

# AP-RX63N-0A (RX63N CPU BOARD) CS+版 USB ホストサンプルプログラム解説

2.1 版 2023 年 10 月 02 日

## 1. 概要

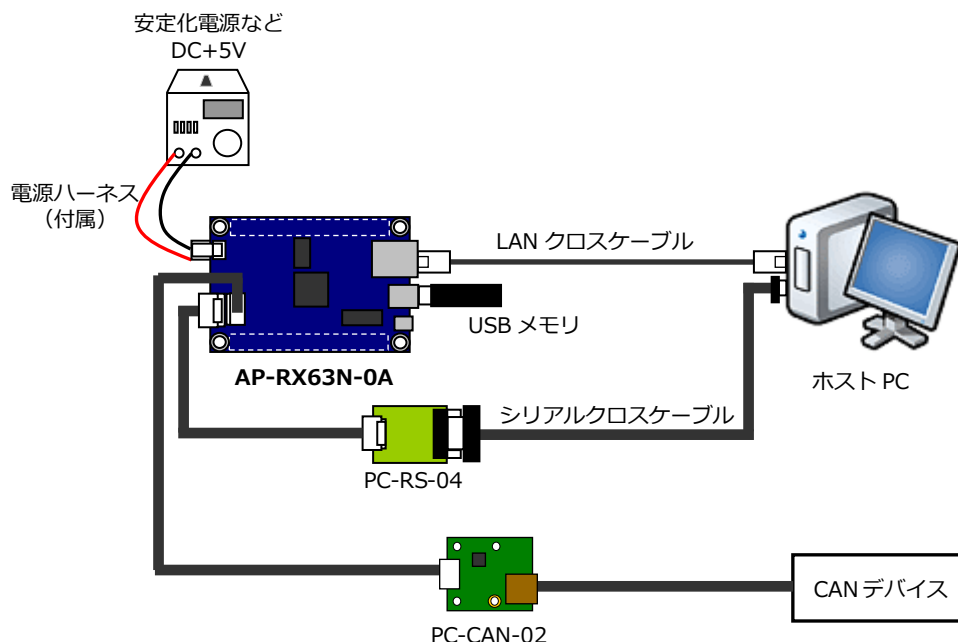
### 1.1 概要

本アプリケーションノートでは、弊社の Web サイトにて公開している AP-RX63N-0A のサンプルプログラムのうち「¥Sample¥ap\_rx63n\_0a\_sample\_host\_cs」以下にある「USB ホストサンプルプログラム」について説明します。AP-RX63N-0A の「USB ファンクションサンプルプログラム」につきましては、弊社 Web サイトにて公開中のアプリケーションノート「AN1515 CS+版 USB ファンクションサンプルプログラム解説」を参照してください。

| サンプルプログラム                       | 動作内容  |
|---------------------------------|---|
| AP-RX63N-0A<br>USB ホストサンプルプログラム | <ul style="list-style-type: none"> <li>・ USB ホスト ファイル書き込み</li> <li>・ ネットワーク通信</li> <li>・ シリアル通信</li> <li>・ タイマ割り込み</li> <li>・ CAN 通信</li> </ul> |

### 1.2 接続概要

「USB ホストサンプルプログラム」の動作を確認する上で必要な CPU ボードとホスト PC 間の接続例を以下に示します。詳細な接続に関しては後述の「3.動作内容」を参照してください。



### 1.3 本サンプルプログラムについて

本サンプルプログラムは、ルネサス エレクトロニクス株式会社提供のミドルウェア及びドライバを AP-RX63N-0A に移植しています。

また、タイマ、シリアル通信、クロック設定については、ルネサス エレクトロニクス株式会社提供の Peripheral Driver Generator を使用して作成しております。

各ミドルウェア及びドライバの詳細については、以下の資料を参照してください。

|   |
|---|
| ● USB ホスト   |
| ・ 資料名<br>Renesas USB MCU and USB ASSP USB Host Mass Storage Class Driver (HMSC) |
| ● ネットワーク通信  |
| ・ 資料名<br>ネットワークソフトウェアライブラリ 超小型 TCP/IP プロトコルスタック [高速版]                           |
| ● CAN 通信  |
| ・ 資料名<br>RX600 シリーズ CAN アプリケーションプログラミングインターフェース                                 |
| ● タイマ、シリアル通信、クロック設定   |
| ・ 資料名<br>Peripheral Driver Generator V2.07 リファレンスマニュアル (RX63N,RX631 グループ)       |

