RX ファミリ

アナログ入力信号を FFT するサンプルプログラム

要旨

本アプリケーションノートは、RX マイコンで FFT 処理を行う事例を紹介します。
FFT は、Fast Fourier Transform（高速フーリエ変換）の略で、センサ入力信号や音声信号などの周波数スペクトル解析に広く応用される信号処理です。
添付のサンプルプログラムは、RX マイコン 1 つでアナログ入力信号の A/D 変換と FFT 処理を行います。A/D 変換は 12 ビット、約 1kHz のサンプリング周波数で行い、FFT 処理は 1024 ポイントで行います。

図 1 は、サンプルプログラムの応用例です。周期性、あるいは規則性のある動作をするシステムの評価を行います。最初にターゲットのシステムに取り付けられたセンサの出力信号を FFT 処理し、周波数スペクトルの振幅特性を得ます。得られた周波数スペクトルと期待する周波数スペクトルを比較し、異常を検出します。

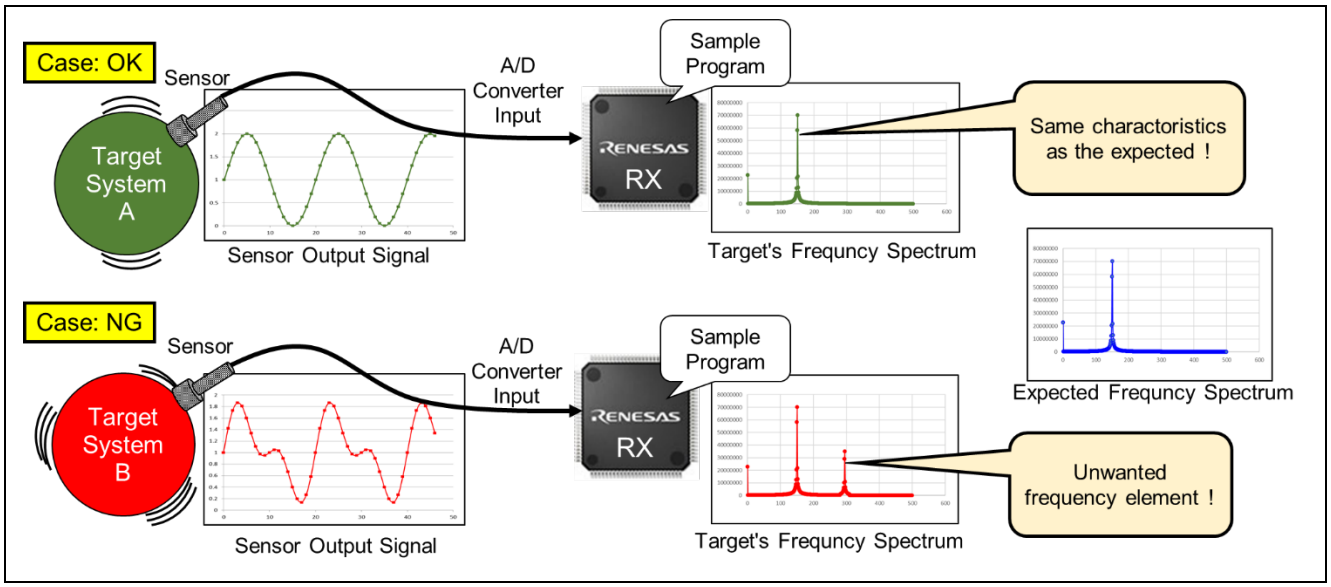


図 1　サンプルプログラムの応用例

対象とする読者

本アプリケーションノートは、マイコンを使用したシステム開発の経験者を対象に説明します。FFT 処理には、RX DSP ライブラリの API を使用するので、FFT、デジタル信号処理について特別な知識は問いません。

動作確認デバイス

RX231 グループ

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合わせて変更し、十分評価してください。

目次

1.　仕様 3
1.1　アプリケーションノートの構成 5
1.2　ソフトウェアの構成 6
1.3　動作環境 7

2.　サンプルプログラム 8
2.1　概要 8
2.2　サンプルプログラムの構成 8
2.3　処理シーケンス 9
2.4　処理フロー 10
2.4.1　初期化 12
2.4.2　FFT 処理 13

3.　参考 14
3.1　使用メモリ 14
3.2　消費電流 14
3.3　サンプリング周波数について 14
3.4　ソフトウェアモジュールの設定 15

4.　開発環境の入手 17
4.1　e²studio の入手方法 17
4.2　コンパイラパッケージの入手方法 17

5.　プロジェクトをインポートする方法 18
5.1　e² studio での手順 18

6.　補足 20
6.1　無償評価版の「RX ファミリ用 C/C++コンパイラパッケージ」を利用する場合の注意事項 20

7.　参考資料 20

改訂記録 21

1. 仕様

サンプルプログラムのシステム概要を図 2 に示します。

このシステムは RX231 に入力されたアナログ信号に FFT 処理を行います。例として、図 3 の正弦波をこのシステムに入力し、FFT 処理を実行し得られた周波数スペクトルの振幅特性を図 4 に示します。

このシステムは次の処理を行います。

- A/D 変換
  12 ビット A/D コンバータ（S12AD）、コンペアマッチタイマ（CMT）、イベントリンクコントローラ（ELC）を使って約 1kHz のサンプリング周波数で A/D 変換を行います。図 5 に示すよう CMT で約 1μs の周期でコンペアマッチイベントを生成、このイベントを ELC で S12AD の A/D 変換開始トリガとして与えます。変換後のデータは DMA コントローラ（DMAC）により入力バッファに転送されます。
- FFT 処理
  RX DSP ライブラリ API Version 5.0 を使用し 1024 ポイントの FFT 処理を実行します。入力信号の周波数スペクトルの振幅特性を出力バッファに格納します。
- CPU 負荷低減
  S12AD、CMT、ELC、DMAC は、一旦設定されれば CPU の介在なしに動作します。この特徴を利用し、図 5 に示すようソフトウェアによる処理実行時のみ通常動作モードで CPU を動作させます。それ以外は WAIT 命令により CPU をスリープモードに遷移させ CPU 負荷を低減します。スリープから通常動作モードへは、DMA 転送終了割り込みにより遷移します。

