

PC-WiFi-01

サンプルプログラム解説

2011年08月10日

目次

1. 概要.....	1
1.1 概要.....	1
1.2 動作環境.....	2
1.3 動作モード.....	3
1.4 開発環境について.....	5
1.5 ワークスペースについて.....	5
2. サンプルプログラムの構成.....	6
2.1 フォルダ構成.....	6
2.2 ファイル構成.....	7
3. TCP/IP 通信サンプルプログラム.....	13
3.1 ビルド・デバッグ方法.....	13
3.2 動作説明（TCP/IP 通信）.....	16
3.2.1 サンプルプログラム概要（TCP/IP 通信 アドホックモード）.....	16
3.2.2 サンプルプログラム概要（TCP/IP 通信 インフラストラクチャモード）.....	18
3.2.3 TCP/IP 通信エコーバックサーバ動作.....	19
3.3 RAM 動作時のメモリマップ（TCP/IP 通信サンプルプログラム共通）.....	21
3.4 ROM 動作時のメモリマップ（TCP/IP 通信サンプルプログラム共通）.....	22
4. UDP 通信サンプルプログラム.....	23
4.1 ビルド・デバッグ方法（UDP 通信サンプルプログラム）.....	23
4.2 動作説明（UDP 通信）.....	26
4.2.1 サンプルプログラム概要（UDP 通信 アドホックモード）.....	26
4.2.2 サンプルプログラム概要（UDP 通信 インフラストラクチャモード）.....	28
4.2.3 UDP 通信エコーバックサーバ動作.....	29
4.3 RAM 動作時のメモリマップ（UDP 通信サンプルプログラム共通）.....	31
4.4 ROM 動作時のメモリマップ（UDP 通信サンプルプログラム共通）.....	32

5. PC-WIFI-01 制御方法	33
5.1 概要	33
5.2 シリアルインタフェース	34
5.2.1 シリアルインタフェース仕様	34
5.2.2 シリアルインタフェースフロー	34
5.3 AT コマンド	35
5.4 ボーレートの自動設定	36
5.4.1 オートボーレート	36
5.4.2 ボーレート設定コマンド	37
5.5 コマンドリファレンス	38
5.5.1 Band コマンド	38
5.5.2 Init コマンド	38
5.5.3 Scan コマンド	39
5.5.4 ネットワークタイプ設定コマンド	40
5.5.5 Pre Shared Key コマンド	40
5.5.6 Authentication Mode コマンド	41
5.5.7 Join コマンド	41
5.5.8 IP Parameter 設定コマンド	43
5.5.9 TCP/IP Client 設定コマンド	44
5.5.10 TCP/IP Server 設定コマンド	44
5.5.11 UCP Client 設定コマンド	45
5.5.12 UDP Server 設定コマンド	45
5.5.13 Send data コマンド	46
5.5.14 Receive data コマンド	47
5.5.15 Close Socket コマンド	48
5.5.16 Disassociate コマンド	48
5.5.17 ボーレート設定コマンド	49
5.5.18 Soft Reset コマンド	49

1. 概要

1.1 概要

本アプリケーションノートでは、PC-WiFi-01 に付属するサンプルプログラムについて解説します。

PC-WiFi-01 には、弊社製 AP-SH2A-0A 用サンプルプログラムが付属しています。

本サンプルプログラムの概要を以下に示します

サンプルプログラム	動作内容
TCP/IP 通信サンプルプログラム (アドホックモード クリエイト)	<ul style="list-style-type: none">・TCP/IP 通信エコーバックサーバ・シリアル通信・タイマ割り込み
TCP/IP 通信サンプルプログラム (アドホックモード ジョイン)	<ul style="list-style-type: none">・TCP/IP 通信エコーバックサーバ・シリアル通信・タイマ割り込み
TCP/IP 通信サンプルプログラム (インフラストラクチャモード)	<ul style="list-style-type: none">・TCP/IP 通信エコーバックサーバ・シリアル通信・タイマ割り込み
UDP 通信サンプルプログラム (アドホックモード クリエイト)	<ul style="list-style-type: none">・UDP 通信エコーバックサーバ・シリアル通信・タイマ割り込み
UDP 通信サンプルプログラム (アドホックモード ジョイン)	<ul style="list-style-type: none">・UDP 通信エコーバックサーバ・シリアル通信・タイマ割り込み
UDP 通信サンプルプログラム (インフラストラクチャモード)	<ul style="list-style-type: none">・UDP 通信エコーバックサーバ・シリアル通信・タイマ割り込み

1.2 動作環境

各サンプルプログラムの動作確認に必要な機器を以下に記します。

サンプルプログラム	動作確認に必要な機器
TCP/IP 通信サンプルプログラム (アドホックモード クリエイト)	・アドホック通信可能なホスト PC
TCP/IP 通信サンプルプログラム (アドホックモード ジョイン)	・アドホック通信可能なホスト PC
UDP 通信サンプルプログラム (アドホックモード クリエイト)	・アドホック通信可能なホスト PC
UDP 通信サンプルプログラム (アドホックモード ジョイン)	・アドホック通信可能なホスト PC
TCP/IP 通信サンプルプログラム (インフラストラクチャモード)	・ネットワーク通信可能なホスト PC
UDP 通信サンプルプログラム (インフラストラクチャモード)	・ネットワーク通信可能なホスト PC

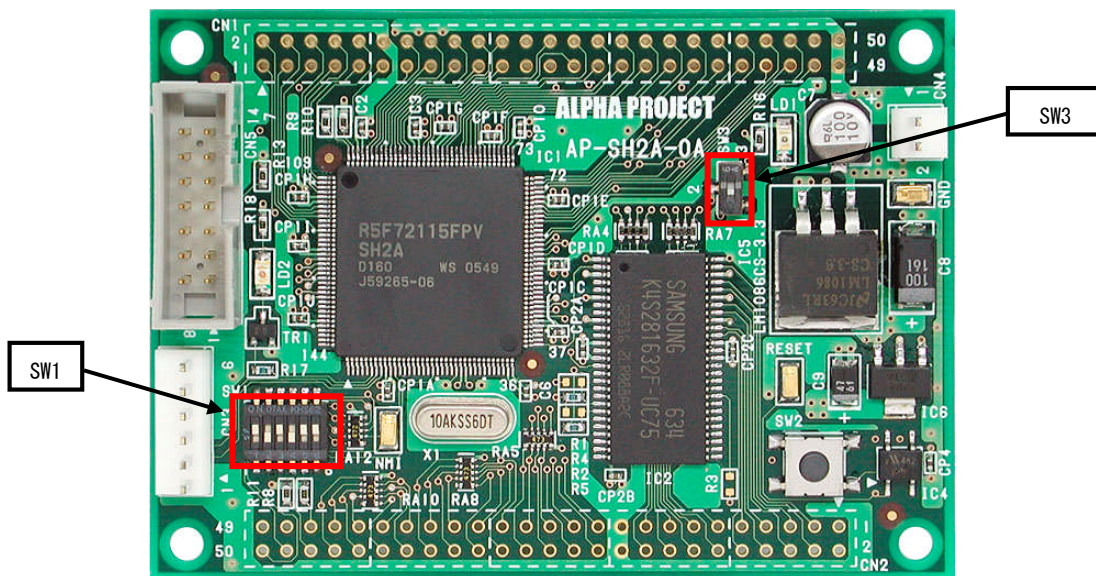
1.3 動作モード

本サンプルプログラムは、AP-SH2A-0A 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-0A 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。

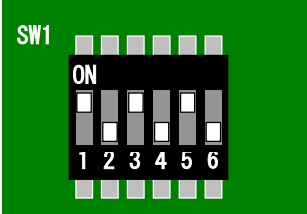
CPU 動作モード	:	モード 2
デバッグモード	:	OFF
クロックモード	:	モード 6
外付け SDRAM 設定	:	外付け SDRAM を使用する



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。

プログラム動作時

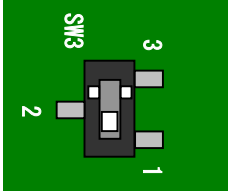
・ SW1



〈SW1 設定〉

クロックモード	: モード6
CPU 動作モード	: モード2
デバッグモード	: OFF
FLASH 書き込み	: 禁止

・ SW3

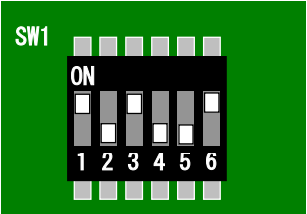


〈SW3 設定〉

ボード上の SDRAM	: 使用する
-------------	--------

プログラム書き込み時

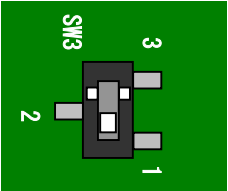
・ SW1



〈SW1 設定〉

クロックモード	: モード6
CPU 動作モード	: モード2
デバッグモード	: ON
FLASH 書き込み	: 許可

・ SW3



〈SW3 設定〉

ボード上の SDRAM	: 使用する
-------------	--------

Fig 1.3-1 動作モード設定

1.4 開発環境について

本サンプルプログラムは総合開発環境 High-performance Embedded Workshop（以下 HEW と表記します）を用いて開発されています。

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。

開発環境	バージョン	コンパイラ名	バージョン	備考
High-performance Embedded Workshop	V 3.01 (release 1) 以降	SHC ※1	V9.0.0.0 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++コンパイラパッケージ」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。

1.5 ワークスペースについて

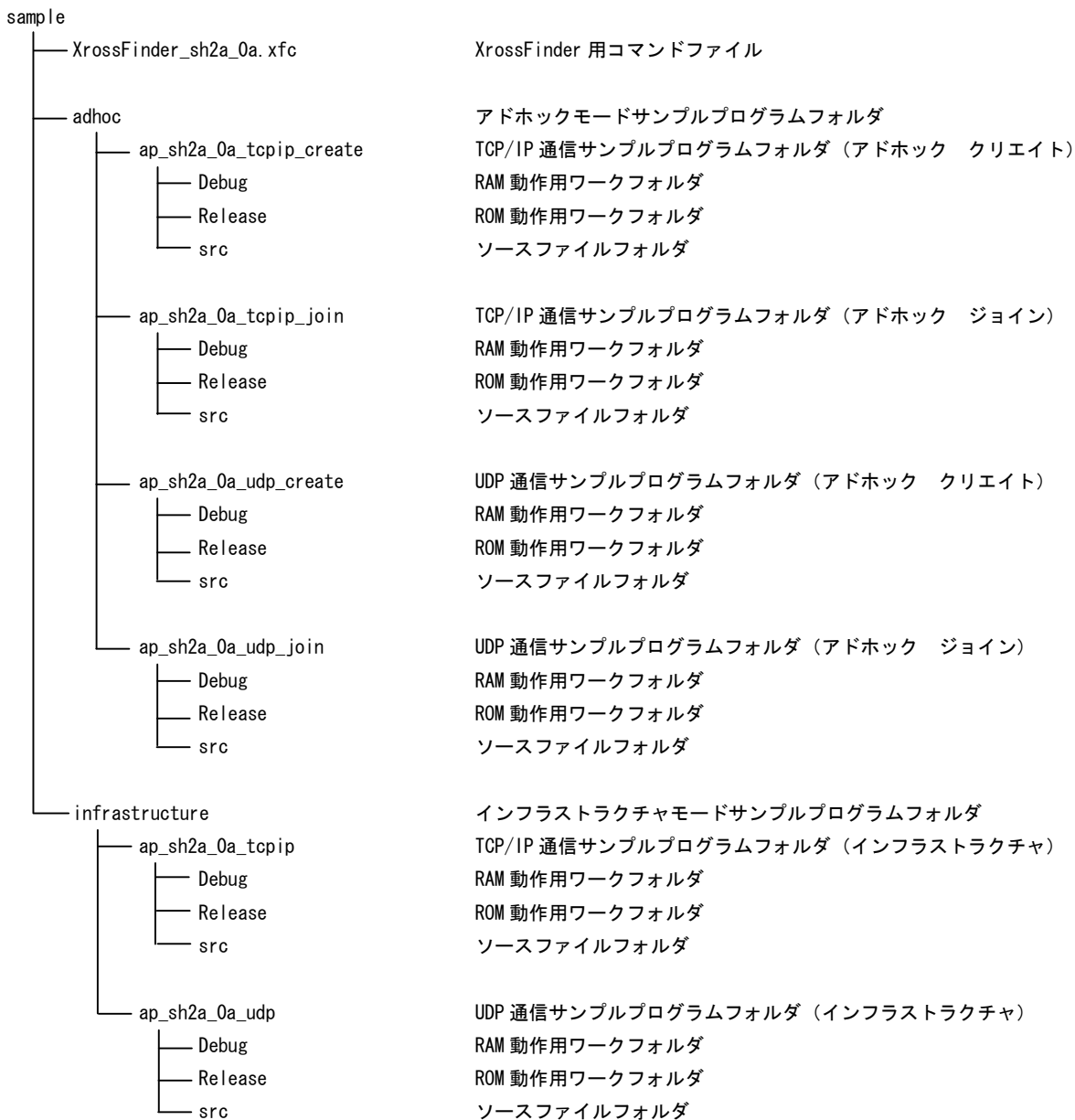
本サンプルプログラムの総合開発環境 High-performance Embedded Workshop ワークスペースは次のフォルダに格納されています。