(※) 資料をダウンロードする際にはルネサス エレクトロニクス株式会社の MYRENESAS への登録が必要となります。

### 1.4 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「CS+」を用いて開発されています。

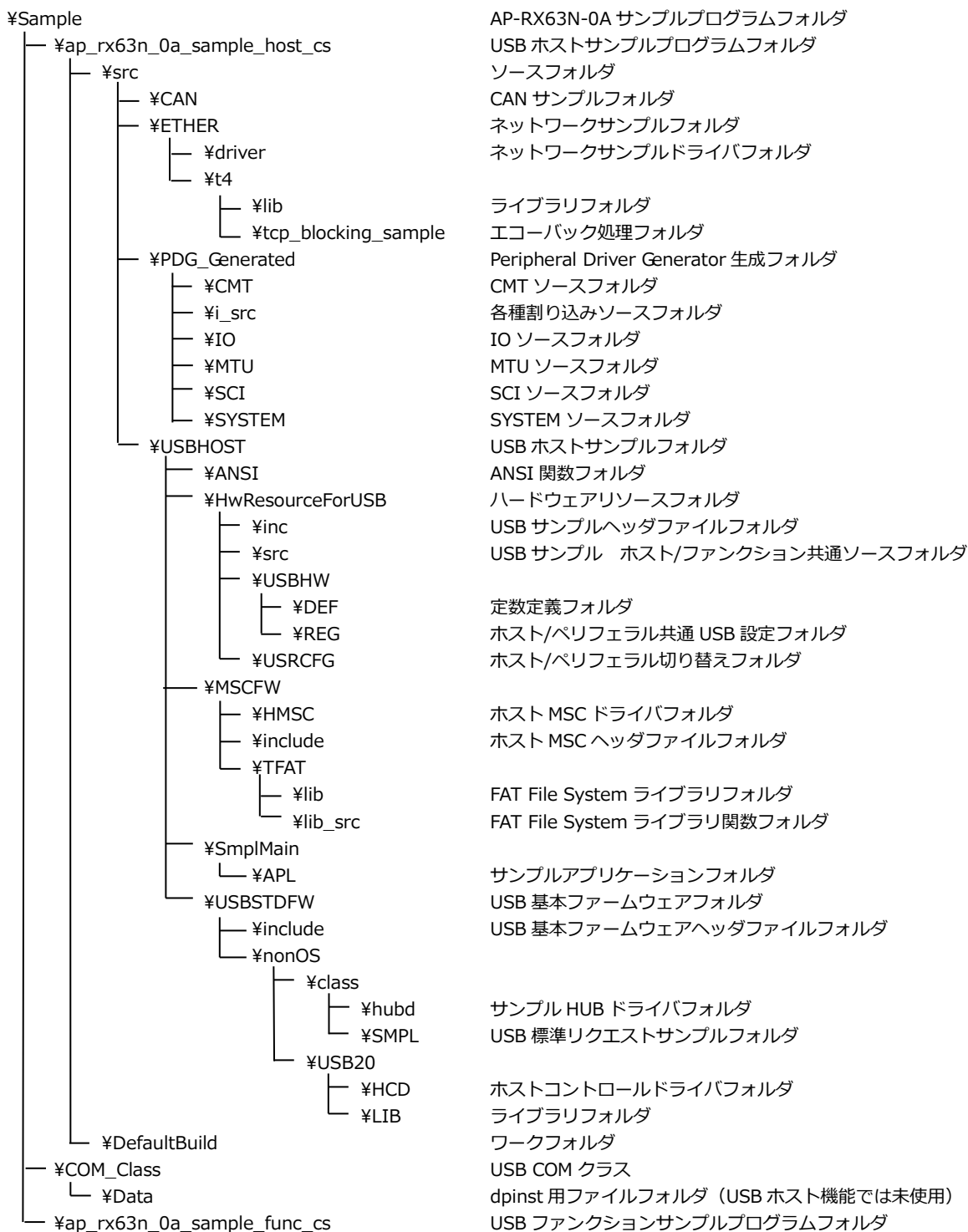
本サンプルプログラムに対応する開発環境、コンパイラのバージョンは次の通りです。

| ソフトウェア          | バージョン    | 備考 |
|-----------------|----------|----|
| CS+             | v8.04.00 | –  |
| RX 用コンパイラ CC-RX | V3.02.00 | –  |