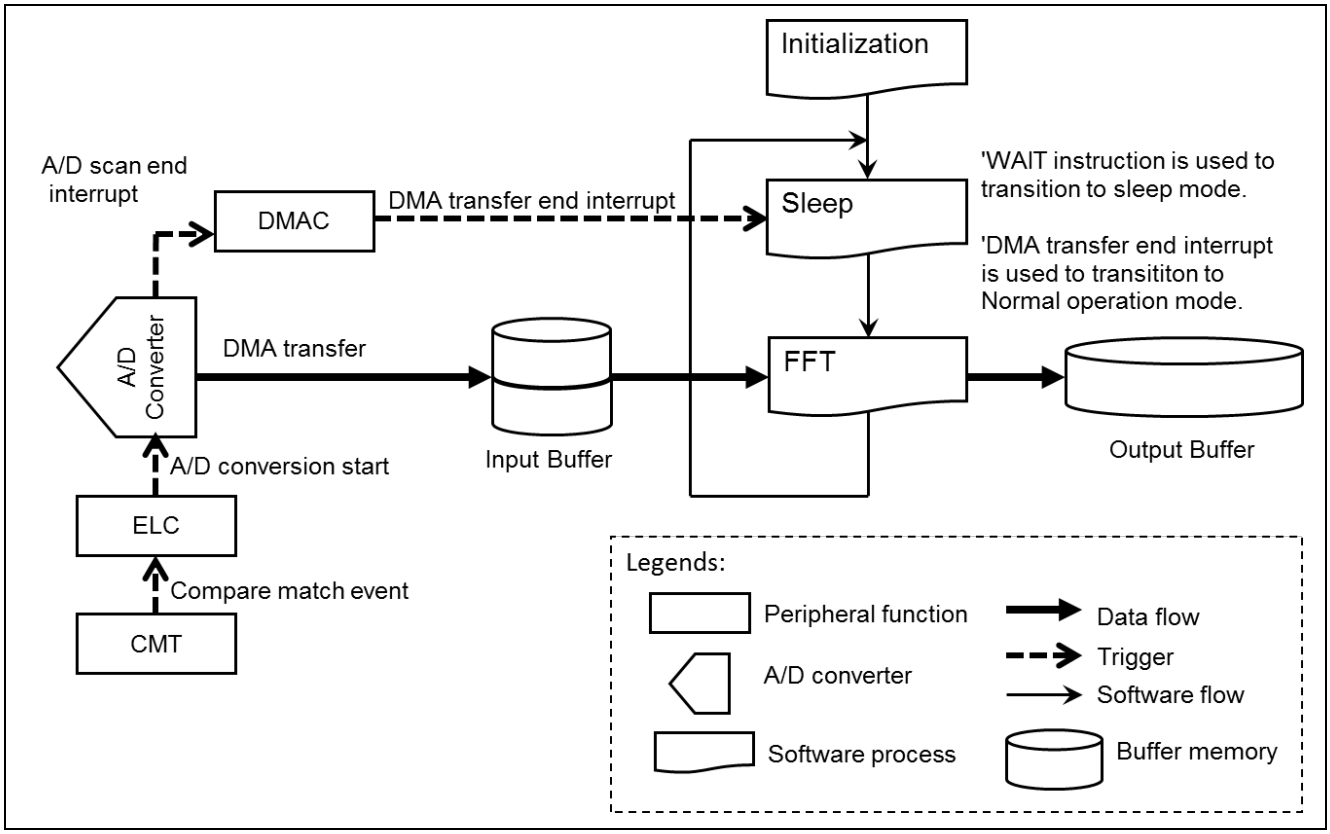


図 2　システム概要

以降、本書はサンプルプログラムについて説明します。サンプルプログラムが使用する周辺機能のドライバ関数、ライブラリ等については、各アプリケーションノートを参照してください。

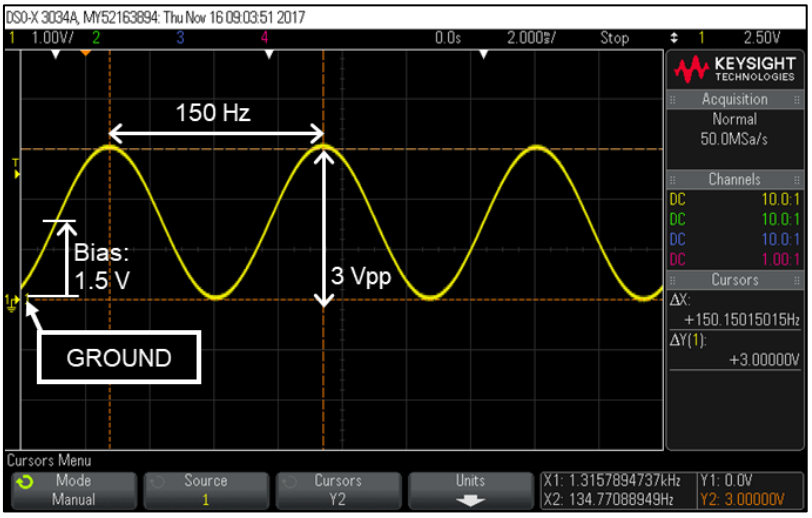


図 3　入力信号（周波数 150Hz、振幅 3Vpp の正弦波）

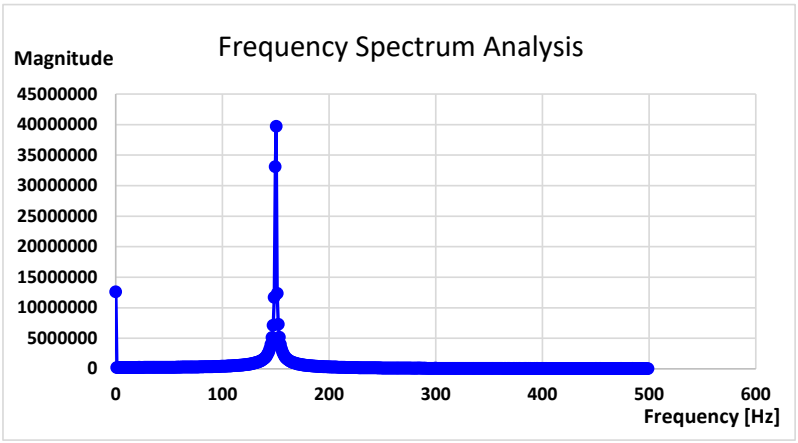


図 4　FFT 処理結果例

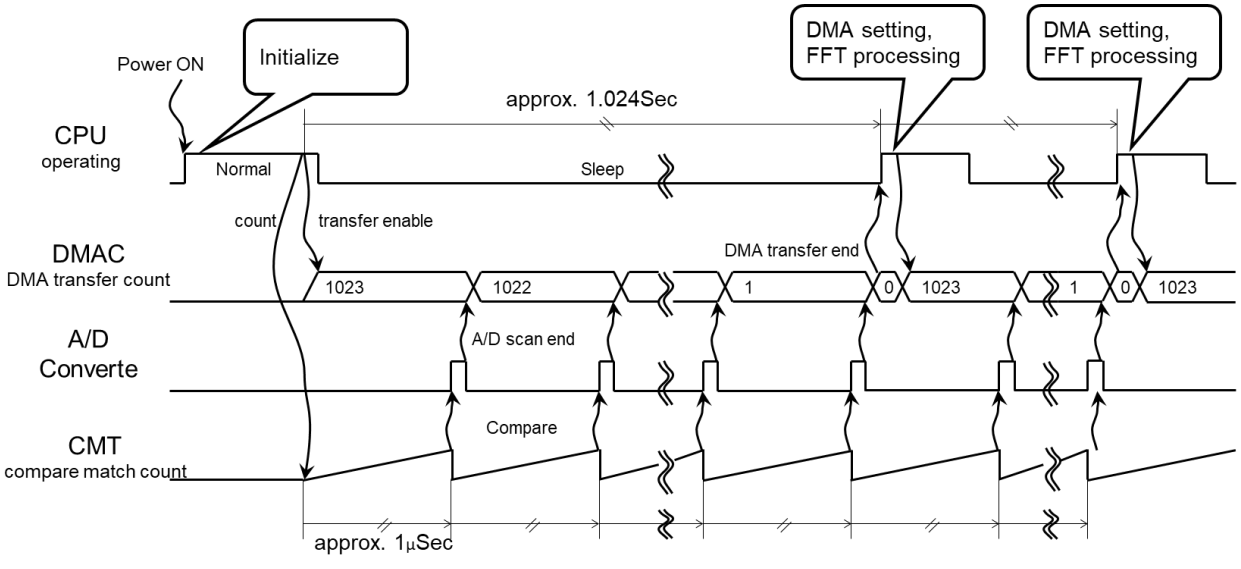


図 5　CPU の動作タイミング

1.1　アプリケーションノートの構成

図 6 に本アプリケーションノートの構成を示します。本アプリケーションノート提供 ZIP ファイルを展開すると、ZIP ファイルと同名のフォルダが作成されます。その下にある「e² studio プロジェクト（rx231_fft_sample）」は、e²studio 形式のプロジェクトです。プロジェクトのフォルダには、図 7 に示すようサンプルプログラムのソースコードのほか、e²studio の設定ファイル、本アプリケーションノートを含みます。本書では、図中の C ソースコードについて説明します。

