

## AP-SH2A-1A (SH7263 CPU BOARD)

## サンプルプログラム解説

6版 2021年02月05日

## 目次

1. 概要 .....	1
1. 1 概要 .....	1
1. 2 動作モード .....	1
1. 3 開発環境について .....	3
1. 4 ワークスペースについて .....	3
2. サンプルプログラムの構成 .....	4
2. 1 フォルダ構成 .....	4
2. 2 ファイル構成 .....	5
2. 2. 1 USB ホスト用サンプルプログラムのファイル構成 .....	5
2. 2. 2 USB ファンクション用サンプルプログラムのファイル構成 .....	6
3. ビルド・デバッグ方法 .....	7
3. 1 USB ホスト用サンプルプログラム .....	7
3. 2 USB ファンクション用サンプルプログラム .....	9
4. 動作説明 .....	11
4. 1 USB ホスト用サンプルプログラムの動作説明 .....	11
4. 1. 1 USB ホスト用サンプルプログラム概要 .....	11
4. 1. 2 USB ホスト動作 .....	13
4. 1. 3 USB ホスト用サンプルプログラム API 一覧 .....	14
4. 1. 4 USB サンプルプログラム注意事項 .....	14
4. 2 USB ファンクション用サンプルプログラムの動作説明 .....	15
4. 2. 1 USB ファンクション用サンプルプログラム概要 .....	15
4. 2. 2 USB ファンクション動作 .....	17
4. 2. 3 USB サンプルプログラム注意事項 .....	17
4. 3 RAM 動作時のメモリマップ .....	18
4. 4 ROM 動作時のメモリマップ .....	19

## 1. 概要

### 1. 1 概要

本アプリケーションノートでは、AP-SH2A-1A に付属するサンプルプログラムについて解説します。  
AP-SH2A-1A には、下記のサンプルプログラムが付属しています。

サンプルプログラム	動作内容
USB ホスト用サンプルプログラム	<ul style="list-style-type: none"><li>• USB ホスト デバイス情報表示</li><li>• シリアル通信</li><li>• タイマ割り込み</li><li>• CAN 通信</li></ul>
USB ファンクション用サンプルプログラム	<ul style="list-style-type: none"><li>• USB ファンクション 仮想シリアル通信</li><li>• シリアル通信</li><li>• タイマ割り込み</li><li>• CAN 通信</li></ul>

詳細な動作内容に関しては、後述の「4. 動作説明」を参照してください。

### 1. 2 動作モード

本サンプルプログラムは、AP-SH2A-1A で動作します。CPU 動作モード、各メモリ設定は下記のようになっています。  
モードの設定方法等につきましては、「AP-SH2A-1A ハードウェアマニュアル」をご覧ください。  
なお、下記以外の条件で動作させる場合には、ソースファイルやコンパイラオプションなどを変更する必要があります。

クロックモード	: MODE0
エリア 0 空間バス幅	: 16bit
FlashROM 設定	: FlashROM を使用する
FlashROM ライトプロテクト設定	: ライトプロテクト解除
SDRAM 設定	: SDRAM を使用する
動作モード	: XrossFinder 使用時は DEBUG に、不使用時は NORMAL にしてください
USB チャンネル設定	: 使用するサンプルプログラムに合わせてください

CPU ボードの設定を製品出荷時の状態とし、以下の各スイッチの設定を行ってください。

<p>SW1</p>	<p>クロックモード : MODE0                  エリア 0 空間バス幅 : 16bit</p>
<p>SW2</p>	<p>FlashROM 設定 : FlashROM 使用                  FlashROM ライトプロテクト設定 : ライトプロテクト解除                  SDRAM 設定 : SDRAM 使用</p>
<p>DEBUG SS1 NORMAL</p>	<p>動作モード : XrossFinder 使用時</p>
<p>DEBUG SS1 NORMAL</p>	<p>動作モード : XrossFinder 不使用時</p>
<p>FUNC HOST JSW1</p>	<p>USB チャンネル設定 : USB ホスト使用時</p>
<p>FUNC HOST JSW1</p>	<p>USB チャンネル設定 : USB ファンクション使用時</p>

Fig1.2-1 動作モード設定

### 1. 3 開発環境について

本サンプルプログラムは統合開発環境 High-performance Embedded Workshop(以下、「HEW」という)を用いて開発されております。サンプルプログラムに対応する開発環境、コンパイラのバージョンは次のようになります。

フォルダ	開発環境	バージョン	コンパイラ名	バージョン	備考
shc	HEW	V 3.01.08.000 以降	SHC ※1	V9.0.1.0 以降	SuperHファミリ用C/C++コンパイラパッケージに付属

