

AP-SH2A-3A(SH7670 CPU BOARD)

サンプルプログラム解説

4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
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
2. 2. 3 ネットワーク用サンプルプログラムのファイル構成	7
3. ビルド・デバッグ方法	8
3. 1 USB ホスト用サンプルプログラム	8
3. 2 USB ファンクション用サンプルプログラム	10
3. 3 ネットワーク用サンプルプログラム	12
4. 動作説明	14
4. 1 USB ホスト用サンプルプログラムの動作説明	14
4. 1. 1 USB ホスト用サンプルプログラム概要	14
4. 1. 2 USB ホスト動作	15
4. 1. 3 USB ホスト用サンプルプログラム API 一覧	16
4. 1. 4 USB サンプルプログラム注意事項	16
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	17
4. 2. 1 USB ファンクション用サンプルプログラム概要	17
4. 2. 2 USB ファンクション動作	18
4. 2. 3 USB サンプルプログラム注意事項	18
4. 3 ネットワーク用サンプルプログラムの動作説明	19
4. 3. 1 ネットワーク用サンプルプログラム概要	19
4. 3. 2 ネットワーク動作	20
4. 3. 3 ネットワークサンプルプログラム注意事項	21
4. 4 RAM 動作時のメモリマップ (USB ホスト、ファンクション用サンプルプログラム)	22
4. 5 ROM 動作時のメモリマップ (USB ホスト、ファンクション用サンプルプログラム)	23
4. 6 RAM 動作時のメモリマップ (ネットワーク用サンプルプログラム)	24
4. 7 ROM 動作時のメモリマップ (ネットワーク用サンプルプログラム)	25

1. 概要

1. 1 概要

本アプリケーションノートでは、AP-SH2A-3A 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-3A 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"> • USB ホスト デバイス情報表示 • シリアル通信 • タイマ割り込み
USB ファンクション用サンプルプログラム	<ul style="list-style-type: none"> • USB ファンクション 仮想シリアル通信 • シリアル通信 • タイマ割り込み
ネットワーク用サンプルプログラム	<ul style="list-style-type: none"> • ネットワーク通信 • シリアル通信 • タイマ割り込み

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。

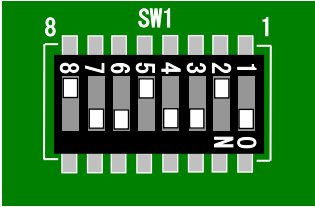
1. 2 動作モード

本サンプルプログラムは、AP-SH2A-3A 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-3A 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
USB ソフトウェア設定	: 無効
USB HOST VBUS 設定	: PE10 で制御
USB HOST 過電流検出	: PB8 で行う
動作モード	: XrossFinder 使用時は DEBUG に、未使用時は NORMAL にして下さい
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下さい

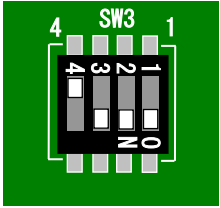
CPU ボードの設定を製品出荷時の状態とし、使用方法に合わせて以下の各スイッチの設定を行って下さい。

・ SW1



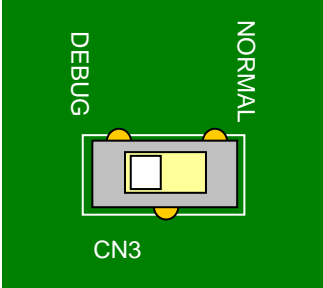
FlashROM 設定	:	FlashROM 使用
FlashROM ライトプロテクト設定	:	ライトプロテクト解除
SDRAM 設定	:	SDRAM 使用
USB ソフトウェア設定	:	無効
USB HOST VBUS 設定	:	PE10 で制御
USB HOST 過電流検出	:	PB8 で行う

