

サーボ制御向けサンプルプログラム

RA ファミリ

要旨

本アプリケーションノートはルネサス製マイクロコントローラを使用したエンコーダベクトル、誘導センサベクトル制御に付加するイナーシャ推定・原点復帰機能に関して説明することを目的としています。エンコーダベクトル、及び誘導センサベクトル制御については、永久磁石同期モータのエンコーダ利用ベクトル制御アプリケーションノート（R01AN5923/R01AN6208）、永久磁石同期モータの誘導センサ使用ベクトル制御アプリケーションノート（R01AN6467）を参照ください。

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

動作確認デバイス

サンプルソフトウェアの動作確認は下記のデバイスで行っております。

- ・ RA6T1 (R7FA6T1AD3CFP)
- ・ RA6T2 (R7FA6T2BD3CFP)

対象ソフトウェア

本アプリケーションノートの対象ソフトウェアを下記に示します。

- ・ RA6T1_ESB_SPM_ENCD_FOC_SERVO_E2S_V102
- ・ RA6T2_MCILV1_SPM_ENCD_FOC_SERVO_E2S_V111
- ・ RA6T2_MCILV1_SPM_IS_FOC_SERVO_E2S_V111

目次

1. 概要	3
2. 開発環境	3
2.1 動作確認環境	3
2.2 ハードウェア仕様	4
2.2.1 ハードウェア構成図	4
2.2.2 ボードユーザインタフェース	6
2.3 ソフトウェア構成	8
2.3.1 FSP スタック図	8
2.3.2 ソフトウェア・ファイル構成	9
2.3.3 モジュール構成	11
3. 制御ソフトウェア説明	12
3.1 制御内容	12
3.1.1 イナーシャ推定	12
3.1.2 原点復帰機能	12
3.2 サーボ制御ソフトウェア関数仕様	13
3.2.1 イナーシャ推定モジュール API	13
3.2.2 原点復帰モジュール API	14
3.3 Contents of control	15
3.3.1 Configuration Option	15
4. Renesas Motor Workbench(RMW)を用いた操作	16
4.1 プロジェクトのインポート	16
4.2 ビルドとデバッグ	17
4.3 RMW 起動	18
4.4 イナーシャ推定機能操作	20
4.5 サーボ調整機能操作	22
4.6 原点復帰機能操作	24
4.7 位置動作確認機能 (Point to Point) 操作	25
5. 参考ドキュメント	26

1. 概要

本アプリケーションノートでは、RA6T1/RA6T2 マイクロコントローラに実装したエンコーダベクトル制御、及び誘導センサ使用ベクトル制御プログラムに追加するイナーシャ推定、及び原点復帰機能について説明することを目的としています。

なお、ベクトル制御に関しては「永久磁石同期モータのエンコーダベクトル制御 (アルゴリズム編)」(R01AN3789)を参照してください。

2. 開発環境

2.1 動作確認環境

サンプルソフトウェアの開発環境を表 2-1、表 2-2 に示します。

表 2-1 ハードウェア開発環境

分類	使用製品
マイコン / CPU カード型名	RA6T1 (R7FA6T1AD3CFP) / RTK0EMA170C00000BJ RA6T2 (R7FA6T2BD3CFP) / RTK0EMA270C00000BJ
インバータボード / 型名	RA6T1 : インバータボード / RTK0EM0000B10020BJ RA6T2 : MCI-LV-1 / RTK0EM0000S04020BJ
モータ	BLY171D-24V-4000
センサ	エンコーダ : AMT102-V (CUI Devices 社製) 誘導センサ : IPS2200

表 2-2 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
e ² studio : 2023-04	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。

