RL78/F24

R01AN6308JJ0110
Rev.1.10
2023.06.30

MCU によるブラシレス DC モータのセンサレス 120 度通電制御編

要旨

本アプリケーションノートは RL78/F24 の機能を使用して三相ブラシレス DC モータをセンサレス 120 度通電方式で駆動するサンプルプログラムについて説明することを目的としています。

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うえでご使用下さい。

動作確認デバイス

サンプルプログラムの動作確認は下記のデバイスで行っております。

・RL78/F24 (R7F124FGJ)

目次

1. 概説 4
1.1 システムの利用 4
1.2 開発環境 4

2. システム概要 5
2.1 ハードウェア構成 5
2.2 ハードウェア仕様 6
2.2.1 端子インタフェース 6
2.2.2 周辺機能 6
2.3 ソフトウェア構成 7
2.3.1 ソフトウェア・ファイル構成 7
2.3.2 モジュール構成 8
2.4 ソフトウェア仕様 9

3. モータ制御方法 10
3.1 BLDC モータのセンサレス 120 度通電制御 10
3.2 ゼロクロスの検出方法 13
3.3 始動方法 15
3.4 位置推定演算 16
3.5 速度制御 17
3.6 角度の表現 19

4. 使用周辺機能説明 20
4.1 A/D コンバータ 20
4.2 タイマ・アレイ・ユニット TAU 機能 23
4.3 タイマ RDe 機能 24
4.3.1 PWM デューティ指令値から PWM のデューティ設定へ 25

5. 制御プログラム説明 26
5.1 制御ブロック図 26
5.2 制御内容 28
5.2.1 モータ起動／停止／変速 28
5.2.1.1 経過時間による制御 28
5.2.1.2 PWM 信号入力による制御 28
5.2.2 インバータ・バス電圧 28
5.2.3 モータ三相電圧 29
5.2.4 回転速度演算 29
5.2.5 速度 PI 制御 29
5.2.6 モータ電流 30
5.2.7 状態遷移 30
5.2.8 システム保護機能 31
5.3 システム･リソース 34
5.3.1 割り込み 34
5.3.2 ポート機能 35
5.3.3 PWM 出力部 35

5.4 関数仕様 36
5.5 変数一覧 41
5.6 マクロ定義 44
5.7 制御フロー 47
5.7.1 メイン処理 47
5.7.2 1 [ms]インターバル・タイマ割り込み処理 48
5.7.3 ユーザ・インタフェース 49
5.7.4 キャリア周期割り込み処理 50
5.7.5 ゼロクロス検出処理 51
5.7.6 A/D 変換完了割り込み処理 52

改訂記録 53

1. 概説

本アプリケーションノートは、RL78/F24 を使用し、ブラシレス DC モータ（以降、BLDC モータ）のセンサレス 120 度通電方式による速度制御の例を説明するものです。

1.1 システムの利用

本システム（サンプルプログラム）は、RL78/F24 マイコン搭載ボード（RTK7F124FGS00000BJ）および、BLDC モータ（TG-55N-KA）を使用し、120 度通電制御を実現しています。

備考：

- TG-55N-KA モータは、ツカサ電工株式会社の製品です。
  ツカサ電工株式会社（https://www.tsukasa-d.co.jp/ ）

1.2 開発環境

本アプリケーションノートが対象とする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 ソフトウェア開発環境

Renesas CS+		
	IDE Version	CS+ for CC E8.07.00 [01 Dec 2021]
	Compiler	CC-RL E1.11.00
IAR		
	IDE Version	IAR Embedded Workbench IDE 8.5.2.7561 (8.5.2.7561)
	Compiler	IAR C/C++ Compiler for Renesas RL78 4.21.3.2447 (4.21.3.2447)

表 1-2 ハードウェア環境

On-chip Debugging Emulator	E2 emulator Lite
	E2 emulator
MCU Part Name	RL78/F24 (R7F124FGJ)
Inverter Board	RTK7F124FGS00000BJ
BLDC Motor	PMSM (TG-55N-KA)

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 ハードウェア構成

ハードウェア構成を次に示します。

図 2-1 ハードウェア構成図

2.2　ハードウェア仕様

2.2.1　端子インタフェース

本システムの RL78/F24 マイコンの端子インタフェース一覧を表 2-1 に示します。

表 2-1 端子インタフェース一覧

端子名	機　能
P86 / ANI8	U 相電圧測定
P87 / ANI9	V 相電圧測定
P90 / ANI10	W 相電圧測定
P85 / ANI5	V_{dc} 電圧測定
P91 / ANI11	I_{dc} 電流測定
P84 / ANI4	モータコイルエンド温度測定
P125 / ANI24	基板温度測定
P11 / TRDIOB0	ポート or PWM 出力 (U_P)
P12 / TRDIOD0	ポート or PWM 出力 (U_N)
P15 / TRDIOA1	ポート or PWM 出力 (V_P)
P17 / TRDIOB1	ポート or PWM 出力 (W_P)
P16 / TRDIOC1	ポート or PWM 出力 (V_N)
P30 / TRDIOD1	ポート or PWM 出力 (W_N)
P140	ゲートドライバ有効化／無効化信号出力
P83 / IVCMP01	過電流検出用コンパレータ入力
P00 / TI05	PWM デューティ指令値入力

2.2.2　周辺機能

本システムで使用する周辺機能一覧を表 2-2 に示します。

詳細は「4. 使用周辺機能説明」を参照してください。

表 2-2 周辺機能一覧

周辺機能	用途
A/D コンバータ	● U、V、W 相電圧測定 ● DC バス電圧測定 ● モータ電流測定 ● 基板温度測定 ● モータコイルエンド温度測定
タイマ RDe	● 拡張相補 PWM モードを使用した PWM 出力（正相 3 本、逆相 3 本） ● PWM 出力強制遮断（PWMOPA を使用）
出力ポート	● モータコントロール信号出力 ● ゲートドライバ有効化／無効化信号出力
タイマ・アレイ・ユニット	● 1 [ms]インターバル・タイマ ● PWM 信号指令値入力
D/A コンバータ	内蔵コンパレータのしきい値出力
コンパレータ	過電流検出用

2.3　ソフトウェア構成

2.3.1　ソフトウェア・ファイル構成

サンプルプログラムのフォルダとファイル構成を以下に記します。

表 2-3 サンプルプログラムのフォルダとファイル構成

RL78F24_120_ADC_V105		
prj	RL78F24_120_ADC.mtpj	CS+プロジェクト・ファイル（CS+用）
	RL78F24_120_ADC.dep	IAR ワークスペース・ファイル一式（IAR 用）
	RL78F24_120_ADC.ewd	
	RL78F24_120_ADC.ewp	
	RL78F24_120_ADC.ewt	
	RL78F24_120_ADC.eww	
inc	iodefine.h	SFR 定義ファイル（CC-RL 用）
	mtr_cpu_setting.h	CPU 依存処理部ヘッダ
	mtr_fix_calc.h	固定小数点演算処理用ヘッダ
	mtr_function.h	共通関数用ヘッダ
	mtr_interrupt_bemf.h	モータ制御割り込み用ヘッダ
	mtr_inv_setting.h	インバータ依存処理部ヘッダ
	r_compiler_common.h	コンパイラ依存処理部ヘッダ
	r_mtr_sequence_api.h	状態遷移制御用ヘッダ
	r_mtr_user_control_api.h	ユーザ制御 API 用ヘッダ
	r_mtr_user_control_cfg.h	ユーザ制御コンフィグ用ヘッダ
	r_typedefs.h	型定義用ヘッダ
src	mtr_main.c	メイン関数
	mtr_cpu_setting.c	CPU 依存処理
	mtr_fix_calc.c	固定小数点演算処理
	mtr_function.c	共通関数処理
	mtr_interrupt_bemf.c	モータ制御割り込み処理
	mtr_inv_setting.c	インバータ依存処理
	r_mtr_sequence_api.c	状態遷移制御処理
	r_mtr_user_control_api.c	ユーザ制御処理
	mcu_debug_option_for_iar.c	オプション・バイト設定（IAR コンパイラ用）
asm	cstart.asm	スタートアップ・ルーチン（CC-RL 用）
	hwinit.asm	ハードウェア初期化（CC-RL 用）
	stkinit.asm	スタック初期化（CC-RL 用）
	f24opt.asm	オプション・バイト補足（CC-RL 用）