## 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。



(※ USB ファンクションサンプルプログラムに関しては  
アプリケーションノート

「AN1515 CS+版 USB ファンクションサンプルプログラム」  
を参照してください)

## 2.2 ファイル構成

USB ホスト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。

本章では、サンプルプログラムの作成にあたって追加したファイルについてのみ記述し、ミドルウェア・ドライバ等の既存のファイルに関しては説明を省略してあります。

|  |                        |
|--|------------------------|
| <¥Sample¥ap_rx63n_0a_sample_host_cs¥>            |                        |
| ap_rx63n_0a_sample_host_cs.mtpj                  | … CS+用プロジェクトファイル       |
| ap_rx63n_0a_sample_host_cs.rcpe                  | … e2studio 用プロジェクトファイル |
| <¥Sample¥ap_rx63n_0a_sample_host_cs¥src>         |                        |
| ap_rx63n_0a_sample_host_cs.c                     | … メイン処理                |
| BoardDepend.h                                    | … ボード依存定義ヘッダファイル       |
| clock_init.c                                     | … 初期化時クロック設定処理         |
| clock_init.h                                     | … 初期化時クロック設定処理ヘッダファイル  |
| common.h   | … 共通ヘッダファイル            |
| hwsetup.c  | … 初期化処理                |
| hwsetup.h  | … 初期化処理ヘッダファイル         |
| iodef.h  | … 内部レジスタ定義ヘッダファイル      |
| non_existent_port_init.c                         | … 初期化時ポート設定処理          |
| non_existent_port_init.h                         | … 初期化時ポート設定処理ヘッダファイル   |
| resetprg.c                                       | … リセット・電源投入後起動処理       |
| sample.c   | … サンプルプログラムメイン処理       |
| sbrk.c   | … メモリ確保処理              |
| sbrk.h   | … メモリ確保ヘッダファイル         |
| sci.c  | … シリアル処理               |
| sci_main.c                                       | … シリアルメイン処理            |
| stacksct.h                                       | … スタック定義ヘッダファイル        |
| tmr.c  | … タイマ処理                |
| typedef.h  | … 型定義ヘッダファイル           |
| vect.h   | … 割り込みベクタテーブルヘッダファイル   |
| vecttbl.c  | … 割り込みベクタテーブル          |
| <¥Sample¥ap_rx63n_0a_sample_host_cs¥src¥CAN>     |                        |
| can_main.c                                       | … CAN メイン処理            |
| <¥Sample¥ap_rx63n_0a_sample_host_cs¥src¥ETHER>   |                        |
| ether_main.c                                     | … ネットワークメイン処理          |
| <¥Sample¥ap_rx63n_0a_sample_host_cs¥src¥USBHOST> |                        |
| usbhost_main.c                                   | … USB ホストメイン処理         |

### 3. 動作説明

#### 3.1 サンプルプログラムの動作

本サンプルプログラムは下記の動作を行いません。

- USB ホスト

USB ホストに USB メモリを接続すると、USB メモリにテキストファイルを書き込みます。

※ USB ホスト動作については後述の「3.5 USB ホスト動作」を参照してください。

- ネットワーク通信

Ethernet でエコーバックを行います。

※ ネットワーク動作については後述の「3.6 ネットワーク通信動作」を参照してください。

- シリアル通信

SCI0 でエコーバックを行いません。(送受信割り込み使用)

SCI0 から受信をした値を、そのまま SCI0 へ送信します。

シリアルの設定は、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、ホスト PC 上のターミナルソフト (ハイパーターミナル等) を使用して下さい。

- タイマ割り込み

LD2 (緑の LED) を 1000msec 間隔で点滅させます。(CMT1 割り込み使用)

また、CN1 の出力端子から方形波を出力します。

周期とピン番号は Table 「3.1-1 サンプルプログラム周期・ピン番号表」を参照してください。

- CAN 通信

CAN 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'10101010101、受信 ID:B'10101010100、スタンダードフォーマット、データフレーム、データ長 1 バイト、