・ SW3

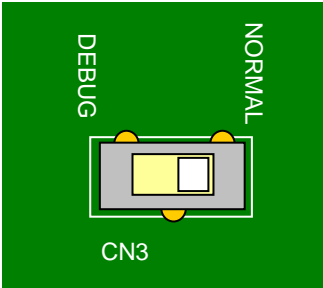


クロックモード	:	MODE0
エリア 0 空間バス幅	:	16bit

・ SS1

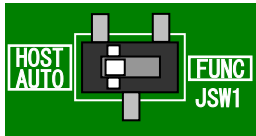


XrossFinder 使用時
DEBUG モード

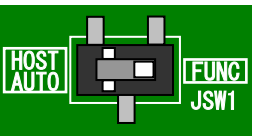


XrossFinder 未使用時
NORMAL モード

・ JSW1



USB ホスト使用時



USB ファンクション使用時

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。

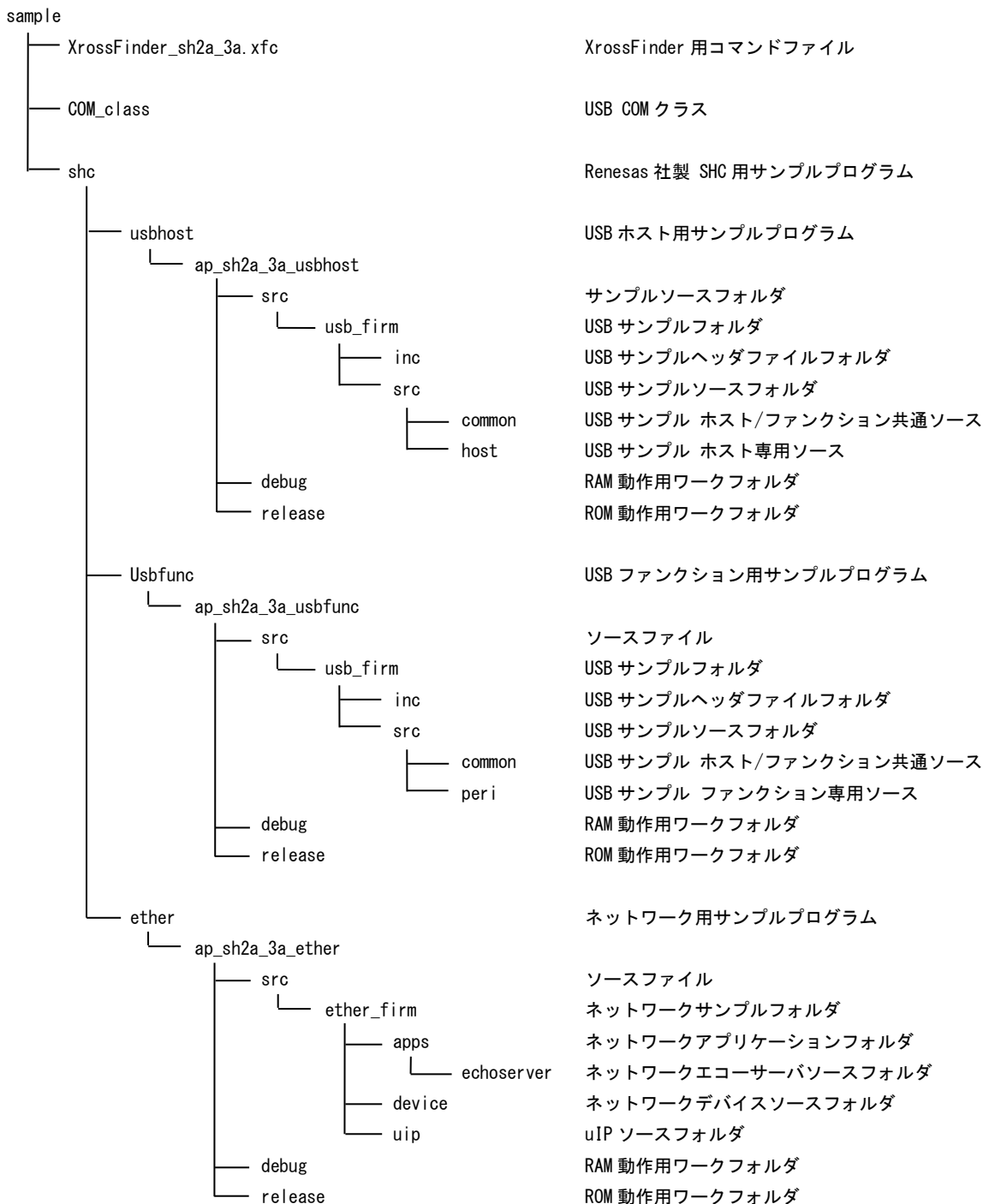
フォルダ	開発環境	バージョン	コンパイラ名	バージョン	備考
shc	HEW	V 3.01.08.000 以降	SHC ^{※1} (ルネサス エレクトロニクス)	V9.0.1.0 以降	SuperH ファミリ用 C/C++コンパイラパッケージに付属

※1： ルネサス エレクトロニクス社製「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。

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_3a_usbhost. hws … HEW 用ワークスペースファイル

<shc¥usbhost¥ap_sh2a_3a_usbhost フォルダ内>

ap_sh2a_3a_usbhost. hwp … HEW 用プロジェクトファイル

<shc¥usbhost¥ap_sh2a_3a_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
 boot. c … CPU 初期化処理
 common. h … 共通ヘッダファイル
 7670. h … SH7670 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_3a_usbhost¥src¥usb_firm フォルダ内>

c_version. h … ルネサス USB のバージョンバージョン情報定義ファイル

<shc¥usbhost¥ap_sh2a_3a_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_3a_usbhost¥src¥usb_firm¥inc フォルダ内>

c_debugprint. h … デバッグ用ヘッダファイル
 c_def7670. h … USB コントローラのレジスタ情報ヘッダファイル
 cfg_usb_sh7670. h … USB 用ユーザ設定ヘッダファイル
 usb. h … USB ドライバヘッダファイル
 usb_firm. h … USB 用 define 設定ヘッダファイル

<shc¥usbhost¥ap_sh2a_3a_usbhost¥Debug フォルダ内>

ap_sh2a_3a_usbhost. abs … RAM 動作用オブジェクトファイル(elf 形式)
 ap_sh2a_3a_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_3a_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_3a_usbhost¥Release フォルダ内>

ap_sh2a_3a_usbhost. abs … ROM 動作用オブジェクトファイル (elf 形式)
 ap_sh2a_3a_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_3a_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

USB ファンクション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%usbfunc フォルダ内>

ap_sh2a_3a_usbfunc. hws … HEW 用ワークスペースファイル

<shc%usbfunc%ap_sh2a_3a_usbfunc フォルダ内>

ap_sh2a_3a_usbfunc. hwp … HEW 用プロジェクトファイル

<shc%usbfunc%ap_sh2a_3a_usbfunc%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
 boot. c … CPU 初期化処理
 common. h … 共通ヘッダファイル
 7670. h … SH7670 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%usbfunc%ap_sh2a_3a_usbfunc%src%usb_firm フォルダ内>

c_version. h … バージョン情報ファイル

<shc%usbfunc%ap_sh2a_3a_usbfunc%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
 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

<shc%usbfunc%ap_sh2a_3a_usbfunc%Debug フォルダ内>

ap_sh2a_3a_usbfunc. abs … RAM 動作用オブジェクトファイル(elf 形式)
 ap_sh2a_3a_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_3a_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

<shc%usbfunc%ap_sh2a_3a_usbfunc%Release フォルダ内>

ap_sh2a_3a_usbfunc. abs … ROM 動作用オブジェクトファイル (elf 形式)
 ap_sh2a_3a_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_3a_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

