# RL78/F23, F24
# タイマ RDe と AAU を用いた DC/DC 制御

R01AN6838JJ0100
Rev.1.00
2023.07.31

## 要旨

本アプリケーションノートでは、RL78/F23, F24 のタイマ RDe、および AAU を使用した DC/DC 電源制御例について説明します。

使用条件により例に挙げた動作と実際のマイコン動作が異なる場合があります。実装後は十分な評価を実施してください。また、各機能の詳細は必ずユーザーズマニュアルハードウェア編で確認してください。

## 動作確認デバイス

- RL78/F23, F24

# 目次

1. DC/DC 制御の概要 ....4
1.1 デジタル電源制御に関する RL78/F2x の特徴 ....4
1.2 デジタル電源制御で使用する RL78/F2x の周辺機能 ....5
1.2.1 タイマ RDe ....5
1.2.2 アプリケーション・アクセラレータ・ユニット（AAU） ....8
1.3 デジタル電源制御のハードウェア構成 ....9
1.4 使用端子一覧 ....9

2. 制御ソフトウェア ....10
2.1 RL78/F2x 内蔵機能の初期化 ....10
2.1.1 内蔵クロック ....10
2.1.2 タイマ RDe ....10
2.1.3 A/D コンバータ ....11
2.1.4 DTC ....11
2.1.5 アプリケーション・アクセラレーション・ユニット(AAU) ....11
2.2 制御ソフトウェアの DC/DC 制御 ....12
2.2.1 A/D 値の取得制御 ....13
2.2.2 PI 制御 ....14
2.2.3 D 制御 ....15
2.2.4 PWM へのフィードバック ....16
2.3 DC/DC 制御フロー ....17
2.4 変数一覧 ....19
2.5 関数一覧 ....20
2.6 関数仕様 ....21
2.6.1 main ....21
2.6.2 error_proc ....21
2.6.3 r_APP_PID_Create ....22
2.6.4 r_APP_PID_UpdateParam ....22
2.6.5 R_Config_AAU_Create ....23
2.6.6 R_AAU_Start ....23
2.6.7 R_AAU_GetPIControlData ....24
2.6.8 R_Config_DTC_Create ....24
2.6.9 R_Config_DTC_Create_UserInit ....25
2.6.10 R_DTCH0_Start ....25
2.6.11 R_DTCH0_Stop ....25
2.6.12 R_Config_S12AD0_Create ....26
2.6.13 R_Config_S12AD0_Create_UserInit ....26
2.6.14 R_Config_S12AD0_Start ....27
2.6.15 R_Config_S12AD0_Stop ....27
2.6.16 r_Config_S12AD0_Interrupt ....27
2.6.17 R_Config_TRD0_Create ....28
2.6.18 R_Config_TRD0_Create_UserInit ....29
2.6.19 R_Config_TRD0_Start ....29
2.6.20 R_Config_TRD0_Stop ....29
2.6.21 R_Config_PORT_Create ....30

2.6.22 R_Config_TRD0_Set_TRD0_ReloadTrigger ..... 30
2.7 フローチャート ..... 31
2.7.1 起動シーケンス ..... 31
2.7.2 システム制御フローチャート ..... 32
2.7.3 ポート初期設定 ..... 33
2.7.4 タイマ初期設定 ..... 34
2.7.5 A/D コンバータ初期設定 ..... 35
2.7.6 DTC 初期設定 ..... 36
2.7.7 AAU 初期設定 ..... 37
2.7.8 タイマ制御 ..... 38
2.7.9 A/D コンバータ制御 ..... 39
2.7.10 DTC 制御 ..... 40
2.7.11 AAU 制御 ..... 41

3. 参考資料 ..... 42

改訂記録 ..... 43

# 1. DC/DC 制御の概要

本章では、RL78/F2x による デジタル電源システムの DC/DC 制御例について説明します。

## 1.1 デジタル電源制御に関する RL78/F2x の特徴

RL78/F2x は、デジタル電源システムを効率良く制御する以下の様な機能を搭載しています。

(1) タイマ RDe の拡張 PWM モードを使用して、同一周期の PWM 波形を最大 4 本（タイマ RDe カウンタあたり 2 本）まで出力可能

(2) タイマ RDe の PWM 出力はディザリング機能に対応

(3) タイマ RDe の PWM 出力はゲート機能に対応

(4) マイクロコントローラ外部からタイマリスタート端子（TRDiRES）への信号入力により、ソフトウェアを介さずに PWM 出力の中断、再開が可能

(5) A/D 変換器の性能向上による応答性の向上

(6) 演算支援用ハードウェア（AAU）の搭載により、デジタル電源の PI 制御の演算を効率的に実行可能

## 1.2　デジタル電源制御で使用する RL78/F2x の周辺機能

### 1.2.1　タイマ RDe

(1)　ディザリング機能

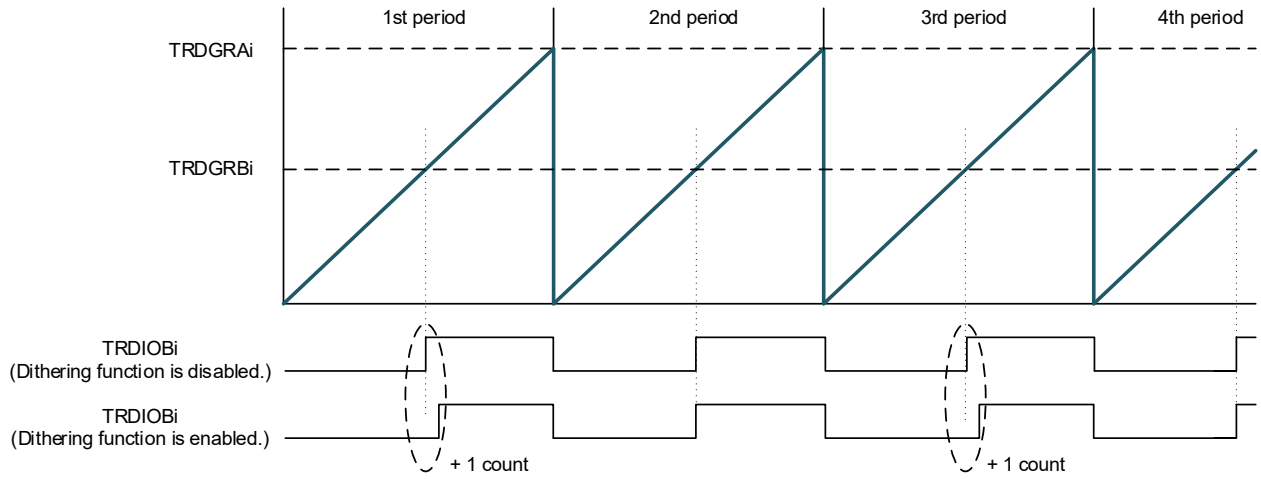
拡張 PWM モード使用時、TRDIOBi と TRDIODi (i = 0 ,1)の出力にディザリング機能を使用することができます。PWM 波形の 16 周期を基準として、選択した周期のアクティブ・パルス幅を 1 カウント分だけ延長することにより、PWM の平均分解能を向上させることができます。

TRDDNRi レジスタの DNR3～DNR0 ビットの設定により、アクティブ・パルス幅を 1 カウント延長する周期を選択します。下図はディザリング機能による PWM 出力動作を示しています。

| DTEN | DNR[3:0] | PWM Output Period | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 0000B | | | | | | | | | | | | | | | | |
| | 0001B | ■ | | | | | | | | | | | | | | | |
| | 0010B | ■ | | | | | | | | ■ | | | | | | | |
| | 0011B | ■ | | | | ■ | | | | ■ | | | | | | | |
| | 0100B | ■ | | | | ■ | | | | ■ | | | | ■ | | | |
| | 0101B | ■ | | ■ | | ■ | | | | ■ | | | | ■ | | | |
| | 0110B | ■ | | ■ | | ■ | | | | ■ | | ■ | | ■ | | | |
| | 0111B | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | | | |
| | 1000B | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | |
| | 1001B | ■ | ■ | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | | ■ | |
| | 1010B | ■ | ■ | ■ | | ■ | | ■ | | ■ | ■ | ■ | | ■ | | ■ | |
| | 1011B | ■ | ■ | ■ | | ■ | ■ | ■ | | ■ | ■ | ■ | | ■ | | ■ | |
| | 1100B | ■ | ■ | ■ | | ■ | ■ | ■ | | ■ | ■ | ■ | | ■ | ■ | ■ | |
| | 1101B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | ■ | ■ | ■ | | ■ | ■ | ■ | |
| | 1110B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | |
| | 1111B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | |
| 0 | - | | | | | | | | | | | | | | | | |

■ ディザリング機能により PWM 値の ON 時間を 1 カウント変化させる周期
□ レジスタに設定された PWM 値を出力する周期

**図 1-1 ディザリング機能による PWM 出力動作仕様**




Conditions: TRDDNRi.DNR[3:0] = 1000B (1 clock is added in the 1st , 3rd , 5th, 7th, 9th, 11th, 13th, and 15th periods.)
TRDPOCRi.POLB = 1 (Active level: High)

**図 1-2 ディザリング機能による PWM 出力波形例**

(2) ゲート機能

拡張 PWM モード使用時、TRDIOBi と TRDIODi(i = 0 ,1)の出力にゲート機能を使用することができます。PWM の 16 周期を基準として、選択した n 周期の期間マスク（ゲート）制御します。TRDGPRi レジスタの GPR3～GPR0 ビットの設定により、出力マスクする周期を選択します。下図はゲート機能の PWM 出力動作を示しています。

TRDGPRi.GPAT = 0（パターン 1 選択）：

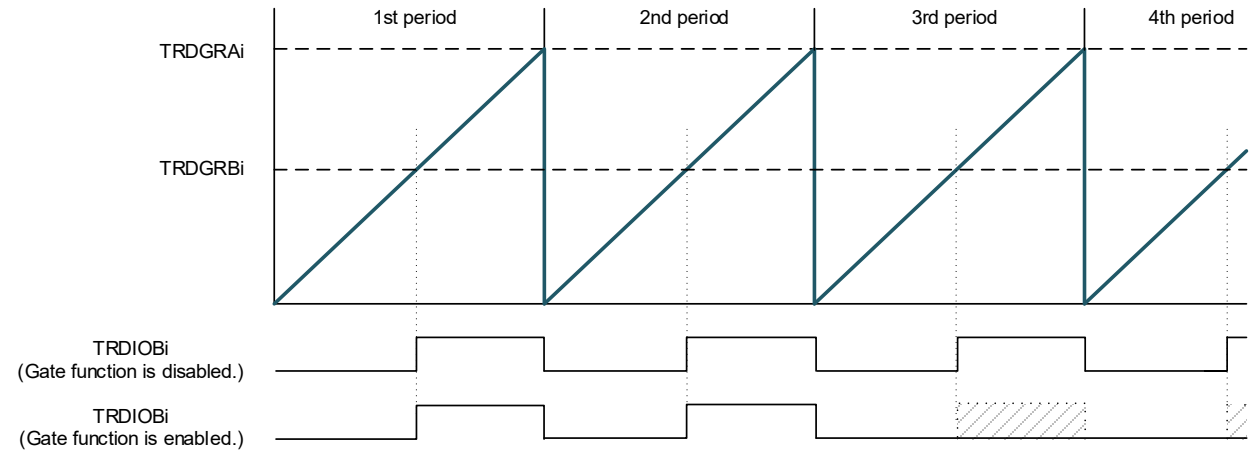
| GTEN | GPR[3:0] | PWM Output Period | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 0000B | | | | | | | | | | | | | | | | |
| | 0001B | ■ | | | | | | | | | | | | | | | |
| | 0010B | ■ | ■ | | | | | | | | | | | | | | |
| | 0011B | ■ | ■ | ■ | | | | | | | | | | | | | |
| | 0100B | ■ | ■ | ■ | ■ | | | | | | | | | | | | |
| | 0101B | ■ | ■ | ■ | ■ | ■ | | | | | | | | | | | |
| | 0110B | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | | | | | |
| | 0111B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | | | | |
| | 1000B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | | | |
| | 1001B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | | |
| | 1010B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | | |
| | 1011B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | | |
| | 1100B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | | |
| | 1101B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | |
| | 1110B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | |
| | 1111B | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | |
| 0 | - | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |

TRDGPRi.GPAT = 1（パターン 2 選択）：

| GTEN | GPR[3:0] | PWM Output Period | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 0000B | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0001B | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0010B | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0011B | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0100B | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0101B | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0110B | | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 0111B | | | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 1000B | | | | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| | 1001B | | | | | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ |
| | 1010B | | | | | | | | | | | | ■ | ■ | ■ | ■ | ■ |
| | 1011B | | | | | | | | | | | | | ■ | ■ | ■ | ■ |
| | 1100B | | | | | | | | | | | | | | ■ | ■ | ■ |
| | 1101B | | | | | | | | | | | | | | | ■ | ■ |
| | 1110B | | | | | | | | | | | | | | | | ■ |
| | 1111B | | | | | | | | | | | | | | | | |
| 0 | - | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |

■ レジスタに設定された PWM 値を出力する周期
□ PWM 出力がマスクされる周期（ゲート機能有効）

**図 1-3 ゲート機能の PWM 出力動作仕様**

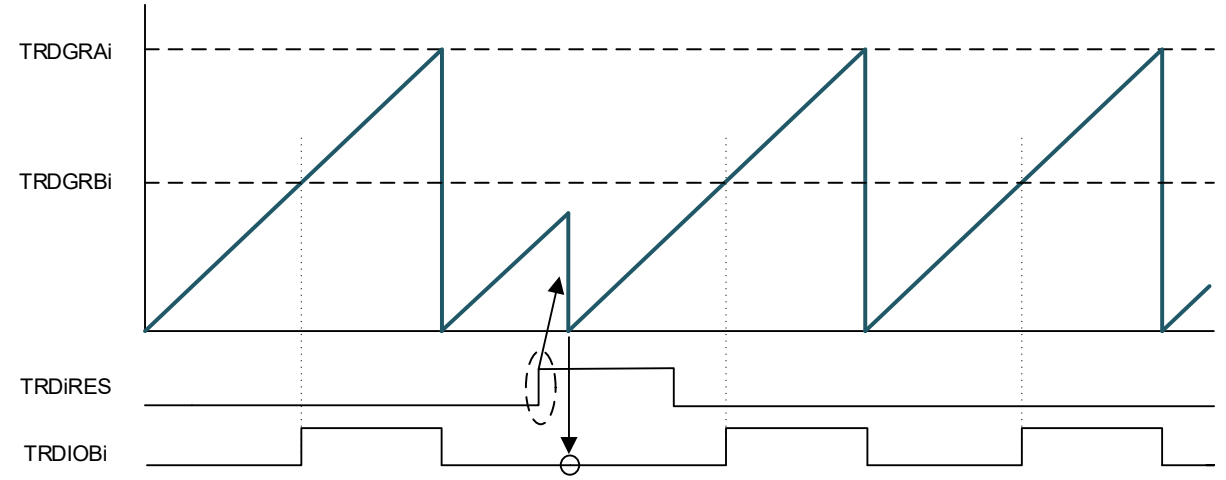



Conditions: TRDGPRi.GPAT = 0 (When Patter.1 is selected.)
TRDGPRi.GPR[3:0] = 0010B (Only 1st and 2nd periods are PWM output. PWM output is masked after 3rd periods.
TRDPOCRi.POLB = 1 (Active level: High)