通信速度 500kbps(TSEG1 = 6(7Tq), TSEG2 = 3(4Tq), SJW = 0(1Tq), BSP = 7)です。

CN1 方形波出力端子一覧

| コネクタ | ピン番号 | ピン名 | 周期     | 備考     |
|------|------|-----|--------|--------|
| CN1  | 51   | P54 | 20msec | CMT 使用 |
|      | 53   | P55 | 20msec | CMT 使用 |
|      | 57   | P62 | 10msec | MTU 使用 |
|      | 58   | P63 | 10msec | MTU 使用 |

Table 3.1-1 サンプルプログラム周期・ピン番号表

3.2 メモリマップ

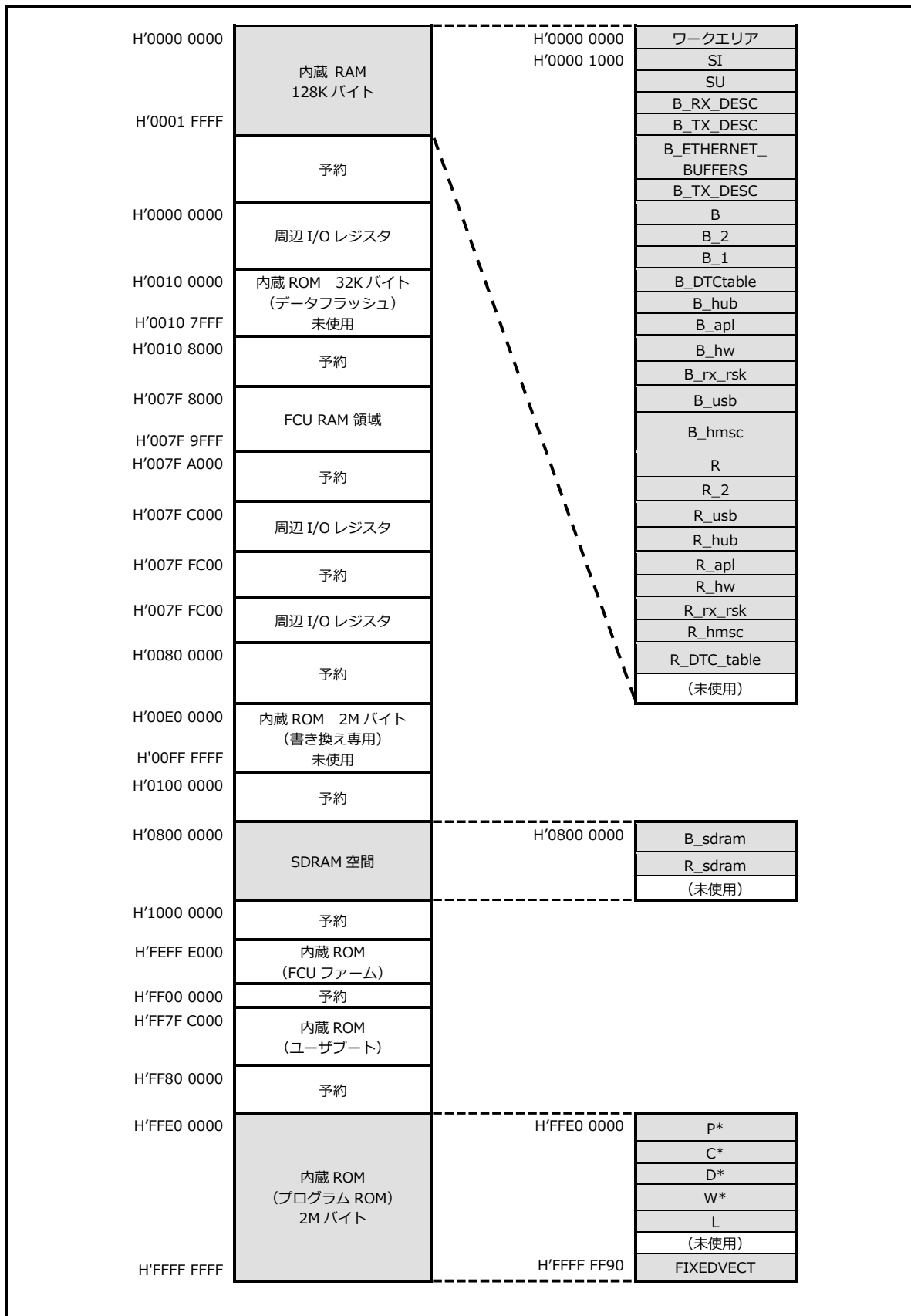


Fig 3.2-1 ネットワーク通信 + USB ホストサンプルプログラムメモリマップ

### 3.3 サンプルプログラムのダウンロード

サンプルプログラムを CPU ボード上で実行するためには、ビルドしたサンプルプログラムの実行ファイルを CPU ボードにダウンロードする必要があります。

サンプルプログラムのビルド方法および CPU ボードにサンプルプログラムをダウンロードする方法については、アプリケーションノート「**AN1526 RX 開発環境の使用法(CS+、Renesas Flash Programmer)**」に詳細な手順が記されていますので、参照してください。