2.2 ハードウェア仕様

2.2.1 ハードウェア構成図

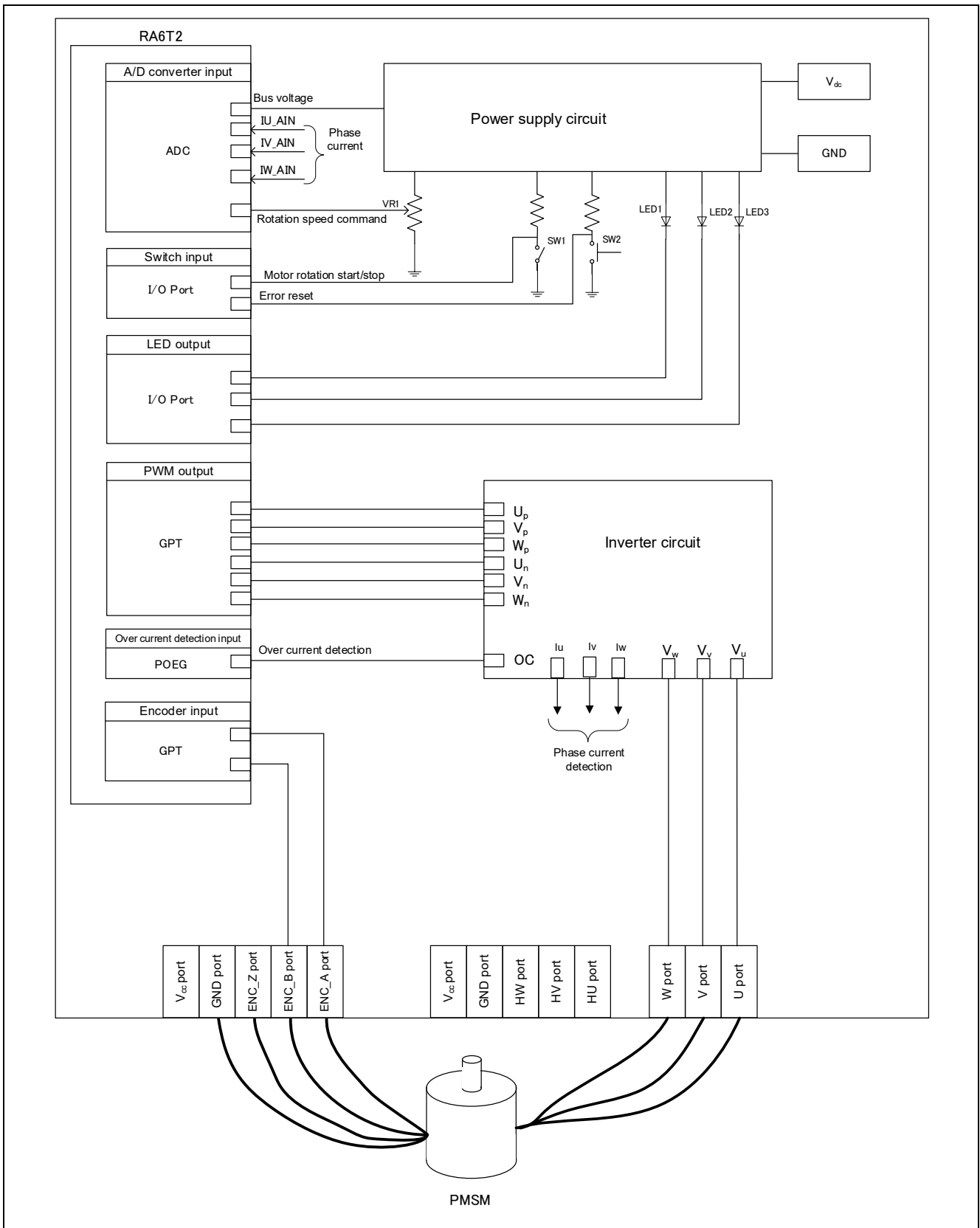


図 2-1 エンコーダ対応ハードウェア構成

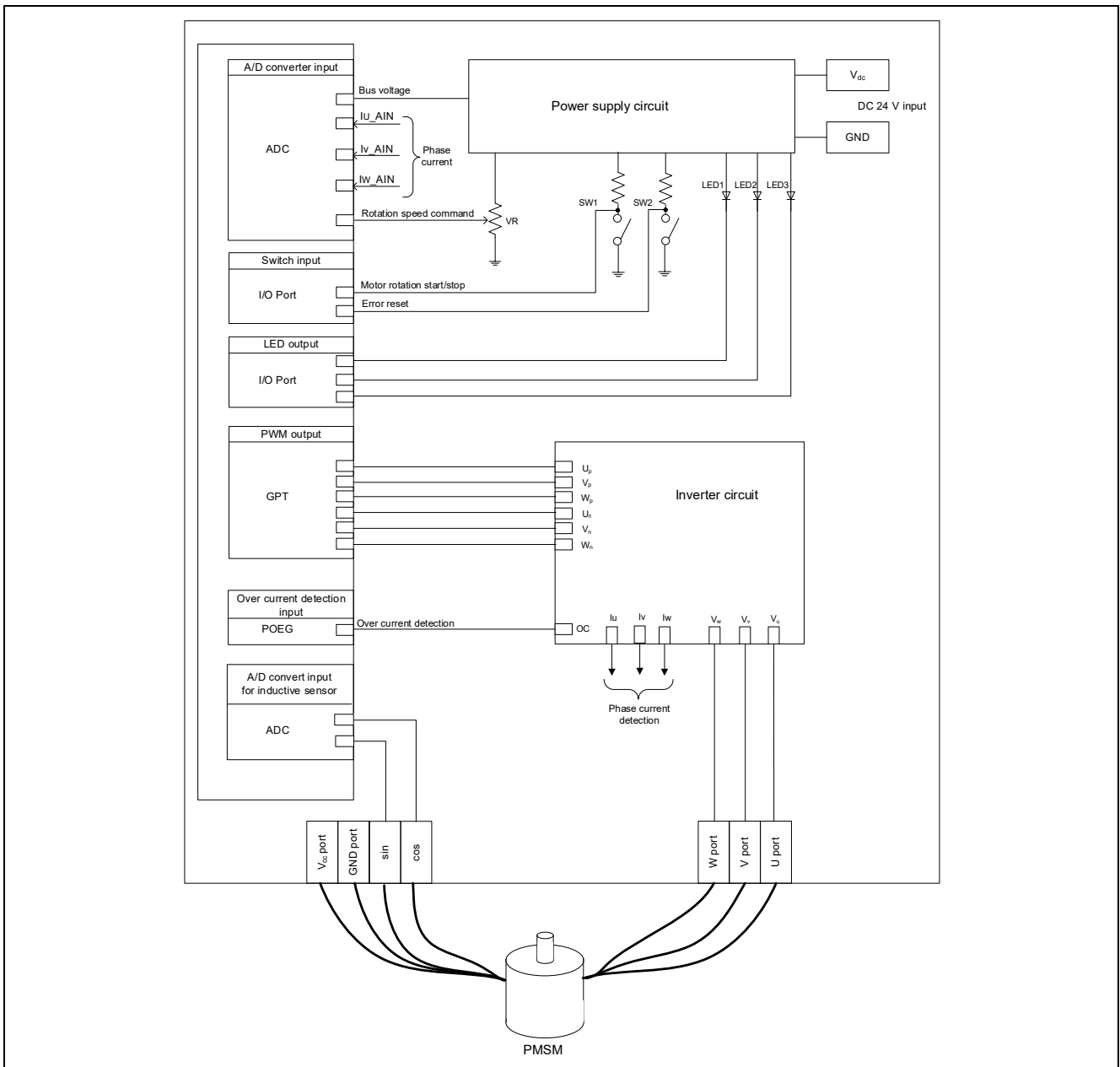


図 2-2 誘導センサ対応ハードウェア構成

2.2.2 ボード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。

表 2-3 インバータボードユーザインタフェース

項目	インタフェース部品	機能
回転速度指令	可変抵抗器	回転速度指令値入力
START / STOP	トグルスイッチ(SW1)	モータ回転開始/停止指令
ERROR RESET	プッシュスイッチ(SW2)	エラー状態からの復帰指令
LED1	橙色 LED	・モータ駆動時 : 点灯 ・モータ停止時 : 消灯
LED2	橙色 LED	・エラー検出時 : 点灯 ・通常動作時 : 消灯
LED3	橙色 LED	未使用

表 2-4 CPU ボードユーザインタフェース

項目	インタフェース部品	機能
LED1	橙色 LED	・モータ駆動時 : 点灯 ・モータ停止時 : 消灯
LED2	橙色 LED	・エラー検出時 : 点灯 ・通常動作時 : 消灯
LED3	橙色 LED	未使用
RESET	プッシュスイッチ(RESET)	システムリセット

本システムの端子インタフェースを以下に示します。

表 2-5 端子インタフェース

機能	RA6T2	RA6T1
インバータ母線電圧測定	PA06 / AN006	P014 / AN005
回転速度指令値入力用(アナログ値)	PB00 / AN008	P502 / AN017
START/STOP トグルスイッチ(SW1)	PD04	P302
ERROR RESET プッシュスイッチ(SW2)	PD07	P301
LED1 点灯/消灯制御	PD01	P408
LED2 点灯/消灯制御	PD02	P407
U 相電流測定	PA04 / AN004	P000 / AN000
V 相電流測定	PA02 / AN002	P001 / AN001
W 相電流測定	PA00 / AN000	P002 / AN002
PWM 出力(U _p)	PB04 / GTIOC4A	P415 / GTIOC0A
PWM 出力(V _p)	PB06 / GTIOC5A	P113 / GTIOC2A
PWM 出力(W _p)	PB08 / GTIOC6A	P111 / GTIOC3A
PWM 出力(U _n)	PB05 / GTIOC4B	P414 / GTIOC0B
PWM 出力(V _n)	PB07 / GTIOC5B	P114 / GTIOC2B
PWM 出力(W _n)	PB09 / GTIOC6B	P112 / GTIOC3B
エンコーダ A 相入力	PC14 / GTIOC3A	P107 / GTIOC8A
エンコーダ B 相入力	PC15 / GTIOC3B	P106 / GTIOC8B
誘導センサ cos 入力 (IPS2200)	PE11 / AN023	RA6T1 は非対応
誘導センサ sin 入力 (IPS2200)	PE12 / AN024	RA6T1 は非対応
過電流検出時の PWM 緊急停止入力	PC13 / GTETRGD	P503 / GTETRGC

2.3 ソフトウェア構成

2.3.1 FSP スタック図

各プロジェクトの FSP スタック図を以下に示します。

赤枠で囲まれたモジュールが追加するイナーシャ推定・原点復帰モジュールとなります。

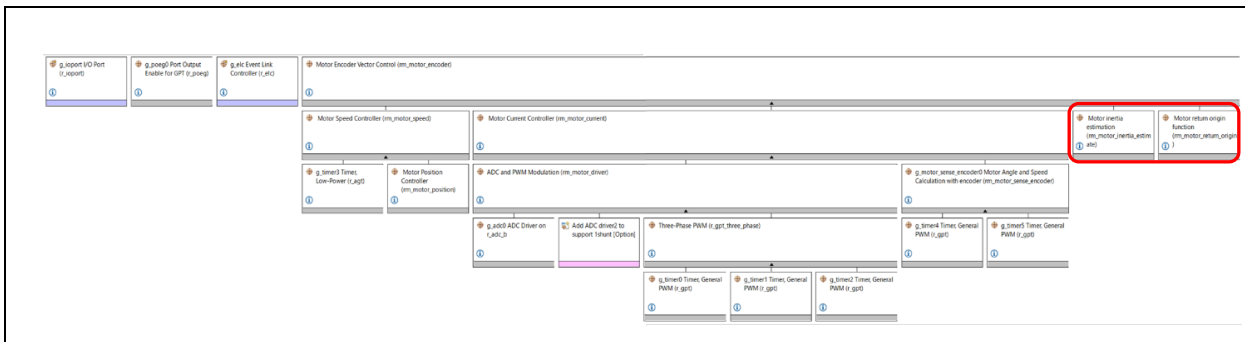


図 2-3 RA6T2 エンコーダ使用ベクトル制御プロジェクト FSP スタック構成

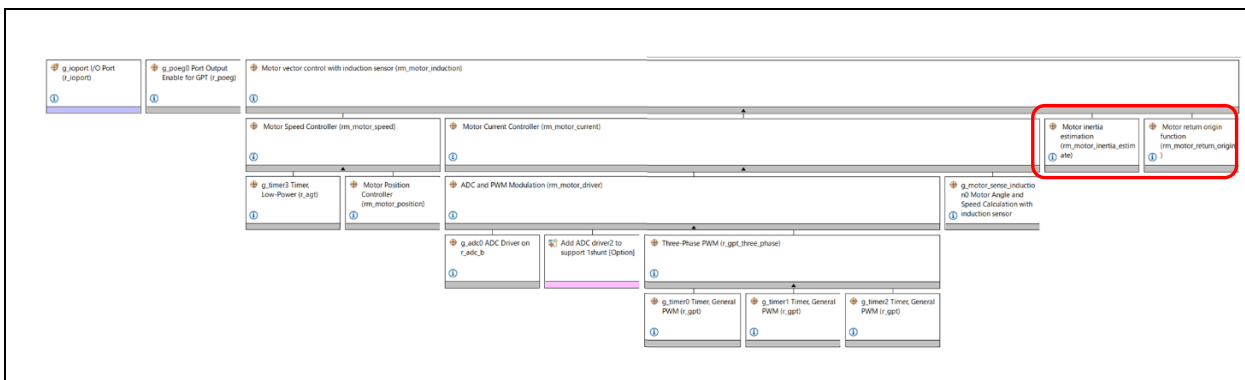


図 2-4 RA6T2 誘導センサ使用ベクトル制御プロジェクト FSP スタック構成

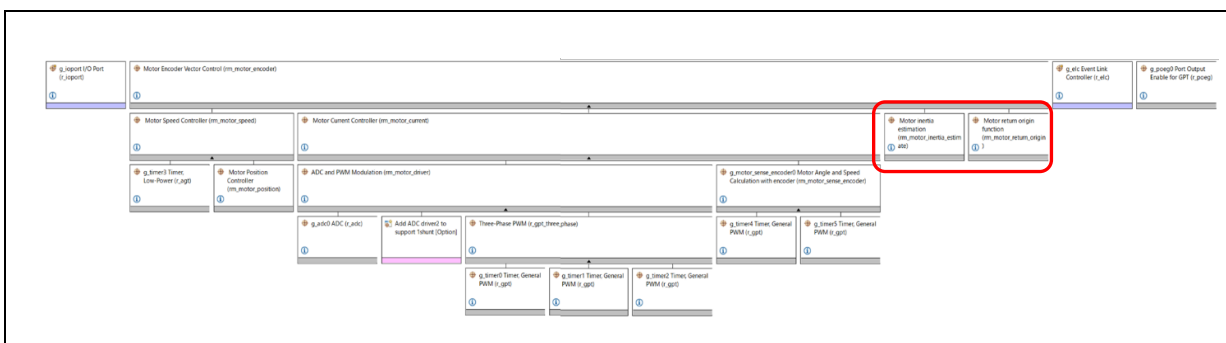


図 2-5 RA6T1 エンコーダ使用ベクトル制御プロジェクト FSP スタック構成

2.3.2 ソフトウェア・ファイル構成

サンプルソフトウェアのフォルダとファイル構成を以下に示します。

表 2-6 ソフトウェアフォルダ構成[1/2]