サンプルプログラム	フォルダ
TCP/IP サンプルプログラム (アドホックモード クリエイト)	¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create.hws
TCP/IP サンプルプログラム (アドホックモード ジョイン)	¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join.hws
UDP サンプルプログラム (アドホックモード クリエイト)	¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create.hws
UDP サンプルプログラム (アドホックモード ジョイン)	¥sample¥adhoc¥ap_sh2a_0a_udp_join¥ap_sh2a_0a_udp_join.hws
TCP/IP サンプルプログラム (インフラストラクチャモード)	¥sample¥infrastructure¥ap_sh2a_0a_tcpip¥ap_sh2a_0a_tcpip.hws
UDP サンプルプログラム (インフラストラクチャモード)	¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp.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

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。

<¥sample フォルダ内>

adhoc	...	アドホックモードサンプルプログラム
infrastructure	...	インフラストラクチャモードサンプルプログラム
XrossFinder_sh2a_0a.xfc	...	XrossFinder 用コマンドファイル

<¥sample¥adhoc フォルダ内>

ap_sh2a_0a_tcpip_create	...	TCP/IP 通信サンプルプログラムフォルダ (アドホック クリエイト)
ap_sh2a_0a_tcpip_join	...	TCP/IP 通信サンプルプログラムフォルダ (アドホック ジョイン)
ap_sh2a_0a_udp_create	...	UDP 通信サンプルプログラムフォルダ (アドホック クリエイト)
ap_sh2a_0a_udp_join	...	UDP 通信サンプルプログラムフォルダ (アドホック ジョイン)

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_create フォルダ内>

ap_sh2a_0a_tcpip_create.hws	...	TCP/IP 通信サンプルプログラム HEW 用ワークスペース ファイル (アドホック クリエイト)
-----------------------------	-----	---

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create フォルダ内>

ap_sh2a_0a_tcpip_create.hwp	...	TCP/IP 通信サンプルプログラム HEW 用プロジェクト ファイル (アドホック ジョイン)
-----------------------------	-----	---

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create¥Debug フォルダ内>

ap_sh2a_0a_tcpip_create.abs	...	TCP/IP 通信サンプルプログラム RAM 動作用オブジェクト ファイル (アドホック クリエイト) (elf 形式)
ap_sh2a_0a_tcpip_create.mot	...	TCP/IP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル (アドホック クリエイト)
ap_sh2a_0a_tcpip_create.map	...	TCP/IP 通信サンプルプログラム RAM 動作用マップファイル (アドホック クリエイト) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create¥Release フォルダ内>

ap_sh2a_0a_tcpip_create.abs	...	TCP/IP 通信サンプルプログラム ROM 動作用オブジェクト ファイル (アドホック クリエイト) (elf 形式)
ap_sh2a_0a_tcpip_create.mot	...	TCP/IP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル (アドホック クリエイト)
ap_sh2a_0a_tcpip_create.map	...	TCP/IP 通信サンプルプログラム ROM 動作用マップファイル (アドホック クリエイト) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバファイル
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_join フォルダ内>

ap_sh2a_0a_tcpip_join.hws	...	TCP/IP 通信サンプルプログラム HEW 用ワークスペース ファイル (アドホック ジョイン)
---------------------------	-----	--

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join フォルダ内>

ap_sh2a_0a_tcpip_join.hwp	...	TCP/IP 通信サンプルプログラム HEW 用プロジェクト ファイル (アドホック ジョイン)
---------------------------	-----	---

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join¥Debug フォルダ内>

ap_sh2a_0a_tcpip_join.abs	...	TCP/IP 通信サンプルプログラム RAM 動作用オブジェクト ファイル (アドホック ジョイン) (elf 形式)
ap_sh2a_0a_tcpip_join.mot	...	TCP/IP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル (アドホック ジョイン)
ap_sh2a_0a_tcpip_join.map	...	TCP/IP 通信サンプルプログラム RAM 動作用マップファイル (アドホック ジョイン) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join¥Release フォルダ内>

ap_sh2a_0a_tcpip_join.abs	...	TCP/IP 通信サンプルプログラム ROM 動作用オブジェクト ファイル (アドホック ジョイン) (elf 形式)
ap_sh2a_0a_tcpip_join.mot	...	TCP/IP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル (アドホック ジョイン)
ap_sh2a_0a_tcpip_join.map	...	TCP/IP 通信サンプルプログラム ROM 動作用マップファイル (アドホック ジョイン) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバファイル
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

<¥sample¥adhoc¥ap_sh2a_0a_udp_create フォルダ内>

ap_sh2a_0a_udp_create.hws	...	UDP 通信サンプルプログラム HEW 用ワークスペースファイル (アドホック クリエイト)
---------------------------	-----	---

<¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create フォルダ内>

ap_sh2a_0a_udp_create.hwp	...	UDP 通信サンプルプログラム HEW 用プロジェクトファイル (アドホック クリエイト)
---------------------------	-----	--

<¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create¥Debug フォルダ内>

ap_sh2a_0a_udp_create.abs	...	UDP 通信サンプルプログラム RAM 動作用オブジェクト ファイル (アドホック クリエイト) (elf 形式)
ap_sh2a_0a_udp_create.mot	...	UDP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル (アドホック クリエイト)
ap_sh2a_0a_udp_create.map	...	UDP 通信サンプルプログラム RAM 動作用マップファイル (アドホック クリエイト) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create¥Release フォルダ内>

ap_sh2a_0a_udp_create.abs	...	UDP 通信サンプルプログラム ROM 動作用オブジェクト ファイル (アドホック クリエイト) (elf 形式)
ap_sh2a_0a_udp_create.mot	...	UDP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル (アドホック クリエイト)
ap_sh2a_0a_udp_create.map	...	UDP 通信サンプルプログラム ROM 動作用マップファイル (アドホック クリエイト) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバ
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

<¥sample¥adhoc¥ap_sh2a_0a_udp_join フォルダ内>

ap_sh2a_0a_udp_join.hws	...	UDP 通信サンプルプログラム HEW 用ワークスペースファイル (アドホック ジョイン)
-------------------------	-----	--

<¥sample¥adhoc¥ap_sh2a_0a_udp_join¥ap_sh2a_0a_udp_join フォルダ内>

ap_sh2a_0a_udp_join.hwp	...	UDP 通信サンプルプログラム HEW 用プロジェクトファイル (アドホック ジョイン)
-------------------------	-----	---

<¥sample¥adhoc¥ap_sh2a_0a_udp_join¥ap_sh2a_0a_udp_join¥Debug フォルダ内>

ap_sh2a_0a_udp_join.abs	...	UDP 通信サンプルプログラム RAM 動作用オブジェクト ファイル (アドホック ジョイン) (elf 形式)
ap_sh2a_0a_udp_join.mot	...	UDP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル (アドホック ジョイン)
ap_sh2a_0a_udp_join.map	...	UDP 通信サンプルプログラム RAM 動作用マップファイル (アドホック ジョイン) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_udp_join ¥ap_sh2a_0a_udp_join¥Release フォルダ内>

ap_sh2a_0a_udp_join.abs	...	UDP 通信サンプルプログラム ROM 動作用オブジェクト ファイル (アドホック ジョイン) (elf 形式)
ap_sh2a_0a_udp_join.mot	...	UDP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル (アドホック ジョイン)
ap_sh2a_0a_udp_join.map	...	UDP 通信サンプルプログラム ROM 動作用マップファイル (アドホック ジョイン) コンパイル後は、.obj, .lib 等のファイルが生成されます

<¥sample¥adhoc¥ap_sh2a_0a_udp_join¥ap_sh2a_0a_udp_join¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバファイル
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

<¥sample¥infrastructure フォルダ内>

ap_sh2a_0a_tcpipserver	...	TCP/IP 通信サンプルプログラムフォルダ
ap_sh2a_0a_udp	...	UDP 通信サンプルプログラムフォルダ

<¥sample¥infrastructure¥ap_sh2a_0a_tcpip フォルダ内>

ap_sh2a_0a_tcpip.hws	...	TCP/IP 通信サンプルプログラム HEW 用 ワークスペースファイル
----------------------	-----	---

<¥sample¥infrastructure¥ap_sh2a_0a_tcpipserver¥ap_sh2a_0a_tcpipserver フォルダ内>

ap_sh2a_0a_tcpip.hwp	...	TCP/IP 通信サンプルプログラム HEW 用 プロジェクトファイル
----------------------	-----	--

<¥sample¥infrastructure¥ap_sh2a_0a_tcpip¥ap_sh2a_0a_tcpip¥Debug フォルダ内>

ap_sh2a_0a_tcpip.abs	...	TCP/IP サンプルプログラム通信 RAM 動作用オブジェクト ファイル(elf 形式)
ap_sh2a_0a_tcpip.mot	...	TCP/IP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル
ap_sh2a_0a_tcpip.map	...	TCP/IP 通信サンプルプログラム 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

<¥sample¥infrastructure¥ap_sh2a_0a_tcpip¥ap_sh2a_0a_tcpip¥Release フォルダ内>

ap_sh2a_0a_tcpip.abs	...	TCP/IP 通信サンプルプログラム ROM 動作用オブジェクト ファイル(elf 形式)
ap_sh2a_0a_tcpip.mot	...	TCP/IP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル
ap_sh2a_0a_tcpip.map	...	TCP/IP 通信サンプルプログラム 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

<¥sample¥infrastructure¥ap_sh2a_0a_tcpip¥ap_sh2a_0a_tcpip¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバファイル
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

<¥sample¥infrastructure¥ap_sh2a_0a_udp フォルダ内>

ap_sh2a_0a_udp.hws	...	UDP 通信サンプルプログラム HEW 用ワークスペースファイル
--------------------	-----	----------------------------------

<¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp フォルダ内>

ap_sh2a_0a_udp.hwp	...	UDP 通信サンプルプログラム HEW 用プロジェクトファイル
--------------------	-----	---------------------------------

<¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp¥Debug フォルダ内>

ap_sh2a_0a_udp.abs	...	UDP 通信サンプルプログラム RAM 動作用オブジェクト ファイル(elf 形式)
ap_sh2a_0a_udp.mot	...	UDP 通信サンプルプログラム RAM 動作用 モトローラ S フォーマット形式ファイル
ap_sh2a_0a_udp.map	...	UDP 通信サンプルプログラム 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

<¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp¥Release フォルダ内>

ap_sh2a_0a_udp.abs	...	UDP 通信サンプルプログラム ROM 動作用オブジェクト ファイル (elf 形式)
ap_sh2a_0a_udp.mot	...	UDP 通信サンプルプログラム ROM 動作用 モトローラ S フォーマット形式ファイル
ap_sh2a_0a_udp.map	...	UDP 通信サンプルプログラム 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

<¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp¥src フォルダ内>

7211.h	...	SH7211 内部レジスタ定義ヘッダファイル
boot.c	...	CPU 初期化処理ファイル
BoardDepend.h	...	ボード依存ファイル
common.h	...	共通ヘッダファイル
main.c	...	メイン処理ファイル
NetworkSetting.h	...	ネットワーク設定定義ファイル
sci.c	...	シリアル処理ファイル
section.src	...	セクション定義ファイル
timer.c	...	タイマ処理ファイル
vector.c	...	割込ベクタテーブルファイル
wifi.c	...	WiFi モジュールサンプルドライバファイル
wifi.h	...	WiFi モジュールサンプルドライバヘッダファイル

3. TCP/IP 通信サンプルプログラム

3.1 ビルド・デバッグ方法

TCP/IP 通信サンプルプログラムのビルド・デバッグ方法を以下に記します。

アドホックモード（クリエイト・ジョイン）、インフラストラクチャモードを問わず、ビルド・デバッグの方法は同一です。

(1) ビルド

- ① HEW を起動し、「Table 3.1-1 TCP/IP 通信サンプルプログラム HWS 一覧」からビルド・デバッグを行うサンプルプログラムの HWS ファイルを読み込みます。

サンプルプログラムの種類	読み込むファイル
アドホックモード クリエイト	¥sample¥adhoc¥ap_sh2a_0a_tcpip_create¥ap_sh2a_0a_tcpip_create.hws
アドホックモード ジョイン	¥sample¥adhoc¥ap_sh2a_0a_tcpip_join¥ap_sh2a_0a_tcpip_join.hws
インフラストラクチャモード	¥sample¥infrastructure¥ap_sh2a_0a_tcpip¥ap_sh2a_0a_tcpip.hws