### 3.4 サンプルプログラムの使用方法

サンプルプログラムを実行する際には、「¥Sample¥ap\_rx63n\_0a\_sample\_host\_cs¥src」にある sample.c において、使用する機能を選択してください。

sample.c では、define 定義を使用して処理の有効及び無効を決定しています。

有効にする機能の define 定義を 1 に設定し、使用しない機能は 0 に設定することで機能の選択を行います。

複数の機能を同時に動作させることはできないため、有効にする機能は必ず 1 つだけを選択してください。

### 3.5 USB ホスト動作

本サンプルプログラムはルネサス エレクトロニクス株式会社提供の FAT ファイルシステムドライバを移植しており、FAT12、FAT16、FAT32 にそれぞれ対応しています。

USB ホストサンプルでは FAT ファイルシステムを利用して、USB メモリにテキストファイルの書き込みを行います。

USB ホストサンプルを動作させる際には以下の手順に従い、USB デバイス接続時の動作を確認してください。

- ① CPU ボードに電源を投入し、USB ホストサンプルプログラムを起動します。
- ② CPU ボードの USB ホストポート(CN5)に USB メモリを挿入します。
- ③ USB メモリを挿入すると、USB メモリにテキストファイルが書き込まれます。
- ④ PC に USB メモリを挿入し、USB メモリのルートディレクトリに「hmscdemo.txt」という名前のテキストファイルが作成されていることを確認してください。
- ⑤ 「hmscdemo.txt」を開き、「a」が 512 バイト書き込まれていることを確認してください。
- ⑥ 以上で USB デバイス接続時の動作確認は終了です。

### 3.6 ネットワーク通信動作

USB ホストサンプルプログラムに実装されたネットワーク通信の確認に必要な推奨環境は以下の通りです。

|          |                             |
|----------|-----------------------------|
| ホスト PC   | PC/AT 互換機                   |
| OS       | Windows 10/11               |
| LAN ポート  | 10/100BASE-TX 以上対応の LAN ポート |
| LAN ケーブル | クロスケーブル                     |

#### 3.6.1 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-36-XX-XX<br>※ XX-XX の値は製品ごとに異なります。 |

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「¥Sample¥ap\_rx63n\_0a\_sample\_host\_cs¥src¥ETHER¥t4¥tcp\_blocking\_sample¥config\_tcpudp.c」で行われています。また、MAC アドレスは EEPROM の先頭 6byte に格納されています。

| アドレス          | 格納値  |
|---------------|------|
| 先頭アドレス + 0x00 | 0x00 |
| + 0x01        | 0x0C |
| + 0x02        | 0x7B |
| + 0x03        | 0x36 |
| + 0x04        | 0xXX |
| + 0x05        | 0xXX |

※ 0xXX の値は製品ごとに異なります

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となります。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、設定してください。



### 3.6.2 ネットワーク動作設定

以下の手順に従い、ネットワーク動作を確認してください。

- ① LAN クロスケーブルを使い CPU ボードの LAN コネクタ (CN3) とホスト PC を接続します。
- ② ホスト PC 上でネットワークの設定を行います。  
CPU ボードの設定に合わせるため、ホスト PC のネットワーク設定を下記の内容に変更してください。

|          |               |
|----------|---------------|
| 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 ホスト PC 上でネットワーク通信が可能なターミナルソフト (ハイパーターミナルなど) を起動し、TCP/IP 通信を行います。  
TCP/IP の設定は、IP 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 packets 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. 開発環境使用時の各設定値

