

RA Family

永久磁石同期モータのホールベクトル制御 — 1 シャント電流検出方式 Motor Flexible Control Kit シリーズ用

要旨

本アプリケーションノートは、RA6T2 マイクロコントローラで永久磁石同期モータを駆動するホールセンサ使用ベクトル制御ソフトウェア及びモータ制御開発支援ツール「Renesas Motor Workbench」の使用方法について説明することを目的としています。

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

動作確認デバイス

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

- RA6T2 (R7FA6T2BD3CFP)

目次

1. 概説	3
1.1 開発環境	3
2. システム概要	4
2.1 ハードウェア構成	4
2.2 ハードウェア設定変更内容	5
2.3 ハードウェア仕様	6
2.3.1 ユーザインタフェース	6
2.3.2 周辺機能	7
2.4 ソフトウェア構成	14
2.4.1 ソフトウェア・ファイル構成	14
2.4.2 モジュール構成	15
2.5 ソフトウェア仕様	16
2.6 割込み優先順位	17
3. 制御ソフトウェア説明	18
3.1 制御内容	18
3.1.1 モータ起動/停止	18
3.1.2 A/D 変換	18
3.1.3 変調	19
3.1.4 状態遷移	21
3.1.5 電気速度・角度推定	22
3.1.5.1 速度推定	22
3.1.5.2 角度推定	22

3.1.6	始動方法	23
3.1.7	システム保護機能	24
3.1.8	1 シャント抵抗電流測定方法	25
3.1.8.1	1 シャント抵抗電流測定タイミング	25
3.1.8.2	RA6T2 の機能を用いた 1 シャント抵抗電流測定方法	26
3.1.8.3	デューティ調整	27
3.1.9	AD トリガ	28
3.2	ホールセンサ使用ベクトル制御ソフトウェア関数仕様	29
3.3	Contents of control	36
3.3.1	Configuration Options	36
3.3.2	Configuration Options for included modules	37
3.4	制御フロー（フローチャート）	45
3.4.1	メイン処理	45
3.4.2	50 [μs]周期割り込み（キャリア同期割り込み）処理	46
3.4.3	500 [μs]周期割り込み処理	47
3.4.4	過電流検出割り込み処理	48
4.	サンプルソフトウェアの操作概要	49
4.1	プロジェクトのインポート	49
4.2	ビルドとデバッグ	50
4.3	クイックスタート	51
4.4	モータ制御開発支援ツール「Renesas Motor Workbench」	52
4.4.1	概要	52
4.4.2	Easy 機能操作例	53
4.4.3	Analyzer 機能用変数一覧	55
4.4.4	Analyzer 機能操作例	56
4.4.5	通信速度の変更例	58
4.4.6	ビルトイン型通信ライブラリの使用方法	59
5.	参考ドキュメント	62
	改訂記録	63

1. 概説

本アプリケーションノートでは、RA6T2 マイクロコントローラでホールセンサを用いて永久磁石同期モータ(PMSM)を駆動するベクトル制御ソフトウェアの実装方法及びモータ開発支援ツール「Renesas Motor Workbench」の使用方法について説明します。

なお、このソフトウェアのベクトル制御に関しては、「永久磁石同期モータのセンサレスベクトル制御 (アルゴリズム編)」(R01AN3786)のアルゴリズムを使用していますので、アルゴリズムの詳細はそちらを参照してください。

1.1 開発環境

サンプルソフトウェアの開発環境を表 1-1、表 1-2 に示します。

表 1-1 ハードウェア開発環境

マイコン	評価ボード 注1	モータ 注2
RA6T2(R7FA6T2BD3CFP)	MCK-RA6T2 (インバータボード、CPU ボード)	R42BLD30L3

表 1-2 ソフトウェア開発環境

e ² studio バージョン	FSP バージョン	ツールチェーン バージョン
V2023-04	V4.4.0	GCC ARM Embedded : V10.3.1.20210824

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

- 【注】
1. インバータボード(RTK0EM0000B12020BJ)、CPU ボード(RTK0EMA270C00000BJ)は、キット製品 MCK-RA6T2(RTK0EMA270S00020BJ)に同梱しており、ルネサスエレクトロニクス株式会社の製品です。
 2. R42BLD30L3 は、MOONS' 社の製品です。
MOONS' 社(<https://www.moonsindustries.com/jp>)

2. システム概要

サンプルソフトウェアの概要を以下に説明します。

2.1 ハードウェア構成

ハードウェア構成を次に示します。

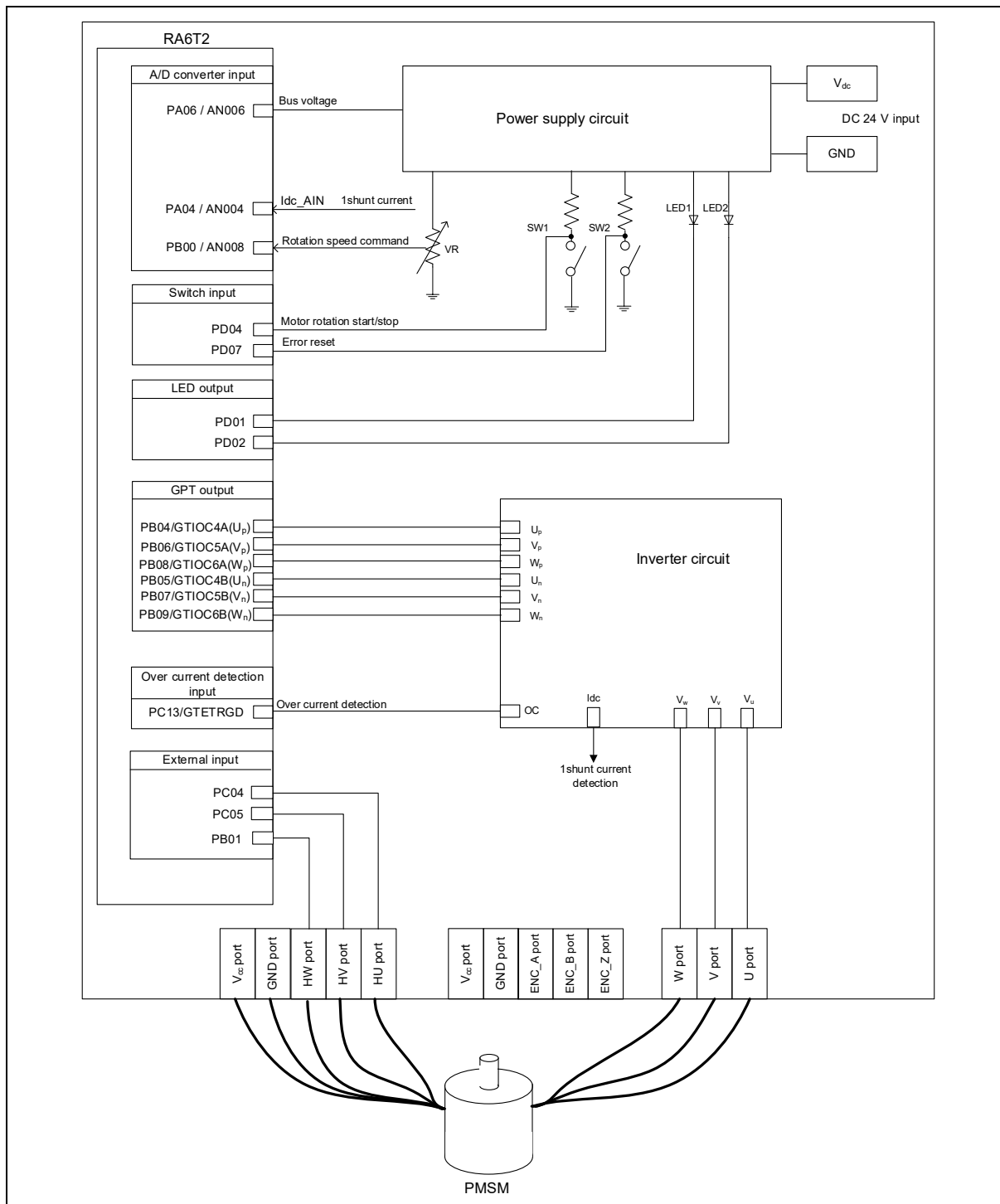


図 2-1 ハードウェア構成図

2.2 ハードウェア設定変更内容

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変更が必要になります。以下に変更内容を示します。

- ジャンパー変更

- (1) ジャンパー(J08)を 1-2 ピン接続から 2-3 ピン接続へ差し換えてください。
- (2) ジャンパー(J11)を 1-2 ピン接続から 2-3 ピン接続へ差し換えてください。

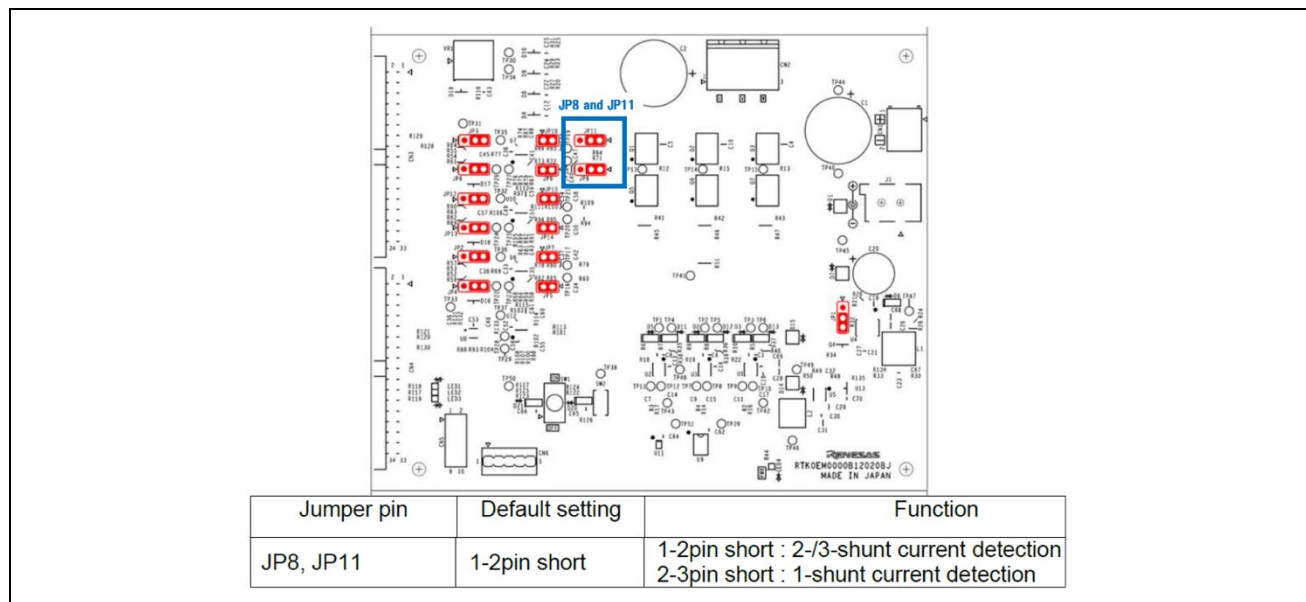


図 2-2 ジャンパー変更

2.3 ハードウェア仕様

2.3.1 ユーザインタフェース

サンプルソフトウェアのユーザインタフェース一覧を表 2-1 に示します。

表 2-1 インバータボードユーザインタフェース

項目	インタフェース部品	機能
回転速度指令	可変抵抗器(VR1)	回転速度指令値入力
START / STOP	トグルスイッチ(SW1)	モータ回転開始／停止指令
ERROR RESET	プッシュスイッチ(SW2)	エラー状態からの復帰指令
LED1	橙色 LED(LED1)	- モータ駆動時 : 点灯 - モータ停止時 : 消灯
LED2	橙色 LED(LED2)	- エラー検出時 : 点灯 - 通常動作時 : 消灯

サンプルソフトウェアの端子インタフェースを表 2-2 に示します。

表 2-2 端子インタフェース

R7FA6T2BD3CFP 端子名	機能
PA06 / AN006	インバータ母線電圧測定
PB00 / AN008	回転速度指令値入力用(VR_1)
PD04	START/STOP トグルスイッチ(SW1)
PD07	ERROR RESET プッシュスイッチ(SW2)
PD01	LED1 点灯/消灯制御
PD02	LED2 点灯/消灯制御
PA04 / AN004	相電流測定
PB04 / GTIOC4A	PWM 出力(U_p)
PB06 / GTIOC5A	PWM 出力(V_p)
PB08 / GTIOC6A	PWM 出力(W_p)
PB05 / GTIOC4B	PWM 出力(U_n)
PB07 / GTIOC5B	PWM 出力(V_n)
PB09 / GTIOC6B	PWM 出力(W_n)
PC04	ホールセンサ入力(HU)
PC05	ホールセンサ入力(HV)
PB01	ホールセンサ入力(HW)
PC13 / GTETRGD	過電流検出時の PWM 緊急停止入力

2.3.2 周辺機能

サンプルソフトウェアで使用する周辺機能一覧を表 2-3 に示します。

表 2-3 周辺機能対応表

周辺機能	リソース	用途
12 ビット A/D コンバータ	AN004, AN006, AN008	- 回転速度指令値入力 1 シャント電流測定 - インバータ母線電圧測定
AGT	AGT0	500 [μs]インターバルタイマ
GPT	CH4, CH5, CH6	相補 PWM 出力
POEG	Group D	PWM 出力端子をハイインピーダンス状態にし、PWM 出力を停止

(1) 12 ビット A/D コンバータ(ADC12)

1 シャント電流、インバータ母線電圧、回転速度指令値を、「シングルスキャンモード」で測定します(ハードウェアトリガを使用)。A/D 変換は、キャリア同期割込みと連動して動作させています。

(2) 低消費電力非同期汎用タイマ (AGT)

500 [us]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 4、5、6 の PWM 出力動作モードを使用して、デッドタイム付きの出力を行います。

(4) GPT 用ポートアウトプットイネーブル (POEG)

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

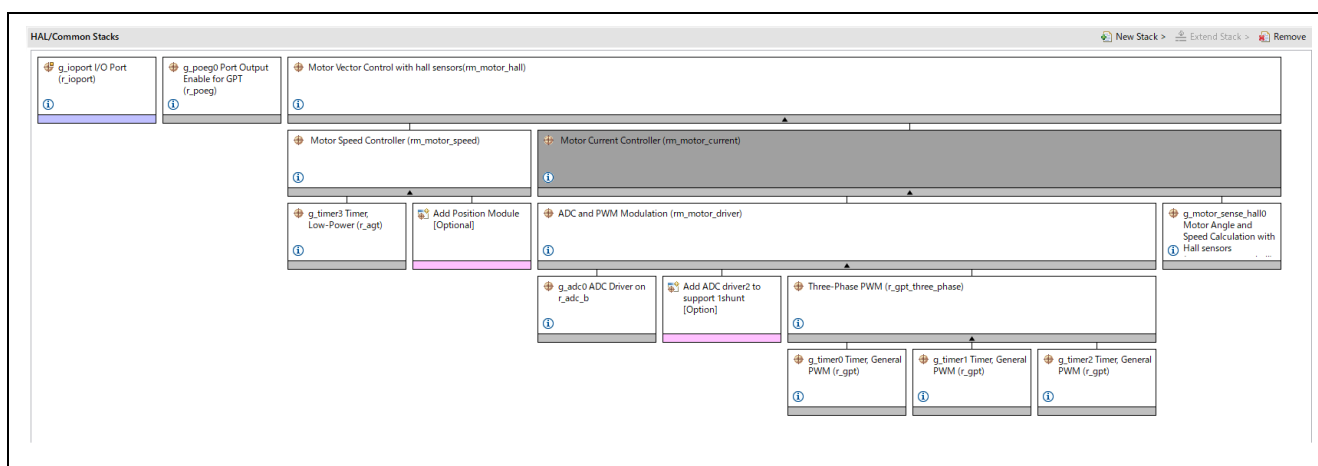


図 2-3 FSP スタック全体図

g_adc0 ADC Driver on r_adc_b		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	▼ Module g_adc0 ADC Driver on r_adc_b	
	▼ General	
	▼ Operation	
	▼ ADC 0	
	Conversion Method	SAR Mode
	Scan Mode	Single Scan
	▼ ADC 1	
	Conversion Method	SAR Mode
	Scan Mode	Single Scan
	> ADC Successive Approximation Time	
	> Synchronous Operation	
	> Calibration	
	> Sampling State Table	
	Name	g_adc0
	> Clock Configuration	
	▼ Interrupts	
	> Limiter Clip Priority	
	> Conversion Error Priority	
	> Overflow Priority	
	> Calibration End Priority	
	▼ Scan End Priority	
	Group 0	Disabled
	Group 1	Disabled
	Group 2	Disabled
	Group 3	Disabled
	Group 4	Disabled
	Group 5 to 8	Disabled
	> FIFO Priorities	
	Callback	NULL
	> Sample and Hold	
	> Programmable Gain Amplifier	
	> User Offset Table	
	> User Gain Table	
	> Limiter Clipping	

図 2-4 ADC ドライバの FSP コンフィグレーション [1/3]

g_adc0 ADC Driver on r_adc_b

Settings	Property	Value
API Info	▼ Virtual Channel 0	
	Scan Group	None
	Channel Select	AN000
	Sampling State Table ID	Sampling State Entry 0
	Channel Gain Table	Disabled
	Channel Offset Table	Disabled
	Add/Average Mode	Disabled
	Add/Average Count	1-time conversion (Normal Conversion)
	Limit Clip Table Id	Disabled
	Conversion Data Format Select	12-bit Data Format
	Digital Filter Selection	Disabled
	▼ Virtual Channel 1	
	Scan Group	None
	Channel Select	AN002
	Sampling State Table ID	Sampling State Entry 0
	Channel Gain Table	Disabled
	Channel Offset Table	Disabled
	Add/Average Mode	Disabled
	Add/Average Count	1-time conversion (Normal Conversion)
	Limit Clip Table Id	Disabled
	Conversion Data Format Select	12-bit Data Format
	Digital Filter Selection	Disabled
	▼ Virtual Channel 2	
	Scan Group	Scan Group 0
	Channel Select	AN004
	Sampling State Table ID	Sampling State Entry 0
	Channel Gain Table	Disabled
	Channel Offset Table	Disabled
	Add/Average Mode	Disabled
	Add/Average Count	1-time conversion (Normal Conversion)
	Limit Clip Table Id	Disabled
	Conversion Data Format Select	12-bit Data Format
	Digital Filter Selection	Disabled
	▼ Virtual Channel 3	
	Scan Group	Scan Group 1
	Channel Select	AN006
	Sampling State Table ID	Sampling State Entry 0
	Channel Gain Table	Disabled
	Channel Offset Table	Disabled
	Add/Average Mode	Disabled
	Add/Average Count	1-time conversion (Normal Conversion)
	Limit Clip Table Id	Disabled
	Conversion Data Format Select	12-bit Data Format
	Digital Filter Selection	Disabled
	▼ Virtual Channel 4	
	Scan Group	Scan Group 1
	Channel Select	AN008
	Sampling State Table ID	Sampling State Entry 0
	Channel Gain Table	Disabled
	Channel Offset Table	Disabled
	Add/Average Mode	Disabled
	Add/Average Count	1-time conversion (Normal Conversion)
	Limit Clip Table Id	Disabled

