

RXファミリ

R01AN3503JJ0105

Rev.1.05

Mar 1, 2021

USBマストレージ経由内蔵FlashROM書き換えプログラム

要旨

このアプリケーションノートでは、USBメモリを使用したFlashROM書き換えプログラムについて説明します。

対象デバイス

RX111, RX113, RX231, RX23W

RX62N/RX621, RX63N/RX631, RX63T

RX64M, RX71M, RX65N/RX651, RX66T/RX72T

RX72M, RX72N, RX66N, RX671

目次

1. 資料概要	2
2. USBマストレージ経由内蔵FlashROM書き換えプログラム概要	4
3. USBマストレージ経由内蔵FlashROM書き換えプログラムのセットアップ	9
4. USBマストレージ経由内蔵FlashROM書き換えプログラムの実行	11
5. ユーザプログラム開発時の注意事項	21
6. USBマストレージ経由内蔵FlashROM書き換えプログラムとユーザプログラムの設定	24
7. USBメモリへのユーザプログラムの書き込み	30
8. USBマストレージ経由内蔵FlashROM書き換えプログラムについて	31
9. e ² studio用プロジェクトをCS+で使用する場合	42

1. 資料概要

本書は、USB Mass Storage(USB Host)を使用した内蔵 FlashROM 書き換えプログラムのアプリケーションノートです。本書は「1.2 関連ドキュメント」と併用してご利用ください。

1.1 機能

本プログラムは USB メモリに格納したユーザプログラム(mot ファイル)を USB(Mass Storage Class)経由で MCU の内蔵 FlashROM に書き込みを行います。

1.2 関連ドキュメント

1. Universal Serial Bus Revision 2.0 specification
2. RX Family Flash Module Using Firmware Integration Technology アプリケーションノート
3. RX ファミリ ボードサポートパッケージモジュール アプリケーションノート
4. 各 MCU ユーザーズマニュアル ハードウェア編

ルネサス エレクトロニクスホームページより入手できます。

ルネサス エレクトロニクスホームページ

【<http://japan.renesas.com/>】

USB デバイスページ

【<http://japan.renesas.com/usb/>】

1.3 注意事項

1. 本アプリケーションノートは、動作を保証するものではありません。本アプリケーションノートをシステムに適用される場合は、お客様における動作検証は十分に実施いただきますようお願いいたします。
2. ご使用になる USB メモリによっては、USBマストレージ経由内蔵FlashROM書き換えプログラムが正常に動作しない場合があります。
3. USB Hub はご使用いただけません。
4. 必ず FAT32 でフォーマットした USB メモリを使用してください。
5. 本プログラムは、Little Endian 設定になっています。ユーザプログラムが Big Endian の場合は、本プログラムの Endian も Big Endian に変更して下さい。Endian 設定については、「6.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの設定」を参照してください。
6. 本プログラムをお客様のシステムに実装する場合は、必ず「6 USBマストレージ経由内蔵FlashROM書き換えプログラムとユーザプログラムの設定」、「8.4 注意事項」を参照してください。
7. 本プログラムは、各 FIT モジュールを使用しています。本プログラムでは Web 公開中の FIT モジュールのソースコードに対し変更を行っています。
8. RSSK(RX23W)をご使用の場合、RSSK に対し以下の抵抗の移動が必要です。

R89	-->	R90
R96	-->	R97
R112	-->	R113

9. 本書に記載された「USB0 モジュール」および「USB1 モジュール」という用語は、MCU ごとに示すモジュールが異なりますので、以下を参照してください。

用語	MCU	USB モジュール名
USB0 モジュール (開始アドレス:0xA0000)	RX62N/RX621	USB モジュール
	RX63N/RX631	USBa モジュール
	RX63T	USBa モジュール
	RX65N/RX651	USBb モジュール
	RX64M	USBb モジュール
	RX71M	USBb モジュール
	RX72M	USBb モジュール
	RX72N	USBb モジュール
	RX66N	USBb モジュール
	RX66T/RX72T	USBb モジュール
	RX671	USBb モジュール
	RX111	USBc モジュール
	RX113	USBc モジュール
	RX231	USBd モジュール
	RX23W	USBc モジュール
USB1 モジュール (開始アドレス:0xA0200 / 0xD0400)	RX62N/RX621	USB モジュール
	RX64M	USBA モジュール
	RX71M	USBAa モジュール
	RX671	USBb モジュール

1.4 用語一覧

本書で使用される用語と略語は以下のとおりです。

API	: Application Program Interface
BSP	: Renesas Board support package module
e ² studio	: Eclipse embedded studio (RX対応)
H/W	: Renesas USB device
MCU	: Micro control Unit
MSC	: Mass Storage Class
RSK	: Renesas Starter Kit
RSSK	: Renesas Solution Starter Kit
USB	: Universal Serial Bus

2. USBマストレージ経由内蔵FlashROM書き換えプログラム概要

2.1 概要

本プログラムは、USBメモリ内にあるユーザプログラム(motファイル)をUSB(Mass Storage Class)経由でRSK/RSSKに転送し、転送されたユーザプログラムは、Flash Self programmingライブラリを使用してROM上の任意の場所に書き込まれます。

本プログラムの構成は、次の通りです。

1. USBマストレージ経由内蔵FlashROM書き換えプログラム
2. 動作確認用サンプルプログラム

動作確認のためのファイルです。これらのプログラムはUSBマストレージ経由内蔵FlashROM書き換えプログラムによって内蔵FlashROMに書き込まれます。

次に、プログラムのデータの流れを表します。

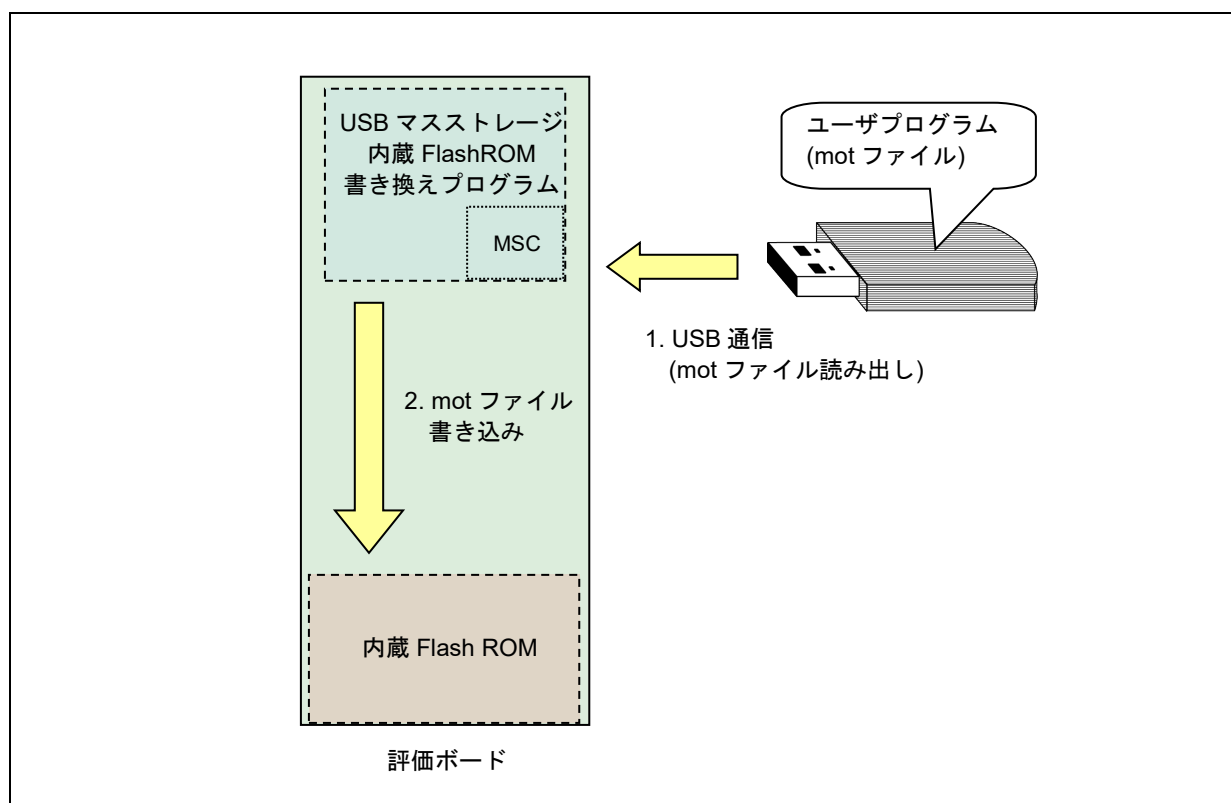


Figure 2-1 FlashROM 書き換えプログラムのデータの流れ

特定条件下でRSK/RSSKを起動後、USBメモリを接続するとUSBマストレージ経由内蔵FlashROM書き換えプログラムが動作し、USBメモリに格納されたユーザプログラム(motファイル)が内蔵Flash ROMに書き込まれます。

2.2 特長

本プログラムの特長を以下に示します。

1. 本プログラムは **USB Host** モードで動作します。
2. 本プログラムは、以下の **USB** 規格をサポートしています。
 - (1). **USB 2.0** 規格の **Full-speed** 転送に対応
 - (2). **USB** デバイスクラスは、**Mass Storage Class** に対応
 - (3). **Mass Storage Sub Class** は、**SFF-8070i** に対応
3. 本プログラムは、内蔵 **FlashROM** に対する消去およびユーザプログラムの書き込みを行います。
4. 本プログラムがサポートしているユーザプログラムのフォーマットは、モトローラ **S** フォーマットです。
5. 本プログラムが配置されている領域を除いたすべての内蔵 **FlashROM** 領域にユーザプログラムの配置が可能です。
6. 本プログラムは **FlashROM** に対する **Write & Verify** をサポートしています。
7. デュアルモードをサポートしています。(デュアルモードについては、デュアルモードをサポートする **MCU** のハードウェアマニュアルを参照してください。)
8. バックアップ機能をサポートしています。バックアップ機能の詳細については、「**8.1 バックアップ機能**」を参照してください。
9. ユーザプログラムは、すべての割り込みを使用できます。

2.3 ソフトウェア構成

USBマストレージ経由内蔵FlashROM書き換えプログラムのソフトウェア構成をFigure 2-2に示します。

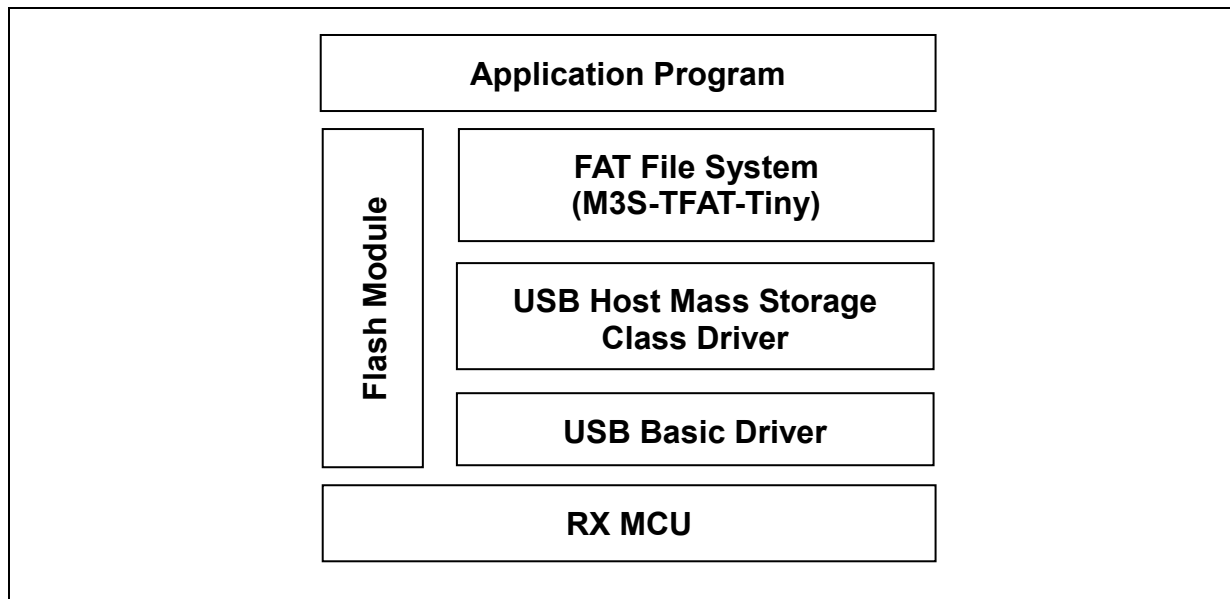


Figure 2-2 USBマストレージ経由内蔵FlashROM書き換えプログラムソフトウェア構成

Application Program では、以下の処理が行われます。

1. FAT ファイルシステムを使って USB メモリ内にあるユーザプログラム(mot ファイル)の読み出し処理を行います。
2. ユーザプログラム(mot ファイル)の解析を行います。
3. Flash Module を使って、FlashROM に対する消去および書き込み処理を行います。

2.4 ROM サイズ

本プログラムが使用する ROM サイズを以下に示します。

ROMサイズ : 29Kバイト

Note:

USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域に配置する場合のROMサイズは以下の通りです。

ROMサイズ : 32Kバイト

2.5 対象デバイスと FLASH TYPE

RX の FLASH Type は 4 タイプ存在します。ご使用の MCU が、どの Flash Type かは下記の表を参照下さい。詳しくは RX Family Flash Module Using Firmware Integration Technology アプリケーションノートをご参照下さい。

Table 2-1 MCU の Flash 書き込みタイプ

Flash 書き込みタイプ	対応デバイス
Flash Type1	RX111, RX113, RX231, RX23W
Flash Type2	RX62N, RX63N/RX631, RX63T
Flash Type3	RX64M, RX71M, RX66T/RX72T
Flash Type4	RX65N/RX651, RX72M, RX72N, RX66N, RX671

2.6 動作確認環境

本プログラムは、下記環境にて動作確認を行っております。

1. ハードウェア環境

- (1). 評価ボード RSK/RSSKボード
- (2). MCU RX71M, RX64M, RX63N, RX651, RX62N, RX63T, RX111, RX113, RX231, RX72T, RX72M, RX72N, RX66N, RX23W, RX671
- (3). エミュレータ E2 Lite

Note:

RX23Wでは、RSSKボードを使用しています。

2. ソフトウェア環境

- (1). 統合開発環境 e² studio
- (2). コンパイラ RXファミリ用C/C++コンパイラパッケージ CC-RX V.3.01
- (3). FATファイルシステム Open Source FAT File System [M3S-TFAT-Tiny] Module Firmware Integraion (R20AN0038)
- (4). Flashライブラリ Flash Module Using Firmware Integration Technology(R01AN2184)
- (5). Host Mass Storage Class USB Host Mass Storage Class Driver for USB Mini Firmware Integration Technology (R01AN2169)
- (6). Flash書き込みツール Renesas Flash Programmer

Note:

- a. RX62N では、USB0 モジュールを使った動作確認を行っていません。
- b. RX671 では、RX ファミリ用 C/C++コンパイラパッケージ CC-RX V.3.03 を使って動作確認を行いました。

2.7 フォルダ構成

本プログラムのフォルダ構成を示します。

(Top Directroy)

