

RH850/U2B6

レゾルバ付き永久磁石同期モータの制御

要旨

本アプリケーションノートでは、RH850/U2B6 でレゾルバ付き永久磁石同期モータを制御する方法について説明します。

動作確認デバイス

RH850/U2B6

本アプリケーションノートを他のマイコンへ適用する場合、そのマイコンの仕様にあわせて変更し、十分評価してください。

目次

1. レゾルバ付永久磁石同期モータのベクトル制御	4
1.1 PMSM の動作	4
1.1.1 合成磁場とトルク	5
1.2 モータ制御の概念	5
1.3 U/V/W 相電圧から PWM 出力へ	7
1.3.1 三角波比較法	7
1.4 dq 座標系の PMSM の電圧方程式(参考資料)	8
1.4.1 電圧方程式	8
1.4.2 非干渉制御	8
2. EMU3S の代表的な制御例	9
3. 全自動処理動作	10
3.1 仕様	10
3.2 動作条件	14
3.3 端子	14
3.4 動作概要	15
3.4.1 R/D コンバータ 3AL (RDC3AL)	15
3.4.2 ペリフェラルインタコネクション (PIC)	15
3.4.3 A/D コンバータ(ADCK)	15
3.4.4 モータコントロールタイマ 3 (TSG3)	15
3.4.5 エンハンスドモータコントロールユニット 3S (EMU3S)	17
3.4.5.1 EMU3S 共通部	17
3.4.5.2 角度生成 IP	18
3.4.5.3 入力 IP	18
3.4.5.4 PI 制御 IP	20
3.4.5.5 PWM IP	22
3.4.5.6 演算精度例	27
3.4.6 ファイル構成	28
3.4.7 関数一覧	28
3.4.8 関数仕様	29
3.4.9 フローチャート	31
4. CPU 処理のある制御動作	38
4.1 仕様	38
4.2 動作条件	41
4.3 端子	41
4.4 動作概要	41
4.4.1 R/D コンバータ 3AL (RDC3AL)	41
4.4.2 ペリフェラルインタコネクション (PIC)	41
4.4.3 A/D コンバータ(ADCK)	41
4.4.4 モータコントロールタイマ 3 (TSG3)	41
4.4.5 エンハンスドモータコントロールユニット 3S (EMU3S)	42
4.4.5.1 EMU3S 共通部	42
4.4.5.2 角度生成 IP	42

4.4.5.3	入力 IP	42
4.4.5.4	PI 制御 IP	42
4.4.5.5	PWM IP	42
4.4.6	ファイル構成	43
4.4.7	関数一覧	43
4.4.8	関数仕様	43
4.4.9	フローチャート	44
5.	矩形波出力動作	47
5.1	仕様	47
5.2	動作条件	50
5.3	端子	50
5.4	動作概要	50
5.4.1	R/D コンバータ 3AL (RDC3AL)	50
5.4.2	ペリフェラルインタコネクション (PIC)	50
5.4.3	A/D コンバータ (ADCK)	50
5.4.4	モータコントロールタイマ 3 (TSG3)	50
5.4.5	エンハンスドモータコントロールユニット 3S (EMU3S)	51
5.4.5.1	EMU3S 共通部	51
5.4.5.2	角度生成 IP	51
5.4.5.3	入力 IP	51
5.4.5.4	PI 制御 IP	51
5.4.5.5	PWM IP	51
5.4.5.6	独立矩形 IP3	52
5.4.6	ファイル構成	55
5.4.7	関数一覧	55
5.4.8	関数仕様	56
5.4.9	フローチャート	58
6.	参考ドキュメント	62
	改訂記録	63

1. レゾルバ付永久磁石同期モータのベクトル制御

本章では、レゾルバ付永久磁石同期モータ（Permanent Magnetic Synchronous Motor、以下「PMSM」という。）のベクトル制御の概念について説明します。

1.1 PMSM の動作

PMSM は筐体などに固定されているステータと回転するロータで構成されています。ステータには 120° ごとに U 相コイル、V 相コイル、W 相コイルが配置されており、ロータには永久磁石が組み込まれています。

コイルの両端に電圧を印加すると電流が流れ、それにより磁場が生成されます。この磁場内に永久磁石を置くと永久磁石に引力や斥力が働きます。U 相、V 相、W 相のコイルに電流を流し、各コイルが作る磁場をベクトル合成した合成磁場に引き寄せられるようにロータが動きます。この合成磁場を回転させることにより、ロータが回転します。U 相、V 相、W 相の各コイルに、 120° の位相差をもつ正弦波の電流を流すと、磁場の大きさが同じで回転速度が一定な回転磁場を作ることができます。図 1.1 に PMSM の構造と合成磁場を示します。

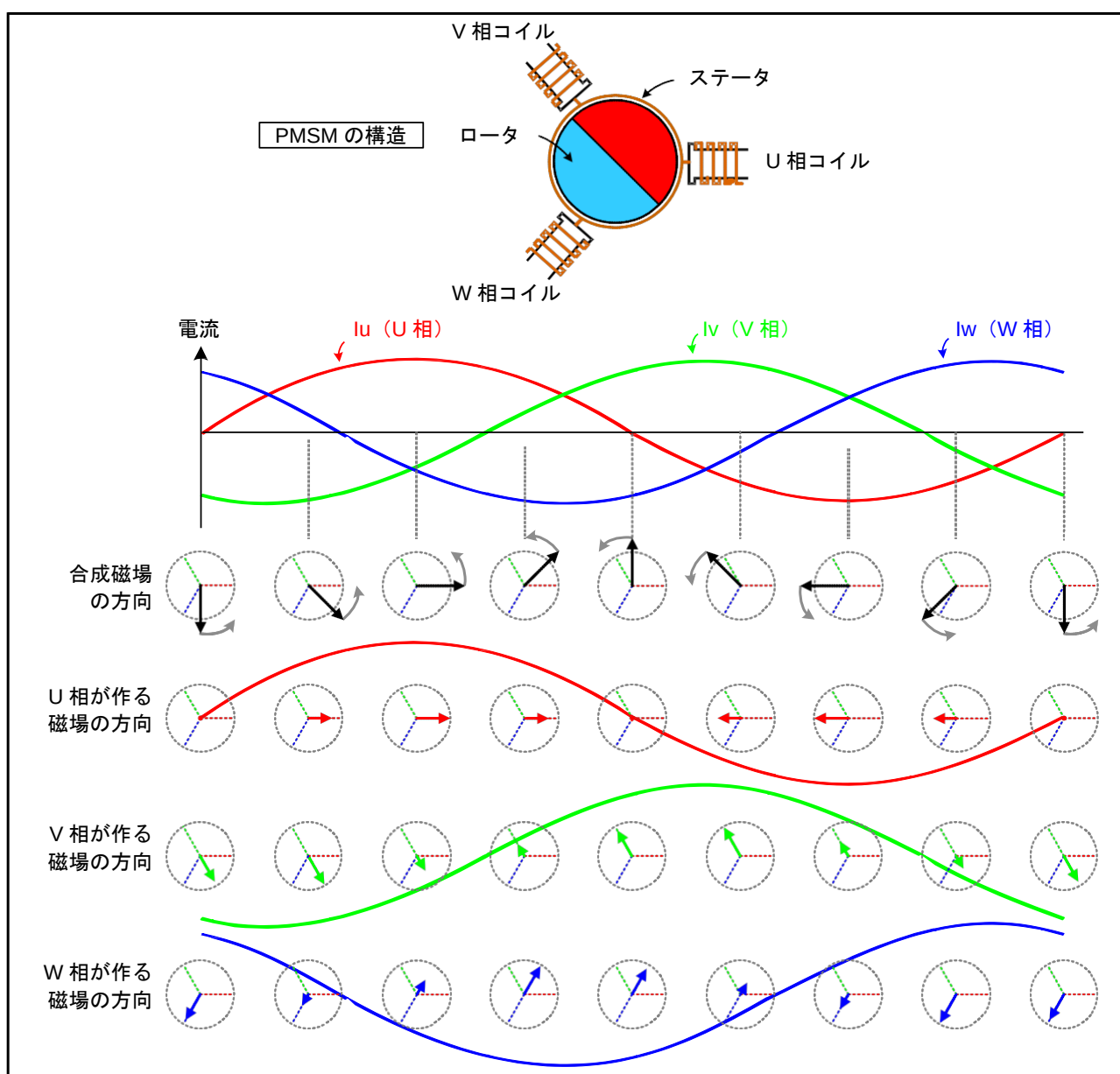


図 1.1 PMSM の構造と合成磁場

1.1.1 合成磁場とトルク

図 1.2 に合成磁場とトルクの関係を示します。

永久磁石の磁極と合成磁場のなす角を 90° にした場合、トルク(永久磁石を回す力)が最大になります。この磁極と直交した合成磁場を直交磁場と呼びます。ここで、永久磁石の磁極方向を d 軸、直交磁場の方向を q 軸とします。

モータを効率よく回転させるためには、常に直交磁場が生成されるよう制御する必要があります。直交磁場を生成するために、q 軸方向にコイルが配置されていると仮定し、そのコイルに流す電流を q 軸電流(I_q) とします。このとき直交磁場の強さは q 軸電流の大きさに比例します。

モータの回転速度が想定よりも遅い場合は q 軸電流を大きくし、トルクを強め加速します。また、モータの回転速度が想定よりも速い場合は q 軸電流を小さくし、トルクを弱め減速します。

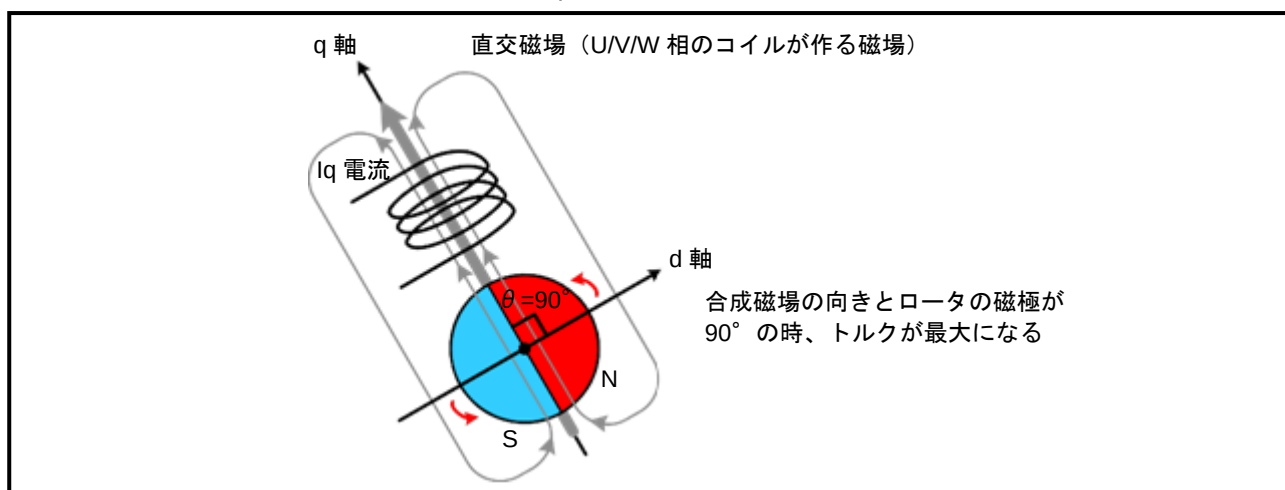


図 1.2 合成磁場とトルクの関係

1.2 モータ制御の概念

モータを効率良く制御するためには、回転速度に応じて d 軸、q 軸電流を調整する必要がありますので、この d 軸、q 軸電流を求めます。q 軸電流は流れる電流の大きさと流れる向きがあり、ベクトル量として取り扱われるため、この制御方法をベクトル制御と呼びます。

目標の回転速度と現在の回転速度の差分から d 軸、q 軸電流の目標値(指令値)を決めます。簡単なモデルではトルク成分に寄与しない d 軸電流を 0 とみなして処理します。d 軸、q 軸の各電流指令値と、測定したコイル電流から求めた d 軸、q 軸電流値の差分(偏差)から次の電圧値を求めます。このように、前回の結果(現状のモータ回転速度やコイル電流値)を帰還させて次回の出力に反映する制御をフィードバック制御と呼びます。

現在の電流値の偏差から次回の電圧値を求めるには、現在の偏差に比例した成分と、過去の累積した偏差に比例した成分を加算します。この処理を繰り返す行うことにより出力電圧、その結果流れるコイル電流を調整し、目標の回転速度に合わせ込みます。この方法を PI 制御(Proportional-Integral control)と呼びます。

コイル電流値を座標変換(3相2相変換→回転座標変換)することにより、q 軸電流値と d 軸電流値を求めます。図 1.3 に座標変換を示します。

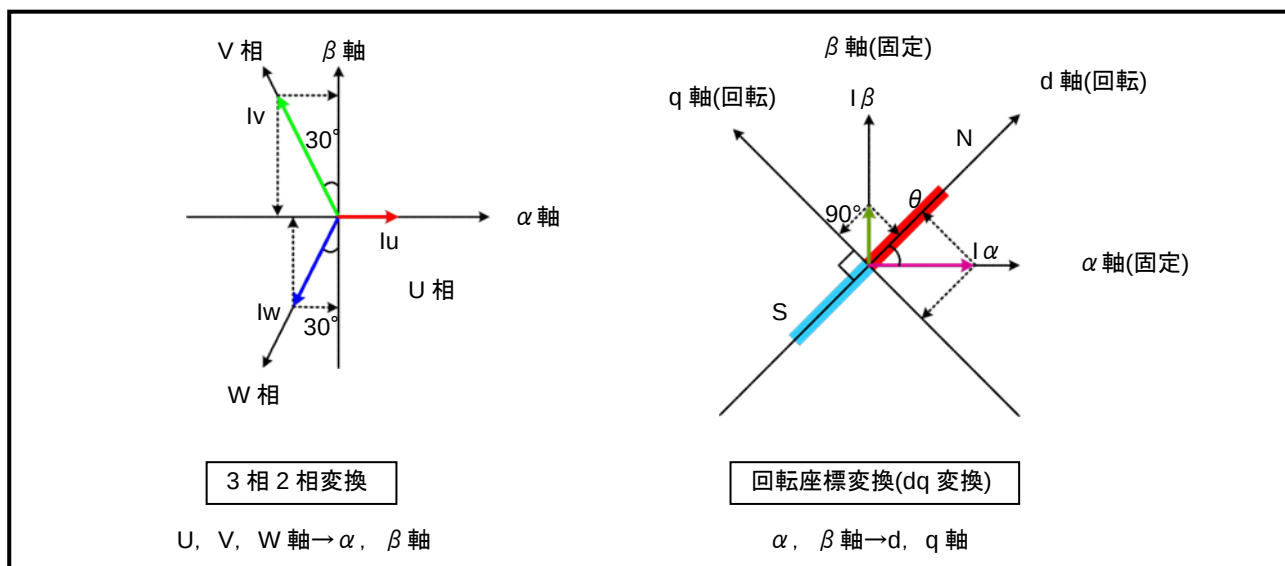


図 1.3 座標変換

n 回目に測定した U/V/W 各コイルに流れる電流値 I_u 、 I_v 、 I_w を 3 相 2 相変換することにより α 、 β 軸の電流値 I_α 、 I_β を求めます。

$$\begin{pmatrix} I_\alpha(n) \\ I_\beta(n) \end{pmatrix} = \sqrt{\frac{2}{3}} \begin{pmatrix} \cos 0 & \cos \frac{2}{3}\pi & \cos \frac{4}{3}\pi \\ \sin 0 & \sin \frac{2}{3}\pi & \sin \frac{4}{3}\pi \end{pmatrix} \begin{pmatrix} I_u(n) \\ I_v(n) \\ I_w(n) \end{pmatrix}$$

3 相 2 相変換で得た I_α 、 I_β を回転座標変換することにより d、q 軸の電流値 I_d と I_q を求めます。

$$\begin{pmatrix} I_d(n) \\ I_q(n) \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} I_\alpha(n) \\ I_\beta(n) \end{pmatrix}$$

d 軸電流と q 軸電流の偏差を求め、PI 制御により d 軸電圧、q 軸電圧を求めます。

$$\varepsilon_d(n) = I_{dt} - I_d(n)$$

$$\varepsilon_q(n) = I_{qt} - I_q(n)$$

$$V_d(n) = V_d(n-1) + K_P \cdot \{\varepsilon_d(n) - \varepsilon_d(n-1)\} + K_I \cdot \varepsilon_d(n) \cdot \Delta t$$

$$V_q(n) = V_q(n-1) + K_P \cdot \{\varepsilon_q(n) - \varepsilon_q(n-1)\} + K_I \cdot \varepsilon_q(n) \cdot \Delta t$$

備考	I_{dt}	: d 軸電流の指令値
	I_{qt}	: q 軸電流の指令値
	$I_d(n)$	: n 回目にサンプリングした d 軸電流
	$I_q(n)$	: n 回目にサンプリングした q 軸電流
	K_P	: 比例ゲイン
	K_I	: 積分ゲイン
	Δt	: サンプリング時間
	ε_d	: d 軸電流の偏差
	ε_q	: q 軸電流の偏差

ここで算出した q 軸電圧、d 軸電圧を座標変換(固定座標変換、2 相 3 相変換)することにより、U/V/W 各相の出力電圧を求めます。固定座標変換は、次式になります。

$$\begin{pmatrix} V_{\alpha}(n) \\ V_{\beta}(n) \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} V_d(n) \\ V_q(n) \end{pmatrix}$$

2 相 3 相変換は、次式になります。

$$\begin{pmatrix} V_u(n) \\ V_v(n) \\ V_w(n) \end{pmatrix} = \sqrt{\frac{2}{3}} \begin{pmatrix} \cos 0 & \sin 0 \\ \cos \frac{2}{3}\pi & \sin \frac{2}{3}\pi \\ \cos \frac{4}{3}\pi & \sin \frac{4}{3}\pi \end{pmatrix} \begin{pmatrix} V_{\alpha}(n) \\ V_{\beta}(n) \end{pmatrix}$$

1.3 U/V/W 相電圧から PWM 出力へ

マイクロコントローラから実際に PWM 波形を出力するためには、操作量である d 軸電圧、q 軸電圧を三相電圧変換した U/V/W 相電圧(V_u , V_v , V_w)を PWM のデューティに反映させる必要があります。その方法について説明します。

1.3.1 三角波比較法

三角波比較法は PWM 制御をおこなう制御方法の一つで、図 1.4 に示すように、U/V/W 相電圧波形と三角波の大小比較で PWM 生成をおこないます。この方法はアナログ回路で PWM 生成した時代のなごりですが、今なお最も一般的に広く使われています。この三角波をキャリア波(搬送波)と呼び、その周波数は PWM の動作周期となります。U/V/W 相電圧と三角波を比較し、U/V/W 相電圧が三角波より大きいとき PWM をハイレベル(アクティブレベル)にし、三角波より小さいときはロウレベル(インアクティブレベル)にします。これにより、平均電圧を変化させていけば、図 1.4 の U/V/W 相電圧(正弦波)を疑似的に再現できます。詳細は、専門書を参照してください。

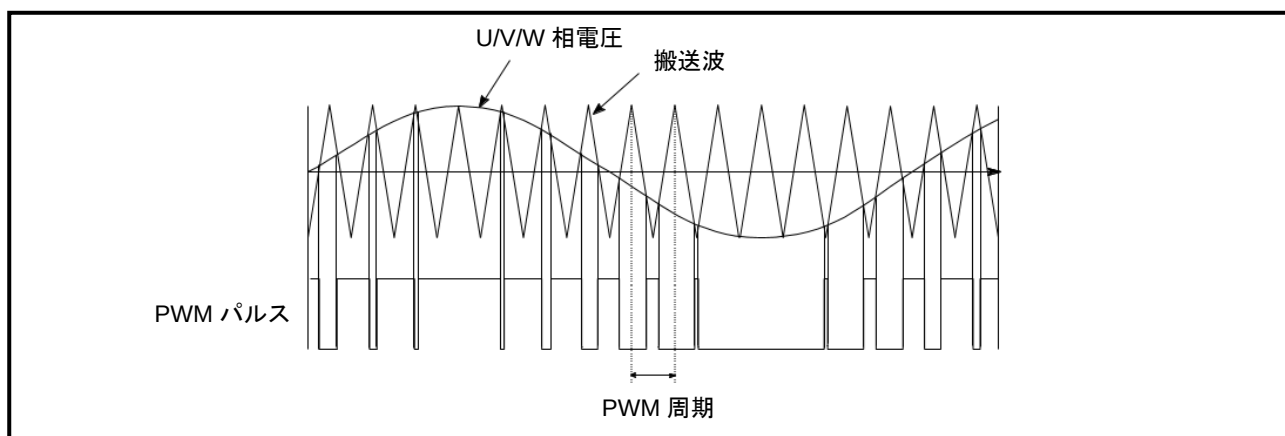


図 1.4 三角波比較法

この三角波比較法は、三角波変調とも呼びます。この三角波比較法を使い、永久磁石同期モータの 3 相電圧を PWM 制御すれば良いことになります。この PWM 出力は、相補 PWM 出力(デッドタイム付き PWM 出力)で行います。

1.4 dq 座標系の PMSM の電圧方程式(参考資料)

1.4.1 電圧方程式

dq 座標系における PMSM の電圧方程式は以下の式になります。電圧方程式を用いて、電流 I_d , I_q を電圧 V_d , V_q に変換できます。ただし、本サンプルプログラムでは、PI 制御を用いて電圧に変換しています。PI 制御に関しては、「3.4.5.4(1)PI 制御」を参照してください。

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} R_a + pL_d & -\omega L_q \\ \omega L_d & R_a + pL_q \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} 0 \\ \omega \phi_a \end{bmatrix} \quad p = \frac{d}{dt}$$

備考 v_d, v_q : 各相電機子電圧

R_a : 各相電機子抵抗

L_d, L_q : 各相自己インダクタンス

i_d, i_q : 各相電機子電流

ω : モータ角速度

ϕ : 永久磁石磁束 $\phi_a = \sqrt{\frac{2}{3}}\phi$

p : 微分演算子

この式の導出過程の詳細については、専門書を参照してください。

実際の制御では、この式は使用しておりません (非干渉制御導出にのみ使用)。

1.4.2 非干渉制御

1.4.1 に示されたモータの電圧方程式にラプラス変換を施し、変形すると以下の式になります。

$$I_d = \frac{1}{R_a + sL_d}(V_d + \omega L_q I_q)$$

$$I_q = \frac{1}{R_a + sL_q} \left(V_q - \omega \left(L_d I_d + \sqrt{\frac{2}{3}} \phi \right) \right)$$

備考 s : ラプラス演算子

この式から分かるように q 軸側の情報が d 軸側に、d 軸側の情報が q 軸側にあらわれています。この情報をあらかじめ引いておく制御方法を「非干渉制御」と言います。

この非干渉制御を行う場合、モータのパラメータを前もって知っておく必要があります。また、この制御はフィードフォワード制御となります。

実際の電流制御は PI 制御を用いますので、各電流制御(I_d , I_q)が安定していない時に他の項の影響があらわれると PI 制御の調整が難しくなります。そのため、非干渉制御を用います。

2. EMU3S の代表的な制御例

本アプリケーションノートでは、12 ビット A/D コンバータ(ADCK)、R/D コンバータ(RDC3AL)、エンハンスドモータコントロールユニット 3S(EMU3S)、モータ制御タイマ(TSG3)、ペリフェラルインタコネクション(PIC)を用いてモータ制御をおこないます。