2. 2. 3 ネットワーク用サンプルプログラムのファイル構成

ネットワーク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¥ether フォルダ内>

ap_sh2a_3a_ether.hws … HEW 用ワークスペースファイル

<shc¥ether¥ap_sh2a_3a_ether フォルダ内>

ap_sh2a_3a_ether.hwp … HEW 用プロジェクトファイル

<shc¥ether¥ap_sh2a_3a_ether¥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
 boot.c … CPU 初期化処理
 common.h … 共通ヘッダファイル
 7670.h … SH7670 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¥ether¥ap_sh2a_3a_ether¥src¥ether_firm フォルダ内>

apps … ネットワークアプリケーションフォルダ
 device … ネットワークデバイスソースフォルダ
 uip … uIP ソースフォルダ

<shc¥ether¥ap_sh2a_3a_ether¥src¥ether_firm¥apps フォルダ内>

echoserver … ネットワークエコーサーバソースフォルダ

<shc¥ether¥ap_sh2a_3a_ether¥Debug フォルダ内>

ap_sh2a_3a_ether.abs … RAM 動作用オブジェクトファイル(elf 形式)
 ap_sh2a_3a_ether.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_3a_ether.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¥ether¥ap_sh2a_3a_ether¥Release フォルダ内>

ap_sh2a_3a_ether.abs … ROM 動作用オブジェクトファイル(elf 形式)
 ap_sh2a_3a_ether.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_3a_ether.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_3a_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」を選択して下さい。
- 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下さい。
- 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 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 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] を実行して下さい。ap_sh2a_3a_usbhost.mot、ap_sh2a_3a_usbhost.abs が出力されます。このとき、マップファイルは、¥Debug フォルダ、もしくは、¥Release 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下さい。

(2) RAM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ホスト使用時に設定して下さい。
- ② Xs Sight を起動し、¥sample フォルダ直下にある XrossFinder_sh2a_3a.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_3a_usbhost.abs を Xs Sight でダウンロードして動作を確認して下さい。

(3) ROM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ホスト使用時に設定して下さい。
- ② ¥Release フォルダ内の ap_sh2a_3a_usbhost.abs を Xs Sight で読み込みます。
- ③ ¥sample フォルダ直下にある XrossFinder_sh2a_3a.xfc コマンドファイルを読み込みます。
- ④ Xs Sight 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、動作を確認して下さい。

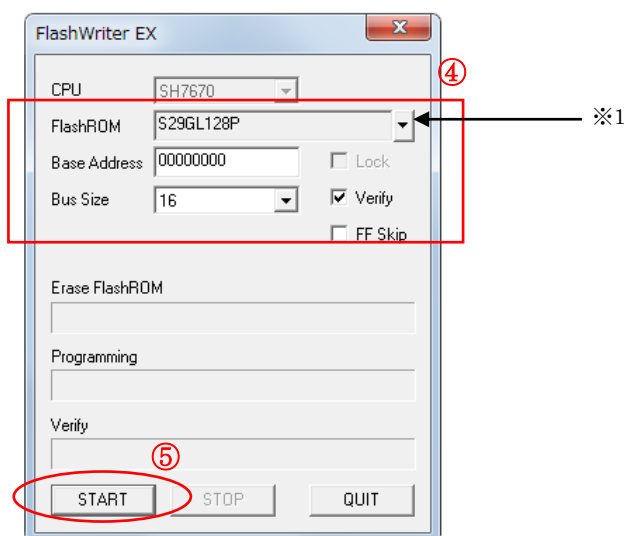


Fig3.1-1 FlashWriter EX for Xs Sight の設定

(4) Xs Sight 不使用時の確認方法

・FlashWriterEX 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 FlashROM の書き込みを行うために AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ホスト使用時に設定して下さい。
- ③ FlashWriterEX を起動して、「Table3.1-1 FlashWriterEX の設定」を参考に設定を行って下さい。
- ④ FlashWriterEX で、¥sample フォルダ直下にある XrossFinder_sh2a_3a. xfc コマンドファイルを使用するように設定して下さい。
- ⑤ ¥Release フォルダ内の ap_sh2a_3a_usbhost. mot をボードに書き込みます。
- ⑥ 書き込んだプログラムを動作させるために AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は NORMAL モード、JSW1 は USB ホスト使用時に設定して下さい。

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下さい。

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 使用時のみ)	10MHz 以下
CPU	SH7670
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下さい。

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_3a_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」を選択して下さい。
- 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下さい。
- 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 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 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] を実行して下さい。ap_sh2a_3a_usbfunc.mot、ap_sh2a_3a_usbfunc.abs が出力されます。このとき、マップファイルは、¥Debug フォルダ、もしくは、¥Release 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下さい。

(2) RAM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ファンクション使用時に設定して下さい。
- ② Xs Sight を起動し、¥sample フォルダ直下にある XrossFinder_sh2a_3a.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_3a_usbfunc.abs を Xs Sight でダウンロードして動作を確認して下さい。

(3) ROM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ファンクション使用時に設定して下さい。
- ② ¥Release フォルダ内の ap_sh2a_3a_usbfunc.abs を Xs Sight で読み込みます。
- ③ ¥sample フォルダ直下にある XrossFinder_sh2a_3a.xfc コマンドファイルを読み込みます。
- ④ Xs Sight 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、動作を確認して下さい。