+-reference

| +-SampleProgram (動作確認用サンプルプログラム)

| +- (MCU名)

+-workspace (USBマストレージ経由内蔵FlashROM書き換えプログラムサンプルプロジェクト一式)

+-(MCU名_FirmwareUpdater)

次に、各フォルダの説明を示します。

(1). reference*SampleProgram

サンプル・ユーザプログラムが格納されているフォルダです。

sample.mot : 下記の順番で LED が点灯します。

LED表示				
LED3	LED2	LED1	LED0	順序
○	●	●	●	
●	○	●	●	
●	●	○	●	
●	●	●	○	

(2). workspace

各MCUのUSBマストレージ経由内蔵FlashROM書き換えプログラムが格納されているフォルダです。本フォルダ下のフォルダ構成の詳細については、「8.3 フォルダ/ファイル構成」を参照してください。

3. USBマストレージ経由内蔵FlashROM書き換えプログラムのセットアップ

この章では、本プログラムのセットアップ手順を説明します。

3.1 プロジェクトのセットアップ

workspace フォルダ下からご使用の MCU と同じ名前のフォルダ名を選択し、以下の手順に沿ってプロジェクトのセットアップを行ってください。なお、以下の手順は e² studio を使ったセットアップ手順になります。

- (1). e² studio を起動してください。

※ はじめてe² studio を起動する場合、Workspace Launcher ダイアログが表示されますので、プロジェクトを格納するためのフォルダを指定してください。

- (2). [ファイル] → [インポート]を選択してください。インポートの選択ダイアログが表示されます。
- (3). インポートの選択画面で、[既存プロジェクトをワークスペースへ]を選択してください。

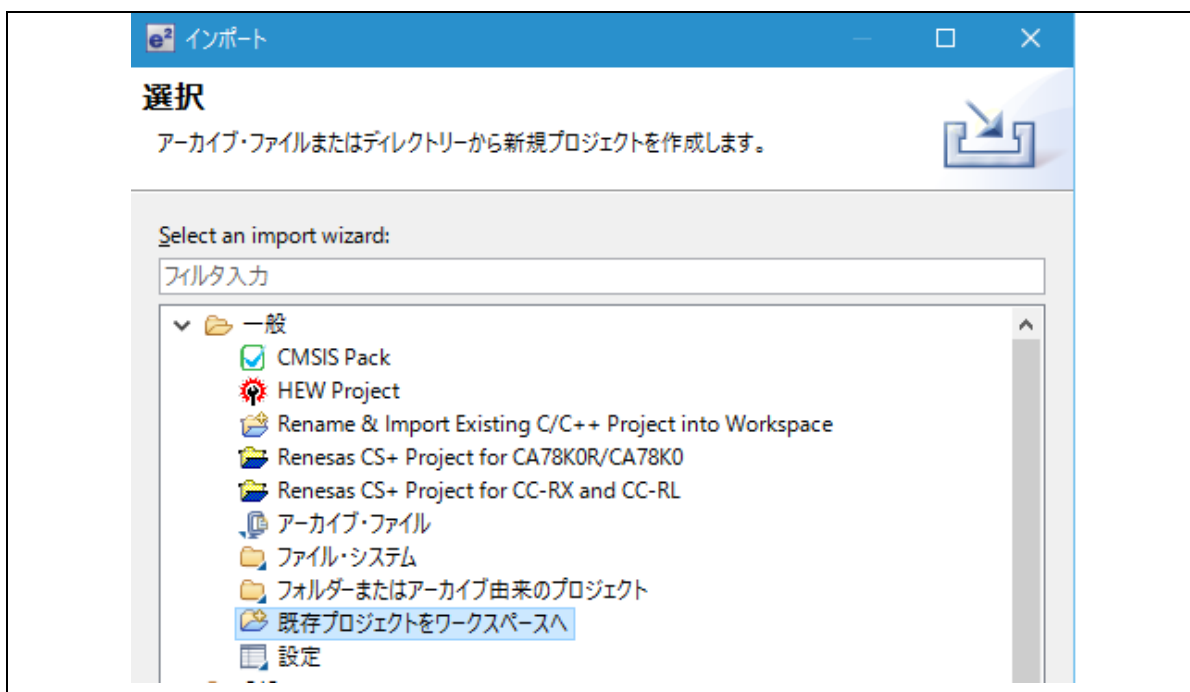


Figure 3-1 インポートの選択

- (4). [ルートディレクトリの選択] の [参照] ボタンを押下して、「.cproject」(プロジェクトファイル) が格納されたフォルダを選択して下さい。

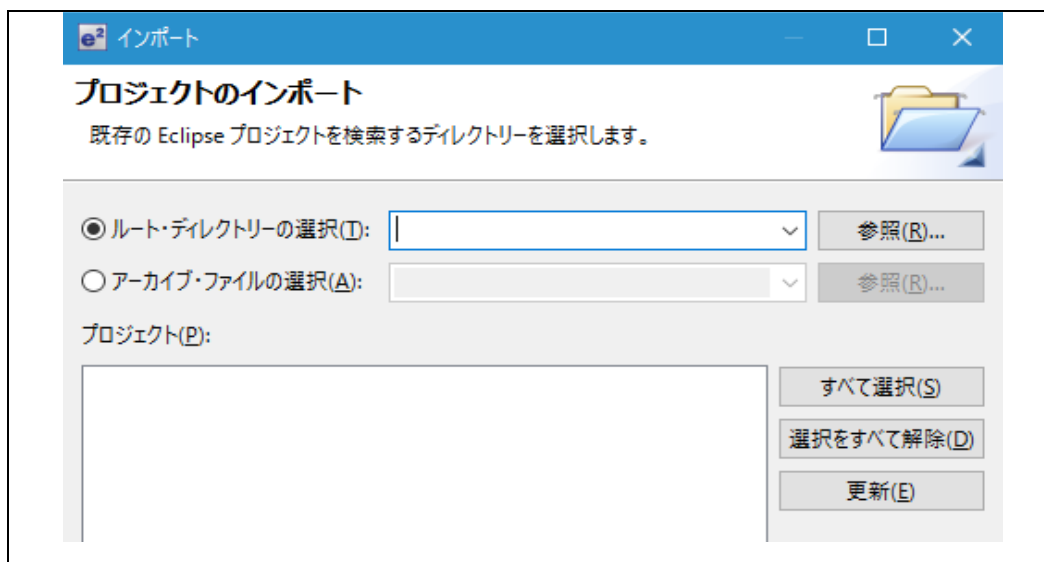


Figure 3-2 プロジェクトのインポート画面

(5). [終了]をクリック して下さい。

プロジェクトのワークスペースへのインポートが完了します。

Note:

デュアルモードをサポートする MCU をリニアモードでご使用になる場合、**Figure 3-3**内の"デバイスの変更"(赤枠部分)よりリニアモード用のデバイスへ変更してください。たとえば、R5F565NEHxFB のデバイスをご使用の場合、R5F565NEHxFB_DUAL(デュアルモード)から R5F565NEHxFB(リニアモード)へデバイス変更してください。

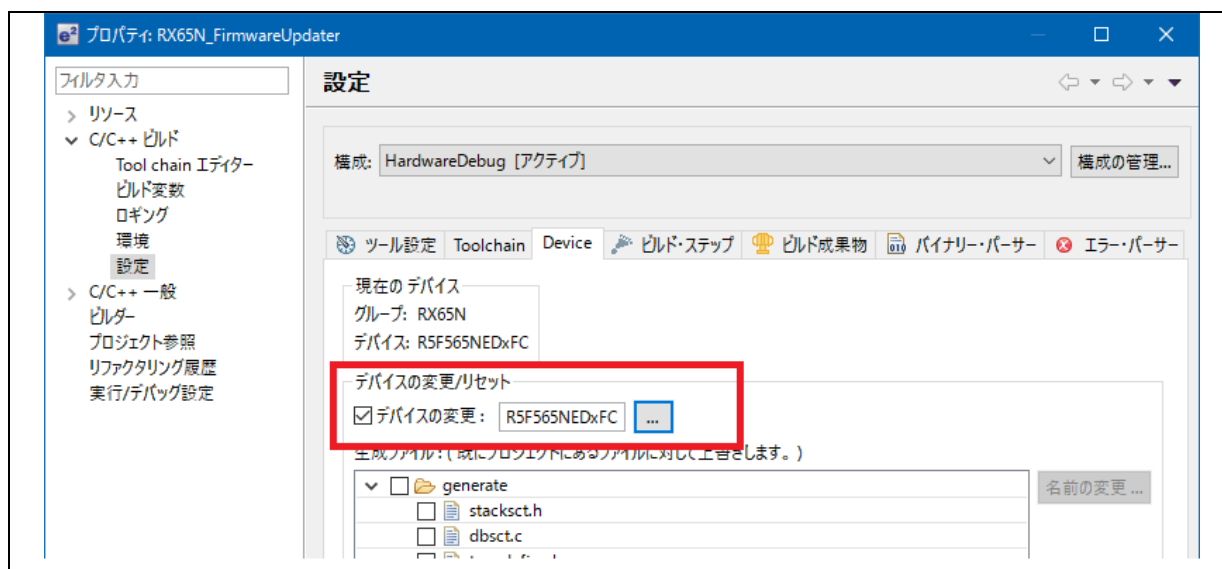


Figure 3-3 デバイスの変更画面

4. USBマストレージ経由内蔵FlashROM書き換えプログラムの実行

本プログラムの実行方法について説明します。

ここではRSK/RSSKボードを使って、動作確認用サンプルプログラムが動作することを確認します。

4.1 USBマストレージ経由内蔵FlashROM書き換えプログラムのROM書き込みおよび実行

この章では、本プログラムを実行し、書き換え処理を行うときの手順を説明します。

4.1.1 USBマストレージ経由内蔵FlashROM書き換えプログラムのROM書き込み

(1). ハードウェアのセットアップ

USBマストレージ経由内蔵FlashROM書き換えプログラムをご使用のMCUに書き込む場合の接続図を以下に示します。

a. エミュレータを使用する場合

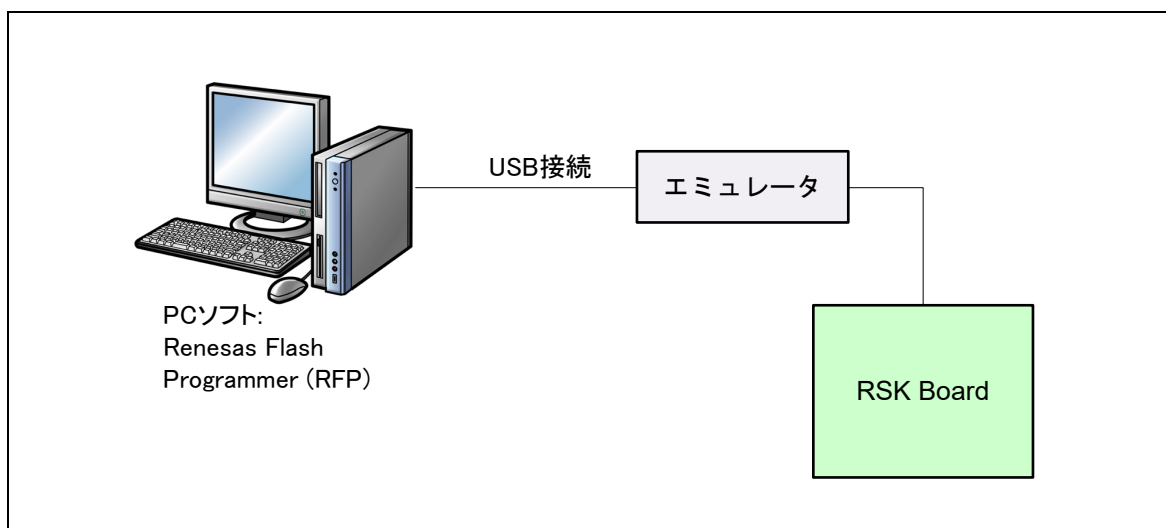


Figure 4-1 エミュレータを使用する場合の接続図

b. エミュレータを使用しない場合

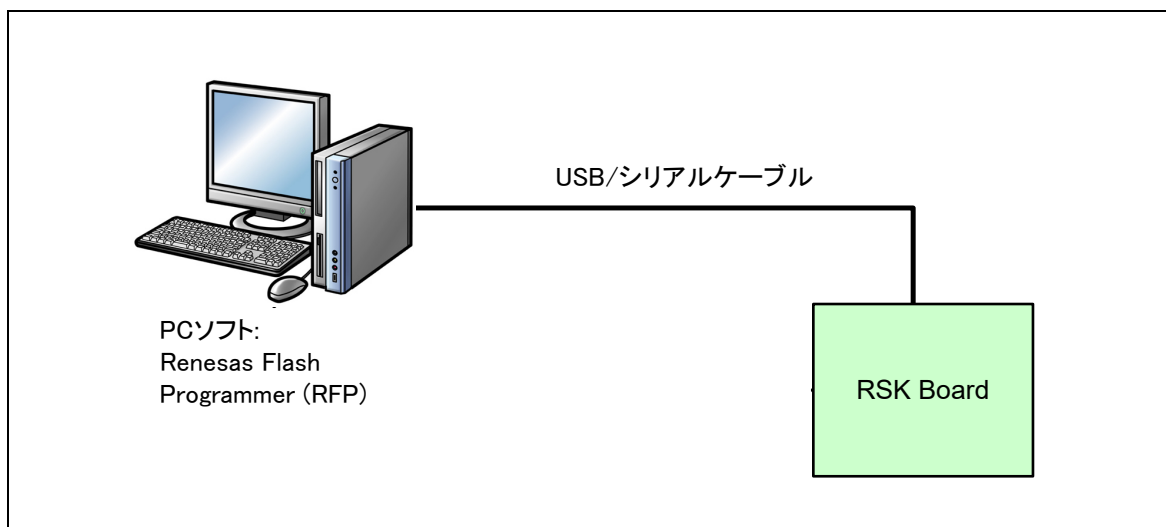


Figure 4-2 エミュレータを使用しない場合の接続図

(2). USBマストレージ経由内蔵FlashROM書き換えプログラムの書き込み

Renesas Flash Programmer(RFP)を起動し、「ユーザ/データエリア」の参照ボタンから Workspace¥(MCU名)フォルダにあるUSBマストレージ経由内蔵FlashROM書き換えプログラムの書き込みファイルを選択します。スタートを押下すると、ターゲットボードにプログラムがダウンロードされます。出力パネルに **PASS** と表示され、パネル上部に緑地で「正常終了」と表示されれば、書き込みは完了です。

