Rev 1.00.00

Renesas S7G2 用サンプル(ewarm WiFi_Socket_HTTP)の説明

(ewarm Version:8.23.1 / SSP Version 1.5.3)

1． Sample の免責について

・Sample に関する Tel/Fax でのご質問に関してはお受けできません。ただし、メールでのご質問に関してはお答えするよう努力はしますが、都合によりお答えできない場合もありますので予めご了承願います。

・Sample ソフトの不具合が発見された場合の対応義務はありません。また、この関連ソフトの使用方法に関する質問の回答義務もありませんので承知の上ご利用下さい。

・Sample ソフトは、無保証で提供されているものであり、その適用可能性も含めて，いかなる保証も行いません。また，本ソフトウェアの利用により直接的または間接的に生じたいかなる損害に関しても，その責任を負わないものとします。

2．サンプルのプロジェクト名

ワークスペース名	概要	プロジェクト名
S7G2_ewarm_WiFi_NetX_3	PMODA に WiFi モジュール(GT202)を接続して、固定 IP アドレスによる HTTP と Socket 通信のサンプル	WiFi_Socket_HTTP_GT202_PMODA ThreadX モードで動作 Socket 通信 (nx_tcp_socket_.....) NetX HTTP Server (g_http_server0)

統合開発環境
IAR ewarm(Version 8.23.1)
SSP(Version1.5.3)

3．Tera Term Pro のインストール

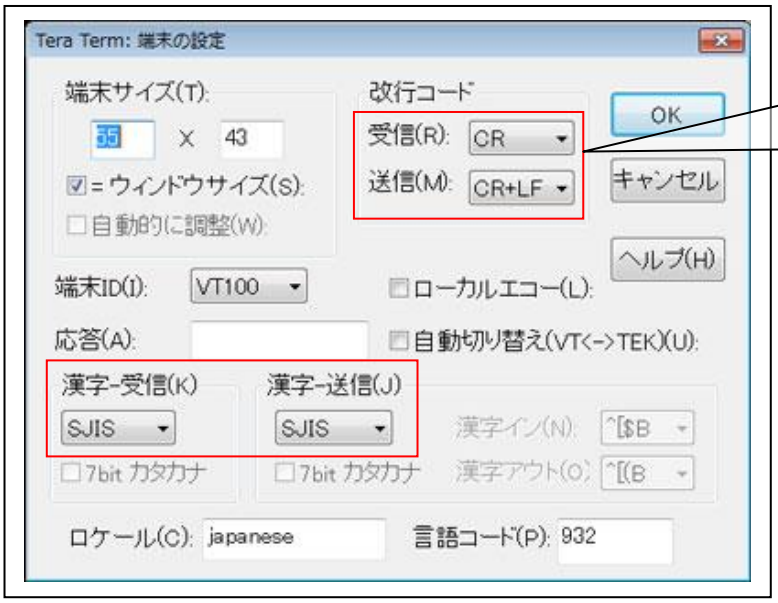
①「teraterm-4.80.exe」を検索してダウンロードする。
②PC にインストールし実行する
③シリアルポートの設定