フォルダ	サブフォルダ	ファイル	備考
ra_cfg			自動生成のコンフィグヘッダ
ra_gen			自動生成のレジスタ設定値、メイン関数等
ra	arm		CMSIS ソースコード
	board		ボード関連関数定義
	fsp/inc/api	bsp_api.h	BSP API 定義
		r_adc_api.h	AD API 定義
		r_elc_api.h	ELC API 定義
		r_ioport_api.h	I/O API 定義
		r_poeg_api.h	POEG API 定義
		r_three_phase_api.h	3相 PWM API 定義
		r_timer_api.h	タイマ API 定義
		r_transfer_api.h	データ転送 API 定義
		rm_motor_angle_api.h	角度 API 定義
		rm_motor_api.h	モータ API 定義
		rm_motor_current_api.h	電流制御 API 定義
		rm_motor_driver_api.h	モータドライバ API 定義
		rm_motor_inertia_estimate_api.h	イナーシャ推定 API 定義
		rm_motor_position_api.h	位置制御 API 定義
		rm_motor_return_origin_api.h	原点復帰 API 定義
		rm_motor_speed_api.h	速度 API 定義
	fsp/inc/instances	r_adc_b.h(RA6T2) r_adc.h(RA6T1)	AD 関連定義
		r_agt.h	AGT 関連定義
		r_elc.h	ELC 関連定義
		r_gpt_three_phase.h	3相 PWM 関連定義
		r_gpt.h	GPT 関連定義
		r_ioport.h	I/O 関連定義
		r_poeg.h	POEG 関連定義
		rm_motor_current.h	電流制御関連定義
		rm_motor_driver.h	モータドライバ関連定義
		rm_motor_encoder.h	エンコーダ使用モータ 動作管理関連定義
		rm_motor_induction.h	誘導センサ使用モータ 動作管理関連定義
		rm_motor_position_api.h	位置制御関連定義
		rm_motor_sense_encoder.h	エンコーダセンサ処理 関連定義
	rm_motor_sense_induction.h	誘導センサ処理関連定義	
	rm_motor_speed.h	速度関連定義	
fsp/lib		ライブラリファイル	

表 2-7 ソフトウェアフォルダ構成[2/2]

フォルダ	サブフォルダ	ファイル	備考
ra	fsp/src	bsp	BSP 関連フォルダ
		r_adc_b/r_adc_b.c (RA6T2) r_adc/r_adc.c (RA6T1)	AD ドライバ
		r_agt/r_agt.c	AGT ドライバ
		r_elc/r_elc.c	ELC ドライバ
		r_gpt/r_gpt.c	GPT ドライバ
		r_gpt_three_phase/r_gpt_three_phase.c	3 相 PWM ドライバ
		r_ioport/r_ioport.c	I/O ドライバ
		r_poeg/r_poeg.c	POEG ドライバ
		rm_motor_current/rm_motor_current.c	電流制御ドライバ
		rm_motor_current/rm_motor_current_library.h	電流制御ライブラリ API 定義
		rm_motor_driver/rm_motor_driver.c	モータドライバ
		rm_motor_encoder/rm_motor_encoder.c	エンコーダ使用モータ 動作管理部
		rm_motor_induction/rm_motor_induction.c	誘導センサ使用モータ 動作管理部
		rm_motor_position/rm_motor_position.c	位置制御ドライバ
		rm_motor_position/rm_motor_position_library.h	位置制御ライブラリ API 定義
		rm_motor_sense_encoder /rm_motor_sense_encoder.c	エンコーダセンサハンドラ (エンコーダ時)
		rm_motor_sense_induction /rm_motor_sense_induction.c	誘導センサハンドラ (誘導センサ時)
		rm_motor_speed/rm_motor_speed.c	速度制御ドライバ
		rm_motor_speed/rm_motor_speed_library.h	速度制御ライブラリ API 定義
		rm_motor_inertia_estimate /rm_motor_inertia_estimate.c	イナーシャ推定モジュール
rm_motor_return_origin/rm_motor_return_origin.c	原点復帰モジュール		
src	application/main	mtr_main.h , mtr_main.c	ユーザメイン関数
		r_mtr_control_parameter.h	制御パラメータ定義
		r_mtr_motor_parameter.h	モータパラメータ定義
		mtr_adjust_function.h , mtr_adjust_function.c	サーボ GUI 対応処理
	application/ user_interface/ics	r_mtr_ics.h , r_mtr_ics.c	Analyzer UI 関連関数定義
		ICS2_RA6T2.h , ICS2_RA6T1.h	ツール用通信ライブラリ
		ICS2_RA6T2.o , ICS2_RA6T1.o	ツール用通信関連定義

2.3.3 モジュール構成

サンプルソフトウェアのモジュール構成を以下に示します。

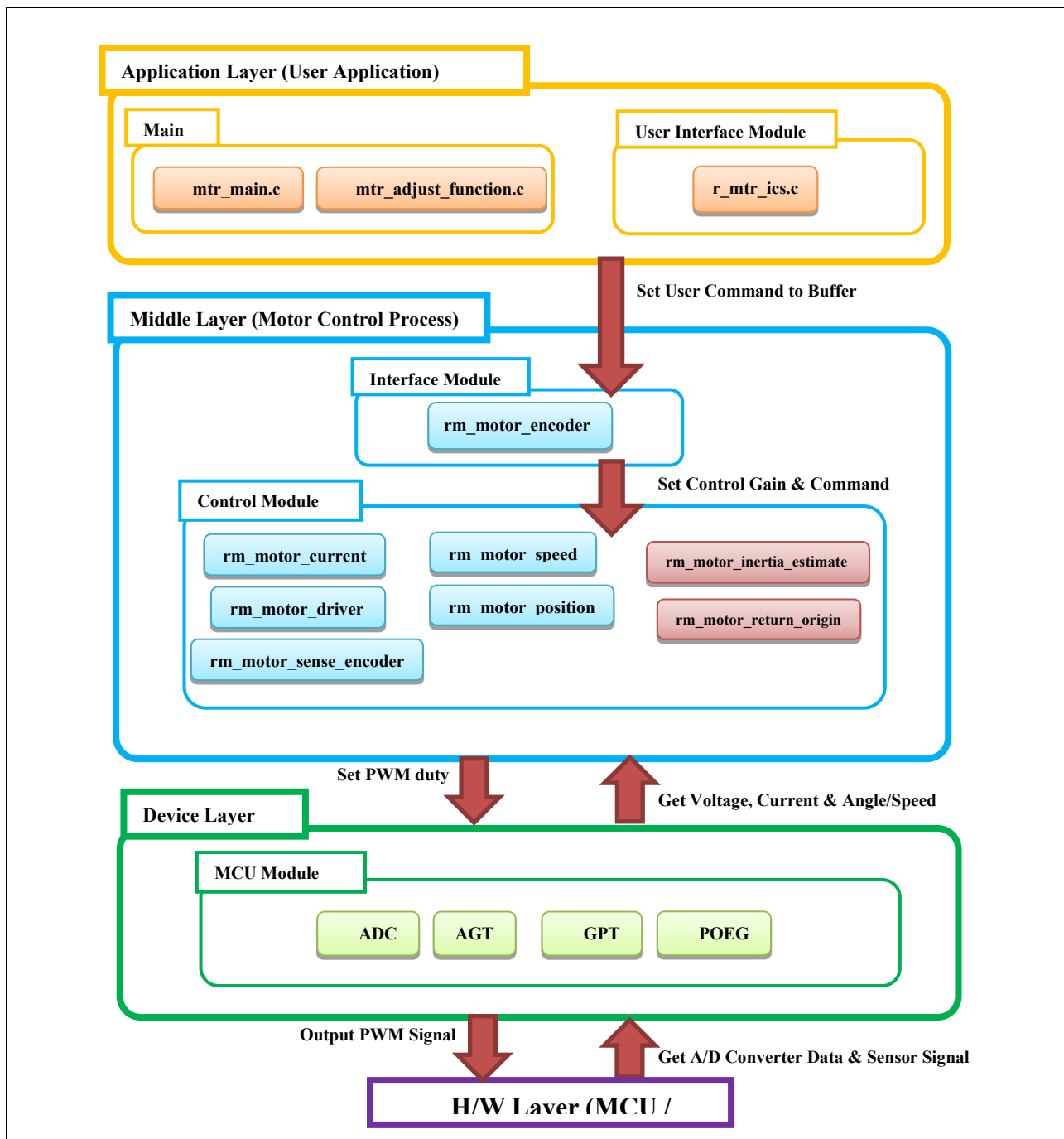


図 2-6 モジュール構成

3. 制御ソフトウェア説明

サーボ調整プログラムはベースとなるエンコーダ使用ベクトル制御ソフトウェア、及び誘導センサ使用ベクトル制御ソフトウェアと同じです。ここでは追加機能のイナーシャ推定機能、及び原点復帰機能に関して説明します。