Table 3.1-1 TCP/IP 通信サンプルプログラム HWS ファイル一覧

- 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 [ビルド]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 [ビルド] - [ビルド] を実行して下さい。モトローラファイル（拡張子が mot のファイル）、アプソリュートファイル（拡張子が abs のファイル）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下さい。

(2) RAM 上でのデバッグ

- ① XsSight を起動し、¥sample フォルダ直下にある XrossFinder_sh2a_0a.xfc コマンドファイルを読み込みます。
- ② 「(1) ビルド」で出力したアプソリュートファイルを XsSight から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 SP-SH2A-0A のスイッチを、「1.3 動作モード」を参考に設定します。
¥sample フォルダ内の XrossFinder_sh2a_0a.xfc と「(1) ビルド」で出力したアプソリュートファイルを XsSight で読み込みます。
- ② XsSight のメニューから FlashWriterEX を選択し、下図 Fig3.1-1 のように設定を行ってください。
- ③ 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。

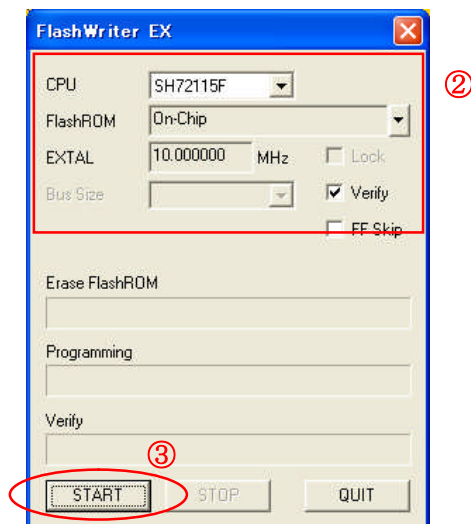


Fig3.1-1 FlashWriterEX for XsSight の設定

(4) XsSight 未使用時の確認方法

・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-0A のスイッチを、「1.3 動作モード」 「Fig1. 3-1 動作モード設定」を参考に設定します。
- ③ FlashWriterEX を起動して、「Table3. 1-2 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_0a.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 フォルダ内のモトローラファイルをボードに書き込みます。

FlashWriterEX の使用方法の詳細につきましては、FlashWriterEX 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 使用時のみ)	5MHz 以下
CPU	SH72115F
EXTAL	10. 0MHz
FLASHROM	0n-Chip

Table3. 1-2 FlashWriterEX の設定

3.2 動作説明 (TCP/IP 通信)

3.2.1 サンプルプログラム概要 (TCP/IP 通信 アドホックモード)

TCP/IP 通信サンプルプログラム (アドホックモード) は、下記の動作を行います。

1) クリエイト

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、TCP/IP エコーバックサーバを構築します。その後、受信したデータをそのまま送信元に送信します。
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など) を使用して行ってください。
※ TCP/IP エコーバックサーバ動作の詳細は、「3.2.3 TCP/IP 通信エコーバックサーバ動作」を参照してください。

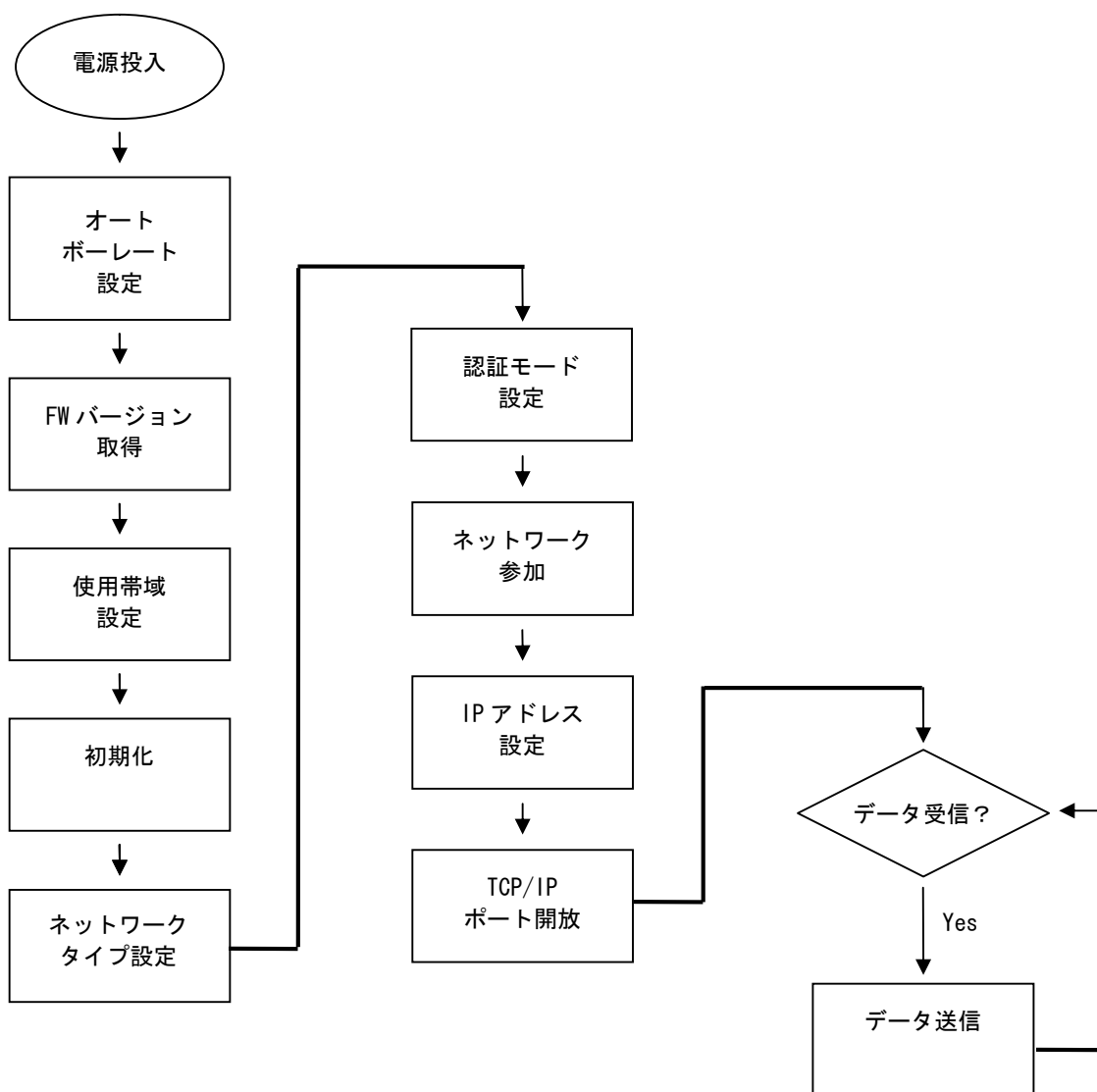


Fig 3.2-1 TCP/IP 通信サンプルプログラム アドホッククリエイト PC-WiFi-01 制御フロー

2) ジョイン

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、TCP/IP エコーバックサーバを構築します。その後、受信したデータをそのまま送信元に送信します。
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など）を使用して行ってください。
※ TCP/IP エコーバックサーバ動作の詳細は、「3.2.3 TCP/IP 通信エコーバックサーバ動作」を参照してください。

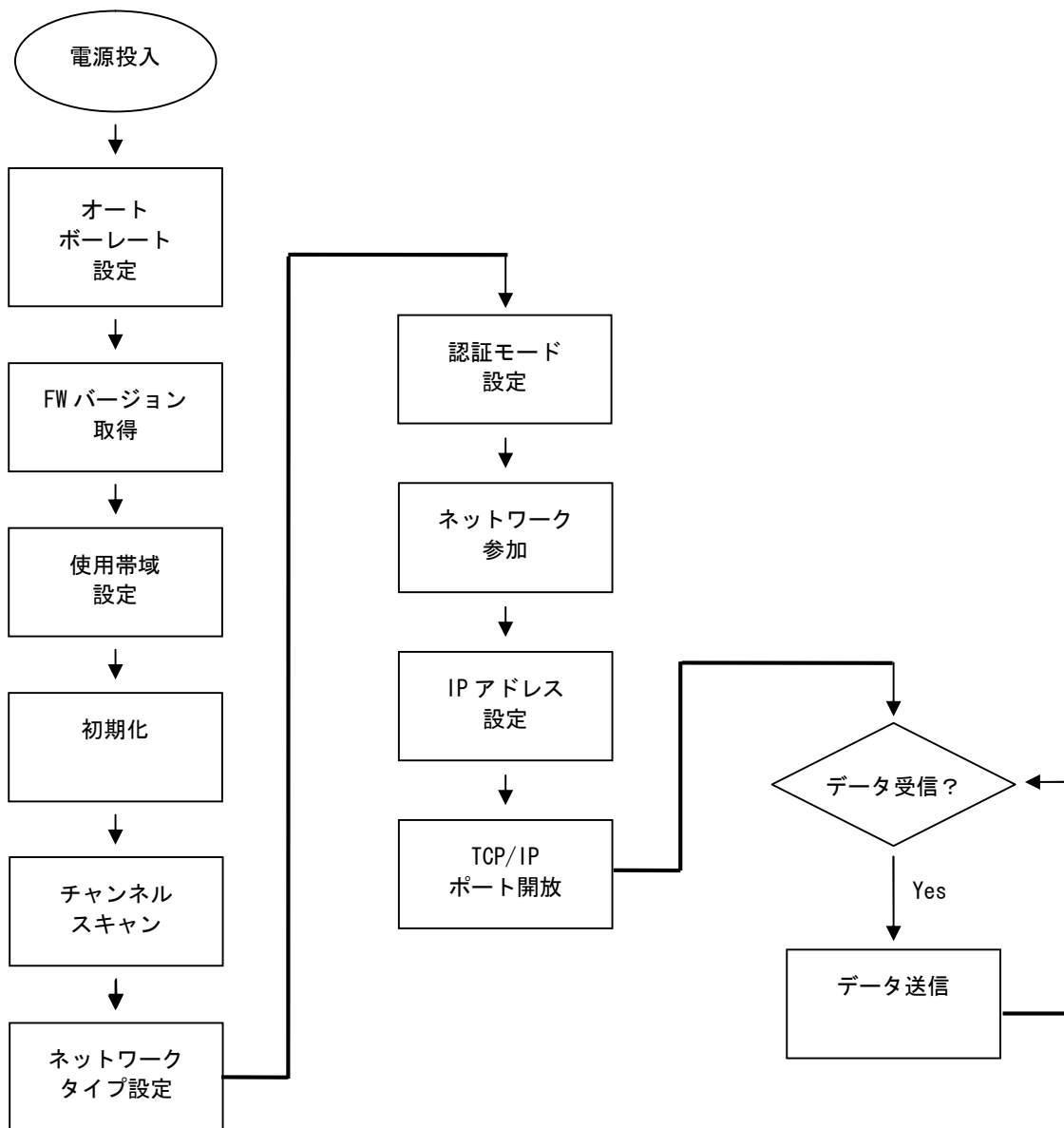


Fig 3.2-1 TCP/IP 通信サンプルプログラム アドホッククリエイト PC-WiFi-01 制御フロー

3.2.2 サンプルプログラム概要 (TCP/IP 通信 インフラストラクチャモード)

TCP/IP 通信サンプルプログラム (インフラストラクチャモード) は、下記の動作を行います。

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、インフラストラクチャモードでアクセスポイントに接続した後、TCP/IP エコーバックサーバを構築します。
その後、受信したデータをそのまま送信元に送信します。
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など) を使用して行ってください。
※ TCP/IP エコーバックサーバ動作の詳細は、「3.2.3 TCP/IP 通信エコーバックサーバ動作」を参照してください。

