

## RL78/G24

### CPU と FAA による A/D コンバータの共有

---

#### 要旨

本アプリケーションノートでは、RL78/G24 の CPU とフレキシブル・アプリケーション・アクセラレータ（FAA）の機能および、CPU と FAA で A/D コンバータを共有する方法を説明します。

#### 動作確認デバイス

RL78/G24

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. FAA の仕様	3
1.1 機能説明	3
1.2 周辺機能の割り当て	5
2. アプリケーション仕様	6
2.1 仕様概要	6
2.2 動作説明	8
3. 動作確認条件	9
4. ハードウェア説明	10
4.1 ハードウェア構成例	10
4.2 使用端子一覧	10
5. ソフトウェア説明	11
5.1 スマート・コンフィグレータの設定	11
5.1.1 システム設定	11
5.1.2 コンポーネントの設定	13
5.2 フォルダ構成	20
5.3 オプション・バイトの設定一覧	22
5.4 定数一覧	22
5.5 変数一覧	23
5.6 関数一覧	24
5.7 関数仕様	24
5.8 フローチャート	27
5.8.1 メイン処理	27
5.8.2 r_faa_adcontrol_start 関数	28
5.8.3 r_faa_adcontrol_stop 関数	28
5.8.4 r_faa_adcontrol_requestadc 関数	29
5.8.5 r_faa_adcontrol_getadc 関数	29
5.8.6 FAA 処理 : _P_ADControl	30
5.8.7 FAA 処理 : _P_ADControl_TAU0_Interrupt	30
5.8.8 FAA 処理 : _P_ADControl_GetAd	32
6. サンプルコード	33
7. 参考ドキュメント	33
改訂記録	34

## 1. FAA の仕様

### 1.1 機能説明

フレキシブル・アプリケーション・アクセラレータ (FAA) は、ハーバード・アーキテクチャを採用したルネサスエレクトロニクス株式会社のオリジナルのアプリケーションアクセラレータで、32 ビット乗算、加算、減算を 1 サイクルで実行します。また、FAA から一部の周辺機能に直接アクセスすることが可能であり、CPU と FAA を組み合わせた動作も可能なことから、システムの特性を向上させることができます。

FAA のオプション設定やツールの操作、デバッグ方法等は、「フレキシブル・アプリケーション・アクセラレータ (FAA) ツールガイド CS+ 編 (e<sup>2</sup> studio 編)」を参照してください。

図 1-1 に FAA のイメージ、表 1-1 に FAA の機能を示します。

図 1-1 FAA のイメージ

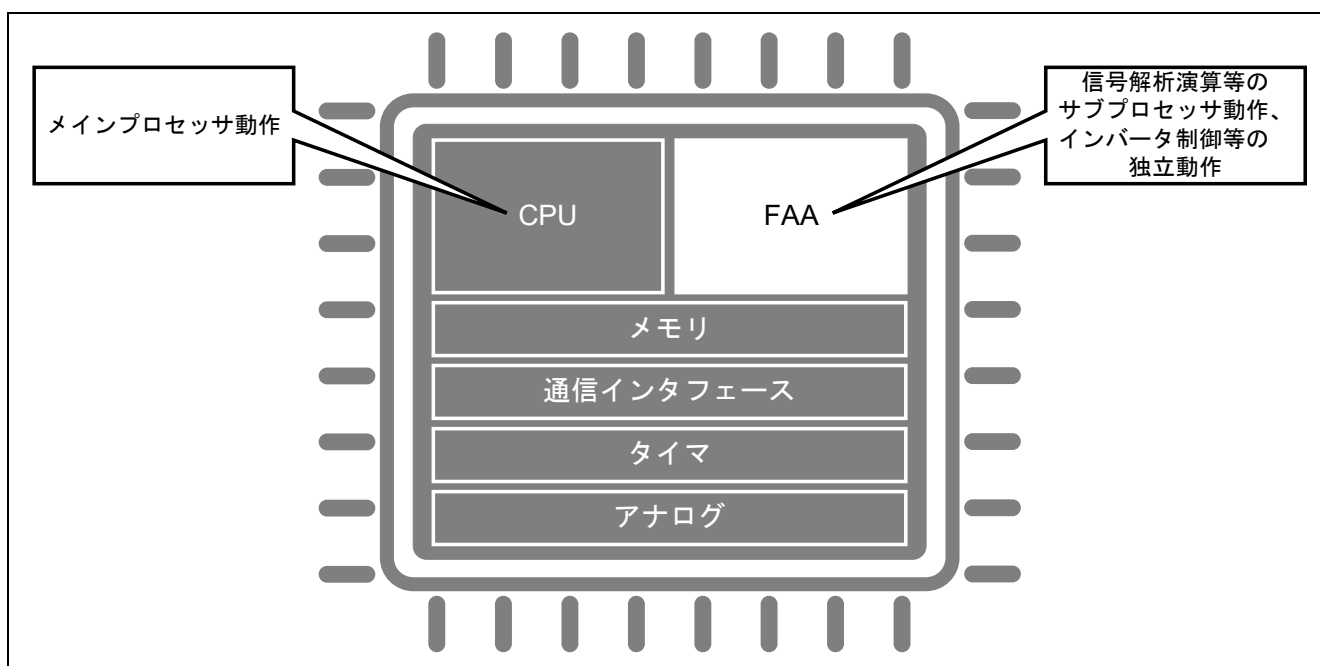


表 1-1 FAA の機能

項目	機能
プロセッサ	<p>演算命令は、1 命令/1 クロックで実行 (連続して演算命令を実行する場合、2 つ目以降の命令は、2 サイクルで実行)</p> <ul style="list-style-type: none"> <li>乗算：符号付き 32 ビット×符号付き 32 ビット→符号付き 32 ビット 乗算結果 (64 ビット) を任意のビット数分右シフト可能</li> <li>加算：符号付き 32 ビット+符号付き 32 ビット→符号付き 32 ビット 内部的に 33 ビット精度で演算実行</li> <li>減算：符号付き 32 ビット-符号付き 32 ビット→符号付き 32 ビット 内部的に 33 ビット精度で演算実行</li> <li>リミット演算：上限/下限の設定可能演算パラメータ・レジスタ (33 ビット×4)</li> </ul> <p>プロセッサ内部レジスタ</p> <ul style="list-style-type: none"> <li>演算パラメータ・レジスタ (32 ビット×6)</li> <li>アドレス・ポインタ・レジスタ (12 ビット×6)</li> <li>フラグ・ビット・レジスタ (16 ビット×1 (うち、有効フラグビットは 4 ビット))</li> </ul>
メモリ	<ul style="list-style-type: none"> <li>インストラクション・コード・メモリ：4KB</li> <li>データ・メモリ：2KB</li> <li>データ共有メモリ (SHDMEM)：32 バイト</li> </ul>
割り込み	<ul style="list-style-type: none"> <li>多重割り込み可能</li> <li>割り込み要因 入力イベント検出割り込み：10 タイミング・コンペア・マッチ割り込み：6</li> </ul>
入力イベント処理	<ul style="list-style-type: none"> <li>入力チャンネル数：10</li> <li>検出エッジ：立ち上がり、立ち下がり、両エッジ</li> </ul>
タイミング処理	<ul style="list-style-type: none"> <li>基準タイミング・カウンタのビット数：24 ビット</li> <li>コンペア・マッチのチャンネル数：6</li> </ul>

## 1.2 周辺機能の割り当て

RL78/G24 の下記周辺機能は、FAA から制御することが可能です。

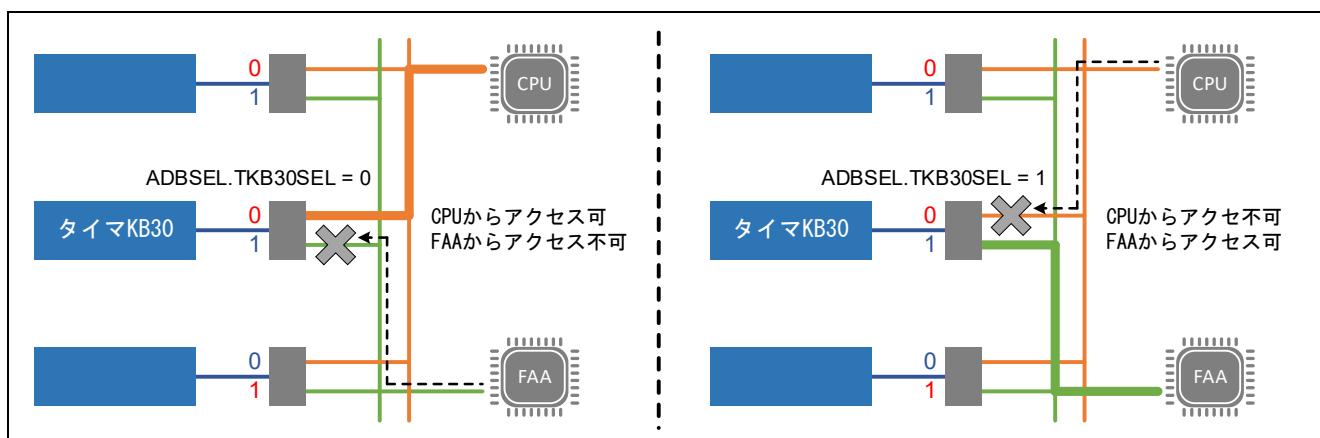
- A/D コンバータ
- D/A コンバータ
- タイマ RD2/PWM オプション・ユニット A (PWMOPA)
- タイマ RG2
- タイマ RX
- 16 ビット・タイマ KB30, KB31, KB32
- コンパレータ
- プログラマブル・ゲイン・アンプ
- シリアル・アレイ・ユニット
- タイマ・アレイ・ユニット
- ポート機能 (ポート 1)
- セキュリティ機能 (TRNG)

上記の周辺機能は、アドレス・バス選択レジスタ (ADBSEL) で、CPU か FAA のどちらに割り当てるかを選択できます。各周辺機能に対応するバス選択ビット (xxSEL) に「0」を設定すると、CPU からのバスアクセスを許可し、「1」を設定すると、FAA からのバスアクセスを許可します。CPU に割り当てられた周辺機能は、FAA からアクセスできません。逆も同様です。

アドレス・バス選択レジスタ (ADBSEL) については、「RL78/G24 ユーザーズマニュアル ハードウェア編 (R01UH0961)」を参照してください。

図 1-2 にアドレス・バス機能選択のイメージを示します。また、本アプリケーションノートのサンプルコードのアドレス・バス選択レジスタ (ADBSEL) の割り当ては図 2-1 を確認してください。

図 1-2 アドレス・バス選択機能のイメージ



## 2. アプリケーション仕様

### 2.1 仕様概要

本アプリケーションノートでは、CPU と FAA の 2 つのプロセッサを使用し、それぞれのプロセッサでアナログ入力の A/D 変換、変換結果の出力を行います。

A/D コンバータは FAA に割り当てます。CPU が A/D 変換を行う際は、FAA が CPU の代わりに A/D コンバータを制御します。変換結果に応じて、CPU は TKB31、FAA は TKB30 から出力する PWM のデューティ比を変更します。