3.1 制御内容

3.1.1 イナーシャ推定

イナーシャ推定は、モータ軸に接続されているロータ+負荷イナーシャを推定します。イナーシャはモータを正常に駆動するために重要なパラメータなので、本機能を用いて推定を行います。

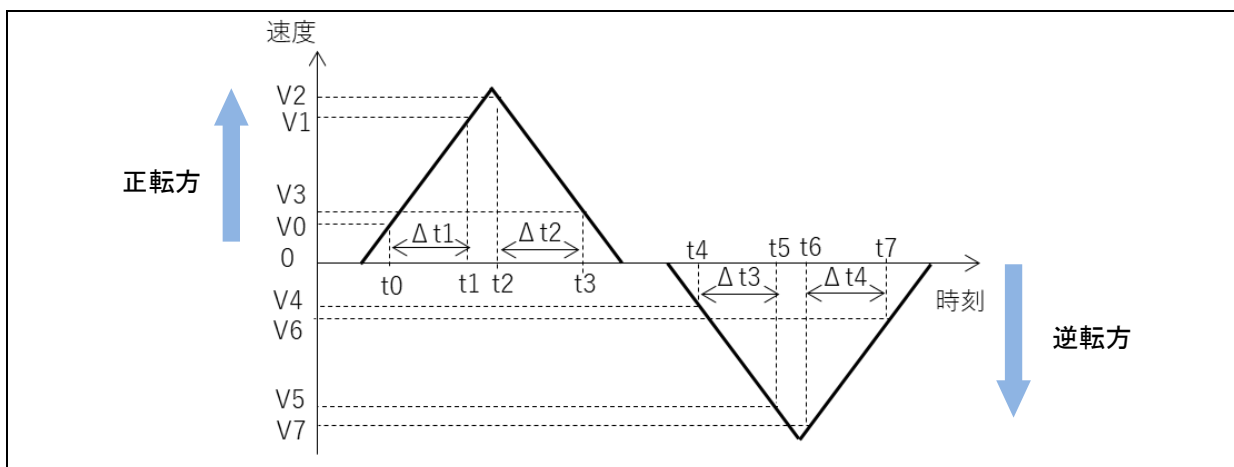


図 3-1 イナーシャ推定時動作

イナーシャ値の推定は、位置制御を用いてモータを正転方向と逆転方向に同じ移動量で動作させます。この時の正転方向加速時間($\Delta t1$)、正転方向減速時間($\Delta t2$)、逆転方向加速時間($\Delta t3$)、及び逆転方向減速時間($\Delta t4$)、各期間のq軸電流平均値及び平均加速度を用いてイナーシャ値を推定します。

3.1.2 原点復帰機能

原点復帰は、モータを機械的な基準位置に位置決めする動作です。起動時に基準位置が分からない場合に、基準位置に戻るために使用します。

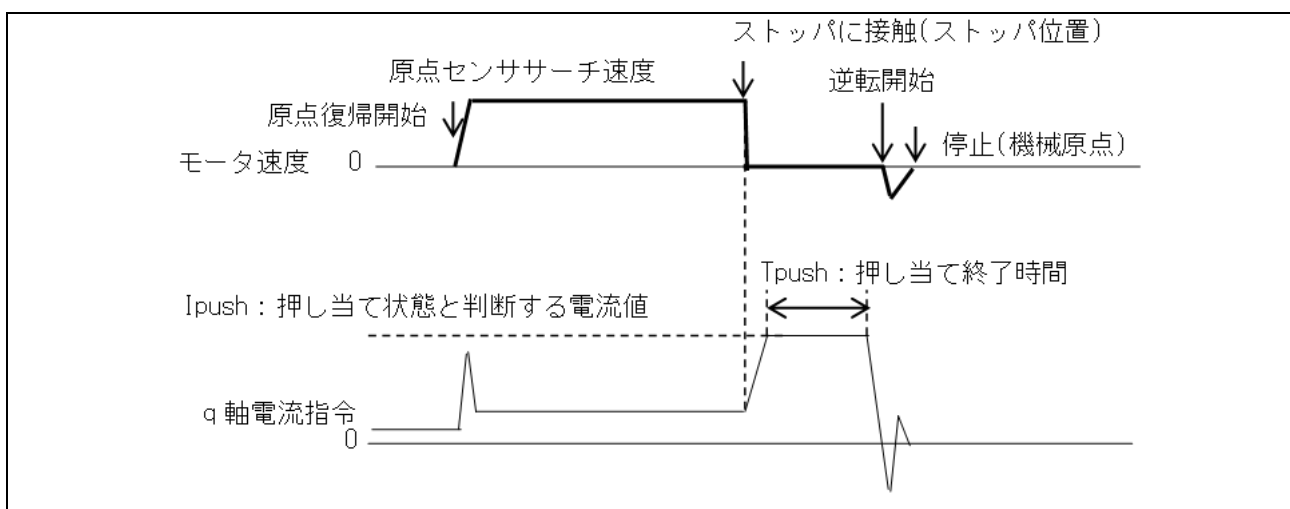


図 3-2 押し当て原点復帰動作例

原点復帰動作を開始すると、モータは指定方向へ指定された速度で駆動します。原点に設置されているストッパに当たるとモータはそれ以上の駆動が出来ず、出力トルクを増加させるため q 軸電流が上昇します。原点検出電流値に q 軸電流が到達すると、原点を検出したと判断し、そこから押し当て時間を待ちます。押し当て時間を経過したら、戻し量だけモータを逆回転させ原点復帰動作を終了します。

3.2 サーボ制御ソフトウェア関数仕様

3.2.1 イナーシャ推定モジュール API

イナーシャ推定モジュールの API 一覧を以下に示します。

表 3-1 イナーシャ推定モジュール API 一覧

API 関数	引数	処理概要
RM_MOTOR_INERTIA_ESTIMATE_Open	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ motor_inertia_estimate_cfg_t const * const p_cfg / イナーシャ推定モジュールコンフィグレーションデータポインタ	イナーシャ推定モジュールオープン
RM_MOTOR_INERTIA_ESTIMATE_Close	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定モジュールクローズ
RM_MOTOR_INERTIA_ESTIMATE_Start	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定開始
RM_MOTOR_INERTIA_ESTIMATE_Stop	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定停止(中止)
RM_MOTOR_INERTIA_ESTIMATE_Reset	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定処理リセット
RM_MOTOR_INERTIA_ESTIMATE_InfoGet	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ motor_inertia_estimate_info_t * const p_info / イナーシャ推定情報取得用構造体ポインタ	イナーシャ推定処理情報取得
RM_MOTOR_INERTIA_ESTIMATE_DataSet	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ motor_inertia_estimate_set_data_t * const p_set_data / イナーシャ推定情報設定用構造体ポインタ	イナーシャ推定処理必要データ設定
RM_MOTOR_INERTIA_ESTIMATE_SpeedCyclic	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定速度制御周期処理実行
RM_MOTOR_INERTIA_ESTIMATE_CurrentCyclic	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ	イナーシャ推定電流制御周期処理実行
RM_MOTOR_INERTIA_ESTIMATE_ParameterUpdate	motor_inertia_estimate_ctrl_t * const p_ctrl / イナーシャ推定モジュール変数構造体ポインタ motor_inertia_estimate_cfg_t const * const p_cfg / イナーシャ推定モジュールコンフィグレーションデータポインタ	イナーシャ推定処理コンフィグレーションデータ更新

3.2.2 原点復帰モジュール API

原点復帰処理モジュールの API 一覧を以下に示します。

表 3-2 原点復帰処理モジュール API 一覧

API 関数	引数	処理概要
RM_MOTOR_RETURN_ORIGIN_Open	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ motor_return_origin_cfg_t const * const p_cfg / 原点復帰処理モジュールコンフィグレーションデータポインタ	原点復帰処理 モジュールオープン
RM_MOTOR_RETURN_ORIGIN_Close	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ	原点復帰処理 モジュールクローズ
RM_MOTOR_RETURN_ORIGIN_Start	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ	原点復帰処理開始
RM_MOTOR_RETURN_ORIGIN_Stop	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ	原点復帰処理停止(中止)
RM_MOTOR_RETURN_ORIGIN_Reset	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ	原点復帰処理リセット
RM_MOTOR_RETURN_ORIGIN_InfoGet	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ motor_return_origin_info_t * const p_info / 原点復帰処理情報取得用構造体ポインタ	原点復帰処理情報 取得
RM_MOTOR_RETURN_ORIGIN_DataSet	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ motor_return_origin_set_data_t * const p_set_data / 原点復帰処理情報設定用構造体ポインタ	原点復帰処理必要データ 設定
RM_MOTOR_RETURN_ORIGIN_SpeedCyclic	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ	原点復帰処理速度制御 周期処理実行
RM_MOTOR_RETURN_ORIGIN_ParameterUpdate	motor_return_origin_ctrl_t * const p_ctrl / 原点復帰処理モジュール変数構造体ポインタ motor_return_origin_cfg_t const * const p_cfg / 原点復帰処理モジュールコンフィグレーションデータポインタ	原点復帰処理 コンフィグレーション データ更新

3.3 Contents of control

3.3.1 Configuration Option

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。

表 3-3 構成オプション (rm_motor_inertia_estimate.h)