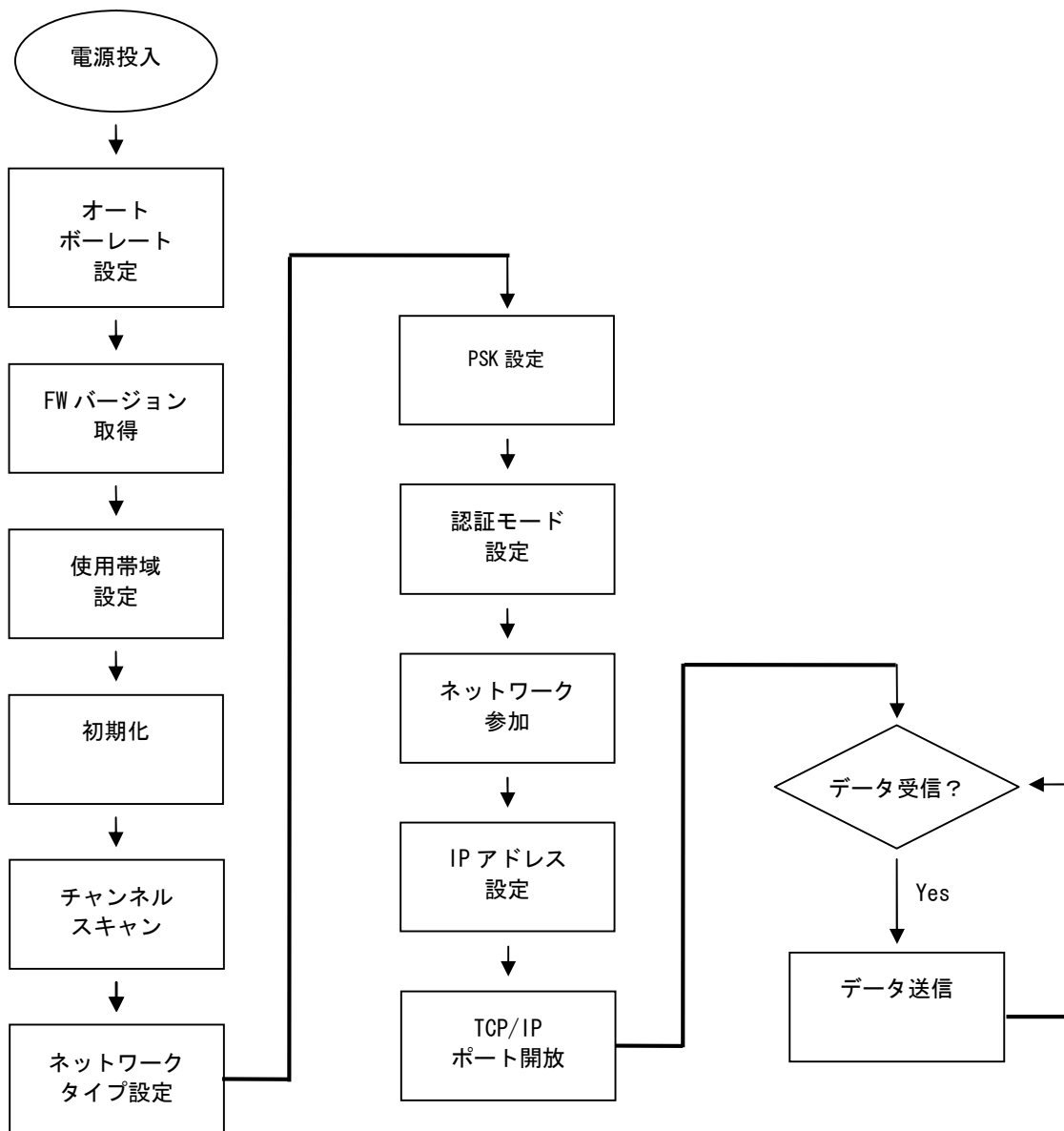


Fig 3.2-3 TCP/IP 通信サンプルプログラム インフラストラクチャ PC-WiFi-01 制御フロー

3.2.3 TCP/IP 通信エコーバックサーバ動作

(1) TCP/IP ネットワーク設定

以下にサンプルプログラムのネットワーク設定を記します。※

TCP/IP ネットワーク設定 (アドホックモード)	
使用帯域	2.4GHz
使用チャンネル	1ch
ネットワーク接続	アドホックモード
送信レート	自動設定
送信レベル	ロウレベルモード
PSK	-
アクセスポイント	PC-WiFi-01Sample
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	0.0.0.0
使用するポート	8001

Table3.2-1 TCP/IP 通信ネットワーク設定 (アドホックモード)

TCP/IP ネットワーク設定 (インフラストラクチャモード)	
使用帯域	2.4GHz
使用チャンネル	1ch
ネットワーク接続	インフラストラクチャモード
送信レート	自動設定
送信レベル	ロウレベルモード
PSK	PC-WiFi-01SamplePSK
アクセスポイント	PC-WiFi-01Sample
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	0.0.0.0
使用するポート	8001

Table3.2-2 TCP/IP 通信ネットワーク設定 (インフラストラクチャモード)

- ※ これらの設定は弊社の環境において設定した値となっています。ご利用時は、お使いの環境のネットワーク管理者に問い合わせ、ご利用になられる環境に沿ったそれぞれ適切な値を設定しビルドしてください。
- ※ これらの設定値は、各サンプルプログラムのソースフォルダ内にある「NetworkSetting.h」にて定義されています。

(2) TCP/IP 通信エコーバックサーバ動作 (アドホックモード クリエイト)

以下の手順に従い、TCP/IP 通信エコーバックサーバの動作を確認してください。

- ① 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。
- ② CPU ボードの LD2 が点灯したことを確認し、アドホック通信機器の設定を行います。
その際、使用する設定は「Table 3.2-1 TCP/IP 通信ネットワーク設定 (アドホックモード)」で設定した値となります。
- ③ アドホック機器を無線 LAN ネットワークに接続し、エコーバックが行われることを確認してください。
- ④ 以上で TCP/IP 通信エコーバックサーバ動作 (アドホックモード: クリエイト) の動作は終了です。

(3) TCP/IP 通信エコーバックサーバ動作 (アドホックモード ジョイン)

以下の手順に従い、TCP/IP 通信エコーバックサーバの動作を確認してください。

- ① アドホック通信機器の設定を行い、無線 LAN ネットワークに接続します。
その際、使用する設定は「Table 3.2-1 TCP/IP 通信ネットワーク設定 (アドホックモード)」で設定した値となります。
- 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 CPU ボードの LD2 が点灯したことを確認し、アドホック通信機器の側からエコーバックが行われることを確認してください。
- ④ 以上で TCP/IP 通信エコーバックサーバ動作 (アドホックモード クリエイト) の動作は終了です。

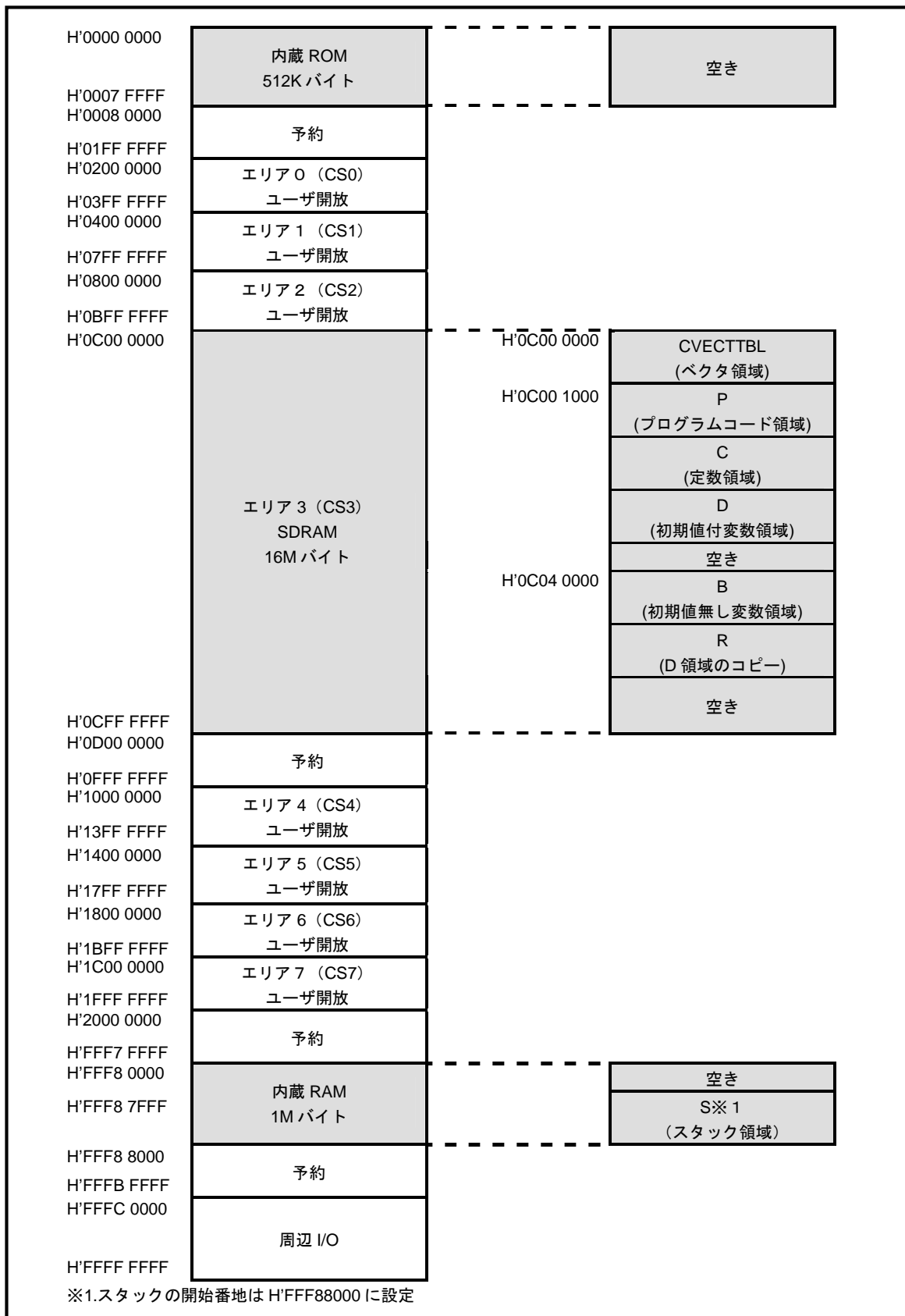
(4) TCP/IP 通信エコーバックサーバ動作 (インフラストラクチャモード)

以下の手順に従い、TCP/IP 通信エコーバックサーバの動作を確認してください。

- ① 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。
- ② ホスト PC 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、Ethernet 接続の設定を行います。
その際、使用する設定は「3.2-2 TCP/IP 通信ネットワーク設定 (インフラストラクチャモード)」で設定した値となります。
- ③ 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。
- ④ 以上で TCP/IP 通信エコーバックサーバ動作 (インフラストラクチャモード) の動作は終了です。

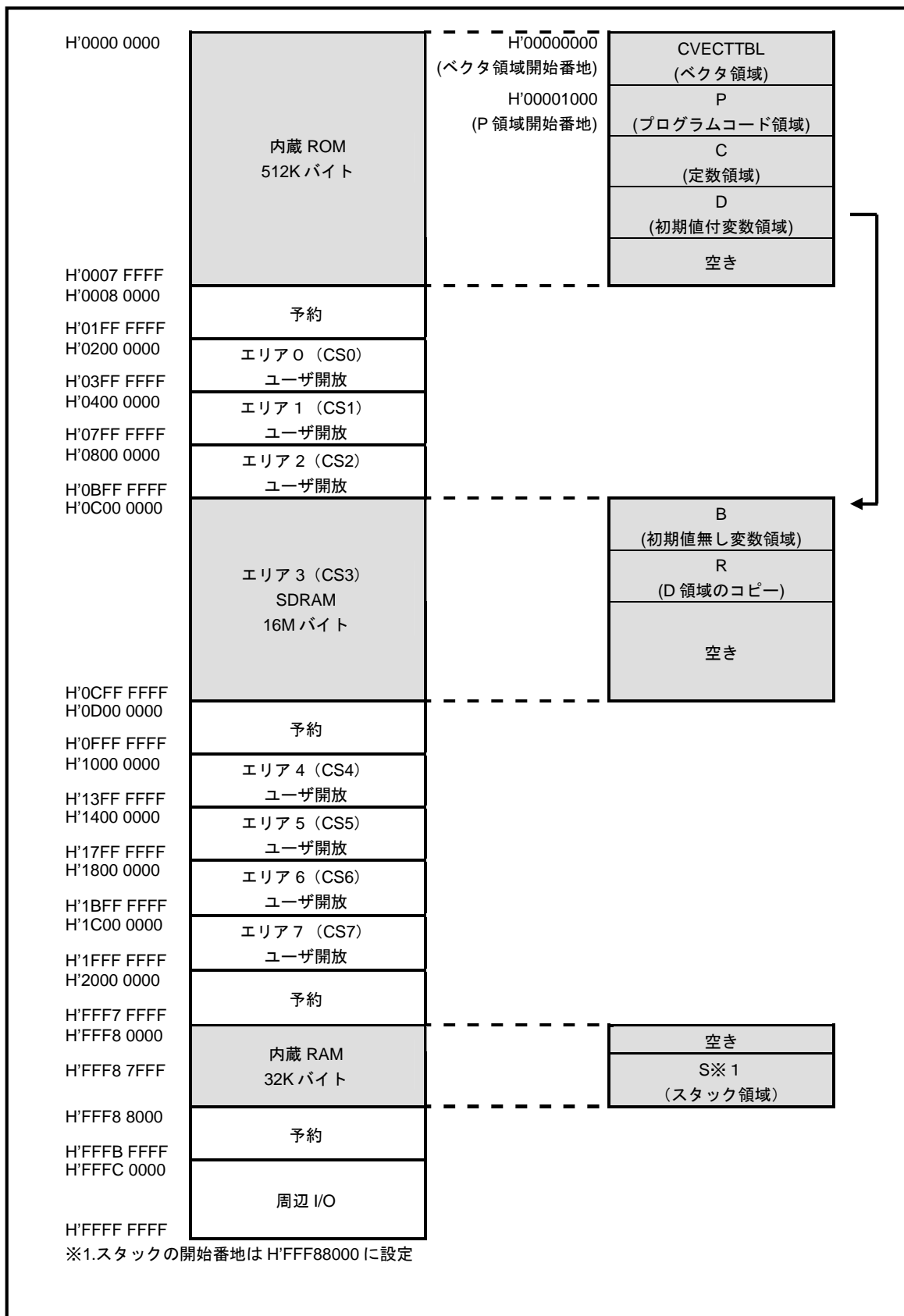
3.3 RAM 動作時のメモリマップ (TCP/IP 通信サンプルプログラム共通)