COM 番号は、
PC 側でシリアル通信可能な番号を指定する。

115200BPS
8bit
none
1bit
none

の仕様にする。

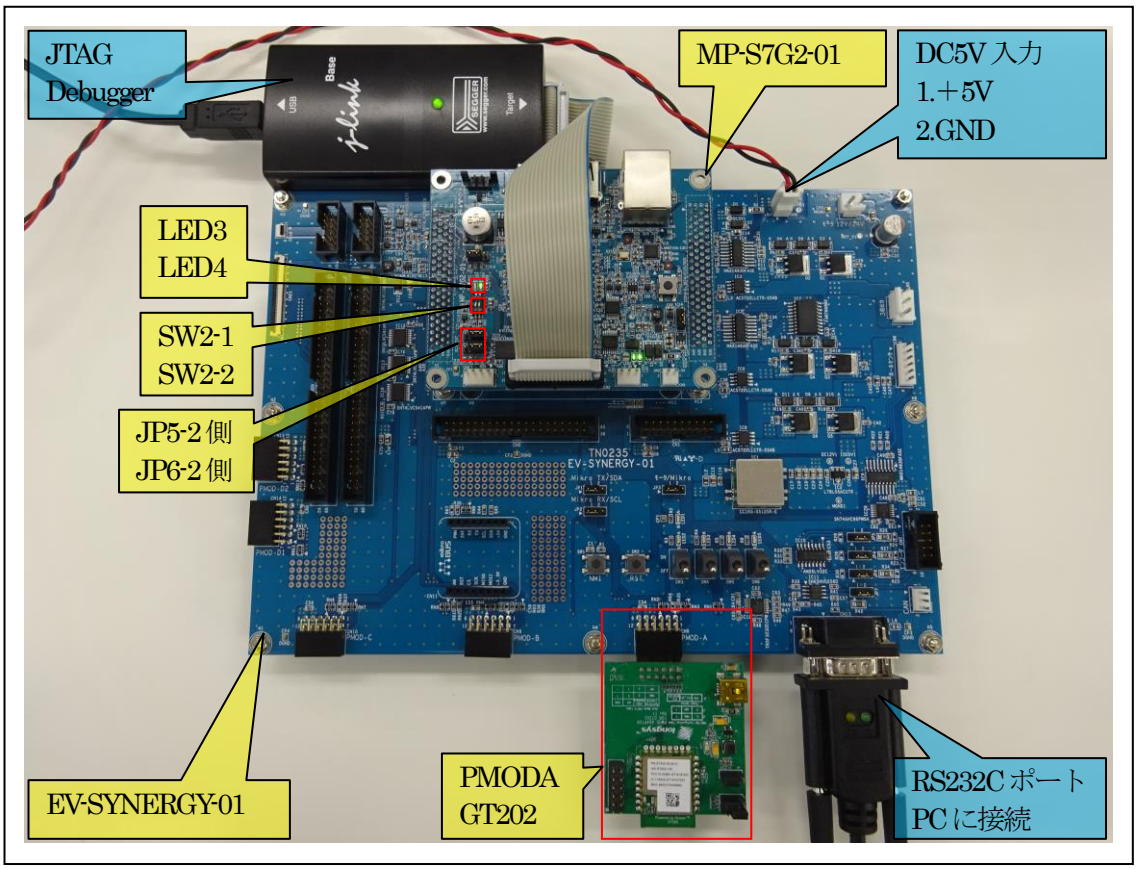
④端末の設定



USB シリアルコンバータ使用時に CR コードがカットされる設定の場合は、受信：LF にして下さい。

赤枠の設定にする。

4．動作構成



①PC 機と接続する RS232C ケーブルは、市販「クロスケーブル」でも可能です。
②USB-シリアル変換ケーブルを使用される場合は、「StarTech.com 社　ICUSB232FIN」推奨
③自作する場合は、下記の配線になります。

EV-RX/RZ-xx

DSUB 9pin オス	
1	NC
2	RXD
3	TXD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	NC

PC-Windows

DSUB 9pin オス	
1	NC
2	RXD
3	TXD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	NC

５．「S7G2_ewarm_WiFi_NetX_3」サンプルの説明

５－１．「WiFi_Socket_HTTP_GT202_PMODA」フォルダ構成とファイル名

S7G2_ewarm_WiFi_NetX_3¥WiFi_Socket_HTTP_GT202_PMODA			
Debug	Exe	WiFi_Socket_HTTP_GT202_PMODA.out	ELF ファイル、JTAG で使用
		WiFi_Socket_HTTP_GT202_PMODA.srec	モトローラーHEX ファイル
	List	WiFi_Socket_HTTP_GT202_PMODA.map	MAP ファイル、アドレス情報管理
	Obj	その他	自動生成ファイル
MP-S7G2-01（リンク指定）		dipsw.c	MP 基板上 DIP-SW 処理モジュール
		dipsw.h	dipsw.c 用ヘッダーファイル
		Led_blink.c	LED 処理モジュール
		led_brink.h	led_brink.c 用ヘッダーファイル
		sci2.c	シリアル通信処理モジュール
		sci2.h	sci2.c 用ヘッダーファイル
		stchar.h	文字系処理モジュール
		stchar.h	stchar.c 用ヘッダーファイル-
Script		S7G2.ld	ロケーション定義ファイル
Src		blink_thread_entry.c	led blink thread サンプルファイル
		dipsw_thread_entry.c	dipsw thread サンプルファイル
		http_server_fix_thread_entry.c	HTTP server Thread サンプルファイル
		http_server.c	const html サンプルファイル
		http_server.h	http_server.c 用ヘッダーファイル
		Socket_thread_entry.c	Socket server Thread サンプルファイル
		filex_stub.h	Filex 未使用時のダミーファイル
		(*1)http_server_fix_thread.c	synergy_gen 作成ファイル
		(*1)http_server_fix_thread.h	http_server_thread.c 用ヘッダーファイル
synergy_gen		Generate を行うと作成されるファイル	
synergy		Generate を行うと作成されるファイル	
synergy_cfg			
Configuration.xml		プロジェクト Generation ファイル	
PIN-EV-SYNERGY-S7G2.pincfg		PIN configuration 用ファイル	
その他		自動生成ファイル	

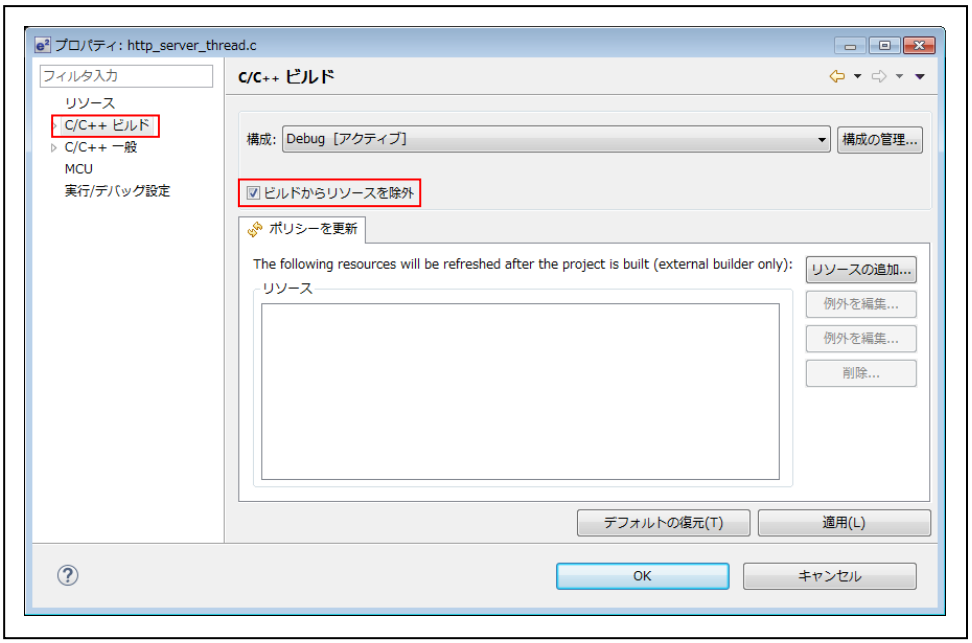
(*1) ５－２．FileX を使用しない場合の手順 を参照

５－２．FileX を使用しない場合の手順

① <synergy_gen>下記２ファイルを<src>に Copy する。
http_server_fix_thread.c
http_server_fix_thread.h

② <synergy_gen>下記２ファイルを「ビルド除外」にする。
http_server_fix_thread.c
http_server_fix_thread.h