※1 「SuperHファミリ用C/C++コンパイラ」です。ルネサス エレクトロニクス社のウェブサイトより評価版をダウンロードできます。

### 1. 4 ワークスペースについて

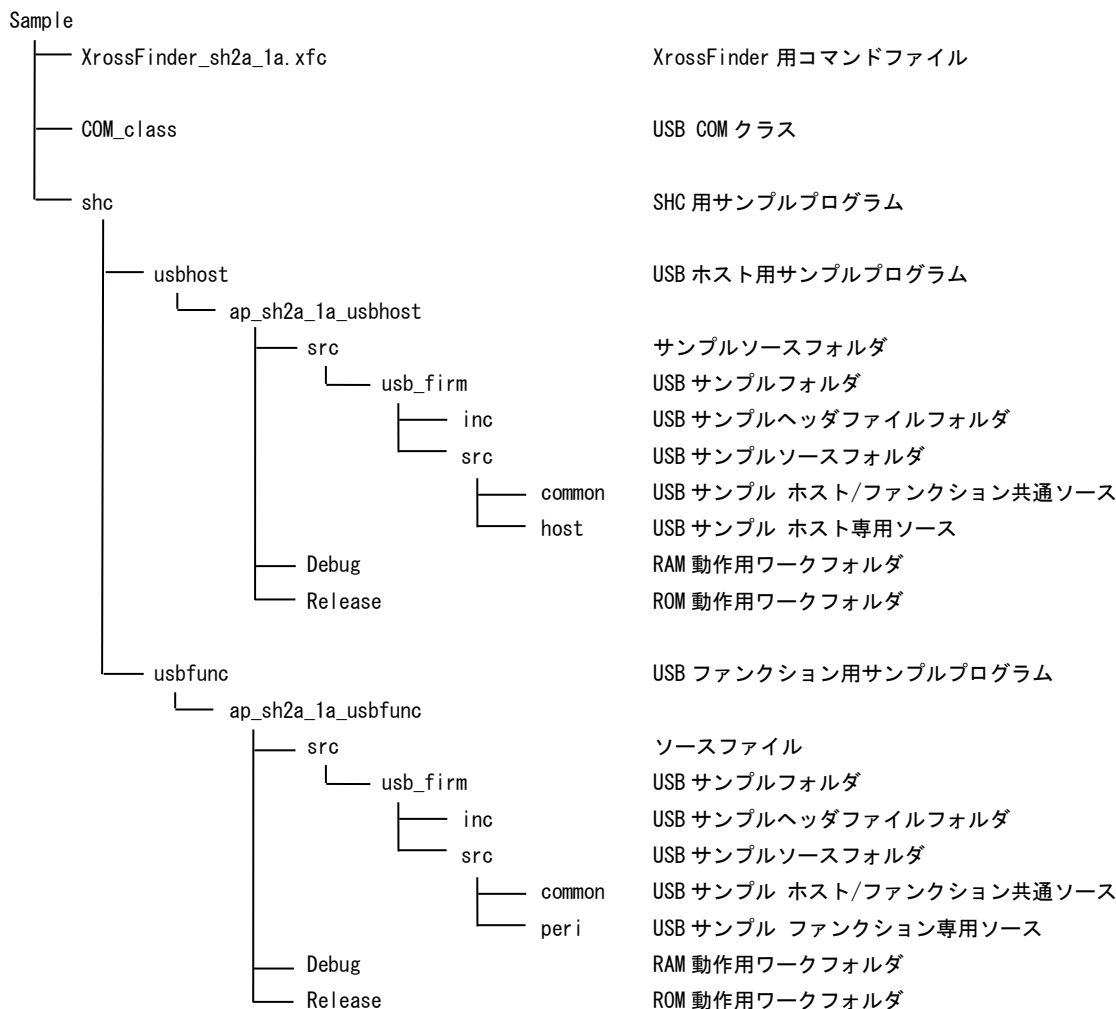
本サンプルプログラムの統合開発環境 HEW ワークスペースは次のフォルダに格納されています。

サンプルプログラム	フォルダ
USB ホストサンプルプログラム	¥Sample¥shc¥usbhost¥ap_sh2a_1a_usbhost. hws
USB ファンクションサンプルプログラム	¥Sample¥shc¥usbfunc¥ap_sh2a_1a_usbfunc. hws

## 2. サンプルプログラムの構成

### 2. 1 フォルダ構成

サンプルプログラムは下記のようなフォルダ構成になっています。



## 2. 2 ファイル構成

### 2. 2. 1 USB ホスト用サンプルプログラムのファイル構成

USB ホスト用サンプルプログラムは以下のファイルで構成されています。

<shc%usbhost フォルダ内>

ap\_sh2a\_1a\_usbhost. hws … HEW 用ワークスペースファイル

<shc%usbhost%ap\_sh2a\_1a\_usbhost フォルダ内>

ap\_sh2a\_1a\_usbhost. hwp … HEW 用プロジェクトファイル

<shc%usbhost%ap\_sh2a\_1a\_usbhost%src フォルダ内>

main. c … メイン処理  
 timer. c … タイマ処理  
 sci. c … シリアル処理  
 can. c … CAN 処理  
 boot. c … CPU 初期化処理  
 common. h … 共通ヘッダファイル  
 7263. h … SH7263 内部レジスタ定義ヘッダファイル  
 vector. c … 割込ベクタテーブル  
 BordDepend. h … ボード依存ヘッダファイル  
 typedefine. h … タイプ定義ファイル  
 section. src … セクション定義ファイル

<shc%usbhost%ap\_sh2a\_1a\_usbhost%src%usb\_firm フォルダ内>

c\_version. h … バージョン情報ファイル

<shc%usbhost%ap\_sh2a\_1a\_usbhost%src%usb\_firm%src フォルダ内>

c\_datatbl. c … USB サンプル用データバッファ  
 h\_main. c … USB サンプル用メインソース  
 common … USB サンプル ホスト/ファンクション共通ソースフォルダ  
 host … USB サンプル ホスト専用ソースフォルダ

<shc%usbhost%ap\_sh2a\_1a\_usbhost%Debug フォルダ内>

ap\_sh2a\_1a\_usbhost. abs … RAM 動作用オブジェクトファイル(elf 形式)  
 ap\_sh2a\_1a\_usbhost. mot … RAM 動作用モトローラ S フォーマット形式ファイル  
 ap\_sh2a\_1a\_usbhost. map … RAM 動作用マップファイル  
 コンパイル後は、.obj,.lib 等のファイルが生成されます

<shc%usbhost%ap\_sh2a\_1a\_usbhost%Release フォルダ内>

ap\_sh2a\_1a\_usbhost. abs … ROM 動作用オブジェクトファイル(elf 形式)  
 ap\_sh2a\_1a\_usbhost. mot … ROM 動作用モトローラ S フォーマット形式ファイル  
 ap\_sh2a\_1a\_usbhost. map … ROM 動作用マップファイル  
 コンパイル後は、.obj,.lib 等のファイルが生成されます