CPU プログラムで、A/D コンバータ、TKB30、TKB31、タイマ・アレイ・ユニット (TAU0) の初期設定を行い、TAU チャンネル 0 カウント完了割り込み (INTTM00) で一定周期 (10 $\mu$ s) のタイミングごとに、FAA の動作を開始します。また、FAA へ A/D 変換のリクエストを行い、変換完了後、変換結果を取得し、TKB31 のデューティ比を変更します。

FAA プログラムでは、A/D コンバータと TKB30 の制御を行います。INTTM00 の発生で動作を開始し、P22 / ANI2 端子の A/D 変換を行い、TKB30 のデューティ比を変更後、動作を停止します。CPU からの A/D 変換リクエストがあった場合は、P23 / ANI3 端子の A/D 変換も行き、変換結果を SHDMEM に格納後、動作を停止します。

表 2-1 に使用する周辺機能と用途を示します。

表 2-1 使用する周辺機能と用途

周辺機能	用途
フレキシブル・アプリケーション・アクセラレータ (FAA)	A/D コンバータと TKB30 の制御を行い、TKBO00 端子から出力する PWM 出力のデューティ比を変更する
A/D コンバータ (アドバンスド・モード ON)	P22 / ANI2 端子、P23 / ANI3 端子のアナログ入力電圧を A/D 変換する
16 ビット・タイマ KB30 (TKB30)	TKBO00 端子から PWM 出力する
16 ビット・タイマ KB31 (TKB31)	TKBO10 端子から PWM 出力する
タイマ・アレイ・ユニット (TAU0)	周期 10 $\mu$ s ごとにカウント完了割り込み (INTTM00) を発生する
イベント・リンク・コントローラ (ELC)	TAU チャンネル 0 カウント完了割り込み (INTTM00) を FAA の入力イベントにリンクする

図 2-1 にアドレス・バス選択レジスタ（ADBSEL）の割り当てを示します。

図 2-1 アドレス・バス選択レジスタ（ADBSEL）の割り当て

	15	14	13	12	11	10	9	8
ADBSEL	FAADIVSEL	0	TRNGSEL	PORTSEL	TKB32SEL	TKB31SEL	TKB30SEL	TRGSEL
設定値	0	0	0	0	0	0	1	0
	7	6	5	4	3	2	1	0
	TRD0SEL	PWMOPSEL	TRXSEL	DACSEL	PGACMPSEL	ADCSEL	SAU0SEL	TAU0SEL
	0	0	0	0	0	1	0	0

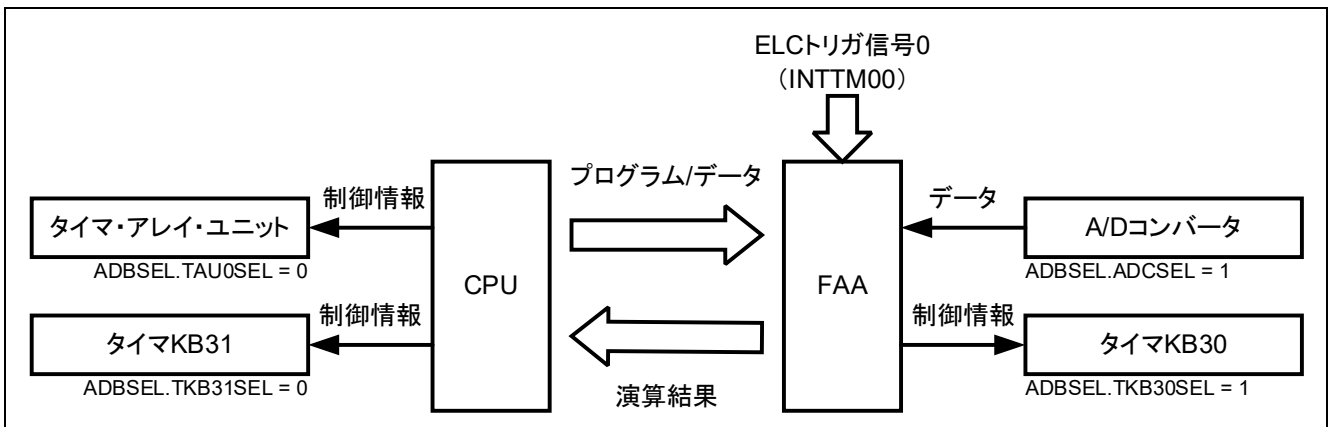
  

0	CPUからのバスアクセスを許可
1	FAAからのバスアクセスを許可

TKB31SEL : 16 ビット・タイマ KB31  
 TKB30SEL : 16 ビット・タイマ KB30  
 ADCSEL : A/D コンバータ  
 TAU0SEL : タイマ・アレイ・ユニット

図 2-2 にプロセッサ動作のイメージを示します。

図 2-2 プロセッサ動作のイメージ



## 2.2 動作説明

本サンプルコードでは、16ビット・タイマ KTB30 (TKB30) と 16ビット・タイマ KTB31 (TKB31) を単体動作モード (TKBCR00 レジスタによる周期制御) で使用し、P12 / TKBO00 端子と P14 / TKBO10 端子から PWM 出力を行います。PWM 出力のデューティ比は A/D コンバータによる変換結果を用います。

TAU0 の周期を 10 $\mu$ s とし、チャンネル 0 カウント完了ごとに発生する割り込み (INTTM00) で CPU から FAA を起動し、FAA で A/D 変換と TKB30 の PWM 出力のデューティ比を変更します。

1. [CPU プログラム] A/D コンバータ、TKB30、TKB31 の動作を許可します。
2. [CPU プログラム] A/D コンバータと TKB30 の SFR アクセスを FAA バスに設定します。
3. [CPU プログラム] TAU0 の動作を開始し、CPU 側の A/D 変換を FAA にリクエストします。
4. [CPU プログラム] タイマ動作開始後、10 $\mu$ s ごとに TAU チャンネル 0 カウント完了割り込み (INTTM00) が発生し、FAA ヘックロック供給を許可し、FAA の動作を許可します。
5. [CPU プログラム] FAA のスタックポインタ、FAA プログラムの開始アドレスを設定し、FAA の動作を開始させます。
6. [FAA プログラム] P22 / ANI2 端子の A/D 変換を行い、変換結果によってコンペア・レジスタ (TKBCR01) を更新して TKBO00 出力のデューティ比を変更します。
7. [FAA プログラム] CPU から A/D 変換のリクエストがあった場合、P23 / ANI3 端子の A/D 変換を行い、SHDMEM に格納後、FAA は動作を停止します。
8. [CPU プログラム] FAA のプログラム実行が完了すると、FAA ヘックロック供給を停止し、FAA の動作を禁止します。
9. [CPU プログラム] FAA の A/D 変換が正常に完了していた場合、SHDMEM に格納された変換結果を取得し、変換結果によってコンペア・レジスタ (TKBCR11) を更新して TKBO10 出力のデューティ比を変更します。
10. [CPU プログラム] 次の A/D 変換を FAA にリクエストし、HALT モードに移行します。
11. [CPU プログラム] 4 に戻り、再び TAU チャンネル 0 カウント完了割り込み (INTTM00) の発生を待ちます。



### 3. 動作確認条件

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 3-1 動作確認条件

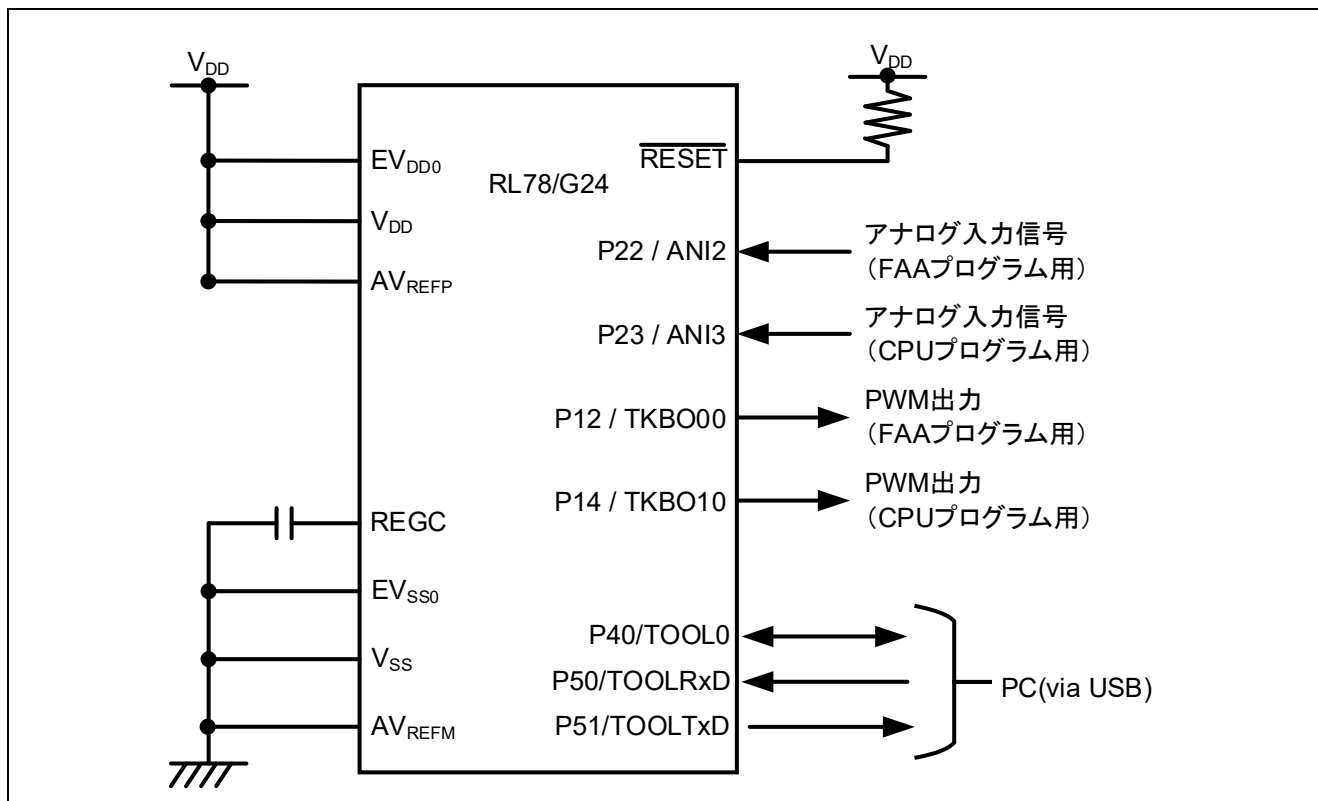
項目	内容
使用マイコン	RL78/G24 (R7F101GLG)
動作周波数	<ul style="list-style-type: none"> <li>高速オンチップ・オシレータ・クロック (f<sub>HOCO</sub>) : 8MHz</li> <li>PLL 発振回路出力 (f<sub>PLL</sub>) : 96MHz</li> <li>CPU/周辺ハードウェア・クロック (f<sub>CLK</sub>) : 48MHz</li> </ul>
動作電圧	<ul style="list-style-type: none"> <li>3.3V (2.7V~5.5V で動作可能)</li> <li>LVD0 動作 (V<sub>LVD0</sub>) : リセット・モード 立ち上がり時 TYP. 2.97V 立ち下がり時 TYP. 2.91V</li> </ul>
統合開発環境 (CS+)	ルネサスエレクトロニクス製 CS+ for CC V8.12.00
C コンパイラ (CS+)	ルネサスエレクトロニクス製 CC-RL V1.14.00
統合開発環境 (e <sup>2</sup> studio)	ルネサスエレクトロニクス製 e <sup>2</sup> studio 2024-10 (24.10.0)
C コンパイラ (e <sup>2</sup> studio)	ルネサスエレクトロニクス製 CC-RL V1.14.00
統合開発環境 (IAR)	IAR システム製
C コンパイラ (IAR)	IAR Embedded Workbench for Renesas RL78 V5.10.3
スマート・コンフィグレータ	V.1.11.0
ボードサポートパッケージ (r_bsp)	V.1.70
エミュレータ	CS+、e <sup>2</sup> studio : COM ポート IAR : E2 エミュレータ Lite
使用ボード	RL78/G24 Fast Prototyping Board (RTK7RLG240C00000BJ)