メモリマップを以下に示します。



3.4 ROM 動作時のメモリマップ (TCP/IP 通信サンプルプログラム共通)

メモリマップを以下に示します。



4. UDP 通信サンプルプログラム

4.1 ビルド・デバッグ方法（UDP 通信サンプルプログラム）

UDP 通信サンプルプログラムのビルド・デバッグ方法を以下に記します。

アドホックモード、インフラストラクチャモードを問わず、ビルド・デバッグの方法は同一です。

(1) ビルド

- ① HEW を起動し、「Table 4.1-1 UDP 通信サンプルプログラム HWS ファイル一覧」からビルド・デバッグを行うサンプルプログラムの HWS ファイルを読み込みます。

サンプルプログラムの種類	読み込むファイル
アドホックモード クリエイト	¥sample¥adhoc¥ap_sh2a_0a_udp_create¥ap_sh2a_0a_udp_create.hws
アドホックモード ジョイン	¥sample¥adhoc¥ap_sh2a_0a_udp_join¥ap_sh2a_0a_udp_join.hws
インフラストラクチャモード	¥sample¥infrastructure¥ap_sh2a_0a_udp¥ap_sh2a_0a_udp.hws

Table 4.1-1 UDP 通信サンプルプログラム HWS ファイル一覧

- 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 [ビルド] 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 [ビルド] - [ビルド] を実行して下さい。モトローラファイル (拡張子が mot のファイル)、アブソリュートファイル (拡張子が abs のファイル) 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下さい。

(2) RAM 上でのデバッグ

- ① XsSight を起動し、¥sample フォルダ直下にある XrossFinder_sh2a_0a.xfc コマンドファイルを読み込みます。
- ② 「(1) ビルド」で出力したアブソリュートファイルを XsSight から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 SP-SH2A-0A のスイッチを、「1.3 動作モード」を参考に設定します。
- ② ¥sample フォルダ内の XrossFinder_sh2a_0a.xfc と「(1) ビルド」で出力したアプソリュートファイルを XsSight で読み込みます。
- ③ XsSight のメニューから FlashWriterEX を選択し、下図 Fig4.1-1 のように設定を行ってください。
- ④ 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。

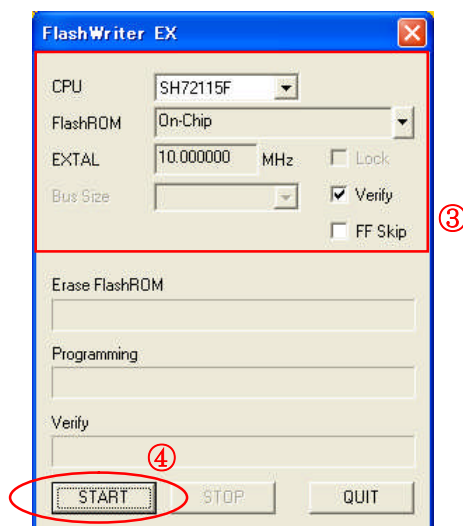


Fig4.1-1 FlashWriterEX for XsSight の設定

(4) XsSight 未使用時の確認方法

・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 PC-WiFi-01 のスイッチを、「1.3 動作モード」 「Fig1.3-1 動作モード設定」を参考に設定します。
- ③ FlashWriterEX を起動して、「Table4.1-2 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_0a.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 フォルダ内のモトローラファイルをボードに書き込みます。
- ⑥ PC-WiFi-01 のスイッチを、「1.3 動作モード」 「Fig1.3-1 動作モード設定」を参考に設定します。
- ⑦ FlashWriterEX の使用方法の詳細につきましては、FlashWriterEX 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 使用時のみ)	5MHz 以下
CPU	SH72115F
EXTAL	10.0MHz
FLASHROM	On-Chip

Table4.1-2 FlashWriterEX の設定

4.2 動作説明 (UDP 通信)

4.2.1 サンプルプログラム概要 (UDP 通信 アドホックモード)

UDP 通信サンプルプログラム (アドホックモード) は、下記の動作を行います。

1) クリエイト

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、UDP ポートを開放します。その後、UDP 通信で受信したデータをそのまま UDP 通信で送信元へ送信します。
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など) を使用して行ってください。
※ UDP エコーバックサーバ動作の詳細は、「4.2.3 UDP 通信エコーバックサーバ動作」を参照してください。

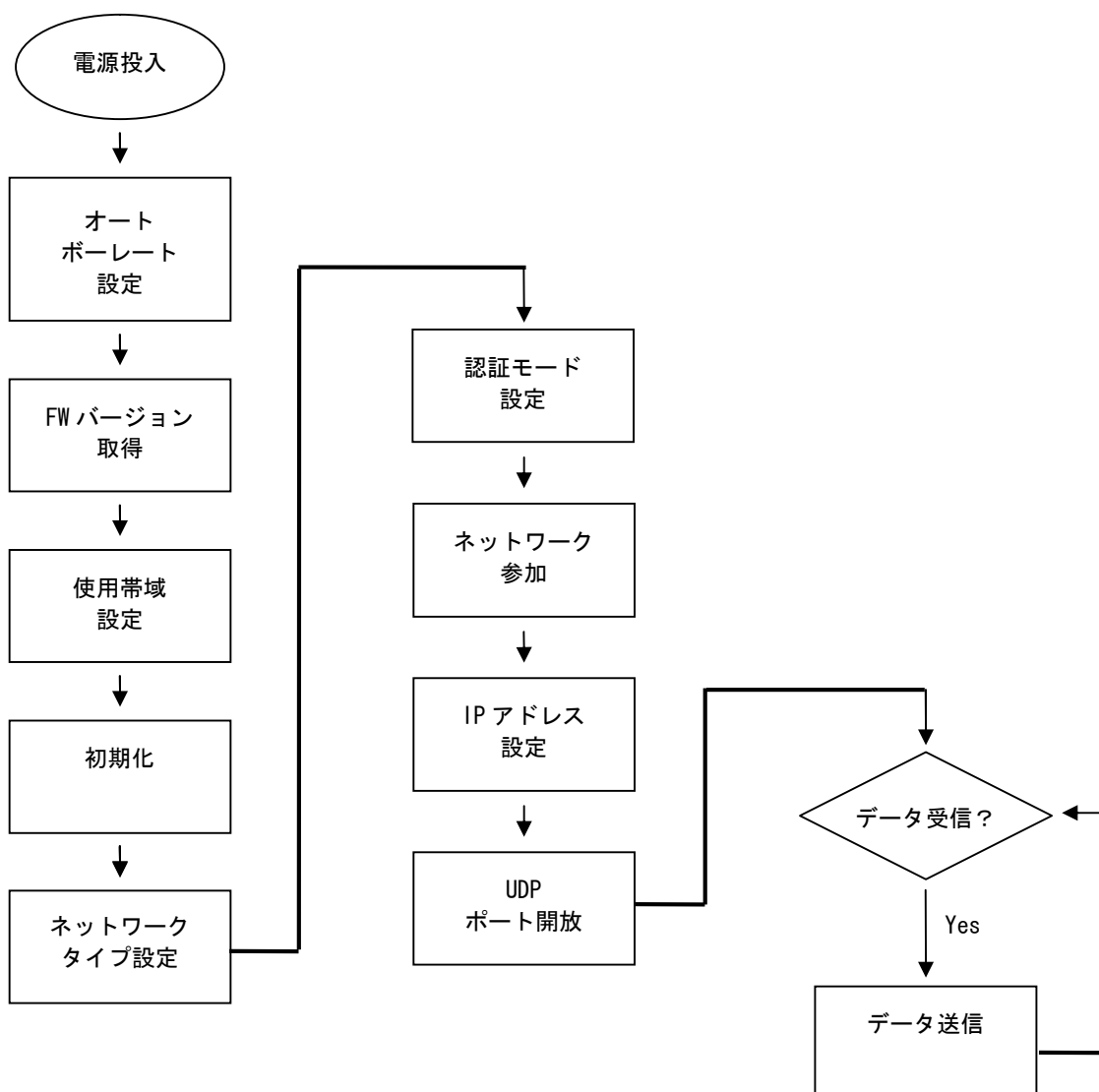


Fig 4.2-1 UDP 通信サンプルプログラム アドホッククリエイイト PC-WiFi-01 制御フロー

2) ジョイン

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、アドホックモードでアクセスポイントに接続した後、UDP 通信で受信したデータをそのまま UDP 通信で送信元に送信します。
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など）を使用して行ってください。
※ UDP エコーバックサーバ動作の詳細は、「4.2.3 UDP 通信エコーバックサーバ動作」を参照してください。

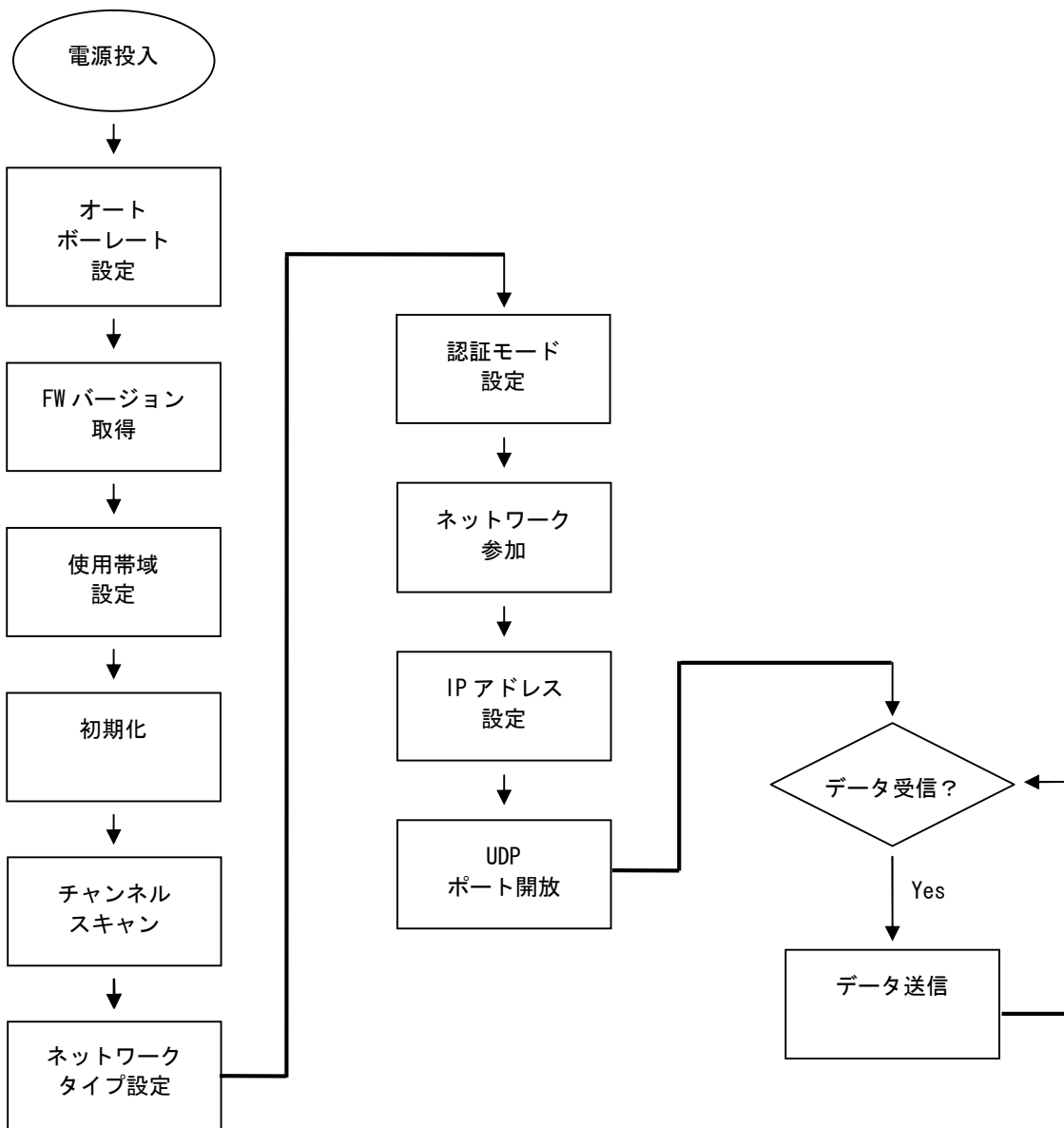


Fig 4. 2-1 UDP 通信サンプルプログラム アドホッククリエイト PC-WiFi-01 制御フロー

4.2.2 サンプルプログラム概要 (UDP 通信 インフラストラクチャモード)

