: int32_t *p_s4_ad_idq :: ADC 電流 st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_mtr_acr_t *p_st_acr :: ACR 構造体ポインタ 出力: なし	電流 PI 制御
	mtr_carrier_ctrl_decoupling 【inline function】 入力: st_mtr_parameter_t *p_st_motor :: モータパラメータ構造体ポインタ int32_t *p_s4_ad_idq :: ADC 電流 st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_mtr_asr_t *p_st_asr :: ASR 構造体ポインタ 出力: なし	非干渉制御
	mtr_carrier_dq2uvw_voltage 【inline function】 入力: st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ 出力: なし	指令電圧情報の変換 (d,q 軸→U,V,W 相)
	mtr_carrier_set_pwm_ss 【inline function】 入力: st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_mtr_mod_t *p_st_mod :: 変調構造体ポインタ st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ 出力: なし	PWM 出力設定 [1 シャントモード時]
	mtr_carrier_set_pwm_ts 【inline function】 入力: st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_mtr_mod_t *p_st_mod :: 変調構造体ポインタ st_mtr_tscs_t *p_st_tscs :: 3 相電流検出構造体ポインタ st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ 出力: なし	PWM 出力設定 [3 シャントモード時]

表 3-22 “rm_mtr_carrier.c”関数一覧(3/4)

ファイル	関数	処理概要
rm_mtr_carrier.c	mtr_carrier_conv_ad_ss 【inline function】 入力: st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ 出力: なし	A/D 変換タイミングの計算 [1 シャントモード時]
	mtr_carrier_2phase_duty_cross 【inline function】 入力: st_coordinate28 *p_st_ad_i :: ADC 電流座標系構造体ポインタ st_coordinate28 *p_st_repro_i :: 再現電流座標系構造体ポインタ st_mtr_lpf1_t *p_st_repro_id :: 電流再現用 d 軸電流 LPF ポインタ st_mtr_lpf1_t *p_st_repro_iq :: 電流再現用 q 軸電流 LPF ポインタ 出力: なし	2 相近接の電流補償
	mtr_carrier_deadtime_comp 【inline function】 入力: st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_mtr_s4_comp_t *p_st_s4_comp :: デッドタイム補償パラメータ構造体ポインタ 出力: なし	デッドタイム補償処理
	mtr_carrier_mod_ss 【inline function】 入力: st_mtr_mod_t *p_st_mod :: 変調構造体ポインタ st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ uint8_t *p_u1_drv_pat :: 変調率大小関係パターンのポインタ st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ 出力: なし	変調処理 [1 シャントモード時]
	mtr_carrier_mod_ts 【inline function】 入力: st_mtr_mod_t *p_st_mod :: 変調構造体ポインタ st_coordinate28 *p_st_ref_v :: 指令電圧座標系構造体ポインタ st_coordinate28 *p_st_ref_i :: 指令電流座標系構造体ポインタ 出力: なし	変調処理 [3 シャントモード時]
	mtr_carrier_pwm_duty_ss 【inline function】 入力: st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ 出力: なし	デューティ算出 [1 シャントモード時]
	mtr_carrier_pwm_duty_nonadc_ss 【inline function】 入力: st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ 出力: なし	デューティ算出 [1 シャントモード時]
	mtr_carrier_pwm_duty_ts 【inline function】 入力: st_mtr_tscs_t *p_st_tscs :: 3 相電流検出構造体ポインタ st_mtr_mod_t *p_st_mod :: 変調構造体ポインタ 出力: なし	デューティ算出 [3 シャントモード時]
	mtr_carrier_set_duty_adj_ss 【inline function】 入力: st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ 出力: なし	デューティ設定 [1 シャントモード時]
	mtr_carrier_lim_uvw_voltage 【inline function】 入力: int32_t *p_s4_ref_v_uvw :: UVW 相電圧 int32_t s4_voltage_limit :: 電圧リミット値 出力: なし	3 相電圧の制限処理

表 3-23 “rm_mtr_carrier.c”関数一覧(4/4)

ファイル	関数	処理概要
rm_mtr_carrier.c	mtr_carrier_startup_sequence 【inline function】 入力: なし 出力: なし	ベクトル制御の準備
	mtr_carrier_offset_ss 【inline function】 入力: st_mtr_sscs_t *p_st_sscs :: 1 シャント抵抗電流検出構造体ポインタ 出力: uint8_t :: ドライブモードのステータス	電流オフセット検出 [1 シャントモード時]
	mtr_carrier_offset_ts 【inline function】 入力: st_mtr_tscs_t *p_st_tscs :: 3 相電流検出構造体ポインタ 出力: uint8_t :: ドライブモードのステータス	電流オフセット検出 [3 シャントモード時]
	mtr_limit 【inline function】 入力: int32_t s4_value :: 対象の値 int32_t s4_max :: 最大値 int32_t s4_min :: 最小値 出力: int32_t :: リミット処理値	最大値、最小値のリミット処理
	mtr_limit_or_zero 【inline function】 入力: int16_t s2_value :: 対象の値 int16_t s2_max :: 最大値 int16_t s2_min :: 最小値 int16_t s2_zero_line :: 0 補正閾値 出力: int16_t :: リミット処理値	0 補正付き最大値、最小値のリミット処理
	mtr_lpf1_run 【inline function】 入力: st_mtr_lpf1_t *p_st_lpf :: LPF 構造体ポインタ int32_t s4_input :: LPF 入力 const uint8_t u1_q :: LPF の Q 値 出力: int32_t :: LPF 出力	1 次 LPF 処理
	mtr_pi_run_acr 【inline function】 入力: st_mtr_pi_t *p_st_pi :: PI 制御構造体ポインタ int32_t s4_err :: 偏差 const uint8_t u1_kp_q :: 比例ゲイン Q 値 const uint8_t u1_q_err_shift :: エラー番号のシフト値 const uint8_t u1_q_kidt_shift :: 積分ゲイン Q 値 const uint8_t u1_q_intg_shift :: 整数部のシフト値 出力: int32_t :: PI 出力	電流 PI 制御処理
	mtr_abs 【inline function】 入力: int32_t s4_value :: 入力値 出力: int32_t :: 出力値	入力の絶対値を出力
	mtr_limit_abs 【inline function】 入力: int32_t s4_value :: 入力値 int32_t s4_limit_value :: 制限値 出力: int32_t :: 出力値	入力を絶対値で制限
	mtr_carrier_pwm_duty_calc 【inline function】 入力: int32_t s4_value :: デューティ値 出力: int16_t :: 計算値	PWM デューティ値の計算

表 3-24 "rm_mtr_est_phase_err.o"関数一覧

ファイル	関数	処理概要
rm_mtr_est_phase_err.o	RM_MTR_ESTIMATE_AngleSpeedGet 入力: st_mtr_est_phe_t *st_phe :: 軸誤差推定構造体ポインタ int32_t s4_est_speed :: 速度 st_mtr_est_vect_t *p_st_est_vect :: 位相誤差推定構造体ポインタ 出力: int32_t :: 位相誤差	軸誤差推定処理

表 3-25 "RM_DSP_RA2_GCC_32BIT.o"関数一覧

ファイル	関数	処理概要																		
RM_DSP_RA2_GCC_32BIT.o	R_motor_uw2dq_abs_FIX28 入力: st_coordinate28 *p_uw2dq :: 座標変換構造体ポインタ 入力範囲 : $-4\sqrt{2/3} \leq u, w \leq 4\sqrt{2/3}$ (FIX28) 出力:なし 出力範囲: $-4 \leq d, q \leq 4$ (FIX28)	Clark 変換と Park 変換の剛性変換(絶対変換) U 相(FIX28 u)と W 相(FIX28 w)の値を d 軸と q 軸の値に変換し、それぞれ(FIX28 d)と(FIX28 q)に格納																		
	R_motor_dq2uvw_abs_FIX28 入力: st_coordinate28 *p_dq2uvw :: 座標変換構造体ポインタ 入力範囲 : $-4 \leq d, q \leq +4$ (FIX28) $-1 \leq \text{angle.sin}, \text{angle.cos} \leq 1$ (FIX28) 出力:なし 出力範囲: $-4\sqrt{2/3} \leq U, V, W \leq 4\sqrt{2/3}$ (FIX28)	逆 Park+逆 Clark 変換 d 軸(FIX28 d)と q 軸(FIX28 q)の値を UVW 相の値に変換し、それぞれ(FIX28 u)、(FIX28 v)、(FIX28 w)に格納																		
	R_motor_sincos_FIX28 入力: st_sincos28 *p_sincos :: 角度構造体ポインタ 入力範囲 : $-2\pi \leq \text{angle_rad} \leq 2\pi$ (FIX28) 出力:なし 出力範囲: $-1 \leq \sin, \cos \leq 1$ (FIX28)	角度(FIX28 angle)から正弦と余弦の値を計算し、それぞれ(FIX28 sin) と(FIX28 cos)に格納																		
	R_motor_atan2_FIX28 入力 : FIX28 x :: 入力値 x FIX28 y :: 入力値 y 入力範囲 : $-4 \leq x, y \leq 4$ (FIX28) 出力 : FIX28 atan(y/x) 出力範囲: $-\pi \leq \text{atan} \leq \pi$ (FIX28)	入力値 x (FIX28 x)と y (FIX28 y)から (y/x)に対する逆正接の主値を計算し (FIX28)の型で出力 入力に 0 を含む場合の出力 <table><tr><td>x</td><td>y</td><td>atan</td></tr><tr><td>0</td><td>正</td><td>$\pi/2$</td></tr><tr><td>0</td><td>負</td><td>$-\pi/2$</td></tr><tr><td>0</td><td>0</td><td>$\pi/4$</td></tr><tr><td>正</td><td>0</td><td>0</td></tr><tr><td>負</td><td>0</td><td>$-\pi$</td></tr></table>	x	y	atan	0	正	$\pi/2$	0	負	$-\pi/2$	0	0	$\pi/4$	正	0	0	負	0	$-\pi$
	x	y	atan																	
	0	正	$\pi/2$																	
0	負	$-\pi/2$																		
0	0	$\pi/4$																		
正	0	0																		
負	0	$-\pi$																		
R_motor_sqrt_sum_FIX28 入力 : FIX28 x :: 入力値 x FIX28 y :: 入力値 y 入力範囲 : $-4 \leq x, y \leq 4$ (FIX28) $\text{且つ} \sqrt{x^2 + y^2} \leq 4$ (FIX28) 出力 : FIX28 $\sqrt{x^2 + y^2}$ 出力範囲: $0 \leq \text{sqrt} \leq 4\sqrt{2}$ (FIX28)	入力値 x と y から $\sqrt{x^2 + y^2}$ の値の値を計算し(FIX28)の型で出力																			
R_motor_sqrt_dif_FIX28 入力 : FIX28 x :: 入力値 x FIX28 y :: 入力値 y 入力範囲 : $-4 \leq x, y \leq 4$ (FIX28) 出力 : FIX28 $\sqrt{x^2 - y^2}$ 出力範囲: $0 \leq \text{sqrt} \leq 4$ (FIX28)	入力値 x と y から $\sqrt{x^2 - y^2}$ の値の値を計算し(FIX28)の型で出力																			

表 3-26 “rm_mtr_speed.c”関数一覧(1/2)

ファイル	関数	処理概要
rm_mtr_speed.c	mtr_speed_cyclic 入力 : timer_callback_args_t *p_args :: 割り込みハンドラ引数ポインタ 出力 : なし	速度制御割り込みハンドラ 周期タイマ割り込み 周期 : 1[ms] ・ 始動制御 ・ 弱め磁束制御/最大トルク制御 ・ d 軸 q 軸電流、回転速度の指令値設定処理呼び出し ・ 速度 PI 制御処理呼び出し ・ エラー監視処理呼び出し
	mtr_lpf1_run 入力 : st_mtr_lpf1_t *p_st_lpf :: LPF 構造体ポインタ int32_t s4_input :: LPF 入力 const uint8_t u1_q :: LPF の Q 値 出力 : int32_t :: LPF 出力	1 次 LPF 処理
	mtr_ref_speed_get 入力 : st_mtr_foc_speed_t *p_st_speed :: FOC 速度制御データ構造体ポインタ 出力 : int32_t :: 速度指令値	速度制御用指令値の設定
	mtr_pi_run_asr 入力 : st_mtr_pi_t *p_st_pi :: PI 制御構造体ポインタ int32_t s4_err :: 偏差 const uint8_t u1_kp_q :: 比例ゲイン Q 値 const uint8_t u1_q_err_shift :: エラー番号のシフト値 const uint8_t u1_q_kidt_shift :: 積分ゲイン Q 値 const uint8_t u1_q_intg_shift :: 整数部のシフト値 出力 : int32_t :: PI 出力	速度 PI 制御処理
	mtr_speed_drive_ctrl 入力 : st_mtr_common_foc_t *p_st_foc_common :: センサレス FOC メインデータ構造体ポインタ 出力 : int8_t :: 制御ステータス	モータ速度駆動制御
	mtr_speed_process_opl_ctrl 入力 : st_mtr_common_foc_t *p_st_foc_common :: センサレス FOC メインデータ構造体ポインタ uint8_t u1_state_ctrl :: 制御ステータス int32_t s4_ref_temp_idq_ctrl[2] :: 電流成分の基準値 int32_t *p_s4_ref_temp_speed_ctrl :: 基準速度制御値 出力 : int8_t :: 制御ステータス	オープンループ制御モードのモータ速度制御処理
	mtr_speed_process_cli_ctrl 入力 : st_mtr_common_foc_t *p_st_foc_common :: センサレス FOC メインデータ構造体ポインタ uint8_t u1_state_ctrl :: 制御ステータス int32_t s4_ref_temp_idq_ctrl[2] :: 電流成分の基準値 int32_t *p_s4_ref_temp_speed_ctrl :: 基準速度制御値 出力 : int8_t :: 制御ステータス	クローズドループ制御モードのモータ速度制御処理

表 3-27 “rm_mtr_speed.c”関数一覧(2/2)

ファイル	関数	処理概要
rm_mtr_speed.c	mtr_speed_state_ctrl_get 入力：なし 出力：uint8_t :: ドライブ制御ステータス	ドライブ制御ステータスの取得
	mtr_ref_iq_get 入力：st_mtr_speed_t *p_st_speed :: FOC 構造体ポインタ st_mtr_foc_carrier_t *p_st_carrier :: 速度制御構造体ポインタ 出力：int32_t :: q 軸電流指令値	q 軸電流指令値の設定
	mtr_ref_id_get 入力：st_mtr_foc_speed_t *p_st_speed :: FOC 速度制御データ構造体ポインタ st_mtr_foc_carrier_t *p_st_carrier :: 速度制御構造体ポインタ 出力：int32_t :: d 軸電流指令値	d 軸電流指令値の設定
	mtr_offset_calc_init 入力：なし 出力：なし	電流検出のオフセット演算を初期化
	mtr_offset_calc_status_get 入力：なし 出力：uint8_t :: 演算処理ステータス	電流検出のオフセット演算ステータス取得
	mtr_error_check 入力：st_mtr_foc_carrier_t *p_st_carrier :: 速度制御構造体ポインタ 出力：なし	エラー処理 ・ 過電圧検出 ・ 低電圧検出 ・ 過速度検出
	mtr_abs 【inline function】 入力：int32_t s4_value :: 入力値 出力：int32_t :: 出力値	入力の絶対値を出力
	mtr_limit_abs 【inline function】 入力：int32_t s4_value :: 入力値 int32_t s4_limit_value :: 制限値 出力：int32_t :: 出力値	入力を絶対値で制限

表 3-28 “rm_mtr_ol2cl_ctrl.o”関数一覧

ファイル	関数	処理概要
rm_mtr_ol2cl_ctrl.o	RM_MTR_OplOL2CLlqCalc 入力: st_mtr_ol2cl_t *st_ol2cl :: センサレス切り替え制御構造体ポインタ int32_t s4_ref_id :: d 軸電流指令値 int8_t s1_dir :: 回転方向 出力: なし	センサレス切り替え制御時のトルク電流算出
	RM_MTR_OplOL2CLRefIqGet 入力: st_mtr_ol2cl_t *st_ol2cl :: センサレス切り替え制御構造体ポインタ int32_t s4_ref_id :: d 軸電流指令値 出力: int32_t :: q 軸電流指令値	センサレス切り替え制御時の q 軸電流指令値の算出
	RM_MTR_OpenloopDampCtrl 入力: st_mtr_damp_t *st_damp :: 軸誤差推定構造体ポインタ int32_t s4_ed :: d 軸誘起電圧 int32_t s4_ref_speed :: 指令速度 出力: int32_t :: ダンピング指令速度	ダンピング制御処理
	mtr_hpf1_run 【inline function】 入力: st_mtr_hpf1_t *st_hpf :: HPF 構造体ポインタ int32_t s4_input :: HPF 入力 const uint8_t u1_q :: HPF の Q 値 出力: int32_t :: HPF 出力	1 次 HPF 処理
	mtr_limit_abs 【inline function】 入力 : int32_t s4_value :: 入力値 int32_t s4_limit_value :: 制限値 出力 : int32_t :: 出力値	入力を絶対値で制限
	mtr_abs 【inline function】 入力 : int32_t s4_value :: 入力値 出力 : int32_t :: 出力値	入力の絶対値を出力