A/D コンバータでモータ電流値を測定し、R/D コンバータで角度情報を取得します。EMU3S はモータ電流値と角度情報をもとにベクトル制御をおこない、PWM コンペア値を生成します。そのコンペア値をもとに TSG3 が三角波比較方式で正弦波 PWM 波形を生成し出力します。

EMU3S の代表的な制御例を以下に示します。

- ・ 全自動処理動作

CPU が介在しないでモータ制御を行う場合の制御例です。

詳細は「3.全自動処理動作」を参照してください。

- ・ CPU 処理のある制御動作

EMU3S の機能として、ユーザ独自の処理を挿入することもできます。

詳細は「4.CPU 処理のある制御動作」を参照してください。

- ・ 矩形波出力動作

EMU3S の機能として、角度に応じた矩形波を出力する場合の制御例です。

詳細は「5.矩形波出力動作」を参照してください。

3. 全自動処理動作

3.1 仕様

本章では、CPU は電流指令値の設定のみを行い、その他の制御は CPU を介さずにハードウェアのみでおこなっています。

表 3.1 にモータ制御の仕様を、表 3.2 に各モジュールの設定内容を、図 3.1 に 2 モータ対応のシステム構成図(RH850/U2B6)を、図 3.2 に全自動処理の制御フロー図を示します。

表 3.1 モータ制御の仕様

項目	仕様
出力波形	相補三相 PWM 波形
キャリア周期	10 kHz (100 usec/サイクル)
制御方式	180 度通電駆動方式
アクティブレベル	High アクティブ
短絡防止時間	4 usec
コンペアレジスタ、キャリア周期の更新タイミング	キャリア波の谷
A/D 変換開始タイミング	キャリア波の谷
割り込み	使用する(100us 割り込み)
モータ電流	ADCK0 で U 相、V 相の電流値を取得
モータの角度情報	RDC3AL でレゾルバからの角度情報を取得

表 3.2 各モジュールの設定内容

周辺機能	設定内容
RDC3AL	・ RDC3AL 内部で生成した 10kHz の励磁信号を使用 ・ PI 補償器帯域は自動設定 ・ BIST, 自己診断は実施しない
PIC	・ ADCK0 スキャングループ 4 のトリガに EMU30 からの信号を使用
ADCK	・ スキャングループ 4(SG4)の仮想チャンネル 0(V 相),1(W 相),2(U 相)を A/D 変換 ・ T&HA/D 変換モード ・ マルチサイクルスキャンモードでトリガ毎に 1 回スキャン ・ スキャングループ 4 ハードウェアトリガ入力有効 ・ スキャングループ 4 終了割り込み信号の出力禁止 ・ EMU3S への A/D 変換結果転送機能
TSG3	・ HT-PWM モード ・ キャリア周期 : 100us ・ デッドタイム : 4us ・ キャリア周期と PWM デューティ設定を EMU3S から直接転送
EMU3S	・ 入力 IP は AD 完了トリガで自動的に起動、他の各演算部の IP は前段の IP 完了で自動的に起動 ・ 各演算には前段 IP の演算結果を使用 ・ ADCK からの電流値と RDC3AL からの角度値、角速度値を演算に使用 ・ キャリア波の谷タイミングで A/D 変換起動 ・ A/D 変換と R/D 変換のトリガ出力遅延 1usec

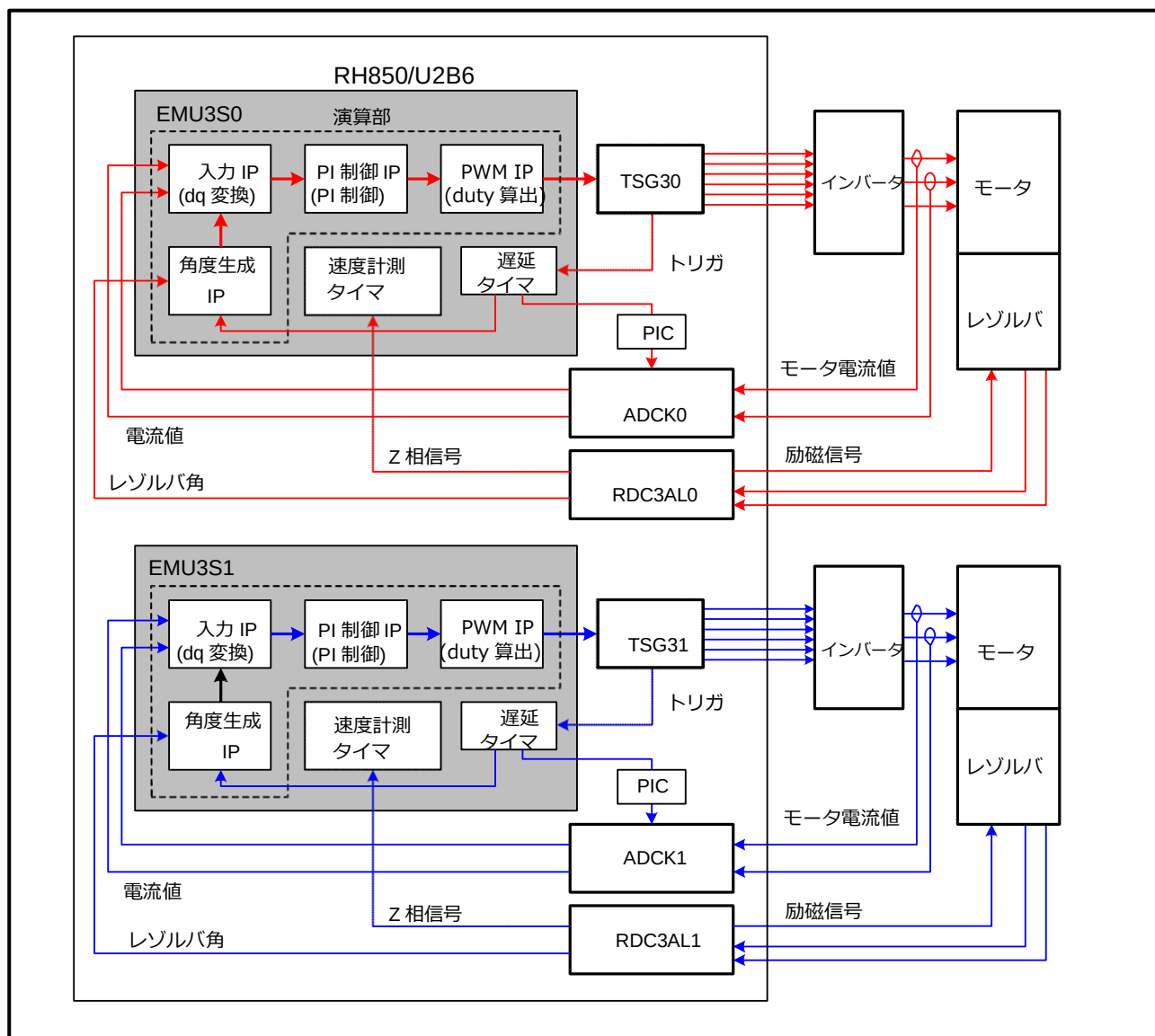


図 3.1 2 モータ対応のシステム構成図 (RH850/U2B6)

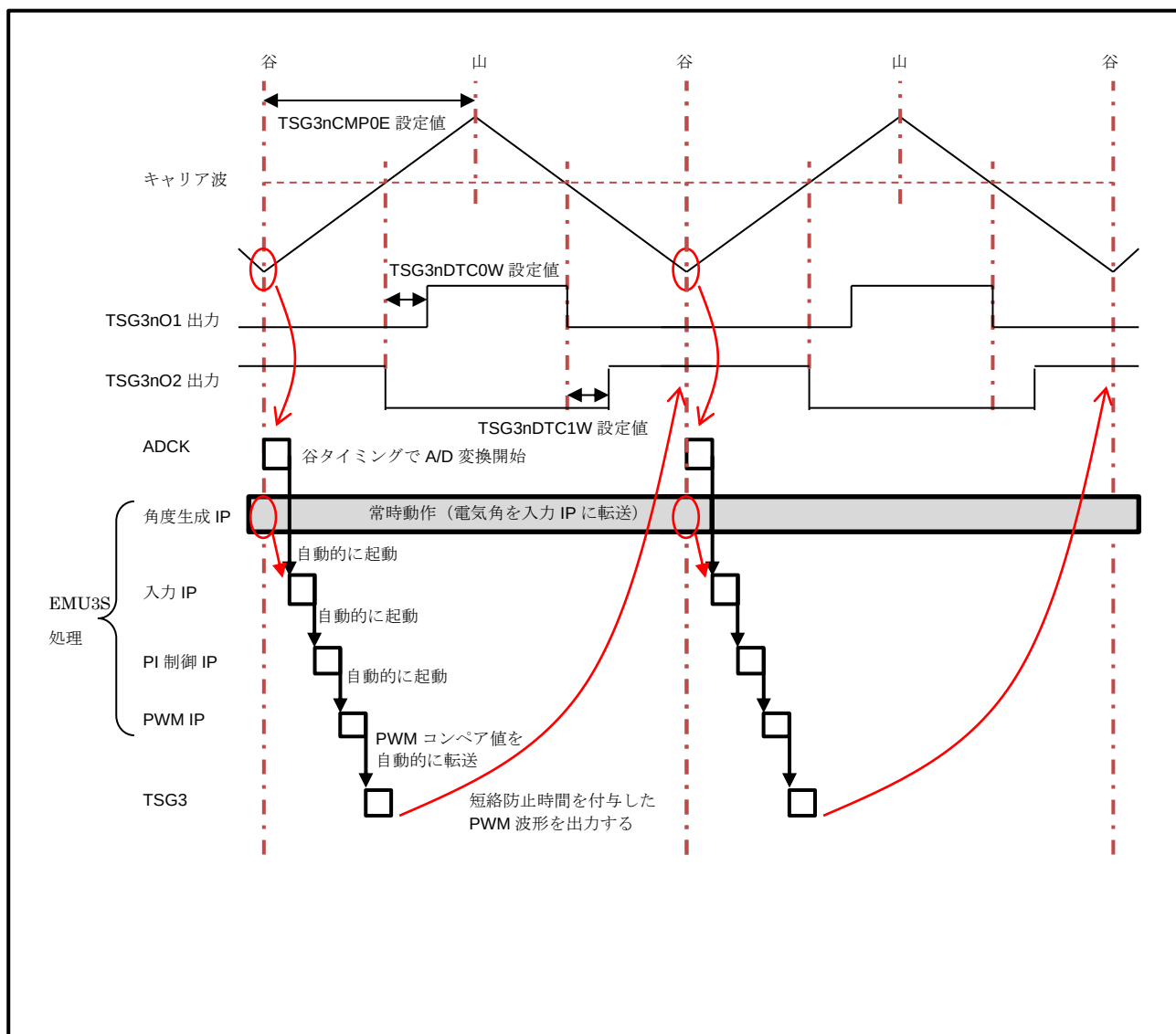


図 3.2 全自動処理の制御フロー図

3.2 動作条件

本アプリケーションノートのサンプルコードは、下記の条件で動作を確認しています。

表 3.3 動作確認条件

項目	内容
使用マイコン	RH850/U2B6
動作周波数	・メイン OSC : 20MHz ・CPU クロック : 400MHz - 非変調高速周辺クロック : 80MHz - 非変調低速周辺クロック : 40MHz - モータ制御 H/W アクセラレータクロック : 100MHz
動作電圧	・VDD=1.12V ・VCC=5.0V
統合開発環境	ルネサスエレクトロニクス製 CS+ V8.07.00
C コンパイラ	ルネサスエレクトロニクス製 RH850 ファミリ用 C コンパイラパッケージ V2.02.00 コンパイルオプション 統合開発環境のデフォルト設定を使用しています
サンプルコードのバージョン	Version 1.00

3.3 端子

表 3.4 に使用端子と機能を示します。

表 3.4 使用端子と機能

モジュール	端子名	入出力	機能
ADCK	AN000	入力	V 相電流の測定
	AN001	入力	W 相電流の測定
	AN002	入力	U 相電流の測定
RDC3AL	RDC3AL0S1	入力	レゾルバ信号入力($\cos \theta$)
	RDC3AL0S3	入力	レゾルバ信号入力($\cos \theta$)
	RDC3AL0S2	入力	レゾルバ信号入力($\sin \theta$)
	RDC3AL0S4	入力	レゾルバ信号入力($\sin \theta$)
	RDC3AL0RSO	出力	励磁信号出力
	RDC3AL0COM	出力	励磁信号用コモン電圧出力
TSG3	TSG30O1	出力	上アーム U 相出力
	TSG30O2	出力	下アーム U 相出力
	TSG30O3	出力	上アーム V 相出力
	TSG30O4	出力	下アーム V 相出力
	TSG30O5	出力	上アーム W 相出力
	TSG30O6	出力	下アーム W 相出力

3.4 動作概要

モータ制御の動作を説明します。

- (1) キャリア波の“谷”タイミングで EMU3S から PIC 経由で A/D コンバータにトリガ信号が入力されると、ADCK0 グループ 4 の仮想チャンネル 0~3 を A/D 変換しモータの U 相と V 相の電流値を取得します。同時に、EMU3S 演算部の角度生成 IP が RDC3AL から取得したレゾルバ角をモータの電気角に変換し、演算部の入力 IP に転送します。
- (2) A/D 変換が完了すると、変換結果が EMU3S のレジスタに格納され、入力 IP が自動的に起動します。入力 IP は、モータ電流値と電気角をもとに dq 変換などの演算を行い、d 軸電流と q 軸電流の帰還値を算出します。
- (3) 入力 IP の演算が完了すると、演算部の PI 制御 IP が自動的に起動します。PI 制御 IP は、d 軸電流と q 軸電流の帰還値とソフトウェアで設定した各指令値をもとに PI 制御を行い、d 軸電圧と q 軸電圧を算出します。
- (4) PI 制御 IP の演算が完了すると、演算部の PWMIP が自動的に起動します。PWMIP は、d 軸電圧と q 軸電圧をもとに U/V/W 相の PWM コンペア値(デューティ値)を算出します。また RDC3AL から取得される角速度で非干渉制御処理を行います。
- (5) PWM コンペア値は、TSG3 に転送され、次のリロードタイミングで TSG3 のコンペアレジスタに設定されます。そして対応する TSG3 端子から三相 PWM 波形が出力されます。
- (6) OSTM の周期割り込みで EMU3S に対し電流指令値を設定します。これにより、制御するモータに回転指令があたえられます。

3.4.1 R/D コンバータ 3AL (RDC3AL)

レゾルバから出力されるアナログ信号をデジタル変換し、角度値(レゾルバ角)、角速度を算出します。

3.4.2 ペリフェラルインタコネクション (PIC)

EMU3S からキャリア波に同期して出力される信号をもとに、ADCK ハードウェアトリガ信号を生成します。

3.4.3 A/D コンバータ(ADCK)

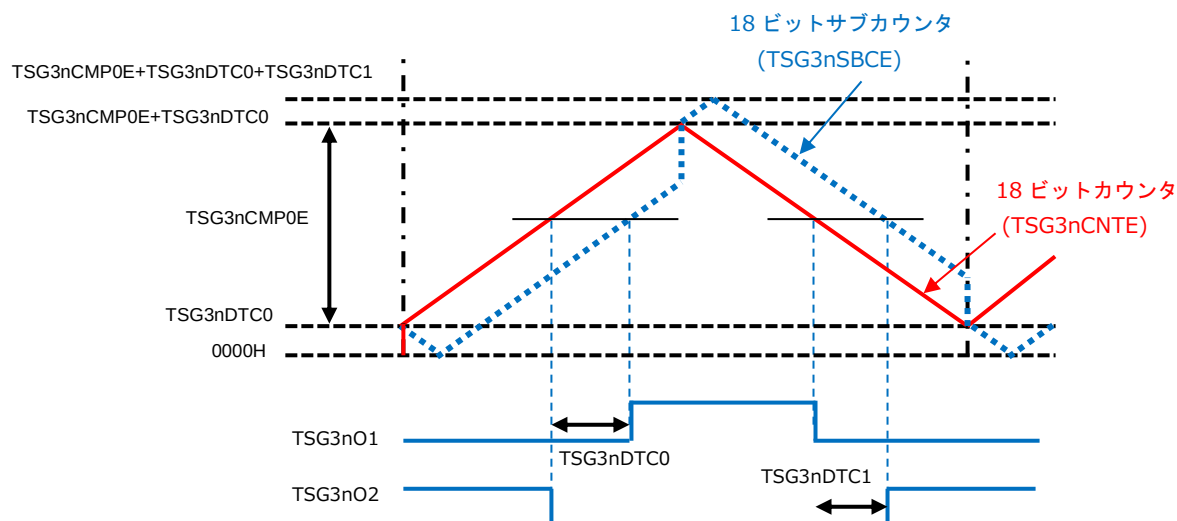
モータ電流値を A/D 変換し、変換値を EMU3S に入力します。本アプリケーションノートでは、EMU3S からの出力信号に応じて PIC が生成するトリガにより A/D 変換を開始させます。トリガが入力されると、AN000 で V 相の電流値を、AN001 で W 相の電流値を、AN002 で U 相の電流値をそれぞれ変換します。W 相の電流値は処理では使用しません。

3.4.4 モータコントロールタイマ 3 (TSG3)

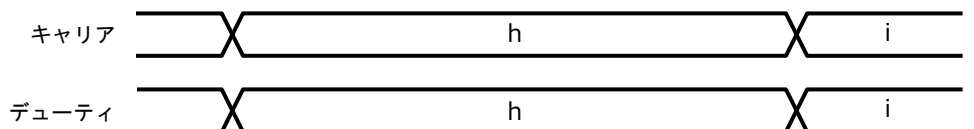
TSG3 は EMU3S から転送されたコンペア値をもとに三相 PWM 波形を出力します。

TSG3 は 18 ビットカウンタと 18 ビットサブカウンタとのコンペア一致 (EMU3S との組み合わせでは 16bit 分を使用) からデューティを生成し、短絡防止時間を付加した三相 PWM 波形を出力します。

図 3.3 に三相 PWM 波形タイミング図を示します。



【反映タイミング：キャリア谷更新】



TSG3nCMP0E: PWM 周期設定

TSG3nDTC0W: デッドタイム値(逆相インアクティブから正相アクティブ間)を設定

TSG3nDTC1W: デッドタイム値(正相インアクティブから逆相アクティブ間)を設定

TSG3nCNTE: 18 ビットカウンタリードバッファレジスタ

TSG3nSBCE: 18 ビットサブカウンタリードバッファレジスタ

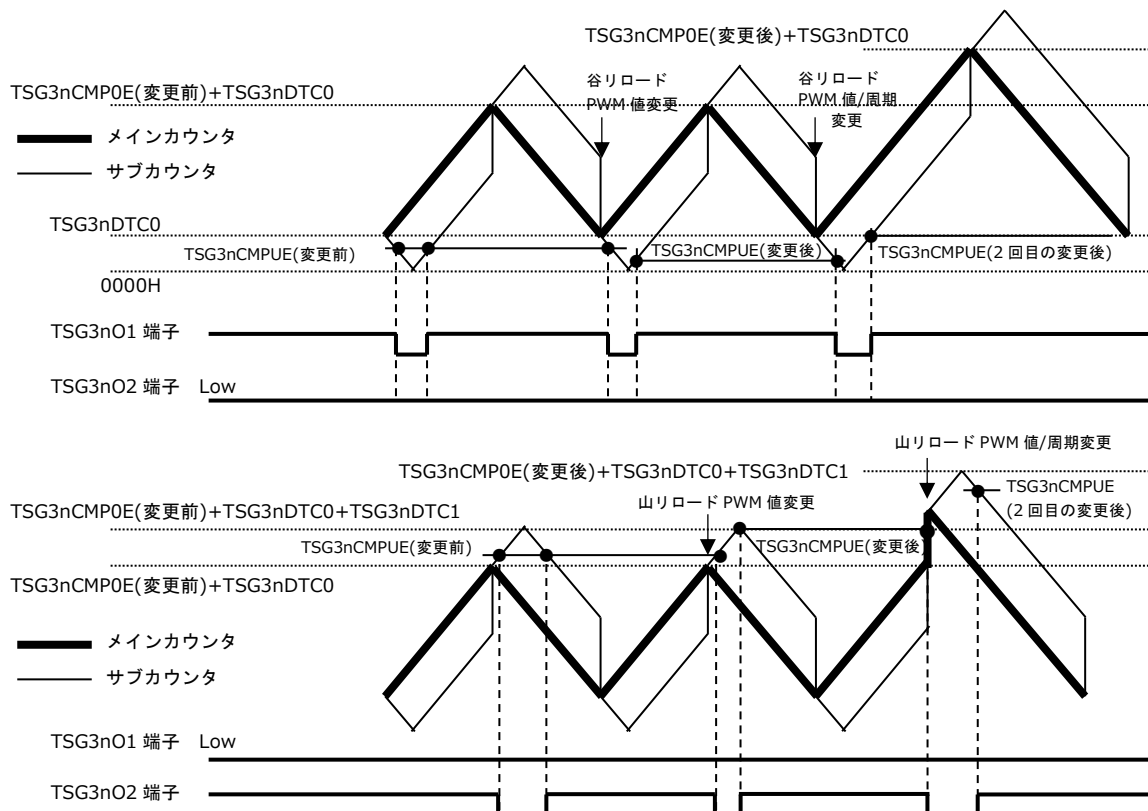


図 3.3 三相 PWM 波形タイミング図

3.4.5 エンハンスドモータコントロールユニット 3S (EMU3S)

3.4.5.1 EMU3S 共通部

EMU3S はモータ電流値と角度情報をもとにベクトル制御をおこない、PWM コンペア値を生成します。本アプリケーションノートでは、電流指令設定処理以外、CPU を介さずにハードウェアのみで制御をおこなっています。

【関連レジスタの設定値】

- ・ EMU3nCTR レジスタの EMUST ビット = “1” (EMU3n 動作)
- ・ EMU3nIPTRG レジスタの INIPTRG[1:0] = “10B” (A/D 変換完了タイミングで入力 IP を起動)
- ・ EMU3nIPTRG レジスタの PIIPTRG = “1” (入力 IP 完了タイミングで PI 制御 IP を起動)
- ・ EMU3nIPTRG レジスタの PWMIPTRG = “1” (PI 制御 IP 完了タイミングで PWMIP を起動)
- ・ EMU3nSMCTR レジスタの SMVAN = “1” (キャリア谷タイミングで A/D 起動)
- ・ EMU3nADDCNT レジスタの ADDATA = “80” (A/D 変換の遅延時間 1μsec)
- ・ EMU3nRDDCNT レジスタの RDDATA = “80” (R/D 変換トリガ出力の遅延時間 1μsec)
- ・ EMU3nINT0~7 レジスタ = “00000000H” (割り込み禁止)
- ・ INTC2EIC38~45 レジスタ = “00CFH” (割り込み処理を禁止)

3.4.5.2 角度生成 IP

角度生成 IP は、RDC3AL からの角度データの値(RDC3ALnENC1 レジスタの上位 16 ビット)が変化すると起動します。そして RDC3ALnENC1 レジスタの値にオフセット値(EMU3nANGOFS レジスタ)を加算してから電気角を生成します。生成された電気角は EMU3nTHTEFIX レジスタに格納されます。

図 3.4 に角度生成 IP のブロック図を示します。

【関連レジスタの設定値】

- ・ EMU3nANGCTR レジスタ = “00H” (RDC3AL から角度データ/Z 相パルスを入力する)
- ・ EMU3nRESRLD レジスタ = “01H” (レゾルバ極数-1 を設定)
例)レゾルバ極=2 極の場合、EMU3nRESRLD レジスタ = “01H”
- ・ EMU3nANGOFS レジスタ = “0000H” (角度オフセット値設定)
- ・ EMU3nPXR レジスタ = “0100H” (レゾルバ角 : 電気角=1 : 1)
例)レゾルバ角 : 電気角=1 : 2 の場合、EMU3nPXR レジスタ = “0200H”