TCP/IP 通信サンプルプログラム (インフラストラクチャモード) は、下記の動作を行います。

- SCIF1 に接続された PC-WiFi-01 に対してコマンドを送信し、UDP エコーバックサーバを構築します。
UDP 通信で受信したデータをそのまま UDP 通信で送信元に送信します。
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など) を使用して行ってください。
※ UDP エコーバックサーバ動作の詳細は、「4.2.3 TCP/IP 通信エコーバックサーバ動作」を参照してください。

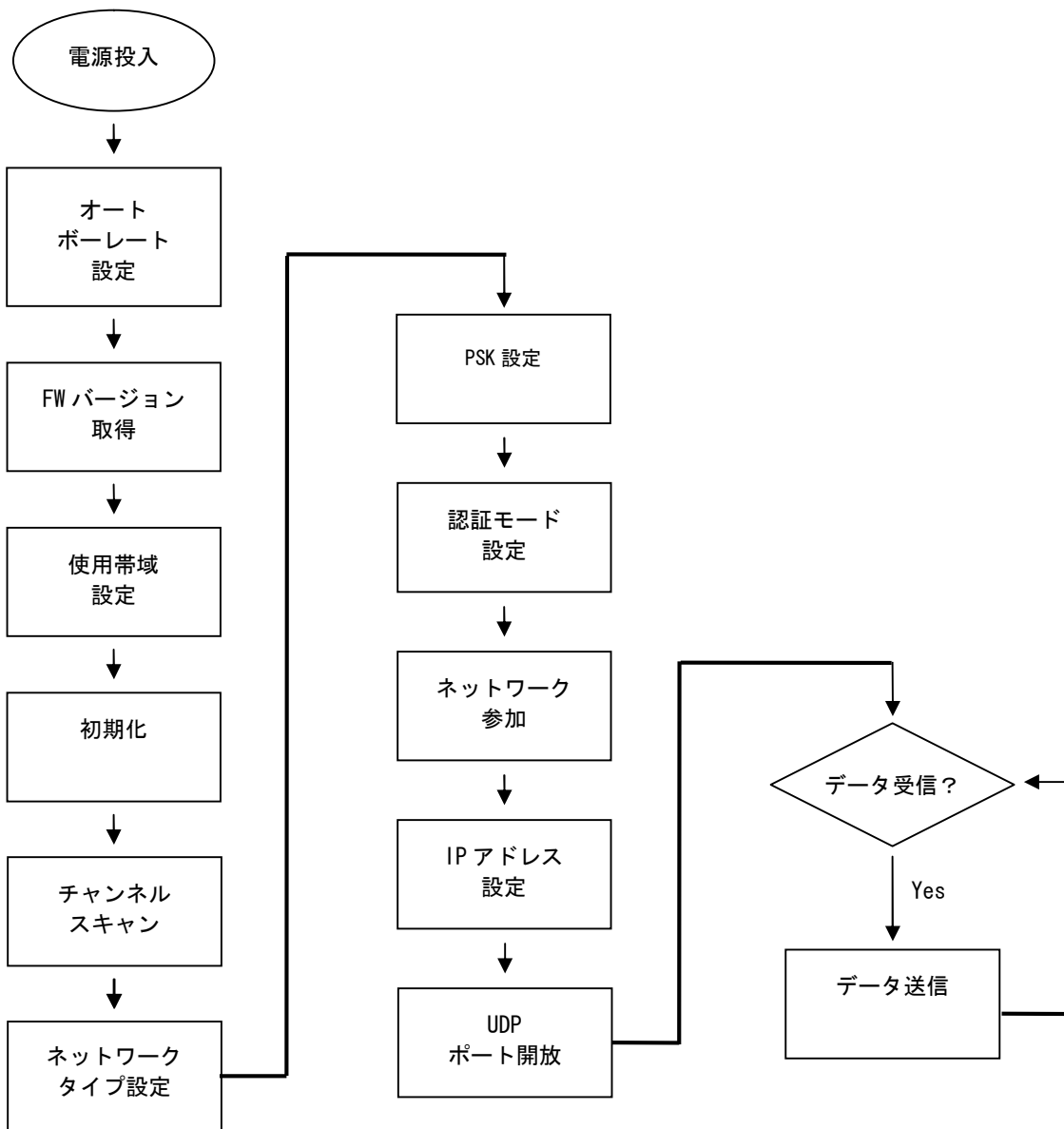


Fig 4.2-3 UDP 通信サンプルプログラム インフラストラクチャ PC-WiFi-01 制御フロー

4.2.3 UDP 通信エコーバックサーバ動作

(1) ネットワーク設定

以下にUDP 通信エコーバックサーバのネットワーク設定を記します。※

UDP 通信サンプルプログラムネットワーク設定 (アドホックモード)	
使用帯域	2.4GHz
使用チャンネル	1ch
ネットワーク接続	アドホックモード
送信レート	自動設定
送信レベル	ロウレベルモード
PSK	-
アクセスポイント	PC-WiFi-01Sample
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	0.0.0.0
使用ポート	8001

Table4.2-1 UDP 通信ネットワーク設定 (アドホックモード)

UDP 通信サンプルプログラムネットワーク設定 (インフラストラクチャモード)	
使用帯域	2.4GHz
使用チャンネル	1ch
ネットワーク接続	インフラストラクチャモード
送信レート	自動設定
送信レベル	ロウレベルモード
PSK	PC-WiFi-01SamplePSK
アクセスポイント	PC-WiFi-01Sample
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	0.0.0.0
使用ポート	8001

Table4.2-2 UDP 通信ネットワーク設定 (インフラストラクチャモード)

- ※ これらの設定は弊社の環境において設定した値となっています。ご利用時は、お使いの環境のネットワーク管理者に問い合わせ、ご利用になられる環境に沿ったそれぞれ適切な値を設定してください。
- ※ これらの設定値は、各サンプルプログラムのソースフォルダ内にある「NetworkSetting.h」にて定義されています。

(2) UDP 通信エコーバックサーバ動作 (アドホックモード クリエイト)

以下の手順に従い、UDP 通信エコーバックサーバの動作を確認してください。

- ① 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。
- ② CPU ボードの LD2 が点灯したことを確認し、アドホック通信機器の設定を行います。
その際、使用する設定は「**4.2-1 UDP 通信ネットワーク設定 (アドホックモード)**」で設定した値となります。
- ③ アドホック機器を無線 LAN ネットワークに接続し、エコーバックが行われることを確認してください。
- ④ 以上で UDP 通信エコーバックサーバ動作 (アドホックモード クリエイト) の動作は終了です。

(3) UDP 通信エコーバックサーバ動作 (アドホックモード ジョイン)

以下の手順に従い、UDP 通信エコーバックサーバの動作を確認してください。

- ① アドホック通信機器の設定を行い、無線 LAN ネットワークに接続します。
その際、使用する設定は「**4.2-1 UDP 通信ネットワーク設定 (アドホックモード)**」で設定した値となります。
- 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 CPU ボードの LD2 が点灯したことを確認し、アドホック通信機器の側からエコーバックが行われることを確認してください。
- ④ 以上で UDP 通信エコーバックサーバ動作 (アドホックモード ジョイン) の動作は終了です。

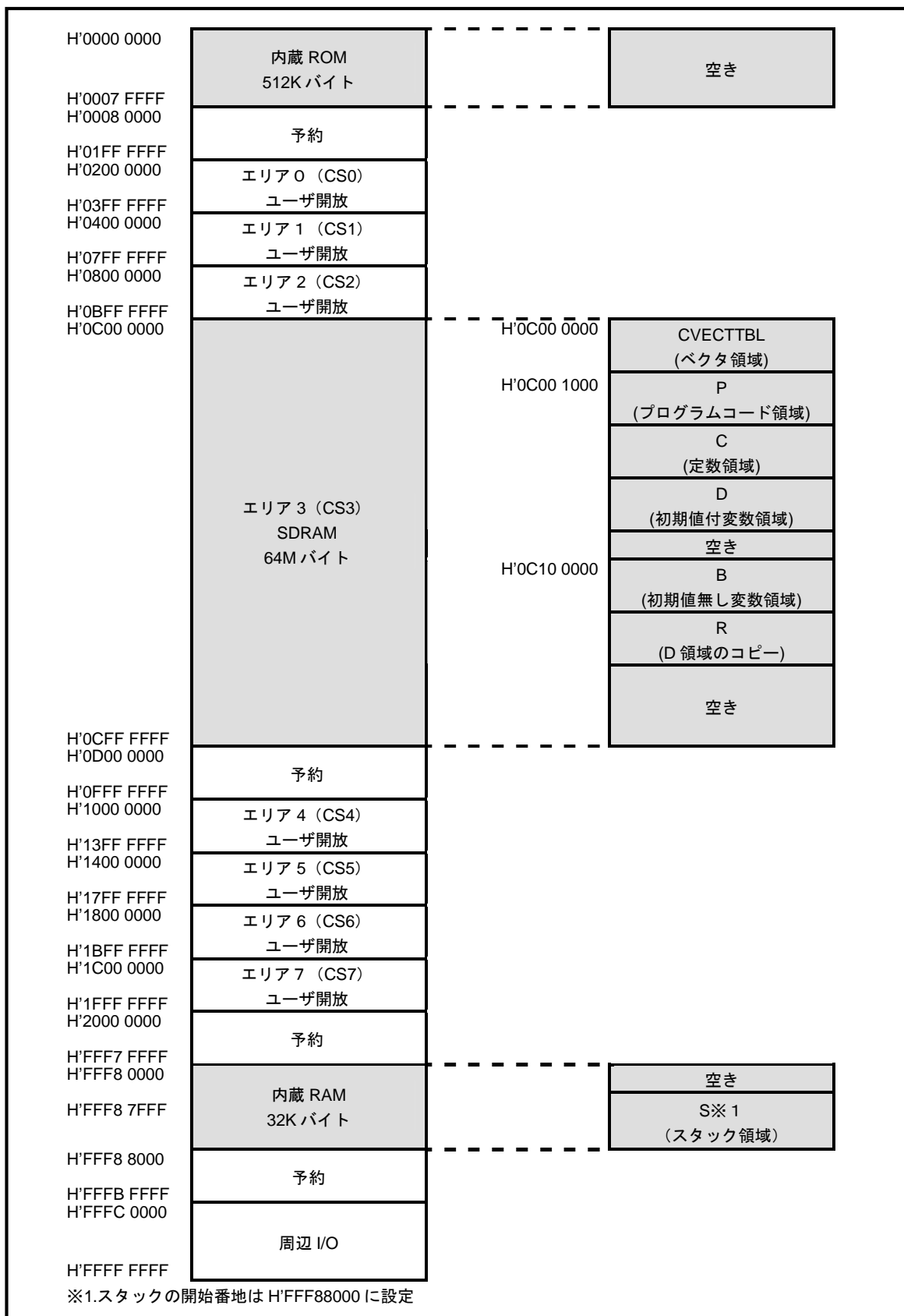
(4) UDP 通信エコーバックサーバ動作 (インフラストラクチャモード)

以下の手順に従い、UDP 通信エコーバックサーバの動作を確認してください。

- ① 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。
- ② ホスト PC 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、Ethernet 接続の設定を行います。
その際、使用する設定は「**4.2-2 UDP 通信ネットワーク設定 (インフラストラクチャモード)**」で設定した値となります。
- ③ 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。
- ④ 以上で UDP 通信エコーバックサーバ動作 (インフラストラクチャモード) の動作は終了です。