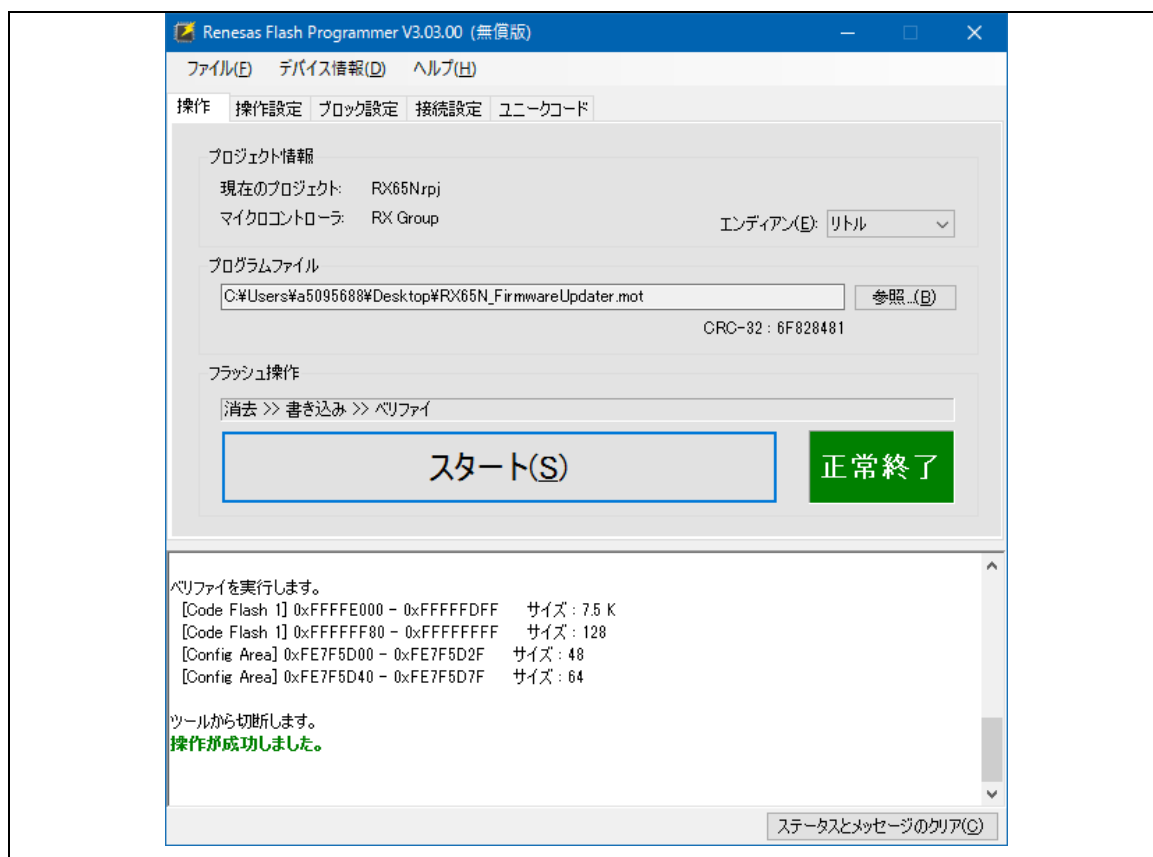


Figure 4-3 ファイルの指定

Note:

- a. Renesas Flash Programmer)についての詳細は、以下のURLを参照してください

URL: (日本語)

<https://www.renesas.com/ja-jp/products/software-tools/tools/programmer/renesas-flash-programmer-programming-gui.html>

URL: (英語)

<https://www.renesas.com/en-us/products/software-tools/tools/programmer/renesas-flash-programmer-programming-gui.html>

- b. USBマストレージ経由内蔵FlashROM書き換えプログラムの配置については「4.1.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの配置」を参照してください。
- c. Flash ROM (プログラムROM領域)の全ブロックが消去される設定を行ってください。

(3). 更新対象領域への FlashROM 書き換えプログラムのコピー (デュアルモード選択時)

上記(2)によりUSBマストレージ経由内蔵FlashROM書き換えプログラムの書き込みが完了した後、RSK/RSSK を電源投入/リセットすると、起動バンク内のUSBマストレージ経由内蔵FlashROM書き換えプログラムが更新対象領域へ書き換えプログラム自身のコピーを行います。

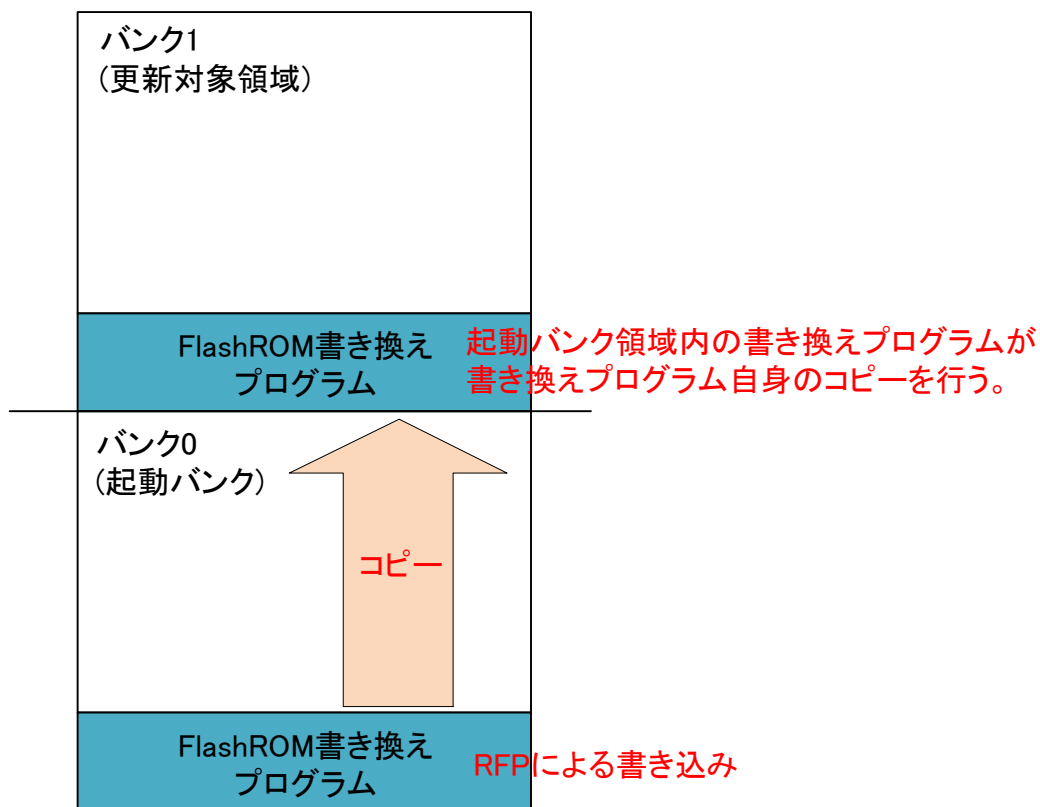


Figure 4-4 FlashROM 書き換えプログラムのコピー

Note:

このコピー処理が失敗した場合、以下の点灯パターンが点滅します。

●: 消灯 ○: 点滅

点灯パターン			
LED3	LED2	LED1	LED0
○	○	○	●

4.1.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの配置

本プログラムの配置アドレスについて説明します。

(1). ユーザブート領域以外の ROM 領域に配置する場合

USBマストレージ経由内蔵FlashROM書き換えプログラムは、以下の領域に配置してください。

USBマストレージ経由内蔵FlashROM書き換えプログラム配置領域		
0xFFFF8000	-	0xFFFFFFFF

なお、デュアルモード使用時、USBマストレージ経由内蔵FlashROM書き換えプログラムが自分自身を以下の領域にコピーします。

Table 4-1 ROM 容量 2MB の場合

USBマストレージ経由内蔵FlashROM書き換えプログラム配置領域		
0xFFEF8000	-	0xFFEFFFFFF

Table 4-2 ROM 容量 4MB の場合

USBマストレージ経由内蔵FlashROM書き換えプログラム配置領域		
0xFFDF8000	-	0xFFDFFFFFF

下記はRX65Nのメモリマップです。メモリマップの詳細に関しては、対象MCUのユーザーズマニュアル ハードウェア編 を参照してください。

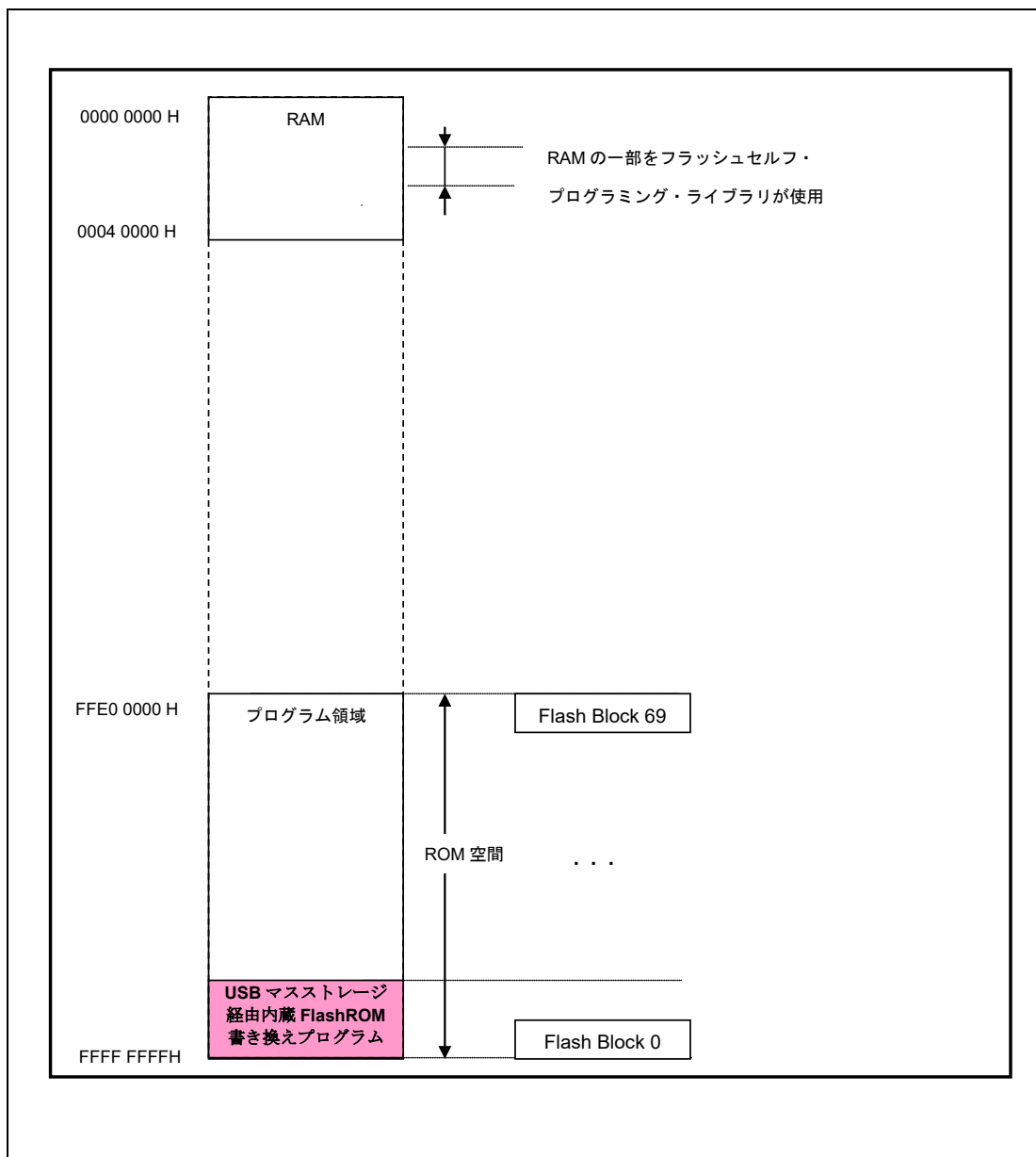


Figure 4-5 メモリマップ (ユーザブート領域非使用時&デュアルモード非使用時)

デュアルモード使用時のメモリマップは以下の通りです。

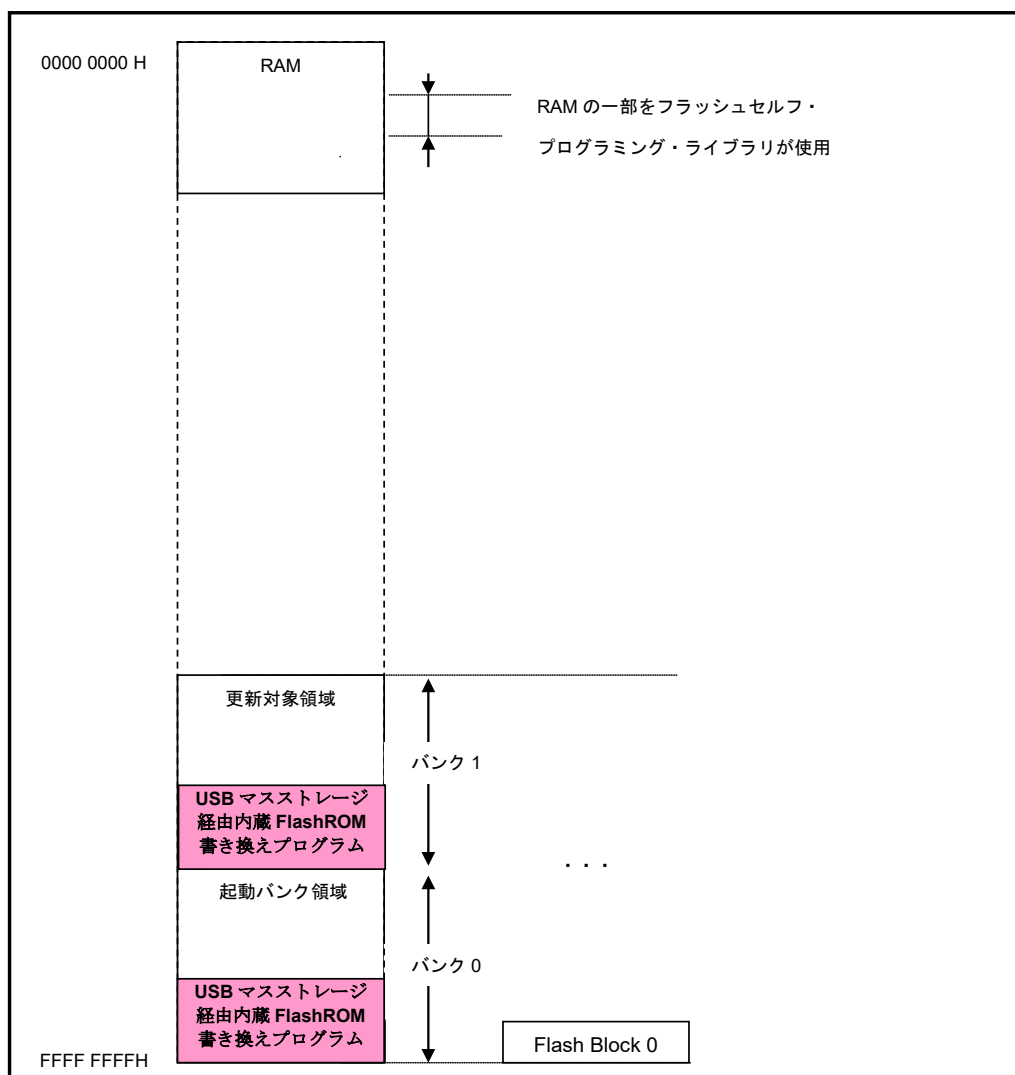


Figure 4-6 メモリマップ (デュアルモード使用時)

Note:

1. USBマストレージ経由内蔵FlashROM書き換えプログラムをコンパイルする場合は、e² studio 上の[関数の分岐幅]に対し 24bit を選択してください。なお、[分岐幅]は、[ファイル]→[プロパティ]→[C/C+ビルド]→[設定] を選択後、[Common]→[CPU] で指定できます。
2. Dual モードの場合、USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域に配置することはできません。

(2). ユーザブート領域に配置する場合

ご使用の MCU が RX71M、RX64M または RX66T/RX72T の場合、USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域に配置することができます。ユーザブート領域については、Table 4-3を参照してください。

Table 4-3 MCU のユーザブート領域情報

MCU	サイズ	user boot address
RX71M	32KB	0xFF7F8000 - 0xFF7FFFFF

RX64M RX66T/RX72T				
----------------------	--	--	--	--

Note:

1. USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域に配置してコンパイルする場合は、**e² studio** 上の[分岐幅]の設定に対し「指定しない」を選択してください。（デフォルトは「24ビット以内」になっています） なお、[分岐幅]は、[ファイル]->[プロパティ]->[C/C+ビルド]->[設定] を選択後、[Common]->[CPU] で指定できます。
2. ご使用のMCUがRX62N/RX621, RX63N/RX631, RX63Tの場合、ユーザブート領域にUSBマストレージ経由内蔵FlashROM書き換えプログラムを配置できません。

下記はUSBマストレージ経由内蔵FlashROM書き換えプログラムをRX64Mのユーザブート領域に配置した場合のメモリ・マップになります。

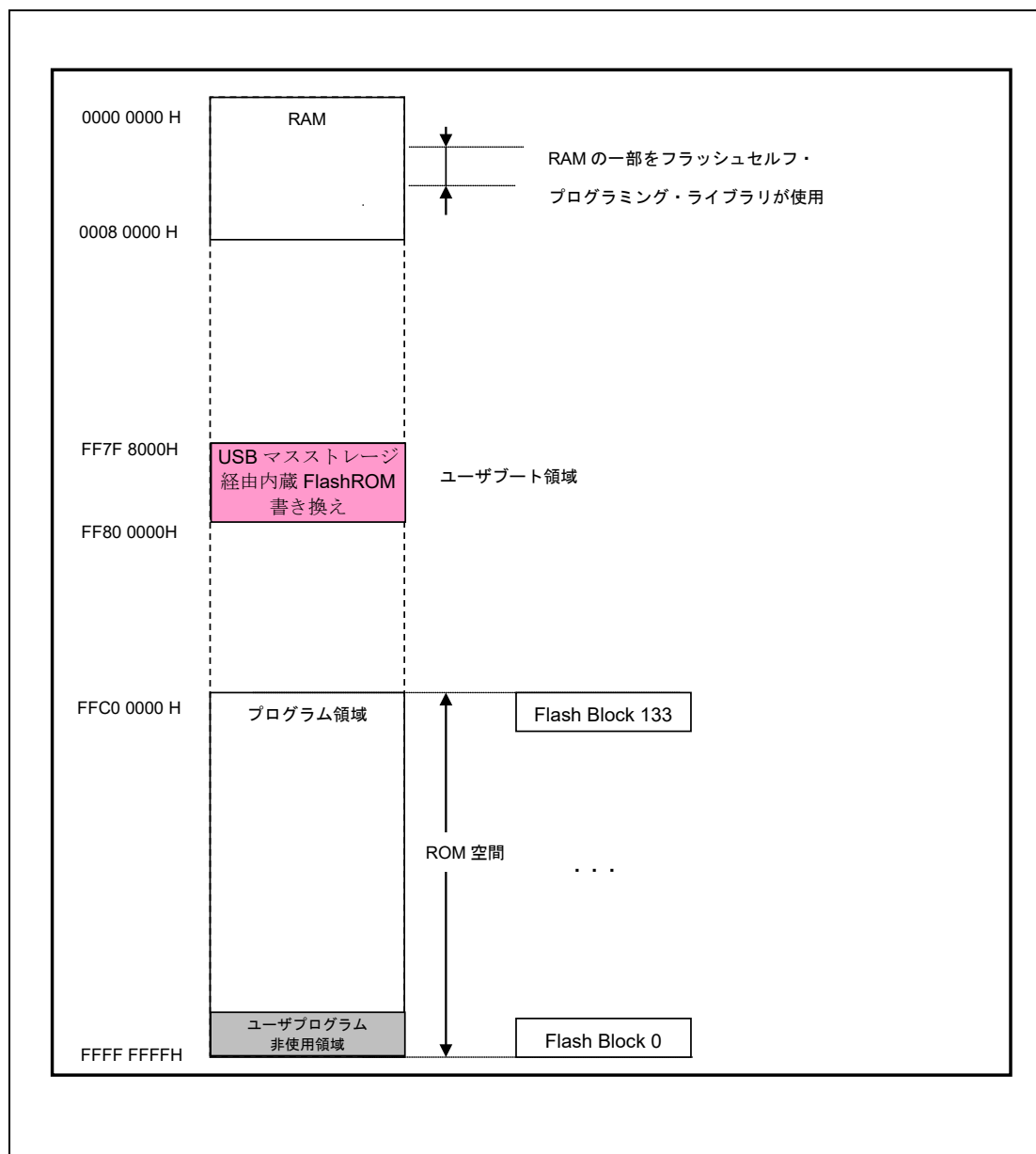


Figure 4-7 メモリ・マップ (ユーザブート領域使用時)

4.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの実行(mot ファイル書き込み)

本章では、USBマストレージ経由内蔵FlashROM書き換えプログラムを実行し、mot ファイル(ユーザプログラム)の書き込み手順について説明します。

1. USBマストレージ経由内蔵FlashROM書き換えプログラムの起動

RSK/RSSK 上のスイッチ(RSK:スイッチ 3, RSSK:スイッチ 2)を押下しながらリセット・ボタンを押してください。USBマストレージ経由内蔵FlashROM書き換えプログラムが起動し、USB メモリの接続を待ちます。この時、RSK/RSSK 上のLED0 が点滅します。なお、FlashROM および USB モジュールの初期化処理に問題が発生した場合は、LED1、LED2 と LED3 が点滅します。これらのLED が点滅しているときにUSB メモリを接続してもUSBマストレージ経由内蔵FlashROM書き換えプログラムは動作しません。

2. USB メモリの接続

ユーザプログラム(mot ファイル)を格納した USB メモリを RSK/RSSK に接続してください。USBマストレージ経由内蔵FlashROM書き換えプログラムは、内蔵 FlashROM へのユーザプログラム(mot ファイル)の書き込み処理を開始します。

(1). 内蔵 FlashROM 消去

本プログラムが FlashROM を消去している間、以下の RSK/RSSK 上のLED(LED0 と LED1)が点灯します。


●: 消灯 ○:点灯

LED表示			
LED3	LED2	LED1	LED0
●	●	○	○

(2). ユーザプログラム(mot ファイル)書き込み処理中

FlashROM に対する書き込み処理中、RSK/RSSK 上のLED はシフト表示されます。

●: 消灯 ○:点灯

LED表示				
LED3	LED2	LED1	LED0	順序
●	●	●	○	
●	●	○	●	
●	○	●	●	
○	●	●	●	

Note:

- ユーザプログラム書き込み中に USB メモリの Detach を行わないでください。書き込み中に USB メモリを Detach した場合、RX MCU をリセットする必要があります。
- FlashROM 消去中、USBマストレージ経由内蔵FlashROM書き換えプログラムは、LED に対する処理を行いませんので、USB メモリ接続後、最大 15 秒程度、LED のシフト表示が開始されません。

(3). ユーザプログラム(mot ファイル)書き込み完了

FlashROM に対する書き込み処理が完了(正常終了)すると、USBマストレージ経由内蔵FlashROM書き換えプログラムはソフトウェアリセットを行い、書き込んだユーザプログラムを起動します。

Note:

RX62N はソフトウェアリセットをサポートしていませんので、RX62N をご使用の場合、内蔵 FlashROM へのユーザプログラム(mot ファイル)の書き込み完了後、RSK/RSSK ボードをリセットするとユーザプログラムが動作します。なお、FlashROM に対する書き込みが完了(正常終了)すると、RSK/RSSK 上の LED はブリンク表示されます(下記参照)。

●: 消灯 ○:点灯

LED表示				
LED3	LED2	LED1	LED0	順序
●	●	●	●	
○	○	○	○	

(4). ユーザプログラム(mot ファイル)書き込みエラー

FlashROM に対する書き込み処理が異常終了した場合、以下の点灯パターンが点滅します。以下の点灯パターンが表示された場合は、USBマストレージ経由内蔵FlashROM書き換えプログラムを再起動してください。

●: 消灯 ○:点滅

点灯パターン				内容
LED3	LED2	LED1	LED0	
●	●	●	○	アクセス可能なドライブが検出されなかった。
●	●	○	●	ドライブマウント処理に失敗した。
●	●	○	○	ファイルオープン処理に失敗した。(Note a)
●	○	●	●	ファイル読み出し処理に失敗した。(Note d)
●	○	●	○	ファイルクローズ処理に失敗した。
●	○	○	●	チェックサムエラー
●	○	○	○	フォーマットエラー
○	●	●	●	FlashROM Eraseエラー
○	●	●	○	FlashROM書き込みエラー
○	●	○	●	アドレスエラー(Note b)
○	●	○	○	書き込みデータベリファイエラー
○	○	●	●	ファイルエンドエラー (Note c)
○	○	○	●	その他のエラー(Note e)

Note:

- a. ユーザプログラム(mot ファイル)のファイル名が 8.3 形式以外の場合もこのエラーになります。
- b. 書き込みエリア外への書き込みデータを受信した場合のエラー。
- c. FAT ライブラリでファイルの終端を検出したにも関わらず S タイプフォーマットのエンドコードを受信しなかった場合のエラー。
- d. FlashROM の Erase またはユーザプログラムの書き込み処理中に USB メモリがデタッチされた場合もこの点灯パターンが表示されます。
- e. 以下のいずれかのエラーです。
 - a) ユーザプログラム(mot ファイル)中にあるセキュリティコードとUSBマストレージ経由内蔵FlashROM書き換えプログラムに登録されたセキュリティコードが不一致。
 - b) ユーザプログラム(mot ファイル)内にオプション設定メモリ領域が含まれている。

4.3 ユーザプログラム書き込み時の注意事項

1. USBマストレージ経由内蔵FlashROM書き換えプログラムが書き込まれた領域にユーザプログラムを上書きしてしまった場合、USBマストレージ経由内蔵FlashROM書き換えプログラムの書き込みからやり直して下さい。

※ MCUによってROMのEraseブロック単位が異なりますので、ご注意ください。

2. 内蔵FlashROMに対する消去または書き込みを行っている間に瞬停やMCU Resetが発生した場合、再度「4.1.1 **USBマストレージ経由内蔵FlashROM書き換えプログラムのROM書き込み**」の作業が必要になる場合があります。
3. 固定ベクタを含むブロックをEraseしないようご注意ください。固定ベクタが消去されるとUSBマストレージ経由内蔵FlashROM書き換えプログラムが起動できなくなります。

5. ユーザプログラム開発時の注意事項

この章では、ユーザプログラムを開発するうえで注意すべき事項について説明します。

5.1 UserApp Header 領域 (ユーザ・アプリケーション・ヘッダ)

USBマストレージ経由内蔵FlashROM書き換えプログラムを使用してユーザプログラムを書き込む場合、そのユーザプログラムにはUserApp Header(ユーザ・アプリケーション・ヘッダ)領域が必要になります。UserApp Header領域のサイズは、ユーザプログラムのスタートアドレス格納領域(4バイト)とセキュリティコード格納領域(4バイト)の計8バイトです。(Figure 5-1参照)

UserApp Header領域の作成については、「6.1 ユーザプログラムに対する設定」を参照してください。

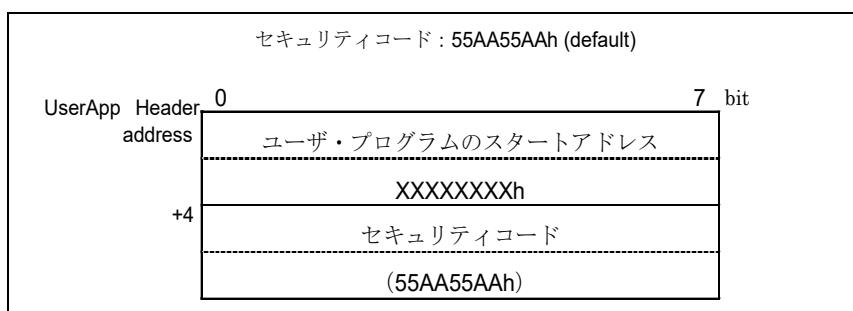


Figure 5-1 UserApp Header 領域

このヘッダ情報をUSBマストレージ経由内蔵FlashROM書き換えプログラムが起動時に読み取ることにより、UserAppの起動シーケンスに移行します。詳しくは、「8.2.1 電源投入時/リセット時の動作フロー」を参照してください。

5.2 オプション設定メモリ

ユーザプログラム(mot ファイル)内でオプション設定メモリに対する設定は行わないでください。オプション設定メモリの設定を行っているユーザプログラムのFlashROM書き込みを行った場合、書き込みエラーとなります。この書き込みエラーは、RSK/RSSK上にLEDの点灯パターンによって確認できます。LEDの点灯パターンについては、「4.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの実行(motファイル書き込み)」を参照してください。

Note:

オプション設定メモリの設定は、USBマストレージ経由内蔵FlashROM書き換えプログラムに対して行ってください。詳細については、「6.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの設定」を参照してください。

5.3 固定ベクタ領域

ユーザプログラム(motファイル)には固定ベクタ領域を含めないでください。

Note:

固定ベクタは、USBマストレージ経由内蔵FlashROM書き換えプログラム側の固定ベクタが使用されます。

5.4 ユーザプログラム(mot ファイル)のファイル名

USBマストレージ経由内蔵FlashROM書き換えプログラムがサポートするユーザプログラム(mot ファイル)のファイル名は、8.3 形式のみです。ファイル名が 8.3 形式以外の場合、USBマストレージ経由内蔵FlashROM書き換えプログラムはエラーとして RSK/RSSK 上の LED を点灯させ、ユーザプログラム(mot ファイル)の書き込み処理を行いません。

Note:

エラー時の LED の点灯パターンについては、「4.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの実行(motファイル書き込み)」を参照してください。

5.5 バックアップ機能使用時のセクション設定

ユーザプログラムはArea1領域内で実行されますので、ユーザプログラムのcode属性およびromdata属性のセクション設定はArea1の領域を指定し、ビルドを行ってください。

code属性 : 実行命令を格納します。
romdata属性 : ROMデータを格納します。

Note:

バックアップ機能については、「8.1 バックアップ機能」を参照してください。

5.6 ファイルフォーマット

USBマストレージ経由内蔵FlashROM書き換えプログラムがサポートしているファイル形式はモトローラ・タイプ S(S3フォーマット)のみです。

5.6.1 注意事項

1. レコード構成

Figure 5-2にUSBマストレージ経由内蔵FlashROM書き換えプログラムが対応するレコード構成を示します。Figure 5-2の順序以外で構成された S タイプフォーマットには対応していません。