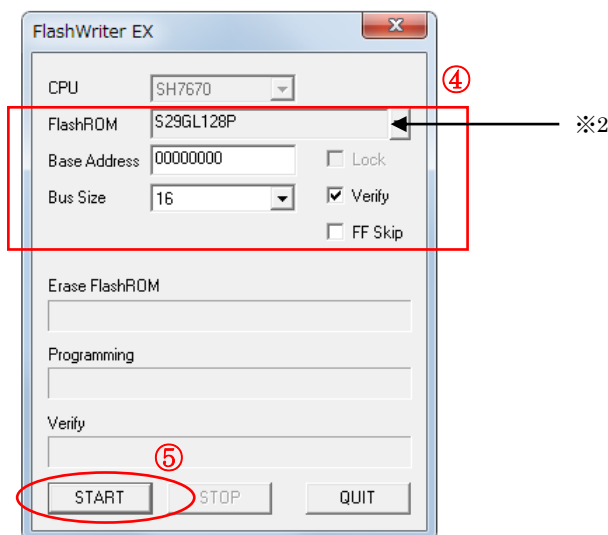


Fig3.2-1 FlashWriter EX for Xs Sight の設定

(4) Xs Sight 不使用時の確認方法

・FlashWriterEX 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 AP-SH2A-3A のスイッチを、「1.2 動作モード」「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モード、JSW1 は USB ファンクション使用時に設定して下さい。
- ③ FlashWriterEX を起動して、「Table3.2-1 FlashWriterEX の設定」を参考に設定を行って下さい。
- ④ FlashWriterEX で、¥sample フォルダ直下にある XrossFinder_sh2a_3a. xfc コマンドファイルを使用するように設定して下さい。
- ⑤ ¥Release フォルダ内の ap_sh2a_3a_usbfunc. mot をボードに書き込みます。
- ⑥ AP-SH2A-3A のスイッチを、「1.2 動作モード」「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は NORMAL モード、JSW1 は USB ファンクション使用時に設定して下さい。

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下さい。

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 使用時のみ)	10MHz 以下
CPU	SH7670
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下さい。

3. 3 ネットワーク用サンプルプログラム

(1) ビルド

- ① HEW を起動し、ap_sh2a_3a_ether.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」を選択して下さい。
- 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下さい。
- 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 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 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] を実行して下さい。ap_sh2a_3a_ether.mot、ap_sh2a_3a_ether.abs が出力されます。
このとき、マップファイルは、¥Debug フォルダ、もしくは、¥Release 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下さい。

(2) RAM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モードに設定して下さい。(JSW1 の設定はどちらでも構いません)
- ② Xs Sight を起動し、¥sample フォルダ直下にある XrossFinder_sh2a_3a.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_3a_ether.abs を Xs Sight でダウンロードして動作を確認して下さい。

(3) ROM 上でのデバッグ

- ① AP-SH2A-3A のスイッチを、「1.2 動作モード」 「Fig1. 2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モードに設定して下さい。(JSW1 の設定はどちらでも構いません)
- ② ¥Release フォルダ内の ap_sh2a_3a_ether.abs を Xs Sight で読み込みます。
- ③ ¥sample フォルダ直下にある XrossFinder_sh2a_3a.xfc コマンドファイルを読み込みます。
- ④ Xs Sight のメニューから FlashWriter EX を選択し、下図 Fig3. 3-1 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、動作を確認して下さい。

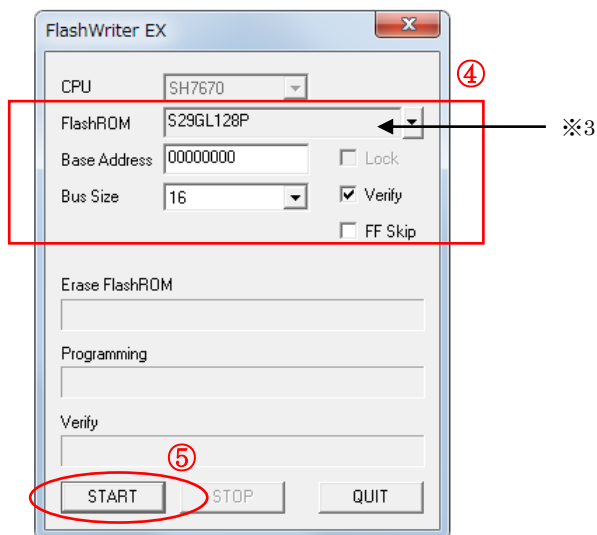


Fig3.3-1 FlashWriter EX for Xs Sight の設定

(4) Xs Sight 不使用時の確認方法

・FlashWriterEX 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 AP-SH2A-3A のスイッチを、「1.2 動作モード」「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は DEBUG モードに設定して下さい。(JSW1 の設定はどちらでも構いません)
- ③ FlashWriterEX を起動して、「Table3.3-1 FlashWriterEX の設定」を参考に設定を行って下さい。
- ④ FlashWriterEX で、¥sample フォルダ直下にある XrossFinder_sh2a_3a. xfc コマンドファイルを使用するように設定して下さい。
- ⑤ ¥Release フォルダ内の ap_sh2a_3a_ether. mot をボードに書き込みます。
- ⑥ AP-SH2A-3A のスイッチを、「1.2 動作モード」「Fig1.2-1 動作モード設定」を参考に設定します。なお、SS1 は NORMAL モードに設定して下さい。(JSW1 の設定はどちらでも構いません)

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下さい。

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 使用時のみ)	10MHz 以下
CPU	SH7670
BaseAddress	00000000
FlashROM	S29GL128P ※3
Bus Size	16

Table3.3-1 FlashWriter EX の設定

- ※3. 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下さい。

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。
- SCIF2 でエコーバックを行います。(送受信割り込み使用)
SCIF2 から受信した値をそのまま、SCIF2 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) を使用して行って下さい。
- 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。(CMT0 割り込み使用)
- 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に示します。
- USB ホストに USB メモリを接続すると、SCIF2 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。