## 2. 2. 2 USB ファンクション用サンプルプログラムのファイル構成

SHC 用サンプルプログラムは以下のファイルで構成されています。

## &lt;shc%usbfunc フォルダ内&gt;

ap\_sh2a\_1a\_usbfunc.hws … HEW 用ワークスペースファイル

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc フォルダ内&gt;

ap\_sh2a\_1a\_usbfunc.hwp … HEW 用プロジェクトファイル

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc%src フォルダ内&gt;

main.c … メイン処理  
 timer.c … タイマ処理  
 sci.c … シリアル処理  
 can.c … CAN 処理  
 boot.c … CPU 初期化処理  
 common.h … 共通ヘッダファイル  
 7263.h … SH7263 内部レジスタ定義ヘッダファイル  
 vector.c … 割込ベクタテーブル  
 BordDepend.h … ボード依存ヘッダファイル  
 typedefine.h … タイプ定義ファイル  
 section.src … セクション定義ファイル

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc%src%usb\_firm フォルダ内&gt;

c\_version.h … バージョン情報ファイル

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc%src%usb\_firm%src フォルダ内&gt;

c\_datatbl.c … USB サンプル用データバッファ  
 p\_main.c … USB サンプル用メインソース  
 common … USB サンプル ホスト/ファンクション共通ソースフォルダ  
 per i … USB サンプル ファンクション専用ソースフォルダ

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc%Debug フォルダ内&gt;

ap\_sh2a\_1a\_usbfunc.abs … RAM 動作用オブジェクトファイル(elf 形式)  
 ap\_sh2a\_1a\_usbfunc.mot … RAM 動作用モトローラ S フォーマット形式ファイル  
 ap\_sh2a\_1a\_usbfunc.map … RAM 動作用マップファイル  
 コンパイル後は、.obj,.lib 等のファイルが生成されます

## &lt;shc%usbfunc%ap\_sh2a\_1a\_usbfunc%Release フォルダ内&gt;

ap\_sh2a\_1a\_usbfunc.abs … ROM 動作用オブジェクトファイル(elf 形式)  
 ap\_sh2a\_1a\_usbfunc.mot … ROM 動作用モトローラ S フォーマット形式ファイル  
 ap\_sh2a\_1a\_usbfunc.map … ROM 動作用マップファイル  
 コンパイル後は、.obj,.lib 等のファイルが生成されます

### 3. ビルド・デバッグ方法

#### 3. 1 USB ホスト用サンプルプログラム

##### (1) ビルド

- ① HEW を起動し、ap\_sh2a\_1a\_usbhost.hws を読み込みます。
- ② 最初の読み込みを行ったときに、「ワークスペース (Workspace) が移動しました」という内容の確認メッセージが表示されますので「はい」を選択してください。
- ③ 最初の読み込みを行ったときに、コンパイラバージョンによって、バージョンの選択を行うダイアログが表示されることがあります。表示された場合には、使用するコンパイラバージョンを選択してください。
- ④ [Build] ボタン横のリストボックス [Configuration Section] から、[Debug] または [Release] を選択します。  
[Debug] を選択した場合、¥Debug ワークフォルダ内に RAM 動作用のオブジェクトが生成されます。  
[Release] を選択した場合、¥Release ワークフォルダ内に ROM 動作用のオブジェクトが生成されます。
- ⑤ メニューの [Build] - [Build] を実行してください。ap\_sh2a\_1a\_usbhost.mot、ap\_sh2a\_1a\_usbhost.abs が出力されます。このとき、マップファイルは、ワークフォルダに作成されます。

HEW の詳細な使用方法につきましては、HEW のマニュアルを参照してください。

##### (2) RAM 上でのデバッグ

- ① AP-SH2A-1A のスイッチを「1.2 動作モード」「Fig1.2-1 動作モード設定」に合わせて設定します。
- ② XsSight を起動し、¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを読み込みます。
- ③ ¥Debug フォルダ内の ap\_sh2a\_1a\_usbhost.abs を XsSight でダウンロードして動作を確認してください。

##### (3) ROM 上でのデバッグ

- ① AP-SH2A-1A のスイッチを「1.2 動作モード」「Fig1.2-1 動作モード設定」に合わせて設定します。
- ② ¥Release フォルダ内の ap\_sh2a\_1a\_usbhost.abs を XsSight で読み込みます。
- ③ ¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを読み込みます。
- ④ XsSight のメニューから FlashWriter EX を選択し、下図 Fig3.1-1 のように設定を行ってください。
- ⑤ START ボタンを押してプログラムの書き込みを行い、動作を確認してください。

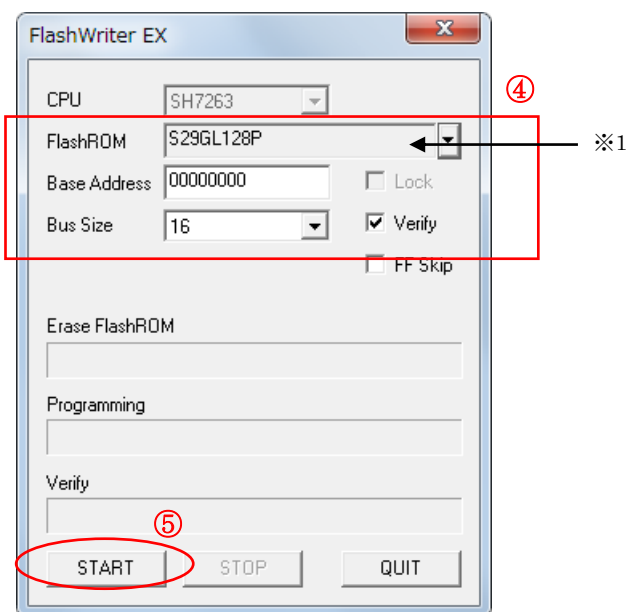


Fig3.1-1 FlashWriter EX for XsSight の設定



## (4) XsSight 不使用时の確認方法

## ・FlashWriter EX を使用する場合

- ① アダプタ (HJ-LINK / XrossFinder / XrossFinder Evo) を使用して PC とボードを繋ぎます。
- ② FlashWriter EX を起動して、「Table3.1-1 FlashWriter EX の設定」を参考に設定を行ってください。
- ③ FlashWriter EX で、¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを使用するように設定してください。
- ④ ¥Release フォルダ内の ap\_sh2a\_1a\_usbhost.mot をボードに書き込み、動作確認を行ってください。