**図 1-4 ゲート機能の PWM 出力波形例**

(3) カウンタ・リスタート機能

拡張 PWM モードでは、カウンタ・リスタート機能を使用することで、TRDiRES (i = 0, 1)端子入力によりタイマ RDi カウンタ（TRDi）を初期化し、再びカウントを開始することができます。下図にカウンタ・リスタート機能の PWM 出力動作例を示します。




Conditions: TRDEMRi.CCLV[1:0] = 10B (Counter restart at the rising edge of TRDiRES input.)
TRDOCRi.TOBi = 0 (Initial output level: Low)
TRDPOCRi.POLB = 1 (Active level: High)

**図 1-5 カウンタ・リスタート機能の PWM 出力動作例**

### 1.2.2 アプリケーション・アクセラレータ・ユニット（AAU）

AAU の演算機能を使用して、DC/DC 制御向けの PI 演算アルゴリズムを実行することができます。1 チャネル用から 3 チャネル用の処理を選択して使用できます。

DC/DC 制御向け PI 演算アルゴリズムを以下に示します。

D (n) = D (n-1) + A1 × E (n) + A2 × E (n-1)

D (n): Calculated Duty value
D (n-1): Previous Duty value
E (n): Current error value = Reference current value − Measurement current value
E (n-1): Last current error value = Reference current value − Last measurement current value
A1, A2: Coefficient

n: Sampling count number

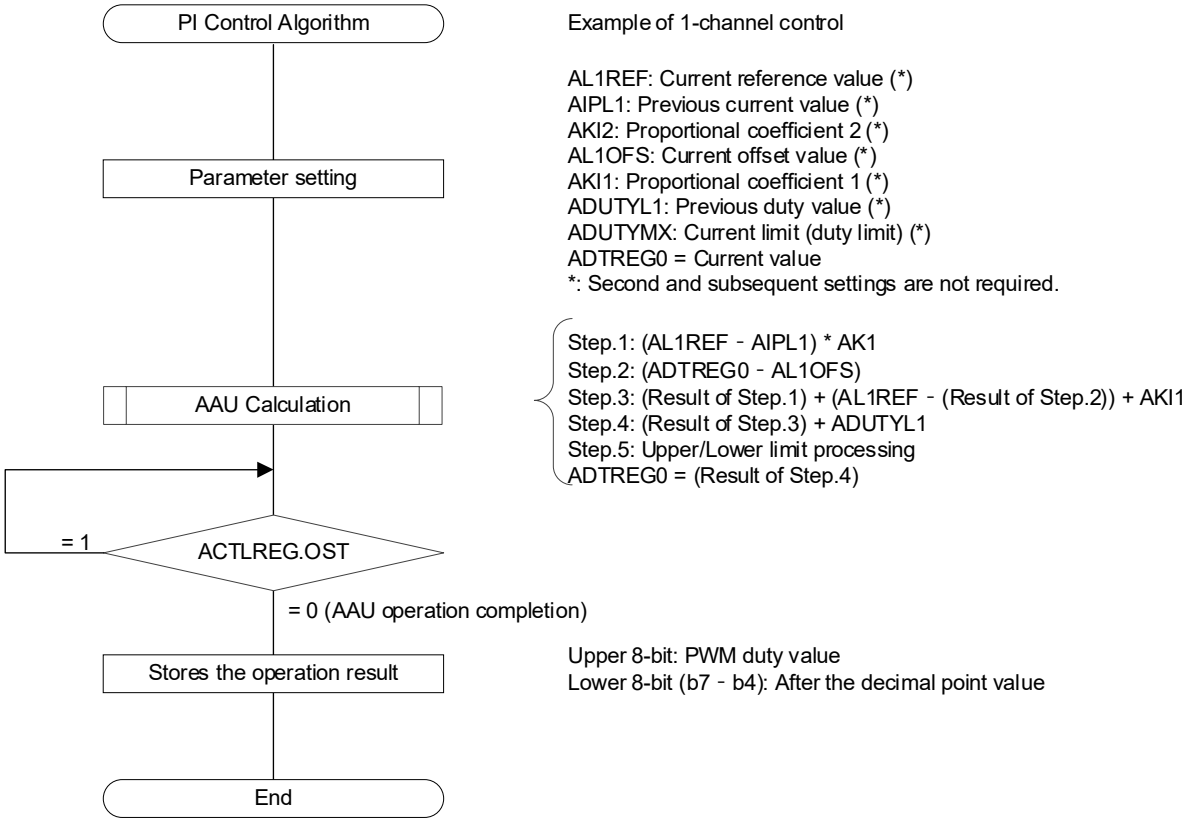



**図 1-6 DC/DC 制御向け PI 演算アルゴリズム**

## 1.3 デジタル電源制御のハードウェア構成

DC/DC 制御向けの PI 演算アルゴリズムを用いたデジタル電源制御のハードウェア構成を図 1-7 に示します。また、ハードウェア仕様を表 1-1 に示します。

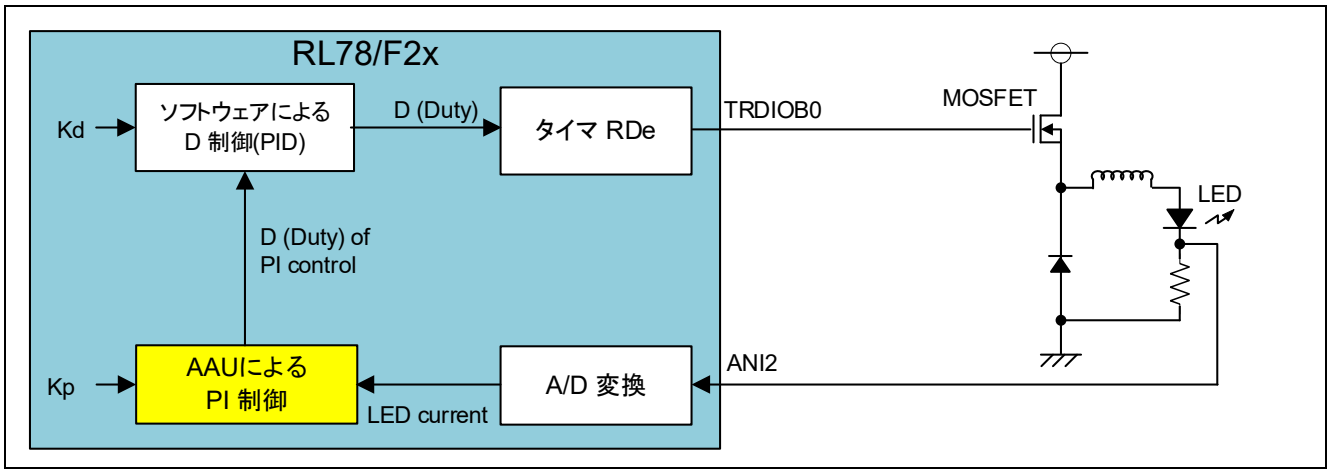



**図 1-7 デジタル電源制御用ハードウェア構成**

**表 1-1 ハードウェア仕様**

| 機能 | 仕様 |
|---|---|
| CPU | $f_{CLK}$ = 40 MHz |
| タイマ RDe | • $f_{MP}$ = 80 MHz を $f_{TRD}$（カウントクロック）として使用<br>• 拡張 PWM モードを使用<br>• PWM 出力（ディザリング機能を使用） |
| A/D コンバータ | • LED 電流測定に使用<br>• タイマ RDe の INTTRD0_IFA を A/D 変換開始イベントに使用 |
| DTC | • 高速 DTC を使用<br>• A/D 終了割り込みを DTC 転送開始要求に使用 |
| アプリケーション・アクセラレータ・ユニット（AAU） | • DC/DC 制御用 PI 演算で使用 |

## 1.4 使用端子一覧

表 1-2 に RL78/F2x の使用端子とデジタル電源制御システムにおける機能を示します。

**表 1-2 使用端子一覧**

| 使用端子 | Input / Output | 機能 |
|---|---|---|
| TRDIOB0/P11 | Output | タイマ RDe の PWM 出力用 |
| ANI2/P82 | Input | デジタル電源からのフィードバック電流入力用 |

# 2. 制御ソフトウェア

本アプリケーションノートの制御ソフトウェアは、タイマ RDe の拡張 PWM 出力により昇圧型 DC/DC コンバータのゲート制御を行います。また、デジタル電源回路からのフィードバック電流値を A/D コンバータで監視し、PID 制御によりゲートの開閉タイミングを調整します。

## 2.1 RL78/F2x 内蔵機能の初期化

制御ソフトウェアは、以下の RL78/F2x マイクロコントローラ内蔵周辺機能を使用します。

### 2.1.1 内蔵クロック

制御ソフトウェアでは、内蔵クロックを以下の設定で使用しています。

- タイマ RDe カウントクロックとして、PLL クロック（80MHz）を使用
- メイン・システムクロックとして、PLL クロック（40MHz）を使用

### 2.1.2 タイマ RDe

制御ソフトウェアでは、タイマ RDe を以下の設定で使用しています。

**表 2-1 タイマ RDe の設定**

| 項目 | 内容 | 用途 |
|---|---|---|
| カウントクロック($f_{TRD}$) | $f_{MP}$（PLL クロック）= 80 [MHz] | PWM 出力<br>ディザリンク機能 |
| 使用モード | 拡張 PWM モード | |
| 出力端子 | TRDIOB0 | |
| PWM 周期 | 5 [μs] | |
| 出力レベル | 初期出力：Low、アクティブ：High | |
| A/D 変換トリガ | INTTRD0_IFA を A/D スタートトリガとして出力 | |
| その他機能 | • ゲート機能は使用しません<br>• カウンタ・リスタート機能は使用しません | |

### 2.1.3 A/D コンバータ

制御ソフトウェアでは、A/D コンバータを以下の設定で使用しています。

**表 2-2 A/D コンバータの設定**

| 項目 | 内容 | 用途 |
|---|---|---|
| 使用チャネル | ANI2 （高速アナログ入力端子）を使用 | 電流値測定 |
| A/D 変換クロック（ADCLK） | $f_{CLK}$/1 = 40 [MHz] | |
| 動作モード | シングル・スキャン・モード | |
| 同期トリガ | タイマ RDe の割り込み（INTTRD0_IFA）を A/D 変換開始の同期トリガとして使用する | |
| 基準電圧 | $AV_{REFP}$、$AV_{REFM}$ を使用する | |
| 割り込み出力 | ANI2 端子のスキャン終了ごとに INTAD（A/D 変換終了）を生成 | |

### 2.1.4 DTC

制御ソフトウェアでは、DTC を以下の設定で使用しています。

**表 2-3 DTC の設定**

| 項目 | 内容 | 用途 |
|---|---|---|
| 使用チャネル | 高速 DTC チャネル 0 を使用 | A/D スキャン結果の転送 |
| 使用モード | リピート・モード | |
| データサイズ | 16 ビットデータ、<br>ANI2 の A/D 変換結果を 8 回転送 | |
| DTC ベクタ<br>ベースアドレス | FFD00H | |
| DTC 起動トリガ | A/D 変換終了割り込み | |

### 2.1.5 アプリケーション・アクセラレーション・ユニット(AAU)

制御ソフトウェアでは、AAU を以下の設定で使用しています。

**表 2-4 AAU の設定**

| 項目 | 内容 | 用途 |
|---|---|---|
| 演算アルゴリズム | DC/DC 制御向け PI 演算モード（1 チャネル）を使用 | PI 演算の実施 |
| 開始トリガ | ソフトウェア・トリガ（OST ビットへの書き込みにより開始） | |

## 2.2 制御ソフトウェアの DC/DC 制御

制御ソフトウェアの DC/DC 制御は、タイマ RDe の拡張 PWM モード機能を使用し、図 2-1 の様に RL78/F2x の TRDIOB0 端子より PWM 波形を出力します。PWM 波形の High 出力、Low 出力は、タイマ RDe のレジスタ TRDGRA0 と TRDGRB0 レジスタに設定するカウント値と、TRD0 カウンタとのコンペアマッチにより制御することができます。

また、PWM 信号波形は、デジタル電源の MOSFET に入力され、昇圧回路のゲート ON/OFF 制御に使用されます（図 1-7 参照）。昇圧回路の DC/DC 制御では、MOSFET のゲート ON 時間を長く設定すると、デジタル電源のチャージ（蓄電）期間を長くすることができます。デジタル電源の電圧は、回路に接続されている LED で消費されます。安定した電圧供給により、LED を安定して点灯させるため、制御ソフトウェアではデジタル電源からの現在の電流値を A/D 変換により取得し、PID 制御により PWM 信号にフィードバックします。