## 4. ハードウェア説明

### 4.1 ハードウェア構成例

図 4-1 に本アプリケーションノートのサンプルコードで使用するハードウェア構成例を示します。

図 4-1 ハードウェア構成例



- 注意 1. この回路イメージは接続の概要を示す為に簡略化しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください（入力専用ポートは個別に抵抗を介して  $V_{DD}$  又は  $V_{SS}$  に接続して下さい）。
- 注意 2.  $EV_{SS}$  で始まる名前の端子がある場合には  $V_{SS}$  に、 $EV_{DD}$  で始まる名前の端子がある場合には  $V_{DD}$  にそれぞれ接続してください。
- 注意 3.  $V_{DD}$  は  $LVD0$  にて設定したリセット解除電圧 ( $V_{LVD0}$ ) 以上にしてください。

### 4.2 使用端子一覧

表 4-1 に使用端子と機能を示します。

表 4-1 使用端子と機能

端子名	入出力	内容
P22 / ANI2	入力	A/D コンバータ アナログ入力 (FAA プログラム用)
P23 / ANI3	入力	A/D コンバータ アナログ入力 (CPU プログラム用)
P12 / TKBO00	出力	PWM 出力 (FAA プログラム用)
P14 / TKBO10	出力	PWM 出力 (CPU プログラム用)

注意 本アプリケーションノートは、使用端子のみを端子処理しています。実際に回路を作成される場合は、端子処理などを適切に行い、電気的特性を満たすように設計してください。

## 5. ソフトウェア説明

### 5.1 スマート・コンフィグレータの設定

本サンプルコードにおけるスマート・コンフィグレータの設定を示します。スマート・コンフィグレータの設定における各表の項目、設定内容は設定画面の表記で記載しています。

#### 5.1.1 システム設定

本サンプルコードで使用しているシステム設定を以下に示します。

なお、本サンプルコードで使用しているシステム設定は、統合開発環境 e<sup>2</sup> studio と CS+ は同じですが、IAR は異なります。ご使用の環境に合わせて適切な設定を行ってください。

まず、図 5-1 に本サンプルコード (e<sup>2</sup> studio、CS+) で使用しているシステム設定を示します。

RL78/G24 Fast Prototyping Board (RTK7RLG240C00000BJ) で COM port デバッグを行う場合、統合開発環境 (e<sup>2</sup> studio、CS+) 内の設定を適切に行う必要があります。詳細は、「RL78/G24 Fast Prototyping Board ユーザーズマニュアル (R20UT5091J)」の「7.1 e<sup>2</sup> studio で COM port デバッグを使用する場合」と「7.2 CS+ を COM port デバッグを使用する場合」を参照してください。

図 5-1 システム設定 (e<sup>2</sup> studio、CS+)



次に、図 5-2 に本サンプルコード（IAR）で使用しているシステム設定を示します。

図 5-2 システム設定（IAR）

システム設定

オンチップ・デバッグ設定

オンチップ・デバッグ動作設定

使用しない  エミュレータを使う  COMポート

エミュレータ設定

E2  E2 Lite

疑似RRM/DMM機能設定

使用しない  使用する

Start/Stop関数機能設定

使用しない  使用する

通過ポイント機能設定

使用しない  使用する

トレース機能設定

使用しない  使用する

セキュリティID設定

セキュリティIDを設定する

セキュリティID

セキュリティID認証失敗時の設定

フラッシュ・メモリのデータを消さない  フラッシュ・メモリのデータを消去する

チェックをつける

チェックをつける

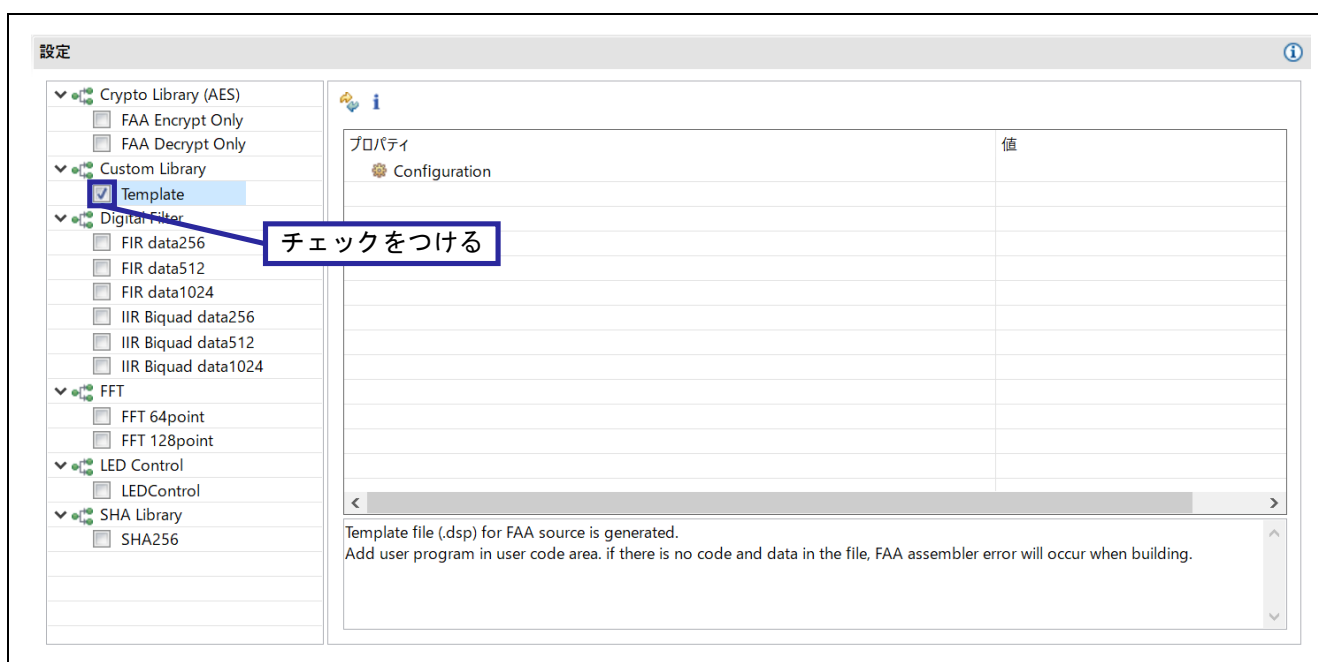
## 5.1.2 コンポーネントの設定

本サンプルコードで使用しているコンポーネントの設定を以下に示します。

表 5-1 コンポーネントの設定 (FAA)

項目	内容
コンポーネント	フレキシブル・アプリケーション・アクセラレータ
コンフィグレーション名	Config_FAA
リソース	FAA

図 5-3 FAA の設定



**注意** サンプルコードを読み込み後に FAA ライブラリが表示されない場合、「フレキシブル・アプリケーション・アクセラレータ(FAA)ツールガイド CS+編 (e<sup>2</sup> studio 編)」の「2.3.1 FAA コンポーネントの追加」の No.11 を参照して、FAA ライブラリをダウンロードしてください。

表 5-2 コンポーネントの設定 (A/D コンバータ)

項目	内容
コンポーネント	A/D コンバータ
コンフィグレーション名	Config_ADC
リソース	ADC
動作モード	アドバンスド・モード

図 5-4 A/D コンバータの設定 (1/2)

**設定**

コンバータ動作設定  
 停止                       許可

分解能設定  
 10ビット                       8ビット                       12ビット  
チェックをつける

VREF(+)設定  
 VDD                       AVREFP                       内部基準電圧  
チェックをつける

VREF(-)設定  
 VSS                       AVREFM  
チェックをつける

同時サンプリング機能設定  
 同時サンプリング                      未使用

トリガ要因                      INTTM01 信号

1st S&H回路入力ソース                      ANI0

2nd S&H回路入力ソース                      ANI2

3rd S&H回路入力ソース                      ANI3

変換の優先順位                      Low

動作モード設定  
 ワンショット・セレクト・モード

A/Dチャンネル0設定                      チェックをつける  
 A/Dチャンネル0の有効(ADS0)  
 トリガ要因                      ソフトウェア・トリガ                      ソフトウェア・トリガに変更  
 入力ソース                      ANI2                      ANI2に変更  
 変換の優先順位                      低

A/Dチャンネル1設定                      チェックをつける  
 A/Dチャンネル1の有効(ADS1)  
 トリガ要因                      ソフトウェア・トリガ                      ソフトウェア・トリガに変更  
 入力ソース                      ANI3                      ANI3に変更  
 変換の優先順位                      低

図 5-5 A/D コンバータの設定 (2/2)

**A/Dチャンネル2設定**

A/Dチャンネル2の有効(ADS2)

トリガ要因: ソフトウェア・トリガ

入力ソース: ANI3

変換の優先順位: 低

**A/Dチャンネル3設定**

A/Dチャンネル3の有効(ADS3)

トリガ要因: ELCからのイベント入力

入力ソース: ANI3

変換の優先順位: 低

**変換時間設定**

fCLKは48 MHz以下に設定してください。

変換時間モード: 標準1

サンプリング・クロック・サイクル: 20 fAD

変換時間: 43/fCLK (0.8958 μs)

20fADに変更

43/fCLKに変更

**変換結果上限/下限値設定**

ADLL ≧ ADCRn ≧ ADULで割り込み要求信号(INTAD0からINTAD3)を発生

ADUL < ADCRnまたはADLL > ADCRnで割り込み要求信号(INTAD0からINTAD3)を発生

上限値(ADUL): 255

下限値(ADLL): 0

**割り込み設定**

チェックをつける

A/Dチャンネル0 割り込み(INTAD0)を使用 優先順位: レベル3(低優先順位)