＜除外方法＞
（1） ファイルを選択してプロパティを指定する。
（2）「C/C++ビルド」を指定する。
（3）「☑ビルドからリソースを除外」にする。



③ <src>に Copy した「http_server_fix_thread.c」内「void http_server_init0(void)」引数を変更する。

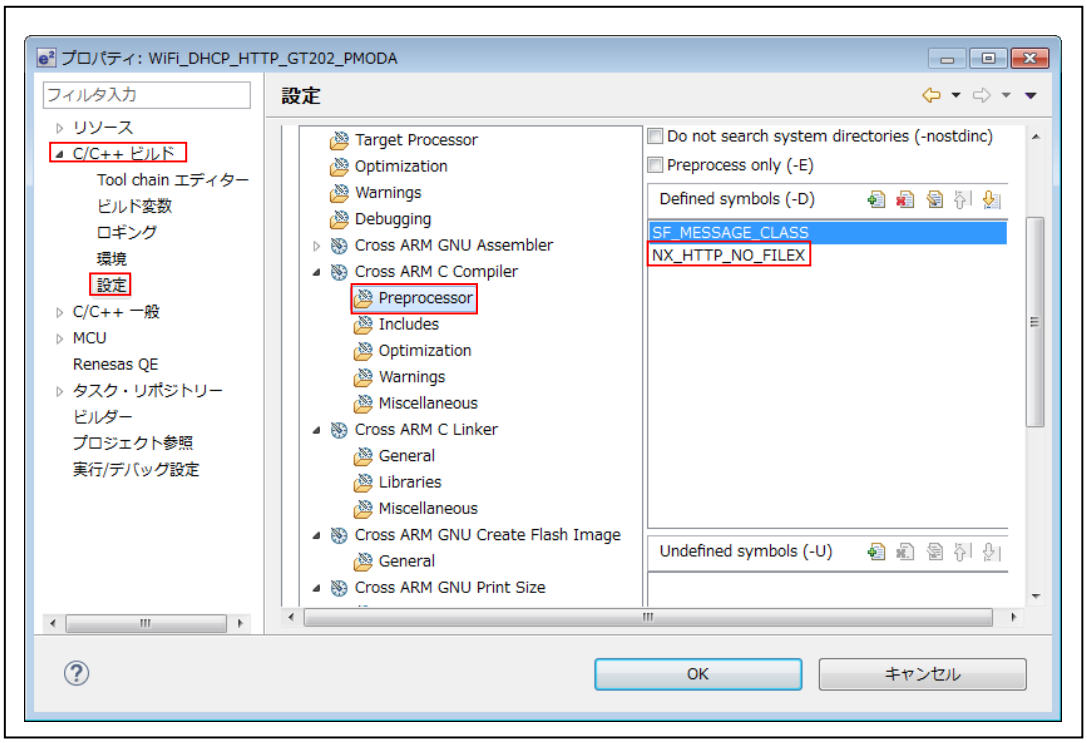
g_http_server0_err=nx_http_server_create(&g_http_server0, "g_http_server0 HTTP Server", &g_ip0,
&SYNERGY_NOT_DEFINED, &g_http_server0_stack_memory[0], 4096,
↑NX_NULL,に変更する。

```
g_http_server0_err = nx_http_server_create (&g_http_server0, "g_http_server0 HTTP Server", &g_ip0,
                                            NX_NULL, &g_http_server0_stack_memory[0], 4096,
                                            &g_packet_pool0, authentication_check, request_notify);
```

④ <src>に Copy した「http_server_fix_thread.h」に「#include "filex_stub.h"」を追記する。

```
#include "filex_stub.h"
#include "nx_http_server.h"  // 必ず「nx_http_server.h」の上に追記
```