FlashWriter EX の使用方法の詳細につきましては、FlashWriter EX のマニュアルを参照してください。

アダプタ設定	XrossFinder Evo 使用時は「XrossFinder Evo」 XrossFinder 使用時は「XrossFinder」 HJ-LINK 使用時は「HJ-LINK」
JTAG クロック (XrossFinder / XrossFinder Evo 使用時)	10MHz 以下
CPU	SH7263
BaseAddress	00000000
FlashROM	S29GL128P ※1
Bus Size	16

Table3.1-1 FlashWriter EX の設定

- ※1. 本ボードに実装されている FlashROM は、生産中止等の理由により変更することがございます。本アプリケーションノートでの設定は、「S29GL128P90TFIR20 (SPANSION)」が実装されているボードでの設定となります。お手元の CPU ボードに実装されている FlashROM の型番と異なっている場合には、お手元のボードに実装されている FlashROM の型番にあわせて設定を行ってください。

### 3. 2 USB ファンクション用サンプルプログラム

#### (1) ビルド

- ① HEW を起動し、ap\_sh2a\_1a\_usbfunc.hws を読み込みます。
- ② 最初の読み込みを行ったときに、「ワークスペース (Workspace) が移動しました」という内容の確認メッセージが表示されますので「はい」を選択してください。
- ③ 最初の読み込みを行ったときに、コンパイラバージョンによって、バージョンの選択を行うダイアログが表示されることがあります。表示された場合には、使用するコンパイラバージョンを選択してください。
- ④ [Build] ボタン横のリストボックス [Configuration Section] から、[Debug] または [Release] を選択します。  
[Debug] を選択した場合、¥Debug ワークフォルダ内に RAM 動作のオブジェクトが生成されます。  
[Release] を選択した場合、¥Release ワークフォルダ内に ROM 動作のオブジェクトが生成されます。
- ⑤ メニューの [Build] - [Build] を実行してください。ap\_sh2a\_1a\_usbfunc.mot、ap\_sh2a\_1a\_usbfunc.abs が出力されます。このとき、マップファイルは、ワークフォルダに作成されます。

HEW の詳細な使用方法につきましては、HEW のマニュアルを参照してください。

#### (2) RAM 上でのデバッグ

- ① AP-SH2A-1A のスイッチを「1.2 動作モード」「Fig1.2-1 動作モード設定」に合わせて設定します。
- ② XsSight を起動し、¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを読み込みます。
- ③ ¥Debug フォルダ内の ap\_sh2a\_1a\_usbfunc.abs を XsSight でダウンロードして動作を確認してください。

#### (3) ROM 上でのデバッグ

- ① AP-SH2A-1A のスイッチを「1.2 動作モード」「Fig1.2-1 動作モード設定」に合わせて設定します。
- ② ¥Release フォルダ内の ap\_sh2a\_1a\_usbfunc.abs を XsSight で読み込みます。
- ③ ¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを読み込みます。
- ④ XsSight のメニューから FlashWriter EX を選択し、下図 Fig3.2-1 のように設定を行ってください。
- ⑤ START ボタンを押してプログラムの書き込みを行い、動作を確認してください。

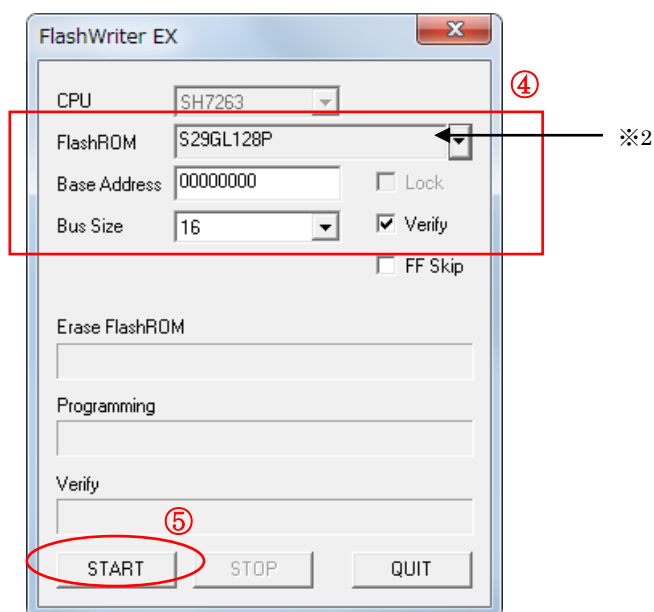


Fig3.2-1 FlashWriter EX for XsSight の設定

## (4) XsSight 不使用时の確認方法

## ・FlashWriter EX を使用する場合

- ① アダプタ (HJ-LINK / XrossFinder / XrossFinder Evo) を使用して PC とボードを繋ぎます。
- ② FlashWriter EX を起動して、「Table3.2-1 FlashWriter EX の設定」を参考に設定を行ってください。
- ③ FlashWriter EX で、¥sample フォルダ直下にある XrossFinder\_sh2a\_1a.xfc コマンドファイルを使用するように設定してください。
- ④ ¥Release フォルダ内の ap\_sh2a\_1a\_usbhost.mot をボードに書き込み、動作確認を行ってください。

FlashWriter EX の使用方法の詳細につきましては、FlashWriter EX のマニュアルを参照してください。

アダプタ設定	XrossFinder Evo 使用時は「XrossFinder Evo」 XrossFinder 使用時は「XrossFinder」 HJ-LINK 使用時は「HJ-LINK」
JTAG クロック (XrossFinder / XrossFinder Evo 使用時)	10MHz 以下
CPU	SH7263
BaseAddress	00000000
FlashROM	S29GL128P ※2
Bus Size	16

Table3.2-1 FlashWriter EX の設定

※2. 本ボードに実装されている FlashROM は、生産中止等の理由により変更することがございます。本アプリケーションノートでの設定は、「S29GL128P90TFIR20 (SPANSION)」が実装されているボードでの設定となります。お手元の CPU ボードに実装されている FlashROM の型番と異なっている場合には、お手元のボードに実装されている FlashROM の型番にあわせて設定を行ってください。