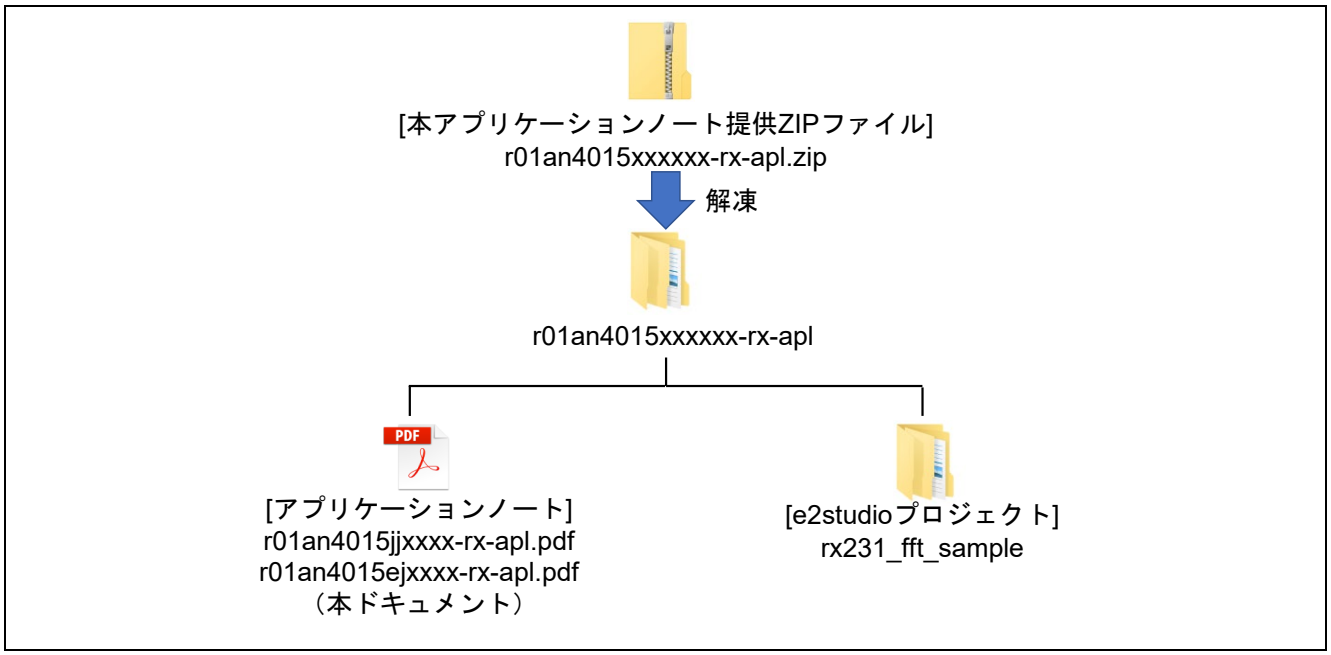


図 6 アプリケーションノートの構成

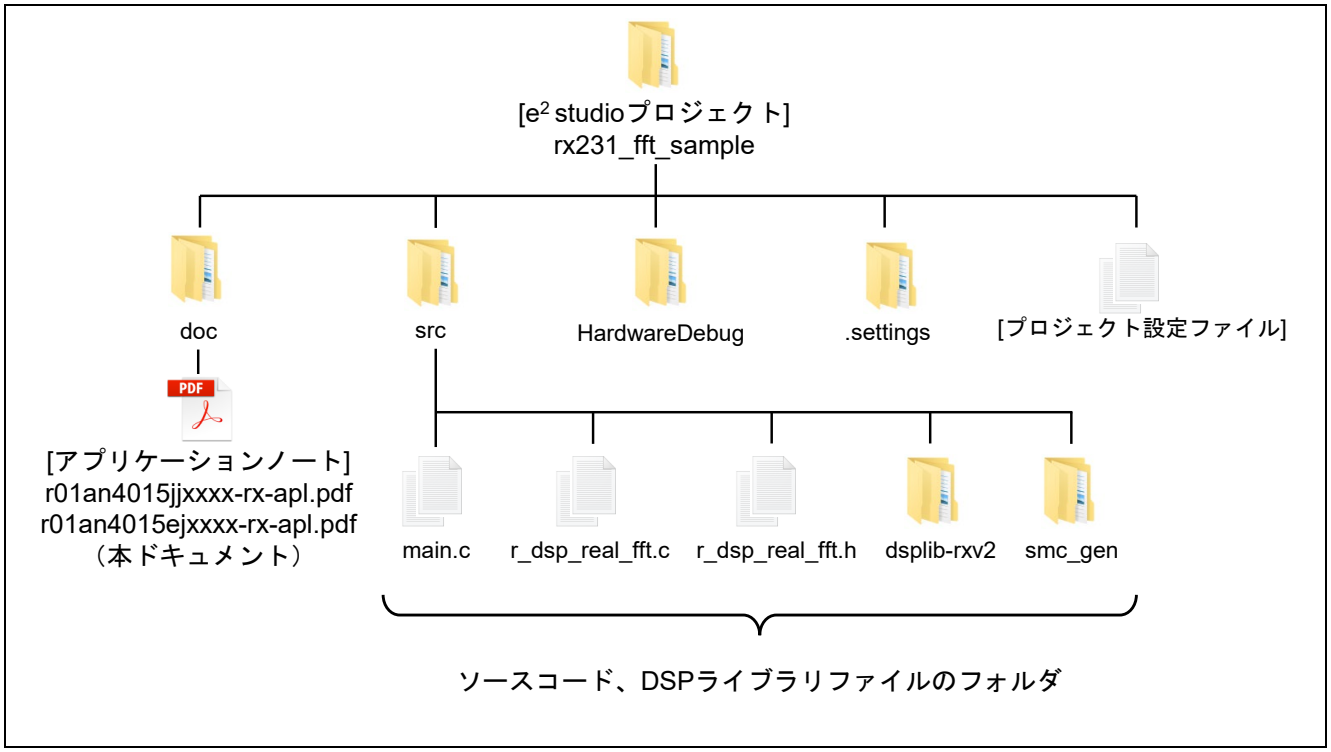


図 7　サンプルプロジェクトのファルダ構成

1.2　ソフトウェアの構成

サンプルプログラムのソフトウェアの構成を図 8 に示します。使用しているソフトウェアモジュールを表 1 に示します。FIT モジュールと DSP ライブラリはルネサスの Web サイトから入手可能です。その他の周辺機能のドライバ関数は e²studio のコード生成機能により生成しています。各ソフトウェアモジュールの詳細は、各アプリケーションノートを参照してください。

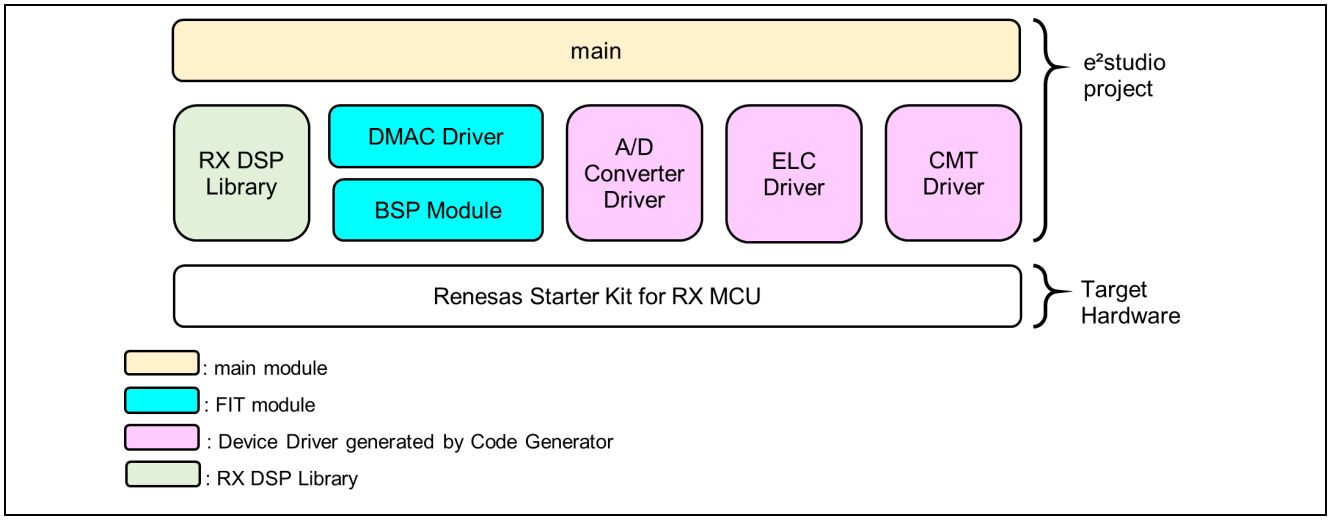


図 8　ソフトウェアの構成

表 1 使用ソフトウェアモジュール一覧

モジュール	ドキュメントタイトル	ドキュメント番号	種類
main	-	-	本アプリケーションノートのために開発した main 関数を含むモジュール
BSP	RX ファミリ ボードサポートパッケージモジュール Firmware Integration Technology	R01AN1685JJ	FIT モジュール
DMAC	RX ファミリ DMA コントローラ DMACA 制御モジュール Firmware Integration Technology	R01AN2063JJ	FIT モジュール
S12AD	-	-	コード生成機能で生成したドライバ関数
CMT	-	-	コード生成機能で生成したドライバ関数
ELC	-	-	コード生成機能で生成したドライバ関数
RX DSP Library	RX Family DSP ライブラリ Version 5.0 (CCRX)	R01AN4359JJ	DSP ライブラリ