⑤「Cross ARM C Compiler」「Preprocessor」に 「NX_HTTP_NO_FILEX」を追加する。



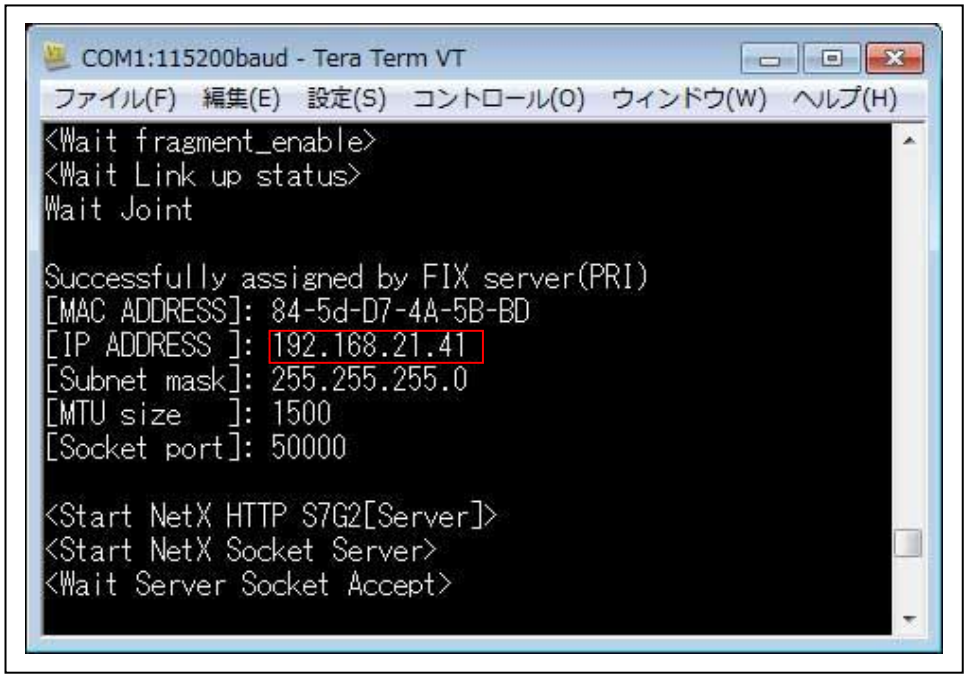
以上の作業が必要です。

５－３．サンプルの動作説明

＜HTTP server Thread＞

① WiFi ルーターに Joint する。

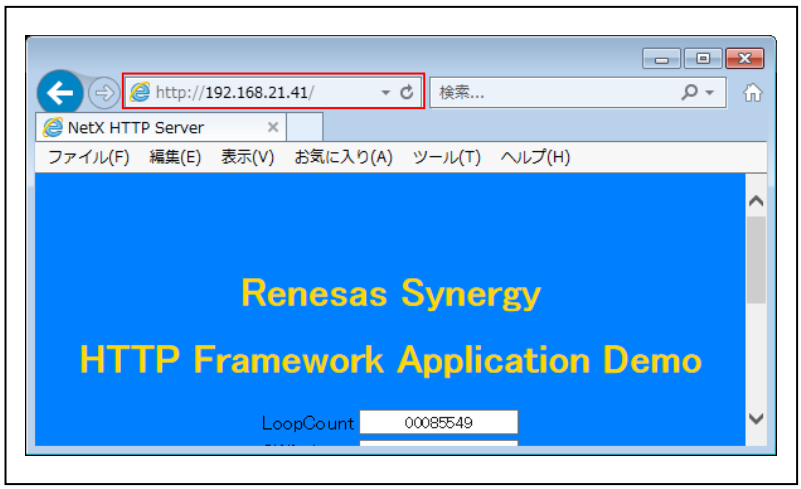
② Joint 成功後、固定 IP アドレス情報を Term 画面に表示する。



【固定 IP アドレス定義】

http_server_fix_thred_entry.c: #define MY_IP IP_ADDRESS (192,168,21,41)

③ Windows PC のインターネット ブラウザを使用して HTTP Server にアクセスする。



＜Socket server Thread＞

① Windows PC のコネクション「接続」を待つ。
接続完了にて、Term 画面「"<The connection established it.>”」に表示する。

② Windows PC からの受信データを待つ。
受信完了にて、Term 画面「"<TCP data receive length(%d) %02X>”,length, rx_buff[0]」と受信バイト数と受信データの１バイト目データを表示する。

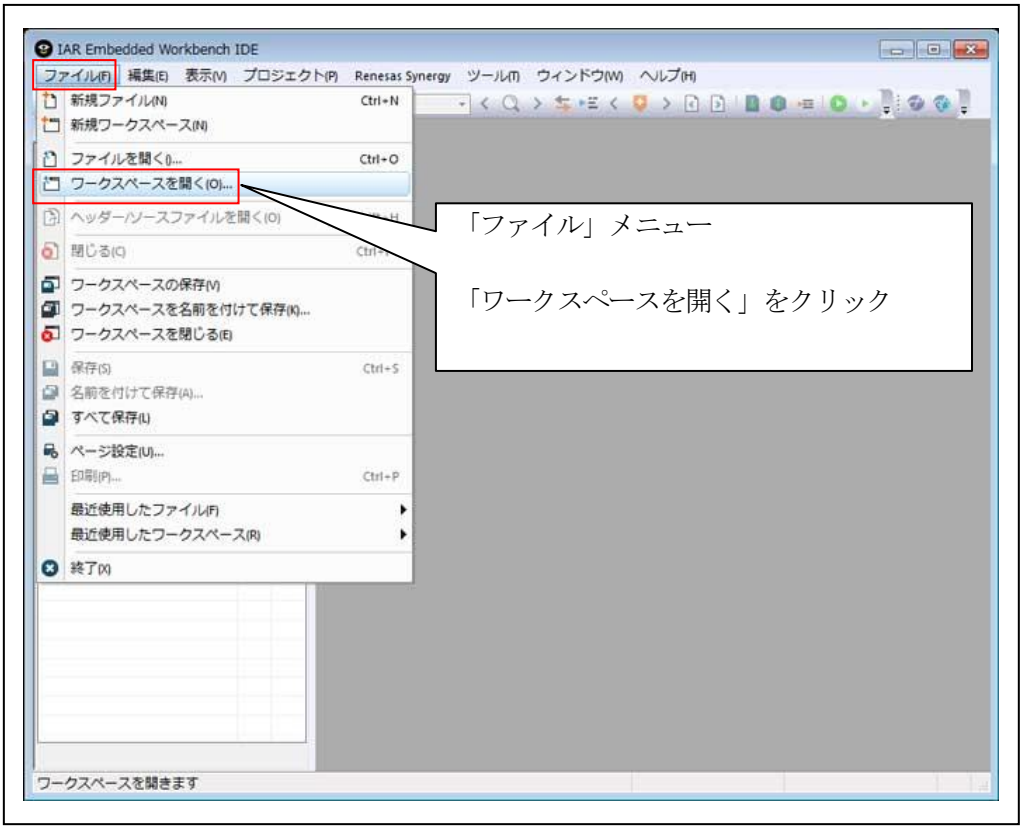
③ ⑤で受信したデータを Windows PC にエコーバックとして送信する。
送信完了にて、Term 画面「” <TCP data send length(%d) %02X>”,length, tx_buff[0]」と送信バイト数と送信データの１バイト目データを表示する。
送信不可の場合、Term 画面「" <TCP data send failed!>”」と表示して終了。