## 4. 動作説明

### 4. 1 USB ホスト用サンプルプログラムの動作説明

#### 4. 1. 1 USB ホスト用サンプルプログラム概要

USB ホスト用サンプルプログラムは、下記の動作を行います。

- ROM 動作の場合、起動時に Flash ROM 内のプログラムやデータを読み出して SDRAM にコピーします。その後、SDRAM 上でアプリケーションを開始します。
- SCIF (SC13) でエコーバックを行います。(送受信割り込み使用)  
SCIF から受信した値をそのまま、SCIF へ送信します。  
COM ポートの設定は、38400bps、ビット長 8、パリティなし、ストップビット 1、フロー制御なしです。  
動作確認は、パソコン上のターミナルソフト (ハイパーターミナルなど) を使用して行ってください。
- LD1 (緑の LED) を 500msec 間隔で ON/OFF します。(CMT0 割り込み使用)
- LD2 (緑の LED) を 1000msec 間隔で ON/OFF します。(CMT1 割り込み使用)
- CAN I/F でエコーバックを行います。以下の設定で、受信したデータをそのまま送信します。  
CAN の設定は、送信 ID:B' 10101010100、受信 ID:B' 10101010101、スタンダードフォーマット、データフレーム、データ長 1byte、通信速度 490kbps (TSG1=10 (11tq), TSE2=4 (5tq), SJW=0, BSP=0, BRP=1) です。
- CN2 のポートより方形波を出力します。周期とピン番号を次のページに示します。
- USB ホストに USB メモリを接続すると、SCIF (SC13) から接続状況とデバイス情報を出力します。 ※1  
※1. USB ホスト動作の詳細は、「4. 1. 2 USB ホスト動作」を参照してください。

GN2 方形波出力端子一覧

ピン番号	ピン名	周期	備考
8	PF30/AUDIO_CLK	20msec	CMT1 使用
9	PF29/SSIDATA3	20msec	CMT1 使用
10	PF28/SSIWS3	20msec	CMT1 使用
11	PF27/SSISCK3	20msec	CMT1 使用
12	PF26/SSIDATA2	20msec	CMT1 使用
13	PF25/SSIWS2	20msec	CMT1 使用
14	PF24/SSISCK2	20msec	CMT1 使用
17	PF23/SSIDATA1/LCD_VEPWC/AUDATA3	20msec	CMT1 使用
18	PF22/SSIWS1/LCD_VCPWC/AUDATA2	20msec	CMT1 使用
19	PF21/SSISCK1/LCD_CLK	20msec	CMT1 使用
20	PF20/SSIDATA0/LCD_FLM	20msec	CMT1 使用
21	PF19/SSIWS0/LCD_M_DISP	20msec	CMT1 使用
22	PF18/SSISCK0/LCD_CL2	20msec	CMT1 使用
23	PF17/#FCE/LCD_CL1	20msec	CMT1 使用
24	PF16/FRB/LCD_DON	20msec	CMT1 使用
29	PF15/NAF7/LCD_DATA15/SD_CD	10msec	CMT0 使用
30	PF14/NAF6/LCD_DATA14/SD_WP	10msec	CMT0 使用
31	PF13/NAF5/LCD_DATA13/SD_D1	10msec	CMT0 使用
32	PF12/NAF4/LCD_DATA12/SD_D0	10msec	CMT0 使用
33	PF11/NAF3/LCD_DATA11/SD_CLK	10msec	CMT0 使用
34	PF10/NAF2/LCD_DATA10/SD_CMD	10msec	CMT0 使用
35	PF9/NAF1/LCD_DATA9/SD_D3	10msec	CMT0 使用
36	PF8/NAF0/LCD_DATA8/SD_D2	10msec	CMT0 使用
39	PF7/FSC/LCD_DATA7/#SCS1	10msec	CMT0 使用
40	PF6/FOE/LCD_DATA6/SS01	10msec	CMT0 使用
41	PF5/FCDE/LCD_DATA5/SSI1	10msec	CMT0 使用
42	PF4/#FWE/LCD_DATA4/SSCK1	10msec	CMT0 使用
43	PF3/TCLKD/LCD_DATA3/#SCS0	10msec	CMT0 使用
44	PF2/TCLKC/LCD_DATA2/SS00	10msec	CMT0 使用
45	PF1/TCLKB/LCD_DATA1/SSI0	10msec	CMT0 使用
46	PF0/TCLKA/LCD_DATA0/SSCK0	10msec	CMT0 使用

信号名に#がついているものは負論理を表します。

## 4. 1. 2 USB ホスト動作

以下の手順に従い、USB ホストの動作を確認してください。

- ① CPU ボードとパソコンをシリアルケーブルで接続します。
- ② パソコン上でターミナルソフト(ハイパーターミナルなど)を起動し、COM ポートの設定を行います。  
COM ポートの設定は、38400bps、ビット長 8、パリティなし、ストップビット 1、フロー制御なしです。
- ③ CPU ボードに電源を投入し、サンプルプログラムを動作させます。
- ④ USB ホストポート(CN7)に USB メモリを挿入します。
- ⑤ USB メモリを挿入すると、自動的に USB メモリの接続状況とデバイス情報がターミナルソフト上に表示されます。以下のような表示が出ていることを確認してください。

```
Attach -----  
  
Hi-Speed Device-----  
  
USB Address is 3-----  
  
Get configuration (Device descriptor) ----- PASS!   Device descriptor dump :  
    12 01 00 02 00 00 00 40 8f 05 87 63 41 01 01 02 03 01  
Device descriptor fields :          bLength          : 0x12  
  
          :  
          :
```

※. 以下に、デバイスのディスクリプタ情報が表示されます。

上記の表示も含め、表示される情報は接続する USB メモリにより異なります。

- ⑥ 上記の表示が出たことを確認した後に、USB メモリを抜いてください。ターミナルソフト上に切断状況が表示されます。以下のような表示が出ていることを確認してください。

```
Detach -----
```