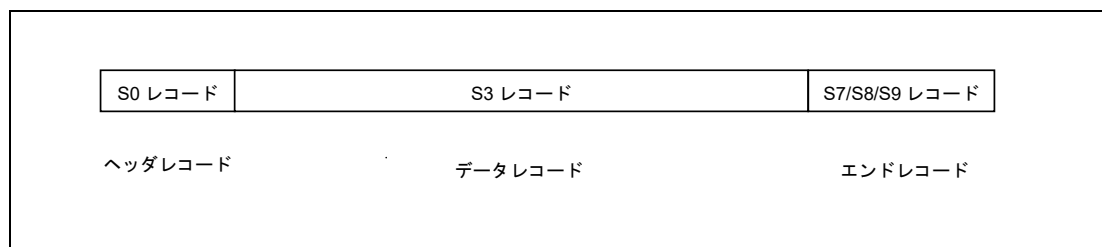


Figure 5-2 USBマストレージ経由内蔵FlashROM書き換えプログラムが対応するレコード構成

2. ロードアドレス

USBマストレージ経由内蔵FlashROM書き換えプログラムでは、ロードアドレスが昇順の S タイプフォーマットのみ対応しています。降順およびアドレスが前後する S タイプフォーマットファイルは使用しないでください。

3. エラー検出

USBマストレージ経由内蔵FlashROM書き換えプログラムでは、受信したSタイプフォーマットに異常があった場合、エラーを検出します。

(1). チェックサムエラー

受信したSタイプフォーマットのレコード毎にチェックサムの検査を行い、検査結果に異常があった場合、「チェックサムエラー」を検出します。

(2). フォーマットエラー

Sタイプフォーマットが以下の条件に当てはまった場合、「フォーマットエラー」を検出します。

- a. 未対応レコード(S1、S2、S4、S5、S6)を検出した場合。
- b. ヘッダレコード(S0)を2回検出した場合
- c. ヘッダレコード(S0)検出前に、データレコード(S3)もしくはエンドレコード(S7、S8、S9)を検出した場合

(3). アドレスエラー

ダウンロードエリア外への書き込みデータを受信した場合、「アドレスエラー」を検出します。

6. USBマストレージ経由内蔵FlashROM書き換えプログラムとユーザプログラムの設定

USBマストレージ経由内蔵FlashROM書き換えプログラムとユーザプログラムに対して必要な設定内容を以下に示します。

6.1 ユーザプログラムに対する設定

1. 設定内容 1

ユーザプログラムでは、Figure 6-1を参考にして UserApp Header 領域を作成してください。UserApp Header 領域については、「5.1 UserApp Header領域 (ユーザ・アプリケーション・ヘッダ)」を参照してください。

2. 設定内容 2

上記1で作成した UserApp Header 領域に対しセクションを設定し、そのセクションを必ずユーザプログラムの先頭に配置してください。

```
/******  
APPLICATION INTERFACE HEADER  
The purpose of the header is for an external application to be able to read  
certain values from known addresses.  
- Start address of UserApp.  
- Security code must match what PCDC Flashloader expects.  
- For revision purposes of applications etc.  
- Do not change the order of these variables!  
*****/  
#pragma section C UserApp_Head_Sect  
/* START ADDRESS of user application header data - Appheader address + 0x00. */  
const uint32_t userapp_entry_addr = (uint32_t) PowerON_Reset_PC;  
/* - Appheader address + 0x04. */  
const uint32_t userapp_sec_code = (uint32_t) USERAPP_SECURITY_CODE;
```

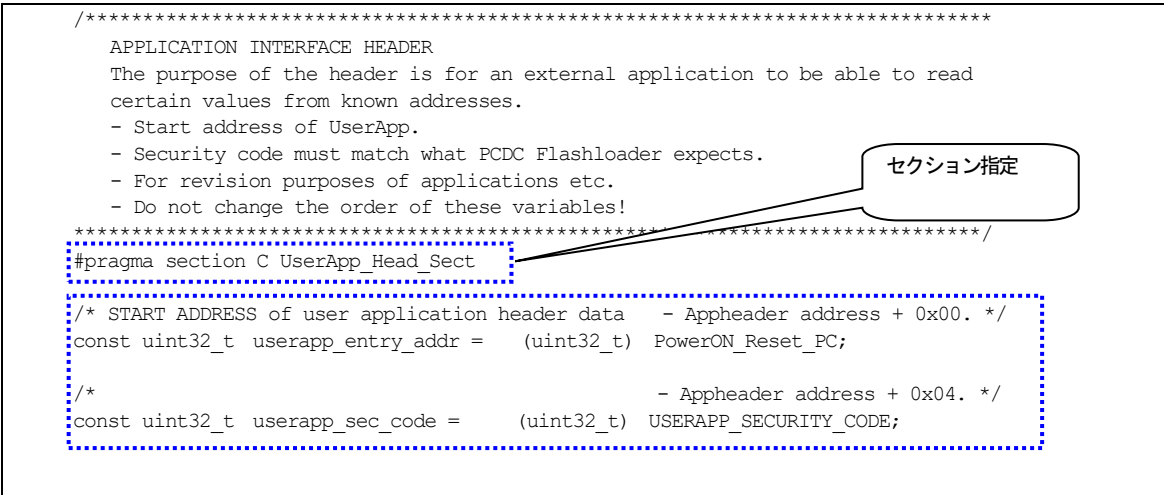


Figure 6-1 UserApp Header コード例

手順：

[プロパティ] → [C/C+ビルド] → [設定] を選択後、ツール設定タブを選び、 [Linker] → [セクション]を選択する。

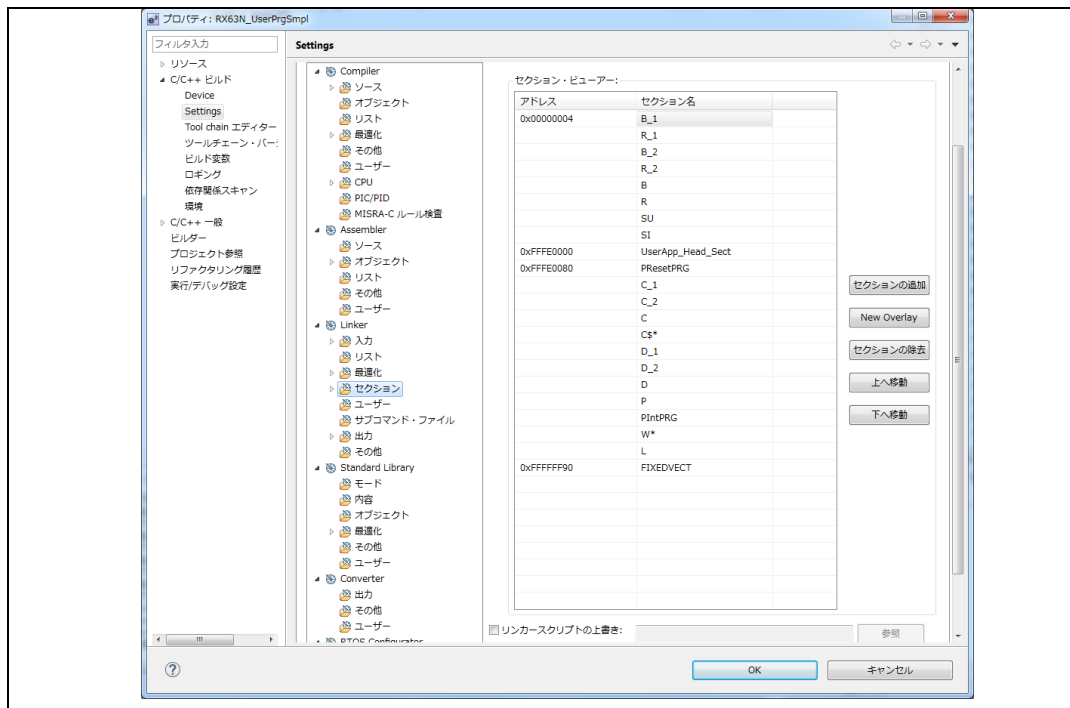


Figure 6-2 UserApp Header 領域の配置

6.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの設定

1. 設定内容 1

"r_config\r_usb_updater_config.h"ファイル内の以下の定義に対し、お客様のシステムに応じた設定を行ってください。

(1). ユーザプログラム(mot ファイル)名指定

USB_CFG_USER_PROGRAMNAME 定義に対し FlashROM 領域に書き込みが行われるユーザプログラム(mot ファイル)名を指定してください。

```
#define USB_CFG_USER_PROGRAMNAME "sample.mot" // ユーザプログラム名
```

Note:

- ユーザプログラム(mot ファイル)名は、ダブルクォーテーションで括ってください。
- ユーザプログラム(mot ファイル)のファイル名は、8.3 形式のみをサポートしています。

(2). USB モジュール指定

USB_CFG_USE_USBIP 定義に対し、使用する USB モジュール番号を指定してください。USB0 モジュールを使用する場合は USB_CFG_USE_USBIP 定義に対し、USB_CFG_IP0 を設定し、USB1 モジュールを使用する場合は、USB_CFG_IP1 を設定してください。

```
#define USB_CFG_USE_USBIP USB_CFG_IP0 // USB0モジュール使用設定  
#define USB_CFG_USE_USBIP USB_CFG_IP1 // USB1モジュール使用設定
```

Note:

ご使用のMCUがUSBモジュールを1つしかサポートしていない場合には、USB_CFG_USE_USBIP 定義に対し、USB_CFG_IP0 を設定してください。

(3). バックアップ機能設定

USB_CFG_BACKUP 定義に対し、バックアップ機能を使用するかどうかを指定してください。バックアップ機能を使用する場合は、USB_CFG_ON を指定し、バックアップ機能を使用しない場合は、USB_CFG_OFF を指定してください。

```
#define USB_CFG_BACKUP USB_CFG_ON // バックアップ機能使用設定  
#define USB_CFG_BACKUP USB_CFG_OFF // バックアップ機能非使用設定
```

Note:

バックアップ機能については、「8.1 バックアップ機能」を参照してください。

(4). PLL クロック周波数設定

USBAa/USBA モジュールをご使用の場合、PHYSET レジスタの PLL クロックソース周波数設定ビット(CLKSEL) に対し、20MHz または 24MHz のいずれかを選択してください。

```
#define USB_CFG_CLKSEL USB_CFG_24MHZ // 24MHz 設定  
#define USB_CFG_CLKSEL USB_CFG_20MHZ // 20MHz 設定
```

Note:

- USBAa/USBA モジュール以外の USB モジュールの場合、この定義は無視されます。
- USBAa/USBA モジュールは、RX71M/RX64M で使用されている USB モジュールです。

(5). CPU バスウェイト設定

USBAa/USBA モジュール内にある BUSWAIT レジスタに設定する数値を USB_CFG_BUSWAIT に対し指定してください。

```
#define USB_CFG_BUSWAIT 7 // 7 ウェイト設定
```

Note:

- a. USB_CFG_BUSWAIT に指定する数値については、RX71M/RX64M のハードウェアマニュアル内の BUSWAIT レジスタの章を参照してください。
- b. USBAa/USBA モジュール以外の USB モジュールの場合、この定義は無視されます。
- c. USBAa/USBA モジュールは、RX71M/RX64M で使用されている USB モジュールです。

(6). USB レギュレータ設定

RX231 がサポートしている USB レギュレータ機能を使用するか、または使用しないかの設定を下記の定義に対し行ってください。

```
#define USB_CFG_REGULATOR USB_CFG_OFF // 非使用
#define USB_CFG_REGULATOR USB_CFG_ON // 使用
```

Note:

RX231 以外の MCU をご使用の場合、この定義は無視されます。

(7). その他の指定

USBマストレージ経由内蔵FlashROM書き換えプログラムは、ユーザプログラム内の UserApp Header 領域の内容を参照しますので、UserApp Header 領域の配置アドレスを変更した場合は、変更後の UseApp Header 領域を参照するよう本プログラムを変更する必要があります。また、セキュリティコードの値を変更した場合も、本プログラムの変更が必要になります。UserApp Header 領域の詳細については、「5.1 UserApp Header領域 (ユーザ・アプリケーション・ヘッダ)」を参照してください。

a. UserApp Header 領域の配置アドレス設定

USB_CFG_USERAPP_HEADER_ADDR 定義に対し、UserApp Header 領域の配置アドレスを設定してください。

```
#define USB_CFG_USERAPP_HEADER_ADDR UserApp Header領域の配置アドレス
```

b. セキュリティコード設定

USB_CFG_USERAPP_SECURITY_CODE 定義に対し、UserApp Header 領域に設定したセキュリティコード(4 バイトデータ)を設定してください。

```
#define USB_CFG_USERAPP_SECURITY_CODE セキュリティコード
```

Note:

セキュリティコードには、0xFFFFFFFF 以外の値を設定してください。

2. 設定内容 2

デュアルモードをサポートする MCU をご使用の場合、"r_config¥r_bsp_config.h"ファイル内の BSP_CFG_CODE_FLASH_BANK_MODE 定義に対し、0 (デュアルモード)または 1 (リニアモード)のいずれかを指定してください。

```
#define BSP_CFG_CODE_FLASH_BANK_MODE 0 // デュアルモード
#define BSP_CFG_CODE_FLASH_BANK_MODE 1 // リニアモード
```

3. 設定内容 3

- (1). USBマストレージ経由内蔵FlashROM書き換えプログラムは、RSK/RSSK 上のスイッチ(RSK:スイッチ 3、RSSK:スイッチ 2)の状態により、ROM 書き換え処理にジャンプするか、ユーザプログラムにジャンプするかの

判定を行っています。この判定処理は、ボードの仕様に依存しますので、ご使用のボードに合わせた判定処理に変更いただきますようお願いします。判定処理は、main 関数内で行っています。

```
void main(void)
{
    if (SW3 != SW_ACTIVE)
    {
        if (USER_PROG_WRITE_OK == fu_user_prog_start())
        {
            usb_cpu_int_disable();
            jump_to_userapp();
        }
    }
    :
}
```

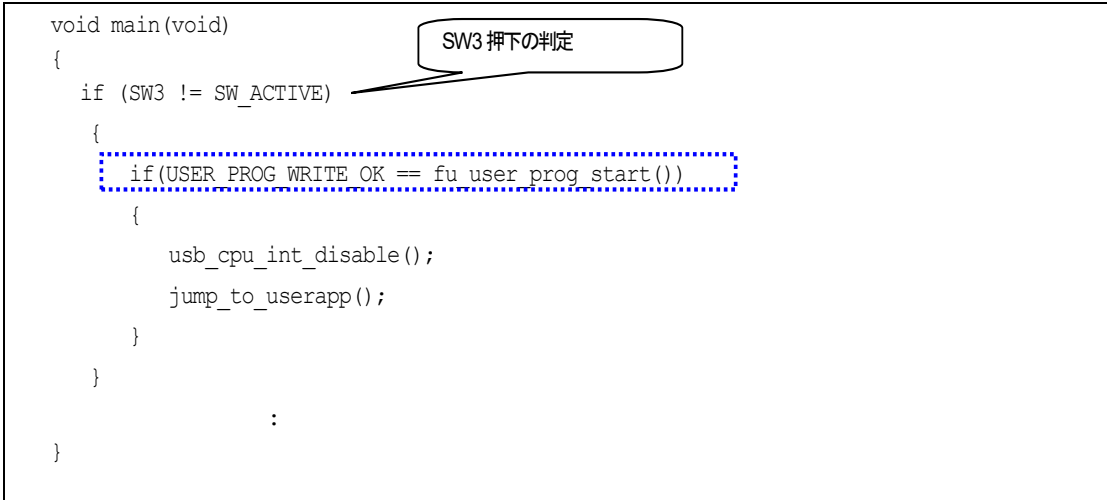


Figure 6-3 main()関数処理

- (2). USBマストレージ経由内蔵FlashROM書き換えプログラムは、RSK/RSSK 上の LED を使用しています。USBマストレージ経由内蔵FlashROM書き換えプログラムをお客様のシステムに組み込む場合、必要に応じて以下の関数および関数呼び出し処理を削除してください。

