
RX111グループ

RTCAによるソフトウェアスタンバイモードからの復帰

Firmware Integration Technology

R01AN1821JJ0100
Rev. 1.00
2014.01.06

要旨

本アプリケーションノートでは、Firmware Integration Technology(FIT)モジュールを使用しています。RX111グループのリアルタイムクロック(RTC)を使って、一定時間ごとにソフトウェアスタンバイモードから復帰し、現在の時刻情報を取得する方法について説明します。

対象デバイス

- | | | |
|-------------|-------|------------------|
| ・RX111 グループ | 64ピン版 | ROM容量：16KB～128KB |
| ・RX111 グループ | 48ピン版 | ROM容量：16KB～128KB |
| ・RX111 グループ | 40ピン版 | ROM容量：16KB～64KB |

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. 仕様.....	3
2. 動作確認条件.....	4
3. 関連アプリケーションノート	4
4. ハードウェア説明.....	5
4.1 ハードウェア構成例	5
4.2 使用端子一覧.....	5
5. ソフトウェア説明.....	6
5.1 動作概要.....	7
5.2 ファイル構成.....	8
5.3 定数一覧.....	9
5.4 構造体/共用体一覧.....	10
5.5 関数一覧.....	11
5.6 関数仕様.....	12
5.7 フローチャート	18
5.7.1 メイン処理.....	18
5.7.2 ALM(アラーム)割り込みの処理.....	19
5.7.3 ソフトウェアスタンバイモード遷移前の処理.....	20
5.7.4 IRQ0 割り込み処理 (SW が押された後の処理)	20
6. プロジェクトをインポートする方法.....	21
6.1 e ² studio での手順.....	21
6.2 CubeSuite+での手順	22
7. 付録.....	23
7.1 コンパイラオプションの設定	23
8. サンプルコード	24
9. 参考ドキュメント.....	24

1. 仕様

RTC を使用して時刻の情報を得るとともに、ソフトウェアスタンバイモードから間欠的に復帰します。

リセット後、コールドスタートであれば、サブクロック発振器と RTC の初期設定を行います。その後、割り込み要求端子の入力レベルをモニタリングして、“L”であればソフトウェアスタンバイモードへ遷移します。以後、ソフトウェアスタンバイモードからは以下のどちらかで復帰します。

- ・1 分間ごとの ALM(アラーム)割り込みが発生した場合
- ・割り込み要求端子の入力レベルが再び“L”になった場合

割り込み要求端子の入力レベルが再び“L”になるまで、1 分間ごとに ALM 割り込みが発生し、ソフトウェアスタンバイモードから復帰します。その後、時刻情報を取得し、再びソフトウェアスタンバイモードへ遷移します。

制限事項

本アプリケーションノートでは、ウォームスタート時には対応していません。

RTC カウントソース	: サブクロック
VBATT 端子	: VCC 端子に接続
間欠復帰の周期	: 1 分

表 1.1 に使用する周辺機能と用途を示します。

なお、本アプリケーションノートでは、ソフトウェアスタンバイモード以外の動作時を「通常モード」と記載します。

表1.1 使用する周辺機能と用途

周辺機能	用途
RTCA	時計、およびソフトウェアスタンバイモードからの復帰
IRQ0	ソフトウェアスタンバイモードに遷移および復帰するための外部入力

2. 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2.1 動作確認条件

項目	内容
使用マイコン	R5F51115ADFM (RX111 グループ)
動作周波数	<ul style="list-style-type: none"> ● メインクロック : 16MHz ● サブクロック : 32.768kHz ● PLL : 32MHz (メインクロック 4 分周 8 逓倍) ● LOCO : 4MHz ● システムクロック (ICLK): 32MHz (PLL 1 分周) ● 周辺モジュールクロック B (PCLKB): 48MHz (PLL 1 分周)
動作電圧	3.3V
統合開発環境	ルネサスエレクトロニクス製 e2 studio Version 2.0.1.6
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family V.2.00 コンパイルオプション 6章の付録を参照してください。
iodefine.h のバージョン	V0.9A
エンディアン	リトルエンディアン
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
サンプルコードのバージョン	Version 1.00
使用する FIT モジュール	<ul style="list-style-type: none"> r_bsp : Version 2.30 r_cgc_rx : Version 1.00 r_irq_rx : Version 1.00 r_mpc_rx : Version 1.00 r_lpc_rx : Version 1.00 r_rtc_rx : Version 1.00
使用ボード	Renesas Starter Kit for RX111 (製品型名: R0K505111S001BE)

3. 関連アプリケーションノート

本アプリケーションノートに関連する FIT モジュールのドキュメントを以下に示します。併せて参照してください。

RX Family Board Support Package FIT module Rev.2.30 (R01AN1685EU0230)

RX111 Group CGC FIT module Rev.1.00 (R01AN1727EU0100)

RX111, RX210, RX63N Groups IRQ (External Pin Interrupt Request) FIT module Rev.1.00 (R01AN1668EU0100)

RX100 & RX200 Series Multi-Function Pin Controller FIT Modules Rev.1.00 (R01AN1724EU0100)

RX100 Series LPC FIT module Rev.1.00 (R01AN1728EU0100)

RX111 Group RTC FIT module Rev.1.00 (R01AN1817ES0100)

上記 FIT モジュールの API 関数を、本アプリケーションノートのサンプルコードで使用しています。Rev は本アプリケーションノート作成時点のものであります。

最新版がある場合、最新版に差し替えて使用してください。最新版はルネサスエレクトロニクスホームページで確認および入手してください。

4. ハードウェア説明

4.1 ハードウェア構成例

図4.1に接続例を示します。

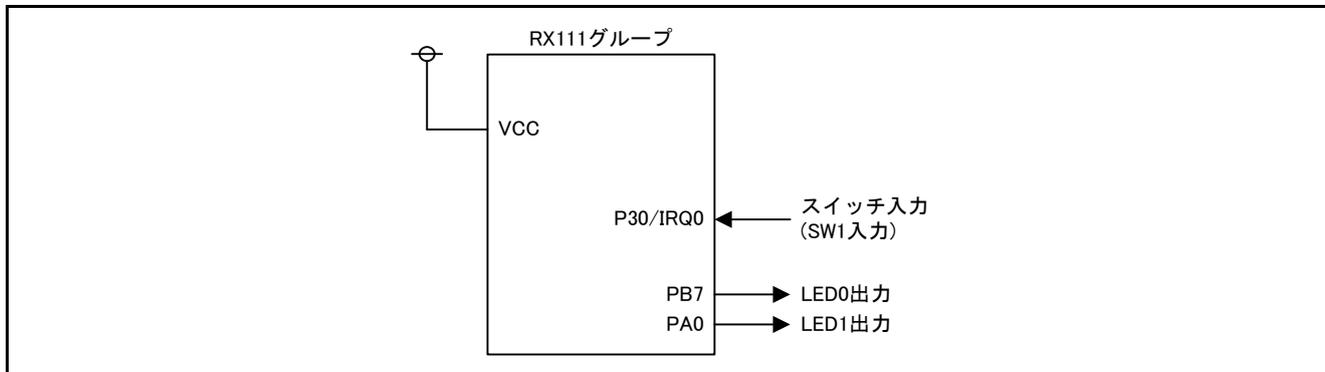


図4.1 接続例

4.2 使用端子一覧

表4.1に使用端子と機能を示します。

使用端子は64ピン版の製品を想定しています。64ピン版未満の製品を使用する場合は、使用する製品に合わせて端子を選択してください。

表4.1 使用端子と機能

端子名	入出力	内容
P30/IRQ0	入力	ソフトウェアスタンバイモードへ遷移／復帰のためのSW1入力
PB7	出力	LED0出力／通常モード時に点灯、ソフトウェアスタンバイ時に消灯
PA0	出力	LED1出力／1分間おきに反転