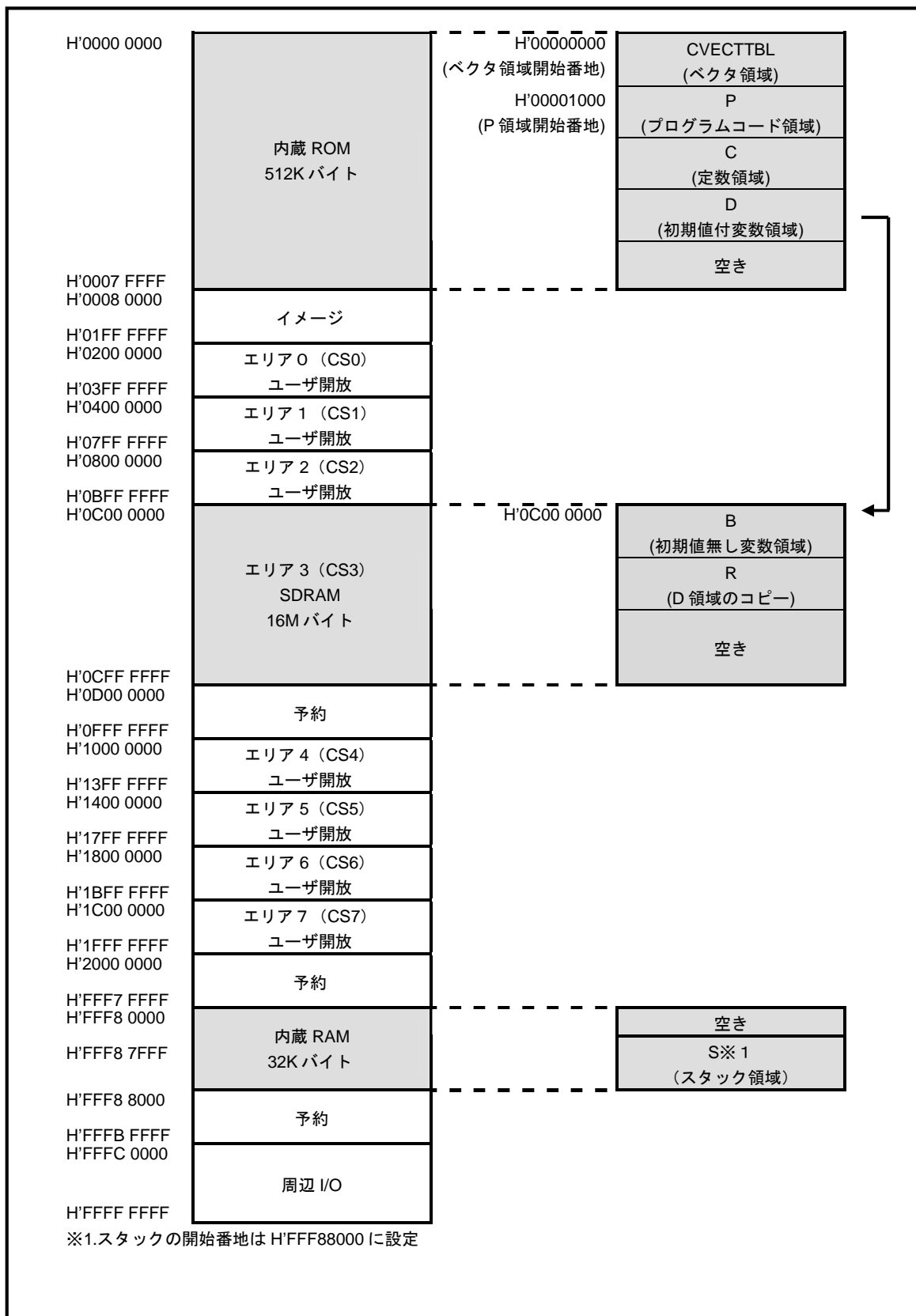
4.3 RAM動作時のメモリマップ (UDP通信サンプルプログラム共通)

メモリマップを以下に示します。



4.4 ROM 動作時のメモリマップ (UDP 通信サンプルプログラム共通)

メモリマップを以下に示します。



5. PC-WIFI-01 制御方法

5.1 概要

PC-WiFi-01 はホスト CPU とのインタフェースにシリアルインタフェースを採用しています。
ホスト CPU はシリアルインタフェースから ASCII コマンドを送信することで PC-WiFi-01 の操作を行い、初期化、ネットワークの設定、データの送受信などを行います。

5.2 シリアルインタフェース

5.2.1 シリアルインタフェース仕様

PC-WiFi-01 のシリアルインタフェース仕様を以下に記します。

機能	仕様
通信方式	調歩同期式
ボーレート	9,600/19,200/38,400/57,600/115,200/200,000 230,400/460,400/921,600/1,843,200/3,686,400 bps
データビット	8 ビット (固定)
ストップビット	2 ビット (固定)
パリティ	なし (固定)
フロー制御	なし (固定)

Table 5.2-1 シリアルインタフェース仕様

5.2.2 シリアルインタフェースフロー

PC-WiFi-01 とホスト CPU とのシリアル通信は以下の流れで行われます。

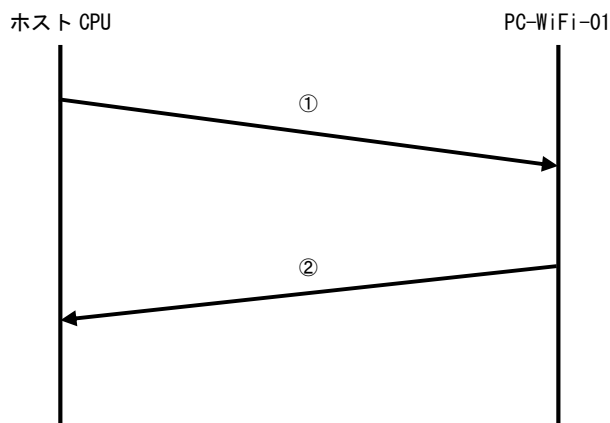


Fig 5.2-1 シリアルインタフェース制御フロー

- ① PC-WiFi-01 へのコマンド入力です。ASCII 文字列によるコマンドを送信することで、ホスト CPU から PC-WiFi-01 を制御することができます。
- ② PC-WiFi-01 からのレスポンスです。ASCII 文字列と Hex データの組み合わせでホスト CPU に応答が返されます。

※ 送信するコマンドおよびレスポンスの詳細に関しましては、「5.5 コマンドリファレンス」か CD に同梱されている「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」を参照してください。

5.3 AT コマンド

PC-WiFi-01 が認識するコマンド (AT コマンド) は、ASCII の文字列によって構成されます。

以下にコマンド例を記します。

コマンドの詳細に関しては、「5.5 コマンドリファレンス」か、CD 内に同梱されている

「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」を参照してください。

at+rsi_band=0<0x0D><0x0A>

コマンド	説明
at+rsi_band=0	使用帯域を設定するコマンドです。例は 0 (=2.4GHz 帯) を使用する設定となります。 データは ASCII の文字列となります。
<0x0D><0x0A>	コマンドの最後に送信する CR (0x0D) と LF (0x0A) コードです。 この 2 つのコードを送信することで、PC-WiFi-01 はコマンドを認識します。

Table 5.2-2 シリアルコマンド例

コマンドに対し、PC-WiFi-01 は以下のようなレスポンスを返します*。

OK<0x0D><0x0A>

コマンド	説明
OK	コマンドが正常に実行されたことを示す ASCII の文字列です。
<0x0D><0x0A>	レスポンスの最後に付属する CR (0x0D) と LF (0x0A) コードです。

Table 5.2-3 シリアルコマンドレスポンス例

* レスポンスの内容は先に送信したコマンドによって異なります。詳細に関しては、「5.5 コマンドリファレンス」か CD 内に同梱されている「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」を参照してください。

5.4 ボーレートの自動設定

PC-WiFi-01 のシリアル通信のボーレートを設定するには、ボーレート設定コマンドを使用するか、オートボーレート設定を行う必要があります。*

※ PC-WiFi-01 の初期ボーレートの値は 115,200bps となります。また、「Table 5.2-1 シリアルインタフェース仕様」のボーレート欄にない値には設定できません。

ボーレートの自動設定が終了した後、PC-WiFi-01 はファームウェアアップデートの確認を行います。PC-WiFi-01 は以下のデータを送信しますので、18 秒以上待つか、<0x6E><0x0A>を送信してください。

```
<0x20>WELCOME<0x20>TO<0x20>REDPINE<0x20>SIGNALS<0x0D><0x0A>
<0x20><0x0D><0x0A>
<0x20>Firmware<0x20>upgrade<0x20>(y/n)<0x0C><0x0D><0x0A>
```

18 秒以上待つか<0x6E><0x0A>を送信すると、PC-WiFi-01 は以下のデータを送信します。

```
<0x20>Loading...<0x20><0x0D><0x0A>
<0x20>Loading<0x20>Done<0x0D><0x0A>
```

これらのデータの受信をもって、PC-WiFi-01 は AT コマンドを受信する準備が整いました。

5.4.1 オートボーレート

オートボーレートは電源 ON 後に行われる自動処理です。ホスト CPU と特定の通信を行うことにより、通信に用いるボーレートを自動で設定することが出来ます。

ホスト CPU は、「Table 5.2-1 シリアルインタフェース仕様」の仕様に合わせて通信パラメータを設定して下さい。

※ オートボーレート処理の詳細は、「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」の「2.1 Auto Baud Rate Detection (ABRD)」を参照して下さい。

- ① ホスト CPU と PC-WiFi-01 を接続し、電源を入れます。
- ② PC-WiFi-01 の起動が完了するまで 100ms 以上待機します。
- ③ ホスト CPU から<0x1C>を送信します。
- ④ ホスト CPU が<0x55>を受信することを確認します。
※ ホスト CPU が<0x55>を受信出来ない場合、200ms 以上経過した後に再度 <0x1C>を送信します。
- ⑤ ホスト CPU から<0x55>を送信します。

以上でオートボーレートの設定は終了です。

※ 電源 ON 後、ホスト CPU から<0x1C>を送信しない場合、約 18 秒後に自動的にオートボーレート処理が完了しボーレートの値は 115,200bps に設定されます。

5.4.2 ボーレート設定コマンド

PC-WiFi-01 のボーレートは、AT コマンドで変更することが可能です。

ボーレートの変更コマンドに関しては、「5.5.19 ボーレート設定コマンド」を参照してください。

5.5 コマンドリファレンス

PC-WiFi-01 の各サンプルプログラムで使用している AT コマンドを以下に記します。

各コマンドの詳細、他のコマンドに関しては CD 内に同梱されている「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」を参照してください。

※赤文字で書かれている部分に関しては、技術基準適合証明に関わる部分です。PC-WiFi-01, WM-RP-xx は「2.4GHz、1-13CH、送信出力電力 10dBm」にて技術基準適合証明を取得しております。これ以外の動作を行った場合、弊社は一切責任を負いませんのでご了承ください。

5.5.1 Band コマンド

コマンド説明	
説明	仕様周波数帯域の設定を行います。
コマンド	at+rsi_band
使用方法	at+rsi_band=band_val<0x0D><0x0A>
パラメータ説明	band_val 0 : 2.4GHz 帯
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A>

※「RS9110-N-11-22_24_26_28_Software_PRM-v2.54a.pdf」の「2.3.1 Band」では「5GHz」の設定の記載がありますが、「5GHz」は使用出来ません。

5.5.2 Init コマンド

コマンド説明	
説明	at+rsi_band コマンド送信後に要求されるコマンドです。
コマンド	at+rsi_init
使用方法	at+rsi_init<0x0D><0x0A>
パラメータ説明	-
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0xEC><0x0D><0x0A>

5.5.3 Scan コマンド

コマンド説明	
説明	指定したチャンネルを走査します。
コマンド	at+rsi_scan
使用方法	at+rsi_scan=chan_num,SSID<0x0D><0x0A>
パラメータ説明	chan_num 0 ~ 13 : 走査するチャンネル番号 (0 : 全チャンネル) SSID 対象のSSID
レスポンス	正常終了時 OK<uSSID1uModeRSSIval1><uSSID2uModeRSSIval2>... <0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0xFD><0x0D><0x0A> or ERROR<0xE7><0x0D><0x0A>

レスポンス詳細	
uSSID	SSID が ASCII で返されます。32 バイト固定で、不足分は <0x00> で補われます。
uMode	セキュリティモードが 1 バイトのバイナリで返されます。 0x00 : オープン 0x01 : WPA 0x02 : WPA2 0x03 : WEP
RSSI	RSSI の値が 1 バイトのバイナリで返されます。

5.5.4 ネットワークタイプ設定コマンド

コマンド説明	
説明	ネットワーク接続形態を設定します。
コマンド	at+rsi_network
使用方法	at+rsi_network=Parameter<0x0D><0x0A>
パラメータ説明	<p>at+rsi_network コマンドのパラメータには以下の種類があります。</p> <ul style="list-style-type: none"> ・ INFRASTRUCTURE インフラストラクチャ ・ IBSS, type, chan_num アドホック (セキュリティなし) ・ IBSS_SEC, type, chan_num アドホック (セキュリティ WEP) <p>IBSS, IBSS_SEC の type, chan_num に関しては以下の通りです。</p> <p>type IBSS の接続形態を ASCII で指定します。 0 : ジョイナー 1 : クリエイター</p> <p>chan_num IBSS を作成する際のチャンネル番号を ASCII で指定します。 type=1 のときのみ有効で、それ以外は 0 になります。</p>
レスポンス	<p>正常終了時 OK<0x0D><0x0A></p> <p>異常終了時 ERROR<0xF8><0x0D><0x0A></p>