ADS0で指定されたアナログ入力チャンネルの変換状態を保存する

A/Dチャンネル1 割り込み(INTAD1)を使用 優先順位: レベル3(低優先順位)

ADS1で指定されたアナログ入力チャンネルの変換状態を保存する

A/Dチャンネル2 割り込み(INTAD2)を使用 優先順位: レベル3(低優先順位)

ADS2で指定されたアナログ入力チャンネルの変換状態を保存する

A/Dチャンネル3 割り込み(INTAD3)を使用 優先順位: レベル3(低優先順位)

ADS3で指定されたアナログ入力チャンネルの変換状態を保存する

表 5-3 コンポーネントの設定 (16 ビット・タイマ KB30)

項目	内容
コンポーネント	PWM 出力
コンフィグレーション名	Config_TKB0
リソース	TKB0
動作	単体動作モード (TKBCR00 レジスタによる周期制御)

図 5-6 16 ビット・タイマ KB30 の設定

**設定**

カウント・ソース設定

動作クロック: CK20

クロック・ソース: fKBKC (クロック周波数: 96000 kHz, PLLをfKBKCとして選択します)

**PWM出力設定**

PWM周期: 10  $\mu$ s (実際の値: 10)

デューティ(TKB00出力): 50 (%) (実際の値: 50)

デューティ(TKB01出力): 50 (%) (実際の値: 50)

デレイ(TKB01出力): 10 (%) (実際の値: 10)

**A/D変換スタート・タイミング信号出力機能設定**

TKBTGCR0値: 100

**出力設定**

TKB000出力許可 (チェックをつける)

デフォルト・レベル: Lowレベル

アクティブ・レベル: Highレベル

TKB001出力許可

デフォルト・レベル: Lowレベル

アクティブ・レベル: Highレベル

**PWM出力ソフト・スタート機能設定**

ソフト・スタート初期デューティは、以下の条件で設定してください。  
 $0\% \leq \text{TKBO00ソフト・スタート初期デューティ} < \text{デューティ(TKB000出力)} \leq 100\%$   
 $\text{デレイ(TKB001出力)} \leq \text{TKBO01ソフト・スタート初期デューティ} < \text{デレイ(TKB001出力)} + \text{デューティ(TKB001出力)} \leq 100\%$

TKB000ソフト・スタート機能を有効にする

TKBO00ソフト・スタート初期デューティ: 10 (%) (実際の値: 10)

TKBO00ソフト・スタート・ステップ幅: 1

TKB001ソフト・スタート機能を有効にする

TKBO01ソフト・スタート初期デューティ: 10 (%) (実際の値: 10)

TKBO01ソフト・スタート・ステップ幅: 1

**割り込み設定**

TKB000強制出力停止解除時に割り込みを発生させる  
優先順位: レベル3(低優先順位)

TKB000強制出力停止発動時に割り込みを発生させる  
優先順位: レベル3(低優先順位)

TKB001強制出力停止解除時に割り込みを発生させる  
優先順位: レベル3(低優先順位)

TKB001強制出力停止発動時に割り込みを発生させる  
優先順位: レベル3(低優先順位)

16ビット・タイマKB30エンド・カウントを有効にする (チェックを外す)

優先順位: レベル3(低優先順位)



表 5-4 コンポーネントの設定 (16 ビット・タイマ KB31)

項目	内容
コンポーネント	PWM 出力
コンフィグレーション名	Config_TKB1
リソース	TKB1
動作	単体動作モード (TKBCR00 レジスタによる周期制御)

図 5-7 16 ビット・タイマ KB31 の設定

**設定**

カウント・ソース設定

動作クロック: CK20

クロック・ソース: fKBKC (クロック周波数: 96000 kHz, fPLLをfKBKCとして選択します)

**PWM出力設定**

PWM周期: 10  $\mu$ s (実際の値: 10)

デューティ(TKB10出力): 50 (%) (実際の値: 50)

デューティ(TKB11出力): 50 (%) (実際の値: 50)

デイレイ(TKB11出力): 10 (%) (実際の値: 10)

**A/D変換スタート・タイミング信号出力機能設定**

TKBTGCR1値: 100

**出力設定**

TKBO10出力許可 (チェックをつける)

デフォルト・レベル: Lowレベル

アクティブ・レベル: Highレベル

TKBO11出力許可

デフォルト・レベル: Lowレベル

アクティブ・レベル: Highレベル

**PWM出力ソフト・スタート機能設定**

ソフト・スタート初期デューティは、以下の条件で設定してください。  
 $0\% \leq \text{TKBO10ソフト・スタート初期デューティ} < \text{デューティ(TKB10出力)} \leq 100\%$   
 $\text{デイレイ(TKB11出力)} \leq \text{TKBO11ソフト・スタート初期デューティ} < (\text{デイレイ(TKB11出力)} + \text{デューティ(TKB11出力)}) \leq 100\%$

TKBO10ソフト・スタート機能を有効にする

TKBO10ソフト・スタート初期デューティ: 10 (%) (実際の値: 10)

TKBO10ソフト・スタート・ステップ幅: 1

TKBO11ソフト・スタート機能を有効にする

TKBO11ソフト・スタート初期デューティ: 10 (%) (実際の値: 10)

TKBO11ソフト・スタート・ステップ幅: 1

---

**割り込み設定**

TKBO10強制出力停止解除時に割り込みを発生させる

優先順位: レベル3(低優先順位)

TKBO10強制出力停止発動時に割り込みを発生させる

優先順位: レベル3(低優先順位)

TKBO11強制出力停止解除時に割り込みを発生させる

優先順位: レベル3(低優先順位)

TKBO11強制出力停止発動時に割り込みを発生させる

優先順位: レベル3(低優先順位)

16ビット・タイマKB31エンド・カウントを有効にする (チェックを外す)

優先順位: レベル3(低優先順位)

表 5-5 コンポーネントの設定 (タイマ・アレイ・ユニット)

項目	内容
コンポーネント	PWM 出力
コンフィグレーション名	Config_TAU0_0
リソース	TAU0_0
動作	PWM 機能

図 5-8 タイマ・アレイ・ユニットの設定

設定

**PWMクロック設定**

動作クロック CK00 ▼

クロック・ソース fCLK ▼ (クロック周波数: 48000 kHz)

**PWM周期設定**

周期設定 10 μs ▼ (実際の値: 10)

10に変更

**割り込み設定**

タイマ・チャンネル0のカウンタ完了で割り込み発生(INTTM00)

優先順位 レベル3(低優先順位) ▼

**PWMスレープ選択設定**

チャンネル1スレープ       チャンネル2スレープ       チャンネル3スレープ

マスタ・チャンネルを複数使用する場合、マスタ・チャンネルをまたいだスレープ・チャンネルの設定はできません。

**PWMスレープ設定**

スレープ1

**PWMデューティ設定**

デューティ 50 (%) (実際の値: 50%)

**出力設定**

初期出力値 0 ▼

出力レベル アクティブ・ハイ ▼

**割り込み設定**

タイマ・チャンネル1のカウンタ完了で割り込み発生(INTTM01)

チェックを外す

優先順位 レベル3(低優先順位) ▼

表 5-6 コンポーネントの設定 (イベント・リンク・コントローラ)

項目	内容
コンポーネント	イベント・リンク・コントローラ
コンフィグレーション名	Config_ELC
リソース	ELC

図 5-9 イベント・リンク・コントローラの設定

**設定**

出力先設定

<input type="checkbox"/> A/D変換開始	イベント発生元	外部割り込みエッジ検出0
<input type="checkbox"/> TAU00デレイ・カウント開始、インプット・キャプチャ、イベント・カウント	イベント発生元	外部割り込みエッジ検出1
<input type="checkbox"/> TAU01デレイ・カウント開始、インプット・キャプチャ、イベント・カウント	イベント発生元	外部割り込みエッジ検出2
<input type="checkbox"/> タイマRJ0イベント・カウント	イベント発生元	外部割り込みエッジ検出3
<input type="checkbox"/> TRGIOBのインプットキャプチャ	イベント発生元	外部割り込みエッジ検出4
<input type="checkbox"/> TRDIOD0のインプット・キャプチャ、タイマRD0のパルス出力遮断	イベント発生元	外部割り込みエッジ検出5
<input type="checkbox"/> TRDIOD1のインプット・キャプチャ、タイマRD1のパルス出力遮断	イベント発生元	キー・リターン信号検出
<input type="checkbox"/> DA0リアルタイム出力	イベント発生元	RTC定周期信号/アラーム一致検出
<input type="checkbox"/> DA1リアルタイム出力	イベント発生元	TRD0インプット・キャプチャA/コンパア一致A
<input type="checkbox"/> DA2リアルタイム出力	イベント発生元	TRD0インプット・キャプチャB/コンパア一致B
<input type="checkbox"/> PWMOPAパルス出力強制遮断	イベント発生元	TRD1インプット・キャプチャA/コンパア一致A
<input checked="" type="checkbox"/> FAA入力イベント検出割り込み0	イベント発生元	TAU チャンネル0カウント完了/キャプチャ完了
<input type="checkbox"/> FAA入力イベント検出割り込み1	イベント発生元	タイマRD2カウンタ1アンダフロー
<input type="checkbox"/> FAA入力イベント検出割り込み2	イベント発生元	タイマRJアンダフロー
<input type="checkbox"/> FAA入力イベント検出割り込み3	イベント発生元	タイマRG2インプットキャプチャA/コンパア一致A
<input type="checkbox"/> FAA入力イベント検出割り込み4	イベント発生元	タイマRG2インプットキャプチャB/コンパア一致B
<input type="checkbox"/> FAA入力イベント検出割り込み5	イベント発生元	タイマRG2コンパア一致C
<input type="checkbox"/> FAA入力イベント検出割り込み6	イベント発生元	タイマRG2コンパア一致D
<input type="checkbox"/> FAA入力イベント検出割り込み7	イベント発生元	32ビット・インターバル・タイマチャンネル0インターバル信号検出

TAUチャンネル0カウント完了/  
キャプチャ完了  
に変更

チェックをつける

## 5.2 フォルダ構成

表 5-7、表 5-8 にサンプルコードの使用するソースファイル／ヘッダファイルの構成を示します。なお、統合開発環境で自動生成されるファイル、bsp 環境のファイルは除きます。

表 5-7 フォルダ構成 (1/2)