2.3.2 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以下に記します。

図 2-2 サンプルプログラムのモジュール構成

2.4 ソフトウェア仕様

本システムのソフトウェアの基本仕様を表 2-4 に示します。

表 2-4 ソフトウェア基本仕様

項目	内容
制御方式	120 度通電方式
モータ回転開始／停止／変速	以下 2 つのうち、どちらか一方を選択して動作する ● リセット解除後以下のタイミングで制御回転状態を遷移する – 3 [s]後、1000 [rpm]を回転速度指令値としてモータ回転開始 – 上記動作から 10 [s]毎、3000 [rpm]に到達するまで回転速度指令値を 500 [rpm]増加 – 3000 [rpm]到達から 10 [s]毎、回転速度指令値を 500 [rpm]減少 – 0 [rpm]到達後、モータ停止 ● リセット解除後、PWM 信号入力により回転速度指令値を制御する – 入力する PWM 信号は 10 [Hz] ～ 1k [Hz]とする – duty が 100%のとき、回転速度指令値を 3000 [rpm]とする – duty が 0%のとき、回転速度指令値を 0 [rpm]とする
回転子磁極位置検出	A/D コンバータを用い、誘起電圧値により位置を検出する（60 度毎）
キャリア周波数（PWM）	20 [kHz]
制御周期	● キャリア周期毎に誘起電圧からゼロクロス判定を行う ● パターン切り替え時、PWM デューティ設定と通電パターンの決定を行う
回転速度制御範囲	CW/CCW 共に 500 [rpm] ～ 3000 [rpm]
回転速度演算	パターン切り替え時、電気角 1 回転分のキャリアパルス数から回転数を算出する
速度制御（速度 PI 制御）	速度指令値設定関数により速度指令値を変更し、PI 制御（10 [ms]周期）により速度制御を行う
保護停止処理	以下 7 つのうち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 – モータ駆動電圧が 28.0 [V]を超過もしくは 8.0 [V]未満（キャリア割り込み毎に監視） – バス電流が 10.0 [A]を超過（キャリア割り込み毎に監視） – 回転速度が 10000 [rpm]を超過（1 [ms]毎に監視） – センサレス駆動時、ゼロクロス検出が 200 [ms]間未発生 – ハードウェアにより過電流を検出した場合 – 基板温度が 125 [°C]を超過（1 [ms]毎に監視） – モータコイルエンド温度が 180 [°C]を超過（1 [ms]毎に監視）

3. モータ制御方法

本章ではサンプルプログラムで用いる、BLDC モータのセンサレス 120 度通電制御と速度制御について説明します。

3.1 BLDC モータのセンサレス 120 度通電制御

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

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

例えば、図 3-1 のようにコイルに磁石を近づけた場合を考えます。この場合、コイルを貫く磁束は増加するので、コイルは磁束の増加を妨げるために図の方向に電流を流す起電力を発生させます。（右ねじの法則により、磁石の磁束と逆向きの磁束が発生します。）

図 3-1 コイルと磁石による誘起電圧

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

$$E_m = \frac{d}{dt}\varphi_m \cdots \quad \text{式 (1)}$$

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

図 3-2 回転する永久磁石同期モータにおける誘起電圧

図 3-3 は、U 相における鎖交磁束の変化を表したもので、縦軸に鎖交磁束の大きさ、横軸に永久磁石の位相を表しています。また、永久磁石の N 極が U 相のコイルを向いている位置を θ=0 としています。

図 3-3 鎖交磁束の変化

U 相の鎖交磁束は余弦波状に変化します。

さらに、V 相、W 相についても同様に考えた場合、V 相、W 相は U 相からそれぞれ 2π/3、4π/3 位相がずれているので、各相の鎖交磁束は以下の式で表すことができます。

$$\varphi_u = \varphi_m \cos\theta$$
$$\varphi_v = \varphi_m \cos(\theta - \frac{2}{3}\pi)$$
$$\varphi_w = \varphi_m \cos(\theta - \frac{4}{3}\pi)$$

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

$$E_u = \frac{d}{dt}\varphi_u = \frac{d}{dt}\varphi_m \cos\theta = -\omega\varphi_m \sin\theta = \omega\varphi_m \cos(\theta + \frac{\pi}{2})$$
$$E_v = \frac{d}{dt}\varphi_v = \frac{d}{dt}\varphi_m \cos(\theta - \frac{2}{3}\pi) = -\omega\varphi_m \sin(\theta - \frac{2}{3}\pi) = \omega\varphi_m \cos(\theta - \frac{\pi}{6})$$
$$E_w = \frac{d}{dt}\varphi_w = \frac{d}{dt}\varphi_m \cos(\theta - \frac{4}{3}\pi) = -\omega\varphi_m \sin(\theta - \frac{4}{3}\pi) = \omega\varphi_m \cos(\theta - \frac{5}{6}\pi)$$

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

図 3-4 誘起電圧のゼロクロス

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

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

三相モータでは、このゼロクロスがモータ 1 回転（電気角）する間で各相 2 回、合計 6 回発生します。つまり、このゼロクロスを検出することで 60 度毎の位置検出ができ、ホールセンサと同等の分解能で位置情報を推定することができます。

図 3-5 通電パターンとゼロクロスの関係（上アームチョッピング時）

ただし、このゼロクロス検出信号をそのままホールセンサ信号と同様に扱うことはできません。

図 3-5 のように、ゼロクロス検出信号は理想的な通電パターンの切り替えタイミングから位相がπ/6 ずれたところで発生します。そのため、実際の制御ではゼロクロスを検出してからπ/6 進んだところで通電パターンを切り替えています。

3.2　ゼロクロスの検出方法

ゼロクロスの検出方法は様々なものが考案されています。ここではマイコンの A/D コンバータを用いて、誘起電圧と中点電圧の値をソフトウェアで比較してゼロクロスを検出する方式を紹介します。電圧の比較にコンパレータを使用しないため、コンパレータレス方式とも言われています。

図 3-6 コンパレータレス方式

実際に検出される誘起電圧は、通電パターン切り替え時に発生する転流電圧と他相の PWM から受ける影響を考慮しなくてはいけません。その影響は図 3-7 に表したような形で現れます。

図 3-7 転流と他相 PWM の影響概念図

本システムでは簡単なフィルタ回路の使用と以下のソフトウェア機能により、この影響を除去しています。

転流の影響の除去は具体的にはソフトウェアで以下の 2 つの対策を実施しています。

はじめにゼロクロス検出開始直後にゼロクロス検出を無効にする区間を設定し、転流の影響を除外します。このとき転流後のゼロクロス検出無効パルス数に指定した回数だけ、キャリア周期毎に行われるゼロクロス検出を無効にします。

図 3-8 転流後のゼロクロス検出無効区間設定

表 3-1 ゼロクロス検出無効区間のパラメータ

パラメータ名	初期設定値	Q 表記	概要
MTR_BEMF_DETECT_GUARD_CNT	2	–	転流後のゼロクロス検出無効パルス数

次にゼロクロス検出電圧の範囲を設定し、範囲外の誘起電圧を検出した場合はゼロクロス検出を無効とします。

図 3-9 ゼロクロス検出電圧の範囲設定

表 3-2 ゼロクロス検出電圧のパラメータ

パラメータ名	初期設定値	Q 表記	概要
USER_CFG_BEMF_DETECT_END_TH	30	–	ゼロクロス検出電圧しきい値 [AD Value]

3.3　始動方法

センサレス 120 度通電制御は、永久磁石（回転子）の磁束の変化による誘起電圧を利用し、電気角 60 度毎の磁極の位置を推定します。しかし、誘起電圧は回転することで発生するため、始動時は磁極の位置を推定することができません。

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

図 3-10 始動運転の図

3.4　位置推定演算

本システムでは、プログラム内部で推定角度情報を持ちます。回転速度からキャリア毎の増分角度を算出してキャリア割り込み毎に推定角度に積算し、推定角度がパターン切り替え角度を超過した時にパターン切り替えを実施します。また、ゼロクロス検出時は推定角度をゼロクロス検出角度に補正します。

推定角度の補正値はキャリア 2 周期分の遅れを考慮して、ゼロクロス検出角度にキャリア 2 周期分の増加角度を加算した値としています。キャリア 2 周期分の遅れの要因は以下の通りです。

- ゼロクロスの誤検出防止のため、ゼロクロス通過判定は 2 回連続でゼロクロス検出が行えた時点で確定します。そのため、誘起電圧がゼロクロスを跨いでからプログラム内部で検出完了となるまでにキャリア 1 周期分の遅れが発生します。
- パターン切り替え処理は推定角度がパターン切り替え角度を超過した後、次のキャリア割り込みで行われる仕様となります。そのため、キャリア 1 周期分の遅れが発生します。

図 3-11 仮想ホールセンサ・パターン（上アームチョッピング時）

3.5 速度制御

本システムでは、PWM キャリア割り込み処理で割り込みの発生回数をグローバルパルス番号としてカウントしています。PWM キャリア割り込みは、PWM の周期毎に発生するため、このグローバルパルス番号を用いて任意の区間の時間計測が可能です。

速度算出処理では、6 回分のパターン切り替え毎に経過時間をグローバルパルス番号から求め、この経過時間から回転速度を計算します。

本システムでは、制御を安定させるために求めた速度値に対して LPF（指数平滑移動平均）処理を実施し、平滑化した速度値を使用します。

図 3-12 モータ回転速度の演算方法

表 3-3 モータ回転速度算出のパラメータと変数

式中の変数	変数名／マクロ名	Q 表記	概要
V_{rpm}	g_mtr_rpm_speed	–	回転速度（LPF 後）[rpm]
FREQPWM	TRDE_CARRIER_FREQUENCY_HZ	–	PWM キャリア周波数 [Hz]
N	–	–	電気角 360°回転時の PWM パルス数
PPMOTOR	USER_CFG_MOTOR_POLE_PAIRS	–	モータ極対数
N_k	g_mtr_global_pulse_number	–	現在のグローバルパルス番号
N_{k-6}	g_mtr_global_pulse_number_old	–	速度算出起点でのグローバルパルス番号