図 2-5 ADC ドライバの FSP コンフィグレーション [2/3]

g_adc0 ADC Driver on r_adc_b		
Settings	Property	Value
API Info	▼ Scan Group 0	
	> Self Diagnosis	
	> External Trigger Enable	
	> ELC Trigger Enable	
	▼ GPT Trigger Enable	
	GPT Channel 0 Request A	☐
	GPT Channel 1 Request A	☐
	GPT Channel 2 Request A	☐
	GPT Channel 3 Request A	☐
	GPT Channel 4 Request A	☒
	GPT Channel 5 Request A	☐
	GPT Channel 6 Request A	☐
	GPT Channel 7 Request A	☐
	GPT Channel 8 Request A	☐
	GPT Channel 9 Request A	☐
	GPT Channel 0 Request B	☐
	GPT Channel 1 Request B	☐
	GPT Channel 2 Request B	☐
	GPT Channel 3 Request B	☐
	GPT Channel 4 Request B	☒
	GPT Channel 5 Request B	☐
	GPT Channel 6 Request B	☐
	GPT Channel 7 Request B	☐
	GPT Channel 8 Request B	☐
	GPT Channel 9 Request B	☐
	Enable	Enable
	Converter Selection	ADC 0
	Start Trigger Delay	0
	Scan End Interrupt Enable	Disable
	Limit Clip Interrupt Enable	Disable
	FIFO Enable	Enable
	FIFO Interrupt Enable	Disable
	FIFO Interrupt Generation Level	0
g_adc0 ADC Driver on r_adc_b		
Settings	Property	Value
API Info	▼ Scan Group 1	
	> Self Diagnosis	
	> External Trigger Enable	
	> ELC Trigger Enable	
	▼ GPT Trigger Enable	
	GPT Channel 0 Request A	☐
	GPT Channel 1 Request A	☐
	GPT Channel 2 Request A	☐
	GPT Channel 3 Request A	☐
	GPT Channel 4 Request A	☐
	GPT Channel 5 Request A	☒
	GPT Channel 6 Request A	☐
	GPT Channel 7 Request A	☐
	GPT Channel 8 Request A	☐
	GPT Channel 9 Request A	☐
	GPT Channel 0 Request B	☐
	GPT Channel 1 Request B	☐
	GPT Channel 2 Request B	☐
	GPT Channel 3 Request B	☐
	GPT Channel 4 Request B	☐
	GPT Channel 5 Request B	☐
	GPT Channel 6 Request B	☐
	GPT Channel 7 Request B	☐
	GPT Channel 8 Request B	☐
	GPT Channel 9 Request B	☐
	Enable	Enable
	Converter Selection	ADC 1
	Start Trigger Delay	0
	Scan End Interrupt Enable	Disable
	Limit Clip Interrupt Enable	Disable
	FIFO Enable	Disable
	FIFO Interrupt Enable	Disable
	FIFO Interrupt Generation Level	0

図 2-6 ADC ドライバの FSP コンフィグレーション[3/3]

g_timer3 Timer, Low-Power (r_agt)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	Pin Output Support	Disabled
	Pin Input Support	Disabled
	▼ Module g_timer3 Timer, Low-Power (r_agt)	
	▼ General	
	Name	g_timer3
	Channel	0
	Mode	🔒 Periodic
	Period	30000
	Period Unit	Raw Counts
	Count Source	PCLKB
	> Output	
	> Input	
	▼ Interrupts	
	Callback	🔒 rm_motor_speed_cyclic
	Underflow Interrupt Priority	Priority 10

図 2-7 AGT ドライバの FSP コンフィグレーション

g_timer0 Timer, General PWM (r_gpt)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	Pin Output Support	Enabled with Extra Features
	Write Protect Enable	Disabled
	Clock Source	PCLKD
	▼ Module g_timer0 Timer, General PWM (r_gpt)	
	▼ General	
	Name	g_timer0
	Channel	4
	Mode	Triangle-Wave Asymmetric PWM (Mode 3)
	Period	50
	Period Unit	Microseconds
	▼ Output	
	> Custom Waveform	
	Duty Cycle Percent (only applicable in PWM mode)	50
	GTIOCA Output Enabled	True
	GTIOCA Stop Level	Pin Level Low
	GTIOCB Output Enabled	True
	GTIOCB Stop Level	Pin Level High
	> Input	
	> Interrupts	
	▼ Extra Features	
	▼ Output Disable	
	> Output Disable POEG Trigger	
	POEG Link	POEG Channel 3
	GTIOCA Disable Setting	Set Hi Z
	GTIOCB Disable Setting	Set Hi Z
	▼ ADC Trigger	
	▼ Start Event Trigger (Channels with GTINTAD only)	
	Trigger Event A/D Converter Start Request A Durir	☒
	Trigger Event A/D Converter Start Request A Durir	☐
	Trigger Event A/D Converter Start Request B Durir	☒
	Trigger Event A/D Converter Start Request B Durir	☐
	▼ Dead Time	
	Dead Time Count Up (Raw Counts)	240
	Dead Time Count Down (Raw Counts) (Channels with	240
	▼ ADC Trigger (Channels with GTADTRA only)	
	ADC A Compare Match (Raw Counts)	10
	▼ ADC Trigger (Channels with GTADTRB only)	
	ADC B Compare Match (Raw Counts)	80
	> Interrupt Skipping (Channels with GTITC only)	
	Extra Features	Enabled

図 2-8 GPT ドライバの FSP コンフィグレーション

g_poeg0 Port Output Enable for GPT (r_poeg)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	▼ Module g_poeg0 Port Output Enable for GPT (r_poeg)	
	▼ General	
	▼ Trigger	
	GTETRG Pin	☒
	GPT Output Level	☐
	Oscillation Stop	☐
	ACMPHS0	☐
	ACMPHS1	☐
	ACMPHS2	☐
	ACMPHS3	☐
	Name	g_poeg0
	Channel	3
	▼ Input	
	GTETRG Polarity	Active Low
	GTETRG Noise Filter	PCLKB/32
	▼ Interrupts	
	Callback	g_poe_overcurrent
	Interrupt Priority	Priority 0 (highest)

図 2-9 POEG ドライバの FSP コンフィグレーション

2.4 ソフトウェア構成

2.4.1 ソフトウェア・ファイル構成

ソフトウェアのフォルダとファイル構成を下記に示します。

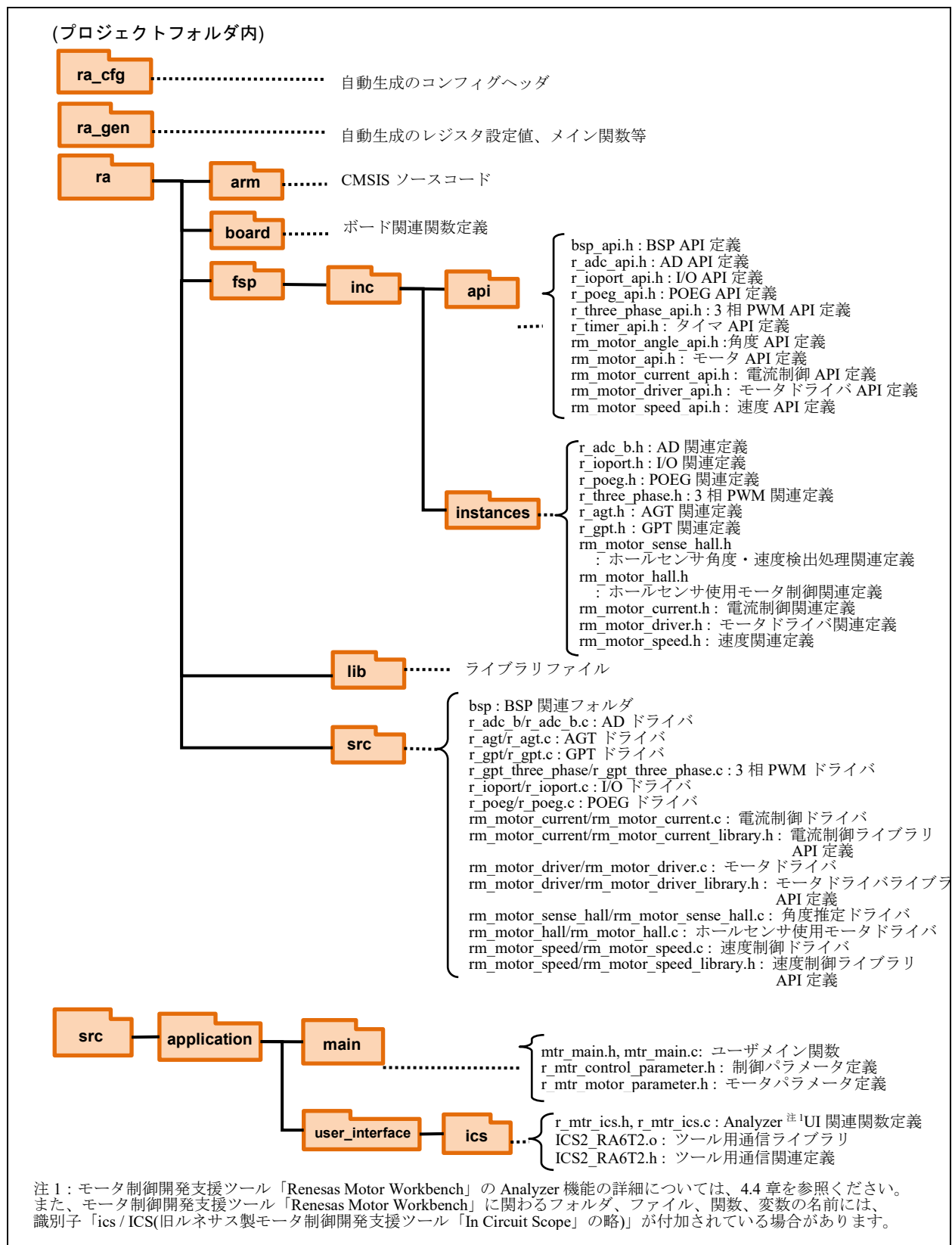


図 2-10 フォルダ・ファイル構成

2.4.2 モジュール構成

ソフトウェアのモジュール構成を図 2-11 に示します。

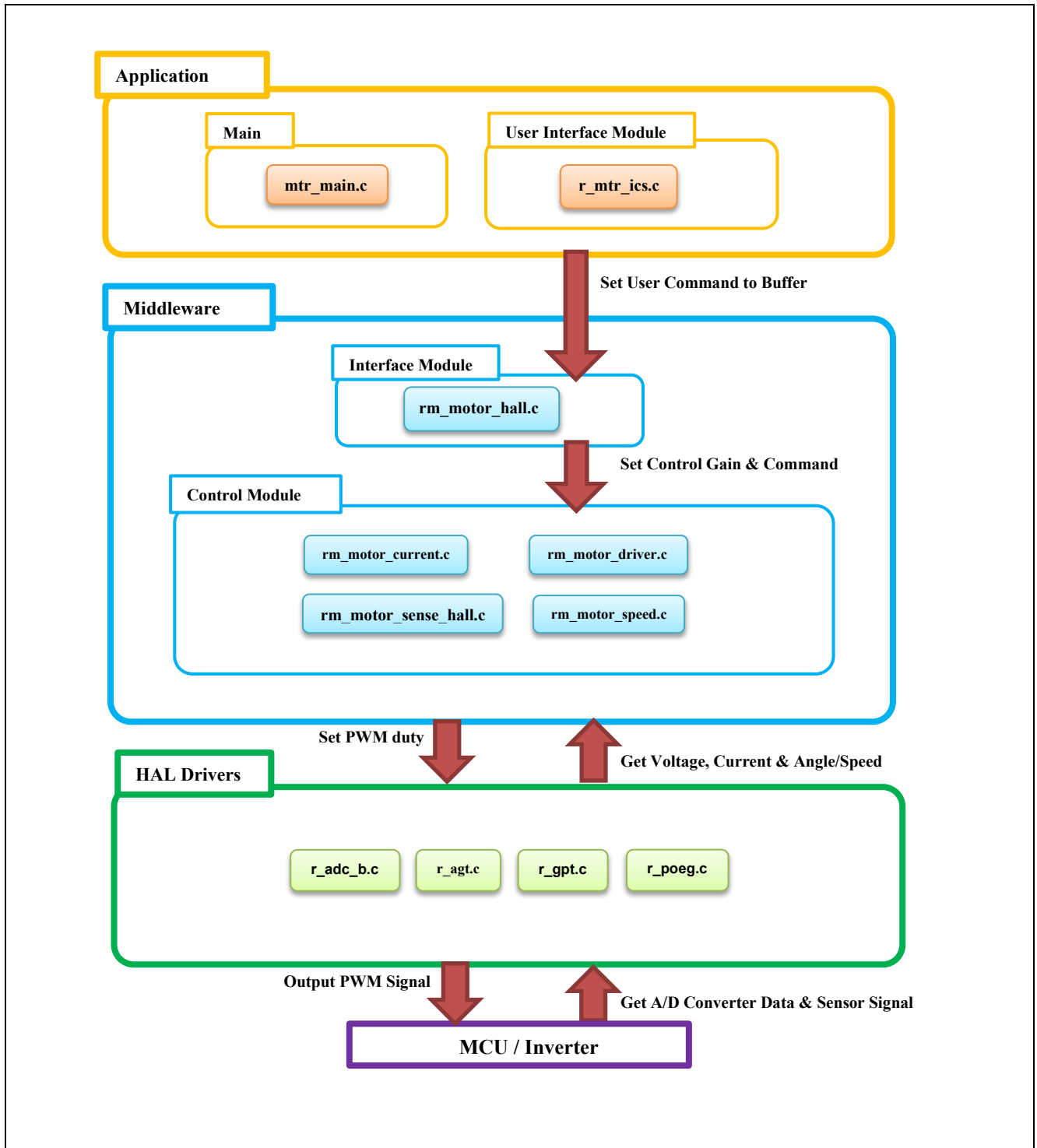


図 2-11 モジュール構成

2.5 ソフトウェア仕様

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

表 2-4 ホールセンサ使用ベクトル制御ソフトウェア基本仕様

項目	内容
制御方式	ベクトル制御
回転子磁極位置検出	ホールセンサ
モータ回転開始/停止	SW1 のレベルにより判定 または Renesas Motor Workbench から入力
入力電圧	DC 24V
メインクロック周波数	240 [MHz]
キャリア(PWM)周波数	20 [kHz](キャリア周期 : 50 [μs])
デッドタイム	2 [μs]
制御周期	電流制御 / 位置・速度推定 : 50 [μs] 速度制御 : 500 [μs]
回転速度範囲	CW : 0 [rpm] ~ 2400 [rpm] CCW : 0 [rpm] ~ 2400 [rpm]
各制御系固有周波数	電流制御系 : 300 [Hz] 速度制御系 : 5 [Hz] 誘起電圧推定系 : 1000 [Hz] 位置推定系 : 50 [Hz]
コンパイラ最適化設定	最適化レベル Optimize more(-O2) (デフォルト設定)
ROM/RAM サイズ	ROM : 32.6KB RAM : 3.7KB
保護停止処理	以下のいずれかの条件の時、モータ制御信号出力(6 本)を非アクティブにする (1) 各相の電流が $3.54(=1.67 \times \sqrt{2} \times 1.5)$ [A] を超過(50 [μs]毎に監視) (2) インバータ母線電圧が 60 [V] を超過(50 [μs]毎に監視) (3) インバータ母線電圧が 8[V]未満(50 [μs]毎に監視) (4) 回転速度が 4500 [rpm] を超過(50 [μs]毎に監視) 外部からの過電流検出信号(GTETRGD 端子に Low レベルを検出)を検出した場合、PWM 出力端子をハイインピーダンスにする

2.6 割り込み優先順位

本システムで使用している割り込みと優先順位を下記に示します。

表 2-5 割り込み優先順位

割り込み レベル	優先度	処理
15	Min Max	
14		
13		
12		
11		
10		AGT0 INT 500[us]割り込み処理
9		
8		
7		
6		
5		ADC0 ADI0 A/D 変換完了割り込み
4		
3		
2		
1		
0		POEG3 EVENT 過電流異常割り込み

Allocations		
Interrupt	Event	ISR
0	POEG3 EVENT (Port Output disable interrupt D)	poeg_event_isr
1	AGT0 INT (AGT interrupt)	agt_int_isr
2	GPT4 COUNTER OVERFLOW (Overflow)	gpt_counter_overflow_isr

図 2-12 FSP 割り込みコンフィグレーション

3. 制御ソフトウェア説明

サンプルソフトウェアについて説明します。

3.1 制御内容

3.1.1 モータ起動/停止

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

SW1 には汎用ポートが割り当てられ、“Low”レベルのときスタートスイッチが押されていると判断し、“High”レベルのときはモータを停止すると判断します。

3.1.2 A/D 変換

(1) モータ回転速度指令値

モータの回転速度指令値は Renesas Motor Workbench からの入力または VR1 の出力値(アナログ値)を A/D 変換することによって決定します。A/D 変換された VR の値は、以下の表のように、回転速度指令値として使用します。

表 3-1 回転速度指令値の変換比

項目	変換比 (指令値 : A/D 変換値)		チャンネル
回転速度指令値	CW	0 [rpm]~2400[rpm] : 0800H~0FFFH	AN008
	CCW	0 [rpm]~2400[rpm] : 07FFH~0000H	

(2) インバータ母線電圧

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

表 3-2 インバータ母線電圧の変換比

項目	変換比 (インバータ母線電圧 : A/D 変換値)	チャンネル
インバータ母線電圧	0 [V]~73.26 [V] : 0000H~0FFFH	AN006

(3) 1 シャント電流

以下の表のように、1 シャント電流を測定し、ベクトル制御に使用します。

表 3-3 1 シャント電流の変換比

項目	変換比 (1 シャント電流 : A/D 変換値)	チャンネル
1 シャント電流	-8.25 [A]~8.25 [A] : 0000H~0FFFH ^注 電流値=(3.3V-1.65V)÷(0.01Ohm×20)=8.25A	AN004