CN2 方形波出力端子一覧

ピン番号	ピン名	周期	備考
23	PE07/ST1_D7/SSIWS1	10msec	CMT0 使用
24	PE06/ST1_D6/SSIDATA1	10msec	CMT0 使用
25	PE05/ST1_D5/#RTS1	10msec	CMT0 使用
26	PE04/ST1_D4/#CTS1	10msec	CMT0 使用
27	PE03/ST1_D3/SCK1	10msec	CMT0 使用
28	PE02/ST1_D2/RxD1	10msec	CMT0 使用
29	PE01/ST1_D1/TxD1	10msec	CMT0 使用
33	PG23/#HIFCS	10msec	CMT0 使用
34	PG22/HIFRS	10msec	CMT0 使用
35	PG21/#HIFWR	10msec	CMT0 使用
36	PG20/#HIFRD	10msec	CMT0 使用
37	PG19/#HIFINT	10msec	CMT0 使用
38	PG18/HIFDREQ	10msec	CMT0 使用
39	PG17/HIFRDY	10msec	CMT0 使用
40	PG16/HIFEFL	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(CN6)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 Port -----  
  
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 Port -----
```

- ⑦ 以上で USB ホストの動作は終了です。

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 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。
- SCIF2 でエコーバックを行います。(送受信割り込み使用)
SCIF2 から受信した値をそのまま、SCIF2 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) を使用して行って下さい。
- 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。(CMT0 割り込み使用)
- 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
23	PE07/ST1_D7/SSIWS1	10msec	CMT0 使用
24	PE06/ST1_D6/SSIDATA1	10msec	CMT0 使用
25	PE05/ST1_D5/#RTS1	10msec	CMT0 使用
26	PE04/ST1_D4/#CTS1	10msec	CMT0 使用
27	PE03/ST1_D3/SCK1	10msec	CMT0 使用
28	PE02/ST1_D2/RxD1	10msec	CMT0 使用
29	PE01/ST1_D1/TxD1	10msec	CMT0 使用
33	PG23/#HIFCS	10msec	CMT0 使用