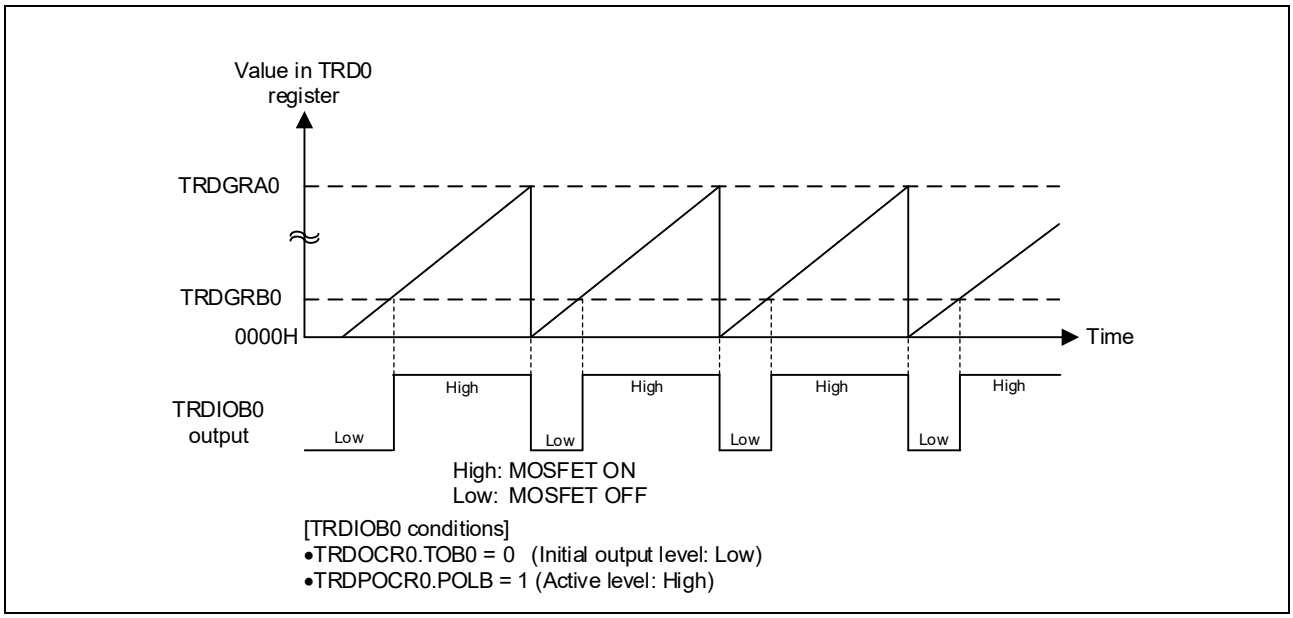



**図 2-1 DC/DC ゲート制御例**

### 2.2.1 A/D 値の取得制御

制御ソフトウェアでは、タイマ RD0 に同期して周期的に LED 電流値の A/D 変換を行います。変換結果の A/D 値は、DTC により RAM 領域へ転送されます。制御ソフトウェアでは、DTC の設定により 8 回の RAM 転送を 1 セットとして扱います。8 回の転送完了後、A/D 変換終了割り込みが CPU に通知され、A/D 割り込み処理ハンドラが実行されます。

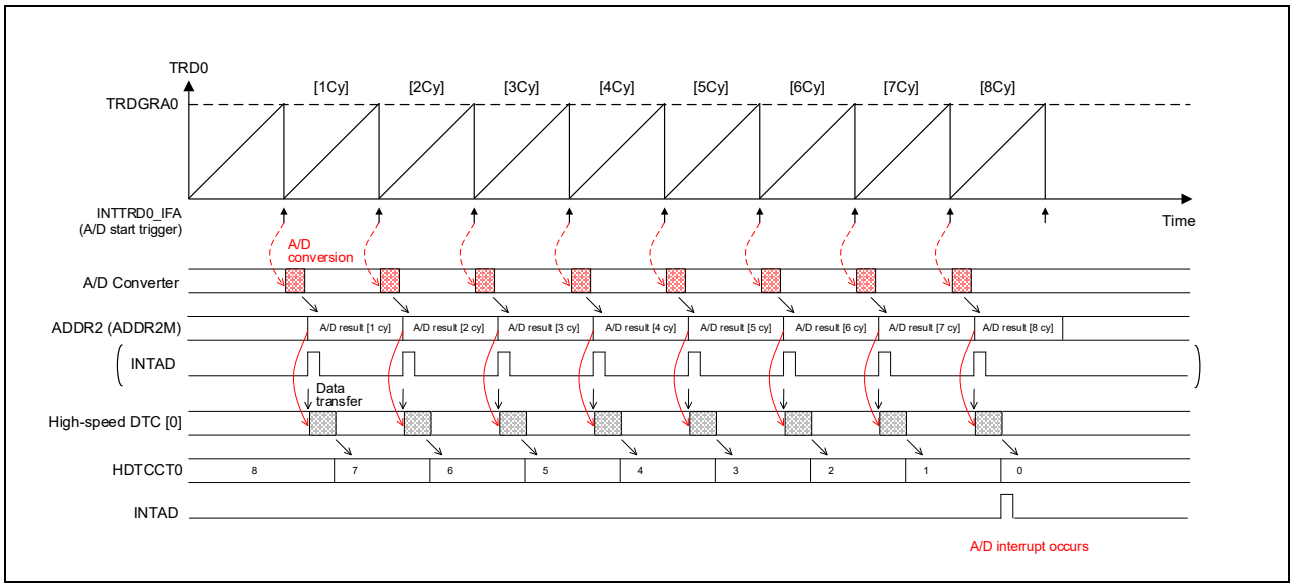



**図 2-2 制御ソフトウェアの A/D 取得イメージ**

制御ソフトウェアでは、8 回の A/D 値の平均を求め、結果を LED 電流値（計測値）として使用します。

### 2.2.2 PI 制御

制御ソフトウェアの PID 制御のうち、P 制御（比例）と I 制御（積分）の演算は、RL78/F2x の AAU を使用します。

制御ソフトウェアの PI 演算は、以下の手順で実施します。

(1) PI 演算で使用する各パラメータを AAU レジスタに設定する

・AAU に各パラメータを設定
ADUTYMX（電流デューティ制限値）、AKI1（比例定数 Kp1）、AKI2（比例定数 Kp2）、
AL1REF（基準電流）、AL1OFS（電流オフセット）、ADUTYL1（前回のデューティ値）、
AIPL1（前回の電流値）

(2) AAU による演算を実行する

・取得した電流値を AAU レジスタに設定
ADTREG0 = 電流値;
・PID 演算後の前回の PWM デューティ値を AAU レジスタに設定
ADUTYL1 = 前回の PWM デューティ値（ディザリング回数値を含む）
・演算を開始する
ACTLREG.OST ← 1

(3) AAU による演算完了を確認する

・ACLTREG.OST ビットが 0 を確認する

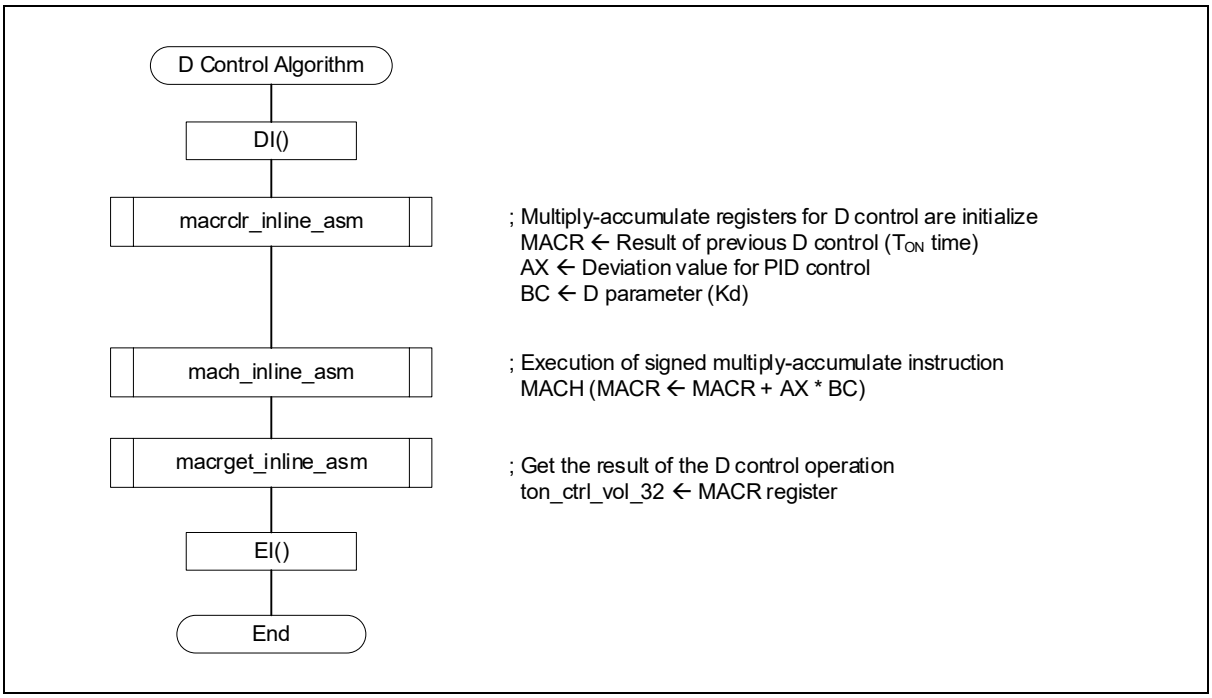
(4) PI 演算の結果を保存する

・AAU の演算結果（ADTREG0）を保存する

### 2.2.3 D 制御

制御ソフトウェアの PID 制御のうち、D 制御（微分）は、RL78/F2x の CPU 命令によるソフトウェア演算を行います。D 制御の実施条件として、A/D 値（計測値）と基準電流値（目標値）に偏差がある場合に制御を実施します。

(1) マイコン命令により D 制御の演算を行う




(2) D 制御の演算結果より、次回のデューティ値、ディザリング回数を設定する

・次回の PWM デューティ値取得
```
led_ch1_pid_param.dutys[1] = (uint16_t)(ton_ctrl_vol_32 >> 16);
```
・次回のディザリング回数取得
```
led_ch1_pid_param.remainder[1] = (16U) - (uint8_t)(ton_ctrl_vol_32 & 0x0000FFFF) >> 12);
```

### 2.2.4 PWM へのフィードバック

制御ソフトウェアの PID 制御により得た演算結果から、デューティ値、ディザリング回数をタイマ RDe のバッファ・レジスタに設定します。制御ソフトウェアでは、タイマ RDe の設定値更新の際にタイマレジスタの一斉書換え機能を使用します。

(1) PID 制御の演算結果をタイマ値に反映する

・TRDGRB0 のバッファ・レジスタ TRDGRD0 に PWM デューティ値を書き込む
```
trd0_update->trdgrdn = led_ch1_pid_param.dutys[1];   // タイマデューティ値
```
・TRDDNR0 レジスタにディザリング回数を書き込む
```
trd0_update->trddnrn = led_ch1_pid_param.remainder[1];   // ディザリング回数
```

(2) タイマレジスタの一斉書き換え要求

・TRDRDT0 レジスタの RDT ビットに 1 を書き込む
```
R_Config_TRD0_Set_TRD0_ReloadTrigger(trd0_update);
```

