RX ファミリ

MPC モジュール Firmware Integration Technology

要旨

本アプリケーションノートは、Firmware Integration Technology (FIT)を使用した MPC モジュールについて説明します。本モジュールを使って、マルチファンクションピンコントローラ（MPC）をシステムに組み込んでご使用いただけます。以降、本モジュールを MPC FIT モジュールと称します。

対象デバイス

- RX110 グループ、RX111 グループ、RX113 グループ
- RX130 グループ
- RX13T グループ
- RX140 グループ
- RX230 グループ
- RX231 グループ
- RX23T グループ
- RX23W グループ
- RX23E-A グループ
- RX23E-B グループ
- RX24T グループ
- RX24U グループ
- RX260 グループ、RX261 グループ
- RX26T グループ
- RX64M グループ
- RX651 グループ、RX65N グループ
- RX660 グループ
- RX66T グループ
- RX66N グループ
- RX671 グループ
- RX71M グループ
- RX72T グループ
- RX72M グループ
- RX72N 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あわせて変更し、十分評価してください。

ターゲットコンパイラ

- ルネサスエレクトロニクス製 C/C++ Compiler Package for RX Family
- GCC for Renesas RX
- IAR C/C++ Compiler for Renesas RX

各コンパイラの動作確認環境に関する詳細な内容は、セクション「6.1 動作確認環境」を参照してください。

目次

1. 概要 5
1.1 MPC FIT モジュールとは 5
1.2 MPC FIT モジュールの概要 5
1.3 MPC FIT モジュールを使用する 5
1.3.1 MPC FIT モジュールを C++プロジェクト内で使用する 5
1.4 API の概要 5

2. API 情報 6
2.1 ハードウェアの要求 6
2.2 ソフトウェアの要求 6
2.3 制限事項 6
2.3.1 RAM の配置に関する制限事項 6
2.4 サポートされているツールチェーン 6
2.5 使用する割り込みベクタ 6
2.6 ヘッダファイル 6
2.7 整数型 6
2.8 コンパイル時の設定 7
2.9 コードサイズ 7
2.10 引数 7
2.10.1 MPC の端子設定 7
2.11 戻り値 8
2.12 コールバック関数 8
2.13 FIT モジュールの追加方法 8
2.14 for 文、while 文、do while 文について 9

3. API 関数 10
R_MPC_Write 10
R_MPC_Read 12
R_MPC_GetVersion 13

4. 端子設定 14

5. デモプロジェクト 15
5.1 mpc_demo_rskrx113, mpc_demo_rskrx64m, mpc_demo_rskrx71m, mpc_demo_rskrx65n and mpc_demo_rskrx65n_2m, mpc_demo_rskrx113_gcc, mpc_demo_rskrx64m_gcc, mpc_demo_rskrx71m_gcc, mpc_demo_rskrx65n_gcc and mpc_demo_rskrx65n_2m_gcc 15
5.2 mpc_demo_rskrx231, mpc_demo_rskrx231_gcc 15
5.3 mpc_demo_rskrx72m, mpc_demo_rskrx72m_gcc 15
5.4 mpc_demo_rskrx671, mpc_demo_rskrx671_gcc 15
5.5 ワークスペースにデモを追加する 15
5.6 デモのダウンロード方法 15

6. 付録 16
6.1 動作確認環境 16
6.2 トラブルシューティング 29

7. 参考ドキュメント 30

テクニカルアップデートの対応について 30

改訂記録 31

1. 概要

1.1 MPC FIT モジュールとは

本モジュールは API として、プロジェクトに組み込んで使用します。本モジュールの組み込み方については、「2.13 FIT モジュールの追加方法」を参照してください。

1.2 MPC FIT モジュールの概要

最近の MCU は小ピンパッケージを維持する一方で、装備される周辺機能は増加し続けています。小ピンパッケージで多数の周辺機能を備えるには、各端子に複数の機能を割り当てることになります。RX MCU では、端子への機能の割り当ては、MPC によって制御されます。本モジュールでは、GPIO FIT モジュールと連動して MPC 機能が使用できます。

1.3 MPC FIT モジュールを使用する

1.3.1 MPC FIT モジュールを C++プロジェクト内で使用する

C++プロジェクトでは、FIT GPIO モジュールのインタフェースヘッダファイルを extern “C”の宣言に追加してください。

```
Extern “C”
{
#include “r_smc_entry.h”
#include “r_mpc_rx_if.h”
}
```

1.4 API の概要

表 1.1 に本モジュールに含まれる API 関数を示します。

表 1.1API 関数一覧

関数	関数説明
R_MPC_Write()	端子の機能を設定します。
R_MPC_Read()	端子の機能設定を読み出します。
R_MPC_GetVersion()	本モジュールのバージョン番号を返します。

2. API 情報

本 FIT モジュールは、下記の条件で動作を確認しています。

2.1 ハードウェアの要求

ご使用になる MCU が以下の機能をサポートしている必要があります。

- MPC

2.2 ソフトウェアの要求

このドライバは以下の FIT モジュールに依存しています。

- ボードサポートパッケージ (r_bsp) v5.20 以上
- GPIO モジュール（r_gpio_rx）

2.3 制限事項

2.3.1 RAM の配置に関する制限事項

FIT では、API 関数のポインタ引数に NULL と同じ値を設定すると、パラメータチェックにより戻り値がエラーとなる場合があります。そのため、API 関数に渡すポインタ引数の値は NULL と同じ値にしないでください。