関数名 : USB_RSK_LED_SET

4. 設定内容 4 (オプション設定メモリ)

オプション設定メモリの設定は、下記の項目に対してのみ行うことが可能です。下記以外はすべてデフォルト値を設定してください。

- (1). FASTSTUP ビット
- (2). LVDAS / STUPLVD1REN ビット
- (3). VDSEL / STUPLVD1LVL ビット
- (4). MDE ビット

なお、USBマストレージ経由内蔵FlashROM書き換えプログラムは、ユーザプログラム内にあるオプション設定メモリの ROM 書き込みを行いません。本プログラムのオプション設定メモリが、ユーザプログラムでも使用されますので、オプション設定メモリに対する設定は、本プログラムに対して行ってください。

Note:

- a. USBマストレージ経由内蔵FlashROM書き換えプログラムのオプション設定メモリの初期値は、すべてデフォルト値となっています。
- b. RX62N は、オプション設定メモリをサポートしていません。
- c. オプション設定メモリの詳細については、MCU のユーザズマニュアル ハードウェア編を参照してください。

5. 設定内容 5(コンパイルオプション)

e² studio 上の設定オプションである[関数の分岐幅]に対し、以下を設定してください。

- (1). USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域外の ROM 領域に配置する場合

e² studio 上の[分岐幅]に対し[24 ビット以内]を選択してください。

- (2). USBマストレージ経由内蔵FlashROM書き換えプログラムをユーザブート領域に配置する場合
e² studio 上の[分岐幅]に対し[設定しない]を選択してください。

Note:

[分岐幅]は、[ファイル]→[プロパティ]→[C/C+ビルド]→[設定] を選択後、[Common]→[CPU] で指定できます。

6.3 USBマストレージ経由内蔵FlashROM書き換えプログラムの USB 端子設定

お客様のシステムに応じた USB 端子の設定を行ってください。USB 端子設定は以下の関数で行われています。

関数名	ファイル名
usb_pin_setting	demo_src¥main.c

6.4 ユーザプログラムの配置

ユーザプログラムは、USBマストレージ経由内蔵FlashROM書き換えプログラムが書き込まれたROM領域と重ならない領域に配置してください。

Note:

1. ユーザプログラムが以下のROM領域に配置されるように配置設定を行ってください。なお、デュアルモード使用時は、起動バンク領域にユーザプログラムが配置されるよう設定を行ってください。

バックアップ機能	ユーザプログラム配置可能領域		
OFF	プログラム ROM 領域先頭アドレス	-	0xFFFF7FFB
ON	プログラム実行領域先頭アドレス	-	0xFFFF7FFB

Note:

バックアップ機能およびプログラム実行領域については、「8.1 バックアップ機能」を参照してください。

2. ユーザプログラムがFlashROMへ書き込まれる前に、USBマストレージ経由内蔵FlashROM書き換えプログラム領域を除く全FlashROM領域が消去されます。
3. Flash Self programmingライブラリはRAM領域の一部を使用しますが、USBマストレージ経由内蔵FlashROM書き換えプログラム実行時のみ使用するため、ユーザプログラムの動作には影響がありません。
4. 0xFFFF7FFC – 0xFFFF7FFFの領域(4バイト)は、USBマストレージ経由内蔵FlashROM書き換えプログラムが管理領域として使用します。

7. USBメモリへのユーザプログラムの書き込み

FAT32でフォーマットしたUSBメモリのルートフォルダにユーザプログラム(motファイル)をコピーしてください。

Note:

1. USBマストレージ経由内蔵FlashROM書き換えプログラムは、"r_config\r_usb_updater_config.h"ファイル内のUSB_CFG_USER_PROGRAMNAME定義に設定したユーザプログラムのFlashROM書き込みを行います。
2. USB_CFG_USER_PROGRAMNAME定義に設定したユーザプログラムがUSBメモリのルートフォルダに存在しない場合は、FlashROM書き込みは行われません。
3. USB_CFG_USER_PROGRAMNAME定義については、「6.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの設定」を参照してください。
4. 必ずFAT32でフォーマットしたUSBメモリを使用してください。
5. ユーザプログラム(motファイル)のファイル名は、8.3形式のみをサポートしています。

8. USBマストレージ経由内蔵FlashROM書き換えプログラムについて

8.1 バックアップ機能

USBマストレージ経由内蔵FlashROM書き換えプログラムは、FlashROM書き換え中、USB通信失敗等によりFlashROM書き換えに失敗しても、すでに特定領域に保持されているユーザプログラムが起動するバックアップ機能をサポートしています。

バックアップ機能のFlashROM書き換え処理概要は以下の通りです。

1. USBマストレージ経由内蔵FlashROM書き換えプログラムは、内蔵FlashROM(プログラムROM領域)を2つの領域に分け、1つの領域をプログラム実行領域(Area1)、もう1つの領域をユーザプログラム格納領域(Area2)として使用します。Area1とArea2は内蔵FlashROM領域の中心から2つに分けた領域です。これらの2つのROM領域のサイズは同一です。なお、Area2の領域には未使用領域が存在します。これは、Area1領域内のUSBマストレージ経由内蔵FlashROM書き換えプログラムがArea2の領域には存在しないためです。



Figure 8-1 バックアップ機能使用時のFlashROM領域

2. バックアップ機能を有効にした場合、USBマストレージ経由内蔵FlashROM書き換えプログラムは、必ずArea2の領域にユーザプログラム(mot1)の書き込みを行います。

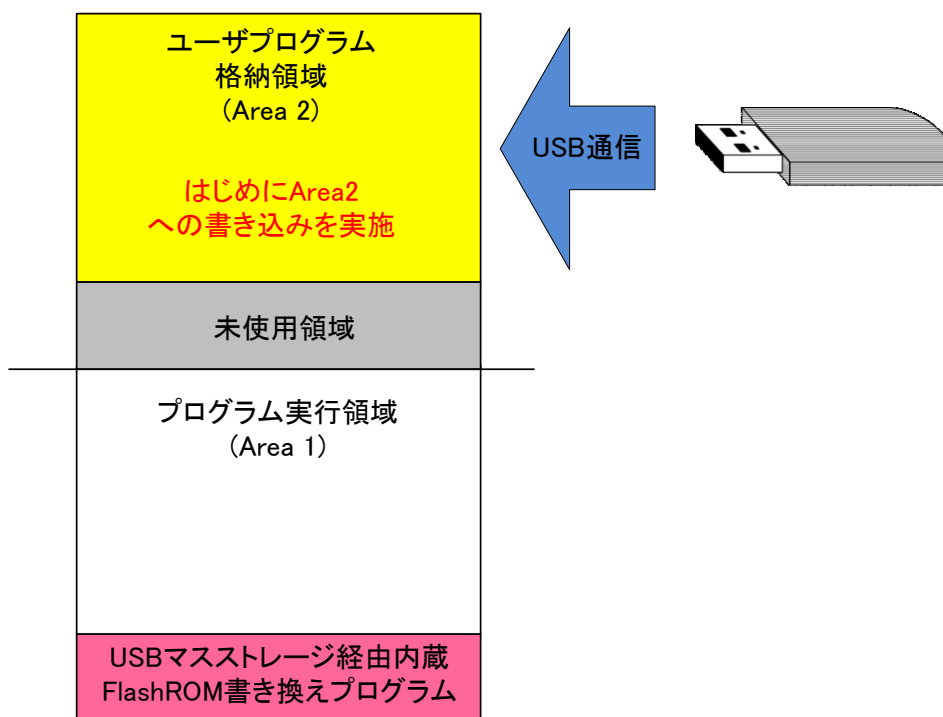


Figure 8-2 バックアップ機能使用時のFlash ROM 領域

- 書き込み正常終了後、USBマストレージ経由内蔵FlashROM書き換えプログラムがArea2からArea1へコピー処理を行います。Area1へのコピー完了後、Area1内にあるユーザプログラムが実行されます。

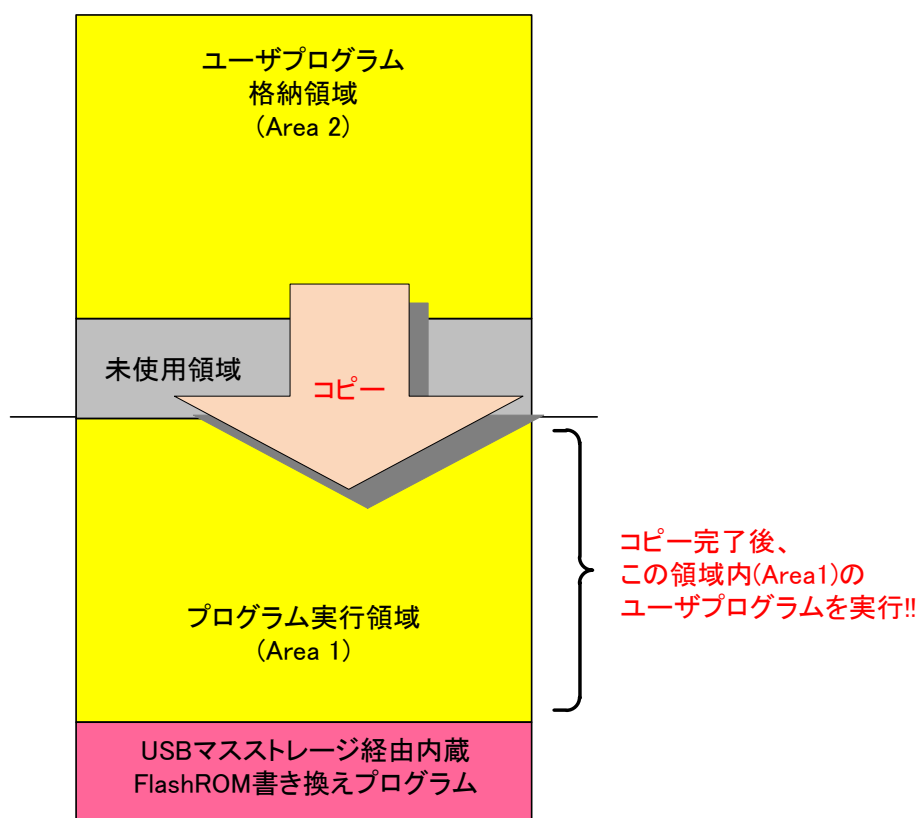


Figure 8-3 ユーザプログラムのコピー処理

4. USBマストレージ経由内蔵FlashROM書き換えプログラムによりユーザプログラムを更新する場合、Area2の領域を消去した後、Area2へのFlashROM書き込みが行われ、書き込み完了後、Area1へのコピー処理が行われます。

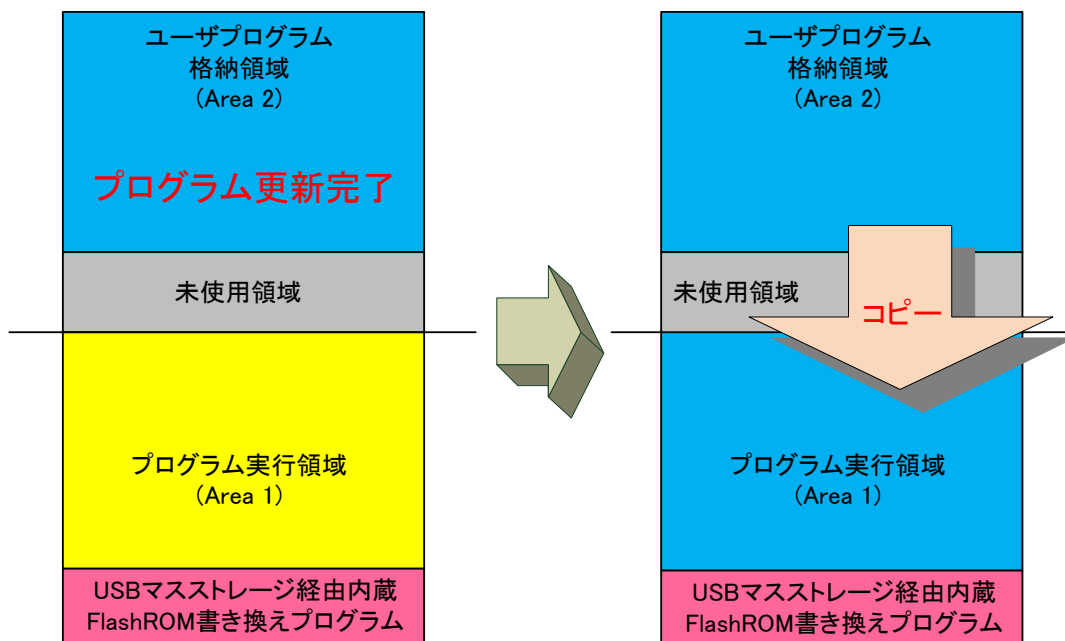


Figure 8-4 ユーザプログラムの更新

Note:

Area2への書き込みが正常完了した後、なんらかの理由によりArea1を消去できなかった場合、Area1内のユーザプログラムは更新されず、以前、Area1に書き込んだユーザプログラムが再度起動します。Area1を消去できない現象が発生した場合は、Area2への書き込み処理を再度実行してください(上記2参照)。

5. Area2への書き込み中、USB通信失敗等によりFlashROM書き込みが失敗しても、Area1領域には、上記4で書き込まれたユーザプログラムは保持されたままです。FlashROM書き込み失敗前のユーザプログラムが起動されます。



Figure 8-5 ユーザプログラムの更新失敗

Note:

1. Area2 から Area1 へのコピー中、何らかの理由によりそのコピーに失敗した場合、RSK/RSSK のリセット/電源投入を行ってください。再度USBマストレージ経由内蔵FlashROM書き換えプログラムが Area2 から Area1 へのコピー処理を行います。コピー処理が正常に終了するとユーザプログラムが起動されます。このコピー処理には RSK/RSSK のリセット/電源投入から最大 10 秒程度の時間を要します。