1.3　動作環境

本アプリケーションノートのサンプルプログラムは、表 2 の条件で動作を確認しています。

表 2 動作確認条件

項目	説明
MCU	R5F52318ADFP（RX231 グループ）
動作周波数	・　メインクロック: 8 MHz ・　PLL 回路出力: 54 MHz （メインクロックを 2 分周し 13.5 逓倍する） ・　システムクロック（ICLK）: 54 MHz （PLL 回路出力を 1 分周する）
動作電圧	3.3V
動作モード	シングルチップモード
プロセッサモード	スーパバイザモード
統合開発環境	ルネサス エレクトロニクス e^2 studio 2021-01
C コンパイラ	ルネサス エレクトロニクス RX Compiler CC-RX V3.03.00
	コンパイルオプション ・　-lang = c99 ・　-fpu ・　-save_acc
エンディアン	・　データ・エンディアン：リトル・エンディアン ・　デバッグツール設定：リトル・エンディアン
iodefine.h	Version 1.0I
サンプルプログラム	Version 1.1
Evaluation board	ルネサス エレクトロニクス Renesas Starter Kit for RX231 (R0K505231S000BE) の CPU ボード ・　搭載 MCU：上記 ・　ボード設定：デフォルト ・　電源：エミュレータから電源を供給する
Emulator	ルネサス エレクトロニクス E2 エミュレータ Lite
ファンクション ジェネレータ	正弦波を出力できるアナログ信号出力端子を備えた信号発生器。 出力信号は GND に対しバイアス 1. 5V、振幅 3.0Vpp に設定。図 3 参照。 ・　アナログ信号出力(+)を Evaluation board の JA1.10 端子に接続します。 JA1.10 端子に印可された信号は S12AD の AN001 の入力となります。 ・　アナログ信号出力(GND)を Evaluation board の JA1.2 端子に接続します。 JA1.2 端子は Evaluation board の GROUND に接続されています。

2. サンプルプログラム

2.1 概要

サンプルプログラムは、主に main 処理と DMA 転送終了割り込み処理からなります。表 3 に各処理の役割を示します。A/D 変換結果の入力バッファへの転送は、DMAC 機能が担います。

表 3 各処理の役割

処理	役割
main 処理	・ 周辺機能と FFT 処理の初期化 ・ 最初の DMA 転送の開始 ・ DMA 転送終了割り込み処理への DMA 転送先の通知 ・ FFT 処理 ・ CPU をスリープモードに遷移させる
DMA 転送終了割り込み処理	・ 2 回目以降の DMA 転送の設定と転送の開始

2.2 サンプルプログラムの構成

表 1 の main モジュールのファイル構成を表 4 に示します。各 c ソースファイルの関数を表 5、表 6 に示します。その他のモジュールの詳細は各アプリケーションノートを、設定は「3.4 ソフトウェアモジュールの設定」を参照してください。

表 4　main モジュールのファイル構成

ファイル	説明
main.c	・ main 処理と DMA 転送終了割り込み処理。
r_dsp_real_fft.c	・ FFT 処理の初期化、FFT 処理
r_dsp_real_fft.h	・ r_dsp_real_fft.c のヘッダファイル。

表 5 main.c ファイルの関数一覧

関数名	説明
main	・ 周辺モジュールの API で初期化を行う。 ・ 最初の DMA 転送を開始させる。 ・ DMA 転送終了割り込み処理 に次の DMA 転送先を通知する。 ・ r_dsp_real_fft.c の関数で FFT 処理の初期化、FFT 処理を行う。
set_buf_info	・ main-DMA 転送終了割り込み処理間通信の変数に次の DMA 転送先アドレスとデータ数を設定する。
init_dmac_s12ad	・ S12AD を転送元として DMAC のチャンネル 0 を初期化する。
callback_dmac_s12ad	main 処理により DMAC モジュールに登録されるコールバック関数。 ・ DMAC チャンネル 0 の DMA 転送終了割り込み処理を行う。 ・ 次の転送先を DMAC に設定し、DMA 転送を許可する。 ・ 次の転送先の情報を main 処理に要求する。

表 6 r_dsp_real_fft.c ファイルの関数一覧

関数名	説明
R_DSP_REAL_FFT_Init	・ DSP ライブラリの API を使って FFT 処理の初期化を行う。
R_DSP_REAL_FFT_Operation	・ DSP ライブラリの API を使って FFT 処理を行う。

2.3　処理シーケンス

サンプルプログラムの処理シーケンスを図 9 に示します。

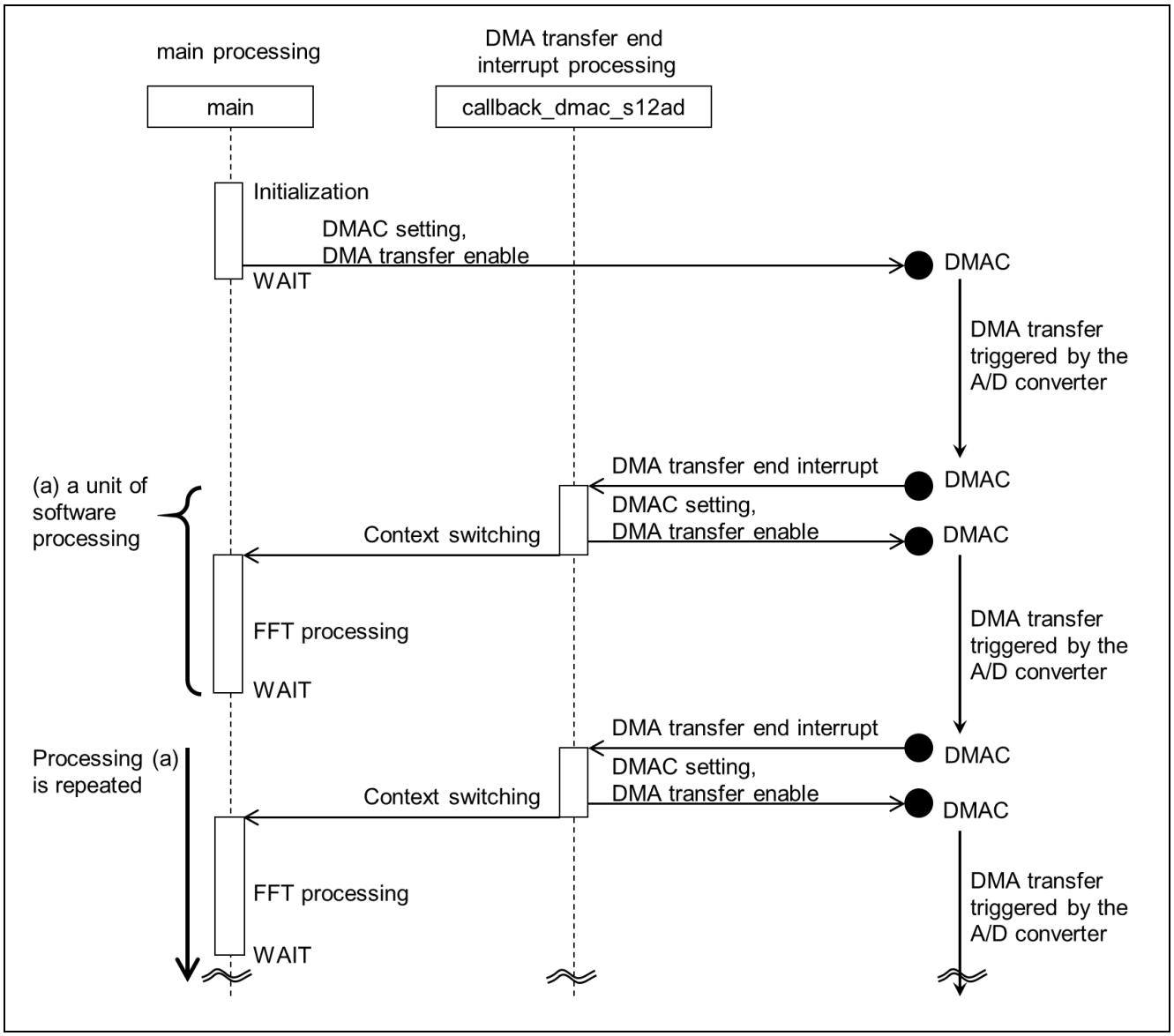


図 9　サンプルプログラムの処理シーケンス

図中、最初の DMA 転送は main 処理が設定し転送を開始させます。指定データ数の DMA 転送の終了による DMA 転送終了割り込みが発生します。これをトリガとして DMA 転送終了割り込み処理と main 処理を順次実行します。以後同じ処理を繰り返し実行します。