ライブラリ関数の仕様で NULL の値は 0 と定義されています。そのため、API 関数のポインタ引数に渡す変数や関数が RAM の先頭番地(0x0 番地)に配置されていると上記現象が発生します。この場合、セクションの設定変更をするか、API 関数のポインタ引数に渡す変数や関数が 0x0 番地に配置されないように RAM の先頭にダミーの変数を用意してください。

なお、CCRX プロジェクト(e2 studio V7.5.0)の場合、変数が 0x0 番地に配置されることを防ぐために RAM の先頭番地が 0x4 になっています。GCC プロジェクト(e2 studio V7.5.0)、IAR プロジェクト(EWRX V4.12.1)の場合は RAM の先頭番地が 0x0 になっていますので、上記対策が必要となります。

IDE のバージョンアップによりセクションのデフォルト設定が変更されることがあります。最新の IDE を使用される際は、セクション設定をご確認の上、ご対応ください。

2.4 サポートされているツールチェーン

本 FIT モジュールは「6.1 動作確認環境」に示すツールチェーンで動作確認を行っています。

2.5 使用する割り込みベクタ

なし

2.6 ヘッダファイル

すべての API 呼び出しとそれをサポートするインタフェース定義は r_mpc_rx_if.h に記載しています。

2.7 整数型

このドライバは ANSI C99 を使用しています。これらの型は stdint.h で定義されています。

2.8 コンパイル時の設定

本モジュールのコンフィギュレーションオプションの設定は、r_mpc_rx_config.h で行います。

オプション名および設定値に関する説明を、下表に示します。

Configuration options in r_mpc_rx_config.h	
MPC_CFG_PARAM_CHECKING_ENABLE	= 1：ビルド時にパラメータチェック処理をコードに含めます。 = 0：ビルド時にパラメータチェック処理をコードから省略します。 = BSP_CFG_PARAM_CHECKING_ENABLE（デフォルト）：システムのデフォルト設定を使用します。 注: ビルド時にパラメータチェックのコードを省略することで、コードサイズを小さくすることができます。

2.9 コードサイズ

本モジュールのコードサイズを下表に示します。

ROM (コードおよび定数) と RAM (グローバルデータ) のサイズは、ビルド時の「2.8 コンパイル時の設定」のコンフィギュレーションオプションによって決まります。掲載した値は、「2.4 サポートされているツールチェーン」の C コンパイラでコンパイルオプションがデフォルト時の参考値です。コンパイルオプションのデフォルトは最適化レベル：2、最適化のタイプ：サイズ優先、データ・エンディアン：リトルエンディアンです。コードサイズは C コンパイラのバージョンやコンパイルオプションにより異なります。

ROM、RAM およびスタックのコードサイズ

デバイス	分類	使用メモリ					
		ルネサス製コンパイラ		GCC		IAR	
		パラメータチェック処理あり	パラメータチェック処理なし	パラメータチェック処理あり	パラメータチェック処理なし	パラメータチェック処理あり	パラメータチェック処理なし
全デバイス	ROM	125 バイト	107 バイト	152 バイト	128 バイト	436 バイト	404 バイト
	RAM	0 バイト	0 バイト	0 バイト	0 バイト	0 バイト	0 バイト
	最大のスタック使用量	20 バイト	20 バイト	-	-	52 バイト	52 バイト

2.10 引数

API 関数の引数である構造体を示します。この構造体は、API 関数のプロトタイプ宣言とともに r_mpc_rx_if.h に記載されています。

2.10.1 MPC の端子設定

本データ構造体は、端子の機能設定に使用します。"*pin_function*"に有効な設定は、お使いの MCU のハードウェアマニュアルで「マルチファンクションピンコントローラ（MPC）」の章をご覧ください。対象の端子のポートに対応した端子機能制御レジスタを選択します。マニュアルでは、選択されたポートの各端子に割り当て可能な機能の一覧が示されています。以下に例を示します。

```
/* 端子の MPC レジスタ設定オプション*/
typedef struct
{
    uint8_t pin_function;  //端子に割り当てる周辺機能
    bool    irq_enable;    //端子を IRQ 端子として使用
    bool    analog_enable; //端子を ADC 入力端子、または DAC 出力端子として、
                           //または LVD(CMPA2)に使用
} mpc_config_t;
```

2.11 戻り値

API 関数の戻り値を示します。この列挙型は、API 関数のプロトタイプ宣言とともに r_mpc_rx_if.h で記載されています。

以下に R_MPC_Write()関数で使用できる戻り値を示します。

```
/* 関数の戻り値*/
typedef enum
{
    MPC_SUCCESS = 0,
    MPC_ERR_INVALID_CFG, // 指定された設定はこの端子には
                         // 適用できません。
} mpc_err_t;
```