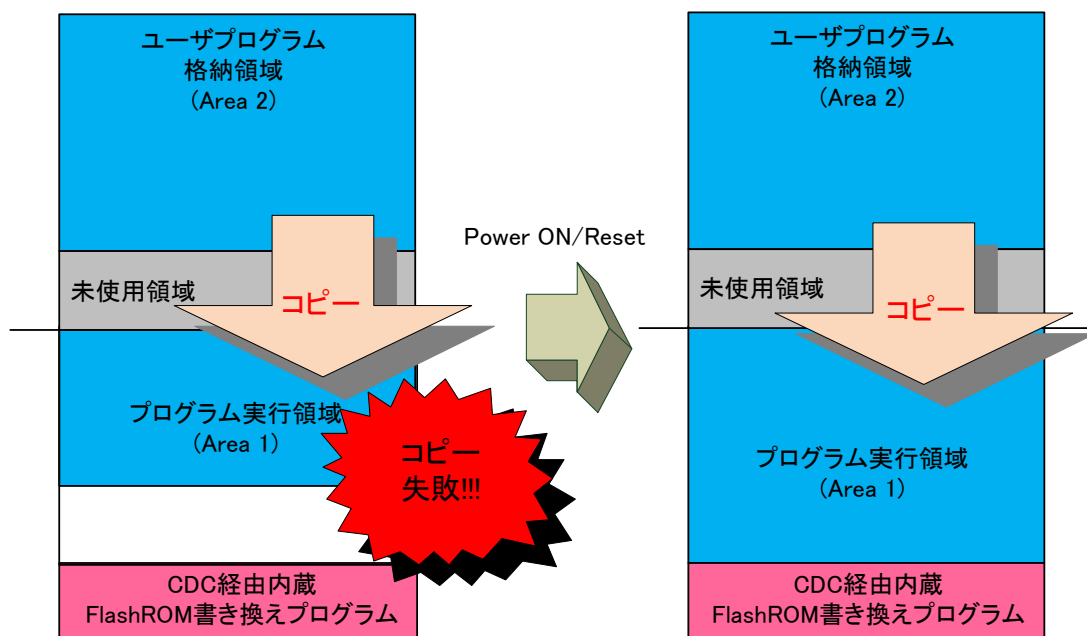


Figure 8-6 Area2 から Area1 へのコピー失敗

2. ユーザプログラムは Area1 領域内で実行されますので、ユーザプログラムの code 属性および romdata 属性のセクション設定は Area1 の領域を指定し、ビルドを行ってください。

code属性 : 実行命令を格納します。
romdata属性 : 固定データを格納します。

3. バックアップ機能のサポート/非サポートは `r_usb_fwupdater_config.h` 内のマクロ定義に対する設定により行います。この設定の詳細については、「6.2 USBマストレージ経由内蔵FlashROM書き換えプログラムの設定」を参照してください。
4. デュアルモードをサポートするMCUをご使用の場合、デュアルモードをご使用ください。

8.2 リセット直後の動作

マイコンをリセット後、メイン関数（C言語記述：`main()`）が実行される前に実行する処理を指します。

RXマイコンでは、リセット後の初期化処理として、以下が行われます。

1. スタック領域の確保とスタック・ポインタの設定
2. `main` 関数の引数領域の確保
3. `data` 領域、スタック領域の初期化
4. `hdwinit` 関数でのユーザプログラムへの分岐及びMCU周辺デバイスの初期化
5. `main` 関数への分岐

リセット後、USBマストレージ経由内蔵FlashROM書き換えプログラムからユーザプログラムへジャンプして行くため、必ずUSBマストレージ経由内蔵FlashROM書き換えプログラムが処理され、上記に記載したマイコンの初期化処理等が行われます。

8.2.1 電源投入時/リセット時の動作フロー

次に、USBマストレージ経由内蔵FlashROM書き換えプログラムの電源投入時/リセット時の動作フローについて説明します。

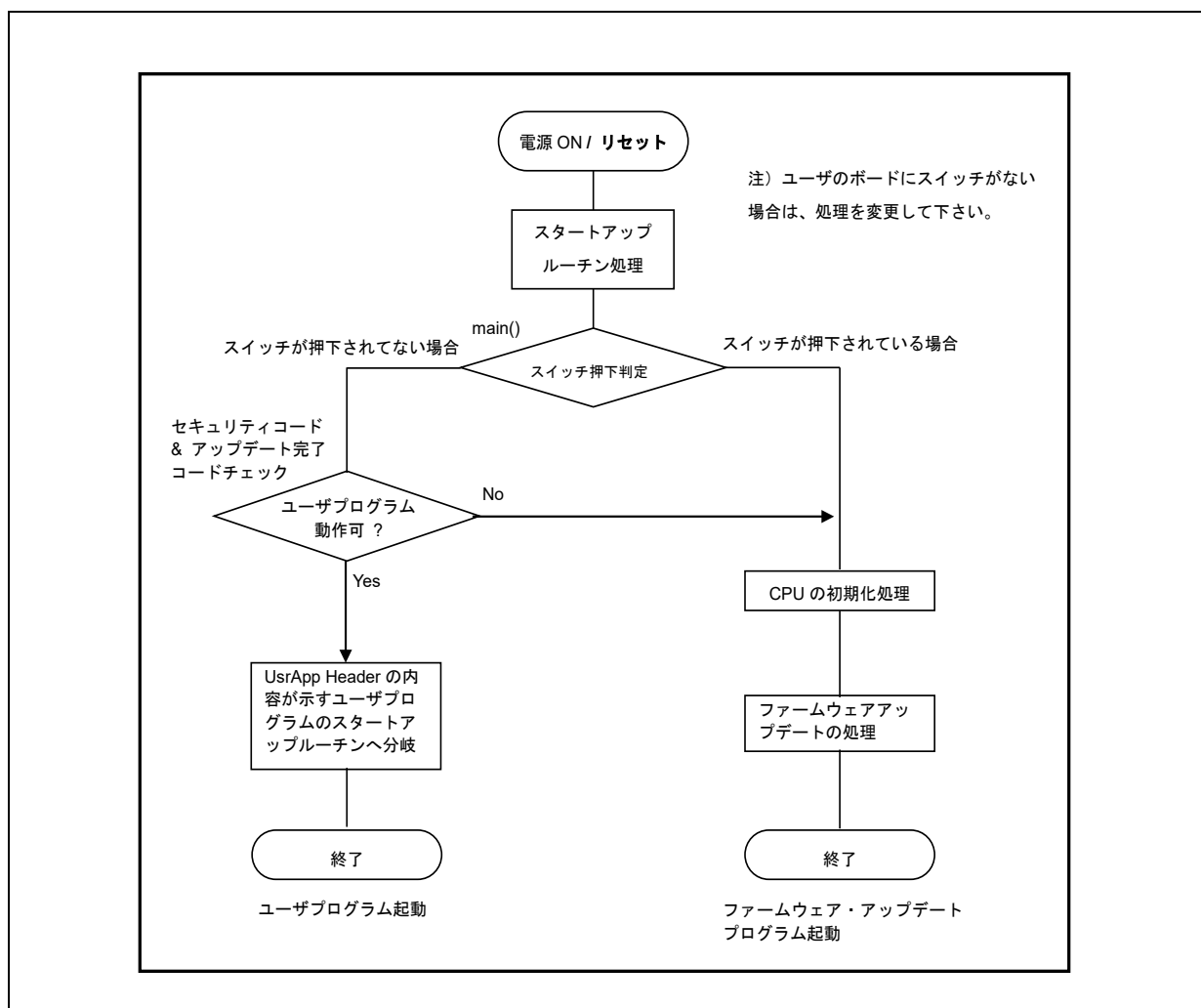


Figure 8-7 電源投入時/リセット時の動作フロー

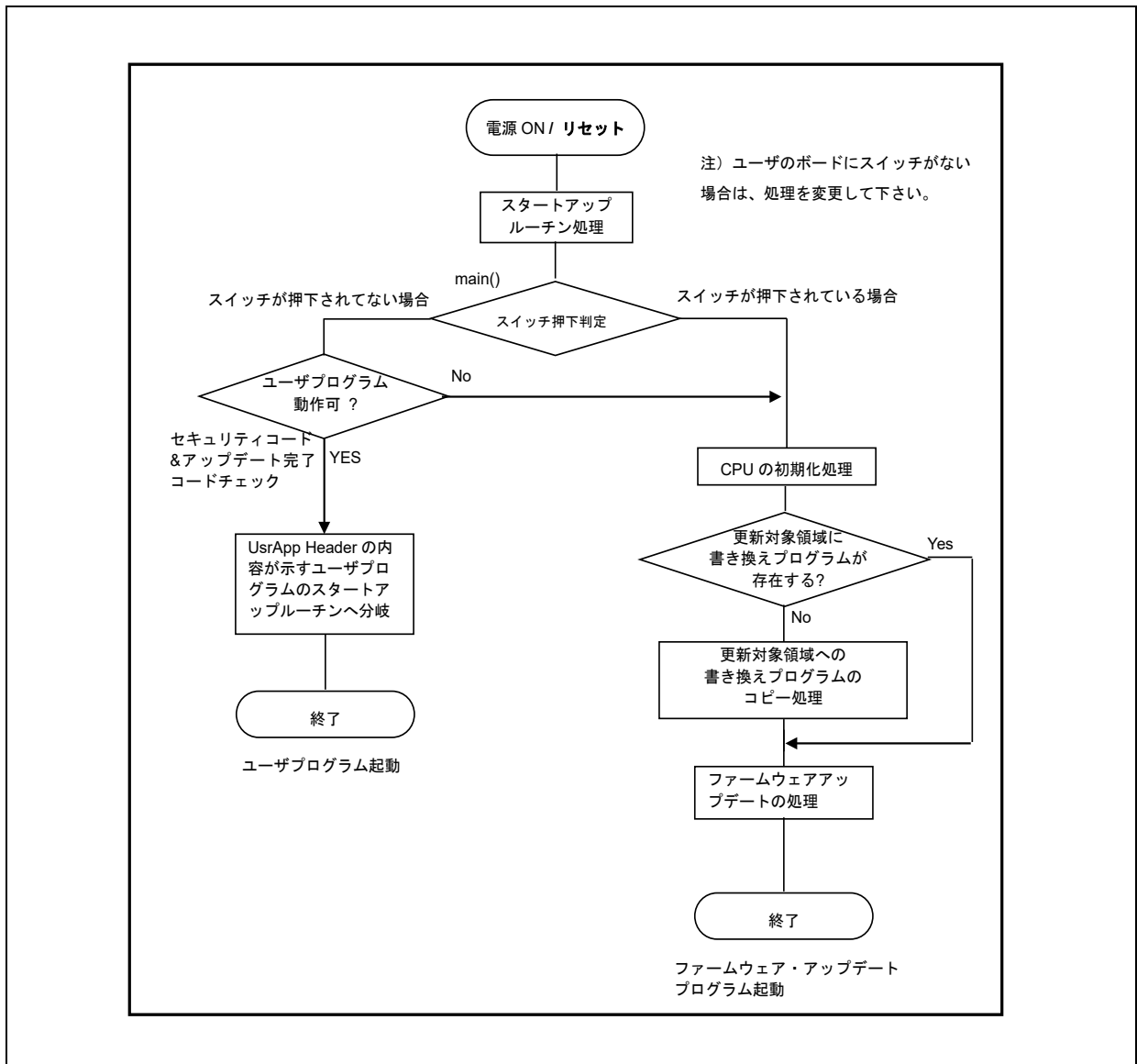


Figure 8-8 電源投入時/リセット時の動作フロー (デュアルモード使用時)

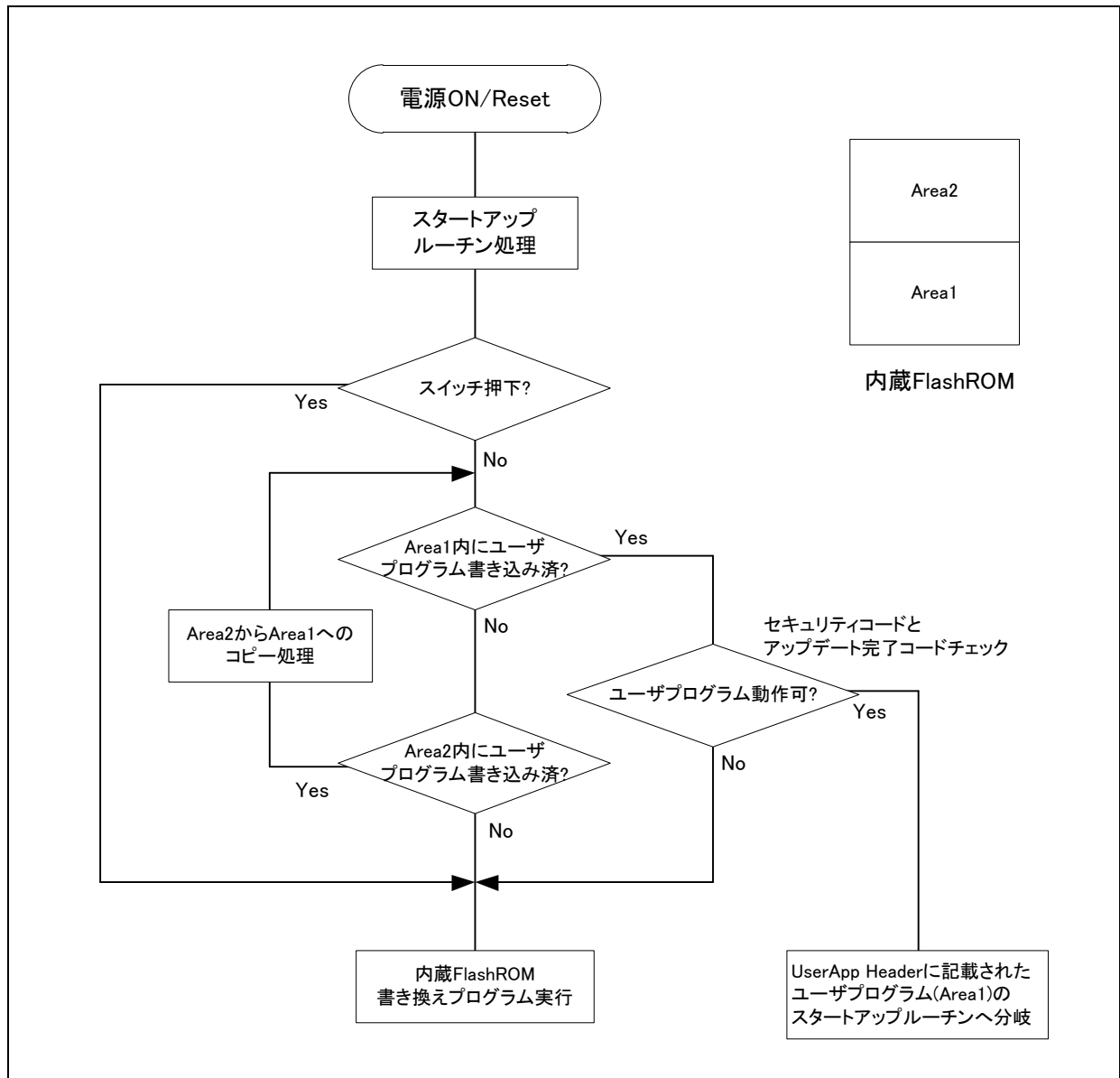


Figure 8-9 電源投入時/リセット時の動作フロー (バックアップ機能使用時)

セキュリティコードおよびユーザプログラムへの分岐アドレス情報に関しては、「5.1 UserApp Header領域 (ユーザ・アプリケーション・ヘッダ)」を参照して下さい。