また、本システムは、PI 制御を用いた速度制御を行います。速度 PI 制御は、1ms タイマ割り込み内で 10ms 毎に実施します。この実施間隔は、パラメータ g_mtr_speed_pi_interval によって変更可能です。

速度 PI による PWM 指令値の算出は、下式によって行います。

$$Nerr_k = N^* - N_{rpm}$$
$$d^*_k = d^*_{k-1} + K_P \times (Nerr_k - Nerr_{k-1}) + K_I \times err_k$$

表 3-4 速度 PI のパラメータと変数

式中の変数	変数名／マクロ名	Q 表記	概要
$Nerr_k$	–	–	速度偏差 [rpm]
N^*	g_mtr_rpm_ref	–	速度指令値 [rpm]
N_{rpm}	g_mtr_rpm_speed	–	現在の速度 [rpm]
d^*_k	g_mtr_next_duty	Q14	PWM 指令値
K_P	g_mtr_speed_kp_factor	Q14	比例項ゲイン
K_I	g_mtr_speed_ki_factor	Q14	積分項ゲイン

さらに、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 3-13 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 3-13 PWM 制御

また、本システムでは、相補 PWM チョッピング（120 度）を採用し、出力電圧及び速度の制御を行っています。図 3-14 に、相補 PWM チョッピング（120 度）時のモータ制御信号出力波形例を示します。

図 3-14 相補 PWM チョッピング（120 度）

3.6 角度の表現

本システムでは、角度を正規化した小数部 14 ビットの固定小数点数で表現しています。

表 3-5 正規化した角度表

度数法	弧度法	正規化した角度値 (Q14)
0 (360)	0	0
60	1.047197551	2731
120	2.094395102	5462
180	3.141592654	8192
240	4.188790205	10923
300	5.235987756	13654
359	6.265732015	16338

4. 使用周辺機能説明

本システムで使用する周辺機能を以下に説明します。

- A/D コンバータ
- タイマ・アレイ・ユニット TAU
- タイマ RDe

4.1 A/D コンバータ

A/D コンバータは、アナログ入力をデジタル値に変換します。対象マイコン（RL78/F24 48 ピン製品）では、12 ビットの A/D コンバータを 1 回路搭載しています。変換チャネルを制御することで最大 19 チャネルのアナログ入力をデジタル値に変換できます。

本システムでは、A/D コンバータを表 4-1, 表 4-2 のように使用しています。

表 4-1 A/D コンバータの使用内容（INTTM01 同期取得）

チャネル	項目	A/D 変換値の 1 ビットあたりの物理量
ANI4	モータコイルエンド温度測定	表 4-4 を参照
ANI24	基板温度測定	表 4-5 を参照

備考 INTTM01: タイマ・アレイ・ユニット 01 の割り込み要求

表 4-2 A/D コンバータの使用内容（INTTRD_ADTRG 同期取得）

チャネル	項目	A/D 変換値の 1 ビットあたりの物理量
ANI5	V_{dc} 電源電圧測定	65.0 [V] / 4095 = 0.0159 [V]
ANI8	U 相電圧測定	25.0 [V] / 4095 = 0.0061 [V]
ANI9	V 相電圧測定	25.0 [V] / 4095 = 0.0061 [V]
ANI10	W 相電圧測定	25.0 [V] / 4095 = 0.0061 [V]
ANI11	I_{dc} 電流測定	50.0 [A] / 4095 = 0.0122 [A]

備考 INTTRD_ADTRG: タイマ RDe A/D 変換開始トリガ

表 4-3 A/D 変換対象一覧

取得対象	変数名	Q 表記	チャネル
V_{dc} 電源電圧	g_mtr_vdc_value	Q8	ANI5
無通電相電圧	float_voltage	Q8	ANI8, ANI9, ANI10
モータ電流	g_mtr_idc_value	Q10	ANI11
モータコイルエンド温度	g_th_cur_temp_motor	–	ANI4
基板温度	g_th_cur_temp_board	–	ANI24

図 4-1 コイルエンド温度測定 A/D 端子電圧対温度グラフ

表 4-4 コイルエンド温度測定 A/D 端子電圧対温度表

電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]
0.000	-59.393	1.328	19.469	2.657	45.264	3.985	79.031
0.078	-33.636	1.407	21.111	2.735	46.818	4.063	82.049
0.156	-23.353	1.485	22.716	2.813	48.396	4.142	85.326
0.234	-16.808	1.563	24.289	2.891	50.002	4.220	88.917
0.313	-11.870	1.641	25.836	2.969	51.641	4.298	92.896
0.391	-7.838	1.719	27.362	3.048	53.317	4.376	97.367
0.469	-4.391	1.797	28.870	3.126	55.035	4.454	102.478
0.547	-1.354	1.875	30.364	3.204	56.801	4.532	108.450
0.625	1.381	1.954	31.849	3.282	58.620	4.611	115.640
0.703	3.884	2.032	33.326	3.360	60.501	4.689	124.664
0.781	6.205	2.110	34.799	3.438	62.450	4.767	136.738
0.860	8.378	2.188	36.272	3.516	64.477	4.845	154.778
0.938	10.430	2.266	37.748	3.595	66.592	4.923	189.159
1.016	12.382	2.344	39.228	3.673	68.808	5.000	431.619
1.094	14.250	2.422	40.717	3.751	71.141		
1.172	16.047	2.501	42.217	3.829	73.607		
1.250	17.783	2.579	43.732	3.907	76.228		

図 4-2 基板温度測定 A/D 端子電圧対温度グラフ

表 4-5 基板温度測定 A/D 端子電圧対温度表

電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]	電圧 [V]	温度 [°C]
0.000	-46.154	1.328	41.174	2.657	77.864	3.985	128.909
0.078	-28.744	1.407	43.453	2.735	80.135	4.063	133.649
0.156	-15.757	1.485	45.688	2.813	82.45	4.142	138.83
0.234	-7.357	1.563	47.887	2.891	84.814	4.220	144.548
0.313	-0.95	1.641	50.055	2.969	87.233	4.298	151.285
0.391	4.326	1.719	52.2	3.048	89.716	4.376	161.566
0.469	8.869	1.797	54.326	3.126	92.27	4.454	171.848
0.547	12.898	1.875	56.44	3.204	94.905	4.532	182.129
0.625	16.546	1.954	58.545	3.282	97.629	4.611	192.41
0.703	19.903	2.032	60.647	3.360	100.456	4.689	202.691
0.781	23.029	2.110	62.75	3.438	103.397	4.767	212.973
0.860	25.969	2.188	64.858	3.516	106.468	4.845	223.254
0.938	28.758	2.266	66.975	3.595	109.688	4.923	233.535
1.016	31.42	2.344	69.107	3.673	113.076	5.000	243.656
1.094	33.978	2.422	71.257	3.751	116.659		
1.172	36.447	2.501	73.43	3.829	120.465		
1.250	38.842	2.579	75.631	3.907	124.533		

4.2　タイマ・アレイ・ユニット TAU 機能

タイマ・アレイ・ユニット TAU は、8 個の 16 ビット・タイマを搭載しています。各 16 ビット・タイマは「チャネル」と呼び、それぞれを単独のタイマとして使用することはもちろん、複数のチャネルを組み合わせて高度なタイマ機能として使用することもできます。対象マイコン（RL78/F24）では、2 ユニット（計 16 チャネル）を搭載しています。

図 4-3 タイマ・アレイ・ユニット

本システムではタイマ・アレイ・ユニットを表 4-6 のように使用しています。

表 4-6 タイマ・アレイ・ユニットの使用内容

Unit	CH	項目	内容	用途
0	1	タイマの動作モード	インターバル・タイマ機能	1 [ms]生成用タイマ
		ソース・クロック	CK00	
		カウント・クロック周波数	40 [MHz]	
		割り込み周期	1 [ms]	
		タイマ・データ・レジスタ 1（TDR01）設定値	39999 ((1 [ms] / (1/40 [MHz])) - 1)	
	5	タイマの動作モード	キャプチャ機能	PWM 周期測定用タイマ
		ソース・クロック	CK01	
		カウント・クロック周波数	625 [kHz]	
		割り込み周期	両エッジ検出毎	
		タイマ・データ・レジスタ 5（TDR05）設定値	–	
	6	タイマの動作モード	ワンカウント機能	PWM 周期測定タイムアウト生成用タイマ
		ソース・クロック	CK01	
		カウント・クロック周波数	625 [kHz]	
		割り込み周期	100 [ms]	
		タイマ・データ・レジスタ 6（TDR06）設定値	62499 ((100 [ms] / (1/625 [kHz])) - 1)	

