

【注意事項】

R20TS0507JJ0100

Rev.1.00

2019.11.01 号

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,

RX Driver Package

概要

タイトルに記載している製品の使用上の注意事項を連絡します。

1. CS+使用時の“iodefine.h”のインクルードに関する注意事項

1. CS+使用時の“iodefine.h”のインクルードに関する注意事項

1.1 該当製品

(1) ボードサポートパッケージモジュール Firmware Integration Technology (BSP FIT モジュール)

該当するリビジョンおよびドキュメントは、以下のとおりです。

表 1.1 BSP FIT モジュール該当製品一覧

BSP FIT モジュールのリビジョン	資料番号
Rev.5.20	R01AN1685JJ0520
Rev.5.21	R01AN1685JJ0521

(2) RX Driver Package

(1) の BSP FIT モジュールは、RX Driver Package にも同梱されています。

該当する RX Driver Package の製品名、リビジョン、および同梱している BSP FIT モジュールのリビジョンは、以下のとおりです。

表 1.2 BSP FIT モジュール同梱製品一覧

RX Driver Package の製品名	RX Driver Package のリビジョン	資料番号	同梱している BSP FIT モジュールのリビジョン
RX ファミリ RX Driver Package Ver.1.20	Rev.1.20	R01AN4794JJ0120	Rev.5.20
RX ファミリ RX Driver Package Ver.1.22	Rev.1.22	R01AN4873JJ0122	Rev.5.20

1.2 該当デバイス

- ・ RX110、RX111、RX113、および RX130 グループ
- ・ RX230、RX231、RX23T、RX23E-A、RX24T、および RX24U グループ
- ・ RX64M、RX65N、RX651、および RX66T グループ
- ・ RX71M、RX72T、および RX72M グループ

1.3 内容および発生条件

以下の現象が発生する可能性があります。

(1) 使用中のコンパイラとは異なるコンパイラ用の“ iodef.h”をインクルード

以下の条件をすべて満たす場合に発生します。

- (a) 手動またはスマート・コンフィグレータのコード生成を利用して Rev.5.20 以降の BSP FIT モジュールを CS+のプロジェクトに組み込んでいる。
- (b) ユーザ作成のソースファイルで“ iodef.h”を直接指定し、インクルードしている。

[内容の詳細]

CS+の「追加のインクルード・パス」設定には以下の“ iodef.h”パスが自動で追加^(注)されます。そのため、使用中のコンパイラに関わらず、一番上に設定されたパスに存在する“ iodef.h”がインクルードされます。

注：パスの自動追加は、“ iodef.h”、または、このファイルを含むフォルダを CS+プロジェクトに新規登録する場合に発生します。

● “ iodef.h”パス

- ・ src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥iccrx
- ・ src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥gnuc
- ・ src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥ccrx

なお、上記のパスのいずれかを手動削除後、再度、パスの自動追加が発生した場合、パスの順番は上記と異なる場合があります。

(2) 多重定義によるビルドエラー（RX65N および RX651 グループのみ該当）

以下の条件をすべて満たす場合に発生します。

- (a) CS+を使用している。
- (b) ユーザ作成のソースファイルで“platform.h”と“iodifne.h”の両方をインクルードしている。

[内容の詳細]

以下の原因により“iodefine.h”のインクルードガードが機能しないため、ビルドエラーが発生します。

- (i) “platform.h”がインクルードする“`¥board¥<ボード名>¥r_bsp.h`”は、以下のソースコードによって使用するコンパイラに適切な“iodefine.h”をインクルードします。

```
#if defined(__CCRX__)
#include "mcu/rx72m/register_access/ccrx/iodefine.h"
#elif defined(__GNUC__)
#include "mcu/rx72m/register_access/gnuc/iodefine.h"
#elif defined(__ICCRX__)
#include "mcu/rx72m/register_access/iccrx/iodefine.h"
#endif /* defined(__CCRX__), defined(__GNUC__), defined(__ICCRX__) */
```

- (ii) ユーザ作成のソースファイルで“iodifne.h”もインクルードしている場合、1.3 項(1)のとおり使用するコンパイラに不適切な“iodifne.h”がインクルードされる可能性があります。
- (iii) RX65N および RX651 用の BSP FIT モジュールでは、各コンパイラ用（ルネサス製 CC-RX コンパイラ、IAR システムズ製コンパイラ、および GNU コンパイラ用）の“iodefine.h”間でインクルードガード用の定義が以下のように異なります。

- ・ `src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥ccrx¥iodefine.h`

```
#ifndef __RX651IODEFINE_HEADER__
#define __RX651IODEFINE_HEADER__
----- (中略) -----
#endif
```

- ・ `src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥iccrx¥iodefine.h`、および、`src¥smc_gen¥r_bsp¥mcu¥<デバイス名>¥register_access¥gnuc¥iodefine.h`

```
#ifndef __RX65NIODEFINE_HEADER__
#define __RX65NIODEFINE_HEADER__
----- (中略) -----
#endif
```

1.4 回避策

以下のいずれかの方法で 1.3 項(1)および(2)の現象を回避できます。

(1) ユーザ作成のソースファイル修正

ユーザ作成のソースファイルにおいて、“ iodef.h”ではなく“ platform.h”をインクルードしてください。“ platform.h”は使用するコンパイラに応じて適切な“ iodef.h”をインクルードします。

以下に修正例を記載します。赤字の部分が修正箇所です。

・ 修正前

```
#include "iodef.h"
```

・ 修正後

```
#include "platform.h"
```

(2) 「追加のインクルード・パス」設定を変更

CS+の「追加のインクルード・パス」設定で、使用しないコンパイラの“ iodef.h”のインクルード・パスを削除してください^(注)。

注：BSP FIT モジュールをプロジェクトから外し、再びプロジェクトに登録した場合、削除前の状態に戻りますので、再度、削除が必要です。

1.5 恒久対策

今後のバージョンで改修予定です。

以上

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Nov.01.19	-	新規発行

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ものです。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。

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。

ニュース本文中の URL を予告なしに変更または中止することがありますので、あらかじめご承知ください。

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。