【注】A/D 変換特性の詳細に関しては、「RA6T2 グループ ユーザーズマニュアル ハードウェア編」を参照してください。

3.1.3 変調

本アプリケーションノート対象ソフトウェアでは、モータへの入力電圧はパルス幅変調（以降、PWM）によって生成し、PWM 波形は三角波比較法によって生成します。

- 三角波比較法

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

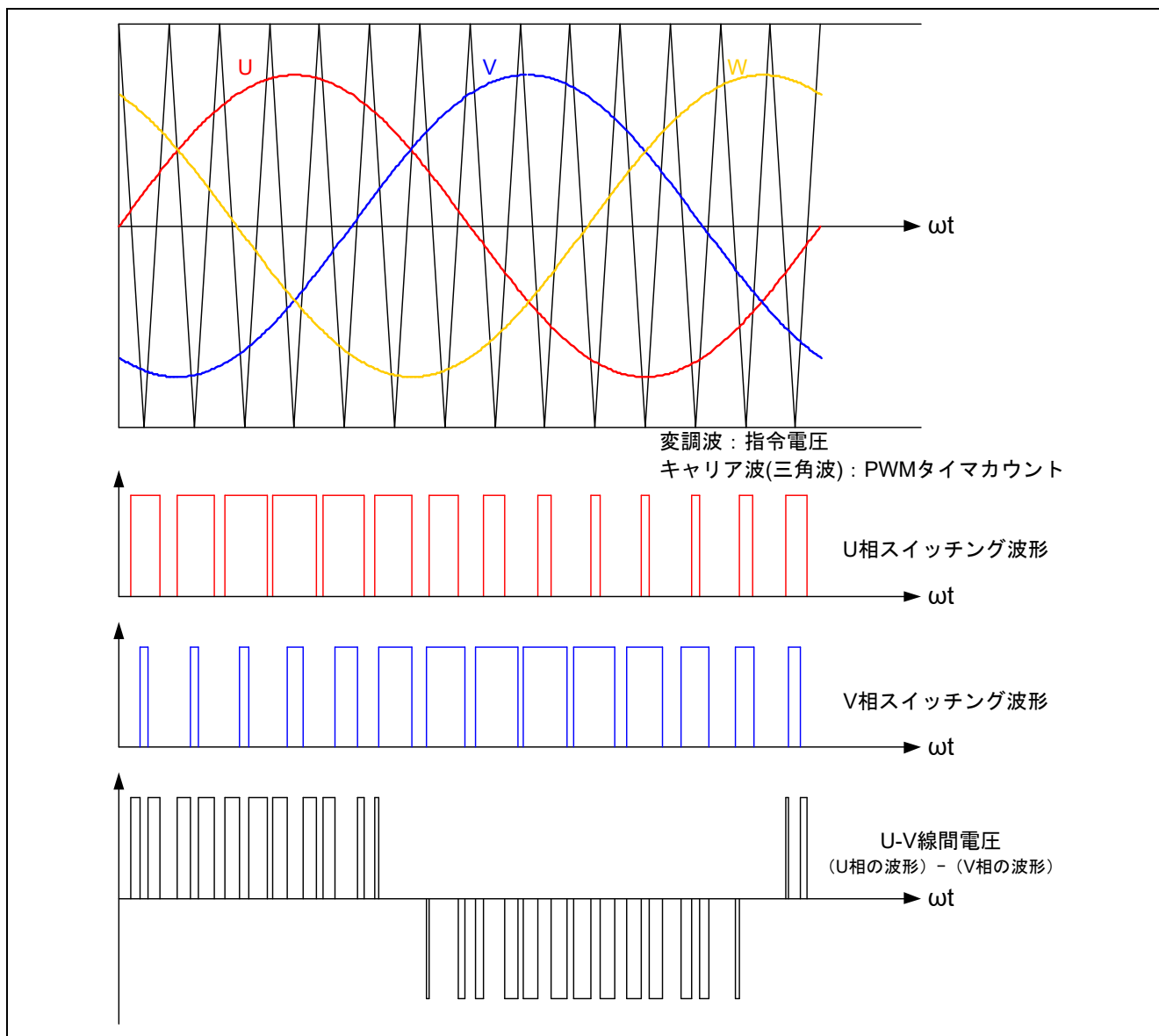


図 3-1 三角波比較法の概念図

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

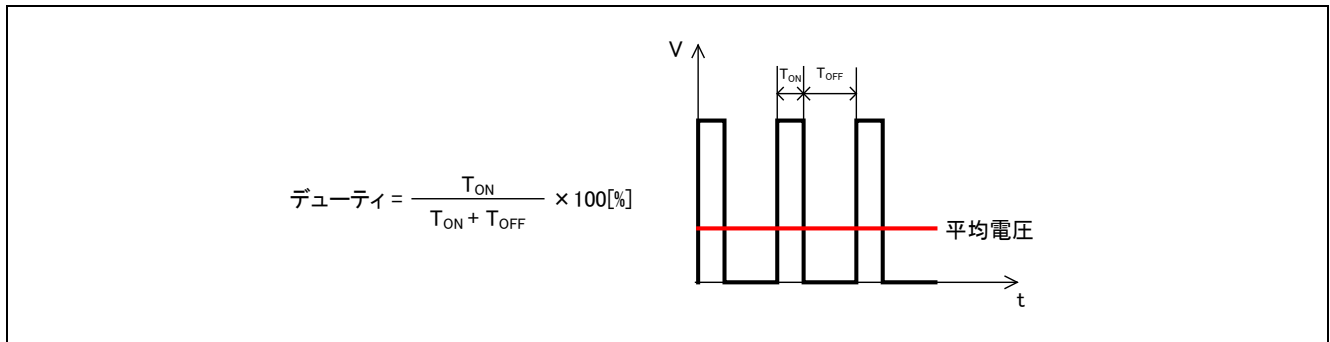


図 3-2 デューティの定義

また、変調率 m を以下のように定義します。

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

m : 変調率 V : 指令値電圧 E : インバータ母線電圧

この変調率を、PWM デューティを決めるレジスタに反映させることで所望の制御を行います。

3.1.4 状態遷移

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

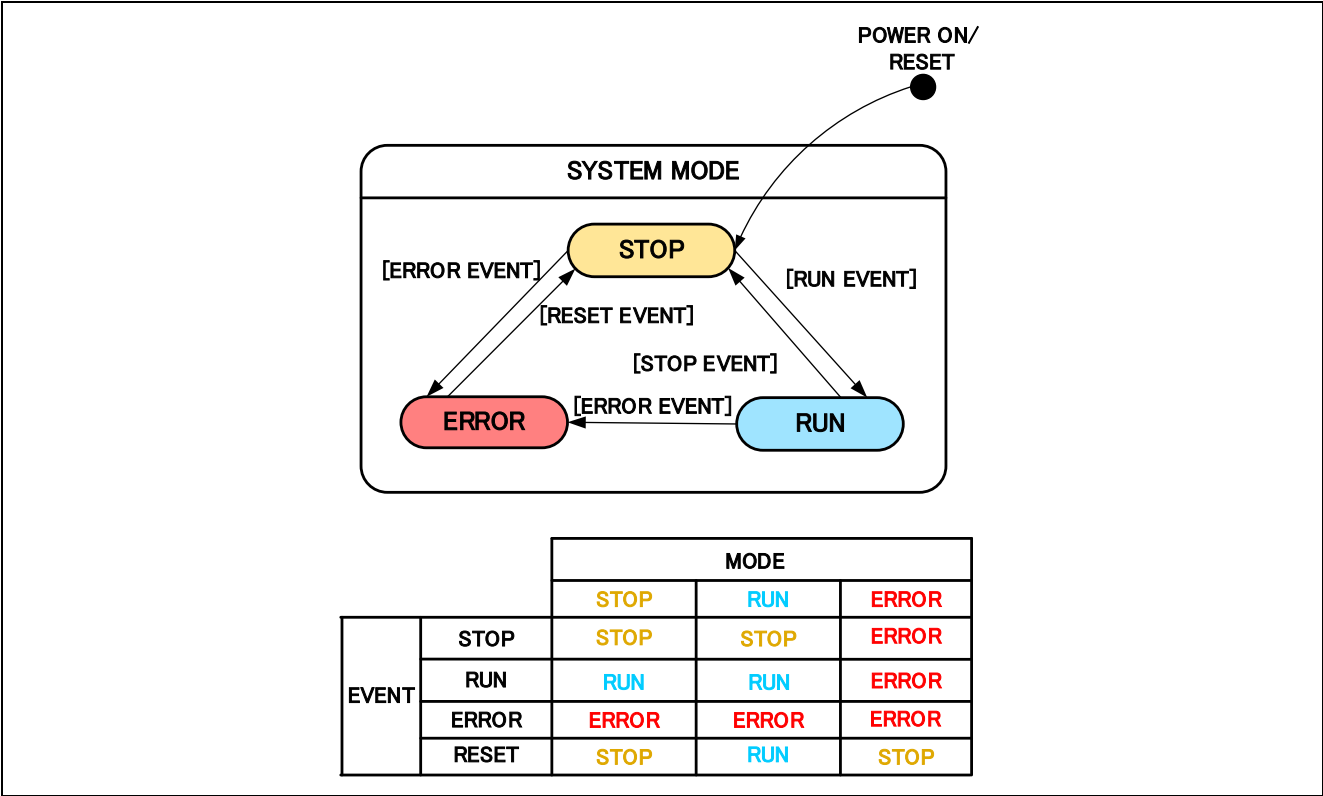


図 3-3 ホールセンサ使用ベクトル制御ソフトウェアの状態遷移図

- (1) SYSTEM MODE
システム動作状態を表します。各イベント(EVENT)の発生により、状態が遷移します。システムの動作状態は、モータ駆動停止 (INACTIVE)、モータ駆動 (ACTIVE)、異常状態 (ERROR) があります。
- (2) EVENT
各 SYSTEM MODE 中に EVENT が発生すると、その EVENT に従って、システム動作状態が図 3-3 中の表の様に遷移します。各 EVENT の発生要因は下記となります。

表 3-4 EVENT 一覧

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

3.1.5 電気速度・角度推定

3.1.5.1 速度推定

回転速度の推定は以下のアルゴリズムによって行います。

ホールセンサ信号の変化をキャリア割り込みで確認し、ホール信号の変化の間に何回のキャリア割り込みが発生したかをカウントすることで電気角 60 度の期間計測を行います。

$$\text{電気角 60 度期間} = \text{キャリア割り込み回数} \times \text{キャリア割り込み期間}(50\mu\text{sec})$$

これにより電気角速度を得ることが出来ます。

$$\text{電気角速度}[\text{rad/sec}] = (2\pi \times 60 / 360) \div \text{電気角 60 度期間}[\mu\text{sec}]$$

但し、ホール信号変化 1 区間での推定だと信号誤差による影響が大きいと考えられるため内部処理としては、直前の信号変化 6 回分(電気角 1 周)の値を用いて演算を行っています。

$$\text{電気角速度}[\text{rad/sec}] = 2\pi \div \text{電気角 360 度期間}(60 \text{ 度期間 } 6 \text{ 回分})[\mu\text{sec}]$$

3.1.5.2 角度推定

以下の情報から電気角を推定します。

- A) 回転方向
- B) 推定速度

回転方向はホールセンサ信号パターンにより検出します。ホールセンサ信号のパターンは使用するモータにより決まっているので、信号が変化した際にその直前のホールセンサ信号と今回のホールセンサ信号を比較することで回転方向を検出することが可能となります。

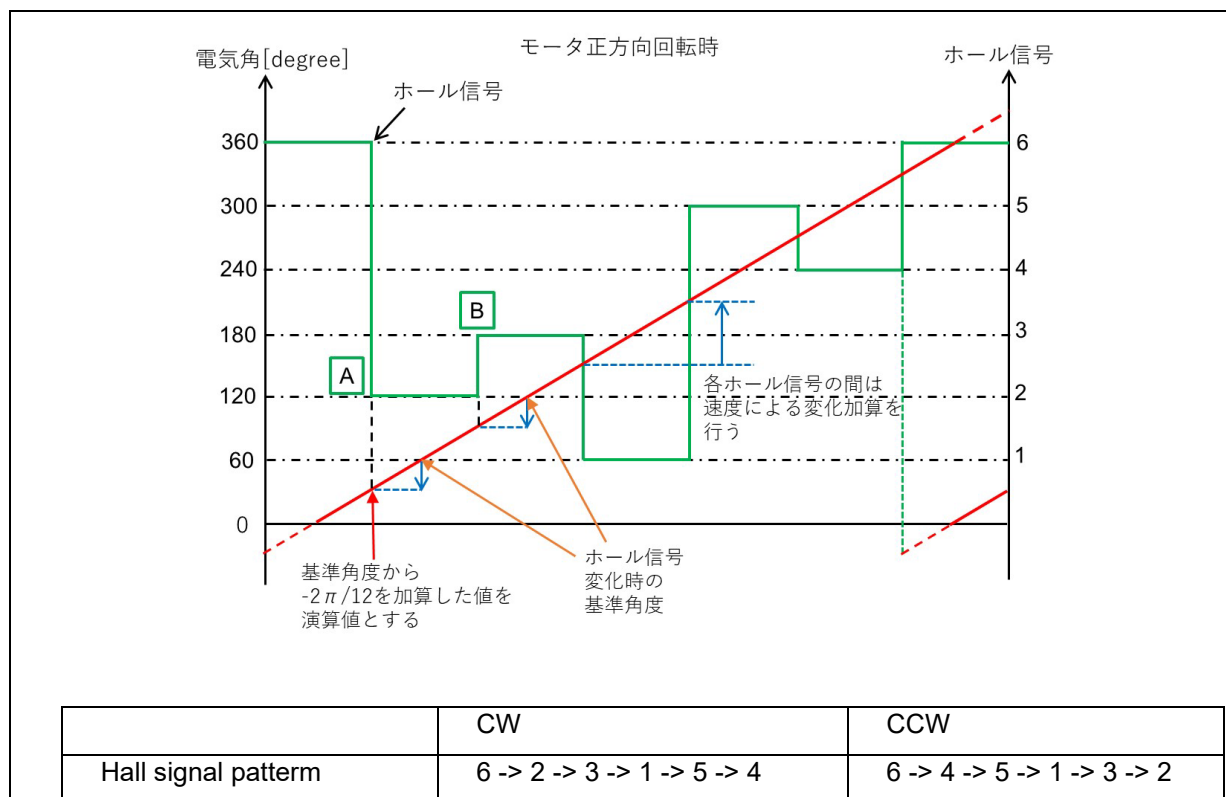


図 3-4 電気角度推定（正方向回転時）

図 3-4 A 点ではホール信号が 6→2 と変化しています。これからモータの回転方向を正方向と判断出来ます。ここで電気角を以下の様に設定します。

$$\text{電気角[rad]} = 2\pi \times 60/360 + \text{ホール信号内角度[rad]} + \text{オフセット値[rad]}$$

ホール信号内角度はホール信号変化点において正方向回転であれば $-2\pi/12$ 、逆方向回転であれば $2\pi/12$ を設定します。そして、キャリア割り込みの発生時にその時の推定速度情報に基づいて差分値を正方向であれば加算、逆方向であれば減算していきます。但し、信号誤差や速度変動を考慮して $-2\pi/12 \sim 2\pi/12$ の範囲の値としています。

- 正方向回転時

$$\text{ホール信号内角度[rad]} = \text{初期値}(-2\pi/12) + \text{推定速度[rad/sec]} \times \text{キャリア割り込み周期}(50\mu\text{sec}) \times \text{キャリア割り込み回数}$$

- 逆方向回転時

$$\text{ホール信号内角度[rad]} = \text{初期値}(2\pi/12) - \text{推定速度[rad/sec]} \times \text{キャリア割り込み周期}(50\mu\text{sec}) \times \text{キャリア割り込み回数}$$

いずれも $-2\pi/12 \sim 2\pi/12$ の範囲を超えた場合は丸め込みます。図 3-4 B 点では電気角を以下の様に設定します。

$$\text{電気角[rad]} = 2\pi \times (60 \times 2)/360 + \text{ホール信号内角度[rad]} + \text{オフセット値[rad]}$$

つまり、各ホール信号の変化点において 60 度の基準角度をもって電気角を推定します。

オフセット値はモータ固有の調整値で主にホール信号誤差（センサ取り付け誤差）等を吸収する為に設定します。

3.1.6 始動方法

本アプリケーションノート対象ソフトウェアの始動制御内容を図 3-5 に示します。d 軸電流、q 軸電流、速度それぞれの指令値を管理するフラグによってモードをコントロールしています。

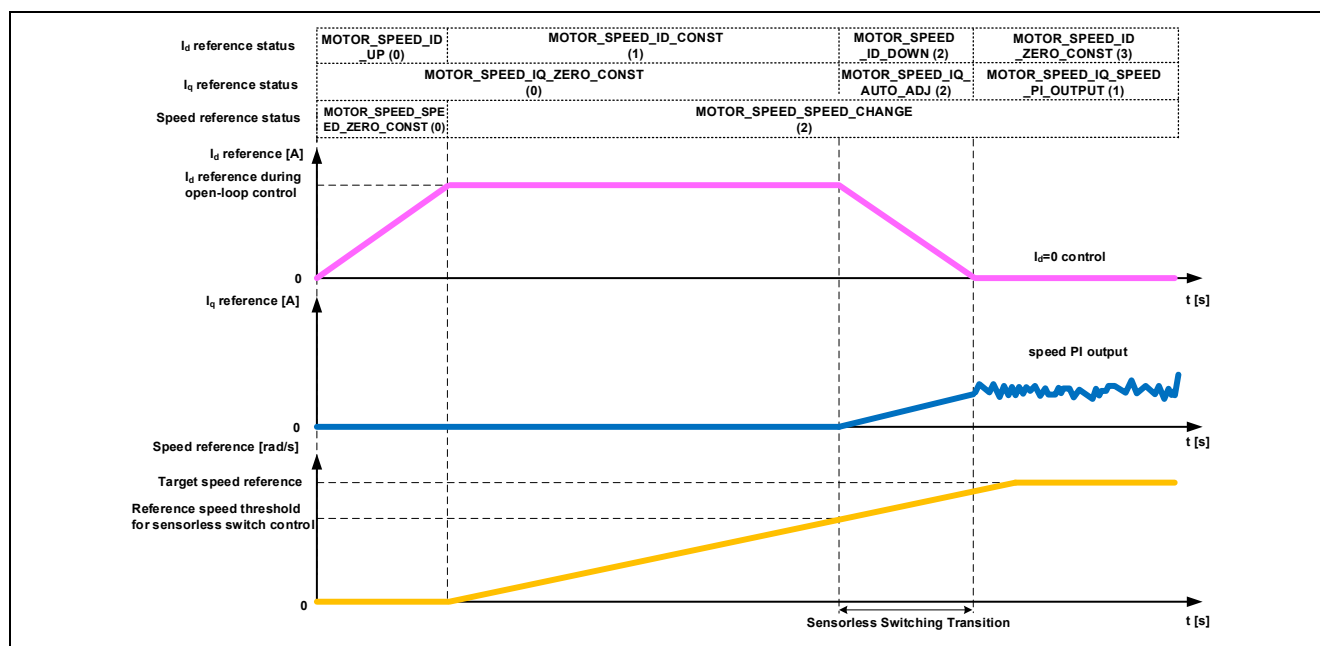


図 3-5 ホールセンサ使用ベクトル制御ソフトウェアの始動制御内容

3.1.7 システム保護機能

サンプルソフトウェアは、以下のエラー状態を持ち、それぞれの場合に緊急停止する機能を実装しています。システム保護機能に関わる各設定値は表 3-5 を参照してください。