- ⑦ 以上で USB ホストの動作は終了です。

**注意** : USB ホスト動作において、対応する USB デバイスは、ハイスピードとフルスピードに限ります。

ロースピードデバイス是对应していませんので、ご注意ください。

#### 4. 1. 3 USB ホスト用サンプルプログラム API 一覧

USB ホスト用サンプルプログラムの主要 API 一覧は以下の通りです。

API 名	内容説明
H_ModeInit	USB ホストドライバ初期化
H_DataIn	データイントランザクション実行
H_DataOut	データアウトトランザクション実行
H_CtrlTransfer	コントロール転送実行

#### 4. 1. 4 USB サンプルプログラム注意事項

サンプルプログラムのソース「USB サンプル ホスト/ファンクション共通ソース common フォルダ」内に 1msec 間待ち処理を行う「usb\_delay\_1ms」関数がありますが、この関数は正確に 1msec の待ち処理を行うものではありません。本サンプルプログラムでは、あくまで、指定した時間以上の待ち時間を得るために使用しておりますので、ご注意ください。正確に 1msec の待ち時間が必要な場合には、上記の関数のループ回数を調整していただくか、タイマ（CMT、MTU2 等）をご使用ください。

## 4. 2 USB ファンクション用サンプルプログラムの動作説明

### 4. 2. 1 USB ファンクション用サンプルプログラム概要

USB ファンクション用サンプルプログラムは、下記の動作を行います。

- ROM 動作の場合、起動時にFlash ROM 内のプログラムやデータを読み出して SDRAM にコピーします。その後、SDRAM 上でアプリケーションを開始します。
- SCIF (SC13) でエコーバックを行います。(送受信割り込み使用)  
SCIF から受信した値をそのまま、SCIF へ送信します。  
COM ポートの設定は、38400bps、ビット長 8、パリティなし、ストップビット 1、フロー制御なしです。  
動作確認は、パソコン上のターミナルソフト (ハイパーターミナルなど) を使用して行ってください。
- LD1 (緑の LED) を 500msec 間隔で ON/OFF します。(CMT0 割り込み使用)
- LD2 (緑の LED) を 1000msec 間隔で ON/OFF します。(CMT1 割り込み使用)
- CAN I/F でエコーバックを行います。以下の設定で、受信したデータをそのまま送信します。  
CAN の設定は、送信 ID:B' 10101010100、受信 ID:B' 10101010101、スタンダードフォーマット、データフレーム、データ長 1byte、通信速度 490kbps (TSG1=10 (11tq), TSE2=4 (5tq), SJW=0, BSP=0, BRP=1) です。
- CN2 のポートより方形波を出力します。周期とピン番号を次のページに示します。
- USB ファンクションをパソコンに接続すると、仮想 COM ポートとして認識され、USB シリアルとしてエコーバックを行います。 ※1  
※1. USB ファンクション動作の詳細は、「4.2.2 USB ファンクション動作」を参照してください。



CN2 方形波出力端子一覧

ピン番号	ピン名	周期	備考
8	PF30/AUDIO_CLK	20msec	CMT1 使用
9	PF29/SSIDATA3	20msec	CMT1 使用
10	PF28/SSIWS3	20msec	CMT1 使用
11	PF27/SSISCK3	20msec	CMT1 使用
12	PF26/SSIDATA2	20msec	CMT1 使用
13	PF25/SSIWS2	20msec	CMT1 使用
14	PF24/SSISCK2	20msec	CMT1 使用
17	PF23/SSIDATA1/LCD_VEPWC/AUDATA3	20msec	CMT1 使用
18	PF22/SSIWS1/LCD_VCPWC/AUDATA2	20msec	CMT1 使用
19	PF21/SSISCK1/LCD_CLK	20msec	CMT1 使用
20	PF20/SSIDATA0/LCD_FLM	20msec	CMT1 使用
21	PF19/SSIWS0/LCD_M_DISP	20msec	CMT1 使用
22	PF18/SSISCK0/LCD_CL2	20msec	CMT1 使用
23	PF17/#FCE/LCD_CL1	20msec	CMT1 使用
24	PF16/FRB/LCD_DON	20msec	CMT1 使用
29	PF15/NAF7/LCD_DATA15/SD_CD	10msec	CMT0 使用
30	PF14/NAF6/LCD_DATA14/SD_WP	10msec	CMT0 使用
31	PF13/NAF5/LCD_DATA13/SD_D1	10msec	CMT0 使用
32	PF12/NAF4/LCD_DATA12/SD_D0	10msec	CMT0 使用
33	PF11/NAF3/LCD_DATA11/SD_CLK	10msec	CMT0 使用
34	PF10/NAF2/LCD_DATA10/SD_CMD	10msec	CMT0 使用
35	PF9/NAF1/LCD_DATA9/SD_D3	10msec	CMT0 使用
36	PF8/NAF0/LCD_DATA8/SD_D2	10msec	CMT0 使用
39	PF7/FSC/LCD_DATA7/#SCS1	10msec	CMT0 使用
40	PF6/FOE/LCD_DATA6/SS01	10msec	CMT0 使用
41	PF5/FCDE/LCD_DATA5/SSI1	10msec	CMT0 使用
42	PF4/#FWE/LCD_DATA4/SSCK1	10msec	CMT0 使用
43	PF3/TCLKD/LCD_DATA3/#SCS0	10msec	CMT0 使用
44	PF2/TCLKC/LCD_DATA2/SS00	10msec	CMT0 使用
45	PF1/TCLKB/LCD_DATA1/SSI0	10msec	CMT0 使用
46	PF0/TCLKA/LCD_DATA0/SSCK0	10msec	CMT0 使用