3.3 センサレスベクトル制御ソフト変数一覧

本制御プログラムで使用する変数一覧を次に示します。ただし、ローカル変数は記載していません。
また、本制御プログラムの制御値は、各値をスケーリングした上で算出しています。Q 表記が適用される変数について、スケール欄にある Qn は小数部分が n ビットであることを表します。ただし、一部の変数と構造体メンバの Q 表記に関しては r_mtr_scaling_parameter.h にある定義で計算されるので、この場合はスケール欄にデフォルトの Q 表記を記載しています。PU 単位が適用される変数・構造体メンバの単位は [PU([元単位])] のように記述します。

表 3-29 “mtr_main.c” 変数一覧

変数	型	Qn	PU	内容	備考
g_u1_system_mode	static uint8_t	Q0	-	モードシステム管理用変数	
g_u1_motor_status	static uint8_t	Q0	-	モータステータス管理用変数	
g_u2_error_status	static uint16_t	Q0	-	エラーステータス管理用変数	
g_u2_conf_hw	uint16_t	Q0	-	RMW configuration 用変数	
g_u2_conf_sw	uint16_t	Q0	-		
g_u2_conf_tool	uint16_t	Q0	-		
gui_u1_active_gui	uint8_t	Q0	-		
g_u2_conf_sw_ver	uint16_t	Q0	-		
com_u1_run_event	uint8_t	Q0	-	ランモード遷移用変数	0 : MTR_EVENT_STOP 1 : MTR_EVENT_DRIVE 2 : MTR_EVENT_ERROR 3 : MTR_EVENT_RESET
g_u1_run_event	uint8_t	Q0	-		

表 3-30 “r_mtr_ics.c” 変数一覧

変数	型	Qn	PU	内容	備考
st_ics	st_mtr_ctrl_input_t	-	-	ICS 変数受け渡し構造体	構造体
g_st_com	st_com_t	-	-	COM 変数構造体	構造体
g_st_com_init	st_com_t	-	-	COM 変数構造体の初期値	構造体 const
g_st_design_param_macro	st_mtr_design_parameter_macro_t	-	-	制御パラメータ構造体の初期値	構造体 const
g_s2_enable_write	int16_t	Q0	-	変数書き換え許可	
s_u1_cnt_ics	static uint8_t	Q0	-	通信処理周期間引き変数	

表 3-31 “rm_mtr_pin_data.c” 変数一覧

変数	型	Qn	PU	内容	備考
g_user_pin_cfg_data[]	ioport_pin_cfg_t	-	-	端子コンフィグ情報	
g_user_pin_ctrl	ioport_instance_ctrl_t	-	-	端子情報	
g_user_pin_cfg	ioport_cfg_t	-	-	端子コンフィグ情報	

表 3-32 “rm_mtr_driver.c” 変数一覧

変数	型	Qn	PU	内容	備考
g_st_ctrl_params	st_mtr_design_parameter_t	-	-	コントロールパラメータ	
g_st_gain_buf	st_mtr_ctrl_gain_t	-	-	入力ゲインデータ	
g_u1_trig_enable_write	uint8_t	Q0	-	受け渡し完了フラグ	
g_u1_stop_req	uint8_t	Q0	-	モータ停止フラグ	
g_s2_cnt	int16_t	Q0	-	カウンタ	
s_state_transition_table [MTR_SIZE_EVENT] [MTR_SIZE_STATE]	static uint8_t	Q0	-	状態遷移用のマクロ配列	
s_action_table [MTR_SIZE_EVENT] [MTR_SIZE_STATE]	static mtr_action_t	Q0	-	状態遷移用の関数配列	

表 3-33 “rm_mtr_carrier.c” 変数一覧

変数	型	Qn	PU	内容	備考
g_st_foc_common	st_mtr_common_foc_t	-	-	電流制御構造体	構造体

3.4 センサレスベクトル制御ソフト構造体一覧

本制御プログラムで使用する構造体一覧を次に示します。使用されていない構造体は省略します。

表 3-34 “rm_mtr_parameter.h / 構造体 : ”st_mtr_parameter_t” 変数一覧

変数	型	Qn	PU	内容	備考
u2_mtr_pp	uint16_t	Q0	-	極対数	
s4_mtr_r	int32_t	Q31	抵抗(電圧/電流)	抵抗 [PU]	
s4_mtr_ld	int32_t	Q32	インダクタンス(抵抗/角周波数)	d 軸インダクタンス [PU]	
s4_mtr_lq	int32_t	Q32	インダクタンス(抵抗/角周波数)	q 軸インダクタンス [PU]	
s4_mtr_ke	int32_t	Q28	誘起電圧定数(電圧/角周波数)	誘起電圧定数 [PU]	
s4_mtr_j	int32_t	Q25	イナーシャ (誘起電圧定数×電流 ×(極対数/角周波数) ²)	イナーシャ [PU]	

表 3-35 “rm_mtr_ctrl_gain.h / 構造体 : ”st_mtr_design_parameter_t” 変数一覧

変数	型	Qn	PU	内容	備考
st_macro	st_mtr_design_parameter_macro_t	-	-	制御パラメータ構造体	
st_com	st_mtr_design_parameter_com_t	-	-	COM 変数構造体	
f4_i_repro_cof_hz	float	-	-	電流再現用 LPF カットオフ周波数 [Hz]	
f4_r	float	-	-	抵抗 [Ω]	
f4_ld	float	-	-	d 軸インダクタンス [H/rad]	
f4_lq	float	-	-	q 軸インダクタンス [H/rad]	
f4_ke	float	-	-	誘起電圧定数 [V・s/rad]	
f4_j	float	-	-	イナーシャ [kgm ² /rad ²]	
f4_ol_ref_id	float	-	-	オープンループ指令 Id [A]	
f4_ol2cl_speed	float	-	-	オープンループからの切り替え速度[rpm]	
f4_ramp_limit_speed	float	-	-	加速度制限値 [rpm/ms]	
f4_do_lpf_cof_hz	float	-	-	外乱抑制 LPF カットオフ周波数[Hz]	
f4_do_hpf_cof_hz	float	-	-	外乱抑制 HPF カットオフ周波数[Hz]	
f4_d0_div_pm	float	-	-	静止摩擦係数/(極対数×誘起電圧定数) [kg・m ² /(rad・s ²)]	
f4_d1_div_p2m	float	-	-	動摩擦係数/(極対数 ² ×誘起電圧定数) [kg・m ² /(rad ² ・s)]	
f4_damp_hpf_cof_hz	float	-	-	ダンピング制御 HPF カットオフ周波数[Hz]	
f4_damp_zeta	float	-	-	ダンピング制御減衰係数	
f4_pherr_lpf_cof_hz	float	-	-	切り替え制御のための位相誤差の LPF カットオフ周波数[Hz]	
f4_ol2cl_switch_time	float	-	-	切り替え時間[s]	
f4_fw_pi_nf_hz	float	-	-	弱め磁束制御 PI コントローラ固有周波数[Hz]	
f4_fw_speed_err_lpf_cof_hz	float	-	-	速度偏差 LPF カットオフ周波数 [Hz]	
u2_mtr_pp	uint16_t	Q0	-	極対数	

表 3-36 “rm_mtr_ctrl_gain.h / 構造体 : ”st_mtr_design_parameter_macro_t” 変数一覧

変数	型	Qn	PU	内容	備考
f4_dt	float	-	-	電流制御周期	
f4_dt_speed	float	-	-	速度制御周期	
f4_pu_sf_afreq	float	-	-	周波数のスケールファクタ	
f4_i_lim_vd	float	-	-	vd の現在の PI 積分項の制限	
f4_i_lim_vq	float	-	-	vq の現在の PI 積分項の制限	
f4_speed_lim_rad	float	-	-	制限速度超過 [rad/s]	
u1_q_current	uint8_t	Q0	-	電流の Q 値	
u1_q_acr_kp	uint8_t	Q0	-	電流 PI 比例ゲインの Q 値	
u1_q_acr_kidt	uint8_t	Q0	-	電流 PI 積分ゲイン * 制御周期の Q 値	
u1_q_acr_deadband	uint8_t	Q0	-	電流 PI デッドバンドの Q 値	
u1_q_asr_kp	uint8_t	Q0	-	速度 PI 比例ゲインの Q 値	
u1_q_asr_kidt	uint8_t	Q0	-	速度 PI 積分ゲイン * 制御周期の Q 値	
u1_q_asr_deadband	uint8_t	Q0	-	速度 PI デッドバンドの Q 値	
u1_q_pll_kp	uint8_t	Q0	-	PLL 比例ゲインの Q 値	
u1_q_pll_kidt	uint8_t	Q0	-	PLL 積分ゲイン * 制御周期の Q 値	
u1_q_pll_deadband	uint8_t	Q0	-	PLL デッドバンドの Q 値	
u1_q_acr_lpf_k	uint8_t	Q0	-	電流 LPF ゲインの Q 値	
u1_q_asr_lpf_k	uint8_t	Q0	-	速度 LPF ゲインの Q 値	
u1_q_i_repro_lpf_k	uint8_t	Q0	-	電流再現用 LPF ゲインの Q 値	
u1_q_do_lpf_k	uint8_t	Q0	-	外乱抑制 LPF の Q 値	
u1_q_do_hpf_k	uint8_t	Q0	-	外乱抑制 HPF の Q 値	
u1_q_j_div_tc_p2m	uint8_t	Q0	-	(イナーシャ) / (LPF の時定数 × 極対数 ² × 誘起電圧定数)	
u1_q_damp_k	uint8_t	Q0	-	ダンピング制御ゲインの Q 値	
u1_q_damp_hpf_k	uint8_t	Q0	-	ダンピング制御 HPF の Q 値	
u1_q_pherr_lpf_k	uint8_t	Q0	-	位相誤差 LPF ゲインの Q 値	
u1_q_ol2cl_current_k1	uint8_t	Q0	-	センサレス切り替え処理ゲインの Q 値	
u1_q_coef_mtpa_current	uint8_t	Q0	-	(LD-LQ)/KE 計算の Q 値	
u1_q_fw_lpf_k	uint8_t	Q0	-	弱め磁束制御用 LPF の Q 値	
u1_q_fw_kp	uint8_t	Q0	-	弱め磁束制御 PI 比例ゲインの Q 値	
u1_q_fw_kidt	uint8_t	Q0	-	弱め磁束制御 PI ki * dt の Q 値	
u1_q_d1_div_p2m	uint8_t	Q0	-	1 次摩擦係数の Q 値	
u1_q_pi_run_acr_err	uint8_t	Q0	-	ACR PI エラーの Q 値	
u1_q_pi_run_asr_err	uint8_t	Q0	-	ASR PI エラーの Q 値	
u1_q_pi_run_pll_err	uint8_t	Q0	-	PLL PI エラーの Q 値	
u1_q_afreq	uint8_t	Q0	-	角周波数の Q 値	

表 3-37 “rm_mtr_ctrl_gain.h / 構造体 : ”st_mtr_design_parameter_com_t” 変数一覧

変数	型	Qn	PU	内容	備考
f4_acr_nf_hz	float	-	-	電流 PI 制御固有周波数 [Hz]	
f4_acr_lpf_cof_hz	float	-	-	電流 LPF カットオフ周波数 [Hz]	
f4_acr_deadband_lsb	float	-	-	電流 PI 制御デッドバンド [LSB]	
f4_asr_nf_hz	float	-	-	速度 PI 制御固有周波数 [Hz]	
f4_asr_lpf_cof_hz	float	-	-	速度 LPF カットオフ周波数 [Hz]	
f4_asr_deadband_lsb	float	-	-	速度 PI 制御デッドバンド [LSB]	
f4_asr_ki_aug	float	-	-	速度 PI 制御積分器増幅率	
f4_pll_nf_hz	float	-	-	PLL 固有周波数 [Hz]	
f4_pll_deadband_lsb	float	-	-	PLL デッドバンド [LSB]	

表 3-38 “rm_mtr_ctrl_gain.h / 構造体 : ”st_mtr_ctrl_gain_t” 変数一覧

変数	型	Qn	PU	内容	備考
st_pi_acr_id	st_mtr_pi_t	-	-	d 軸電流制御構造体	
st_pi_acr_iq	st_mtr_pi_t	-	-	q 軸電流制御構造体	
st_pi_asr	st_mtr_pi_t	-	-	速度制御構造体	
s4_k_lpf_in_asr	int32_t	Q31	-	速度 LPF 入力係数	
s4_k_lpf_in_acr	int32_t	Q31	-	電流 LPF 入力係数	
s4_k_do_j_div_tc_p2m	int32_t	Q31	-	(イナーシャ) / (LPF の時定数 × 極対数 ² × 誘起電圧定数)	
s4_k_lpf_in_do	int32_t	Q31	-	外乱抑制 LPF 入力係数	
s4_k_hpf_in_do	int32_t	Q31	-	外乱抑制 HPF 係数	
s4_coef_d0_div_pm	int32_t	Q28	-	静止摩擦係数/(極対数×誘起電圧定数)	
s4_coef_d1_div_p2m	int32_t	Q37	-	動摩擦係数/(極対数 ² ×誘起電圧定数)	
s4_k_lpf_in_i_repro	int32_t	Q31	-	電流再現用 LPF 入力係数	
st_pi_pll	st_mtr_pi_t	-	-	PLL 制御構造体	
s4_k_damp	int32_t	Q28	速度/電圧	ダンピング制御ゲイン	
s4_k_hpf_in_damp	int32_t	Q31	-	ダンピング制御 HPF 係数	
s4_k0_ol2cl_theta2current	int32_t	Q28	-	角度から電流への変換係数の 0 次成分	
s4_k1_ol2cl_theta2current	int32_t	Q42	-	角度から電流への変換係数の 1 次成分	
s4_k_lpf_in_pherr	int32_t	Q31	-	位相誤差 LPF 入力係数	
s4_coef_mtpa_current	int32_t	-	-	MTPA 電流係数 (Id - Iq) / ke	
s4_kp_fw	int32_t	-	-	FW PI 比例ゲイン	
s4_kidt_fw	int32_t	-	-	FW PI ki * dt	
s4_k_lpf_in_fw	int32_t	-	-	FW LPF 入力係数	
s4_lim_asr_err	int32_t	Q28	角周波数	速度 PI 制御最大誤差制限 [PU(rad/s)]	

表 3-39 “RM_DSP_RA2_GCC_32BIT.h / 構造体 : ” st_sincos28” 変数一覧

変数	型	Qn	PU	内容	備考
sin	int32_t	Q28	-	正弦 (FIX28)	
cos	int32_t	Q28	-	余弦 (FIX28)	
angle_rad	int32_t	Q28	1	角度 [rad] (FIX28)	

表 3-40 “RM_DSP_RA2_GCC_32BIT.h / 構造体 : ” st_coordinate28” 変数一覧

変数	型	Qn	PU	内容	備考
ph_u	int32_t	Q28	-	U 相 (FIX28)	
ph_v	int32_t	Q28	-	V 相 (FIX28)	
ph_w	int32_t	Q28	-	W 相 (FIX28)	
ax_d	int32_t	Q28	-	d 軸 (FIX28)	
ax_q	int32_t	Q28	-	q 軸 (FIX28)	
st_angle	st_sincos28	-	-	角度構造体	

(FIX n) は固定少数による変数表現で以下の様に定義しています。

1 bit 符号	31-n bit 整数部	n bit 小数部
-------------	-----------------	--------------

表 3-41 “rm_mtr_driver.h / 構造体 : ”st_mtr_ctrl_input_t” 変数一覧

変数	型	Qn	PU	内容	備考
s1_ref_direction	int8_t	Q0	-	回転方向指令値	
u2_cnt_offset_calc	uint16_t	Q0	-	オフセット検出時間	
s4_ref_speed	int32_t	Q28	角周波数	指令速度 [PU]	電気角
s4_lim_accel	int32_t	Q28	角周波数	加速度制限値 [PU]	電気角
s4_max_speed	int32_t	Q28	角周波数	最大速度 [PU]	電気角
s4_th_speed_cl2ol	int32_t	Q28	角周波数	センサレスからオープンループへの切り替え速度[PU]	電気角
s4_th_speed_ol2cl	int32_t	Q28	角周波数	オープンループからセンサレスへの切り替え速度[PU]	電気角
s4_ref_id_ol	int32_t	Q28	電流	オープンループ d 軸指令電流[PU]	
s4_cnt_wait_draw_in	int32_t	Q0	-	引き込み時間のカウンタ回数	
s4_coef_init_intg	int32_t	Q28	電流	センサレス移行時の ASR の積分項初期値	
s4_lim_current	int32_t	Q13	電流	電流の上昇制限値[PU/ms]	
s4_mod_3ph2ph_speed_rad	int32_t	Q0	-	三相変調から二相変調への切り替え速度	【1 シャントモード・二相変調時】
s4_mod_2ph3ph_speed_rad	int32_t	Q0	-	二相変調から三相変調への切り替え速度	
s4_ad_point_a_adj_cnt_3ph	int32_t	Q0	-	A/D 変換タイミング A の調整値 (三相変調)	
s4_ad_point_b_adj_cnt_3ph	int32_t	Q0	-	A/D 変換タイミング B の調整値 (三相変調)	
s4_ad_point_a_adj_cnt_2ph	int32_t	Q0	-	A/D 変換タイミング A の調整値 (二相変調)	
s4_ad_point_b_adj_cnt_2ph	int32_t	Q0	-	A/D 変換タイミング B の調整値 (二相変調)	
s2_ad_point_a_adj_cnt	int16_t	Q0	-	A/D 変換タイミング A の調整値	【1 シャントモード時】
s2_ad_point_b_adj_cnt	int16_t	Q0	-	A/D 変換タイミング B の調整値	
s2_mod_2ph_bot_change_cnt	int16_t	Q0	-	二相変調遷移カウンタ	【二相変調時】
s4_speed_limit_rate	int32_t	Q0	-	ダンピング制御の速度制限値	【オープンループダンピング制御】
st_motor	st_mtr_parameter_t	-	-	モータパラメータ構造体	構造体