フォルダ、ファイル名	説明	スマート・コンフィグレータを使用
\r01an7250_faa_ad_cotrol<DIR> <sup>注3</sup>	サンプルコードのフォルダ	
\src<DIR>	プログラム格納用フォルダ	
main.c	サンプルコードソースファイル	
r_faa_ad_control.c	FAA 用ソースファイル	
r_faa_ad_control.h	FAA 用ヘッダファイル	
\smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ	√
\Config_ADC<DIR>	ADC 用プログラム格納フォルダ	√
Config_ADC.c	ADC 用ソースファイル	√
Config_ADC.h	ADC 用ヘッダファイル	√
Config_ADC_user.c	ADC 用割り込みソースファイル	√ <sup>注1</sup>
\Config_ELC<DIR>	ELC 用プログラム格納フォルダ	√
Config_ELC.c	ELC 用ソースファイル	√
Config_ELC.h	ELC 用ヘッダファイル	√
Config_ELC_user.c	ELC 用割り込みソースファイル	√ <sup>注1</sup>
\Config_FAA<DIR>	FAA 用プログラム格納フォルダ	√
Config_FAA_common.c	Common FAA module のソースファイル	√
Config_FAA_common.h	Common FAA module のヘッダファイル	√
Config_FAA_common.inc	FAA 用インクルードファイル	√
Config_FAA_src.dsp	FAA 用アセンブラ・ソースファイル	√ <sup>注2</sup>
\Config_TAU0_0<DIR>	TAU0_0 用プログラム格納フォルダ	√
Config_TAU0_0.c	TAU0_0 用ソースファイル	√
Config_TAU0_0.h	TAU0_0 用ヘッダファイル	√
Config_TAU0_0_user.c	TAU0_0 用割り込みソースファイル	√ <sup>注1</sup>
\Config_TKB0<DIR>	TKB0 用プログラム格納フォルダ	√
Config_TKB0.c	TKB0 用ソースファイル	√
Config_TKB0.h	TKB0 用ヘッダファイル	√
Config_TKB0_user.c	TKB0 用割り込みソースファイル	√ <sup>注1</sup>

表 5-8 フォルダ構成 (2/2)

フォルダ、ファイル名	説明	スマート・コンフィグレータを使用
\\r01an7250_faa_ad_cotrol<DIR> <sup>注3</sup>	サンプルコードのフォルダ	
\\src<DIR>	プログラム格納用フォルダ	
\\Config_TKB1<DIR>	TKB1 用プログラム格納フォルダ	√
Config_TKB1.c	TKB1 用ソースファイル	√
Config_TKB1.h	TKB1 用ヘッダファイル	√
Config_TKB1_user.c	TKB1 用割り込みソースファイル	√ <sup>注1</sup>
¥general<DIR>	初期化、共通プログラム格納フォルダ	
¥r_bsp<DIR>	BSP 用プログラム格納フォルダ	
¥r_config<DIR>	プログラム格納フォルダ	

補足 ” <DIR> ” は、ディレクトリを意味します。

注1. 本サンプルコードでは使用しません。

注2. 本サンプルコードでは、FAA ライブラリの Custom Library を選択しているため、コード生成直後はテンプレートのみでコードは記載されていません。サンプルコード用にコードを追加しています。また、FAA のプログラムはアセンブラで記載する必要があります。FAA のアセンブラ命令の詳細は「RL78/G24 ユーザーズマニュアル ハードウェア編 (R01UH0961)」の「4.16 各命令の説明」を参照してください。

注3. IAR 版のサンプルコードは r01an7250\_faa\_ad\_cotrol.ipcf を格納しています。ipcf ファイルについては、「RL78 スマート・コンフィグレータ ユーザーガイド : IAR 編 (R20AN0581)」を確認してください。

### 5.3 オプション・バイトの設定一覧

表 5-9 にオプション・バイト設定を示します。

表 5-9 オプション・バイト設定

アドレス	設定値	内容
000C0H/040C0H	1110 1111B (EFH)	ウォッチドッグ・タイマ動作停止 (リセット解除後、カウント停止)
000C1H/040C1H	1111 1011B (FBH)	LVD0 リセット・モード 検出電圧：立ち上がり 2.97V／立下り 2.91V
000C2H/040C2H	1110 1010B (EAH)	フラッシュ動作モード：高速メインモード 高速オンチップ・オシレータの周波数：8MHz
000C3H/040C3H	1000 0101B (85H)	オンチップ・デバッグ動作許可

### 5.4 定数一覧

表 5-10、表 5-11 に本サンプルコードで使用する定数一覧を示します。

表 5-10 サンプルコードで使用する定数 (CPU プログラム)

定数名	設定値	内容
FAA_ALREADY_RUNNING	1	FAA 実行中を示す値
FAA_SUCCESS	0	FAA 処理成功を示す値
FAA_ADC_COMPLETED	1	A/D 変換完了を示す値
FAA_ADC_FAILED	2	A/D 変換失敗を示す値
FAA_ADC_NOTCOMPLETED	0	A/D 変換未完了を示す値

表 5-11 サンプルコードで使用する定数 (FAA プログラム)

定数名	設定値	内容
_C_ADControl_ADS	2	A/D 変換を行うアナログチャンネルの設定値
_C_ADControl_0	0	A/D 変換失敗時の変換結果 (CPU プログラム用)
_C_ADControl_1	1	A/D 変換成功を示す値
_C_ADControl_2	2	CPU からリクエストされた A/D 変換の有無を判定する値
_C_ADControl_3	3	A/D 変換失敗を示す値
_C_ADControl_4	4	未使用
_C_ADControl_8	8	未使用
_C_ADControl_FAAAP_ADS0	#ADS0_PTR (015H)	ADS0 レジスタに FAA・アドレス・ポインタでアクセス時の設定値
_C_ADControl_FAAAP_ADINST	#ADINTST_PTR (029H)	ADINTST レジスタに FAA・アドレス・ポインタでアクセス時の設定値
_C_ADControl_FAAAP_ADM3	#ADM3_PTR (014H)	ADM3 レジスタに FAA・アドレス・ポインタでアクセス時の設定値
_C_ADControl_ADCS	80H	A/D コンバータをトリガ待機状態にするための設定値 (ADM0 設定値)
_C_ADControl_ADTRSWT	80H	ソフトウェア・トリガを発生させるための設定値 (ADM3 設定値)
_C_ADControl_FAAAC_ADINTST_CLEAR	00H	A/D 変換結果のステータスをクリアするための設定値 (ADINTST 設定値)
_C_ADControl_TKBTRG_TKBRDT_REQ	01H	TKB30 コンペアレジスタ・書き換え要求 (TKBRDT0 設定値)
_C_ADControl_AD_FAILD_VALUE	FFFFH	A/D 変換失敗時の変換結果 (FAA プログラム用)
_C_ADControl_ADINTST_ST0F_ST0S	03H	A/D 変換完了待ちを判定する値
_C_ADControl_ADINTST_ST0S	02H	A/D 変換成功を判定する値

## 5.5 変数一覧

表 5-12 に本サンプルコードで使用する変数一覧を示します。

表 5-12 サンプルコードで使用する変数 (CPU プログラム)

変数名	型	内容	使用関数
result_buffer	uint16_t	A/D 変換結果を格納	main
ad_status	uint16_t	A/D 変換の状態を格納	main
adc_status	uint16_t	SHDMEM から A/D 変換の状態を格納	r_faa_adcontrol_getad
adc_result	e_faa_result_adc_t	A/D 変換の状態を示す戻り値を格納	r_faa_adcontrol_getad
buffer	uint16_t	SHDMEM から A/D 変換結果を格納	r_faa_adcontrol_getad

## 5.6 関数一覧

表 5-13、表 5-14 にサンプルコードで使用する関数、処理を示します。ただし、スマート・コンフィグレータで生成された関数の内、変更を行っていないものは除きます。

表 5-13 関数一覧 (CPU プログラム)

関数名	概要	ソースファイル
main	メイン処理	main.c
r_faa_adcontrol_start	FAA 開始処理	r_faa_ad_control.c
r_faa_adcontrol_stop	FAA 停止処理	r_faa_ad_control.c
r_faa_adcontrol_requestadc	指定されたアナログチャネルの A/D 変換リクエスト処理	r_faa_ad_control.c
r_faa_adcontrol_getad	A/D 変換結果取得処理	r_faa_ad_control.c

表 5-14 処理一覧 (FAA プログラム)

ラベル名	概要	ソースファイル
_P_ADControl	割り込み許可、割り込み待ち	Config_FAA_src.dsp
_P_ADControl_TAU0_Interrupt	TKBO00 端子のデューティ比の変更、TKBO10 端子用のデューティ比を SHDMEM に格納	Config_FAA_src.dsp
_P_ADControl_GetAd	A/D 変換実行	Config_FAA_src.dsp

## 5.7 関数仕様

サンプルコードの関数仕様を示します。

### CPU プログラム

#### [関数名] main

概要	メイン処理
ヘッダ	r_smc_entry.h、platform.h、r_faa_ad_control.h、Config_TKB0.h、Config_TKB1.h、Config_ADC.h
宣言	void main (void);
説明	A/D コンバータ、TKB30、TKB31、タイマ・アレイ・ユニット (TAU0) の初期設定を行い、TAU チャネル 0 カウント完了割り込み (INTTM00) で一定周期 (10 $\mu$ s) のタイミングごとに、FAA の動作を開始させます。また、FAA へ A/D 変換のリクエストを行い、変換完了後、変換結果を取得し、TKB31 のデューティ比を変更します。
引数	なし
リターン値	なし
備考	なし



## [関数名] r\_faa\_adcontrol\_start

---

概要	FAA 開始処理	
ヘッダ	Config_FAA_common.h、 r_faa_ad_control.h	
宣言	FAA_Status_t r_faa_adcontrol_start(void);	
説明	FAA のスタックポインタ、FAA プログラムの開始アドレスを設定し、FAA の動作を開始します。	
引数	なし	
リターン値	FAA_SUCCESS	FAA 処理成功
	FAA_ALREADY_RUNNING	FAA 実行中
備考	なし	

## [関数名] r\_faa\_adcontrol\_stop

---

概要	FAA 停止処理	
ヘッダ	Config_FAA_common.h、 r_faa_ad_control.h	
宣言	void r_faa_adcontrol_stop(void);	
説明	FAA の動作を停止します。	
引数	なし	
リターン値	なし	
備考	なし	

## [関数名] r\_faa\_adcontrol\_requestadc

---

概要	指定されたアナログチャンネルの A/D 変換リクエスト処理	
ヘッダ	Config_FAA_common.h、 r_faa_ad_control.h	
宣言	void r_faa_adcontrol_requestadc (e_ad_channel_t channel);	
説明	指定されたアナログチャンネルの A/D 変換を FAA 機能にリクエストします。	
引数	channel : A/D 変換を行うアナログチャンネル	
リターン値	なし	
備考	なし	

## [関数名] r\_faa\_adcontrol\_getad

---

概要	A/D 変換結果取得処理	
ヘッダ	Config_FAA_common.h、 r_faa_ad_control.h	
宣言	e_faa_result_adc_t r_faa_adcontrol_getad (uint16_t * const buffer);	
説明	r_faa_adcontrol_requestadc()によってリクエストされた A/D 変換の結果を取得します。	
引数	buffer : A/D 変換結果	
リターン値	FAA_ADC_COMPLETED	A/D 変換完了
	FAA_ADC_FAILED	A/D 変換失敗
	FAA_ADC_NOTCOMPLETED	A/D 変換未完了
備考	なし	