DMA 転送は、約 1kHz のサンプリング周波数で周期的に動作する A/D コンバータを転送元としており、main 処理と同時に実行されます。

WAIT は CPU をスリープモードに遷移させるコマンドです。DMA 転送終了割り込みによりスリープモードは解除されます。

2.4　処理フロー

サンプルプログラムの処理フローを図 10 に示します。

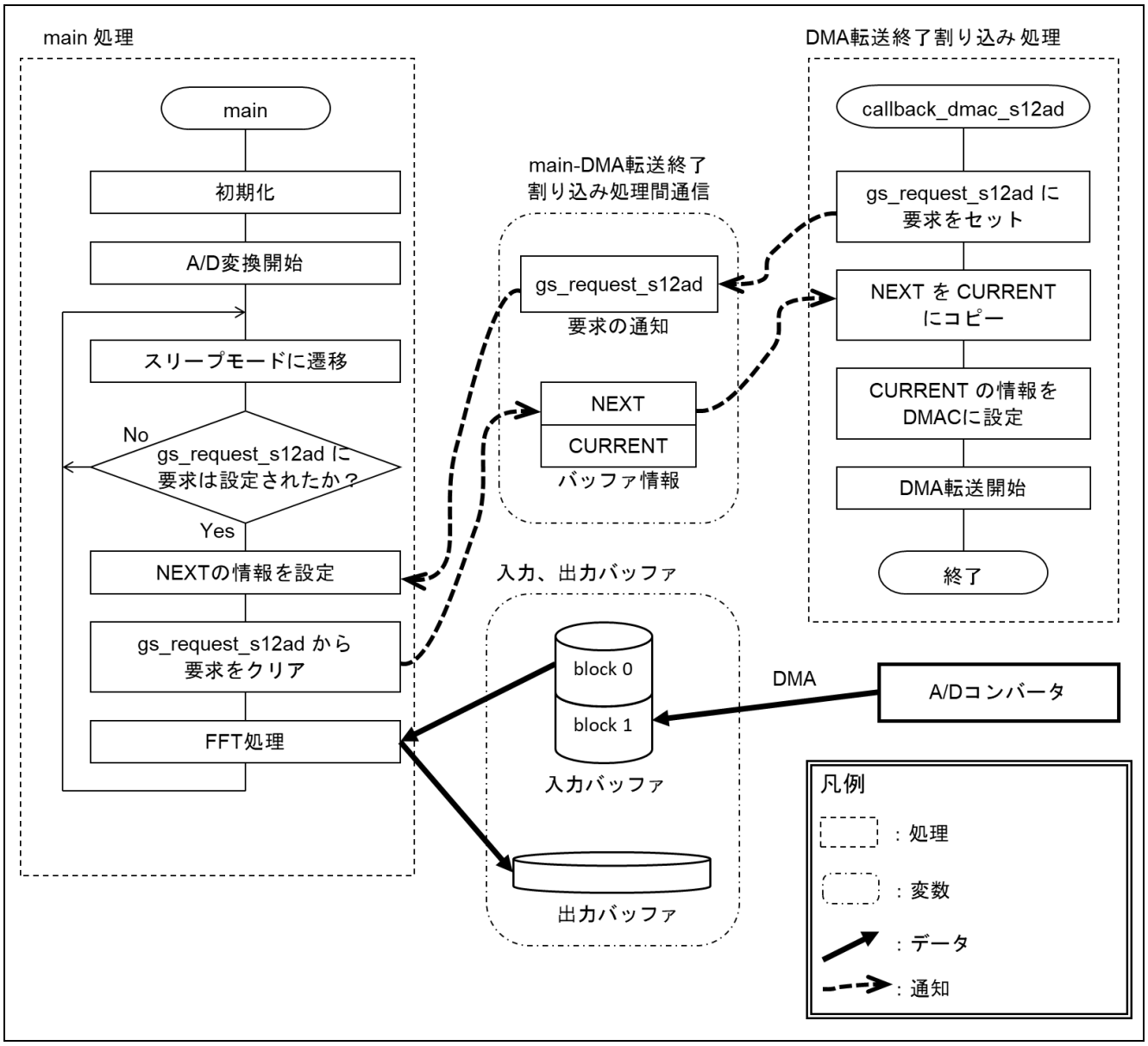


図 10　サンプルプログラムの処理フロー

図 10 の要素について以下に説明します。

- main 処理
  最初に初期化を行い、スリープモードに遷移します。その後、DMAC により A/D 変換結果がバッファに格納されたら、次の DMA 転送先の設定と FFT 処理を行い、CPU をスリープモードに遷移させます。以後同じ処理を繰り返し実行します。
- DMA 転送終了割り込み処理
  次の DMA 転送先の更新を main 処理に要求し、今回の DMA 転送を開始させます。DMA 転送終了割り込みは CPU を通常動作モードに遷移させます。
- main-DMA 転送終了割り込み処理間通信
  main 処理から DMA 転送終了割り込み処理への次の DMA 転送先の通知、DMA 転送終了割り込み処

理から main 処理への次の DMA 転送先の更新の要求は、本処理間通信の変数を介して行います。表 7 に各変数について説明します。

- 入力バッファ
  DMAC が A/D 変換結果を格納し、FFT 処理が入力データとして読み出しを行います。
  CPU（FFT 処理）と DMAC のアクセス競合を避けるため 2 ブロック構成とします。CPU、DMAC それぞれがアクセスするバッファは、図 11 に示すように DMA 転送終了割り込みをトリガとして切り替えます。図 12 にデータのアライメントを示します。
- 出力バッファ
  FFT 処理が処理結果を格納します。図 12 にデータのアライメントを示します。

表 7 main-DMA 転送終了割り込み処理間通信の変数

種類		説明
(a)　要求の通知		DMA 転送終了割り込み処理 が、(b)のバッファの情報の更新を main 処理に要求するための変数。DMA 転送終了割り込み処理が要求をセットする。セットされている場合 main 処理は、(b)　の NEXT 面の情報を更新し、要求をクリアする。 初期化時、main 処理がクリアする。
(b)　バッファ情報		DMA 転送先のバッファの先頭アドレスとデータ数を格納する変数。 処理間のアクセス競合を避けるため NEXT と CURRENT の 2 面で構成する。初期値は 2 面共 main 処理が設定する。
	NEXT	main 処理が、(a)に応じ先頭アドレスとデータ数を格納する面。 以下に初期値を示す。 ・　先頭アドレス：　Input Buffer の block 1 の Data 0 ・　データ数：1024
	CURRENT	DMA 転送終了割り込み処理が、DMAC に設定する先頭アドレスとデータ数を参照する面。DMA 転送終了割り込み処理自身が NEXT 面の情報を CURRENT 面にコピーする。 以下に初期値を示す。 ・　先頭アドレス：　Input Buffer の block 0 の Data 0 ・　データ数：1024

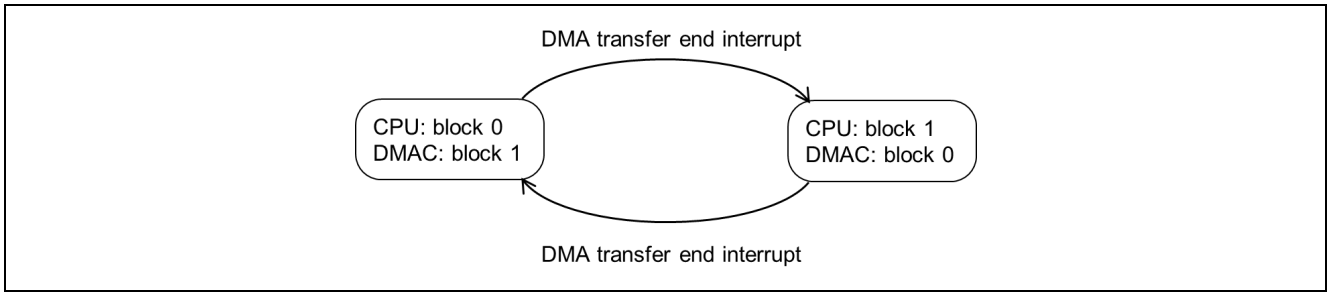
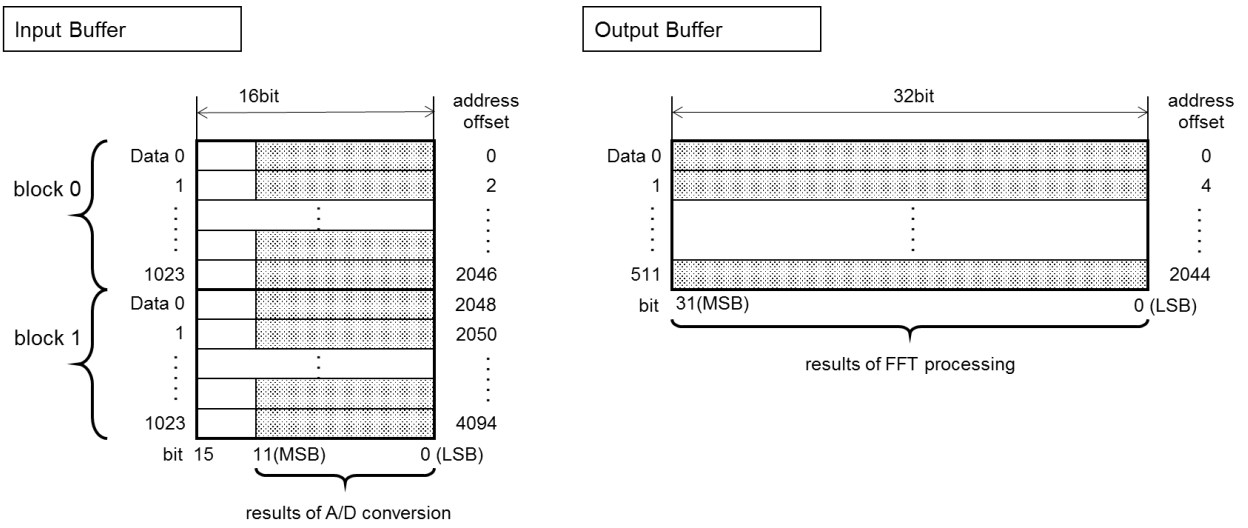