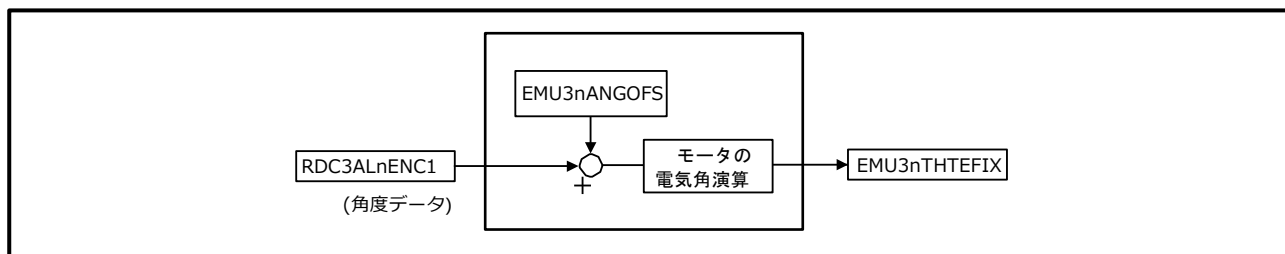


図 3.4 角度生成 IP のブロック図

3.4.5.3 入力 IP

EMU3nIPTRG レジスタの INIPTRG[1:0]ビットを “B’ 10” に設定した場合、A/D 変換が完了すると入力 IP が自動的に起動します。本アプリケーションノートでは、入力 IP は二相のモータ電流値から残り一相の電流値を求め、さらに求めた電流値と電気角をもとに dq 変換を行います。EMU3nTHTEFIX レジスタに格納された値は、A/D トリガ発生時に入力 IP へ転送され、EMU3nTHTE レジスタに格納されます。図 3.5 に入力 IP のブロック図を示します。

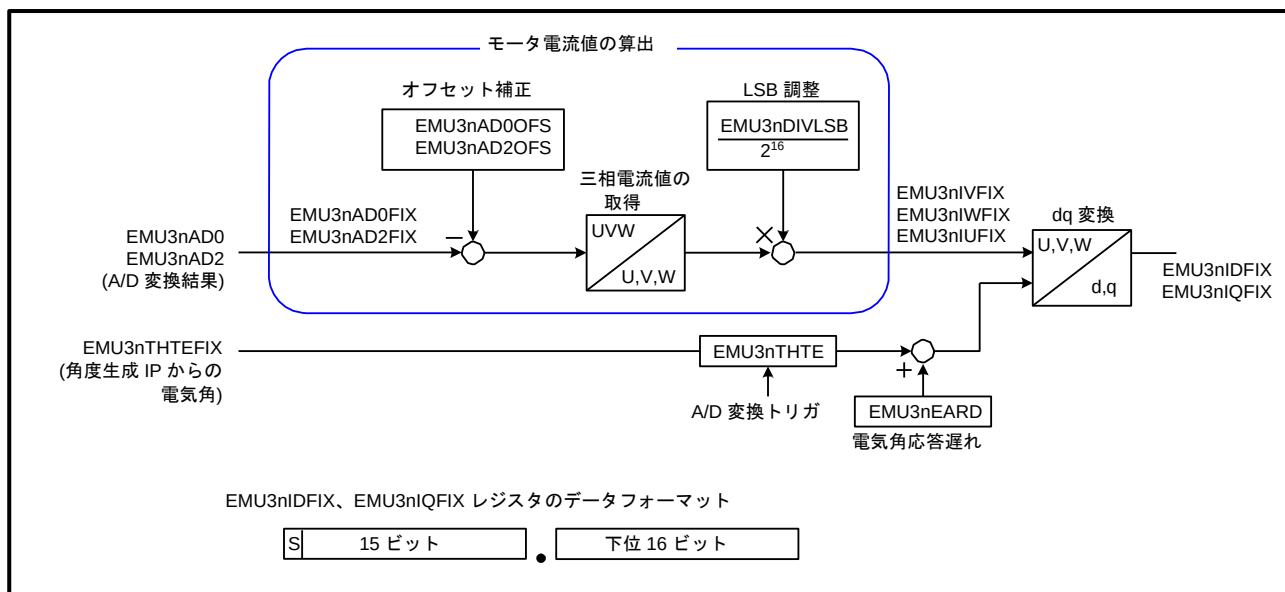


図 3.5 入力 IP のブロック図

(1) モータ電流値算出

EMU3nAD0 レジスタと EMU3nAD2 レジスタに格納された二相の電流値から、三相の電流値を算出します。結果は EMU3nIVFIX、EMU3nIWFIX、EMU3nIUFIX に格納されます。A/D 変換値のオフセット補正用レジスタ(EMU3nAD0OFS,EMU3nAD2OFS)、LSB 調整レジスタ(EMU3nDIVLSB)を設定すると、入力 IP が設定値に応じて演算を行います。

【関連レジスタの設定値】

- ・ EMU3nAD0OFS レジスタ=(モータ電流が 0 のときのオフセット値を入力)
- ・ EMU3nAD2OFS レジスタ=(モータ電流が 0 のときのオフセット値を入力)
- ・ EMU3nDIVLSB レジスタ = “00010000H”
- ・ EMU3nCTRINMD レジスタの CMES ビット = “1” (CMUVW[2:0]ビットで選択)
- ・ EMU3nCTRINMD レジスタの CMUVW[2:0]ビット = “100B” (2 相(U 相と V 相)を測定)

【入力 IP による演算】

0~4095 の A/D 変換値をオフセット補正し、-2048~2047 の値に変換します。図 3.6 に A/D 変換結果のオフセット補正説明図を示します。

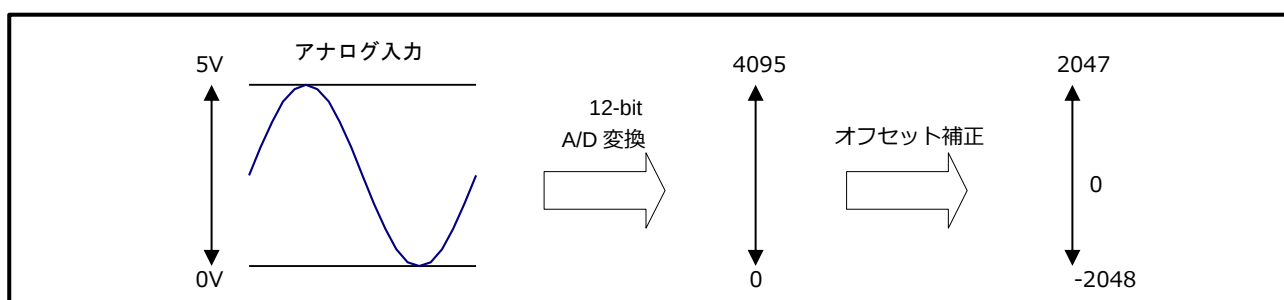


図 3.6 A/D 変換結果のオフセット補正

$$\text{EMU3nIUFIX} \leftarrow ((\text{EMU3nAD0FIX} - \text{EMU3nAD0OFS}) \times \text{EMU3nDIVLSB}) \gg 16$$

$$\text{EMU3nIVFIX} \leftarrow ((\text{EMU3nAD1FIX} - \text{EMU3nAD1OFS}) \times \text{EMU3nDIVLSB}) \gg 16$$

$$\text{EMU3nIWFIX} \leftarrow -(\text{EMU3nIUFIX} + \text{EMU3nIVFIX})$$

(2) dq 変換

EMU3nIUFIX レジスタ、EMU3nIVFIX レジスタ、EMU3nIWFIX レジスタに格納された U/V/W 相電流値と EMU3nTHTE レジスタに格納された電気角を用いて、d/q 軸電流変換を行います。その結果、d 軸電流値が EMU3nIDFIX レジスタに、q 軸電流値が EMU3nIQFIX レジスタに格納されます。

【関連レジスタの設定値】

- ・ EMU3nCTRINMD レジスタの FREGIN ビット = “1” (角度生成 IP で生成された電気角を使用)
- ・ EMU3nSR2 レジスタ = “0000D106H” (d/q 軸電流変換係数)

【入力 IP による演算】

$$\begin{pmatrix} \text{EMU3nIDFIX} \\ \text{EMU3nIQFIX} \end{pmatrix} = \text{EMU3nSR2} \times \begin{pmatrix} \sin(\theta_e + 90^\circ) & -\sin(\theta_e + 150^\circ) & -\sin(\theta_e + 30^\circ) \\ -\sin(\theta_e + 0^\circ) & \sin(\theta_e + 60^\circ) & -\sin(\theta_e + 120^\circ) \end{pmatrix} \begin{pmatrix} \text{EMU3nIUFIX} \\ \text{EMU3nIVFIX} \\ \text{EMU3nIWFIX} \end{pmatrix}$$

3.4.5.4 PI 制御 IP

EMU3nIPTRG レジスタの PIIPTRG ビットを“1”に設定した場合、入力 IP の処理が完了すると、PI 制御 IP が自動的に起動します。PI 制御 IP は、dq 変換で算出された d 軸電流(帰還値)と q 軸電流(帰還値)と、ソフトウェアで設定した d 軸電流(指令値)と q 軸電流値(指令値)を用いて、d 軸電圧と q 軸電圧を算出します。図 3.7 に PI 制御 IP のブロック図を示します。

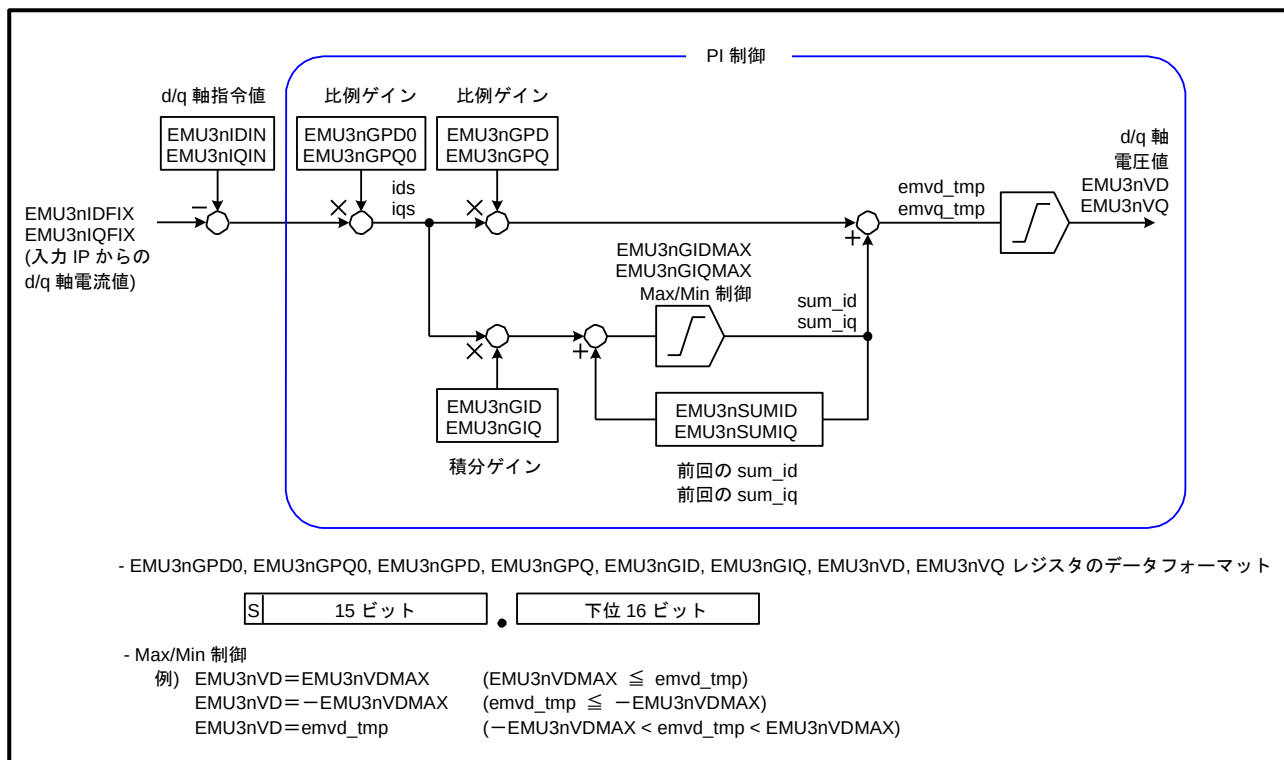


図 3.7 PI 制御 IP のブロック図

(1) PI 制御

EMU3nIDFIX、EMU3nIQFIX レジスタに格納された帰還値と EMU3nIDIN、EMU3nIQIN レジスタに設定した指令値から、操作量である d 軸電圧と q 軸電圧を算出します。その結果、d 軸電圧が EMU3nVD レジスタに、q 軸電圧が EMU3nVQ レジスタに格納されます。

【関連レジスタの設定値】

- ・ EMU3nPICTR レジスタの FSUMIQ ビット = “1” (q 軸積分項は EMU 演算結果を使用)
- ・ EMU3nPICTR レジスタの FSUMID ビット = “1” (d 軸積分項は EMU 演算結果を使用)
- ・ EMU3nIDIN レジスタ: d 軸の指令値を設定する
- ・ EMU3nIQIN レジスタ: q 軸の指令値を設定する
- ・ EMU3nGPD0 レジスタ = “0001 0000H” (d 軸の比例ゲインは 1)
- ・ EMU3nGPQ0 レジスタ = “0001 0000H” (q 軸の比例ゲインは 1)
- ・ EMU3nGID レジスタ = “0000 0100H” (d 軸の積分ゲインは 1/256)
- ・ EMU3nGIQ レジスタ = “0000 0100H” (q 軸の積分ゲインは 1/256)
- ・ EMU3nGPD レジスタ = “0001 0000H” (d 軸の比例ゲインは 1)

- ・ EMU3nGPQ レジスタ = “0001 0000H” (q 軸の比例ゲインは 1)
- ・ EMU3nGIDMAX レジスタ = “0000 0800H” (d 軸の積分項最大値)
- ・ EMU3nGIQMAX レジスタ = “0000 0800H” (q 軸の積分項最大値)
- ・ EMU3nVDMAX レジスタ = “7FFF FFFFH” (d 軸の操作量最大値)
- ・ EMU3nVQMAX レジスタ = “7FFF FFFFH” (q 軸の操作量最大値)

3.4.5.5 PWM IP

EMU3nIPTRG レジスタの PWMIPTRG ビットを“1”に設定した場合、PI IP の処理が完了すると、PWM IP が自動的に起動します。PWM IP は、PI 制御 IP で算出された d 軸電圧と q 軸電圧を用いて、U/V/W 相の出力電圧を求め、その電圧値から PWM 波形出力における各相のデューティ比とコンパレクタ値を算出します。図 3.8 に PWM IP のブロック図を示します。EMU3nPWMCTR の SETDEC ビットを“1”に設定した場合、d 軸電圧、q 軸電圧に非干渉制御が、“0”の場合は電圧補正が施されます。

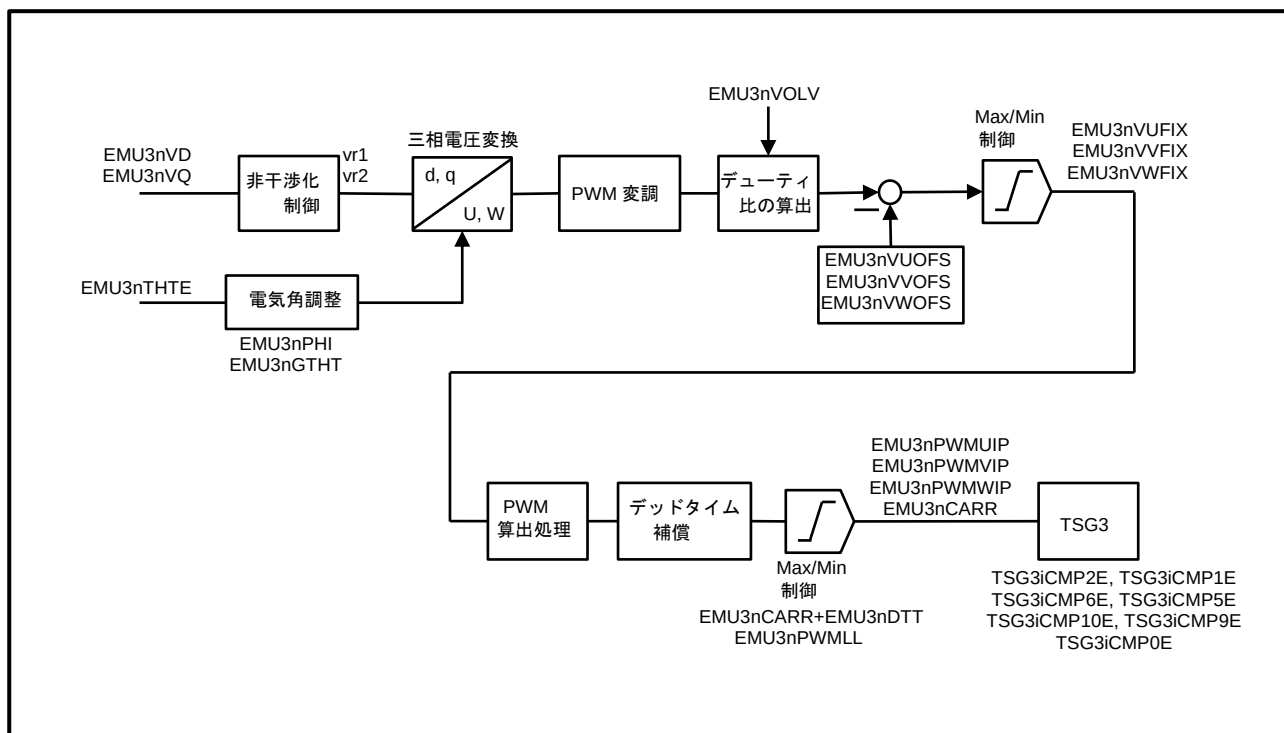


図 3.8 PWM IP のブロック図

(1) d/q 軸電圧補正/非干渉制御

PI 制御 IP で算出した d 軸電圧と q 軸電圧に対して補正、あるいは非干渉補正を行うことができます。補正値は制御の必要に応じて設定してください。また軸電圧補正/非干渉制御を行う場合は、EMU 内部回路へレジスタ値を反映するために、EMU3nREFCTR レジスタの FPWMREFPER ビットを“1”に設定してください。

なお、d 軸電圧と q 軸電圧の最大値は EMU3nREFCTR レジスタの FPWMREFPER ビットの設定によらず、必ず設定してください。

【関連レジスタの設定値】

- ・ EMU3nVD2MAX レジスタ = “7FFF FFFFH”
- ・ EMU3nVQ2MAX レジスタ = “7FFF FFFFH”

・ 電圧補正

EMU3nVDCRCT レジスタ、EMU3nVQCRCT レジスタで補正を行います。各パラメータは制御に応じて、適切な値を設定してください。本サンプルプログラムでは、値に 0 が設定されているため、補正量は 0（補正しない）です。

【関連レジスタの設定値】

- ・ EMU3nVDCRCT レジスタ = “0000 0000H”（補正しない）
- ・ EMU3nVQCRCT レジスタ = “0000 0000H”（補正しない）

また、EMU3nGPD0、EMU3nGPQ0 レジスタに“0”を設定することによって、PI 制御を使用せず、EMU3nVDCRCT と EMU3nVQCRCT レジスタから三相電圧変換のパラメータを設定することも可能です。

$$vdi \leftarrow 0 + \text{EMU3nVDCRCT}$$

$$vqi \leftarrow 0 + \text{EMU3nVQCRCT}$$

・ 非干渉制御

EMU3nDECVELG レジスタ、EMU3nDECFLUX レジスタ、EMU3nDECLD レジスタ、EMU3nDECLQ レジスタ及び、RDC3AL から得られた角速度、d 軸電流、q 軸電流を用いて、d 軸電圧と q 軸電圧は次式で補正されます。

各パラメータは制御に応じて、適切な値を設定してください。本サンプルプログラムでは、値に 0 が設定されているため、補正量は 0（補正しない）です。

$$d \text{ 軸電圧} \leftarrow d \text{ 軸電圧} - (we \times Gwe \times Lq \times Iq)$$

$$q \text{ 軸電圧} \leftarrow q \text{ 軸電圧} + (we \times Gwe \times Ld \times Id + we \times Gwe \times \Phi)$$

(we: 電気角の角速度、Φ: 磁束、I: 電流、L: インダクタンス、Gwe: 角速度ゲイン)

【関連レジスタの設定値】

- ・ EMU3nDECVELG レジスタ = “0000 0000H”
- ・ EMU3nDECFLIX レジスタ = “0000 0000H”
- ・ EMU3nDECLD レジスタ = “0000 0000H”
- ・ EMU3nDECLQ レジスタ = “0000 0000H”

(2) 三相電圧変換

PI 制御 IP により算出された EMU3nVD、EMU3nVQ レジスタ値と EMU3nTHTE レジスタに格納された電気角を用いて、三相電圧変換を行い、U 相電圧と W 相電圧を求めます。また、モータ電流を計測してから PWM 波形出力までの遅延を考慮して、電気角を d 軸基準電圧レジスタ (EMU3nPHI) や電気角調整用レジスタ (EMU3nGTHT) で調整することができます。本アプリケーションノートでは、これらの調整は行いません。

【関連レジスタの設定値】

- ・ EMU3nPWMCTR レジスタの FLININIP ビット = “1” (電気角は EMU 演算結果を使用)
- ・ EMU3nPWMCTR レジスタの SETVEL ビット = “1” (RDC 速度を使用)
- ・ EMU3nPWMCTR レジスタの SETDEC ビット = “1” (非干渉制御選択時)
- ・ EMU3nSR23 レジスタ = “0000 D106H” (三相電圧変換係数)
- ・ EMU3nPHI レジスタ = “0000H” (電気角の位相を調整しない)
- ・ EMU3nGTHT レジスタ = “0000 0000H” (電気角の位相を調整しない)

【PWM IP による演算】

$$\begin{pmatrix} v_{uo} \\ v_{wo} \end{pmatrix} = EMU3nSR23 \times \begin{pmatrix} \cos(\theta + 0^\circ) & -\sin(\theta + 0^\circ) \\ \cos(\theta + 120^\circ) & -\sin(\theta + 120^\circ) \end{pmatrix} \begin{pmatrix} v_{di} \\ v_{qi} \end{pmatrix}$$

$$v_{vo} = -(v_{uo} + n_{wo})$$

(3) デューティ比の算出

三相電圧変換で生成された U 相と W 相電圧値と入力電圧レジスタ (EMU3nVOLV)、桁合わせレジスタ 1 (EMU3nPWMK1) の値から、三相 PWM 波形のデューティ比を求めます。算出したデューティ比にオフセットを加算し、Max/Min 処理した結果はデューティ比算出用 U/V/W 相出力電圧レジスタ (EMU3nVUFIX、EMU3nVVFIX、EMU3nVWFIX) に格納されます。本アプリケーションノートでは、オフセット加算は行いません。EMU3nPWMCTR レジスタの SETHARM ビットの設定により、vuo、vwo、vvo に各種変調波が重畳されます。本アプリケーションノートでは変調は行いません。