## FAA プログラム

## [ラベル名] \_P\_ADControl

---

概要	割り込み許可、割り込み待ち
ヘッダ	Config_FAA_common.inc
宣言	—
説明	割り込みを許可し、TAU のチャンネル 0 カウント完了割り込み (INTTM00) の発生を待ちます。
引数	なし
リターン値	なし
備考	なし

## [ラベル名] \_P\_ADControl\_TAU0\_Interrupt

---

概要	TKBO00 端子のデューティ比の変更、TKBO10 端子用のデューティ比を SHDMEM に格納
ヘッダ	Config_FAA_common.inc
宣言	—
説明	A/D 変換の結果を元に TKBO00 端子の PWM 出力のデューティ比を変更します。また、CPU からの A/D 変換リクエストがあった場合、指定されているアナログチャンネルの A/D 変換も行い、SHDMEM に格納します。
引数	なし
リターン値	なし
備考	なし

## [ラベル名] \_P\_ADControl\_GetAd

---

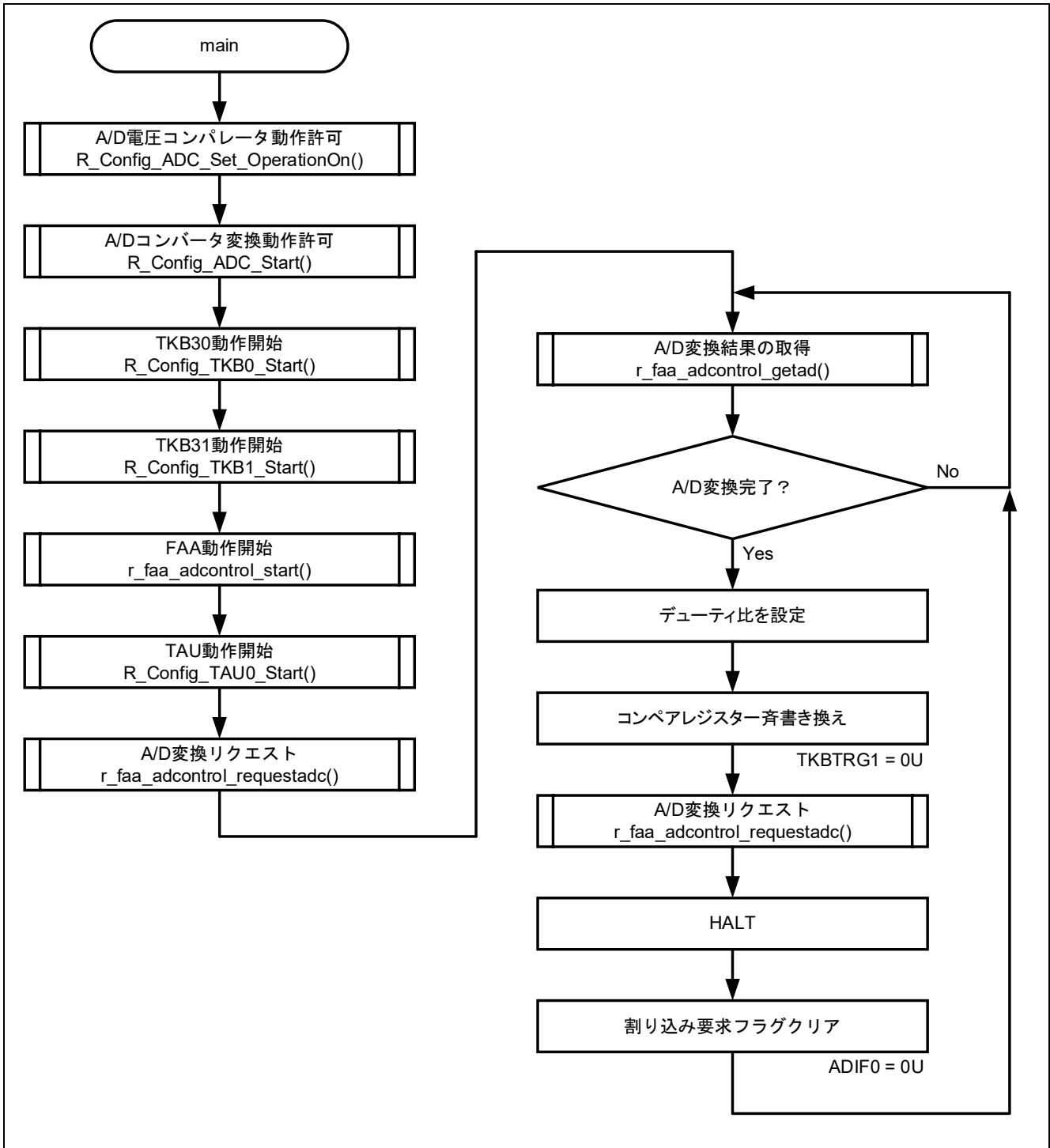
概要	A/D 変換実行
ヘッダ	Config_FAA_common.inc
宣言	—
説明	A/D 変換を行い、変換結果ステータスをクリアします。
引数	なし
リターン値	なし
備考	なし