2.12 コールバック関数

なし

2.13 FIT モジュールの追加方法

本モジュールは、使用するプロジェクトごとに追加する必要があります。ルネサスでは、Smart Configurator を使用した(1)、(2)の追加方法を推奨しています。ただし、Smart Configurator は、一部の RX デバイスのみサポートしています。サポートされていない RX デバイスについては(3)の方法を使用してください。

(1) e² studio 上で Smart Configurator を使用して FIT モジュールを追加する場合
e² studio の Smart Configurator を使用して、自動的にユーザプロジェクトに FIT モジュールを追加します。詳細は、アプリケーションノート「Renesas e² studio スマート・コンフィグレータ ユーザーガイド (R20AN0451)」を参照してください。

(2) CS+上で Smart Configurator を使用して FIT モジュールを追加する場合
CS+上で、スタンドアロン版 Smart Configurator を使用して、自動的にユーザプロジェクトに FIT モジュールを追加します。詳細は、アプリケーションノート「Renesas e² studio スマート・コンフィグレータ ユーザーガイド (R20AN0451)」を参照してください。

(3) CS+上で FIT モジュールを追加する場合
CS+上で、手動でユーザプロジェクトに FIT モジュールを追加します。詳細は、アプリケーションノート「RX ファミリ CS+に組み込む方法 Firmware Integration Technology (R01AN1826)」を参照してください。

2.14 for 文、while 文、do while 文について

本モジュールでは、レジスタの反映待ち処理等で for 文、while 文、do while 文（ループ処理）を使用しています。これらループ処理には、「WAIT_LOOP」をキーワードとしたコメントを記述しています。そのため、ループ処理にユーザがフェイルセーフの処理を組み込む場合は、「WAIT_LOOP」で該当の処理を検索できます。

以下に記述例を示します。

```
while 文の例：
/* WAIT_LOOP */
while(0 == SYSTEM.OSCOVFSR.BIT.PLOVF)
{
    /* The delay period needed is to make sure that the PLL has stabilized.*/
}

for 文の例：
/* Initialize reference counters to 0. */
/* WAIT_LOOP */
for (i = 0; i < BSP_REG_PROTECT_TOTAL_ITEMS; i++)
{
    g_protect_counters[i] = 0;
}

do while 文の例：
/* Reset completion waiting */
do
{
    reg = phy_read(ether_channel, PHY_REG_CONTROL);
    count++;
} while ((reg & PHY_CONTROL_RESET) && (count < ETHER_CFG_PHY_DELAY_RESET)); /* WAIT_LOOP */
```

3. API 関数

R_MPC_Write

この関数は、端子の機能を設定する関数です。

Format

```
mpc_err_t    R_MPC_Write (
        gpio_port_pin_t             pin,
        mpc_config_t                * pconfig
)
```

Parameters

gpio_port_pin_t pin
設定する端子

*mpc_config_t *pconfig*
端子の設定情報を含む構造体へのポインタ。セクション 2.10.1 参照。

Return Values

```
[MPC_SUCCESS]          /* 成功; 端子が設定されました。          */
[MPC_ERR_INVALID_CFG]  /* エラー; 無効な設定が入力されました。  */
```

Properties

r_mpc_rx_if.h にプロトタイプ宣言されています。

Description

本関数は"mpc_config_t"構造体の情報を元に端子を設定します。すべての端子が同じ機能をサポートしているわけではありません。例えば、どの端子でも ADC や DAC のアナログ端子として設定できるわけではありません。また、機能の組み合わせもどれでもいいというわけではなく、アナログ端子と周辺機能の端子としての機能は同時に設定できません。

端子に対して割り当て可能な機能については、お使いの MCU のハードウェアマニュアルで「マルチファンクションピンコントローラ（MPC）」の章をご覧ください。対象の端子のポートに対応した端子機能制御レジスタを選択します。マニュアルでは、選択されたポートの各端子に割り当て可能な機能表をご覧いただけます。

本関数によって設定する端子は、r_gpio_rx モジュールの"gpio_port_pin_t"型を使って定義されます。

Example

```
mpc_config_t      config;
gpio_port_pin_t   pin;

/* PE0 を ADC のアナログ端子として使用するように設定*/
pin = GPIO_PORT_E_PIN_0;

config.analog_enable = true;
config.irq_enable = false;
config.pin_function = 0;

if (MPC_SUCCESS != R_MPC_Write(pin, &config))
```

```
{
    /* エラー、端子はこの設定に対応していません。ハンドルエラー*/
    …
}

/* P27 を IRQ 端子、および SCI の処理で使用するように設定*/
pin = GPIO_PORT_2_PIN_7;

config.analog_enable = false;
config.irq_enable = true;
config.pin_function = 0xA;

if (MPC_SUCCESS != R_MPC_Write(pin, &config))
{
    /* エラー、端子はこの設定に対応していません。ハンドルエラー*/
    …
}
```

Special Notes:
なし

R_MPC_Read

端子の機能設定の読み出しをする関数です。

Format

```
void R_MPC_Read (
    gpio_port_pin_t     pin,
    mpc_config_t        * pconfig
)
```