## 2.3 DC/DC 制御フロー

図 2-3 に DC/DC 制御の全体フローを示します。

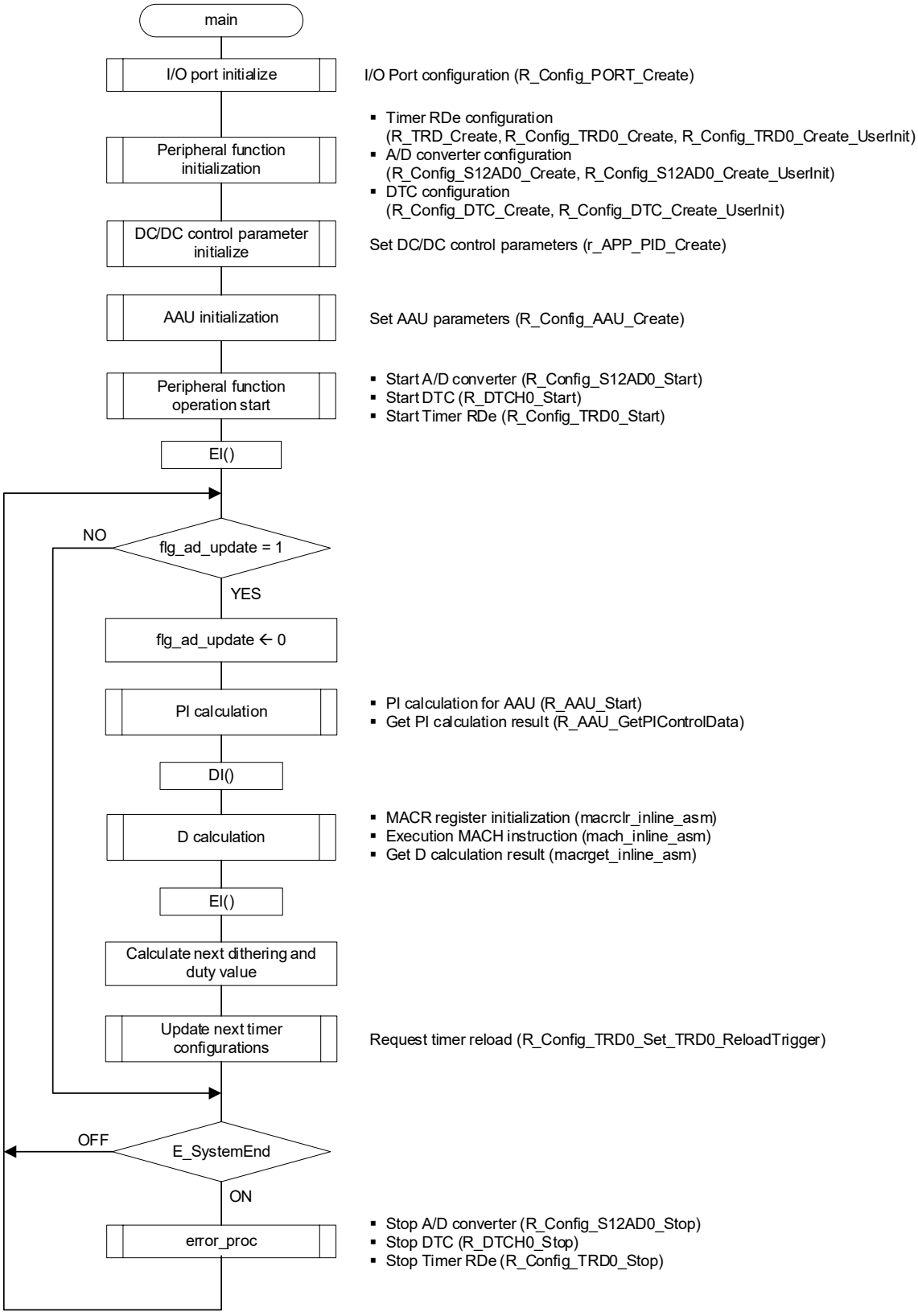



**図 2-3 DC/DC 制御の全体フローチャート (1/2)**

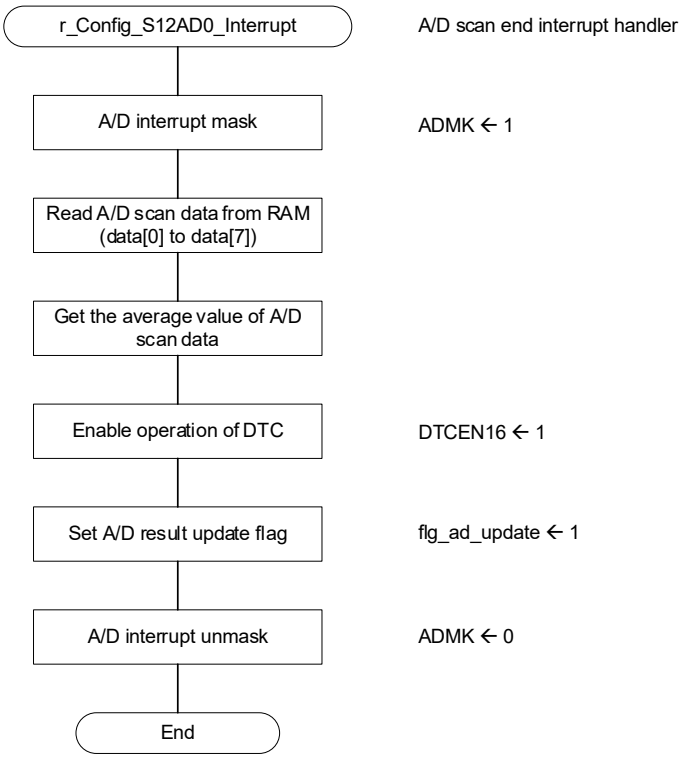



**図 2-3 DC/DC 制御の全体フローチャート (2/2)**

## 2.4 変数一覧

制御ソフトウェアで使用する変数型定義を表 2-5 に示します。

**表 2-5 PID_PARAM 型定義**

| メンバ名 | 型 | 用途 | 使用目的 |
|---|---|---|---|
| paramKp1 | int16_t | PI 演算用 | PI 演算用の定数 |
| paramKp2 | int16_t | PI 演算用 | PI 演算用の定数 |
| duty_limit | uint8_t | PI 演算用 | 電流デューティの最大値 |
| current_offset | uint16_t | PI 演算用 | 電流値オフセット |
| currents[2] | uint16_t | PI 演算用 | 電流計測値（[0]:前回値、[1]:最新値） |
| dutys[2] | uint16_t | PI 演算用 | 電流デューティ（[0]:前回値、[1]:最新値） |
| remainder[2] | uint16_t | PI 演算用 | ディザリング設定値（[0]:前回値、[1]:最新値） |
| paramKd | uint16_t | D 制御用 | D 制御演算用の定数 |
| devias[2] | int16_t | D 制御用 | 偏差（基準電流値 – 計測電流値）（[0]:前回値、[1]:最新値） |
| differs[2] | int16_t | D 制御用 | 偏差の差（[0]:前回値、[1]:最新値） |
| current_ad | uint16_t | D 制御用 | 現在の A/D 値 |
| current_target | int16_t | PI,D 共通 | 目標電流値 |
| dither_count | uint8_t | PI,D 共通 | ディザリング回数 |

制御ソフトウェアが使用する変数一覧を以下に示します。

**表 2-6 制御ソフトウェアの変数一覧**

| 変数名 | 型 | 使用目的 |
|---|---|---|
| led_ch1_pid_param | PID_PARAM | PID 制御内部管理変数 |
| E_SystemEnd | uint8_t | システム終了フラグ |
| dtc_vectortable[] | uint8_t | DTC ベクタ管理テーブル |
| flg_ad_update | uint8_t | A/D 値更新フラグ |

## 2.5 関数一覧

制御ソフトウェアが使用する関数一覧を以下に示します。

**表 2-7 制御ソフトウェア関数一覧**

| 関数名 | 概要 |
|---|---|
| main | デジタル電源制御のメイン処理 |
| error_proc | システムエラー時処理 |
| r_APP_PID_Create | PID 制御の初期化 |
| r_APP_PID_Start | PID 制御の開始処理 |
| r_APP_PID_Stop | PID 制御の終了処理 |
| r_APP_PID_UpdateParam | PID 制御設定の更新 |
| R_Config_AAU_Create | AAU コンポーネントの初期化 |
| R_AAU_Start | AAU による演算開始処理 |
| R_AAU_GetPIControlData | AAU の演算（DC/DC 制御向け PI 演算）の結果データ取得 |
| R_Config_PORT_Create | ポートの初期化 |
| R_Config_DTC_Create | DTC の初期化 |
| R_Config_DTC_Create_UserInit | DTC の初期化（制御ソフトウェアの固有設定） |
| R_DTCH0_Start | DTC 制御の開始処理 |
| R_DTCH0_Stop | DTC 制御の終了処理 |
| R_Config_S12AD0_Create | A/D コンバータの初期化 |
| R_Config_S12AD0_Create_UserInit | A/D コンバータの初期化（制御ソフトウェアの固有設定） |
| r_Config_S12AD0_Interrupt | A/D コンバータの割り込み処理ハンドラ |
| R_Config_S12AD0_Start | A/D コンバータ制御の開始 |
| R_Config_S12AD0_Stop | A/D コンバータ制御の終了 |
| R_TRD_Create | タイマ RDe の初期化（タイマ制御前の初期化処理） |
| R_Config_TRD0_Create | タイマ RDe の初期化 |
| R_Config_TRD0_Create_UserInit | タイマ RDe の初期化（制御ソフトウェアの固有設定） |
| R_Config_TRD0_Start | タイマ RDe の開始処理 |
| R_Config_TRD0_Stop | タイマ RDe の終了処理 |
| R_Config_TRD0_Set_TRD0_ReloadTrigger | タイマ RDe の設定更新処理 |

## 2.6 関数仕様

サンプルコードの関数仕様を示します。

### 2.6.1 main

| 宣言 | void main(void) |
|---|---|
| 概要 | デジタル電源制御のメイン処理 |
| 説明 | 制御ソフトウェアのメイン処理です。<br>メイン処理開始時、各周辺機能の初期化を行います。<br>• R_Config_PORT_Create : RL78/F2x の I/O ポート設定を行います<br>• R_TRD_Create : タイマ RDe の初期設定を行います<br>• R_Config_S12AD0_Create : 12 ビット A/D コンバータの初期設定を行います<br>• R_Config_DTC_Create : 高速 DTC の初期設定を行います<br>• r_APP_PID_Create : デジタル電源制御のパラメータ設定を行います<br>• R_DTCH0_Start : 高速 DTC の処理を開始します<br>• R_Config_S12AD0_Start : A/D コンバータの処理を開始します<br>• R_Config_TRD0_Start : タイマ RDe0 のカウントを開始します<br>• EI() : 割り込みを許可します<br><br>その後、DC/DC 部の電流値の確定（flg_ad_update = 1）時に、以下の手順でデジタル電源制御を処理します。<br>• R_AAU_Start : AAU に電流値を設定し、演算を開始します<br>• R_AAU_GetPIControlData : AAU の演算結果を取得します<br>• D control calculation : 積和演算命令を使用して D 制御を行い、次回の PWM デューティ、ディザリング回数をタイマ RDe のバッファ・レジスタに設定します<br>• R_Config_TRD0_Set_TRD0_ReloadTrigger : タイマ RDe のバッファ・レジスタからコンペア・レジスタに転送を要求します<br><br>異常を検出（E_SystemEnd = 1）した場合、error_proc 関数を呼び出します |
| 引数 | なし |
| リターン値 | なし |

### 2.6.2 error_proc

| 宣言 | void error_proc(void) |
|---|---|
| 概要 | システムエラー時処理 |
| 説明 | システムエラー発生時、タイマ、A/D コンバータ、DTC を停止します。<br>• R_Config_TRD0_Stop : タイマ RD0 のカウントを停止します<br>• R_Config_S12AD0_Stop : A/D コンバータの動作を終了します<br>• R_DTCH0_Stop : 高速 DTC の動作を終了します |
| 引数 | なし |
| リターン値 | なし |

### 2.6.3 r_APP_PID_Create

| | |
|---|---|
| 宣言 | void r_APP_PID_Create(void) |
| 概要 | PID 制御の初期化 |
| 説明 | PID 制御の内部管理変数 led_ch1_pid_param のメンバに初期値を設定します。<br>[PI 制御用メンバ]<br>• led_ch1_pid_param.paramKp1 = (int16_t) PI_CONTROL_PARAM_A1;<br>• led_ch1_pid_param.paramKp2 = (int16_t) PI_CONTROL_PARAM_A2;<br>• led_ch1_pid_param.duty_limit = (uint8_t)PI_CONTROL_CURRENT_DUTY_LIMIT;<br>• led_ch1_pid_param.current_offset = (uint16_t)PI_CONTROL_CURRENT_AD_OFFSET;<br>• led_ch1_pid_param.current_target = (int16_t) PI_CONTROL_CURRENT_AD_TARGET;<br>• led_ch1_pid_param.currents[0] = 0;<br>• led_ch1_pid_param.currents[1] = 0;<br>• led_ch1_pid_param.dutys[0] = 0;<br>• led_ch1_pid_param.dutys[1] = (uint16_t)PI_CONTROL_CURRENT_DUTY_INIT;<br><br>[D 制御用メンバ]<br>• led_ch1_pid_param.paramKd = (uint16_t)D_CONTROL_PARAM_KD;<br>• led_ch1_pid_param.devias[0] = 0;<br>• led_ch1_pid_param.devias[1] = 0;<br><br>[PI, D 制御共通]<br>• led_ch1_pid_param.dither_count = 8;<br><br>最後に AAU 初期化のため R_Config_AAU_Create 関数をコールする。 |
| 引数 | なし |
| リターン値 | なし |

### 2.6.4 r_APP_PID_UpdateParam

| | |
|---|---|
| 宣言 | void r_APP_PID_UpdateParam(void) |
| 概要 | PID 制御設定の更新 |
| 説明 | PID 制御の終了時に、led_ch1_pid_param メンバの前回値に今回演算で使用した値をコピーします。<br>置き換えを行う led_ch1_pid_param メンバは以下の通りです。<br>• led_ch1_pid_param.currents[0] ← led_ch1_pid_param.currents[1];<br>• led_ch1_pid_param.dutys[0] ← led_ch1_pid_param.dutys[1];<br>• led_ch1_pid_param.devias[0] ← led_ch1_pid_param.devias[1];<br>• led_ch1_pid_param.differs[0] ← led_ch1_pid_param.differs[1]; |
| 引数 | なし |
| リターン値 | なし |

### 2.6.5 R_Config_AAU_Create

<table>
<tr><td>宣言</td><td>void R_Config_AAU_Create(void)</td></tr>
<tr><td>概要</td><td>AAU コンポーネントの初期化</td></tr>
<tr><td>説明</td><td>AAU コンポーネントに対し以下のレジスタ設定を行います。
<table>
<tr><th>レジスタ名</th><th>AAUWINR</th><th>設定値</th><th>内容</th></tr>
<tr><td>PER2</td><td>–</td><td>0x40</td><td>AAU に入力クロックを供給</td></tr>
<tr><td>ACTLREG</td><td>0x00</td><td>0x82</td><td>演算モード「PI control for DC/DC control 1ch」を選択</td></tr>
<tr><td>ADUTYMX</td><td>0x00</td><td>ユーザ設定</td><td>led_ch1_pid_param.duty_limit：電流デューティの最大値</td></tr>
<tr><td>AL1REF</td><td>0x00</td><td>ユーザ設定</td><td>led_ch1_pid_param.current_target：目標電流値</td></tr>
<tr><td>AKI1</td><td>0x01</td><td>ユーザ設定</td><td>led_ch1_pid_param.paramKp1：PI 演算用の定数</td></tr>
<tr><td>AKI2</td><td>0x01</td><td>ユーザ設定</td><td>led_ch1_pid_param.paramKp2：PI 演算用の定数</td></tr>
<tr><td>AL1OFS</td><td>0x01</td><td>ユーザ設定</td><td>led_ch1_pid_param.current_offset：電流値オフセット</td></tr>
<tr><td>AIPL1</td><td>0x02</td><td>ユーザ設定</td><td>led_ch1_pid_param.currents[0]：電流計測値（前回値）</td></tr>
<tr><td>ADUTYL1</td><td>0x02</td><td>ユーザ設定</td><td>led_ch1_pid_param.dutys[0]：電流デューティ（前回値）</td></tr>
</table></td></tr>
<tr><td>引数</td><td>なし</td></tr>
<tr><td>リターン値</td><td>なし</td></tr>
</table>

### 2.6.6 R_AAU_Start

<table>
<tr><td>宣言</td><td>void R_AAU_Start(void)</td></tr>
<tr><td>概要</td><td>AAU による演算開始</td></tr>
<tr><td>説明</td><td>AAU コンポーネントに対し以下のレジスタ設定を行います。
<table>
<tr><th>レジスタ名</th><th>AAUWINR</th><th>設定値</th><th>内容</th></tr>
<tr><td>ADTREG0</td><td>0x00</td><td>電流計測値</td><td>led_ch1_pid_param.currents[1]：電流計測値（最新値）</td></tr>
<tr><td>ADUTYL1</td><td>0x02</td><td>前回の PWM デューティ値</td><td>((led_ch1_pid_param.duty[1] << 7) & 0xFF80) |<br>((led_ch1_pid_param.remainder[1] << 5) & 0x0070)：前回の PWM デューティ値</td></tr>
</table>
ADTREG0 レジスタに最新の電流計測値を設定し、PI 演算を開始します。<br>演算開始後、6 クロック経過後に演算が終了します。</td></tr>
<tr><td>引数</td><td>なし</td></tr>
<tr><td>リターン値</td><td>なし</td></tr>
</table>

### 2.6.7 R_AAU_GetPIControlData

| | |
|---|---|
| 宣言 | void R_AAU_GetPIControlData(void) |
| 概要 | AAU の演算（DC/DC 制御向け PI 演算）の結果データ取得 |
| 説明 | AAU 演算の終了を確認し、演算結果を取得します。<br>演算終了の確認方法は、ACTLREG.OST ビットの状態を読み出すことで確認できます。<br>本関数では、演算完了時に演算結果が格納される ADTREG0 レジスタ値を取得します。<br>取得後、以下の led_ch1_pid_param メンバに演算結果を格納します。<br>• led_ch1_pid_param.current_ad ← ADTREG0; PI 演算結果 |
| 引数 | なし |
| リターン値 | なし |