5.8 フローチャート

5.8.1 メイン処理

図 5-10 にメイン処理のフローチャートを示します。

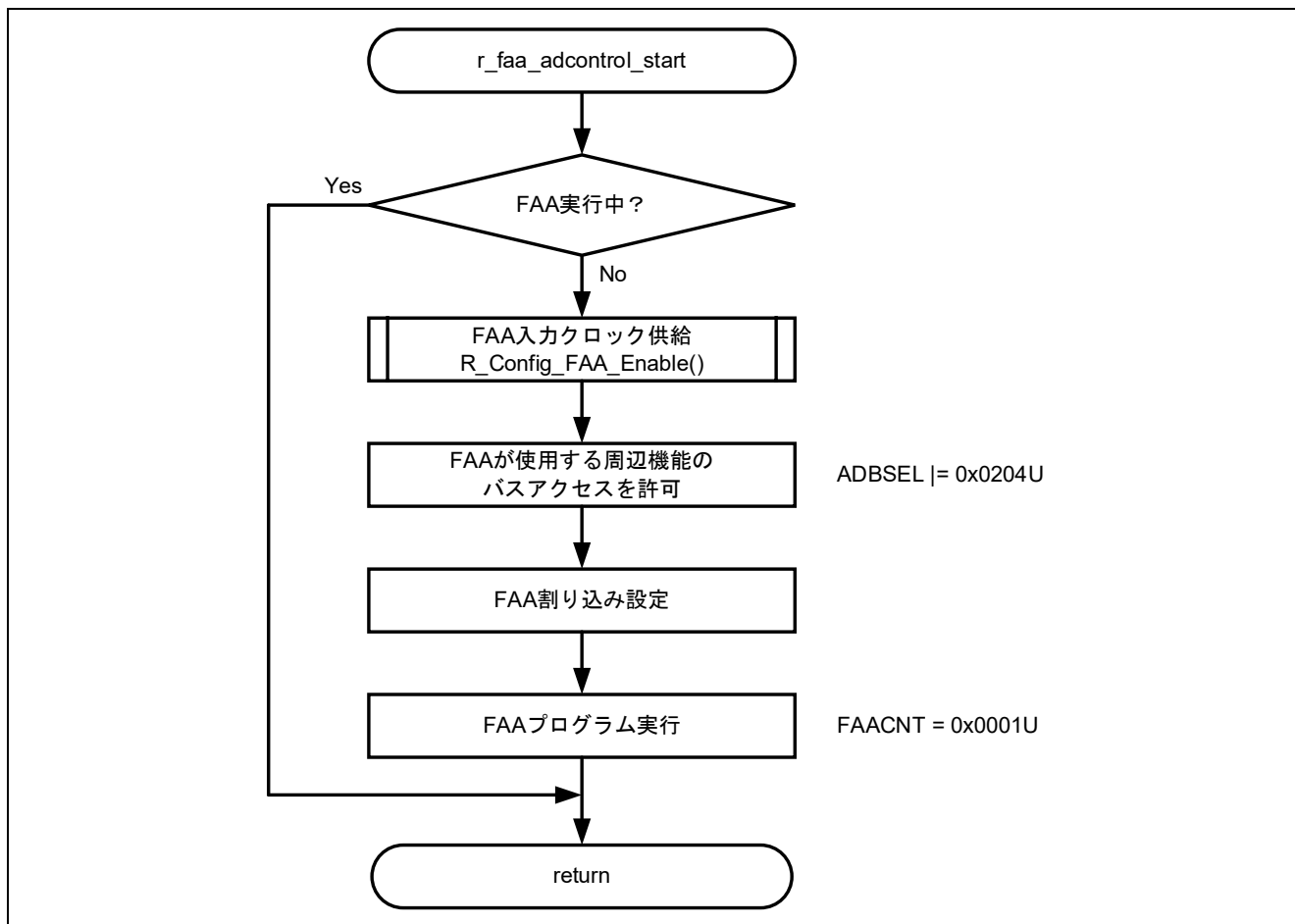
図 5-10 メイン処理



5.8.2 r\_faa\_adcontrol\_start 関数

図 5-11 に r\_faa\_adcontrol\_start のフローチャートを示します。

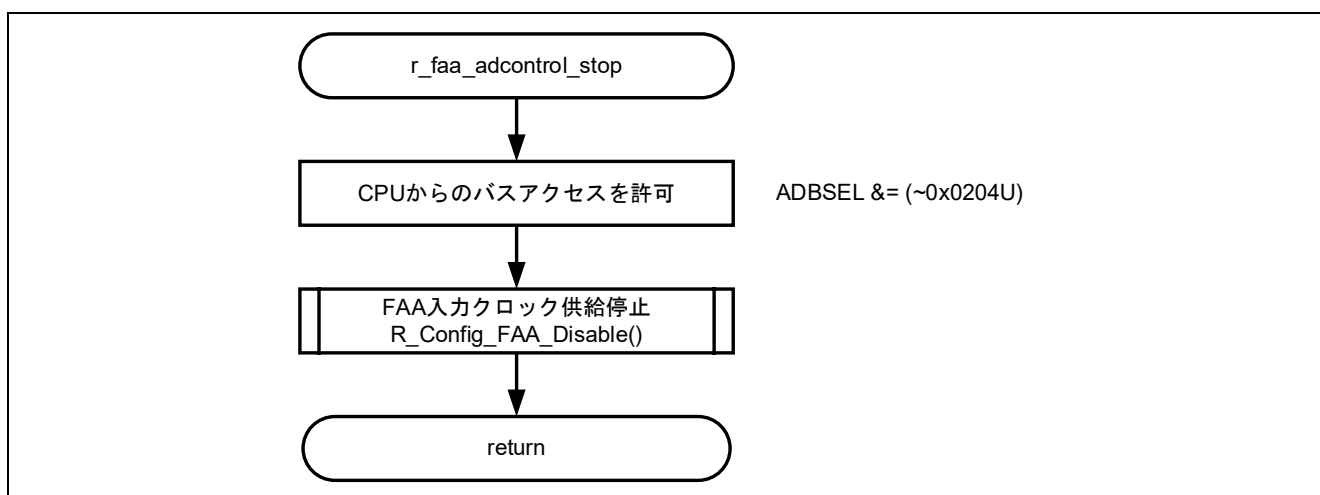
図 5-11 r\_faa\_adcontrol\_start 関数



5.8.3 r\_faa\_adcontrol\_stop 関数

図 5-12 に r\_faa\_adcontrol\_stop のフローチャートを示します。

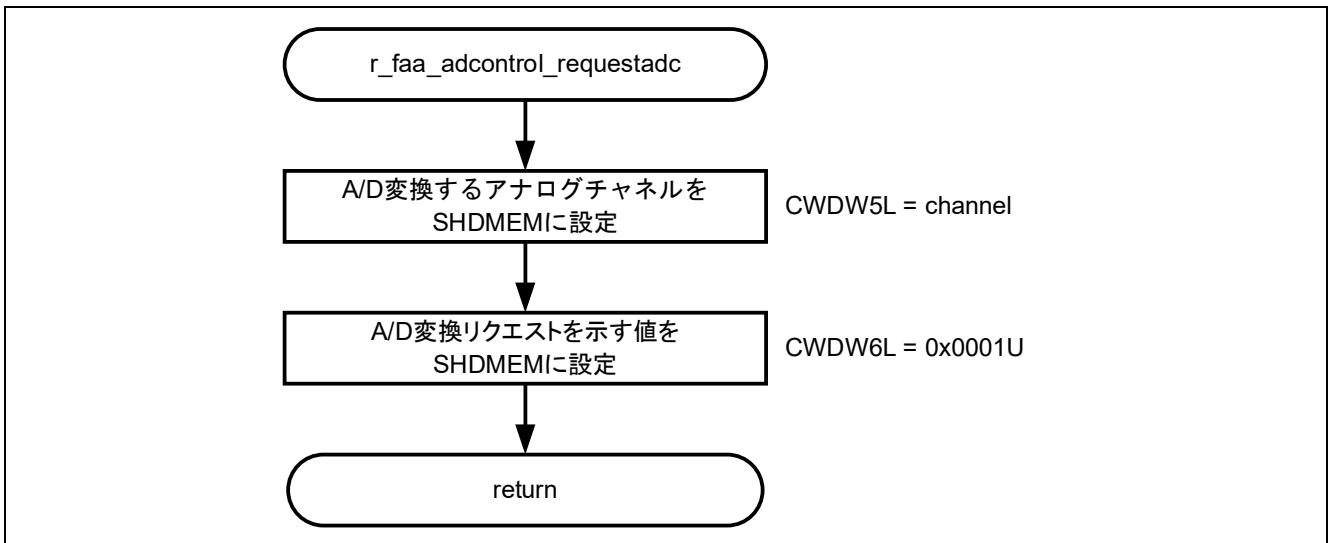
図 5-12 r\_faa\_adcontrol\_stop 関数



5.8.4 r\_faa\_adcontrol\_requestadc 関数

図 5-13 に r\_faa\_adcontrol\_requestadc のフローチャートを示します。

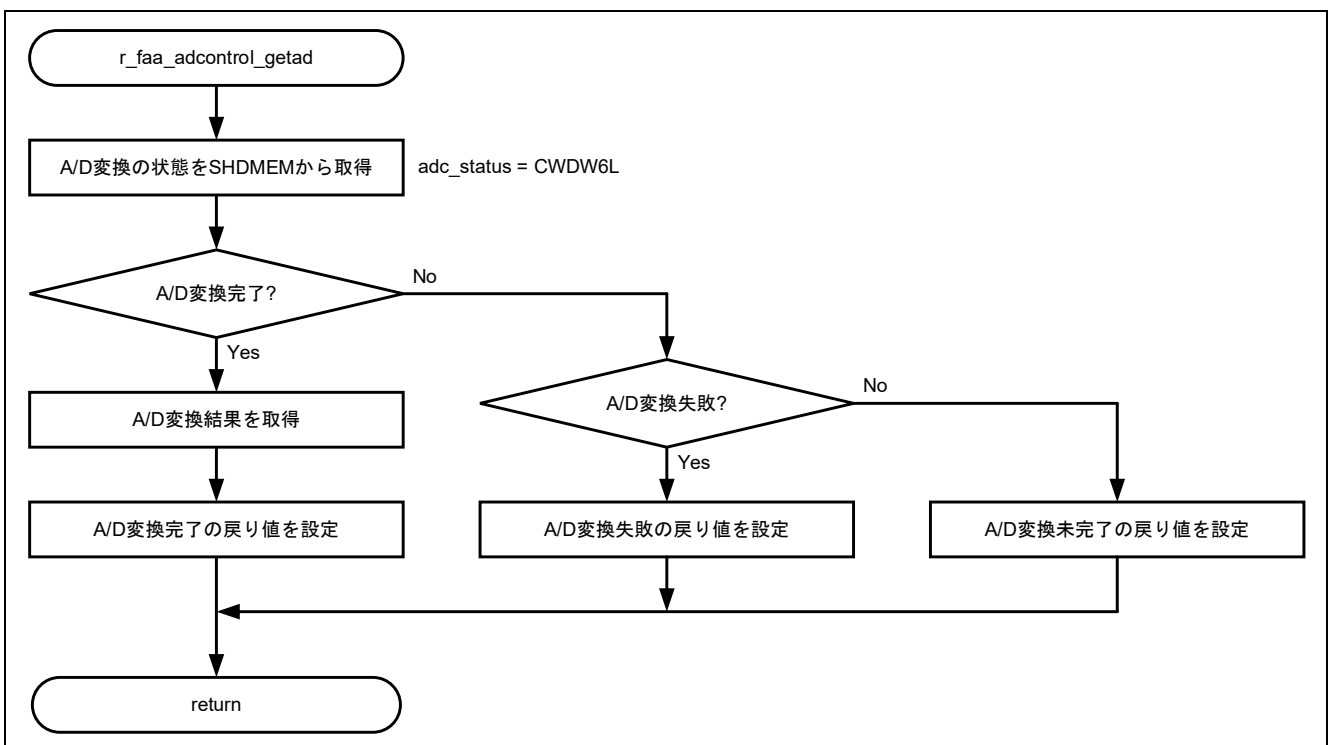
図 5-13 r\_faa\_adcontrol\_requestadc 関数



5.8.5 r\_faa\_adcontrol\_getad 関数

図 5-14 に r\_faa\_adcontrol\_getad のフローチャートを示します。

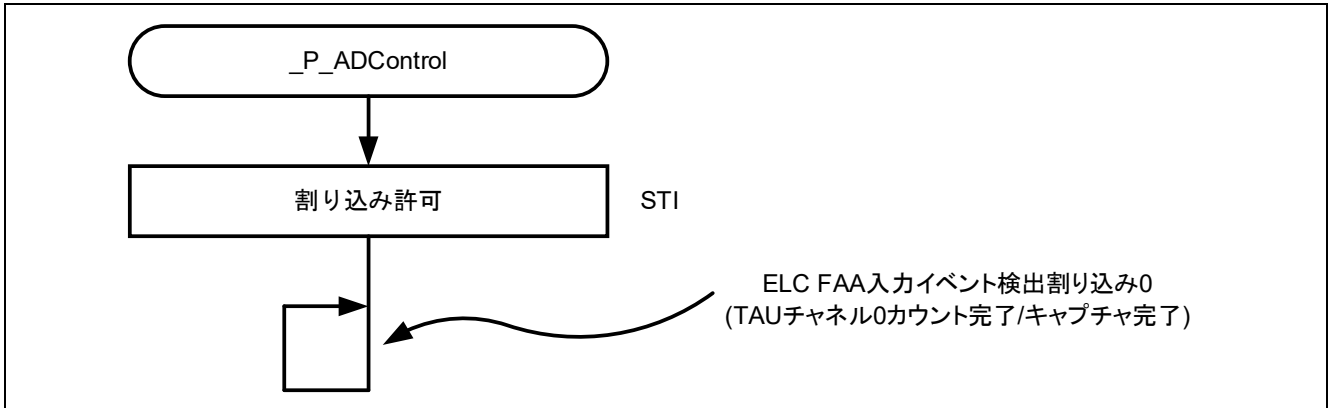
図 5-14 r\_faa\_adcontrol\_getad 関数



5.8.6 FAA 処理 : \_P\_ADControl

図 5-15 に \_P\_ADControl のフローチャートを示します。

図 5-15 \_P\_ADControl



5.8.7 FAA 処理 : \_P\_ADControl\_TAU0\_Interrupt

図 5-16、図 5-17 に \_P\_ADControl\_TAU0\_Interrupt のフローチャートを示します。

図 5-16 \_P\_ADControl\_TAU0\_Interrupt (1/2)