Parameters

gpio_port_pin_t pin
設定情報を読み出す端子

*mpc_config_t *pconfig*
端子の設定情報を含む構造体へのポインタ。セクション 2.10.1 参照。

Return Values

なし

Properties

ファイル r_mpc_rx_if.h にプロトタイプ宣言されています。

Description

本関数は端子の設定情報を読み出し、ユーザによって提供される構造体に格納します。

Example

```
mpc_config_t config;

/* P03 は DAC のアナログ端子として既に設定されているかどうかを確認します。*/
R_MPC_Read(GPIO_PORT_0_PIN_3, &config);

if (config.analog_enable == true)
{
    /* P03 は既にアナログ端子として設定されています。*/
    …
}
else
{
    /* P03 はまだ設定されていません。ここで設定してください。*/
    …
}
```

Special Notes:

なし

R_MPC_GetVersion

実行時に本モジュールのバージョンを返す関数です。

Format
uint32_t R_MPC_GetVersion (void)

Parameters
なし

Return Values
本モジュールのバージョン

Properties
ファイル r_mpc_rx_if.h にプロトタイプ宣言されています。

Description
この関数は本モジュールのバージョンを返します。バージョン番号は符号化され、最上位の 2 バイトがメジャーバージョン番号を、最下位の 2 バイトがマイナーバージョン番号を示しています。例えば、ver. 4.25 の場合、“0x00040019”が返されます。

Example
```
uint32_t cur_version;

/* 実行中の r_mpc_rx API のバージョンを取得*/
cur_version = R_MPC_GetVersion();

/* 本アプリケーションを使用可能なバージョンであることを確認*/
if (MIN_VERSION > cur_version)
{
    /* お使いの r_mpc_rx バージョンは、本アプリケーションで必要な XXX 機能を備えていない警告
       旧バージョンです。*/
    …
}
```

Special Notes:
本関数は r_mpc_rx.c のインライン関数となります。

4. 端子設定

MPC FIT モジュールは端子設定を使用しません。

5. デモプロジェクト

デモプロジェクトには、FIT モジュールとそのモジュールが依存するモジュール（例：r_bsp）を使用する main()関数が含まれます。本 FIT モジュールには以下のデモプロジェクトが含まれます。

5.1 mpc_demo_rskrx113, mpc_demo_rskrx64m, mpc_demo_rskrx71m, mpc_demo_rskrx65n and mpc_demo_rskrx65n_2m, mpc_demo_rskrx113_gcc, mpc_demo_rskrx64m_gcc, mpc_demo_rskrx71m_gcc, mpc_demo_rskrx65n_gcc and mpc_demo_rskrx65n_2m_gcc

mpc_demo_rskrx113、mpc_demo_rskrx64m、mpc_demo_rskrx71m、mpc_demo_rskrx65n および mpc_demo_rskrx65n_2m は RX113、RX64M、RX71M、RX65N および RX65N-2MB における MPC FIT モジュールのデモプログラムです。これらのプログラムで、MPC を使って、割り込みの入力としてポートビットを設定する方法をデモします。デモでは割り込みとして IRQ2 が選択され、SW2 の押下の検出に使用されます。コードがコンパイルされ、対象のボードにダウンロードして実行されると、SW2 を押して、IRQ2（RX65N では IRQ9、RX65N-2MB では IRQ13）割り込みを発生させることができます。次に IRQ2（RX65N では IRQ9、RX65N-2MB では IRQ13）割り込み処理で、LED3 のステートがトグルされます。

5.2 mpc_demo_rskrx231, mpc_demo_rskrx231_gcc

mpc_demo_rskrx231 プロジェクトは、RX231 MPC FIT モジュールのデモプログラムです。プログラムの動作は mpc_demo_rskrx113 と同様ですが、SW2 の押下の検出に IRQ4 が使用されます。

5.3 mpc_demo_rskrx72m, mpc_demo_rskrx72m_gcc

mpc_demo_rskrx72m, mpc_demo_rskrx72m_gcc プロジェクトは、RX72M MPC FIT モジュールのデモプログラムです。プログラムの動作は mpc_demo_rskrx71m, mpc_demo_rskrx71m_gcc と同様ですが、SW2 の押下の検出に IRQ12 が使用されます。

5.4 mpc_demo_rskrx671, mpc_demo_rskrx671_gcc

mpc_demo_rskrx671, mpc_demo_rskrx671_gcc プロジェクトは、RX671 MPC FIT モジュールのデモプログラムです。プログラムの動作は mpc_demo_rskrx71m, mpc_demo_rskrx71m_gcc と同様ですが、SW2 の押下の検出に IRQ10 が使用されます。

5.5 ワークスペースにデモを追加する

デモプロジェクトは、本アプリケーションノートで提供されるファイルの FITDemos サブディレクトリにあります。ワークスペースにデモプロジェクトを追加するには、「ファイル」→「インポート」を選択し、「インポート」ダイアログから「一般」の「既存プロジェクトをワークスペースへ」を選択して「次へ」ボタンをクリックします。「インポート」ダイアログで「アーカイブ・ファイルの選択」ラジオボタンを選択し、「参照」ボタンをクリックして FITDemos サブディレクトリを開き、使用するデモの zip ファイルを選択して「完了」をクリックします。