Configuration Options (rm_motor_inertia_estimate.h)	
オプション名	内容
Moved position distance to measure (degree)	推定時のロータ移動角度[度] システムの最大移動量を超えないように設定してください。
Maximum speed (rpm)	推定時のモータ最大回転速度[rpm]
Acceleration time	推定時速度の加速値[rpm/s]
Motor inertia	初期設定ロータイナーシャ
Low threshold to judge speed	最高速度に対する測定開始下側速度の比率
High threshold to judge speed	最高速度に対する測定開始上側速度の比率
Time to wait moving stability (sec)	位置制御安定を待つ時間[sec]
Cyclic period of current control (sec)	電流制御周期[sec]
Cyclic period of speed control (sec)	速度制御周期[sec]
Motor pole pairs	極対数
Motor magnet flux (Wb)	磁束 [Wb]
Interval time	位置制御の安定時間

表 3-4 構成オプション (rm_motor_return_origin.h)

Configuration Options (rm_motor_return_origin.h)	
オプション名	内容
Search speed (rpm)	原点探索時の最高速度[rpm]
Acceleration of speed (rpm/sec)	戻し時に使用される加速度[rpm/s]
Cyclic period of speed control (sec)	速度制御周期[sec]
Maximum current (A)	押し当て時最大電流[A]
Percentage of current to judge pushing (%)	押し当て発生判断電流比(最大電流に対しての比率)[%]
Pushing time (sec)	押し当て時間[sec]
Degree to judge none stopper	原点未検出判断角度[度]
Degree to return	原点からの戻し角度[度]
Mechanical gear ratio	外部ギア比