5.5.5 Pre Shared Key コマンド

コマンド説明	
説明	セキュリティ接続を行う際に必要な PSK (Pre Shared Key) の設定を行います。
コマンド	at+rsi_psk
使用方法	at+rsi_psk=Pre Shared Key<0x0D><0x0A>
パラメータ説明	<p>Pre Shared Key PSK を ASCII で指定します (最大 31 文字)。 ただし、WEP 時は 16 進数での指定となるため、WEP-64bit は 10 桁、WEP-128bit は 26 桁の指定となります。</p>
レスポンス	<p>正常終了時 OK<0x0D><0x0A></p> <p>異常終了時 ERROR<0xF8><0x0D><0x0A></p>

5.5.6 Authentication Mode コマンド

コマンド説明	
説明	認証モードの設定を行います。
コマンド	at+rsi_authmode
使用方法	at+rsi_authmode=value<0x0D><0x0A>
パラメータ説明	value 設定値を ASCII で指定します。 0 : Open Key Authentication 1 : Shared Key Authentication
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A>

5.5.7 Join コマンド

コマンド説明	
説明	SSID への接続を行います。
コマンド	at+rsi_join
使用方法	at+rsi_join=SSID name, TxRate, TxPower<0x0D><0x0A>
パラメータ説明	SSID name 接続する SSID を ASCII で指定します (最大 32 文字)。 TxRate 「Table 5.5-1 送信レート表」参照 TxPower [※] 送信時の電力を ASCII で指定します。 0 : Low Power (7dBm) 1 : Medium Power (10dBm) 2 : High Power (16 or 17dBm)
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xFD><0x0D><0x0A> or ERROR<0xF8><0x0D><0x0A> or ERROR<0xF4><0x0D><0x0A> or ERROR<0xF1><0x0D><0x0A> or ERROR<0xEE><0x0D><0x0A> or ERROR<0xED><0x0D><0x0A> or ERROR<0xEB><0x0D><0x0A>

通信レート (Mbps)	802.11 b/g/n	TxRate パラメータ
Auto-rate	-	0
1	b	1
2	b	2
5.5	b	3
11	b	4
6	g	5
9	g	6
12	g	7
18	g	8
24	g	9
36	g	10
48	g	11
54	g	12
MCS0 (6.5)	n	13
MCS1 (13)	n	14
MCS2 (19.5)	n	15
MCS3 (26)	n	16
MCS4 (39)	n	17
MCS5 (52)	n	18
MCS6 (58.5)	n	19
MCS7 (65)	n	20

Table 5.5-1 送信レート表

5.5.8 IP Parameter 設定コマンド

コマンド説明	
説明	IP アドレス、サブネットマスク、デフォルトゲートウェイの設定を行います。
コマンド	at+rsi_ipconf
使用方法	at+rsi_ipconf=DHCP_MODE, IP Address, SUBNET, GATEWAY<0x0D><0x0A>
パラメータ説明	<p>DHCP_MODE DHCP の設定を ASCII で指定します。 0 : Manual 1 : DHCP 2 : Auto-IP</p> <p>IP Address IP アドレスを ASCII で指定します。DHCP 時は「0」のみの指定としてください。</p> <p>SUBNET サブネットマスクを ASCII で指定します。DHCP 時は「0」のみの指定としてください。</p> <p>GATEWAY ゲートウェイを ASCII で指定します。DHCP 時は「0」のみの指定としてください。</p>
レスポンス	<p>正常終了時 OK<MAC_Address><IP_Address><SUBNET><Gateway><0x0D><0x0A> MAC_Address 64 バイトの MAC アドレスがバイナリで返されます。 IP_Address 4 バイトの IP アドレスが返されます。 SUBNET 4 バイトのサブネットマスクが返されます。 Gateway 4 バイトのゲートウェイが返されます。</p> <p>異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0xFC><0x0D><0x0A> or ERROR<0xFB><0x0D><0x0A> or ERROR<0xF9><0x0D><0x0A> or ERROR<0xF6><0x0D><0x0A> or ERROR<0xF0><0x0D><0x0A> or ERROR<0x7E><0x0D><0x0A> or ERROR<0x7D><0x0D><0x0A></p>

5.5.9 TCP/IP Client 設定コマンド

コマンド説明	
説明	TCP/IP Client 接続を行います
コマンド	at+rsi_tcp
使用方法	at+rsi_tcp=host,port,lport<0x0D><0x0A>
パラメータ説明	host 接続先 IP アドレスを ASCII で指定します。 port 接続先ポート番号を ASCII で指定します。 lport 自分のポート番号を ASCII で指定します。
レスポンス	正常終了時 OK<sock_handle><0x0D><0x0A> sock_handle オープンしたソケット番号が1バイトのバイナリで返されます。 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0x84><0x0D><0x0A> or ERROR<0x81><0x0D><0x0A>

5.5.10 TCP/IP Server 設定コマンド

コマンド説明	
説明	TCP/IP Server 接続を行います
コマンド	at+rsi_ltcp
使用方法	at+rsi_ltcp=port<0x0D><0x0A>
パラメータ説明	port 自分のポート番号を ASCII で指定します。
レスポンス	正常終了時 OK<sock_handle><0x0D><0x0A> sock_handle オープンしたソケット番号が1バイトのバイナリで返されます。 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0x81><0x0D><0x0A>

5.5.11 UCP Client 設定コマンド

コマンド説明	
説明	UDP Client 接続を行います
コマンド	at+rsi_udp
使用方法	at+rsi_udp=lphost,rport,lport<0x0D><0x0A>
パラメータ説明	lphost 接続先 IP アドレスを ASCII で指定します。 rport 接続先ポート番号を ASCII で指定します。 lport 自分のポート番号を ASCII で指定します。
レスポンス	正常終了時 OK<sock_handle><0x0D><0x0A> sock_handle オープンしたソケット番号が1バイトのバイナリで返されます。 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0x81><0x0D><0x0A>

5.5.12 UDP Server 設定コマンド

コマンド説明	
説明	UDP Server 接続を行います
コマンド	at+rsi_ludp
使用方法	at+rsi_ludp=lport<0x0D><0x0A>
パラメータ説明	lport 自分のポート番号を ASCII で指定します。
レスポンス	正常終了時 OK<sock_handle><0x0D><0x0A> sock_handle オープンしたソケット番号が1バイトのバイナリで返されます。 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0x81><0x0D><0x0A>

5.5.13 Send data コマンド

コマンド説明	
説明	オープンしたソケットからデータを送信します。
コマンド	at+rsi_snd
使用方法	at+rsi_snd=hn, sz, Dip, Dport, stream<0x0D><0x0A>
パラメータ説明	hn 送信に用いるソケット番号を ASCII で指定します。 sz 送信サイズを ASCII で指定します (最大 1400 バイト)。 Dip 送信先の IP アドレスを ASCII で指定します。 Dport 送信先のポート番号を ASCII で指定します。 stream※ 送信データです。
レスポンス	正常終了時 OK<sz><0x0D><0x0A> sz 実際に送信したサイズが 2 バイトのバイナリで返されます。 異常終了時 ERROR<0xFE><0x0D><0x0A> or ERROR<0xF8><0x0D><0x0A> or ERROR<0xF7><0x0D><0x0A> or ERROR<0xE0><0x0D><0x0A> or ERROR<0x1F><0x0D><0x0A>

※ コマンドの終端として、<0x0D><0x0A>を使用している為 送信したいバイナリデータ上に<0x0D><0x0A>が存在する場合、stream の該当するデータ箇所を以下の様に変更する必要があります。

- 例 1) <0x41><0x42><0x43><0x0D><0x0A>を送信する場合
 <0x0D><0x0A>の部分を<0xDB><0xDC>と置き換えて、
 <0x41><0x42><0x43><0xDB><0xDC>として送信する。
- 例 2) <0x41><0x42><0x43><0x0D><0x0A><0x31><0x32>を送信する場合
 <0x0D><0x0A>の部分を<0xDB><0xDC>と置き換えて、
 <0x41><0x42><0x43><0xDB><0xDC><0x31><0x32>として送信する。
- 例 3) <0x41><0x42><0x43><0xDB><0x31><0x32>を送信する場合
 <0xDB>の部分を<0xDB><0xDD>として、
 <0x41><0x42><0x43><0xDB><0xDD><0x31><0x32>として送信する。
- 例 4) <0x41><0x42><0x43><0xDB><0xDC><0x31><0x32>を送信する場合
 <0xDB><0xDC>の部分を<0xDB><0xDD><0xDC>として、
 <0x41><0x42><0x43><0xDB><0xDD><0xDC><0x31><0x32>として送信する。

例 5) <0x41><0x42><0x43><0x0D><0x0A><0xDB><0x31><0x32>を送信する場合
 <0x0D><0x0A><0xDB>の部分を<0xDB><0xDC><0xDB><0xDD>として、
 <0x41><0x42><0x43><0xDB><0xDC><0xDB><0xDD><0x31><0x32>として送信する。

例 6) <0x41><0x42><0x43><0x0D><0x0A><0xDB><0xDC><0x31><0x32>をする場合
 <0x0D><0x0A><0xDB><0xDC>の部分を<0xDB><0xDC><0xDB><0xDD><0xDC>として、
 <0x41><0x42><0x43><0xDB><0xDC><0xDB><0xDD><0xDC><0x31><0x32>として送信する。

5.5.14 Receive data コマンド

コマンド説明	
説明	オープンしたソケットからの受信を行います。
コマンド	-※
使用方法	-※
パラメータ説明	-
レスポンス	AT+RSI_READ<hn><sz><Sip><Sport><stream><0x0D><0x0A> hn 受信したソケットの番号が1バイトのバイナリで与えられます。 sz 受信したサイズが2バイトのバイナリで与えられます。 Sip 受信元の IP アドレスが4バイトのバイナリで与えられます。 Sport 受信元のポート番号が1バイトのバイナリで与えられます。 stream 受信したデータのストリームがバイナリで与えられます。

※ 受信をするためのコマンドは存在しません。PC-WiFi-01 が接続先からのデータを受信した場合、ホスト CPU にレスポンスに表した形でデータの受信を知らせます。

5.5.15 Close Socket コマンド

コマンド説明	
説明	オープンしたソケットのクローズを行います。
コマンド	at+rsi_cls [※]
使用方法	at+rsi_cls=hn<0x0D><0x0A> [※]
パラメータ説明	hn クローズするソケット番号を ASCII で指定します。
レスポンス	正常終了時 OK<sock_handle><0x0D><0x0A> sock_handle オープンしたソケット番号が1バイトのバイナリで返されます。 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0xFA><0x0D><0x0A> or ERROR<0x40><0x0D><0x0A>

※ 接続相手側が切断を行った場合、「AT+RSI_CLOSE<sock_handle><0x0D><0x0A>」が返ってきます。

5.5.16 Disassociate コマンド

コマンド説明	
説明	接続しているアクセスポイントからの切断を行います。
コマンド	at+rsi_disassoc
使用方法	at+rsi_disassoc<0x0D><0x0A>
パラメータ説明	-
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A>

5.5.17 ボーレート設定コマンド

コマンド説明	
説明	ボーレートの設定を行います
コマンド	at+rsi_baudrate
使用方法	at+rsi_baudrate=b_rate<0x0D><0x0A>
パラメータ説明	b_rate 設定するボーレートの値を ASCII で指定します。 有効な値に関しては下記の「Table 5.5-2 b_rate表」を参照してください。
レスポンス	正常終了時 OK 異常終了時 ERROR<0xF8><0x0D><0x0A> or ERROR<0xEF><0x0D><0x0A>

b_rate
9600
19200
38400
57600
115200
200000
230400
460800
921600
1843200
3686400

え w w

5.5.18 Soft Reset コマンド

コマンド説明	
説明	Soft Reset を行います。
コマンド	at+rsi_reset
使用方法	at+rsi_reset<0x0D><0x0A>
パラメータ説明	-
レスポンス	正常終了時 OK<0x0D><0x0A> 異常終了時 ERROR<0xF8><0x0D><0x0A>

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では一切責任を負いませんのでご了承下さい。
- ・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下さい。
- ・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

- ・ SH7211 は、株式会社ルネサスエレクトロニクスの登録商標、商標または商品名称です
 - ・ SuperH は、株式会社ルネサスエレクトロニクスの登録商標、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。

 **ALPHA PROJECT Co.,LTD.**

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834
<http://www.apnet.co.jp>
E-MAIL : query@apnet.co.jp