4.3　タイマ RDe 機能

タイマ RDe は、16 ビットタイマを 2 本（タイマ RD0、タイマ RD1）持ちます。
また、タイマ RDe には、以下の 6 つのモードがあります。

- タイマモード
- リセット同期 PWM モード
- 相補 PWM モード
- PWM3 モード
- 拡張 PWM モード
- 拡張相補 PWM モード

本システムでは、タイマ RDe を表 4-7 のように使用しています。

表 4-7 タイマ RDe の使用内容

使用タイマ	項目	内容	用途
タイマ RDe	使用モード	拡張相補 PWM モード（対称波形出力）	三相相補 PWM 出力
	PWM 周期	50 [μs]	
	短絡防止時間（Dead Time）	1.0 [μs]	
	カウント周波数	80 [MHz]	
	出力レベル	初期出力 “Low”、アクティブ・レベル “High”	
	バッファ動作	あり	
	パルス出力強制遮断制御	有効（PWM オプション・ユニット A を使用）（遮断時の出力値：ハイインピーダンス出力）	
	出力端子	表 5-13 参照	
	A/D 変換トリガ	PEAK_COUNT - DEADTIME_COUNT（TRD1 最大値）	

備考：拡張相補 PWM モードは、タイマ RD0 とタイマ RD1 のカウンタやレジスタを組み合わせて三相相補 PWM 波形を出力します。

拡張相補 PWM モードの PWM 出力波形例を図 4-4 に示します。

図 4-4 拡張相補 PWM モード時の PWM 出力波形例

4.3.1 PWM デューティ指令値から PWM のデューティ設定へ

拡張相補 PWM モード（対称波形出力）時のデューティの設定方法をまとめます。

正規化された PWM デューティ指令値と PWM 周期カウント値から正相アクティブ・レベル幅を求め、バッファ・レジスタ TRDGRD0、TRDGRC1、TRDGRD1 への設定値を求めます。

正相アクティブ・レベル幅 = PWM デューティ指令値 × PWM 周期カウント値
TRDGRD0, TRDGRC1, TRDGRD1 = PWM 周期カウント値 − 正相アクティブ・レベル幅

5. 制御プログラム説明

本システムの制御プログラムについて説明します。

5.1 制御ブロック図

本システムではオープンループ制御によりモータを始動し、その後は以下のブロック図に従い、制御を行っています。

図 5-1 制御ブロック図

表 5-1 制御ブロック図内の変数

名称	変数名	Q 表記	内容
N	–	–	回転速度
N_{lpf}	g_mtr_rpm_speed	–	LPF 後回転速度
N*	g_mtr_rpm_ref_request	–	回転速度指令値
N_{err}	–	–	回転速度偏差
d	g_mtr_next_duty	–	PWM デューティ指令値
PWM	–	–	PWM 出力信号
θ	g_mtr_current_angle	Q14	推定角度値
$\Delta\theta$	g_mtr_increase_angle	Q14	パルス毎の増分角度
V_{dc}	–	–	インバータ Vdc 電圧
V_U, V_V, V_W	–	–	相電圧

制御プログラムの機能構成は、以下の通りです。

(1) 速度 PI 制御

速度指令値と現在の速度の偏差から PI 制御を実施し PWM デューティ指令値を算出します。

(2) 通電パターン設定

現在の推定角度値を使用して 60°毎の通電のパターン切り替えと、PWM デューティ指令値から PWM 出力値の更新を行います。PWM 出力値の更新を行う際に PWM デューティ指令値は、上限パラメータによる制限値に丸めて使用します。

(3) インバータ

CPU から出力される PWM 信号により直流を三相交流に変換し BLDC モータに供給します。

(4) BLDC モータ

インバータから供給される電力を回転トルクに変換します。

(5) ゼロクロス検出

BLDC モータの相電圧（V_U, V_V, V_W）を監視し BEMF [注]のゼロクロス・タイミングを検出します。また、ゼロクロス検出時に推定角度値の補正と、キャリア毎の増分角度値の計算を行います。

(6) 推定角度計算

PWM キャリア割り込み発生毎にキャリア毎の増分角度値を積算して現在の推定角度値を更新します。

(7) 回転速度計算

6 回分のパターン切り替え処理による経過時間をグローバルパルス番号から算出し、現在の回転速度を計算します。ここで計算される回転速度は、回転速度 LPF 演算により平均化してから使用します。

(8) 回転速度 LPF 演算

指数平滑移動平均による回転速度の平均化を行います。

注 逆起電力（back electromotive force）

5.2　制御内容

5.2.1　モータ起動／停止／変速

5.2.1.1　経過時間による制御

モータの回転開始と停止は、H/W リセット解除後の経過時間によって制御します。1 [ms]インターバル・タイマ処理でカウンタを用意し、カウント数に応じて制御を変更します。

本ソフトウェアでは、リセット解除 3 [s]経過後に 500 [rpm]の速度指令値を設定し、動作モードを変更して回転制御を開始します。

動作開始後、10 [s]経過毎に回転数指令値に USER_CFG_MOTOR_DIFF_SPEED で指定した値（初期値：500 [rpm]）を加算します。回転数指令値が USER_CFG_MOTOR_MAX_SPEED で指定した値（初期値：3000 [rpm]）まで到達した後は、10 [s]経過毎に回転数指令値から USER_CFG_MOTOR_DIFF_SPEED で指定した値（初期値：500）を減算します。

回転数指令値が 0 [rpm]になった場合、動作モードを変更し回転制御を停止します。停止から 3 [s]後に、再度上記の動作を繰り返し実行します。

5.2.1.2　PWM 信号入力による制御

モータの回転開始と停止は、外部から TI05 に PWM 信号を入力することによって制御します。なお、入力周波数は 10 [Hz]から 1 [kHz]の間とする必要があり、範囲外の信号が入力された場合の動作については期待する動作となりません。本ソフトウェアでは入力信号のデューティ比によって回転数指令値が更新されます。デューティ比と回転数指令値の関係は以下の式の通りです。

回転数指令値 =
(PWM_COM_DUTY_MAX – 入力信号のデューティ比) × USER_CFG_MOTOR_MAX_SPEED
PWM_COM_DUTY_MAX：4096 [rpm],
USER_CFG_MOTOR_MAX_SPEED：3000 [rpm]

5.2.2　インバータ・バス電圧

インバータ・バス電圧は、表 5-2 のように測定します。変換計算した値は、過電圧の検出に使用します（過電圧検出した場合、PWM 出力は停止します）。

表 5-2 インバータ・バス電圧の変換比

項　目	変換比（インバータ・バス電圧 Vdc：A/D 変換値）	チャネル
インバータ・バス電圧	0 [V] ～ 65 [V] : 000H ～ FFFH	ANI5

図 5-2 インバータ・バス電圧の外部回路の概念図

5.2.3　モータ三相電圧

三相電圧（U, V, W 相電圧）は、表 5-3 のように測定し、ゼロクロス判定に使用します。

表 5-3 U, V, W 相電圧の変換比

項　目	変換比（三相電圧：A/D 変換値）	チャネル
U, V, W 相電圧	0 [V] ～ 25 [V]：000H ～ FFFH	ANI8, ANI9, ANI10

図 5-3 U, V, W 相電圧測定外部回路の概念図

5.2.4　回転速度演算

回転速度は、ゼロクロス検出とグローバルパルス番号を使用して計算します。ゼロクロス検出によるパターン切り替えタイミングで、グローバルパルス番号を取得し、その値から次式による速度演算を行います。

回転速度 (N) = (60 × 20 [kHz]) / (グローバルパルス番号 × モータ極対数)

備考　　20 [kHz]：PWM キャリア割り込み周波数

また、本システムでは、速度演算結果に対して LPF（指数平滑移動平均）処理を行っています。

5.2.5　速度 PI 制御

本システムでは、次のパターン切り替えまでの間に複数回 PI 制御が行われることを避けるために、速度 PI 制御は 10 [ms]周期で行います。

電圧指令値（V*）は以下の式により作成します。

比例項（P）：K_P × (現在の回転速度偏差 – 前回の回転速度偏差)

積分項（I）：K_I × (現在の回転速度偏差)

電圧指令値 (V*) = 前回の電圧指令値 + 比例項 + 積分項

備考　　K_P：　比例ゲイン (1.50)
　　K_I：　積分ゲイン (0.30)
　　K_P, K_I の値は、使用するシステムに依存します。

PI 制御の詳細については、専門書を参照してください。

5.2.6　モータ電流

モータ電流は、図 5-4 のように測定します。変換計算した値は、モータ電流エラーの検出に使用します（エラーを検出した場合、PWM 出力は停止します）。

表 5-4 モータ電流の変換

項　目	変換比（モータ電流 Idc：A/D 変換値）	チャネル
モータ電流	0 [A] ～ 50 [A]：000H ～ FFFH	ANI11

図 5-4 モータ電流の外部回路の概念図

5.2.7　状態遷移

図 5-5 にサンプルプログラムにおける状態遷移図を示します。

図 5-5 サンプルプログラムの状態遷移図

5.2.8　システム保護機能

本制御プログラムは、以下の 6 種のエラー状態を持ち、それぞれの場合に緊急停止機能を実現しています。

(1)　ハードウェアによる過電流エラー

ハードウェアからの緊急停止信号（過電流検出）により、パルス出力強制遮断を行います（CPU を介さない緊急停止）。マイコン内蔵のコンパレータを使用し過電流検出を行います。

(2)　回転速度異常エラー

1 [ms]間隔で回転速度演算値を監視し、回転速度異常値を検出した時に、CPU によって緊急停止します。

表 5-5 回転速度異常エラー判定のパラメータ

パラメータ名	初期設定値	Q 表記	概要
g_err_over_speed_rpm	10000	–	回転速度異常しきい値 [rpm]

