

## RX ファミリ

R20AN0431JJ0100

## システムタイマモジュール

Rev.1.00

2016.11.30

## Firmware Integration Technology

## 要旨

本資料は、RX ファミリの CMT を活用したシステムタイマモジュールを解説します。システムタイマは年月日時間分秒をカウントします。また、簡易的なスケジューラ機能も持ちます。ユーザはシステムタイマに呼び出し間隔を指定して関数ポインタを登録することで、定常的に行う必要のある処理をシステムタイマに実行させることができます。

## 動作確認デバイス

RX ファミリ

## 目次

1. 概要 .....	2
1.1 ファイル構成 .....	2
1.2 開発環境 .....	4
1.3 コンパイラオプション情報 .....	4
1.4 ROM / RAM サイズ情報 .....	4
1.5 スタックサイズ情報 .....	4
1.6 セクション情報 .....	4
2. API 情報 .....	5
3. API 関数 .....	7
3.1 API 一覧 .....	7
4. API 関数詳細説明 .....	8

## 1. 概要

### 1.1 ファイル構成

本アプリケーションノート付属のサンプルコードは、以下の表 1-1 のファイルが含まれます。

表 1-1 ファイル構成 1

ファイル/ディレクトリ(太字)名		内容
r20an0431jj0100_rx_middle.pdf		システムタイマモジュールアプリケーションノート(日本語)
<b>reference_documents</b>		FIT モジュールを各種統合開発環境で使用方法を記したドキュメントを格納するフォルダ
	r01an1826jj0102_rx.pdf	CS+に組み込む方法
	r01an1723ju0111_rx.pdf	e2studio に組み込む方法
<b>FITModules</b>		FIT モジュールフォルダ
	r_sys_time_rx_v.1.00.zip	システムタイマモジュール
	r_sys_time_rx_v.1.00.xml	システムタイマモジュール e2 studio FIT コンフィグレータ用 XML ファイル

r\_sys\_time\_rx\_v.1.00.zip を解凍したフォルダには、以下の表 1-2 のファイルが含まれます。

表 1-2 ファイル構成 2

ファイル/ディレクトリ(太字)名		内容
<b>r_config</b>		システムタイマコンフィグファイルフォルダ
	r_sys_time_rx_config.h	システムタイマコンフィグファイル(デフォルト設定)
<b>r_sys_time_rx</b>		システムタイマ FIT Module フォルダ
<b>src</b>		システムタイマソースコードフォルダ
	r_sys_time_rx.c	システムタイマソースコード
	r_sys_time_rx_private.h	システムタイマ内部ヘッダファイル
<b>doc</b>		システムタイマ ドキュメントフォルダ
<b>ja</b>		システムタイマ ドキュメントフォルダ(日本語)
	r20an0431jj0100_rx_middle.pdf	システムタイマ アプリケーションノート(日本語)
<b>en</b>		システムタイマドキュメントフォルダ(英語)
	r20an0431ej0100_rx_middle.pdf	システムタイマアプリケーションノート(英語)
<b>ref</b>		システムタイマコンフィグファイル(テンプレート)フォルダ
	r_sys_time_rx_config_reference.h	システムタイマ コンフィグファイル(テンプレート)
r_sys_time_rx_if.h		システムタイマヘッダファイル
readme.txt		readme

## 1.2 開発環境

システムタイマモジュールは以下の開発環境を用いて開発しました。ユーザアプリケーション開発時は以下のバージョン、またはより新しいものをご使用ください。

(1)統合開発環境

e<sup>2</sup> studio V5.2.0.020

(2)C コンパイラ

CC-RX V2.05.00

(3)エミュレータデバッガ

E1/E20

(4)評価ボード

Renesas Starter Kit for RX231: 製品型名 R0K505231S000BE (RX231 以外でも動作可能です)

評価は e2 studio で実施しました。サンプルプロジェクトは e2 studio で作成しました。

プロジェクト変換機能で e2 studio から CS+への変換が可能です。コンパイルエラー等問題が発生する場合はお問い合わせください。

## 1.3 コンパイラオプション情報

システムタイマモジュールは以下のデフォルトコンパイラオプションにて開発しました。

-debug -nologo -isa=rxv2 -fpu -alias=noansi (-include, -define を除く)

**【注】** システムタイマモジュールはリトルエンディアンのみで動作確認を行っています。コンパイラオプションに-endian=big を指定しないでください。

## 1.4 ROM / RAM サイズ情報

ROM 約 1.2KB、RAM 約 0.2KB

## 1.5 スタックサイズ情報

スタック 108 バイト (R\_SYS\_TIME\_SetCurrentTime()実行時)