5. ソフトウェア説明

本サンプルコードでは、リセット後、IRQ0 割り込みの IR フラグをモニタリングし、奇数回目の IRQ0 割り込みでソフトウェアスタンバイモードへ遷移、偶数回目の IRQ0 割り込みで通常モードに復帰します。

また、ALM(アラーム)割り込みは通常モード、ソフトウェアスタンバイモードにかかわらず1分ごとに発生し、RTCの時刻情報の取得とアラーム時刻の更新を行います。ソフトウェアスタンバイモードからALM(アラーム)割り込みで復帰した場合は、再び遷移します。

使用する周辺機能の設定を表 5.1、表 5.2に示します。

表5.1 RTC の設定

設定名	内容
カウントソース	サブクロックで動作
初期設定時刻	2013 年 1 月 1 日(火) 00 時 00 分 00 秒
時間モード	24 時間モード
RTCOUT 出力	使用禁止
誤差補正	使用しない
割り込み	アラーム割り込み(ALM) : 発生周期 1 分

表5.2 IRQ0 の設定

設定名	内容
検出方法	立ち下がリエッジ検出
デジタルフィルタ	有効
デジタルフィルタの サンプリングクロック	PCLK/64
割り込み	外部割り込み(IRQ0)使用

5.1 動作概要

(1) 初期設定

リセット後、RSTSR1.CWSF フラグが“0” (コールドスタート)であれば、サブクロック発振器の動作、RTCの初期設定、を行います。RTCの初期設定では、時刻とアラーム時間を設定し、ALM(アラーム)割り込み要求を許可します。さらに、ソフトウェアスタンバイモードへの遷移のための設定を行い、動作LED0、動作LED1を点灯します。その後、1分ごとにALM(アラーム)割り込みが発生し時刻情報の取得とアラーム情報の更新、動作LED1の反転を行います。

また、IRQ0割り込みのIRフラグをモニタリングします。

(2) ソフトウェアスタンバイモードへの遷移

IRQ0割り込みのIRフラグが“1”になると、動作LED0を消灯し、WAIT命令を実行してソフトウェアスタンバイモードへ遷移します。

(3) ソフトウェアスタンバイモードからの復帰

ALM割り込みが発生した場合、もしくはIRQ0割り込みのIRフラグをモニタリングし、偶数回目のIRQ0割り込みが発生した場合にソフトウェアスタンバイモードから復帰します。

(3-1) ALM割り込みによる復帰

復帰後は、動作LED0を点灯、動作LED1を反転、1/128秒経過後にRTCの時刻情報を読み出します。さらに、IRQ0割り込みのIRフラグが“1”でない場合は、再びソフトウェアスタンバイモードに遷移します。偶数回目のIRQ0割り込みが発生するまで(3)を繰り返します。

(3-2) IRQ0割り込みによる復帰

復帰後は、動作LED0を点灯します。

図 5.1にタイミング図を示します。

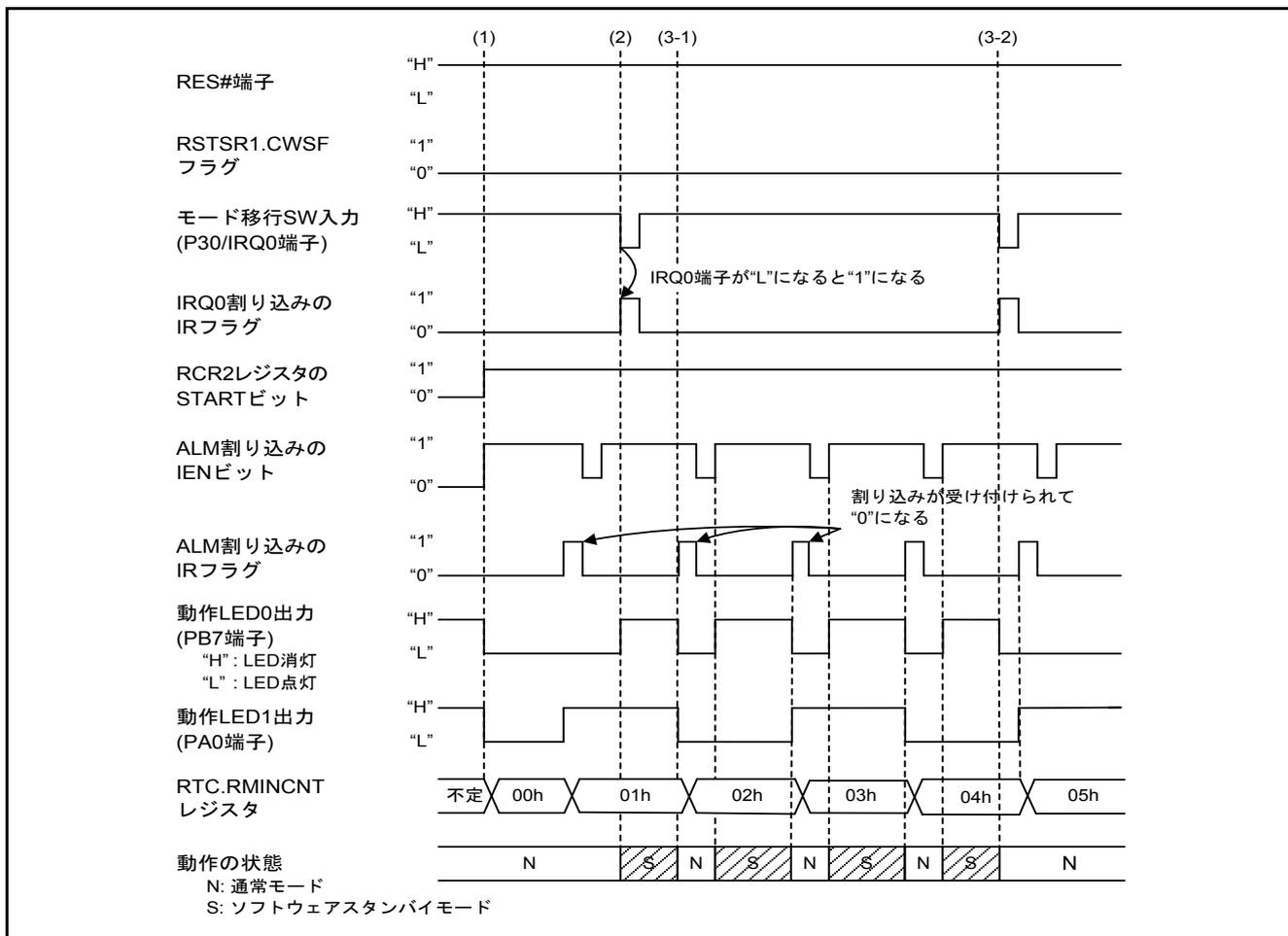


図 5.1 タイミング図

5.2 ファイル構成

表5.3にサンプルコードで使用するファイル、表5.4 サンプルコードで使用するFIT モジュールを示します。
なお、統合開発環境で自動生成されるファイルは除きます。