【関連レジスタの設定値】

- ・ EMU3nVUOFS レジスタ = “0000H” (三相デューティ比オフセット加算しない)
- ・ EMU3nVVOFS レジスタ = “0000H” (三相デューティ比オフセット加算しない)
- ・ EMU3nVWOFS レジスタ = “0000H” (三相デューティ比オフセット加算しない)
- ・ EMU3nVOLV レジスタ = “1000H” : システム電圧に応じて調整してください。
- ・ EMU3nPWMK1 レジスタ = “0080 0000H” (デューティ比のレンジ調整)
- ・ EMU3nDTUL レジスタ = “7FFF FFFFH” (デューティ比の上限値設定)
- ・ EMU3nDTLL レジスタ = “8000 0000H” (デューティ比の下限値設定)

【PWM IP による演算】

$$EMU3nVUFIX = v_{uo} \times \frac{EMU3nPWMK1}{2^{16}} \times \frac{1}{EMU3nVOLV} + EMU3nVUOFS$$

$$EMU3nVVFIX = v_{uo} \times \frac{EMU3nPWMK1}{2^{16}} \times \frac{1}{EMU3nVOLV} + EMU3nVUOFS$$

$$EMU3nVWFIX = v_{uo} \times \frac{EMU3nPWMK1}{2^{16}} \times \frac{1}{EMU3nVOLV} + EMU3nVUOFS$$

(4) コンペア値の算出

EMU3nVUFIX、EMU3nVVFIX、EMU3nVWFIX レジスタ値を用いて U/V/W 相コンペア値を算出し、結果は EMU3nPWMUIP、EMU3nPWMVIP、EMU3nPWMWIP レジスタに格納されます。演算には、キャリア周期レジスタ(EMU3nCARR)、短絡防止時間設定レジスタ(EMU3nDTT)、桁数あわせレジスタ2(EMU3nPWMK2)の値も用いておこなわれます。

【関連レジスタの設定値】

- ・ EMU3nPWMCTR レジスタの SHIPWM ビット = “0” (各設定値をシフトせずに出力)
- ・ EMU3nPWMCTR レジスタの PWMSEL ビット = “1” (キャリア周期と短絡防止時間から生成)
- ・ EMU3nPWMCTR レジスタの SETPWM ビット = “1” (PWM コンペア値は EMU 演算結果を使用)
- ・ EMU3nCARR レジスタ = “1F40H (8000)” (キャリア周期は 100 usec = 12.5nsec x 8000)
(TSG3 に出力し、キャリア周期として使用。TSG3 の設定と一致していることが望ましい)
- ・ EMU3nDTT レジスタ = “280H (640)” (短絡防止時間は 8 usec = 12.5nsec x 640)
- ・ EMU3nPWMK2 レジスタ = “01000000H” (PWM コンペア値のレンジ調整)
- ・ EMU3nPWMUL レジスタ = “FFFF H” (PWM コンペア値の上限値設定)
- ・ EMU3nPWMLL レジスタ = “0000 H” (PWM コンペア値の下限値設定)
- ・ EMU3nPWMU レジスタ = “10E0H” (U 相 PWM ソフト入力コンペア値(EMU3nCARR+EMU3nDTT)/2)
- ・ EMU3nPWMV レジスタ = “10E0H” (V 相 PWM ソフト入力コンペア値(EMU3nCARR+EMU3nDTT)/2)
- ・ EMU3nPWMW レジスタ = “10E0H” (W 相 PWM ソフト入力コンペア値(EMU3nCARR+EMU3nDTT)/2)

【PWM IP による演算】

U 相の演算式を示します。V/W 相に関しても同様です。

$$pwm_u = \left(EMU3nVUFIX \times \frac{(EMU3nCARR + EMU3nDTT)}{2} \times EMU3nPWMK2 \right) \times \frac{1}{2^{32}} + \frac{(EMU3nCARR + EMU3nDTT)}{2}$$

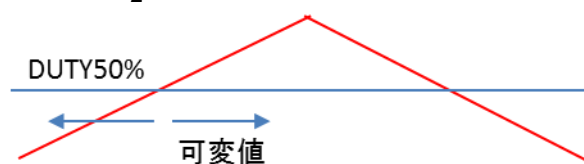
$$EMU3nPWMUIP = (EMU3nCARR + EMU3nDTT) / 2 \quad (EMU3nVOLV = 0)$$

$$EMU3nPWMUIP = EMU3nCARR + EMU3nDTT \quad (pwm_u \geq (EMU3nCARR + EMU3nDTT - EMU3nPWMUL))$$

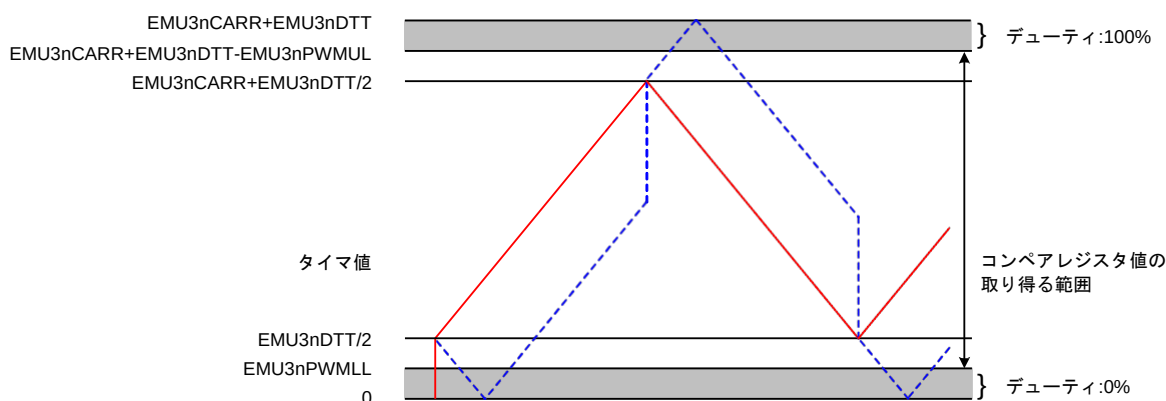
$$EMU3nPWMUIP = 0 \quad (EMU3nVOLV = 0)$$

$$EMU3nPWMUIP = pwm_u \quad (EMU3nVOLV = 0)$$

$$\text{DUTY50\%} = \frac{\text{EMU3nCARR} + \text{EMU3nDTT}}{2}$$



$$\text{可変値} = \left(\text{EMU3nVUFIX} \times \frac{\text{EMU3nCARR} + \text{EMU3nDTT}}{2} \times \text{EMU3nPWMK2} \right) \times \frac{1}{2^{16}}$$



EMU3nCARR: キャリア半周期レジスタ
 EMU3nDTT: 短絡防止時間設定レジスタ
 EMU3nPWMUL: PWM 下限値レジスタ
 EMU3nPWMUL: PWM 上限値レジスタ

図 3.9 PWM 値とコンペア値のリミット処理

3.4.6 ファイル構成

表 3.5 にサンプルコードで使用するファイルを示します。なお、統合開発環境で自動生成されるファイルは除きます。本サンプルプログラムはルネサス評価基板での動作を確認しております。

動作確認のための処理が含まれています。

表 3.5 ファイル構成

ファイル名	概要	備考
main.c	PE0 メインモジュール	PE1:main1.c, PE2:main2.c, PE1,PE2 メインは無限ループ
rdc3al.c	R/D コンバータ 3AL 設定モジュール	
pic.c	PIC 設定モジュール	
adck.c	A/D コンバータ設定モジュール	
tsg3.c	TSG3 設定モジュール	
emu3s.c	EMU3S 設定モジュール	
port.c	端子設定モジュール	
ostm.c	OSTM 設定モジュール	
user_int.c	割り込み処理を記述する	

3.4.7 関数一覧

表 3.6 に関数を示します。

表 3.6 関数

関数名	概要
rdc3al_init	R/D コンバータ 3AL の初期化
pic_init	PIC の初期化
adck_init	A/D コンバータの初期化
tsg3_init	TSG3 の初期化
emu3s_init	EMU3S の初期化
port_init	端子の初期化
ostm_init	OSTM の初期化
int_ostm_100us	電流指令値書き込み処理

3.4.8 関数仕様

サンプルコードの関数仕様を示します。

rdc3al_init	
概 要	R/D コンバータ 3AL の初期化をおこないます。
ヘッダ	iodef.h
宣 言	void rdc3al_init(void)
説 明	R/D コンバータ 3AL の初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	
pic_init	
概 要	PIC の初期化をおこないます。
ヘッダ	iodef.h
宣 言	void pic_init(void)
説 明	PIC の初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	
adck_init	
概 要	A/D コンバータの初期化をおこないます。
ヘッダ	iodef.h
宣 言	void adck_init(void)
説 明	A/D コンバータの初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	
tsg3_init	
概 要	TSG3 の初期化をおこないます。
ヘッダ	iodef.h
宣 言	void tsg3_init(void)
説 明	TSG3 の初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	

emu3s_init	
概 要	EMU3S の初期化をおこないます。
ヘッダ	iodef.h
宣 言	void emu3s_init(void)
説 明	EMU3S の初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	
port_init	
概 要	端子の初期化をおこないます。
ヘッダ	iodef.h
宣 言	void port_init(void)
説 明	端子の初期設定をおこないます。
引 数	なし
リターン値	なし
備 考	
ostm_init	
概 要	OSTM の初期化を行います。
ヘッダ	iodef.h
宣 言	void ostm_init (void)
説 明	ウェイト処理をおこないます。
引 数	なし
リターン値	なし
備 考	
int ostm_100us	
概 要	ADCK0 VR3 から電圧値を読み込み、EMU30 の Q 電流指令レジスタに電流指令値を書き込みます。
ヘッダ	iodef.h
宣 言	void int_ostm_100us (void)
説 明	電流指令値を書き込みます。
引 数	なし
リターン値	なし
備 考	Q 設定 (AD 読み込み値 × 2) - 4096 D 設定 0

3.4.9 フローチャート

図 3.11 にメイン処理、図 3.12 に RDC3AL の初期設定、図 3.13 に PIC の初期設定、図 3.14 に ADCK の初期設定、図 3.15 に TSG3 の初期設定、図 3.16 に EMU3S の初期設定(1)、図 3.17 に EMU3S の初期設定(2)、図 3.18 に端子の設定、図 3.19 に OSTM 初期設定、図 3.20 に 100us 割り込みでの電流指令値書き込み処理のフローチャートを示します。