## 1.6 セクション情報

システムタイマモジュールはデフォルトセクション (P, C) を使用します。

---

## 2. API 情報

---

---

### 2.1 ハードウェアの要求

---

システムタイマモジュールは、RX ファミリ内蔵の CMT 機能に依存します。

---

### 2.2 ソフトウェアの要求

---

システムタイマモジュールは、以下モジュールに依存します。

- r\_bsp rev3.40 以降をご使用ください。(BSP=Board Support Package)
- r\_cmt\_rx rev3.00 以降をご使用ください。(CMT=Compare Match Timer)

---

### 2.3 サポートされているツールチェーン

---

システムタイマモジュールは、以下のツールチェーンで動作を確認しています。

RX ファミリ用 C/C++コンパイラパッケージ V2.05.00

---

### 2.4 ヘッドファイル

---

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\_sys\_time\_rx\_if.h に記載しています。

---

### 2.5 整数型

---

このプロジェクトは ANSI C99 を使用しています。

---

### 2.6 コンフィグレーション

---

なし

---

### 2.7 API データ構造

---

システムタイマモジュールが使用するデータ構造体についての情報は r\_sys\_time\_rx\_if.h を参照してください。

---

## 2.8 戻り値

---

```
typedef enum e_sys_time_err
{
    SYS_TIME_SUCCESS=0,           /* Normally terminated. */
    SYS_TIME_BAD_CHANNEL,         /* Non-existent channel number. */
    SYS_TIME_BAD_INTERVAL,        /* Bad interval parameter is specified. */
    SYS_TIME_BAD_TIME_OFFSET,     /* Bad time offset is set. */
    SYS_TIME_BAD_FUNCTION_POINTER, /* Bad function pointer is set. */
    SYS_TIME_BAD_SYS_TIME,        /* Bad system timer value is input */
    SYS_TIME_ALREADY_STARTED,     /* System timer is already started. */
    SYS_TIME_NOT_STARTED,         /* System timer is not started. */
    SYS_TIME_FULL_REGISTERED,     /* All register table is used. */
    SYS_TIME_ALREADY_REGISTERED,  /* Specified function pointer has been
                                   already registered. */
}
sys_time_err_t;
```

---

## 2.9 モジュールの追加方法

---

本モジュールは既存の e2 studio プロジェクトに追加する必要があります。e2 studio plug-in を使用することによって自動的にインクルードファイルパスを更新することができるため、プロジェクトへの追加には plug-in の使用を推奨します。CS+の場合、手動でソースコードやインクルードパスの設定が必要です。

詳細は、e2 Studio/CS+に組み込む方法”[r01an1723ju0111\\_rx.pdf](#)/[r01an1826jj0102\\_rx.pdf](#)”をご参照ください。

### 3. API 関数

#### 3.1 API 一覧

表 3-1 API

API	説明
R_SYS_TIME_Open	システムタイマモジュールを開始します
R_SYS_TIME_GetCurrentTime	システムタイマからシステム時間を取得します
R_SYS_TIME_SetCurrentTime	システムタイマにシステム時間を設定します
R_SYS_TIME_ConvertUnixTimeToSystemTime	Unix 時間をシステムタイマの書式に変換します
R_SYS_TIME_RegisterPeriodicCallback	周期起動関数を登録します。(最大 30 個)
R_SYS_TIME_UnregisterPeriodicCallback	周期起動関数の登録を解除します
R_SYS_TIME_IsPeriodicCallbackRegistered	周期起動関数の登録の有無を確認します
R_SYS_TIME_Close	システムタイマモジュールを終了します
R_SYS_TIME_GetVersion	システムタイマモジュールのバージョンを取得します。

---

## 4. API 関数詳細説明

---

### R\_SYS\_TIME\_Open

---

#### Format

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_Open(void);
```

#### Parameters

なし

#### Return Values

SYS_TIME_SUCCESS	正常終了
SYS_TIME_BAD_CHANNEL	CMT のチャネルの空きが無い
SYS_TIME_ALREADY_STARTED	既に開始済み

#### Description

システムタイマモジュールを起動します。

#### Reentrant

この関数は再入可能（リエントラント）です。



---

**R\_SYS\_TIME\_GetCurrentTime**

---

**Format**

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_GetCurrentTime(SYS_TIME *sys_time);
```

**Parameters**

sys_time	入力/出力	システムタイマからシステム時間を取得する領域
----------	-------	------------------------