4. Renesas Motor Workbench(RMW)を用いた操作

サーボ制御機能に関しては GUI ツールである Renesas Motor Workbench(以下 RMW)を用いて簡単に処理を実施することが出来ます。

本章ではその操作方法に関して説明します。

4.1 プロジェクトのインポート

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

1. File → Import

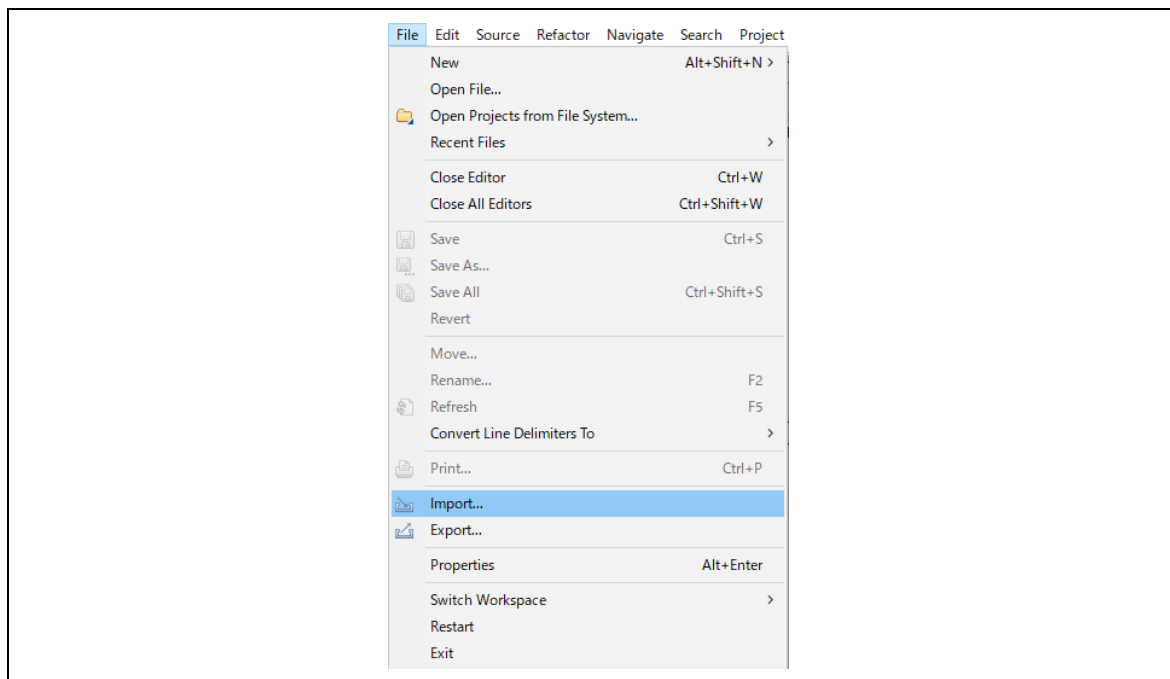


図 4-1 ファイルメニュー

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

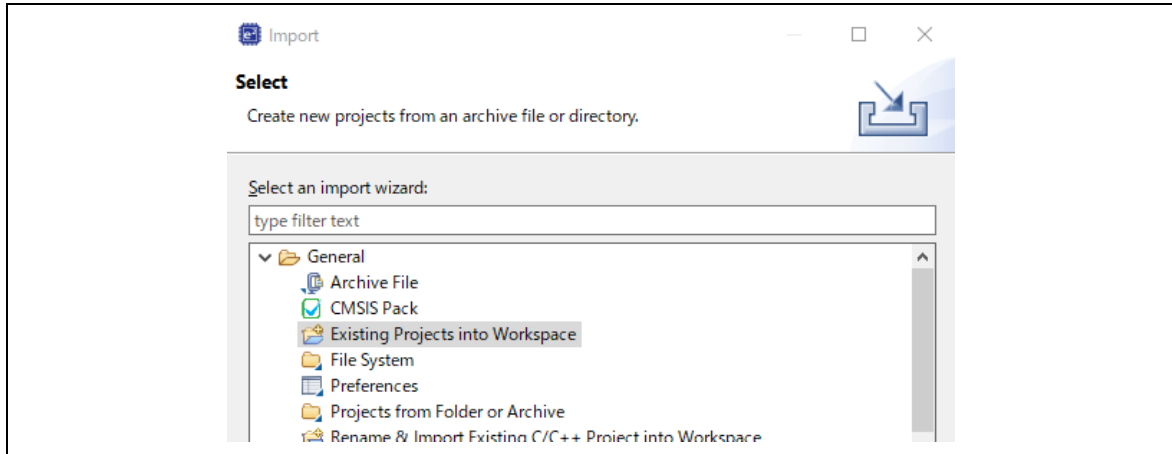


図 4-2 インポートメニュー

- 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。

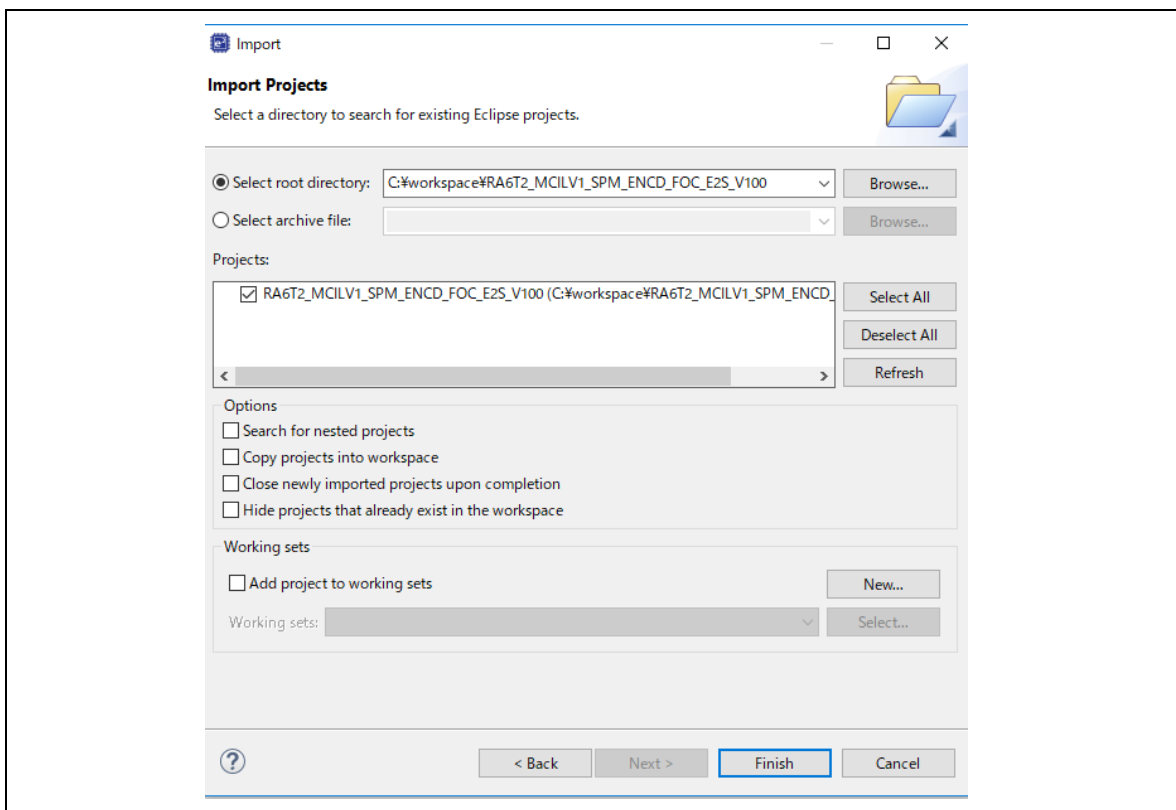


図 4-3 プロジェクトのインポート

4.2 ビルドとデバッグ

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

4.3 RMW 起動

RMW を起動すると以下の画面が表示されます。

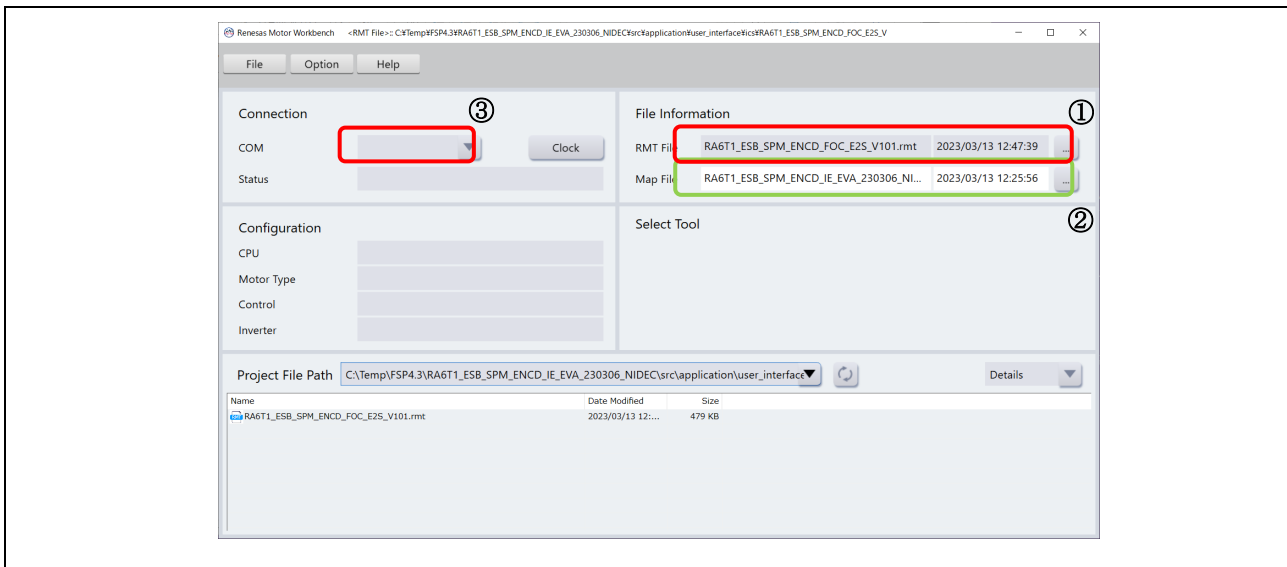


図 4-4 RMW 起動画面

最初に以下の操作を行ってください。

- ① 「File Information」⇒「RMT File」の右のアイコン「...」をクリックし、対象 RMT ファイルの選択を行います。プロジェクトの「src/application/user_interface/ics」の下にある「* * *.rmt」ファイルを選択してください。
- ② 次に「Map File」の右のアイコン「...」をクリックし、対象 MAP ファイルの選択を行います。同梱 Project の RMT ファイルと同じフォルダの「* * *.map」ファイルを選択してください。
変数反映ウィンドウがポップアップしますので、「Set」をクリックして反映します。
- ③ 以上が終了したら COM の接続を選択します。「Connection」⇒「COM」の右の「▼」をクリックし、プルダウンメニューから接続している COM ポートを選択してください。

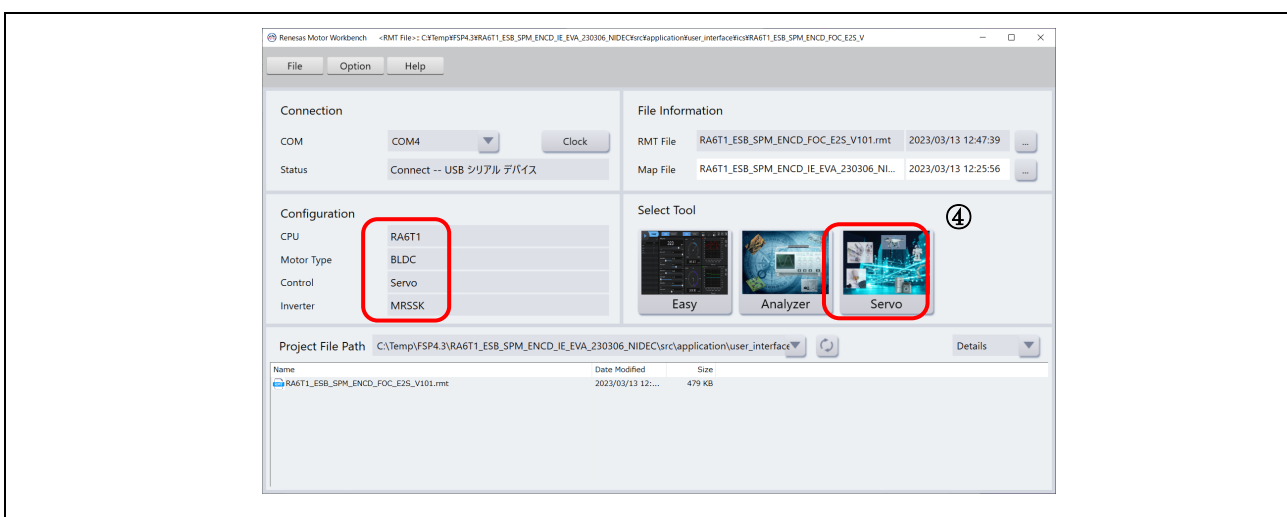


図 4-5 RMW メインウィンドウ

正しく通信が行われている場合上図の様な表示になります。

④ 「Select Tool」 ⇒ 「Servo」 アイコンをクリックします。

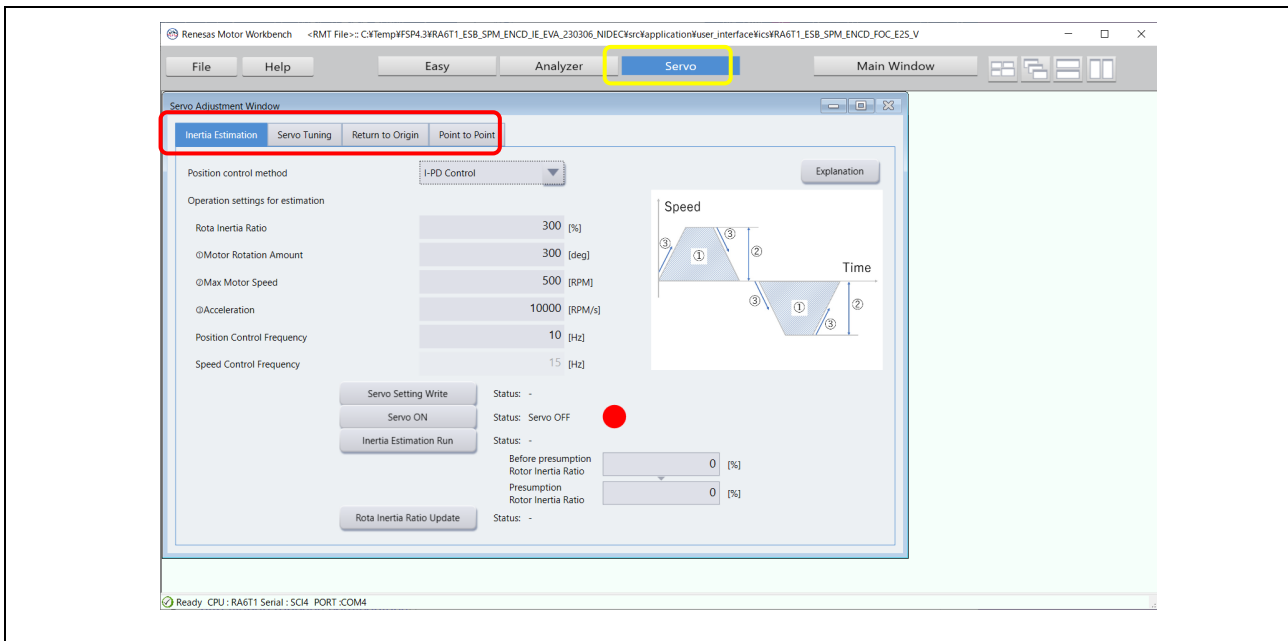


図 4-6 サーボ機能画面（イナーシャ推定機能選択時）

正しく動作している場合、上記の様な表示に切り替わります。

この画面は画面上部（黄色枠）に表記されている様に「Servo（サーボ機能）」画面となります。
この画面では赤枠で囲まれた TAB を選択することにより各機能を動作させることができます。

- ・ イナーシャ推定機能（Inertia Estimation）
- ・ 原点復帰機能（Return to Origin）
- ・ サーボ調整機能（Servo Tuning）
- ・ 位置動作確認機能（Point to Point）