表5.3 サンプルコードで使用するファイル

ファイル名	概要	備考
main.c	メイン処理	

表5.4 サンプルコードで使用するFIT モジュール

モジュール名	概要	備考
CGC モジュール	クロック関連モジュール	クロック設定、変更を使用
IRQ モジュール	外部割り込み関連モジュール	IRQ0 割り込みに使用
MPC モジュール	マルチファンクションピン端子関連モジュール	SW1 のピン設定に使用
RTC モジュール	リアルタイムクロック関連モジュール	時刻、アラーム設定に使用
LPC モジュール	低消費電力関連モジュール	ソフトウェアスタンバイ モード遷移/復帰に使用

5.3 定数一覧

表 5.5～表 5.8にサンプルコードで使用する定数を示します。本サンプルコードに関連する定数を記載しています。以下以外の定数の設定値は、デフォルト値です。

表5.5 サンプルコードで使用する定数：BSP (r_bsp_config.h)

定数名	設定値	内容
BSP_CFG_USE_CGC_MODULE	0	クロックの初期設定に CGC モジュール使用しない
BSP_CFG_CLOCK_SOURCE	4	立ち上がり時のシステムクロック：PLL
BSP_CFG_XTAL_HZ	1600000	メインクロックの周波数
BSP_CFG_PLL_DIV	4	PLL の分周
BSP_CFG_PLL_MUL	8	PLL の通倍
BSP_CFG_ICK_DIV	1	ICLK の分周
BSP_CFG_PCKB_DIV	1	PCLKB の分周
BSP_CFG_PLKD_DIV	1	PCLKD の分周
BSP_CFG_FCK_DIV	1	FCLK の分周

表5.6 サンプルコードで使用する定数：CGC モジュール (r_cgc_rx111_config.h)

定数名	設定値	内容
CGC_CFG_SUBCLOCK_DRIVE	0x02	ドライブ能力 低

表5.7 サンプルコードで使用する定数：RTC モジュール (r_rtc_rx_config.h)

定数名	設定値	内容
RTC_CFG_CALCULATE_YDAY	0	年：省略する

表5.8 サンプルコードで使用する定数：IRQ モジュール (r_irq_rx_config.h)

定数名	設定値	内容
IRQ_CFG_USE_IRQ0	1	IRQ0：使用
IRQ_PORT_IRQ0	PORT3	IRQ0 のポート：ポート 3 設定
IRQ_PORT_BIT_IRQ0	IRQ_BIT0	IRQ0 のビット：ビット 0 設定
IRQ_CFG_FILT_EN_IRQ0	1	IRQ0 のデジタルフィルタ：有効
IRQ_CFG_FILT_PLCK_IRQ0	IRQ_CFG_PCL K_DIV64	IRQ0 のデジタルフィルタのサンプリングクロック： 64 分周

5.4 構造体/共用体一覧

図 5.2～図 5.4に RTC モジュールの関数で使用する構造体を示します。また、図 5.5に MPC モジュールで使用する構造体を示します。

```
typedef struct
{
    int tm_sec;        /* Seconds (0-59) */
    int tm_min;        /* Minute (0-59) */
    int tm_hour;       /* Hour (0-23) */
    int tm_mday;       /* Day of the month (1-31) */
    int tm_mon;        /* Month (0-11, 0 = January) */
    int tm_year;       /* Year since 1900 */
    int tm_wday;       /* Day of the week (0-6, 0 = Sunday) */
    int tm_ydat;       /* Day of the year (0-365) */
    int tm_isdst;      /* Day Light Saving enable (>0), disabled (=0), or unknown (<0) */
} tm_t;
```

図 5.2 時刻設定、アラーム時間設定 (R_RTC_Open 関数、R_RTC_Control 関数)で使用する構造体

```
typedef struct
{
    uint32_t    configuration;
    uint16_t    periodic;
    uint8_t     int_priority_alm;    // alm INT priority; 1 = low, 15 = high
    uint8_t     int_priority_prd;    // prd INI priority; 1 = low, 15 = high
    RTCCallBackFunc p_callback;
} rtc_init_t;
```

図 5.3 RTC の初期設定 (R_RTC_Open 関数)で使用する構造体

```
typedef struct
{
    uint32_t configuration;
    uint16_t periodic;
    uint16_t update_selection;
} rtc_cfg_t;
```

図 5.4 RTC の情報の更新(R_RTC_Control 関数)で使用する構造体

```
typedef struct
{
    uint8_t pin_function; //which peripheral function is assigned to this pin
    bool irq_enable //This pin is used as IRQ pin.
    bool analog_enable //This pin is used as ADC input, DAC output, or for LVD (CMPA2)
} mpc_config_t
```

図 5.5 外部割り込みピンの設定 (R_MPC_Write 関数)で使用する構造体

5.5 関数一覧

表 5.9にサンプルコードで使用する関数を示します。

表5.9 サンプルコードで使用する関数

関数名	概要
main	メイン処理
rtc_callback	ALM (アラーム) 割り込み時の処理
lpc_callback	ソフトウェアスタンバイモード遷移前の処理
irq_callback	IRQ0 割り込み処理
R_CGC_Open	CGC の初期設定 (サブクロックのドライブ能力設定)
R_CGC_ClockStart	クロックの発振開始
R_RTC_Open	RTC の初期設定
R_RTC_Control	RTC 情報の更新
R_RTC_Read	RTC 情報の読み出し
R_MPC_Write	外部割り込みのピン設定
R_IRQ_Open	IRQ の初期設定
R_LPC_LowPowerModeConfigure	低消費電力モード遷移のための設定
R_LPC_LowPowerModeActivate	低消費電力モードへの遷移

5.6 関数仕様

サンプルコードの関数仕様を示します。

main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	サブクロックの発振、RTCの初期設定、外部割り込みの初期設定後、ソフトウェアスタンバイモード遷移のための設定を行い、動作LED0、動作LED1を点灯します。その後SW1が押下されるとソフトウェアスタンバイモードへ遷移します。また、通常モード、ソフトウェアスタンバイモードに関わらず1分ごとにALM割り込みが発生し、RTCの時刻情報を読み出し、アラーム時間の更新を行います。ソフトウェアスタンバイモードからALM割り込みで復帰した場合は、再びソフトウェアスタンバイモードへ遷移します。
引数	なし
リターン値	なし
rtc_callback	
概要	ALM (アラーム) 割り込み時の処理
ヘッダ	r_rtc_rx_if.h
宣言	void rtc_callback (void * p_flag)
説明	ALM 割り込み発生後の処理を行います。処理は、LED1を反転し、ソフトウェアスタンバイモード遷移中にALM割り込みによって復帰した場合はLED0を点灯、1/128秒ソフトウェアループします。その後、RTC情報を読み出し、アラーム時間を1分後に更新します。
引数	p_flag ALM 割り込みもしくは周期割り込みのどちらのが発生したかを示すポインタ
リターン値	なし
lpc_callback	
概要	ソフトウェアスタンバイモード遷移前の処理
ヘッダ	r_lpc_rx_if.h
宣言	void lpc_callback (void * pdata)
説明	LED0を消灯し、ソフトウェアスタンバイ遷移フラグ (ソフトウェアスタンバイモード中かどうかを判断するフラグ)を1に設定します。
引数	pdata 低消費電力モードに遷移する前に呼び出す関数のポインタ
リターン値	なし
irq_callback	
概要	IRQ0 割り込み処理
ヘッダ	r_irq_rx_if.h
宣言	void irq_callback (void * pargs)
説明	SW1が押下された場合、SW1が押下された回数のカウンタをインクリメントし、LED0を点灯、ソフトウェアスタンバイ遷移フラグを“0”に設定します。
引数	pargs コールバック関数のポインタ
リターン値	なし