**Return Values**

SYS_TIME_SUCCESS	正常終了
------------------	------

**Description**

システムタイマからシステム時間を取得します。

**Reentrant**

この関数は再入可能（リエントラント）です。

---

**R\_SYS\_TIME\_SetCurrentTime**

---

**Format**

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_SetCurrentTime(SYS_TIME *sys_time);
```

**Parameters**

sys_time	入力/出力	システムタイマにシステム時間を設定する領域
----------	-------	-----------------------

**Return Values**

SYS_TIME_SUCCESS	正常終了
SYS_TIME_BAD_SYS_TIME	異常なシステム時間が設定されました。

**Description**

システムタイマにシステム時間を設定します。システム時間に内蔵される Unix 時間も更新します。

**Reentrant**

この関数は再入可能（リエントラント）です。

---

## R\_SYS\_TIME\_ConvertUnixTimeToSystemTime

---

### Format

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_ConvertUnixTimeToSystemTime(
    uint32_t unix_time, SYS_TIME *sys_time, uint8_t *time_offset);
```

### Parameters

unix_time	入力	Unix 時間
sys_time	入力/出力	システムタイマにシステム時間を設定する領域
time_offset	入力	タイムゾーンを指定する文字列

### Return Values

SYS_TIME_SUCCESS	正常終了
SYS_TIME_BAD_TIME_OFFSET	異常なタイムゾーンが設定されました。

### Description

Unix 時間を用いてシステムタイマにシステム時間を設定します。**time\_offset** にタイムゾーンを表す文字列を指定することで、システム時間をタイムゾーンに補正できます。タイムゾーンを表す文字列は、**r\_sys\_time\_rx\_if.h** に定義されています。

### Reentrant

この関数は再入可能（リエントラント）です。

---

**R\_SYS\_TIME\_RegisterPeriodicCallback**

---

**Format**

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_RegisterPeriodicCallback(
    callback_from_sys_time_t function_pointer, uint32_t interval)
```

**Parameters**

function_pointer	入力	関数ポインタ
interval	入力	起動周期 (単位=10ms)

**Return Values**

SYS_TIME_SUCCESS	正常終了
SYS_TIME_BAD_FUNCTION_POINTER	異常な関数ポインタが指定された
SYS_TIME_BAD_INTERVAL	異常な起動周期が指定された
SYS_TIME_FULL_REGISTERED	関数ポインタ登録数の上限に達した
SYS_TIME_ALREADY_REGISTERED	既に登録済みの関数ポインタが指定された

**Description**

ユーザは本関数を使用し起動周期を指定して関数ポインタを登録することで、定常的に行う必要のある処理をシステムタイマに実行させることができます。最大 30 個の関数ポインタを登録することができます。関数ポインタは CMT 割り込み内で起動されます。この CMT 割り込みでは多重割り込みを許可しています。リアルタイム性の必要な処理は、CMT 割り込みの優先度(r\_cmt\_rx\_config.h で設定)より高い優先度の割り込みで処理してください。

**Reentrant**

この関数は再入可能（リエントラント）です。

---

**R\_SYS\_TIME\_UnregisterPeriodicCallback**

---

**Format**

```
#include "r_sys_time_rx_if.h"
```

```
sys_time_err_t R_SYS_TIME_UnregisterPeriodicCallback(callback_from_sys_time_t function_pointer);
```

**Parameters**

function\_pointer 入力                      関数ポインタ

**Return Values**

SYS\_TIME\_SUCCESS

正常終了

SYS\_TIME\_BAD\_FUNCTION\_POINTER

異常な関数ポインタが指定された

**Description**

周期起動関数の登録を解除できます。

**Reentrant**

この関数は再入可能（リエントラント）です。

---

**R\_SYS\_TIME\_IsPeriodicCallbackRegistered**

---

**Format**

```
#include "r_sys_time_rx_if.h"
```

```
bool R_SYS_TIME_IsPeriodicCallbackRegistered(callback_from_sys_time_t function_pointer);
```

**Parameters**

function_pointer 入力	関数ポインタ
---------------------	--------

**Return Values**

true	登録済み
false	登録無し

**Description**

周期起動関数の登録を確認できます。

**Reentrant**

この関数は再入可能（リエントラント）です。

---

## R\_SYS\_TIME\_Close

---

### Format

```
#include "r_sys_time_rx_if.h"

sys_time_err_t R_SYS_TIME_Close(void);
```

### Parameters

なし

### Return Values

SYS_TIME_SUCCESS	正常終了
SYS_TIME_BAD_CHANNEL	CMT チャネルのクローズに失敗
SYS_TIME_NOT_STARTED	システムタイマが起動していない

### Description

システムタイマを停止します。

### Reentrant

この関数は再入可能（リエントラント）です。

---

## R\_SYS\_TIME\_GetVersion

---

### Format

```
#include "r_sys_time_rx_if.h"

uint32_t R_SYS_TIME_GetVersion(void);
```

### Parameters

なし

### Return Values

本モジュールのバージョン

### 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。

### Reentrant

この関数は再入可能（リエントラント）です。



## 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/>

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	2016/11/30	-	新規作成

## 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（予約領域）のアクセス禁止

【注意】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、将来の機能拡張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。

### 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、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、内部ROM、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。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。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事情報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が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 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■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/>