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

表 3-5 各システム保護機能設定値

エラー	閾値	
過電流エラー	過電流リミット値 [A]	3.54
	監視周期 [μs]	50
過電圧エラー	過電圧リミット値 [V]	60
	監視周期 [μs]	50
低電圧エラー	低電圧リミット値 [V]	8
	監視周期 [μs]	50
回転速度エラー	速度リミット値 [rpm]	4500
	監視周期 [μs]	50

3.1.8 1 シャント抵抗電流測定方法

サンプルソフトウェアで用いる 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。

3.1.8.1 1 シャント抵抗電流測定タイミング

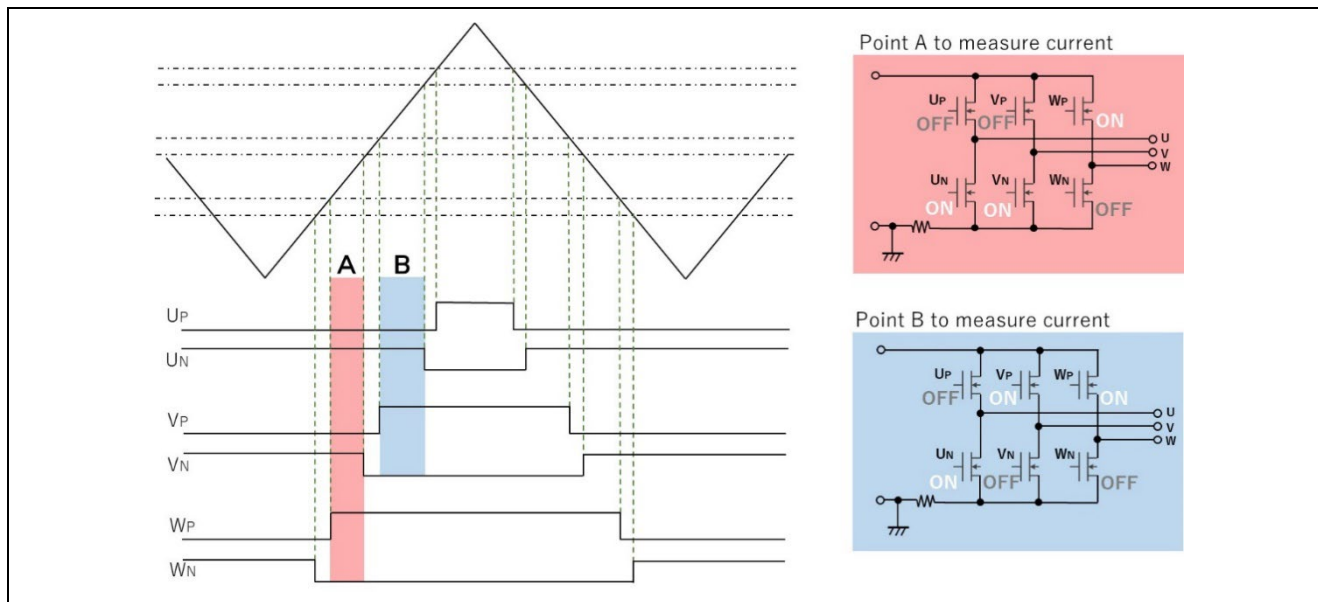


図 3-6 相補 PWM 波形図 (例: デューティ大小関係 $W > V > U$)

サンプルソフトウェアでは GPT ユニットを用いて、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 3-6 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 (I_{dc}) と W 相電流 (I_w) は $I_{dc} = I_w$ という関係になります。

図のポイント B では下アームの U 相のみが ON している状態となります。この時、 I_{dc} と U 相電流 (I_u) は $I_{dc} = -I_u$ という関係になります。残りの 1 相である V 相電流 (I_v) はキルヒホッフの第一法則を用いて $I_v = -I_u - I_w$ と復元できます。

従って、図のポイント 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、三相の電流を得ることが出来ます。

上記の例はデューティの大小関係が $W > V > U$ となる場合です。PWM の出力変化に伴いこの大小関係は 6 パターンの組み合わせが発生することとなります。ポイント A、B で検出出来る電流値はパターンに従って相との関係が変化しますので、それに基づいた三相への割り当てが必要となります。デューティを設定する時点でこの大小関係は分かっているので、それに基づいて検出した電流値の各相への割り当てを切り換えることが可能となっています。

表 3-6 デューティパターンと相電流

Duty pattern	Point A	Point B
$W > V > U$	I_w	$-I_u$
$W > U > V$	I_w	$-I_v$
$V > W > U$	I_v	$-I_u$
$V > U > W$	I_v	$-I_w$
$U > W > V$	I_u	$-I_v$
$U > V > W$	I_u	$-I_w$

3.1.8.2 RA6T2 の機能を用いた 1 シャント抵抗電流測定方法

3.1.8.1 に示した様に 1 シャント抵抗におけ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制御する必要が有ります。サンプルソフトウェアでは RA6T2 GPT モジュールの GTADTRA、及び GTADTRB レジスタと GTCNT カウンタとのコンペアマッチによる A/D 変換開始要求機能によりこれを実現しています。

g_timer0 Timer, General PWM (r_gpt)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	Pin Output Support	Enabled with Extra Features
	Write Protect Enable	Disabled
	Clock Source	PCLKD
	▼ Module g_timer0 Timer, General PWM (r_gpt)	
	> General	
	> Output	
	> Input	
	> Interrupts	
	▼ Extra Features	
	> Output Disable	
	▼ ADC Trigger	
	▼ Start Event Trigger (GPTE/GPTEH only)	
	Trigger Event A/D Converter Start Request A Durin	☒
	Trigger Event A/D Converter Start Request A Durin	☐
	Trigger Event A/D Converter Start Request B Durin	☒
	Trigger Event A/D Converter Start Request B Durin	☐
	> Dead Time	

図 3-7 GPT ADC トリガ設定

3.1.8.3 デューティ調整

3.1.8.1 に示した様なタイミングが確保出来る場合は 1 シャント抵抗による電流検出を行うことが出来ますが、動作中の PWM デューティの設定条件によっては A/D 変換に十分な時間が確保出来ない為、正しく電流値を取得することが出来ません。タイミングが確保出来ない条件に関して以下の 2 つの対応を実装してあります。

(1) 2 つの相の切り替えタイミングが近接している場合

2 つの相の切り換えタイミングが近接していて A/D 変換の為の時間が確保出来ない場合に、PWM のデューティは変更せず、後から切り換わる相の切り換えタイミングを A/D 変換に必要な時間分後ろへずらすことで変換時間を確保します。

(2) タイミング遅延が不可能な場合

上記の様に PWM の切り換えタイミングを遅延させた場合にデューティが広く、PWM のキャリア周期の最後に到達してしまう様な場合はタイミングを遅延させることが出来ません。この様な場合は変調率が 1 に近い場合となるので、PWM の切り換えタイミングをキャリア周期の最後になる様に変調率に制限をかけるものとしています。

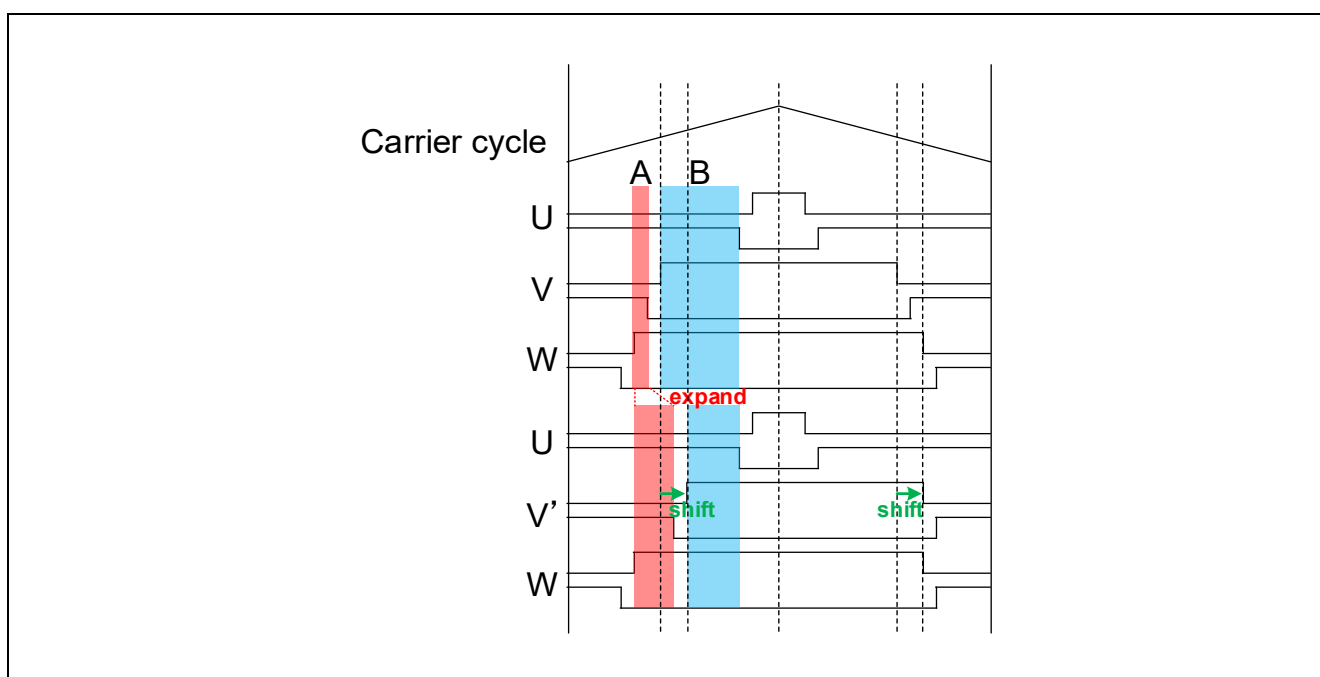


図 3-8 デューティ調整

3.2 ホールセンサ使用ベクトル制御ソフトウェア関数仕様

サンプルソフトウェアにおける制御処理は、主に 50[μs]周期(PWM キャリア周期)割り込みと、500[μs]周期割り込みの2つの割り込みにより構成されています。図 3-10 にあるように、赤破線部が 50[μs]周期毎に実行される処理で、青破線部が 500[μs]周期毎に実行される処理になります。

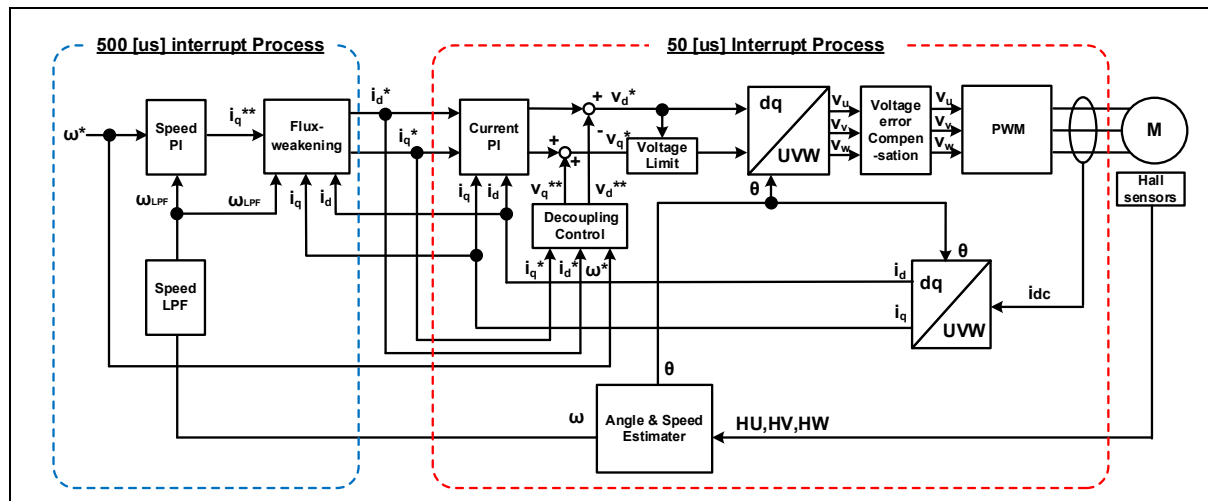


図 3-10 ホールセンサ使用ベクトル制御概略ブロック図

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

表 3-7 50[μs]周期割り込み関数内実行関数一覧(1/4)

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

表 3-8 50[μs]周期割り込み関数内実行関数一覧(2/4)

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

表 3-9 50[μs]周期割り込み関数内実行関数一覧(3/4)

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

表 3-10 50[μs]周期割り込み関数内実行関数一覧(4/4)

ファイル名	関数名	処理概要
rm_motor_current.c	motor_current_transform_dq_uvuw_abs 入力: (const float) f_angle / 回転角度 (const float *) f_dq / dq 軸ポインタ (float *) f_uvuw / UVW 相ポインタ 出力: なし	dq → UVW 座標変換(絶対変換)
librm_motor_current.a	rm_motor_voltage_error_compensation_main 入力: (motor_currnt_voltage_compensation_t *) st_volt_comp / 電圧誤差補償データ (float *) p_f4_v_array / リファレンス電圧 (float *) p_f4_i_array / リファレンス電流 (float) f4_vdc / 母線電圧 出力: なし	電圧誤差補償
rm_motor_sense_hall.c	RM_MOTOR_SENSE_HALL_FlagPiCtrlSet 入力: (motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (uint32_t const) flag_pi / PI 制御実行中フラグ 出力: fsp_err_t / 関数実行結果	PI 制御実行のフラグを設定
	RM_MOTOR_SENSE_HALL_AngleSpeedGet 入力: (motor_angle_ctrl_t * const) p_ctrl / 角度制御構造体ポインタ (float * const) p_angle / 角度データ (float * const) p_speed / 速度データ (float * const) p_phase_err / 位相誤差データ 出力: fsp_err_t / 関数実行結果	ロータの角度と回転速度を取得 (位相誤差データは取得出来ません)
r_gpt_three_phase.c	R_GPT_THREE_PHASE_DutyCycleSet 入力: (three_phase_ctrl_t * const) p_ctrl / 三相タイマ制御構造体ポインタ (three_phase_duty_cycle_t * const) p_duty_cycle / 設定デューティサイクル 出力: fsp_err_t / 関数実行結果	各タイマのデューティサイクル設定

表 3-11 500[μs]周期割り込み関数内実行関数一覧(1/2)

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

表 3-12 500[μs]周期割り込み関数内実行関数一覧(2/2)

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

3.3 Contents of control

3.3.1 Configuration Options

ホールセンサ使用ベクトル制御モジュールの構成オプションは、RA Configurator を使用して構成できます。変更されたオプションは、コードの生成時に `common_data.c/h`、及び `hal_data.c/h` に自動的に反映されます。オプション名と設定値を、表 3-13 に記載します。

表 3-13 構成オプション

Configuration Options (rm_motor_hall.h)	
オプション名	内容
Limit of over current (A) 初期値: 1.67	相電流がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over voltage (V) 初期値: 60.0	母線電圧がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over speed (rpm) 初期値: 4500.0	回転速度がこの値を超えると、PWM 出力ポートがオフに設定されます。
Limit of over speed (rpm) 初期値: 8.0	母線電圧がこの値を下回ると、PWM 出力ポートがオフに設定されます。

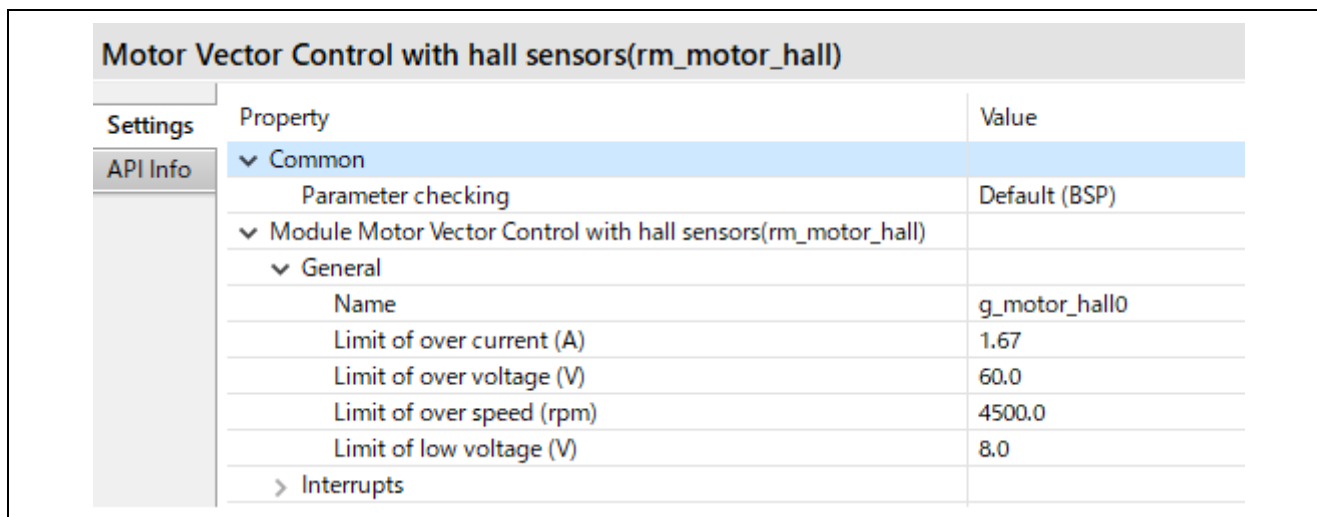


図 3-11 ホールセンサ使用ベクトル制御の FSP コンフィグレーション

3.3.2 Configuration Options for included modules

ホールセンサ使用ベクトル制御モジュールには、以下のモジュールが含まれます。

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

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

表 3-14 モータ電流コントローラ構成オプション

Configuration Options (rm_motor_current.h)	
オプション名	内容
Voltage error compensation 初期値: Enable	電圧誤差補償の有効／無効を選択します。
Shunt type 初期値: 1shunt	シャント抵抗をいくつ用いて電流検出を行うかを選択します。
Motor Parameter Pole pairs 初期値: 4	極対数
Motor Parameter Resistance (ohm) 初期値: 1.3	抵抗 [ohm].
Motor Parameter Inductance of d-axis (H) 初期値: 0.013	d 軸インダクタンス [H].
Motor Parameter Inductance of q-axis (H) 初期値: 0.013	q 軸インダクタンス [H].
Motor Parameter Permanent magnetic flux (Wb) 初期値: 0.01119	磁束 [Wb].
Motor Parameter Rotor inertia (kgm ²) 初期値: 0.000003666	イナーシャ [kgm ²].
Design Parameter Current PI loop omega 初期値: 300.0	電流制御系固有周波数 [Hz]
Design Parameter Current PI loop zeta 初期値: 1.0	電流制御系減衰係