表 3-42 “rm_mtr_statemachine.h / 構造体 : ”st_mtr_statemachine_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_status	uint8_t	Q0	-	モータステータス	
u1_status_next	uint8_t	Q0	-	遷移後モータステータス	
u2_status_error	uint16_t	Q0	-	エラーステータス	
u1_crnt_event	uint8_t	Q0	-	実行イベント	

表 3-43 “rm_mtr_est_phase_err.h / 構造体 : ”st_mtr_estimate_phe_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_est_ed	int32_t	Q28	電圧	d 軸誘起電圧	
s4_est_eq	int32_t	Q28	電圧	q 軸誘起電圧	
s4_est_e	int32_t	Q28	電圧	誘起電圧 $\sqrt{(ed^2+eq^2)}$	
s4_coef_reci_ke	int32_t	Q28	1/誘起電圧定数	誘起電圧定数の逆数	

表 3-44 “rm_mtr_est_phase_err.h / 構造体 : ” st_mtr_est_vect_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_ref_vd	int32_t	Q28	電圧	vd	
s4_ref_vq	int32_t	Q28	電圧	vq	
s4_est_r_id	int32_t	Q28	電圧	R×id	
s4_est_r_iq	int32_t	Q28	電圧	R×iq	
s4_est_speed_ld_id	int32_t	Q28	電圧	速度×Ld×id	
s4_est_speed_lq_iq	int32_t	Q28	電圧	速度×Lq×iq	

表 3-45 “rm_mtr_common.h / 構造体 : ” st_mtr_lpf1_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_in_k	int32_t	電流 : Q31 速度 : Q31 位相 : Q31	-	LPF 入力ゲイン	
s4_out_k	int32_t	電流 : Q31 速度 : Q31 位相 : Q31	-	LPF 前回出力ゲイン	
s4_pre_out	int32_t	電流 : Q28 速度 : Q28 位相 : Q28	電流 : 電流 速度 : 角周波数 位相 : 角度	前回出力値	

表 3-46 “rm_mtr_common.h / 構造体 : ” st_mtr_hpf1_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_k	int32_t	Q31	-	HPF 入力ゲイン	
s4_pre_in	int32_t	Q28	-	前回入力値	
s4_pre_out	int32_t	Q28	-	前回出力値	
u1_q_k	uint8_t	Q0	-	HPF 入力ゲインの Q 値	

表 3-47 “rm_mtr_foc_less_speed.h / 構造体 : ” st_mtr_pi_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_kp	int32_t	速度 : Q29 電流 : Q32 PLL : Q33	電流 : 抵抗 速度 : 電流/角周波数 PLL : 角周波数/角度	比例ゲイン	
s4_kidt	int32_t	速度 : Q38 【1 シャントモード時】 電流 : Q36 PLL : Q40 【3 シャントモード時】 電流 : Q37 PLL : Q41	電流 : 抵抗 速度 : 電流/角周波数 PLL : 角周波数/角度	積分ゲイン×制御周期	
s4_max_intg	int32_t	速度 : Q22 【1 シャントモード時】 電流 : Q26 PLL : Q20 【3 シャントモード時】 電流 : Q25 PLL : Q19	電流 : 電流 速度 : 角周波数 PLL : 角度	積分上限	
s4_min_intg	int32_t	速度 : Q22 【1 シャントモード時】 電流 : Q26 PLL : Q20 【3 シャントモード時】 電流 : Q25 PLL : Q19	電流 : 電流 速度 : 角周波数 PLL : 角度	積分下限	
s4_decimal	int32_t	速度 : Q22 【1 シャントモード時】 電流 : Q26 PLL : Q20 【3 シャントモード時】 電流 : Q25 PLL : Q19	電流 : 電流 速度 : 角周波数 PLL : 角度	小数部の格納値	

表 3-48 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_acr_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_flg_err_zero	uint8_t	Q0	-	電流偏差 0 フラグ 0:2 相近接補償なし 1:2 相近接補償あり	【1 シャントモード時】 2 相近接補償時は電流誤差をゼロにリセット
s4_ref_id	int32_t	Q28	電流	d 軸電流指令	
s4_ref_iq	int32_t	Q28	電流	q 軸電流指令	
s4_ref_id_ol	int32_t	Q28	電流	オープンループ d 軸電流指令値	
s4_lim_ramp_current	int32_t	Q28	電流	電流の上昇制限値[PU/ms]	
s4_ref_vd_pre	int32_t	Q28	電圧	d 軸出力電圧指令前回値	
s4_ref_vq_pre	int32_t	Q28	電圧	q 軸出力電圧指令前回値	
s4_ref_vd_buffer	int32_t	Q28	電圧	d 軸電圧のバッファ値	
s4_ref_vq_buffer	int32_t	Q28	電圧	q 軸電圧のバッファ値	
st_ad_iq_lpf	st_mtr_lpf1_t	-	-	q 軸電流 LPF 構造体	構造体
st_ad_id_lpf	st_mtr_lpf1_t	-	-	d 軸電流 LPF 構造体	
st_pi_id	st_mtr_pi_t	-	-	d 軸電流 PI 構造体	
st_pi_iq	st_mtr_pi_t	-	-	q 軸電流 PI 構造体	

表 3-49 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_pll_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_est_speed	int32_t	Q28	角周波数	速度検出値	
st_pi	st_mtr_pi_t	-	-	PI 構造体	構造体

表 3-50 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_s4_comp_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_comp_vu	int32_t	Q28	電圧	U 相電圧補償値	
s4_comp_vv	int32_t	Q28	電圧	V 相電圧補償値	
s4_comp_vw	int32_t	Q28	電圧	W 相電圧補償値	

表 3-51 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_asr_t” 変数一覧

変数	型	Qn	PU	内容	備考
s1_ref_dir	int8_t	-	-	回転方向指令	1 : CW -1 : CCW
s4_ref_speed	int32_t	Q28	角周波数	指令速度	
s4_ref_speed_ctrl	int32_t	Q28	角周波数	指令速度制御値	
s4_speed	int32_t	Q28	角周波数	速度検出推定値	
s4_lim_ramp_speed	int32_t	Q28	角周波数	加速度制限値	
s4_cnt_ramp_deci_sample	int32_t	Q0	角周波数	加速度制限値の間引き回数	
s4_max_speed	int32_t	Q28	角周波数	最大速度	
s4_init_intg	int32_t	Q28	電流	センサレス切り替え時積分項初期値	
s4_th_cl2ol_speed	int32_t	Q28	角周波数	センサレスからオープンループへの切り替え速度	
s4_th_ol2cl_speed	int32_t	Q28	角周波数	オープンループからセンサレスへの切り替え速度	
s4_coef_d0_div_pm	int32_t	Q28	-	静止摩擦係数/(極対数×誘起電圧定数)	
s4_coef_d1_div_p2m	int32_t	Q37	-	動摩擦係数/(極対数^2×誘起電圧定数)	
u2_cnt_cl2ol_judge_wait	uint16_t	Q0	-	センサレスからオープンループを判断するためのカウンタ	
st_pi	st_mtr_pi_t	-	-	速度 PI 構造体	構造体
s4_lim_asr_err	int32_t	-	-	速度 PI 制御最大誤差制限	
st_lpf	st_mtr_lpf1_t	-	-	速度 LPF 構造体	構造体
s4_ref_speed_damp_ctrl	int32_t	Q28	角周波数	オープンループダンピング制御の基準速度	

表 3-52 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_mod_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_flg_uvw_min_pattern	uint8_t	-	-	相電流最小パターン	
u1_flg_uvw_min_pattern_pre	uint8_t	-	-	相電流最小パターン前回値	
s2_cnt_2ph_bot_change	int16_t	-	-	二相変調遷移カウンタ	[二相変調]
s4_v_diff_sum	int32_t	-	-	中間から最小の 2 相電圧差の積分	
s4_mod_u	int32_t	Q28	-	U 相変調率	
s4_mod_v	int32_t	Q28	-	V 相変調率	
s4_mod_w	int32_t	Q28	-	W 相変調率	
s4_ad_reci_vdc	int32_t	Q28	1/電圧	電圧の逆数	
s4_lim_vout	int32_t	Q28	電圧	電圧リミット	

表 3-53 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_tscs_t” 変数一覧[3 シャントモード時]

変数	型	Qn	PU	内容	備考
u4_offset_idc_adc	uint32_t	Q0	-	DC リンク電流オフセット値	
u4_offset_calc_cnt	uint32_t	Q0	-	オフセット電流測定回数	
u4_offset_sample_cnt	uint32_t	Q0	-	オフセット電流測定サンプル数	
s4_offset_iu	int32_t	Q0	-	U 相電流オフセット値	
s4_offset_iv	int32_t	Q0	-	V 相電流オフセット値	
s4_offset_iw	int32_t	Q0	-	W 相電流オフセット値	
s2_duty_u	int16_t	Q0	-	U 相 Duty(PWM レジスタ設定値)	
s2_duty_v	int16_t	Q0	-	V 相 Duty(PWM レジスタ設定値)	
s2_duty_w	int16_t	Q0	-	W 相 Duty(PWM レジスタ設定値)	
u2_ad_current_uph	uint16_t	Q0	-	U 相電流 AD 変換値	
u2_ad_current_vph	uint16_t	Q0	-	V 相電流 AD 変換値	
u2_ad_current_wph	uint16_t	Q0	-	W 相電流 AD 変換値	

表 3-54 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_sscs_t” 変数一覧[1 シャントモード時] (1/2)

変数	型	Qn	PU	内容	備考
u4_offset_calc_cnt	uint32_t	Q0	-	オフセット電流測定回数	
u4_offset_sample_cnt	uint32_t	Q0	-	オフセット電流測定サンプル数	
u1_drv_pattern_get	uint8_t	Q0	-	変調率大小関係パターン参照	
u1_drv_pattern_set	uint8_t	Q0	-	変調率大小関係パターン設定	
u1_drv_pattern_put	uint8_t	Q0	-	変調率大小関係パターン出力	
u1_flag_mod_2ph	uint8_t	Q0	-	二相変調フラグ	
s2_duty_max	int16_t	Q0	-	最大デューティ値	
s2_duty_mid	int16_t	Q0	-	中間デューティ値	
s2_duty_min	int16_t	Q0	-	最小デューティ値	
s2_duty_u	int16_t	Q0	-	U 相デューティ値	
s2_duty_v	int16_t	Q0	-	V 相デューティ値	
s2_duty_w	int16_t	Q0	-	W 相デューティ値	
s2_duty_max_adj	int16_t	Q0	-	最大デューティ調整値	
s2_duty_mid_adj	int16_t	Q0	-	中間デューティ調整値	
s2_duty_min_adj	int16_t	Q0	-	最小デューティ調整値	
s2_duty_max_adj_comp	int16_t	Q0	-	最大デューティ調整補償値	
s2_duty_mid_adj_comp	int16_t	Q0	-	中間デューティ調整補償値	
s2_duty_min_adj_comp	int16_t	Q0	-	最小デューティ調整補償値	
s2_duty_u_adj	int16_t	Q0	-	U 相デューティ調整値	
s2_duty_v_adj	int16_t	Q0	-	V 相デューティ調整値	
s2_duty_w_adj	int16_t	Q0	-	W 相デューティ調整値	

表 3-55 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_sscs_t” 変数一覧[1 シャントモード時] (2/2)

変数	型	Qn	PU	内容	備考
s2_duty_u_adj	int16_t	Q0	-	U相デューティ調整補償値	
s2_duty_v_adj	int16_t	Q0	-	V相デューティ調整補償値	
s2_duty_w_adj	int16_t	Q0	-	W相デューティ調整補償値	
s2_ad_point_a_cnt	int16_t	Q0	-	A/D変換ポイントAタイマカウント値	
s2_ad_point_b_cnt	int16_t	Q0	-	A/D変換ポイントBタイマカウント値	
s4_mod_3ph2ph_speed_rad	int32_t	Q0	-	三相変調から二相変調への切り替え速度	[二相変調]
s4_mod_2ph3ph_speed_rad	int32_t	Q0	-	二相変調から三相変調への切り替え速度	
s2_ad_point_a_adj_cnt_2ph	int16_t	Q0	-	A/D変換ポイントAカウント調整値 (二相変調)	
s2_ad_point_b_adj_cnt_2ph	int16_t	Q0	-	A/D変換ポイントBカウント調整値 (二相変調)	
s2_ad_point_a_adj_cnt_3ph	int16_t	Q0	-	A/D変換ポイントAカウント調整値 (三相変調)	
s2_ad_point_b_adj_cnt_3ph	int16_t	Q0	-	A/D変換ポイントBカウント調整値 (三相変調)	
s2_ad_point_a_adj_cnt	int16_t	Q0	-	A/D変換ポイントAカウント調整値	
s2_ad_point_b_adj_cnt	int16_t	Q0	-	A/D変換ポイントBカウント調整値	
s2_ad_ss_a	int16_t	Q0	-	ポイントA A/D変換結果	
s2_ad_ss_b	int16_t	Q0	-	ポイントA A/D変換結果	
s4_offset_ia	int32_t	Q0	-	ポイントA電流オフセット値	
s4_offset_ib	int32_t	Q0	-	ポイントB電流オフセット値	
u2_crnt_ad_a	uint16_t	Q0	-	ポイントA A/Dデータ	
u2_crnt_ad_b	uint16_t	Q0	-	ポイントB A/Dデータ	

表 3-56 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_common_foc_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_state_drive	uint8_t	-	-	ドライブモードステータス	0x00 : 停止中 0x01 : 駆動停止 0x81 : 駆動中 0x82 : 駆動初期化 0x83 : 駆動開始 0x84 : オフセット計測
u1_state_ctrl	uint8_t	-	-	制御ステータス	0x00 : 制御停止 0x01 : 引き込み制御 0x02 : 強制同期制御 0x03 : オープンループクローズドループ切り替え制御 0x04 : クローズドループオープンループ切り替え制御 0x05 : オープンループ速度制御 0x80 : オープンループクローズドループ切り替え遷移中 0x81 : クローズドループ速度制御
s1_direction	int8_t	-	-	現在の回転方向	1 : CW -1 : CCW
s4_ad_vdc	int32_t	Q28	電圧	電源電圧	
s4_phase_err_rad	int32_t	Q28	角度	位相誤差[rad]	
st_foc_carrier	st_foc_carrier_t	-	-	キャリア周期制御構造体	
st_foc_speed	st_foc_speed_t	-	-	速度制御構造体	

表 3-57 “rm_mtr_foc_less_speed.h / 構造体 : ” st_mtr_foc_carrier_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_flg_charge_cap	uint8_t	-	-	母線電圧チャージ完了フラグ	0 : 未到達 1 : 到達
u1_flg_offset_end	uint8_t	-	-	オフセット演算完了フラグ	
u2_error_status	uint16_t	-	-	エラーステータス	0x0000 : エラーなし 0x0001 : 過電流エラー(H/W) 0x0002 : 過電圧エラー 0x0004 : 回転速度エラー 0x0008 : ホールタイムアウトエラー 0x0010 : 誘起電圧タイムアウト エラー 0x0020 : ホールパターンエラー 0x0040 : 誘起電圧パターンエラー 0x0080 : 低電圧エラー 0x0100 : 過電流エラー(S/W) 0x0400 : チャージキャパシタ タイムアウト 0xFFFF : 未定義エラー
st_stm	st_mtr_state machine_t	-	-	ステートマシン構造体	
st_ad_i	st_coordinat e28	-	-	電流検出値構造体	
st_ref_v	st_coordinat e28	-	-	指令電圧構造体	
st_ref_i	st_coordinat e28	-	-	指令電流構造体	
st_motor	st_mtr_para meter_t	-	-	モータパラメータ構造体	
st_phe	st_mtr_estim ate_phe_t	-	-	位相誤差推定構造体	
st_tscs	st_mtr_tscs_ t	-	-	3 相電流検出構造体	[3 シャントモード時]
st_sscs	st_mtr_sscs_ t	-	-	1 シャント抵抗電流検出構造体	[1 シャントモード時]
st_acr	st_mtr_acr_t	-	-	ACR 構造体	電流 PI 制御
st_mod	st_mtr_mod_ t	-	-	変調構造体	
st_pll	st_mtr_pll_t	-	-	PLL 制御構造体	
st_i_repro	st_coordinat e28	-	-		
st_iq_repro	st_mtr_lpf1_t	-	-	電流再現用 q 軸電流 LPF	
st_id_repro	st_mtr_lpf1_t	-	-	電流再現用 d 軸電流 LPF	