なお、UserApp Header領域にセキュリティコードが正常に設定されていても、ユーザプログラムのスタートアドレスが正しくない場合、ユーザプログラムは動作しません。

8.2.2 ユーザプログラムの起動条件

以下の条件をすべて満たした場合、UserApp Header領域に設定されたユーザプログラムが起動します。

1. 正しいセキュリティコードが設定されている。
2. 正しいユーザプログラムの先頭アドレスが設定されている。
3. アップデート完了コードが正常に書かれている。

正常に書き込みが完了した時に、当該書き換えプログラムが自動的にアップデート完了コードの書き込みを行います。

なお、セキュリティコードおよびアップデート完了コードが不一致(正しくない)の場合、またはアップデート完了コードが書かれていない場合、USBマストレージ経由内蔵FlashROM書き換えプログラムのみが起動し、ユーザプログラムは起動しません。

8.2.3 USBマストレージ経由内蔵FlashROM書き換えプログラムの起動条件

1. ユーザプログラムが ROM に書かれている場合

RSK/RSSK 上のスイッチ(RSK:スイッチ 3, RSSK:スイッチ 2)を押下した状態でリセット起動するとUSBマストレージ経由内蔵FlashROM書き換えプログラムが起動します。

2. ユーザプログラムが ROM に書かれていない場合

USBマストレージ経由内蔵FlashROM書き換えプログラムが起動します。

8.2.4 USBマストレージ経由内蔵FlashROM書き換えプログラムへの分岐

USBマストレージ経由内蔵FlashROM書き換えプログラムのmain()関数内でユーザプログラムにジャンプするかUSBマストレージ経由内蔵FlashROM書き換えプログラムを継続するかの分岐判定を行います。

条件分岐を経て、CPU内蔵機能・周辺回路の初期化を行ったあと、USBマストレージ経由内蔵FlashROM書き換えプログラムを実行します。

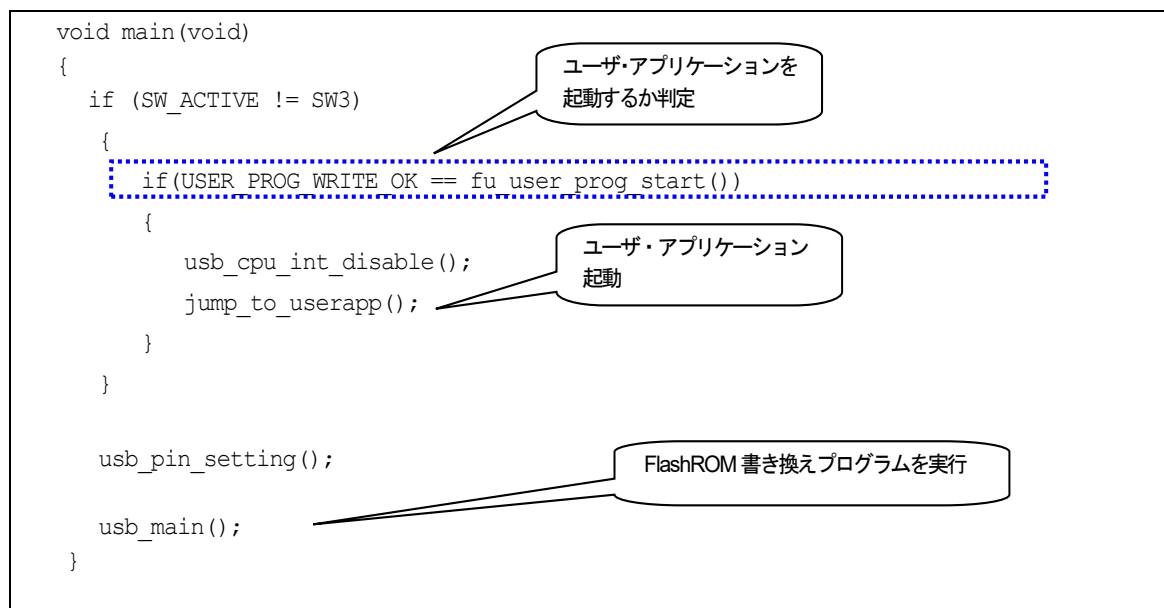


Figure 8-10 main()関数処理

8.2.5 ユーザ・アプリケーションへの分岐

ユーザプログラムへのジャンプ処理は、jump_to_userapp() 関数によって行われます。なお、ジャンプ先であるユーザプログラムの先頭アドレス指定については、「5.1 UserApp Header領域 (ユーザ・アプリケーション・ヘッダ)」を参照してください。

8.3 フォルダ/ファイル構成

USBマストレージ経由内蔵FlashROM書き換えプログラムのソースファイルとフォルダ構成を示します。

(MCU 名)	
+HardwareDebug	ビルド結果
+src	
+----r_bsp	[Renesas Board Support Package]
+----r_config	[Configuration File]
+----r_flash_rx	[Flash Module]
+----r_tfat_driver_rx	[FAT Driver]
+----r_tfat	[FAT]
+----r_usb_basic_mini	[USB Basic Driver]
+----r_usb_hmsc_mini	[USB Host Mass Storage Class Driver]
+----demo_src	[Application Program]

Figure 8-11 USBマストレージ経由内蔵FlashROM書き換えプログラムのフォルダ構成

8.3.1 r_config フォルダ

システム設定用ヘッダファイルが格納されているフォルダです。

8.3.2 r_flash_rx フォルダ

Flash Module関連のソースファイルとヘッダファイルが格納されているフォルダです。詳細は、Flash Module Using Firmware Integration Technology のアプリケーションノートを参照してください。

ボードサポートパッケージ(r_bsp)でMCUを選択すると、フラッシュ書き込みタイプが自動的に選択されます。

8.3.3 r_bsp フォルダ

BSP関連のソースファイルとヘッダファイルが格納されているフォルダです。詳細はRX Family Board Support Package Moduleのアプリケーションノート(Document番号:R01AN1685)を参照してください。

8.3.4 r_tfat_rx フォルダ

FAT関連のソースファイルとヘッダファイルが格納されているフォルダです。詳細はRX Family Open Souce FAT File System [M3S-TFAT-Tiny] Moduleのアプリケーションノート(Document番号:R20AN0038)を参照してください。

8.3.5 r_tfat_driver_rx フォルダ

FATドライバ関連のソースファイルとヘッダファイルが格納されているフォルダです。詳細はRX Family M3S-TFAT-Tiny Memory Driver Interface Moduleのアプリケーションノート(Document番号:R20AN0335)を参照してください。

8.3.6 r_usb_basic_mini, r_usb_hmsc_mini フォルダ

USB ドライバ関連のソースファイルとヘッダファイルが格納されているフォルダです。詳細は Renesas USB MCU USB Basic Mini Host and Periperal Driver のアプリケーションノート(Document 番号:R01AN2166)や Renesas USB MCU USB Host Mass Storage Class Driver(Document 番号:R01AN2169)を参照してください。

8.3.7 demo_src フォルダ

USBマストレージ経由内蔵FlashROM書き換えプログラムのアプリケーションプログラムが格納されているフォルダです。

Table 8-1 USBマストレージ経由内蔵FlashROM書き換えプログラムのソースファイル

ファイル名	説明
main.c	C言語メイン関数記述ファイル
r_flash_apl.c	ROM書き換えプログラム処理ファイル
r_flash_apl.h	ROM書き換えプログラム処理用ヘッダファイル
r_usb_read_buffer.c	受信バッファ関連処理
r_usb_read_buffer.h	受信バッファ関連処理用ヘッダファイル
r_fwupdater_apl.c	フラッシュ書き換えデータ処理
r_fwupdater_apl.h	フラッシュ書き換えデータ処理用ヘッダファイル
その他	USBドライバ用アプリケーションプログラムファイル RSK/RSSK用周辺制御ファイル

8.3.8 HardwareDebug フォルダ

ビルド時にUSBマストレージ経由内蔵FlashROM書き換えプログラムの実行可能なオブジェクト・ファイルとmotファイルが格納されるフォルダです。

8.4 注意事項

USBマストレージ経由内蔵FlashROM書き換えプログラムは、RSK/RSSK上のスイッチ(RSK:スイッチ3, RSSK:スイッチ2)の状態により、USBマストレージ経由内蔵FlashROM書き換えプログラムにジャンプするか、またはユーザプログラムにジャンプするかの判定を行っています。この判定処理は、ボードの仕様に依存しますので、ご使用のボードに合わせた判定処理に変更いただきますようお願いいたします。判定処理を行っている部分は、USBマストレージ経由内蔵FlashROM書き換えプログラムのmain関数です。

9. e² studio 用プロジェクトを CS+で使用する場合

USBマストレージ経由内蔵FlashROM書き換えプログラムのプロジェクトは、統合環境 e² studio で作成されています。USBマストレージ経由内蔵FlashROM書き換えプログラムを CS+で動作させる場合は、下記の手順でインポート処理を行ってください。

Note:

1. srcフォルダとrcpcファイルを格納するフォルダの名前は、"MCU名_FirmwareUpdater"にしてください。
例えば、RX63Nの場合、フォルダ名は"RX63N_FirmwareUpdater"になります。
2. 「プロジェクト変換設定」ウィンドウ内の「変換直前のプロジェクト構成ファイルをまとめてバックアップする」のチェックを外してください。

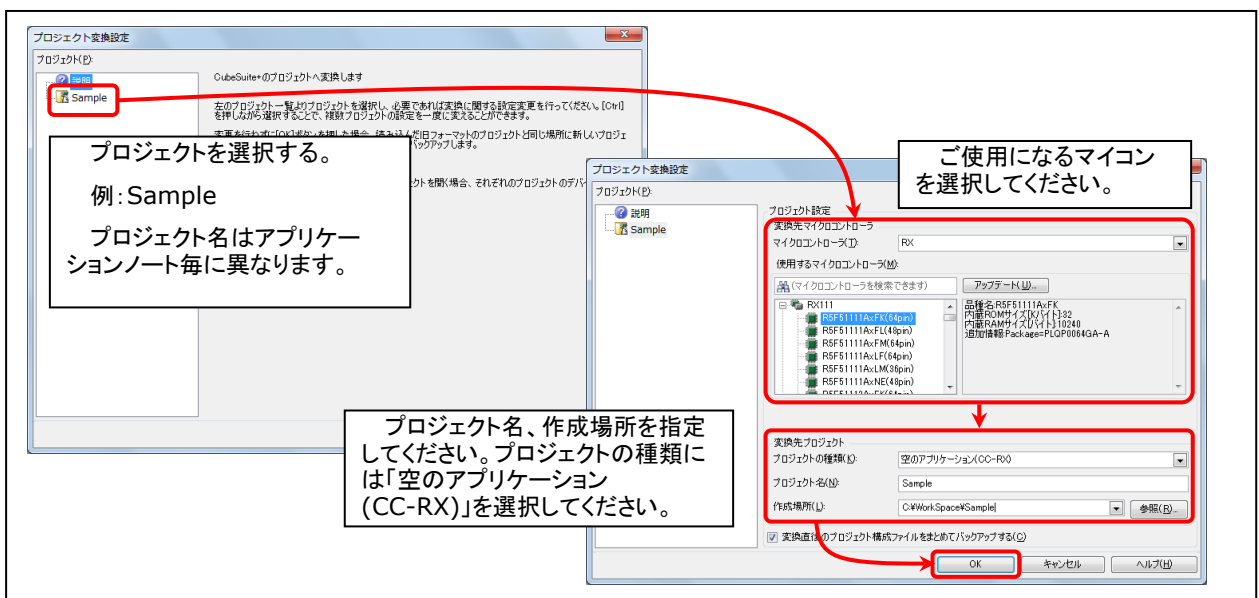
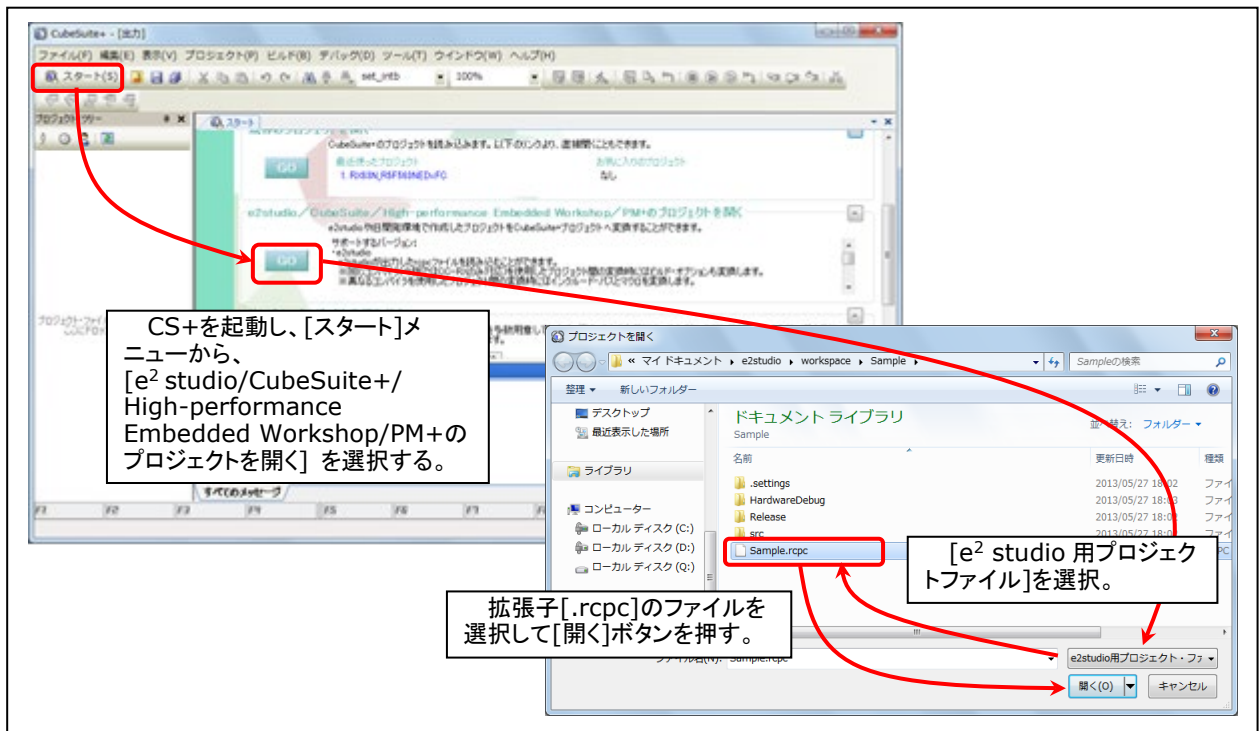


Figure 9-1 e² studio 用プロジェクトの CS+読み込み方法

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	Mar 31, 2017	—	初版発行
1.01	Jan 31, 2018	—	1. RX65N/RX651-2M をサポート 2. デュアルモードをサポート
1.02	Apr 16, 2018	—	バックアップ機能をサポート
1.03	Apr 16, 2019	—	対象デバイスに RX66T/RX72T を追加
1.04	Mar 1, 2020	—	対象デバイスに RX72M/RX72N/RX66N/RX23W を追加
1.05	Mar 1, 2021	—	対象デバイスに RX671 を追加

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しないでください。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っていると、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、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、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

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。

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/