各関数には必要に応じてレジスタ同期処理が挿入されていますが、フローチャートでは省略されています。

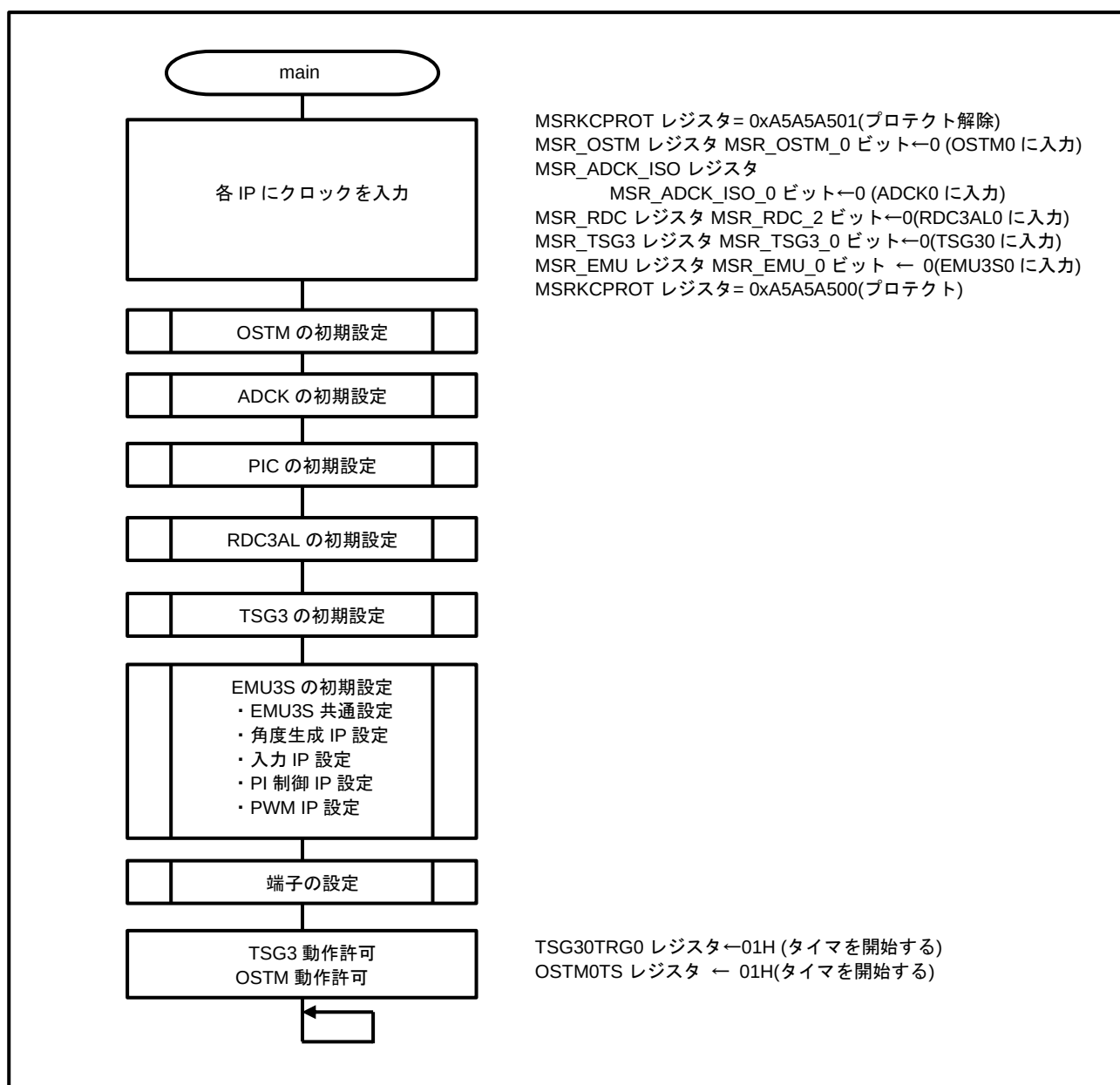


図 3.11 PE1 メイン処理

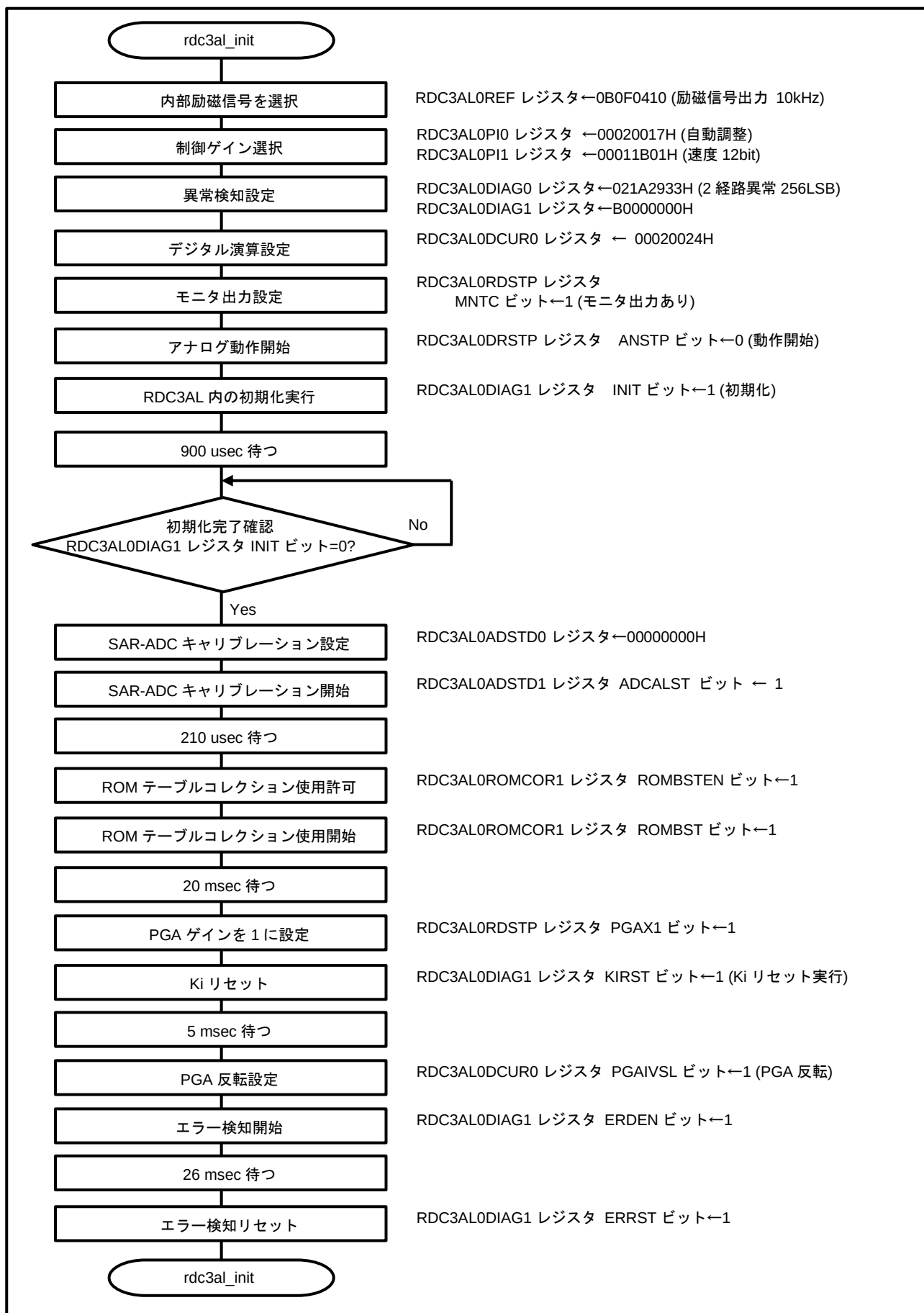


図 3.12 RDC3AL の初期設定

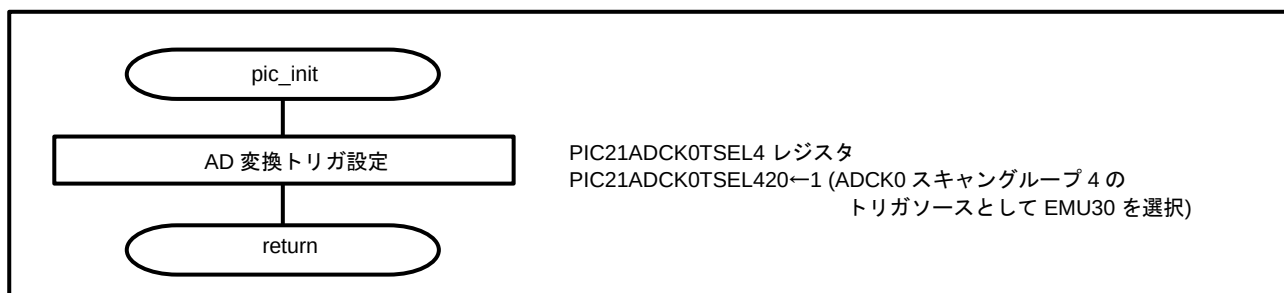


図 3.13 PIC の初期設定

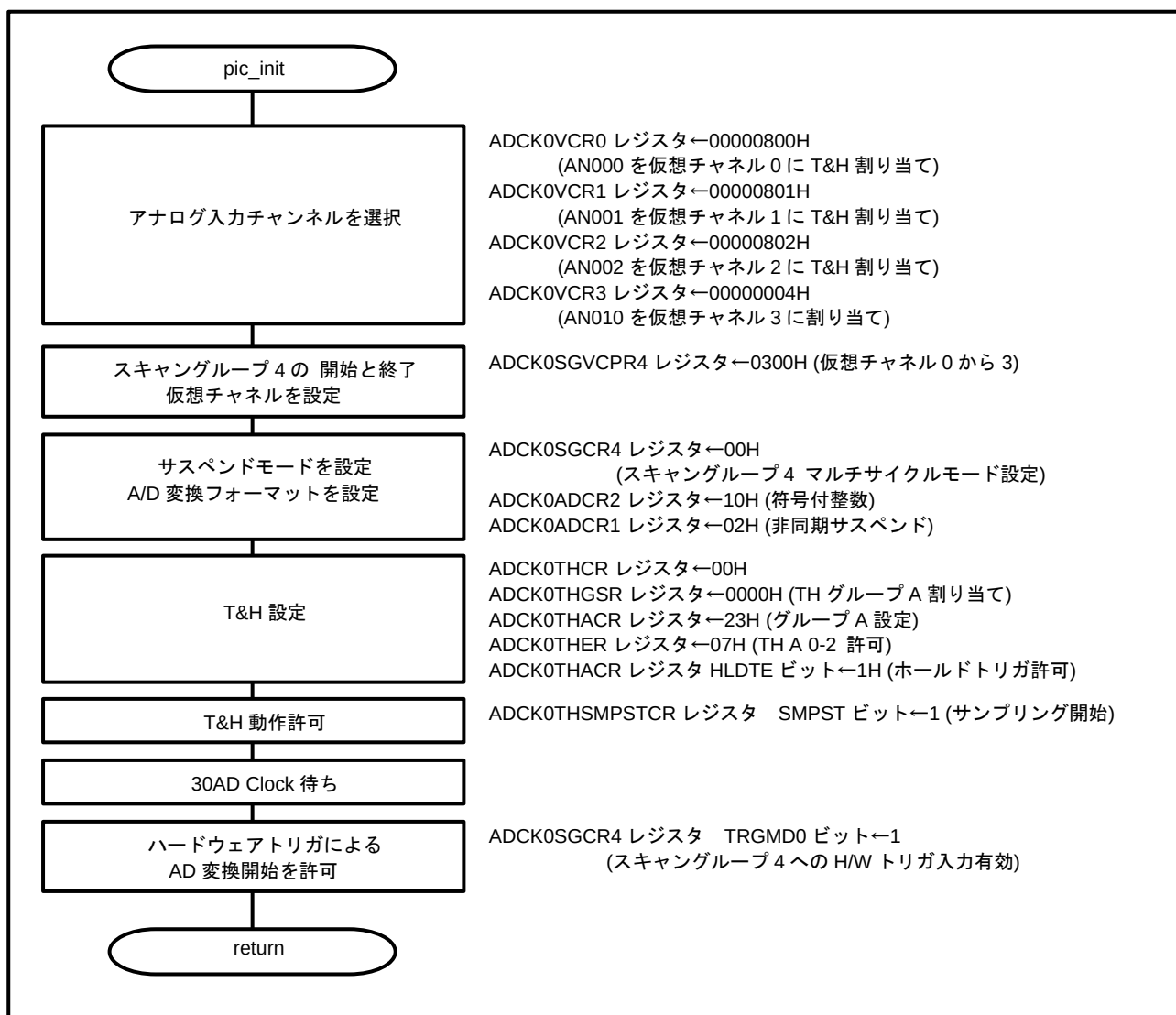


図 3.14 ADCK の初期設定

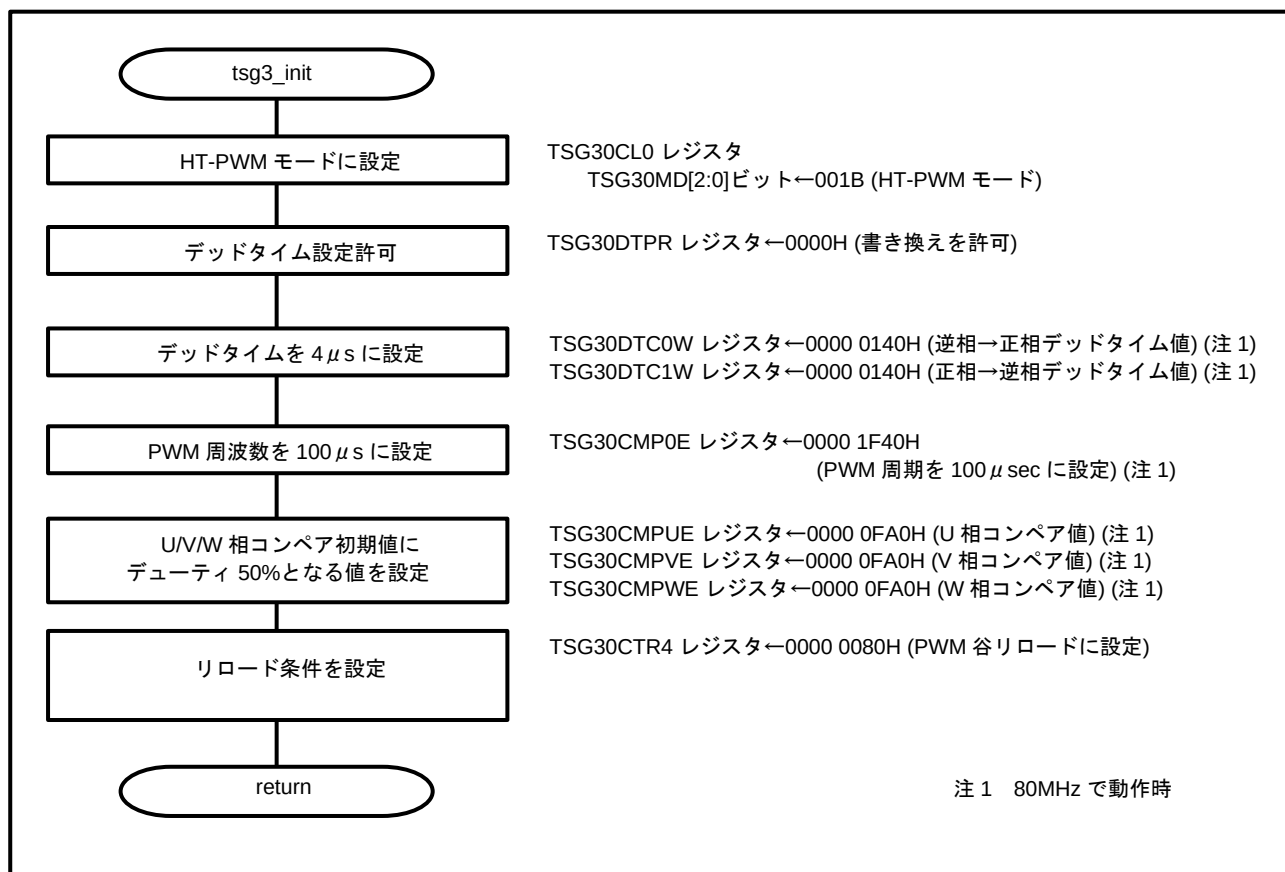


図 3.15 TSG3 の初期設定

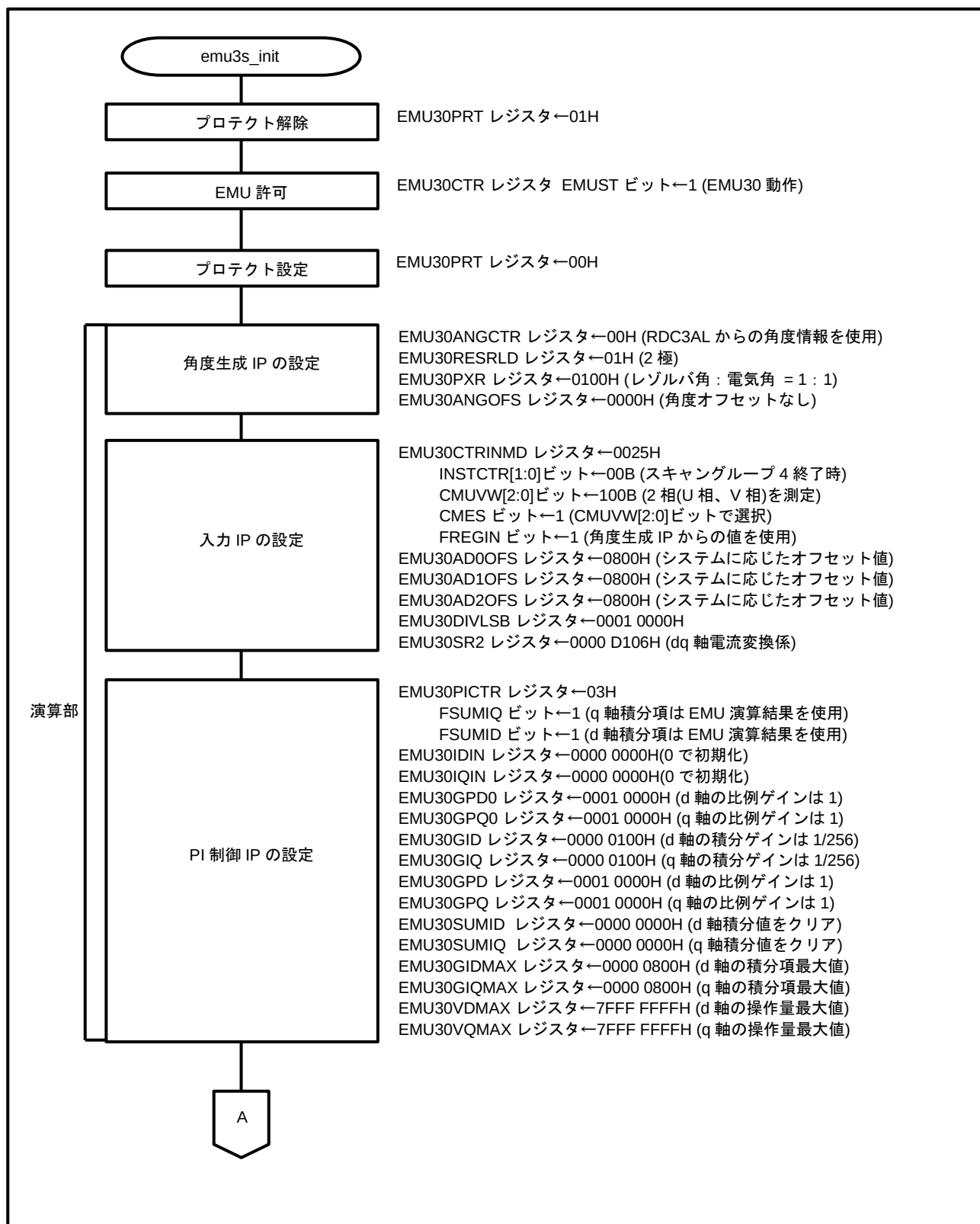


図 3.16 EMU3S の初期設定 (1)

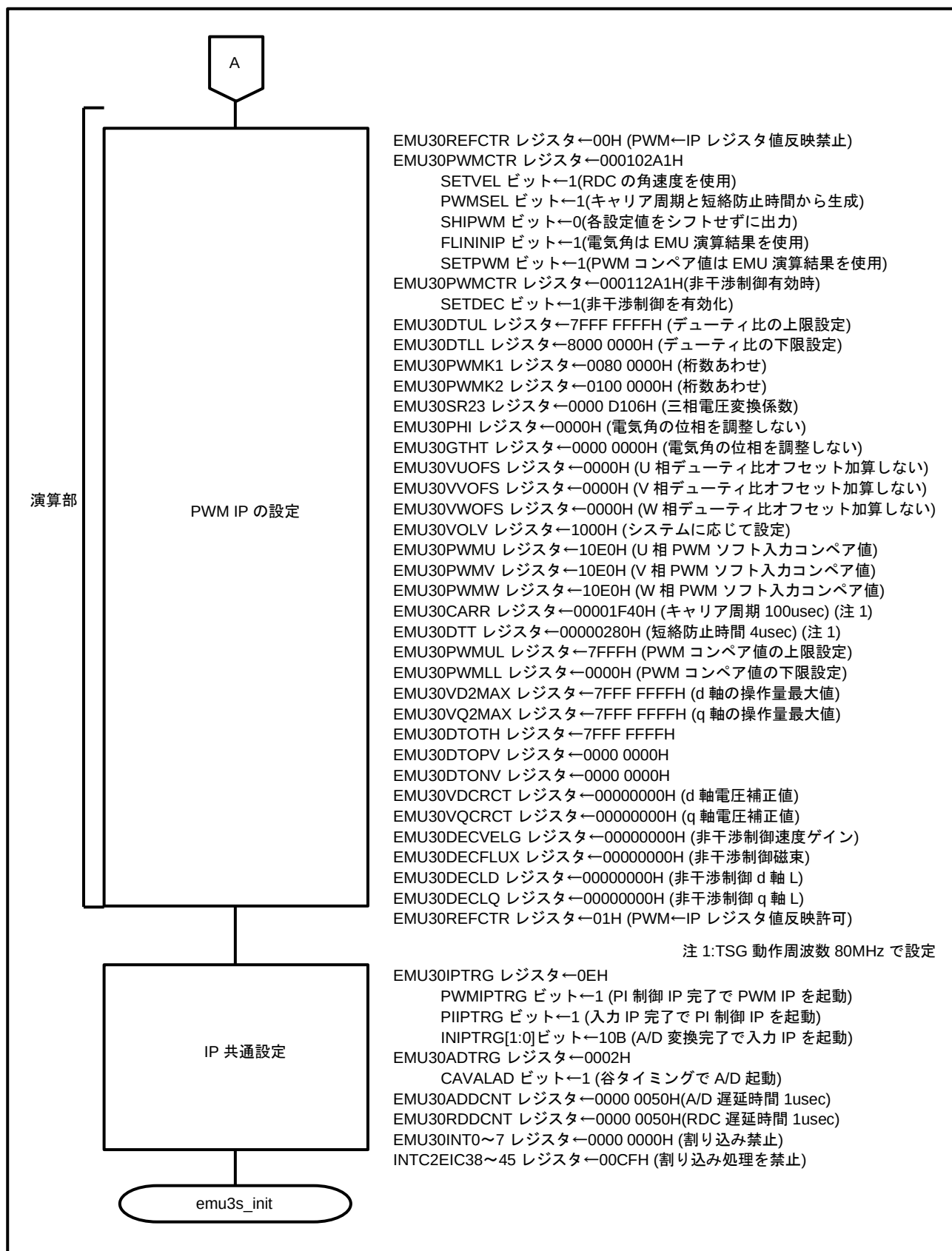


図 3.17 EMU3S の初期設定 (2)

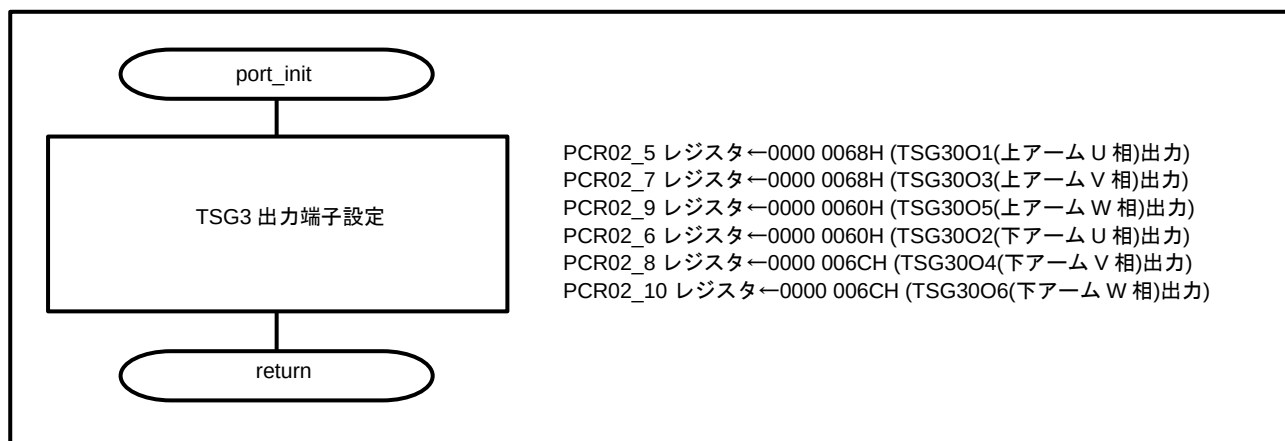


図 3.18 端子の設定

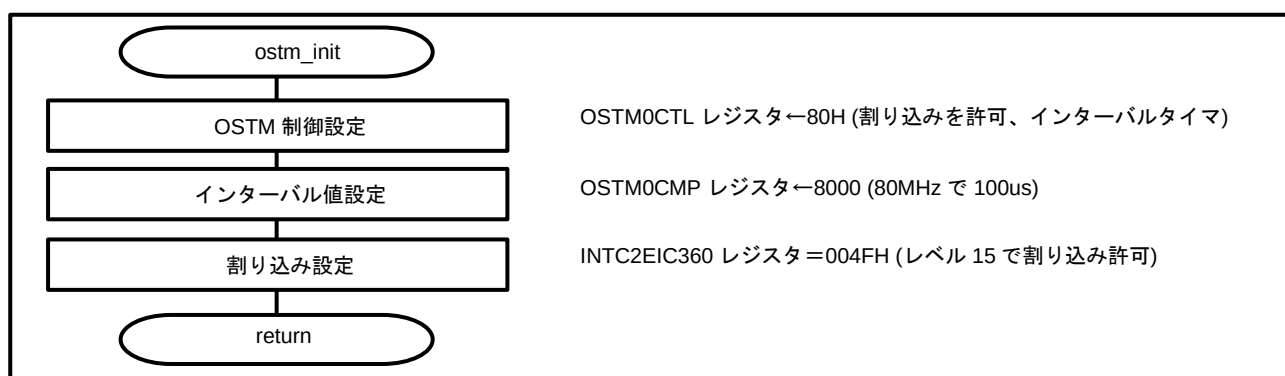


図 3.19 OSTM の初期設定

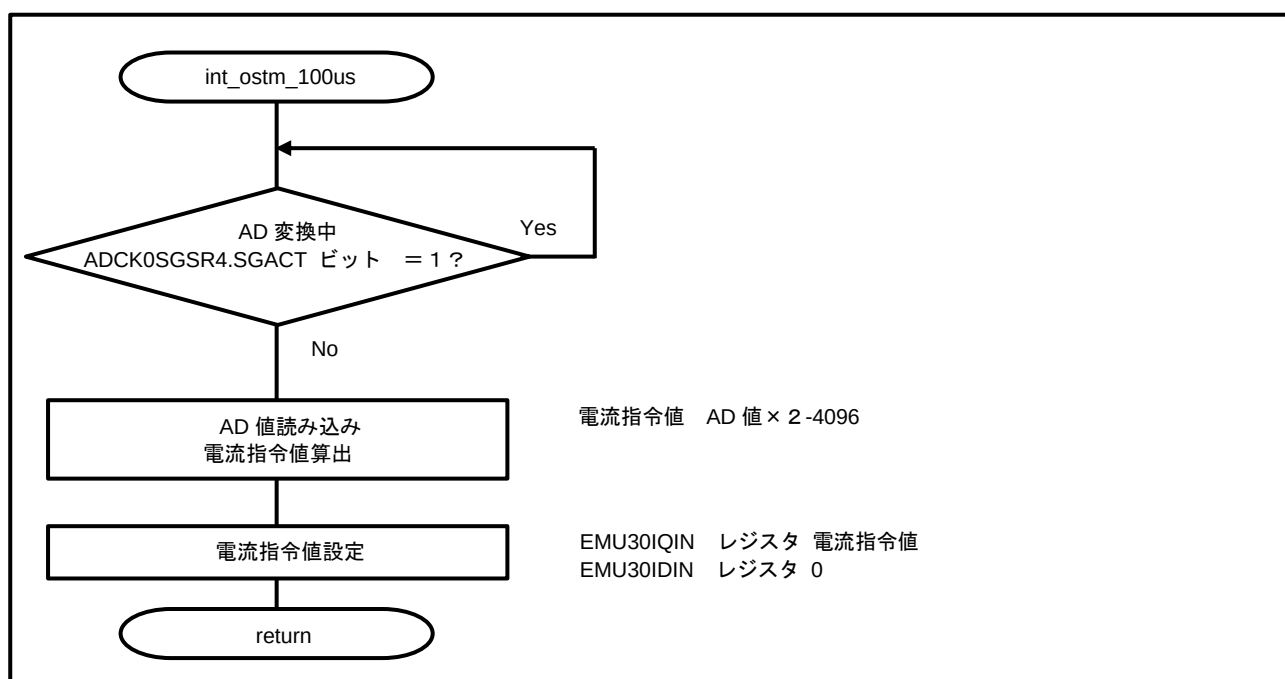


図 3.20 100us 割り込みでの電流指令値書き込み処理

4. CPU 処理のある制御動作

4.1 仕様

本章では、CPU は電流指令値の設定とフィードバック処理（割り込み）を行います。

表 4.1 にモータ制御の仕様を、表 4.2 に各モジュールの設定内容を、図 4.1 に 2 モータ対応のシステム構成図(RH850/U2B6)を、図 4.2 に CPU 処理のある制御フロー図を示します。

表 4.1 モータ制御の仕様

項目	仕様
出力波形	相補三相 PWM 波形
キャリア周期	10 kHz (100 usec/サイクル)
制御方式	180 度通電駆動方式
アクティブレベル	High アクティブ
短絡防止時間	4 usec
コンペアレジスタ、キャリア周期の更新タイミング	キャリア波の谷
A/D 変換開始タイミング	キャリア波の谷
割り込み	使用する (WAIT 遷移割り込み、100us 割り込み)
モータ電流	ADCK0 で U 相、V 相の電流値を取得
モータの角度情報	RDC3AL でレゾルバからの角度情報を取得

表 4.2 各モジュールの設定内容

周辺機能	設定内容
RDC3AL	表 3.2 を参照してください。
PIC	表 3.2 を参照してください。
ADCK	表 3.2 を参照してください。
TSG3	表 3.2 を参照してください。
EMU3S	・入力 IP は AD 完了トリガで自動的に起動、PWM IP は CPU からのソフトウェアトリガにより起動 ・PWMIP 演算には CPU 処理での演算結果を使用 ・ADCK からの電流値と RDC3AL からの角度値、角速度値を演算に使用 ・キャリア波の谷タイミングで A/D 変換起動 ・A/D 変換と R/D 変換のトリガ出力遅延 1usec

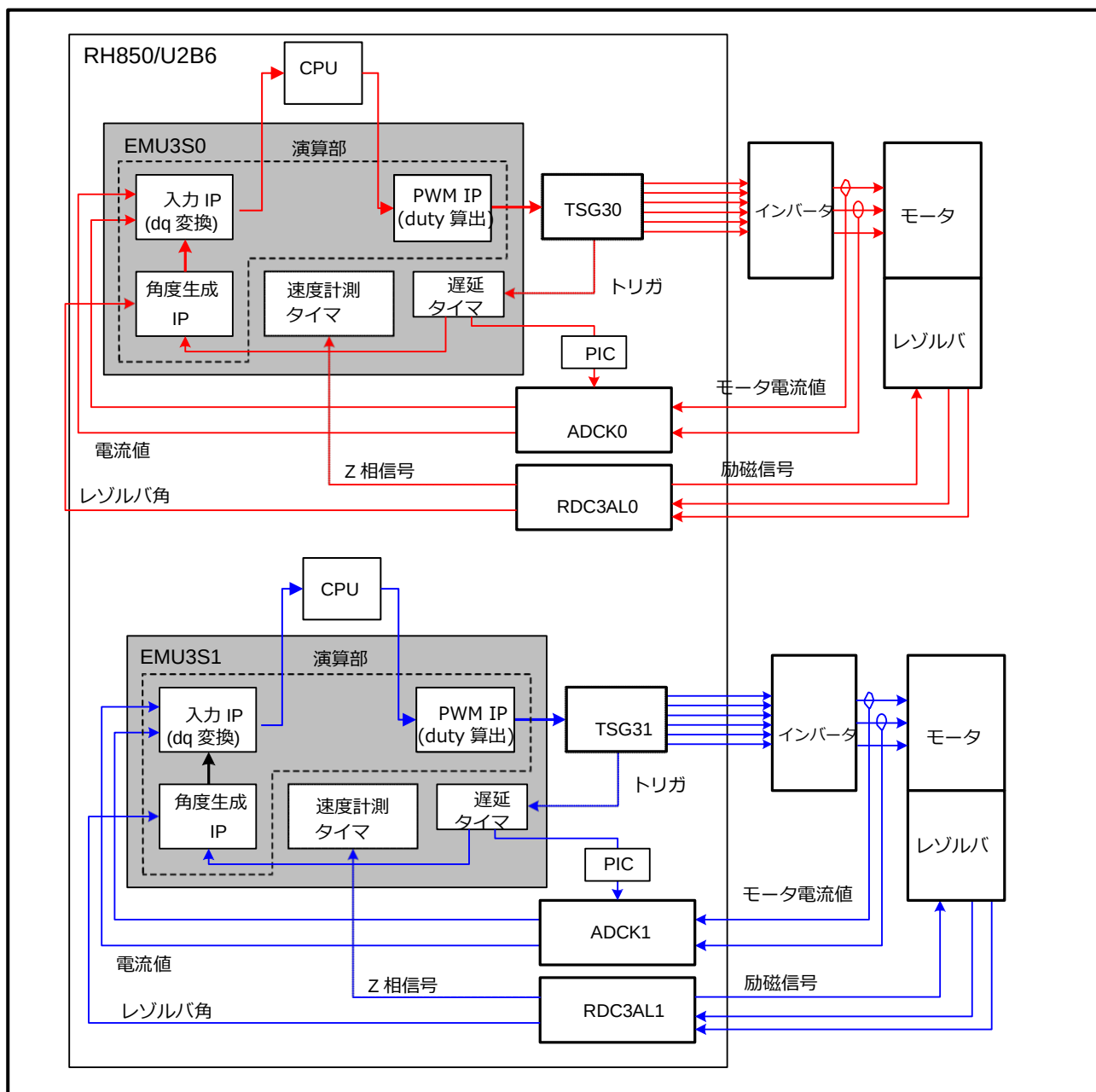


図 4.1 2 モータ対応のシステム構成図 (RH850/U2B6)

4.2 動作条件

「 3.2 動作条件 」を参照してください。

4.3 端子

「 3.3 端子 」を参照してください。

4.4 動作概要

モータ制御の動作を説明します。

- (1) キャリア波の“谷”タイミングで EMU3S から PIC 経由で A/D コンバータにトリガ信号が入力されると、ADCK0 グループ 4 の仮想チャネル 0~3 を A/D 変換しモータの U 相と V 相の電流値を取得します。同時に、EMU3S 演算部の角度生成 IP が RDC3AL から取得したレゾルバ角をモータの電気角に変換し、演算部の入力 IP に転送します。
- (2) A/D 変換が完了すると、変換結果が EMU3S のレジスタに格納され、入力 IP が自動的に起動します。入力 IP は、モータ電流値と電気角をもとに dq 変換などの演算を行い、d 軸電流と q 軸電流の帰還値を算出します。
- (3) 入力 IP の演算が完了すると、割り込みを発生させ、EMU3S は WAIT 状態になります。CPU 処理でフィードバック演算を行います。その結果を PWM IP のレジスタに反映させます。反映が終了した段階で PWM IP をソフトウェアから起動します。
- (4) PWMIP は、d 軸電圧と q 軸電圧をもとに U/V/W 相の PWM コンペア値(デューティ値)を算出します。
- (5) PWM コンペア値は、TSG3 に転送され、次のリロードタイミングで TSG3 のコンペアレジスタに設定されます。そして対応する TSG3 端子から三相 PWM 波形が出力されます。
- (6) OSTM の周期割り込みで EMU3S に対し電流指令値を設定します。これにより、制御するモータに回転指令があたえられます。