表 3-58 “rm_mtr_foc_less_speed.h / 構造体 : ”st_mtr_foc_speed_t” 変数一覧

変数	型	Qn	PU	内容	備考
u1_state_ref_id	uint8_t	-	-	d 軸電流指令値生成ステータス	0 : d 軸電流 0 1 : d 軸電流直接制御 2 : 弱め磁束制御/最大トルク制御
u1_state_ref_iq	uint8_t	-	-	q 軸電流指令値生成ステータス	0 : q 軸電流 0 1 : q 軸電流直接制御 2 : センサレス切り替え制御 3 : 速度 PI 出力
u1_state_ref_speed	uint8_t	-	-	速度指令値生成ステータス	0 : 速度 0 1 : 速度変化
u2_cnt_draw_in_wait	uint16_t	Q0	-	引き込み時間のカウンタ回数	
u2_cnt_draw_in_wait_buff	uint16_t	Q0	-	引き込み時間のカウンタ回数バッファ	
u2_cnt_draw_in_time_calc	uint16_t	Q0	-	引き込み時間の回数	
st_asr	st_mtr_asr_t	-	-	ASR 構造体	速度 PI 制御
st_ol2cl	st_mtr_ol2cl_t	-	-	センサレス切り替え制御構造体	
st_phase_err_lpf	st_mtr_lpf1_t	-	-	位相誤差 LPF 構造体	
s4_damp_speed	int32_t	Q28	速度	ダンピング速度出力	

表 3-59 “rm_mtr_ol2cl_ctrl.h” / 構造体 : ”st_mtr_damp_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_k	int32_t	Q28	速度/電圧	ダンピング制御ゲイン	
s4_speed_lim_rate	int32_t	Q28	速度	速度リミット	
u1_q_damp_speed_calc	uint8_t	Q0	-	ダンピング制御速度 Q 値	
u1_q_damp_speed_limit_calc	uint8_t	Q0	-	ダンピング制御速度リミット Q 値	
st_hpf	st_mtr_hpf1_t	-	-	HPF 構造体	

表 3-60 “rm_mtr_ol2cl_ctrl.h” / 構造体 : ”st_mtr_ol2cl_t” 変数一覧

変数	型	Qn	PU	内容	備考
s4_lim_ramp_i	int32_t	Q28	電流	切り替え処理時の電流の上昇制限値[A/ms]	
s4_iq_ol2cl	int32_t	Q28	電流	切り替え処理時の q 軸電流	
s4_k0_angle2current	int32_t	Q28	-	基準電流 = $k_0 + k_1 \times (\text{位相誤差})$	
s4_k1_angle2current	int32_t	Q42	-	基準電流 = $k_0 + k_1 \times (\text{位相誤差})$	
s4_th_phase_err_rad_cl	int32_t	Q28	角度	センサレス切り替え処理時の位相誤差	
u2_cnt_switch	uint16_t	Q0	-	切り替え時間[cnt]	
u1_q_k1_angle2current	uint8_t	Q0	-	基準電流の角度誤差のための Q 値	
u1_q_idelta_calc	uint8_t	Q0	-	電流の q 軸成分算出のための Q 値	
st_ph_err	st_sincos28	-	-	位相誤差の角度構造体	
st_damp	st_mtr_damp_t	-	-	ダンピング制御の構造体	

表 3-61 “r_mtr_ics.h” / 構造体 : “st_com_t” 変数一覧

変数	型	Qn	PU	内容	備考
st_com	st_mtr_design_parameter_com_t	-	-	COM 変数構造体	
com_u1_direction	uint8_t	Q0	-	回転方向	0 : CW 1 : CCW
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				
com_f4_mtr_j	float	-	-	イナーシャ[kg m ² /rad ²]	
com_f4_mtr_d0	float	-	-	静止摩擦係数[kgm ² /(rad・s ²)]	
com_f4_mtr_d1	float	-	-	静止摩擦係数[kgm ² /(rad ² ・s)]	
com_u2_mtr_pp	uint16_t	Q0	-	極対数	
com_u2_offset_calc_cnt	uint16_t	Q0	-	電流オフセット検出時間	
com_f4_ref_speed_rpm	float	-	-	目標速度 [rpm]	機械角
com_f4_ramp_limit_speed_rpm	float	-	-	加速度制限値 [rpm/ms]	機械角
com_f4_max_speed_rpm	float	-	-	最大速度 [rpm]	機械角
com_f4_cl2ol_speed_rpm	float	-	-	センサレスからオープンループへの切り替え速度[rpm]	機械角
com_f4_ol2cl_speed_rpm	float	-	-	オープンループからセンサレスへの切り替え速度[rpm]	機械角
com_f4_ol_ref_id	float	-	-	オープンループ d 軸指令電流[A]	
com_f4_draw_in_wait_time	float	-	-	引き込み待ち時間 [sec]	
com_f4_init_asr_intg	float	-	-	センサレス移行時の ASR の積分項初期値	
com_f4_ramp_limit_current	float	-	-	電流の上昇制限値[A/ms]	
com_f4_i_repro_cof_hz	float	-	-	電流再現用 LPF カットオフ周波数 [Hz]	【1 シャントモード時】
com_s2_ad_point_a_adj_cnt	int16_t	Q0	-	A/D 変換タイミング A の調整値	
com_s2_ad_point_b_adj_cnt	int16_t	Q0	-	A/D 変換タイミング B の調整値	
com_f4_mod_3ph2ph_speed_rpm	float	-	-	三相変調から二相変調への切り替え速度 [rpm]	【1 シャントモード・二相変調時】
com_f4_mod_2ph3ph_speed_rpm	float	-	-	二相変調から三相変調への切り替え速度 [rpm]	
com_s2_ad_point_a_adj_cnt_3ph	int16_t	Q0	-	A/D 変換タイミング A の調整値 (三相変調)	
com_s2_ad_point_b_adj_cnt_3ph	int16_t	Q0	-	A/D 変換タイミング B の調整値 (三相変調)	
com_s2_ad_point_a_adj_cnt_2ph	int16_t	Q0	-	A/D 変換タイミング A の調整値 (二相変調)	
com_s2_ad_point_b_adj_cnt_2ph	int16_t	Q0	-	A/D 変換タイミング B の調整値 (二相変調)	
com_s2_mod_2ph_bot_change_cnt	int16_t	Q0	-	二相変調遷移時のカウント回数	
com_f4_damp_hpf_cof_hz	float	-	-	ダンピング制御の HPF カットオフ周波数 [Hz]	【オープンループダンピング制御】
com_f4_damp_zeta	float	-	-	ダンピング制御の減衰係数	
com_f4_damp_speed_limit_rate	float	-	-	ダンピング制御の速度制限値	
com_f4_pherr_lpf_cof_hz	float	-	-	切り替え制御のための位相誤差の LPF カットオフ周波数[Hz]	【センサレス切り替え処理】
com_f4_ol2cl_switch_time	float	-	-	切り替え時間[s]	
com_s2_enable_write	int16_t	Q0	-	変数書き換え許可	

3.5 センサレスベクトル制御ソフトマクロ定義一覧

本制御プログラムで使用するマクロ定義一覧を次に示します。

表 3-62 “r_mtr_config.h” マクロ定義一覧

マクロ	定義値	内容	備考
IP_MCK	-	インバータボードの選択	
MP_R42BLD30L3_A_a1	-	モータパラメータの選択	
CP_R42BLD30L3_A_a1	-	制御パラメータの選択	
SINGLE_SHUNT	0	1 シャントモード	
THREE_SHUNT	1	3 シャントモード	
CURRENT_SENS_METHOD	SINGLE_SHUNT THREE_SHUNT	電流検出方式 注1	
USE_DEADTIME_COMP	0 : 1	デッドタイム補償使用の選択 (1:Enable 0:Disable)	デフォルト設定 1
USE_SPEED_LPF	0 : 1	速度 LPF 使用の選択(1:Enable 0:Disable)	デフォルト設定 1
USE_CURRENT_LPF_IQ	0 : 1	q 軸電流 LPF 使用の選択(1:Enable 0:Disable)	デフォルト設定 0
USE_CURRENT_LPF_ID	0 : 1	d 軸電流 LPF 使用の選択(1:Enable 0:Disable)	デフォルト設定 0
USE_MTPA_AND_FW	0 : 1	弱め磁束制御/最大トルク制御使用の選択 (1:Enable 0:Disable)	デフォルト設定 0
USE_OPENLOOP_DAMPING	0 : 1	オープンループダンピング制御使用の選 択 (1:Enable 0:Disable)	デフォルト設定 1
USE_OL2CL_CTRL	0 : 1	センサレス切り替え制御使用の選択 (1:Enable 0:Disable)	デフォルト設定 1
USE_DISTURBANCE_SUPPRES S	0 : 1	外乱抑制使用の選択 (1:Enable 0:Disable)	デフォルト設定 0
PS_IPD_SAL	0	突極モータ向け初期位置検出	未実装
PS_IPD_NON_SAL	1	非突極モータ向け初期位置検出	未実装
PS_IPD_UNKNOWN	2	突極情報無しモータ向け初期位置検出	
PS_DRAW_IN	3	引き込み	
PS_METHOD	PS_IPD_SAL PS_IPD_NON_SAL PS_IPD_UNKNOWN PS_DRAW_IN	始動準備方式	デフォルト設定 PS_DRAW_IN
CP_CARRIER_FREQ_20KHZ	0	キャリア周波数 20kHz	
CP_CARRIER_FREQ_40KHZ	1	キャリア周波数 40kHz	
CP_CARRIER_FREQ_16KHZ	2	キャリア周波数 16kHz	
CP_CARRIER_FREQ_SEL	CP_CARRIER_FREQ_20KHZ CP_CARRIER_FREQ_40KHZ CP_CARRIER_FREQ_16KHZ	キャリア周波数選択	デフォルト設定 CP_CARRIER_ FREQ_20KHZ
MOD_3PH_SPWM	0	正弦波変調	未実装
MOD_3PH_TOW	1	3 次高調波加算	
MOD_2PH_BOT	2	二相変調	
MOD_METHOD	MOD_3PH_SPWM MOD_3PH_TOW MOD_2PH_BOT	変調方式	デフォルト設定 MOD_3PH_TO W

注 1 : 本マクロの切り替えを行う場合、config フォルダ内の電流検出に関わるパラメータを適宜設定してください。

表 3-63 “r_mtr_motor_parameter.h”マクロ定義一覧

マクロ	定義値	内容	備考
MP_POLE_PAIRS	4	極対数	R42BLD30L3_24V
MP_RESISTANCE	1.32f	抵抗 [Ω]	R42BLD30L3_24V
MP_D_INDUCTANCE	0.0021f	d 軸インダクタンス [H]	R42BLD30L3_24V
MP_Q_INDUCTANCE	0.0021f	q 軸インダクタンス [H]	R42BLD30L3_24V
MP_BEMF_CONSTANT	0.0115f	誘起電圧定数 [V・s/rad]	R42BLD30L3_24V
MP_ROTOR_INERTIA	0.0000032f	イナーシャ [kg・m ² /(rad ²)]	R42BLD30L3_24V
MP_FRICTION_0TH_ORDER	0.003583013f	静止摩擦係数 [kg・m ² /(rad・s ²)]	R42BLD30L3_24V
MP_FRICTION_1ST_ORDER	0.000001850949f	動摩擦係数 [kg・m ² /(rad ² ・s)]	R42BLD30L3_24V
MP_RATED_CURRENT	1.13f	定格電流 [A]	R42BLD30L3_24V
MP_RATED_SPEED	4000	定格速度 [rpm]	R42BLD30L3_24V
STAR	0	スター結線	デフォルト設定
DELTA	1	デルタ結線	
MP_MOTOR_WIRE_CONNECTION	STAR	モータの結線方式の選択	スター結線 or デルタ結線

表 3-64 “r_mtr_control_parameter.h”マクロ定義一覧 [1/2]

マクロ	定義値	内容	備考
CP_PWM_TIMER_FREQ	32.0f	PWM タイマ周波数 [MHz]	
CP_INTVAL_TIMER_FREQ	32.0f	インターバル・タイマ周波数 [MHz]	
CP_CARRIER_FREQ	20.0f	PWM キャリア周波数	
CP_TRX_TIMER_FREQ	32.0f	TRX タイマ周波数 [MHz]	
CP_INT_DECIMATION	1[1 シャントモード時] 0[3 シャントモード時]	割り込み間引き回数	
CP_V_PHASE_LEAD_COEF	1.0f[1 シャントモード時] 1.5f[3 シャントモード時]	電圧の位相進み係数	
CP_SPEED_CTRL_PERIOD	0.001f	速度制御周期 [秒]	
CP_AD_CONVERSION_TIME	1.1375f	A/D 変換時間[us]	
CP_AD_RINGING_WAIT_CNT	32	A/D 変換安定待ち調整値	
CP_AD_RINGING_WAIT_2PH_CNT	42	A/D 変換安定待ち調整値(二相変調)	
CP_ACR_NF_HZ	500.0f	電流 PI 制御固有周波数 [Hz]	
CP_ASR_NF_HZ	16.00000f	速度 PI 制御固有周波数 [Hz]	
CP_PLL_NF_HZ	80.00000f	PLL 制御固有周波数 [Hz]	
CP_ACR_DEADBAND_LSB	0.05000f	電流 PI のデッドバンド [LSB]	
CP_ASR_DEADBAND_LSB	0.05000f	速度 PI のデッドバンド [LSB]	
CP_PLL_DEADBAND_LSB	0.05000f	PLL 制御のデッドバンド [LSB]	
CP_ASR_LPF_COF_HZ	200.00000f	速度 LPF のカットオフ周波数 [Hz]	
CP_ACR_LPF_COF_HZ	2000.00000f	電流 LPF のカットオフ周波数 [Hz]	
CP_I_REPRO_COF_HZ	2000.00000f	電流再現用 LPF のカットオフ周波数 [Hz]	
CP_ASR_KI_AUG	8.00000f	速度 PI 制御積分器増幅率	
CP_MAX_SPEED_RPM	4000	最大速度(機械角)[rpm] 指令最大速度	
CP_SPEED_LIMIT_RPM	8000	制限速度(機械角)[rpm] 回転速度エラーリミット値	
CP_OC_LIMIT	2.26000f	過電流リミット値 [A]	
CP_OL_REF_ID	1.13000f	d 軸電流指令値 [A]	

表 3-65 “r_mtr_control_parameter.h” マクロ定義一覧 [2/2]

マクロ	定義値	内容	備考
CP_DRAW_IN_WAIT_TIME	0.20000f	引き込み待ち時間	
CP_INIT_ASR_INTEG	0.07641f	q 軸電流 PI 積分項初期値 [A]	
CP_RAMP_LIMIT_CURRENT	0.0121680f	電流の上昇制限値[PU/ms]	
CP_MOD_3PH2PH_SPEED_RPM	1600.0000000f	3 相変調から 2 相変調への切り替え速度 [rpm]	[1 シャントモード・二相変調]
CP_MOD_2PH3PH_SPEED_RPM	1400.0000000f	2 相変調から 3 相変調への切り替え速度 [rpm]	
CP_OL2CL_SPEED_RPM	800.0000000f	センサレスからオープンループへの切り替え速度(機械角)[rpm]	
CP_CL2OL_SPEED_RPM	600.0000000f	オープンループからセンサレスへの切り替え速度(機械角)[rpm]	
CP_CL2OL_JUDGE_WAIT_TIME	12.5000000f	オープンループ判定の待ち時間 [ms]	
CP_RAMP_LIMIT_SPEED_RPM	6.8223750f	加速度制限値 [rpm/ms]	
CP_RAMP_SPEED_CNT_DECIMATION	0	加速度制限の間引き回数	
CP_OFFSET_CALC_CNT	512	電流オフセット値計算回数	
CP_OFFSET_CALC_ST_WAIT_CNT	100	電流オフセット安定待ち回数	
CP_AD_POINT_A_ADJ_CNT	32	A/D 変換タイミング A の調整値	[1 シャントモード]
CP_AD_POINT_B_ADJ_CNT	32	A/D 変換タイミング B の調整値	
CP_AD_POINT_A_ADJ_2PH_CNT	42	A/D 変換タイミング A の調整値	[1 シャントモード・二相変調]
CP_AD_POINT_B_ADJ_2PH_CNT	42	A/D 変換タイミング B の調整値	
CP_MOD_2PH_BOT_CHANGE_CNT	200	二相変調遷移カウンタ	[二相変調]
CP_DAMP_HPF_COF_HZ	5.0000000f	ダンピング制御の HPF カットオフ周波数	[オープンループダンピング制御]
CP_DAMP_ZETA	1.0000000f	ダンピング制御減衰係数	
CP_DAMP_SPEED_LIMIT_RATE	0.2000000f	ダンピング制御速度リミット	
CP_PHASE_ERR_LPF_COF_HZ	10.0000000f	センサレス切り替え制御のための位相誤差 LPF カットオフ周波数[Hz]	[センサレス切り替え処理]
CP_OL2CL_SWITCH_TIME	0.1729000f	センサレス切り替え時間[s]	
CP_OL2CL_SWITCH_ANGLE_MIN	5.0000000f	オープンループ切り替え最小角度[deg]	