5.6 デモのダウンロード方法

デモプロジェクトは、RX Driver Package には同梱されていません。デモプロジェクトを使用する場合は、個別に各 FIT モジュールをダウンロードする必要があります。「スマートブラウザ」の「アプリケーションノート」タブから、本アプリケーションノートを右クリックして「サンプル・コード（ダウンロード）」を選択することにより、ダウンロードできます。

6. 付録

6.1 動作確認環境

本 FIT モジュールの動作確認環境を以下に示します。

表 6.1 動作環境の確認（Rev.5.01）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2025-01 IAR Embedded Workbench for Renesas RX 5.10.1
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 V3.07.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202411 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 5.10.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.5.01
使用ボード	-

表 6.2 動作環境の確認（Rev.5.00）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2024-07 IAR Embedded Workbench for Renesas RX 5.10.1
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 V3.06.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202405 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 5.10.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.5.00
使用ボード	Evaluation Kit for RX261（型名：RTK5EK2610S00011BJ）

表 6.3 動作環境の確認（Rev.4.90）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2023-04 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.05.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202204 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.90
使用ボード	Renesas Solution Starter Kit for RX23E-B（型名：RTK0ES1001C00001BJ）

表 6.4 動作環境の確認（Rev.4.80）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2022-10 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.05.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202204 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.80
使用ボード	Renesas Flexible Motor Control Kit for RX26T(型名：RTK0EMXE70S00020BJ)

表 6.5 動作環境の確認（Rev.4.70）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2023-01 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202204 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.70

表 6.6 動作環境の確認（Rev.4.60）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2022-10 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202204 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.60

表 6.7 動作環境の確認（Rev.4.50）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e^2 studio V.2022-07 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202104 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.50
使用ボード	Renesas Starter Kit for RX113 (型名 : R0K505113CxxxBE) Renesas Starter Kit for RX231 (型名: R0K505231SxxxBE) Renesas Starter Kit for RX64M (型名: R0K50564MxxxxBE) Renesas Starter Kit+ for RX65N (型名: RTK5005651CxxxxxBE) Renesas Starter Kit+ for RX65N-2MB (型名: RTK50565N2CxxxxxBR) Renesas Starter Kit+ for RX671 (型名: RTK55671EDCxxxxxBJ) Renesas Starter Kit for RX71M (型名: R0K50571MCxxxBE) Renesas Starter Kit+ for RX72M (型名: RTK5572MNDCxxxxxBJ)

表 6.8 動作環境の確認（Rev.4.40）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2022-04 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202104 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.40
使用ボード	Renesas Starter Kit for RX660（型名：RTK556609HCxxxxxBJ）

表 6.9 動作環境の確認（Rev.4.30）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2021-10 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202104 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.30
使用ボード	Renesas Starter Kit for RX66T（型名：RTK50566T0SxxxxxBE）

表 6.10 動作環境の確認（Rev.4.20）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2022-01 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.04.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202104 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.20
使用ボード	Renesas Starter Kit+ for RX140（型名：RTK55140xxxxxxxxxx）

表 6.11 動作環境の確認（Rev.4.10）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2021-07 IAR Embedded Workbench for Renesas RX 4.20.3
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 V3.03.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202004 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.3 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.10
使用ボード	Renesas Starter Kit+ for RX671（型名：RTK55671xxxxxxxxxx）

表 6.12 動作環境の確認（Rev.4.00）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2021-07 IAR Embedded Workbench for Renesas RX 4.20.1
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 V3.03.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202004 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.4.00
使用ボード	Renesas Starter Kit+ for RX140（型名：RTK5RX140xxxxxxxxx）

表 6.13 動作環境の確認（Rev.3.90）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2021-07 IAR Embedded Workbench for Renesas RX 4.20.1
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 V3.03.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202004 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.90
使用ボード	Renesas Starter Kit+ for RX671（型名：RTK55671xxxxxxxxxx）

表 6.14 動作環境の確認（Rev.3.80）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio 2021-01 IAR Embedded Workbench for Renesas RX 4.20.1
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 V3.03.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202004 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.20.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.80
使用ボード	Renesas Starter Kit+ for RX72M（型名：RTK5572Mxxxxxxxxxx）

表 6.15 動作環境の確認（Rev.3.70）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.2021-01
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 V3.03.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.202004 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	リトルエンディアン
モジュールのリビジョン	Rev.3.70
使用ボード	Renesas Solution Starter Kit for RX23W（型名：RTK5523Wxxxxxxxxx）

表 6.16 動作環境の確認（Rev.3.60）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.7.8.0
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 V3.02.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.201904 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	リトルエンディアン
モジュールのリビジョン	Rev.3.60
使用ボード	Renesas Starter Kit+ for RX72M（型名：RTK5572Mxxxxxxxxxx） Renesas Starter Kit+ for RX65N-2MB （型名：RTK50565N2CxxxxxBR） Renesas Starter Kit+ for RX65N （型名：RTK50565NCxxxxxBE） Renesas Starter Kit+ for RX64M （型名：RTK50564Mxxxxxxxx） Renesas Starter Kit+ for RX71M （型名：RTK50571Mxxxxxxxx） Renesas Starter Kit+ for RX113 （型名：RTK505113xxxxxxxx） Renesas Starter Kit+ for RX231 （型名： RTK505231xxxxxxxx）