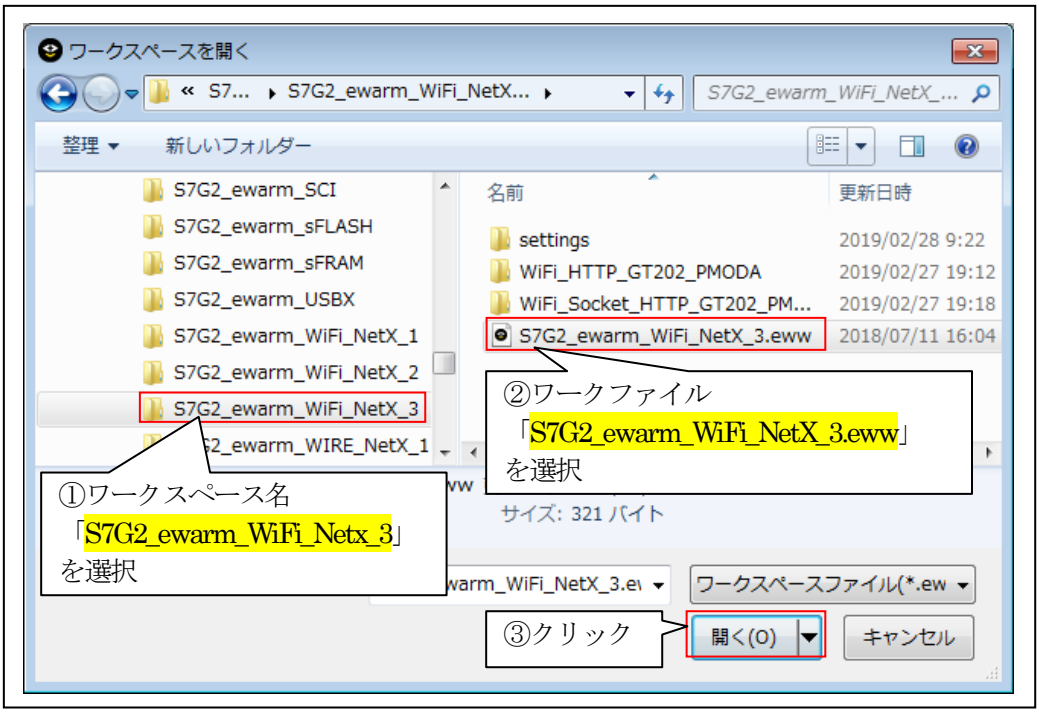
④ Windows PC 側でディスコネクション「切断」するまで、④からの処理を繰り返す。

⑤ Windows PC 側でディスコネクション「切断」した場合は、③からの処理を繰り返す。

6．「S7G2_ewarm_WiFi_NetX_3」をインポートする。

6－1．ワークスペース名の指定





6－2．コンフィグレータの起動

☆詳細操作は「ewarm_synergy_Import.pdf」の２－３項を参照して下さい。

７．デバッグ操作

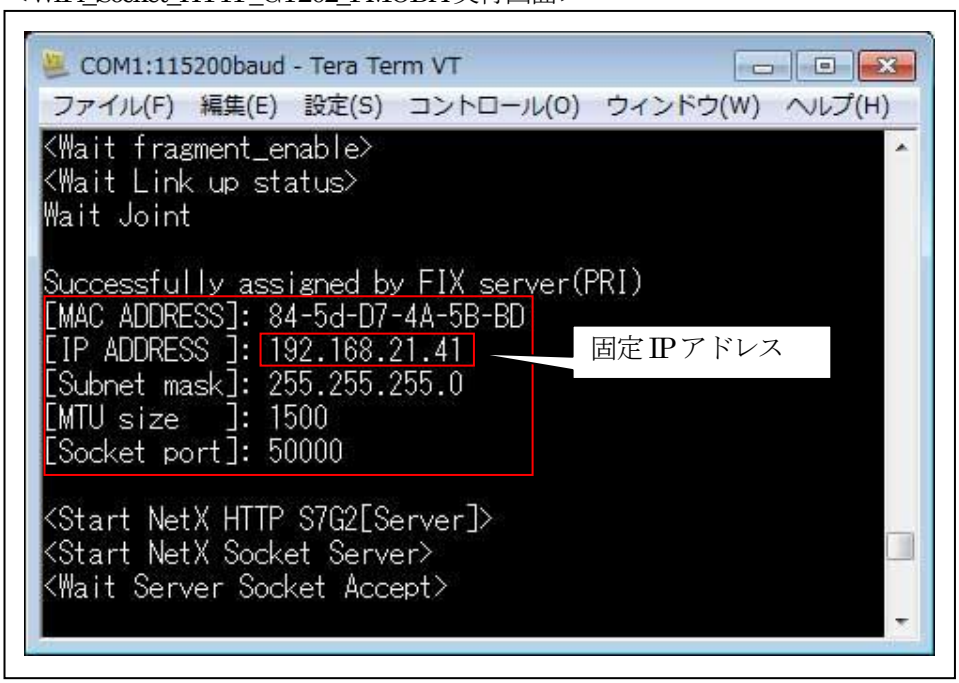
７－１．デバッグ構成の設定

☆詳細操作「J-Link 版」は「ewarm_synergy_Import.pdf」の３－１項を参照して下さい。
☆詳細操作「I-jet 版　」は「ewarm_synergy_Import.pdf」の３－２項を参照して下さい。