4.4.1 R/D コンバータ 3AL (RDC3AL)

「 3.4.1 R/D コンバータ 3AL (RDC3AL) 」を参照してください。

4.4.2 ペリフェラルインタコネクション (PIC)

「 3.4.2 ペリフェラルインタコネクション (PIC) 」を参照してください。

4.4.3 A/D コンバータ (ADCK)

「 3.4.3 A/D コンバータ (ADCK) 」を参照してください。

4.4.4 モータコントロールタイマ 3 (TSG3)

「 3.4.4 モータコントロールタイマ 3 (TSG3) 」を参照してください。

4.4.5 エンハンスドモータコントロールユニット 3S (EMU3S)

4.4.5.1 EMU3S 共通部

EMU3S はモータ電流値と角度情報をもとにベクトル制御をおこない、PWM コンペア値を生成します。本章では、フィードバック処理を PI 制御 IP ではなく CPU で行うため、入力 IP 完了後に WAIT に遷移し、CPU 割り込みを発生させ、フィードバック処理完了後、PWM IP をソフトウェアから起動します。

【関連レジスタの設定値】

- ・ EMU3nCTR レジスタの EMUST ビット = “1” (EMU3n 動作)
- ・ EMU3nIPTRG レジスタの INIPTRG[1:0] = “10B” (A/D 変換完了タイミングで入力 IP を起動)
- ・ EMU3nIPTRG レジスタの PIIPTRG = “0” (ソフトウェアから PI 制御 IP を起動)
- ・ EMU3nIPTRG レジスタの PWMIPTRG = “0” (ソフトウェアから PWMIP を起動)
- ・ EMU3nSMCTR レジスタの SMVAN = “1” (キャリア谷タイミングで A/D 起動)
- ・ EMU3nADDCNT レジスタの ADDATA = “80” (A/D 変換の遅延時間 1usec)
- ・ EMU3nRDDCNT レジスタの RDDATA = “80” (R/D 変換トリガ出力の遅延時間 1usec)
- ・ EMU3nFUNCFLGRPA0 レジスタ = “0080H”(input3 終了後 WAIT へ遷移)
- ・ EMU3nFUNCWAITGRPA レジスタ = “0301H” (PWM IP の Func(pwm1)から起動)
- ・ EMU3nINT0 レジスタ = “0080 0000H” (WAIT 遷移で割り込み)
- ・ EMU3nINT1~7 レジスタ = “0000 0000H” (割り込み禁止)
- ・ INTC2EIC38 レジスタ = “0040H” (割り込み処理を許可、割り込みレベル 0)
- ・ INTC2EIC39~45 レジスタ = “00CFH” (割り込み処理を禁止)

4.4.5.2 角度生成 IP

「 3.4.5.2 角度生成 IP 」を参照してください。

4.4.5.3 入力 IP

「 3.4.5.3 入力 IP 」を参照してください。

入力 IP 完了後 WAIT へと遷移し、割り込みを発生いたします。

4.4.5.4 PI 制御 IP

本動作例では何もいたしません。代わりに割り込み処理を行います。

4.4.5.5 PWM IP

割り込み処理完了後、ソフトウェアからのトリガによって PWM IP が起動します。PWM IP は、PI 制御 IP の代わりに割り込み処理で計算、レジスタに書き込まれた d 軸電圧と q 軸電圧を用います。その電圧値から U/V/W 相の出力電圧を求めます。続いて PWM 波形出力における各相のデューティ比とコンペアレジスタ値を算出します。

以降の処理は「 3.4.5.5 PWM IP 」と同一のため、そちらを参照してください。

4.4.6 ファイル構成

「 3.4.6 ファイル構成 」から変更がある内容について表 4.3 に示します。なお、統合開発環境で自動生成されるファイルは除きます。

表 4.3 変更ファイル構成

ファイル名	概要	備考
user_int.c	Wait 遷移による割り込み処理	

4.4.7 関数一覧

「 3.4.7 関数一覧 」から変更がある内容について表 4.4 に示します。

表 4.4 変更関数

関数名	概要
int_shift2wait	入力 IP から WAIT に遷移した時の割り込み

4.4.8 関数仕様

「 3.4.8 関数仕様 」から変更があるサンプルコードの関数仕様を示します。

int_shift2wait	
概 要	入力 IP の結果を読み込み、フィードバック処理を行い、PWM IP のレジスタに書き込み、PWM IP を起動します。
ヘッダ	iodef.h
宣 言	void int_shift2wait (void)
説 明	フィードバック処理を行います。
引 数	なし
リターン値	なし
備 考	

4.4.9 フローチャート

「 3.4.9 フローチャート 」から変更のあるフローチャートを図 4.3 から図 4.5 に示します。

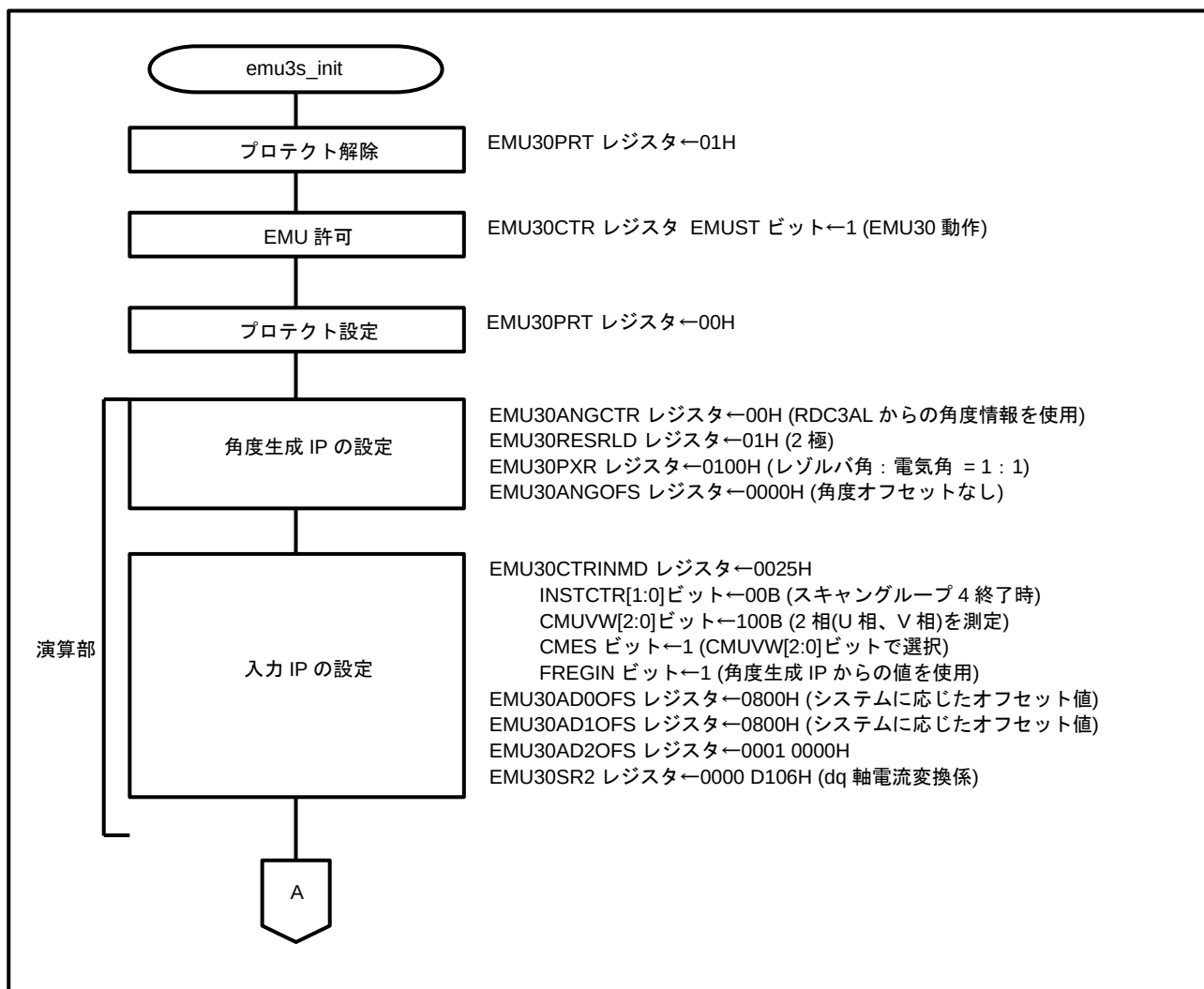


図 4.3 EMU3S の初期設定 (1)

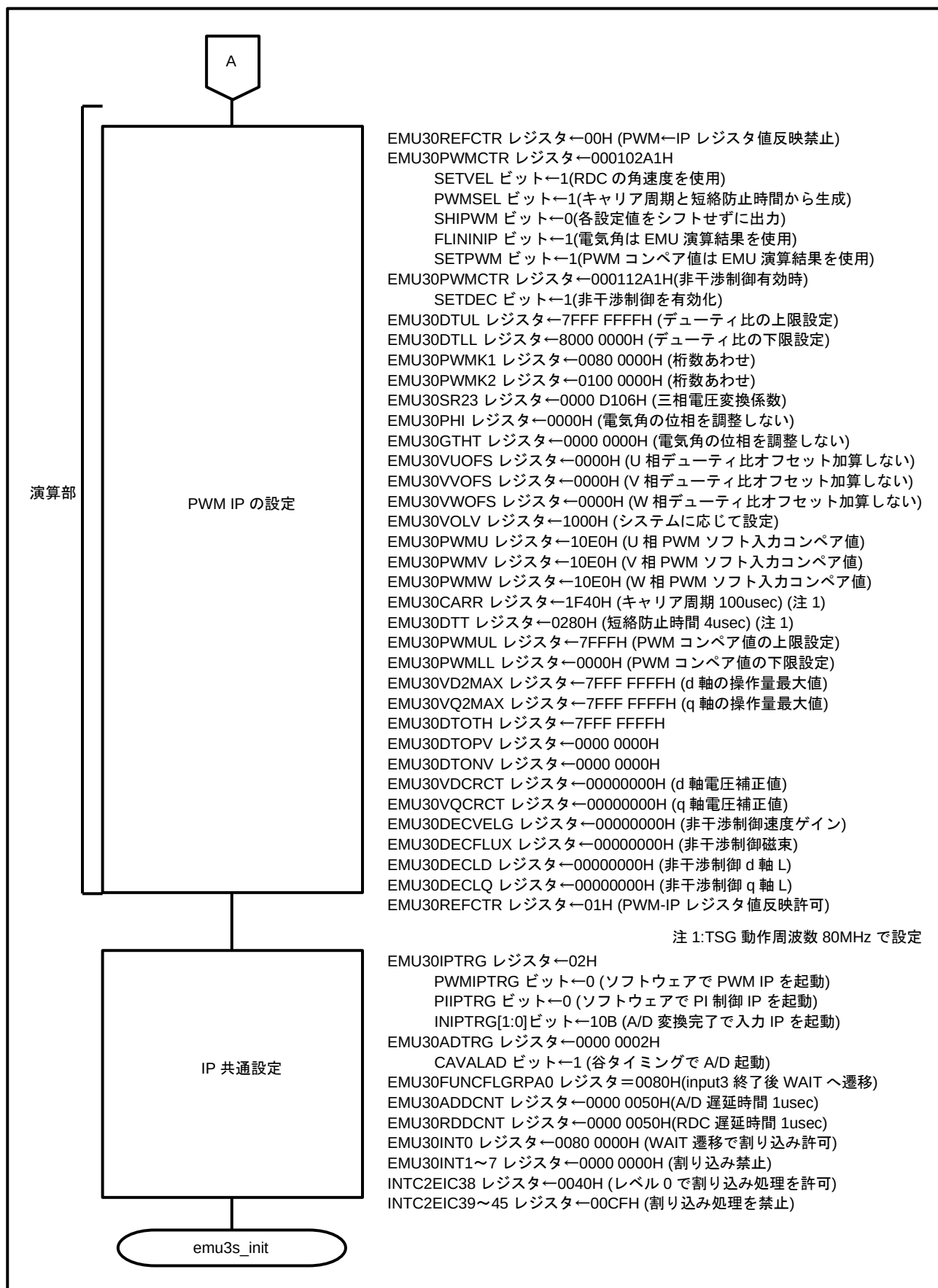


図 4.4 EMU3S の初期設定 (2)

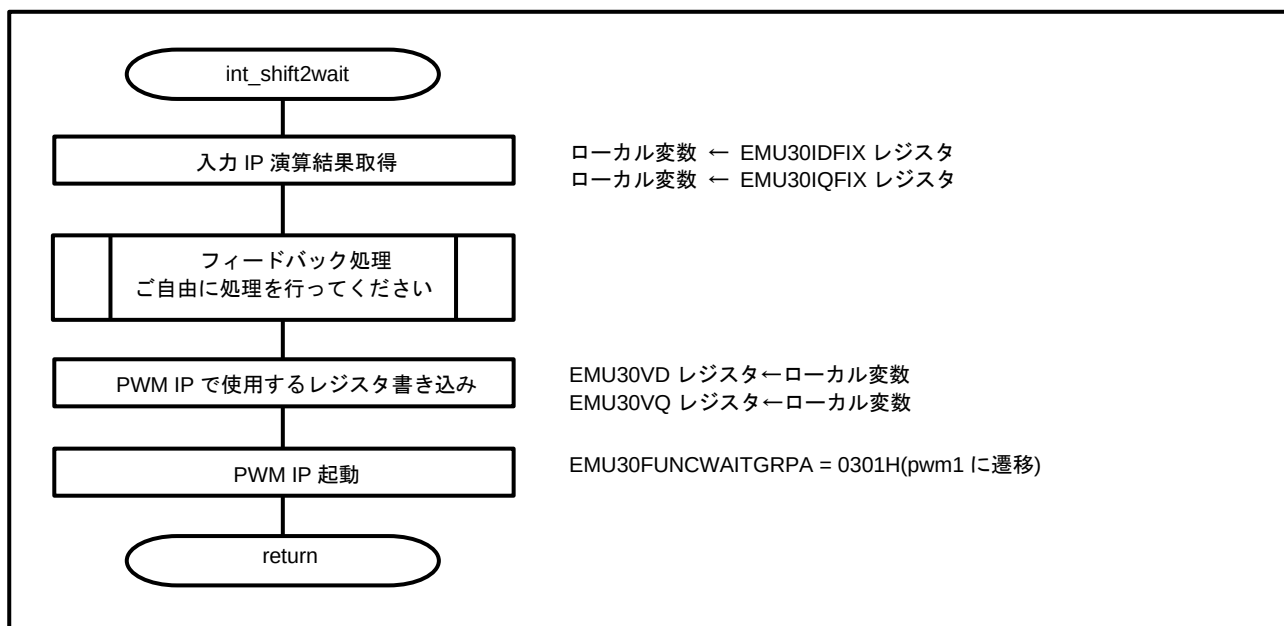


図 4.5 WAIT 遷移時の割り込み処理

5. 矩形波出力動作

5.1 仕様

本章では、角度に応じた矩形波出力処理を行います。

表 5.1 にモータ制御の仕様を、表 5.2 に各モジュールの設定内容を、図 5.1 に 2 モータ対応のシステム構成図(RH850/U2B6)を、図 5.2 に矩形波出力イメージ図を示します。

表 5.1 モータ制御の仕様

項目	仕様
出力波形	矩形波
制御方式	180 度通電駆動方式
アクティブレベル	High アクティブ
短絡防止時間	4 usec
コンペアレジスタの更新タイミング	電気角 0 度
A/D 変換開始タイミング	電気角 0 度
割り込み	使用する (U 相コンペア一致割り込み)
モータ電流	ADCK0 で U 相、V 相の電流値を取得
モータの角度情報	RDC3AL でレゾルバからの角度情報を取得

表 5.2 各モジュールの設定内容

周辺機能	設定内容
RDC3AL	表 3.2 を参照してください。
PIC	表 3.2 を参照してください。
ADCK	表 3.2 を参照してください。
TSG3	EMU3S からの矩形波を出力
EMU3S	・入力 IP は AD 完了トリガで自動的に起動、PWM IP は CPU からのソフトウェアトリガにより起動 ・PWMIP 演算には CPU 処理での演算結果を使用 ・電気角 0 度で A/D 変換起動 ・A/D 変換と R/D 変換のトリガ出力遅延 1usec

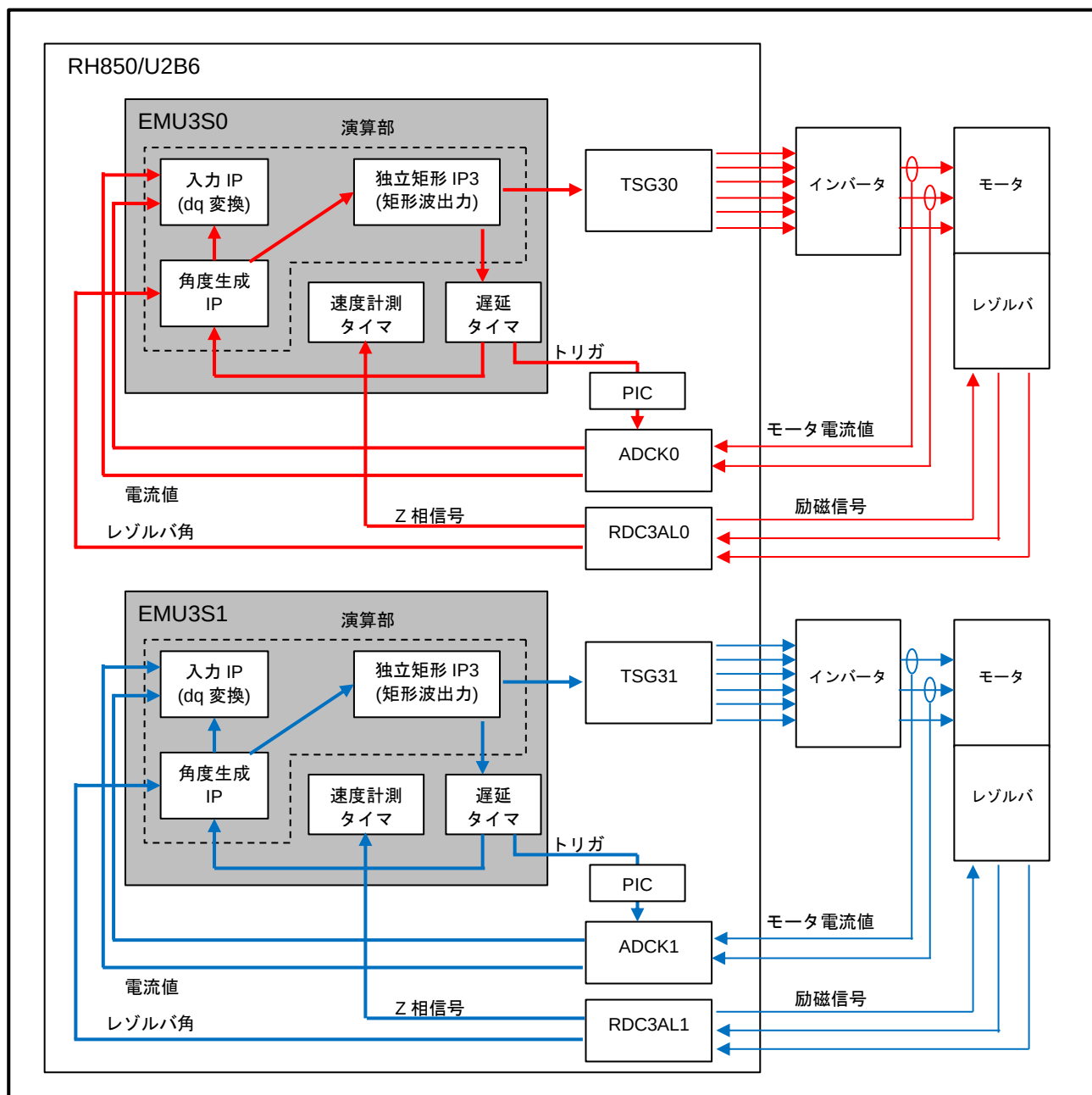


図 5.1 2 モータ対応のシステム構成図 (RH850/U2B6)

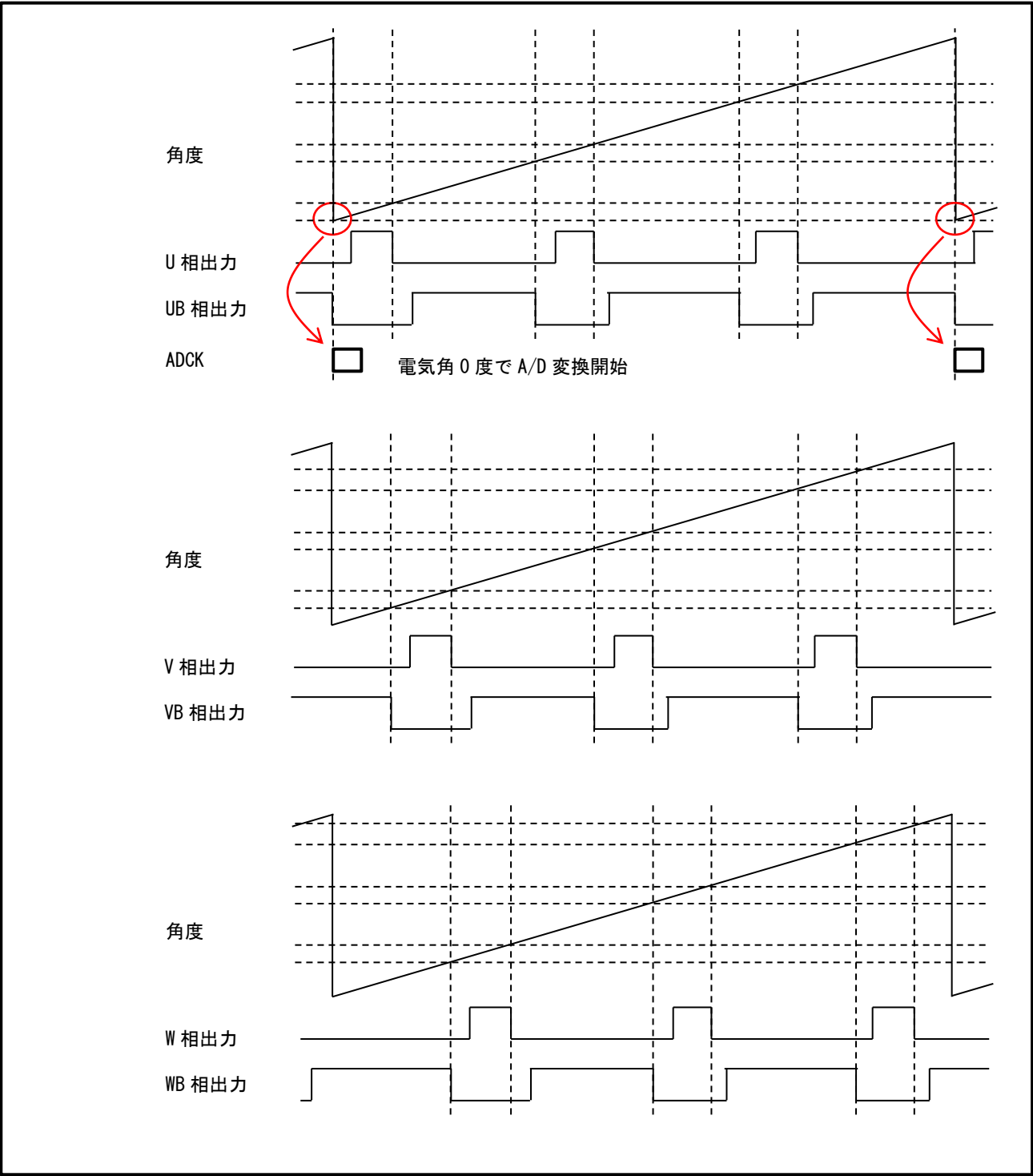


図 5.2 矩形波出力図

5.2 動作条件

「 3.2 動作条件 」を参照してください。

5.3 端子

「 3.3 端子 」を参照してください。

5.4 動作概要

モータ制御の動作を説明します。

- (1) 電気角 0 度のタイミングで EMU3S から PIC 経由で A/D コンバータにトリガ信号が入力されると、ADCK0 グループ 4 の仮想チャンネル 0~3 を A/D 変換しモータの U 相と V 相の電流値を取得します。
- (2) EMU3S 演算部の角度生成 IP が RDC3AL から取得したレゾルバ角をモータの電気角に変換し、演算部の入力 IP と独立矩形 IP3 に転送します。
- (3) A/D 変換が完了すると、変換結果が EMU3S のレジスタに格納され、入力 IP が自動的に起動します。入力 IP は、モータ電流値と電気角をもとに dq 変換などの演算を行い、d 軸電流と q 軸電流の帰還値を算出します。入力 IP 終了後 IDLE 状態に遷移します。
- (4) 独立矩形 IP3 は電気角が変更されると動作します。動作すると設定されたコンペア値と電気角の比較を行います。比較結果に従い出力を更新します。
- (5) 独立矩形 IP3 により更新された出力は TSG3 に転送されます。そして対応する TSG3 端子から三相波形が出力されます。