以下に各機能の操作方法を説明します。

4.4 イナーシャ推定機能操作

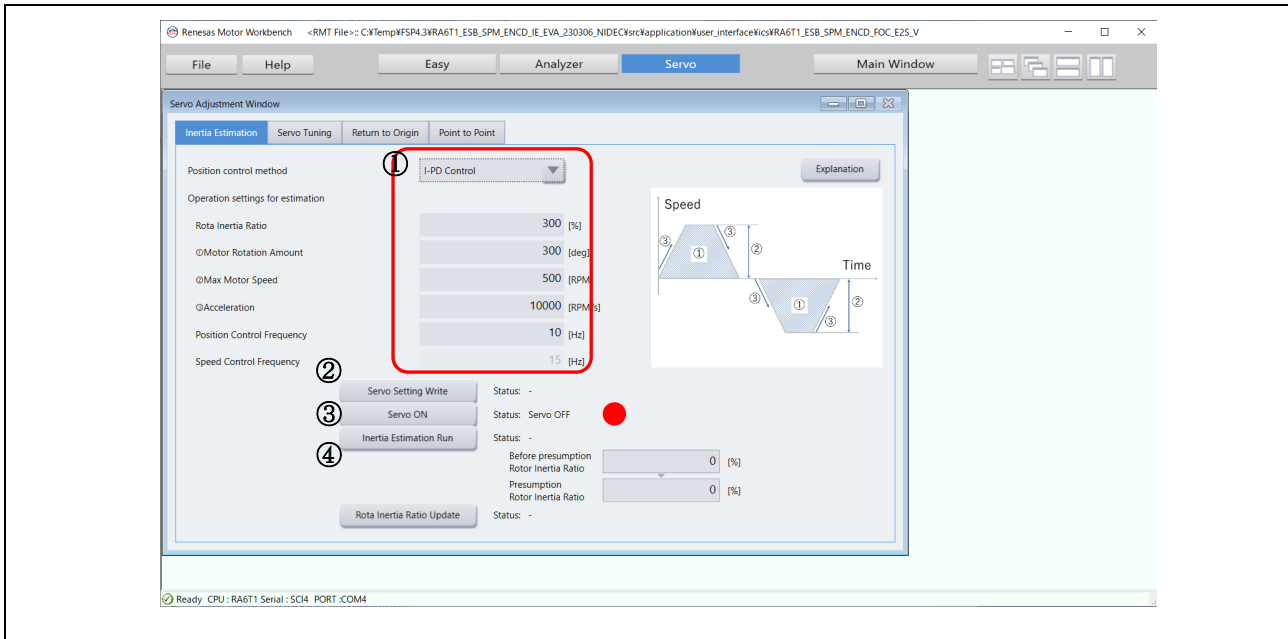


図 4-7 イナーシャ推定機能操作画面

この画面では以下の手順でイナーシャ推定動作を行うことができます。

- ① 推定を行う際の各種パラメータ設定を行います。

画面の初期値には 2.1 動作確認環境に記載の環境で動作確認した際の値が表示されていますので、参考にしてください。

表 4-1 イナーシャ推定パラメータ

名称	説明
Position control method	位置制御方式を指定します。
Rota Inertia Ratio	ロータイナーシャ比を設定します。計算値がある場合はその値を入力してください。ない場合は、初期値 300%のままにしてください。
Motor Rotation Amount	モータの移動量を設定します。 メカ機構に接続した状態で許容回転範囲を超えた移動量を設定すると、装置の故障の恐れがあります。十分注意してご使用ください。
Max Motor Speed	モータの最大速度を設定します。実際に駆動可能な値を設定してください。
Acceleration	加速度を設定します。実際に駆動可能な値を設定してください。
Position Control Frequency	位置制御系固有周波数を設定します。基本的に初期値 10Hz のままで問題ありません。
Speed Control Frequency	速度制御系固有周波数を設定します。 「Position control method」が「PID Control」の場合に入力ができます。 基本的に初期値 15Hz のままで問題ありません。

- ② 設定したパラメータを反映する為に「Servo Setting Write」ボタンをクリックします。変更をしていない場合も初期設定を反映させるために押下してください。
- ③ 「Servo ON」ボタンを押下します。問題が無ければ Status が「Servo ON」状態に成り、右のインジケータが赤から緑に変わります。モータは位置制御状態に入ります。また、ボタンは「Servo OFF」表示に切り替わります。

- ④ 「Inertia Estimation Run」 ボタンを押下します。イナーシャ推定が開始され、ボタン表示が「Inertia Estimation Stop」表示に変わります。
問題無くイナーシャ推定が行われると自動的にボタンが「Inertia Estimation Run」状態に復帰します。
エラーが発生した場合はエラーウィンドウがポップアップします。

他の画面に遷移する際には「Servo Stop」ボタンをクリックして「Servo ON」状態（位置制御状態）を解除してください。

4.5 サーボ調整機能操作

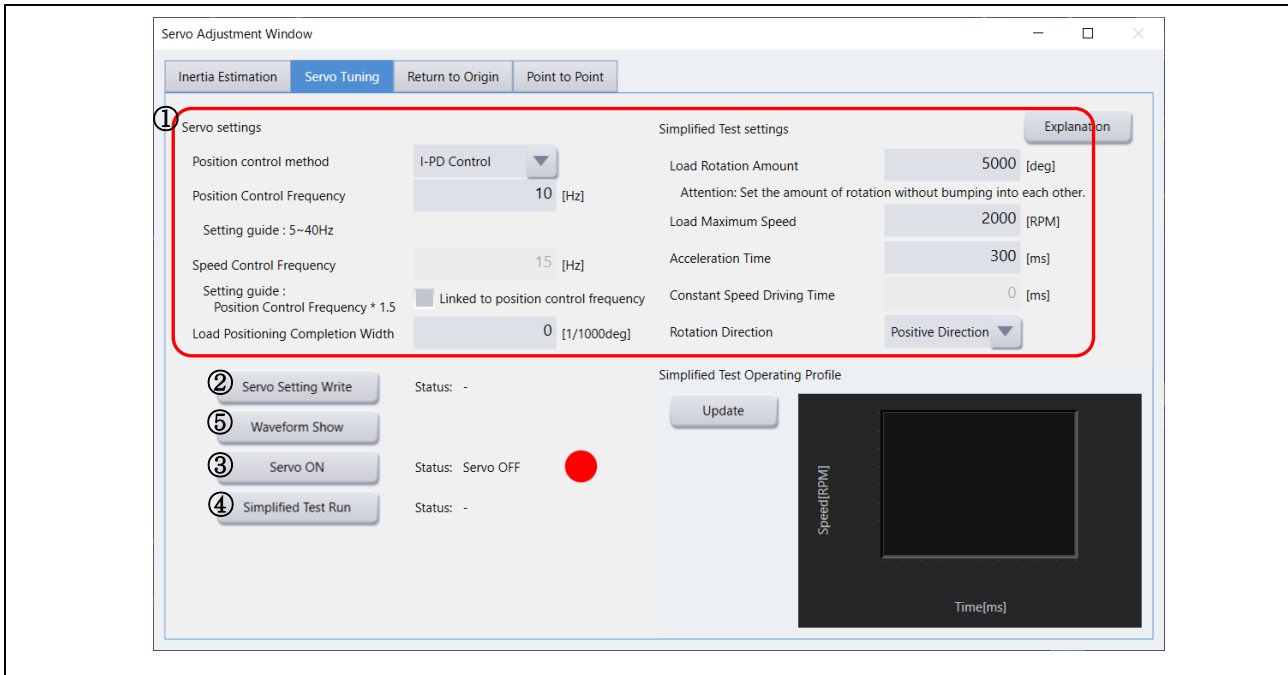


図 4-8 サーボ調整機能操作画面

この画面では以下の手順でサーボ調整機能動作を行うことができます。

- ① 動作時の各種パラメータ設定を行います。

画面の初期値には 2.1 動作確認環境に記載の環境で動作確認した際の値が表示されていますので、参考にしてください。

表 4-2 サーボ調整パラメータ

名称	説明
Position control method	位置制御方式を指定します。
Position Control Frequency	位置制御系固有周波数を設定します。
Speed Control Frequency	速度制御系固有周波数を設定します。 「Position control method」が「PID Control」の場合に入力ができます。
Linked to position control frequency	チェックを入れると位置制御系固有周波数を 1.5 倍にした値を速度制御系固有周波数に自動設定します。No.3 に設定した値は上書きされます。
Load Positioning Completion	負荷の位置決め完了幅を設定します。
Load Rotation Amount	負荷の移動量を設定します。実際に駆動可能な値を設定してください。
Load Maximum Speed	負荷の最大速度を設定します。実際に駆動可能な値を設定してください。
Acceleration Time	加速時間を設定します。実際に駆動可能な値を設定してください。
Rotation Direction	回転方向を指定します。

- ② 設定したパラメータを反映する為に「Servo Setting Write」ボタンをクリックします。変更をしていない場合も初期設定を反映させるために押下してください。
- ③ 「Servo ON」ボタンを押下します。問題が無ければ Status が「Servo ON」状態に成り、右のインジケータが赤から緑に変わります。モータは位置制御状態に入ります。また、ボタンは「Servo OFF」表示に切り替わります。

- ④ 「Simplified Test Run」 ボタンを押下します。サーボ調整動作が開始され、ボタン表示が「Simplified Test Stop」表示に変わります。