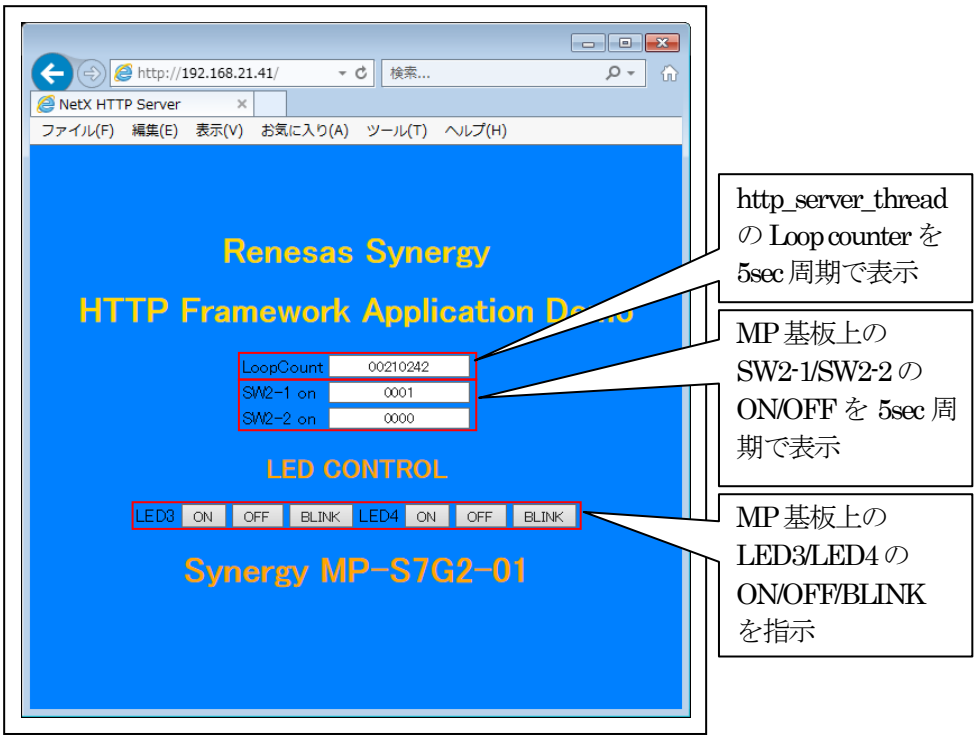
７－２．デバッグの開始

☆詳細操作は「ewarm_synergy_Import.pdf」の３－３項を参照して下さい。

＜WiFi_Socket_HTTP_GT202_PMODA 実行画面＞



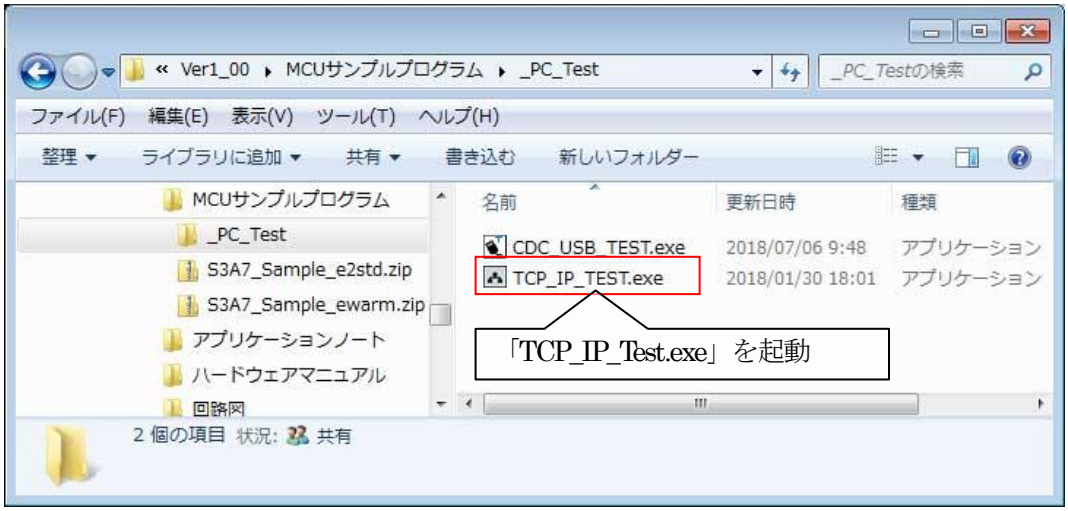
＜インターネット　ブラウザ実行画面＞



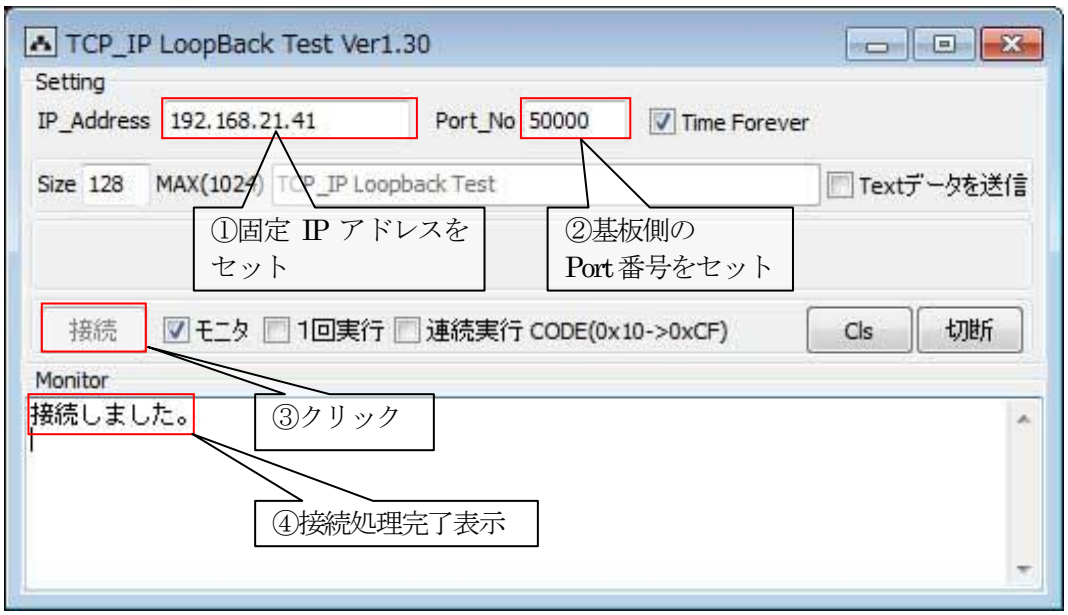
７－３．Windows PC 側のテスト用プログラムを使用して動作確認

１）「TCP_IP_TEST」を起動する。

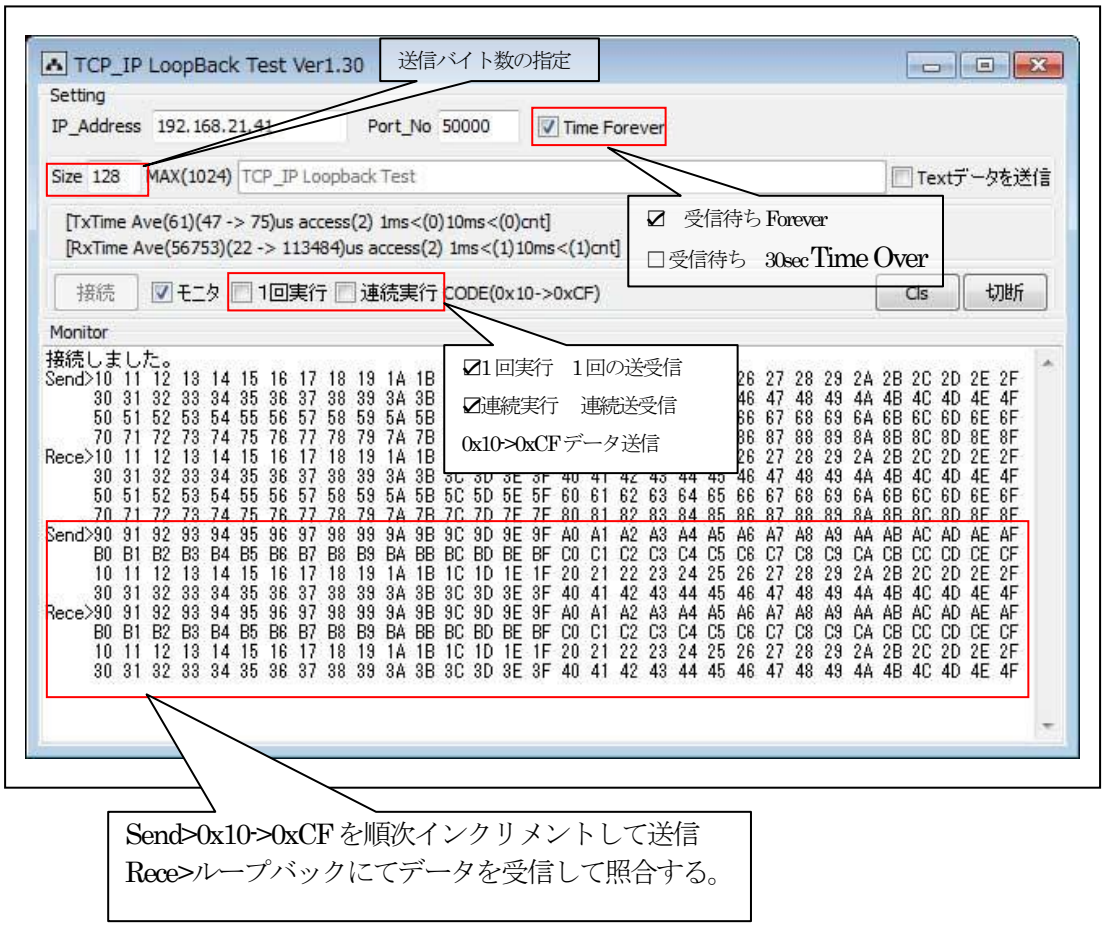
プログラム場所【ご購入 CD¥MCU サンプルプログラム¥_PC_Test】



２）TCP/IP Port を「接続」する。



3）ループバック送受信を確認する。

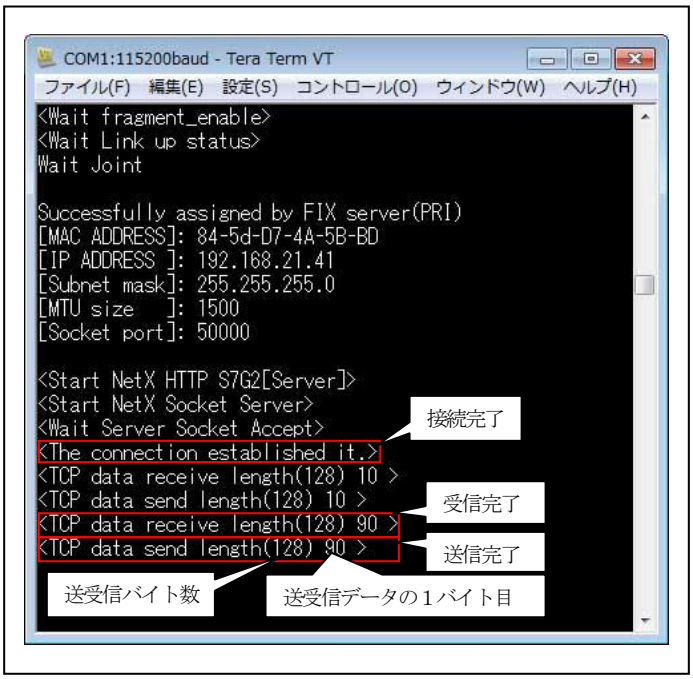


Send>0x10->0xCF を順次インクリメントして送信
Rece>ループバックにてデータを受信して照合する。