34	PG22/HIFRS	10msec	CMT0 使用
35	PG21/#HIFWR	10msec	CMT0 使用
36	PG20/#HIFRD	10msec	CMT0 使用
37	PG19/#HIFINT	10msec	CMT0 使用
38	PG18/HIFDREQ	10msec	CMT0 使用
39	PG17/HIFRDY	10msec	CMT0 使用
40	PG16/HIFEFL	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 (CN7) 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。

注意 : USB ファンクション動作において、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は対応しておりませんので、ご注意ください。

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 等) をご使用ください。

4. 3 ネットワーク用サンプルプログラムの動作説明

4. 3. 1 ネットワーク用サンプルプログラム概要

ネットワーク用サンプルプログラムは、TCP/IP プロトコルスタックに uIP (マイクロアイビー) を使用しています。uIP に関する詳細は、uIP の関連書籍やウェブページを参照してください。

ネットワーク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。
- SCIF2 でエコーバックを行います。(送受信割り込み使用)
SCIF2 から受信した値をそのまま、SCIF2 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) を使用して行って下さい。
- 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。(CMT0 割り込み使用)
- 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に示します。
- EtherNet でエコーバックを行います※1
※1. ネットワーク動作の詳細は、「4. 3. 2 ネットワーク動作」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
23	PE07/ST1_D7/SSIWS1	10msec	CMT0 使用
24	PE06/ST1_D6/SSIDATA1	10msec	CMT0 使用
25	PE05/ST1_D5/#RTS1	10msec	CMT0 使用
26	PE04/ST1_D4/#CTS1	10msec	CMT0 使用
27	PE03/ST1_D3/SCK1	10msec	CMT0 使用
28	PE02/ST1_D2/RxD1	10msec	CMT0 使用
29	PE01/ST1_D1/TxD1	10msec	CMT0 使用
33	PG23/#HIFCS	10msec	CMT0 使用