5.4.1 R/D コンバータ 3AL (RDC3AL)

「 3.4.13.4.1 R/D コンバータ 3AL (RDC3AL) 」を参照してください。

5.4.2 ペリフェラルインタコネクション (PIC)

「 3.4.2 ペリフェラルインタコネクション (PIC) 」を参照してください。

5.4.3 A/D コンバータ (ADCK)

「 3.4.3 A/D コンバータ (ADCK) 」を参照してください。

5.4.4 モータコントロールタイマ 3 (TSG3)

TSG3 は EMU3S から転送された矩形波出力にデッドタイムを付与して出力します。

TSG3 オプションレジスタ 2 を設定することで可能となります。

5.4.5 エンハンスドモータコントロールユニット 3S (EMU3S)

5.4.5.1 EMU3S 共通部

EMU3S は電気角を基にコンペア値と比較を行い、矩形波を生成します。本章では、電気角 0 度で割り込みと A/D 変換トリガを発生させます。入力 IP 完了後に IDLE に遷移します。

【関連レジスタの設定値】

- ・ EMU3nCTR レジスタの EMUST ビット = “1” (EMU3n 動作)
- ・ EMU3nIPTRG レジスタの INIPTRG[1:0] = “10B” (A/D 変換完了タイミングで入力 IP を起動)
- ・ EMU3nIPTRG レジスタの PIIPTRG = “0” (ソフトウェアから PI 制御 IP を起動)
- ・ EMU3nIPTRG レジスタの PWMIPTRG = “0” (ソフトウェアから PWMIP を起動)
- ・ EMU3nADDCNT レジスタの ADDATA = “80” (A/D 変換の遅延時間 1usec)
- ・ EMU3nRDDCNT レジスタの RDDATA = “80” (R/D 変換トリガ出力の遅延時間 1usec)
- ・ EMU3nFUNCFLGRPA0 レジスタ = “0040H”(input3 終了後 IDLE へ遷移)
- ・ EMU3nINT0 レジスタ = “0008 0000H” (U 相コンペア一致で割り込み)
- ・ EMU3nINT1~7 レジスタ = “0000 0000H” (割り込み禁止)
- ・ INTC2EIC38 レジスタ = “0040H” (割り込み処理を許可、割り込みレベル 0)
- ・ INTC2EIC39~45 レジスタ = “00CFH” (割り込み処理を禁止)

5.4.5.2 角度生成 IP

「 3.4.5.2 角度生成 IP 」を参照してください。

5.4.5.3 入力 IP

「 3.4.5.3 入力 IP 」を参照してください。

入力 IP 完了後 IDLE へと遷移します。

5.4.5.4 PI 制御 IP

本動作例では何もいたしません。

5.4.5.5 PWM IP

本動作例では何もいたしません。

5.4.5.6 独立矩形 IP3

角度生成 IP が算出した電気角を元にコンペア値と比較を行います。比較結果に従い出力を更新します。独立矩形 IP3 における出力とコンペア値の関係は図 5.3 のようになっています。IP3CMPn レジスタによって領域 n の終了角度と領域 n 中の出力を決定します。

このとき各領域のコンペア値は 領域 $n \leq$ 領域 $n+1$ の関係を維持してください。また、IR3VALN レジスタで使用する領域の数を指定する必要があります。図 5.3 の場合は、 $EMU30.IR3VALN.BIT.UVL=6$ （領域 0～5 の 6 領域を使用）と設定する必要があります。

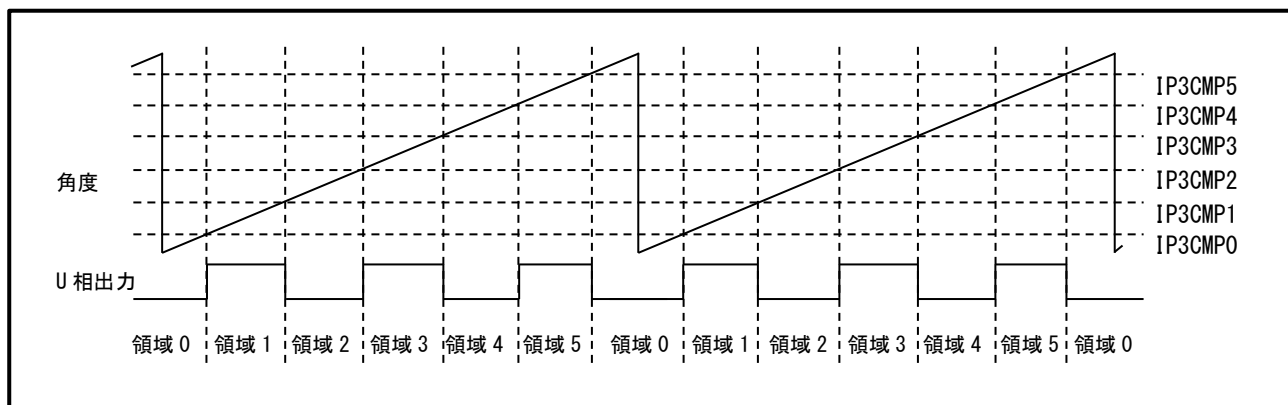


図 5.3 出力とコンペア値の関係（U 相）

また、独立矩形 IP3 におけるコンペア値は 96 個（32 個×3 相）設定できます。それに対し、Read/Write を行う IP3CMPn レジスタは 32 個存在し、どの相に対して Read/Write を行うか選択することで 96 個を Read/Write することを可能としています。

IR3CMPWRMD レジスタで Write 先を選択します。IR3CMPRDMD レジスタで Read 元を選択します。また、IR3TRSMODE レジスタ CMP ビットでバッチ転送モードを選択している場合、IR3TRG レジスタでバッチ転送のトリガを発生させ、書き込みデータを有効にしてください。

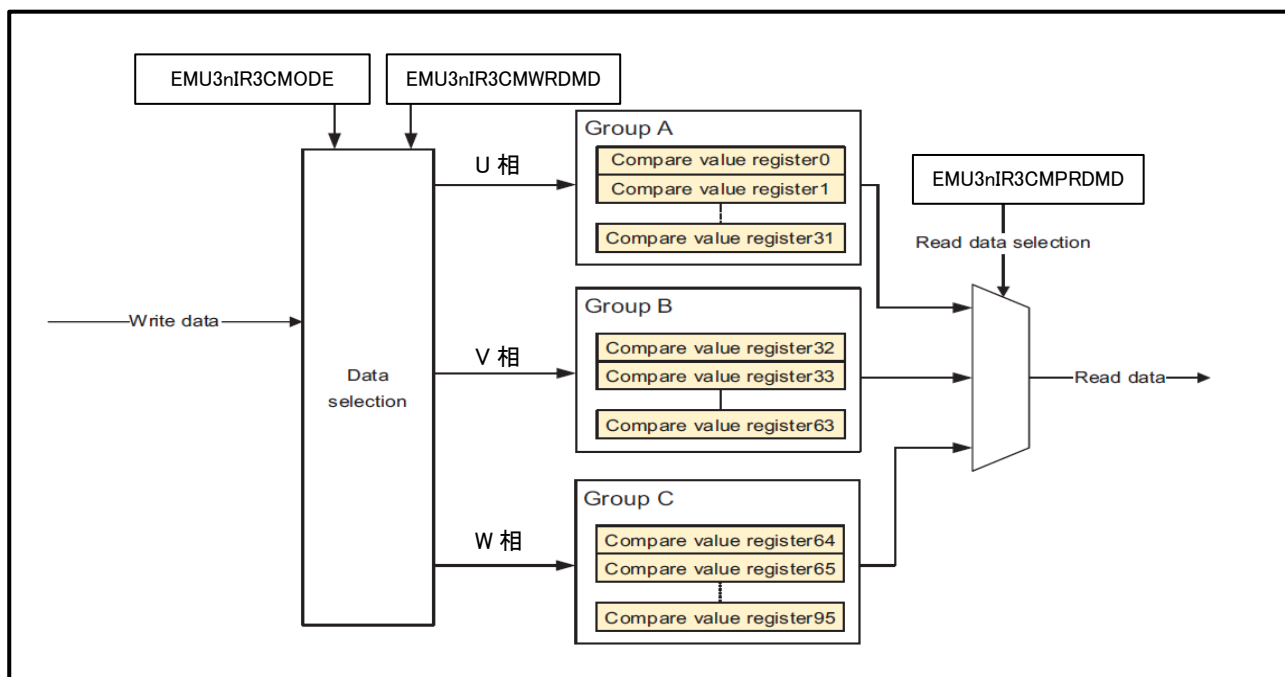


図 5.4 IP3CMPn レジスタによる Read/Write

本章では IP3CMPn レジスタを使用して初期コンペア値を表 5.3～表 5.5 のように設定いたします。

表 5.3 U 相の初期設定値

U 相	設定値	意味	U 相	設定値	意味
CMP0	0x0000	0xF40 ～ 0x000 , Low 出力	CMP16	0x0800	0x740 ～ 0x800 , Low 出力
CMP1	0x8040	0x000 ～ 0x040 , High 出力	CMP17	0x8840	0x740 ～ 0x840 , High 出力
CMP2	0x0100	0x040 ～ 0x100 , Low 出力	CMP18	0x0900	0x840 ～ 0x900 , Low 出力
CMP3	0x8140	0x100 ～ 0x140 , High 出力	CMP19	0x8940	0x900 ～ 0x940 , High 出力
CMP4	0x0200	0x140 ～ 0x200 , Low 出力	CMP20	0x0A00	0x940 ～ 0xA00 , Low 出力
CMP5	0x8240	0x200 ～ 0x240 , High 出力	CMP21	0x8A40	0xA00 ～ 0xA40 , High 出力
CMP6	0x0300	0x240 ～ 0x300 , Low 出力	CMP22	0x0B00	0xA40 ～ 0xB00 , Low 出力
CMP7	0x8340	0x300 ～ 0x340 , High 出力	CMP23	0x8B40	0xB00 ～ 0xB40 , High 出力
CMP8	0x0400	0x340 ～ 0x400 , Low 出力	CMP24	0x0C00	0xB40 ～ 0xC00 , Low 出力
CMP9	0x8440	0x400 ～ 0x440 , High 出力	CMP25	0x8C40	0xC00 ～ 0xC40 , High 出力
CMP10	0x0500	0x440 ～ 0x500 , Low 出力	CMP26	0x0D00	0xC40 ～ 0xD00 , Low 出力
CMP11	0x8540	0x500 ～ 0x540 , High 出力	CMP27	0x8D40	0xD00 ～ 0xD40 , High 出力
CMP12	0x0600	0x540 ～ 0x600 , Low 出力	CMP28	0x0E00	0xD40 ～ 0xE00 , Low 出力
CMP13	0x8640	0x600 ～ 0x640 , High 出力	CMP29	0x8E40	0xE00 ～ 0xE40 , High 出力
CMP14	0x0700	0x640 ～ 0x700 , Low 出力	CMP30	0x0F00	0xE40 ～ 0xF00 , Low 出力
CMP15	0x8740	0x700 ～ 0x740 , High 出力	CMP31	0x8F40	0xF00 ～ 0xF40 , High 出力

表 5.4 V 相の初期設定値

V 相	設定値	意味	V 相	設定値	意味
CMP0	0x0040	0xF80 ～ 0x040 , Low 出力	CMP16	0x0840	0x780 ～ 0x840 , Low 出力
CMP1	0x8080	0x040 ～ 0x080 , High 出力	CMP17	0x8880	0x840 ～ 0x880 , High 出力
CMP2	0x0140	0x080 ～ 0x140 , Low 出力	CMP18	0x0940	0x880 ～ 0x940 , Low 出力
CMP3	0x8180	0x140 ～ 0x180 , High 出力	CMP19	0x8980	0x940 ～ 0x980 , High 出力
CMP4	0x0240	0x180 ～ 0x240 , Low 出力	CMP20	0x0A40	0x980 ～ 0xA40 , Low 出力
CMP5	0x8280	0x240 ～ 0x280 , High 出力	CMP21	0x8A80	0xA40 ～ 0xA80 , High 出力
CMP6	0x0340	0x280 ～ 0x340 , Low 出力	CMP22	0x0B40	0xA80 ～ 0xB40 , Low 出力
CMP7	0x8380	0x340 ～ 0x380 , High 出力	CMP23	0x8B80	0xB40 ～ 0xB80 , High 出力
CMP8	0x0440	0x380 ～ 0x440 , Low 出力	CMP24	0x0C40	0xB80 ～ 0xC40 , Low 出力
CMP9	0x8480	0x440 ～ 0x480 , High 出力	CMP25	0x8C80	0xC40 ～ 0xC80 , High 出力
CMP10	0x0540	0x480 ～ 0x540 , Low 出力	CMP26	0x0D40	0xC80 ～ 0xD40 , Low 出力
CMP11	0x8580	0x540 ～ 0x580 , High 出力	CMP27	0x8D80	0xD40 ～ 0xD80 , High 出力
CMP12	0x0640	0x580 ～ 0x640 , Low 出力	CMP28	0x0E40	0xD80 ～ 0xE40 , Low 出力
CMP13	0x8680	0x640 ～ 0x680 , High 出力	CMP29	0x8E80	0xE40 ～ 0xE80 , High 出力
CMP14	0x0740	0x680 ～ 0x740 , Low 出力	CMP30	0x0F40	0xE80 ～ 0xF40 , Low 出力
CMP15	0x8780	0x740 ～ 0x780 , High 出力	CMP31	0x8F80	0xF40 ～ 0xF80 , High 出力

表 5.5 W 相の初期設定値

W 相	設定値	意味	W 相	設定値	意味
CMP0	0x0080	0xFC0 ~ 0x080, Low 出力	CMP16	0x0880	0x7C0 ~ 0x880, Low 出力
CMP1	0x80C0	0x080 ~ 0x0C0, High 出力	CMP17	0x88C0	0x880 ~ 0x8C0, High 出力
CMP2	0x0180	0x0C0 ~ 0x180, Low 出力	CMP18	0x0980	0x8C0 ~ 0x980, Low 出力
CMP3	0x81C0	0x180 ~ 0x1C0, High 出力	CMP19	0x89C0	0x980 ~ 0x9C0, High 出力
CMP4	0x0280	0x1C0 ~ 0x280, Low 出力	CMP20	0x0A80	0x9C0 ~ 0xA80, Low 出力
CMP5	0x82C0	0x280 ~ 0x2C0, High 出力	CMP21	0x8AC0	0xA80 ~ 0xAC0, High 出力
CMP6	0x0380	0x2C0 ~ 0x380, Low 出力	CMP22	0x0B80	0xAC0 ~ 0xB80, Low 出力
CMP7	0x83C0	0x380 ~ 0x3C0, High 出力	CMP23	0x8BC0	0xB80 ~ 0xBC0, High 出力
CMP8	0x0480	0x3C0 ~ 0x480, Low 出力	CMP24	0x0C80	0xBC0 ~ 0xC80, Low 出力
CMP9	0x84C0	0x480 ~ 0x4C0, High 出力	CMP25	0x8CC0	0xC80 ~ 0xCC0, High 出力
CMP10	0x0580	0x4C0 ~ 0x580, Low 出力	CMP26	0x0D80	0xCC0 ~ 0xD80, Low 出力
CMP11	0x85C0	0x580 ~ 0x5C0, High 出力	CMP27	0x8DC0	0xD80 ~ 0xDC0, High 出力
CMP12	0x0680	0x5C0 ~ 0x680, Low 出力	CMP28	0x0E80	0xDC0 ~ 0xE80, Low 出力
CMP13	0x86C0	0x680 ~ 0x6C0, High 出力	CMP29	0x8EC0	0xE80 ~ 0xEC0, High 出力
CMP14	0x0780	0x6C0 ~ 0x780, Low 出力	CMP30	0x0F80	0xEC0 ~ 0xF80, Low 出力
CMP15	0x87C0	0x780 ~ 0x7C0, High 出力	CMP31	0x8FC0	0xF80 ~ 0xFC0, High 出力

【関連レジスタの設定値】

- ・ EMU3nIR3CMODE レジスタ CMD ビット="1" (3 相独立モード)
- ・ EMU3nIR3CMODE レジスタ MODE ビット="00b" (のこぎり波モード)
- ・ EMU3nIR3RDI レジスタ="0x00" (正回転)
- ・ EMU3nIR3TRSMODE レジスタ="0x00" (バッチ転送モード)
- ・ EMU3nIR3INTCLEAR レジスタ="0x07" (割り込みステータスクリア)
- ・ EMU3nIR3INTEN レジスタ LOE ビット="1" (180 度未満で割り込みを有効化)
- ・ EMU3nIR3INTEN レジスタ EN ビット="00b" (U 相のみで割り込みを有効化)
- ・ EMU3nIR3INT0 レジスタ INT0 ビット="1" (U 相領域 0 のコンペア一致で割り込みを発生)
- ・ EMU3nIR3INT1 レジスタ="0x00000000" (V 相ではコンペア一致で割り込みを発生しない)
- ・ EMU3nIR3INT2 レジスタ="0x00000000" (W 相ではコンペア一致で割り込みを発生しない)
- ・ EMU3nIR3ADCCLEAR レジスタ="0x07" (A/D 変換トリガステータスをクリア)
- ・ EMU3nIR3ADCEN レジスタ LOE ビット="1" (180 度未満で A/D 変換トリガを有効化)
- ・ EMU3nIR3ADCEN レジスタ EN ビット="00b" (U 相領域 0 のコンペア一致で A/D 変換トリガを発生)
- ・ EMU3nIR3ADC0 レジスタ ADI0 ビット="1" (U 相領域 0 のコンペア一致で A/D 変換トリガを発生)
- ・ EMU3nIR3ADC1 レジスタ="0x00000000" (V 相ではコンペア一致で A/D 変換トリガを発生しない)
- ・ EMU3nIR3ADC2 レジスタ="0x00000000" (W 相ではコンペア一致で A/D 変換トリガを発生しない)

- ・ EMU3nIR3COFSALL レジスタ="0x0000" (電気角のオフセットはなし)
- ・ EMU3nIR3COFSU レジスタ="0x0000" (U 相に対する電気角のオフセットはなし)
- ・ EMU3nIR3COFSV レジスタ="0x0000" (V 相に対する電気角のオフセットはなし)
- ・ EMU3nIR3COFSW レジスタ="0x0000" (W 相に対する電気角のオフセットはなし)
- ・ EMU3nIR3VALN レジスタ UVL ビット="32" (U 相のコンペア値を 32 個使用する)
- ・ EMU3nIR3VALN レジスタ VVL ビット="32" (V 相のコンペア値を 32 個使用する)
- ・ EMU3nIR3VALN レジスタ WVL ビット="32" (W 相のコンペア値を 32 個使用する)
- ・ EMU3nIR3TRG レジスタ TRG ビット="1" (UV/W 相のコンペアレジスタを反映する)
- ・ EMU3nIR3CCTR レジスタ EN ビット="1" (独立矩形 IP3 を有効化)

5.4.6 ファイル構成

「 3.4.6 ファイル構成 」から変更がある内容について表 5.6 に示します。なお、統合開発環境で自動生成されるファイルは除きます。

表 5.6 変更ファイル構成

ファイル名	概要	備考
emu3s.c	独立矩形 IP3 の初期設定	
user_int.c	U 相コンペア一致割り込み処理	

5.4.7 関数一覧

「 3.4.7 関数一覧 」から変更がある内容について表 5.7 に示します。

表 5.7 変更関数

関数名	概要
emu3s_init	角度生成 IP と独立矩形 IP3 の初期設定
emu3s_rec_angle_init	独立矩形 IP3 のコンペア値を設定
emu3s_rec_angle_update	独立矩形 IP3 のコンペア値をレジスタに反映
emu3s_rec_angle_update_u	独立矩形 IP3 の U 相コンペア値を設定
emu3s_rec_angle_update_v	独立矩形 IP3 の V 相コンペア値を設定
emu3s_rec_angle_update_w	独立矩形 IP3 の W 相コンペア値を設定
emu3s_rec_angle_reflect	独立矩形 IP3 のコンペア値を反映
int_rec_ip3_compare	電気角 0 度で U 相コンペア一致時の割り込み

5.4.8 関数仕様

「 3.4.8 関数仕様 」から変更があるサンプルコードの関数仕様を示します。

emu3s_init	
概 要	角度生成 IP と独立矩形 IP3 の初期設定
ヘッダ	iodef.h
宣 言	void emu3s_init (void)
説 明	EMU3S（角度生成 IP と独立矩形 IP3）の初期設定を行います。
引 数	なし
リターン値	なし
備 考	
emu3s_rec_angle_init	
概 要	独立矩形 IP3 のコンペア値を設定
ヘッダ	なし
宣 言	void emu3s_rec_angle_init (void)
説 明	独立矩形 IP3 で使用するコンペア値配列に値を設定します。
引 数	なし
リターン値	なし
備 考	
emu3s_rec_angle_update	
概 要	独立矩形 IP3 のコンペア値をレジスタに反映
ヘッダ	iodef.h
宣 言	void emu3s_rec_angle_update (unsigned short* data)
説 明	独立矩形 IP3 のコンペア値を IP3CMP レジスタに反映します。
引 数	なし
リターン値	なし
備 考	
emu3s_rec_angle_update_u	
概 要	独立矩形 IP3 の U 相コンペア値を設定。
ヘッダ	iodef.h
宣 言	void emu3s_rec_angle_update_u (void)
説 明	IP3CMP レジスタに U 相データを書き込める設定に変更し、 emu3s_rec_angle_update をコールします。
引 数	なし
リターン値	なし
備 考	

emu3s_rec_angle_update_v

概 要 独立矩形 IP3 の V 相コンペア値を設定。

ヘッダ iodef.h

宣 言 void emu3s_rec_angle_update_v (void)

説 明 IP3CMP レジスタに V 相データを書き込める設定に変更し、
emu3s_rec_angle_update をコールします。

引 数 なし

リターン値 なし

備 考

emu3s_rec_angle_update_w

概 要 独立矩形 IP3 の W 相コンペア値を設定。

ヘッダ iodef.h

宣 言 void emu3s_rec_angle_update_w (void)