### 2.6.8 R_Config_DTC_Create

| | |
|---|---|
| 宣言 | void R_Config_DTC_Create(void) |
| 概要 | DTC の初期化 |
| 説明 | DTC コンポーネントに対し以下のレジスタ設定を行います。 |

| レジスタ名 | 設定値 | 内容 |
|---|---|---|
| PER1 | 0x80 | DTC に入力クロックを供給 |
| DTCEN0 | 0x00 | すべての DTC を起動禁止に設定 |
| DTCEN1 | 0x00 | |
| DTCEN2 | 0x00 | |
| DTCEN3 | 0x00 | |
| DTCEN4 | 0x00 | |
| DTCEN5 | 0x00 | |
| DTCBAR | 0xFD | DTC ベクタ・テーブル領域のベースアドレスを FFD00H（RAM 領域）に設定 |
| SELHS0 | 0x09 | 起動要因の要因番号 9（INTAD）を高速チャネル 0 に選択 |
| HDTCCR0 | 0x61 | "HSZ0=1 (Size: 16-bit), HRPTINT0=1 (Enable interrupt), HMODE0=1 (Repeat mode)" |
| HDTCCT0 | 0x08 | データ転送回数を 8 回に設定 |
| HDTRLD0 | 0x08 | データ転送回数を 8 回に設定（リロード用設定） |
| HDTSAR0 | 0x06A4 | A/D コンバータの ADDR2M（アドレス：0x06A4）を、高速 DTC のソースアドレスに指定 |
| HDTDAR0 | ユーザ設定 | 高速 DTC の転送先アドレスを指定 |

| | |
|---|---|
| | 最後に制御ソフトウェア固有設定のため、R_Config_DTC_Create_UserInit()をコールします。 |
| 引数 | なし |
| リターン値 | なし |

### 2.6.9 R_Config_DTC_Create_UserInit

| | |
|---|---|
| 宣言 | void R_Config_DTC_Create_UserInit(void) |
| 概要 | DTC の初期化（制御ソフトウェアの固有設定） |
| 説明 | DTC ベクタ・テーブル領域の初期化と、DTC の起動要因を設定します。<br>DTC 転送の起動要因として、要因番号 9（A/D 変換終了）を設定します。<br><br>dtc_vectortable[9U] = 0x40U; /* A/D 変換終了 */ |
| 引数 | なし |
| リターン値 | なし |

### 2.6.10 R_DTCH0_Start

| | |
|---|---|
| 宣言 | void R_DTCH0_Start(void) |
| 概要 | DTC の動作開始 |
| 説明 | DTC 転送の起動要因として、A/D 変換終了割り込みを許可します。<br><br>DTCEN1 \|= 0x40; /* A/D 変換終了 */ |
| 引数 | なし |
| リターン値 | なし |

### 2.6.11 R_DTCH0_Stop

| | |
|---|---|
| 宣言 | void R_Config_DTC_Stop(void) |
| 概要 | DTC の動作終了 |
| 説明 | DTC 転送の起動を禁止します。<br><br>DTCEN1 &= ~0x40; /* DTC 起動禁止 */ |
| 引数 | なし |
| リターン値 | なし |

### 2.6.12 R_Config_S12AD0_Create

<table>
<tr><td>宣言</td><td>void R_Config_S12AD0_Create(void)</td></tr>
<tr><td>概要</td><td>A/D コンバータの初期化</td></tr>
<tr><td>説明</td><td>A/D コンバータに対し以下のレジスタ設定を行います。
<table>
<tr><th>レジスタ名</th><th>ADWINR</th><th>設定値</th><th>内容</th></tr>
<tr><td>PER0</td><td>-</td><td>0x20</td><td>A/D コンバータへクロック供給</td></tr>
<tr><td>ADCSR</td><td>0x00</td><td>0x1200</td><td>ADCS[1:0]=00B（シングル・スキャン・モード）、<br>ADIE=1（INTAD 割り込み許可）、<br>TRGE=1（トリガ入力による A/D 変換開始を許可）</td></tr>
<tr><td>ADANSA0</td><td>0x00</td><td>0x0004</td><td>ANI2 を変換対象に設定</td></tr>
<tr><td>ADSTRGR</td><td>0x01</td><td>0x0400</td><td>TRSA[5:0]=000100B（A/D 変換開始トリガ：INTTRD0_IFA）</td></tr>
<tr><td>ADHVREFCNT</td><td>0x08</td><td>0x11</td><td>基準電源選択（AVREFM、AVREFP）</td></tr>
<tr><td>ADCER</td><td>0x00</td><td>0x0000</td><td>ADRFMT=0（右詰め）、DIAGM=0（自己診断未使用）</td></tr>
<tr><td>ADDISCR</td><td>0x07</td><td>0x00</td><td>断線検出アシスト機能未使用</td></tr>
<tr><td>ADSSTR2</td><td>0x0E</td><td>0x0D</td><td>ANI2 サンプルイング・ステート設定（13.5 ステート）</td></tr>
</table>
上記設定後、ANI2、AVREFP、AVREFM 端子のポート設定を行います。<br>
ANI2:<br>
• PMC8.PMC82 をアナログ入力に設定します（PMC82 = 1）<br>
• PM8.PM82 を入力モードに設定します（PM82 = 1）<br><br>
AVREFP:<br>
• PMC3.PMC33 をアナログ入力に設定します（PMC33 = 1）<br>
• PM3.PM33 を入力モードに設定します（PM33 = 1）<br><br>
AVREFM:<br>
• PMC3.PMC34 をアナログ入力に設定します（PMC34 = 1）<br>
• PM3.PM34 を入力モードに設定します（PM34 = 1）<br><br>
最後に制御ソフトウェア固有設定のため、R_Config_S12AD0_Create_UserInit()をコールします。</td></tr>
<tr><td>引数</td><td>なし</td></tr>
<tr><td>リターン値</td><td>なし</td></tr>
</table>

### 2.6.13 R_Config_S12AD0_Create_UserInit

| | |
|---|---|
| 宣言 | void R_Config_S12AD0_Create_UserInit(void) |
| 概要 | A/D コンバータの初期化（制御ソフトウェアの固有設定） |
| 説明 | A/D コンバータの初期化処理（固有設定）を行います。<br>本制御ソフトウェアでは、処理を行っていません。 |
| 引数 | なし |
| リターン値 | なし |

### 2.6.14 R_Config_S12AD0_Start

| | |
|---|---|
| 宣言 | void R_Config_S12AD0_Start(void) |
| 概要 | A/D コンバータの動作開始 |
| 説明 | A/D 変換を開始します。<br>• IF1H.ADIF の割り込み要求フラグビットを 0 にクリアします<br>• MK1H. ADMK の割り込みマスクビットを 0 にクリアし、割り込み発生を許可します<br>• ADWINR レジスタに 0x00 を設定します<br>• ADCSR.TRGE のビットを 1 に設定します（トリガ入力による A/D 変換の開始設定を有効にします） |
| 引数 | なし |
| リターン値 | なし |

### 2.6.15 R_Config_S12AD0_Stop

| | |
|---|---|
| 宣言 | void R_Config_S12AD0_Stop(void) |
| 概要 | A/D コンバータの動作終了 |
| 説明 | A/D 変換を終了します。<br>• ADWINR レジスタに 0x00 を設定します（ウィンドウ 0 に設定します）<br>• ADCSR.TRGE の設定をクリアします（トリガ入力による A/D 変換の開始設定をクリアします）<br>• ADCSR.ADST の設定をクリアします（A/D コンバータによる A/D 変換を停止します）<br>• MK1H.ADMK の割り込みマスクビットを 1 にし、割り込み発生を禁止します<br>• IF1H.ADIF の割り込み要求フラグビットを 0 にクリアします |
| 引数 | なし |
| リターン値 | なし |

### 2.6.16 r_Config_S12AD0_Interrupt

| | |
|---|---|
| 宣言 | static void __near r_Config_S12AD0_Interrupt(void) |
| 概要 | A/D コンバータの割り込み処理ハンドラ |
| 説明 | A/D 変換終了割り込み（INTAD）発生時の割り込み処理を行う。<br>INTAD 信号は、高速 DTC の起動要因として使用します。8 回のデータ転送完了後、本割り込みハンドラを実行します。<br>• A/D 割り込みの発生を禁止に設定します<br>• RAM に転送された 8 回分の A/D 変換結果の平均値を計算します<br>• DTC を許可設定します（A/D 変換終了割り込みをトリガとする DTC 起動の許可設定）<br>• 管理変数 flg_ad_update に 1 を設定します（A/D 変換結果の更新）<br>• A/D 割り込みを許可に設定します |
| 引数 | なし |
| リターン値 | なし |

## 2.6.17 R_Config_TRD0_Create

<table>
<tr><td>宣言</td><td>void R_Config_TRD0_Create(void)</td></tr>
<tr><td>概要</td><td>タイマ RDe の初期化</td></tr>
<tr><td>説明</td><td>タイマ RDe に対し以下のレジスタ設定を行います。
<table>
<tr><th>レジスタ名</th><th>設定値</th><th>内容</th></tr>
<tr><td>TRDMR</td><td>0x20</td><td>TRDGRD0 を TRDGRB0 のバッファ・レジスタとして使用</td></tr>
<tr><td>TRDPMR</td><td>0x05</td><td>TRDIOB0 端子は PWM 機能で使用</td></tr>
<tr><td>TRDFCR</td><td>0xA0</td><td>拡張 PWM モードを選択</td></tr>
<tr><td>TRDCR0</td><td>0x20</td><td>カウント・ソース（fTRD）を fCLK の分周なしに設定</td></tr>
<tr><td>TRDEMR0</td><td>0x20</td><td>DTENB=1（ディザリング機能有効）、GTENB=0（ゲート機能無効）、CCLV[1:0]=00B（カウンタ・リスタート機能無効）</td></tr>
<tr><td>TRDGRA0</td><td>0x018F</td><td>PWM 周期 = 5μs（200kHz）</td></tr>
<tr><td>TRDGRB0</td><td>0x0027</td><td>PWM duty = 92%（500 – 40）</td></tr>
<tr><td>TRDGRC0</td><td>0x018F</td><td>TRDGRA0 と同じ値を設定</td></tr>
<tr><td>TRDGRD0</td><td>0x0027</td><td>TRDGRB0 と同じ値を設定（バッファ・レジスタ）</td></tr>
<tr><td>PWMDLY0</td><td>0x0000</td><td>PWM 出力遅延なし</td></tr>
<tr><td>TRDOER1</td><td>0xFD</td><td>EB0=0（TRDIOB0 端子を出力許可）</td></tr>
<tr><td>TRDOCR</td><td>0x00</td><td>TOB0=0（TRDIOB0 端子の初期出力レベル設定）</td></tr>
<tr><td>TRDPOCR0</td><td>0x01</td><td>POLB=1（TRDIOB0 端子の出力レベルを H アクティブに設定）</td></tr>
<tr><td>TRDDNR0</td><td>0x00</td><td>ディザリング回数を設定</td></tr>
<tr><td>TRDOER2</td><td>0x00</td><td>強制遮断機能禁止</td></tr>
<tr><td>TRDELC</td><td>0x00</td><td>強制遮断機能禁止</td></tr>
<tr><td>TRDIER0</td><td>0x00</td><td>割り込み禁止（INTTRD0）</td></tr>
</table>
上記設定後、TRDIOB0 端子のポート設定を行います。<br>
• POM1.POM11 を通常出力モードに設定します（POM11 = 0）<br>
• P1.P11 の初期出力を 0 に設定します（P11 = 0）<br>
• PM1.PM11 を出力モードに設定します（PM11 = 0）<br>
最後に制御ソフトウェア固有設定のため、R_Config_TRD0_Create_UserInit()をコールします。</td></tr>
<tr><td>引数</td><td>なし</td></tr>
<tr><td>リターン値</td><td>なし</td></tr>
</table>

### 2.6.18 R_Config_TRD0_Create_UserInit

| | |
|---|---|
| 宣言 | void R_Config_TRD0_Create_UserInit(void) |
| 概要 | タイマ RDe の初期化（制御ソフトウェアの固有設定） |
| 説明 | タイマ RD0 の開始カウント値と、ディザリング回数の初期値を設定します。<br>レジスタ名 / 設定値 / 内容<br>TRD0 / 0x0000 / タイマ RD0 の初期カウント値を設定します<br>TRDDNR0 / ユーザ設定 / led_ch1_pid_param.dither_count の設定値を使用します |
| 引数 | なし |
| リターン値 | なし |

### 2.6.19 R_Config_TRD0_Start

| | |
|---|---|
| 宣言 | void R_Config_TRD0_Start(void) |
| 概要 | タイマ RDe のカウント開始 |
| 説明 | タイマ RD0 のカウントを開始します。<br>• TRDSTR.CSEL0 ビットに 1 を設定します（TRDGRA0 レジスタとのコンペアー致後もタイマ RD0 のカウントを継続します）<br>• TRDSTR.TSTART0 ビットに 1 を設定します（タイマ RD0 のカウントを開始します） |
| 引数 | なし |
| リターン値 | なし |

### 2.6.20 R_Config_TRD0_Stop

| | |
|---|---|
| 宣言 | void R_Config_TRD0_Stop(void) |
| 概要 | タイマ RDe のカウント動作終了 |
| 説明 | タイマ RD0 のカウントを停止します。<br>• TRDSTR.CSEL0 ビットに 0 を設定します（TRDGRA0 レジスタとのコンペアー致でタイマ RD0 のカウントを停止します）<br>• TRDSTR.TSTART0 ビットに 0 を設定します<br>• MK0H.TRDMK0 の割り込みマスクビットをクリアします（割り込み処理を禁止します）<br>• IF0H.TRDIF0 の割り込み要求フラグビットをクリアします（割り込みフラグをクリアします）<br>• TRDSR0 レジスタの読み出しを行います<br>• TRDSR0 レジスタに 00H を書き込みます（割り込み通知を消去します） |
| 引数 | なし |
| リターン値 | なし |

### 2.6.21 R_Config_PORT_Create

| | |
|---|---|
| 宣言 | void R_Config_PORT_Create(void) |
| 概要 | I/O ポートの初期化 |
| 説明 | RL78/F2x の I/O ポートの初期化を行います。<br>タイマ RDe、A/D コンバータが制御するポートについては、以下の関数で設定します。<br>• タイマ RDe : 2.6.17 R_Config_TRD0_Create<br>• A/D コンバータ : 2.6.12 R_Config_S12AD0_Create |
| 引数 | なし |
| リターン値 | なし |

### 2.6.22 R_Config_TRD0_Set_TRD0_ReloadTrigger

| | |
|---|---|
| 宣言 | MD_STATUS R_Config_TRD0_Set_TRD0_ReloadTrigger(st_extpwm_buffer_registers_t * buffer) |
| 概要 | タイマ RDe の設定更新 |
| 説明 | タイマ RDe のバッファ・レジスタの設定を行い、コンペア・レジスタのリロードを要求します。<br>また、TRDDNR0 レジスタ設定を行い、ディザリング回数の書き替えも要求します。<br>• TRDGRD0 レジスタ（TRDGRB0 レジスタのバッファ・レジスタ）<br>• TRDDNR0 レジスタ（タイマ RD0 のディザリング・ナンバ・レジスタ）<br>TRDRDT0.RDT ビットを 1 にセットし、レジスタのリロードを要求します。<br><br>[本関数の使用例]<br>引数*buffer が使用する st_extpwm_buffer_registers_t の型定義は以下の通りです。<br>`typedef struct {`<br>`  uint16_t trdgrcn;  // TRDGRCn レジスタへの設定値（n=0,1）`<br>`  uint16_t trdgrdn;  // TRDGRDn レジスタへの設定値（n=0,1）`<br>`  uint16_t trdcmpdn;  // TRDCMPDn レジスタへの設定値（n=0,1）`<br>`  uint8_t trddnrn;  // TRDDNRn レジスタへの設定値（n=0,1）`<br>`} st_extpwm_buffer_registers_t;`<br><br>引数 *buffer にリロードで設定したいレジスタ値を登録します。<br>登録後、本関数を実行します。<br>`// set next Timer compare count`<br>`buffer.trdgrcn = 0x0000;`<br>`buffer.trdgrdn = 0x018F;`<br>`buffer.trdcmpdn = 0x0000;`<br>`// set next dithering count`<br>`buffer->trddnrn = 0x0A;`<br><br>`// Timer reload request`<br>`R_Config_TRD0_Set_TRD0_ReloadTrigger(&buffer);` |
| 引数 | st_extpwm_buffer_registers_t * buffer |
| リターン値 | MD_STATUS |