開発環境を使用する際の、AP-RX63N-0A 固有の設定を以下に示します。

表内の「項目番号」はアプリケーションノート

「AN1526 RX 開発環境の使用方法(CS+、Renesas Flash Programmer)」内で示されている

項目番号を示していますので、対応したそれぞれの設定値を参照してください。

| ビルド・動作確認方法   |      |   |
|--------------|------|---|
| 項目名          | 項目番号 | 設定値   |
| 出力フォルダ       | 2-2  | Sample¥ap_rx63n_0a_usbhost_sample_cs¥DefaultBuild                                       |
| モトローラファイル名   | 2-3  | Sample¥ap_rx63n_0a_usbhost_sample_cs<br>¥DefaultBuild¥ap_rx63n_0a_usbhost_sample_cs.mot |
| アブソリュートファイル名 | 2-4  | Sample¥ap_rx63n_0a_usbhost_sample_cs<br>¥DefaultBuild¥ap_rx63n_0a_usbhost_sample_cs.abs |
| マップファイル      | 2-5  | Sample¥ap_rx63n_0a_usbhost_sample_cs<br>¥DefaultBuild¥ap_rx63n_0a_usbhost_sample_cs.map |

| Renesas Flash Programmer を使用した Flash 書き込み方法 (シリアルポート(SCI)を使用する方法) |      |   |
|---|------|---|
| 項目名   | 項目番号 | 設定値   |
| ボード設定 (Flash 書き込み)  | 3-1  | ボード : Fig 5-1 を参照      ケーブル接続 : CN6   |
| Flash に書き込むファイル   | 3-3  | Sample¥ap_rx63n_0a_usbhost_sample_cs<br>¥DefaultBuild¥ap_rx63n_0a_usbhost_sample_cs.mot |
| ボード設定 (動作)  | 3-4  | Fig 4-3 を参照   |

| Renesas Flash Programmer を使用した Flash 書き込み方法 (USB ブートモードを使用する方法) |      |  |
|---|------|--|
| 項目名   | 項目番号 | 設定値  |
| ボード設定 (Flash 書き込み)  | 3-5  | ボード : Fig 4-2 を参照      ケーブル接続 : CN4 (USB miniB)                                  |
| ツール選択   | 3-6  | [USB Direct]   |
| Flash に書き込むファイル   | 3-7  | ap_rx63n_0a_usbhost_sample_cs<br>¥DefaultBuild¥ap_rx63n_0a_usbhost_sample_cs.mot |
| ボード設定 (動作)  | 3-8  | Fig 4-3 を参照  |