Motor Current Controller (rm_motor_current)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	▼ Module Motor Current Controller (rm_motor_current)	
	▼ General	
	Name	g_motor_current0
	Sensor type	 Hall
	Shunt type	1 shunt
	Current control decimation	0
	PWM carrier frequency (kHz)	20.0
	Input voltage (V)	24.0
	Sample delay compensation	Enable
	Voltage error compensation	Enable
	Voltage error compensation table of voltage 1	0.477
	Voltage error compensation table of voltage 2	0.742
	Voltage error compensation table of voltage 3	0.892
	Voltage error compensation table of voltage 4	0.979
	Voltage error compensation table of voltage 5	1.009
	Voltage error compensation table of current 1	0.021
	Voltage error compensation table of current 2	0.034
	Voltage error compensation table of current 3	0.064
	Voltage error compensation table of current 4	0.158
	Voltage error compensation table of current 5	0.400
	▼ Interrupts	
	Callback	 rm_motor_hall_current_callback
	▼ Design Parameter	
	Current PI loop omega (Hz)	300.0
	Current PI loop zeta	1.0F
	▼ Motor Parameter	
	Pole pairs	4
	Resistance (ohm)	1.3
	Inductance of d-axis (H)	0.0013
	Inductance of q-axis (H)	0.0013
	Permanent magnetic flux (Wb)	0.01119
	Rotor inertia (kgm ²)	0.000003666

図 3-12 モータ電流コントローラの FSP コンフィグレーション

表 3-15 モータスピードコントローラ構成オプション(1/2)

Configuration Options (rm_motor_speed.h)	
オプション名	内容
Speed control period (sec) 初期値: 0.0005	速度制御周期[sec]
Step of speed climbing (rpm) 初期値: 0.5	速度変動のステップ値[rpm]。加速と減速でこの値によって速度を制御します。
Maximum rotational speed (rpm) 初期値: 2400.0	最大速度 [rpm]
Speed LPF omega 初期値: 10.0	速度 LPF 固有周波数 [Hz]
Speed at Id climbing (rpm) 初期値: 400	d 軸電流の増加を制御する閾値[rad / s]。速度がこの値に達するまで、起動時に d 軸電流を増加させます。
Limit of q-axis current (A) 初期値: 1.67	q 軸電流リミット [A]
Step of speed feedback at open-loop 初期値: 0.2	オープンループ時指令速度ステップ
Flux weakening 初期値: Disable	弱め磁束制御の選択

表 3-16 モータスピードコントローラ構成オプション(2/2)

Configuration Options (rm_motor_speed.h)	
オプション名	内容
Open-Loop Step of d-axis current climbing 初期値: 0.3	d 軸電流指令値加算ステップ [A/msec]
Open-Loop Step of d-axis current descending 初期値: 0.3	d 軸電流指令値減算ステップ [A/msec]
Open-Loop Step of q-axis current descending ratio 初期値: 1.0	q 軸電流指令値減算ステップ [A/msec]
Open-Loop Reference of d-axis current 初期値: 0.3	オープンループ制御時 d 軸電流指令値 [A]
Open-Loop Threshold of speed control descending 初期値: 500.0	d 軸電流指令値減算開始速度[rpm]
Open-Loop Threshold of speed control climbing 初期値: 400.0	d 軸電流指令値加算開始速度 [rpm]
Open-Loop Period between open-loop to BEMF (sec) 初期値: 0.025	オープンループからの切り替え処理時間 [s]
Design parameter Speed PI loop omega 初期値: 5.0	速度制御系固有周波数 [Hz]
Design parameter Speed PI loop zeta 初期値: 1.0	速度制御系減衰係数
Motor Parameter Pole pairs 初期値: 4	極対数
Motor Parameter Resistance (ohm) 初期値: 1.3	抵抗 [ohm]
Motor Parameter Inductance of d-axis (H) 初期値: 0.0013	d 軸インダクタンス [H]
Motor Parameter Inductance of q-axis (H) 初期値: 0.0013	q 軸インダクタンス [H]
Motor Parameter Permanent magnetic flux (Wb) 初期値: 0.01119	磁束 [Wb]
Motor Parameter Rotor inertia (kgm ²) 初期値: 0.000003666	イナーシャ [kgm ²]

Motor Speed Controller (rm_motor_speed)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	Position Support	Disabled
	▼ Module Motor Speed Controller (rm_motor_speed)	
	▼ General	
	Name	g_motor_speed0
	Speed control period (sec)	0.0005
	Step of speed climbing (rpm)	0.5
	Maximum rotational speed (rpm)	2400.0
	Speed LPF omega	10.0
	Speed at Id climbing (rpm)	400
	Limit of q-axis current (A)	1.67
	Step of speed feedback at open-loop	0.2
	Natural frequency	100.0
	Open-loop damping	Enable
	Flux weakening	Disable
	Torque compensation for sensorless transition	Disable
	Speed observer	Enable
	Selection of speed observer	Disturbance
	Control method	PID
	Control type	🔒 Hall
	▼ Open-Loop	
	Step of d-axis current climbing	0.3
	Step of d-axis current descending	0.3
	Step of q-axis current descending ratio	1.0
	Reference of d-axis current	0.3
	Threshold of speed control descending	500
	Threshold of speed control climbing	400
	Period between open-loop to BEMF (sec)	0.025
	Phase error(degree) to decide sensor-less switch timing	10
	▼ Design parameter	
	Speed PI loop omega	5.0
	Speed PI loop zeta	1.0
	Estimated d-axis HPF omega	2.5
	Open-loop damping zeta	1.0
	Cutoff frequency of phase error LPF	10.0
	Speed observer omega	200.0
	Speed observer zeta	1.0
	▼ Motor Parameter	
	Pole pairs	4
	Resistance (ohm)	1.3
	Inductance of d-axis (H)	0.0013
	Inductance of q-axis (H)	0.0013
	Permanent magnetic flux (Wb)	0.01119
	Rotor inertia (kgm ²)	0.000003666
	▼ Interrupts	
	Callback	🔒 rm_motor_hall_speed_callback
	Input data	🔒 (g_motor_hall0_ctrl.st_speed_input)
	Output data	🔒 (g_motor_hall0_ctrl.st_speed_output)

図 3-13 モータスピードコントローラの FSP コンフィグレーション

表 3-17 モータアングルドライバ構成オプション

Configuration Options (rm_motor_sense_hall.h)	
オプション名	内容
PMW Carrier Frequency (kHz) 初期値: 20.0	キャリア周波数 [kHz]
Correction parameter of rotor angle 初期値: 0.4	角度補正值
Default counts of carrier interrupt 初期値: 1000	規定キャリア計測回数
Maximum counts of one rotation 初期値: 5000	ホールセンサ信号間計測回数最大値

g_motor_sense_hall0 Motor Angle and Speed Calculation with Hall sensors (rm_motor_sense_hall)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	▼ Module g_motor_sense_hall0 Motor Angle and Speed Calculation with	
	▼ General	
	Name	g_motor_sense_hall0
	▼ Hall sensor	
	U phase input port	BSP_IO_PORT_12_PIN_04
	V phase input port	BSP_IO_PORT_12_PIN_05
	W phase input port	BSP_IO_PORT_11_PIN_01
	sensor pattern #1	1
	sensor pattern #2	5
	sensor pattern #3	4
	sensor pattern #4	6
	sensor pattern #5	2
	sensor pattern #6	3
	PWM Carrier Frequency (kHz)	20.0
	Correction parameter of rotor angle	0.4
	Default counts of carrier interrupt	1000
	Maximum counts of one rotation	5000

図 3-14 モータアングルドライバのFSPコンフィグレーション

表 3-18 ADC と PWM モジュレーションドライバ構成オプション

Configuration Options (rm_motor_driver.h)	
オプション名	内容
PWM timer frequency (MHz) 初期値: 120	PWM タイマ周波数 [MHz]
PWM carrier period (micro seconds) 初期値: 50	PWM キャリア周期 [Micro seconds]
Dead time (raw counts) 初期値: 240	デッドタイムカウント値 [Raw Counts]
Current range (A) 初期値: 16.5	電流検出レンジ [A]
Voltage range (V) 初期値: 73.26	電圧検出レンジ [V]
Counts for current offset measurement 初期値: 500	オフセット取得時積算回数
Shunt type 初期値: 1shunt	シャント抵抗をいくつ使って各相電流の検出を行うかを選択出来ます。
A/D conversion channel for U phase current 初期値: 4	U 相電流検出 A/D チャンネル番号
A/D conversion channel for main line voltage 初期値: 6	母線電圧検出 A/D チャンネル番号
Input voltage 初期値: 24.0	母線電圧入力範囲
Resolution of A/D conversion 初期値: 0xFFFF	A/D コンバータ分解能 ADC モジュールの分解能を変更した場合に 合わせてください。
Offset of A/D conversion for current 初期値: 0x7FFF	A/D コンバータ入力オフセット 回路に合わせて設定してください。
Conversion level of A/D conversion for voltage 初期値: 1.0	電圧変換レベル CPU 電圧が異なる場合に設定してください。
GTIOCA stop level 初期値: Pin level Low	上アーム停止時レベル
GTIOCB stop level 初期値: Pin level High	下アーム停止時レベル
Maximum duty 初期値: 0.9375	PWM 最大デューティ デッドタイムを除いた最大デューティ

ADC and PWM Modulation (rm_motor_driver)		
Settings	Property	Value
API Info	▼ Common	
	Parameter Checking	Default (BSP)
	ADC_B Support	Enabled
	▼ Module ADC and PWM Modulation (rm_motor_driver)	
	▼ General	
	Name	g_motor_driver0
	Shunt type	1 shunt
	Modulation method	SVPWM
	PWM output port UP	0
	PWM output port UN	0
	PWM output port VP	0
	PWM output port VN	0
	PWM output port WP	0
	PWM output port WN	0
	PWM Timer Frequency (MHz)	120
	PWM Carrier Period (Microseconds)	50
	Dead Time (Raw Counts)	240
	Current Range (A)	16.5
	Voltage Range (V)	73.26
	Counts for current offset measurement	500
	A/D conversion channel for U Phase current	4
	A/D conversion channel for W Phase current	0
	A/D conversion channel for Main Line Voltage	6
	A/D conversion channel for V Phase current	2
	A/D conversion channel for sin signal	27
	A/D conversion channel for cos signal	28
	Adjustment value to current A/D	0.0
	Minimum difference of PWM duty	480
	Adjustment delay of A/D conversion	120
	Input Voltage (V)	24.0
	Resolution of A/D conversion	0xFFFF
	Offset of A/D conversion for current	0x7FF
	Conversion level of A/D conversion for voltage	1.0
	GTIOCA Stop Level	Pin Level Low
	GTIOCB Stop Level	Pin Level High
	▼ Modulation	
	Maximum Duty	0.9375
	▼ Interrupts	
	Callback	 rm_motor_current_cyclic