表 6.17 動作環境の確認（Rev.3.50）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.7.7.0 IAR Embedded Workbench for Renesas RX 4.12.1
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 V3.02.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 8.3.0.201904 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.12.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.50
使用ボード	Renesas Solution Starter Kit+ for RX23E-A CPU カード （製品番号：RTK0ESXBxxxxxxxxxx）

表 6.18 動作環境の確認（Rev.3.40）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.7.7.0 IAR Embedded Workbench for Renesas RX 4.12.1
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 V3.01.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 4.8.4.201902 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.40
使用ボード	Renesas Starter Kit+ for RX72N（型名：RTK5572Nxxxxxxxxxx）

表 6.19 動作環境の確認（Rev.3.30）

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.7.7.0 IAR Embedded Workbench for Renesas RX 4.12.1
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 V3.01.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 4.8.4.201902 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FCC 周辺モジュールで宣言された割り込み関数をリンカーが誤って破棄する GCC リンカーの問題を回避するためです。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.12.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.30
使用ボード	RX13T CPU カード（製品番号：RTK0EMXA10C00000BJ）

表 6.20 動作確認環境 (Rev.3.20)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e2 studio V.7.5.0 IAR Embedded Workbench for Renesas RX 4.12.1
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 V3.01.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 4.8.4.201902 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FIT 周辺機器モジュール内で宣言されている割り込み関数をリンカが誤って破棄（discard）することを回避（work around）するための対策です。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.12.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.20
使用ボード	Renesas Starter Kit+ for RX72M（型名：RTK5572Mxxxxxxxxxx）

表 6.21 動作確認環境 (Rev.3.10)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e^2 studio V.7.5.0
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 V3.01.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.10
使用ボード	Renesas Solution Starter Kit for RX23W（型名：RTK5523Wxxxxxxxxx）

表 6.22 動作確認環境 (Rev.3.00)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e2 studio V.7.4.0 IAR Embedded Workbench for Renesas RX 4.10.1
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 V3.01.00 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
	GCC for Renesas RX 4.8.4.201803 コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -std=gnu99 リンクオプション：「Optimize size (サイズ最適化) (-Os)」を使用する場合、統合開発環境のデフォルト設定に以下のオプションを追加 -Wl,--no-gc-sections これは、FIT 周辺機器モジュール内で宣言されている割り込み関数をリンカが誤って破棄（discard）することを回避（work around）するための対策です。
エンディアン	IAR C/C++ Compiler for Renesas RX version 4.10.1 コンパイルオプション：統合開発環境のデフォルト設定
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.3.00
使用ボード	Renesas Starter Kit+ for RX65N-2MB（型名：RTK50565Nxxxxxxxxx）

表 6.23 動作確認環境 (Rev.2.50)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e^2 studio V.7.3.0
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 V3.01.00
	コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.2.50
使用ボード	Renesas Starter Kit for RX72T（型名：RTK5572Txxxxxxxxx)

表 6.24 動作確認環境 (Rev.2.41)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e^2 studio V.7.3.0
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 V3.01.00
	コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.2.41
使用ボード	Renesas Starter Kit for RX66T（型名：RTK50566T0SxxxxxBE) Renesas Starter Kit+ for RX 65N-2MB（型名：RTK50565N2CxxxxxBR） Renesas Starter Kit+ for RX130-512KB（型名：RTK5051308CxxxxxBR）

表 6.25 動作確認環境 (Rev.2.40)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V.7.0.0
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 V3.00.00
	コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン／リトルエンディアン
モジュールのリビジョン	Rev.2.40
使用ボード	Renesas Starter Kit for RX66T（型名：RTK50566T0SxxxxxBE) Renesas Starter Kit+ for RX 65N-2MB（型名：RTK50565N2CxxxxxBR） Renesas Starter Kit+ for RX130-512KB（型名：RTK5051308CxxxxxBR）

表 6.26 動作確認環境 (Rev.2.31)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V6.0.0
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V2.07.00
	コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン/リトルエンディアン
モジュールのリビジョン	Rev2.31
使用ボード	Renesas Starter Kit+ for RX 65N-2MB（型名：RTK50565N2CxxxxxBR） Renesas Starter Kit+ for RX130-512KB（型名：RTK5051308CxxxxxBR）

表 6.27 動作確認環境 (Rev.2.30)

項目	内容
統合開発環境	ルネサスエレクトロニクス製 e² studio V6.0.0
C コンパイラ	ルネサスエレクトロニクス製 C/C++ Compiler for RX Family V2.07.00
	コンパイルオプション：統合開発環境のデフォルト設定に以下のオプションを追加 -lang = c99
エンディアン	ビッグエンディアン/リトルエンディアン
モジュールのリビジョン	Rev2.30
使用ボード	Renesas Starter Kit+ for RX 65N-2MB（型名：RTK50565N2CxxxxxBR） Renesas Starter Kit+ for RX130-512KB（型名：RTK5051308CxxxxxBR）

6.2 トラブルシューティング

(1) Q：本 FIT モジュールをプロジェクトに追加しましたが、ビルド実行すると「Could not open source file "platform.h"」エラーが発生します。