(3)　回転ロックエラー

一定時間ゼロクロス検出によるパターン切り替えが発生しない場合、CPU によって緊急停止します。

図 5-6 回転ロックエラー判定

表 5-6 回転ロックエラー判定のパラメータ

パラメータ名	初期設定値	Q 表記	概要
g_err_lock_detect_time	200	–	回転ロック検出時間 [ms]

(4) バス電圧エラー（過電圧、低電圧）

A/D コンバータを使用して Vdc の入力レベルを監視し、過電圧／低電圧を検出した時に CPU によって緊急停止します。

図 5-7 バス電圧エラー判定

表 5-7 バス電圧エラー判定のパラメータ

パラメータ名	初期設定値	Q 表記	概要
g_err_over_vol_level	28.0	Q8	過電圧判定しきい値 [V]
g_err_under_vol_level	8.0	Q8	低電圧判定しきい値 [V]

(5) モータ電流エラー

A/D コンバータを使用して Idc の入力レベルを監視し、過電流を検出した時に CPU によって緊急停止します。

図 5-8 モータ電流エラー判定

表 5-8 モータ電流エラー判定のパラメータ

パラメータ名	初期設定値	Q 表記	概要
g_err_over_cur_level	10.0	Q10	過電流判定しきい値 [A]
g_err_over_cur_detect_time	3	–	過電流検出判定時間 [キャリアパルス回数]

(6) 過熱エラー（モータコイルエンド温度、基板温度）

A/D コンバータを使用してモータコイルエンド温度と基板温度の入力レベルを監視し、過熱を検出した時に CPU によって緊急停止します。

図 5-9 過熱エラー判定

表 5-9 過熱エラー判定のパラメータ

パラメータ名	初期設定値	Q 表記	概要
g_err_over_temp_level_board	125	–	基板温度過熱判定しきい値 [°C]
g_err_over_temp_level_motor	180	–	モータコイルエンド温度過熱判定しきい値 [°C]

エラー番号：

駆動中にエラーを検出した場合、発生したエラーを示す番号を g_err_flag_cur_loop に格納します。複数のエラーを検出した場合、発生したエラーを表すビットが “OR” される形で格納します。

表 5-10 エラー番号

変数名	値	概要
g_err_flag_cur_loop	0x0001 (bit 0)	過電圧エラー
	0x0002 (bit 1)	低電圧エラー
	0x0010 (bit 4)	過電流エラー
	0x0020 (bit 5)	ハードウェアによる過電流エラー
	0x0100 (bit 8)	回転ロックエラー
	0x0200 (bit 9)	回転速度異常エラー
	0x1000 (bit 12)	基板温度過熱エラー
	0x2000 (bit 13)	モータコイルエンド温度過熱エラー

5.3 システム･リソース

5.3.1 割り込み

本制御プログラムで使用する割り込みの一覧を次に示します。

表 5-11 割り込みリソース

割り込み	割り込みハンドラ	割り込み条件	主な機能
キャリア同期割り込み (INTTRD1)	static void carrier_isr(void)	50.0 [μs] (20 [kHz])	● ゼロクロス検出処理 ● 通電パターン切り替え ● 回転速度演算 ● エラー監視
インターバル・タイマ割り込み (INTTM01)	static void interval_timer_isr(void)	1 [ms] (1 [kHz))	● ユーザ・インタフェース処理 ● タイムアウト検出タイミング生成 ● オープンループ制御 ● 最低回転数制御 ● 速度 PI 制御 ● 温度取得 ● 過熱エラー判定
A/D 変換完了割り込み (INTAD)	static void ad_completion_isr(void)	A/D 変換完了	バス電圧、BEMF 電圧、モータ電流取得
キャプチャ・タイマ割り込み (INTTM05)	static void pwm_com_isr(void)	TI05 入力信号のエッジ検出	PWM 信号入力による制御
ワンカウント・タイマ割り込み (INTTM06)	static void pwm_com_timeout_isr(void)	100 [ms] (10 [Hz])	PWM 信号入力による制御

5.3.2　ポート機能

本制御プログラムで使用するポート機能の一覧を次に示します。

表 5-12 ポート機能

入出力	ポート番号	機　能	備　考
出力	P140	ゲートドライバ有効／無効出力	–
	P11	U 相上アームモータ制御信号ポート出力 (U_P)	論理設定は “High” アクティブ
	P12	U 相下アームモータ制御信号ポート出力 (U_N)	
	P15	V 相上アームモータ制御信号ポート出力 (V_P)	
	P16	V 相下アームモータ制御信号ポート出力 (V_N)	
	P17	W 相上アームモータ制御信号ポート出力 (W_P)	
	P30	W 相下アームモータ制御信号ポート出力 (W_N)	

5.3.3　PWM 出力部

本制御プログラムで使用する PWM 出力部の一覧を次に示します。

表 5-13 PWM 出力機能

入出力	出力端子	機　能	備　考
出力	TRDIOB0 / P11	U 相上アームモータ制御信号 PWM 出力 (U_P)	論理設定は “High” アクティブ
	TRDIOD0 / P12	U 相下アームモータ制御信号 PWM 出力 (U_N)	
	TRDIOA1 / P15	V 相上アームモータ制御信号 PWM 出力 (V_P)	
	TRDIOC1 / P16	V 相下アームモータ制御信号 PWM 出力 (V_N)	
	TRDIOB1 / P17	W 相上アームモータ制御信号 PWM 出力 (W_P)	
	TRDIOD1 / P30	W 相下アームモータ制御信号 PWM 出力 (W_N)	

5.4 関数仕様

本制御プログラムでは、複数の制御関数を使用しています。制御関数の一覧を以下に示します。
より詳細な処理については、フローチャート、またはソースファイルを参照してください。

表 5-14 制御関数一覧 (1/5)

関数仕様	処理概要
ファイル：mtr_main.c	
関数名：main () 入力：なし 出力：なし	• 各種初期化関数呼び出し – CPU 初期化関数呼び出し – インバータ依存部初期化関数呼び出し – モータ依存部初期化関数呼び出し – ユーザ入力初期化関数呼び出し • メイン処理 – ウォッチドッグ・タイマ・クリア
ファイル：mtr_cpu_setting.c (1/2)	
関数名：cpu_init () 入力：なし 出力：なし	CPU 初期化 • 各周辺機能初期化関数呼び出し – クロック初期化関数呼び出し – I/O ポート初期化関数呼び出し – TAU01 モジュール初期化関数呼び出し – A/D モジュール初期化関数呼び出し – D/A モジュール初期化関数呼び出し – CMP モジュール初期化関数呼び出し – TRDe モジュール初期化関数呼び出し – TAU05 モジュール初期化関数呼び出し – TAU06 モジュール初期化関数呼び出し • 各種タイマ・スタート – TAU01 カウント開始関数呼び出し – TRDe カウント開始関数呼び出し – TAU05 カウント開始関数呼び出し – TAU06 カウント開始関数呼び出し
関数名：cgc_init () 入力：なし 出力：なし	クロック初期化
関数名：port_init () 入力：なし 出力：なし	I/O ポート初期化
関数名：tau01_init () 入力：uint16_t freq 出力：なし	TAU01 モジュール初期化 freq：TAU01 割り込み周波数データ
関数名：tau01_start () 入力：なし 出力：なし	TAU01 カウント開始
関数名：trde_init () 入力：なし 出力：なし	TRDe モジュール初期化
関数名：trde_start () 入力：なし 出力：なし	TRDe カウント開始

表 5-14 制御関数一覧 (2/5)

関数仕様	処理概要
ファイル：mtr_cpu_setting.c (2/2)	
関数名：trde_disable_inv_out () 入力：なし 出力：なし	インバータ出力停止
関数名：pwmopa_release_hz () 入力：なし 出力：なし	PWM 出力強制遮断解除
関数名：adc_init () 入力：なし 出力：なし	A/D モジュール初期化
関数名：adc_set_ad_ch_bemf () 入力：uint16_t channel 出力：なし	A/D 変換取得チャネル更新 channel：A/D 取得チャネル・データ
関数名：dac_init () 入力：なし 出力：なし	D/A モジュール初期化
関数名：cmp_init () 入力：なし 出力：なし	CMP モジュール初期化
関数名：port_change_pattern () 入力：uint8_t pattern 出力：なし	PWM 出力パターン変更 pattern：出力パターン・データ
関数名：trde_calc_pwm_ref () 入力：int16_t ref 出力：なし	デューティ設定値から TRDe ジェネラル・レジスタ設定値を計算 ref：PWM デューティ指令値
関数名：tau05_init () 入力：なし 出力：なし	TAU05 モジュール初期化
関数名：tau05_start () 入力：なし 出力：なし	TAU05 カウント開始
関数名：tau06_init () 入力：なし 出力：なし	TAU06 モジュール初期化
関数名：tau06_start () 入力：なし 出力：なし	TAU06 カウント開始
関数名：wdt_clear () 入力：なし 出力：なし	ウォッチドッグ・タイマ・クリア

表 5-14 制御関数一覧 (3/5)