図 11　CPU と DMAC がアクセスする入力バッファの切り替え



注. DSP ライブラリの実数 FFT の API 仕様により、FFT の結果のポイント数は入力の 1/2 となります。
　　このため出力バッファのデータ数は入力バッファの 1/2 とします。

図 12　入力バッファと出力バッファ

2.4.1　初期化

サンプルプログラムは、下記の順序で初期化を行います。

1. 周辺機能の初期化
   CMT、S12AD、ELC、を初期化する。
2. FFT 処理の初期化
   FFT のポイント数、オプション等を設定する。
3. DMAC モジュールの初期化

   使用する DMA チャンネルの選択、コールバック関数の設定、DMA 転送終了割り込みの優先度の設定を行う。
4. main-DMA 転送終了割り込み処理間通信の変数の初期化

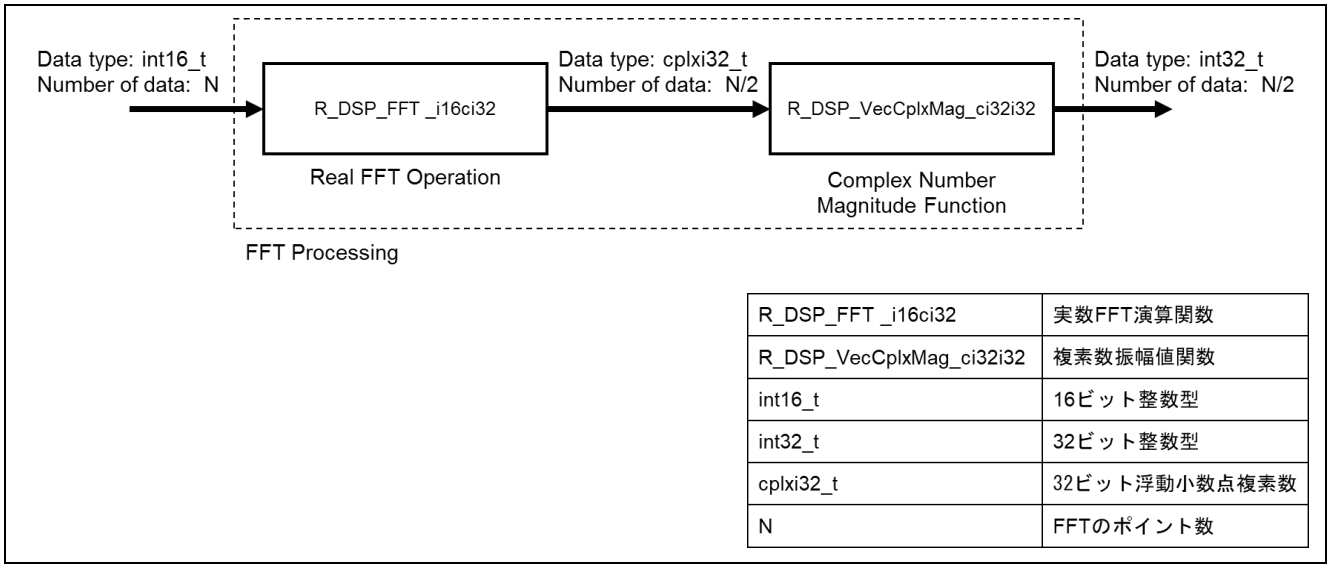
   表 7 の main-DMA 転送終了割り込み処理間通信の変数を初期化する。
5. DMA 転送を許可する

   起動要因、転送モード、転送元アドレス、転送先アドレス、転送データ数等を設定し、DMA 転送を許可する。

この後、A/D 変換動作を開始させると、DMA 転送がはじまります。

2.4.2　FFT 処理

サンプルプログラムは、図 13 に示す DSP ライブラリの API 関数を使って入力信号の周波数スペクトルの振幅特性を出力します。



R_DSP_FFT _i16ci32	実数FFT演算関数
R_DSP_VecCplxMag_ci32i32	複素数振幅値関数
int16_t	16ビット整数型
int32_t	32ビット整数型
cplxi32_t	32ビット浮動小数点複素数
N	FFTのポイント数

図 13　FFT 処理のデータフロー

- FFT 処理を構成する API
  FFT 処理は、実数 FFT 演算関数 R_DSP_FFT_i16ci32 と複素数振幅値関数 R_DSP_VecCplxMag_ci32i32 からなります。
  R_DSP_FFT_i16ci32 は、FFT 結果を複素数で出力します。サンプルプログラムは、入力信号の周波数スペクトルの振幅特性を得ることを目的としたためため、R_DSP_FFT_i16ci32 の複素数出力を R_DSP_VecCplxMag_ci32i32 で整数による振幅値に変換しています。

- 入力、出力データについて
  FFT 処理は、N 個の入力データに対し、N/2 個の結果データを出力します。N は FFT のポイント数に相当し、R_DSP_FFT_i16ci32 の 1 回の処理単位です。R_DSP_FFT_i16ci32 の API 仕様により、結果のポイント数は入力の 1/2 となります。また、R_DSP_VecCplxMag_ci32i32 は Q2.30 フォーマットで出力する仕様のため、出力される振幅値は入力に対し 1/2 となります。

  API 仕様の詳細は「RX DSP ライブラリ API Version 5.0 ユーザーズマニュアル ソフトウェア編」を参照してください。

- 出力データと周波数の関係
  FFT 処理の出力データは、周波数に対する振幅を示します。図 12 に示す出力バッファに表 8 のように周波数に対応する振幅値が格納されます。
  周波数の刻みは、サンプリング周波数と FFT のポイント数により得られます。サンプルプログラムの場合、以下の式により約 0.97Hz 刻みとなります。
  サンプリング周波数/FFT のポイント数 = 1000.296384/1024 = 0.976851938Hz

表 8 出力データと周波数の関係

データ番号	周波数 [Hz]
0	0.97… x 0 = 0
1	0.97… x 1 = 0.97
:	:
510	0.97… x 510 = 498.04
511	0.97… x 511 = 499.02

3. 参考

3.1 使用メモリ

サンプルプログラムのメモリ使用量を表 9 に示します。

ROM と RAM は表 2 の条件で生成した.map ファイルを元に算出、ユーザスタックと割り込みスタックはサンプルプログラムを実行し実際に使用されたスタックメモリを測定しています。

表 9 サンプルプログラムの使用メモリ（参考）

項目	測定値 [byte]	説明
ROM	20398	サンプルプログラムの ROM 使用量。
RAM	16426	サンプルプログラムの RAM 使用量。
ユーザスタック	156	サンプルプログラムのスタック使用量。
割り込みスタック	88	

3.2 消費電流

サンプルプログラム実行時の消費電流をスリープモードの使用／非使用の場合別に表 10 に示します。

RSKRX231 ボード単体でサンプルプログラムを実行し、以下条件で電源からボードに流入する電流を測定しています。

- DC 安定化電源で RSKRX231 ボードの PWR 端子に 3.3V を供給
- A/D 変換、FFT 処理が動作している状態

表 10 サンプルプログラムの消費電流（参考）

スリープモード	測定値 [mA]	説明
使用する	25	サンプルプログラムで測定。
使用しない	40	サンプルプログラムから WAIT 命令だけを削除したプログラムで測定。

3.3 サンプリング周波数について

A/D 変換のサンプリング周期は、CMT1 のコンペアマッチイベントで生成しています。以下に実際に生成されるサンプリング周波数、表 11 に CMT1 のクロック設定を示します。