## 2.7 フローチャート

### 2.7.1 起動シーケンス

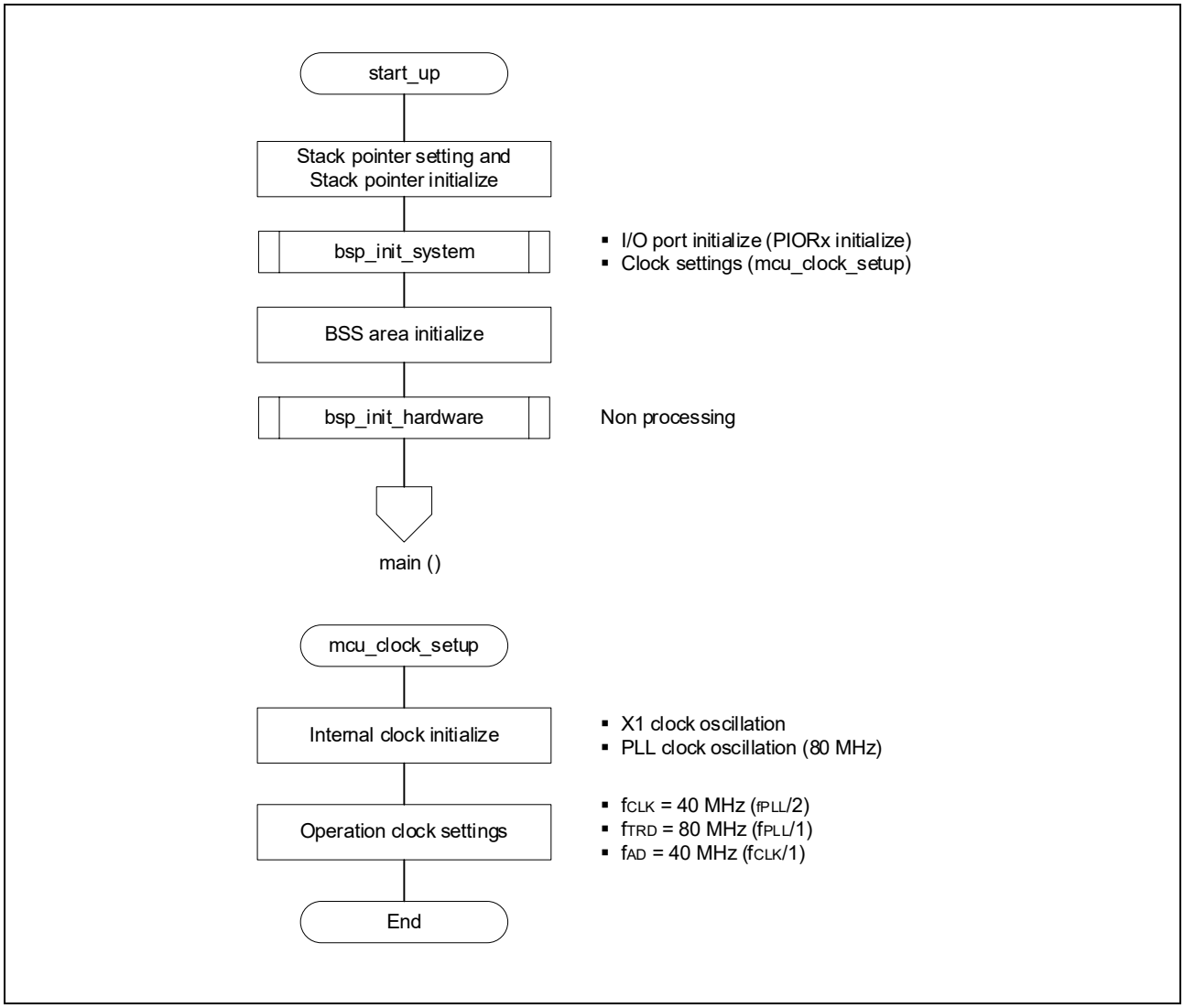



**図 2-4 起動シーケンス**

### 2.7.2 システム制御フローチャート

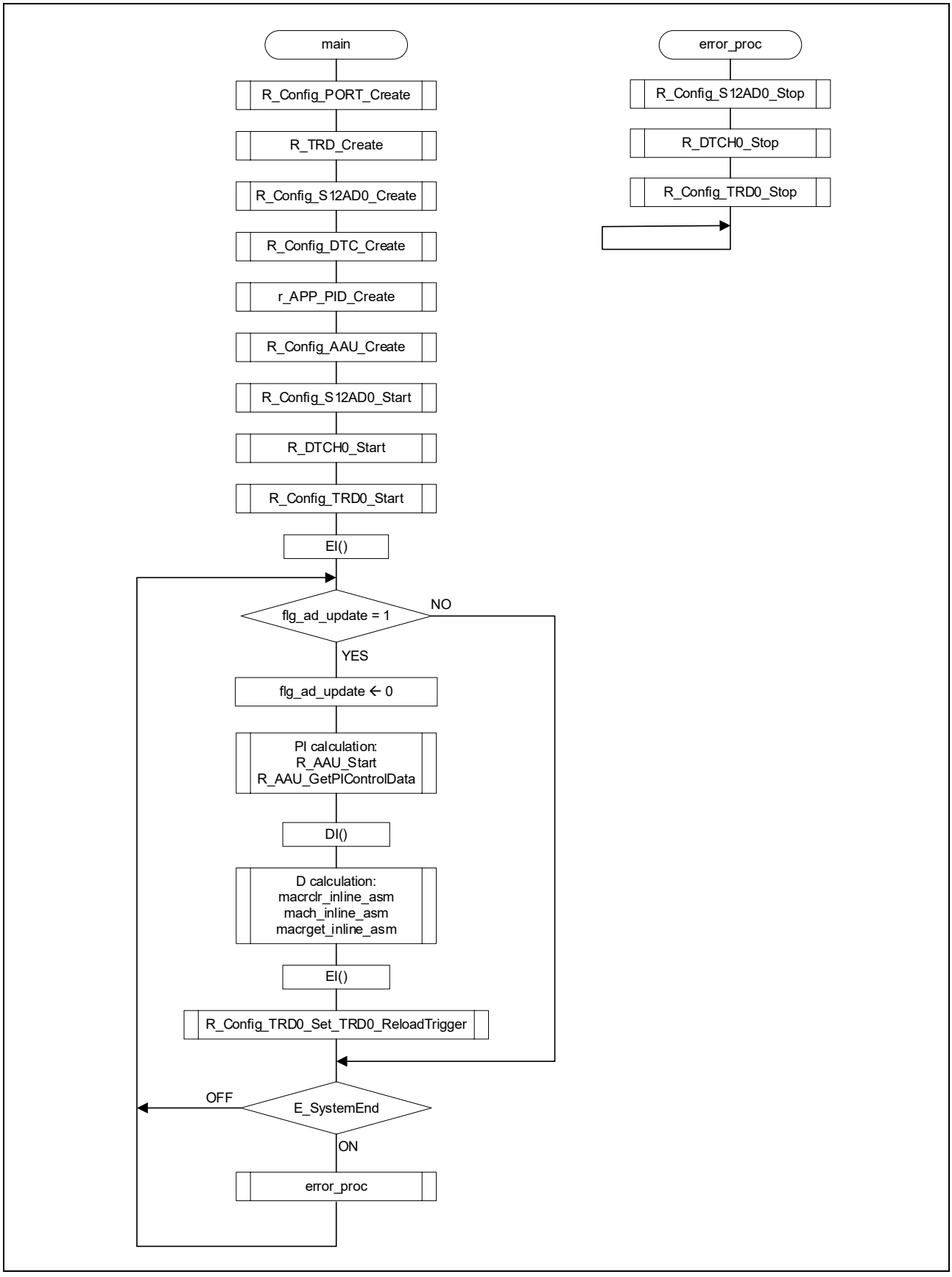



**図 2-5 システム制御フローチャート**

### 2.7.3 ポート初期設定

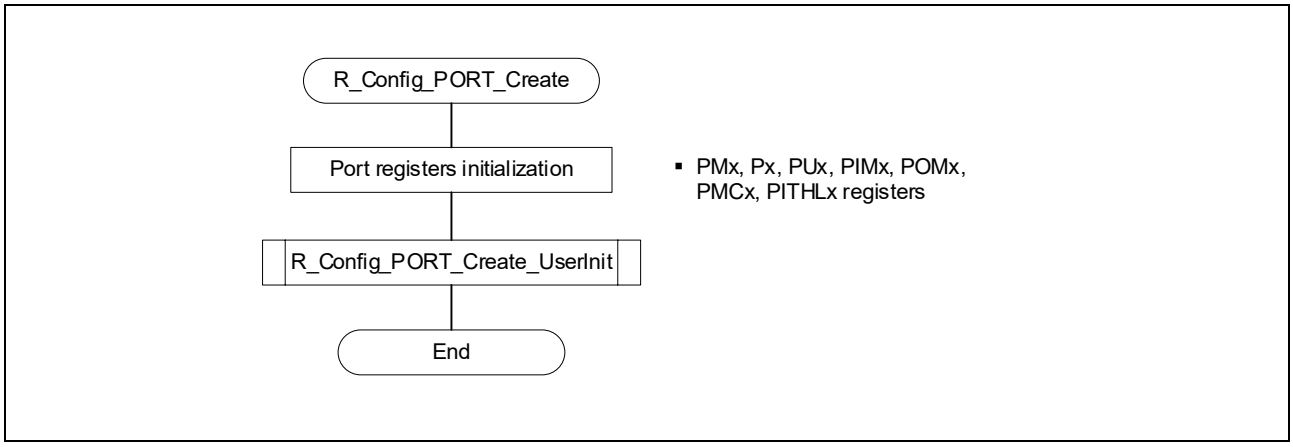



**図 2-6 汎用ポート初期化フローチャート**

タイマ RDe、A/D コンバータが使用する端子の初期化は、タイマ RDe、および A/D コンバータの初期化フローチャートを参照して下さい。

### 2.7.4 タイマ初期設定

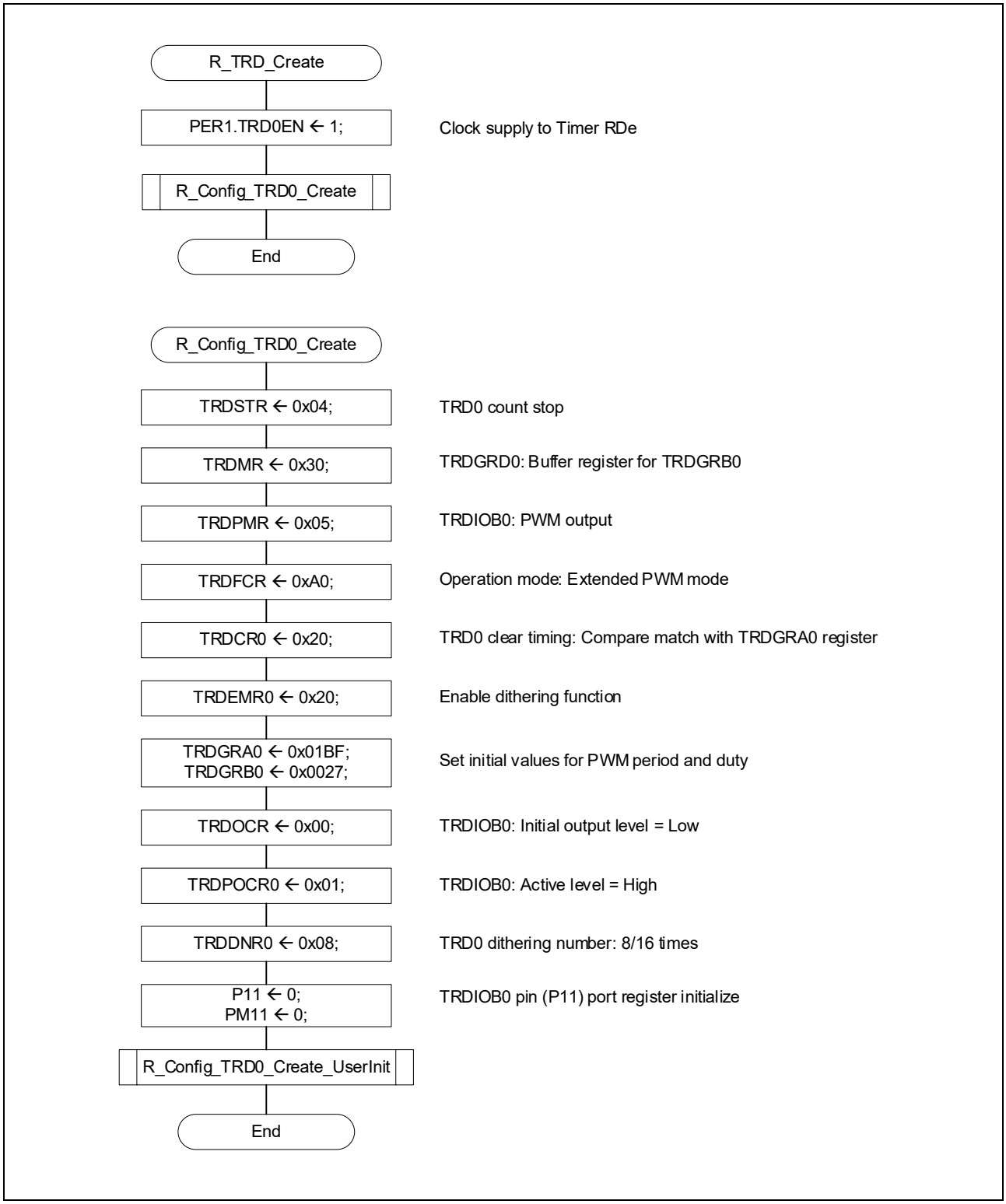



**図 2-7 タイマ初期化フローチャート**

### 2.7.5 A/D コンバータ初期設定

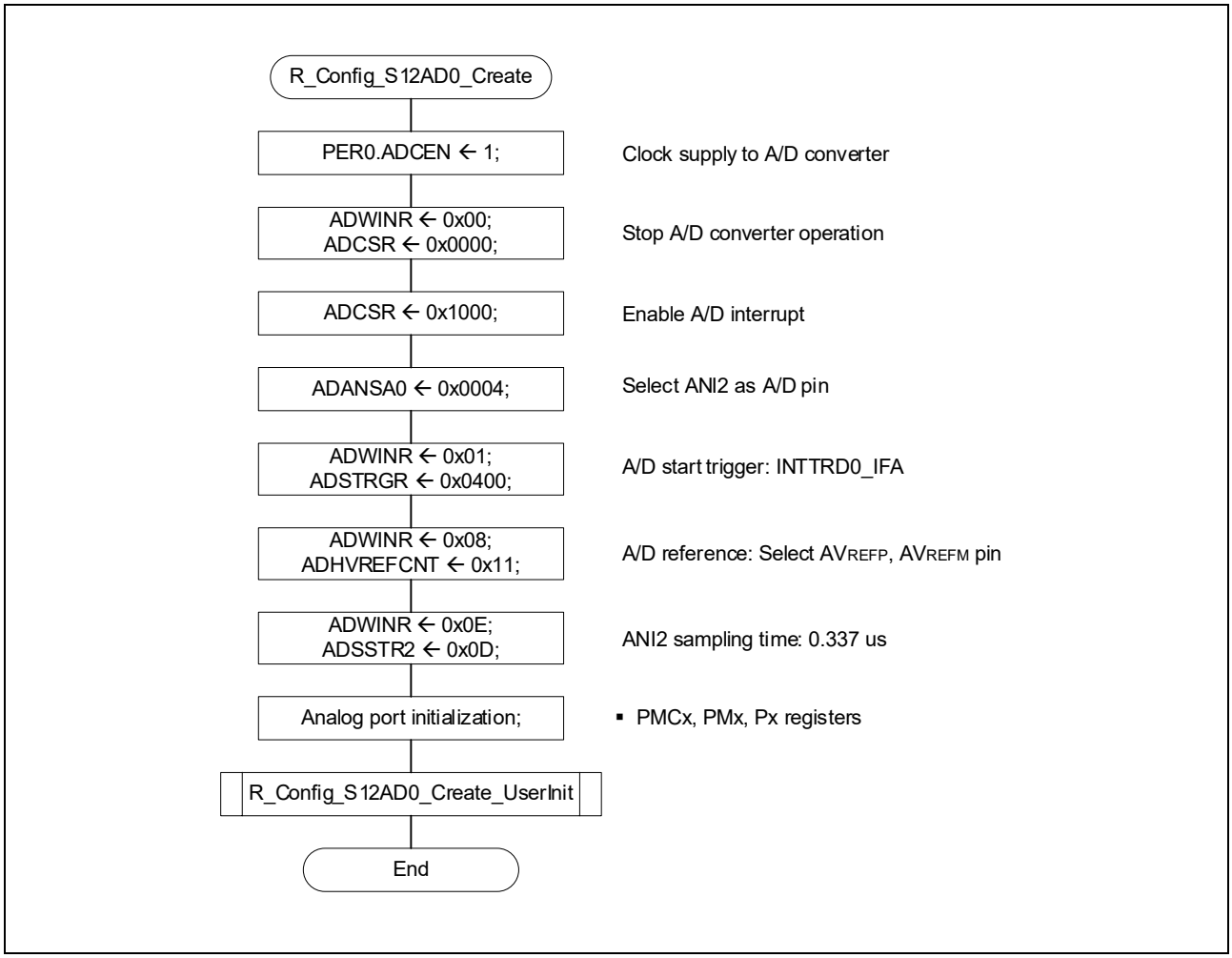



**図 2-8 A/D コンバータ初期化フローチャート**

### 2.7.6 DTC 初期設定

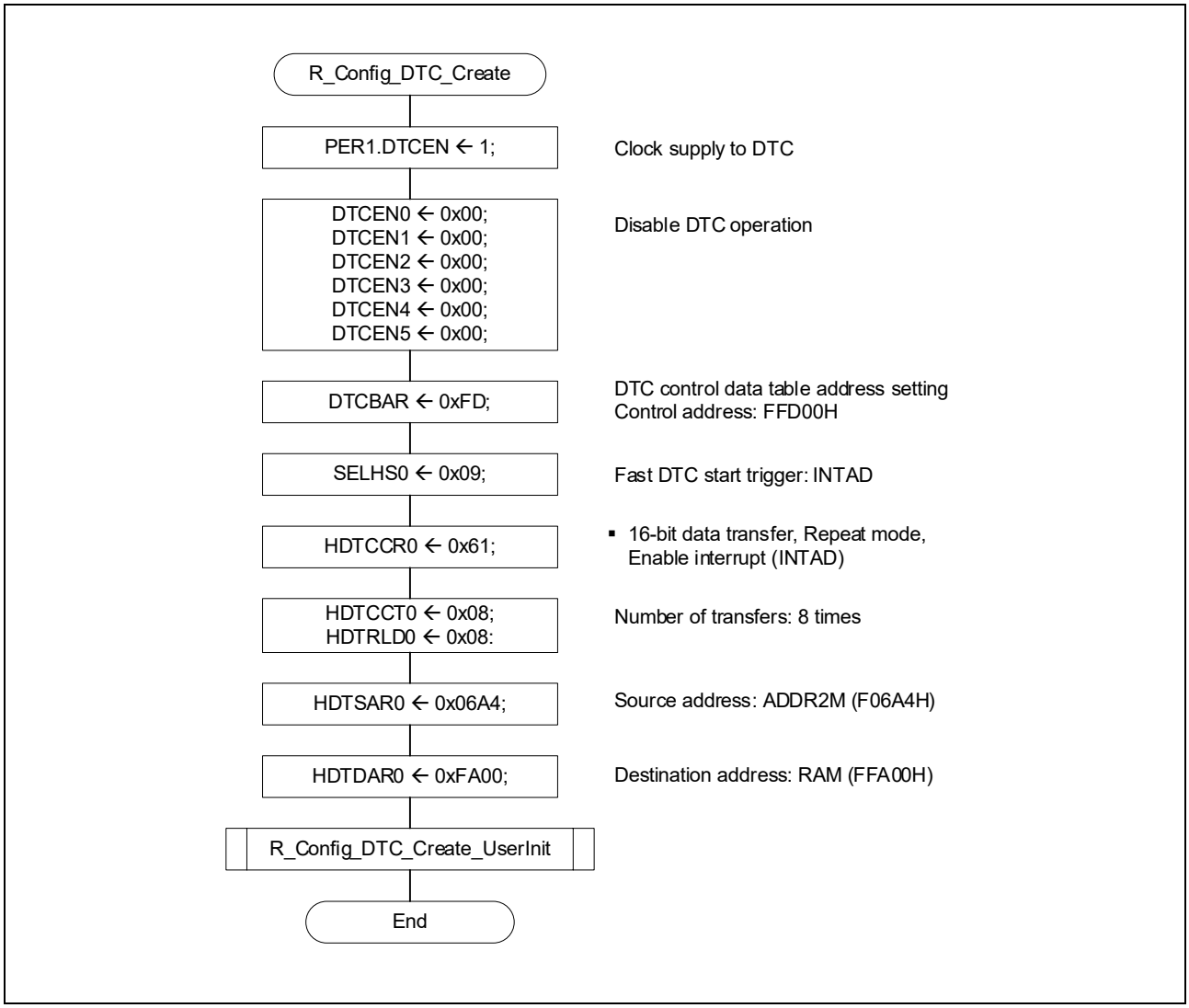



**図 2-9 DTC 初期化フローチャート**

### 2.7.7 AAU 初期設定

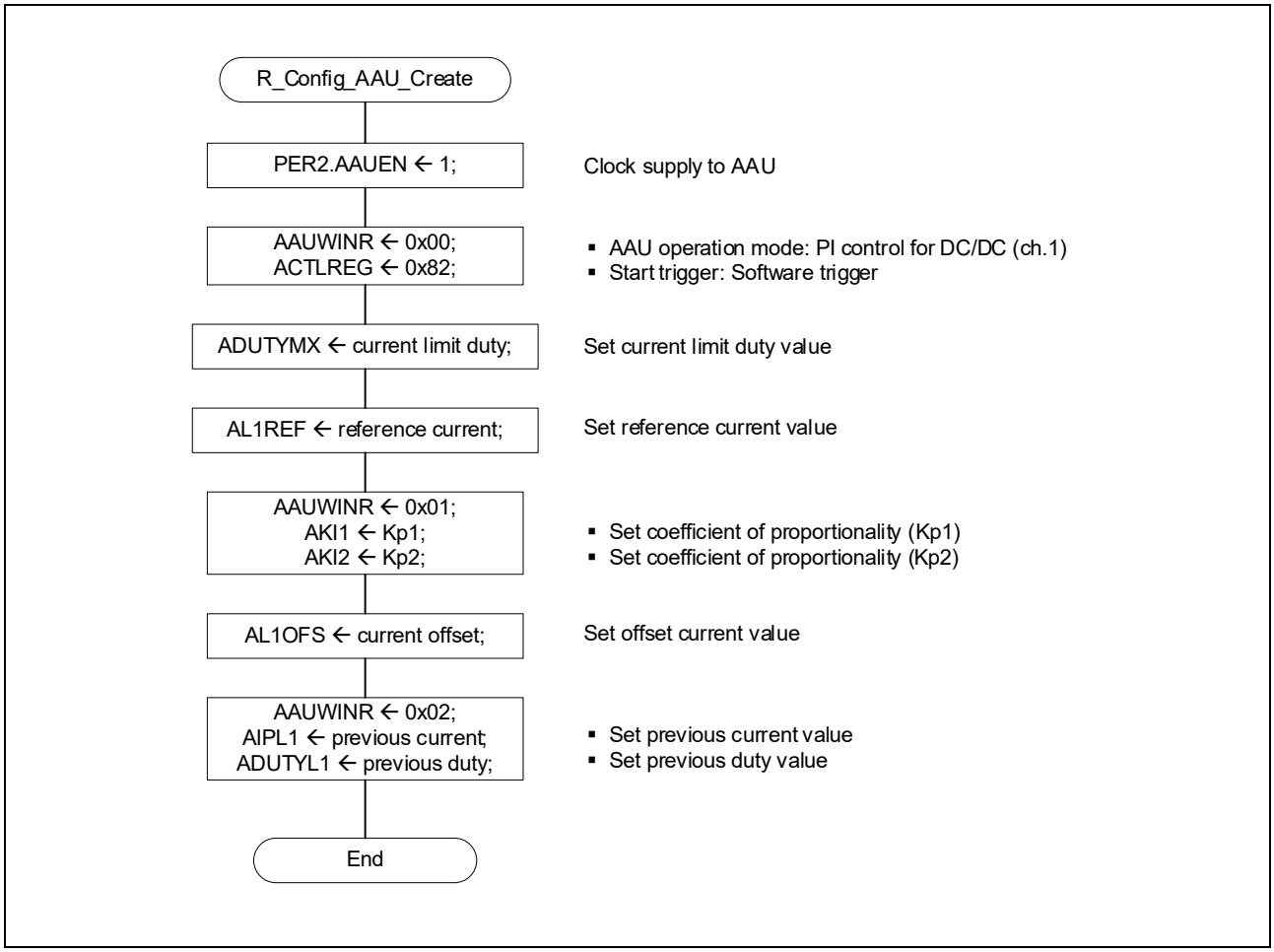



**図 2-10 AAU 初期化フローチャート**

### 2.7.8 タイマ制御

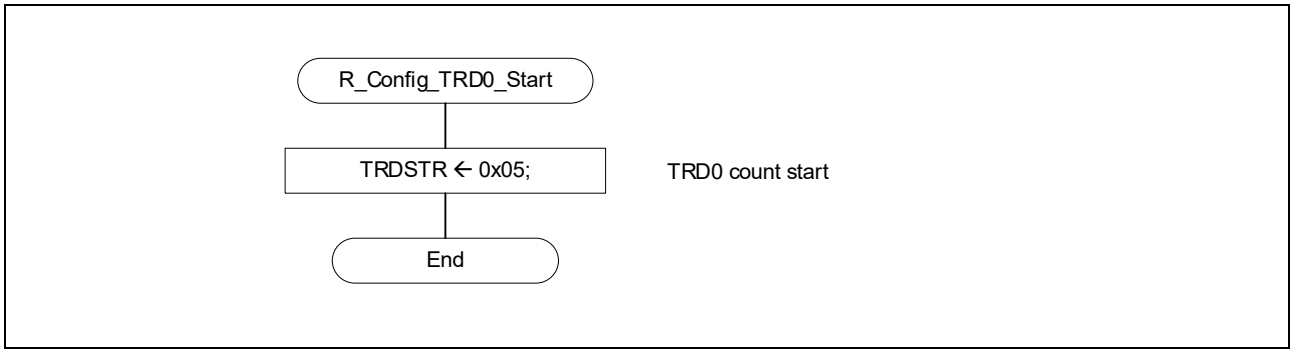



**図 2-11 タイマ開始フローチャート**

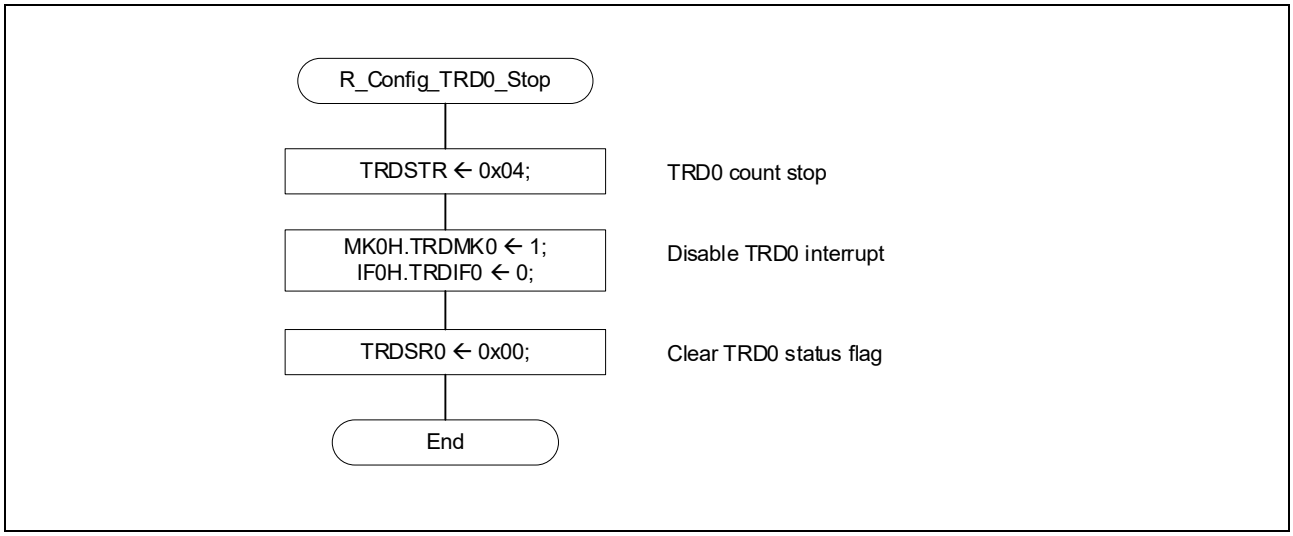