表 3-66 “r_mtr_inverter_parameter.h” マクロ定義一覧

マクロ	定義値	内容	備考
IP_DEADTIME	2.0f	デッドタイム	
IP_CURRENT_RANGE	16.5f	電流スケーリングレンジ [A]	
IP_VDC_RANGE	73.5f	電圧スケーリングレンジ [V]	
IP_INPUT_V	24.0f	入力電圧 [V]	
IP_CURRENT_LIMIT	21.4f	電流リミット値 [A]	
IP_OVERVOLTAGE_LIMIT	48.0f	過電圧リミット [V]	
IP_UNDERVOLTAGE_LIMIT	8.0f	低電圧リミット [V]	
IP_PHASE_SHUNT_RESISTANCE	0.010f	位相シャント抵抗[Ω]	
IP_PHASE_AMPLIFICATION_GAIN	20.000f	相電流増幅ゲイン	
IP_DC_SHUNT_RESISTANCE	0.010f	DC リンクシャント抵抗[Ω]	
IP_DC_AMPLIFICATION_GAIN	20.0f	DC リンク電流増幅ゲイン	
IP_BSC_CHARGE_CNT	150	ブートストラップコンデンサの充電時間	
IP_CHARGE_CAP_WAIT_CNT	300	DC コンデンサの充電時間	

表 3-67 “r_mtr_scaling_parameter.h” マクロ定義一覧 [1/2]

マクロ	定義値	内容	備考
FP_SF_VOLTAGE	822226302	電圧 PU 変換値 $((IP_VDC_RANGE) \times PU_SF_VOLTAGE) \times (1 < MTR_Q_VOLTAGE)$ の値を設定	
FP_SF_CURRENT	3919632764	電流 PU 変換値 $((IP_CURRENT_RANGE) \times PU_SF_CURRENT) \times (1 < MTR_Q_CURRENT)$ の値を設定	
FP_SF_CURRENT_AD	FP_SF_CURRENT >> MTR_ADC_12BIT	電流 PU 変換値 AD 変換係数補正	
PU_BASE_CURRENT_A	MP_RATED_CURRENT	電流基準値 [A]	
PU_BASE_VOLTAGE_V	IP_INPUT_V	電圧基準値 [V]	
PU_BASE_FREQ_Hz	MTR_TWOP/CP_MAX_SPEED_RPM*MP_POLE_PAIRS/60	周波数基準値 [Hz]	
PU_BASE_ANGLE_ROT	1.0f	角度基準値 [rad]	
PU_SF_CURRENT	1.0f / PU_BASE_CURRENT_A	電流スケール [1/A]	
PU_SF_VOLTAGE	1.0f / PU_BASE_VOLTAGE_V	電圧スケール [1/V]	
PU_SF_AFREQ	1.0f / PU_BASE_FREQ_Hz	角周波数スケール [s /rad]	
PU_SF_ANGLE	1.0f / PU_BASE_ANGLE_ROT	角度スケール [1/rad]	
PU_SF_TIME	PU_SF_ANGLE / PU_SF_AFREQ	時間スケール [1/s]	
PU_SF_RES	PU_SF_VOLTAGE / PU_SF_CURRENT	抵抗スケール [1/Ω]	
PU_SF_IND	PU_SF_RES / PU_SF_AFREQ	インダクタンススケール [rad/H]	
PU_SF_BEMF_CONST	PU_SF_VOLTAGE / PU_SF_AFREQ	誘起電圧定数スケール [rad/(V・s)]	
PU_SF_INERTIA	PU_SF_BEMF_CONST * PU_SF_CURRENT / (MP_POLE_PAIRS * MP_POLE_PAIRS * PU_SF_AFREQ * PU_SF_AFREQ)	イナーシャスケール [rad^2/kg・m^2]	
PU_SF_D1_DIV_P2M	(PU_SF_CURRENT / PU_SF_AFREQ)	動摩擦係数/ (極対数^2 × 誘起電圧定数) のスケール [A・s/rad]	
PU_SF_RPM_RAD	1.0f / CP_MAX_SPEED_RPM	[rpm] から [rad/s] への変換スケール	
PU_SF_RAD_RPM	CP_MAX_SPEED_RPM	[rad/s] から [rpm] への変換スケール	
PU_SF_ACR_KP	PU_SF_RES	電流 PI 比例ゲインスケール	
PU_SF_ACR_KIDT	PU_SF_RES	電流 PI 積分ゲインスケール	
PU_SF_ASR_KP	PU_SF_CURRENT / PU_SF_AFREQ	速度 PI 比例ゲインスケール	
PU_SF_ASR_KIDT	PU_SF_CURRENT / PU_SF_AFREQ	速度 PI 積分ゲインスケール	
PU_SF_PLL_KP	PU_SF_AFREQ / PU_SF_ANGLE	PLL 比例ゲインスケール	
PU_SF_PLL_KIDT	PU_SF_AFREQ / PU_SF_ANGLE	PLL 積分ゲインスケール	
MTR_Q_ANGLE	28	角度の Q 値	
MTR_Q_CURRENT	28	電流の Q 値	
MTR_Q_VOLTAGE	28	電圧の Q 値	
MTR_Q_VMOD	28	PWM 変調率の Q 値(変更不可)	
MTR_Q_AFREQ	28	角周波数の Q 値	
MTR_Q_CTRL_TIME	33[1 シャントモード] 34[3 シャントモード]	FOC 制御周期の Q 値	
MTR_Q_CTRL_TIME_SPEED	30	速度制御周期の Q 値	
MTR_Q_RESISTANCE	32	抵抗の Q 値	
MTR_Q_INDUCTANCE	33	インダクタンスの Q 値	
MTR_Q_BEMF_CONST	28	誘起電圧定数の Q 値	
MTR_Q_INERTIA	25	イナーシャの Q 値	
MTR_Q_D1_DIV_P2M	37	動摩擦係数/ (極対数^2 × 誘起電圧定数) の Q 値	
MTR_Q_RECIV	28	電圧の逆数の Q 値	
MTR_Q_RECIM	30	誘起電圧定数の逆数の Q 値	

表 3-68 “r_mtr_scaling_parameter.h” マクロ定義一覧 [2/2]

マクロ	定義値	内容	備考
MTR_Q_ACR_KP	32	電流 PI 比例ゲインの Q 値	
MTR_Q_ACR_KIDT	36[1 ショントモード時] 37[3 ショントモード時]	電流 PI 積分ゲイン×制御周期の Q 値	
MTR_Q_ASR_KP	29	速度 PI 比例ゲインの Q 値	
MTR_Q_ASR_KIDT	37	速度 PI 積分ゲイン×制御周期の Q 値	
MTR_Q_PLL_KP	32	PLL 比例ゲインの Q 値	
MTR_Q_PLL_KIDT	39[1 ショントモード時] 40[3 ショントモード時]	PLL 積分ゲイン×制御周期の Q 値	
MTR_Q_SPEED_LPF_CO	31	速度 LPF ゲインの Q 値	
MTR_Q_CURRENT_LPF_CO	31	電流 LPF ゲインの Q 値	
MTR_Q_CURRENT_REPO_LPF_CO	31	電流再現用 LPF ゲインの Q 値	
MTR_Q_DAMP_K	28	ダンピング制御ゲインの Q 値	
MTR_Q_DAMP_HPF_CO	31	ダンピング制御 HPF ゲインの Q 値	
MTR_Q_DAMP_SL_RATE	33	ダンピング制御速度リミットの Q 値	
MTR_Q_PHERR_LPF_CO	31	位相誤差 LPF の Q 値	
MTR_Q_OL2CL_K1	40	センサレス切り替え制御のゲインの Q 値	
MTR_Q_FW_I	28	弱め界磁制御用電流の Q 値	
MTR_Q_FW_I_SQ	26	弱め界磁制御における電流の二乗の Q 値	
MTR_Q_KE_DIV_2LDQ	31	磁束の Q 値 / $(2 \times (q \text{ 軸インダクタンス} - d \text{ 軸インダクタンス}))$	
MTR_Q_FW_KP	26	弱め磁束制御の PI 比例ゲインの Q 値	
MTR_Q_FW_KIDT	42[1 ショントモード時] 43[3 ショントモード時]	弱め磁束制御の PI 積分ゲイン×制御周期の Q 値	
MTR_Q_FW_LPF_CO	31	弱め磁束制御の速度偏差 LPF の Q 値	
MTR_Q_DO_LPF_CO	31	外乱抑制 LPF の Q 値	
MTR_Q_J_DIV_TC_P2M	34	動摩擦係数 / (極対数 ² ×誘起電圧定数)の Q 値	
MTR_Q_V_PHASE_LEAD_COEF	31[1 ショントモード時] 30[3 ショントモード時]	電圧位相進み係数の Q 値	
MTR_Q_DIV_DSP	32	DSP 関数割り算の Q 値(変更不可)	
MTR_Q_SIN_COS_DSP	28	DSP 関数の三角関数の Q 値(変更不可)	
MTR_Q_DUTY	12	PWM デューティの Q 値	
MTR_Q_PI_RUN_ACR_ERR	25[1 ショントモード時] 24[3 ショントモード時]	ACR PI エラーの Q 値	
MTR_Q_PI_RUN_PLL_ERR	21[1 ショントモード時] 20[3 ショントモード時]	PLL PI エラーの Q 値	
MTR_Q_PI_RUN_ACR_INTG	12	ACR INTG の Q 値	
MTR_Q_PI_RUN_ASR_ERR	23	ASR PI エラーの Q 値	
MTR_Q_PI_RUN_ASR_INTG	12	ASR INTG の Q 値	

表 3-69 “mtr_main.h” マクロ定義一覧

マクロ	定義値	内容	備考
MODE_INACTIVE	0x00	インアクティブモード	
MODE_ACTIVE	0x01	アクティブモード	
MODE_ERROR	0x02	エラーモード	
SIZE_STATE	3	モード状態数	

表 3-70 “r_mtr_ics.h” マクロ定義一覧

マクロ	定義値	内容	備考
TS_300US	-	ICS 通信速度	
ICS_BRR	4	ICS ビットレートレジスタ選択	
MTR_ICS_DECIMATION	2[1 シャントモード時] 5[3 シャントモード時]	ICS 処理間引き回数	
ICS_ADDR	0xFE00	ICS 用アドレス指定	
ICS_INT_LEVEL	3	ICS 用割り込みレベル設定	
ICS_NUM	0x40	ICS 通信データサイズ	
ICS_MODE	0	ICS 割り込みモード設定	

表 3-71 “r_mtr_ra2t1.h” マクロ定義一覧 [1/2]

マクロ	定義値	内容	備考
USE_PWMOPA	MTR_SET	PWMOPA による過電流使用の選択	
USE_ADC_FAST_MODE	MTR_SET	高速 ADC 変換モードの選択	
USE_TIME_MEASURE	MTR_SET	実行時間測定の選択	
USE_DUTY_SHIFT	MTR_CLR	デューティサイクルシフトの選択	
USE_U_PHASE_CURRENT	MTR_CLR	U 相電流測定の選択	
TRD_SKIPPING_COUNT	CP_INT_DECIMATION	TRD 割り込み間引き数	
MTR_PWM_TIMER_FREQ	CP_PWM_TIMER_FREQ	PWM タイマ周波数[kHz]	
MTR_INTVAL_TIMER_FREQ	CP_INTVAL_TIMER_FREQ	インターバル・タイマ周波数[kHz]	
MTR_CARRIER_FREQ	CP_CARRIER_FREQ	キャリア割り込み周波数[kHz]	
MTR_DEADTIME	IP_DEADTIME	デッドタイム [μs]	
MTR_DEADTIME_CNT	$(\text{int}16_t)(\text{MTR_DEADTIME} * \text{MTR_PWM_TIMER_FREQ})$	デッドタイム設定値	
MTR_CARRIER_CNT	$(\text{uint}16_t)(\text{MTR_PWM_TIMER_FREQ} * 1000 / \text{MTR_CARRIER_FREQ} * 0.5f)$	キャリア設定値	
MTR_HALF_CARRIER_CNT	$(\text{uint}16_t)(\text{MTR_CARRIER_CNT} * 0.5f)$	キャリア設定値(中間値)	
MTR_HALF_DEADTIME_CNT	$(\text{uint}16_t)(\text{MTR_DEADTIME_CNT} * 0.5f)$	デッドタイム中間値	
MTR_DUTY_RANGE_CNT	$(\text{uint}16_t)((\text{MTR_CARRIER_CNT} - \text{MTR_DEADTIME_CNT}) * 0.5f)$	変調率範囲のカウント値	
MTR_CURRENT_ADCONV_TIME	$\text{CP_AD_CONVERSION_TIME} * 2.0f$	A/D 変換開始を進める時間 [μs]	[3 シャントモード時]
MTR_ADC_SH_CNT	44	サンプルホールドカウンタ値	
MTR_ADC_RINGING_CNT	CP_AD_RINGING_WAIT_CNT	A/D リンギングカウンタ値	
MTR_ADC_TS_CNT	$(\text{MTR_ADC_SH_CNT} + \text{MTR_ADC_RINGING_CNT} + 1) / 2$	A/D 変換時間	
MTR_VOLTAGE_LIMIT_OFFSET	$(\text{int}32_t)((\text{MTR_CURRENT_ADCONV_TIME} + \text{MTR_DEADTIME} * 2) / (1000 / \text{MTR_CARRIER_FREQ})) * 0.5f * (1 \ll \text{MTR_Q_VOLTAGE})$	電圧のオフセット制限値 [PU(V)]	
MTR_CENTER_AMPLITUDE_CNT	$(\text{MTR_CARRIER_CNT} + \text{MTR_ADC_TS_CNT} + \text{MTR_DEADTIME_CNT} + 1) / 2$	PWM タイマの振幅中心	[3 シャントモード・二相変調時]
MTR_OFFSET_CENTER_AMPLITUDE_CNT	MTR_CENTER_AMPLITUDE_CNT	オフセット計測時のタイマの振幅中間値	
MTR_CURRENT_ADCONV_TIME	$\text{CP_AD_CONVERSION_TIME} * 2.0f$	A/D 変換開始を進める時間 [μs]	
MTR_VOLTAGE_LIMIT_OFFSET	0	電圧のオフセット制限値 [PU(V)]	
MTR_CENTER_AMPLITUDE_CNT	$(\text{int}16_t)(\text{MTR_HALF_CARRIER_CNT} + \text{MTR_DEADTIME_CNT})$	PWM タイマの振幅中心	
MTR_OFFSET_CENTER_AMPLITUDE_CNT	MTR_CENTER_AMPLITUDE_CNT	オフセット計測時のタイマの振幅中間値	[1 シャントモード・二相変調時]
MTR_VOLTAGE_LIMIT_OFFSET	$(\text{int}16_t)((\text{MTR_DEADTIME} * 2) / (1000 / \text{MTR_CARRIER_FREQ})) * 0.5f * (1 \ll \text{MTR_Q_VOLTAGE})$	電圧のオフセット制限値 [PU(V)]	
MTR_VOLTAGE_LIMIT_OFFSET	0	電圧のオフセット制限値 [PU(V)]	

表 3-72 “r_mtr_ra2t1.h” マクロ定義一覧 [2/2]

マクロ	定義値	内容	備考
MTR_DEADTIME_RATIO	$(\text{MTR_DEADTIME} * 1.0f * \text{MTR_CARRIER_FREQ}) / 1000.0f$	デッドタイム補償係数	
MTR_DEADTIME_CURRENT_LIMIT	$(\text{float})(\text{IP_CURRENT_RANGE}) / (1 < \text{MTR_ADC_12BIT})$	電流制限値	
MTR_CTRL_PERIOD	$(\text{MTR_INT_DECIMATION} + 1) / (\text{MTR_CARRIER_FREQ} * 1000)$	電流制御周期	
MTR_SPEED_CTRL_PERIOD	CP_SPEED_CTRL_PERIOD	速度制御周期	
MTR_ADC_DATA_SHIFT	0	AD 変換値シフト量	
MTR_ADC_OFFSET	0x7FFF	AD 変換値オフセット	
MTR_CHARGE_CAP_WAIT_CNT	IP_CHARGE_CAP_WAIT_CNT	コンデンサ充電時間	
MTR_V_PHASE_LEAD_COEF	FIX_fromfloat(CP_V_PHASE_LEAD_COEF, MTR_Q_V_PHASE_LEAD_COEF)	電圧位相進み係数	
MTR_LOWEST_DUTY_CNT	MTR_CARRIER_CNT + MTR_DEADTIME_CNT	タイマカウント：最低	
MTR_MAX_DUTY_CNT	MTR_CARRIER_CNT - 1	タイマカウント：最大	
MTR_MIN_DUTY_CNT	1	タイマカウント：最小	
POEG_RECOVER_TIMEOUT	3	POEG 解除タイムアウトカウント値	
ADC_CONVERT_TIMEOUT	0xFFFF	A/D 変換完了タイムアウトカウント値	

表 3-73 “r_mtr_common.h” マクロ定義一覧

マクロ	定義値	内容	備考
R_MTR_ADCR_IDC	ADC_CHANNEL_0	DC リンク電流のチャンネル選択	[1 シャントモード時]
R_MTR_ADCR_IDCDUP	ADC_CHANNEL_DUPLEX	DC リンク電流ダブルトリガのチャンネル選択	
R_MTR_ADCR_VDC	ADC_CHANNEL_8	インバータ母線電圧のチャンネル選択	
R_MTR_ADC_SCAN_MASK_VW	0x00000006U	VW 電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_UVW	0x00000007U	UVW 電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_VW_IDC	0x00000007U	VW 電流、DC リンク電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_VDC	0x00000100U	インバータ母線電圧のチャンネルマスク	
R_MTR_ADCR_IU	ADC_CHANNEL_0	Iu の ADCR 選択	[3 シャントモード時]
R_MTR_ADCR_IV	ADC_CHANNEL_1	Iv の ADCR 選択	
R_MTR_ADCR_IW	ADC_CHANNEL_2	Iw の ADCR 選択	
R_MTR_ADS_CHANNEL_IDC	ADC_CHANNEL_0	DC リンク電流のチャンネル選択	
R_MTR_ADS_CHANNEL_VDC	ADC_CHANNEL_8	インバータ母線電圧のチャンネル選択	
R_MTR_ADC_SCAN_MASK_VW	0x00000006U	VW 電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_UVW	0x00000007U	UVW 電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_VW_IDC	0x00000007U	VW 電流、DC リンク電流のスキャンマスク	
R_MTR_ADC_SCAN_MASK_VDC	0x00000100U	インバータ母線電圧のチャンネルマスク	