関数仕様	処理概要
ファイル：mtr_fix_calc.c	
関数名：fix8_mul_int16 () 入力：int16_t fix_1, fix_2 出力：int16_t temp	固定小数点演算：(16 bit × 16 bit) >> 8 fix_1：入力データ 1 fix_2：入力データ 2 temp：演算結果データ
関数名：fix10_mul_int16 () 入力：int16_t fix_1, fix_2 出力：int16_t temp	固定小数点演算：(16 bit × 16 bit) >> 10 fix_1：入力データ 1 fix_2：入力データ 2 temp：演算結果データ
関数名：fix12_mul_int16 () 入力：int16_t fix_1, fix_2 出力：int16_t temp	固定小数点演算：(16 bit × 16 bit) >> 12 fix_1：入力データ 1 fix_2：入力データ 2 temp：演算結果データ
関数名：fix14_mul_int16 () 入力：int16_t fix_1, fix_2 出力：int16_t temp	固定小数点演算：(16 bit × 16 bit) >> 14 fix_1：入力データ 1 fix_2：入力データ 2 temp：演算結果データ
関数名：fix16_mul_int16 () 入力：int16_t fix_1, fix_2 出力：int16_t temp	固定小数点演算：(16 bit × 16 bit) >> 16 fix_1：入力データ 1 fix_2：入力データ 2 temp：演算結果データ
ファイル：mtr_function.c	
関数名：err_check_error_cur_loop () 入力：なし 出力：なし	エラーチェックを実行 • 過電流エラー検出 • 過電圧／低電圧エラー検出 • 回転ロックエラー検出関数呼び出し • ハードウェア過電流エラー検出 • 回転速度異常エラー検出
関数名：err_check_over_temp () 入力：なし 出力：なし	過熱エラーチェックを実行 • インバータボード過熱エラー検出 • モータコイルエンド過熱エラー検出
関数名：err_error_stop () 入力：なし 出力：なし	エラー検出によるモータ緊急停止 • インバータ出力停止関数呼び出し
関数名：mtr_detect_synchro_loss () 入力：なし 出力：uint8_t detect	回転ロックエラー検出 detect：エラー結果出力
ファイル：mtr_interrupt_bemf.c (1/2)	
関数名：mtr_detect_zerocross () 入力：なし 出力：なし	ゼロクロス検出
関数名：mtr_speed_calc () 入力：なし 出力：なし	回転速度計算
関数名：pwm_com_get_duty () 入力：なし 出力：uint32_t s_pwm_com_duty	外部入力 PWM 信号のデューティ比取得 s_pwm_com_duty：入力デューティ比データ

表 5-14 制御関数一覧 (4/5)

関数仕様	処理概要
ファイル：mtr_interrupt_bemf.c (2/2)	
関数名：interval_timer_isr () 入力：なし 出力：なし	1 [ms]割り込み • ユーザ・インタフェース処理 • 回転ロックエラー検出タイミング生成 • オープンループ制御 • 低速回転制御 • 速度 PI 制御 • シーケンス処理 • 過熱エラー検出
関数名：carrier_isr () 入力：なし 出力：なし	キャリア同期割り込み • パターン切り替え • 回転速度計算関数呼び出し • 無通電相電圧 A/D 変換チャネル設定 • バス電圧物理値計算 • ゼロクロス検出関数呼び出し • デューティ値設定関数呼び出し • エラーチェック関数呼び出し
関数名：ad_completion_isr () 入力：なし 出力：なし	A/D 変換完了割り込み • バス電圧 A/D 値取得 • 無通電相電圧 A/D 値取得 • モータ電流 A/D 値取得 • モータ電流物理値計算
関数名：pwm_com_isr () 入力：なし 出力：なし	外部入力 PWM 信号のパルス幅測定
関数名：pwm_com_timeout_isr () 入力：なし 出力：なし	外部入力 PWM 信号パルス幅測定のタイムアウト判定
ファイル：mtr_inv_setting.c	
関数名：inv_init () 入力：なし 出力：なし	インバータ依存部初期化 • バス電圧物理値変換係数設定 • モータ電流物理値変換係数設定
関数名：th_get_temp_board () 入力：なし 出力：float32_t temp1	基板温度取得 temp1：基板温度データ
関数名：th_get_temp_motor () 入力：なし 出力：float32_t temp1	モータコイルエンド温度取得 temp1：モータコイルエンド温度データ

表 5-14 制御関数一覧 (5/5)

関数仕様	処理概要
ファイル：r_mtr_sequence_api.c	
関数名：R_SEQ_ExecEvent () 入力：uint8_t event 出力：なし	• ステータスの変更 • 発生イベントに対する実行関数呼び出し event：発生イベント・データ
関数名：seq_act_run () 入力：uint8_t state 出力：uint8_t state	モータ駆動時変数初期化関数呼び出し state：シーケンス・ステータス・データ
関数名：seq_act_stop () 入力：uint8_t state 出力：uint8_t state	• インバータ出力停止関数呼び出し • モータ駆動時変数初期化関数呼び出し • PWM 出力強制遮断解除関数呼び出し state：シーケンス・ステータス・データ
関数名：seq_act_none () 入力：uint8_t state 出力：uint8_t state	処理なし state：シーケンス・ステータス・データ
関数名：seq_act_reset () 入力：uint8_t state 出力：uint8_t state	• モータ駆動時変数初期化関数呼び出し • PWM 出力強制遮断解除関数呼び出し state：シーケンス・ステータス・データ
関数名：seq_act_error () 入力：uint8_t state 出力：uint8_t state	インバータ出力停止関数呼び出し state：シーケンス・ステータス・データ
関数名：seq_act_break () 入力：uint8_t state 出力：uint8_t state	• ブレーキ・カウントの設定 • PWM 出力パターン変更関数呼び出し state：シーケンス・ステータス・データ
関数名：seq_init_start () 入力：なし 出力：なし	モータ駆動時変数初期化
ファイル：r_mtr_user_control_api.c	
関数名：R_USER_InitMotor () 入力：なし 出力：なし	モータ依存部初期化 • モータ・パラメータ初期化
関数名：R_USER_InitCtrl () 入力：なし 出力：なし	ユーザ入力初期化 • ユーザ設定パラメータ初期化
関数名：R_USER_MotorControl () 入力：なし 出力：なし	ユーザ・インタフェース処理 • モータ・ステータスの変更（モータ動作開始） • 回転速度指令値の決定
関数名：user_change_ref_speed () 入力：なし 出力：なし	速度指令値の段階的な変更

5.5 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、（※）と記載された変数は、二次的に使用する変数のため書き換えは行わないで下さい。

表 5-15 変数一覧 (1/3)

変数名	型	内容
ファイル：mtr_function.c		
g_err_over_vol_level	int16_t	過電圧エラー判定しきい値
g_err_under_vol_level	int16_t	低電圧エラー判定しきい値
g_err_over_cur_level	int16_t	過電流エラー判定しきい値
g_err_over_cur_detect_time	uint16_t	過電流エラー判定時間
g_err_over_cur_time_cnt	uint16_t	過電流エラー検出時間カウンタ（※）
g_err_lock_detect_time	uint16_t	回転ロックエラー判定時間
g_err_flag_cur_loop	uint16_t	エラー発生フラグ（※）
g_err_timeout_cnt	uint16_t	回転ロック検出経過時間カウンタ（※）
g_err_over_speed_rpm	uint16_t	回転速度異常エラー判定しきい値
g_err_over_temp_level_board	uint16_t	基板温度過熱エラー判定しきい値
g_err_over_temp_level_motor	uint16_t	モータコイルエンド温度過熱エラー判定しきい値
ファイル：mtr_inv_setting.c		
g_inv_vdc_range	int16_t	バス電圧物理値変換係数（※）
g_inv_cur_range	int16_t	モータ電流物理値変換係数（※）
g_inv_bemf_range	int16_t	BEMF 物理値変換係数（※）
ファイル：r_mtr_user_control_api.c		
g_user_TargetRotateDir	uint8_t	回転方向 [0：回転方向 CW, 1：回転方向 CCW]
g_user_MotorAccelDir	uint8_t	モータの加減速状態（※）
ファイル：mtr_interrupt_bemf.c (1/3)		
g_mtr_rotate_dir	uint8_t	現在の回転方向（※） [0：回転方向 CW, 1：回転方向 CCW]
g_mtr_mode_system_request	uint8_t	ユーザ・イベント入力 [0：停止, 1：起動, 2：エラー, 3：リセット, 4：ブレーキ]
g_mtr_mode_system	uint8_t	内部運転状態（※） [0：停止中, 1：運転中, 2：エラー発生中]
g_err_error_status	uint8_t	エラー番号（※）
g_mtr_run_mode	uint8_t	動作モード（※） [0：オープンループ, 1：BEMF]
g_mtr_rpm_speed	int16_t	現在の速度（※）
g_mtr_ref_limit	int16_t	デューティ制限値
g_mtr_rpm_ref	int16_t	目標回転数（※）
g_mtr_rpm_ref_request	int16_t	目標回転数指示値
g_mtr_rpm_ref_request_old	int16_t	目標回転数指示値（前回値）（※）
g_mtr_slope_time_count	uint8_t	目標速度スロープ制御用カウンタ（※）
g_mtr_setting_min_rpm	uint16_t	最低回転数設定値
g_mtr_speed_lpf_factor	uint16_t	回転速度フィルタ値
g_mtr_speed_culc_counter	uint8_t	回転速度計算用セグメント数（※）