問題無くサーボ調整動作が行われると自動的にボタンが「Simplified Test Run」状態に復帰します。

エラーが発生した場合はエラーウィンドウがポップアップします。

- ⑤ 「Waveform Show」 ボタンを押下すると、測定された波形表示がポップアップします。

他の画面に遷移する際には「Servo Stop」 ボタンをクリックして「Servo ON」状態（位置制御状態）を解除してください。

4.6 原点復帰機能操作

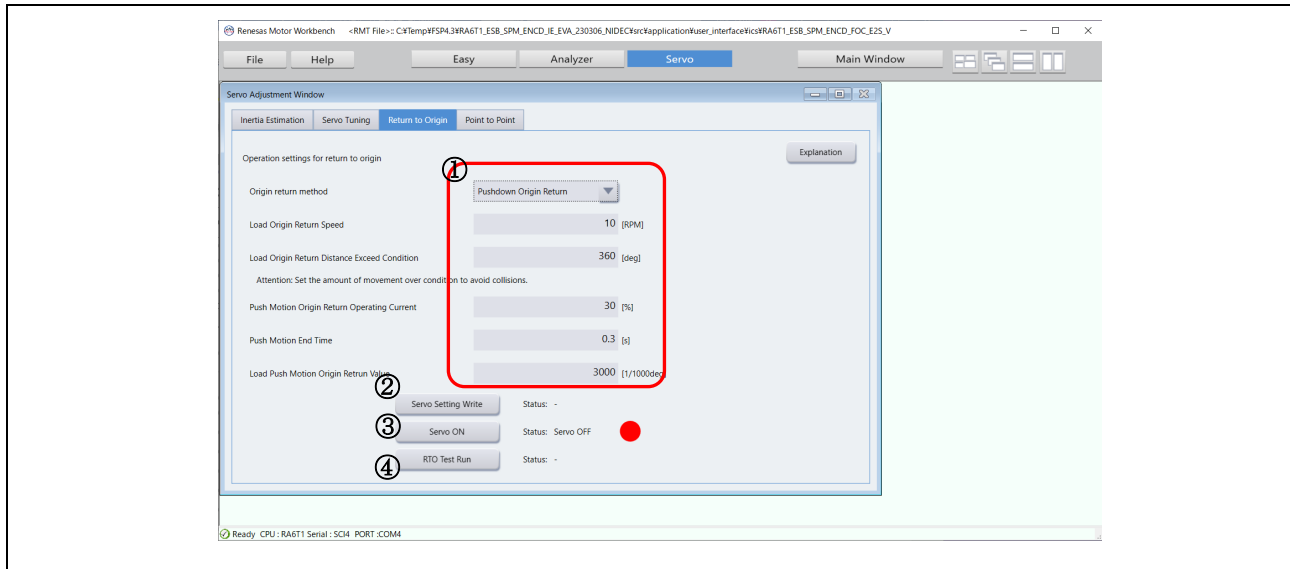


図 4-9 原点復帰機能操作画面

この画面では以下の手順で原点復帰動作を行うことができます。

① 動作時の各種パラメータ設定を行います。

画面の初期値には 2.1 動作確認環境に記載の環境で動作確認した際の値が表示されていますので、参考にしてください。

表 4-3 原点復帰パラメータ

名称	説明
Load Origin Return Speed	負荷の原点復帰速度を設定します。
Load Origin Return Distance Exceed Condition	原点探索時の探索不可判定の角度を設定します。
Push Motion Origin Return Operating Current	押し当て原点復帰運転電流を、定格電流からのパーセントで設定します。
Push Motion End Time	押し当て終了時間を設定します。
Load Push Motion Origin Return Value	負荷の押し当て原点復帰戻り値を設定します。

- ② 設定したパラメータを反映する為に「Servo Setting Write」ボタンをクリックします。変更をしていない場合も初期設定を反映させるために押下してください。
- ③ 「Servo ON」ボタンを押下します。問題が無ければ Status が「Servo ON」状態に成り、右のインジケータが赤から緑に変わります。モータは位置制御状態に入ります。また、ボタンは「Servo OFF」表示に切り替わります。
- ④ 「RTO Test Run」ボタンを押下します。原点復帰動作が開始され、ボタン表示が「RTO Test Stop」表示に変わります。
問題無く原点復帰動作が行われると自動的にボタンが「RTO Test Run」状態に復帰します。
エラーが発生した場合はエラーウィンドウがポップアップします。

他の画面に遷移する際には「Servo Stop」ボタンをクリックして「Servo ON」状態（位置制御状態）を解除してください。

4.7 位置動作確認機能（Point to Point）操作

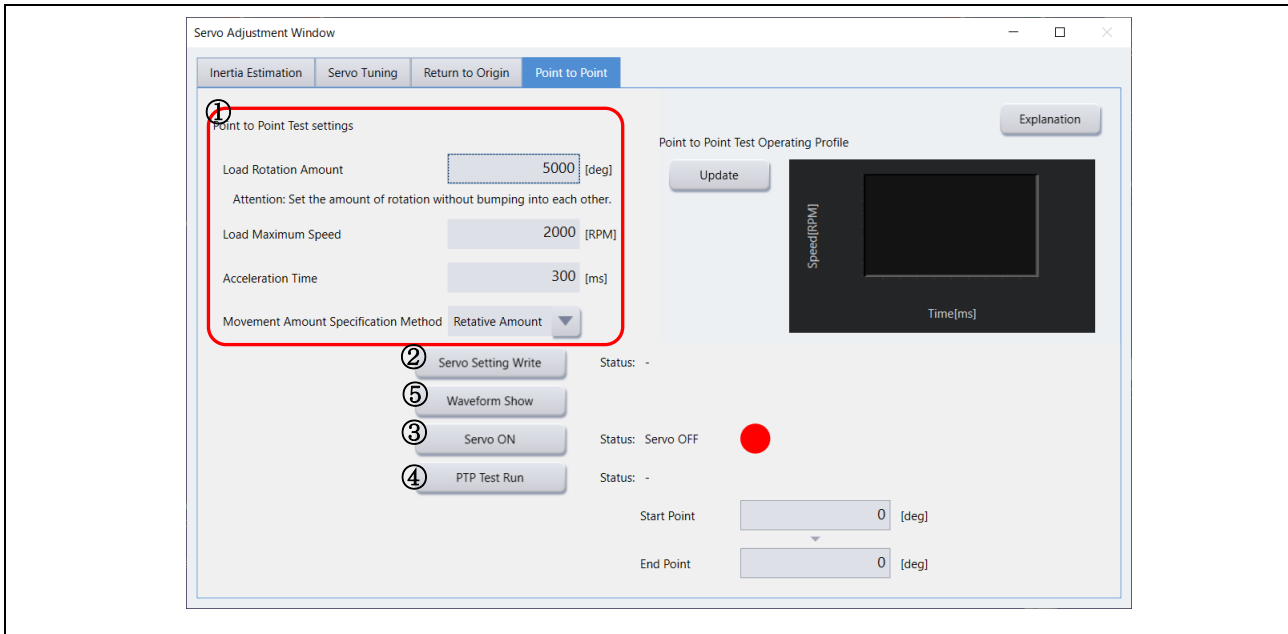


図 4-10 位置動作確認機能操作画面

この画面では以下の手順で位置動作確認機能動作を行うことができます。

- ① 動作時の各種パラメータ設定を行います。

画面の初期値には 2.1 動作確認環境に記載の環境で動作確認した際の値が表示されていますので、参考にしてください。

表 4-4 位置動作確認パラメータ

名称	説明
Load Rotation Amount	負荷の移動量を設定します。実際に駆動可能な値を設定してください。
Load Maximum Speed	負荷の最大速度を設定します。実際に駆動可能な値を設定してください。
Acceleration Time	加速時間を設定します。実際に駆動可能な値を設定してください。
Movement Amount Specification Method	移動量指定方法を指定します。

- ② 設定したパラメータを反映する為に「Servo Setting Write」ボタンをクリックします。変更をしていない場合も初期設定を反映させるために押下してください。
- ③ 「Servo ON」ボタンを押下します。問題が無ければ Status が「Servo ON」状態に成り、右のインジケータが赤から緑に変わります。モータは位置制御状態に入ります。また、ボタンは「Servo OFF」表示に切り替わります。
- ④ 「PTP Test Run」ボタンを押下します。位置動作確認動作が開始され、ボタン表示が「PTP Test Stop」表示に変わります。

問題無く位置動作確認動作が行われると自動的にボタンが「PTP Test Run」状態に復帰します。

エラーが発生した場合はエラーウィンドウがポップアップします。

- ⑤ 「Waveform Show」ボタンを押下すると、測定された波形表示がポップアップします。

他の画面に遷移する際には「Servo Stop」ボタンをクリックして「Servo ON」状態（位置制御状態）を解除してください。

5. 参考ドキュメント

Renesas Motor Workbench ユーザーズマニュアル (R21UZ0004)
Renesas Motor Workbench クイックスタートガイド (R21QS0011)
RA6T1 グループ ユーザーズマニュアル ハードウェア編 (R01UH0897)
RA6T2 グループ ユーザーズマニュアル ハードウェア編 (R01UH0951)
RA6T1 永久磁石同期モータのエンコーダ利用ベクトル制御 (R01AN5923)
RA6T2 永久磁石同期モータのエンコーダ利用ベクトル制御 (R01AN6208)
RA6T2 永久磁石同期モータの誘導センサ使用ベクトル制御 (R01AN6467)
永久磁石同期モータのエンコーダベクトル制御 アルゴリズム編 (R01AN3789)
Motor Control Evaluation System for RA Family (R12UZ0078)
RA6T1 CPU カード ユーザーズマニュアル (R12UZ0077)
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	2023.5.30	—	初版発行
1.01	2024.8.30	1	対象ソフトウェアを更新

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

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、変更、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、変更、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
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なります。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

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。