**図 2-12 タイマ停止フローチャート**

### 2.7.9 A/D コンバータ制御

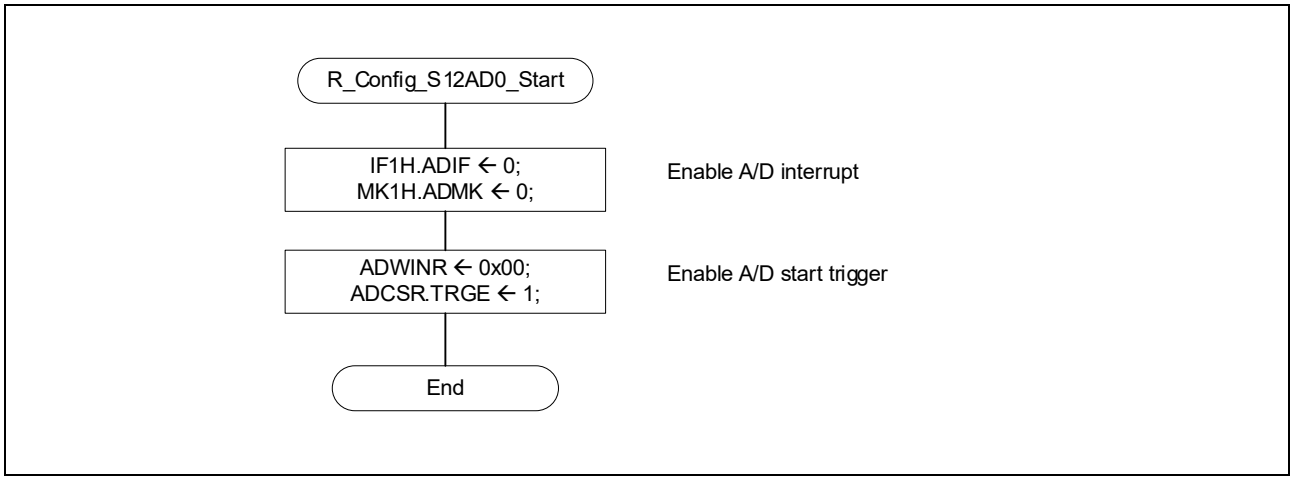



**図 2-13 A/D コンバータ開始フローチャート**

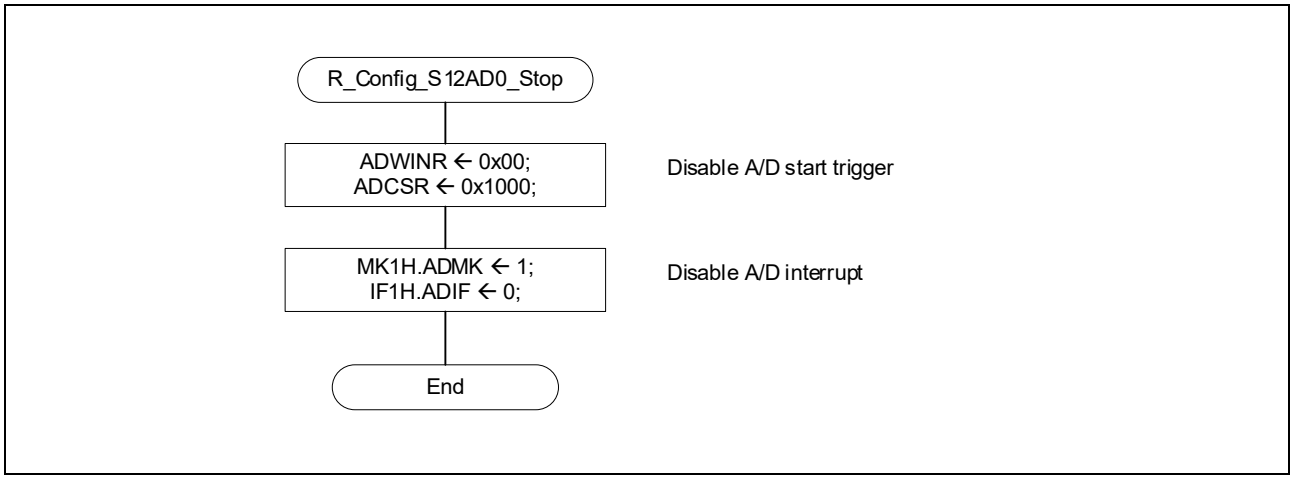



**図 2-14 A/D コンバータ停止フローチャート**

### 2.7.10 DTC 制御

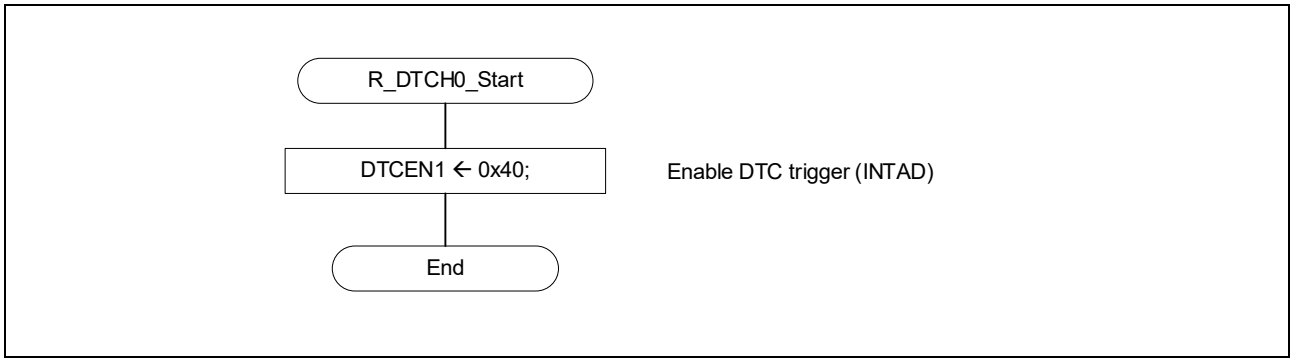



**図 2-15 DTC 開始フローチャート**

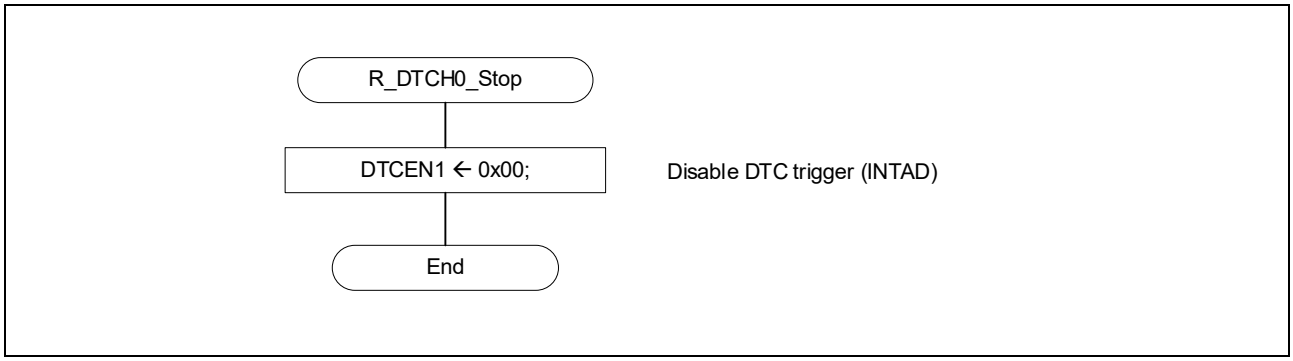



**図 2-16 DTC 終了フローチャート**

### 2.7.11 AAU 制御

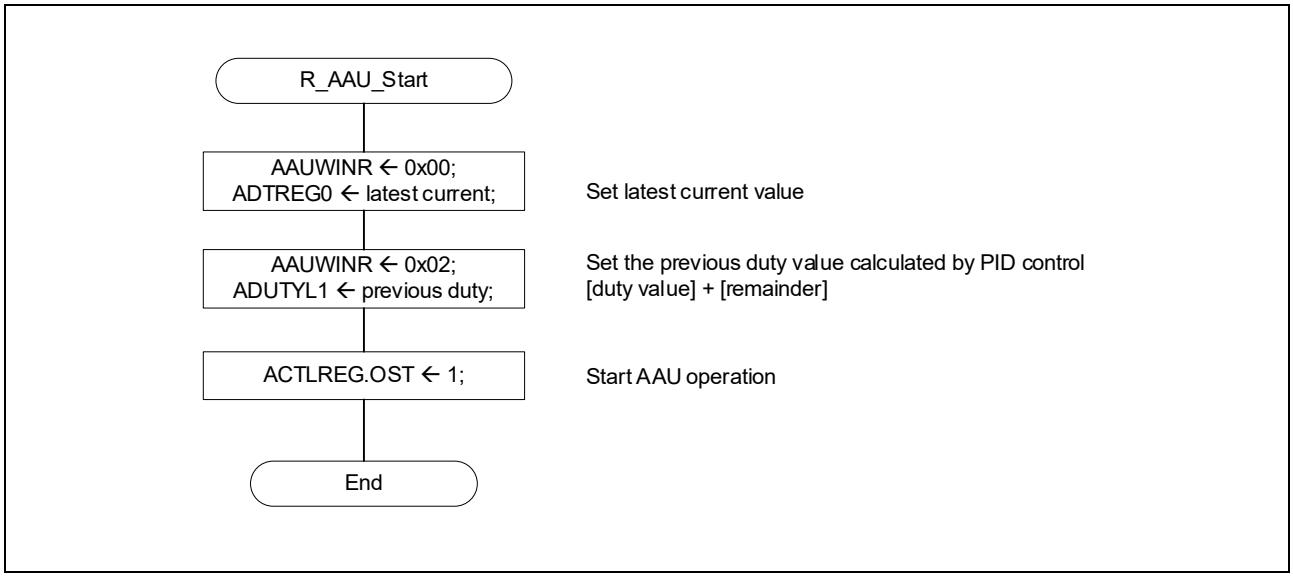



**図 2-17 AAU 演算開始フローチャート**

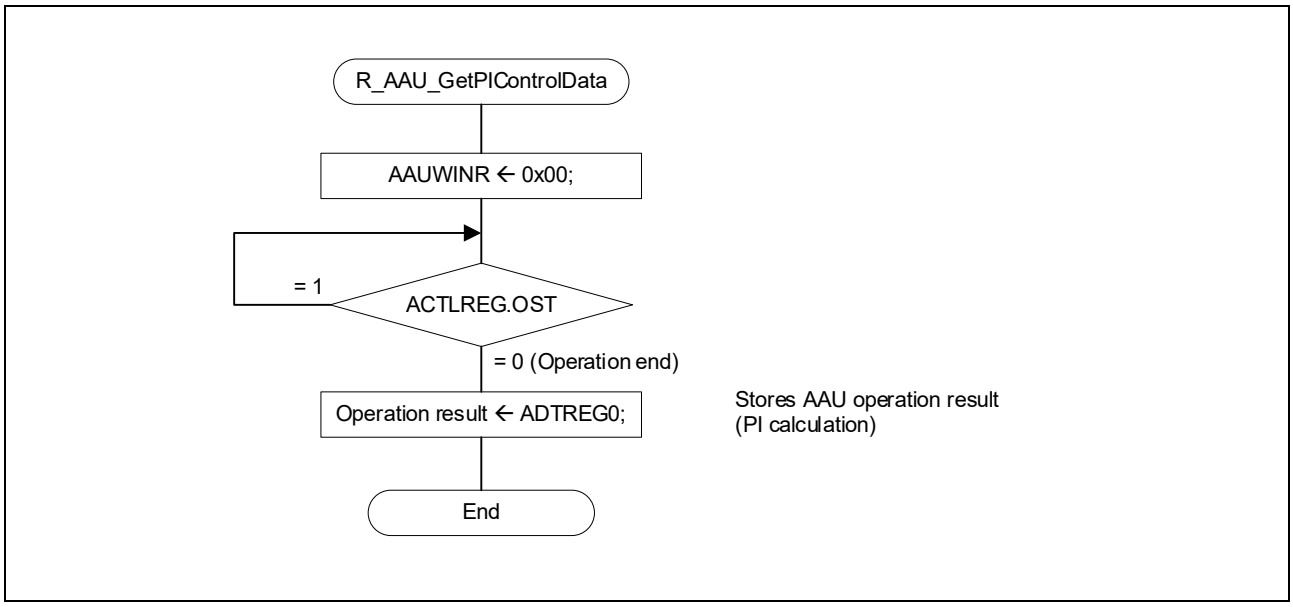



**図 2-18 AAU 演算結果読み出しフローチャート**

## 3. 参考資料

本アプリケーションノートにおける参考資料を以下に示します。参照の際は、ルネサスエレクトロニクスホームページから最新版を入手してください。

・RL78/F23, F24 ユーザーズマニュアル ハードウェア編 Rev. 1.00

・RL78 ファミリユーザーズマニュアル ソフトウェア編 Rev. 2.30

## 改訂記録

| Rev. | 発行日 | 改訂内容 | |
|---|---|---|---|
| | | ページ | ポイント |
| 1.00 | 2023.07.31 | - | 初版発行 |
| | | | |

# 製品ご使用上の注意事項

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

1. 静電気対策
   CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。
2. 電源投入時の処置
   電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。
3. 電源オフ時における入力信号
   当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。
4. 未使用端子の処理
   未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。
5. クロックについて
   リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。
6. 入力端子の印加波形
   入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、$V_{IL}$（Max.）から $V_{IH}$（Min.）までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、$V_{IL}$（Max.）から $V_{IH}$（Min.）までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。
7. リザーブアドレス（予約領域）のアクセス禁止
   リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。
8. 製品間の相違について
   型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ幅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.