R_CGC_Open	
概要	CGCの初期設定 (サブクロックのドライブ能力設定)
ヘッダ	なし
宣言	cgc_err_t R_CGC_Open(void)
説明	CGCの初期設定として、サブクロックの停止、サブクロックのドライブ能力の設定、HOCOの発振安定待ち時間の設定、メインクロックの発振待ち時間の設定、メインクロックのドライブ能力の設定を行います。
引数	なし
リターン値	成功値もしくはエラー値
備考	CGC_SUCCESS : ドライブ能力設定が完了した場合 本サンプルコードでは、サブクロックのドライブ能力に“ドライブ能力 低”を設定します。本関数の詳細は、アプリケーションノートRX111 Group CGC FIT module Rev.1.00 (R01AN1727EU0100)」を参照してください。
R_CGC_ClockStart	
概要	クロックの発振開始
ヘッダ	なし
宣言	cgc_err_t R_CGC_(cgc_clock_t clock_source, cgc_clock_config_t* pClock_config)
説明	第一引数で渡されたクロックの発振を開始します。
引数	cgc_clock_t clock_source : 発振するクロック cgc_clock_config_t* pClock_config : PLLの分周、逡倍
リターン値	成功値もしくはエラー値
備考	CGC_SUCCESS : 問題なく発振が開始した場合 CGC_ERR_NOT_STABILIZED : クロックが発振されていない場合 CGC_ERR_CLOCK_ACTIVE : クロックがすでに発振している場合 CGC_MAIN_OCO_INACTIVE : PLL発振時にメインクロックが停止している場合 CGC_ERR_ILL_PARAM : パラメータエラーの場合 CGC_ERR_NULL_PTR : 第二引数がNULLの場合 本サンプルコードでは、サブクロックを発振します。本関数の詳細は、アプリケーションノートRX111 Group CGC FIT module Rev.1.00 (R01AN1727EU0100)」を参照してください。

R_RTC_Open

概要	RTCの初期設定
ヘッダ	r_rtc_if.h
宣言	rtc_err_t R_RTC_Open(rtc_init * p_init tm_t * p_current tm_t * p_alarm)
説明	RTCの初期設定として、RTCの初期化、時刻の初期設定、アラーム時刻の設定を行います。
引数	rtc_init * p_init : 初期設定の構造体のポインタ tm_t * p_current : 時刻設定の構造体のポインタ tm_t * p_alarm : アラーム時刻設定の構造体のポインタ
リターン値	成功値もしくはエラー値 RTC_SUCCESS : RTCの初期設定成功 RTC_ERR_INIT : RTCがすでにオープンしている場合 RTC_ERR_SUBCLOCK : サブクロックが動作していない場合 RTC_ERR_ILL_PARAM : パラメータエラーの場合 RTC_ERR_ILL_CALLBACK : コールバック関数が不適切な場合 RTC_ERR_ILL_PRIORITY : 割り込みの優先レベルが不適切な場合 RTC_ERR_ILL_CURRENT : 設定時刻が不適切な場合 RTC_ERR_ILL_ALARM : アラーム時刻が不適切な場合
備考	本サンプルコードでは、第一引数で“分”のアラームを有効、周期割り込みは禁止、アラーム割り込みの優先レベルは“15”に設定します。また第二引数の時刻は2013年1月1日(火)00時00分00秒、第三引数のアラーム時刻は2013年1月1日(火)00時01分00秒を設定します。本関数の詳細は、アプリケーションノートRX111 Group RTC FIT module Rev.1.00 (R01AN1817ES0100)を参照してください。

R_RTC_Control

概要	RTC情報の更新
ヘッダ	r_rtc_if.h
宣言	rtc_err_t R_RTC_Control(rtc_cgc_t * p_cfg tm_t * p_current tm_t * p_alarm)
説明	RTCの設定情報を更新します。
引数	rtc_cgc_t * p_cfg : RTC設定情報の構造体のポインタ tm_t * p_current : 時刻設定の構造体のポインタ tm_t * p_alarm : アラーム時刻設定の構造体のポインタ
リターン値	成功値もしくはエラー値 RTC_SUCCESS : RTCの初期設定成功 RTC_ERR_UNINIT : RTCが初期化されていない場合 RTC_ERR_ILL_PARAM : パラメータエラーの場合 RTC_ERR_ILL_CURRENT : 設定時刻が不適切な場合 RTC_ERR_ILL_ALARM : アラーム時刻が不適切な場合
備考	本サンプルコードでは、RTCの時刻情報読み出した後、時刻設定は変更しないので第二引数には“FIT_NO_PTR”を設定します。第三引数にはアラームの時刻を1分更新した構造体のポインタを渡します。本関数の詳細は、アプリケーションノートRX111 Group RTC FIT module Rev.1.00 (R01AN1817ES0100)を参照してください。

R_RTC_Read	
概要	RTC情報の読み出し
ヘッダ	r_rtc_if.h
宣言	rtc_err_t R_RTC_Read(int8_t * status, tm_t * p_current tm_t * p_alarm)
説明	その時点の時刻とアラーム時刻を読み出します。
引数	int8_t * status : RTCのカウンタのステータス (停止/動作)の格納先 tm_t * p_current : その時点の時刻情報の格納先のポインタ tm_t * p_alarm : アラーム時刻の格納先のポインタ
リターン値	成功値もしくはエラー値 RTC_SUCCESS : RTCの初期設定成功 RTC_ERR_UNINIT : RTCが初期化されていない場合 RTC_ERR_READ : 読み出せるサイズを超えた場合
備考	本サンプルコードでは、RTCの時刻情報を読み出す際に使用します。RTCのカウンタのステータス、時刻情報、アラーム時刻を全て読み出すので、それぞれの格納先のポインタを渡します。本関数の詳細は、アプリケーションノート「RX111 Group RTC FIT module Rev.1.00 (R01AN1817ES0100)」を参照してください。