説 明 IP3CMP レジスタに W 相データを書き込める設定に変更し、
emu3s_rec_angle_update をコールします。

引 数 なし

リターン値 なし

備 考

emu3s_rec_angle_reflect

概 要 独立矩形 IP3 のコンペア値を反映

ヘッダ iodef.h

宣 言 void emu3s_rec_angle_reflect (void)

説 明 IP3CMP レジスタに書き込んだ値を内部の保持レジスタに反映します。

引 数 なし

リターン値 なし

備 考

int_rec_ip3_compare

概 要 電気角 0 度で U 相コンペア一致時の割り込み

ヘッダ iodef.h

宣 言 void int_rec_ip3_compare (void)

説 明 U 相での割り込みによって、独立矩形 IP3 の IP3CMP レジスタを更新します。

引 数 なし

リターン値 なし

備 考

5.4.9 フローチャート

「 3.4.9 フローチャート 」から変更のあるフローチャートを図 5.5 から図 5.12 に示します。

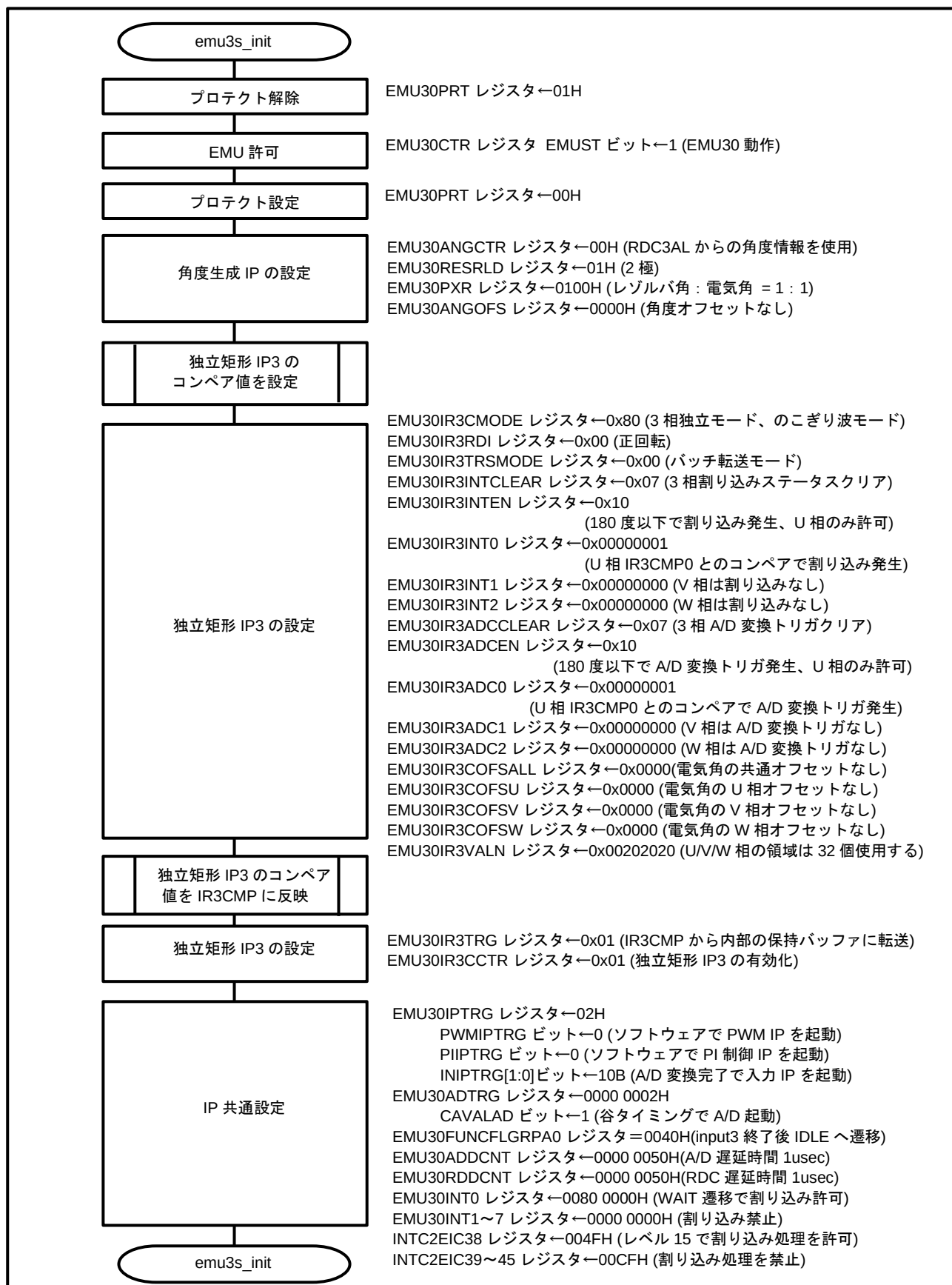


図 5.5 EMU3S の初期設定

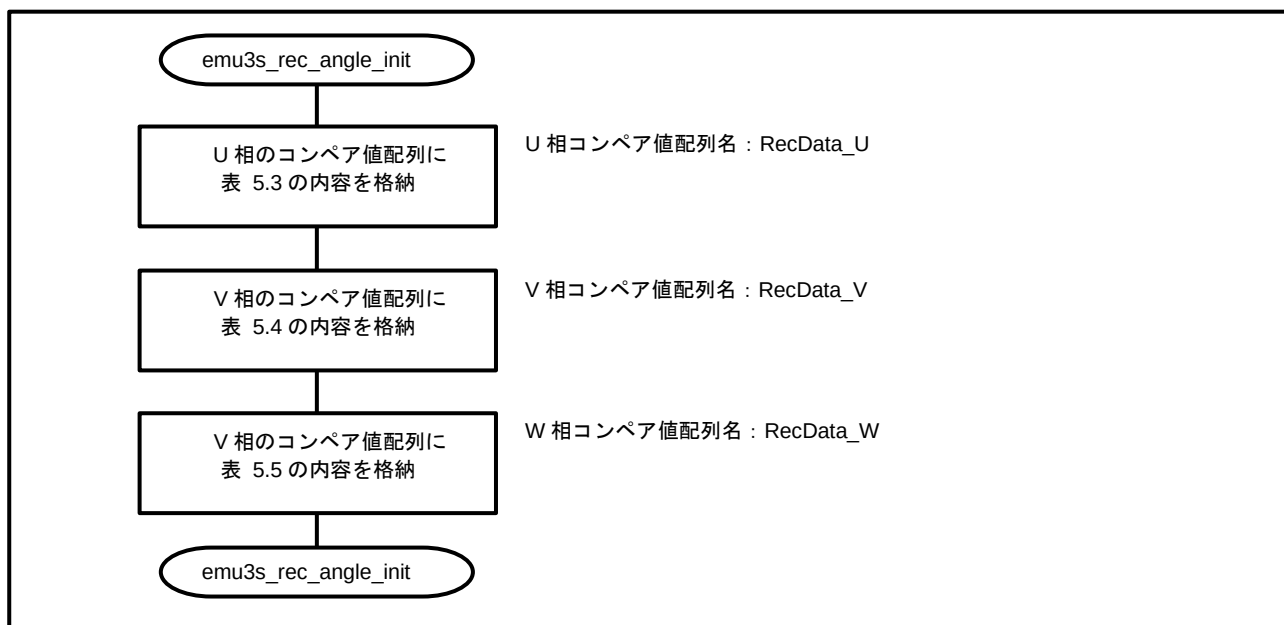


図 5.6 コンペア値配列の設定

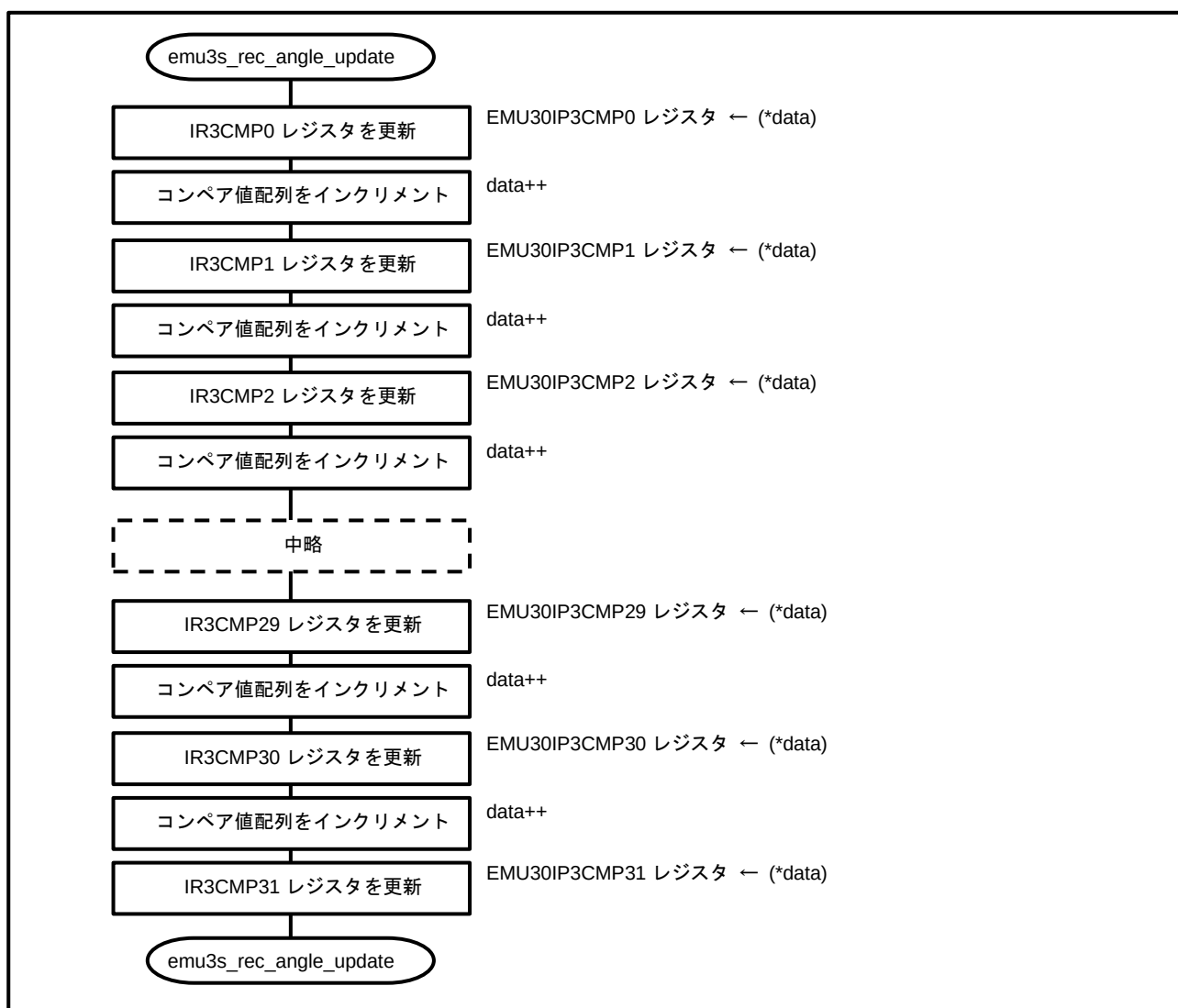


図 5.7 IR3CMP レジスタの更新

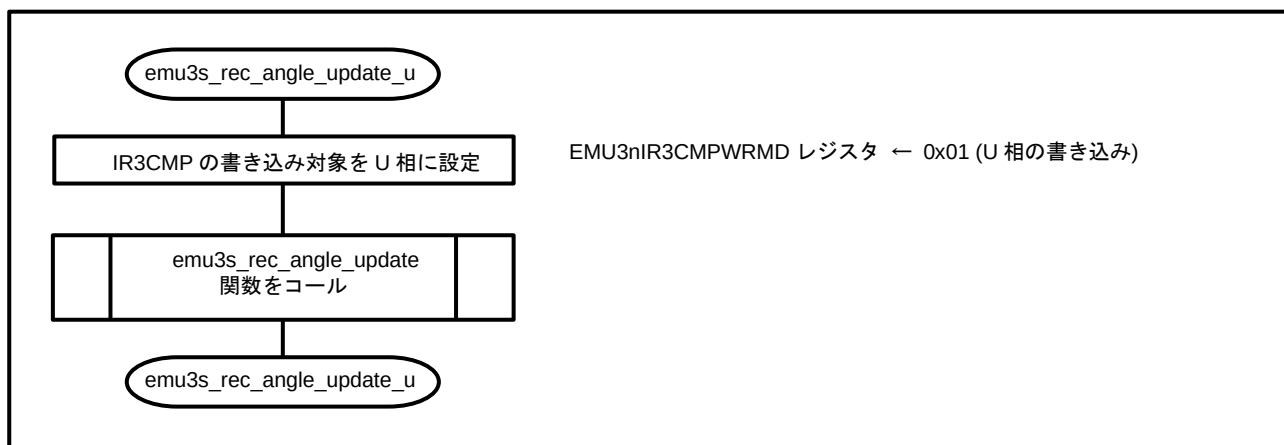


図 5.8 IR3CMP レジスタの更新 (U 相)

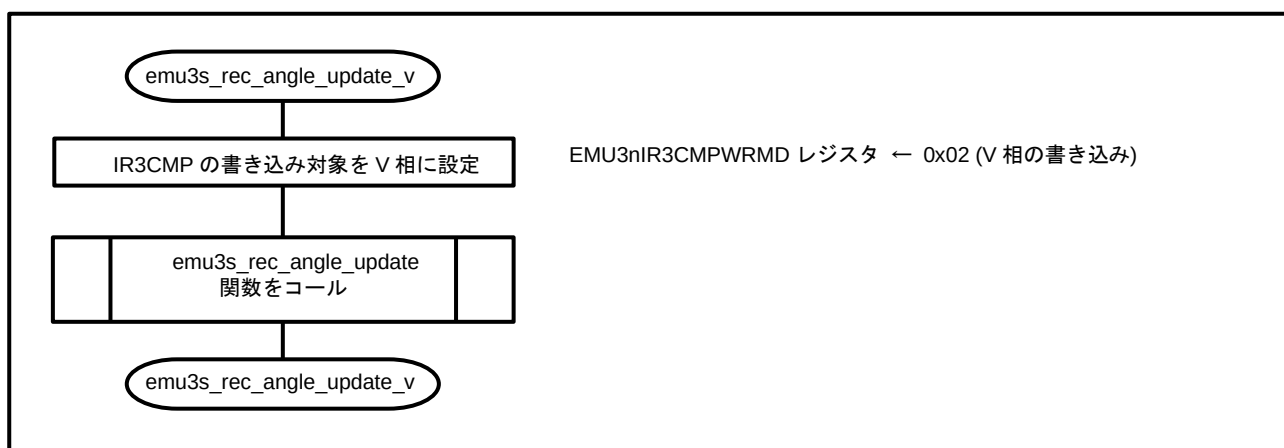


図 5.9 IR3CMP レジスタの更新 (V 相)

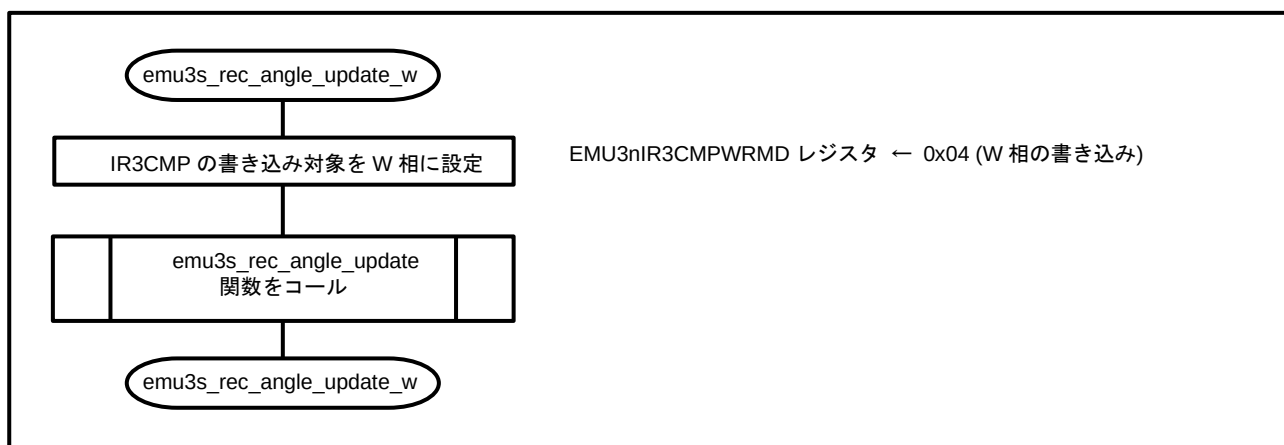


図 5.10 IR3CMP レジスタの更新 (W 相)

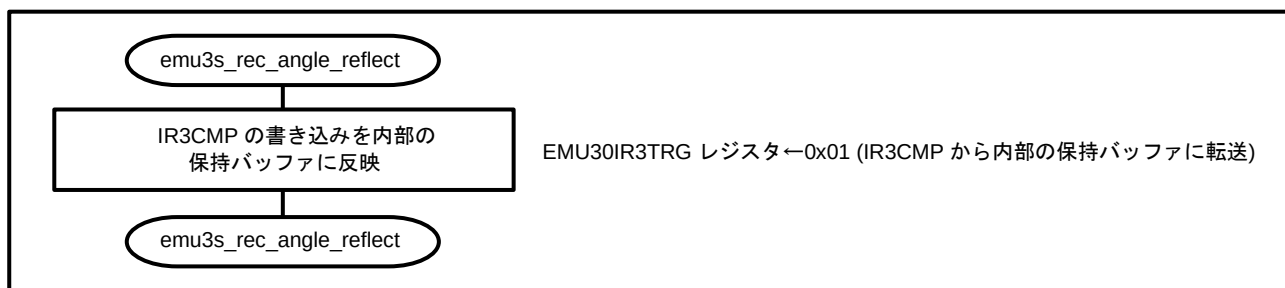


図 5.11 保持バッファへの反映

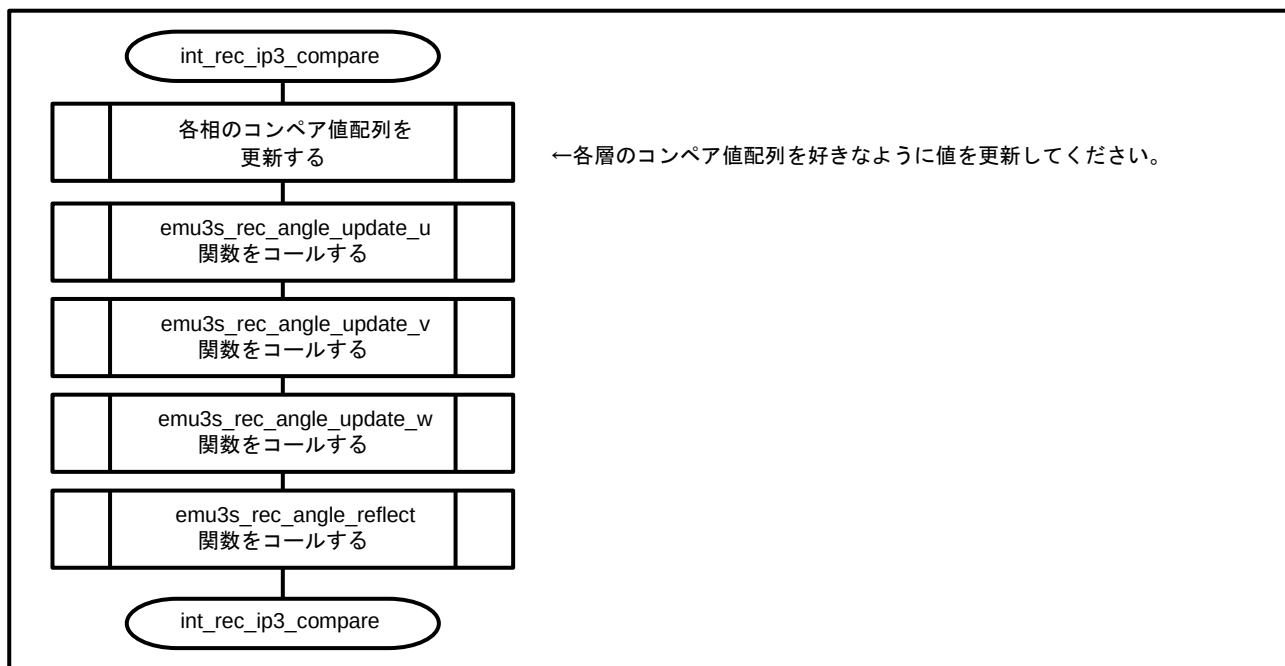


図 5.12 U 相コンペアでの割り込み処理

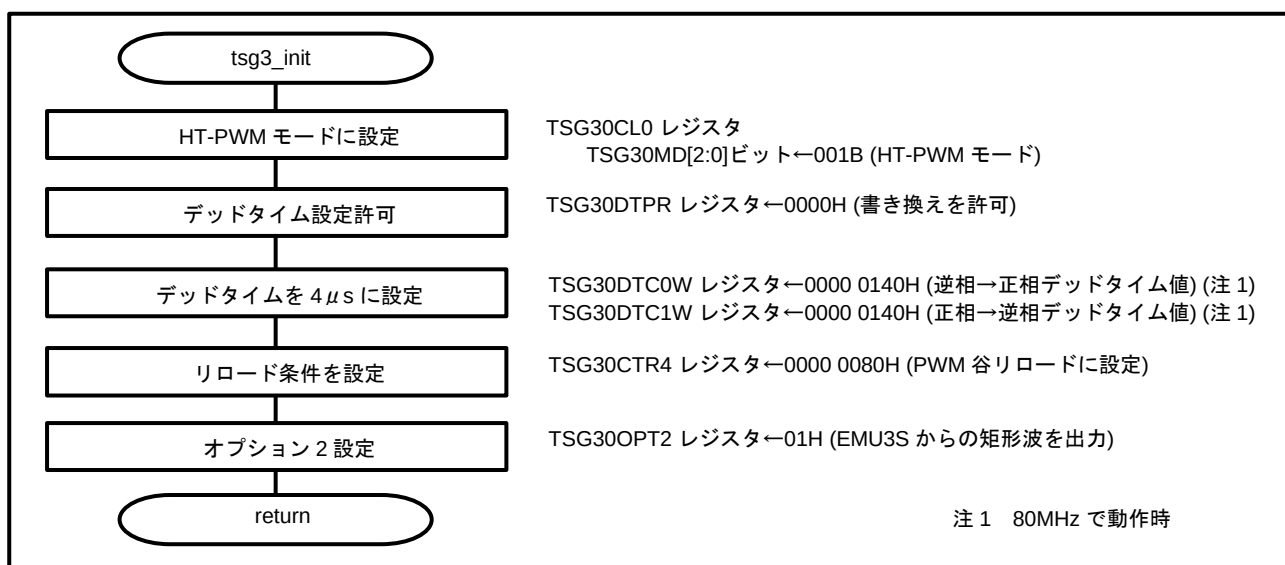


図 5.13 TSG3 の初期設定

6. 参考ドキュメント

RH850/U2B Group User's Manual: Hardware
(最新版をお問い合わせください。)

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2022.03.31	-	初版発行
1.10	2025.01.31	15, 30, 31	誤記修正

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

ご注意書き

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

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

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

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

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

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

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

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

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。