- サンプリング周波数

  CMT1 のクロックソース/カウント = (27000kHz / 8) / (3374 + 1) =1.000kHz

表 11 CMT1 のクロック設定

項目	設定値
周辺モジュールクロック B(PCLKB)	27MHz
クロックソース	PCLK/8
カウント	3374

3.4　ソフトウェアモジュールの設定

サンプルプログラムで使用した FIT モジュール、e²studio のコード生成機能の設定、DSP ライブラリの設定を表 12～表 17 に示します。各ソフトウェアモジュールの詳細は、それぞれのアプリケーションノートを参照してください。

表 12 BSP モジュールの設定

分類	項目	設定、説明
スマート・コンフィグレータ >> コンポーネント >> r_bsp		プロパティは下記の変更以外はデフォルトの設定とする。
	Parameter checking	Disabled
スマート・コンフィグレータ >> クロック		「クロック」タブを下記の設定とし r_bsp_config.h に反映させる。
	VCC 設定	3.3(V)
	メインクロック設定	動作：チェックする 発振源：発振子 周波数：8 (MHz) 発振安定時間：8192 サイクル、2048 (μs)
	PLL 回路設定	分周比： x 1/2 逓倍比： x 13.5
	サブクロック発振器設定	動作：チェックを外す
	HOCO クロック設定	動作：チェックを外す
	LOCO クロック設定	動作：チェックを外す
	システムクロック設定	クロックソース：PLL 回路 システムクロック(ICLK)：x1　54 (MHz) 周辺モジュールクロック(PCLKA)：x1　54 (MHz) 周辺モジュールクロック(PCLKB)：x1/2　27 (MHz) 周辺モジュールクロック(PCLKD)：x1　54 (MHz) 外部バスクロック(BCLK)：x1/64　0.84375 (MHz) FlashIF クロック(FCLK)：x1/64　0.84375 (MHz)
	IWDT 専用クロック設定	動作：チェックを外す
	USB-PLL 回路設定	分周比： x 1/2 逓倍比： x 12
	外部バスクロック端子(BCLK 端子)設定	動作：チェックを外す

表 13 DMAC モジュールの設定

分類	項目	設定、説明
r_dmaca_rx_config.h		下記の変更以外はデフォルトの設定とする。
	DMACA_CFG_PARAM_CHECKING_ENABLE	0 に変更（コンパイル時にパラメータチェック処理をコードから省略する）

表 14 スマート・コンフィグレータの設定（S12AD）

分類	項目	設定
スマート・コンフィグレータ>> コンポーネント >> シングルスキャンモード S12AD（Config_S12AD0）		下記の設定とし、コードを生成する。
	アナログ入力モード設定	ダブルトリガモード：チェックを外す
	アナログ入力チャネル設定	AN001 のみチェックする
	変換開始トリガ設定	開始トリガソース：ELC からのトリガ
	割り込み設定	AD 変換終了割り込みを許可（S12ADI0）にチェックする。 優先順位：レベル 0（割り込み禁止）
	AD 変換値を加算/平均	AN001 のチェックを外す
	A/D 変換動作選択ビット	高速変換動作

	高電位側基準電圧選択ビット	VREFH0
	低電位側基準電圧選択ビット	VREFL0
	自己診断設定	モード：未使用
	断線検出アシスト設定	チャージ設定：未使用
	データレジスタ設定	データレジスタフォーマット：右詰にする 自動クリアイネーブル：自動クリアを禁止 加算/平均モード選択：加算モード 加算回数：1 回変換
	データ格納バッファ設定	禁止
	ウィンドウ機能設定	禁止
	ウィンドウ A/B 動作設定	比較ウィンドウ A 有効：チェックを外す 比較ウィンドウ B 有効：チェックを外す
	入力サンプリング時間設定	AN001：3.667 (μs)
	イベントリンクコントロールビット設定	ELC 用スキャン終了イベント：すべてのスキャン終了時にイベント発生

表 15 スマート・コンフィグレータの設定（CMT1）

分類	項目	設定、説明
スマート・コンフィグレータ>> コンポーネント >> コンペアマッチタイマ （Config_CMT1）		下記の設定とし、コードを生成する。
	クロック設定	PCLK/8
	コンペアマッチ設定	インターバル時間：1000 μs (実際の値:1000) レジスタ（CMCOR）：3374 コンペアマッチ割り込みを許可（CMI1）：チェックを外す

表 16 スマート・コンフィグレータの設定（ELC）

分類	項目	設定、説明
スマート・コンフィグレータ>> コンポーネント >> イベントリンクコントローラ（Config_ELC）		下記の設定とし、コードを生成する。
	ソース	設定：Config_CMT1 リソース：CMT1 イベント：CMT1・コンペアマッチ 1
	送先	設定：Config_S12AD0 リソース：S12AD0 動作：A/D 変換開始

表 17 DSP ライブラリの設定

分類	項目	設定、説明
使用するファイル		DSP ライブラリの RX_DSP_Sample_CCRX_e²studio プロジェクトのフォルダにある以下のファイルを使用する。
	.lib ファイル	RX_DSP_FPU_LE.lib
	.h ファイル	r_dsp_complex.h r_dsp_transform.h r_dsp_typedefs.h r_dsp_types.h r_dsp_ver_info.h

4. 開発環境の入手

4.1 e²studio の入手方法

以下の URL にアクセスし、e² studio をダウンロードしてください。

https://www.renesas.com/ja-jp/products/software-tools/tools/ide/e2studio.html

なお、本ドキュメントは e² studio 2021-01 以降を使用することを前提としています。2021-01 よりも古い Ver.を使用した場合、e² studio の一部機能を使用できない可能性があります。ダウンロードする場合、ホームページに掲載されている最新 Ver.の e² studio を入手してください。

4.2 コンパイラパッケージの入手方法

以下の URL にアクセスして、RX ファミリ用 C/C++コンパイラパッケージをダウンロードしてください。

https://www.renesas.com/ja-jp/products/software-tools/tools/compiler-assembler/compiler-package-for-rx-family.html

5.　プロジェクトをインポートする方法

サンプルコードは e² studio のプロジェクト形式で提供しています。本章では、 e² studio へプロジェクトをインポートする方法を示します。インポート完了後、ビルドおよびデバッグの設定を確認してください。

5.1　e² studio での手順

e² studio でご使用になる際は、下記の手順で e² studio にインポートしてください。

(使用する e² studio のバージョンによっては画面が異なる場合があります。)