信号名に#がついているものは負論理を表します。

#### 4. 2. 2 USB ファンクション動作

以下の手順に従い、USB シリアル動作を確認してください。

Win10 よりも前の OS での USB ファンクション動作確認は、あらかじめ USB 仮想シリアルドライバを PC にインストールしておく必要があります。

USB 仮想シリアルドライバのインストール方法につきましては、「AN178 USB 仮想シリアルドライバ インストールガイド」を参照してください。

- ① USB ケーブルを使い、パソコンの USB ポートと CPU ボードの USB ファンクションポート (CN8) を接続します。
- ② CPU ボードに電源を投入し、サンプルプログラムを動作させます。
- ③ パソコン上でターミナルソフト (ハイパーターミナルなど) を起動し、COM ポートの設定を行います。  
その際、使用する COM ポートは「AN178 USB 仮想シリアルドライバ インストール方法」で確認した仮想 COM ポートを選択してください。  
COM ポートの設定は、38400bps、ビット長 8、パリティなし、ストップビット 1、フロー制御なしです。
- ④ ターミナルソフトを使用し、エコーバックが行われることを確認してください。
- ⑤ 以上で USB シリアル動作は終了です。

#### 4. 2. 3 USB サンプルプログラム注意事項

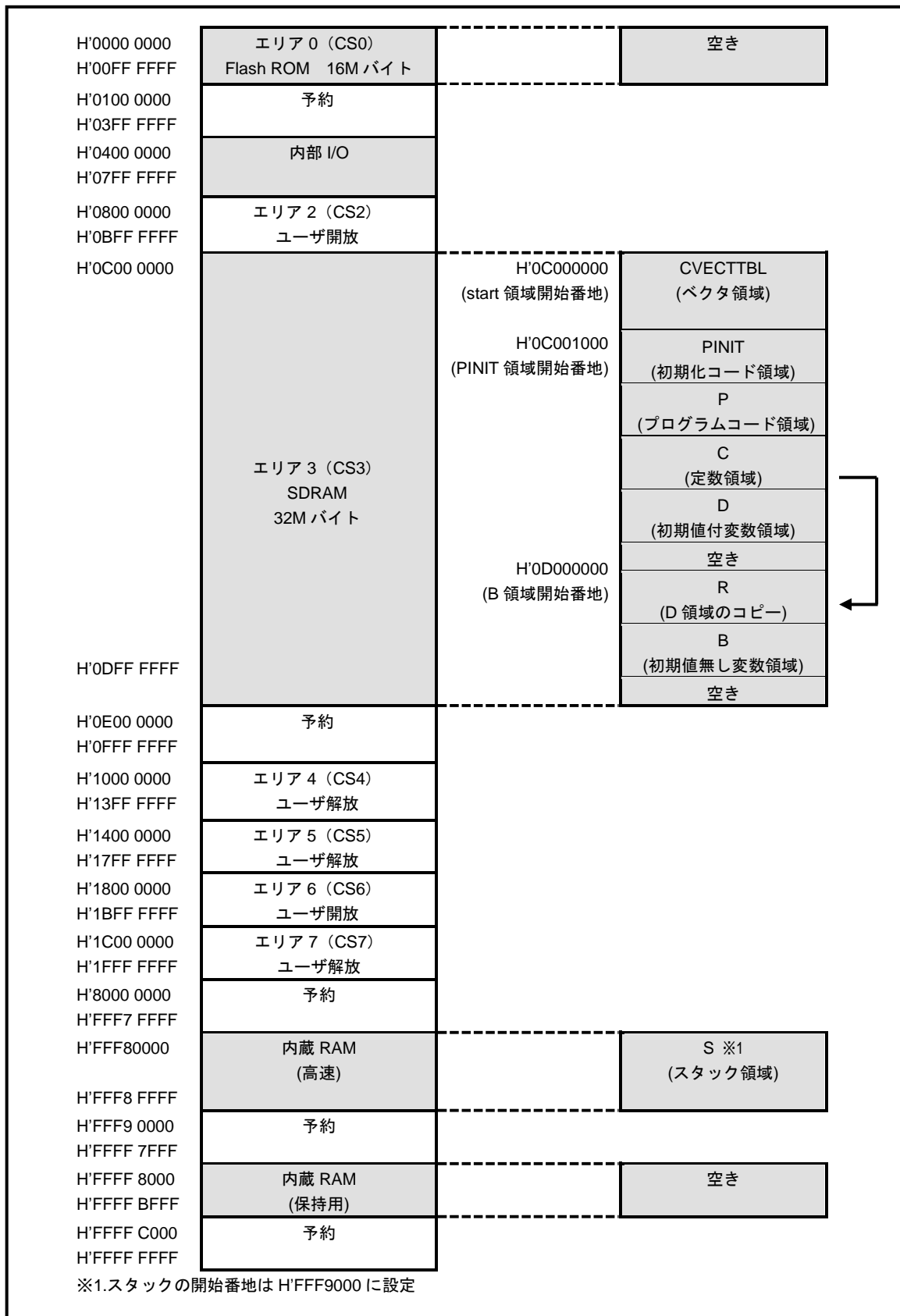
サンプルプログラムのソース「USB サンプル ホスト/ファンクション共通ソース common フォルダ」内に 1msec 間待ち処理を行う「usb\_delay\_1ms」関数がありますが、この関数は正確に 1msec の待ち処理を行うものではありません。

本サンプルプログラムでは、あくまで、指定した時間以上の待ち時間を得るために使用しておりますので、ご注意ください。  
正確に 1msec の待ち時間が必要な場合には、上記の関数のループ回数を調整していただくか、タイマ (CMT, MTU2 等) をご使用ください。