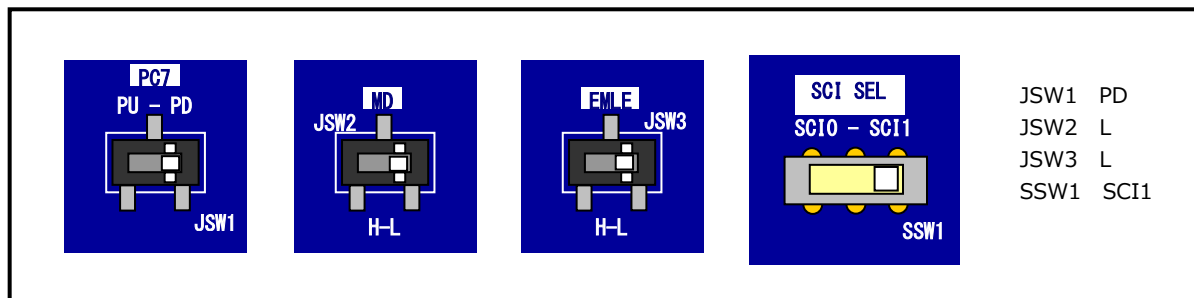


Fig 4-1 Flash 書き込み(シリアルポート使用)時のボード設定

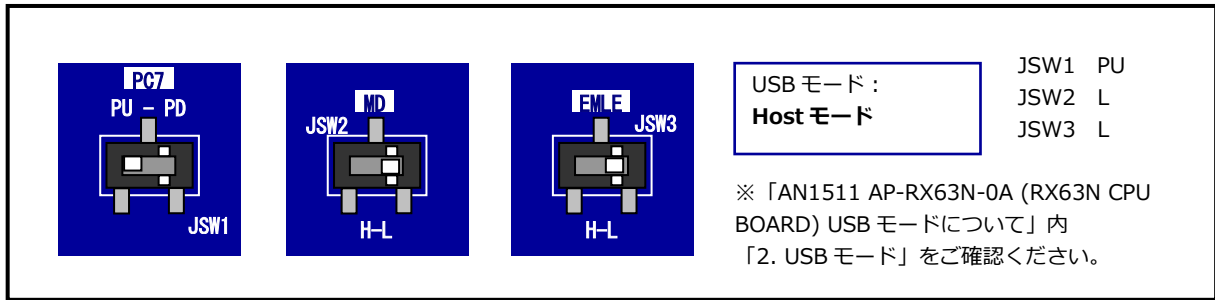


Fig 4-2 Flash書き込み(USBブートモード)時のボード設定

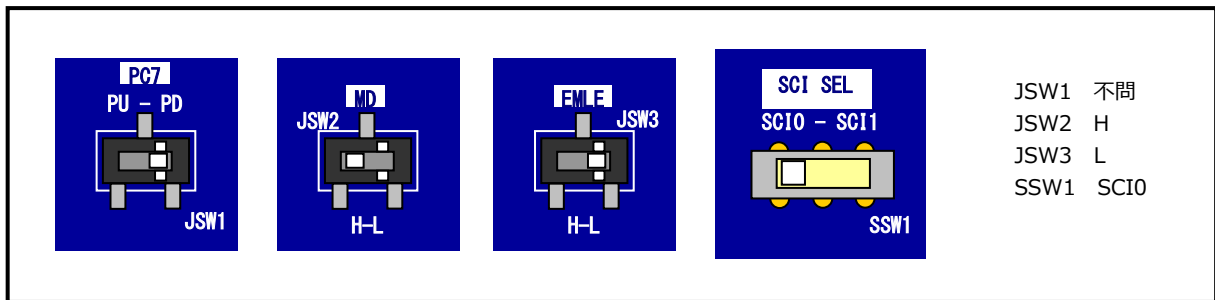


Fig 4-3 サンプルプログラム動作時のボード設定

| E1 エミュレータ/E2 エミュレータ Lite を使用したデバッグ方法 |      |  |
|--------------------------------------|------|--|
| 項目名                                  | 項目番号 | 設定値  |
| ボード設定                                | 4-1  | Fig 4-4 を参照  |
| JTAG クロック                            | 4-10 | E1 エミュレータを使用する場合 : 16.5(MHz)<br>E2 エミュレータ Lite を使用する場合 : 6.00(MHz) |
| EXTAL クロック                           | 4-11 | 12(MHz)  |

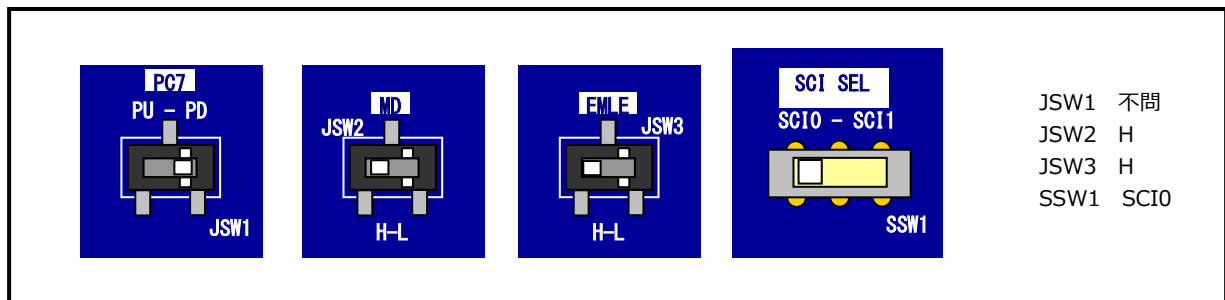


Fig 4-4 E1 エミュレータ/E2 エミュレータ Lite デバッグ時のボード設定

## 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。
- ・本サンプルプログラムで使用されているミドルウェアおよびドライバ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内容およびサンプルプログラムについてのサポートは一切受け付けておりません。
- ・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

- ・RX 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。
- ・CS+ 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。
- ・E1 エミュレータはルネサス 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。
- ・E2 エミュレータ Lite 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。
- ・Renesas Flash Programmer 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®10、Windows®11 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®11 は Windows 11 もしくは Win11
- ・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  
静岡県浜松市中央区積志町 834  
<https://www.apnet.co.jp>  
E-Mail: [query@apnet.co.jp](mailto:query@apnet.co.jp)