表 5-15 変数一覧 (2/3)

変数名	型	内容
ファイル：mtr_interrupt_bemf.c (2/3)		
g_ad_data_bemf	uint16_t	無通電相 A/D 取得値（※）
g_ad_data_vdc	uint16_t	バス電圧 A/D 取得値（※）
g_ad_data_idc	int16_t	モータ電流 A/D 取得値（※）
g_ad_data_temp_board	uint16_t	基板温度 A/D 取得値（※）
g_ad_data_temp_motor	uint16_t	モータコイルエンド温度 A/D 取得値（※）
g_mtr_time_setting_offset	uint16_t	起動時オフセット初期化用時間設定
g_mtr_time_cnt_offset	uint16_t	起動時オフセット初期化用カウンタ（※）
g_mtr_vdc_value	int16_t	バス電圧物理値
g_mtr_idc_value	int16_t	モータ電流物理値
g_mtr_offset_idc	int16_t	モータ電流オフセット値
g_mtr_offset_idc_lpf_factor	int16_t	モータ電流オフセット・フィルタ定数
g_mtr_idc_lpf_factor	uint16_t	モータ電流フィルタ定数
g_mtr_vdc_lpf_factor	uint16_t	バス電圧フィルタ定数
g_th_cur_temp_board	float32_t	基板温度物理値
g_th_cur_temp_motor	float32_t	モータコイルエンド温度物理値
g_mtr_current_angle	uint16_t	推定角度値（※）
g_mtr_increase_angle	uint16_t	キャリア毎の増分角度（※）
g_mtr_const_rpm_to_inc_angle	int16_t	RPM から角度への変換係数
g_mtr_speed_pi_cnt	uint8_t	速度 PI 制御タイミング生成用カウンタ（※）
g_mtr_speed_kp_factor	int16_t	速度 PI 制御 P 項係数
g_mtr_speed_ki_factor	int16_t	速度 PI 制御 I 項係数
g_mtr_speed_ref	int16_t	速度 PI 制御デューティ指令値（※）
g_mtr_speed_ref_p	int16_t	速度 PI 制御 P 項計算値（※）
g_mtr_speed_ref_i	int16_t	速度 PI 制御 I 項計算値（※）
g_mtr_speed_delta_pre	int16_t	速度 PI 制御一次遅れ偏差（※）
g_mtr_speed_pi_err_limit	int16_t	速度 PI 制御偏差制限値
g_mtr_speed_pi_interval	uint8_t	速度 PI 制御実行間隔
g_mtr_pole_pairs	uint8_t	モータ極対数
g_mtr_rpm_slope	int16_t	目標速度変化レート（※）
g_mtr_rpm_slope_request	int16_t	目標速度変化レート指示値
g_mtr_rpm_slope_ol	int16_t	オープンループ目標速度変化レート
g_mtr_arign_cnt_ol_1st	uint16_t	1 回目位置決め処理時間カウンタ（※）
g_mtr_setting_arign_time_ol_1st	uint16_t	1 回目位置決め処理設定時間
g_mtr_arign_cnt_ol_2nd	uint16_t	2 回目位置決め処理時間カウンタ（※）
g_mtr_setting_arign_time_ol_2nd	uint16_t	2 回目位置決め処理設定時間
g_mtr_open_loop_mode	uint16_t	オープンループ動作モード [1：1 回目位置決め処理, 2：2 回目位置決め処理, 3：オープンループ]
g_mtr_change_bemf_rpm_ol	uint16_t	センサレス制御切り替え速度
g_mtr_initial_angle_ol_1st	int16_t	1 回目位置決め処理設定角度
g_mtr_initial_angle_ol_2nd	int16_t	2 回目位置決め処理設定角度
g_mtr_current_pattern	uint8_t	現在の出力パターン番号（※）
g_mtr_next_pattern	uint8_t	出力パターン変更予約番号（※）

表 5-15 変数一覧 (3/3)

変数名	型	内容
ファイル：mtr_interrupt_bemf.c (3/3)		
g_mtr_float_port	uint8_t	無通電相取得ポート番号（※）
g_mtr_pattern_change_enable	uint8_t	パターン変更許可フラグ（※）
g_mtr_pattern_table_120deg_cw	st_pattern_table_t	120 度通電用パターン・テーブル［時計回り］（※）
g_mtr_pattern_table_120deg_ccw	st_pattern_table_t	120 度通電用パターン・テーブル［反時計回り］（※）
* gp_pattern_table	st_pattern_table_t	パターン・テーブル・ポインタ（※）
g_mtr_detect_mode	uint8_t	BEMF 検出モード ［0：BEMF 検出なし, 1：BEMF 検出あり］
g_mtr_segment_pulse_number	uint16_t	セグメントパルス番号（※）
g_mtr_global_pulse_number	uint16_t	グローバルパルス番号（※）
g_mtr_global_pulse_number_old	uint16_t	速度算出起点でのグローバルパルス番号（※）
g_mtr_permit_bemf_mode	uint8_t	BEMF モード遷移許可フラグ（※）
g_mtr_advance_angle	uint16_t	進角度（※）
g_mtr_bemf_detect_end_threshold	uint16_t	ゼロクロス検出電圧値
g_mtr_correction_angle	uint16_t	補正角度（※）
g_mtr_detect_slope	uint8_t	BEMF 検出スロープ（※）
g_mtr_initial_duty	int16_t	デューティ初期設定値
g_mtr_next_duty	int16_t	デューティ変更予約値（※）
g_mtr_current_duty	int16_t	現在の出力デューティ値（※）
g_mtr_auto_power_off_time	uint16_t	モータ自動停止時間（※）
g_mtr_auto_power_off_tick_count	uint16_t	モータ自動停止時間カウンタ（※）
g_mtr_brake_count	uint16_t	ブレーキ・カウント（※）
g_mtr_brake_count_request	uint16_t	ブレーキ・カウント指令値（※）
g_mtr_brake_tick_count	uint16_t	ブレーキ・カウント判定用カウンタ（※）

5.6 マクロ定義

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 5-16 マクロ定義一覧 (1/4)

マクロ名	定義値	内容
ファイル：mtr_cpu_setting.h		
MCUIO_PLL_CLOCK_HZ	80000000	PLL クロック周波数 [Hz]
MCUIO_SYSTEM_CLOCK_HZ	40000000	CPU クロック周波数 [Hz]
MCUIO_PERIPHERAL_CLOCK_HZ	40000000	周辺ハードウェア・クロック周波数 [Hz]
ファイル：mtr_interrupt_bemf.h (1/2)		
MTR_MODE_RUN_OPENLOOP	1	動作モード：オープンループ
MTR_MODE_RUN_BEMF	2	動作モード：BEMF モード
MTR_MODE_OPENLOOP_ARIGN_1ST	1	オープンループモード：位置決め 1 回目
MTR_MODE_OPENLOOP_ARIGN_2ND	2	オープンループモード：位置決め 2 回目
MTR_MODE_OPENLOOP_ROTATE	3	オープンループモード：オープンループ
ERROR_OVER_VOLTAGE	0x0001	過電圧エラー番号
ERROR_UNDER_VOLTAGE	0x0002	低電圧エラー番号
ERROR_OVER_CURRENT	0x0010	過電流エラー番号
ERROR_OVER_CURRENT_HW	0x0020	H/W 過電流検出エラー番号
ERROR_STEP_OUT	0x0100	回転ロックエラー番号
ERROR_OVER_SPEED	0x0200	回転速度エラー番号
ERROR_OVER_TEMPERATURE_BOARD	0x1000	基板温度過熱エラー番号
ERROR_OVER_TEMPERATURE_MOTOR	0x2000	モータコイルエンド温度過熱エラー番号
MTR_SLOPE_UP	0	上り BEMF 検出
MTR_SLOPE_DOWN	1	下り BEMF 検出
MTR_PATTERN_NONE	0	パターン番号：パターンなし
MTR_U_PWM_V_LOW_W_OFF	1	パターン番号：U 相→V 相
MTR_U_PWM_W_LOW_V_OFF	2	パターン番号：U 相→W 相
MTR_V_PWM_W_LOW_U_OFF	3	パターン番号：V 相→W 相
MTR_V_PWM_U_LOW_W_OFF	4	パターン番号：V 相→U 相
MTR_W_PWM_U_LOW_V_OFF	5	パターン番号：W 相→U 相
MTR_W_PWM_V_LOW_U_OFF	6	パターン番号：W 相→V 相
MTR_PATTERN_NUM	6	パターン総数
MTR_ALL_LOW_ON	7	パターン番号：ブレーキ
MTR_BEMF_MODE_NONE	0	BEMF 検出モード：検出なし
MTR_BEMF_MODE_DETECT	1	BEMF 検出モード：検出あり
MTR_BEMF_DETECT_GUARD_CNT	2	転流後検出無効パルス数
ANGLE_30_DEGREE	1366	角度正規化値：30°
ANGLE_60_DEGREE	2731	角度正規化値：60°
ANGLE_90_DEGREE	4096	角度正規化値：90°
ANGLE_150_DEGREE	6827	角度正規化値：150°