A：FIT モジュールがプロジェクトに正しく追加されていない可能性があります。プロジェクトへの追加方法をご確認ください。

- CS+を使用している場合
  アプリケーションノート RX ファミリ CS+に組み込む方法 Firmware Integration Technology (R01AN1826)」
- e² studio を使用している場合
  アプリケーションノート RX ファミリ e² studio に組み込む方法 Firmware Integration Technology (R01AN1723)」

また、本 FIT モジュールを使用する場合、ボードサポートパッケージ FIT モジュール(BSP モジュール)もプロジェクトに追加する必要があります。BSP モジュールの追加方法は、アプリケーションノート「ボードサポートパッケージモジュール(R01AN1685)」を参照してください。

(2) Q：本 FIT モジュールをプロジェクトに追加しましたが、ビルド実行すると「This MCU is not supported by the current r_mpc_rx module.」エラーが発生します。

A：追加した FIT モジュールがユーザプロジェクトのターゲットデバイスに対応していない可能性があります。追加した FIT モジュールの対象デバイスを確認してください。

(3) Q：本 FIT モジュールをプロジェクトに追加しましたが、ビルド実行すると「コンフィグ設定が間違っている場合のエラーメッセージ」エラーが発生します。

A：「r_mpc_rx_config.h」ファイルの設定値が間違っている可能性があります。
「r_mpc_rx_config.h」ファイルを確認して正しい値を設定してください。詳細は「2.7 コンパイル時の設定」を参照してください。

7. 参考ドキュメント

ユーザーズマニュアル：ハードウェア
（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 CC-RX コンパイラ ユーザーズマニュアル（R20UT3248）
（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の対応について

本モジュールは以下のテクニカルアップデートの内容を反映しています。
なし

改訂記録

Rev.	発行日	改訂内容	
		ページ	
1.00	2013.11.15	—	初版発行
1.20	2014.04.17	1,3	FIT モジュールの RX64M グループ対応
1.30	2014.07.02	—	Fixed RX63N receive bug. Used latest Colophon.
1.40	2014.12.12	Various 1 5	Updated to current app-note template Added RX113 to the list of supported MCUs Added a Code Size section.
1.50	2015.01.17	—	FIT モジュールの RX71M グループ対応
1.60	2015.06.30	—	FIT モジュールの RX231 グループ対応
1.70	2015.09.30	—	FIT モジュールの RX23T グループ対応
1.80	2015.10.01	—	FIT モジュールの RX130 グループ対応
1.90	2015.12.01	— 1, 6 12	FIT モジュールの RX24T グループ対応 アプリケーションノート「ボードサポートパッケージモジュール Firmware Integration Technology」のドキュメント番号変更 「4. デモプロジェクト」追加
2.00	2016.02.01	— 13	FIT モジュールの RX230 グループ対応 「テクニカルアップデートの対応について」追加
2.01	2016.07.29	12	「4. デモプロジェクト」に RSKRX64M を追加 「4.1 mpc_demo_rskrx113、mpc_demo_rskrx64m、mpc_demo_rskrx71m」の誤記修正 ・”cmt_demo_rskrx113” -> “ mpc_demo_rskrx113” ・”cmt_demo_rskrx71M” -> “ mpc_demo_rskrx71M”
2.10	2016.10.01	— 5	FIT モジュールの RX65N グループ対応 2.9「コードサイズ」の表フォーマットを変更 2.9「コードサイズ」に RX65N グループのコードサイズを追加
2.20	2017.02.28	— 4	FIT モジュールの RX24U グループ対応 「2.5 対応ツールチェーン」に RXC v2.06.00 を追加
2.30	2017.07.21	- 4 7	FIT モジュールの RX130 グループ（ROM 512KB 版）と RX65N グループ（ROM 2MB 版）対応 「2.5 対応ツールチェーン」に RXC v2.07.00 を追加 「2.12 FIT モジュールをプロジェクトに追加する方法」を更新
2.31	2017.10.31	13 14	RSKRX65N と RSKRX65N-2MB を、「4. デモプロジェクト」に追加 「4.4 デモのダウンロード方法」を追加 「5. 付録」を追加
2.40	2018.09.28	1 14	RX66T のサポートを追加。 「6.1 動作確認環境」に、Rev.2.40 に対応する表を追加
2.41	2018.11.16	— 14	XML 内にドキュメント番号を追加。 Renesas Starter Kit+ for RX66T の型名を変更。 Rev.2.41 に対応する表を追加
2.50	2019.02.01	— 1 8-11 14	RX72T グループのサポートを追加。 RX72T グループのサポートを追加。 各 API 関数で「Reentrant」の説明を削除。 「6.1 動作確認環境」Rev 2.50 に対応する表を追加。