図 3-15 ADC と PWM モジュレーションドライバの FSP コンフィグレーション

3.4 制御フロー（フローチャート）

3.4.1 メイン処理

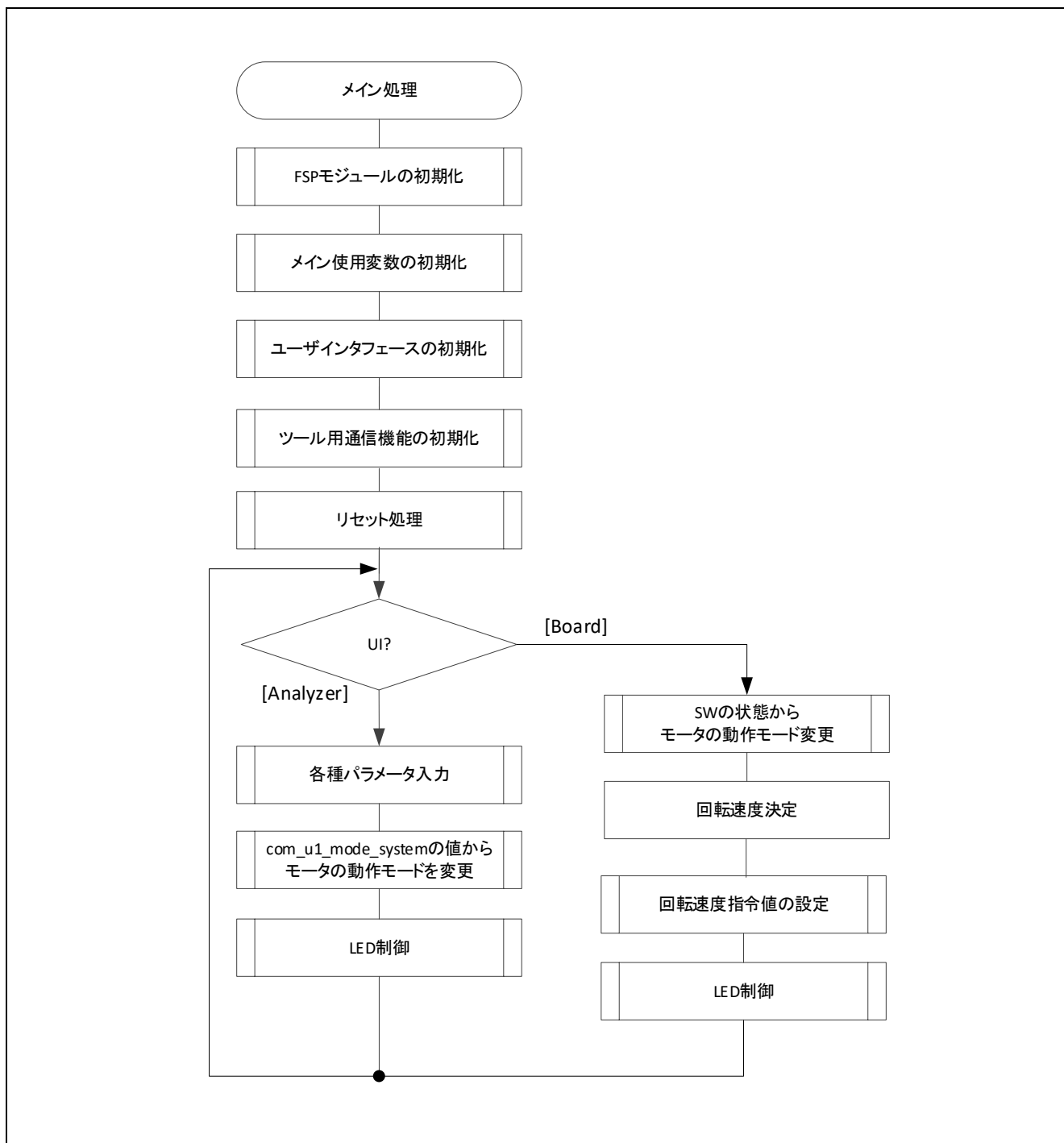


図 3-16 メイン処理フローチャート

3.4.2 50 [μs]周期割り込み（キャリア同期割り込み）処理

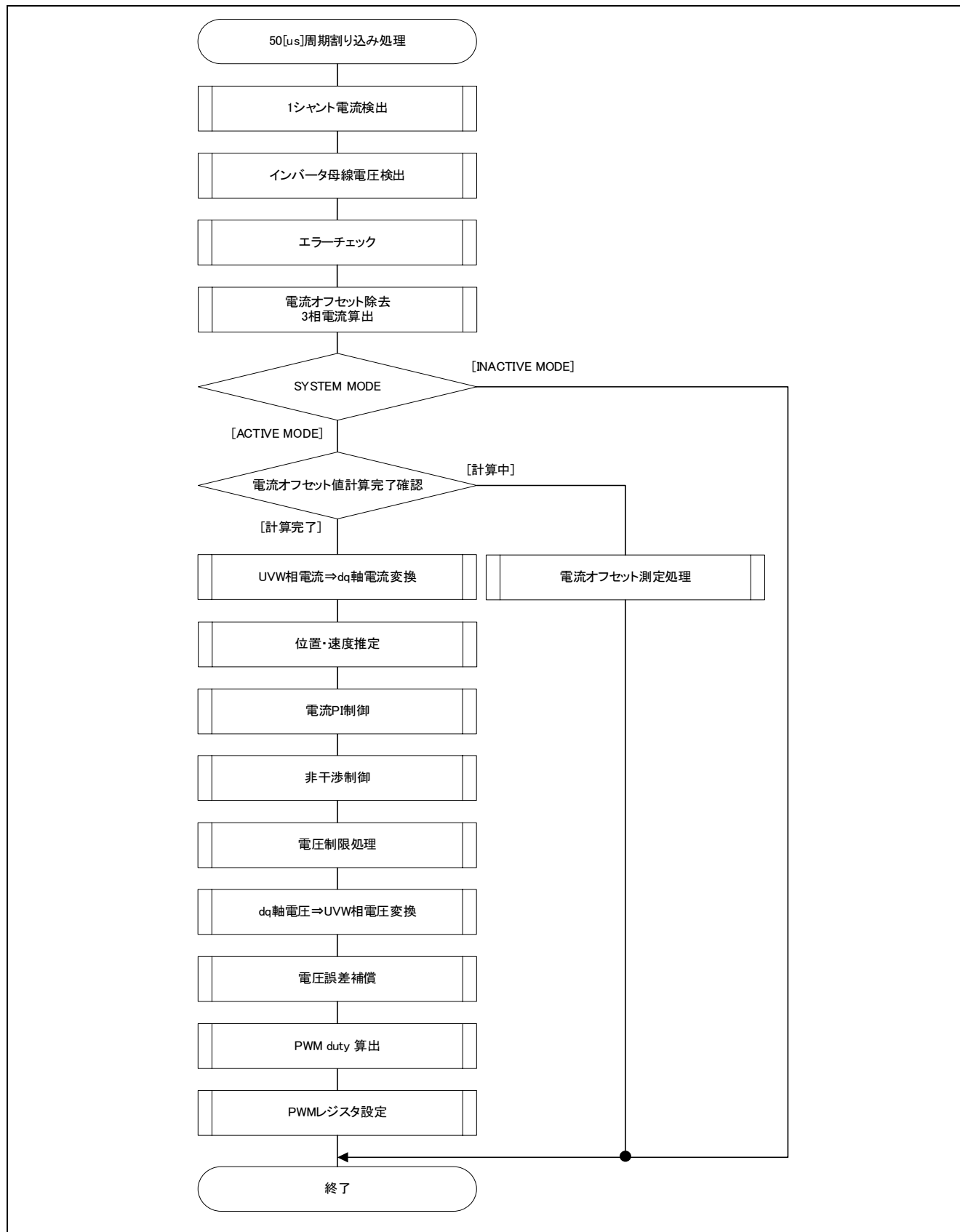


図 3-17 50[μs]周期割り込み処理フローチャート

3.4.3 500 [μs]周期割り込み処理

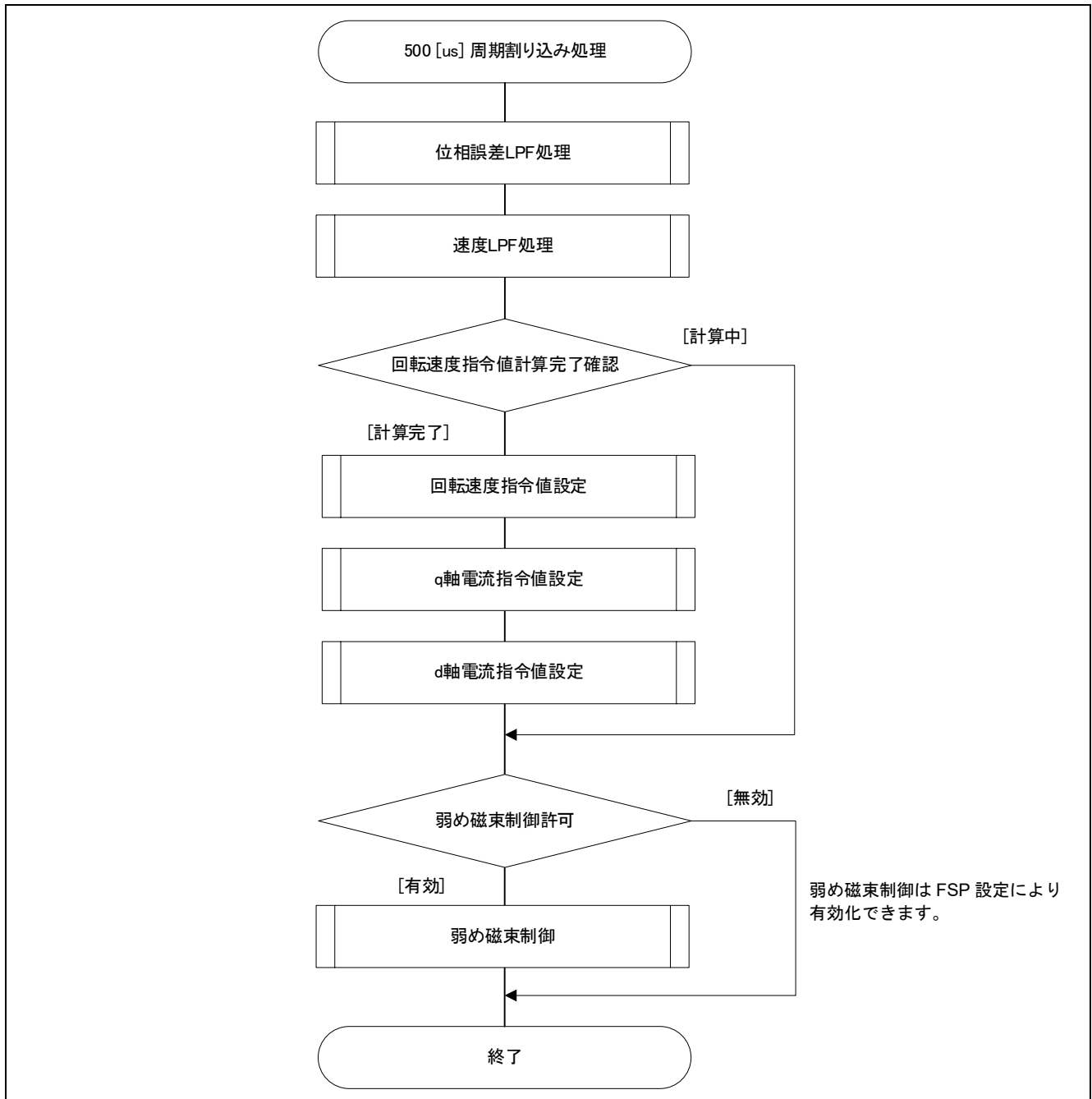


図 3-18 500 [μs]周期割り込み処理フローチャート

3.4.4 過電流検出割り込み処理

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

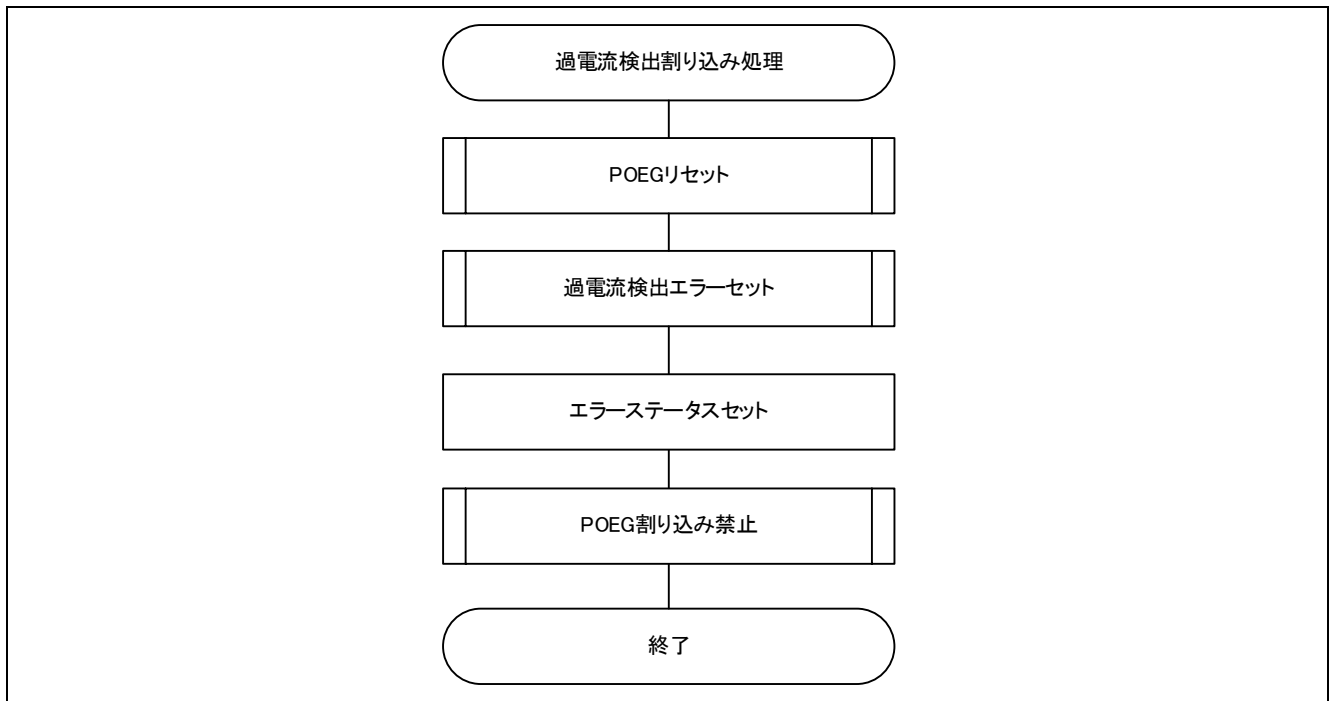


図 3-19 過電流検出割り込み処理フローチャート

4. サンプルソフトウェアの操作概要

4.1 プロジェクトのインポート

サンプルソフトは、以下の手順で e2 studio にインポートできます。

(1) File → Import

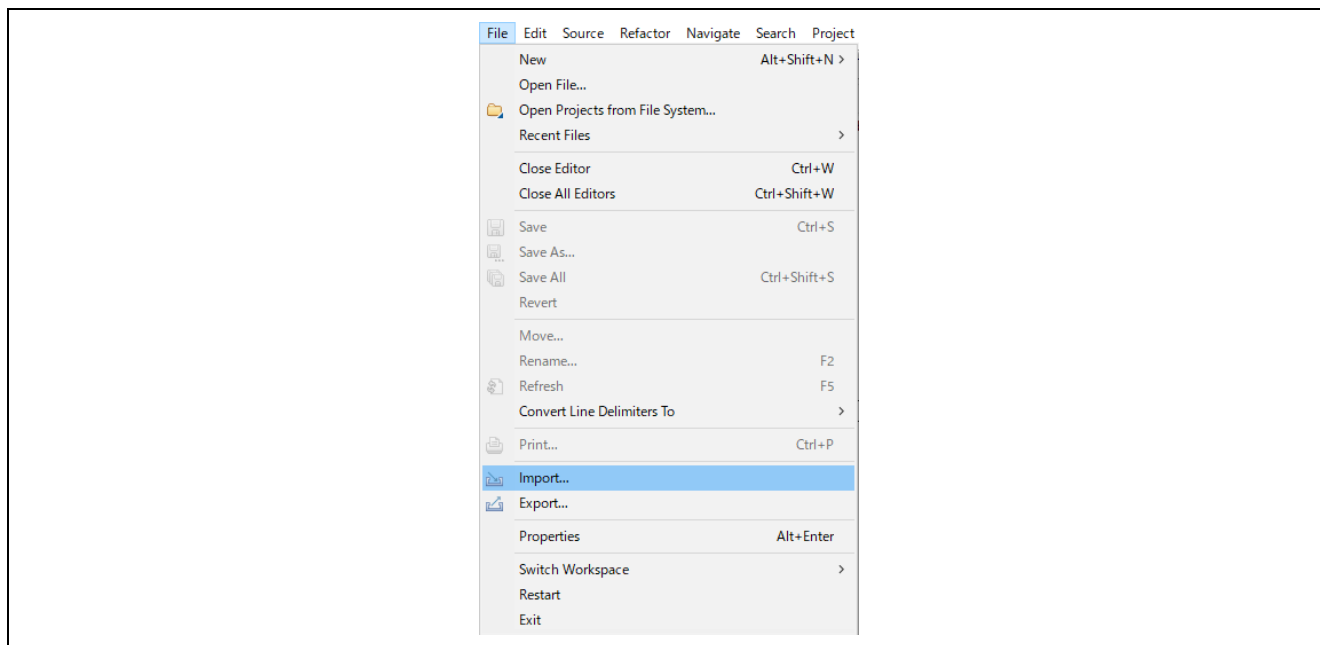


図 4-1 ファイルメニュー

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

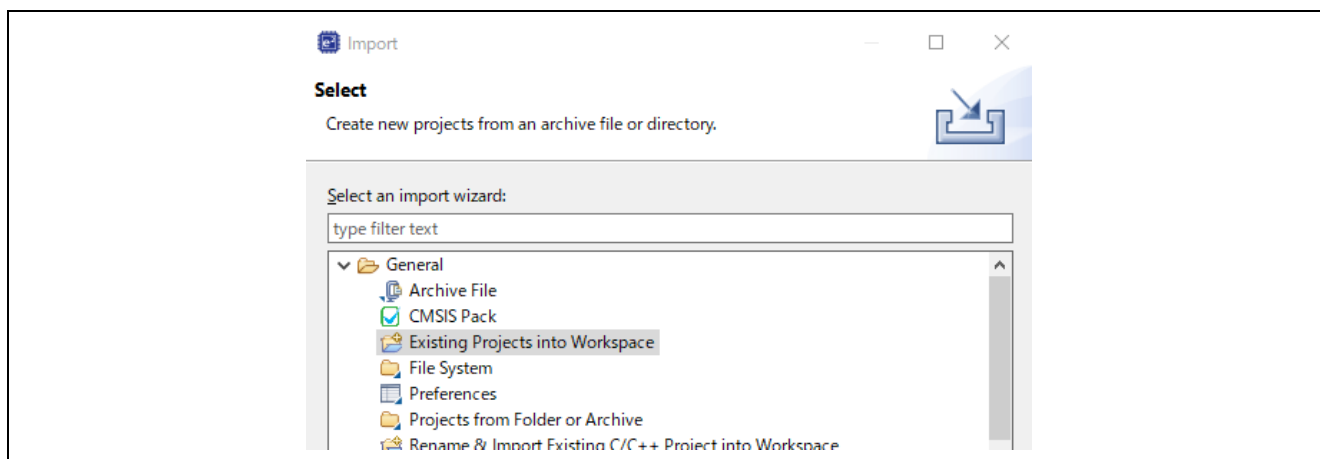


図 4-2 インポートメニュー

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

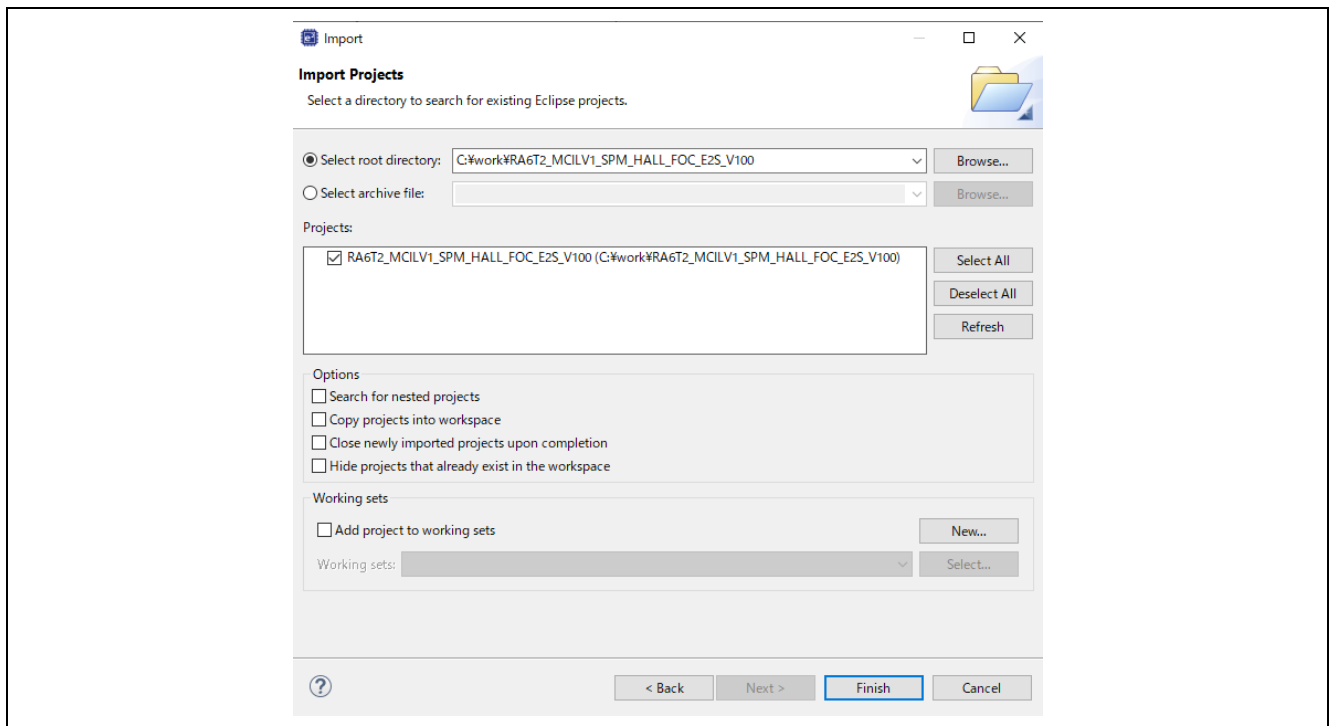


図 4-3 プロジェクトのインポート

4.2 ビルドとデバッグ

「e2 studio ユーザーズマニュアルクイックスタートガイド (R20UT5210)」を参照してください。

4.3 クイックスタート

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

- (1) 安定化電源投入後、またはリセット後はインバータボード上の LED1、LED2 は消灯状態で、モータは停止しています。
- (2) インバータボード上のトグルスイッチ(SW1)を ON にするとモータが回転します。トグルスイッチ(SW1)を切り替えるごとにモータの回転開始/停止を繰り返します。モータが正常に回転している場合はインバータボード上の LED1 が点灯します。このとき、インバータボード上の LED2 が点灯している場合はエラーが発生しています。
- (3) モータの回転方向を変更する場合は、インバータボード上のボリューム抵抗(VR)で調整します。
 - ボリューム抵抗(VR)を右に回す：時計回りに回転
 - ボリューム抵抗(VR)を左に回す：反時計回りに回転
- (4) エラーが発生した場合、インバータボード上の LED2 が点灯し、回転が停止します。復帰するためにはインバータボードのトグルスイッチ(SW1)を OFF にした上でプッシュスイッチ(SW2)を押してください。
- (5) 動作確認を終了する場合は、モータの回転が停止していることを確認し、安定化電源の出力を OFF にします。

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

4.4.1 概要

サンプルソフトウェアでは、モータ制御開発支援ツール「Renesas Motor Workbench」をユーザインタフェース(回転/停止指令、回転速度指令等)として使用します。使用方法などの詳細は「Renesas Motor Workbench ユーザーズマニュアル」を参照してください。