34	PG22/#HIFRS	10msec	CMT0 使用
35	PG21/#HIFWR	10msec	CMT0 使用
36	PG20/#HIFRD	10msec	CMT0 使用
37	PG19/#HIFINT	10msec	CMT0 使用
38	PG18/#HIFDREQ	10msec	CMT0 使用
39	PG17/#HIFRDY	10msec	CMT0 使用
40	PG16/#HIFEFL	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. 3. 2 ネットワーク動作

ネットワーク動作に必要な推奨環境は以下のとおりです。

パーソナルコンピュータ	PC/AT 互換機
OS	WindowsXP/7/8/10
LAN ポート	10/100BASE-TX 以上対応の LAN ポート

(1) ネットワーク設定

本 CPU ボードのネットワーク設定は以下のようになっています。

IP アドレス	192.168.1.200
サブネットマスク	255.255.255.0
ゲートウェイ	192.168.1.254
MAC アドレス	00-0C-7B-27-XX-XX ※1

※1. XX-XX の値はボードごとに異なります

上記設定のうち、IP アドレス・サブネットマスク・ゲートウェイの設定はサンプルプログラム内の「main.c」で行われています。

また、MAC アドレスは EEPROM の先頭 3 ワード(6 バイト)に格納されています。格納イメージを以下に示します。

アドレス	格納値
先頭アドレス	0x000C
先頭アドレス+1 ワード	0x7B27
先頭アドレス+2 ワード	0xXXXX ※2

※2. XXXX の値はボードごとに異なります

本製品の MAC アドレスは、弊社が米国電気電子学会(IEEE)より取得したアドレスになります。

MAC アドレスを変更される際は、お客様にて IEEE より MAC アドレスを取得してください。

(2) ネットワーク動作内容

以下の手順に従い、ネットワークの動作を確認してください。

- ① CPU ボードの LAN コネクタ (CN8) とパソコンを LAN クロスケーブルで接続します。
- ② パソコン上でネットワークの設定を行います。

以下に示す内容で設定を行ってください。

IP アドレス	192.168.1.201
サブネットマスク	255.255.255.0
ゲートウェイ	192.168.1.254

- 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 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、TCP/IP の設定を行います。
TCP/IP の設定は、ホストアドレス「192.168.1.200」、ポート番号「50000」です。
- ⑤ ターミナルソフト上で接続が確認できましたら、任意のパケットを送信してください。
エコーバック動作が確認できれば終了です。

4. 3. 3 ネットワークサンプルプログラム注意事項

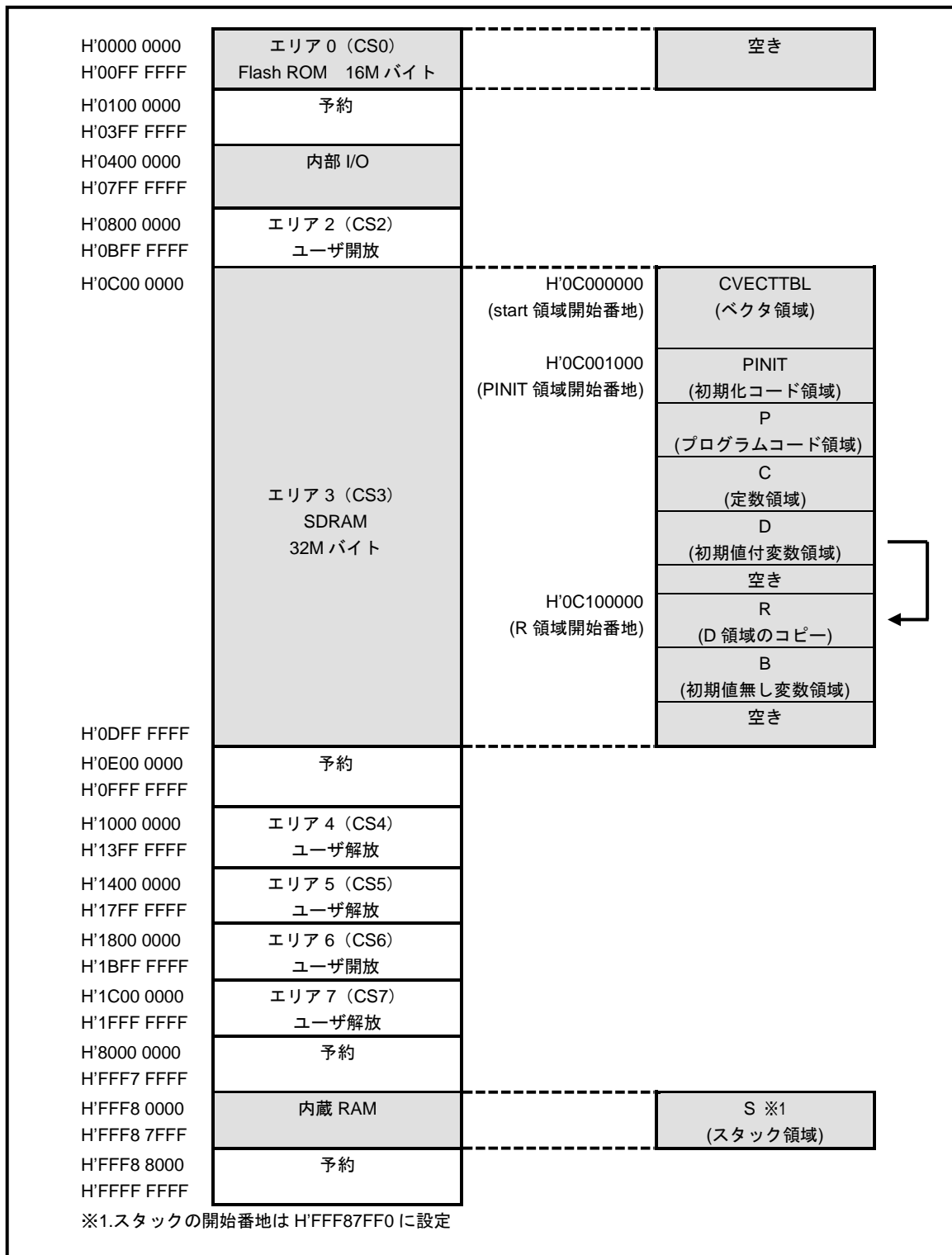