表 3-74 “rm_mtr_common.h” マクロ定義一覧

マクロ	定義値	内容	備考
MTR_TWOPi	$2 * 3.14159265f$	2π	
MTR_SQRT_3	$1.7320508f$	$\sqrt{3}$	
MTR_CW	1	CW	
MTR_CCW	-1	CCW	
MTR_ON	0	オン	
MTR_OFF	1	オフ	
MTR_CLR	0	フラグクリア	
MTR_SET	1	フラグセット	
MTR_OPL	0	オープンループ	
MTR_CLL	1	クローズドループ	
MTR_UNREACHED	0	未到達	
MTR_REACHED	1	到達	
MTR_WAIT_CNT_300US	1500	300us 待ち	
MTR_ADC_12BIT	12	12bit ADC	

表 3-75 “rm_mtr_parameter.h” マクロ定義一覧

マクロ	定義値	内容	備考
MTR_PWM_DUTY_RANGE	$(1L << MTR_Q_VMOD)-1$	Duty レンジ	
MTR_INPUT_V	IP_INPUT_V	入力電圧	
MTR_HALF_VDC	$MTR_INPUT_V * 0.5f$	電圧の 50%	
MTR_MCU_ON_V	$MTR_INPUT_V * 0.8f$	電圧の 80%	
MTR_OVERVOLTAGE_LIMIT	IP_OVERVOLTAGE_LIMIT	過電圧リミット値	
MTR_UNDERVOLTAGE_LIMIT	IP_UNDERVOLTAGE_LIMIT	低電圧リミット値	
MTR_RECIV_COEF	$(int32_t)(2 * PU_BASE_VOLTAGE_V / IP_VDC_RANGE * (1 << MTR_Q_RECIV))$	電圧基準値の逆数	
MTR_ANGLE_RANGE	$(int32_t)(MTR_TWOPi * PU_SF_ANGLE * (1 << MTR_Q_ANGLE))$	角度レンジ 2π	
MTR_ANGLE_QUAT_RANGE	$(int32_t)(MTR_ANGLE_RANGE / 4)$	角度レンジ $\pi/2$	
MTR_OVERCURRENT_LIMIT	CP_OC_LIMIT	電流リミット値	
MTR_I_LIMIT_VD	$1.5f * IP_INPUT_V * 0.5f$	Vd の電流 PI リミット	
MTR_I_LIMIT_VQ	$1.5f * IP_INPUT_V * 0.5f$	Vq の電流 PI リミット	
MTR_RPM_RAD	$(MP_POLE_PAIRS * MTR_TWOPi) / 60.0f$	[rpm]から[rad/s]への変換	
MTR_SPEED_LIMIT_RAD	$CP_SPEED_LIMIT_RPM * MTR_RPM_RAD$	速度制限値[rad/s]	
MTR_MAX_SPEED_RAD	$CP_MAX_SPEED_RPM * MTR_RPM_RAD$	最大速度[rad/s]	
MTR_LIMIT_IQ	$MP_RATED_CURRENT * MTR_SQRT_3$	速度 PI の出力制限値	
MTR_I_LIMIT_CURRENT	$MP_RATED_CURRENT * MTR_SQRT_3$	速度 PI の積分項の出力制限値	
MTR_CL2OL_SPEED_RAD	$CP_CL2OL_SPEED_RPM * MTR_RPM_RAD$	センサレスからオープンループへの切り替え速度 [rad/s]	
MTR_OL2CL_SPEED_RAD	$CP_OL2CL_SPEED_RPM * MTR_RPM_RAD$	オープンループからセンサレスへの切り替え速度[rad/s]	
MTR_RECIM	$1.0f/(MP_BEMF_CONSTANT * PU_SF_BEMF_CONST)$	誘起電圧定数の逆数[PU]	
MTR_DRAW_IN_WAIT_CNT	$CP_DRAW_IN_WAIT_TIME / MTR_SPEED_CTRL_PERIOD$	引き込み時間のカウンタ値	
MTR_SWITCH_COUNT	$(uint16_t)(CP_OL2CL_SWITCH_TIME / MTR_SPEED_CTRL_PERIOD)$	センサレス切り替え制御時間カウンタ値[cnt]	
MTR_CL2OL_JUDGE_WAIT_CNT	$(uint16_t)(CP_CL2OL_JUDGE_WAIT_TIME / (CP_SPEED_CTRL_PERIOD * 1000))$	オープンループ判定の待ち回数	

表 3-76 “rm_mtr_statemachine.h” マクロ定義一覧

マクロ	定義値	内容	備考
MTR_MODE_INIT	0x00	初期化モード	
MTR_MODE_DRIVE	0x01	ドライブモード	
MTR_MODE_STOP	0x02	ストップモード	
MTR_SIZE_STATE	3	ステート数	
MTR_EVENT_STOP	0x00	ストップイベント	
MTR_EVENT_DRIVE	0x01	ランイベント	
MTR_EVENT_ERROR	0x02	エラーイベント	
MTR_EVENT_RESET	0x03	リセットイベント	
MTR_SIZE_EVENT	4	イベント数	
MTR_STATEMACHINE_ERROR_NONE	0x00	ステートマシンエラーなし	
MTR_STATEMACHINE_ERROR_EVENTOUTBOUND	0x01	イベントインデックスが範囲外	
MTR_STATEMACHINE_ERROR_STATEOUTBOUND	0x02	ステートインデックスが範囲外	
MTR_STATEMACHINE_ERROR_ACTIONEXCEPTION	0x04	アクションの失敗	

表 3-77 “r_mtr_foc_less_speed.h” マクロ定義一覧 [1/2]

マクロ	定義値	内容	備考
MTR_ERROR_NONE	0x0000	エラーなし	
MTR_ERROR_OVER_CURRENT_HW	0x0001	過電流エラー	
MTR_ERROR_OVER_VOLTAGE	0x0002	過電圧エラー	
MTR_ERROR_OVER_SPEED	0x0004	過速度エラー	
MTR_ERROR_HALL_TIMEOUT	0x0008	ホールタイムアウトエラー	
MTR_ERROR_BEMF_TIMEOUT	0x0010	誘起電圧タイムアウトエラー	
MTR_ERROR_HALL_PATTERN	0x0020	ホールパターンエラー	
MTR_ERROR_BEMF_PATTERN	0x0040	誘起電圧パターンエラー	
MTR_ERROR_UNDER_VOLTAGE	0x0080	低電圧エラー	
MTR_ERROR_OVERCURRENT_SW	0x0100	ソフトウェア過電流エラー	
MTR_ERROR_CHARGE_CAP_TIMEOUT	0x0400	コンデンサ充電タイムアウトエラー	
MTR_ERROR_UNKNOWN	0xffff	未定義エラー	
MTR_ID_ZERO_CONST	0	d 軸電流 0 制御	
MTR_ID_MANUAL	1	d 軸電流マニュアル制御	
MTR_ID_MTPA2FW	2	d 軸電流最大トルク & 弱め磁束制御	
MTR_IQ_ZERO_CONST	0	q 軸電流 0 制御	
MTR_IQ_MANUAL	1	q 軸電流マニュアル制御	
MTR_IQ_OL2CL	2	q 軸電流のセンサレス切り替え制御	
MTR_IQ_SPEED_PI_OUTPUT	3	速度 PI 制御出力	
MTR_SPEED_ZERO_CONST	0	速度 0 制御	
MTR_SPEED_MANUAL	1	速度マニュアル制御	
MTR_STOP	0x00	停止中	
MTR_DRIVE	0x81	駆動中	
MTR_DRIVE_INIT	0x82	駆動初期化	
MTR_DRIVE_START	0x83	駆動開始	
MTR_OFFSET_CALC_EXE	0x84	オフセット計測	
MTR_DRIVE_END	0x01	駆動停止	
MTR_MASK_RUN_MODE	0x80	駆動中マスク	
MTR_CTRL_NONE	0x00	制御停止	
MTR_CTRL_OPL_DRAWIN	0x01	引き込み制御	
MTR_CTRL_OPL_FORCE_SYNC	0x02	強制同期制御	
MTR_CTRL_OPL_OPL2CLL	0x03	オープンループクローズドループ 切り替え制御	
MTR_CTRL_OPL_CLL2OPL	0x04	クローズドループオープンループ 切り替え遷移中	
MTR_CTRL_OPL_FORCE_SYNC_ACCEL	0x05	オープンループ速度制御	
MTR_CTRL_CLL_OPL2CLL	0x80	オープンループクローズドループ 切り替え遷移中	
MTR_CTRL_CLL_SPEED_CTRL	0x81	クローズドループ速度制御	
MTR_DRIVE_OPL	0x00	速度オープンループ制御	
MTR_DRIVE_CLL	0x80	速度クローズドループ制御	
MTR_PHASE_U	0	U 相	
MTR_PHASE_V	1	V 相	
MTR_PHASE_W	2	W 相	

表 3-78 “r_mtr_foc_less_speed.h” マクロ定義一覧 [2/2]

マクロ	定義値	内容	備考
MTR_DRV_UVW	1	デューティ大小関係 $U>V>W$	[1 シャントモード時]
MTR_DRV_UWV	2	デューティ大小関係 $U>W>V$	
MTR_DRV_VUW	3	デューティ大小関係 $V>U>W$	
MTR_DRV_VWU	4	デューティ大小関係 $V>W>U$	
MTR_DRV_WUV	5	デューティ大小関係 $W>U>V$	
MTR_DRV_WVU	6	デューティ大小関係 $W>V>U$	
MTR_DRV_MASK	0x0F	デューティ大小関係のマスク	
MTR_DRV_CURRENT_UNAVAILABLE	0x10	電流検出不可フラグ	
MTR_DRV_DUTY_SHIFT_RUN	0x20	PWM Duty ずらし実行	
MTR_DRV_PWM_SHIFT_RUN	0x40	PWM ずらし実行	
MTR_CROSS_MID_MIN	1	Duty 中間と最小値がクロス	
MTR_CROSS_MAX_MID	2	Duty 最大と中間値がクロス	
MTR_PWM_SHIFT_TIMER_HALF	(uint16_t)(MTR_CARRIER_CNT)	キャリア周期の半分	
MTR_PWM_SHIFT_DEADTIME	(uint16_t)(MTR_DEADTIME_CNT)	デッドタイムカウント値	
MTR_PWM_SHIFT_AD_CONVERT_CNT	(uint16_t)(MTR_PWM_TIMER_FREQ * CP_AD_CONVERSION_TIME)	AD 変換シフトカウント値	
MTR_PWM_SHIFT_TIMER_SET_100	(uint16_t)(0)	100% Duty カウント値 (デッドタイム無)	
MTR_PWM_SHIFT_TIMER_NEAR_100	(uint16_t)(MTR_DEADTIME_CNT)	100% Duty カウント値 (デッドタイム有)	
MTR_PWM_SHIFT_TIMER_NEAR_0	(uint16_t)(MTR_CARRIER_CNT)	0% Duty カウント値 (デッドタイム無)	
MTR_PWM_SHIFT_TIMER_SET_0	(uint16_t)(MTR_CARRIER_CNT + MTR_DEADTIME_CNT)	0% Duty カウント値 (デッドタイム無)	

3.6 割り込み処理の仕様

本サンプルソフトウェアの割り込み処理について説明します。

割り込み処理は、3 シャントモードはキャリア周期割り込み(50[μs]で間引き無し)、1 シャントモードはキャリア周期割り込み(50[μs]で間引き 1 回 100[μs])と 1[ms]周期割り込みの 2 つの周期割り込みにより構成されます。

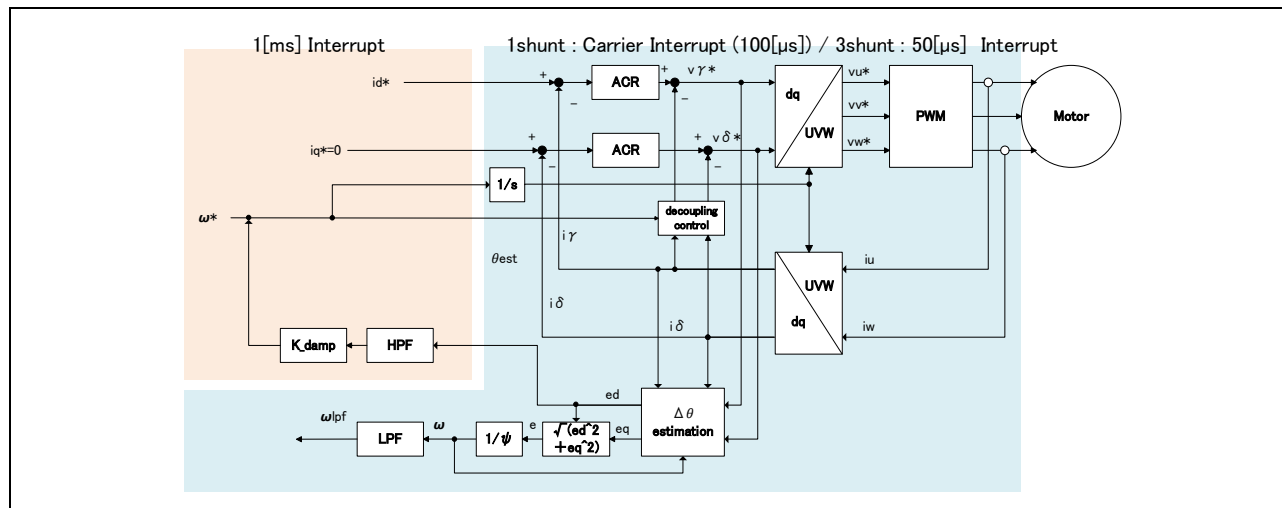


図 3-12 制御ブロック内の割り込み処理(オープンループ制御時)

オープンループ時の駆動概略を説明します。d 軸電流指令値を与えて、速度 0 から指令速度をランプ状に上昇させ、速度指令値を積分した角度情報を用いて角度を更新します。軸誤差推定器から出力される誘起電圧値を用いて推定速度を観測します。

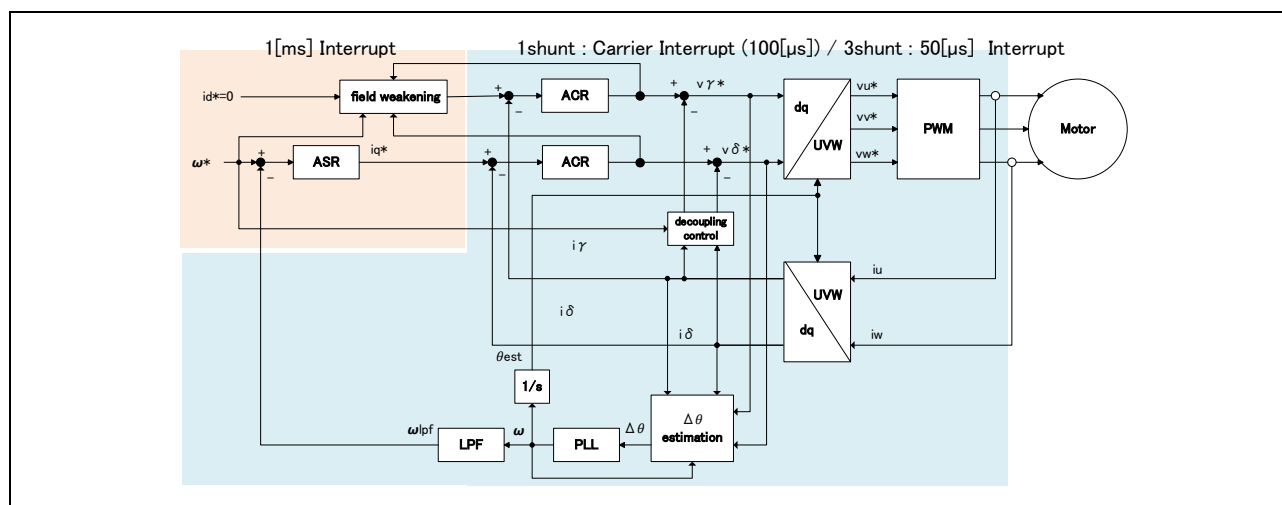


図 3-13 制御ブロック内の割り込み処理(クローズドループ制御時)

クローズドループ時の駆動概略を説明します。d 軸電流を 0 として、速度指令値と軸誤差推定器から出力される軸誤差から求める推定速度の偏差を速度制御器に入力し、出力として q 軸電流指令値を得ます。推定速度を積分した角度情報を用いて角度を更新します。

割り込み処理の占有率を 60%程度とするため、1 シャントモード時のキャリア周期割り込みは 1 回間引きの設定になっています。従って電流制御系の制御周期は 50[μs]の 1 回間引きとなるため 100[μs]です。