R_MPC_Write	
概要	外部割り込みのピン設定
ヘッダ	r_mpc_rx_if.h
宣言	rtc_err_t R_MPC_Write (gpio_port_pin_t pin, mpc_config_t * pconfig)
説明	使用する外部割り込みの機能選択、IRQ入力端子として使用するか、アナログ端子として使用するかを設定します。
引数	gpio_port_pin_t pin : 使用するピン mpc_config_t * pconfig : 使用するピンの情報の格納先のポインタ
リターン値	成功値もしくはエラー値 MPC_SUCCESS : 外部割り込みのピン設定が問題なく完了した場合 MPC_ERR_INVALID_CFG : 入力値が無効の場合
備考	本サンプルコードでは、SW1を使用するため第一引数には、SW1のP30を設定します。SW1はIRQ入力端子として使用するため、第二引数の機能選択は“0”(Hi-zを設定)、IRQ入力割り込みとしては“true”(使用する)、アナログ端子として“false”(使用しない)を設定します。 本関数の詳細は、アプリケーションノート「RX100 & RX200 Series Multi-Function Pin Controller FIT Modules Rev.1.00 (R01AN1724EU0100)」を参照してください。

R_IRQ_Open	
概要	IRQの初期設定
ヘッダ	r_irq_rx_if.h
宣言	irq_err_t R_RTC_Open (uint8_t irq_number irq_trigger_t trigger, irq_prio_t priority irq_handle_t *phandle void (*const pcallback)(void *pargs))
説明	IRQの初期設定として、割り込みのトリガレベル、優先度などを設定します。
引数	uint8_t irq_number : 初期化するIRQ irq_trigger_t trigger : 使用するトリガレベル irq_prio_t priority : 割り込みの優先レベル irq_handle_t *phandle : IRQ設定の構造体のポインタ void (*const pcallback)(void *pargs) : 割り込み復帰からのコールバック関数のポインタ
リターン値	成功値もしくはエラー値 IRQ_SUCCESS : IRQの初期設定成功 IRQ_ERR_BAD_NUM : IRQ番号が不適切だった場合 IRQ_ERR_NOT_CLOSED : 先にR_IRQ_Close()が実行されている場合 IRQ_ERR_NULL_PTR : 設定ポインタがNULLだった場合 IRQ_ERR_INVALID_ARG : パラメータエラーの場合 IRQ_ERR_LOCK : 選択されたIRQ番号にロックをかけることができない場合
備考	本サンプルコードでは、第一引数に“0”、第二引数にLowレベル検出、第三引数に割り込みレベルは“15”に設定します。 本関数の詳細は、アプリケーションノート「RX111, RX210, RX63N Groups IRQ (External Pin Interrupt Request) FIT module Rev.1.00 (R01AN1668EU0100)」を参照してください。

R_LPC_LowPowerModeConfigure	
概要	低消費電力モード遷移のための設定
ヘッダ	r_lpc_rx111_if.h
宣言	lpc_err_t R_LPC_LowPowerModeConfigure (lpc_low_power_mode_t e_mode)
説明	低省電力モードに遷移するための設定を行います。
引数	r_lpc_low_power_mode_t e_mode : 遷移する低消費電力モード
リターン値	成功値もしくはエラー値 LPC_SUCCESS : 低消費電力モード遷移のための設定成功 LPC_ERR_ILL_PARAM : パラメータエラーの場合
備考	本サンプルコードでは、第一引数に“LPC_LP_SW_STANDBY”（ソフトウェアスタンバイモード）を設定します。 本関数の詳細は、アプリケーションノート「RX100 Series LPC FIT module Rev.1.00 (R01AN1728EU0100)」を参照してください。

R_LPC_LowPowerModeActivate	
概要	低消費電力モードへの遷移
ヘッダ	r_lpc_rx111_if.h
宣言	lpc_err_t R_LPC_LowPowerModeActivate((*pcallback)(void *pdata))
説明	WAIT 命令を実行し、低消費電力モードに遷移します。
引数	(*pcallback)(void *pdata) : 遷移する前の処理を行うコールバック関数のポインタ
リターン値	成功値もしくはエラー値 LPC_SUCCESS : 低消費電力モード遷移成功 LPC_ERR_OSC_STOP_ENABLED : ソフトウェアスタンバイモードへ遷移する際に、発振停止検出が許可になっていた場合 LPC_ERR_ILL_CLOCK_SOURCE : -スリープモードに遷移する場合: 復帰後のクロックにサブクロックが選択されていた場合 -スリープモードから復帰後のクロックが HOCO のとき、復帰後に中速モードになる場合 -スリープモードから復帰後のクロックがメインクロックかつ中速モードの際に、システムクロック(ICLK)が 8MHz もしくは 12MHz になっている場合
備考	本サンプルコードでは、第一引数にコールバック関数を設定します。 本関数の詳細は、アプリケーションノート「RX100 Series LPC FIT module Rev.1.00 (R01AN1728EU0100)」を参照してください。