サンプルプログラムのソース「ネットワークデバイスソースフォルダ」内に 100us 間待ち処理を行う「LOOP_100us」定義がありますが、この定義は正確に 100us の待ち処理を行うものではありません。

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 100us の待ち時間が必要な場合には、上記の定義の値を調整していただくか、タイマ (CMT 等) をご使用ください。

4. 4 RAM 動作時のメモリマップ (USB ホスト、ファンクション用サンプルプログラム)

メモリマップを以下に示します。

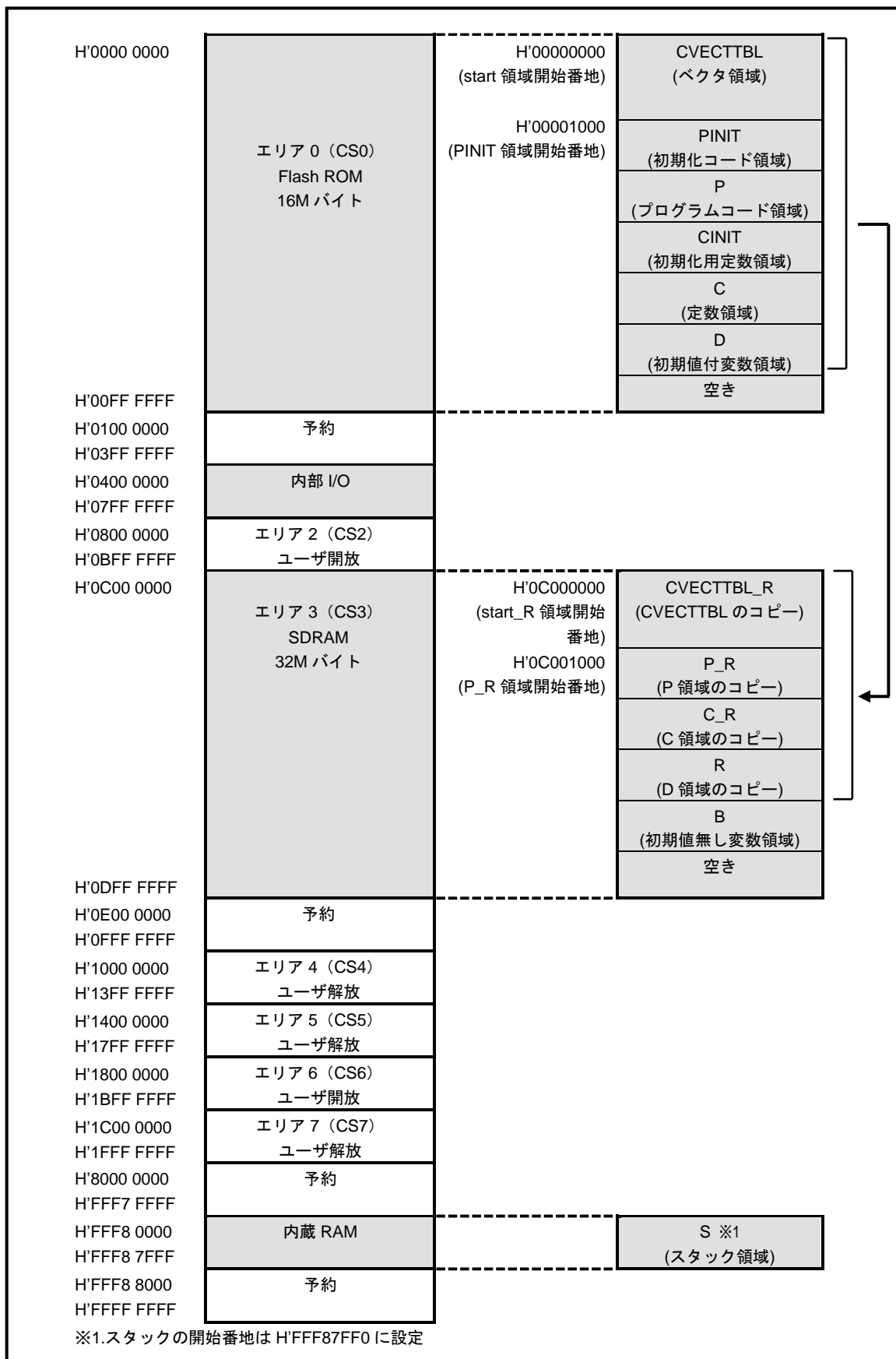
USB ホスト用、USB ファンクション用共にプログラム領域の割り当ては同じです。



4. 5 ROM 動作時のメモリマップ (USB ホスト、ファンクション用サンプルプログラム)

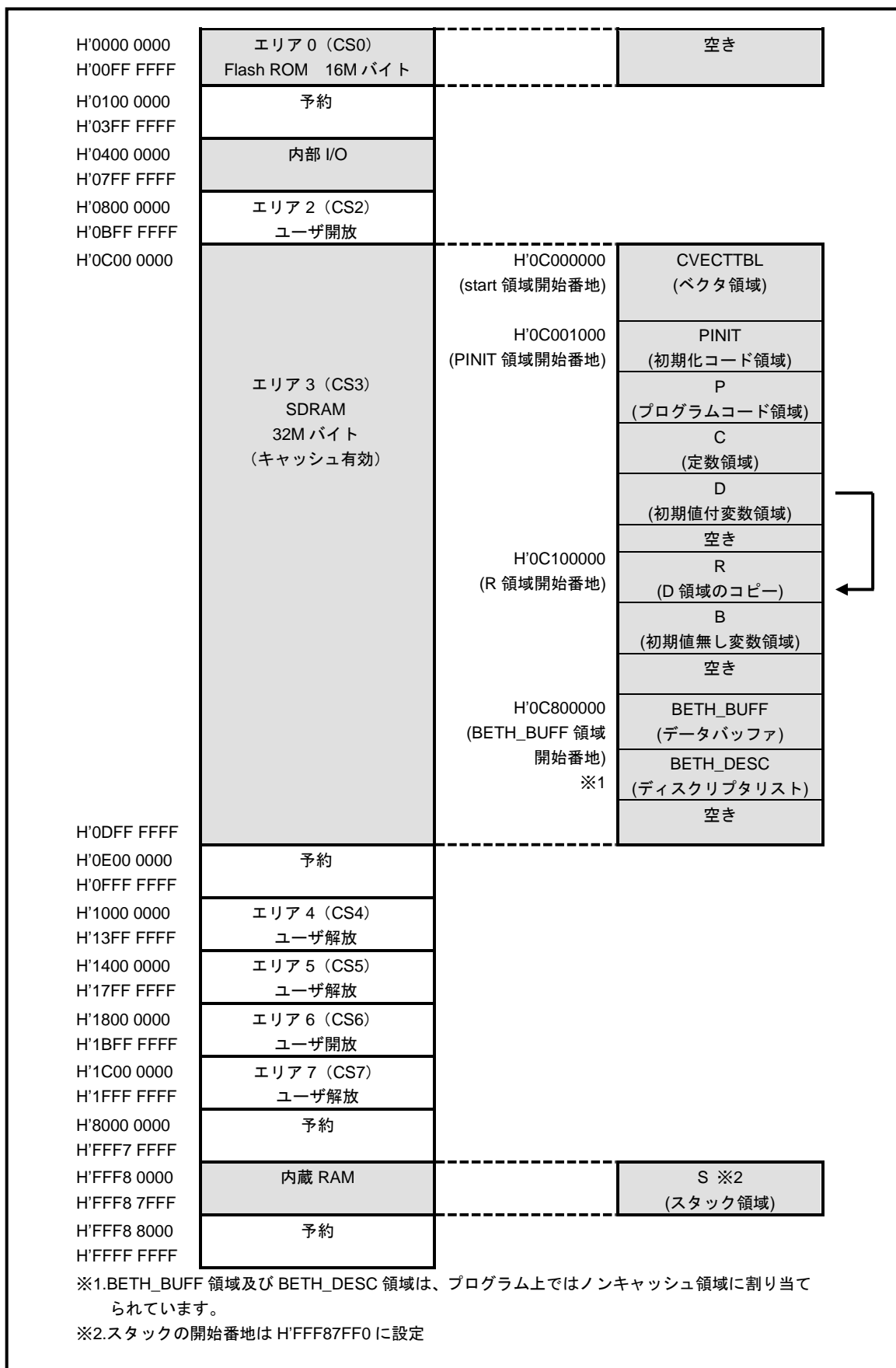
メモリマップを以下に示します。

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。



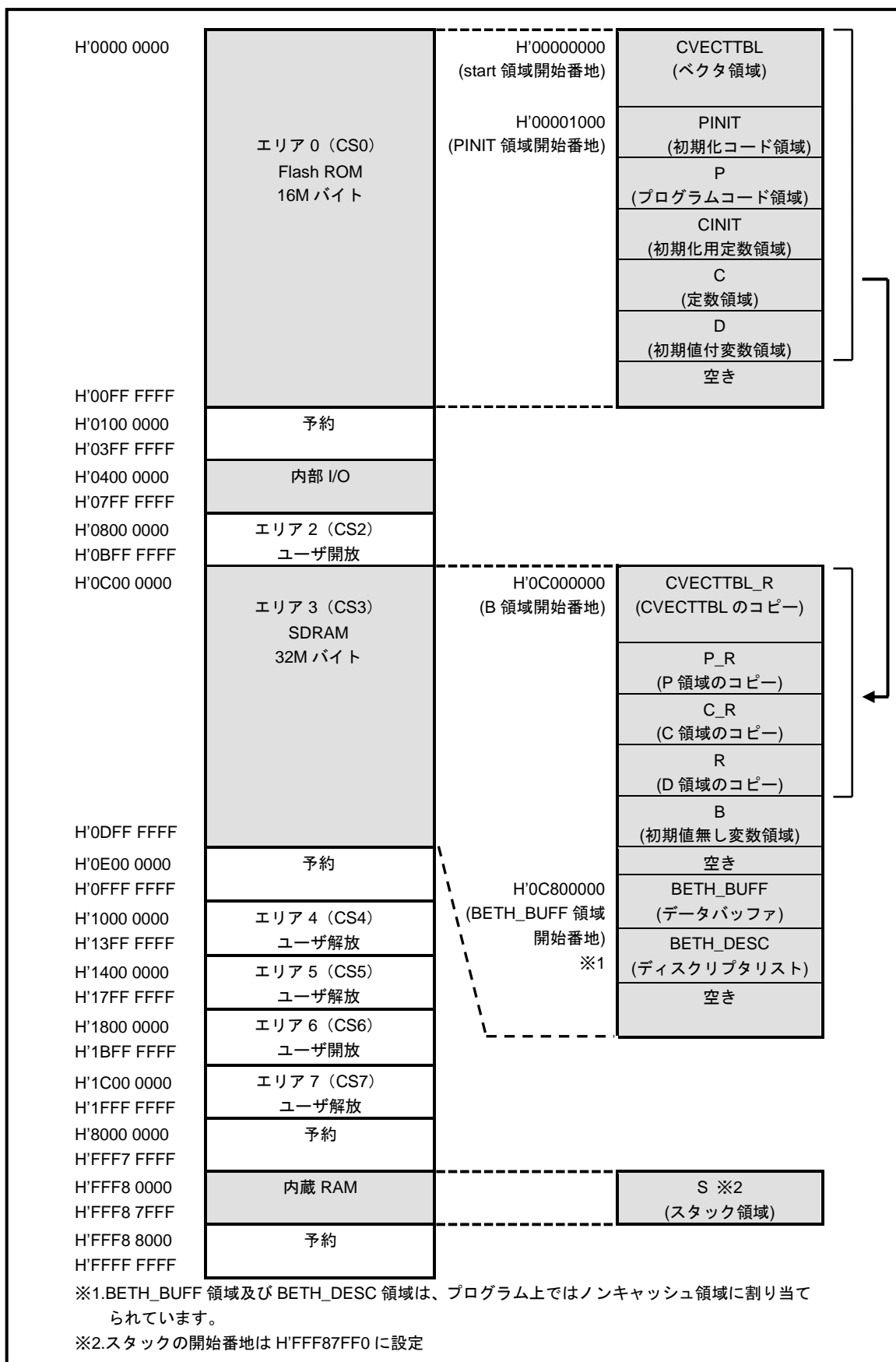
4. 6 RAM 動作時のメモリマップ (ネットワーク用サンプルプログラム)

ネットワーク用サンプルプログラムのメモリマップを以下に示します。



4. 7 ROM 動作時のメモリマップ (ネットワーク用サンプルプログラム)

ネットワーク用サンプルプログラムのメモリマップを以下に示します。



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ネットワークサンプルプログラム内の uIP(TCP/IP プロトコルスタック)には、BSD ライセンスが規定されています。
BSD ライセンスは、無保証であることの明記と著作権表示だけを再配布の条件とするライセンス規定です。
上記ライセンス規定に従い、uIP は無保証であり、著作権は「Adam Dunkels and the Swedish Institute of Computer Science」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。万が一不審な点、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

- ・ SH7670 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。
- ・ 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
- ・ 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。

 **ALPHA PROJECT**
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