モータ制御開発支援ツール「Renesas Motor Workbench」は弊社 WEB サイトより入手してください。

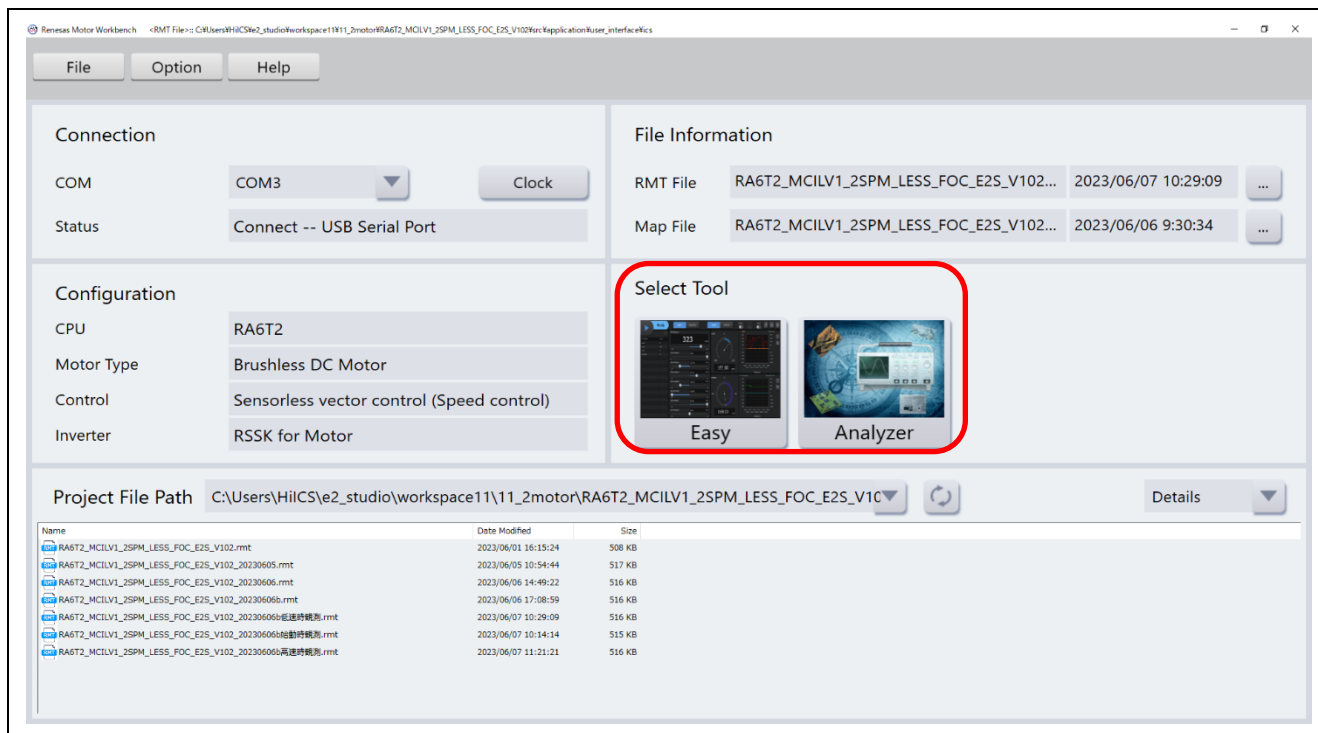


図 4-4 Renesas Motor Workbench 外観

● 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します。
- (2) Main Window の MENU バーから、[File] → [Open RMT File(O)]を選択します。
プロジェクトフォルダの“src/application/user_interface/ics”フォルダ内にある RMT ファイルを読み込みます。
- (3) “Connection”の COM で接続されたキットの COM を選択します。
- (4) 右側の Select Tool の Easy または Analyzer ボタンをクリックします。
- (5) “4.4.2 Easy 機能操作例”または“4.4.3 Analyzer 機能用変数一覧”を元にモータを駆動します。

4.4.2 Easy 機能操作例

Easy 機能を使用し、モータを操作する例を以下に示します。

- ユーザインタフェースを Renesas Motor Workbench 使用に変更します。
(1) “RMW UI”を ON にします。

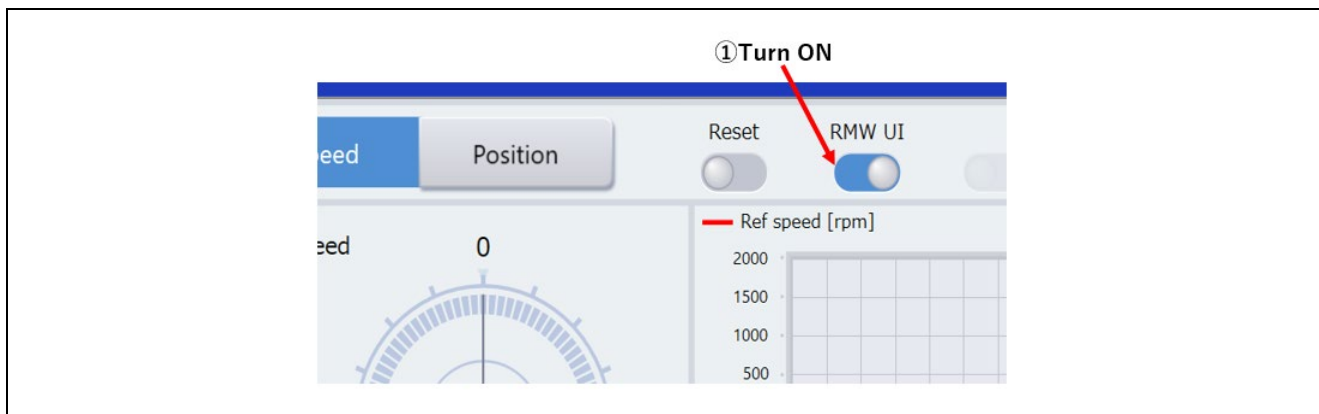


図 4-5 Renesas Motor Workbench 使用に変更する手順

- モータを回転
(1) “Run”ボタンをクリックします。
(2) 指令回転速度を“Ref speed”スライダで入力します。

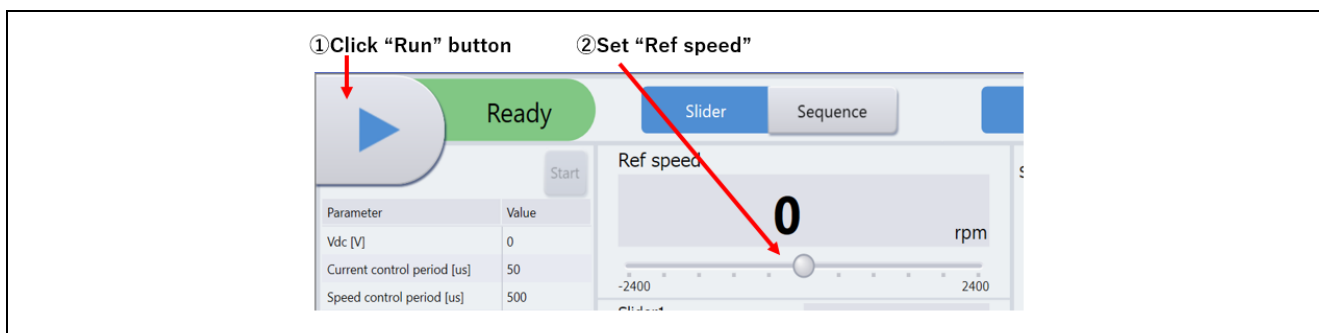


図 4-6 モータ回転の手順

- モータを停止
(1) “Stop”ボタンをクリックします。

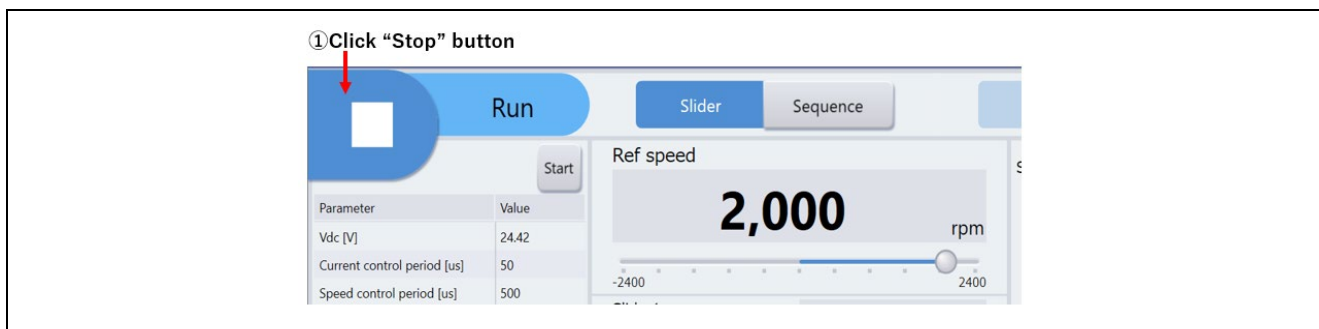


図 4-7 モータ停止の手順

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

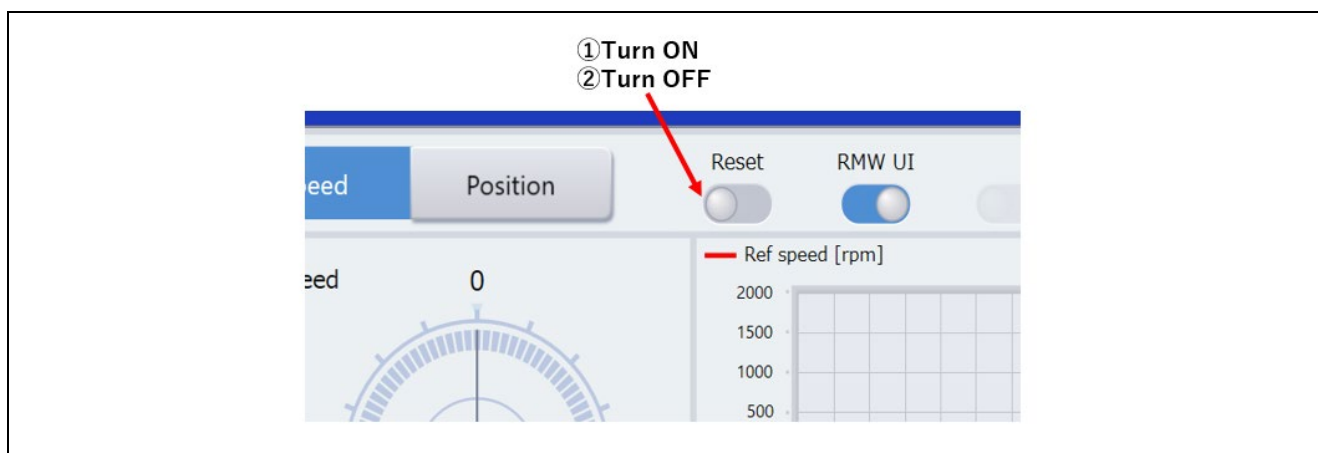


図 4-8 エラー解除の手順

4.4.3 Analyzer 機能用変数一覧

Analyzer ユーザインタフェース使用時の入力用変数一覧を表 4-1 に示します。なお、これらの変数への入力値は com_u1_enable_write に g_u1_enable_write と同じ値を書込んだ場合に対応する変数へ反映されます。ただし、(*)が付けられた変数は com_u1_enable_write に依存しません。

表 4-1 Analyzer 機能入力用変数一覧

変数名	型	内容
com_u1_sw_userif(*)	uint8_t	ユーザインタフェーススイッチ 0 : Analyzer 使用 1 : ボード使用 (デフォルト)
com_u1_mode_system1(*)	uint8_t	ステート管理 0 : ストップモード、1 : ランモード、3 : リセット
com_f4_ref_speed_rpm	float	速度指令値 (機械角) [rpm]
com_u2_mtr_pp	uint16_t	極対数
com_f4_mtr_r	float	抵抗 [Ω]
com_f4_mtr_ld	float	d 軸インダクタンス [H]
com_f4_mtr_lq	float	q 軸インダクタンス [H]
com_f4_mtr_m	float	磁束 [Wb]
com_f4_mtr_j	float	イナーシャ [kgm^2]
com_f4_current_omega	float	電流制御系固有周波数 [Hz]
com_f4_current_zeta	float	電流制御系減衰係数
com_f4_speed_omega	float	速度制御系固有周波数 [Hz]
com_f4_speed_zeta	float	速度制御系減衰係数
com_f4_ref_id	float	オープンループ制御時 d 軸電流指令値 [A]
com_f4_ol_id_up_step	float	d 軸電流指令値加算ステップ
com_f4_ol_id_down_step	float	d 軸電流指令値減算ステップ
com_f4_id_down_speed_rpm	float	d 軸電流指令値減算開始速度(機械角) [rpm]
com_f4_id_up_speed_rpm	float	d 軸電流指令値加算開始速度(機械角) [rpm]
com_f4_max_speed_rpm	float	速度最大値(機械角) [rpm]
com_f4_overspeed_limit_rpm	float	速度超過エラー閾値(機械角) [rpm]
com_f4_overcurrent_limit	float	電流超過エラー閾値[A]
com_f4_iq_limit	float	q 軸電流最大値[A]
com_f4_limit_speed_change	float	速度指令最大増減幅(電気角) [rad/s]
com_u1_enable_write	uint8_t	変数書き換え許可 (g_u1_enable_write と同じ値を書き込んだ場合にも書き込み許可)

4.4.4 Analyzer 機能操作例

Analyzer 機能を使用し、モータを操作する例を以下に示します。操作は、図 4-9 で示す"Control Window"で行います。"Control Window"の詳細は、「Renesas Motor Workbench ユーザーズマニュアル」を参照してください。

- ユーザインタフェースを Renesas Motor Workbench 使用に変更
 - (1) "com_u1_sw_userif"の[W?]欄に"チェック"が入っていることを確認します。
 - (2) [Write]欄に 0 を入力します。
 - (3) "Write"ボタンをクリックします。
- モータを回転させる
 - (1) "com_u1_mode_system","com_f4_ref_speed_rpm", "com_u1_enable_write"の [W?] 欄にチェックが入っていることを確認します。
 - (2) 指令回転速度を"com_f4_ref_speed_rpm"の [Write] 欄に入力します。
 - (3) "Write"ボタンをクリックします。
 - (4) "Read"ボタンをクリックして、現在の"com_f4_ref_speed_rpm","g_u1_enable_write"の [Read] 欄を確認します。
 - (5) MCU 内の変数値へ反映させるため、"com_u1_enable_write" に④で確認した"g_u1_enable_write"と同じ値を入力します。
 - (6) "com_u1_mode_system"の [Write]欄に"1"を入力します。
 - (7) "Write"ボタンをクリックします。

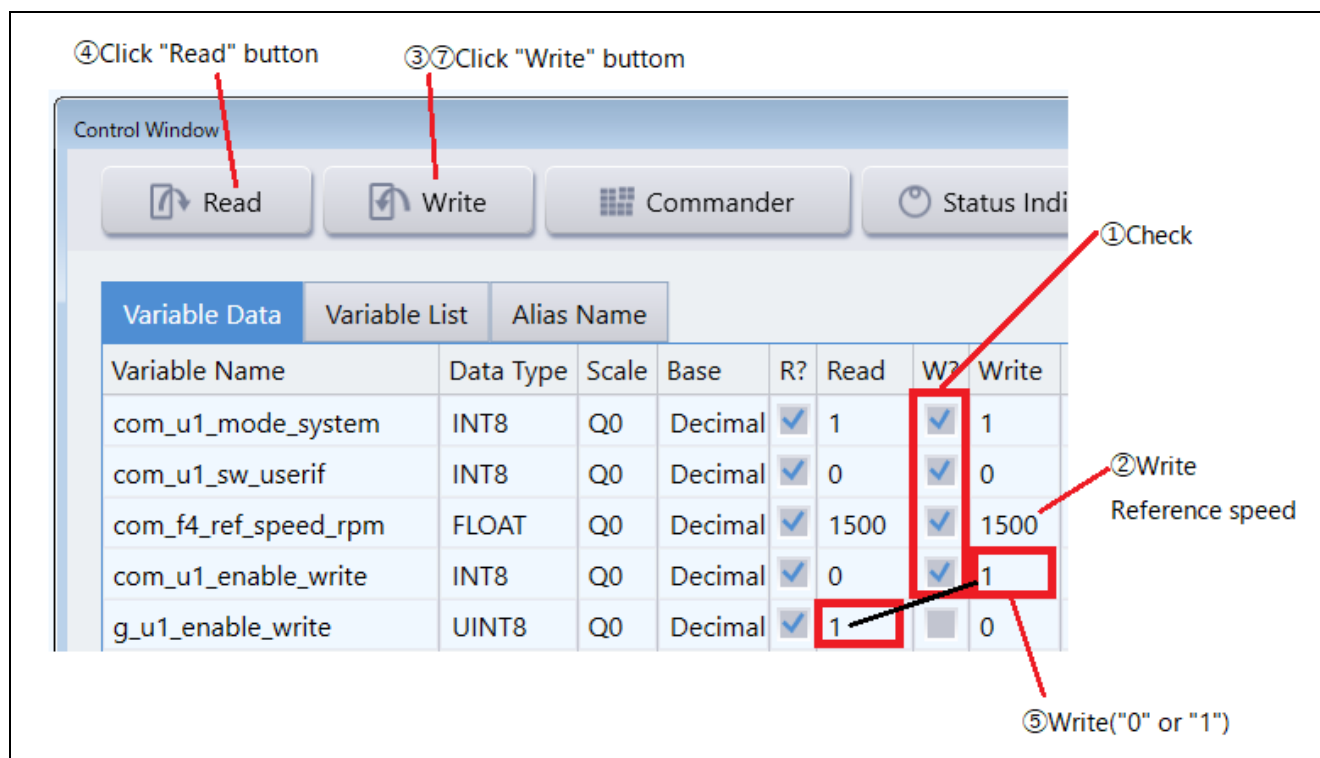


図 4-9 モータ回転の手順

- モータを停止させる
 - “com_u1_mode_system”の[Write]欄に“0”を入力します。
 - “Write”ボタンをクリックします。

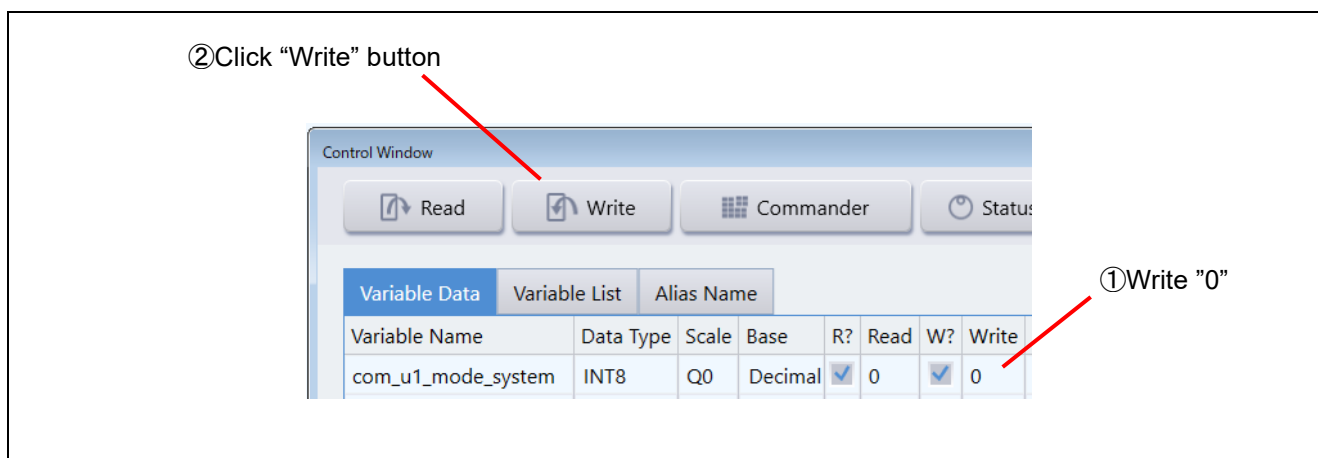


図 4-10 モータ停止の手順

- 止まってしまった（エラー）場合の処理
 - “com_u1_mode_system”の[Write]欄に“3”を入力します。
 - “Write”ボタンをクリックします。