5.7 フローチャート

5.7.1 メイン処理

図 5.6にメイン処理のフローチャートを示します。

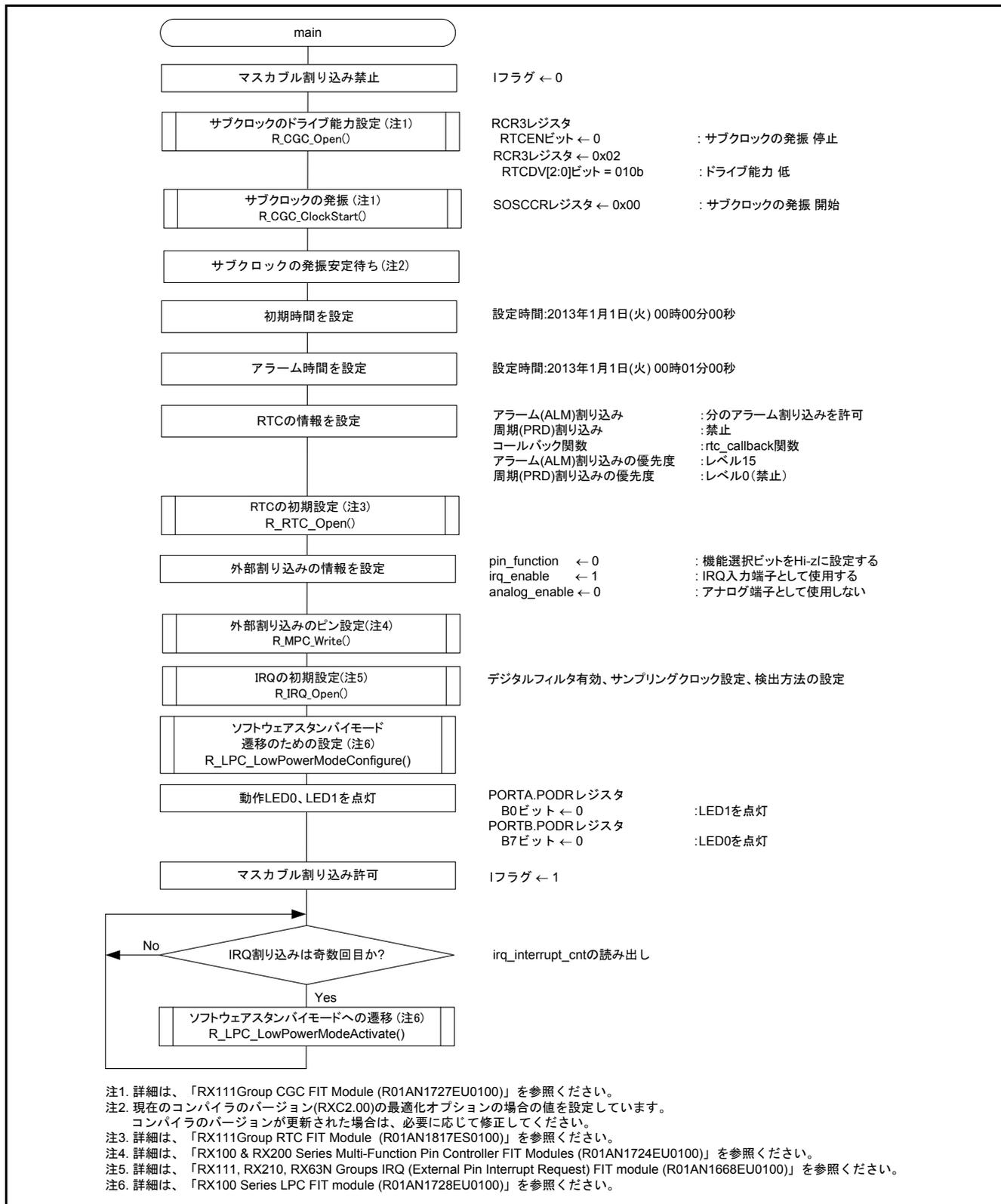


図 5.6 メイン処理

5.7.2 ALM(アラーム)割り込みの処理

図 5.7に RTC の ALM 割り込み処理ルーチンのフローチャートを示します。

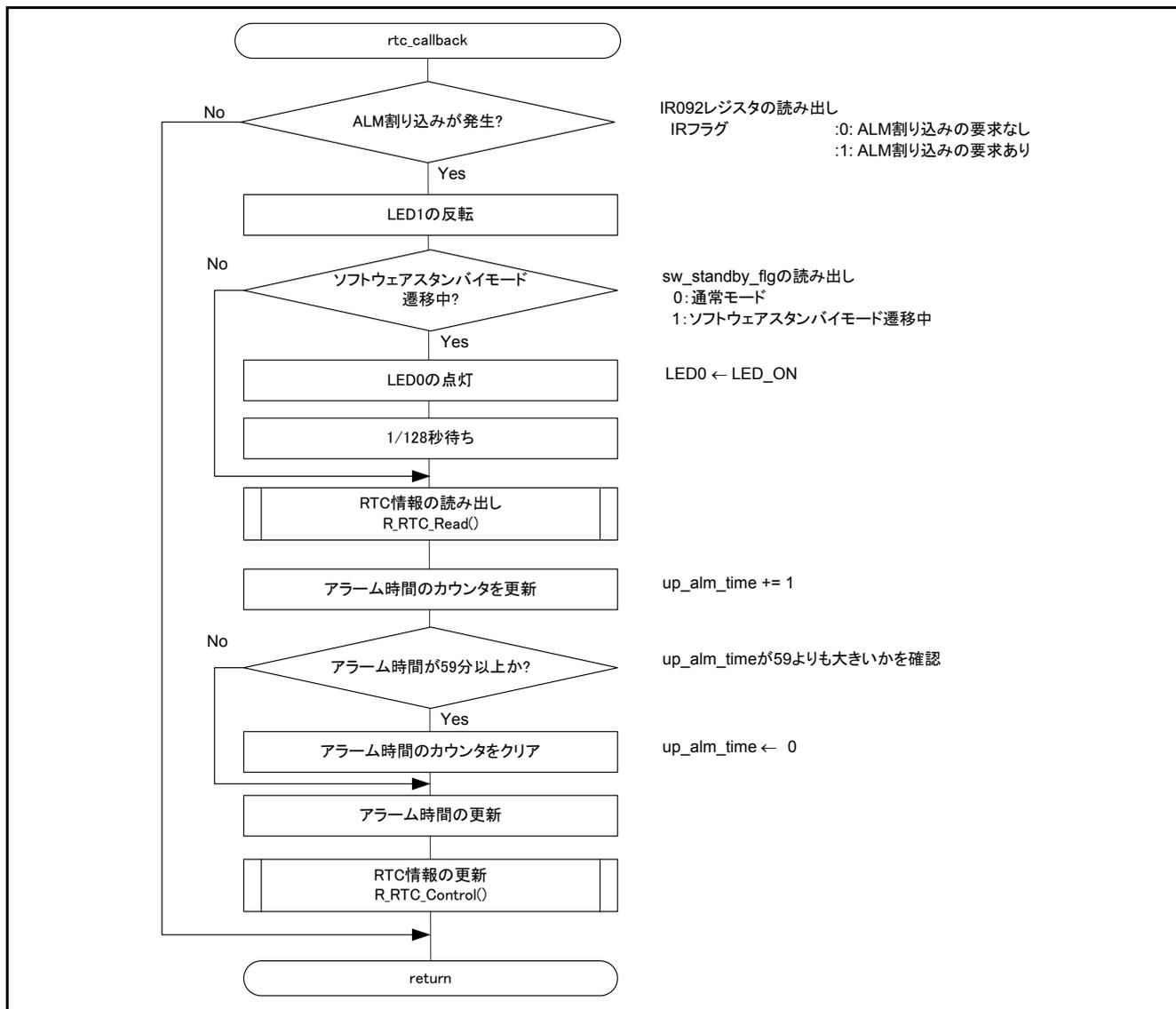


図 5.7 ALM(アラーム)割り込みの処理

5.7.3 ソフトウェアスタンバイモード遷移前の処理

図 5.8にソフトウェアスタンバイモード遷移前の処理のフローチャートを示します。

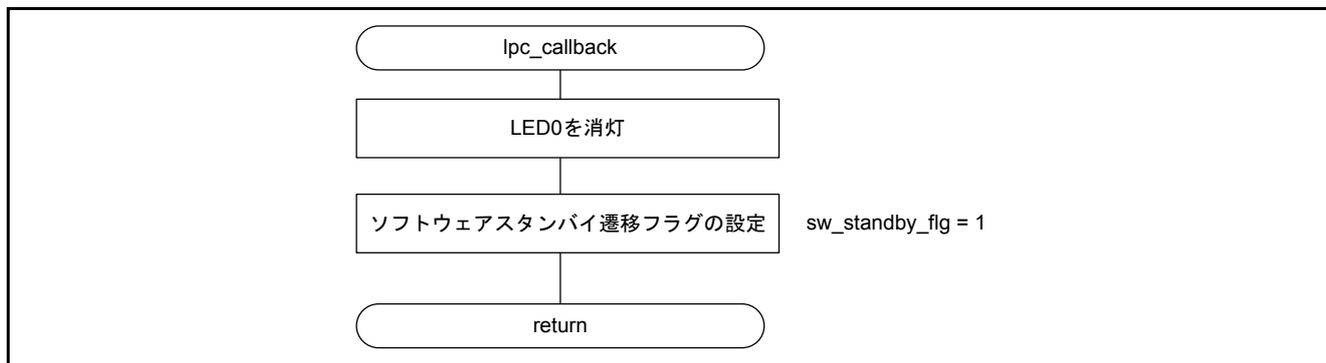


図 5.8 ソフトウェアスタンバイモード遷移前の処理

5.7.4 IRQ0 割り込み処理 (SW が押された後の処理)

図 5.9に IRQ0割り込み処理のフローチャートを示します。

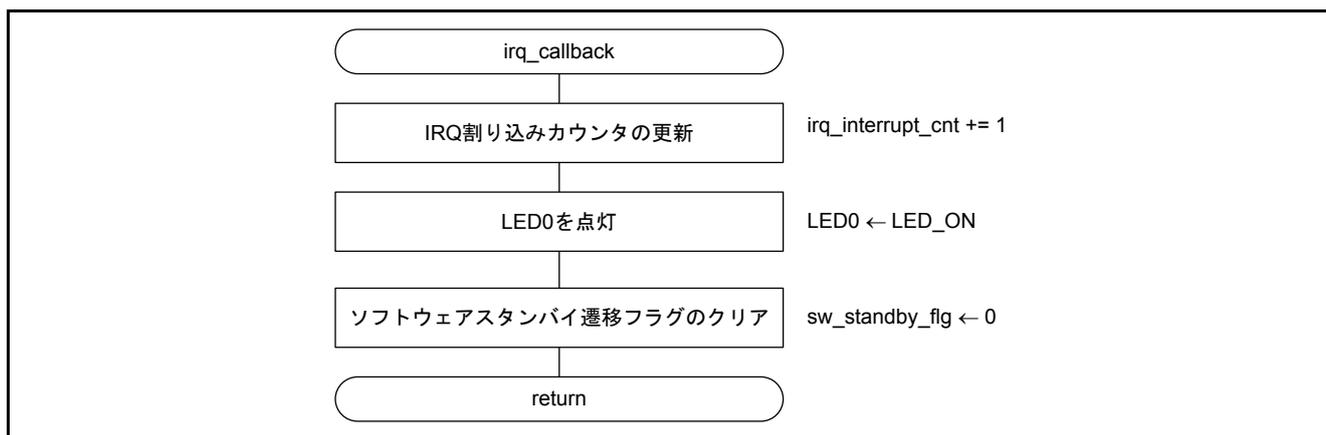


図 5.9 IRQ0 割り込み処理

6. プロジェクトをインポートする方法

6.1 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。

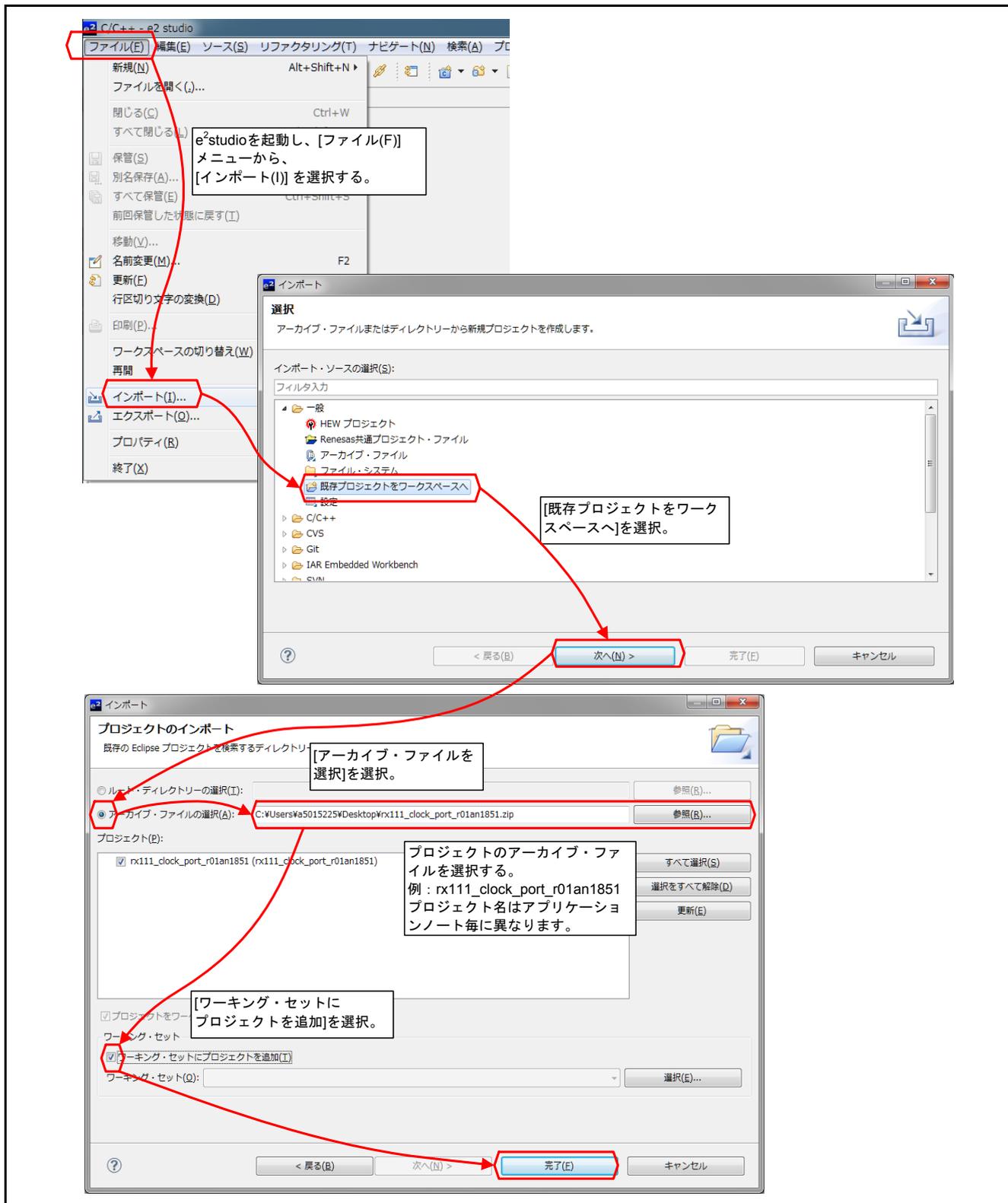


図6.1 プロジェクトを e² studio にインポートする方法

6.2 CubeSuite+での手順

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

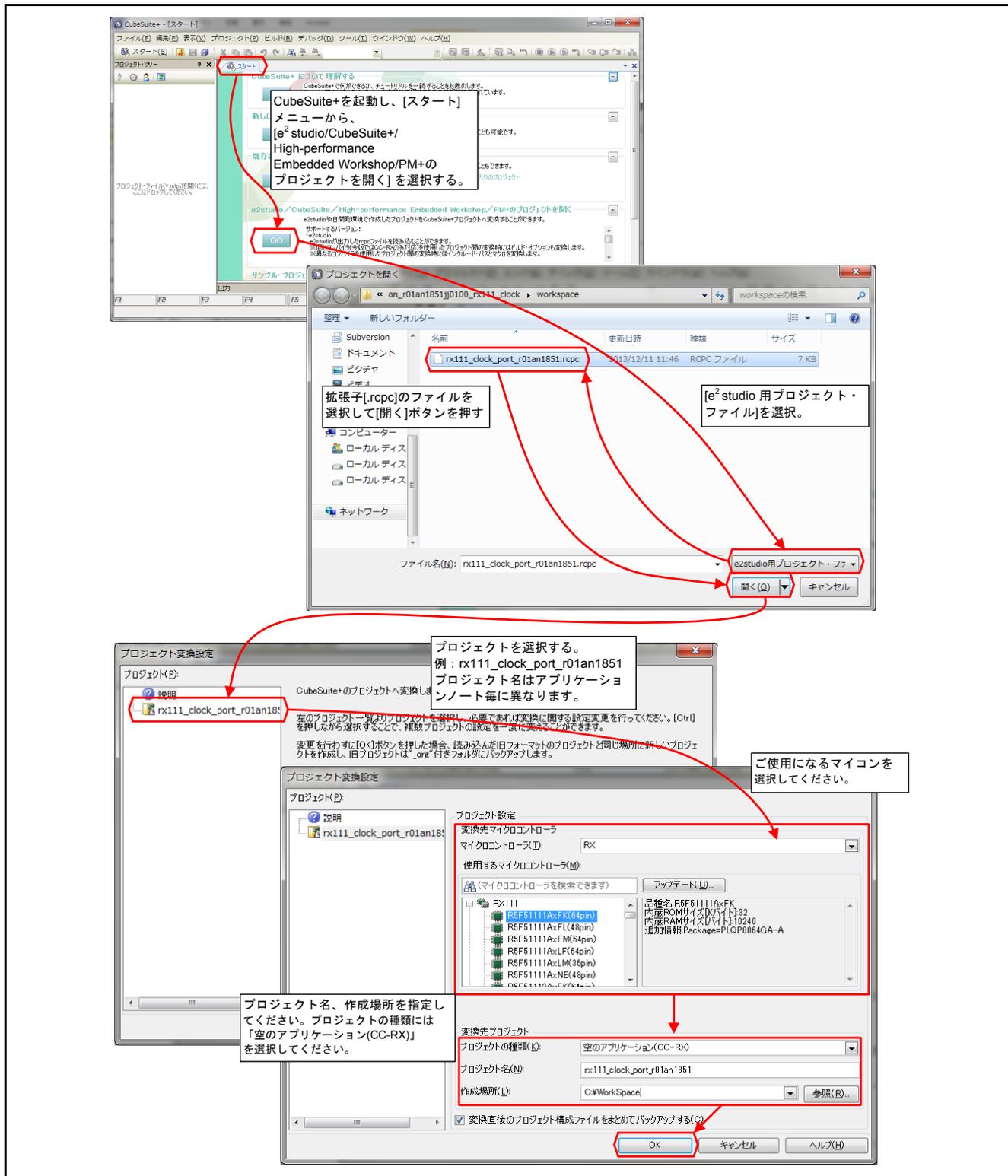


図 6.1 プロジェクトを CubeSuite+にインポートする方法

7. 付録

7.1 コンパイラオプションの設定

図 7.1 にコンパイラオプションのパスを示します。パスは以下のように設定してください。またセクションの割り付けについては、RX Family Board Support Package FIT module Rev.2.30(R01AN1685EU0230)の6章を参照してください。

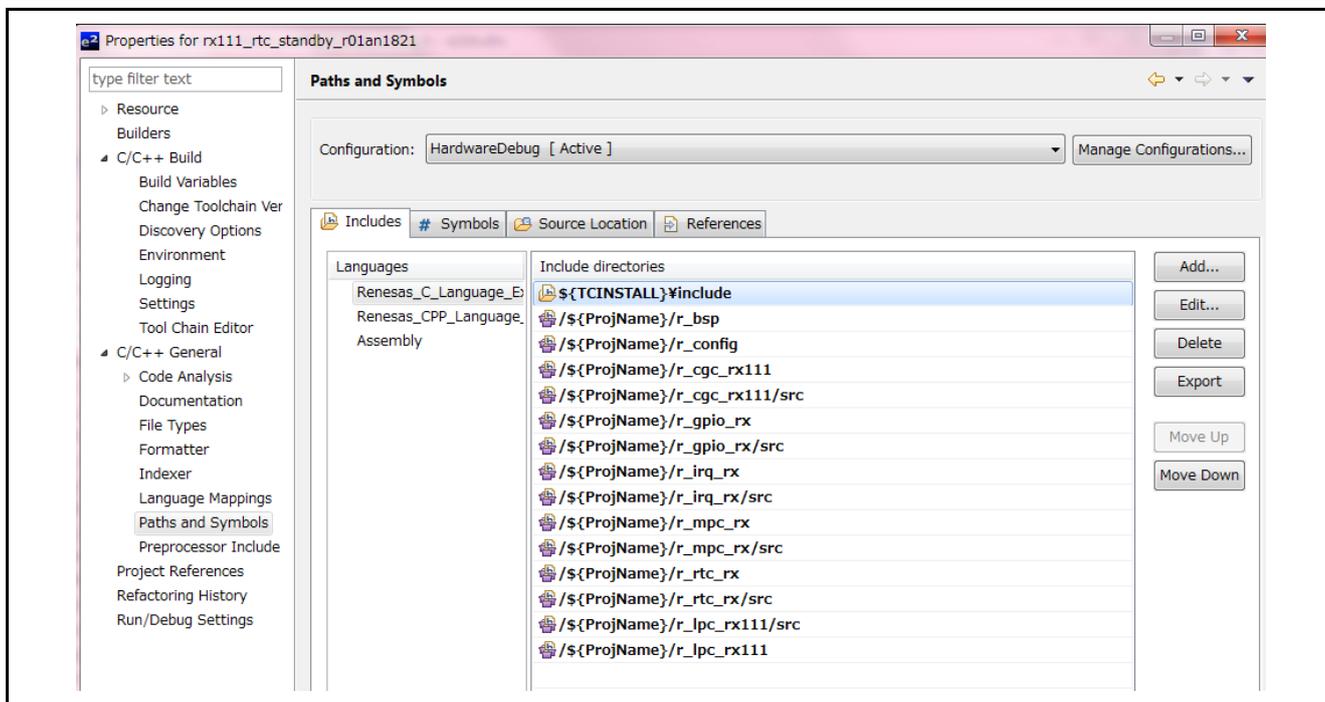


図 7.1 コンパイラオプションのパス設定

8. 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。

9. 参考ドキュメント

ユーザーズマニュアル：ハードウェア

RX111グループ ユーザーズマニュアル ハードウェア編 Rev.1.00 (R01UH0365JJ)
(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。)

ユーザーズマニュアル：開発環境

RX ファミリー C/C++コンパイラパッケージ V.2.00 ユーザーズマニュアル Rev.1.00 (R20UT0570JJ)
(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。)

ホームページとサポート窓口

ルネサス エレクトロニクスホームページ

<http://japan.renesas.com>

お問い合わせ先

<http://japan.renesas.com/contact/>

改訂記録	RX111グループ アプリケーションノート RTCAによるソフトウェアスタンバイモードからの復帰
------	---

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2014.01.06	—	初版発行

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。

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. 未使用端子の処理

【注意】未使用端子は、本文の「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。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】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. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。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。

4. クロックについて

【注意】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。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。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。

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。お客様の機器・システムの設計において、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には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 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に関し、当社は、何らの責任を負うものではありません。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. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製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。
5. 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）、交通用信号機器、
防災・防犯装置、各種安全装置等
当社製品は、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（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障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。
7. 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、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、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。
8. 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い合わせください。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。お客様がかかる法令を遵守しないこと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。
9. 本資料に記載されている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。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍用用途に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 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。

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とその総株主の議決権の過半数を直接または間接に保有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。



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/contact/>