4. 3 RAM 動作時のメモリマップ

メモリマップを以下に示します。

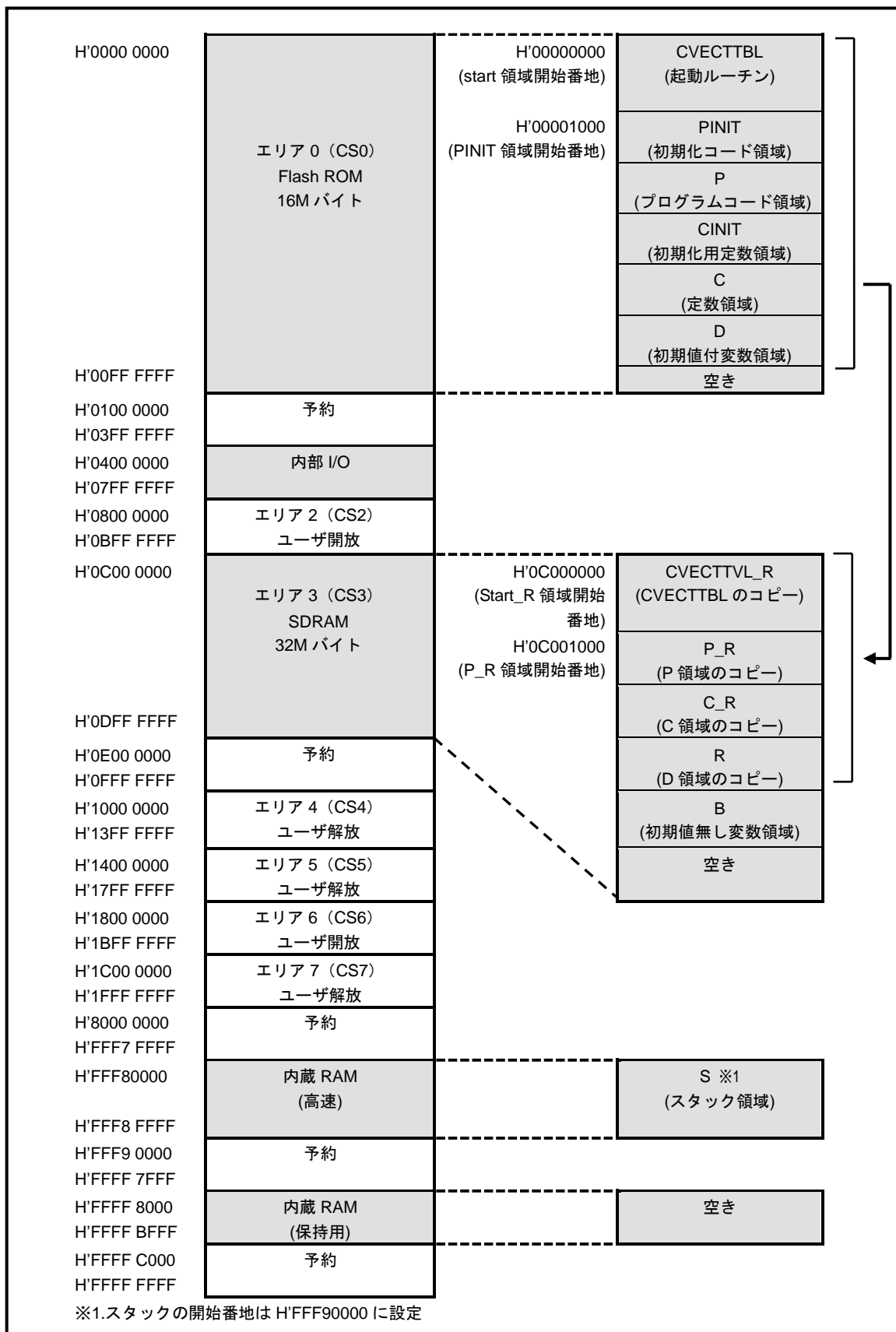
USB ホスト用、USB ファンクション用共にプログラム領域の割り当ては同じです。



4. 4 ROM 動作時のメモリマップ

メモリマップを以下に示します。

USB ホスト用、USB ファンクション用共にプログラム領域の割り当ては同じです。



## ご注意

- ・本文書の著作権は株式会社アルファプロジェクトが保有します。
- ・本文書の内容を無断で転載することは一切禁止します。
- ・本文書に記載されている USB 部分のサンプルソースの著作権はルネサス エレクトロニクス株式会社が保有します。
- ・本文書に記載されているサンプルプログラムの著作権は株式会社アルファプロジェクトが保有します。
- ・本文書に記載されている内容およびサンプルプログラムについての技術サポートは一切受け付けておりません。
- ・本サンプルプログラムに関して、ルネサス エレクトロニクス株式会社へのお問い合わせはご遠慮ください。
- ・本文書の内容およびサンプルプログラムに基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社およびルネサス エレクトロニクス株式会社では一切責任を負いませんのでご了承ください。
- ・本文書の内容については、万全を期して作成いたしました。が、万一ご不審な点、誤りなどお気づきの点がありましたら弊社までご連絡ください。
- ・本文書の内容は、将来予告なしに変更されることがあります。

## 商標について

- ・Windows®の正式名称は Microsoft®Windows®Operating System です。
- ・Microsoft、Windows は、米国 Microsoft Corporation.の米国およびその他の国における商標または登録商標です。
- ・Windows®10、Windows®8、Windows®7、Windows®XP は、米国 Microsoft Corporation.の商品名称です。
- ・本文書では下記のように省略して記載している場合がございます。ご了承ください。  
Windows®10 は Windows 10 もしくは Win10  
Windows®8 は Windows 8 もしくは Win8  
Windows®7 は Windows 7 もしくは Win7  
Windows®XP は Windows XP もしくは WinXP  
High-performance Embedded Workshop は HEW
- ・IBM-PC/AT は、米国 IBM 社の登録商標、商標または商品名称です。
- ・DOS/V は、日本 IBM 社の登録商標、商標または商品名称です。
- ・SH7263 は、ルネサス エレクトロニクス株式会社の登録商標、商標または商品名称です。
- ・SuperH は、ルネサス エレクトロニクス株式会社の登録商標、商標または商品名称です。
- ・その他の会社名、製品名は、各社の登録商標または商標です。



株式会社アルファプロジェクト  
〒431-3114  
静岡県浜松市東区積志町 8 3 4  
<https://www.apnet.co.jp>  
E-Mail : [query@apnet.co.jp](mailto:query@apnet.co.jp)