1[ms]周期割り込み内では速度制御系の処理と dq 軸電流の指令値を出力します。速度制御系の制御周期は 1[ms]です。

1 シャントモード、3 シャントモードの割り込みタイミングは同じです。キャリア周期割り込みは 50[μs] の 1 回間引き設定のため 100[μs]ごとの谷で発生し、PWM デューティも同様に谷で更新されます。

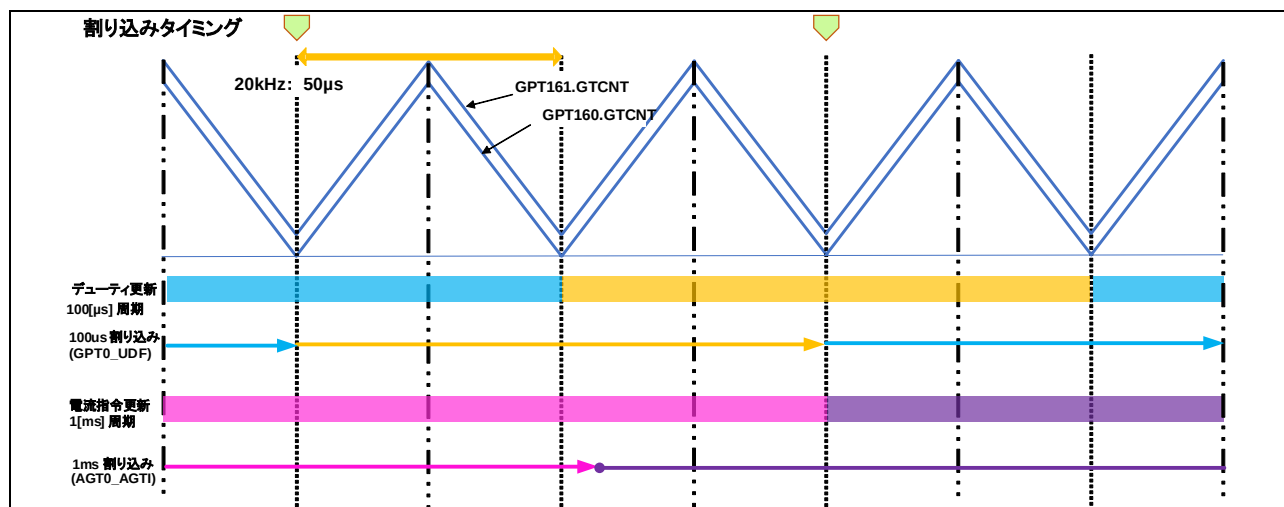


図 3-14 割り込み発生と出力値の更新イメージ

3.7 制御フロー（フローチャート）

3.7.1 メイン処理

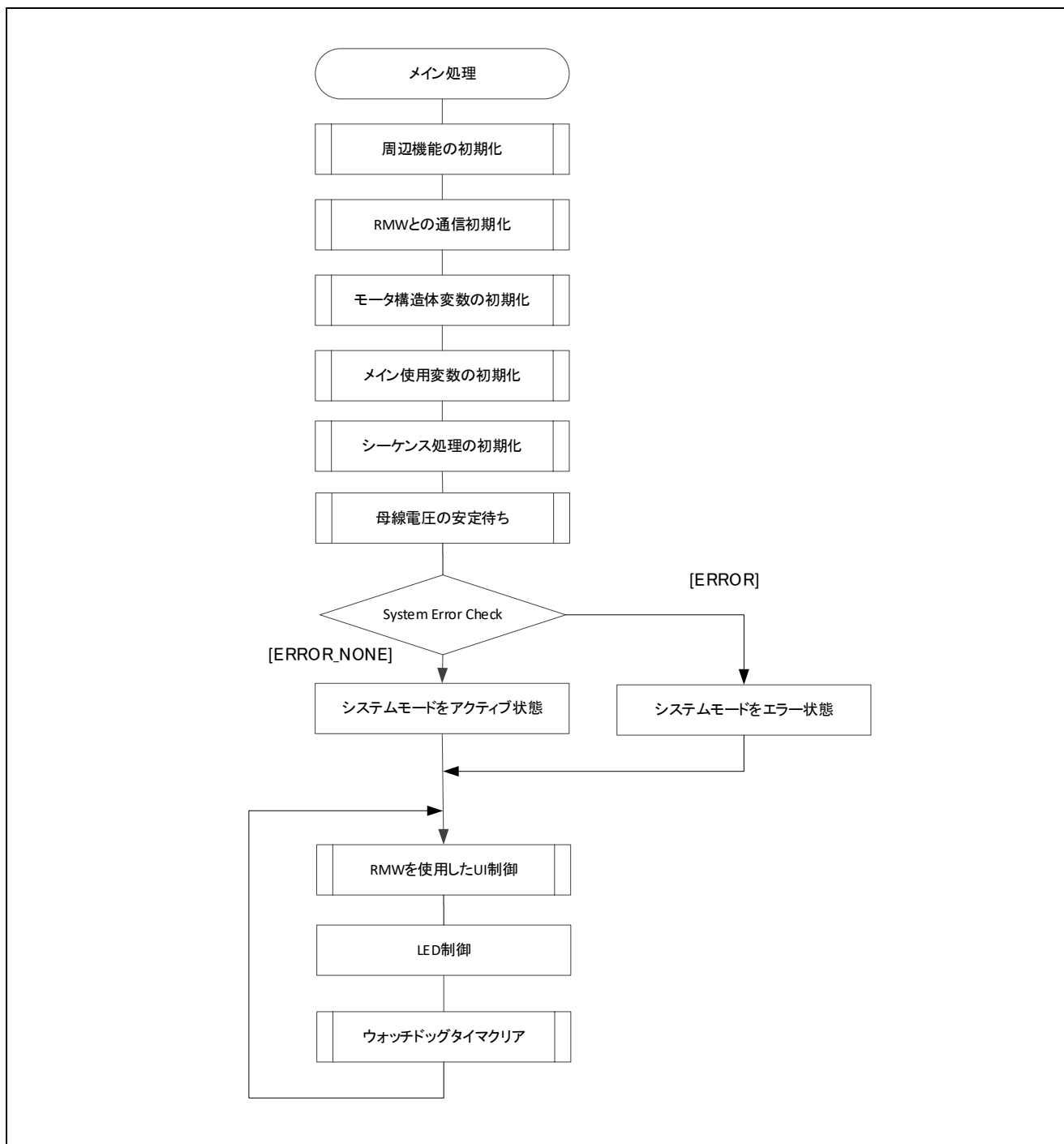


図 3-15 メイン処理フローチャート

3.7.2 キャリア周期割り込み処理 (1 シャントモード時)

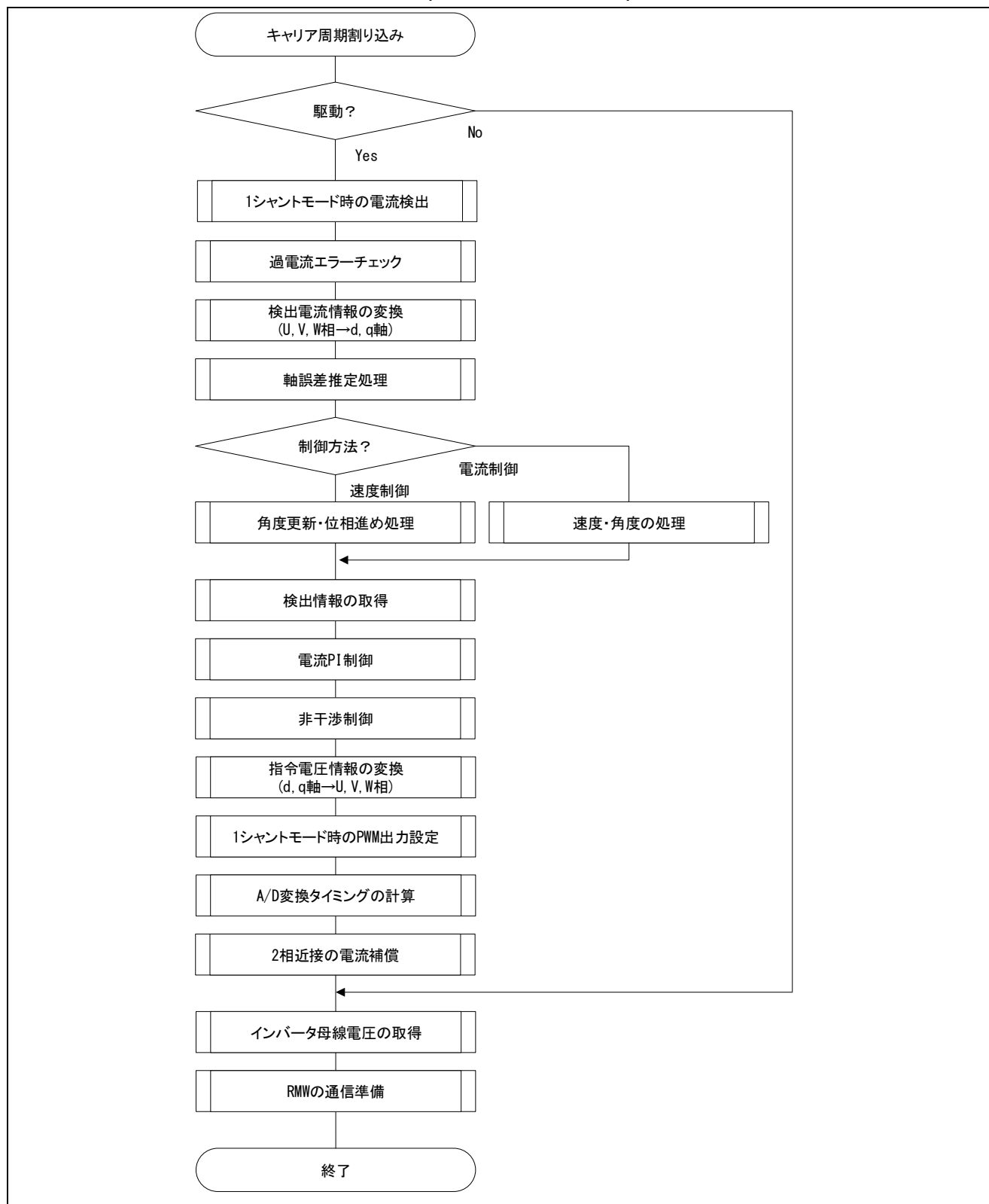


図 3-16 キャリア周期割り込み処理フローチャート

3.7.3 キャリア周期割り込み処理 (3 シャントモード時)