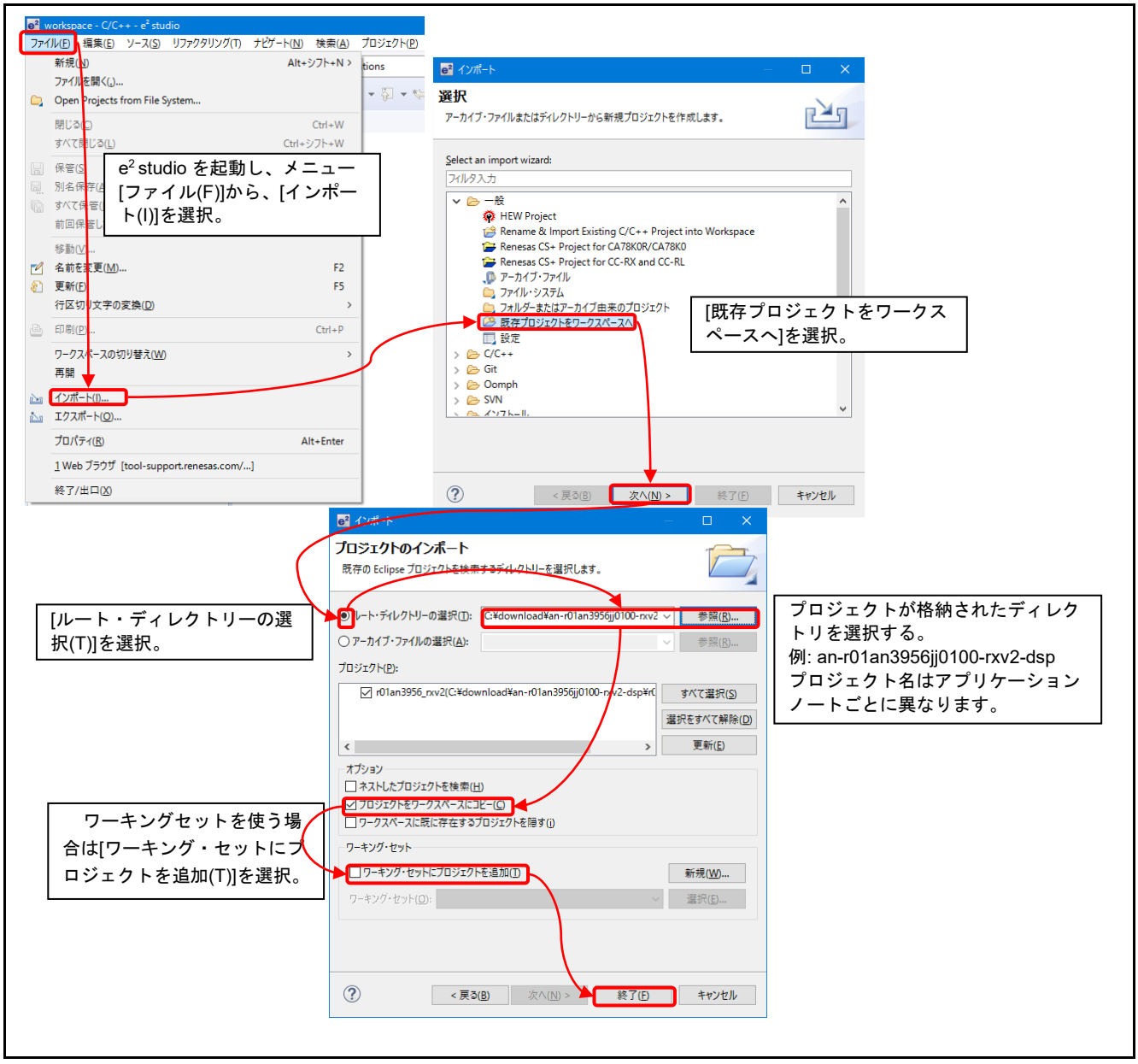


図 14　プロジェクトを e² studio にインポートする方法

5.2　CS+での手順

CS+でご使用になる際は、下記の手順で CS+にインポートしてください。

(使用する CS+のバージョンによっては画面が異なる場合があります。)

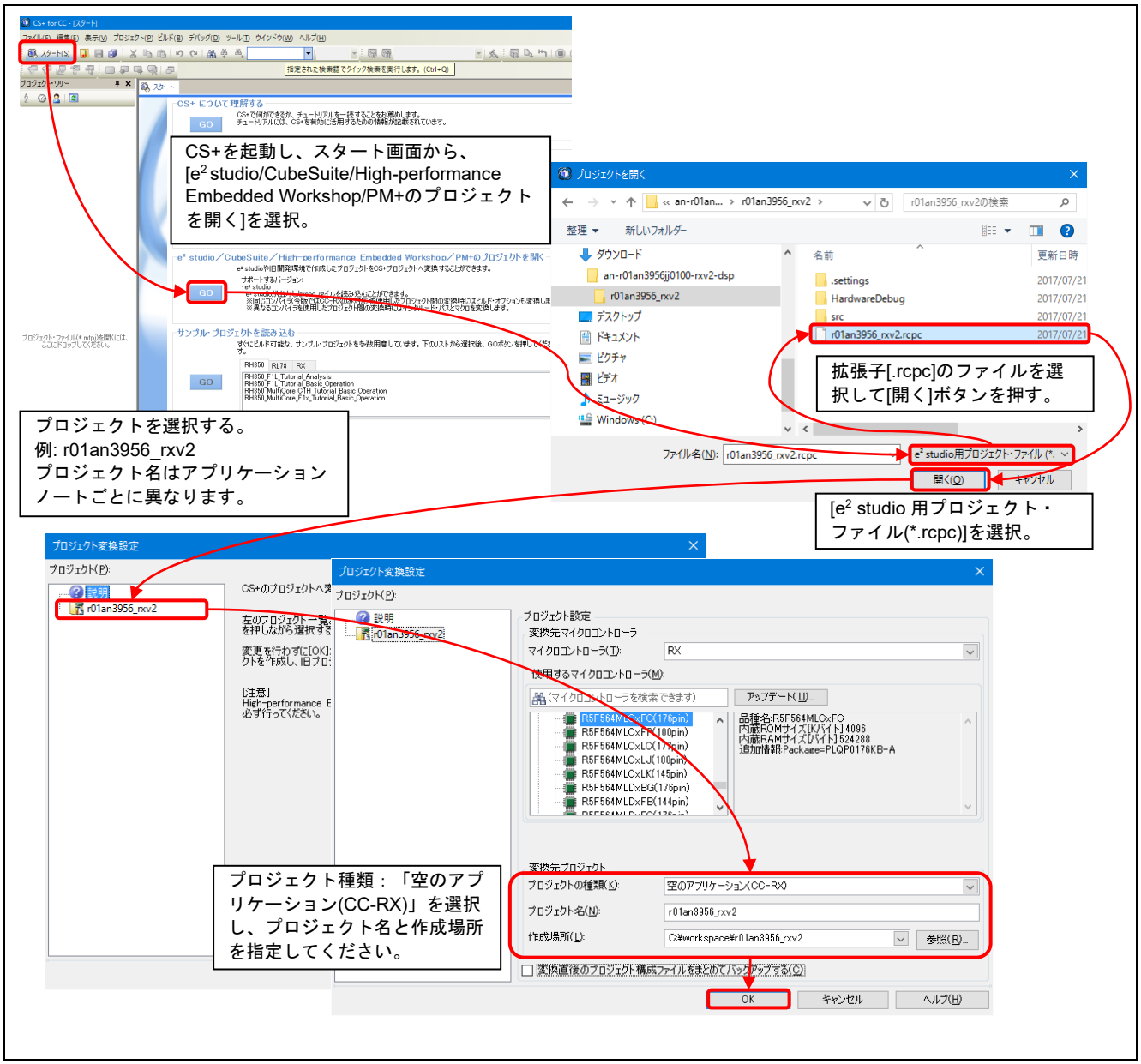


図 15　プロジェクトを CS+にインポートする方法

6. 補足

6.1 無償評価版の「RX ファミリ用 C/C++コンパイラパッケージ」を利用する場合の注意事項

無償評価版の「RX ファミリ用 C/C++コンパイラパッケージ」には、使用期限と使用制限があります。使用期限が過ぎた場合、リンクサイズが 128K バイト以内に制限されるためロードモジュールが正しく生成されなくなる場合があります。

詳しくは、ルネサスのホームページにある、無償版ソフトウェアツールのページを参照してください。

https://www.renesas.com/ja-jp/products/software-tools/evaluation-software-tools.html

7. 参考資料

- RX ファミリ ボードサポートパッケージモジュール Firmware Integration Technology (R01AN1685)
- RX ファミリ DMA コントローラ DMACA 制御モジュール Firmware Integration Technology (R01AN2063)
- e2 studio コード生成ツール ユーザーズマニュアル RX API リファレンス編 (R20UT2864)
- RX DSP ライブラリ Version 5.0 (CC-RX) (R01AN4359JJ0100)
- RX230 グループ、RX231 グループ ユーザーズマニュアル ハードウェア編 (R01UH0496)
- Renesas Starter Kit for RX231 ユーザーズマニュアル (R20UT3027)
- Renesas Starter Kit for RX231 CPU ボード回路図 (R20UT3026)

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。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Dec,15,17	-	初版発行
1.10	Apr,26,21	3	1. 仕様 DSP ライブラリのバージョンを更新 図 4 FFT 処理結果例　更新
		5	1.1 アプリケーションノートの構成 図 6 アプリケーションノートの構成　更新 図 7 サンプルプロジェクトのファルダ構成　更新
		6	1.2 ソフトウェアの構成 表 1 使用ソフトウェアモジュール一覧　更新
		7	1.3 動作環境 表 2 動作確認条件　更新
		13	2.4.2 FFT 処理 入力、出力データについて　説明追記 表 8 出力データと周波数の関係　誤記修正
		14	3.1 使用メモリ 更新 3.2 消費電流 更新 3.3 サンプリング周波数について 誤記修正
		15-16	3.4 ソフトウェアモジュールの設定 更新
		17	4.1 e²studio の入手方法 更新
		19	5.2 CS+での手順 追加
		20	7. 参考資料 更新
		プログラム	・動作環境（e²studio，コンパイラ、FIT モジュール、DSP ライブラリなど）の更新 ・DSP 処理の仕様を以下のように変更 - DSP ライブラリの更新にともない FFT 処理結果の周波数振幅特性の出力フォーマットを Q1.31 から Q2.30 に変更 - FFT 処理結果の周波数振幅特性の要素[0]の導出の方法を $R_{N/2}$ の絶対値を取得して Q2.30 フォーマットにするように変更 (r_dsp_real_fft.c)

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。

© 2021 Renesas Electronics Corporation. All rights reserved.