3.00	2019.05.20	—	以下のコンパイラをサポート。 - GCC for Renesas RX - IAR C/C++ Compiler for Renesas RX
		1	「ターゲットコンパイラ」のセクションを追加。 関連ドキュメントを削除。
		4	「2.2 ソフトウェアの要求」r_bsp v5.20 以上が必要
		5	「2.8 コードサイズ」セクションを更新。
		14	表 6.1「動作確認環境」： Rev.3.00 に対応する表を追加。
		18	「Web サイトおよびサポート」のセクションを削除。
		プログラム	GCC と IAR コンパイラに関して、以下を変更。 R_MPC_GetVersion 関数のインライン展開を削除。
3.10	2019.06.28	1	RX23W のサポートを追加。
		14	「6.1 動作確認環境」： Rev.3.10 に対応する表を追加。
		プログラム	RX23W のサポートを追加。
3.20	2019.08.15	1	RX72M のサポートを追加。
		14	「6.1 動作確認環境」： Rev.3.20 に対応する表を追加。 表 6.2：RX23W ボード名変更。
		プログラム	RX72M のサポートを追加。
3.30	2019.11.25	1	RX13T のサポートを追加。
		4	2.3 制限事項 制限事項を追加。
		5	2.9 コードサイズのセクションを更新。
		14	「6.1 動作確認環境」： Rev.3.30 に対応する表を追加。
		プログラム	RX13T のサポートを追加。 API 関数のコメントを Doxygen スタイルに変更。
3.40	2019.12.30	1	RX72N, RX66N のサポートを追加。
		5	2.9 コードサイズのセクションを更新。
		14	「6.1 動作確認環境」： Rev.3.40 に対応する表を追加。
		プログラム	RX72N, RX66N のサポートを追加。
3.50	2020.03.31	1	RX23E-A のサポートを追加。
		6	2.9 コードサイズのセクションを更新。
		15	「6.1 動作確認環境」： Rev.3.50 に対応する表を追加。
		プログラム	RX23E-A のサポートを追加。
3.60	2020.06.30	14	デモプロジェクトの更新と追加 「5. デモプロジェクト」に RSKRX72M を追加。
		15	「6.1 動作確認環境」： Rev.3.60 に対応する表を追加。
		プログラム	デモプロジェクトの更新と追加
3.70	2021.04.01	15	「6.1 動作確認環境」： Rev.3.70 に対応する表を追加。
		プログラム	RX23W-83pin のサポートを追加。
3.80	2021.04.01	15	「6.1 動作確認環境」： Rev.3.80 に対応する表を追加。
		プログラム	RX72M-100pin, 144pin のサポートを追加。

3.90	2021.04.07	1 4 6 15 プログラム	RX671 のサポートを追加。 「1.3 MPC FIT モジュールを使用する」のセクションを追加。 「1.3.1 MPC FIT モジュールを C++プロジェクト内で使用する」のセクションを追加。 RX671 に対応するコードサイズを追加。 「6.1 動作確認環境」： Rev.3.90 に対応する表を追加。 RX671 のサポートを追加。
4.00	2021.04.15	1 6 15 プログラム	RX140 のサポートを追加。 RX140 に対応するコードサイズを追加。 「6.1 動作確認環境」： Rev.4.00 に対応する表を追加。 RX140 のサポートを追加。 デモプロジェクトに CS+ のサポートを追加。
4.10	2021.09.13	14 15 プログラム	「5. デモプロジェクト」に RSKRX671 を追加。 「6.1 動作確認環境」： Rev.4.10 に対応する表を追加。 デモプロジェクトの更新と追加。
4.20	2021.11.11	15 プログラム	「6.1 動作確認環境」： Rev.4.20 に対応する表を追加。 RX140-256KB のサポートを追加。
4.30	2022.03.14	15 プログラム	「6.1 動作確認環境」： Rev.4.30 に対応する表を追加。 RX66T-48pin のサポートを追加。
4.40	2022.03.31	1 7 16 プログラム	RX660 のサポートを追加。 RX660 に対応するコードサイズを追加。 「6.1 動作確認環境」： Rev.4.40 に対応する表を追加。 RX660 のサポートを追加。
4.50	2022.06.28	16 プログラム	「6.1 動作確認環境」： Rev.4.50 に対応する表を追加。 デモプロジェクトを更新
4.60	2022.12.15	16 プログラム	「6.1 動作確認環境」： Rev.4.60 に対応する表を追加。 依存モジュールのバージョンを更新。
4.70	2023.02.28	16 プログラム	「6.1 動作確認環境」： Rev.4.70 に対応する表を追加。 依存モジュールのバージョンを更新。
4.80	2023.04.07	1 7 16 プログラム	RX26T のサポートを追加。 RX26T に対応するコードサイズを追加。 「6.1 動作確認環境」： Rev.4.80 に対応する表を追加。 RX26T のサポートを追加。 依存モジュールのバージョンを更新。
4.90	2023.05.29	1 7	RX23E-B のサポートを追加。 RX23E-B に対応するコードサイズを追加。

		8 16 プログラム	「2.13 FIT モジュールの追加方法」から FIT configurator の記述を削除 「6.1 動作確認環境」： Rev.4.90 に対応する表を追加。 RX23E-B のサポートを追加。
5.00	2024.06.28	1 7 16 プログラム	RX260、RX261 のサポートを追加。 RX260、RX261 に対応するコードサイズを追加。 「6.1 動作確認環境」： Rev.5.00 に対応する表を追加。 RX260、RX261 のサポートを追加。
5.01	2025.03.15	16 プログラム	「6.1 動作確認環境」： Rev.5.01 に対応する表を追加。 FIT モジュールの免責事項と著作権を更新。

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。

© 2025 Renesas Electronics Corporation. All rights reserved.