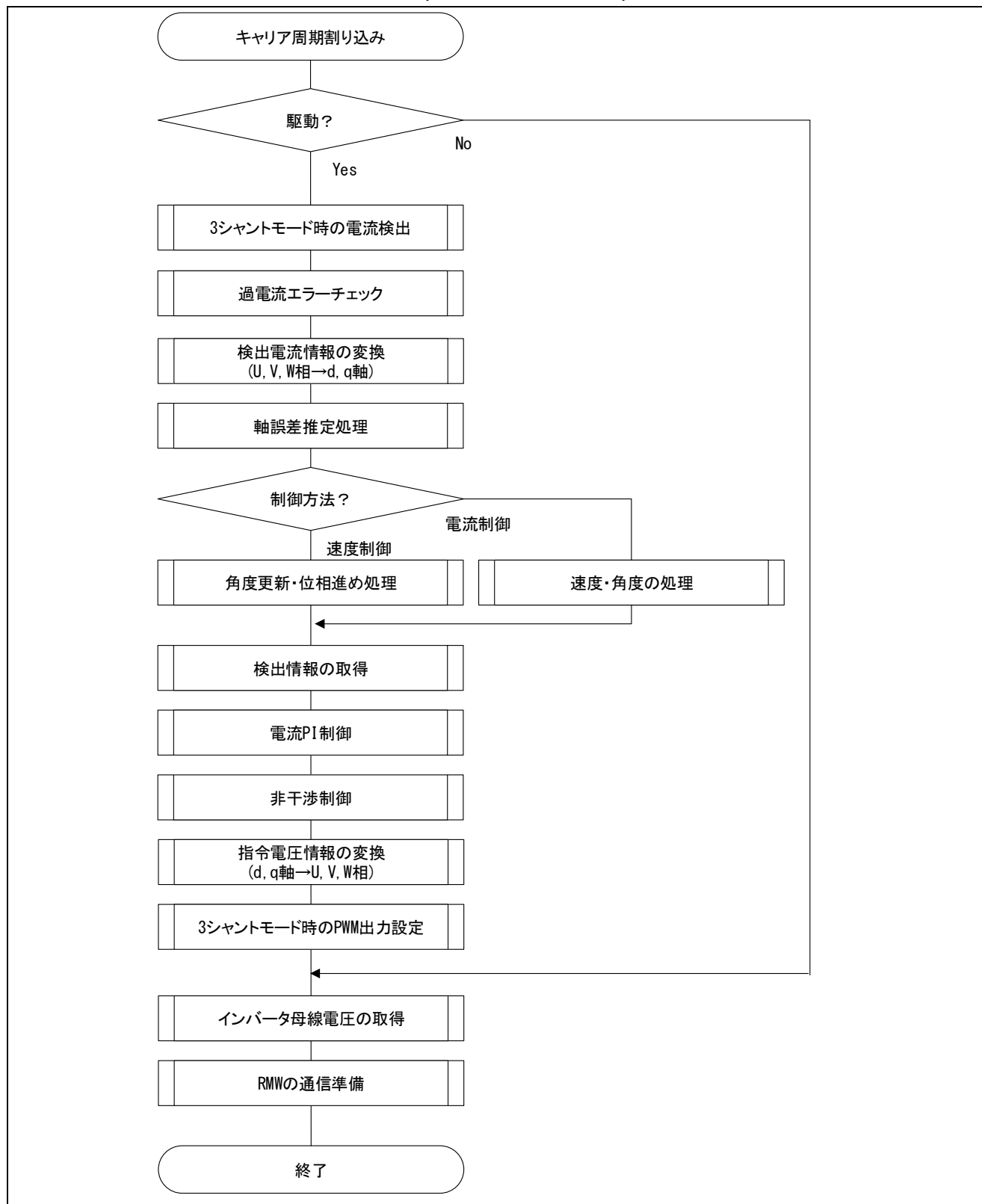


図 3-17 キャリア周期割り込み処理フローチャート

3.7.4 1 [ms]割り込み処理

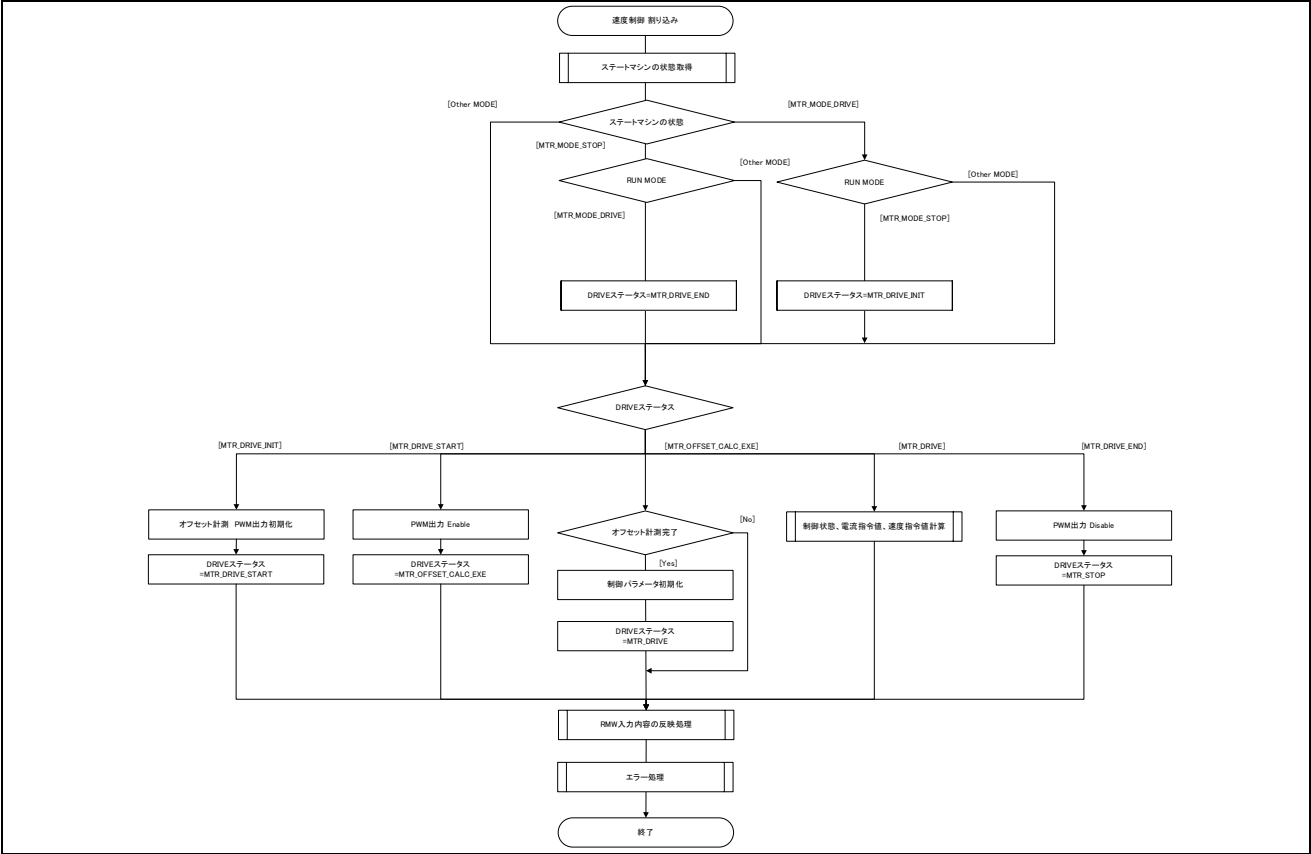


図 3-18 1 [ms]割り込み処理フローチャート

4. モータ制御開発支援ツール「Renesas Motor Workbench」の利用方法

4.1 概要

本アプリケーションノート対象サンプルプログラムでは、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。

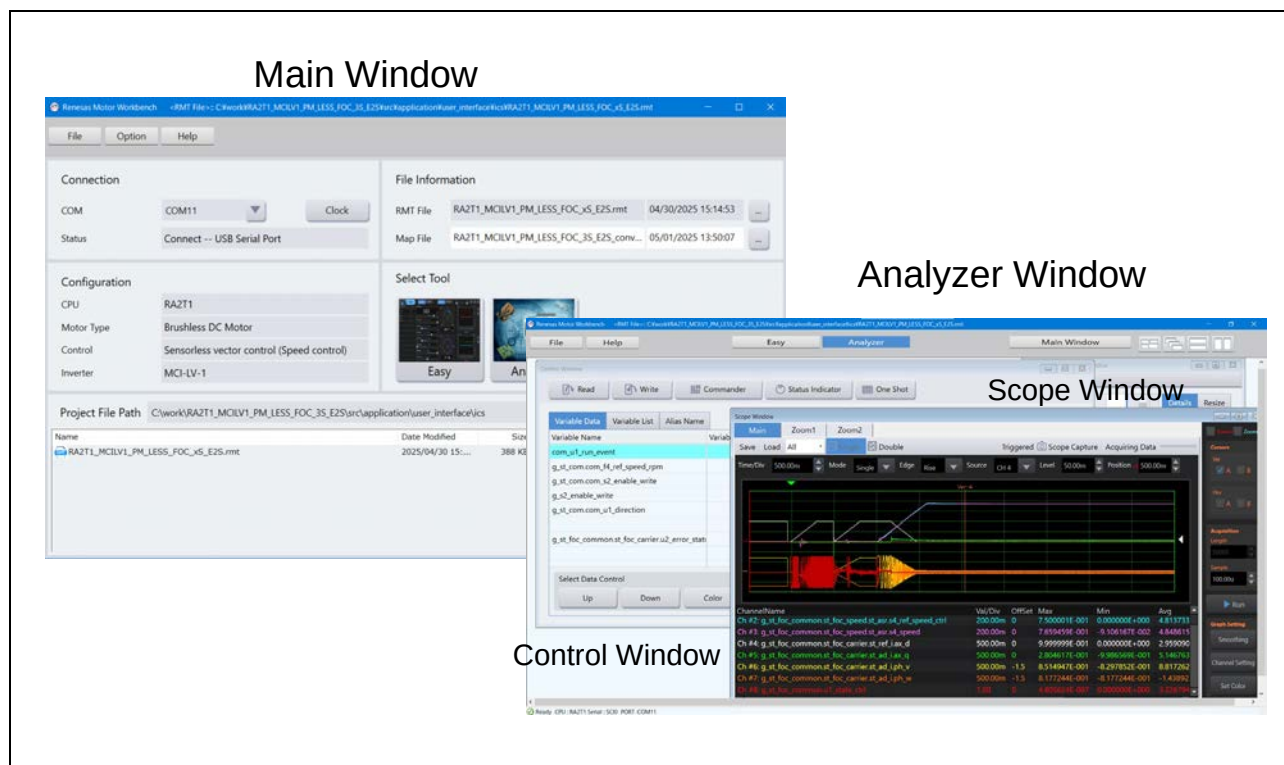


図 4-1 Renesas Motor Workbench 外観

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



- 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]/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 Analyzer ボタンをクリックし、Analyzer 機能を起動する。
(起動すると Analyzer Window 画面に切り替わります。)
- ⑤ ”Analyzer 操作例”を元にモータを駆動させる。

4.2 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 4-1 に示します。なお、これらの変数の値は g_st_com.s2_enable_write に g_s2_enable_write と同じ値を書き込んだ際に反映されます。ただし、(*) が付けられた変数は g_st_com.s2_enable_write に依存しません。

表 4-1 Analyzer 機能入力用変数一覧(1/2)

変数名	型	内容	備考 ([]: 反映先の変数名)
com_u1_run_event (*)	uint8_t	ランモード遷移用変数 0: ストップイベント 1: ドライブイベント 2: エラーイベント 3: リセットイベント	[g_u1_run_event]
g_st_com.st_com.f4_acr_nf_hz	float	電流 PI 制御固有周波数 [Hz]	[g_st_ctrl_params.st_com.f4_acr_nf_hz]
g_st_com.st_com.f4_acr_lpf_cof_hz	float	電流 LPF 固有周波数 [Hz]	[g_st_ctrl_params.st_com.f4_acr_lpf_cof_hz]
g_st_com.st_com.f4_acr_deadband_lsb	float	電流 PI のデッドバンド [LSB]	[g_st_ctrl_params.st_com.f4_acr_deadband_lsb]
g_st_com.st_com.f4_asr_nf_hz	float	速度 PI 固有周波数 [Hz]	[g_st_ctrl_params.st_com.f4_asr_nf_hz]
g_st_com.st_com.f4_asr_lpf_cof_hz	float	速度 LPF 固有周波数 [Hz]	[g_st_ctrl_params.st_com.f4_asr_lpf_cof_hz]
g_st_com.st_com.f4_asr_deadband_lsb	float	速度 PI のデッドバンド [LSB]	[g_st_ctrl_params.st_com.f4_asr_deadband_lsb]
g_st_com.st_com.f4_asr_ki_aug	float	速度 PI 制御積分器増幅率	[g_st_ctrl_params.st_com.f4_asr_ki_aug]
g_st_com.st_com.f4_pll_nf_hz	float	PLL 固有周波数 [Hz]	[g_st_ctrl_params.st_com.f4_pll_nf_hz]
g_st_com.st_com.f4_pll_deadband_lsb	float	PLL のデッドバンド [LSB]	[g_st_ctrl_params.st_com.f4_pll_deadband_lsb]
g_st_com.com_u1_direction	uint8_t	回転方向 0 : CW 1 : CCW	[g_st_foc_common.st_foc_speed.st_asr.s1_ref_dir]
g_st_com.com_f4_mtr_r	float	抵抗 [Ω]	[g_st_foc_common.st_foc_carrier.st_motor.s4_mtr_r]
g_st_com.com_f4_mtr_ld	float	d 軸インダクタンス[H]	[g_st_foc_common.st_foc_carrier.st_motor.s4_mtr_ld]
g_st_com.com_f4_mtr_lq	float	q 軸インダクタンス[H]	[g_st_foc_common.st_foc_carrier.st_motor.s4_mtr_lq]
g_st_com.com_f4_mtr_m	float	誘起電圧定数 [$V \cdot s / rad$]	[g_st_foc_common.st_foc_carrier.st_motor.s4_mtr_ke] [g_st_foc_common.st_foc_carrier.st_phe.s4_coef_reci_ke]
g_st_com.com_f4_mtr_j	float	イナーシャ [$kg \cdot m^2 / (rad^2)$]	[g_st_foc_common.st_foc_carrier.st_motor.s4_mtr_j]
g_st_com.com_f4_mtr_d0	float	静止摩擦係数 [$kg \cdot m^2 / (rad \cdot s^2)$]	[g_st_ctrl_params.f4_d0_div_pm]
g_st_com.com_f4_mtr_d1	float	動摩擦係数 [$kg \cdot m^2 / (rad^2 \cdot s)$]	[g_st_ctrl_params.f4_d1_div_p2m]
g_st_com.com_u2_mtr_pp	uint16_t	極対数	[g_st_foc_common.st_foc_carrier.st_motor.u2_mtr_pp]
g_st_com.com_u2_offset_calc_cnt	uint16_t	電流オフセット検出時間	[g_st_foc_common.st_foc_carrier.st_tscs.u4_offset_calc_cnt]
g_st_com.com_f4_ref_speed_rpm	float	目標速度 [rpm]	[g_st_foc_common.st_foc_speed.st_asr.s4_ref_speed]
g_st_com.com_f4_ramp_limit_speed_rpm	float	加速度制限値 [rpm/ms]	[g_st_foc_common.st_foc_speed.st_asr.s4_lim_ramp_speed]
g_st_com.com_f4_max_speed_rpm	float	最大速度 [rpm]	[g_st_foc_common.st_foc_speed.st_asr.s4_max_speed]
g_st_com.com_f4_cl2ol_speed_rpm	float	センサレスからオープンループへの切り替え速度[rpm]	[g_st_foc_common.st_foc_speed.st_asr.s4_th_cl2ol_speed]
g_st_com.com_f4_ol2cl_speed_rpm	float	オープンループからセンサレスへの切り替え速度[rpm]	[g_st_foc_common.st_foc_speed.st_asr.s4_th_ol2cl_speed]

表 4-2 Analyzer 機能入力用変数一覧(2/2)

変数名	型	内容	備考 ([]: 反映先の変数名)
g_st_com.com_f4_ol_ref_id	float	オープンループ d 軸指令電流[A]	[g_st_foc_common.st_foc_carrier.st_acr.s4_ref_id_ol]
g_st_com.com_f4_draw_in_wait_time	float	引き込み待ち時間 [sec]	[g_st_foc_common.st_foc_speed.u2_cnt_draw_in_wait_buff]
g_st_com.com_f4_init_asr_intg	float	センサレス移行時の ASR の積分項初期値	[g_st_foc_common.st_foc_speed.st_asr.s4_init_intg]
g_st_com.com_f4_ramp_limit_current	float	電流の上昇制限値[A/ms]	[g_st_foc_common.st_foc_carrier.st_acr.s4_lim_ramp_current]
g_st_com.com_f4_i_repro_cof_hz	float	電流再現用 LPF カットオフ周波数 [Hz]	[g_st_ctrl_params.f4_i_repro_cof_hz]
g_st_com.com_f4_mod_3ph2ph_speed_rpm	int32_t	三相変調から二相変調への切り替え速度[rpm]	[g_st_foc_common.st_foc_carrier.st_ssc.s.s4_mod_3ph2ph_speed_rad]
g_st_com.com_f4_mod_2ph3ph_speed_rpm	int32_t	二相変調から三相変調への切り替え速度[rpm]	[g_st_foc_common.st_foc_carrier.st_ssc.s.s4_mod_2ph3ph_speed_rad]
g_st_com.com_s2_ad_point_a_adj_cnt_3ph	int16_t	A/D 変換タイミング A の調整値 (三相変調)	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_a_adj_cnt_3ph]
g_st_com.com_s2_ad_point_b_adj_cnt_3ph	int16_t	A/D 変換タイミング B の調整値 (三相変調)	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_b_adj_cnt_3ph]
g_st_com.com_s2_ad_point_a_adj_cnt_2ph	int16_t	A/D 変換タイミング A の調整値 (二相変調)	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_a_adj_cnt_2ph]
g_st_com.com_s2_ad_point_b_adj_cnt_2ph	int16_t	A/D 変換タイミング B の調整値 (二相変調)	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_b_adj_cnt_2ph]
g_st_com.com_s2_ad_point_a_adj_cnt	int16_t	A/D 変換タイミング A の調整値	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_a_adj_cnt]
g_st_com.com_s2_ad_point_b_adj_cnt	int16_t	A/D 変換タイミング B の調整値	[g_st_foc_common.st_foc_carrier.st_ssc.s.s2_ad_point_b_adj_cnt]
g_st_com.com_s2_mod_2ph_bot_change_cnt	int16_t	二相変調遷移時のカウント回数	[g_st_foc_common.st_foc_carrier.st_mod.s2_cnt_2ph_bot_change]
g_st_com.com_f4_damp_hpf_cof_hz	float	ダンピング制御の HPF カットオフ周波数[Hz]	[g_st_ctrl_params.f4_damp_hpf_cof_hz]
g_st_com.com_f4_damp_zeta	float	ダンピング制御の減衰係数	[g_st_ctrl_params.f4_damp_zeta]
g_st_com.com_f4_damp_speed_limit_rate	float	ダンピング制御の速度制限値	[g_st_foc_common.st_foc_speed.st_ol2cl.st_damp.s4_speed_lim_rate]
g_st_com.com_f4_pherr_lpf_cof_hz	float	切り替え制御のための位相誤差の LPF カットオフ周波数[Hz]	[g_st_ctrl_params.f4_pherr_lpf_cof_hz]
g_st_com.com_f4_ol2cl_switch_time	float	センサレス切り替え時間[sec]	[g_st_foc_common.st_foc_speed.st_ol2cl.u2_cnt_switch]
g_st_com.com_s2_enable_write	int16_t	変数書き換え許可	[g_s2_enable_write]

4.3 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、図 4-1 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」を参照して下さい。

- モータを回転させる
 - ① “com_u1_run_event”, “g_st_com.f4_ref_speed_rpm”, “g_st_com.s2_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“g_st_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“g_st_com.f4_ref_speed_rpm”, “g_s2_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、“g_st_com.s2_enable_write”に④で確認した“g_s2_enable_write”と同じ値を入力する。
 - ⑥ “com_u1_run_event”の [Write]欄に“1”を入力する。
 - ⑦ “Write”ボタンを押す。

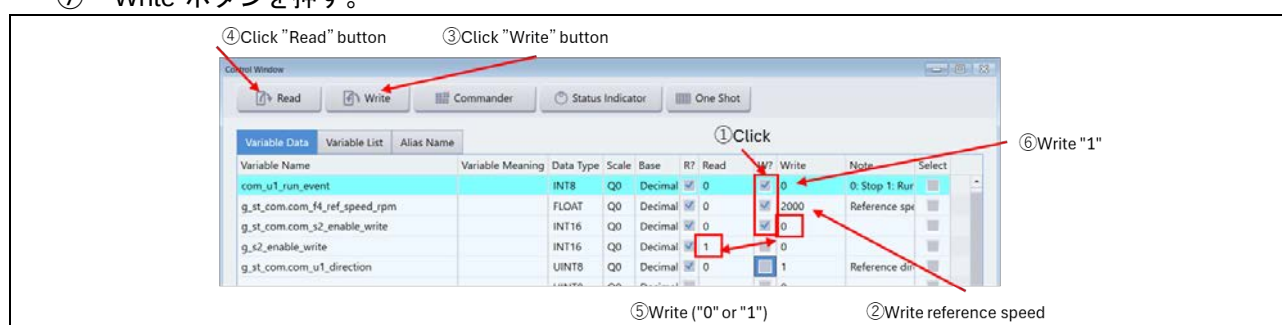


図 4-2 モータ回転の手順

- モータを停止させる
 - ① “com_u1_run_event”の[Write]欄に“0”を入力する。
 - ② “Write”ボタンを押す。

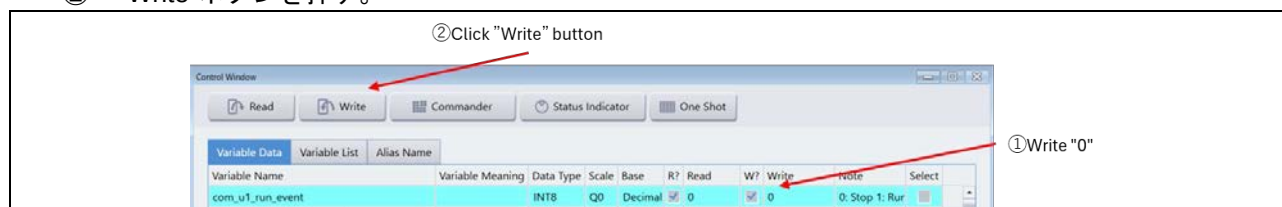


図 4-3 モータ停止の手順

- 止まってしまった（エラー）場合の処理
 - ① “com_u1_run_event”の[Write]欄に“3”を入力する。
 - ② “Write”ボタンを押す。

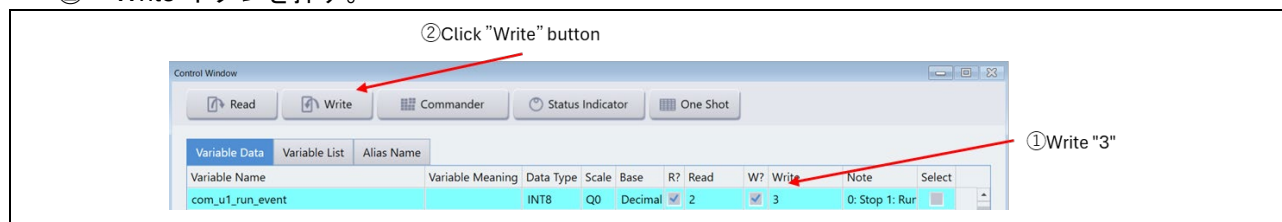


図 4-4 エラー解除の手順

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2025/7/9	－	新規発行
1.01	2026/6/30	－	・ 開発環境の更新 ・ モータパラメータの変更

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くと、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、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。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. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）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。
 8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
 11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
 13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

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。

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/