表 5-16 マクロ定義一覧 (2/4)

マクロ名	定義値	内容
ファイル：mtr_interrupt_bemf.h (2/2)		
ANGLE_210_DEGREE	9558	角度正規化値：210°
ANGLE_270_DEGREE	12288	角度正規化値：270°
ANGLE_330_DEGREE	15019	角度正規化値：330°
ANGLE_360_DEGREE	16384	角度正規化値：360°
PWM_COM_DUTY_MAX	4096	PWM 入力最大値：デューティ = 100%
PWM_COM_DUTY_MIN	0	PWM 入力最小値：デューティ = 0%
ファイル：mtr_inv_setting.h		
INV_VDC_VOL_DIV	13	バス電圧抵抗分圧係数
INV_IDC_CUR_RANGE	50.0f	モータ電流入力レンジ
INV_BEMF_VOL_DIV	5	逆起電力抵抗分圧係数
ファイル：r_mtr_sequence_api.h		
SEQ_CFG_MODE_STOP	0x00	動作モード：停止
SEQ_CFG_MODE_RUN	0x01	動作モード：動作
SEQ_CFG_MODE_ERROR	0x02	動作モード：エラー
SEQ_CFG_MODE_BRAKE	0x03	動作モード：ブレーキ
SEQ_CFG_SIZE_STATE	4	動作モード数
SEQ_CFG_EVENT_STOP	0x00	ユーザ・イベント入力：停止
SEQ_CFG_EVENT_RUN	0x01	ユーザ・イベント入力：動作
SEQ_CFG_EVENT_ERROR	0x02	ユーザ・イベント入力：エラー
SEQ_CFG_EVENT_RESET	0x03	ユーザ・イベント入力：リセット
SEQ_CFG_EVENT_BRAKE	0x04	ユーザ・イベント入力：ブレーキ
SEQ_CFG_SIZE_EVENT	5	ユーザ・イベント入力数
SEQ_CFG_UNKNOWN_ERROR	0xFF	ユーザ・イベント入力：未定義
ファイル：r_mtr_user_control_cfg.h (1/2)		
USER_CFG_MOTOR_POLE_PAIRS	2	モータ極対数
USER_CFG_MOTOR_MAX_SPEED	3000	回転速度上限（機械角）[rpm]
USER_CFG_REF_SPEED_ACCEL	0	加速指令状態
USER_CFG_REF_SPEED_DECEL	1	減速指令状態
USER_CFG_MOTOR_INITIAL_DIR	0	回転方向（0：CW 1：CCW）
USER_CFG_MOTOR_DIFF_SPEED	500	加減速差分値（機械角）[rpm]
USER_CFG_TIME_SETTING_OFFSET	30000	起動時オフセット設定時間 ［キャリアパルス回数］
USER_CFG_TIME_CNT_OFFSET	0	起動時オフセット・カウンタ
USER_CFG_OFFSET_IDC	0	モータ電流オフセット初期値
USER_CFG_OFFSET_IDC_LPF_FACTOR	0.1f	モータ電流オフセット・フィルタ定数
USER_CFG_SPEED_PI_INTERVAL	10	速度 PI 制御周期 [ms]
USER_CFG_SPEED_PI_ERROR_LIMIT	9000	速度 PI 制御エラー上限
USER_CFG_KP_FACTOR	1.50f	速度 PI 制御比例ゲイン K_P
USER_CFG_KI_FACTOR	0.30f	速度 PI 制御積分ゲイン K_I
USER_CFG_INITIAL_RPM_SLOPE	0	目標速度変化レート初期値
USER_CFG_RPM_SLOPE	10	目標速度変化レート
USER_CFG_INITIAL_RPM_REF	0.0	目標速度初期値
USER_CFG_RPM_CLOSE_TO_OPEN	500	オープンループ突入速度（機械角）[rpm]

表 5-16 マクロ定義一覧 (3/4)

マクロ名	定義値	内容
ファイル：r_mtr_user_control_cfg.h (2/2)		
USER_CFG_RPM_SLOPE_OL	1	オープンループ目標速度変化レート
USER_CFG_RPM_OPEN_TO_CLOSE	600	クローズドループ突入速度（機械角）[rpm]
USER_CFG_ARIGN_TIME_OL_1ST	200	1 回目位置決め処理時間 [ms]
USER_CFG_ARIGN_TIME_OL_2ND	20	2 回目位置決め処理時間 [ms]
USER_CFG_ARIGN_DEGREES_OL_1ST	5462	1 回目位置決め処理角度 [°]
USER_CFG_ARIGN_DEGREES_OL_2ND	0	2 回目位置決め処理角度 [°]
USER_CFG_INITIAL_DUTY_OL	0.20f	オープンループ出力デューティ（正規化）
USER_CFG_DUTY_LIMIT	0.95f	デューティ上限
USER_CFG_IDC_LPF_FACTOR	0.10f	I_{dc} ローパスフィルタ定数
USER_CFG_VDC_LPF_FACTOR	0.25f	V_{dc} ローパスフィルタ定数
USER_CFG_SPEED_LPF_FACTOR	0.40f	回転速度ローパスフィルタ定数
USER_CFG_BEMF_DETECT_END_TH	30	ゼロクロス検出電圧しきい値 [A/D 取得値]
USER_CFG_ADVANCE_ANGLE	0	進角値
USER_CFG_OVER_VOLTAGE_LEVEL	28.0f	過電圧検出電圧値 [V]
USER_CFG_UNDER_VOLTAGE_LEVEL	8.0f	低電圧検出電圧値 [V]
USER_CFG_OVER_CUR_LEVEL_HW_CUR	20.0f	H/W によるモータ電流異常検出値 [A]
USER_CFG_OVER_CUR_LEVEL_SW	10.0f	モータ電流異常検出値 [A]
USER_CFG_OVER_CUR_DETECT_TIME	3	モータ電流異常検出時間 [キャリアパルス回数]
USER_CFG_LOCK_DETECT_TIME	200	回転ロック検出時間 [ms]
USER_CFG_OVER_SPEED_RPM	10000	回転速度異常検速度（機械角）[rpm]
USER_CFG_OVER_CUR_LEVEL_HW	2.5f	H/W によるモータ電流異常検出値 [V]
USER_CFG_OVER_TEMP_BOARD	125	基板温度異常検出値 [°C]
USER_CFG_OVER_TEMP_MOTOR	180	モータコイルエンド温度異常検出値 [°C]
USER_CFG_AUTO_POWEROFF_TIME	0	モータ自動停止までの時間 [s]
USER_CFG_BRAKE_TIME	2	ブレーキ時間 [s]
USER_CFG_TIME_CONTROL	–	本マクロが定義されている場合： モータ回転数は時間経過で変化 本マクロが定義されていない場合： モータ回転数は PWM 信号入力で制御

表 5-16 マクロ定義一覧 (4/4)

マクロ名	定義値	内容
ファイル：mtr_fix_calc.h		
FIX_C (x, b)	((int16_t) (((float32_t) (x)) * ((float32_t) (1UL << (b)))))	16 ビット変数を指定ビット数だけ左シフト
LFIX_C (x, b)	((int32_t) (((float32_t) (x)) * ((float32_t) (1UL << (b)))))	32 ビット変数を指定ビット数だけ左シフト

5.7 制御フロー

本制御プログラムのフローチャートを示します。

5.7.1 メイン処理

図 5-10 メイン関数

5.7.2 1 [ms]インターバル・タイマ割り込み処理

図 5-11 インターバル・タイマ割り込み

5.7.3 ユーザ・インタフェース

図 5-12 ユーザ・インタフェース処理

5.7.4 キャリア周期割り込み処理

図 5-13 キャリア周期割り込み処理

5.7.5 ゼロクロス検出処理

図 5-14 ゼロクロス検出処理

5.7.6 A/D 変換完了割り込み処理

図 5-15 A/D 変換完了割り込み処理

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2022. 9.30	–	初版発行
1.10	2023. 6.30	P.7	ソフトウェアを V.1.04 から V.1.05 に更新
		P.9	表 2-4 保護停止処理の以下誤記を修正 回転速度が ~~9000~~ 10000 [rpm]を超過
		P.9	表 2-4 保護停止処理（インバータボード温度、モータコイルエンド温度）に判定タイミングを記載
		P.34	インターバル・タイマ割り込み (INTTM01)に過熱判定処理を追加
		P.38	以下関数の変数定義を static 変数（s_temp）から auto 変数（temp）に修正 • fix8_mul_int16 () • fix10_mul_int16 () • fix12_mul_int16 () • fix14_mul_int16 () • fix16_mul_int16 ()
		P.38	err_check_error_cur_loop () 関数から過熱判定処理を削除
		P.38	err_check_over_temp () 関数を新規追加
		P.39	interval_timer_isr () 関数に過熱判定処理を追加
		P.48	インターバル･タイマ割り込みフローに過熱エラー判定を追加

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

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。

© 2023 Renesas Electronics Corporation. All rights reserved.