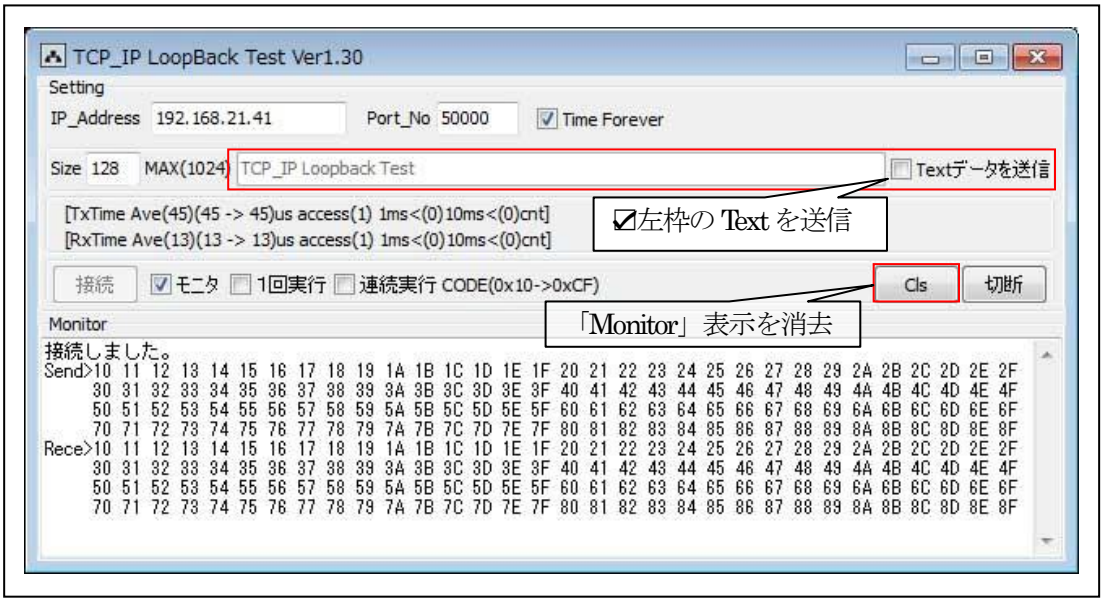
【Error 表示】

・受信待ち 30secTime Over 表示「"Receive Time Over Error !!"」
・送信と受信データの照合エラー表示「" Send <->Rece 照合 Error !!"」

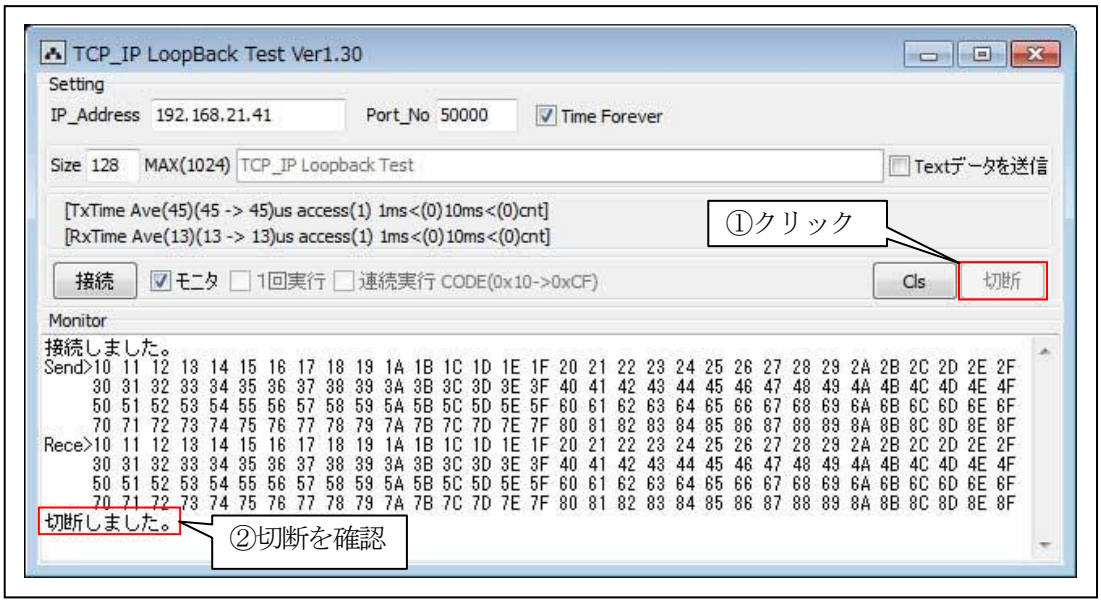
＜Term 画面　MP-S7G2-01 側の表示＞



4）その他の説明



5）TCP/IP Port を「切断」する。



７－４．デバッグの終了

☆詳細操作は「ewarm_synergy_Import.pdf」の３－４項を参照して下さい。

８．注意事項

・本文書の著作権は、エーワン（株）が保有します。
・本文書を無断での転載は一切禁止します。
・本文書に記載されている内容についての質問やサポートはお受けすることが出来ません。
・本文章に関して、ルネサス　エレクトロニクス社への問い合わせは御遠慮願います。
・本文書の内容に従い、使用した結果、損害が発生しても、弊社では一切の責任は負わないものとします。
・本文書の内容に関して、万全を期して作成しましたが、ご不審な点、誤りなどの点がありましたら弊社までご連絡くだされば幸いです。
・本文書の内容は、予告なしに変更されることがあります。

９．商標

・EWARM は、IAR システムズ株式会社の登録商標、または商品名称です。
・Renesas Synergy™および S3A7/S5D9/S7G2 は、ルネサス　エレクトロニクス株式会社の登録商標、または商品名です。
・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。

１０．参考文献

・「S3A7　ユーザーズマニュアル　ハードウェア編」　ルネサス　エレクトロニクス株式会社
・「S7G2　ユーザーズマニュアル　ハードウェア編」　ルネサス　エレクトロニクス株式会社
・ルネサス　エレクトロニクス株式会社提供のサンプル集
・「IDE プロジェクト管理およびビルドガイド」　IAR システムズ株式会社
・「SSP vx.x.x User's Manual」　ルネサス　エレクトロニクス株式会社
・「X-Ware Component Documents for Renesas Synergy™」ルネサス　エレクトロニクス株式会社
・その他

〒486-0852
愛知県春日井市下市場町 6-9-20
エーワン株式会社
https://www.robin-w.com