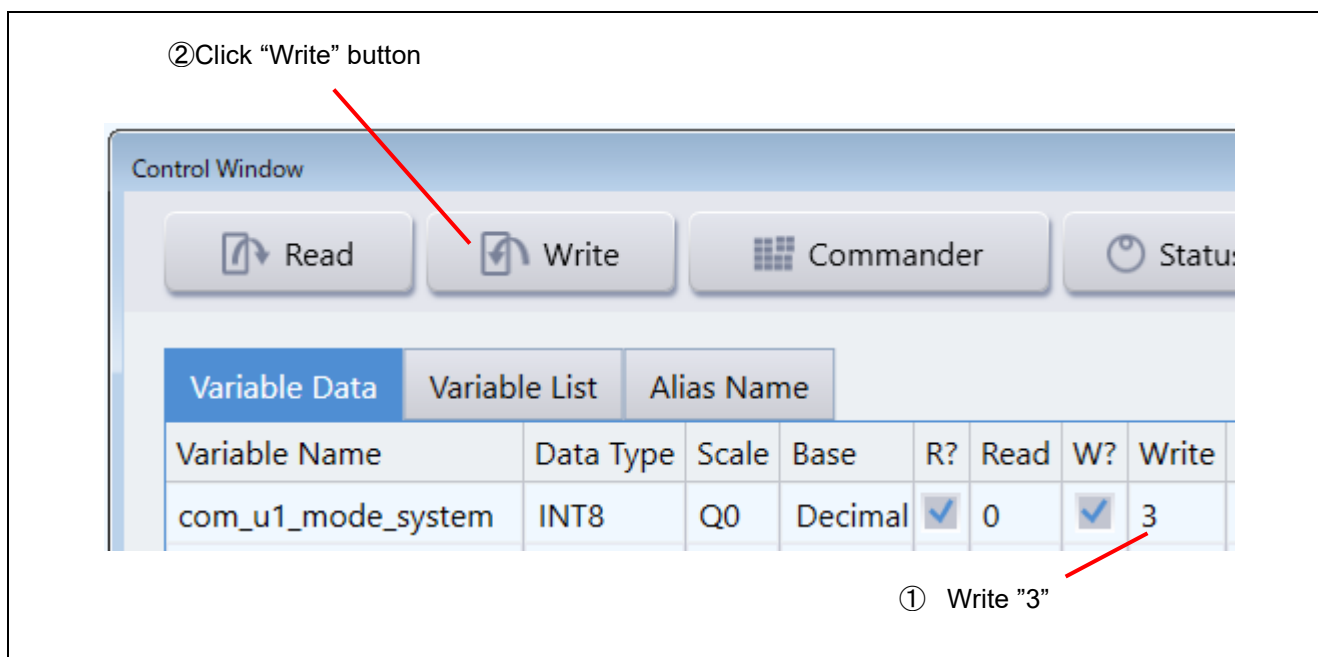


図 4-11 エラー解除の手順

4.4.5 通信速度の変更例

サンプルソフトウェアで Renesas Motor Workbench の通信速度を変更する手順を示します。変更する値については、「Renesas Motor Workbench ユーザーズマニュアル」を参照してください。

- サンプルソフトウェアの通信速度設定を変更する(通信レート 10Mbps の場合)
 - (1) r_mtr_ics.h の ICS_BRR の値を 1 に変更します。
 - (2) r_mtr_ics.h の MTR_ICS_DECIMATION の値を 1 に変更します。

```
#define MTR_ICS_DECIMATION (1)

/* For ICS */
#define ICS_BRR (1)
#define ICS_INT_MODE (1)
" " "
```

図 4-12 r_mtr_ics.h の修正

- Renesas Motor Workbench の通信速度設定を変更して接続する
 - (1) Main Window の Clock ボタンを押して値を 80,000,000 に変更します。
(この値は通信レートを 1Mbps から 10Mbps に変更したため、デフォルト値の 8,000,000 を 10 倍して得られます。)
 - (2) Connection の COM で接続中のキットの COM を選択します。

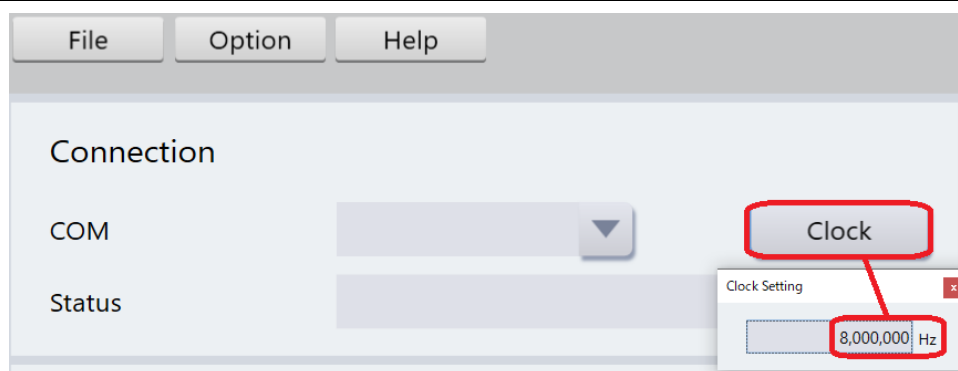


図 4-13 Clock 周波数の設定

接続に失敗する場合は、通信ボードのリセット後に再接続する手順を繰り返してください。

4.4.6 ビルトイン型通信ライブラリの使用方法

サンプルソフトウェアで通信ボードを使わずにビルトイン型通信ライブラリを用いて Renesas Motor Workbench と接続する手順を示します。

- PC と CPU ボードの接続
 - (1) CPU ボードと PC を USB/シリアル変換基板等を介して接続します。
- ビルトイン型通信用プロジェクトの準備(921600bps の例)
 - (1) ICS2_RA6T2.o の登録を解除します。

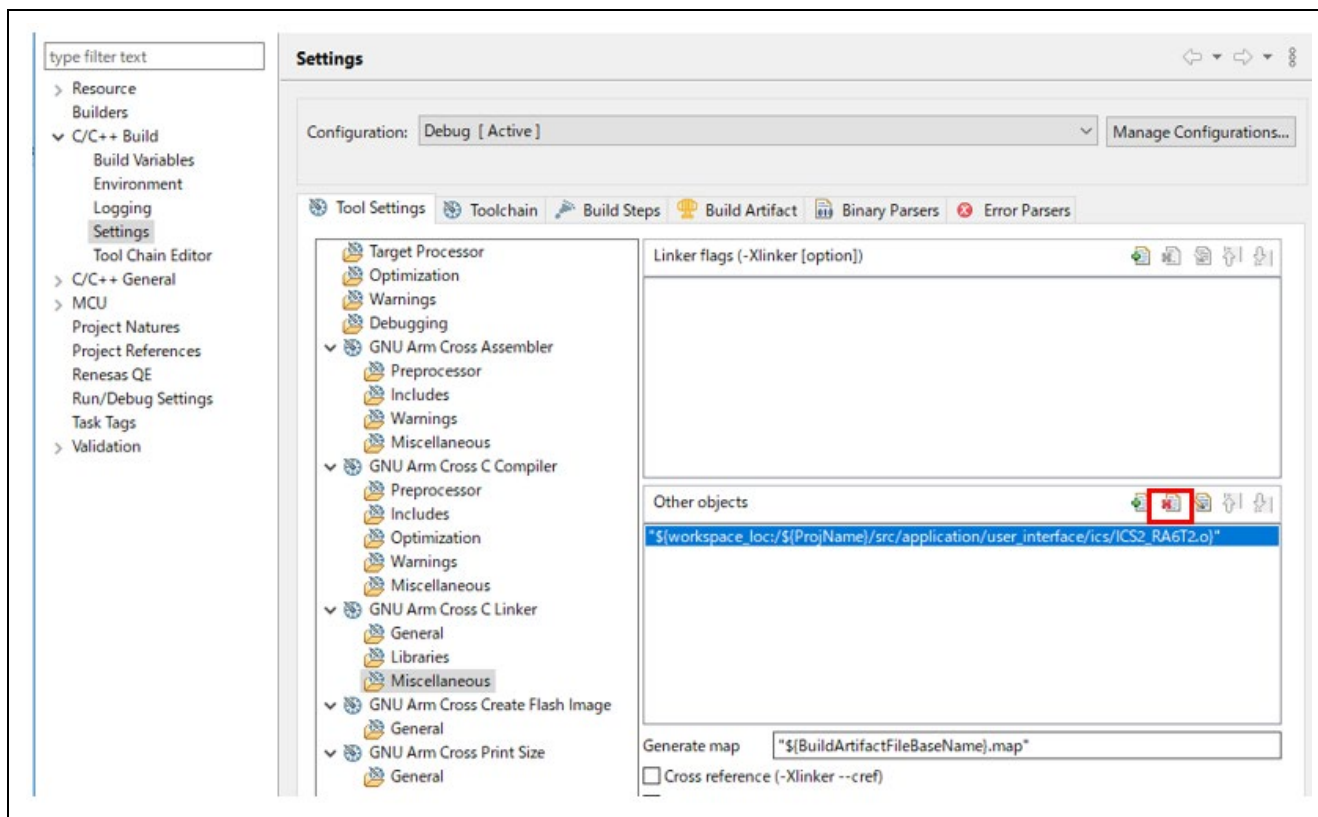


図 4-14 ICS2_RA6T2.o の登録解除

(2) ICS2_RA6T2_Built_in.o を登録します。

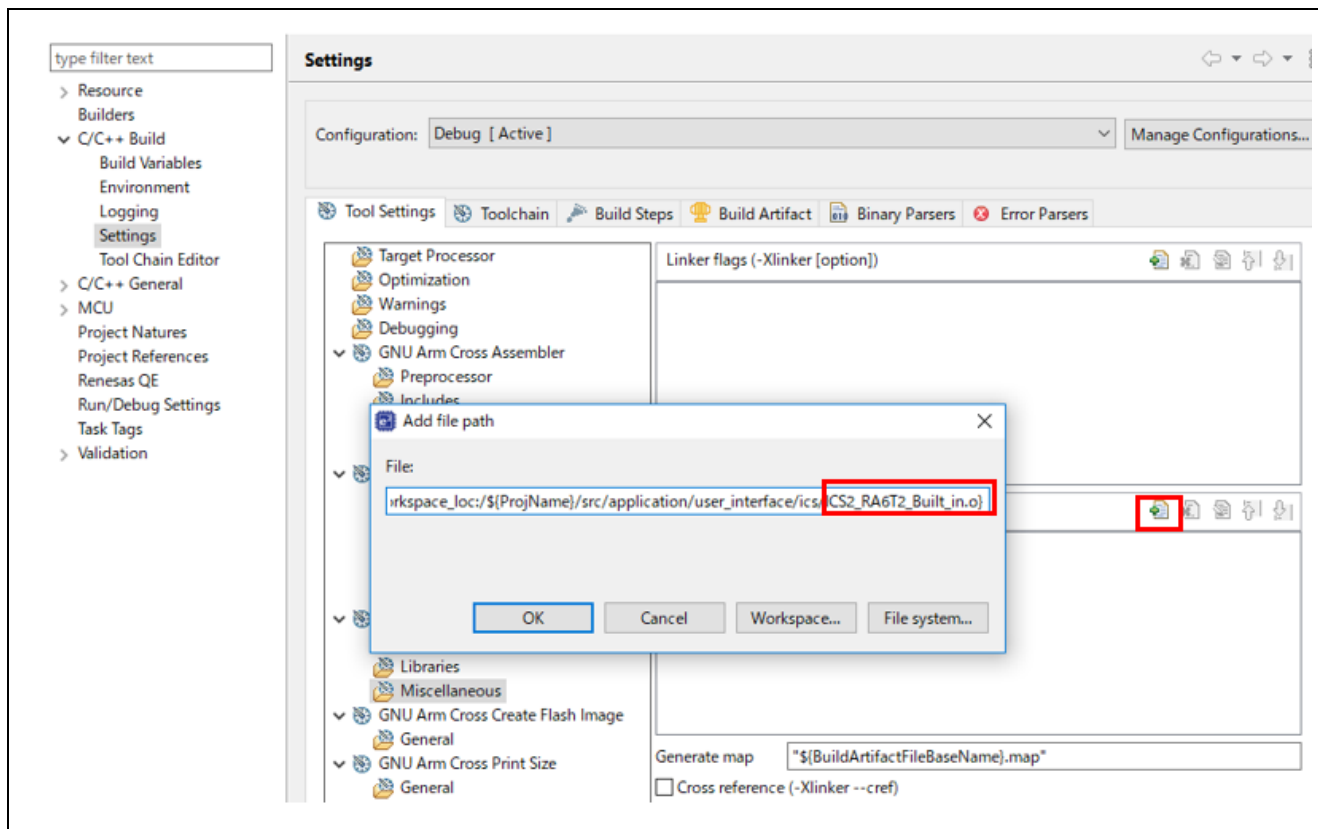


図 4-15 ICS2_RA6T2_Built_in.o を登録

(3) r_mtr_ics.h の USE_BUILT_IN の値を 1 に変更します。

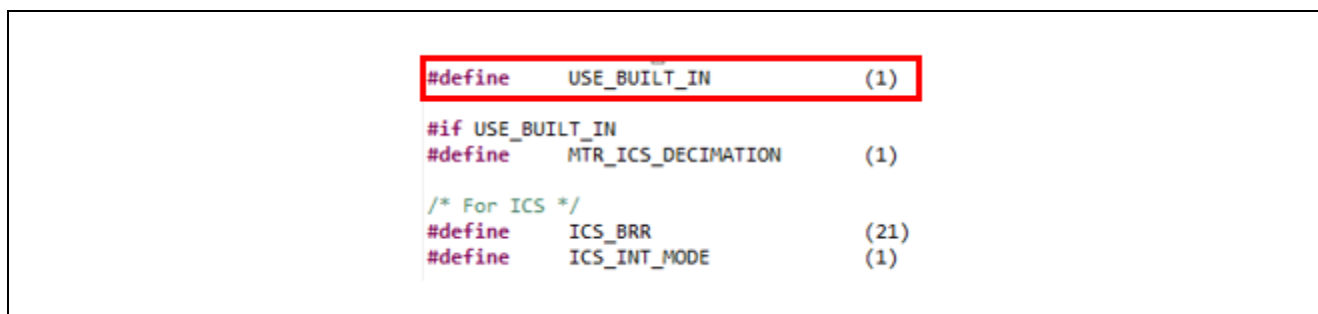


図 4-16 r_mtr_ics.h の修正

- Renesas Motor Workbench の通信ボーレート設定を変更して接続する
 - (1) Main Window の Option メニューから、Baudrate Dialog で値を 921,600 に変更します。
 - (2) Connection の COM 欄から、接続中のキットの COM を選択します。

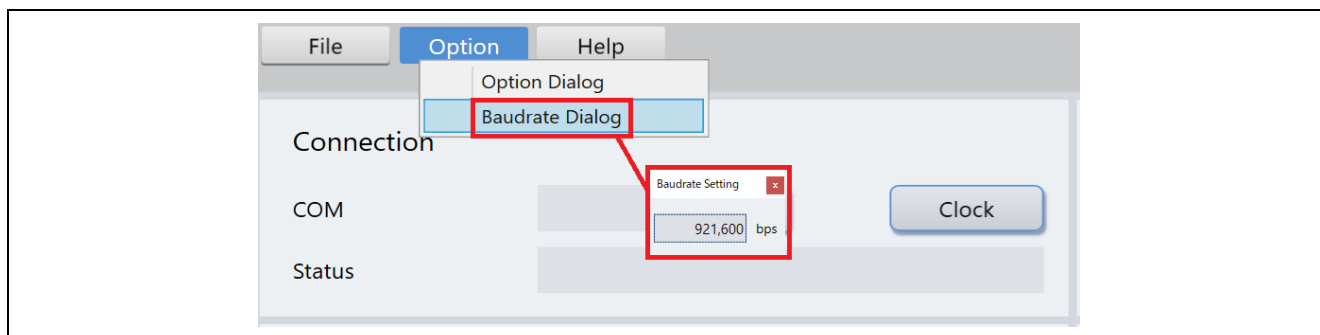


図 4-17 Baudrate の設定

5. 参考ドキュメント

- RA6T2 グループ ユーザーズマニュアル ハードウェア編 (R01UH0951)
- Renesas Flexible Software Package User's manual
(PDF 版 : R11UM0155, Web 版 : RA Flexible Software Package Documentation)
- 永久磁石同期モータのセンサレスベクトル制御 (アルゴリズム編) (R01AN3786)
- Renesas Motor Workbench ユーザーズマニュアル (R21UZ0004)
- MCK-RA6T2 ユーザーズマニュアル (R12UZ0091)

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2023.08.04	-	初版発行

製品ご使用上の注意事項

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

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

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

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

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

5. クロックについて

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

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 V_{IL} (Max.) から V_{IH} (Min.) までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 V_{IL} (Max.) から V_{IH} (Min.) までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

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

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ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）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

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

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としします。
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

お問合せ窓口

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

www.renesas.com/contact/

商標について

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