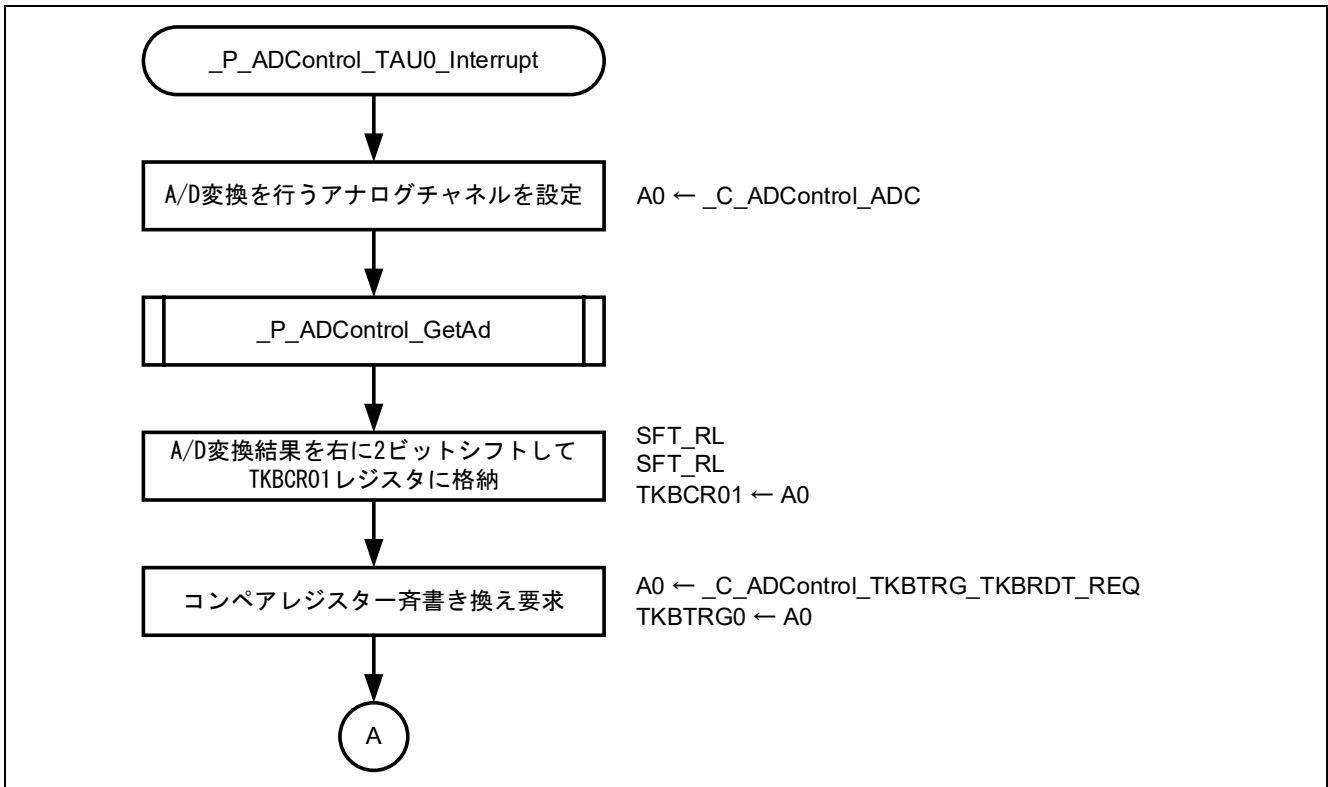
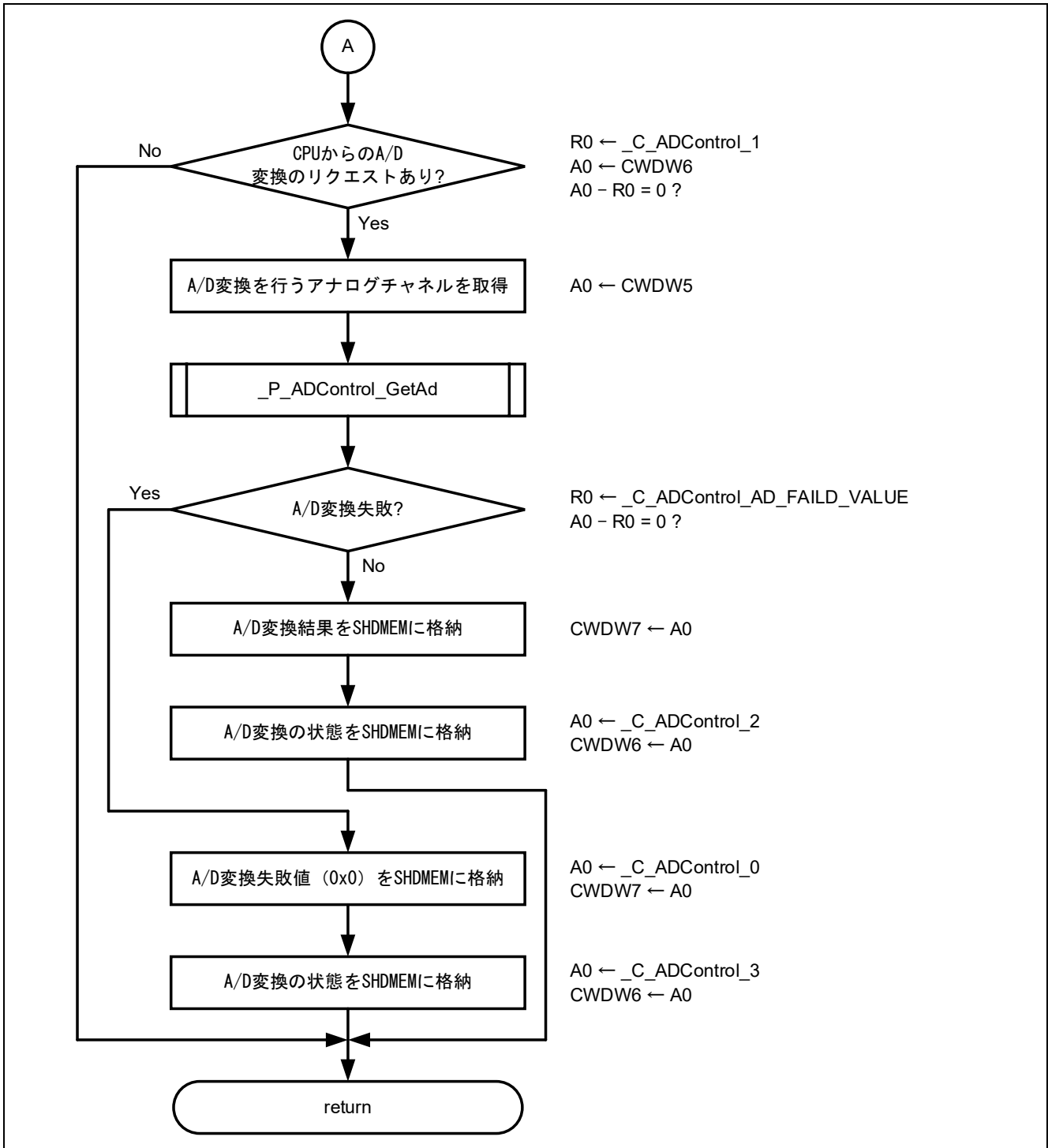


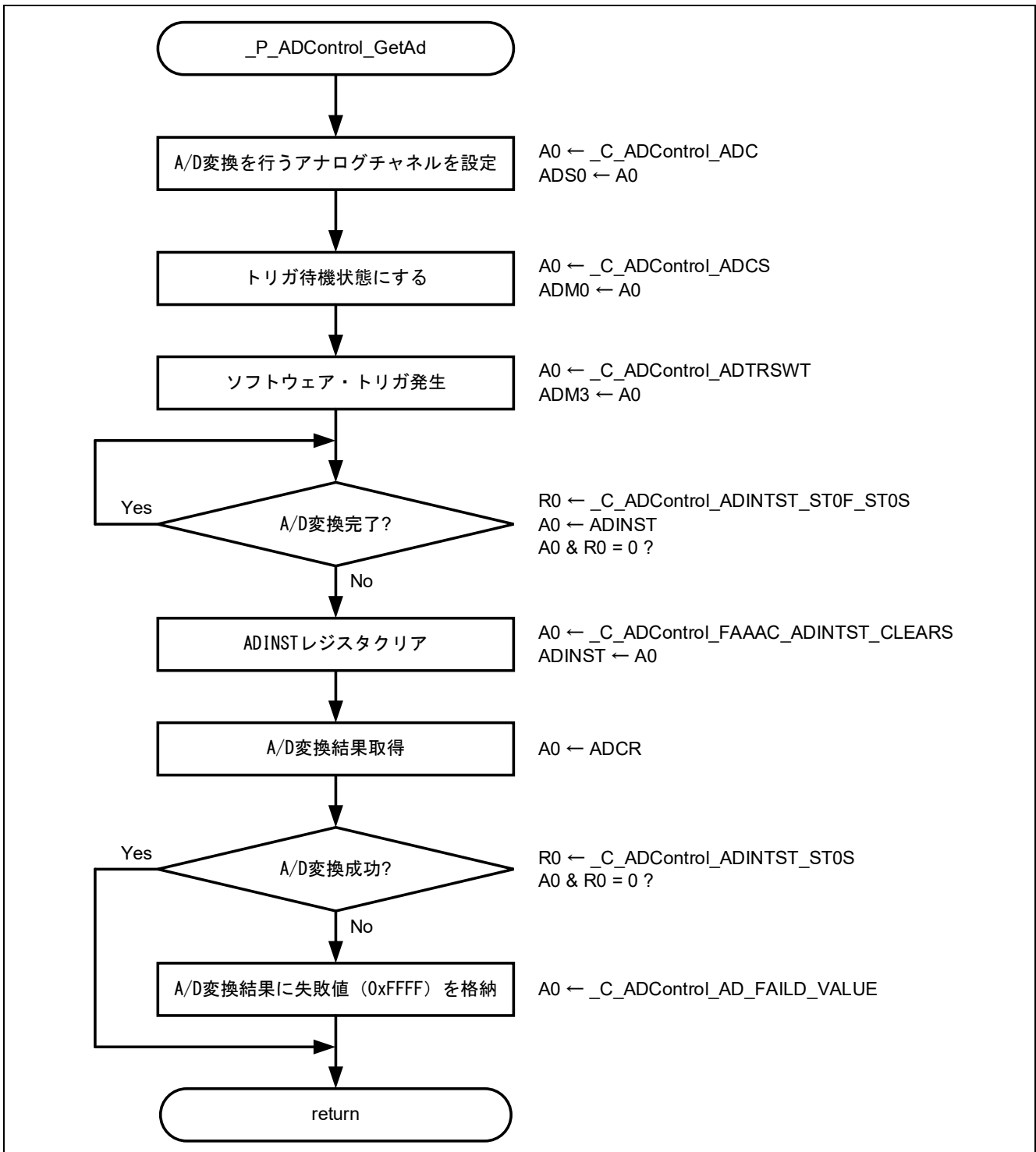
図 5-17 \_P\_ADControl\_TAU0\_Interrupt (2/2)



5.8.8 FAA 処理 : \_P\_ADControl\_GetAd

図 5-18 に \_P\_ADControl\_GetAd のフローチャートを示します。

図 5-18 \_P\_ADControl\_GetAd





## 6. サンプルコード

サンプルコードは、ルネサス エレクトロニクスホームページから入手してください。

## 7. 参考ドキュメント

RL78/G24 ユーザーズマニュアル ハードウェア編 (R01UH0961)

RL78 ファミリ ユーザーズマニュアル ソフトウェア編 (R01US0015)

RL78/G24 Fast Prototyping Board ユーザーズマニュアル (R20UT5091)

RL78 スマート・コンフィグレータ ユーザーガイド : CS+編 (R20AN0580)

RL78 スマート・コンフィグレータ ユーザーガイド : e<sup>2</sup> studio 編 (R20AN0579)

RL78 スマート・コンフィグレータ ユーザーガイド : IAR 編 (R20AN0581)

フレキシブル・アプリケーション・アクセラレータ(FAA)ツールガイド CS+編 (R01AN7094)

フレキシブル・アプリケーション・アクセラレータ(FAA)ツールガイド e<sup>2</sup> studio 編 (R01AN7095)

(最新版をルネサス エレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新版の情報をルネサス エレクトロニクスホームページから入手してください。)

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。

## 改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2024.12.05	—	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しないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

### 4. 未使用端子の処理

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

### 5. クロックについて

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

### 6. 入力端子の印加波形

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

### 7. リザーブアドレス（予約領域）のアクセス禁止

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

### 8. 製品間の相違について

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

## ご注意書き

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

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

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

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

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

[www.renesas.com](http://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/](http://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。